

## Звіт

### Лабораторна робота 4. Інтерактивні консольні програми для платформи Java SE

**Мета роботи:** реалізація діалогового режиму роботи з користувачем в консольних програмах мовою Java.

## ВИМОГИ

- 1) Використовуючи програму рішення завдання лабораторної роботи №3, відповідно до прикладної задачі забезпечити обробку команд користувача у вигляді текстового меню:
  - введення даних;
  - перегляд даних;
  - виконання обчислень;
  - відображення результату;
  - завершення програми і т.д.
- 2) Забезпечити обробку параметрів командного рядка для визначення режиму роботи програми:
  - параметр "-h" чи "-help": відображається інформація про автора програми, призначення (індивідуальне завдання), детальний опис режимів роботи (пунктів меню та параметрів командного рядка);
  - параметр "-d" чи "-debug": в процесі роботи програми відображаються додаткові дані, що полегшують налагодження та перевірку працездатності програми: діагностичні повідомлення, проміжні значення змінних, значення тимчасових змінних та ін.

**1.1. Розробник:** Капелька Ярослав Іванович, КІТ119-а, варіант №9.

## 2. ОПИС ПРОГРАМИ

**2.1. Засоби ООП:** клас, метод класу, поле класу.

**2.2. Ієрархія та структура класів:** один публічний клас Main з двома методами та клас HelperClass з класом util та одним методом.

**2.3. Важливі фрагменти програми:**

```
public class HelpMenu
{
    public static class util
    {
        static StringBuilder builder = new StringBuilder();
    }
}
```

```

static java.util.Scanner in = new java.util.Scanner(System.in);
static java.util.HashSet<String> myHashSet = new
java.util.HashSet<String>();
static String getString()
{
    System.out.println("Введите текст: ");
    String res = in.nextLine();
    return res;
}
static void replacement(String text)
{
    text += " ";
    for (int i = 0; i < text.length(); i++)
    {
        if (text.charAt(i) != ' ')
        {
            builder.append(text.charAt(i));
        }
        else
        {
            if (!builder.toString().equals(""))
            {
                myHashSet.add(builder.toString());
                builder.setLength(0);
            }
        }
    }
}

static int substringCount(String s, String pattern, boolean debug)
{
    int result = 0;
    if(debug)
    {
        System.out.println("Метод substringCount()");
        System.out.println("Значение переменной s: " + s);
        System.out.println("Значение переменной pattern: " +
pattern);
        System.out.println("Промежуточное значение переменной
result: " + result);
        System.out.println("-----");
    }
    else
    {
        System.out.println("Нет данных про переменные.");
    }
    s += " ";
    pattern += " ";
    for (int i = 0; i < s.length(); i++)
    {
        if(i + pattern.length() <= s.length())
        {
            if (s.substring(i, i +
pattern.length()).equals(pattern))
            {
                result++;
                i += pattern.length()-1;
            }
        }
    }
    return result;
}

```

```

    }
    public static void task(boolean debug)
    {
        String data = getString();
        replacement(data);
        int res;
        if(debug)
        {
            System.out.println("Метод task()");
            System.out.println("Значение переменной data: " + data);
        }
        else
        {
            System.out.println("Нет данных про переменные.");
        }
        for(String s : myHashSet)
        {
            res = substringCount(data,s,true);
            System.out.println("Повторения Слово ");
            System.out.println(" " + res + " " + s + " ");
        }
    }
    public static void task(String data,boolean debug)
    {
        replacement(data);
        int res;
        if(debug)
        {
            System.out.println("Метод task()");
            System.out.println("Значение переменной data: " + data);
            System.out.println("-----");
        }
        else
        {
            System.out.println("Нет данных про переменные.");
        }
        for(String s : myHashSet)
        {
            res = substringCount(data,s,true);
            System.out.println("Повторения Слово ");
            System.out.println(" " + res + " " + s + " ");
        }
    }
}
private static boolean debug = false;
public static void debugMod(boolean debugMod)
{
    debug = debugMod;
    System.out.printf("Debug mod ");
    if (debug)
    {
        System.out.printf("on\n");
    }
    else
    {
        System.out.printf("off\n");
    }
}
}

```

## Результат виконання програми:

---

```
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта,который хотите выполнить:
1
Введите текст:
КУ КУ Привет Привет Приветики Приветики
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта,который хотите выполнить:
2
Вот введенные данные:
КУ КУ Привет Привет Приветики Приветики
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта,который хотите выполнить:
3
Метод task()
Значение переменной data: КУ КУ Привет Привет Приветики Приветики
-----
Метод substringCount()
Значение переменной s: КУ КУ Привет Привет Приветики Приветики
Значение переменной pattern: Приветики
Промежуточное значение переменной result: 0
-----
Повторения      Слово
      2      Приветики
Метод substringCount()
Значение переменной s: КУ КУ Привет Привет Приветики Приветики
Значение переменной pattern: КУ
Промежуточное значение переменной result: 0
-----
Повторения      Слово
      2      КУ
Метод substringCount()
Значение переменной s: КУ КУ Привет Привет Приветики Приветики
Значение переменной pattern: Привет
Промежуточное значение переменной result: 0
-----
Повторения      Слово
      2      Привет
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта,который хотите выполнить:
4
|
```

---

## Результат виконання консолі:

```
Администратор: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.572]
(c) Корпорация Майкрософт (Microsoft Corporation), 2020. Все права защищены.

C:\Users\Admin>cd eclipse-workspace\Kapelka-Yaroslav

C:\Users\Admin\workspace\Kapelka-Yaroslav>javac -sourcepath ./src -d bin src/ua/khpi/ooop/Kapelka04/Main04.java

C:\Users\Admin\workspace\Kapelka-Yaroslav>java -classpath ./bin ua.khpi.ooop.Kapelka04.Main04 -h -d
Автор: Капелька Ярослав
Ввести текст. Найти та вывести, скільки разів повториться в тексті кожне слово. Результат вывести у вигляді таблиці.
Параметр "-h" чи "-help": відображається інформація про автора програми, призначення (індивідуальне завдання), детальний опис режимів роботи (пунктів меню та параметрів командного рядка);
Параметр "-d" чи "-debug": в процесі роботи програми відображаються додаткові дані, що полегшують налагодження та перевірку працездатності програми: діагностичні повідомлення, проміжки значення змінних, значення
тимчасових змінних та ін.
Debug mod on
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта, который хотите выполнить:
1
Введите текст:
Hi Hi Hello Hello Doggy Doggy
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта, который хотите выполнить:
2
Вот введенные данные:
Hi Hi Hello Hello Doggy Doggy
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта, который хотите выполнить:
3
Метод task()
Значение переменной data: Hi Hi Hello Hello Doggy Doggy
-----
Метод substringCount()
Значение переменной s: Hi Hi Hello Hello Doggy Doggy
Значение переменной pattern: Hi
Промежуточное значение переменной result: 0
-----
Повторения    Слово
      2      Hi
Метод substringCount()
Значение переменной s: Hi Hi Hello Hello Doggy Doggy
Значение переменной pattern: Hello
Промежуточное значение переменной result: 0
-----
Повторения    Слово
      2      Hello
Метод substringCount()
Значение переменной s: Hi Hi Hello Hello Doggy Doggy
Значение переменной pattern: Doggy
Промежуточное значение переменной result: 0
-----
Повторения    Слово
      2      Doggy
1.Ввести текст
2.Посмотреть введенные данные
3.Посмотреть результат
4.Закончить работу
Выберите номер пункта, который хотите выполнить:
4

C:\Users\Admin\workspace\Kapelka-Yaroslav>
```

## Висновки

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи з інтерактивними консольними програмами для платформи Java SE.

Програма протестована, виконується без помилок.