Iteration

Tags	Completed

Resources

- **Vocab**
  - Looping

    The repetition of code until some condition terminates the loop
  - Loop update

    One or more statements that could cause the loop condition to evaluate to false
  - Loop termination condition

    An event that causes loop condition to evaluate to false
  - Iteration

    One execution of the loop body/pass
  - Sentinel value

    A special value that causes the loop condition to be false
- **Labs**

  Lab 12 (Logic)

```
count = 0;
initialize variables
prompt for input
retrieve input

while (count < 5) {
  if (input < 0 || input > 100) {
    prompt bad input
    ask for new input
    retrieve input
```

```
  } else
    count++;
    ask for another valid input value
    retrieve input

    if (count == 1) {
      max = input;
      min = input;
    } else {
      if (input < min) {
        input = min;
      }
      if (input > max) {
        input = max;
      }
      prompt for another valid input
      retrieve input
    } //end of count if/then loop
      retrieve max, min, and average
  } //end of main if/then loop
} //end of while loop
```

Notes

▼ Priming Read

```
input = scan.nextInt();
```

▼ While Loop

Use if you do not know how many times the loop will run

```
while (condition) { //may execute 0 times
  top of loop
  loop body
  bottom of loop
}
```

a. initialize the variable (prior to loop)

b. test the loop variable in the header

c. update the loop variable

▼ Example

```
import java.util.*;

public class WhileExample {
    public static void main(String[] args) {
        final int SENTINAL = -1;
        int input;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter an integer");
        input = scan.nextInt(); // Priming read

        int sum = 0;
        int count = 0;
        int min;
        input = scan.nextInt(); // Priming read
        min = input;
        while (input != SENTINAL) { // Pre-Test
            count++;
            sum += input;
            sum += sum;
            System.out.println("Enter next number");
            input = scan.nextInt(); // Priming read

            if (input < min && input != SENTINAL) {
                min = input;
            }
        }

        System.out.println("You entered " + count + " numbers");
        System.out.println("The sum is " + sum);
        System.out.println("The min is " + min);
    }
}
```

▼ For Loop

Use to control conditions of loop

```
for (initialize; test condition; change) { //loop header
  loop body
}
```

▼ Example 1

```
public class ForExample {
    public static void main(String[] args) {
        for (int i = 0; i <= 3; i++) {
            System.out.println(i);
```

```
        }

        //System.out.println(i);
    }
}
```

▼ Example 2

```
public class ForExample {
    public static void main(String[] args) {
        for (int i = 1; i <= 3; i++) { //Row
            System.out.println("This is row " + i + ":");

            for (int j = 1; j <= 7; j++) { //Column
                System.out.print(j + " ");
            }

            System.out.println();
        }
    }
}
```

▼ Last Loop

```
do { //event controlled loop (post-test)
  loop body
} while (condition);
```

▼ Brief Outline (Multiplication Table)

```
import

public class MultiTable {
  public static void main (String[] args) {
    initialize scanner

    int amax = scan.nextInt();
    int bmax = scan.nextInt();

    create input

    print x
```

```
    for (int a = 1; a <= amax; a++){
      print a
      for (int b = 1; b <= bmax; b++){
        print b
      }
    }
  }
}
```