**DEV #6 – 2020.04.27**

>> Java Project를 만들어 서블릿을 생성하시오

1. 패키지생성 / CLASS생성 – class는 HttpServlet 을 상속한다.
2. HttpServlet 객체 사용을 위한 톰캣라이브러리 Java Build Path에 등록
3. doGet method를 override하여 서블릿 생성
4. Servlet 3.0 이후 적용된 Annotation 방식으로 url-mapping (@WebServlet(“/first”))

>> 프로젝트 루트에 웹이 실행될 webroot를 생성하고 톰캣서버에서 서블릿을 기동하시오.

1. 프로젝트 루트에 webroot폴더 생성 후 톰캣서버에 External Module로 등록함.
2. 서블릿 규약에 따라 webroot폴더를 구성. (WEB-INF, lib, classes)
3. 톰캣서버 start -> “localhost:8080/서블릿명” 접속확인
4. webroot에 index.html파일 추가

>> embed tomcat으로 웹서버를 기동하시오. (제어권 일부 회수)

1. embed tomcat을 다운로드하여 “.jar”라이브러리를 WEB-INF/lib 에 복사
2. Java Build Path에 emdbed tomcat 라이브러리 추가 (Add Jar)
3. 기존 톰캣서버 라이브러리는 Java Build Path에게 제거
4. class를 추가하고 main() method 구현
5. main() method에서 embed tomcat 구현 ( Tomcat tomcat = new Tomcat() )
6. main() method 실행 -> 웹서버 구동 확인

>> Maven Project로 변경

1. maven project로 convert ( config -> convert to maven project)
2. pom.xml 수정 / index.html 삭제 / index.jsp 추가 / jsp JSTL taglib 추가
3. embed tomcat 라이브러리 dependency pom.xml에 추가 (Maven->Update Project)
4. jsp JSTL 라이브러리 dependency pom.xml에 추가 (Maven->Update Project)
5. 기존 embed tomcat 라이브러리 삭제 (WEB-INF/lib/\*\*) / Java Build Path에서 embed tomcat 라이브러리 제거
6. Run As ) -> Maven Clean && Maven Install

>> Maven Install로 만들어지 war파일을 외부톰캣에 배포하시오.

1. 외부톰캣(apache-tomcat-8.5.54) /bin 디렉토리에 setenv.bat 파일 생성
2. setenv.bat 파일에 JAVA\_HOME 환경변수 지정
3. /bin 디렉토리의 startup.bat 실행 – 외부톰갯 구동 ( localhost:8080 접속확인)
4. Java Project target폴더의 “.war”파일명을 적당히 변경 (버전정보제거)
5. war파일을 외부톰캣 /webapps 에 복사 (Deploy = 배포)
6. { [파일명].war = [context명].war } “localhost:8080/[context명]” 으로 배포여부 확인
7. “localhost/[context명]/[서블릿]” 으로 서블릿 구동여부 확인