

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»



**Лабораторна робота №14**

з курсу:

**“ОБ’ЄКТНО ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ”**

**Виконав:**

ст. гр.

КН-110

Халус

Богдан

**Прийняв:**

Гасько Р.Т.

Львів – 2018 р.

# Лабораторна робота № 14

**Завдання:** Колекції в Java

**Виконання роботи**

GitHub link - [https://github.com/bhalus/OOP\\_Java/tree/master/lab\\_14](https://github.com/bhalus/OOP_Java/tree/master/lab_14)

```
1 import java.util.Scanner;
2 import java.util.ArrayList;
3 import java.nio.charset.Charset;
4 import java.io.*;
5
6 class Timer implements Runnable {
7
8     private Thread t;
9     private String threadName;
10    private long currTime, stTime;
11
12    Timer ( String name)
13    {
14        threadName = name;
15        stTime = System.currentTimeMillis();
16        currTime = 0;
17    }
18
19    public void run ()
20    {
21        try {
22            while (true)
23            {
24                currTime = (System.currentTimeMillis() - stTime) / 1000;
25                Thread.sleep(500);
26            }
27        } catch (InterruptedException e) {
28            System.out.print("Timer Interrupted \n");
29        }
30    }
31
32    public void start ()
33    {
34        if (t == null)
35        {
36            t = new Thread ( this, threadName);
37            t.start();
38        }
39    }
40
41    public void stop ()
42    {
43        t = null;
44    }
45
46    public long getSeconds ()
47    {
48        return currTime;
49    }
50
51 }
```

```
54 public class Main
55 {
56
57     private static ArrayList<Excursion> excursions = new ArrayList<Excursion>();
58
59     private static void save() throws IOException
60     {
61         FileOutputStream out = null;
62
63         try {
64             out = new FileOutputStream("saves.sv");
65
66             for (int i = 0; i < excursions.size(); i++)
67             {
68                 out.write(excursions.get(i).getDate().getBytes(Charset.defaultCharset()));
69                 out.write((byte) '\n');
70                 out.write(excursions.get(i).getStartTime().getBytes(Charset.defaultCharset()));
71                 out.write((byte) '\n');
72                 out.write(String.valueOf(excursions.get(i).getDuration()).getBytes(Charset.defaultCharset()));
73                 out.write((byte) '\n');
74                 out.write(excursions.get(i).getPlace().getBytes(Charset.defaultCharset()));
75                 out.write((byte) '\n');
76                 out.write(excursions.get(i).getDescription().getBytes(Charset.defaultCharset()));
77                 out.write((byte) '\n');
78                 out.write(String.valueOf(excursions.get(i).getNumberOfParticipants()).getBytes(Charset.defaultCharset()));
79                 out.write((byte) '\n');
80
81                 for (int j = 0; j < excursions.get(i).getNumberOfParticipants(); j++)
82                 {
83                     out.write(excursions.get(i).getParticipantName(j).getBytes(Charset.defaultCharset()));
84                     out.write((byte) '\n');
85                 }
86             }
87             finally {
88                 out.close();
89             }
90
91         } catch (IOException e) {
92             e.printStackTrace();
93         }
94
95     private static void restore () throws IOException
96     {
97         FileInputStream in = null;
98
99         try {
100             in = new FileInputStream("saves.sv");
101
102             while (in.available() > 0)
103             {
104                 byte[] date = new byte[10];
105                 in.read(date);
106                 Excursion excursion = new Excursion();
107                 excursion.setDate(new Date());
108                 excursion.setStartTime(new Date());
109                 excursion.setDuration(0);
110                 excursion.setPlace("");
111                 excursion.setDescription("");
112                 excursion.setNumberOfParticipants(0);
113                 excursion.setParticipantNames(new ArrayList<String>());
114                 excursions.add(excursion);
115             }
116         } catch (IOException e) {
117             e.printStackTrace();
118         }
119     }
120 }
```