Lab Serialization:

Description:

In this lab we create an instance of ListVsSetDemo. Then we are going to serialize and deserialize the instance.

Getting Started:

- Download labSerialization.zip

   Unzip the file and import it into Eclipse. **Check** the checkbox *Create top-level folder*

   At this point you have a package called labSerialization with one java file inside.

   Run the program. You should get an output similar to the one described on the right

Description:

In ColoredSquare:

- Implement interface Serializable from package java.io

- Notice the warning from Eclipse: have Eclipse auto-generate a serialVersionUID

   Right-click the yellow warning icon > Quick Fix > Add Generated Serial Version ID

   A private static final field of type long is added

In ListVsSetDemo:

- Implement interface Serializable

- Have Eclipse auto-generate a serialVersionUID

In LabSerialization:

- Add a method called `serialize`.

   It has no return value and two parameters: `demo` of type ListVsSetDemo and `filename` of type String

   - Use a **try with resource statement** when you serialize the ListVsSetDemo instance

      Save the result in the file name specified

   - In case of an exception print the message provided by the exception object

- In the main method comment out testDemo(demo);

   Then call the method serialize. For the file name choose a relative path that create the file **Demo.ser** in the directory `labSerialization`

   Print a brief message to let the user know that serialization is complete.

   Run the program and verify that the instance got serialized and that the file got created.

- Add a method called deserialize.

   It returns a ListVsSetDemo and has one parameter: a filename of type String

   - Use a try with resource statement to implement the deserialize method

   - Use a single catch block to catch both IOExceptions and FileNotFoundExceptions

- In main call the method deserialize.

   Assign the ListVsSetDemo returned to a new variable called **newDemo** and pass it to the method **testDemo**

Output:
```
List:
side:4 #FF0000
side:6 #0000FF
side:4 #FF0000
side:8 #FFFF00

Set:
side:6 #0000FF
side:8 #FFFF00
side:4 #FF0000
```