Міністерство освіти і науки України

Національний технічний університет

«Харківський політехнічний інститут»

Кафедра «Обчислювальна техніка та програмування»

ЗВІТ

Про виконання лабораторної роботи №6

“Серіалізація/десеріалізація об’єктів.

Бібліотека класів користувача”

Виконавець:

студент гр. КІТ-120В

Олексієнко Микита

Харків 2021

Лабораторна робота №6.

**Серіалізація/десеріалізація об’єктів.**

**Бібліотека класів користувача**

**Мета**

* Тривале зберігання та відновлення стану об'єктів.
* Ознайомлення з принципами серіалізації/десеріалізації об'єктів.
* Використання бібліотек класів користувача.

**1.Вимоги**

**1.1 Розробник:**

Олексієнко Микита Віталійович

студент групи КІТ-120В

15 варіант

**1.2 Загальне завдання**

Реалізувати програму відповідно до індивідуального завдання.

1. Реалізувати і продемонструвати тривале зберігання/відновлення раніше розробленого контейнера за допомогою серіалізації/десеріалізації.
2. Обмінятися відкомпільованим (без початкового коду) службовим класом (Utility Class) рішення задачі л.р. №3 з іншим студентом (визначає викладач).
3. Продемонструвати послідовну та вибіркову обробку елементів розробленого контейнера за допомогою власного і отриманого за обміном службового класу.
4. Реалізувати та продемонструвати порівняння, сортування та пошук елементів у контейнері.
5. Розробити консольну програму та забезпечити діалоговий режим роботи з користувачем для демонстрації та тестування рішення.

**1.3 Прикладні задачи**

Ввести текст. У тексті знайти та вивести всі слова-паліндроми (однаково читається в обох напрямках - зліва направо та справа наліво. Наприклад: "noon", "civic", "radar", "level", "rotor", "refer").

**2. Опис програми**

**2.1 Засоби ООП:**

У лабороторного роботі 6 , я використовую клас Program, який э головним класом та в якому безпосередньо оголошую приватні поля і публічні статичні методи. Також я оголосив клас Palindromes, який використовую для обробки знайдених паліндромів та клас Menu. Він дозволяє використовувати введення та виведення даних.

Клас Container я використовую для зберігання введеного масиву рядків та їх обробки. Він є серіалізуємим.

**2.2 Ієрархія та структура класів:**

* Program.java
* Palindomes.java
* Menu.java
* Container.java
* Iterator.java ( Interface )

**2.3 Важливі фрагменти програми:**

package ua.khpi.oop.alekseenko06;  
  
import javax.management.ObjectName;  
import java.io.FileInputStream;  
import java.util.Scanner;  
  
import java.io.FileOutputStream;  
import java.io.ObjectOutputStream;  
import java.io.ObjectInputStream;  
  
public class Palindromes {  
 private static final String ANSI\_RESET = "\u001B[0m";  
 private static final String ANSI\_YELLOW = "\u001B[33m";  
 private static final String ANSI\_GREEN = "\u001B[32m";  
 private static final String ANSI\_RED = "\u001B[31m";  
 private static final String ANSI\_WHITE = "\u001B[37m";  
  
 private Container container = new Container();  
 private Iterator iterator = container.getIterator();  
  
 public void Save () throws Exception {  
 FileOutputStream fileOutputStream = new FileOutputStream("save");  
 ObjectOutputStream objectOutputStream = new ObjectOutputStream(fileOutputStream);  
 objectOutputStream.writeObject(container);  
 objectOutputStream.close();  
 }  
  
 public void Load () throws Exception {  
 FileInputStream fileInputStream = new FileInputStream("save");  
 ObjectInputStream objectInputStream = new ObjectInputStream(fileInputStream);  
 container = (Container) objectInputStream.readObject();  
 }  
  
 public void InputText () {  
 System.out.println(ANSI\_YELLOW + "WARNING:" + ANSI\_RESET);  
 System.out.println("To finish entering text, leave a blank line and press ENTER.\n" + ANSI\_YELLOW  
 + "---------------" + ANSI\_RESET);  
  
 final int BUFFER\_SIZE = 100;  
  
 Scanner scan = new Scanner(System.in);  
  
 for ( int i = 0; i < BUFFER\_SIZE; i++ ) {  
 container.add( scan.nextLine() );  
 if ( container.contains("") ) {  
 container.setSize(i);  
 break;  
 }  
 }  
  
 System.out.print("\n");  
 }  
  
 public void ShowText () {  
  
 System.out.println(ANSI\_GREEN + "SHOW TEXT:" + ANSI\_RESET);  
 System.out.println(container.textToString() == null ? "null" : container.textToString());  
 System.out.println(ANSI\_WHITE + "- " + ( container.textToString().equals("")  
 ? ( ANSI\_RED + "null" + ANSI\_RESET ) : container.textToString() + ANSI\_RESET));  
  
 System.out.println();  
 }  
  
 public void ShowFind () {  
 int result = Find ( true );  
 if ( !(result == -1) ) {  
 System.out.println(ANSI\_GREEN + "SHOW FOUND COUNT:" + ANSI\_RESET);  
 System.out.println("- " + result + "\n");  
 }  
 }  
  
 public int Find ( boolean DEBUG ) {  
 if (container.getSize() == 0) {  
 System.err.println("ERROR");  
 System.out.println("- Please input text first [1]\n");  
 return -1;  
 }  
  
 int count = 0;  
 boolean isFirstWord = true;  
  
 while ( iterator.hasNext() ) {  
  
 String help = iterator.next();  
 StringBuilder str = new StringBuilder(help);  
  
 while (!str.toString().equals("")) {  
  
 int SpaceIndex = str.indexOf(" ");  
 StringBuilder word = new StringBuilder(SpaceIndex == -1 ? str : str.substring(0, SpaceIndex));  
 StringBuilder wordReverse = new StringBuilder(SpaceIndex == -1 ? str : str.substring(0, SpaceIndex));  
  
 wordReverse.reverse();  
  
 if (word.toString().equals(wordReverse.toString())) {  
 count++;  
 if (DEBUG) {  
 if (isFirstWord) {  
 isFirstWord = false;  
 System.out.println(ANSI\_GREEN + "FOUND PALINDROMES:" + ANSI\_RESET);  
 }  
 System.out.println("- " + word);  
 }  
 }  
  
 if (SpaceIndex == -1) {  
 str.delete(0, str.length());  
 } else {  
 str.delete(0, SpaceIndex + 1);  
 }  
 }  
  
 if (count == 0 && DEBUG) {  
 System.out.println("\n" + ANSI\_YELLOW + "WARNING:" + ANSI\_RESET);  
 System.out.println("There are no palindrome words in this text.\n");  
 }  
 }  
  
 System.out.println();  
 iterator.reset();  
 return count;  
 }  
}

**3. Висновок:**

* Навчився використовувати тривале зберігання та відновлення стану об'єктів.
* Ознайомився з принципами серіалізації/десеріалізації об'єктів.
* Навчився використовувати бібліотеки класів користувача.