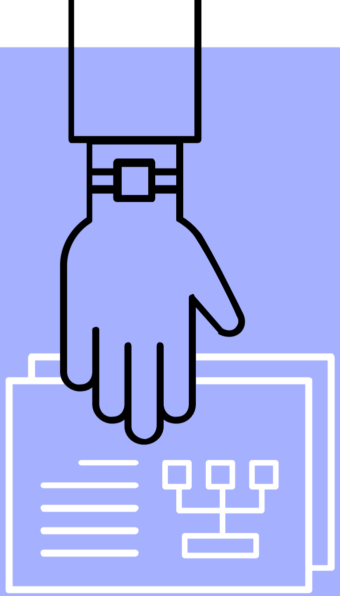
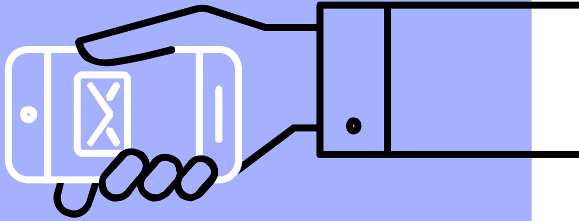
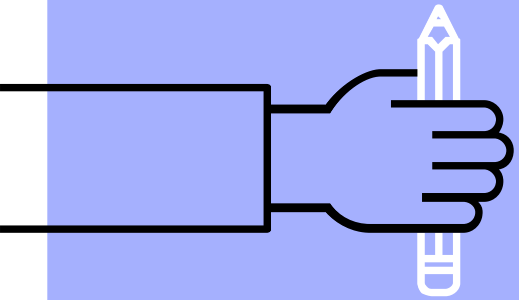
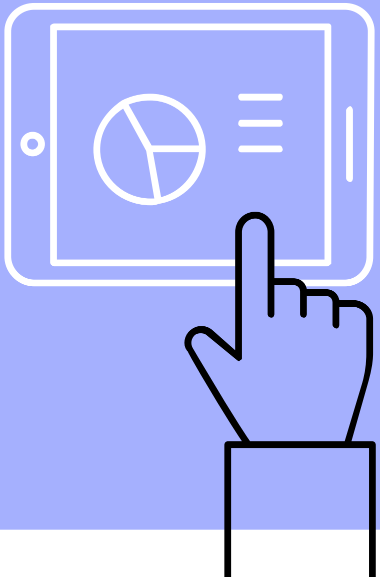




11/18 JAVA REVIEW

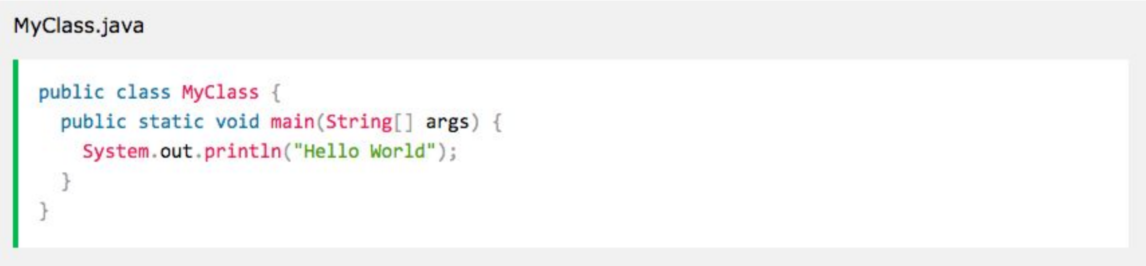




Classes

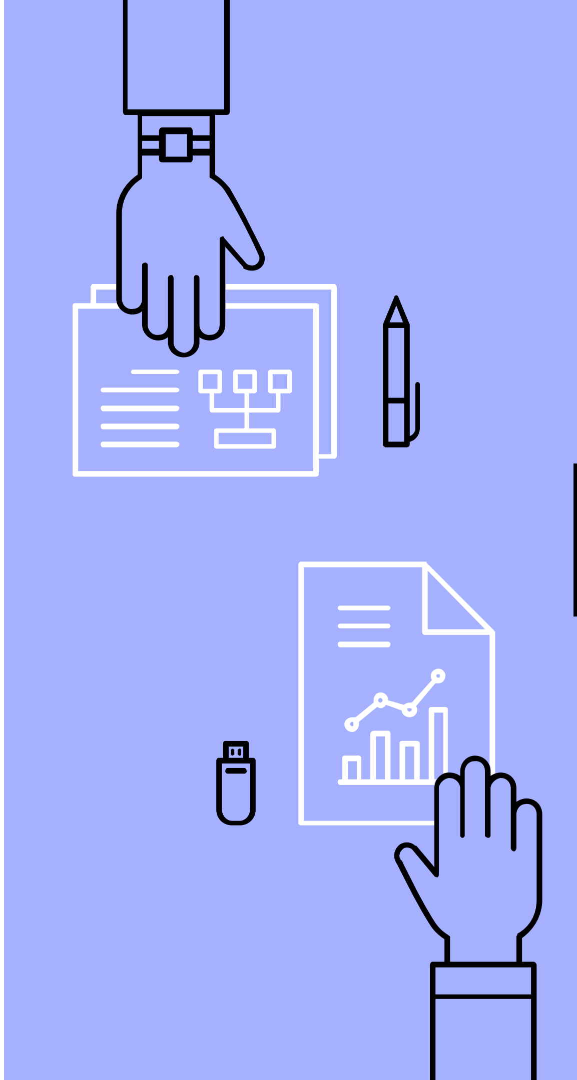
Every line of code that runs in Java must be inside a class. Java Naming convention for classes: Every class name starts with an uppercase letter. Every new word also starts with an uppercase letter. No spaces. (ex. MyClass)

The name of the Java File must match the class name. Save java files with the following filename: ClassName.java (Ex. MyClass.java)

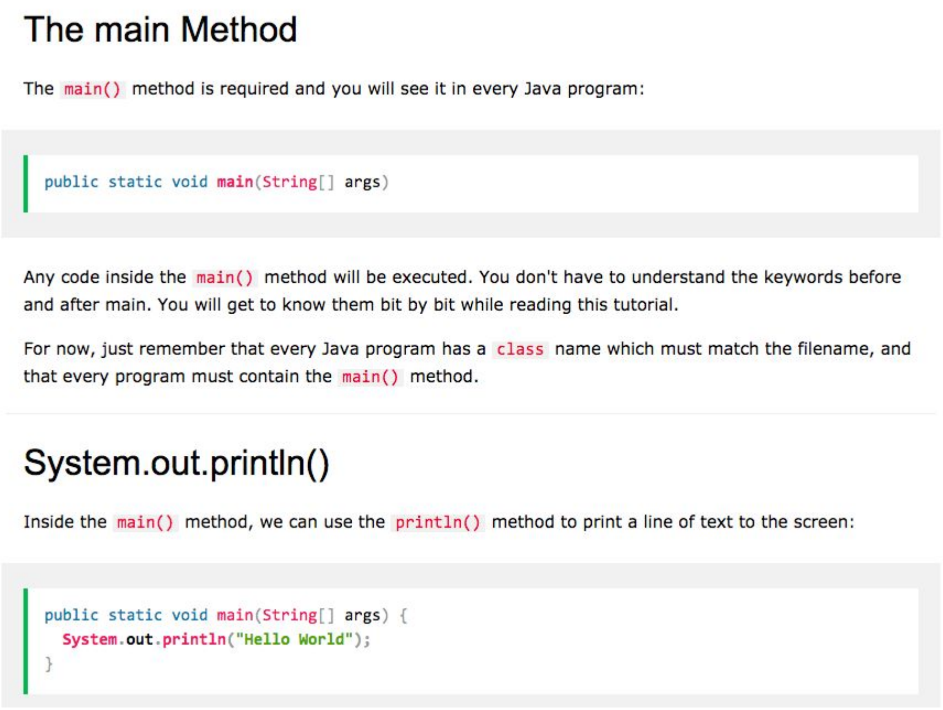


MyClass.java

```
public class MyClass {
  public static void main(String[] args) {
    System.out.println("Hello World");
  }
}
```



Main Method & Printing Text!



The main Method

The `main()` method is required and you will see it in every Java program:

```java
public static void main(String[] args)
```

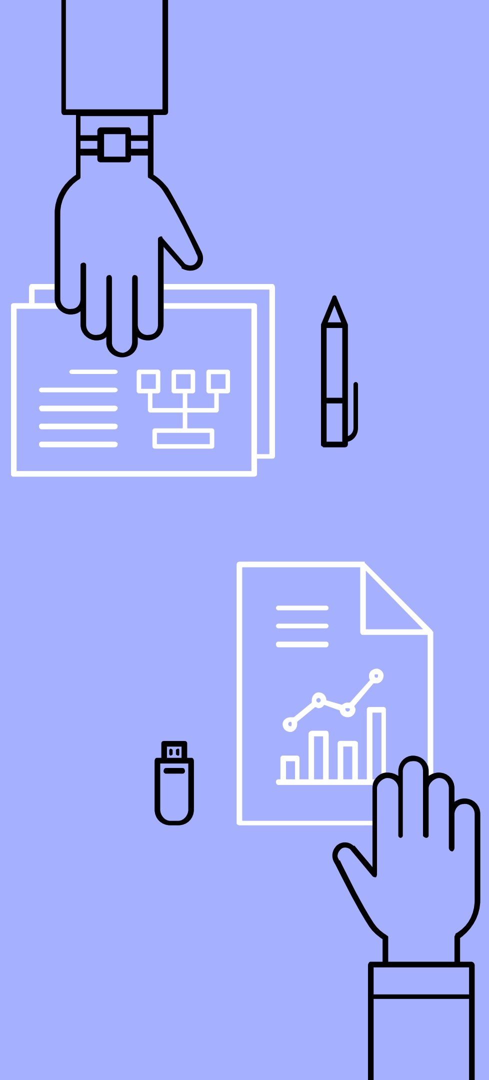
Any code inside the `main()` method will be executed. You don't have to understand the keywords before and after main. You will get to know them bit by bit while reading this tutorial.

For now, just remember that every Java program has a `class` name which must match the filename, and that every program must contain the `main()` method.

System.out.println()

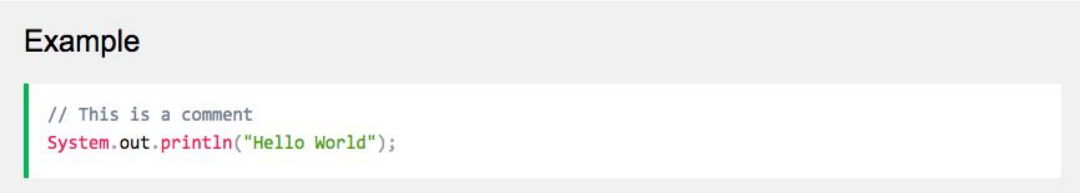
Inside the `main()` method, we can use the `println()` method to print a line of text to the screen:

```java
public static void main(String[] args) {
  System.out.println("Hello World");
}
```



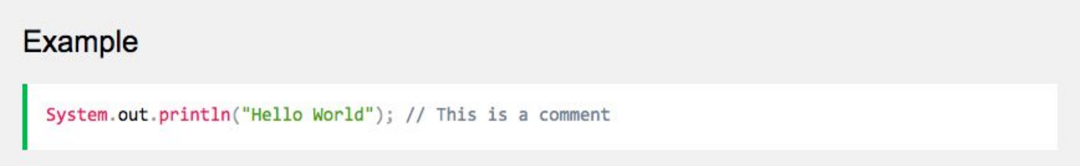
Comments

//Single Line Comments



Example

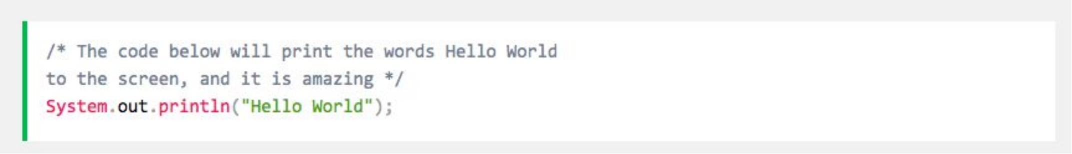
```
// This is a comment
System.out.println("Hello World");
```



Example

```
System.out.println("Hello World"); // This is a comment
```

/* Multi Line Comments */

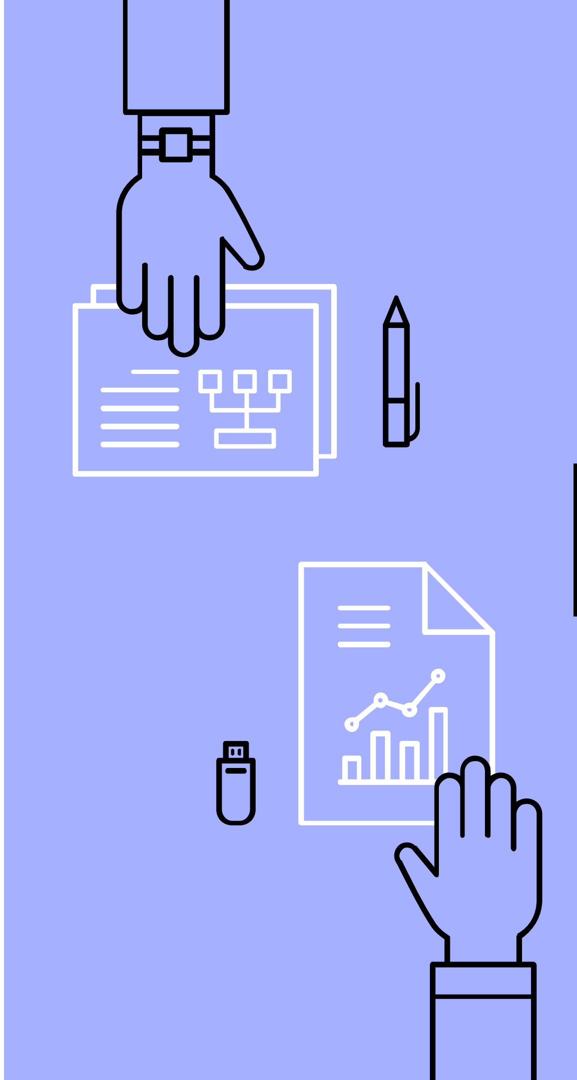


```
/* The code below will print the words Hello World
to the screen, and it is amazing */
System.out.println("Hello World");
```

Variables

Common Variable Types (Primitive Data Types)

- String - stores text, such as ”Hello”. String values are surrounded by double quotes
- int - stores integers (whole numbers), without decimals, such as 123 or -123
- double - stores larger number, with decimals, such as 1.99 or -1.99
- char - stores single characters, such as ’a’ or ’B’. Values are surrounded by single quotes.
- boolean - stores values as true or false



Initializing and Declaring Variables

```
//declaring and initializing variables
type variableName = value;
```

```
type variableName; //declaring variables
variableName = value //initializing variables
```

Java Variable Naming Convention: camelCase

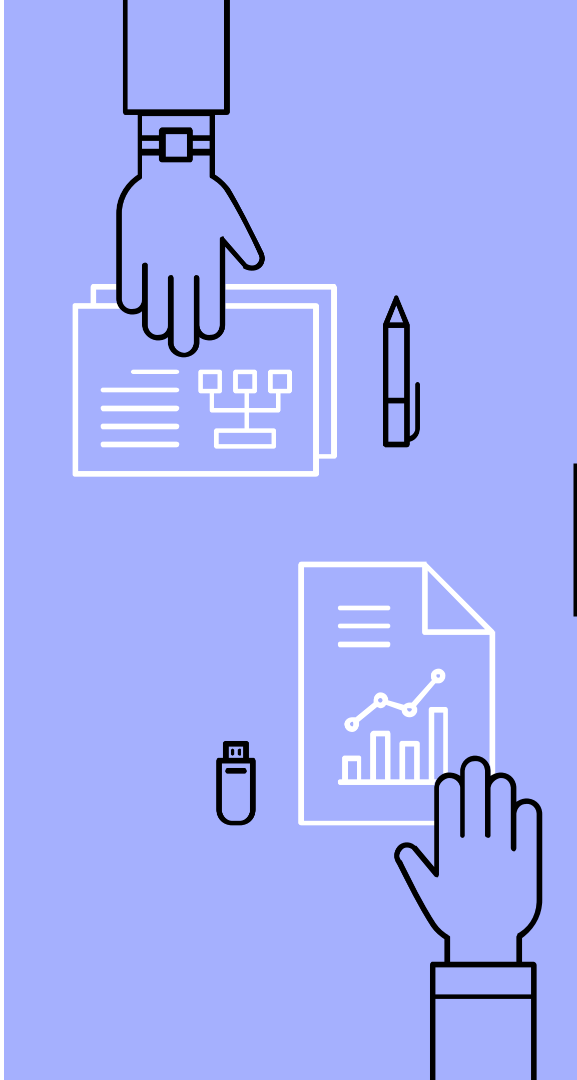
Start with a lowercase letter, and add an uppercase letter at the beginning of every new word. No spaces. (ex. variableName)

Constants

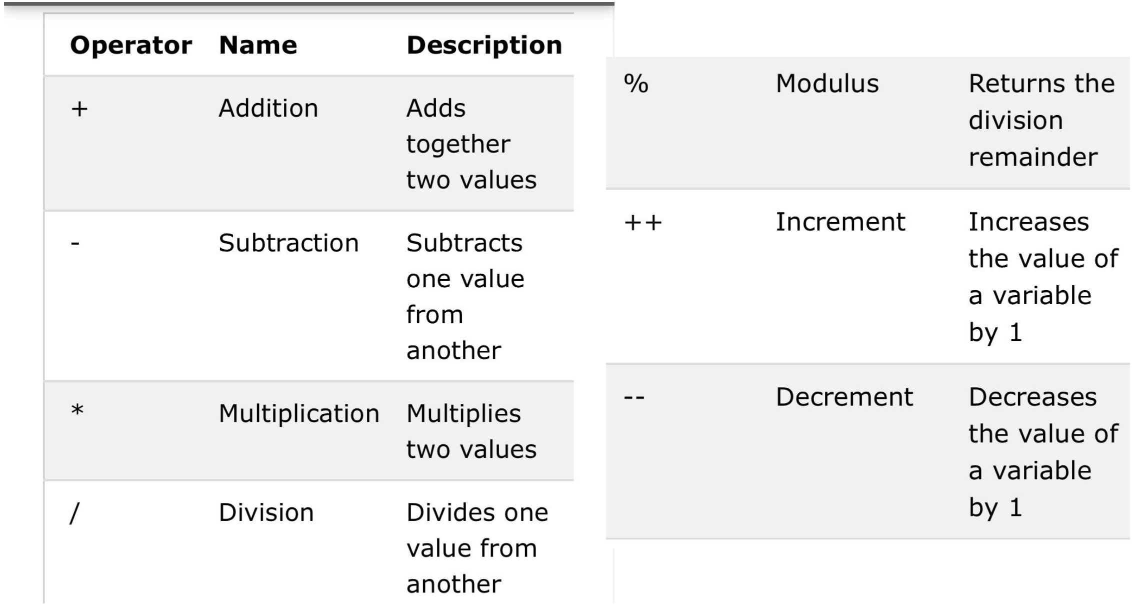
```
// initializing and declaring a constant
final type CONSTANT_NAME = value;
```

A constant is the opposite of a variable – it doesn't vary. The word final indicates that a value is set as a constant.

Java Constant Naming Conventions: Letters in all caps, separated by underscores (_). No spaces



Operators



Operator	Name	Description
+	Addition	Adds together two values
-	Subtraction	Subtracts one value from another
*	Multiplication	Multiplies two values
/	Division	Divides one value from another
%	Modulus	Returns the division remainder
++	Increment	Increases the value of a variable by 1
--	Decrement	Decreases the value of a variable by 1