Intro to JavaScript and Node.js

```
console.log('hello world'); // FIX ME
```

What is JavaScript

- A scripting language build for the web
- Good for beginners, improving for experts
- Used in:
 - Browsers
 - Servers
 - Command line
 - Desktop
 - Mobile

What is Node.js

- What is a JavaScript runtime?
- What does Node.js add?
- What are the alternatives?

Things that are common in (most) other languages

- // Comment, or /* Comment */
- Logical operators: && | |
- Math operators: + / * ++ -- += -= *= /=

Comparison and logical operators

Equality and inequality	== !=
Strict equality and inequality	=== !==
Relative value	< > <= >=
And	&&
Or	П
Not	!

Variables

- var
- let
- const

```
var az = 'Arizona';
let ca = 'California';
const pr = 'Puerto Rico';
```

Scope and hoisting

- A variable lives within a context
- Its context depends on <u>where</u> and <u>how</u> it was defined
- In general, let and const are more strictly scoped than var
- Hoisting separates a variable's declaration and initialization
- A var is initialized to undefined, let and const are not

```
console.log(az); // undefined
console.log(ca); // ReferenceError: ca is not defined
console.log(pr); // ReferenceError: pr is not defined
var az = 'Arizona';
let ca = 'California';
const pr = 'Puerto Rico';
```

Basic variable types

- String
- Number
- Boolean
- Null
- Undefined

```
console.log('hello world'); // String
console.log(42); // Number
console.log(true); // Boolean
console.log(null); // Null
console.log(undefined); // Undefined
```

Conditional flow

- if/else/else if
- switch/case
- condition ? true : false

```
const state = 'Florida';
if (state == 'Puerto Rico') {
  console.log('state is Puerto Rico');
|} else if (state == 'Arizona') {
  console.log('state is Arizona');
} else {
  console.log('state is something else');
```

```
const state = 'Florida';
switch (state) {
  case 'Puerto Rico':
    console.log('state is Puerto Rico');
   break;
  case 'Arizona':
    console.log('state is Arizona');
   break;
  default:
    console.log('state is something else');
```

```
const state = 'Florida';
console.log('state is ' + state == 'Arizona' ? state : 'something else');
```

Complex variable types

- Object
- Array
- Function

```
let states = {
                            let eastValley = [
                                                      function sayHello() {
az: 'Arizona',
                             'Chandler',
                                                       console.log('hello!');
ca: 'California',
fl: 'Florida',
                             'Gilbert',
};
                            ];
```

Loops

- for
- while
- do/while
- for...in

Functions

- Plain
 - o function() {}
- Arrow
 - o () => {}
- return

Exceptions

- try
- catch
- finally

Prototype and Class

- Prototype
 - Here is a "Thing"
 - Make a new "Thing"
- Class
 - Here is a description for a "Thing"
 - Make a new "Thing"

Asynchronous flow

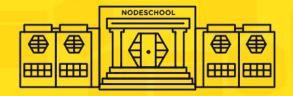
- setTimeout / setInterval
 - o timers
- callback
 - Here's a Function, run it when you're done
- Promise
 - Do this, then that, then...
- async/await
 - Wait for this to finish. After that's done, keep running my code.

Modules

- How can you share code with the world?
- What if you just want to share it with your co-workers?
- require
- Static import and export
- Dynamic import()

Examples of JavaScript & Node.js

- Browser-based
- Server-side
- Command line
- Electron
- React Native



NODESCHOOL

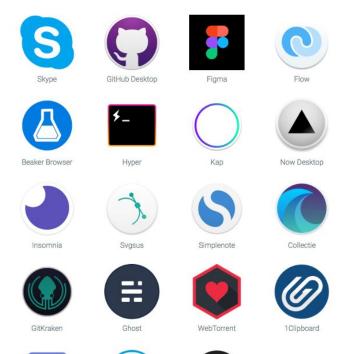
Open source workshops that teach web software skills. Do them on your own or at a workshop nearby.

Tutorials Events Chapters About Build Host

Get started by installing one of our **core workshops**.

Apps built on Electron

With Electron, creating a desktop application for your company or idea is easy. Initially developed for GitHub's Atom editor, Electron has since been used to create applications by companies like Microsoft, Facebook, Slack, and Docker.











Docs Community Blog Q Search

Who's using React Native?

Thousands of apps are using React Native, from established Fortune 500 companies to hot new startups. If you're curious to see what can be accomplished with React Native, check out these apps!



Facebook iOS · Android



Facebook Ads Manager iOS · Android



Facebook Analytics iOS · Android



Instagram iOS · Android

Using React Native in the How We Built the First Cross-Platform React Facebook App Native App

React Native at Instagram



F8 iOS · Android



Bloomberg iOS · Android



Pinterest iOS · Android

at Pinterest



Skype iOS · Android

Tutorial: Building the F8 conference app

How Bloomberg Used Supporting React Native React Native to Develop its new Consumer App



Tesla iOS · Android



Uber Powering UberEATS with React Native and Uber Engineering



Walmart iOS · Android



Wix.com iOS · Android

React Native at Walmart Building a React Native

Additional resources

- Mozilla Developer Network https://developer.mozilla.org/
- Google Developers | Web https://developers.google.com/web/
- Eloquent JavaScript https://eloquentjavascript.net/
- You Don't Know JS https://github.com/getify/You-Dont-Know-JS
- Explore on your own!
 - Twitter
 - YouTube
 - Medium
 - Stack Overflow

Did I finish early?

- JavaScript equality: https://dorey.github.io/JavaScript-Equality-Table/
- The Future of JavaScript!
- What is a linter, and why should everyone use them?
- Opinions and arguments in JavaScript
- Yarn and npm
- Angular, React, and Vue
- JavaScript, but strict: TypeScript and Flow
- "Full stack" JavaScript
- Functional programming with JavaScript
- Grunt, Gulp, Webpack, and other tools