

```

1 package com.timbuchalka;
2
3 import java.util.ArrayList;
4
5
6 /**
7  * Main
8  *
9  */
10 public class Main {
11
12     public static void main(String[] args) {
13
14         String[] strArray = new String[10];
15         int[] intArray = new int[10];
16
17         ArrayList<String> strArrayList = new ArrayList<String>();
18         strArrayList.add("Tim");
19
20
21         /*
22          * Custom WrapperClass "IntClass"
23          */
24         // ArrayList<int> intArrayList = new ArrayList<int>(); // error: Type argument cannot be of primitive type!
25         ArrayList<IntClass> intClassArrayList = new ArrayList<IntClass>(); // type argument = custom IntClass = bad practice
26         intClassArrayList.add(new IntClass(54));
27
28
29         /*
30          * Wrapper class showcase (Integer and Double)
31          */
32         Integer integerValue = new Integer(54); // manual boxing
33         Double doubleValue = new Double(12.25); // manual boxing
34
35
36         /*
37          * WrapperClass
38          * ArrayList<WrapperClass>
39          */
40         ArrayList<Integer> intArrayList = new ArrayList<Integer>();
41         for(int i=0; i<=10; i++) {
42             /*
43              * Autoboxing
44              */
45             intArrayList.add(Integer.valueOf(i)); // boxing -> manually convert primitive type into WrapperClass
46             // intArrayList.add(i); // autoboxing -> automatically convert primitive type into WrapperClass
47         }
48
49         for(int i=0; i<=10; i++) {
50             /*
51              * Unboxing
52              */
53             System.out.println(i + " --> " + intArrayList.get(i).intValue()); // unboxing
54             // System.out.println(i + " --> " + intArrayList.get(i)); // auto-unboxing
55         }
56         System.out.println();
57
58
59
60         /*
61          * Autoboxing
62          */
63         Integer myIntValue = 56; // compiled by Java to -> Integer.valueOf(56);
64
65         /*
66          * Auto-Unboxing
67          */
68         int myInt = myIntValue; // compiled by Java to -> myIntValue.intValue();
69
70
71
72         ArrayList<Double> myDoubleValues = new ArrayList<Double>();
73         for(double dbl=0.0; dbl<=10.0; dbl += 0.5) {
74             /*
75              * Autoboxing
76              */
77             // myDoubleValues.add(Double.valueOf(dbl)); // manual boxing
78             myDoubleValues.add(dbl); // autoboxing
79         }
80
81         for(int i=0; i<myDoubleValues.size(); i++) {
82             /*
83              * Auto-Unboxing
84              */
85             // double value = myDoubleValues.get(i).doubleValue(); // manual unboxing
86             double value = myDoubleValues.get(i); // auto-unboxing
87             System.out.println(i + " --> " + value);
88         }
89     }
90 }
91
92
93 /**
94  * IntClass
95  * Custom int-Wrapper class
96  * IntClass(int myValue)
97  */

```

```
98  * getMyValue()
99  * setMyValue(int myValue)
100 */
101 class IntClass {
102     private int myValue;
103
104     public IntClass(int myValue) {
105         this.myValue = myValue;
106     }
107
108     public int getMyValue() {
109         return myValue;
110     }
111
112     public void setMyValue(int myValue) {
113         this.myValue = myValue;
114     }
115 }
```