**Lists and Hooks**

1. useState
2. useEffect
3. useRef
4. UseState

* useState is React Hook that allows you to add state to a functional component. It returns an array with two values: the current state and a function to update it. The Hook takes an initial state value as an argument and returns an updated state value whenever the setter function is called.

Example:-

import {React,useState} from 'react'

import { Button } from 'reactstrap'

import 'bootstrap/dist/css/bootstrap.min.css';

function UseStatedec() {

const[count, setCount]=useState(10)

const[plusecount,setplusecount]=useState(0)

function UpdateCount(){

setCount(count - 1)

// console.log(count);

}

function DeleteCount()

{

setplusecount(plusecount + 1)

}

return (

<div>

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

<Button onClick={UpdateCount} color="danger" outline>

Decrease Count

</Button>

<p>Text: {plusecount}</p>

<Button onClick={DeleteCount} color="success" outline>

Increse Count

</Button>

</div>

);

}

export default UseStatedec;

1. UseEffect

Example:-

* 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>)

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

import { Button } from "reactstrap";

export default function UseEffectCom() {

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

let [count2, setCount2] = useState(0);

// call everytime

useEffect(() => {

console.log("------useEffect without Depe--------");

});

// call only first time not call on update of component

useEffect(() => {

console.log("------useEffect with [] array--------");

}, []);

// its call when counts value change

useEffect(() => {

console.log("------useEffect with depend on count--------");

// setCount(++count);

}, [count]);

// its call when count or count2 value change

useEffect(() => {

console.log("------useEffect with depend on count and count 2--------");

}, [count, count2]);

return (

<>

<h1>UseEffectCom</h1>

<h1>count is {count}</h1>

<h1>count 2 is {count2}</h1>

<Button color="danger" outline onClick={() => setCount(++count)}>

increment

</Button>

<Button color="danger" outline onClick={() => setCount2(++count2)}>

increment

</Button>

</>

);

}

3.UseRef

* UseRef is a React Hook that lets you reference a value that’s not needed for rendering.

Example:-

import React, { useRef } from "react";

import { Button } from "reactstrap";

export default function UseRefCom() {

let headRef = useRef(null);

let inputRef= useRef(null);

function fun() {

inputRef.current.focus();

headRef.current.textContent = "test"

console.log("------>");

}

return (

<div>

<h1 ref={headRef}>input text</h1>

<input ref={inputRef} type="text" />

<Button color="danger" outline onClick={() => fun()}>

Click Me

</Button>

</div>

);

}