**3. Maven**

1. Why do we need a build tool like maven?

Maven is a powerful build automation tool primarily used for Java projects. It simplifies and standardizes the build process, making it easier to manage dependencies, compile source code, run tests, and package the application.

1. Install maven
2. Display output of maven version
3. What is the pom.xml file?

The pom.xml (Project Object Model) file is the core of Maven-based projects. It is an XML file that contains project configuration and metadata, including project dependencies, build settings, plugins, and other project-related information.

1. Explain these tags found in pom.xml files?

|  |  |
| --- | --- |
| groupId | This element defines a unique identifier for the group or organization that owns the project. |
| artifactId | This element specifies the name of the project's artifact (e.g., JAR file, WAR file). It represents the name of the project module or library. |
| version | This element indicates the version of the project artifact. It specifies the current version of the project module or library. |
| packaging | This element specifies the type of packaging used for the project artifact. It defines how the project is packaged into a distributable format. |
| dependencies | This element contains a list of project dependencies. Dependencies are external libraries or modules required by the project to compile, run, or test. |
| dependency | This element defines a single project dependency within the dependencies section of the pom.xml file. |
| properties | This element allows you to define custom properties or variables that can be referenced throughout the pom.xml file. |

1. Create a method which accept an integer as parameter and returns the square of it
2. Add junit (v5) dependency
3. Add a unit test to test the method
4. Run unit tests with maven. What is the command you used?
5. Create a Student class with following attributes and add getters/setters

* id: int
* name: String
* age: int
* subjects: List<String>

1. In your main method create and student object and set these values

id: 1

name: “john”

age: 20

subjects: [“Maths”, “English”, “History”]

1. Then print these student values using getters (e.g: s.getName())
2. Add lombok dependency and remove getter/setter methods from Student class
3. Explain the usage of these commands

|  |  |
| --- | --- |
| mvn clean | This command is used to clean the project by deleting the target directory, which contains the compiled bytecode, packaged JAR/WAR files, and other generated files. |
| mvn install | This command is used to compile the source code, run the tests, and package the project |
| mvn package | This command is used to compile the source code, run the tests, and package the project.  It generates the target artifact (JAR, WAR, etc.) containing the compiled bytecode and resources, without installing it into the local repository. |

1. Explain 3 types of maven repositories

* local repository : The local repository is located on the developer's machine and is typically stored in the .m2 directory in the user's home directory (~/.m2/repository). When Maven downloads dependencies from a remote repository, it caches them in the local repository. The local repository is used to store project dependencies locally on the developer's machine, allowing Maven to reuse dependencies across different projects and reducing the need for downloading dependencies repeatedly.
* Central repository : The Central Repository is the default remote repository for Maven, hosted by the Maven community. When Maven cannot find a dependency in the local repository, it automatically searches for it in the Central Repository and downloads it if found.
* Remote repository : Remote repositories are additional repositories hosted by organizations or individuals to provide specific dependencies or custom artifacts not available in the Central Repository. Organizations may host their own remote repositories to manage and distribute internal libraries, proprietary dependencies, or third-party artifacts that are not publicly available.

1. Add your codes and answer sheet to a directory named “maven-basic-training” and push it to your training github repository.