# Project Name: Adding a New Product in the Database.

## Source Code:

### index.jsp

<html>

<link rel="stylesheet"

    href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"

    integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"

    crossorigin="anonymous">

</head>

<body>

    <h3>Search for a product by ID :::</h3>

    <div style="width: 40%; height: 30%; margin: 20px;">

        <form class="form-inline" action="getProduct">

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Password</label>

                <!-- <input type="text" name="pid" class="form-control" placeholder="ID number"> -->

                <input type="number" id="replyNumber" name="pid" min="0"

                    data-bind="value:replyNumber"  Placeholder="Product ID" />

            </div>

            <button type="submit" class="btn btn-primary mb-2">Find</button>

        </form>

    </div>

    <!-- Show the user if the product saved successfully or not -->

    <%

    Object strMessage = request.getAttribute("message");

    if( strMessage != null)  { %>

    <h3><% out.println(strMessage.toString()); %></h3>

    <% }%>

    <!-- Alert the client that the id should be number not text -->

    <%

    Object idmessage = request.getAttribute("idMessage");

    if( idmessage != null)  { %>

    <h3><% out.println(idmessage.toString()); %></h3>

    <% }%>

    <h3>Add a New product:</h3>

    <div style="width: 40%; height: 50%; margin: 30px;">

        <form class="form-inline" action="saveProduct">

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product ID</label> <input type="number"

                    id="replyNumber" name="sid" min="0" data-bind="value:replyNumber" placeholder="Product ID" />

            </div>

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product Name</label> <input type="text"

                    class="form-control" name="sname" placeholder="Product name">

            </div>

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product Company</label> <input type="text"

                    class="form-control" name="scompany" placeholder="Product company">

            </div>

            <button type="submit" class="btn btn-primary mb-2">submit</button>

        </form>

    </div>

</body>

</html>

<!-- <input type="number" id="replyNumber" name="pid" min="0"

    data-bind="value:replyNumber" />

 -->

### showProduct.jsp

<%@page import="com.file.web.model.Product"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

    pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Insert title here</title>

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet"

    href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"

    integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"

    crossorigin="anonymous">

</head>

<body bgcolor="green">

    <h1>The Prouduct</h1>

    <%

    Object strMessage = request.getAttribute("message");

    if( strMessage != null)  { %>

    <h3><% out.println(strMessage.toString()); %></h3>

    <% }%>

    <%

    Product p1 = (Product) request.getAttribute("product");

    if( p1.getName() != null && p1.getId() >= 0 )  { %>

         <div style="width: 40%; height: 100%;">

        <table class="table table-striped table-dark">

            <thead>

                <tr>

                    <th scope="col">Product ID</th>

                    <th scope="col">Product Name</th>

                    <th scope="col">Product Company</th>

                </tr>

            </thead>

            <tbody>

                <tr>

                    <th scope="row">

                        <%

                        out.print(p1.getId());

                        %>

                    </th>

                    <td>

                        <%

                        out.print(p1.getName());

                        %>

                    </td>

                    <td>

                        <%

                        out.print(p1.getCompany());

                        %>

                    </td>

                </tr>

                <tr>

            </tbody>

        </table>

    </div>

      <% } else { %>

      <div class="p-3 mb-2 bg-info text-white" style=" width: 1000px; height: 200px;" >Sorry no product in this ID !</div>

      <% } %>

   </body>

</html>

</body>

</html>

### saveProduct.jsp

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

    pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Insert title here</title>

</head>

<body>

<h3>Add a New product:</h3>

<%

    Object strMessage = request.getAttribute("message");

    if( strMessage != null)  { %>

    <h3><% out.println(strMessage.toString()); %></h3>

    <% }%>

    <div style="width: 40%; height: 100%; margin: 20px;">

        <form class="form-inline" action="saveProduct">

        <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product ID</label>

                <input type="number" id="replyNumber" name="sid" min="0"

                    data-bind="value:replyNumber" />

            </div>

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product Name</label> <input type="text"

                    class="form-control" name="sname" placeholder="Product name">

            </div>

            <div class="form-group mx-sm-3 mb-2">

                <label class="sr-only">Product Company</label> <input type="text"

                    class="form-control" name="scompany" placeholder="Product company">

            </div>

            <button type="submit" class="btn btn-primary mb-2">submit</button>

        </form>

    </div>

</body>

</html>

### ProductDAO.java:

package com.file.web.dao;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import com.file.web.model.Product;

public class ProductDAO {

//  Create product method :

//  1-Connect to the database

//  2-Execute a query to get data from the database

//  3-Save the data in a new product object.

//  4-Return the product to be used in the (Product Controller)

    public Product getProduct(int pid) {

        Product product = new Product();

        try {

            //1-Connect to the database

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

            Connection connectionToDB = DriverManager

                    .getConnection("jdbc:mysql://localhost:3306/ecommerce?createDatabaseIfNotExist=true", "root", "Abc@123");

            Statement st = connectionToDB.createStatement();

            //2-Execute a query to get data from the database

            ResultSet resultData = st.executeQuery("select \* from eproduct where id=" + pid + ";");

            if (resultData.next()) {

                //3-Save the data in a new product object.

                product.setId(resultData.getInt("id"));

                product.setName(resultData.getString("name"));

                product.setCompany(resultData.getString("company"));

            }

        } catch (SQLException e) {

            System.out.println("sql problem:" + e);

        } catch (ClassNotFoundException e) {

            e.printStackTrace();

        }

        //4-Return the product to be used in the (Product Controller)

        return product;

    }

//  Save new product:

//      1-Get the data from the product object (Created in the saveProductController)

//      2-Connect to the database

//      3-Execute a query to see if the project exist already in database or not

//      4-If product exist, return the product and notify the user(Product exist)

//      5-If not, then save the data in a new product object.

    public Product saveProduct(int id, String sname, String scompany) {

        Product product = new Product(id, sname, scompany);

        //1-Get the data from the product object (Created in the saveProductController)

        System.out.println(product.getId());

        System.out.println(product.getName());

        System.out.println(product.getCompany());

        try {

            //2-Connect to the database

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

            Connection connection = DriverManager

                    .getConnection("jdbc:mysql://localhost:3306/ecommerce?createDatabaseIfNotExist=true", "root", "Abc@123");

            Statement st = connection.createStatement();

            //3-Execute a query to see if the project exist already in database or not

            ResultSet resultData = st.executeQuery("select \* from eproduct where id=" + product.getId() + ";");

            //4-If product exist, return the product and notify the user(Product exist)

            if (resultData.next()) {

                return product;

            }  else {

                //5-If not, then save the data in a new product object.

                int result = st.executeUpdate("INSERT INTO eproduct(id,name,company) VALUES (" + product.getId() + ",'"

                        + product.getName() + "','" + product.getCompany() + "');");

            }

        } catch (SQLException e) {

            System.out.println("Sorry the is a problem with database" + e);

        } catch (ClassNotFoundException ee) {

            // TODO: handle exception

            System.out.println("Sorry class not found"+ee);

        }

        return null;

    }

}

### Product.java:

package com.file.web.model;

public class Product {

    @Override

    public String toString() {

        return "Product [id=" + id + ", name=" + name + ", company=" + company + "]";

    }

    int id;

    String name;

    String company;

    public int getId() {

        return id;

    }

    public void setId(int id) {

        this.id = id;

    }

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public String getCompany() {

        return company;

    }

    public void setCompany(String company) {

        this.company = company;

    }

    public Product(int id, String name, String company) {

        super();

        this.id = id;

        this.name = name;

        this.company = company;

    }

    public Product() {

        super();

    }

}

### ProductController.java

package com.file.web;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.file.web.dao.ProductDAO;

import com.file.web.model.Product;

/\*\*

 \* Servlet implementation class ProductController

 \*/

public class ProductController extends HttpServlet {

    protected void doGet(HttpServletRequest request, HttpServletResponse response)

            throws ServletException, IOException {

        // Create a dispatcher and get the Product id from index.js

        // Create Product DAO to get the data from the database

        // Inside Product DAO Create Product object and save the data from the Database in it

        // Pass the Product object to showProduct.jsp to the user to see it.

        try {

            RequestDispatcher reqDis = request.getRequestDispatcher("showProducts.jsp");

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

            // Create Product DAO to get the data from the database

            // Inside Product DAO Create Product object and save the data from the Database in it

            ProductDAO pDAO = new ProductDAO();

            Product prd = pDAO.getProduct(pid);

            // Pass the Product object to showProduct.jsp to the user to see it.

            request.setAttribute("product", prd);

            reqDis.forward(request, response);

        } catch (Exception e) {

            // TODO: handle exception

            RequestDispatcher reqDisforID = request.getRequestDispatcher("index.jsp");

            request.setAttribute("idMessage", "Please Product ID must be a posative number only.");

            reqDisforID.forward(request, response);

        }

    }

}

### SaveProductController.java

package com.file.web;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.file.web.dao.ProductDAO;

import com.file.web.model.Product;

/\*\*

 \* Servlet implementation class ProductController

 \*/

public class SaveProductController extends HttpServlet {

    protected void doGet(HttpServletRequest request, HttpServletResponse response)

            throws ServletException, IOException {

//       Create a dispatcher and get the Product id,Product name, Product company from user (index.js)

//       Create Product DAO to save the entered data into the database

//       Inside Product DAO Create Product object and save the data from the user in it

//       Notify the user if the product saved successfully/not, exist/not

//       Alert the user, if the Product ID entered was a letters instead of positive numbers.

        RequestDispatcher reqDisforID = request.getRequestDispatcher("index.jsp");

        try {

            // Create a dispatcher and get the Product id,Product name, Product company from user (index.js)

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

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

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

            // Create Product DAO to save the entered data into the database

            ProductDAO pDAO = new ProductDAO();

            // Inside Product DAO Create Product object and save the data from the user in it

            Product checkresult = pDAO.saveProduct(sid, sname, scompany);

            if (checkresult == null) {

                // Notify the user if the product saved successfully/not, exist/not

                RequestDispatcher reqDis = request.getRequestDispatcher("index.jsp");

                request.setAttribute("message", "The Product added successfully");

                reqDis.forward(request, response);

            } else {

                // Notify the user if the product exist/not

                System.out.println("it is already exist");

                RequestDispatcher reqDis = request.getRequestDispatcher("index.jsp");

                request.setAttribute("message", "The Product already exists! ");

                reqDis.forward(request, response);

            }

        } catch (Exception e) {

            // Alert the user, if the Product ID entered was a letters instead of positive numbers.

            request.setAttribute("idMessage", "Please Product id must be a posative a number only.");

            reqDisforID.forward(request, response);

        }

    }

}

### Web.xml

<!DOCTYPE web-app PUBLIC

 "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"

 "http://java.sun.com/dtd/web-app\_2\_3.dtd" >

<web-app>

  <display-name>Archetype Created Web Application</display-name>

  <servlet>

    <servlet-name>GetProductController</servlet-name>

    <display-name>GetProductController</display-name>

    <description></description>

    <servlet-class>com.file.web.ProductController</servlet-class>

  </servlet>

  <servlet>

    <servlet-name>SaveProductController</servlet-name>

    <display-name>SaveProductController</display-name>

    <description></description>

    <servlet-class> com.file.web.SaveProductController</servlet-class>

  </servlet>

  <servlet-mapping>

    <servlet-name>GetProductController</servlet-name>

    <url-pattern>/getProduct</url-pattern>

  </servlet-mapping>

  <servlet-mapping>

    <servlet-name>SaveProductController</servlet-name>

    <url-pattern>/saveProduct</url-pattern>

  </servlet-mapping>

</web-app>