

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

Кафедра ВТ

**ОТЧЕТ
по лабораторной работе №7**

**по дисциплине «Java программирование
интернет-приложений»**

**Тема: Организация передачи данных между запросами
пользователей**

Студент гр. 8307

Репин С.А.

Преподаватель

Павловский М.Г.

Санкт-Петербург
2022

СОДЕРЖАНИЕ

| | |
|----------------------------------|---|
| 1. Цель работы | 3 |
| 2. Выполнение работы | 3 |
| 3. Вывод | 5 |
| Исходный код Login.jsp | 5 |
| Исходный код LoginProcessor.java | 6 |
| Исходный код Welcome.jsp | 8 |

1. ЦЕЛЬ РАБОТЫ

Знакомство с методами передачи информации между соединениями, открываемыми в рамках одного сеанса работы пользователя

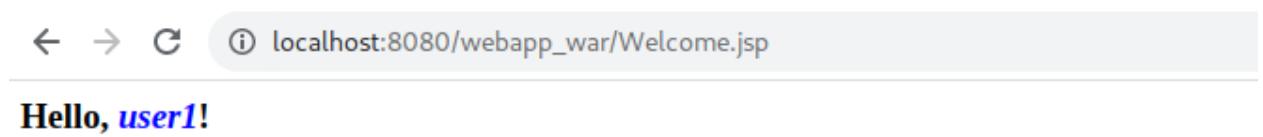
2. ВЫПОЛНЕНИЕ РАБОТЫ

В проект добавлена страницы Login.jsp, которая содержит форму для ввода имени пользователя и цвета. Эта страница передает GET-запросом данные в сервер LoginProcessor, который создает нужные cookie и переменные HTTP-сессии (имя пользователя, цвет пользователя, число обращений к странице и дату последнего обращения). Затем сервер переадресует запрос на страницу Welcome.jsp, которая читает созданные переменные и отображает соответствующий текст пользователю.

The screenshot shows a web browser window with the following details:

- Title Bar:** Shows the title "Login" and a "+" button for opening new tabs.
- Address Bar:** Displays the URL "localhost:8080/webapp_war/".
- Form Fields:** Two input fields:
 - Username:** Contains the value "user1".
 - Your color:** Contains the value "blue".
- Buttons:** A single "Login" button.

Рис. 2.1 Ввод данных на страницу Login.jsp



This page has been visited 1 time(s). Last on 2022-03-19T15:51:42.071343346Z

Рис. 2.2 Результат на странице Welcome.jsp

| Name | Value | Domain | Path | Expires / ... | Size | HttpOnly |
|------------------|--|-----------|-------------|---------------|------|----------|
| user.name | user1 | localhost | /webapp_... | 2022-03-2... | 14 | |
| Idea-f5030c7d | d7dc96bb-e350-46aa-84a4-698f3b7ef842 | localhost | / | 2032-02-1... | 49 | ✓ |
| JSESSIONID | 519a23a228cec324340db3f977da | localhost | / | Session | 38 | ✓ |
| user.color | blue | localhost | /webapp_... | 2022-03-2... | 14 | |
| treeForm_tree-hi | treeForm:tree:applications:lab4-1_0-SNAPSHOT | localhost | / | Session | 60 | |
| io | mCnNF9WxbVJfjkrxAAAD | localhost | / | Session | 22 | ✓ |
| JSESSIONID | A347F8A77A5C7025334DC0F9A28A42A | localhost | /webapp_... | Session | 42 | ✓ |

Рис. 2.3 Получившиеся в итоге cookie

Login

localhost:8080/webapp_war/

Username:
user2

Your color:
red

Login

Рис. 2.4 Ввод данных на страницу Login.jsp

localhost:8080/webapp_war/Welcome.jsp

Hello, user2!

This page has been visited 2 time(s). Last on 2022-03-19T15:52:03.076661037Z

Рис. 2.5 Результат на странице Welcome.jsp

| Name | Value | Domain | Path | Expires / ... | Size | HttpOnly |
|------------------|--|-----------|-------------|---------------|------|----------|
| user.name | user2 | localhost | /webapp_... | 2022-03-2... | 14 | |
| Idea-f5030c7d | d7dc96bb-e350-46aa-84a4-698f3b7ef842 | localhost | / | 2032-02-1... | 49 | ✓ |
| JSESSIONID | 519a23a228cec324340db3f977da | localhost | / | Session | 38 | ✓ |
| user.color | red | localhost | /webapp_... | 2022-03-2... | 13 | |
| treeForm_tree-hi | treeForm:tree:applications:lab4-1_0-SNAPSHOT | localhost | / | Session | 60 | |
| io | mCnNF9WxbVJfjkrxAAAD | localhost | / | Session | 22 | ✓ |
| JSESSIONID | 7A10ED0D9E94FFADE55213060CFBFEB | localhost | /webapp_... | Session | 42 | ✓ |

Рис. 2.6 Получившиеся в итоге cookie

Username:
user3

Your color:
yellow

Login

Рис. 2.7 Ввод данных на страницу Login.jsp

Hello, user3!

This page has been visited 3 time(s). Last on 2022-03-19T15:55:52.177295729Z

Рис. 2.8 Результат на странице Welcome.jsp

| Name | Value | Domain | Path | Expires / ... | Size | HttpOnly |
|------------------|--|-----------|-------------|---------------|------|----------|
| user.name | user3 | localhost | /webapp_... | 2022-03-2... | 14 | |
| Idea-f5030c7d | d7dc96bb-e350-46aa-84a4-698f3b7ef842 | localhost | / | 2032-02-1... | 49 | ✓ |
| JSESSIONID | 519a23a228cec324340db3f977da | localhost | / | Session | 38 | ✓ |
| user.color | yellow | localhost | /webapp_... | 2022-03-2... | 16 | |
| treeForm_tree-hi | treeForm:tree:applications:lab4-1_0-SNAPSHOT | localhost | / | Session | 60 | |
| io | mCnNF9WxbVjfkxAAAAD | localhost | / | Session | 22 | ✓ |
| JSESSIONID | 7A10ED0D9E94FADE55213060CFBFEAB | localhost | /webapp_... | Session | 42 | ✓ |

Рис. 2.9 Получившиеся в итоге cookie

3. ВЫВОД

В ходе выполнения лабораторной работы были получены практические навыки использования cookie и переменных HTTP-сессии в web-приложения на языке Java.

ИСХОДНЫЙ КОД LOGIN.JSP

```
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
```

```

<head>
    <title>Login</title>
</head>
<body>
    <form action="LoginProcessor">
        <label for="username">Username:</label>
        <br>
        <input id="username" name="userName" autocomplete="off" />
        <br>
        <label for="color">Your color: </label>
        <br>
        <input id="color" name="userColor" autocomplete="off" />
        <br><br>
        <input type="submit" value="Login" />
    </form>
</body>
</html>

```

ИСХОДНЫЙ КОД LOGINPROCESSOR.JAVA

```

package app.servlet;

import jakarta.servlet.http.*;
import jakarta.servlet.annotation.*;

import java.io.IOException;
import java.net.URLEncoder;
import java.nio.charset.StandardCharsets;
import java.time.Instant;
import java.util.Optional;

/**
 * Сервлет, обрабатывающий попытку входа пользователя.
 */
@WebServlet(name = "LoginProcessor", value = "/LoginProcessor")
public class LoginProcessor extends HttpServlet {
    /**
     * Обрабатывает GET-запрос.
     * @param request Объект запроса
     * @param response Объект ответа
     * @throws IOException Ошибка ввода/вывода
     */
    @Override
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
            throws IOException {
        // Получаем HTTP-сессию.

```

```

var session = request.getSession(true);

response.setContentType("text/html; charset=UTF-8");
request.setCharacterEncoding("utf-8");

// Читаем параметры, переданные из формы.
var userName = readParameter(request, response, "userName");
var userColor = readParameter(request, response, "userColor");

if (userName == null || userColor == null) {
    return;
}

// Создаем куки с именем пользователя и его цветом.
createCookie(response, "user.name", userName);
createCookie(response, "user.color", userColor);
// Обновляем переменные сессии:
//     - Инкрементируем число посещений.
//     - Обновляем дату последнего входа.
updateOrCreateSessionAttribute(session,
                               "page.welcome.accessedCount",
                               x -> x.map(y -> ((Integer)y) + 1).orElse(1));
updateOrCreateSessionAttribute(session,
                               "page.welcome.accessedDate",
                               x -> Instant.now());

// Производим редирект на целевую страницу.
var redirectPage = request.getContextPath() + "/Welcome.jsp";
response.sendRedirect(response.encodeRedirectURL(redirectPage))
}

/**
* Создает новое куки.
* @param response Объект ответа.
* @param name Имя куки.
* @param value Значение куки.
*/
private void createCookie(HttpServletRequest response,
                           String name, String value) {
    var c = new Cookie(name, URLEncoder.encode(value,
                                                StandardCharsets.UTF_8));
    c.setMaxAge(60 * 60 * 24);
    response.addCookie(c);
}

/**
* Представляет метод, обновляющий значение переменной HTTP-сессии.
*/
interface HttpSessionAttributeUpdater {
    Object update(Optional<Object> oldValue);
}

```

```

    /**
     * Обновляет переменную HTTP-сессии.
     * @param session Объект сессии.
     * @param name Имя переменной.
     * @param updater Объект-обновитель.
     */
    private void updateOrCreateSessionAttribute(HttpSession session,
                                                String name, HttpSessionAttributeUpdater updater) {
        var oldValue = session.getAttribute(name);
        var newValue = updater.update(Optional.ofNullable(oldValue));
        session.setAttribute(name, newValue);
    }

    /**
     * Читает параметр GET-запроса.
     *
     * При отсутствии параметра устанавливает HTTP-код и сообщение.
     * При отсутствии параметра устанавливает HTTP-код и сообщение.
     *
     * @param request Объект запроса.
     * @param response Объект ответа.
     * @param name Имя параметра.
     * @return Значение параметра.
     * @throws IOException Ошибка ввода/вывода.
     */
    private String readParameter(HttpServletRequest request,
                                HttpServletResponse response, String name)
            throws IOException {
        var parameter = request.getParameter(name);
        if(parameter == null) {
            var error = "Parameter \\" + name + "\\ is not set";
            response
                .sendError(HttpServletResponse.SC_BAD_REQUEST, error);
        }
        return parameter;
    }
}

```

ИСХОДНЫЙ КОД WELCOME.JSP

```

<%@ page import="java.util.Optional" %>
<%@ page import="java.util.Arrays" %>
<%@ page import="java.time.Instant" %>
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<%!
    /**
     * Ищет куки по имени и возвращает его значение.
     */

```

```

* @param request Объект запроса.
* @param name имя куки.
* @return Значение куки, если оно найдено.
*/
Optional<String> findCookieValue(HttpServletRequest request,
    String name) {
    return Arrays.stream(
        request.getCookies())
        .filter(x -> x.getName().equals(name))
        .findFirst()
        .map(Cookie::getValue);
}
%>
<%
// Находит значения переменных, сохраненные в куки.
String userName = findCookieValue(request, "user.name")
    .orElse("Anon");
String userColor = findCookieValue(request, "user.color")
    .orElse("black");
// Находит значения переменных, сохраненные в сессии.
Integer accessedCount =
    (Integer)session.getAttribute("page.welcome.accessedCount");
Instant accessedDate =
    (Instant)session.getAttribute("page.welcome.accessedDate");
%>

<html>
<head>
    <title>Title</title>
    <style>
        em {
            color: <%=userColor%>;
        }
    </style>
</head>
<body>
    <h3>
        Hello, <em><%=userName%></em> !
    </h3>
    <br>
    <h3>
        This page has been visited <em><%=accessedCount%></em> time(s).
        Last on <em><%=accessedDate%></em>
    </h3>
</body>
</html>

```