Maven Build Life cycle:

* It is a fundamental concept that defines the sequence of phases and goals executed to build and manage Maven project.

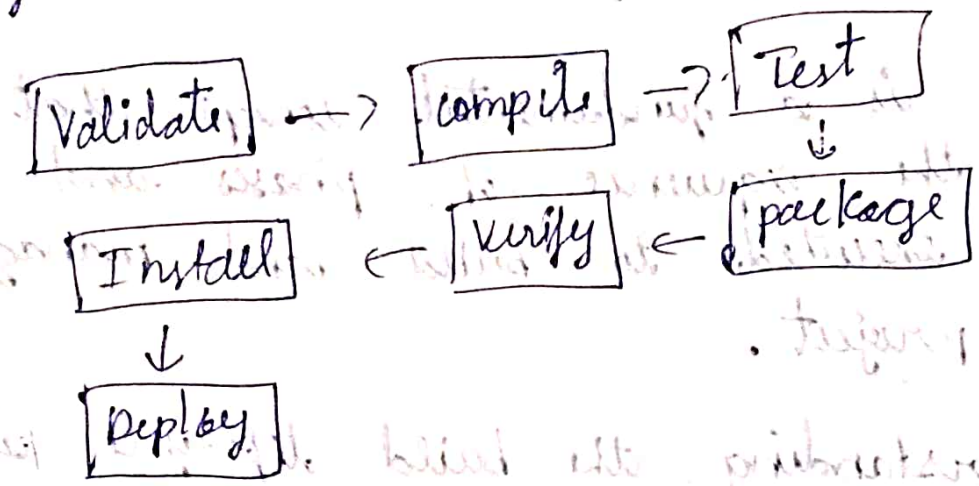
* Understanding the build lifecycle is crucial for efficiently managing the build process, from compilation to deployment.

* It includes a lot of build phases to complete the build process in the Maven.

Types of Maven Build Lifecycle:

1. Default: This is main life cycle, as it's responsible for project deployment.
2. clean: Handles project cleaning, ensuring that all artifacts generated by previous builds are removed.
3. site: Manages the creation of the project's site documentation.

Diagram :



1. Validate Phase :

* The First phase of the Maven build lifecycle.
* Ensures that the project structure and configuration are correct.
* Check whether all necessary dependencies are available.
* Verifies if the pom.xml (Project Object Model) file is correctly defined.
* If any required configuration is missing, Maven will stop execution with an error.

2. compile phase:

* compiles the source code of the project
* Uses the compiler plugin (by default, the Java compiler).
* The compiled .class files are placed in the target directory.
* If the source code has syntax errors, Maven will stop the build process.

3. Test phase:

* Runs unit tests using a testing framework like JUnit or TestNG
* Ensures that the code functions as expected before packaging.
* Does not require the full application to be packaged or deployed.

4. Package phase:

* packages the compiled code and dependencies into a distributable format.
* Generates JAR (Java Archive) or WAR (Web Application Archive) files.
* The output file is placed in the target directory.
* The compiled files and dependencies are bundled together.
* If the project is a:
  - → Library, it produces a .jar file.
  - → Web application, it produces a .war file.

5. Verify phase:

* Runs additional checks to ensure the package is valid.
* May include integration tests, code analysis, or security scans.

* Ensures that the package meets Quality standards before installation.

6. Install phase

* Installs the built package into the local Maven repository (~/.m2/ repository)
* other local projects can use this package as a dependency.

7. Deploy phase :

* uploads the package to a remote Maven repository (eg., Nexus, Artifactory, or maven central)
* Makes it available for other developers and projects.

Maven commands:

1. mvn validate
2. mvn compile
3. mvn test
4. mvn package
5. mvn verify
6. mvn install
7. mvn deploy

Clean Lifecycle:

mvn pre-clean : Executes processes needed before cleaning.

mvn post-clean : Executes processes needed after cleaning.

mvn clean : Removes files generated by the previous build.

site Lifecycle:

mvn site-deploy : Uploads the site online.

mvn pre-site - prepares for site generation

mvn site - creates project documentation

mvn post-site - Finalizes the site.