Звіт

Лабораторна робота №3

УТИЛІТАРНІ КЛАСИ. ОБРОБКА МАСИВІВ І РЯДКІВ

**Мета.** Розробка власних утилітарних класів. Набуття навичок вирішення прикладних задач з використанням масивів і рядків.

**Розробник:**

- Чугунов Вадим Юрійович

- КІТ-119а;

- Варіант №24.

**Вимоги:**

1. Розробити та продемонструвати консольну програму мовою Java в середовищі Eclipse для вирішення прикладної задачі за номером, що відповідає збільшеному на одиницю залишку від ділення на 15 зменшеного на одиницю номера студента в журналі групи.
2. При вирішенні прикладних задач використовувати латинку.
3. Продемонструвати використання об’єктів класу StringBuilder або StringBuffer.
4. Застосувати функціональну (процедурну) декомпозицію - розробити власні утилітарні класи (особливий випадок допоміжного класу, див. Helper Class) та для обробки даних використовувати відповідні статичні методи.
5. Забороняється використовувати засоби обробки регулярних виразів: класи пакету java.util.regex (Pattern, Matcher та ін.), а також відповідні методи класу String (matches, replace, replaceFirst, replaceAll, split).

ЗАВДАННЯ ДО РОБОТИ

Ввести текст. Знайти та вивести, скільки разів повторюється в тексті кожне слово. Результат вивести у вигляді таблиці.

ОПИС ПРОГРАМИ

**Опис змінних**

String text; // текст, у якому проводиться пошук

String[] words; // масив слів, у якому проводиться розрахунок

HashMap<String, Integer> wordToCount; //хеш таблиця для слів

Scanner in; // змінна для введення даних

**Ієрархія та структура класів**

**class** Main – точка входу в програму.

**class** Helper – утилітарний клас, який виконує завдання

ТЕКСТ ПРОГРАМИ

Текст **class** Main

package ua.oop.khpi.chugunov03;

import java.util.Scanner;

import java.util.HashMap;

public class Main {

public static void main(String[] args) {

String text = Helper.printQuest();

String[] words = Helper.SplitString(text);

HashMap<String, Integer> wordToCount = new HashMap<>();

for (String word : words) {

if (!wordToCount.containsKey(word)) {

wordToCount.put(word, 0);

}

wordToCount.put(word, wordToCount.get(word) + 1);

}

for (String word : wordToCount.keySet()) {

System.out.println(word + " " + wordToCount.get(word));

}

}

}

Текст **class** Helper

package ua.oop.khpi.chugunov03;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Helper {

public static String printQuest()

{

System.out.println("Введите текст:");

Scanner in = new Scanner(System.in);

return in.nextLine();

}

public static String[] SplitString(String text) {

List<String> array = new ArrayList<>();

StringBuilder builder = new StringBuilder();

for(char symbol : text.toCharArray()) {

if((int)symbol == 32 | (int)symbol == 44 | (int)symbol == 46) {

array.add(builder.toString());

builder = new StringBuilder();

continue;

}

builder.append(symbol);

}

if(builder.length() != 0) {

array.add(builder.toString());

}

for (int i = 0; i < array.size(); i++) {

if(array.get(i).length() == 0) {

array.remove(i);

}

}

String[] output = new String[array.size()];

for (int i = 0; i < array.size(); i++) {

output[i] = array.get(i);

}

return output;

}

}

ВАРІАНТИ ВИКОРИСТАННЯ

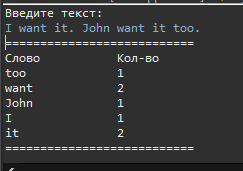


Рис. 1 – Результат роботи програми

Дану програму можна використовувати для пошуку потрібного нам слова в тексті, та знайти скільки разів воно повторюється.

ВИСНОВОК

Під час виконання лабораторної роботи було створено програму, яка має в собі утилітарний клас Helper з використанням об’єкту класу StringBuilder для вирішення індивідуального завдання.