**JavaScript Review**

One action, or method, that is built into the console object is the .log() method. When we write console.log() what we put inside the parentheses will get printed, or logged, to the console.

The semicolon denotes the end of the line, or statement.

***A*single line comment** will comment out a single line and is denoted with two forward slashes // preceding it.

// Prints 5 to the console

**Multiple lines**

/\*

This is all commented

console.log(10);

None of this is going to run!

console.log(99);

\*/

**Data Types**

In JavaScript, there are seven fundamental data types:

* **Number**: Any number, including numbers with decimals: 4, 8, 1516, 23.4
* **String**: Any grouping of characters on your keyboard (letters, numbers, spaces, symbols, etc.) surrounded by single quotes: ' ... ' or double quotes " ... ". Though we prefer single quotes. Some people like to think of string as a fancy word for text.
* **Boolean**: This data type only has two possible values— either true or false (without quotes). It’s helpful to think of booleans as on and off switches or as the answers to a “yes” or “no” question.
* **Null**: This data type represents the intentional absence of a value, and is represented by the keyword null (without quotes).
* **Undefined**: This data type is denoted by the keyword undefined (without quotes). It also represents the absence of a value though it has a different use than null.
* **Symbol**: A newer feature to the language, symbols are unique identifiers, useful in more complex coding. No need to worry about these for now.
* **Object**: Collections of related data.

**Arithmetic Operators**

1. Add: +
2. Subtract: -
3. Multiply: \*
4. Divide: /
5. Remainder: %

**String Concatenation**

When a + operator is used on two strings, it appends the right string to the left string:

The computer will join the strings exactly, so we needed to make sure to include the space we wanted between the two strings.

console.log('Hello' + 'World');

console.log('Hello'+' ' + 'World');

**Properties**

Every string instance has a property called length that stores the number of characters in that string. You can retrieve property information by appending the string with a period and the property name:

//get the number of letters in a sentence

console.log('Teaching the world how to code'.length);

**Methods**

Remember that methods are actions we can perform. JavaScript provides a number of string methods.

We call, or use, these methods by appending an instance with:

* a period (the dot operator)
* the name of the method
* opening and closing parentheses

E.g. 'example string'.methodName()