Unit 7 Review 2 worksheet   Name: Key 1/24/22

This class goes with problems 1 through 3:

```
import java.util.ArrayList;

class Main {

  public static void main(String[] args) {

    //code not shown creates an ArrayList of Strings called words filled with valid data
   System.out.println(findLongest(words)); (3)
  }
  HERE (1)
}
```

1. If you were asked to write a method "findLongest" that accepts an ArrayList of String objects and returns the longest String, where would you add it above? Write "HERE" at that spot.

2. Write the complete method here. To recap, the method **findLongest** accepts an ArrayList of String objects and returns the longest String from the ArrayList. Precondition: the ArrayList contains at least one String that contains at least one character.

```
public static String findLongest(ArrayList<String> words)
{
   String longest = words.get(0);
   for(string w: words)
      { if(w.length() > longest.length()) longest = w;
      }
   return longest; }
```

3. Write one line of code in the correct spot up in the class up top that calls **findLongest** and prints the result.

4. What are the contents of the ArrayList **nums** after this code runs?

```
ArrayList<Integer> nums = new  ArrayList<Integer>();
nums.add(8);
nums.add(0,6);
nums.add(7);
nums.add(5);
nums.set(3,3);
nums.add(2,0);
nums.add(2,9);
nums.remove(4);
```

68903

8
68
687
6875
6873
68073
689073 → 68903

5. Assume code you cannot see added several more elements to the **nums** from problem 4. Write code to delete the last item from the **nums**:

```
nums.remove(nums.size()-1);
```

6. Write out the state of the following integer array after each pass of an **insertion** sort. Write the state of the array for each pass even if nothing changed.

1 9 6 3 2 5

196325 pass 1 (no change)

169325 pass 2

136925 pass 3

123695 pass 4

123569 pass 5

7. Write out the state of the following integer array after each pass of an **selection** sort. Write the state of the array for each pass even if nothing changed.

1 9 6 3 2 5

196325 pass 1 no change

126395 pass 2

123695 pass 3

123596 pass 4

123569 pass 5

8. Add code below which creates an ArrayList **newVals** and then copies all values from the array **vals** into **newVals**:

```
int[] vals = //code not shown fills vals with a bunch of integers
ArrayList<Integer> newVals = new ArrayList<Integer>();
for(int n : vals)
    newVals.add(n);
```

9. Write code to go through the ArrayList **newVals** from problem 8 that deletes any elements that are less than zero:

```
for (int i = newVals.size()-1 ; i>-1; i--)
  {
   if (newVals.get(i)<0) newVals.remove(i);
  }
```