The University of the West Indies, St. Augustine
COMP 2603 Object Oriented Programming 1
Lab #1

Part 1: Getting Started with BlueJ

Welcome to your first Java programming lab! Follow the steps below to create your first program.

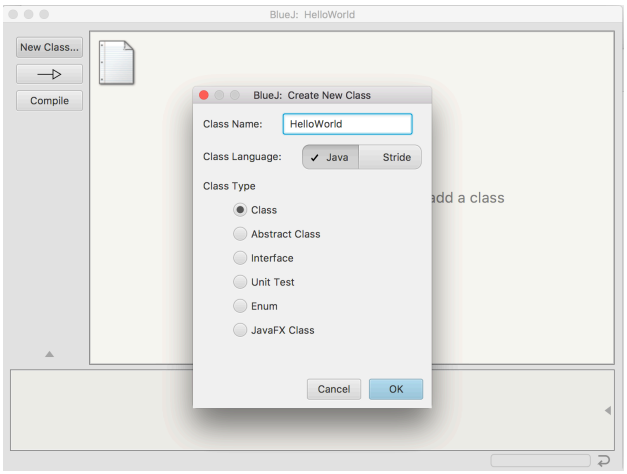
1) Download and install the BlueJ Editor on your machine.

Download Links
BlueJ: http://www.bluej.org/
JDK 11 or above: https://www.oracle.com/java/technologies/javase-jdk11-downloads.html

2) Open the BlueJ Editor and create a new Project called Lab1.

3) Create a new Java class by clicking the *New Class* button.

4) Specify *HelloWorld* as the name of the class and save the file in your Lab1 folder. Your new Java class will be displayed in the editor.



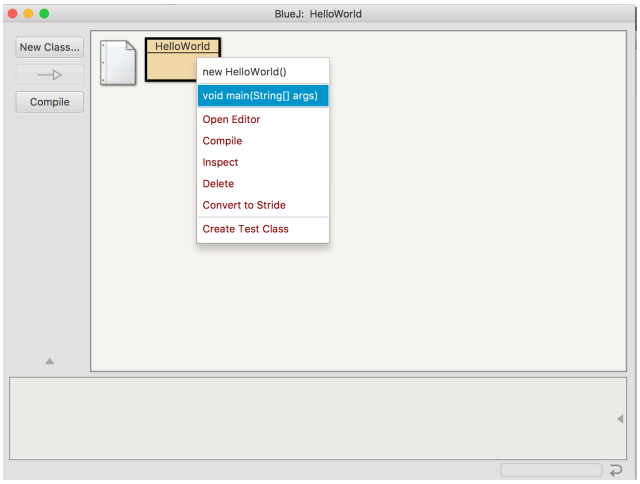
5) Delete the existing code and **type** the following code (do not copy & paste):

```
// My First Java Program
public class HelloWorld{

      public static void main(String[ ] args){
            System.out.println("Hello World");
      }
}
```

6) Compile the program and observe that there are no errors displayed in the debugging output pane.

7) Execute the program and observe the output printed on the screen. To execute, right-click on the class in the main window pane and select the option 'void main(String[ ] args)'



Part 2: Getting Started with Java

Create a new Java class called Lab1PartTwo that achieves the following within the main method:

1) Prints the following message to the screen: `My name is Englebert.`
2) Prints the following using a String variable.

```
My name is Englebert Humperdinck
```

String variables are declared and initialised as follows:

```
String lastname;            // declaration
lastname = "Humperdinck";   // initialisation
```

This may be done in several ways, try them out:

```
System.out.println("My name is Englebert" + lastname);
System.out.println("My name is Englebert Humperdinck");
System.out.println("My name is Englebert" + "Humperdinck");
```

3) Prints your first name and last name.
4) Read a user's name entered via the keyboard, and prints the name out to the screen within a sentence. Example:

```
  Englebert
>> My name is Englebert
```

The Scanner class is used to capture input from the user via the keyboard

```
Scanner keyboard = new Scanner(System.in);
```

You need to include the following import statement at the top of your program

```
import java.util.Scanner;
```

5) Simulates the following exchange where lines 1 and 3 are outputs from the program and line 2 is the line typed by the user.

```
>> Hi, what's your name?
 Englebert
>> Nice to meet you Englebert!
```

6) Simulates the following exchange where lines 1 and 3 are outputs from the program and line 2 is the line typed by the user. Tip: You will need to parse the String to extract the name

```
>> Hi, what's your name?
 My name is Englebert
>> Nice to meet you Englebert!
```

The substring(..)method extracts a portion of a String starting from a starting index value to until an ending index value

```
String abracadabra = "abracadabra";
String abc = abracadabra.substring(0,2) + abracadabra.substring(4,5);
```

Part 3: Exercises

Create a new Java class called Lab1PartThree.
You should write separate methods for these exercises and invoke them as follows:

```
public class Lab1PartThree{
      public static void exercise1( ){
            // code for exercise 1 goes here
      }
      public static void exercise2( ){
            // code for exercise 2 goes here
      }
      public static void main(String[] args){
            exercise1(); // invokes the exercise1( ) method
            exercise2(); // invokes the exercise2( ) method
      }
}
```

1) Write code to calculate the area of a circle, where the radius (double value) is specified by the user. Example (note the result is printed to 2 decimal places - see String.format :

```
 12
>> A circle with radius 12.0 has an area of 452.39 units
```

The Math.PI variable can be used to specify π

2) Write code to print all of the even numbers from 1 to n, where n is specified by the user. Example:

```
22
>> Even numbers from 22 : 2 4 6 8 10 12 14 16 18 20 22
```

Loops are useful for this question. The following is an example of a while loop that counts down from 10 to 0.

```
public class LoopDemo{
      public static void main(String[] args){
            int k = 10;
            System.out.println("Blast off in..");
            while( k >= 0){
                  System.out.println(k);
                  k = k - 1;
            }
            System.out.println("Zooooom");
      }
}
```

3) Write code to randomly generate two numbers, print a question, and produce the answer in the form:

```
>>What is the product of 983 and 828 ?
>>The answer is 813924
```

Random numbers can be generated using the Random class. You will need to import java.util.Random. Here's a sample program that generates a random number from 0 up to 9

```
import java.util.Random;
public class RandomDemo{
      public static void main(String[] args){
            Random r = new Random();
            int someRandomInt = r.nextInt(10);
            System.out.println(someRandomInt);
      }
}
```

4) Write code to simulate a chat with the user where the user exits the chat by typing exit. A series of preset words are randomly chosen from an array of length 3. Example:

```
>>hi
hi
>>howdy-doo
hello
>>hey
ok
>>hey
bye
>>howdy-doo
exit
```

An array in Java is declared and initialised as follows:

variableType[ ] arrayVariableName ; // declaration

arrayVariableName = new variableType[arrayLength]; // initialisation.

Examples:

```
int[] marks = new int[100]; // array that can hold 100 int variables
String[] words = new String[25]; //array that can hold 25 Strings
```

Assignment of values to array locations:

```
marks[0] = 100;
words[10] = "Red";
```