**UseEffect:**

* a hook in React that enables the execution of side effects in functional components.

**Syntax:**

useEffect(() => {

// Side effect code

}, [dependencies]);

[ ] => Mounting Phase runs during First render

[dependencies]) => Change during dependency change

Example:

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

const SimpleExample = () => {

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

// useEffect to run some code after each render

useEffect(() => {

console.log('Component rendered');

// Cleanup function (optional)

return () => {

console.log('Component will unmount');

};

});

return (

<div>

<h1>Simple Example</h1>

<p>Count: {count}</p>

<button onClick={() => setCount(count + 1)}>Increment Count</button>

</div>

);

};

export default SimpleExample;

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

const DependencyExample = () => {

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

// useEffect with a dependency to run code when count changes

useEffect(() => {

console.log('Count changed:', count);

// Cleanup function (optional)

return () => {

console.log('Cleanup: Component will unmount or count changed');

};

}, [count]); // Run the effect whenever count changes

return (

<div>

<h1>Dependency Example</h1>

<p>Count: {count}</p>

<button onClick={() => setCount(count + 1)}>Increment Count</button>

</div>

);

};

export default DependencyExample;

**Destructuring:**

Allows to extract values from arrays or properties from objects and bind them to variables.

Examples:

const numbers = [1, 2, 3];

const [a, b, c] = numbers;

console.log(a); // Output: 1

console.log(b); // Output: 2

console.log(c); // Output: 3

Ex:2

const colors = ['red', 'green'];

const [primaryColor, secondaryColor = 'blue'] = colors;

console.log(primaryColor); // Output: red

console.log(secondaryColor); // Output: green

EX:1

const person = { name: 'John', age: 30, city: 'New York' };

const { name, age, city } = person;

console.log(name); // Output: John

console.log(age); // Output: 30

console.log(city); // Output: New York

Ex:2

const person = {

name: 'Alice',

age: 25,

address: {

city: 'Wonderland',

country: 'Fictionland'

}

};

const { name, address: { city, country } } = person;

console.log(name); // Output: Alice

console.log(city); // Output: Wonderland

console.log(country);// Output: Fictionland