

# Pom, comandos y goals:

## El POM y sus partes:

<?xml version="1.0" encoding="UTF-8"?>

```
<project xmlns="https://maven.apache.org/POM/4.0.0"
  xmlns:xsi="https://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> ppss </groupId>
  <artifactId> nombre </artifactId>
  <version> 1.0-SNAPSHOT </version>
```

organización a la que pertenece el proyecto  
Coordenadas

dependencias

```
  <dependencies>
    <dependency>
      <groupId> org.junit.jupiter </groupId>
      <artifactId> junit-jupiter-engine </artifactId>
      <version> 5.8.2 </version>
      <scope> test </scope>
    <dependency>
    <dependency>
      <groupId> org.easymock </groupId>
      <artifactId> easymock </artifactId>
      <version> 4.2 </version>
      <scope> test </scope>
    <dependency>
  </dependencies>
```

proceso de construcción

```
  <build>
    <plugins>
      <plugin>
        <groupId> org.apache.maven.plugins </groupId>
        <artifactId> maven-compiler-plugin </artifactId>
        <version> 3.9.0 </version>
      <plugin>
        <!-- JUnit requiere Surefire version 2.22 o superior -->
      <plugin>
        <groupId> org.apache.maven.plugins </groupId>
        <artifactId> maven-surefire-plugin </artifactId>
        <version> 3.0.0-M5 </version>
        <configuration>
          <argLine> --add-opens java.base/java.lang = ALL-UNNAMED </argLine>
        </configuration>
      </plugin>
    </plugins>
  </build>
```

```
</project>
```

## Fases y goals del proyecto

Fase	Plugin : Goal	acción Realizada
process-resources •	maven-resources-plugin : resources	Copia *.* de src/main/resources en target
Compile •	maven-compiler-plugin : compile	Compila *.java de src/main/java
process-test-resources	maven-resources-plugin : testResources	Copia *.* de src/test/resources en target
test-compile •	maven-compiler-plugin : testCompile	Compila *.java de src/test/java
test •	maven-surefire-plugin : test	Ejecuta los test unitarios
package •	maven-jar-plugin : jar	Empaqueta *.class + recursos en un jar
install	maven-install-plugin : install	Copia el fichero jar en repositorio local.
deploy	maven-deploy-plugin : deploy	Copia el fichero jar en repositorio remoto

## Comandos maven

- Correr test de una clase en específico  
mvn clean test -D test = ClaseTest
- Correr test de una clase pero sin los taggeados con NoCorrer  
mvn clean test -D test = ClaseTest -D excludedGroups = NoCorrer
- Correr test taggeados con SiCorrer y testados  
mvn clean test -D -Dgroups = SiCorrer, testados
- Correr todos los test  
mvn test -D