**Integer and Double**

So far in this course, we have come into contact with the wrapper classes Integer and Double on two occasions. The one you probably remember occurred when we [explained](https://www.eimacs.com/eimacs/mainpage?epid=E2365694465&cid=162149#Wrappers) by what means it is possible to store the primitive data types int and double in ArrayLists. At the time, we limited ourselves to stating that such classes exist, explaining how to declare ArrayLists with the corresponding type parameter, and assuring you that, having declared an ArrayList appropriately, you could add an int (or a double) to the ArrayList and get it back out of the ArrayList without having to be concerned about the conversions between ints and Integers (and between doubles and Doubles) that are going on behind the scenes. Let's take a moment now to consider those conversions.

Instances of the Integer class are created using that class's constructor, whose signature is

Integer( int n )

To retrieve the value of an Integer as an int we may use the accessor method intValue. The constructor and this method provide the tools with which the behind-the-scenes conversions between ints and Integers are actually performed. Similarly, the constructor for the Double class has the following signature:

Double( double d )

and the value of a Double may be retrieved as a double using the accessor method doubleValue.

Run the following code to verify that the behavior when storing ints in an ArrayList<Integer> and then retrieving the ints from the ArrayList is the same whether you perform the conversions yourself by hand or you allow them to be performed automatically.

    int[] t = { 1, 2, 3, 6, 10, 15, 21 };   
    ArrayList<Integer> a = new ArrayList<Integer>();   
    ArrayList<Integer> b = new ArrayList<Integer>();   
  
    // make two ArrayLists from array t   
    for ( int n : t )   
    {   
      a.add ( new Integer( n ) );   
      b.add ( n );   
    }   
  
    // print the contents of ArrayList a   
    for ( Integer x : a )   
      System.out.print( x.intValue() + " " );   
  
    // newline   
    System.out.println();   
  
    // print the contents of ArrayList b   
    for ( int n : b )   
      System.out.print( n + " " );

[Show program details »](https://www.eimacs.com/eimacs/mainpage?cid=162149&epid=E2365696285)

1 2 3 6 10 15 21    
1 2 3 6 10 15 21

**Exercise 147**

The program below is intended to find the arithmetic mean of the numbers stored in the array q in two ways: once by storing the numbers in an ArrayList d, where you allow all the necessary conversions to be performed automatically; and once by storing them in an ArrayList e, where you perform all the conversions by hand. Complete the program.

    double[] q = { 0.5, 2.4, 7.4, 2.8, -6.2 };   
    ArrayList<Double> d = new ArrayList<Double>();   
    ArrayList<Double> e = new ArrayList<Double>();   
  
    for ( double x : q )   
    {   
      d.add( x );   
      e.add ( new Double ( x ) );   
    }   
  
    double dTotal = 0.0,   
           eTotal = 0.0;

  for ( double x : d )

     dTotal += x;

   for ( Double y : e )

     eTotal += y.doubleValue();

   System.out.println( "Mean of d is " + (dTotal / d.size()) );

   System.out.println( "Mean of e is " + (eTotal / e.size()) );

Mean of d is 1.3800000000000003   
Mean of e is 1.3800000000000003



The other contact we have had with the Integer class and the Double class was rather more surreptitious. It occurred much earlier in this course, when we [introduced](https://www.eimacs.com/eimacs/mainpage?epid=E2395202336&cid=162149#WrapperMethods) the methods Integer.parseInt, Integer.toString, Double.parseDouble, and Double.toString. Now that you have more experience, you will readily recognize that the dot notation used to name these methods is a sure indication that they are methods of an Integer class and a Double class. We remark that these are just a few of the methods that are provided by these wrapper classes. As far as the Advanced Placement examination is concerned, however, you are only required to be familiar with a small number of them. For a complete listing, click [here](javascript:secWindow('mainpage?epid=E2257518102&cid=162149&s=2','WMethPop',540,460,50,50,'menubar,scrollbars,resizable')).

The Integer and Double methods with which you need to be familiar for the Advanced Placement examination are as follows:

|  |  |
| --- | --- |
| Integer | |
| int intValue() | [Review](https://www.eimacs.com/eimacs/mainpage?epid=E2257518102&cid=162149&s=2) |
| int compareTo( Integer i ) | [Later](javascript:void(0)) |
| Double | |
| double doubleValue() | [Review](https://www.eimacs.com/eimacs/mainpage?epid=E2257518102&cid=162149&s=2) |
| int compareTo( Double d ) | [Later](javascript:void(0)) |
| Both | |
| boolean equals( Object obj ) | [Review](https://www.eimacs.com/eimacs/mainpage?epid=E2257518102&cid=162149&s=2) |
| String toString() | [Review](https://www.eimacs.com/eimacs/mainpage?epid=E2257518102&cid=162149&s=2) |

For a complete listing of all the methods available, we recommend that you refer to the Java language specifications for [Integer](http://docs.oracle.com/javase/6/docs/api/java/lang/Integer.html#method_summary) and [Double](http://docs.oracle.com/javase/6/docs/api/java/lang/Double.html#method_summary) on the Oracle Corporation website.