Звіт

Автор: Васильченко С., 1.КІТ101.8б

Дата: 18.09.2019

Лабораторна робота №4

ІНТЕРАКТИВНІ КОНСОЛЬНІ ПРОГРАМИ ДЛЯ ПЛАТФОРМИ JAVA SE

***Мета:***

* Реалізація діалогового режиму роботи з користувачем в консольних програмах мовою Java.

**Вимоги:**

1. Використовуючи програму рішення завдання [лабораторної роботи №3](https://oop-khpi.github.io/#task_03), відповідно до [прикладної задачі](https://oop-khpi.github.io/#task_03_app) забезпечити обробку команд користувача у вигляді текстового [меню](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D0%BD%D1%8E_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)#.D0.9C.D0.B5.D0.BD.D1.8E_.D0.B2_.D0.B8.D0.BD.D1.82.D0.B5.D1.80.D1.84.D0.B5.D0.B9.D1.81.D0.B5_.D0.BA.D0.BE.D0.BC.D0.B0.D0.BD.D0.B4.D0.BD.D0.BE.D0.B9_.D1.81.D1.82.D1.80.D0.BE.D0.BA.D0.B8):
   * введення даних;
   * перегляд даних;
   * виконання обчислень;
   * відображення результату;
   * завершення програми і т.д.
2. Забезпечити обробку параметрів [командного рядка](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D1%80%D1%8F%D0%B4%D0%BA%D0%B0) 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.

ЗАВДАННЯ ДО РОБОТИ

Ввести декілька рядків. Упорядкувати, а потім вивести рядки за алфавітом (перший пріоритет) та в порядку зростання їх довжини (другий пріоритет).

ОПИС ПРОГРАМИ

***2.1 Опис змінних:***

String[] str1 //масив строк

ArgsCMD Arguments //обработка аргументів командного рядка

Scanner scan //сканер водимих даних

boolean loop //для запуску цикла

***2.2 Ієрархія та структура класів.***

public class Lab03 – головний клас. Містить метод main(точку входу у програму).

В класі використовується додаткова функція private static void ShowStringMass(String[] str), яка відповідає за виведення строки.

public class StringLengthSort implements Comparator<String> - клас, який використовується для порівняння двох рядків, для подальшого сортування за довжиною.

class Functions - допоміжний клас в якому реалізуються функції

class ArgsCMD – класс-обробник командної строки

**ТЕКСТ ПРОГРАМИ**

File Lab04.java:

package Lab4;

import java.util.Arrays;

import java.util.Scanner;

import static Lab4.Functions.\*;

class Lab04 {

public static void main(String[] args) {

ArgsCMD Arguments = new ArgsCMD(args);

if (!Arguments.isEmpty()) {

Arguments.arguments();

}

Scanner scan = new Scanner(System.in);

boolean loop = true;

int choose;

String[] str1 = null;

while (loop) {

ChooseMenu();

choose = scan.nextInt();

switch (choose) {

case 1:

if (str1 != null){

if (ArgsCMD.DBG)

System.out.println("You choose 1. But string has been already created");

System.out.println("Error! String has been already created.");

break;

}

else {

if(ArgsCMD.DBG)

System.out.println("You choose 1. Creating string");

str1 = InitializeString();

break;

}

case 2:

if (str1 == null){

if(ArgsCMD.DBG) {

System.out.println("You choose 2. But your string is empty");

}

System.out.println("Error! String is empty!");

break;

}

else {

if (ArgsCMD.DBG) {

System.out.println("You choose 2. Showing your string");

}

System.out.println("Your string:");

ShowStringMass(str1);

break;

}

case 3:

if (str1 == null){

if (ArgsCMD.DBG) {

System.out.println("You choose 3. But your string is empty");

}

System.out.println("Error! String is empty!");

break;

}

else {

if (ArgsCMD.DBG)

System.out.println("You choose 3. Sorting now");

System.out.println("\nSorting alphabetically:");

Arrays.sort(str1);

System.out.println("Successfully!");

break;

}

case 4:

if (str1 == null){

if (ArgsCMD.DBG)

System.out.println("You choose 4. But your string is empty");

System.out.println("Error! String is empty!");

break;

}

else {

if (ArgsCMD.DBG)

System.out.println("You choose 4. Sorting now");

System.out.println("\nSort by length: ");

StringLengthSort sort1 = new StringLengthSort();

Arrays.sort(str1, sort1);

System.out.println("Successfully!");

break;

}

case 5:

if (str1 == null){

if(ArgsCMD.DBG)

System.out.println("You choose 5. But your string is empty");

System.out.println("Error! String is empty!");

break;

}

else {

if (ArgsCMD.DBG)

System.out.println("You choose 5. Deleting string now");

str1 = null;

System.out.println("Deleting string successfully!");

break;

}

case 0:

if (ArgsCMD.DBG)

System.out.println("You choose 0. You the end of Work");

System.out.println("\n\nThanks for working!");

loop = false;

break;

default:

System.out.println("Error. Mistake number!");

break;

}

}

}

}

Functions.java:

package Lab4;

import java.util.Scanner;

class Functions {

static void ChooseMenu() {

System.out.println("\n\nChoose the right variant:");

System.out.println("1. Create string");

System.out.println("2. Show string");

System.out.println("3. Sorting alphabetically");

System.out.println("4. Sort by length");

System.out.println("5. Delete string");

System.out.println("0. The End");

System.out.print("Write:");

}

static void ShowStringMass(String[] str){

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

System.out.print(str[i]+" ");

}

static String[] InitializeString(){

Scanner scan = new Scanner(System.in);

System.out.print("Enter number of words: ");

int col = scan.nextInt();

String[] str1 = new String[col];

System.out.printf("Enter %d words:%n", col);

for (int i = 0; i < str1.length; i++)

str1[i] = scan.next();

return str1;

}

}

StringLengthSort.java:

package Lab4;

import java.util.Comparator;

public class StringLengthSort implements Comparator<String> {

@Override

public int compare(String o1, String o2) {

if (o1.length() > o2.length()) {

return 1;

} else {

if (o1.length() < o2.length())

return -1;

else

return 0;

}

}

}

ArgsCMD.java:

package Lab4;

class ArgsCMD {

static boolean DBG = false;

private String[] args;

ArgsCMD(String[] args) {

this.args = args;

}

boolean isEmpty(){

return args.length == 0;

}

void arguments(){

for (String i: args){

if (i.equals("-d") || i.equals("-debug"))

ArgsCMD.debug();

else if(i.equals("-h") || i.equals("-help"))

ArgsCMD.help();

else

System.out.println("Error. Use only this arguments: \"-d\", \"-debug\", \"-h\", \"-help\"");

}

}

private static void help() {

System.out.println("Author: Vasilchenko Sergey Sergeevich");

System.out.println("Individual tasks:");

System.out.println("Using the program of solving the problem of laboratory work #3, \naccording to the application task provide the processing of user commands in the form of a text menu:\n" +

"data entry;\n" +

"view data;\n" +

"view data;\n" +

"displaying the result;\n" +

"program completion, etc.");

}

private static void debug() {

DBG = true;

System.out.println("Debug");

}

}

**ВАРІАНТИ ВИКОРИСТАННЯ**

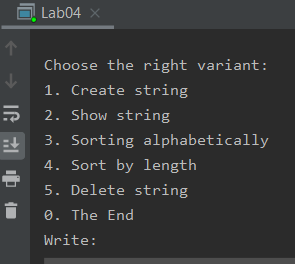
****

Рис. 4.1 – Результат роботи програми

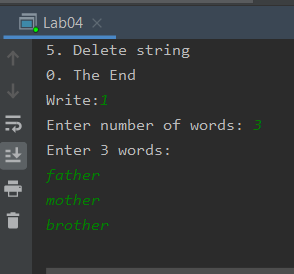
****

Рис. 4.2 – Результат роботи програми

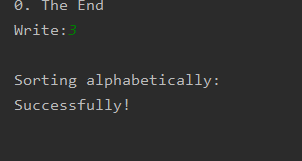
****

Рис. 4.3 – Результат роботи програми

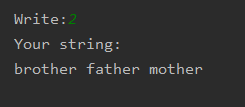
****

Рис. 4.4 – Результат роботи програми

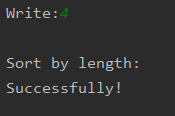
****

Рис. 4.5 – Результат роботи програми

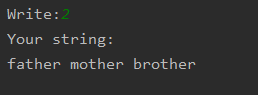
****

Рис. 4.6 – Результат роботи програми

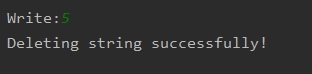
****

Рис. 4.7 – Результат роботи програми

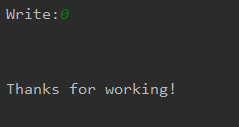
****

Рис. 4.8 – Результат роботи програми

Програму можна використовувати задля форматованої роботи з рядками. А саме сортування за алфавітом та за довжиною слова. Використовувати інтерактивне меню задля простого доступу до елементів програми.

**ВИСНОВКИ**

Завдяки даній лабораторній роботі було закріплено знання щодо розробки власних утилітарних класів. Було набуто навички, щодо використання рядків та масивів. Було реалізовано діалоговий режим роботи з користувачем консольних програм. Завдання виконано. Програма працює успішно!