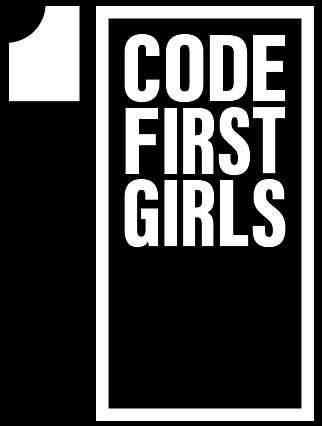
WELCOME TO CFG
YOUR INTRODUCTION TO JAVASCRIPT



TECH SHOULDN'T JUST BE A BOYS CLUB.

COURSE JOURNEY

MODULE 01	MODULE 02	MODULE 03	MODULE 04	MODULE 05	MODULE 06	MODULE 07	MODULE 08
INTRO JAVASCRIPT	CONDITIONS & LOGIC	THE DOM	INTRO REACT	REACT COMPONENTS	STYLING COMPONENTS	STATES & EVENTS	PROJECT PRESENTATION

04 JAVASCRIPT

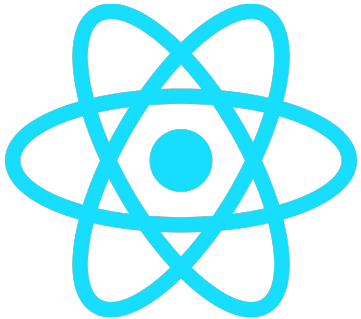
What is React?

Creating a web app

Pros/Cons

Alternatives

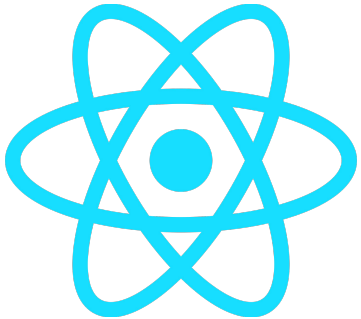
Introducing React



React is a free and open-source front-end JavaScript library for building user interfaces based on UI components.

Firstly, React was deployed by Facebook in 2011 and 2012, later open sourced in 2013.

Introducing React



React is a free and open-source front-end JavaScript library for building user interfaces based on UI components.

Firstly, React was deployed by Facebook in 2011 and 2012, later open sourced in 2013.

JS+... JSX

One of the biggest differences between React and any other framework is JSX. It's essentially **JS with the ability to write HTML-like tags**. HTML-like... not HTML, we'll cover some differences as we're going.

```
const element = <h1>Hello, world!</h1>;
```

This funny tag syntax is neither a string nor HTML.

It is called JSX, and it is a syntax extension to JavaScript. It's recommend using it with React to describe what the UI should look like. JSX may remind you of a template language, but it comes with the full power of JavaScript.

JSX isn't essential for React but it just makes things easier. Without you're basically writing a HTML file in JS

```
React.createElement('h1', null, 'Hello World!');
```

Standard Styling... If you want

Cascading Style Sheet (CSS) is a stylesheet language - it describes how HTML should be displayed (to the browser).

For example, it can point to a specific element (e.g. the paragraph tag) and say that it should have **bold text** - from that point on, the <p> tag will be displayed as bold!

Consider HTML as the **bland body you create** (e.g. we should have two eyes!), whereas **CSS is the attributes you set** (e.g. those eyes should be the colour brown!)

App.jsx

```
import './App.css';

function App() {
 return (
   <div className="App">
     <header className="App-header">
       <p>Edit <code>src/App.js</code>
     </header>
   </div>
 );
}

export default App;
```

App.css

```
.App {
 text-align: center;
}

.App-header {
 background-color: #282c34;
 min-height: 100vh;
 display: flex;
 flex-direction: column;
 align-items: center;
 justify-content: center;
 font-size: calc(10px + 2vmin);
 color: white;
}
```

Pulling it all together

Node.js is a cross-platform, open-source server environment that enables us to run JavaScript to run on Windows, Linux, Unix, macOS.

npm (Node Package Manager) is a library and registry for JavaScript software packages. npm also has command-line tools to help you install the different packages and manage their dependencies

Key Commands:

`npm i` - installs dependencies

`npm run <script name>` - runs a specific script in "packages.json"


```
{
 "name": "test_app",
 "version": "0.1.0",
 "dependencies": {
   "react": "^18.2.0",
   "react-dom": "^18.2.0",
   "react-scripts": "5.0.1",
   "web-vitals": "^2.1.4"
 },
 "scripts": {
   "start": "react-scripts start",
   "build": "react-scripts build",
   "test": "react-scripts test"
 }
}
```

package.json

NOW LET'S PRACTICE TOGETHER

CREATING A REACT APP

10 MINS

Exercise 1.0 - Run create-react-app

Open a terminal in VSC, navigate to a suitable location and run (this could take some time)

```
npx create-react-app {your project name here}
```

Exercise 1.1

Poke around the generated files in VSC (most notably the "src" and "public" directories)

5 MINS

Exercise 1.2

In the same terminal navigate to the directory created and run the app

e.g. `cd {your project name here}`

```
npm start
```

Exercise 1.3

Edit `src/App.js` (text in the p and a tags is good) and save

Let's Break That Down

```
npx create-react-app {your project name here}
```

We used npx to run the Node Package Manager (npm) controlled package "create-react-app".

This provided us with boilerplate code to get an initial app up and running in a typical file structure.

Running `npm start` in the apps root directory will run a development server locally that "hot reloads". **Notice the http address provided to view your app.**

Changing files in "src" and "public" directly impact the app and (if running a development server in the background) show in your local browser.

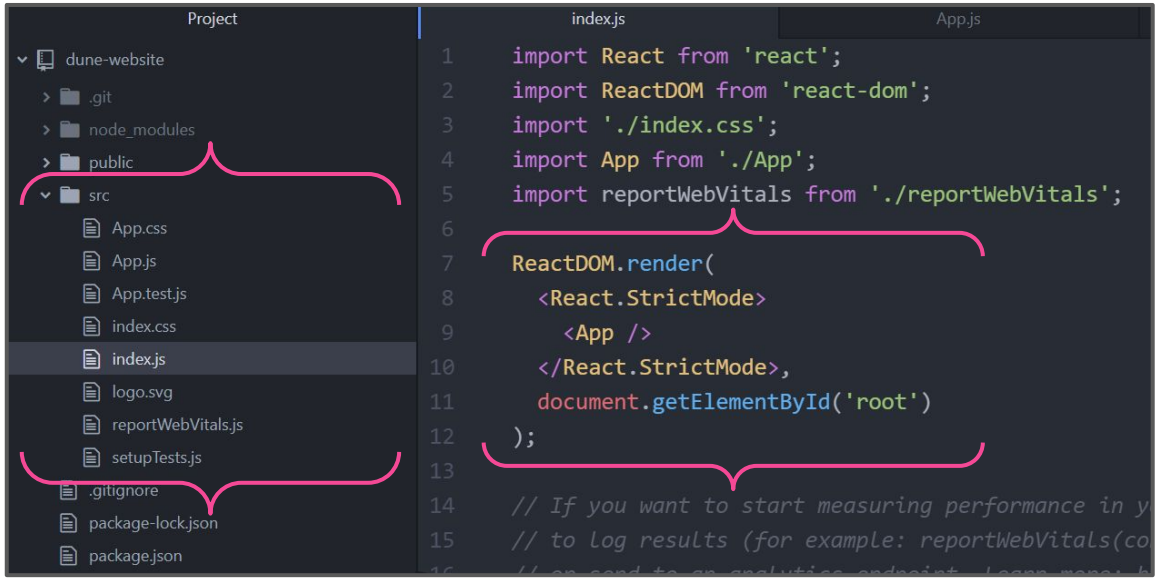
Exercise: Inspect, what's going on?

Spend several minutes to understand the file structure!

- **Task 1:**
  Consider the similarities with HTML tags, particularly how some parts are structured in these **<></>** brackets

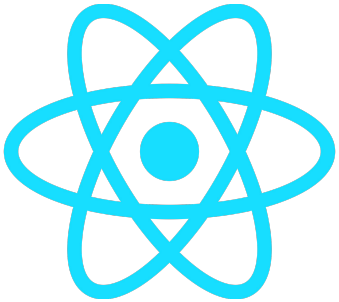
- **Task 2:**
  Consider what is ReactDOM! And what happens if we remove **<App />.**

- **Task 3:**
  Look into the imported **App** file (**App.js**) - what's inside it? What happens if you change stuff? What's **'App.css'** as well?



*You have approx. **10 minutes** for this (depending on your instructor's discretion + current time!). Google when you can!*

Pros of React

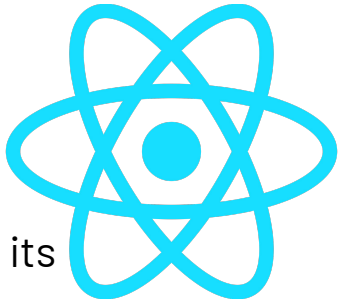


Its purpose - why is it useful, why even care?
React is all about creating and using 'components' - self contained elements that we manually create through code

For example, we can create a new button ourselves that fits **our** vision of what it should be (e.g. perhaps we need some specific design). React allows us to contain all of this code into one 'button' component (like a object) that can be **deployed over and over again** (making it incredibly easy for us to be consistent).

React's second benefit is its **virtual DOM**. Effectively it keeps a 2nd copy of our DOM which is updated when we make changes, and publishes / pushes this changes into the real DOM when its ready. The benefit of this is **speed** - instead of reloading a page 5 times, React will often batch new changes itself so the **entirety of the real DOM doesn't have to reload or change**

Cons of React



High Pace of Development

React is not just a rapidly growing library, it is also rapidly changing, which forces its developers to update the way they write code.

Flexibility and Lack of Conventions

More a result of being a JavaScript based framework but anything is possible and the barn doors are wide open.

Not a full-featured framework

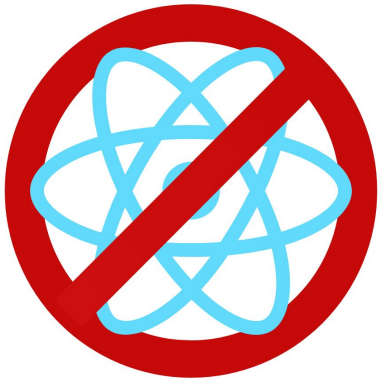
If you look at the MVC (Model View Controller) architecture, React only handles the view part. Whereas frameworks like Angular provide the complete MVC framework, which is more structured, and well managed.

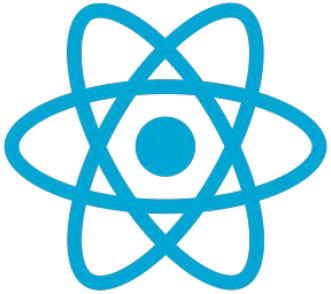
JSX

This is like JavaScript and html syntax, and allows to mix html and JavaScript together but this can make it difficult to work with when you start with React.

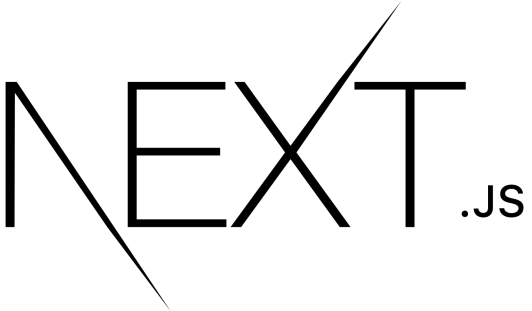
Alternatives

SVELTE





React Native



Dev Tools & Debugging

Firstly, you will make mistakes and you will see errors.

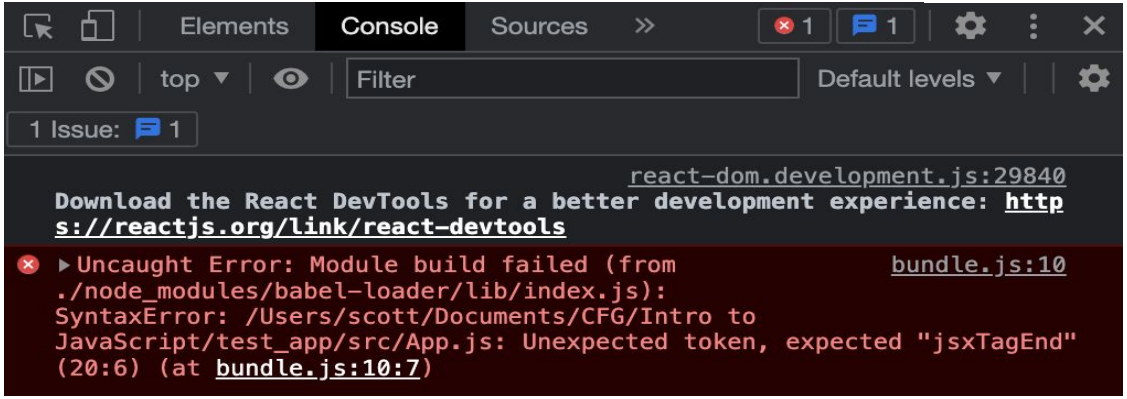
There's 2 key places to look:

1. Dev Tools - an extension to the in browser console (F12)

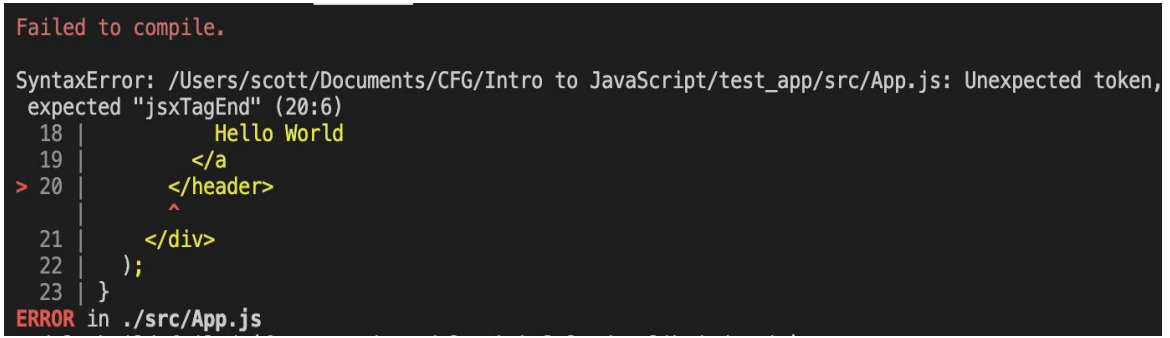
2. Terminal - the place you're running your app

Both are useful and will show different errors. Most of the time even point to the exact issue.

Dev Tools



Terminal



Exercise 2: Modifying our website

Modify your App.js to have new content!

- **Using the following tags....**
  <h1> (or h2, h3, ..., h5, h6), <p>

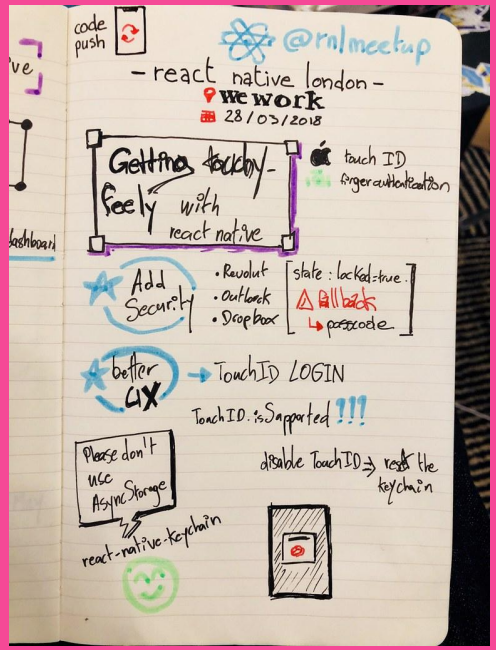
- **.....add the following information to your website:**
- Your favourite hobby or activity you like to do (e.g. for downtime)
- Any relevant rules (e.g. how to play if its a sport hobby)
- Why do you do it, why do you enjoy it
- Effectively your hobby 101 website

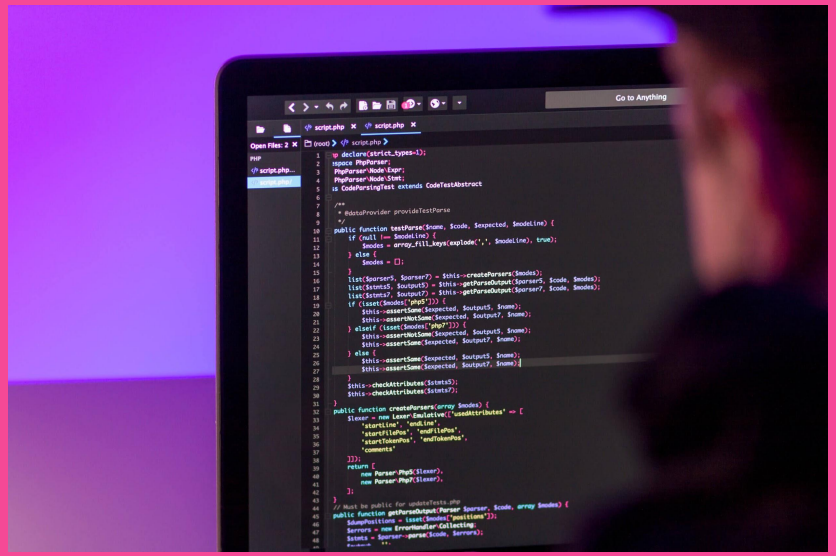
- **Use a combination of tags: you must have at least one of headings + paragraph tags**
  (Remember - you can google throughout!)

```
3
4   function App() {
5     return (
6       <div className="App">
7         <header className="App-header">
8
9           {/* These are how comments are made in JSX! */}
10          {/* Write some new tag here - preferrably a h1! */}
11
12
13          <img src={logo} className="App-logo" alt="logo" />
14          <p>
15            Edit <code>src/App.js</code> and save to reload.
16          </p>
17          <a
18            className="App-link"
```

*You have approx. **8 minutes** for this (depending on your instructor's discretion + current time!). Google when you can!*

HOMEWORK





+ **Homework Task**

Continue creating your app using create-react-app, dig around the file structure and make changes (check the terminal & dev tools for errors). Feel free to come armed with questions next week!

Hint - experiment with some other tags to use, or get more comfortable with the h1-h6 tags on the previous page. Add anything you want about your hobby - and don't be afraid to explore more about React on the [Create React App website](#)!

THANK YOU
HAVE A GREAT WEEK!