Apache Maven:-

- Apache Maven is a widely used open source build automation and project management tool In software development.
- Maven uses a declarative approach through an XML-based configuration file (pom.xml) to define project settings and dependencies.
- Developers use Maven to define the build process, maintain project structure, and manage dependencies, making it easier to collaborate on software projects.

Maven Archetype:

- In Maven, an archetype is a project templating toolkit used to create new Maven projects.
- It provides a way to generate the initial project structure, including the standard directory layout, default files, and necessary configurations for the technology you are working on.
- Maven archetypes make it easier to start new projects by eliminating the need to manually set up the project structure and configuration files.

pom.xml.

- pom stands for Project Object Model.
- It is a configuration file that helps in declaring the dependencies, plugins and other configuration details.
- The pom.xml automatically downloads the required dependencies and plugins, based on the declaration.

Steps to create a Maven Project:-

1. Click on File > New => Maven Project

OR

Click on File > Others => search for maven => select Maven Project => click on finish

2. In the 1st dialogue box click on next.

3. In the 2nd dialogue box we need to select the "Catalog" as Internal and select the archetype maven-archetype-quickstart (version 1.1).

4. In the 3rd dialogue box we need to set the Group-Id, Artifact-Id and also the basepackage name if required.

5. Then click on finish.

6. After Maven has created the project we need to type y in the console(Y:: y) to load the project into the workspace.

Key Features

1. Dependencies Management
    - Maven simplifies dependency management By allowing you to declare the dependencies in xml format in pom.xml file.
    - Maven Utilizes a centralized repository called Maven Repository, to store and share Project dependencies.
    - It automatically downloads the required dependencies by reading or utilizing the declaration.

2. IDE Support:-

- Maven integrates seamlessly with popular IDEs such Eclipse, Spring Tool Suite an IntelliJ IDEA, This helps developers to build and maintain the maven projects in their prefered IDEs

3. Archetypes Integration:-

- Maven archetypes are the templates or a blueprint used by maven to build the project.
- They provides an initial project structure with directory layouts, configurations files etc.

4. Standardized Structure:-

- Maven archetypes lets you build a standard project structure and lets you organise the code in a systematic way, which makes the codebase more readable and manageable.

Servers

- A server is a large computer system or a small software program that provides services or resources to other computers, known as clients, over a network.
- It is the one which serves client request.
- For a server to serve a client request, resource must be available.
- Types of web servers:-

    1. Web server

    2.Application server

    3.Database server

1. Web Servers:-

- A web server handles HTTP requests from clients, typically web browsers, and delivers web content like HTML.
- It serves static content like HTML pages, Images, and files.
  Ex:- Apache HTTP server, Caddy, Nginx, LiteSpeed web server.

2. Application Servers:-

- An application server is responsible for executiong application code and business logic
- It handles dynamic content generation and often interacts with databases.
- Common tasks include transaction management, security and connection pooling.
  Ex:- Apache Tomcat, WildFly and IBM WebSphere.

3. Database Servers:-

- A database server manages and provides access to a database.
- It stores, retrieves, and manipulates data based on requests from application servers or other clients.
  Ex: MySQL, PostgreSQL, Oracle Database, MS SQL server.

Steps to add Server to the Eclipse IDE:-

1. On the top right corner of the IDE click on search button.

2. Search for "Servers" and click on Enter.

3. In the "Servers" tab click on the option "Click this link to create a new server".

4. In the 1st dialogue box select the server (Apache Tomcat v9.0) and click on Next.

5. In the 2nd dialogue box click on "Browse" button and specify the path of the destination folder where Tomcat server is downloaded.

6. Click on Finish.

Properties File in JDBC:-

- A properties file in JDBC is a text file used to store configuration details required to connect with a databse.
- This file typically includes key-value pair specifying the necessary connection details such as database url, username and password.
- It is loaded using the Properties class of java present in java.util package.

Advantages:-

1. Separation of Configuration and Code:
   Storing database configuration parameters in a properties file helps keep the code clean and maintainable by separating configuration details from the application logic.

2. Easier Maintenance:
   Changes to the database connection details can be made directly in The properties file without altering the java code.

3. Security:
   Sensitive information such as database passwords can be managed more securely, and access can be restricted to the properties file.

Steps to read data from the properties file:-

- Create a file with ".properties" extension directly inside the project.
- In the Java Application create an object of FileReader class and pass the configuration file name inside the constructor in order to read it inside the java code.
- Then create an instance of Properties class which is needed to lead the content present inside a properties file.
- Next invoke the load() method using the Instance of Properties class and pass the FileReader object reference as the parameter in order to load it to the Properties object.
- If you have some properties defined inside the file whose key won't be recognized by the Properties object then you have to manually load it using the getProperty() method using Properties object and pass the key as parameter.
- After that invoke getConnection() method accepting 2 parameters and pass the dbURL and Properties object ref as parameter.

Eg:- Connection conn = DriverManager.getConnection(String dbURL, Properties p);