

**МИНОБРНАУКИ РОССИИ  
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ  
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
«ЛЭТИ» ИМ. В.И.УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра ВТ**

**ОТЧЕТ  
по лабораторной работе №5  
по дисциплине «Java программирование  
интернет-приложений»  
Тема: Построение WEB-приложений с использованием  
технологии JSP**

Преподаватель

\_\_\_\_\_  
\_\_\_\_\_

Репин С.А.

Павловский М.Г.

Санкт-Петербург  
2022

## СОДЕРЖАНИЕ

1. Цель работы	3
2. Выполнение работы	3
3. Вывод	5
Исходный код MedicineRequest.jsp	6
Исходный код MedicineList.jsp	6
Исходный код ErrorManager.jsp	7

## 1. ЦЕЛЬ РАБОТЫ

Знакомство с технологией построения Web-приложений на основе JSP.

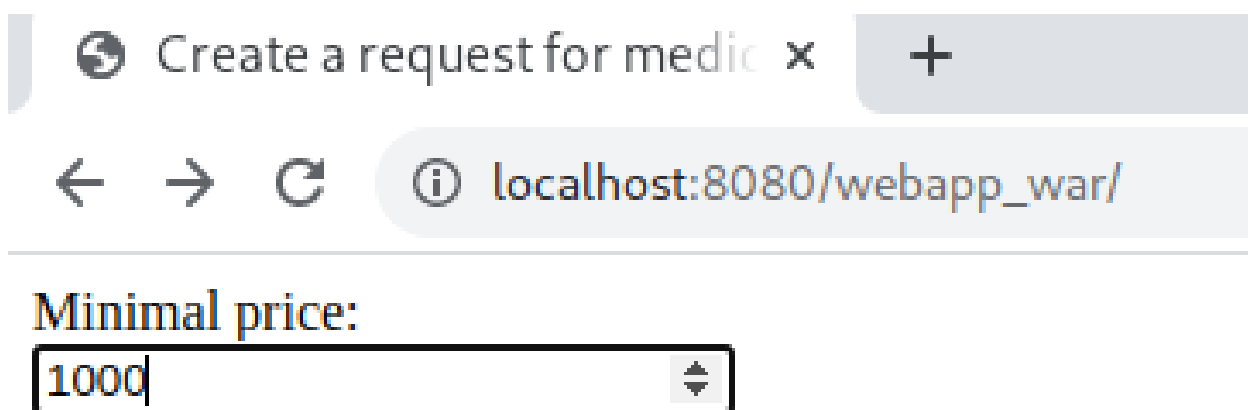
## 2. ВЫПОЛНЕНИЕ РАБОТЫ

С помощью технологии JSP разработана программа, аналогичная такой из лабораторной работы № 3.

В качестве предметной области взята тема из курсового проекта по объектно-ориентированному программированию на языке Java: разработка средства управления аптеков. Разработана JSP-страница, отображающий список лекарств в аптеке (ID, имя, количество единиц, цена единицы) и способный фильтровать лекарства по максимальной цене.

Проект содержит три JSP-страницы:

- форма ввода критерия фильтрации – максимальной цены (MedicineRequest.jsp);
- страница, отображающая результат фильтрации (MedicineList.jsp);
- страница, отображаемая при ошибке (ErrorManager.jsp).



The screenshot shows a web browser window with a single tab titled "Create a request for medic x". The address bar displays "localhost:8080/webapp\_war/". Below the address bar, there is a form with a label "Minimal price:" in blue text. Underneath the label is a text input field with a black border, containing the number "1000". To the right of the input field is a small, light gray button with a black up/down arrow icon.

Рис. 2.1 Ввод критерия фильтрации

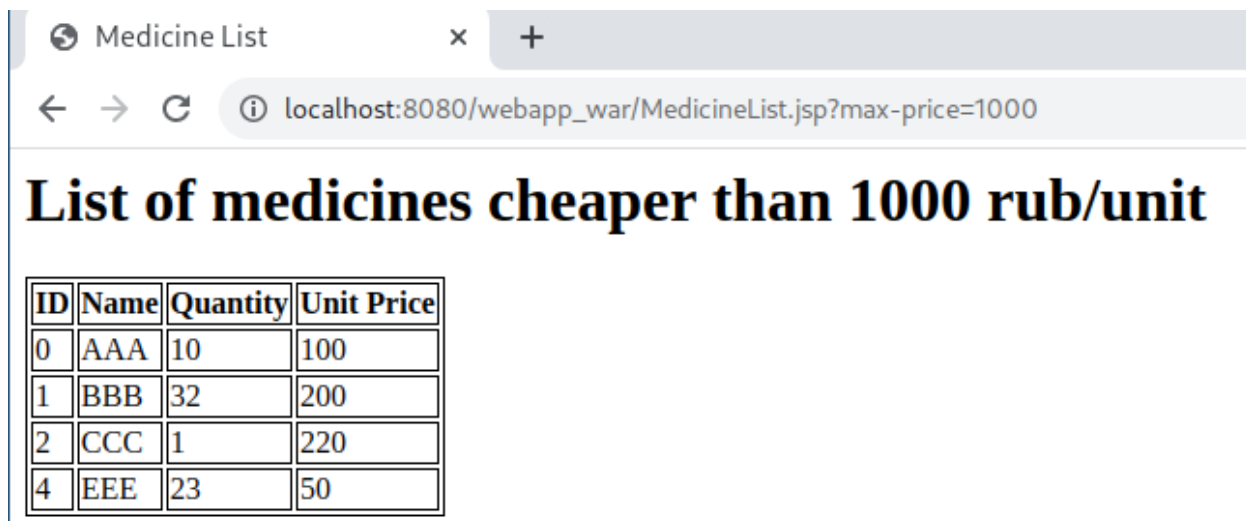


Рис. 2.2 Отображение фильтрованного списка

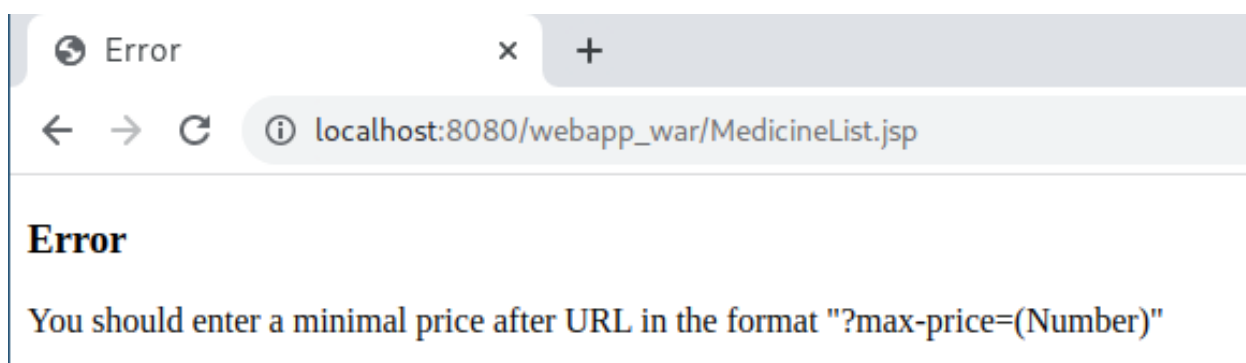


Рис. 2.3 Отображение ошибки, при неправильных параметрах запроса

OVERVIEW
PACKAGE
CLASS
USE
TREE
INDEX
HELP

SUMMARY: NESTED | FIELD | CONSTR | METHOD
DETAIL: FIELD | CONSTR | METHOD

Package services

### Class MedicineService

java.lang.Object
services.MedicineService

```

public class MedicineService
extends Object

```

#### Constructor Summary

Constructors	
Constructor	Description
MedicineService()	Конкструктор.

#### Method Summary

All Methods	Instance Methods	Concrete Methods	
Modifier and Type	Method		Description
List<Medicine>	findAll()		Возвращает список всех доступных лекарств.
List<Medicine>	findByMaxPrice(int maxPrice)		Возвращает список лекарств, у которых цена за штуку меньше `maxPrice`.

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Constructor Details

MedicineService

```

public MedicineService()

```

Конкструктор.

#### Method Details

##### findAll

```

public List<Medicine> findAll()

```

Возвращает список всех доступных лекарств.

##### findByMaxPrice

```

public List<Medicine> findByMaxPrice(int maxPrice)

```

Возвращает список лекарств, у которых цена за штуку меньше `maxPrice`.

Рис. 2.4 Пример документации, сгенерированной JavaDoc

### 3. ВЫВОД

В ходе выполнения лабораторной работы были получены практические на-  
выки разработки веб-приложений с применением JSP-страниц.

## ИСХОДНЫЙ КОД MEDICINEREQUEST.JSP

```
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<head>
    <title>Create a request for medicines</title>
</head>
<body>
    <form action="MedicineList.jsp">
        <label for="max-price">Minimal price:</label><br>
        <input type="number" id="max-price" name="max-price"
            autocomplete="off" min="0" value="0">
    </form>
</body>
</html>
```

## ИСХОДНЫЙ КОД MEDICINELIST.JSP

```
<%@ page import="services.MedicineService" %>
<%@ page import="models.Medicine" %>
<%@ page contentType="text/html; charset=UTF-8" %>
<html>
<head>
    <title>Medicine List</title>
    <style>
        table, th, td {
            border: 1px solid black;
        }
    </style>
</head>
<body>
    <%! static final MedicineService medicineService =
        new MedicineService();
    %>

    <%
        request.setCharacterEncoding("UTF-8");
        String maxPriceStr = request.getParameter("max-price");
        RequestDispatcher dispatcher;
        if (maxPriceStr == null) {
            dispatcher = request.getServletContext()
                .getRequestDispatcher("/ErrorManager.jsp");
            dispatcher.forward(request, response);
            return;
        }
        int maxPrice = Integer.parseInt(maxPriceStr);
```

```

%>

<h1>List of medicines cheaper than <%=maxPrice%> rub/unit</h1>
<table>
  <tr>
    <td><b>ID</b></td>
    <td><b>Name</b></td>
    <td><b>Quantity</b></td>
    <td><b>Unit Price</b></td>
  </tr>
  <% for (Medicine med :
    medicineService.findByMaxPrice(maxPrice)) {%>
    <tr>
      <% for (String field : med.fieldsToStringList()) { %>
        <td><%=field%></td>
      <% } %>
    </tr>
  <% } %>
</table>
</body>
</html>

```

## ИСХОДНЫЙ КОД ERRORMANAGER.JSP

```

<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Error</title>
</head>
<body>
  <h3>Error</h3>
  <p>
    You should enter a minimal price after URL in the format
    "<%=max-price%>=(Number)"
  </p>
</body>
</html>

```