



개발환경

JAVA

windows power shell 사용하면 리눅스 명령어 사용 가능!!!

→ vi 이용하기 위해서 vim -v 명령어 사용

BEAN : 자바 객체

자바 환경변수 쓰는 이유 : 경로를 찾아가기가 너무 길기 때문에

→ JDK 버전이 여러 개면 사용 할 버전을 path에서 제일 위로 올리기

javac: 컴파일 명령어

```
C:\Users\deare>javac
Usage: javac <options> <source files>
```

java: 자바 실행 명령어

```
C:\Users\deare>java
사용법: java [-options] class [args...]
        (클래스 실행)
또는   java [-options] -jar jarfile [args...]
        (jar 파일 실행)
```

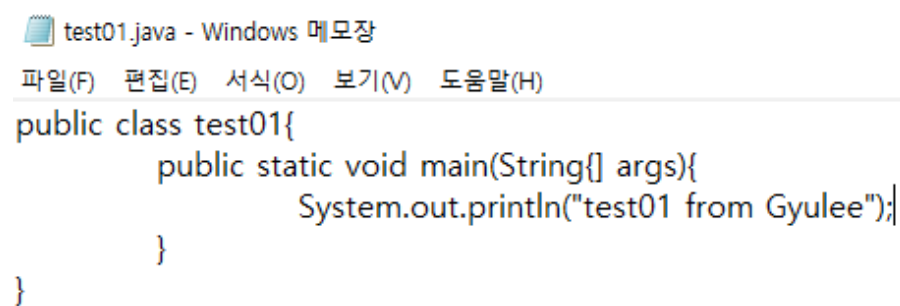
jstack: 스레드 내역을 보여줌(실행되는지, 잠 자고 있는지)

jmap: 자바오브젝트 내역이 저장되는 곳을 보여주는 명령어

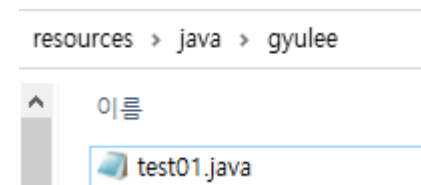
jvisualvm: JAVA Visual VM 실행

<자바 프로그램 작성>

1. 메모장 or vi편집기에서 코드 작성



```
test01.java - Windows 메모장
파일(F)  편집(E)  서식(O)  보기(V)  도움말(H)
public class test01{
    public static void main(String[] args){
        System.out.println('test01 from Gyulee');
    }
}
```



```
resources > java > gyulee
이름
test01.java
```

2. 컴파일 (.class로 만들어주는 작업)

```

C:\Users\deare>cd C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>dir
C 드라이브의 볼륨: Windows
볼륨 일련 번호: 4000-489D

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee 디렉터리

2023-02-02 오후 04:42 <DIR>      .
2023-02-02 오후 04:42 <DIR>      ..
2023-02-02 오후 04:45              114 test01.java
                1개 파일              114 바이트
                2개 디렉터리  408,721,887,232 바이트 남음

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>javac test01.java

```

```

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>dir
C 드라이브의 볼륨: Windows
볼륨 일련 번호: 4000-489D

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee 디렉터리

2023-02-02 오후 04:45 <DIR>      .
2023-02-02 오후 04:45 <DIR>      ..
2023-02-02 오후 04:45              424 test01.class
2023-02-02 오후 04:45              114 test01.java
                2개 파일              538 바이트
                2개 디렉터리  408,724,901,888 바이트 남음

```

3. 윈도우 자바 클래스 실행 java -cp . Hello2 (-cp: classpath / '.': 현재경로)

```

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>java -cp . test01
test01 from Gyulee

```

4. .java가 있는 위치 말고 다른 위치에서 컴파일 하는 법

javac -d ./classes ./src/Hello2.java : 클래스 파일을 원하는 위치에 (./classes)

→ 옮기고 실행 java -cp ./classes Hello2

```

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java>javac -d ./classes ./gyulee/test01.java

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java>cd ./classes

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\classes>dir
C 드라이브의 볼륨: Windows
볼륨 일련 번호: 4000-489D

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\classes 디렉터리

2023-02-02 오후 04:53 <DIR>      .
2023-02-02 오후 04:53 <DIR>      ..
2023-02-02 오전 10:34 <DIR>      com
2023-02-02 오전 10:57              403 Hello2.class
2023-02-02 오후 04:53              424 test01.class
                2개 파일              827 바이트
                3개 디렉터리  408,725,016,576 바이트 남음

```

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\classes>java -cp . test01
test01 from Gyulee
```

5. 코드에 한국어가 있을 때

```
test02.java - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
public class test02{
    public static void main(String[] args){
        System.out.println("안녕하세요");
    }
}
```

javac -encoding utf-8 -d classes .\src\com\example\main\PropertiesMain.java: 한국어로 인코딩

javac -encoding [인코딩 할 언어] -d [클래스 파일 저장 할 폴더] [자바 파일이 들어있는 경로]

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java>javac -encoding utf-8 -d gyulee ./gyulee/test02.java
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java>cd gyulee
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>dir
C 드라이브의 볼륨: Windows
볼륨 일련 번호: 4000-489D

C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee 디렉터리

2023-02-02 오후 04:58 <DIR> .
2023-02-02 오후 04:58 <DIR> ..
2023-02-02 오후 04:45 424 test01.class
2023-02-02 오후 04:45 114 test01.java
2023-02-02 오후 04:58 421 test02.class
2023-02-02 오후 04:56 111 test02.java
                4개 파일             1,070 바이트
                2개 디렉터리 408,717,996,032 바이트 남음
```

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee>java -cp . test02
안녕하세요
```

6. java -cp ./classes -Dmy.msg=hello com.example.main.PropertiesMain 1 2 3 4 5

java -cp [클래스 파일이 들어있는 폴더] -D[시스템 속성 정의] [클래스 파일이 들어있는 패키지 경로]

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java>java -cp ./classes -Dmy.msg=hello com.example.main.PropertiesMain 1 2
args
1
2
```

System.getenv(); → 시스템 환경변수를 가져옴

```
env
-----
USERDOMAIN_ROAMINGPROFILE = LAPTOP-NQDHOPU9
LOCALAPPDATA = C:\Users\deare\AppData\Local
PROCESSOR_LEVEL = 6
```

System.lineSeparator() → 엔터

```
properties
-----
-- listing properties --
java.runtime.name=Java(TM) SE Runtime Environment
sun.boot.library.path=C:\Program Files\Java\jdk1.8.0_351\jr...
```

-Dmy.msg=hello → 기준위치를 정해서 문자를 넣어줌

```
user defined msg property
-----
my.msg=hello
```

MAVEN

- 소스코드 파일을 컴퓨터에서 실행할 수 있는 독립 소프트웨어 가공물로 변환하는 과정 또는 그에 대한 결과물을 만들기 위한 도구

메이븐 환경 설정

- MAVEN_HOME=C:\mocomsys\apache-maven-3.2.5
- PATH=%PATH%;%MAVEN_HOME%\bin


메이븐 라이프사이클(미리 정해진 빌드 순서)

단계	설명
compile	소스 코드를 컴파일해서 클래스 출력 폴더에 클래스를 생성
test	테스트를 실행
package	컴파일 된 코드와 자원 파일들을 jar, war와 같은 배포 형식으로 패키징
install	로컬 리포지토리에 패키지를 복사

<MAVEN 공부할 때 사이트>

Maven Getting Started Guide

At first glance Maven can appear to be many things, but in a nutshell Maven is an attempt to apply patterns to a project's build infrastructure in order to promote comprehension and productivity by providing a clear path in the use of best practices.

 <https://maven.apache.org/guides/getting-started/index.html>

- project object model(POM)
 - pom.xml → 메이븐을 이용하는 프로젝트의 root에 존재하는 xml 파일
 - 모든 설정, 의존성 등을 알 수 있음

```
<groupId>com.example</groupId>
<artifactId>example</artifactId>
<version>1.0</version>
```

필수로 들어가는 내용

- 프로젝트를 관리해주는 툴
 - 라이브러리를 관리해주는게 가장 큰 일 dependencies가 가장 중요

<https://mvnrepository.com/>

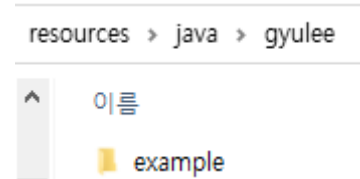
메이븐 라이브러리 사이트

<메이븐 프로그램 작성>

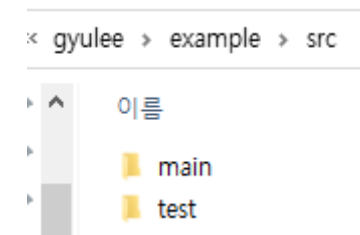
1. 프로젝트 생성

`mvn archetype:generate -DarchetypeArtifactId=maven-archetype-quickstart`

```
mvn archetype:generate -DgroupId=com.example -DartifactId=example -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4
```



→ 프로젝트 빌드 → src 밑에 메인과 테스트 폴더가 생김 → 패키지는 같음

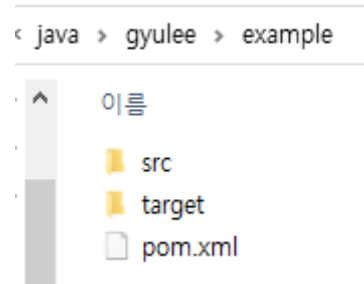


2. 컴파일 / 인스톨

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee\example>mvn compile
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building example 1.0
[INFO] -----
[INFO] --- maven-resources-plugin:3.0.2:resources (default-resources) @ example ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\main\resources
[INFO] --- maven-compiler-plugin:3.8.0:compile (default-compile) @ example ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee\example\target\classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.431 s
[INFO] Finished at: 2023-02-03T09:47:46+09:00
[INFO] Final Memory: 13M/241M
[INFO] -----
```

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee\example>mvn install
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building example 1.0
[INFO] -----
```

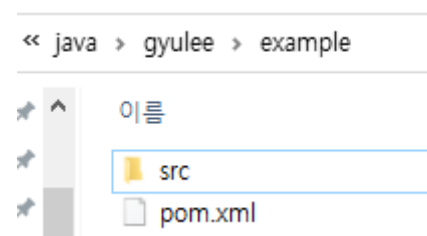
→ target이라는 폴더가 생김



3. 클린

```
C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee\example>mvn clean
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building example 1.0
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ example ---
[INFO] Deleting C:\Users\deare\OneDrive\바탕 화면\goodluck\resources\java\gyulee\example\target
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.207 s
[INFO] Finished at: 2023-02-03T09:54:29+09:00
[INFO] Final Memory: 8M/304M
[INFO] -----
```

→ target 폴더 사라짐



4. 다른 사람 경로 target 폴더에서 프로젝트 실행

- 1) mvn package → target 폴더 생김
- 2) target 폴더로 들어가서 → java 명령어로 프로젝트 실행 → 동일한 결과

```
java -cp ./classes -D"my.msg=hello" com.example.main.PropertiesMain 1 2 3 4 5
```

→ 윈도우에서는 '.'을 확장자로 인식하기 때문에 ""로 묶어서 시스템 속성 사용하기

<도커로 리눅스 사용할 수 있도록 maven 실행 (최초 생성 할 때)>

-v : 볼륨(디스크)

```
C:\WINDOWS\system32>docker run -it --name maven-env -v C:\Github\goodluck\resources\gyulee\example:/usr/src/mymaven -v
"C:\Users\dearew.m2":/root/.m2 -w /usr/src/mymaven maven:3.3-jdk-8 bash
Unable to find image 'maven:3.3-jdk-8' locally
3.3-jdk-8: Pulling from library/maven
6d827a3ef358: Pull complete
2726297beaf1: Pull complete
7d27bd3d7fec: Pull complete
e61641c845ed: Pull complete
cce4cca5b76b: Pull complete
6826227500b0: Pull complete
c03b117ffd91: Extracting [=====>] 76.87MB/130.4MB
821a1547b435: Download complete
2bd47f6b1b42: Download complete
e4cf3e9f705c: Download complete
3733107c5c01: Download complete
```

리눅스 모드로 변경 완료

```
root@e3e18baf1c76:/usr/src/mymaven#
```

- 도커에서 컨테이너를 내리는 순간(삭제) 작성한 파일들이 날라가는데 로컬에서 작성한 파일들은 날라가지 않는다.

<한 개의 파일만 메이븐 실행>




```
mvn clean package -pl mint-common -Dmaven.test.skip=true
```

-pl : 지정해서

-D : 테스트 파일 스킵

1) 도커 메이븐 실행

```
PS C:\> docker start maven-env
maven-env
```




<input type="checkbox"/>	 maven-env e3e18baf1c76	maven:3.3-jdk-8	Running	1 minute ago		
--------------------------	--	---------------------------------	---------	--------------	---	---

2) 도커 메이븐 bash로 접속

```
PS C:\> docker exec -it maven-env bash
root@e3e18baf1c76:/usr/src/mymaven#
```

3) 도커 메이븐 스탑

```
root@e3e18baf1c76:/usr/src/mymaven# exit
exit
PS C:\> docker stop maven-env
maven-env
PS C:\>
```

<input type="checkbox"/>	 maven-env e3e18baf1c76	maven:3.3-jdk-8	Exited			
--------------------------	--	---------------------------------	--------	--	---	---

프로젝트(한글 계산기)

jar파일 생성(라이브러리 프로젝트에서 메이븐 install)

```
C:\mocomsys\workspace>cd CalcLib
C:\mocomsys\workspace\CalcLib>mvn clean install
```

→ pom.xml 파일에 repository 설정 안해도 됨

```
[INFO] Installing C:\mocomsys\workspace\CalcLib\pom.xml to C:\mocomsys\apache-maven-3.2.5\repository\com\exercises\calc\CalcLib\1.0-SNAPSHOT\CalcLib-1.0-SNAPSHOT.pom
```

→ 이 위치에 jar 파일 설치가 된다는 것을 알려줌

모든 프로젝트를 포함하는 폴더의 pom.xml에 포함하는 모듈도 넣어주기

```
<modules>
  <module>calclibrary</module>
  <module>calmain</module>
</modules>
```

main 프로젝트 pom.xml에 dependency 설정해주기

```

    <dependency>
      <groupId>com.main</groupId>
      <artifactId>calclibrary</artifactId>
      <version>0.0.1-SNAPSHOT</version>
    </dependency>
```

<jar파일 포함해서 파일 실행시키기>

java -cp "[jar파일 경로]"; [class파일 경로] [파라미터]

```
C:\mocomsys\workspace\korea-calculator\target\classes>java -cp "C:\mocomsys\workspace\korea-calculator\lib\com.exercises.calculator\korea-calculator-0.0.1-SNAPSHOT\korea-calculator-0.0.1-SNAPSHOT.jar"; com.exercises.calculator.korea_calculator.korea_calculator 이백 + 오 + 십
이백 + 오 + 십 = 이백일십오
```

C:\mocomsys\workspace\korea-calculator\target\classes>

java -cp "C:\mocomsys\workspace\korea-calculator\lib\com.exercises.calculator\korea-calculator\0.0.1-SNAPSHOT\korea-calculator-0.0.1-SNAPSHOT.jar"; com.exercises.calculator.korea_calculator.korea_calculator 이백 + 오 + 십

이백 + 오 + 십 = 이백일십오

or

C:\mocomsys\workspace\korea-calculator\target>

**java -cp "C:\mocomsys\workspace\korea-calculator\lib\korea-calculator.jar";.classes
com.exercises.calculator.korea_calculator.korea_calculator 백 + 오십 + 구**

백 + 오십 + 구 = 일백오십구

or

```
calculators.bat (C:\mocomsys\workspace\calculators) - VIM
set LIBS=C:\mocomsys\workspace\calculators\calclibrary\target\calclibrary-0.0.1-SNAPSHOT.jar
set LIBS=%LIBS%;C:\mocomsys\workspace\calculators\calmain\target\calmain-0.0.1-SNAPSHOT.jar;.\classes
java -cp %LIBS% org.calmain.Main %*
```

→ .bat파일에 jar파일이 들어있는 경로를 지정해줌

- java -cp %LIBS% com.calmain.Main "%1" → 파라미터 한 개만 받기

- java -cp %LIBS% com.calmain.Main %* → 들어온 파라미터 전부 받기

```
C:\mocomsys\workspace\korean-calculator>calculator.bat 백+오십+오
C:\mocomsys\workspace\korean-calculator>set LIBS=C:\mocomsys\workspace\korean-calculator\lib\target\korean-calculator-lib-1.0.jar
C:\mocomsys\workspace\korean-calculator>set LIBS=C:\mocomsys\workspace\korean-calculator\lib\target\korean-calculator-lib-1.0.jar;"C:\mocomsys\workspace\korean-calculator\app\target\korean-calculator-app-1.0.jar"
C:\mocomsys\workspace\korean-calculator>java -cp C:\mocomsys\workspace\korean-calculator\lib\target\korean-calculator-lib-1.0.jar;"C:\mocomsys\workspace\korean-calculator\app\target\korean-calculator-app-1.0.jar" com.whoana.app.Main "백+오십+오"
"백+오십+오" 는 "백오십오" 이지롱 ~ ~
```

→ .bat파일만 실행

TOMCAT

▼ 기능

1. WAS(Web Application Server)
 - 웹 애플리케이션과 서버 환경을 만들어 동작 시키는 기능을 제공
2. HTTP 기반의 서비스를 수행하는 HTTP 서버 기능도 포함
3. 상용 WAS의 레퍼런스 → 대부분 WAS가 톰캣을 기반으로 만들어짐
4. 100% 오픈 소스

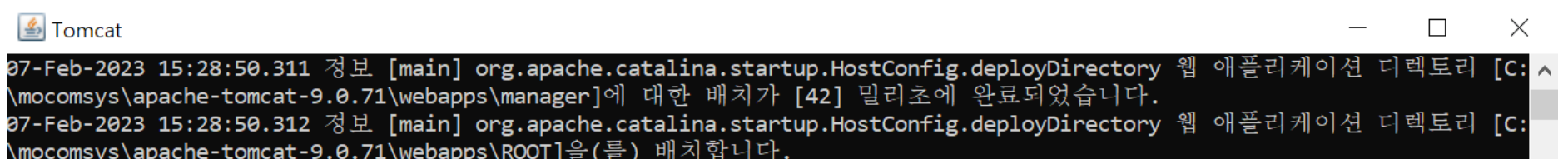
톰캣 환경설정

- set CATALINA_HOME=C:\mocomsys\apache-tomcat-9.0.71
- set PATH=C:\mocomsys\apache-tomcat-9.0.71\bin
- catalina.bat run

```
C:\mocomsys\apache-tomcat-9.0.71\bin>catalina.bat run
Using CATALINA_BASE:   "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_HOME:   "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_TMPDIR: "C:\mocomsys\apache-tomcat-9.0.71\temp"
Using JRE_HOME:        "C:\Program Files\Java\jdk1.8.0_351"
Using CLASSPATH:       "C:\mocomsys\apache-tomcat-9.0.71\bin\bootstrap.jar;C:\mocomsys\apache-tomcat-9.0.71\bin\tomcat-juli.jar"
Using CATALINA_OPTS:   ""
07-Feb-2023 16:38:35.438 ?뽕뽕 [main] org.apache.catalina.startup.VersionLoggerListener.log ?췘췘 췘췘췘 췘대췘:   Apache Tomcat/9.0.71
```

- startup.bat / catalina.bat start

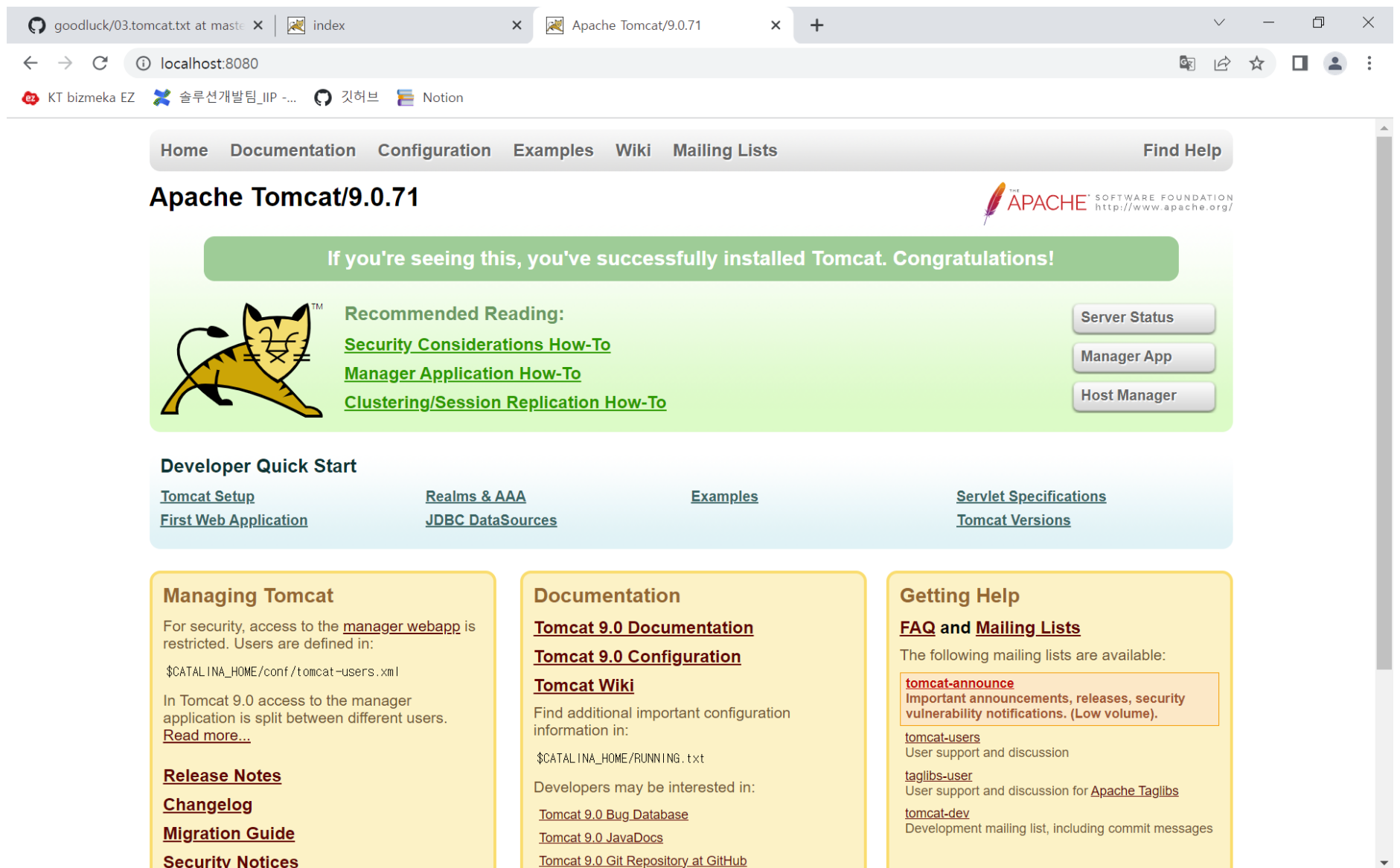
```
C:\mocomsys\apache-tomcat-9.0.71\bin>startup.bat
Using CATALINA_BASE:   "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_HOME:   "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_TMPDIR: "C:\mocomsys\apache-tomcat-9.0.71\temp"
```



The screenshot shows a window titled "Tomcat" with a standard Windows interface (minimize, maximize, close buttons). The log text inside the window is as follows:

```
07-Feb-2023 15:28:50.311 정보 [main] org.apache.catalina.startup.HostConfig.deployDirectory 웹 애플리케이션 디렉토리 [C:\mocomsys\apache-tomcat-9.0.71\webapps\manager]에 대한 배치가 [42] 밀리초에 완료되었습니다.
07-Feb-2023 15:28:50.312 정보 [main] org.apache.catalina.startup.HostConfig.deployDirectory 웹 애플리케이션 디렉토리 [C:\mocomsys\apache-tomcat-9.0.71\webapps\ROOT]을(를) 배치합니다.
```

→ localhost:8080 홈페이지를 띄울 수 있음



- tomcat-user 변경

```
C:\mocomsys\apache-tomcat-9.0.71\bin>cd ..
C:\mocomsys\apache-tomcat-9.0.71>cd conf
C:\mocomsys\apache-tomcat-9.0.71\conf>vim tomcat-users.xml
```

```
<user username="admin" password="admin" roles="manager-gui, admin-gui"/>
<user username="robot" password="robot" roles="manager-script"/>
```

```
<role rolename="tomcat"/>
<role rolename="role1"/>
<user username="tomcat" password="tomcat" roles="tomcat"/>
<user username="both" password="both" roles="tomcat,role1"/>
<user username="role1" password="role1" roles="role1"/>
```

- password와 roles을 변경하면 됨
- 이렇게 변경하면 tomcat 화면의 권한이 필요한 곳에 접속 할 수 있음

< 톰캣 윈도우 서비스로 올리기 / 웹 어플리케이션 하나 올려보기 → 수작업으로(html로 작성)>

1. webapps에 폴더 하나 생성

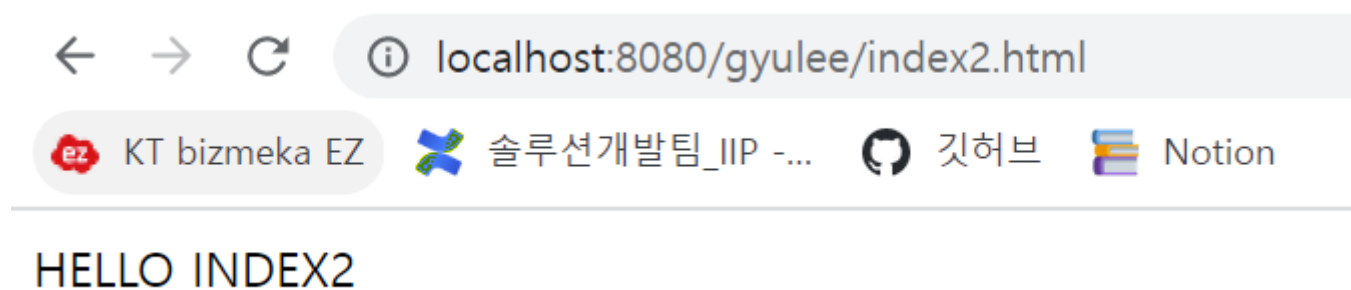
mocomsys > apache-tomcat-9.0.71 > webapps > gyulee

2. vi로 html파일 생성

```
C:\mocomsys\apache-tomcat-9.0.71>cd webapps
C:\mocomsys\apache-tomcat-9.0.71\webapps>cd gyulee
C:\mocomsys\apache-tomcat-9.0.71\webapps\gyulee>vim index2.html
```

```
index2.html + (C:\mocomsys\apache-tomcat-9.0.71\webapps\gyulee) - VIM
<html>
  <head>
    <title>index2</title>
  </head>
  <body>
    HELLO INDEX2
  </body>
</html>
```

3. localhost:8080/gyulee/index2.html 실행



- 한글깨짐현상 해결해보기

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>index</title>
  </head>
  <body>
    안녕하세요. hello
  </body>
</html>
```

<툼캣 공부 사이트 1-10번까지 보면 좋음 - 심화 9,10번>

Apache Tomcat 9

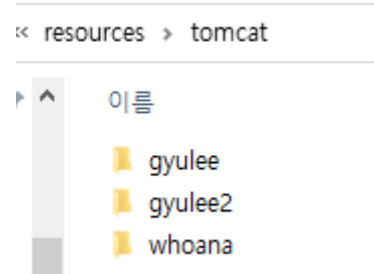
This is the top-level entry point of the documentation bundle for the Apache Tomcat Servlet/JSP container. Apache Tomcat version 9.0 implements the Servlet 4.0 and JavaServer Pages 2.3 specifications from the Java Community Process, and includes many additional features that make it a useful platform for developing and deploying web applications and web services.

 <https://tomcat.apache.org/tomcat-9.0-doc/index.html>

<TOMCAT 여러 개 띄우기>

- CATALINA_BASE : 설치파일은 하나, 여러개의 톰캣

1. 다른 TOMCAT을 띄울 경로를 만들



2. 경로로 이동해서 run.bat 파일을 만들어줌 → CATALINA_BASE 경로 잡아주기

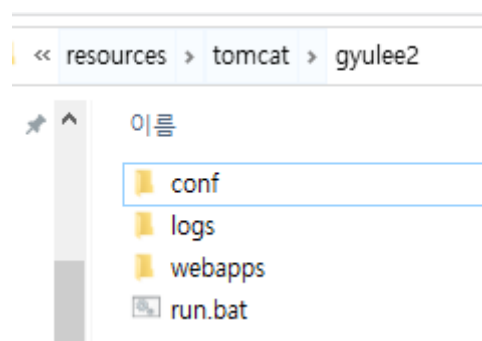
run.bat (C:\Github\goodluck\resources\tomcat\gyulee2) - VIM

```
echo "CATALINA_HOME: %CATALINA_HOME%"
set CATALINA_BASE=C:\Github\goodluck\resources\tomcat\gyulee2
#catalina.bat configtest
startup.bat run
```

```
C:\Github\goodluck\resources\tomcat\gyulee2>vim run.bat
C:\Github\goodluck\resources\tomcat\gyulee2>startup.bat
Using CATALINA_BASE: "C:\Github\goodluck\resources\tomcat\gyulee2"
Using CATALINA_HOME: "C:\mocomsys\apache-tomcat-9.0.71"
```

→ CATALINA_BASE 경로 바뀐 것 확인

3. conf 파일 생성해서 항상 필요한 파일 server.xml, web.xml 가져오기



```
C:\Github\goodluck\resources\tomcat\gyulee2>cd conf
C:\Github\goodluck\resources\tomcat\gyulee2\conf>copy C:\mocomsys\apache-tomcat-9.0.71\conf\server.xml
1개 파일이 복사되었습니다.
C:\Github\goodluck\resources\tomcat\gyulee2\conf>copy C:\mocomsys\apache-tomcat-9.0.71\conf\web.xml
1개 파일이 복사되었습니다.
C:\Github\goodluck\resources\tomcat\gyulee2\conf>copy C:\mocomsys\apache-tomcat-9.0.71\conf\context.xml
1개 파일이 복사되었습니다.
```

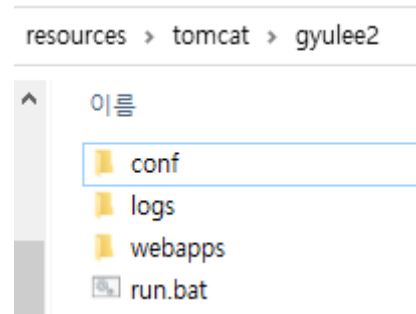
4. run.bat 명령어 실행 후 localhost:8080 입력 → 만들어진 폴더가 없기 때문에 404에러

HTTP 상태 404 – 찾을 수 없음

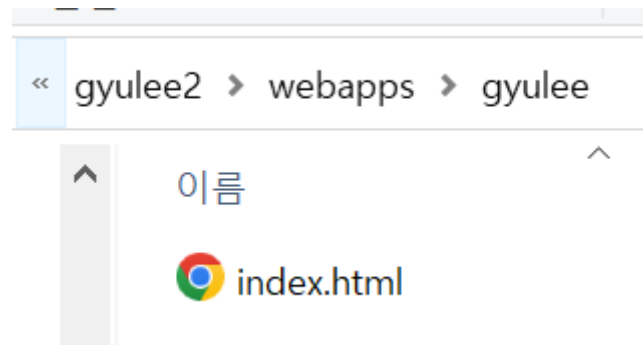
타입 상태 보고

설명 Origin 서버가 대상 리소스를 위한 현재의 representation을 찾지 못했거나, 그것이 존재하는지를 밝히려 하지 않습니다.

Apache Tomcat/9.0.71



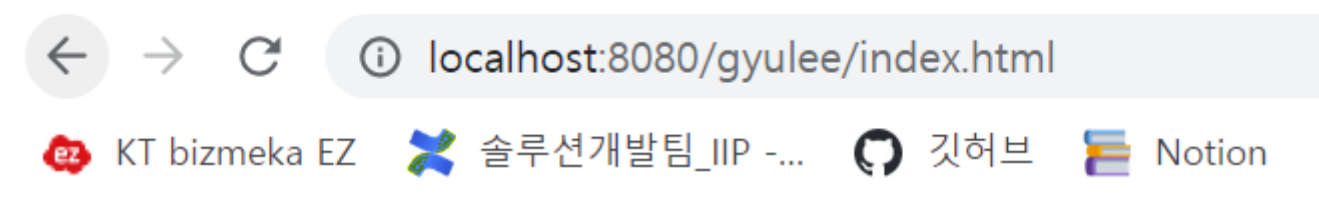
→ logs, webapps폴더가 저절로 생김



→ webapps에 html에 파일은 만들어준다

```
C:\Github\goodluck\resources\tomcat\gyulee2\webapps>startup.bat
Using CATALINA_BASE: "C:\Github\goodluck\resources\tomcat\gyulee2"
Using CATALINA_HOME: "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_TMPDIR: "C:\Github\goodluck\resources\tomcat\gyulee2\temp"
Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_351"
Using CLASSPATH: "C:\mocomsys\apache-tomcat-9.0.71\bin\bootstrap.jar;C:\mocomsys\apache-tomcat-9.0.71\bin\tomcat-juli.jar"
Using CATALINA_OPTS: ""
C:\Github\goodluck\resources\tomcat\gyulee2\webapps>startup.bat
Using CATALINA_BASE: "C:\Github\goodluck\resources\tomcat\gyulee2"
Using CATALINA_HOME: "C:\mocomsys\apache-tomcat-9.0.71"
Using CATALINA_TMPDIR: "C:\Github\goodluck\resources\tomcat\gyulee2\temp"
Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_351"
Using CLASSPATH: "C:\mocomsys\apache-tomcat-9.0.71\bin\bootstrap.jar;C:\mocomsys\apache-tomcat-9.0.71\bin\tomcat-juli.jar"
Using CATALINA_OPTS: ""
```

→ startup.bat 명령어를 사용하여 tomcat 구동



안녕하세요. hello

→ webapps에 있는 파일의 경로를 적어주면 html이 실행

Eclipse

<SVN 설정>

```
C:\Users\dearew.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.5.v20221102-0933\jre\conf\security>
vim java.security
```

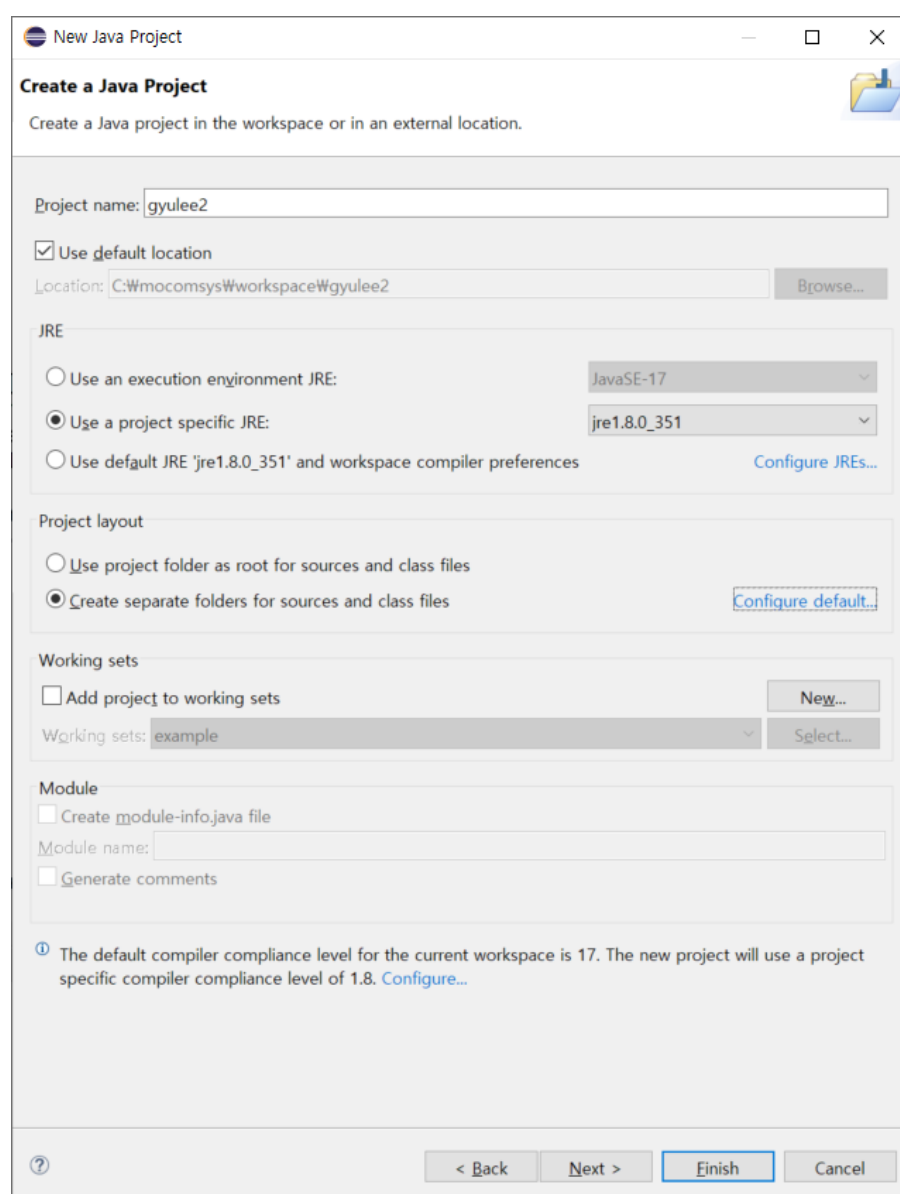
→ 이 위치의 java.security 파일 열어서

```
#      rsa_pkcs1_sha1, secp224r1
jdk.tls.disabledAlgorithms=SSLv3, TLSv1, TLSv1.1, RC4, DES, MD5withRSA, #
      DH keySize < 1024, EC keySize < 224, 3DES_EDE_CBC, anon, NULL

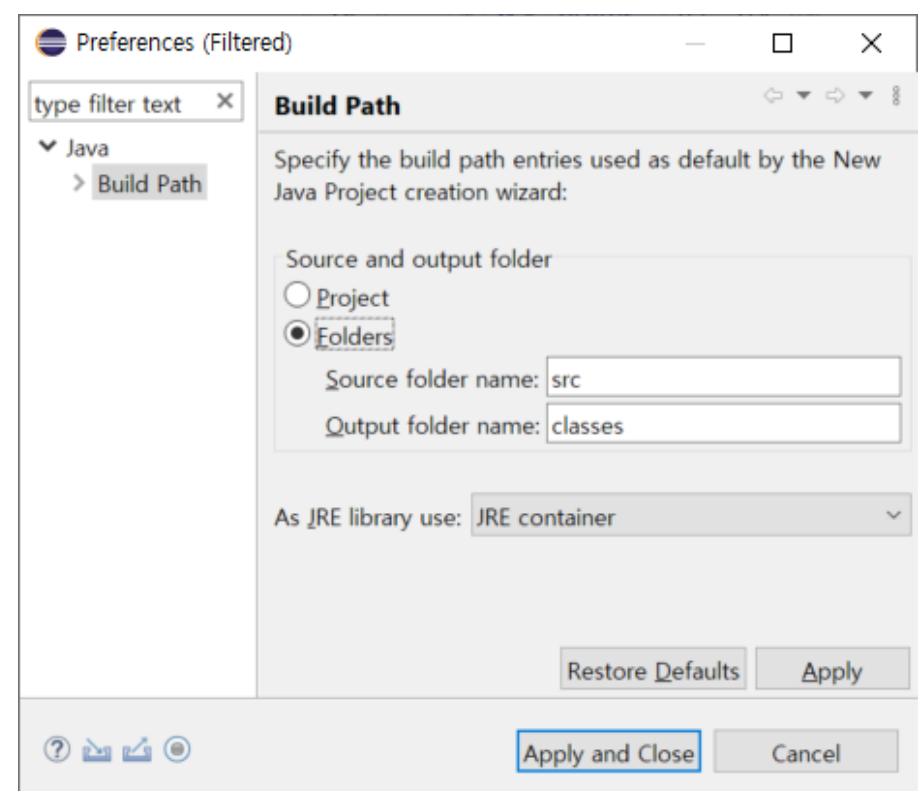
#
# Legacy algorithms for Secure Socket Layer/Transport Layer Security (SSL/TLS)
# processing in JSSE implementation.
#
# In some environments, a certain algorithm may be undesirable but it
# cannot be disabled because of its use in legacy applications. Legacy
# algorithms may still be supported, but applications should not use them
# as the security strength of legacy algorithms are usually not strong enough
/TLS
```

→ TLS 검색해서 TLSv1, TLSv1.1 삭제하고 저장

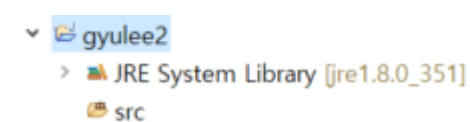
<프로젝트 생성하기>



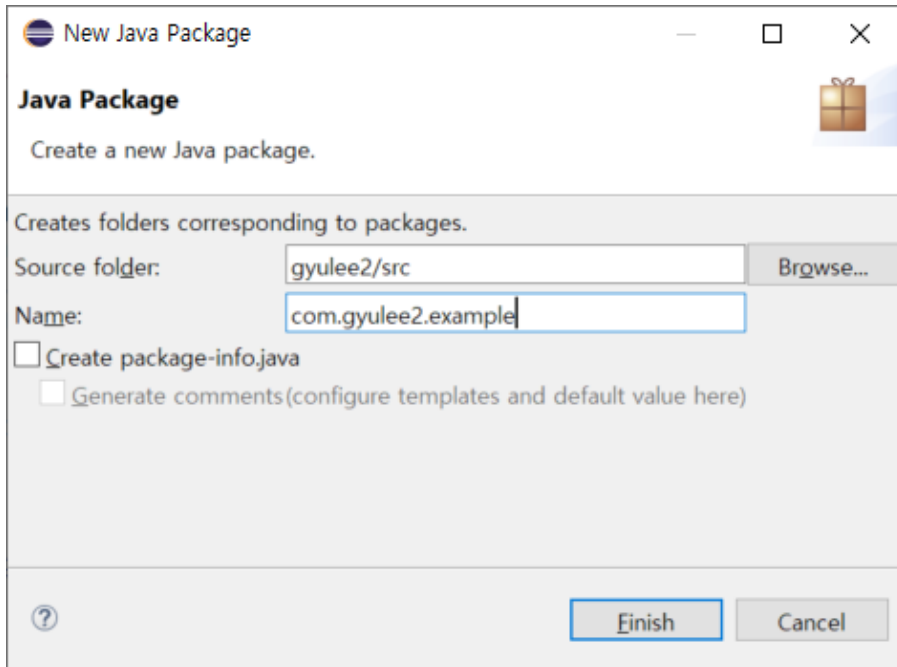
→ 프로젝트 이름, JRE 설정



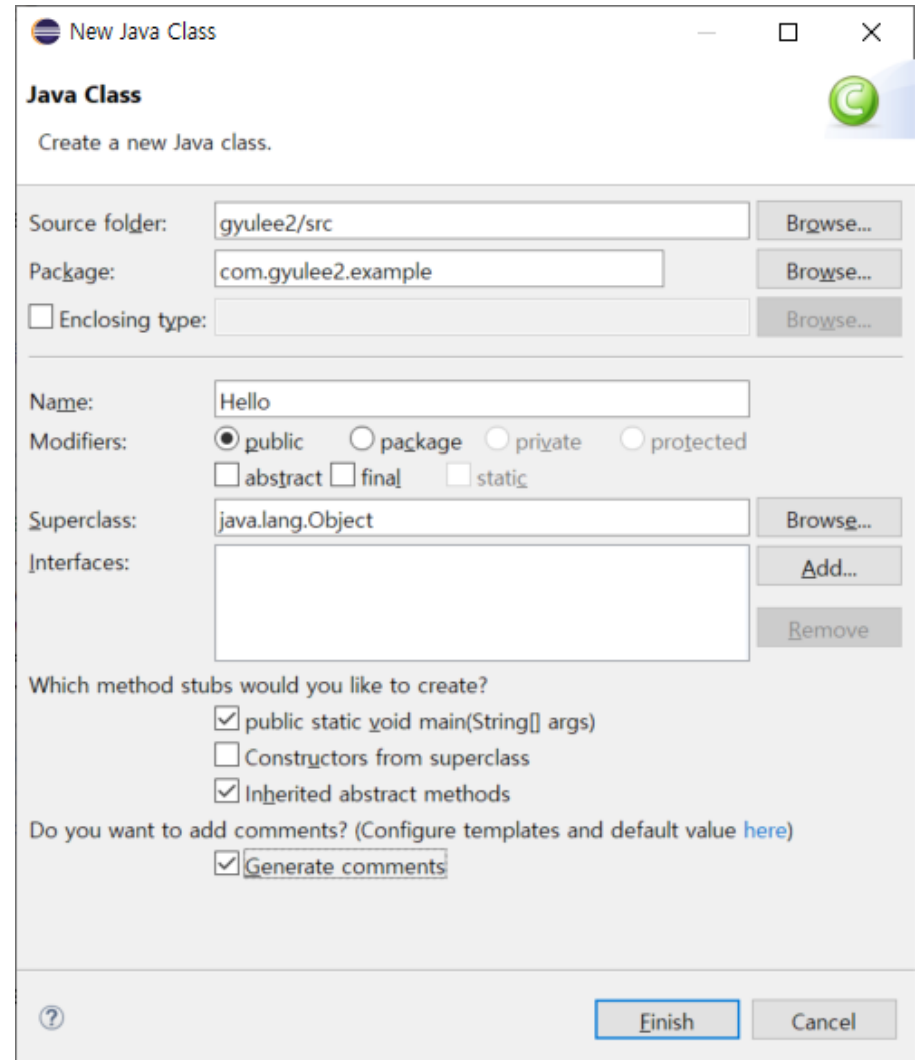
→ 경로 설정



→ 프로젝트 생성 완료

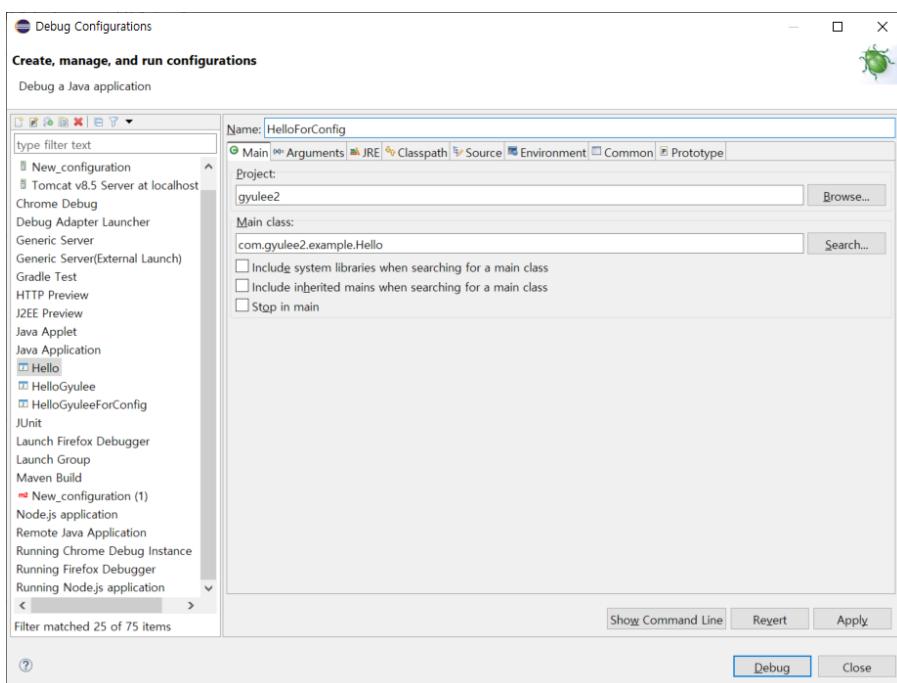


→ 패키지 생성

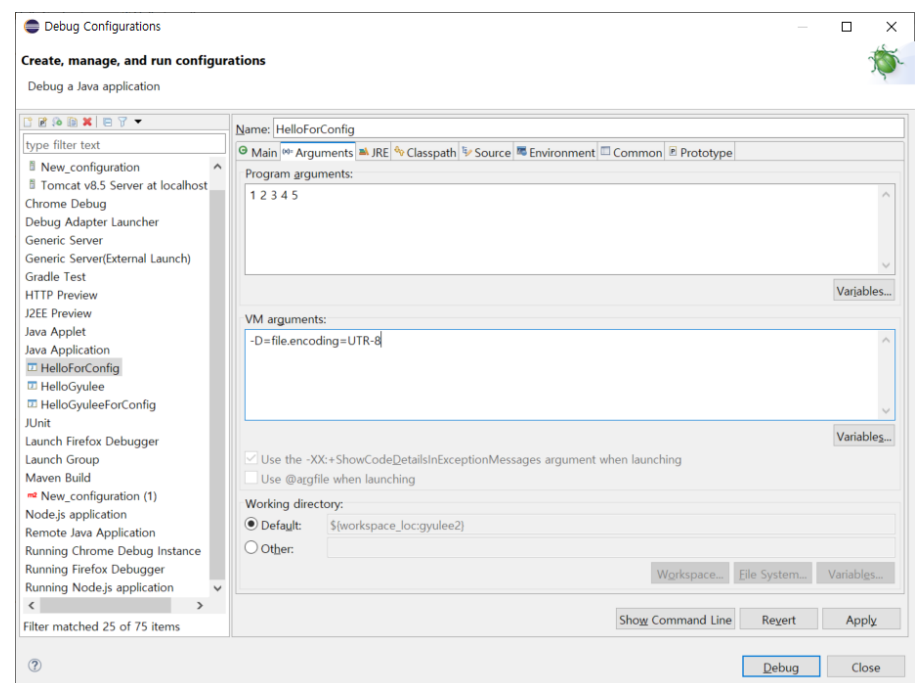


→ 클래스 생성(main함수와 커멘트 함께 생성)

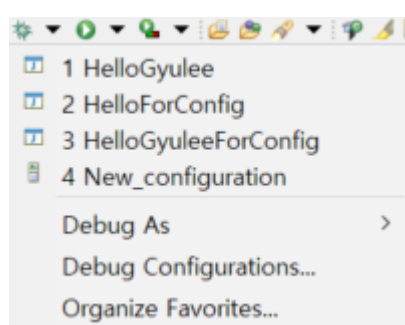
<디버깅하기>



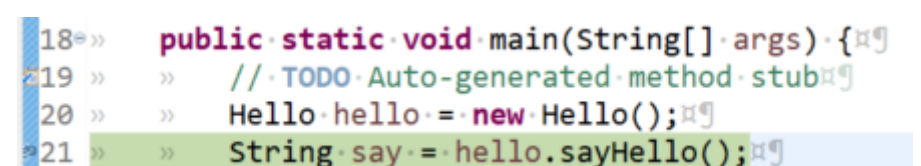
→ Debug Configurations에 가서 이름을 변경



→ Arguments(파라미터를 지정해주는 것)



→ 디버그 원하는 파일 선택

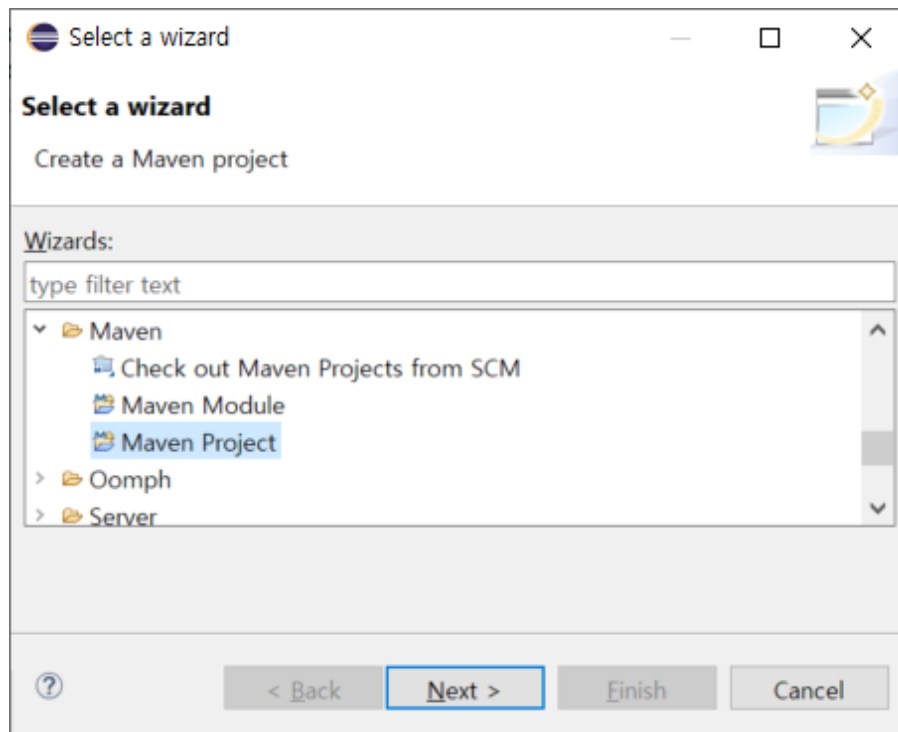


→ 값을 보고 싶은 곳을 고정

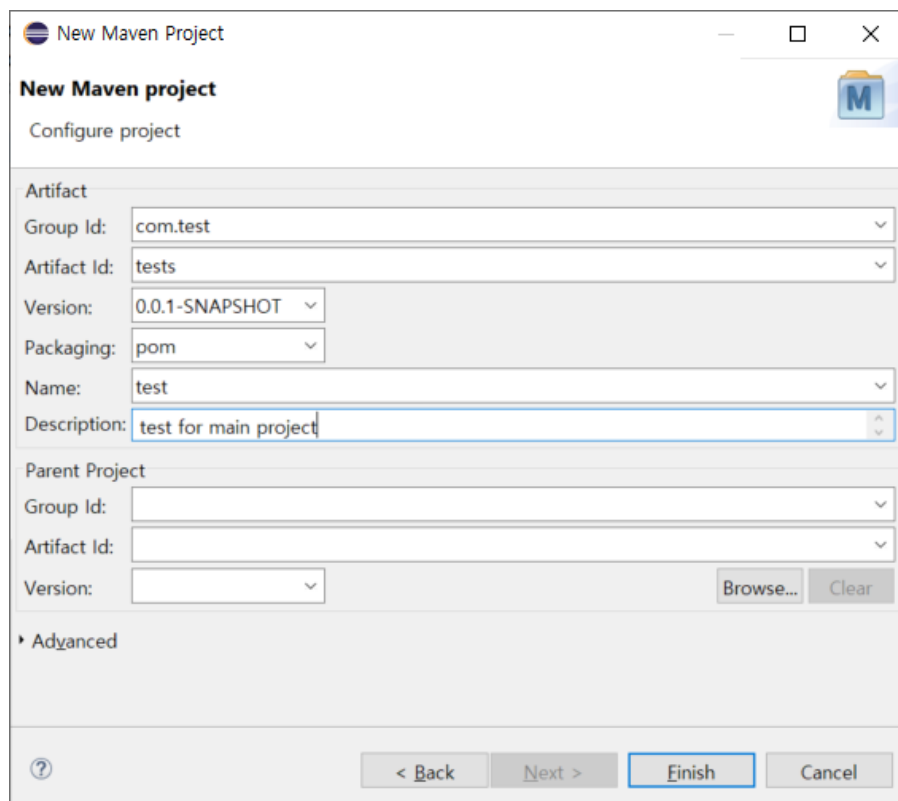
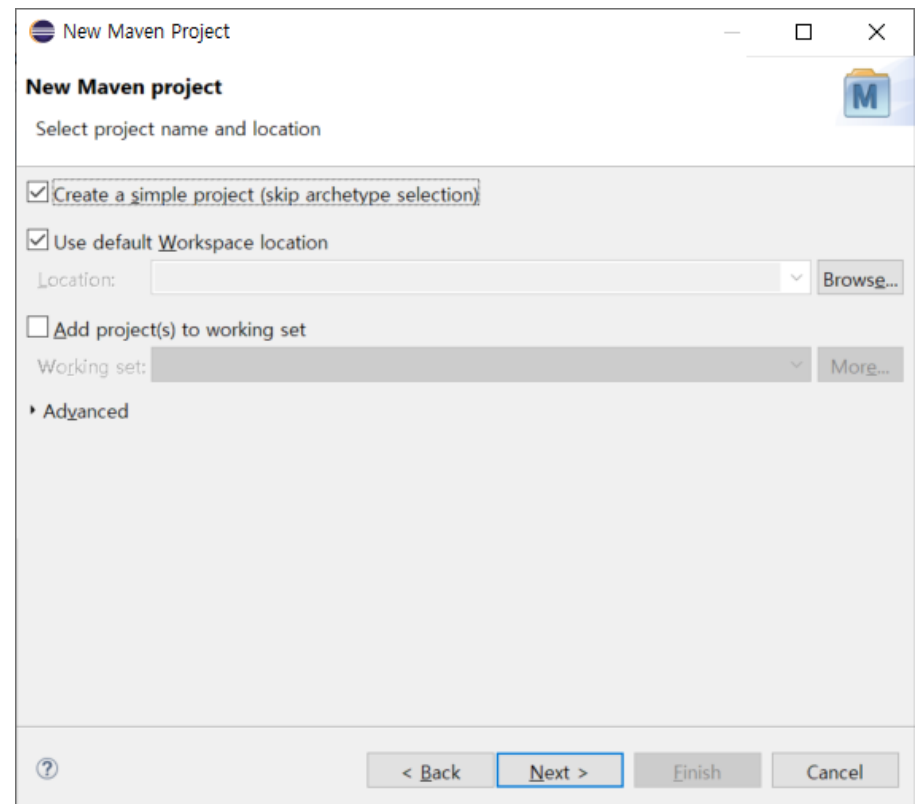
Name	Value
<init>() returned	(No explicit return value)
args	String[5] (id=16)
[0]	"1" (id=19)
[1]	"2" (id=23)
[2]	"3" (id=24)
[3]	"4" (id=25)
[4]	"5" (id=26)

→ 파라미터로 정해진 값 확인

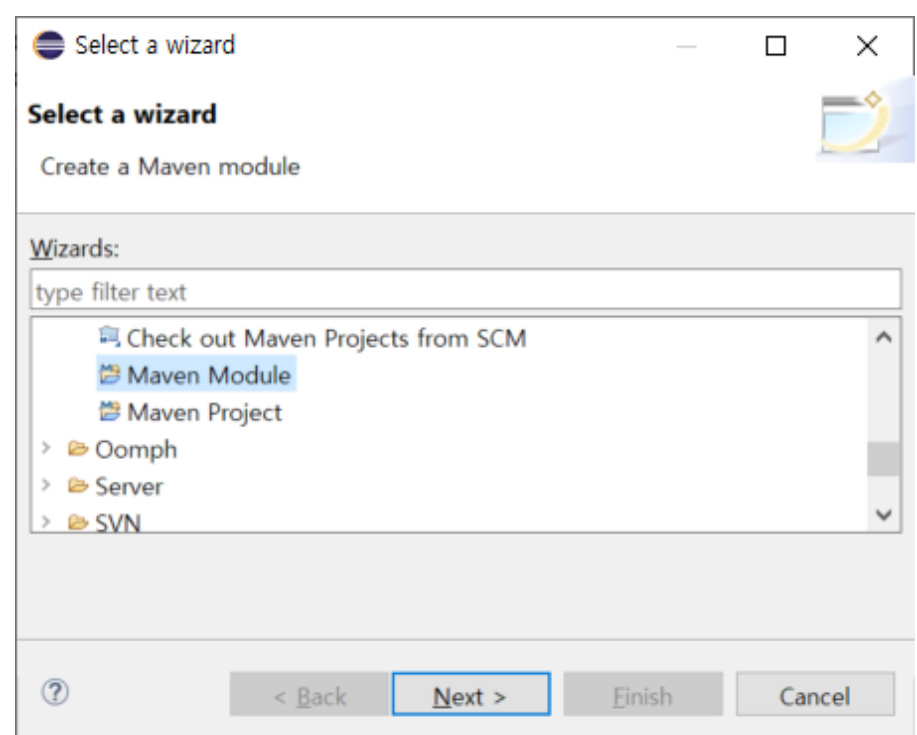
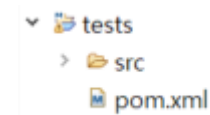
<모듈 프로젝트 만들기>



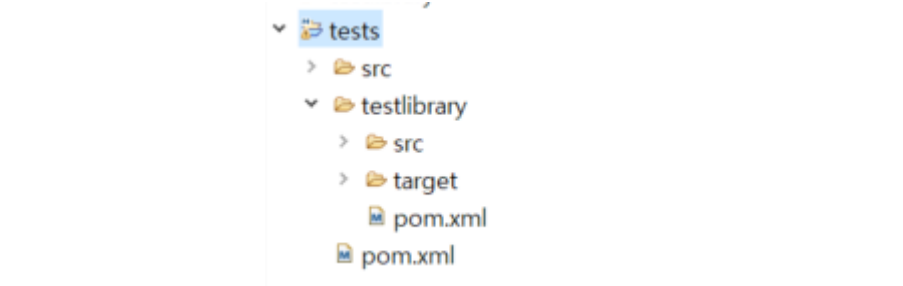
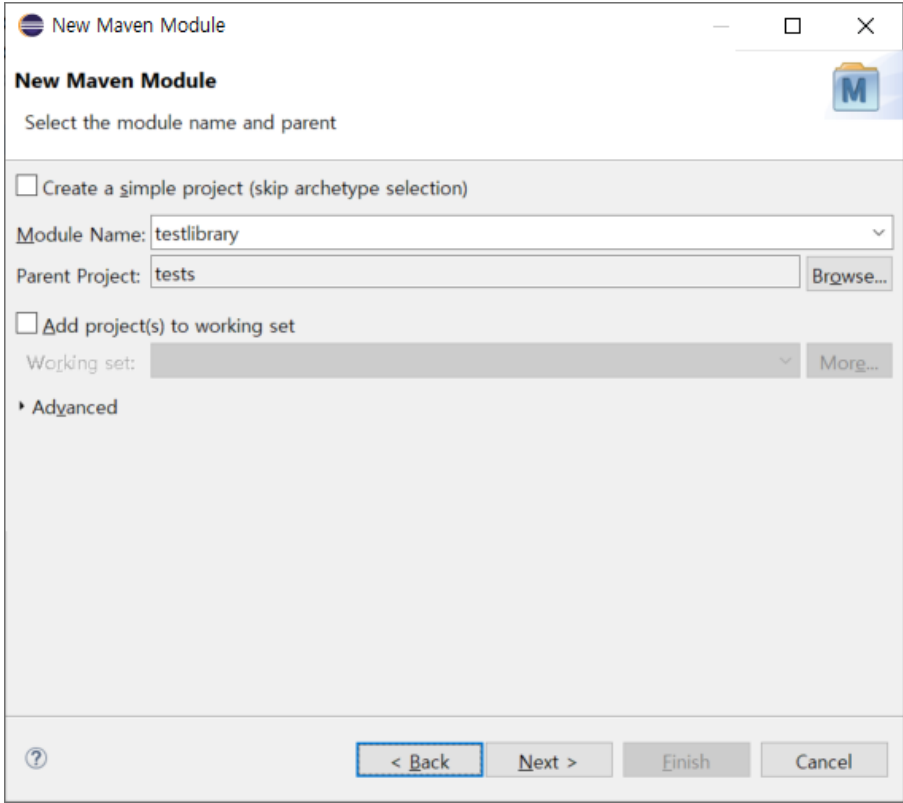
→ 메이블 프로젝트 선택



→ 기본정보 작성, pom으로 패키징



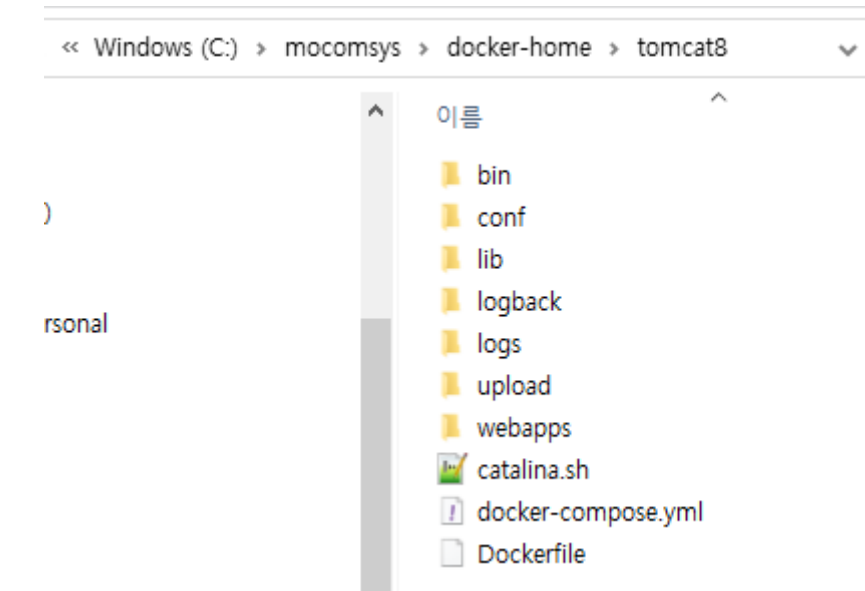
→ 프로젝트 우클릭 후 모듈 작성



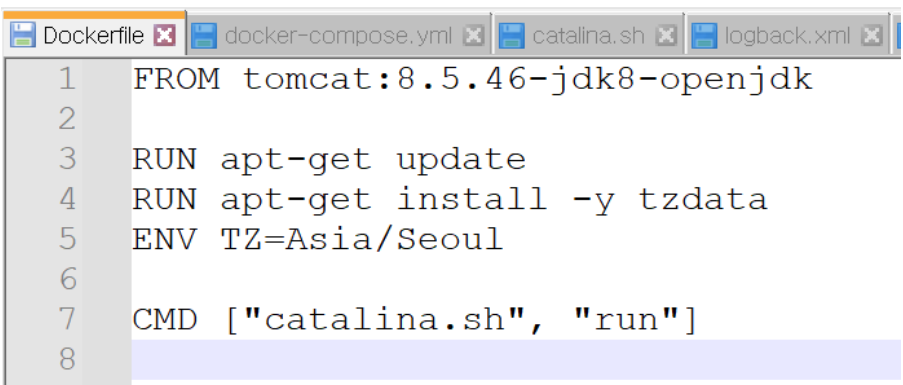
→ 모듈을 포함한 프로젝트 완성

→ next 눌러서 빌드 성공

도커 리눅스 환경으로



→ docker-home 만들고 tomcat8 파일 생성



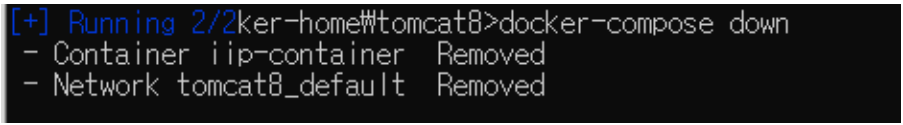
→ Dockerfile 생성



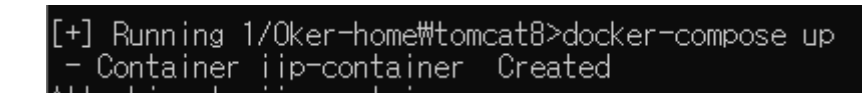
→ docker-compose.yml 파일 / ':'을 기준으로 '로컬경로 : 도커경로'



→ 톰캣 이미지 빌드

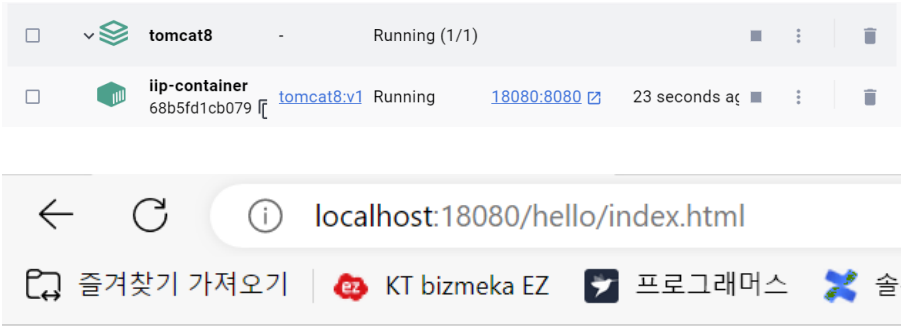


→ 도커 종료



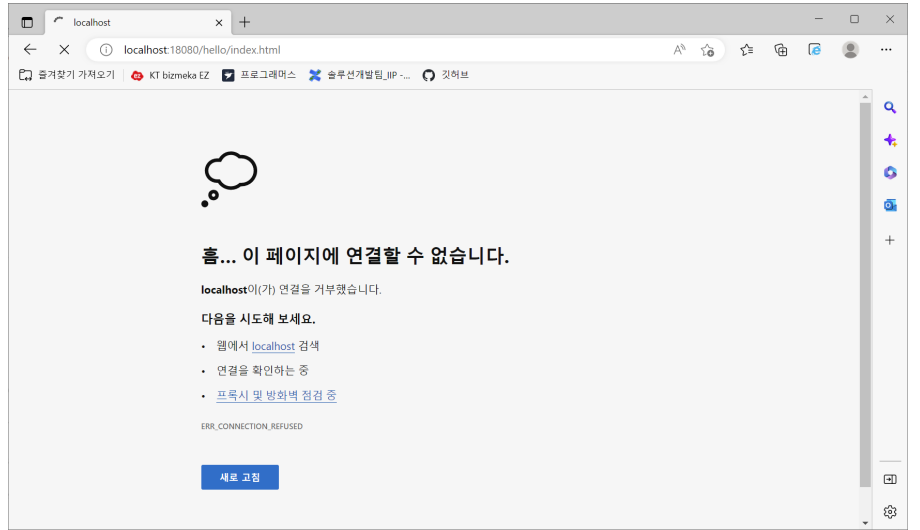
→ 도커 실행

	Name	Image	Status	Port(s)
	maven-env	e3e18baf1c76	Exited	
	oracle11g	173989657c34	Exited (137)	1521:1521 8080:8089



→ 주소 입력하면 webapps에 작성한 html 파일이 실행됨

→ 컨테이너 사라짐



→ 접속 불가