

Gradle 설치

그루비(Groovy)언어를 기반으로 한 빌드 도구

거의 모든 유형의 소프트웨어를 빌드할 수 있을 만큼 유연한 오픈 소스 빌드 자동화 도구

주로 SpringBoot와 Android에서 사용되고 있음.

프로젝트에는 빌드 스크립트가 포함되며, 일반적으로 빌드 스크립트는 프로젝트의 루트 디렉토리에 있는 `build.gradle` 또는 `build.gradle.kts`라는 이름의 파일임

빌드 스크립트는 해당 프로젝트에 대한 태스크, 의존성, 플러그인 및 기타 구성을 정의함.

1. Windows

<https://gradle.org/releases/> 에서 다운로드 --> `binary-only`

다운로드 받은 파일의 압축을 적절한 위치에 해제한다. (`C:\gradle`)

`Path`환경 변수에 gradle의 bin 폴더 경로를 추가한다.

2. Mac

```
$ brew install gradle
```

3. 설치 확인 (Windows, mac 공통)

```
$ gradle -v
```

#02. 프로젝트 생성

1. 자바 프로젝트가 생성될 폴더를 미리 만들어 둔다 --> `30-Lombok(Gradle)`
2. `ctrl + Shift + P: Java: Create Java Project` 선택
3. `Gradle` 선택
4. 미리 만들어둔 프로젝트 폴더를 지정한다. --> `30-Lombok(Gradle)`
5. `Groovy` 선택
6. 프로그램 이름 입력. `lombok_ex`
7. 생성된 프로젝트를 VSCode로 연다.
8. 패키지 이름을 변경한다.
 - 지정된 폴더 이름으로 패키지가 지정되기 때문에 패키지 이름을 변경할 필요가 있다.
 - `30-Lombok(Gradle)`의 경우
 - 숫자로 시작할 수 없으므로 `30-`는 제외됨
 - 패키지 이름은 모두 소문자가 관례이므로 소문자로 변경됨
 - `Lombok(Gradle)`에서 괄호를 표시할 수 없으므로 `lombok.gradle`로 지정됨
 - 여기서는 `kr.hossam.lombok_ex`로 변경한다.

#03. 라이브러리 참조하기

1. 필요한 라이브러리 검색

<https://mvnrepository.com/>에서 필요한 라이브러리를 검색한다.

여기서는 **gson**으로 검색

검색 결과 중에서 **com.google.code.gson » gson**의 최신 버전(**2.11.0**) 세부 페이지로 진입한다.

2. 라이브러리 참조 코드 복사

화면 중앙의 탭에서 **Gradle (Short)** 페이지의 코드를 복사한다.

```
// https://mvnrepository.com/artifact/com.google.code.gson/gson
implementation 'com.google.code.gson:gson:2.11.0'
```

3. /app/build.gradle 파일 편집

파일 중앙의 **dependencies { ... }** 안에 복사한 코드를 삽입한다.

같은 방법으로 **Lombok**으로 검색하여 **org.projectlombok » lombok**의 최신버전(**1.8.34**)의 코드를 복사하여 삽입한다.

```
// https://mvnrepository.com/artifact/org.projectlombok/lombok
compileOnly 'org.projectlombok:lombok:1.18.34'
```

Lombok의 경우 아래의 코드도 직접 입력해서 추가해야 한다.

```
// 이 구문은 직접 추가한다.
annotationProcessor 'org.projectlombok:lombok:1.18.34'
```

4. build.gradle의 최종 결과

```
plugins {
    // Apply the application plugin to add support for building a CLI
    application in Java.
    id 'application'
}

repositories {
    // Use Maven Central for resolving dependencies.
    mavenCentral()
}

dependencies {
```

```
// Use JUnit Jupiter for testing.
testImplementation libs.junit.jupiter

testRuntimeOnly 'org.junit.platform:junit-platform-launcher'

// This dependency is used by the application.
implementation libs.guava

// https://mvnrepository.com/artifact/com.google.code.gson/gson
implementation 'com.google.code.gson:gson:2.11.0'

// https://mvnrepository.com/artifact/org.projectlombok/lombok
compileOnly 'org.projectlombok:lombok:1.18.34'

// 이 구문은 직접 추가한다.
annotationProcessor 'org.projectlombok:lombok:1.18.34'
}

// Apply a specific Java toolchain to ease working on different
environments.
java {
    toolchain {
        languageVersion = JavaLanguageVersion.of(21)
    }
}

application {
    // Define the main class for the application.
    mainClass = 'kr.hossam.lombok_ex.App01'
}
```