**МИНОБРНАУКИ РОССИИ**

**САНКТ–ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра ВТ**

**ОТЧЕТ**

**по лабораторной работе №6**

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

**Тема: «ОРГАНИЗАЦИЯ ПЕРЕДАЧИ ДАННЫХ**

**МЕЖДУ ЗАПРОСАМИ ПОЛЬЗОВАТЕЛЕЙ»**

Cтудентка гр. 7306 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мищенко А.В.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Павловский М.Г.

Санкт–Петербург

2021

**Цель работы:**

Знакомство с методами передачи информации между соединениями, открываемыми в рамках одного сеанса работы пользователя.**Задание:**

1. Создайте приложение, в котором JSP-страница запрашивает имя пользователя и цвет страницы вывода информации с сохранением этих параметров в Cookie. Эта же страница должна в переменных сессии подсчитывать количество обращений пользователей и дату последнего обращения, а затем передать управление другой JSP-странице, которая будет отображать содержимое Cookie и переменных сессии в заданном цвете.
2. Снимите несколько скриншотов окна настроек браузера, в котором отображаются различные значения Cookie, а также скриншоты результатов работы приложения при различных входных параметрах.

**Ход работы:**

Для выполнения лабораторной работы было написано приложение, состоящее из двух jsp файлов и сервлета на Java. Их исходные коды представлены в приложении. На рисунках 1-5 показана работа приложения.

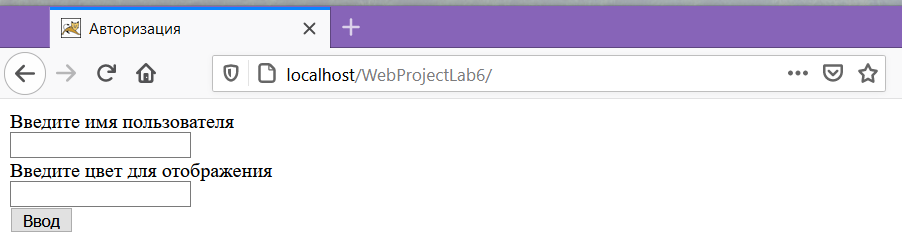


Рисунок 1. Отображение страницы ввода

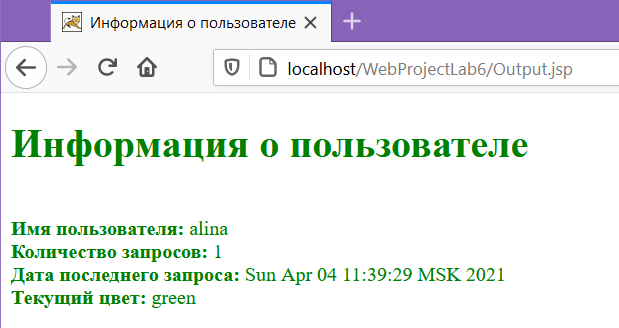


Рисунок 2. Вывод с параметрами alina и green

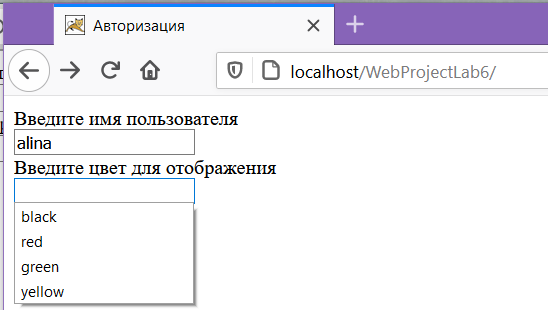


Рисунок 3. Выбор уже введенных значений цвета

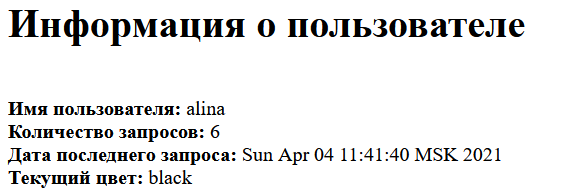


Рисунок 4. Вывод с параметрами alina и black

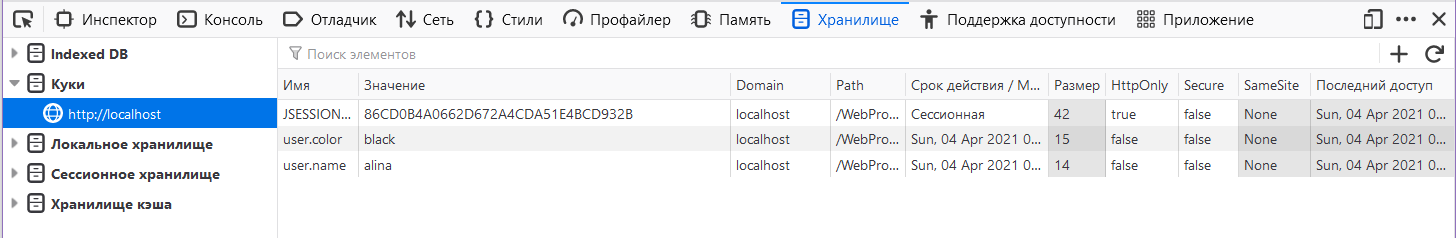


Рисунок 5. Файлы cookie на текущей странице

**Вывод:**

В результате выполнения лабораторной работы я научилась использовать cookie для передачи информации между соединениями, открываемыми в рамках одного сеанса работы пользователя.

**Приложение:**

*Приложение 1.1.Файл Start.jsp*

<%@page import=*"java.net.URLDecoder"*%>

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Авторизация</title>

</head>

<body>

<form METHOD=*GET* action=*"Processor"*>Введите имя пользователя<br>

<INPUT TYPE=*TEXT* NAME=*"name"*

<%

// Выбор всех Cookie

Cookie [] cookies = request.getCookies();

**if** (cookies != **null**)

**for** (**int** i = 0; i < cookies.length; i++)

**if** ("user.name".equals(cookies[i].getName())) {

// Запись значения в поле ввода, если найден Cookie

out.print(" value='" + URLDecoder.decode(cookies[i].getValue(), "UTF-8") + "' ");

**break**;

}

%>

><br>Введите цвет для отображения<br>

<INPUT TYPE=*TEXT* NAME=*"color"*

<%

**if** (cookies != **null**)

**for** (**int** i = 0; i < cookies.length; i++)

**if** ("user.color".equals(cookies[i].getName())) {

// Запись значения в поле ввода, если найден Cookie

out.print(" value='" + URLDecoder.decode(cookies[i].getValue(), "UTF-8") + "' ");

**break**;

}

%>

><br><INPUT TYPE=*SUBMIT* value=*"Ввод"*>

</form>

</body>

</html>

*Приложение 1.2.Файл Output.jsp*

<%@ page import=*"java.util.Date"* %>

<%@ page import=*"java.net.URLDecoder"* %>

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Информация о пользователе</title>

</head>

<body>

<%

// Получим цвет текста и имя пользователя

String color = **null**;

String userName = **null**;

Cookie[] cookies = request.getCookies();

**if**(cookies != **null**)

**for**(**int** i = 0; i < cookies.length; i++) {

**if**("user.color".equals(cookies[i].getName())) {

color = cookies[i].getValue();

} **else** **if** ("user.name".equals(cookies[i].getName())) {

userName = cookies[i].getValue();

}

}

// Получим дату последнего обращения и количество запросов

Date lastReq = (Date) request.getSession().getAttribute("lastDate");

Integer reqNum = (Integer) request.getSession().getAttribute("requestNum");

%>

<font color=*"*<%=color%>*"*>

<h1>Информация о пользователе</h1>

<br><b>Имя пользователя: </b>

<%= URLDecoder.decode(userName, "UTF-8")%>

<br><b>Количество запросов: </b>

<%= reqNum%>

<br><b>Дата последнего запроса: </b>

<%= lastReq%>

<br><b>Текущий цвет: </b>

<%= URLDecoder.decode(color, "UTF-8")%>

</font>

</body>

</html>

*Приложение 1.3.Файл Processor.java*

**package** edu.etu.web;

**import** java.io.IOException;

**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.net.URLEncoder;

**import** java.util.Date;

**import** javax.servlet.http.Cookie;

/\*\*

\* Servlet implementation class Processor

\*/

@WebServlet("/Processor")

**public** **class** Processor **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** Processor() {

**super**();

// **TODO** Auto-generated constructor stub

}

**protected** **void** processRequest(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

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

request.setCharacterEncoding("utf-8");

// Получение параметров из строки запроса

String userName = request.getParameter("name");

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

**if** (userName == **null** || color == **null**) {

// Сообщение об ошибке, если сервлет вызван без параметов

response.sendError(HttpServletResponse.***SC\_BAD\_REQUEST***, "Не задан один из параметров");

**return**;

}

// Получим текущее значение количества обращений

Integer reqNum = (Integer) request.getSession().getAttribute("requestNum");

**if** (reqNum == **null**) {

// Если не было обращений, то инициализируем одно обращение

reqNum = 1;

} **else** {

// Иначе увеличим значение обращений на единицу

reqNum++;

}

request.getSession().setAttribute("requestNum", reqNum);

//Получим дату последнего обращения

Date lastDate = **new** Date(System.*currentTimeMillis*());

request.getSession().setAttribute("lastDate", lastDate);

// Сохранение пользователя в Cookie

Cookie cookiesUser = **new** Cookie("user.name", URLEncoder.*encode*(userName, "UTF-8"));

// Сохранение цвета в Cookie

Cookie cookiesColor = **new** Cookie("user.color", URLEncoder.*encode*(color, "UTF-8"));

// Установка времени жизни Cookie в секундах

cookiesUser.setMaxAge(1000);

cookiesColor.setMaxAge(1000);

response.addCookie(cookiesUser);

response.addCookie(cookiesColor);

// Перенаправление на страницу с информацией

response.sendRedirect(response.encodeRedirectURL(request.getContextPath()+"/Output.jsp"));

}

/\*\*

\* **@see** HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

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

// **TODO** Auto-generated method stub

processRequest(request, response);

}

/\*\*

\* **@see** HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

processRequest(request, response);

}

}

**package** edu.etu.web;

**import** java.io.IOException;

**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.net.URLEncoder;

**import** java.util.Date;

**import** javax.servlet.http.Cookie;

/\*\*

\* Servlet implementation class Processor

\*/

@WebServlet("/Processor")

**public** **class** Processor **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** Processor() {

**super**();

// **TODO** Auto-generated constructor stub

}

**protected** **void** processRequest(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

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

request.setCharacterEncoding("utf-8");

// Получение параметров из строки запроса

String userName = request.getParameter("name");

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

**if** (userName == **null** || color == **null**) {

// Сообщение об ошибке, если сервлет вызван без параметов

response.sendError(HttpServletResponse.***SC\_BAD\_REQUEST***, "Не задан один из параметров");

**return**;

}

// Получим текущее значение количества обращений

Integer reqNum = (Integer) request.getSession().getAttribute("requestNum");

**if** (reqNum == **null**) {

// Если не было обращений, то инициализируем одно обращение

reqNum = 1;

} **else** {

// Иначе увеличим значение обращений на единицу

reqNum++;

}

request.getSession().setAttribute("requestNum", reqNum);

//Получим дату последнего обращения

Date lastDate = **new** Date(System.*currentTimeMillis*());

request.getSession().setAttribute("lastDate", lastDate);

// Сохранение пользователя в Cookie

Cookie cookiesUser = **new** Cookie("user.name", URLEncoder.*encode*(userName, "UTF-8"));

// Сохранение цвета в Cookie

Cookie cookiesColor = **new** Cookie("user.color", URLEncoder.*encode*(color, "UTF-8"));

// Установка времени жизни Cookie в секундах

cookiesUser.setMaxAge(1000);

cookiesColor.setMaxAge(1000);

response.addCookie(cookiesUser);

response.addCookie(cookiesColor);

// Перенаправление на страницу с информацией

response.sendRedirect(response.encodeRedirectURL(request.getContextPath()+"/Output.jsp"));

}

/\*\*

\* **@see** HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

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

// **TODO** Auto-generated method stub

processRequest(request, response);

}

/\*\*

\* **@see** HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

processRequest(request, response);

}

}