

Framework Training
React
London July 2018
Exercise M-hocs

- This exercise creates a **Higher Order** component that works with different data and views.

Installation

- Install and run the starter version of the project.

```
npm install
npm start
```

Higher Order functions

- Higher Order functions (HOFs) return another function not just a value.
- double is an ordinary function which doubles numbers

```
let double = n => n*2;
```

- mult is a HOF which returns another function that multiplies numbers by some parameter.

```
let mult = a => b => a*b
```

- mult(2) returns a function that doubles numbers.

```
let double = mult(2)
double(2);
```

```
let quad = mult(4)
```

```
quad(2)
```

- HOFs can be immediately invoked

```
mult(2)(2);
```

Review the existing project

- Component DataComponent reads a local JSON file using the Fetch API.
- Its render method contains an instance of the stateless view component called Regions.

```
render() { return <Regions data={this.state.data} /> }
```

- An instance of DataComponent is created in index.js.

```
<DataComponent json="spain-2017.json" />
```

- We can change DataComponent into a Higher Order component which will work with different views and data.
- CreateDataComponent is a function which takes one JSON url as its argument.
- It returns another function which takes a View component as its argument.

```
let CreateDataComponent = jsonData => ViewComponent => (  
  class DataComponent extends Component { .. }  
)
```

- Change the render method. ViewComponent will be passed in as an argument.

```
render() { return <ViewComponent data={this.state.data} />;}
```

- Invoking CreateDataComponent return a new component that reads a specific JSON file.

```
let Spain2017 = CreateDataComponent("spain-2017.json");
```

- We can pass a View component to Spain2017 to create a new component that displays the JSON data in a specific view.

```
let Spain2017Regions = Spain2017(Regions);
```

- We can create an instance of this new component in index.js

```
import { Spain2017Regions } from "./Hofs";  
render(<Spain2017Regions />, document.querySelector("#a"));
```

Exercises

- Create a new stateless view that displays temperature information.

```
const Temps = ({ data:spain }) => (  
  <ul className="flex">  
    {spain.map((city,n) => (  
      <li key={n}>{city.name}<span>{city.temp}</span></li>
```

```
    )})  
  </ul>  
)
```

- Create a new component using the same 2017 data with this view.

```
let Spain2017 = CreateDataComponent("spain-2017.json");  
  
let Spain2017Regions = Spain2017(Regions);  
let Spain2017Temps = Spain2017(Temps);  
  
export { Spain2017Regions, Spain2017Temps }
```

- Export and import the new component names into index.js

```
import {  
  Spain2017Regions,  
  Spain2017Temps  
} from "../Hofs";  
  
render(<Spain2017Regions />, document.querySelector("#a"));  
render(<Spain2017Temps />, document.querySelector("#b"));
```

- Create component instances that work with the 2018 data for both views using immediately-invoked syntax.

```
let Spain2018Regions = CreateDataComponent("spain-2018.json")  
  (Regions);  
  
let Spain2018Temps =  
  CreateDataComponent("spain-2018.json")(Temps);
```