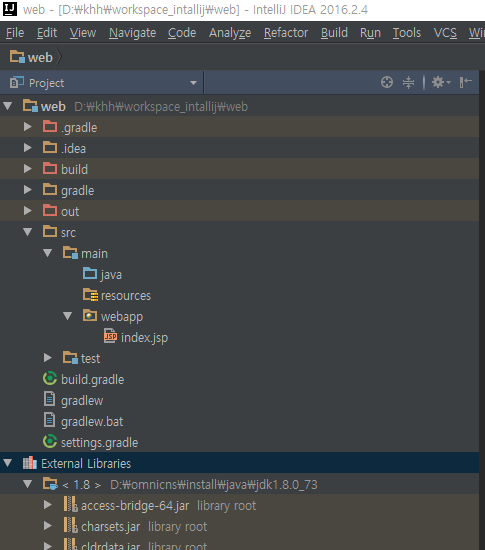
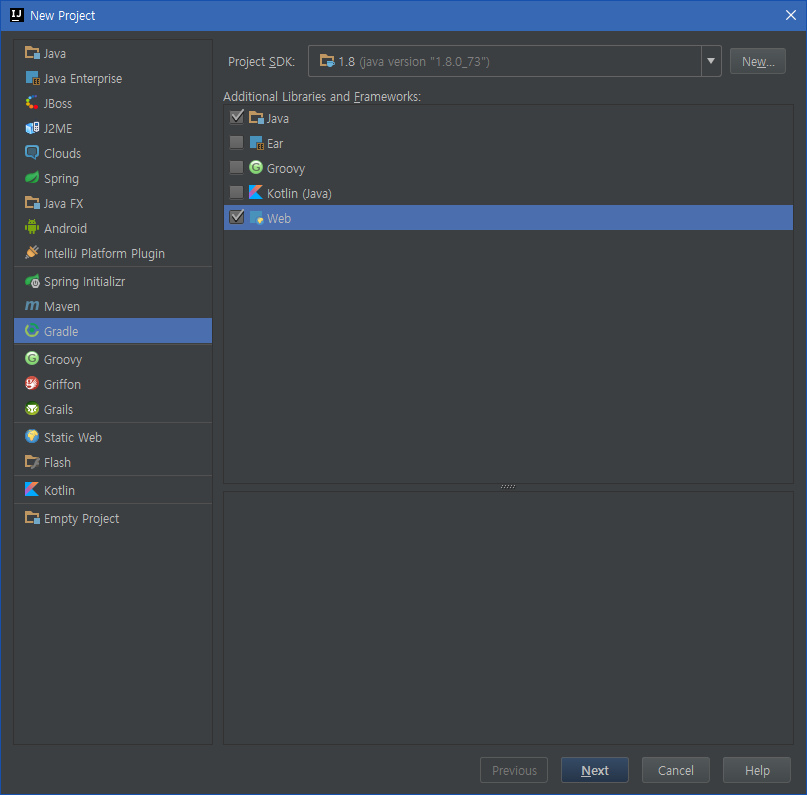
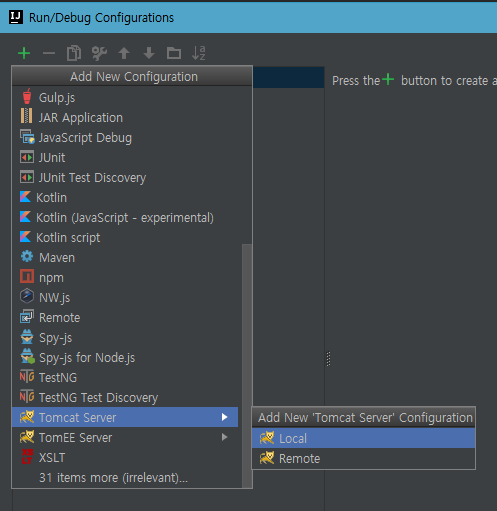
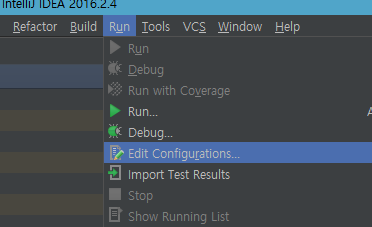
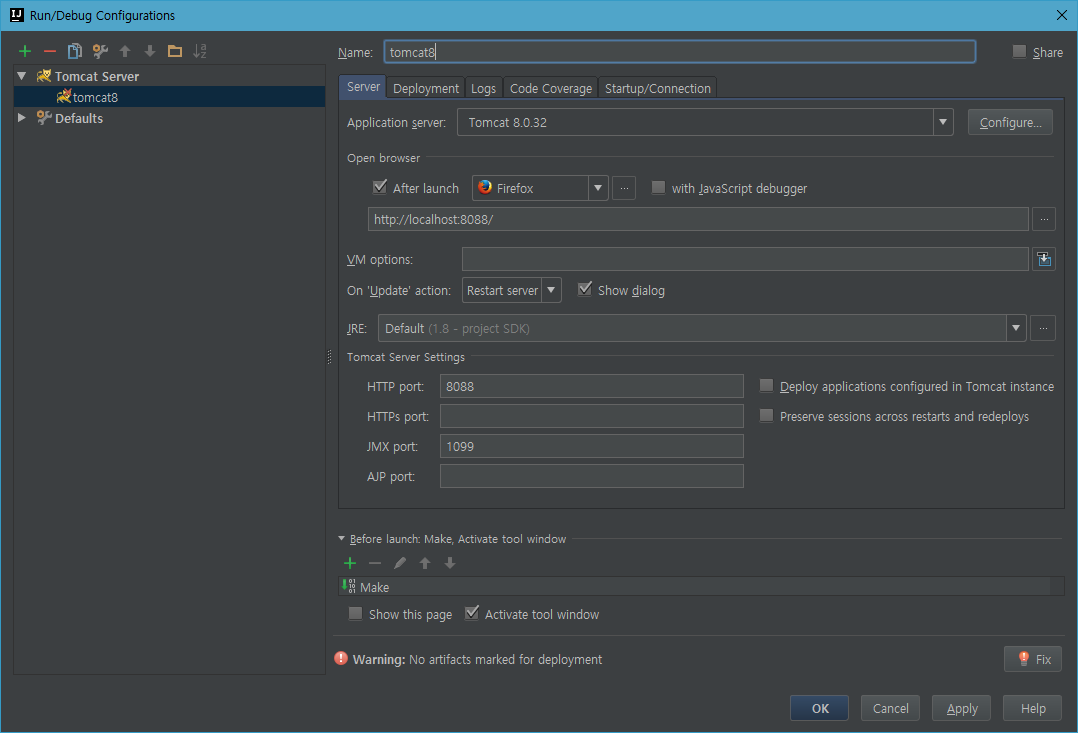
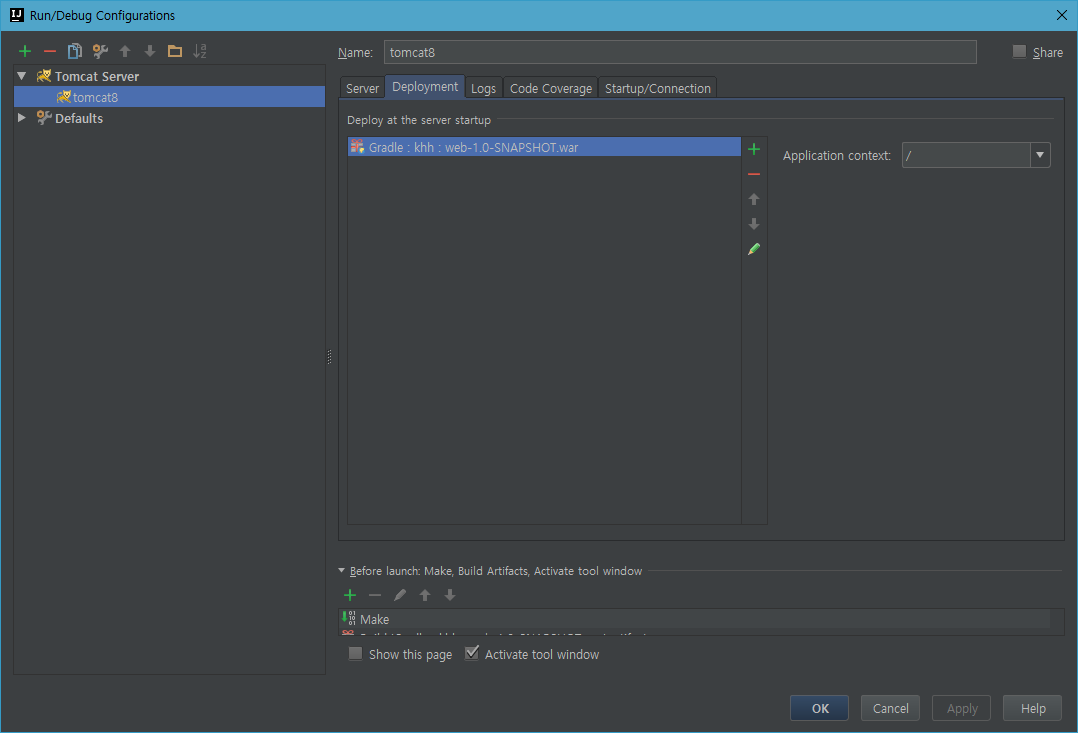
intellij 시작하기.

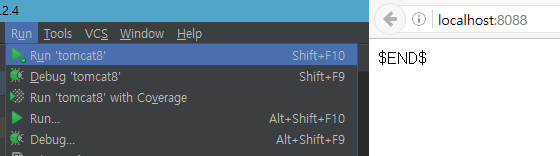




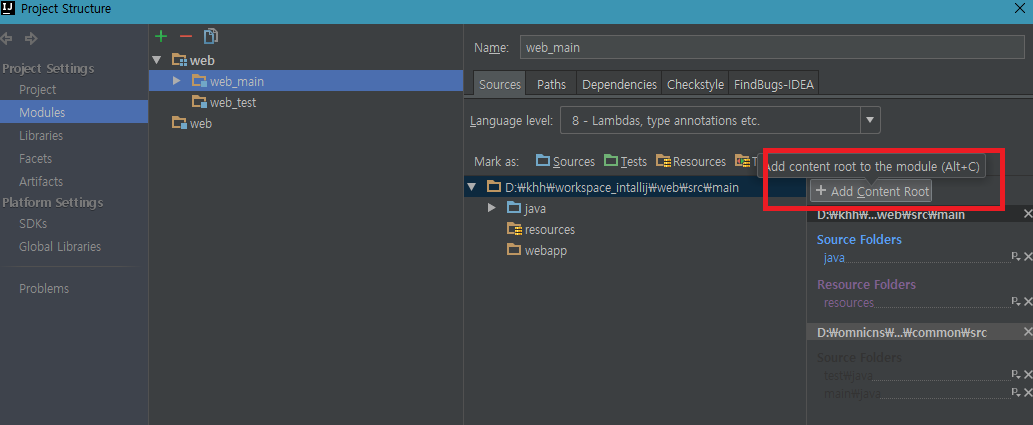


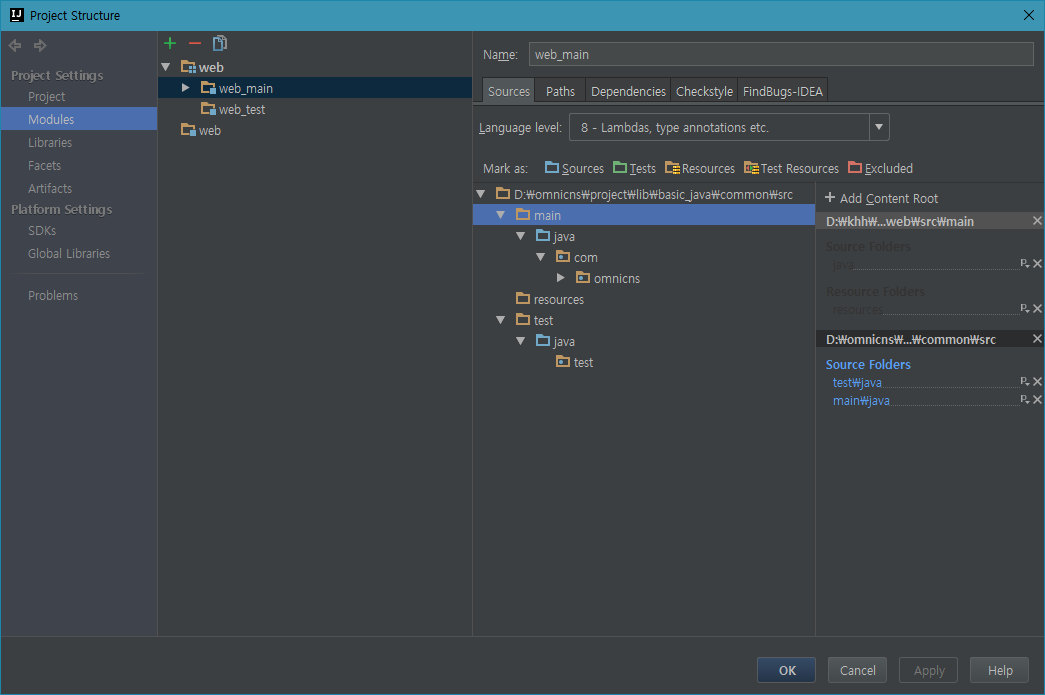
아래 Warning 부분의 Fix 눌러 Deployment 설정

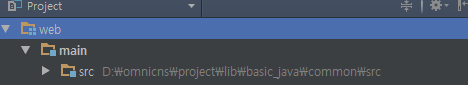




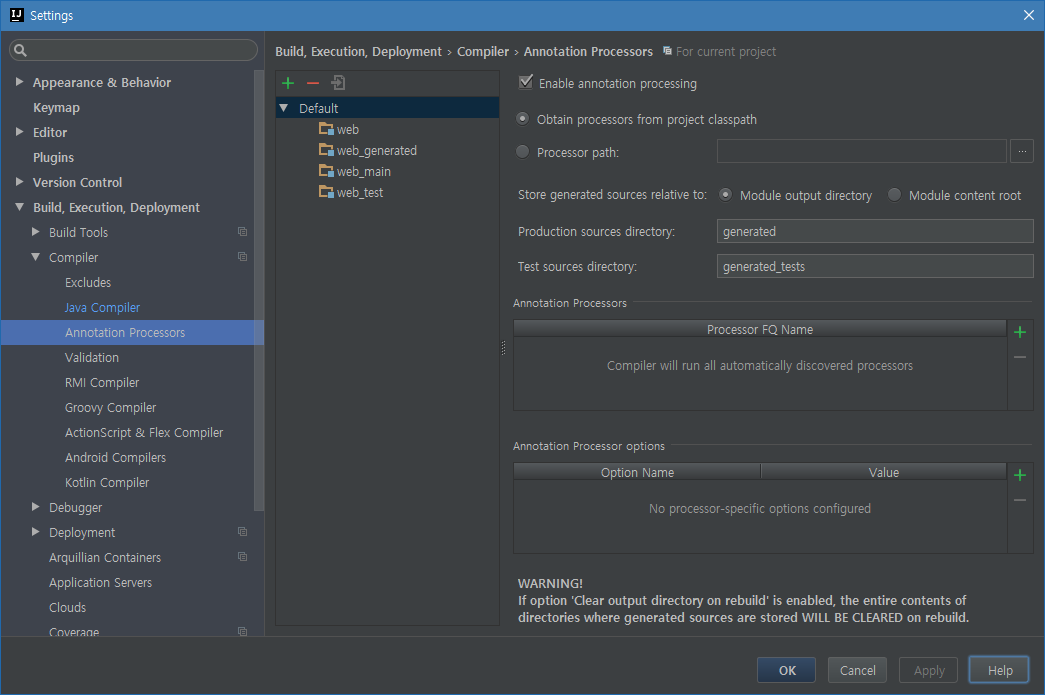
링크 소스







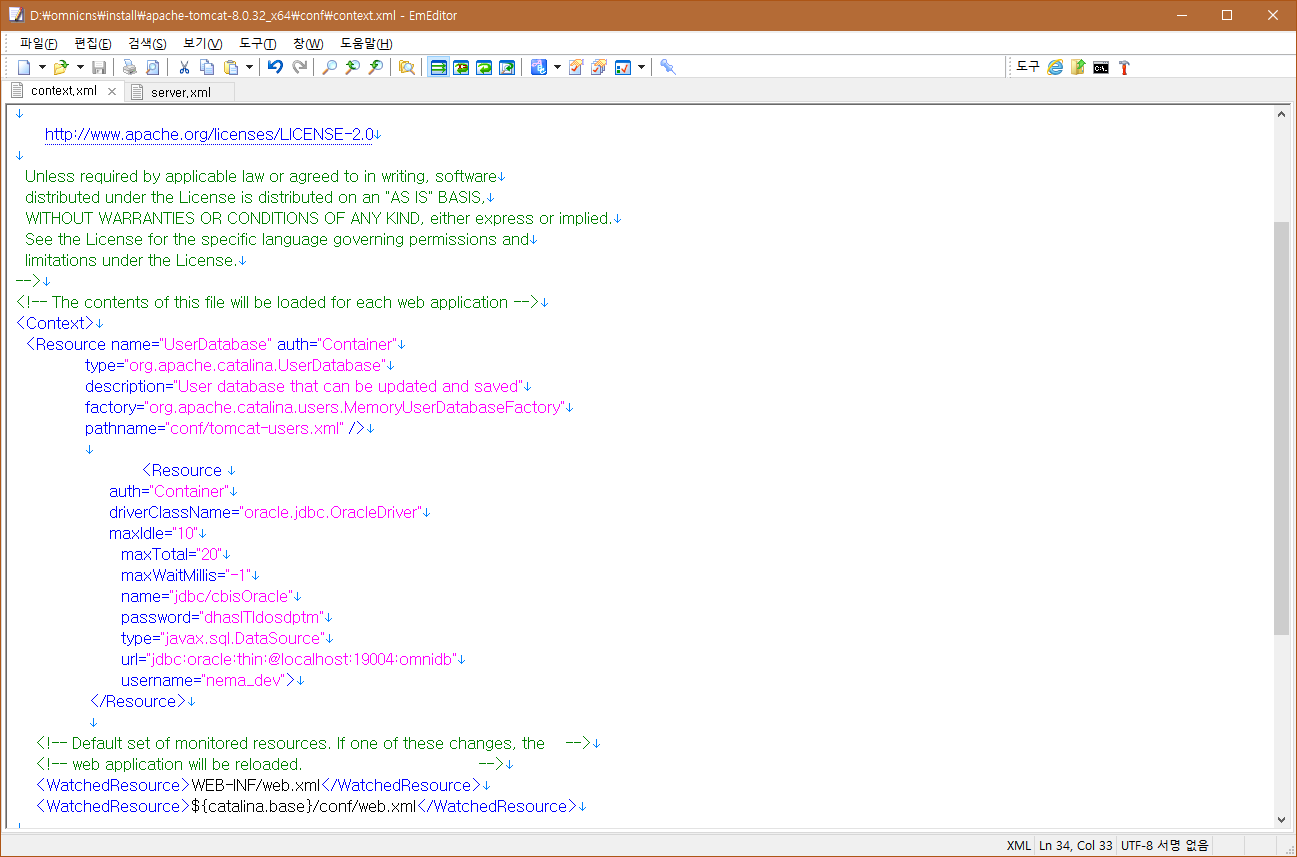
롬복



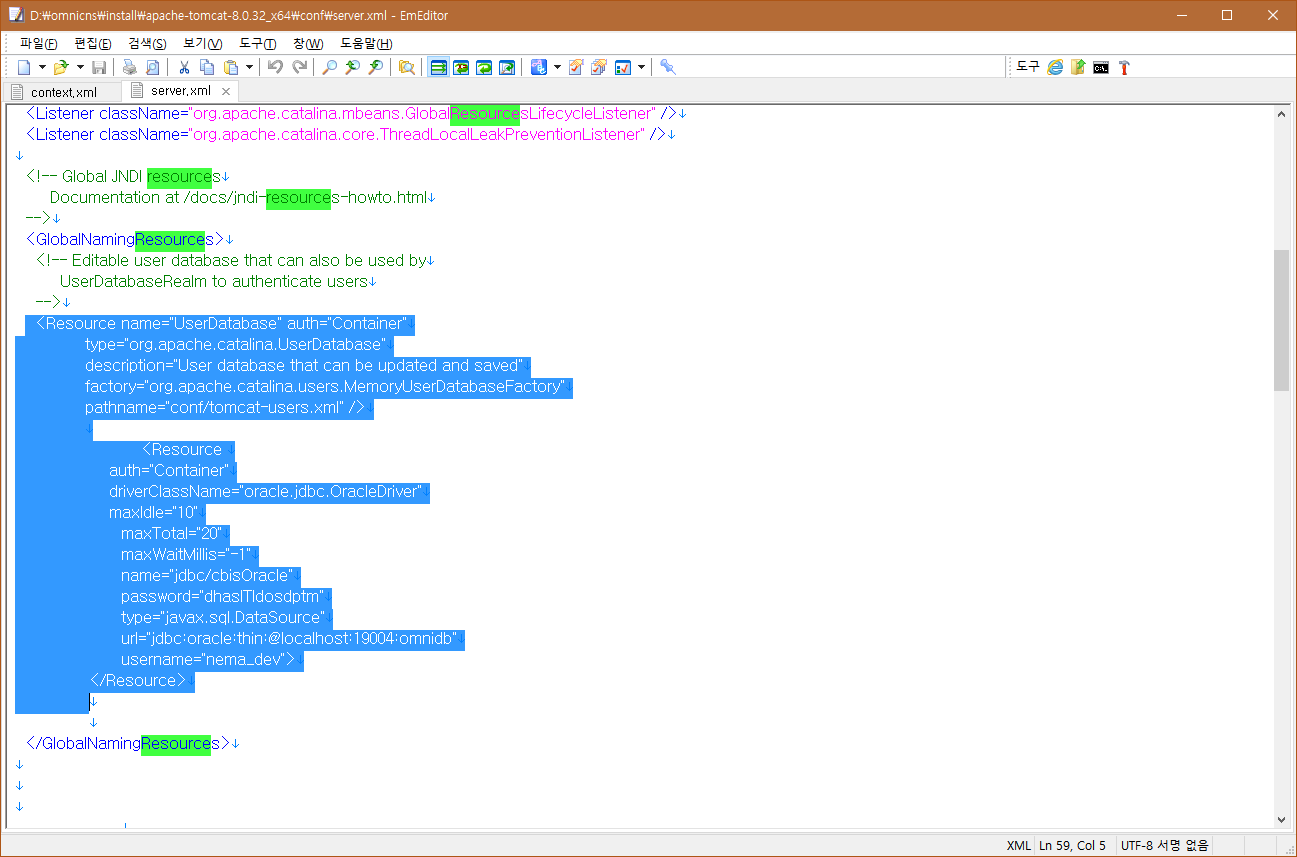
enable annotation procession

|  |  |
| --- | --- |
| 168 down vote accepted | I have fixed it in IDEA 12 by setting check box Enable annotation processing in:  **Settings -> Compiler -> Annotation Processors>Enable annontation processing**  For IDEA 2016.2:  **Preferences... > Build, Execution, Deployment > Compiler > Annotation Processors** |

JNDI설정



tomcat/conf/context.xml



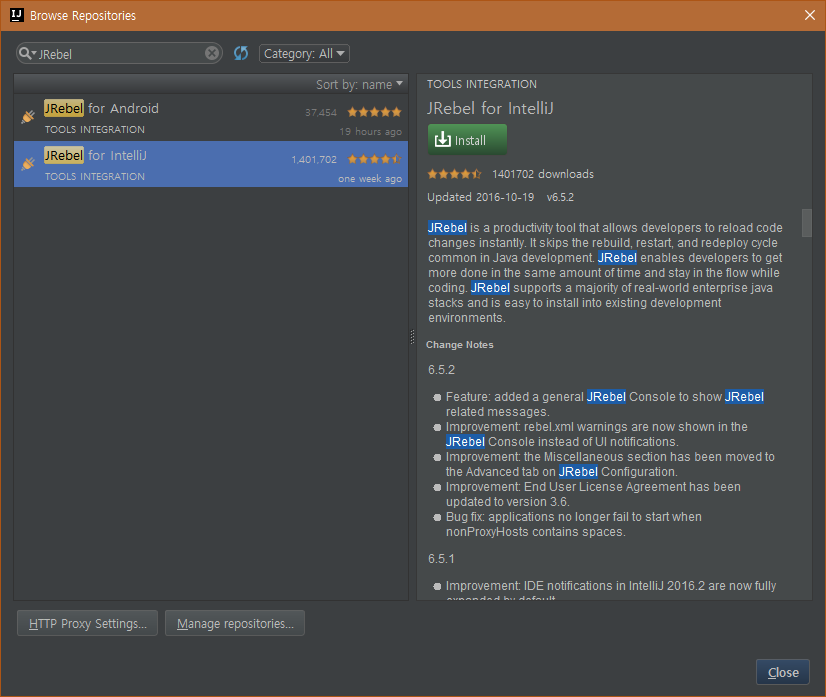
server.xml

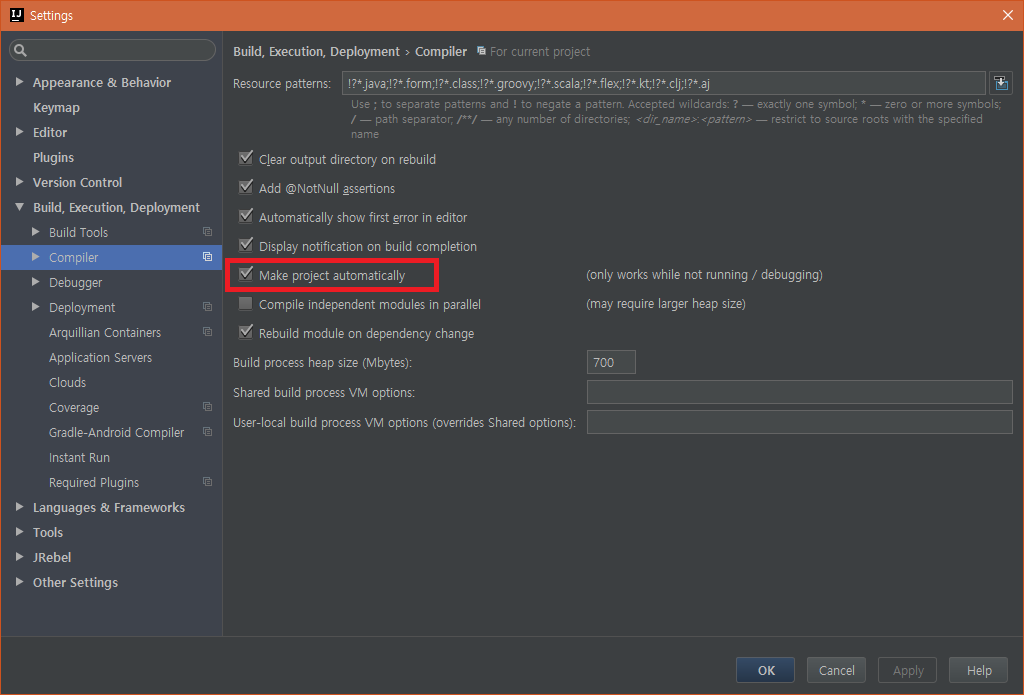
JRebel

<https://manuals.zeroturnaround.com/jrebel/ide/intellij.html>

<http://tiveloper.tistory.com/entry/IntelliJ-Ultimate-14-JRebel-%EC%A0%81%EC%9A%A9%ED%95%98%EA%B8%B0>

https://zeroturnaround.com/software/jrebel/quickstart/intellij/#!/reload





<http://lahuman.jabsiri.co.kr/137>

idea gradle

<https://docs.gradle.org/current/dsl/org.gradle.plugins.ide.idea.model.IdeaModule.html>

<https://youtrack.jetbrains.com/issue/IDEA-122577>

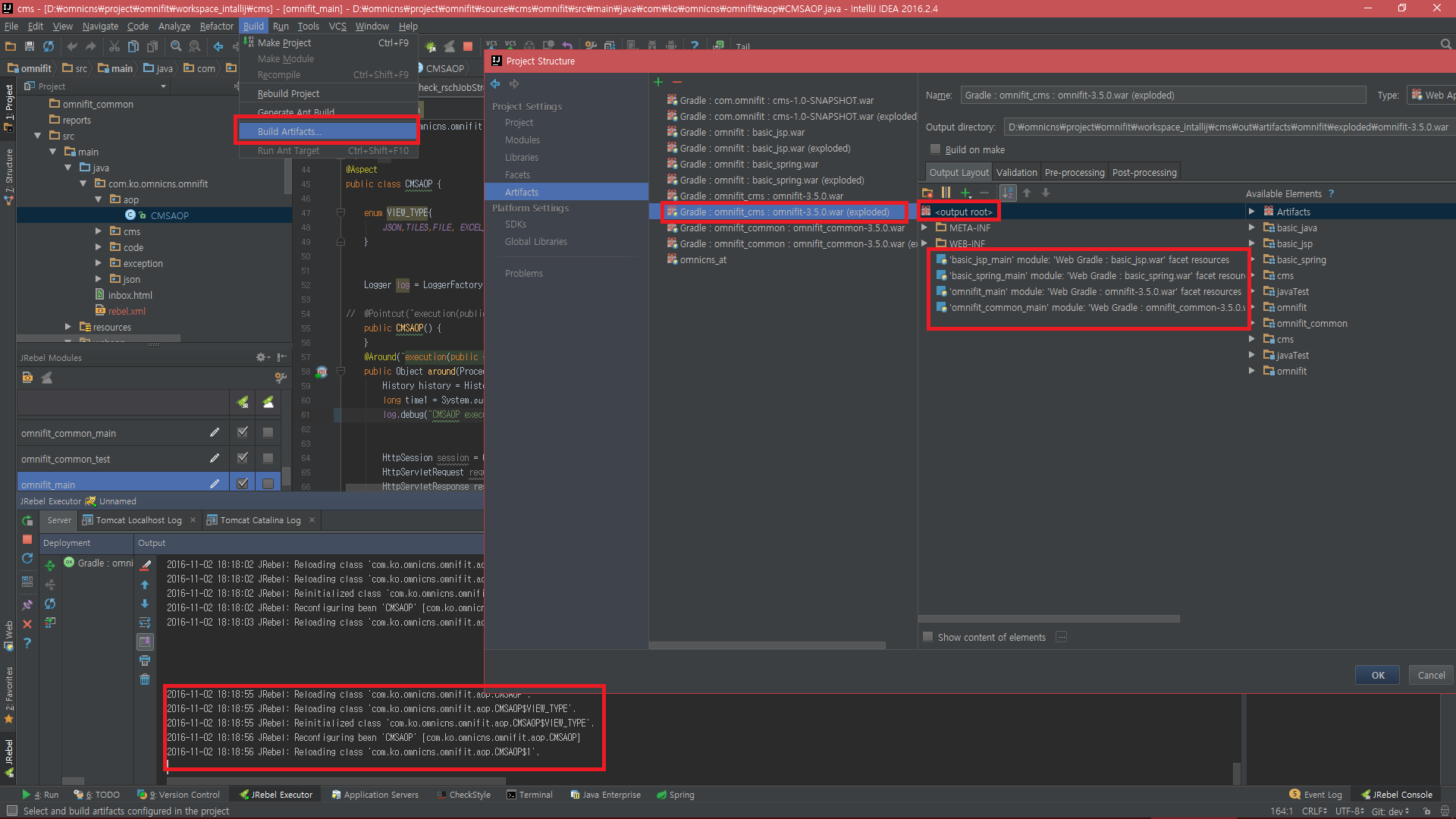
위처럼 하면 될것같지만 안된다..그래서

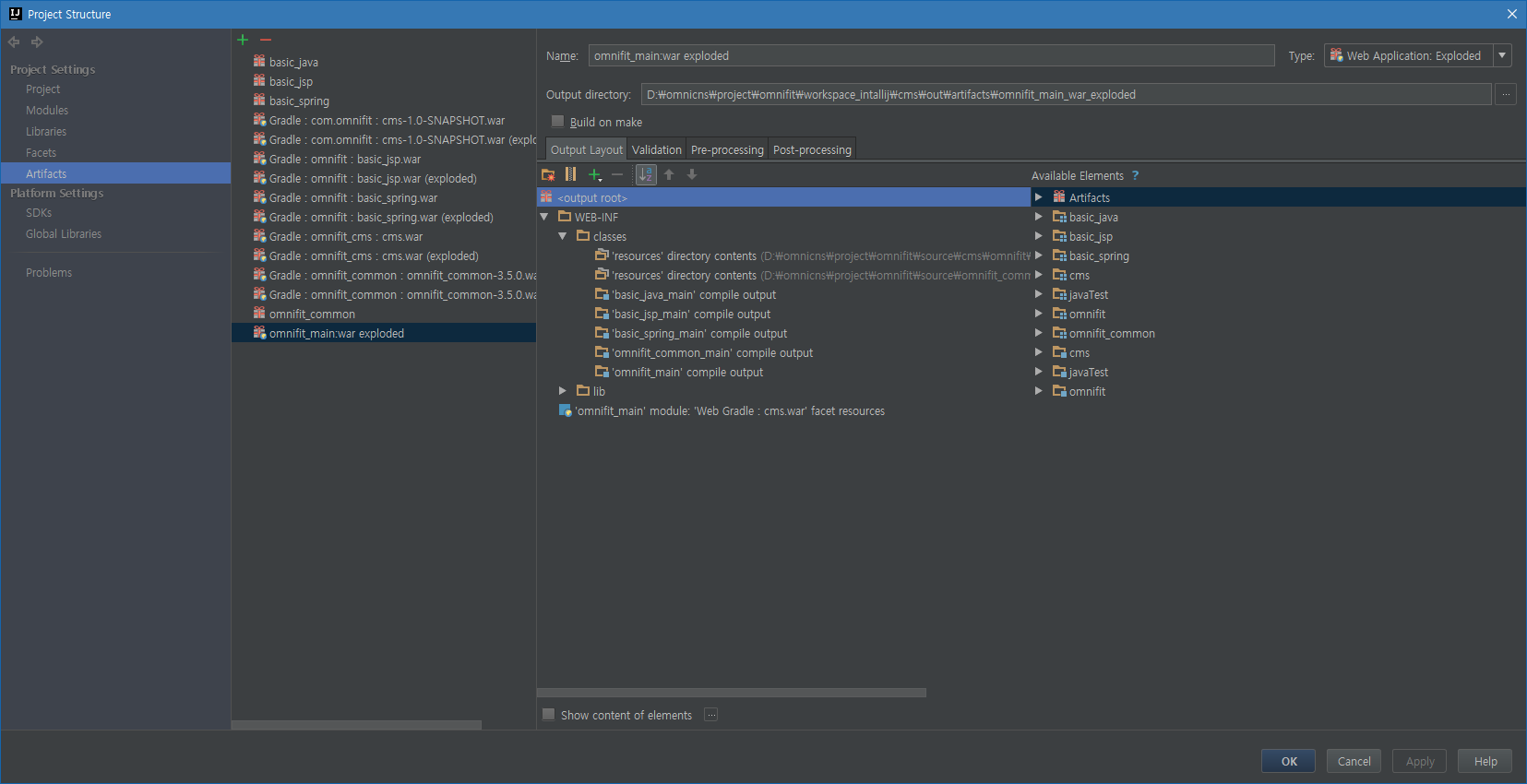
tomcat실행할때 Artifacts 를 선택하게되는데 이것을 exploded로 선택(war풀린 폴더자체로 실행)

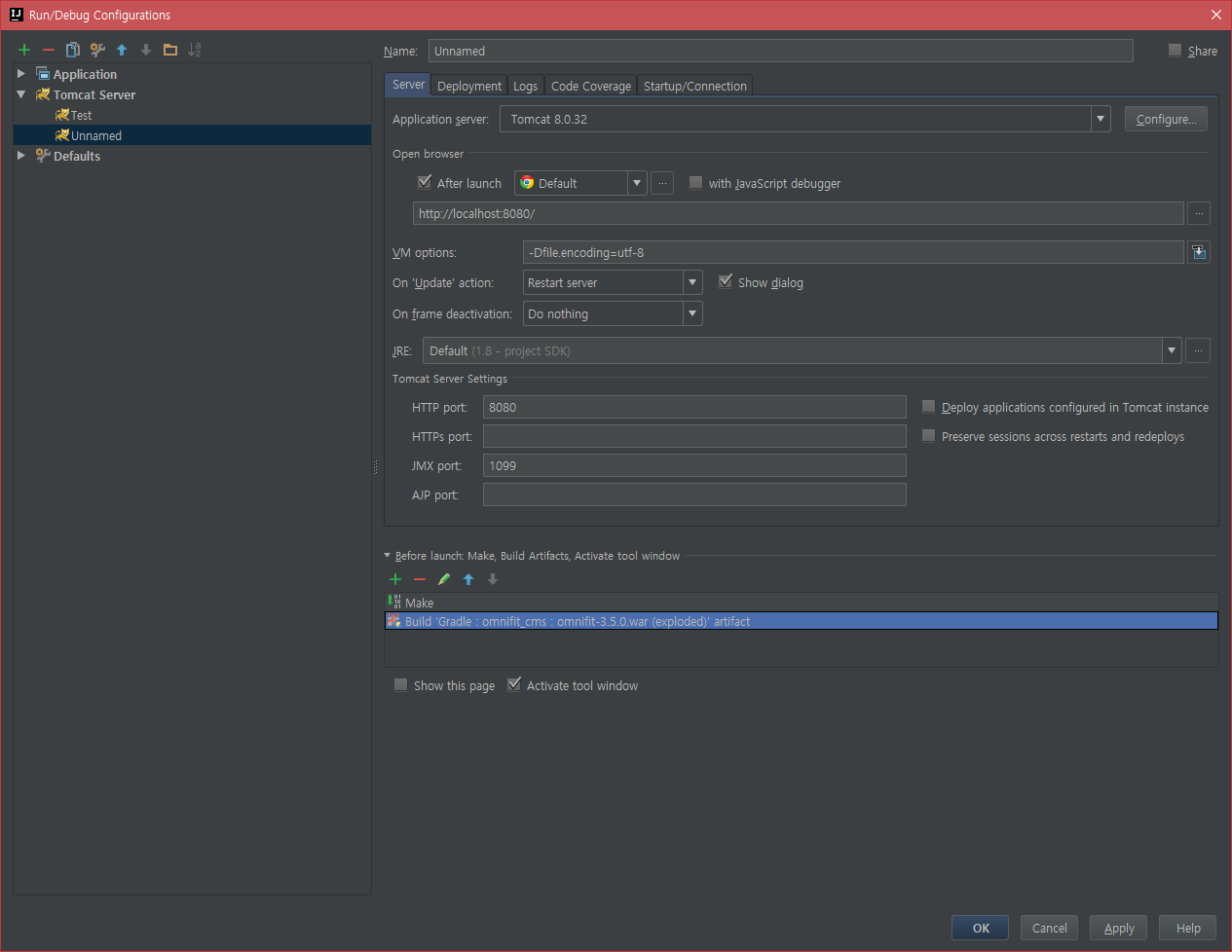
빌디패스 걸거 있으면 Artifacts설정가서 타겟Artifact쪽에 걸어줄거 걸어주고

tomcat쪽에서 그 Artifacts를 선택후 실행시키면된다

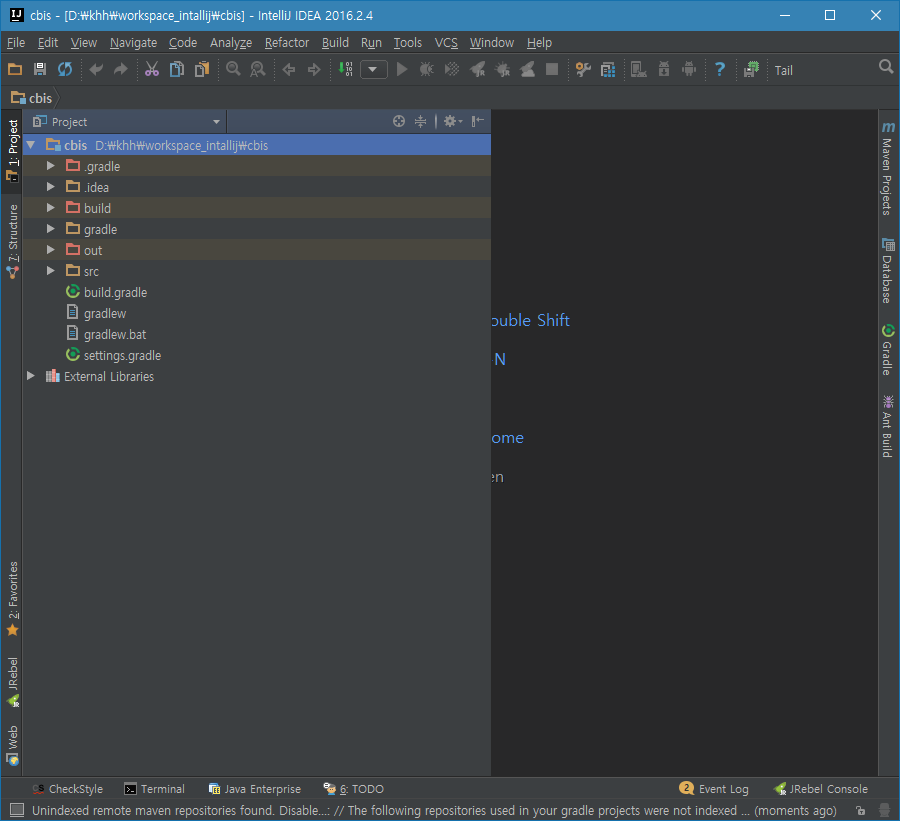
그리고 class 수정후 리컴파일한다음 Build Artifact를 해주면된다..삽질겁나했네 아오~



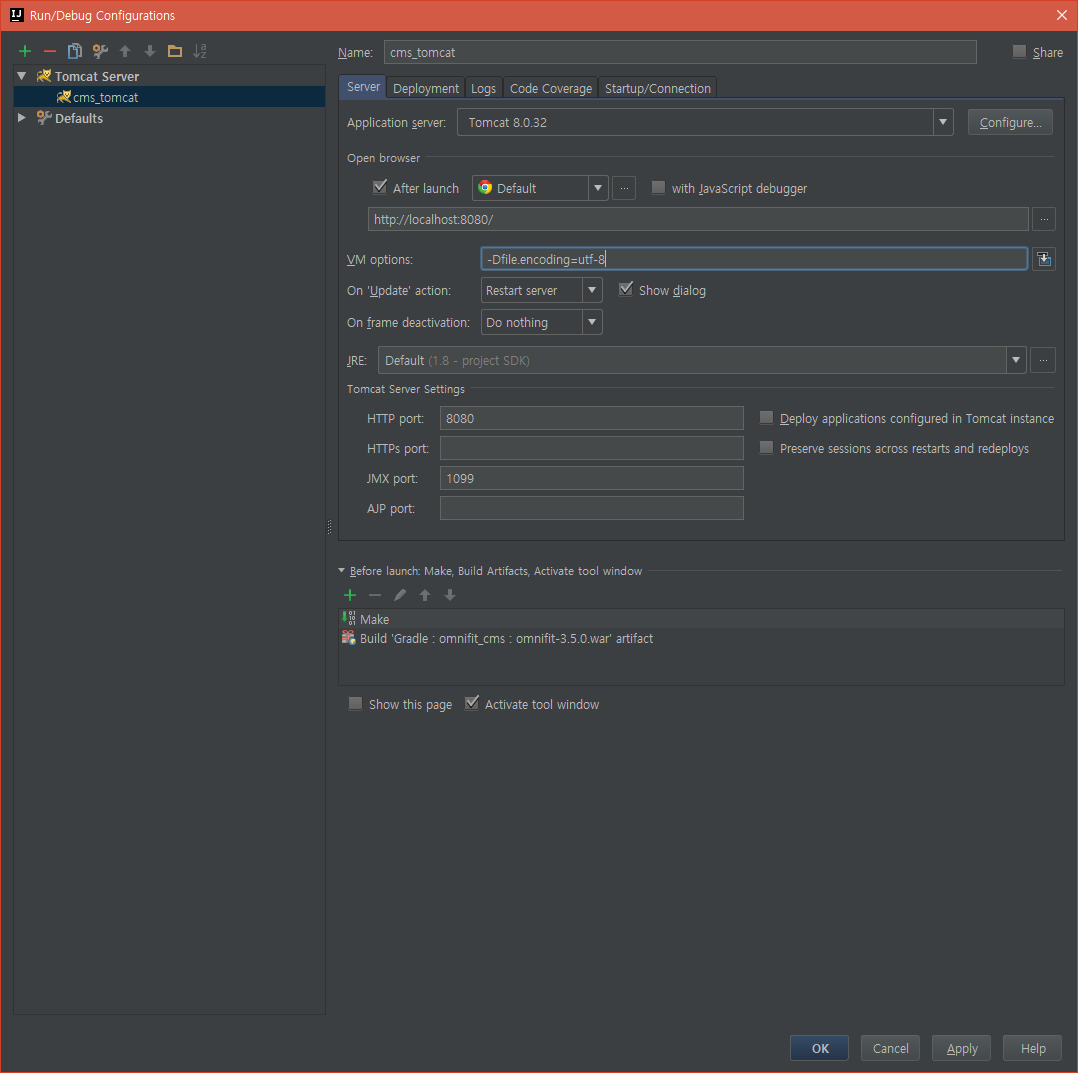




cbis 프로젝트 셋팅 해보자 intellij



1. import module
   1. import cbis source gradle



3. 근데 Tomcat Console 에서 한글이 제대로 나오지 않는 문제가 있어 톰캣 VM Option으로 -Dfile.encoding=utf-8로 줌

**소스 바로 반영시키는 Srpign loaded 사용하면된다 (서버재기동 안해도됨)**

**화면에 바로 live적으로 적용된거 확인시키게하고싶을때**

**dependencies {**

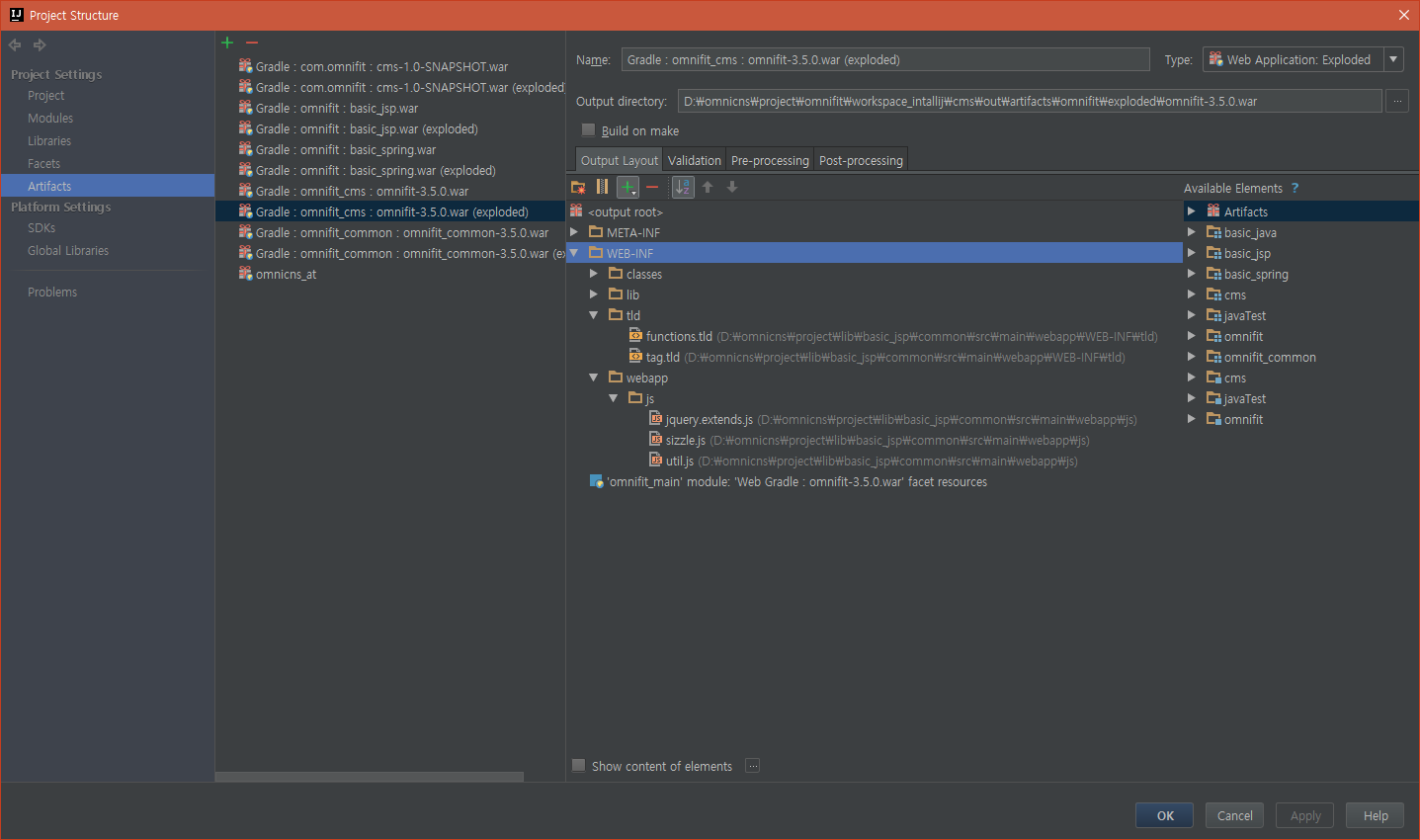
**compile("org.springframework.boot:spring-boot-devtools")**

**}**

**걸어주고 크롬 웹스토어에서 liveReload 설치하고 enable걸어주면 바로 확인가능하다.**

**Artifacts**

**링크소스같은거**



**여기서 묵을때 어떤거 참조해서 넣을건지 +눌러서 등록시킨다**

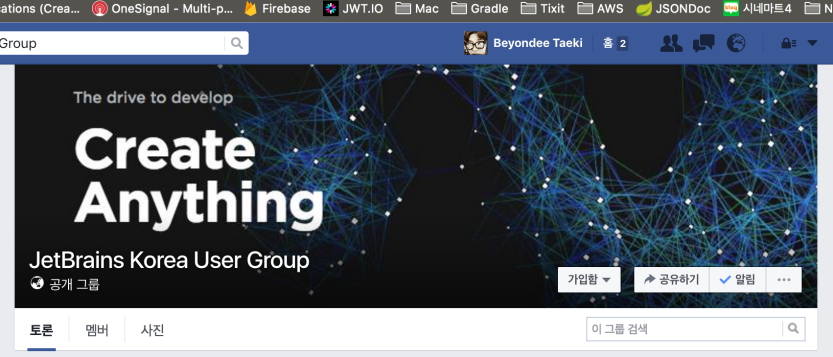
**gradle설정**

**.**

# 빠른 생산성을 위해서 재기동 없이 서버 Hot swapping 하는 방법 (IntelliJ & Eclipse)

***Facebook JetBrains Korea User Group***

### <https://www.facebook.com/groups/1821879151380372/>

새롭게 JetBrains Facebook User Group을 만들었습니다.  
많이 오셔서 공유 해요  


[](https://beyondj2ee.files.wordpress.com/2015/10/code.jpg)

(source : http://tomselt.files.wordpress.com/2013/05/codeswitching1.jpg?w=1200)

PIKICAST 입사 후에 오랫만에 블로깅을 하네요.  
정말 블로깅 할 내용들이 많이 있지만 점점 시니어로써 성장 하면서  
시간을 내기가 쉽지는 않네요..  
그래도 틈틈히 실무에 도움이 될만한 것들은 공유를 해볼려고 합니다.

이번 포스팅 내용은 “***빠른 생산성을 위해서 재기동 없이 톰캣 Hot swapping 하는 방법*** ”  
입니다.

웹/REST 서버를 개발을 할때 소스를 수정하고 로컬에 톰캣 서버를 계속 restart  하는 것 입니다.

특히나 Spring을 사용하면 더 실감을 하시 겠죠.. (스프링 무겁다는 이유가 이런가 아닐런지..)

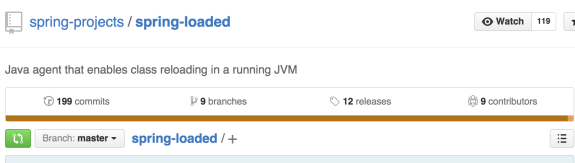
아마 생산성에 관심 있으신분들은 한번쯤은 research 해봤지 않았을까..

**1. What Kind of**

[](https://beyondj2ee.files.wordpress.com/2015/10/jrebel.png)(<http://zeroturnaround.com/software/jrebel/>)

JRebel 매우 유명한 툴 입니다. 사용법은 PC에 설치를 하고 IDE에 설정만 하면 쉽게  
적용이 가능하고 우리가 원하는 restart 없이 개발이 가능 합니다.  
한가지 안쉬운 점이 “**상용버전**” 즉, 돈을 내야 한다는 거죠.. ㅋㅋ  
하지만 학생들은 “**커뮤니티**” 버전을 제공하기 때문에 무료로 사용이 가능  
합니다.

“**상용을 뛰어넘는 오픈소스는 없다**“가 개인적인 생각이라서 분명 안정성이나  
여러 측면에서 좋지 않을까 하네요..

[](https://beyondj2ee.files.wordpress.com/2015/10/springloaded.png)  
(<https://github.com/spring-projects/spring-loaded>)

바로 오늘 말씀 드릴 “**Spring Loaded**” 입니다.  
SpringBoot(<http://projects.spring.io/spring-boot/>)의 Document에서  
“**Hot Swapping**” 를 보면 언급이 되어 있습니다.

무엇보다 “**Free Free Free   무료 무료!!**” 라는 점.

(개인적으로 Spring Boot을 사용하지않고 필요한 기능만  
컴팩트하게 프레임웍을 구성하는걸 좋아해서.. Boot을 잘 안써요 )

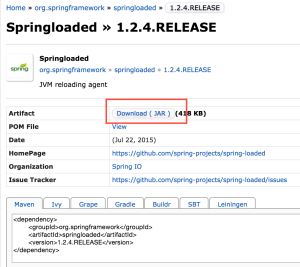
Tomcat뿐만 아니라 Daemon에서도 가능하다고 합니다. (테스트 해보니.. Daemon  
은 nice 하지 않음.)

Spring Boot에서는 메이븐 또는 Gradle에서 설정 하고 간단하게 “**Run**” 하면 됩니다.  
(참고 사이트 : http://reiphiel.tistory.com/89)

하지만 우리가 현재 운영중이고 개발하는 시스템들은 거의 “**Spring Boot**“으로 되어 있지  
않습니다.

그래서 “**Spring Boot**“을 사용하지 않더라도 “**Hot Swapping 하는 방법**“이 훨씬  
유용한 Tip이라고 생각 하네요

**2. Use IntelliJ**

[](https://beyondj2ee.files.wordpress.com/2015/10/springload.png)  
(<http://mvnrepository.com/artifact/org.springframework/springloaded/1.2.4.RELEASE>)

먼저 “spring-loaded” 라이브러리(.jar)를 다운로드 받으세요..  
(최신 버전은 각자 확인 하세요)

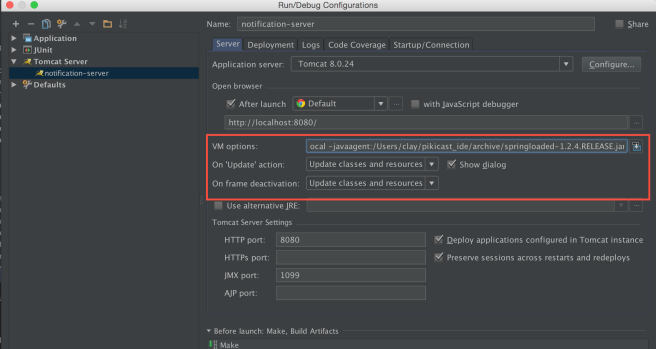
저 같은 경우는 “/Users/clay/pikicast\_ide/archive/springloaded-1.2.4.RELEASE.jar”  
에 다운로드를 받았습니다. (다운로드 받는 경로 기억 하세요)

그리고 IntelliJ에서 톰캣 VM option에 아래와 같이 추가를 합니다.

|  |  |
| --- | --- |
| 1 | -javaagent:/Users/clay/pikicast\_ide/archive/springloaded-1.2.4.RELEASE.jar   -noverify |

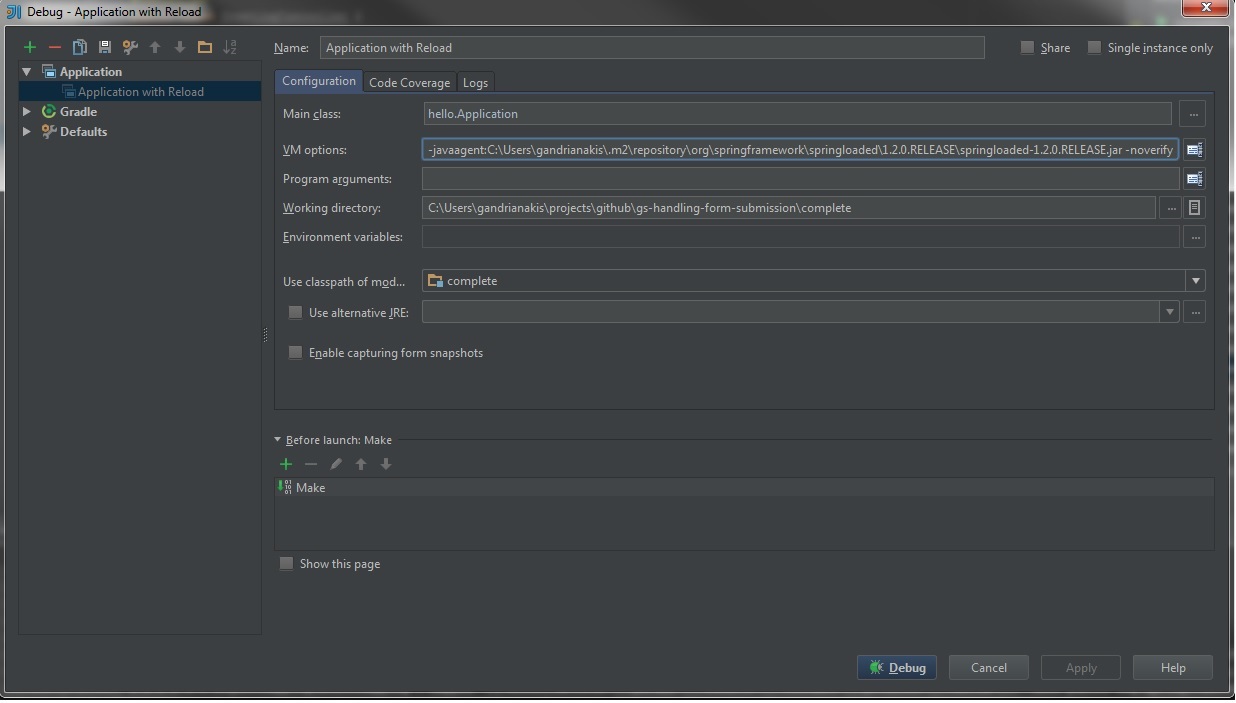
(본인 경로로 설정 하세요!!!)

-javaagent:/path/to/jar/springloaded.jar -noverify

[](https://beyondj2ee.files.wordpress.com/2015/10/loaded1.png)

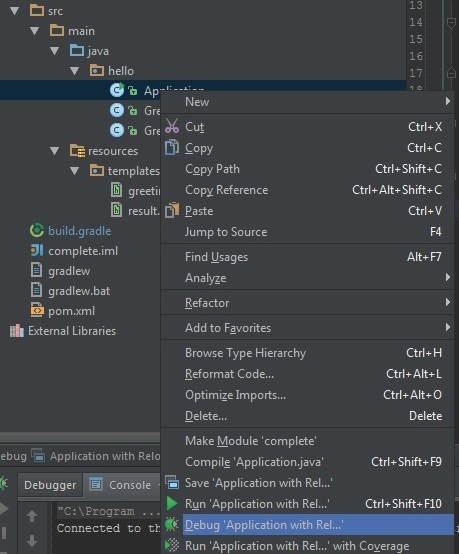
설정이 모두 완료가 되었습니다.  
서버를 기동하고 아래와 같이 Controller에서 서버 기동 후 소스를 수정 하면  
restart 없이 바로 반영 되는 것을 확인 할수 있습니다.

1. Configure the run configuration in IntelliJ to use the SpringLoaded agent. This is easy to do and an example is shown in the following screenshot:



Notice how I have added a VM Option: -javaagent:/path/to/springloaded-${version}.jar -noverify (which you can download [here](https://github.com/spring-projects/spring-loaded))

1. Debug using Right Click -> Debug like the following screenshot:



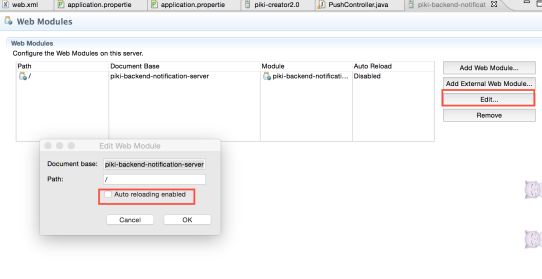
1. Everytime you make a change and want to reload it, just compile the project. The default shortcut is Cntrl+F9, but you can also access it from the menu Build -> Make Project

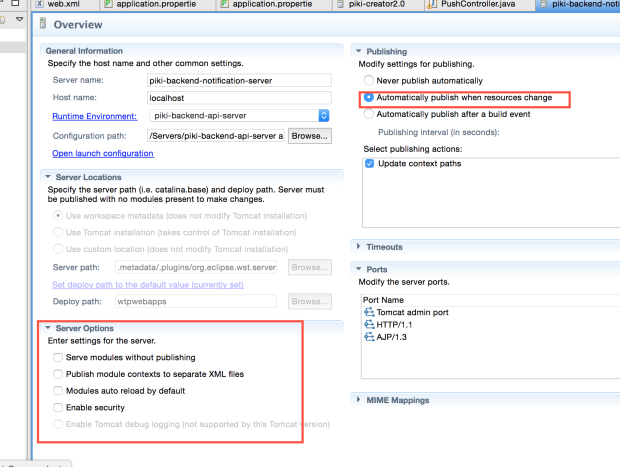
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | @RequestMapping(value = "pikicast/v1.0/hello", method = RequestMethod.POST)    public void hello() throws Exception {        log.info("#############&gt;1");    }    출력 : INFO - #############&gt;1    @RequestMapping(value = "pikicast/v1.0/hello", method = RequestMethod.POST)    public void hello() throws Exception {        log.info("#############&gt;2");    }    출력 : INFO - #############&gt;2 |

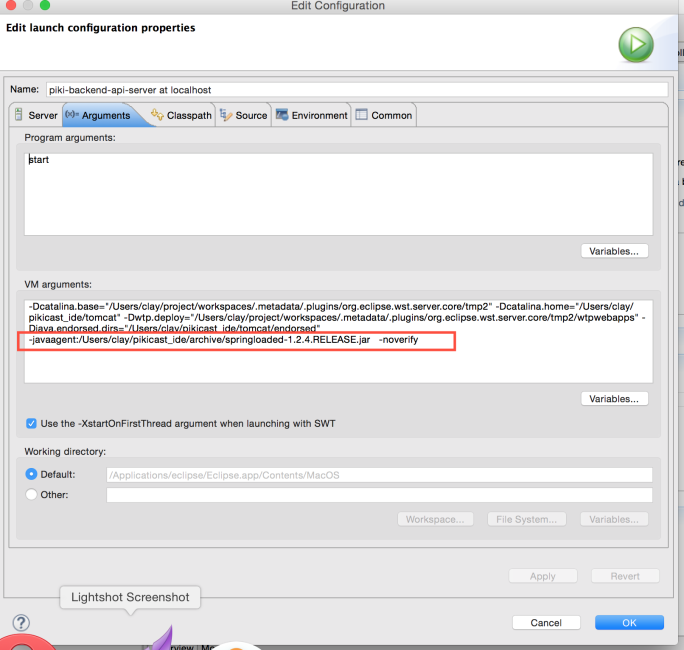
**3. Use Eclipse**

IntelliJ 와 동일하게 다운로드 받습니다.

“**Server**” – “**Module**” – ” **Auto reloading enabled**“를 해제  합니다.

[](https://beyondj2ee.files.wordpress.com/2015/10/e1.png)  
“**Server Options**” 모두 해제, “**Publishing**“은 “**Automatically publish when resources change**” 선택

[](https://beyondj2ee.files.wordpress.com/2015/10/e2.png)  
마지막으로 “**VM Option**“을 추가 합니다.

[](https://beyondj2ee.files.wordpress.com/2015/10/e3.png)  
모든 설정이 완료가 되었습니다. IntelliJ 부분에서 설명 했듯이 서버 기동 후  
소스를 수정하면 restart 없이 소스가 런타임으로 반영 되는 것을 확인 할수 있습니다.

**Conclusion**

실제로 써보면 생각보다 즉시 반영 되지 않고 약간의 지연 시간이 있고 로딩도 합니다.  
하지만 확실히 서버를 restart 하는 것 보다 개발 비용 & 생산성이 좋아 지더군요..  
특히 최근에는 front 와 backend  개발이 분리가 되어서 REST API을 주로 개발하다보니  
정말 많은 도움을 받고 있습니다.

대부분의  환경이 “Spring Boot”을 사용하지 않기 때문에 SI부터 Startup까지  
Spring 과 IDE (IntelliJ or Eclipse) 사용하는 모든 자바 개발자분들에게  
조금이나마 도움이 되는 포스팅이길 바랍니다.