Assignment Questions 7

Q1.What is the use of JDBC in java?

JDBC or Java Database Connectivity is a Java API to connect and execute the query with the database.JDBC helps you to write Java applications that manage these three programming activities:

- Connect to a data source, like a database
- Send queries and update statements to the database

Q2.What are the steps involved in JDBC?

The fundamental steps involved in the process of connecting to a database and executing a query consist of the following:

- Import JDBC packages.
- Load and register the JDBC driver.
- Open a connection to the database.
- Create a statement object to perform a query.
- Execute the statement object and return a query resultset.
- Process the resultset.
- Close the resultset and statement objects.

Q3.What are the types of statement in JDBC in java?

There are three types of statements in JDBC namely:
1. Statement
2. Prepared Statement
3. Callable statement.

Q4.What is Servlet in Java?

A Servlet is a Java class that is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by web servers. Servlets are server-side components that provide a powerful mechanism for developing server-side programs. They are used to extend the functionality of web servers by providing support for dynamic response and data persistence.

Q5.Explain the life Cycle of servlet?

The life cycle of a servlet is controlled by the container in which the servlet has been deployed. When a request is mapped to a servlet, the container performs the following steps:

- **If an instance of the servlet does not exist, the web container loads the servlet class.**
- **The container creates an instance of the servlet class.**
- **The container calls the init method.**
- **The container calls the service method, which handles client requests.**
- **The container may remove the servlet instance by calling the destroy method[3][4][5].**

Q6.Explain the difference between the RequestDispatcher.forward() and HttpServletResponse.sendRedirect() methods?

The difference between the RequestDispatcher.forward() and HttpServletResponse.sendRedirect() methods is that the forward() method sends the request to another resource on the server, while the sendRedirect() method sends the response to a resource on a different server or context .

Q7.What is the purpose of the doGet() and doPost() methods in a servlet?

The purpose of the doGet() and doPost() methods in a servlet is to handle HTTP GET and POST requests respectively. The doGet() method is used to handle GET requests, while the doPost() method is used to handle POST requests. The doGet()

method is called by the web container to process an HTTP GET request, while the doPost() method is called to process an HTTP POST request .

Q8.Explain the JSP Model-View-Controller (MVC) architecture.

The JSP architecture is based on the Model-View-Controller (MVC) design pattern, separating the page's content from its presentation. This makes it possible to easily change the page's look without modifying the underlying code. The MVC architecture consists of three components:

- **Model:** The model represents the data and business logic of the application.
- **View:** The view represents the presentation layer of the application.
- **Controller:** The controller acts as an interface between the model and view components.

Q9.What are some of the advantages of Servlets?

Some of the advantages of Servlets are:

- Servlets are platform-independent and can run on any web server that supports the Java Servlet API.
- Servlets are efficient and can handle multiple requests concurrently.

- Servlets are easy to maintain and can be updated without restarting the web server.
- Servlets are secure and can be used to implement authentication and authorization mechanisms.
- Servlets can be used to generate dynamic content such as HTML, XML, or JSON .

Q10.What are the limitations of JSP?

Some of the limitations of JSP are:

- JSP pages can be difficult to debug because they mix HTML and Java code.
- JSP pages can become complex and difficult to maintain as the application grows.
- JSP pages can be slow to load because they must be compiled before they can be executed.
- JSP pages can be vulnerable to security attacks such as cross-site scripting (XSS) and SQL injection.