

REACT HOOKS SERIES

useEffect Hook (Practical Guide)



Zain Sadaqat

save it for later

Basic Syntax



react-hooks - Syntax.js

```
1 // 1. import useEffect
2 import { useEffect } from 'react';
3
4 function UseEffectSyntax() {
5   // 2. call it above the returned JSX
6   // 3. pass two arguments to it: a function and an array
7   useEffect(() => {}, []);
8
9   // return ...
10 }
```



Zain Sadaqat

save it for later

useEffect Example



react-hooks - Syntax.js

```
1 import { useEffect } from 'react';
2
3 function User({ name }) {
4   useEffect(() => {
5     document.title = name;
6   }, [name]);
7
8   return <h1>{name}</h1>;
9 }
```



Zain Sadaqat

save it for later

The function passed to useEffect is a callback function. This will be called after the component renders.



Zain Sadaqat

save it for later

In this function, we can perform our **side effects** or multiple side effects if we want.



Zain Sadaqat

save it for later

The second argument is an array, called the **dependencies array**. This array should include all of the values that our **side effect** relies upon.



Zain Sadaqat

save it for later

What this array will do is it will check and see if a value (in this case **name**) has changed between renders. If so, it will execute our **use effect** function again.



Zain Sadaqat

save it for later

What this array will do is it will check and see if a value (in this case **name**) has changed between renders. If so, it will execute our **use effect** function again.



Zain Sadaqat

save it for later

This makes sense because if the **name** changes, we want to display that changed **name** and therefore run our **side effect** again.



Zain Sadaqat

save it for later

Did you find this
post helpful?



Zain Sadaqat

save it for later

Thank You!



Zain Sadaqat

save it for later