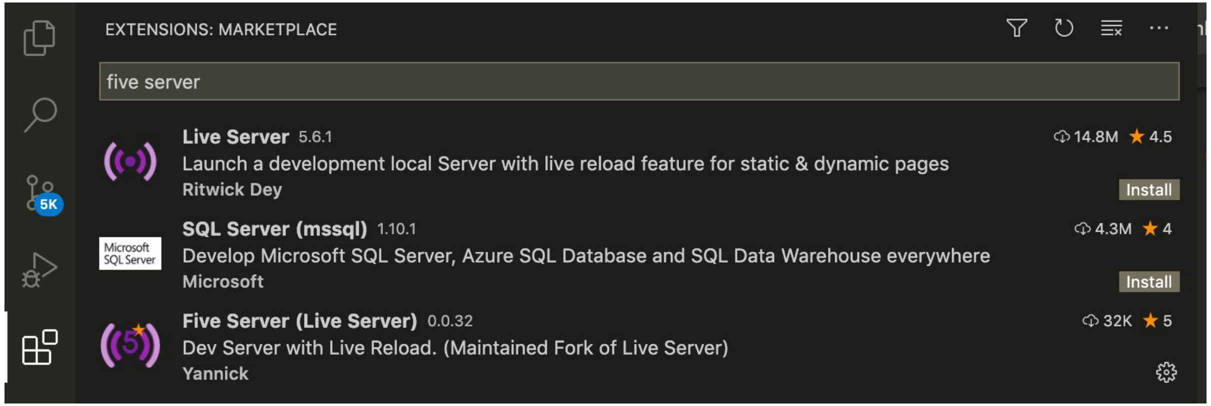
# Javascript

# Getting Started


EXTENSIONS: MARKETPLACE

five server

**Live Server** 5.6.1 — 14.8M ★ 4.5
Launch a development local Server with live reload feature for static & dynamic pages
Ritwick Dey — Install

**SQL Server (mssql)** 1.10.1 — 4.3M ★ 4
Develop Microsoft SQL Server, Azure SQL Database and SQL Data Warehouse everywhere
Microsoft — Install

**Five Server (Live Server)** 0.0.32 — 32K ★ 5
Dev Server with Live Reload. (Maintained Fork of Live Server)
Yannick

## **Javascript**

- As long as you have a browser you're good
- Make sure Javascript is enabled
  - It probably

## **Visual Studio Code**

- Not super necessary but could be nice to have since I'm using the Five Server extension
- https://code.visualstudio.com/
- Five server extension
  1. Go to extension marketplace
  2. Add Five Server
     a. Live server is fine, no live reupload so you might have to hit refresh

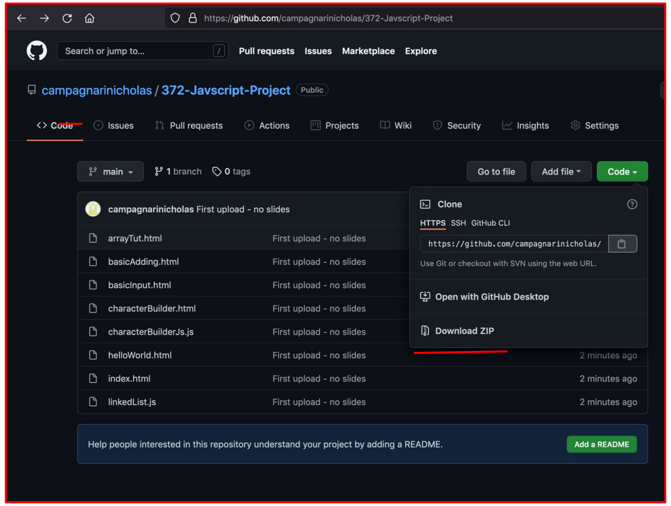
## **Quickest Way to Follow Along**

- https://playcode.io/new/

# Gitting the Project

1. Go to https://github.com/campagnarinicholas/372-Javscript-Project
2. Grab the clone url
3. Go to terminal
4. run "https://github.com/campagnarinicholas/372-Javscript-Project"
5. You can open it with your editor of choice

Or just grab the zip file and open that up.
You can do that too.



campagnarinicholas First upload - no slides

| | | |
|---|---|---|
| arrayTut.html | First upload - no slides | |
| basicAdding.html | First upload - no slides | |
| basicInput.html | First upload - no slides | |
| characterBuilder.html | First upload - no slides | |
| characterBuilderJs.js | First upload - no slides | |
| helloWorld.html | First upload - no slides | 2 minutes ago |
| index.html | First upload - no slides | 2 minutes ago |
| linkedList.js | First upload - no slides | 2 minutes ago |

```
https://github.com/campagnarinicholas/
```

Use Git or checkout with SVN using the web URL.

Help people interested in this repository understand your project by adding a README.

## Brief History of Javascript

**Created by Netscape Developer Brendan Eich in 1995**

**<u>Used for website creation</u>**

- **Started with building websites for Netscape**
- **Today used by 67.8% of developers**
- **“Javascript” comes from Netscape’s support of Java applets within its browser**
  - **Originally called LiveScript**

*Brendan Eich*

## Javascript Overview

Ran by ECMA (European Computer Manufacturers Association)

ECMA-262 Spec is the definite standard version of core Javascript

**According to them**

- Javascript is a lightweight, interpreted programming language
- Designed for network-centric applications
- Complementary to and integrated with Java and HTML
- Open and cross platform
- When you open up a webpage, the HTML document references or includes client-side Javascript

**Multi-Paradigm**

- Object-oriented
- Imperative
- ***Functional***

## Javascript Uses

Javascript is primarily used for web development

- Static website
- Dynamic web applications

Other popular use cases

- Servers and Server Applications
  - NodeJS and expressJS
  - Wal-mart uses NodeJS for its backend (2009)
- Mobile Apps

## Advantages and Disadvantages

### Advantages

- Asynchronous interaction- validate user input before page reaches server
- UI/UX- Create reactive interfaces that provide immediate feedback

### Disadvantages

- (Client-side) doesn’t read or write files for security
- No multithreading or multi-processor capabilities

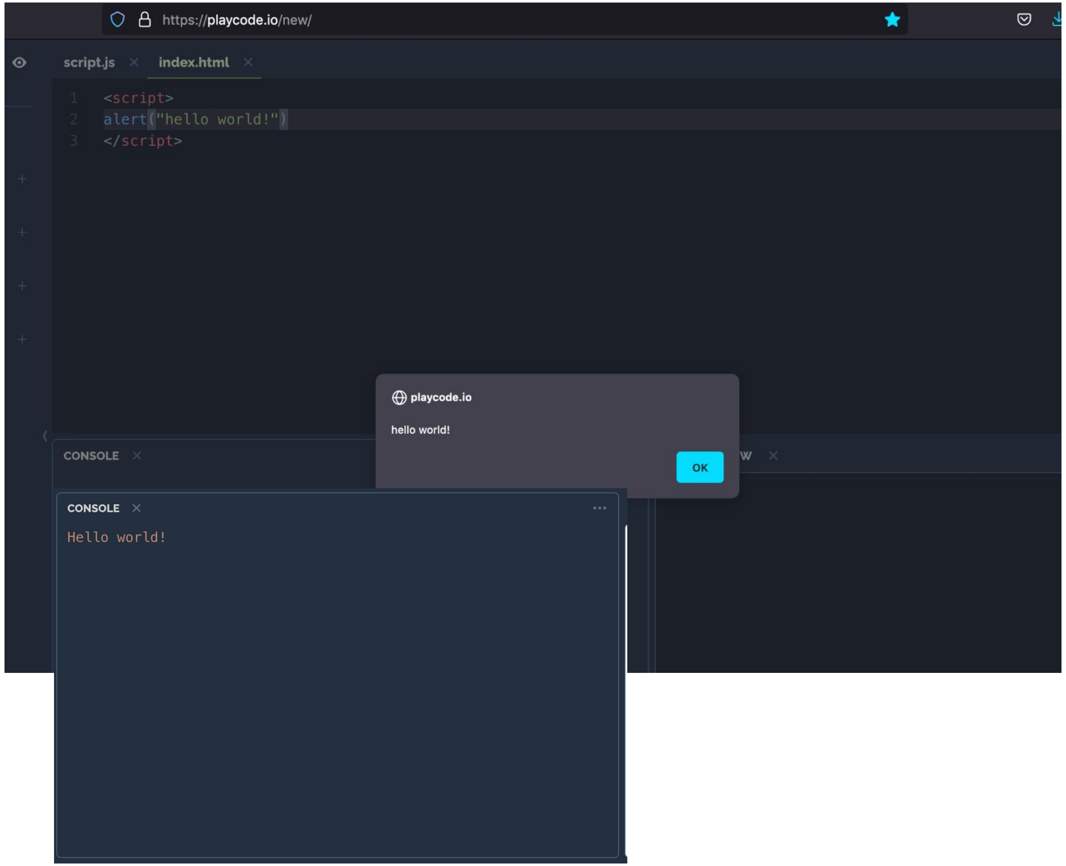
## Hello World With Javascript

**Send to console-**

console.log(“Hello World!”);

**Send a browser alert-**

```
 <script>
    alert( 'Hello world!' );
  </script>
```



https://playcode.io/new/

```
<script>
alert("hello world!")
</script>
```

playcode.io

hello world!

```
Hello world!
```

## Primitive Data, Arithmetic, and If-Statements

- **7 Primitive Data Types**
  - String
  - Number
  - Bigint- larger than 2^53 - 1
    - Behaves a lot like numbers
    - No overflow!
  - Boolean
  - Symbol
  - Undefined - empty var is undefined
  - Null
- **Arithmetic Operators**
  - %, /, *, +, - , =
    - Floating point # system
    - ½==.5==true
  - ++, --
    - Increment/decrement
    - x=3, --x returns 3 and decrements, x-- decrements and returns 2

- **if-statements:**

```
if (condition) {

  //  block of code to be executed if the condition is true

}

else if  {

  //  block of code to be executed if the condition is false

}else {

  //  block of code to be executed if the condition is false

}
```

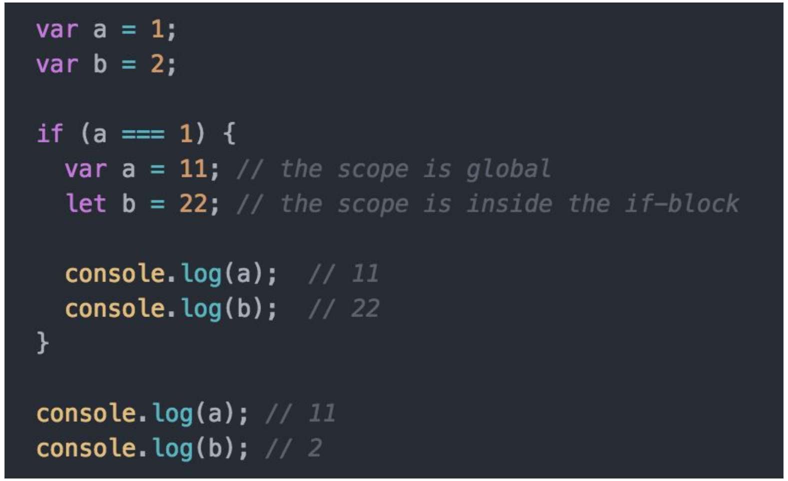
- == vs ===
  - **== converts the variable values to the same type before performing comparison**
  - **=== doesn’t**

## **Scopes**- Variable Declaration is Weird

**Different Ways to Initialize Variables**

- **var**
  - **Global scope (not block)**
  - **Supports reassignment**
- **let / const**
  - **Block scope**
  - **let**
    - **Supports reassignment (good for loops)**
  - **const**
    - **Doesn't support reassignment**

**All are local/function scope if defined in a function (like most languages)**



```
var a = 1;
var b = 2;

if (a === 1) {
  var a = 11; // the scope is global
  let b = 22; // the scope is inside the if-block

  console.log(a);  // 11
  console.log(b);  // 22
}

console.log(a); // 11
console.log(b); // 2
```

## Oops! All Loops

### Many different types of loops

- ***For***

```
for (let i = 0; i < array.length; i++) {
    console.log(array[i]);
}
```

- ***For - in***

```
for (let i in array) {
        console.log(i);
        }
```

### Many different types of loops

- ***For - of***

```
for (let i of array) {
    console.log(array[i]);
}
```

- ***While***

```
let i = 0;
while (let i < 100) {
        i++;
}
```

## Arrays

Declaring an array:

**let** animals = [‘donkey’, ‘horse’, ‘pig’];

- Supports indexing
- No declaring length upon initialization
- Many methods:
  - .includes(‘donkey’) - returns true
  - toString()- return “donkey,horse,pig”
  - .join(“ * “)- returns “donkey * horse * pig”
  - .pop()- removes and returns “pig”
  - .push(“dog”)- adds “dog” to end of list
  - .shift()- adds element to front and returns length of list
  - .unshift()- removes first element and returns length of list element removed
- **Note: You can use** ***delete animals[0]*** **to remove first element**
  - **Don’t.**
  - **It’s bad.**
  - **Just use pop() or shift()**

# Objects and Arrow Functions

Javascript allows objects to be created with functions and properties

"Animal" object has numberOfLegs attribute and a(n arrow) function called sleep

**Arrow Functions:**

- Compact but limited ways of doing things
- Gets rid of "function functionName" when not necessary

**Traditional:**

```
function sleep() {
        print("Zzz");
}
```

**Arrow:**

```
(sleep) => print("Zzz");
```

**Animal object**

```
const animal = {
  numberOfLegs: 4,
  sleep: () => print("Zzz")
}
```

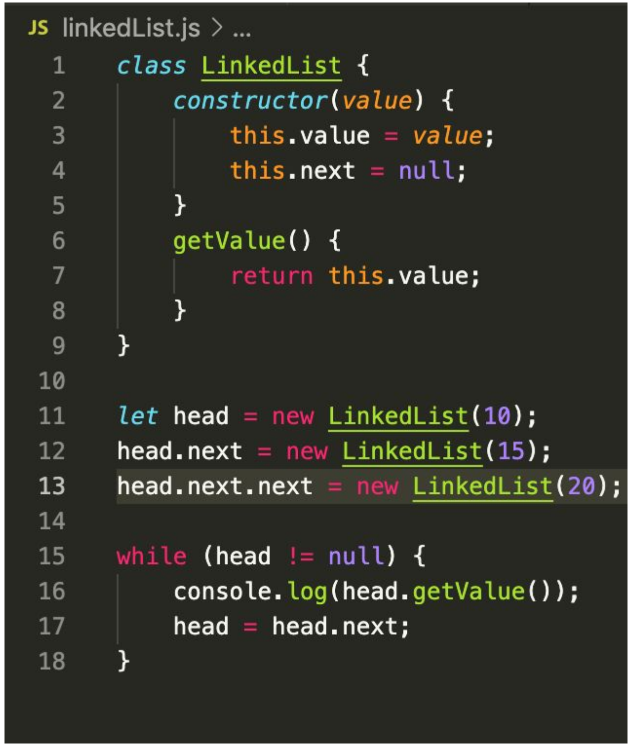
**Dog inherits from animal**

```
const dog = {
  prototype: animal,
  bark: () => print("Woof!")
}
```

## Classes

ECMAScript 2015 introduced JavaScript Classes. They behave like most other classes.

- Keyword **class**
- Reference self-object with **this**
- Methods don't need **function** before them



```
class LinkedList {
    constructor(value) {
        this.value = value;
        this.next = null;
    }
    getValue() {
        return this.value;
    }
}

let head = new LinkedList(10);
head.next = new LinkedList(15);
head.next.next = new LinkedList(20);

while (head != null) {
    console.log(head.getValue());
    head = head.next;
}
```

## Javascript Takeaway

**Javascript is one of the single most useful programming languages there are. Without it, modern web applications would not be as dynamic and interactive as they are today. Now if you want to make a dynamic web app, you would probably not use raw JS, but whatever you do use would be built upon the back of lessons learned from raw JS.**

As a refresher:

- Dynamically typed
- Client side scripting - Used for Websites
- Multi-paradigm
  - Functional
  - Imperative
  - OOP
- Easy to learn
  - Hard to master?