**Hooks in react js**

Hooks were added to **React in version 16.8**.

They allow **function components** to use **state** and other React features, which were previously only available in class components. Because of this, **class components are now rarely needed**.

### **3 Rules of Hooks**

1. **Only call Hooks inside React function components**
   * Don't call Hooks inside regular JavaScript functions.
2. **Only call Hooks at the top level**
   * Don't call Hooks inside loops, conditions, or nested functions.
3. **Use Hooks in the same order every time a component renders**
   * This helps React keep track of the Hook states properly.

### **Why Use Hooks?**

* Makes code **cleaner and reusable**
* Eliminates the need for **class components**
* Encourages **functional programming** in React

### **Importing Hooks**

You must import Hooks from React before using them.

To use useState, import it like this:

import { useState } from "react";

**useState**

* useState is a **React Hook** that lets **function components** store and manage state.
* Before Hooks, only **class components** could have state. Now, with useState, you can have state **inside functional components**, making your code cleaner and more modern.

**Syntax:**

const [currentState, setStateFunction] = useState(initialState);

* currentState: The current value of the state.
* setStateFunction: A function to update the state.
* initialState: The initial value you want for that state.

### **Example:**

**Write a program to build React app having a button which increase count by 1 while clicking it.**

**US1.js**

import React, { useState } from 'react';

function US1() {

// Declare a new state variable, which we'll call "count"

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

function handleCount() { setCount(count+1) }

return (

<div>

<p>You clicked {count} times</p>

<button onClick={handleCount}>

Click me

</button>

</div>

);

}

export default US1

**Or inline function calling**

import { useState } from "react";

function US1 () {

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

return (

<div>

<p>You clicked {count} times</p>

<button onClick=**{() => setCount(count + 1)**}>Click me</button>

</div>

);

}export default US1

**Example:**

**Create a program to build React app having buttons to increment and decrement the number by clicking that respective button. Also, increment of the number should be performed only if number is less than 10 and decrement of the number should be performed if number is greater than 0.**

**US.js**

import { useState } from "react";

function US () {

const [num,setnum]=useState(0)

function increment(){

if(num<10){

setnum(num+1);

}else{

return false;

}

}

function decrement(){

if(num > 0){

setnum(num-1);

}else{

return false;

}

}

return (

<div>

<button onClick={increment}>Increment</button>

<button onClick={decrement}>Decrement</button>

<h1> {num} </h1>

</div>

)

}

export default US

**Example:**

**Write a program to build React app to perform the tasks as asked below.**

* **Add three buttons “Change Text”, “Change Color”, “Hide/Show”.**
* **Add heading “LJ University” in red color(initial) and also add “React Js Hooks” text in h2 tag.**
* **By clicking on “Change text” button text should be changed to “Welcome students” and vice versa.**
* **By clicking on “Change Color” button change color of text to “blue” and vice versa. This color change should be performed while double clicking on the button.**
* **Initially button text should be “Hide”. While clicking on it the button text should be changed to “Show” and text “React Js Hooks” will not be shown.**

**US1.js**

import {useState} from "react";

function US1(){

//useState to Change text

const [name,**setName**] = useState("LJ University");

//useState to Change Color

const [textColor,**setcolor**] = useState("Red");

//useState to Show Hide text

const [hideText,**setHide**]=useState("React Js Hooks");

const[buttontext,**setButtontext**]= useState("Hide")

// Function to show and hide text and also button text

function showhide() {

if(buttontext==="Hide")

{

**setButtontext**("Show");

**setHide**("")

}

else{

**setButtontext**("Hide");

**setHide**("React Js Hooks")

}

};

// function to change text value

function changeName(){

if(name === "LJ University"){

**setName**("Welcome Students")

}else{

**setName**("LJ University")

}

}

// Function to change color of the text

function changeColor(){

if(textColor === 'red'){

**setcolor**("blue")

}else{

**setcolor**("red")

}

}

return(

<div>

<button onClick={changeName}>Change Text</button>

<button onDoubleClick={changeColor}>Change Color</button>

<button onClick = {showhide}> {**buttontext**}</button>

<h1 style={{color:**textColor**}}>{**name**}</h1>

<h2>{**hideText**}</h2>

</div>

)}

export default US1

**Example:**

**Write a program to build React app having a button which changes image by clicking it.**

**US2.js**

import { useState } from 'react';

import img1 from "./img1.png";

import img2 from "./img2.png";

function US2 () {

const [myImage,setImage]=useState(img1);

function changeImage () {

if(myImage === img1){

setImage(img2)

}else{

setImage(img1)

}

}

return (

<div>

<img src={myImage} heigth="200px" width="200px" alt="logo" />

<button onClick={changeImage}>Change Image</button>

</div>

) }

export default US2

**Example:**

**Write React component having a button and image. By clicking on button, image changes randomly from a given array of images.**

**US3.js**

import {useState} from "react";

import img1 from "./img1.jpg"

import img2 from "./img2.jpg"

import img3 from "./img3.png"

import img4 from "./img4.jpg"

import img5 from "./img5.jpg"

function US3()

{

const **arr** = [img1,img2,img3,img4,img5]

const [**myimage**,**setimage**] = useState(**arr[0]**);

function **changeImage** {

const **randomIndex** = Math.floor(Math.random() \* **arr**.length);

**setimage**(**arr**[**randomIndex**]);

};

return (

<div className="App">

<header className="App-header">

<h1>Random Image Generator</h1>

<img src={**myimage**} alt="Random" width="500" height="500"/>

<button onClick={**changeImage**}>Change Image</button>

</header>

</div>

);

}

export default US3

**Example:**

**Create a React component that manages multiple form input fields using a single state object and displays the values in real-time**

**US4.js**

import { useState } from 'react'

function US4() {

const[**data**,**setdata**]=useState(**{}**);

function **handleChange**(e) {

**const { name, value } = e.target;**

**setdata**({...**data**,[name]: **value**});

};

return (

<div>

<div><input type="text" name="**firstName**" onChange={**handleChange**} placeholder='First Name'/></div>

<div><input type="text" name="**lastName**" onChange={**handleChange**} placeholder='Last Name'/></div>

<h1>First Name: {**data**.**firstName**} Lastname: {**data**.**lastName**}</h1>

</div>

) }

export default US4

**Example:**

**Write a react component for todo list.**

* **Add 1 input field and button and by clicking on button display entered task on the same page.**
* **Also, add delete button with each added task to delete the task.**

**Todo.js**

import {useState} from 'react'

function Todo() {

const[Task, **setTask**]= useState("");

const[**Todolist**, **setTodoList**]=useState([]);

function **handleChange**(event) {

**setTask**(event.target.value);

}

**// To add task**

function **addTask**(event) {

**setTodoList**([...**Todolist**,Task]);

};

**//To add delete functionality**

**function deleteTask(taskName){**

**setTodoList(**

**Todolist.filter((task)=>{**

**if (task!==taskName){**

**return true;**

**} else{**

**return false;**

**}**

**})**

**)**

**}**

return (

<div>

<h1> Enter Task </h1>

<input onChange={**handleChange**}/>

<button onClick={**addTask**}> Add Task </button>

{**Todolist**.map((task)=>{

return(

<div>

<h1> {task}</h1>

<button onClick={() => deleteTask(task)}>Delete</button>

</div>

);

})}

</div>

);

}

export default Todo

**Create react app which takes user defined inputs number 1 and number 2 and perform addition, subtraction, multiplication, division of the numbers. (Use useState hook)**

US5.js

import { **useState** } from 'react'

function US5() {

const[**data**,**setdata**]=**useState**({});

const[**result**,**setresult**]=**useState**();

const **handleChange** = (e) => {

const { name, value } = e.target;

**setdata**({...**data**,[name]: value});

};

function addition(){

**setresult**(parseInt(data.num1) + parseInt(data.num2))

}

function sub(){

setresult(parseInt(data.num1) - parseInt(data.num2))

}

function mult(){

setresult(parseInt(data.num1) \* parseInt(data.num2))

}

function division(){

setresult(parseInt(data.num1) / parseInt(data.num2))

}

return (

<div>

<div><input type="number" name="num1" onChange={**handleChange**} placeholder='First Name'/></div>

<div><input type="number" name="num2" onChange={**handleChange**} placeholder='Last Name'/></div>

<button onClick={addition}>addition</button>

<button onClick={sub}>Subtraction</button>

<button onClick={mult}>Multiplication</button>

<button onClick={division}>Division</button>

<h1> {**result**}</h1>

</div>

) }

export default US5