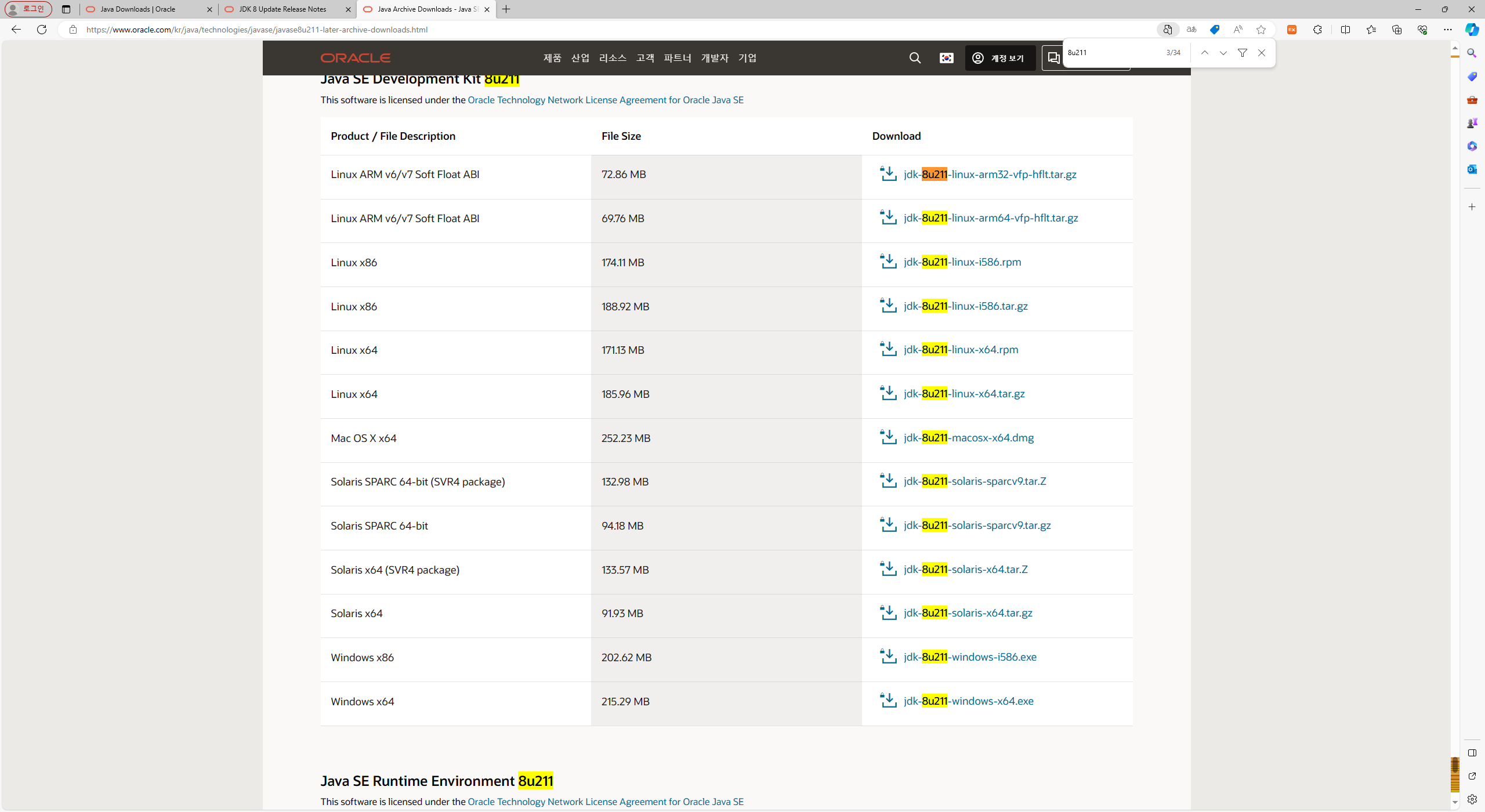
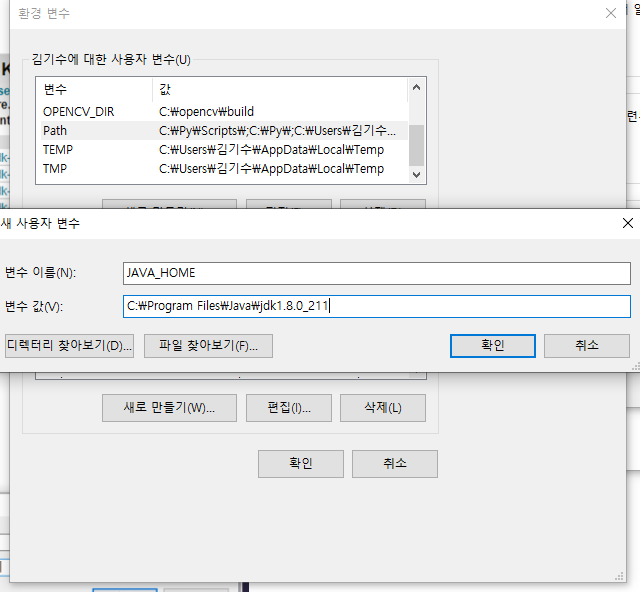
**1-1. JDK 1.8 설치하기**

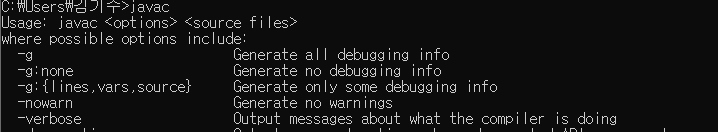


JDK 8(8u211)버전 인스톨러를 오라클 사이트에서 다운로드 받는다.

[Java Archive Downloads - Java SE 8u211 and later | Oracle 대한민국](https://www.oracle.com/kr/java/technologies/javase/javase8u211-later-archive-downloads.html)

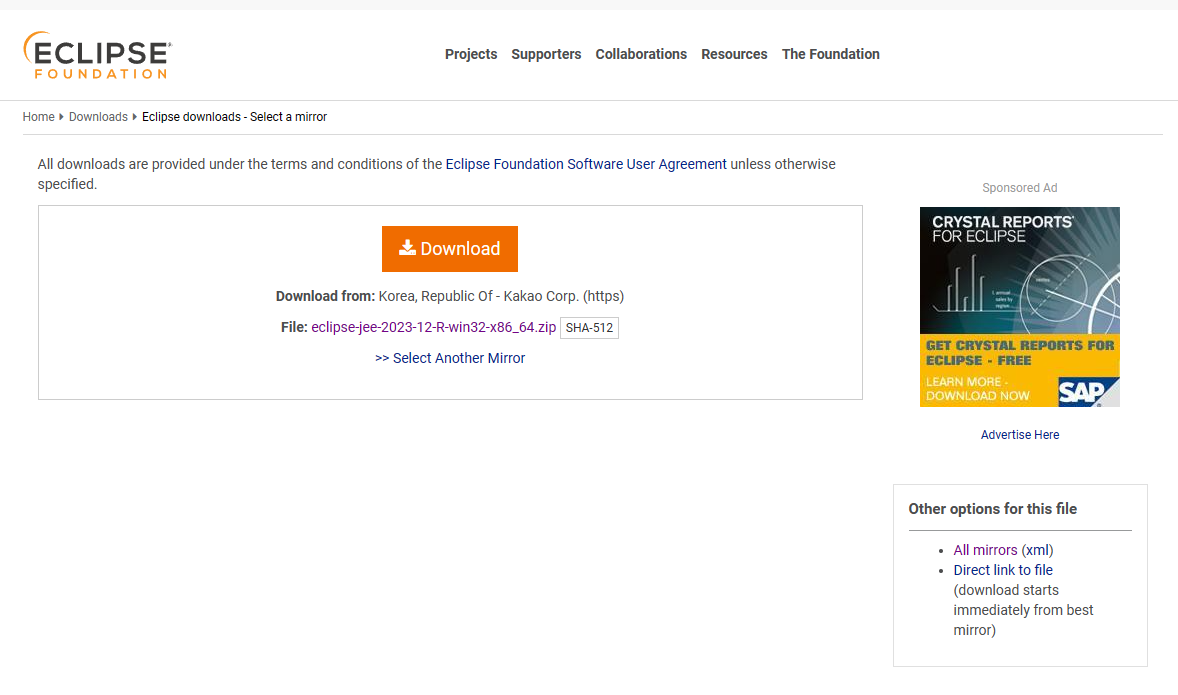


JAVA\_HOME 환경변수를 설치되어있는 위치로 설정한다. Path 변수에 %JAVA\_HOME%\bin 를 추가한다.



설치 및 환경변수 설정이 잘 되어있다.

**1-2. Eclipse IDE 설치하기**

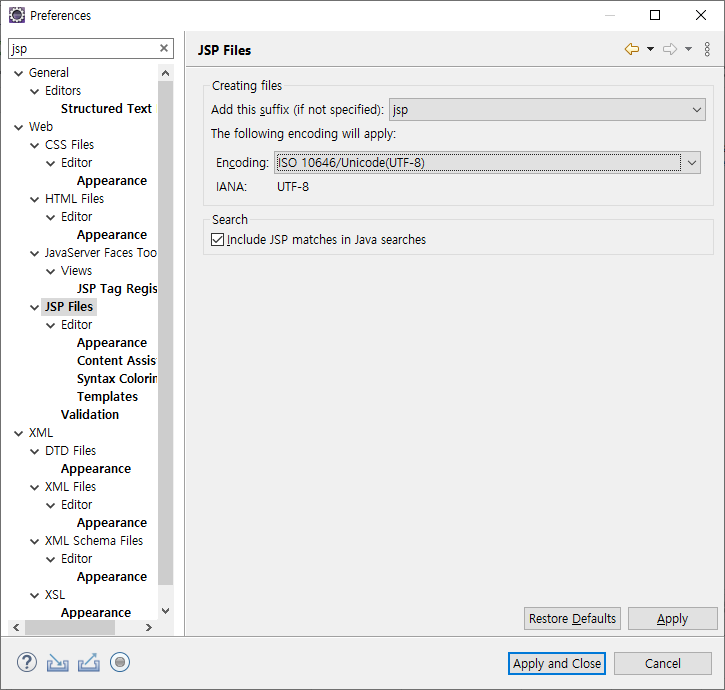


Java EE for developers 설치



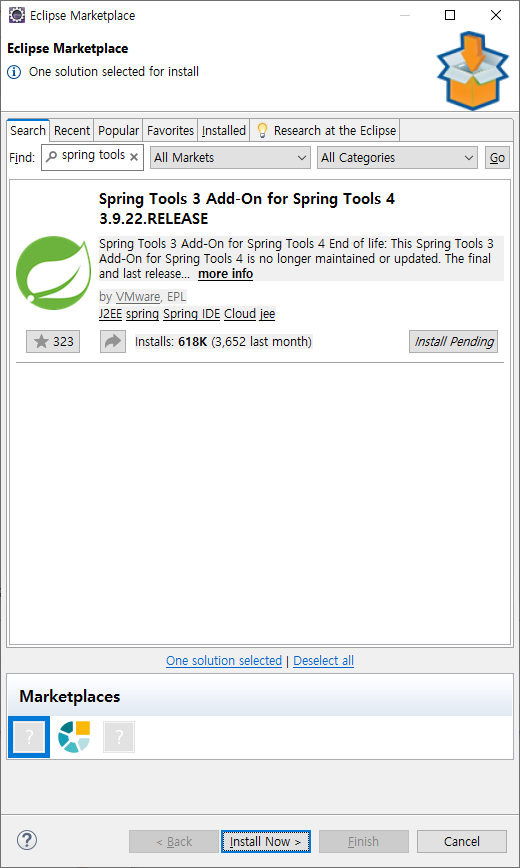
설치 후 eclipse.ini에 -vm C:\Program Files\Java\jdk1.8.0\_221\bin\javaw.exe 추가

아마 아까 설치했던 jdk 1.8버전의 jvm을 자동으로 사용되는 스크립트인 것 같다.

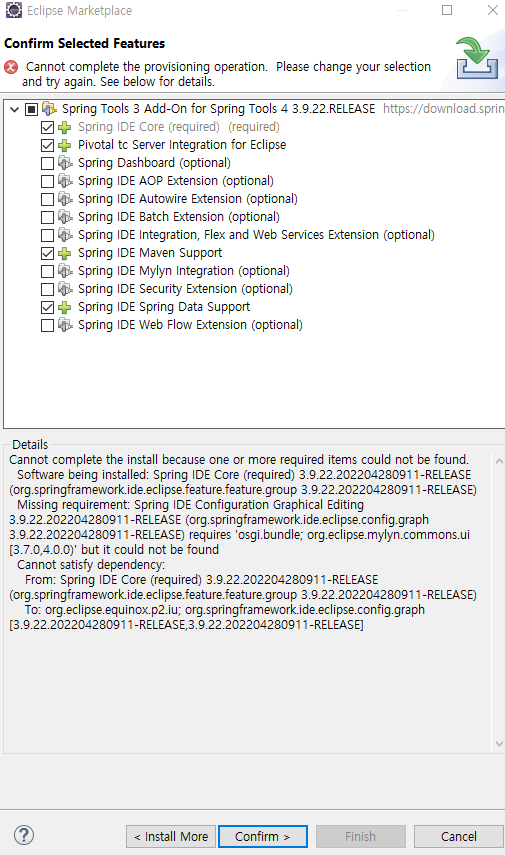


이클립스에 들어가서 preperences를 설정. Jsp, html, css 인코딩을 UTF-8로 변경

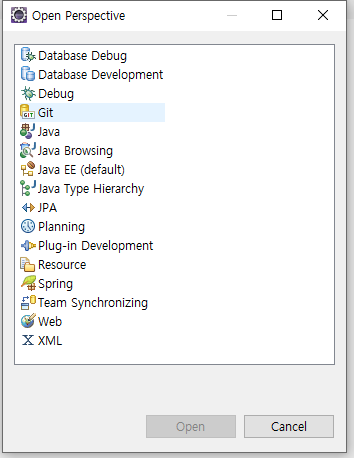
**1-3. Spring 프레임워크 설치하기**



이클립스 내의 help->eclipse market place를 통해서 다운로드 받을 수 있다.

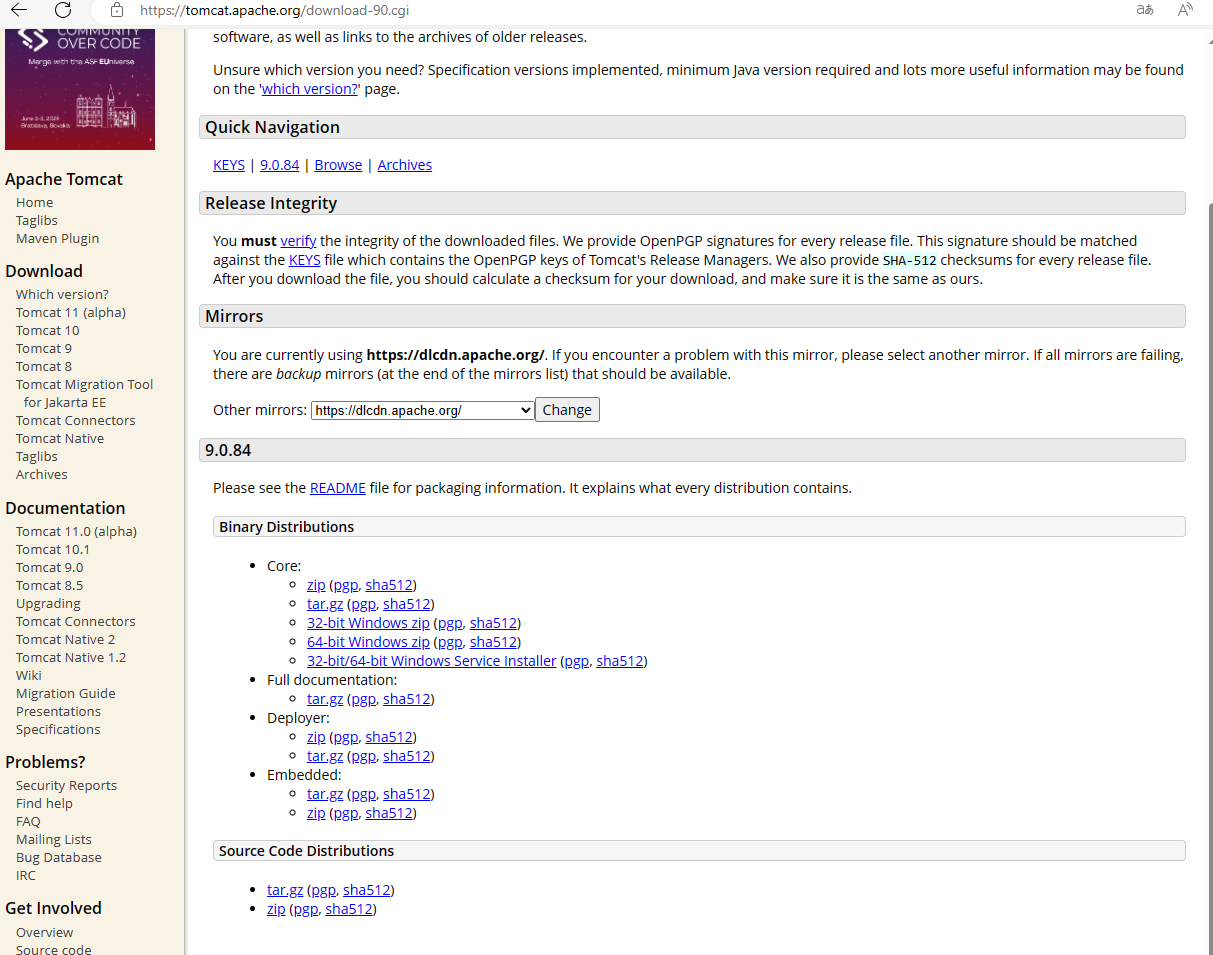


버전충돌이 일어났다. 찾아보니 sts3 버전은 21년 9월 이전 버전을 쓰는게 맞다고 한다. 그래서 20-12버전(4.18.0)다운받았다.

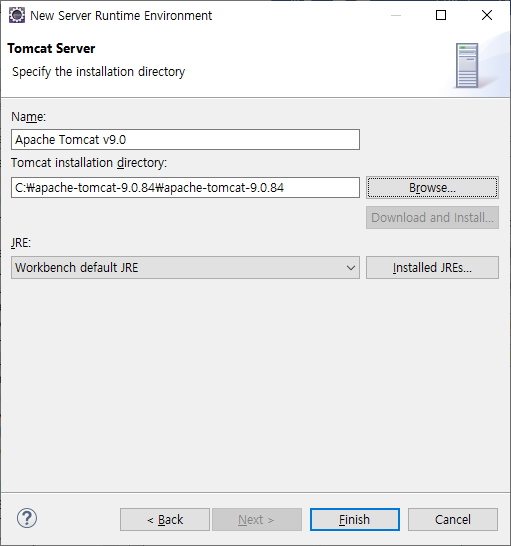


잘 설치되었다.

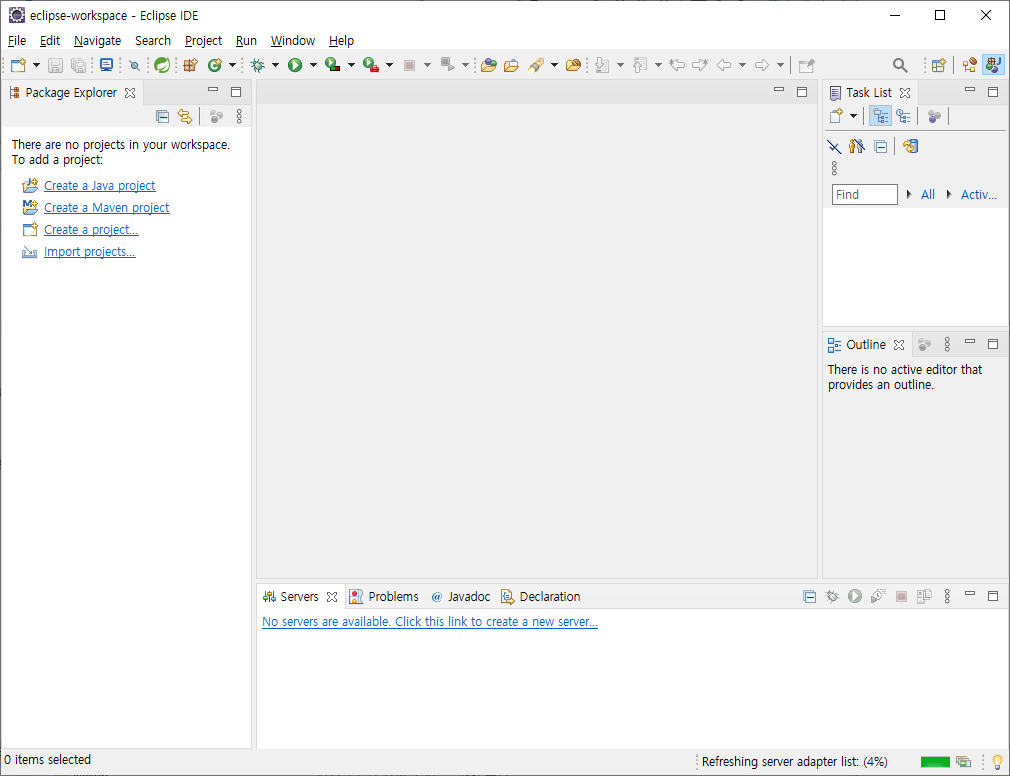
**1-4. Tomcat 설치하기**



톰캣 9버전 설치



이클립스와 동일한 폴더에 압축을 푼 후에 이클립스 설정에서 톰캣을 추가한다.

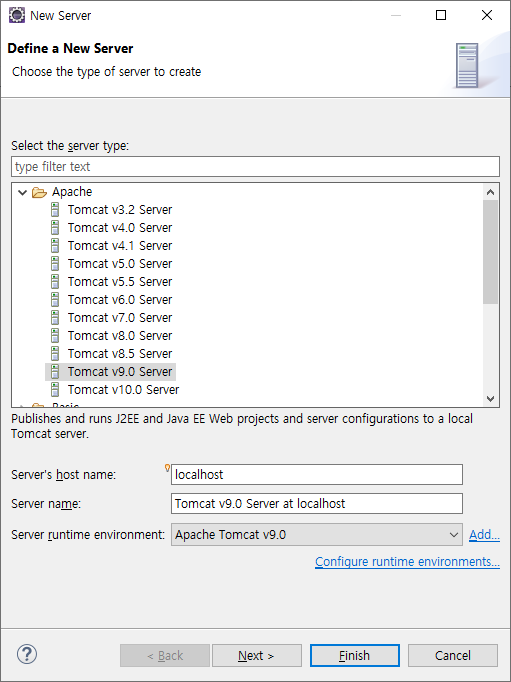


서버 상황을 보기 편하게 하기 위해 windows > show view > others > servers를 클릭하여 하단 탭에서 볼 수 있게 추가한다.

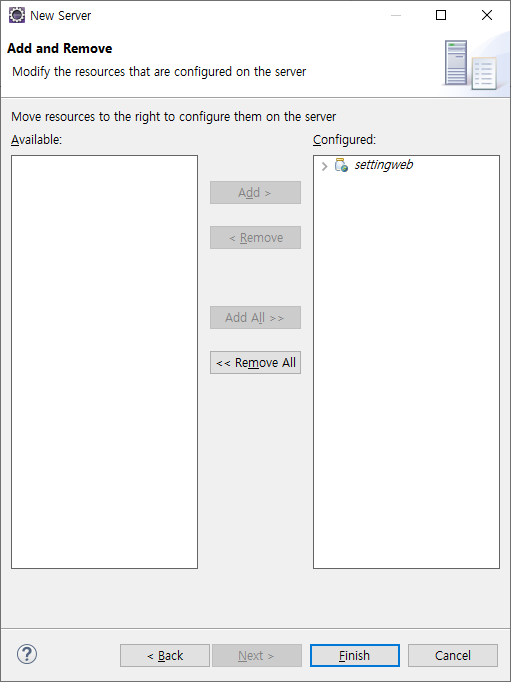
**1-5. spring helloworld**

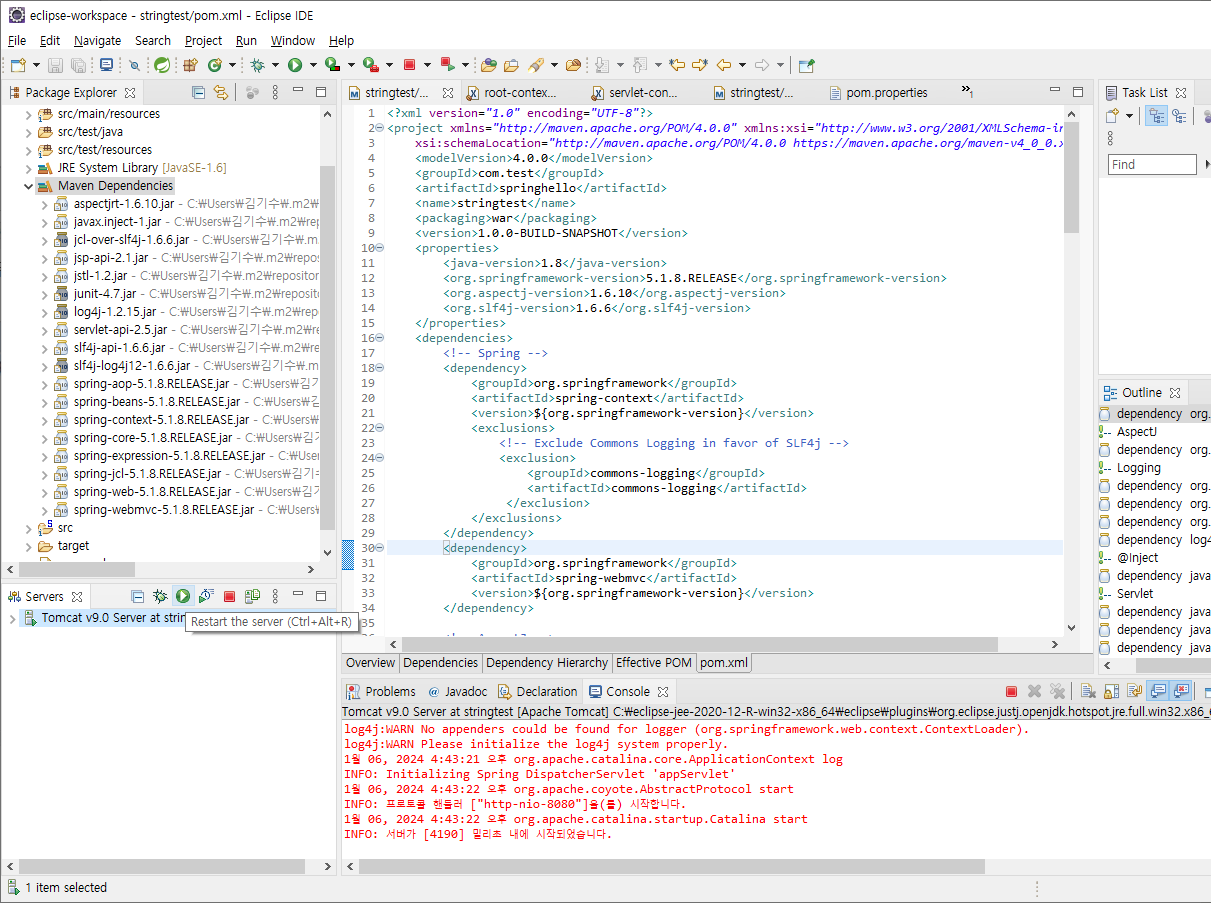


Spring과 java 버전을 5.1.8, 1.8로 변경한다.



서버에 대한 설정을 한다. Next를 눌러 추가해준다.



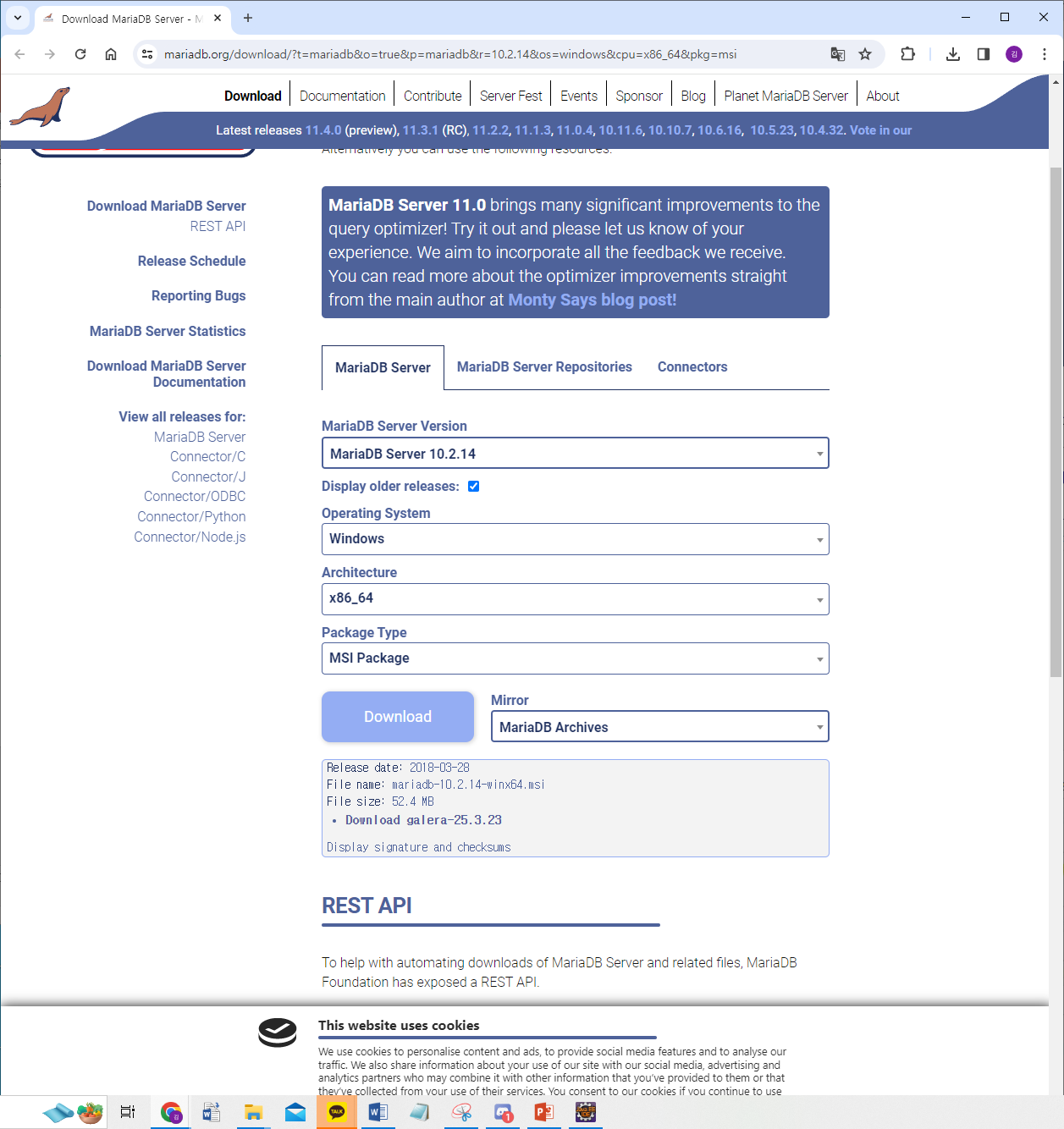


우측 하단의 서버 시작버튼을 클릭한 후에 빈 창을 띄워 localhost:8080/settingweb/ 로 접속한다.

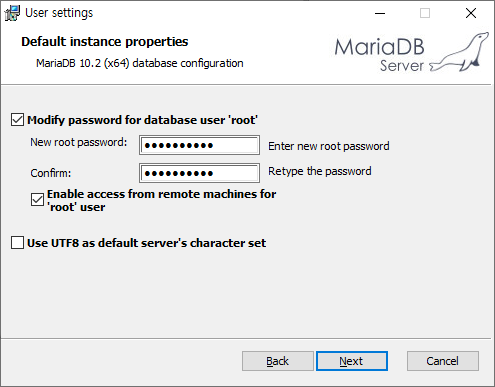


잘 접속된 것을 확인할 수 있다.

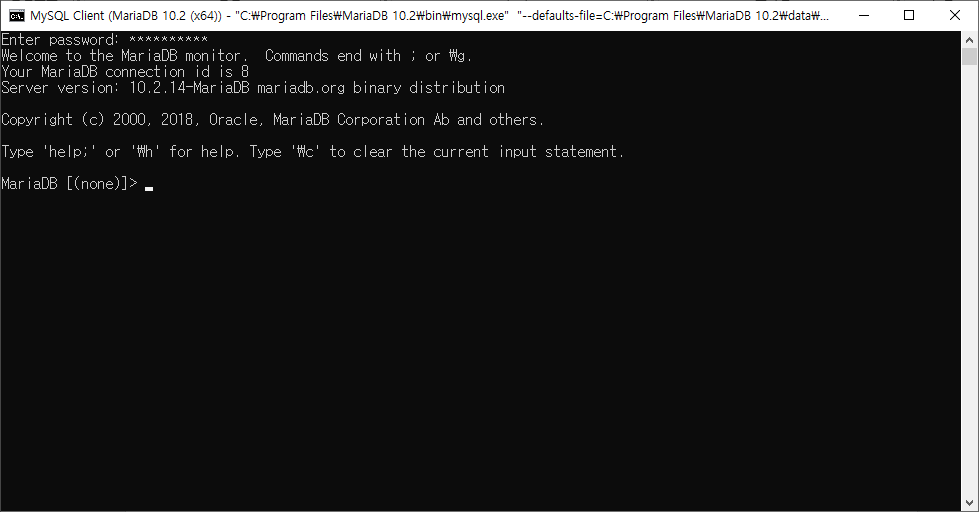
**1-6. mariadb, mySql Workbench 설치 및 샘플 DB 구축하기**



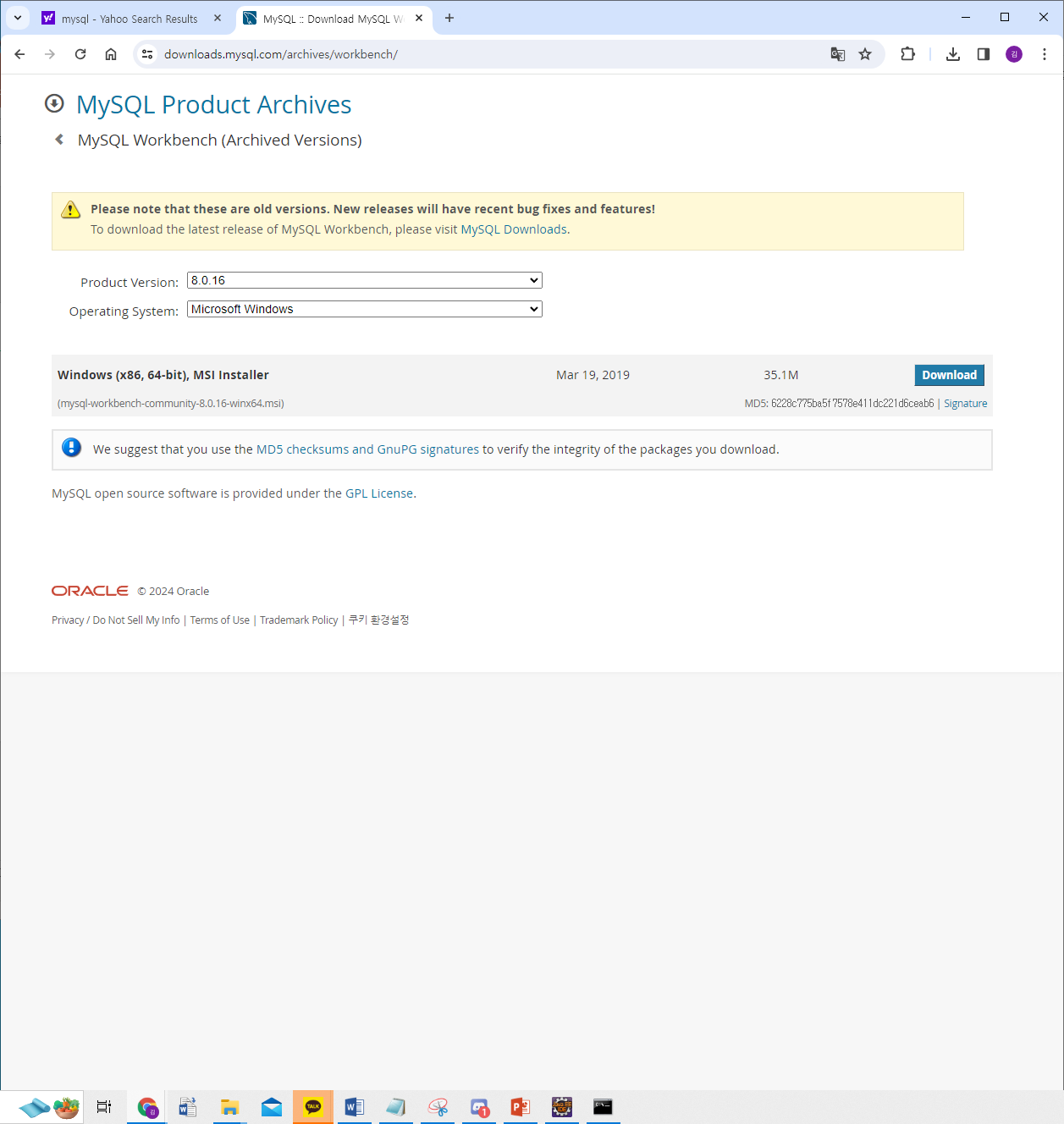
Maria DB 10.2.14 버전을 다운로드한다.



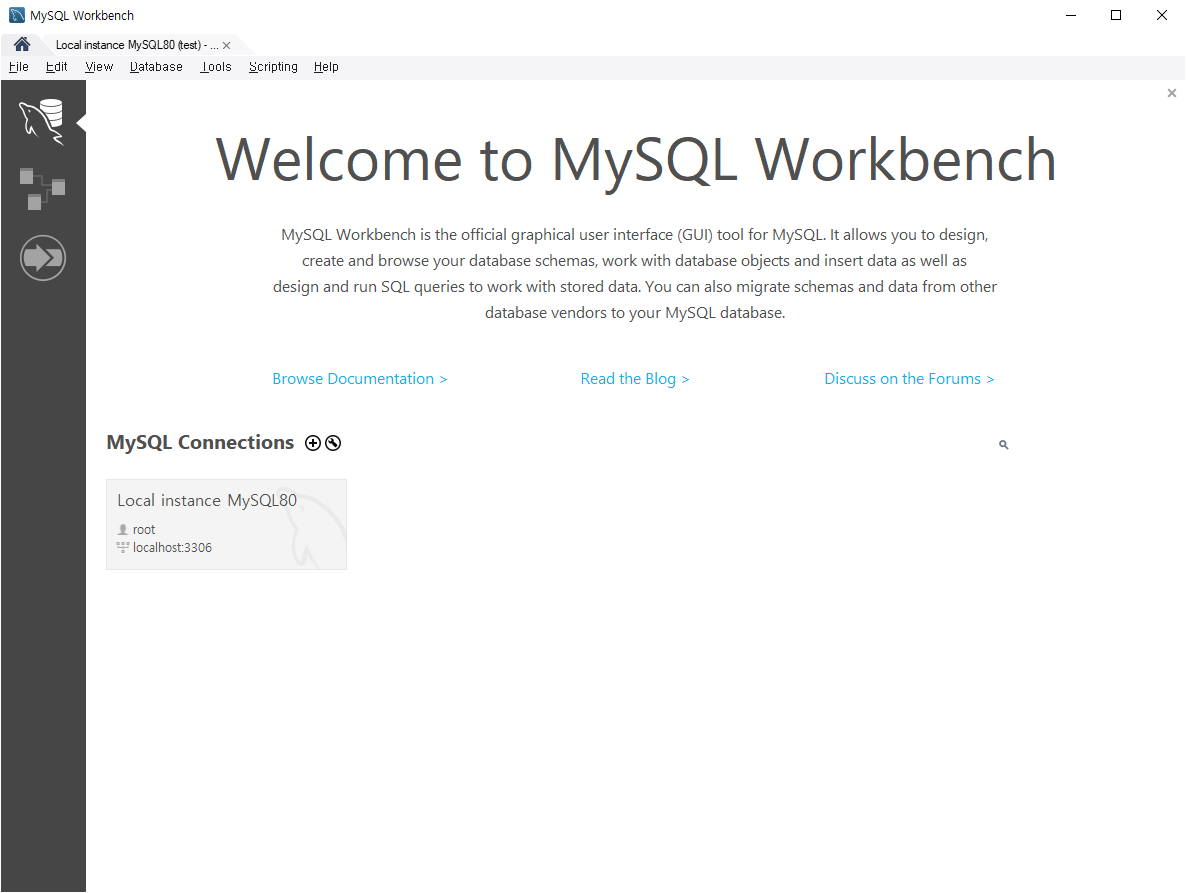
Root 계정의 비밀번호 설정과 UTF8 설정에 체크한다.



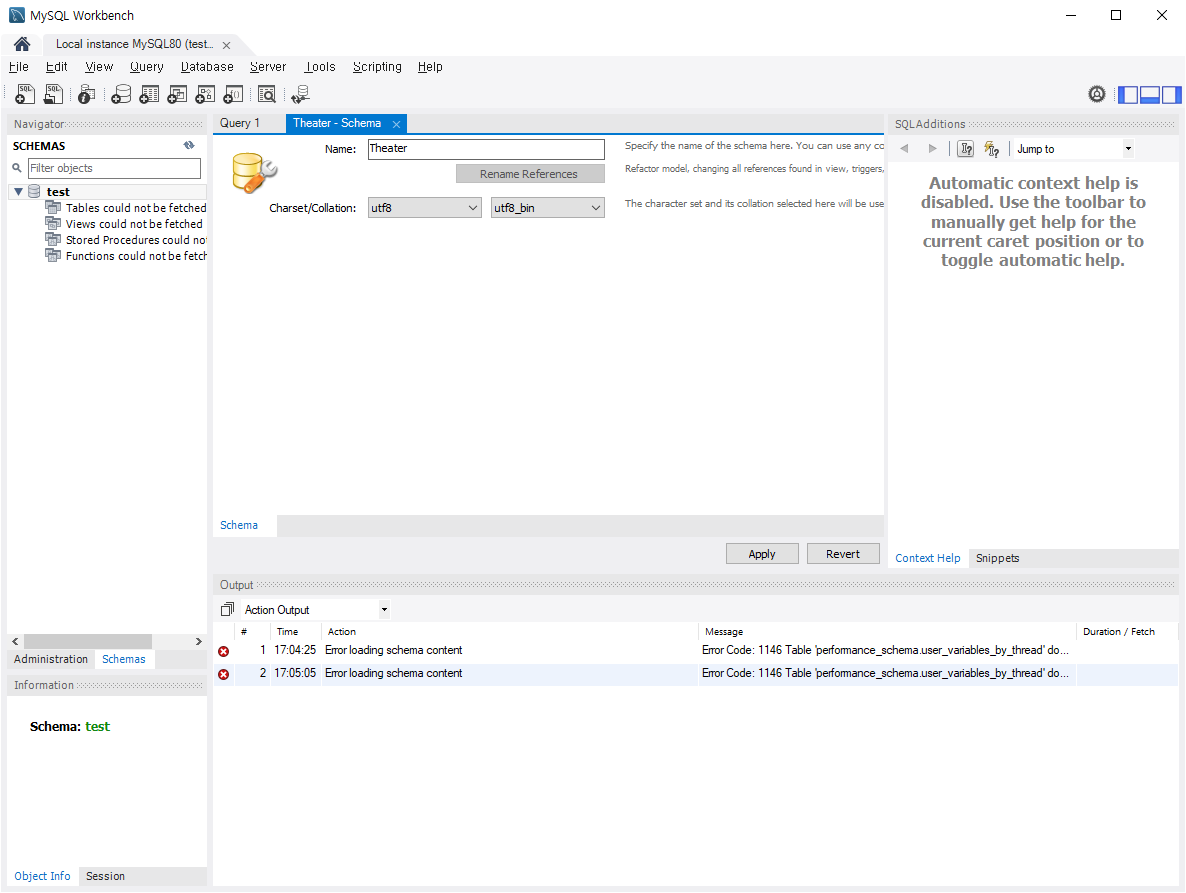
Maria DB가 잘 설치된 것을 확인했다.



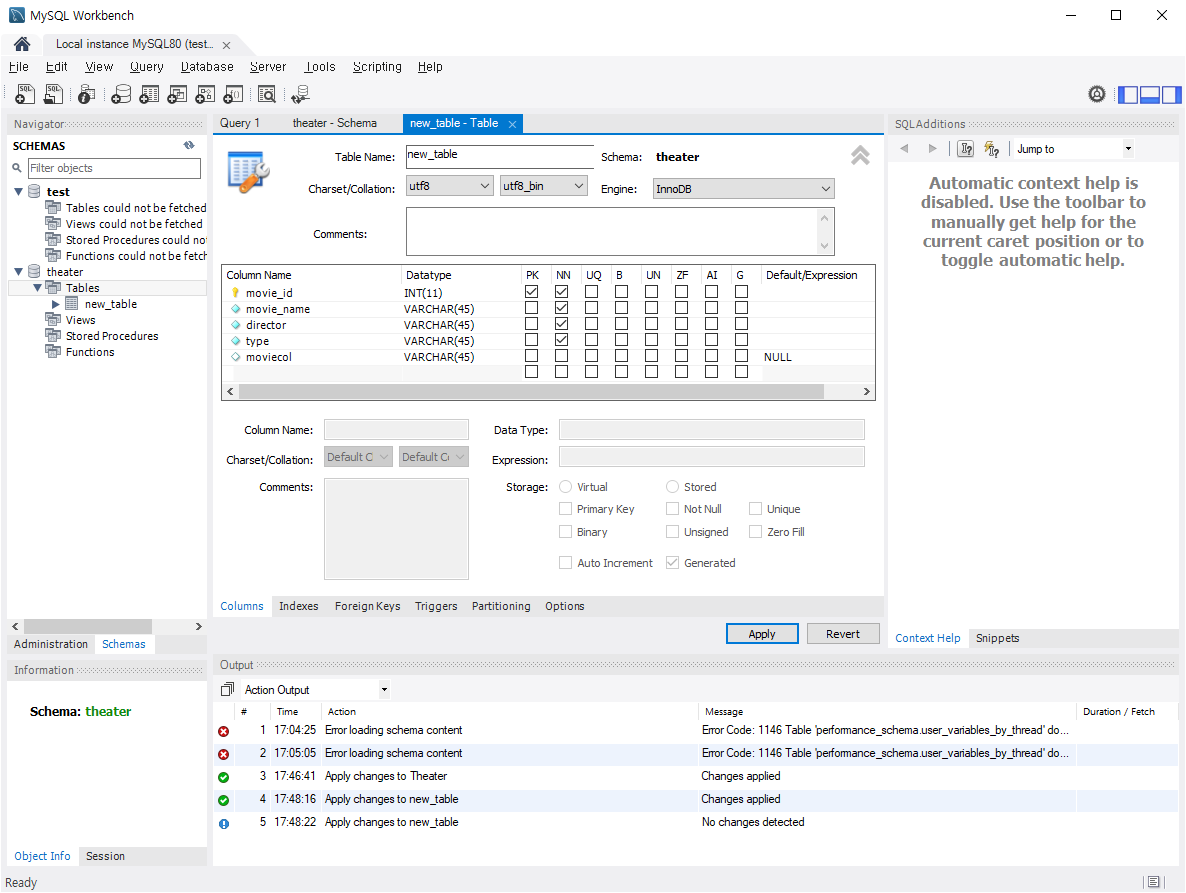
MySQL workbench 8.0.16 버전을 다운받는다.

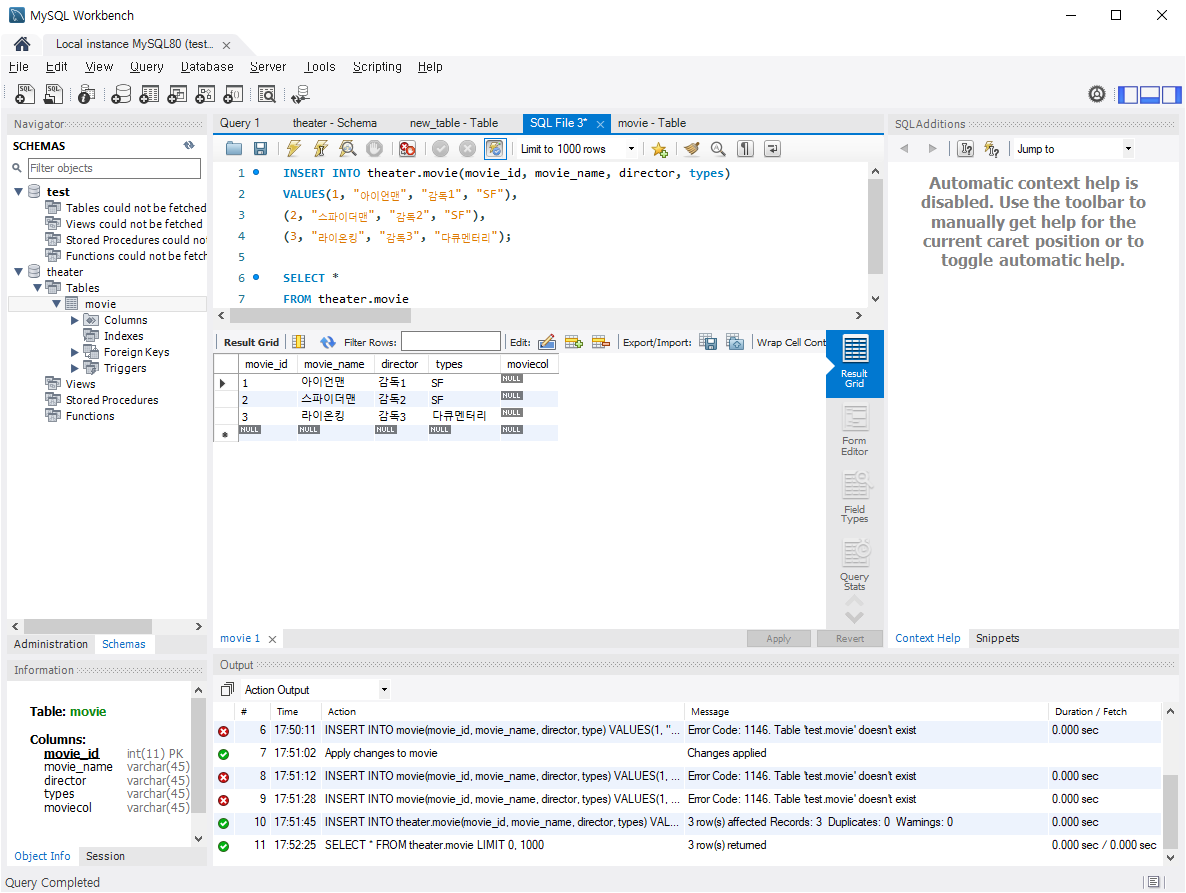


Workbench에 접속하여 local에 커넥션한다.



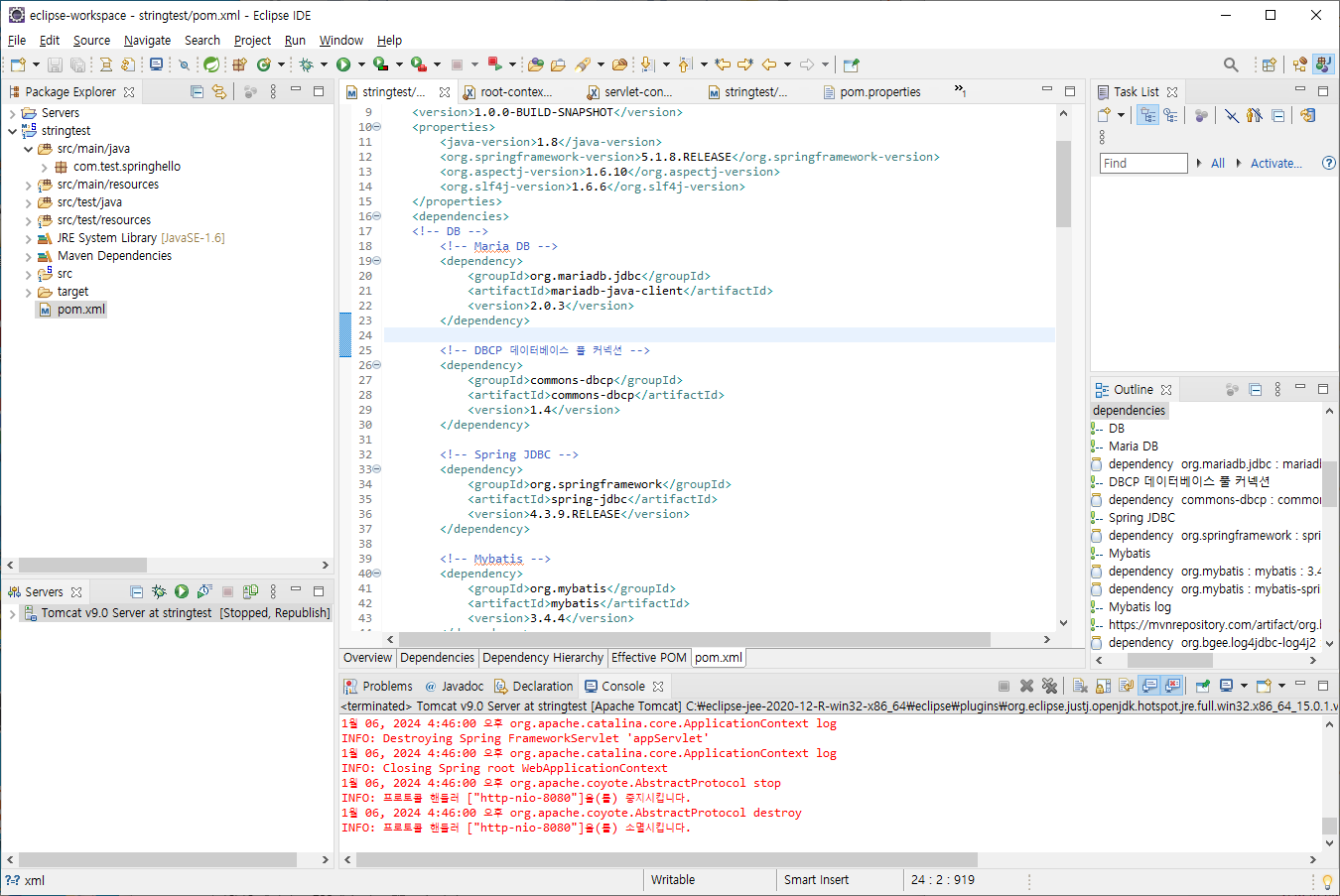
Schema를 생성 후에 테이블을 만들고 데이터를 넣는다.



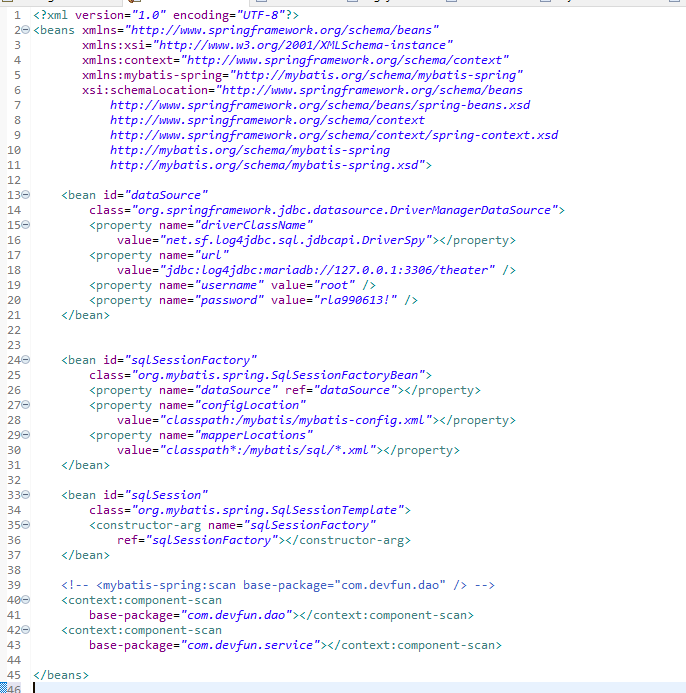


잘 조회되는 것을 확인할 수 있다.

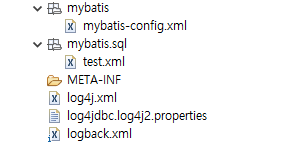
**1-7. 스프링, Mariadb, MyBatis 연동, 데이터 조회하기**



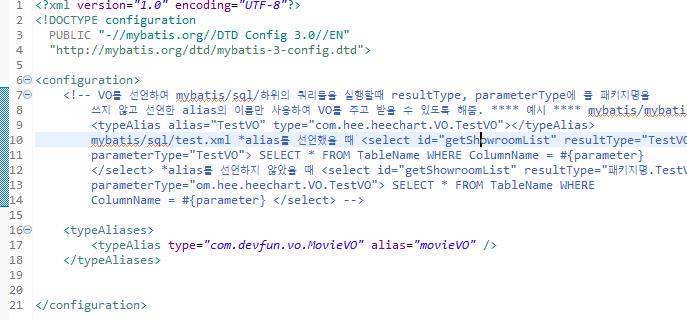
Pom.xml 파일에서 mariadb, mybatis 관련된 dependnecy를 추가한다.



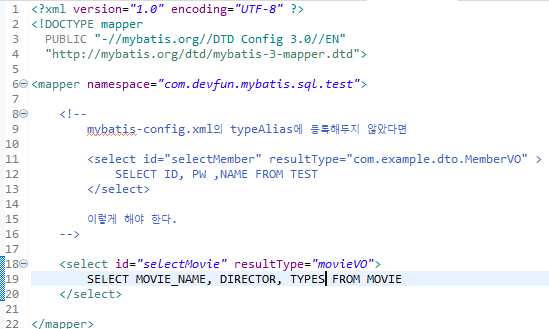
Root-context.xml 파일을 다음과 같이 수정했다. 해당 파일은 웹 이외의 부분을 기술하기 위해 사용한다.



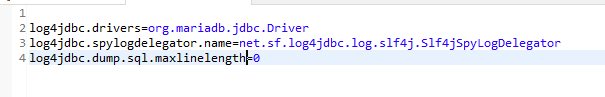
위에서 설정해둔 디렉토리 구조를 만들어 준 후에 mybatis-config.xml, test.xml, log4jdbc.log4j2.properties, logback.xml 파일을 만든다.



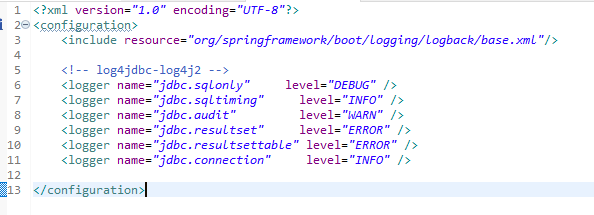
Mybatis-config.xml 파일을 위와 같이 작성한다.



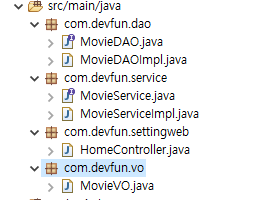
test.xml 파일을 위와 같이 작성한다.



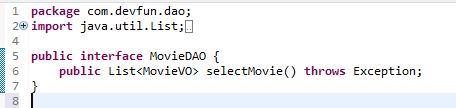
log4jdbc.log4j2.properties 파일을 위와 같이 작성한다.



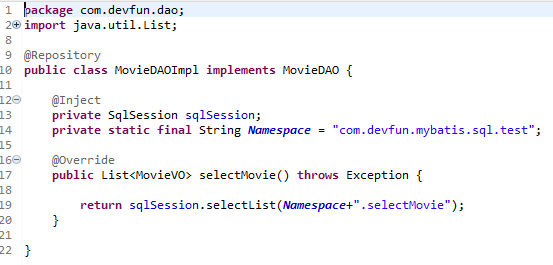
Logback.xml 파일을 위와 같이 작성한다.



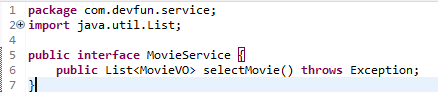
데이터를 조회하기 위하여 위와 같은 구조로 java 파일을 작성한다.



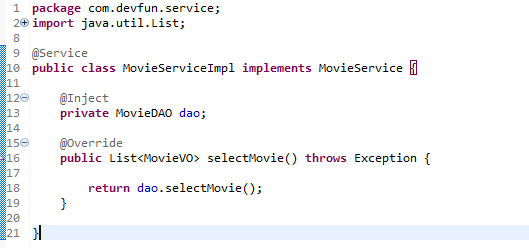
- MovieDAO.java



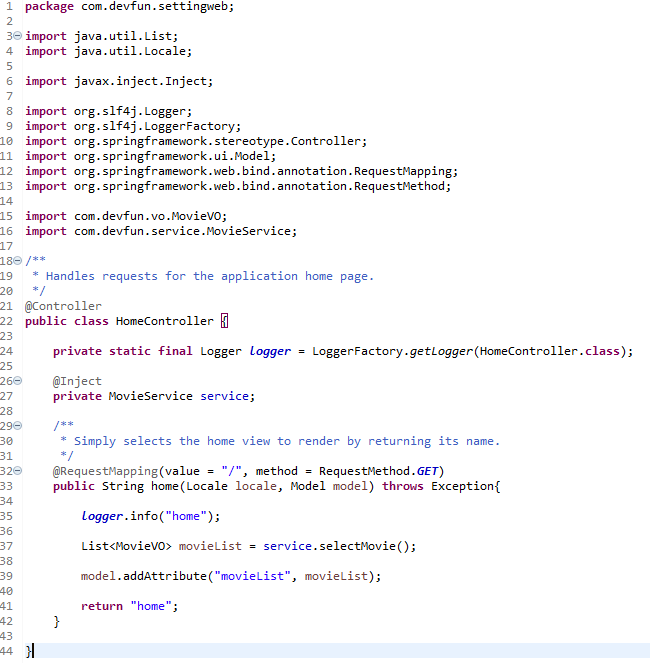
- MovieDAOImpl.java



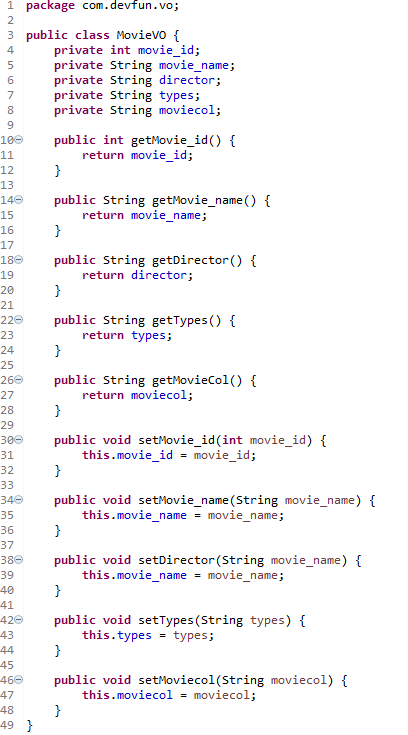
- MovieService.java



- MovieServiceImpl.java



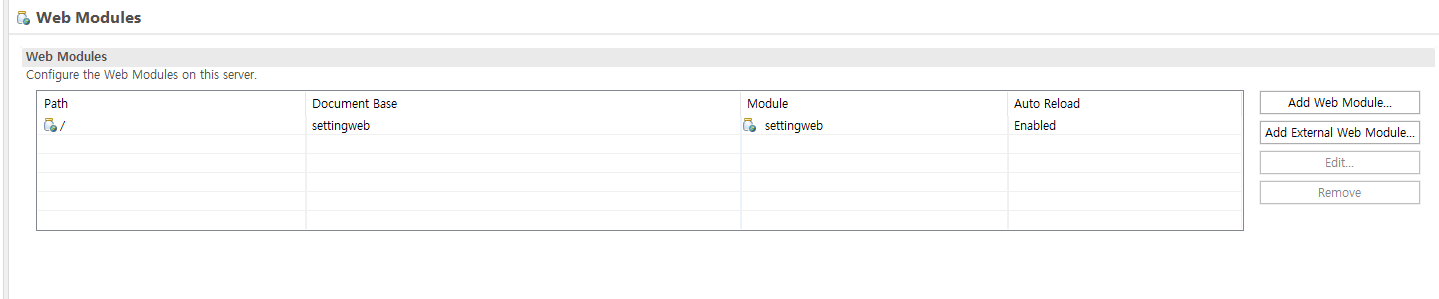
- HomeController.java



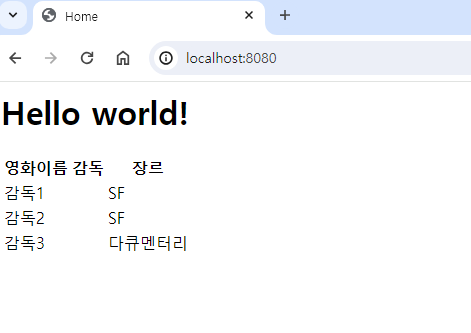
- MovieVO.java



테스트할 화면을 home.jsp 파일로 구성한다.



톰캣 설정에 들어가 /settingweb -> / 으로 변경해 localhost:8080/으로만 검색해도 화면이 출력되게 바꾸었다.



설정 후에 잘 동작하는 것을 확인했다.

**1-7. 스프링과 스프링부트의 환경 구축의 차이점**

먼저 스프링부트가 만들어진 이유에 대해서 알아볼 필요가 있었다. 스프링부트는 스프링의 dependency 버전 관리, 설정 파일(xml, properties 등) 생성, 프로젝트 설정 등 초기 설정에 대한 불편함을 해소하고 개발자가 보다 편리하게 사용할 수 있게 만드는 도구이다. 따라서 초기 설정 단계에 스프링이 스프링부트보다 힘들다는 단점이 있다. 스프링은 의존성, 설정 파일 생성, 프로젝트 설정 등을 다 설정해야되지만 스프링부트는 자동으로 해준다는 차이점이 존재한다.