Aplicativos em PHP

De Wikibooks

28 de junho de 2007

Desenvolvimento de Aplicativos em PHP

For Linux e for Windows

Livro destinado ao desenvolvimento de aplicativos web em PHP.
Traga sua colaboração e façamos algo juntos que seja grande e de muita utilidade para você e para todos os que lidam com desenvolvimento web.

1. Introdução
    1. História
    2. Características e Recursos
2. InteligênciaEmocional
    1. Caindo na Real
3. Instalação e configurações do ambiente
    1. Via Xampp
    2. Individualmente
4. Editores e IDEs para PHP, CSS, JavaScript e HTML
    1. Eclipse com PHPEclipse, Aptana e outros plugins
    2. PHPDesigner
    3. Kate
    4. Dreamweaver
    5. Zend Studio
    6. VS.PHP
    7. Delphi for PHP
    8. KDevelop

    9. PSPAD
    10. Quanta
    11. BlueFish
    12. PHPEdit
5. Referências
    1. Windows
    2. Linux
    3. HTML
    4. JavaScript
    5. CSS
    6. PHP
    7. E-books free
    8. Cursos na área
6. Administração de SGBDs
    1. MySQL
    2. PostgreSQL
    3. SQLite
7. Geradores de Aplicativos
    1. phpCodeGenie (com MySQL)
    2. SQLMaestro (com MySQL, Oracle, MS SQL Server, PostgreSQL, SQLite, Firebird e MaxDB)
    3. phpMyEdit
    4. DadaBik
    5. PHPLibDev
    6. Web Form Generator
8. CMSs (Sistemas Gerenciadores de Conteúdo)
    1. Joomla (Portais)
    2. Mambo (Portais)
    3. Drupal (Portais)
    4. Xoops (Portais)

    5. MediaWiki (wikis)
    6. WordPress (blogs)
    7. eGroupWare (colaboração)
    8. Moodle (eLearning)
9. FrameWorks
    1. P4A
    2. Zend
    3. Seagull
    4. CodeIgniter
    5. QCodo
    6. Zephyr-php
    7. CakePHP
    8. Symfony
10. Abstrações de Bancos de Dados
    1. AdoDB
    2. PEAR/MDB2
11. Trabalhando em PHP com
    1. Formulários
    2. URLs
    3. Funções
    4. Constantes mágicas e super globais
    5. Session
    6. Arquivos e Diretórios
    7. Path
    8. Includes
    9. Formatação da saída
    10. Imagens e Gráficos
    11. Números
    12. Strings

    13. Permissões de arquivos e diretórios
    14. Validação de Dados
    15. Arrays
    16. XML
    17. Cookies

12. Integração do PHP com SGBDs
    1. MySQL
    2. PostgreSQL
    3. SQLite

13. Recursos Extras
    1. Controle de Estoque
    2. Loja Virtual
    3. Código de Barras
    4. Boleto bancário
    5. PHP-GTK
    6. Contadores de visita
    7. Livros de visita
    8. Chats
    9. Suporte Online
    10. Forums
    11. Enquetes
    12. Geradores de Relatórios
    13. Integração com AJAX
    14. Integração com Flash
    15. Integração com Java
    16. Bibliotecas de Funções
    17. Tratamento de Erros
    18. Compilando o PHP

14. Orientação a Objetos

    1. Teoria e exercícios
    2. Aplicativos de exemplo
    3. Ferramentas

15. Aplicativos de Exemplo com
    1. MySQL
    2. PostgreSQL
    3. SQLite

16. Apêndices
    1. Análise e Projeto
    2. Design
    3. Hospedagem e Domínio
    4. Segurança
    5. Documentação
    6. Informações Diversas
    7. Oportunidade de Trabalho
        1. Elaboração de Currículos
        2. Técnicas de Entrevistas
        3. Preços de Serviços
    8. Redes
    9. Editores Gráficos

Contribuidores

- Daniel Ratkevicius (PHP+FLASH) - Contatos: daniel@ratkevicius.com.br[1] ou [2]
- Haroldo Barbosa - Contatos: haroldob@gmail.com[3]
- Jackson Araujo - Contatos: jackson@designrio.com[4]
- Jamires Lontra [Varginha/MG] - Contato: jlontra@etdevarginha.com.br
- Ribamar FS - Contatos: ribafs[ ]users.sourceforge.net [5] e [6].
- William G. Comnisky - Contatos: w.comnisky@gmail.com[7] ou [8]

Retirado de "http://pt.wikibooks.org/wiki/Aplicativos_em_PHP"

Introdução/História

De Wikibooks

Conteúdo

- 1 PHP na Wikipedia
- 2 A História do PHP
  - 2.1 Versão 1
  - 2.2 Versão 2
  - 2.3 Versão 3
  - 2.4 Versão 4
  - 2.5 Versão 5
  - 2.6 Versão 6

PHP na Wikipedia

A linguagem surgiu por volta de 1994, como um subconjunto de scripts Perl criados por Rasmus Lerdof, com o nome PHP/FI (Personal Home Page Tools/Forms Interpreter). Com as adições de Zeev Suraski e Andi Gutmans, dois programadores israelitas pertencentes ao Technion, o Instituto Israelita de Tecnologia, que reescreveram o parser, era lançada em 1997 a PHP 3, primeira versão estável e parecida com a linguagem atual. Ao reescrever o parser, foi criado o Zend Engine, que é mantido oficialmente pela empresa Zend em conjunto com a comunidade PHP. Em Maio de 2000 veio a público a versão 4, e em Julho de 2004, a versão 5, onde a principal mudança foi uma nova API para orientação a objetos provida pelo Zend Engine 2.

Trata-se de uma linguagem extremamente modularizada, o que a torna ideal para instalação e uso em servidores web. Diversos módulos são criados no repositório de extensões PECL (PHP Extension Community Library) e alguns destes módulos são introduzidos como padrão em novas versões da linguagem. É muito parecida, em tipos de dados, sintaxe e mesmo funções, com a linguagem C e com a C++. Pode ser, dependendo da configuração do servidor, embutida no código HTML. Existem versões do PHP disponíveis para os seguintes sistemas operacionais: Windows, Linux, FreeBSD, Mac OS, OS/2, AS/400, Novell Netware, RISC OS, IRIX e Solaris

A Wikipédia funciona sobre um software inteiramente escrito em PHP, usando bases de dados MySQL: o MediaWiki.

Construir uma página dinâmica baseada em bases de dados é simples ,(em parte, vale lembrar), com PHP, este provê suporte a um grande número de bases de dados: Oracle, Sybase, PostgreSQL, InterBase, MySQL, SQLite, MSSQL, Firebird etc, podendo abstrair o banco com a biblioteca

ADOdb, entre outras.

PHP tem suporte aos protocolos: IMAP, SNMP, NNTP, POP3, HTTP, LDAP, XML-RPC, SOAP. É possível abrir sockets e interagir com outros protocolos. E as bibliotecas de terceiros expandem ainda mais estas funcionalidades.

Existem iniciativas para utilizar o PHP como linguagem de programação de sistemas fixos (desktop). A mais notável é a PHP-GTK. Trata-se de um conjunto do PHP com a biblioteca GTK, portada do C++, fazendo assim softwares inter-operacionais entre Windows e Linux.

Segundo o Manual Oficial do PHP

A História do PHP e projetos relacionados

A História do PHP

O PHP percorreu um longo caminho ao nesses poucos anos. Criada para ser uma das mais poderosas e proeminentes linguagens da Web não foi uma tarefa fácil. Aqueles que se interessaram momentaneamente em ver como o PHP cresceu e se tornou o que é hoje, continue lendo. Versões do antigo PHP pode ser encontrado em » PHP Museum.

A História do PHP

Versão 1

PHP/FI

O PHP sucede de um produto mais antigo, chamado PHP/FI. PHP/FI foi criado por Rasmus Lerdorf em 1995, inicialmente como simples scripts Perl como estatísticas de acesso para seu currículo online. Ele nomeou esta série de script de 'Personal Home Page Tools'. Como mais funcionalidades foram requeridas, Rasmus escreveu uma implementação C muito maior, que era capaz de comunicar-se com base de dados, e possibilitava à usuários desenvolver simples aplicativos dinâmicos para Web. Rasmus resolveu » disponibilizar o código fonte do PHP/FI para que todos pudessem ver, e também usá-lo, bem como fixar bugs e melhorar o código.

PHP/FI, que significa Personal Home Page / Forms Interpreter, incluía algumas funcionalidades básicas do PHP que nós conhecemos hoje. Ele usava variáveis no estilo Perl, interpretação automática de variáveis vindas de formulário e sintaxe embutida no HTML. A sua própria sintaxe era similar a do Perl, porém muito mais limitada, simples, e um pouco inconsistente.

Versão 2

Em 1997, PHP/FI 2.0, a segunda versão da implementação C, obteve milhares de usuários ao redor do mundo (estimado), com aproximadamente 50,000 domínios reportando que tinha PHP/FI 2.0 instalado, angariando 1% dos domínios da Internet. Enquanto isto havia milhares de pessoas contribuindo com pequenos códigos para o projeto, e ainda assim

O PHP/FI 2.0 foi oficialmente lançado somente em Novembro de 1997, após perder a maior parte de sua vida em versões betas. Ele foi rapidamente substituído pelos alfas do PHP 3.0.

Versão 3

O PHP 3.0 foi a primeira versão que se assemelha ao PHP que nós conhecemos hoje. Ela foi criada por Andi Gutmans e Zeev Suraski em 1997 e foi totalmente reescrito, após eles descobrirem que o PHP/FI 2.0 poderia ajudá-los a desenvolver suas próprias aplicações de eCommerce de um projeto da Universidade. No esforço cooperativo e iniciativa de começar o PHP/FI à partir da base-usuário existente, Andi, Rasmus e Zeev decidiram cooperare anunciar o PHP 3.0 como uma versão oficial de seu sucessor o PHP/FI 2.0, e o desenvolvimento do PHP/FI 2.0 foram descontinuados.

Uma das maiores características do PHP 3.0 era sua forte capacidade de extensibilidade. Além de oferecer aos usuários finais uma infraestrutura sólida para diversos bancos de dados, protocolos e APIs, o extensibilidade do PHP 3.0 atraí dezenas de desenvolvedores para se juntar e submeter novos módulos. Esta é a chave do tremendo sucesso do PHP 3.0. Outras características chaves introduzidas no PHP 3.0 foram o suporte à sintaxe para orientação à objetos e uma sintaxe muito mais poderosa e consistente.

Toda a nova versão da linguagem foi realizada sob um novo nome, que removeu a impressão do limitado uso pessoal que o PHP/FI 2.0 prendeu. Ela foi nomeada simplesmente 'PHP', com o significado que é um acrônimo - PHP: Hypertext Preprocessor.

No final de 1998, o PHP obteve uma base de dezenas de milhares de usuários (estimativa) e centenas de milhares de Web sites relatando que o tinham instalado. Em seu pico, o PHP 3.0 foi instalado em aproximadamente 10% dos servidores Web da Internet.

O PHP 3.0 foi oficialmente lançado em Junho de 1998, depois de ter passado aproximadamente 9 meses em testes públicos.

Versão 4

O PHP 3.0 foi oficialmente lançado em Junho de 1998, depois de ter passado aproximadamente 9 meses em testes públicos. Os objetivos do projeto eram melhorar a performance de aplicações complexas, e melhorar a modularidade do código base do PHP.Tais aplicações foram possíveis por causa das novas características do PHP 3.0 e o suporte a uma variedade de banco de dados de terceiros e APIs, mas o PHP 3.0 não foi projetado para trabalhar com aplicações muito complexas eficientemente.

A nova engine, dublado 'Zend Engine' (conhecidos pelos seus primeiros nomes, Zeev e Andi), fazendo desse objetivo um sucesso, e foi introduzida em meados de 1999. PHP 4.0, baseado nesta engine, e acompanhado com uma série de novas características, foi oficialmente lançada em Maio de 2000, quase dois anos após o seu predecessor, o PHP 3.0. Além da altíssimo melhoramento da performance desta versão, o PHP 4.0 incluiu outras características chave como o suporte para muitos servidores WEB, sessões HTTP, buffer de saída, maneiras mais seguras de manipular input de usuários e muitas construções novas na linguagem.

PHP 4 é a última versão lançada do PHP. O trabalho já começou na modificação e melhoramento da Zend Engine para integrar novas características que foram designadas para o PHP 5.0.

Hoje, o PHP está começando a ser usado por centenas de milhares de desenvolvedores (estimativa), e muitos milhões de sites reportam que tem o PHP instalado, que explica os 20% de domínios da Internet.

A equipe de desenvolvimento do PHP contém dezenas de desenvolvedores, bem como dezenas de outros que trabalham com projetos relacionados ao PHP como o PEAR e a documentação do projeto.

Versão 5

A última versão 5 estável do PHP é a 5.2.3

Download em http://www.php.net/downloads.php#v5

Notas da versão http://www.php.net/releases/5_2_3.php

Changelog http://www.php.net/ChangeLog-5.php#5.2.3

Versão 6

- Primeiras informações do PHP 6.0 - http://www.robsonjunior.com.br/post/primeiras-informacoes-php6/

- O PHP6 ainda não saiu mas já está causando polêmicas. Veja os artigos:

Prepare for PHP 6 - http://www.corephp.co.uk/archives/19-Prepare-for-PHP-6.html

- Mudanças no PHP6 são polêmicas - http://www.meiobit.com/destaque/mudancas_no_php6_serao_polemicas

- PHP6 - O PRINCIPIO DA MUDANÇA - http://phpexperts.blogspot.com/2007/03/php6-o-principio-da-mudana.html

Introdução/Características e Recursos

De Wikibooks

Segundo a Wikipedia em http://pt.wikipedia.org/wiki/Php#Hist.C3.B3rico

A linguagem PHP é uma linguagem de programação de domínio específico, ou seja, seu escopo se concentra em um campo de atuação que é o Desenvolvimento Web, embora tenha variantes como o PHP-GTK.

Seu propósito é solucionar problemas Webs da maneira mais fácil possível.

Conteúdo

- 1 Características principais:
- 2 O Que o PHP Pode Fazer?
- 3 Esses são os maiores campos onde os scripts PHP podem ser utilizados:
- 4 Suporte a Sistemas Operacionais
- 5 Suporte a Flash, PDF e XML
- 6 Suporte a SGBDs
- 7 Diversos Outros Recursos

Características principais:

```
* Orientação a objeto
* Portabilidade
* Independência de plataforma - "write once run anywhere";
* Tipagem fraca
* Sintaxe similar a Linguagem C/C++ e o PERL
```

Segundo o Manual oficial

O Que o PHP Pode Fazer?

Qualquer coisa. O PHP é focado para ser uma linguagem de script do lado do servidor, portanto, você pode fazer qualquer coisa que outro programa CGI pode fazer, como: coletar dados de formulários, gerar páginas com conteúdo dinâmico ou enviar e receber cookies. Mas o PHP pode fazer muito mais.

Esses são os maiores campos onde os scripts PHP podem ser utilizados:

- Script no lado do servidor (server-side). Este é o mais tradicional e principal campo de atuação do PHP. Você precisa de três coisas para seu trabalho. O interpretador do PHP (como CGI ou módulo), um servidor web e um browser. Basta rodar o servidor web conectado a um PHP instalado. Você pode acessar os resultados de seu programa PHP com um browser, visualizando a página PHP através do servidor web. Todos eles podem rodar na sua máquina, em casa, para você experimentar programação com o PHP. Veja a seção das instruções de instalação para mais informações. - Script de linha de comando. Você pode fazer um script PHP funcionar sem um servidor web ou browser. A única coisa necessária é o interpretador. Esse tipo de uso é ideal para script executados usando o cron ou o Agendador de Tarefas (no Windows). Esses scripts podem ser usados também para rotinas de processamento de texto. Veja a seção Utilizando o PHP em linha de comando para maiores informações. - Escrevendo aplicações desktop. O PHP provavelmente não é a melhor linguagem para criação de aplicações desktop com interfaces gráficas, mas se você conhece bem o PHP, e gostaria de usar alguns dos seus recursos avançados nas suas aplicaçõesdo lado do cliente, você pode usar o PHP-GTK para escrever programas assim. Isso ainda lhe habilita a escrever aplicações multi-plataformas. O PHP-GTK é uma extensão do PHP, não disponibilizada na distribuição oficial. Caso esteja interessado no PHP-GTK, visite o site do projeto (http://www.gtk.org/).

Suporte a Sistemas Operacionais

O PHP pode ser utilizado na maioria dos sistemas operacionais, incluindo Linux, várias variantes Unix (incluindo HP-UX, Solaris e OpenBSD), Microsoft Windows, Mac OS X, RISC OS, e provavelmente outros. O PHP também é suportado pela maioria dos servidores web atuais, incluindo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape and iPlanet Servers, Oreilly Website Pro Server, Caudium, Xitami, OmniHTTPd, e muitos outros. O PHP pode ser configurado como módulo para a maioria dos servidores, e para os outros como um CGI comum.

Com o PHP, portanto, você tem a liberdade para escolher o sistema operacional e o servidor web. Do mesmo modo, você pode escolher entre utilizar programação estrutural ou programação orientada a objeto, ou ainda uma mistura deles. Mesmo sem todos os recursos da POO (Programação Orientada a Objetos) implementados no PHP 4, muitas bibliotecas de código e grandes aplicações (incluindo a biblioteca PEAR) são escritas somente em código POO. O PHP 5 corrige as fraquezas da POO do PHP 4, e introduz um modelo de objetos completo.

Suporte a Flash, PDF e XML

Com PHP você não está limitado a gerar somente HTML. As habilidades do PHP incluem geração de imagens, arquivos PDF e animações Flash (utilizando libswf ou Ming) criados dinamicamente, on the fly. Você pode facilmente criar qualquer padrão texto, como XHTML e outros arquivos XML. O PHP pode gerar esses padrões e os salvar no sistema de arquivos, em vez de imprimi-los, formando um cache dinâmico de suas informações no lado do servidor.

Suporte a SGBDs

Talvez a mais forte e mais significativa característica do PHP é seu suporte a uma ampla variedade de SGBDs. Escrever uma página que consulte um banco de dados é incrivelmente simples. Os seguintes SGBDs são atualmente suportados:

```
Adabas D         InterBase       PostgreSQL
dBase         FrontBase       SQLite
Empress       mSQL    Solid
FilePro (read-only)   Direct MS-SQL   Sybase
Hyperwave     MySQL   Velocis
IBM DB2       ODBC    Unix dbm
Informix      Oracle (OCI7 and OCI8)
Ingres        Ovrimos
```

Também foi providenciada uma abstração de banco de dados DBX permitindo a você utilizar qualquer SGBD transparentemente com sua extensão. Adicionalmente, o PHP suporta ODBC (Open Database Connection, ou Padrão Aberto de Conexão com Bancos de Dados), permitindo que você utilize qualquer outro SGBD que suporte esse padrão mundial.

Diversos Outros Recursos

O PHP também tem suporte para comunicação com outros serviços utilizando protocolos como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (em Windows) e incontáveis outros. Você pode abrir sockets de rede e interagir diretamente com qualquer protocolo. O PHP também suporta o intercâmbio de dados complexos WDDX, utilizado em virtualmente todas as linguagens de programação para web. Falando de comunicação, o PHP implementa a instanciação de objetos Java e os utiliza transparentemente como objetos PHP. Você ainda pode usar sua extensão CORBA para acessar objetos remotos.

O PHP é extremamente útil em recursos de processamento de texto, do POSIX Estendido ou expressões regulares Perl até como interpretador para documentos XML. No processamento de XML, o PHP 4 suporta os padrões SAX e DOM, além de você também poder utilizar a extensão XSL para transformar documentos XML. O PHP 5 padroniza toda a extensão XML a partir da base sólida da

libxml2, além de estender os recursos com o acréscimo ao SimpleXML e XMLReader.

Utilizando o PHP no campo do e-commerce, você poderá usar as funções específicas para Cybescash, CyberMUT, Verysign Payflow Pro e MCVE, práticos sistemas de pagamento online.

Por último mas longe de terminar, temos também outras extensões interessantes: funções para o search engine mnoGoSearch, funções para Gateway IRC, vários utilitários de compressão (gzip, bz2), calendário e conversões de datas, tradução...

Como você pode ver, esta página não é suficiente para descrever todos os recursos e benefícios que o PHP pode oferecer. Leia nas seções sobre a Instalação do PHP, e veja a referência das funções para detalhes das extensões mencionadas aqui.

Popularidade do PHP

Atualmente o PHP é uma das mais populares linguagens de desenvolvimento da Web, provavelmente a mais. Veja as estatísticas neste site: http://www.nexen.net/chiffres_cles/phpversion/.

Vale lembrar que ser a mais popular gera muitas vantagens mas também gera desvantagens. Não é bom ficar se iludindo que apenas por ser a mais popular é a melhor para tudo, pois não é verdade. PHP é uma ótima linguagem, fácil de aprender, conta com muitos recursos já prontos para o programador, tem uma ótima documentação e uma grande comunidade ao redor do mundo, mas está longe de ser perfeita. Sei também que nenhuma é perfeita, mas lembre-se que para que você possa fazer uma avaliação adequada de que é realmente necessário para o que você precisa, deverá conhecer bem a linguagem para só então fazer uma avaliação correta.

InteligênciaEmocional

De Wikibooks

Conteúdo

- 1 O Que PHP tem a ver com Inteligência Emocional?
- 2 Ser Inteligente de Fato
- 3 Falta de Inteligência Emocional
- 4 Valorização do Profissional Inteligente Emocionalmente
- 5 Algumas Idéias do Daniel Goleman
- 6 Como Melhorar a IE?
- 7 Inteligência Emocional na Wikipedia
- 8 Administrando com Inteligência Emocional
- 9 Concluindo sem fechar
- 10 Resumo do Livro Caindo na Real
- 11 Dicas para ser mais criativo
- 12 Regras de Ouro para uma boa Estimulação
- 13 Referências

O Que PHP tem a ver com Inteligência Emocional?

"Até pouco tempo atrás o sucesso de uma pessoa era avaliado pelo raciocínio lógico e habilidades matemáticas e espaciais (QI). Mas o psicólogo Daniel Goleman, PhD, com seu livro "Inteligência Emocional" retoma uma nova discussão sobre o assunto. Ele traz o conceito da inteligência emocional como maior responsável pelo sucesso ou insucesso das pessoas. A maioria das situações de trabalho é envolvida por relacionamentos entre as pessoas. Desta forma pessoas com qualidades de relacionamento humano, como afabilidade, compreensão, gentileza têm mais chances de obter o sucesso."

Essa imagem representa uma forma sadia de crescer na vida e profissionalmente: subir ajudando

outros a subirem. Isso mostra generosidade e sabedoria. Imagine alguém que cresce assim: como será o comportamento do colega que ele ajudou a crescer? Será que ele irá procurar prejudicá-lo em sua carreira.

Agora visualizemos (nem mostrarei imagem :)) alguém que sobe pisando em quem estiver em seu caminho. Será essa a forma mais sábia de subir? Será que os colegas machucados em seu caminho irão ajudá-lo a continuar subindo?

Acontece que todo programador é um ser humano, com sentimentos e emoções como qualquer outro. É importante lembrar que esse lado do programador é muito importante em seu desempenho final e dependendo de como lidarmos com nossas emoções poderemos ser mais ou menos bem sucedidos em nossos empreendimentos.

Geralmente qualquer especialista considera-se superior aos demais por ter um bom domínio de uma área que os demais não conhecem. Isso o leva a menosprezar outros conhecimentos que não sejam os de sua área. Com isso fecha-se deixando de aprender muitas coisas importantes e isola-se em seus relacionamentos.

Ser Inteligente de Fato

Para sermos realmente inteligentes devemos ser sensíveis a todas as áreas de atuação humano e até além delas (já que existem áreas que estão além dos conhecimentos, como a criatividade, a estética, etc). Particularmente a área psicológica deve ser levada a sério. Em nossa cultura praticamente não se fala nem se estuda sobre nosso interior, nossas emoções e cia.

Falta de Inteligência Emocional

De que adianta ter um grande potencial, ter muito conhecimento, ser muito inteligente e criativo em algumas áreas e não conseguir conviver bem com suas emoções e sair criando confusão a todo o momento com clientes, colegas de trabalho, familiares, etc? Para realmente usufruir do nosso potencial e ainda por cima melhorá-lo a cada dia devemos aprender a lidar da melhor maneira possível com nossas emoções e sentimentos.

Valorização do Profissional Inteligente Emocionalmente

Atualmente as grandes e organizadas empresas estão dando muita importância para algo que se está chamando de Inteligência Emocional, que é esta capacidade de lidar bem com o lado emocional, especialmente para os cargos mais importantes das empresas.

Algumas Idéias do Daniel Goleman

Existe polêmica sobre quem é o pai da inteligência emocional mas quanto ao seu principal divulgador não existe dúvida, é o psicólogo Daniel Goleman.

Algumas habilidades emocionais são consideradas importantes para que uma pessoa alcance seus objetivos, seja feliz e alcance sucesso na vida. Dentre elas são citadas o controle do temperamento, adaptabilidade, persistência, amizade, respeito, amabilidade e empatia. Goleman apresenta os seguintes níveis de Inteligência Emocional:

- Auto-conhecimento emocional - Autoconsciência: conhecimento que o ser humano tem de si próprio, de seus sentimentos ou intuição. Esta competência é fundamental para que o homem tenha confiança em si (autoconfiança) e conheça seus pontos fortes e fracos;

- Controle emocional - Capacidade de gerenciar os sentimentos: é importante saber lidar com os sentimentos. A pessoa que sabe controlar seus próprios sentimentos se dá bem em qualquer lugar que esteja ou em qualquer ato que realize.

- Auto- motivação - Ter vontade de realizar, otimismo: Pôr as emoções a serviço de uma meta. A pessoa otimista consegue realizar tudo que planeja pois tem consciência que todos os problemas são contornáveis e resolvíveis.

- Reconhecer emoções nos outros - Empatia: saber se colocar no lugar do outro. Perceber o outro. Captar o sentimento do outro. A calma é fundamental para que isso aconteça. Os problemas devem ser resolvidos através de conversas claras. As explosões devem ser evitadas para que não prejudique o relacionamento com os outros.

- Habilidade em relacionamentos inter-pessoais - Aptidão social: a capacidade que a pessoa deve ter para lidar com emoções do grupo. A arte dos relacionamentos deve-se, em grande parte em saber lidar com as emoções do outro. Saber trabalhar em equipe é fundamental no mundo atual.

Os sentimentos mais fortes do homem são a tristeza, a alegria e a raiva. É fundamental saber lidar com eles. As pessoas que sabem controlar suas emoções são aquelas que obtém mais sucesso na vida, em qualquer tipo de medição, provas de vestibular, etc.

Segundo Goleman: "emoções são sentimentos a se expressarem em impulsos e numa vasta gama de intensidade, gerando idéias, condutas, ações e reações. Quando burilados, equilibrados e bem-conduzidos transformam-se em sentimentos elevados, sublimados, tornando-se, aí sim – virtudes."

Como Melhorar a IE?

Vale ressaltar que não ajuda ficar dizendo ou pensando: "Eu sou assim mesmo e não posso mudar". O primeiro passo é perceber e depois admitir que existe algo em você que precisa melhorar. Depois disso é procurar se informar ao máximo sobre o assunto e ter consciência de que pode e deve fazer algo para mudar. Fique atento, observe a si mesmo nas relações, procure perceber por que age e reage assim. Sempre que perceber uma reação que deseja mudar, pare, perceba e procure mudar. Não

condene a si por ser como é, isso de nada ajuda. Encare os fatos:existe algo em você que pretende mudar e é isso que irá fazer. Não desanime com as dificuldades, lembre que está fazendo algo que o tornará melhor, melhor profissional, mais valorizado, mais importante para os colegas, para a empresa e para você também. Alguém melhor.

Inteligência Emocional na Wikipedia

Na psicologia, inteligência emocional é um tipo de inteligência que envolve as habilidades para perceber, entender e influenciar as emoções. Foi introduzida e definida por John D. Mayer e Peter Salovey. Inteligência emocional, chamada também EI é medida frequentemente como um Quociente de inteligência Emocional ou um QE emocional, descrevem uma habilidade, uma capacidade, ou uma habilidade de perceber, para avaliar e controlar as emoções de si mesmo, de outro, e dos grupos. Entretanto, sendo uma área relativamente nova, a definição da inteligência emocional está ainda em um estado do fluxo. Alguns, tais como John D. Mayer (2005a) preferem distinguir o conhecimento emocional da inteligência emocional, como discutido abaixo.

Administrando a IE

Em empresas ultrapassadas, o executivo que não se deixava levar pelo lado emocional em relaçãoaos seus funcionários era valorizado, pela simples crença de que seria impossível lidar com as pessoas em situações que exigissem uma atitude mais drástica. Pensava-se que usar o coração no lugar da cabeça iria causar conflitos entre interesses pessoais e as metas organizacionais. As decisões duras não poderiam ser tomadas sem um distanciamento afetivo, mesmo havendo a probabilidade de tomá-las de modo mais humano. Entretanto, esse tipo de profissional é, claramente, do passado. **O executivo atual é aquele que possui uma elevada habilidade de relacionamento inter-pessoal**. **Um administrador mais humano é extremamente necessário nos dias atuais e futuramente ainda mais**. **O custo-benefício da utilização da IE nas empresas é uma idéia nova, porém muito evidente. O administrador deve proporcionar abertura para sugestões, saber criticar pela melhor forma, assim como elogiar, deve também saber como dissuadir preconceitos e trabalhar em equipe. O ambiente de trabalho não deve ser um ambiente de discórdia, intriga, sem abertura e amedrontador.**

Administrando com Inteligência Emocional

Começarei pelo 1º fator importante na administração da IE:

- **O feedback** Como é o melhor meio de se aplicar o feedback? O feedback não pode ser postergado. Uma vez que ele não é efetuado enquanto o assunto está "quente", ele acaba não sendo mais tão específico, fica acumulado e acaba sendo aplicado de forma ampla e destrutiva.

- **Ser específico**. É muito importante utilizar a especificidade tanto na crítica, como no elogio. O foco deve ser somente no problema e não no trabalho inteiro.

- **Oferecer uma solução**. A crítica deve sempre ser acompanhada de uma sugestão de como melhorar. Sem uma solução, a crítica fica vazia e acaba atingindo o indivíduo pessoalmente, deixando-o desmoralizado.

- **Fazer a crítica pessoalmente**. Não deve-se criticar ou elogiar por outro meio que não seja cara a cara. Este é o meio mais franco de se passar alguma informação, além de estar fornecendo a oportunidade de esclarecimentos.

- **Seja sensível**. Tente ser empático com seus comandados, você deve possuir uma sintonia emocional com as pessoas, sinta o que elas precisam, o que querem dizer, ou seja, o que elas estão sentindo.

Concluindo sem fechar

Lembre que você vive no mundo em constante relacionamento com outros seres humanos que poderão te ajudar ou te prejudicar, dependendo muito de como você os trata. Não basta apenas ajudar com a intenção de ser ajudado ou de evitar ser prejudicado, é importante que realmente sinta a vontade de colaborar, de melhorar a vida de outras pessoas. Mesmo quando estiver pretendendo tirar proveito, obter lucro, mesmo nestas situações, tenha em mente que o seu serviço ou produto irá ajudar a melhorar a vida de outras pessoas. Ganhar dinheiro não deve ser o objetivo principal, mas uma conseqüência de um trabalho, ou serviço, bem feito e que de fato ajuda outras pessoas.

Resumo do Livro Caindo na Real

Apenas clique no linque acima para ler o resumo.

Dicas para ser mais criativo

- Nunca se contente com a primeira idéia que lhe ocorrer. Busque outras para, entre muitas, escolher a melhor

- Não se acomode. Sempre existe uma maneira de fazer melhor, mais rápido ou com menor custo aquilo que você já faz. Se você não pensar nisso, alguém irá pensar

- Seja curioso. Evite reproduzir tarefas mecanicamente. Busque as causas, os porquês, as implicações. Muitas idéias surgem daí

- Idéias não saem do nada. Associe, adapte, substitua, modifique, reduza. As combinações são infinitas

- Não acredite em bordões como "isso nunca vai funcionar" ou "em time que está ganhando não se

mexe". O novo sempre assusta. Toda idéia tem de quebrar resistências

- Tenha iniciativa. Muitas boas idéias acabam no fundo da gaveta porque seus autores não tomam a decisão de mostrá-las aos outros

- Ouça os outros. Principalmente se eles pensam diferente de você. As idéias se desenvolvem com a divergência

- Faça de vez em quando coisas que contrariem seus hábitos, no trabalho ou no lazer. Por exemplo: se você gosta de filmes de ação, assista a um drama romântico. Se é fã de rock, tente o jazz. Sair da rotina é sempre estimulante para o cérebro

---

Esta mensagem pode ser encontrada no site "Contando Histórias", no endereço http://www.contandohistorias.com.br/historias/2004097.php

---

Veja também:

-- Um aluno diferente -- [http://www.contandohistorias.com.br/historias/2006195.php]

-- O humor cura -- [http://www.contandohistorias.com.br/historias/2006089.php]

-- A cobra e o vagalume -- [http://www.contandohistorias.com.br/historias/2004007.php]

-- As estatísticas do amor -- [http://www.contandohistorias.com.br/historias/2004047.php]

-- Amizade -- [http://www.contandohistorias.com.br/historias/2004028.php]

Regras de Ouro para uma boa Estimulação

Do site Portal da Família - http://www.portaldafamilia.org/artigos/artigo161d.shtml

Recomendações para pais mas boa parte se aplica a praticamente todas os casos.

A mente bem desenvolvida, a paixão por aprender e a habilidade de fazer o conhecimento funcionar são as novas chaves para o Futuro.

```
Regras de ouro para uma boa estimulação

 - Deve ser nova;
 - Interessante;
 - Alegre;
 - Variada;
 - Curta, que não cansa;
 - Lúdica: seu objetivo é brincar aprendendo;
 - Do mais fácil ao mais complicado;
 - Empregando uma pedagogia positiva, reforçando seus êxitos;
 - O melhor presente é um sorriso de seus pais ou uma carícia;
 - Sem perguntas de nenhum tipo – não se fazem exames;
```

```
- Os “melhores jogos” de estimulação são vocês mesmos.

* Para aprofundar no assunto, procure ler:

- “Como multiplicar a inteligência do seu bebê” – Glenn Doman
- “Como ensinar seu bebê a ler” – Glenn Doman
- “Como ensinar matemática a seu bebê” – Glenn Doman
    Editora Artes e Ofícios
```

Referências

- Bom site com artigos e links sobre Inteligência Emocional - http://www.din.uem.br/ia/emocional/

- Teste de Inteligência Emocional - http://www.cpsimoes.net/testeqie/testeqie.html

- Artigos sobre Inteligência Emocional - http://www.seducaoecomportamento.com.br/autodes06.htm

- Inteligência Emocional na Sala de Aula - http://www.centrorefeducacional.com.br/intemoci.htm

- Entrevista com Daniel Goleman - http://www.abrae.com.br/entrevistas/entr_gol.htm

- Um Estudo de Validade sobre a Capacidade de Perceber Emoções - http://www.scielo.br/pdf/prc/v16n2/a08v16n2.pdf

- AUTODIAGNÓSTICO: TESTE SUA INTELIGÊNCIA EMOCIONAL - http://www.guiarh.com.br/z84.htm

InteligênciaEmocional/Resumo do Livro Caindo na Real

De Wikibooks

Resumo do Livro "Caindo na Real"

O que é Caindo na Real?

Quer construir uma aplicação web de sucesso? Então é hora de Cair na Real. Caindo na Real é o menor, mais rápido e melhor caminho para construir software.

- Caindo na Real é sobre pular todas as coisas que 'não' * representam a realidade (cartas, gráficos, caixas, setas, esquemas, wireframes, etc.) e **realmente construir a coisa real**.

- Caindo na Real é menos. Menos massa, menos software, menos funcionalidades, menos papéis, **menos tudo que não é essencial** (e a maioria do que você pensa ser essencial realmente não é).

- Caindo na Real é **permanecer pequeno e ser ágil**.

- Caindo na Real **inicia com a construção da interface**, ou seja, as telas reais que as pessoas irão utilizar. Começa com as experiências reais dos clientes, construindo a partir disso para trás. Dessa forma você obtém a interface adequada antes de obter um software errado.

- Caindo na Real é sobre **iterações e baixar os custos da mudança**. Caindo na Real tem tudo a ver com lançamento, refinamento e melhorar constantemente, o que o torna o caminho perfeito para software baseado em web.

- Caindo na Real **entrega exatamente o que os clientes precisam** e elimina qualquer coisa que não precisam.

* adicionei por considerar que faltava

Os benefícios de Caindo na Real

Caindo na Real entrega melhores resultados porque o força a lidar com os problemas reais que está tentando resolver em vez de suas idéias sobre esses problemas. Ele o força a lidar com a realidade.

Caindo na Real pula especificações funcionais e outras documentações transitórias em favor de construir telas reais. Uma especificação funcional é para inglês ver, uma ilusão de um acordo, enquanto uma página web pronta é realidade. É isso que seus clientes irão ver e usar. É isso que importa. Caindo na Real o leva lá mais rápido. E isso signfica que está tomando decisões de software

baseado na coisa real em vez de noções abstratas.

Finalmente, Caindo na Real é a maneira que se encaixa idealmente para software baseado em web. O modelo convencional de entregar software em uma caixa e então esperar um ano ou dois para entregar uma atualização está desaparecendo. Diferente de software instalado, aplicações web podem evoluir constantemente de maneira diária. Caindo na Real abre essa vantagem por tudo que ele vale.

Como Escrever Software Vigoroso

Escrita vigorosa é concisa. Uma sentença não deve conter palavras desnecessárias, um parágrafo não deve conter sentenças desnecessárias, pela mesma razão que desenhar não deve ter linhas desnecessárias e uma máquina não deve ter partes desnecessárias. Isso requer não que o escritor torne todas as sentenças curtas ou evite todos os detalhes e trate os assuntos apenas em ítens, mas sim que cada palavra fale. --De "Os Elementos de Estilo" de William Strunk Jr.

Números de versão? Jogue pela janela. Você precisa construir, lançar e refinar. Então recomece e repita.

Acreditamos que software é muito complexo. Funcionalidades demais, botões demais, coisa demais para aprender. Nossos produtos fazem menos do que a concorrência -- intencionalmente. Construímos produtos que funcionam de forma mais esperta, que parecem melhor, que lhe permitem fazer suas coisas e são mais fáceis de usar.

O primeiro passo é quebrar em pequenas unidades. Quando existem pessoas demais envolvidas, nada acontece. Quanto mais enxuto você for, mais rápido – e melhor – as coisas acontecem.

Lance menos funcionalidades, mas de qualidade. Você não precisa usar a forma big bang com todo novo lançamento e amontoados de funcionalidades. Dê aos usuários pedaços minúsculos que eles possam digerir.

Construa software para você mesmo

Uma grande maneira de escrever software é começar resolvendo seus próprios problemas. Você será o público-alvo e saberá o que é importante e o que não é. Isso lhe dá um bom adiantamento na entrega de um produto fora de série.

A chave aqui é entender que não está sozinho. Se estiver tendo problemas, é provável que centenas de milhares de outras pessoas estão no mesmo barco. Esse é seu mercado. Não foi fácil?

Basecamp se originou em um problema: como uma empresa de design precisávamos de uma maneira simples de comunicar nossos clientes sobre os projetos. Começamos fazendo isso através da extranet dos clientes, que atualizávamos manualmente. Mas modificar o HTML na mão toda vez que o projeto precisava ser atualizado simplesmente não estava funcionando. Esses sites de projetos sempre

pareciam ficar travados e eventualmente eram abandonados. Era frustrante porque nos deixava desorganizados e deixava os clientes no escuro.

Então começamos a procurar outras opções. Ainda assim cada ferramenta que encontrávamos ou 1) não fazia o que precisávamos ou 2) era gorda de funcionalidades que não precisávamos – como cobrança, controles estritos de acesso, planilhas, gráficos, etc. Sabíamos que deveria haver uma maneira melhor então decidimos construir nossa própria.

Quando resolvemos nossos próprios problemas, criamos uma ferramenta que nos apaixona. E paixão é a chave. Paixão significa que realmente a usaremos e cuidaremos dela. E essa é a melhor maneira de fazer os outros se sentirem apaixonados sobre ela também. Arranhando sua própria coceira

O mundo de Código Aberto abraçou esse mantra há muito tempo – eles chamam de "arranhando sua própria coceira". Para os desenvolvedores de código aberto, significa que terão as ferramentas que querem, entregues da maneira que querem. Mas os benefícios vão mais a fundo.

Como designer ou desenvolvedor de uma nova aplicação, você precisa encarar centenas de micro-decisões todos os dias: azul ou verde? Uma tabela ou duas? Estática ou dinâmica? Abortar ou recuperar? Como tomamos essas decisões? Se é algo que reconhecemos como importante, poderíamos perguntar. O resto, chutamos. E todos esses chutes constroem um tipo de débito em nossas aplicações – uma rede interconectada de coisas que assumimos.

Como um desenvolvedor, detesto isso. O conhecimento de todas essas bombas-relógio em pequena escala nas aplicações que escrevo somam-se ao meu stress. Desenvolvedores de código aberto, arranhando suas próprias coceiras, não sofrem isso. Porque eles são seus próprios usuários, eles sabem a resposta correta para 90% das decisões que precisam tomar. Acho que é uma das razões que as pessoas chegam em casa após um dia duro de trabalho de codificação e ainda trabalham com código aberto: é relaxante.

— Dave Thomas, The Pragmatic Programmers

Você precisa de importar sobre isso

Quando você escreve um livro, precisa de mais do que uma história interessante. Precisa ter um desejo de contar a história. Precisa investir pessoalmente de alguma maneira. Se vai viver com alguma coisa por dois anos, três anos, o resto de sua vida, precisa se importar sobre isso. ""Precisa sentir paixão"". —Malcolm Gladwell, autor (de Algumas Finas Fatias de Malcolm Gladwell)

Dinheiro de fora é plano B

A primeira prioridade de muitas empresas iniciantes é adquirir fundos de investidores. Mas lembre-se, se nos viramos para gente de fora para fundos, teremos que responder a eles também. Crescem expectativas. Investidores querem seu dinheiro de volta – e rapidamente. O fato triste é que dinheiro

entrando nem sempre significa a construção de um produto de qualidade.

Atualmente não é preciso muito para começar. Hardware é barato e uma boa parte de grandes softwares de infra-estrutura são código aberto e de graça. E paixão não vem com uma etiqueta de preço.

Então faça o que puder com o dinheiro que tem em mãos. Pense muito e determine o que é realmente essencial e o que pode viver sem. O que pode fazer com três pessoas em vez de dez? O que pode fazer com R$ 40 mil em vez de R$ 200 mil? O que pode fazer em três meses em vez de seis? O que pode fazer se puder manter seu emprego e construir sua aplicação nas horas vagas?

Restrições forçam a criatividade

Dirija com recursos limitados e será forçado a contar com restrições mais cedo e mais intensamente. E isso é uma coisa boa. Restrições dirigem inovação.

Um retorno rápido é bem improvável. Então foque em construir uma ferramenta de qualidade que você e seus clientes poderão viver com por um bom tempo.

Nunca jogue mais tempo ou dinheiro em um problema, apenas diminue o escopo.

'O mais tarde é eterno, o agora está voando.'

Lançar alguma coisa grande que está um pouco menor em escopo do que o planejado é melhor do que lançar alguma coisa medíocre e cheio de buracos porque precisou atingir uma janela mágica de prazo, orçamento e escopo.

Agora, com tudo isso dito, também é importante não ficar muito obcecado com a concorrência. Analise demais outros produtos e você vai começar a limitar sua maneira de pensar. Dê uma olhada e vá em frente para sua própria visão e suas próprias idéias. Se sua aplicação não o excita, algo está errado. Se está trabalhando nela apenas para ganhar dinheiro, isso vai aparecer. Da mesma forma, se você se sentir apaixonado pela aplicação, também vai aparecer no produto final. As pessoas conseguem ler nas entrelinhas.

Entusiasmo se manifesta prontamente, claro, mas indiferença é igualmente inesquecível. Se seu compromisso não vem com paixão genuína para o trabalho às mãos, isso se torna um vazio que é quase impossível de conciliar, não importa o quão elaborado ou atrativo é o design. —Khoi Vinh, Subtraction.com

Quanto mais enxuto for, mais fácil é para mudar

Quanto mais massa tiver um objeto, mais energia é necessária para mudar sua direção. É uma verdade tanto para o mundo dos negócios como para o mundo físico.

'Deixe as limitações lhe guiar para soluções criativas'

Nunca há suficiente para dar a volta. Sem tempo suficiente. Sem dinheiro suficiente. Sem pessoal suficiente.

Isso é uma coisa boa.

Em vez de se desesperar com essas restrições, aceite-as. Deixe que elas o guiem. Restrições incentivam inovação e forçam o foco. Em vez de tentar removê-las, use-as em seu benefício.

Diferencie-se das companhias maiores sendo amigável e pessoal

Muitas pequenas empresas cometem o erro de tentarem atuar grande. É como se elas entendessem seu tamanho como uma fraqueza que precisa ser encoberta. Muito ruim. Ser pequeno pode realmente ser uma grande vantagem, especialmente quando isto representa comunicação.

Pequenas empresas gostam de menos formalidades, menos burocracia e mais liberdade. Menores empresas são mais próximas dos clientes por padrão. Isto significa que elas podem se comunicar com seus clientes de forma mais direta e pessoal. Se a empresa é pequena, pode-se usar uma linguagem familiar ao invés de jargão. Seu site e seu produto podem ter uma voz humana ao invés de soar como um zumbido corporativo. Ser pequeno significa poder falar com os clientes, e não "se submeter a eles."

Sempre disponível

Não importa em qual negócio você está, um bom serviço ao cliente tornou-se o maior requisito que qualquer cliente estabelecerá. Nós demandamos isso dos serviços que usamos então por que com nossos clientes seria diferente? Desde o começo nós deixamos fácil e transparente para nossos clientes contatar-nos por toda e qualquer questão que tiverem. Em nosso website nós listamos um grande número de ferramentas gratuitas que redireciona para nossos celulares e nossos cartões de visita listam os números de cada um de nós. Nós enfatizamos para nossos consumidores que eles podem nos contatar a qualquer hora independente do problema. Nossos clientes apreciam essenível de confiança ninguém jamais abusou deste serviço. —Edward Knittel, Diretor de Vendas e Marketing, KennelSource

Faça um Mantra

Organizações precisam de pontos-guia. Precisam de linhas gerais; funcionários precisam saber a cada dia quando acordam porque estão indo trabalhar. Essas linhas devem ser curtas e doces, e bem compreensivas: Por que você existe? O que o motiva? Chamo isso de mantra – uma descrição de três ou quatro palavras de porque você existe. —Guy Kawasaki, autor (de Make Mantra)

Sucesso e satisfação estão nos detalhes

Entretanto, o sucesso não é a única coisa que encontrará nos detalhes. Também encontrará estagnação, desacordo, reuniões e atrasos. Essas coisas podem acabar com a moral e diminuir suas chances de sucesso.

Quantas vezes se encontrou travado em um único design ou elemento de código por um dia inteiro? Quantas vezes se deu conta de que o progresso que fez hoje não foi progresso real? Isso acontece quando você foca nos detalhes cedo demais no processo. Há tempo suficiente para ser um perfeccionista. Apenas faça isso mais tarde.

Não se preocupe com o tamanho da fonte do cabeçalho na primeira semana. Você não precisa empregar o tom perfeito de verde na segunda semana. Não precisa mover em três pixels o botão de "submeter" na terceira semana. Apenas coloque as coisas na página por enquanto. Então use. Garanta que funciona. Mais tarde você pode ajustar e aperfeiçoar.

Os detalhes se revelam ao se usar o que está construindo. Você verá o que precisa de mais atenção. Sentirá o que está faltando. Saberá quais crateras pavimentar porque ficará sempre caindo nelas. É quando precisa prestar atenção, e não antes.

O Diabo está nos Detalhes

Quase me cansei da atitude "entre nos detalhes imediatamente" depois de tomar algumas aulas de desenho … Se começar a desenhar os detalhes imediatamente pode ter certeza que o desenho será uma droga. De fato, você está perdendo completamente o ponto.

Você deve começar pegando as proporções corretas da cena toda. Então rascunha os grandes objetos na sua cena, indo até os menores. O rascunho deve ser bem vago nesse ponto. Então pode proceder sombreando, o que consiste em dar volume à vida. Você começa com apenas três tons (claro, médio, escuro). Isso dá um rascunho de tons. Então, para cada porção do seu desenho reavalia três tons e os aplica. Faça isso até os volumes aparecerem (requer múltiplas iterações) ...

Funciona do grande para o pequeno. Sempre. — Patrick Lafleur, Creation Object Inc. (de Signal vs. Noise)

Faça Software que tem Opinião

Seu aplicativo deve tomar partido

Algumas pessoas defendem que o software deve ser agnóstico. Dizem que é arrogante da parte dos desenvolvedores limitar a funcionalidade ou ignorar pedidos de novos recursos. Dizem que o software deve ser sempre o mais flexível possível.

Para nós isso é papo-furado. O melhor software traz consigo uma visão. O melhor software toma partido. Quando alguém usa um software, não está procurando apenas recursos, está procurando uma abordagem. Está procurando uma visão. Decida qual é sua visão e atenha-se a ela.

E lembre, se não gostarem da sua visão há um monte de outras visões por aí. Não corra atrás de quem você nunca irá contentar.

Um ótimo exemplo é o projeto original do wiki. Ward Cunningham e seus amigos deliberadamente desproveram o wiki de muitos recursos que no passado eram considerados parte indispensável da colaboração de documentos. Em vez de atribuir cada mudança do documento a uma pessoa determinada, eles removeram muito da representação visual de propriedade. Eles tornaram o conteúdo atemporal e destituído de ego. Eles decidiram que não importava quem escreveu o conteúdo ou quando ele foi escrito. E isso fez toda a diferença. Essa decisão despertou nas pessoas um senso de comunidade e foi peça-chave no sucesso da Wikipédia.

Nossos aplicativos trilharam um caminho parecido. Eles não tentam ser todas as coisas para todas as pessoas. Eles têm uma atitude. Eles vão atrás de clientes que são no fundo parceiros. Eles têm apelo para as pessoas que partilham de nossa visão. Ou se está do lado de dentro ou se está do lado de fora.

'*Comece com Não*'

Cada vez que você diz sim para uma funcionalidade, você está adotando um filho. Você tem que levar seu bebê através de toda uma cadeia de eventos (exemplo: design, implementação, testes etc.). Uma vez que está funcionalidade está lá, você está preso a ela. Apenas tente removê-la e veja o quão irados ficarão os clientes. Não concorde com tudo

Faça com que cada funcionalidade dê duro para ser implementada. Ponha cada uma delas à prova e mostre que é uma sobrevivente. É como no filme "O Clube da Luta". Você deveria considerar apenas funcionalidades que estejam dispostas a ficar aguardando na porta por três dias para serem aceitas.

É por isso que você tem que começar com um não. Cada novo pedido de funcionalidade que vem até nós – ou de nós – encontra um não. Nós ouvimos mas não agimos. A resposta inicial é "agora não". Se o pedido continua a aparecer, então sabemos que é hora de um olhar mais profundo. Somente então nós começamos a pensar na funcionalidade de fato.

E o que dizer às pessoas que reclamam quando nós não adotamos a sua idéia? Lembre-os do porque eles gostam da aplicação em primeiro lugar. "Você gosta dele porque nós dizemos não. Você gosta dele porque ele não faz outras 100 coisas. Você gosta dele porque ele não tenta agradar a todos sempre."

Crie algo que você possa gerenciar

Deixe os clientes informarem o que é importante

Os clientes querem absolutamente tudo. Eles virão com uma avalanche de pedidos de funcionalidades. Dê uma olhada nos fóruns de nossos produtos; A categoria 'pedido de funcionalidade' sempre sobrepuja as com larga vantagem.

Nós vamos ouvir sobre "essa pequena funcionalidade extra" ou "não pode ser difícil" ou "não seria fácil colocar isso" ou "vai levar apenas uns segundos para inserí-la" ou "se você adicionar isso, eu pagaria o dobro" e assim por diante.

Claro que não podemos culpar as pessoas por pedir funcionalidades. Nós as encorajamos e queremos ouvir o que elas tem a dizer. A maior parte das funcionalidades que inserimos em nossos produtos começaram como sugestões de nossos clientes. Mas, como dissemos antes, sua primeira resposta deve ser um não. Então o que você faz com todos esses pedidos? Onde você os guarda? Como você os gerencia? Você não faz isso. Você apenas os lê e então os joga fora.

Sim, leia, jogue fora e esqueça-os. Pode soar como heresia mas os realmente importantes irão, com certeza, reaparecer. Esses são os únicos que você precisa se lembrar. Esses são os realmente esseciais. Não se preocupe em organizar e guardar cada pedido que aparecer. Deixe seus clientes serem sua memória. Se a funcionalidade for realmente necessária, eles te lembrarão até que você não consiga esquecer.

Da Idéia à Implementação

Vá do brainstorm à esboços à HTML à codificação

Aqui vai o processo que usamos para Cair na Real:

Brainstorm

Traga idéias à tona. O que este produto irá fazer? Para o Basecamp, nós olhamos para nossas próprias necessidades. Queríamos publicar atualizações de projeto. Queríamos participação dos clientes. Sabíamos que projetos tinham datas-chave. Queríamos centralizar arquivos para que as pessoas pudessem revisar coisas antigas com facilidade. Queríamos ter uma visão da figura maior, uma vista aérea do que estava acontecendo com todos os nossos projetos. Juntas, estas premissas e algumas outras, serviram como nossa fundação.

Esse estágio nao é sobre os mínimos detalhes. É sobre grandes questões. O que a aplicação precisa fazer? Como saberemos quando será útil? O que exatamente faremos? Isso é sobre idéias de alto nível, nao discussões no nível dos pixels. Nesse estágio, esses tipos de detalhe simplesmente não têm sentido. Papel de Padeiro

Esboços são rápidos, sujos e baratos e é exatamente como você quer começar. Desenhe coisas. Rabisque coisas. Caixas, círculos, linhas. Arranque as idéias da cabeça para o papel. O objetivo nesse ponto deve ser converter conceitos em designs grosseiros de interface. Esse passo é apenas sobre

experimentação. Não há respostas erradas. Crie telas HTML

Faça uma versão HTML dessa funcionalidade (ou seção, ou fluxo, se for mais apropriado). Pegue algo real e publique para que todos possam ver como fica na tela.

Para o Basecamp, primeiro fizemos a tela de "postar mensagens", então a tela de "editar mensagens" e a coisa prosseguiu daí.

Não escreva nenhum código de programação ainda. Apenas faça um protótipo em html e css. A implementação vem depois.

Codifique

Quando o protótipo parecer bom e demonstrar o suficiente das funcionalidades necessárias, vá em frente e conecte o código de programação.

Durante todo esse processo, se lembre de permanecer flexível e esperar múltiplas iterações. Você deve se sentir livre para jogar fora qualquer parte entregável de qualquer passo particular e começar novamente se ela se mostrar lixo. É natural passar por esse ciclo múltiplas vezes.

Teste sua aplicação com uso do mundo real

Não tenha reuniões

Você precisa mesmo de reuniões? Reuniões geralmente acontecem quando um conceito não está claro o suficiente. Ao invés de recorrer a uma reunião, tente simplificar o conceito, para que você possa discutí-lo rapidamente por email ou IM ou Campfire. O objetivo é evitar reuniões. Cada minuto que você gasta em uma reunião é um minuto que você poderia estar trabalhando.

Não existe nada mais tóxico à produtividade do que uma reunião. Aqui vão alguns motivos:

- Elas quebram seu trabalho diário em pequenos períodos, que acabam por quebrar o fluxo do trabalho

- Elas geralmente tratam apenas de palavras e conceitos abstratos, não de coisas reais (como um trecho de código ou algum detalhe do design de interface)

- Elas geralmente tratam de uma pequena quantidade de informações por minuto

- Elas quase sempre tem uma pessoa que inevitavelmente vai fazer com que todos percam o tempo com assuntos não relacionados

- O assunto principal vai embora muito facilmente

- Freqüentemente tem pautas tão vagas que ninguém tem certeza do assunto principal

- Requerem uma preparação prévia, que quase ninguém faz

Em casos em que reuniões são realmente necessárias (faça disso um raro evento), siga estas regras simples:

```
   * Coloque um alarme pra 30 minutos. Assim que ele tocar, a reunião acabou.
Ponto final.
   * Chame o menor número de pessoas possível.
   * Nunca tenha uma reunião sem uma pauta bem clara.
```

Trabalhe com possíveis funcionários na base do "teste antes"

Uma coisa é olhar o portfólio, curriculum, exemplo de código ou trabalhos anteriores. Outra coisa é efetivamente trabalhar com alguém. Sempre que possível, faça um "test-drive" com possíveis novos membros da equipe.

Isso significa que você pode julgar pessoas pelas ações ao invés de apenas palavras. Você pode tomar decisões com base no que realmente importa:

- Qualidade do trabalho

Muitos programadores falam bonito, mas afinam na hora do "vamos ver". Com open source, você consegue ver com detalhes as práticas e conhecimentos de programação de uma pessoa.

- Perspectiva cultural

Programar é tomar decisões. Muitas delas. Decisões são tomadas com base na cultura, nos valores e em ideais. Veja as decisões específicas feitas por um candidato enquanto está programando e testando, e veja seus argumentos na comunidade para ver se o candidato está dentro do que a empresa espera. Se não se encaixa na empresa, as decisões podem parecer erradas.

- Nivel de paixão

Por definição, envolvimento em projetos open source requerem um nível mínimo de paixão. Se não, porque outro motivo a pessoa perderia tempo na frente de um monitor? O tamanho do envolvimento em movimentos open source mostra quanto um candidato realmente se importa com programação.

- Porcentagem de finalização

Toda a inteligência, toda a cultura e paixão não se transformam em software de valor se o candidato não consegue terminá-lo. Infelizmente, muitos programadores não terminam seus projetos. Então, procure a exceção. Contrate aquele que consegue sair pela porta e está disposto a fazer as trocas pragmáticas que o trabalho exige.

- Lado social

Trabalhar com alguém por um bom período de tempo, durante tanto as horas de stress e descontração e altos e baixos vão mostrar a verdadeira personalidade do candidato. Se alguém não tem modos ou um lado sociável, deixe-os de lado.

Procure por generalistas que aprendem rápido em vez dos especialistas limitados

Nunca contrataremos alguém que seja um arquiteto de informação. É simplesmente específico demais. Com uma equipe pequena como a nossa, não faz sentido contratar pessoas com um conjunto de conhecimento tão limitado.

Equipes pequenas precisam de pessoas que possam vestir diferentes chapéis. Precisamos de designers que saibam escrever. Precisamos de programadores que entendam de design. Todos devem ter noção de como arquitetar informação (seja lá o que isso signifique). Todos precisam ter mentes organizadas. Todos precisam saber se comunicar com clientes.

E todos precisar querer e serem capazes de diminuir a marcha pela estrada. Tenha em mente que equipes pequenas eventualmente precisam mudar de direção rapidamente. Queremos alguém que possa se ajustar, aprender e fluir ao contrário de um pé-na-lama que só consegue fazer uma coisa.

Contrate bons escritores

Se está tentando decidir entre poucas pessoas para preencher uma posição, sempre contrate o melhor escritor. Não importa se essa pessoa é um designer, programador, marketing, vendedor ou o que for, essa habilidade leva a escrever mais efetivamente e concisamente código, design, emails, mensagens instantâneas e mais.

Isso porque ser um bom escritor é mais do que apenas palavras. Bons escritores sabem como se comunicar. Eles tornam as coisas mais fáceis de entender. Eles podem se colocar no lugar dos outros. Eles sabem o que omitir. Eles pensam claramente. E essas são as qualidades que você precisa.

Uma Mente Organizada

Boas habilidades de escrita são um indicador de uma mente organizada que é capaz de arranjar informação e argumentos de uma maneira sistemática e também ajudar (não fazer) outras pessoas a entender as coisas. Isso aparece no código, comunicação pessoal, mensagens instantâneas (para aqueles colaboradores de longa distância) e até esses conceitos exotéricos como profissionalismo e confiança.

—Dustin J. Mitchell, developer (de Signal vs. Noise) Escrita Clara leva a Pensamento

Escrita clara leva a pensamento claro. Você não sabe o que sabe até tentar expressar esse conhecimento. Boa escrita é em parte uma questão de caráter. Em vez de fazer o que é fácil para você, faça o que é mais fácil para seu leitor.

—Michael A. Covington, professor de ciências da computação da Universidade da Geórgia (de Como Escrever mais Claramente, Pensar mais Claramente e aprender Material Complexo mais Facilmente)

Faça Design para quando as coisas derem errado

Vamos admitir: As coisa vão dar errado online. Não importa o quão cuidadoso você faça o design de sua aplicação, não importa quanto teste fizer, os clientes ainda vão encontrar problemas. Então como você gerencia essas quedas inevitáveis? Com design defensivo.

Escolha ferramentas que estimulem e motive o seu time

Um programador feliz é um programador produtivo. É por isso que nós otimizamos para felicidade e você deveria fazer o mesmo. Não escolha as ferramentas e práticas baseado simplesmente no padrão do mercado ou métricas de desempenho. Avalie os atributos intangiveis: a ferramenta foi criada com paixão, orgulho e dedicação?. Você seria feliz trabalhando neste ambiente oito horas por dia?

O Código Fala

Ouça quando seu código diz "não"

Ouça seu código. Ele oferecerá sugestões. Ele irá dizer "não". Ele lhe dirá onde ficam as armadilhas. Ele irá sugerir novas maneiras de fazer as coisas. Ele irá ajudá-lo a se manter em um modelo de menos software.

Uma nova funcionalidade está requerendo semanas de tempo e milhares de linhas de código? Isso é seu código lhe dizendo que provavemente existe uma maneira melhor. Existe uma maneira simples de codificar alguma coisa em uma hora em vez de uma maneira complicada que consumirá dez horas? Novamente, esse é seu código o guiando. Ouça.

Seu código pode guiá-lo a consertos que são baratos e leves. Preste atenção quando um caminho mais fácil emerge. Claro, a funcionalidade que é fácil de fazer pode não ser exatamente a mesma que você originalmente tinha em mente, mas e daí? Se funciona bem o suficiente e lhe dá mais tempo para trabalhar em outra coisa, é um ganhador.

Ouça

Não se preocupe com o design, se ouvir seu código um bom design vai aparecer ... Ouça as pessoas técnicas. Se eles estão reclamando sobre a dificuldade de fazer mudanças, então leve essas reclamações a sério e lhes dê tempo para consertar as coisas.

—Martin Fowler, Cientista Chefe, ThoughtWorks (de Is Design Dead?)

Abra as Portas

Publique dados para o mundo via RSS, APIs, etc.

Não tente prender seus usuários. Deixe que eles possam ter acesso a suas informações quando

quiserem, da forma que preferirem. Para tal, você precisa deixar de lado a idéia de manter os dados de seus usuários trancados a sete chaves. Em vez disso, deixe que a informação flua. Garanta o acesso à informação através de feeds RSS. Ofereça APIs que permitam a terceiros construir aplicações integradas à sua. Tais atitudes tornarão a vida dos usuários mais conveniente e expandirão as possibilidades do que sua aplicação é capaz de fazer.

No passado, as pessoas acostumaram-se a pensar nos feeds RSS apenas como uma boa maneira de se agregar conteúdo de sites de blogs e sites de notícia. Contudo, os feeds são mais poderosos que isto. Eles também podem permitir ao usuário manter-se atualizado sobre mudanças internas à aplicação sem a necessidade de logar-se repetidas vezes. Através do site do Basecamp, por exemplo, o usuário pode cadastrar sua url em um agregador de RSS e assim receber notificações de mensagens de projetos, listas de tarefas e objetivos sem a necessidade de conectar-se constantemente ao site em busca de informações atualizadas.

APIs permitem que desenvolvedores construam plugins adicionais à sua aplicação, que geralmente agregam valor ao seu produto. Por exemplo, a API disponibilizada pelo Backpack foi utilizada pela Chipt Productions na construção de um widget para o Mac os X. A pequena aplicação permite aos usuários adicionar e editar lembretes, listagens de items e muito mais a partir de seus desktops. Muitos usuários apontaram o widget como uma ótima ferramenta, e alguns mesmo apontaram-no como um fator decisivo na escolha da utilização do Backpack.

Outros bons exemplos de empresas que liberaram dados como uma maneira de conseguir um 'efeito bumerangue':

- A API do Google Maps permitiu o surgimento de toda sorte de pequenas aplicações que recuperam dados de outras fontes (ex.: uma listagem de apartamentos) e os exibem em um mapa.

- Linkrolls oferece aos usuários exibir seus últimos bookmarks do del.icio.us em seu próprio site.

- O Flickr permite que outros negócios acessem as suas APIs comerciais, de forma a permitir aos usuários comprar livros de fotos, posters, backups em DVD e selos. "O objetivo é manter as portas completamente abertas e permitir o maior número possível de possibilidades de utilização de suas fotos", diz Stewart Butterfield, do Flickr.

Um Widget Faz a Diferença

Quando a 37signals lançou o Backpack, há algum tempo atrás, minha primeira impressão foi… er... bem...

Ocorreu mais ou menos na época em que a Chipt Productions lançava um widget Backpack para o Sistema Operacional Tiger — que parecia interessante demais para passar despercebido — com isso dei uma segunda olhada no Backpack. O resultado? Uma grande diferença.

Hoje, sempre que uma nova idéia surge, abro o widget, digito e salvo — e pronto. Recebo algum e-

mail com algo que devo fazer? Abro o widget, digito e salvo — e pronto. O widget tornou-se um tipo de bloco de notas indispensável, que instalo em todo Mac que uso. E por se tratar de uma aplicação totalmente web, não há necessidade de nenhum tipo de controle de versão ou sincronizaçao de dados — apenas a fluidez de digitar-se dados sem ter que se preocupar em saber para onde os dados foram, nem como acessá-los mais tarde. —Todd Dominey, fundador, Dominey Design (de Trying on Backpack)

Amostra Grátis

Dê alguma coisa de graça

É um mundo barulhento lá fora. Para que as pessoas o notem no meio da multidão, dê alguma coisa de graça.

Empresas espertas sabem que dar brindes é uma excelente maneira de fisgar clientes. Veja a Apple. Eles oferecem o software iTunes de graça de forma a gerar demanda para o iPod e a loja de música iTunes. No mundo offline, as lojas fazem a mesma coisa. A Starbucks diz que uma nova compra é estimulada para cada cinco amostras de bebidas que eles dão aos clientes. Nada mau.

Para nós, Writeboard e Ta-da list são aplicativos completamente grátis que usamos para colocar as pessoas no caminho para usar nossos outros produtos. Adicionalmente, sempre oferecemos algum tipo de versão grátis de todos os nossos aplicativos.

Queremos que as pessoas experimentem o produto, a interface, a utilidade do que construímos. Uma vez fisgados, eles são muito mais propensos a atualizar para um dos planos pagos (que permitem mais projetos ou páginas e dá acesso a funcionalidades adicionais como upload de arquivos e encriptação de dados com SSL). Pedacinhos

Faça pedacinhos: crie ofertas especializadas, pequenas para que os clientes mordam. Subdivida pelo menos um produto ou serviço em pedacinhos que são baratos, fáceis ou divertidos.

—Ben McConnell e Jackie Huba, autores do Church of the Customer Blog (de What is customer evangelism?)

Dê Sua Música de Maior Sucesso

Considere doar uma de suas músicas (por álbum) como download gratuito promocional para o mundo – para ser como um trailer de cinema – como o single de sucesso enviado ao rádio – a música que faz as pessoas quererem comprar sua música.

Não se preocupe com pirataria dessa música. Deixe as pessoas tocarem, copiarem, compartilharem. Tenha a confiança que, se o mundo a ouviu, irão pagar por mais. —Derek Sivers, presidente e programador, CD Baby e HostBaby (de Free Promo Track)

Um Poderoso Site Promocional

Vá do Trailer para a Prévia para o Lançamento

A melhor ferramenta promocional é um grande produto. A palavra vai se espalhar se tivermos uma aplicação que as pessoas acham realmente útil.

Ainda assim, precisamos de um bom site promocional também. O que devemos incluir nesse site? Algumas idéias:

```
   * Apresentação: Explique sobre a aplicação e seus benefícios.
   * Turismo: Guie as pessoas pelas várias funcionalidades
   * Fotos de tela e vídeos: Mostre às pessoas como sua aplicação realmente se
parece e como usá-la.
   * Manifesto: Explique a filosofia e idéias por trás dela.
   * Estudos de Caso: Dê exemplos reais que mostram o que é possível.
   * Euforia: Frases testimoniais de clientes, revisões, imprensa, etc.
   * Fórum: Ofereça um local para membros da comunidades se ajudarem uns aos
outros.
   * Precificação e Assinatura": Leve as pessoas à aplicação o mais rápido
possível.
   * Weblog": Blogs mantém seu site atualizado com notícias, dicas, etc.
```

Sinta a Dor

Derrube as paredes entre suporte e desenvolvimento

No negócio de restaurantes, existe uma enorme diferença entre aqueles que trabalham na cozinha daqueles que estão na linha de frente lidando com clientes. É importante para ambos os lados entender e simpatizar com o outro. É por isso que escolas de culinária e restaurantes normalmente terão chefs trabalhando como garçons para que a equipe da cozinha possa interagir com clientes ever como é realmente estar na linha de frente.

Muitas empresas desenvolvedoras de software tem uma divisão similar. Designers e programadores trabalham na "cozinha" enquanto o suporte lida com clientes. Infelizmente, isso significa que chefs de software nunca ouvem o que o cliente realmente está dizendo. Isso é problemático porque ouvir clientes é a melhor maneira de se ligar nas partes fortes e fracas do seu produto.

A solução? Evite construir paredes entre seus clientes e a equipe de desenvolvimento/design. Não terceirize o suporte a seus clientes. Faça você mesmo o suporte. Você e sua equipe inteira, devem saber o que seu cliente está dizendo. Quando seu cliente está incomodado, você precisa saber disso. Você pecisa ouvir as reclamações. Você precisa ficar incomodado também.

Na 37signals, todos os e-mails de suporte são respondidos pessoalmente pelo pessoal que realmente construiu o produto. Por que? Primeiro, isso fornece melhor suporte aos clientes. Eles estão recebendo uma resposta diretamente do cérebro de alguém que construiu a aplicação. Além disso, isso nos mantém em contato com a pessoa que usa nossos produtos e com os problemas que estão

encontrando. Quando estão frustrados, nós ficamos frustrados. Podemos dizer sinceramente que "eu sinto sua dor".

Pode ser tentador se apoiar em análises estatísticas para revelar seus pontos problemáticos. Mas estatísticas não são como vozes reais. Você precisa eliminar a maior quantidade possível de atravessadores entre você e as vozes reais de seus clientes.

As linhas de frente são onde a ação está. Vá até lá. Faça seus chefs trabalharem como garçons. Leia e-mails de clientes, ouça suas frustrações, escute suas sugestões e aprenda com elas.

Treinamento Zero

Use ajuda em contexto e FAQs para que seu produto não precise de um manual ou treinamento

Você não precisa de um manual para usar o Yahoo! ou Google ou Amazon. Então por que você não pode construir um produto que não requer manual? Se esforce para construir uma ferramenta que requer treinamento zero. Como fazer isso? Bem, como mencionamos antes, você começa mantendo tudo simples. Quanto menos complexa for sua aplicação, menos precisará ajudar as pessoas sem necessidade. Depois disso, uma grande maneira de suporte pró-ativo é usando ajuda em contexto e FAQs em potenciais pontos de confusão.

Por exemplo, oferecemos suporte pró-ativo na tela que permite as pessoas a fazer upload de seus logotipos ao Basecamp. Algumas pessoas experimentaram um problema onde continuavam vendo um logotipo antigo por causa do cache do browser. Então, próxima à área de "envie seu logotipo", adicionamos um link a um FAQ que instruía os clientes a forçar um recarregamento de seus browsers para ver o novo logotipo. Antes de fazermos isso recebíamos 5 e-mails por dia sobre esse problema. Agora, não recebemos nenhum.

Em Fórum Afinado

Use fóruns ou chats para deixar os clientes se ajudarem

Fórum e chats de grupo baseados na web são uma grande maneira de deixar clientes fazerem perguntar e ajudar uns aos outros. Eliminando o intermediário – esse é você – você fornece uma linha aberta de comunicação e economiza seu tempo no processo.

Em nossos fóruns de produtos, os clientes publicam dicas e truques, requisições de funcionalidades, histórias e mais coisas. Nós aparecemos de tempos em tempos para oferecer assistência, mas os fóruns são principalmente um lugar para a comunidade se ajudar e compartilhar experiências com o produto.

Você ficará surpreso com quantas pessoas querem se ajudar.

Original em:

http://gettingreal.37signals.com/GR_por.php

Instalação e configurações do ambiente/Via Xampp

De Wikibooks

Conteúdo

- 1 O que é o Xampp?
- 2 Sistemas Operacionais Suportados
- 3 Utilitários
- 4 Instalação
- 5 No Windows
- 6 No Linux
- 7 Usando o Xampp
- 8 Testando PHP
- 9 Instalar PostgreSQL e integrar com PHP
- 10 Tornando o Xampp mais Seguro
- 11 Mudando o DocumentRoot
- 12 Adicionando Domínio Virtual no Apache com Xampp

O que é o Xampp?

O Xampp é um utilitário que instala o Apache, PHP (PEAR e PECL), MySQL, Perl, phpMyAdmin, proFtpd e outros. Atualmente se encontra na versão XAMPP 1.6.2 para o windows e já vem configurado com:

```
* MySQL 5.0.41
* PHP 5.2.2
* PHP 4.4.7
* phpMyAdmin 2.10.1
```

Sistemas Operacionais Suportados

Tem versões para Linux, Windows e outros SOs.

Utilitários

Traz ainda em sua index (http://127.0.0.1) um menu com a relação dos seus utilitários e alguns

exemplos em php e perl. Este menu é facilmente incrementado ao se editar os arquivos .inc em:

C:\Arquivos de programas\xampp\HTDOCS\xampp.

Instalação

Faça o download do site oficial de acordo com o seu sistema (http://xampp.sf.net) ou se preferir em português do Brasil: http://www.apachefriends.org/pt_br/xampp.html

No Windows

Execute o instalador

Diretório web default = C:\Arquivos de programas\xampp\HTDOCS

php.ini - C:\Arquivos de programas\xampp\apache\bin

httpd.conf - C:\Arquivos de programas\xampp\apache\conf

No Linux

Descompacte em /opt com

tar zxpvf xampp.*.tar.gz -C /opt

Diretório web - /opt/lampp/htdocs

php.ini e httpd.conf - /opt/lampp/etc

ATENÇÃO: Não esqueça de, após instalar o xampp, mudar as senhas default do apache, mysql e do próprio xampp. Ignorando este aviso, o xampp ficaria perigosamente exposto à invasões ou afins. (para mais informações de como proceder para mudar as senhas, vide a seção abaixo - Tornando o xampp mais seguro)

Usando o Xampp

No windows o executável abre um menu onde podemos executar os serviços separadamente. Após executar um ícone fica na barra de tarefas, o que facilita o gerenciamento.

No Linux – execute o arquivo "/opt/lampp/lampp serviço ação", onde:

serviço – um dos serviços disponíveis (httpd, mysql, etc)

ação – start, stop, status

/opt/lampp/lampp start (inicia todos os serviços disponíveis)

Configurando o Servidor de FTP (proFtpd)

editar /opt/lampp/etc/proftpd.conf e alterar usuário, senha e diretório a gosto

O diretório default é o /opt/lampp/htdocs

Sugestão: criar dentro do htdocs um sub com nome ftp.

A senha do proftpd não pode ser trocada manualmente, pois é criptografada.

Para trocar devemos usar o security:

/opt/lampp/lampp security

Dica: Se você não quiser ter que digitar o caminho completo (/opt/lampp/lampp start) sempre que quiser lançar o servidor xampp, você pode criar um link para o executável no "path" do seu sistema com um simples:

$ sudo ln -s /opt/lampp/lampp /usr/local/bin/lampp

Com isso, um

$ sudo lampp start

bastará para lançar o servidor.

Testando PHP

Tanto no windows quanto no linux, startar o apache e abrir o browser apontando para http://localhost ou http://127.0.0.1

Se tudo correu bem verá a index do Xampp com todos os seus recursos,onde poderá inclusive optar pelo idioma Português do Brasil.

Instalar PostgreSQL e integrar com PHP

Download do site oficial (http://www.postgresql.org), escolhendo pelo sistema operacional.

Instalação

Windows – execute o instalador

Linux – Para distribuições com instalação de binários fica mais fácil

Instalar dos fontes. Apenas descompacte e leia o arquivo INSTALL, onde encontrará passos resumidos para a instalação.

Configuração

Para configurar que máquinas terão acesso ao postgresql edite o arquivo pg_hba.conf, que fica no subdiretório data do postgresql.

Configure acesso via TCP/IP no arquivo postgresql.conf que fica no mesmo diretório do pg_hba.conf (em algumas distribuições fica no /etc/postgresql).

Integrando com PHP

No Windows descomente a linha com "extension=php_pgsql.dll" e restarte o apache. No Linux o suporte já vem habilitado por default.

Tornando o Xampp mais Seguro

No Linux - Abra a interface web (127.0.0.1)

- Clique no link Segurança à esquerda e leia as instruções

- Para ativar a segurança execute como super usuário:

/opt/lampp/lampp security

E ative tudo que desejar.

Após isso, para acessar o phpMyAdmin de forma mais confortável, edite o arquivo:

/opt/lampp/lib/xampp.users e alterar o nome do user default de "lamp" para o seu.

Então ao acessar entre com seu usuário e senha (a que cadastrou ao adicionar a segurança no Xampp para acessar o sistema de arquivos) e logo em seguida entre com root e senha (do banco).

Mudando o DocumentRoot

Em algumas vezes é mais cômodo ou então mais seguro alterar o diretório default do DocumentRoot do Apache. Neste caso, **para o Linux** proceder da seguinte forma:

Mude o caminho para/home/ribafs:

- Acesse o terminal

- sudo gedit /opt/lampp/etc/httpd.conf

- Tecle Ctrl+F para localizar "htdocs"

- Troque as duas ocorrências com htdocs para "/home/ribafs/htdocs"

- Crie a pasta: md /home/ribafs/htdocs

- Então no terminal novamente execute:

sudo cp -Ra /opt/lampp/htdocs /home/ribafs/htdocs

- Mude as permissões para o usuário "ribafs" assim:

sudo chown -R ribafs:ribafs /home/ribafs/htdocs

sudo chmod -R 705 /home/ribafs/htdocs

Então

cd /home/ribafs/htdocs/xampp

chmod 777 lang.tmp

Adicionando Domínio Virtual no Apache com Xampp

Mesmo estando usando o PHP localmente podemos ter um outro domínio ou subdomínio. Quando instalamos o PHP via Xampp localmente, nós temos o IP 127.0.0.1 ou o nome localhost, de forma que abrimos no navegador com http://127.0.0.1 ou http://localhost. Podemos também ter localmente um site que abre em outro endereço como, por exemplo, http://ribafs.localhost. Como realizar isso?

Os passos seguintes são um resumo do turotial:

Preparando o terreno para o desenvolvimento em php com o Xampp no Ubuntu

- Edite o seu /etc/hosts e adicione a linha:

-- 127.0.0.1 ribafs.localhost

- Edite o /opt/lampp/etc/httpd.conf e descomente a linha:

-- Include etc/extra/httpd-vhosts.conf

- Edite o /opt/lampp/etc/extra/httpd-vhosts.conf e insira o trecho de código:

```
<VirtualHost *:80>
DocumentRoot /home/ribafs/htdocs/ribafssite
ServerName ribafs.localhost
ErrorLog logs/ribafs.localhost-error_log
CustomLog logs/ribafs.localhost-access_log common
</VirtualHost>
```

Lembre de preservar o trecho do localhost:

```
<VirtualHost *:80>
ServerAdmin ribafs@yahoo.com
DocumentRoot /home/ribafs/htdocs
ServerName localhost
ErrorLog logs/localhost-error_log
CustomLog logs/localhost-access_log common
</VirtualHost>
```

Então reinicie o Apache.

Agora você tem um domínio virtual chamado ribafs.localhost, que inclusive atende via ping ribafs.localhost e pode ser chamado pelo navegador:

http://ribafs.localhost

No diretório indicado em DocumentRoot /home/ribafs/htdocs/ribafssite podemos adicionar o conteúdo de outro site.

Instalação e configurações do ambiente/Individualmente

De Wikibooks

3.2 - Instalando o PHP dos Fontes ou pegando os pacotes individualmente

Aqui seguem vários links com tutoriais de como instalar o ambiente para uso do PHP pegando cada um dos pacotes, inclusive compilando para Windows e para Linux.

Também podemos utilizar diversos outros pacotes instaladores existentes como EasyPHP e outros, alertando que em sua maioria existem apenas para Windows, com algumas exceções como é o caso do

Uniform Server

Veja esta dica do Haroldo Barbosa sobre o Uniform Server:

http://www.dicas-l.com.br/dicas-l/20070519.php

Agora os tutoriais sobre a instalação do ambiente para o PHP:

Quem precisa de um controle rigoroso sobre cada pacote e sobre as características de cada um destes, para ter uma maior segurança e conhecimento dos mesmos, nestes casos a instalação de cada pacote individualmente e compilando dos fontes para controle mais rigoroso do comportamento é o caminho mais indicado.

No Viva o Linux

Apache 2.0 + PHP + PostgreSQL no Slackware

Debian com Apache, PHP4, PHP5 e MySQL

Instalação do Apache, MySQL e PHP

Instalando Apache2, PHP4 e MySQL

Instalando e configurando o Apache, PHP e MySQL no Slackware 10.1

Instalando servidor Apache + PHP + MySQL + phpMyadmin + no-ip no Ubuntu 6.10 Server

PHP5 + Apache2 no Ubuntu 7.04, com suporte ao MySQL e SQL Server

No Guia do Hardware.net

Instalando o Apache + PHP + MySQL no Windows

No Linux Ajuda

Apache2 + PHP + PostgreSQL + phpPgAdmin

Editores e IDEs para PHP, CSS, JavaScript e HTML

De Wikibooks

4 - Editores e IDEs para PHP, CSS, JavaScript e HTML

"A grande mídia não é wiki. Ela ainda não sabe trabalhar com a pluralidade pois se considera déspota da informação." Jaime Balbino

1. Eclipse com PHPEclipse, Aptana e outros plugins
2. PHPDesigner
3. Kate
4. Dreamweaver
5. Zend Studio
6. VS.PHP
7. Delphi for PHP
8. KDevelop
9. PSPAD
10. Quanta
11. BlueFish
12. PHPEdit

Editores e IDEs para PHP

Editores e IDEs são ferramentas que podem tornar o trabalho do programador algo mais eficiente, mais rápido e mais agradável. Ele é bom quando agiliza o nosso trabalho. Mesmo que tenha muitos recursos mas venha a tornar o trabalho mais lento ou maçante não será produtivo. Aí se reforça a utilidade de testar várias ferramentas para descobrir qual melhor se adapta a cada um de nós.

O fato de um desenvolvedor ou um grupo considerar um certo editor ou IDE como o melhor, geralmente não significa que é o melhor para todos. O gosto pessoal ou de um grupo é algo particular e jamais representará o de todos.

Daí a diversidade de ferramentas no mercado para atender a diversos gostos, formações e experiências. Vale reforçar que não existe o melhor para todos, mesmo que exista um com a maior quantidade de recursos. Portanto experimente o máximo que puder e somente após eleja o seu preferido, aquele com o qual mais sente afinidade. Vale lembrar ainda que podemos eleger mais de um como preferido, existindo aquele principal e aquele para casos específicos.

No site PHP Editor Review (http://www.php-editors.com/) existe uma boa relação de editores para PHP, inclusive com uma busca onde podemos selecionar o Sistema Operacional e a licença.

A quantidade de editores e IDEs para trabalhar com PHP é relativamente grande. Aqui vou abordar apenas alguns que experimentei e com outros fiz poucos testes.

Este texto é fruto de uma busca no Google (http://www.google.com.br), uma pesquisa no site PHP Editor Review, um debate na lista PHPfortaleza (http://groups.yahoo.com/group/phpfortaleza) e algo da própria experiência do colaborador.

Inicialmente a intenção foi de encontrar um bom editor free para Windows, já que para Linux existem diversas boas alternativas nativas e então acabei fazendo uma pesquisa maior.

Aqui citarei pequenos editores simples, editores especializados para programação e IDEs para PHP. Os editores simples são aqueles que apenas editam texto puro e contam com alguns poucos recursos úteis, como numeração de linhas e outros. Os editores para programação em geral e alguns específicos para PHP. As IDEs normalmente contam com a maior quantidade de recursos.

Um detalhe importante que deve pesar na escolha do editor/IDE é se o projeto do mesmo ainda está em andamento, o tamanho da comunidade por traz do mesmo ou empresa e a data da última versão. Toda regra tem exceção, portanto fique atento.

Características importantes

- Colorir sintaxe
- Autocompletar código (funções do PHP)
- Autocompletar funções do usuário
- Dicas de funções (exibindo os parâmetros)
- Debug

- Integração com SGBDs
- Executar o script internamente ou via browser
- Assistentes para criação de código HTML, JS, CSS, etc
- Assistente para classes, funções, estruturas de controle do PHP
- Assistentes para CSS, JS e SQL
- Numeração de linhas
- Procurar/Sobrescrever no script (sensível ao caso e palavra completa)
- Suporte a várias codificações (UTF-8, ISO-8859-1, etc)
- Code Browser (classes e funções)
- Help integrado

Editores e IDEs para PHP, CSS, JavaScript e HTML/Eclipse com PHPEclipse, Aptana e outros plugins

De Wikibooks

Conteúdo

- 1 Introdução
- 2 Características do PHPEclipse
- 3 Alterando o Help Online para Português do Brasil
- 4 Instalações
  - 4.1 Instalação do Eclipse
  - 4.2 Instalação do PHPEclipse
  - 4.3 Instalação do JSEclipse
  - 4.4 Instalação do EclipseTidy
  - 4.5 Instalação do Quantum DB
  - 4.6 Configuração do PHPEclipse no Eclipse
  - 4.7 Atualizando o Eclipse e seus plugins
  - 4.8 Criando um Novo Projeto e PHP
  - 4.9 Adicionando arquivos PHP ao Projeto criado
  - 4.10 Adicionando arquivos HTML e JavaScript
  - 4.11 Auto_completando o código
  - 4.12 Debugando Scripts PHP
  - 4.13 Configuração do PHP e Apache para melhor Debug
  - 4.14 Tipos de Erros e avisos do PHP

Introdução

O desenvolvimento de um aplicativo Web em PHP requer basicamente conhecimentos nas seguintes áreas: HTML, JavaScript, CSS, PHP e de um banco de dados. Com a IDE Eclipse podemos juntar alguns plugins e formar um ótimo ambiente de desenvolvimento.

Um recurso muito importante em uma ferramenta web é um bom debugger, pois com o que se perde muito tempo é com a caça aos bugs. O melhor debugger que testei foi o do PHPEclipse.

Agora comparemos o trabalho em um editor de texto comum e no editor do PHPEclipse. No editor de texto normalmente se cria um script até o final e então manda-se para o servidor para testar no browser.

No PHPEclipse a cada linha digitada temos uma ajuda do debugger (caso haja algum erro ele avisa e até corrige com uma solução de sugestão). Tem dúvida sobre alguma palavra-chave? Deixe o cursos de texto sobre a mesma e tecle Ctrl+Shift+H para receber uma página da documentação oficial do PHP, com sintaxe, exemplos, etc. Isso até no seu idioma nativo, como veremos adiante.

Auto-completa código HTML, JavaScript e PHP. Gerenciamento do banco de dados integrado (QuantumDB), Modelagem de dados (Azurry Clay) e muitas outras facilidades integradas.

Com a intenção de encontrar um ambiente de desenvolvimento que torne o trabalho com PHP mais produtivo, testei diversos editores e IDEs, inclusive alguns comerciais, como o ZendStudio e o Dreamweaver. O Eclipse, juntamente com o PHPEclipse, JSEclipse, o EclipseTidy e o QuantumDB formaram o ambiente de desenvolvimento mais produtivo, além de inteiramente free e open-source. Com ele se desenvolve tando o HTML, o JavaScript quanto o código PHP. A característica mais marcante ao meu ver, a mais produtiva para desenvolvimento em PHP é o debugger. Sei que existe a variante pessoal, o gosto de cada um, e algumas pessoas podem preferir esta ou aquela ferramenta, dependendo de sua bagagem anterior e de sua atitude, mas para mim esta foi a melhor alternativa para trabalho com o PHP.

O PHPEclipse é formado por um conjunto de plugins para o Framework Eclipse que traz uma IDE integrada para desenvolvimento de PHP (http://www.plog4u.org/index.php/Using_PHPEclipse). Podemos utilizar o Eclipse com PHPEclipse tanto no Windows quanto no Linux, pois o mesmo é desenvolvido em Java para ambientes gráficos Desktop e roda em qualquer ambiente que suporte Java.

O Eclipse oferece um conjunto de recursos importantes para o desenvolvedor, como o CVS e Subversion, Webdav, busca texto em projetos e muitos plug-ins de terceiros, além de uma grande comunidade de usuários e desenvolvedores.

Características do PHPEclipse

• PHP, HTML, XML and CSS Syntax Highlighting

• Code completion

• Javascript Editor (via JSEclipse)

• Integrated web browser preview (somente no Windows)

• Integrated control of Apache & MySQL servers (if using XAMP distribution)

• Some PHPUnit Integration (?)

• Integrated MySQL client view using Quantum DB (PostgreSQL e outros bancos)

• HTMLTidy integration

• Debug

• Marca início e final de chaves e parênteses, colchetes e chaves

(?) Ainda não foram testados

Integrado aos plugins JSEclipse e EclipseTidy oferece ainda mais recursos.

Características da IDE Eclipse+PHPEclipse+JSEclipse+EclipseTidy:

• Colorir sintaxe.

• Auto-completar código:

• PHP – (Ctrl+BarraDeEspaço)

• HTML (digitar parte e aguardar. Exemplo: digitar "<" e aguardar ou parte e Ctrl+Espaço. Ajuda com Tabelas, forms, etc.

• JavaScript (sililar ao HTML), exemplo: digitar "<script", aguardar um pouco e teclar Ctrl+Espaço

• Tanto para HTML quanto para JavaScript ele completa com trechos de código. Se teclar <html e Ctrl+Espaço ele exibe todo o cabeçalho do HTML. Tecle "$" e aguarde, ou "$n".

Obs.: é sensível ao contexto. Só auto-completará uma função php num script php.

• Ajuda Rápida ao selecionar palavra-chave e mover mouse sobre a mesma.

• Ajuda completa ao selecionar palavra-chave e teclar Ctrl+Shift+H ou botão direito e PHP Help)

• Debugar o código logo que teclamos Enter em novo script, ou seja, a cada linha. Ao abrir script existente. Sempre que existe erro e salvamos o arquivo, o nome do arquivo e o diretório onde ele encontra-se ficam com um pequeno X vermelho acusando o erro.

• Numeração de linhas (Window – Preferences – PHPEclipse... - PHP – Show Line Numbers)

• Controle de Tabulação (Window – Preferences – PHPEclipse... - PHP – Display tab width)

• No Windows ao abrir ou ao salvar mostra o resultado na janela PHP Browser (se corretamente configurado em Window – Preferences – PHPEclipse... – Project Defaults e WebServer local).

Alterando o Help Online para Português do Brasil

(Tradução e resumo da Wikipedia sobre PHPEclipse)

• Fechar Eclipse

• Mover doc.zip de .../plugins/net.sourgeforge.phphelp... para uma pasta temporária

• Descompactar doc.zip

• Baixar o arquivo doc do idioma desejado do site oficial do php, sendo do tipo "Many HTML files"

• Descompactar no diretório doc do doc.zip, socrescrevendo todos

• Compactar o diretório doc como doc.zip (embutindo também o diretório doc). No linux:

zip -r doc.zip doc. No Windows selecionar o diretório doc e compactar com WinRar ou outro.

• Copiar doc.zip novamente para .../plugins/net.sourceforge.net.phphelp

Utilizando: Selecione uma palavra-chave em um script PHP e tecle Ctrl+Shift+H ou clique com o botão direito do mouse e PHP Help.

Instalações

Instalação do Eclipse

- Baixar a versão 3.1.1 ou superior aqui: http://www.eclipse.org/downloads/index.php preferindo um mirror brasileiro (veja abaixo na lista em South America). Requer Java (JRE ou JSDK).

- Descompacte o arquivo

Instalação do PHPEclipse

- Baixar a versão CVS 1.8 ou superior do site oficial: http://www.phpeclipse.de/

- Copie o arquivo .zip para o diretório do Eclipse. Então descompacte.

Instalação do JSEclipse

Baixe também do site:

http://www.interaktonline.com/Products/Eclipse/JSEclipse/Overview/

Instale de maneira semelhante a instalação do PHPEclipse.

Instalação do EclipseTidy

Download - http://eclipsetidy.sourceforge.net/ e instale da mesma forma dos anteriores.

Instalação do Quantum DB

Características:

• connect to databases using standard JDBC drivers

• review schemas, tables, views and sequences

• look up column, index and foreign key information

• issue ad-hoc queries or other SQL statements against the database

• manage, edit, and work with SQL files (*.sql)

• issue updates, deletes, and inserts using simple, easy-to-use wizards

Trabalha com Adabas, DB2, DB2 on AS400, HSQLDB, Informix, MySQL, Oracle, Pointbase, PostgreSQL, Sybase e vários bancos com um driver JDBC.

Download - http://sourceforge.net/projects/quantum/

Também descompacte para o mesmo diretório do Eclipse.

Para adicionar um banco, instale o JDBC correspondente a ele.

Para o PostgreSQL baixe daqui:

http://jdbc.postgresql.org/download.html (Para a versão 7.4 baixar o apropriado pacote).

Para o MySQL: http://www.aquafold.com/docs-jdbcdrivers-mysql.html

- Após descompactar abra o Eclipse e Window – Open Perpective – Other... Quantum DB e OK

- Na Perpectiva Database Bookmark (esquerda) clique com o botão direito e New Bookmark.

- Clique em Add Driver – Add External Jar... e indique o arquivo .jar que você baixou.

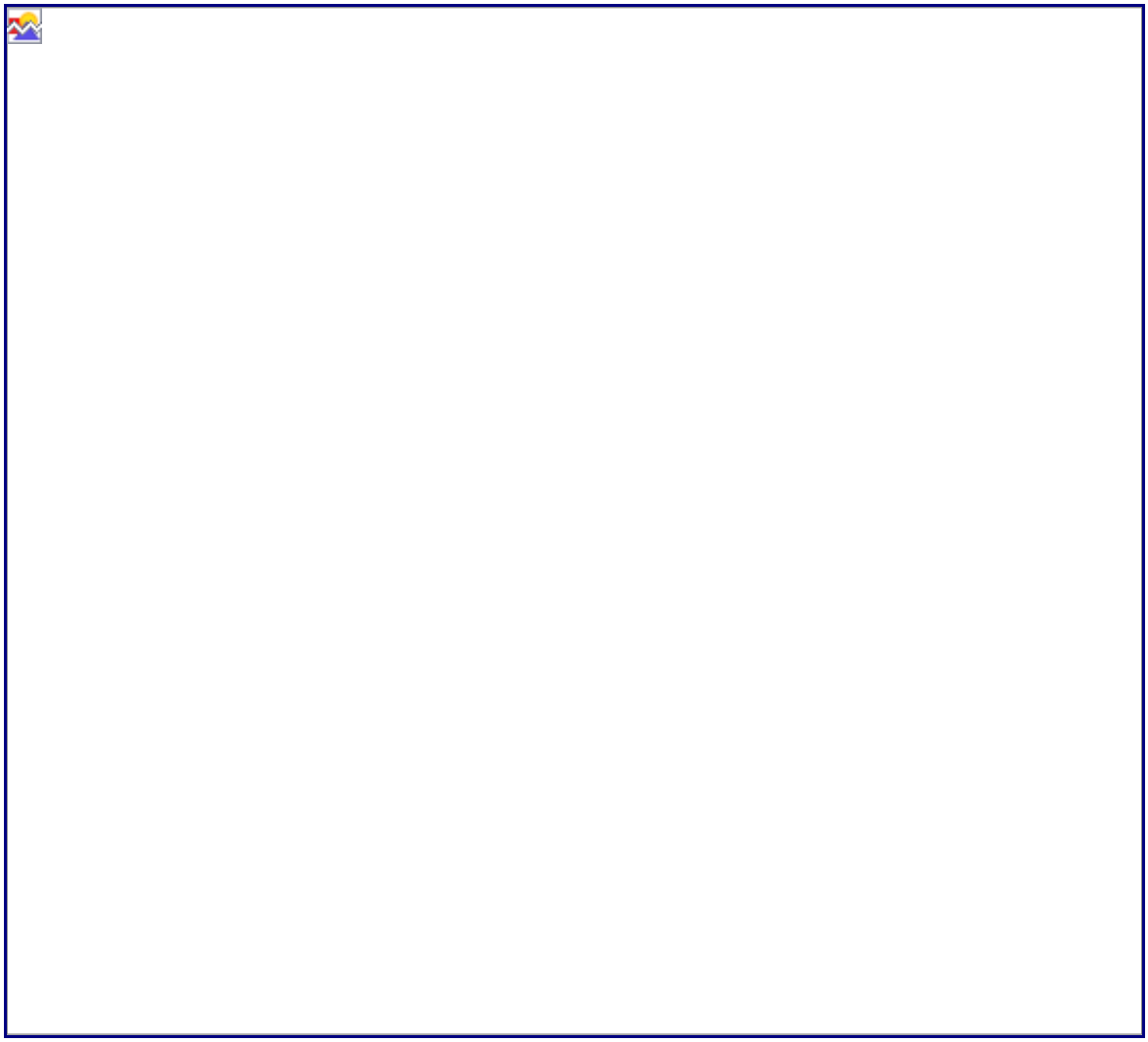
- Clique em Browse... e selecione o nome da classe e OK e Finish.

- Selecione o JDBC instalado e clique em Next.
- Userid – nomeuser e Password - ****** Hostname (127.0.0.1), Port (5432) Database name (banco).
- Clique em Next e digite um nome para esta conexão (sugestão: nome do banco) e clique em Next.
- Clique em Finish.
- Duplo clique no novo Bookmark que apareceu à esquerda para abrir (conectar) o banco.
- Ao selecionar e duplo clique em uma tabela, seu conteúdo é exibido abaixo e à esquerda.
- No painel de cima à esquerda podemos inserir consultas na sintaxe SQL.

Veja a tela abaixo para detalhes.

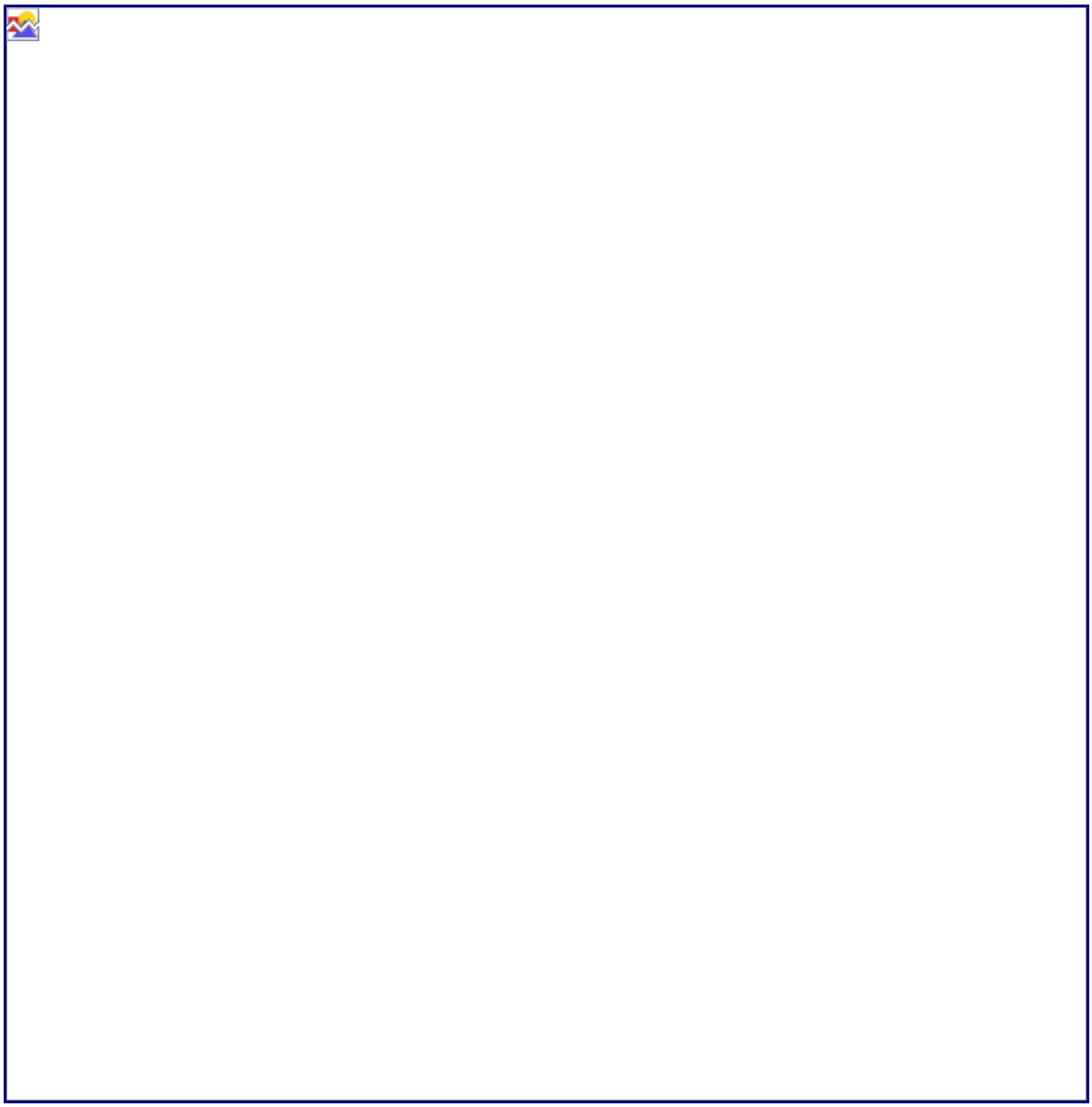
Configuração do PHPEclipse no Eclipse

Abrir o Eclipse – menu Window – Preferences

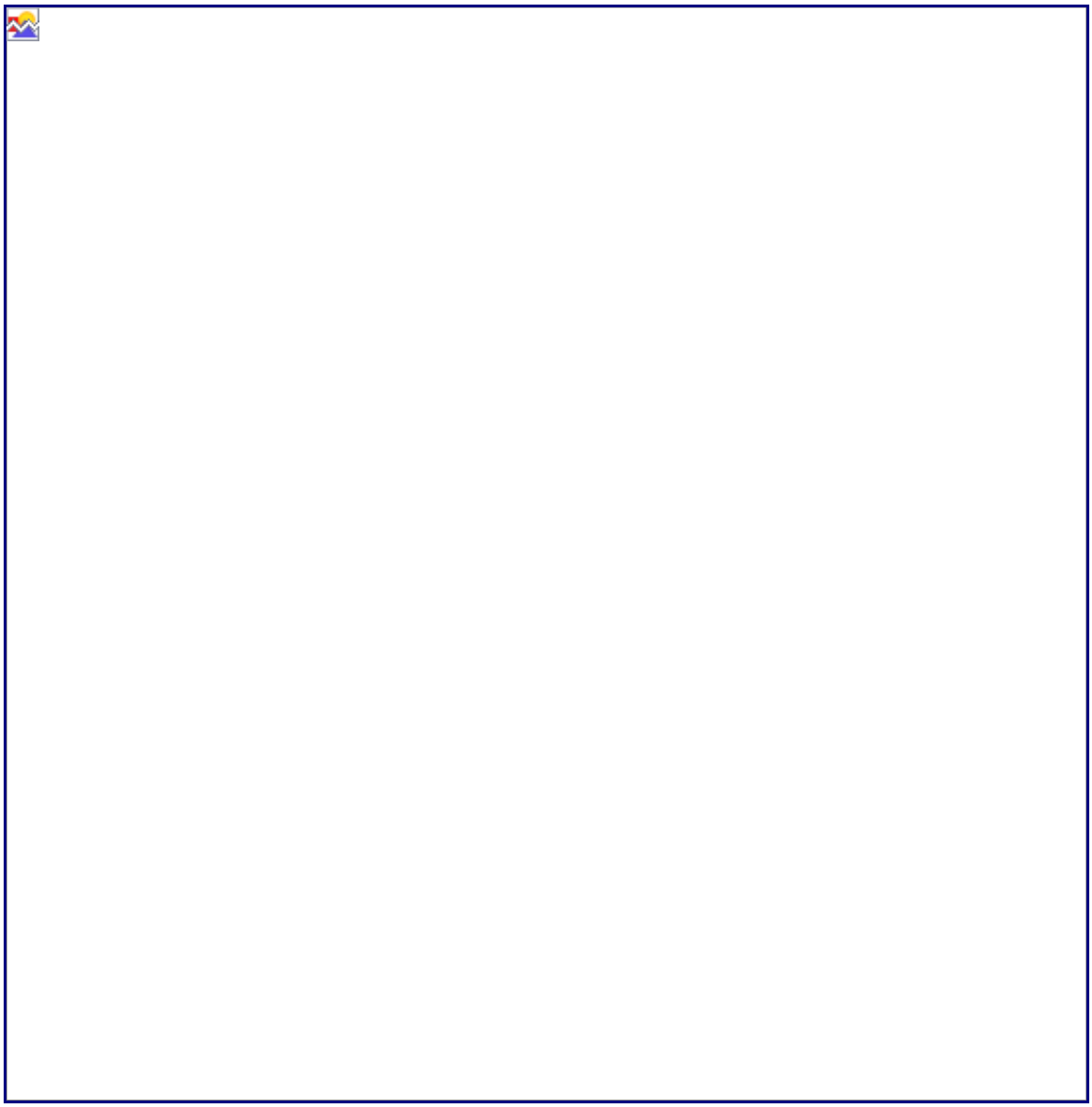


Observe que já temos JSEclipse e PHPEclipse Web Developn.

- Clique no sinal de + à esquerda de PHPEclipse

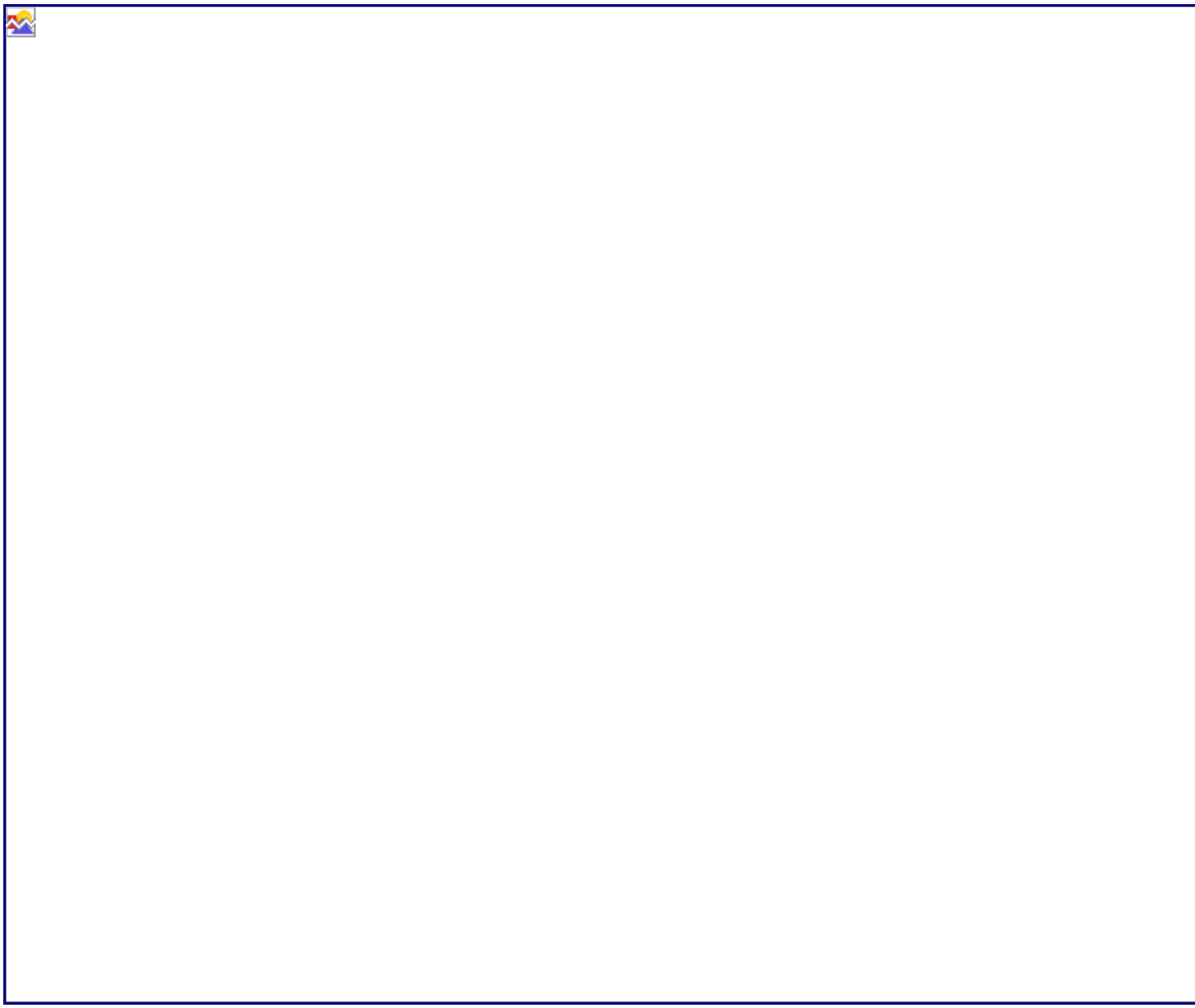


- Em Browser Configuration podemos adicionar outros navegadores ou alterar o atual. - Em PHP uma boa opção é marcar "Show line number" que exibirá os números de linha no editor. - Agora clique em PHP External Tools



Uma configuração no WindowsXP.

- Agora vejamos Project Defaults



Veja que as configurações aqui são para um servidor web rodando localmente.

- Adicionando o Help do PHP ao Eclipse (clique no sinal de + em PHP)

Indique o caminho (local) do arquivo de help do PHP no formado .chn.

A saída mais eficiente (pelos meus testes) é a de criar um novo doc.zip (como indicado).

Configurando a Atualização Automática do Eclipse

- Window – Preferences – Install/Update

Adicionar o servidor e a porta do Proxy.

Clicar em Automatic Updates e Clicar em Automacically find new updates and notify me.

Atualizando o Eclipse e seus plugins

- Help – Software Updates – Find and Install...

- Search for updates of the currently installed features

(Para atualizar o Eclipse atual e os Plugins instalados)

Selecionar um mirror e atualizar.

- Search for new features to install

(Procurar novos plugins para instalar)

Exemplo: Next – New Remote Site

Name - AnyEdit tools

URL - http://andrei.gmxhome.de/eclipse/

Criando um Novo Projeto e PHP

- File – New – Project

- PHP – Sinal de + e PHP Project e Next

- Digite um nome para o projeto e clique em Finish.

Obs.: Antes de finalizar podemos desmarcar "Use default" e salvar o projeto noutra pasta.

- Clicar sobre o nome do Projeto no painel (Navigator) da esquerda com o botão direito e New – PHP File

- Em File name informe o nome do arquivo e clique em Finish.

Recebemos então o seguinte código:

```
<?php
/*
* Created on 10/10/2005
*
* To change the template for this generated file go to
* Window - Preferences - PHPeclipse - PHP - Code Templates
*/
?>
```

Abrir uma linha em branco acima de ?> e digitar:

echo "Teste!";

Teclar Ctrl+S para salvar e observe que abaixo no painel PHP Browser já aparece o resultado: Teste!
Este recurso até o momento está disponível apenas em Windows.

Caso queira maximizar o painel PHP Browser para ver o resultado em tela cheia:

- Clique no botão maximize à direita:

Adicionando arquivos PHP ao Projeto criado

- Clique sobre o nome do projeto com o botão direito e New – PHP File.

- Em File name digite um nome para o arquivo e Finish.

Adicionando arquivos HTML e JavaScript

- Clique sobre o nome do projeto com o botão direito e New – Other – Sinal de + de PHP e HTML file.

- Em File name digite um nome para o arquivo e Finish.

Auto_completando o código

Logo abaixo da linha do echo digite "pg_" e então tecle Ctrl+BarraDeEspaços.

Veja o que aparece:

Uma relação de funções do PHP iniciadas com pg_ . A função selecionada tem uma pequena descrição aparecendo à direita.

Podemos também inserir outros arquivos no nosso projeto PHP: HTML, JS e outros formatos que também contarão com bons recursos do plugin PHPEclipse e do JSEclipse.

Debugando Scripts PHP

Procurar e corrigir erros em códigos é algo muitas vezes trabalhoso e outras vezes mais ainda. Para lidar melhor com estes problemas e resolvê-los mais rápido e eficientemente, devemos: ter um bom conhecimento dos mecanismos de correção da ferramenta em uso, da sintaxe das funções e estruturas do PHP, além de usar as melhores ferramentas de programação disponíveis: IDEs, editores, funções/classes, livros tradicionais, livros eletrônicos, o manual oficial do PHP em português do Brasil, aplicativos de exemplo e o que sobrar pesquisar no site oficial do PHP e na Internet em geral.

Algumas vezes o Interpretador acusa erro em uma linha e o erro de fato encontra-se em outra bem diferente (normalmente acima), como é o caso de strings não fechadas adequadamente e concletes, chaves ou parêntesis.

Maior atenção para os pontos:

(ponto e vírgula) ao final das instruções

aspas (") fechando com apóstrofos (') ao invés de aspas

),} ou] apenas abrindo ou fechando expressões em if com um igual (=) quando o correto são dois iguais (= =)

muita atenção com expressões lógicas

echos ou prints exibindo variáveis em pontos estratégicos. Exemplo: print "Valor $mValor
" para debugar;

Exemplo de uso do Debug do PHPEclipse

Crie um arquivo com o código abaixo:

<?

//include "conexao.inc.php";

//echo "Aguarde...;

/*

for ($campo=20;$campo<9020;$campo++){ $str="insert into clientes values ($campo, 'Ribamar.$campo','ribafs.$campo@gmail.com','Fortaleza.$campo', 'Ceará.$campo',$campo*2)"; pg_query($conexao,$str);

}

- /

```
//echo "
Concluido!";
$x=4;
if (($x ==0) && ($x>5){
echo "Furado!";
}else{
echo "Ok";
}
?>
```

Copie o arquivo para o diretório web como está. Então execute no browser.

Veja que ele retornará o seguinte erro:

Parse error: syntax error, unexpected '{' in C:\Arquivos de programas\xampp151\HTDOCS\teste.php on line 13

Não dá para identificar diretamente pela mensagem de erro, já que a { está correta.

Agora abra o Eclipse e num novo arquivo de projeto PHP cole o mesmo código acima.

Então tecle Ctrl+S para salvar. Veja que ao salvar ele grifa as duas chaves do if e adiciona dois pequenos círculos vermelhos à esquerda das linhas de número 13 e 15.

Pare o ponteiro do mouse sobre o círculo da linha 13. Receberá uma mensagem assim:

Parser error “’)’ expected after ‘if’ condition. “

Esta mensagem já nos ajuda mais que a retornada pelo browser, pois acusa de fato o problema, falta fechar o último parêntesis do if.

Configuração do PHP e Apache para melhor Debug

Isso na fase de desenvolvimento, pois na fase de produção o ideal é inibir as mensagens de erro.

Editar o arquivo php.ini e configurar:

display_errors = On

log_erros = On

error_reporting = E_ALL & ~E_NOTICE (Ativa todos, exceto os Notice)

Salvar, parar o Apache e iniciar.

Tipos de Erros e avisos do PHP

Parse error – problema com a sintaxe do script. O interpretador pára de rodar quando encontra um erro destes.

Fatal error – Um sério problema com o conteúdo do seu programa, como a chamada a uma função que ainda não foi definida. O interpretador pára ao encontrar um erro destes.

Warning – Aviso do interpretador que tem algo errado em seu programa. Exemplo: chamar função com número errado de argumentos. Mas este erro não impede de que o interpretador continue trabalhando.

Notice – Uma dica do interpretador. Exemplo: mostrar uma variável sem primeiro inicializar ela com algum valor. Não fará o interpretador parar.

Editores e IDEs para PHP, CSS, JavaScript e HTML/PHPDesigner

De Wikibooks

4.2 - PHPDesigner

Site Oficial

http://www.mpsoftware.dk/downloads.php

Preferido dos visitantes do site PHP Editor Review e também deste colaborador, com versões Freeware e comercial, mas somente para Windows.

Professional is a powerful PHP Integrated Development Environment (IDE) for both beginners and professional developers that enhances the process of editing, analyzing and publishing applications and websites powered by PHP and other web languages.

Boost your productivity and simplify complex coding projects through an intelligent set of editing tools that includes full support for PHP 5.2, syntax highlighters, intelligent nested code completion, code tips that are displayed as you type, a project manager, code snippets, large built-in code-libraries, bracket/tag highlighting (matched and unmatched), and numerous tools and wizards to speed up your development.

The strength of PHP lies in its ability to embed it into, and together with, other web languages and technologies. Therefore, PHP Designer 2007 not only supports PHP, but also other web languages such as HTML, MySQL, XML, CSS, JavaScript, VBScript, Java, C#, Perl, Python and Ruby!

Support for WAMP/LAMP and AJAX developers is provided through intelligent syntax highlighting that automatically switches between syntax highlighters for PHP, HTML, CSS, and JavaScript depending on your position in the document, and dims the rest of the code!

Getting detailed information about more than 3000 native PHP functions on the fly makes the learning curve for PHP as easy and short as possible.

Jump to any declaration in both source and project files with filtering by class, interface, function, variable or constant.

Easily document your code with phpDocumentor, the standard documentation tool for the PHP language, with the built-in phpDocumentor Wizard.

The philosophy behind PHP Designer 2007 is that it be enjoyable to use for hours and hours for both beginners and professional developers through an easy-to-use, intuitive, customizable and tabbed

interface. The professional edition is for any commercial, business, educational, institutional or governmental purpose!

Editores e IDEs para PHP, CSS, JavaScript e HTML/Kate

De Wikibooks

4.3 - Kate

Editor com suporte para um grande número de linguagens programação.

Seu plugin "Auto-completar palavras" apresenta uma grande produtividade, maior até que grandes IDEs.

Site Oficial

http://www.kate-editor.org/

- Freeware apenas for Linux (KDE)

- HighLighting da sintaxe

- Numeração de linhas

- Reconhece chaves (início e final)

- Auto-completa todas as palavras (via Plugin)

- Edição em servidores remotos via FTP e SFTP

Editores e IDEs para PHP, CSS, JavaScript e HTML/Dreamweaver

De Wikibooks

4.4 - Dreamweaver

IDE muito popular entre os designers ecom suporte ao PHP, inclusive também gera código.

Site Oficial

https://store1.adobe.com/cfusion/store/index.cfm?store=OLS-US&NR=0#loc=en_us&view=ols_prod&store=OLS-US&categoryOID=1147942&distributionOID=105&NR=0

Versão comercial para Windows e MAC.

Dreamweaver 8 is the industry-leading web development tool, enabling users to efficiently design, develop and maintain standards-based websites and web applications.

Editores e IDEs para PHP, CSS, JavaScript e HTML/Zend Studio

De Wikibooks

4.5 - ZendStudio

Uma das mais populares IDEs para PHP.

Comercial, com versões para Windows, Linux e Mac.

Site Oficial

http://www.zend.com/products/zend_studio

Principais Recursos

Code Completion, Code Folding, PHPDoc / PHPDocumentor Support, Code Templates, Debugger Profiler, Firewall/NAT Traversal, Proxy Settings, PHP Code Analysis, Smart Goto Source, Navigation Web Services Support (SOAP), Java, Embedded Internet Explorer, Code Snippets, SQL Support, Find in Files, Zend Platform Integration, Zend Framework Integration, Source Control, CVS Integration, Subversion Integration, FTP Advanced Integration, Help, Languages

Editores e IDEs para PHP, CSS, JavaScript e HTML/VS.PHP

De Wikibooks

4.6 - VS.PHP

IDE tendo como base o Visual Studio da Microsoft, por isso VS.PHP.

Site Oficial

http://www.jcxsoftware.com/

Versão Comercial somente para Windows.

VS.Php is a Php integrated development environment based on Visual Studio. With VS.Php you can design, develop, debug and deploy Php applications within the Visual Studio IDE. VS.Php key features are around providing rich Php and Smarty editing capabilities as well as its ability to debug Php scripts locally as well as remotely.

Editores e IDEs para PHP, CSS, JavaScript e HTML/Delphi for PHP

De Wikibooks

4.7 - Delphi for PHP

Uma das últimas IDEs a aparecer para PHP.

Versão comercial para Windows.

Site Oficial

http://www.codegear.com/products/delphi/php

The RAD visual PHP development environment

```
   * Accelerate PHP web development with Delphi's proven visual (RAD) framework
for PHP
   * Simplify PHP programming with Delphi's comprehensive PHP for VCL libraries
   * Sophisticated editing and integrated debugging speeds up the process of
tracking down bugs and errors in the code
   * Integrated development environment streamlines code management and
navigation so managing and organizing PHP code has never been easier.
```

Tutorial - Uma introdução ao Delphi for PHP -
http://webinsider.uol.com.br/index.php/2007/06/10/uma-introducao-ao-delphi-for-php/

Bom artigo sobre a ferramenta no iMasters -
http://www.imasters.com.br/artigo/5990/php/delphi_for_php//imprimir/

Outro bom artigo - http://www.activedelphi.com.br/print.php?sid=489

Editores e IDEs para PHP, CSS, JavaScript e HTML/KDevelop

De Wikibooks

4.8 - KDevelop

O KDevelop é uma IDE for Linux (KDE) com suporte a diversas linguagens de programação, inclusive ao PHP.

Vejamos algumas dicas de como utilizá-lo com PHP.

Passos para a instalação no Linux Ubuntu (versão 6.0.6).

Requisitos

```
- Vamos adicionar todos os repositórios:

- Abrir o Synaptic - Configurações - Repositórios

- Em Mídia de instalação marque todos

- Clique em Adicionar, marque os 4 componentes e novamente Adicionar

- Clique em Adicionar novamente e agora selecione o segundo Canal e os 4
componentes e Adicionar. Repita para os 4 canais então clique em Fechar.

- Clique no botão Recarregar para atualizar os repositórios e feche o Synaptic.
```

Instalar o Kdevelop para PHP

```
- Clique no botão Aplicações - Adicionar/Remover

- Selecione o grupo Programação à esquerda

- Em aplicações acima selecione KDevelop: 'Scripiting'

- Clique em Aplicar e novamente Aplicar e então em OK
```

Configurações

Configurando para nossa acentuação:

- Tools - Encoding - Western European - ISO-8859-1

Configurando para PHP

- Tools - Highlighting - Scripting - PHP (HTML)

Configurações Diversas no Editor

- Setting - Configure Editor - Editing - Tab width - 4

- Open/Save - Encoding - Western European (ISO-8859-1)

- Highlighting - Highlight - Scripts/PHP (HTML)

Usando

- Abra o KDevelop Scripting que agora encontra-se no grupo Programação

Preparar o KDevelop para trabalhar com PHP

- Clique em Project - Project Options

- À esquerda clique em PHP Specific

Em General - "Parser PHP &Include Path" indique "/opt/lampp/bin/php-5.1.4" em sendo seu PHP instalado via Xampp, caso contrário veja em seu php.ini qual o include_path.

- Na aba Invocation, marque "Using a existing webserver".

- Na aba Webserver indique http://localhost/cadclientes/ ou outro, se for seu caso.

- Na aba Shell, em PHP Executable indique "/opt/lampp/bin/php-5.1.4", ou outro, se for seu caso.

```
Em PHP ini File indique "/opt/lampp/etc/php.ini".

Caso utilize indique "Load Zend Extension".
```

À esquerda clique em Doxygen. À direita em Output language selecione Portuguese.

Em Input indique o diretório onde irá armazenar os arquivos.

Criando um Projeto

- Clique em Project - New Project
- Expanda a pasta PHP e selecione Simple PHP Script
- Em Application name digite um nome para a aplicação (Exemplo: cadclientes)
- Clique em Next
- Em Project Options clique em Next
- Em version control system clique em Next
- Em template clique em Finish
- Vamos testar e clique no botão Run (aquele azul com uma roldana) ou tecle F9.
- Será exibida a mensagem Hello World!, como resultado do script gerado pelo KDevelop.

Adicionar Novo arquivo ao Projeto

- Clique em File - New ou Tecle Ctrl+N para abrir o diálogo "New File".
- Clique em Blank File e selecione PHP.
- Digite o nome do arquivo (Ex.: index.php)
- Deixe marcada a checkbox "Add to project".
- Abra o arquivo criado com File - Open ...

Importando Projetos PHP

- Project
- Import Existing Project - Project Type - PHP Application

Criação de Snippets

Muito útil para armazenar dicas, funções e classes.

Editores e IDEs para PHP, CSS, JavaScript e HTML/PSPAD

De Wikibooks

Site oficial - http://www.pspad.com/en/

Editor para vários tipos de linguagens: HTML, PHP, Pascal, JScript, VBScript, MySQL, MS-Dos, Perl,...

Com várias boas características:

```
   * do you work with various programming environments ?
   * do you like highlighted syntax in your source code ?
   * do you need a small tool with simple controls and the capabilities of a
mighty code editor ?
   * are you looking for a tool that handles plain text ?
   * do you want to save money and still have the functionality of professional
products ?
   * then PSPad is for you.
```

```
   * work with projects
   * work with several documents at the same time (MDI)
   * save desktop sessions to later reopen all session files
   * FTP client - edit files directly from the web
   * macro recorder to record, save and load macros
   * search and replace in files
   * text difference with color-coded differences highlighted
   * templates (HTML tags, scripts, code templates...)
   * installation contains templates for HTML, PHP, Pascal, JScript, VBScript,
MySQL, MS-Dos, Perl,...
   * syntax highlighting according to file type
   * user-defined highlighters for exotic environments
   * auto correction
   * intelligent internal HTML preview using IE and Mozilla
   * full HEX editor
   * call different external programs for different environments
   * external compiler with output catcher, log window and log parser for an
"IDE" effect in every environment
   * color syntax highlight printing and print preview
   * integrated TiDy library for formatting and checking HTML code, conversion to
CSS, XML, XHTML
   * integrated free version of the top CSS editor TopStyle Lite
   * export with highlight to RTF, HTML, TeX format to file or clipboard
   * column block select, bookmarks, line numbers, ...
   * reformat and compress HTML code, tag character case change
```

```
* line sorting with ability to sort on defined columns and drop duplicates
* ASCII chart with HTML entities
* Code explorer for Pascal, C/C++, INI, HTML, XML, PHP and more in development
* spell checker
* internal web browser with APACHE support
* matching bracket highlighting
* ...(many more features, too numerous to list)
```

Editores e IDEs para PHP, CSS, JavaScript e HTML/Quanta

De Wikibooks

O Quanta+ é uma IDE para desenvolvimento de projetos de Sistemas de Aplicação que, através de sua configuração, permite seu ajuste a muitos modos de interação com várias linguagens de programação.

Seu universo de execução é o sistema operacional Linux, e está disponível para muitas distribuições. É comum encontrá-lo como pacote padrão de Fedora, Mandriva, RedHat, Ubuntu, Kurumin e muitas outras.

Editores e IDEs para PHP, CSS, JavaScript e HTML/BlueFish

De Wikibooks

Editor BlueFish for Linux

Um editor web for Linux com support a PHP e a diversas outras linguagens de programação.

Site oficial - http://bluefish.openoffice.nl/index.html

Disponível como binário para a maioria das distribuições.

Suas características - http://bluefish.openoffice.nl/features.html

Bluefish - um poderoso editor para web designers

Bom tutorial em português sobre o BlueFish - http://www.vivaolinux.com.br/artigos/impressora.php?codigo=1079

Editores e IDEs para PHP, CSS, JavaScript e HTML/PHPEdit

De Wikibooks

IDE com capacidade de debug remoto.

Possui sintaxe para JavaScript, PHP, HTML. Ha uma caracteristica interessante: Ele faz o highlight apenas da linguagem sendo selecionada deixando as demais em cinza claro de forma a destacar aquilo que se está editando como foco.

Ferramenta leve que é uma alternativa mais barata que o pacote da Zend.

http://www.waterproof.fr/products/PHPEdit

Referências/Windows

De Wikibooks

Conteúdo

- 1 Windows na Wikipedia
- 2 Instalação do Windows XP Professional SP2 em Português do Brasil
  - 2.1 Criar antes uma partição para backup
  - 2.2 Particionamento
  - 2.3 Baixar os seguintes softwares antes de instalar o windows:
  - 2.4 Recomendação
  - 2.5 Instalação
  - 2.6 Concluir
  - 2.7 Configurações
  - 2.8 Otimizações de Desempenho e Segurança
  - 2.9 Instalando Alguns Dispositivos não instalados
  - 2.10 Algumas teclas de Atalho úteis
  - 2.11 Utilitários Importantes

Windows na Wikipedia

http://pt.wikipedia.org/wiki/Windows

Microsoft Windows é uma popular família de sistemas operacionais criados pela Microsoft, empresa fundada por Bill Gates e Paul Allen.

O Windows é um produto comercial, com preços diferenciados para cada uma de suas versões, embora haja uma enorme quantidade de cópias ilegais instaladas, ele é o sistema operacional mais usado do mundo.

Apesar do sistema ser conhecido pelas suas falhas críticas na segurança e como plataforma de vírus de computador e programas-espiões (spywares), o impacto deste sistema no mundo atual é simplesmente incalculável devido ao enorme número de cópias instaladas.

Instalação do Windows XP Professional SP2 em Português do Brasil

Criar antes uma partição para backup

Uma boa recomendação é para se criar pelo menos duas partições, uma para o Windows e outra para backup.

Caso se tenha um HD de 80 GB, sugiro 25 ou 30 GB para a partição do Windows e o restante para Backup.

Caso queira instalar também o Linux veja a sugestão:

swap - 512MB ou 1GB / - 20GB Windows - 20 Backup - Restante

Obs.: Sempre instale primeiro o windows e depois o Linux, pois ao ser instalado o Windows sobrescreve a área de boot (MBR) o que nos deixa sem acesso direto ao Linux.

Particionamento

Uma boa opção gratuita de programa que redimensiona partições é o GParted (Gnome Partition Editor), que oferece inclusive um live CD, que vem com o Linux Slackware e o GParted.

http://gparted.sourceforge.net/

Live CD - http://gparted.sourceforge.net/livecd.php

Baixar os seguintes softwares antes de instalar o windows:

- Firewall Zone Alarme - http://www.zonelabs.com/
  `http://www.zonelabs.com/store/content/company/products/znalm/freeDownload.jsp`

- Clamav for Windows - http://www.clamwin.com/

- Spybot - http://www.safer-networking.org/en/download/index.html

- Editor, corretor do Registro do Windows - sugestão MV RegClean -
  `http://www.velasco.com.br/` ou

  `http://superdownloads.uol.com.br/download/i17746.html`

- Atualmente o Google está oferecendo diversos utilitários muito bons, como o Norton Security Scan (Detecta e elimina vírus e worms da Internet e Atualizações gratuitas da detecção e verificação programada), como também o Spyware Doctor Starter Edition (Utilitário Anti-spyware Detecta/rem. spyware, adware, cavalos de Tróia e keyloggers, Inclui Smart Updates e programador para proteger o seu PC) além de outros bons utilitários. Caso seu micro tenha 512 MB de RAM e Internet banda larga prefira usar o Norton do Google e também o Spyware Doctor. Endereço:

```
http://pack.google.com/intl/pt-br/pack_installer_required.html?hl=pt-br&gl=br
```

- Br Office (Office grátis) - www.openoffice.org.br

- FireFox (navegador grátis) - www.mozilla.org

- ThunderBird (cliente de e-mail grátis) - www.mozilla.org

- InkSkape - Editor gráfico profissional gratuito - http://www.inkscape.org/

Recomendação

: Caso seja a primeira vez que realiza uma instalação do Windows, leia com atenção as recomendações do instalador.

Instalação

1 - Dar boot com CD do windows no drive e com boot pelo CD habilitado

2 - Fique atento e quando solicitado pressione qualquer tecla para iniciar pelo CD e aguarde... Observe as mensagens para instalação de outros dispositivos (F6 para instalar SCSI por exemplo)

3 - Bem vindo

```
Tecle Enter
```

4 - Licença Tecle Enter

5 - Selecionar partição NTFS ou FAT32 Caso já tenha uma partição com windows instalado, pressione Esc para instalar novamente sobre a partição existente.

6 - Selecione a partição onde deseja instalar e tecle D para deletar

7 - Tecle Enter para continuar a exclusão

8 - Tecle L para realmente excluir

9 - Selecione o espaço escluído e tecle C para criar uma nova partição

10 - Tecle enter para criar com todo o espaço ou entre com o tamanho e Enter

11 - Selecione a partição criada e tecle Enter

12 - Formatar com NTFS (recomendada), Enter e aguarde...

13 - Aguarde a cópia dos arquivos para a instalação...

14 - Deixe reiniciar agora pelo HD e aguarde a instalação ...

15 - Opções regionais e idioma

Clique em Avançar e opós a instalação faça os ajustes necessários

16 - Digite nome, organização e Avançar

17 - Entre com a chave e Avançar

18 - Entre com:

- Nome do computador (importante se em rede)

- Senha do administrador

- Confirme e avançar

19 - Ajuste fuso horário, data e hora e Avançar e aguarde...

20 - Configurações de rede

- Caso esteja em rede e seja necessário, clique em Configurações personalizadas e entre com as informações

- Caso contrário clique em Avançar

21 - Domínio em grupo

- Entre com dominio se em rede

- Caso contrário apenas clique em Avançar e aguarde... (o micro irá reiniciar pela primeira vez)

22 - Bem-vindo

Clique em Avançar

23 - Caso tenha pouca memória e/ou prefira um melhor desempenho clique em Não agora

Caso contrário clique na primeira opção e Avançar

24 - Como este computador se conecta à Internet?

Marcar ou ignorar

25 - Registro

Selecionar e Avançar

26 - Usuário(s)

Digite pelo menos um e Avançar

27 - Obrigado

Concluir

Barreiras de Proteção para Suprir as Deficiências de Segurança do Windows

Antes de sua primeira conexão à Internet, instale os programas sugeridos no início.

Após a instalação, reinicie o micro e então os atualize em sua primeira conexão.

Configurações

Mudar Resolução para 1024 X 768

- Clique com o botão direito numa área livre da tela - Propriedades - Configurações

- Em Resolução da tela arraste para 1024 X 768 ou outra desejada

- Clique em OK e confirme

Alterar menu iniciar para o tipo clássico (aquele usado pelo Windows 98)

- Clicar com botão direito sobre botão Iniciar - Propriedades

- Clicar em "menu iniciar clássico"

Exibir Internet Explorer e outros na Barra de Tarefas

- Clicar com botão direito sobre botão Iniciar - Propriedades - Aba Barra de Tarefas

- Mostrar barra "Inicialização Rápida"

Otimizações de Desempenho e Segurança

- Veja o tutorial do Carlos Morimoto em seu site, seção Tutoriais:

http://www.guiadohardware.net/tutoriais/049/print.php

Obs.: Tome cuidado para não desabilitar serviços importantes.

Instalando Alguns Dispositivos não instalados

- Vá ao painel de controle - Sistema - Hardware - Gerenciador de Dispositivos

- Remova todos os itens com uma interrogação amarela e tenha o CD com o respectivo driver em mãos

- Insira o CD com o driver do dispositivo e o instale

Algumas teclas de Atalho úteis

B = Bandeirinha do windows

B + E - Abrir o windows explorer

B + M - Minimiza todas as janelas abertas

B + U - Abre o Gerenciador de utilitários para portadores de deficiência

Ctrl+Esc - Abre o menu iniciar

Alt+Tab - Alterna entre as janelas abertas

Utilitários Importantes

- Desfragmentação

- Recuperação do sistema (Acessórios - Ferramentas de Sistema)

Em caso de algum problema de difícil solução podemos com esta ferramenta "voltar o tempo" para antes da ocorrência do problema e assim resolver de forma mais cômoda o problema.

Apenas o hardware e software instalados serão removidos, os arquivos criados ou copiados não serão afetados e a operação é reversível, como também podermos trocar para outro ponto de recuperação.

Em caso de problema pode ser útil, mas o ideak é sempre realizar backup dos seus dados importantes.

Referências/Linux

De Wikibooks

Conteúdo

- 1 Linux na Wikipedia
- 2 Instalação do Ubuntu 6.06
- 3 Instalação do GParted (particionador)
- 4 Dicas sobre o Linux Ubuntu
- 5 Configurando um servidor de email com Postfix e outros
- 6 Uso e Recuperação de Pendrives, MP3, Câmeras, etc
- 7 Permissões - referência rápida
- 8 Sistema de Vigilância Digital com ZoneMinder
- 9 Gravar CD de música pelo K3B
- 10 Evitando gargalos de hardware
- 11 VirtualBox Opção Livre para Virtualização
- 12 Personalizando a Tela de Login do Ubuntu 7.04
- 13 Instalando sem complicações o Java no Ubuntu 6.06, 6.10 e 7.04

Linux na Wikipedia

http://pt.wikipedia.org/wiki/Linux

Linux refere-se a qualquer sistema operacional do tipo Unix que utiliza o núcleo Linux. É um dos mais proeminentes exemplos de desenvolvimento com código aberto e de software livre. O seu código fonte está disponível sob GPL para qualquer pessoa utilizar, estudar, modificar e distribuir livremente.

GNU/Linux refere-se a qualquer sistema operacional do tipo Unix que utiliza o núcleo Linux e também os programas de sistema GNU. Como os casos de sistemas de núcleo Linux sem os programas de sistema GNU são raros, freqüentemente GNU/Linux e Linux são sinônimos.

Inicialmente desenvolvido e utilizado por nichos de entusiastas em computadores pessoais, o sistema Linux passou a ter a colaboração de grandes empresas, como a IBM, a Sun Microsystems, a Hewlett-Packard, e a Novell, ascendendo como principal sistema operacional para servidores -- oito dos dez serviços de hospedagem mais confiáveis da Internet utilizam o sistema Linux em seus servidores web.

Linux tornou-se o sistema capaz de funcionar no maior número de arquiteturas computacionais possíveis. É utilizado em aparelhos variando desde supercomputadores, até celulares, e vem ganhando popularidade no mercado de computadores pessoais.

Instalação do Ubuntu 6.06

Esta instalação refere-se à versão 6.06 do Ubuntu mas é semelhante às versões posteriores, pelo menos até a versão 7.04:

Nesta instalação considera-se que já temos o windows instalado na máquina, o que representa alguma diferença.

Alerta: no caso de se ter dados importantes no HD, quando se deve fazer backup em CD ou de outro tipo de mídia, pois no momento do particionamento sempre se corre o risco de excluir acidentalmente e indevidamente alguma ou todas as partições.

Caso não existisse o Windows, uma sugestão para um HD de 80 GB seria:

1 - swap (512 MB)

2 - / (raiz, 20 GB)

3 - /home (restante)

A partição home funciona como backup e sempre que se precisar formatar a raiz, preservar a home.

Instalação

1 - Dar boot com CD no drive (com boot pelo CD habilitado)

2 - No primeiro menu texto que aparece apensa tecle Enter e aguarde

3 - Quando entrar no ambiente gráfico e estiver pronto, clique no botão Install

4 - Idioma - Selecionar e avançar

5 - Fuso Horário - Selecione com o mouse e avance

6 - Teclado - Selecione e avance

7 - Usuário - Entre com nome, login, senha e nome do computador (importante somente se numa rede) e avançar

8 - Tipo de Instalação

- Editar manualmente a tabela de partições

Observação - Para HD virgem escolha a primeira opção

9 - Preparar Partição

Caso queira rdimensionar alguma partição, selecione-a e com o botão direito escolha redimensionar

Avançar

10 - Preparar Pontos de Montagem

Caso seu HD tenha partição que deseja preservar fique muito atento nesta etapa, especialmente deverá identificar as partições que não deseja reformatar.

Caso esteja em dúvida:

- Acesse o segundo terminal com "Ctrl+Alt+F2"

- Digite: "sudo fdisk -l"

- Anote o número das partições

- Volte para o terminal gráfico com "Ctrl+Alt+F7"

Somente após escolher com segurança os pontos de montagem marque os checkboxs Reformatar.

Caso seu HD seja virgem ou não tenha nada para preservar e não saiba o que fazer, uma sugestão é:

1 - swap (512 MB)

2 - / (raiz, 20 GB)

3 - /home (restante)

11 - Pronto para Instalar

Mais uma oportunidade para checar as partições que serão reformatadas.

Verifique e em caso de engano volte e corrija.

Clique em Install e aguarde...

12 - Instalação Finalizada

Clique em Reiniciar agora

Quando o CD for ejetado, remova- e tecle Enter

13 - Logo no primeiro login (se conectado à Internet) poderá realizar as atualizações clicando no pequeno botão laranja.

14 - Realização de algumas configurações restantes:

- Sistema - Administração - Rede

15 - Instalar alguns programas extras:

Adicionar/Remover (lembre de marcar os dois checkbox: Instalar programas não suportados e Instalar programas ...)

K3B - Gravação de CD/DVD

KolourPaint

Instalação do GParted (particionador)

Tutorial Passo-a-passo de Uso do Gparted

O Gparted é um particionador free que oferece um liveCD para esta função.

Suporta diversos tipos de sistemas de arquivos: ext2, ext3, raiserfs, ntfs, fat32, etc.

A distribuição do liveCD é o Slackware e o iso tem apenas 27,9MB.

Site oficial - http://gparted.sourceforge.net/

Live CD - http://gparted.sourceforge.net/livecd.php

Aqui vou considerar que tenho dois HDs e pretendo excluir todas as partições de um e criar uma única.

O primeiro HD está como master na porta primária, portanto o hda no Linux.

O segundo HD está como master na secundária, portanto hdc. Muito cuidado no momento de excluir as partições, para selecionar o HD correto.

1 - Dar boot com o liveCD

2 - Ao surgir o prompt apenas tecle Enter

3 - Language

Selecione portuguese e Enter

4 - Keymap

Selecione querty/br-abnt2.map para tipo ABNT2

5 - Display depth

Apenas Enter

6 - XRES

Enter e aguarde...

7 - Acima e à direita existe uma combo contendo os HDs existentes. Por default aparece o hda. No meu caso vou selecionar o hdc.

- Esta é a etapa em que também podemos redimensionar alguma partição. Para isso apenas

selecionamos a partição e clicamos sobre a mesma com o botão direito e Resize/Move.

Então clicamos na seta para a direita e arrastamos até o tamanho desejado. E clicamos em Aplicar.

- Muita atenção nesta etapa para realmente selecionar o HD correto.

- Selecionar cada uma das partições a Apagar e clicar em Apagar.

- Caso tenha partições extendidas, antes deve apagar todas as suas partições lógicas e só então excluir a extendida.

- Após excluir todas clicar em Aplicar, confirmar e Fechar.

- Então selecionar o espaço "unallocated" e clicar em Novo.

- Podemos informar o tamanho da nova partição ou aceitar o total

- Podemos também selecionar o tipo de sistema de arquivos

- Finalmente clicar em Aplicar, confirmar e Fechar

8 - Clicar no menu GParted e Sair

9 - Clicar no botão vermelho do canto direito

10 - Selecionar a opção Eject e Reboot e OK

11 - Caso não ocorra a reinicialização pressione o botão de reset.

Pronto, seu HD está particionado.

Dicas sobre o Linux Ubuntu

Estas dicas também se aplicam à quase todas as distribuições baseadas na distribuição Debian.

Site oficial do Ubuntu - http://www.ubuntulinux.org

Formatar Pendrive

sudo umount /dev/sda1

sudo mkdosfs -F 32 -n Tiago /dev/sda1 (Tiago é o nome do volume)

Ou podemos usar o fdisk para remover e criar novamente a partição

Usando apt - Gerencia pacotes dos repositórios (instala, remove, etc)

Instalar pacote, inclusive dependências. Podemos usar * no nome:

sudo apt-get install nomepacote

Atualiza lista de repositórios

sudo apt-get update

Atualiza todos os pacotes da distribuição

sudo apt-get upgrade

Atualiza todos os pacotes da distribuição para uma nova versão

sudo apt-get dist-upgrade

Desinstalar pacote

sudo apt-get remove nomepacote

Remover pacote e seus scripts de configuração

apt-get --purge remove nomepacote

Instalar xorg e dependências

sudo apt-get build-dep xserver-xorg

Instalando dependências

sudo apt-get install -y nomedependeicia1 nomedependeicia2

Corrigir Falhas nos Arquivos Instalados

sudo apt-get install -f

Procurar ocorrências de nome nos repositórios

sudo apt-cache search nome

Exibir informações sobre o pacote

sudo apt-cache show package

Exibir informações sobre as dependências do pacote

sudo apt-cache depends package

Instalar GCC e cia

sudo apt-get install build-essential

sudo apt-get install g++-3.4 export CC=/usr/bin/gcc-3.4 sudo apt-get install linux-headers-$(uname -r)

Suporte a Java no Mozilla

sudo apt-get install j2re1.4-mozilla-plugin

Suporte a Flash no Firefox

sudo apt-get install swf-player

Instalar fontes do Java

sudo apt-get source java-package

Remover configurações e remover o pacote

sudo apt-get --purge remove nomepacote

Atualizar Firefox no Ubuntu para pt-br

sudo apt-get install mozilla-firefox-locale-pt-br

Remover arquivos não mais necessários

sudo apt-get autoremove

Usando dpkg

Instala pacotes localmente. Antes devemos ter feito download dos pacotes .deb

dpkg -i nomepacote.deb // Instalar pacote

dpkg -r nomepacote.deb // Desinstalar pacote

dpkg-deb --contents nomepacote.deb // Exibir o conteúdo de um pacote

dpkg-deb -x nomepacote.deb // Extrai o conteúdo deum pacote

dpkg-reconfigure xserver-xorg //RECONFIGURAR PLACA DE VÍDEO (ambiente gráfico)

dpkg --purge nomepacote // Remove inclusive scripts de configuração

dpkg --search nomearquivo // Procurar em que pacote está um arquivo

URL sobre dpkg - http://pt.wikipedia.org/wiki/Dpkg

Repositório de todos os pacotes do Ubuntu, com pacotes .deb (muito bom):

http://packages.ubuntulinux.org/

- Digite o nome do pacote em Keywork
- Clique em Search
- Clique no link mais adequado à sua busca
- Role a tela e clique na sua arquitetura. Algumas vezes All.
- Clique agora num dos espelhos de pacotes para fazer o download.

Criação de Disquete de Boot

fdformat /dev/fd0

ou

mformat a:

dd if=bootfloppy-usb.img of=/dev/fd0 bs=8192

dd if=boot.img of=/dev/fd0 bs=1024 conv=sync ; sync

dd if=bootfloppy-usb.img of=/dev/fd0 bs=36b

Criando o disco de boot automaticamente

grub-floppy /dev/fd0

Manualmente

cd /lib/grub/i386-pc

dd if=stage1 of=/dev/fd0 bs=512 count=1

dd if=stage2 of=/dev/fd0 bs=512 seek=1

Converter utf-8 para iso8859-1

No linux

iconv -f utf-8 -t iso8859-1 arquivo

Reinstalando o GRUB

grub-install /dev/hda

Identificando Distribuição

dmesg | head -1

sudo cat /proc/version

cat /etc/issue

Dicas do Fábio B. Pinto para as ótimas Dicas-L

Multimídia

Conversor de Vários Formatos de Vídeo/Som

sudo apt-get install soundkonverter

Converter Wav para MP3

sudo apt-get install audacity

For windows free - http://www.vuplayer.com e http://www.mediamonkey.com/download.htm

HOW-TO rodar Adobe Photoshop CS2 on Ubuntu

http://blog.publicidadpixelada.com/2006/10/10/how-to-adobe-photoshop-cs2-on-ubuntu-10-steps/

VLC - Player com diversos codecs

apt-get install vlc

apt-get install mozilla-plugin-vlc

K3B com plugins

sudo apt-get install k3b libk3b2-mp3 k3b-i18n

sudo apt-get install normalize-audio toolame movixmaker-2

sudo apt-get install vcdimager vorbis-tools sox transcode

sudo ln -s /usr/bin/normalize-audio /usr/bin/normalize

Renomeando nomes de arquivos com diferentes codificações

apt-get install convmv

convmv -f iso-8859-1 -t UTF-8 ação.txt --notest

Renomeará para aÃ§Ã£o.txt

CODECS extras para o Ubuntu

Instalar todos os pacotes sugeridos em:

https://wiki.ubuntu.com/RestrictedFormats?highlight=%28codecs%29%7C%28format%29

sudo apt-get install wget

wget -c http://www.debian-multimedia.org/pool/main/w/w32codecs/w32codecs_20061022-0.0_i386.deb

sudo dpkg -i w32codecs/w32codecs_20061022-0.0_i386.deb

sudo apt-get install gstreamer0.10-pitfdll gstreamer0.10-ffmpeg gstreamer0.10-plugins-bad gstreamer0.10-plugins-bad-multiverse gstreamer0.10-plugins-ugly gstreamer0.10-plugins-ugly-multiverse gxine libxine-main1 libxine-extracodecs ogle ogle-gui

Tocando MP3 no Ubuntu (e outras distros) - da Dicas-L Colaboração: Henrique Meira

O Ubuntu (e outras distros) por padrão não toca som MP3, para resolver isto basta instalar um plugin do GStreamer do conjunto "ugly".

No Ubuntu, você pode abrir o Gerenciador de Pacotes Synaptic e tentar localizar o pacote 'gstreamer0.10-plugins-ugly'.

Ou via linha de comando:

apt-get install gstreamer0.10-plugins-ugly

Este pacote está na seção Universe.

E isso já resolve tudo, já pode abrir o Rhythmbox por exemplo e ouvir suas músicas.

Criar imagem ISO de arquivos ou diretório

mkisofs -r -o nomedaimagem.iso /caminho/diretorioorigem/

Gravação de CD (ISO) via linha de comando

cdrecord dev=/dev/cdrw -v -eject nomeimagem.iso

Criando Imagem ISO via linha de comando

dd if=/dev/cdrw of=nomeimagem.iso

Montar Imagem ISO em partição

mount -o loop nomeimagem.iso /dev/hda3

Renomeando Múltiplos Arquivos via Shell

Colaboração: Euriam Barros de Araujo

Em algumas ocasiões, necessitamos renomear vários arquivos que possuem um determinado padrão.

Para facilitar este trabalho, segue o script abaixo:

```
for i in `ls files`
do
 x=${i#*left_padrao}
 y=${x%right_padrao*}
 z=${i%file_name*}
 mv $i $y-$z.extensao
done
```

Variáveis:

```
  * x: armazena a partir do padrão, removendo tudo à esquerda do mesmo
  * y: armazena o padrão, removendo tudo à direta do mesmo contido em x
  * z: armazena o nome do arquivo, exceto o padrão e sua extensão

    ==Exemplo:==
```

No diretório atual existem os arquivos abaixo e se deseja deslocar o tipo do trabalho para o início do arquivo:

```
trabalho-A.sxw

trabalho-B.sxw

trabalho-C.sxw

trabalho-D.sxw

for i in `ls trabalho*`
```

```
do
 x=${i#*-}
 y=${x%.*}
 z=${i%-[A-D]*}
 mv $i $y-$z.sxw
done
```

Renomeando Múltiplos Arquivos via Batch do DOS/Windows:

ren *.html *.php

Codificação de caracteres no Ubuntu Edgy:

Pesquisando na internet, reuni algumas dicas as quais funcionaram bem no meu computador.

1. Instalar locale com encoding pt_BR.ISO-8859-1:

sudo /usr/share/locales/install-language-pack pt_BR

2. Alterar o arquivo /etc/environment para:

LANG="pt_BR.ISO-8859-1"

3. Reconfigurar locales:

sudo dpkg-reconfigure locales

Lembrar de reiniciar o X para que as alterações tenham efeito.

Após restartar acessar um terminal e digitar:

locale

Para verificar como estão.

Julio Neto

Codificação de caracteres no Ubuntu 6.06

sudo /usr/share/locales/install-language-pack pt_BR

Instalar o arquivo localeconf:

sudo apt-get install localeconf

Rodar:

sudo dpkg-reconfigure localeconf

sim

ok

pt_BR ISO-8859-1

OK

Marcar todos

Em todos os próximos selecionar pt_BR ISO-8859-1

Restartar o ambiente gráfico.

Após restartar acessar um terminal e digitar:

locale

Para verificar como estão.

Julio Neto

Execução periódica de programa em linha de comando usando comando 'watch'

Caso você queira listar continuamente o conteúdo de um diretório para sua monitoração, basta digitar:

```
% watch -n 1 'ls -l'
```

Com a opção '-n' você pode especificar de quanto em quanto tempo o comando será executado pelo 'watch'.

Ao executar o comando acima, ele mostrará no topo da tela o comando que está em execução, data, e horário em tempo real, bem como a saída do comando logo abaixo.

Outro exemplo útil:

Mostrar os usuários logados no samba em tempo real:

```
% watch -n 1 'smbstatus -b'
```

Seleção de Faixa Vertical de Texto com vim ou gvim

Um recurso muito útil e raro nos editores de texto e processadores inclusive.

Exemplo de uso: baixei um arquivo cujas linhas são numeradas e gostaria de remover somente os números das linhas.

- Abra o arquivo no vim ou gvim

- Posicione o cursor no ínicio da faixa

- Tecle Ctrl+v

- Mova o cursor para o final da faixa

- Tecle Ctrl+v novamente

- Com o cursor para baixo mova até onde deseja selecionar

Listando 10 maiores diretórios por ordem decrescente

du -s * | sort -rn | head -10

O "du -s *" cria uma lista com os tamanhos e nomes dos objetos no diretório corrente - no caso de diretórios o tamanho dos objetos dentro dele serão somados (parametro "-s") , o "sort -rn" pega a lista gerada pelo du e ordena a primeira coluna na forma numérica (o "-n") e de forma inversa (o parametro "-r") e o "head -10" mostra somente as 10 primeiras linhas.

Assim se você vai caçar comedores de disco em uma conta, talvez seja melhor utilizar a construção (inclusive ocultos)

du -s `ls -A` | sort -rn | head -10

Nessa linha de trocar a expansão por uma lista, o céu torna-se o limite

Rodando Java 5

Instalar pelo Synaptic o Java JDK 5 da Sun

Procurar por sun e marcar sun-java5-jdk

Após isso exportar a variável de ambiente com:

export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun-1.5.0.06/

Que versão do Java Utilizar? Execute e selecione a versão desejada:

sudo update-alternatives --config java

Uso do RAR na linha de comando do Linux:

rar a destino.rar /origem/* -m0 -v1495m

onde: 0 é o nível de compactação (0 - 5)

1495 é o tamanho dos arquivos em MBytes

Dica de: Ricardo de Miranda Amorim - Engenheiro de Computação

Teclas de Atalho no Ubuntu

- Criar lançador para gconf-editor

- Executar

- Expandir apps

- Rolar e expandir metacity
- Clicar em global_keybindings à esquerda
- À direita clicar em run_command_1 e na mesma linha abaixo de Valor digitar a
  `combinação de teclas para o programa a abrir, como por exemplo <Control><Alt>k`
- Clicar em keybinding_commands e à direita digitar o nome do comando (se no path)
  `ou o caminho completo (exemplo kolourpaint).`

Editor de texto amigável para o prompt

nano

ou

mcedit (do pacote mc: sudo apt-get install mc)

Adicionar Senha de Root

sudo passwd root

Entrar com a senha de usuário (administrador)

Depois entrar com a senha de root e repetir.

Usar root temporariamente

sudo -s

Configurar Problemas de Acentuação

(somente para a versão 5.10):

Abrir o terminal, maximizar a tela e executar:

sudo dpkg-reconfigure locales

Desmarcar todos os itens e marcar apenas "Pt_BR-ISO 8859-1"

Alterar IP pela console

/etc/network/interfaces

Após alterar:

ifdown eth0

ifup eth0

Localizar Arquivos Rapidamente

sudo updatedb (executar apenas quando houver instalado ou desinstalado algo)

locate nomearquivo

Outra alternativa

whereis nomecomando (só funciona paracomandos executáveis do sistema operacional)

Bom Player de Filmes, MP3 e MP4

sudo apt-get install mplayer

Ativar DMA para o CDROM

hdparm -d1 /dev/cdrom

Para Instalar K3B dos Fontes

instalar xlibs-dev e kdebase-dev

apt-get install xlibs-dev

apt-get install kdebase-dev

Depois disso baixar os fontes, descompactar e compilar do site oficial.

Audotoria de Hardware

kinfocenter

sudo apt-get install hwinfo

Executar:

hwinfo --short > maquina.txt (Gera o arquivo com o hardware)

Outro bom:

sudo lshw > maquina.txt

Informações do hardware:

cat /proc/ide/hda/model

cat /proc/ide/hda/capacity

cat /proc/meminfo | grep MemTotal

cat /proc/cpuinfo

lspci

lspci -v

lspci -vv

ifconfig -a (dados da placa de rede)

netstat -na (quem está conectado nesta máquina)

netstat -nputa (portas e PID)

ntop -A (para dar uma senha)

ntop (informações da rede)

smbstatus - status do samba

dmesg |less

Listando Todas as Partições do Micro

sudo fdisk -l

Lista de Dispositivos PCIs

http://pciids.sourceforge.net/

- Guia Introdutório do Linux II

Configurando um servidor de email com Postfix e outros

- Configurando um servidor de email com Postfix, MySQL, SASL, MailScanner e MailWatch no CentOS

Uso e Recuperação de Pendrives, MP3, Câmeras, etc

Slax no Pendrive

- Formatando e Consertando o Pendrive no Linux

- Recuperando dados e fotos de câmeras digitais e pen drives

Permissões - referência rápida

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=6420

Sistema de Vigilância Digital com ZoneMinder

http://www.guiadohardware.net/tutoriais/sistema-vigilancia-zoneminder/

Gravar CD de música pelo K3B

http://www.vivaolinux.com.br/dicas/impressora.php?codigo=8742

Evitando gargalos de hardware

http://canecas.blogspot.com/2007/06/sucata-iv-evitando-gargalos-de-hardware.html

VirtualBox Opção Livre para Virtualização

http://localdomain.wordpress.com/2007/06/21/virtualbox/

Personalizando a Tela de Login do Ubuntu 7.04

http://www.vivaolinux.com.br/dicas/impressora.php?codigo=8822

Instalando sem complicações o Java no Ubuntu 6.06, 6.10 e 7.04

http://www.vivaolinux.com.br/artigos/impressora.php?codigo=5502

http://blogetilico.wordpress.com/2007/01/24/instalacao-e-configuracao-java-jdk-no-ubuntu-610/

http://www.herbetferreira.com/instalando-jdk-6-no-ubuntu-704-the-feisty-fawn/

Referências/HTML

De Wikibooks

Conteúdo

- 1 HTML na Wikipedia
- 2 Título - Este aparece como título no Navegador
- 3 Body - Entra as tags <body> e </body> aparece praticamente toda a área útil da página
- 4 H1 - Texto para Título de tamanho 1. Tem tamanhos até o 6 (H6)
- 5 Tag form - Formulários são muito utilizadados para recceber informações do visitante
    - 5.1 Radio - Observe a prática de usar o mesmo name para agrupar vários radios
    - 5.2 Campo Desabilitado e Somente Leitura
- 6 Formatação
- 7 Fontes
- 8 Links(âncoras)
- 9 Imagem
- 10 Incorporar Vídeo e Som
- 11 Títulos
- 12 Select Múltiplo
- 13 Upload
- 14 Linha Horizontal
- 15 Frames
- 16 Respeitando a posição de digitação com <pre> ... </pre>
- 17 Ótimo Tutorial sobre HTML do Tiago Daniel de Souza no site Linha de Código
- 18 HTML Wizards

HTML na Wikipedia

http://pt.wikipedia.org/wiki/Html

HTML (acrônimo para a expressão inglesa HyperText Markup Language, que significa Linguagem de Marcação de Hipertexto) é uma linguagem de marcação utilizada para produzir páginas na Web. Documentos HTML podem ser interpretados por navegadores. A tecnologia é fruto do "casamento" dos padrões HyTime e SGML.

HyTime é um padrão para a representação estruturada de hipermédia e conteúdo baseado em tempo. Um documento é visto como um conjunto de eventos concorrentes dependentes de tempo (como áudio, vídeo, etc.), conectados por hiper-ligações. O padrão é independente de outros padrões de processamento de texto em geral.

Título - Este aparece como título no Navegador

<HTML><HEAD><TITLE>Exemplo de Página HTML</TITLE></HEAD>

Body - Entra as tags <body> e </body> aparece praticamente toda a área útil da página

Terá texto verde, fundo branco e a imagem "background.gif" de fundo.

<body text="green" bgcolor="white" link="#FFFF00" vlink="#FF0000" background="http://www.seudominio.com/imagens/background.gif">

H1 - Texto para Título de tamanho 1. Tem tamanhos até o 6 (H6)

<H1 ALIGN="center">Exemplo de Página HTML</H1>

Tag form - Formulários são muito utilizadados para recceber informações do visitante

Veja que existem diversos controles nos forms: text, radio, select, textarea, etc.

```
<form method="POST" action="teste.php">
<input type="hidden" name="recipient" value>
Nome:<input type="text" name="nome" size="40" maxlength="40">
Senha:<input type="password" name="senha" size="10" maxlength="10">
```

Radio - Observe a prática de usar o mesmo name para agrupar vários radios

```
<input type="radio" name="estadocivil" value="solteiro">Solteiro
<input type="radio" name="estadocivil" value="casado">Casado
<input type="radio" name="estadocivil" value="desquitado">Desquitado
<input type="radio" name="estadocivil" value="naoinformado">Não informado

<input type="checkbox" name="personalidade" value="Hostil">Hostil
<input type="checkbox" name="maturidade" value="Adulto">Adulto

<textarea name="comentario" rows="4" cols="60" wrap="virtual" style="color:green;background-color:yellow">Texto default</textarea>
```

Campo Desabilitado e Somente Leitura

```
<input type ="text" value ="Valor" disabled readonly>

<select name=selecao>
<option name="leite" value=1 SELECTED>Leite</option>
<option name="cafe" value=2>Café</option>
<option name="cha" value=3>Chá</option>
</select>

<SELECT NAME="grupo">
<OPTION VALUE="1">Equipamento
<OPTION VALUE="2">Mão de Obra
<OPTION VALUE="3">Ferramentas
<OPTION VALUE="4" SELECTED>Material
<OPTION VALUE="5">Serviços Auxiliares
<OPTION VALUE="6">Transporte
</SELECT>
```

<input type="submit" name="Request" value="Enviar"><input type="reset" name="Clear" value="Limpar">
</form>

Formatação

Negrito Itálico <u>Sublinhado</u> <s>Riscado</s> ^{Sobrescrito} _{Subscrito}

Negrito *Itálico* Sublinhado ~~Riscado~~ Sobrescrito Subscrito

Fontes

```
<font COLOR="#00FF00">This text is Green</font>  <!-- ou "gree" -->
<font face="arial,helvetica,courier" size=1>Múltiplas Font Faces</font>
```

This text is Green
Múltiplas Font Faces

Links(âncoras)

<a HREF="http://www.2kweb.net" target="_BLANK">This links to 2K Communications
<a HREF="mailto:sales@2kweb.net">Click here to e-mail our Sales Department

Links para a página atual
1 - Criar destino - <a NAME="destino1">
2 - Criar link - <a HREF="http://yourdomain.com/index.html#destino1">Click here to go there

Imagem

<img src="http://www.2kweb.net/image-file.gif" alt="Explicação sobre a imagem" ALIGN="LEFT" height=50 width=50>

Incorporar Vídeo e Som

<bgsound src="http://www.2kweb.net/sound.mid" loop=0"> (IE em loop. loop=-1 para parar)

<embed src="http://www.2kweb.net/sound.mid">

Títulos

```
<h1>Heading Size 1 - Maior</h1>
<h2>Heading Size 2 - Tamanho 2</h2>
<h3>Heading Size 3 - Tamanho 3</h3>
<h4>Heading Size 4 - Tamanho 4</h4>
<h5>Heading Size 5 - Tamanho 5</h5>
<h6>Heading Size 6 - Menor</h6>
```

Tabelas

```
<table border="1" width="630">
<tr>
        <td width="630" colspan=2> Largura de 630 pixels </td>
</tr>
<tr>
    <td width="550">Largura de 550 pixels</td> <td width="80">Largura de 80
pixels</td>
</tr>
</table>
```

Largura de 630 pixels	
Largura de 550 pixels	Largura de 80 pixels

colspan = funde colunas

rowspan = funde linhas

```
<TABLE BORDER=1>
<TR><TH COLSPAN=2>Fusão de 2 colunas</TH></TR>
<TR><TD>linha1, coluna 1</TD><TD> linha 1, coluna 2</TD></TR>
<TR><TD>linha 2, coluna 1</TD><TD>linha 2, coluna 2</TD></TR>
<TR><TH ROWSPAN=3>Fusão de 3 linhas</TH><TD>uma linha</TD></TR>
<TR><TD>duas linhas</TD></TR>
<TR><TD>tres linhas</TD></TR>
</TABLE>
```

Fusão de Colunas	
linha1, coluna 1	linha 1, coluna 2
linha 2, coluna 1	linha 2, coluna 2
Fusão de 3 linhas	uma linha
	duas linhas
	tres linhas

Select Múltiplo

```
<select name="combomulti" size="4" multiple>
<option value="lista1">Seleção 1</option>
<option value="lista2">Seleção 2</option>
<option value="lista3">Seleção 3</option>
<option value="lista4">Seleção 4</option>
<option value="lista5" selected>Seleção 5</option> <!--dEFAULT-->
<option value="lista6">Seleção 6</option>
</select>
```

Pressiona-se Ctrl ou Shift enquanto seclica ou move as setas para selecionar. </BODY> </HTML>

Upload

```
<FORM NAME="form1">
Arquivo a enviar: <INPUT TYPE="file" NAME="myUploadObject">
<P>Receber as propriedades<BR>
<INPUT TYPE="button" VALUE="Nome"   onClick="alert('name: ' +
document.form1.myUploadObject.name)">
<INPUT TYPE="button" VALUE="Valor"   onClick="alert('value: ' +
document.form1.myUploadObject.value)">
<BR></FORM>
```

Linha Horizontal

```
<HR width="40%" align="center">
```

Frames

Criar arquivo index.html com:

```
<frameset rows="25%,75%" frameborder="1" framespacing="0">
        <frame src=superior.html name=superior>
        <frameset cols=50%,50%>
                <frame src=esquerda.html name=esquerda frameborder="1"
framespacing="0" scrolling="yes">
                <frame src=direita.html name=direita frameborder="1"
framespacing="0" >
        </frameset>
</frameset><br>
```

Arquivo esquerda.html:

```
<a href="conteudohist.html" target="direita">História</a><br>
<a href="conteudomat.html" target="direita">Matemática</a><br>
<a href="conteudoport.html" target="direita">Português</a><br>
```

Criar os três arquivos acima (conteudohist.html, conteudomat.html e conteudoport.html).

Veja também um pequeno assistente de frames em:

http://ribafs.net/down/tut/html/wizards/

Respeitando a posição de digitação com <pre> ... </pre>

O Código:

```
<pre>
      O que for digitado
  Será visto na mesma
  posição quando usarmos este comando.
<pre>
```

Acarretará:

```
      O que for digitado
  Será visto na mesma
  posição quando usarmos este comando.
```

Ótimo Tutorial sobre HTML do Tiago Daniel de Souza no site Linha de Código

Guia Prático de HTML - Parte 1

Guia Prático de HTML - Parte 2

Guia Prático de HTML - Parte 3

Guia Prático de HTML - Parte 4

Guia Prático de HTML - Parte 5

HTML Wizards

http://ribafs.net/down/tut/html/wizards/

Referências/JavaScript

Conteúdo

- 1 JavaScript na Wikipedia
- 2 Introdução
- 3 Sintaxe
- 4 Primeiro script JavaScript
- 5 Location - onde usar JS?
- 6 Exemplo no head
- 7 Externo - Importando um arquivo JS externo
- 8 Operadores
    - 8.1 Aritméticos
    - 8.2 Exemplo de Operadores Javascript com Variáveis
    - 8.3 Operadores de Comparação
- 9 Variáveis
- 10 Nomes de variáveis
- 11 Funções
- 12 Eventos
- 13 Tipos de Instruções (Estruturas de Controle)
- 14 if
- 15 else
- 16 else if
- 17 for
- 18 Comentários
- 19 Array
- 20 Ordenação de arrays
- 21 Alert
- 22 Confirm
- 23 Prompt
- 24 Print
- 25 Time Delay
- 26 Preparar para redirecionar
- 27 PopUp
- 28 Date
- 29 Current Time Clock
- 30 Form
- 31 Validação de Formulários
- 32 Form
- 33 Void 0
- 34 Comparando JavaScript com seu primo ActionScript (script do Flash)
- 35 Referências sobre JavaScript no site do Mozilla
- 36 Outras boas Referências sobre JavaScript
- 37 Grandes repositórios de JavaScript

JavaScript na Wikipedia

http://pt.wikipedia.org/wiki/Javascript

JavaScript é uma linguagem de programação criada pela Netscape em 1995, que a princípio se chamava LiveScript, para atender, principalmente, as seguintes necessidades:

- Validação de formulários no lado cliente (programa navegador); - Interação com a página. Assim, foi feita como uma linguagem de script. Javascript tem sintaxe semelhante a do Java, mas é totalmente diferente no conceito e no uso.

1. Oferece tipagem dinâmica - tipos de variáveis não são definidos; 2. É interpretada, ao invés de compilada; 3. Possui ótimas ferramentas padrão para listagens (como as linguagens de script, de modo geral); 4. Oferece bom suporte a expressões regulares (característica também comum a linguagens de script).

Sua união com o CSS é conhecida como DHTML. Usando o Javascript, é possível modificar dinamicamente os estilos dos elementos da página em HTML.

Dada sua enorme versatilidade e utilidade ao lidar com ambientes em árvore (como um documento HTML), foi criado a partir desta linguagem um padrão ECMA, o ECMA-262, também conhecido como ECMAScript. Este padrão é seguido, por exemplo, pela linguagem ActionScript da Adobe(Antigamente Macromedia, porém a empresa foi vendida a Adobe).

Além de uso em navegadores processando páginas HTML dinâmicas, o JavaScript é hoje usado também na construção do navegador Mozilla, o qual oferece para a criação de sistemas GUI todo um conjunto de ferramentas (em sua versão normal como navegador, sem a necessidade de nenhum software adicional), que incluem (e não apenas) um interpretador de Javascript, um comunicador Javascript <-> C++ e um interpretador de XUL, linguagem criada para definir a interface gráfica de aplicações.

Introdução

O JavaScript é uma ferramenta muito util para a construção de Web Sites, especialmente se utilizada em conjunto com HTML, CSS, PHP, entre outras ferramentas.

Também pode criar diversas ferramentas para sites:

```
* Jogos
* Relógios
* Animações com mouse
* Menus com select
* Mensagens de alerta
* Janelas Popup
* Validação de Formulários HTML
```

Sintaxe

Como escrever código JavaScript?

```
* Usar a tag <script para dizer ao browse que você está usando Javascript.
* Escrever ou fazer download de Javascript
* Testar scripts
```

Primeiro script JavaScript

HTML e JS

```
<html>
<body>
<script type="text/javascript">
<!--
document.write("Olá Mundo do JavaScript!")
//-->
</script>
</body>
</html>
```

Location - onde usar JS?

```
# Dentro da tag head
# Na tag body
# Em um arquivo externo
```

Exemplo no head

```
<html>
<head>
<script type="text/javascript">
<!--
function popup() {
alert("Olá Mundo do JavaScript (function)")
}
//-->
</script>
</head>
<body>
<input type="button" onclick="popup()" value="Executar">
</body>
</html>
```

Externo - Importando um arquivo JS externo

O arquivo deve estar em JS e somente com sintaxe JS e usar extensão .js

Importante: os arquivos a serem incluídos não devem conter as tags

<script> e </script>.

```
<html>
<head>
<script src="meujs.js">
</script>
</head>
<body>
<input type="button" onclick="popup()" value="Clique Me!">
</body>
</html>
```

Operadores

Aritméticos

```
Operator Descrição     Example
+              Addition          2 + 4
-                Subtraction     6 - 2
*               Multiplication  5 * 3
/                Division           15 / 3
%              Modulus          43 % 10
```

Exemplo de Operadores Javascript com Variáveis

```
<body>
<script type="text/javascript">
<!--
var two = 2
var ten = 10
var linebreak = "<br />"

document.write("two plus ten = ")
result = two + ten
document.write(result)
document.write(linebreak)


document.write("ten * ten = ")
result = ten * ten
document.write(result)
document.write(linebreak)

document.write("ten / two = ")
result = ten / two
```

```
document.write(result)
//-->
</script>
</body>
```

Operadores de Comparação

```
Operator    Descrição         Example     Result
==             Equal To         $x == $y     false
!=             Not Equal To     $x != $y     true
<             Less Than         $x < $y     true
>             Greater Than     $x > $y     false
<=             Menor ou igual     $x <= $y     true
>=             Maior ou igual     $x >= $y     false
```

Variáveis

Podemos imaginar uma variável como uma sexta de frutas e os dados das variáveis como as frutas da sexta.

Quando usar uma variável pela primeira vez é uma boa prática de programação usar a palavra reservada var antes do nome da variável.

```
<body>
<script type="text/javascript">
<!--
var linebreak = "<br />"
var my_var = "Hello World!"

document.write(my_var)
document.write(linebreak)

my_var = "I am learning javascript!"
document.write(my_var)
document.write(linebreak)

my_var = "Script is Finishing up..."
document.write(my_var)
//-->
</script>
</body>
```

Nomes de variáveis

- evitar palavras reservadas do JavaScript

- escolher nomes descritivos

- nomes de variáveis não podem iniciar com algarismos

- uma boa prática é iniciar com minúsculas

Funções

Funções em JavaScript são similares a funções em outras linguagens como PHP, C, C++, Java.

Uma função é um pedaço de código que fica dormindo até que seja chamado para "funcionar".

Ao invés de digitar todo o código sempre que precisar, podemos simplesmente chamar novamente a função. Isto é conhecido como reutilização de código.

```
<html>
<head>
<script type="text/javascript">
<!--
function popup() {
    alert("Olá Mundo do JS!")
}
//-->
</script>
</head>
<body>
<input type="button" onclick="popup()" value="Executar">
</body>
</html>
```

Eventos

Este é o grande recurso do JavaScript, que possibilita interagir com o usuário no site, como:

- clique do mouse

- página sendo carregada

- mouse movendo sobre algum componente da página

- Seleção de item de Select

- Toque de tecla

- etc

```
<html>
<head>
<script type="text/javascript">
<!--
function popup() {
alert("Hello World")
//-->
</script>
</head>
<body>
<input type="button" value="Click Me!" onclick="popup()"><br />
<a href="#" onmouseover="" onMouseout="popup()"">
Hover Me!</a>
```

```
</body>
</html>
```

Todo código em JavaScript é formado por instruções (statments).

Uma instrução é uma variável igual a um valor.

Como também é algo como uma chamada de função, por exemplo document.write(). As instruções definem o que o script deve fazer e como fazer.

Em algumas linguagens como o PHP, o final das instruções obrigatoriamente deve terminar com ponto e vírgula. Mas em JavaScript o ; ao final das instruções é opcional. Obrigatoriamente as instruções terminam com uma quebra de linha.

Tipos de Instruções (Estruturas de Controle)

- Condicionais

- Loops

- Manipulação de Objetos

- Comentários

- Manupulação de exceções

if

```
<script type="text/javascript">
<!--
var myNum = 7;

if(myNum == 7){
    document.write("Lucky 7!");
}
//-->
</script>
```

else

```
<script type="text/javascript">
<!--
var myNum = 10;
```

```
if(myNum == 7){
    document.write("Lucky 7!");
}else{
    document.write("You're not very lucky today...");
}
//-->
</script>
```

else if

```
<script type="text/javascript">
<!--
var visitor = "principal";

if(visitor == "teacher"){
    document.write("My dog ate my homework...");
}else if(visitor == "principal"){
    document.write("What stink bombs?");
} else {
    document.write("How do you do?");
}
//-->
</script>
```

while

```
<script type="text/javascript">
<!--
var myCounter = 0;
var linebreak = "<br />";
document.write("While loop is beginning");
document.write(linebreak);

while(myCounter < 10){
    document.write("myCounter = " + myCounter);
    document.write(linebreak);
    myCounter++;
}

document.write("While loop is finished!");
</script>
```

for

```
<script type="text/javascript">
<!--
var linebreak = "<br />";
document.write("For loop code is beginning");
document.write(linebreak);

for(i = 0; i < 5; i++){
    document.write("Counter i = " + i);
    document.write(linebreak);
}

document.write("For loop code is finished!");
</script>
```

Comentários

```
<script type="text/javascript">
<!--
// This is a single line Javascript comment

document.write("I have comments in my Javascript code!");
//document.write("You can't see this!");
//-->
</script>
```

Array

São tipos especiais de variáveis.

Valores são armazenados em arrays passando o nome do array e a localização no array entre colchetes.

myArray[2] = "Hello World";

Valores em arrays são acessados com o nome do array e a localização do valor:

myArray[2]

JavaScript traz algumas funções para trabalhar com arrays.

```
<script type="text/javascript">
<!--
var myArray = new Array();

myArray[0] = "Baseball";
```

```
myArray[1] = "Cricket";
myArray[2] = "Football";

document.write(myArray[0] + myArray[1] + myArray[2]);
//-->
</script>
```

Ordenação de arrays

```
<script type="text/javascript">
<!--
var myArray = new Array();

myArray[0] = "Baseball";
myArray[1] = "Cricket";
myArray[2] = "Football";

myArray.sort();

document.write(myArray[0] + myArray[1] + myArray[2]);
//-->
</script>
```

Alert

Alert emite uma caixa de mensagem (janela popup) com mensagem texto.

```
<form>
<input type="button" onclick= "alert('Realmente excluir o registro atual?')"
value="Corfirmar Exclusão">
</form>
```

Confirm

Solicita confirmação com uma mensagem e exibindo dois botões para confirmar ou desistir.

```
<html>
<head>
<script type="text/javascript">
<!--
function confirmation() {
    var answer = confirm("Deixar o curso_php.org?")
    if (answer){
        alert("Até outro dia!")
        window.location = "http://www.google.com/";
    }
    else{
        alert("Grato por permanecer!")
```

```
    }
}
//-->
</script>
</head>
<body>
<form>
<input type="button" onclick="confirmation()" value="Deixar o curso_php.org">
</form>
</body>
</html>
```

Prompt

Abra uma janela com uma mensagem e uma caixa de texto para receber resposta do usuário.

```
<head>
<script type="text/javascript">
<!--
function prompter() {
var reply = prompt("Olá amigo visitante! Qual seu nome?", "")
alert ('Seu nome é ' + reply)
}
//-->
</script>
</head>
<body>
<input type="button" onclick="prompter()" value="Dizer meu nome!">
</body>
```

Print

Abre o diálogo de impressão do browser para imprimir a janela atual

```
<form>
<input type="button" value="Print This Page" onClick="window.print()" />
</form>
```

Redirect

```
<script type="text/javascript">
<!--
window.location = "http://www.google.com/"
//-->
</script>
```

Time Delay

```
<html>
<head>
<script type="text/javascript">
<!--
function delayer(){
    window.location = "../javascriptredirect.php"
}
//-->
</script>
</head>
<body onLoad="setTimeout('delayer()', 5000)">
```

Preparar para redirecionar

This page is a time delay redirect, please update your bookmarks to our new location!

</body> </html>

PopUp

<head> <script type="text/javascript"> </script> </head> <body> <form> <input type="button" onClick="myPopup()" value="POP!"> </form>

CLICK ME TOO!

</body>

```
   *  dependent - Subwindow closes if parent(the window that opened it) window
closes
   * fullscreen - Display browser in full screen mode
   * height - The height of the new window, in pixels
   * width - The width of the new window, in pixels
   * left - Pixel offset from the left side of the screen
   * top - Pixel offset from the top of the screen
   * resizable - Allow the user to resize the window or prevent resizing
   * status - Display the status bar or not
```

<head> <script type="text/javascript"> </script> </head> <body> <form> <input type="button" onClick="myPopup2()" value="POP2!"> </form>

CLICK ME TOO!

</body>

Date

Isto é agora

```
<script type="text/javascript">
<!--
var currentTime = new Date()
//-->
</script>

   * getTime() - Number of milliseconds since 1/1/1970 @ 12:00 AM
   * getSeconds() - Number of seconds (0-59)
   * getMinutes() - Number of minutes (0-59)
   * getHours() - Number of hours (0-23)
   * getDay() - Day of the week(0-6). 0 = Sunday, ... , 6 = Saturday
   * getDate() - Day of the month (0-31)
   * getMonth() - Number of month (0-11)
   * getFullYear() - The four digit year (1970-9999)
```

Agora podemos imprimir as informações. Usaremos os métodos getDate, getMonth, e getFullYear neste exemplo.

Agora

```
<script type="text/javascript">
<!--
var currentTime = new Date()
var month = currentTime.getMonth() + 1
var day = currentTime.getDate()
var year = currentTime.getFullYear()
document.write(month + "/" + day + "/" + year)
//-->
</script>
```

Current Time Clock

Agora

```
<script type="text/javascript">
<!--
var currentTime = new Date()
var hours = currentTime.getHours()
var minutes = currentTime.getMinutes()
if (minutes < 10)
minutes = "0" + minutes
document.write(hours + ":" + minutes + " ")
if(hours > 11){
document.write("PM")
} else {
document.write("AM")
}
```

```
//-->
</script>
```

Form

Checando por não vazio

function notEmpty(elem){ var str = elem.value; if(str.length == 0){

```
   alert("You must fill in all required fields (*)");
   return false;} else {
   return true;}
```

}

Validação de Formulários

function formValidation(form){

```
   if(notEmpty(form.req1)){
       if(notEmpty(form.req2)){
           return true;
       }
   } else {
       return false;
   }
```

}

Form

<head> <script type="text/javascript"> </script> </head> <body> <form name="example" onSubmit="return formValidation(this)">

- Required 1:<input type="text" name="req1" />
- Required 2:<input type="text" name="req2" />

<input type="submit" />

</form> </body>

Void 0

News Flash

I am a useless link

 Set myNum Please

Comparando JavaScript com seu primo ActionScript (script do Flash)

http://www.webreference.com/programming/javascript/j_s/column2/

Referências sobre JavaScript no site do Mozilla

http://developer.mozilla.org/en/docs/JavaScript

Outras boas Referências sobre JavaScript

http://www.javascriptkit.com/jsref/

http://javascript-reference.info/

http://www.w3schools.com/jsref/default.asp

http://docs.sun.com/source/816-6408-10/

http://www.devguru.com/technologies/JavaScript/home.asp

http://wp.netscape.com/eng/mozilla/3.0/handbook/javascript/

http://devedge-temp.mozilla.org/library/manuals/2000/javascript/1.5/reference/

http://www.comptechdoc.org/independent/web/cgi/javamanual/

http://www.quirksmode.org/js/contents.html

http://argento.bu.edu/~ccruz/javascript/reference.html

http://www.java2s.com/Code/JavaScriptReference/CatalogJavaScriptReference.htm

Grandes repositórios de JavaScript

http://www.javafile.com/

http://www.java-scripts.net/

http://dhtml-menu.com/

http://www.tizag.com/javascriptT/ (site com o original deste tutorial)

http://javascript.internet.com/

Referências/CSS

De Wikibooks

Conteúdo

- 1 CSS na Wikipedia
- 2 Agrupar Seletores
- 3 Cor do Texto
- 4 Cor de Fundo
- 5 Imagem de Fundo
- 6 Exemplo
- 7 Tipos de Fontes
- 8 Propriedades dos Estilos de Texto
- 9 Tipos de Folhas de Estilo
    - 9.1 - Integrada ao HTML
    - 9.2 - Embutida na página
    - 9.3 - Externo
- 10 Usando duas Folhas de Estilo
- 11 Folhas de Estilo Persistentes
- 12 Folhas de Estilo Default
- 13 Folhas de Estilo Alternativas
- 14 Compatibilidade entre Navegadores
- 15 Posicionando com Estilo
    - 15.1 Propiedades de Texto
    - 15.2 Prppriedades de Fontes
    - 15.3 Propriedades de Quadro
    - 15.4 Propriedades de Cor e Fundo
    - 15.5 Propriedades de Classificação
- 16 Ferramentas para auxílio no CSS
- 17 Referências

CSS na Wikipedia

http://pt.wikipedia.org/wiki/Cascading_Style_Sheets

Cascading Style Sheets, ou simplesmente CSS, é uma linguagem de estilo utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como HTML ou XML. Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento.

Ao invés de colocar a formatação dentro do documento, o desenvolvedor cria um link (ligação) para uma página que contém os estilos, procedendo de forma idêntica para todas as páginas de um portal. Quando quiser alterar a aparência do portal basta portanto modificar apenas um arquivo.

Com a variação de atualizações dos navegadores (browsers) como Internet Explorer que ficou sem nova versão de 2001 a 2006, o suporte ao CSS pode variar. O Internet Explorer 6, por exemplo, tem suporte total a CSS1 e praticamente nulo a CSS2. Navegadores mais modernos como Opera, Internet Explorer 7 e Mozilla Firefox tem suporte maior, inclusive até a CSS 3, ainda em desenvolvimento.

A interpretação dos browsers pode ser avaliada com o teste Acid2, que se tornou uma forma base de revelar quão eficiente é o suporte de CSS, fazendo com que a nova versão em desenvolvimento do Firefox seja totalmente compatível a ele assim como o Opera já é.

Agrupar Seletores

Podemos agrupar vários seletores da seguinte forma:

```
h1,h2,h3,h4,h5,h6
{
color: green;
}
```

Cor do Texto

```
cortexto{color:green;}
```

Cor de Fundo

```
corfundo{background-color:blue;}
```

Imagem de Fundo

```
imagem{ background-image:url(fondo.jpg);}
```

Exemplo

```
<HTML><HEAD><TITLE>Exemplo</TITLE>
<STYLE type="text/css">
<!--
P{background-color:black;
 color:purple; width:400px;
 border-color:blue; border-width:12px;}
.tuestilo{background-image:url(greenmar.gif); color:yellow;
border-style:inset; border-width:5pt; float:left; width:60px;}
H1{background-color:#afaefd; width:40px; clear:both;
padding-right:10px; padding-top:12px; color:navy;}
H3{background-color:blue; margin-left:14px; margin-top:35px;}
-->
</STYLE></HEAD>
<BODY>
<P>Este parágrafo terá texto violeta com em fundo preto e largura de 400 pixels,
assim como uma borda azul com uma altura de 12 pixels. </P>
<DIV class="tuestilo">Este parágrafo usará a clase de estilo "tuestilo", usará
uma imagen com fundo assim como uma borda estilo "inset"
com uma altura de 5 pontos, permitirá que haja elementos flutuando à sua direita
e terá uma largura de 60 pixels.</DIV>
<H1>Entre outras características este bloco tem uma margen interna superior de 12
pixels e uma margen interna direita de 10 pixels.</H1>
<H3>Este bloco usará uma margen esquerda de 14 pixels e uma superior de 35
pixels</H3>
</DIV>
</BODY></HTML>
```

Tipos de Fontes

```
<HTML><HEAD><TITLE>UN SIMPLE EJEMPLO </TITLE>
<STYLE TYPE="text/css">
<!--
.meu_estilo{font-size:14pt; font-family:Comic Sans MS; font-weight:Bold; font-
style:italic;}
H3 {font-size:25px; font-family:Symbol; font-weight:bold;}
-->
</STYLE></HEAD>
<BODY>
<DIV class="miestilo">Estas linhas estão usando o estilo chamado
"meu_estilo".</DIV>
<H3>A TAG H3 foi redefinida</H3>
</BODY></HTML>
```

Propriedades dos Estilos de Texto

```
<HTML><HEAD><TITLE> MAIS EXEMPLOS </TITLE>
<STYLE type="text/css">
.subrayado{text-decoration:underline; color:red;}
.tachado{text-decoration:line-through; color:blue;}
.testado{text-decoration:overline; color:purple;}
.parpadea{text-decoration:blink; color:green;}
.sangria{text-indent:100px; color:navy;}
.centrado{text-align:center; color:olive;}
.capitales{text-transform:capitalize; color:silver;}
.mayusculas{text-transform:uppercase; color:#aace34;}
.demoaltura{line-height:150%; color:#eeee44;}
</STYLE>
</HEAD>
<BODY>
<SPAN class="subrayado">SUBRAYADO</SPAN> <SPAN class="tachado">TACHADO</SPAN>
<SPAN class="testado">OVERLINE</SPAN> <SPAN class="parpadea">Parpadea</span>
<BR><BR><DIV class="sangria">SANGRIA abcdefghijklmnñopqrstuvwxyz,
abcdefghijklmnñopqrstuvwxyz
abcdefghijklmnñopqrstuvwxyz, abcdefghijklmnñopqrstuvwxyz,
abcdefghijklmnñopqrstuvwxyz</DIV>
<br>
<DIV class="centrado">CENTRADO</Div>
<br>
<DIV class="capitales">capitales, abcdef, ghijkl, mnñopq, rstuvw, xyz,
capitales.</diV>
<br>
<DIV class="mayusculas">abcdefghijklmnñopqrstuvwxyz</DIV>
<p class="demoaltura">
AAAAAAAAAA<br>BBBBBBBB<br>CCCCCCCCC<br>DDDDDDDDD
</p>
</BODY></HTML>
```

Tipos de Folhas de Estilo

- Integrado

- Embutido

- Externo

- Integrada ao HTML

Usado nas próprias TAGs HTML, em seu atributo STYLE, como:

```
<P STYLE="color: #00ff00;">
```

Para evitar problemas de compatibilidade usar entre as TAGs <head> e </head>:

<META HTTP-EQUIV="Content-Style-Type" CONTENT="text/css">

- Embutida na página

Se usa no cabeçalho do HTML com as tags <STYLE> ... </STYLE>

```
<STYLE TYPE="text/css">
<!--
IMG { vertical-align: 50%;}
P {color: #00ff00;}
A {font-family: sans-serif;}
A:link {color: #0000ee;}
A:visited {color: #551a8b;}
A:active {color: #ff0000;}
-->
</STYLE>
```

As linhas:

```
<!--
e
-->
```

São utilizadas para evitar problemas em navegadores sem suporte a CSS, que no caso considerarão como comentário.

Cria-se trechos de código como o acima na TAG head e utiliza-se os elementos definidos nas TAGs HTML da página, assim:

```
<P>Este parágrafo terá a cor do texto
Azul, pois o elemento P foi definido no trecho acima
como azul </P>
```

- Externo

A opção de uso do CSS recomendada é a de criação de um arquivo externo inteiramente em CSS e se importando esse na página, assim:

```
<STYLE TYPE="text/css">
<!--
@import url(http://pagina.net)
-->
</STYLE>
```

Para compatibilidade com navegadores mais antigos usa-se a sinta-se:

<LINK REL="stylesheet" HREF="estilo.css" TYPE="text/css">

Usando duas Folhas de Estilo

(Exibir e Imprimir)

```
<LINK REL="stylesheet" MEDIA="screen" HREF="tela.css" TYPE="text/css">
<LINK REL="stylesheet" MEDIA="print" HREF="impressora.css" TYPE="text/css">
```

Quando o documento é exibido na tela se utilizará o "tela.css", ignorando la segunda folha. E ao imprimir a página se usará a folha "impressora.css", ignorando a anterior.

Folhas de Estilo Persistentes

(Que os navegadores não podem desabilitar)

Aplica-se sempre ao documento sem possibilidade de se desabilitá-la.

Folhas de Estilo Default

Semelhantes às anteriores mas contendo o atributo NAME. Na carga do navegador elas são lançadas na memória e utilizadas na página mas o navegador pode decidir anular seu efeito.

Folhas de Estilo Alternativas

Estas não se aplicam ao documento a menos que o navegador as selecione.

```
<LINK REL="stylesheet" HREF="folhapersistente.css" TYPE="text/css">
<LINK REL="stylesheet" NAME="Folha Default" HREF="default.css" TYPE="text/css">
<LINK REL="alternate stylesheet" NAME="Folha Alternativa" HREF="alternativa.css"
TYPE="text/css">
```

Compatibilidade entre Navegadores

Vale lembrar que tanto o CSS quanto o JavaScript e outros recursos utilizados na construção de sites não têm seu uso inteiramente padronizado pelos navegadores. Alguns navegadores não seguem ao pé da letra os padrões o que acarreta alguns sites não serem exibidos de forma adequada por todos os navegadores mas apenas por aquele para o qual o programador previu em sua programação. Cabe a nós desenvolvedores evitar o uso de práticas e recursos que não sejam compatíveis com os principais navegadores, para evitar que alguns visitantes tenham problemas ao acessar o site.

Posicionando com Estilo

```
<html><head><title>Posicionando com CSS</title></head>
<body>

<div id="listagem"
style="position:absolute;left:1%;top:1%;width:20%;height:95%;border: solid
brown">
<P align="center"><?php include("includes/index.php"); ?></P>
</div>

<div id="cabecalho"
style="position:absolute;left:22%;top:1%;width:60%;height:10%;border: solid
brown">
<P align="center"><?php include("includes/cabecalho.php"); ?></P>
</div>

<div id="fck"
style="position:absolute;left:22%;top:13%;width:60%;height:75%;border: solid
brown">
<P align="center"><?php include("includes/fck.php"); ?></P>
</div>

<div id="tags"
style="position:absolute;right:1%;top:1%;width:15.5%;height:95%;border: solid
brown">
<P align="center"><?php include("includes/tags.php"); ?></P>
</div>

<div id="rodape"
style="position:absolute;left:22%;top:90.3%;width:60%;height:5.6%;border: solid
brown">
<P align="center"><?php include("includes/rodape.php"); ?></P>
</div>
</body>
</html>
```

Algumas Propriedades

Propiedades de Texto

```
word-spacing
letter-spacing
text-decoration
vertical-align
text-transform
text-align
text-indent
line-height
```

Prppriedades de Fontes

```
font-family
font-style
font-variant
font-weight
font-size
font
```

Propriedades de Quadro

```
margin-top
margin-botton
margin-right
margin-left
margin
padding-top
padding-right
padding-botton
padding-left
padding
border-top-width
border-right-width
border-botton-width
border-left-width
border-width
border-color
border-style
border-top
border-right
border-botton
border-left
border
width
height
float
clear
```

Propriedades de Cor e Fundo

```
color
background-color
background-image
background-repeat
background-attachment
background-position
background
```

Propriedades de Classificação

```
display
white-space
list-style-type
list-style-image
list-style-position
list-style
```

Ferramentas para auxílio no CSS

Existem diversas boas ferramentas para trabalhar ajudando no CSS de sites.

Dreamweaver

É uma ferramenta comercial somente for Windows muito utilizada, especialmente pelos designers.

Aptana

http://www.aptana.com, que inclusive oferece um plugin para o Eclipse. Ele mostra a compatibilidade de cada tag com os principais navegadores, além de contar com ótimo recurso de auto-completar o HTML, JavaScript e CSS. É uma ferramenta free for Windows, Linux e Mac.

BlueFish e Quanta

São muito boas ferramentas para Linux.

Referências

- Free Menu Designs (http://e-lusion.com/design/menu/)

- Gerando estilo CSS para botões - http://

- Dicas de CSS - http://www.phpriot.com/d/articles/client-side/css-tips/index.html

- 53 Técnicas de CSS que você não pode viver sem - http://www.smashingmagazine.com/2007/01/19/53-css-techniques-you-couldnt-live-without/

- Tutorial básico de hojas de estilo en cascada - http://www.geocities.com/lunamatic/tutcss/index.html

- Referências CSS 2.1 - http://www.w3.org/TR/2004/CR-CSS21-20040225

- Recomendações CSS-1 - http://www.w3.org/TR/REC-CSS1

- Recomendações CSS-2 - http://www.w3.org/TR/WD-css2

- CSS Wizard - http://ribafs.net/down/tut/html/wizards/csswiz.html

- CSS para Webdesign - http://www.maujor.com/

- glish.com: CSS Layout Techniques - http://www.glish.com/css/

- tanfa - CSS design, style & fun - http://www.tanfa.co.uk/

- CSS Zen Garden - http://www.csszengarden.com/

Referências/PHP

De Wikibooks

5.6 - PHP

1. Primeira Parte
2. Segunda Parte

PHP na Wikipedia

http://pt.wikipedia.org/wiki/Php

PHP (um acrónimo recursivo para "PHP: Hypertext Preprocessor") é uma linguagem de programação de computadores interpretada, livre e muito utilizada para gerar conteúdo dinâmico na Web. Apesar de ser uma linguagem de fácil aprendizagem e de utilização para pequenos scripts dinâmicos simples, o PHP é uma linguagem poderosa orientada a objetos.

Referências/PHP/PHP1

Este texto teve como principal fonte de informações o manual oficial do PHP em http://www.php.net/manual/pt_BR/

Conteúdo

- 1 - INTRODUÇÃO
- 1.1 Popularidade do PHP
- 1.2 Estatísticas sobre o PHP
- 1.3 História do PHP
- 1.4 Delimitadores do PHP
- 1.5 Grandes Forças do PHP
- 1.6 O Que é Possível Realizar em PHP?
- 2 - TUTORIAL SIMPLES
- 2.1 Informações sobre o PHP
- 2.2 Checar navegador em uso
- 2.3 Teste com echo e print
- 3 - CONFIGURAÇÕES
- 3.1 php.ini
- 3.2 httpd.conf
- 4 - Referência da Linguagem
- 4.1 Separador de Instruções
- 4.2 Comentários
- 5 - Tipos de Dados
- 5.1 Tipo de Variável
- 5.2 Funções que Retornal o Tipo
- 5.3 Casting
- 5.4 Boleanos
- 5.5 Valores que são considerados FALSE
- 5.6 Os demais são TRUE
- 5.7 Inteiros
- 5.8 Alerta
- 5.9 Ponto Flutuante
- 5.10 Strings
- 5.11 Especificando Strings
- 6 - VARIÁVEIS
- 6.1 Variáveis Predefinidas
- 6.2 Escopo de Variáveis
- 6.3 Escopo de variáveis em funções
- 6.4 Usando $GLOBALS no lugar de global
- 6.5 Utilizando Variáveis Estáticas
- 6.6 Variáveis Variáveis
- 6.7 Determiando o Tipo das Variáveis
- 7 - Constantes
- 7.1 Constantes Mágicas
- 7.2 Algumas constantes "mágicas" do PHP
- 8 - EXPRESSÕES
- 9 - Operadores
- 9.1 Tipos de Operadores
- 9.2 Precedência de Operadores
- 9.3 Exemplo
- 9.4 Para facilitar a percepção das precedências usa-se parêntesis
- 9.5 Operadores Aritméticos
- 9.6 Exemplos de operadores aritméticos
- 9.7 Operadores de Atribuição
- 9.8 Operadores de Controle de Erro
- 9.9 Operadores de Execução
- 9.10 Operadores de Incremento e de Decremento
- 9.11 Operadores Lógicos
- 9.12 Operadores de String
- 9.13 Convertendo strings em números
- 9.14 Operações com Strings
- 9.15 Operadores com Arrays

- INTRODUÇÃO

O Que é PHP?

É uma linguagem de script, Open Source, de uso geral, no lado do servidor, embutível no HTML e especialmente voltada para o desenvolvimento Web. Originalmente chamou-se PHP (Personal Home Page) mas depois teve seu nome alterado pela comunidade para PHP: Hypertext Processor, um acrônimo recursivo.

Um pequneno script em PHP

```
<?php
echo "Script em PHP!";
?>
```

Popularidade do PHP

O PHP atualmente deve ser a linguagem web mais usada no planeta.

Segundo estatísticas do site:

http://www.drews.cx/2006/03/23/php-usage-stats-go-up-again/

Onde cita as 4 mais populares linguagens: C, C++, Java e PHP. Nesta ordem.

Observação: todas oriundas do C.

Na prática, entre as linguagens Web o PHP fica em segundo lugar. Acontece que Java é não somente uma linguagem mas uma plataforma, contando com recursos para desktop, microedition, etc. Já PHP tem o uso restrito praticamente à Web.

Veja a lista geral:

http://www.tiobe.com/index.htm?tiobe_index

Em março/2006 ele já está presente em mais de 20 milhões de domínios ao redor do planeta (http://www.php.net/usage.php).

Estatísticas sobre o PHP

http://www.nexen.net/chiffres_cles/phpversion/php_statistics_for_april_2006.php

História do PHP

- Criado em 1995 por Rasmus Lerdorf, na forma de scripts Perl para coletar estatísticas online de seu currículo. Com um interpretador em C e comunicação com SGBDs.

- Versão 2 aparece em novembro de 1997, quando recebe seu nome inicial e é enriquecido com um interpretador de formulários (FI) - PHP/FI. Teve seu código fonte disponibilizado para a comunidade. Contava com 50.000 domínios que o utilizavam (em torno de 1% dos existentes na época).

- Versão 3 sai em seguida, logo em julho de 1998. Similar ao PHP atual, esta versão foi totalmente reescrita por Andi Gutmans e Zeev Suraski, programadores israelenses. Inicia o suporte à orientação a objetos e a sua extensibilidade, que atraiu muitos programadores. Rebatizado de PHP: Hypertext Processor. Já estava presente em 10% dos servidores web da Internet.

- Versão 4 sai em maio de 2000. Melhora de performance, suporte a muitos servidores web, a session, entre outros. Já está presente em 20% dos domínios da Internet.

- Versão 5 sai em julho de 2004. Seu foco principal é a orientação a objetos, que corrije deficiências e traz novos e amplos recursos para a orientação a objetos.

Delimitadores do PHP

Para que o interpretador reconheça quando tem que interpretar o script em PHP e quando não, usa-se delimitadores para que quando os encontre ele interprete corretamente. Quando ele encontra o delimitador <?php ele começa a processar como PHP e quando encontra ?> ele pára de processar e tudo o mais envia para o browser como está. Existem outros tipos de delimitadores do PHP mas estes <?php ... ?> são os recomendados, por serem os mais abrangentes e terem suporte inclusive a XML.

Grandes Forças do PHP

- Sua simplicitade é muito importante para o programador que se inicia no PHP.

- Seus recursos atendem ao programador experiente.

- Sua documentação rica em conteúdo e exemplos facilita a solução de problemas, além da busca online.

- O suporte de maneira simples à maioria dos SGBDs do mercado atende às necessidades de sites dinâmicos com acesso a bancos de dados.

O Que é Possível Realizar em PHP?

- Em sendo uma linguagem de uso geral, podemos realizar praticamente qualquer atividade realizada com outra linguagem. Ele roda no lado servidor, como aplicação Web e também roda no lado cliente, com o PHP-GTK.

- Existem edições para os principais sistemas operacionais (Linux, BSDs, Windows, Mac OS X, etc).

- Suportado pela maioria dos servidores Web, especialmente pelo Apache.

- Bom suporte à orientação a objetos, em especial com a chegada da versão 5.

- Comunicação com Java.

- Suporte aos principais SGBDs do mercado (MySQL, PostgreSQL, Oracle, SQL Server, etc), atualmente são mais de vinte os suportados.

- Suporte a LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (em Windows somente), a IRC, compressão (gzip, bz2, rar), PDF, GD, Flash, XML, etc..

- TUTORIAL SIMPLES

Para continuar você deve ter um servidor web instalado e com suporte a PHP e, caso queira usar bancos de dados, instale um SGBD.

Os arquivos em PHP somente funcionam corretamente no navegador, se estivem no diretório reconhecido pelo servidor web, que é o DocumentRoot. Considerando que o PHP e Apache tenham sido instalados com o Xampp e que o diretório web seja o default, este fica em:

C:\Arquivos de Programas\Xampp\htdocs

Além disso qualquer script que contenha algo em PHP deve ter a extensão .php para que seja corretamente processado pelo Apache.

Criar um arquivo chamado olamundo.php na pasta C:\Arquivos de Programas\Xampp\htdocs, com o conteúdo:

<?php echo "Olá PHP!"; ?>

Startar o Apache e abrir no browser assim:

http://127.0.0.1/olamundo.php

Visualizar resultado de script PHP no Browser

Após ter executado o olamundo.php no Browser solicite a visualização do código fonte da página exibida.

Veja que nada chega de PHP. O PHP é processado no servidor e envia apenas o resultado em HTML para o browser.

Alerta sobre Editores e Processadores de Texto com PHP Para criar scripts PHP evite o uso de processadores de texto e de editores como o Bloco de Notas, pois deixam sujeira no código e dificultam o salvamento do arquivo.

Preferentemente use uma IDE especializada em PHP, que inclusive tem outros recursos úteis, a exemplo do PHPEclipse ou um editor de texto como o freeware Win32pad.

Informações sobre o PHP

Para obter diversas informações úteis sobre o PHP instalado no seu servidor, execute um script com a função:

```
<?php
    phpinfo();
?>
```

Checar navegador em uso

```
if (strstr($_SERVER["HTTP_USER_AGENT"], "MSIE")) {
     echo "Seu navegador é o Internet Explorer!";
} else {
     echo "Seu navegador não é o IE!";
}
```

strstr - procura substrings.

Teste com echo e print

```
<?php
// Teste de echo e print
//print "primeiro, ", "segundo"; // Gera erro de parser
echo "primeiro, ", "segundo, ", "terceiro"; //Funciona
// Ou seja, print pode trabalhar apenas com uma string de cada vez e echo com
várias
?>
```

print retorna valor enquanto que echo não retorna.

```
<?php
$print = print "";
echo "Retorno do print: " . $print;
?>
```

Agora teste esse:

```
<?php
    $echo = echo "";
    echo "Retorno do echo: " . $echo;
?>
```

Irá receber um erro fatal (do parser).

- CONFIGURAÇÕES

php.ini

register_globals = off (por questões de segurança)

No Xampp o php.ini traz register_globals ativo por default.

GD, pgsql, mysql e outras extensões que queira usar deverá descomentar no php.ini, seção Extensions.

httpd.conf

Para poder alterar o diretório web default deverá mudar dois parâmetros no arquivo httpd.conf do Apache:

DocumentRoot "/opt/lampp/htdocs"

<Directory "/opt/lampp/htdocs">

Os diretórios acima são para a edição for Linux do Xampp. Para o Windows observe que o Apache utiliza, não barras, que são utilizadas neste SO, mas sim contra-barras e as vezes contra-barras duplas.

No caso queremos mudar o diretório default para /home/www, então alteramos o httpd.conf para que fique assim:

DocumentRoot "/home/www"

<Directory "/home/www">

No Windows fica assim:

DocumentRoot "c:/Arquivos de Programas/Xampp/htdocs"

<Directory "<c:/Arquivos de Programas/Xampp/htdocs">

Que devem ficar assim:

DocumentRoot "c:/1www"

<Directory "<c:/1www">

Configuração das extensões suportadas. Altere a linha:

DirectoryIndex index.html index.html.var index.php index.php3 index.php4

Veja que acima configurou-se para perceber inclusive arquivos com as extensões .php3 e .php4.

Caso não fossem incluídas, arquivos com extensão .php3 e .php4 não poderiam ser abertos neste servidor.

- Referência da Linguagem

Separador de Instruções

O ponto e vírgula ; é o separador de instruções em PHP (como em Perl e em C).

Sempre que uma instrução terminar devemos digitar um ponto e vírgula (echo "ola";).

Comentários

Em PHP podemos usar 3 tipos de comentários (/* ... */, // e #) mas devemos utilizar somente os dois primeiros e o primeiro é o mais eficiente, que é o /* ... */, que veio do C, já que # está em processo de obsolescência. Ou seja, devemos usar // ou /* ... */, de preferência este último.

- Tipos de Dados

O PHP suporta os oito tipos primitivos:

- boolean, integer, float e string (básicos) - array e object (compostos) - resource e NULL (especiais)

float é sinônimo de double em PHP.

Tipo de Variável

O tipo de uma variável em PHP não é controlado pelo programador, depende do valor da variável e é avaliado em tempo de execução. Não é permitido ao programador declarar o tipo de variáveis.

Funções que Retornal o Tipo

gettype

todas as is_type

Casting

(tipo) variavel;

Boleanos

Pode ser TRUE ou FALSE (case-insensitive)

Exemplos

```
<?php
echo gettype((bool) "")."<br>";        // bool(false)
echo gettype((bool) 1)."<br>";        // bool(true)
echo gettype((bool) -2)."<br>";        // bool(true)
echo gettype((bool) "foo")."<br>";    // bool(true)
echo gettype((bool) 2.3e5)."<br>";    // bool(true)
echo gettype((bool) array(12))."<br>"; // bool(true)
echo gettype((bool) array())."<br>";  // bool(false)
?>
```

Convertendo Explicitamente para Boleano

Usa-se o modificador (bool) ou (boolean).

Valores que são considerados FALSE

FALSE, 0, 0.0, "0", "", array vazio, objeto sem membros e NULL (e variáveis indefinidas).

Os demais são TRUE

Inteiros

Em PHP, inteiro é um número do conjunto matemático dos Inteiros (Z), que contem os negativos, o zero e os positivos.

Em PHP os inteiros podem ser decimais, octais ou hexadecimais.

octal - precedido por 0.

hexadecimal - precedido por 0x.

Exemplos

```
<?php
$a =1234;
echo $a."<br>"; // número decimal
$a =-123;
echo $a."<br>"; // um número negativo
$a =0123;
echo $a."<br>"; // número octal (equivalente a 83 em decimal)
$a =0x1A;
echo $a."<br>"; // número hexadecimal (equivalente a 26 em decimal)
?>
```

O tamanho dos inteiros depende da plataforma e é de 32 bits com sinal. O PHP não suporta inteiros sem sinal.

Overflow - caso seja especificado um número inteiro além dos limites, será interpretado como flutuante.

Convertendo Explicitamente para Inteiro

Usar o modificador (int) ou (integer).

Ou com a função intval().

De boleanos - FALSE será retornado como 0 e TRUE como 1.

De flutuantes - ao converter para inteiros serão truncados

De strings - A string será avaliada como um ponto flutuante se contiver qualquer um dos caracteres '.', 'e', ou 'E'. Em outros casos, ela será avaliada como um inteiro.

De outros tipos - não têm precisão, exatidão, portanto é melhor evitar.

Alerta

echo (int) ((0.1 + 0.7) * 10); // Exibirá 7 ao invés do esperado 8

Ponto Flutuante

É o float, double ou real.

Exemplos

1.234 ou 1.2e3 ou 7E-10

```
<?php
$a = 1.234;
echo $a."<br>";
$b = 1.2e3;
echo $b."<br>";
$c = 7E-4;
echo $c;
?>
```

O tamanho de um float depende também da plataforma e é de 64bits no formato IEEE(*).

Nunca compare números em ponto flutuante em igualdades, sob pena de cometer erros.

- - (Wikipedia - http://pt.wikipedia.org)O Instituto de Engenheiros Eletricistas e Eletrônicos ou IEEE (pronuncia-se I-3-E) é uma organização profissional sem fins lucrativos, fundada nos Estados Unidos. É a maior (em número de sócios) organização profissional do mundo. O IEEE foi formado em 1963 pela fusão do Instituto de Engenheiros de Rádio (IRA) com o Instituto Americano de Engenheiros Elétricistas (AIEE). O IEEE tem filiais em muitas partes do mundo, sendo seus sócios engenheiros eletricistas, engenheiros da computação, cientistas da computação, profissionais de telecomunicações etc. Sua meta é promover conhecimento no campo da engenharia elétrica, eletrônica e computação. Um de seus papéis mais importantes é o estabelecimento de padrões para formatos de computadores e dispositivos.

Strings

Em PHP um caractere ocupa um byte. Até a versão 5 o PHP não tem suporte a UNICODE, mas está previsto este suporte para a próxima versão (Fonte: Wikipedia - http://www.wikipedia.org).

Não há limite para o tamanho de uma string em PHP.

Especificando Strings

- apóstrofos (chamados de aspas simples ')

- aspas (chamadas de aspas duplas ")

- heredoc (<<<)

Exemplos

```
<?php
echo 'isto é uma string comum';
echo 'Você pode incluir novas linhas em strings,
dessa maneira que estará
tudo bem';

// Imprime: Arnold disse uma vez: "I\'ll be back"
echo 'Arnold once said: "I\'ll be back"';

// Imprime: Você tem certeza em apagar C:\*.*?
echo 'Você tem certeza em apagar C:\\*.*?';

// Imprime: Você tem certeza em apagar C:\*.*?
echo 'Você tem certeza em apagar C:\*.*?';

// Imprime: Isto não será substituido: \n uma nova linha
echo 'Isto não será substituido: \n uma nova linha';

// Imprime: Variaveis $também não $expandem
echo 'Variaveis $também não $expandem';

echo '<br>------------<br>';

$str = <<<EOD
Exemplo de uma string
distribuída em várias linhas
utilizando a sintaxe heredoc.
EOD;

/* Exemplo mais complexo, com variáveis */
class foo
{
   var $foo;
   var $bar;
```

```
   function foo()
   {
       $this->foo = 'Foo';
       $this->bar = array('Bar1', 'Bar2', 'Bar3');
   }
}

$foo = new foo();
$name = 'Meu nome';

echo <<<EOT
Meu nome é "$name". Eu estou imprimindo $foo->foo.
Agora, eu estou imprimindo {$foo->bar[1]}.
Isto deve imprimir um 'A' maiúsculo: \x41
EOT;

?>
```

- VARIÁVEIS

Em PHP as variáveis são iniciadas por um sinal de dólar $.

Variáveis Predefinidas

São as que já vêm definidas no próprio PHP. A função phpinfo() mostra também estas variáveis.

Com o PHP 4.2 o valor default da diretiva register_globals passou a ser off. Com isso as variáveis passaram a ser acessadas de forma diferente e muitos scripts deixaram de funcionar quanto tiveram o PHP atualizado e outros novos scripts não funcionaram devido esperar uma semelhante a anteerior.

```
On                                                          Off

$DOCUMENT_ROOT                                  $_SERVER['DOCUMENT_ROOT'];
$HOME $_ENV['HOME'];
$GLOBALS
$_SERVER
$_GET
$_POST
$_REQUEST
$_SESSION
```

Obs.: Agora, como o default do PHP é register_globals = Off, faz-se necessário usar $_POST['nomecampo'], para receber o valor de um campo de form em script PHP.

Escopo de Variáveis

O escopo de uma variável é o contexto onde ela foi definida e geralmente o escopo é local.

```
$a = 1;
include ("teste.php");

// $a estará disponível, será vista por teste.php, pois foi definida antes.

include ("teste.php");
$a = 1;

// Aqui, como $a foi definida após a inclusão, não será visto pelo teste.php
```

Escopo de variáveis em funções

A palavra-chave global pode preceder uma variável para tornar seu escopo global, como também $GLOBALS[].

Exemplos:

```
<?php
$a = 1; /* escopo global */

function Teste(){
   echo $a; /* referencia uma variável do escopo local (não definida) */
}

Teste();
?>

<?php
$a = 1; /* escopo global */

function Teste(){
    global $a;
    echo $a; /* referencia a variável do escopo global */
}

Teste();
?>
```

Usando $GLOBALS no lugar de global

```
<?php
$a = 1;
$b = 2;

function Soma(){
   $GLOBALS["b"] = $GLOBALS["a"] + $GLOBALS["b"];
}
```

```
Soma();
echo $b;
?>
```

Utilizando Variáveis Estáticas

Variáveis estáticas guardam o valor de variáveis entre execuções de funções.

Ao fechar o programa, ao fechar ou atualizar o browser, o valor da variável se perde.

Exemplos

```
<?php
function Teste (){
   $a = 0;
   echo $a;
   $a++;
}

for ($x=1;$x<10;$x++){
    Teste();
}
echo "<br><br>";

function Teste2(){
   static $a = 0;
   echo $a;
   $a++;
}

for ($x=1;$x<10;$x++){
    Teste2();
}

echo "<br><br>";

// Função recursiva
function Teste3()
{
   static $count = 0;

   $count++;
   echo $count;
   if ($count < 10) {
       Teste3 ();
   }
   $count--;
}

for ($x=1;$x<5;$x++){
    Teste3();
```

```
    if ($x < 4) echo " - ";

}

echo "<br><br>";
//Declarando variáveis static

function foo(){
   static $int = 0;          // correro
   //static $int = 1+2;        // errado (é uma expressão)
   //static $int = sqrt(121);  // wrong  (é uma expressão também)

   $int++;
   echo $int;
}

foo();

?>
```

Variáveis Variáveis

São variáveis cujos nomes podem ser criados dinamicamente.

Variável comun -> $variavel;

Variável variável -> $$variavelvariavel;

Ela torna o valor de uma variável e o trata como se fosse o nome de uma variável.

Obs.: variáveis variáveis não podem ser utilizadas com os arrays superglobais.

Determiando o Tipo das Variáveis

gettype

is_array, is_float, is_int, is_object, is_string, is_numeric

- Constantes

O valor de uma constante não pode ser alterado durante a execução do script.

Convenciona-se usar o nome de constantes com todas as letras em maiúsculas.

define ("NOME", "valor");

Exemplos

```
<?php
// Nomes de constantes válidos
define("FOO",    "alguma coisa");
define("FOO2",    "alguma outra coisa");
define("FOO_BAR", "mais alguma outra coisa")

// Nomes de constantes inválidas
define("2FOO",    "alguma coisa");

// Isto é válido, mas deve ser evitado:
// O PHP pode vir a fornercer uma constante mágica
// que danificará seu script
define("__FOO__", "alguma coisa");

?>
```

Obs.: Somente dados escalares(bolean, integer, float, e string) podem ser armazanados nos valores de constantes.

A função constant() retorna o valor de uma constante.

A função get_defined_constantes() retorna todas as constantes definidas.

Enquanto que defined() checa se uma constante foi definida.

```
define (NOME, valor);
define ("PESO", 70);
print "O peso vale " . PESO . " KG";

<?php
switch (PHP_OS){
    case "WIN32":
        echo "Sistema Windows";
        break;
    case "Linux":
        echo "Sistema Linux";
        break;
    case "OS/2":
        echo "Sistema OS/2";
        break;
    default:
        echo "Sistema não é Windows, Linux nem OS/2";
```

```
        break;
}
?>

<?php
if (strtoupper(substr(PHP_OS, 0, 3)) === 'WIN') {
    echo 'Este é um servidor usando Windows!';
} else {
    echo 'Este é um servidor que não usa Windows!';
}
?>
```

Constantes Mágicas

Essas são as cinco constantes "mágicas", que mudam dependendo de onde elas são utilizadas. Por exemplo, o valor de __LINE__ depende do número da linha em que é utilizada em seu script. Essas constantes especiais são insensíveis ao caso:

Algumas constantes "mágicas" do PHP

Nome Descrição

__LINE__ A linha atual do script. __FILE__ O caminho completo e nome do arquivo. Se utilizado dentro de um include, o nome do arquivo incluído será retornado. __FUNCTION__ O nome da função (Acrescentado no PHP 4.3.0). A partir do PHP 5 esta constante retorna o nome da função como ela foi declarada (sensível a maiúsculas e minúsculas). No PHP 4 sempre retorna o nome em minúsculas. __CLASS__ O nome da classe (Acrescentado no PHP 4.3.0). A partir do PHP 5 esta constante retorna o nome da função como ela foi declarada (sensível a maiúsculas e minúsculas). No PHP 4 sempre retorna o nome em minúsculas. __METHOD__ O nome do método de classe. (Acrescentado no PHP 5.0.0). O nome do método é retornado como foi declarado (sensível a maiúsculas e minúsculas).
Veja também get_class(), get_object_vars(), file_exists(), e function_exists().

- Expressões

Uma expressão é qualquer coisa que tem um valor, normalmente na forma de constantes ou variáveis.

$a = 5;

Temos acima uma equação formada por duas expressões, a da esquerda composta por uma variável e a da direita composta por uma constante.

Funções são expressões cujo valor é igual ao seu valor de retorno.

O PHP é uma linguagem orientada as expressões.

Atribuições são analisadas da direita para a esquerda.

$b = $a = 5; // $a recebe 5 e então $b recebe 5

Exemplo

```
<?php
function dobro($i){
   return $i*2;
}
$b = $a = 5;echo $b."<br>";        /* atribui o valor cinco às variáveis $a e $b
*/
$c = $a++;echo $c."<br>";          /* pós-incremento, atribui o valor original de
$a
                        (5) para $c */
$e = $d = ++$b;echo $e."<br>";    /* pré-incremento, atribui o valor incrementado
de
                        $b (6) a $d e $e */

/* neste ponto, tanto $d quanto $e são iguais a 6 */

$f = dobro($d++);echo $f."<br>";  /* atribui o dobro do valor de $d antes
                        do incremento, 2*6 = 12 a $f */
$g = dobro(++$e);echo $g."<br>";  /* atribui o dobro do valor de $e depois
                        do incremento, 2*7 = 14 a $g */
$h = $g += 10;echo $h."<br>";      /* primeiro, $g é incrementado de 10 e termina
com o
                        valor 24. o valor da atribuição (24) é
                        então atribuído a $h, e $h termina com o valor
                        24 também. */
?>
```

- Operadores

Um operador é algo que alimentado com um ou mais valores devolve outro valor. Assim as funções e outros construtores que retornam valor são operadores.

Tipos de Operadores

- unários - operam apenas em um valor (!, ++, ...)

- binários - operam em dois valores (+, - , ...)

- ternário - selecionar entre 2 valores, dependendo de um terceiro. Emglobá-los entre parênteses facilita o entendimento.

Precedência de Operadores

A precedência de um operador em relação a outro diz que um operador deve ser executado antes do outro.

Exemplo

1 + 5 * 3

Dependendo da precedência dos operadores +ou * teremos resultados diferentes. Se o operador + tiver precedência superior ao * então resolveremos assim:

Somamos 1 + 5 e somente depois multiplicamos por 3 -> (1+5)*3 = 6*3 = 18

Este resultado está incorreto, pois o operador * tem precedência superior ao + e deveria ser assim:

1+(5*3) = 1+15=16

Para facilitar a percepção das precedências usa-se parêntesis

1 + (5 * 3) = Sempre devemos resolver antes o que estiver entre parêntesis.

1 + 15 = 16 (Neste caso fica mais claro).

Veja a tabela contendo a precedência de dos operadores do PHP, da maior precedência para a menor:

- e / têm precedência sobre + e -.

Obs.: Para uma relação completa das precedências veja o manual oficial no site do PHP.

Operadores Aritméticos

São os operadores correspondentes às quatro operações comuns da matemática adicionados ao operador módulo:

• soma (+) • subtração (-) • multiplicação (*) • divisão (/) • módulo (%)

O módulo é o resto de um número dividido por outro.

Exemplos de operadores aritméticos

```
$a = 9;
$b = 4;
echo "\$a + \$b = ".($a + $b); // 13
echo "\$a + \$b = ".$a + $b; // Retorna 4, pois após o ponto é considerado
strings
echo "\$a - \$b = ".($a - $b); // 5
echo "\$a * \$b = ".$a * $b; // 36
echo "\$a / \$b = ".$a / $b; // 2.25 - Divisão Quociente de $a por $b.
echo "\$a % \$b = ".$a % $b; // 1 - Resto de $a por $b
```

Operadores de Atribuição

O operador básico de atribuição é o sinal de igualdade =.

Exemplo

```
$x = $y + 3; // O que representa: 3 será adicionado a $y e o resultado será
atribuído a $x
$a = 3;
$a += 5; // Que é semelhante a $a = $a + 5;
$b = "Bom ";
$b .= "Dia!"; // Similar a $b = $b . "Dia!";
```

Exemplos

```
<?php
$a = 3;
$a += 5; // $a recebe 5 e soma com seus 3, tipo: $a = $a + 5;
echo "\$a vale " . $a;

$b = "Bom ";
$b .= "Dia!"; // $b fica com "Bom Dia!", como em $b = $b . "Dia!";
echo "\$b vale " . $b;

$c=2;
$a -= $c;    //$a = $a - $c    Subtração
echo "\$a -= \$c vale " . $a;
$a *= $c;    //$a = $a * $c    Multiplicação
echo "\$a *= \$c vale " . $a;
$a /= $c;    //$a = $a / $c    Divisão
echo "\$a /= \$c vale " . $a;
$resto = $a % $c;    //$a = $a % $c    Módulo (resto)
echo "Resto de $a % $c vale: " . $resto;
?>
```

Operadores de Controle de Erro

Representado pelo símbolo de arroba @. Quando precede uma expressão ele abafa as mensagens de erro.

Observação - Somente funciona em expressões, ou seja, em qualquer construção que retorne algo.

Recomendação - Deixar todas as mensagens de erro originais do PHP livres em ambientes de desenvolvimento (de testes). Somente utilizar @ em ambiente de produção, após executar os testes.

Exemplos

```
<?php
/* Erro intencional */
$a = 6;
$b = 0;

echo "Camuflando erro de divisão por zero";
$c = @($a / $b);
?>
```

Operadores de Execução

Existem algumas funções de execução de programas (shell_exec e outras) como também existe um operador, que é formado por dois sinais de crase` `.

Nota: Caso safe_mode esteja desativado no php.ini como também shell_exec() então o operador de execução também fica dasativado.

Exemplo

```
<?php
// Em PHP a crase ` é um operador de execução de arquivos do SO
// Como em scripts bash

// Exibir todos os arquivos do diretório atual, inclusive os ocultos
if (PHP_OS == "WINNT"){

    $output = `dir/o/p`;
    echo "<pre>$output

";

} elseif (PHP_OS == "Linux){

   $output = `ls -la`;

echo "
```

```
$output

";

}else{
   echo "Você está usando um SO diferente de Linux e de Windows!"

} ?>
```

Operadores de Incremento e de Decremento

Os operadores de pré e pós-incremento/decremento são suportados pelo PHP.

```
++$a (pré-incremento) - Primeiro incrementa $a de 1, depois retorna $a
incrementado
$a++ (pós-incremento) - Primeiro retorna $a, depois incrementa $a de 1
--$a (pré-decremento) - Primeiro decrementa $a de 1, depois retorna $a
decrementado
$a-- (pós-decremento) - Primeiro retorna $a, depois decrementa $a de 1
```

Exemplos:

```
<?php
echo "<h3>Pós-incremento</h3>";
$a = 5;
echo "\$a = ".$a."<br><br>";
echo "\$a++ deve ser: " . $a++ . "<br />\n";
echo "\$a deve ser: " . $a . "<br />\n";

echo "<h3>Pré-incremento</h3>";
$a = 5;
echo "++\$a deve ser: " . ++$a . "<br />\n";
echo "\$a deve ser: " . $a . "<br />\n";

echo "<h3>Pós-decremento</h3>";
$a = 5;
echo "\$a-- deve ser: " . $a-- . "<br />\n";
echo "\$a deve ser: " . $a . "<br />\n";

echo "<h3>Pré-decremento</h3>";
$a = 5;
echo "--\$a deve ser: " . --$a . "<br />\n";
echo "\$a deve ser: " . $a . "<br />\n";
?>
```

Operadores Lógicos

Utilizados para comparar duas expressões e o resultado será TRUE ou FALSE.

Exemplos:

```
<?php
$a = true; $b = FALSE; // true e false são insensitivos

echo ($a and $b)? "T<br>":"F<br>";    //E    Verdadeiro (TRUE) se tanto $a quanto
$b são verdadeiros.
echo ($a or $b)? "T<br>":"F<br>";    //OU    Verdadeiro se $a ou $b são
verdadeiros.
echo ($a xor $b)? "T<br>":"F<br>";    //XOR    Verdadeiro se $a ou $b são
verdadeiros, mas não ambos.
echo (! $a)? "T<br>":"F<br>";        //NÃO    Verdadeiro se $a não é verdadeiro.
echo ($a && $b)? "T<br>":"F<br>";    //E    Verdadeiro se tanto $a quanto $b são
verdadeiros.
echo ($a || $b)? "T<br>":"F<br>";    //OU    Verdadeiro se $a ou $b são
verdadeiros.
?>
```

Operadores de String

Strings em PHP são concatenadas com o operador ponto final ".".

Exemplos:

```
<?php
$a = "Olá ";
$b = $a . "mundo do PHP!";
echo $b;

$a = "Olá ";
$a .= "meu mundo!";

echo "<br>" . $a;
?>
```

Convertendo strings em números

```
<?php
$foo = 1 + "10.5";echo $foo."<br>";             // $foo é float (11.5)
$foo = 1 + "-1.3e3";echo $foo."<br>";              // $foo é float (-1299)
$foo = 1 + "bob-1.3e3";echo $foo."<br>";          // $foo é integer (1)
$foo = 1 + "bob3";echo $foo."<br>";                // $foo é integer (1)
$foo = 1 + "10 Small Pigs";echo $foo."<br>";      // $foo é integer (11)
$foo = 4 + "10.2 Little Piggies";echo $foo."<br>"; // $foo é float (14.2)
```

```
$foo = "10.0 pigs " + 1;echo $foo."<br>";          // $foo é float (11)
$foo = "10.0 pigs " + 1.0;echo $foo."<br>";        // $foo é float (11)
?>
```

Operações com Strings

```
<?php
// Pega o primeiro caracter da string
$str = 'Isto é um teste.';
$first = $str{0};
echo $first."<br>";
// Pega o terceiro caracter da string
$third = $str{2};
echo $third."<br>";
// Pega o último caracter da string
$str = 'Isto ainda é um teste.';
$last = $str{strlen($str)-1};
echo $last."<br>";
// Modifica o ultimo caracter da string
$str = 'Olhe o mal';
echo $str{strlen($str)-1} = 'r';
?>
```

Operadores com Arrays

Exemplo Nome Resultado

$a + $b União União de $a e $b.

$a == $b Igualdade TRUE se $a e $b tem os mesmos elementos.

$a === $b Identidade TRUE se $a e $b tem os mesmos elementos na mesma ordem.

$a != $b Desigualdade TRUE se $a não é igual a $b.

$a <> $b Desigualdade TRUE se $a não é igual a $b.

$a !== $b Não identidade TRUE se $a não é identico a $b.

O operador + acrescenta o array da direita no array da esquerda, contudo, chaves duplicadas NÃO são sobrescritas.

Exemplos:

```
<?php
$a = array("a" => "maçã", "b" => "banana");
$b = array("a" =>"pêra", "b" => "framboesa", "c" => "morango");

$c = $a + $b; // Uniao de $a e $b
echo "União de \$a e \$b: \n";
var_dump($c);

$c = $b + $a; // União de $b e $a
echo "União de \$b e \$a: \n";
var_dump($c);
?>
```

Observar que na união de $a+$b o valor de "b" é banana em $a, ele não foi sobrescrito por framboesa de $b.

Assim como framboesa "b" em $b não foi substituído por banana de $a na união de $b+$a.

Referências/PHP/PHP2

De Wikibooks

Conteúdo

- 1 Continuação
- 2 - Estruturas de Controle
  - 2.1 if
  - 2.2 else
  - 2.3 elseif
  - 2.4 while
  - 2.5 do ... while
  - 2.6 for
  - 2.7 foreach
  - 2.8 break
  - 2.9 continue
  - 2.10 switch
- 3 Estruturas em Obsolescência
- 4 Formatando Números
- 5 Validação de Ano Bisexto
- 6 Algumas Funções Matemáticas
- 7 Enviar E-mail via PHP
- 8 Sugestão de Livros
- 9 Links

Continuação

- Estruturas de Controle

As principais estruturas para controlar o fluxo dos scripts PHP.

if, else, elseif, while, do... while, for, break, consinue e switch.

Um script PHP é formado por instruções (cada instrução termina com ;).

Uma instrução pode ser:

```
- uma atribuição
- uma chamada de função
- um laço (loop)
- uma instrução condicional
- até uma instrução nula (;;).
```

As instruções podem ser agrupadas com chaves, formando blocos.

if

Uma construção muito importante e versátil. Permite a execução condicional de fragmentos de código.

Sintaxe:

```
if (expressao) {
    instrucoes;
}
```

expressao - é avaliada como TRUE ou FALSE

```
- Se TRUE as instrucoes serão executadas

- Se FALSE as instrucoes serão ignoradas
```

Exemplos:

```
<?php
$a = 5;
$b = 3;
if ($a > $b){
   echo "a é maior que b";
}
?>
```

else

Executa instrucoes2 caso expressao seja FALSE.

Sintaxe:

```
if (expressao) {
    instrucoes;
} else {
    instrucoes2;
}
```

Exemplos:

```
<?php
$a = 3;
$b = 5;
if ($a > $b){
   echo "a é maior que b";
} else {
   echo "a NÃO é maior que b";
}
?>
```

elseif

É uma combinação de if com else. Caso a expressao do if seja FALSE então o elseif testa a expresao2, se esta for TRUE instrucoes2 serão executadas, caso contrário instrucoes3 serão executadas.

Sintaxe:

```
if (expressao) {
    instrucoes;
} elseif (expressao2) {
    instrucoes2;
} else {
    instrucoes3;
}
```

```
<?php
$a = 3;
$b = 3;
if ($a > $b){
   echo "a é maior que b";
} elseif ($a==$b){
   echo "a é igual a b";
} else {
   echo "a NÃO é maior que b nem igual a b";
}
?>
```

while

Executa instruções várias vezez, enquanto uma expressão for verdadeira (TRUE).

Sintaxe:

```
while (expressao) {
    instrucoes;
}
```

Em cada iteração expressao é avaliada, se TRUE instrucoes serão executadas, se FALSE, serão ignoradas.

Exemplos:

```
<?php

$i = 1;
while ($i <= 10) {
   echo $i++;  /* o valor impresso será
                  $i depois do acréscimo
                  (post-increment) */
}

?>
```

do ... while

Semelhante ao while, diferindo no fato da expressão condicional ficar ao final das iterações.

No do ... while a primeira iteração sempre executará as instruções incondicionalmente.

Sintaxe:

```
do {
    instrucoes;
} while (expressoes);    // Atentar para o ponto e vírgula ao final
```

Exemplos:

```
<?php
// Executa pelo menos uma vez incondicionalmente
$i = 6; $fator = 2; $minimo = 10;
do {
   if ($i < 5) {
       echo "\$i não é grande o suficiente";
       break;
   }
   $i *= $fator;
   if ($i < $minimo) {
       break;
```

```
   }
   echo "\$i está Ok e vale " . $i;

} while (0);

/* Exemplo simples
$i = 0;
do {
   echo $i;
} while ($i > 0);
*/
?>
```

for

Laço composto por 3 instruções. A primeira é a de inicialização da variável, a segunda será avaliada a cada iteração e a terceira é a de incremento. As 3 separadas por ponto e vírgula. Todas as 3 são opcionais e caso a segunda seja nula o loop será infinito.

Sintaxe:

```
for (expr1; expr2; expr3) {
    instrucoes;
}
```

- expr1 será avaliada uma única vez incondicionalmente;

- expr2 será avaliada no início de cada iteração:

```
   - Se TRUE o laço continua e as instruções serão executadas

   - Se FALSE o laço termina.
```

- expr3 é avaliada ao final de cada iteração, no caso incrementa a variável

Exemplos:

```
<?php
/* exemplo 1 - Controla o fluso no for*/
echo "<br><br>1- ";
for ($i = 1; $i <= 10; $i++) {
   echo $i;
}
echo "<br><br>2- ";
/* exemplo 2 - Controle o fluxo no if interno*/
for ($i = 1; ; $i++) {
   if ($i > 10) {
       break;
   }
   echo $i;
}
```

```
echo "<br><br>3- ";
/* exemplo 3 - Controle o fluxo no if interno*/
$i = 1;
for (; ; ) {
   if ($i > 10) {
       break;
   }
   echo $i;
   $i++;
}
echo "<br><br>4- ";
/* exemplo 4 */
for ($i = 1; $i <= 10;$i++);
    echo $i;
?>
```

foreach

Laço para varrer os elementos de uma matriz (array).

Atentar para o fato de que o foreach não requer reset() antes, pois quando é executado o ponteiro interno do array já é zerado automaticamente.

Sintaxe:

```
foreach (expressao_array as $valor){
 instrucoes;
}

foreach (expressao_array as $chave => $valor) {
 instrucoes;
}
```

Exemplos:

```
<?php
//Você pode ter notado que os seguintes itens são funcionalmente idênticos:
$arr = array("um", "dois", "três");
reset ($arr); // Aponta para o primeiro elemento
while (list(, $value) = each ($arr)) {
   echo "Valor: $value<br />";
}
foreach ($arr as $value) {
   echo "Valor: $value<br />";
}

//Os seguintes também são funcionalmente idênticos:
$arr = array("one", "two", "three");
reset($arr);
while (list($key, $value) = each ($arr)) {
   echo "Chave: $key; Valor: $value<br />";
}
foreach ($arr as $key => $value) {
```

```
   echo "Chave: $key; Valor: $value<br />";
}

/* exemplo foreach 1: somente valores */
$a = array(1, 2, 3, 17);
foreach ($a as $v) {
   echo "Valor atual de \$a: $v.<br>";
}

/* exemplo foreach 2: valores (com as chaves impressas para ilustração) */
$a = array(1, 2, 3, 17);
$i = 0; /* para exemplo somente */
foreach ($a as $v) {
   echo "\$a[$i] => $v.<br>";
   $i++;
}

/* exemplo foreach 3: chaves e valores */
$a = array (
   "um" => 1,
   "dois" => 2,
   "três" => 3,
   "dezessete" => 17
);
foreach ($a as $k => $v) {
   echo "\$a[$k] => $v.<br>";
}
?>
```

break

Cancela a execução de laços: for, foreach, while, do ... while ou switch.

break n é suportado, onde n é o número de estruturas a serem canceladas.

Sintaxe:

```
break;

break n;
```

Exemplos:

```
<?php
$arr = array('um', 'dois', 'três', 'quatro', 'PARE', 'cinco');
while (list (, $val) = each ($arr)) {
   if ($val == 'PARE') {
       break;    /* Você poderia colocar 'break 1;' aqui. */
   }
   echo "$val<br />";
}
echo "<br />";
/* Utilizando o argumento opcional. */
```

```
$i = 0;
while (++$i) {
   switch ($i) {
   case 5:
       echo "No 5<br />";
       break 1;  /* Sai somente do switch. */
   case 10:
       echo "No 10; saindo<br />";
       break 2;  /* Sai do switch e while. */
   default:
       break;
   }
}
?>
```

continue

Sai da atual iteração de um loop para continuar na próxima iteração.

Sintaxe:

```
continue;

continue n;
```

Enquanto o break encerra definitivamente um laço, o continue encerra somente a iteração atual.

Exemplos:

```
<?php
$arr = array(1,2,3,4,5,6,7,8);
while (list ($key, $value) = each ($arr)) {
   if (!($key % 2)) { // pula itens pares, ou seja, processa somente ímpares
       continue;
   }
   echo ($value);
}
echo "<br>";
$i = 0;
while ($i++ < 5) {
   echo "Fora<br />";
   while (1) {
       echo "  Meio<br />";
       while (1) {
           echo "  Dentro<br />";
           continue 3;
       }
       echo "Isto nunca será exibido.<br />";
   }
   echo "Nem isso.<br />";
}

// Outro exemplo
  for ($i = 0; $i < 5; ++$i) {
```

```
     if ($i == 2)
         continue
     print "$i\n";
  }
?>
```

switch

Similar a uma série de construções if seguidas.

Sintaxe:

```
switch (variavel) {
    case valor1:
        instrucoes;
        break;
    case valor2:
        instrucoes2;
        break;
    case valorN;
        instrucoesN;
        break;
    default:
        instrucoesDefault;
        break;
}
```

Exemplos:

```
<?php

$i = 1;

// Estrutura com if
if ($i == 0) {
   echo "\$i igual a 0";
} elseif ($i == 1) {
   echo "\$i igual a 1";
} elseif ($i == 2) {
   echo "\$i igual a 2";
}
echo "<br>";
// Estruturas com switch
switch ($i) {
   case 0:
       echo "\$i igual a 0";
       break;
   case 1:
       echo "\$i igual a 1";
       break;
   case 2:
```

```
        echo "\$i igual a 2";
        break;
}
echo "<br>";
$i = 2;
// Executará todos, falta o break
switch ($i) {
   case 0:
        echo "\$i igual a 0";
   case 1:
        echo "\$i igual a 1";
   case 2:
        echo "\$i igual a 2";
}
echo "<br>";
// Simulando intervalos
switch ($i) {
   case 0:
   case 1:
   case 2:
        echo "\$i é menor que 3 mas não negativo";
        break;
   case 3:
        echo "\$i é 3";
}
echo "<br>";
// Valor default
switch ($i) {
   case 0:
        echo "\$i igual a 0";
        break;
   case 1:
        echo "\$i igual a 1";
        break;
   case 2:
        echo "\$i igual a 2";
        break;
   default:
        echo "\$i não é igual a 0, 1 ou 2";
}
?>
```

Estruturas em Obsolescência

Devemos evitar, pois em futuras versões não mais serão utilizados.

Funções com nomes antigos, tipo:

pg_numrown, pg_fecharray e similares.

Caso se faça uma busca por estas funções no site do PHP nada será retornado.

As funções atuais tem nomes com espaço separando palavras, como:

pg_num_rown, pg_fech_array e similares.

Algumas funções agora tem outro nome, por exemplo:

pg_exec agora deve ser substuida por pg_query

Algumas estruturas de controle usavam os dois pontos:

while: ... endwhile, if: ... endif, devemos preferir:

while(){ ... }, if(){ ... }

Evitar o uso do comentário tipo shell #.

Estruturas do tipo 4HTTP_POST_VARS e similares foram substituídas por: $_POST e similares.

Formatando Números

number_format

number_format -- Formata um número com os milhares agrupados

string number_format (float number [, int decimals])

string number_format (float number, int decimals, string dec_point, string thousands_sep)

number_format() retorna uma versão formatada de number. Esta função aceita um, dois ou quatro parâmetros (não três):

Se apenas um parâmetro é dado, number será formatado sem decimais, mas com uma virgula (",") entre cada grupo de milhar.

Se dois parâmetros são dados, number será formatado com o número de casas decimais especificadas em decimals com um ponto (".") na frente, e uma vírgula (",") entre cada grupo de milhar.

Se todos os quatro parâmetros forem dados, number será formatado com o número de casas decimais em decimals, dec_point ao invés do ponto (".") antes das casas decimais e thousands_sep ao invés de uma vírgula (",") entre os grupos de milhares.

Somente o primeiro caractere de thousands_sep é usado. Por exemplo, se você usar foo como o parâmetro thousands_sep no número 1000, number_format() irá retornar 1f000.

Exemplo 1. Exemplo number_format()

Por exemplo, a notação Francesa usa duas casas decimais, vírgula (',') como separador decimal, e espaço (' ') como separador de milhar. Isto é feito com a linha :

```
<?php

// string number_format ( float number, int decimals, string dec_point, string
thousands_sep )
```

```
$number = 1234.56;

'''// Notação Brasileira'''
$numero_format_brasil = number_format($numero, 2, ',', '.');

// 1.234,56
echo "O número' $number' no formato brasileiro fica
'$numero_format_brasil'<br><br>";
?>
```

Validação de Ano Bisexto

```
<?php
function ano_bisexto($ano){
    return (((($ano%4)==0 && ($ano%100) != 0) || ($ano%400)==0);
}

if(ano_bisexto(2006))
    echo "Ano bisexto";
else
    echo "Ano não bisexto";
?>
```

Algumas Funções Matemáticas

```
abs -- Valor absoluto
mixed abs ( mixed número )
<?php
$abs = abs(-4.2); // $abs = 4.2; (double/float)
$abs2 = abs(5);  // $abs2 = 5; (inteiro)
$abs3 = abs(-5);  // $abs3 = 5; (inteiro)
?>

ceil -- Arredonda frações para cima
float ceil ( float valor )

<?php
echo ceil(4.3);    // 5
echo ceil(9.999);  // 10
?>

floor -- Arredonda frações para baixo
float floor ( float valor )
<?php
echo floor(4.3);  // 4
echo floor(9.999); // 9
?>

max -- Localiza o maior número
mixed max ( number arg1, number arg2 [, number ...] )
```

```
mixed max ( array numbers [, array ...] )
<?php
echo max(1, 3, 5, 6, 7);  // 7
echo max(array(2, 4, 5)); // 5

echo max(0, 'hello');    // 0
echo max('hello', 0);    // hello
echo max(-1, 'hello');    // hello

// Com arrays múltiplos, max compara da esquerda para direita,
// assim nesse exemplo: 2 == 2, mas 4 < 5
$val = max(array(2, 4, 8), array(2, 5, 7)); // array(2, 5, 7)

// Se forem informados um array e um não array, o array
// é sempre retornado como se ele fosse o maior
$val = max('string', array(2, 5, 7), 42);  // array(2, 5, 7)
?>

min -- Localiza o menor número
mixed min ( number arg1, number arg2 [, number ...] )
mixed min ( array numbers [, array ...] )

<?php
echo min(2, 3, 1, 6, 7);  // 1
echo min(array(2, 4, 5)); // 2

echo min(0, 'hello');    // 0
echo min('hello', 0);    // hello
echo min('hello', -1);    // -1

// Com arrays múltiplos, min compara da esquerda para direita,
// assim nesse exemplo: 2 == 2, mas 4 < 5
$val = min(array(2, 4, 8), array(2, 5, 1)); // array(2, 4, 8)

// Se ambos forem um array e um não array, o array
// nunca será retornado porque ele é sempre considerado o maior
$val = min('string', array(2, 5, 7), 42);  // string
?>

count -- Conta o número de elementos de uma variável
int count ( mixed var [, int mode] )
<?php
$a[0] = 1;
$a[1] = 3;
$a[2] = 5;
$result = count($a);
// $result == 3

$b[0] = 7;
$b[5] = 9;
$b[10] = 11;
$result = count($b);
// $result == 3;
?>
```

Exemplo 2. Uso recursivo da função count() (PHP >= 4.2.0)

```
<?php
$food = array( 'fruits'  => array('orange', 'banana', 'apple'),
'veggie'  => array('carrot', 'collard','pea'));
// recursive count
echo count($food,COUNT_RECURSIVE);  // mostra 8
// normal count
echo count($food);                  // mostra2 2
?>
```

```
pow -- Potência
number pow ( number base, number exp )

var_dump( pow(2,8) ); // int(256)
echo pow(-1,20); // 1
echo pow(0, 0); // 1
echo pow(-1, 5.5); // erro
```

```
rand -- Gera um número aleatório
int rand ( [int min, int max] )

echo "Exibir 20 números aleatórios entre 20 e 160<br><br>";
for($x=20;$x<40;$x++){
    echo rand($x, 4*$x)."<br>";
}
exit("Saindo...");
```

```
round -- Arredonda um número
float round ( float val [, int precision] )

<?php
echo round(3.4);        // 3
echo round(3.5);        // 4
echo round(3.6);        // 4
echo round(3.6, 0);      // 4
echo round(1.95583, 2);  // 1.96
echo round(1241757, -3); // 1242000
echo round(5.045, 2);    // 5.04
echo round(5.055, 2);    // 5.06
?>
```

```
sqrt -- Raiz quadrada
float sqrt ( float arg )

<?php
// Precisão depende de sua diretiva precision
echo sqrt(9); // 3
echo sqrt(10); // 3.16227766 ...
?>
```

Forçando a Limpeza do Cache

```
<?php
// Este é para os servidores de proxy. Diz para baixar, bypassando o proxy
header ("Cache-Control: no-cache, must-revalidate");
```

```
// Este é para o navegador e nem sempre funciona (falta de padrão entre eles)
header ("Pragma: no-cache");
?>
```

Redirecionamento de Páginas

```
header("location: novapagina.php");
```

Exibir o conteúdo de um arquivo

readfile -- Lê e exibe o conteúdo de um arquivo

readfile('http://www.google.com.br');

readfile('/home/1www/pagina2.php');

Enviar E-mail via PHP

```
<?php
$to = "ribamar.sousa@dnocs.gov.br";
$subject="Apenas testando";
$message = "Estou testando o envio de e-mail pelo PHP.";
$email="ribafs@gmail.com";
$ret=mail($to, $subject, $message, "From: $email\r\nReply-to: $email\r\n");
echo $ret;
?>
```

Usando Favicon em sites

Aquele pequeno ícone que fica à esquerda da URL do site, na caixa Location do Browser, que inclusive personaliza a barra de links quando arrastamos o endereço do site. Para que nosso site apareça para o visitante com um link, devemos seguir os seguintes procedimentos:

- Criar uma imagem com o logo do site, no formato png, jpg, gif, etc com 32x32 pixels

- Salvar como favicon.png ou outra extensão

- Adicionar a TAG abaixo, na TAG head, como abaixo:

```
<head>
...
<link href="http://www.seusite.com.br/diretorio/favicon.png" type="image/gif"
rel="icon">
...
</head>
```

Também pode ser assim:

rel="shortcut icon"

Este ícone pode ser inclusive animado.

Referência: lista da Dicas-L (http://www.dicas-l.com.br).

Sugestão de Livros

```
Desenvolvendo Web Sites com PHP
Editora Novatec
Juliano Niederauer

PHP para quem conhece PHP
Editora Novatec
Juliano Niederauer

PHP Guia do Desenvolvedor
Ed. Berkeley
Sterling Hughes
(Do grupo de desenvolvimento do PHP)

Desvendando aplicações na Web com PHP 4.0
Ed. Ciência Moderna
Tobias Retschiller e
Till Gerken
```

Links

PHP

http://www.php.net

http://www.php.net/manual/pt_BR/

http://www.php.net/downloads.php

http://www.phpbrasil.com/

http://www.planet-source-code.com/vb/default.asp?lngWId=8

http://www.hotscripts.com/PHP/index.html

http://www.zend.com (empresa dos dois membros israelenses da equipe de desenvolvimento do PHP, especial ênfase na seção Developer Zone)

http://www.phpwizard.net

http://www.phpclasses.org

http://www.weberdev.com

http://www.devshed.com

http://www.phpmania.org/

http://www.phpnet.us/ - Hospedagem free para PHP e MySQL (300MB)

http://www.superphp.com.br

http://www.scriptbrasil.com/

http://www.faqts.com/knowledge_base/index.phtml/fid/51/

http://www.alt-php-faq.org/

http://ribafs.byethost2.com/

CURSOS GRÁTIS

http://cursos.cdtc.org.br/brasil/ - Cursos Grátis para Func.Públ.

http://www.apostilando.com/download.php?cod=171&categoria=PHP

http://www.aprendaemcasa.com.br/apcasa6.htm

http://www.solocursosgratis.com/cursos_gratis_php-slctema264.htm

CLIPARTS http://www.digitmania.holowww.com/digital.html – Dígitos (cliparts)

POSTGRESQL

http://www.postgresql.org/docs/current/interactive/

http://pgdocptbr.sourceforge.net/pg80/index.html

http://www.designmagick.com/category/3/PostgreSQL

MYSQL

http://dev.mysql.com/doc/refman/4.1/pt/index.html

http://dev.mysql.com/doc/refman/5.0/en/index.html

SQL

http://www.firstsql.com/tutor.htm

http://sqlzoo.net/

http://www.sql-tutorial.net/SQL-tutorial.asp

http://www.programmingtutorials.com/sql.aspx

http://tutorials.findtutorials.com/

http://www.hardened-php.net/home.8.html - Hardened PHP Project

http://www.1phpscripts.com/

http://www.weberdev.com/

http://www.goldsofts.com/scriptscategory/10/0/1/0.html

Referências/E-books frees

De Wikibooks

5.7 - E-Books Free

PHP 5 Power Programming - Livro muito bom sobre PHP5 - http://www.phptr.com/content/images/013147149X/downloads/013147149X_book.pdf

Muitos E-Books sobre diversas áreas:

- http://www.e-book.com.au/freebooks.htm

- http://www.intelligentedu.com/free_computer_books.html

Projeto Gutemberg com mais de 18.000 e-books frees - http://www.gutenberg.org/index.php

Referências/Cursos na área

De Wikibooks

5.8 - Cursos na Área

Cursos Gratuitos

Cursos Gratuitos para Funcionários Públicos

Cursos de Apache, PHP, Joomla, Xoops, Moodle, MySQL, PostgreSQL, HTML, etc. Todos de ótima qualidade e com direito a certificado - http://cursos.cdtc.org.br/brasil/

Cursos Gratuitos abertos para a Comunidade

A mesma estrutura oferecendo os mesmos cursos abertos à comunidade - http://comunidade.cdtc.org.br/

Curso de HTML, PHP e outros

http://portalfarmaceuticon.com/cursos/index.php

Curso de JavaScript

http://neosite.ilogic.com.br/dicas/2007/04/curso-gratuito-programando-em-javascript.html

Cursos de PHP e outros

http://cursos.ribafs.net

Administração dos SGBDs/MySQL

De Wikibooks

Conteúdo

- 1 Instalação
- 2 Executando
- 3 Trocando a senha do usuário root
- 4 Criando Usuários e Concedendo Privilégios
- 5 Removendo Usuários
- 6 Instalando como Serviço
- 7 Criando Bancos e Tabelas
- 8 Usando o MySQL
- 9 Importação e Exportação de Dados e Extrutura
- 10 Mudar Conjunto de Caracters para LATIN1
- 11 Importar CSV no MySQL
- 12 Funções com Datas
- 13 Ativando o suporte a INNODB no MySQL do XAMPP
- 14 Conversão de Funções do MySQL para o PostgreSQL no PHP
- 15 Replicação no MySQL
- 16 Migrando .DBF para MySQL
- 17 Referências

Instalação

Para testes locais uma instalação prática é com o pacote Xampp

http://xampp.sf.net

Para uso em servidores onde se exige segurança idealmente instalar compilando os fontes para um comtrole maior das características instaladas.

Executando

Uma boa opção de administração do MySQL é o phpmyadmin, que também acompanha o Xampp.

Para administração pela linha de comando use:

mysql -h host -u user -p (o super usuário default é root)

mysql -u root (quando estiver sem senha)

Trocando a senha do usuário root

```
Acessar o servidor do MySQL:
mysql -u root mysql (Usuário root acessar banco mysql)

Alterar senha atual do root para novasenha:
UPDATE user SET Password=PASSWORD("novasenha") WHERE user="root";

Atualizar os procedimentos:
FLUSH PRIVILEGES;

Ou

mysql -u root clientes

SET PASSWORD FOR root=PASSWORD('senhadoroot');
```

Criando Usuários e Concedendo Privilégios

mysql --user=root mysql

GRANT ALL PRIVILEGES ON *.* TO super@localhost

IDENTIFIED BY 'senha' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON *.* TO super@"%"

IDENTIFIED BY 'some_pass' WITH GRANT OPTION;

super - é um total super usuário que pode se conectar no localhost e de qualquer lugar ("%"), mas precisa usar senha

GRANT RELOAD,PROCESS ON *.* TO admin@localhost;

admin - usuário que pode se conectar no localhost sem senha.

Pode executar os comandos mysqladmin reload, mysqladmin refresh, and mysqladmin flush-* e mysqladmin processlist . Não tem nenhum privilégio relacionado aos bancos.

GRANT USAGE ON *.* TO fraco@localhost;

fraco - pode conectar somente via localhost sem senha mas sem privilégios, somente para uso.

Exemplo:

GRANT ALL PRIVILEGES ON *.* TO ribafs@localhost IDENTIFIED BY 'ribafs' WITH GRANT OPTION;

mysql -u ribafs // Dá erro de senha

mysql -u ribafs -p //Funciona após entrar a senha ribafs

Criando Usuários de Olho na Segurança

Este usuário somente conecta o MySQL via localhost.

```
- Abrir phpMyAdmin como super usuário
- Clicar no link Privilégios
- Clicar em Adicionr novo usuário
- Entre com o nome do usuário
- Em servidor selecione Local
- Em senha selecione Sem senha
- Abaixo em Privilégios globais selecione todas as checkbox Dados
- Caso queira marcar mais algum dos privilégios, faça-o e clique em Executar.
```

Pronto, este usuário somente poderá realizar conexões locais e não terá nenhum privilégio a não ser os de cadastrar os dados (nada de excluir, criar ou alterar a estrutura do banco).

Removendo Usuários

DROP USER nomeusuario;

Privilégios

REVOKE GRANT ALL ON nomebancooutabelaou*ou*.* FROM nomeusuario

- - todas as tabelas
- .* todos os bancos e todas as tabelas

banco.* - todas as tabelas do banco

GRANT SELECT,INSERT,UPDATE ON nomebanco.* TO nomeuser;

GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON nomebanco.* TO usuario@localhost

IDENTIFIED BY 'senha';

GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON nomebanco.* TO usuario@dominio.com.br IDENTIFIED BY 'senha';

GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON nomebanco.* TO usuario@'%' IDENTIFIED BY 'senha';

Instalando como Serviço

Instalar MySQL como serviço no Windows para trabalhar com Java (J2EE):

mysqld-nt --install --ansi --sql-mode=ANSI_QUOTES

Instalar como serviço:

bin\mysqld-nt --install mysql

Remover o serviço:

bin\mysqld --remove mysql

Remover serviço ansi:

bin\mysqld --remove --ansi

Removendo Serviços no Windows XP/NT

mysql\bin\mysqld -- remove (remove o serviço mysql) -- remove --ansi (remover o serviço ansi)

Criando Bancos e Tabelas

```
CREATE DATABASE db_name
[[DEFAULT] CHARACTER SET charset_name]
[[DEFAULT] COLLATE collation_name]
ALTER DATABASE db_name
[[DEFAULT] CHARACTER SET charset_name]
[[DEFAULT] COLLATE collation_name]

CREATE TABLE tbl_name (column_list)
[[DEFAULT] CHARACTER SET charset_name] [COLLATE collation_name]]
ALTER TABLE tbl_name
[[DEFAULT] CHARACTER SET charset_name] [COLLATE collation_name]
Example:
CREATE TABLE t1 ( ... ) CHARACTER SET latin1 COLLATE latin1_danish_ci;

col_name {CHAR | VARCHAR | TEXT} (col_length)
[CHARACTER SET charset_name] [COLLATE collation_name]

Exemplo:
CREATE TABLE Table1
(
column1 VARCHAR(5) CHARACTER SET latin1 COLLATE latin1_german1_ci
);
```

Criar Tabelas com Relacionamentos

```
create table produto(
        codigo int not null primary key,
        nome varchar(50) not null unique,
        descricao varchar(200),
        valor real(6, 2)
) ENGINE=INNODB;

create table cliente(
        codigo int not null primary key,
        nome varchar(50) not null,
        email varchar(100) not null unique,
        cpf varchar(11) not null
) ENGINE=INNODB;

create table pedido(
        numero int not null primary key auto_increment,
```

```
        codigocliente int not null references cliente(codigo),
        valortotal real(7,2) DEFAULT '0.00' NOT NULL
) ENGINE=INNODB;

create table item(
        numeropedido int not null references pedido(numero),
        codigoproduto int not null references produto(codigo),
        quantidade int not null,
        primary key(numeropedido, codigoproduto)
) ENGINE=INNODB;

CREATE TABLE product (
        category INT NOT NULL,
        id INT NOT NULL,
        price DECIMAL,
        PRIMARY KEY(category, id)
) ENGINE=INNODB;

CREATE TABLE product_order (
        no INT NOT NULL AUTO_INCREMENT,
        product_category INT NOT NULL,
        product_id INT NOT NULL,
        customer_id INT NOT NULL,
        PRIMARY KEY(no),
        INDEX (product_category, product_id),
        FOREIGN KEY (product_category, product_id)
        REFERENCES product(category, id) ON UPDATE CASCADE ON DELETE RESTRICT,
        INDEX (customer_id),
        FOREIGN KEY (customer_id)
        REFERENCES customer(id)
) ENGINE=INNODB;
```

O tipo InnoDb dá suporte à constraint Foreign Key (references).

Usando o MySQL

1) mysql -u root -p ou mysql -u root

mysql -h host -u user -p banco

Obs: Caso receba a mensagem: Can't connect to MySQL server on 'localhost'

Falta startar o MySQL

2) create database nomebanco;

3) use nomebanco;

4) create table nometabela(campos tipos...);

5) select * from nometabela;

6) show databases;

7) show tables;

8) describe nometabela;

Importação e Exportação de Dados e Extrutura

Exportando:

bin\mysqldump -u user -p passwd banco > banco.sql

Importando:

bin\mysql -u user -p password banco < banco.sql

IMPORTAR Todos os Bancos de um Script

Temos um script contendo diversos bancos então:

mysql -u root < varios_bancos.sql

Mudar Conjunto de Caracters para LATIN1

musql -u root

\C latin1

Importação e Exportação com o phpMyAdmin

Exportar todo um banco

- Abrir o phpMyAdmin e selecionar o banco

- Clicar no botão Exportar

- Basta rolar a tela e clicar em Executar abaixo

- O conteúdo será exibido na tela. Devemos então selecionar e criar um arquivo texto com o mesmo. Por convenção criamos arquivos com a extensão .sql.

- Também podemos exportar para um arquivo compactado. Basta clicar na opção "Compactado com zip ou gzip.

- Existem muitas outras opções para a exportação.

Caso queira exportar somente uma tabela o processo é o mesmo, mudando apenas que devemos

selecionar apenas a tabela desejada.

Populando Tabelas após a criação

O comando LOAD DATA pode ser utilizado para popular tabelas, trazendo de arquivos:

LOAD DATA LOCAL INFILE '/path/arquivo.txt' INTO TABLE nometabela;

SELECT DATABASE();

SHOW CHARACTER SET;

Importar CSV no MySQL

```
$handle = fopen ('./file.csv', 'r');
                while (($data = fgetcsv($handle, 1000, ',', '"')) !== FALSE)
                {
                        $query = "INSERT INTO services VALUES ('". implode("','",
$data)."')";
                        $query = @mysql_query($query);
                }


LOAD DATA INFILE "./ImportData.csv"
     INTO TABLE table1
     FIELDS TERMINATED BY ","
     OPTIONALLY ENCLOSED BY """"
     LINES TERMINATED BY "\r\n";

OPTIONALLY ENCLOSED is optional.
```

Funções com Datas

```
DATE_SUB
SELECT something FROM tbl_name WHERE DATE_SUB(CURDATE(),INTERVAL 30 DAY) <=
date_col;
SELECT DATEDIFF('1997-12-31 23:59:59','1997-12-30');

DATE_ADD
SELECT DATE_ADD('2006-05-00',INTERVAL 1 DAY);

SELECT CURDATE();

SELECT CURTIME();

DATE_FORMAT
SELECT date_format( '2006-04-30', '%d/%m/%Y' ); -- 30/04/2006
SELECT DATE_FORMAT('2003-10-03',GET_FORMAT(DATE,'EUR')); -- 03.10.2003
SELECT DATE_FORMAT('2006-06-00', '%d/%m/%Y');
```

```
SELECT NOW();

SELECT TO_DAYS('1997-10-07'); -- RETORNA DIAS

SELECT YEAR('2000-01-01');
```

Ativando o suporte a INNODB no MySQL do XAMPP

Editar o arquivo my.cnf:

D:_xampplite\mysql\bin\my.cnf

Caso apareça um arquivo "my" sem extensão clique com o botão direito, enviar para e abra num editor de texto e faça as seguintes alterações:

```
...

#skip-innodb
# Uncomment the following if you are using InnoDB tables
innodb_data_home_dir = D:/_xampplite/mysql/data/
innodb_data_file_path = ibdata1:10M:autoextend
innodb_log_group_home_dir = D:/_xampplite/mysql/data/
innodb_log_arch_dir = D:/_xampplite/mysql/data/
# You can set .._buffer_pool_size up to 50 - 80 %
# of RAM but beware of setting memory usage too high
set-variable = innodb_buffer_pool_size=16M
set-variable = innodb_additional_mem_pool_size=2M
# Set .._log_file_size to 25 % of buffer pool size
set-variable = innodb_log_file_size=5M
set-variable = innodb_log_buffer_size=8M
innodb_flush_log_at_trx_commit=1
set-variable = innodb_lock_wait_timeout=50

...
```

Pronto, agora você pode criar tabelas com suporte a INNODB (relacionamentos).

Conversão de Funções do MySQL para o PostgreSQL no PHP

Na conversão de scripts em PHP com MySQL para PHP com PostgreSQL temos que atendar para vários detalhes: nomes das funções (ex.: mysql_connect para pg_connect), sintaxe das funções, que varia em muitas delas (veremos adiante), estrutura do banco: tipos de tabelas, tipos de dados, auto-incremento, etc. Óbvio que para converter entre dois SGBDs devemos conhecer as características de ambos. Exemplo: como vou usar um tipo de dados do MySQL no PostgreSQL, se este não tem o referido tipo? Deverei encontrar no PostgreSQL, um tipo que satisfaça os requisitos daquele do MySQL (para isso precisarei conhecer as características dos tipos de ambos, suas faixas de valores, tipos de dados, etc).

Alguns Exemplos de Conversão de Funções. Não vou me preocupar com tratamento de erros nem com outros detalhes, mostrarei apenas as funções para comparar:

Conexão ao Banco de Dados

No MySQL:

Abrir a conexão:

$con_my = mysql_connect("localhost:porta", "usuario", "senha");

Quando for usar, selecionar o banco:

mysql_select_db('nomebanco', $con_my);

No PostgreSQL:

Etapa única:

$con_pg = pg_connect("host=127.0.0.1 port=5432 dbname=banco user=usuario password=senha");

Obs.: na conexão já se seleciona o banco a ser usado.

Consultas

MySQL:

$q = mysql_query("SELECT * FROM tabela", $con_my);

PostgreSQL:

$q = pg_query($con_pg, "SELECT * FROM tabela");

Obs.: Veja que a ordem dos parâmetros é invertida.

LIMIT

Aqui temos uma boa diferença entre ambos.

Quando for a forma resumida não há diferença entre ambos. Por exemplo:

SELECT * FROM tabela ORDER BY campo LIMIT 5;

Obs.: Altamente recomendado usar ORDER BY antes de LIMIT, para um retorno coerente e o campo do ORDER BY deve ser o campo chave primparia.

Quando o LIMIT trouxer os dois parâmetros então teremos diferença.

Sintaxe no PostgreSQL:

SELECT lista_de_campos FROM expressão [LIMIT { número | ALL }] [OFFSET inicio]

LIMIT ALL – mesmo que omitir LIMIT.

OFFSET inicio – orienta para que a consulta retorne somente a partir de inicio.

OFFSET 0 – mesmo que omitir OFFSET.

LIMIT 50 OFFSET 11 – Deverá trazer 50 registros do 12 até o 50, caso existam.

Exemplos:

SELECT * FROM cliente ORDER BY codigo LIMIT 2 OFFSET 0;

Irá retornar os registros do 1 e 2.

SELECT * FROM cliente ORDER BY codigo LIMIT 2 OFFSET 1;

Irá retornar os registros do 2 e 3.

SELECT * FROM cliente ORDER BY codigo LIMIT 2 OFFSET 2;

Irá retornar os registros do 3 e 4 (Se existirem).

Ou seja, o primeiro parâmetro do é a quantidade e o segundo o inicial (começando do 0).

Sintaxe no MySQL:

LIMIT [início,] linhas

Retorna o número de linhas especificado. Se o valor início for fornecido, aquelas linhas são puladas

antes do dado ser retornado. A primeira linha é 0.

Exemplo:

SELECT * FROM cliente ORDER BY codigo LIMIT 3,2;

O comando acima pede apenas os registros de código 4 e 5, os dois após o 3.

SELECT * FROM cliente ORDER BY codigo LIMIT 3,1;

Aqui retornará 1 registro, de código 4, que é o próximo após o 3.

SELECT * FROM cliente ORDER BY codigo LIMIT 2 , 4;

```
    Aqui retornará os registros de código 3,4,5 e 6.
```

No MySQL o início é o primeiro parâmetro e a quantidade é o segundo.

Experimente os exemplos acima num gerenciador dos SGBDs para consolidar o conhecimento (phpmyadmin e phppgadmin).

Replicação no MySQL

An introduction to replication1

How to start replicating - the slave server

Migrando .DBF para MySQL

No ótimo site Vivaolinux - http://www.vivaolinux.com.br/dicas/verDica.php?codigo=8792

Referências

- Manual Online do MySQL 4.1 em Português - com opção de busca pelo conteúdo.

http://dev.mysql.com/doc/refman/4.1/pt/index.html

- Manual do MySQL em vários idiomas e em vários formatos para download - http://mysql.org/doc/

- Removendo Duplicidades em MySQL - http://www.dicas-l.com.br/print/20060930.html

Administração dos SGBDs/PostgreSQL

Aqui apenas passo o link para o livro PostgreSQL Prático - http://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico

Conteúdo

- 1 PostgreSQL Prático
- 2 Tutorial sobre o phpPgAdmin
- 3 Criando Chave Estrangeira Composta no phpPgAdmin e no PGAdmin
    - 3.1 phpPgAdmin
    - 3.2 PGAdmin
- 4 PlPgSQL Tutorial
    - 4.1 Delimitadores
    - 4.2 Declaração de Variáveis
    - 4.3 Declarado Funções
    - 4.4 Alias para Parâmetros de Funções
    - 4.5 Executando Funções
    - 4.6 Tipos Polimórficos
    - 4.7 Copiando Tipos
    - 4.8 Tipos row
    - 4.9 Tipo registro (record)
    - 4.10 Renomeando Variáveis
    - 4.11 Atribuições
    - 4.12 Execução de Expressão ou Consulta sem Resultado
    - 4.13 Execução de Comandos Dinâmicos
    - 4.14 Estruturas de Controle
    - 4.15 Condicionais
    - 4.16 Laços através do resultado de consultas
    - 4.17 Capturar Erros
    - 4.18 Declaração de Variáveis do tipo Cursor
    - 4.19 Erros e Mensagens
    - 4.20 Procedimentos de Gatilho (Trigger)
    - 4.21 NEW
    - 4.22 OLD
    - 4.23 TG_NAME
    - 4.24 TG_WHEN
    - 4.25 TG_LEVEL
    - 4.26 TG_OP
    - 4.27 TG_RELID
    - 4.28 TG_RELNAME
    - 4.29 TG_NARGS
    - 4.30 TG_ARGV[]
    - 4.31 Gatilho para registrar inserções e atualizações
    - 4.32 Gatilho para auditoria
    - 4.33 Autoditoria ao nível de campos
    - 4.34 Gatilho para manter uma tabela sumário

PostgreSQL Prático

http://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico

Tutorial sobre o phpPgAdmin

- Baixar de - http://phppgadmin.sourceforge.net/

- Copiar para o diretório web e descompactar no diretório "phppgadmin"

- Editar o arquivo conf/config.inc.php e alterar para dois servidores (um local e outro remoto) caso assim queira:

```
        //...
        // Servidor Local
        // Display name for the server on the login screen
        $conf['servers'][0]['desc'] = 'Local';


        // Hostname or IP address for server.  Use '' for UNIX domain socket.
        // use 'localhost' for TCP/IP connection on this computer
        $conf['servers'][0]['host'] = '127.0.0.1';
        $conf['servers'][0]['defaultdb'] = 'nomebancodefault';

        //...
        // Servidor Remoto
        // Example for a second server
        $conf['servers'][1]['desc'] = 'Remoto';
        $conf['servers'][1]['host'] = '10.99.00.11';
        $conf['servers'][1]['port'] = 5432;
        $conf['servers'][1]['defaultdb'] = 'nomebancodefault';

        //...
        // If extra login security is true, then logins via phpPgAdmin with no
        // password or certain usernames (pgsql, postgres, root, administrator)
        // will be denied. Only set this false once you have read the FAQ and
        // understand how to change PostgreSQL's pg_hba.conf to enable
        // passworded local connections.
        $conf['extra_login_security'] = false;

        // Importantes para Importar/Exportar
        $conf['servers'][0]['pg_dump_path'] = '/usr/local/pgsql/bin/pg_dump';
        $conf['servers'][0]['pg_dumpall_path'] =
'/usr/local/pgsql/bin/pg_dumpall';
```

- Pronto, basta abrir http://localhost/phppgadmin

Criando Chave Estrangeira Composta no phpPgAdmin e no PGAdmin

O exemplo é para uma chave com 4 campos, mas o procedimento é válido para qualquer chave composta.

Relacionar composicoes com precos.

Quero que ao digitar um preço em composições o SGBD verifique a integridade do mesmo em precos.

Temos uma tabela cuja chave primária é composta por 4 campos:

tabela, insumo_grupo, insumo, fornecedor e queremos relacionar com outratabela que também contém os mesmos campos com os mesmos tipos de dados.

Como isso pode ter algum engano, seguem os passos.

Esquema das tabelas:

```
CREATE TABLE "composicoes" (
    "servico" character varying(10) NOT NULL,
    "insumo_grupo" character varying(6) NOT NULL,
    "insumo" character varying(10) NOT NULL,
    "quantidade" real NOT NULL DEFAULT 0,
    "coeficiente" real NOT NULL DEFAULT 0,
    "data_inclusao" timestamp without time zone,
    "data_alteracao" timestamp without time zone,
    "data_cancelamento" timestamp without time zone,
    "uid_inclusao" character varying(10),
    "uid_alteracao" character varying(10),
    "uid_cancelamento" character varying(10),
    "tabela" smallint NOT NULL DEFAULT 2006,
    "fornecedor" smallint NOT NULL DEFAULT 1,
    CONSTRAINT "composicoes_fk_insumo" FOREIGN KEY (insumo_grupo, insumo)
REFERENCES insumos(grupo, insumo),
    CONSTRAINT "composicoes_fk_servicos" FOREIGN KEY (servico) REFERENCES
servicos(servico),
    CONSTRAINT "composicoes_pkey" PRIMARY KEY (servico, insumo_grupo, insumo)
) WITH OIDS;

CREATE TABLE "precos" (
    "tabela" smallint NOT NULL,
    "insumo_grupo" character varying(6) NOT NULL,
    "insumo" character varying(10) NOT NULL,
    "fornecedor" smallint NOT NULL,
    "custo_produtivo" real,
    "custo_improdutivo" real,
    "data_coleta" timestamp without time zone,
    "data_inclusao" timestamp without time zone,
    "data_alteracao" timestamp without time zone,
    "data_cancelamento" timestamp without time zone,
    "uid_inclusao" character varying(10),
    "uid_alteracao" character varying(10),
```

```
    "uid_cancelamento" character varying(10),
    CONSTRAINT "precos_fornecedor_fk" FOREIGN KEY (fornecedor) REFERENCES
cadastro.fornecedor(codigo_fornecedor) ON UPDATE RESTRICT ON DELETE RESTRICT,
    CONSTRAINT "precos_insumos_fk" FOREIGN KEY (insumo_grupo, insumo) REFERENCES
insumos(grupo, insumo),
    CONSTRAINT "precos_pk" PRIMARY KEY (tabela, insumo_grupo, insumo, fornecedor)
) WITH OIDS;
```

phpPgAdmin

Observe que na tabela de destino (precos), de onde queremos trazer uma informação, a ordem da chave é "tabela, insumo_grupo, insumo e fornecedor". Já em coposicoes, estes campos estão em ordem diferente. Mas no momento de criar a chave estrangeira devemos adicionar na mesma ordem da precos. Veja abaixo.

- Abrir a tabela (composicoes) para a qual queremos inserir uma chave estrangeira no phpPgAdmin

- Clicar na aba Restrição

- Clicar em adicionar chave estrangeira

- Digitar um nome para a chave: composicoes_precos_fk

- Agora vem a dica: selecione nesta ordem à esquerda e clique na seta dupla após selecionar cada um:

```
 - primeiro tabela

 - depois insumo_grupo

 - depois insumo

 - por fim fornecedor (caso altere esta ordem a chave não será criada)
```

- Em Tabela alvo selecione preco

- Clique em Adicionar (abaixo)

- Á esquerda selecione os quatro campos de precos (tabela, insumo_grupo, insumo e fornecedor) e clique na seta dupla para a direita.

- Clique em Adicionar abaixo.

PGAdmin

- Abrir a tabela no PgAdmin (duplo clique no nome da tabela à esquerda)

- Clique em Constraints

- Selecione Fireign Key abaixo e clique em Add

- Digite um nome em name "composicoes_precos_fk"

- Em references selecione "precos"

- Clique na aba Columns

- Em Local column selecione 'tabela'

- Em Referecing selecione 'tabela' e clique em Add

- De forma semelhante adicione os 3 pares restantes: insumo_grupo - insumo_grupo, insumo - insumo e fornecedor - fornecedor.

Tive problema ao adicionar insumo, então adicionei fornecedor antes e depois insumo e foi bem.

- Então clique em OK, confira e OK novamente.

PlPgSQL Tutorial

Esta é uma das linguagens de funções armazenadas com que trabalha o PostgreSQL. É a mais popular das linguagens no PostgreSQL.

Delimitadores

O código de uma função plpgsql é especificado em CREATE FUNCTION como uma string literal delimitado por aspas.

1 Apóstrofo (corpo da função)

CREATE FUNCTION olamundo() RETURNS integer AS '

. . . .

' LANGUAGE plpgsql;

Inicia após AS e termina antes de LANGUAGE

2 Apóstrofos (mascar string literal no corpo da função)

a_output := *Blah*;

SELECT * FROM users WHERE f_name=*foobar*;

4 Apóstrofos (string constante no corpo da função)

a_output := a_output || *AND name LIKE '**foobar**' AND xyz*

Mais detalhes na documentação oficial.

Declaração de Variáveis

```
user_id integer;
quantity numeric(5);
url varchar;
myrow tablename%ROWTYPE;
myfield tablename.columnname%TYPE;
arow RECORD;
```

Sintaxe geral da declaração de variáveis:

nome [CONSTANT] tipo [NOT NULL] [{ DEFAULT | := } expressão];

Exemplos:

```
quantity integer DEFAULT 32;
url varchar := 'http://mysite.com';
user_id CONSTANT integer := 10;
```

Declarado Funções

```
CREATE FUNCTION func_escopo() RETURNS integer AS $$
DECLARE
quantidade integer := 30;

BEGIN
RAISE NOTICE 'Aqui a quantidade é %', quantidade; -- A quantidade aqui é 30
quantidade := 50;
--
-- Criar um sub-bloco
--
DECLARE
quantidade integer := 80;
BEGIN
RAISE NOTICE 'Aqui a quantidade é %', quantidade; -- A quantidade aqui é 80
END;
RAISE NOTICE 'Aqui a quantidade é %', quantidade; -- A quantidade aqui é 50
RETURN quantidade;
END;
$$ LANGUAGE plpgsql;

Execute
SELECT func_escopo() as escopo;
```

Obs.: O psql exibe as mensagens disparadas pelo RAISE.

Obs2.: BEGIN e END na plpgsql não agrupa transações, apenas grupos de comandos.

Alias para Parâmetros de Funções

Sintaxe:

nome ALIAS FOR $n;

Exemplos:

```
CREATE FUNCTION vendas_taxa(real) RETURNS real AS $$
DECLARE
    subtotal ALIAS FOR $1;
BEGIN
    RETURN subtotal * 0.06;
END;
$$ LANGUAGE plpgsql;

$1 é um alias para o parâmetro da função que é do tipo real.

CREATE FUNCTION instr(varchar, integer) RETURNS integer AS $$
DECLARE
    v_string ALIAS FOR $1;
    index ALIAS FOR $2;
BEGIN
    -- Aqui fazemos alguns cálculos usando v_string e index
END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION concat_campos(tab tbl_clientes) RETURNS text AS $$
BEGIN
    RETURN tab.nome || '' '' || tab.email || '' '' || tab.cidade || '' '' ||
tab.estado;
END;
$$ LANGUAGE plpgsql;
```

Também podemos (mais claro) usar os nomes dos parâmetros explicitamente:

```
CREATE FUNCTION sales_tax(subtotal real, OUT tax real) AS $$
BEGIN
    tax := subtotal * 0.06;
END;
$$ LANGUAGE plpgsql;

O parâmetro OUT (output, saída) é mais útil quando retorna múltiplos parâmetros:

CREATE FUNCTION sum_n_product(x int, y int, OUT sum int, OUT prod int) AS $$
BEGIN
    sum := x + y;
    prod := x * y;
END;
$$ LANGUAGE plpgsql
```

Executando Funções

Usa-se o select para executar uma função, como se fosse uma view ou consulta comum.

select sum_n_product(5, 6);

Tipos Polimórficos

Quanto tipos polimórficos (anyelement e anyarray) são usados para declarar de funções, um parâmetro especial ($0) é criado. Este tipo de dados é o atual retorno da função.

É inicializado como NULL e pode ser modificado pela função.

Exemplo

Função que trabalha com qualquer tipo de dados e que suporta o operador +:

```
CREATE FUNCTION add_three_values(v1 anyelement, v2 anyelement, v3 anyelement)
RETURNS anyelement AS $$
DECLARE
    result ALIAS FOR $0;
BEGIN
    result := v1 + v2 + v3;
    RETURN result;
END;
$$ LANGUAGE plpgsql;

Ou:

CREATE FUNCTION add_three_values(v1 anyelement, v2 anyelement, v3 anyelement,
                                 OUT sum anyelement)
AS $$
BEGIN
    sum := v1 + v2 + v3;
END;
$$ LANGUAGE plpgsql;
```

Veja, que ao usar OUT não há necessidade de retorno (já está implícito).

Copiando Tipos

variavel%TYPE

%TYPE fornece o tipo de dados de uma variável ou de um campo de tabela.

Para declarar uma variável com o mesmo tipo de dado de usuarios.id_usuario deve ser escrito:

id_usuario usuarios.id_usuario%TYPE;

Tipos row

Pode armazenar um registro resultante de um SELECT ou de um FOR.

nome nome_da_tabela%ROWTYPE;

nome nome_do_tipo_composto;

Os campos podem ser acessados com nomevariavel.nomecampo;

Exemplo de uso:

```
CREATE FUNCTION mesclar_campos(t_linha nome_da_tabela) RETURNS text AS $$
DECLARE
t2_linha nome_tabela2%ROWTYPE;
BEGIN
SELECT * INTO t2_linha FROM nome_tabela2 WHERE ... ;
RETURN t_linha.f1 || t2_linha.f3 || t_linha.f5 || t2_linha.f7;
END;
$$ LANGUAGE plpgsql;
SELECT mesclar_campos(t.*) FROM nome_da_tabela t WHERE ... ;
```

Tipo registro (record)

nome record;

Renomeando Variáveis

RENAME nome_antigo TO novo_nome;

```
CREATE FUNCTION logfunc1(logtxt text) RETURNS timestamp AS $$
BEGIN
INSERT INTO logtable VALUES (logtxt, 'now');
RETURN 'now';
END;
$$ LANGUAGE plpgsql;
e
CREATE FUNCTION logfunc2(logtxt text) RETURNS timestamp AS $$
DECLARE
curtime timestamp;
BEGIN
curtime := 'now';
INSERT INTO logtable VALUES (logtxt, curtime);
RETURN curtime;
END;
$$ LANGUAGE plpgsql;
```

Atribuições

identificador := expressão;

```
SELECT INTO meu_registro * FROM emp WHERE nome_emp = meu_nome;
IF NOT FOUND THEN
RAISE EXCEPTION ''não foi encontrado o empregado %!'', meu_nome;
```

```
END IF;
```

Execução de Expressão ou Consulta sem Resultado

PERFORM create_mv('cs_session_page_requests_mv', my_query);

Não Fazer Nada

NULL;

Por exemplo, os dois fragmentos de código a seguir são equivalentes:

```
BEGIN
        y := x / 0;
EXCEPTION
        WHEN division_by_zero THEN
        NULL; -- ignorar o erro
END;
BEGIN
        y := x / 0;
EXCEPTION
        WHEN division_by_zero THEN -- ignorar o erro
END;
```

Execução de Comandos Dinâmicos

EXECUTE cadeia_de_caracteres_do_comando;

```
EXECUTE 'UPDATE tbl SET '
|| quote_ident(nome_da_coluna)
|| ' = '
|| quote_literal(novo_valor)
|| ' WHERE key = '
|| quote_literal(valor_chave);

EXECUTE 'UPDATE tbl SET '
|| quote_ident(nome_da_coluna)
|| ' = $$'
|| novo_valor
|| '$$ WHERE key = '
|| quote_literal(valor_chave);
```

Estruturas de Controle

Return next

RETURN NEXT expressão;

Quando uma função PL/pgSQL é declarada como retornando SETOF algum_tipo, o procedimento a ser seguido é um pouco diferente. Neste caso, os itens individuais a serem retornados são especificados em comandos RETURNNEXT, e um comando RETURN final, sem nenhum argumento, é utilizado para indicar que a função chegou ao fim de sua execução. O comando

RETURN NEXT pode ser utilizado tanto com tipos de dado escalares quanto compostos; no último caso toda uma "tabela" de resultados é retornada.

As funções que utilizam RETURN NEXT devem ser chamadas da seguinte maneira:

SELECT * FROM alguma_função();

Condicionais

```
• IF ... THEN
• IF ... THEN ... ELSE
• IF ... THEN ... ELSE IF
• IF ... THEN ... ELSIF ... THEN ... ELSE
• IF ... THEN ... ELSEIF ... THEN ... ELSE

IF linha_demo.sexo = 'm' THEN
sexo_extenso := 'masculino';
ELSE
IF linha_demo.sexo = 'f' THEN
sexo_extenso := 'feminino';
END IF;
END IF;

IF expressão_booleana THEN
instruções
[ ELSIF expressão_booleana THEN
instruções
[ ELSIF expressão_booleana THEN
instruções
...]]
[ ELSE
instruções ]
END IF;

IF numero = 0 THEN
resultado := 'zero';
ELSIF numero > 0 THEN
resultado := 'positivo';
ELSIF numero < 0 THEN
resultado := 'negativo';
ELSE
-- hmm, a única outra possibilidade é que o número seja nulo
resultado := 'NULL';
END IF;

Laços

[<<rótulo>>]
LOOP
instruções
END LOOP;

Exit

EXIT [ rótulo ] [ WHEN expressão ];
```

```
LOOP
-- algum processamento
IF contador > 0 THEN
EXIT; -- sair do laço
END IF;
END LOOP;
LOOP
-- algum processamento
EXIT WHEN contador > 0; -- mesmo resultado do exemplo acima
END LOOP;
BEGIN
-- algum processamento
IF estoque > 100000 THEN
EXIT; -- causa a saída do bloco BEGIN
END IF;
END;
```

While

```
[<<rótulo>>]
WHILE expressão LOOP
instruções
END LOOP;

WHILE quantia_devida > 0 AND saldo_do_certificado_de_bonus > 0 LOOP
-- algum processamento
END LOOP;
WHILE NOT expressão_booleana LOOP
-- algum processamento
END LOOP;
```

For (laços internos)

```
[<<rótulo>>]
FOR nome IN [ REVERSE ] expressão .. expressão LOOP
instruções
END LOOP;

FOR i IN 1..10 LOOP
-- algum processamento
RAISE NOTICE 'i é %', i;
END LOOP;
FOR i IN REVERSE 10..1 LOOP
-- algum processamento
END LOOP;
```

Laços através do resultado de consultas

```
[<<rótulo>>]
FOR registro_ou_linha IN comando LOOP
instruções
END LOOP;

CREATE FUNCTION cs_refresh_mviews() RETURNS integer AS $$
DECLARE
        mviews RECORD;
```

```
BEGIN
        PERFORM cs_log('Atualização das visões materializadas...');
                FOR mviews IN SELECT * FROM cs_materialized_views ORDER BY
sort_key LOOP
                -- Agora "mviews" possui um registro de cs_materialized_views
                PERFORM cs_log('Atualizando a visão materializada ' ||
quote_ident(mviews.mv_name) || ' ...');
                EXECUTE 'TRUNCATE TABLE ' || quote_ident(mviews.mv_name);
                EXECUTE 'INSERT INTO ' || quote_ident(mviews.mv_name) || ' ' ||
mviews.mv_query;

        END LOOP;
        PERFORM cs_log('Fim da atualização das visões materializadas.');
        RETURN 1;
END;
$$ LANGUAGE plpgsql;

[<<rótulo>>]
FOR registro_ou_linha IN EXECUTE texto_da_expressão LOOP
instruções
END LOOP;
```

Capturar Erros

```
[ <<rótulo>> ]
[ DECLARE
declarações ]
BEGIN
instruções
EXCEPTION
WHEN condição [ OR condição ... ] THEN
instruções_do_tratador
[ WHEN condição [ OR condição ... ] THEN
instruções_do_tratador
... ]
END;


INSERT INTO minha_tabela(nome, sobrenome) VALUES('Tom', 'Jones');
BEGIN
        UPDATE minha_tabela SET nome = 'Joe' WHERE sobrenome = 'Jones';
        x := x + 1;
        y := x / 0;
EXCEPTION
        WHEN division_by_zero THEN
                RAISE NOTICE 'capturado division_by_zero';
                RETURN x;
END;
```

Declaração de Variáveis do tipo Cursor

Todos os acessos aos cursores na linguagem PL/pgSQL são feitos através de variáveis cursor, que sempre são do tipo de dado especial refcursor. Uma forma de criar uma variável cursor é simplesmente declará-la como sendo do tipo refcursor. Outra forma é utilizar a sintaxe de declaração de cursor, cuja forma geral é:

nome CURSOR [(argumentos)] FOR comando ;

Exemplos:

```
DECLARE
curs1 refcursor;
curs2 CURSOR FOR SELECT * FROM tenk1;
curs3 CURSOR (chave integer) IS SELECT * FROM tenk1 WHERE unico1 = chave;

CREATE TABLE teste (col text);
INSERT INTO teste VALUES ('123');
CREATE FUNCTION reffunc(refcursor) RETURNS refcursor AS '
BEGIN
OPEN $1 FOR SELECT col FROM teste;
RETURN $1;
END;
' LANGUAGE plpgsql;
BEGIN;
SELECT reffunc('funccursor');
reffunc
------------
funccursor
(1 linha)
FETCH ALL IN funccursor;

COMMIT;
```

Erros e Mensagens

RAISE nível 'formato' [, variável [, ...]];

Os níveis possíveis são DEBUG, LOG, INFO, NOTICE, WARNING, e EXCEPTION. O nível EXCEPTION causa um erro (que normalmente interrompe a transação corrente); os outros níveis apenas geram mensagens com diferentes níveis de prioridade. Se as mensagens de uma determinada prioridade são informadas ao cliente, escritas no log do servidor, ou as duas coisas, é controlado pelas variáveis de configuração log_min_messages e client_min_messages.

Procedimentos de Gatilho (Trigger)

É criado pelo comando CREATE FUNCTION, declarando o procedimento como uma função sem argumentos e que retorna o tipo trigger. Deve ser observado que a função deve ser declarada sem

argumentos, mesmo que espere receber os argumentos especificados no comando CREATE TRIGGER — os argumentos do gatilho são passados através de TG_ARGV, conforme descrito abaixo.

Quando uma função escrita em PL/pgSQL é chamada como um gatilho, diversas variáveis especiais são criadas automaticamente no bloco de nível mais alto. São estas:

NEW

Tipo de dado RECORD; variável contendo a nova linha do banco de dados, para as operações de INSERT/UPDATE nos gatilhos no nível de linha. O valor desta variável é NULL nos gatilhos no nível de instrução.

OLD

Tipo de dado RECORD; variável contendo a antiga linha do banco de dados, para as operações de UPDATE/DELETE nos gatilhos no nível de linha. O valor desta variável é NULL nos gatilhos no nível de instrução.

TG_NAME

Tipo de dado name; variável contendo o nome do gatilho disparado.

TG_WHEN

Tipo de dado text; uma cadeia de caracteres contendo BEFORE ou AFTER, dependendo da definição do gatilho.

TG_LEVEL

Tipo de dado text; uma cadeia de caracteres contendo ROW ou STATEMENT, dependendo da definição do gatilho.

TG_OP

Tipo de dado text; uma cadeia de caracteres contendo INSERT, UPDATE, ou DELETE, informando para qual operação o gatilho foi disparado.

TG_RELID

Tipo de dado oid; o ID de objeto da tabela que causou o disparo do gatilho.

TG_RELNAME

Tipo de dado name; o nome da tabela que causou o disparo do gatilho.

TG_NARGS

Tipo de dado integer; o número de argumentos fornecidos ao procedimento de gatilho na instrução CREATE

TG_ARGV[]

Tipo de dado matriz de text; os argumentos da instrução CREATE TRIGGER. O contador do índice começa por 0.

Índices inválidos (menor que 0 ou maior ou igual a tg_nargs) resultam em um valor nulo.

Exemplo:

O gatilho deste exemplo garante que:

- quando é inserida ou atualizada uma linha na tabela, fica sempre registrado nesta linha o usuário que efetuou a inserção ou a atualização

- quando isto ocorreu.

- além disso, o gatilho verifica se é fornecido o nome do empregado

- e se o valor do salário é um número positivo.

```
CREATE TABLE emp (
        nome_emp text,
        salario integer,
        ultima_data timestamp,
        ultimo_usuario text
);

CREATE FUNCTION emp_gatilho() RETURNS trigger AS $emp_gatilho$
        BEGIN
                -- Verificar se foi fornecido o nome e o salário do empregado
                IF NEW.nome_emp IS NULL THEN
                        RAISE EXCEPTION 'O nome do empregado não pode ser nulo';
                END IF;
                IF NEW.salario IS NULL THEN
                        RAISE EXCEPTION '% não pode ter um salário nulo',
NEW.nome_emp;
                END IF;
                -- Quem paga para trabalhar?
                IF NEW.salario < 0 THEN
                        RAISE EXCEPTION '% não pode ter um salário negativo',
NEW.nome_emp;
                END IF;
                -- Registrar quem alterou a folha de pagamento e quando
                NEW.ultima_data := 'now';
                NEW.ultimo_usuario := current_user;
                RETURN NEW;
        END;

$emp_gatilho$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER emp_gatilho BEFORE INSERT OR UPDATE ON emp
        FOR EACH ROW EXECUTE PROCEDURE emp_gatilho();

INSERT INTO emp (nome_emp, salario) VALUES ('João',1000);
INSERT INTO emp (nome_emp, salario) VALUES ('José',1500);
INSERT INTO emp (nome_emp, salario) VALUES ('Maria',2500);

SELECT * FROM emp;
```

Gatilho para registrar inserções e atualizações

```
CREATE TABLE emp (
        nome_emp text,
        salario integer,
        usu_cria text, -- Usuário que criou a linha
        data_cria timestamp, -- Data da criação da linha
        usu_atu text, -- Usuário que fez a atualização
        data_atu timestamp -- Data da atualização
);

CREATE FUNCTION emp_gatilho() RETURNS trigger AS $emp_gatilho$
        BEGIN
                -- Verificar se foi fornecido o nome do empregado
                IF NEW.nome_emp IS NULL THEN
                        RAISE EXCEPTION 'O nome do empregado não pode ser nulo';
                END IF;
                IF NEW.salario IS NULL THEN
                        RAISE EXCEPTION '% não pode ter um salário nulo',
NEW.nome_emp;
                END IF;
                -- Quem paga para trabalhar?
                IF NEW.salario < 0 THEN
                        RAISE EXCEPTION '% não pode ter um salário negativo',
NEW.nome_emp;
                END IF;
                -- Registrar quem criou a linha e quando
                IF (TG_OP = 'INSERT') THEN
                        NEW.data_cria := current_timestamp;
                        NEW.usu_cria := current_user;
                -- Registrar quem alterou a linha e quando
                ELSIF (TG_OP = 'UPDATE') THEN
                        NEW.data_atu := current_timestamp;
                        NEW.usu_atu := current_user;
                END IF;
                RETURN NEW;
END;
$emp_gatilho$ LANGUAGE plpgsql;

CREATE TRIGGER emp_gatilho BEFORE INSERT OR UPDATE ON emp
        FOR EACH ROW EXECUTE PROCEDURE emp_gatilho();

INSERT INTO emp (nome_emp, salario) VALUES ('João',1000);
INSERT INTO emp (nome_emp, salario) VALUES ('José',1500);
INSERT INTO emp (nome_emp, salario) VALUES ('Maria',250);
UPDATE emp SET salario = 2500 WHERE nome_emp = 'Maria';
```

```
SELECT * FROM emp;
```

Gatilho para auditoria

Todas as operações na tabela emp serão registradas na tabela emp_audit

```
CREATE TABLE emp (
        nome_emp text NOT NULL,
        salario integer
);

CREATE TABLE emp_audit(
        operacao char(1) NOT NULL,
        usuario text NOT NULL,
        data timestamp NOT NULL,
        nome_emp text NOT NULL,
        salario integer
);

CREATE OR REPLACE FUNCTION processa_emp_audit() RETURNS TRIGGER AS $emp_audit$
        BEGIN
                --
                -- Cria uma linha na tabela emp_audit para refletir a operação
                -- realizada na tabela emp. Utiliza a variável especial TG_OP
                -- para descobrir a operação sendo realizada.
                --
                IF (TG_OP = 'DELETE') THEN
                        INSERT INTO emp_audit SELECT 'E', user, now(), OLD.*;
                        RETURN OLD;
                ELSIF (TG_OP = 'UPDATE') THEN
                        INSERT INTO emp_audit SELECT 'A', user, now(), NEW.*;
                        RETURN NEW;
                ELSIF (TG_OP = 'INSERT') THEN
                        INSERT INTO emp_audit SELECT 'I', user, now(), NEW.*;
                        RETURN NEW;
                END IF;
                RETURN NULL; -- o resultado é ignorado uma vez que este é um
gatilho AFTER
        END;
$emp_audit$ language plpgsql;

CREATE TRIGGER emp_audit
AFTER INSERT OR UPDATE OR DELETE ON emp
        FOR EACH ROW EXECUTE PROCEDURE processa_emp_audit();

INSERT INTO emp (nome_emp, salario) VALUES ('João',1000);
INSERT INTO emp (nome_emp, salario) VALUES ('José',1500);
INSERT INTO emp (nome_emp, salario) VALUES ('Maria',250);
UPDATE emp SET salario = 2500 WHERE nome_emp = 'Maria';
DELETE FROM emp WHERE nome_emp = 'João';

SELECT * FROM emp;

SELECT * FROM emp_audit;
```

Autoditoria ao nível de campos

```
CREATE TABLE emp (
        id serial PRIMARY KEY,
        nome_emp text NOT NULL,
        salario integer
);

CREATE TABLE emp_audit(
        usuario text NOT NULL,
        data timestamp NOT NULL,
        id integer NOT NULL,
        coluna text NOT NULL,
        valor_antigo text NOT NULL,
        valor_novo text NOT NULL
);

CREATE OR REPLACE FUNCTION processa_emp_audit() RETURNS TRIGGER AS $emp_audit$
BEGIN
                --
                -- Não permitir atualizar a chave primária
                --
                IF (NEW.id <> OLD.id) THEN
                        RAISE EXCEPTION 'Não é permitido atualizar o campo ID';
                END IF;
                --
                -- Inserir linhas na tabela emp_audit para refletir as alterações
                -- realizada na tabela emp.
                --
                IF (NEW.nome_emp <> OLD.nome_emp) THEN
                        INSERT INTO emp_audit SELECT current_user,
current_timestamp,
                        NEW.id, 'nome_emp', OLD.nome_emp, NEW.nome_emp;
                END IF;
                IF (NEW.salario <> OLD.salario) THEN
                        INSERT INTO emp_audit SELECT current_user,
current_timestamp,
                        NEW.id, 'salario', OLD.salario, NEW.salario;
                END IF;
                RETURN NULL; -- o resultado é ignorado uma vez que este é um
gatilho AFTER
        END;
$emp_audit$ language plpgsql;

CREATE TRIGGER emp_audit
AFTER UPDATE ON emp
        FOR EACH ROW EXECUTE PROCEDURE processa_emp_audit();

INSERT INTO emp (nome_emp, salario) VALUES ('João',1000);
INSERT INTO emp (nome_emp, salario) VALUES ('José',1500);
INSERT INTO emp (nome_emp, salario) VALUES ('Maria',2500);
UPDATE emp SET salario = 2500 WHERE id = 2;
UPDATE emp SET nome_emp = 'Maria Cecília' WHERE id = 3;
UPDATE emp SET id=100 WHERE id=1;
ERRO: Não é permitido atualizar o campo ID

SELECT * FROM emp;
```

```
SELECT * FROM emp_audit;
```

Gatilho para manter uma tabela sumário

O esquema que está detalhado a seguir é parcialmente baseado no exemplo Grocery Store do livro The Data Warehouse Toolkit de Ralph Kimball.

```
--
-- Main tables - time dimension and sales fact.
--

CREATE TABLE time_dimension (
        time_key integer NOT NULL,
        day_of_week integer NOT NULL,
        day_of_month integer NOT NULL,
        month integer NOT NULL,
        quarter integer NOT NULL,
        year integer NOT NULL
);

CREATE UNIQUE INDEX time_dimension_key ON time_dimension(time_key);

CREATE TABLE sales_fact (
        time_key integer NOT NULL,
        product_key integer NOT NULL,
        store_key integer NOT NULL,
        amount_sold numeric(12,2) NOT NULL,
        units_sold integer NOT NULL,
        amount_cost numeric(12,2) NOT NULL
);

CREATE INDEX sales_fact_time ON sales_fact(time_key);

--
-- Summary table - sales by time.
--

CREATE TABLE sales_summary_bytime (
        time_key integer NOT NULL,
        amount_sold numeric(15,2) NOT NULL,
        units_sold numeric(12) NOT NULL,
        amount_cost numeric(15,2) NOT NULL
);

CREATE UNIQUE INDEX sales_summary_bytime_key ON sales_summary_bytime(time_key);

--
-- Function and trigger to amend summarized column(s) on UPDATE, INSERT, DELETE.
--

CREATE OR REPLACE FUNCTION maint_sales_summary_bytime() RETURNS TRIGGER AS
$maint_sales_summary_bytime$
        DECLARE
                delta_time_key integer;
```

```
                delta_amount_sold numeric(15,2);
                delta_units_sold numeric(12);
                delta_amount_cost numeric(15,2);
        BEGIN
                -- Work out the increment/decrement amount(s).
                IF (TG_OP = 'DELETE') THEN
                        delta_time_key = OLD.time_key;
                        delta_amount_sold = -1 * OLD.amount_sold;
                        delta_units_sold = -1 * OLD.units_sold;
                        delta_amount_cost = -1 * OLD.amount_cost;
                ELSIF (TG_OP = 'UPDATE') THEN
                        -- forbid updates that change the time_key -
                        -- (probably not too onerous, as DELETE + INSERT is how
most
                        -- changes will be made).
                        IF ( OLD.time_key != NEW.time_key) THEN
                                RAISE EXCEPTION 'Update of time_key : % -> % not
allowed', OLD.time_key,
                                NEW.time_key;
                        END IF;
                        delta_time_key = OLD.time_key;
                        delta_amount_sold = NEW.amount_sold - OLD.amount_sold;
                        delta_units_sold = NEW.units_sold - OLD.units_sold;
                        delta_amount_cost = NEW.amount_cost - OLD.amount_cost;
                ELSIF (TG_OP = 'INSERT') THEN
                        delta_time_key = NEW.time_key;
                        delta_amount_sold = NEW.amount_sold;
                        delta_units_sold = NEW.units_sold;
                        delta_amount_cost = NEW.amount_cost;
                END IF;
                -- Update the summary row with the new values.
                UPDATE sales_summary_bytime
                        SET amount_sold = amount_sold + delta_amount_sold,
                                units_sold = units_sold + delta_units_sold,
                                amount_cost = amount_cost + delta_amount_cost
                WHERE time_key = delta_time_key;

                -- There might have been no row with this time_key (e.g new
data!).
                IF (NOT FOUND) THEN
                        BEGIN
                                INSERT INTO sales_summary_bytime (
                                        time_key,
                                        amount_sold,
                                        units_sold,
                                        amount_cost)
                        VALUES (
                                delta_time_key,
                                delta_amount_sold,
                                delta_units_sold,
                                delta_amount_cost
                                );
                        EXCEPTION
                                --
                                -- Catch race condition when two transactions are
adding data
                                -- for a new time_key.
```

```
                                --
                                WHEN UNIQUE_VIOLATION THEN
                                        UPDATE sales_summary_bytime
                                                SET amount_sold = amount_sold +
delta_amount_sold,
                                                        units_sold = units_sold +
delta_units_sold,
                                                        amount_cost = amount_cost
+ delta_amount_cost
                                                WHERE time_key = delta_time_key;
                        END;
                END IF;
                RETURN NULL;
        END;
$maint_sales_summary_bytime$ LANGUAGE plpgsql;

CREATE TRIGGER maint_sales_summary_bytime
AFTER INSERT OR UPDATE OR DELETE ON sales_fact
        FOR EACH ROW EXECUTE PROCEDURE maint_sales_summary_bytime();
```

Este é um resumo da documentação oficial em inglês e em português do Brasil, que devem ser consultados para informações mais detalhadas em:

- http://www.postgresql.org/docs/8.2/interactive/index.html

- http://pgdocptbr.sourceforge.net/pg80/index.html

Administração dos SGBDs/SQLite

De Wikibooks

Conteúdo

- 1 SQLite na Wikipedia
- 2 Características atuais
- 3 Criando Banco
- 4 Efetuando Consultas
- 5 Usando SQLite com PHP
- 6 Exemplo de aplicativo simples em PHP acessando SQLite

SQLite na Wikipedia

http://pt.wikipedia.org/wiki/Sqlite

SQLite é uma biblioteca C que implementa um banco de dados SQL embutido. Programas que usam a biblioteca SQLite podem ter acesso a banco de dados SQL sem executar um processo RDBMS separado.

SQLite não é uma biblioteca de cliente usada para conectar com um grande servidor de banco de dados. SQLite é o servidor. A biblioteca SQLite lê e escreve diretamente para e do arquivo do banco de dados no disco.

O uso do SQLite é recomendado onde a simplicidade da administração, implementação e manutenção são mais importantes que incontáveis recursos que SGBDs mais voltados para aplicações complexas possivelmente implementam. Entretando situações onde a simplicidade é a melhor escolha são muito mais freqüentes do que pode-se imaginar.

Exemplos de uso do SQLite são, não restrito a, sites (com menos de cem mil requisições por dia), dispositivos e sistemas embarcados, aplicações desktop, ferramentas estatísticas e de análise, aprendizado de banco de dados, implementação de novas extensões à SQL. Não se recomenda o uso do SQLite para sites com muitos acessos, grande quantidades de dados (talvez maior que algumas duzias de gigabytes), sistemas com grande concorrência, aplicações cliente/servidor.

Programas que usam a biblioteca SQLite podem ter acesso a banco de dados SQL sem executar um processo RDBMS separado.

A biblioteca SQLite lê e escreve diretamente para e do arquivo de banco de dados no disco.

Características atuais

- Transações são atômicas, consistentes, isoladas e duráveis (ACID) mesmo que o sistema trave ou a energia falhe.

- Configuração-zero - nenhuma instalação ou administração necessária.

- Implementação da maior parte do SQL92.

- Um banco de dados completo é armazenado em apenas um arquivo de sistema.

- Arquivos de banco de dados podem ser livremente compartilhados entre máquinas com diferentes ordens de byte.

- Suporta bases de dados de até 2 terabytes de tamanho.

- Tamanho de strings e BLOBs limitados apenas pela memória disponível.

- Mais rápido que populares bancos de dados cliente/servidor para a maioria das operações comuns.

- API simples e fácil de usar.

- TCL bindings inclusas. Bindings para a maioria das linguagens disponíveis separadamente.

- Código fonte bem comentado, com mais de 95% coberto por testes.

- Auto-contido: sem dependências externas.

- Fontes estão em domínio público. Use para qualquer propósito.

A distribuição SQLite vem com um programa de linha de comando (sqlite (http://sqlitebrasil.codigolivre.org.br/?pagina=doc/sqlite)) que pode ser usado para administrar um banco de dados SQLite e que serve como exemplo de como usar a biblioteca SQLite.

Além do programa em linha de comando, você pode utilizar alguns dos programas de terceiros com interface gráfica, como o SQLiteManager (http://sqlitemanager.sourceforge.net/) (web, no estilo PHPMyAdmin) ou o SQLiteBrowser (http://sqlitebrowser.sourceforge.net/) (QT).

O SQLite está embutido no PHP (http://br.php.net/sqlite) 5 e disponível como extensão no PHP 4. Assim, qualquer aplicação PHP pode utilizar um banco de dados sem necessitar de um SGDB.

Há um driver (http://dba.openoffice.org/drivers/sqlite/index.html) (alpha) disponível para conectar bancos de dados SQLite com o OpenOffice.org. Tornando possível a criação de relatórios, formulários, etc.

Suas características o tornam ideal para desenvolver programas standalone, pequenos e médios sites, etc. Veja quando usar (e não usar) (http://localhost/sqlitewww/?pagina=doc/quando) o SQLite.

Criando Banco

sqlite clientes (Com este comando ele cria o banco clientes e já acessa a console deste banco)

Criando Tabela (A sintaxe não tem diferença dos grandes SGBDs, é puro SQL)

```
CREATE TABLE cliente (
        cpf     VARCHAR(11) PRIMARY KEY,
        nome VARCHAR(45),
        fone VARCHAR(10)
);
```

Efetuando Consultas

Inserindo Registros (Também puro SQL)

INSERT INTO cliente (cpf, nome, fone) VALUES ('11111111111', 'João Abreu', '34543456');

SELECT * FROM cliente;

Outros comandos do sqlite. Estando na console apenas digite ".help":

```
sqlite> .help
.databases             List names and files of attached databases
.dump ?TABLE? ...      Dump the database in a text format
.echo ON|OFF           Turn command echo on or off
.exit                  Exit this program
.explain ON|OFF        Turn output mode suitable for EXPLAIN on or off.
.header(s) ON|OFF      Turn display of headers on or off
.help                  Show this message
.indices TABLE         Show names of all indices on TABLE
.mode MODE             Set mode to one of "line(s)", "column(s)",
                       "insert", "list", or "html"
.mode insert TABLE     Generate SQL insert statements for TABLE
.nullvalue STRING      Print STRING instead of nothing for NULL data
.output FILENAME       Send output to FILENAME
.output stdout         Send output to the screen
.prompt MAIN CONTINUE  Replace the standard prompts
.quit                  Exit this program
.read FILENAME         Execute SQL in FILENAME
.schema ?TABLE?        Show the CREATE statements
.separator STRING      Change separator string for "list" mode
.show                  Show the current values for various settings
.tables ?PATTERN?      List names of tables matching a pattern
.timeout MS            Try opening locked tables for MS milliseconds
.width NUM NUM ...     Set column widths for "column" mode
```

Escrevendo o Resultado em um Arquivo:

```
sqlite> .mode list
sqlite> .separator |
sqlite> .output test_file_1.txt
sqlite> select * from tbl1;
sqlite> .exit
$ cat test_file_1.txt
hello|10
goodbye|20
```

Usando SQLite com PHP

```
<?php

$db=sqlite_open("/home/ribafs/bancos/clientes.db");

$registros = sqlite_query($db, "SELECT * FROM cliente");

while ($i = sqlite_fetch_array($registros)) {
        print $i[0] . "__" . $i[1]. "__" . $i[2]. "<br>";
}

?>
```

Exemplo de aplicativo simples em PHP acessando SQLite

http://www.ribafs.net/down/appsexemplo/sqlite/iniciante_sqlitephp.zip

Aplicativos em PHP/Geradores de Aplicativos

De Wikibooks

7 . Geradores de Aplicativos

"Um livro é um mudo que fala, um surdo que responde, um cego que guia, um morto que vive." (Padre António Vieira)

1. phpCodeGenie (com MySQL)
2. SQLMaestro (com MySQL, Oracle, MS SQL Server, PostgreSQL, SQLite, Firebird e MaxDB)
3. phpMyEdit (com MySQL)
4. DadaBik (com vários)
5. PHPLibDev (com PostgreSQL)
6. Web Form Generator (com MySQL)

Geradores de Aplicativos

Segundo a wikipédia em http://pt.wikipedia.org/wiki/geradores

Dentro do diversificado leque de categorias de ferramentas que prestam apoio às atividades da Engenharia de Software (CASE), uma específica vem ganhando cada vez mais destaque e, sobre ela, tem-se aplicado muito investimento nos últimos tempos: as Ferramentas de Geração de Código, ou simplesmente Geradores de Código. Destes surgiu o Gerador de Aplicativos.

Dessa forma, Gerador de Código é aquela ferramenta que possui a capacidade de gerar código a partir de um determinado modelo de software. Inclusive, de acordo com alguns pontos de vista e a partir das características específicas do tipo de Gerador de Código, ele passa a ser conversor de códigos de linguagens distintas. Isso acontece, por exemplo, com o compilador, que transforma um código escrito através de uma linguagem de programação para código de máquina ou código objeto.

Tipos mais comuns

Alguns tipos de Gerador de Código valem a pena serem mencionados. Eles possuem características

específicas e uma maneira de trabalhar própria que está diretamente ligada aos seus objetivos. O mais básico tipo de gerador de código é o já mencionado compilador, como pode ser visto no artigo Code generation.

Outros dois tipos de gerador de código que estão sendo cada vez mais utilizados em ambiente de produção de software são os geradores de código de IDE's visuais e os geradores de código baseados em templates.

Geradores de código de IDE's visuais

Algumas ferramentas de edição integrada (IDE's), principalmente as chamadas ferramentas de programação visual, precisaram adotar um tipo de gerador de código capaz de traduzir, em linhas de código, tudo aquilo que é "desenhado" pelo usuário desenvolvedor. Dessa maneira, por trás de ações como arrastar componentes para uma janela de uma aplicação que está sendo construída e configurar visualmente seu comportamento e forma de apresentação, vai existir um mecanismo responsável por transformar em instruções que, quando executadas, reproduzam exatamente aquilo que foi definido pelo usuário desenvolvedor, através de linhas de código da linguagem de programação correspondente à ferramenta IDE. São os também chamados mecanismos WYSIWYG. Tal mecanismo pode ser encontrado em ferramentas como o Borland Delphi, o Oracle JDeveloper ou o Microsoft Visual Studio, quando do desenvolvimento de aplicações tanto de ambiente desktop, como Web.

Geradores de código baseados em templates

Ultimamente, os geradores de código baseados em templates começaram a figurar o elenco das ferramentas CASE presentes em ambiente de produção de software como sendo uma opção para a geração automática e massiva de funcionalidades de aplicações, seja de maneira completa ou parcial.

Ferramentas tais como Velocity (do Apache Jakarta Project), Transformica (do Hammurapi Group) ou CodeFSW (da Unitech Tecnologia de Informação), além de diversas outras que vêm surgindo neste cenário, a depender de como sejam utilizadas, podem viabilizar a geração de código para as diversas camadas de uma aplicação, a partir de templates e parâmetros de entrada. Dessa maneira, tendo a necessidade de possuir diversas funcionalidades semelhantes em suas estrutura e que englobem códigos das camadas da aplicação adotadas em sua arquitetura, uma aplicação pode ser desenvolvida, em grande parte, com a ajuda de tais ferramentas.

Principais vantagens

Certamente que as vantagens adquiridas ao se adotar Geradores de Código em um ambiente de produção de software só aparecem quando da boa utilização dessas ferramentas. A partir do uso adequado dos geradores de código, sejam quais tipos forem, a equipe de produção de software perceberá um ganho significativo em sua produtividade, sendo capaz de realizar a construção de funcionalidades das aplicações de maneira mais eficiente e segura, garantindo também a diminuição dos custos de projeto.

Uma outra vantagem que é vista de maneira bastante clara se refere à padronização percebida no código gerado, visto que pode se basear, a depender do tipo, tanto em um template utilizado, como na filosofia de trabalho da ferramenta IDE. E está definitivamente consolidada a idéia de que é um ponto positivo com relação à qualidade do software, ter uma aplicação em cujo código-fonte se percebe um estilo único de programação.

Por fim, pode-se considerar também como uma vantagem do uso de geradores de código, a depender novamente de como são utilizados, a construção de aplicações isentas de grande parte dos artifícios que facilitam a programação propriamente, mas, por outro lado, afetam diretamente o quesito performance. É sabido, por exemplo, que o uso de Reflexão, artifício que é suportado em linguagens como Java e C# e que permite a introspecção e manipulação de classes, interfaces e objetos no geral, afeta negativamente a performance de funcionalidades de aplicações. O uso adequado de geradores de código no processo de desenvolvimento de software pode fazer desnecessário o uso de tal artifício, melhorando o desempenho do produto final. Muitas outras vantagens podem ser percebidas com o uso de geradores de código no processo de construção de software. Mas também vale alertar que tais ferramentas, em vez de contribuir com vantagens, podem atrapalhar o andamento do projeto quando é feito uso inadequado ou desnecessário das mesmas.

Geradores de código em PHP

Como para diversas outras linguagens, também existem bons geradores de código em PHP, existindo ferramentas disponíveis em software livre e software proprietário. Daremos maior ênfase aquelas da categoria software livre, que estão mais de acordo com a filosofia deste projeto e também porque apresentam resultados semelhantes e as vezes até melhores do que as outras.

A grande maioria dos geradores de código existentes em PHP, trabalha de forma similar quando da construção das aplicações. Necessitam que um banco de dados já construído e de uma conexão com esse banco de dados e a partir das informações aí existentes, geram o código para a aplicação. Poderíamos também classificar os geradores de código em PHP em dois diferentes modelos: ativo e passivo.

No modelo ativo, o código se mantém enquanto o gerador estiver ativo, ou seja, para que o aplicativo funcione, é necessária a presença do gerador ou pelo menos de parte integrante dele (um bom exemplo nesse caso é o phpMyEdit). Já no modelo passivo, o gerador cria o código inicial e sua manutenção e funcionalidade passa a depender completamente do desenvolvedor, o PHP-MySQL Wizard pode ser enquadrado nesse modelo. Existem geradores que trabalham somente com OO, linguagem procedural ou um misto.

Vantagens adicionais:

Conforme artigo no code generation, existem vantagens adicionais em se utilizar geradores de código:

1.Padrão de qualidade: o código feito a mão tem sua qualidade variando durante o ciclo de vida de um projeto. Pode começar no alto e decair ou vice e versa. O código gerado aumenta sua qualidade com o tempo pois os erros encontrados podem ser uniformemente reparados na base do código. Nesse quesito, os geradores que têm projetos ativos e que constantemente lançam novas versões ganham pontos. A familiaridade com o gerador por parte do programador também ajuda bastante, visto que dependendo dos seus conhecimentos, ele mesmo pode fazer reparos ou customizações na ferramenta (quando open source) e não somente no código gerado.

2.Consistência: as aplicações geradas são consistentes na estrutura de classes, nomeação de variáveis etc.

3.Produtividade: os geradores constroem o código numa fração muita curta de tempo, economizando horas de trabalho (muitas vezes repetitivo). Isso libera o desenvolvedor para outras tarefas que exijam soluções mais criativas.

4.Abstração: Os geradores frequentemente fornecem uma camada de abstração entre o projeto e a base do código, facilitando o manuseio das regras de negócio.

É interessante notar que entre os desenvolvedores médios e avançados em PHP e creio que também nas outras linguagens, existe uma certa resistência ou menosprezo pelo uso de geradores. É importante estarmos abertos para utilizarmos quaisquer ferramentas que melhorem nossa produtividade, nos poupem trabalho repetitivo e pouco criativo, deixem nosso tempo livre para outras atividades e consequentemente melhorem nossa qualidade de vida. Assim, antes de iniciar seu projeto você deve se perguntar: que ferramentas usarei? Esse ou aquele gerador me será útil em alguma fase? Ele funciona corretamente? Cumpre sua finalidade? Se a resposta é sim, vá em frente e esqueça o preconceito. Com certeza existem projetos para os quais compensa até construir um gerador. Ressalte-se ainda que os geradores são somente ferramentas e a qualidade do código gerado vai depender diretamente da sua habilidade em manuseá-las, aplicar ajustes necessários, etc. Como em todo projeto de criação de software, aqueles criados com geradores de código apresentam falhas. Baterias exaustivas de testes, conhecimento profundo da ferramenta e bom planejamento ajudam a corrigí-las. Uma dica valiosa é dar preferência a geradores com projetos ativos, boa documentação, fóruns e/ou listas de discussão. Hoje, alguns dos melhores frameworks em PHP tais como o Cake, Symfony e CodeIgniter possuem ferramentas para gerar código entre suas funcionalidades.

Geradores de Aplicativos/phpCodeGenie (com MySQL)

De Wikibooks

phpCodeGenie (PCG)

Um excelente gerador de código em PHP é o phpCodeGenie (PCG). Ele é um gerador do tipo passivo e permite níveis bastante altos de customização. Conforme a apresentação do PCG "Melhor que gastar muito tempo criando código comum, é poder gastá-lo na lógica de negócios de nossas aplicações e deixar PCG fazer o código aborrecido, pois o phpCodeGenie podegerar o código simples para os novatos onde tudo é feito no mesmo script PHP ou o código orientado a objeto da estrutura PHP, que segue o phpCodeGenie da estrutura de PCG que requer projetar suas tabelas e então o gênio pode gerar o código para a leitura e a escrita à base de dados, os formulários do HTML para incorporar ou editar dados à base de dados, scripts para listar dados, scripts para excluir dados, formulários de busca, scripts de busca entre outros. O phpCodeGenie foi projetado com o usuário iniciante em mente como também com o programador experiente em PHP que quer gerar uma aplicação complexa."

Infelizmente hoje o PCG é um projeto inativo, mas foi desenvolvido o bastante para ainda ser bastante útil. Os requisitos para instalação são básicos: servidor web, PHP e MySQL instalados. A última versão que temos conhecimento é a 3.0.2. Após, o responsável pelo projeto, Nilesh Dosooye, abandonou o mesmo.

Principais características:

- É um gerador do tipo passivo. Basta gerar o código e fazer as modificações necessárias. Ele pode ser livremente transferido para outros diretórios, apenas ajustando-se o path nos scripts common/footer.php e common/header.php.

- É opensource e licenciado sob GNU – GPL . Compatível com Linux e Windows .

- Não possui muita documentação.

- Só funciona com MySQL, embora aparentemente o autor tivesse a pretensão de dar suporte a outros SGBDs.

- O código pode ser gerado individualmente para uma tabela de cada vez ou para um banco inteiro.

- Quando da geração do CRUD simples, o PCG gera uma página (ou mais) para cada função. No caso do CRUD, o código gerado requer alguns ajustes, principalmente de natureza estética.

- O download e instalação apresentam alguma dificuldade. O acesso ao gerador se dá via browser.

- Não existe comunidade ativa para o projeto (pelo menos de nosso conhecimento).

- O Projeto está inativo, o que significa que você mesmo terá que se virar com bugs, etc.

- É bastante customizável, exigindo no entanto o mínimo de conhecimento de html e CSS, bons conhecimentos de linguagem PHP procedural ou OO.

Possui tradução para o português, por Ribamar FS.

Corrigindo um pequeno Bug

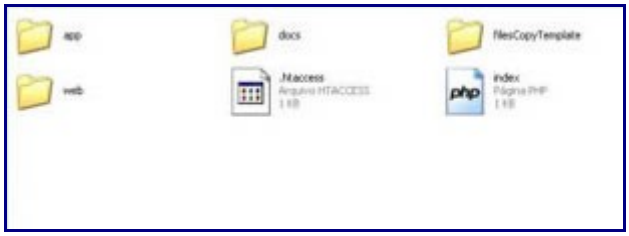
Edite o script common/footer.php e elimine o fechamento de tag:

Que é a primeira linha do script. Sempre terá que remover esta linha dos aplicativos gerados ou procurar a ocorrência no gerador e corrigir.

Instalação e utilização

Download Souce Forge:[1] No site do ribafs [2], você encontra um bom tutorial de instalação e o arquivo com a tradução para o português.

Após instalar e configurar, você deve ter a seguinte estrutura de pastas:



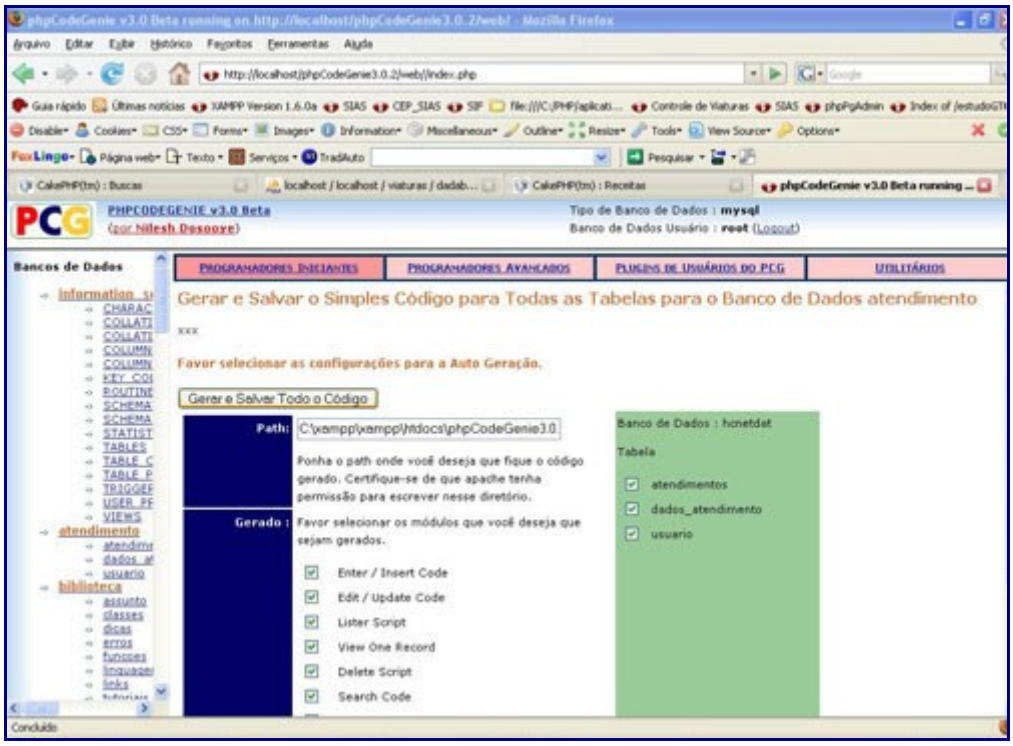
Os aplicativos gerados na opção programadores iniciantes, por exemplo, ficam na sub-pasta \web\generatedCode\simple\nome_do_aplicativo. Essa opção default, pode ser modificada.

Se não houve problemas, quando acessar o endereço do PCG via browser, você terá uma tela assim:

Preenchendo os dados para conexão, você é remetido a uma nova tela onde aparecem os bancos existentes no SGBD selecionado. Daí, é escolher a opção desejada e com alguns cliques, seu código está pronto. Você tem opções de gerar por exemplo todo o CRUD ou somente parte dele. Fazer isso utilizando linguagem procedural ou Orientação a Objeto etc.

Para gerar um aplicativo completo utiliza o PHP procedural e para gerar alguns utilitários utiliza o PHPOO.

Abaixo, tela mostrando as opções do PCG após a geração de um aplicativo.


PHPCODEGENIE v3.0 Beta
(por Nilesh Desouve)
Tipo de Banco de Dados : mysql
Banco de Dados Usuário : root (Logout)
Bancos de Dados
PROGRAMADORES INICIANTES
PROGRAMADORES AVANÇADOS
PLUGINS DE USUÁRIOS DO PCG
UTILITÁRIOS
Gerar e Salvar o Simples Código para Todas as Tabelas para o Banco de Dados atendimento
Favor selecionar as configurações para a Auto Geração.
Gerar e Salvar Todo o Código
Path:
C:\xampp\xampp\htdocs\phpCodeGenie3.0
Ponha o path onde você deseja que fique o código gerado. Certifique-se de que apache tenha permissão para escrever nesse diretório.
Gerado :
Favor selecionar os módulos que você deseja que sejam gerados.
Enter / Insert Code
Edit / Update Code
Lister Script
View One Record
Delete Script
Search Code
Banco de Dados : hcnetdat
Tabela
atendimentos
dados_atendimento
usuario
atendimento
biblioteca
Concluído

Geradores de Aplicativos/SQLMaestro (com MySQL, Oracle, MS SQL Server, PostgreSQL, SQLite, Firebird e MaxDB)

De Wikibooks

7.2 - SQLMaestro (MySQL, Oracle, MS SQL Server, PostgreSQL, SQLite, Firebird e MaxDB)

Geradores de aplicativos em PHP com os principais SGBDs do mercado.

Site oficial - http://www.sqlmaestro.com/download/

Os geradores para todos os SGBDs no site são free.

Basta fazer o download, instalar e usar.

Geradores de Aplicativos/phpMyEdit

De Wikibooks

7.3 - phpMyEdit

Em se tratando de geradores de código CRUD (acrônimo para Create, Retrieve, Update e Delete) para PHP, uma das melhores opções é o phpMyEdit. Como os desenvolvedores do projeto informam na página de abertura, o phpMyEdit é provavelmente a maneira mais rápida de gerar formulários para interagir com uma base de dados MySQL.

O phpMyEdit gera o código para edição de tabelas no MySQL com uma enorme gama de funções de manipulação (inserção, alterações, visualização, cópia, remoção de registros inseridos). Para isso você necessita apenas criar um script de chamada das funções (o phpMyEdit traz uma aplicação já pronta para desenvolver esse arquivo). Você pode ainda executar buscas com filtro, lookups etc. O gerador é open source e liberado sob licença GNU-GPL ou licença comercial. Ainda conforme os desenvolvedores, ele é utilizado em diversos sistemas ativos (há referências no site). Os requisitos são os básicos: servidor web (os desenvolvedores recomendam o Apache), PHP e MySQL instalados.

Principais características:

É um gerador do tipo ativo. Para que seu aplicativo funcione, no mínimo o arquivo phpMyEdit.class.php e os arquivos de linguagem devem estar presentes no seu diretório.

É opensource e licenciado sob GNU –GPL ou licença comercial.

Compatível com Linux, Windows e outros.

Possui boa documentação e tutoriais, disponiveis no site.

Só funciona com MySQL. O código é gerado individualmente para uma tabela de cada vez.

O download e instalação são bastante fáceis. O acesso ao gerador se dá via browser

Existe uma comunidade ativa com fóruns e listas de discussão que ajudam no desenvolvimento, dúvidas e outros.

O Projeto está ativo, o que significa melhorias, correções de eventuais bugs etc.

É bastante customizável, exigindo no entanto o mínimo de conhecimento de OO, html e java script.

Possui um arquivo de tradução para o português, exibindo botões, chamadas e mensagens em nossa língua. Esse arquivo pode ser modificado e merece algumas correções, porém mais de 90% do

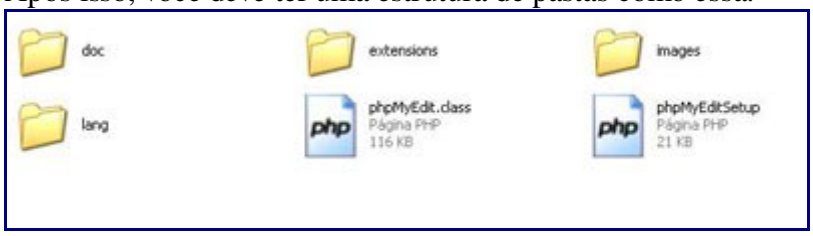
trabalho já está pronto.

Instalação e utilização

Faça o Download em

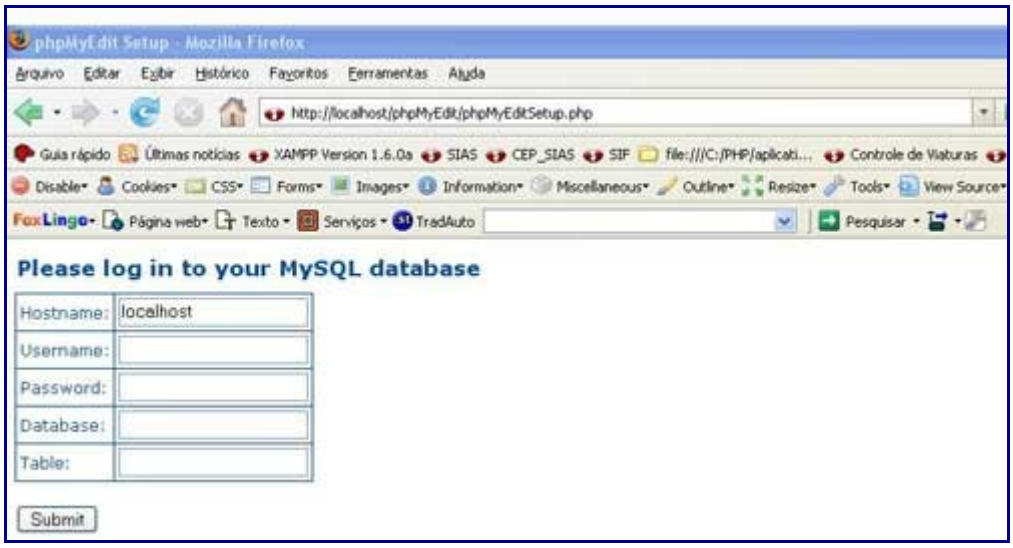
Descompacte o arquivo em seu diretório web (Se no Linux ou similar dê permissão de escrita no diretório do phpMyEdit).

Após isso, você deve ter uma estrutura de pastas como essa:

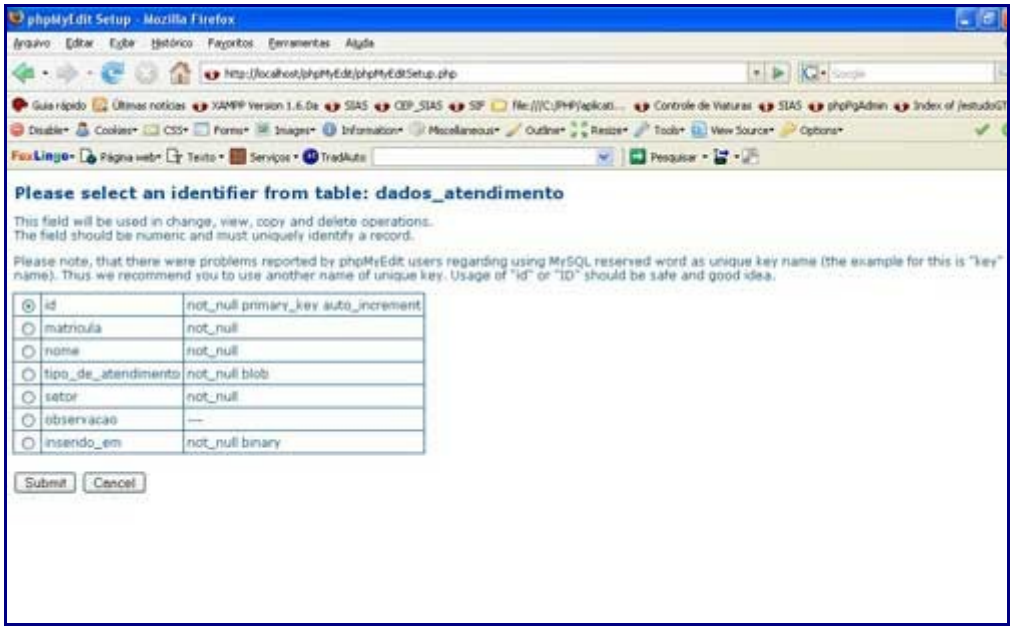


Agora, acesse pelo browser o arquivo phpMyEditSetup.php

Se tudo correu bem, você vai ter uma tela como esta:

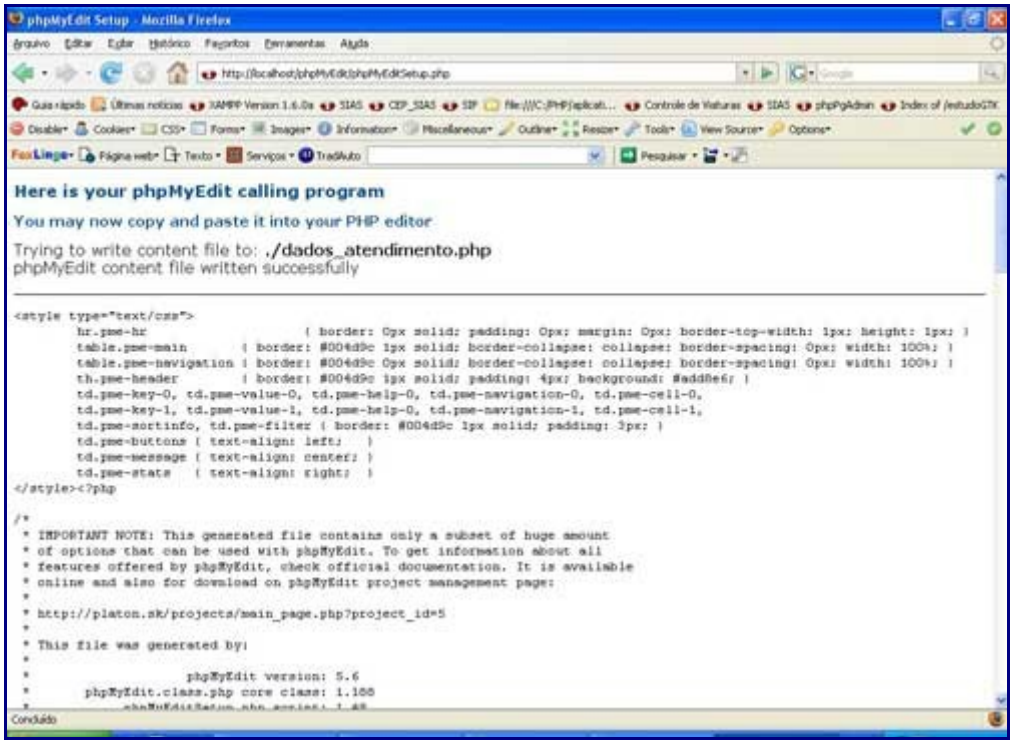


Preenchendo os dados relativos ao SGGB, o phpMyEdit realiza a conexão e já deve mostrar os dados da tabela. Veja que é recomendado ter um campo de identificação para ser utilizado como identificador nas operações (editar, adicionar etc). Recomenda-se também que este campo precisa ser numérico e único, a chave primária da tabela, sendo uma boa idéia nomeá-lo como "id". Dependendo da sua tabela, você pode criar um campo autoincrement. No exemplo, estou usando a tabela dados_atendimento.

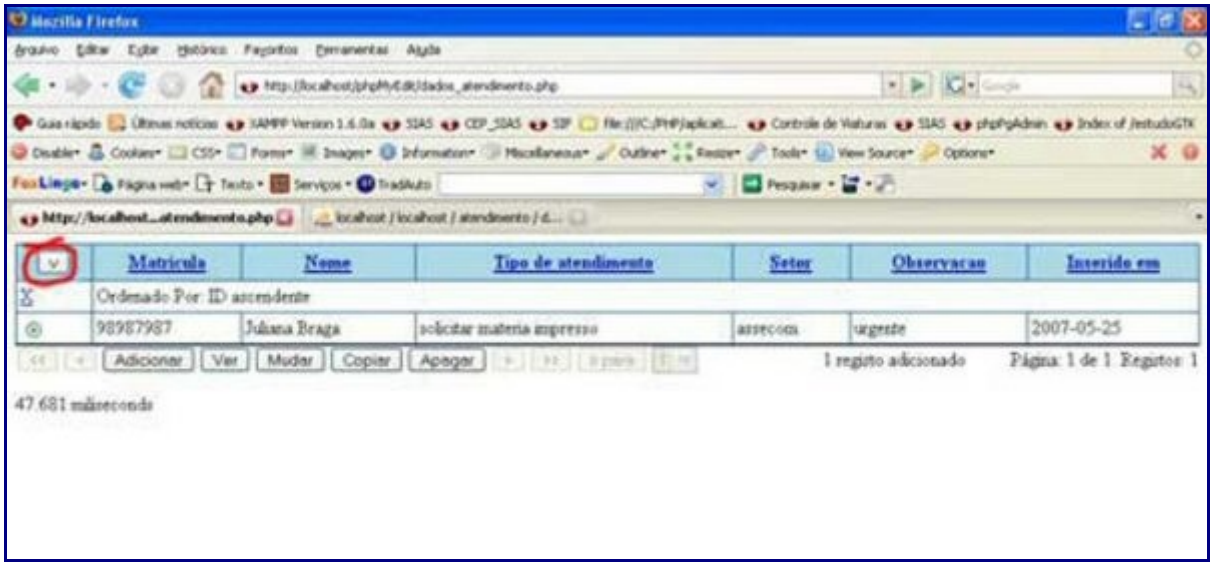


Em seguida, o phpMyEdit vai lhe apresentar a tela com opções adicionais. Você pode gerar um cabeçalho (título) para a página, o html de um cabeçalho e rodapé e uma folha de estilo(CSS básico). Tudo isso é gerado em um único arquivo. Por default, vem marcada a penas a opção da folha de estilo. Caso ache as opções do CSS limitadas,você pode editá-las acrescentando ou modificando, bem como utilizar outro arquivo de CSS, bastando vinculá-lo à página.

Bom, depois dessa série simples de cliques, você já tem seu aplicativo pronto para usar. Veja que o phpMyEdit salva o arquivo PHP do aplicativo em sua pasta base e com o nome da tabela. Ele também apresenta no browser o código gerado. Você pode copiá-lo e criar um novo arquivo em outtra pasta. Como já dissemos, nesse caso, você deve mover para essa mesma pasta pelo menos o arquivo phpMyEdit.class.php e a pasta lang. Se na configuração quiser usar imagens em vez dos tradicionais botões Javascript, mova também a pasta images. O phpMyEdit possui uma série de estensões como cal pop-up (calendário), messages etc. Caso queira usá-las, mova também a pasta extensions.



Essa é a interface básica do arquivo gerado. Note que os nomes dos campos são idênticos aos existentes no banco de dados. Com relação à linguagem, o phpMyEdit já roda o aplicativo com a configuração de linguagem padrão do browser (testei com o Mozilla Firefox 2 e o Internet Explorer 6 e 7 (ou seja se seu seu navegador está em português, é nessa linguagem que você verá seu aplicativo). O botão em destaque abre a opção de busca, onde você pode utilizar mais de um campo como filtro para as pesquisas.



Editando o arquivo criado, você pode alterar as opções para gerenciamento do CRUD, deixando por exemplo somente a visualização. Também pode alterar a posição dos botões colocando-os na parte superior ou inferior do aplicativo ou até em ambas, estipular a quantidade de registros a ser exibida em cada página, definir se utilizará imagens em vez de botões tradicionais ou em conjunto com eles, modificar o nome dos campos da tabela para exibição etc. Como já foi dito, é possível modifcar o CSS ou criar um novo. Também podemos configurar mensagens em javascript para, por exemplo, quando um campo da tabela for not null, avisar ao usuário que o mesmo deve ser preenchido. Todas essas opções são facilmente customizáveis. O arquivo gerado vem todo comentado, o que facilita muito. Como estamos usando PHP, obviamente também é podemos modificar a forma de exibição de datas, já que no padrão MySQL, elas vêm na forma ano- mês-dia. Enfim, o phpMyEdit é um ótimo gerador de código e editor de tabelas, repleto de possibilidades. No código abaixo, simplesmente suprimindo ou acrescentando letras, já modificamos as opções de exibição: no primeiro trecho em destaque, estamos permitindo somente adicionar, ver e modificar os registros. No segundo trecho, estamos inserido imagens dessas mesmas no menu de opções ao lado de cada registro.

```
46  // Number of records to display on the screen
47  // Value of -1 lists all records in a table
48  $opts['inc'] = 15;
49
50  // Options you wish to give the users
51  // A - add,  C - change, P - copy, V - view, D - delete,
52  // F - filter, I - initial sort suppressed
53  $opts['options'] = 'AVC';
54
55  // Number of lines to display on multiple selection filters
56  $opts['multiple'] = '4';
57
58  // Navigation style: B - buttons (default), T - text links, G - graphic links
59  // Buttons position: U - up, D - down (default)
60  $opts['navigation'] = 'DG';
61
62  // Display special page elements
63  $opts['display'] = array(
```

Links PhpMyEdit :

site:[1]

Documentação:[2]

Fórum:[3]

Geradores de Aplicativos/DadaBik

De Wikibooks

7.4 - DadaBik

Mais um gerador de aplicativos para o PHP.

Site oficial - http://www.dadabik.org/

Ele usa o AdoDB como abstração de bancos de dados e segundo o site já foi bem testado nos SGBDs MySQL, PostgreSQL, Oracle e MS SQL Server.

Segundo o autor o DadaBik foi criado para ser customizável, tanto que para cada campo de tabela você pode escolher:

```
   * Se o campo será incluído ou não no formulário search/insert/update e
resultados da tabela
   * Rótulos dos campos
   * Formatos dos campos (numérico, alfabético, e-mail, url......)
   * O tipo para cada campo do form (select, date, text, rich text editor,
password......)
   * Os possíveis valores, também vindos de outra tabela (foreign key support)
   * e mais...
```

Veja mais detalhes e recursos no site oficial.

Geradores de Aplicativos/PHPLibDev

De Wikibooks

7.5 - phpLibDev

Pequeno e experimental gerador de aplicativos que trabalha com PHP e PostgreSQL

É interessante observar as rotinas de metadados, que pegam informações diretamente do SGBD.

Existem duas versões: uma com funções do PHP acessanfo o PostgreSQL e outra usando a abstração PEAR/DB.

Projeto no Sourceforge - http://phplibdev.sourceforge.net/

Geradores de Aplicativos/Web Form Generator

De Wikibooks

7.7 - Web Form Generator

Gerador de Aplicativos para PHP com MySQL

Site oficial. Requer JRE 1.5 ou superior (http://developers.sun.com/downloads/top.jsp).

O Gerador é for Windows. Um stand alone em Java.

Muito flexível, pois permite, antes de gerar:

- Adicionar filtro para os campos (busca)

- editar os rótulos de cada campo

- alterar o tipo de controle dos campos do form (text, textarea, list, combo, radio, etc)

- alterar o tamanho dos campos (length) e o maxlength,

- adicionar validação como texto, e-mail ou número com mensagem de erro

- requeridos ou não

- marcar que campos aparecerão em cada tela: I (incluir), U (atualizar), D (excluir) e L (listar)

Traz recursos de paginação de resultados e estilo.

A lista de features no site

CMSs (Gerenciadores de Conteúdo)/Joomla (Portais)

Conteúdo

- 1 Instalação
  - 1.1 Requisitos do Joomla
  - 1.2 Licença
  - 1.3 Configurações do Banco
  - 1.4 Nome do Portal
  - 1.5 Acessar Site ou Administração
  - 1.6 Portal Joomla Instalado
  - 1.7 Portal Joomla (Frontend, ou seja, o site)
- 2 Administração do Joomla
  - 2.1 Verificar lista de itens mais populares do site
  - 2.2 Configurações do Site
  - 2.3 Módulos
  - 2.4 Seções e Categorias
  - 2.5 Imagem
  - 2.6 Banners
  - 2.7 Enquete (Pools)
  - 2.8 Contato
  - 2.9 Mensagens entre Usuários
  - 2.10 Enviar E-mail em Massa
  - 2.11 Usuários
  - 2.12 Estatísticas
  - 2.13 Instalar e Desinstalar
- 3 Extensões Favoritas
  - 3.1 Componentes
    - 3.1.1 JoomlaXplore (Gerenciador de Arquivos)
    - 3.1.2 PUArcade (jogos online)
    - 3.1.3 Fun Games (jogos online)
    - 3.1.4 EasyBook (Livro de Visitas)
    - 3.1.5 RSGallery2 (Galeria de Imagens)
    - 3.1.6 Biorritmo
    - 3.1.7 JomComment (Comentários nas Seções)
    - 3.1.8 MamboWiki (Adiciona mediawiki no Joomla)
    - 3.1.9 Joomap (Mapa do Site)
  - 3.2 Módulos
    - 3.2.1 Bible Verse of The Day (VOTD)
    - 3.2.2 javascript module
  - 3.3 Mambots
    - 3.3.1 JoomlaFCK
    - 3.3.2 JoomlaTinyFCK (JTF)
  - 3.4 Templates
    - 3.4.1 Site com 1500 templates
  - 3.5 Language
    - 3.5.1 Português do Brasil
- 4 Referências

Instalação

Instalação do CMS Joomla 1.0.12

Download do Site Oficial - http://joomlacode.org/gf/project/joomla/frs/

Descompacte no seu DocumentRoot numa pasta, por exemplo, joomla1012 ou portal.

Então iniciar o apache e abrir o navegador com - http://localhost/joomla1012

Veja que será redirecionado para - http://127.0.0.1/joomla1012/installation/index.php

Requisitos do Joomla

O Joomla requer PHP e MySQL além de algumas configurações nesse sistema.

Observe as configurações e se seu sistema está pronto para a versão do Joomla que está instalando.

Em sistemas linux, caso não tenha criado o arquivo configuration.php e dado permissão de escrita, receberá o aviso:

```
" configuration.php         Unwriteable
You can still continue the install as the configuration will be displayed at the
end, just copy & paste this and upload."
```

Neste caso pode seguir em frente, mas fique bastante atento para

- Quando chegar ao final e o Joomla exibir o código para o configuration.php,

- Então copie o código,

- Crie o script configuration.php

- Abra o script e cole o código copiado

Caso seu register_globals esteja como "On" no php.ini, você será alertado sobre isso. Se puder altere para "Off".

Caso receba este aviso:

"Joomla! RG_EMULATION setting is `ON` instead of `OFF` in file globals.php `ON` by default for compatibility reasons"

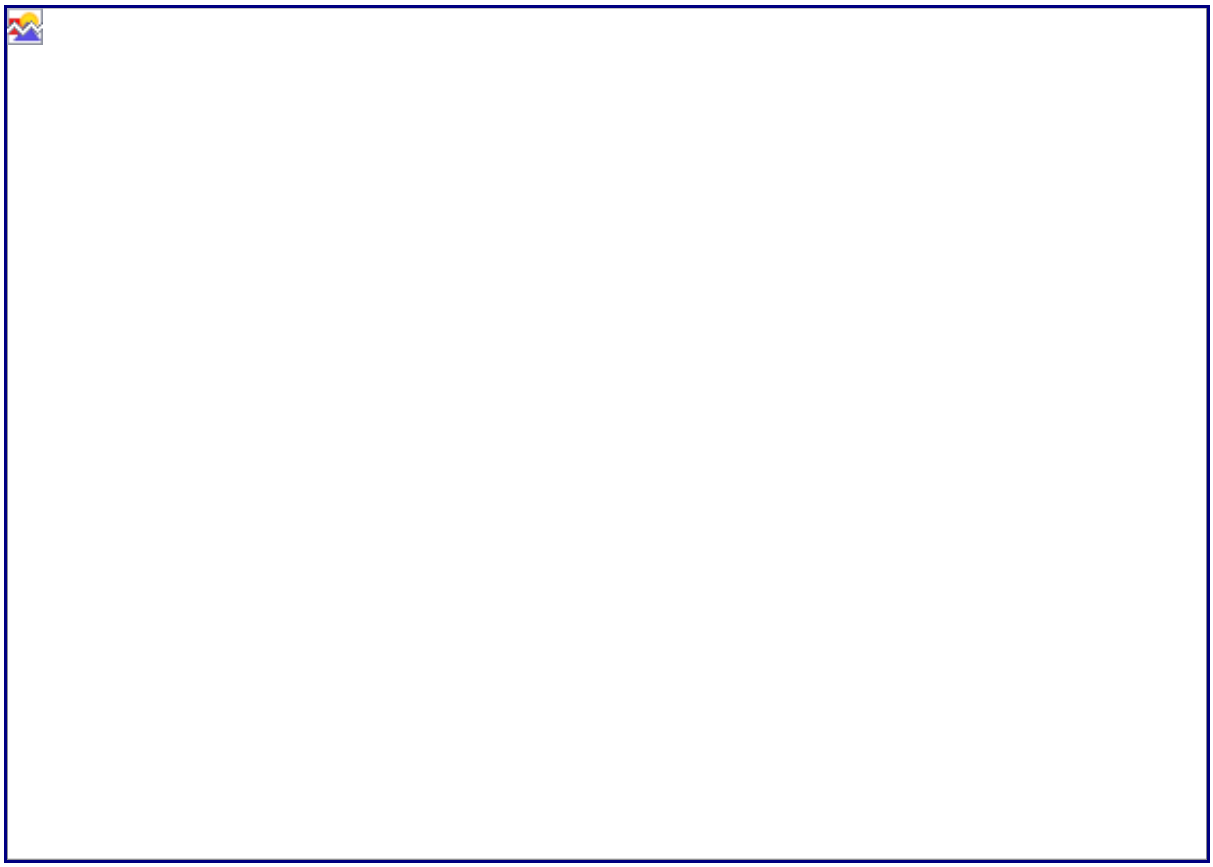
Deverá alterar o arquivo globals.php, que se encontra no raiz do Joomla, mudando a linha:

define('RG_EMULATION', 1);

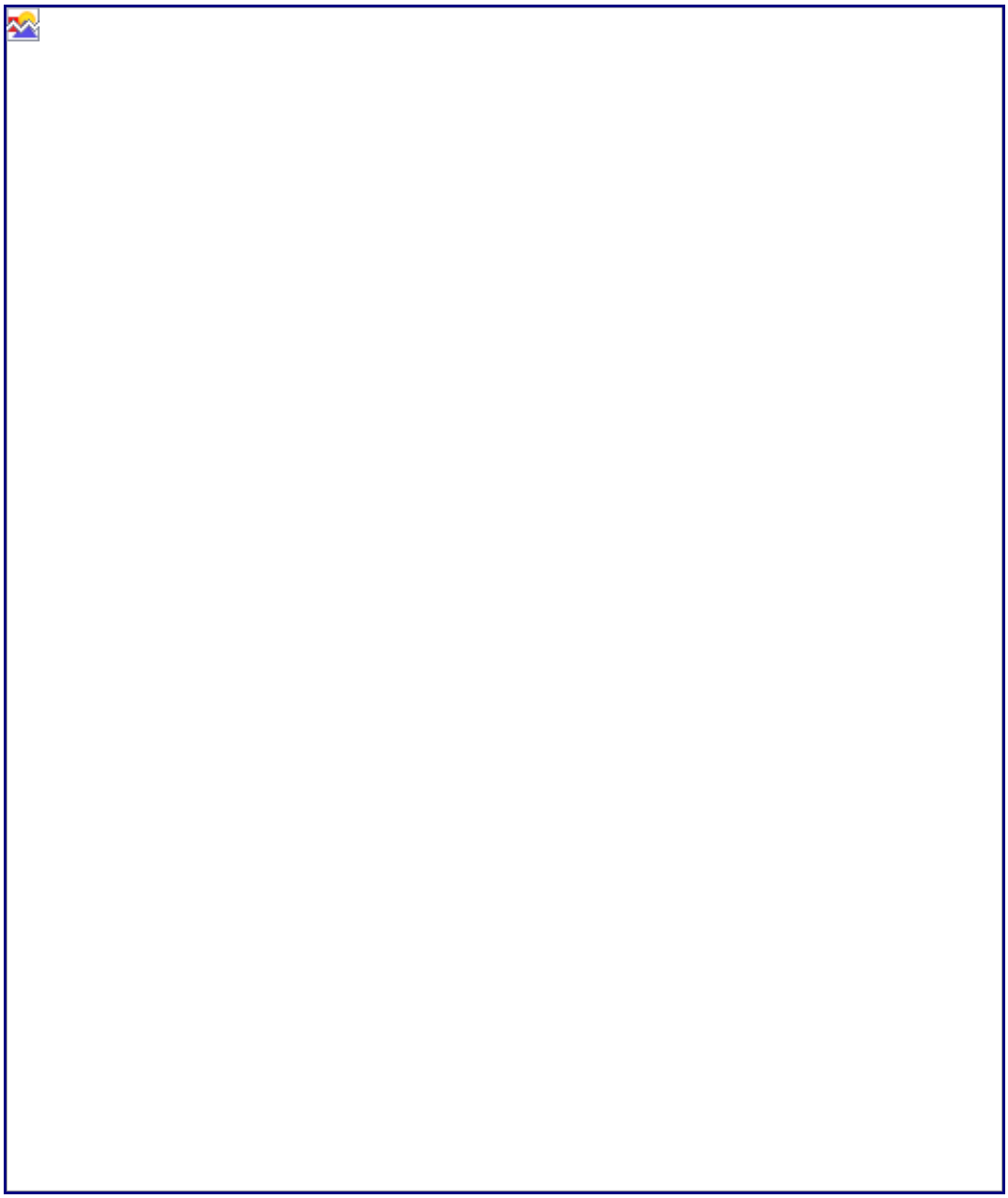
Alterando de 1 para 0.

A equipe do Joomla sabendo que muitos administradores de sites não tem acesso ao php.ini criou um emulador do mesmo, onde podemos emular o register_globals setando para Off.

Veja também as permissões de vários diretórios que requerem permissão de escrita. Caso esteja usando Linux altere as permissões desses diretórios para 777 (sudo chmod 777 nomediretorio).



Ao chegar nessa tela, altere as permissões e altere o globals.php.



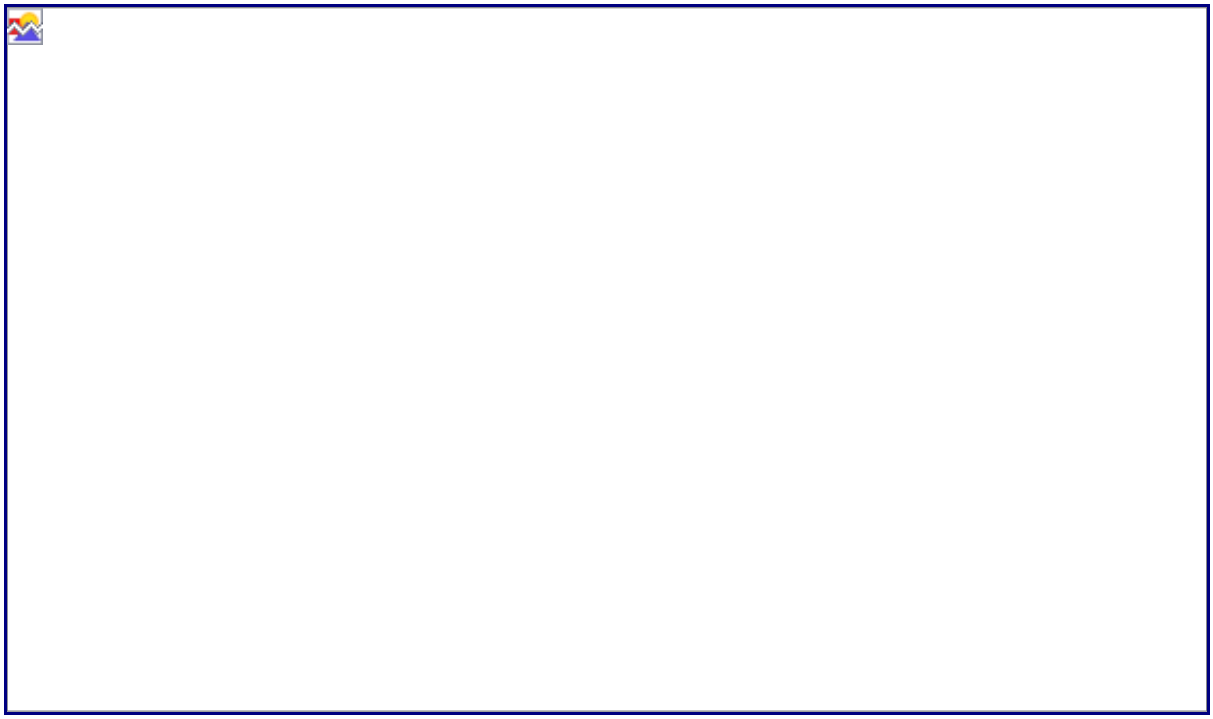
Então tecle F5 ou clique em Check Again para atualizar.

Somente quando não tiver mais pendências siga em frente. Algumas pendências não impedem de seguir em frente.

Então clique em Next.

Licença

Receberá então a tela da Licença. Verifique o conteúdo ou simplesmente clique em Next.



Configurações do Banco

Aqui entre com os dados do bancode dados. Veja como exemplo os da imagem abaixo. Somnete no caso de já existir o banco e queira sobrescrever o existente, então marque a caixa "Drop Existing Tables".

Então clique em Next e confirme.

Nome do Portal

Entre com o nome do site e clique em Next

Acessar Site ou Administração

Apenas entre com seu e-mail e a senha para o usuario administrador. É recomendável trocar a senha default sob pena de você não lembrar da mesma, que é uma senha aleatória. Lembre que com isso irá acessar a seção de administração do site com login "admin" e esta senha.

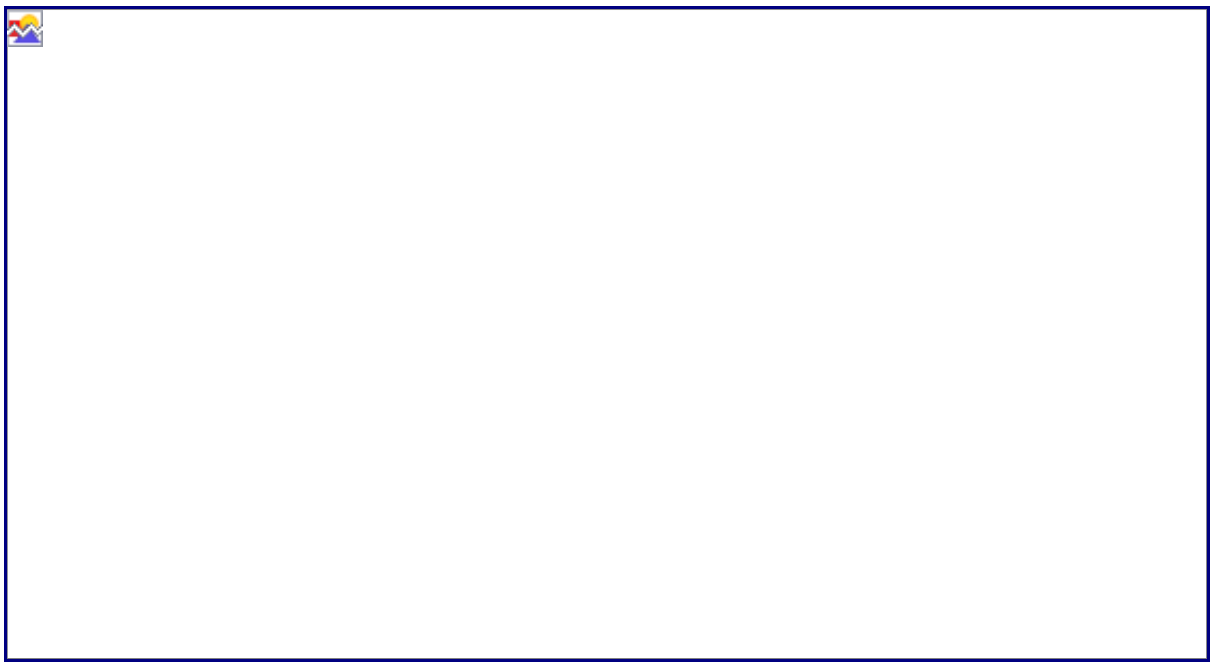
Assim:

http://localhost/joomla1012/administrator

Após fazer o login altere o nome do usuário administrador.

Clique em Next.

Portal Joomla Instalado

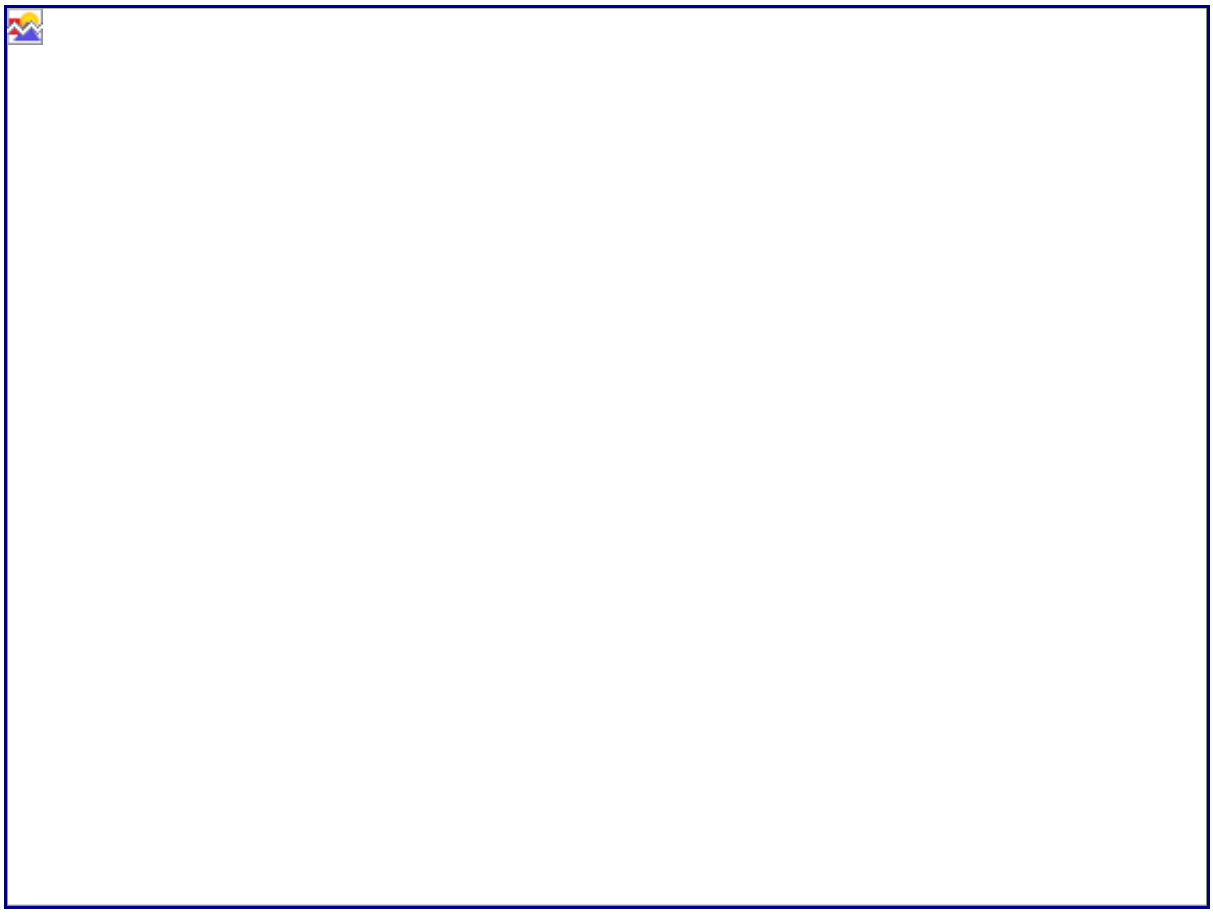


Aqui é obrigatório (por segurança) que remova o diretório "installation" para poder acessar o site.

Nesta fase também, se no Linux, quando não deu permissão de escrita ao configuration.php, aparece o código para esse script. É quando deve copiar e colar no arquivo que deve criar (configuration.php).

Então acesse o site clicando em View Site ou a seção de administração clicando em Administration.

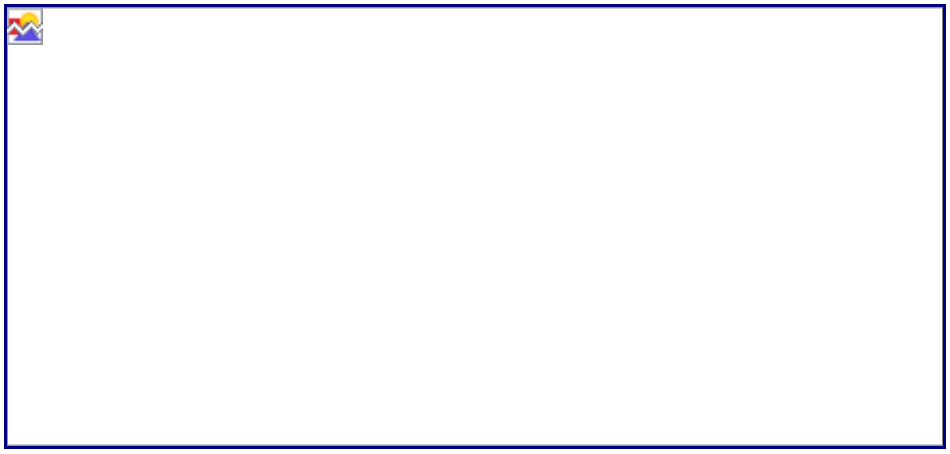
Portal Joomla (Frontend, ou seja, o site)



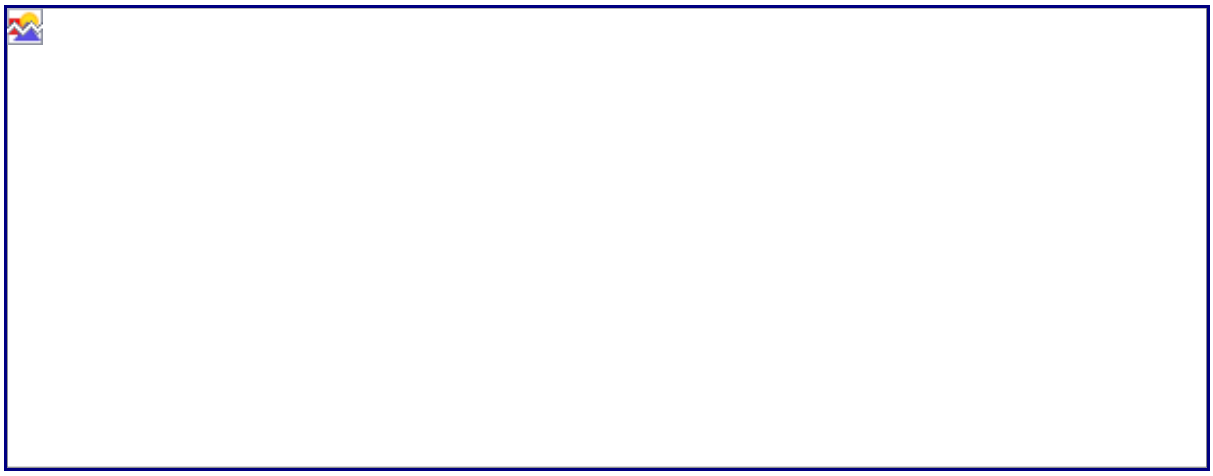
Administração do Joomla

A adminsitração do Joomla é feita num outro site, http://localhost/joomla1012/admin

Login dos Administradores para backend



Painel de Controle da seção de administração do Joomla



Alguns recursos da área administrativa

- Usuários logados no portal

Veja que após o login como usuário super administrador você verá quais usuários estão logados neste momento e poderá inclusive derrubar (efetuar logout) algum usuário, se desejar, apenas clicando no X vermelho à direita do usuário, estando na aba Logado.

Verificar lista de itens mais populares do site

Na aba Popular aparecem as seções mais visitadas do site. Veja que existem outras abas que são atalhos para as principais funções encontradas nos menus e ícones.

Configurações do Site

Através do menu Site - Global Configuration podemos alterar diversas configurações do portal. Somente o super administrador poderá fazer essas alterações. Somente o usuário superadministrador tem acesso à esse item.

Aqui podemos trocr o editor HTML default

Aqui podemos trocr o nome do Site

Aqui podemos deixar o site offline (para manutenção)

Aqui podemos trocr o e-mail do contato configurado em Contato

Aqui podemos habilitar as estatísticas, para poder habilitar o contador (módulo Statistics).

E muitas outras configurações úteis.

Obs.: Alterações nesse item requerem permissão de escrita no arquivo configuration.php (do raiz do

Joomla). Em termos de segurança é bom liberar quando for alterar e bloquear após as alterações. Para isso marque o checkbox "Override write protection while saving" antes de salvar.

Módulos

Após instalar o Joomla e instalar o idioma português do Brasil, então devemos fazer várias alterações manualmente: nomes dos módulos, nomes de menus, etc.

Ao editar um módulos podemos alterar a posição em que ele ficará na página: direita, esquerda, após qual módulo existente, etc. Também podemos configurar o acesso ao mesmo: público, registrado e especial (somente aqueles com maiores privilégios). Podemos e devemos alterar o título além de outras propriedades.

Seções e Categorias

Recursos muito úteis. Com eles podemos deixar a índex do site bem enxuta: somente com seções, que abrirão categorias, que abrirão itens de conteúdo.

```
seção M
    categoria A
              Item de Conteúdo 1
              Item de Conteúdo 2
              Item de Conteúdo 3
    categoria B
              Item de Conteúdo 1
              Item de Conteúdo 2
    categoria C
              Item de Conteúdo 1
Seção N
    categoria J
              Item de Conteúdo 1
              Item de Conteúdo 2
              Item de Conteúdo 3
    categoria K
              Item de Conteúdo 1
              Item de Conteúdo 2
    categoria L
              Item de Conteúdo 1
```

Imagem

Existe um recurso muito útil para enviar imagens por upload e outras mídias e também as administrar, que é o Midia Manager.

Sempre que quiser enviar uma imagem abra o Midia Manager, clique em Arquivo, selecione o arquivo

e então clique em Enviar. Também podemos criar diretórios e mudar de diretórios.

Após enviar um arquivo para lá então podemos usar o editor HTML para adicionar a imagem a um artigo, por exemplo.

Banners

Sempre fique atento para o tamanho que se adapte corretamente ao template em que vai usar o banner.

Veja que pode zerar o contador de cliques para banner existente.

Pode controlar o período em que o banner será exibido.

Pode cadastrar o cliente e outras. O banner é um componente.

Enquete (Pools)

Personalize a enquete atual. Também pode adicionar tantas quantas queira. Uma enquete é um componente.

Contato

É importante configurar o contato logo que mande o site para o ar, adicionando os dados corretos do contato, e-mail, endereço (se for o caso), etc.

Mensagens entre Usuários

Recurso especial do Joomla, especialmente quando o site estiver sendo construído ou mantido por mais de um usuário. Neste caso os usuários podem trocar mensagens entre si. A área de mensagens, onde pode visualizar se tem ou não mensagem para você fica acima e à direita: um ícone de um envelope amarelo com uma seta curva e verde claro abaixo dele. Quando existir mensagem nova para você aparecerá o número de mensagens novas à esquerda do ícone. Ao clicar no ícone poderá ler as mensagens, responder, enviar nova ou excluir as lidas, além da opção de busca existente.

Enviar E-mail em Massa

Enviar um e-mail para todos os usuários registrados ou para grupos de usuários. Somente o super administrador tem este privilégio.

Usuários

O Joomla tem dois grupos de usuários, os do frontend (que tem acesso ao site) e os do backend (que tem acesso ao site administrativo).

```
Frontend
  '''public''' (tem privilégio apenas de ver conteúdo liberado como públic)
    '''registered''' (estes visualizam tudo que é public e também aqueles
conteúdos para usuários registrados. Quando logado pode enviar links)
      '''author''' (estes tem os privilégios dos anteriores e também tem o de
enviar notícias e editar as próprias notícias)
        '''editor''' (privilégios dos anteriores mais os de editar conteúdo de
todos do seu grupo)
          '''publisher''' (enviar, editar e publicar qualquer conteúdo no
frontend. Veja nenhum dos anteriores pode publicar seu conteúdo).

Backend
  public
    manager
      administrator
        super administrator
```

Estatísticas

Caso queira habilitar o contador de visitas, deverá ir em no menu Site - Global Configuration, aba Statistics e habilitar o item Statistics. Somente o super administrador tem este privilégio.

Então edite o módulo Statistics, no menu Modules - Site Modules. Então ative apenas o Contador.

Instalar e Desinstalar

Instalar e desinstalar extensões no Joomla é algo extremamente prático, usa-se apenas o mouse instalando os arquivos compactados.

Atente apenas para os tipos de extensões:

- **módulos** (estes ao serem instalados e publicados aparecem automaticamente no site. ao serem instalados estão despublicadso). Os módulos ao serem instalados permanecem despublicados.

- **componentes** normalmente requerem a criação de um item de menu que os clame ou de um link.

- **template** - transformam a aparência do site por inteiro. Após instalado para alterar o template do site vá no menu Site - Template manager - Site Tempaltes.

- **mambot** - geralmente são editores HTML. Após instalado vá no menu Site - Global Configuration e Default WYSIWYG Editor altere o editor. Somente o super administrador tem este privilégio

- **language** - instalar novo idioma.

Existem muitos outros recursos e a cada versão nova aparecem ainda outros novos recursos.

Estude, pesquise e compartilhe suas descobertas com outros por aqui.

Extensões Favoritas

Componentes

JoomlaXplore (Gerenciador de Arquivos)

Gerenciador de arquivos com muitos recursos úteis.

http://joomlacode.org/gf/project/joomlaxplorer/frs/

PUArcade (jogos online)

Jogos Clássicos: snake, asteroids, siga-me, space invader, pac-man, tetris e frogger.

http://www.pragmaticutopia.com/

Fun Games (jogos online)

407 Games free dos mais variados tipos.

http://www.freegaming.de/upload/com_fungames.zip

EasyBook (Livro de Visitas)

Bom livro de visitas com controle de spam.

http://joomlacode.org/gf/project/easyjoomla/frs/?action=FrsReleaseBrowse&frs_package_id=225

RSGallery2 (Galeria de Imagens)

Ótima galeria de imagens em português (quase tudo), com opção para envio de arquivos compactados, com galeria de imagens aleatórias, etc.

http://rsgallery2.net/

Biorritmo

Exibe gráfigo com biorritmo.

http://www.alikonweb.it/451/

JomComment (Comentários nas Seções)

Ótimo componente que permite adição de comentários nas seções. Veja uma amostra aqui

http://www.azrul.com/products/joomla_comment_system.html

MamboWiki (Adiciona mediawiki no Joomla)

Insere de fato o mediawiki no Joomla. Muito bom.

http://www.lyquidity.com/mambo/index.php?option=com_download&Itemid=180

Joomap (Mapa do Site)

Simples mapa de site.

http://www.koder.de/joomap.html

Módulos

Bible Verse of The Day (VOTD)

Exibe um versículo da Bíblia a cada entrada no site.

http://downloads.korske.com/files/mod_votd.zip

Bíblia Completa Online em PHP e MySQL

Existe uma bíblia completa (antigo e novo testamento) em PHP com MySQL que pode ser instalada e usada confortavelmente via wrapper no Joomla.

Veja aqui: http://br.groups.yahoo.com/group/bibliaonlinephp/

Cadastre-se gratuitamente na lista para fazer o download.

javascript module

Este módulo serve para adicionar código em JavaScript ou HTML, como por exemplo o AdSense do Google.

http://www.joomlastop.com/content/view/28/47/

Mambots

JoomlaFCK

Editor baseado no FCKEditor.

http://joomlacode.org/gf/project/joomlafck/frs/

JoomlaTinyFCK (JTF)

Editor tendo como base também o FCK.

http://www.hackjoom.web.id/JoomlaTinyFCK-mambot.html

Obs.: O editor default, que vem junto ao Joomla é muito bom.

Templates

Site com 1500 templates

http://www.joomlaos.de/joomla_template_galerie.html

Language

Português do Brasil

Para Joomla 1.0.12

http://www.geraldosales.com.br/portal/joomla/brazilian1.0.12.01.zip

Referências

Site Oficial do Joomla - http://www.joomla.org

Site com grande catálogo de extensões para Joomla - http://extensions.joomla.org/

CMSs (Gerenciadores de Conteúdo)/Mambo (Portais)

De Wikibooks

8.2 - Mambo

O Mambo já foi o CMS mais premiado. Criado pela empresa Miro e aberto à comunidade com a licença GPL. Houve desentendimento entre a empresa e a comunidade fazendo surgir há pouco mais de um ano o Joomla.

Site para download - http://www.mamboserver.com/

Tutoriais em Português

- http://mambobrasil.org/

- http://ribafs.net/down/tut/

CMSs (Gerenciadores de Conteúdo)/Drupal (Portais)

De Wikibooks

Instalação

- Crie um banco de dados "drupal"
- Importe o script da pasta database
- Configure o script da pasta sites/default para os dados do SGBD
- Abra no browser com http://127.0.0.1/drupal-4.7.3/
- Então receberá:

Welcome to your new Drupal website!

Please follow these steps to set up and start using your website:

1.Create your administrator account To begin, create the first account. This account will have full administration rights and will allow you to configure your website.

2.Configure your website Once logged in, visit the administration section, where you can customize and configure all aspects of your website.

3.Enable additional functionality Next, visit the module list and enable features which suit your specific needs. You can find additional modules in the Drupal modules download section.

4.Customize your website design To change the "look and feel" of your website, visit the themes section. You may choose from one of the included themes or download additional themes from the Drupal themes download section.

5.Start posting content Finally, you can create content for your website. This message will disappear once you have published your first post.

For more information, please refer to the Help section, or the online Drupal handbooks. You may also post at the Drupal forum, or view the wide range of other support options available."

Dica - clique com o botão direito sobre o primeiro link para abrir numa outra janela e ter os links originais sempre à disposição.

Após criar o user administrador:

- Clique em administer
- Depois em settings para configurar o CMS. Verifique e altere todas que julgar importantes.
- Então clique em modules para ver a lista de módulos ativos e ativar/desativar os desejados. Habilite o módulo locale para mudar para português do Brasil, por exemplo.
- Baixe o arquivo de tradução do site oficial e:
    - administer - localization - manage languages - import. Selecione o arquivo e em import into selecione Português.
- Altere a default para Português

- Módulos adicionais - http://drupal.org/project/modules
- Customizando o tema do site - http://127.0.0.1/drupal-4.7.3/?q=admin/themes
- Selecionando um tema - http://drupal.org/project/themes
- Começar a adicionar conteúdo - http://127.0.0.1/drupal-4.7.3/?q=node/add
- Ajuda - http://127.0.0.1/drupal-4.7.3/?q=admin/help, http://drupal.org/handbooks, http://drupal.org/forum, http://drupal.org/support, etc.
- Criando Conteúdo http://127.0.0.1/drupal-4.7.3/?q=node/add

Ativando/Desativando módulos

http://127.0.0.1/drupal-4.7.3/?q=admin/modules

Algumas sugestões.

- Verifique as respectivas explicações na seção para ver se concorda:

- aggregator, archiver, blog, blogapi, book, forum, pool, search, statistcs, tracker (ativar)

Caso tenha marcado os itens acima e acesse novamente: http://127.0.0.1/drupal-4.7.3/?q=node/add

agora verá algo como:

Principal

criar conteúdo

Escolha o ítem apropriado na lista:

enquete

Enquete é uma pesquisa de opinião de múltipla escolha onde os usuários podem votar em uma das opções apresentadas.

entrada de blog

Um blog é um diário ou jornal atualizado regularmente feito por contribuições individuais mostradas em ordem cronológica reversa. Um blog está estreitamente ligado ao autor, portanto cada usuário terá seu "próprio" blog.

matéria

Matérias são artigos em sua forma simples: elas possuem um título, uma chamada e um corpo, mas podem ser extendidas por outros módulos. O teaser é parte do corpo também. Matérias podem ser usadas como um blog pessoal ou para artigos de notícias.

página

Se você deseja adicionar uma página estática, como uma página de contato ou uma página "sobre", use a página.

página de livro

Um livro é um esforço de criação coletiva: os usuários podem contribuir escrevendo páginas para o livro, posicionando as páginas na ordem correta e revisando e modificando páginas previamente escritas. Assim, quando você tem alguma informação para compartilhar, quando leu uma página do livro que não gostou ou se achou que uma página poderia ser melhor escrita você pode fazer algo para mudar as coisas.

tópico de fórum

Criar um novo tópico para discussão nos fóruns.

Edita cada seção e crie seu portal.

Para Instalar com postgresql

Criar Usuário

create role drupal with nocreateuser nocreaterole nocreatedb encrypted password 'drupal';

Criar banco:

create database drupal with encoding 'unicode' owner drupal;

Dica: Caso esteja instalando em micro sem servidor de e-mail:

Execute este código:

```
print md5('seulogin'); //Exibirá o hash md5
```

Então abra o phpmyadmin com o banco do drupal, selecione a tabela user e visualize os registros edite o registro do seu login e substitua o hash md5

CMSs (Gerenciadores de Conteúdo)/Xoops (Portais)

De Wikibooks

CMS para criação de Portais

O Xoops é um dos CMS mais populares.

Site oficial - http://www.xoops.org

Download - http://www.xoops.org/modules/core/

Instalação - http://xoopsdocs.net/modules/docs/en/xu-002/online/index.html

A instalação é semelhante a de outros CMS.

Documentação - http://xoopsdocs.net/modules/docs/

CMSs (Gerenciadores de Conteúdo)/MediaWiki (wikis)

De Wikibooks

O Mediawiki é o CMS que deu origem à Wikipedia, ao Wikibooks e a toda essa geração de CMS que trouxe a democracia ao conhecimento.

Conteúdo

- 1 Instalação do Mediawiki
- 2 Configurações de pós-instalação
- 3 Usando o MediaWiki
  - 3.1 Formatação de Texto
- 4 Criando Índice em HTML
- 5 Inserindo Imagens e Outros arquivos
  - 5.1 Outras opções para inserir imagem
- 6 Criando um Índice com itens e subitens
- 7 Esboço de página
- 8 Evitar Interpretação do Mediawiki
- 9 Seções
- 10 Comunicação entre Usuários registrados
- 11 Código Fonte
- 12 Wiki: ser ou não ser?
- 13 Extensão para Firefox da MediaWiki
- 14 Referências

Instalação do Mediawiki

Instalação no Linux Ubuntu (é semelhante a uma instalação no Windows ou em outro Linux).

Compatibilidade

Lembrando que o Mediawiki atual (1.10.0) funciona somente com o PHP5 ou superior. Caso precise instalar em algum servidor com o cPanel precisará de uma versão para PHP 4.3.x ou superior, no caso a versão 1.6.10 ou outra compatível.

- Baixar de http://www.mediawiki.org

- Fazer o download para o documentRoot no diretório "mediawiki"

- Dar permissão de escrita ao diretório config (chmod a+w config)

- Criar o usuário e o banco, tendo o usuário privilégios sobre o banco.

- Chamar no browser http://localhost/mediawiki

- Preencha devidamente os dados, atentando para os requeridos.

- Após confirmar deverá mover o arquivo config/LocalSettings.php para o raiz do mediawiki.

- Então acesse http://127.0.0.1/mediawiki/index.php

Para acessar como administrador clique em Entrar e faça o login.

Configurações de pós-instalação

Alterando o logotipo

O logotipo que aparece no canto superior esquerdo poderá ser alterado. Para isso, no arquivo LocalSettings.php, insira ou altere o valor da linha $wgLogo.

Exemplo:

$wgLogo = "/wiki/eriberto.png"

Lembre-se de fazer upload do novo arquivo LocalSettings.php e da imagem referente ao logotipo.

Alterando o rodapé

Para alterar o rodapé do Wiki (retirando o Disclaimer, por exemplo), edite o arquivo skins/MonoBook.php e remova as opções indesejadas constantes na linha $footerlinks. As opções existentes nesta linha podem ser vistas no FAQ da MediaWiki (veja How do I change the footer?).

Alterando o fuso horário

Para que o Wiki utilize o mesmo fuso horário do usuário, no momento de uma edição, adicione no final do arquivo LocalSettings.php, antes de ?>, a seguinte linha:

$wgLocalTZoffset = date("Z") / 60;

Alterando as mensagens de tela e menus

É possível alterar as mensagens que aparecem na tela e os nomes que aparecem nos menus. Para isso, edite o arquivo languages/messages/MessagesPt_br.php.

Permitindo o upload de imagens

Para permitir que as pessoas que editam o wiki façam o upload de arquivos, altere o arquivo LocalSettings.php e altere a linha:

$wgEnableUploads = false

para

$wgEnableUploads = true

Usando o MediaWiki

Para texto puro é semelhante aos editores de texto comuns e aos editores HTML, com exceção da quebra de linha que aqui é efetuada teclando-se duas vezes no Enter ao invés de uma.

Formatação de Texto

Negrito - Apenas selecione o texto e clique no botão B na barra do editor acima.

Alternativamente selecione o texto e insira "' antes e após o texto selecionado (o mesmo efeito de clicar no **B**).

Itálico - Proceder de forma semelhante clicando no ***I***.

Título

Para formatar um texto como título clique no botão **Texto a negrito**A da barra do editor acima.

Indentação

Para indentar usa-se dois pontos ":"

```
:Primeiro nível
::Segundo
:::Terceiro
```

Aparecerá assim:

Primeiro nível
Segundo
Terceiro

Indentar com Pontos

```
*Primeiro nível
**Segundo
***Terceiro
```

Listas Numeradas

Listas Numeradas

```
#Primeiro

#Segundo

##SubdoSegundo

#Terceiro
```

Aparecerá:

1. Primeiro
2. Segundo
    1. SubdoSegundo
3. Terceiro

Quebra de Linha

Podemos usar a quebra de linha do HTML <br>.

Assim como podemos utilizar vários comandos do HTML, mas nem todos.

Caso se digite:

```
Este parágrafo está
em três
linhas.
```

O Mediawiki (Wikibooks) irá exibir o seguinte parágrafo:

Este parágrafo está em três linhas.

Para que seja de fato exibido em três linhas deverá inserir sempre uma linha em branco entre cada uma delas, assim:

Este parágrafo está

em três

linhas.

Ou usando <br>, assim:

```
Este parágrafo está<br>
em três<br>
linhas.<br>
```

Que aparecerá assim:

Este parágrafo está
em três
linhas.

Criando Índice em HTML

Caso queiramos criar automaticamente um índice da página que estamos criando, basta usar a tag título <H1>, <H2> ... <H6>. Edite esta página ou outra com este recurso e confira.

Somente funciona se a página contiver um título usado com o ícone A, ou seja ==Título==.

Link

Insira link selecionando a URL e clicando no botão com o globo da barra.

Links para Páginas Externas

[http://pt.wikibooks.org/ WikiBooks]

Abre colchete, URL, Descrição, espaço em branco e fecha colchete.

Se apenas digitarmos http://pt.wikibooks.org/ também aparecerá o link e o rótulo será a própria URL.

Criando Nova Página

- Suponha que esteja em http://pt.wikibooks.org/wiki/Aplicativos_em_PHP/Geradores_de_Aplicativos e queira criar uma nova página para um outro gerador ainda não conttemplado

- Apenas faça login (de preferência) e abra no navegador a seguinte URL:

http://pt.wikibooks.org/wiki/Aplicativos_em_PHP/Geradores_de_Aplicativos/MeuGerador

Irá receber a mensagem "(Não há atualmente nenhum texto nesta página)". Então apenas clique em Editar e insira o conteúdo para o gerador.

Inserindo Imagens e Outros arquivos

A lista de arquivos permitidos no Wikibooks é somente esta:

png, gif, jpg, jpeg, xcf, xls, pdf, mid, sxw, sxi, sxc, sxd, ogg, svg, djvu

```
- Clique no ícone para inserir imagem na barra ou simplesmente digite no texto
[[Imagem:nomearquivo.jpg]]
- Salve a página
- Clique no link gerado
- Faça o upload do arquivo
- Salve e volte ao texto contendo a imagem (atualize com F5)
- Clique sobre a imagem para exibi-la
- Clique em editar e digite uma das predefinições para imagem, como por exemplo:

{{GFDL}}
```

No caso das imagens, as predefinições a serem usadas para licença são:

```
    * {{Cc-by-sa-2.5}} para licença Creative Commons Atribuição -
Compartilhamento pela mesma licença (veja Predefinição:Cc-by-sa-2.5)
    * {{GFDL}} para licença pública GNU FDL (veja Predefinição:GFDL)
    * {{DP}} para imagens em domínio público. (veja Predefinição:DP)
```

Outras opções para inserir imagem

```
[[image:nomeimagem.png|thumb|rótulo]]
```

Exibirá a miniatura da imagem com o rótulo.

```
[[image:nomeimagem.png|left|thumb|50px|]]
```

Exibirá a miniatura da imagem com o rótulo e tamanho 50 pixel e alinhado pela esquerda.

Criando um Índice com itens e subitens

Faça como o exemplo abaixo:

```
#[[/Introdução/]] [[Imagem:3de8.png]]
##[[/Introdução/História|História]]
##[[/Introdução/Características e Recursos|Características e Recursos]]
##[[/Introdução/A quem se destina este livro|A quem se destina este livro]]
#[[/InteligênciaEmocional/]]
#[[/Instalação e configurações do ambiente/]]
##[[/Instalação e configurações do ambiente/Via Xampp|Via Xampp]]
##[[/Instalação e configurações do ambiente/Individualmente|Individualmente]]
#[[/Editores e IDEs para PHP, CSS, JavaScript e HTML/]]
##[[/Editores e IDEs para PHP, CSS, JavaScript e HTML/PHPDesigner|PHPDesigner]]
##[[/Editores e IDEs para PHP, CSS, JavaScript e HTML/Kate|Kate]]
```

Detalhe importante: ao criar um índice assim bastará clicar em qualquer um dos capítulos ou itens para que a página seja criada em branco e possamos adicionar conteúdo.

Obs.: Veja que na primeira linha aparece uma pequena imagem ao lado direito. O wikibooks tem diversas dessas imagens. Veja na Biblioteca.

Para que apareça uma imagem representando o estágio atual do texto de um capítulo ou item use:

`[[Imagem:3de8.png]]`

à direita do item.

Evitar Interpretação do Mediawiki

Uma forma de evitar que o media wiki interprte o que escreve é usando a tag <pre>... </pre>.

Para que de fato nunca interprete usar a tag <nowiki>... </nowiki>.

Para conhecer mais detalhes do uso do Mediawiki edite qualquer página e observe o conteúdo atentamente.

Seções

```
==Secão==
===Subseção===
====Sub-subseção====
```

Comunicação entre Usuários registrados

Os usuários registrados do Wikibooks (mwdiawiki) comunicam-se através de seus forums, cujo link "discussão" e "minha discussão" acima. Este último aparece somente quando estamos logados.

- No caso devemos acessar a área do usuário com quem queremos conversar

- Clicamos em discussão

- E clicamos no link do assunto já aberto.

- Caso queiramos abrir um novo tópico clicamos no sinal de + e criamos o tópico

Sempre ao final das mensagens devemos nos lembrar de assinar a mensagem.

Assinar Mensagens

Assina-se a mensagem com ~~~ (3 tils, que irão exibir o login do usuário)

ou com ~~~~ que mostrarão o login e a data completa.

Código Fonte

Ao chegar a um site wiki e ao invés de editar aparecer Exibir Código Fonte é porque o administrador impediu a edição por algum motivo.

Wiki: ser ou não ser?

Ótimo artigo do Carlos Nepomuceno sobre os valores do wiki - http://www.dicas-l.com.br/conhecimento_em_rede/conhecimento_em_rede_20070510.php

Extensão para Firefox da MediaWiki

Após ter se acostumado com a sintaxe do mediawiki é interessante e produtivo usar algo como essa extensão que ajuda na criação e edição das páginas.

Site para instalação - https://addons.mozilla.org/en-US/firefox/addon/1662

Referências

- Tutorial de Mediawiki em português do Eriberto - http://www.eriberto.pro.br/wiki/index.php?title=Instala%C3%A7%C3%A3o_do_MediaWiki_em_sites_pessoais_em_WebHostings

CMSs (Gerenciadores de Conteúdo)/WordPress (blogs)

De Wikibooks

8.6 - Wordpress

Instalação

Instalação do CMS Wordpress

Página oficial - http://wordpress.org/

Download - http://wordpress.org/download/

Documentação - http://codex.wordpress.org/Main_Page

Existem dois modos de instalar o Wordpress. Um deles é utilizando o painel de administração do seu provedor de serviços(host), o outro é instalando manualmente o sistema através de acesso FTP. Como as instalações do tipo "clicou-instalou" devem seguir os procedimentos estabelecidos pelo host, iremos abordar a instalação via FTP.

Requerimentos

- Browser de sua preferência;

- Acesso ao disco rígido em que o sistema será instalado;

- Editor de textos;

- PHP versão 4.2 ou superior;

- MYSql versão 4.0 ou superior.

Estando tudo de acordo seguimos para a instalação do sistema.

- Faça o download do pacote direto do site oficial;

- Descompacte o arquivo no diretório que quiser. Isto irá descompactar um diretório de nome *wordpress*;

- Entre neste diretório e edite o arquivo *wp-config-sample.php*. Para isso, abra o arquivo e você verá, entre outras, as três linhas seguintes:

```
define('DB_NAME', wordpress);
define('DB_USER', username);
define('DB_PASSWORD', password);
```

Você deve substituir os termos em negrito de acordo com o que está abaixo:

DB_NAME: É o nome do banco de dados (que você criou ou ainda criará);

DB_USER: O nome de usuário do banco de dados;

DB_PASSWORD: A senha de usuário do banco de dados;

- Salve o arquivo como wp-config.php

- Carregue todos os arquivos desse diretório no host que você está utilizando;

- Crie o banco de dados apontado no arquivo acima no host, caso ainda não tenha criado.

- Ao apontar o browser para a pasta em que o wordpress foi colocado uma mensagem pedindo para que você acesse o arquivo "install.php" será exibida. Clique no link que liga a este arquivo e a instalação terá início.

- Clique em First Step

- Dê um título ao seu WebLog e entre com o e-mail do admin.

- Então clique em Continue to Second Step

- Agora serão criadas as tabelas e populadas com informações. Anote o login (admin) e a senha indicada para o acesso. Aqui indicou "admin" e "94db5c".

Então clique no link

- http://127.0.0.1/wordpress/wp-login.php para fazer o acesso, fornecendo os dados anotados.

CMSs (Gerenciadores de Conteúdo)/eGroupWare (colaboração)

De Wikibooks

8.7 - eGroupWare (Colaboração)

O eGroupWare é um software de colaboração escrito em PHP com suporte aos SGBDs MySQL, PostgreSQL, Oracle, SAP/ODBC, SQL Sevrver

Instalação em distribuições baseadas no Debian

apt-get install egroupware

apt-get install egroupware-manual

Instalação no Windows

- Download da versão .zip de www.egroupware.org

40MB a versão 1.2-104 e após descompactado ocupa

- Descompacte no diretório web

- Abra no browser com http://127.0.0.1/egroupware

- Selecione o idioma (Brazil)

- Clique no link Executar Testes de Instalação

- Corrija os erros (vermelho) e avisos (amarelo) e repita a checagem.

Então clique em Prosseguir para o Cabeçalho Admin.

- Veja que ele já tenta detectar os SGBDs instalados.

Preencha as informações que criação o script header.inc.php

Lembre de criar antes o banco de dados (o egroupware pode criar na instalação), o usuário para o egroupware e entrar em Usuário da base de dados inclusive senha ou usar um usuário existente.

Como também o usuário de configuração.

e clique em Configuração de Escrita

- Clique em Continuar

- Entre com login e senha para definir a conexão com o Administrador e clique em Login.

- Clique em Verificar novamente a instalação

- Clique em "Instalar" todos os aplicativos.

- Clique em Verificar novamente a instalação

- Clique em Editar configuração atual (caso apareça, e-mail por exemplo) e corrija as configurações.

- Clique em Criar conta do administrador

Entre com os dados para o administrador e clique em Salvar.

Quando tudo estiver OK clique em Desconectar e acesse:

http://localhost/egroupware

Faça o login como usuário de configuração e gerencie o eGroupWare, crie compromissos, usuários, tarefas, projetos, etc.

Muito bom este CMS. Após instalar confira seus recursos e o personalize à vontade.

CMSs (Gerenciadores de Conteúdo)/Moodle (eLearning)

De Wikibooks

8.8 - Moodle (eLearning)

Moodle Gerenciador de Conteúdos para Cursos Online

Este tutorial foi baseado em uma instalação do Moodle 1.6.1 for Windows.

Com as devidas alterações pode ser adaptado para uma instalação no Linux.

1 - Download - www.moodle.org

Baixar sempre a última versão estável para seu sistema operacional.

2 - Descompactar no diretório web.

Se no Linux ou similar crie o diretório moodledata no mesmo nível do diretório do moodle, com permissão de escrita.

3 - Criar o banco para o moodle (moodle). Atualmente suporta MySQL e PostgreSQL.

4 - Abrir o browser com http://127.0.0.1/moodle/

5 - Selecionar o idioma (Português - Brasil) e Next

6 - Se tudo OK clique em Próximo. Caso contrário corrija as pendências.

7 - Na tela Instalação apenas clique em Próximo

8 - Na tela de configuração do SGBD, entre com os dados e clique em Próximo.

9 - Se tudo OK clique em Próximo

10 - Se conectado à Internet e sugerido "Baixar o Pacote de Idioma", então faça o download e depois clique em Próximo.

11 - Caso o Moodle não consiga criar o arquivo config.php no raiz do diretório moodle, então crie manualmente o arquivo e dê permisão de escrita ao mesmo. Ou então faca o download sugerido no diretório do moodle.

12 - GPL License. Ao final clique em Yes.

13 - Setting up database. Ao final clique em Continue

14 - Administration. Clique em Continue.

15 - Variables. Faça as devidas alterações e ao final clique em Save changes.

16 - Setting up module tables. Ao final clique em Continue.

17 - Setting up plugin tables. Ao final clique em Continue.

18 - Upgrading database. Ao final clique em Continue.

19 - Upgrading database. Ao final clique em Continue.

20 - Setting up block tables. Ao final clique em Continue.

21 - Setting up plugin tables. Ao final clique em Continue.

22 - Site settings.

Entre com as informações para a criação do site e faça as configurações que desejar. Ao final clique em Save changes.

23 - Configurar conta do administrador. Entre com os dados do usuário administrador e clique em Update profile.

24 - Concluído. Estamos no site, logados como admin.

25 - Adicionar Nova Categoria

- Administração (Admin...) - Cursos (Courses) - Administrar Cursos (Manage courses)

Mais detalhes em - http://docs.moodle.org/en/Installing_Moodle

Cópia local em - lang/en_utf8/help/install.html (or your language)

Curso sobre o Moodle em http://cursos.cdtc.org.br e http://comunidade.cdtc.org.br

Aplicativos em PHP/FrameWorks

De Wikibooks

9 - FrameWorks

"O primeiro dever da inteligência é desconfiar dela mesma." (Einstein)

1. P4A
2. Zend
3. Seagull
4. CodeIgniter
5. QCodo
6. Zephyr-php
7. CakePHP
8. Symfony

Framework na Wikipedia

http://pt.wikipedia.org/wiki/Framework

No desenvolvimento do software, um framework ou arcabouço é uma estrutura de suporte definida em que um outro projeto de software pode ser organizado e desenvolvido. Um framework pode incluir programas de suporte, bibliotecas de código, linguagens de script e outros softwares para ajudar a desenvolver e juntar diferentes componentes de um projeto de software.

Frameworks são projetados com a intenção de facilitar o desenvolvimento de software, habilitando designers e programadores a gastarem mais tempo determinando nas exigências do software do que com detalhes tediosos de baixo nível do sistema.

Orientação a Objeto

Especificamente em orientação a objeto, framework é um conjunto de classes com objetivo de reutilização de um design, provendo um guia para uma solução de arquitetura em um domínio específico de software.

Framework se diferencia de uma simples biblioteca (toolkit), pois esta se concentra apenas em

oferecer implementação de funcionalidades, sem definir a reutilização de uma solução de arquitetura (design).

Opções de frameworks para PHP

http://phpbrasil.com/articles/article.php/id/1428

FrameWorks/P4A

De Wikibooks

Conteúdo

- 1 P4A – PHP for Application
- 2 RAD
- 3 Framework
- 4 Elementos do P4A
  - 4.1 Masks
  - 4.2 Widgets
- 5 Características
  - 5.1 Busca
  - 5.2 Widgets
- 6 Desenvolvendo Aplicativos com o P4A
  - 6.1 Exemplo de Estrutura Mínima de uma Aplicação em P4A
  - 6.2 Primeiro Exemplo – Olá Mundo
  - 6.3 Segundo Exemplo – Acessando Banco de Dados
  - 6.4 Exemplo Com 3 Tabelas
- 7 Dicas Extras

P4A – PHP for Application

Site oficial - http://p4a.sf.net

P4A (PHP For Applications) é um framework em PHP tipo RAD e orientado a objetos para construção event-driven de aplicações web. Usa tableless HTML, suporte às teclas de atalho, point&click gerador de aplicativos, reconhecimento automático de datos, UTF-8, i18n/l10n, integração com a biblioteca PEAR.

De autoria do italiano Fabrizio Balliano.

Segundo a Wikipédia (http://pt.wikipedia.org):

RAD

Rapid Application Development (RAD), também conhecido como Desenvolvimento Rápido de Aplicação, é um modelo de processo de desenvolvimento de software iterativo e incremental que

enfatiza um ciclo de desenvolvimento extremamente curto (entre 60 e 90 dias). O termo foi registrado por James Martin em 1991 e tem substituído gradativamente o termo de prototipação rápida que já foi muito utilizada no passado

Framework

No desenvolvimento do software, um framework é uma estrutura de suporte definida para que outro projeto de software possa ser organizado e desenvolvido. Tipicamente, um framework pode incluir programas de apoio, bibliotecas de código, linguagens de script e outros softwares para ajudar a desenvolver e juntar diferentes componentes de um projeto.

Elementos do P4A

Masks

É o objeto básico de interfaces, que contem todos os widgets e geralmente exibe objetos.

Localizado em /p4a/objects/mask.php (linha 45)

```
P4A_Object
    |
    --P4A_Mask
```

Widgets

São os elementos GUI prontos que podemos usar para compor nossas máscaras (masks). Classe base para objetos que permitem a usuários interagirem com a aplicação: botões, menus, .

Localizado em /p4a/objects/widget.php (linha 48)

```
P4A_Object
    |
    --P4A_Widget
```

O P4A é um projeto abrigado no SourceForge desde fevereiro de 2003, que chegou ao 12o. lugar no ranking geral dos mais de 100.000 projetos do SourceForge, tendo ganhado alguns prêmios neste repositório. Atualmente fica em primeiro lugar entre os frameworks de PHP no SourceForge.net.

Já conta com locales para uns 160 países e tradução para uns 60, inclusive o Brasil (pt_BR).

P4A é um framework PHP, um software contendo bibliotecas, módulos e controles (widgets) usados para a construção de aplicações que devem ser usadas com um web browser conectado à Internet. Com P4A você estará habilitado a desenvolver e escrever código de aplicações web como você faria com as ferramentas RAD mais evoluídas. O resultado deverá ser uma aplicação que o usuário deverá usar pensando que está usando uma aplicação cliente clássica (tipo desktop).

Você não deve se preocupar com a criação da interface gráfica para seus projetos, porque P4A já tem um tema gráfico cliente agradável.

Características

- Escrito em PHP (www.php.net), compatível com as versões PHP 4.3.x e 5.x.x e Apache 1.3.x e 2.0.x em Linux e Windows

- Integrado com a biblioteca Pear/DB (http://pear.php.net) e com isso suporta todos os bancos da suportados pela PEAD/DB: PostgreSQL, MySQL, SQL Server, Oracle e outros.

- Lógica e Design separados pelo template (http://pear.php.net/package/HTML_Template_Flexy)

Busca

Outras informações sobre o Projeto P4A:

```
- Home Page – http://p4a.sourceforge.net
- Downloads - http://sourceforge.net/project/showfiles.php?group_id=98294
- Demo online - http://p4a.sourceforge.net/demo
- Forum em italiano - http://sourceforge.net/forum/forum.php?forum_id=473897
- Forum em inglês - http://sourceforge.net/forum/forum.php?forum_id=340765
- Quem está usando - http://p4a.sourceforge.net/who-s-using-p4a
- Referência de Código - http://p4a.sourceforge.net/code-reference
- Tutoriais - http://p4a.sourceforge.net/tutorials
- Widgets - http://p4a.sourceforge.net/widgets
- Icon Packs - http://p4a.sourceforge.net/icons-packs
- Contribuições - http://p4a.sourceforge.net/info-about-contribs (Para instalar
uma contribuição, basta descompactar no diretória da sua aplicação.
Após instalar o contrib todos os objetos necessários ao P4A serão automaticamente
incluídos na sua aplicação).
- Protótipo de Contribuição - http://p4a.sourceforge.net/contrib-prototype
(protótipo vazio para ajudar pessoas a criarem uma nova contribuição).
- Logo mais abaixo da seção anterior (no site oficial) outros aplicativos de
terceiros úteis.
```

Widgets

São os elementos GUI prontos que podemos usar para compor nossas máscaras (masks). Classe base para objetos que permitem a usuários interagirem com a aplicação: botões, menus, .

Esta lista (dos principais) é aninhada para reproduzir a herança das classes. Todos estes objetos

extendem a classe P4A_Widget. Veja a referência de código para detalhes.

```
    * Button
    * Canvas
    * Field
    * Frame
          o Fieldset
    * Menu
    * Message
    * Table
    * Tab pane
    * Toolbar
          o Actions toolbar
          o Navigation toolbar
          o Simple toolbar
          o Standard toolbar
          o Quit toolbar
```

Desenvolvendo Aplicativos com o P4A

Pré-Requisitos:

- Sistemas Operacionais Linux ou Windows

- PHP 4.3.x/4.4.x/5.x.x com Apache 1.3.x/2.0.x

- A biblioteca PEAR/DB

- Um banco de dados suportado pela bibliteca acima, caso o seu aplicativo use banco

Quando instalamos o Xampp (http://xampp.sf.net), ele provê todos os requisitos do P4A, inclusive tem versão para ambos os sistemas operacionais.

Quando descompactamos o p4a ele cria a seguinte estrutura de diretórios:

```
p4a/applications
p4a/docs
p4a/icons
p4a/libraries
p4a/p4a
p4a/themes
p4a/index.php
p4a/p4a.php
```

Os aplicativos criados devem ficar na pasta “applications”.

Na pasta “docs” está toda a referência do P4A.

Em "icons" estão os ícones utilizados, em vários tamanhos.

A pasta "p4a" é a principal do framework. Nela estão as bibliotecas, os objetos, os locales e mais alguns componentes importantes.

Os temas devem ficar na pasta "themes"

Exemplo de Estrutura Mínima de uma Aplicação em P4A

```
<ola_mundo>
index.php
   <objects>
    ola_mundo.php
    ola_mascara.php
```

Caso a aplicação utilize algum upload deverá ter uma pasta extra "uploads", contendo uma sub pasta "tmp" e ambas com permissão 777.

Permissões: os demais arquivos: index.php, ola_mundo.php e ola_mascara.php devem ter permissão de leitura, escrita e execução para o dono e leitura e execução para Outros.

Primeiro Exemplo – Olá Mundo

1)Criamos uma pasta ola_mundo dentro da pasta applications.

2)Criamos dentro de ola_mundo o arquivo index.php com o seguinte conteúdo:

index.php

```
<?php
 // Carregar a biblioteca do framework
require_once( dirname(__FILE__) . '/../../p4a.php' );

// Instanciar a classe que define a aplicação ola_mundo.
// A chasse encontra-se no diretório objects em um arquivo
// com o mesmo nome da classe
$app =& p4a::singleton("ola_mundo");

// Lançar a aplicação
$app->main();
?>
```

No arquivo index.php existe uma chamada para a classe principal do aplicativo que encontra-se no

arquivo de mesmo nome (ola_mundo.php) dentro da pasta objects.

ola_mundo.php

```
<?php

// A classe principal da aplicação sempre estende a classe p4a do framework
class ola_mundo extends P4A{
        // Construtor da classe
        function ola_mundo ()   {
                // A primeira coisa a fazer, sempre é lançar o construtor da
super classa p4a
                parent::p4a();

                // A seguir o código para executar, tipicamente abre a primeira
máscara (masks)
                $this->openMask("mensagem");
        }
}
?>
```

Veja que a classe principal abre o arquivo de máscara (mensagem.php).

mensagem.php

```
<?php

// Para criar uma máscara, vem criar uma classe que herda da classe P4A_Mask.
class mensagem extends P4A_Mask{
        // Construtor da classe
        function mensagem()     {
                // Para a aplicação a primeira coisa que se lança é o construtor
da classe pai
                parent::P4A_Mask();

                // Eles vem instanciar o widget para posicionar sobre a máscara
usando o método
                // build da máscara (ou melhor, da classe P4A_Objects da qual
tudo deriva

                // O objeto box é um Label de texto não modificável.
                // Uma vez instanciado o texto adota a mesma posição, formatável
em HTML
                // acessável com $this->box

                $this->build("p4a_box","box");
                $this->box->setValue("Olá Mundo");

                // O objeto frame é um tipo de container disponível que
                // colabora para o posicionamento relativo (sem tabela html) dos
widget
                //em questão, caso se acesse o widget criado usando o apontador
retornado com build
                $frm =& $this->build("p4a_frame","frame");
                $frm->anchorCenter($this->box);
```

```
                // Finalmente visualizamos o container (e o widget contido)
                // em uma de 3 seções do template (por default são top,menu,main)
                $this->display("main", $frm);

                $this->box->addAction("onClick");
                $this->intercept($this->box, "onClick", "change_box");
        }
        function change_box()
        {
                if ($this->box->getValue() == "Olá mundo") {
                        $this->box->setValue("Clique para acessar");
                } else {  // Abaixo adiciona um ponto ao final
                        $this->box->setValue($this->box->getValue() . ".");
                }
        }
}
?>
```

A pasta objects pode ou não conter uma sub pasta "masks". Caso criemos a pasta "masks", ela deverá contr os arquivos de interface da aplicação (masks). Em termos de organização e entendimento da estrutura é adequada a criação da subpasta masks.

Experimente criar este pequeno exemplo com e sem a pasta masks. Nela apenas ficará o arquivo mensagem.php. Dica: instale a extensão "Web develop" no Firefox e sempre que houver algum erro, corrija o erro e clique em Cookies – Clear Session Cookies e só então clique no botão Refresh do browser.

Se no Internet Explorer, feche e abra o browser novamente após qualquer alteração, ou faça alguma configuração correspondento a da extensão do FF.

Segundo Exemplo – Acessando Banco de Dados

Este exemplo usa um banco de nome "p4a", com o SGBD PostgreSQL. Teremos apenas uma tabela, funcionarios, com a estrutura abaixo:

```
CREATE TABLE funcionarios (
    codigo integer NOT NULL,
    nome character varying(40),
    siape character varying(10),
    foto text,
    observacao text,
    primary key (codigo)
);
```

A estrutura do aplicativo será:

```
<funcionarios>
```

```
        index.php
        <objects>
                cadastro.php
                funcionarios.php
        <uploads> (Com permissão de escrita - 777>
                <tmp> (Com permissão de escrita - 777>
```

Arquivos:

index.php

```
<?php

// Ampliar retornos de erro do P4A
define("P4A_EXTENDED_ERRORS", 'TRUE');

// Define locales para pt_BR. Muito útil para o usuário este recurso, já que
campos:
// numéricos receberão a máscara 1.234,56 e campos data
// 25/12/2005 além de outras facilidades
define("P4A_LOCALE", 'pt_BR');

// Define dados da conexão ao banco
define("P4A_DSN", 'pgsql://postgres:suasenha@127.0.0.1:5432/p4a');

//define("P4A_DSN", 'mysql://root:@127.0.0.1/p4a');
//Incluir lib P4A
require_once( dirname(__FILE__) . '/../../p4a.php' );

/*
Now we instance the application with the singleton
method, than we call the "main" method. main is executed
every page call (click and reload included).
The application must be a class under the "objects" directory
and have to extend "p4a" class.
Attention, in p4a all object have to be assigned using the "=&"
operator or you will loose all references. You also have to
use the "&" operator in the method definition, when the method
returns objects (also for the class constructor).
Take a look at "cadastro" class for a better understanding.
*/

// Checar instalação e configuração.
// As linhas abaixo devem ser removidas após a primeira execução
/*
$check = p4a_check_configuration();

// Here we go
if (is_string($check)) {
        print $check;
} else {
        $p4a->main();
}
*/
$app =& p4a::singleton("cadastro");
$app->main();
?>
```

objects/cadastro.php

```
<?php
 // A classe da aplicação sempre estende a classe p4a do framework
class cadastro extends P4A{
        // Construtor da classe
        function cadastro()     {
                // A primeira coisa a fazer sempre é lançar o construtor da
classe p4a
                parent::p4a();

                // Construção do menu
                $this->build("p4a_menu", "menu");
                $this->menu->addItem("cadastro", "C&adastro");

                $this->menu->items->cadastro-
>addItem("funcionarios","Funcionários");
                $this->intercept($this->menu->items->cadastro->items-
>funcionarios,
                                                "onClick", "menuClick");

                //Em ... addItem("funcionarios","Funcionários"),
                // funcionarios é a classe e Funcionários o label.

                $this->menu->items->cadastro-
>addItem("funcionarios2","Funcionários2");
                $this->intercept($this->menu->items->cadastro->items-
>funcionarios2,"onClick",
                                                            "menuClick");

                // Código da execução, tipicamente abre a primeira máscara (mask)
                $this->openMask("funcionarios");
        }

        function menuClick()    {
                $this->openMask($this->active_object->getName());
        }
}
?>
```

objects/funcionarios.php

```
<?php
// Para criar uma máscara, cria uma classe que herda da classe P4A_Mask
class Funcionarios extends P4A_Mask{
        // Construtor da classe funcionarios
        function Funcionarios() {
                //Para a aplicação a primeira coisa se lança o costrutor da
classe pai
                $this->p4a_mask();
                $p4a =& p4a::singleton();

                // DB Source
                $this->build("p4a_db_source", "source");
                /*$this->source->setFields(array("funcionarios.*" => "*"));*/
                $this->source->setTable("funcionarios");
                $this->source->setPk("codigo");
```

```
$this->source->addOrder("nome");
$this->source->setPageLimit(10);
$this->source->load();
//$this->source->fields->codigo->setSequence("funcionarios");
$this->source->fields->codigo->setSequence("codigo");

$this->setSource($this->source);
$this->source->firstRow();

// Personalizar propriedades dos campos (vide função abaixo)
$this->setFieldsProperties();
$fields =& $this->fields;

// Busca em registros
$fs_search =& $this->build("p4a_fieldset","fs_search");
$fs_search->setTitle("LocalizaÃ§Ão de FuncionÃirios");
$txt_search =& $this->build("p4a_field", "txt_search");
$txt_search->addAction("onReturnPress");
$this->intercept($txt_search, "onReturnPress","search");
$txt_search->setLabel("Nome ou parte");
$cmd_search =& $this->build("p4a_button","cmd_search");
$cmd_search->setValue("Ok");
$this->intercept($cmd_search, "onClick","search");
$fs_search->anchor($txt_search);
$fs_search->anchorLeft($cmd_search);

// Toolbar
$this->build("p4a_standard_toolbar", "toolbar");
$this->toolbar->setMask($this);

// Tabela (exibe funcionários, um registro por linha)
$table =& $this->build("p4a_table", "table");
$table->setWidth(725);
$table->setSource($this->source);
$table->setVisibleCols(array("codigo","nome","siape"));
$table->cols->codigo->setLabel("CÃ³digo");

while ($col =& $table->cols->nextItem()) {
        $col->setWidth(150);
}
$table->showNavigationBar();

// Mensagem de erro
$message =& $this->build("p4a_message", "message");
$message->setWidth("300");

//Fieldset con l'elenco dei campi
$fset=& $this->build("p4a_fieldset", "frame");
$fset->setTitle("Cadastro do FuncionÃirio");

$fset->anchor($this->fields->codigo);
$fset->anchor($this->fields->nome);
$fset->anchor($this->fields->siape);
$fset->anchor($this->fields->foto);
$fset->anchor($this->fields->observacao);

// Frame (abriga mensagem, tabela, busca e campos)
```

```
                $frm=& $this->build("p4a_frame", "frm");
                $frm->setWidth(730);
                $frm->anchor($fs_search);
                $frm->newRow();
                $frm->anchorCenter($message);
                $frm->anchor($table);
                $frm->anchor($fset);

                // Campos requeridos
            $this->mf = array("codigo", "nome", "siape");
                foreach($this->mf as $mf){
                        $fields->$mf->label->setFontWeight("bold");
                }

                // Exibir controles (Display)
                $this->display("main", $frm);
                $this->display("menu", $p4a->menu);
                $this->display("top", $this->toolbar);
        }

        function main() {
                parent::main();
                foreach($this->mf as $mf){
                        $this->fields->$mf->unsetStyleProperty("border");
                }
        }
        // Função que personaliza propriedades dos campos
        function setFieldsProperties()  {
                $p4a =& p4a::singleton();
                $fields =& $this->fields;

                $fields->codigo->setLabel("CÃ³digo");
                $fields->codigo->setWidth(200);
                $fields->codigo->enable(false);

                $fields->nome->setWidth(400);

                $fields->siape->setLabel("SIAPE");
                $fields->siape->setWidth("400");

                $fields->foto->setType("image");

                $fields->observacao->setType("rich_textarea");
                $fields->observacao->enableUpload();
        }
        // Função que salva os registros
        function saveRow()      {
                $valid = true;

                foreach($this->mf as $mf){
                        $value = $this->fields->$mf->getNewValue();
                        if(trim($value) === ""){ // Abaixo, em vermelho as bordas
dos campos
                                $this->fields->$mf->setStyleProperty("border",
"1px solid red");
                                $valid = false;
                        }
```

```
                }

                if ($valid) {
                        parent::saveRow();
                }else{
                        $this->message->setValue("Favor preencher todos os campos
requeridos!");
                }
        }
        // Função que implementa a busca
        function search()       {
                $value = $this->txt_search->getNewValue();
                $this->data->setWhere("nome ILIKE '%{$value}%'"); // Busca case-
insensitive
                $this->data->firstRow();
                $num_rows = $this->data->getNumRows();

                if (!$num_rows) {
                        $this->message->setValue("Nenhum nome encontrado
contendo: $value!");
                        $this->data->setWhere(null);
                        $this->data->firstRow();
                }
        }
}
?>
```

Exemplo Com 3 Tabelas

Este é um exemplo que acompanha o P4A original,contendo 3 tabelas, com exemplos de joins e que roda bem no MySQL. No PostgreSQL, devido a algumas incompatibilidades com a PEAR/DB, ele não roda bem. Segundo o autor do P4A, Fabrizi, esta incompatibilidade é apenas com os joins.

A estrutura das tabelas é:

```
CREATE TABLE brands (
  brand_id INTEGER NOT NULL,
  description TEXT NOT NULL,
  visible BOOL NOT NULL,
  PRIMARY KEY(brand_id)
);

CREATE TABLE categories (
  category_id INTEGER NOT NULL,
  description TEXT NOT NULL,
  visible BOOL NOT NULL,
  PRIMARY KEY(category_id)
);

CREATE TABLE products (
  product_id VARCHAR(50) NOT NULL,
  brand_id INTEGER NOT NULL,
  category_id INTEGER NOT NULL,
```

```
  model TEXT NOT NULL,
  purchasing_price DECIMAL(10,2) NOT NULL,
  selling_price DECIMAL(10,2) NOT NULL,
  discount INTEGER NOT NULL,
  little_photo TEXT NULL,
  big_photo TEXT NULL,
  is_new BOOL NOT NULL,
  visible BOOL NOT NULL,
  description TEXT NOT NULL,
  PRIMARY KEY(product_id)
);
```

Crie o banco p4a no MySQL com as tabelas acima.

A estrutura dos arquivos da aplicação:

```
<products_catalogue>
        index.php
        <objects>
                products_catalogue.php
                products.php
                categories.php
                brands.php
        <uploads> (Com permissão de escrita - 777>
                <tmp> (Com permissão de escrita - 777>
```

Observe que:

index.php em $p4a =& p4a::singleton("products_catalogue") chama a classe products_catalog no arquivo de mesmonome.

products_catalog.php em $this->openMask("products") chama a classe products no arquivo products.php

Através do menu se tem acesso as demais classes dos arquivos brands.php e categories.php.

O código do aplicativo acompanha o P4A, portanto não há necessidade de colar aqui, somente colarei alguns trechos para comentar:

objects/products_catalogue.php

Neste arquivo encontramos este trecho de código:

```
                // Data sources
                $this->build("p4a_db_source", "brands");
                $this->brands->setTable("brands");
                $this->brands->setPk("brand_id");
                $this->brands->addOrder("description");
                $this->brands->load();
                $this->brands->fields->brand_id->setSequence("brands");
```

```
                $this->build("p4a_db_source", "categories");
                $this->categories->setTable("categories");
                $this->categories->setPk("category_id");
                $this->categories->addOrder("description");
                $this->categories->load();
                $this->categories->fields->category_id-
>setSequence("categories");
```

Este diz quais as duas outras tabelas (db_sources) que serão utilizadas no aplicativo.

Trecho de "objects/products.php" definindo uma combo com acesso ao banco:

```
                $fields->brand_id->setLabel("Brand");
                $fields->brand_id->setWidth(200);
                $fields->brand_id->setType("select");
                $fields->brand_id->setSource($p4a->brands);
                $fields->brand_id->setSourceDescriptionField("description");
```

O index.php deve ficar parecido com este. Ajuste para os dados do seu MySQL:

```
<?php
// Caso apareça algum erro, é útil descomentar alinha abaixo para melhor debug
//define("P4A_EXTENDED_ERRORS", 'TRUE');
define("P4A_LOCALE", 'en_US');
define("P4A_DSN", 'mysql://root:@localhost/p4a');
//define("P4A_DSN", 'pgsql://postgres:postabir@localhost:5432/p4a');

require_once dirname(__FILE__) . '/../../p4a.php';

// Check Installation and configuration.
// This lines should be removed after the first run.
$p4a =& p4a::singleton("products_catalogue");
$check = p4a_check_configuration();

// Here we go
if (is_string($check)) {
        print $check;
} else {
        $p4a->main();
}
?>
```

Dicas Extras

Suporte ao Ajax

A versão 1.99.2 já traz suporte ao Ajax. Basta ver o calendário adicionado ao lado dos campos data.

Acentuação

Quando à acentuação vale alertar que a codificação do P4A é a UTF8. Caso tenha algum problema com acentuação deverá encontrar um editor com suporte a essa codificação para corrigir os acentos, como é o caso dos editores do Linux gEdit e Kate.

Dicas do forum do projeto (no sourceForge.net).

Tenho duas tabelas. Quando clicar no botão Save, na Toolbar, quero salvar as duas tabelas.

R - Sobrescreva o método saveRow() na sua mask. Se você tem 2 db_sources você pode fazer:

parent::saveRow(); $outro_db_source->saveRow();

E assim por diante.

- Eu tenho duas mask e preciso compartilhar dados entre ambas. Ao clicar no registro eu preciso abrir a outra mask

R – Você pode acessar dados de qualquer mask com: $p4a->masks->nomemask->desejado

Como interceptar o método Save na Toolbar?

R - $this->intercept($toolbar->button->save, “onClick”, “method”)

Como inserir dados?

R- Você podecriar um db_source e fazer e circular por todos os elementos que você deseja inserir

```
$db_source->newRow();
$db_source->setValue(“valor”);
$db_source->saveRow();
```

Como alterar Labels?

R -

$this->fields->fieldname->setLabel(“Meulabel”);

$table->cols->nomecoluna->setLabel(“Meulabel”);

Como exibir em um campo data a data de hoje por default?

R-

$source->fields->order_date->setDefaultValue(P4A_Date::now(P4A_DATE));

Ou

$format = '%d/%m/%Y-%H/%M/%S'; // Ou outro formato

.

.

P4A_Date::now($format);

Instalando Novos Temas no P4A

Acesse a página - http://p4a.sourceforge.net/icons-packs

E faça o download do tema desejado. Veja que para instalar basta editar a index.php do P4A e adicionar a linha:

define("P4A_ICONS_PATH", "/path/to/the/icons/directory");

Atualmente (06/2007) o P4A está na versão estável 2.0.6 e 2.1.3 em desenvolvimento.

FrameWorks/Zend

De Wikibooks

O Zend Framework é o framework desenvolvido por desenvolvedores da engine do PHP.

Zend Framework para um PHP mais poderoso

http://felipetonello.com/blog/2007/02/28/zend-framework-para-um-php-mais-poderoso/

FrameWorks/Seagull

De Wikibooks

9.4 - Framework Seagull

Seagull é um framework OOP maduro para a construção de aplicativos web, aplicações em linha de comando e em GUI. Licenciado sob BSD, o projeto permite que desenvolvedores PHP facilmente integrem e gerenciem seus códigos e construam aplicações complexas rapidamente.

Site oficial - http://seagull.phpkitchen.com/

Download - http://sourceforge.net/project/showfiles.php?group_id=92482

Instalação

- O diretório seagull/var deve ter permissão de escrita e também o seagull/www/images.

- Chame no navegador http://localhost/seagull/www

- Siga os passos de uma instalação interativa

FrameWorks/CodeIgniter

De Wikibooks

Framework CodeIgniter

Site Oficial - http://www.codeigniter.com/

Site com vários tutoriais e exemplos - http://swik.net/codeigniter+Tutorial

Download - http://codeigniter.com/download.php

Tutoriais em Vídeo - http://codeigniter.com/tutorials/

User Guide - http://codeigniter.com/user_guide/

Forum - http://codeigniter.com/forums/

Construídos com CodeIgniter - http://codeigniter.com/projects/

FrameWorks/QCodo

De Wikibooks

9.5 - Framework QCodo

Ótimo Framework que também gera aplicativos completos apenas com a indicação do banco.

Site oficial - http://www.qcodo.com/

Documentação - http://www.qcodo.com/documentation/

1 - Download - http://www.qcodo.com/downloads/

Faça o download do official release para seu distema e descompacte. Descompacte o arquivo. Crie uma pasta "qcodo" no seu diretório web e copie todo o conteúdo da pasta wwwroot para esta pasta qcodo criada.

Edite o arquivo includes/_configuration.php

2 - Defina o diretório do qcodo

define ('DOCROOT_SUBFOLDER', '/qcodo');

Caso esteja usando o qcodo em um diretório virtual ajuste adequadamente

DOCROOT_VIRTUAL_DIRECTORY no arquivo includes/_configuration.php

3 - Defina os dados do SGBD

define('DBCONFIG_1','Adapter=MySqli5Database; Server=localhost; Port=; Database=test; Username=root; Password=; Profiling=false; ScriptPath=;');

Faça as devidas adaptações: Server, Port, Database, Username, Password. Por exemplo:

define('DBCONFIG_1','Adapter=MySqli5Database; Server=localhost; Port=; Database=condominio; Username=root; Password=; Profiling=false; ScriptPath=;');

Atualmente suporta:

MySqlDatabase (MySQL v4.x, using the old mysql extension)

MySqliDatabase (MySQL v4.x, using the new mysqli extension)

MySqli5Database (MySQL v5.x, using the new mysqli extension)

SqlServerDatabase (Microsoft SQL Server)

PostgreSqlDatabase (PostgreSQL)

4 - Abra o browser em http://localhost/qcodo/_devtools/codegen.php para gerar o aplicativo para o banco indicado

5 - Abra o browser em http://localhost/qcodo/form_drafts/ para ver sua aplicação gerada.

Alerta: caso mude o arquivo de configuração para trabalhar com outro banco, antes renomeie o diretório forms_draft para o nome da aplicacao e mova para o raiz web (sugestão).

Depois recrie o diretório forms_draft e copie para ele apenas o index.php que existia.

Antes de rodar a aplicação gerada em outro diretório:

- altere o include_path do /opt/lampp/etc/php.ini adicionando o caminho do diretório includes, como por exemplo:

include_path = ".:/opt/lampp/lib/php:/opt/lampp/htdocs/qcodo/includes"

- Após copiar o aplicativo para outro diretório e adicionar o includes no include_path altere a terceira linha do

```
index.php para que fique assim:

require('prepend.inc');
```

Este pequeno tutorial visa apenas a geração de código com o qcodo e para isso devemos ter permissão de escrita em vários diretórios. A maneira segura, em sistemas Unix/Linux, sugerida no _README.txt é rodar:

chmod -R ugo+w diretório qcodo.

No Windows, clique com o botão direito no diretório qcodo e Propriedades - Geral - Desmarcar

Somente leitura - Aplicar as alterações a esta pasta, subpastas e arquivos.

Mais detalhes, veja a documentação no site e no _README.txt que acompanha o framework.

A equipe também oferece hospedagem para sites que usem o qcodo:

http://www.qcodohosting.org/page_start.php

FrameWorks/Zephyr-php

De Wikibooks

Guia Rápido de Instalação e Uso do Framework ZEPHYR

Site oficial - http://zephyr-php.sf.net

Conteúdo

- 1 Introdução
- 2 Características
- 3 Instalação
- 4 Iniciando a Usar o Zephyr
- 5 Conexão com o banco de dados
- 6 Operações com Bancos de Dados
- 7 Créditos

Introdução

Um Framework baseado em AJAX e MVC para desenvolvedores de PHP5

Podemos desenvolver aplicações comerciais facilmente usando um framework robusto.

Ele é extremamente fácil de aprender e muito simples de implementar.

Com ele podemos entregar uma aplicação comercial completa em AJAX e em camadas no backend (MVC) dentro de alguns minutos.

Instalação e desenvolvimento de pacotes que você desenvolve para Zephyr são agradáveis.

Alé disso devemos receber todas os recursos do mais popular template engine "smarty" e da poderosa camada de abstração de bancos de dados "ADOdb".

Vamos seguir com a documentação para descobrir como é fácil desenvolver aplicações AJAX usando o framework Zephyr.

Zephyr é um framework que utiliza MVC (model-view-controller) e suporta completamente a separação da camada de negócio da camada de apresentação. O framework foi construído usando bibliotecas populares testadas no backend.

Por isso o código desenvolvido com Zephyr é mais estável e flexível para usuários finais.

Características

Aqui estão algumas das mais importantes características do Zephyr:

1. Usa o template engine Smarty. Assim o desenvolvimento com os templates no zephyr é flexível e estruturado.

```
  Além disso podemos usar todas os poderosos recursos do smarty em nossos
templates.

  Assim a curva de aprendizado é baixa.
```

2. A adoDB é usada como DAL (data access layer) camada de abstração de bancos de dados no zephyr.

3. Podemos gerenciar múltiplos bancos de dados concorrentemente nas ações do zephyr.

4. Fácil integração de scripts personalizados em PHP e de bibliotecas em javascript.

5. O frontend do zephyr é desenvolvido em cima da famosa biblioteca prototype do Sam Stephenson.

```
  Isto reduz o trabalho de usar javascript nos templates e dá a você enorme
flexibilidade.
```

6. Zephyr usa CPAINT como interface para AJAX. Assim você pode facilmente passar múltiplos caracteres de e para suas ações usando zephyr.

7. Cada pacote (package) ou aplicação tem seu próprio inicializador.

8. Implementa processadores de pré ações que ajudam-nos a executar algumas ações específicas antes das ações normais.

```
  Isto deve nos ajudar a implementar validações mais robustas.
```

9. Suporta ações do cron que executam em intervalos regulares.

10. Fácil manipulação de registros do banco de dados. Requisições internas do gerente do zephyr gerencia todos os parâmetros requisitados de uma forma flexível e você não precisa acessar e configurar esses parâmetros exclusivamente.

11. Podemos integrar facilmente com qualquer pacote PEAR.

12. Você pode gerenciar funções de agregação de bancos de dados através do objeto interno do zephyr de agregação de bancos de dados.

13. Suporta múltiplos níveos de filtros definidos pelo usuário.

14. Fácil integração com os ezComponents.

Instalação

Download - http://sourceforge.net/project/showfiles.php?group_id=151993

Apenas faça o download e descompacte no diretório web.

Altere as permissões do diretório "temp" para 777.

Uma estrutura típica do zephyr:

```
- zephyr
  | + abstract
  | + documentation
  | + functions
  | + helper
  | + images
  | + interfaces
  | + internal
  | + javascript
  | + packages
  | + temp
  | + thirdparty
  |   .htaccess
  |   index.php
```

Iniciando a Usar o Zephyr

Todas as aplicações no Zephyr estão no diretório "packages".

Caso pretenda criar uma aplicação com o Zephyr, apenas crie um sub-diretório no diretório "packages" ou então faça uma cópia de um dos aplicativos de exemplo. Esta última opção é mais indicada para quem está dando seus primeiros passos.

Basicamente:

Criar o banco - agenda (MySQL)

```
CREATE TABLE `agd` (
  `codigo` int(11) NOT NULL default '0',
  `nome` varchar(255) NOT NULL default '',
  `email` varchar(50) NOT NULL default '',
  `site` varchar(50) NOT NULL default '',
  PRIMARY KEY  (`codigo`)
)
```

Alterar o script dbinfo.class.php no diretório "helper" da aplicação com as informações corretas.

- Faça uma cópia do aplicativo "student_db" para o nome desejado, "agenda" por exemplo

- Então, tendo em vista sua tabela e campos, edite cada um dos arquivos e troque as ocorrencias de student, students, std, por agenda, agendas, agd e os campos também pelos respectivos.

Para facilitar crie a tabela com a mesma quantidade de campos e com tipos idêntivos aos originais.

- Comece editando o arquivo XML (package.xml) que se encontra no raiz da aplicação

- O arquivo readme.txt ou leiame-txt contém algumas informações para rodar a aplicação, inclusive o esquema do banco.

- Depois percorra cada um dos arquivos do aplicativo.

- Para facilitar use o recurso procurar e sobrescrever, tendo o cuidado de ativar a opção procurar palavra inteira.

Aproveite para observar a estrutura das pastas, arquivos, classes e funções do framework.

Veja a estrutura:

```
- zephyr
  | - packages
  |    | + student_db
  |    | + agenda
```

Script para o PostgreSQL:

```
CREATE TABLE agd (
  codigo int NOT NULL default 0,
  nome varchar(255) NOT NULL default '',
  email varchar(50) NOT NULL default '',
  site varchar(50) NOT NULL default '',
  PRIMARY KEY  (codigo)
)
```

Para suporte ao postgresql adicione este código:

```
case "postgres":
     $dsn =  "{$dbinfo->get_dbtype()}://{$dbinfo->get_dbuser()}:{$dbinfo-
>get_dbpwd()}@{$dbinfo->get_dbhost()}/{$dbinfo->get_dbname()}?persist={$dbinfo-
>get_persist()}";
     break;
```

Logo após a entrada do mysql no arquivo zephyr/helper/DatabaseConector.class.php

Veja aqui um exemplo que criei partindo do student_db (agenda, 7,98Kb).

Conexão com o banco de dados

Os dados encontram-se naclasse dbinfo.class.php dentro do diretório helper:

```
<?
class dbinfo extends abstractdbinfo {
    public function __construct()
    {
            $this->dbhost = "localhost";
        $this->dbname = "agenda";
        $this->dbuser = "root";
        $this->dbpwd = "";
        $this->dbtype = "mysql";
        $this->persist = 1;
```

```
    }
}
?>
```

Para PostgreSQL

```
<?
class dbinfo extends abstractdbinfo {
    public function __construct()
    {
        $this->dbhost = "localhost";
        $this->dbname = "agenda";
        $this->dbuser = "postgres";
        $this->dbpwd = "postgres";
        $this->dbtype = "postgres";
        $this->persist = 1;
    }
}
?>
```

O Zephyr vem uma estrutura mínima do ADOdb, que atualmente traz:

- MySQL

- SQLite

- Postgresql

- MSSQL (Both ODBC and Normal)

- ODBC

- Oracle

Mas caso queira trabalhar com qualquer outro SGBD nao contemplado, apenas baixe o ADODB e utilize sem problema.

Apenas para o SQLite, que é baseado em arquivos, existe algumas diferenças. Veja:

Devemos criar um diretório para o banco "sqlitedb".

```
<?
class dbinfo extends abstractdbinfo
{
    public function __construct()
    {
        $pm = new packagemanager();
        $package_path = $pm->get_package_path(); //physical path to this package
        $this->dbhost = $package_path."/sqlitedb/";
        $this->dbname = "test.sqlite";
        $this->dbtype = "sqlite";
        $this->persist = 1;
    }
}
?>
```

Operações com o SQLite:

//viewreport.class.php

```
<?
class viewreport implements action
{
    public $params;
    public function execute()
    {
        $dao = new DAO(); //instantiate Data Access Object
        $dao->execute("create table students (name varchar(100))");

        $dao->execute('insert into students values("ayesha")');
        $dao->execute('insert into students values("didar")');
        $dao->execute('insert into students values("shahana")');
        $dao->execute('insert into students values("tipoo")');

        $result = $dao->selectBySQL("select * from students");
        $data = array("students"=>$result);
        return array("view_file"=>"reports","data"=>$data);
    }
}
?>
```

Veja agora na camada view:

```
//reports.tpl
<h1>All Students</h1>
{foreach item=student from=$students name=std}
{$smarty.foreach.std.iteration}.{$student.name}<br>
{/foreach}
```

Operações com Bancos de Dados

Inserir registos em tabelas com o framework Zephyr é muito fácil:

Veja a estrutura da tabela student:

Field	Type	Null	Key	Default	Extra
name	varchar(255)				
roll	int(11)				
class	int				

Agora vamos criar o modelo de domínio, que é uma classe simples, contendo todos os campos, exceto os de tipo auto_increment. Então para a classe student:

```
class student
{
    public $name;
    public $roll;
    public $class;
```

```
}
```

Agora vamos criar um arquivo na camada view para entrada dos dados:

```
     <input type="text" id="name">
     <input type="text" id="roll">
     <input type="text" id="class">
     <input type="button" value=" = " onclick="insert_std();" />
```

Veja a função:

```
function insert_std()
{
  serialized = group_serialize("name","roll","class");
  load_action_value('insert_record', serialized,'result');
}
```

Esta função serializa os dados e invoca a ação "insert_record". Veja a ação:

```
<?
load_db_domain("student");

class insertstudent implements action {

    public $params ;

    public function execute()
    {
        $cls =     auto_fill_domain("student");
        $dao = new DAO($cls);
        $query = $dao->insert();
        $students = $dao->selectBySQL("SELECT * FROM   students");
        $data = array("students"=>$students);

        return array("view_file"=>"liststudents", "data"=>$data);
    }

}
?>
```

Alterando e excluindo registros:

```
<?
load_db_domain("student");

class updatestudent implements action {

    public $params ;

    public function execute()
    {
        cls =     auto_fill_domain("student");
        $dao = new DAO($cls);
        $query = $dao->update("roll=1");
        $students = $dao->selectBySQL("SELECT * FROM   students");
        $data = array("students"=>$students);
```

```
        return array("view_file"=>"liststudents", "data"=>$data);
    }
}
?>
```

Faz a tualização "where roll = 1".

Créditos

Lead developer : Hasin Hayder

Javascript and Frontend : Rasiul Kabir

DAO and Helper classes : Prashun Paul

Feedback - Equipe de Desenvolvimento (zephyr@phpxperts.net or zephyr@phpxperts.org)

Forum - http://zephyr-forum.phpxperts.net

Documentação online e offline distribuida juntamente com o pacote para download.

Para maiores detalhes (em inglês), veja a documentação oficial.

A intenção deste pequeno guia é apenas a de facilitar a vida de quem não tem um bom domínio do inglês.

Tradução direto do tutorial original do site por Ribamar FS - ribafs[ ] yahoo.com - 22:06 28/11/2006

FrameWorks/CakePHP

De Wikibooks

Framework CakePHP

Um framework em PHP que se inspirou no famoso Ruby on Rails.

Site Oficial - http://cakephp.org/

Downloads - http://cakephp.org/downloads

Tutorial do Elton - Desenvolva rapidamente utilizando o framework CakePHP - http://www.eltonminetto.net/docs/cake_fisl.pdf

Tutorial do Carlos Silva na Revista PHP - http://www.revistaphp.com.br/artigo.php?id=93

CakePHP Brasil - http://www.cakephp.com.br/

Tutorial - http://www.cakephp.com.br/modules/news/index.php?storytopic=0&storynum=5&start=10

FrameWorks/Symfony

De Wikibooks

Predefinição:Info/Software

Symfony é um framework web escrito em PHP que segue o paradigma MVC (model-view-controller). Disponível sob a Licença MIT, Symfony é um Software livre.

Os criadores do **Symfony** afirmam que a curva de aprendizagem necessária para seu uso proficiente está reduzida a menos de um dia. Ele é projetado para permitir que os desenvolvedores apliquem princípios ágeis do desenvolvimento (tais como DRY, KISS ou XP) e foquem na regras de negócio sem necessitar escrever muitos arquivos de configuração XML, comuns nos frameworks atuais.

Symfony ainda tem como objetivo construir aplicações robustas em contexto empresarial, e dar aos desenvolvedores controle total sobre a configuração: da estrutura de diretórios às bibliotecas externas, quase tudo pode ser personalizado. Para atingir os guidelines de desenvolvimento das empresas, o framework é empacotado com ferramentas adicionais para ajudar os desenvolvedores nos testes, depuração e documentação dos projetos.

Symfony é usado pelo serviço de perguntas e respostas Askeet e por muito mais aplicações, incluindo o Yahoo Bookmarks, aplicação com mais de 20 milhões de usuários.

Referências

Leitura recomendada

- Potencier, Fabien and Zaninotto, François. (2007). *The Definitive Guide to Symfony.* Apress. ISBN 1590597869.

Ligações externas

- Symfony Project - Site oficial do projeto.
- Symfony Brasil - Site da comunidade brasileira.
- symfony-pt - Lista de discussão da comunidade brasileira.

Abstrações de Bancos de Dados/AdoDB

De Wikibooks

Conteúdo

- 1 Introdução
- 2 Exemplo prático para mostrar alguns recursos da ADOdb
- 3 Alguns aliases para a conexão
- 4 Exemplo com as funções RecordCount() e FieldCount()
- 5 Restringir o número de registros recebidos podemos usar o método SelectLimit()
- 6 Obter uma lista dos bancos e tabelas do SGBD
- 7 Quando precisamos executar uma consulta várias vezes
- 8 Uso de transações
- 9 Cache de Consultas
- 10 Exportar um resultset para diversos formatos
- 11 Debugando
- 12 Recordset Filters
- 13 Inserts and Updates
- 14 MetaTypes

Introdução

ADOdb (Active Data Objects DataBase) é uma abstração de bancos de dados para PHP. Também oferece uma versão para Python.

Atualmente suporta diversos SGBDs: MySQL, Oracle, Microsoft SQL Server, Sybase, Sybase SQL Anywhere, Informix, PostgreSQL, FrontBase, Interbase (Firebird and Borland variants), Foxpro, Access, ADO e ODBC.

Via ODBC podemos também conectar com diversos SGBDs: Progress, SQLite, DB2 e outros.

É uma das mais ricas e eficientes abstrações existentes para PHP atualmente.

Caso não se utilize uma abstração no código, ao migrar para outro SGBD o código sofrerá diversas alterações e além disso exigirá o conhecimento das funções de ambos os SGBDs, mas utilizando algo como ADOdb a alteração será apenas nos dados da conexão sem necessidade de conhecer as funções do PHP com os SGBDs.

Exemplo prático para mostrar alguns recursos da ADOdb

Site oficial - http://php.weblogs.com/ADODB

Download - http://adodb.sourceforge.net/#download

Documentações - http://adodb.sourceforge.net/#docs

Este tutorial teve como base os tutoriais do site oficial e também o ótimo tutorial:

PHP Application Development With ADODB

http://www.devshed.com/c/a/PHP/PHP-Application-Development-With-ADODB-part-1/

Banco de Dados para o exemplo: dbbiblioteca

create database dbbiblioteca (com o postgresql mas para usar outro SGBD basta trocar os dados)

Tabela - biblioteca

```
CREATE TABLE biblioteca (
    id serial NOT NULL PRIMARY KEY,
    titulo character(100),
    autor character(45),
    data date
);
```

Vamos inserir alguns registros:

```
INSERT INTO biblioteca (id, titulo, autor, data) VALUES (1, 'Os Sertões',
'Euclides da Cunha','2005-12-25');
INSERT INTO biblioteca (id, titulo, autor, data) VALUES (2, 'Os Lusíadas',
'Camões','2005-12-25');
INSERT INTO biblioteca (id, titulo, autor, data) VALUES (3, 'A Divina Comédia',
'Dante','2005-12-25');
INSERT INTO biblioteca (id, titulo, autor, data) VALUES (4, 'Contos',
'Voltaire','2005-12-25');
INSERT INTO biblioteca (id, titulo, autor, data) VALUES (5, 'O Conde de Monte
Cristo', 'Alexandre Dumas Pai','2005-12-25');
```

Fazer o download e descompactar numa pasta do seu DocumentRoot. Neste exemplo descompactei no raiz.

Alguns aliases para a conexão

MySQL - mysql

PostgreSQL - postgres

Interbase/Firebird - ibase

SQLite - sqlite

Vamos agora criar alguns scripts PHP para acessar o banco usando ADOdb.

Esta parte inicial se repetirá em todos os scripts, portanto apenas farei a citação de "inicial" nos demais scripts

inicial

```
<?php
// Descomentar a linha abaixo para visualizar como plaintext no browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

// create an object instance
// Configurar para uma conexão tipo PostgreSQL
$db = NewADOConnection("postgres"); // MySQL seria "mysql"

// Abrir uma conexão com o banco de dados
// $db->Connect("servidor", "usuario", "senha", "banco")
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Falha
na conexão!");
//final do inicial

<h2>executar a consulta</h2>

$query = "SELECT * FROM biblioteca";
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());

// Iteração através do resultset
// imprimir dados em colunas no formato TÍTULO - AUTOR

while (!$result->EOF){
        echo $result->fields[1] . " - " . $result->fields[2] . "<br>";
        $result->MoveNext(); // Veja que função útil, como também o EOF
                             // (End Of File, enquanto não chegar ao final)
}

<h2>receber e imprimir o número de registros do resultset com muita
simplicidade</h2>
echo "<br>[" . $result->RecordCount() . " registros retornados]<br>";

<h2>Fechar a conexão com o banco</h2>
$db->Close();
```

```
?>
```

ADODB também oferece um número de métodos alternativos para processar um resultset. Por exemplo, você pode receber o resultset como um array associativo indexado de string, onde as chaves são nomes de campos e os valores os correspondentes valores dos campos.

Um exemplo:

inicial - (aqui cole o trecho inicial) referido acima

```
// get resultset as associative array
$ADODB_FETCH_MODE = ADODB_FETCH_ASSOC;

// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->Execute($query) or die("Error in query: $query. " . $db-
>ErrorMsg());

// iterate through resultset
// print column data in format TITLE - AUTHOR
while (!$result->EOF)
{
        echo $result->fields['titulo'] . " - " . $result->fields['autor'] .
"<br>";
        $result->MoveNext();
}

// get and print number of rows in resultset
echo "<br>[" . $result->RecordCount() . " registros retornados]<br>";

// close database connection
$db->Close();

?>
```

Você pode usar o método GetAll() no lugar do Execute() que retorna o resultset completo com um array bidimensional de pares campo-valor. Este array pode então ser processado com um simples "foreach" ou um loop "for".

Um exemplo:

inicial

```
// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->GetAll($query) or die("Error in query: $query. " . $db-
>ErrorMsg());

// clean up
$db->Close();

// uncomment the following line to see the returned array.
// print_r($result);
```

```
// iterate through resultset
// print column data in format TITLE - AUTHOR
foreach ($result as $row)
{
        echo $row[1] . " - " . $row[2] . "<br>";
}

// get and print number of rows in resultset
echo "<br>[" . sizeof($result) . " registros retornados]<br>";

?>
```

Exemplo com as funções RecordCount() e FieldCount()

inicial

```
// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->Execute($query) or die("Error in query: $query. " . $db-
>ErrorMsg());

// get and print number of rows in resultset
echo $result->RecordCount() . " registros retornados<br>";

// get and print number of fields in resultset
echo $result->FieldCount() . " campos retornados<br>";

// clean up
$db->Close();

?>
```

Podemos obter informações sobre cada campo com o método FetchField(), que retorna um objeto contendo informações detalhadas sobre as propriedades dos campos, incluindo seus nomes e tipos. Um exemplo:

inicial

```
// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->Execute($query) or die("Error in query: $query. " . $db-
>ErrorMsg());

// get field information
for($x=0; $x<$result->FieldCount(); $x++) {
        print_r($result->FetchField($x));
}

// clean up
$db->Close();
?>
```

Para a execução de consultas INSERT em tabela contendo chave primária com auto-incremento podemos obter o último ID gerado do auto-incremento usando o método Insert_ID(). Exemplo:

inicial

```
// execute query
$titulo = $db->qstr("It's Not Me, It's You!");
$autor = $db->qstr("J. Luser");
$query = "INSERT INTO biblioteca (titulo, autor) VALUES ($titulo, $autor)";
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());

// print auto-generated ID
if ($result)
{
   echo "O último ID inserido foi " . $db->Insert_ID();
}

// clean up
$db->Close();
?>
```

Obs.: o método qstr() é usado para "escapar" caracteres especiais em consultas com strings.

Quando estamos utilizando consultas que afetam registros de tabelas, como insert, delete ou update o método Affected_Rows() retorna o número de registros afetados.

inicial

```
// execute query
$query = "DELETE FROM biblioteca WHERE author = 'Euclides da Cunha'";
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());

// return number of affected rows
if ($result)
{
   echo $db->Affected_Rows() . " registros excluídos";
}

// clean up
$db->Close();
?>
```

Restringir o número de registros recebidos podemos usar o método SelectLimit()

inicial

```
// execute query
// receber 2 registros, iniciando do terceiro, ou seja, o terceiro e o quarto
registros
$query = "SELECT * FROM biblioteca";
$result = $db->SelectLimit($query, 2, 3) or die("Erro na consulta: $query. " .
$db->ErrorMsg());

// iterate through resultset
while (!$result->EOF)
{
        echo $result->fields[1] . " - " . $result->fields[2] . "<br>";
        $result->MoveNext();
}

// clean up
$db->Close();
?>
```

Alerta: Cuidado com o copiar e colar. Este exemplo acima e vários outros acusaram erro ao executar.

Normalmente o erro era na linha (echo $result->fields[1] . " - " . $result->fields[2] . "
";).

O original estava no Write do OpenOffice. Normalmente apenas excluo os espaços antes do echo e tá resolvido ou então redigito apenas a linha do erro.

Algumas vezes precisei remover alguns espaços após o ponto e vírgula.

Obter uma lista dos bancos e tabelas do SGBD

Através dos métodos MetaDatabases() e MetaTables().

inicial

```
// get database list
echo "Bancos:<br>";
foreach($db->MetaDatabases() as $d){
        echo "=> $d<br>";
}

// get table list
echo "<br>Tabelas no banco atual:<br>";
foreach($db->MetaTables() as $table)
{
```

```
        echo "=> $table<br>";
}

// clean up
$db->Close();
?>
```

Quando precisamos executar uma consulta várias vezes

Como por exemplo diversos INSERTs. O ADOdb conta com diversos recursos úteis. Vejamos:

inicial

```
// prepara a consulta e a deixa em "banho maria", sem a executar ainda
$query = $db->Prepare("INSERT INTO biblioteca (id, titulo, autor) VALUES (?,
 ?, ?)");

// ler lista titulo-autor do arquivo CSV
$data = file("lista.txt");

// iteração através de cada linha do arquivo
foreach ($data as $l){
        // separa com vírgulas
        $arr = explode(",", $l);
        // insere os valores na consulta preparada anteriormente
        $result = $db->Execute($query, array($arr[0], '$arr[1]', '$arr[2]')) or
die("Erro na consulta: $query. " . $db->ErrorMsg());
}

// clean up
$db->Close;
?>
```

Atentar para o fato de que o prepare apenas deixa a consulta pronta e o execute finaliza a execução.

Isto melhora muito o desempenho quando temos muitas consultas a realizar.

Uso de transações

Caso o SGBD tenha suporte:

inicial

```
// turn off auto-commit
// begin transaction block
$db->BeginTrans();

// first query
$query = "INSERT INTO biblioteca (titulo, autor) VALUES ('Titulo A', 'Autor B')";
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());
```

```
// use ID from first query in second query
if ($result)
{
  $id = $db->Insert_ID();
   $query = "INSERT INTO purchase_info (id, price) VALUES ($id, 'USD 39.99')";
   $result = $db->Execute($query) or die("Error in query: $query. " . $db-
>ErrorMsg());
}

// if no failures
if ($result)
{
   // commit
   $db->CommitTrans();
}
// else rollback
else
{
   $db->RollbackTrans();
}

// clean up
$db->Close;
?>
```

Cache de Consultas

ADOdb tembém oferece suporte para cache de consultas. Que oferece um excelente ganho de desempenho, principalmente nos casos em que precisamos executar uma mesma consulta diversas vezes.

Para ver a diferença abaixo temos uma consulta normal:

inicial

```
// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->Execute($query) or die("Erroo na consulta: $query. " . $db-
>ErrorMsg());

// iterate through resultset
// print column data in format TITLE - AUTHOR
while (!$result->EOF)
{
        echo $result->fields[1] . " - " . $result->fields[2] . "<br>";
        $result->MoveNext();
}

// get and print number of rows in resultset
echo "<br>[" . $result->RecordCount() . " registros retornados]<br>";

// close database connection
$db->Close();
?>
```

Agora usando o cache de consulta:

inicial

```
// execute query
$query = "SELECT * FROM biblioteca";
$result = $db->CacheExecute(300,$query) or die("Erro na consulta: $query. " .
$db->ErrorMsg());

// iterate through resultset
// print column data in format TITLE - AUTHOR
while (!$result->EOF)
{
        echo $result->fields[1] . " - " . $result->fields[2] . "<br>";
        $result->MoveNext();
}

// get and print number of rows in resultset
echo "<br>[" . $result->RecordCount() . " registros retornados]<br>";

// close database connection
$db->Close();

?>
```

Algo que dá um bom trabalho é criar um select que seja populado de uma tabela. O menu drop-down. O ADOdb tem um método especificamente para esta finalidade.

```
<html>
<head></head>
<body>
<?php
// include the ADODB library
include("adodb/adodb.inc.php");

// create an object instance
// configure it for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

// execute query
$query = "SELECT titulo, id FROM biblioteca"; // O primeiro será exibido e o
segundo armazenará o resultado
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());

// print HTML menu
print $result->GetMenu("biblioteca", '', false); // primeiro parâmetro é o nome
do select

// close database connection
$db->Close();
?>
```

```
</body>
</html>
```

Exportar um resultset para diversos formatos

```
- texto separado por vírgula
```

- texto separado por tabulação

- tabela HTML

Estes recursos não fazem parte da classe ADODB, portanto precisaremos importar outras classes.

```
<?php
// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

// include conversion functions
include("adodb/toexport.inc.php");

// create an object instance
// configure library for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

// execute query
$query = "SELECT title, id FROM library";
$result = $db->Execute($query) or die("Error in query: $query. " . $db-
>ErrorMsg());

// return a CSV string
echo rs2csv($result);

// close database connection
$db->Close();

?>
```

Podemos suprimir a primeira linha (nomes dos campos), adicionando um parâmetro para a chamada de rs2csv():

// return a CSV string

echo rs2csv($result, false);

Formatando com separador tab: Apenas troque a função (método) de rs2csv() para rs2tab().

Formatando a saída para tabela HTML: Para esta precisamos de outro importe, confira:

```
<html>
<head></head>
<body>
<?php
// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

// include conversion functions
include("adodb/tohtml.inc.php");

// create an object instance
// configure it for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

// execute query
$query = "SELECT titulo, id FROM biblioteca";
$result = $db->Execute($query) or die("Erro na consulta: $query. " . $db-
>ErrorMsg());

// return a table
echo rs2html($result);

// close database connection
$db->Close();

?>
</body>
</html>
```

Formatando Data de Saída

```
<?php
// Select a table, display the first two columns.
// If the second column is a date or timestamp, reformat the date to Brazilian
d/m/Y.

// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

// create an object instance
// configure library for a PostgreSQL connection
```

```
$db = NewADOConnection("postgres");
// Antes alterar a tabela adicionando o campo data:
// \c dbbiblioteca
// ALTER TABLE biblioteca ADD COLUMN data date;

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

$recordSet = &$db->Execute('select id,data from biblioteca');
if (!$recordSet)
        print $db->ErrorMsg();
else
while (!$recordSet->EOF) {
        $fld = $recordSet->FetchField(1);
        $type = $recordSet->MetaType($fld->type);

        if ( $type == 'D' || $type == 'T')
                print $recordSet->fields[0].' '.
                        $recordSet->UserDate($recordSet-
>fields[1],'d/m/Y').'<BR>';
        else
                print $recordSet->fields[0].' '.$recordSet->fields[1].'<BR>';

        $recordSet->MoveNext();
}
$recordSet->Close(); # optional
$db->Close(); # optional

?>
```

Exporting in CSV or Tab-Delimited Format

```
<?php
// Exporting in CSV or Tab-Delimited Format

// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

include_once('adodb/toexport.inc.php');

// include the ADODB library
include("adodb/adodb.inc.php");

// create an object instance
// configure library for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

$rs = $db->Execute('select titulo as "Título", autor as "Autor" from
biblioteca');

print "<pre>";
print rs2csv($rs); # return a string, CSV format
```

```
print '<hr>';

$rs->MoveFirst(); # note que alguns databases não suportam MoveFirst
print rs2tab($rs,false); # return a string, tab-delimited
# false == suppress field names in first line
print '<hr>';
$rs->MoveFirst();
rs2tabout($rs); # send to stdout directly (there is also an rs2csvout function)
print "


";

$rs->MoveFirst(); $fp = fopen($path, "w"); if ($fp) {

 rs2csvfile($rs, $fp); # write to file (there is also an rs2tabfile function)
 fclose($fp);


}

?>
```

INSERT com Arquivo SCV

```
<?php
// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

// create an object instance
// configure library for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

// prepare query
$query = $db->Prepare("INSERT INTO biblioteca (id, titulo, autor) VALUES (?, ?)
");

// read title-author list in from CSV file
$data = file("lista.txt");

// iterate through each line in file
foreach ($data as $l){
        // split on comma
        $arr = explode(",", $l);
        // insert values into prepared query
        $result = $db->Execute($query, array($arr[0], $arr[1])) or die("Error in
query: $query. " . $db->ErrorMsg());
}
```

```
// clean up
$db->Close;
echo "Arquivo CSV inserido com sucesso!";
?>
```

Arquivo lista.txt

```
Sandálias do Pescador, Morris West
Ana Karenina, Leon Tolstoi
Terras do Sem Fim, Jorge Amado
Helena, Machado de Assis
Paginação com muita Simplicidade

<?php
// Paginação com muita simplicidade

// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
include("adodb/adodb.inc.php");

include_once('adodb/adodb-pager.inc.php');
session_start();


// create an object instance
// configure library for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");

$sql = "select * from biblioteca ";

$pager = new ADODB_Pager($db,$sql);
$pager->Render($rows_per_page=5);

?>
<pre>


<h2>Tratamento de Strings</h2>

<pre>
<?php
// Insert a row to the Orders table containing dates and strings that need to be
// quoted before they can be accepted by the database, eg: the single-quote in
the word John's.

// uncomment this to see plaintext output in your browser
// header("Content-Type: text/plain");

// include the ADODB library
```

```
include("adodb/adodb.inc.php");

// create an object instance
// configure library for a PostgreSQL connection
$db = NewADOConnection("postgres");

// open connection to database
$db->Connect("localhost", "postgres", "postabir", "dbbiblioteca") or die("Unable
to connect!");
$autor = $db->qstr("John's Old Shoppe"); // Tratamento de string com qstr antes
de inserir no banco

$sql = "insert into biblioteca (titulo,id,data,autor) ";
$sql .= "values ('Teste2',36,'2006-10-10',$autor)"; // Veja o original para
detalhes

if ($db->Execute($sql) === false) { // Só insere se não ocorrer erro
        print 'error inserting: '.$db->ErrorMsg().'<BR>';
}

echo "Inserido com sucesso!";
?>
```

Debugando

```
<?
include('adodb.inc.php');          # load code common to ADOdb
$conn = &ADONewConnection('access');        # create a connection
$conn->PConnect('northwind');   # connect to MS-Access, northwind dsn
$shipto = $conn->qstr("John's Old Shoppe");
$sql = "insert into orders (customerID,EmployeeID,OrderDate,ShipName) ";
$sql .= "values ('ANATR',2,".$conn->FormatDate(time()).",$shipto)";
$conn->debug = true;
if ($conn->Execute($sql) === false) print 'error inserting';
?>
```

Conectando-se a dois bancos de dados de SGBDs diferentes ao mesmo tempo

```
<?
include('adodb.inc.php');        # load code common to ADOdb
$conn1 = &ADONewConnection('mysql');  # create a mysql connection
$conn2 = &ADONewConnection('oracle');  # create a oracle connection

$conn1->PConnect($server, $userid, $password, $database);
$conn2->PConnect(false, $ora_userid, $ora_pwd, $oraname);

$conn1->Execute('insert ...');
$conn2->Execute('update ...');
?>
```

Recordset Filters

Sometimes we want to pre-process all rows in a recordset before we use it. For example, we want to ucwords all text in recordset.

```
include_once('adodb/rsfilter.inc.php');
include_once('adodb/adodb.inc.php');
// ucwords() every element in the recordset
function do_ucwords(&$arr,$rs)
{
        foreach($arr as $k => $v) {
                $arr[$k] = ucwords($v);
        }
}

$db = NewADOConnection('mysql');
$db->PConnect('server','user','pwd','db');

$rs = $db->Execute('select ... from table');
$rs = RSFilter($rs,'do_ucwords');
```

Executing the SQL

```
$result = $db->Execute("SELECT * FROM employees");
if ($result === false) die("failed");
```

Inserts and Updates

Let's say you want to insert the following data into a database.

```
ID = 3
TheDate=mktime(0,0,0,8,31,2001) /* 31st August 2001 */
Note= sugar why don't we call it off

$sql = "INSERT INTO table (id, thedate,note) values ("
   . $ID . ','
   . $db->DBDate($TheDate) .','
   . $db->qstr($Note).")";
$db->Execute($sql);
```

MetaTypes

You can find out more information about each of the fields (I use the words fields and columns interchangebly) you are selecting by calling the recordset method FetchField($fieldoffset). This will return an object with 3 properties: name, type and max_length.

For example:

$recordset = $conn->Execute("select adate from table");

$f0 = $recordset->FetchField(0);

Then $f0->name will hold 'adata', $f0->type will be set to 'date'. If the max_length is unknown, it will be set to -1.

One problem with handling different databases is that each database often calls the same type by a different name. For example a timestamp type is called datetime in one database and time in another. So ADODB has a special MetaType($type, $max_length) function that standardises the types to the following:

```
C: character and varchar types
X: text or long character (eg. more than 255 bytes wide).
B: blob or binary image
D: date
T: timestamp
L: logical (boolean)
I: integer
N: numeric (float, double, money)
```

In the above date example,

```
$recordset = $conn->Execute("select adate from table");
$f0 = $recordset->FetchField(0);
$type = $recordset->MetaType($f0->type, $f0->max_length);
print $type; /* should print 'D' */
```

Trabalhando em PHP com/Formulários

De Wikibooks

Excluir Registros através de CheckBox

Temos a seguinte situação: ao listar registros de uma tabela queremos selecionar alguns deles através de um checkbox e excluir os selecionados.

<?php /* Banco - excluir_varios Tabela CREATE TABLE `produtos` (

```
 `id` int(11) NOT NULL,
 `produto` char(45) default NULL,
 `categoria` char(45) default NULL,
 PRIMARY KEY  (`id`)
```

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

- /

$conexao = mysql_connect('localhost','root',*);* mysql_select_db('excluir_varios',$conexao); $consultar = "SELECT * FROM produtos ORDER BY id"; $resultado = mysql_query($consultar, $conexao);

if(mysql_num_rows($resultado) != 0){ echo "<form name='frmExcluir' method='post' action='excluir_varios.php'>";

echo "";

while($linha = mysql_fetch_row($resultado)){

echo "";

}

echo ""; echo "

	Produto:	Categoria:
<input type='checkbox' name='id[]' value='$linha[0]'>	$linha[1]	$linha[2]
<input type='submit' name='excluir' value='Excluir!'>		

</form>";

}else{ echo "Nenhum registro foi encontrado!"; } if(isset($_POST['id'])){ $opcoes = $_POST['id']; $opcoes_text = implode(", ", $opcoes); $strexcluir = "DELETE FROM produtos WHERE id in (" . $opcoes_text . ")"; mysql_query($strexcluir, $conexao) or die("Ocorreu algum erro"); }else{ echo "É

necessário escolher quem será excluído
"; echo "Voltar"; } ?>

Manipulando dados de formulários com PHP do Alfred Reinold Baudisch

Parte 1

Parte 2

Formulários acessíveis à prova de spam

[1]

Trabalhando em PHP com/URLs

De Wikibooks

11.2 - Trabalhando com URL em PHP

Passando Parâmetros pela URL

Primeiro (âncora)

Link

```
arquivo.php
$par1=$_GET['parametro1'];
$par2=$_GET['parametro2'];
$par3=$_GET['parametro3'];
```

Segundo (action de form)

```
<form name=frm method=post
action="arquivo2.php?parametro1=valor1&parametro2=valor2">
...
```

arquivo2.php

```
$par1=$_POST['parametro1'];
$par2=$_POST['parametro1'];
```

Terceiro (URL)

http://localhost/teste.php?parametro1=valor1

teste.php

$par1=$_GET['parametro1'];

Quarto (location no javascript)

```
<?php
// Já vindo de outro script, chamado via POST
$a = $_POST['a'];

?>
<script>
if(confirm("Confirma?")){
        location="vai.php?a='<?=$a?>' ";
}else{
        location='volta.php';
}
```

```
</script>
```

Reconstruct URL string in PHP

```
 // find out the domain:
 $domain = $_SERVER['HTTP_HOST'];
 // find out the path to the current file:
 $path = $_SERVER['SCRIPT_NAME'];
 // find out the QueryString:
 $queryString = $_SERVER['QUERY_STRING'];
 // put it all together:
 $url = "http://" . $domain . $path . "?" . $queryString;
 echo "The current URL is: " . $url . "
";

 // An alternative way is to use REQUEST_URI instead of both
 // SCRIPT_NAME and QUERY_STRING, if you don't need them seperate:
 $url2 = "http://" . $domain . $_SERVER['REQUEST_URI'];
 echo "The alternative way: " . $url2;
```

Do site - http://snippets.dzone.com/posts/show/4054

Trabalhando em PHP com/Funções

De Wikibooks

Funções são blocos de código, conjuntos de instruções, que ficam quietos, sem vida, até que sejam chamadas em algum lugar do código. Funções reduzem o trabalho dedigitação, o tamanho dos scripts em geral, os erros, reaproveitando código, facilitando e organizando o trabalho de desenvolvimento.

Conteúdo

- 1 Exemplos de funções definidas pelo usuário
- 2 Passando Argumentos através de Funções
- 3 Por Valor
- 4 Por Referência
- 5 Otimização do tempo de execução
- 6 Trabalhando com Funções OB

Exemplos de funções definidas pelo usuário

```
function quadrado($numero) {
        print $numero*$numero;
}
<pre>

Executando:

quadrado(6); // Saída: 36


<h2>Variáveis globais</h2>
<pre>
$var1 = 5;
function testeGlobal1(){
        $var1 = 1;
        print "<br>Valor de \$var1: $var1";
}
echo testeGlobal1();

echo "Variável externa acessível na função<br>";

$var2 = 10;
function testeGlobal2(){
        global $var2;
        print "<br>Valor de \$var2 $var2";
}
```

```
echo testeGlobal2();

$var5 = 15;
function testeGlobal5(){
        $var5 = 5;
    print "<br><br>A variável global vale $GLOBALS[var5], ";
    print "Já a variável local vale $var5<br><br>";
}

testeGlobal5();

function cliente($codigo, $idade = 18){
        print "Código = $codigo, Idade = $idade";
}

cliente(1);  //Exibirá: Código = 1, Idade = 18

function cubo($num){
        return ($num*$num*$num);
}

$var1 = 2 * cubo(5);echo "<br>".$var1;

echo "Variáveis estáticas<br>";
function contador(){
        static $x = 0;
        return $x++;
}
echo "<br>";
echo contador();echo contador();echo contador();
//A saída será: 012

function contador2(){
        $x = 0;
        return $x++;
}
echo "<br>";
echo contador2();echo contador2();echo contador2();
//A saída será: 000.

  function staticfunction() {
   static $count = 0;
   $count++;
        if ($count==1){
       echo "A Função foi executada $count vez<br>";
        }else{
           echo "A Função foi executada $count vezes<br>";
        }
  }

  for($i = 0; $i < 5; $i++) {
   staticfunction();
  }

function Teste1()
{
    static $a = 0;
```

```
    echo $a;
    $a++;
}

for($x=0;$x<=10;$x++){
        echo Teste1()." ";
}

echo "<br>";

echo "Recursiva<br>";
function Teste()
{
    static $count = 0;

    $count++;
    echo $count." ";
    if ($count < 10) {
        Teste ();
    }
    $count--;
}

Teste();

echo "Declarando variáveis static<br>";
function foo(){
    static $int = 0;          // correro
   // static $int = 1+2;        // errado (é uma expressão)
   // static $int = sqrt(121);  // errado  (é uma expressão também)

    $int++;
    echo $int;
}


function aumentoSalario($sal, $perc=5){
        $salario = $sal * $perc/100;
        echo $salario;
}
echo "<br>Aumento: " . aumentoSalario(1956);

function redirecionar($url){
        header("Location: $url");
}
echo "<br>";
redirecionar("http://ribafs.phpnet.us/");
echo "<br>";

echo "Retornar mais de um valor de uma função: usa-se arrays e list()
array() retorna e list() exibe <br>";

//Exemplo:

function recebe_hoje(){
        $data_abreviada=date("d/m/Y");
        $data_extensa=date("l, d F \d\e Y");
```

```
        return array($data_abreviada, $data_extensa);
}

list($data_abreviada, $data_extensa)=recebe_hoje();
        print $data_extensa;
echo "<br>";
        print $data_abreviada;


echo "Declaração dinâmica de função:<br>";

if ($f == 1){
        function f1(){
                echo "funcao1";
        }
}else{
        function f2(){
                echo "funcao2";
        }
}

echo "Retornando o número de argumentos de uma função:<br>";

function ret_args_funcao() {
    $numargs = func_num_args();
    echo "Número de argumentos: $numargs<br>\n";
    if ($numargs >= 2) {
    echo "Segundo argumento vale : " . func_get_arg (1) . "<br>\n";
    }
    $arg_list = func_get_args();
    for ($i = 0; $i < $numargs; $i++) {
    echo "Argumento $i vale: " . $arg_list[$i] . "<br>\n";
    }
}

ret_args_funcao (1, 2, 3);
```

Passando Argumentos através de Funções

O default é 'por valor', que passa uma cópia do valor da variável.

Também podemos passar 'por referência', onde se passa o endereço da própria variável. Quando atribuímos uma variável a outra passando como referência, não fazemos cópia alguma, mas sim passamos o endereço da variável original, portanto qualquer alteração nesta referência refletirá na variável original.

ByRef é mais eficiente em termos de memória ao lidar com variáveis e arrays grandes e também permite alterar o valor da variável original, o que não acontece com o ByVal, mas a vantagem de desempnho somente é percebida em grandes arrays em grandes loops.

Para maior segurança setar:

allow_call_time_pass_reference no php.ini

Impede a passagem de valores por referência nas chamadas, mas permite somente na definição das funções.

$var1 = & $var2;

Agora ambas apontam para o mesmo endereço e valor.

Reter valor de variáveis entre chamadas (static)

(Guarda o valor da última chamada até o final da execução do script, tantas vezes quantas a função for chamada).

Exemplo:

```
$valor = 4;
$ref = &$valor;

$ref = 3;

$valor = 4;
$ref = &$valor; // Aqui tornamos ambas as variáveis com o mesmo endereço
                // O que alterarmos em $ref alteramos em $valor

$ref = 3;       // Com isso também alteramos $valor para 3, veja abaixo.
echo $valor . "<br>";

$valor=0;          // Com isso também alteramos $ref para 0, veja abaixo.

echo $ref;
```

Por Valor

```
function val_subtracao($num1, $num2){
        if($num1 < $num2){
                die("Números negativos");
        }else{
                $return_result=0;
                while($num1 > $num2){
                        $num1 = $num1 - 1;
                        $return_result = $return_result + 1;
                }
        }
        return($return_result);
}

$primeiro_op=493;
$segundo_op=355;
```

```
$resultado1 = val_subtracao($primeiro_op, $segundo_op);
print ("Resultado1 é $resultado1<br>");
$resultado2 = val_subtracao($primeiro_op, $segundo_op);
print("Resultado2 é $resultado2<br>");
```

Por Referência

```
function subtracao_ref(&$num1, &$num2){
        if($num1 < $num2){
                die("Números negativos");
        }else{
                $return_result=0;
                while($num1 > $num2){
                        $num1 = $num1 - 1;
                        $return_result = $return_result + 1;
                }
        }
        return($return_result);
}

$primeiro_op=493;
$segundo_op=355;
$resultado1 = subtracao_ref($primeiro_op, $segundo_op);
print ("<br><br>Resultado1 é $resultado1<br>");
$resultado2 = subtracao_ref($primeiro_op, $segundo_op);
print("Resultado2 é $resultado2<br>");

echo "Agora, se se nós executarmos exatamente a mesma chamada da subtração como
fizemos a primeira vez, receberemos a saída:
resultado1 é 138 e resultado2 é 0";


/*
Sugestão de chamada de função:

if (nome_funcao($argumento){
        ....
        ....
}else{
        ....
}

*/
?>

<?php
// Retorna o tipo e o valor de variável
function ss_as_string (&$thing, $column = 0) {
    if (is_object($thing)) {
        return ss_object_as_string($thing, $column);
    }
    elseif (is_array($thing)) {
        return ss_array_as_string($thing, $column);
    }
    elseif (is_double($thing)) {
```

```
        return "Double(".$thing.")";
    }
    elseif (is_long($thing)) {
        return "Long(".$thing.")";
    }
    elseif (is_string($thing)) {
        return "String(".$thing.")";
    }
    else {
        return "Unknown(".$thing.")";
    }
}

// Retorna o tipo e o valor de array
function ss_array_as_string (&$array, $column = 0) {
    $str = "Array(<BR>\n";
    while(list($var, $val) = each($array)){
        for ($i = 0; $i < $column+1; $i++){
            $str .= "    ";
        }
        $str .= $var.' ==> ';
        $str .= ss_as_string($val, $column+1)."<BR>\n";
    }
    for ($i = 0; $i < $column; $i++){
        $str .= "    ";
    }
    return $str.')';
}

// Retorna o tipo e o valor de objeto
function ss_object_as_string (&$object, $column = 0) {
    if (empty($object->classname)) {
        return "$object";
    }
    else {
        $str = $object->classname."(<BR>\n";
        while (list(,$var) = each($object->persistent_slots)) {
            for ($i = 0; $i < $column; $i++){
                $str .= "    ";
            }
            global $$var;
            $str .= $var.' ==> ';
            $str .= ss_as_string($$var, column+1)."<BR>\n";
        }
        for ($i = 0; $i < $column; $i++){
            $str .= "    ";
        }
        return $str.')';
    }
}

$var="Riba";
echo ss_as_string($var);
//echo ss_as_string($GLOBALS);
```

Otimização do tempo de execução

```
function ss_timing_start ($name = 'default') {
    global $ss_timing_start_times;
    $ss_timing_start_times[$name] = explode(' ', microtime());
}

function ss_timing_stop ($name = 'default') {
    global $ss_timing_stop_times;
    $ss_timing_stop_times[$name] = explode(' ', microtime());
}

function ss_timing_current ($name = 'default') {
    global $ss_timing_start_times, $ss_timing_stop_times;
    if (!isset($ss_timing_start_times[$name])) {
        return 0;
    }
    if (!isset($ss_timing_stop_times[$name])) {
        $stop_time = explode(' ', microtime());
    }
    else {
        $stop_time = $ss_timing_stop_times[$name];
    }
    // do the big numbers first so the small ones aren't lost
    $current = $stop_time[1] - $ss_timing_start_times[$name][1];
    $current += $stop_time[0] - $ss_timing_start_times[$name][0];
    return $current;
}

?>
```

Trabalhando com Funções OB

- Artigo do Adriano Oliveira Gonçalves na Revista do PHP

Trabalhando em PHP com/Constantes mágicas e super globais

De Wikibooks

Conteúdo

- 1 Variáveis do servidor
- 2 Exemplos
- 3 Constantes Mágicas
- 4 Tutorial Sobre as Super Globais

Variáveis do servidor

$_SERVER

Este é um array (vetor) 'superglobal', ou automaticamente global. Isto significa que ele é disponível em todos os escopos (níveis) de um script. Você não precisa fazer um: ... global $_SERVER; ... para poder acessá-lo dentro de funções ou métodos, como era necessário com $HTTP_SERVER_VARS. O array superglobal $_SERVER existe em qualquer sessão PHP e já contém um conjunto de chaves (índices) pré definidos e valorados. Os índices mais importantes são:

'REQUEST_URI'

O URI fornecido para acessar a página atual, por exemplo, '/index.html'.

'SCRIPT_NAME'

Contém o caminho completo do script atual. Útil para páginas que precisam apontar para elas mesmas (dinamicamente). A constante __FILE__ contém o caminho completo e nome do arquivo (mesmo incluído) atual.

'PHP_SELF'

O nome do arquivo do script atualmente em uso, relativo ao document root. Por exemplo, $_SERVER['PHP_SELF'] em um script com o endereço http://example.com/test.php/foo.bar pode ser /test.php/foo.bar. A constante __FILE__ contém o caminho completo e nome do arquivo (mesmo incluído) atual.

Se estiver rodando o PHP em linha de comando, esta variável não está disponível.

'SERVER_NAME'

O nome host do servidor onde o script atual é executado. Se o script está rodando em um host virtual, este será o valor definido para aquele host virtual.

'REQUEST_METHOD'

Contém o método de request utilizando para acessar a página. Geralmente 'GET', 'HEAD', 'POST' ou 'PUT'.

'QUERY_STRING'

A query string (string de solicitação), se houver, pela qual a página foi acessada.

'DOCUMENT_ROOT'

O diretório raiz sob onde o script atual é executado, como definido no arquivos de configuração do servidor.

'SCRIPT_FILENAME'

O caminho absoluto o script atualmente em execução.

Nota: Se o script for executado pela CLI com um caminho relativo, como file.php ou ../file.php, $_SERVER['SCRIPT_FILENAME'] irá conter o caminho relativo especificado pelo usuário.

Exemplos

```
$current_script = dirname($_SERVER['SCRIPT_NAME']);

$current_path  = dirname($_SERVER['SCRIPT_FILENAME']);

$request_uri = $_SERVER['REQUEST_URI'];

// Pick the predefined variable that works on your server
     return $_ENV['SCRIPT_URL'];

$_SERVER['QUERY_STRING'])

      $sPathPS = $_SERVER[PHP_SELF];
       $sPathFS = __FILE__;

echo 'http';
if($_SERVER['HTTPS']=='on'){echo 's';}
echo '://'.$_SERVER['SERVER_PORT'].$_SERVER['SCRIPT_NAME'];
if($_SERVER['QUERY_STRING']>' '){echo '?'.$_SERVER['QUERY_STRING'];}
```

Constantes Mágicas

```
__LINE__         A linha atual do script.
__FILE__        O caminho completo e nome do arquivo. Se utilizado dentro de um
```

```
include, o nome do arquivo incluído será retornado.
__FUNCTION__    O nome da função (Acrescentado no PHP 4.3.0). A partir do PHP 5
esta constante retorna o nome da função como ela foi
   declarada (sensível a maiúsculas e minúsculas). No PHP 4 sempre retorna o nome
em minúsculas.
__CLASS__       O nome da classe (Acrescentado no PHP 4.3.0). A partir do PHP 5
esta constante retorna o nome da função como ela foi
   declarada (sensível a maiúsculas e minúsculas). No PHP 4 sempre retorna o nome
em minúsculas.
__METHOD__      O nome do método de classe. (Acrescentado no PHP 5.0.0). O nome
do método é retornado como foi declarado (sensível a
   maiúsculas e minúsculas).

if (realpath(__FILE__) == realpath($_SERVER['SCRIPT_FILENAME'])) {
  exit;
}
```

Tutorial Sobre as Super Globais

Do Pedro Henrique da Silva Gregório no Vivaolinux -

Trabalhando com Session em PHP

Crie um pequeno aplicativo com os seguintes arquivos. Copie os arquivos para um diretório no seu servidor web e chame o index pelo browser:

index.php

```
<?
session_start();
?>
<center><h1>Trabalhando com SESSION em PHP</h1>
Podemos preservar valores de variáveis enquanto durar uma sessão do browser
através do uso de SESSION.<br>
Para isso devemos startar a sessão em cada página que desejamos usar esta
variável com<br><br>
sesssion_start();<br><br>
Lembrando que esta função deve vir antes de qualquer comando que mande algo para
a tela, caso<br>
o session esteja configurado para usar cookie.<br>
Na primeira página deve ter um formulário com algum campo que devemos usar no
session.<br>
Experimente gravar a URL de uma das páginas internas e acessar diretamente
(http://localhost/session3)
<br>Primeiro feche todas as seções do browser e depois abra o browser com essa
URL.<br>
<br>
Veja que SESSION é muito bom para preservar o valor de variáveis entre páginas de
um site numa seção.<br>
Portanto seu uso é muito útil quando pretendemos autenticas os visitantes de
todas as as páginas de um site.<BR>
Como também para outros usos em que pretendemos reaproveitar o valor de variáveis
(algo como global).<BR>
Acompanhe este exemplo para ver detalhes.<br><br><br>

<form method=post action=session2.php>
        Login<input type=text size=8 name=login><br>
        <input type=submit value=Enviar>
</form></center>
```

destruir.php

```
<?
session_start();
if (isset($_SESSION['login'])){
  session_unset(); // Eliminar todas as variáveis da sessão
  session_destroy(); // Destruir a sessão
  echo "Entre. Sessão Destruida. <a href=session3.php>Session3</a><br>";
} else {
  echo "Acesso não autenticado!";
}
?>
```

session2.php

```
<?
session_start();
$_SESSION['login']=$_POST['login'];

if (isset($_SESSION['login'])){
        echo "Entre. Session2. <a href=session3.php>Session3</a>";
} else {
        echo "Acesso não autenticado!";
}
?>
```

session3.php

```
<?
session_start();
if (isset($_SESSION['login'])){
  echo "Entre. Session3. <a href=session4.php>Session4</a><br>";
  echo "<a href=destruir.php>Drestruir Sessão</a>";
} else {
  echo "Acesso não autenticado!";
}
?>
```

session4.php

```
<?
session_start();
if (isset($_SESSION['login'])){
        echo "Entre. Session4. <a href=session5.php>Session5</a>";
} else {
        echo "Acesso não autenticado!";
}
?>
```

session5.php

```
<?
session_start();
if (isset($_SESSION['login'])){
        echo "Entre. Session5. <a href=index.php>Index</a><br><br>";
        echo "Informações: <br>ID da Sessão: <b>" . session_id() .
        "</b><br>Variável mantida pela SuperGlobal \$_SESSION: <b>" .
$_SESSION['login'];
} else {
        echo "Acesso não autenticado!";
}
?>
```

Capítulo sobre Session na Documentação Oficial do PHP

http://www.php.net/manual/pt_BR/ref.session.php

Artigos Sobre Session em PHP

http://www.revistaphp.com.br/print.php?id=79

http://www.scriptbrasil.com.br/forum/lofiversion/index.php/t79221.html

http://www.htmlstaff.org/ver.php?id=5353

http://maozinhadaweb.blogspot.com/2007/05/tutorial-de-php-parte-3-sesses-e.html

http://www.devshed.com/c/a/PHP/Sessions-and-Cookies/

Trabalhando em PHP com/Arquivos

Conteúdo

- 1 Abrir arquivo
- 2 Gravar em Arquivo
- 3 Ler Arquivo
- 4 fgets()
- 5 fgetc()
- 6 Ler Todo um Arquivo
- 7 file()
- 8 Ler Arquivo Via URL
- 9 feof(handle)
- 10 Contando o Número de Linhas de um arquivo
- 11 Contar palavras de um arquivo mostrando duplicadas
- 12 Ler de forma inversa um arquivo, linha a linha
- 13 Ler aleatoriamente linha de arquivo
- 14 Ler linha específica de arquivo
- 15 Operações com Diretórios
- 16 Mostrando conteúdo de diretório
- 17 Excluindo arquivos do SO
- 18 Copiando arquivos
- 19 Processando todos os arquivos de um diretório
- 20 Teste se Arquivo pode ser lido
- 21 Testar se Arquivo Permite Escrita
- 22 Testar se Arquivo Existe
- 23 Testar se é Arquivo ou Diretório
- 24 Outras Funções
- 25 Espaço Total no Disco
- 26 Espaço Livre no Disco
- 27 Tamanho de Diretório, número de arquivos e sub-diretórios
    - 27.1 Detalhes : $path
- 28 Tamanho de diretório
- 29 Trechos do Tutorial - The right way to read files with PHP da IBM
- 30 Lê e imprime todo o conteúdo de um arquivo CSV
- 31 rmdir -- Remove um diretório
- 32 rename -- Renomear um arquivo ($antigo, $novo)
- 33 unlink -- Apaga um arquivo
- 34 mkdir -- Criar um diretório
- 35 file_exists -- Checa se um arquivo ou diretório existe
- 36 disk_free_space -- Retorna o espaço disponivel no diretório
- 37 Receber Conteúdo de URL
- 38 Recursively find files by filename pattern
- 39 Referência

Abrir arquivo

fopen(filename, mode, [use_include_path]);

filename: pode ser simplesmente um nome, ou um caminho completo.

Exemplos: "arquivo.txt", "./arquivo.dat", "/data/data.txt".

mode: especifica o modo de abertura, ou seja, se o arquivo deve ser aberto para leitura, escrita, etc.
Modos de abertura:

```
- r: abre o arquivo no modo somente leitura e posiciona o ponteiro no início do
arquivo;
        o arquivo já deve existir;
- r+: abre o arquivo para leitura/escrita, posiciona o ponteiro no início do
arquivo;
- w: abre o arquivo no modo somente escrita; se o arquivo já existir, será
sobrescrito;
        senão, será criado um novo;
- w+: abre o arquivo para escrita/leitura; se o arquivo já existir, será
sobrescrito;
        senão, será criado um novo;
- a: abre o arquivo para anexar dados, posiciona o ponteiro no final do arquivo;
        se o arquivo não existir, será criado um novo;
- a+: abre o arquivo para anexo/leitura, posiciona o ponteiro no final do
arquivo;
        se o arquivo não existir, será criado um novo;

<?php
        $fp = fopen("./arquivo.dat", "r"); // $fp conterá o handle do arquivo que
abrimos
?>
<pre>


<h2>Fechar Arquivo</h2>

<pre>
fclose(handle_arquivo);

<?php
fclose($fp);
?>
```

Gravar em Arquivo

```
<?php
$fp = fopen("./dados.txt", "w");
fwrite($fp, "Olá mundo do PHP!"); // grava a string no arquivo. Se não existir
será criado
fclose($fp);
?>
```

Ler Arquivo

```
<?php

$fp = fopen("./dados.txt", "r");
$texto = fread($fp, 20); // lê 20 bytes do arquivo e armazena em $text
fclose($fp);
echo $texto;
?>
```

fgets()

Esta função é usada na leitura de strings de um arquivo. fgets() lê "length \u2013 1" bytes do arquivo. Se for encontrado o final da linha e o número de bytes especificados não tiver sido atingido, fgets() terminará a leitura no final da linha (ou no final do arquivo, se for o caso). Eis a sua sintaxe:

fgets(handle, length);

- handle: handle do arquivo de onde os dados serão lidos;

- length: tamanho em bytes do buffer de leitura;

Exemplo:

```
<?php
$fp = fopen("./dados.txt", "r");
$texto = fgets($fp, 3);
fclose($fp);
echo $texto;
?>
```

fgetc()

Esta função permite ler caractere por caractere de um arquivo. Seguem a sintaxe e um exemplo de utilização:

fgetc(handle);

- handle: handle do arquivo de onde os dados serão lidos;

Ler Todo um Arquivo

```
<?php
$fp = fopen("./dados.txt", "r");
while (!feof($fp)){
    $char .= fgetc($fp);
}
fclose($fp);
echo $char."<br><br>";
?>
```

file()

Esta função lê um arquivo completo, e armazena cada linha do arquivo como um elemento de um array. Depois de ler todo o conteúdo do arquivo, file() o fecha automaticamente, não sendo necessária uma chamada a fclose(); Vejamos a sintaxe:

Ler Arquivo Via URL

$fh = fopen("http://127.0.0.1/", "r");

file(filename);

- filename: nome ou caminho completo de um arquivo.

Exemplo:

```
<?php
// file() lê todo o arquivo
$file_lines = file("./dados.txt");
echo "Primeira linha: " . $file_lines[0]."<br>";
echo "Segunda linha: " . $file_lines[1]."<br>";
echo "Terceira linha: " . $file_lines[2];
?>
```

Além dessas funções paraleitura e escrita, existe ainda uma função bastante útil, que testa se o final do arquivo foi atingido. É a função feof(), que tem a seguinte sintaxe:

feof(handle)

- handle: handle do arquivo;

Exemplo:

```
<?php
$fp = fopen("./dados.txt", "r");
while(!feof($fp)) {
    $char .= fgetc($fp);
}
fclose($fp);
echo $char;
?>
```

Contando o Número de Linhas de um arquivo

```
<?php
// Contar o número de linhas de um arquivo, iniciando com 1
$fp = "./dados.txt";
```

```
$line_count = count (file ($fp));
echo $line_count;
?>
```

Contar palavras de um arquivo mostrando duplicadas

```
/*
// Contar palavras repetidas em um arquivo
$fn = "./dados.txt";
$f_contents = preg_split ("/[\s+/", implode ("", file ($fn)));
foreach ($f_content as $palavra) {
        $ar[$palavra]++;
}
print "A seguinte palavra tem duplicatas<br>";
foreach ($ar as $palavra => $conta_palavra) {
        if (conta_palavra > 1) {
                print "Palavra: $palavra<br>Número de ocorrências:
$conta_palavra<br><br>";
        }
}
*/
```

Ler de forma inversa um arquivo, linha a linha

```
<?php
$fn = "./dados.txt";
$f_contents = array_reverse (file ($fn));
foreach ($f_contents as $linha_inversa) {
        print $linha_inversa;
}
?>
```

Ler aleatoriamente linha de arquivo

```
<?php
$fn = "./pensamentos.txt";
$f_contents = file ($fn);
srand ((double)microtime()*1000000);
$linha_aleatoria = $f_contents[ rand (0, (count ($f_contents) - 1)) ];
print $linha_aleatoria;
?>
```

Ler linha específica de arquivo

```
<?php
$fn = "./dados.txt";
$nr_linha = 38;
$f_contents = file ($fn);
$sua_linha = $f_contents [$nr_linha];
```

```
print $sua_linha;
?>
```

Operações com Diretórios

Mostrando conteúdo de diretório

```
<?php
$dn = opendir ("/home/1www/");
while ($file = readdir ($dn)) {
        print "$file<br>";
}
closedir($dn);
?>
```

Excluindo arquivos do SO

```
<?php
$fn = "./dados0.txt";
// Excluindo arquivo
$ret = unlink ($fn);
if ($ret){
         die ("Arquivo excluído!");
}else{
        die ("Erro ao excluir arquivo");
}
?>
```

Copiando arquivos

```
<?php
$fn = "./dados.txt";

if (copy ($fn, "dados0.txt")){
         die ("Arquivo '$fn' copiado para dados0.txt ");
}else{
        die ("Erro ao copiar arquivo");
}
?>
```

Processando todos os arquivos de um diretório

```
<?php
$dh = dir ("/home/1www/");
while ($entrada = $dh->read()) {
        print $entrada . "<br>";
}
$dh->close();
?>
```

Teste se Arquivo pode ser lido

```
<?php
if (is_readable('http://127.0.0.1/index.html')) {
   header('Location: http://127.0.0.1/index.html');
}else{
        echo "Este arquivo não pode ser lido!";
}

?>


<?php
// TESTAR SE ARQUIVO PERMITE LEITURA

print '<br>';
$filename = 'teste2.php';
if (is_readable($filename)) {
   echo 'O arquivo permite leitura';
} else {
   echo 'O arquivo não permite leitura';
}
print '<br>';
```

//Outro

<?php if (is_readable('http://127.0.0.1/index.html')) {

```
  header('Location: http://127.0.0.1/index.html');
```

}else{ echo "Este arquivo não pode ser lido!"; }

?>

Testar se Arquivo Permite Escrita

```
if (is_writable($filename)) {
   echo 'O arquivo permite escrita';
} else {
   echo 'O arquivo não permite escrita';
}

?>
```

Testar se Arquivo Existe

```
<?php
print '<br>';
// TESTAR SE ARQUIVO EXISTE
$filename = 'teste2.php';
```

```
if (file_exists($filename)) {
   echo "O arquivo $filename existe";
} else {
   echo "O arquivo $filename não existe";
}
?>
```

Testar se é Arquivo ou Diretório

```
<?php
print '<br>';
// TESTAR SE ARQUIVO É UM ARQUIVO COMUN OU SE É DIRETÓRIO
$filename = 'teste2.php';

$filename2 = 'c:\windows';

if (is_file($filename)) {
   echo "O arquivo $filename é comun";
}else{
  echo "O arquivo $filename não é um arquivo comun";
}
print '<br>';

if (is_file($filename2)){
   echo "O arquivo $filename2 é comun";
}else{
  echo "O arquivo $filename2 não é um arquivo comun";
}

print '<br>';

if (is_dir($filename2)){
   echo "$filename2 é um diretório";
}else{
  echo "$filename2 não é um diretório";
}
```

Outras Funções

```
is_link($diretorio)
readlink($dir_link) // retorna o path completo do link
bool symlink ( string $destino, string $linkorigem ) // Cria um link simbólico
```

Espaço Total no Disco

<?php // ESPAÇO TOTAL NO DISCO $diretorio="c:/"; print disk_total_space($diretorio); print "
";

Espaço Livre no Disco

// ESPAÇO LIVRE NO DISCO // $df contém o número de bytes disponível em "/" $df = disk_free_space("c:/"); print $df ?>

Tamanho de Diretório, número de arquivos e sub-diretórios

<?php // CALCULANDO TAMANHO OCUPADO POR UM DIRETÓRIO, NR DE ARQUIVOS E SUBDIRETÓRIOS // http://www.go4expert.com/forums/showthread.php?t=290

function getDirectorySize($path) {

```
 $totalsize = 0;
 $totalcount = 0;
 $dircount = 0;
 if ($handle = opendir ($path))
 {
   while (false !== ($file = readdir($handle)))
   {
     $nextpath = $path . '/' . $file;
     if ($file != '.' && $file != '..' && !is_link ($nextpath))
     {
       if (is_dir ($nextpath))
       {
         $dircount++;
         $result = getDirectorySize($nextpath);
         $totalsize += $result['size'];
         $totalcount += $result['count'];
         $dircount += $result['dircount'];
       }
       elseif (is_file ($nextpath))
       {
         $totalsize += filesize ($nextpath);
         $totalcount++;
       }
     }
   }
 }
 closedir ($handle);
 $total['size'] = $totalsize;
 $total['count'] = $totalcount;
 $total['dircount'] = $dircount;
 return $total;
```

}

function sizeFormat($size) {

```
   if($size<1024)
   {
       return $size." bytes";
```

```
    }
    else if($size<(1024*1024))
    {
        $size=round($size/1024,1);
        return $size." KB";
    }
    else if($size<(1024*1024*1024))
    {
        $size=round($size/(1024*1024),1);
        return $size." MB";
    }
    else
    {
        $size=round($size/(1024*1024*1024),1);
        return $size." GB";
    }

}
```

// Usando

$path="D:/_xampplite/htdocs/desweb/7AplicativosExemplo/extras"; $ar=getDirectorySize($path);

echo "

Detalhes : $path

";

echo "Tamanho total : ".sizeFormat($ar['size'])."
"; echo "No. de arquivos : ".$ar['count']."
"; echo "No. de diretórios : ".$ar['dircount']."
"; ?>

Tamanho de diretório

<? // CALCULANDO TAMANHO (BYTES) OCUPADO POR UM DIRETÓRIO//
http://www.weberdev.com/get_example-4171.html

function dir_size($dir) {

```
    if( !$dir or !is_dir( $dir ) )
    {
        return 0;
    }

    $ret = 0;
    $sub = opendir( $dir );
    while( $file = readdir( $sub ) )
    {
        if( is_dir( $dir . '/' . $file ) && $file !== ".." && $file !== "." )
```

```
        {
            $ret += dir_size( $dir . '/' . $file );
            unset( $file );
        }
        elseif( !is_dir( $dir . '/' . $file ) )
        {
            $stats = stat( $dir . '/' . $file );
            $ret += $stats['size'];
            unset( $file );
        }
    }
    closedir( $sub );
    unset( $sub );
    return $ret;

}

echo dir_size("D:/_xampplite/htdocs/desweb/1LinguagemPHP/php/tutoriais/trabalhando_com"); ?>


<?php

// MAIS UMA ÓTIMA FUNÇÃO PARA LER O TAMANHO DE UM DIRETÓRIO

/*

* PHP Freaks Code Library
* http://www.phpfreaks.com/quickcode.php
*
* Title: Directory Size
* Version: 1.0
* Author: Nathan Taylor aka(Lakario)
* Date: Saturday, 12/20/2003 - 12:34 PM
*
*
*
* NOTICE: This code is available from PHPFreaks.com code Library.
*         This code is not Copyrighted by PHP Freaks.
*
*         PHP Freaks does not claim authorship of this code.
*
*         This code was submitted to our website by a user.
*
*         The user may or may not claim authorship of this code.
*
*         If there are any questions about the origin of this code,
*         please contact the person who submitted it, not PHPFreaks.com!
*
*         USE THIS CODE AT YOUR OWN RISK! NO GUARANTEES ARE GIVEN!
*
* SHAMELESS PLUG: Need WebHosting? Checkout WebHost Freaks:
*                 http://www.webhostfreaks.com
*                 WebHosting by PHP Freaks / The Web Freaks!
```

- /

// * Description / Example: // * // * This code will allow an individual to quickly obtain the size and number of files inside a directory recursively. // * // * It also includes a convenient byte value converter to kilobyte, megabyte, gigabyte, or trilobyte accordingly.

?>

<?php function DirStat($directory) { global $FolderCount, $FileCount, $FolderSize;

chdir($directory); $directory = getcwd(); if($open = opendir($directory)) { //while($file = readdir($open)) { while(false !== ($file = readdir($open))) { if($file == '..' || $file == '.') continue; if(is_file($file)) { $FileCount++; $FolderSize += filesize($file); } elseif(is_dir($file)) { $FolderCount++; } } if($FolderCount > 0) { $open2 = opendir($directory); while($folders = readdir($open2)) { $folder = $directory.'/'.$folders; if($folders == '..' || $folders == '.') continue; if(is_dir($folder)) { DirStat($folder); } } closedir($open2); } closedir($open); } }

function ByteSize($bytes) { $size = $bytes / 1024; if($size < 1024){ $size = number_format($size, 2); $size .= 'kb'; } else { if($size / 1024 < 1024) { $size = number_format($size / 1024, 2); $size .= 'mb'; } elseif($size / 1024 / 1024 < 1024) { $size = number_format($size / 1024 / 1024, 2); $size .= 'gb'; } else { $size = number_format($size / 1024 / 1024 / 1024, 2); $size .= 'tb'; } } return $size; }

$folder = 'D:/1Enviar/Hoje'; $dir = getcwd(); DirStat($folder, 0); chdir($dir); $FolderSize = ByteSize($FolderSize);

echo '**Folder Name:** '.$folder.'
'.chr(10); echo '**File Count:** '.$FileCount.'
'.chr(10); echo '**Folder Size:**'.$FolderSize.'
'.chr(10); ?>

Trechos do Tutorial - The right way to read files with PHP da IBM

USANDO fscanf

fscanf

Coming back to string processing, fscanf again follows the traditional C file library functions. If you're unfamiliar with it, fscanf reads field data into variables from a file.

list ($field1, $field2, $field3) = fscanf($fh, "%s %s %s");

FUNÇÃO fpassthru

No matter how you've been reading your file, you can dump the rest of your data to your standard output channel using fpassthru.

```
fpassthru($fh);
```

```
my_file = file_get_contents("myfilename");
echo $my_file;
```

Although it isn't best practice, you can write this command even more concisely as:

```
echo file_get_contents("myfilename");
```

This article is primarily about dealing with local files, but it's worth noting that you can grab, echo, and parse other Web pages with these functions, as well.

```
echo file_get_contents("http://127.0.0.1/");
```

This command is effectively the same as:

```
$fh = fopen("http://127.0.0.1/", "r");
fpassthru($fh);
```

You must be looking at this and thinking, "That's still way too much effort." The PHP developers agree with you. So you can shorten the above command to:

```
readfile("http://127.0.0.1/");
```

The readfile function dumps the entire contents of a file or Web page to the default output buffer. By default, this command prints an error message if it fails. To avoid this behavior (if you want to), try:

```
@readfile("http://127.0.0.1/");
```

Of course, if you actually want to parse your files, the single string that file_get_contents returns might be a bit overwhelming. Your first inclination might be to break it up a little bit with the split() function.

```
$array = split("\n", file_get_contents("myfile"));
```

But why go through all that trouble when there's a perfectly good function to do it for you? PHP's file() function does this in one step: It returns an array of strings broken up by lines.

$array = file("myfile");

It should be noted that there is a slight difference between the above two examples. While the split command drops the newlines, the newlines are still attached to the strings in the array when using the file command (as with the fgets command).

PHP's power goes far beyond this, though. You can parse entire PHP-style .ini files in a single command using parse_ini_file. The parse_ini_file command accepts files similar to Listing 4.

Listing 4. A sample .ini file

```
; Comment
[personal information]
name = "King Arthur"
quest = To seek the holy grail
favorite color = Blue

[more stuff]
Samuel Clemens = Mark Twain
Caryn Johnson = Whoopi Goldberg
```

The following commands would dump this file into an array, then print that array:

```
$file_array = parse_ini_file("holy_grail.ini");
print_r $file_array;
```

The following output is the result:

```
Listing 5. Output
Array
(
    [name] => King Arthur
    [quest] => To seek the Holy Grail
    [favorite color] => Blue
    [Samuel Clemens] => Mark Twain
    [Caryn Johnson] => Whoopi Goldberg
)
```

Of course, you might notice that this command merged the sections. This is the default behavior, but you can fix it easily by passing a second argument to parse_ini_file: process_sections, which is a Boolean variable. Set process_sections to True.

$file_array = parse_ini_file("holy_grail.ini", true);

print_r $file_array;

And you'll get the following output:

Listing 6. Output

```
Array
(
    [personal information] => Array
        (
            [name] => King Arthur
            [quest] => To seek the Holy Grail
            [favorite color] => Blue
        )

    [more stuff] => Array
        (
            [Samuel Clemens] => Mark Twain
            [Caryn Johnson] => Whoopi Goldberg
        )

)
```

PHP placed the data into an easily parsable multidimensional array.

This is just the tip of the iceberg when it comes to PHP file processing. More complex functions like tidy_parse_file and xml_parse can help you handle HTML and XML documents, respectively. See Resources for details on how these particular functions work. These are well worth looking at if you'll be dealing with those types of files, but instead of considering every possible file type you might run into in detail in this article, here are a few good general rules for dealing with the functions I've described thus far.

Good practice

Never assume that everything in your program will work as planned. For example, what if the file you're looking for has moved? What if the permissions have been altered and you're unable to read the contents? You can check for these things in advance by using file_exists and is_readable.

Listing 7. Use file_exists and is_readable

```
$filename = "myfile";
if (file_exists($filename) && is_readable ($filename)) {
        $fh = fopen($filename, "r");
        # Processing
        fclose($fh);
}

In practice, however, such code is probably overkill. Processing the return value
of fopen is simpler and more accurate.

if ($fh = fopen($filename, "r")) {
        # Processing
        fclose($fh);
}
```

Final do trecho do tut IBM----------

Lê e imprime todo o conteúdo de um arquivo CSV

```
<?php
// Lê e imprime todo o conteúdo de um arquivo CSV
$row = 1;
$handle = fopen ("test.csv","r");
while ($data = fgetcsv ($handle, 1000, ",")) {
   $num = count ($data);
   print "<p> $num campos na linha $row: <br>";
   $row++;
   for ($c=0; $c < $num; $c++) {
       print $data[$c] . "<br>";
   }
}
fclose ($handle);
?>
<pre>

Outra:
<pre>
<?php

define('CSV_BOTH', 1);
define('CSV_ASSOC', 2);
define('CSV_NUM', 3);
```

```
function parse_csv($filename, $result_type = CSV_BOTH) {
   if(!file_exists($filename)) {
       die("file (" . $filename . ") does not exist\n");
   }

   $lines = file($filename);

   $title_line = trim(array_shift($lines));
   $titles = split(",", $title_line);

   $records = array();
   foreach($lines as $line_num => $line) {
       $subject = trim($line);
       $fields = array();
       for($field_num = 0; $field_num < count($titles); $field_num++) {
           if($subject{0} == '"') {
               preg_match('/^"(([^"]|\\")*)",?(.*)$/', $subject, $matches);

               $value = $matches[1];
               $subject = $matches[3];

               if($result_type == CSV_BOTH || $result_type == CSV_ASSOC) {
                   $fields[$titles[$field_num]] = $value;
               }

               if($result_type == CSV_BOTH || $result_type == CSV_NUM) {
                   $fields[$field_num] = $value;
               }
           } else {
               preg_match('/^([^,]*),?(.*)$/', $subject, $matches);

               $value = $matches[1];
               $subject = $matches[2];

               if($result_type == CSV_BOTH || $result_type == CSV_ASSOC) {
                   $fields[$titles[$field_num]] = $value;
               }

               if($result_type == CSV_BOTH || $result_type == CSV_NUM) {
                   $fields[$field_num] = $value;
               }
           }
       }

       $records[] = $fields;
   }

   return $records;
}

?>
<pre>

This version is conditional - it only adds quotes if needed:
<pre>
<?
function csv_escape($str) {
```

```
   $str = str_replace(array('"', ',', "\n", "\r"), array('""', ',', "\n", "\r"),
$str, &$count);
   if($count) {
       return '"' . $str . '"';
   } else {
       return $str;
   }
}
?>
```

```
<?php
$caminho = "/home/httpd/html/index.php";
$arquivo = basename ($caminho);        // $arquivo = "index.php"
$arquivo = basename ($caminho,".php"); // $arquivo = "index"
?>
```

rmdir -- Remove um diretório

This functions deletes or empties the directory. Without using recursive functions!

```
<?php

/**
 * Removes the directory and all its contents.
 *
 * @param string the directory name to remove
 * @param boolean whether to just empty the given directory, without deleting the
given directory.
 * @return boolean True/False whether the directory was deleted.
 */
function deleteDirectory($dirname,$only_empty=false) {
   if (!is_dir($dirname))
       return false;
   $dscan = array(realpath($dirname));
   $darr = array();
   while (!empty($dscan)) {
       $dcur = array_pop($dscan);
       $darr[] = $dcur;
       if ($d=opendir($dcur)) {
           while ($f=readdir($d)) {
               if ($f=='.' || $f=='..')
                   continue;
               $f=$dcur.'/'.$f;
               if (is_dir($f))
                   $dscan[] = $f;
               else
                   unlink($f);
           }
           closedir($d);
       }
   }
   $i_until = ($only_empty)? 1 : 0;
   for ($i=count($darr)-1; $i>=$i_until; $i--) {
```

```
        echo "\nDeleting '".$darr[$i]."' ... ";
        if (rmdir($darr[$i]))
            echo "ok";
        else
            echo "FAIL";
    }
    return (($only_empty)? (count(scandir)<=2) : (!is_dir($dirname)));
}

?>

//Outra

<?php
/* Function to remove directories, even if they contain files or
subdirectories.  Returns array of removed/deleted items, or false if nothing
was removed/deleted.

by Justin Frim.  2007-01-18

Feel free to use this in your own code.
*/

function rmdirtree($dirname) {
    if (is_dir($dirname)) {    //Operate on dirs only
        $result=array();
        if (substr($dirname,-1)!='/') {$dirname.='/';}    //Append slash if
necessary
        $handle = opendir($dirname);
        while (false !== ($file = readdir($handle))) {
            if ($file!='.' && $file!= '..') {    //Ignore . and ..
                $path = $dirname.$file;
                if (is_dir($path)) {    //Recurse if subdir, Delete if file
                    $result=array_merge($result,rmdirtree($path));
                }else{
                    unlink($path);
                    $result[].=$path;
                }
            }
        }
        closedir($handle);
        rmdir($dirname);    //Remove dir
        $result[].=$dirname;
        return $result;    //Return array of deleted items
    }else{
        return false;    //Return false if attempting to operate on a file
    }
}
?>
```

rename -- Renomear um arquivo ($antigo, $novo)

```
<?php
  rename("/tmp/tmp_file.txt", "/home/user/login/docs/my_file.txt");
?>
```

unlink -- Apaga um arquivo

mkdir -- Criar um diretório

mkdir ("/path/to/my/dir", 0700);

is_file -- Diz se o arquivo é um arquivo comum (não é diretório)

file -- Le um arquivo inteiro para um array

```
<?php
// Le um arquivo em um array. Nesse exemplo você pode obter via HTTP para obter
// o código fonte HTML de uma URL.
$lines = file ('http://www.exemplo.com/');

// Roda através do array, mostrando o fonte HTML com numeração de linhas.
foreach ($lines as $line_num => $line) {
   echo "Linha #<b>{$line_num}</b> : " . htmlspecialchars($line) . "<br>\n";
}

// Outro exemplo, onde obtemos a página web inteira como uma string. Veja também
file_get_contents().
$html = implode ('', file ('http://www.exemplo.com/'));
?>
```

file_exists -- Checa se um arquivo ou diretório existe

```
$filename = '/caminho/para/qualquer.txt';

if (file_exists($filename)) {
   print "O arquivo $filename existe";
} else {
   print "O arquivo $filename não existe";
}
```

disk_free_space -- Retorna o espaço disponivel no diretório

Receber Conteúdo de URL

<?php // Trazer conteúdo de arquivo ou de página para string

// Define a context for HTTP. $aContext = array(

```
  'http' => array(
      'proxy' => 'tcp://10.0.0.1:3128', // This needs to be the server and the
port of the NTLM Authentication Proxy Server.
      'request_fulluri' => True,
      ),
  );
```

$cxContext = stream_context_create($aContext);

// Now all file stream functions can use this context. $sFile = file_get_contents("http://www.google.com", False, $cxContext); echo $sFile; ?>

Recursively find files by filename pattern

http://snippets.dzone.com/posts/show/4147

Scans a directory, and all subdirectories for files, matching a regular expression. Each match is sent to the callback provided as third argument. A simple example:

function my_handler($filename) {

```
 echo $filename . "\n";
```

} find_files('c:/', '/php$/', 'my_handler');

And the actual snippet

function find_files($path, $pattern, $callback) {

```
 $path = rtrim(str_replace("\\", "/", $path), '/') . '/';
 $matches = Array();
 $entries = Array();
 $dir = dir($path);
 while (false !== ($entry = $dir->read())) {
   $entries[] = $entry;
 }
 $dir->close();
 foreach ($entries as $entry) {
   $fullname = $path . $entry;
   if ($entry != '.' && $entry != '..' && is_dir($fullname)) {
     find_files($fullname, $pattern, $callback);
   } else if (is_file($fullname) && preg_match($pattern, $entry)) {
     call_user_func($callback, $fullname);
   }
 }
```

}

Referência

http://phpbrasil.com/articles/print.php/id/310

Trabalhando em PHP com/Path

De Wikibooks

Conteúdo

- 1 PATH
- 2 Exemplos simples de uso de funções para path do PHP
- 3 Recebendo o Path Absoluto do Script Atual
- 4 Recebendo o path relativo do webserver do script atual

PATH

Exemplos simples de uso de funções para path do PHP

```
<?php
   $path=dirname(realpath($_SERVER['SCRIPT_FILENAME']));
   $path=substr($path,0,strlen($path) - 5);
echo "<br>Path deste script sem 5 finais caracteres - " . $path;

echo "<br><br>Diretório atual - ".dirname(__FILE__);
echo "<br>Caminho completo do script atual - ".__FILE__;

echo "<br>URL do script atual - " . "http://" . $_SERVER['HTTP_HOST'] .
$HTTP_SERVER_VARS["SCRIPT_NAME"];
?>

<?php
$path = "/etc/passwd";
$file = dirname($path); // $file is set to "/etc"
?>

<?php
$path = "/home/httpd/html/index.php";
$file = basename($path);        // $file is set to "index.php"
$file = basename($path, ".php"); // $file is set to "index"
?>

<?php
echo dirname($_SERVER["REQUEST_URI"]);
?>
```

Recebendo o Path Absoluto do Script Atual

dirname(__FILE__)

Recebendo o path relativo do webserver do script atual

```
function GetRelativePath($path)
{
   $npath = str_replace('\\', '/', $path);
   return str_replace(GetVar('DOCUMENT_ROOT'), '', $npath);
}
```

GetRelativePath(dirname(__FILE__));

<?php

if (DIRECTORY_SEPARATOR=='/')

```
 $absolute_path = dirname(__FILE__).'/';
```

else

```
 $absolute_path = str_replace('\\\\', '/', dirname(__FILE__)).'/';
```

?>

Resultará em um path absoluto no estilo UNIX que funciona também em PHP5 sob Windows.

Em algumas instalações (< 4.4.1) $_SERVER['REQUEST_URI'] não está configurado, usado o código para corrigir:

```
<?php

if (!isset($_SERVER['REQUEST_URI'])) {
   $_SERVER['REQUEST_URI'] = substr($_SERVER['PHP_SELF'],1);
   if (isset($_SERVER['QUERY_STRING']))
$_SERVER['REQUEST_URI'].='?'.$_SERVER['QUERY_STRING'];
}

?>

$my_uri = "http://" . $_SERVER['HTTP_HOST'] . $HTTP_SERVER_VARS["SCRIPT_NAME"];

// então

<?php echo ("$my_uri");?>

<?php
include("{$_SERVER['DOCUMENT_ROOT']}/includes/my_include.php");
?>
```

Você pode usar isso para receber o diretório pai:

dirname(dirname(__FILE__))

...include a file relative to file path:

include(dirname(__FILE__) .'/path/relative/file_to_include.php');

Isso colocará ambos os paths "www" e "file" de forma fácil para transportar o array.

```
<?php
// build the www path:
$me = $_SERVER['PHP_SELF'];
$Apathweb = explode("/", $me);
$myFileName = array_pop($Apathweb);
$pathweb = implode("/", $Apathweb);
$myURL = "http://".$_SERVER['HTTP_HOST'].$pathweb."/".$myFileName;
$PAGE_BASE['www'] = $myURL;

// build the file path:
strstr( PHP_OS, "WIN") ? $strPathSeparator = "\\" : $strPathSeparator = "/";
$pathfile = getcwd ();
$PAGE_BASE['physical'] = $pathfile.$strPathSeparator.$myFileName;

// this is so you can verify the results:
$www = $PAGE_BASE['www'];
$physical = $PAGE_BASE['physical'];

echo "$physical<p>";
echo "$www<p>";
?>
```

retornará algo como:

Windows:

F:\dev\Inetpub\wwwroot\somedirectory\index.php

http://devserver/somedirectory/index.php

Unix:

/home/somepathto/gieson.com/webroot/index.php

http://www.gieson.com/index.php

Path absoluto do script em execução

$path=dirname(realpath($_SERVER['SCRIPT_FILENAME']));

Trabalhando em PHP com/Includes

De Wikibooks

Include

Inclue e avalia o conteúdo do arquivo.

Require

Também inclue e avalia o conteúdo do arquivo incluído.

A diferença entre ambos é que o include ao encontrar um erro, lança um warning apenas, enquanto que o require lança um Fatal Error, que pára o processamento.

Uso do require: para códigos que requerem maior segurança.

Include_Once e Require_Once

São semelhantes ao include e require, sendo que estes incluem um arquivo somente uma vez.

Mostrando uso do require_once:

echo.php

```
<?php
echo "Hello";
?>
```

teste.php

```
<?php
require('echo.php');
require_once('echo.php');
?>
```

Chamar no navegador o script teste.php

saída: "Hello".

Agora teste2.php:

```
<?php
```

```
require('echo.php');
require('echo.php');
?>
```

Executar teste2.php

saída: "HelloHello".

Agora teste3.php:

```
<?php
require_once('echo.php');
require('echo.php');
?>
```

Executar teste3.php

saída: "HelloHello".

Ou seja, ao encontrar require_once, ele verifica se o arquivo já foi incluído, e somente o incluirá novamente se ele ainda não tiver sido incluído.

```
<?php
// Isto está errado e não funcionará como desejado
if ($condition)
   include $arquivo;
else
   include $outro;

// E este está correto
if ($condition) {
   include $arquivo;
} else {
   include $outro;
}

?>
```

```
<?php
$path="/full/path/to/script/";
if (getdomain($path) == 'yourdomain'){
     include($path.'somefile.php');
}
?>
```

"variables.php"

```
<?php
$includer = basename($_SERVER['SCRIPT_NAME']);

switch ($includer) {
```

```
   case 'a.php':
   $this_variable = 'included by script a.php';
   break;

   case 'b.php':
   $this_variable = 'included by script b.php';
   break;

   default:
   $this_variable = 'included by unkown script';
}
echo $this_variable;
?>
```

Test with 3 different files "a.php", "b.php", "c.php", all with the same content:

```
<?php
include 'variables.php';
?>
```

Trabalhando em PHP com/Formatação da saída

De Wikibooks

Conteúdo

- 1 printf -- Mostra uma string formatada
- 2 sscanf
- 3 fscanf
- 4 sprintf
- 5 vprintf

printf -- Mostra uma string formatada

void printf (string format [, mixed args])

sscanf

Interpreta a entrada de uma string de acordo com um formato mixed sscanf (string str, string formato [, string var1])

```
<?php
// Pegando o número serial
$serial = sscanf("SN/2350001","SN/%d");
// e a data de criação
$mandate = "January 01 2000";
list($month, $day, $year) = sscanf($mandate,"%s %d %d");
echo "O Item $serial foi criado em: $year-".substr($month,0,3)."-$day\n";
?>
```

Se parâetros opcionais são passados, a função retornará o número de valores assumidos. Os parâmetros opcionais devem ser passados por referência.

Exemplo 2. sscanf() - usando parâmetros opcionais

```
<?php
// pega informação do autor e gera uma entrada de DocBook
$auth = "24\tLewis Carroll";
$n = sscanf($auth,"%d\t%s %s", &$id, &$first, &$last);
echo "<author id='$id'>
```

```
   <firstname>$first</firstname>
   <surname>$last</surname>
</author>\n";
?>
```

fscanf

Interpreta a leitura de um arquivo de acordo com um formato

mixed fscanf (resource handle, string formato [, string var1])

```
$handle = fopen ("users.txt","r");
while ($userinfo = fscanf ($handle, "%s\t%s\t%s\n")) {
   list ($name, $profission, $countrycode) = $userinfo;
   //... fazer algo com os valores
}
fclose($handle);

$goodevil = array ('There is a difference between %s and %s', 'good', 'evil');
echo call_user_func_array('sprintf',$goodevil);

<?php
$heading1 = "Label 1";
$heading2 = "Label 2";

$value1 = "31298";
$value2 = "98";

print "<pre>\n";
printf ("%'.-15.15s%'.6.6s\n", $heading1, $value1);
printf ("%'.-15.15s%'.6.6s\n", $heading2, $value2);
print "


\n";

?>
```

<?php $f='<?php $f=%c%s%c; printf($f,39,$f,39); ?>'; printf($f,39,$f,39); ?>

sprintf

Retorna uma string formatada

string sprintf (string format [, mixed args])

Um especificador de tipo que diz que o argumento deve ser tratado como do tipo. Os tipos possivéis são:

% - Um caractere porcento. Não é requerido neenhum argumento. b - O argumento é tratado com um inteiro, e mostrado como um binário. c - O argumento é tratado como um inteiro, e mostrado como o caractere ASCII correspondente. d - O argumento é tratado como um inteiro, e mostrado como um

número decimal com sinal. u - O argumento é tratado com um inteiro, e mostrado como um número decimal sem sinal. f - O argumento é tratado como um float, e mostrado como um número de ponto flutuante. o - O argumento é tratado com um inteiro, e mostrado como un número octal. s - O argumento é tratado e mostrado como uma string. x - O argumento é tratado como um inteiro, e mostrado como um número hexadecimal (com as letras minúsculas). X - O argumento é tratado como um inteiro, e mostrado como um número hexadecimal (com as letras maiúsculas).

```
<?php
$format = "There are %d monkeys in the %s";
printf($format,$num,$location);
?>
```

Este deve mostrar, "There are 5 monkeys in the tree". Mas imagine que nós estejamos criando a string de formatação em um arquivo separado, normalmente para internacionalizar e rescrevemos como:

Exemplo 2. Troca de argumentos

```
<?php
$format = "The %s contains %d monkeys";
printf($format,$num,$location);
?>
```

Agora nós temos um problema. A ordem dos argumentos na string de formatação não combina com os argumentos no código. Nós gostariamos de deixar o código como esta e simplesmente indicar na string de formatação quais argumentos pertencem aonde. Podemos escrever a string de formatação assim:

Exemplo 3. Troca de argumento

```
<?php
$format = "The %2\$s contains %1\$d monkeys";
printf($format,$num,$location);
?>
```

Um beneficio adicional disto é ue você pode repetir os especificadores de conversão sem adicionar mais argumentos em seu código. Por exemplo:

Exemplo 4. Troca de argumento

```
<?php
$format = "The %2\$s contains %1\$d monkeys.
           That's a nice %2\$s full of %1\$d monkeys.";
printf($format, $num, $location);
?>
```

Veja também printf(), sscanf(), fscanf(), vsprintf() e number_format().

Exemplos

Exemplo 5. sprintf(): inteiros prenchidos com zero

```
<?php
$isodate = sprintf("%04d-%02d-%02d", $year, $month, $day);
?>
```

Exemplo 6. sprintf(): formatando dinheiro

```
<?php
$money1 = 68.75;
$money2 = 54.35;
$money = $money1 + $money2;
// echo $money irá mostrar "123.1";
$formatted = sprintf("%01.2f", $money);
// echo $formatted irá mostrar "123.10"
?>
```

vprintf

Mostra uma string formatada

void vprintf (string formato, array args)

Mostra uma string formatada de acordo com o formato (o qual é descrito na documentação para a função sprintf()).

Funciona como printf() mas aceita uma matriz de argumentos, ao invés de um número variavel de argumentos.

```
<?php
$fruits = array(1, 'banana',1, 'apples', 3, 'oranges', 2, 'peaches');

vprintf("I have %d %s, %d %s, %d %s and %d %s.", $fruits);
?>
```

Trabalhando em PHP com/Imagens e Gráficos

De Wikibooks

Trabalhando com a biblioteca gráfica GD

(Se no Windows remover o ponto-e-vírgula ";" da linha "extension=php_gd.dll" do php.ini)

Conteúdo

- 1 Gerando Imagens Dinamicamente
- 2 Desenhando retângulos
- 3 Desenhando polígonos
- 4 Desenhando arcos
- 5 Referência

Gerando Imagens Dinamicamente

por Luiz Ribeiro

O PHP oferece uma interface ao módulo GD de Thomas Boutell. Usando tal módulo, você pode criar e editar imagens nos formatos JPEG e PNG. O formato GIF já foi aceito, mas como o algoritmo de compressão do GIF (LZW) contém uma patente de posse da Unisys, os desenvolvedores do módulo foram obrigados à retirar o suporte a esse formato nas versões mais recentes.

Bom, para iniciar vou explicar o procedimento para criar uma imagem usando o módulo GD em PHP. Se você não tem esse módulo, você pode fazer o download dele em http://www.boutell.com/gd/. Normalmente a GD acompanha uma instalação completa do PHP.

Para se criar a imagem, será usada a função ImageCreate(), então serão realizadas as alterações na imagem, então será finalizada a imagem usando ImageJpeg(), ImagePng() ou até ImageGif() se a versão do módulo GD for inferior à 1.4.

Bom, vamos ao que interessa. Primeiramente vamos criar uma pequena imagem com o seguinte texto: PHPBrasil. O código ficará da seguinte forma:

```
<?php
header("Content-type: image/gif"); //Informa ao browser que o arquivo é uma
imagem no formato GIF

$imagem = ImageCreate(150,40); //Cria uma imagem com as dimensões 100x20

$vermelho = ImageColorAllocate($imagem, 255, 0, 0); //Cria o segundo plano da
imagem e o configura para vermelho
$branco = ImageColorAllocate($imagem, 255, 255, 255); //Cria a cor de primeiro
plano da imagem e configura-a para branco

ImageString($imagem, 3, 3, 3, "PHPBrasil", $branco); //Imprime na imagem o texto
PHPBrasil na cor branca que está na variável $branco

ImageGif($imagem); //Converte a imagem para um GIF e a envia para o browser

ImageDestroy($imagem); //Destrói a memória alocada para a construção da imagem
GIF.
?>
```

Bom, o script está todo comentado e acho que você entendeu. Se alguma dúvida ficar martelando aí, manda um comentário. =D

Bom, neste exemplo usamos a função ImageGif() para converter a imagem, $imagem, e depoisa enviamos ao navegador. Mas poderíamos ter salvo esta imagem em um arquivo, ao invés de mostrar ela no navegador. Veja o exemplo:

```
<?php
$arquivo = "imagem1.gif";

$imagem = ImageCreate(150,40);

$vermelho = ImageColorAllocate($imagem, 255, 0, 0);
$branco = ImageColorAllocate($imagem, 255, 255, 255);

ImageString($imagem, 3, 3, 3, "PHPBrasil", $branco);
ImageGif($imagem, $arquivo);

ImageDestroy($imagem);

echo "A imagem foi salva no arquivo $arquivo.";
?>
```

Como você deve ter notado, apenas retiramos aquele header() (que informava ao browser que o arquivo era uma imagem), afinal este exemplo não irá mostrar a imagem no navegador e sim gravar ela em $arquivo, e também mudamos os parâmetros da função ImageGif() para salvar a imagem no arquivo.

Nesta parte do artigo, irei explicar como desenhar retângulos, polígonos e arcos.

Desenhando retângulos

Vamos ao primeiro exemplo, que irá desenhar um simples retângulo preenchido usando GD (o formato da imagem a seguir é PNG).

```
<?php
header("Content-type: image/png");
$imagem = ImageCreate(100, 20);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
$azul = ImageColorAllocate($imagem, 20, 93, 233);
ImageFilledRectangle($imagem, 5, 10, 60, 14, $azul);
ImagePng($imagem);
ImageDestroy($imagem);
?>
```

Bom, neste exemplo só há uma função nova, a função ImageFilledRectangle() que como seu próprio nome diz é uma função que cria um retângulo com as dimensões e posição informadas, e na cor azul, que foi definida na variável $azul.

Já para criar um retângulo sem preenchimento você simplesmente irá trocar a função ImageFilledRectangle() por ImageRectangle(). O exemplo ficará da seguinte forma:

```
<?php
header("Content-type: image/png");
$imagem = ImageCreate(100, 20);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
$azul = ImageColorAllocate($imagem, 20, 93, 233);
ImageRectangle($imagem, 5, 10, 60, 14, $azul);
ImagePng($imagem);
ImageDestroy($imagem);
?>
```

Como foi dito, este exemplo irá criar uma imagem com um retângulo sem preenchimento, mas sua borda terá a cor $azul.

Desenhando polígonos

Para desenhar polígonos, vamos usar a função ImagePolygon(), que irá criar um polígono sem preenchimento, e a função ImageFilledPolygon() que irá desenhar um polígono com preenchimento.

Em nosso primeiro exemplo, vamos desenhar um polígono com vértices de (12, 10), (15, 20), (50, 17) e (70, 10) com uma borda de azul-claro:

```
<?php
header("Content-type: image/png");
$imagem = ImageCreate(100, 20);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
$azul = ImageColorAllocate($imagem, 20, 93, 233);
$pontos = array(12, 10, 15, 20, 50, 17, 70, 10);
ImagePolygon($imagem, $pontos, 4, $azul);
ImagePng($imagem);
ImageDestroy($imagem);
```

```
?>
```

Bom, agora vamos criar um polígono preenchido, você já deve ter pensado que o código será o mesmo, mas ao invés de ImagePolygon() usaremos ImageFilledPolygon(), se você énsou isso, acertou em cheio. Vamos ver como ficaria nossa imagem com um retângulo preenchido:

```
<?php
header("Content-type: image/png");
$imagem = ImageCreate(100, 20);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
$azul = ImageColorAllocate($imagem, 20, 93, 233);
$pontos = array(12, 10, 15, 20, 50, 17, 70, 10);
ImageFilledPolygon($imagem, $pontos, 4, $azul);
ImagePng($imagem);
ImageDestroy($imagem);
?>
```

Desenhando arcos

Bom, agora vamos desenhar alguns arcos em nossas imagens, para isso vamos usar a função ImageArc(). Antes de começarmos, vou passar a sintaxe da função:

int ImageArc(int im, int cx, int cy, int w, int h, int s, int e, int co1);

Esta função desenha um arco em uma imagem, im, com uma posição inicial de X de cx e uma posição inicial Y de cy. O arco é de largura w e altura h, com um ângulo inicial de s e um ângulo final de e, tudo na cor co1.

Agora que já entendemos a função ImageArc() vamos ao nosso primeiro exemplo que irá desenhar uma elipse:

```
<?php
header("Content-type: image/gif");
$imagem = ImageCreate(500, 100);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
ImageColorTransparent($imagem, $branco);
$vermelho = ImageColorAllocate($imagem, 20, 93, 233);
ImageArc($imagem, 40, 50, 50, 40, 0, 360, $vermelho);
ImageGif($imagem);
ImageDestroy($imagem);
?>
```

O código acima funciona, pois para ter uma elipse, você precisa de uma diferença de 360 graus entre a posição inicial e a posição final. Aplicando esse conhecimento, também podemos desenhar um círculo preenchido usando a função ImageFillToBorder(). (Note que isso é um círculo, não uma elipse, porque os parâmetros de largura e altura têm o mesmo valor.)

```
<?php
header("Content-type: image/gif");
```

```
$imagem = ImageCreate(500, 100);
$branco = ImageColorAllocate($imagem, 255, 255, 255);
ImageColorTransparent($imagem, $branco);
$vermelho = ImageColorAllocate($imagem, 20, 93, 233);
ImageArc($imagem, 40, 50, 50, 50, 0, 360, $vermelho);
ImageFillToBorder($imagem, 50, 40, $vermelho);
ImageGif($imagem);
ImageDestroy($imagem);
?>
```

Observação: os exemplos acima foram retirados do livro PHP: Guia do Desenvolvedor que está na lista de livros recomendados da PHPBrasil. Nos exemplos só foram alterados os nomes de algumas variáveis para facilitar a compreensão.

Bom, esse é o básico do módulo GD. Você com certeza tem muito a explorar ainda, em breve vou trazer mais alguns artigos sobre o assunto, para os que se interessaram, ou não entenderam alguma função podem ver no manual do PHP todas as funções de imagem:
http://br.php.net/manual/pt_BR/ref.image.php

Referência

http://www.phpbrasil.com/articles/print.php/id/164

Trabalhando em PHP com/Números

De Wikibooks

11.11 - Trabalhando com Números em PHP

Muito Cuidado ao Lidar com Números em Ponto Flutuante

Teste em PHP

```
<?php
echo (int) ((0.1 + 0.7 ) * 10);
?>
```

Agora teste isso:

echo (int) ((0.2 + 0.7) * 10);

Não conclua muito apressadamente que é deficiência do PHP.

Neste momento devemos ter conhecimento de como se comportam os números, especialmente os floats, que são normalizados pelo IEEE.

Teste em Java

```
class teste {
    public static void main(String[] args) {
        System.out.println((int) ((0.1 + 0.7 ) * 10)); //Display the string.
    }
}
```

Em Java também dá o mesmo resultado do PHP, o que leva a crer que a coisa não depende da linguagem mas das normas de como foram construídos os números pelo IEEE.

O Effective Java sugere que se use int, long ou BigDecimal para representar os valores monetários. A classe BigDecimal foi desenvolvida para resolver dois tipos de problemas associados a números de ponto flutuante (floats e doubles): primeiro, resolve o problema da inexatidão da representação de números decimais; segundo, pode ser usado para trabalhar com números com mais de 16 dígitos significativos. Em compensação, utilizar BigDecimal pode tornar o programa menos legível por não haver sobrecarga dos operadores matemáticos para ela, sendo necessário usar métodos da classe. Veja, por exemplo, como você faria o programa da listagem 1 com BigDecimal:

BigDecimal d1 = new BigDecimal("1.95");

BigDecimal d2 = new BigDecimal("1.03");

System.out.println(d1.subtract(d2));

Utilizar os primitivos normalmente é mais rápido e mais prático, mas o problema fica por conta da definição das casas decimais. Você pode controlar diretamente as casas decimais, por exemplo, utilizando como unidade para os valores o centavo ao invés de real. Um int ou um long passariam a representar a quantidade de centavos presentes no valor, e não a quatidade de reais. Por exemplo:

long l1 = 195;

long l2 = 103;

System.out.println(l1 ? l2);

Listagem 6: Programa da listagem 1 com long

As variáveis acima dizem que você tem 195 centavos (e não R$ 1,95) e vai gastar 103 centavos, e não R$ 1,03. No final você ficará com 92 centavos (e não R$ 0,92).

Agora veja as recomendações do manual do PHP

O tamanho de um float depende também da plataforma e é de 64bits no formato IEEE(*). Nunca compare números em ponto flutuante em igualdades, sob penade cometer erros.

Teste com PostgreSQL

SELECT CAST((0.1 + 0.7)*10 AS INTEGER);

Este sim, retorna o valor esperado.

Em Java:

System.out.println(1.95 - 1.03); // Retorna errado e em PHP retorna OK.

Em Ruby

(1.8+0.1)==(1.9) retorna false

O mesmo ocorre em Phyton.

Trabalhando em PHP com/Strings

De Wikibooks

Conteúdo

- 1 substr -- Retorna uma parte de uma string
- 2 substr_replace
- 3 Encontrar Posição de caractere em String
- 4 Contando Ocorrências de Substring em String
- 5 Trocando Ponto por Vírgula e vice-versa
- 6 Conversão de Strings
- 7 Trabalhando com os Caracteres de Strings
- 8 Validação de Caracteres
- 9 ctype_alnum - Checa por caracteres alfanuméricos
- 10 ctype_alpha - Checa por caracteres alfabéticos
- 11 ctype_digit - Checa por caracteres numéricos
- 12 ctype_lower - Checa por caracteres minúsculos
- 13 ctype_punct - Checa por Caracteres que não sejam espaço em branco nem alfanuméricos
- 14 ctype_space - Checa por espaços em branco
- 15 Validação de Tipos
- 16 Cases
- 17 Índices com Str_Pad
- 18 String para TimeStamp

substr -- Retorna uma parte de uma string

string substr (string string, int start [, int length])

Exemplo 1. Uso básico de substr()

```
<?php
$rest = substr("abcdef", 1);    // retorna "bcdef"
$rest = substr("abcdef", 1, 3); // retorna "bcd"
$rest = substr("abcdef", 0, 4); // retorna "abcd"
$rest = substr("abcdef", 0, 8); // retorna "abcdef"

// Outra opção é acessar atravéz de chaves
$string = 'abcdef';
echo $string{0};                // retorna a
echo $string{3};                // retorna d
?>
```

Se start for negativo, a string retornada irá começar no caractere start a partir do fim de string.

Exemplo 2. Usando um inicio negativo

```
<?php
$rest = substr("abcdef", -1);    // retorna "f"
$rest = substr("abcdef", -2);    // retorna "ef"
$rest = substr("abcdef", -3, 1); // retorna "d"
?>
```

Exemplo 3. Usando um length negativo

```
<?php
$rest = substr("abcdef", 0, -1);  // retorna "abcde"
$rest = substr("abcdef", 2, -1);  // retorna "cde"
$rest = substr("abcdef", 4, -4);  // retorna ""
$rest = substr("abcdef", -3, -1); // retorna "de"
?>

<h2>Sobrescrevendo Strings</h2>

str_replace

str_replace --  Substitui todas as ocorrências da string de procura com a string
de substituição

mixed str_replace ( mixed pesquisa, mixed substitui, mixed assunto [, int &count]
)

<pre>
<?php
// Fornece: <body text='black'>
$bodytag = str_replace("%body%", "black", "<body text='%body%'>");

// Fornece: Hll Wrld f PHP
$vowels = array("a", "e", "i", "o", "u", "A", "E", "I", "O", "U");
$onlyconsonants = str_replace($vowels, "", "Hello World of PHP");

// Fornece: você comeria pizza, cerveja e sorvete todos os dias
$frase  = "você comeria frutas, vegetais, e fibra todos os dias.";
$saudavel = array("frutas", "vegetais", "fibra");
$saboroso  = array("pizza", "cerveja", "sorvete");

$novafrase = str_replace($saudavel, $saboroso, $frase);

// Uso do parâmetro count está disponível no PHP 5.0.0
$str = str_replace("ll", "", "good golly miss molly!", $count);
echo $count; // 2
?>
```

substr_replace

substr_replace -- Substitui o texto dentro de uma parte de uma string

string substr_replace (string string, string replacement, int start [, int length])

```
<?php
$var = 'ABCDEFGH:/MNRPQR/';
echo "Original: $var<hr>\n";

/* Estes dois exemplos substituem tudo de $var com 'bob'. */
echo substr_replace($var, 'bob', 0) . "<br>\n";
echo substr_replace($var, 'bob', 0, strlen($var)) . "<br>\n";

/* Insere 'bob' direto no começo de $var. */
echo substr_replace($var, 'bob', 0, 0) . "<br>\n";

/* Estes dois exemplos substituem 'MNRPQR' em $var com 'bob'. */
echo substr_replace($var, 'bob', 10, -1) . "<br>\n";
echo substr_replace($var, 'bob', -7, -1) . "<br>\n";

/* Deleta 'MNRPQR' de $var. */
echo substr_replace($var, '', 10, -1) . "<br>\n";
?>
```

Encontrar Posição de caractere em String

strpos

strpos -- Encontra a posição da primeira ocorrência de uma string

int strpos (string str, string procurar [, int offset])

Exemplos strpos()

```
<?php
//$str = 'abc';
$str = 'cba';
$procurar  = 'a';
$posicao = strpos($str, $procurar);

// Note o uso de ===.  Simples == não funcionaria como esperado
// por causa da posição de 'a' é 0 (primeiro) caractere.
if ($pos === false) {
   echo "A string '$procurar' não foi encontrada na string '$str'";
} else {
   echo "A string '$procurar' foi encontrada na string '$str'";
   echo " e está na posição $posicao";
}

?>
```

```
<?php

//$email = 'ribafs@gmail.com.br';
$email = 'ribafs@gmail.com';
$usuario = substr ($email, 0, strpos ($email, '@'));
// Lembrando: substr ( string string, int start [, int length] )
$dominio = substr ($email, strpos ($email, '@')+1);
echo "Usuário '$usuario' e Domínio '$dominio'"; // o comprimento default é até o
final
?>
```

Contando Ocorrências de Substring em String

substr_count -- Conta o número de ocorrências de uma substring

int substr_count (string str, string conte_me)

substr_count() retorna o número de vezes que a substring conte_me ocorre na string str.

```
<?php
  $str = "Olá mundo do PHP";

  if (substr_count($str, "do") == 0)
   echo "nenhum";

  // same as:

  if (strpos($str, "do") === false)
   echo "nenhum";
?>
```

Exemplo 1. Exemplo substr_count()

```
<?php
print substr_count("This is a test", "is"); // mostra 2
?>
```

Trocando Ponto por Vírgula e vice-versa

Se temos campos tipo moeda, devemos exibir com vírgula e gravar no banco com ponto.

Para isso uma boa saída é usar a dupla de funções implode e explode.

Antes de exibir na tela (em consultas):

$f_custo_produtivo=explode(".",$f_custo_produtivo);

$f_custo_produtivo=implode(",",$f_custo_produtivo);

Antes de gravar no banco (inclusão e atualização):

$f_custo_produtivo=explode(",",$f_custo_produtivo);

$f_custo_produtivo=implode(".",$f_custo_produtivo);

Conversão de Strings

```
$foo = 1 + "10.5";echo $foo."<br>";             // $foo é float (11.5)
$foo = 1 + "-1.3e3";echo $foo."<br>";              // $foo é float (-1299)
$foo = 1 + "bob-1.3e3";echo $foo."<br>";          // $foo é integer (1)
$foo = 1 + "bob3";echo $foo."<br>";                // $foo é integer (1)
$foo = 1 + "10 Small Pigs";echo $foo."<br>";      // $foo é integer (11)
$foo = 4 + "10.2 Little Piggies";echo $foo."<br>"; // $foo é float (14.2)
$foo = "10.0 pigs " + 1;echo $foo."<br>";          // $foo é float (11)
$foo = "10.0 pigs " + 1.0;echo $foo."<br>";        // $foo é float (11)
```

Trabalhando com os Caracteres de Strings

```
// Pega o primeiro caracter da string
$str = 'Isto é um teste.';
$first = $str{0};
echo $first."<br>";
// Pega o terceiro caracter da string
$third = $str{2};
echo $third."<br>";
// Pega o último caracter da string
$str = 'Isto ainda é um teste.';
$last = $str{strlen($str)-1};
echo $last."<br>";
// Modifica o ultimo caracter da string
$str = 'Olhe o mal';
echo $str{strlen($str)-1} = 'r';
```

Validação de Caracteres

```
ctype_alnum
ctype_alpha
ctype_cntrl
ctype_digit
ctype_graph
ctype_lower
ctype_print
ctype_punct
ctype_space
ctype_upper
ctype_xdigit
```

ctype_alnum - Checa por caracteres alfanuméricos

```
$strings = array('AbCd1zyZ9', 'foo!#$bar');

foreach ($strings as $testcase) {
   if (ctype_alnum($testcase)) {
       echo "The string $testcase consists of all letters or digits.\n";
   } else {
       echo "The string $testcase does not consist of all letters or digits.\n";
   }
}
```

ctype_alpha - Checa por caracteres alfabéticos

```
$strings = array('KjgWZC', 'arf12');
foreach ($strings as $testcase) {
   if (ctype_alpha($testcase)) {
       echo "The string $testcase consists of all letters.\n";
   } else {
       echo "The string $testcase does not consist of all letters.\n";
   }
}
```

ctype_digit - Checa por caracteres numéricos

```
$strings = array('1820.20', '10002', 'wsl!12');
foreach ($strings as $testcase) {
   if (ctype_digit($testcase)) {
       echo "The string $testcase consists of all digits.\n";
   } else {
       echo "The string $testcase does not consist of all digits.\n";
   }
}
// Alerta: Ao executar veja que somente é válido quando todos são dígitos
// Não é indicado para testar valores decimais, com ponto ou vírgula
```

ctype_lower - Checa por caracteres minúsculos

```
$strings = array('aac123', 'qiutoas', 'QASsdks');
foreach ($strings as $testcase) {
   if (ctype_lower($testcase)) {
       echo "The string $testcase consists of all lowercase letters.\n";
   } else {
       echo "The string $testcase does not consist of all lowercase letters.\n";
   }
}
```

ctype_punct - Checa por Caracteres que não sejam espaço em branco nem alfanuméricos

```
$strings = array('ABasdk!@!$#', '!@ # $', '*&$()');
foreach ($strings as $testcase) {
   if (ctype_punct($testcase)) {
       echo "The string $testcase consists of all punctuation.\n";
   } else {
       echo "The string $testcase does not consist of all punctuation.\n";
   }
}
```

ctype_space - Checa por espaços em branco

Validação de Tipos

```
intval
is_array
is_bool
is_callable
is_double
is_float
is_int
is_integer
is_long
is_null
is_numeric
is_object
is_real
is_resource
is_scalar
is_string
isset
print_r
serialize
settype
strval
unserialize
unset
```

Cases

```
strtoupper($str) - tudo maiúsculo
strtolower($str) - tudo minúsculo
ucfirst($str) - Converte para maiúscula o primeiro caractere de uma STRING
ucwords($STR) - Converte para maiúsculas o primeiro caractere de cada PALAVRA
```

Índices com Str_Pad

str_pad -- Preenche uma string para um certo tamanho com outra string

string str_pad (string input, int pad_length [, string pad_string [, int pad_type]])

Exemplo:

$players =

```
  array("DUNCAN, king of Scotland"=>"Larry",
        "MALCOLM, son of the king"=>"Curly",
        "MACBETH"=>"Moe",
        "MACDUFF"=>"Rafael");
```

echo "

```
";

// Print a heading
echo str_pad("Dramatis Personae", 50, " ", STR_PAD_BOTH) . "\n";

// Print an index line for each entry
foreach($players as $role=>$actor)
  echo str_pad($role, 30, ".")
      . str_pad($actor, 20, ".", STR_PAD_LEFT) . "\n";
```

echo "

";

Resultado:

```
                    Dramatis Personae
DUNCAN, king of Scotland.....................Larry
MALCOLM, son of the king.....................Curly
MACBETH.......................................Moe
MACDUFF.....................................Rafael
```

String para TimeStamp

```
// Absolute dates and times
$var = strtotime("25 December 2002");
$var = strtotime("14/5/1955");
$var = strtotime("Fr1, 7 Sep 2001 10:28:07 -1000");

// The current time: equivalent to time(  )
$var = strtotime("now");

// Relative times
echo strtotime("+1 day");
echo strtotime("-2 weeks");
echo strtotime("+2 hours 2 seconds");
```

```
//Care should be taken when using strtotime( ) with user-supplied dates. It's
better to limit the use of strtotime( ) to cases when
//the string to be parsed is under the control of the script, for example,
checking a minimum age using a relative date:
// date of birth: timestamp for 16 August, 1983
$dob = mktime(0, 0, 0, 16, 8, 1982);

// Now check that the individual is over 18
if ((float)$dob < (float)strtotime("-18 years"))
  echo "Legal to drive in the state of Victoria";
```

Trabalhando em PHP com/Permissões de arquivos e diretórios

De Wikibooks

Conteúdo

- 1 chmod - altera permissões de arquivos e diretórios
- 2 chown
- 3 chgrp -- Modifica o grupo do arquivo
- 4 is_writable -- Diz se pode-se escrever para o arquivo (writable)
- 5 umask -- Modificar a umask atual

chmod - altera permissões de arquivos e diretórios

```
<?php
chmod ("/arquivo/diretorio", 755);  // decimal; provavelmente incorreto
chmod ("/arquivo/diretorio", "u+rwx,go+rx"); // string; incorreto
chmod ("/arquivo/diretorio", 0755);  // octal; representaÃ§Ã£o correta do modo
?>

function permissoes($arquivo,$perms,$acao){
        print "<form name=frm method=post action=acoes.php>";
        print "<input name=pm value=$perms>";
        print "<input type=hidden name=perms value=$perms>";
        print "<input type=hidden name=ar value=$arquivo>";
        print "<input type=hidden name=acao value=$acao>";
        print "<input name=ar value=$arquivo readonly style='background-
color:#FAEBD7'>";
        print "<input type=submit name=prm value=Alterar>";
        print "</form>";

        if (isset($_POST['prm'])){
                $ar=$_POST['ar'];
                $perms=octdec($_POST['pm']);
                $ch = chmod($ar, $perms);
                if(!$ch) {
                        die ("Erro ao alterar as permissÃ¬¬µes!");
                }else{
                        print "<script>location='index.php'</script>";
                }
        }
}
```

```
<?php
// Escrita e leitura para o proprietario, nada ninguem mais
chmod ("/somedir/somefile", 0600);

// Escrita e leitura para o proprietario, leitura para todos os outros
chmod ("/somedir/somefile", 0644);

// Tudo para o proprietario, leitura e execucao para os outros
chmod ("/somedir/somefile", 0755);

// Tudo para o proprietario, leitura e execucao para o grupo do prop
chmod ("/somedir/somefile", 0750);
?>

Value    Permission Level
400    Owner Read
200    Owner Write
100    Owner Execute
40    Group Read
20    Group Write
10    Group Execute
4    Global Read
2    Global Write
1    Global Execute


<?php
function chmodnum($mode) {
   $mode2=$mode;
   $realmode = "";
   $legal =  array("","w","r","x","-");
   $attarray = preg_split("//",$mode);
   for($i=0;$i<count($attarray);$i++){
       if($key = array_search($attarray[$i],$legal)){
           $realmode .= $legal[$key];
       }
   }
   $mode = str_pad($realmode,9,'-');
   $trans = array('-'=>'0','r'=>'4','w'=>'2','x'=>'1');
   $mode = strtr($mode,$trans);
   $newmode = '';
   $newmode .= $mode[0]+$mode[1]+$mode[2];
   $newmode .= $mode[3]+$mode[4]+$mode[5];
   $newmode .= $mode[6]+$mode[7]+$mode[8];
   return $mode2.' = '.$newmode;
}

echo chmodnum('drwxr-xr-x');
?>

alguns exemplos:

drwxr-xr-x => 755
drwxr-xr-x => 755
dr-xr-xr-x => 555
drwxr-xr-x => 755
drwxr-xr-x => 755
```

```
drwxr-xr-x => 755
drwxr-xr-x => 755
drwxrwxrwt => 776
drwxr-xr-x => 755
drwxr-xr-x => 755
lrwxrwxrwx => 777
```

chown

Esta função não trabalha com arquivos remotos

```
<?php

$file_name= "test";
$path = "/var/www/html/test/" . $file_name ;

$user_name = "root";

chown($path, $user_name);

?>

<?php
function recurse_chown_chgrp($mypath, $uid, $gid)
{
   $d = opendir ($mypath) ;
   while(($file = readdir($d)) !== false) {
       if ($file != "." && $file != "..") {

           $typepath = $mypath . "/" . $file ;

           //print $typepath. " : " . filetype ($typepath). "<BR>" ;
           if (filetype ($typepath) == 'dir') {
               recurse_chown_chgrp ($typepath, $uid, $gid);
           }

           chown($typepath, $uid);
           chgrp($typepath, $gid);

       }
   }

 }

recurse_chown_chgrp ("uploads", "ribafs", "meugrupo") ;
?>

<?php
function recurse_chown_chgrp($path2dir, $uid, $gid){
   $dir = new dir($path2dir);
   while(($file = $dir->read()) !== false) {
       if(is_dir($dir->path.$file)) {
           recurse_chown_chgrp($dir->path.$file, $uid, $gid);
```

```
        } else {
            chown($file, $uid);
            chgrp($file, $gid);
        }
    }
    $dir->close();
}
?>
```

chgrp -- Modifica o grupo do arquivo

filegroup -- Lê o grupo do arquivo

fileperms -- Lê as permissões do arquivo

fileowner -- Lê o dono (owner) do arquivo

is_readable -- Diz se o arquivo/diretório é legivel (readable)

```
<?php
if (is_readable('my_link')) {
  header('Location: /my_link');
}
?>
```

is_writable -- Diz se pode-se escrever para o arquivo (writable)

```
<?php

$file = '/home/vincent/arquivo.sh';

if(is_executable($file)) {
   echo $file.' Ã© executÃ¡vel';
} else {
   echo $file.' nÃ£o Ã© executÃ¡vel';
}

?>
```

umask -- Modificar a umask atual

```
<?php
umask(0670);                     //- set umask
$handle = fopen('file', 'w');  //- 0006
mkdir("/path/dir");            //- 0107
?>
```

```
calculate the result:
<?php
$umask = 0670;
umask($umask);
//- if you are creating a new directory, $permission = 0777;
//- if you are creating a new file, $permission = 0666.
printf( "result: %04o", $permission & ( 0777 - $umask) );
?>
```

Trabalhando em PHP com/Validação de Dados

De Wikibooks

Ótimo artigo em 5 partes na Revista do PHP de autoria do Er Abbott

- Validação de Formulários - 5º e última parte: Validando no servidor

- Validação de Formulários - Parte 04 (A fonteira cliente/servidor)

- Validação de Formulários - Parte 3: O baile de máscaras

- Validação de Formulários - Parte 2: Os Campos Especiais

- Validação de formulários - Parte 1 (O Planejamento)

Turbinando a Validação de Formulários

Tipos de Variáveis

```
<?php

function ss_array_as_string (&$array, $column = 0) {
    $str = "Array(<BR>\n";
    while(list($var, $val) = each($array)){
        for ($i = 0; $i < $column+1; $i++){
            $str .= "    ";
        }
        $str .= $var.' ==> ';
        $str .= ss_as_string($val, $column+1)."<BR>\n";
    }
    for ($i = 0; $i < $column; $i++){
        $str .= "    ";
    }
    return $str.')';
}

function ss_object_as_string (&$object, $column = 0) {
    if (empty($object->classname)) {
        return "$object";
    }
    else {
        $str = $object->classname."(<BR>\n";
```

```
        while (list(,$var) = each($object->persistent_slots)) {
            for ($i = 0; $i < $column; $i++){
                $str .= "    ";
            }
            global $$var;
            $str .= $var.' ==> ';
            $str .= ss_as_string($$var, column+1)."<BR>\n";
        }
        for ($i = 0; $i < $column; $i++){
            $str .= "    ";
        }
        return $str.')';
    }
}

function ss_as_string (&$thing, $column = 0) {
    if (is_object($thing)) {
        return ss_object_as_string($thing, $column);
    }
    elseif (is_array($thing)) {
        return ss_array_as_string($thing, $column);
    }
    elseif (is_double($thing)) {
        return "Double(".$thing.")";
    }
    elseif (is_long($thing)) {
        return "Long(".$thing.")";
    }
    elseif (is_string($thing)) {
        return "String(".$thing.")";
    }
    else {
        return "Unknown(".$thing.")";
    }
}

$my_variable=3;
//echo ss_as_string($my_variable);
echo ss_as_string($GLOBALS);
?>
```

Trabalhando em PHP com/Arrays

De Wikibooks

Conteúdo

- 1 Algumas das funções
- 2 array_fill -- Preenche um array com valores
- 3 array_merge -- Funde dois ou mais arrays
- 4 array_pad -- Expande um array para um certo comprimento utilizando um determinado valor
- 5 array_pop -- Retira um elemento do final do array
- 6 array_push -- Adiciona um ou mais elementos no final de um array
- 7 array_reverse -- Retorna um array com os elementos na ordem inversa
- 8 array_search
- 9 array_shift -- Retira o primeiro elemento de um array
- 10 array_sum -- Calcula a soma dos elementos de um array
- 11 array -- Cria um array
- 12 arsort
- 13 asort
- 14 count -- Conta o número de elementos de uma variável
- 15 current -- Retorna o elemento corrente em um array
- 16 each -- Retorna o par chave/valor corrente de um array e avança o seu cursor
- 17 Percorrendo um array com each()
- 18 end -- Faz o ponteiro interno de um array apontar para o seu último elemento
- 19 key -- Retorna uma chave da posição atual de um array associativo
- 20 next -- Avança o ponteiro interno de um array
- 21 prev -- Retrocede o ponteiro interno de um array
- 22 sizeof -- Apelido de count()
- 23 sort -- Ordena um array pelo seu valor
- 24 Exemplo de array multidimensional
- 25 Exemplo de Array

Um array é uma variável, mas diferente das demais ele armazena uma coleção de valores e não somente um. E ainda por cima podem conter outras variáveis e de tipos diferentes.
Detalhe importante: Quando em uma funçãoprecisarmos rtornar mais de um valor array é a saída, basta retornar todos os valors em forma de array.

Além disso é semelhante ao que estudamos na matemática: linhas e colunas. Matriz 3x4 (3 linhas e 4

colunas).

Um array no PHP é um mapa ordenado, que relaciona valores com chaves (em linhas e colunas).

Especificando um array()

array([chave =>] valor, ...);

A chave pode ser uma string ou um inteiro.

O valor pode ser qualquer coisa.

Algumas das funções

Essas funções abaixo permitem a interação e manipulação de arrays de várias formas. Arrays são essenciais para armazenar, gerenciar, operar sobre um conjunto de variáveis.

Arrays (matrizes) simples e multidimensionais são suportados e podem ser criados pelo usuário ou por outras funções. Existem diversas funções específicas para bancos de dados, que preenchem arrays com os dados retornados em consultas, e vários outros tipos de funções também retornam arrays.

array_fill -- Preenche um array com valores

array array_fill (int start_index, int num, mixed value)

```
<?php
$a = array_fill(5, 6, 'banana');
print_r($a);
?>
```

array_merge -- Funde dois ou mais arrays

array array_merge (array array1, array array2 [, array ...])

```
<?php
$array1 = array();
$array2 = array(1 => "data");
$result = array_merge($array1, $array2);
?>
```

Não esqueça que as chaves numéricas serão reordenadas!

```
Array
(
   [0] => data
)
```

Se você quer preservar os arrays e apenas concatená-los, o operador +:

```
<?php
$array1 = array();
$array2 = array(1 => "data");
$result = $array1 + $array2;
?>
```

As chaves numéricas serão preservadas e as associações originais permanecem.

array_pad -- Expande um array para um certo comprimento utilizando um determinado valor

array array_pad (array input, int pad_size, mixed pad_value)

Exemplo 1. Exemplo de array_pad()

```
<?php
$input = array(12, 10, 9);

$result = array_pad($input, 5, 0);
// $result é array(12, 10, 9, 0, 0)

$result = array_pad($input, -7, -1);
// $result é array(-1, -1, -1, -1, 12, 10, 9)

$result = array_pad($input, 2, "noop");
// Não será expandido
?>
```

array_pop -- Retira um elemento do final do array

mixed array_pop (array array)

```
<?php
$cesta = array("laranja", "banana", "melancia", "morango");
$fruta = array_pop($cesta);
print_r($cesta);
?>
```

array_push -- Adiciona um ou mais elementos no final de um array

int array_push (array array, mixed var [, mixed ...])

```
<?php
$cesta = array("laranja", "morango");
```

```
array_push($cesta, "melancia", "batata");
print_r($cesta);
?>
```

array_reverse -- Retorna um array com os elementos na ordem inversa

array array_reverse (array array [, bool preserve_keys])

```
<?php
$input = array("php", 4.0, array ("verde", "vermelho"));
$result = array_reverse($input);
$result_keyed = array_reverse($input, TRUE);
print_r($result_keyed);
?>
```

array_search

-- Procura por um valor em um array e retorna sua chave correspondente caso seja encontrado. Caso contrário retorna FALSE.

mixed array_search (mixed procurar_este, array procurar_neste [, bool strict])

```
<?php
$a=array("a","b",0,"c","d");
echo "a: ".array_search("a",$a)."<br>";
echo "b: ".array_search("b",$a)."<br>";
echo "c: ".array_search("c",$a)."<br>";
echo "d: ".array_search("d",$a)."<br>";
echo "0: ".array_search("0",$a)."<br>";
echo "x: ".array_search("x",$a)."<br>";
echo "1: ".array_search("1",$a);
?>

<?php
if (array_search($needle, $array)!== FALSE) {
   //code goes here (
}
?>

<?php

function array_replace($search, $replace, &$array) {
  foreach($array as $key => $value) {
   if($value == $search) {
     $array[$key] = $replace;
   }
  }
}
```

```
?>

<?
$Projects[0] = array(123, "Text 1");
$Projects[1] = array(456, "Text 2");
$Projects[2] = array(789, "Text 3");

$search_value = "ext 3";

foreach ($Projects as $key => $row){
     foreach($row as $cell){
       if (strpos($cell, $search_value) !== FALSE){
                echo  "<p>Project ".$key;
       }
     }
 }
?>
```

array_shift -- Retira o primeiro elemento de um array

mixed array_shift (array array)

```
<?php
$cesta = array("laranja", "banana", "melancia", "morango");
$fruta = array_shift($cesta);
print_r($cesta);
?>
```

array_sum -- Calcula a soma dos elementos de um array

mixed array_sum (array arr)

```
<?php
$a = array(2, 4, 6, 8);
echo "soma(a) = ".array_sum($a)."<br>";

$b = array("a" => 1.2, "b" => 2.3, "c" => 3.4);
echo "soma(b) = ".array_sum($b)."<br>";
?>
```

array_unique -- Remove os valores duplicados de um array

array array_unique (array array)

```
<?php
```

```
$input = array("a" => "verde", "vermelho", "b" => "verde", "azul", "vermelho");
$result = array_unique($input);
print_r($result);
?>
```

Exemplo 2. array_unique() e tipos

```
<?php
$input = array(4, "4", "3", 4, 3, "3");
$result = array_unique($input);
var_dump($result);
?>
<pre>


<h2>array_values -- Retorna todos os valores de um array</h2>

array array_values ( array input )

//Retorna os valores, as chaves não
<pre>
<?php
$array = array("tamanho" => "G", "cor" => "dourado");
print_r(array_values ($array));
?>
```

array -- Cria um array

array array ([mixed ...])

Exemplo 1. Exemplo de array()

```
<?php
$frutas = array (
   "frutas" => array("a"=>"laranja", "b"=>"banana", "c"=>"maçã"),
   "numeros" => array(1, 2, 3, 4, 5, 6),
   "buracos" => array("primeiro", 5 => "segundo", "terceiro")
)
?>
```

Exemplo 2. Indexação automática com array()

```
<?php
$array = array(1, 1, 1, 1,  1, 8 => 1,  4 => 1, 19, 3 => 13);
print_r($array);
?>
```

arsort

-- Ordena um array em ordem descrescente dos valores mantendo a associação entre índices e valores

void arsort (array array [, int sort_flags])

```
<?php
$frutas = array("d" => "limao", "a" => "laranja", "b" => "banana", "c" =>
"melancia");
arsort($frutas);
reset($frutas);
while (list($chave, $valor) = each($frutas)) {
   echo "$chave = $valor\n";
}
?>
```

asort

-- Ordena um array em ordem crescente dos valores mantendo a associação entre índices e valores

void asort (array array [, int sort_flags])

```
<?php
$frutas = array("d" => "limao", "a" => "laranja", "b" => "banana", "c" =>
"melancia");
asort($frutas);
reset($frutas);
while (list($chave, $valor) = each($frutas)) {
   echo "$chave = $valor\n";
}
?>
```

count -- Conta o número de elementos de uma variável

int count (mixed var [, int mode])

```
<?php
$a[0] = 1;
$a[1] = 3;
$a[2] = 5;
$a[3] = 6;
$result = count($a);
// $result == 4
print $result."<br>";

$b[0] = 7;
$b[5] = 9;
```

```
$b[10] = 11;
$result = count($b);
// $result == 3;
print $result;
?>
```

Exemplo 2. Uso recursivo da função count() (PHP >= 4.2.0)

```
<?php
$food = array( 'fruits'  => array('orange', 'banana', 'apple'),
'veggie'  => array('carrot', 'collard','pea'));
// recursive count
echo count($food,COUNT_RECURSIVE);  // mostra 8
// normal count
echo count($food);                  // mostra2 2
?>
```

```
<?php
$food = array( 'fruits'  => array('orange', 'banana', 'apple'),
'veggie'  => array('carrot', 'collard','pea'));
// recursive count
echo count($food,COUNT_RECURSIVE)."<br>";  // mostra 8
// normal count
echo count($food);                  // mostra2 2
?>
```

current -- Retorna o elemento corrente em um array

mixed current (array array)

```
<?php
$transport = array('foot', 'bike', 'car', 'plane');
$mode = current($transport); // $mode = 'foot';
echo "Atual $mode<br>";
$mode = next($transport);    // $mode = 'bike';
echo "Atual $mode<br>";
$mode = current($transport); // $mode = 'bike';
echo "Atual $mode<br>";
$mode = prev($transport);    // $mode = 'foot';
echo "Atual $mode<br>";
$mode = end($transport);    // $mode = 'plane';
echo "Atual $mode<br>";
$mode = current($transport); // $mode = 'plane';
echo "Atual $mode<br>";
?>
```

each -- Retorna o par chave/valor corrente de um array e avança o seu cursor

array each (array array)

```
<?php
$foo = array("bob", "fred", "jussi", "jouni", "egon", "marliese");
$bar = each($foo);
print_r($bar);
?>

<?php
  $foo = array("Robert" => "Bob", "Seppo" => "Sepi");
  $bar = each($foo);
  print_r($bar);
?>
```

Percorrendo um array com each()

```
<?php
$fruit = array('a' => 'apple', 'b' => 'banana', 'c' => 'cranberry');
reset($fruit);
while (list($key, $val) = each($fruit)) {
     echo "$key => $val\n";
}
/* Saída:

  a => apple
  b => banana
  c => cranberry

  */
?>
```

end -- Faz o ponteiro interno de um array apontar para o seu último elemento

mixed end (array array)

```
<?php
  $frutas = array('melancia', 'banana', 'morango');
  print end($frutas); // morango
?>
```

key -- Retorna uma chave da posição atual de um array associativo

mixed key (array array)

```
<?php
  $array = array(
     'fruit1' => 'apple',
     'fruit2' => 'orange',
     'fruit3' => 'grape',
     'fruit4' => 'apple',
     'fruit5' => 'apple');

  // este ciclo  exibirá todas as chaves do array associativo
  // ausiliado pela função next()
  while ($fruit_name = current($array)) {
         echo key($array).'<br>';
             next($array);
  }
  ?>
```

```
<?php
  $array = array(
     'fruit1' => 'apple',
     'fruit2' => 'orange',
     'fruit3' => 'grape',
     'fruit4' => 'apple',
     'fruit5' => 'apple');

  // este ciclo  exibirá toda a chave do array associativo
  // onde o valor é igual a "apple"
  while ($fruit_name = current($array)) {
     if ($fruit_name == 'apple') {
         echo key($array).'<br>';
     }
     next($array);
  }
  ?>
```

next -- Avança o ponteiro interno de um array

mixed next (array array)

```
<?php
  $transport = array('foot', 'bike', 'car', 'plane');
  $mode = current($transport); // $mode = 'foot';
  print"$mode<br>";
  $mode = next($transport);    // $mode = 'bike';
  print"$mode<br>";
  $mode = next($transport);    // $mode = 'car';
  print"$mode<br>";
  $mode = prev($transport);    // $mode = 'bike';
```

```
  print"$mode<br>";
  $mode = end($transport);    // $mode = 'plane';
  print"$mode<br>";
?>
```

prev -- Retrocede o ponteiro interno de um array

mixed prev (array array)

```
<?php
  $transport = array('foot', 'bike', 'car', 'plane');
  $mode = current($transport); // $mode = 'foot';
print"$mode<br>";
  $mode = next($transport);    // $mode = 'bike';
print"$mode<br>";
  $mode = next($transport);    // $mode = 'car';
print"$mode<br>";
  $mode = prev($transport);    // $mode = 'bike';
print"$mode<br>";
  $mode = end($transport);    // $mode = 'plane';
print"$mode<br>";
?>
<pre>


<h2>reset --  Faz o ponteiro interno de um array apontar para o seu primeiro
elemento</h2>

mixed reset ( array array )
<pre>
<?php
  $array = array('primero passo', 'segundo passo', 'terceiro passo', 'quarto
passo');

  // por definição, o ponteiro está sobre o primeiro elemento
  echo current($array)."<br>\n"; // "Primeiro passo"

  // pula dois passos
  next($array);
  next($array);
  echo current($array)."<br>\n"; // "passo três"

  // reinicia o ponteiro, começa novamente o primeiro passo
  reset($array);
  echo "Depois de resetado...: " . current($array)."<br>\n"; // "primeiro passo"
?>
```

sizeof -- Apelido de count()

sort -- Ordena um array pelo seu valor

void sort (array array [, int sort_flags])

```
<?php

$frutas = array("limao", "laranja", "banana", "melancia");
sort($frutas);
reset($frutas);
while (list($chave, $valor) = each($frutas)) {
   echo "frutas[".$chave."] = ".$valor."<br>";
}
?>
```

Os seguintes também são funcionalmente idênticos:

```
<?php
$arr = array("one", "two", "three");
reset($arr);
while (list($key, $value) = each ($arr)) {
   echo "Chave: $key; Valor: $value<br />\n";
}

foreach ($arr as $key => $value) {
   echo "Chave: $key; Valor: $value<br />\n";
}
?>
```

Mais alguns exemplos para demonstrar os usos:

```
<?php
/* exemplo foreach 1: somente valores */

$a = array(1, 2, 3, 17);

foreach ($a as $v) {
   echo "Valor atual de \$a: $v.\n";
}

/* exemplo foreach 2: valores (com as chaves impressas para ilustração) */

$a = array(1, 2, 3, 17);

$i = 0; /* para exemplo somente */

foreach ($a as $v) {
   echo "\$a[$i] => $v.\n";
   $i++;
}
```

```
/* exemplo foreach 3: chaves e valores */

$a = array (
   "um" => 1,
   "dois" => 2,
   "três" => 3,
   "dezessete" => 17
);

foreach ($a as $k => $v) {
   echo "\$a[$k] => $v.\n";
}

/* exemplo foreach 4: arrays multidimensionais */

$a[0][0] = "a";
$a[0][1] = "b";
$a[1][0] = "y";
$a[1][1] = "z";

foreach ($a as $v1) {
   foreach ($v1 as $v2) {
       echo "$v2\n";
   }
}

/* exemplo foreach 5: arrays dinâmicos */

foreach (array(1, 2, 3, 4, 5) as $v) {
   echo "$v\n";
}
?>
```

Exemplo de array multidimensional

```
$produto[1][codigo] = "1";
$produto[1][nome] = "João Pereira Brito";
$produto[1][email] = "joao@joao.org";
$produto[1][rua] = "Vasco da Gama";
$produto[1][numero] = "1345";

$produto[2][codigo] = "2";
$produto[2][nome] = "AntÙnio queiroz";
```

Exemplo de Array

```
$i=0;
while($i < $numregs){
        $codigo=pg_result($consulta,$i,codigo);
        $nome=pg_result($consulta,$i,nome);
        $venc=pg_result($consulta,$i,vencimento);
        $apartamento=pg_result($consulta,$i,apartamento);
```

```
        $pessoas=pg_result($consulta,$i,pessoas);
        $cota_agua=pg_result($consulta,$i,cota_agua);
        $cota_condominio=pg_result($consulta,$i,cota_condominio);
        $cota_reserva=pg_result($consulta,$i,cota_reserva);

        $total = $cota_agua + $cota_condominio + $cota_reserva;
        $total = number_format($total,2, ',','.');
...
$i++;
}
```

Também podemos ter um array formado por outros arrays (neste caso, cada sub array é uma linha do principal)

```
$arrayvarios = array(
        array(1, 3, 5, 7),
        array(2, 4, 6, 8),
        array(1, 1, 1, 1)
);
```

Neste caso temos um array 2x4 (2 linhas por 4 colunas, que iniciam sempre com índice zero).

Então se queremos retornar o valor 8, que está na linha 2 e coluna 4, devemos retornar o índice 1,3 (linha2=índice 1, coluna4=índice3).

print $arrayvarios[1][3];

Agora veremos com detalhes os pares: chave => valor:

```
$alunos = array(
        "0732355" => "Ribamar FS",
        "0823456" => "AntÙnio Brito",
        "0654345" => "Roberto Queiroz"
);
```

O que isto retornaria?

```
print $alunos["0732355"];
print $alunos[0];
```

Experimente!!

Atribuindo valores às chaves de arrays

Também podemos fazer diretamente assim:

print $alunos["0732355"] = "João Brito";

Lembrando que, a chave, é exclusiva. Podemos ter

```
$alunos["0732355"] = "João Brito";
$alunos["0932355"] = "João Brito";

Mas não podemos ter:
$alunos["0732355"] = "João Brito";
$alunos["0732355"] = "Ribamar FS";
```

Anexo agora um excelente tutorial sobre Arrays do Celso Goya publicado em:
http://www.xoopstotal.com.br/modules/wfsection/article.php?articleid=51

Trabalhando com arrays

Visão geral Para facilitar o entendimento, vamos definir array como um conjunto de valores, que podem ser identificados em grupo ou então separadamente. Estes conjuntos podem ser muito úteis enquanto programamos, pois em alguns casos podem substituir uma tabela em banco de dados ou então utilizando métodos mais avançados podemos carregá-los dinâmicamente e utilizar quase como um banco de dados em memória.

A linguagem PHP oferece uma incrível gama de recursos para se trabalhar com arrays. Com destaque para as funções auxiliares que permitem fazer desde uma simples contagem de elementos até a conversão automática de um array em string.

Neste artigo desenvolveremos como exemplo uma função para gerar combo boxes com os estados do Brasil. Muitas vezes criamos uma tabela no banco de dados para armazenar a lista de estados do Brasil sendo que neste caso existe um número finito de registros e menor que 100, então as operações de banco de dados não são tão ágeis quanto o uso de um array.

Criando o primeiro array

Para utilizar um array, antes de mais nada é preciso criar uma variável do tipo array.

```
<?php
$estados = array();
?>
```

O próximo passo é montar nossa lista de estados.

```
<?php
$estados = array();
$estados[0] = "Acre";
$estados[1] = "Alagoas";
$estados[2] = "Amapá";
$estados[3] = "Amazonas";
?>
```

Os colchetes servem para identificar qual elemento do nosso conjunto estamos nos referindo e o número entre colchetes é o código identificador do elemento.

Podemos fazer o seguinte teste:

```
<?php
$estados = array();
$estados[0] = "Acre";
$estados[1] = "Alagoas";
$estados[2] = "Amapá";
$estados[3] = "Amazonas";
echo($estados[0]);
?>
```

Neste caso será exibida a palavra Acre, pois indicamos o ítem [0] da variável $estados, que é um array.

Você deve estar se perguntando "O que há de tão fantástico em um array?". Agora vamos mostrar alguns recursos.

Criando o array de estados

Nosso array não será de grande valia se não permitir que as siglas dos estados sejam armazenadas também, pois desta forma podemos guardar no banco de dados apenas os dois caractéres correspondentes à sigla do estado, ou seja, utilizaremos apenas dois bytes no banco de dados.

Então vamos criar um array com duas colunas, sendo a primeira a sigla do estado e a segunda seu nome por extenso.

```
<?php
$estados = array();
$estado[0][0] = "AC";
$estado[0][1] = "Acre";
$estado[1][0] = "AL";
$estado[1][1] = "Alagoas";
$estado[2][0] = "AP";
$estado[2][1] = "Amapá";
$estado[3][0] = "AM";
$estado[3][1] = "Amazonas";
$estado[4][0] = "BA";
$estado[4][1] = "Bahia";
$estado[5][0] = "CE";
$estado[5][1] = "Ceará";
$estado[6][0] = "DF";
$estado[6][1] = "Distrito Federal";
$estado[7][0] = "ES";
$estado[7][1] = "Espírito Santo";
$estado[8][0] = "GO";
$estado[8][1] = "Goiás";
$estado[9][0] = "MA";
$estado[9][1] = "Maranhão";
$estado[10][0] = "MG";
$estado[10][1] = "Minas Gerais";
$estado[11][0] = "MT";
$estado[11][1] = "Mato Grosso";
$estado[12][0] = "MS";
$estado[12][1] = "Mato Grosso do Sul";
$estado[13][0] = "PA";
$estado[13][1] = "Pará";
$estado[14][0] = "PR";
$estado[14][1] = "Paraná";
$estado[15][0] = "PE";
$estado[15][1] = "Pernambuco";
$estado[16][0] = "PI";
$estado[16][1] = "Piauí";
$estado[17][0] = "RJ";
```

```
$estado[17][1] = "Rio de Janeiro";
$estado[18][0] = "RN";
$estado[18][1] = "Rio Grande do Norte";
$estado[19][0] = "RS";
$estado[19][1] = "Rio Grande do Sul";
$estado[20][0] = "RO";
$estado[20][1] = "RondÙnia";
$estado[21][0] = "RR";
$estado[21][1] = "Roraima";
$estado[22][0] = "SC";
$estado[22][1] = "Santa Catarina";
$estado[23][0] = "SP";
$estado[23][1] = "São Paulo";
$estado[24][0] = "SE";
$estado[24][1] = "Sergipe";
$estado[25][0] = "TO";
$estado[25][1] = "Tocantins";
?>
```

A diferença neste exemplo é que utilizamos dois identificadores de elemento, ou seja, agora para cada elemento do array possuimos mais outros dois dependentes. Da mesma forma que criamos dois elementos o 0 e 1 para cada item de estado poderiamos criar n novos sub-elementos, por exemplo:

```
<?php
$estado = array();

$estado[0][0] = "SP";
$estado[0][1] = "São Paulo";
$estado[0][2] = "Sudeste";
?>
```

Vamos considerar à partir de agora que um array possui linhas e colunas, onde as linhas são equivalentes ao primeiro conjunto de colchetes e as colunas são equivalentes ao segundo conjunto de colchetes.

A função de exibição do combo box de estados

Agora vamos exibir todos os elementos de nosso array em uma função:

```
<?php
/*Nossa função reebe 3 parâmetros
$pNome :: Corresponde ao nome do SELECT
$pSelected :: Corresponde ao elemento que deverá possuir o status de selecionado
automaticamente
$extra :: Caso precise adicionar um style, ou então opção de multiple
*/
function renderCombo($pNome = "", $pSelected = "SP", $extra = ""){
        echo("<SELECT NAME='".$pNome."' ".$extra.">");

/*
Para exibir todos os itens do nosso combo utilizamos o comando for ,
lembre-se que como usamos números para identificar nosso array,
então podemos substituí-lo automaticamente com o for
*/
        //Realiza o loop em todos os elementos do array
```

```
        for( $i = 0; $i < 26;$i++ ){
                //Imprime a TAG OPTION usando a primeira coluna do array
                echo("<OPTION VALUE='".$estado[$i][0]."'");
                //Efetua a comparação para verificar se este é o ítem selecionado
                if( $estado[$i][0] == $pSelected ){
                        //Caso a comparação seja verdadeira seleciona o ítem
                        echo(" SELECTED");
                }
                //Imprime o nome por extenso do estado, equilavente a segunda
coluna do array
                echo(">".$estado[$i][1]."</option>\n");
        }
        //Finaliza a tag SELECT
        echo("</SELECT>\n");
}
?>
```

Eureka! Esta feita uma função para exibir um combo de estados.

Identificadores alternativos

Na linguagem PHP podemos utilizar palavras para identificar um elemento de um array, este recurso é muito bom, pois facilita muito a depuração e o entendimento de programas que utilizam arrays.

Vamos utilizar nosso array de estados como exemplo:

```
<?php
$estado[0]["sigla"] = "SP";
$estado[0]["nome"] = "São Paulo";
$estado[0]["regiao"] = "Sudeste";

echo($estado[0]["sigla"]);
?>
```

Desta forma podemos deixar o código de nossos programas mais fáceis de se compreender. Repare que utilizamos uma string simples para identificar um elemento do array, sendo assim, podemos utilizar variáveis para identificar um item do array, por exemplo:

```
<?php
$estado[0]["sigla"] = "SP";
$estado[0]["nome"] = "São Paulo";
$estado[0]["regiao"] = "Sudeste";

$variavel = "sigla";

echo($estado[0][$variavel]);
?>
```

É importante lembrar que mesmo existindo uma string para identificar um elemento do array ainda podemos utilizar números se quisermos, por exemplo:

```
<?php
```

```
$estado[0]["sigla"] = "SP";
$estado[0]["nome"] = "São Paulo";
$estado[0]["regiao"] = "Sudeste";

echo($estado[0][0]);
?>
```

Nos três casos o resultado é o mesmo, diferindo apenas no método como chamamos o array.

O que você viu neste artigo é o básico sobre arrays, caso você se interessar pelo assunto e queira dar uma pesquisada rápida na web, vai encontrar outras formas de declarar arrays bem como usos diferenciados. O XOOPS utiliza muito este recurso. É só dar uma olhada em algum arquivo xoops_version.php, que você vai encontrar um exemplo prático do uso de arrays.

Final do tutorial do Celso Goya.

Trabalhando em PHP com/XML

De Wikibooks

11.16 - Trabalhando com XML em PHP

Ler Arquivo com XML

Observação: A função simplexml_load_string está apenas disponível no **PHP5**.

```
<?php
include 'xml_lido.php'; // arquivo a ser lido

$xml = simplexml_load_string($xmlstr);

echo $xml->filme[0]->comentario;
print '<br>';
echo $xml->filme[0]->titulo;
print '<br>';
echo $xml->filme[0]->personagens[0]->personagem[0]->nome;
print '<br>';
echo $xml->filme[0]->votos[0];
print '<br>';
echo $xml->filme[0]->votos[1];
?>
```

// Outro exemplo

```
<?php
//Mostra estrutura do elemento XML
$file = "data.xml";
$depth = array();

function startElement($parser, $name, $attrs) {
   global $depth;
   for ($i = 0; $i < $depth[$parser]; $i++) {
       echo "  ";
   }
   echo "$name\n";
   $depth[$parser]++;
}

function endElement($parser, $name) {
   global $depth;
   $depth[$parser]--;
}

$xml_parser = xml_parser_create();
xml_set_element_handler($xml_parser, "startElement", "endElement");
if (!($fp = fopen($file, "r"))) {
   die("could not open XML input");
```

```
}

while ($data = fread($fp, 4096)) {
   if (!xml_parse($xml_parser, $data, feof($fp))) {
       die(sprintf("XML error: %s at line %d",
                   xml_error_string(xml_get_error_code($xml_parser)),
                   xml_get_current_line_number($xml_parser)));
   }
}
xml_parser_free($xml_parser);
?>
```

Arquivo a ser lido (xml_lido.php):

```
<?php
$xmlstr = <<<XML
<?xml version='1.0' encoding='ISO-8859-1' ?>
<filmes>
 <filme>
  <titulo>PHP: Iniciando o Parser</titulo>
  <personagens>
   <personagem>
   <nome>João de Brito</nome>
   <actor>Brito</actor>
   </personagem>
   <personagem>
   <nome>Manoel Cunha</nome>
   <actor>Manoel</actor>
   </personagem>
  </personagens>
  <comentario>
   O XML é uma linguagem. Ela é como uma linguagem de programação. Ou uma
   linguagem de script? Tudo será revelado após ler bem toda a
   documentação.
  </comentario>
  <votos type="thumbs">7</votos>
  <votos type="stars">5</votos>
 </filme>
</filmes>
XML;
?>
```

Trabalhando em PHP com/Cookies

De Wikibooks

Conteúdo

- 1 Definindo
- 2 Exemplo
- 3 Debugando Cookies
- 4 Removendo cookies
- 5 Cookies com Arrays

Definindo

Cookies são mecanismos para armazenar e consultar informações nos navegadores dos visitantes da página. O PHP atribui cookies utilizando a função **setcookie() ou setrawcookie()**, que deve ser utilizada antes da tag <html> numa página.

Qualquer cookie enviado por você para o cliente automaticamente será uma variável do PHP assim como dados de postagens GET ou POST, dependendo dos valores de register_globals e variables_order. Se você deseja assimilar vários valores em um único cookie, simplesmente acrescente [] ao nome do cookie.

Exemplo

```
<?php
$value = 'alguma coisa de algum lugar';

setcookie("TesteCookie", $value, time()+3600);  /* expira em 1 hora */
setcookie("TesteCookie", $value, time()+3600, "/home/ribafs/", ".exemplo.com",
1);
?>
```

Debugando Cookies

```
<?php
// Exibe um cookie individual
echo $_COOKIE["TesteCookie"];
echo $HTTP_COOKIE_VARS["TesteCookie"];
```

```
// Outra maneira de debugar/testar é visualizando todos os cookies
print_r($_COOKIE);
?>
```

Removendo cookies

```
<?php
// Ajustar para expirar em 1 hora
setcookie ("TesteCookie", "", time() - 3600);
setcookie ("TesteCookie", "", time() - 3600, "/home/ribafs/", ".exemplo.com", 1);
?>
```

Cookies com Arrays

```
<?php
// Ajustar cookies
setcookie("cookie[três]", "cookiethree");
setcookie("cookie[dois]", "cookietwo");
setcookie("cookie[um]", "cookieone");

// após a página ser carregada, mostra a saída
if (isset($_COOKIE['cookie'])) {
    foreach ($_COOKIE['cookie'] as $name => $value) {
        echo "$name : $value <br />\n";
    }
}
?>
```

Integração do PHP com SGBDs/MySQL

De Wikibooks

Conteúdo

- 1 Conectar ao MySQL
- 2 Fechar a Conexão
- 3 Selecionar um Banco após a Conexão
- 4 Executar Consulta SQL
- 5 Mover Ponteiro para um Registro Específico
- 6 Retornar Todos os Registos em Forma de Array
- 7 Total de Registros
- 8 Criar Banco de Dados
- 9 Nomes dos Bancos de Dados
- 10 Excluir Banco
- 11 Informações sobre os Campos
- 12 Retornar um Registro em Forma de Array de Campos
- 13 Tamanho de Campos
- 14 Nomes de Campos
- 15 Nome de Tabela
- 16 Tipos de Dados dos Campos
- 17 ID do Insert
- 18 Lista de Campos
- 19 Lista de Campos de Tabela
- 20 Lista de Tabelas de um Banco
- 21 Número de Campos
- 22 Nome de Tabela
- 23 Teste se tabela existe no MySQL

Conectar ao MySQL

Usaremos:

banco - cliente

```
create table clientes(id int primary key auto_increment, nome char(45), data
date);
INSERT INTO `clientes` VALUES (1, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (2, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (3, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (4, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (5, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (6, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (7, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (8, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (9, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (10, 'JoÃ£o Brito10', '2007-01-12');
INSERT INTO `clientes` VALUES (11, 'JoÃ£o Brito11', '2007-01-13');
INSERT INTO `clientes` VALUES (12, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (13, 'Jorge da Cunha', '2007-01-12');
INSERT INTO `clientes` VALUES (14, 'Jorge da Cunha', '2007-01-12');

Recomendações:

- Usar sempre os mesmos nomes de variáveis para mesmas funções. Exemplo: $link
para receber a conexÃ£o.
- Usar sempre as mesmas mensagens de erro para certos erros. Exemplo:
  mysql_query... 'Erro na consulta: ' . mysql_error() nÃ£o esquecer de passar a
funÃ£oo do MySQL que retorna o erro
```

```
$link = mysql_connect('localhost', 'root', '');
if (!$link) {
    die('Erro ao conectar conectar: ' . mysql_error());
}
```

Fechar a Conexão

```
print "<h2>FECHAR A CONEXÃO COM O MYSQL</h2>";
mysql_close($link);
```

Selecionar um Banco após a Conexão

```
print "<h2>SELECIONAR BANCO APÓS CONEXÃO</h2>";
$link = mysql_connect('localhost', 'root', '');
if (!$link) {
    die('Erro ao conectar conectar : ' . mysql_error());
}else{
        if(!$db_selected = mysql_select_db('cliente', $link))  die ('Erro ao
selecionar o banco cliente : ' . mysql_error());
}
```

Executar Consulta SQL

```
print "<h2>EXECUTAR CONSULTA(Qualquer consulta SQL)</h2>";
$result = mysql_query('SELECT * FROM clientes',$link); //se $link não for
especificado, será usado o ltimo aberto
if (!$result) {
    die('Erro na consulta: ' . mysql_error());
}
```

Mover Ponteiro para um Registro Específico

```
print "<h2>MOVER PARA UM REGISTRO ESPECÍFICO</h2>";
$query = 'SELECT nome, data FROM clientes';
$result = mysql_query($query,$link);
if (!$result) {
    die('Erro na consulta: ' . mysql_error());
}
/* obter as linhas (registros) em ordem reversa */
for ($i = mysql_num_rows($result) - 1; $i <= 0; $i--) {
    if (!mysql_data_seek($result, $i)) {
        echo "Não foi possível mover para a linha $i: " . mysql_error() . "\n";
        continue;
    }

    if (!($row = mysql_fetch_assoc($result))) {
        continue;
    }

    echo $row['nome'] . ' ' . $row['data'] . "<br />\n";
}
//mysql_free_result($result);
```

Retornar Todos os Registos em Forma de Array

```
print "<h2>RETORNAR TODOS OS REGISTROS COMO ARRAY</h2>";
```

```
mysql_fetch_array($result, MYSQL_BOTH); // Retorna todos os registros como array
numÃ©rico e nomes

$result = mysql_query("SELECT id, nome FROM clientes");

while ($row = mysql_fetch_array($result, MYSQL_BOTH)) {
    printf ("ID: %s  Nome: %s", $row[0], $row["nome"].'<br>');
}
```

Total de Registros

```
print "<h2>TOTAL DE REGISTROS</h2>";
$result = mysql_query("SELECT * FROM clientes", $link);
$num_rows = mysql_num_rows($result);

echo "$num_rows registros<br>";


<h2>Receber Registros Afetados por Consulta</h2>

<pre>
print "<h2>REGISTROS AFETADOS (INSERT, DELETE E UPDATE)</h2>";
mysql_query('DELETE FROM clientes WHERE id < 10');
printf("Registros excluídos: %d\n", mysql_affected_rows());

/* Com uma cláusula WHERE que nunca é verdadeira isso deve retornar 0 */
mysql_query('DELETE FROM mytable WHERE 0');
printf("Registros excluídos: %d\n", mysql_affected_rows());
```

Criar Banco de Dados

```
print "<h2>CRIAR BANCO DE DADOS</h2>";
$sql = 'CREATE DATABASE cliente2';
if (mysql_query($sql, $link)) {
    echo "O banco de dados cliente2 foi criado<br>";
} else {
    echo 'Erro criando o banco de dados: ' . mysql_error() . "<br>";
}
```

Nomes dos Bancos de Dados

```
print "<h2>NOMES DOS BD</h2>";
$db_list = mysql_list_dbs($link);

$i = 0;
$cnt = mysql_num_rows($db_list);
```

```
while ($i < $cnt) {
    echo mysql_db_name($db_list, $i) . "<br>";
    $i++;
}
```

Excluir Banco

```
print "<h2>EXCLUIR BANCO</h2>";
$sql = 'DROP DATABASE cliente2';
if (mysql_query($sql, $link)) {
    echo "O banco de dados foi excluÃdo com sucesso<br>";
} else {
    echo 'Erro ao excluir o banco de dados: ' . mysql_error() . "<br>";
}
<pre>


<h2>Retornar Número do Erro</h2>

<pre>
print "<h2>MYSQL_ERRNO</h2>";
if (!mysql_select_db("naoexistentebd", $link)) {
    echo mysql_errno($link) . ": " . mysql_error($link). "<br>";
}
```

Informações sobre os Campos

```
print "<h2>INFORMAÃ«'ES SOBRE CAMPOS</h2>";
$i = 0;
while ($i < mysql_num_fields($result)) {
    echo "InformaÃ§Ã£o para a coluna $i:<br><br>";
    $meta = mysql_fetch_field($result, $i);
    if (!$meta) {
        echo "Sem informaÃ§Ã£o disponÃvel<br>";
    }
    echo "<pre>
blob:         $meta->blob
max_length:   $meta->max_length
multiple_key: $meta->multiple_key
name:         $meta->name
not_null:     $meta->not_null
numeric:      $meta->numeric
primary_key:  $meta->primary_key
table:        $meta->table
type:         $meta->type
default:      $meta->def
unique_key:   $meta->unique_key
unsigned:     $meta->unsigned
zerofill:     $meta->zerofill


";

   $i++;


}
```

Retornar um Registro em Forma de Array de Campos

```
print "<h2>RETORNAR UM REGISTRO COM ARRAY DE CAMPOS</h2>";
$result = mysql_query("SELECT id,nome FROM clientes WHERE id = '12'");
if (!$result) {
    echo 'Erro na consulta: ' . mysql_error();
    exit;
}
$row = mysql_fetch_row($result);

echo $row[0]; // 42
echo $row[1]; // o valor do email
```

Tamanho de Campos

```
print "<h2>TAMANHO DE CAMPOS</h2>";
```

```
$result = mysql_query("SELECT id,nome FROM clientes WHERE id = '42'");
if (!$result) {
    echo 'Erro na consulta: ' . mysql_error();
    exit;
}

// Receberão tamanho do campo id como especificado no banco
// schema.
$length = mysql_field_len($result, 0);
echo $length;
```

Nomes de Campos

```
print "<h2>NOMES DE CAMPOS</h2>";
$res = mysql_query('select * from clientes', $link);

echo mysql_field_name($res, 0) . "<br>";
echo mysql_field_name($res, 2);
```

Nome de Tabela

```
print "<h2>NOME DE TABELA</h2>";
$table = mysql_field_table($result, $nomedeumcampodatabela);
echo $table; // people
```

Tipos de Dados dos Campos

```
print "<h2>TIPO DE DADOS DE CAMPO</h2>";
$result = mysql_query("SELECT * FROM clientes");
$fields = mysql_num_fields($result);
$rows   = mysql_num_rows($result);
$table  = mysql_field_table($result, 0);
echo "Sua tabela '" . $table . "' tem " . $fields . " campos e " . $rows . "
registros<br>";
echo "A tabela tem os seguintes campos:<br>";
for ($i=0; $i < $fields; $i++) {
    $name  = mysql_field_name($result, $i);
    $type  = mysql_field_type($result, $i);
    $len   = mysql_field_len($result, $i);
    $flags = mysql_field_flags($result, $i);
    echo $name . " " . $type . "  " . $len . " " . $flags . "<br>";
}
```

ID do Insert

```
print "<h2>RECEBENDO ID GERADO POR INSERT</h2>";
mysql_query("INSERT INTO clientes (nome) values ('Jorge da Cunha')");
printf("O ltimo registro inserido tem id %d<br>", mysql_insert_id());
```

Lista de Campos

```
print "<h2>LISTAR BANCOS DO MYSQL</h2>";
$db_list = mysql_list_dbs($link);

while ($row = mysql_fetch_object($db_list)) {
     echo $row->Database . "<br>";
}
```

Lista de Campos de Tabela

```
print "<h2>LISTAR CAMPOS DE TABELA</h2>";
while ($row = mysql_fetch_assoc($result)) {
    echo $row["id"].'-';
    echo $row["nome"].'-';
    echo $row["data"]."<br>";
}
```

Lista de Tabelas de um Banco

```
print "<h2>LISTAR TABELAS DE BANCO</h2>";
$sql = "SHOW TABLES FROM cliente";
$result = mysql_query($sql);

if (!$result) {
    echo "Erro no banco, nÃ£o pode listas as tabelas<br>";
    echo 'Erro no MySQL: ' . mysql_error();
    exit;
}

while ($row = mysql_fetch_row($result)) {
    echo "Tabela: {$row[0]}<br>";
}
```

Número de Campos

```
print "<h2>NÚMERO DE CAMPOS</h2>";
$result = mysql_query("SELECT id,nome FROM clientes WHERE id = '42'");
if (!$result) {
    echo 'Erro na consulta: ' . mysql_error();
    exit;
}

/* retorna 2 porque id,nome === 2 campos */
echo mysql_num_fields($result);
```

Nome de Tabela

```
print "<h2>NOME DE TABELA</h2>";
mysql_connect("localhost", "root", "");
$result = mysql_list_tables("cliente");

for ($i = 0; $i < mysql_num_rows($result); $i++) {
    echo "Tabela: ", mysql_tablename($result, $i), "<br>";
}


<h2>Automaticamente instalar uma tabela no MySQL com PHP</h2>

<pre>
$table_def = "id INT(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,";
$table_def .= "student_id INT(11) NOT NULL,";
$table_def .= "f_name TINYTEXT NOT NULL,";
$table_def .= "l_name TINYTEXT NOT NULL,";
$table_def .= "supervisor TINYTEXT NOT NULL,";
$table_def .= "building TINYTEXT NOT NULL,";
$table_def .= "email TINYTEXT NOT NULL,";
$table_def .= "score SMALLINT(6) NULL,";
$table_def .= "stamp DATETIME NOT NULL,";
$table_def .= "UNIQUE KEY id (id)";

if (!@mysql_query ("CREATE TABLE $tablename ($table_def)")) {
echo "The database table, '$tablename', could not be created.";
} else {
echo "Successfully created the '$tablename' table.";
}
```

Teste se tabela existe no MySQL

```
function table_exists ($table, $db) {
        $tables = mysql_list_tables ($db);
        while (list ($temp) = mysql_fetch_array ($tables)) {
                if ($temp == $table) {
                        return TRUE;
                }
        }
        return FALSE;
}

/** How to use it **/
if (table_exists(test_table, my_database)) {
        echo"Yes the table is there.";
}
/*
akxter, http://www.oxyscripts.com/itemdisplay.php?id=1003&code=yes

And a shorter way. In the above example mysql_list_tables is deprecated in favor
of mysql_query().
*/

// here is a much more elegant method to check if a table exists ( no error
generate)

if( mysql_num_rows( mysql_query("SHOW TABLES LIKE '".$table."'")))
{
//...
}
```

Integração do PHP com SGBDs/PostgreSQL

De Wikibooks

Conteúdo

- 1 Conectar ao Banco do SGBD PostgreSQL
- 2 Fechar a Conexão
- 3 Executar Consulta SQL
- 4 Encontrar Registros Afetados
- 5 Retornar Nome do Banco Conectado
- 6 Retornar um registro em forma de array de campos
- 7 Receber 1 registro na forma de array
- 8 Número de Campo no Array de Campos de um Registro
- 9 Tamanho Interno de Campo
- 10 Tipo de Dados de Campo
- 11 Última Mensagem de Erro da Conexão
- 12 Meta Dados de Tabela
- 13 Número de Campos de Consulta
- 14 Número de Registros
- 15 Codificação do Cliente
- 16 Alterar Codificação do Cliente
- 17 Versão do PostgreSQL

Conectar ao Banco do SGBD PostgreSQL

Usaremos:

banco - cliente

```
create table clientes(id serial primary key, nome char(45), data date);
INSERT INTO clientes VALUES (1, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (2, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (3, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (4, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (5, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (6, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (7, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (8, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (9, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (10, 'João Brito10', '2007-01-12');
INSERT INTO clientes VALUES (11, 'João Brito11', '2007-01-13');
INSERT INTO clientes VALUES (12, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (13, 'Jorge da Cunha', '2007-01-12');
INSERT INTO clientes VALUES (14, 'Jorge da Cunha', '2007-01-12');

Recomendações:

- Usar sempre os mesmos nomes de variáveis para mesmas funções. Exemplo: $dbcon
para receber a conexão.
- Usar sempre as mesmas mensagens de erro para certos erros. Exemplo:
  pg_query... 'Erro na consulta: ' . pg_last_error() não esquecer de passar a
função do MySQL que retorna o erro



<?php
print "<h2>CONECTAR AO BANCO NO POSTGRESQL</h2>";
$bdcon = pg_connect("dbname=cliente");
//conecta a um banco de dados chamado "cliente"

$con_string = "host=localhost port=5432 dbname=cliente user=postgres
password=postgres";
if(!$dbcon = pg_connect($con_string)) die ("Erro ao conectar ao
banco<br>".pg_last_error($dbcon));
//coneta a um banco de dados chamado "cliente" na máquina "localhost" com um
usuário e senha
```

Fechar a Conexão

```
print "<h2>
== FECHA ==
R CONEXÃO COM O BANCO</H2>";
//pg_close($dbcon)
```

Executar Consulta SQL

```
print "<h2>EXECUTAR CONSULTA (QUALQUER CONSULTA SQL)</h2>";
$result = pg_query($dbcon, "SELECT id, nome FROM clientes");
if (!$result) {
  echo "Erro na consulta.<br>";
  exit;
}

while ($row = pg_fetch_row($result)) {
  echo "ID: $row[0]  Nome: $row[1]";
  echo "<br />\n";
}
```

Encontrar Registros Afetados

```
print "<h2>NÚMERO DE REGISTROS AFETADOS (INSERT,DELETE E UPDATE)</h2>";
$result = pg_query($dbcon, "INSERT INTO clientes (id, nome,data) VALUES (40,
'João Brito', '2007-03-25')");
$cmdtuples = pg_affected_rows($result);
if ($cmdtuples==0){
        echo "Nenhum registro foi afetado.<br>";
}elseif($cmdtuples==1){
        echo $cmdtuples . " registro afetado.<br>";
}else{
        echo $cmdtuples . " registros foram afetados.<br>";
}
<pre>


<h2>Inserir Registros de Array em Tabela</h2>

<pre>
print "<h2>INSERIR REGISTROS DE ARRAY EM TABELA</h2>";
   //$db = pg_connect("dbname=cliente") or die("Erro ao conectar ao banco");
   $rows = pg_copy_to($dbcon, 'clientes');
   pg_query($dbcon, "DELETE FROM clientes");
   pg_copy_from($dbcon, 'clientes', $rows);
```

Retornar Nome do Banco Conectado

```
print "<h2>RETORNAR NOME DO BANCO CONECTADO</h2>";
  echo pg_dbname(); // mary

   if ($dbcon) {
       print "<br>Conectado com sucesso ao banco: " . pg_dbname($dbcon) .
         " em " .  pg_host($dbcon) . "<br/>\n";
   } else {
       print pg_last_error($dbcon);
```

```
        exit;
    }
```

Retornar um registro em forma de array de campos

```
print "<h2>RETORNAR UM REGISTRO COM CAMPOS EM ARRAY</h2>";

while ($row=pg_fetch_row($result,$i)) {
  for($j=0; $j < count($row); $j++) {
     echo $row[$j] . "<BR>;nbsp;";
  }
```

Receber 1 registro na forma de array

```
print "<h2>RECEBER 1 REGISTRO COMO ARRAY</h2>";
$result = pg_query($dbcon, "SELECT id, nome FROM clientes");
if (!$result) {
  echo "Erro na consulta.\n";
  exit;
}

$arr = pg_fetch_array($result, 0, PGSQL_NUM);
echo $arr[0] . " <- Registro 1 ID<br>";
echo $arr[1] . " <- Registro 1 Nome<br>";

print "<h2>NOME DE CAMPO</h2>";
  $res = pg_query($dbcon, "select * from clientes where id = '23'");
  $i = pg_num_fields($res);
  for ($j = 0; $j < $i; $j++) {
     echo "campo $j<br>";
     $fieldname = pg_field_name($res, $j);
     echo "nome do campo: $fieldname<br>";
     echo "tamanho na tela: " . pg_field_prtlen($res, $fieldname) . "
caracteres<br>";
     echo "tamanho no banco: " . pg_field_size($res, $j) . " bytes<br>";
     echo "tipo de dados: " . pg_field_type($res, $j) . " <br><br>";
  }
```

Número de Campo no Array de Campos de um Registro

```
print "<h2>NÚMERO DE CAMPO NO ARRAY DE CAMPOS DE UM REGISTRO</h2>";
  $res = pg_query($dbcon, "select id,nome,data from clientes where id = '18'");
  echo "Campo 'nome' é o campo número: ", pg_field_num($res, 'nome');
```

Tamanho Interno de Campo

```
print "<h2>TAMANHO INTERNO DE CAMPO</h2>";
  $res = pg_query($dbcon, "select * from clientes where id = '23'");
  $i = pg_num_fields($res);
  for ($j = 0; $j < $i; $j++) {
     echo "campo $j<br>";
     $fieldname = pg_field_name($res, $j);
     echo "nome do campo: $fieldname<br>";
     echo "tamanho na tela: " . pg_field_prtlen($res, $fieldname) . "
caracteres<br>";
     echo "tamanho no banco: " . pg_field_size($res, $j) . " bytes<br>";
     echo "tipo de dados: " . pg_field_type($res, $j) . " <br><br>";
  }
```

Tipo de Dados de Campo

```
print "<h2>TIPO DE DADOS DE CAMPO</h2>";
  // Assume 'title' is a varchar type
  $res = pg_query($dbcon, "select nome from clientes where id = '11'");
  echo "Tipo do campo nome: ", pg_field_type($res, 0);
```

Última Mensagem de Erro da Conexão

```
print "<h2>ÚLTIMA MENSAGEM DE ERRO DA CONEXÃO</h2>";
  // Query that fails
  $res = pg_query($dbcon, "select * from tabelaquenaoexiste");
  echo pg_last_error($dbcon);
```

Meta Dados de Tabela

```
print "<h2>META DADOS DE TABELA (EXPERIMENTAL)</h2>";
  $meta = pg_meta_data($dbcon, 'clientes');
  if (is_array($meta)) {
```

```
     echo '<pre>';
     var_dump($meta);
     echo '

';
 }
```

Número de Campos de Consulta

```
print "<h2>NÚMERO DE CAMPOS</h2>";
$num = pg_num_fields($result);
echo $num . " campos(s) retornados.<br>";
```

Número de Registros

```
print "<h2>NÚMERO DE REGISTROS</h2>";
$rows = pg_num_rows($result);
echo $rows . " registro(s) retornados.<br>";
```

Codificação do Cliente

```
print "<h2>RETORNAR CODIFICA«ÃO DO CLIENTE</h2>";
$encoding = pg_client_encoding($dbcon);
echo "A codificação do cliente é: ", $encoding, "<br>";
```

Alterar Codificação do Cliente

```
print "<h2>ALTERAR CODIFICA«ÃO DO CLIENTE</h2>";
//$dbcon = pg_pconnect("dbname=cliente");
if (!$dbcon) {
  echo "Erro na conexão.<br>";
  exit;
}

// Configurar a codificação do cliente para UNICODE. Dados devem ser
automaticamente convertidos
// da codificação do backend para a do frontend.
/* Valores possíveis SQL_ASCII, EUC_JP, EUC_CN, EUC_KR, EUC_TW, UNICODE,
MULE_INTERNAL,
LATINX (X=1...9), KOI8, WIN, ALT, SJIS, BIG5, WIN1250. */
pg_set_client_encoding($dbcon, LATIN1);
$result = pg_query($dbcon, "SELECT id,nome FROM clientes");
```

```
if (!$result) {
  echo "Erro na consulta.<br>";
  exit;
}

// Escrever dados em UTF-8
while ($row = pg_fetch_row($result)) {
  echo "ID: $row[0]  Nome: $row[1]";
  echo "<br/>";
}
```

Versão do PostgreSQL

```
print "<h2>VERSÃO DO POSTGRESQL</h2>";
print "Retorna um array com versão do cliente, protocolo e servidor (quando
disponíveis)<br>";

  $v = pg_version($dbcon);
  echo $v['client'];
```

Integração do PHP com SGBDs/SQLite

De Wikibooks

12.3 - Integração do PHP com SQLite

O SQLite é um SGBD leve que pode ser utilizado integrado ao PHP para pequenos e médios aplicativos.

Aqui encontrará um aplicativo com o básico CRUD em PHP com acesso ao SQLite:

http://www.ribafs.net/down/appsexemplo/sqlite/iniciante_sqlitephp.zip

Integração do PHP com SGBDs/IBM DB2 Express

De Wikibooks

```
Configuração utilizada no sistema operacional Windows XP SP2:
```

-Baixe a extensão (pecl5.2-win32-latest) do IBM DB2 para o PHP em:

-(http://pecl.php.net/package/PDO/0.3)

-(http://snaps.php.net/win32/)

- Instale a mesma (php_ibm_db2.dll) na pasta do PHP (ex:C:\web\php-5.1.4-Win32\ext);

- Edite as linhas do php.ini identificando o diretório dos arquivos de extensão

(ex: extension_dir = "c:/web/php-5.1.4-Win32/ext/") e habilite a linha que refere-se ao db2

(extension=php_ibm_db2.dll) retirando o comentário(;);

- Depois é só aproveitar os recursos do PHP e DB2:

- Exemplo de conexão (conecta.php):

```
<?php
function dbconnect($verbose) {
    $dbname = "Nome_Banco";
    $username = "Usuario_de_Conexao";
    $password = "Senha";

    $dbconn = odbc_connect($dbname, $username, $password);

    if (($verbose == TRUE) && ($dbconn == 0)) {
        $sqlerror = odbc_errormsg($dbconn);
    }
    return($dbconn);
   }
?>
```

- Exemplo de Inserção:

```
<?php
include("conecta.php");

  $sql = "INSERT INTO estado VALUES(3,'Rio Grande do Sul')";

  $verbose = TRUE;
  $dbconn = dbconnect($verbose);

  if ($dbconn != 0) {
```

```
     $result = odbc_exec($dbconn, $sql);

     if ($result == 0) {
        echo("INSERT Falhou !!");
        $sqlerror = odbc_errormsg($dbconn);
        echo($sqlerror);
     }
     else {
          echo("INSERT realizado com sucesso !!");
     }
}
?>
```

Recursos Extras/Controle de Estoque

De Wikibooks

13.1 - Controle de Estoque

Exemplo de controle de Estoque no Viva o Linux - http://www.vivaolinux.com.br/scripts/verScript.php?codigo=694

LinuxStok - Gestão Comercial - http://linuxstok.wikidot.com/

Stoq - Gestão Comercial - Instalação e configuração - http://www.vivaolinux.com.br/artigos/impressora.php?codigo=6767

Implantação de Controle de Estoque em uma Clínica Odontológica - http://www.em.ufop.br/depro/curso/monografias/2003renata.pdf

Sistemas de Controle de Estoque no superdownloads - Não são em PHP mas ajudam a ter idéia como a coisa funciona e algumas funcionalidades que devemos implementar:

http://superdownloads.uol.com.br/busca/controle-de-estoque.html

Recursos Extras/Loja Virtual

De Wikibooks

osCommerce

Site oficial - http://www.oscommerce.com/

Download da versão traduzida pelo pessoal do OmeuNuke - http://www.omeunuke.com/modules.php?name=Downloads&op=getit&lid=22

Adicionais - http://www.omeunuke.com/modules.php?name=Downloads&cid=5

Módulos de Pagamento (boleto, transferência bancária, BRPay e contra entrega) - http://www.omeunuke.com/modules.php?name=Downloads&cid=8

Templates - http://www.omeunuke.com/modules.php?name=Downloads&cid=9

Tutoriais:

Instalaço do osCommerce

http://www.omeunuke.com/modules.php?name=Sections&op=printpage&artid=29

Outros tutoriais: http://www.omeunuke.com/modules.php?name=Sections&op=listarticles&secid=5

Forums - http://www.omeunuke.com/modules.php?name=Forums

Obs.: Sempre ao acessar cada uma dessas páginas role a tela até o final da mesma.

Obs2.: Faça seu registro no site omeunuke para ter acesso ao conteúdo e arquivos.

VirtueMart

Este funciona com o Joomla. Existe uma versão do Joomla apenas com o a loja virtual VirtueMart:

http://virtuemart.net/index.php?option=com_docman&task=cat_view&gid=101&Itemid=66

Basta instalar em uma instalação tipo a do Joomla.

Recursos Extras/Código de Barras

De Wikibooks

Exemplo de Código de Barras em PHP - do Sidiney Sodré Gali de Souza

```
<?php
/*
Autor: Sidiney Sodré Gali de Souza <sidsodre@bol.com.br>
Data: 21/11/2005

Código de Barras - código-fonte   em:
http://www.vivaolinux.com.br/scripts/verFonte.php?codigo=1394
*/

function CodigoBarras($code) {
  $lw = 2; $hi = 100;
  $Lencode = array('0001101','0011001','0010011','0111101','0100011',
                   '0110001','0101111','0111011','0110111','0001011');
  $Rencode = array('1110010','1100110','1101100','1000010','1011100',
                   '1001110','1010000','1000100','1001000','1110100');
  $ends = '101'; $center = '01010';
  /* UPC-A Must be 11 digits, we compute the checksum. */
  if ( strlen($code) != 11 ) { die("UPC-A Must be 11 digits."); }
  /* Compute the EAN-13 Checksum digit */
  $ncode = '0'.$code;
  $even = 0; $odd = 0;
  for ($x=0;$x<12;$x++) {
   if ($x % 2) { $odd += $ncode[$x]; } else { $even += $ncode[$x]; }
  }
  $code.=(10 - (($odd * 3 + $even) % 10)) % 10;
  /* Create the bar encoding using a binary string */
  $bars=$ends;
  $bars.=$Lencode[$code[0]];
  for($x=1;$x<6;$x++) {
   $bars.=$Lencode[$code[$x]];
  }
  $bars.=$center;
  for($x=6;$x<12;$x++) {
   $bars.=$Rencode[$code[$x]];
  }
  $bars.=$ends;
  /* Generate the Barcode Image */
  $img = ImageCreate($lw*95+30,$hi+30);
  $fg = ImageColorAllocate($img, 0, 0, 0);
  $bg = ImageColorAllocate($img, 255, 255, 255);
  ImageFilledRectangle($img, 0, 0, $lw*95+30, $hi+30, $bg);
  $shift=10;
  for ($x=0;$x<strlen($bars);$x++) {
   if (($x<10) || ($x>=45 && $x<50) || ($x >=85)) { $sh=10; } else { $sh=0; }
```

```
   if ($bars[$x] == '1') { $color = $fg; } else { $color = $bg; }
   ImageFilledRectangle($img, ($x*$lw)+15,5,($x+1)*$lw+14,$hi+5+$sh,$color);
  }
  /* Add the Human Readable Label */
  ImageString($img,4,5,$hi-5,$code[0],$fg);
  for ($x=0;$x<5;$x++) {
   ImageString($img,5,$lw*(13+$x*6)+15,$hi+5,$code[$x+1],$fg);
   ImageString($img,5,$lw*(53+$x*6)+15,$hi+5,$code[$x+6],$fg);
  }
  ImageString($img,4,$lw*95+17,$hi-5,$code[11],$fg);
  /* Output the Header and Content. */
  header("Content-Type: image/png");
  ImagePNG($img);
}

CodigoBarras('54879874585');
?>
```

Outro exemplo, este do Marcolino

```
<?php

/* Geração de código de barras padrão p/ boletos bancários brasileiros.

Autor: MARCOLINO, Alexandre de Jesus ( marcolino@facil.com )

Este código pode ser usado e copiado livremente desde que site e comunique o
autor por email.

Testado sob o Internet Explorer 6 e o Netscape 7.

*/

function montacodigodebarras($valor) {
  $lw = 1 ; $hi = 50;
/* Tabela de código de barras Padrão Brasileiro para boletos bancários

  Barra valor 00 =0000111100
  Barra valor 01 =0100101001
  Barra valor 02 =0001101001
  Barra valor 03 =0101101000
  Barra valor 04 =0000111001
  Barra valor 05 =0100111000
  Barra valor 06 =0001111000
  Barra valor 07 =0000101101
  Barra valor 08 =0100101100
  Barra valor 09 =0001101100
  Barra valor 10 =1000010110
  Barra valor 11 =1100000011
  Barra valor 12 =1001000011
  Barra valor 13 =1101000010
  Barra valor 14 =1000010011
  Barra valor 15 =1100010010
```

```
Barra valor 16 =1001010010
Barra valor 17 =1000000111
Barra valor 18 =1100000110
Barra valor 19 =1001000110
Barra valor 20 =0010010110
Barra valor 21 =0110000011
Barra valor 22 =0011000011
Barra valor 23 =0111000010
Barra valor 24 =0010010011
Barra valor 25 =0110010010
Barra valor 26 =0011010010
Barra valor 27 =0010000111
Barra valor 28 =0110000110
Barra valor 29 =0011000110
Barra valor 30 =1010010100
Barra valor 31 =1110000001
Barra valor 32 =1011000001
Barra valor 33 =1111000000
Barra valor 34 =1010010001
Barra valor 35 =1110010000
Barra valor 36 =1011010000
Barra valor 37 =1010000101
Barra valor 38 =1110000100
Barra valor 39 =1011000100
Barra valor 40 =0000110110
Barra valor 41 =0100100011
Barra valor 42 =0001100011
Barra valor 43 =0101100010
Barra valor 44 =0000110011
Barra valor 45 =0100110010
Barra valor 46 =0001110010
Barra valor 47 =0000100111
Barra valor 48 =0100100110
Barra valor 49 =0001100110
Barra valor 50 =1000110100
Barra valor 51 =1100100001
Barra valor 52 =1001100001
Barra valor 53 =1101100000
Barra valor 54 =1000110001
Barra valor 55 =1100110000
Barra valor 56 =1001110000
Barra valor 57 =1000100101
Barra valor 58 =1100100100
Barra valor 59 =1001100100
Barra valor 60 =0010110100
Barra valor 61 =0110100001
Barra valor 62 =0011100001
Barra valor 63 =0111100000
Barra valor 64 =0010110001
Barra valor 65 =0110110000
Barra valor 66 =0011110000
Barra valor 67 =0010100101
Barra valor 68 =0110100100
Barra valor 69 =0011100100
Barra valor 70 =0000011110
Barra valor 71 =0100001011
Barra valor 72 =0001001011
```

```
  Barra valor 73 =0101001010
  Barra valor 74 =0000011011
  Barra valor 75 =0100011010
  Barra valor 76 =0001011010
  Barra valor 77 =0000001111
  Barra valor 78 =0100001110
  Barra valor 79 =0001001110
  Barra valor 80 =1000011100
  Barra valor 81 =1100001001
  Barra valor 82 =1001001001
  Barra valor 83 =1101001000
  Barra valor 84 =1000011001
  Barra valor 85 =1100011000
  Barra valor 86 =1001011000
  Barra valor 87 =1000001101
  Barra valor 88 =1100001100
  Barra valor 89 =1001001100
  Barra valor 90 =0010011100
  Barra valor 91 =0110001001
  Barra valor 92 =0011001001
  Barra valor 93 =0111001000
  Barra valor 94 =0010011001
  Barra valor 95 =0110011000
  Barra valor 96 =0011011000
  Barra valor 97 =0010001101
  Barra valor 98 =0110001100
  Barra valor 99 =0011001100
*/

/* A lógica abaixo aplica a tabela acima para dentro da variável tabcodbarra{} */

  $tabcodbarra[0] = "00110" ;
  $tabcodbarra[1] = "10001" ;
  $tabcodbarra[2] = "01001" ;
  $tabcodbarra[3] = "11000" ;
  $tabcodbarra[4] = "00101" ;
  $tabcodbarra[5] = "10100" ;
  $tabcodbarra[6] = "01100" ;
  $tabcodbarra[7] = "00011" ;
  $tabcodbarra[8] = "10010" ;
  $tabcodbarra[9] = "01010" ;

  for($f1=9;$f1>=0;$f1--){
    for($f2=9;$f2>=0;$f2--){
      $f = ($f1 * 10) + $f2 ;
      $texto = "" ;
      for($i=1;$i<6;$i++){
        $texto .=  substr($tabcodbarra[$f1],($i-1),1) .
substr($tabcodbarra[$f2],($i-1),1);
      }
      $tabcodbarra[$f] = $texto;
   }
  }


/* Gerando a Imagen do Código de Barras */
```

```
/* Gera a base para o preenchimento do código.
$img    -> imagem do código
$branco -> cor branca
$preto  -> cor preta ( ou afrobrasileira, sem preconceito nenhum pÙ )
*/
      $img = ImageCreate($lw*95+1000,$hi+30);

      $preto  = ImageColorAllocate($img, 0, 0, 0);
      $branco = ImageColorAllocate($img, 255, 255, 255);

/* Cria o retángulo principal onde a imagem será aplicada. */

      ImageFilledRectangle($img, 0, 0, $lw*95+1000, $hi+30, $branco);

# Abertura do código de barras.
      ImageFilledRectangle($img, 1,5,1,65,$preto);
      ImageFilledRectangle($img, 2,5,2,65,$branco);
      ImageFilledRectangle($img, 3,5,3,65,$preto);
      ImageFilledRectangle($img, 4,5,4,65,$branco);

# Aqui começa a varredura no valor enviado para confecção do código de barras.

$fino = 1 ;
$largo = 2.72 ; // Para o windows funciona mas o tamanho ideal seria 3
$pos   = 5 ;
$texto = $valor ;
if((strlen($texto) % 2) <> 0){
        $texto = "0" . $texto;
}

// Draw dos dados
while (strlen($texto) > 0) {
  $i = round(esquerda($texto,2));
  $texto = direita($texto,strlen($texto)-2);

  $f = $tabcodbarra[$i];

  for($i=1;$i<11;$i+=2){
    if (substr($f,($i-1),1) == "0") {
      $f1 = $fino ;
    }else{
      $f1 = $largo ;
    }

/* Imprimindo uma barra preta */
  ImageFilledRectangle($img, $pos,5,$pos-1+$f1,65,$preto)  ;
  $pos = $pos + $f1 ;
/* Fim da impressão da barra preta */

  if (substr($f,$i,1) == "0") {
      $f2 = $fino ;
    }else{
      $f2 = $largo ;
    }

/* Imprimindo uma barra branca */
  ImageFilledRectangle($img, $pos,5,$pos-1+$f2,65,$branco)  ;
```

```
  $pos = $pos + $f2 ;
/* Fim da impressão da barra branca */

  }
}

# Fechamento do código de barras.

ImageFilledRectangle($img, $pos,5,$pos-1+$largo,65,$preto);
$pos=$pos+$largo;

ImageFilledRectangle($img, $pos,5,$pos-1+$fino,65,$branco);
$pos=$pos+$fino;


ImageFilledRectangle($img, $pos,5,$pos-1+$fino,65,$preto);
$pos=$pos+$fino;


/* MANDA A IMAGEM PRO BROWSER */
      header("Content-Type: image/png");
      ImagePNG($img);
}
//montacodigodebarras('23794225800000041933394060092001280100001020');

//$valorpassado = $_SERVER['QUERY_STRING'] ;

montacodigodebarras('23791227400000041933394060092001280200001020');

montacodigodebarras($_SERVER['QUERY_STRING']) ;

function esquerda($entra,$comp){
        return substr($entra,0,$comp);
}

function direita($entra,$comp){
        return substr($entra,strlen($entra)-$comp,$comp);
}

?>
```

Recursos Extras/Boleto bancário

De Wikibooks

13.4 - Boleto Bancário

PHP Boleto

O objetivo desse projeto é de criar um sistema de geração e manutenção de boletos. Ele poderá ser usado por sites de e-Commerce ou empresas em geral para simplificar o modelo de pagamento e incentivar os clientes a pagarem seus boletos pela própria Web.

http://sourceforge.net/projects/phpboleto

Contribuindo: Encontrei esse projeto e espero que seja útil. http://www.boletophp.com.br/

Recursos Extras/PHP-GTK

De Wikibooks

Através da biblioteca GTK podemos utilizar o PHP para criação de aplicativos desktop.

PHP-GTK na Wikipedia

PHP-GTK é uma extensão para a linguagem de programação PHP que implementa o binding da linguagem para o GTK+. Proporciona uma interface orientada a objeto para as classes e funções do GTK+ e simplifica, e muito, o desenvolvimento de aplicações desktop multi-plataforma. Seu autor original é Andrei Zmievski e o desenvolvimento está em andamento.

Usos significantes do PHP-GTK incluem:

```
* Agata Report - Uma aplicação de geração de relatórios;

* Tulip - Um ambiente de desenvolvimento para PHP e PHP-GTK;
* LinuxStok - Controle de Estoque e Financeiro.
```

Links externos

```
* Website oficial do PHP-GTK
* Comunidade brasileira de PHP-GTK
```

Ferramenta RAD para criação de Interfaces em PHP-GTK

Disponível para Windows e Linux

Site oficial - http://anubis.php-gtk.com.br/

Anubis é um aplicativo RAD (Rapid Application Development) IDE (Integrated Development Environment) auxilia o desenvolvimento de aplicações na linguagem PHP-GTK - com ele você "desenha" as telas de seus aplicativos e o código é gerado automaticamente na linguagem - isto implica em maior rapidez na execução do programa pois não há necessidade de se interpretar arquivos externos como os do glade.

Instalação do PHP-GTK no Windows

Download - http://www.php-gtk.com.br/windows

Executar o Instalador PhpGtk2setup.exe

Veja a estrutura criada:

C:\php-gtk2

```
    php.exe = Interpretador PHP5 com suporte ao GTK2;
    php-gtk-brasil.exe = Nosso programa de boas-vindas;
    php.ini = arquivo de configuração do PHP;
    \ext = extensões do PHP;
```

As aplicações php-gtk podem ser inicializadas pela linha de comando:

C:\php-gtk2\php C:\php-gtk2\start.php

Ou sem a janela do MS-DOS:

C:\php-gtk2\php-win C:\php-gtk2\start.php

Recursos Extras/Contadores de visita

De Wikibooks

Contadores

Contador de Acesso em PHP

Do Ândres Oliveira na Revista do PHP.

http://revistaphp.com.br/artigo.php?id=139

Criamos um contador, programado em PHP, que leva a conta das impressões que foram realizadas numa página web, utilizando um arquivo de texto como apoio.

O objetivo é levar uma contagem das vezes que uma página foi visitada.

Colocamos o seguinte script PHP ao final da página, se entenderá bem se for lido o artigo citado antes.

```
<?php
  $arquivo = "contador.txt";
  $contador = 0;
  $fp = fopen($arquivo,"r");
  $contador = fgets($fp, 26);
  fclose($fp);
  ++$contador;
  $fp = fopen($arquivo,"w+");
  fwrite($fp, $contador, 26);
  fclose($fp);
  echo "Esta página foi visitada $contador vezes";
?>
```

www.revistaphp.com.br

Ademais, criamos um arquivo chamado "contador.txt" que salvamos no mesmo diretório que a página. Tal arquivo foi inicializado com um zero (0) como único texto.

Nota: Se tiver problemas na hora de escrever em um arquivo, quase com toda probabilidade, estará protegido contra escritura. Ou o arquivo ou então o diretório.

Se você tiver seu próprio servidor terá que modificar as permissões do tal arquivo ou diretório por sua conta. Entretanto, se estiver publicando em uma hospedagem contratada em uma provedor terá que se informar de que mecanismo há que colocar em funcionamento nesse provedor para conseguir as

permissões. Em muitos casos existirá um painel de controle onde modificar essas coisas, em outros casos terá que escrever ao suporte técnico para que eles façam a mão ou que digam como fazer, se é que é possível.

Com isso já está feito um contador muito simples, mas muito funcional.

Site: www.netcenterdesign.com.br

Recursos Extras/Livros de visita

De Wikibooks

Livros de Visitas em PHP

Exemplo do Éder - http://www.phpbrasil.com/scripts/script.php/id/93

Livro de Visitas Prático - http://www.phpbrasil.com/scripts/script.php/id/126

Livro de Visitas Personalizável - http://www.phpbrasil.com/scripts/script.php/id/183

Livro de Visitas com Admin - http://www.phpbrasil.com/scripts/script.php/id/2557

Livro de Visitas TXT - http://www.phpbrasil.com/scripts/script.php/id/2832

Advanced GuestBook - http://www.planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=1362&lngWId=8

Recursos Extras/Chats

De Wikibooks

Chats em PHP

PHPOpenChat - Chat bom em PHP

Download - http://phpopenchat.sourceforge.net/

Faça o download e descompacte no diretório web em phpopenchat

Abrir no browser - http://127.0.0.1/phpopenchat/

- Clicar em use the installer - Alterar a permissão de escrita para: - config.inc.php - tmp - include/templates/openchat/images/icons/chatter

- Clique em next

A great and simple online chat

http://www.Planet-Source-Code.com/vb/scripts/ShowCode.asp?txtCodeId=1091&lngWId=8

Um Simples Chat sem Banco de Dados

http://www.phpbrasil.com/articles/article.php/id/166

Com este exstem diversos outros exemplos de chat no PHPBrasil.

Recursos Extras/Geradores de Relatórios

De Wikibooks

Conteúdo

- 1 BIRT - Business Intelligence and Report Tools
  - 1.1 Criando um Novo Projeto do Tipo Relatório
  - 1.2 1 - Criando um relatório partindo de um blank report design
  - 1.3 2 - Adicionar um Data Source (Fonte de dados)
  - 1.4 3 - Criar um Data Set
  - 1.5 4 - Layout do relatório
  - 1.6 5 - Ordenando registros
  - 1.7 6 - Adicionando Título ao Relatório
- 2 Utilizando o BIRT com PHP ou simplesmente via Web
  - 2.1 Tutorial de Instalação e uso do Tomcat e do BIRT
  - 2.2 Instalação e Configuração do J2SE
  - 2.3 Instalando e configurando o Tomcat
  - 2.4 Testando o Tomcat
  - 2.5 Administrando o Tomcat
  - 2.6 Instalando e usando o Visualizador de Relatórios do BIRT
  - 2.7 Instalando o Driver JDBC do seu SGBD
  - 2.8 Abrindo Relatório de Script PHP
  - 2.9 Passando parâmetros
  - 2.10 Parâmetros de formulários
  - 2.11 Opções avançadas
- 3 Gerador de Relatório iReport
  - 3.1 Ótimo tutorial sobre o iReport, inclusive com o acesso via web (com java)
  - 3.2 Tutorial de iReport
  - 3.3 Starting with JasperReports by Gregory Beumer
  - 3.4 Manual do iReport
  - 3.5 iReport Documentação
  - 3.6 Tutorial JasperReports - IReports
  - 3.7 Outro tutorial em português
  - 3.8 A Tutorial on Generating Reports by iReport

BIRT - Business Intelligence and Report Tools

O BIRT é um gerador de relatórios aos moldes do Crystal Report e do iReport, que inclusive tem integração com o PHP e outras linguagens server side.

Download - http://download.eclipse.org/birt/downloads/

Criando um Novo Projeto do Tipo Relatório

- New - Project - Business Intelligence and Report Tools - Report Project

- Clicar sobre o nome do Projeto criado com o botão direito e New - Report

1 - Criando um relatório partindo de um blank report design

- Crie um novo projeto

- Crie um novo Relatório

- Em File name entre com clientes.rptdesign e Next

- Em Report template deixe Blank Report e finish

- Observe que a janela de Layout (ao centro e à direita) exibe o nome do relatório acima e seu conteúdo está vazio. À esquerda temos a Paleta de Ferramentas para o design do relatório. Logo à direita da Paleta temos a aba Data Explorer, para a conexão com bancos de dados.

2 - Adicionar um Data Source (Fonte de dados)

Do tipo JDBC do PostgreSQL, para usar um banco de dados no nosso relatório.

- Clicar na aba Data Explorer (à direita da Paleta)

- Clicar com botão direito - New DataSource

- Selecionar JDBC Data Source - Digitar um nome (Usuário) e Next

- Deve antes ter o jdbc para o postgreSQL em algun diretório

- Clicar em Manage Drivers

- Clicar em Add e indicar o driver jdbc do portgresql

- Selecionar e OK

- Em Driver Class selecionar o driver correto

- Em Database URL entre com algo como:

jdbc:postgresql://127.0.0.1:5432/db_usuario

- Digite username e password do postgresql

- Clique em Test Connection

- Se tudo ok clique em OK

3 - Criar um Data Set

- Clicar em Data Sets com o botão direito e New DataSet

- Em Data set name entre com um nome (Usuário Ceará)

- Em Data source aceite o default

- Em data set type também aceite o default e clique em Next

- Na caixa da direita clique após o select e digite "nome, email". Clique após o from e dê um duplo clique no nome da tabela. Ao invés de digitar os campos também podemos efetuar duplos clique nos mesmos.

Para ter uma consulta dinâmica adicione parâmetros assim: "select nome,email from public.usuario WHERE nome LIKE " + params["nome"]

- Clique em Finish. Apenas clique em OK.

- Duplo clique no data set clientesceara à esquerda e clique em Preview Results para visualizar os registros e se certificar de que está tudo ok.

Veja que também existe suporte a JOIN. Basta clicar com o botão direito sobre Data Sets e New Join Data Set

4 - Layout do relatório

Tabelas (tables) interagem com todos os registros retornados por um data set. Isto torna fácil a exibição de registros em forma de linhas e colunas.

- Arraste o controle Table da paleta e solte no editor de layout. Será perguntado sobre quantas colunas e quantos detalhes. Deixe com 2 e 1 e OK e feche o Data Explorer, pois usaremos somente os campos nome e email.

- No Data Explorer, expanda Data Sets, então expanda Usuário Ceará. As colunas especificadas na consulta aparecem abaixo de Usuário Ceará.

- Arraste "nome" para a primeira célula do detalhe. De forma que fique na segunda linha e apareça o

label na primeira.

Para mudar a formatação de qualquer componente do relatório apenas selecione e clique no botão abaixo, em Properties - Font ou outra formatação.

Clique em Preview para ver como está.

Também para uma visualização profissional vá em File - View

5 - Ordenando registros

- Abra o report no layout editor e selecione abaixo properties. No editor clique em Table (abaixo e à esquerda da área da tabela).

- Ao mover o mouse por essa região aparece Table, então clique nesse botão Table. Abaixo aparecerão as propriedades da Tabela.

- Clique na Aba Sorting. Clique em Add para adicionar uma expressão. Em Key selecione "nome" e deixe Accendente.

- Execute o Preview para ver o resultado.

Observe que nomes iniciados com maiúsculas aparecem no início da lista. O BIRT ordena dados tipo string usando códigos UCS2. Em conjuntos de caracteres baseados em ASCII, as letras maiúsculas têm valor semelhante às minúsculas. Para ordenar case-insensitivamente usar:

row["nome"].toUpperCase( )

Na expressão para ordenar.

6 - Adicionando Título ao Relatório

Para isso usaremos um controle Text e tags HTML.

Selecione o controle Text na Paleta e arraste para cima da tabela.

Ao aparecer o diálogo digite:

```
<CENTER><B><span style="font-size: larger"> Relatório de Usuários do DNOCS
</B></span><BR>

<FONT size="small">Somente para uso interno</FONT><BR><BR> Relatório Gerado em
<VALUE-OF>new Date( )</VALUE-OF></CENTER><BR><BR>
```

Utilizando o BIRT com PHP ou simplesmente via Web

Tutorial de Instalação e uso do Tomcat e do BIRT

J2SE Download - http://java.sun.com/javase/downloads/index.jsp

Aqui para baixar somente o JDK (baixei o JDK 5.0 Update 9)

https://sdlc2d.sun.com/ECom/EComActionServlet;jsessionid=65966D4F31A5BD1A572B8A9AB6B2BBDB# (Windows)

https://sdlc2d.sun.com/ECom/EComActionServlet;jsessionid=65966D4F31A5BD1A572B8A9AB6B2BBDB# (Linux)

Tomcat Download - http://tomcat.apache.org/

Direto na versão 5.5.20 - http://tomcat.apache.org/download-55.cgi#5.5.20

http://mirrors.uol.com.br/pub/apache/tomcat/tomcat-5/v5.5.20/bin/apache-tomcat-5.5.20.tar.gz (Linux)

http://mirrors.uol.com.br/pub/apache/tomcat/tomcat-5/v5.5.20/bin/apache-tomcat-5.5.20.exe (Windows)

Caso sua versão do J2SE seja 1.4 ou superior baixar também o JDK 1.4 Compatability Package:

http://mirrors.uol.com.br/pub/apache/tomcat/tomcat-5/v5.5.20/bin/apache-tomcat-5.5.20-compat.tar.gz (Linux)

http://mirrors.uol.com.br/pub/apache/tomcat/tomcat-5/v5.5.20/bin/apache-tomcat-5.5.20-compat.zip (Windows)

Instalação e Configuração do J2SE

Executar

Mover para o diretório /opt

sudo mv jdk-1_5_0_09-linux-i586.bin

chmod u+s

sudo ./jdk-1_5_0_09-linux-i586.bin

Editar seu script de inicialização para adicionar a variável de ambiente JAVA_HOME:

sudo gedit /etc/bash.bashrc

JAVA_HOME=/opt/jdk1.5.0_09

CLASSPATH=.:$JAVA_HOME/lib/tools.jar

CATALINA_HOME=/home/ribafs/prog/tomcat # antes e após o igual não pode haver espaços

PATH=$JAVA_HOME/bin:$PATH

export JAVA_HOME CLASSPATH PATH CATALINA_HOME

Para surtir efeito restart o ambiente gráfico fazendo um logoff ou Ctrl+Alt+Backspace.

Instalando e configurando o Tomcat

- Descompactar apache-tomcat-5.5.20.tar.gz

- Criar a variável de ambiente para o Tomcat:

- Descompactar o apache-tomcat-5.5.20-compat.tar.gz no raiz do diretório do tomcat /home/ribafs/prog/tomcat

Testando o Tomcat

Apontar o browser para o endereço http://localhost:8080

Administrando o Tomcat

Edite o arquivo /home/ribafs/prog/tomcat/conf/tomcat-users.xml e adicionar última linha abaixo:

<?xml version='1.0' encoding='utf-8'?> <tomcat-users>

```
 <role rolename="tomcat"/>
 <role rolename="role1"/>
 <role rolename="manager"/>
 <role rolename="admin"/>
 <user username="tomcat" password="tomcat" roles="tomcat"/>
 <user username="role1" password="tomcat" roles="role1"/>
 <user username="both" password="tomcat" roles="tomcat,role1"/>
 <user username="admin" password="admin" fullName="Tomcat Administrator"
roles="admin,manager"/>
```

</tomcat-users>

Com usuário e senha (admin para ambos)

Abrir no browser em http://127.0.0.1:8080/admin/

Entre com admin e admin

Instalando e usando o Visualizador de Relatórios do BIRT

Download - http://download.eclipse.org/birt/downloads/ (procure por Report Engine):

http://www.eclipse.org/downloads/download.php?file=/birt/downloads/drops/M-R1-2.2M1-200610171021/birt-runtime-2.2M1.zip

Observe que estou usando a versão 2.2M1 do BIRT.

Descompactar e copiar o subdiretório WebViewerExample para o diretório /home/ribafs/prog/tomcat/webapps.

Renomear o WebViewerExample para birt-viewer

Abra o link no browser http://localhost:8080/manager/html

Entre com admin e admin.

À esquerda clique em /birt-viewer e veja que abre-se o visualizador web de relatórios do BIRT.

Caso deseje abrir um relatório seu já criado, copie para /home/ribafs/prog/tomcat/webapps/birt-viewer e chame no browser assim:

http://localhost:8080/birt-viewer/frameset?__report=seurelatorio.rptdesign&parametro=valordoparametro

Instalando o Driver JDBC do seu SGBD

Copie para a pasta:

/home/ribafs/prog/tomcat/webapps/WEB-

INF/platform/plugins/org.eclipse.birt.report.data.oda.jdbc_2.2.0.v20061009-0630/drivers

postgresql-8.1-407.jdbc3.jar (no meu caso)

Abrindo Relatório de Script PHP

```
<?php
$fname = "./clientes.rptdesign";
// Redirect browser
$dest = "http://localhost:8080/birt-viewer/run?__report=";
$dest .= urlencode( realpath( $fname ) );
header("Location: $dest" );
?>
```

Passando parâmetros

```
<?php
$fname = "./clientes.rptdesign";
// Redirect browser
$dest = "http://localhost:8080/birt-viewer/run?__report=";
$dest .= ";sample=" . urlencode( $paramValue );
$dest .= urlencode( realpath( $fname ) );
header("Location: $dest" );
?>
```

Parâmetros de formulários

Deally, we could use PHP to create a UI form that prompts for the report parameters. The BIRT viewer creates this form in Java using information in the report design. Unfortunately, at present, there is no way to retrieve the raw parameter descriptions from the BIRT viewer using a URL. Instead, there are two alternatives we can use.

First, if we know the parameters ahead of time, we can design a custom form in PHP that prompts the user for them. This works if we have a small number of reports, or if we need to create a specialized parameter page for each report anyway.

Second, we can let the BIRT viewer display the parameter page using the frameset URL. Generating Reports Dynamically

Finally, PHP provides one additional BIRT integration option: the ability to generate report designs dynamically for a specific task. For example, suppose you have a bug tracking system, and you'd like your user to create their own reports via the web. You can ask the user for the columns to display, then use PHP to create a BIRT report design customized to display those columns. PHP is ideal for this: it allows us to insert scripting directly into HTML. Since a BIRT design is XML, and XML is close

enough to HTML for PHP, we can "trick" PHP into generating a BIRT report design instead of an HTML page.

To generate a report design, do the following:

- Create a BIRT report design typical of the kind of report you want to create.
- Create a PHP template file that contains this design. Insert the contents of the design file in place of the HTML you'd usually put into a PHP file.
- Redirect PHP's output from the template file into a report design file.
- Use PHP to generate BIRT XML for the table headings and cells the user wants. Use your report design as a template for what is needed. Consult the ROM spec for details on various elements and properties.
- Within the data set in your template, create an SQL query that fetches the required columns. (Works with other data set types as well.)
- Use the code above to redirect the browser to run that report using the BIRT viewer.

The following PHP code redirects the output of a PHP page, template.inc, into a report design called temp.rptdesign:

```
ob_start( );
require "template.inc";
$page = ob_get_contents( );
ob_end_clean( );
$fw = fopen( "temp.rptdesign", "w" );
fputs( $fw, $page, strlen( $page ) );
fclose( $fw );
```

Opções avançadas

http://localhost:8080/birt-viewer/run?__report=report%5CSalesInvoice.rptdesign&OrderNumber=10010

List of Options

The available viewer options include:

```
Option  Description     Values  Default         frameset        run
__format        The output format       html or pdf     html    N       Y
__isnull        Identifies that a report parameter has a null value     Parameter
name  None. Required.         N       Y
__locale        Report locale   Java locale value such as en, en-us or ch-zh.
JVM locale      Y       Y
__report        The path to the report design.          None. Required.         Y
Y
__document      The path to the report document.                None. Required.
Y       N
reportParam     Report parameter.       As specified in the report design.
As specified in the report design.      Y       Y
```

Veja os originais para maiores detalhes:

http://www.eclipse.org/birt/phoenix/deploy/viewerSetup.php

http://www.eclipse.org/birt/phoenix/deploy/usingPHP.php

http://www.eclipse.org/birt/phoenix/deploy/viewerUsage.php

Gerador de Relatório iReport

Site oficial - http://jasperforge.org/sf/projects/ireport

Ótimo tutorial sobre o iReport, inclusive com o acesso via web (com java)

www.furutani.eti.br/tutoriais/Mini-Tutorial_Relatorios_Java_JasperReports_e_iReport.pdf

Tutorial de iReport

http://www.javafree.org/javabb/viewtopic.jbb?t=3154

Starting with JasperReports by Gregory Beumer

http://technology.amis.nl/blog/index.php?p=346

Manual do iReport

http://jasperforge.org/sf/wiki/do/viewPage/projects.ireport/wiki/IReportManualV1.2.1

iReport Documentação

http://jasperforge.org/sf/wiki/do/viewPage/projects.ireport/wiki/HomePage

Tutorial JasperReports - IReports

HTML - http://br.geocities.com/robertofurutani/java/Tutorial_JasperReports/

PDF - http://www.furutani.eti.br/tutoriais/Tutorial_Sub_Relatorio_ArrayList.pdf

Fontes - http://br.geocities.com/robertofurutani/java/Tutorial_JasperReports/Tutorial.zip

Outro tutorial em português

http://www.livramento.yu.com.br/tutoriais/ireport.html

A Tutorial on Generating Reports by iReport

http://www.cise.ufl.edu/~otopsaka/CIS4301/ReportDemo/

Recursos Extras/Integração com AJAX

De Wikibooks

Conteúdo

- 1 Palavras Acentuadas com AJAX
- 2 Migrating Web-Based PHP Applications to Ajax
- 3 Ajax e PHP - Carregando dados sem refresh
- 4 Exemplos de AJAX em PHP utilizando XAJAX
- 5 Ajax+Php sites
- 6 Busca PHP com Ajax
- 7 Explain the Difference AND How it works between POST and GET in AJAX
- 8 AJAX - Pesquisa Em Tempo Real
- 9 Ajax com PHP - Do Iniciante ao Profissional
- 10 CEP AJAX
- 11 Frameworks com suporte a AJAX
  - 11.1 Dojo
- 12 Artigos Relacionados a AJAX no site da RevistaPHP
- 13 Engatinhando com AJAX e PHP. Parte 1
- 14 Parte 2 do tutorial AJAX + PHP

Palavras Acentuadas com AJAX

Dica de Fabrício Nogueira Magri no PHPBrasil

Migrating Web-Based PHP Applications to Ajax

http://www.onlamp.com/pub/a/php/2007/05/10/migrating-web-based-php-applications-to-ajax.html

Ajax e PHP - Carregando dados sem refresh

http://www.imasters.com.br/artigo/3918/javascript/ajax_e_php_-_carregando_dados_sem_refresh//imprimir/

Exemplos de AJAX em PHP utilizando XAJAX

http://www.japs.etc.br/2006/02/exemplos-de-ajax-em-php-utilizando-xajax/

Ajax+Php sites

Estes websites são escolhidos e avaliados por contribuidores de pt.blinklist.com.

http://pt.blinklist.com/tag/ajax+php/

Busca PHP com Ajax

http://www.conexaomx.com.br/artigos/ajax/busca-php_ajax

Explain the Difference AND How it works between POST and GET in AJAX

http://www.planet-source-code.com/vb/scripts/ShowZip.asp?lngWId=8&lngCodeId=1961&strZipAccessCode=tp%2FB19613132

AJAX - Pesquisa Em Tempo Real

http://www.php5.com.br/?php=_scripts/mostra&id_s=105

Ajax com PHP - Do Iniciante ao Profissional

http://www.altabooks.com.br/pdf/ajax_php.pdf

CEP AJAX

http://www.imasters.com.br/forum/lofiversion/index.php/t172741.html

Frameworks com suporte a AJAX

Dojo

Download - http://dojotoolkit.org/

Adding Dojo to Your Pages - http://dojotoolkit.org/book/dojo-book-0-4/part-2-out-box-dojo/getting-started/adding-dojo-your-pages

Artigos Relacionados a AJAX no site da RevistaPHP

http://revistaphp.com.br/artigos_resultado.php?c=2

Engatinhando com AJAX e PHP. Parte 1

http://felipetonello.com/blog/2006/11/14/engatinhando-com-ajax-e-php-parte-1/

Parte 2 do tutorial AJAX + PHP

http://felipetonello.com/blog/2007/01/20/engatinhando-com-ajax-e-php-parte-2/

Recursos Extras/Integração com Flash

De Wikibooks

13.14 - Integração com Flash

Abordando aspectos básicos do uso de Flash em sites com PHP. Edição e pequenos ajustes de arquivos Flash, assim como a criação de pequenas animações.

Este tutorial teve como base a versão 8 professional do Flash mas deve ser válida (com pequenas alterações) para outras versões.

O Flash é uma ferramenta que pode tornar um site bem mais atraente e interativo.

Para isso deve ser usado por profissionais de design com talento e com muita cautela, pois também pode tornar o site pesado para carregar.

Usado com bom senso é um grande aliado, pois traz recursos muito atraentes e difíceis de conseguir com outra ferramenta.

Este pequeno guia não é um tutorial do Flash, mas apenas um guia de uso do Flash para edição de filmes (fontes .FLA).

- Caso o arquivo que vá abrir seja de versão anterior, inicie fazendo uma cópia do mesmo para preservar o original, pois normalmente ao ser editado e salvo em versão mais rcente não mais poderá ser aberto na versão mais antiga.

- Abra o arquvo no Flash e tecle Ctrl+Enter para executar no visualizador. Depois feche para voltar ao Flash.

- Ao selecionar um objeto (movie clip, button ou graphic) podemos alterar suas propriedades e seu ActionScript. Para exibir a janela de propriedades tecle Ctrl+F3 e para exibir a do ActionScript para alterar ou adicionar algun código tecle F9.

- Temos duas ferramentas de seleção (seta preta - Selection Tool - tecla de atalho V e seta branca - Subselection Tool - tecla de atalho A). A selection com um duplo clique seleciona todas as camadas ou componentes de um objeto e pode alterar as propriedades de todos de uma vez, já a Subselection não tem esta função, por exemplo, selecione um segmento de reta e ao clica em um dos extremos pode apenas mover um extremo e deixar o outro fixo. Já a selection moverá o segmento de reta por inteiro.

- Observe que por default a Timeline mostra Scene 1. Ao editar um button (com um duplo clique), ao lado de Scene aparece Symbol 2, pois entramos no modo de edição de Símbolo. Logo abaixo 4 fases do botão, Up, Over, Down e Hit. Up é quando o mouse ao clicar no botão inicia o clique, o down é

quando o mouse ao clicar termina o clique e solta o botão. o Over é quando o mouse é movido passando sobre o botão e o Hit é a porção de área clicável do botão. Para mover de uma região desta para outra arraste o mouse para a outra.

- Algumas vezes certos textos e outros objetos não aparecem no Flash (apenas no visualizador) por mais que selecionemos todos os objetos. Nestes casos tecle Ctrl+L para exibir a biblioteca (Library) com todos os objetos do arquivo. Com um clique sobre o nome ou o ícone selecionamos. Com um duplo clique sobre o ícone selecionamos para edição no centro.

- Ao mostrar o editor de símbolos com um botão teclando Enter ele mostra o comportamento e as demais fases.

- Adicionando ação a um botão

- Primeiro convertemos o objeto em Symbol - Button.

- Depois Selecionamos o objeto e teclamos F9

- Então clicamos no sinal de + da barra de ActionScripts

- Selecionamos Global Functions - Movie Clip Control - on e duplo clique em press

Ficará assim:

```
on (press) {

}
```

- Insira uma linha em branci acima do fecha chaves e Novamente clicar no sinal de +

```
on (press) {
}
```

- Global Functions - Browser/Network - getURL e então digite a URL:

Ficará assim:

```
on (press) {
        getURL("http://ribafs.net", "_blank")

}
```

Como também podemos abrir um arquivo local - getURL ("3Produtividade/index.php","_blank");

- Um recurso importante é fazer o Flash gerar a página HTML que abre um arquivo .SWF.

Vá em File - Publish Settings. Na aba Formats, Type marque HTML.

Agora para publicar (gerar o HTML) basta teclar Shift+F12 ou File - Publish.

- Ctrl+3 exibe a área central de edição de cenários para caber na tela.

-- Integração com Flash e Banco de Dados

- Uma Abordagem muito simples seria criar um codigo em php no qual retornaria na tela com o echo ...

por exemplo:

```
echo "&var1=dadosdavariavel 1&";
echo "&var2=dadosdavariavel 2&";
```

e assim por diante para com demais variaveis ... vc poderia fazer uma chamada no banco de dados e utilizar esta estrutura de variaveis para importarmos para o Flash, salve como dados.php.

para passarmos estas variaveis para o Flash vamos utilizar o LoadVars

Abra o Flash crie dois textos dinâmicos e instâncie eles como os repectivos nomes: Edit1doFlash e Edit2doFlash ... clique no primeiro frame e coloque o seguinte código:

```
//define variavel para carregamento dos dados
carrega = new LoadVars();
//função apos carregamento dos dados...
carrega.onLoad = function(ok) {
      //testa se a carga obteve sucesso
      if (ok) {
              //pega os resultados que o php gera
              Edit1doFlash.text = carrega.var1;
              Edit2doFlash.text = carrega.var2;
      }
//aqui chamamos o php juntamento como o load ... para dar inicio onLoad
carrega.load("http://www.meusite.com.br/dados.php");
//podemos usar este outro com parametros na chamada do php
carrega.load("http://www.meusite.com.br/dados.php?par1=chama1&par2=chama2");

stop();
```

este foi um exemplo bem simples da integração, trazendo dados do php para o flash.

Recursos Extras/Bibliotecas de Funções

De Wikibooks

Encontrar subString sabendo apenas o início e o final

```
// Replace a substring of which you only know the beginning and end
function replaceIndefiniteSubstring ($start_needle, $end_needle, $replacement,
$target) {
    $start_pos = strpos($target, $start_needle);
    $end_pos = strpos($target, $end_needle) + strlen($end_needle);
    return substr_replace($target, $replacement, $start_pos, ($end_pos-
$start_pos));
}
```

Site original - http://snippets.dzone.com/posts/show/4211

Recebebdo o Nome do Diretório que contém o Script Atual

```
function getCurrentDirectory() {
    $path = dirname($_SERVER['PHP_SELF']);
    $position = strrpos($path,'/') + 1;
    return substr($path,$position);
}
```

Site original - http://snippets.dzone.com/posts/show/4210

Aqui mais de 3mil pedaços de código - http://snippets.dzone.com/

Recursos Extras/Compilando o PHP

De Wikibooks

Transformar qualquer Script em um arquivo Executável (.exe)

http://phpbrasil.com/articles/article.php/id/1492

Orientação a Objetos/Teoria e exercícios

De Wikibooks

Conteúdo

- 1 Tradicional classe inicial - olaMundo
- 2 Construtor
- 3 Variáveis e Métodos Static, onde podemos usar sem instanciar a classe
- 4 Ou chamando a variável diretamente "Estatica::$varStatic"
- 5 Métodos Final
- 6 Método Construtor e Destrutor
- 7 Constantes da Classe
- 8 Clonando Objetos
- 9 InstanceOf (Testar se classe é instância de outra)
- 10 Classes Abstratas
- 11 Interfaces
- 12 Tratamento de erros lógicos - Exceptions
- 13 Singleton
- 14 Pattern Factory
- 15 Listando os métodos de um Objeto
- 16 Listando as variáveis de uma classe
- 17 Listar variáveis de um objeto
- 18 Testando existência de classe e subclasse
- 19 Devolver nome da classe pai para objeto ou classe
- 20 Checar se método da classe existe
- 21 Programação Orientado a Objeto - Design Partners
- 22 Padrões de Projeto - Singleton

Tradicional classe inicial - olaMundo

```
class OlaMundo {
        function OlaMundo(){
                return "Olá Mundo do PHPOO!";
        }
}

$ola = new OlaMundo();
print $ola->OlaMundo();
```

```
// Classe Pessoa
class Pessoa {
        private $nome;
        function setNome($nome){
                $this->nome = $nome;
        }

        function getNome(){
                return $this->nome;
        }
}

$joao = new Pessoa();
$joao->setNome("João Brito");
$pedro = new Pessoa();
$pedro->setNome("Pedro Ribeiro");

print '<b><br><br>Classe Pessoa:<br></b>';
print $joao->getNome();
print '<br>';
print $pedro->getNome();
```

Construtor

```
class Construtor {
   function __construct() {
       print "No construtor da Classe";
   }
}

print '<b><br><br>Classe Construtor:<br></b>';
$obj = new Construtor();
print $obj->Construtor;


<h2>Controle de acessos</h2>

class Acessos{
        public $variavelPublic = "Variável Pública<br>";
        protected $variavelProtected = "Variável Protegida<br>";
        private $variavelPrivate = "Variável Privada<br>";

        public function getPublic(){
                return $this->variavelPublic;
        }

        public function getProtected(){
                return $this->variavelProtected;
        }

        public function getPrivate(){
                return $this->variavelPrivate;
        }
```

```
        public function getMetodoPrivate(){
                return Acessos::getPrivate();
        }

}

$especificacaoAcesso = new Acessos();
echo $especificacaoAcesso->getPublic();
echo $especificacaoAcesso->getMetodoPrivate();
//echo $especificaAcesso->getPrivate(); // Dará um erro fatal
```

Variáveis e Métodos Static, onde podemos usar sem instanciar a classe

```
class Estatica{
        static $varStatic = "Variável Estática<br>";

        static function getStatic(){
                return Estatica::$varStatic;
        }
}
```

Ou chamando a variável diretamente "Estatica::$varStatic"

```
echo Estatica::getStatic();
```

Métodos Final

```
class ClasseFinal{
        final function getFinal(){
                echo "Método Final";
        }
}

$classeFinal = new ClasseFinal();
$classeFinal->getFinal();
```

Método Construtor e Destrutor

```
class ContrutorDestrutor{

        private $varMethod;
        function __construct(){
                $this->varMethod = "Construtor()";
                echo "Método {$this->varMethod}<br>";
        }
```

```
        function __destruct(){
                $this->varMethod = "Destrutor()";
                echo "Método {$this->varMethod}<br>";
        }
}

$contrutorDestrutor = new ContrutorDestrutor();
unset($contrutorDestrutor);

class ContrutorDestrutorFilho extends ContrutorDestrutor{
        function __construct(){
                parent::__construct();
                echo "Método Filho Construtor<br>";
        }

        function __destruct(){
                parent::__destruct();
                echo "Método Filho Destrutor<br>";
        }
}

echo "<br>";
$contrutorDestrutorFilho = new ContrutorDestrutorFilho();
```

Constantes da Classe

```
class Constante{
        const constante = "Minha Constante";
}
echo Constante::constante;
```

Clonando Objetos

```
class ClasseClonando{
    public $varClone;
    function __construct(){
        $this->varClone = "<br>Php5<br>";
    }

    function __clone(){
        $this->varClone = "Php5 Clone<br>";
    }
}

$classeClonando = new ClasseClonando();
$cloneClasseClonando = clone $classeClonando;
echo $classeClonando->varClone . "<br>" . $cloneClasseClonando->varClone;
```

InstanceOf (Testar se classe é instância de outra)

```
class TesteInstanceOf
{
//....
}

class ClasseInstanceOf{
    function __construct($obj){
        if ($obj instanceof TesteInstanceOf) {
            echo "Objeto da classe(TesteInstanceOf)<br>";
        } else {
            echo "Não é um objeto da classe(TesteInstanceOf)<br>";
        }
    }
}

$testeInstanceOf = new TesteInstanceOf();
$classeInstanceOf = new ClasseInstanceOf($testeInstanceOf);
```

Classes Abstratas

```
abstract class Abstrata{
    public abstract function setNome($nome);
    public abstract function getNome();
}

class ClasseAbstrata extends Abstrata{
    private $nome;
    public function setNome($newNome){
        $this->nome = $newNome;
    }

    public function getNome(){
        return $this->nome;
    }
}

$classeAbstrata = new ClasseAbstrata();
$classeAbstrata->setNome("Php5");
echo $classeAbstrata->getNome()."<br>";
```

Interfaces

```
interface IPessoa{
    public function setNome($nome);
    public function getNome();
}
```

```
interface IPessoaFisica{
    public function setCpf($cpf);
    public function getCpf();
}

interface IPessoaJuridica{
    public function setCnpj($cnpj);
    public function getCnpj();
}

class ClassePessoa implements IPessoa, IPessoaFisica, IPessoaJuridica{
    function __construct($nome, $cpf, $cnpj){
        ClassePessoa::setNome($nome);
        ClassePessoa::setCpf($cpf);
        ClassePessoa::setCnpj($cnpj);
    }

    /* Métodos Set */
    public function setNome($nome){
        $this->nome = $nome;
    }
    public function setCpf($cpf){
        $this->cpf = $cpf;
    }
    public function setCnpj($cnpj){
        $this->cnpj = $cnpj;
    }
    /* Métodos Get */
    public function getNome(){
        return $this->nome;
    }
    public function getCpf(){
        return $this->cpf;
    }
    public function getCnpj(){
        return $this->cnpj;
    }
    function __destruct(){
        echo
ClassePessoa::getNome()."<br>".ClassePessoa::getCpf()."<br>".ClassePessoa::getCnp
j();
    }
}

$classePessoa = new ClassePessoa("Rodrigo", "324.541.588-98", "6545.2101/0001");
```

Tratamento de erros lógicos - Exceptions

```
class BusinessException extends Exception{
    function __construct($msg){
        // Vai para a função construtora do Exception.
        parent::__construct($msg);
```

```
    }
}

class Excecao{
    function __construct($nome){
        try {
            if ($nome == "") {
                throw new BusinessException("Nome não pode ser em branco");
            } elseif(strlen($nome) < 5) {
                throw new BusinessException("Nome precisa ter no mínimo 5
letras");
            } elseif(strtolower($nome) == "corinthians") {
                throw new BusinessException("Corinthians campeão");
            } else {
                throw new BusinessException("Paramêtro inválido");
            }
        } catch (BusinessException $businessException) {
            echo $businessException->getMessage();
        }
    }
}

$excecao = new Excecao("Corinthians");
```

Singleton

```
class Singleton {
    private static $instance = null;

    public static function getInstance(){
        if (Singleton::$instance == null) {
            Singleton::$instance = new Singleton();
        }
        return Singleton::$instance;
    }
}

$objA = Singleton::getInstance();
$objB = Singleton::getInstance();
if ($objA == $objB) {
    echo "<br>Instância única";
} else {
    echo "<br>Instâncias diferentes";
}
```

Pattern Factory

```
abstract class AbstractFactory
{
    private $nome;
```

```
    private $rendaMensal;

    function __construct($nome, $rendaMensal){
        $this->setNome($nome);
        $this->setRendaMensal($rendaMensal);
    }

    public function setNome($newNome){
        $this->nome = $newNome;
    }

    public function setRendaMensal($newRendaMensal){
        $this->rendaMensal = $newRendaMensal;
    }

    public function getNome(){
        return $this->nome;
    }

    public function getRendaMensal(){
        return $this->rendaMensal;
    }

    public abstract function analisarCredito(); // Boolean
    public abstract function getCategoria(); // String
}

class ClientePadrao extends AbstractFactory{
    function __construct($nome, $rendaMensal){
        parent::__construct($nome, $rendaMensal);
    }

    // Foi declarada no AbstractFactory
    public function analisarCredito(){
        return true;
    }

    // Foi declarada no AbstractFactory
    public function getCategoria(){
        return "Cliente Padrão";
    }
}

class ClienteRisco extends AbstractFactory{
    function __construct($nome, $rendaMensal){
        parent::__construct($nome, $rendaMensal);
    }

    // Foi declarada no AbstractFactory
    public function analisarCredito(){
        return false;
    }

    // Foi declarada no AbstractFactory
    public function getCategoria(){
        return "Cliente Risco";
    }
```

```
}

class ClienteSeguro extends AbstractFactory{
    function __construct($nome, $rendaMensal){
        parent::__construct($nome, $rendaMensal);
    }

    // Foi declarada no AbstractFactory
    public function analisarCredito(){
        return true;
    }

    // Foi declarada no AbstractFactory
    public function getCategoria(){
        return "Cliente com alta credibilidade";
    }
}

class SingletonFactory{
    private static $rendaMedia = 500;
    private static $rendaBaixa = 240;
    private static $instance = null;

    public static function getCliente($nome, $rendaMensal){
        if ($rendaMensal <= SingletonFactory::$rendaBaixa) {
            SingletonFactory::$instance = new ClienteRisco($nome, $rendaMensal);
        } elseif ($rendaMensal > SingletonFactory::$rendaBaixa and
                                $rendaMensal <= SingletonFactory::$rendaMedia) {
            SingletonFactory::$instance = new ClientePadrao($nome, $rendaMensal);
        } else {
            SingletonFactory::$instance = new ClienteSeguro($nome, $rendaMensal);
        }

        $clienteAprovacao = "reprovado";
        if (SingletonFactory::$instance->analisarCredito()) {
            $clienteAprovacao = "aprovado";
        }

        echo "<br>Cliente = ".SingletonFactory::$instance->getNome()."<br>";
        echo "Categoria = ".SingletonFactory::$instance->getCategoria()."<br>";
        echo "Crédito = ".$clienteAprovacao;
        echo "<hr>";
    }
}

SingletonFactory::getCliente("Rodrigo", 1977);
SingletonFactory::getCliente("Corinthians", 350);
SingletonFactory::getCliente("John", 220);
```

Listando os métodos de um Objeto

```
class OlaMundo2 {
   // constructor
```

```
   function OlaMundo2(){
       return(true);
   }
   // method 1
   function funcao1(){
       return(true);
   }

   // method 2
   function funcao2(){
       return(true);
   }
}

$meus_objetos = new OlaMundo2();
$metodos_classe = get_class_methods(get_class($meus_objetos));

foreach ($metodos_classe as $nome_metodo) {
   echo "$nome_metodo<br>";
}
```

Listando as variáveis de uma classe

```
class Variaveis {

   var $variavel1; // esta não tem valor default...
   var $variavel2 = "xyz";
   var $variavel3 = 100;

   // construtor
   function Variaveis() {
       // mudar algumas propriedades
         $this->var1 = "foo";
         $this->var2 = "bar";
         return true;
   }

}

$minha_classe = new Variaveis();
$variaveis = get_class_vars(get_class($minha_classe));

foreach ($variaveis as $nome => $value) {
   echo "$nome = $value<br>";
}
```

Listar variáveis de um objeto

```
class Point2D {
   var $x, $y;
```

```
   var $label;

   function Point2D($x, $y){
       $this->x = $x;
       $this->y = $y;
   }

   function setLabel($label){
       $this->label = $label;
   }

   function getPoint(){
       return array("x" => $this->x,
                    "y" => $this->y,
                    "label" => $this->label);
   }
}

// "$label" is declared but not defined
$p1 = new Point2D(1.233, 3.445);
print_r(get_object_vars($p1));

$p1->setLabel("point #1");
print_r(get_object_vars($p1));
```

Testando existência de classe e subclasse

```
class Foo { var $myVar; }

class Foo_Bar extends Foo { var $myVar2;}

echo class_exists('Foo')."<br>"; //true
echo class_exists('foo')."<br>"; //true
echo class_exists('Foo_Bar')."<br>"; // true
echo get_parent_class('Foo_Bar')."<br>"; // foo (NOTE: NOT Foo!)
```

Devolver nome da classe pai para objeto ou classe

```
class dad {
   function dad(){
   // implemente alguma lógica
   }
}

class child extends dad {
   function child(){
       echo "Eu sou a classe <b>" , get_parent_class($this) , "'s </b>filho<br>";
   }
```

```
}

class child2 extends dad {
   function child2(){
       echo "Eu também sou a classe <b>" , get_parent_class('child2') , "'s
</b>filho<br>";
   }
}

$foo = new child();
$bar = new child2();
```

Checar se método da classe existe

```
class Foo1 {
  public function bar() {
   echo "Eu sou private Foo1::bar()<br>";
  }
}

class Foo2 {
  private function bar() {
   echo "Eu sou public Foo2::bar()<br>";
  }
}

$f1=new Foo1;
$f2=new Foo2;

if(is_callable(array($f1,"bar"))) {
   echo "Foo1::bar() é acessível<br>";
} else {
   echo "Foo1::bar() não é acessível<br>";
}
if(is_callable(array($f2,"bar"))) {
   echo "Foo2::bar() é acessível<br>";
} else {
   echo "Foo2::bar() não é acessível<br>";
}
if(in_array("bar",get_class_methods($f1))) {
   echo "Foo1::bar() é acessível<br>";
} else {
   echo "Foo1::bar() não é acessível<br>";
}
if(in_array("bar",get_class_methods($f2))) {
   echo "Foo2::bar() é acessível<br>";
} else {
   echo "Foo2::bar() não é acessível<br>";
}
```

Programação Orientado a Objeto - Design Partners

http://www.revistaphp.com.br/print.php?id=140

Padrões de Projeto - Singleton

http://www.revistaphp.com.br/print.php?id=142

Orientação a Objetos/Aplicativos de exemplo

De Wikibooks

Exemplos de Aplicativos em PHP Orientados a Objetos

Iniciante PHPOO - http://www.ribafs.net/down/appsexemplo/my/iniciantePHPOO.zip

Orientação a Objetos/Ferramentas

De Wikibooks

Ferramentas para Trabalhar com PHP Orientado a Objetos

O Umbrello é uma boa ferramenta para trabalhar com UML com suporte ao PHP - http://uml.sourceforge.net/index.php

Bons rcursos de UML e também gera o código PHP. Até o momento somente com versão for Linux.

O ArgoUML é uma ferramenta que tem suporte a Linux e a Windows que trabalha com UML e tem suporte ao PHP - http://argouml.tigris.org/

Aplicativos de Exemplo com/MySQL

De Wikibooks

15.1 - Aplicativos de Exemplo com MySQL

Vários Aplicativos de Exemplo em PHP com MySQL

Aplicativos de Exemplo com/PostgreSQL

De Wikibooks

15.2 - Aplicativos de Exemplo com PostgreSQL

Alguns Aplicativos de Exemplo em PHP com PostgreSQL

Aplicativos de Exemplo com/SQLite

De Wikibooks

15.3 - Aplicativos de Exemplo com SQLite

Um Aplicativo de Exemplo em PHP com SQLite

Apêndices/Análise e Projeto

De Wikibooks

Conteúdo

- 1 Análise de Sistemas na Wikipedia
- 2 Projeto na Wikipedia
- 3 Análise e Projeto
- 4 Algoritmo na Wikipedia
- 5 Lógica e computadores
- 6 Projeto de Bancos de Dados
  - 6.1 Introdução Teórica
  - 6.2 Dicas sobre Campos
  - 6.3 Selecionando o Campo para a Chave Primária
  - 6.4 Chave Primária
  - 6.5 Tipos de Relacionamentos
  - 6.6 Relacionamento Um para Um
  - 6.7 Relacionamento Vários para Vários
  - 6.8 Integridade Referencial
  - 6.9 Normalização de Tabelas
    - 6.9.1 1a Forma Normal
    - 6.9.2 2ª Forma Normal
    - 6.9.3 3ª Forma Normal
  - 6.10 Projeto
  - 6.11 Etapas na Estruturação e Projeto de um Banco de Dados
  - 6.12 Projeto Exemplo
  - 6.13 Hardware indicado para abrigar o servidor de banco de dados
  - 6.14 Software indicado como o servidor dos bancos
  - 6.15 Adotar um modelo e justificá-lo
  - 6.16 Modelo Lógico
  - 6.17 Referências

Análise de Sistemas na Wikipedia

A atividade de análise tem como finalidade realizar estudos de processos afim de encontrar o melhor e mais racional caminho para que a informação possa ser processada. O analista de sistema estuda os diversos sistemas existentes entre hardwares (equipamento), softwares (programas) e o usuário final, seus comportamentos e aplicações, desenvolvendo a partir de então soluções que serão padronizadas e transcritas da forma que o computador possa executar. Gerando o que se chama de softwares

(programas), que são executados em hardwares (equipamentos) operados por usuários, individuos, preparados e treinados em procedimentos operacionais padronizados, dotados de conhecimentos do software e hardware para seu trabalho. A partir de então a análise de sistemas é uma profissão, cujas responsabilidades concentram-se na programação e na administração de sistemas computacionais. Cabe a este profissional parte da organização, implantação e manutenção de aplicativos e redes de computadores. Como é uma enfâse, o foco e o núcleo de trabalho está voltado para Administração, levando em conta a area tecnológica em que irá auxiliar.

Fonte: http://pt.wikipedia.org

Projeto na Wikipedia

Segundo o Project Management Institute, projeto é um esforço temporário empreendido para criar um produto ou serviço único. Desta forma, um projeto tem início e fim definidos e resulta em um produto ou serviço de alguma forma diferente de todos os outros anteriormente produzidos.

Seu resultado pode ser:

Um produto ou objeto produzido, quantificável e que pode ser um item final ou um item componente

Uma capacidade de realizar um serviço, como funções de negócios que dão suporte à produção ou à distribuição

Um resultado, como resultados finais ou documentos. Por exemplo, um projeto de pesquisa desenvolve um conhecimento que pode ser usado para determinar se uma tendência está presente ou não ou se um novo processo irá beneficiar a sociedade.

A singularidade é uma característica importante das entregas do projeto. Por exemplo, muitos milhares de prédios de escritórios foram construídos, mas cada prédio em particular é único—tem proprietário diferente, projeto diferente, local diferente, construtora diferente, etc. A presença de elementos repetitivos não muda a singularidade fundamental do trabalho do projeto.

Antes do projeto, é comum ainda o trabalhador fazer a preparação de um Anteprojeto, que é o estudo preparatório do projeto.

Nos contextos de software, projeto é usado com o sentido do ato de projetar, de conceber antecipadamente. Neste caso, costuma-se também empregar a palavra design.

Fonte: http://pt.wikipedia.org

Análise e Projeto

1.**Introdução**

2.**Objetivos e Justificativa**

Devem ser definidos o problema existente, o propósito do site (objetivos), por que ele é importante ou necessário (justificativa), para quem ele será útil (usuários), qual o escopo do site (intranet, internet ou extranet) e outras informações que você julgar convenientes e necessárias.

3.**Descrição dos requisitos**

Requisitos de conteúdo - quais informações o site oferece

Requisitos funcionais - quais serviços são oferecidos aos usuários

Requisitos de desenvolvimento - quais recursos materiais e humanos foram utilizados

Requisitos operacionais - quais equipamentos de hardware e quais sistemas de software serão necessário para o web site funcionar. Como será feita a conexão com a internet.

4.**Design - descrição gráfica acompanhada de um texto descrevendo**:

Estrutura organizacional do site - O design da organização envolve a estruturação das diversas páginas que formarão o site, da interligação entre elas.

Aspectos estéticos: Esquema de Layout do banco de dados e dos scripts, estilos de fontes, uso das cores, etc., justificando a sua escolha

5.**Anexos: descrição dos arquivos de HTML, CSS, JavaScript, PHP, etc, que forem utilizados.**

Adaptado do original em - http://www.dimap.ufrn.br/~jair/piws/exercicios2004.html

Faça as perguntas:

- O que iremos desenvolver?

- Como desenvolver?

- Já existe aplicativo similar?

- Se existe, podemos reutilizar algo do mesmo?

Algoritmo e Lógica

Algoritmo na Wikipedia

Um algoritmo é uma sequência finita e não ambígua de instruções para solucionar um problema. Mais especificamente, em matemática, constitui o conjunto de processos (e símbolos que os representam) para efectuar um cálculo. Algoritmos podem ser implementados por programas de computadores. A palavra algoritmo tem origem no sobrenome, Al-Khwarizmi, do matemático persa do século IX, Mohamed ben Musa, cujas obras foram traduzidas no ocidente cristão no século XII, tendo uma delas recebido o nome "Algorithmi de numero indorum", sobre os algoritmos usando o sistema de numeração decimal (indiano). Outros autores, contudo, defendem a origem da palavra em Al-goreten (raiz - conceito que se pode aplicar aos cálculos).

O conceito de algoritmo é freqüentemente ilustrado pelo exemplo de uma receita, embora muitos algoritmos sejam mais complexos. Eles podem repetir passos (fazer iterações) ou necessitar de decisões (tais como comparações ou lógica) até que a tarefa seja completada. Um algoritmo corretamente executado não irá resolver um problema se o algoritmo estiver incorreto ou não for apropriado ao problema.

Um algoritmo não representa, necessariamente, um programa de computador, e sim os passos necessários para realizar uma tarefa. Sua implementação pode ser feita por um computador, por outro tipo de autômato ou mesmo por um ser humano.

Diferentes algoritmos podem realizar a mesma tarefa usando um conjunto diferenciado de instruções em mais ou menos tempo, espaço ou esforço do que outros. Por exemplo, um algoritmo para se vestir pode especificar que você vista primeiro as meias e os sapatos antes de vestir a calça enquanto outro algoritmo especifica que você deve primeiro vestir a calça e depois as meias e os sapatos. Fica claro que o primeiro algoritmo é mais difícil de executar que o segundo.

Fonte: http://pt.wikipedia.org

Lógica e computadores

A Lógica é extensivamente usada em áreas como Inteligência Artificial, e Ciência da computação.

Nas décadas de 50 e 60, pesquisadores previram que quando o conhecimento humano pudesse ser expresso usando lógica com notação matemática, supunham que seria possível criar uma máquina com a capacidade de pensar, ou seja, inteligência artificial. Isto se mostrou mais difícil que o esperado em função da complexidade do raciocínio humano. programação lógica é uma tentativa de fazer

computadores usarem raciocínio lógico e a linguagem de programação Prolog é comumente utilizada para isto.

Na lógica simbólica e lógica matemática, demonstrações feitas por humanos podem ser auxiliadas por computador. Usando demonstração automática de teoremas os computadores podem achar e checar demonstrações, assim como trabalhar com demonstrações muito extensas.
Na ciência da computação, a álgebra booleana é a base do projeto de hardware.

Projeto de Bancos de Dados

O que me levou a compilar este tutorial foi a grande carência de material nesta área, especialmente em português. Se aplicam a qualquer SGBD.

Introdução Teórica

O projeto do banco de dados e também os testes são muito importantes para a eficiência e consistência das informações e do aplicativo. É muito importante gastar algum tempo nesta etapa, pois depois de algum tempo de implantado fica muito trabalhoso alterar estruturas de bancos e aplicativos.

Projetos de banco de dados ineficazes geram consultas que retornam dados inesperados, relatórios que retornam valores sem sentido, etc. Um banco de dados bem projetado fornece um acesso conveniente às informações desejadas e resultados mais rápidos e precisos.

Lembrando:

- SGBDR (Sistema Gerenciador de Bancos de Dados Relacional) – PostgreSQL

- Software de administração de banco de dados para o PostgreSQL – PGAdmin

Informações de bancos de dados relacionais são armazenadas em tabelas ou entidades no Modelo ER (Entidade-Relacionamento).

Dicas sobre Campos

Não armazenar resultado de cálculos ou dados derivados de outros

Armazenar todas as informações (campos) separadamente. Cuidado com campos que contém duas ou mais informações.

Selecionando o Campo para a Chave Primária

A chave primária é o campo ou campos que identificam de forma exclusiva cada registro.

Não são permitidos valores nulos nem duplicados na chave nem duplicados

Caso a tabela não tenha um campo que a identifique, pode-se usar um campo que numere os registros seqüencialmente.

Dica de Desempenho: O tamanho da chave primária afeta o desempenho das operações, portanto usar o menor tamanho que possa acomodar os dados do campo.

Exemplo

```
Tabela – Clientes
Campo – Nome (atributo)
Chave Primária (Primary Key) – CPF
```

Todos os campos correspondentes a um único CPF juntamente com seus valores formam um Registro ou Linha (Row)

A correta determinação das tabelas, bem como dos campos é algo primordial no sucesso do projeto do banco de dados.

Chave Primária

Obriga que todos os registros terão o campo correspondente à chave primária exclusivo (único - unique). Num cadastro de clientes, todos os clientes cadastrados terão um campo CPF exclusivo. Caso se tente inserir dois clientes com o mesmo CPF o banco não permitirá e emitirá uma mensagem de erro acusando tentativa de violação da chave primária.

Exemplos de Campos indicados para chave primária:

```
CPF
CNPJ
Matrícula de aluno
Matrícula de funcionário
```

Uma chave primária pode ser formada por mais de um campo, quando um único campo não é capaz de caracterizar a tabela.

Cada tabela somente pode conter uma única chave primária.

Relacionamentos – Um banco de dados é formado por várias tabelas. Idealmente essas tabelas devem ser relacionadas entre si para facilitar a troca de informações e garantir a integridade. Para relacionar tabelas usamos chaves existentes nas mesmas.

Tipos de Relacionamentos

```
Um para um
Um para vários
```

```
Vários para vários
```

Relacionamento Um para Um

Aquele onde os campos que fazem o relacionamento são chaves primárias. Cada registro de uma tabela se relaciona com apenas um registro da outra tabela.

Este relacionamento não é muito comum.

Exemplo: CorrentistaBanco - Conjuge

Relacionamento Um para Vários

Aquele onde uma tabela tem um campo chave primária (PK) que se relaciona com outra tabela através de um campo chave estrangeira (FK) . É o tipo de relacionamento mais utilizado.

Exemplos:

```
Clientes – Pedidos
Produtos – Itens
Categorias – Itens
Fornecedores – Produtos
NotaFiscal - Produtos
```

Veja que cada registro da esquerda se relaciona com vários registros da direita.

Importante:

O tipo de dados dos campos do relacionamento deve ser igual, assim como o tamanho dos campos e formatos

Chave primária – Chave estrangeira (um – vários)

Relacionamento Vários para Vários

Este tipo de relacionamento não dá para ser implementado no modelo relacional, portanto sempre que nos deparamos com um deles devemos dividir em dois relacionamentos um para vários (criando uma terceira tabela, que armazenará o lado vários dos relacionamentos).

Exemplo:

Pedidos – Produtos (1 --- N e N --- 1)

Cada pedido pode conter vários produtos, assim como cada produto pode estar em vários pedidos. A saída é criar uma tabela que contenha os itens do pedido.

Pedidos – Pedidos_Itens – Produtos

Pedidos 1 - N Pedidos_Itens N - 1 Produtos

Integridade Referencial

Ela garante a integridade dos dados nas tabelas relacionadas. Um bom exemplo é quando o banco impede que se cadastre um pedido para um cliente inexistente, ou impede que se remova um cliente que tem pedidos em seu nome.

Também se pode criar o banco de forma que quando atualizamos ou excluímos o CPF de um cliente ele seja atualizado ou excluído em todos os seus pedidos.

Normalização de Tabelas

Normalizar tabelas e bancos tem o objetivo de tornar o banco mais eficiente, impondo integridade aos dados.

Uma regra muito importante ao criar tabelas é atentar para que cada tabela contenha informações sobre um único assunto, de um único tipo.

1a Forma Normal

Os campos não devem conter grupos de campos que se repetem nos registros.

Exemplo:

Alunos: matricula, nome, data_nasc, serie, pai, mae

Se a escola tem vários filhos de um mesmo casal haverá repetição do nome dos pais. Estão para atender à primeira regra, criamos outra tabela com os nomes dos pais e a matrícula do aluno.

2ª Forma Normal

- Quando a chave primária é composta por mais de um campo. - Devemos observar se todos os campos que não fazem parte da chave dependem de todos os campos que fazem parte da chave.

Caso algum campo dependa somente de parte da chave, então devemos colocar este campo em outra tabela.

Exemplo:

TabelaAlunos

Chave (matricula, codigo_curso)

avaliacao

descricao_curso

Neste caso o campo descricao_curso depende apenas do codigo_curso, ou seja, tendo o código do curso conseguimos sua descrição. Então esta tabela não está na 2ª Forma Normal.

Solução:

Dividir a tabela em duas (alunos e cursos):

TabelaAlunos

Chave (matricula, codigo_curso)

avaliacao

TabelaCursos

codigo_curso descricao_curso

3ª Forma Normal

Quando um campo não é dependente diretamente da chave primária ou de parte dela, mas de outro campo da tabela que não pertence à chave primária. Quando isso ocorre esta tabela não está na terceira forma normal e a solução é dividir a tabela.

Lembrando: Engenharia Reversa (parte de um banco existente ou de um script sql e gera o modelo).

Projeto

Fases do Projeto do Banco de Dados

Modelo Conceitual

Modelo Lógico

Modelo Físico (faz parte apenas da implementação)
Observação.: Trataremos apenas de novos projetos.

Modelo Conceitual – Define apenas quais os dados que aparecerão no banco de dados, sem se importar com a implementação do banco. Para essa fase o que mais se utiliza é o DER (Diagrama Entidade-Relacionamento).

Modelo Lógico – Define quais as tabelas e os campos que formarão as tabelas, como também os campos-chave, mas ainda não se preocupa com detalhes como o tipo de dados dos campos, tamanho, etc.

Etapas na Estruturação e Projeto de um Banco de Dados

Problemas a serem solucionados com o banco de dados

Determinar as tabelas necessárias (cada uma com um único assunto exclusivo)

Determinar os campos de cada tabela

Criar um DER

Verificar a estimativa do crescimento do banco e preparar-se para isso

Investigar como são armazenadas as informações atualmente e recolher a maior quantidade de informações para o projeto

Adotar um modelo e justificá-lo

(Os itens acima fazem parte do Modelo Conceitual e abaixo do Lógico)

Determinar a chave primária de cada tabela. Pode haver tabela sem chave primária.

Determinar os relacionamentos e seus tipos

Obs.: Somente quando da implementação (modelo físico) serão tratados os detalhes internos de armazenamento. O modelo físico é a tradução do modelo lógico para a linguagem do SGBDR adotado.

Projeto Exemplo

Vamos elaborar um exemplo de projeto de banco de dados que será um controle de funcionários.

Controle de Funcionários

Modelo Conceitual

Problemas a serem solucionados com o banco de dados

Atualmente os funcionários são cadastrados em fichas de papel guardadas em pastas. Isso acarreta dificuldade no resgate de informações e fragilidade das mesmas.

O objetivo do banco de dados será armazenar as informações sobre os funcionários, possibilitando consultas ágeis que retornem as informações de maneira rápida e prática.

Determinar as tabelas necessárias (cada uma com um único assunto exclusivo)

Utilizaremos apenas a tabela funcionarios, contendo todos os dados dos funcionários.

Determinar os campos da tabela

Os campos serão: cpf, nome, email, endereco, cep, cidade, estado, fone, celular, conjuge, filhos.

Criar um DER (uma ótima ferramenta é o plugin para Eclipse Azzurri Clay).

```
funcionarios
cpf
nome
...
```

Hardware indicado para abrigar o servidor de banco de dados

O hardware mais adequado deve ter um disco rígido rápido, boa quantidade de memória RAM, uma boa placa-mãe e um bom processador. As especificações dependem de cada caso.

Software indicado como o servidor dos bancos

Como SGBDR a indicação vai para o PostgreSQL,por ser robusto, estável, bom desempenho, boa documentação, grande comunidade através da Internet e Licença free e open-source para todos os usos.

Como sistema operacional para o servidor a indicação vai para o Linux Debian.

Como sistema operacional para os clientes a indicação vai para o Linux Ubuntu.

Verificar a estimativa do crescimento do banco e preparar-se para isso

A empresa é pequena mas tem perspectiva de crescimento a médio e longo prazo...

Adotar um modelo e justificá-lo

O modelo adotado foi o MER (Modelo Entidade-Relacionamento), por atender às necessidades do cliente e ser o modelo padrão de mercado atualmente.

Modelo Lógico

Teremos apenas uma tabela, funcionarios, cuja chave primária é o campo CPF.

Aplicar as 3 formas normais para testar a coerência das tabelas.

Primeira - Os campos não devem conter grupos de campos que se repetem nos registros.

Cada funcionário terá todos os campos diferentes, exceto alguns campos isolados, como cep, cidade, mas não temos grupos de campos que se repetem.

Sugestão: Ter cidade, cep e estado em tabelas diferentes e relacionadas, deixando em funcionários apenas combos onde seriam selecionados.

Mas tanto o campo conjuge quanto o filhos devem ir para tabelas separadas, já que cônjuge pode armazenar várias informações e filhos também.

Então teremos as tabelas:

funcionarios, conjuges e filhos, de acordo com o DER abaixo.

Utilizando o DBDesigner, o DER acima gerou o script abaixo:

```
CREATE TABLE funcionarios (
  cpf VARCHAR NOT NULL,
  nome VARCHAR NULL,
  endereco VARCHAR NULL,
  cep VARCHAR NULL,
  cidade VARCHAR NULL,
  estado VARCHAR NULL,
  cod_conjuge VARCHAR NULL,
  cod_filhos VARCHAR NULL,
  PRIMARY KEY(cpf)
);

CREATE TABLE func_filhos (
  codigo INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  funcionarios_cpf VARCHAR NOT NULL,
  nome VARCHAR NULL,
  cpf_funcionario VARCHAR NULL,
  data_nascimento DATE NULL,
  PRIMARY KEY(codigo),
  FOREIGN KEY(funcionarios_cpf)
    REFERENCES funcionarios(cpf)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION
);

CREATE TABLE func_conjuge (
  cpf_funcionario VARCHAR NOT NULL AUTO_INCREMENT,
  funcionarios_cpf VARCHAR NOT NULL,
  nome INTEGER NULL,
  data_nascimento DATE NULL,
  PRIMARY KEY(cpf_funcionario),
  FOREIGN KEY(funcionarios_cpf)
    REFERENCES funcionarios(cpf)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION
);

funcionarios 1 – 1 conjuge
funcionarios 1 – N filhos
```

Tutorial sobre uso do DBDesigner com PostgreSQL: http://wwwpostgresql.org.br

Segunda Forma Normal – Quando a chave primária é composta por mais de um campo.

Não é o caso, pois CPF é suficiente para representar cada funcionário.

Terceira Forma Normal- Quando um campo não é dependente diretamente da chave primária ou de parte dela, mas de outro campo da tabela que não pertence à chave primária. Não é o caso.

Referências

- O Modelo Relacional de Dados (em cinco artigos, de Júlio Battisti)
http://www.imasters.com.br/artigo.php?cn=2419&cc=149

- Conceitos Fundamentais de Banco de Dados (de Ricardo Rezende)
http://www.sqlmagazine.com.br/Colunistas/RicardoRezende/02_ConceitosBD.asp

Apêndices/Design

De Wikibooks

Web Design, o artista da Web

Design na Wikipedia

Entende-se o design (em alguns casos projeto ou projecto) como o esforço criativo relacionado à configuração, concepção, elaboração e definição de algo, como um objeto, uma imagem, entre outros, em geral voltados a uma determinada função. De uma forma ampla o termo design, porém, refere-se à concepção de uma solução prévia para um problema. Mas em uma acepção mais específica, design se refere à profissão da pessoa que projeta. Como tal, tem diversas especializações, de acordo com o tipo de coisa a projetar. O profissional que trabalha na área de design é chamado, portanto, de designer, visto a palavra pertencer à língua inglesa e normalmente não se traduz (ver o problema etimológico).

Design é, portanto, um esforço criativo através do qual se projetam todo tipo de coisas, incluindo utensílios, vestimentas, peças gráficas, livros, máquinas, ambientes e (recentemente) também interfaces de programas.

As especializações mais comuns são o design de produto, design gráfico e o design de moda. O design está intimamente ligado às artes aplicadas, à arquitetura, e à engenharia, mas a concorrência profissional muitas vezes leva à animosidade entre essas áreas[Carece de fontes?].

O termo deriva, originalmente, de designare, palavra em latim, sendo mais tarde adaptado para o inglês design. Houve uma série de tentativas de tradução do termo, mas os possíveis nomes como projética industrial acabaram em desuso.

Usabilidade

Uma boa coleção de links sobre usabilidade - http://www.usableweb.com/

Apêndices/Hospedagem e Domínio

De Wikibooks

16.3 - Hospedagem e Domínio

Basicamente para publicar um site na Internet precisamos desses dois serviços: uma hospedagem que abrigará o material do site e um serviço que administra o domínio (o nome, o endereço do site).

Sua própria empresa de hospedagem

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=6407

Apêndices/Segurança

De Wikibooks

Algumas Dicas sobre Segurança no Desenvolvimento de Aplicativos

Conteúdo

- 1 - Senhas
- 2 - Informações
- 3 - Código
- 4 Não Confiar em Variáveis Globais
- 5 Evite Falsos Uploads
- 6 Idéias Adicionais
- 7 Não armazene números de cartões de crédito
- 8 Não insira Conteúdo Sigiloso no raiz do Aplicativo
- 9 Muita atenção aos Serviços de Hospedagem
- 10 Evite ao máximo as funções
- 11 Além do Código (Um projeto de segurança forte)
- 12 Nunca inclua, requeira ou abra um arquivo cujo nome seja baseado em entrada do usuário, sem antes checar
- 13 Nunca execute consultas a bancos sem usar funções de escape
- 14 Restringe o diretório admin para utilizar senhas
- 15 Prevenindo Injeções SQL
- 16 Escrevendo Código Robusto em PHP
- 17 PHP Captcha Security Images
- 18 Referências

- Senhas

Ataques contra senhas normalmente usam o método da força bruta (brute force), portanto evite senhas simples e que constem de dicionários.

Evite senhas do tipo:

- Somente numéticas ou somente caracteres

- Seu nome, nomes de filhos, número de identidade, data de nacimento, CEP ou outro constante de dicionários

- Senhas com poucos caracteres

- Não use a mesma senha para todos os servidores

Senha recomendada:

- Com 15 ou mais caracteres

- Misture letras, algarismos e caracteres especiais, maiúsculas e minúsculas

- Muito importante: use uma senha que você possa se lembrar

- Informações

Cuidado com o Acesso em Lan houses

Segundo a PNAD 2005 (IBGE) 18,6% das residências brasileiras têm computadores e destes apenas 13,7% com acesso à Internet. Muita gente então acessa a Internet em Lan hauses.

Os especialistas em segurança recomendam não efetuar transações bancárias ou qualquer outro tipo onde sejam requeridos dados pessoais, como CPF, cartão de crédito, RG, etc.

Existem diversas formas de se monitorar os passos de alguém na Internet (programas mal intencionados) e até mesmo as câmaras existentes nas lan houses, se em mãos erradas podem representar perigo para os usuários.

Cuidado com Notebooks

Os notebooks também estão na mira dos criminosos da Internet. Evite dados sigilosos nos mesmos ou use uma boa criptografia para proteger seus dados. Use senhas fortes para o acesso ao notebook e faça backup regularmente de suas informações.

Instale e tenha sempre atualizados: bom antivirus, antispy e firewall.

Esteja atento para as atualizações do sistema operacional e de todos os softwares importantes.

- Código

- Ofereça uma quantidade fixa (3) de tentativas de login. Após as 3 o login deve ser desabilitado, por segurança.

Segurança em Aplicativos PHP

Este texto é formado pela tradução de partes de alguns textos em inglês (vide Referências) e de alguns exemplos de código e recomendações que adicionei.

Em primeiro lugar devemos atentar para uma boa análise e projeto da aplicação. Da qualidade destes depende a qualidade da aplicação. Então devemos planejar o banco de dados cuidadosamente: tabelas, relacionamentos, campos, tipos de dados, etc. Mais importantes ainda em termos de segurança é a criação de usuários que tenham acesso somente ao aplicativo e com privilégios somente para suas operações com nomes e senhas seguras, como também usuários do banco com respectivas permissões e permissões do sistema operacional.

As aplicações Web contam com formas populares de acesso global a dados, a serviços e a produtos. Enquanto este acesso global é uma das grandes vantagens da Web, qualquer regra de segurança nesses aplicativos também é globalmente exposta e freqüentemente explorada. É muito fácil escrever aplicações que contém regras de segurança. Vide aplicações famosas como phpMyAdmin, PHPShop e FreeTrade.

Algumas Recomendações a favor da Segurança

- Evitar uso de Variáveis quando acessando Arquivos

Cuidado com as funções:

- readfile

- fopen

- file

- include

- require

Caso decida assim mesmo utilizar, tome precauções. Uma boa precaução é que o valor das variáveis seja definido com o uso da função "define", garantindo que seu conteúdo seja conhecido e testado.

- Checar os nomes dos arquivos em uma lista de nomes válidos. Veja um exemplo: $valid_pages = array("umapagina.php" => "", "outra.php" => "", "mais.php" => "");

if (!isset($valid_pages[$page])) { // Aborte o script //Você deve provavelmente escrever uma mensagem de log aqui também die("Requisição inválida"); }

- Caso realmente precise usar variáveis de um browser, cheque os valores das variáveis usando um código como o seguinte: if (!(eregi("^[a-z_./]*$", $page) && !eregi("\\.\\.", $page))) { // Abortar o script //Você deve provavelmente escrever uma mensagem de log aqui também die("Requisição inválida "); }

- Use as configurações de variáveis “allow_url” e “open_basedir” para limitar as localizações de onde os arquivos podem ser abertos.

Utilizar Caracteres de Escape em Instruções SQL

// Usar a função para testar a existência de registro if (record_exists($query)) { echo "Acesso ragantido"; } else { echo "Acesso negado"; }

O uso da diretriz “magic_quotes_gpc” setada para On no php.ini insere caracteres de escape nas Super Globais $_POST, $_GET e COOKIES. Veja o exemplo abaixo do site oficial:

<?php echo get_magic_quotes_gpc(); // 1 echo $_POST['lastname']; // O\'reilly echo addslashes($_POST['lastname']); // O\\\'reilly

if (!get_magic_quotes_gpc()) {

```
  $lastname = addslashes($_POST['lastname']);
```

} else {

```
  $lastname = $_POST['lastname'];
```

}

echo $lastname; // O\'reilly $sql = "INSERT INTO lastnames (lastname) VALUES ('$lastname')"; ?>

Use addslashes e stripslashes, caso esteja usando variáveis globais (register_globals = On) e não esteja usando magic_quotes_gpc como no exemplo abaixo (adiciona antes de inserir no banco e remove antes de exibir na tela):

// Recebendo do Form

$thisCodigo_curso = addslashes($_REQUEST['thisCodigo_cursoField']); $thisNome = addslashes($_REQUEST['thisNomeField']);

$sqlQuery = "INSERT INTO curso (codigo_curso , nome) VALUES ('$thisCodigo_curso' , '$thisNome')";

// Enviando para a tela

eval('function cndstrips($str) {

```
 return '.(get_magic_quotes_gpc()?'stripslashes($str)':'$str').';
```

}');

Obs.: O uso simultâneo de magic_quotes_gpc = On com addslashes e stripslashes gera problemas, assim como não podemos usar register_globals = On simultaneamente com as Super Globais $_POST, $_GET.

Caso esteja usando variáveis que espera números para seu conteúdo nas instruções SQL, esteja seguro de que realmente contém números. Existem diversas funções em PHP incluindo sprintf, ereg e is_long para realizar a checagem. Também podemos utilizar o JavaScript para checar as entradas logo no formulário.

Não Confiar em Variáveis Globais

Se register_globals = On no php.ini, então o PHP criará variáveis globais para cada requisição GET, POST e variáveis Cookie.

Preste bastante atenção nas seguintes áreas:

- Código de checagem de autenticação e permissão

- Uso de variáveis antes de serem inicializadas. (Podemos ajustar error_reporting para ser alertados sempre que se usar variáveis não inicializadas.

- Uso de variáveis designadas para ser usadas por requisições GET ou POST.

Veja da documentação oficial:

Exemplos error_reporting()

<?php

// Desativa o relatório de todos os erros error_reporting(0);

// Reporta erros simples error_reporting(E_ERROR | E_WARNING | E_PARSE);

// Reportar E_NOTICE pode ser bom também (para reportar variáveis não iniciadas // ou eros de digitação em nomes de variáveis ...) error_reporting(E_ERROR | E_WARNING | E_PARSE | E_NOTICE);

// Reportar todos os erros exceto E_NOTICE // Este é o valor padrão no php.ini error_reporting(E_ALL ^ E_NOTICE);

// Reporta todos os erros (bitwise 63 deve ser usado no PHP 3) error_reporting(E_ALL);

// O mesmo que error_reporting(E_ALL); ini_set('error_reporting', E_ALL);

?>

Possíveis Correções e Melhorias

Desabilitar "register_globals" no php.ini. Após esta mudança somente podemos acessar entradas de formulários usando $_POST ou $_GET. Fica mais trabalhoso um pouco mas bem mais seguro (vale a pena).

Cuidado com o recurso "Esqueceu a Senha" de formulários de Login.

Escreva código para inicializar todas as variáveis globais.

Evite Falsos Uploads

Examine todos os scripts que respondem a upload de arquivos.

Use as funções is_uploaded_file e move_upload_file que permitem ao programador estar seguro de que está trabalhando com os devidos arquivos enviados.

Caso não tenha certeza de estar rodando em uma versão atual do PHP, configure upload_tmp_dir que executa checagem de entrada que o arquivo que estamos trabalhando está neste diretório.

Idéias Adicionais

- Encripte ou use hashes de senhas quando armazenando. A função md5 é útil para isso.

Exemplo retirado do www.phpbrasil.com:

```
//BY ADEMIR BATISTA PEREIRA $senha = "ribafs"; $resultado = md5($senha); $resultado2 =
bin2hex($senha); if($senha== ""){ $texto = ""; } else{ $texto = "
Criptografia tipo A para: $senha é $resultado"; echo $texto; $texto2 = "
Criptografia tipo B para: $senha é $resultado2"; echo $texto2; }
```

Outro exemplo do mesmo site:

<?php //***** enc - Encriptador de String //***** Autor: Ricardo Antonio Duarte - ricardo NOSPAM banhado.com

function enc($string){

```
    if((isset($string)) && (is_string($string))){
        $enc_string = base64_encode($string);
        $enc_string = str_replace("=","",$enc_string);
        $enc_string = strrev($enc_string);
        $md5 = md5($string);
        $enc_string = substr($md5,0,3).$enc_string.substr($md5,-3);
    }else{
        $enc_string = "Parâmetro incorreto ou inexistente!";
    }
    return $enc_string;
```

} //***** Fim do enc

//***** des - Desencriptador de String //***** Autor: Ricardo Antonio Duarte - ricardo NOSPAM banhado.com

function des($string){

```
    if((isset($string)) && (is_string($string))){
        $ini = substr($string,0,3);
        $end = substr($string,-3);
        $des_string = substr($string,0,-3);
        $des_string = substr($des_string,3);
        $des_string = strrev($des_string);
        $des_string = base64_decode($des_string);
        $md5 = md5($des_string);
        $ver = substr($md5,0,3).substr($md5,-3);
        if($ver != $ini.$end){
            $des_string = "Erro na desencriptação!";
        }
    }else{
        $des_string = "Parâmetro incorreto ou inexistente!";
    }
    return $des_string;
```

} //***** Fim do des

echo "Enc: ".enc("ribafs")."
";

echo "Des: ".des("53czZWYilmc1d1");

?>

Não armazene números de cartões de crédito

Force senhas seguras. No mínimo exija senhas com 8 caracteres de tamanho e que contenham algum caractere não alfanumérico.

Avoid SQL injection

This is a function which will format the passed string depending of it's specified to be a number or a string, in order to avoid problems with SQL injections in scripts.

Type: code fragment

Version:

Requires:

Added by: bto (email author)

Entered: 19/08/2004

Last modified: 08/12/2003

Rating - (fewer than 3 votes)

Views 3024

<?php function ToDBString($string, $link, $isNumber=false) {

```
   //If $isNumber==true we are specting a number
   if($isNumber)
   {
       //A correct number must be composed of:
       // - Zero or more integers followed by a decimal point and one or more
integers (i.e.: .9 (0.9) or 9.9)
       // - One or more integers followed by a decimal point. (i.e.: 9. (9.0))
       // - One or more integers (i.e.: 999)
       if(preg_match("/^d*[.,']d+|d+[.,']|d+$/A", $string))
       //If it's a correct number we change the colon, quote or point ("'", ","
or ".") by a decimal piont.
           return preg_replace( array(
                                      "/^(d+)[.,']$/"     , //9.
                                      "/^(d*)[.,'](d+)$/"  //.9 or 9.9
                                     ),
                                array(
                                      "\1."               ,
                                      "\1.\2"
                                     )
                                , $string);
       else
       //If it's not a correct number we show ERROR
           die("ERROR: Not a number"".$string.""");
   }
   else
    //If $string is a string ($isNumber==false) we return "'$string'" correctly
escaped (in this version I also strip HTML tags and modify some things in the
```

```
string, change it if you wish).
    return
"'".mysql_real_escape_string(htmlentities(strtoupper(trim(strip_tags($string)))),
$link)."'";

} ?>
```

Example

$link=mysql_db_connect("HOST", "USER", "PASSWORD");

$foo=ToDBString($_POST["string"], $link); $bar=ToDBString($_POST["number"], $link, true);

$result=mysql_db_query("DATABASE", "SELECT * FROM secret WHERE foo LIKE $foo AND bar=$bar", $link);

//If $_POST["foo"] or $_POST["bar"] are a string of this kind: " *OR 1=1" and we don't use ToDBString we will show all the info of the table!!!!*

- Usar sempre extensões tipo PHP em todos os scripts.

Nunca usar extensões .inc, .class, etc. Podemos usar assim: nome.inc.php.

Arquivos com extensões .inc, .class e similares ao serem abertos no browser exibem todo o seu conteúdo, inclusive senhas de banco, trechos de código PHP e outras, já script .php, antes de serem abertos no browser são processados pelo servidor web e só chega ao browser o resultado em HTML.

Não insira Conteúdo Sigiloso no raiz do Aplicativo

Imagens de proteção de senha, documentos e outras imagens devem ficar fora do raiz do aplicativo.

Alternativamente armazene no banco de dados.

Proteger os diretórios usando características do Apache, como arquivos .htaccess, que previnem o acesso direto ao conteúdo dos diretórios.

Muita atenção aos Serviços de Hospedagem

Estes servidores compartilham suas máquinas com diversos usuários, que têm acesso aos arquivos. Como também podem criar um arquivo de session (que armazena em /tmp por default), que pode conter maliciosos users e senhas para bypassar sua autenticação.

Verifique sempre as informações do PHP (phpinfo()) do servidor, caso use esses serviços.

Idealmente use um servidor dedicado ao invés.

Assegure-se de que o servidor ative safe_mode no php.ini.

As permissões dos arquivos são outro ponto importante. Devem somente ser lidos pelo web Server. No UNIX use algo como 711.

Use sempre validações para garantir que os usuários realmente entram com informações válidas. is_long, is_numeric, etc.

Fuja ou Evite Entrada de Usuários quando Construindo Comandos de Strings Funções como exec e eval são muito flexíveis mas requerem muito cuidado, pois os usuários podem entrar com comandos inesperados.

Evite ao máximo as funções

- eval

- preg_replace (quando usada com /e deve interpretar parâmetros como código PHP)

- exec

- passthru

- system

- popen

- (pode ser usado para executar comandos)

Além do Código (Um projeto de segurança forte)

Projeto de Aplicação Segura

- Considere o uso do HTTPS para encriptar transmissões.

- Considere restringir acesso aos diretórios usando .htaccess do Apache. Checar através das variáveis de ambiente do PHP, como $REMOTE_ADDR

- Usar pacotes de segurança existentes, como o PHPLib.

Todas as linguagens tem seus pontos fracos, mas tomando diversos cuidados preventivos e atentando para as boas regras de segurança, estes pontos podem ser protegidos. Seguindo os passos mostrados aqui podemos desenvolver um código mais seguro que o usual.

Um dos mais importantes conceitos a se ter em mente em termos de segurança é o de nunca confiar que o usuário irá digitar exatamente o que se espera que ele digite.

Nunca inclua, requeira ou abra um arquivo cujo nome seja baseado em entrada do usuário, sem antes checar

- Seja muito cuidados quando usando “register_globals = On”.

Isso foi feito para tornar o uso do PHP algo fácil, o que realmente aconteceu. Mas em contrapartida trouxe sérios problemas de segurança. A partir da versão 4.2.0 este parâmetro já vem setado como Off por default. Neste caso para pegar o valor de variáveis lançadas pelo formulário devemos utilizar as superglobais $_POST, $_GET, $_REQUEST, $_COOKIE. Ou $_SESSION.

O recomendado é que se trabalhe com “register_globals = Off”.

Usando “error_reporting = E_ALL” no php.ini recebemos uma notificação sempre que tentarmos chamar variáveis que ainda não tenham sido definida. Sabemos que o PHP não exige a definição de variáveis, mas recomenda-se que nos acostumemos a definir todas as variáveis e inclusive a iniciá-las, em termos de segurança.

Nunca execute consultas a bancos sem usar funções de escape

O PHP traz ativa por default, uma proteção contra a entrada de caracteres especiais nos formulários, que é o “magic_quotes_gpc = On”.

- Nunca confie em dados de fontes externas.

- Toda entrada de usuário deve ser validada e formatada para garantir a segurança.

Protegendo Arquivos e Diretórios com .htaccess via Apache

- Para ativar o .htaccess em todo o servidor web, edite o httpd.conf e adicione:

<Directory />

```
   Options FollowSymLinks Indexes
   AllowOverride AuthConfig
```

</Directory>

- Então criamos o arquivo .htaccess existente no diretório raiz e adicionamos tags de acordo com nossos propósitos. Exemplos:

- Restringindo o acesso por IP/Host

1. Deixa a Intranet acessar

Order allow,deny allow from 192.168.0. deny from all

Ou

1. Deixa todo mundo acessar, menos o IP 192.168.0.25

Order deny,allow deny from 192.168.0.25 allow from all

- Restringindo o acesso por user/senha

$ mkdir /etc/httpd/auth

$ cd /etc/httpd/auth

$ htpasswd -c acesso hugo

New password:

Re-type new password:

Adding password for user hugo

$ htpasswd acesso eitch

New password:

Re-type new password:

Adding password for user eitch

$ htpasswd acesso sakura

New password:

Re-type new password:

Adding password for user sakura

- Agora criar o .htaccess:

AuthName "Acesso Restrito à Usuários"

AuthType Basic

AuthUserFile /etc/httpd/auth/acesso

require valid-user

AuthUserFile /etc/httpd/auth/acesso – onde estão as senhas e users.

- Opções para arquivos e diretórios específicos:

1. Restringe o arquivo_secreto.html somente para o IP 192.168.0.30

<Files arquivo_secreto.html>

```
  Order allow,Deny
  Allow from 192.168.0.30
  Deny from all

</Files>
```

Restringe o diretório admin para utilizar senhas

<Directory /admin>

```
  AuthName "Acesso Restrito à Usuários"
  AuthType Basic
  AuthUserFile /etc/httpd/auth/acesso
  AuthGroupFile /etc/httpd/auth/grupos
  require group admin

</Directory>
```

1. Nega o acesso dos clientes ao .htaccess (bom colocar no httpd.conf)
2. - Vem com a configuração padrão do Apache

<Files ~ "^\.ht">

```
   Order allow,deny
   Deny from all

</Files>
```

Prevenindo Injeções SQL

Uma Função para prevenir tais injeções.

Esta é uma função que deve formatar a string passada dependendo de se foi especificado um número ou uma string, para evitar o problema com injeções SQL em scripts.

Autor: bto (no site

<?php function ToDBString($string, $link, $isNumber=false) {

```
   //If $isNumber==true we are specting a number
   if($isNumber) {
   //A correct number must be composed of:
   //Zero or more integers followed by a decimal point and one or more integers
(i.e.: .9 (0.9) or 9.9)
       // - One or more integers followed by a decimal point. (i.e.: 9. (9.0))
       // - One or more integers (i.e.: 999)

       if(preg_match("/^d*[.,']d+|d+[.,']|d+$/A", $string))
   //If it's a correct number we change the colon, quote or point ("'", "," or
".") by a decimal piont.
           return preg_replace( array(
                                       "/^(d+)[.,']$/"     , //9.
                                       "/^(d*)[.,'](d+)$/"  //.9 or 9.9
                                      ),
                                 array(
                                       "\1."                ,
                                       "\1.\2"
                                      )
                                 , $string);
       else
       //If it's not a correct number we show ERROR
           die("ERROR: Not a number"".$string.""");
   }
   else
    //If $string is a string ($isNumber==false) we return "'$string'" correctly
escaped (in this version I also strip HTML tags and modify some things in the
string, change it if you wish).
    return
"'".mysql_real_escape_string(htmlentities(strtoupper(trim(strip_tags($string)))),
$link)."'";

} ?>
```

Example

$link=mysql_db_connect("HOST", "USER", "PASSWORD");

$foo=ToDBString($_POST["string"], $link); $bar=ToDBString($_POST["number"], $link, true);

$result=mysql_db_query("DATABASE", "SELECT * FROM secret WHERE foo LIKE $foo AND bar=$bar", $link);

//If $_POST["foo"] or $_POST["bar"] are a string of this kind: " *OR 1=1" and we don't use ToDBString we will show all the info of the table!!!!*

Escrevendo Código Robusto em PHP

PHP Captcha Security Images

Sistema de imagens para controle de spam em PHP

http://www.white-hat-web-design.co.uk/articles/php-captcha.php

Link para download - http://www.white-hat-web-design.co.uk/articles/captcha.zip

Referências

1 – http://www.onlamp.com/lpt/a/3305 - Ten Security Checks for PHP

2 - http://www.onlamp.com/lpt/a/4045 - PHP Security

3 - http://www.devshed.com/c/a/PHP/PHP-Security-Mistakes/ - PHP Security Mistakes

4 - http://www.devin.com.br/eitch/htaccess/ - Uso e Segurança com o .htaccess

5 - http://www.securiteam.com/securityreviews/5DP0N1P76E.html - SQL injeções

6 - http://www.zend.com/codex.php?id=1405&single=1 - SQL injeções

7 – http://www.imasters.com.br/imprimir.php?cn=292&cc=44

8 – http://www.imasters.com.br/imprimir.php?cn=293&cc=44

9 – http://www.imasters.com.br/imprimir.php?cn=319&cc=44

10 - Escrevendo o código seguro de PHP

11 - - Writing Secure PHP

12 - - Vulnerabilidade em formulário PHP

Apêndices/Documentação

De Wikibooks

Documentando o Desenvolvimento

A documentação do código, do projeto, do banco e de todas as fases de um aplicativo é algo importante e mais quanto maior e complexo for o aplicativo.

Existe um utilitário que foi inspirado no JavaDoc, chamado PHPDoc ou PHPDocumentor, encontrado em - http://www.phpdoc.org/, que ajuda na documentação do código PHP.

Apêndices/Informações diversas

De Wikibooks

- PHP.net
- PHPBrasil
- Artigos, tutoriais e dicas de PHP

100 Melhores Aplicações Web

http://www.webware.com/html/ww/100.html

O que você deve saber para comprar um notebook

http://idgnow.uol.com.br/computacao_pessoal/2007/06/22/idgnoticia.2007-06-22.4095141761/IDGNoticiaPrint_view

Artigo no IDGNow com várias recomendações para comprar um Notebook.

Várias delas também se aplicam também à compra de computadores Desktop.

Eu particularmente chamaria a atenção especialmente para alguns aspectos:

- Teclado ABNT2 (português do Brasil)

- Já com o sistema operacional que você vai utilizar. Não se iluda comprando com Linux e depois instalar o Windows, por exemplo. Quase certeza que não irá encontrar todos os drivers no site do fabricante, se encontrar não há garantia de total compatibilidade. HP, Semp Toshiba, muito cuidado. Se pretende usar o Linux já compre com o Linux e com o CD de recuperação. Se pretende usar o Windows já compre com Windows e o CD de recuperação. Atualmente boa parte dos fabricantes oferecem uma partição no HD com o software de recuperação. Logo que comprar faça uma cópia desta partição ou melhor, duas cópias.

Apêndices/Oportunidade de Trabalho

De Wikibooks

16.7 - Oportunidade de Trabalho

1. Elaboração de Currículos
2. Técnicas de Entrevistas
3. Preços de Serviços

Os Sete Segredos dos que Nunca estão Desempregados

Maria Jesús Ribas em seu artigo "Como encontrar um emprego e se fortalecer no mercado de trabalho" para o Yahoo! Notícias aborda tópicos do livro "Os Sete Segredos dos que Nunca estão Desempregados" dos especialistas franceses Nicolás Buisson e Charles Henri Dumon. Veja alguns dos tópicos que achei interessantes e que podem ajudar cada um de nós:

- O dado negativo é que o sucesso profissional não está garantido a ninguém.

- O marketing pessoal, o famoso "vender o seu próprio peixe", não tem grandes segredos: consiste em estudar os pontos fortes e fracos de sua carreira para tirar proveito de suas virtudes e corrigir seus defeitos.

- Os que nunca estão desempregados - são pessoas que entendem como evolui o mercado de trabalho e sabem em qual setor ele está fortalecido. Para isso é preciso analisar seu entorno profissional, avaliando riscos e oportunidades.

Além disso estes trabalhadores conhecem bem a si mesmos, avaliando seu desempenho dentro de uma companhia, a utilidade de sua produção, sua necessidade dentro de uma companhia, a utilidade de sua produção, sua necessidade dentro da empresa, o relacionamento com seus chefes e a avaliação externa do seu trabalho.

Alguém pode estar em uma companhia perfeita, em um cargo perfeito, mas o ambiente de trabalho ou o relacionamento com as pessoas a sua volta não ser tão bom. Neste caso é preferível a busca por outro projeto.

- Pense como uma empresa - Este conceito precisa que cada profissional se veja como uma mini-

empresa que comercializa seu valor dentro da companhia. Uma pessoa passiva, que se importa apenas com seu salário e que pensa que tem um trabalho a ser feito e nada mais, precisa rever suas atitudes. O pensamento correto é o de se dedicar ao máximo para conquistar uma escalada profissional dentro de sua empresa, ocupando cada vez mais cargos importantes.

- Aumente seu nível de empregabilidade - Em vez de esperar que a companhia lhe dê uma maior formação e te presenteie com promoções, busque se especializar por conta própria. Para isso comunique aos demais o trabalho que está fazendo, sem fazer política de si mesmo e sem pisar nos outros. É preciso ter uma rede de contatos importante, tanto dentro da empresa quanto fora dela. Deve-se estudar o que pode ser melhorado, assim como suas fraquezas e não parar de "se vender" em momento algum, para que seus superiores pensem em você quando eles precisarem de alguém qualificado.

Guias de Profissões

Muitas informações úteis para quem está ainda indeciso sobre a profissão a seguir ou para quem precisa de mais informações sobre profissões:

http://www.portaldeensino.com.br/profissoes.php

http://www.guiadasprofissoes.com.br/

http://www.brasilescola.com/guiadeprofissao/

http://www3.catho.com.br/guia/busca_alfabetica.php

Apêndices/Oportunidade de Trabalho/Elaboração de Currículos

De Wikibooks

16.7.1 - Elaboração de Currículos

Deve ser conciso, resumido e preciso, para evitar que não seja lido.

Deve ser bem objetivo para atender ao objetivo solicitado.

Fale de sua experiência profissional e estudo.

Demonstre com segurança que de fato é o profissional de que a empresa precisa.

Seja convincente.

Apêndices/Oportunidade de Trabalho/Técnicas de Entrevistas

De Wikibooks

16.7.2 - Técnicas de Entrevistas

É hora de mostrar que de fato conta com as habilidades relatadas no currículo.

Fale com segurança e dê exemplos do que faz e deseja fazer.

Não aborde assuntos que não domina mas se for solicitado ao do gênero fale que não conheçe mas que tem facilidade e boa vontade para aprender.

Procure ser inovador e criativo.

Seja organizado.

Procure conhecer a empresa e traga soluções que agregem valor para a mesma.

"Nunca haverá um fim no aprendizado, a não ser que acreditemos que o conhecimento que já temos é suficiente para nós."

"O que nos fará mais desenvolvidos não é o conhecimento ao qual teremos acesso e sim o que faremos para aprimorar a humanidade com todo esse conhecimento." Frederick Motero

Apêndices/Oportunidade de Trabalho/Preços de Serviços

De Wikibooks

16.7.3 - Preços de Serviços

Uma boa idéia é procurar os sindicatos e cooperativas da área especialmente em sua região ou cidade.

Eles sempre oferecem uma tabela de preços dos serviços da categoria.

Aqui segue o endereço do **Sindicato dos Trabalhadores em Processamento de Dados e Informática no Ceará**

http://www.sindpdce.org.br/

Documento da convenção do referido sindicato, contendo tabelas com salários bases das categorias:

http://www.sindpdce.org.br/download/convencaocoletiva/cct_sindpd_seacec_jan_2006.pdf

Adicione o do seu estado e de outros estados aqui também para que tenhamos um bom banco de dados sobre nossa área.

...

Um bom artigo de Osmar Alves de queiroz Filho

Site com boas ferramentas que auxiliam no cálculo do preço de serviços de informática e outros

Tarifador de Prestação de Serviços

Programa desenvolvido por Marcelo Souza Lima para ajudar no cálculo do preço de serviços de desenvolvimento ou outros serviços.

Faça o download (tarifador.zip) do site MegaMiner.com:

http://www.megaminer.com/index.php?c=all&t=all&q=tarifador

Abaixo E-Book free sobre Empreendedorismo

Professor Dailton Felipine, do site http://e-commerce.org.br (Tudo sobre comércio eletrônico).

Apêndices/Redes

De Wikibooks

16.8 - Redes

```
Rede            IPs             Máscaras

Classe A        10.0.0.0        255.0.0.0

Classe B        172.16.0.0 até
                172.31.255.255  255.255.0.0

Classe C        192.168.0.0 até
                192.168.255.255 255.255.255.0
```

Gateway - número do IP do servidor, utilizado nas máquinas clientes.

Mais detalhes no site - http://www.infowester.com/internetprotocol.php

Descobrir o MAC address de um IP com arping

Autor: Rodrigo

Data: 28/06/2007

Descobrir o MAC address de um IP com arping

Para descobrir o MAC address de um IP:

1. arping -c2 192.168.0.1

Onde 192.168.0.1 é o IP desejado neste exemplo.

Site original - http://www.vivaolinux.com.br/dicas/verDica.php?codigo=8804

Apêndices/Editores Gráficos

De Wikibooks

Conteúdo

- 1 Gimp
  - 1.1 Vídeo Curso de Gimp em 10 Lições para Iniciantes no site do Antônio Cláudio
  - 1.2 http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=2755
  - 1.3 Edição pesada de imagens com o Gimp (parte 1)
  - 1.4 Edição pesada de imagens com o Gimp (parte 2)
  - 1.5 Edição pesada de imagens com o Gimp (parte 3)
  - 1.6 "Tatuando" pessoas com o Gimp
  - 1.7 Forum O Gimp
  - 1.8 Alguém aí falou em deformação?
  - 1.9 Deformar ou recriar?
- 2 KolourPaint
- 3 InkScape
- 4 Krita
- 5 Xara Xtreme

Gimp

O Gimp é o editor Gráfico com mais recursos do Linux. Alguns o consideram similar ao Adobe Photoshop.

A maioria das distribuições Linux já vem com ele instalado.

Vídeo Curso de Gimp em 10 Lições para Iniciantes no site do Antônio Cláudio

http://www.tuxresources.org/blog/?p=183

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=2755

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=2755

Edição pesada de imagens com o Gimp (parte 1)

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=2755

Edição pesada de imagens com o Gimp (parte 2)

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=3902

Edição pesada de imagens com o Gimp (parte 3)

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=3904

"Tatuando" pessoas com o Gimp

http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=3920

Forum O Gimp

http://www.ogimp.com.br/modules/wfsection/article.php?articleid=31

Alguém aí falou em deformação?

http://vivaolinux.com.br/artigos/impressora.php?codigo=4457

Deformar ou recriar?

http://www.vivaolinux.com.br/artigos/impressora.php?codigo=6621

KolourPaint

Um bom editor gráfico para Linux cujo nome e funções lembra o Pain do Windows mas conta ainda com mais recursos que o Paint.

No Linux Ubuntu ele não vem instalado por padrão mas através do Adicionar/Remover pode ser instalado. Em Exibir marque Todos os aplicativos disponíveis.

É um editor simples não se comparando ao Gimp, mas ideal para pequenas correções e edições de imagens.

InkScape

Outro bom editor de imagens sendo este vetorial e trabalha com arquivos do tipo SVG.

Krita

Outro editor de imagens for Linux com bons recursos.

Xara Xtreme

Site oficial - http://www.xaralx.org/

Recursos (do site):

```
   * Is very fast, very slick to use.
   * Offers some of the most powerful graphics tools available.
   * Is simple to use and learn.
   * Has a clean, un-cluttered user interface. Few floating dialogs, palettes,
menus etc.
   * Is able to create a huge range of outstanding graphics.
   * Has a huge resource of learning material, tutorials, movies, tips and a very
active enthusiastic and growing user community.
```