**HISTORY OF JAVA WEB SERVLET and JavaServer Pages (JSP)**

* Java is built naturally with internet functionality. In **June 1997**, Sun Microsystems announced the Java Servlet.
* Java Servlet is one of the fundamental building block in developing different components of a Java Web specifically mainstream server-side Java which runs in a single process by using grain threads.
* While in **1998,** Sun released the JavaServer Pages **(JSP).**
* JSP is used to easily code a dynamic content of the web’s HTML pages.
* Both Servlets and JSPs lets the programmers build or develop a portable, easy to maintain, modular and scalable Web applications.

**THE JAVA SERVLET**

* In web applications, servlets were the first one which got the full access and power of Java. Wherein just like applets, it is completely and fully written in Java programming language.
* It is used for extending the server’s capability to host different applications that are accessed through a programming model, the request-response model. Which means it takes the request of the client and generates the response to the request.
* Java Servlet’s version 2.4 (current) which is included in the Java 2 Enterprise Edition (J2EE). Through the Tomcat project, it is available for free and also an open source.

**Features of Servlet 2.4:**

* Web Applications: In this section, servlet is always part of it (the web application), wherein it provides all the resources of a website.

**Servlet Container** – it manages all the Servlets in the basis of Web application.

*The web container facilitates the conversion to and from the HTTP request/response message to HTTP Servlet Request/HTTP Servlet Response*.

* Servlets and HTTP Servlets:
* Java servlet used to handle HTTP request and generate HTTP responses
* Nested in web container which is a component in Java Application server

e.g. Apache Tomcat, Red Hat JBoss/Wildfly, Oracle Glassfish, IBM WebSphere

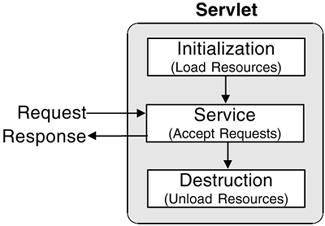
* the service() method call is routed to a doXXX() call, depending on the HTTP request method,

*e.g. doGet(), doPost(), doHead(), doPut(), doDelete(), etc…)*

* the doXXX() methods are passed two arguments:
  + HTTP Servlet Request
    - Object representation of the HTTP request sent by the client and received by the servlet.

*e.g. HTTP request method, request URL, query string, message headers, message payload, etc.*

* HTTP Servlet Response
  + Object representation of the HTTP response generated by the servlet and sent back to the requesting client
  + Used to set information in the response message *e.g. HTTP response status code, message headers, message payload, etc.*

* Filters:
* Security:
* Internalization:
* Servlet Life Cycle

1. *Instantiation:*

* The web container created the instance only once in the cycle.

1. *Initialization:* **init() method is invoked**

* This phase represents the creation of different resources to service requests. The init() method is invoked only once and before servicing of request takes place, the servlet invokes it first.

Syntax:

public void init(ServletConfig config) throws ServletException

1. *Request Handling:* **service() method is invoked**

* This phase represents all requests interactions invoked by each client.

Two parameters of service() method: Represents client’s request and Servlets’s client’s response.

1. javax.servlet.ServletRequest
2. javax.servlet.ServletResponse

Syntax:

public void service(ServletRequest request, ServletResponse response)

  throws ServletException, IOException

1. *Destruction:* **destroy() method is invoked**

* Represents when the Servlet is removed in a container, simply, the destruction of a life cycle phase. The container calls the method destry() and terminate the resources that have been created.

Syntax:

public void destroy()