## Звіт

# Лабораторна 4. Інтерактивні консольні програми для платформи Java SE

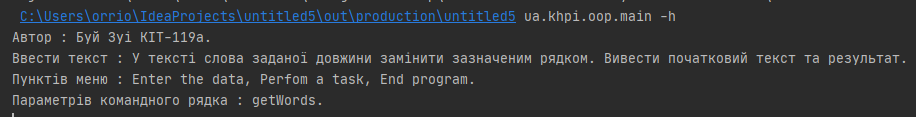
## Мета

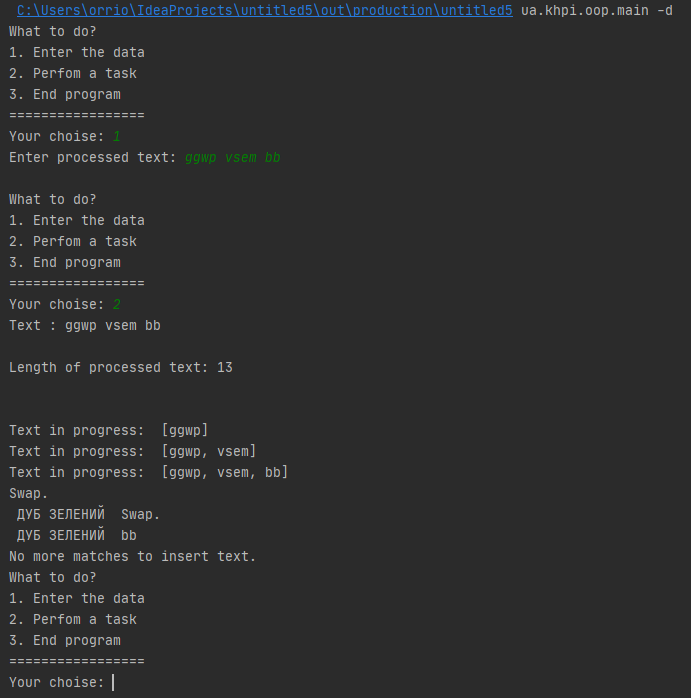
Реалізація діалогового режиму роботи з користувачем в консольних програмах мовою Java.

## Вимоги

1. Використовуючи програму рішення завдання лабораторної роботи №3, відповідно до [прикладної задачі](https://oop-khpi.gitlab.io/%D0%B7%D0%B0%D0%B2%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F/task03/" \l "_4) 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 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. **Розробник**: Буй Зуі , КІТ119-а, варіант №26.
3. **ОПИС ПРОГРАМИ**
   1. **Засоби ООП**: клас, метод класу, поле класу.
   2. **Важливі фрагменти програми:**
4. package ua.khpi.oop;  
     
   import java.util.Scanner;  
     
   public class HelperClass {  
    public void OutputProcessedText(StringBuilder line)  
    {  
    System.*out*.println(line);  
    }  
     
    public String inputText()  
    {Scanner scan = new Scanner(System.*in*);  
     
    String inputText = scan.nextLine();  
     
    return inputText;  
    }  
     
    public void HelpInfo()  
    {  
    System.*out*.println("Aвтор : Буй Зуі КІТ-119а.");  
    System.*out*.println("Ввести текст : У тексті слова заданої довжини замінити зазначеним рядком. Вивести початковий текст та результат.");  
    System.*out*.println("Пунктів меню : Enter the data, Perfom a task, End program.");  
    System.*out*.println("Параметрів командного рядка : getWords.");  
    System.*out*.println();  
    }  
   }
5. package ua.khpi.oop;  
     
     
   import java.text.BreakIterator;  
   import java.util.ArrayList;  
   import java.util.Scanner;  
   import java.lang.StringBuilder;  
     
   import java.util.\*;  
     
   public class main {  
    public static int *k* = 4;  
    public static String *sentence* = " ДУБ ЗЕЛЕНИЙ " ;  
    public static HelperClass *help* = new HelperClass();  
    public static boolean *debugFlag* = false;  
    public static StringBuilder *text* = new StringBuilder();  
    public static String *newtext* = new String();  
    public static StringBuilder *insertedText* = new StringBuilder();  
     
     
    public static void main(String[] args) {  
     
    for(String bbc: args)  
    {  
    if(bbc.equals("-h") || bbc.equals("-help")) *help*.HelpInfo();  
    if(bbc.equals("-d") || bbc.equals("-debug")) *debugFlag* = true;  
    }  
     
    main.*menu*();  
     
    }  
     
    static void menu()  
    {  
    Scanner scan = new Scanner(System.*in*);  
    String choise;  
    boolean stop = false;  
     
    while(stop != true)  
    {  
    System.*out*.println("What to do?");  
    System.*out*.println("1. Enter the data");  
    System.*out*.println("2. Perfom a task");  
    System.*out*.println("3. End program");  
    System.*out*.println("=================");  
    System.*out*.print("Your choise: ");  
     
    choise = scan.nextLine();  
     
    switch(choise) {  
    case "1":  
     
    System.*out*.print("Enter processed text: ");  
    *text* = new StringBuilder(*help*.inputText());  
     
    System.*out*.println(" ");  
    break;  
     
    case "2":  
     
     
    if (*text*.toString().isEmpty() )  
    System.*out*.println("Error. Enter the data before.\n");  
     
    else  
    *newtext* = *text*.toString();  
    *getWords*( *newtext*, *k*, *sentence*);  
     
    break;  
     
    case "3":  
    stop = true;  
    break;  
     
    default:  
    System.*out*.println("Error. Wrong command.\n");  
    break;  
    }  
    }  
     
    scan.close();  
    }  
     
    static void getWords(String text,int k, String sentence) {  
    if(*debugFlag*)  
    {  
    System.*out*.print("Text : " + text + "\n\n");  
    System.*out*.println("Length of processed text: " + text.length());  
    System.*out*.print("\n\n");  
     
    }  
     
    List<String> words = new ArrayList<String>();  
    BreakIterator breakIterator = BreakIterator.*getWordInstance*();  
    breakIterator.setText(text);  
    int lastIndex = breakIterator.first();  
    while (BreakIterator.*DONE* != lastIndex) {  
     
    int firstIndex = lastIndex;  
    lastIndex = breakIterator.next();  
    if (lastIndex != BreakIterator.*DONE* && Character.*isLetterOrDigit*(text.charAt(firstIndex))) {  
    words.add(text.substring(firstIndex, lastIndex));  
    if(*debugFlag*){  
    System.*out*.println("Text in progress: "+ words);}  
    }  
     
    }  
    for(int i=0; i< words.size();i++){  
     
    if(words.get(i).length() == k){  
    if(*debugFlag*) System.*out*.println("Swap.");  
    words.set(i,sentence);  
     
     
    }  
     
    System.*out*.print(words.get(i)+" ");  
     
     
    }  
    System.*out*.println(" ");  
    if(*debugFlag*) System.*out*.println("No more matches to insert text.");  
    }  
     
     
    }

**3.Результат**





**Висновки**

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи с серіалізацією та десеріалізацією об’єктів .

Програма протестована, виконується без помилок.