React JS-2

Miscellaneous Tasks

Task-1

Create a React app with a student registration form that includes fields for User Name, Age, Gender (radio), and City (dropdown), and validates that all fields are filled and Age is less than or equal to 60 before displaying the submitted data in an alert box.

```
import React, { useState } from 'react';
function StudentForm() {
  const [formData, setFormData] = useState({});

  function handleChange (e){
    const name = e.target.name;
    const value = e.target.value;
   setFormData({ ...formData , [name] :value});
  };

  function handleSubmit (e) {
    e.preventDefault();

    if (formData.name === undefined ||formData.age === undefined || formData.gender ===
undefined || formData.city===undefined) {
      alert('Enter all details');
    }

    else if ( parseInt(formData.age) >= 60) {
      alert('Age must be 60 or below!');
    }
else{
    alert(`Submitted Data:
Name: ${formData.name}
Age: ${formData.age}
Gender: ${formData.gender}
city: ${formData.city}`);
  };}

  return (
    <div>
      <h2>Registration</h2>
      <form onSubmit={handleSubmit}>
        <div>
```

```
        <label>Name:</label><br />
        <input type="text" name="name" onChange={handleChange} />
      </div><br />

      <div>
        <label>Age:</label>
        <input type="number" name="age" onChange={handleChange} />
      </div>

      <div>
        <label>Gender:</label>
        <input type="radio" name="gender" value="Male" onChange={handleChange} /> Male
        <input type="radio" name="gender" value="Female" onChange={handleChange} />
Female
      </div>

      <div>
        <label>city:</label>
        <select name="city" onChange={handleChange}>
          <option value="Select" default selected hidden >--Select--</option>
          <option value="Ahmedabad" >Ahmedabad</option>
          <option value="Surat">Surat</option>
          <option value="Rajkot">Rajkot</option>
        </select>
      </div>

      <button type="submit">Submit</button>
    </form>
  </div>
 );
}

export default StudentForm;
```

Task -2 Implement a character counter that displays the number of characters entered in a textarea input.

```
import { useState } from 'react';
function CharCounter() {
const [text, setText] = useState("");
const handleChange = (e) => {
setText(e.target.value);
};
return (
<div>
```

```
<h2>Character Counter</h2>
<textarea value={text} onChange={handleChange} />
<p>Characters: {text.length}</p>
</div>
);
}
export default CharCounter;
```

Task: 3 (Both result has to display)**
Create react app which takes user defined inputs number 1 and number 2 and perform addition and subtraction of the numbers. (Use useState hook)

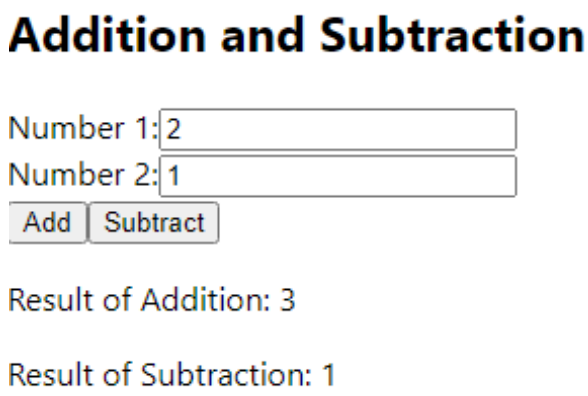
Calc.js

```
import React, { useState } from 'react';
function Calc() {
  const [num1, setNum1] = useState(0);
  const [num2, setNum2] = useState(0);
  const [addResult, setAddResult] = useState(0);
  const [subResult, setSubResult] = useState(0);

  const handleAddition = () => {
    const sum = num1 + num2;
    setAddResult(sum);
  };
  const handleSubtraction = () => {
    const difference = num1 - num2;
    setSubResult(difference);
  };
  return (
    <div>
      <h2>Addition and Subtraction</h2>
      <div>
        <label>
          Number 1:
          <input type="number" value={num1} onChange={(e) =>
setNum1(parseInt(e.target.value))} />
        </label>
      </div>
      <div>
        <label>
          Number 2:
          <input type="number" value={num2} onChange={(e) =>
setNum2(parseInt(e.target.value))} />
        </label>
      </div>
      <div>
        <button onClick={handleAddition}>Add</button>
```

```
      <button onClick={handleSubtraction}>Subtract</button>

    </div>
    <div>
      <p>Result of Addition: {addResult}</p>
      <p>Result of Subtraction: {subResult}</p>
    </div>
   </div>
  );
};
export default Calc;
```



Task-4 Todo list (Add and delete the task using index) (Reference)

```
import { useState } from 'react';
function Todo() {
const [task, setTask] = useState('');
const [todoList, setTodoList] = useState([]);
const addTask = () => {
setTodoList([...todoList, task]);
setTask('');
};
const deleteTask = (id) => {
setTodoList(todoList.filter((task, index) => index !== id));
};
return (
<div>
<input value={task} onChange={(e) => setTask(e.target.value)}/>
<button onClick={addTask}>Add</button>
<ul>
{todoList.map((item, index) => (
 <li key={index}>
 {item}
 <button onClick={() => deleteTask(index)}>Delete</button>
```

```
 </li>
))}
</ul>
</div>
);
}
export default Todo;
```

Task -5 Create react app which takes user defined inputs number 1 and number 2 and perform addition, subtraction, multiplication, division of the numbers. **(Use useReducer hook)**

```
import React, { useReducer } from "react";

// Reducer function
function reducer(state, action) {
  const num1 = Number(state.num1);
  const num2 = Number(state.num2);

  switch (action.type) {
    case "SET_NUM1":
      return { ...state, num1: action.value };
    case "SET_NUM2":
      return { ...state, num2: action.value };
    case "ADD":
      return { ...state, result: num1 + num2 };
    case "SUB":
      return { ...state, result: num1 - num2 };
    case "MUL":
      return { ...state, result: num1 * num2 };
    case "DIV":
      return {
        ...state,
        result: num2 !== 0 ? num1 / num2 : "Cannot divide by zero",
      };
    default:
      return state;
  }
}

function Calculator() {
  const [state, dispatch] = useReducer(reducer, { num1: "", num2: "",result: ""});

  return (
    <div style={{ padding: "20px" }}>
      <h2>Simple Calculator (useReducer)</h2>
```

```
    <input       type="number"  placeholder="Enter number 1"      value={state.num1}
      onChange={(e) => dispatch({ type: "SET_NUM1", value: e.target.value })}    />

    <input  type="number"  placeholder="Enter number 2"    value={state.num2}
      onChange={(e) => dispatch({ type: "SET_NUM2", value: e.target.value })}   />

    <div style={{ marginTop: "10px" }}>
      <button onClick={() => dispatch({ type: "ADD" })}>Add</button>
      <button onClick={() => dispatch({ type: "SUB" })}>Subtract</button>
      <button onClick={() => dispatch({ type: "MUL" })}>Multiply</button>
      <button onClick={() => dispatch({ type: "DIV" })}>Divide</button>
    </div>

    <h3>Result: {state.result}</h3>
   </div>
  );
}
export default Calculator;
```

API – Random dog

```
import React,{ useState,useEffect } from 'react'
import axios from "axios";
const Randomimage = () =>
{
   const[myimg,setimg]=useState("");

   useEffect(() => {
      setInterval(() => {
         axios
         .get('https://dog.ceo/api/breeds/image/random')
         .then((response)=>{console.log(response.data);setimg(response.data);})
         .catch((error)=>{console.error(error);})
      }, 10000)
   },[])

   return(
      <div>
         <img src={myimg.message} alt='image' height={300} width={300}/>
      </div>
   )
}
export default Randomimage
```

API Random user

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

const baseURL = "https://randomuser.me/api";

const App = () => {
  const [post, setPost] = useState(null);

  const fetchUser = () => {
    axios.get(baseURL)
      .then((res) => {
        setPost(res.data.results[0]);
      })
      .catch((err) => {
        console.error("Error fetching user:", err);
      });
  };

  useEffect(() => {
    fetchUser(); // Fetch once on initial render
  }, []);

  if (!post) return <p>Loading...</p>;

  const { name, email, picture } = post;

  return (
    <div>
      <h1>Random User</h1>
      <div>
        <img src={picture.large} alt="User" />
        <p>Name: {name.first} {name.last}</p>
        <p>Email: {email}</p>
        <button onClick={fetchUser}>Load New User</button>
      </div>
    </div>
  );
};

export default App;
```

Advance Level Miscellaneous Tasks [Ref*]

Task 1 : Create react app to filter images based on category while clicking on respective buttons. In example,
Categories – All, Samsung, Mi 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
```

Task 2:

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:

o First Name (Input type text)
o Last Name (Input type text)
• 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.

[Also add Message (Textarea), City (Dropdown) , Gender (Radio Button) if required]

```
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>
            <form onSubmit={handleSubmit}>
              <input type="text" name="firstname" onChange={handleChange} />
              <input type="text" name="lastname" onChange={handleChange} />
              <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(() => {
          if (Object.keys(user).length > 0) {
          alert(JSON.stringify(user));
      }
   }, [user]);

//Object.keys(user) is a JavaScript method that returns an array of the object's own
enumerable property names (i.e., its keys).

   return (
      <div>
         <h2>User Profile</h2>
         <p>Name: {user.firstname} {user.lastname}</p>

      </div>
   );
};
export default UserProfile;
```

Task 3: Create a React application that allows users to toggle between Light and Dark themes. Use useState to manage the theme state and useContext to share this state between components. Define the context inside the same file. When the user clicks a button, the theme should toggle, and the background and text colors should change accordingly. Display the current mode as "Light Mode" or "Dark Mode" in the center of the screen.

Main.js (Defines Context, provides it)

```
import React, { useState, createContext } from "react";
import Ur1 from "./Ur1";
// Define context inside Main.js
export const ThemeContext = createContext();

function Main() {
  const [darkMode, setDarkMode] = useState(false);
  const toggleTheme = () => setDarkMode((prev) => !prev);
  const value = {darkMode, toggleTheme};
  return (
    <ThemeContext.Provider value={value}>
      <Ur1 />
    </ThemeContext.Provider>
  )}
export default Main;
```

✅ Ur1.js (Consumes Context)

```
import React, { useContext } from "react";
import { ThemeContext } from "./Main"; // Import context from Main.js

function Ur1() {
  const { darkMode, toggleTheme } = useContext(ThemeContext);
  const style = {
    backgroundColor: darkMode ? "#222" : "#eee",
    color: darkMode ? "#fff" : "#000",
    height: "100vh",    display: "flex",    justifyContent: "center",    alignItems: "center",
flexDirection: "column"
  };
  return (
    <div style={style}>
      <h2>{darkMode ? "Dark Mode" : "Light Mode"}</h2>
      <button onClick={toggleTheme}>Toggle Theme</button>
    </div> );}
export default Ur1;
```