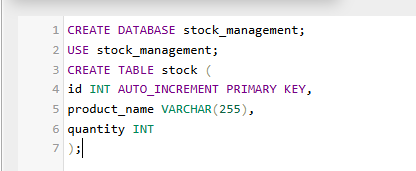
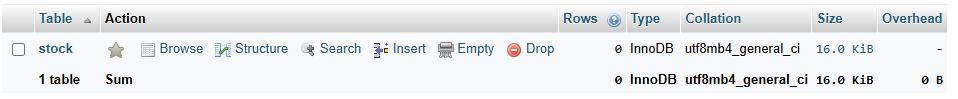
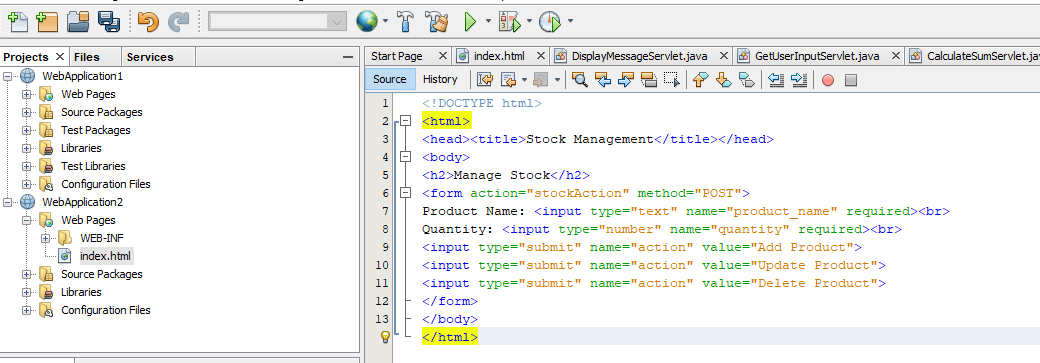
**Lab Task 4: Java Servlet with Database CRUD Operations  
  
Database Setup :**



**HTML Form (stockForm.html)**

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

package com.example;

import java.io.\*;

import java.sql.\*;

import javax.servlet.\*;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

@WebServlet("/stockAction")

public class StockManagementServlet extends HttpServlet {

// Initialize driver when servlet loads

@Override

public void init() throws ServletException {

try {

Class.forName("com.mysql.cj.jdbc.Driver");

} catch (ClassNotFoundException e) {

throw new ServletException("MySQL JDBC Driver not found", e);

}

}

private Connection getConnection() throws SQLException {

String url = "jdbc:mysql://localhost:3306/stock\_management?useSSL=false&serverTimezone=UTC";

String username = "root";

String password = "316830059";

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

}

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String action = request.getParameter("action");

String productName = request.getParameter("product\_name");

int quantity = Integer.parseInt(request.getParameter("quantity"));

try (Connection conn = getConnection()) {

switch (action) {

case "Add Product":

addProduct(conn, productName, quantity, out);

break;

case "Update Product":

updateProduct(conn, productName, quantity, out);

break;

case "Delete Product":

deleteProduct(conn, productName, out);

break;

default:

out.println("<h1>Invalid Action</h1>");

}

} catch (SQLException e) {

out.println("<h1>Database Error: " + e.getMessage() + "</h1>");

e.printStackTrace();

}

out.println("<br><a href='stockForm.html'>Back to Form</a>");

}

private void addProduct(Connection conn, String name, int quantity, PrintWriter out)

throws SQLException {

String sql = "INSERT INTO stock (product\_name, quantity) VALUES (?, ?)";

try (PreparedStatementstmt = conn.prepareStatement(sql)) {

stmt.setString(1, name);

stmt.setInt(2, quantity);

stmt.executeUpdate();

out.println("<h1>Product Added Successfully</h1>");

}

}

private void updateProduct(Connection conn, String name, int quantity, PrintWriter out)

throws SQLException {

String sql = "UPDATE stock SET quantity = ? WHERE product\_name= ?";

try (PreparedStatementstmt = conn.prepareStatement(sql)) {

stmt.setInt(1, quantity);

stmt.setString(2, name);

int rows = stmt.executeUpdate();

if (rows > 0) {

out.println("<h1>Product Updated Successfully</h1>");

} else {

out.println("<h1>Product Not Found</h1>");

}

}

}

private void deleteProduct(Connection conn, String name, PrintWriter out)

throws SQLException {

String sql = "DELETE FROM stock WHERE product\_name= ?";

try (PreparedStatementstmt = conn.prepareStatement(sql)) {

stmt.setString(1, name);

int rows = stmt.executeUpdate();

if (rows > 0) {

out.println("<h1>Product Deleted Successfully</h1>");

} else {

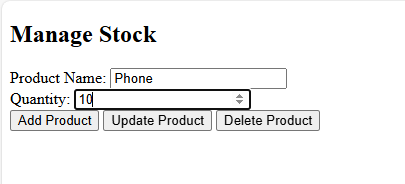
out.println("<h1>Product Not Found</h1>");

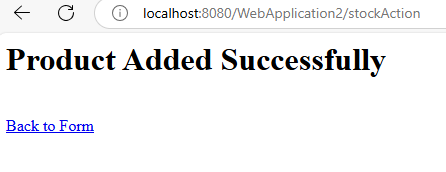
}

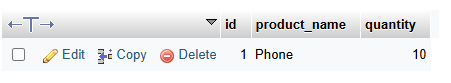
}

} }

Output







**Lab Task 5: Display Data from Database on Another Web Page**  
  
**Servlet Code (DisplayProductsServlet.java):**  
  
package com.example;

import java.io.\*;

import java.sql.\*;

import javax.servlet.\*;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

@WebServlet("/displayProducts")

public class DisplayProductsServlet extends HttpServlet {

// Reuse your existing connection method

private Connection getConnection() throws SQLException {

String url = "jdbc:mysql://localhost:3306/stock\_management?useSSL=false&serverTimezone=UTC";

String username = "root";

String password = "316830059";

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

}

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Stock List</title>");

out.println("<style>");

out.println("table { border-collapse: collapse; width: 50%; margin: 20px auto; }");

out.println("th, td { border: 1px solid #ddd; padding: 8px; text-align: left; }");

out.println("th { background-color: #f2f2f2; }");

out.println("</style>");

out.println("</head>");

out.println("<body>");

out.println("<h1 style='text-align: center;'>Current Stock List</h1>");

try (Connection conn = getConnection();

Statement stmt = conn.createStatement();

ResultSetrs = stmt.executeQuery("SELECT \* FROM stock")) {

out.println("<table>");

out.println("<tr><th>ID</th><th>Product Name</th><th>Quantity</th></tr>");

while (rs.next()) {

out.println("<tr>");

out.println("<td>" + rs.getInt("id") + "</td>");

out.println("<td>" + rs.getString("product\_name") + "</td>");

out.println("<td>" + rs.getInt("quantity") + "</td>");

out.println("</tr>");

}

out.println("</table>");

} catch (SQLException e) {

out.println("<h2 style='color: red; text-align: center;'>Error retrieving stock: "

+ e.getMessage() + "</h2>");

e.printStackTrace(); }

out.println("<div style='text-align: center; margin-top: 20px;'>");

out.println("<a href='stockForm.html'>Back to Stock Management</a>");

out.println("</div>");

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

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

}

}

**Updated stockForm.html**

<!DOCTYPE html>

<html>

<head><title>Stock Management</title>

<style> body { font-family: Arial, sans-serif; margin: 20px; }

form { max-width: 500px; margin: 0 auto; padding: 20px; border: 1px solid #ddd; border-radius: 5px; }

input[type="text"], input[type="number"] { width: 100%; padding: 8px; margin: 5px 0 15px; }

input[type="submit"] { padding: 8px 15px; margin-right: 10px; }

.view-link { display: block; text-align: center; margin-top: 20px; }

</style>

</head>

<body><h2 style="text-align: center;"> Manage Stock </h2>

<form action="stockAction" method="POST">

Product Name: <input type="text" name="product\_name" required><br>

Quantity: <input type="number" name="quantity" required><br>

<input type="submit" name="action" value="Add Product">

<input type="submit" name="action" value="Update Product">

<input type="submit" name="action" value="Delete Product">

</form>

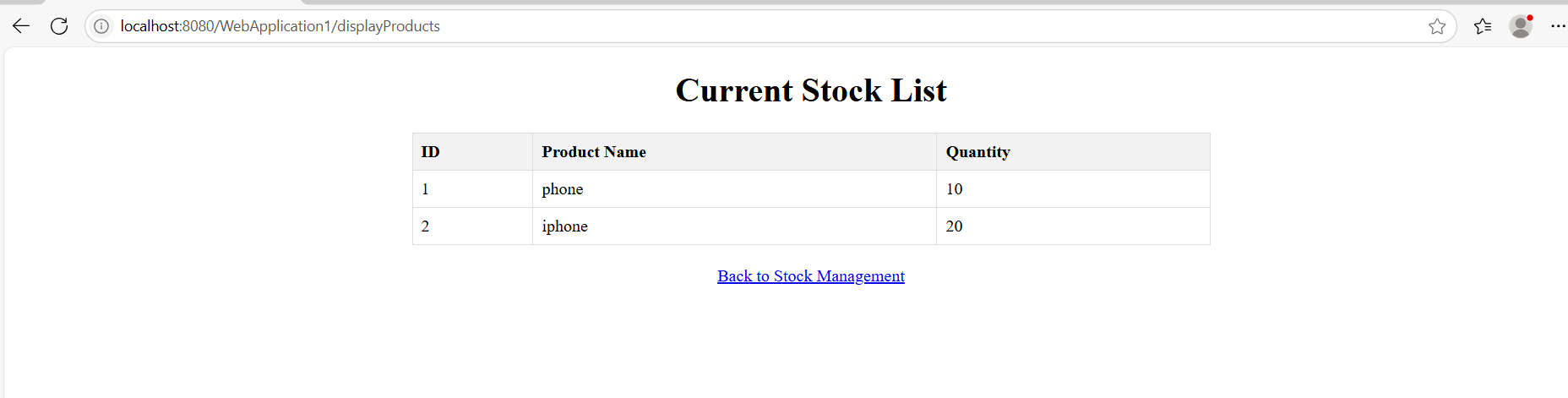
<div class="view-link">

<a href="displayProducts"> View All Products </a>

</div>

</body>

</html>

Output

Database

