**Introduction to Servlets Quiz**

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

Web Server or HTTP Server: A server which is capable of handling HTTP requests, sent by a client and respond back with a HTTP response. Web server serves static contents.

Web container is used to manage the components like Servlets, JSP. It is a part of the web server. It serves dynamic contents.

The Container gives your web app communications support, lifecycle management, multithreading support, declarative security, and support for JSPs, so that you can concentrate on your own business logic.

The Container creates a request and response object that servlets (and other parts of the web app) can use to get information about the request and send information to the client.

1. What is a servlet?

A servlet is server-side java code that can handle http requests and return dynamic content. Servlets are managed by a servlet engine or container.

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

When a request comes to the web server, if the server sees the request is for a servlet, it passes the request data to the servlet container.

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.

The web server sends the response back to the client browser to be rendered.

1. Who creates request objects?

Container receives new request for a servlet and Creates HttpServletRequest and HttpServletResponse objects

• Calls service method on HttpServlet object in thread

• When thread completes, converts response object into HTTP response message

1. What are the states in the servlet lifecycle?

Servlet states: Load, Create, Init, Service, Destroy

1. Who calls init and when?

Container calls init to initialize a servlet after instantiate servlet and it must complete before Container can call service().

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

Init() -> Gives you a chance to initialize your servlet before handling any client requests. ***We can override****.*

Service()-> This method looks at the request, determines the HTTP method (GET, POST, etc.) and invokes the matching doGet(), doPost(), etc. on the servlet. ***We should NOT override the service() method.***

doGet() or doPost() ***->ALWAYS at least ONE of them!***

1. In what sense are servlets multi-threaded?

The Container runs multiple threads to process multiple requests to a single servlet. And every client request generates a new pair of request and response objects.

Each client gets a separate thread for each request, and the Container allocates new request and response objects.

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