Quản lý dự án với MAVEN

Mentor: Lê Nhật Tùng

Maven là gì?



MAVEN là một công cụ quản lý dự án phổ biến trong phát triển phần mềm.

Nó giúp đơn giản hóa quá trình xây dựng, quản lý dependencies, testing và deployment của dự án. MAVEN sử dụng cấu trúc dự án chuẩn và tập trung vào việc tự động hóa các tác vụ phát triển. Bằng cách sử dụng MAVEN, người phát triển có thể dễ dàng quản lý các thư viện, kiểm soát phiên bản và tạo ra các sản phẩm phần mềm đáng tin cậy và nhất quán.

Lợi ích của MAVEN trong phát triển phần mềm

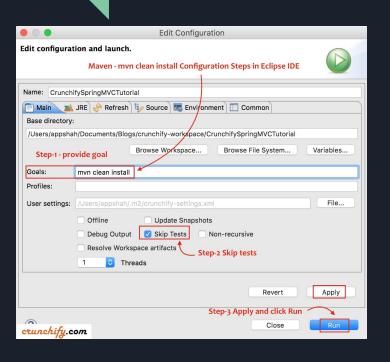
- Quản lý Dependencies dễ dàng
- Tự động hóa quá trình Build
- Quản lý phiên bản và Release
- Tích hợp dễ dàng với IDE
- Chuẩn hóa cấu trúc dự án

Quản lý Dependencies dễ dàng



MAVEN giúp tự động quản lý các dependencies của dự án. Bằng cách đinh nghĩa dependencies file trong pom.xml, MAVEN sẽ tự động tải về và quản lý các thư viện và phiên bản tương thích cho dự án.

Tự động hóa quá trình Build



MAVEN cung cấp khả năng tự động hóa quá trình build dự án. Bằng cách định nghĩa các mục tiêu (goals) trong file pom.xml, người phát triển có thể thực hiện các tác vụ như biên dịch mã nguồn, đóng gói ứng dụng và tạo ra các phiên bản phần mềm một cách dễ dàng.

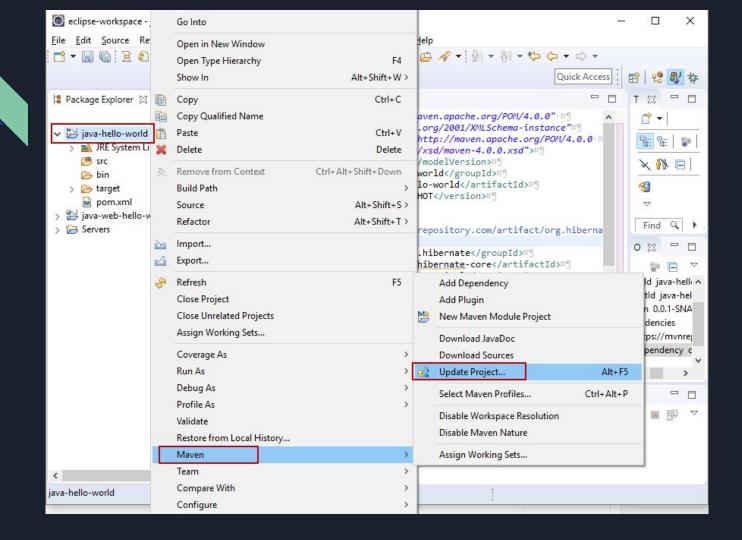
Quản lý phiên bản và Release

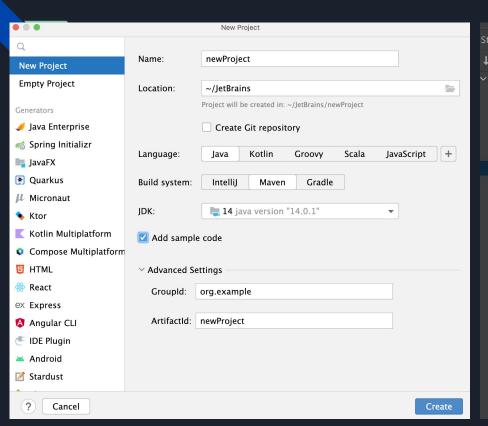


MAVEN hỗ trợ quản lý phiên bản của dự án một cách chặt chẽ. Người phát triển có thể dễ dàng xác định và cập nhật phiên bản của dự án thông qua file pom.xml. Ngoài ra, MAVEN cung cấp các plugin hữu ích để thực hiện quá trình release và deployment của dự án.

Tích hợp dễ dàng với IDE

MAVEN tích hợp tốt với các môi trường phát triển tích hợp (IDE) phổ biến như Eclipse, IntelliJ IDEA và NetBeans. Điều này giúp người phát triển dễ dàng thực hiện các tác vụ MAVEN từ giao diện IDE mà không cần phải sử dụng dòng lệnh.





```
<arg>--enabl ♣1 ^ ∨
                     $ -
                                                       </compilerArgs>
IIII Dependency
                                                   </configuration>
                                              </plugin>
                                          </plugins>
     > 🐂 maven-surefire-plugii
                                     </build>
    > 🕌 kotlin-maven-plugin:
     > 👫 maven-compiler-plug

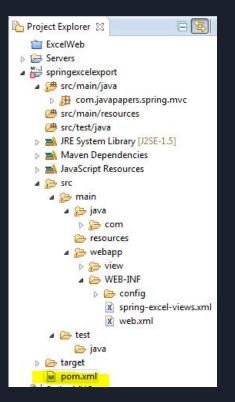
    Dependencies

     IIII com.jetbrains:annotation
                                          <dependency>
     IIII org.apache.commons:coi
                                              <groupId>com.jetbrains
     IIII org.junit.jupiter:junit-jup
     IIII org.hamcrest:hamcrest-li
                                              <artifactId>annotation-inference
     IIII org.jetbrains:annotations
                                              <version>1.0
    IIII org.jetbrains.kotlin:kotlin
                                              <scope>system</scope>
   > IIII org.jetbrains.kotlin:kotlin
                                              <systemPath>${basedir}/lib/annota
                                          </dependency>
                                          <dependency>
                                              <groupId>org.apache.commons
                                              <artifactId>commons-lang3</artifa
                                              <version>3.11
                                          </dependency>
```

Chuẩn hóa cấu trúc dự án

MAVEN thúc đẩy việc sử dụng cấu trúc dự án chuẩn, giúp tổ chức dự án một cách rõ ràng và dễ quản lý. Sự chuẩn hóa này giúp tăng khả năng mở rộng, tái sử dụng mã nguồn và tích hợp dễ dàng với các công cụ và quy trình phát triển khác.

Cấu trúc chuẩn của dự án Maven



Directory	Description
src/main/java	Your Java source code
src/main/resources	Properties / config files used by your app
src/main/webapp	JSP files and web config files other web assets (images, css, js, etc)
src/test	Unit testing code and properties
target	Destination directory for compiled code. Automatically created by Maven

File POM.XML

https://maven.apache.org/guides/introduction/introduction-to-the-pom.html

Minimal POM

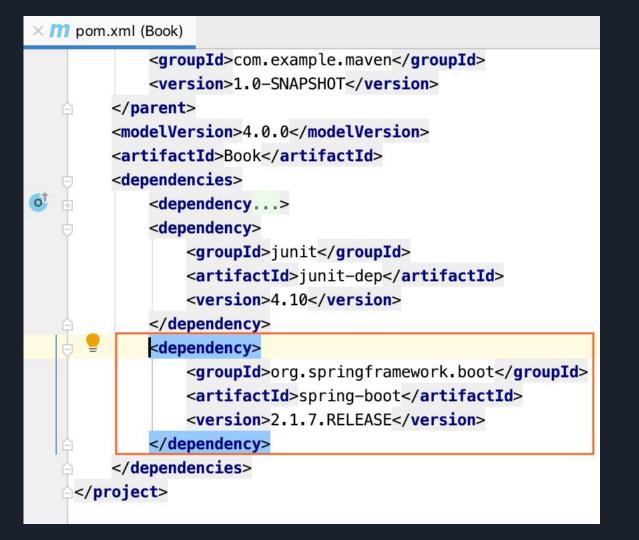
The minimum requirement for a POM are the following:

- project root
- modelVersion Should be set to 4.0.0
- groupId the id of the project's group.
- artifactId the id of the artifact (project)
- version the version of the artifact under the specified group

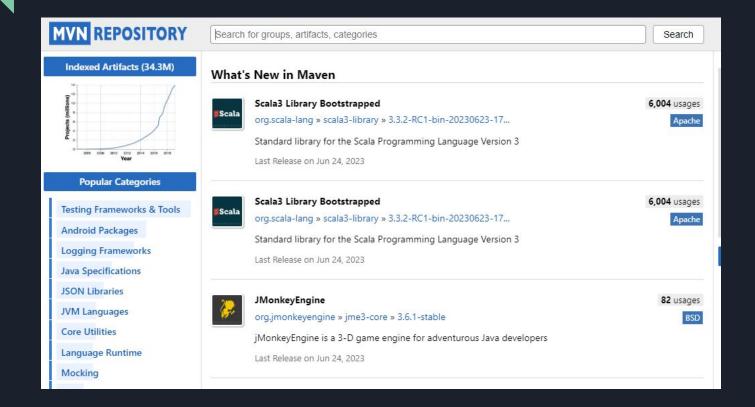
Here's an example:

- 1. <project>
- <modelVersion>4.0.0</modelVersion>
- 3.
- 4. <groupId>com.mycompany.app</groupId>
- <artifactId>my-app</artifactId>
- 6. <version>1</version>
- 7. </project>

```
1. <project>
      cproperties>
        <mayenVersion>3.0</mayenVersion>
 4.
 5.
      </properties>
 6.
      <dependencies>
 7.
        <dependency>
 8.
          <groupId>org.apache.maven</groupId>
 9.
          <artifactId>maven-artifact</artifactId>
10.
11.
          <version>${mavenVersion}</version>
12.
       </dependency>
13.
       <dependency>
          <groupId>org.apache.maven
14.
          <artifactId>maven-core</artifactId>
15.
16.
          <version>${mavenVersion}</version>
17.
        </dependency>
18.
      </dependencies>
19.
20. </project>
```



MAVEN REPOSITORY



Thực hành tạo một dự án MAVEN đơn giản

- Sử dụng MySQL
- Sử dụng JDBC
- ...