

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. 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 вибірккову 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.

1.1 Розробник

- П.І.Б: Абдуллін О. Р.
- Група: КІТ-119а
- Варіант: 1

2 ОПИС ПРОГРАМИ

2.1 Було використано наступні засоби:

StringBuilder sb = new StringBuilder(str) – створення рядку типу StringBuilder;

string.length() – визначення довжини рядка;

Iterator<String> iterator – виділення строки за заданими позиціями;

FileOutputStream fos = new FileOutputStream("Serialization.ser");

ObjectOutputStream oos = new ObjectOutputStream(fos);

oos.writeObject(container); - серіалізація

FileInputStream fis = new FileInputStream("Serialization.ser");

ObjectInputStream ois = new ObjectInputStream(fis);

Container temp = (Container) ois.readObject(); - десеріалізація

2.2 Ієрархія та структура класів

Було створено 3 класи

1) public class Main - містить метод main

2) public class Conatiner - клас, що містить методи для роботи з контейнером.

3) public class MyIterator – клас, що містить методи для роботи з ітератором.

2.3 Важливі фрагменти програми

Конструктор класу Container

```
public Container (String... masstr) {  
    if(masstr.length != 0)  
    {  
        size = masstr.length;  
        arraystring = new String [size];  
        for(int i = 0; i < size; i++)  
            arraystring[i] = masstr[i];  
    }  
}
```

Метод повертає контейнер у вигляді рядку

```
public String toString() {  
    StringBuilder sb = new StringBuilder();  
    for(int i = 0; i < size; i++)  
        sb.append(arraystring[i] + " ");  
    return sb.toString();  
}
```

Метод додає рядок у кінець масиву

```
public void add(String str) {  
    String[] newmas = new String[size+1];  
    for(int i = 0; i < size; i++)  
        newmas[i] = arraystring[i];  
    newmas[size] = str;  
    size++;  
    arraystring = newmas;  
}
```

Метод очищення контейнеру

```
void clear() {  
    for(int i = 0; i < size; i++)  
        arraystring[i] = null;  
    size = 0;  
}
```

Метод видалення першого випадку вказаного елемента з контейнеру

```
boolean remove(String str) {  
    boolean bool = false;  
    int position = -1;  
    for(int i = 0; i < size; i++) {  
        if(arraystring[i].equals(str))  
        {  
            bool = true;  
            position = i;  
            break;  
        }  
    }  
    if(position != -1) {  
  
        String[] newmas = new String[size-1];  
        for(int i = 0; i < position; i++) {  
            newmas[i]=arraystring[i];  
        }  
        for(int i = position, n = position + 1; n < size; n++, i++) {  
            newmas[i] = arraystring[n];  
        }  
        size--;  
        arraystring = newmas;  
    }  
}
```

```

        return bool;
    }

```

Метод повертає масив даних

```

public Object[] toArray() {
    Object[] obj = new Object[size];
    for(int i = 0; i < size; i++)
        obj[i] = arraystring[i];
    return obj;
}

```

Метод, що повертає кількість елементів у контейнері

```

int size() {
    return size;
}

```

Метод, який повертає , якщо контейнер містить вказаний елемент

```

boolean contains(String str) {
    for(int i = 0; i < size; i++)
        if(arraystring[i].equals(str))
            return true;
    return false;
}

```

Метод, який повертає , якщо контейнер містить всі вказані елементи
зазначеного у параметрах

```

boolean containsAll(Container container) {
    int count = 0;
    if(size < container.size() || container.size() == 0)
        return false;
    for(int i = 0; i < size; i++ )
        for(int j = 0; j < container.size(); j++)
            if(arraystring[i].equals(container.arraystring[j])) {
                count++;
                break;
            }
    if(count == container.size())
        return true;
    else
        return false;
}

```

Метод виводу усіх даних у консоль

```

public void printContainer() {
    if(size() > 0)
    {
        for(String str : arraystring)
            System.out.println(str);
    }
    else
    {
        System.out.println("Conatainer is empty.");
    }
    System.out.println();
}

```

Метод пошуку позиції вказаного елемента контейнера

```

public int find(String str) {
    int position = 0;
    for(String string : arraystring)
    {
        if(string.equals(str))
            return position;
        position++;
    }
}

```

```

        return -1;
    }

```

Метод сортування за алфавітом (по зростанню, чи убиванню)

```

public void sortAlphabet(int typeofsorting) {
    if(typeofsorting == 1)
    {
        for(int i = 0; i < size - 1; i++)
            for(int j = 0; j < size - 1; j++)
                if(arraystring[j].compareTo(arraystring[j+1]) > 0)
                {
                    String temp = arraystring[j];
                    arraystring[j] = arraystring[j+1];
                    arraystring[j+1] = temp;
                }
    }
    else if (typeofsorting == 2)
    {
        for(int i = 0; i < size - 1; i++)
            for(int j = 0; j < size - 1; j++)
                if(arraystring[j].compareTo(arraystring[j+1]) < 0)
                {
                    String temp = arraystring[j];
                    arraystring[j] = arraystring[j+1];
                    arraystring[j+1] = temp;
                }
    }
}

```

Метод порівняння порівнює елементів у контейнері

```

public int compareElements (int position1, int position2)
{
    if(position1 > size || position2 > size)
        return -1;
    if(arraystring[position1 - 1].equals(arraystring[position2 - 1]))
        return 1;
    else
        return 0;
}

```

РЕЗУЛЬТАТ ВИКОНАННЯ РОБОТИ ПРОГРАММИ

Введений текст: It's a text. 1It's a sentece wich started neither vowel nor conconent!
 You are the best! Is that sentence started on vowel? Is that sentence started on
 conconent? This sentence stareted on conconent. 5*5=25?

```

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
2

It's a text.
1It's a sentece wich started neither vowel nor conconent!
You are the best! Is that sentence started on vowel?
Is that sentence started on conconent?
This sentence stareted on conconent.
5*5=25?

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
3

Enter data to add:
H3

It's a text.
1It's a sentece wich started neither vowel nor conconent!
You are the best! Is that sentence started on vowel?
Is that sentence started on conconent?
This sentence stareted on conconent.
5*5=25?
H3

```

Рисунок 6.1 – Результат роботи програми

```

Enter option:
6

Enter element to find:
5*5=25?
Position: 6

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
4

Enter element to delete:
H3

Container:
It's a text.
1It's a sentece wich started neither vowel nor conconent!
You are the best! Is that sentence started on vowel?
Is that sentence started on conconent?
This sentence starated on conconent.
5*5=25?

```

Рисунок 6.2 – Результат роботи програми

```

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
7

1. Ascending
2. Descending
1
|
1It's a sentece wich started neither vowel nor conconent!
5*5=25?
Is that sentence started on conconent?
It's a text.
This sentence starated on conconent.
You are the best! Is that sentence started on vowel?

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:

```

Рисунок 6.3 – Результат роботи програми

```

Enter option:
2

1It's a sentece wich started neither vowel nor conconent!
5*5=25?
Is that sentence started on conconent?
It's a text.
This sentence stareted on conconent.
You are the best! Is that sentence started on vowel?
H3
H3

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
8

1It's a sentece wich started neither vowel nor conconent!
5*5=25?
Is that sentence started on conconent?
It's a text.
This sentence stareted on conconent.
You are the best! Is that sentence started on vowel?
H3
H3

Enter positions of elements (from 1 to 8):
7
8
Elements on positions 7 and 8 are equal

```

Рисунок 6.4 – Результат роботи програми

```

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
9
|
This senteces are started on vowel:
Is that sentence started on conconent?
It's a text.
You are the best!
Is that sentence started on vowel?
The longest sentence is: Is that sentence started on conconent?
It has length: 38

This senteces are started on conconent:
This sentence stareted on conconent.
The longest sentence is: This sentence stareted on conconent.
It has length: 36

This senteces are started neither vowel nor conconent:
1It's a sentece wich started neither vowel nor conconent!
5*5=25?
The longest sentence is: 1It's a sentece wich started neither vowel nor conconent!
It has length: 57

```

Рисунок 6.5 – Результат роботи програми

```
Enter option:
11

Serialization is complete

Menu:
1. Enter new data
2. Show data
3. Add element in container
4. Delete element from container
5. Clear container
6. Find element in container
7. Sort container by alphabet
8. Compare elements in container
9. Type of string.
10. Zanochkin class
11. Serialize container
12. Deserialize
13. Exit
Enter option:
12

Deserialization is complete

1It's a sentece wich started neither vowel nor conconent!
5*5=25?
Is that sentence started on conconent?
It's a text.
This sentence stareted on conconent.
You are the best! Is that sentence started on vowel?
H3
H3
```

Рисунок 6.6 – Результат роботи програми

Висновок

Під час виконання лабораторної роботи було набуто навички роботи з серіалізацією/десеріалізацією та з розробки бібліотеки класів користувача у середовищі Eclipse IDE.