## Lab Assignment:  Exploring Iteration

The purpose of this lab is for students to see how iteration allows a program to perform a task and the effects of the various iteration parameters.

1. Get the file `Iteration_Explore.java` and open it in BlueJ.

2. Compile and run the program as it is written.

3. What is the program computing?

4. How many times does the program ***iterate*** ("loop")?

5. Look at the structure of the block of code below:

```
sumInt = 0;
for(int i = 1; i <= 10; i = i + 1) {
    sumInt = sumInt + i;
       square = i * i;
    System.out.println(i + "\t" + sumInt + "\t" + square);
}
```

Based on the behavior of the program, describe what you think is the function of each part below:

`for`

`int i = 1;`

`i <= 10;`

`i = i + 1`

`sumInt = sumInt + i;`

`square = i * i;`

6. What is the purpose of the statement `sumInt = 0;` before the `for` statement?  Try commenting out `sumInt = 0;` and see what happens.

7. There are some tasks in programming that are so common, developers of programming languages have created shortcuts for them. One of these tasks is incrementing (or decrementing) a number. In the program, change the following statements and run the program again.

Change `i = i + 1` to `i++` and change `sumInt = sumInt + i;` to `sumInt += i;`

8. Did anything change about how the program ran? What does this tell you about the meaning of the expressions `i++` and `sumInt+= i`? We will look at these operations in depth later.

9. What if you wanted the program to add more numbers, such as 1 to 100? What would you need to change? Try it.

10. What if you wanted the program to start adding at a number other than 1, such as 25 to 100. What would you need to change? Try it.

11. What if you wanted the loop to count backwards, say from 100 to 1? Answer the questions and try it.

    What would be the starting value of `i`?

    What logic would you need to make the loop stop?

    How would you make the value of `i` decrease instead of increase?

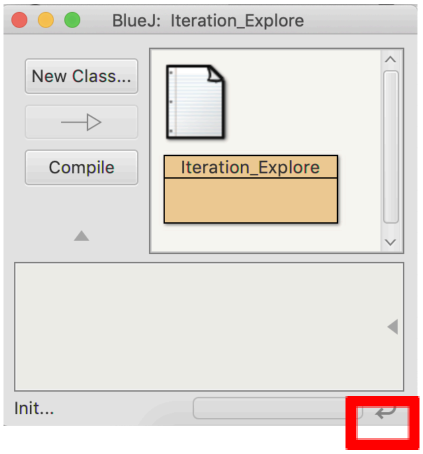
12. How could you make the program count by a number other than by one? Think about how the `+=` operator might work for `i`. For example, try `i += 2` instead of `i++`.

13. What if you wanted only the even values of `i` to print? Try it.

14. Try changing the statement `i <= 10` to `i >= 10` and run the program. What happens? Why might this behavior occur?

15. Try changing the statement `i++` to `i = i` and run the program. What happens? Try to explain why this happened.

    Note: To make it stop, click the ↩ button in the lower right of the BlueJ window to reset the program.



16. Modify the program so that it counts from 100 to 1000 by 25. Verify that the value of `sumInt` at 1000 is `20350`.

17. Below the existing `for` loop, create a new loop that computes the factorial of all numbers from 1 to 10 and prints them in a table similar to the table of sum and squares. Note that the factorial of a number, **n!** is the product of the number and each natural number below it. For example, 5! = 5 * 4 * 3 * 2 * 1 = 120. Of course, the order of multiplication is commutative, so you could just as easily write

5! = 1 * 2 * 3 * 4 * 5. Note also that 0! = 1.

- Use `j` for the index (counter) in the `for` loop.
- You will need to create an appropriate variable for storing the value of the factorial for each iteration.
- Do you need to initialize that variable? If so, to what?

```
//Iteration_Explore.java

/**
 * Lab that explores for loops.
 *
 * @author Roy Chancellor
 * @version 2/7/2019
 */
public class Iteration_Explore
{
    public static void main( String[] args) {
        int i, sumInt, square;

        System.out.println();  //blank line for separation in terminal
        System.out.println("i\tsumInt\tsquare i");  //table header

        sumInt = 0;
        for(i = 1; i <= 10; i = i + 1) {
            sumInt = sumInt + i;
            square = i * i;
            System.out.println(i + "\t" + sumInt + "\t" + square);
        }  //end for

        //your for loop for the factorial will begin on the next line


    } //end main
}  //end Iteration_Explore
```