## 29 Lecture - CS506

## **Important Subjective**

Certainly, here are 10 short-answer questions related to advanced topics in servlets along with their answer
**Question 1:** What is session tracking in servlets?
**Answer:** Session tracking involves maintaining user-specific data across multiple requests and sessions. It's crucial for preserving state in stateless HTTP protocol.
**Question 2:** How is session management achieved in servlets?
**Answer:** Session management is achieved through the `HttpSession` interface. It allows storing and retrieving session-related data between client requests.
**Question 3:** What is the purpose of a servlet filter?
**Answer:** A servlet filter is used to preprocess and postprocess requests and responses globally in web application. It's commonly used for tasks like authentication, logging, or modifying request/response.
**Question 4:** How do servlet filters work?
**Answer:** Filters intercept requests and responses before they reach the servlet or after they leaven it. They can modify requests, responses, or perform operations based on certain conditions.
**Question 5:** What are servlet listeners?

\*\*Answer:\*\* Servlet listeners are interfaces that allow a servlet to receive notifications about events in

the web application, such as context initialization, session creation, and attribute changes.

**Question 6:** How do listeners differ from filters in servlets?
ψψΑ
**Answer:** Listeners respond to application-wide events, like initialization or session changes. Filters process requests and responses globally. Listeners are event-driven, while filters are request/response-centric.
**Question 7:** What is JDBC, and how is it used in servlets?
**Answer:** JDBC (Java Database Connectivity) is an API used to interact with databases. Servlets use JDBC to connect to databases, execute SQL queries, and perform data manipulation.
**Question 8:** How does a `ServletContextListener` differ from a `ServletRequestListener`?
**Answer:** `ServletContextListener` responds to application-level events like context initialization, while `ServletRequestListener` responds to request-specific events.
**Question 9:** How can you ensure efficient resource management in servlets?
**Answer:** Efficient resource management involves closing database connections, releasing memory, and cleaning up resources in the `destroy()` method of servlets or listeners.
**Question 10:** What is the purpose of using a global exception handler in servlets?
**Answer:** A global exception handler, set in the `web.xml` file, captures unhandled exceptions across the application. It allows developers to centralize exception handling and provide consistent error responses.