

Звіт

Лабораторна 5. Розробка власних контейнерів. Ітератори

Мета

- Набуття навичок розробки власних контейнерів.
- Використання ітераторів.

Вимоги

1. Розробити клас-контейнер, що ітерується для збереження початкових даних завдання л.р. №3 у вигляді **масиву рядків** з можливістю додавання, видалення і зміни елементів.
2. В контейнері реалізувати та продемонструвати наступні методи:
 - String toString() повертає вміст контейнера у вигляді рядка;
 - void add(String string) додає вказаний елемент до кінця контейнеру;
 - void clear() видаляє всі елементи з контейнеру;
 - boolean remove(String string) видаляє перший випадок вказаного елемента з контейнера;
 - Object[] toArray() повертає масив, що містить всі елементи у контейнері;
 - int size() повертає кількість елементів у контейнері;
 - boolean contains(String string) повертає true, якщо контейнер містить вказаний елемент;
 - boolean containsAll(Container container) повертає true, якщо контейнер містить всі елементи з зазначеного у параметрах;
 - public Iterator<String> iterator() повертає ітератор відповідно до Interface Iterable.
3. В класі ітератора відповідно до Interface Iterator реалізувати методи:
 - public boolean hasNext();
 - public String next();
 - public void remove().

4. Продемонструвати роботу ітератора за допомогою циклів *while* и *for each*.
5. Забороняється використання контейнерів (колекцій) і алгоритмів з [Java Collections Framework](#).

1.1. Розробник: Буй Зуї , КІТ119-а, варіант №26.

2. ОПИС ПРОГРАМИ

2.1. Засоби ООП: клас, метод класу, поле класу.

2.2. Важливі фрагменти програми:

```
3. package ua.khpi.oop1;
import java.util.Iterator;
public class Main {
    public static void main(String[] args) {
        MyContainer ar = new MyContainer("Мы вольные птийы пора брат.",
        "Туда где за тучей белеет гора.");
        Iterator<String> iter = ar.getIterator();
        String line;

        System.out.println("Вывод массива с помощью цикла while()");
        while(iter.hasNext())
        {
            line = iter.next();
            System.out.println(line);
        }

        System.out.println("\nДобавляем новый элемент : ");
        ar.add("дуб зеленый.");
        System.out.println("\nВывод массива с помощью цикла for() : ");
        iter = ar.getIterator();
        for(int i = 1; iter.hasNext(); i++)
        {
            line = iter.next();
            System.out.println(i + ". " + line);
        }

        System.out.print("\nНаходим предложение \"Мы вольные птийы
        пора брат.\": " + ar.contains("Мы вольные птийы пора брат."));

        System.out.println("\n\nМассив имеет все предложение другого
        массива " + ar.containsAll(new MyContainer("привет", "Мы вольные
        птийы пора брат.", "Туда где за тучей белеет гора.", "дуб зеленый.",
        "пока")));

        System.out.println("\nДлина массива : " + ar.size() + "\nУдаляем
```

```

одну элемент");
ar.remove("дуб зеленый.");
System.out.println("Длина массива : " + ar.size() + "\n");

System.out.println("Завершение работы");
ar.clear();
}
}

```

```

4. package ua.khpi.oop1;
import java.util.Iterator;
import java.util.NoSuchElementException;
public class MyContainer {
    private String[] array;
    private int size;

    public void clear()
    {
        for(int i = 0; i < size; i++)
            array[i] = null;

        size = 0;
    }

    public void add(String string)
    {
        String[] newArray = new String[size+1];

        for (int i = 0; i < size; i++)
            newArray[i] = array[i];

        size++;
        newArray[size - 1] = string;
        array = newArray;
    }

    public String toString()
    {
        StringBuilder string = new StringBuilder();

        for(int i = 0; i < size; i++)
            string.append(array[i] + " ");

        return string.toString();
    }
}

```

```
public boolean remove(String string)
{
    boolean result = false;
    int position = 0;

    for (int i = 0; i < size; i++)
        if(array[i].equals(string))
        {
            result = true;
            position = i;
            break;
        }

    if(result)
    {
        String[] newArray = new String[size-1];

        for (int i = 0; i < position; i++)
            newArray[i] = array[i];
        for (int i = position; i+1 < size; i++)
            newArray[i]=array[i+1];

        size--;
        array=newArray;
    }

    return result;
}

public Object[] toArray()
{
    Object[] object = new Object[size];

    for (int i = 0; i < size; i++)
        object[i]=array[i];

    return object;
}

public int size()
{
    return size;
}
```

```
public boolean contains(String string)
{
    for (int i = 0; i < size; i++)
        if (array[i].equals(string))
            return true;

    return false;
}

public String getLine(int index)
{
    return array[index];
}

public boolean containsAll(MyContainer container)
{
    boolean result = false;

    for (int i = 0; i < size; i++)
    {
        result = false;

        for (int j = 0; j < container.size(); j++)
        {
            if(array[i].equals(container.getLine(j)))
            {
                result = true;
                break;
            }
        }

        if(!result)
        {
            return false;
        }
    }

    return result;
}

public MyContainer(String... strings)
{
    if(strings.length > 0)
    {
```

```

        size = strings.length;
        array = new String[size];

        for (int i = 0; i < size; i++)
            array[i]=strings[i];
    }
}

public MyIterator<String> getIterator()
{
    return new MyIterator<String>();
}

private class MyIterator<String> implements Iterator {
    int index;

    @Override public boolean hasNext()
    {
        if(index < size)
            return true;
        else
            return false;
    }

    @Override public Object next()
    {
        if(index==size)
            throw new NoSuchElementException();

        return array[index++];
    }

    @Override public void remove()
    {
        MyContainer.this.remove(array[--index]);
    }
}
}

```

3.Результат

```
Вывод массива с помощью цикла while():
Мы вольные птийы пора брат.
Туда где за тучей белеет гора.

Добавляем новый элемент :

Вывод массива с помощью цикла for() :
1. Мы вольные птийы пора брат.
2. Туда где за тучей белеет гора.
3. дуб зеленый.

Находим предложение "Мы вольные птийы пора брат.": true

Массив имеет все предложение другого массива true

Длина массива : 3
Удаляем одну элемент
Длина массива : 2

Завершение работы

Process finished with exit code 0
|
```

Висновки

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

Програма протестована, виконується без помилок.