스프링 부트 개발환경 셋팅

실습1) 스프링 부트 개발환경 준비

- 이제 스프링 프레임워크 개발환경 자동화 툴인 STS를 설치해보도록 한다.

※ 진행할 것 : Installing the **OpenJDK**, **JetBrains Toolbox**, **IntelliJ IDEA** and Setting the **Spring boot on IntelliJ**

1. Install the Open JDK (version 1.8)

1.1. https://github.com/ojdkbuild/ojdkbuild <- URL로 접속

1.2. java-1.8.0-openjdk 설치 (빨간박스)


README.md

Azure Pipelines succeeded

Downloads for Windows x86_64

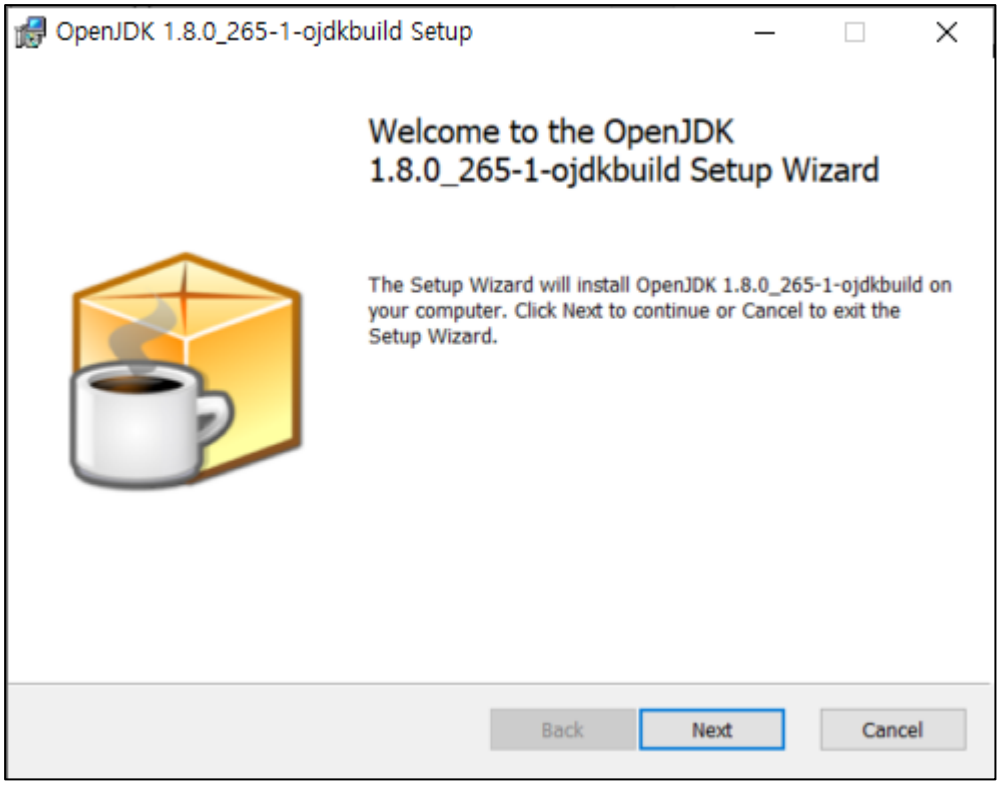
- **1.8.0_265-1** (LTS, supported until May 2026, announcement)
  - java-1.8.0-openjdk-1.8.0.265-1.b01.ojdkbuild.windows.x86_64.zip (sha256)
  - java-1.8.0-openjdk-1.8.0.265-1.b01.ojdkbuild.windows.x86_64.msi (sha256)
- **11.0.8-1** (LTS, supported until October 2024, announcement)
  - java-11-openjdk-11.0.8.10-1.windows.ojdkbuild.x86_64.zip (sha256)
  - java-11-openjdk-11.0.8.10-1.windows.ojdkbuild.x86_64.msi (sha256)
- **13.0.4-1** (announcement)
  - java-13-openjdk-13.0.4.8-1.windows.ojdkbuild.x86_64.zip (sha256)
  - java-13-openjdk-13.0.4.8-1.windows.ojdkbuild.x86_64.msi (sha256)
- **14.0.2-1** (announcement)
  - java-14-openjdk-14.0.2.12-1.windows.ojdkbuild.x86_64.zip (sha256)
  - java-14-openjdk-14.0.2.12-1.windows.ojdkbuild.x86_64.msi (sha256)

Note: obsolete downloads are available at the bottom of this page

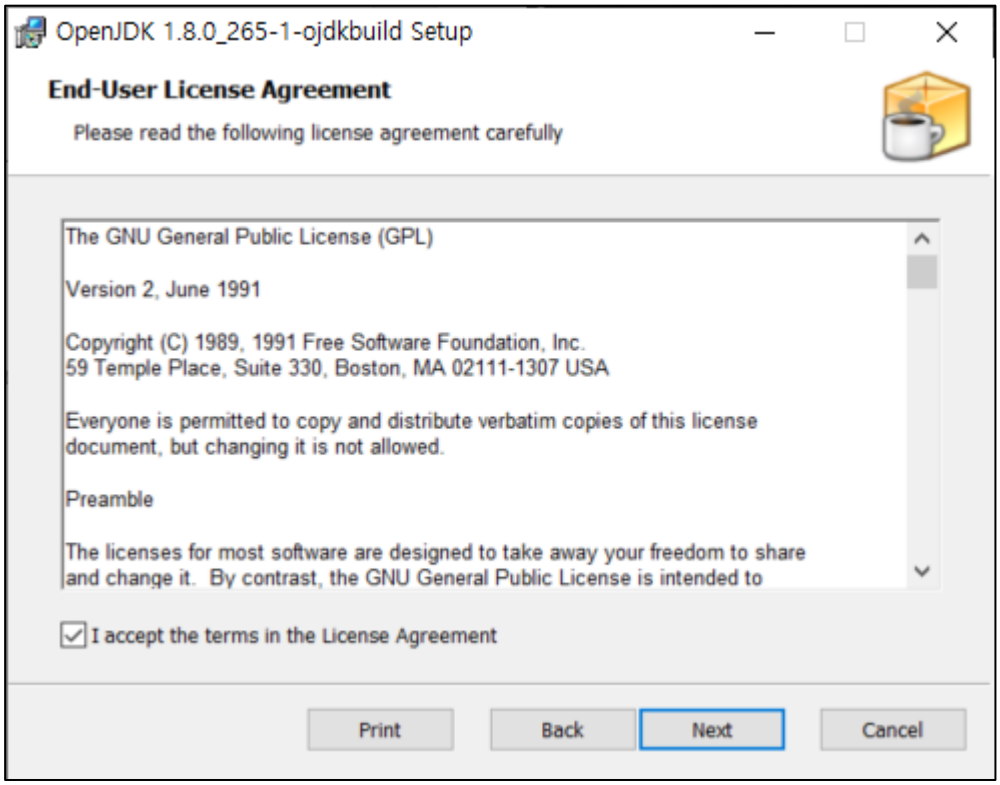
(.zip은 수동설치 / .msi는 자동설치.)

1.3. java-1.8.0-openjdk*.msi 실행

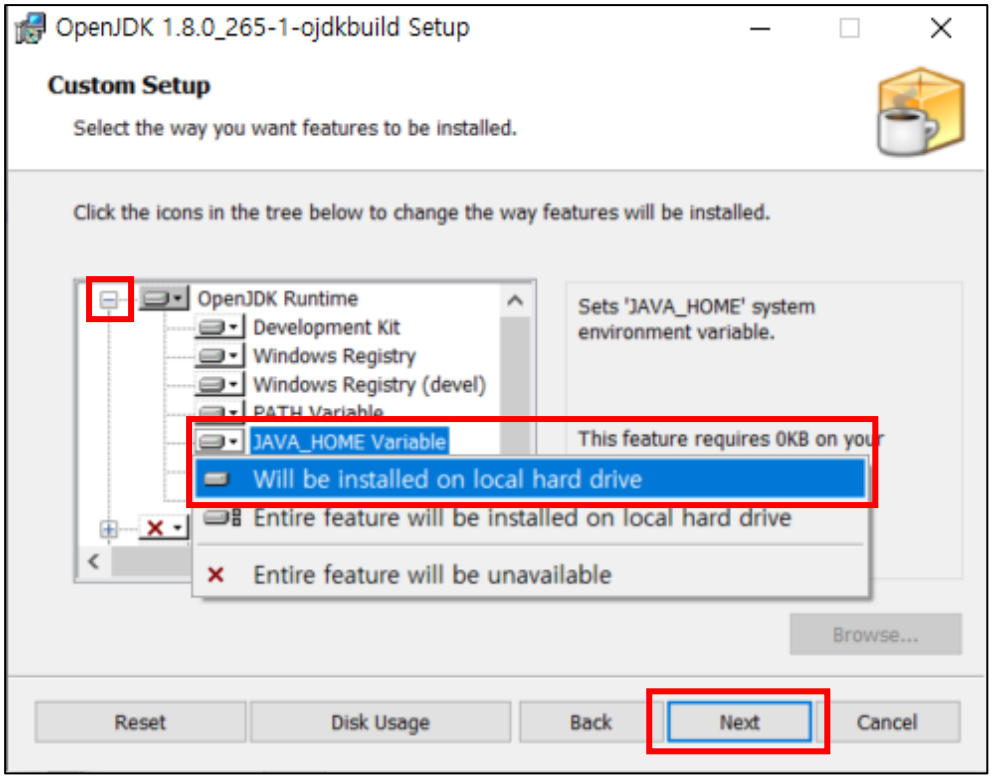
1.3.1. Next 클릭



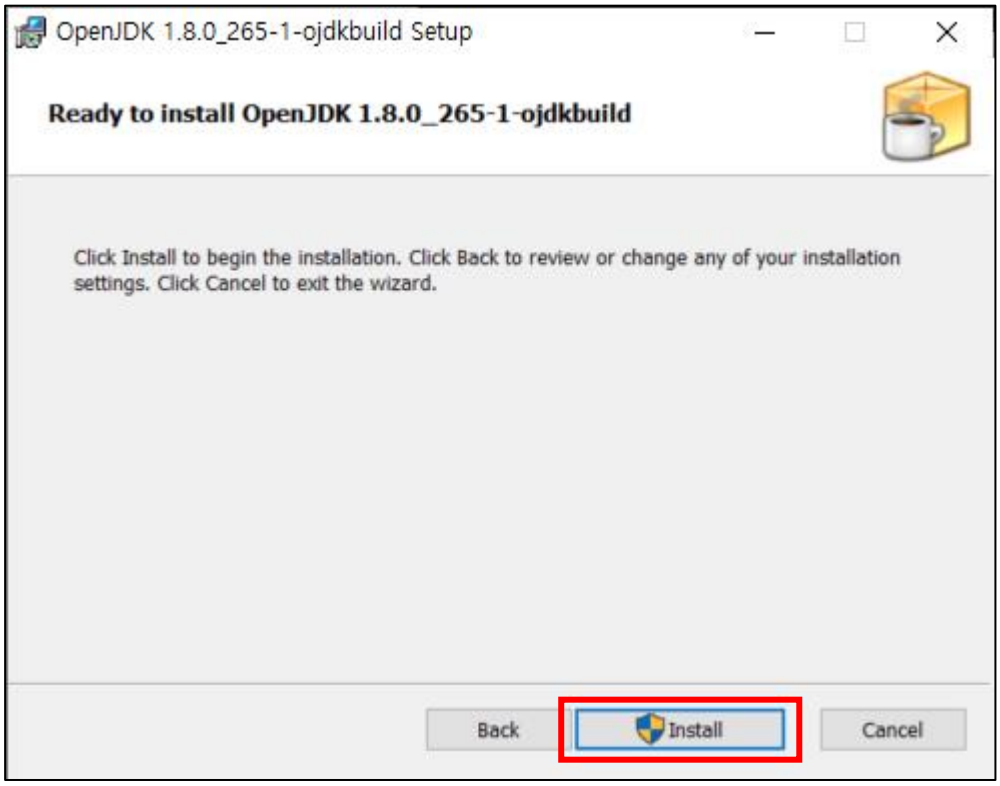
1.3.2. Check 후 Next 클릭



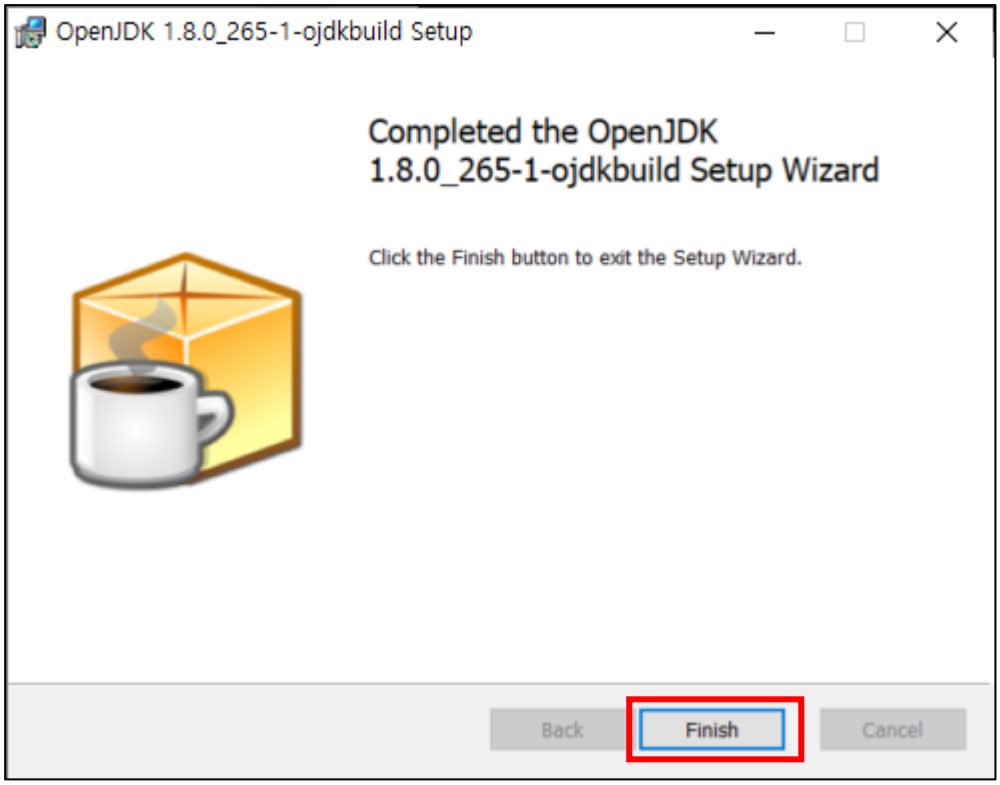
1.3.3. JAVA_HOME 환경변수 셋팅 후 Next 클릭



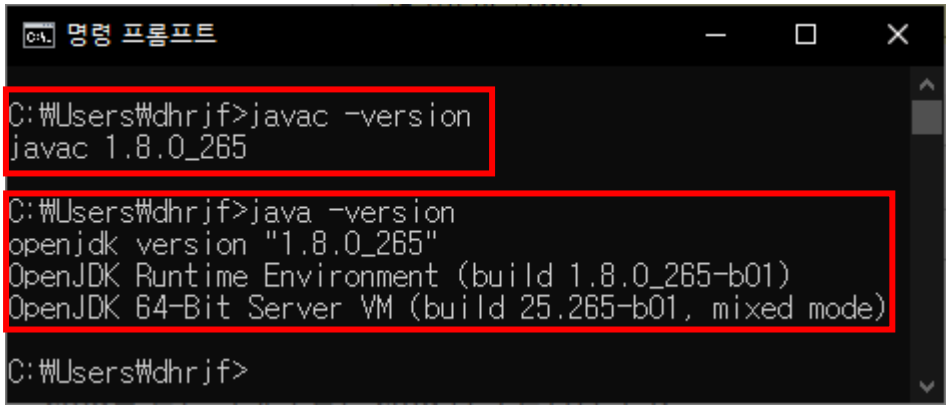
1.3.4. Install 클릭



1.3.5. Install 완료 후 Finish 클릭



1.3.6. OpenJDK 1.8이 올바르게 설치되었는지 명령 프롬프트로 확인

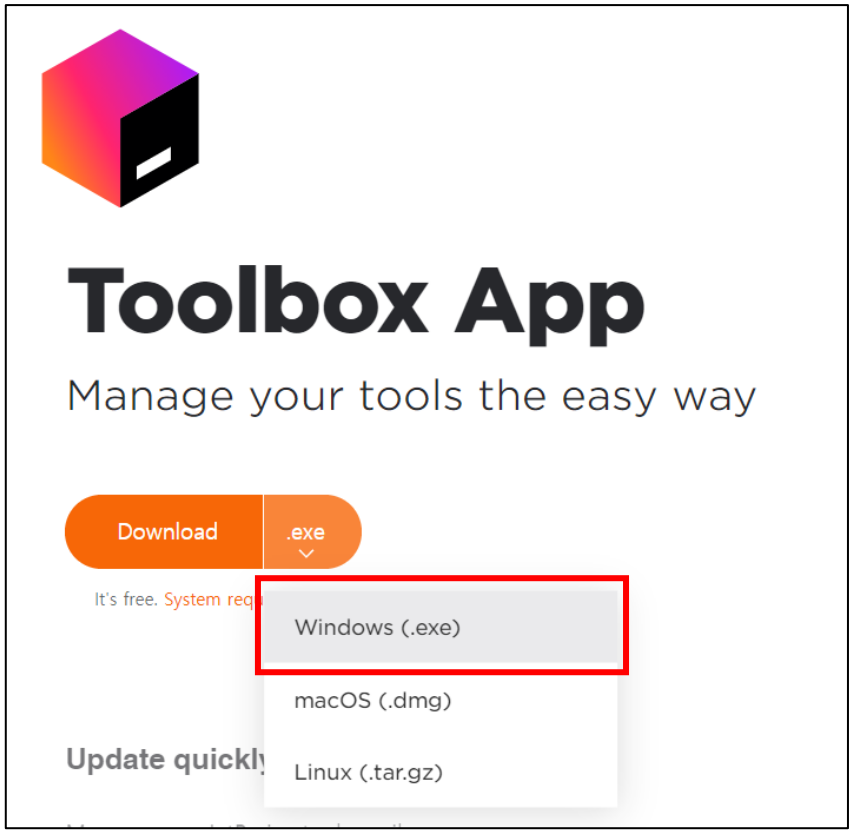


(javac -version and java -version을 입력 시 위 사진과 유사하게 출력되어야 정상 설치된 것이다.)

2. Install the Jetbrain Toolbox

2.1. https://www.jetbrains.com/toolbox-app/ <- URL로 접속

2.2. 사용자 운영체제 환경에 맞는 Toolbox 설치



(Toolbox는 버전 관리와 JVM옵션 등을 조정할 수 있다.)

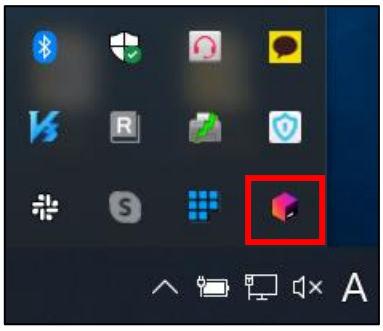
2.3. 설치된 jetbrains-toolbox-....exe실행

2.4. 설치 클릭

2.5. Toolbox 설치완료 후 마침 클릭

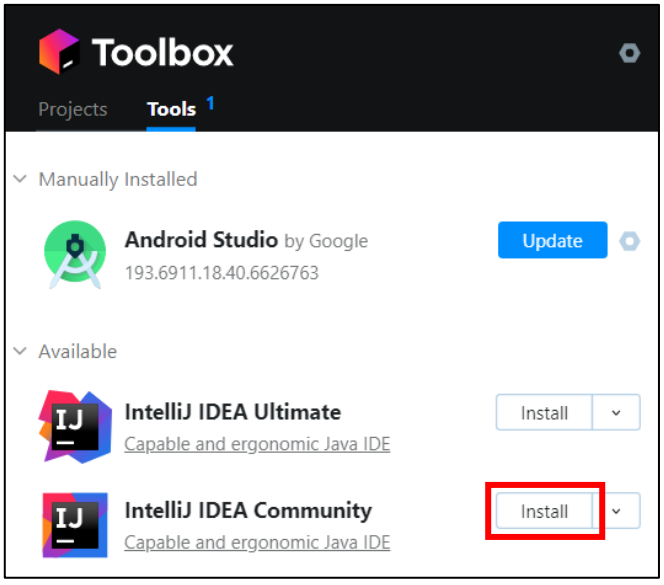
3. Install the IntelliJ (Community)

3.1. Toolbox 실행

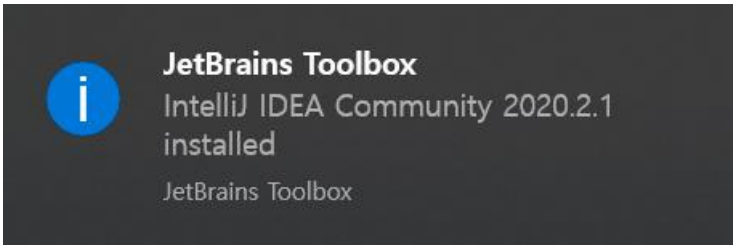


(설치가 완료되었으면 화면 오른쪽 아래에 Toolbox 아이콘이 생성되었을 것이다.)
(해당 아이콘을 더블 클릭하여 Toolbox를 실행하도록 한다.)

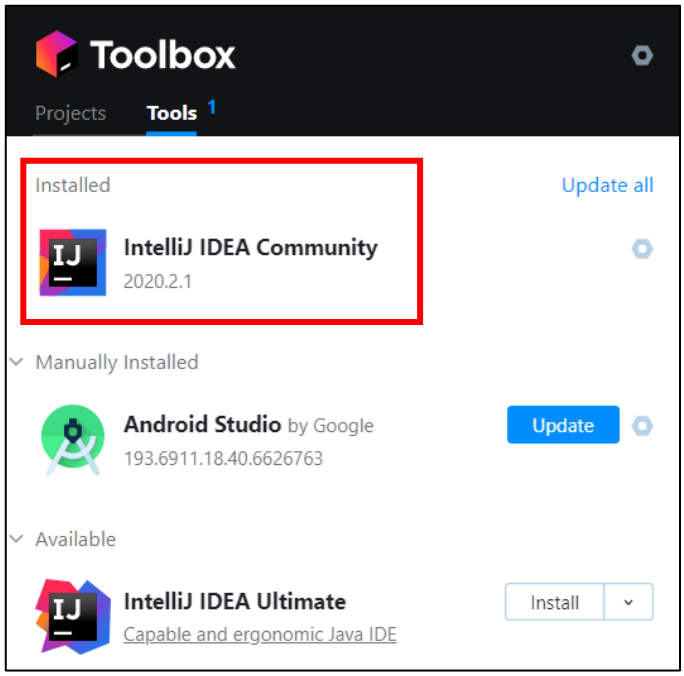
3.2. IntelliJ IDEA Community 설치



3.3. IntelliJ 설치가 완료되었을 시 윈도우10에서는 오른쪽 하단에 아래와 같이 알림이 뜬다.



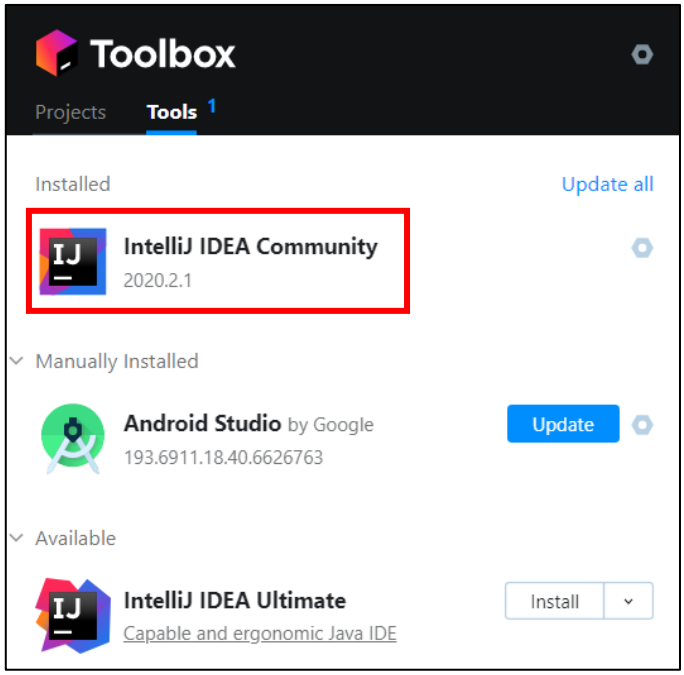
3.4. IntelliJ가 설치되었는지 Toolbox로 확인



(Installed에 IntelliJ IDEA Community가 등록된 것을 확인할 수 있다.)

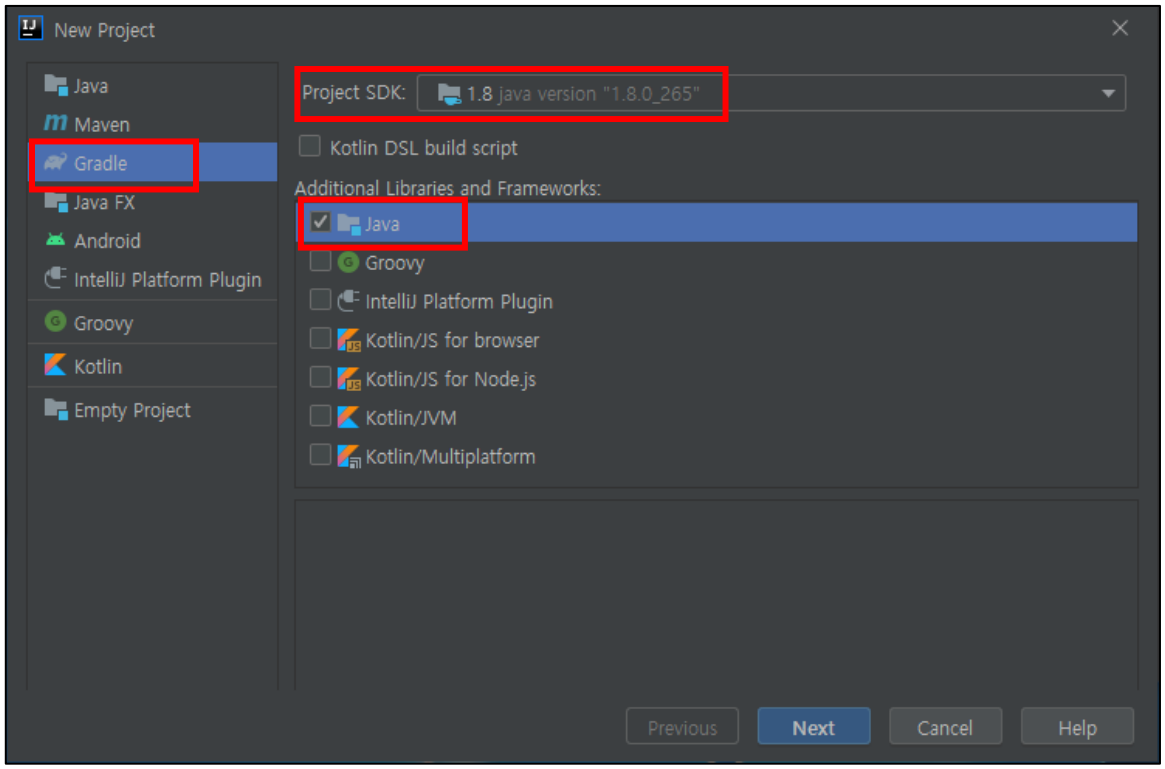
4. Spring Boot 개발환경 설정

4.1. Toolbox에서 IntelliJ 실행



4.2. New Project 클릭

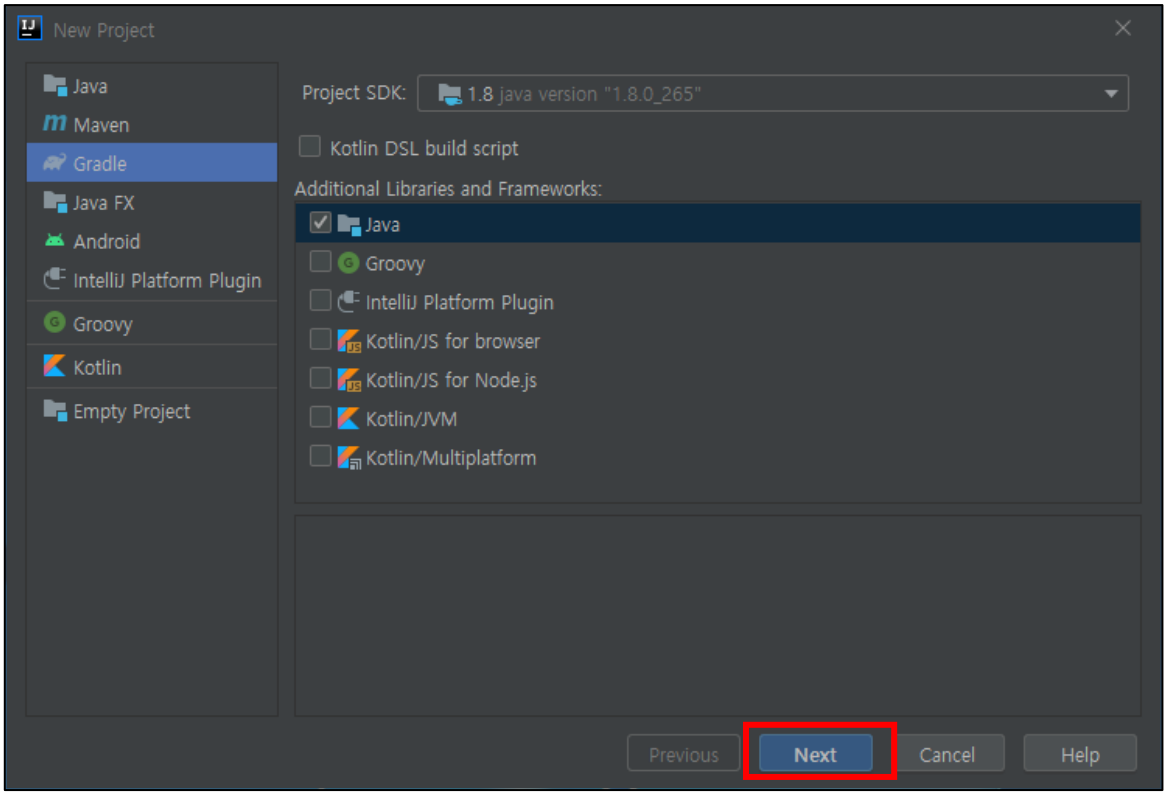
4.3. Gradle – Java



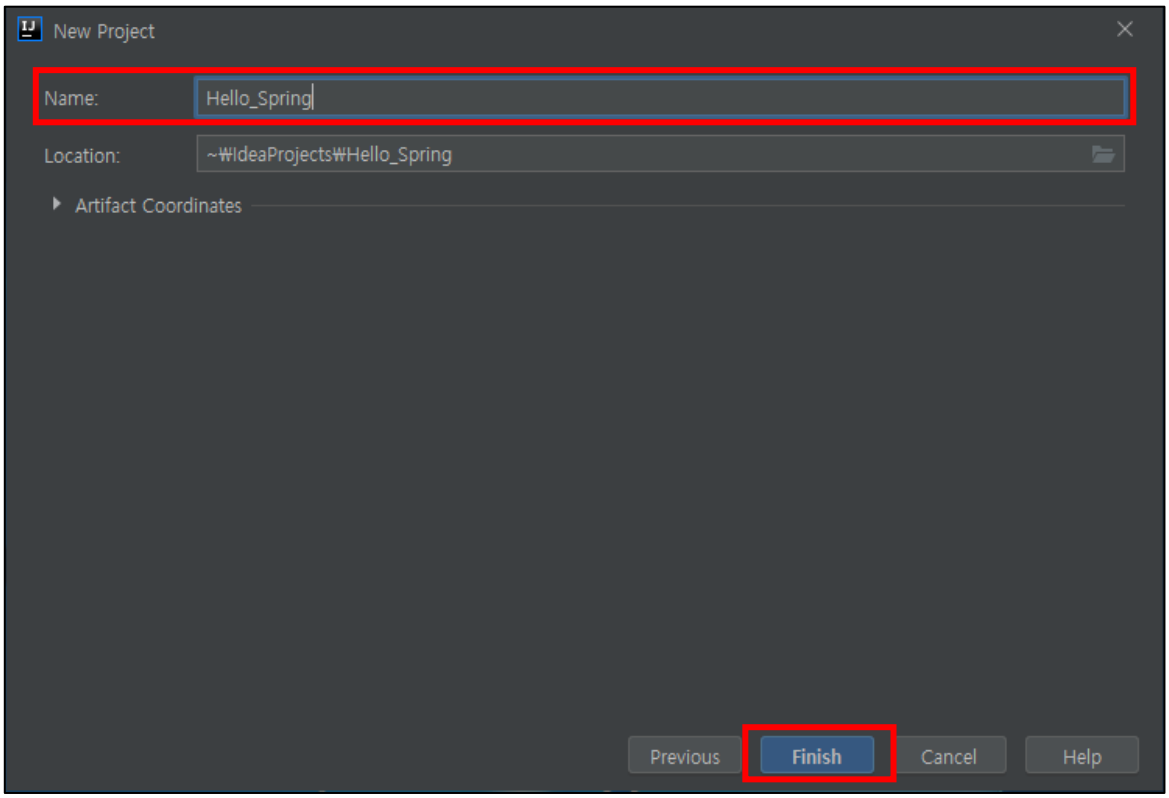
(!! Java가 체크되었는지 확인)

(!! OpenJDK 버전 확인)

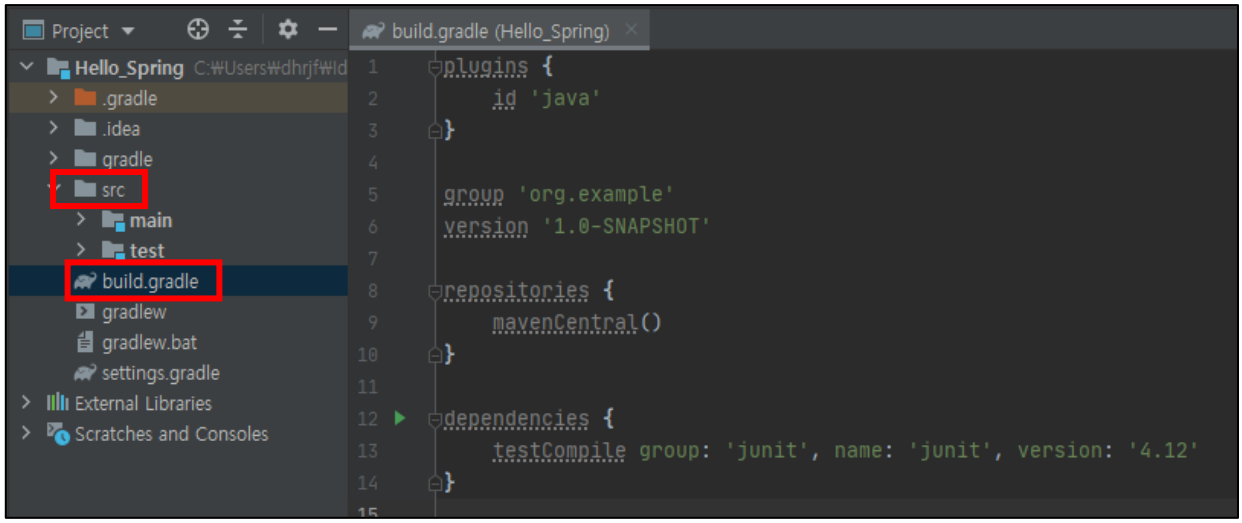
4.4. Next 클릭



4.5. 프로젝트 이름 설정 후 Finish 클릭 (ex. Hello_Spring)



4.6. 왼쪽 프로젝트 목록에서 src/build.gradle 더블클릭



4.7. build.gradle에 빨간 박스 부분을 입력

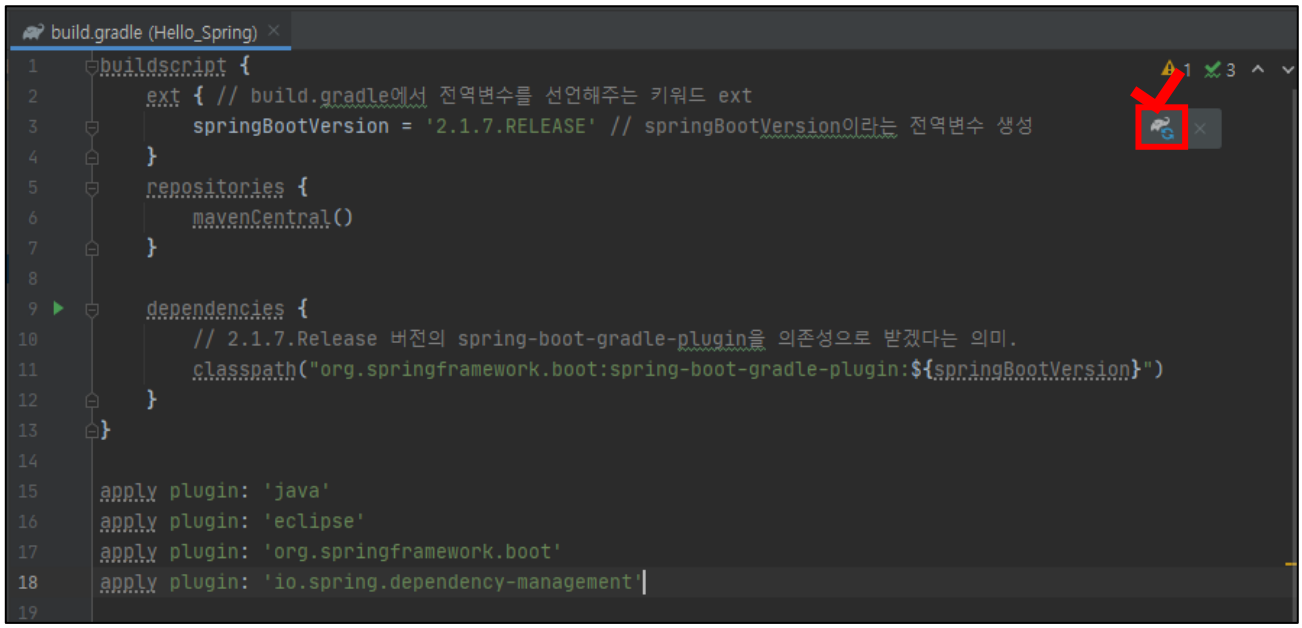
```
buildscript {
    ext {
        springBootVersion = '2.1.7.RELEASE'
    }
    repositories {
        mavenCentral()
    }
    dependencies {
        classpath("org.springframework.boot:spring-boot-gradle-plugin:${springBootVersion}")
    }
}

apply plugin: 'java'
apply plugin: 'eclipse'
apply plugin: 'org.springframework.boot'
apply plugin: 'io.spring.dependency-management'
```

```
plugins {
    id 'java'
}
```

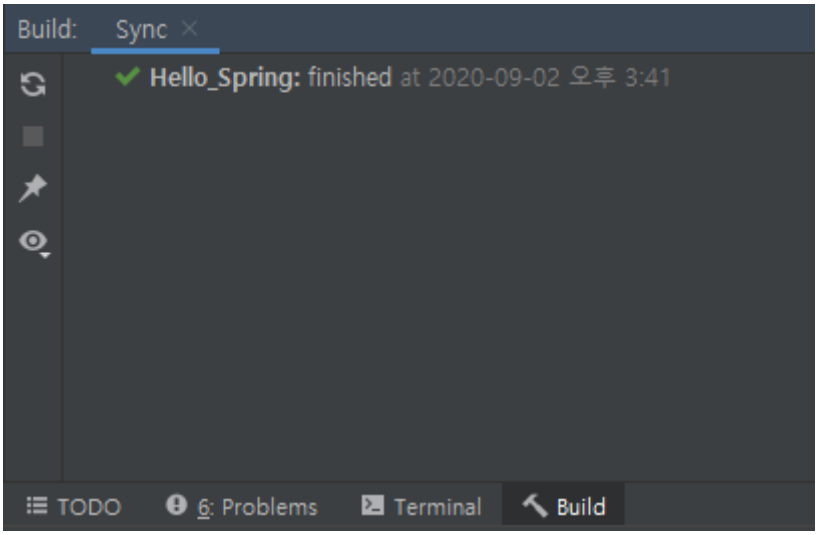
→ 파란박스 부분 삭제

```
group 'org.example'
version '1.0-SNAPSHOT'

sourceCompatibility = 1.8

repositories {
    mavenCentral()
}

dependencies {
    compile('org.springframework.boot:spring-boot-starter-web')
    testCompile('org.springframework.boot:spring-boot-starter-test')

    testCompile group: 'junit', name: 'junit', version: '4.12'
}
```

4.8. build.gradle의 변경된 사항 적용 => 오른쪽 상단 코끼리 아이콘 클릭



4.9. build.gradle이 정상적으로 등록되었는지 확인



(초록색 체크 표시 + finished 가 뜨면 정상적으로 등록된 것이다.)

✓ 4.9까지 완료하였다면 Spring boot 개발환경 셋팅이 완료된 것이다.

참고문헌

- 이동욱 (2019). 스프링 부트와 AWS로 혼자 구현하는 웹 서비스. 20-39.