

CURSO REACT

Clase 03

02

Tipos de programación

Realizaremos el consume de un API

Tipos de Programming

01

Declarative

[Detalle...](#)

02

Imperative

[Detalle...](#)

La programación declarative es una manera en que se *describe* lo que queremos conseguir al hacer código.

Ejemplo: "Me traes un vaso con agua por favor."

Declarative

La programación imperativa es una manera en que se *describe* como funciona la cosa al realizar código.

Ejemplo: Trae el vaso hasta mi mesa

Imperative

```
1 // Imperative
2 const toLowerCase = input => {
3   const output = [];
4   for (let i = 0; i < input.length; i++) {
5     output.push(input[i].toLowerCase());
6   }
7   return output;
8 };
9
10 // Declarative
11 const toLowerCase = input => input.map(value => value.toLowerCase());
```

```
1  const myLatLng = {
2    lat: 51.5258541,
3    lng: -0.08040660000006028
4  };
5
6  // Imperative
7  const map = new google.maps.Map(document.getElementById("map"), {
8    zoom: 5,
9    center: myLatLng
10 });
11
12 const marker = new google.maps.Marker({
13   position: myLatLng,
14   title: "Hello GDG Lima 2017"
15 });
16
17 marker.setMap(map);
18
19 // Declarative
20 import Gmaps from "react-gmaps";
21
22 <Gmaps lat={myLatLng.lat} lng={myLatLng.lng} zoom={12}>
23   <Marker lat={coords.lat} lng={coords.lng} title="Hello GDG Lima 2017" />
24 </Gmaps>;
```

03

Condicionales

IF

```
// Bad
let button;
if (isLoggedIn) {
  button = <LogOutButton />;
}
return <div>{button}</div>;
```

```
// Good
<div>
  {isLoggedIn && <LogOutButton />}
</div>
```

```
// Bad
let button;
if (isLoggedIn) {
  button = <LogOutButton />;
} else {
  button = <LogInButton />;
}
return <div>{button}</div>;
```

```
// Good
<div>
  {isLoggedIn ? <LogOutButton /> : <LogInButton />}
</div>
```

```
// Bad
<div>
  {dataIsReady && (isAdmin || userHasPermissions) && <topSecretComponent />}
</div>

// Good
canShowSecretComponent() {
  const {dataIsReady, isAdmin, userHasPermissions} = this.props;
  return dataIsReady && (isAdmin || userHasPermissions);
}

<div>
  {this.canShowSecretComponent() && <topSecretComponent />}
</div>

// Better
get canShowSecretComponent() {
  const {dataIsReady, isAdmin, userHasPermissions} = this.props;
  return dataIsReady && (isAdmin || userHasPermissions);
}

<div>
  {this.canShowSecretComponent && <topSecretComponent />}
</div>

// Much Better
// https://github.com/ajwhite/render-if
class MyComponent extends Component {
  render() {
    const ifTheUniverseIsWorking = renderIf(1 + 2 === 3);
    return ifTheUniverseIsWorking(<span>The universe is still wroking</span>);
  }
}
```

FOR

```
1 | <ul>
2 |   {items.map(item => (
3 |     <li>{item.title}</li>
4 |   ))}
5 | </ul>
```

Gracias

Brayan LP

Soy Semi senior desarrollador Front-End en Red Científica Peruana, estude la Carrera de ingeniería de sistemas y actualmente estoy certificado en REACT

