

Maven Commands - From Basic to Advanced

Basic Maven Commands

1. **Creating a Maven Project**

Command:

mvn archetype:generate

Description: Creates a new Maven project using a template (archetype).

Example:

**mvn archetype:generate -DgroupId=com.example -DartifactId=my-app
-DarchetypeArtifactId=maven-archetype-quickstart
-DinteractiveMode=false**

2. **Compile the Project**

Command:

mvn compile

Description: Compiles the source code from the src/main/java directory.

3. **Run Unit Tests**

Command:

mvn test

Description: Runs all unit tests under src/test/java.

4. **Create a Package**

Command:

mvn package

Description: Packages the compiled code into a .jar or .war file.

5. **Install the Package to Local Repository**

Command:

mvn install

Description: Installs the package into the local Maven repository.

Intermediate Maven Commands

6. **Clean Project**

Command:

mvn clean

Description: Deletes the target/ directory to ensure a fresh build.

7. **Execute a Specific Goal**

Command:

mvn <plugin>:<goal>

Example:

mvn dependency:tree

8. ****Validate Project Configuration****

Command:

mvn validate

Description: Validates the pom.xml file and checks if all dependencies are available.

9. ****Generate Site Documentation****

Command:

mvn site

Description: Generates project reports and documentation.

10. ****Skip Tests During Build****

Command:

mvn package -DskipTests

Advanced Maven Commands

11. ****Deploy Artifacts to Remote Repository****

Command:

mvn deploy

Example:

mvn **deploy**
-DaltDeploymentRepository=repo::default::http://your-repo.com/relea

ses

12. **Download Dependencies (Without Build)**

Command:

mvn dependency:resolve

13. **Run a Maven Goal in Debug Mode**

Command:

mvn <goal> -X

14. **Force Update of Dependencies**

Command:

mvn clean install -U

15. **Check Dependency Tree**

Command:

mvn dependency:tree

16. **Release a Project (Release Plugin)**

Command:

mvn release:prepare release:perform

17. **Profile-based Build**

Command:

mvn clean install -P<profile-name>

18. **Check Effective POM**

Command:

mvn help:effective-pom

19. **Analyze Dependency Issues**

Command:

mvn dependency:analyze

20. **Run Maven in Offline Mode**

Command:

mvn <goal> -o

Conclusion:

These commands provide a full spectrum of Maven's capabilities, from project creation to advanced build processes.