import "./ExpenseForm.css"

import React, {useState} from 'react';

const ExpenseForm = (props) => {

    //ExpenseForm

    const [inputTitle, setInputTitle] = useState('')

    const [inputAmount, setInputAmount] = useState('')

    const [inputDate, setInputDate] = useState('')

    const titleChangeHandler = (event) => {

        setInputTitle(event.target.value)

    }

    const amountChangeHandler = (event) => {

        setInputAmount(event.target.value)

    }

    const dateChangeHandler = (event) => {

        //setInputDate(new Date(event.target.value))

        setInputDate(event.target.value)

    }

    const submitHandler = (event) => {

        event.preventDefault();

        const expenseData = {

            title:inputTitle,

            amount:inputAmount,

            date:new Date(inputDate)

        }

        console.log(expenseData)

        //props.onSaveExpenseData(expenseData);

        setInputAmount('')

        setInputDate('')

        setInputTitle('')

    }

    return(

      <form onSubmit={submitHandler}>

          <div className="new-expense\_\_controls">

             <div className="new-expense\_\_control">

                 <label>Title</label>

                 <input value={inputTitle} onChange={titleChangeHandler}/>

             </div>

             <div className="new-expense\_\_control">

                 <label>Amount</label>

                 <input type="number" value = {inputAmount} onChange={amountChangeHandler}/>

             </div>

             <div className="new-expense\_\_control">

                 <label>Date</label>

                 <input type="date" value={inputDate} min="2019-01-01" max="2022-12-31" onChange={dateChangeHandler} />

             </div>

          </div>

          <div className="new-expense\_\_actions">

              {/\* <button type="button" onClick={props.onCancel} >Cancel</button> \*/}

              <button type="submit">Add Expense</button>

          </div>

      </form>

    );

}

export default ExpenseForm;

import ExpenseForm from "./ExpenseForm";

import "./NewExpense.css"

import React, {useState} from 'react';

const NewExpense = (props) => {

    /\*const [showForm, setShowForm] = useState(false)

    const showFormHandler = () => {

        setShowForm(true)

    }

    const stopShowForm = () => {

        setShowForm(false)

    }\*/

    const saveExpenseDataHandler = (inputExpenseData) => {

        const expenseData = {

            ...inputExpenseData,

            id:Math.random().toString()

        }

        //console.log("In NewExpense ",expenseData)

        props.onAddExpense(expenseData)

    }

    return(

        <div className="new-expense">

           {/\* {!showForm && <button onClick={showFormHandler}>Add New Expense </button>}

            {showForm && <ExpenseForm onCancel={stopShowForm} onSaveExpenseData={saveExpenseDataHandler}/>}\*/}

<ExpenseForm onSaveExpenseData={saveExpenseDataHandler}/>

        </div>

    );

}

export default  NewExpense;

NewExpense.css

.new-expense {

    background-color: #a892ee;

    padding: 1rem;

    margin: 2rem auto;

    width: 50rem;

    max-width: 95%;

    border-radius: 12px;

    text-align: center;

    box-shadow: 0 1px 8px rgba(0, 0, 0, 0.25);

  }

  .new-expense button {

    font: inherit;

    cursor: pointer;

    padding: 1rem 2rem;

    border: 1px solid #40005d;

    background-color: #40005d;

    color: white;

    border-radius: 12px;

    margin-right: 1rem;

  }

  .new-expense button:hover,

  .new-expense button:active {

    background-color: #510674;

    border-color: #510674;

  }

  .new-expense button.alternative {

    color: #220131;

    border-color: transparent;

    background-color: transparent;

  }

  .new-expense button.alternative:hover,

  .new-expense button.alternative:active {

    background-color: #ddb3f8;

  }

**ExpenseForm.css**

.new-expense\_\_controls {

    display: flex;

    flex-wrap: wrap;

    gap: 1rem;

    margin-bottom: 1rem;

    text-align: left;

  }

  .new-expense\_\_control label {

    font-weight: bold;

    margin-bottom: 0.5rem;

    display: block;

  }

  .new-expense\_\_control input {

    font: inherit;

    padding: 0.5rem;

    border-radius: 6px;

    border: 1px solid #ccc;

    width: 20rem;

    max-width: 100%;

  }

  .new-expense\_\_actions {

    text-align: right;

  }

**App.js**

import './App.css';

import MessageComponent from './Person/MessageComponent';

import Person from './Person/Person';

import ExpenseItem from './components/ExpenseItem';

import NewExpense from './components/NewExpense/NewExpense';

import UseStateComponent from './components/UseStateComponent';

function App() {

  const expenses = [

    {title:"Shopping", amount:2000, date:new Date(2022, 1, 2)},

    {title:"Groceries", amount:7000, date:new Date(2022, 5, 6)},

    {title:"Movies", amount:4000, date:new Date(2022, 1, 5)}

  ]

  const addExpenseHandler = expense => {

    console.log("In App component ", expense)

  }

  return (

    <div>

      <NewExpense onAddExpense={addExpenseHandler} />

      //rest of your code

    </div>

  );

}

export default App;

**Working with Lists**

const ListComponent = () => {

    let arr = [1,2,3,4,5]

    let updatedArr = arr.map(num => {

        return <li>{num\*num}</li>

    })

    const items =[{name:'Item-1'}, {name:'Item-2'}]

    const updatedItems = items.map(item => (

            <p>{item.name}</p>

    ))

    const students = [{name:"Anita",rollno:101},

                      {name:"Sunita",rollno:102},

                      {name:"Kavita",rollno:103}]

    const updatedStudents = students.map(student => (

                <tr>

                    <td>{student.name} </td>

                    <td>{student.rollno} </td>

                </tr>

))

    return (

        <div>

            <h2>Numbers arr : </h2> {updatedArr}

            <h2>Items : </h2>  {updatedItems}

            <h2>Student Data</h2>

            <table border="1">

                <tr><th>Name</th><th>Roll no</th></tr>

                {updatedStudents}

            </table>

        </div>

    )

}

export default ListComponent

**app.js**

const DUMMY\_EXP = [

   { title: 'Groceries', amount: 900, date: new Date(2020, 7, 14)},

   { title: 'New TV', amount: 34000, date: new Date(2021, 2, 12) },

   { title: 'New Sofa Set', amount: 25000, date: new Date(2021, 2, 28)}

  ];

function App() {

  let [expenses, setExpenses] = useState(DUMMY\_EXP);

  /\*const expenses = [

    {title:"Shopping", amount:2000, date:new Date(2022, 1, 2)},

    {title:"Groceries", amount:7000, date:new Date(2022, 5, 6)},

    {title:"Movies", amount:4000, date:new Date(2022, 1, 5)}

  ]\*/

  const addExpenseHandler = expense => {

    console.log("In App component ", expense)

    setExpenses(prevArr => {return [expense, ...prevArr]})

    //console.log("Expense list ", expenses)

  }

  return (

    <div>

      <NewExpense onAddExpense={addExpenseHandler} />

      <h2>Let's get started!</h2>

      {expenses.map(expense =>

                   <ExpenseItem expDate={expense.date}

                                expTitle={expense.title}

                                expAmount={expense.amount} />)}

      {/\*<ExpenseItem expDate={expenses[0].date} expTitle={expenses[0].title} expAmount={expenses[0].amount}/>

      <ExpenseItem expDate={expenses[1].date} expTitle={expenses[1].title} expAmount={expenses[1].amount}/>

      <ExpenseItem expDate={expenses[2].date} expTitle={expenses[2].title} expAmount={expenses[2].amount}/>

    \*/}

      <Person name="Shrilata" age="52"/>

      <Person name="Soha" age="29"/>

      <Person name="Arnav" age="20"/>

      {/\*<MessageComponent />\*/}

      {/\*<UseStateComponent />\*/}

      {/\*<ClickCounter />\*/}

      <ListComponent />

    </div>

  );

}

export default App;

**ExpenseItem.js**

import { useState } from "react";

import ExpenseDate from "./ExpenseDate";

import "./ExpenseItem.css";

const ExpenseItem = (props) => {

   const [title, setTitle] = useState(props.expTitle);

    let

    btnHandler = () => {

        //setTitle("Updated Expense")

        console.log("Button clicked")

    }

    return (

        <div className="expense-item">

            <ExpenseDate expDate={props.expDate}></ExpenseDate>

            <div className="expense-item\_\_description">

                <h2>{props.expTitle}</h2>

                <p className="expense-item\_\_price">Rs {props.expAmount}</p>

            </div>

            <button onClick={btnHandler}>Change Title</button>

        </div>

    )

}

export default ExpenseItem;

**Conditional rendering:**

/\*const users = [

    { id: '1', firstName: 'Shrilata', lastName: 'T' },

    { id: '2', firstName: 'Anita', lastName: 'Patil' },

  ];\*/

  const users = []

  function ListUsers() {

    return (

      <div>

        <h1>Hello Conditional Rendering</h1>

        <List list={users} />

      </div>

    );

  }

  function List(props) {

    if (!props.list) {

      return null;

    }

    if (!props.list.length) {

        return <p>Sorry, the list is empty.</p>;

    }

    return (

      <ul>

        {props.list.map(item => (

          <Item key={item.id} item={item} />

        ))}

      </ul>

    );

  }

  function Item(props) {

    return (

      <li>

        {props.item.firstName} {props.item.lastName}

      </li>

    );

  }

  export default ListUsers;

**Class based components:**

import './User.css';

import React, {Component} from 'react';

class User extends Component{

render(){

// return <li className={user}>{props.name}</li>;

return <li className='user'>Hello User</li>

}

};

export default User;

import React, {Component} from 'react';

import classes from './Users.css';

const DUMMY\_USERS = [

    { id: 'u1', name: 'Max' },

    { id: 'u2', name: 'Manuel' },

    { id: 'u3', name: 'Julie' },

  ];

class Users extends Component{

    render(){

        return(<h2>Hello User</h2>)

    }

}

export default Users;

CounterComponent.js

import { Component } from "react";

class CounterComponent extends Component{

   state = {counter:1}

   btnHandler = () => {

     this.setState({counter: this.state.counter +1})

   }

   render(){

      return (

        <div>

            <div>Counter : {this.state.counter}</div>

            <button onClick={this.btnHandler}>Increment</button>

        </div>

      )

   }

}

export default CounterComponent

Error boundary

**Forms:**

const SimpleInput = (props) => {

    return (

      <form>

        <div className='form-control'>

          <label htmlFor='name'>Your Name</label>

          <input type='text' id='name' />

        </div>

        <div className="form-actions">

          <button>Submit</button>

        </div>

      </form>

    );

  };

  export default SimpleInput;

//index.css

\* {

  box-sizing: border-box;

}

html {

  font-family: 'Noto Sans JP', sans-serif;

}

body {

  margin: 0;

  background-color: #3f3f3f38;

}

.app {

  width: 90%;

  max-width: 43rem;

  padding: 1rem;

  border-radius: 12px;

  background-color: white;

  margin: 3rem auto;

}

.form-control {

  margin-bottom: 1rem;

}

.form-control input,

.form-control label {

  display: block;

}

.form-control label {

  font-weight: bold;

  margin-bottom: 0.5rem;

}

.form-control input,

.form-control select {

  font: inherit;

  padding: 0.5rem;

  border-radius: 4px;

  border: 1px solid #ccc;

  width: 20rem;

  max-width: 100%;

}

.form-control input:focus {

  outline: none;

  border-color: #240370;

  background-color: #e0d4fd;

}

.control-group {

  display: flex;

  column-gap: 1rem;

  flex-wrap: wrap;

}

.control-group .form-control {

  min-width: 15rem;

  flex: 1;

}

button {

  font: inherit;

  background-color: #240370;

  color: white;

  border: 1px solid #240370;

  padding: 0.5rem 1.5rem;

  border-radius: 4px;

  cursor: pointer;

}

button:hover,

button:active {

  background-color: #33059e;

  border-color: #33059e;

}

.form-actions {

  text-align: right;

}

.form-actions button {

  margin-left: 1rem;

}

.invalid input {

  border: 1px solid #b40e0e;

  background-color: #fddddd;

}

.invalid input:focus {

  border-color: #ff8800;

  background-color: #fbe8d2;

}

.error-text {

  color: #b40e0e;

}