JSP ASSIGNMENT

1. What is JSP in Java?

2. Describe the lifecycle of a JSP page.

3. What are the differences between JSP and Servlets?

4. How do you create a simple JSP page?

5. Explain the use of JSP directives.

6. What is a JSP scriptlet?

7. How do you include static content in a JSP page?

8. What are JSP declarations?

9. What is the difference between JSP expressions and scriptlets?

10. How do you use JSP comments, and how are they different from HTML comments?

11. What are JSP implicit objects? Name a few of them.

12. Explain the purpose of JSP action tags.

13. What is the role of the jsp:include action tag?

14. How do you forward a request to another resource using JSP?

15. What are custom tags in JSP?

16. How do you create and use a custom tag in JSP?

17. What is the JavaBeans component model, and how is it used in JSP?

18. Explain the use of the jsp:setProperty and jsp:getProperty tags.

19. How do you handle exceptions in JSP?

20. What are expression language (EL) in JSP, and how are they used?

SERVLET ASSIGNMENT

1. What is a Servlet in Java?

2. Describe the lifecycle of a Servlet.

3. What is the difference between a Servlet and a JSP?

4. How do you configure a Servlet in a web application?

5. What is the web.xml file, and how is it used in configuring Servlets?

6. Explain the purpose of the HttpServlet class.

7. How do you handle GET and POST requests in a Servlet?

8. What is the ServletConfig interface?

9. What is the ServletContext interface, and how is it different from ServletConfig?

10. How do you initialize a Servlet?

11. What are the advantages of using Servlets over CGI?

12. Explain the concept of session management in Servlets.

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

14. How do you handle exceptions in Servlets?

15. Describe the process of file uploading using Servlets.

16. What are filters in Servlets, and how are they used?

17. How do you forward a request from one Servlet to another?

18. What is the role of RequestDispatcher?

19. Explain the difference between forwarding a request and redirecting a request in Servlets.

20. How do you maintain security in a web application using Servlets?

SOCKET PROGRAMMING ASSIGNMENT

1. What is socket programming?

2. Explain the difference between TCP and UDP protocols. How does it relate to socket programming?

3. In Java, what is the purpose of the Socket class?

4. What is a server socket in Java? How is it different from a regular socket?

5. Explain the role of the ServerSocket class in socket programming.

6. What is the purpose of the InputStream and OutputStream in socket programming?

7. How does a server in socket programming distinguish different clients?

8. What is the significance of the accept() method in the ServerSocket class?

9. Explain the terms "blocking" and "non-blocking" in the context of socket programming.

10. How do you handle exceptions in socket programming in Java? Provide examples of common exceptions.

11. What is the role of the PrintWriter class in socket programming, and how is it used?

12. Explain the concept of a port number in socket programming. Why is it necessary?

13. How can you ensure proper communication between a Java server and client if they are on different machines?

14. Discuss the steps involved in creating a simple client-server application in Java using sockets.

15. What is the purpose of the close() method in socket programming? Why is it important to close sockets properly?

16. Describe the differences between synchronous and asynchronous socket communication.

17. How can you handle multiple client connections in a Java server using socket programming?

18. Explain the purpose of the BufferedReader and BufferedWriter classes in socket programming.

19. Discuss the advantages and disadvantages of using sockets for communication in a distributed system.

20. What security considerations should be taken into account when implementing socket programming, and how can you address them?

SOCKET PROGRAMMING ASSIGNMENT

1. What is socket programming?

2. Explain the difference between TCP and UDP protocols. How does it relate to socket programming?

3. In Java, what is the purpose of the Socket class?

4. What is a server socket in Java? How is it different from a regular socket?

5. Explain the role of the ServerSocket class in socket programming.

6. What is the purpose of the InputStream and OutputStream in socket programming?

7. How does a server in socket programming distinguish different clients?

8. What is the significance of the accept() method in the ServerSocket class?

9. Explain the terms "blocking" and "non-blocking" in the context of socket programming.

10. How do you handle exceptions in socket programming in Java? Provide examples of common exceptions.

11. What is the role of the PrintWriter class in socket programming, and how is it used?

12. Explain the concept of a port number in socket programming. Why is it necessary?

13. How can you ensure proper communication between a Java server and client if they are on different machines?

14. Discuss the steps involved in creating a simple client-server application in Java using sockets.

15. What is the purpose of the close() method in socket programming? Why is it important to close sockets properly?

16. Describe the differences between synchronous and asynchronous socket communication.

17. How can you handle multiple client connections in a Java server using socket programming?

18. Explain the purpose of the BufferedReader and BufferedWriter classes in socket programming.

19. Discuss the advantages and disadvantages of using sockets for communication in a distributed system.

20. What security considerations should be taken into account when implementing socket programming, and how can you address them?