Assignment No. 05

AIM:

Implement the sample program demonstrating the use of Servlet.

e.g., Create a database table ebookshop (book_id, book_title, book_author, book_price, quantity) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using servlet.

Learnig Objectives:

1. Understand about basic concepts of html, CSS

2. Understand the basic functionalities of Servlets

Software Requirements:

1 Operating System: Windows 7/8/10/Ubuntu

2 Browser: Firefox/Google Chrome/ Microsoft Edge etc.

3 Software/Editor : Eclipse

4 Any Operating System

5 JDK 7or later

6 Tomcat 7 or later

Hardware Requirements:

1 Processor: Minimum 1 GHz.

2 Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

3 Hard Drive: Minimum 32 GB.

4 Memory (RAM): Minimum 1 GB

Theory:

Servlet:

A Servlet is a server side program and written in Java. Servlet is a web component that is deployed on the server for creating the dynamic web pages. A Java servlet is a Java program that extends the capabilities of a server. Although servlets can respond to any

types of requests, they most commonly execute applications hosted on Web servers.

Java Servlets are Java classes run by a web server that has an interpreter that supports the Java Servlet specification.

Servlets can be created using the packages

• javax.servlet

• javax.servlet.http

Servlets - Life Cycle:

A servlet life cycle can be defined as the entire process from its creation till the destruction. The following are the paths followed by a servlet.

- The servlet is initialized by calling the init() method.
- The servlet calls service() method to process a client's request.
- The servlet is terminated by calling the destroy() method.
- Finally, servlet is garbage collected by the garbage collector of the JVM.

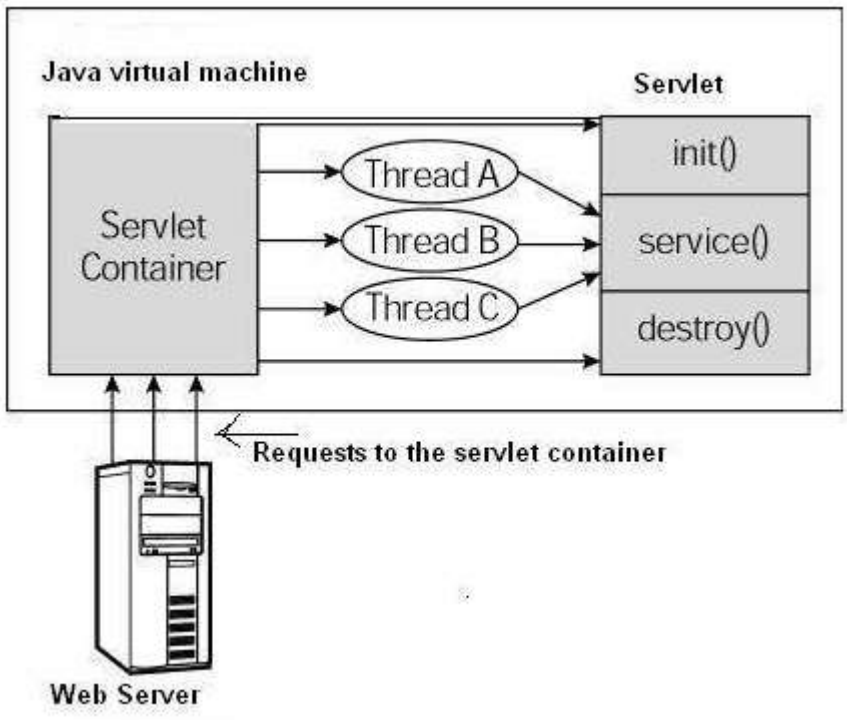

Fig. 01. Servlet Life Cycle

The init() Method:

The init method is called only once. It is called only when the servlet is created, and not called for any user requests afterwards. So, it is used for one-time initializations, just as with the init method of applets.

The servlet is normally created when a user first invokes a URL corresponding to the servlet, but you can also specify that the servlet be loaded when the server is first started.

Syntax:

```
public void init() throws ServletException {
// Initialization code...
}
```

The service() Method:

The service() method is the main method to perform the actual task. The servlet container (i.e. web server) calls the service() method to handle requests coming from the client(browsers) and to write the formatted response back to the client.

Each time the server receives a request for a servlet, the server spawns a new thread and calls service. The service() method checks the HTTP request type (GET, POST, PUT, DELETE, etc.) and calls doGet, doPost, doPut, doDelete, etc. methods as appropriate.

Syntax:

```
public void service(ServletRequest request, ServletResponse response)
 throws ServletException, IOException {
}
```

The doGet() Method:

A GET request results from a normal request for a URL or from an HTML form that has no METHOD specified and it should be handled by doGet() method.

Syntax:

```
public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
```

```
    // Servlet code
}
```

The doPost() Method:

A POST request results from an HTML form that specifically lists POST as the METHOD and it should be handled by doPost() method.

Syntax:

```
public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // Servlet code
}
```

The destroy() Method:

The destroy() method is called only once at the end of the life cycle of a servlet. This method gives your servlet a chance to close database connections, halt background threads, write cookie lists or hit counts to disk, and perform other such cleanup activities.

Syntax:

```
public void destroy() {
  // Finalization code...
}
```

TECHNOLOGY/TOOL:

How to configure tomcat server in Eclipse? (One time Requirement)

• If you are using Eclipse IDE first time, you need to configure the tomcat server First.

• For configuring the tomcat server in eclipse IDE,

• click on servers tab at the bottom side of the IDE -> right click on blank area -> New -> Servers -> choose tomcat then its version -> next -> click on Browse button -> select the apache tomcat root folder previous to bin -> next -> addAll -> Finish.

Steps to run servlet in Eclipse:

• Create a Dynamic web project

• create a servlet

• add servlet-api.jar file

• Run the servlet

Reading HTML Form Data using Servlet:

• getParameter() − You call request.getParameter() method to get the value of a form parameter.

• getParameterValues() − Call this method if the parameter appears more than once and returns multiple values, for example checkbox.

• getParameterNames() − Call this method if you want a complete list of all parameters in the current request.

DESIGN/EXECUTION STEPS:

Following steps are used to Create and Execute web applications,

1. Design html and servlet files with an extension of.html and .java

2. Start the Tomcat Server with port number

Conclusion:

Hence, we have performed the dynamic web application using Servlet and MySQL.

Assignment No. 06

AIM:

Implement the program demonstrating the use of JSP.

e.g., Create a database table students_info (stud_id, stud_name, class, division, city) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using JSP.

Learnig Objectives:

1. Understand about basic concepts of html, CSS
2. Understand the basic functionalities of JSP
3. Having the knowledge of SQL query to create the database

Software Requirements:

1 Operating System: Windows 7/8/10/Ubuntu

2 Browser: Firefox/Google Chrome/ Microsoft Edge etc.

3 Software/Editor : Eclipse

4 Any Operating System

5 JDK 7or later

6 Tomcat 7 or later

7 Java MySQL Connector

Hardware Requirements:

1 Processor: Minimum 1 GHz.

2 Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

3 Hard Drive: Minimum 32 GB.

4 Memory (RAM): Minimum 1 GB

Theory:

Java Server Pages (JSP): It is a server side programming technology that is used to create dynamic web-based applications. JSP have right to use the complete Java APIs, including

the JDBC API to access the databases. It is a technology that helps software developers to create dynamic web pages based on HTML, XML and other document types. It was released in 1999 by Sun Microsystems. It is just like a PHP and ASP, but it uses the Java programming language.

A JSP element is a type of java servlet that is designed to accomplish the role of a user interface for a java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and rooted JSP actions and commands. Using JSP, you can collect input from users through webpage forms, current records from a database or another source and create web pages dynamically.

JSP tags can be used for different purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages, and sharing information between requests, pages etc.

Why we need JSP?

JSP is used for the design of dynamic web page and servlet is used to code the logic that is present i.e. in the MVC (Model-ViewController) architecture, the servlet is the controller and the JSP is the view.

Advantage of JSP over Servlet:

1) Extension to Servlet

2) Easy to maintain

3) Fast Development: No need to recompile and redeploy

4) Less code than Servlet

Architecture of JSP:

1. The request / response part of a JSP is defined in below architecture

2. The client initiated request for a JSP file using browser

3. Webs server (i.e, JSP Engine) invokes the JSP file and interpret the JSP file produce a java code. The created java code will be a Servlet.

4. Once Servlet is created, JSP engine compiles the servlet. Compilation errors will be detected in this phase.

5. Now servlet class is loaded by the container and executes it.

6. Engine sends the response back to the client.

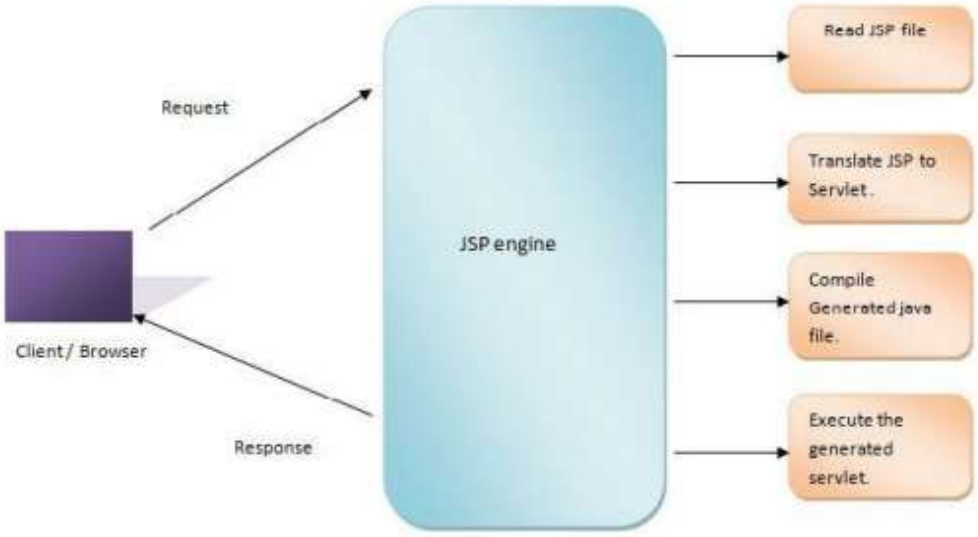

Figure. Architecture of JSP

Syntax of JSP:

JSP declaration is used to declare variables and methods as shown below,

Following is the simple and first example for JSP:

//Hello.jsp

<html>

<% out.println("Welcome to JSP Class"); %>

</html>

Output:

Welcome to JSP Class

Elements of JSP:

Scripting Element	Example
Comment	<%-- comment --%>
Directive	<%@ directive %>
Declaration	<%! declarations %>
Scriptlet	<% scriplets %>
Expression	<%= expression %>

]

JDBC:

Java JDBC is a java API to connect and execute query with the database. JDBC API uses jdbc drivers to connect with the database. JDBC Driver is a software component that enables java application to interact with the database.

There are 4 types of JDBC drivers:

1. JDBC-ODBC bridge driver

2. Native-API driver (partially java driver)

3. Network Protocol driver (fully java driver)

4. Thin driver (fully java driver)

Technology:

1. JSP

2. IDE: Eclipse

3. Databases: MySQL

MySQL: MySQL is a freely available open source Relational Database Management System (RDBMS). It uses the Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing data in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

Conclusion:

Hence, we have performed the dynamic web application using JSP and MySQL.