DELIMITER $$

CREATE PROCEDURE select\_product\_by\_id(IN pprice decimal(10,2))

begin

SELECT \* FROM ecommerce.eproduct WHERE id=pprice;

end;

$$ DELIMITER

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="UTF-8">  
 <title>E-commerce application</title>  
</head>  
<body>  
<h1 align=center>Getting information about product</h1>  
<center>  
 <form method=post action="getinformation">  
 To get product information specify product ID and click "Get" <p>  
 <table width="50%" border="0">  
 <tr valign="top">  
 <td width="40%">Product ID</td>  
 <td><input type="number" name="productid" id="productid" maxlength=50></td>  
 </tr>  
 <tr valign="top">  
 <td colspan=2 width="100%">  
 <button>Get information</button>  
 </td>  
  
 </tr>  
 </table>  
 </form>  
  
 <a href="createandpopulate">Click here to create and populate product table</a>  
</center>  
</body>  
</html>

<?xml version="1.0" encoding="UTF-8"?>  
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_4\_0.xsd" id="WebApp\_ID" version="4.0">  
 <display-name>JDBC Statements and Resultsets</display-name>  
 <welcome-file-list>  
 <welcome-file>index.html</welcome-file>  
 <welcome-file>index.htm</welcome-file>  
 <welcome-file>index.jsp</welcome-file>  
 <welcome-file>default.html</welcome-file>  
 <welcome-file>default.htm</welcome-file>  
 <welcome-file>default.jsp</welcome-file>  
 </welcome-file-list>  
 <servlet>  
 <servlet-name>CreateAndPopulate</servlet-name>  
 <servlet-class>com.ecommerce.CreateAndPopulate</servlet-class>  
 </servlet>  
 <servlet>  
 <servlet-name>GetProductInformation</servlet-name>  
 <servlet-class>com.ecommerce.GetProductInformation</servlet-class>  
 </servlet>  
  
 <servlet-mapping>  
 <servlet-name>CreateAndPopulate</servlet-name>  
 <url-pattern>/createandpopulate</url-pattern>  
 </servlet-mapping>  
  
 <servlet-mapping>  
 <servlet-name>GetProductInformation</servlet-name>  
 <url-pattern>/getinformation</url-pattern>  
 </servlet-mapping>  
  
</web-app>

<?xml version="1.0" encoding="UTF-8"?>  
<project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>com.example</groupId>  
 <artifactId>task214</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <name>task214</name>  
 <packaging>war</packaging>  
  
 <properties>  
 <maven.compiler.target>1.8</maven.compiler.target>  
 <maven.compiler.source>1.8</maven.compiler.source>  
 <junit.version>5.7.1</junit.version>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>mysql</groupId>  
 <artifactId>mysql-connector-java</artifactId>  
 <version>8.0.27</version>  
 </dependency>  
 <dependency>  
 <groupId>javax.servlet</groupId>  
 <artifactId>javax.servlet-api</artifactId>  
 <version>4.0.1</version>  
 <scope>provided</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-api</artifactId>  
 <version>${junit.version}</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-engine</artifactId>  
 <version>${junit.version}</version>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-war-plugin</artifactId>  
 <version>3.3.1</version>  
 </plugin>  
 </plugins>  
 </build>  
</project>

url=jdbc:mysql://localhost:3306/ecommerce?useSSL=false  
userid=root  
password=vestaeagle

package com.ecommerce;  
  
  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;  
  
public class DBConnection {  
  
 private Connection connection;  
  
 public DBConnection(String dbURL, String user, String pwd) throws ClassNotFoundException, SQLException{  
  
 Class.*forName*("com.mysql.jdbc.Driver");  
 this.connection = DriverManager.*getConnection*(dbURL, user, pwd);  
 }  
  
 public Connection getConnection(){  
 return this.connection;  
 }  
  
 public void closeConnection() throws SQLException {  
 if (this.connection != null)  
 this.connection.close();  
 }  
}

package com.ecommerce;  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.PrintWriter;  
import java.math.BigDecimal;  
import java.sql.CallableStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
import java.util.Properties;  
  
@WebServlet(name = "CreateAndPopulate", value = "/CreateAndPopulate")  
public class CreateAndPopulate extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
 try {  
 PrintWriter out = response.getWriter();  
 out.println("<html><body>");  
  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
// Statement stmt = conn.getConnection().createStatement();  
// stmt.executeUpdate("CREATE TABLE IF NOT EXISTS eproduct (ID bigint primary key auto\_increment, name varchar(100), price decimal(10,2), date\_added timestamp default now()) ");  
// out.println("Table eproduct created successfully ");  
// stmt.executeUpdate("INSERT INTO eproduct (name, price) VALUES (\"Laptop\", 12000)");  
// stmt.executeUpdate("INSERT INTO eproduct (name, price) VALUES (\"Car\", 25000)");  
// stmt.executeUpdate("INSERT INTO eproduct (name, price) VALUES (\"House\", 35000)");  
// stmt.executeUpdate("INSERT INTO eproduct (name, price) VALUES (\"Mouse\", 15)");  
// stmt.executeUpdate("INSERT INTO eproduct (name, price) VALUES (\"Suit\", 1500)");  
// out.println("Table eproduct populated successfully");  
  
 CallableStatement stmtCallable = conn.getConnection().prepareCall("{call select\_product\_by\_id(?)}");  
 stmtCallable.setLong(1, 1);  
 ResultSet rst=stmtCallable.executeQuery();  
  
 while (rst.next()) {  
 out.println(rst.getInt("ID")  
 + ", " + rst.getString("name")  
 + ", " + rst.getBigDecimal("price")+"<Br>");  
 }  
  
  
 conn.closeConnection();  
  
  
 out.println("</body></html>");  
 conn.closeConnection();  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 }  
}

package com.ecommerce;  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.PrintWriter;  
import java.sql.CallableStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
import java.util.Properties;  
  
@WebServlet(name = "GetProductInformation", value = "/GetProductInformation")  
public class GetProductInformation extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
 try {  
  
 HttpSession session=request.getSession();  
 Long productId = Long.*parseLong*(request.getParameter("productid"));  
  
 PrintWriter out = response.getWriter();  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
 out.println("<html><body>");  
  
  
  
 if (productId>0) {  
  
 CallableStatement stmtCallable = conn.getConnection().prepareCall("{call select\_product\_by\_id(?)}");  
 stmtCallable.setLong(1, productId);  
 ResultSet rst=stmtCallable.executeQuery();  
  
  
  
 if (rst.next()) {  
 out.println("Product information about product with id="+rst.getInt("ID")  
 + ": name=" + rst.getString("name")  
 + ", price=" + rst.getBigDecimal("price") + "<Br>");  
 }  
  
 else {  
 out.println("Product with ID="+productId+" not found in the database <Br>");  
 }  
  
 stmtCallable.close();  
 }  
 else {  
 out.println("Invalid Id entered. Id must be larger then 0 <Br>");  
 }  
  
  
 out.println("</body></html>");  
 conn.closeConnection();  
  
  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
 } catch (NumberFormatException e) {  
 e.printStackTrace();  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
 }  
}