Mathematical operators in JavaScript

Common Operators

Javascript supports all the commonly used mathematical operators. Namely `+` `-` `*` `/`.

Example:

```
var a = 2
var b = 3
var c = a + b
var d = a * b
var e = a / b
var f = a - b
```

Output:

```
c = 5
d = 6
e = 0.6666666666666666
f = -1
```

`Modulo` or Remainder Operator:

Many programming languages including JavaScript have a modulo operator `%`. This operator returns the remainder when one variable is divided by another.

Example:

```
var a = 10 % 7
```

Output:

```
a = 3
```

This operator is often useful when you want to check if a number is odd or even.

Example:

```
var a = 10 % 2
var b = 11 % 2
```

Output:

```
a = 0
b = 1
```

Try this out for yourself, any even number `%2` returns 0 while any odd number `%2` returns 1.

`Exponentiation` operator:

This operator is represented by `**`. This returns the value of the first operand raised to the power of the second operand. For example 2^4 = 16.

Example:

```
var a = 2 ** 4
var b = 3 ** 2
var c = 10 ** 1.5
```

Output:

```
a = 16
b = 9
c = 31.622776601683793
```

String concatenation

A special property of `Strings` is that they can be combined or concatenated with one another.

Example:

```
var word1 = "Welcome"
var word2 = "Masai"
var word3 = word1 + " to " + word2 + " school!"
console.log(word3)
```

Output:

```
Welcome to Masai school!
```

Strings can also be combined with other types like `numbers`.

Example:

```
var num1 = 1
var num2 = 2
var output = "1 + 2 = " + (num1 + num2)
console.log(output)
```

Output:

```
1 + 2 = 3
```

Note: Notice the circular brackets between `num1 + num2` this tells javascript that we want to add the two numbers mathematically. Without the brackets the output would be `1 + 2 = 12`.

`Booleans`:

The last data type we are going to learn about is a `Boolean`. This data type has only two values `true` and `false`.

Example:

```
var x = true
var y = false
```