

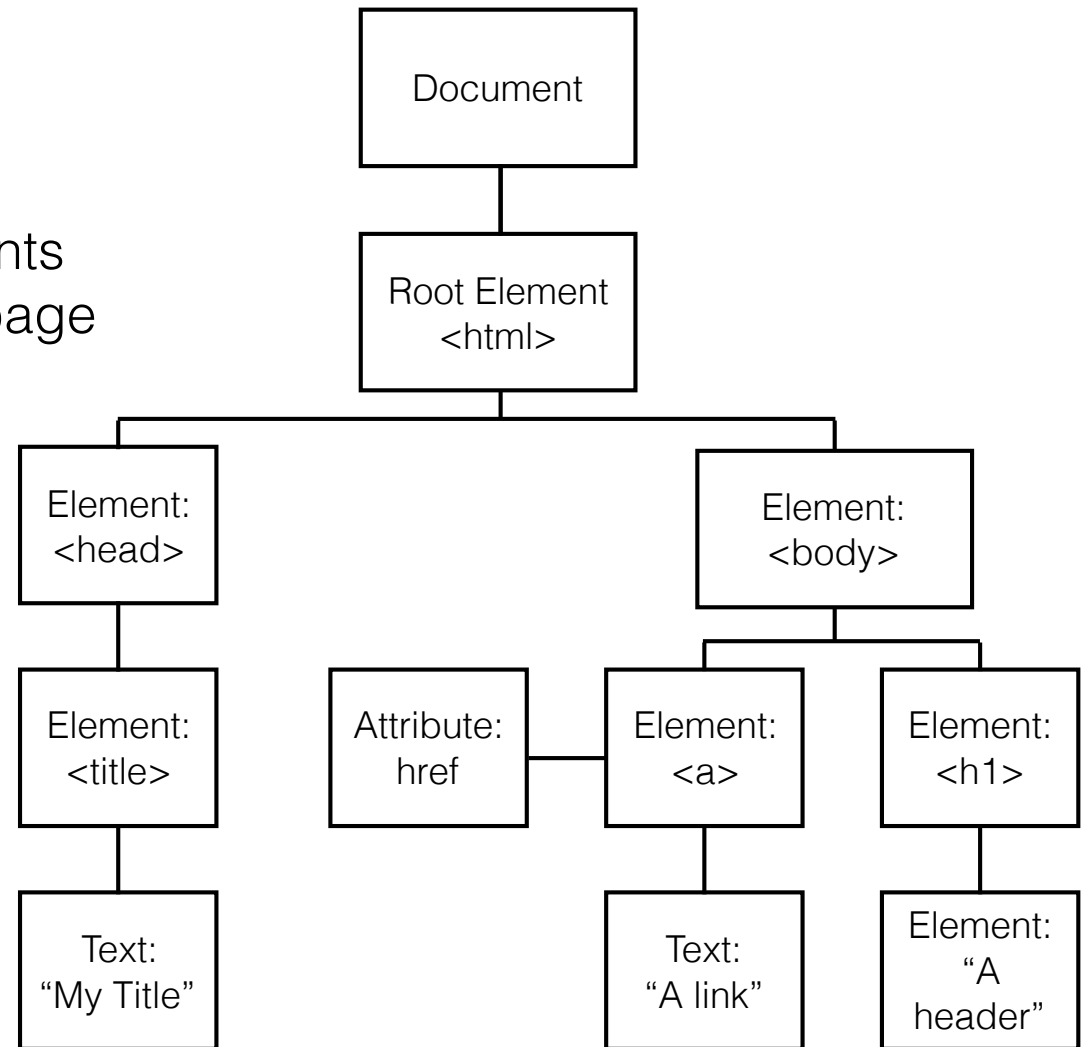
React

```
class App extends Component
```

What happens when the DOM updates?

- Traverse DOM to add elements
- Reflow - compute layout of page

Is this a problem?



```
<!doctype html> <html>  
<head>
```

```
<title>My title</title> </head>  
<body>
```

```
<a href="xyz.html">A link</a> <h1>A header</h1>  
</body> </html>
```

Strategies for preventing excessive DOM updates

- Make changes to the DOM in batch as much as possible
 - Use classes for CSS so that you can just add or remove the class from an element
 - Build a complex component and add it to the DOM at the end

<https://www.learnsteps.com/javascript-understanding-repaint-and-reflow-of-dom-for-increasing-performance/>

Classes

- Recall that there are multiple ways to create objects in Javascript.
- ES6 introduced a “class” structure that looks a lot more like classes from other languages.
- Remember that underlying the new syntax is still the prototype model of objects.

Javascript ES6 and OO

Underneath is Function

signature of the Point
function

```
class Point {  
  constructor(a,b) {  
    this.x = a;  
    this.y = b;  
  }  
  product() {  
    return this.x * this.y;  
  }  
}
```

Class methods have a
simpler syntax

Objects must be created
using new

```
let pt = new Point(10, 20);
```

Inheritance

```
class ThreeDPoint extends Point
  constructor(a,b,c) {
    super(a, b);
    this.z = c;
  }

  product() {
    return super.product() * this.c;
  }
}
```

Call parent
constructor

Call function
defined in parent

this

- Recall that `this` is related to the context in which a function is called.
- ES6 classes make `this` look a little more like what you would expect.
- But you may need to explicitly bind `this` to the enclosing object. (See example)

<https://medium.freecodecamp.org/react-binding-patterns-5-approaches-for-handling-this-92c651b5af56>

What is React?

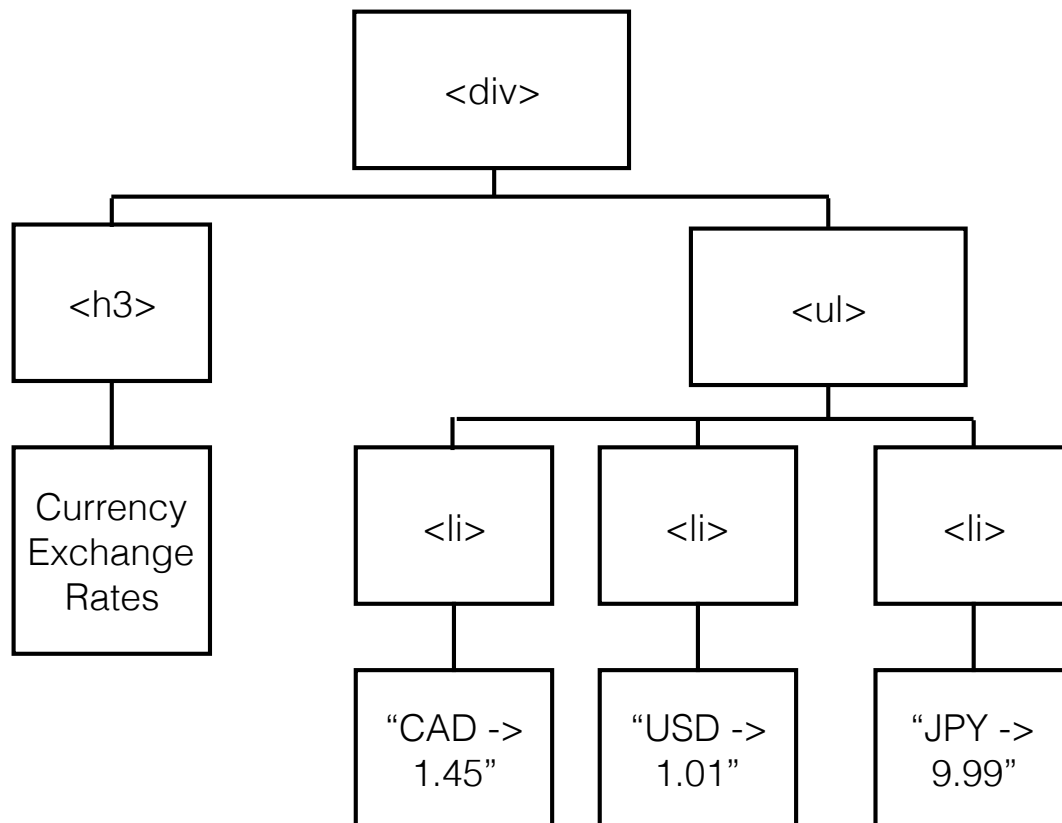
- React is a library used to write the **view** layer
- Think in terms of components

React

- Declarative
- Pure functional
- JSX makes it easier to see what will be rendered

Virtual DOM

DOM



Virtual DOM

A simplified abstract version of the DOM.
Mostly browser agnostic.

React does computations on the virtual DOM and then inserts the resulting component into the real DOM, if something has changed.

- `state` is initialized in the constructor.
- `props` are passed down to nested components and are immutable

Component life cycle

- Mounting
 - constructor
 - componentWillMount
 - render
 - componentDidMount
- Updating - conditional on the change in data
 - componentWillReceiveProps
 - shouldComponentUpdate
 - componentWillUpdate
 - componentDidUpdate
- Unmounting
 - componentWillUnmount