Homework Turnin

Account:	6G_06 (rgalanos@fcps.edu)
Section:	6G
Course:	TJHSST APCS 2016-17
Assignment:	05-07
Receipt ID:	c5d965fd86cf1d47c4d8dfe80aabb026

Execution failed with return code 1 (general error). (Expected for JUnit when any test fails.)

Warning: Your program failed to compile :

```
IteratorLab_shell.java:30: error: missing return statement
      }
      ^
IteratorLab_shell.java:36: error: missing return statement
      }
      ^
IteratorLab_shell.java:43: error: missing return statement
      }
      ^
IteratorLab_shell.java:49: error: missing return statement
      }
      ^
IteratorLab_shell.java:55: error: missing return statement
      }
      ^
IteratorLab_shell.java:61: error: missing return statement
      }
      ^
IteratorLab_shell.java:68: error: missing return statement
      }
      ^
7 errors
```

Please correct your file(s), go back, and try to submit again. If you do not correct this problem, you are likely to lose a large number of points on the assignment. Please **contact your TA** if you are not sure why your code is not compiling successfully.

Turnin Failed! (See above)

There were some problems with your turnin. Please look at the messages above, fix the problems, then Go Back and try your turnin again.

GradeIt has a copy of your submission, but we believe that you will want to fix the problems with your submission by resubmitting a fixed version of your code by the due date.

We have received the following file(s):

IteratorLab.java (3594 bytes)

```
 1.  //Name:      Date:
 2.  // use for-each loops or iterators, not regular for-loops
 3. import java.io.*;
 4. import java.util.*;
 5. public class IteratorLab
 6. {
 7.    public static void main(String[] args)
 8.    {
 9.       System.out.println("Iterator Lab\n");
10.       int[] rawNumbers = {-9, 4, 2, 5, -10, 6, -4, 24, 20, -28};
11.       for(int n : rawNumbers )
12.          System.out.print(n + " ");
13.       ArrayList<Integer> numbers = createNumbers(rawNumbers);
14.       System.out.println("ArrayList: "+ numbers);      //Implicit Iterator!
15.       System.out.println("Count negative numbers: " + countNeg(numbers));
16.       System.out.println("Average: " + average(numbers));
17.       System.out.println("Replace negative numbers: " + replaceNeg(numbers));
18.       System.out.println("Delete zeros: " + deleteZero(numbers));
19.       String[] rawMovies = {"High_Noon", "High_Noon", "Star_Wars", "Tron", "Mary_Poppins",
20.                "Dr_No", "Dr_No", "Mary_Poppins", "High_Noon", "Tron"};
21.       ArrayList<String> movies = createMovies(rawMovies);
22.       System.out.println("Movies: " + movies);
23.       System.out.println("Movies: " +  removeDupes(movies));
24.    }
25.       // pre: an array of just int values
26.       // post: return an ArrayList containing all the values
27.    public static ArrayList<Integer> createNumbers(int[] rawNumbers)
28.    {
29.       ArrayList<Integer> list = new ArrayList<Integer>();
30.       for(int x: rawNumbers)
31.          list.add(x);
32.       return list;
33.    }
34.       // pre: an array of just Strings
35.       // post: return an ArrayList containing all the Strings
36.    public static ArrayList<String> createMovies(String[] rawWords)
37.    {
38.       ArrayList<String> list = new ArrayList<String>();
39.       for(String x: rawWords)
40.          list.add(x);
41.       return list;
42.    }
43.
44.       // pre: ArrayList a is not empty and contains only Integer objects
45.       // post: return the number of negative values in the ArrayList a
46.    public static int countNeg(ArrayList<Integer> a)
47.    {
48.       int count = 0;
49.       Iterator<Integer> it = a.iterator();
50.       while(it.hasNext())
51.          if((int)(it.next())<0)
52.             count++;
53.       return count;
54.    }
55.       // pre: ArrayList a is not empty and contains only Integer objects
56.       // post: return the average of all values in the ArrayList a
57.    public static double average(ArrayList<Integer> a)
58.    {
59.       double avg = 0;
```

```
60.         for(int x: a)
61.            avg+=x;
62.         return avg/a.size();
63.      }
64.         // pre: ArrayList a is not empty and contains only Integer objects
65.         // post: replaces all negative values with 0
66.      public static ArrayList<Integer> replaceNeg(ArrayList<Integer> a)
67.      {
68.         ListIterator<Integer> it = a.listIterator();
69.         while(it.hasNext())
70.            if((int)(it.next())<0)
71.               it.set(new Integer(0));
72.         return a;
73.      }
74.         // pre: ArrayList a is not empty and contains only Integer objects
75.         // post: deletes all zeros in the ArrayList a
76.      public static ArrayList<Integer> deleteZero(ArrayList<Integer> a)
77.      {
78.         Iterator<Integer> it = a.iterator();
79.         while(it.hasNext())
80.            if((int)(it.next())==0)
81.               it.remove();
82.         return a;
83.
84.      }
85.         // pre: ArrayList a is not empty and contains only String objects
86.         // post: return ArrayList without duplicate movie titles
87.         // strategy: start with an empty array and add movies as needed
88.      public static ArrayList<String> removeDupes(ArrayList<String> a)
89.      {
90.         ArrayList<String> list = new ArrayList<String>();
91.         for(String x: a)
92.            if(!list.contains(x))
93.               list.add(x);
94.         return list;
95.      }
96.
97. }
98.
99.
```