Command Line arguments.

Command line arguments are values passed into your program when using the **java** command/program.

```
        java Echo apple pear lemon
```

(**Note**: You would do this having first successfully compiled your program using **javac Echo.java**)

Any information appearing on the same line as the java command and **after the class name (program name)** is passed in to the main method of your program.

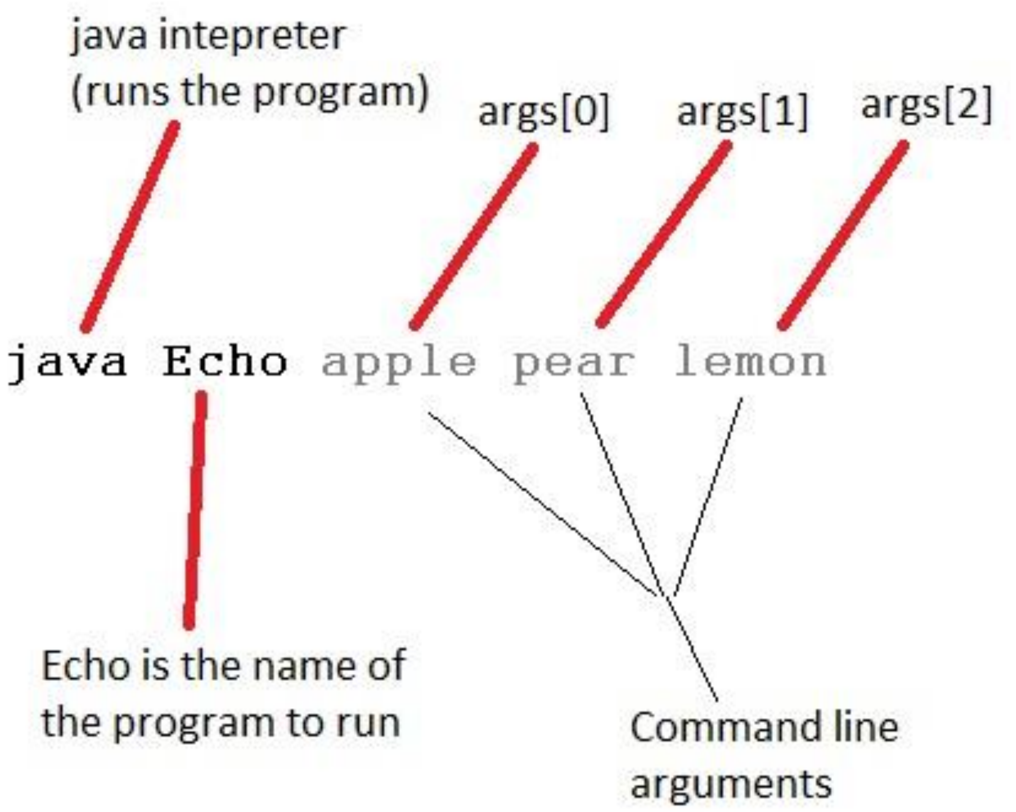
The arguments are stored in the main method formal parameter **String[] args**

The arguments which are elements (**individual Strings**) in a an array can then be used or manipulated in your main method . See example below:

```
System.out.println("The first argument is : " + args[0]);
```

In the example above this will print the message and "**apple**".

Command line arguments



java intepreter
(runs the program)

args[0]

args[1]

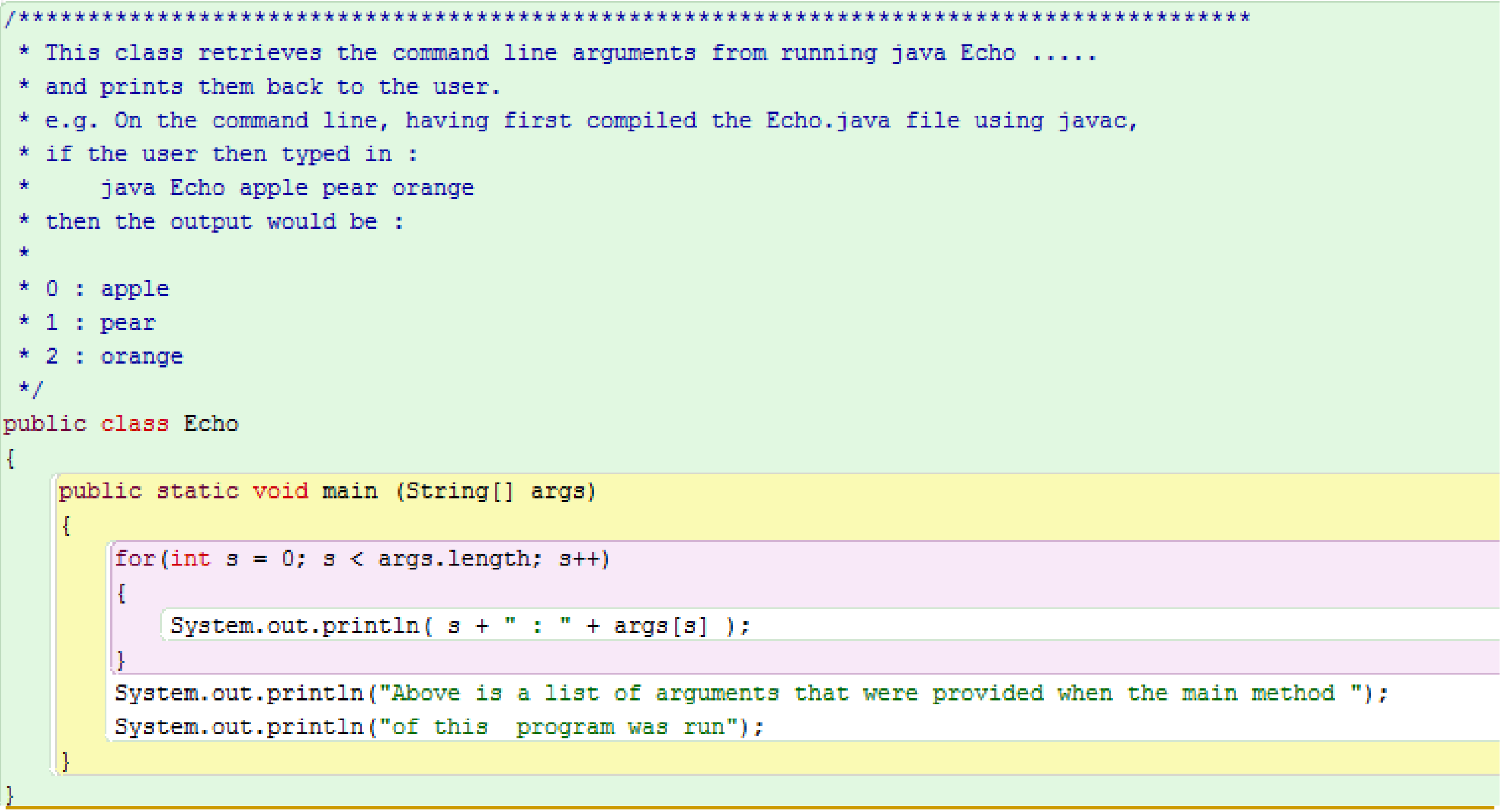
args[2]

```
java Echo apple pear lemon
```

Echo is the name of
the program to run

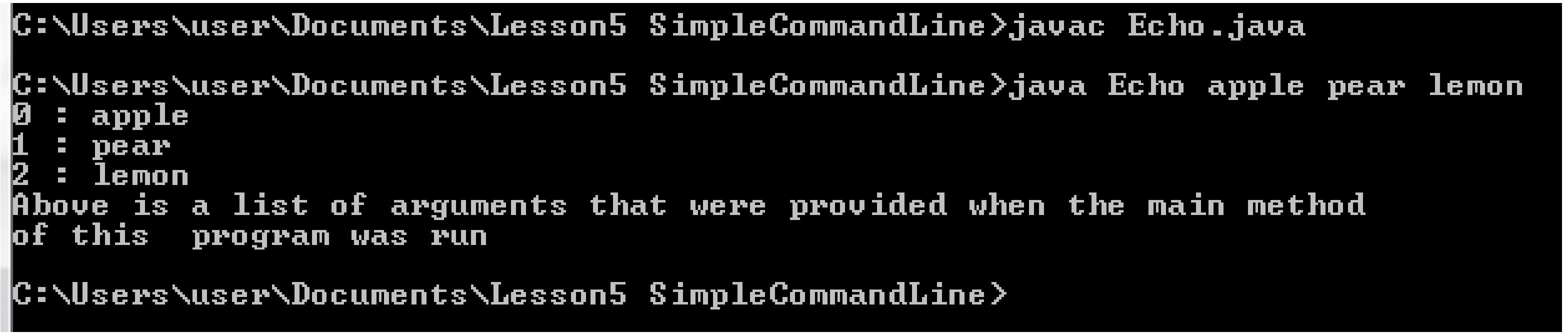
Command line
arguments

The Echo.java program. Prints out the command line arguments to screen



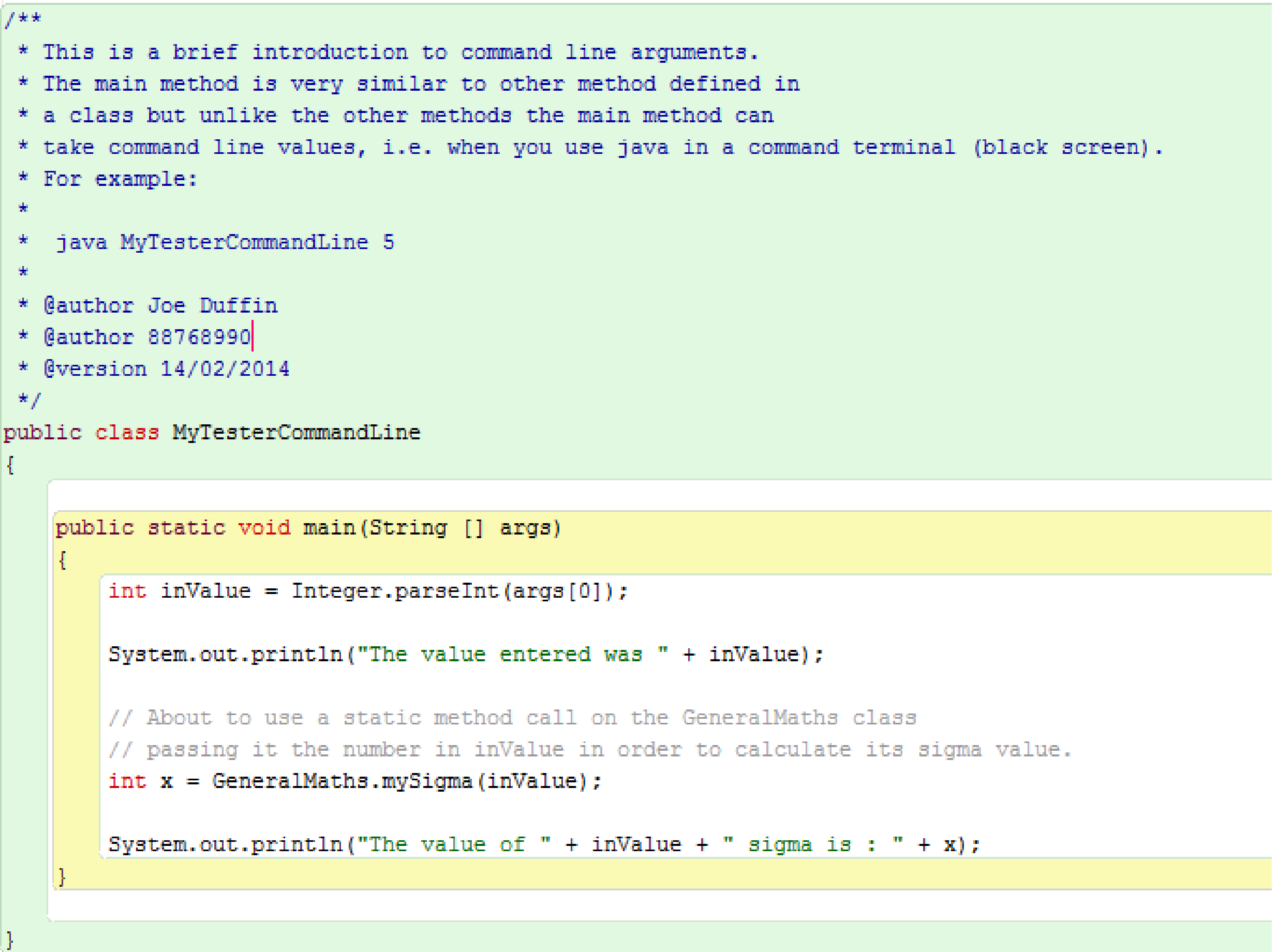
```java
/*****************************************************************************************
 * This class retrieves the command line arguments from running java Echo .....
 * and prints them back to the user.
 * e.g. On the command line, having first compiled the Echo.java file using javac,
 * if the user then typed in :
 *     java Echo apple pear orange
 * then the output would be :
 *
 * 0 : apple
 * 1 : pear
 * 2 : orange
 */
public class Echo
{
    public static void main (String[] args)
    {
        for(int s = 0; s < args.length; s++)
        {
            System.out.println( s + " : " + args[s] );
        }
        System.out.println("Above is a list of arguments that were provided when the main method ");
        System.out.println("of this  program was run");
    }
}
```

Output of the Echo program on the command line



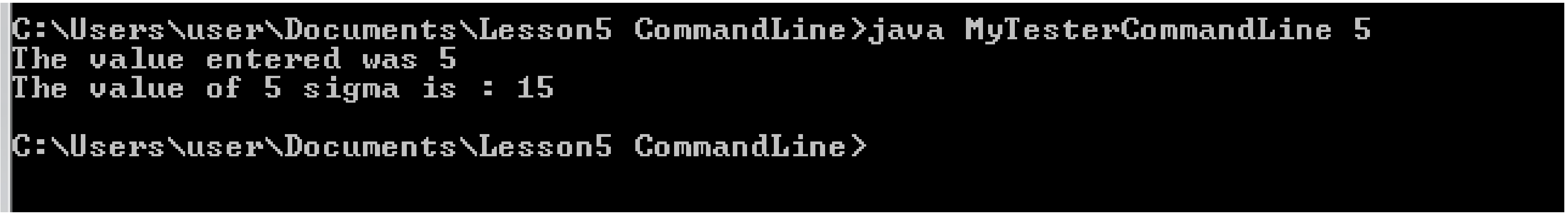
```
C:\Users\user\Documents\Lesson5 SimpleCommandLine>javac Echo.java

C:\Users\user\Documents\Lesson5 SimpleCommandLine>java Echo apple pear lemon
0 : apple
1 : pear
2 : lemon
Above is a list of arguments that were provided when the main method
of this  program was run

C:\Users\user\Documents\Lesson5 SimpleCommandLine>
```

Converting the command line argument into an integer using the Integer.parseInt( ) method.



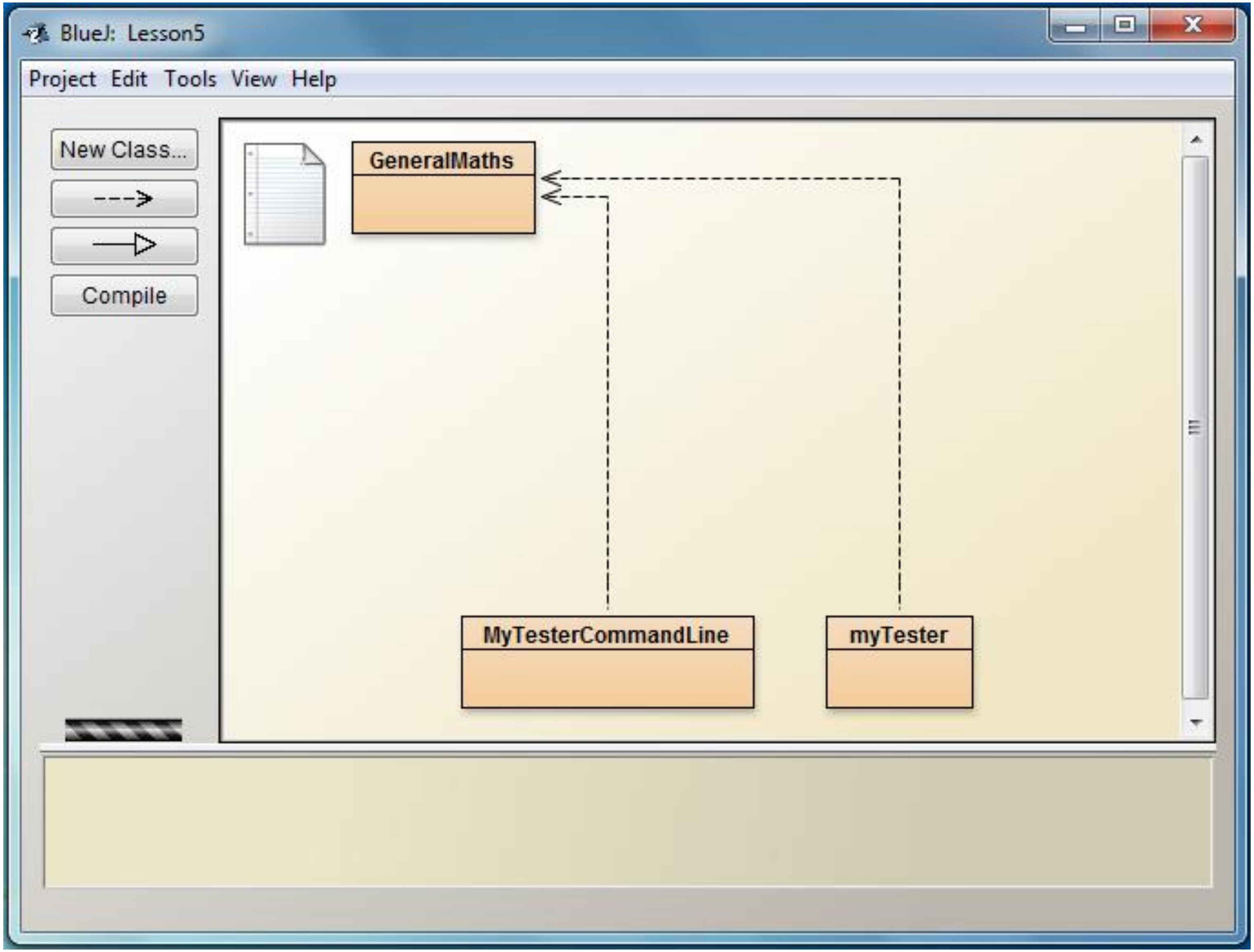
```
/**
 * This is a brief introduction to command line arguments.
 * The main method is very similar to other method defined in
 * a class but unlike the other methods the main method can
 * take command line values, i.e. when you use java in a command terminal (black screen).
 * For example:
 *
 *  java MyTesterCommandLine 5
 *
 * @author Joe Duffin
 * @author 88768990
 * @version 14/02/2014
 */
public class MyTesterCommandLine
{

    public static void main(String [] args)
    {
        int inValue = Integer.parseInt(args[0]);

        System.out.println("The value entered was " + inValue);

        // About to use a static method call on the GeneralMaths class
        // passing it the number in inValue in order to calculate its sigma value.
        int x = GeneralMaths.mySigma(inValue);

        System.out.println("The value of " + inValue + " sigma is : " + x);
    }

}
```

Running MyTesterCommandLine using the java command



```
C:\Users\user\Documents\Lesson5 CommandLine>java MyTesterCommandLine 5
The value entered was 5
The value of 5 sigma is : 15

C:\Users\user\Documents\Lesson5 CommandLine>
```

UML showing two classes using the GeneralMaths class


BlueJ: Lesson5
Project
Edit
Tools
View
Help
New Class...
Compile
GeneralMaths
MyTesterCommandLine
myTester