React Props

Como o react interpreta o jsx?

```
React.createElement(type, [props], [...children])
```

Type - a tag HTML que será incluída

[props] - os atributos que você quer que o elemento tenha.

[children] - todas as tags HTML dentro do elemento.

Exemplo:

JSX

```
export default Componente(){
return(
<h1 id="jsx">This is JSX</h1>
)
};
```

O React irá converter em:

```
React.createElement("h1", { id: "jsx" }, "This is JSX");
```

O objeto gerado será:

```
{
  type: 'h1',
  props: {
   id: 'jsx',
   children: 'This is JSX'
  }
}
```

Como adicionar elementos Javascript no código

Para adicionar código Javascript nós precisamos inseri-lo dentro de chaves.

```
const App = () => {
 const number = 10;
 return (
  <div>
   <p>Number: {number}</p>
  </div>
 );
};
```

O que eu posso adicionar dentro das chaves:

- String
- Números
- Array
- Propriedade de um objeto
- Uma chamada de função
- Uma array Map
- If ternário

O que não pode:

- Loop(for, while ou qualquer outro)
- Declaração de variável
- Declaração de função
- Um if traditional

- Um objeto

O que são as props?

São as propriedades de um componente, que tem a função de transmitir informações entre componentes, criando uma característica **reutilizável** dos componentes.

```
import Welcome from './Welcome';

function App() {
  return (
    <div className="App">
      <Welcome name="John"/>
      <Welcome name="Mary"/>
      <Welcome name="Alex"/>
    </div>
  );
}
```

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}
```

Props por desestruturação

Normalmente as props são passadas por desestruturação, que é quando transformamos cada posição do array em uma variável.

```
function Welcome({name}) {
  return <h1>Hello, {name}</h1>;
}
```

Renderização de listas

Além de passarmos variáveis como props, é bem comum passarmos listas, que normalmente vem de uma API ou Banco de Dados.

```
function Home() {

  const equipe = [
    {nome: 'Caio', email: "a@a.com.br",telefone: 999999 },
    {nome: 'Caio2',email: "a@22a.com.br",telefone: 333 }
  ];

  return (
    <div className="mainInfo">
      <section id='info'>
          <PersonalInfo dadosEquipe={equipe}/>
          <Skills />
        </section>
        <ContactForm />
    </div>
  );
}

export default Home;
```

```
export default function PersonalInfo({dadosEquipe}) {

return (
    <div className="personal-info">
        <h2>Informações Pessoais</h2>
        {dadosEquipe.map((element, index) => (
          <div key={index}>
            <p>Nome: {element.nome}</p>
            <p>Email: {element.email}</p>
            <p>Telefone: {element.telefone}</p>
```

```
            </div>
          )

          )}
    </div>
    );
    }
```

Obs: O React sempre pede o elemento Key, que é uma chave única para reconhecer os elementos dentro do array para que ele saiba qual ele deve atualizar e em qual momento.

Referências

https://www.freecodecamp.org/news/jsx-in-react-introduction/

https://medium.com/reactbrasil/react-entendendo-props-e-state-2a528375ffdd

https://www.freecodecamp.org/portuguese/news/component