

Maven

Gastón Prieto

Mayo 2018

Distribuido bajo licencia [Creative Commons Share-Alike](#)

¿Qué nos resuelve?

- [Creación automática de los proyectos para poder importarlos desde un IDE.](#)
- [Automatización de tareas de compilación/ejecución de test/cobertura, etc.](#)
- [Manejo de las dependencias de nuestro proyecto.](#)

¿Cómo se instala?

<https://maven.apache.org/install.html>

Importante: ¿Cómo sé si instale correctamente Maven?

En la terminal ([linux](#) / [windows](#)) ejecuto el comando
mvn --version

Debería mostrar algo similar a lo siguiente:

Apache Maven 3.5.0

Maven home: /usr/share/maven

.....

¿Cómo funciona?

En el directorio raíz del proyecto se deberá crear un archivo con el nombre ***pom.xml***

En su versión más básica se parecerá a esto:

```
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>nombre-proyecto</groupId>
  <artifactId>nombre-proyecto</artifactId>
  <version>1</version>
</project>
```

Compilar con Java 17, se puede?

- Si, en el *pom.xml* vas a tener que decirle que tiene que compilar con la versión 17.

Un ejemplo de cómo quedaría el primer pom que vimos agregándole la opción de Java 17 sería:

```
<project>
  <modelVersion>4.0.0</modelVersion>
```

```
<groupId>proyecto</groupId>
<artifactId>proyecto</artifactId>
<version>0.0.1-SNAPSHOT</version>
<build>
  <plugins>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.11.0</version>
      <configuration>
        <source>17</source>
        <target>17</target>
      </configuration>
    </plugin>
  </plugins>
</build>
</project>
```

Y con esto ya tenemos configurado nuestro proyecto para Java 17.

¿Cómo construyo mi proyecto (compilación, test, etc)?

Existen muchos comandos, te voy a enseñar los más comunes:

mvn clean

Borra todos los archivos compilados que existan.

mvn compile

Compila todo el código fuente.

mvn test

Primero compila todo el código fuente si no estuviera compilado y luego ejecuta los tests.

mvn install

Ejecuta todos los pasos anteriores (***compile***, ***test***, ***etc***) y además junta el código compilado en un archivo .jar y deja la biblioteca disponible en mi sistema para que la pueda utilizar algún otro proyecto.

¿Cómo agrego una dependencia?

- Antes de saber cómo agregar... ¿Qué es una dependencia?

Podríamos decir que una dependencia en Maven es una biblioteca hecha por terceros que voy a tener disponible para utilizar en mi proyecto.

Si la biblioteca de terceros utiliza otras bibliotecas, entonces Maven se va a dar cuenta de eso y va a bajar esas dependencias también.

- **Por lo que voy entendiendo.. ¿JUnit sería una dependencia?**
- Síí!!! Y podés agregarlo a través de Maven. Te muestro un ejemplo:

```

<project>
.....
<!-- Dependencias -->
<dependencies>
  <dependency>
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter-engine</artifactId>
    <version>5.9.2</version>
    <scope>test</scope>
  </dependency>
</dependencies>
.....
</project>

```

Encontré una biblioteca y quiero agregarla a mi proyecto
¿cómo sé que Maven la va a poder bajar?

Bueno, Maven baja todas las bibliotecas de un repositorio mundial que se lo conoce como Maven Repository. Entonces entrá al sitio de este repositorio y revisá si está en el repositorio y si está mira incluso el snippet que te muestra cómo agregarla. Este es el link al repositorio: <http://mvnrepository.com/>

¿Dónde agrego las clases en el IDE?

Primero vas a tener que crear el siguiente árbol de directorios:

```

.
├── pom.xml
└── src
    ├── main
    │   └── java
    └── test
        └── java

```

- En **src/main/java**, van las clases de dominio
- En **src/test/java**, van las clases que van a testear el dominio

¿Algo más para saber?

Sí, si usás GIT, acordate de agregar al .gitignore los archivos que se autogeneran (.classpath, .project, target, etc.).

¿Y si tengo un proyecto existente y quiero que utilice Maven?

Fácil:

- Creás tu archivo pom.xml en el raíz del proyecto
- Borrás todos los archivos de configuración del IDE (.project, etc..)

- Creás la estructura de directorios antes especificada y ponés las clases donde corresponden.
- Importas el proyecto en el IDE y listo :D

¿Qué links me recomendás para seguir leyendo?

- <https://maven.apache.org/>

¿Tenés algún video de ejemplo?

Si, te dejo el link: https://youtu.be/zJleYRjxM_w (es una versión vieja porque usa Eclipse)