

## 1. What is Maven?

Maven is a build automation and project management tool that simplifies the building, testing, and deployment of software projects.

## 2. What is the primary goal of Maven?

The primary goal of Maven is to provide a consistent way to manage projects, including their dependencies, builds, and documentation.

## 3. What is a "POM" in Maven?

POM stands for "Project Object Model." It's an XML file that contains project information, dependencies, plugins, and configurations.

## 4. How does Maven handle project dependencies?

Maven uses a central repository to manage project dependencies, automatically downloading and managing them during the build process.

## 5. What is a "Maven Artifact"?

A Maven artifact is a file, usually a JAR, WAR, or EAR file, that's produced by a project and stored in the local or remote repository.

## 6. How do you specify project dependencies in the Maven POM?

You define dependencies using the `<dependencies>` section in the POM, specifying the artifact's group ID, artifact ID, and version.

## 7. What is the purpose of the "groupId," "artifactId," and "version" in a Maven POM?

The "groupId" identifies the group to which the project belongs, the "artifactId" is the project's unique name, and the "version" is the project's version.

## 8. How does Maven resolve transitive dependencies?

Maven automatically resolves and downloads transitive dependencies, which are dependencies of your dependencies.

9. How can you create a new Maven project using the command line?

You can use the command `mvn archetype:generate` and select a suitable archetype to create a new project structure.

10. What is a "Maven Plugin"?

- A Maven plugin is a piece of software that extends the build capabilities of Maven by providing additional goals and functionalities.

11. How do you specify project properties in a Maven POM?

- Project properties can be defined in the `<properties>` section of the POM and can be referenced using `${property_name}` syntax.

12. What is the "Maven Repository"?

- The Maven repository is a location where Maven stores project artifacts, dependencies, and plugins for reuse across different projects.

13. How can you run a Maven build?

- Use the command `mvn clean install` to compile the code, run tests, and package the project artifacts.

14. What is the purpose of the "mvn clean" command?

- The `mvn clean` command deletes the target directory, removing the compiled files and other generated artifacts.

15. How does Maven manage project lifecycles and phases?

- Maven organizes the build process into different lifecycles (e.g., `clean`, `validate`, `compile`) that are further divided into phases.

16. Explain the Maven build lifecycle phases "compile," "test," and "package."

- "compile" compiles the project's source code, "test" runs tests, and "package" creates the final artifact (e.g., JAR, WAR) of the project.

17. What is the difference between "clean" and "install" in Maven?

- "clean" removes any previously compiled files, while "install" compiles, tests, and packages the project before installing it in the local repository.

18. What is a "SNAPSHOT" version in Maven?

- A "SNAPSHOT" version is used for an unstable, in-progress version of a project. It can be frequently updated without changing its version number.

19. How can you skip running tests during a Maven build?

- You can skip tests using the -DskipTests flag when running the mvn command.

20. What is the purpose of the "Maven Central Repository"?

- The Maven Central Repository is a public repository where commonly used Java libraries and artifacts are hosted for easy access.

21. How can you specify a custom repository in a Maven POM?

- You can define a <repositories> section in the POM and specify the URL and other details of the custom repository.

22. What is a "Maven Parent POM"?

- A Maven Parent POM is a POM file that contains common configurations and dependencies shared across multiple projects.

23. How do you specify a parent POM for a project?

- You define the parent POM in the <parent> section of the project's POM, specifying the parent's group ID, artifact ID, and version.

24. How can you create a reusable custom Maven archetype?

- You can create a custom archetype by defining a template project structure and using the mvn archetype:create-from-project command.

25. How does Maven manage version conflicts between dependencies?

- Maven uses a dependency resolution mechanism that selects the version based on a set of rules, helping to manage version conflicts.

26. What is the role of the ".m2" directory in Maven?

- The .m2 directory is the default location where Maven stores downloaded dependencies, plugins, and the local repository.

27. How can you skip the compilation phase during a Maven build?

- You can skip the compilation phase using the `-Dmaven.compiler.skip=true` flag when running the `mvn` command.

28. What is the purpose of a "Maven Repository Manager"?

- A Maven Repository Manager is a tool used to manage and host project artifacts, dependencies, and plugins for a team or organization.

29. Explain the concept of "Maven Coordinates."

- Maven Coordinates consist of the group ID, artifact ID, and version of an artifact, helping to uniquely identify it.

30. How do you configure proxy settings for Maven?

- Proxy settings can be configured in the `Maven settings.xml` file, specifying the proxy server's details.