



# hi CheatSheet of useEffect

React Hook #1 useState Hook : <https://youtu.be/zA0vgrt3Vsg>

React Hook #2 useEffect Hook : <https://youtu.be/wx5R0xRp05w>

## What is useEffect Hook?

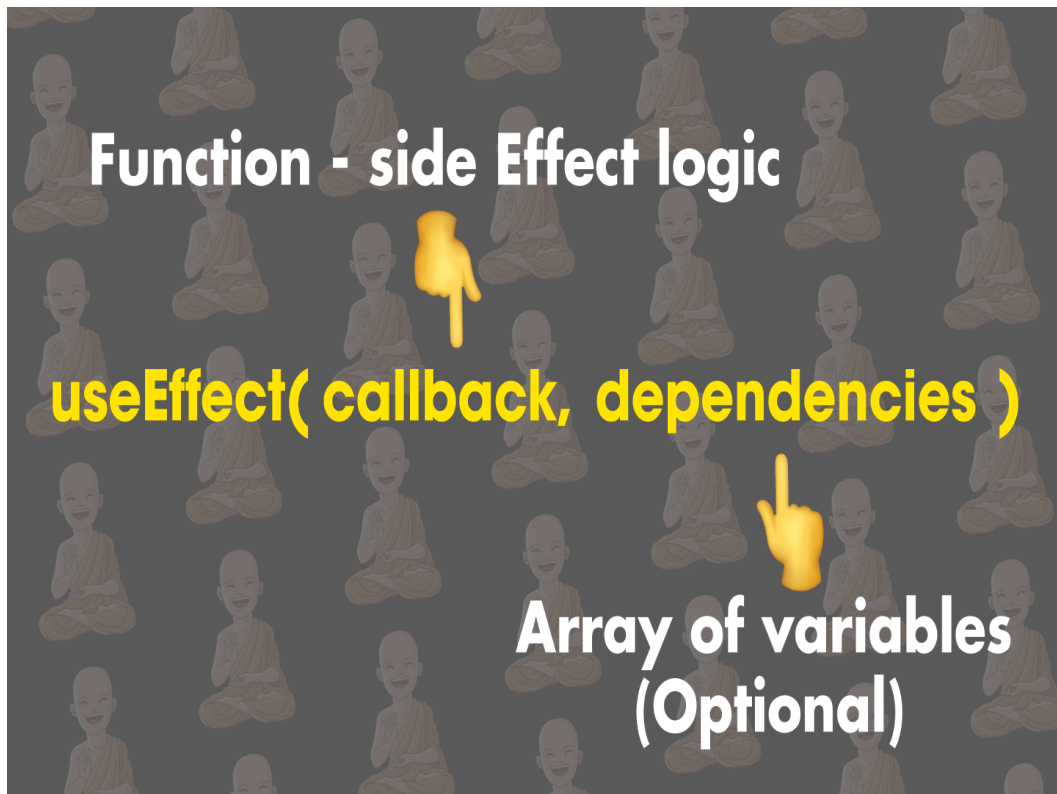
- useEffect Hook is one of the most Important Hook in all React Hook.
- It is run on render and anything changes in that component.
- It is a function to perform side-effects in Functional Components.

## What is Side Effect?

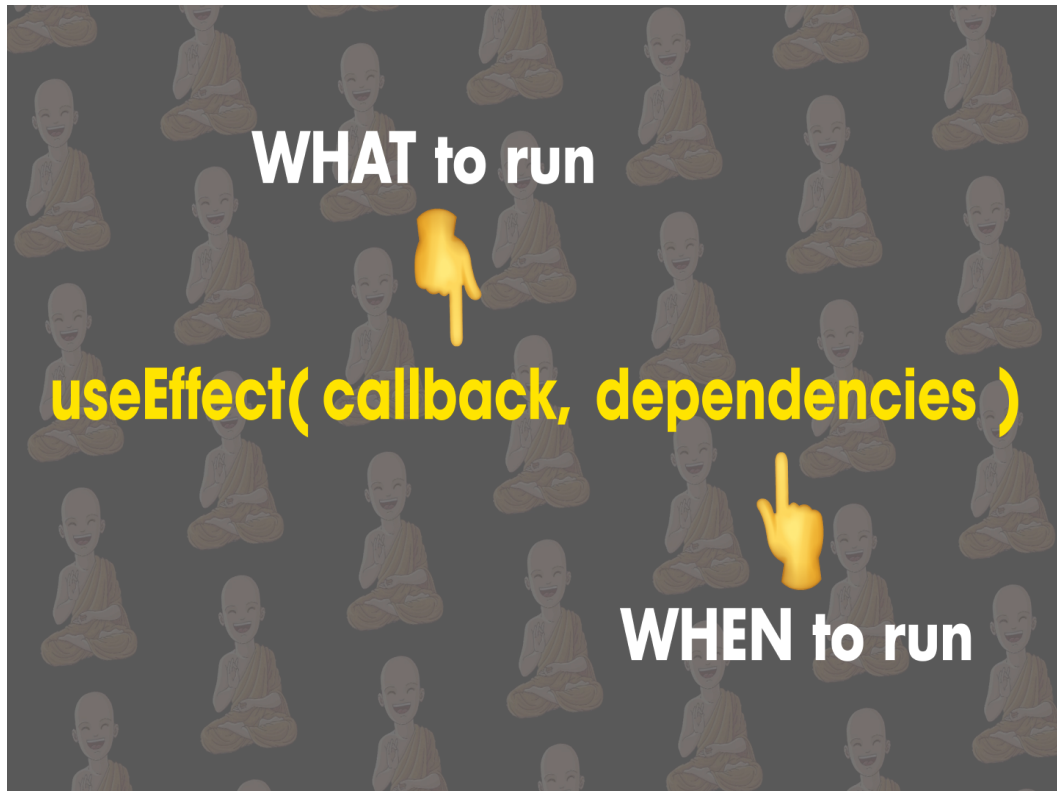
Side effects are actions which are performed with the "Outside world" Like -

- Fetching data from API or Backend
- directly updating the DOM - document or window object
- and timer functions - setTimeout or setInterval.

**useEffect has 2 Arguments :**



Shortcut for remember:



### 3 Variation of useEffect Hook

1. useEffect without any dependencies - it runs with first render and also run on any thing changes in that component.
2. useEffect with an empty array - it runs only on first render. Useful for Fetching data.
3. useEffect with variables - it runs on first render and then any variable that we pass in dependency change it will run.

### Clean-up Function in useEffect

- If we are setting state using `setInterval` and that side effect is not cleaned up, when our component unmounts or we're no longer using it, the state is destroyed with the component – but the `setInterval` function will keep running. and that's make our application slow and low in performance.
- To use clean-up function we need to run return function in useEffect. Timeouts, subscriptions, event listeners, and other effects that are no longer needed should be disposed with the help of cleanup function.

#### Example of Clean up Function

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

const App = () => {
  const [count, setCount] = useState(0);

  useEffect(() => {
    console.log("Run useEffect", count);

    return () => { // clean-up function
      console.log("Clean up", count);
    };
  }, [count]);

  return (
    <>
      <h3>Count {count}</h3>
      <button onClick={() => setCount(count + 1)}>Increase</button>
    </>
  );
}
```

```
    </>  
  );  
};  
  
export default App;
```

**Thanks for Reading 😊**

~ CODE BLESS YOU