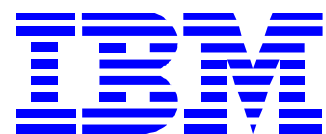


**WebSphere Application Server V8.5**  
**Liberty Profile & WDT**



## 0) IBM WAS Liberty Profile의 간략 소개

이번에 IBM에서 새롭게 출시한 IBM WAS V8.5의 가장 큰 변화 중의 하나는 Liberty Profile이라는 새로운 형태의 경량화 WAS를 함께 제공한다는 것입니다. Liberty Profile은 이미 많이 소개해 드렸지만 마치 Tomcat처럼 아주 작고 가벼워 개발 시점이나, 복잡하거나 크리티컬하지 않은 운영 업무에서 유용하게 사용될 수 있으며 기본적으로 Web, Mobile, OSGi 형태의 애플리케이션을 수행할 수 있는 경량화 형태의 IBM WAS입니다.

비록 EJB 컨테이너가 포함되어 있지 않지만 금융권을 제외하고 요즘 EJB 에 대한 요구사항이 많이 없어 지고 JPA나 Spring/Struts와 같은 프레임워크에서 트랜잭션 처리를 수행하기 때문에 다양한 요구사항에서도 충분히 사용할 수 있습니다.

만약 EJB 를 무조건 사용하신다고 한다면 IBM WAS Liberty profile에서 지원되는 OSGi 표준에서 EJB 3.0 이상을 지원하기 때문에 OSGi 형태로 개발을 진행하시면 EJB 지원도 가능합니다.

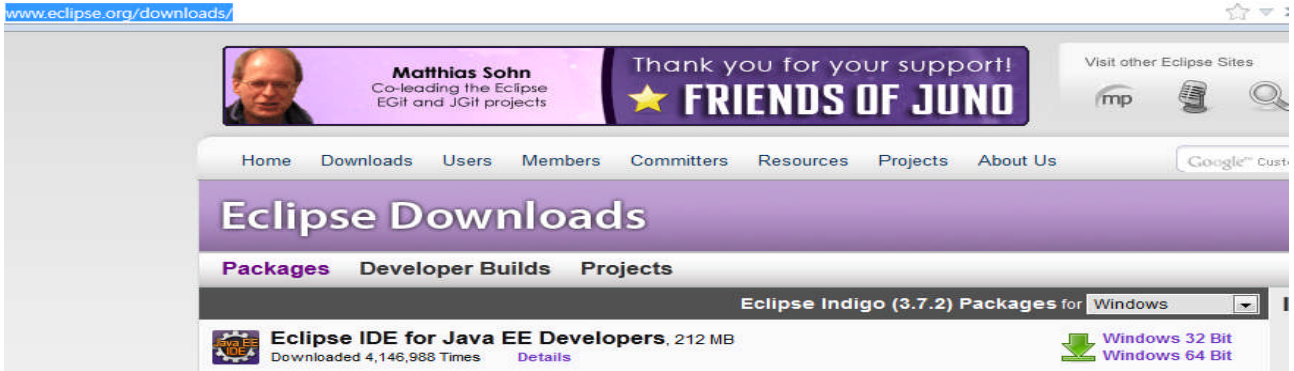
뿐만 아니라 다음과 같은 특징을 통해서 개발자들에게 개발 생산성 향상이라는 이점을 제공할 수 있습니다.

- 개발 목적으로 무료 다운로드 가능
- 설치 크기가 50MB 이하의 아주 가벼운 모듈식의 런타임
- 획기적으로 향상된 5초 이내의 빠른 시작 가능
- 빠른 개발 생산성을 위한 단순화된 설정 (server.xml 단일 설정 파일만 존재)
- Installation manager를 이용한 설치도 가능하지만 unzip 작업만으로도 설치 가능
- Eclipse plug-in으로 제공되는 WebSphere Developer Tool(WDT)

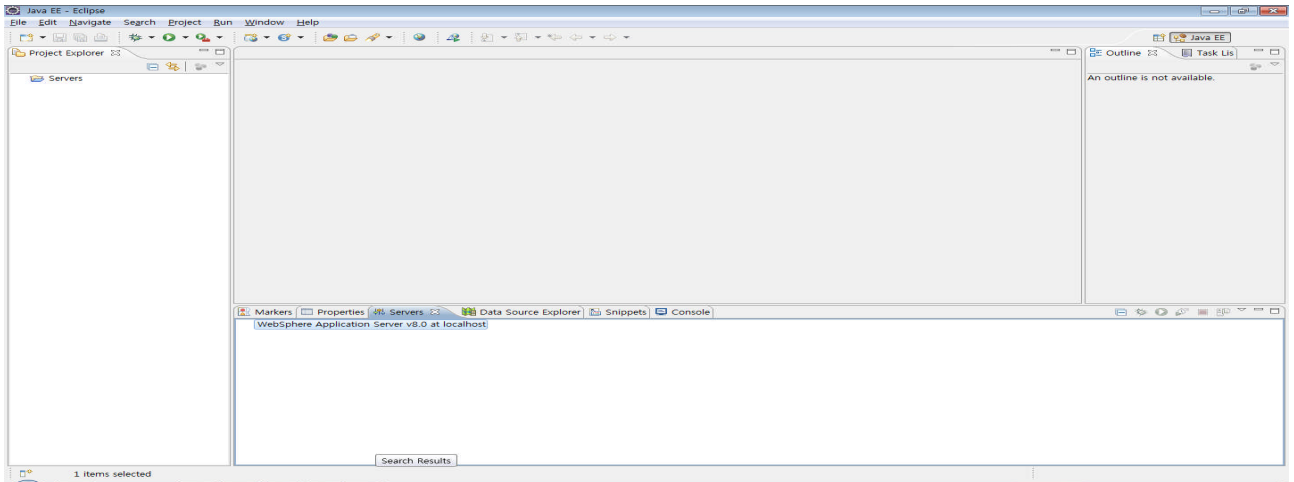
자세한 사항은 이제부터 진행되는 강좌를 참고해 주시기 바랍니다.

## 1) Eclipse 및 IBM WAS V8.5 Liberty Profile Developer Tool 설치

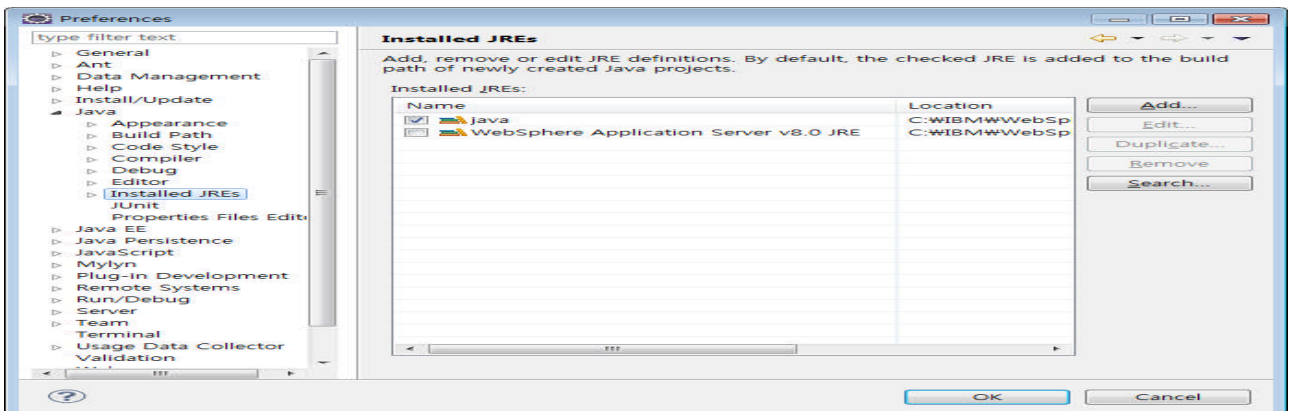
1. 하단의 링크에 접속하여 Eclipse를 설치합니다. (<http://www.eclipse.org/downloads/>) 참고로 WDT 가 지원 가능한 Eclipse 버전은 3.6.2 Helios와 3.7.2 Indigo입니다.



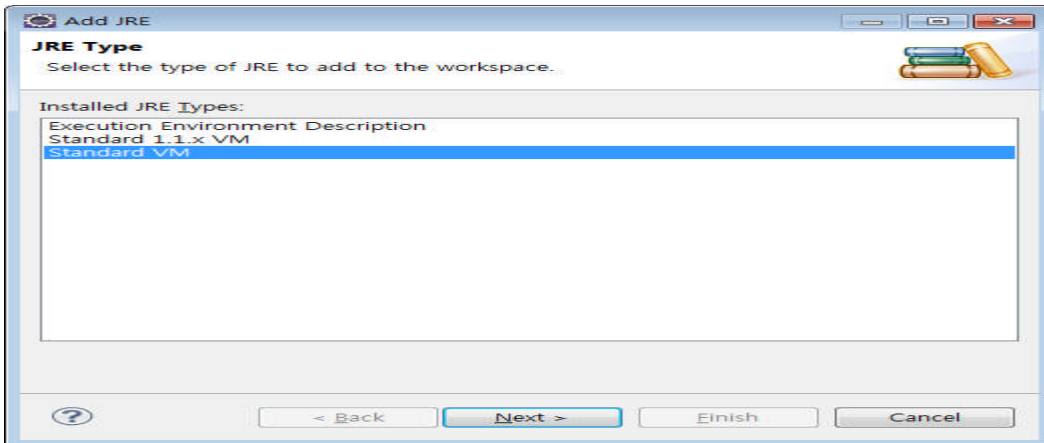
2. 설치가 완료되었으면 해당 Eclipse를 실행합니다.



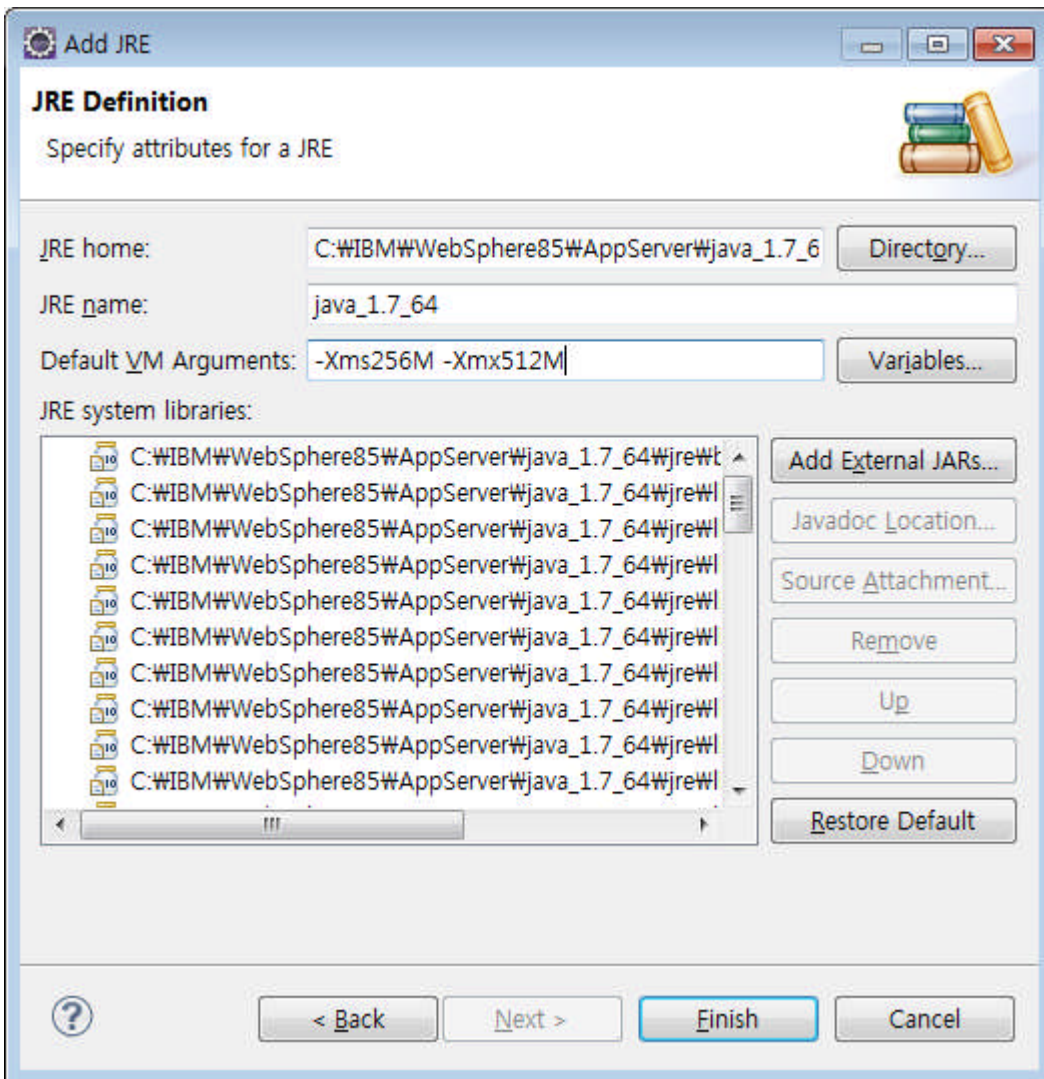
3. 해당 Eclipse를 새롭게 설치한 JDK 7.0 기반으로 JRE를 변경하고 간단하게 Heap을 늘리기 위하여 Windows > preference 메뉴를 클릭한 후 Java > Installed JREs를 선택하고 나서, Add를 클릭합니다.



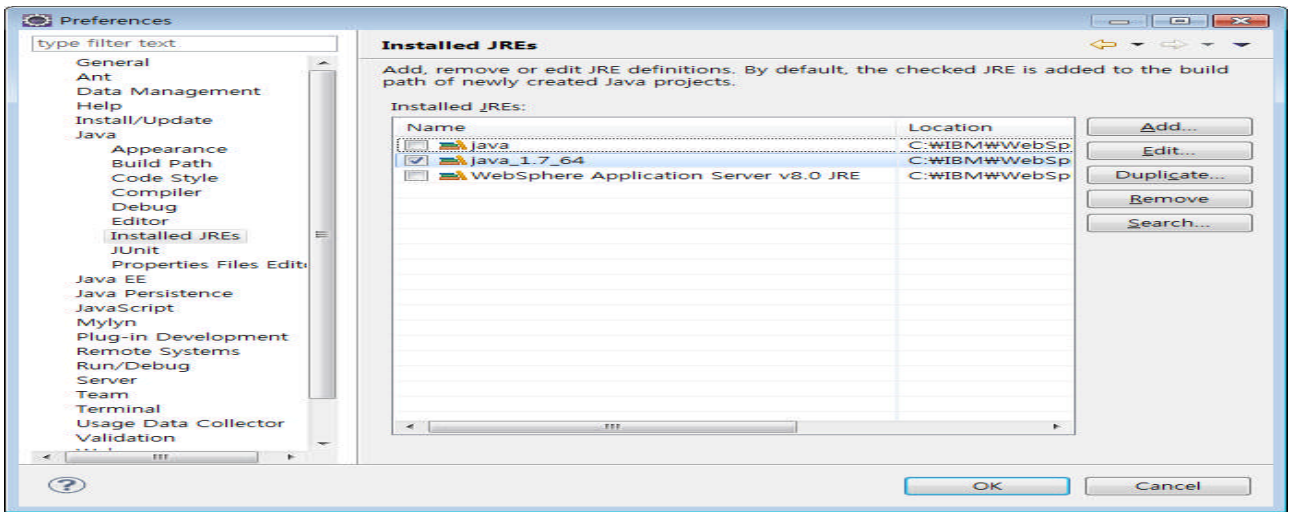
4. JRE 추가를 위하여 Standard VM을 선택하고 다음을 클릭합니다.



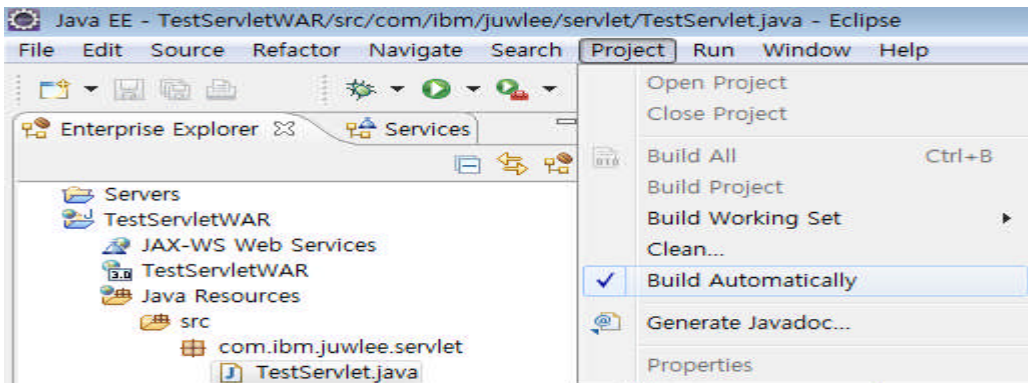
5. 지난 강좌에서 설치한 JDK 7.0 위치를 선택하면 자동으로 다른 항목들이 들어갑니다. 추가적으로 간단한 튜닝을 위해서 "-Xms256M -Xmx512M" 값을 Default VM Arguments 에 추가한 후 Finish를 클릭하여 완료합니다.



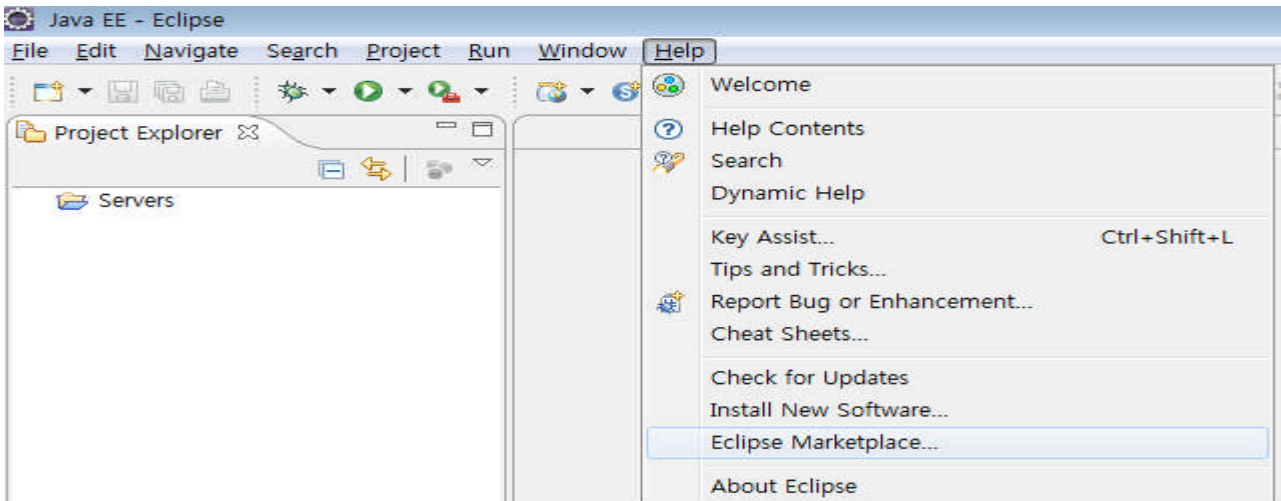
6. 기본 JRE를 방금 추가한 JDK 7.0으로 변경하고 OK를 클릭하면 JRE 변경이 모두 완료됩니다. 해당 JRE 변경 작업이 완료되면 eclipse를 다시 구동합니다.



7. 추가적으로 매번 저장할 때마다 빌드가 되면, 너무 잦은 빌드가 이루어지므로, 수동으로 빌드 타임을 조정할 수 있도록 Project > Build Automatically 체크를 해제해 둡니다.



8. 이전 단계까지 eclipse 설치는 완료되었으며, 이제 실제적으로 IBM WAS V8.5 연동 개발을 위한 plugin인 WDT를 다운로드 받기 위하여 Eclipse Marketplace를 선택합니다.

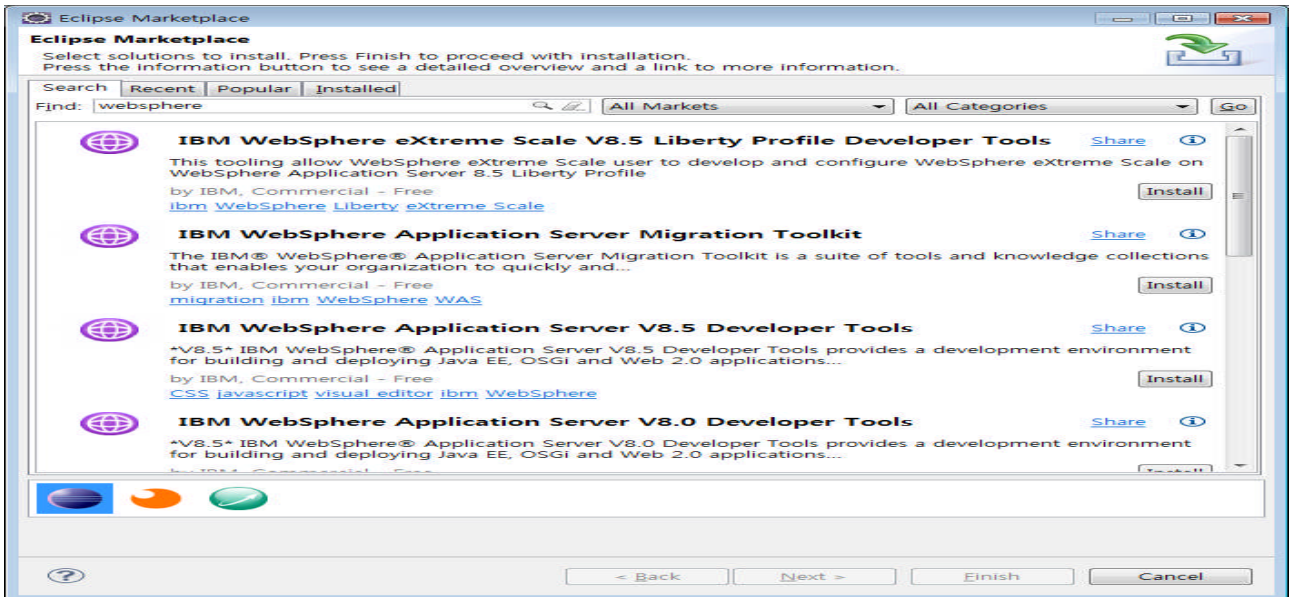




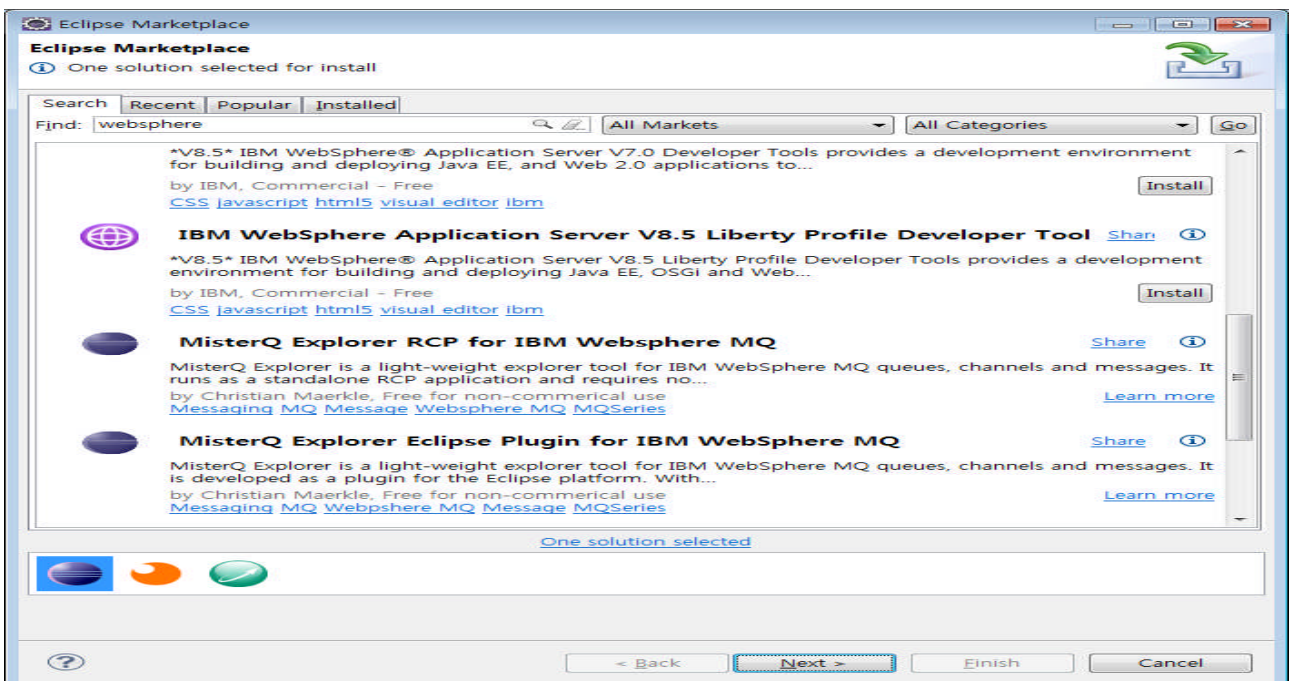
9. Eclipse Marketplace 화면에서 websphere로 검색하면 하단과 같이 다양하게 IBM WAS와 연관된 WDT를 설치할 수 있는 화면이 나옵니다. 참고적으로 만약 3.6.2 Helios 사용을 선택하신 분들은 다운로드 받으신 후에 먼저 최신 버전으로 Eclipse update를 수행하시고 난 후 WDT를 설치하셔야 정상적으로 WDT가 설치됩니다.

(참고 : System requirements for WebSphere Application Server Developer Tools for Eclipse

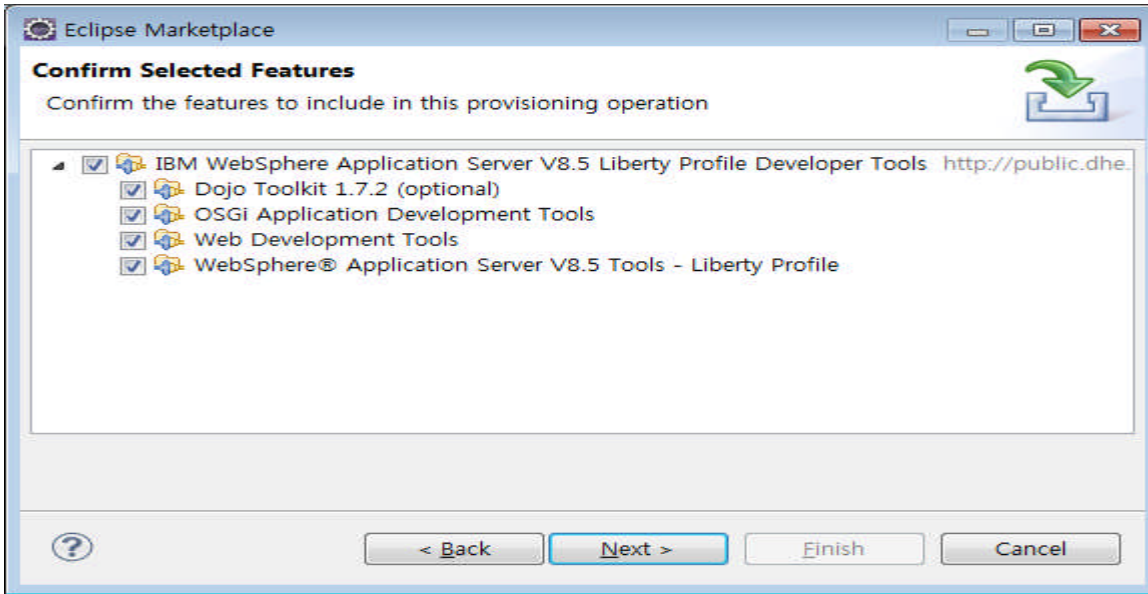
[http://publib.boulder.ibm.com/infocenter/radhelp/v8r5/topic/com.ibm.rad.install.doc/topics/r\\_wdt\\_reqs.html](http://publib.boulder.ibm.com/infocenter/radhelp/v8r5/topic/com.ibm.rad.install.doc/topics/r_wdt_reqs.html)



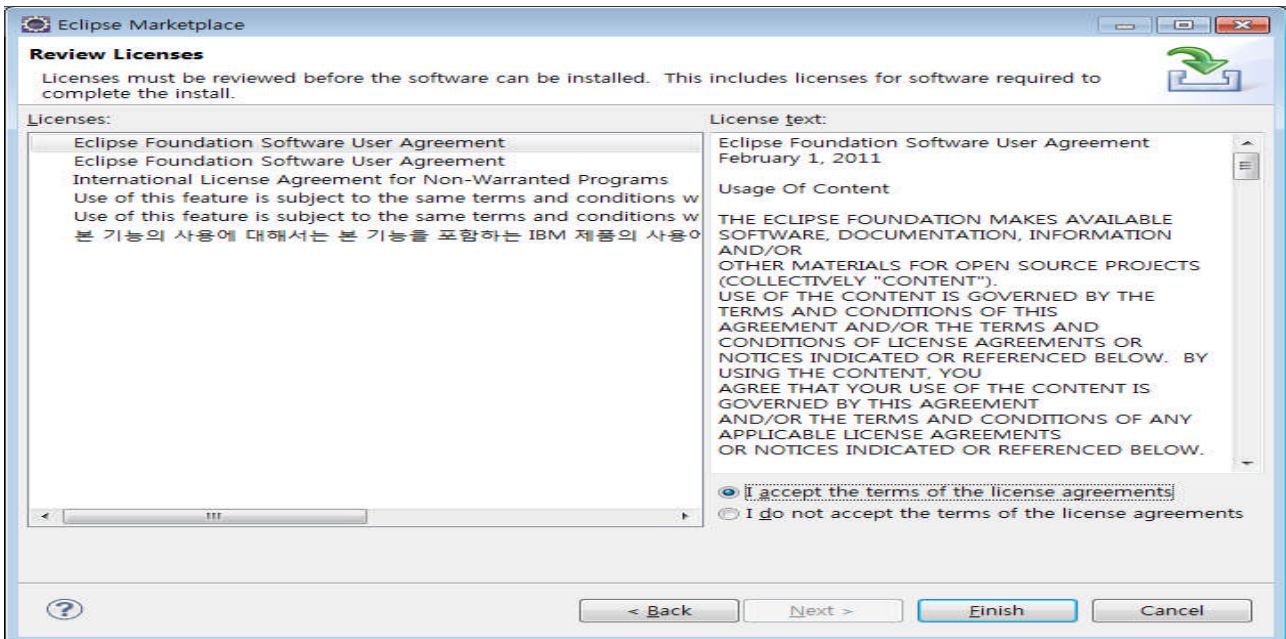
10. 그 중에 이번 강좌에서 사용할 IBM WAS V8.5 Liberty Profile Developer Tool을 찾은 후에 Install 버튼을 클릭합니다.



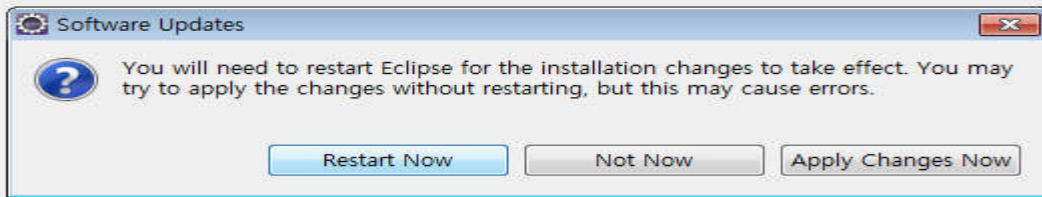
11. IBM WAS v8.5 Liberty Profile Developer Tool의 세부 설치 내용을 확인하고 다음을 클릭합니다.



12. License 사용 동의를 하고 Finish를 클릭하면 실제 설치가 진행됩니다.



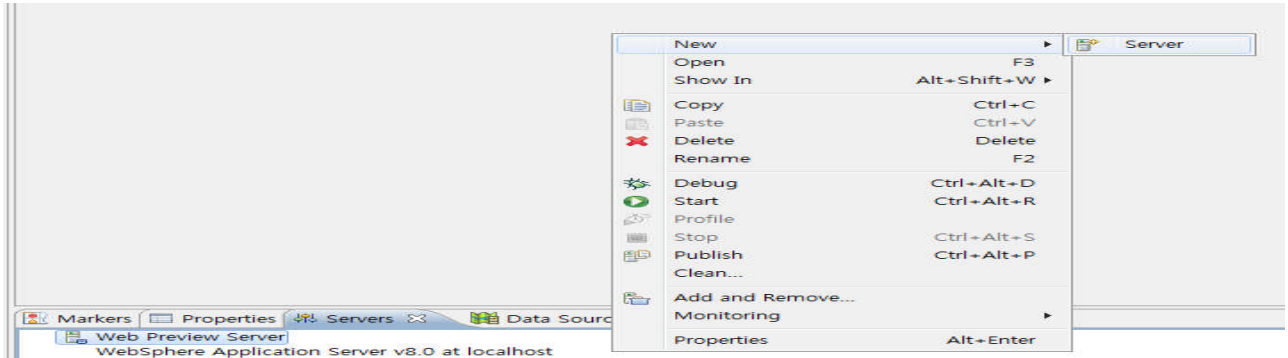
13. 설치가 완료되고 하단과 같은 창이 나오면 재시작을 선택합니다.



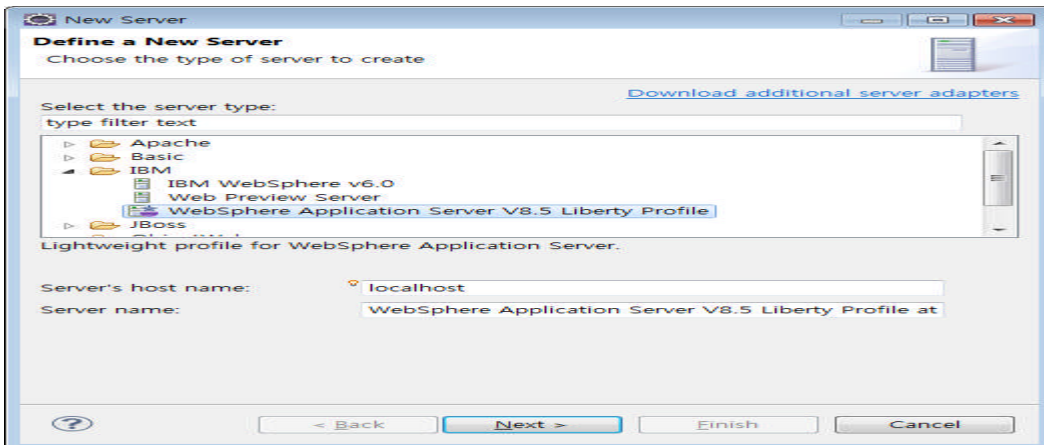


## 2) IBM WAS V8.5 Liberty Profile 추가 및 설정

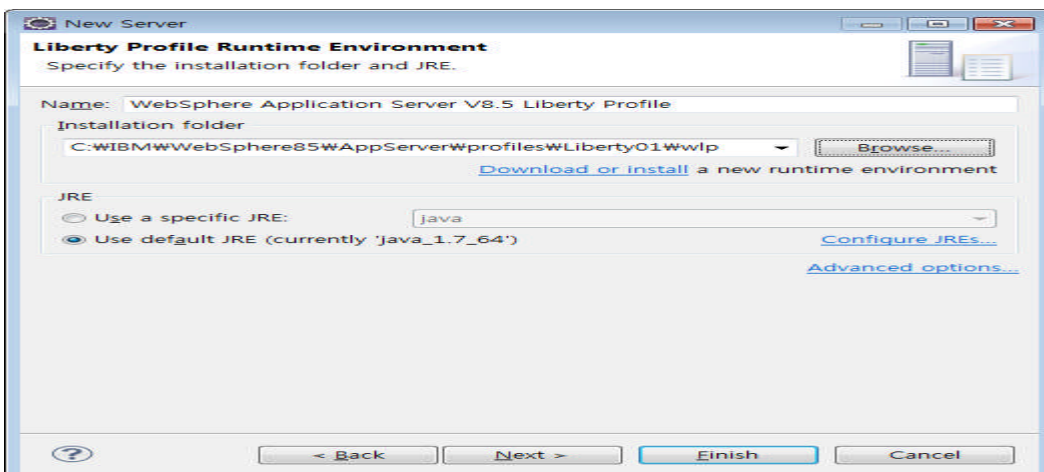
1. 이전 과정에서 설치가 완료되었기 때문에 IBM WAS V8.5 Liberty Profile 추가 및 설정을 위하여 eclipse 하단에 있는 servers 탭으로 가서 마우스 오른쪽을 클릭한 후 New > Server를 선택합니다.



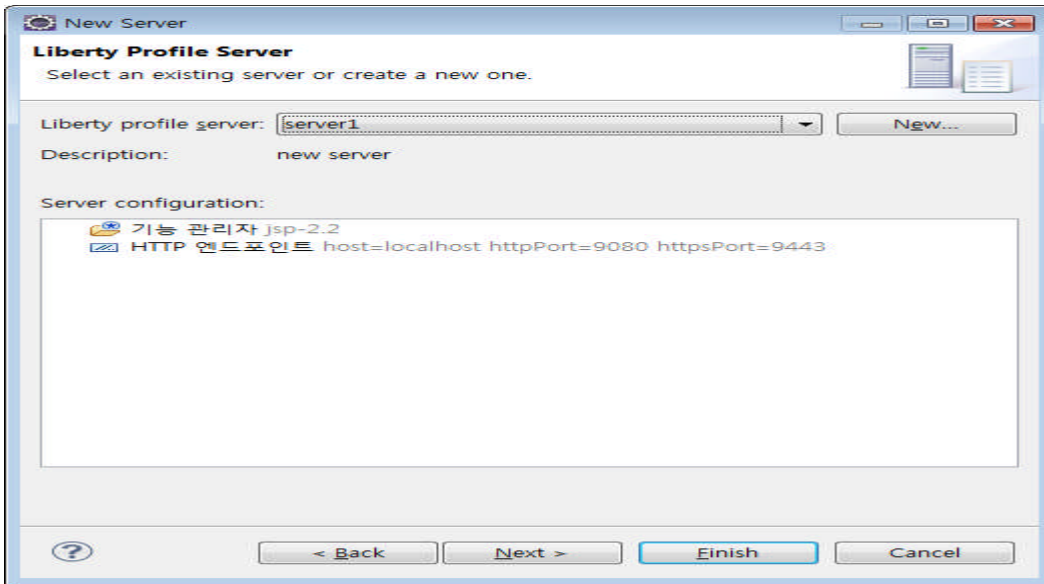
2. Server 리스트 중에서 WebSphere Application Server v8.5 Liberty Profile을 선택하고 다음을 클릭합니다.



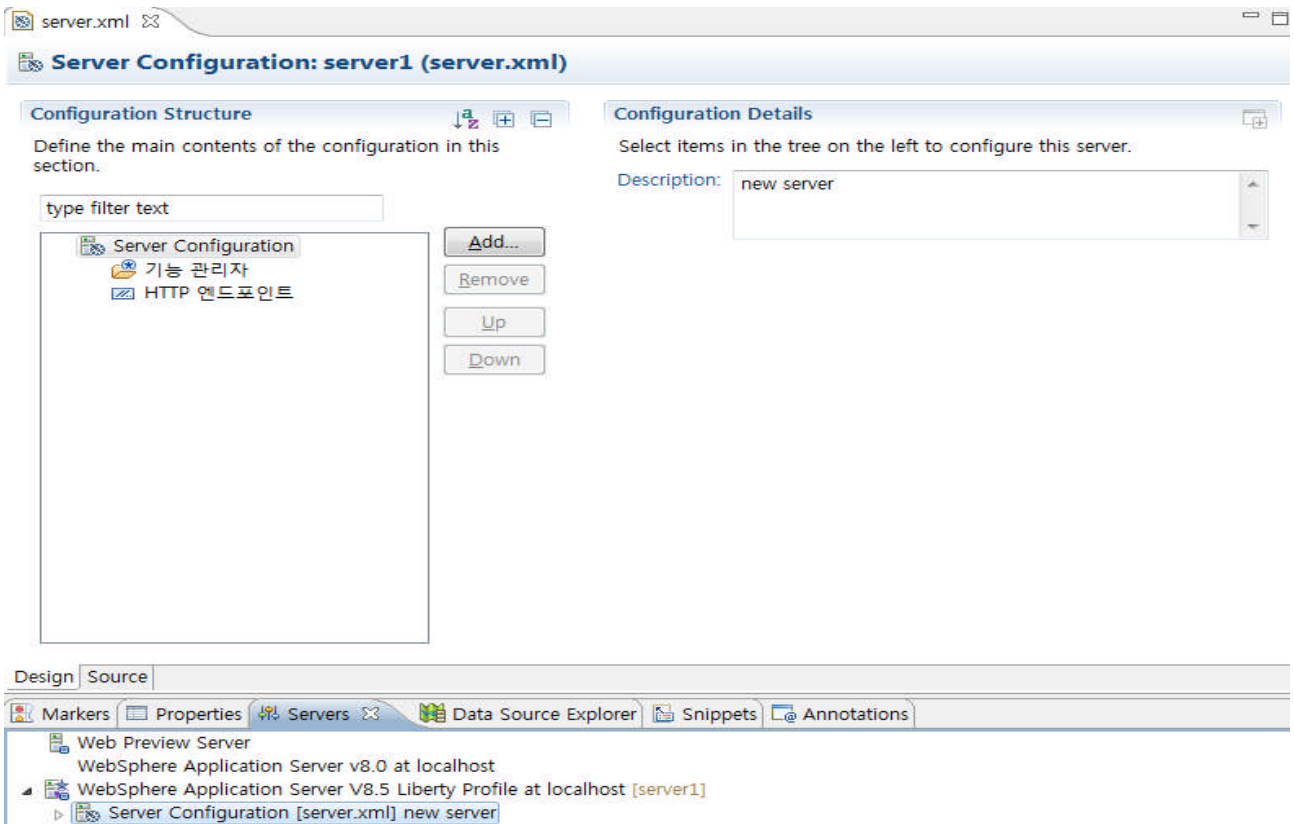
3. 첫 번째 강좌(또는 본 강좌 하단에 Appendix #1 참고)에서 설치한 IBM WAS V8.5 Liberty Profile의 위치를 선택한 후에 Next를 선택합니다.



4. 이전에 만든 server1이 선택된 것과 기본 기능만 설치된 것을 확인하고 설정을 완료합니다.

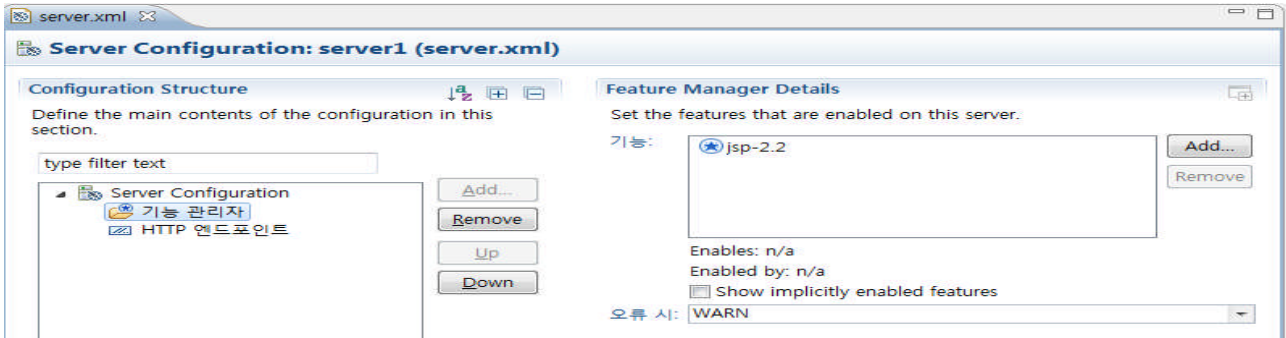


5. 여기까지 작업을 완료하면 하단과 같이 eclipse와 IBM WAS V8.5 Liberty Profile의 연동 작업을 무사히 완료하신 것입니다. 이제 이를 이용해서 개발을 진행하시면 됩니다.

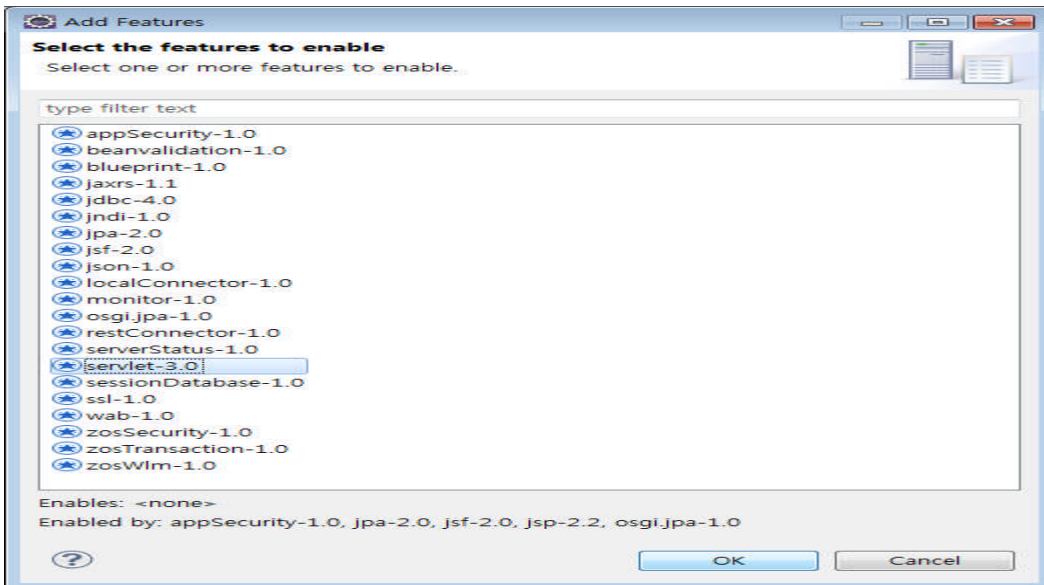


### 3) IBM WAS v8.5 Liberty Profile 환경에서 Servlet Sample 개발

1. Eclipse 와 IBM WAS V8.5 Liberty Profile Developer Tool의 설치 및 준비가 끝났으므로 해당 서버를 이용한 Servlet Sample 페이지 개발을 진행해 보도록 하겠습니다. 기능 추가를 하기 위해 server.xml 을 클릭한 후 기능관리자를 선택하여 Add 버튼을 클릭합니다.



2. Servlet을 개발하기 위하여 servlet-3.0 기능을 선택하고 OK를 클릭합니다.



3. servlet-3.0 기능이 추가된 것을 확인합니다.

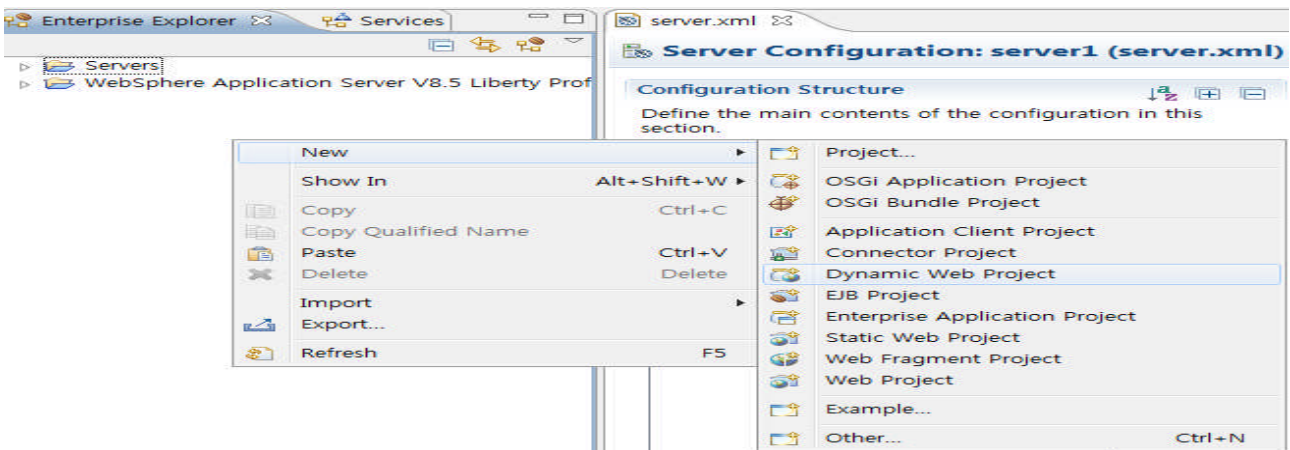


```

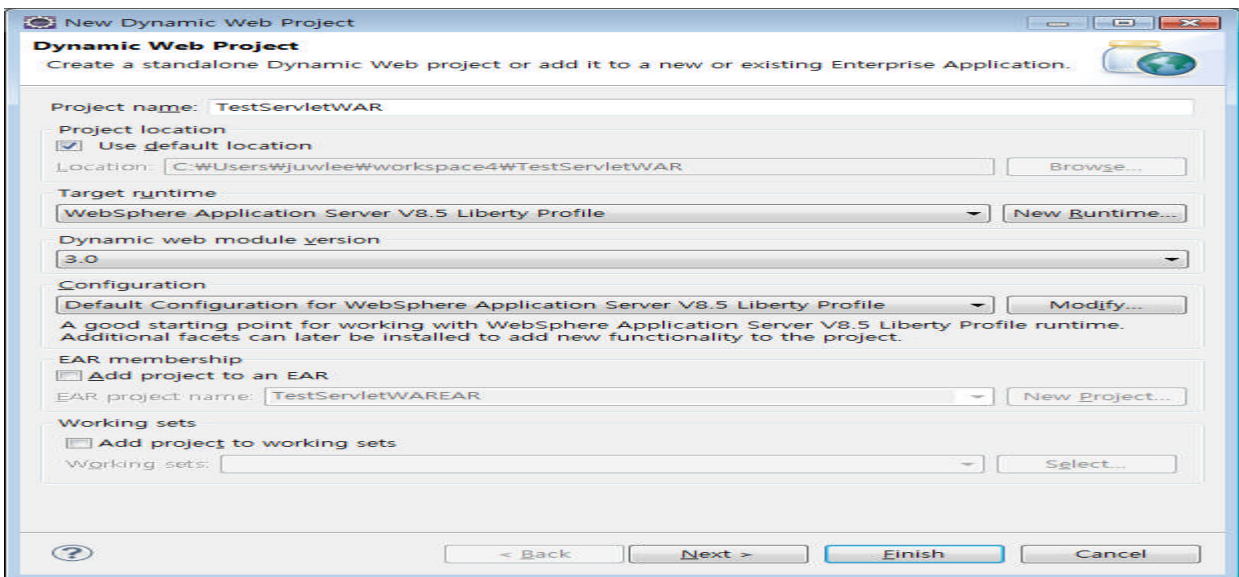
server.xml
<server description="new server">
  <!-- Enable features -->
  <featureManager>
    <feature>jsp-2.2</feature>
    <feature>servlet-3.0</feature>
  </featureManager>

  <httpEndpoint id="defaultHttpEndpoint"
    host="localhost"
    httpPort="9080"
    httpsPort="9443" />
</server>
  
```

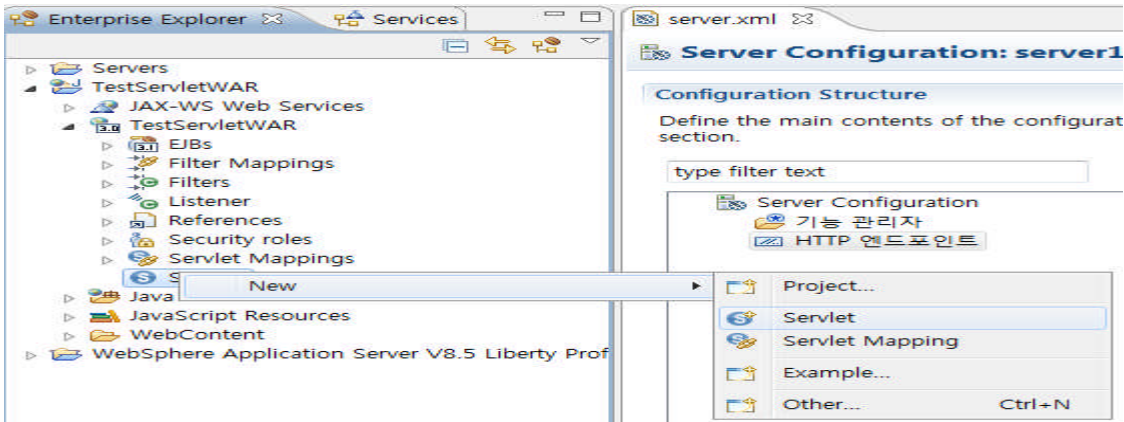
4. Server 쪽의 설정을 끝내고, Project Explorer 메뉴에서 Servlet 샘플 어플리케이션을 만들기 위하여 마우스 오른쪽 클릭을 진행한 후 New > Dynamic Web Project를 선택합니다



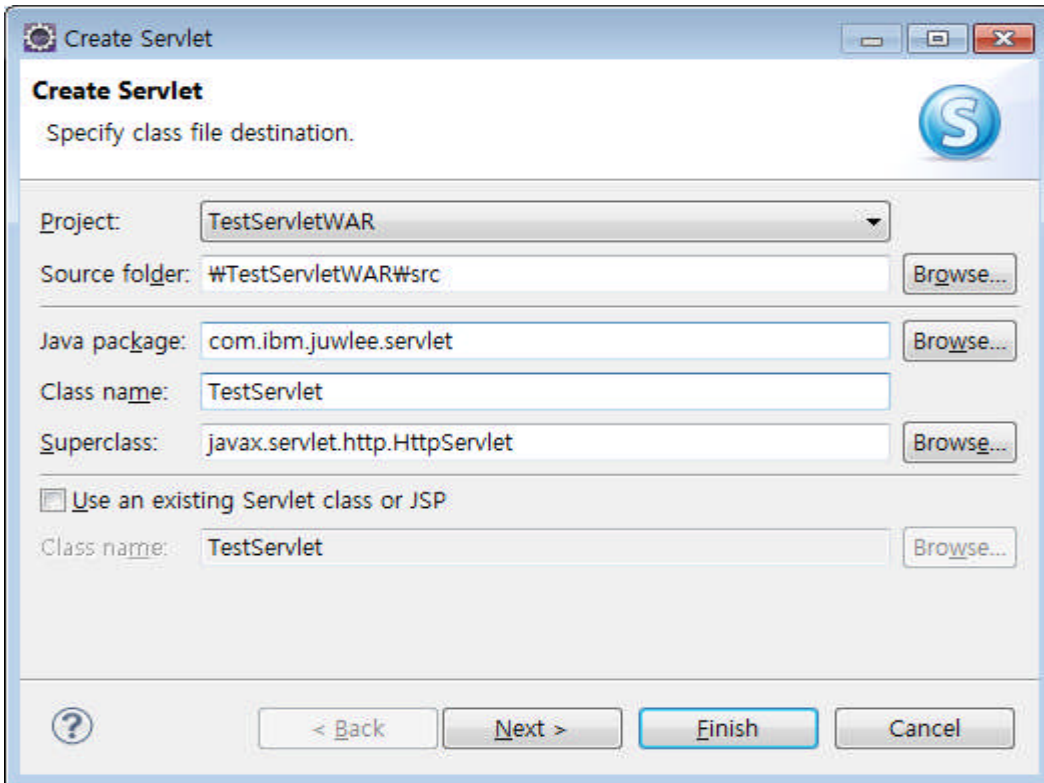
5. 프로젝트 이름에 TestServletWAR 를 입력하고 Target runtime을 IBM WAS V8.5 Liberty Profile로 선택합니다.



6. 해당 프로젝트가 왼쪽 탭에 추가된 것을 확인하고, 실제 Servlet을 만들기 위해서 Servlet에서 마우스 오른쪽 클릭을 수행하여 New > Servlet을 선택합니다.



7. Servlet을 생성하기 위해서 class name과 java package 이름을 입력하고 Finish를 클릭합니다.



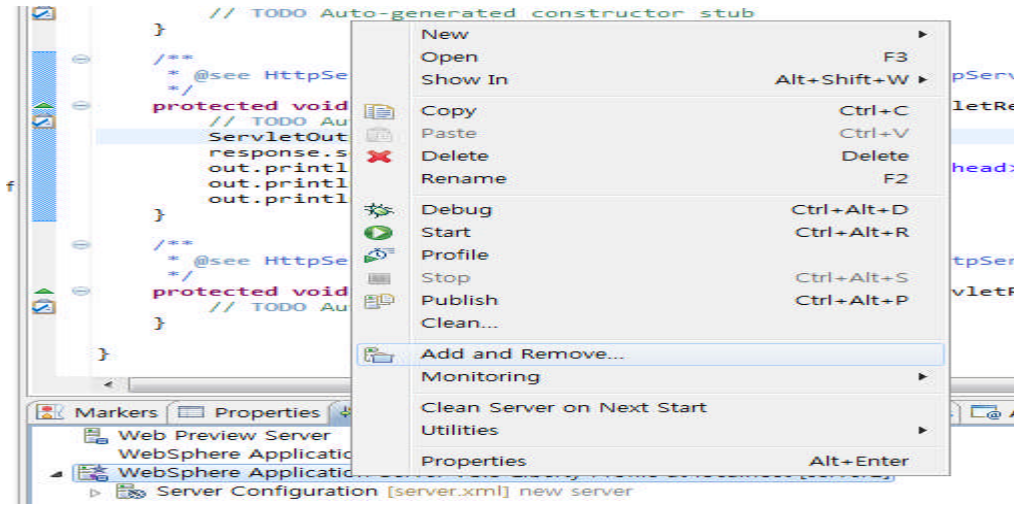
8. eclipse 에서 자동으로 생성된 servlet 템플릿에서 간단한 화면 출력을 하기 위하여 doGet() 메소드에 하단의 내용을 추가합니다.

```

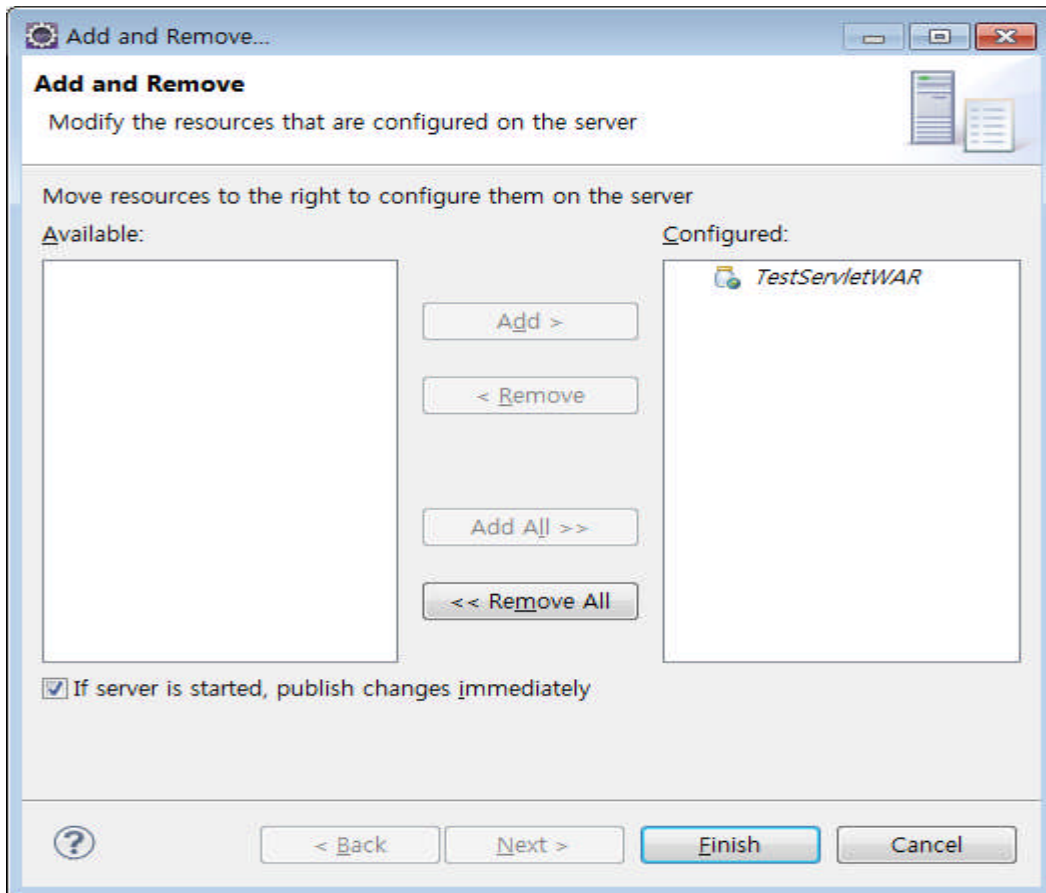
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException
// TODO Auto-generated method stub
ServletOutputStream out = response.getOutputStream();
response.setContentType("text/html");
out.println("<html><head><title>Hello World</title></head>");
out.println("<body><h1>Hello World !!!</h1></body>");
out.println("</html>");
}
    
```



9. 이제 만들어진 TestServletWAR 패키지를 IBM WAS V8.5 Liberty Profile 에 배포하기 위하여, 해당 서버를 선택하고 마우스 오른쪽 클릭을 수행한 후 Add and Remove 메뉴를 선택합니다.

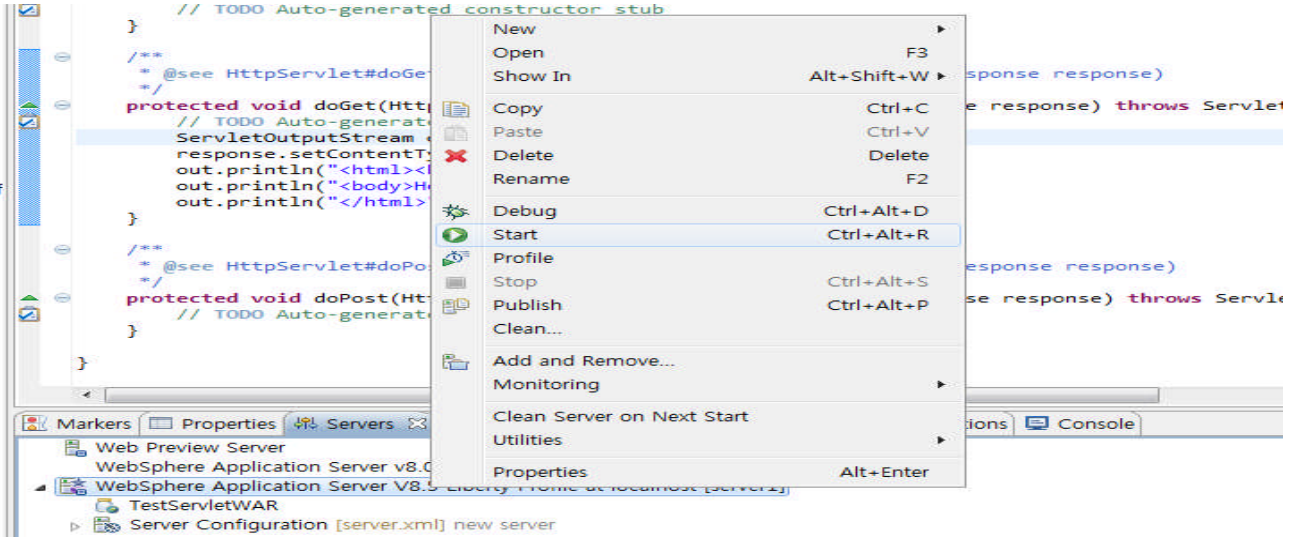


10. 왼쪽에 있는 TestServletWAR를 선택하고 Add 버튼을 클릭하여 오른쪽으로 이동된 것을 확인하고 Finish를 클릭하여 배포를 종료합니다. 이렇게 하면 Servlet을 테스트하기 위한 모든 준비가 완료되었습니다.

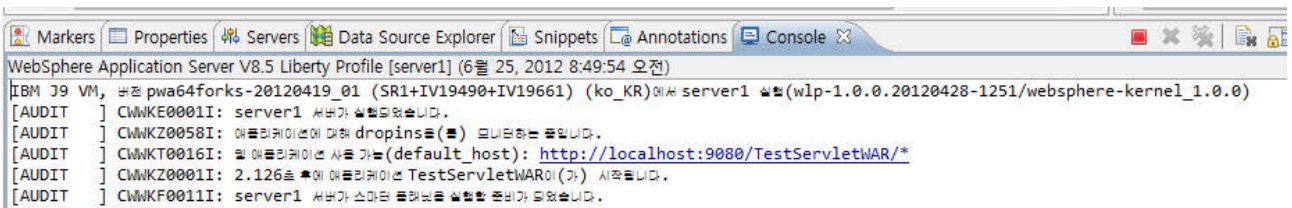


## 4) Servlet Sample 테스트

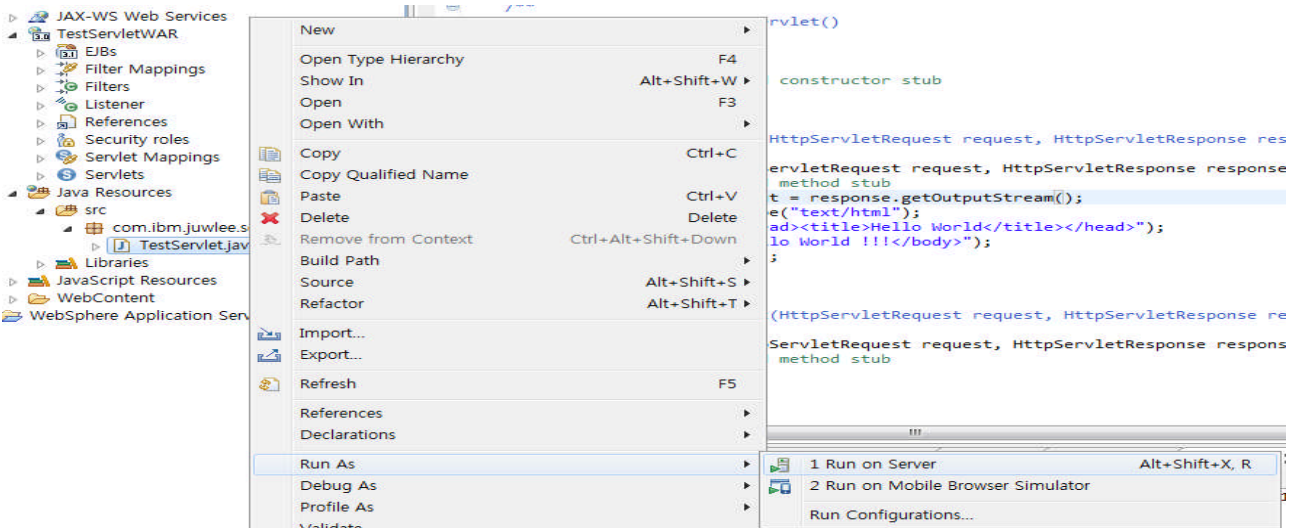
1. 지금까지 만든 Servlet Sample을 테스트하기 위하여 하단의 servers 탭에서 IBM WAS V8.5 Liberty Profile을 선택하고, 마우스 오른쪽 클릭 후 Start 버튼을 클릭하여 해당 서버를 시작시킵니다.



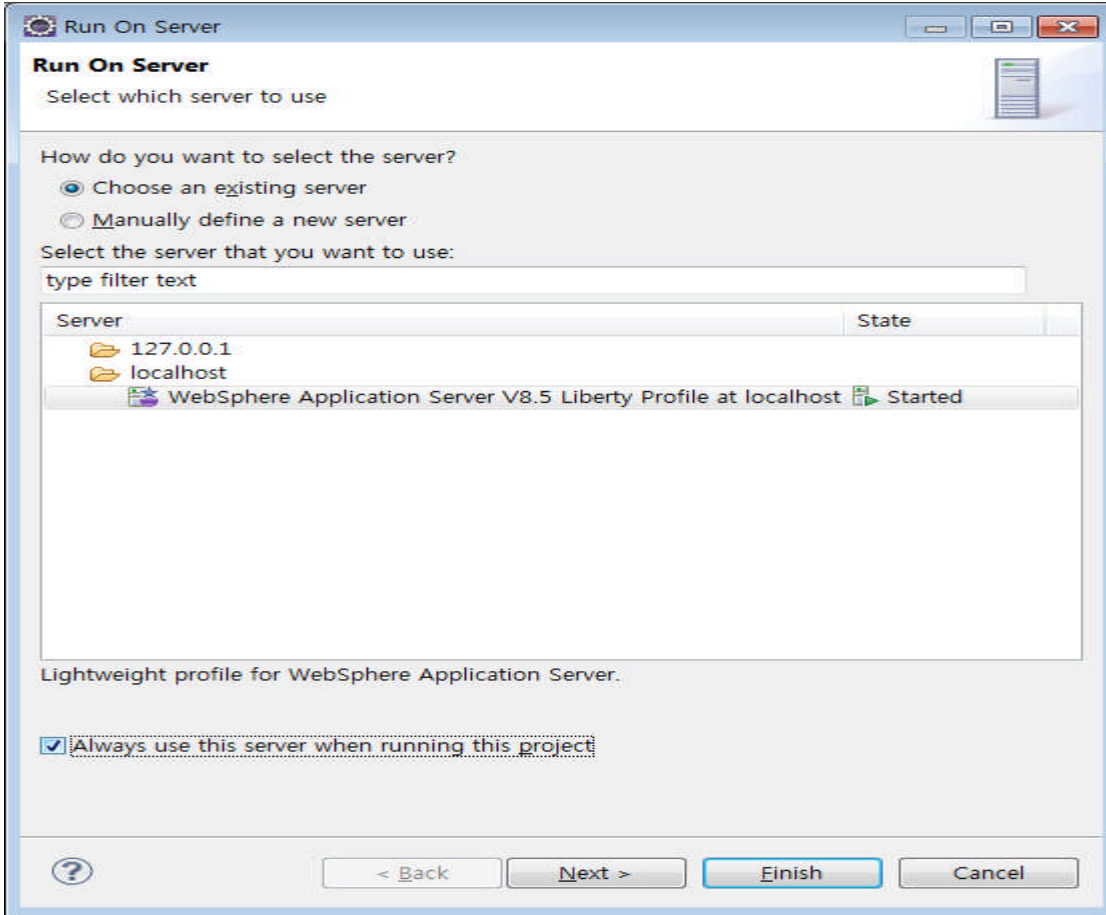
2. IBM WAS V8.5 서버가 순식간에 시작되는 것을 Console에서 확인할 수 있습니다.



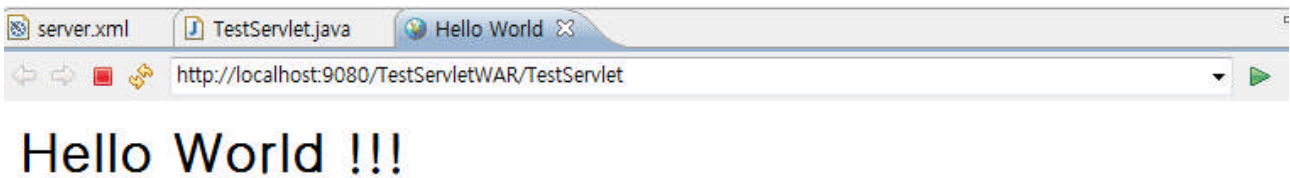
3. 이제 미리 만들어 둔 Servlet Sample을 테스트 하기 위하여 왼쪽 탭에서 해당 클래스를 선택하고, 마우스 오른쪽 버튼을 클릭한 후 Run As > Run on Server를 선택합니다.



4. Servlet Sample을 구동하기 위한 서버 선택 화면이 나오면 WebSphere Application Server V8.5 Liberty Profile을 선택합니다. 이 때, 'Always use this server when running this project'를 체크해 두면 다음부터 해당 창은 더 이상 나오지 않고 무조건 해당 WAS로 실행됩니다.



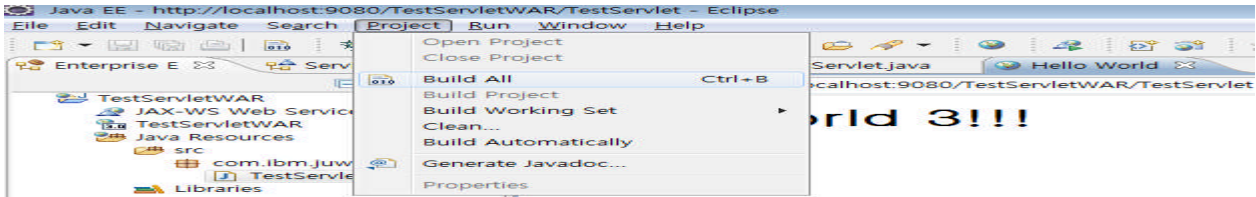
5. 화면에 하단과 같이 정상적으로 'Hello World !!!'가 나오면 Servlet Sample 테스트에 성공하신 것입니다.



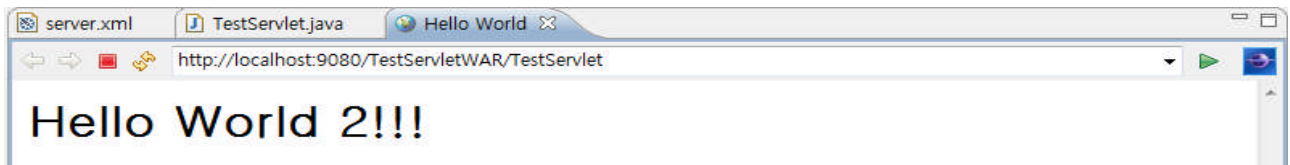
6. 지금 실행된 Servlet인 TestServlet.java 파일을 열어서 약간의 수정을 하고 저장 버튼을 클릭합니다.

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException {
    // TODO Auto-generated method stub
    ServletOutputStream out = response.getOutputStream();
    response.setContentType("text/html");
    out.println("<html><head><title>Hello World</title></head>");
    out.println("<body><h1>Hello World 2!!!</h1></body>");
    out.println("</html>");
}
```

7. 자동 빌드기능을 꺼둔 경우에는 하단과 같이 Project > Build All 메뉴를 클릭하면 애플리케이션 빌드가 수행됩니다.

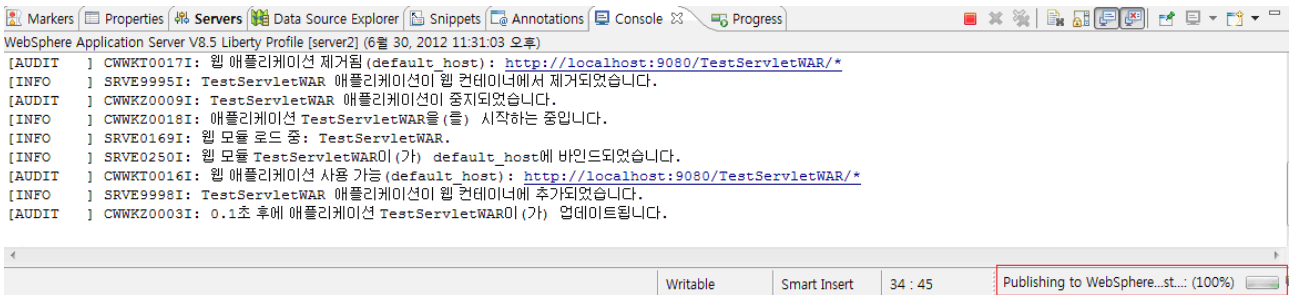


8. 다시 한 번 Servlet 페이지를 수행하면 해당 WAS의 재시작 없이도 변경사항이 하단과 같이 반영된 것을 확인할 수 있습니다.

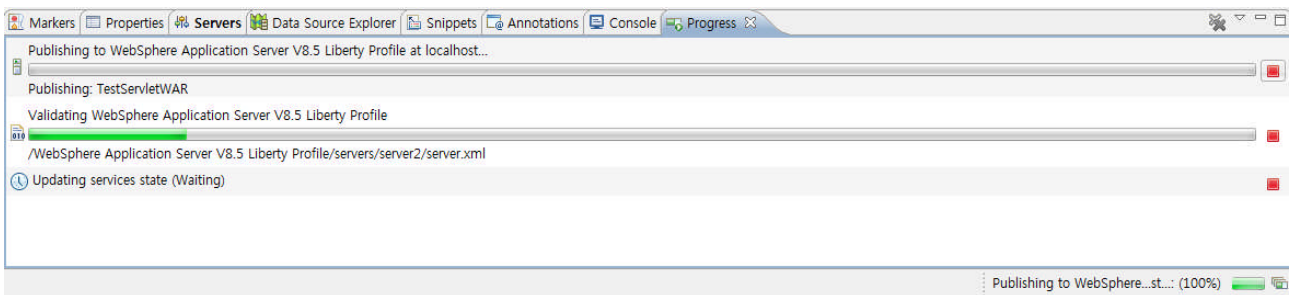


**Tip #1**

만약 Windows 7 한글 환경에서 애플리케이션 변경에 따른 재배포 작업하시는 중에 하단과 같이 Publishing to WebSphere 부분에서 100%에 도달했음에도 불구하고 멈추어 있는 것과 같은 현상이 발생한다면

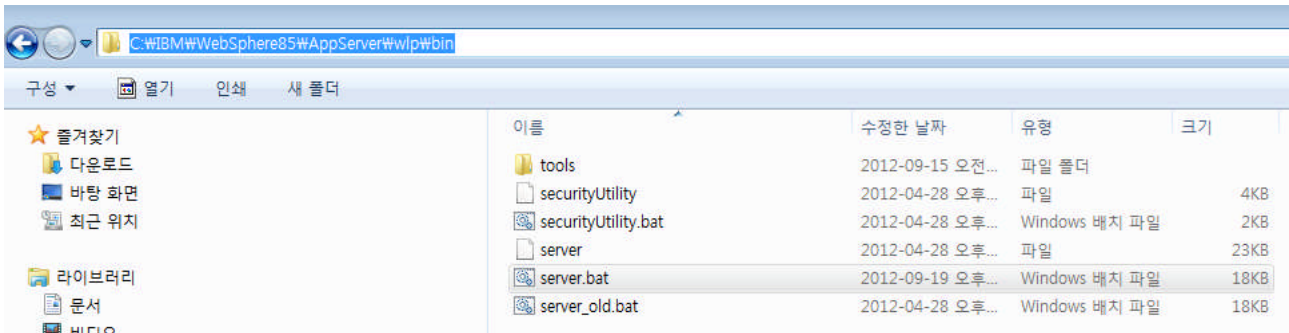


해당 부분을 클릭해서 Progress 부분을 확인하고 Publishing to WebSphere 항목의 정지 버튼을 클릭하여 해당 작업을 중지시키면 정상적으로 반영이 완료됩니다.



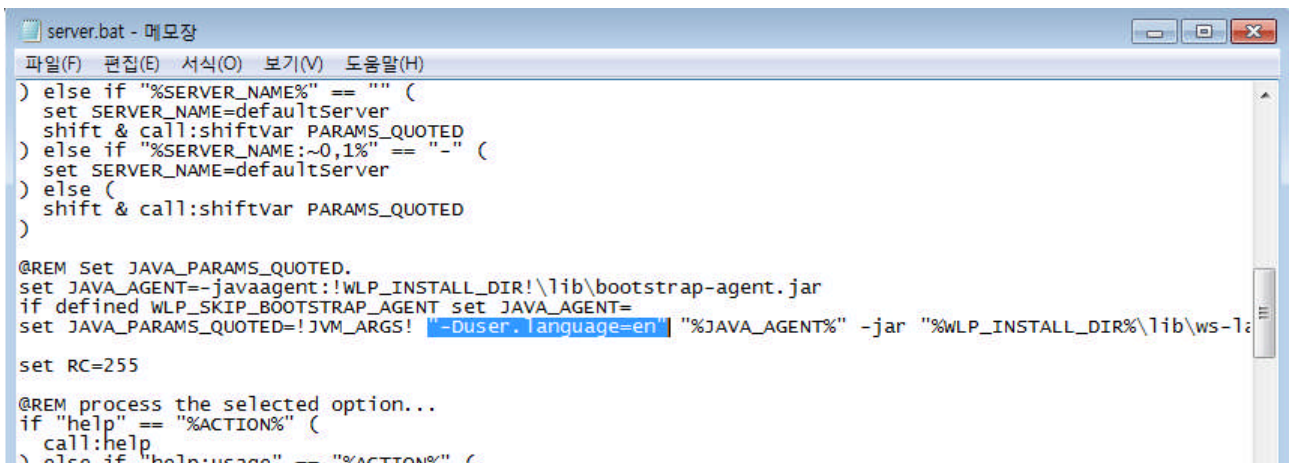


또한, 다른 대안으로 Liberty 설치 위치\bin 디렉토리에 가보면 server를 수행하기 위해 사용되는 server.bat 파일이 있습니다.



해당 파일을 편집모드로 열어 110 번째 라인에서 옵션("-Duser.language=en") 추가하여 하단과 같이 WAS 구동시 언어를 영어로 변경해 주시면 로그가 영어로 나오긴 하지만 해당 문제를 해결할 수 있습니다.

```
set JAVA_PARAMS_QUOTED=!JVM_ARGS! "-Duser.language=en" "%JAVA_AGENT%" -jar "%WLP_INSTALL_DIR%\lib\ws-launch.jar"
```

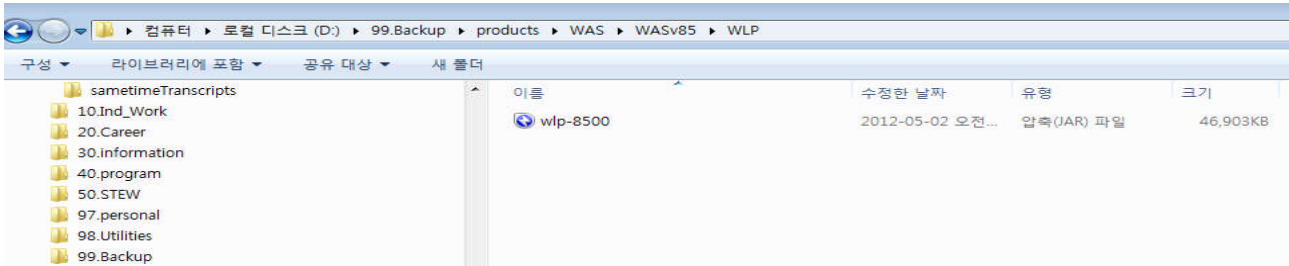




## Appendix #1 – IBM WAS v8.5 Liberty Profile 설치하기

1. IBM WAS V8.5 의 이미지를 풀면 하단과 같이 WLP 폴더에 IBM WAS v8.5 Liberty Profile jar 파일이 들어 있는 것을 확인할 수 있습니다. 또는, 하단의 developerworks 사이트에서 무료로 다운로드 받을 수 있습니다.

[https://www.ibm.com/developerworks/mydeveloperworks/blogs/wasdev/entry/download\\_wlp?lang=en](https://www.ibm.com/developerworks/mydeveloperworks/blogs/wasdev/entry/download_wlp?lang=en)



2. 이를 원하는 폴더에 위치시킨 후에 “java -jar wlp-8500.jar” 명령을 수행하면 IBM WAS V8.5 Liberty Profile에 대한 설치가 하단과 같이 완료됩니다.

```
C:\IBM\WebSphere85\AppServer\profiles\WLiberty01>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 6806-ABBD

C:\IBM\WebSphere85\AppServer\profiles\WLiberty01 디렉터리

2012-06-18 오후 09:53 <DIR> .
2012-06-18 오후 09:53 <DIR> ..
2012-05-02 오전 11:35 48,028,532 wlp-8500.jar
                1개 파일 48,028,532 바이트
                2개 디렉터리 32,506,163,200 바이트 남음

C:\IBM\WebSphere85\AppServer\profiles\WLiberty01>java -jar wlp-8500.jar
IBM WebSphere Application Server Network Deployment Version 8.5을(를)
사용하거나 압축을 풀거나 설치하려면 먼저 국제 프로그램 라이선스 계약의 이용 약관
및 추가 라이선스 정보에 동의해야 합니다.
다음 라이선스 계약을 주의깊게 읽으십시오.

--viewLicenseAgreement 옵션을 사용하여 라이선스 계약을 별도로 볼 수 있습니다.
지금 라이선스 이용 약관을 표시하려면 Enter를 누르고 건너뛰려면 'x'를 누르십시오.
x
--viewLicenseInfo 옵션을 사용하여 추가 라이선스 정보를 별도로 볼 수 있습니다.
지금 추가 라이선스 정보를 표시하려면 Enter를 누르고 건너뛰려면 'x'를 누르십시오.
x
아래 "동의함" 옵션을 선택하면 라이선스 계약의 이용 약관 및 비IBM 이용 약관이 적
용되는 경우 이에 대한 이용 약관에
동의합니다. 동의하지 않는 경우 "동의 안함"을 선택하십시오.

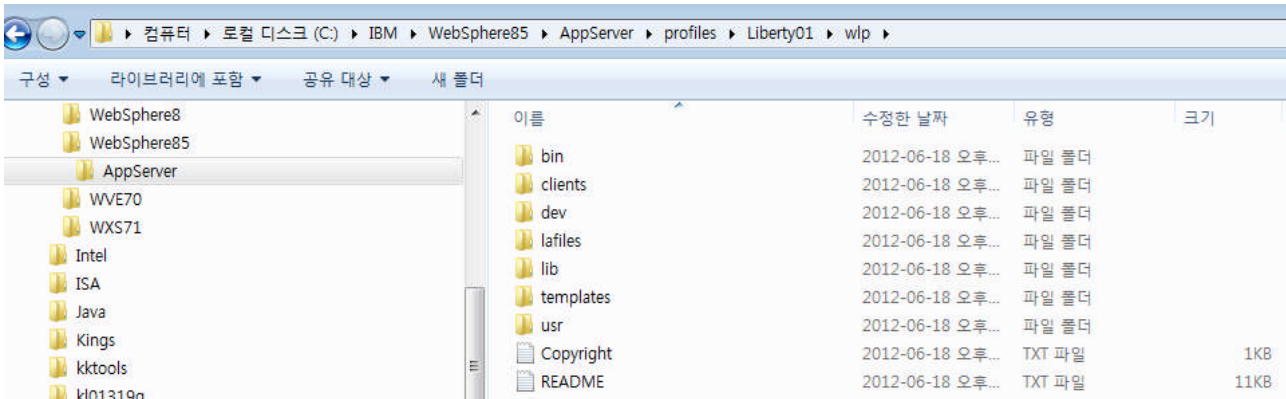
[1] 동의 또는 [2] 동의 안함 선택: 1

제품 파일의 디렉토리를 입력하거나 공백으로 두어 기본값을 승인하십시오.
기본 대상 디렉토리는 C:\IBM\WebSphere85\AppServer\profiles\WLiberty01입니다.

제품 파일의 대상 디렉토리입니까?
C:\IBM\WebSphere85\AppServer\profiles\WLiberty01에 파일 압축을 푸는 중입니다.
모든 제품 파일의 압축 풀기를 완료했습니다.

C:\IBM\WebSphere85\AppServer\profiles\WLiberty01>
```

3. 설치가 완료되면 하단과 같이 wlp 폴더의 세부 디렉토리가 풀린 것을 확인할 수 있습니다. 이게 IBM WAS v8.5 Liberty Profile 의 설치의 전부입니다.



4. 방금 설치한 IBM WAS v8.5 Liberty Profile의 bin 디렉토리에서 "server create server1" 명령을 수행하면 server1이 생성됩니다. 이게 IBM WAS V8.5 Liberty Profile의 서버 생성의 전부입니다.

```

C:\IBM\WebSphere85\AppServer\profiles\Liberty01\wlp\bin>server -help
사용법: server 조치 serverName [옵션]
serverName
  서버의 로컬에서 고유한 이름. 이름을
  ASCII 영숫자 문자(A - Z, a - z, 0 - 9), 밑줄(_), 대시(-) 및
  마침표(.)를 사용하여 구성할 수 있습니다.

조치:

create
  지정된 서버가 없으면 새 서버를 작성합니다.

dump
  진단 정보를 서버에서 아카이브로 덤프합니다.

package
  서버를 아카이브에 패키징합니다.

start
  이름 지정된 서버를 시작합니다.

status
  이름 지정된 서버의 상태를 검사합니다.

stop
  이름 지정된 서버의 실행 중 인스턴스를 중지합니다.

version
  서버 버전 정보를 표시하고 종료합니다.

C:\IBM\WebSphere85\AppServer\profiles\Liberty01\wlp\bin>server create server1
server1 서버가 작성되었습니다.

C:\IBM\WebSphere85\AppServer\profiles\Liberty01\wlp\bin>
    
```

## 5) 결론

이번 강좌에서는 IBM WAS V8.5 Liberty profile과 이를 위한 Eclipse plug-in인 WDT 를 같이 살펴보고, Servlet 샘플 페이지의 작성 및 테스트를 진행해 보았습니다.

설치 및 테스트를 해보면서 아시겠지만 기존의 IBM WAS하고는 약간의 다른 개념으로 개발자에 접근하고 있으며, 개발자 입장에서 아주 빠르고 편리하게 개발 및 테스트를 진행할 수 있습니다. 특히 관리 콘솔을 통한 작업이 없으므로 불편한 점도 있지만 하나의 server.xml을 통해서만 작업하므로, 설정의 직관성과 개발시의 편리함은 더 뛰어납니다.

많은 사용 부탁드립니다.