**Objects**

**Intro to Objects**

Objects are a powerful feature of JavaScript.

Objects allow us to wrap up pieces of related data and functionality, into one single container.

Objects are not just limited to arrays. JavaScript comes with a whole set of other useful built-in objects.

**Objects in Code**

One way to define an object is to create a variable and assign it to a pair of empty curly braces:

var umbrella = {};

console.log(typeof umbrella);

Objects might have properties. To add properties on it, we can define key value pairs for each piece of data:

var umbrella = {

color: "pink",

isOpen: false

};

console.log(umbrella);

An object can also have (do) a method. A method is just a function associated with an object.

var umbrella = {

color: "pink",

isOpen: false,

open: function() {

if (umbrella.isOpen === false) {

umbrella.isOpen = true;

return "Open the umbrella!";

} else {

return "The umbrella is open!";

}

}

};

console.log(umbrella.isOpen, umbrella.open(), umbrella.isOpen);

**Objects**

There are primitive and objects data types:

* Primitive: strings, numbers, booleans, undefined, and null.
* Objects: data structure in JavaScript that stores data about a particular thing, and uses keys to keep track of.

For example, an object can be used to represent a person:

var person = {

name: "Fernando",

dob: "14/02/1992",

sex:"male",

pets: false,

siblings: ["Lucas","Maisa","Jonatas"]

};

To access any of the key value pairs, we can use either bracket or dot notation:

console.log(person["name"], person.name);

**Naming Conventions**

* Don’t need to use quotes for property names.
* Don’t use numbers as the first character of property names.
* Don’t use spaces or hyphens for property names. Instead, use camelCase.

All the rules above apply for variable names.

**Summary of Objects**

Objects are one of the most important data structures in JavaScript. They have properties (information about the object) and methods (functions or capabilities the object has).