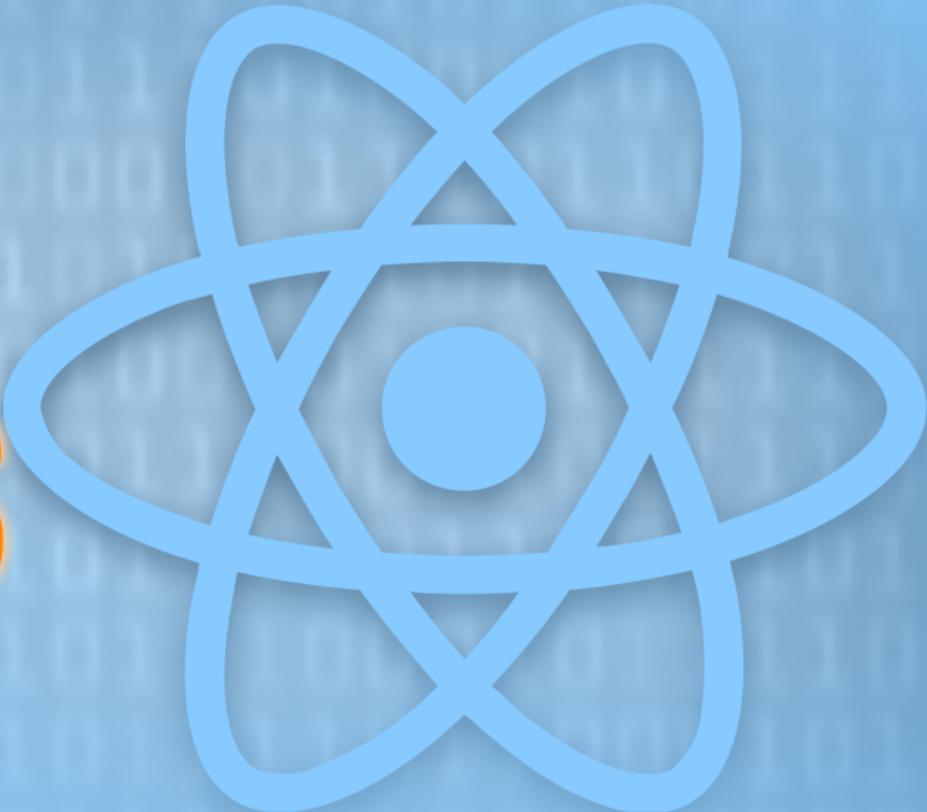


PROPS



Quando estamos falando de react, estamos falando de um sistema onde os componentes se comunicam entre si, apesar de possuir alguns componentes estáticos a principal ideia de usar o react é quando a alteração de um estado altera o resultado do componente, fazendo com que os componentes reajam a alteração;



Sabendo disso podemos iniciar a explicação de props. Por sua vez props é a abreviação de properties (propriedades em inglês), assim como o nome diz, se trata de definir uma propriedade ao componente criado.

Ao lado temos o exemplo da criação de um componente sem props.



COMPONENTE

```
4  const Footer = () => {  
5      return (  
6          <footer>  
7              <p>QODELESS 2021</p>  
8          </footer>  
9      );  
10 };  
11  
12 export default Footer;  
13
```

Agora vamos supor que estamos desenvolvendo um sistema no qual pesquisamos uma cidade via API, e em tempo real é retornado dados meteorológicos como temperatura do ar, umidade do ar, latitude, longitude, etc...



E queremos exibir o como está o clima no dia atual e como ficará nos próximos 5 dias, para isso utilizaremos apenas 2 componentes que se comunicarão com 1 único componente que terá o único objetivo de realizar a pesquisa na API e passar para os outros componentes, ou seja, com apenas 3 componentes criados teremos que exibir 6 tipos de dados diferentes.



COMPONENTE COM PROPS

Nesse caso entra os props, as propriedades no qual o componente deverá receber para que seja possível ser visualizado em tela ou criada uma lógica para tratar as informações necessárias.

Para isso devemos definir na criação dos componentes as propriedades que eles irão receber.



```
19 const Weather: FC<TWeather> = (props) => {
20   const { temperaturaMIN, temperaturaMAX, umidade, cidade } = props;
21   return (
22     <div>
23       <p>{temperaturaMIN}</p>
24       <p>{temperaturaMAX}</p>
25       <p>{umidade}</p>
26       <p>{cidade}</p>
27     </div>
28   );
29 };
30
31 export default Weather;
32
```

Na linha 20 está sendo feito o destructuring para acessar os items dentro do tipo objeto TWeather definido como prop na linha 19

Para passar como props os items do componente pai para o componente criado para o componente criado fazemos desta maneira.



```
27    <Weather  
28        temperaturaMIN={temperaturaMIN}  
29        temperaturaMAX={temperaturaMAX}  
30        umidade={umidade}  
31        cidade={cidade}  
32    />
```

OBS: podemos definir os props que os componentes receberão no próprio arquivo do componente ou podemos criar interfaces ou types para isso, abaixo há exemplos das 2 maneiras de se definir os props que o componente irá receber.

```
type TWeather = {  
    ~~~~~~  
    temperaturaMIN: string;  
    temperaturaMAX: string;  
    Umidade: string;  
    Cidade: string;  
}
```

```
interface IWeather {  
    ~~~~~~  
    temperaturaMIN: string;  
    temperaturaMAX: string;  
    Umidade: string;  
    Cidade: string;  
}
```