JavaScript

The language that powers the web

What all this logos got in common?

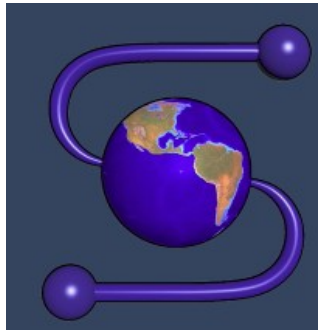
Netscape navegator

skype

facebook

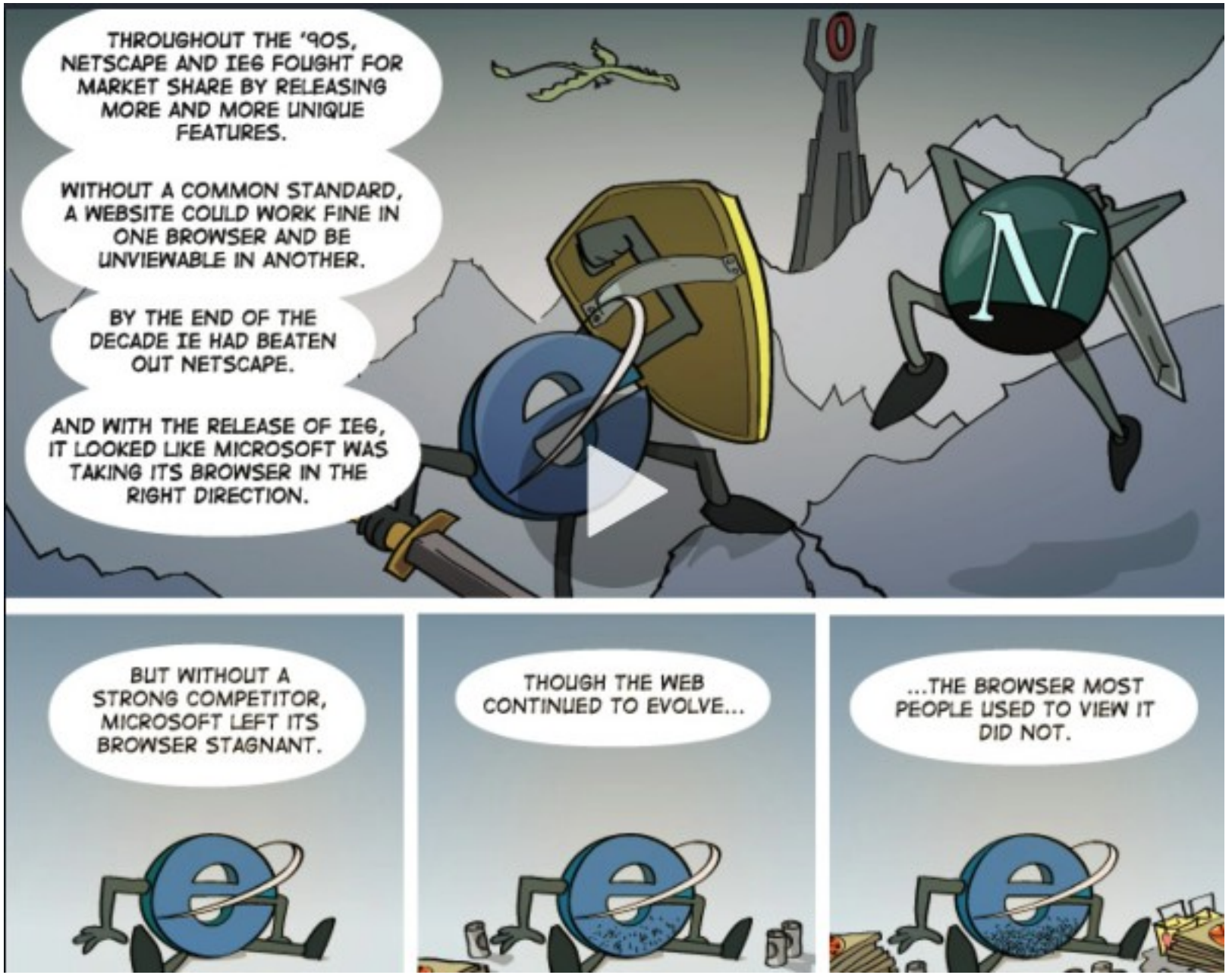
twitter

There was one company that invested in all of these startups, and that's Andreessen Horowitz. And the Andreessen in this case refers to Marc Andreessen who was one of the principle makers of the Netscape.



The browser called Mosaic, Marc worked on this browser before Netscape.

Netscape conquered 80% of the browser usage and it was a war between explore and Netscape.


THROUGHOUT THE '90S, NETSCAPE AND IE6 FOUGHT FOR MARKET SHARE BY RELEASING MORE AND MORE UNIQUE FEATURES.
WITHOUT A COMMON STANDARD, A WEBSITE COULD WORK FINE IN ONE BROWSER AND BE UNVIEWABLE IN ANOTHER.
BY THE END OF THE DECADE IE HAD BEATEN OUT NETSCAPE.
AND WITH THE RELEASE OF IE6, IT LOOKED LIKE MICROSOFT WAS TAKING ITS BROWSER IN THE RIGHT DIRECTION.
BUT WITHOUT A STRONG COMPETITOR, MICROSOFT LEFT ITS BROWSER STAGNANT.
THOUGH THE WEB CONTINUED TO EVOLVE...
...THE BROWSER MOST PEOPLE USED TO VIEW IT DID NOT.

A lot of the technology of Mosaic and Netscape was transmitted to Firefox.

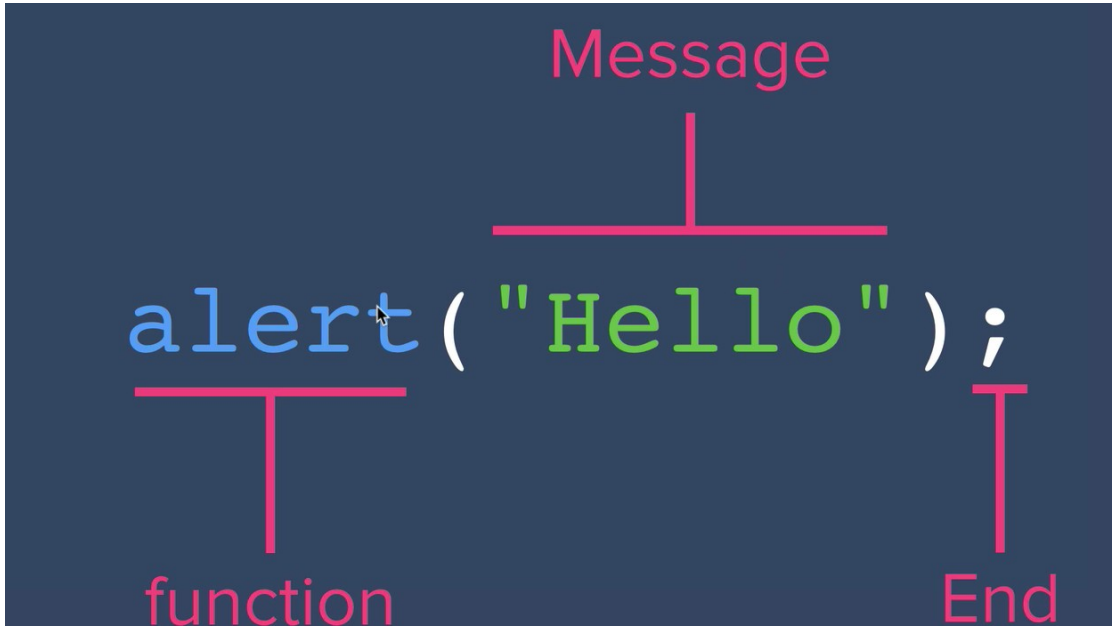
- At the moment, the website was all forms and did not have any function.
- When a website needed some functionality, the website had to send the request to a data server, and it would return the webpage with the data.
- The people of Netscape wanted something dynamic and in order to do this they took away the server and created a scripting language.
- They contracted a guy called Brendan Eich, Brendan created JS in 10 days. Javascript was called liveScript and then Jscript.

Why is called javascript?

- Javascript will tell the website what to do. It will manage the behavior of your website.

  JavaScript snippets

  javaScript Console



Say: “hello”.

```
alert("Hello");
```

Data types

String: string of characters, text.

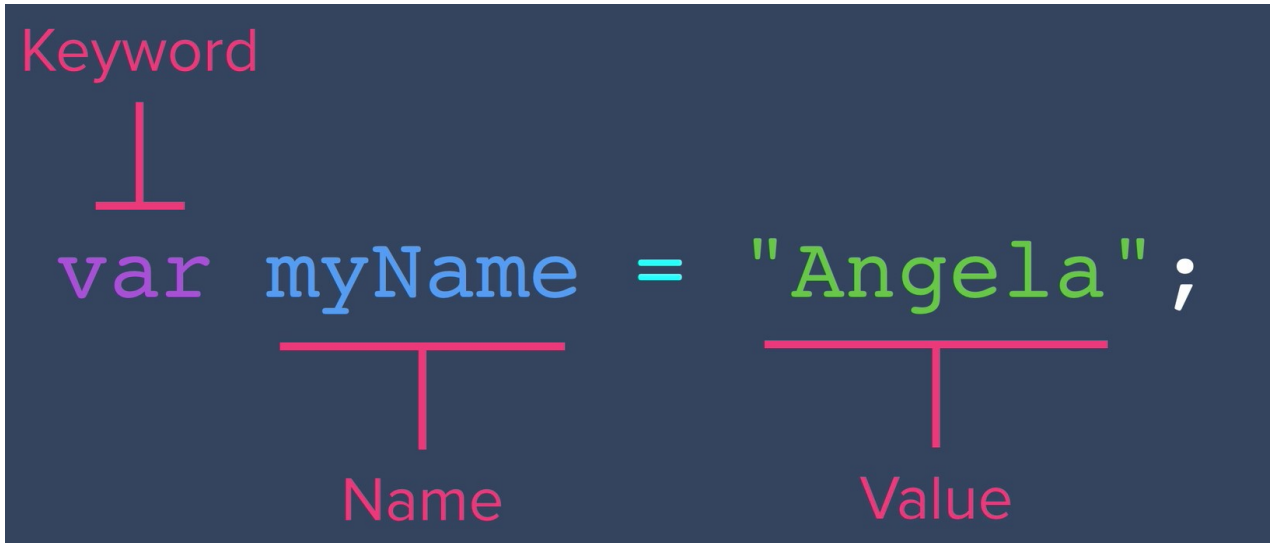
Numbers: 1 2 3

Boolean: describe data that is either true or false

Javascript variable

- Variable: are containers for storing data values.

- Var stands for the word ‘variable’. That means you can change the data. The word var is only use when you are initializing the variable.



Naming and Naming Convention

Rules:

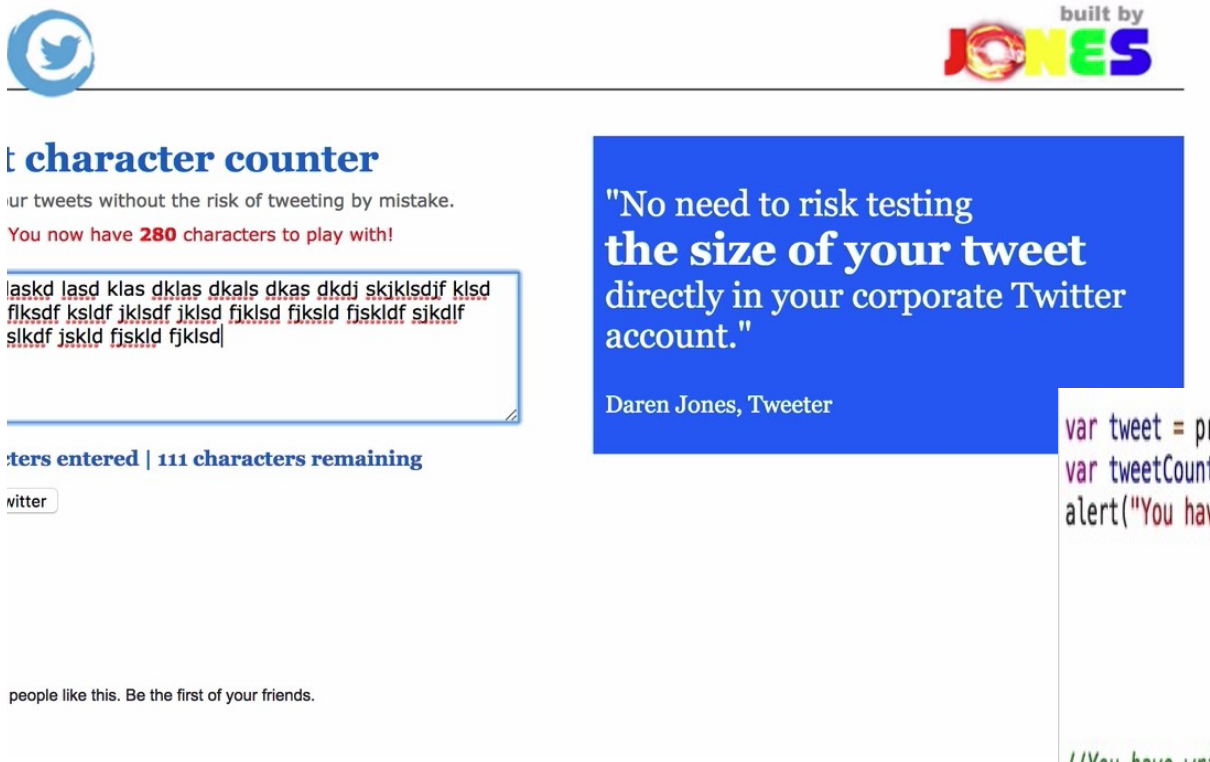
1. Always give meaningful name to your variables. So anybody can figure out easily what is the data that is stored inside.
2. Can not call your variable var.
3. Variable name can not begin with number. It can contain numbers.
4. It can not contain spaces.
5. Variable can only contain letters, numbers, dollar sign and underscore.
6. A good idea is use camelCasing - first word is not capitalized, and the follow word is going to be upper case.

String Concatenation

- ”a” + “b” = “ab” , you will have a single string. you are joining together whatever is in the double quotes.

- Ex; window.alert(“hello” + “world”) = helloworld
- Ex: window.alert(“hello” + “ world”) = hello world

String lengths and retrieving the number of characters



- word.length;

- Var name = “brian”;
- name.length; = 5

```
var tweet = prompt("Compose your tweet:");
var tweetCount = tweet.length;
alert("You have written " + tweetCount + " characters, you have " + (140 - tweetCount) + " characters remaining.");



//You have written 182 characters, you have -42 characters left.
```

Slicing and extracting parts of a string

```
//You have written 182 characters, you have -42 characters left.
```

How can you implement to remove the extra characters?

Slice(): it is a function that separate the string into separate characters. It gets done by position. It starts from position 0.
Ex: var name = ”brian” … name.slice(0,2); output = br

```
var tweet = prompt("Compose your tweet:");
var tweetUnder140 = tweet.slice(0,140);
alert(tweetUnder140);
```

Basic Arithmetic and the Modulo Operator in JS.

Addition

```
var a = 2 + 3; //5
```

Subtraction

```
var b = 10 - 2;
```

Multiplication

```
var c = 3 * 3;
```

Division

```
var d = 6 / 2; //9
```

Modulo

```
var e = 9 % 6; //3
```

Module will give you the reminder of the division.

12%8 = 4
6%4 = 2
45%2 = 1

If the module is 0; its an even number if the module is 1; its an odd number;

Challenge

Dog Age to Human Age Formula

$$humanAge = (dogAge - 2) \times 4 + 21$$

Increment and Decrement Expressions

```
var x = 5;
x = x + 1; //6
```

Equivalent
This is called the increment expression.

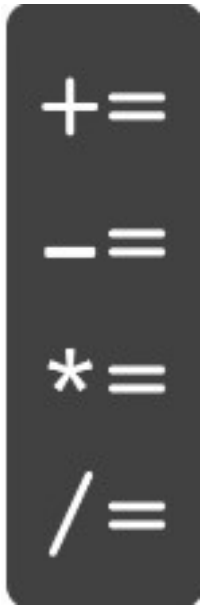
```
var x = 5;
x++; //6
```

?

Equivalent
This is called the decrement expression.

```
var x = 5;
x-- ; //4
```

Also work for:



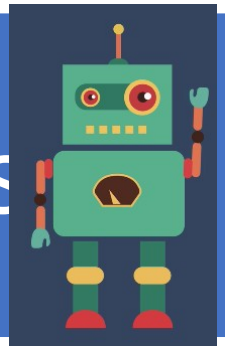
Increasing the value of x by more than 1

```
var x = 5;
x += 2 ; //7
```

Increasing the value of x with the value of another variable.

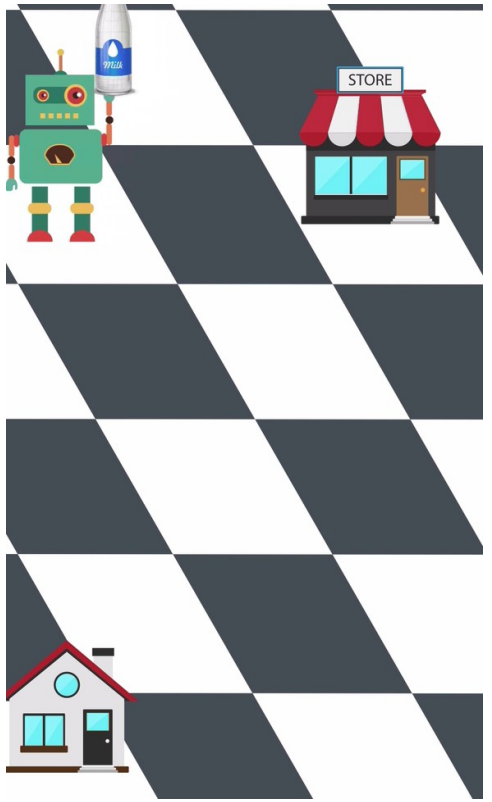
```
var x = 5;
var y = 3;
x += y ; //8
```

Functions



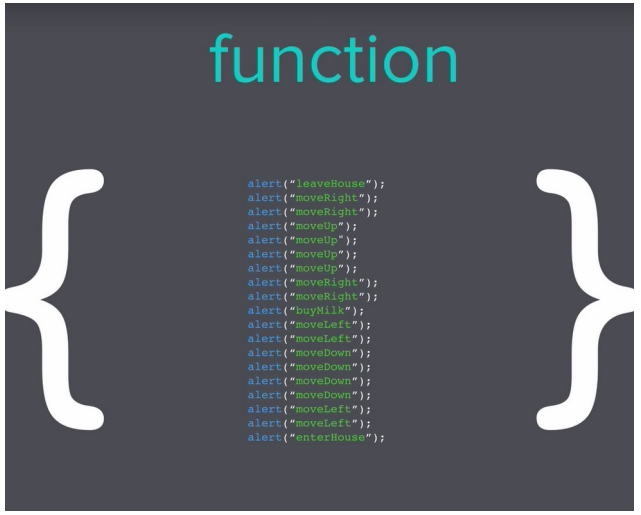
•Imagine I am a robot and I work in your house. You have to give me instruction to command me. So if you want to send me to the store and buy you some milk you have to write the follow instructions. If you want a bottle of milk every day, you will have to write all this code repeatable every day.


```
alert("leaveHouse");
alert("moveRight");
alert("moveRight");
alert("moveUp");
alert("moveUp");
alert("moveUp");
alert("moveUp");
alert("moveRight");
alert("moveRight");
alert("buyMilk");
alert("moveLeft");
alert("moveLeft");
alert("moveDown");
alert("moveDown");
alert("moveDown");
alert("moveDown");
alert("moveLeft");
alert("moveLeft");
alert("enterHouse");
```



Function

- what can you do instead?
  - What if you take this series of instructions that together send the little robot to the store, buys your milk and brings it back.
  - What if you packaged it into a single package of code and you gave it a name?



Create the function

To get the robot to buy you milk, its simple call the function.

Calling the function

Arrow function

Were introduced in ES6.

- Function Hello() {
  Window.alert(”hello”);
- }

- Hello = function() {
  Window.alert(”hello”);
- }

- getMilk => () => {
  Window.alert(”hello”);
- }

Function: Parameters and Arguments

```
getMilk();
```

Specify how many bottle of milk I want the robot to get.

```
getMilk(2);
```

```
function getMilk(bottles){
    alert("leaveHouse");
    alert("moveRight");
    alert("moveRight");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveRight");
    alert("moveRight");
    alert("buy" + bottles + "bottles of Milk");
    alert("moveLeft");
    alert("moveLeft");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveLeft");
    alert("moveLeft");
    alert("enterHouse");
}
```

Creating the function

```
function getMilk (bottles) {
    var cost = bottles * 1.5;
    //Do something with cost
}
```

Calling the function

```
getMilk(2);
```

Create a function that calculate how many bottle of milk the robot will buy depending of the amount of money you are giving.

```
function getMilk(money){

    alert("leaveHouse");
    alert("moveRight");
    alert("moveRight");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveRight");
    alert("moveRight");
    var numBottles = money / 1.5;
    alert("buy" + numBottles + " bottles of Milk");
    alert("moveLeft");
    alert("moveLeft");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveLeft");
    alert("moveLeft");
    alert("enterHouse");
```

Function: outputs and return value

```
function getMilk(money){
    alert("leaveHouse");
    alert("moveRight");
    alert("moveRight");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveUp");
    alert("moveRight");
    alert("moveRight");
    var numBottles = money / 1.5;
    alert("buy" + numBottles + " bottles of Milk");
    alert("moveLeft");
    alert("moveLeft");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveDown");
    alert("moveLeft");
    alert("moveLeft");
    alert("enterHouse");
    return money % 1.5;
```

Function that are able to give an output. We can use the keyword return to specify this function in addition to accept inputs, it gives you an output. In this example, we are getting the reminder as an output.

```
getMilk(4); //dollars
```

In this example, we will have 1-dollar change. We can assign this value to a variable.

```
var change = getMilk(4)
//change = 1
```

Recap

Creating the function

```
function getMilk (money) {
  return money % 1.5;
}
```

We have the function keyword, give the function a name. inside the parenthesis you have to specify whether or not this function will have an input. In order to get an output out the function you have to have the return keyword.

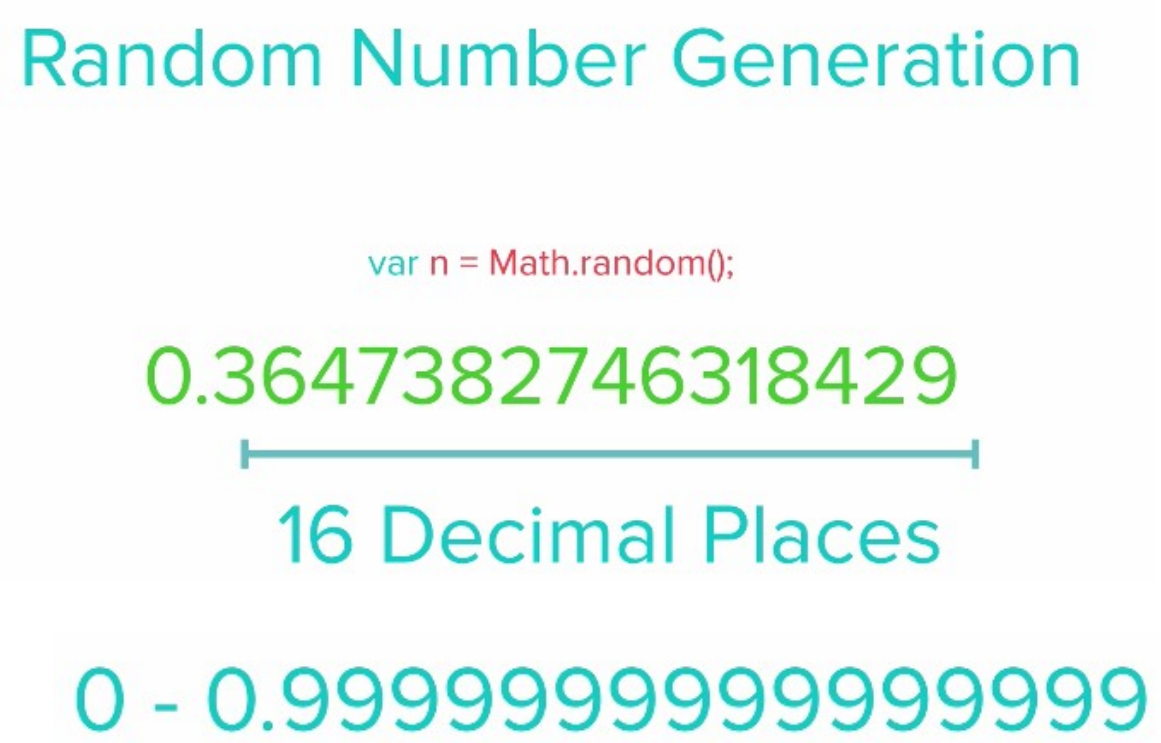
Whatever gets return from the function you can assign to a variable. It needs to be in the right side of the function.

Calling the function

```
var change = getMilk(4);
```

Love calculator

- Random Number Generation
  - var n = Math.random();



It will generate a number from 0 to 0.9999999. It will not reach 1.

The number will be different every time you run the program.

How to make the Math.random() into a whole number.

```
var n = Math.random();
n = n * 6;
n = Math.floor(n);
```

```
0.3647382746318429
2.18842964779
2
```

```
var n = Math.random();
Math.floor(n * 6) + 1
```

1 - 6

If statement

- Use the keyword if the evaluate a statement. It the statement is true, then it will carry out a particular sequence of code. If the statement is false, you will have an else to execute different sequence of code.

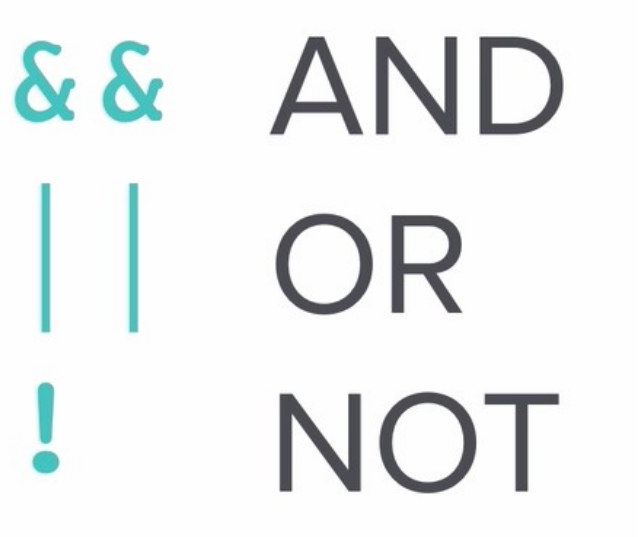
```
var name = “max”;

If (name === “max”) {
        printName();
} else {
        print(“your name is not max”);
}
```

Comparators

===	Is equal to
!==	Is not equal to
>	Is greater than
<	Is lesser than
>=	Is greater or equal to
<=	Is lesser or equal to

Difference between == to ===?

Combining Comparators



Both conditions need to be true.

One of the conditions need to be true.

Not or the opposite.

Collections: working with JavaScript Arrays

JavaScript arrays are used to store multiple values in a single variable. In other words, it is a collection of items that are related and can be stored together into the same variable.

Retrieving an egg in a position 1


```
var myEgg = eggs[1];
```

```
eggs.length;  5
```

Ex from w3school: var cars = ["Saab", "Volvo", "BMW"];

Push and pop

It will add the item to the end of the array.

Remove item of your array. It will remove the last item.

Control Statement: while loops

```
while (something is true) {
    //Do something
}
```

While loop execute the statement base on the condition. Inside the parenthesis there is a statement while that statement is true, the code is going to execute and run again, again, again until the statement is not longer true.

```
var i = 1;

while(i<2) {
   console.log(i);
   i++;
}
```

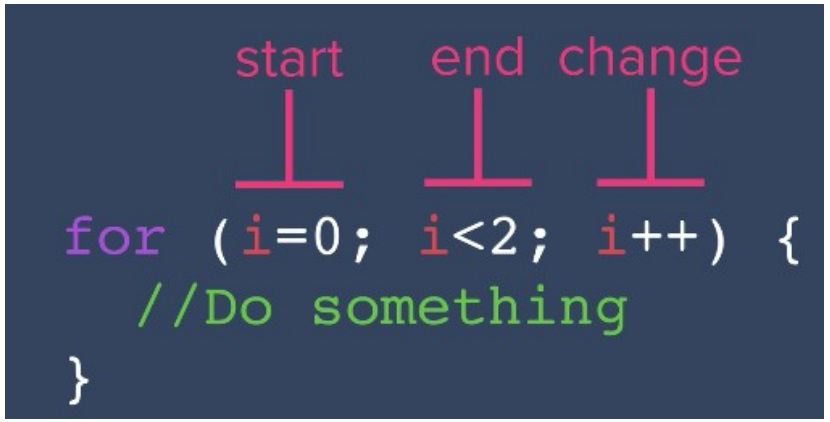
FizzBuzz game

```
var output = [];
var count = 1;

  while(count <= 100) {

  if (count % 3 === 0 && count % 5 === 0) {
          output.push("FizzBuzz");
      } else if (count % 3 === 0) {
          output.push("Fizz");
      } else if (count % 5 === 0) {
          output.push("Buzz");
      } else {
          output.push(count);
      }

   count++;

   }

   console.log(output);
}
```

index.js:22

```
(100) [1, 2, "Fizz", 4, "Buzz", "Fizz", 7, 8, "Fizz", "Buzz", 11, "Fizz", 13, 14, "FizzBuzz", 16, 17, "Fizz", 19, "Buzz", "Fizz", 22, 23, "Fizz", "Buzz", 26, "Fizz", 28, 29, "FizzBuzz", 31, 32, "Fizz", 34, "B
▶uzz", "Fizz", 37, 38, "Fizz", "Buzz", 41, "Fizz", 43, 44, "FizzBuzz", 46, 47, "Fizz", 49, "Buzz", "Fizz", 52, 53, "Fizz", "Buzz", 56, "Fizz", 58, 59, "FizzBuzz", 61, 62, "Fizz", 64, "Buzz", "Fizz", 67, 68, "F
izz", "Buzz", 71, "Fizz", 73, 74, "FizzBuzz", 76, 77, "Fizz", 79, "Buzz", "Fizz", 82, 83, "Fizz", "Buzz", 86, "Fizz", 88, 89, "FizzBuzz", 91, 92, "Fizz", 94, "Buzz", "Fizz", 97, 98, "Fizz", "Buzz"]
```

Control Statements: For Loops



Specify the number of time, we would like the loop to run.

```
for(var i=1; i<2; i++){
   console.log(i);
}
```