

**Maven™**

The logo consists of the word "Maven" in a large, bold, black sans-serif font. A single feather, oriented vertically and slightly angled, is positioned behind the letter "a". The feather has a vibrant color gradient, transitioning from purple at the base to orange and yellow at the top. A small "TM" symbol is located in the top right corner of the "en" part of the word.

# Cos'è Maven?

---

---

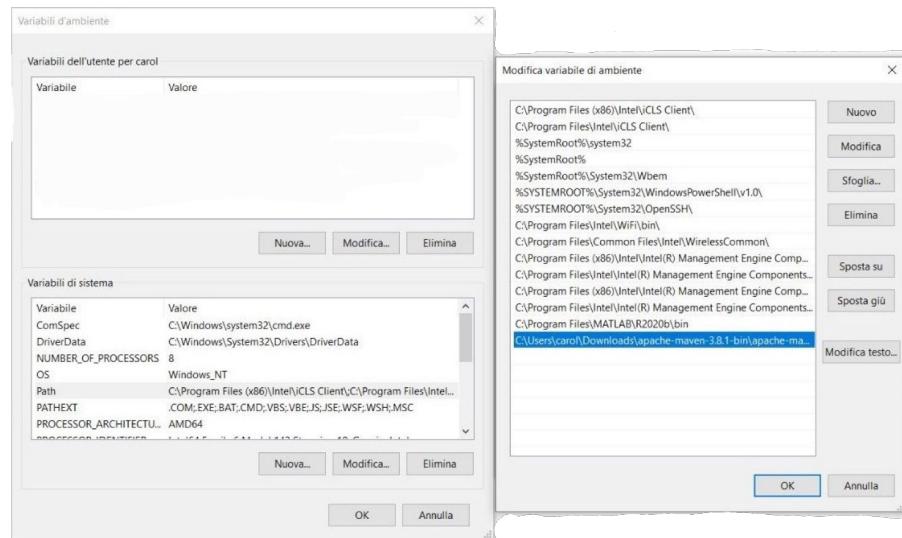
- Maven è uno strumento per la gestione di progetti software.
- Automatizza:
  - compilazione
  - test
  - creazione della documentazione
  - gestione delle dipendenze

```
-----< org.example:test >-----
Building test 1.0-SNAPSHOT
-----[ jar ]-----
--- maven-clean-plugin:2.5:clean (default-clean) @ test ---
--- maven-resources-plugin:2.6:resources (default-resources) @ test ---
[G] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
Copying 0 resource
--- maven-compiler-plugin:2.1:compile (default-compile) @ test ---
Changes detected - recompiling the module!
[G] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
Compiling 2 source files to /home/simone/Downloads/test/target/classes
--- maven-resources-plugin:2.6:resources (default-resources) @ test ---
[G] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
Copying 0 resource
--- maven-compiler-plugin:3.1:compile (default-compile) @ test ---
Nothing to compile - all classes are up to date
--- maven-resources-plugin:2.6:testResources (default-testResources) @ test ---
[G] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
skip non existing resourceDirectory /home/simone/Downloads/test/src/test/resources
--- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ test ---
Nothing to compile - all classes are up to date
--- maven-surefire-plugin:2.12.4:test (default-test) @ test ---
No tests to run.
--- maven-jar-plugin:2.4:jar (default-jar) @ test ---
Building jar: /home/simone/Downloads/test/target/test-1.0-SNAPSHOT.jar
--- maven-assembly-plugin:2.2-beta-5:single (make-assembly) @ test ---
```

# Download e Installazione

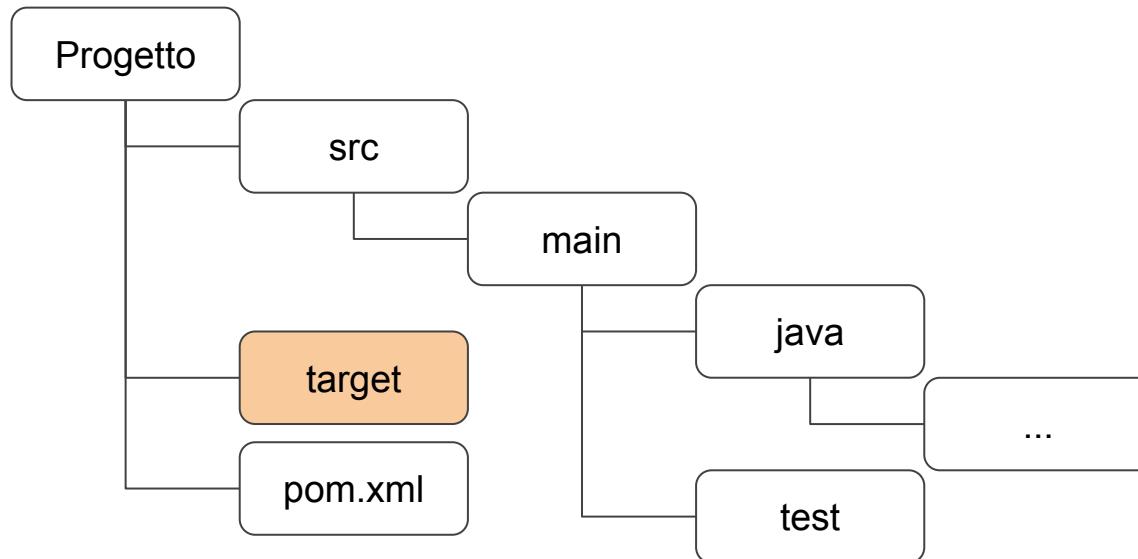
---

- Scaricare il binary zip da:
  - <https://maven.apache.org/>
- Aggiungere il percorso della cartella /bin di Maven alla variabile d'ambiente PATH.



# Una nuova struttura

---



# Project Object Model (pom.xml)

---

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>project-name</artifactId>
    <version>1.0</version>

    <modules>
        <module>
            ...
        </module>
    </modules>

    <build>
        <plugins>
            <plugin>
                ...
            </plugin>
        </plugins>
    </build>

    <dependencies>
        <dependency>
            ...
        </dependency>
    </dependencies>

</project>
```

# Project Object Model (pom.xml)

---

- Contiene informazioni riguardo al progetto.

```
<groupId>org.example</groupId>
  <artifactId>project-name</artifactId>
  <version>1.0</version>

  <licenses>
    <license>
      <name>The Apache Software License, Version 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
    </license>
  </licenses>
  <developers>
    <developer>
      <id>id-sviluppatore</id>
      <name>nome-sviluppatore</name>
      <roles>
        <role>architect</role>
        <role>developer</role>
      </roles>
      <timezone>+8</timezone>
    </developer>
  </developers>
```

# Project Object Model (pom.xml)

---

- Contiene dettagli sulla configurazione usata per gestire il progetto.

```
<plugin>
    <artifactId>maven-assembly-plugin</artifactId>
    <configuration>
        <archive>
            <manifest>
                <mainClass>Test</mainClass>
            </manifest>
        </archive>
        <descriptorRefs>
            <descriptorRef>jar-with-dependencies</descriptorRef>
        </descriptorRefs>
    </configuration>
    <executions>
        <execution>
            <id>make-assembly</id>
            <phase>package</phase>
            <goals>
                <goal>single</goal>
            </goals>
        </execution>
    </executions>
</plugin>
```

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-javadoc-plugin</artifactId>
    <version>3.0.1</version>
</plugin>
```

# Project Object Model (pom.xml)

---

- Contiene dettagli sulla configurazione usata per gestire le dipendenze del progetto.

```
<dependencies>
    <dependency>
        <groupId>com.googlecode.json-simple</groupId>
        <artifactId>json-simple</artifactId>
        <version>1.1.1</version>
    </dependency>
</dependencies>
```

# MVN Repository

---

- Repository Link:
  - <https://mvnrepository.com/>

 **JSON.simple**  
A simple Java toolkit for JSON

License	Apache 2.0
Categories	JSON Libraries
Tags	json
Used By	1,595 artifacts

Central (2) Redhat GA (1) Redhat EA (2) ICM (1) Toutatice (1)

Version	Repository	Usages	Date
1.1.1	Central	1,058	Mar, 2012
1.1	Central	571	Aug, 2009

# MVN Repository

— — —

 **JSON.simple » 1.1.1**  
A simple Java toolkit for JSON

<b>License</b>	Apache 2.0
<b>Categories</b>	JSON Libraries
<b>HomePage</b>	<a href="http://code.google.com/p/json-simple/">http://code.google.com/p/json-simple/</a>
<b>Date</b>	(Mar 21, 2012)
<b>Files</b>	<a href="#">pom (3 KB)</a> <a href="#">bundle (23 KB)</a> <a href="#">View All</a>
<b>Repositories</b>	<a href="#">Central</a> <a href="#">Mulesoft</a> <a href="#">Redhat GA</a> <a href="#">Sonatype</a>
<b>Used By</b>	<a href="#">1,595 artifacts</a>

[Maven](#) [Gradle](#) [SBT](#) [Ivy](#) [Grape](#) [Leiningen](#) [Buildr](#)

```
<!-- https://mvnrepository.com/artifact/com.googlecode.json-simple/json-simple -->
<dependency>
    <groupId>com.googlecode.json-simple</groupId>
    <artifactId>json-simple</artifactId>
    <version>1.1.1</version>
</dependency>
```

Include comment with link to declaration

### Compile Dependencies (1)

Category/License	Group / Artifact	Version	Updates
<a href="#">Testing</a> <a href="#">EPL 2.0</a>	 <a href="#">junit</a> × <a href="#">junit</a>	<a href="#">4.10</a>	<a href="#">5.7.2</a>

# Plugins

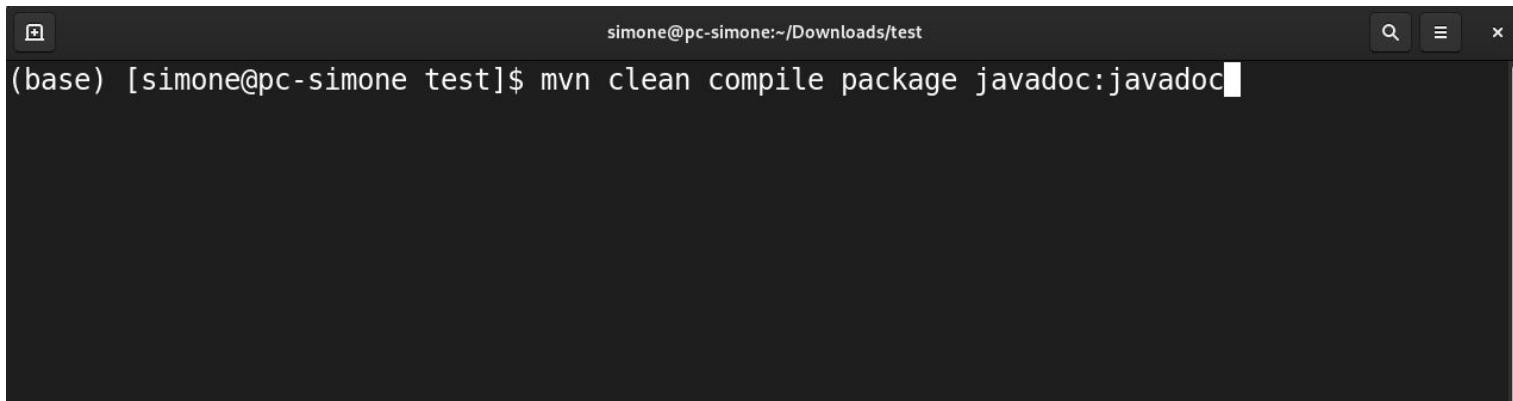
---

- I plugins corrispondono a delle fasi principali.
- Alcuni dei plugin principali (core):
  - clean: pulisce dopo la build del progetto
  - compile: compila i sorgenti Java
  - test: esegue gli unit test per sorgenti compilati
  - package: Prende i sorgenti e li impacchetta (i.e. JAR)
- Altri plugin:
  - javadoc: genera la javadoc per il progetto

# Comandi

---

- mvn clean compile package javadoc:javadoc



A screenshot of a terminal window titled "simone@pc-simone:~/Downloads/test". The window has a dark background and light-colored text. It shows the command "(base) [simone@pc-simone test]\$ mvn clean compile package javadoc:javadoc" followed by a cursor at the end of the line.