

## LAB-6

# Server Side Scripting using Servlet

18MIS7250

Amit Kumar Sahu

### 1. CODE:

#### Index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Product Page</title>
</head>
<body>

<h1>PRODUCTS</h1>
<form method="POST" action="ProductServlet">
<table border="1" style="margin-left:auto;margin-right:auto ">
<tr>
<td>Product ID</td>
<td>
<input type="text" name="pid"/></td>
</tr>
<tr>
<td>
Product Name</td>
<td>
<input type="text" name="pname">
</td>
</tr>
<tr>
<td>
Product Cost</td>
<td>
<input type="text" name="pcost">
</td>
</tr>
<tr>
<td>
Product Category</td>
<td>
<select name="post">
<option>Vegetables</option>
<option>Fruits</option>
<option>Juices</option>
<option>Sweets</option>
<option>Misc</option>
</select>
```

```

</td>
</tr>
<tr>
<td colspan="2">
<input type="submit" value="Submit">
</td>
</tr>

```

```

</table></form>
</body>
</html>

```

OUTPUT:

## PRODUCTS

Product ID	<input type="text"/>
Product Name	<input type="text"/>
Product Cost	<input type="text"/>
Product Category	Vegetables ▾
<input type="submit" value="Submit"/>	

## PRODUCTS

Product ID	P005
Product Name	MANGOES
Product Cost	90
Product Category	Fruits ▾
<input type="submit" value="Submit"/>	

ProductServlet.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 javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class ProdServ
 */
@WebServlet("/ProdServ")
public class ProdServ extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ProdServ() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
    response)
     */
    //protected void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        protected void processRequest(HttpServletRequest request,
    HttpServletResponse response) throws ServletException, IOException
    ,ClassNotFoundException,SQLException{
            response.setContentType("text/html;charset=UTF-8");
            try(PrintWriter out=response.getWriter()){
                String hostName="localhost";
                String dbName="product";
                String userName="root";

                Class.forName("com.mysql.jdbc.Driver");
                String password="";
                String id=request.getParameter("pid");
                String name=request.getParameter("pname");
                Float
    cost=Float.parseFloat(request.getParameter("pcost"));
            }
        }
    }
}
```

```

        String cat=request.getParameter("pcat");
        String connectionURL="jdbc:mysql://" + hostName + ":3306/"
+dbName;

        Connection conn =
DriverManager.getConnection(connectionURL, userName,
        password);

        String sql = "Insert into Prod1 values (?,?,,?)";

        PreparedStatement pstmt = conn.prepareStatement(sql);

        pstmt.setString(1, id);
        pstmt.setString(2, name);
        pstmt.setFloat(3, cost);

        pstmt.setString(4, cat);
        pstmt.executeUpdate();

        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Servlet UserServlet</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h2 style='text-align:center;'>Product added
Successfully</h2>");

        out.println("</body>");
        out.println("</html>");

    }
    catch(Exception p) {
        System.out.println(p);
    }

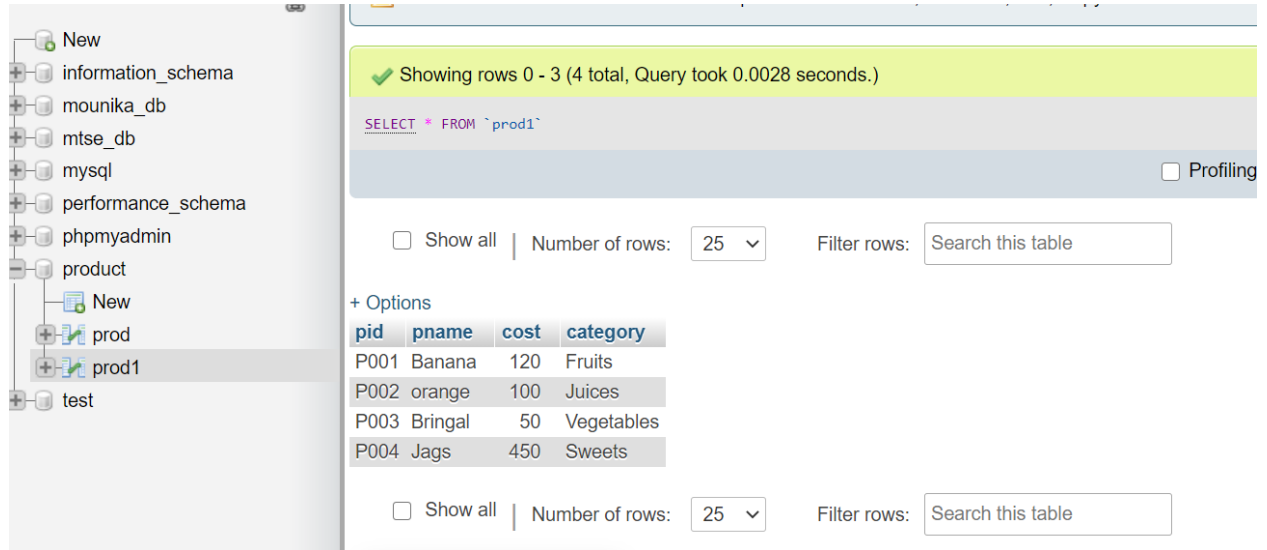
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

}

```

## OUTPUT:



The screenshot shows a database management interface. On the left is a tree view of the database structure, including 'information\_schema', 'mounika\_db', 'mtse\_db', 'mysql', 'performance\_schema', 'phpmyadmin', 'product', and 'test'. The 'product' database is expanded, showing 'prod' and 'prod1' tables. The 'prod1' table is selected. The main area displays a query result for the 'prod1' table. A green status bar at the top indicates 'Showing rows 0 - 3 (4 total, Query took 0.0028 seconds.)'. Below this, the query 'SELECT \* FROM `prod1`' is shown. A 'Profiling' checkbox is on the right. Below the query, there are controls for 'Show all', 'Number of rows' (set to 25), and 'Filter rows' (a search box). The query result is displayed in a table with 4 rows and 4 columns: 'pid', 'pname', 'cost', and 'category'. The data rows are: P001 Banana 120 Fruits, P002 orange 100 Juices, P003 Bringal 50 Vegetables, and P004 Jags 450 Sweets. Below the table, there are additional controls for 'Show all', 'Number of rows' (set to 25), and 'Filter rows' (a search box).

Showing rows 0 - 3 (4 total, Query took 0.0028 seconds.)

SELECT \* FROM `prod1`

☐ Profiling

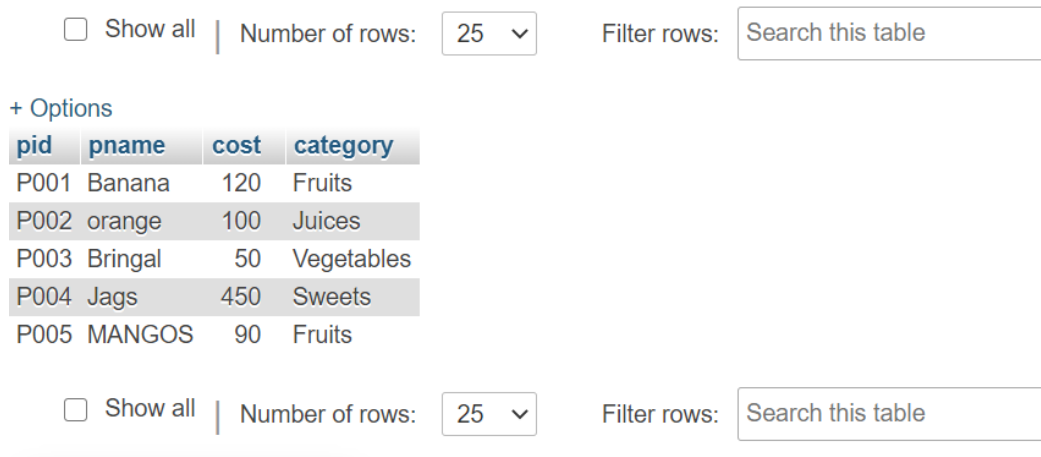
☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

pid	pname	cost	category
P001	Banana	120	Fruits
P002	orange	100	Juices
P003	Bringal	50	Vegetables
P004	Jags	450	Sweets

☐ Show all | Number of rows: 25 | Filter rows: Search this table

## After inserting values



The screenshot shows the same database management interface as before, but now the query result for the 'prod1' table includes an additional row. The status bar at the top indicates 'Showing rows 0 - 4 (5 total, Query took 0.0028 seconds.)'. The query 'SELECT \* FROM `prod1`' is still shown. The 'Profiling' checkbox is on the right. Below the query, there are controls for 'Show all', 'Number of rows' (set to 25), and 'Filter rows' (a search box). The query result is displayed in a table with 5 rows and 4 columns: 'pid', 'pname', 'cost', and 'category'. The data rows are: P001 Banana 120 Fruits, P002 orange 100 Juices, P003 Bringal 50 Vegetables, P004 Jags 450 Sweets, and P005 MANGOS 90 Fruits. Below the table, there are additional controls for 'Show all', 'Number of rows' (set to 25), and 'Filter rows' (a search box).

Showing rows 0 - 4 (5 total, Query took 0.0028 seconds.)

SELECT \* FROM `prod1`

☐ Profiling

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

pid	pname	cost	category
P001	Banana	120	Fruits
P002	orange	100	Juices
P003	Bringal	50	Vegetables
P004	Jags	450	Sweets
P005	MANGOS	90	Fruits

☐ Show all | Number of rows: 25 | Filter rows: Search this table