COP2800C Module 10 Practice Exercise

Design a Java class named LambdaTester which includes a main method that declares a lambda expression reference which implements the following interface:

```
interface SimpleDouble {
    //returns doubled value of parameter d
    double doubleValue(double d);
}
```

The program must declare an array of six Double values, followed by the declaration of an ArrayList which is instantiated using the declared array. Remember to program to the interface: declare the ArrayList as a List of Triple objects but instantiate it as an ArrayList.

Then write a foreach loop (not an integer-based index for loop) which iterates through each element of the ArrayList, calls the lambda expression reference with the element and displays the resulting doubled value.

Sample output follows for a double array declared as

```
Double[] dArray = { 1.0, 1.5, 2.0, 2.5, 3.0, 3.5 };
```

Remember that your doubled values must come from an ArrayList created from an array similar to the one shown above.

Expected output:

```
Double of 1.0: 2.0
Double of 1.5: 3.0
Double of 2.0: 4.0
Double of 2.5: 5.0
Double of 3.0: 6.0
Double of 3.5: 7.0
```

```
// LambaTester.java
// D. Singletary
// 11/12/23
// Implements and tests SimpleDouble functional interface

import java.util.*;

interface SimpleDouble {
    double doubleValue(double d);
}

public class LambdaTester {
    public static void main(String[] args) {

        // declare the data array
        Double[] dArray = { 1.0, 1.5, 2.0, 2.5, 3.0, 3.5 };
        // create the ArrayList from the array
        List<Double> dList = new ArrayList<>(Arrays.asList(dArray));

        // use a lambda to define the interface behavior
        SimpleDouble sd = val -> val * 2.0;

        // apply the lambda behavior to each element
        for (Double d : dList)
            System.out.println("Double of " + d + ": " + sd.doubleValue(d));
    }
}
```