

Sunday, 3 September 2023

## UseEffect

---

useEffect is a hook in react that allows you to perform any kind of operation when some special event occurs.

For Example:

- You wanted to do something when your component loaded first time. Like fetching some initial data to show.
- You wanted to do something every time your component is reloaded.
- You wanted to do something every time a specific or set of states changes.

### Importing `useEffect`:

- To use `useEffect`, you need to import it from the 'react' library:

javascript

 Copy code

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

So to perform any kind of operation on these special types of event. We generally use useEffect hook. It is an higher order function that takes 2 arguments. First is function that it has to run and second is set of dependencies in array.

### Syntax:

- `useEffect` takes two arguments: a function and an optional array of dependencies.
- The function inside `useEffect` is the side effect itself, and it will run after the component renders.
- The dependencies array is an optional argument that specifies which values the effect depends on.

javascript

 Copy code

```
useEffect(() => {  
  // Side effect code here  
}, [dependencies]);
```

## Important Points

1. If you leave the dependency array empty. It indicates that the `useEffect` will call your function only when your component is loaded(mounted) first time.

```
javascript

useEffect(() => {
  // Fetch data from an API
  fetchData();
}, []); // Empty dependency array means this effect runs only once
```

2. If you don't provide dependency array then `useEffect` will call your function on every load.
3. If you specify the name of any states in the dependency array then `useEffect` will call your function every time that specific state will change.

```
javascript

useEffect(() => {
  // Code that depends on someValue
}, [someValue]);
```

## Example

Let us suppose we are creating a todo app. We will be using `useEffect` to fulfil following requirements.

1. We want to fetch all the todos of the current user whenever he/she comes to website first time. Means our component is loaded first time. Then we will specify `fetchTodo()` function in the `useEffect` with `[]` dependency so that it will be called only when component is loaded first time.
2. We wanted to show updated todos on the UI every time you change or add new todo. So for that mapping of todos with `TodoItem` component is specified in the

useEffect with dependency on todos. So anytime your todos state will change the mapping will happen again and you will see update UI.