

04. Java Servlet

4.1 What is servlet?

- A Servlet is a Java class that handles HTTP requests and responses on a web server.

4.2 Simple Servlet - Display Static Message

```
import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

@WebServlet("/displayMessage")
public class DisplayMessageServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h1>Welcome to the Java Servlet Lab!</h1>");
        out.println("</body></html>");
    }
}
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE glassfish-web-app PUBLIC "-//GlassFish.org//DTD GlassFish Application Server
3.1 Servlet 3.0//EN"
"http://glassfish.org/dtds/glassfish-web-app_3_0-1.dtd">
<glassfish-web-app error-url="">
<web-app xmlns="http://java.sun.com/xml/ns/javaee" version="3.0">
<servlet>
<servlet-name>DisplayMessageServlet</servlet-name>
<servlet-class>com.example.DisplayMessageServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>DisplayMessageServlet</servlet-name>
<url-pattern>/displayMessage</url-pattern>
</servlet-mapping>
</web-app>
</glassfish-web-app>
```



4.3 Get User Input from Form and Display

- **Index.html**

```
<!DOCTYPE html>
<html>
<head>
<title>Input Form</title>
<style>
body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #121212;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
    color: #e0e0e0;
}

form {
    background-color: #1e1e1e;
    padding: 25px 35px;
    border-radius: 12px;
    box-shadow: 0 6px 12px rgba(0,0,0,0.3);
    width: 300px;
    border: 1px solid #333;
}
```

```
input[type="text"] {  
    padding: 12px;  
    width: 100%;  
    margin: 8px 0 20px 0;  
    border: 1px solid #444;  
    border-radius: 6px;  
    font-size: 16px;  
    background-color: #2d2d2d;  
    color: #ffffff;  
    transition: border 0.3s;  
}  
  
input[type="text"]:focus {  
    border-color: #9c27b0;  
    outline: none;  
}  
  
input[type="submit"] {  
    background-color: #7b1fa2;  
    color: white;  
    padding: 12px 24px;  
    border: none;  
    border-radius: 6px;  
    font-size: 16px;  
    cursor: pointer;  
    width: 100%;  
    transition: background-color 0.3s;  
}  
  
input[type="submit"]:hover {  
    background-color: #9c27b0;  
}  
  
label {  
    font-weight: 600;  
    color: #bb86fc;  
    font-size: 14px;  
    text-transform: uppercase;  
    letter-spacing: 0.5px;  
}
```

```

h1 {
    color: #bb86fc;
    text-align: center;
    margin-bottom: 25px;
}
</style>
</head>
<body>
<form action="getUserInput" method="POST">
<h1>User Login</h1>
<label for="username">Your Name:</label>
<input type="text" name="username" id="username" required placeholder="Enter your name">
<input type="submit" value="Submit">
</form>
</body>
</html>

```

- **Servlet Code (GetUserInputServlet.java):**

```

import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
/**
 *
 * @author USER
 */
@WebServlet(urlPatterns = {"getUserInput"})
public class GetUserInputServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String username = request.getParameter("username");
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h1>Hello, " + username + "</h1>");
        out.println("</body></html>");    }
}

```

A screenshot of a web browser window. The address bar shows the URL `localhost:8080/Servlet2/`. Below the address bar is a form with a single input field labeled "Name:" containing the text "hansi". Below the input field is a "Submit" button.

A screenshot of a web browser window. The address bar shows the URL `localhost:8080/Servlet2/getUserInput`. The main content area of the browser displays the text "Hello, hansi!" in a large, bold, black font.

4.4 Get Multiple Inputs, Perform Calculation, and Display Result

- **Calculate.html**

```
<!DOCTYPE html>
<html>
<head>
<title>Sum Calculator</title>
<style>
body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f5f7fa;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
}
form {
    background-color: white;
    padding: 30px 40px;
    border-radius: 12px;
    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.08);
    width: 320px;
}
h2 {
    text-align: center;
    margin-bottom: 25px;
    color: #4a6cf7;
    font-size: 24px;
}
label {
    display: block;
    margin-bottom: 8px;
    color: #555;
    font-weight: 500;
}
input[type="number"] {
    width: 100%;
    padding: 12px 15px;
    margin-bottom: 20px;
    border: 1px solid #e0e0e0;
    border-radius: 8px;
    font-size: 16px;
}
```

```
        transition: border 0.3s;
    }
    input[type="number"]:focus {
        border-color: #4a6cf7;
        outline: none;
    }
    input[type="submit"] {
        width: 100%;
        padding: 12px;
        background-color: #4a6cf7;
        color: white;
        font-size: 16px;
        font-weight: 500;
        border: none;
        border-radius: 8px;
        cursor: pointer;
        transition: background-color 0.3s;
    }
    input[type="submit"]:hover {
        background-color: #3a5bd9;
    }
</style>
</head>
<body>
<form action="calculateSum" method="POST">
<h2>Sum Calculator</h2>

<label for="num1">First Number:</label>
<input type="number" name="num1" id="num1" required>

<label for="num2">Second Number:</label>
<input type="number" name="num2" id="num2" required>

<input type="submit" value="Calculate Sum">
</form>
</body>
</html>
```

- **Servlet Code (CalculateSumServlet.java):**

```

import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @author USER
 */
@WebServlet(urlPatterns = {" /calculateSum"})
public class CalculateSumServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Retrieve and parse the input numbers from the request
        int num1 = Integer.parseInt(request.getParameter("num1"));
        int num2 = Integer.parseInt(request.getParameter("num2"));

        // Calculate the sum
        int sum = num1 + num2;

        // Set response content type
        response.setContentType("text/html");

        // Generate HTML response
        PrintWriter out = response.getWriter();
        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Sum Result</title>");
        out.println("<style>");
        out.println("body { font-family: Arial, sans-serif; background-color: #f0f8ff; text-align: center; padding-top: 50px; }");
        out.println("h1 { color: #333; }");
        out.println("</style>");
        out.println("</head>");
        out.println("<body>");
```

```
out.println("<h1>The sum of " + num1 + " and " + num2 + " is: <strong>" + sum +  
"</strong></h1>");  
out.println("</body>");  
out.println("</html>");  
}  
}
```

Sum Calculator

First Number:

Second Number:

Calculate Sum

The sum of 10 and 5 is: 15

localhost:15786/Servlet02/calculateSum

4.5 Java Servlet with Database CRUD Operations

- **Database Setup**

```
CREATE DATABASE stock_management;
USE stock_management;

CREATE TABLE stock (
id INT AUTO_INCREMENT PRIMARY KEY,
product_name VARCHAR(255),
quantity INT
);
```

- **stockForm.html**

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Stock Management</title>
<!-- Bootstrap CSS CDN -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
<style>
.card-header {
background-color: #6f42c1; /* Purple header */
}
.btn-add {
background-color: #20c997; /* Teal add button */
border-color: #20c997;
}
.btn-update {
background-color: #fd7e14; /* Orange update button */
border-color: #fd7e14;
}
.btn-delete {
background-color: #dc3545; /* Red delete button */
border-color: #dc3545;
}
body {
background-color: #f8f9fa;
}
.card {
border: none;
border-radius: 10px;
```

```

        }
    </style>
</head>
<body>

    <div class="container mt-5">
        <div class="card shadow">
            <div class="card-header text-white">
                <h2 class="mb-0">Manage Stock</h2>
            </div>
            <div class="card-body">
                <form action="stockAction" method="POST">
                    <div class="mb-3">
                        <label for="product_name" class="form-label">Product Name:</label>
                        <input type="text" class="form-control" id="product_name" name="product_name" required>
                    </div>

                    <div class="mb-3">
                        <label for="quantity" class="form-label">Quantity:</label>
                        <input type="number" class="form-control" id="quantity" name="quantity" required>
                    </div>

                    <div class="d-flex gap-2">
                        <button type="submit" name="action" value="Add Product" class="btn btn-add text-white">Add Product</button>
                        <button type="submit" name="action" value="Update Product" class="btn btn-update text-white">Update Product</button>
                        <button type="submit" name="action" value="Delete Product" class="btn btn-delete text-white">Delete Product</button>
                    </div>
                </form>
            </div>
        </div>
    </div>

    <!-- Bootstrap JS (Optional) -->
    <script
        src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- **Servlet Code (StockManagementServlet.java):**

```

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

@WebServlet(urlPatterns = {"stockAction"})
public class StockManagementServlet extends HttpServlet {

    private Connection getConnection() throws SQLException {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver"); // Load MySQL driver
        } catch (ClassNotFoundException e) {
            throw new SQLException("MySQL JDBC Driver not found.", e);
        }

        String url =
            "jdbc:mysql://localhost:3306/stock_management?useSSL=false&allowPublicKeyRe
trieval=true&serverTimezone=UTC";
        String username = "root";
        String password = "root"; // Update if you have a MySQL password

        return DriverManager.getConnection(url, username, password);
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String action = request.getParameter("action");
        String productName = request.getParameter("product_name");
        int quantity = Integer.parseInt(request.getParameter("quantity"));

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
    }
}

```

```

try (Connection conn = getConnection()) {
    switch (action) {
        case "Add Product":
            try (PreparedStatement stmt = conn.prepareStatement(
                "INSERT INTO stock (product_name, quantity) VALUES (?, ?)")) {
                stmt.setString(1, productName);
                stmt.setInt(2, quantity);
                stmt.executeUpdate();
                out.write("<h1>✓ Product Added Successfully</h1>");
            }
            break;

        case "Update Product":
            try (PreparedStatement stmt = conn.prepareStatement(
                "UPDATE stock SET quantity = ? WHERE product_name = ?")) {
                stmt.setInt(1, quantity);
                stmt.setString(2, productName);
                stmt.executeUpdate();
                out.write("<h1>🕒 Product Updated Successfully</h1>");
            }
            break;

        case "Delete Product":
            try (PreparedStatement stmt = conn.prepareStatement(
                "DELETE FROM stock WHERE product_name = ?")) {
                stmt.setString(1, productName);
                stmt.executeUpdate();
                out.write("<h1>✗ Product Deleted Successfully</h1>");
            }
            break;
        default:
            out.write("<h1>⚠ Invalid Action</h1>");
    }
} catch (SQLException e) {
    e.printStackTrace();
    out.write("<h1>✗ Database Error: " + e.getMessage() + "</h1>");
}
}

```

Add:

The screenshot shows a web browser window with three tabs at the top: "Eclipse GlassFish 7.0.0 - Error repo", "localhost:15786/Servlet02/getUser", and "localhost:15786/Servlet02/stockAction". The active tab displays a form titled "Manage Stock" with fields for "Product Name" (containing "car") and "Quantity" (containing "20"). Below the form are buttons for "Add Product" (blue), "Update Product" (green), and "Delete Product" (red). A success message "? Product Added Successfully" is displayed below the form. The main content area shows a "Result Grid" table with columns "id", "product_name", and "quantity". It contains two rows: one for Apple (id 1, quantity 10) and one for Banana (id 2, quantity 10). There is also a row with NULL values.

	id	product_name	quantity
▶	1	Apple	10
◀ *	2	Banana	10
NULL	NULL	NULL	

Update:

The screenshot shows a web browser window with the URL "localhost:15786/Servlet02/stockAction". A success message "? Product Updated Successfully" is displayed prominently. Below it is a "Result Grid" table with columns "id", "product_name", and "quantity". It contains one row: one for car (id 4, quantity 20). There is also a row with NULL values.

	id	product_name	quantity
▶*	4	car	20
NULL	NULL	NULL	

Delete:

The screenshot shows a web browser window with the URL `localhost:15786/Servlet02/stockAction`. The main content is a large bold text: **? Product Deleted Successfully**. Below this, there is a "Result Grid" table with the following data:

	id	product_name	quantity
4	car	20	NULL
NULL	NULL	NULL	NULL

4.6 Display Data from Database on Another Web Page

```
@WebServlet(urlPatterns = {"/DisplayProducts"})
public class DisplayProductsServlet extends HttpServlet {
    private Connection getConnection() throws SQLException {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver"); // Load MySQL JDBC driver
        } catch (ClassNotFoundException e) {
            throw new SQLException("MySQL JDBC Driver not found.", e);
        }
        String url =
            "jdbc:mysql://localhost:3306/stock_management?useSSL=false&allowPublicKeyRetrieval=true&serverTimezone=UTC";
        String username = "root";
        String password = "root"; // Use your MySQL password if set
        return DriverManager.getConnection(url, username, password);
    }
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
    }
}
```

```
try (Connection conn = getConnection()) {
    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM stock");

    out.println("<html><head><title>Stock List</title></head><body>");
    out.println("<h1>📦 Stock List</h1>");
    out.println("<ul>");

    while (rs.next()) {
        out.println("<li>" + rs.getString("product_name") + ": " + rs.getInt("quantity") + "</li>");
    }
    out.println("</ul>");
    out.println("</body></html>");
} catch (SQLException e) {
    e.printStackTrace();
    out.println("<h1>✖ Database Error: " + e.getMessage() + "</h1>");
} } }
```

? Stock List

- car: 20