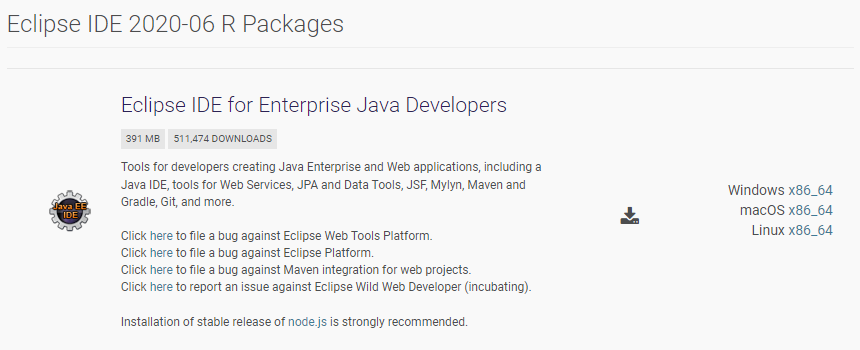
08/05 java EE / 플러그인이 설치된 버전.

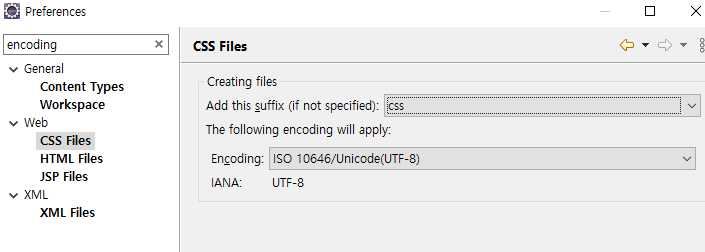


Web Service를 지원해준다.

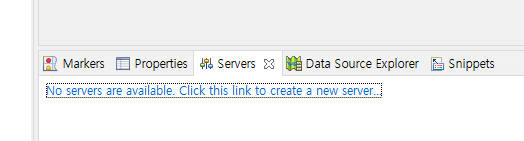
플러그인으로 설치해도 되지만 하나하나 설정해야해서 새로 받음.

모든 설정 다시 하기.

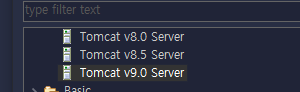
Workspace, CSS, HTML. JSP 전부다 UTF-8 변경.



Tomcat 경로 설정하기 (밑에 없으면 Window탭 → Show view → servers)

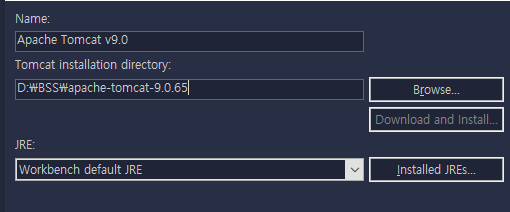


Apache → Tomcat v9.0 server



Next >

Tomcat경로 설정해주기.



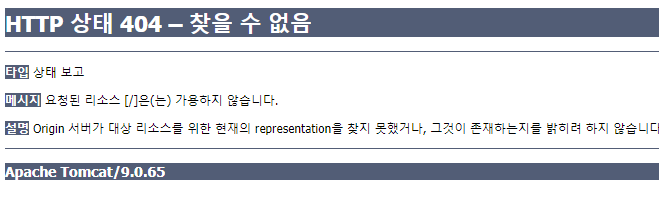
Next> 후 Finish

Catalina Home 인 실제 경로가 아니라, Catalina Base로

새로운Instance → 분신술~



Tomcat 우클릭해서 Start해주면 서버가 작동. 방화벽 허용해주기.



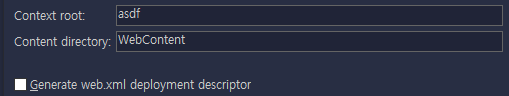
→ 이 화면이 뜨면 작동 중임.

보여줄 파일이 없어서 404에러, Apache Tomcat/9.0.65로 작동 중임!

주의할 점 : 실제 Tomcat이 켜져 있으면, 포트 번호가 겹쳐져서 실행 불가능.

Project Explorer에서

New → others → dynamic web project 생성



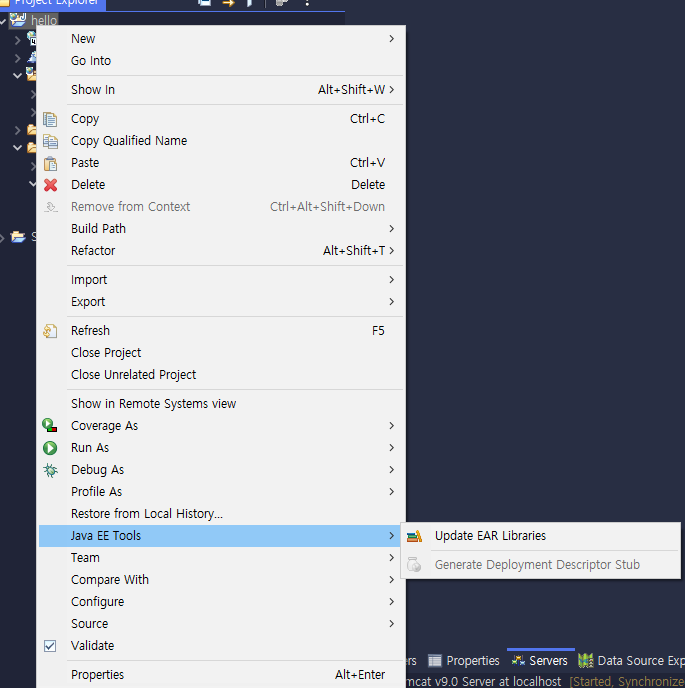
마지막에 Generate저거 체크해주기!!

중요 : Context root 기억하기

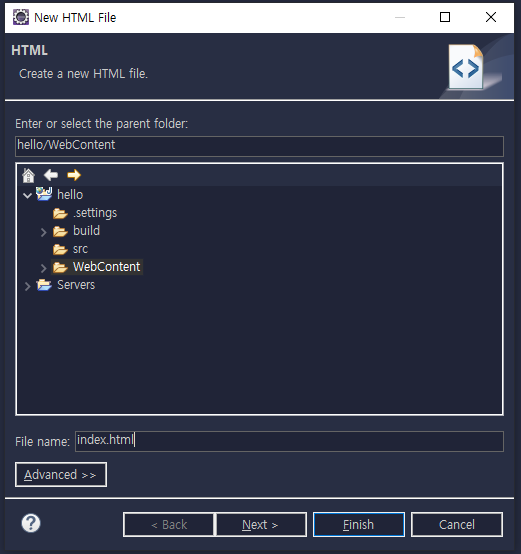


웹 구성에 필요한 구성 요소들을 담고있는 폴더들이다.

 → 마지막에 체크하면 자동으로 생성해주는 것.



혹시 생성되지 않았다면, 프로젝트 우클릭 → Java EE Tools → Generate Deployment Descriptor Stub로 생성할 수 있다.



WebContent 폴더에 new로 html file을 생성해준다.

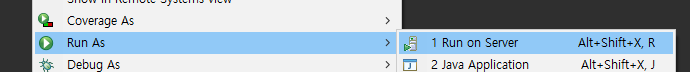
정적 자원들은 WebContent 폴더에 넣어서 접근이 가능하다.

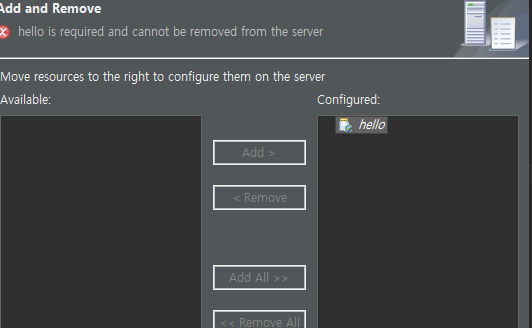


사용할 HTML 버전 →html5

프로젝트 우클릭

→





→ 우측의 파일이 실행될 파일.

Hello2 프로젝트의

Context root → baek으로 설정되어있음.



배포하기





→ 실제 Tomcat폴더에 위치시켰다.

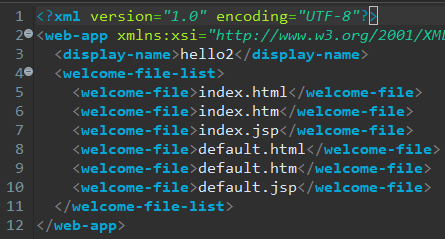
→ 실제 Tomcat이 구동할 준비가 되었음.

Tomcat을 구동시키면, hello.war파일안의 내용을 자동으로 hello폴더를 생성해준다.

→ Tomcat구동하던 방식으로 웹에서 확인 가능!

Tomcat 에서 기본적으로 context root만 입력해도, 기본값으로 index.html을 찾아준다.

→ WEB-INF → web.xml 내용



Welcome-file로 연결할 기본값이 설정되어있다.

없는 경로에 대해서 tomcat이 연결을 해줌.

--------------------------------------------------------------------

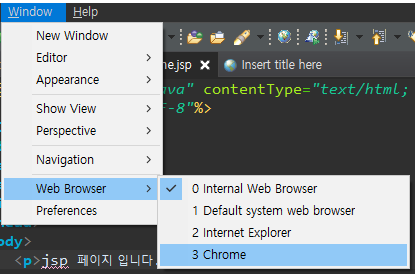
JSP

동일하게 html5버전으로 생성해준다.



→ page directive 라고 부른다.

JSP를 실행하면 html과 똑같이 나타난다. → 결국 html 문서이다.



→ 서버를 볼 웹 브라우저 설정하기

흐름 ?

사용자가 Tomcat에게 요청하면

Tomcat이 요청한 정보에 대한 요청 객체를 만들어 준다.

→ 그때그때마다 동적인 처리를 하기 위함이다.

요청에 대한 정보는 request.로 접근이 가능하다.