Maven Lifecycle Phases Demonstration

In this document, we will demonstrate the use of Maven lifecycle phases by executing them on a sample project.

The Maven lifecycle consists of several phases such as clean, compile, test, package, install, and deploy. Each phase performs specific tasks and can be executed using Maven commands. Sample Project Overview

We have a simple Java project named "SampleProject" with a standard Maven directory structure. It contains source code in the src/main/java directory and test code in the src/test/java directory. The project also includes a pom.xml file with project configuration.

Maven Lifecycle Phases Execution :

1.Clean

This phase removes any previously compiled classes and resources from the target directory.

mvn clean

2. Compile

This phase compiles the source code of the project.

mvn compile

3. Test

This phase runs the tests in the project.

mvn test

4. Package This phase packages the compiled code into distributable format such as JAR, WAR, or EAR.

mvn package

5. Install

This phase installs the package into the local Maven repository, making it available for other projects locally.

mvn install

6. Deploy

This phase copies the final package to the remote repository for sharing it with other developers or projects.

mvn deploy

Conclusion

Each Maven lifecycle phase performs specific tasks, allowing developers to manage the build process efficiently. By understanding and executing these phases, developers can automate various aspects of the software development lifecycle.