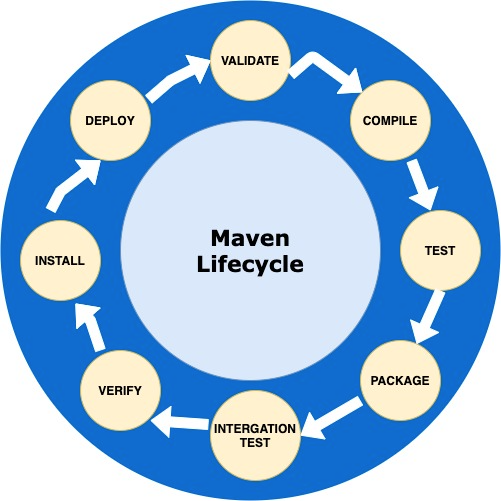
**1. Maven Life Cycle (10 Marks)**

Maven is a build automation tool used in Java-based projects, including Adobe Experience Manager (AEM). It has three major life cycles:

**Default Life Cycle**

This is the primary life cycle that compiles, tests, packages, and deploys the project. It includes phases such as:

* validate – Checks if the project is correct.
* compile – Compiles source code.
* test – Runs unit tests.
* package – Bundles the code (JAR/WAR).
* install – Installs the package in the local repository.
* deploy – Deploys the build to a remote repository.



**2. What is pom.xml and Why We Use It?**

* pom.xml (Project Object Model) is the **configuration file** for Maven projects.
* It defines project dependencies, plugins, build settings, and modules.
* It helps in **automating** the build process in **AEM development**.

**3. How Dependencies Work in Maven?**

* Dependencies are **external libraries** required by your project.
* They are **defined in pom.xml** and fetched automatically from **Maven Repository**.
* Example:

<dependencies>

<dependency>

<groupId>org.apache.sling</groupId>

<artifactId>org.apache.sling.api</artifactId>

<version>2.20.0</version>

</dependency>

</dependencies>

**4. Checking the Maven Repository**

* The **Maven Central Repository** is available at:  
  🔗 <https://mvnrepository.com/>
* You can search for dependencies and add them to pom.xml.

**5. How All Modules Build Using Maven?**

* In **AEM**, multiple modules are built using:

mvn clean install -PautoInstallPackage

* This command **compiles** all modules, installs dependencies, and deploys them to AEM.

**6. Can We Build a Specific Module?**

Yes, you can build a **specific module** using:

cd ui.apps

mvn clean install

This will **only build** the ui.apps module.

**7. Role of ui.apps, ui.content, and ui.frontend Folders in AEM**

| **Folder** | **Purpose** |
| --- | --- |
| ui.apps | Stores AEM **components, templates, dialogs, and OSGi configs**. |
| ui.content | Stores **AEM content, pages, and default configurations**. |
| ui.frontend | Stores **CSS, JavaScript, and client-side libraries** for styling and UI. |
| **8. Why Are We Using Run Mode?**  Run modes in AEM help in **configuring different environments** (e.g., development, testing, production). Example run modes:   * **author** → For content authors * **publish** → For end-users (live site) * **dev**, stage, prod → For different environments   **Example Configurations:**   * config.author/com.day.cq.wcm.core.impl.PageManagerFactoryImpl.cfg.json   config.publish/com.day.cq.replication.impl.ReplicationAgentManagerImpl.cfg.json |  |

**9. What is the Publish Environment in AEM?**

* The publish environment is where the final website is visible to end users.
* Content is replicated from Author to Publish.
* Example: A news article is created in the Author instance, and once approved, it is published to the live site.

**10. Why Are We Using Dispatcher in AEM?**

Dispatcher is an AEM caching and load-balancing tool that:

Caches static content to improve performance  
Filters requests for security  
Balances load between multiple AEM instances

It helps in handling high traffic and reducing server load.