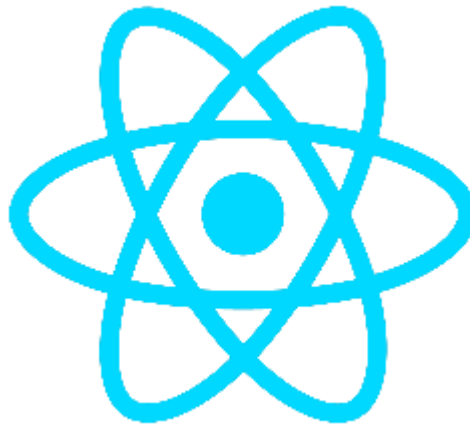


Beginner tutorial to ReactJS



What is React?

- **Javascript library for NodeJS for building interactive web interfaces**
- **Made by Facebook**
- **React is only a View engine (if you want a full stack, some frameworks like Meteor works nicely with React)**



Why React?

- **UI are split into components**
 - Easily visualize the webpage
 - Components have internal states, and will automatically update when their states change
 - More readable code
- **Fast performance**
 - Only re-render components whose state have changed
- **Really popular right now (5th most starred project of all time on GitHub)**



Setting up React

- **Node**
- **create-react-app (also made by Facebook) to quickly create React apps without any configuration**
- **or Meteor**



create-react-app

```
npm install -g create-react-app  
create-react-app hello-world
```



- **npm start**

- Starts the dev server at localhost:3000

- **npm run build**

- Builds your app and puts the result in /build
 - You can then deploy the built HTML, CSS and JS files in your own Node project



Components

- **The building block of React**

```
class Welcome extends React.Component {  
  render() {  
    return <h1>Hello, world!</h1>;  
  }  
}
```

- **The following code then replaces the element with id="root" in the HTML document with the Welcome component above**

```
ReactDOM.render(  
  <Welcome />,  
  document.getElementById('root')  
);
```



Note on JSX

- **An extension to Javascript syntax used by React**
- **Used to directly write HTML in Javascript for React elements**

```
const name = "John";
```

```
const element = <h1>Hello, {name}</h1>;
```

- **React also uses ES6 syntax, though neither ES6 nor JSX is mandatory (but HIGHLY recommended)**



Note on JSX

- **HTML Attributes are written in camelCase in JSX**

```
<h1 className="title" onClick="return false;">  
Hello, world!  
</h1>
```



Props and States

- **Props are attributes passed to the components in the form of HTML tags**
 - Cannot be modified
- **States are variables that each components keep track of privately**
 - Can be modified, and the components will automatically update (re-render) itself



Note on states

- **Initialize in constructor**
- **Must be updated with `setState()`**
- **`setState()` merges with current state (you don't have to re-set the entire state)**
 - Current state: `{name: "John", age: 21}`
 - `this.setState({weight: 100});`
 - New state: `{name: "John", age: 21, weight: 100}`



Events

- **Works a bit differently than DOM**
 - You pass a function directly to handle events
 - CamelCase (onClick instead of onclick)



Code example here



More information

- [**https://facebook.github.io/react/docs/hello-world.html**](https://facebook.github.io/react/docs/hello-world.html)

