Java

Pink Programming | GameDevCamp

== and equals

- The operator “==” works a bit different on **objects**
  - == - will say if they are the same
  - **equals** - will check if they are logically equal
- Example

```
String a = new String("Forest");
String b = new String("Forest");
String sameAsA = a;
boolean r1 = (a == b);           //false, not the same object
boolean r2 = a.equals(b);        //true, logically equal
boolean r3 = (a == sameAsA);     //true, are the same object
```

Arrays

- object - needs to be declared and created
  - `int[] array; // a variable that will hold an array of integers`
  - `array = new int[5]; // creates an array object that can hold 5 integers`
- how do we create an array that can hold 3 words?
  - `String[] words = new String[3];`
- check the size of an array
  - `System.out.println(words.length);`
- access to the array
  - `words[0] = "Forest"; // observe arrays in Java are 0-based`
  - `words[1] = "Pink";`
- `String[] words = {"Pink", "Programming", "Game Dev"}; //alternative`

For loops

```
String[] words = {"Pink", "Programming", "Game Dev"};
for (int i = 0; i < 2; i++) {
     System.out.println(words[i]);
}
```

- **three sections**
  - 1. runs once the loop is entered
  - 2. if the statement returns true, run the code between {}, else exit
  - 3. runs after each loop

- **What is the output?**

While-loops

```
String[] words = {"Pink", "Programming", "Alpine"};
int i = 1;
while (i < 3) {
     System.out.println(words[i]);
     i++;
}
```

- What is the output?

Function

Example:

```
public static void main(String[] args) {
     int a = 3;
     int b = 5;
     int result;
     Calculator c = new Calculator();
     result = c.add(a,b); // add is a function
     result = c.subtract(a,b); // subtract is a function
}
```

Function

How do we create a function?

```
public class Calculator {
     public int add(int first_val, int second_val) {
          int result = first_val + second_val;
          return result;
     }
}
```

- **public:** the visibility of the method *(private, protected)*
  int: return type *(Anything! I.e. String, double, void)*
  add(**type** p1, **type** p2): name of the function, declare arguments
  return: the output from using the function

Function

Our complete Calculator class

```
public class Calculator {
     public int add(int first_val, int second_val) {
          int result = first_val + second_val;
          return result;
     }
     public int subtract(int first_val, int second_val) {
          int result = first_val - second_val;
          return result;
     }
}
```

Function

Example:

```
public static void main(String[] args) {
     int a = 3;
     int b = 5;
     int sum;
     int difference;
     Calculator c = new Calculator(); // the functions exist in Calculator class,
to call the functions we have to create an object of this class.
     sum = c.add(a,b); // call the add function in class Calculator
     difference = c.subtract(a,b); // call the subtract is a function
}
```