

Basic Maven Interview Questions

1. What is Maven?

Maven is a popular open-source build tool developed by the Apache Group to build, publish, and deploy several projects. It is written in **Java** and is used to build projects written in C#, Scala, Ruby, etc.

This tool is used to develop and manage any Java-based project. It simplifies the day-to-day work of Java developers and aids them in their projects.



2. What does Maven help with?

Maven helps with the following:

- **Apache Maven** helps manage all the processes, such as building, documentation, releasing, and distribution in project management.
- The tool simplifies the process of project building. It increases the performance of the project and the building process.
- The task of downloading JAR files and other dependencies is done automatically.
- Maven provides easy access to all the required information.

- Maven makes it easy for the developer to build a project in different environments, without worrying about the dependencies, processes, etc.
- In Maven, it is simple to add new dependencies, all you need to do is to write the dependency code in the pom file.

3. What are the different elements that Maven takes care of?

In the process, it takes care of the following:

- Builds
- Dependencies
- Reports
- Distribution
- Releases
- Mailing list



4. What is the difference between ANT and Maven?

ANT	Maven
<ul style="list-style-type: none"> 1. ANT has no formal conventions, so information is to be provided in the build.xml file. 1. ANT is procedural. 1. ANT has no life cycle. 1. ANT scripts are not reusable. 1. ANT is specifically a build tool. 	<ul style="list-style-type: none"> 1. Maven has conventions, so information is not to be provided in the pom.xml file. 1. Maven is declarative. 1. Maven has a life cycle. 1. Maven is mainly used as a project management tool. 1. Maven plugins are reusable.

5. What is POM?

Project Object Model (POM) refers to the XML files with all the information regarding project and configuration details.

- It has the description of the project, information regarding the versioning and configuration management of the project.
- The XML file is in the project home directory. When we tend to execute a task, Maven searches for the POM in the current directory.



6. What all is included in the POM?

POM contains the following configurations:

- Dependencies
- Developers and contributors
- Plugins
- Plugin configuration
- Resources

7. What are the minimum required elements for POM?

The minimum required elements for POM are:

- project root
- modelVersion – should be 4.0.0
- groupId – project's group id
- artifactId – artifact (project) id
- version – version of the artifact

8. What is meant by the term 'Build Tool'?

A build tool is essential for the process of building. It is needed for the following procedures:

- Generating source code.
- Generating documentation from the source code.
- Compiling of source code.
- Packaging of the compiled codes into JAR files.
- Installing the packaged code in a local repository, server, or central repository.

9. What are the steps to install Maven on Windows?

To install Maven on Windows, observe the following steps:

- Download Maven first, and then extract it.
- In the environment variable, add JAVA_HOME, and MAVEN_HOME.
- Then, add the environment path in the Maven variable.

- Lastly, verify [the Maven installation](#) by checking its version. The command `mvn -version` will display the version installed in the system.

10. What are the steps to install Maven on Ubuntu?

To [install Maven on to Ubuntu](#), observe the following steps:

- The primary step is to install Java.
- Then, download Maven.
- Configure environment variables `JAVA_HOME`, `M3_HOME`, `MAVEN_HOME`, and `PATH`.
- Lastly, verify the Maven installation by checking its version.

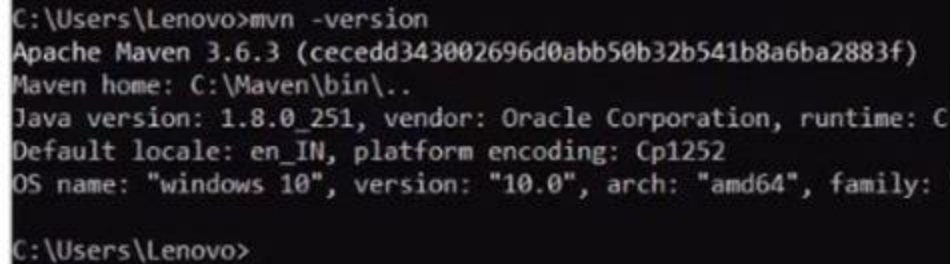


11. What is the command to install JAR files in the Local Repository?

- `mvn install` is used to install JAR files in the local repository.
- To install the JAR manually into the local Maven repository, the following plugin is used: `mvn install:install-file-Dfile=<path to file>`.

12. How do you know the version of Maven being used?

- mvn -version is used to check the version of Maven present in the system.
- The command is typed in the command prompt, and the version of Maven in the system will eventually appear on the screen.



```
C:\Users\Lenovo>mvn -version
Apache Maven 3.6.3 (cecedd343002696d0abb50b32b541b8a6ba2883f)
Maven home: C:\Maven\bin\..
Java version: 1.8.0_251, vendor: Oracle Corporation, runtime: C
Default locale: en_IN, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family:
C:\Users\Lenovo>
```

13. What is Clean, Default, and Site in Maven?

There are three built-in build life cycles:

- Clean: The clean lifecycle looks after project cleaning.
- Default: The default lifecycle handles the project deployment.
- Site: The site lifecycle refers to the creation of the project's site documentation.

14. What is a Maven Repository?

Maven repositories refer to the directories of packaged JAR files that contain metadata. The metadata refers to the POM files relevant to each project. This metadata is what allows Maven to download dependencies.

There are three types of repositories:

- Local Repository
- Remote Repository
- Central Repository

15. What are the different types of Maven Repositories?

There are three types of Maven repositories:

1. Local Repository:

- Local repository refers to the machine of the developer where all the project material is saved.
- The local repository contains all the dependency jars.

2. Remote Repository:

- The remote repository refers to the repository present on a server that is used when Maven needs to download dependencies.
- Whenever anything is required from the remote repository, it is first downloaded to the local repository, and then it is used.

3. Central Repository:

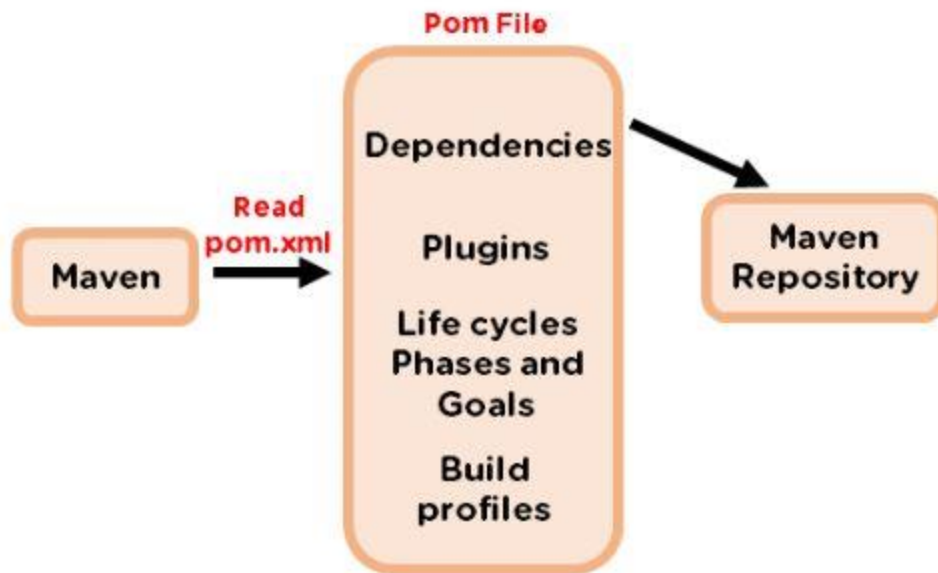
- Central repository refers to the Maven community that comes into action when there is a need for dependencies, and those dependencies cannot be found in the local repository.
- Maven downloads the dependencies from here in the local repository whenever needed.



16. How does Maven Architecture work?

Maven architecture works in three steps, which are as follows:

- The first step is to read the pom.xml file.
- Then, it downloads the dependencies defined in pom.xml into the local repository from the central repository.
- Lastly, it creates and generates a report according to the requirements, and executes life-cycles, phases, goals, plugins, etc.



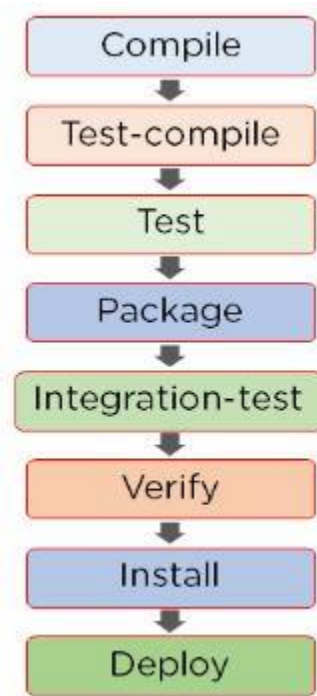
17. What is Maven Build Lifecycle?

Maven lifecycle is a collection of steps that are to be followed, to build a project. There are three built-in build lifecycles:

- **Default:** Handles project deployment.
- **Clean:** Handles project cleaning.
- **Site:** Handles the creation of the project site's documentation.

18. What are the different phases in the Maven Build Lifecycle?

Build Lifecycle has different build phases or stages, which are below:



19. Which command is used to build a Maven site?

- - mvn site command is used to build a Maven site
- The resulting site, by default, is target/site/...

20. What are the different conventions used while naming a project in Maven?

The full name of a project in Maven includes:

<GroupId>: <artifactId>: <version>

For example:

`org.apache.maven: maven: 2.0.1`

Now that we are done with some Basic Maven Interview Questions, let's move on to the Intermediate Maven Interview Questions.

Intermediate Maven Interview Questions

21. What is a Maven Artifact?

Maven Artifact refers to a file, usually a JAR that gets deployed to a Maven repository. The tool creates one or more artifacts, such as a compiled JAR and a source JAR.

Every artifact has its groupId, an artifact ID, and a version string. These three together identify the artifact. For example:

`com.your.package`, any name, and a version string (to uniquely identify).

22. What are the phases of a Clean Life Cycle?

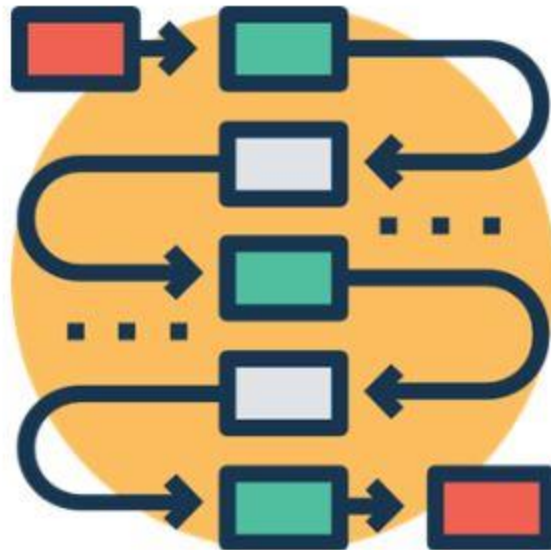
The phases of a clean life cycle include:

- Pre-clean
- Clean
- Post-clean

23. What are the phases of a Site Life Cycle?

The phases of a site life cycle include:

- Pre-site
- Site
- Post-site
- Site-deploy



24. What is meant by Maven Plugins?

- Maven plugins are essential features of Maven that are used to reuse the common build logic across several projects.
- The plugins are needed to carry out tasks like compiling code, testing them, creating JAR files, and documenting the projects.

25. Why are Maven Plugins used?

Maven Plugins are used to:

- Create a JAR file.
- Create a WAR file.
- Compile code files.
- Unit testing of code.
- Create project documentation.
- Create project reports.



26. What are the types of Maven Plugins?

There are two types of Maven Plugins:

- Build plugins – These plugins are executed during the build and are configured in the `<build/>` element of `pom.xml`
- Reporting plugins – These plugins are executed during the stage generation and are configured in the `<reporting/>` element of the `pom.xml`.

27. What is the difference between Convention and Configuration in Maven?

Convention	Configuration
<ul style="list-style-type: none"> • The convention is when the developers are not required to create the build processes. • The users do not have to specify the configuration in detail, and once the project is created it will automatically create a structure. 	<ul style="list-style-type: none"> • The configuration is when developers are supposed to create the build processes manually. • They must specify every configuration in detail.

28. Why is it said that “Maven uses convention over configuration”?

- Maven uses convention instead of configuration, due to the fact that the developers only need to create a Maven project. The rest of the structure is automatically created.
- There are many conventions present in Maven for setting up a project, building the artifacts, releasing the code, and running unit tests.

29. What is Maven’s order of inheritance?

The order of inheritance in Maven is:

- Settings
- CLI parameters
- Parent POM

- Project POM

30. What do Build Life Cycles and phases imply in the basic concepts of Maven?

- Build Life Cycles consist of a sequence of build phases, and each build phase consists of a series of goals.
- When a stage is run, all the purposes related to that phase, and its plugins are also compiled.



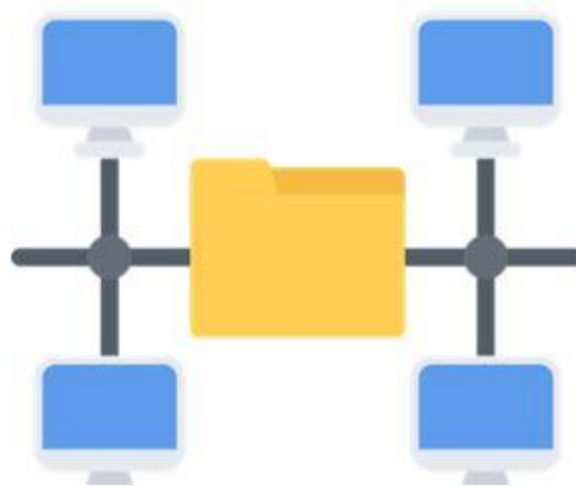
31. What is the 'Goal' in the Maven terminology?

- The term goal refers to a specific task that makes it possible for the project to be built and organized.
- Phases that are a stage in the life cycle define the sequence in which the desired goals are to be accomplished.

32. What is meant by the term 'Dependencies and Repositories' in Maven?

- Dependencies refer to the Java libraries that are needed for the project. Repositories refer to the directories of packaged JAR files.

- If the dependencies are not present in your local repository; then Maven downloads them from a central repository and stores them in the local repository.



33. What is a 'Snapshot' in Maven?

Snapshot refers to the version available in the Maven remote repository. It signals the latest development copy. Maven inspects for a new version of Snapshot in the remote repository, for every new build.

The snapshot is updated by the data service team with an updated source code every time to the repository for each Maven build.

34. What types of projects are available in Maven?

- There are thousands of Java projects provided as templates by Maven.
- This helps the user, as they no longer have to remember every configuration to set up a particular project.
- For example Spring Project, Spring MVC, Spring Boot, etc.

35. What is a Maven Archetype?

- Maven Archetype refers to a Maven plugin entitled to create a project structure as per its template.
- These archetypes are just project templates that are generated by Maven when any new project is created.

Now that we are done with some Intermediate Maven Interview Questions, let's move on to some Advanced Maven Interview Questions.

Advanced Maven Interview Questions

36. What is the command to create a new project based on an archetype?

- The command – `mvn archetype:generate`, is used to create a new project based on an archetype.
- The command is typed in the command prompt, after navigating to the directory in which the project is residing.

37. What does 'Maven Clean' imply?

- Maven clean is a plugin as the name suggests, that approaches to clean the files and directories generated by Maven at the time of its build.
- The plugin removes the target folder that contains all the class files, docs, JAR files.



38. What is a Build Profile?

- Build profiles refer to the set of configuration values required to build a project using different configurations.
- Different build profiles are added to the POM files while enabling different builds.
- A build profile helps in customizing build for different environments.

39. What are the different types of Build Profiles?

There are three different types of build profiles:

- Per Project - defined in pom.xml
- Per - User - defined in Maven settings.xml
- Global - defined in Maven global settings.xml

40. What is meant by the term 'System Dependency'?

System dependency refers to the dependency that is present with the scope system. These dependencies are commonly used to help Maven know the dependencies that are provided by the JDK or VM.

System dependencies are mostly used to resolve dependencies on artifacts that are provided by the JDK.



41. What is the reason for using an Optional Dependency?

- Optional dependencies are used to decrease the transitive burden of some libraries.
- These dependencies are used when it is not feasible to divide a project into sub-modules.
- Some dependencies are only used for a specific feature in the project, and if that feature is not there, then that dependency will not be used.

42. What is the 'Dependency Scope', and how many types of Dependency Scopes are there?

Dependency scope refers to all the dependencies about the current stage of the build.

The different types of dependency scopes are:

- Compile

- Provided
- Runtime
- Test
- System
- Import



43. What is meant by 'Transitive Dependency' in Maven?

- Maven avoids the need to find out and specify libraries that the dependencies require, by including transitive dependencies automatically.
- Transitive dependency says that if X depends on Y and Y depends on Z, then X depends on both Y and Z.

44. How can a Maven Build Profile be activated?

A Maven build profile may be activated in the following ways:

- Explicitly using command console input.

- Through Maven settings.
- Based on environment variables.
- OS Settings.
- Present/missing files.

45. What is meant by 'Dependency Exclusion'?

The exclusion element is used to exclude any transitive dependency. The concept says if X depends on Y and Y depends on Z, then X can be marked Z as excluded.

Learn all about the need for the cloud in DevOps with the [DevOps Engineer Certification Training Course](#). Enroll now

46. What is MOJO?

MOJO can be defined as a Maven plain Old Java Object. Every MOJO is an executable goal in Maven, and a plugin refers to the distribution of such MOJOs. MOJO enables Maven to extend the functionality that is already not found in it.

47. What is the command to produce a new project based on a hard drive?

- -mvn archetype: create is used to create a new project.
- The archetype is created after reading the source and resource file, the values of its parameters, and other properties.



48. What is the Maven settings.xml file?

The Maven settings.xml file contains elements used to define the values needed to configure Maven execution differently.

It contains the following configurations:

- Proxy configuration
- Local repository configuration
- Remote repository configuration
- Central repository configuration

49. What is meant by the term 'Super POM'?

Super POM refers to the default POM of Maven. The POMs of Maven can be derived from a parent or by default. To execute any particular objectives, effective POM is used.

Super POM supports the developers to configure the pom.xml file with the least configurations.



50. Where are Maven dependencies stored?

- All the JARS, dependency files, etc. that are downloaded by Maven are saved in the Maven local repository.
- The Maven local repository is a folder location on the local system where all the artifacts are locally stored.