Тема: Інтерактивні консольні програми для платформи Java SE∗

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

**1 ВИМОГИ**

**1.1 Розробник**

Інформація про розробника:

- Резніченко Дмитро Володимирович

- КІТ-3 19;

- 6 варіант.

**1.2 Загальне завдання**

1. Використовуючи програму рішення завдання [лабораторної роботи №3](https://oop-khpi.gitlab.io/%D0%B7%D0%B0%D0%B2%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F/task03/), 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/#_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](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.

1.3 Задача

Виконати загальне завдання

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** **static** **void** main(**final** String[] args) {

*Menu*(args[0]);/// виклик функцї меню

}

**public** **static** **void** Menu(String operating\_mode) ///функціє проводить координування по можливостям програм

{

**if** (operating\_mode.equals("-debug")||operating\_mode.equals("-d") )

{ Helper.*debug*=**true**; }

**else** **if** (operating\_mode.equals("-help")||operating\_mode.equals("-h") )

{ Helper.*help*=**true**;}

**if** (Helper.*debug*==**true**)System.***out***.println("\n("+Helper.*date*.toString()+") Викликана фукція 'Menu' )");

Helper.*Help*();

**while**(**true**)///нескінченний цикл який дозволяє працювати програмі

{

**if** (Helper.*debug*==**true**)System.***out***.println("\n("+Helper.*date*.toString()+") фукція 'Menu' )");

**if** (Helper.*debug*==**true**)System.***out***.println("\n(("+Helper.*date*.toString()+") text-"+*text*+"; res-"+*res*+")");

System.***out***.print("\nВведіть команду: ");

String s=*in*.nextLine();

**switch**(s)///пошук введеної команди

{

**case** "1": Helper.*PrintLine*(*text*);

**break**;

**case** "2": Helper.*PrintLine*(*res*);

**break**;

**case** "3":*text*=Helper.*Replacement*(*text*);

**break**;

**case** "4": *res*=Helper.*Task6*(*text*);

**break**;

**case** "5" :Helper.*Help*();

**break**;

// case "cls" :

//break;

**case** "6" :

*text*="Eror";

*res*="Eror";

**break**;

**case** "7":

**if** (Helper.*debug*==**true**)System.***out***.println("\n(("+Helper.*date*.toString()+") Програма завершує роботу)");

**return**;

**default** : Helper.*Help*();

**break**;

}

}

}

}

**3 ВАРІАНТИ ВИКОРИСТАННЯ**

На початку програма потребує флагу. Даний флаг вмикає певний режим роботи. Режимів роботи два. –h(-help)- режим у якому відображається завдання та автор програми,-d(--debug) – відображає додаткову інформацію для отладки та інформацію про стан роботи. Після введення флаг обробляється та вмикається відповідний режим. Далі звичайне користування програмою в залежності від режиму

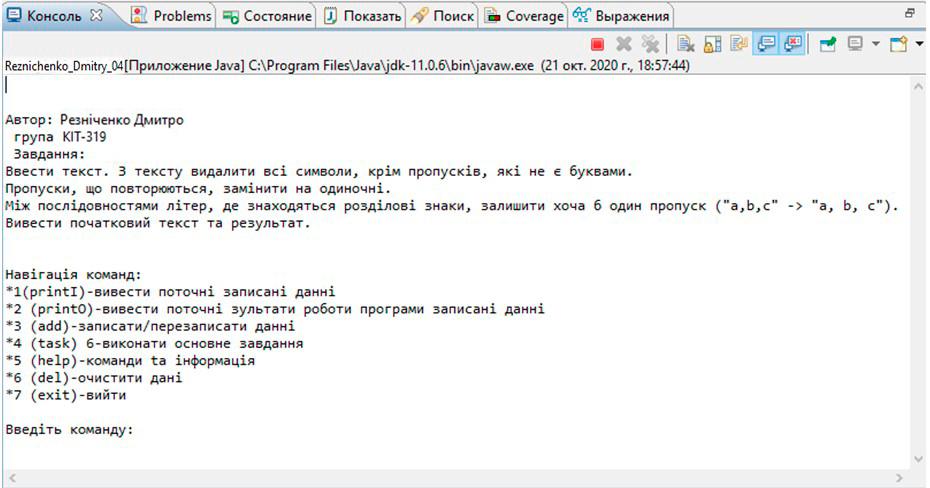


Рисунок 1 – режим роботи програми –h

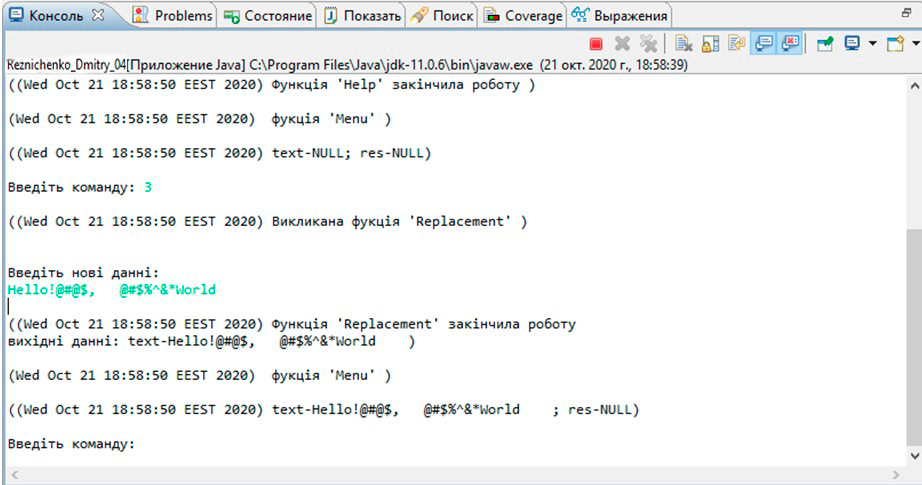


Рисунок 2 – режим роботи програми –d

ВИСНОВКИ

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