**Лабораторна робота №5**

**Розробка власних контейнерів. Ітератори**

**Мета:** Набуття навичок розробки власних контейнерів. Використання ітераторів.

**1 ВИМОГИ**

1. Розробити клас-контейнер, що [ітерується](http://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html) для збереження початкових даних [завдання л.р. №3](https://oop-khpi.gitlab.io/%D0%B7%D0%B0%D0%B2%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F/task03/#_4) 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](http://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html).
3. В класі ітератора відповідно до [Interface Iterator](http://docs.oracle.com/javase/8/docs/api/java/util/Iterator.html) 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](https://docs.oracle.com/javase/8/docs/technotes/guides/collections/).

**1.1 Розробник**

***Виконавець:*** Сиромятников Марк

***Група:*** КІТ-119а

**Варіант:** 19

**2 ОПИС ПРОГРАМИ**

Було використано наступні засоби:

* *import java.util.Scanner* – для вводу даних.
* System.***out***.printf() – для виводу тексту у консоль.
* .close(); - закриття потоку вводу
* import java.util.Iterator - для ітерування елементів контейнеру.
* import java.util.NoSuchElementException – обробка відсутності елементу.

**2.1 Ієрархія та структура класів**

Було створено 1 клас MainClas, який викликає методи виконання завдання та клас-контейнер ContainerClass, що містить у собі дані та відповідні методи, що викликаються першим класом.

**2.2 Важливі фрагменти програми**

public ContainerClass(String...strings) {

arrLength = strings.length;

array = new String[arrLength];

for(int i = 0; i < arrLength; i++) {

array[i] = strings[i];

}

}

public String toString() {

if(isEmpty) {

System.out.println("There are no content in the container.");

return null;

}

StringBuffer sb = new StringBuffer(array[0]);

for(int i = 1; i < arrLength; i++)

sb.append(array[i]);

return sb.toString();

}

public void add(String string) {

String[] newArray = new String[arrLength+1];

for(int i = 0; i < arrLength; i++) {

newArray[i] = array[i];

}

arrLength++;

newArray[arrLength-1] = string;

array = newArray;

}

public void clear() {

array = new String[0];

isEmpty = true;

System.out.println("There are no more content in the container.");

}

public boolean remove(String string) {

int cnt = 0;

boolean match = false;

String[] newArray = new String[arrLength];

for (int i = 0; i < arrLength; i++,cnt++) {

if(array[i] == string) {

match = true;

break;

}

}

if (match) {

newArray = new String[arrLength-1];

for(int i = 0; i < cnt; i++) {

newArray[i] = array[i];

}

for(int i = cnt; i < arrLength-1; i++) {

newArray[i] = array[i+1];

}

arrLength--;

}

array = newArray;

return match;

}

public Object[] toArray() {

String[] arrayCopy = new String[arrLength];

for (int i = 0; i < arrLength; i++)

arrayCopy[i] = array[i];

return arrayCopy;

}

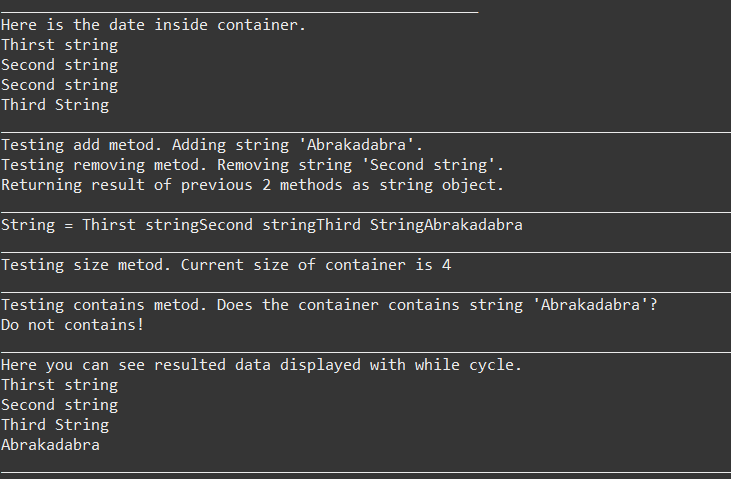
public int size() {

int size = array.length;

return size;

}

***Результат роботи програми***



**ВИСНОВКИ**

У результаті виконання лабораторної роботи було створено клас-контейнер, до нього розроблено відповідні методи та ітератор для переходу по даним.