

ПРЕПОДАВАТЕЛЬ

доцент, к. т. н.		Е. М. Линский
должность , уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №15 СОЗДАНИЕ ПРОГРАММЫ НА ЯЗЫКЕ JAVA

по курсу: ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	5721		А.Е.Ковалева
		подпись, дата	инициалы, фамилия

Задание 15. HttpSession. CSS.

1. Основное задание

Реализовать сервлет для организации доски объявлений. Объявление содержит текст и заголовок (время размещения и имя пользователя).

Для того, чтобы разместить, необходимо ввести логин и пароль (пройти аутентификацию). При старте сервлет загружает базу пользователей и их паролей из текстового файла. Просматривать объявления можно без аутентификации (ввода логина и пароля).

На главной странице находится ссылка “войти в систему” и показывается список объявлений. После входа в систему добавляются ссылки: “выйти из системы”, “добавить объявление”.

1. Для аутентификации необходимо использовать класс HttpSession
2. Для перенаправления пользователя на другую страницу и включения в страницу готов кусков html кода можно воспользоваться классом RequestDispatcher
3. Протестировать, что при очистке cookie в браузере, пользователь выходит из системы.

Между перезагрузками сервера (рестарт Tomcat) список объявлений можно не сохранять.

Дополнительные требования (CSS):

1. Оформление страницы должно быть задано в отдельном файле style.css
2. Должны быть созданы три стиля:
 - a. Заголовок объявления
 - b. Текст объявления
 - c. Ссылка с командой (“войти в систему”, “выйти из системы”, “добавить объявления”)
3. В работе должно быть два варианта оформления (переключение --- переименованием файлов на сервере)

2. Дополнительное задание

Добавить сохранение текущего списка объявлений на диск, чтобы при рестарте сервер список объявлений сохранялся.

Инструкция

1. Основное задание.

Пользователь запускает сервлет. На главной странице ему предлагается авторизоваться. Имена и пароли хранятся в текстовом файле.

Если аутентификация произошла успешно, то пользователь может записать объявление на доску, просмотреть их, выйти из системы, иначе – выходит сообщение о том, что данный пользователь не найден и предоставляется возможность только просмотра стены.

Предусмотрено несколько вариантов оформления.

2. Дополнительное задание.

Авторизовавшись, пользователь на доске объявлений увидит объявления, опубликованные ранее. При перезапуске сервера объявления будут сохранены. Процесс сохранения текущего списка объявлений на диск реализован таким образом, что:

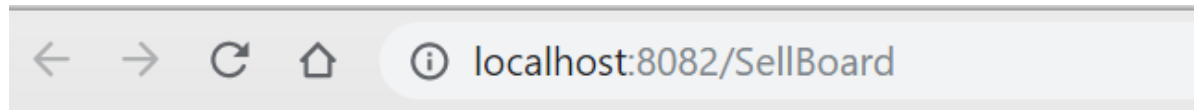
1. Новое объявление записывается на стену и в файл
2. Сервер перезапускается
3. Первым делом на стену выводятся ранее записанные объявления из файла
4. Затем переходим к пункту 1.

Запись объявления в файл происходит путем построчного записывания заголовка, текста, отправителя и id отправителя.

Из файла объявление считывается построчно, затем делим подстроки на слова и добавляем элементы в ArrayList. Из листа считываем элементы и выводим на доску.

Тестирование

1. Основное задание



[Login](#) |

[Logout](#) | [Profile](#) | [Add Ad](#)

Vova 13 Sorry, username or password error!

Name:

Password:

[back to sellboard](#)

Name:

Password:

[Logout](#) | [Profile](#) | [Add Ad](#)

Welcome, Kova

[back to sellboard](#)

bbb

bbb

Owner: Vova

11.12.2019 02.02

1

hallo

Owner: Kova

11.12.2019 02.02



localhost:8082/ProfileSe

[Logout](#) | [Profile](#) | [Add Ad](#)

You have been logged like Kova

[back to sellboard](#)



BaseOfUsers.txt



SellBoard.jav

1

Ak 1311 ;Kova 99 ;

2. Дополнительное задание

adv – Блокнот

Файл Правка Формат Вид Справка

hallo hhhh Vova 3

hhfhff fkfkfkfk Vova 4

← → ↻ 🏠 ⓘ localhost

[Logout](#) | [Profile](#) | [Add Ad](#)

hallo

hhhh

Owner: Vova

11.12.2019 02.06

hhfhff

fkfkfkfk

Owner: Vova

11.12.2019 02.06

1

hi

Owner: Kova

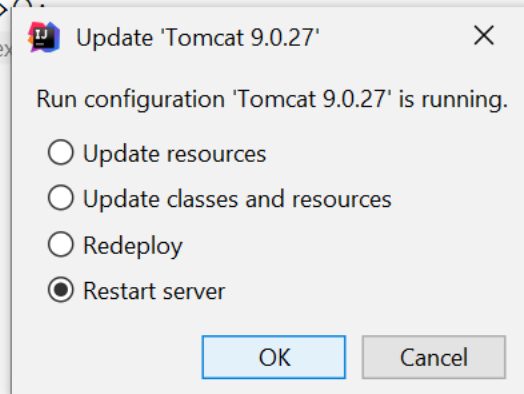
11.12.2019 02.06

```

    .le (r.ready()) {
        int count = 0;
        ArrayList<String> temp1 = new ArrayList<>();
        String[] words = r.readLine().split(regex);
        for (String subStr : words) {
            temp1.add(subStr);
            count++;
        }

        if (count == 4) {
            System.out.println(count);
            String header = temp1.get(0);
            String text = temp1.get(1);
            String owner = temp1.get(2);
            String id0 = temp1.get(3);
            int id = Integer.parseInt(id0);
            Advertisement adv3 = new Advertisement(header, text, owner, id);
            wall.add(adv3);
            id++;
        }
    }
}

```



11.12.2019 02:24

hhfhff

fkfkfkfk

Owner: Vova

11.12.2019 02:24

1

hi

Owner: Kova

11.12.2019 02:24

Da

45ydfhfh

Owner: Kova

11.12.2019 02:24