

Apache Maven

Maven이란?

Maven이란 자바용 프로젝트 관리 도구로, project object model (POM) XML 문서를 통해 해당 프로젝트의 버전 정보 및 라이브러리 정보들을 통합하여 관리하는 프레임워크이다.

라이브러리 종속성

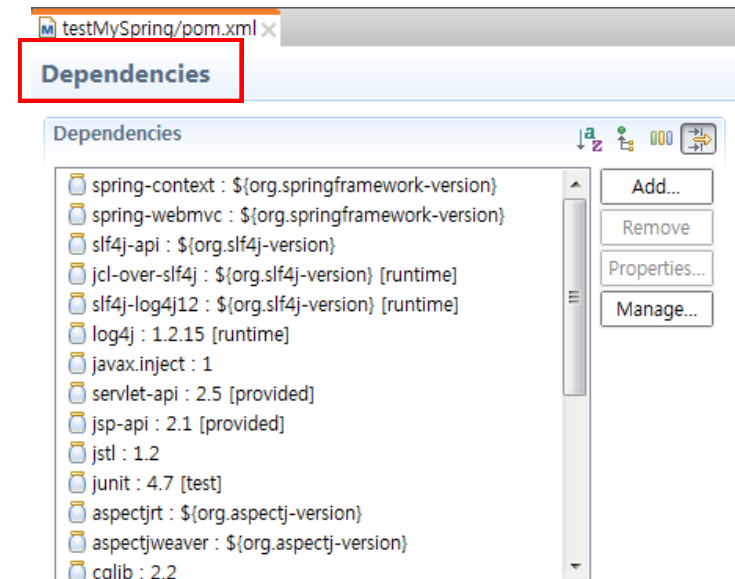
```
Spring      ,      가      ,  
jar         lib  maven
```

일반적인 프로젝트는 개발자가 필요한 라이브러리를 직접 찾아서 추가해야 하지만, Maven을 사용하면 pom.xml 문서에 사용하고자 하는 라이브러리를 등록하여 자동으로 프로젝트에 추가되게 하여 라이브러리 관리의 편리성을 제공해 준다.

Maven의 종속성



maven



- ※ 이제까지 프로젝트 내 lib 폴더에 사용할 라이브러리를 직접 추가하여 관리 해왔다면, maven은 pom.xml 문서 하나 만으로 필요한 라이브러리를 찾아서 자동으로 설치하고 관리할 수 있다.

POM이란?

POM(Project Object Model)은 하나의 프로젝트에서 사용하는 자바 버전, 라이브러리, 플러그인 구성을 통합하여 관리할 수 있게 각 설정 정보를 XML 문서화 한 것을 말한다.

POM.XML 의 구성

```
<project> <!-- Maven 2 버전 이후 POM의 경우 항상 4.0.0. ->  
  <modelVersion> 4.0.0 </ modelVersion>  
  <groupId> com.mycompany.app </ groupId>  
  <!-- 프로젝트 고유 식별 값 그룹 ->  
  <artifactId> my-app </ artifactId>  
  <version> 1.0 </ version>
```

<!-- 라이브러리 의존성 주입 ->

```
<dependencies>  
  <dependency>  
    <groupId> junit </ groupId>  
    <!-- 필수 라이브러리의 이름 ->  
    <artifactId> junit </ artifactId>  
    <version> 3.8.1 </ version>  
    <scope> test </ scope>  
    <!-- 이 라이브러리는 테스트 실행 및 컴파일에만 사용. ->  
  </ dependency>  
</ dependencies>  
</ project>
```

POM.XML

라이브러리 관련 정보 제공 사이트 :

<https://mvnrepository.com/>

Apache Maven 설치하기

Apache Maven 사이트에 접속 <https://maven.apache.org/index.html>



[Apache](#) / [Maven](#) / Welcome to Apache Maven 📄

Last Published: 2018-01-24

MAIN

Welcome

[License](#)

[Download](#)

[Release](#)

[Notes](#)

[Install](#)

[Configure](#)

[Run](#)

[IDE](#)

[Integration](#)

ABOUT MAVEN

[What is](#)

Welcome to Apache Maven

Apache Maven is a software project management and comprehension tool. Based on the concept of a project object model (POM), Maven can manage a project's build, reporting and documentation from a central piece of information.

If you think that Maven could help your project, you can find out more information about in the "About Maven" section of the navigation. This includes an in-depth description of [what Maven is](#), a [list of some of its main features](#), and a set of [frequently asked questions about what Maven is](#).

This site is separated into the following sections, depending on how you'd like to use Maven:

| Use | Download, Install, Run Maven | Configure, Use Maven and Maven Plugins |
|-----|--|--|
|-----|--|--|

| | |
|--|--|
| Information for those needing to build a project that uses Maven | Information for those wanting to use Maven to build their project, including a "10 minute test" that gives a practical overview of Maven's main features in just 10 minutes and plugin list for more information on each |
|--|--|

Apache Maven 설치하기

Downloading Apache Maven 3.5.2

Apache Maven 3.5.2 is the latest release and recommended version for all users.

The currently selected download mirror is <http://apache.mirror.cdnetworks.com/>. If you encounter a problem with this mirror, *backup* mirrors (at the end of the mirrors list) that should be available. You may also consult the [complete list of mirrors](#).

Other mirrors:

⋮ Scroll Down...
•

Files

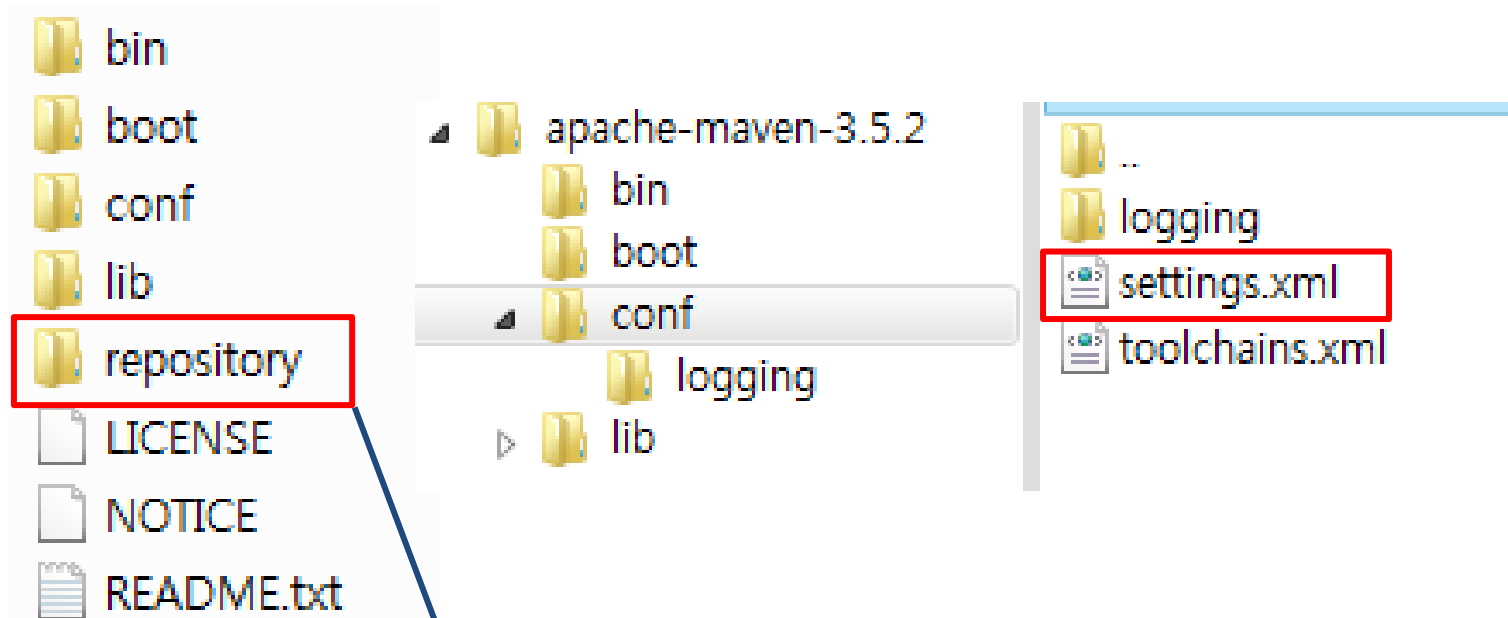
Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release

| | Link | Checksum |
|-----------------------|---|---|
| Binary tar.gz archive | apache-maven-3.5.2-bin.tar.gz | apache-maven-3.5.2-bin.tar.gz.md5 |
| Binary zip archive | apache-maven-3.5.2-bin.zip | apache-maven-3.5.2-bin.zip.md5 |
| Source tar.gz archive | apache-maven-3.5.2-src.tar.gz | apache-maven-3.5.2-src.tar.gz.md5 |
| Source zip archive | apache-maven-3.5.2-src.zip | apache-maven-3.5.2-src.zip.md5 |

Apache Maven 설치하기

다운로드 받은 압축 파일을 dev 폴더에 압축 해제 한 뒤, 같은 경로에 repository라는 저장소 역할의 새 폴더를 하나 만들고, conf 폴더로 접근하여 settings.xml 파일을 열람한다.



repository 폴더는 메이븐을 통해
관리할 라이브러리들이 저장될 경로이다.

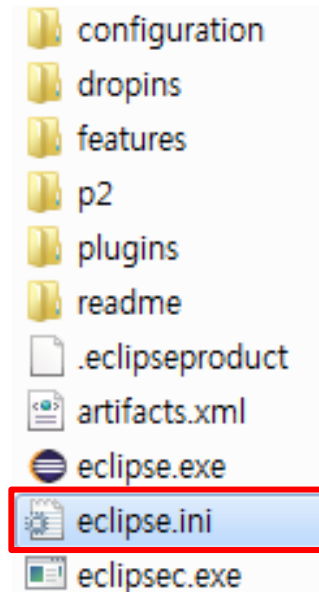
Apache Maven 라이브러리 저장소 설정

settings.xml 문서에서 localRepository 부분을 찾아 주석 바깥으로 뺀 뒤, 저장소로 이용하고자 하는 폴더로 설정한다.

```
<!-- localRepository
| The path to the local repository maven will use to store artifacts.
|
| Default: ${user.home}/.m2/repository
-->
<localRepository>D:\dev\apache-maven-3.5.0\repository</localRepository>
```

Eclipse.ini 설정 추가

Maven과 Eclipse에서 사용하는 java 버전을 일치 시키기 위해 실행 시 이클립스 내장 버전이 아닌 현재 설치된 자바 버전을 인식 하도록 eclipse.ini 설정 파일의 정보에 해당 내용을 추가하고, 이클립스를 재실행한다.



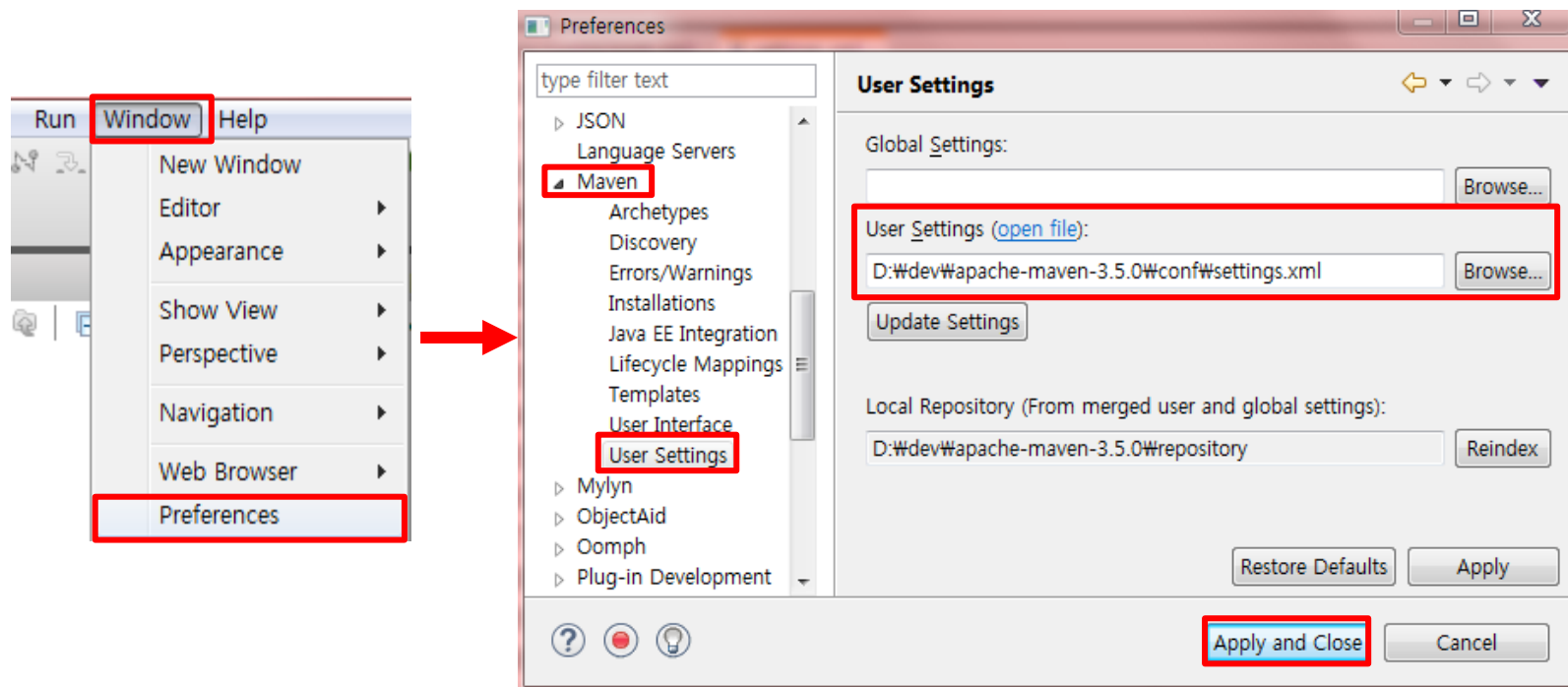
```
-startup
plugins/org.eclipse.equinox.launcher_1.4.0.v20161219-1356
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.
1133
-product
org.eclipse.epp.package.jee.product
-vm C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe
```

Eclipse 와 Maven 연동

Eclipse를 실행하여

[Window] - [Preferences] - [Maven] - [User Settings] 에 접근,

User Settings의 값을 이전에 설정한 conf 의 settings.xml의 위치로 변경한다.



Maven target 폴더

Maven을 사용할 경우 프로젝트 컴파일 시 target/classes 안에 컴파일 된 클래스 파일들이 위치하게 된다.

일반적으로 maven clean 옵션을 사용하면 제거되어 문제는 없지만, 이후 형상관리를 위해 프로젝트를 공유할 시 컴파일 된 결과까지 공유할 필요는 없기 때문에 target 폴더를 공유 목록에서 제외하는 Ignore 작업을 설정하자.

Eclipse target 폴더 제외

[Window] - [Preferences]를 선택한다.

[Team] - [Ignored Resources] 에서 [Add Pattern]을 누른 뒤,
/target/ 을 추가한다.

