

```

<?php

require_once( 'CategoryTravelerBase.php' );

/**
 * Generate HTML of Category Map (Only for the first 2 level categories).
 * Output has the following format:
 * [Category 1]
 *   sub-Category 1 - sub-Category 2 - subCategory 3
 * [Category 2]
 * ...
 *
 * @package MediaWiki
 * @author Zeng Ji(zengji@gmail.com)
 */
class CategoryMap extends CategoryTravelerBase {

    var $mapTitle;

    var $categoryMapHTML;
    var $contentAfterLevel2Category = "&nbsp;|&nbsp;\n";

    function __construct($mapTitle=false) {
        parent::__construct();

        $this->mapTitle = $mapTitle;

        parent::buildCategoryTree();
        parent::travelCategoryTree();
    }

    function getCategoryMapHTML() {
        return $this->categoryMapHTML;
    }

    // Implementation of abstract function
    function travelStart() {
        if ($this->mapTitle != false) {
            $this->categoryMapHTML .= " <h3>" . $this->mapTitle . "</h3>\n";
            $this->categoryMapHTML .= " <hr/>\n";
        }
    }

    function travelEnd() {
    }

    function travelBeforeFirst($level, $categoryNode) {
        if ($level == 2) {
            $this->categoryMapHTML .= "<p>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;\n";
        }
    }
}

```

```

}

function travelAfterLast($level, $categoryNode) {
    if ($level == 2) {
        // Cut the redundant characters
        $len = strlen($this->categoryMapHTML);
        $cutlen = strlen($this->contentAfterLevel2Category);
        $this->categoryMapHTML = substr($this->categoryMapHTML, 0, $len-$cutlen);

        $this->categoryMapHTML .= "</p>\n";
    }
}

function travel($level, $categoryNode) {
    $categoryname = str_replace('_', '', $categoryNode->getCategoryName());
    $hrefstr = $this->genLinkString($categoryname);

    // Note: only output the first 2 level categories.
    if ($level == 1) {
        $this->categoryMapHTML .= "<ul><li>\n";
        $this->categoryMapHTML .= "<strong><font size='2'>\n";
        $this->categoryMapHTML .= "<a href='{ $hrefstr }'>[ ". $categoryname . " ]</a>\n";
        $this->categoryMapHTML .= "</font></strong>\n";
        $this->categoryMapHTML .= "</li></ul>\n";
    }
    elseif ($level == 2) {
        $this->categoryMapHTML .= "<a href='{ $hrefstr }'>". $categoryname . "</a>\n";
        $this->categoryMapHTML .= $this->contentAfterLevel2Category;
    }
}

// PRIVATE FUNCTIONS
function genLinkString($categoryName) {
    global $wgScript;

    $hrefstr = $wgScript . '/Category:' . $categoryName;
    return $hrefstr;
}

}

?>

```