The Two Operation Types

What is an operation type:

In general, when dealing with computers all operations fall into one of two broad categories: Integer and Floating Point.

Integer Operations generally:

- Are faster
- Require less memory
- Are less CPU intensive

Floating Point Operations generally:

- Take advantage of a hardware sub-processor to speed them up
- Are used any time graphics are used on a modern computer
- May also be used when processing sound

Why does this matter?

The short answer is that it matters because it will affect your programs!

It will affect your programs in two primary ways. First, Integer math is faster, and the `int` primitive type requires less memory to store. Secondly, you will get very unexpected results from mixing `int` and `double` types in your mathematical expressions if you don't understand how they work.

For the most part, you only need to worry about the distinction between the two types of math when dealing with division!

For example, what is 5 divided by 2?

The answer depends on which type of operation or math you will be using!

Integer Math:

If both sides of the operator (dividend and divisor) are of type `int` then Integer math will be used. So:

```
int a=5/2;
```

Only the Integer portion of the answer is kept so the variable `a` has the value `2`.

Floating Point Math:

If either side of the operator (dividend **or** divisor) are of type `double` then floating point math will be used. So:

```
double b=5/2.0;
```

The dividend is an int literal but the divisor is a double literal so the floating point math is used resulting in the variable `b` having the value `2.5`.

Examples:

```
double c=5/(double)2;
double d=5.0/2;
double e=(double)5/2;
double f=5.0/2.0;
```

In each of the examples above the dividend, the divisor, or both as `double` values, so all use floating point math!