

## APCS

Write class Class1 to work with the driver shown.

```
public class Driver1
{
    public static void main( String args[] )
    {
        //The following line constructs a Class1 object (koby).
        //This constructor initializes two int variables, one double variable,
        //and one String variable. You pick the default values.
        Class1 koby = new Class1();

        //This next line tells you there is a toString method in Class1. Examine the output to see
        //what this method returns. Remember that the toString method always returns a String
        System.out.println( koby );

        System.out.print("Enter a whole number less than 10 . . . ");
        int a = SavitchIn.readLineInt();
        System.out.print("\nEnter a second whole number less than " +
            "10 (different from the first one) . . . ");
        int b = SavitchIn.readLineInt();

        //The following method setThreeNumbers sets the value of the two int variables
        //and the double variable
        koby.setThreeNumbers( a, b, Math.PI );

        //The following line calls on method sort. This line stands alone,
        //without an assignment operator (equal sign) and not inside a System.out.print
        //statement. This method sorts the two int variables and has a void return value.
        koby.sortTheInts();

        //The next line calls on methods getSmall and getBig. These are contained in
        //a System.out.print statement so they cannot be void and must return a number.
        //We know that the numbers returned are integers so both methods have int return
        //types.
        System.out.println( "\nThe numbers entered, in order, are: " +
            koby.getSmall() + " and " + koby.getBig() );
        //The next line calls on method addNumbers. This method returns an int which is stored in
        //variable sum.
        int sum = koby.addNumbers();

        //The next statement calls on method getAverage. This method returns the average of
        //the two numbers entered. Therefore, the return type of getAverage is double.
        double ave = koby.getAverage();

        System.out.println( "Their sum is: " + sum + " " + "and their average is: " + ave );
        //The next statement calls on method aToPowerB. Math method pow returns a double value
        double aToB = koby.aToPowerB();
        System.out.println( "\n" + a + " to the power " + b + " is: " + aToB );

        System.out.print("\nEnter a third number with a decimal . . . ");
        double c = SavitchIn.readLineDouble();

        //This next line calls on method getSumOfThree. This method received the third
        //number entered, which is a double. So method getSumOfThree should return a double.
        System.out.println("\nThe sum of the three numbers is: " + koby.getSumOfThree( c ) );

        //void set method
        koby.setNum3( 15.2 );

        //New Class1 object. This requires a different constructor which will assign
        //default values.
        Class1 sam = new Class1();

        sam.setName( "Newport" );

        //The next line calls on method toString
        System.out.print( koby );
        System.out.print( sam );
    }
}
```

}

SAMPLE OUTPUT

The data are:

2 5 2.2

The sum of the numbers is 9.2

We are in beautiful Sunny Harbor

Enter a whole number less than 10 . . . 3

Enter a second whole number less than 10 (different from the first one) . . . 9

The numbers entered, in order, are: 3 and 9

Their sum is: 12 and their average is: 6.0

3 to the power 9 is: 19683.0

Enter a third number with a decimal . . . 5.75

The sum of the three numbers is: 17.75

num3 has been changed from 5.75 to 15.2

The data are:

3 9 15.2

The sum of the numbers is 27.2

We are in beautiful Sunny Harbor

The data are:

2 5 2.2

The sum of the numbers is 9.2

We are in beautiful Newport Harbor