

Module 3 - Java

Polymorphism, Event Handling, I18n

Advanced Java Certification Training

Akram M'Tir

1. If you have noticed class SearchHelper does not have a state but just a bunch of helper methods. What changes can you do to such classes? Make the relevant changes in the SearchHelper class.

```
1 package module3;
2
3 public interface ISearchHelper
4 {
5     public boolean search(Integer [] list, Integer number);
6     public boolean search(String [] list, String str);
7 }
```

2. Write an overloaded method in module3.MySearchHelper with the following signature: public Integer [] search(Double [] list, Double number) where the return type is an integer array containing the list of indexes where the number occurs.

```
1 package module3;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class MySearchHelper extends SearchHelper {
7
8     public Integer[] search( Double list[], Double number) {
9
10         List<Integer> indexes = new ArrayList<Integer>();
11
12         for (int i = 0; i<list.length; i++) {
13
14             if(list[i].doubleValue()== number.doubleValue()) {
15                 indexes.add(i);
16             }
17         }
18
19         Integer[] arr = new Integer[indexes.size()];
20         arr = indexes.toArray(arr);
21
22         return arr;
23     }
24 }
```

```

1 package module3;
2
3 import java.util.Arrays;
4
5 public class SearchHelper implements ISearchHelper
6 {
7     static final Double DBL_VAL = 2.3;
8     static final String STR_MESSG = "Found the Double Value %.2f at [Indexes] = %s";
9     public boolean search(Integer [] list, Integer number)
10    {
11        for (int i = 0; i < list.length; i++)
12        {
13            if (list[i].intValue() == number.intValue())
14                return true;
15        }
16        return false;
17    }
18
19    public boolean search(String [] list, String str)
20    {
21        for (int i = 0; i < list.length; i++)
22        {
23            if (list[i] == str)
24                return true;
25        }
26        return false;
27    }
28
29    public static void main(String[] args)
30    {
31        String [] str_list = {"Tom", "Y", "Z", "Q"};
32        Integer [] int_list = {1,2,3,4,5};
33        ISearchHelper sh = new SearchHelper();
34        System.out.println(sh.search(int_list, 2));
35        System.out.println(sh.search(str_list, "Tom"));
36
37        MySearchHelper searchDbl = new MySearchHelper();
38        Double dbl_list[] = {new Double(1.5), DBL_VAL, new Double(3.6), new Double(4.9),
39                             DBL_VAL, new Double(5.0), DBL_VAL};
40        System.out.printf(STR_MESSG, DBL_VAL, Arrays.toString( searchDbl.search(dbl_list,
41                             DBL_VAL) ));
42    }
43 }

```

Problems Javadoc Declaration Console Debug

<terminated> SearchHelper [Java Application] /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.144-0.b01.el7_4.x86_64/jre/bin/ja
true
true
Found the Double Value 2.30 at [Indexes] = [1, 4, 6]

3. There are 2 imports in module3.TestEventSource class

```
import java.util.ArrayList;
```

```
import java.util.Vector;
```

Even though these classes are not used in the program why is there a need to have these imports. [Hint: For a change, you don't need to look at the code, read the Javadoc comments]

We could have many listeners to register for a specific source and event. In that case a list data structure (java.util.ArrayList or java.util.Vector) could be used. To notify the listeners, we need to loop through the list and notify each registered listener. Vector is synchronized. This means if one thread is working on Vector, no other thread can get a hold of it. Unlike ArrayList, only one thread can perform an operation on vector at a time.

```
1 package module3;
2
3 import java.util.ArrayList; //import java.util.Vector;
4
5 // Event source.
6 public final class RandEventSource
7 {
8     private static final int ITERATIONS = 10;
9     // Holds the reference of the event consumer.
10    private ArrayList<RandEventListener> evlistener = new ArrayList<RandEventListener>();
11
12    // A list of listeners (Vector or ArrayList).
13    public void registerListener(RandEventListener listener)
14    {
15        evlistener.add(listener);
16    }
17    // This method is called by the event source to notify the
18    // event consumers about the event.
19    public void notifyListener(RandEvent evObj)
20    {
21        for(RandEventListener list : evlistener )
22            list.eventHandler(evObj);
23    }
24
25    public static void main(String[] args)
26    {
27        // Create event object
28        RandEventSource evSource = new RandEventSource();
29        // Register event consumers, more than one Consumer
30        evSource.registerListener(new RandEventConsumer());
31        evSource.registerListener(new RandEventConsumer());
32        evSource.registerListener(new RandEventConsumer());
33
34        double randomNum;
35        int truncRandomNum;
36        for (int i = 0; i < ITERATIONS; i++)
37        {
38            randomNum = Math.random();
39            truncRandomNum = (int)(randomNum*10000);
40            if (truncRandomNum%2 == 0)
41            {
42                RandEvent ev = new RandEvent("10000th decimal place is even for random number = " + randomNum,
43                    evSource);
44                evSource.notifyListener(ev);
45            }
46        }
47    }
48 }
```

Problems Javadoc Declaration Console Debug

<terminated> RandEventSource [Java Application] /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.144-0.b01.el7_4.x86_64/jre/bin/java (May 5, 2018, 6:00 AM)
Source: RandEventSource Payload: 10000th decimal place is even for random number = 0.6238844787413597
Source: RandEventSource Payload: 10000th decimal place is even for random number = 0.6238844787413597
Source: RandEventSource Payload: 10000th decimal place is even for random number = 0.17380451564689003
Source: RandEventSource Payload: 10000th decimal place is even for random number = 0.17380451564689003

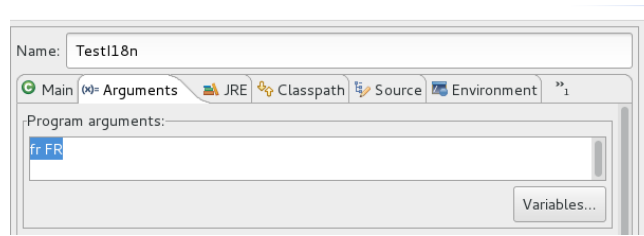
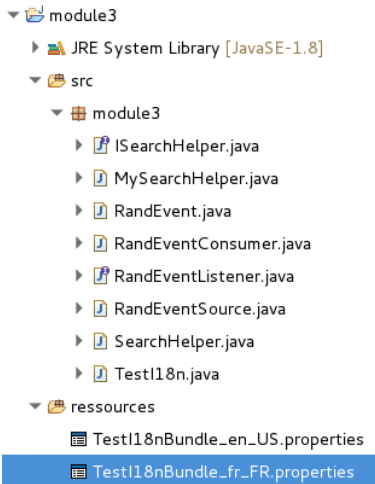
4. A class scans a list of random numbers generated on the fly (using `java.lang.Math.random()` method). If the number in the 10000 th decimal place is an even number then registered listeners would be notified about this event. Write a program to achieve this. If you are able to complete this you can handle any type of custom event handling and you have also learnt **Observer design pattern** (which we will learn in the last module). [Solution in `module3.randevent` package]

```
1 package module3;
2
3 // Class to model the Event
4 public class RandEvent {
5
6     protected String payload;
7     protected Object evSource;
8
9     public RandEvent(String payload, Object evSource)
10    {
11        this.payload = payload;
12        this.evSource = evSource;
13    }
14
15    public String getPayload()
16    { return payload; }
17
18    @Override
19    public String toString()
20    { return evSource.getClass().getSimpleName(); }
21
22 }
```

```
1 package module3;
2
3 // Interface that the event consumers should implement
4 public interface RandEventListener {
5
6     // Method Handler to the Event
7     public void eventHandler(RandEvent evObj);
8
9 }
10
```

```
1 package module3;
2
3 public class RandEventConsumer implements RandEventListener {
4
5     @Override // implementation of the method Handler to the Event
6     public void eventHandler(RandEvent evObj) {
7
8         System.out.println("Source: " + evObj.toString() + " "
9             + " Payload: " + evObj.getPayload());
10    }
11
12 }
13
```

5. Create a resource bundle with the keys being country name and values being their respective capitals in the resource folder. The locale should be for India with language code “en” and country code “IN”.



```

1 package module3;
2
3 import java.util.Locale;
4 import java.util.ResourceBundle;
5
6 public final class TestI18n
7 {
8     public static void main(String[] args)
9     {
10         String language, country;
11         /**
12          * The locale is passed as command line arguments. If nothing is passed
13          * then by default language=en and country=US is set.
14          */
15         if (args.length == 0)
16         {
17             language = "en";
18             country = "US";
19         }
20         else
21         {
22             language = args[0];
23             country = args[1];
24         }
25
26         // Create a new locale with the language and country code
27         Locale locale = new Locale(language, country);
28
29         /** Load the relevant resource bundle based on the locale.
30          * If language=en and country=US then resource from TestI18nBundle_en_US will be loaded.
31          * If language=fr and country=FR then resource from TestI18nBundle_fr_FR will be loaded.
32          */
33         ResourceBundle messages = ResourceBundle.getBundle("TestI18nBundle", locale);
34         // Print the values
35         System.out.println(messages.getString("button_label_1"));
36         System.out.println(messages.getString("button_label_2"));
37     }
38 }

```

Problems Javadoc Declaration Console Debug

<terminated> TestI18n [Java Application] /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.144-0.b01.el7_4.x86_64/jre/bin/java (May 5
Départ
Arrivée

6. Write a program to load this resource bundle and print the country names along with their capitals. The names should be printed in the format: The capital of <country> is <capital>.

```
1 package module3;
2
3 import java.util.Enumeration;
4 import java.util.Locale;
5 import java.util.ResourceBundle;
6
7 public class CapitalCountry {
8
9     public static void main(String args[]) {
10
11         ResourceBundle resBundle =
12             ResourceBundle.getBundle( "I18nBundle", new Locale("fr", "FR") );
13
14         Enumeration<String> keys = resBundle.getKeys();
15         String country = null;
16         String capital = null;
17
18         while(keys.hasMoreElements()) {
19             country = (String)keys.nextElement();
20             capital = resBundle.getString(country);
21             System.out.println("The capital of " + country + " is " + capital + ".");
22         }
23     }
24 }
```

Problems Javadoc Declaration Console Debug

<terminated> CapitalCountry [Java Application] /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.144-0.b01.el7_4.x86_64/j
The capital of France is Paris.

