1. What is web application or client/server architecture?

Answer: A web application runs on a web server and is accessed through a web browser. Client/server architecture consists of clients that request services and servers that provide them, where the client is the front end and the server is the back end.

2. What is a Servlet?

Answer: A Servlet is a Java class that extends the capabilities of servers by enabling them to handle requests and responses in a web application.

3. How does PrintWriter work?

Answer: PrintWriter is a Java class used to send character data to the client. It has methods for writing text data and is used within servlets to output responses to the client.

4. What is servlet architecture?

Answer: Servlet architecture includes a web container that manages the servlet lifecycle, processes incoming requests, and creates responses.

5. What are lifecycle methods of a Servlet?

Answer: \`init()\`: Initializes the servlet.

\`service()\`: Handles requests.

\`destroy()\`: Cleans up before the servlet is taken out of service.

6. What is the difference between doGet and doPost()?

Answer: \`doGet()\`: Manages HTTP GET requests, generally used to retrieve data.

\`doPost()\`: Handles HTTP POST requests, typically used to send data to the server.

7. When does destroy get called?

Answer: The \`destroy()\` method is invoked when the servlet is being taken out of service, allowing it to release resources.

8. What are ways to implement a servlet?

Answer: By extending the \`HttpServlet\` class.

By implementing the `Servlet` interface.

9. What are HTTP methods?

Answer: Common HTTP methods include GET, POST, PUT, DELETE, HEAD, OPTIONS, and PATCH.

10. What HTTP method is used by Servlet?

Answer: Servlets handle various HTTP methods, mainly GET and POST.

11. What is the difference between GenericServlet and HttpServlet?

Answer: `GenericServlet` is protocol-agnostic, while `HttpServlet` is specifically designed for HTTP protocol.

12. What is the use of RequestDispatcher Interface?

Answer: It forwards a request from one servlet to another resource on the server, like another servlet or a JSP.

13. What is the difference between ServletConfig and ServletContext?

Answer: `ServletConfig`: Provides initialization parameters for a specific servlet.

`ServletContext`: Provides information and resources shared across the whole application.

14. What do you mean by InterServlet communication?

Answer: It refers to the interaction between servlets within the same application, typically using `RequestDispatcher` or shared resources.

15. What is the significance of web.xml?

Answer: `web.xml` is a configuration file for a web application, defining servlet mappings, initialization parameters, and other settings.

16. Explain Web Container?

Answer: A web container, or servlet container, manages the lifecycle of servlets, processes requests, and generates responses.

17. What do you mean by Servlet Chaining?

Answer: Servlet chaining involves forwarding a request from one servlet to another for additional processing.

18. Why do we use sendRedirect() method?

Answer: `sendRedirect()` is used to direct the client to a different URL, resulting in a new request.

19. What are Servlet filters?

Answer: Servlet filters are objects that modify requests or responses, often used for logging, authentication, or compression.

20. When to use Servlet filter?

Answer: Servlet filters are used for preprocessing or postprocessing requests, such as for logging, authentication, or validating input.

21. How do we translate JSP?

Answer: JSP is converted into a servlet by the web container before it is compiled and executed.

22. What is JSP?

Answer: JavaServer Pages (JSP) is a technology that allows the creation of dynamic web content using HTML combined with Java code.

23. Why do you use JSP?

Answer: JSP simplifies the creation of dynamic web content by embedding Java code within HTML.

24. What are implicit objects in JSP?

Answer: Implicit objects in JSP include `request`, `response`, `session`, `application`, `out`, `config`, `pageContext`, `page`, and `exception`.

25. What are scriptlets in JSP?

Answer: Scriptlets are blocks of Java code embedded in JSP pages using `<% %>` tags.

26. What are directives?

Answer: Directives in JSP provide page-level instructions, such as `page`, `include`, and `taglib`.

27. How to execute Java code in JSP?

Answer: Java code in JSP can be executed using scriptlets (`<% %>`), expressions (`<%= %>`), and declarations (`<%! %>`).

28. Why use JSP when Servlet can do everything JSP can?

Answer: JSP allows for better separation of presentation and business logic, making it easier to write and maintain HTML mixed with Java code.

29. What are lifecycle phases of JSP?

Answer: The lifecycle phases include translation, compilation, initialization, execution, and cleanup.

30. What are the methods used here?

Answer: In JSP, methods include `jspInit()`, `jspDestroy()`, and `_jspService()` to handle requests.