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

Web Server is capable of handling HTTP requests, sent by client and respond back with HTTP response. Web Container is the component of a web server that interacts with java servlets.

1. **What is a servlet?**

It is a Java class that is used to extend the capabilities of servers.

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

Firstly, the web server receives requests by the client and delegate these requests to the servlet container. Then, the servlet container loads the servlet. And the servlet container handles multiple requests for the same servlet by spawning multiple threads, one thread per request, each executing the service() method of a single instance of the servlet.

**4. Who creates request objects?**

The servlet container.

**5. What are the states in the servlet lifecycle?**

Initialization, Processing and Destroy.

**6. Who calls init and when?**

The servlet container calls the init and it may happen at the container startup or at the first time the client invokes the servlet with a request.

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

doGet, because the init it’s called to initialize the servlet and the service method invokes the doGet or doPost method, that you should override.

8. **In what sense are servlets multi-threaded?**

In the way that the servlet container handles multiple requests for the same servlet by spawning multiple threads, one thread per request, each executing the service() method of a single instance of the servlet.

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

The servlet has a single instance and each instance variables is shared for all the requests, so these variables are not thread safe.