Lab 2

Задание:

Дан HTML документ. Требуется удалить из него все HTML тэги, получив чистый текст. Полученный текст разбить на слова. Удалить и списка всех слов повторения.  
Отсортировать все слова по алфавиту и вывести на экран. Слова хранить в коллекции.

Показать работу программы на паре.

В ходе написания кода везде были оставлены комментарии для дополнительной ясности.

**Алгоритм:  
1) Открытие и считывание файла**

-Открываю и считываю файл, отлавливая ошибки связанные с доступом к файлу

**2)Превращаю html код в чистый текст**

**-**Для обработки html кода использую библиотекуJsoup

-Для устранения знаков пунктуации, спец. символов и лишних пробелов использую replaceAll с соответствующими регулярными выраженими

**3)Создаю TreeSet – коллекцию**

- Данная коллекция удаляет дублирующиеся элементы и сортирует их в алфавитном порядке по умолчанию. Для обработки кирилицы – создаю объект класса Collator, создающий новую языковую локаль(Руссую)

- Передаю в нее массив слов (массив из строки, к которой применяю метод .split(). В качестве сепаратора указываю пробел.)

-Чтоб не перебирать циклом все слова и не добавлять по 1 – использую Collecions.assAll().

**Как результат – программа отрабатывает корректно и на выходе получаю список всех слов, написанных в html-документе, отсортированных в алфавитном порядке.**

Использованные модули и библиотеки:

import org.jsoup.Jsoup;

import java.text.Collator;  
import java.util.Set;  
import java.util.TreeSet;  
import java.util.Collections;  
import java.util.Locale;  
import java.nio.file.Files;  
import java.nio.file.Paths;  
import java.io.IOException;

**Листинг кода:**

import org.jsoup.Jsoup;  
import java.text.Collator;  
import java.util.Set;  
import java.util.TreeSet;  
import java.util.Collections;  
import java.util.Locale;  
import java.nio.file.Files;  
import java.nio.file.Paths;  
import java.io.IOException;  
  
public class Main {  
 public static void main(String[] args) {  
  
 String filePath = "src/MainPackage/index.html";  
 String html = "";  
  
 // Открываю файл и считываю данные из него в строку  
 try {  
 html = Files.*readString*(Paths.*get*(filePath));  
 } catch (IOException e) {  
 System.*err*.println("Ошибка при чтении файла: " + e.getMessage());  
 return; // Завершение программы при ошибке  
 }  
  
 // Преобразование HTML в чистый текст  
 // С помощью relaceAll убираю все знаки пункутуации & спец. символы & пробелы.  
 String text = Jsoup.*parse*(html).text().trim()  
 .replaceAll("\\p{Punct}", "")  
 .replaceAll("\\s+", " ");  
  
  
 // TreeSet - данная коллекция не содержит дубликатов и сортирует элементы в алфавитном пордке  
 // Класс Colator для создания руссой локали и корректной сортирвки слов написанных кирилицей  
 Set<String> wordSet = new TreeSet<>(Collator.*getInstance*(new Locale("ru", "RU"))) ;  
  
  
 // Добавляю в TreeSet массив из слов, на которые разбиваю строку c текстом  
 Collections.*addAll*(wordSet, text.split(" "));  
  
 System.*out*.println(wordSet);  
  
 }  
}

Результат (коллекции):

