Frontend - React

shorturl.at/cirT6
Start at 2:05

- Javascript
- Html & css
- React

Javascript Basics

Variables & Constants

Global variables:

```
1    var variableA = 0;
2    var variableB = "Hello";
```

Block scoped variable:

```
1    let variableA = 0;
2    let variableB = "Hello";
```

Constants:

```
1    const constantC = "World";
```

Data Types

Dynamically typed:

```
const num = 6;
const str = "world";
const b = true;
const x = null;
const arr = ["one", "two", "three"];
```

```
const person = {
    firstName: "John",
    lastName: "Doe",
    age: 12,
    fullName: function() {
      return this.firstName + " " + this.lastName;
    }
  };

const name = person.fullName;
const age = person.age;
```

Functions

Function declaration:

```
function func(input1, input2) {
    // do stuff
}
```

Function Expressions:

```
const func = function(input1, input2) {
    // do stuff
}
```

Arrow functions:

```
const func = (input1, input2) => {
    // do stuff
}
```

Single line functions:

```
const func = (input1, input2) => input1 + input2;
```

Callbacks

Can pass callback functions into functions as arguments and call the callback function at the end.

```
const func = (input1, callbackFunc) => {
    // do stuff
    callbackFunc();
}
```

```
func(input, predefinedFunc);

func(input, newFunc = () => {
    // callback stuff
});
```

Promise (async/await)

Promises lets you wait for something on the side while the rest of the program continues on.

```
1    const asyncFunc = async (input) => {
2        try {
3            const result = await timeConsumingOperation(input);
4            // do stuff
5        } catch (err) {
6            // handle error
7        }
8    }
```

Html & Css

Html

HyperText Markup Language

Used in web pages

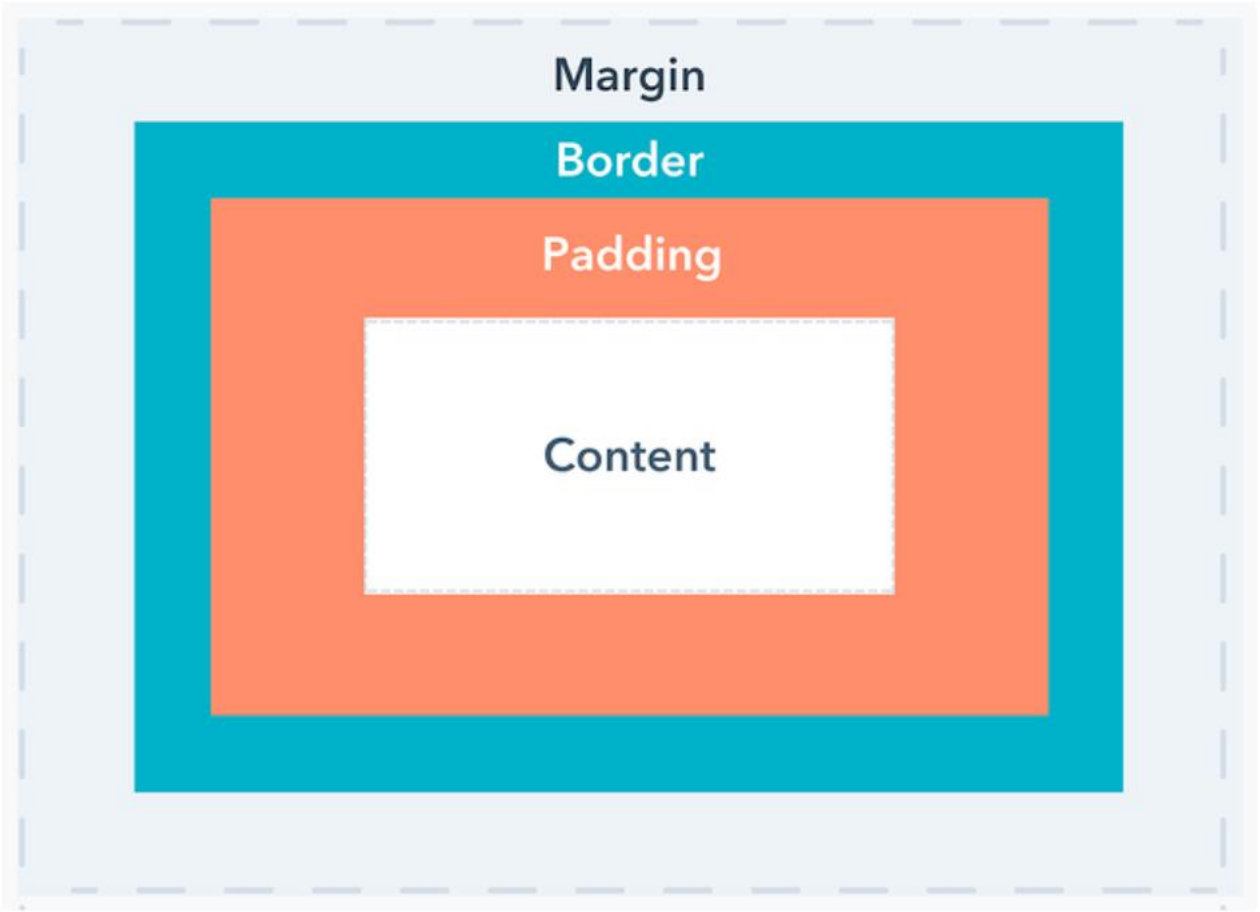
https://codebeautify.org/htmlviewer

Css

Used to style Html documents

https://www.w3schools.com/css/css3_flexbox.asp

Padding vs Margin vs Border



Source: https://blog.hubspot.com/website/css-margin-vs-padding

React

Why React?

- Integrate Javascript into html (JSX)
- Break down web pages into smaller components
- Only re-render components that need to be re-rendered
- Many premade assets on the internet (eg. https://mui.com/)

React Documentation: https://reactjs.org/docs/getting-started.html

Extension for browsers: React Developer Tools

Initialising a React project

1. Install node.js

   https://nodejs.org/en/download/

2. In terminal: `npx create-react-app app-name`

Running the react project:

```
cd app-name

npm start
```

Not covered in this workshop:

- Routing (for multi-page applications)
- Production build
- Error handling (try-catch)
- Security