VANILLA JAVASCRIPT

Everything in Javascript is an OBJECT.

OBJECTS can be simple.

```
var today = 9;
```

(a Number - either Int or Float)

```
var whatIsToday = function(){
  console.log('Tuesday');
};
  (a function)
```

OBJECTS can also be complex. Complex Objects are made up of Properties and Methods.

KEY-VALUE PAIRS

(the properties)

A **KEY** is a unique identifier for some sort of data.

A **VALUE** is the data you are storing with the key.

ACTIONS

(the methods)

An ACTION is a function.

```
var review = {
   shush : function(){
     console.log('Quiet! I'm studying!');
   }
};
```

review.type

will return "Vanilla Javascript"

review.shush();

will run the function shush in the review object.

A variable is an **OBJECT** that you define.

They can have GLOBAL or LOCAL scope.

GLOBAL variables can be accessed anywhere in a script.

LOCAL variables can only be accessed within the function where they are defined.

```
var global = "I love everyone!"
function aloneTime(){
  var local = "I need to be alone."
};
```

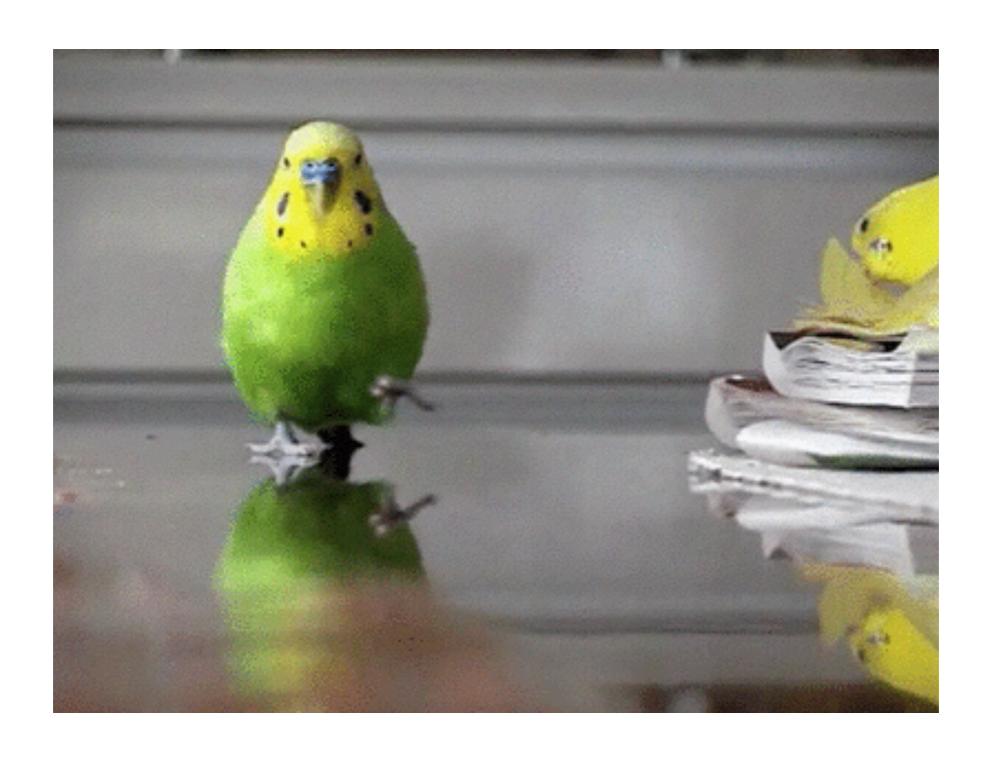
You can pass a local variable between functions by adding it to the **function parameters**.

```
var global = "I love everyone!"
function aloneTime(){
  var local = "I need to be alone."
  tellEveryone(local);
};
function tellEveryone(msg){
  console.log(msg);
};
```

ORDER

Let's talk about ORDER.

<u>ORDER</u>



ORDER

If you reference an object before you define it, this will happen:

<u>ORDER</u>



<u>ORDER</u>

Always structure code from the bottom up (as best you can) to avoid mentioning something that hasn't been defined yet.

Any questions?