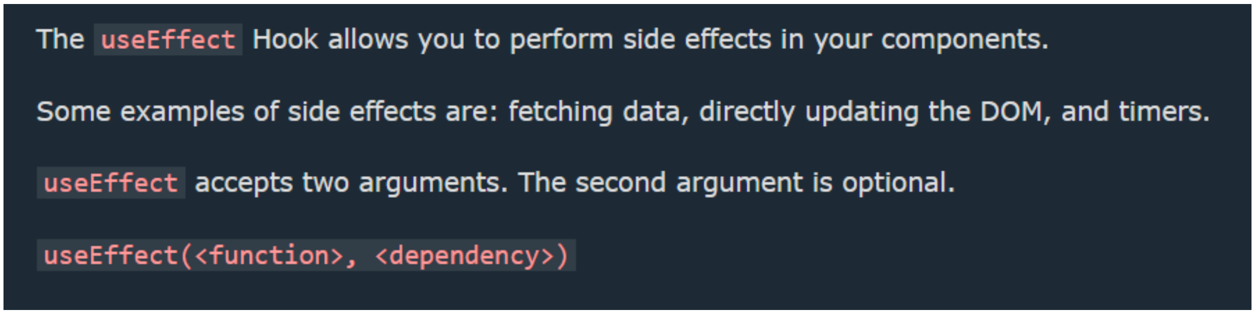
Use Effect Hook

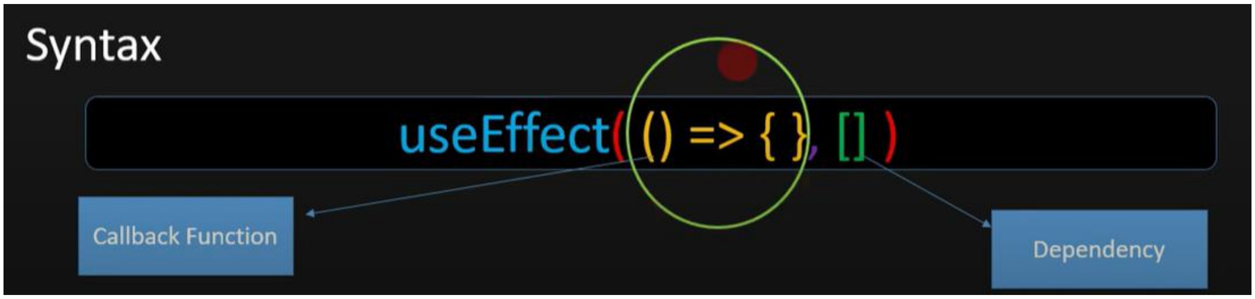


The `useEffect` Hook allows you to perform side effects in your components.

Some examples of side effects are: fetching data, directly updating the DOM, and timers.

`useEffect` accepts two arguments. The second argument is optional.

`useEffect(<function>, <dependency>)`



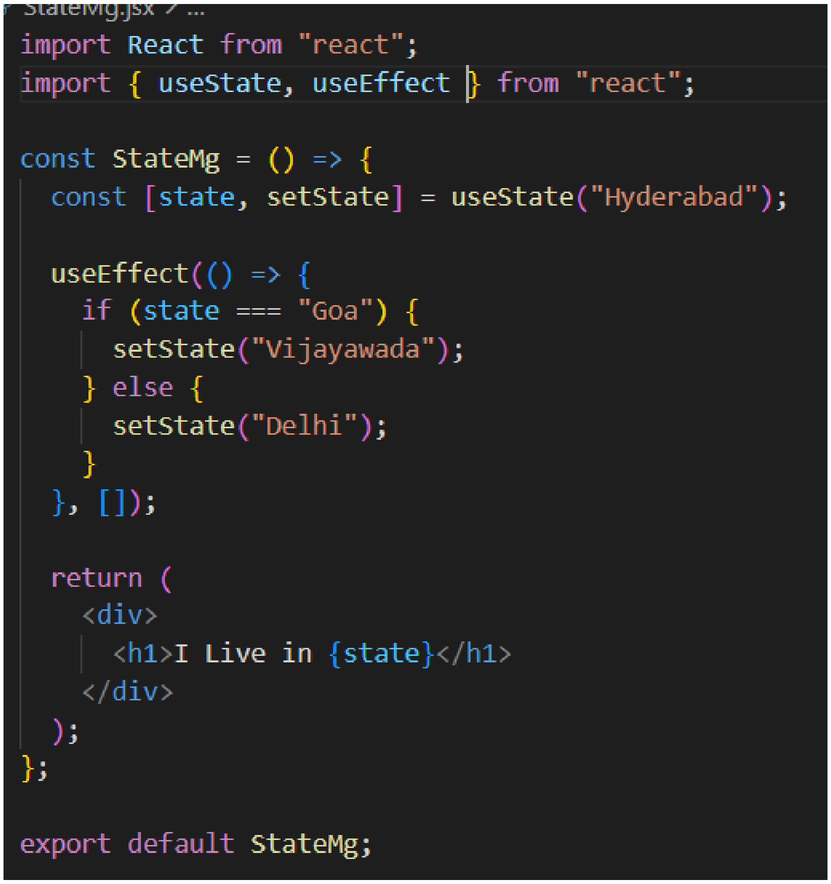
Syntax

```
useEffect( () => { }, [] )
```

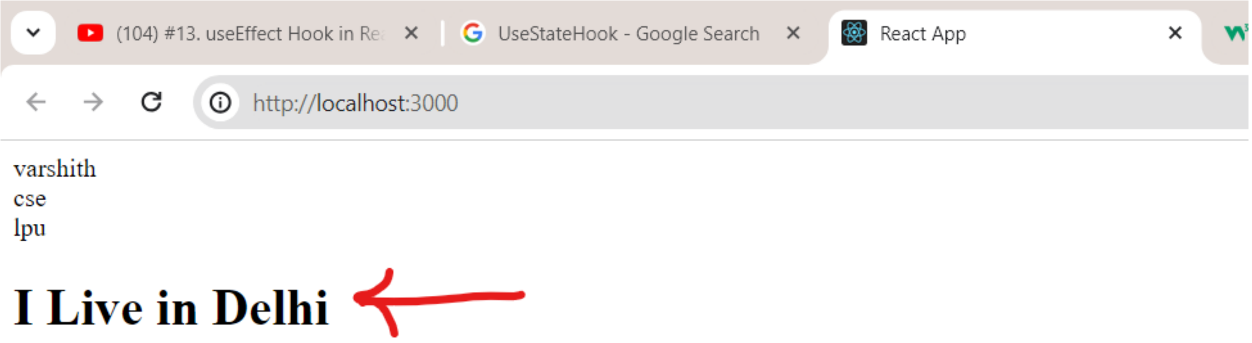
useEffect used to control loop which we faced difficulty in useState

**dependency [] → used to control the loop
Another advantage of Dependence is used to monitor the changes which varible passed in dependences it say thE changes to DOM(explained below)

EX [city]



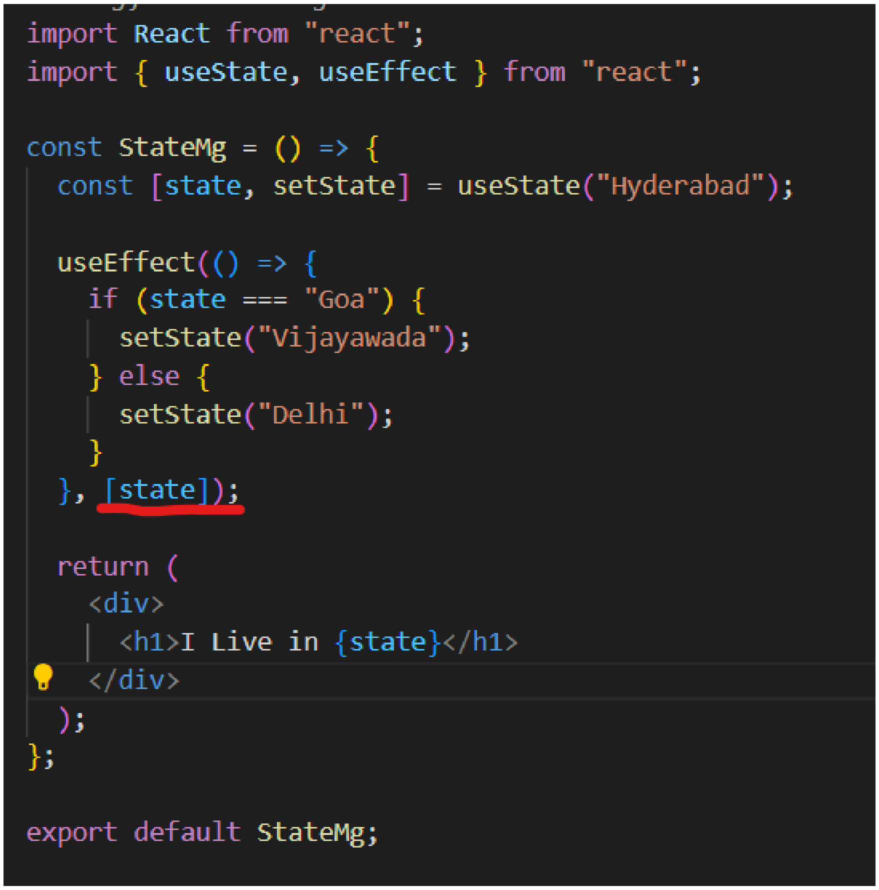
```
import React from "react";
import { useState, useEffect } from "react";

const StateMg = () => {
  const [state, setState] = useState("Hyderabad");

  useEffect(() => {
    if (state === "Goa") {
      setState("Vijayawada");
    } else {
      setState("Delhi");
    }
  }, []);

  return (
    <div>
      <h1>I Live in {state}</h1>
    </div>
  );
};

export default StateMg;
```



varshith
cse
lpu

I Live in Delhi

Explanation of 2nd advantage



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

const StateMg = () => {
  const [state, setState] = useState("Hyderabad");

  useEffect(() => {
    if (state === "Goa") {
      setState("Vijayawada");
    } else {
      setState("Delhi");
    }
  }, [state]);

  return (
    <div>
      <h1>I Live in {state}</h1>
    </div>
  );
};

export default StateMg;
```

To know any changes done on state