Web Programming Spring 2021 #20

Chi-Jen Wu

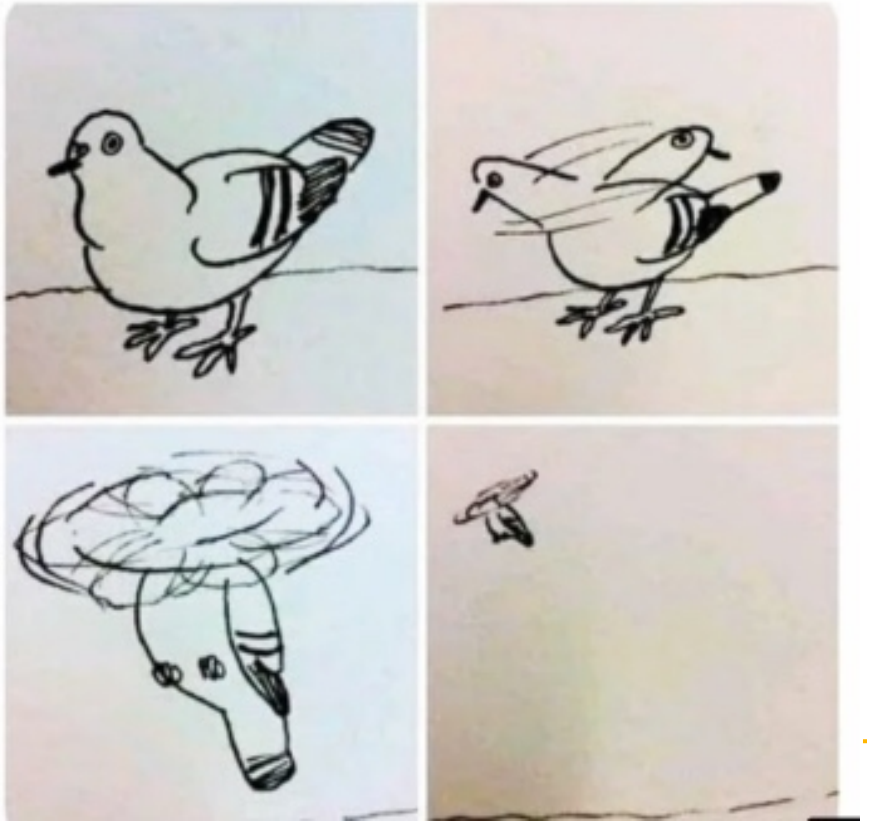
Topics

- The concepts of Web Services
- Web data protocols
  - HTTP, WebSocket, WebRTC
  - HTML, CSS
- Web JavaScript programming
- Cookies and sessions
- **Web Frontend frameworks**
- Web Backend frameworks
- RESTful API design

React / ReactJS

- React Ecosystem
- 開發環境
- **核心概念 — 元件(Components)**
- 單向資料流 (Unidirectional data flow)
- Virtual DOM
- Declarative UI / JSX
- React Router

當你寫的程式完全是一團混亂但依舊發揮功用時



練習

react componentes

把

hello cgu 也變成模組化

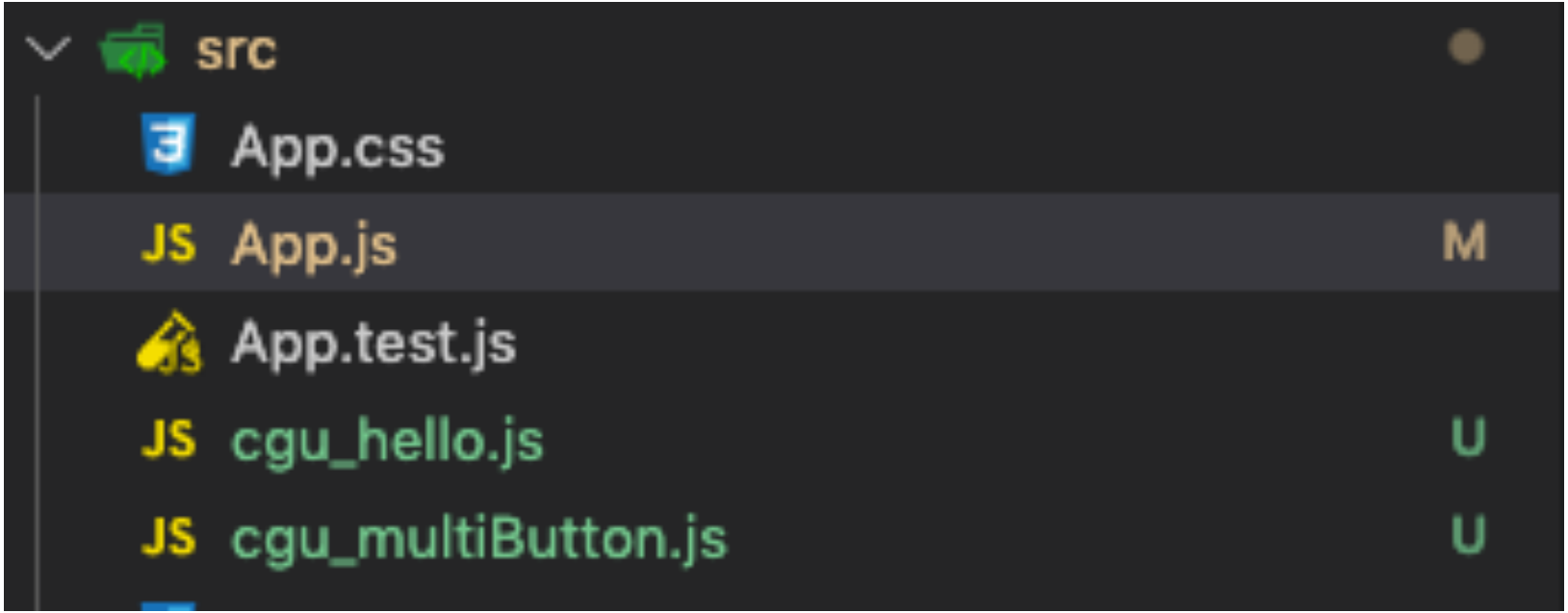

localhost:3000
hello CGU!!
第1個按鍵
第2個按鍵
第3個按鍵
第4個按鍵
第5個按鍵
第6個按鍵
第7個按鍵
第8個按鍵
第9個按鍵
第10個按鍵

程式碼模組化

- 新增一個 cgi_hello.js 在 ./src 底下
- 把 hello cgu copy進去
- 在App.js 裡新增
  - import HelloCGU from ‘./cgu_hello’

- 所以模組裡面可以在模組 像是樂高一樣
  - 把模組堆疊起來 變成一個新的模組

程式碼模組化


src
App.css
JS App.js M
App.test.js
JS cgu_hello.js U
JS cgu_multiButton.js U

Hello 程式碼模組化 index.js

```
import App from './App';

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);
```

一個模組

Hello 程式碼模組化 App.js

```
import MultiButton from './cgu_multiButton'
import HelloCGU from './cgu_hello'

function App() {
  return (
    <div className="App">
      <div>
        { HelloCGU() }
      </div>
      <div>
        { MultiButton(10) }
      </div>
    </div>
  );
}
```

增加 import 模組

一個模組

一個模組

大模組

cgu_hello.js

```
const styleArgument = { fontSize: '100px', color: 'red' };

const HelloCGU=()=>{
  return <h1 style = { styleArgument } > hello CGU!! </h1>;
}

export default HelloCGU;
```

一定要大寫喔！！

cgu_multiButton.js

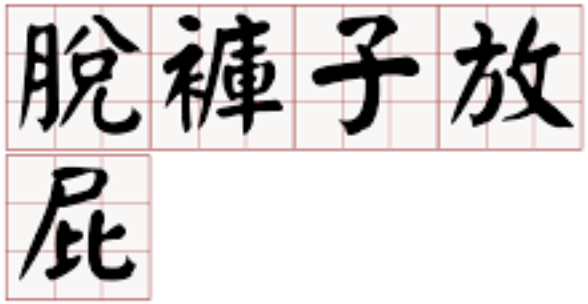
```
const changeText=(event)=>{
  console.log(event.target)
  event.target.innerText = event.target.innerText + "被點了"
}

const MultiButton=(num)=>{
  var output=[];
  for(let i=1;i<num+1;++i)
      output.push(<button onClick={changeText}>第{i}個按鍵</button>)
  return output;
}

export default MultiButton;
```

程式碼模組化 其實我們還沒真的模組化XD

- 我們只是把 react 拆分成
  - App.js
  - cgu_hello.js
  - cgu_multiButton.js
  - 然後再 import 來用
  - 其實是在脫褲子放屁 XD



程式碼模組化 其實我們還沒真的模組化XD

- 假設我們要讓每一個按鈕計算自己被按了幾次
  - 1. 元件自己計算？
  - 2. 有個全域變數來紀錄？
  - 3. 叫隔壁同學幫忙記？

多幾？

程式碼模組化

- 例如我們要讓每一個按鈕計算自己被按了幾次
  - 1. 元件自己計算 ！！

- 模組化後 每個元件應該要可以保持自己的**狀態**
  - **自己管自己的狀態！！**

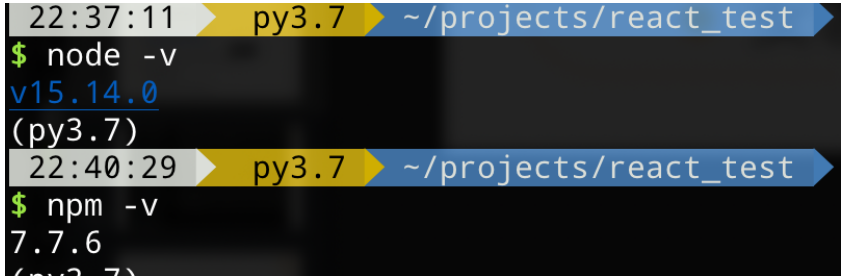
我們來真的模組化吧！！


但是，
我拒绝！

複習一下
今天遇到的

React 開發環境

- 安裝 node.js & npm
  - https://nodejs.org/en/
  - npm -v
  - npm install -g create-react-app
- vscode (option)
  - 安裝ESlint
  - 安裝 JS JSX Snippets



安裝 後會看到

```
 23:37:41  py3.7  ...projects/react_test/hello-world  master ✔
$ ll
total 1008
-rw-r--r--     1 cjw  staff   3.3K  4 13 22:36 README.md
drwxr-xr-x  1052 cjw  staff    33K  4 13 22:37 node_modules
-rw-r--r--     1 cjw  staff   815B  4 13 22:36 package.json
drwxr-xr-x     8 cjw  staff   256B  4 13 22:36 public
drwxr-xr-x    10 cjw  staff   320B  4 13 22:36 src
-rw-r--r--     1 cjw  staff   496K  4 13 22:37 yarn.lock
```

npm start

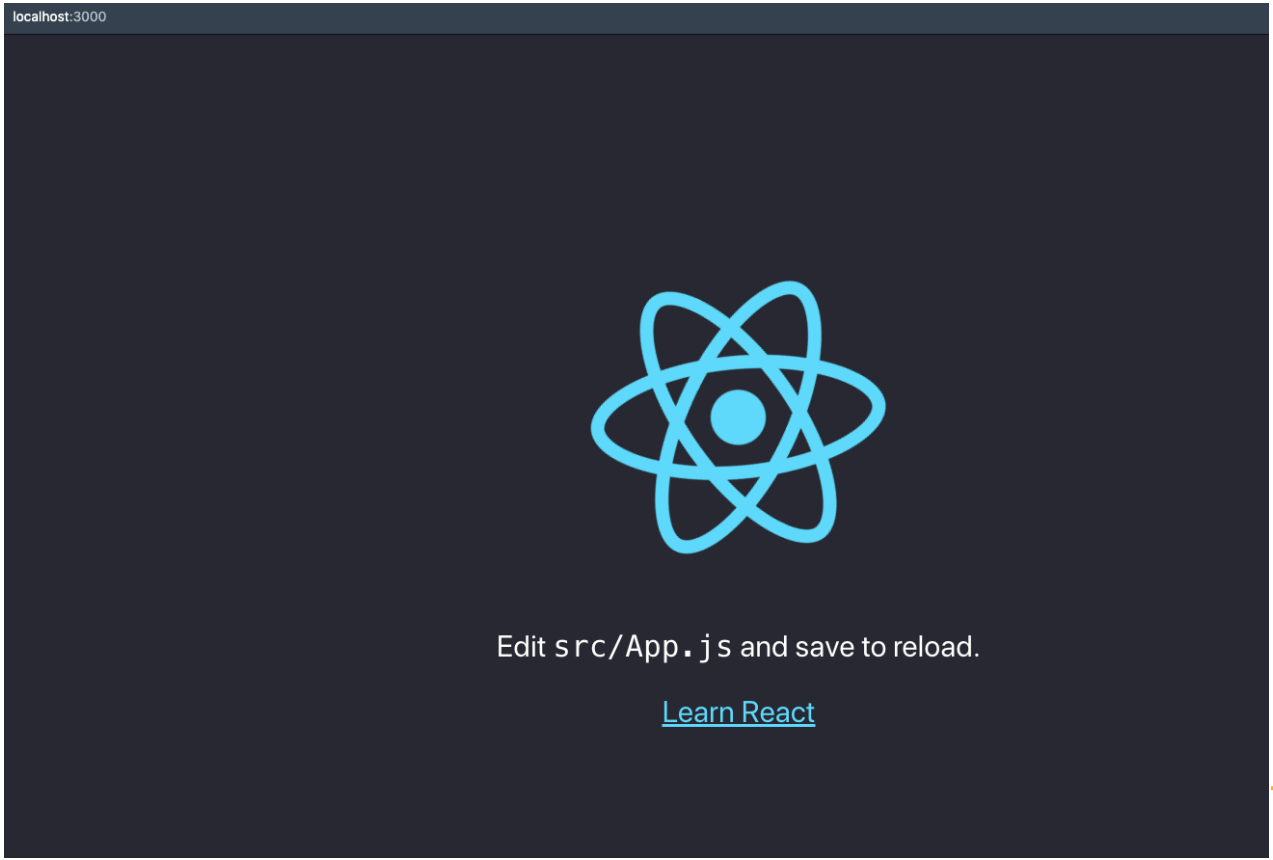
```
Compiled successfully!

You can now view hello-world in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.114:3000

Note that the development build is not optimized.
To create a production build, use yarn build.
```

In browser : http://localhost:3000



開發流程

- 建立react專案
- npm start
- 寫程式碼
- 在local端確認執行結果 http://localhost:3000
- npm run build
- 部署到伺服器 (Build目錄內的檔)

src/index.js

React的程式進入點

```
7   ReactDOM.render(
8       <h1> Hello world!</h1>,
9       document.getElementById('root')
10  );
```

React的程式進入點

```
<script>
import ReactDOM from 'react-dom'

ReactDOM.render(element, document.getElementById('root'));

</script>
```

`ReactDOM.render()`
第一個參數是你要顯示的 React element，
而第二個參數則是要顯示到哪個 HTML DOM element 上

React elements (React 元素)

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

`<h1>Hello, world!</h1>`
這一段 code 就是所謂的 JSX，注意它不是字串喔 (前後沒有引號包著)

React 以元件 (components) 為中心思考

Hello CGU!

```
const styleArgument = { fontSize: '100px', color: 'red' };

ReactDOM.render(
  <h1 style = { styleArgument } > Hello CGU!</h1>,
  document.getElementById('root')
);
```

程式碼模組化

- 重複使用元件
  - 重點!!
- 你寫的元件可以分享給我用
- 我只要 import 你寫的react component

- 讓我們把剛剛的multiButton 程式模組化一下

程式碼模組化

- 新增一個 cgi_multiButton.js 在 ./src 底下
- 把 multiButton, changeText copy進去
- 把App.js 裡的 multiButton, changeText 刪除
- 在App.js 裡新增
  - import multiButton from './cgu_multiButton'

下課！

Thanks!

Open for any questions

CJ Wu

cjwu@mail.cgu.edu.tw