

Para saber mais: React Hooks

React Hooks

A recente atualização do React 16.8 trouxe a possibilidade de utilizarmos state em function components.

A forma de fazer isso é utilizando o **useState**, lembrando que temos que importá-lo conforme abaixo:

```
import React, {useState} from 'react';
```

O useState nos retorna o state atual e uma função que deve ser utilizada para atualizar esse estado, e recebe o estado inicial do componente.

Veja um exemplo abaixo:

Vamos criar um componente que irá contar a quantidade de clicks realizados pelo usuário, primeiro vamos importar o que utilizaremos:

```
import React, { useState } from 'react';
```

Feito isso, vamos criar nosso componente!

```
function ContaClicks() {  
  
  return(  
    <div>  
      <p>Você clicou x vezes </p>  
      <button>clicar</button>  
    </div>  
  );  
}
```

Nosso componente está criado, agora vamos utilizar o **useState** para usar state em um function component!

```
function ContaClicks() {  
  
  const [contador, setContador] = useState(0);  
  
  return(  
    <div>  
      <p>Você clicou x vezes </p>  
      <button>clicar</button>  
    </div>  
  );  
}
```

Repare que recebemos um contador (estado atual), e um método para atualizá-lo, e o **useState** recebe 0 que é nosso estado inicial, feito isso, vamos fazer tudo acontecer!

```
function ContaClicks() {  
  
  const [contador, setContador] = useState(0);  
  
  return(  
    <div>  
      <p>Você clicou {contador} vezes </p>  
      <button onClick={() => setContador(contador + 1)}>clicar</button>  
    </div>  
  );  
  
}
```

Feito isso nosso componente está completo! Nosso código completo é:

```
import React, { useState } from 'react';  
  
function ContaClicks() {  
  
  const [contador, setContador] = useState(0);  
  
  return(  
    <div>  
      <p>Você clicou {contador} vezes </p>  
      <button onClick={() => setContador(contador + 1)}>clicar</button>  
    </div>  
  );  
  
}
```