**Worksheet 10 - ArrayLists**

1. Look at the code snippets below and mark whether they are valid or invalid. (Invalid includes both compile-time and run-time errors).

|  |  |
| --- | --- |
| **Code snippet** | **✓ OR 🗶** |
| ArrayList<String> myList = new ArrayList<>();  myList.add("1");  myList.add("2"); |  |
| ArrayList<int> myList = new ArrayList<>(); |  |
| ArrayList<Integer> myList = new ArrayList<>();    myList.add(50);  myList.add(20);    int i = myList.get(3); |  |
| ArrayList<String> myList = new ArrayList<>();  myList.add("1");  myList.add("2");  System.out.println(myList.size()); |  |
| ArrayList<Integer> myList = new ArrayList<>();    myList.add(50);  myList.add(20);    boolean b = myList.get(0); |  |

1. Map the following primitive types to their corresponding Wrapper classes.

|  |  |
| --- | --- |
| **Primitive** | **Wrapper** |
| int |  |
| char |  |
| boolean |  |
| float |  |
| double |  |

1. Given the code fragment below;

ArrayList<String> names = new ArrayList( );

names.add("A");

names.add(0, "B");

names.add("C");

names.remove(1);

What would the following lines print out?

|  |  |
| --- | --- |
| System.out.print(names.get(0)); |  |
| System.out.print(names.get(1)); |  |
| System.out.print(names.get(2)); |  |

1. Given the following ArrayList:

ArrayList<Integer> myNums = new ArrayList<>();

Write code to do each of the following:

1. Fill up the first 100 locations of the ArrayList with random numbers from 1 to 100.
2. Add 1 to each location in the ArrayList
3. Count the number of even numbers in the ArrayList.
4. Print the elements of the list in reverse order.
5. Find out whether the number 50 is in the list or not.
6. Finally, clear the ArrayList and confirm by printing its size
7. Write a program that accepts 10 names from the user. Each name should be checked to see if it already exists in the ArrayList. If the name already exists you should display a message and not add the name.
8. Write a void, static method named removeZeros that is passed an ArrayList parameter named *list*. The method must remove all values of 0. The main method is the caller of this method, and should print the list again after calling the method to make sure it is working.