**Demonstrate how to create ,select and drop a database in JDBC**

**DBConnection.java**

import java.sql.\*;

public class DBConnection {

private static final String *DRIVER\_CLASS* = "com.mysql.jdbc.Driver";

private static final String *DB\_URL* = "jdbc:mysql://localhost:3306/Products";

private static final String *USERNAME* = "root";

private static final String *PASSWORD* = "root";

public static Connection getConnection() throws ClassNotFoundException, SQLException {

Class.*forName*(*DRIVER\_CLASS*);

Connection connection = DriverManager.*getConnection*(*DB\_URL*, *USERNAME*, *PASSWORD*);

return connection;

}

public static void closeConnection(Connection connection) {

try {

if (connection != null) {

connection.close();

}

} catch (SQLException e) {

e.printStackTrace();

}

}

}

**DBoperation.java**

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class DBoperation extends HttpServlet {

private static final long *serialVersionUID* = 1L;

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

processRequest(request, response);

}

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

processRequest(request, response);

}

protected void processRequest(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

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

try {

Connection connection = DBConnection.*getConnection*();

Statement statement = connection.createStatement();

if (action.equals("create")) {

// Check if the database exists

ResultSet resultSet = connection.getMetaData().getCatalogs();

boolean databaseExists = false;

while (resultSet.next()) {

String databaseName = resultSet.getString(1);

if (databaseName.equals("ecommerce")) {

databaseExists = true;

break;

}

}

resultSet.close();

if (!databaseExists) {

// Create the database

statement.executeUpdate("CREATE DATABASE ecommerce");

out.println("<h3>Database created successfully.</h3>");

} else {

out.println("<h3>Database already exists.</h3>");

}

} else if (action.equals("use")) {

// Check if the database exists

ResultSet resultSet = connection.getMetaData().getCatalogs();

boolean databaseExists = false;

while (resultSet.next()) {

String databaseName = resultSet.getString(1);

if (databaseName.equals("ecommerce")) {

databaseExists = true;

break;

}

}

resultSet.close();

if (databaseExists) {

// Use the database

statement.executeUpdate("USE ecommerce");

out.println("<h3>Using database 'ecommerce'.</h3>");

} else {

out.println("<h3>Database does not exist. Cannot use the database.</h3>");

}

} else if (action.equals("drop")) {

// Check if the database exists

ResultSet resultSet = connection.getMetaData().getCatalogs();

boolean databaseExists = false;

while (resultSet.next()) {

String databaseName = resultSet.getString(1);

if (databaseName.equals("ecommerce")) {

databaseExists = true;

break;

}

}

resultSet.close();

if (databaseExists) {

// Drop the database

statement.executeUpdate("DROP DATABASE ecommerce");

out.println("<h3>Database dropped successfully.</h3>");

} else {

out.println("<h3>Database does not exist. Cannot drop the database.</h3>");

}

}

DBConnection.*closeConnection*(connection);

} catch (ClassNotFoundException | SQLException e) {

e.printStackTrace();

out.println("<h3>Error performing database operation.</h3>");

}

}

}

**index .html**

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<h1>Create, Use, and Drop Database</h1>

<form action=*"DBoperation"* method=*"post"*><br>

<input type=*"hidden"* name=*"action"* value=*"create"*>

<input type=*"submit"* value=*"Create Database"*><br>

</form>

<form action=*"DBoperation"* method=*"post"*><br>

<input type=*"hidden"* name=*"action"* value=*"use"*>

<input type=*"submit"* value=*"Use Database"*><br>

</form>

<form action=*"DBoperation"* method=*"post"*><br>

<input type=*"hidden"* name=*"action"* value=*"drop"*>

<input type=*"submit"* value=*"Drop Database"*><br>

</form>

</body>

</html>

**web.xml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"* id=*"WebApp\_ID"* version=*"2.5"*>

<display-name>DatabaseOperation1</display-name>

<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.jsp</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>default.html</welcome-file>

<welcome-file>default.jsp</welcome-file>

<welcome-file>default.htm</welcome-file>

</welcome-file-list>

<servlet>

<description></description>

<display-name>DBoperation</display-name>

<servlet-name>DBoperation</servlet-name>

<servlet-class>DBoperation</servlet-class>

</servlet>

<servlet-mapping>

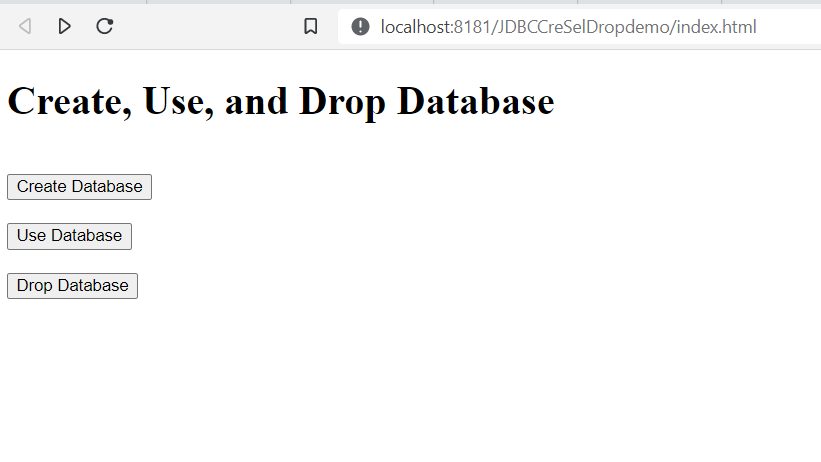
<servlet-name>DBoperation</servlet-name>

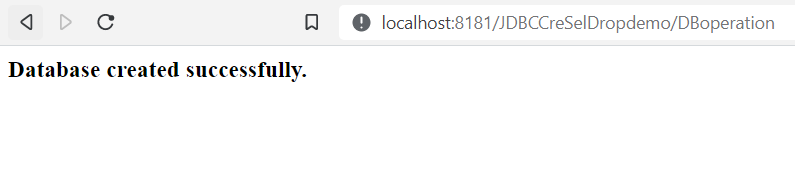
<url-pattern>/DBoperation</url-pattern>

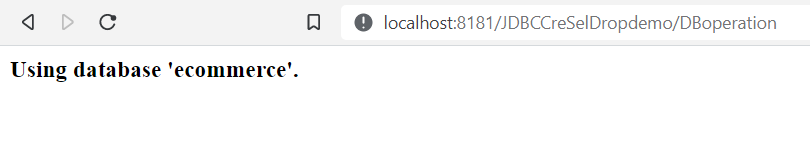
</servlet-mapping>

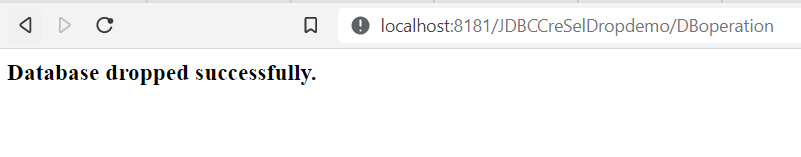
</web-app>

**OUTPUT**

****

****

****

****