ReactJS

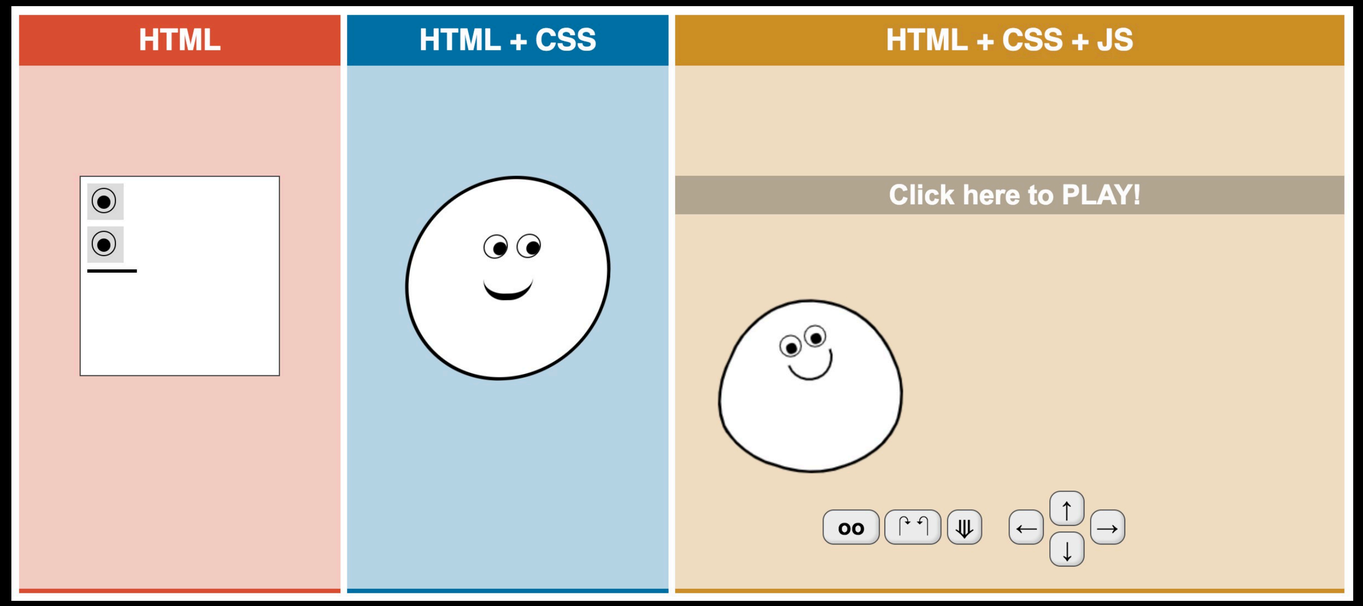
ReactJS

- Frontend & Issues in vanilla JavaScript
- Features
  - Component-based
  - Declarative
  - Learn it once, write anywhere
- Live Coding with Khuyen: Let’s build a todo list!!
  - State / Props
- Q & A
- Extra Slides

Front-end Web App

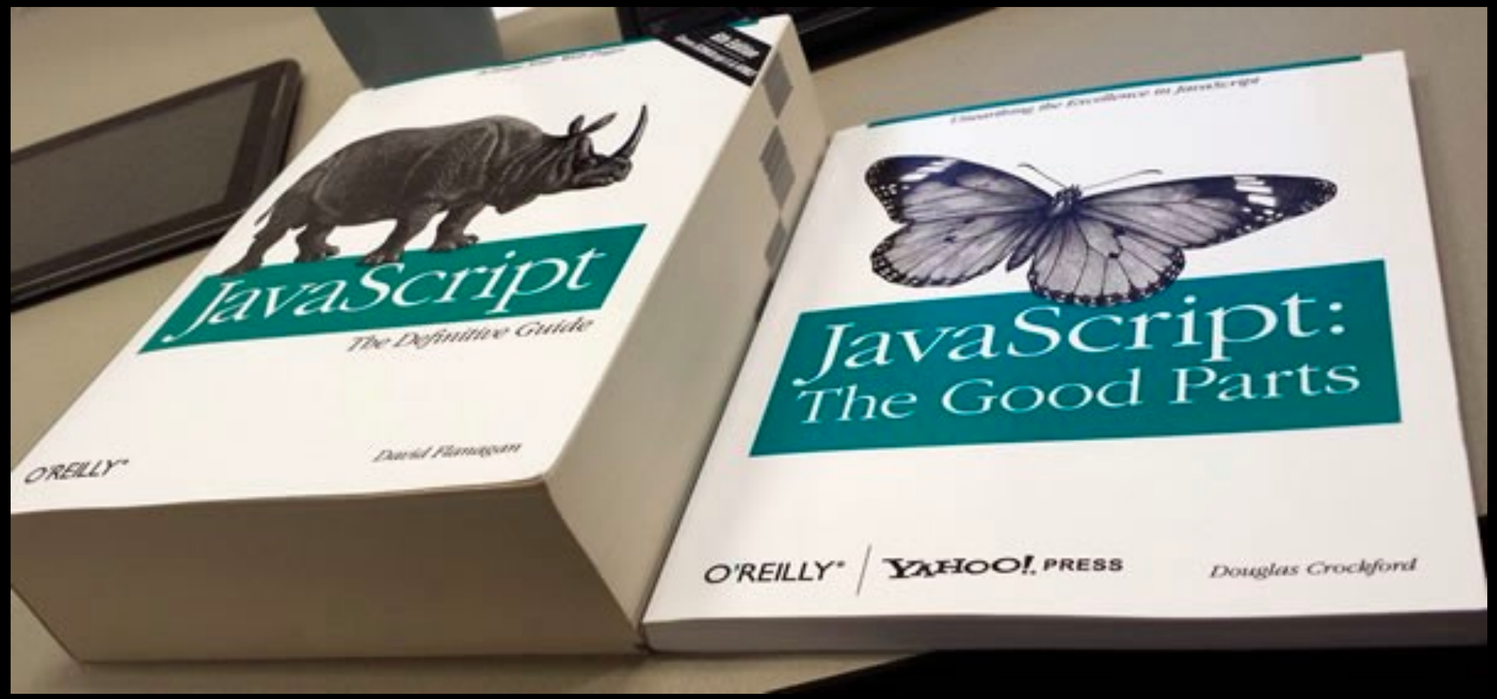
HTML + CSS + JavaScript

- HTML: Skeleton of the website
- CSS: Skin of the website
- JavaScript: Spirit of the website

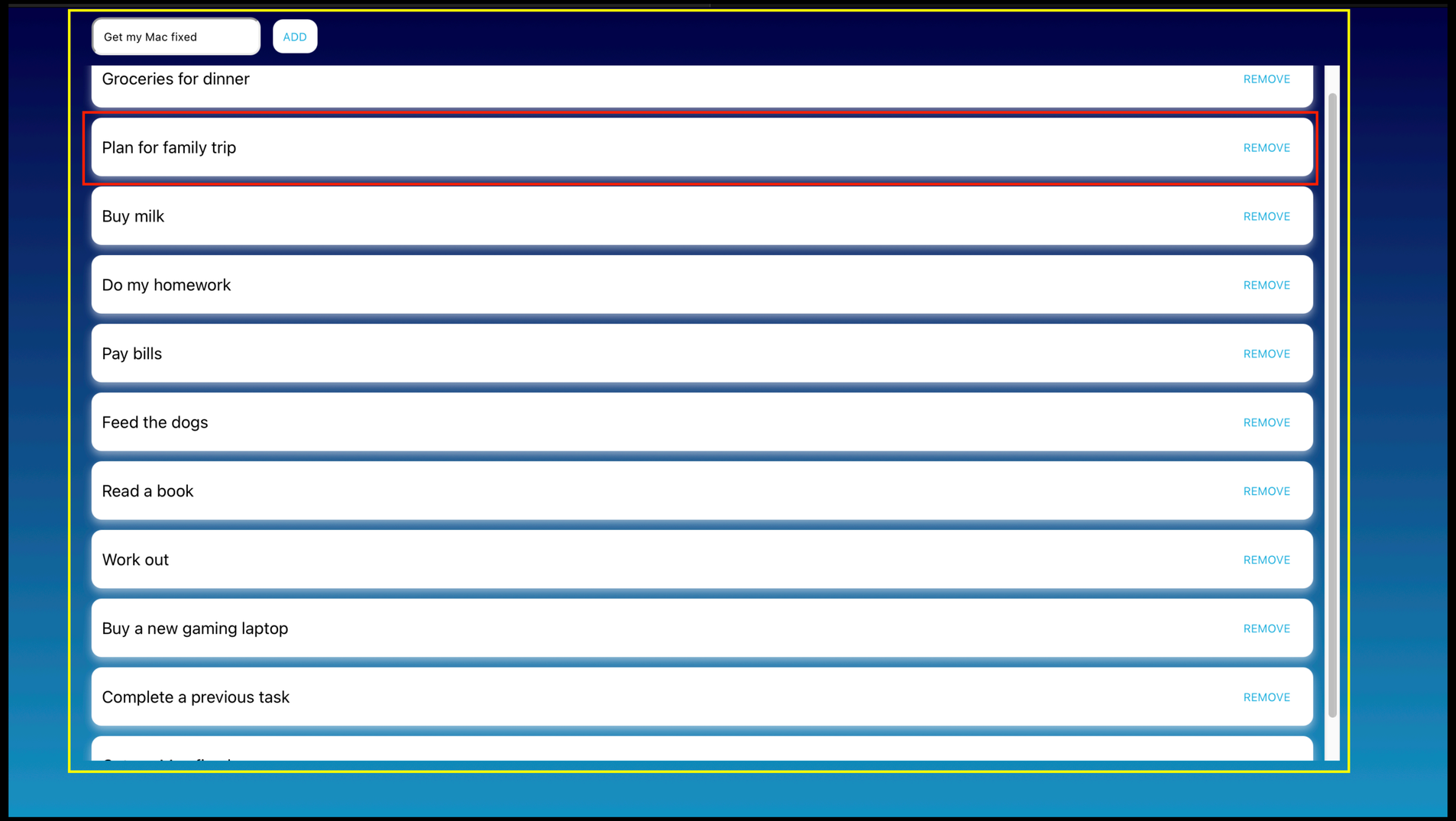


Vanilla JavaScript is not enough

- User interface is getting more and more complicated
- e.g. **62 million** lines of code (Facebook)
  - Extract duplicate codes
  - Add a table of contents



Component-based



Component-based

Table of contents

```
src
  App.js
  index.css
  index.js
  TodoItem.css
  TodoItem.js
  TodoList.css
  TodoList.js
```

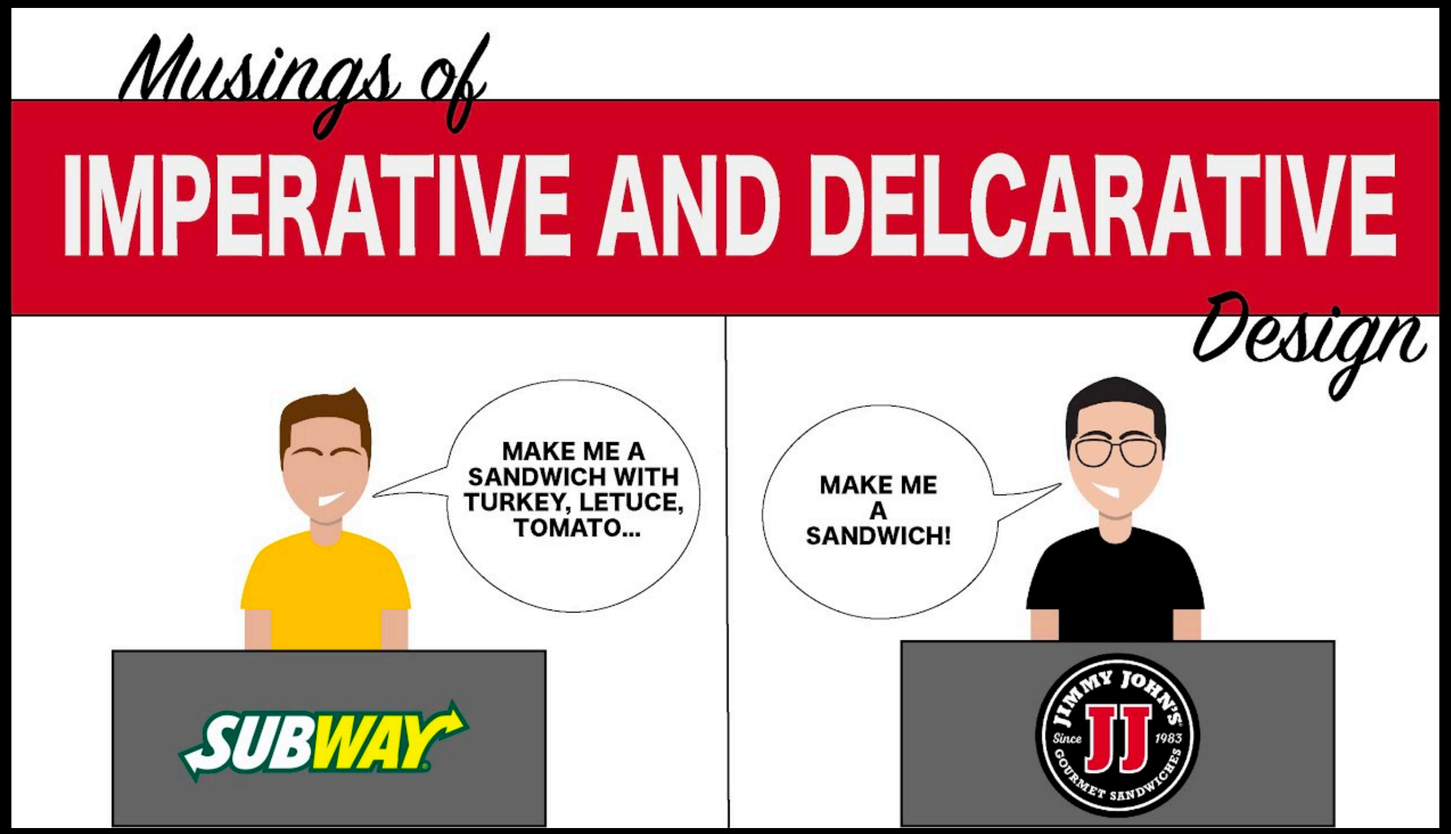
```
import './App.css';
import TodoList from './components/TodoList';

function App() {
  return (
    <TodoList></TodoList>
  );
}

export default App;
```

Declarative when rendering views

- Imperative (Vanilla JavaScript) v.s. Declarative (React)



https://www.wikihow.com/Order-a-Subway-Sandwich

Declarative when rendering views

	Vanilla JavaScript	React
todoItems = empty list	const todoItems = []	const [todoItems, setTodoItems] = useState([])
Describe the view	In HTML:	{todoItems}
Update the view accordingly	updateViewWithData(todoItems)	X
When user click the button: Update data	todoItems.push(newItem)	setTodoItems([…todoItems, newItem]);
Update the view again	updateViewWithData(todoItems)	X

Vanilla JavaScript: https://codepen.io/yi-hung-chou/pen/PomqEyX

Extract the duplicate codes in vanilla javascript

Learn once, write anywhere

- Easy to integrate with legacy code
  - CrowdNews
- Portable
  - ReactNative (Mobile App)
  - Electron + React (Desktop App)
  - ReactVR (Virtual Reality)

React

A JavaScript library for building user interfaces

Get Started Take the Tutorial

Declarative

React makes it painless to create interactive UIs. Design simple views for each state in your application, and React will efficiently update and render just the right components when your data changes.

Declarative views make your code more predictable and easier to debug.

Component-Based

Build encapsulated components that manage their own state, then compose them to make complex UIs.

Since component logic is written in JavaScript instead of templates, you can easily pass rich data through your app and keep state out of the DOM.

Learn Once, Write Anywhere

We don't make assumptions about the rest of your technology stack, so you can develop new features in React without rewriting existing code.

React can also render on the server using Node and power mobile apps using React Native.

Extra Slides

JS & React

Var vs Const vs Let

https://www.freecodecamp.org/news/var-let-and-const-whats-the-difference/

Why don’t we use var anymore?
https://blog.usejournal.com/awesome-javascript-no-more-var-working-title-999428999994

Hoisting

https://developer.mozilla.org/en-US/docs/Glossary/Hoisting

Class & Prototype chain

A class declaration is syntactic sugar over prototypal inheritance with additional enhancements.

References:
MDN Web Docs
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Inheritance_and_the_prototype_chain

JavaScript Tutorial:
https://www.javascripttutorial.net/es6/javascript-class/

```javascript
class Person {
    constructor(name) {
        this.name = name;
    }
    getName() {
        return this.name;
    }
}
```

~=

```javascript
function Person(name) {
    this.name = name;
}

Person.prototype.getName = function () {
    return this.name;
};

var john = new Person("John Doe");
console.log(john.getName());
```

JSX?

https://reactjs.org/docs/jsx-in-depth.html

A syntactic sugar for you to write code that is more readable

```
<MyButton color="blue" shadowSize={2}>
  Click Me
</MyButton>
```

Is equal to

```
React.createElement(
  MyButton,
  {color: 'blue', shadowSize: 2},
  'Click Me'
)
```

Why Hook

I strongly recommend you to go through this to understand what problems they are trying to solve and appreciate the usage of Hook
https://www.youtube.com/watch?v=dpw9EHDh2bM

If you really don't have time, you can read this medium post.
https://medium.com/@dan_abramov/making-sense-of-react-hooks-fdbde8803889

Check the visualization to see the benefits that React Hooks bring us
https://twitter.com/prchdk/status/1056960391543062528

Why this.<functionName>.bind(this) in React

Short answer
https://reactjs.org/docs/faq-functions.html#why-is-binding-necessary-at-all

Longer one
https://www.freecodecamp.org/news/this-is-why-we-need-to-bind-event-handlers-in-class-components-in-react-f7ea1a6f93eb/

Why Async / Await

To prevent callback hell in Promise

```javascript
function hell(win) {
 // for listener purpose
 return function() {
   loadLink(win, REMOTE_SRC+'/assets/css/style.css', function() {
     loadLink(win, REMOTE_SRC+'/lib/async.js', function() {
       loadLink(win, REMOTE_SRC+'/lib/easyXDM.js', function() {
         loadLink(win, REMOTE_SRC+'/lib/json2.js', function() {
           loadLink(win, REMOTE_SRC+'/lib/underscode.min.js', function() {
             loadLink(win, REMOTE_SRC+'/lib/backbone.min.js', function() {
               loadLink(win, REMOTE_SRC+'/dev/base_dev.js', function() {
                 loadLink(win, REMOTE_SRC+'/assets/js/deps.js', function() {
                   loadLink(win, REMOTE_SRC+'/src/' + win.loader_path + '/loader.js', function() {
                     async.eachSeries(SCRIPTS, function(src, callback) {
                       loadScript(win, BASE_URL+src, callback);
                     });
                   });
                 });
               });
             });
           });
         });
       });
     });
   });
 };
}
```

https://dev.to/jessrichmond/node-js-the-promise-that-callback-hell-is-not-inevitable-22jh

```javascript

async function cluedo() {
  const who = await character();
  const where = await room();
  const using = await weapon();

  const whoDidIt = await Promise.all([character(), room(), weapon()])
  console.log(`${ who } ${ where } ${ using }`);
}

cluedo();

```

https://medium.com/@gemma.stiles/understanding-async-await-in-javascript-d2dbf370672b