

Desenvolupament d'aplicacions multiplataforma

Accés a dades

Apache
MavenTM

Índex de continguts

| | |
|---------------------------------------|---|
| Maven..... | 3 |
| Arxiu POM (Project Object Model)..... | 3 |



Maven

Maven és una paraula Yiddish que significa *acumulador de coneixement*. Aquesta eina va començar com un intent de simplificar el procés de construcció del projecte *Jackarta Turbine* que estava format per diversos subprojectes, cada un amb la seva configuració, ... Volien generar una eina capaç de facilitar la tasca de gestionar aquests projectes.

Els objectius de *maven* són facilitar la tasca de construcció del projecte, proporcionar un sistema de construcció uniforme, proporcionar informació de la qualitat del projecte i facilitar les bones pràctiques de desenvolupament d'aplicacions.

Arxiu POM (Project Object Model)

El fitxer *pom.xml* és el centre de la configuració del projecte *maven*. És un simple fitxer de configuració *xml* que conté la majoria de la informació necessària per construir el projecte. Pot arribar a ser molt complexe, però afortunadament no fa falta conèixer profundament tots els seus detalls per usar-lo correctament.

Tot seguit teniu un exemple del fitxer POM d'un projecte Spring Boot.

Les capçaleres identifiquen els esquemes per validar l'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>
```

L'element *parent* definiex l'aplicació com una aplicació hereva de Spring Boot.

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.4.1</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
```

Els següents elements defineixen:

- L'identificador del projecte, el paquet arrel del projecte

```
<groupId>cat.paucasesnovescifp.swdws</groupId>
```

- El nom del paquet arrel d'aquest mòdul del projecte, i del jar

```
<artifactId>demo</artifactId>
```

- La versió. SNAPSHOT vol dir que no és la versió definitiva, que és una versió de treball.

```
<version>0.0.1-SNAPSHOT</version>
```

- El nom i la descripció són opcionals

```
<name>demo</name>
<description>Projecte demostració per Spring Boot</description>
```

- Podem definir les propietats necessàries, en aquest cas la versió de Java. Aquestes propietats es poden utilitzar a altres parts del POM o poden ser utilitzades per els plugins.

```
<properties>
```

```
<java.version>21</java.version>  
</properties>
```

- Les dependències del projecte. No posam el nom del jar, sinó el seu groupId i artifactId. Utilitza aquestes dades per cercar-los al repositori de *maven*, baixar-los i incloure'ls al projecte.

```
<dependencies>  
  <dependency>  
    <groupId>org.springframework.boot</groupId>  
    <artifactId>spring-boot-starter-security</artifactId>  
  </dependency>  
  <dependency>  
    <groupId>org.springframework.boot</groupId>  
    <artifactId>spring-boot-starter-web</artifactId>  
  </dependency>  
  
  <dependency>  
    <groupId>org.springframework.boot</groupId>  
    <artifactId>spring-boot-starter-test</artifactId>  
    <scope>test</scope>  
  </dependency>  
  <dependency>  
    <groupId>org.springframework.security</groupId>  
    <artifactId>spring-security-test</artifactId>  
    <scope>test</scope>  
  </dependency>  
</dependencies>
```

- plugins: maven pròpiament dit només implementa la funcionalitat bàsica per realitzar la seva tasca. Per tasques concretes, com generar un jar o war, com compilar el codi, ... utilitza plugins

```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
</project>
```