DATA TYPES

DATA TYPES

String	`"hello"`	
Number	`2`	
Boolean	`false`	**Primitive**
Undefined	`undefined`	
Null	`null`	

Object	`{property: "value"}`	
Array	`[1,2,3]`	
Date	`new Date()`	**Object**
RegExp	`/.*/g,`	
Function	`function(){}`	

STRING

- String - A series of characters enclosed by an apostrophe (') or quotation mark(")

- Concatenation - Combines or "adds" strings together

- .length - Returns the number of characters in the string, including 0 for an empty string

```
var fullstack = "Fullstack";
var academy = " Academy";
fullstack + academy // "Fullstack Academy
fullstack.length // 9
```

NUMBER

- **Used to perform arithmetic operations**

- **Common Arithmetic Operators**

  - Addition   `5+5;`

  - Subtraction   `8-5;`

  - Multiplication   `7*7;`

  - Division   `7/7;`

BOOLEAN

- A binary data type with two possible values:

```
true
```

```
false
```

NULL AND UNDEFINED

- Undefined - has not been given a value *yet* `undefined`

- Null - has no value `null`

- Not important for now…but be careful not to confuse the two!

FUNCTION

- **A function is a reusable chunk of code**

- **A function is similar to a recipe**

Baking Recipe Example - Cookie Recipe

- **Choose a flavor**
  - Peanut butter
  - Chocolate
  - Caramel
- **Make the batter**
- **Pre-heat the oven**
- **Add the cookies to a baking sheet**
- **Bake for 25 minutes**

```javascript
function bakeCookies(flavor, servings){
  var flavoring = flavor;
  var batter = flavoring + servings;
  preHeatOven();
  addCookiesToSheet();
}
```

```javascript
bakeCookies('chocolate', 5);
```

```javascript
bakeCookies('vanilla', 10);
```

ARRAY

- **An Array is a value that stores multiple values.**

- **Arrays are similar to a list of values.**

- **Each value stored in an array is separated by a comma.**

Opening bracket

Closing bracket

```
[true, "true", false, 22]
```

OBJECT

- Objects use a combination of "properties" and "values," known as key/value pairs, to store data.

- An Object is similar to a dictionary entry, where the "property" represents the word, and the "value" would represent its definition.

- You can think of objects like a group of keys: Once you have the key, you'll be able to access its associated value.

PROPERTY   VALUE

```
{cattywampus: "askew, awry"}
```

COMMENTS

```javascript
// I'm a single line comment
var fullstack = "Fullstack"; // I'm also a comment

/* I'm a multi-line
comment. I can be as long as you need!
*/

console.log(fullstack);
```