**Use multiple contexts in a React application by creating and consuming them across different components.**

**uc1.js: Creates a context for CSS styling and provides it to Comp1.**

**uc2.js: Creates a context for a string value ("Students") and provides it to Comp2.**

**uc3.js: Consumes both contexts and displays a message with the provided styles and string.**

**uc1.js**

import React, { createContext } from "react"

import Comp1 from "./uc2"

const CC = createContext();

const mycss={backgroundColor:'yellow',color:'red',fontSize:"45px"}

function Comp(){

return (

<>

<CC.Provider value={mycss}>

<Comp1/>

</CC.Provider>

</>

)

}

export default Comp

export {CC}

**uc2.js**

import { createContext } from "react"

import Comp2 from "./uc3"

const CC1 = createContext();

function Comp(){

return (

<CC1.Provider value="Students">

<Comp2/>

</CC1.Provider>

)

}

export default Comp

export {CC1}

**uc3.js**

import React, { useContext } from "react"

import { CC } from "./uc1"

import { CC1 } from "./uc2"

function Comp3(){

const mycss = useContext(CC)

const data = useContext(CC1)

return (

<h1 style={mycss}>Welcome to useContext tutorial {data}</h1>

)

}

export default Comp3

**Create react app which to perform following task using function component:**

**• Create one main file name F1.js & other 2 component files F2.js & F3.js.**

**• Main file contains form with following fields:**

* **First Name (Input type text)**
* **Last Name (Input type text)**
* **Message (Textarea)**
* **City (Dropdown)**
* **Gender (Radio Button)**

**• Pass values of all fields from F1.js file to F3.js file. And display all submitted values in alert box using useContext & useState hook.**

**No need to write App.js file.**

**F1.js**

import { createContext, useState } from 'react';

import F2 from './F2';

const AppContext = createContext();

const AppProvider = () => {

**const [user, setUser] = useState({});**

**const [formData, setFormData] = useState({});**

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

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

**setFormData({ ...formData, [name]: value });**

**};**

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

**e.preventDefault();**

**setUser(formData);**

**};**

return (

<AppContext.Provider value={user}>

<div>

<h1>React useState and useContext Example</h1>

<form onSubmit={handleSubmit}>

<label>

First Name:

<input type="text" name="firstname" onChange={handleChange} />

</label>

<br />

<label>

Last Name:

<input type="text" name="lastname" onChange={handleChange} />

</label>

<br />

<label>

Message:

<textarea name="message" onChange={handleChange} />

</label>

<br />

<label>

City:

<select name="city" onChange={handleChange}>

<option value="Ahmedabad">Ahmedabad</option>

<option value="Rajkot">Rajkot</option>

<option value="Gandhinagar">Gandhinagar</option>

</select>

</label>

<br />

<label>

Gender:

<input type="radio" name="gender" value="Male" onChange={handleChange} /> Male

<input type="radio" name="gender" value="Female" onChange={handleChange} /> Female

</label>

<input type='submit'></input>

</form>

**<F2 />**

</div>

</AppContext.Provider>

);};

export default AppProvider

export { AppContext }

**F2.js**

import F3 from './F3';

const F2 = () => {

return (<F3 />)

};

export default F2;

**F3.js**

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

import { **AppContext** } from './F1';

const UserProfile = () => {

const user = useContext(**AppContext**);

**useEffect(() => {**

**const isEmpty = (obj) => Object.keys(obj).length === 0;**

**if (!isEmpty(user)) {**

**alert(JSON.stringify(user));**

**}**

**}, [user]);**

return (

<div>

<h2>User Profile</h2>

<p>Name: {user.firstname} {user.lastname}</p>

<p>Message: {user.message}</p>

</div>

);

};

export default UserProfile;

**Create react app to filter images based on category while clicking on respective buttons.**

**In example,**

**Categories – All, Samsung, Vivo and Oneplus.**

**By clicking on “All” it will display mobiles of all brands. By clicking on specific brand it will display mobiles of respective brand.**

import React, { 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"

const Gallery = [

{ id:1,pic:img1,category:"Samsung"},

{ id:2,pic:img2,category:"Mi"},

{ id:3,pic:img3,category:"Oneplus"},

{ id:4,pic:img4,category:"Mi"},

{ id:5,pic:img5,category:"Oneplus"},

];

function Product () {

const[images,setImage]=useState(Gallery);

function handleproduct(Item){

const finaldata=Gallery.filter((value)=>value.category===Item)

if(Item !== "All"){ setImage(finaldata); }

else{ setImage(Gallery) }

}

return (

<div>

<button onClick={() =>handleproduct('All')}>All</button>

<button onClick={() =>handleproduct('Samsung')}>Samsung</button>

<button onClick={() =>handleproduct('Mi')}>Mi</button>

<button onClick={() =>handleproduct('Oneplus')}>Oneplus</button>

<div>

{

images.map((val)=>

{

return(

<>

<img src={val.pic} height="300" width="300"/>

</>

)

})

}

</div>

</div>

)

}

export default Product

**1**

**Create react app to craete exam form. Fields are as under.  
Name -text,  
Email -email  
Password -password,   
Gender-radio,   
Exam date-datepicker,  
Exam center-dropdown)  
Use useState hook to save the state of the form. Also, add validation for the email and password fields. Display submited values.**

**2**

**Create a React app with three components: parent component P.js and two child components C1.js, C2.js.  
 Use useContext hook to pass two colors from the Parent component P.js to the Child component C2.js. (Yellow color as font color and Blue color as background color)  
In Child component C1.js import child component C2.js .  
In C2.js display “Full stack Development” text in h1 heading with above mentioned background color and font color using context.**

**3**

**Use multiple contexts in a React application by creating and consuming them across different components.**

**uc1.js: Creates a context for CSS styling and provides it to Comp1.**

**uc2.js: Creates a context for a string value ("Students") and provides it to Comp2.**

**uc3.js: Consumes both contexts and displays a message with the provided styles and string.**