SERVLET

Servlet is a Server side Java program, Which performs all the three different types of Logics,

1. Presentation Logic
2. Persistence Logic
3. Business Logic

Along with which it process the http client request and get back some http response.

TYPES OF SERVLET:

Two Types,

- Generic Servlet
- Http Servlet

CRITERIA FOR SERVLET:

- Create Servlet
- Configure Servlet
- Deploy Servlet

CONFIGURE THE SERVLET:

Servlet-Name
URL-Pattern
Fully Qualified Class Name(Servlet-Class)

SERVER:

Server is a Software which manages all the resources along with which it process the client request and serve the client request.

TYPES OF SERVER:

- **Database Server**

Used to deal with only data.

Eg: Oracle, MySQL, MSSQL,Derby.

- **Web-Server**

Used to to execute only the web applications.

Eg:
- Apache=Tomcat Server -----> ***It is basically a web-server, but internally it works as Application server to execute both Web and Enterprise Applications.***
- Oracle GlassFish
- Jetty
- WildFly

- Application Server:

Used to execute Dynamic/Real-time applications.
Used to execute both Web and Enterprise Applications.

Eg:
- JBOSS
- IBM Web-Sphere
- Oracle Web-Logic

DEPLOYMENT:

Process of making all the resources available to the Server.

Two Ways,

- Manual Deployment
- Automated Deployment

Automated Deployment:

Here we are using Automated Tools such as, ANT, MAVEN etc….

Using these automated tools we can create the patch files, which can be accessed through Double Click.
All the Web Applications are deployed it onto the WEB APPS Folder.

APACHE TOMCAT SERVER:

Two Variants are there,

1. Executable variants(.exe)
2. Zip variants

SERVLET CONTAINER NAME: ***CATALINA***

JEE CONTAINER:

JEE Container is used to manage all the JEE Components such as,

Servlet
JSP
EJB(Enterprise Java Bean)

JEE APPLICATION:

It is the specification to develop both web and enterprise applications, which is given in the form of Abstraction API to achieve Loose coupling between Java application and Database server.

STRUCTURE OF JEE APPLICATION:

WELCOME FILE AND LANDING PAGE:

The Page which displayed automatically whenever client uses an application.

INDEX:

Index is the default welcome file or Landing page, sice it is automatically loaded by the JEE Container.

WEB.xml:

- Deployment Descriptor Stub.
- Which is used to store the information with respect to the configurable resources of an application.
- Each and Every application must mandatorily have only one web.xml, without which JEE container fails to load an application where it throws http 404 error.
- Current version of xml is 1.0
- The root tag of web.xml is <web-app>

8-Bit Unicode FORMAT.

Generic servlet:

It is not specific to any protocol.
It does not support Session.

Generic Servlet is present in javax.sql package.

It contains Three methods in it,

1) Service(ServletRequest req, ServletResponse resp) (Abstract Method)
2) Init(Servlet config) (Concrete Method)
3) Destroy() (Concrete Method)

We have to override Service() for two important reasons,

- Since Service() is a abstract method, we have to override it.
- Service() method is the only method which takes ServletRequest and ServletResponse as a parameter which is responsible for processing the client request.

HTTP Servlet:

- It is specific to a particular type of protocol called HTTP Protocol.
- HTTP Servlet contains only Concrete methods in it.
- We have to override doXXX() method for particular type of http request.

8 different type of http request are present,

Post

Get
Put
Remove
Head
Connect
Trace
Option

Web resources can be accessed based on the Unique url pattern.
Since Servlet is also a web resource which can be accessed based on Unique url pattern.

Any resources can be configured in two ways,

1. Using web.xml
2. Using Annotation

UI/FORM DATA:

The data which is entered by the end user on a form page and submitted to the server in the form of key and value pair is known as UI/Form data.

SERVLET LIFE CYCLE:

- Servlet gets a life and begins the cycle only when the ***Servlet object is created***.

- Servlet life cycle represents the events or phases which takes place from Servlet object creation until Servlet Object destruction.
- Entire Servlet life cycle is managed by JEE Container.

FOUR PHASES OF SERVLET LIFE CYCLE:

1. Instantiation/Object Creation
2. Initialization
3. Service
4. Destruction

INSTANTIATION/OBJECT CREATION:

- ✓ In this phase the Servlet object has to be created.
- ✓ Whenever client makes the First request to the servlet one **Servlet object is created by the JEE Container** by calling the **Default Constructor of the Servlet class**, then the Servlet Life cycle begins.
- ✓ If the JEE Container does not find the default constructor, then it throws Servlet Exception with the root cause of InstantiationException.
- ✓ Immediately after the object creation of Servlet class One Servlet config object is created which is used to initialize the resources of that particular Servlet Object.
- ✓ The scope of Servlet config object is always limited to that particular Servlet object.

INITIALIZATION:

In this phase the Servlet object has to be initialized.
The Servlet object has to be initialized using init(Servlet Config) method which takes Servlet Config as a parameter which is used to initialize the resources of the particular Servlet Object.

Init() method is called by the JEE Container only once.
If this phase fails, then the JEE Container will throw **ServletException**.

SERVICE PHASE:

- In this phase the Service() method is called by the JEE Container which is responsible processing each and every client request.
- In this phase JEE Container create one request object and response object for each and every client request including the first client request.
- By default, the servlet is multi-threaded but it can be made as Single-threaded in two ways,
- By writing Servlet class by implementing a marker interface called SingleThreadedModel(Deprecated Interface).
- By making the Service() method as Synchronized.
- Whenever client makes Second or Subsequent request to the same SERVLET only the Service() will execute but Servlet Object won't be created again, because it created for First Request itself.
- The Service() method is called by the JEE Container for multiple times.
- If this phase fails JEE Container throws ServletException.

DESTRUCTION PHASE:

- In this method the destroy() method is called by the JEE Container to close all the costly resources, but not to destroy the Servlet Object.
- The destroy method is called by the JEE Container only once.
- If this phase fails the performance of the application decreases.

Load-on-startup:

- Whenever we Start the Server, the Servlet object is created by the JEE Container by calling the default constructor of the Servlet without waiting for the First Client Request.
- So the delay time made by the First Client Request can be avoided. It increases the performance of the Application.

<Load-on-Startup> ------> It is a Sub-tag of Servlet Tag.

- In case of load-on-startup only service() method will be executed even for the first client request.

- Load-on-Startup must be configured with the Positive Integer Value.
- If two servlets are configured with ***same positive integer value, then Sequential execution takes place.***
- Load-on-Startup with least positive integer value will execute first.
- Whenever Load-on-Startup is ***configured with negative value,*** then the servlet object is created based on the ***First Line Request***.

HTTP Get and Post:

S.No	Get	Post
1	Get request is used to get some contents from the server.	Post request is used to post some contents/ dat from the client to the server.
2	It is safe.	It is not safe.

3	It can be Bookmarked.	It cannot be Bookmarked.
4	It is Idempotent.	It is Non-Idempotent.
5	In case of get request the datas are carried to the server as a part of request object in the form of key and value pair, which is displayed in the URL. So, it is not Secured.	In case of Post request the datas are carried to the server as a part of http request body, which is not displayed even to the end-user. So, it is Secured.
6	Deals with limited data, that is 1024 Characters.	Deals with Unlimited data.