Тема: Утилітарні класи. Обробка масивів і рядків

Мета:

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

**1 ВИМОГИ**

**1.1 Розробник**

Інформація про розробника:

- Гряник Георгій Володимирович

- КІТ-119Д;

- 6 варіант.

**1.2 Загальне завдання**

* 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.
* При вирішенні прикладних задач використовувати [латинку](https://uk.wikipedia.org/wiki/%D0%9B%D0%B0%D1%82%D0%B8%D0%BD%D1%81%D1%8C%D0%BA%D0%B0_%D0%B0%D0%B1%D0%B5%D1%82%D0%BA%D0%B0).
* Продемонструвати використання об'єктів класу [StringBuilder](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuilder.html) або [StringBuffer](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuffer.html).
* Застосувати функціональну (процедурну) декомпозицію - розробити власні утилітарні класи (особливий випадок допоміжного класу, див. [Helper Class](https://en.wikipedia.org/wiki/Helper_class)) та для обробки даних використовувати відповідні [статичні](https://docs.oracle.com/javase/tutorial/java/javaOO/classvars.html) методи.
* Забороняється використовувати засоби обробки регулярних виразів: класи пакету [java.util.regex](https://docs.oracle.com/javase/8/docs/api/java/util/regex/package-summary.html) ([Pattern](https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html), [Matcher](https://docs.oracle.com/javase/8/docs/api/java/util/regex/Matcher.html) та ін.), а також відповідні методи класу [String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) (matches, replace, replaceFirst, replaceAll, split).

1.3 Задача

Ввести текст. З тексту видалити всі символи, крім пропусків, які не є буквами. Пропуски, що повторюються, замінити на одиночні. Між послідовностями літер, де знаходяться розділові знаки, залишити хоча б один пропуск ("a,b,c" -> "a, b, c"). Вивести початковий текст та результат.

2 ОПИС ПРОГРАМИ

2.1 Засоби ООПВикористання стороннього класу суть якого полягає виконання не основних завдань програми .  
2.2 Ієрархія та структура класівКлас “Helper ” виконує роль допоміжного класу який виконує неосновні завдання наприклад : виведення результату або перевірка символів на відповідність

2.3 Важливі фрагменти програми **public** **class** Helper

{

//static functions

**public** **static** **boolean** СonditionalСheck(**char** text)/// перевіряє символ на умову

{

**return** ( text>64&&text<91)||( text>96&&text<=123)||text==32||text==44;

}

**public** **static** **void** PrintLine(StringBuilder text)///вивод тексту на консоль

{

System.***out***.println("\n\nВаш текст(дублювани): "+text);

}

**public** **static** **void** PrintNewLine(StringBuilder text)///вивод тексту на консоль із надписом відредагований

{

System.***out***.println("\n\nВаш текст(Відредагований): "+text);

}

**public** **static** StringBuilder Task6 (StringBuilder s)//////редагування тексту

{

String temp=**new** String();

**boolean** spaise=**false**;

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

{

**if** (Helper.*СonditionalСheck*(s.charAt(i)))

{

**if** (s.charAt(i)==32) spaise=**true**;//перевірка на пробіл

**else**

{

**if**(s.charAt(i)==44)///перевірка на кому

{

temp=temp+s.charAt(i) ;

spaise=**true**;

}

**else** **if**(spaise==**true**)///написати пробіл

{ temp =temp+" "+s.charAt(i);

spaise=**false**;

}

**else** temp =temp+s.charAt(i);///написати символ

}

}

} s=**new** StringBuilder(temp);

**return** s;

}

}

**3 ВАРІАНТИ ВИКОРИСТАННЯ**

Програма пропонує користувачеві ввести текст після чого в циклі програма посимвольно перевіряє текст. У разі якщо це не латинський символ чи пробіл або кома то символ пропускається. Далі програма проводить додаткове редагування тексту , тобто прибирає зайві пробіли. Відредаговану частина записується у тимчасову змінну. По закінченню редагування результат повертається в вхідну змінну.

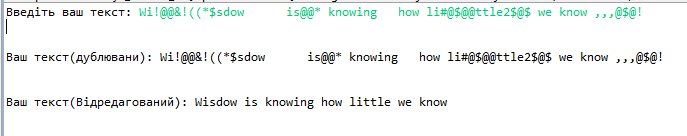


Рисунок 1 – результат редагування тексту

ВИСНОВКИ

Під час виконання лабораторної роботи було набуто навички розробка розробки власного класу, а особливо Help class та вирішення прикладних задач з використанням рядків