

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра систем штучного інтелекту

## **Лабораторна робота №9**

з дисципліни

«Об'єктно-орієнтоване програмування»

**Виконав:**

студент групи КН-108

Яцків Юрій

**Викладач:**

Гасько Р.Т.

Львів – 2018 р.

## Завдання:

Лабораторна робота №9.

**Розробка власних контейнерів. Ітератори.**

**Серіалізація/десеріалізація об'єктів.**

**Бібліотека класів користувача**

### Мета

- Набуття навичок розробки власних контейнерів.
- Використання ітераторів.
- Тривале зберігання та відновлення стану об'єктів.
- Ознайомлення з принципами серіалізації/десеріалізації об'єктів.
- Використання бібліотек класів користувача.

### Вимоги

1. Розробити клас-контейнер, що ітерується ([docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html](https://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html)) для збереження початкових даних Вашого варіанту завдання з роботи №8 (**Прикладні задачі. Список з 1-15 варіантів**) у вигляді масиву рядків з можливістю додавання, видалення і зміни елементів.

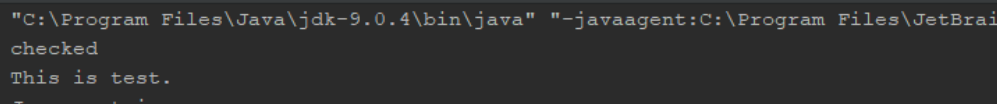
2. В контейнері реалізувати та продемонструвати наступні методи:
  - `String toString()` повертає вміст контейнера у вигляді рядка;
  - `void add(String string)` додає вказаний елемент до кінця контейнеру;
  - `void clear()` видаляє всі елементи з контейнеру;
  - `boolean remove(String string)` видаляє 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/>
6. Реалізувати і продемонструвати тривале зберігання/відновлення розробленого контейнера за допомогою серіалізації/десеріалізації.  
<https://docs.oracle.com/javase/8/docs/technotes/guides/serialization/index.html>
7. Обмінятися відкомпільованим (без початкового коду) службовим класом (Utility Class) рішення одного варіанту задачі (**Прикладні задачі. Список з 1-15 варіантів**) з сусіднім номером. 1 міняється з 2, 2 з 3, 3 з 4, 4 з 5 і т.д. Останній, 15 міняється з 1 варіантом і далі аналогічно.
8. Продемонструвати послідовну та вибірккову обробку елементів розробленого контейнера за допомогою власного і отриманого за обміном службового класу.

### Реалізація:

```
7  }
8
9  /**
10   * @param args cmd line arguments
11   * @return 0 on success, 1 on failure
12   */
13  public static void main(final String[] args) {
14      String[] array = new String[3];
15      array[0] = "This is test.";
16      array[1] = "Java container.";
17      array[2] = "Another one sentence.";
18
19      Container container = new Container(array);
20      container.add("One more test");
21      if (container.contains("This is test. ")) {
22          System.out.println("checked");
23      }
24
25      @SuppressWarnings("rawtypes")
26      Container.Iterator it = container.iterator();
27
28      for (String item : container) {
29          System.out.println(item);
30      }
31
32      while (it.hasNext()) {
```

```
Main.java x Container.java x
118 }
119
120 /**
121  * @author nick
122  *
123  * @param <String>
124  */
125 @SuppressWarnings("hiding")
126 class Iterator<String> implements java.util.Iterator<String> {
127
128     private int cursor;
129     private final int end;
130
131     /**
132      * @param data
133      * @throws IllegalArgumentException
134      */
135     public Iterator(java.lang.String[] data) {
136         this.cursor = 0;
137         this.end = data.length;
138     }
139
140     @Override
141     public boolean hasNext() {
142         return this.cursor < end;
143     }
144 }
```

## Результат:



```
Run Main
"C:\Program Files\Java\jdk-9.0.4\bin\java" "-javaagent:C:\Program Files\JetBrains\IntelliJ..."
checked
This is test.
Java container.
Another one sentence.
One more test
This is test.
Java container.
Another one sentence.
One more test

Process finished with exit code 0
```