Introduction to Servlets Quiz

1. What is the difference between a web server and a web container?

Ans) The Webserver: Just Receive the request from the client and forward to Web container, and do the Response vice versa. Web Container: Creates HTTP Request and Response for the Servlet, calls the appropriate servlet service method (do Get or do Post) to service the client request. Container also gives: Communication support(socket creation), Servlet life cycle management (Init(), Service() and Destroy()), Multithreading support, Declarative security(thru deployment descriptor) and JSP support.

2. What is a servlet?

- Servlet is a technology which is used to create a web application.
- Servlet is an API that provides many interfaces and classes including documentation.
- Servlet is an interface that must be implemented for creating any Servlet.
- Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.
- Servlet is a web component that is deployed on the server to create a dynamic web page.

3. How do web servers and web containers interact with servlets?

Ans:- First, the container loads the servlet's class: • Can happen at container startup or the first time the client invokes the servlet. Container creates an instance of the servlet class (using its constructor) • Container then invokes the init() method.st, the container loads the servlet's class: • Can happen at container startup or the first time the client invokes the servlet.• Container creates an instance of the servlet class (using its constructor) • Container then invokes the init() method. When a client request arrives at the container, container uses the URL to determine which servlet to use. • Container creates request and response objects

4 Who creates request objects?

Ans:- The servlet container locates the servlet, creates request and response objects and passes them to the servlet, and returns to the web server the response stream that the servlet produces.

5. What are the states in the servlet lifecycle?

Ans:- The web container maintains the life cycle of a servlet instance. Let's see the life cycle of the servlet:

- Servlet class is loaded.
- Servlet instance is created.
- init method is invoked.

- service method is invoked.
- destroy method is invoked.

6. Who calls init and when?

Ans:- The web container calls the init method only once after creating the servlet instance. The ini initialize the servlet. It is the life cycle method of the javax.servlet.Servlet interface. Syntax of the below:

public void init(ServletConfig config) **throws** ServletException

7. Which of init, service, and doGet should you override?

Ans:- Instead of overriding `init(ServletConfig)`, simply override the below method and it will be called by `GenericServlet.init(ServletConfig config)`

```
Public void init() throws ServletException {
ServletContext sc= getServletContext();
}
```

8. In what sense are servlets multi-threaded?

Ans:- The web container creates threads for handling the multiple requests to the Servlet. Threads have many benefits over the Processes such as they share a common memory area, lightweight, cost of communication between the threads are low. The advantages of Servlet are as follows:

- Better performance**:** because it creates a thread for each request, not process.
- Portability: because it uses Java language.
- Robust: JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.
- Secure: because it uses java language.

9. What are the implications of this for servlet instance variables?

Ans:- The web container creates the instance of a servlet after loading the servlet class. The servlet instance is created only once in the servlet life cycle.