Iteration (Loops)

Iteration or looping is used when the program needs to repeat the same operations a number of times. In this class, we will use two types of loops, the For loop and the While loop, that can cause statements to be repeated.

For Loops (Determinate Loops)

A **for loop** is sometimes called a determinate loop because we use this type of loop when we know how many times we want to repeat the statements.

The for loop is controlled by a variable, often called the index variable or the count variable, and it will use this variable to count the number of times to execute. In Java, this loop has the following form:

```
for (initialization; termination; increment) {
        [some statements]
}
```

- The initialization part should declare the index variable with an initial value.
- The termination part should give a condition, evaluating to true or false
  - When the condition is true, the for loop keeps executing
  - When the condition is false, the for loop terminates
- The increment part is assigns a new value to the index variable after each iteration of the loop

.

Typical for loop, iterating over the numbers from 1 to 10:

```
for (int i = 1;  i < 11; i= i + 1) {
        [ some statements using the index variable i ]
}
```

Or using the ++ notation for adding 1 to the variable i:

```
for (int i = 1;  i < 11; i++) {
        [ some statements using the index variable i ]
}
```

Example:
Suppose.that we want to sum the numbers from 1 to 100:

```
int sum = 0;
for (int i = 1;  i <= 100; i++) {
        sum = sum + i;
}
```

Note that the “increment” part can add other numbers besides 1, as in this example which sums the odd numbers between 1 and 15:

```
int oddsum = 0;
for (int i = 1;  i <= 15; i= i + 2) {
        oddsum = oddsum + i;
}
```

Although people often use the single letter variables i, j, or k for index variables, the variable can be named anything.

```
for (int index = 1;  index <= 10; index++) {
        sum = sum + index;
}
```

While Loops (Indeterminate Loops)

The while loop is sometimes called an indeterminate loop because we don’t necessarily know how many times we are going to execute the loop.

The while loop has the form:

```
while (condition) {
        [some statements]
}
```

- The condition can be anything that evaluates to true or false, and it is evaluated at the beginning of every iteration of the loop.
    - When the condition is true, the next iteration of the loop is executed.
    - When the condition is false, the loop terminates.

Example:

Suppose that we want to see how many integers we have to sum in order to sum up to a 1000:

```
int sum = 0;
int i = 1;
while (sum < 100) {
        sum = sum + i;
        i++;
}
// the value of i is the final number of those that were summed
```