1. What is the difference between a web server and a web container?
Web server serves static contents like HTML and images. It handles HTTP request sent by a client and responds back with HTTP response.

Web container is a part of web server which knows the life cycle of servlets, JSP, and web services and respond back to the browser.

2. 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. Each request that comes in results in the spawning of a new thread that runs a servlet (eliminating the cost of creating a new process every time).

3. How do web servers and web containers interact with servlets?
Client sends HTTP request to web servers. Servlets get those requests through web containers, and send HTTP response to web servers via web containers in order to display in browser.

Web container manages and runs servlets.

- It provides communication support between servlet and web servers.
- The container controls life cycle of servlets (loading the classes, instantiating, and initializing the servlets, invoking the servlet methods, and making servlet methods, and destroying servlet instances) .
- The container automatically creates a new Java thread for every servlet request it receives.
- With a container, you get to use an XML deployment descriptor to configure (and modify) security without having to hard-code it into servlet class code.

4. Who creates request objects?
Web container creates request objects.

5. What are the states in the servlet lifecycle?

- Servlet class loading
- Servlet instantiation
- call init() method
- Call service() method
- Call destroy() method

6. Who calls init and when?
Web container calls init after a servlet instance is created and before it can service any client requests.

7. Which of init, service, and doGet should you override?
We should always override doGet() method.

8. In what sense are servlets multi-threaded?
Every client request generates a new pair of request and response objects. If container has more than one request at the same time, it runs multiple threads to process multiple requests to a single servlet.

9. What are the implications of this for servlet instance variables?
All the threads share the same instance variables.