CSE 215L: Programming Language II

Week 03: Method and Loops

Repetition Structures

For loop

```
for (int number = 5; number <= 20; number++){
    System.out.print(number + " ");
}
```

While loop

```
int number = 5;                // initialization (executed only once)
while (number<=20) {                  // checking the condition
      System.out.print(number + " ");          // loop body (statement)
      number++;                // iteration
}
```

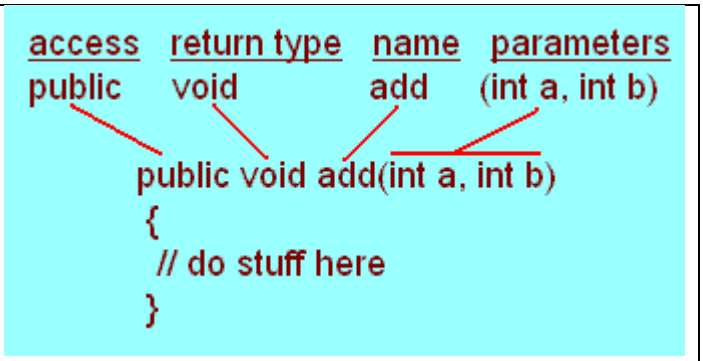
Do while loop

```
int number = 5;  // initialization
do {
      System.out.print(number + " ");   // statement(s)
      number++;             // iteration
} while (number <= 20);        // Checking condition
```

Java Methods

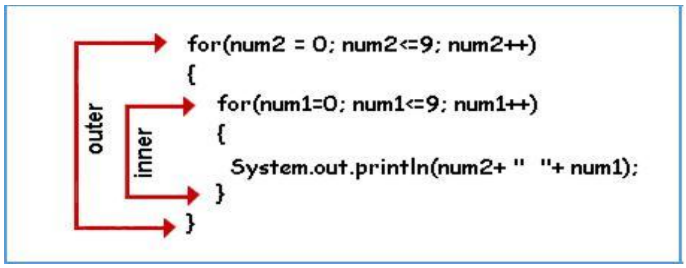
A simple Java method requires a minimum of three items:

Visibility : public, private, protected
Return Type: void, int, double, (etc.)
Name: whatever you want to call the method



Example 1: Write two overloaded methods which will compute average of two integers and three double numbers, respectively.
public static double computeAverage(**int** a, **int** b)
public static double computeAverage(**double** x, **double** y, **double** z)

Nested loop



The outer loop is changed only after the inner loop is completely done or interrupted.

Lab Tasks:

1. Write a method **Combination**(int,int) to compute combination using the following formula:

$$n_{C_r} = \frac{n!}{r!(n-r)!}$$ both n and r are integer inputs here. (n>=r)

You'll need another method **Factorial**(int) to simplify this problem.

n! = 1*2*3*…….*n

```
Enter values of n and r : 6   4
Combination: 15
```

2 Write a void method which will take an integer as parameter and print the following pattern. If the parameter is 4, then your method should display the following:

4 3 2 1

4 3 2

4 3

4

```
public static void printPattern (int n)
```

3. The user should be allowed to enter one-digit positive integers until (s)he enters a negative integer. Convert all the valid inputs into a **positive whole number** and print it.

Enter one-digit integers (enter negative to stop):

4

8

1

5

-3 Whole number: 4815