Hooks

Thaís Silveira

useState()

Passa-se o estado inicial para esta função e ela retorna uma variável com o valor do estado atual e outra função para atualizar este valor.

```jsx
import { useState } from 'react';

function Example() {
  const [count, setCount] = useState(0);

  return (
    <>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button>
    </>
  );
}
```

useEffect()

Você "diz" ao React que ele receberá como primeiro parâmetro uma função que será executada assim que o componente for renderizado.

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

function Example() {
  const [count, setCount] = useState(0);

  useEffect(() => {
    document.title = `You clicked ${count} times`;
  });

  return (
    <>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button>
    </>
  );
}
```

useCallback()

Executa uma função dentro de outra função.

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

function Example() {
  const [count, setCount] = useState(0);
  const decrement = useCallback(() => {
    setCount(count - 1)
  }, [count]);

  return (
    <>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button>
      <button onClick={decrement}>Decrement</button>
    </>
  );
}
```

useMemo()

Retorna um valor (um cálculo, um novo array, uma formatação) baseado na depêndencia de entrada (number), evitando processamento desnecessário em futuras chamadas repetidas.

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

export function CalculateFactorial() {
  const [number, setNumber] = useState(1);
  const [inc, setInc] = useState(0);
  const factorial = useMemo(() => factorialOf(number), [number]);
  const onChange = event => {
    setNumber(Number(event.target.value));
  };
  const onClick = () => setInc(i => i + 1);

  return (
    <div>
      Factorial of
      <input type="number" value={number} onChange={onChange} />
      is {factorial}
      <button onClick={onClick}>Re-render</button>
    </div>
  );
}
```

Espero ter ajudado em algo!

Thaís Silveira