МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«Тольяттинский государственный университет»

|  |
| --- |
| *(наименование института полностью)* |
|  |
| (*Наименование учебного структурного подразделения*) |
| 09.03.03 Прикладная информатика |
| *(код и наименование направления подготовки / специальности)* |
| Разработка социальных и экономических информационных систем |
| *(направленность (профиль) / специализация)* |

**Практическое задание №4**

* по учебному курсу «[Технологии и средства конструирования программного обеспечения](https://edu.rosdistant.ru/course/view.php?id=11365" \o "Технологии и средства конструирования программного обеспечения)»

|  |  |  |
| --- | --- | --- |
| Обучающегося | Елшина И.А. |  |
|  |  |  |
| Группа | ПИбп-2004а |  |
|  |  |  |
| Преподаватель | Гущина О.М. |  |
|  | *(И.О. Фамилия)* |  |

Тольятти 2025

**Цель работы**

Сформировать навыки разработки и реализации веб-приложения.

**Задание.** Разработка веб-приложения. Создание простейшего сервлета.

**Задание.** Разработка веб-приложения.

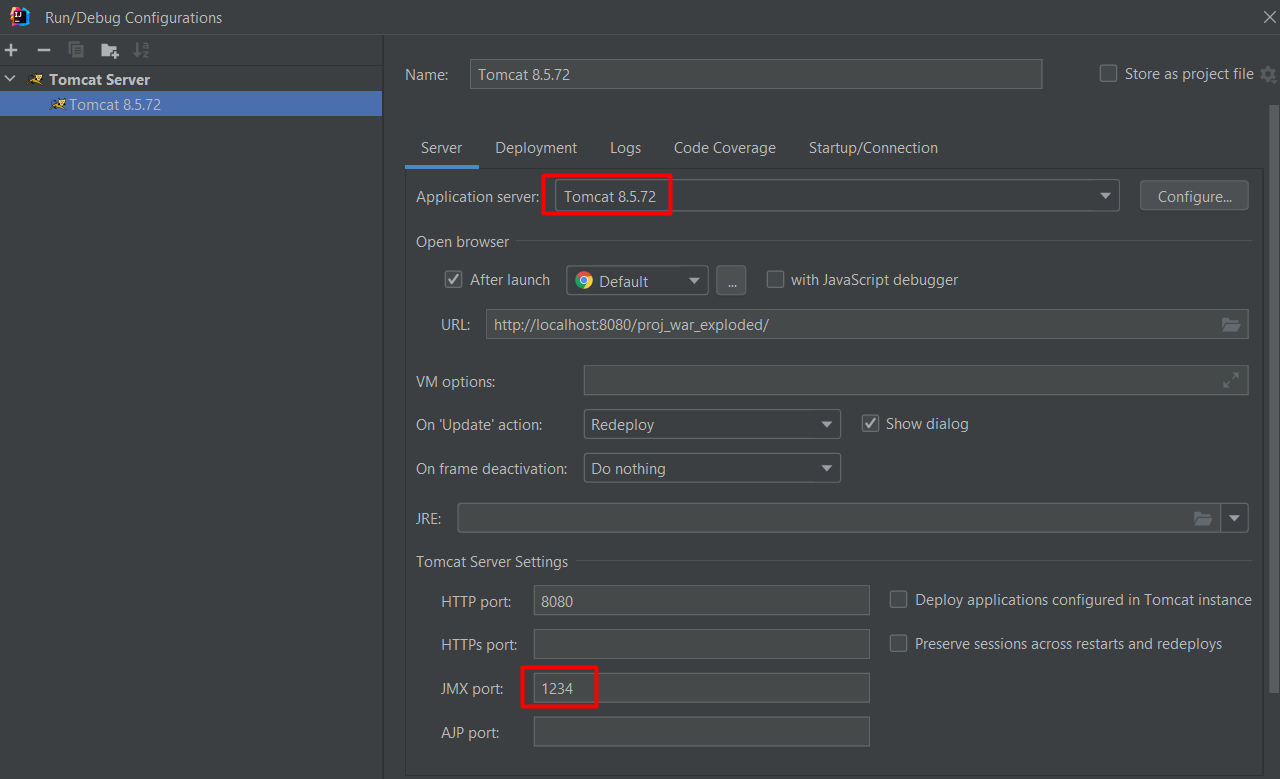
|  |  |
| --- | --- |
| № варианта | Предметная область |
| 2 | Аптечный склад |

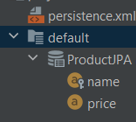
В качестве программной платформы было выбрано решение на основе Java Enterprise Edition c клиент-серверной архитектурой, включающей в себя:

* реляционную СУБД MySQL для хранения данных о товарно-материальных ценностях;
* веб-приложение на Java Servlets и JSP для обеспечения бизнес-логики и формирования пользовательского веб-интерфейса;
* веб-сервер Apache Tomcat в качестве контейнера сервлетов и JSP, а также среды исполнения Java-кода.

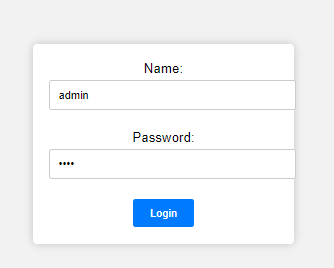
В результате спроектирована и частично реализована трехзвенная ИС со следующими функциями:

* ведение номенклатурного справочника лекарственных препаратов;
* учет остатков и оборота по каждой номенклатурной позиции;
* формирование отчетности по запасам для принятия управленческих решений.

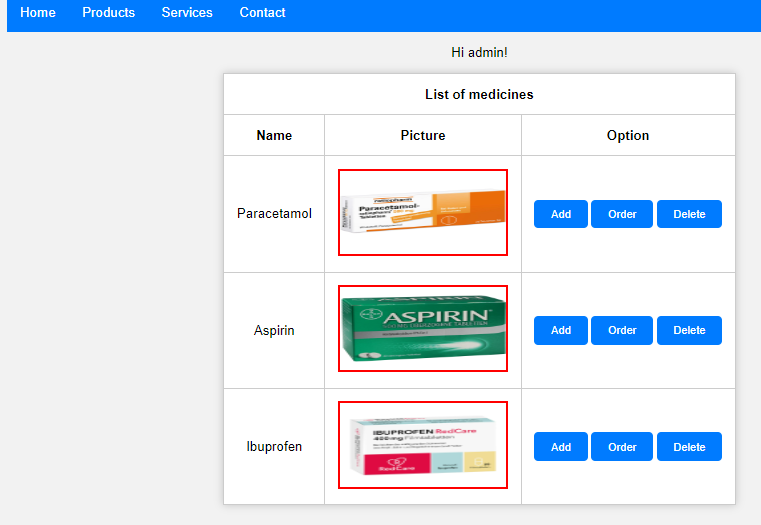




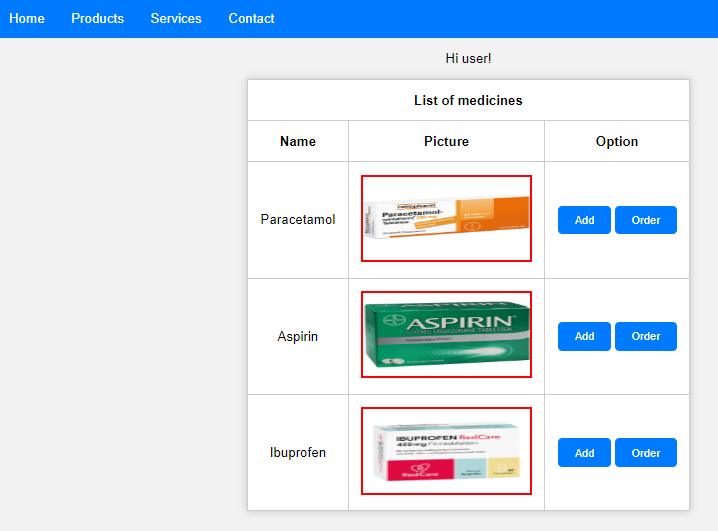
1. Запустить программу
2. Ввести логин и пароль (для админа – admin/pass)
3. Перейти в рабочий кабинет



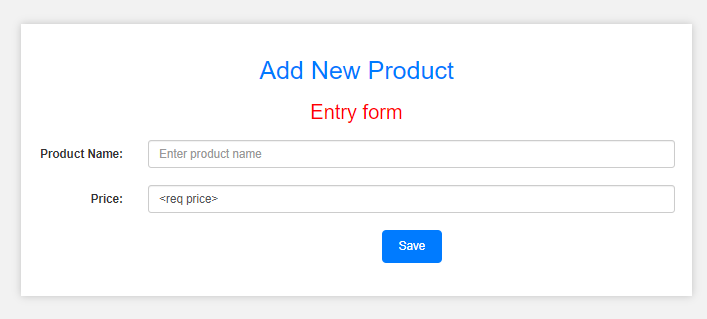
Окно для администратора:



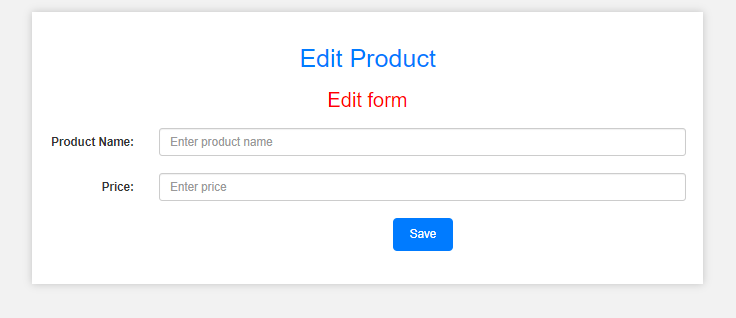
Окно для пользователя:



Форма для добавления лекарства:



Форма для изменения данных:



**Выводы**

В рамках данной работы была предпринята разработка прототипа распределенного веб-ориентированного программного обеспечения для автоматизации учета, анализа и оптимизации товарно-материальных запасов на складах предприятия оптовой торговли фармацевтическими препаратами.

В качестве инструментов реализации данного проекта были выбраны современные технологии корпоративной платформы Java Enterprise Edition, включающие в себя:

* реляционную СУБД MySQL для надежного хранения данных о лекарственных препаратах;
* веб-приложение на Java Servlets и JSP в качестве серверного приложения, реализующего бизнес-логику обработки данных о товарных запасах;
* веб-сервер Apache Tomcat в качестве контейнера сервлетов и JSP и среды выполнения Java-кода;
* технологию Bootstrap Framework для быстрой реализации пользовательского веб-интерфейса.

Разработанный прототип позволяет выполнять управление пользователями системы с разграничением прав доступа, вести номенклатурный справочник лекарственных средств, регистрировать движение материальных ценностей по складу, формировать аналитическую отчётность для принятия управленческих решений по оптимизации запасов.

Дальнейшие направления работ включают развитие подсистемы прогнозной аналитики, интеграцию со складским и бухгалтерским учетом, масштабирование и повышение отказоустойчивости ИС.

**Листинг программы**

HelloServlet.java

package com.example.proj;  
  
import java.io.\*;  
import javax.servlet.RequestDispatcher;  
import javax.servlet.ServletException;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
  
//Сайт для аптеки.  
// логин для аптекаря - admin/pass  
// логин для пользователя - user/user\_pass  
@WebServlet(name = "helloServlet", value = "/hello-servlet")

public class HelloServlet extends HttpServlet {

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

response.setContentType("text/html");

PrintWriter out = response.getWriter();

// Получаем параметры из запроса: логин и пароль

String n = request.getParameter("username");

String p = request.getParameter("userpass");

// Проверяем валидность логина и пароля для администратора и пользователя

ConnectionStatus adminStatus = LoginDao.validateAdmin(n, p);

ConnectionStatus userStatus = LoginDao.validateUser(n, p);

if (adminStatus == ConnectionStatus.WRONG\_PASSWORD || userStatus == ConnectionStatus.WRONG\_PASSWORD) {

// Если пароль неверный для администратора или пользователя, перенаправляем на страницу с ошибкой

RequestDispatcher rd = request.getRequestDispatcher("wrong-login-servlet");

rd.forward(request, response);

} else if (adminStatus == ConnectionStatus.OK) {

// Если администратор успешно вошел, перенаправляем на админскую страницу

RequestDispatcher rd = request.getRequestDispatcher("admin-servlet");

rd.forward(request, response);

} else if (userStatus == ConnectionStatus.OK) {

// Если пользователь успешно вошел, перенаправляем на приветственную страницу

RequestDispatcher rd = request.getRequestDispatcher("welcome-servlet");

rd.forward(request, response);

} else {

// Если ни один логин не соответствует, выводим сообщение об ошибке и перенаправляем на главную страницу

out.print("No such login exists");

RequestDispatcher rd = request.getRequestDispatcher("index.html");

rd.include(request, response);

}

out.close();

}

public void destroy() {

// Дополнительные действия при завершении работы сервлета (по желанию)

}

}

WrongLoginServlet.java

package com.example.proj;  
  
import javax.servlet.ServletException;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
import java.io.IOException;  
import java.io.PrintWriter;  
  
@WebServlet(name = "wrongLoginServlet", value = "/wrong-login-servlet")

public class WrongLoginServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Устанавливаем тип контента в ответе как HTML

response.setContentType("text/html");

// Получаем объект для записи в ответ

PrintWriter out = response.getWriter();

// Выводим сообщение об ошибке входа

out.print("Wrong login!\n");

// Закрываем поток

out.close();

}

}

WelcomeServlet.java

package com.example.proj;  
  
import java.io.IOException;  
import java.io.PrintWriter;  
  
import javax.servlet.ServletException;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
  
@WebServlet(name = "welcomeServlet", value = "/welcome-servlet")  
public class WelcomeServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Устанавливаем тип контента в ответе как HTML

response.setContentType("text/html");

// Получаем объект для записи в ответ

PrintWriter out = response.getWriter();

// Генерируем HTML-страницу

out.print("<!DOCTYPE html> \n" +

" <html>\n");

// Выводим приветственное сообщение для пользователя

out.print("<hr>Welcome back, user! Which medicine would you like to buy?</hr>");

// Вставляем кнопки с надписью "Buy" с использованием метода из класса Pictures

out.println(Pictures.buttonName("Buy"));

// Закрываем HTML-страницу

out.print("</html>\n");

// Закрываем поток

out.close();

}

}

Pictures.java

package com.example.proj;  
  
public class Pictures {

// Метод для генерации HTML-кода таблицы с кнопками

public static String buttonName(String name){

return "<table border=\"3\" bordercolor=\"red\" align=\"center\">\n" +

" <tr>\n" +

" <th colspan=\"3\">List of medicines</th> \n" +

" </tr>\n" +

" <tr>\n" +

" <th>Name</th>\n" +

" <th>Picture</th>\n" +

" <th>Option</th>\n" +

" </tr>\n" +

" <tr>\n" +

" <td>Paracetamol</td>\n" + // Закрытие тега должно быть </td> вместо </th>

" <td><img src=\"WebContent\\par.jpg\" border=3 height=100 width=200></img></td>\n" +

" <td><button type=\"button\">" + name + "</button>" +

"<button type=\"button\">Process order</button></td>\n" + // Закрытие тега должно быть </td> вместо </th>

" </tr>\n" +

" <tr>\n" +

" <td>Aspirin</td>\n" + // Закрытие тега должно быть </td> вместо </th>

" <td><img src=\"WebContent\\asp.jpg\" border=3 height=100 width=200></img></td>\n" +

" <td><button type=\"button\">" + name + "</button>" +

"<button type=\"button\">Process order</button></td>\n" + // Закрытие тега должно быть </td> вместо </th>

" </tr>\n" +

" <tr>\n" +

" <td>Ibuprofen</td>\n" + // Закрытие тега должно быть </td> вместо </th>

" <td><img src=\"WebContent\\ibu.jpg\" border=3 height=100 width=200></img></td>\n" +

" <td><button type=\"button\">" + name + "</button>" +

" <button type=\"button\">Process order</button></td>\n" + // Закрытие тега должно быть </td> вместо </th>

" </tr>\n" +

"</table>\n";

}

}

LoginDao.java

package com.example.proj;  
  
public class LoginDao {  
  
public static ConnectionStatus validateAdmin(String name, String pass) {

// Проверяем, является ли имя "admin" и пароль "pass" правильными

if (name.equals("admin") && pass.equals("pass"))

return ConnectionStatus.OK;

// Если имя "admin", но пароль неверный, возвращаем статус "WRONG\_PASSWORD"

if (name.equals("admin"))

return ConnectionStatus.WRONG\_PASSWORD;

// Если имя не "admin", возвращаем статус "NO"

return ConnectionStatus.NO;

}

public static ConnectionStatus validateUser(String name, String pass) {

// Проверяем, является ли имя "user" и пароль "user\_pass" правильными

if (name.equals("user") && pass.equals("user\_pass"))

return ConnectionStatus.OK;

// Если имя "user", но пароль неверный, возвращаем статус "WRONG\_PASSWORD"

if (name.equals("user"))

return ConnectionStatus.WRONG\_PASSWORD;

// Если имя не "user", возвращаем статус "NO"

return ConnectionStatus.NO;

}

ConnectionStatus.java

package com.example.proj;  
  
public enum ConnectionStatus {  
 *OK*, *WRONG\_PASSWORD*, *NO*}

AdminServlet.java

package com.example.proj;  
  
import java.io.IOException;  
import java.io.PrintWriter;  
  
import javax.servlet.ServletException;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
  
  
@WebServlet(name = "adminServlet", value = "/admin-servlet")

public class AdminServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Устанавливаем тип контента в ответе как HTML

response.setContentType("text/html");

// Получаем объект для записи в ответ

PrintWriter out = response.getWriter();

// Генерируем HTML-страницу

out.print("<!DOCTYPE html> \n" +

" <html>\n");

// Выводим приветственное сообщение для администратора

out.print("<hr>Welcome back, admin! Which medicine is to add today?</hr>");

// Вставляем кнопки с надписью "Add" с использованием метода из класса Pictures

out.println(Pictures.buttonName("Add"));

// Закрываем HTML-страницу

out.print("</html>\n");

// Закрываем поток

out.close();

}

}

header.jspf

<%@ **page** contentType="text/html; charset=UTF-8" language="java" %>  
<%@ **page** contentType="text/html; charset=UTF-8" %>  
  
<!-- Навигационное меню со стилями из HTML-кода -->  
<ul class="navbar">  
 <li><a href="#">Home</a></li>  
 <li><a href="#">Products</a></li>  
 <li><a href="#">Services</a></li>  
 <li><a href="#">Contact</a></li>  
</ul>

Productadd.jsp

<%@ **page** language="java" contentType="text/html; charset=UTF-8"  
 pageEncoding="UTF-8"%>  
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
<%@**page** import="com.example.proj.beans.Product"%>  
<%@**page** import="com.example.proj.dao.PharmacyDao"%>  
<%@**page** import="java.util.List"%>  
<%@ **page** import="javax.ejb.EJB" %>  
<%@ **page** import="com.example.proj.db.ProductJPA" %>  
<%@ **taglib** prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>  
<%@ **taglib** prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>  
<%  
 // Инициализация переменных  
 String errmsg = "";  
 String productname = "";  
 String price = "";  
  
 // Проверка типа HTTP-запроса (GET или POST)  
 if ("GET".equalsIgnoreCase(request.getMethod())) {  
 // Действий не требуется для GET-запроса  
 } else {  
 // Извлечение данных из параметров POST-запроса  
 productname = request.getParameter("productname");  
 price = request.getParameter("price");  
  
 // Проверка наличия данных  
 if(productname == null || productname.isEmpty()){  
 errmsg = "Product name can't be empty!";  
 } else if(price == null || price.isEmpty()){  
 errmsg = "Price can't be empty!";  
 }  
  
 double dprice = 0.0;  
 if (errmsg.isEmpty()) {  
 try {  
 // Попытка преобразования цены в число  
 dprice = Double.parseDouble(price);  
 // Создание объекта Product  
 Product product = new Product();  
 product.setProductName(productname);  
 product.setPrice(dprice);  
 // Создание записи в базе данных  
 PharmacyDao.*insert*(product);  
 // Перенаправление пользователя на страницу со списком продуктов  
 response.sendRedirect("productlist.jsp");  
 } catch (NumberFormatException nfe) {  
 errmsg = "Price must be a number!";  
 }  
 }  
 }  
  
 // Установка атрибутов для передачи данных в JSP  
 pageContext.setAttribute("errmsg", errmsg);  
 pageContext.setAttribute("productname", productname);  
 pageContext.setAttribute("price", price);  
%>  
<html>  
<head>  
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
 <title>Pharmacy</title>  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>  
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  
</head>  
<body>  
<**jsp:include** page="header.jsp" />  
<div class="container">  
 <h2>Add New Product</h2>  
 <h3 style='color:red'>${errmsg}</h3>  
 <form class="form-horizontal" action="productadd.jsp" method="Post">  
 <div class="form-group">  
 <label class="control-label col-sm-2" for="email">Product Name:</label>  
 <div class="col-sm-10">  
 <input class="form-control" id="productname" placeholder="Enter product name" name="productname" value="${productname}">  
 </div>  
 </div>  
 <div class="form-group">  
 <label class="control-label col-sm-2" for="pwd">Price:</label>  
 <div class="col-sm-10">  
 <input class="form-control" id="price" placeholder="Enter price" name="price" value="${price}">  
 </div>  
 </div>  
 <div class="form-group">  
 <div class="col-sm-offset-2 col-sm-10">  
 <button type="submit" class="btn btn-primary">Save</button>  
 </div>  
 </div>  
 </form>  
</div>  
</body>  
</html>

Productdel.jsp

<%@ **page** language="java" contentType="text/html; charset=UTF-8"  
 pageEncoding="UTF-8"%>  
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
<%@**page** import="com.example.proj.dao.PharmacyDao"%>  
<%  
 // Инициализация переменных  
 String id = request.getParameter("id");  
 String errmsg = "";  
  
 // Проверка наличия параметра "id"  
 if (id == null || id.isEmpty()) {  
 errmsg = "Invalid parameter!";  
 } else {  
 // Преобразование "id" в целое число  
 int productid = Integer.parseInt(id);  
  
 // Проверка существования продукта по "id"  
 if (PharmacyDao.*exists*(productid)) {  
 // Удаление продукта  
 PharmacyDao.*delete*(productid);  
 // Перенаправление пользователя на страницу со списком продуктов  
 response.sendRedirect("productlist.jsp");  
 } else {  
 errmsg = "No Product found!";  
 }  
 }  
  
 // Установка атрибутов для передачи данных в JSP  
 pageContext.setAttribute("errmsg", errmsg);  
%>  
<html>  
<head>  
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
 <title>Pharmacy</title>  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>  
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  
</head>  
<body>  
<**jsp:include** page="header.jsp" />  
<div>  
 <h3>Delete Product</h3>  
 <h3 style='color:red'>${errmsg}</h3>  
</div>  
</body>  
</html>

Productedit.jsp

<%@ **page** language="java" contentType="text/html; charset=UTF-8"  
 pageEncoding="UTF-8"%>  
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
<%@**page** import="com.example.proj.beans.Product"%>  
<%@**page** import="com.example.proj.dao.PharmacyDao"%>  
<%@**page** import="java.util.List"%>  
<%@ **taglib** prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>  
<%@ **taglib** prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>  
<%  
 // Инициализация переменных  
 String id = request.getParameter("id");  
 String errmsg = "";  
 String productname = "";  
 String price = "";  
  
 // Проверка наличия параметра "id"  
 if (id == null || id.isEmpty()) {  
 errmsg = "Invalid parameter!";  
 } else {  
 // Преобразование "id" в целое число  
 int productid = Integer.parseInt(id);  
  
 // Проверка метода HTTP запроса  
 if ("GET".equalsIgnoreCase(request.getMethod())) {  
 // Получение продукта по "id" и заполнение данных для редактирования  
 Product product = PharmacyDao.getProduct(productid);  
 productname = product.getProductName();  
 price = Double.toString(product.getPrice());  
 } else {  
 // Получение данных из параметров запроса  
 productname = request.getParameter("productname");  
 price = request.getParameter("price");  
  
 // Проверка наличия и валидность данных  
 if(productname == null || productname.isEmpty()){  
 errmsg = "Product name can't be empty!";  
 } else if(price == null || price.isEmpty()){  
 errmsg = "Price can't be empty!";  
 }  
  
 double dprice = 0.0;  
 if (errmsg.isEmpty()) {  
 try {  
 // Преобразование цены в число  
 dprice = Double.parseDouble(price);  
 } catch (NumberFormatException nfe) {  
 errmsg = "Price must be number!";  
 }  
 // Получение продукта по "id" и обновление данных  
 Product product = PharmacyDao.*getProduct*(productid);  
 product.setProductName(productname);  
 product.setPrice(dprice);  
 // Обновление продукта  
 PharmacyDao.*update*(product);  
 // Перенаправление пользователя на страницу со списком продуктов  
 response.sendRedirect("productlist.jsp");  
 }  
 }  
 }  
  
 // Установка атрибутов для передачи данных в JSP  
 pageContext.setAttribute("errmsg", errmsg);  
 pageContext.setAttribute("

Productlist.jsp

<%@ **page** language="java" contentType="text/html; charset=UTF-8"  
 pageEncoding="UTF-8"%>  
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
<%@**page** import="com.example.proj.beans.Product"%>  
<%@**page** import="com.example.proj.dao.PharmacyDao"%>  
<%@**page** import="java.util.List"%>  
<%@ **taglib** prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>  
<%@ **taglib** prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>  
<%  
 // Инициализация переменных  
 String errmsg = "";  
 List<Product> products = PharmacyDao.*getList*();  
  
 // Проверка наличия продуктов  
 if (products == null || products.size() == 0) {  
 errmsg = "There is no product!";  
 }  
  
 // Установка атрибутов для передачи данных в JSP  
 pageContext.setAttribute("errmsg", errmsg);  
 pageContext.setAttribute("products", products);  
%>  
<html>  
<head>  
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
 <title>Pharmacy</title>  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>  
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  
</head>  
<body>  
<**jsp:include** page="header.jsp" />  
  
<div class="container">  
 <h2>Products</h2>  
 <p>Data from Mysql</p>  
 <table class="table">  
 <thead>  
 <tr>  
 <th>Product ID</th>  
 <th>Product Name</th>  
 <th>Price</th>  
 <th>Operations</th>  
 </tr>  
 </thead>  
 <tbody>  
 <**c:choose**>  
 <**c:when** test="${not empty errmsg}">  
 <tr style='color:red'><td>${errmsg}</td></tr>  
 </**c:when**>  
 <**c:otherwise**>  
 <**c:forEach** var="product" items="${products}">  
 <tr>  
 <td><**c:out** value="${product.productId}"/></td>  
 <td><**c:out** value="${product.productName}"/></td>  
 <td><**fmt:setLocale** value="en\_US"/><**fmt:formatNumber** value="${product.price}" type="currency"/></td>  
 <td><a class="btn btn-success" href="productedit.jsp?id=${product.productId}">Edit</a>&nbsp;<a class="btn btn-danger" href="productdel.jsp?id=${product.productId}" onclick="return confirm('Are you sure to delete this product?')">Delete</a></td>  
 </tr>  
 </**c:forEach**>  
 </**c:otherwise**>  
 </**c:choose**>  
 </tbody>  
 </table>  
</div>  
</body>  
</html>