Servlets Quiz

1. What is the difference between a web server and a web container?
   Answer :
   Web server: is a program that uses HTTP to serve the files that form **Web** pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. On the other hand, **Web container** serves static content -files, images, pdfs, videos, exactly as they are on the server machine. It manages Servlets where each request that comes in results in the spawning of a new thread that runs a servlet.

2. What is a servlet?
   Answer: *-A servlet is a Java class that extends the capabilities of servers that host applications access by means of a request-response programming model.*
   -A servlet is server-side java code that can handle http requests and return dynamic content.
3. How do **web servers** and **web containers** interact with servlets?
   Answer: Web server accepts HTTP request from a client. And then it forwards to container for services. It Creates HttpServletRequest and HttpServletResponse objects. Calls service method on HttpServlet object in thread. When thread completes, converts response object into HTTP response message. Finally the web server respond to the client request.

4. Who creates request objects?
   Answer: Web Container Creates HttpServletRequest and HttpServletResponse objects.
5. What are the states in the servlet lifecycle?
   Answer:
   a. Load servlet class
   b. Instantiate servlet
   c. init() called only once in the servlet's life.
   d. service() (called for each request, each request runs in a separate thread)
   e. destroy() (called only once)

6. Who calls init and when?
   f. **Answer**: init is performed by Servlet and must complete before servlet container can call service().

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

8. In what sense are servlets multi-threaded?
   Answer: There is always one instance of servlet and new thread created for every request. Then service () called on the thread. All threads share instance variables, but each thread has own stack for local variables.

9. What are the implications of **this** for servlet instance variables?
   Answer: it works similar to the usual java class where we can differentiate instant variable from methods or constructor arguments(parameters).