Basic Concepts

Hello World Component

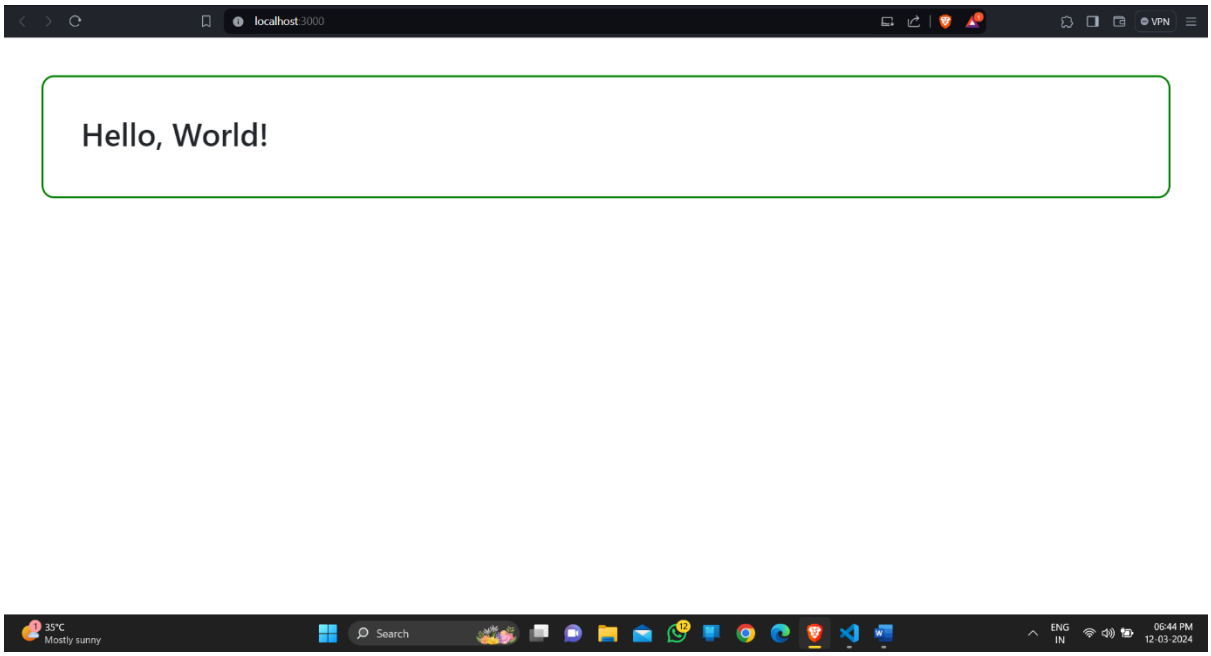
Greeting.js

```
import React from 'react';
const Greeting = () => {
  return (
    <h1>Hello, World!</h1>
  );
};
export default Greeting;
```

App.js

```
import './App.css';
import Greeting from './Greeting';
function App() {
  return (
    <Greeting />
  );
}
export default App;
```

OUTPUT:



Nested Components

App.js

```
import './App.css';
import Content from './Content';
import Footer from './Footer';
import Header from './Header';
function App() {
  return (
    <>
    <Header/>
    <Content />
    <Footer />
    </>
  );
}
export default App;
```

Header.js

```
import React from 'react'
const Header = () => {
  return (
    <div style={headerStyle}>Payhuddle</div>
  )
}
const headerStyle = {
    backgroundColor: '#06121F',
    color: '#fff',
    textAlign: 'center',
    fontSize: '2rem',
    fontWeight: 'bold',
    padding: '30px'
}
export default Header
```

Content.js

```
import React from 'react'
const Content = () => {
  return (
    <div style={contentStyle}>
        <h2>Enabling transaction performance with
confidence</h2>
      <p>
      It doesn’t matter whether you are an acquiring bank, a
payment processor, a terminal vendor, a payment scheme, or an
organisation that is in the business of enabling transactions
- we have you covered.       </p>
      <p>
      As a payment solutions organisation we can help you
ensure the confidence and performance of your transactions.
```

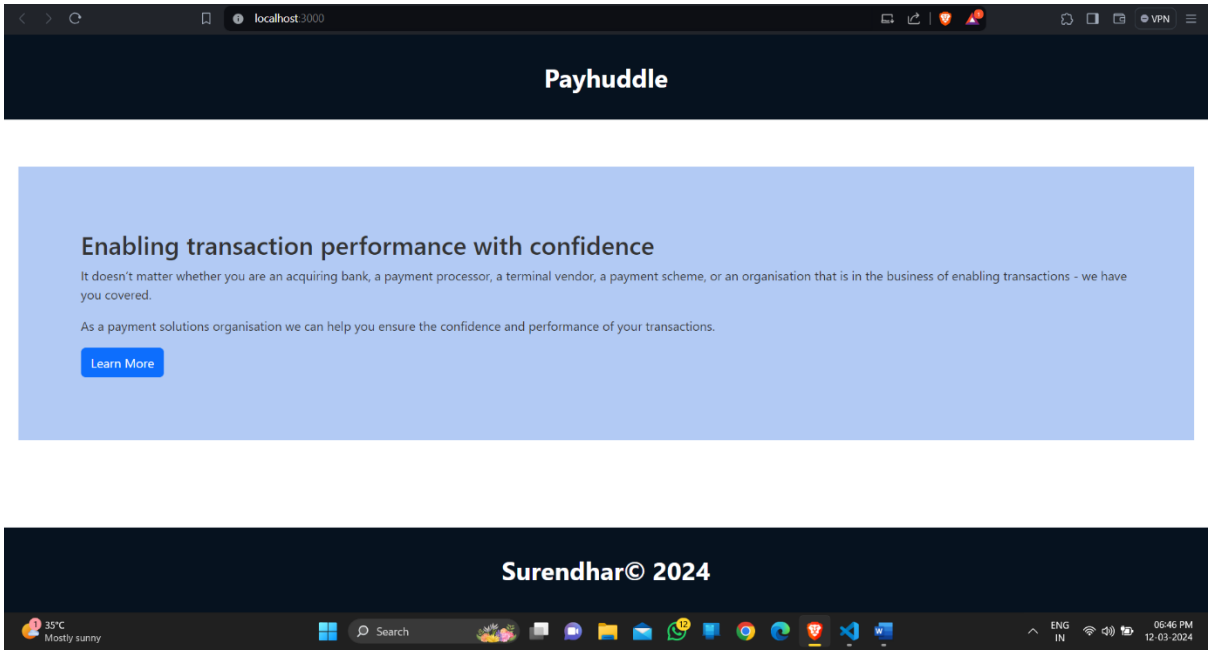
```
      </p>
      <button className='btn btn-primary'>Learn
More</button>
    </div>
  )
}
const contentStyle = {
    backgroundColor: '#B2CAF4',
    color: '#333',
    textAlign: 'start',
    padding: '80px',
    margin: '60px auto',
    maxWidth: '1500px'
}
export default Content
```

Footer.js

```
import React from 'react'
const Footer = () => {
  return (
    <div style={footerStyle}>
        Surendhar&copy; 2024
    </div>
  )
}
const footerStyle = {
    backgroundColor: '#06121F',
  color: '#fff',
  textAlign: 'center',
  fontSize: '2rem',
  fontWeight: 'bold',
```

```
  padding: '30px',
  position: 'fixed',
  bottom: '0',
  width: '100%'
}
export default Footer
```

Output:



JSX List Rendering

App.js

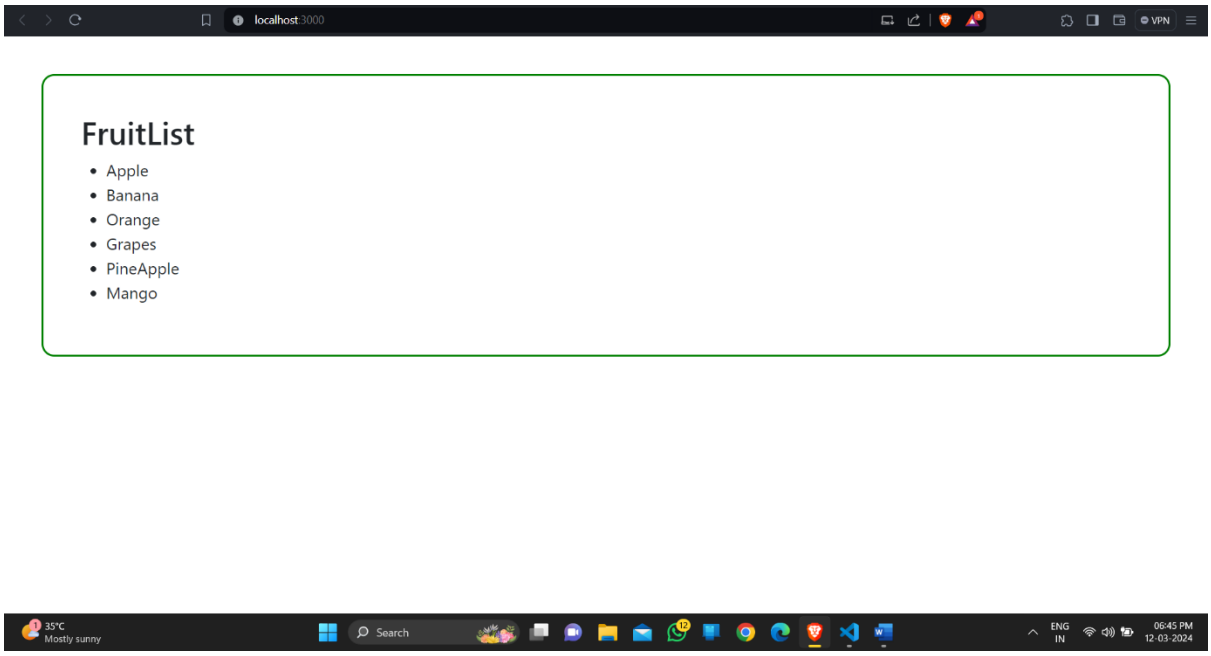
```
import './App.css';
import FruitList from './FruitList';
function App() {
  const fruits = ['Apple', 'Banana', 'Orange', 'Grapes',
'PineApple', 'Mango']
  return (
    <FruitList fruits = {fruits}
    />
  );
```

```
}
export default App;
```

FruitList.js

```
import React from 'react'
const FruitList = ({fruits}) => {
  return (
    <div style={{margin:'2rem'}}>
        <h1>FruitList</h1>
        <ul>
        {fruits.map((item) => (
            <li>{item}</li>
        ))}
        </ul>
    </div>
  )
}
export default FruitList
```

Output:



Props in Component

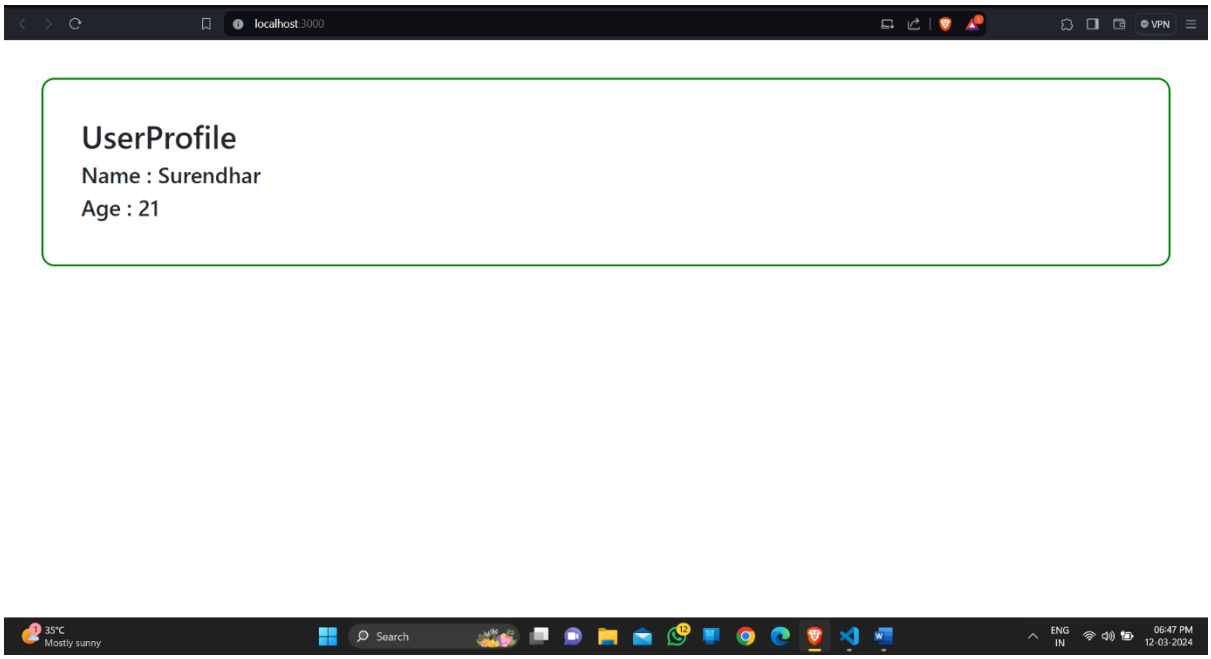
App.js

```
import React from 'react'
import UserProfile from './UserProfile'
const App = () => {
  const user = {
    name: 'Surendhar',
    age: 21,
  }
  return (
    <UserProfile
    user = {user}
    />
  )
}
export default App
```

UserProfile.js

```
import React from 'react'
const UserProfile = ({user}) => {
  return (
    <div>
        <h1>UserProfile</h1>
        <h3>Name : {user.name}</h3>
        <h3>Age  : {user.age}</h3>
    </div>
  )
}
export default UserProfile
```

Output:



State Management

App.js

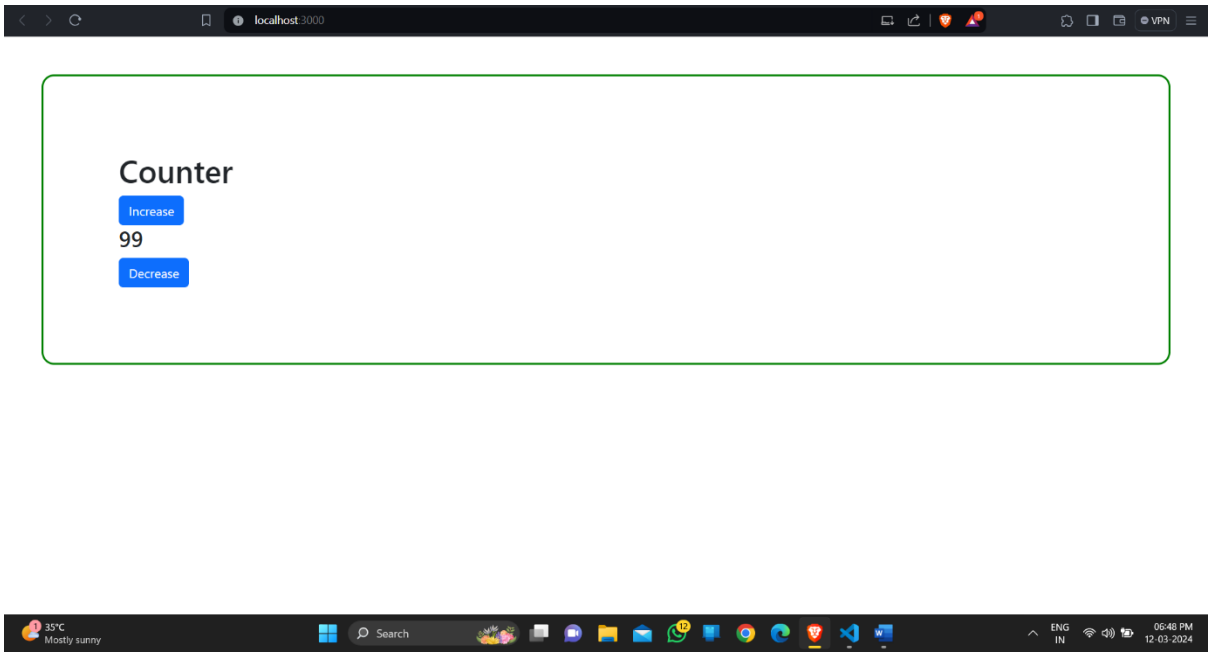
```
import React from 'react'
import Counter from './Counter'
const App = () => {
  return (
    <Counter />
  )
}
export default App
```

Counter.js

```
import React, { useState } from 'react'
const Counter = () => {
    const [count, SetCount] = useState(99);
    const handleIncrement = () =>{
        SetCount(count + 1);
    }
```

```
    const handleDecrement = () =>{
        SetCount(count - 1);
    }
  return (
    <div style={{margin:'3rem'}}>
        <h1>Counter</h1>
        <button className='btn btn-primary'
onClick={handleIncrement}>Increase</button>
        <h3>{count}</h3>
        <button className='btn btn-primary'
onClick={handleDecrement}>Decrease</button>
    </div>
  )
}
export default Counter
```

Output:



Event Handling

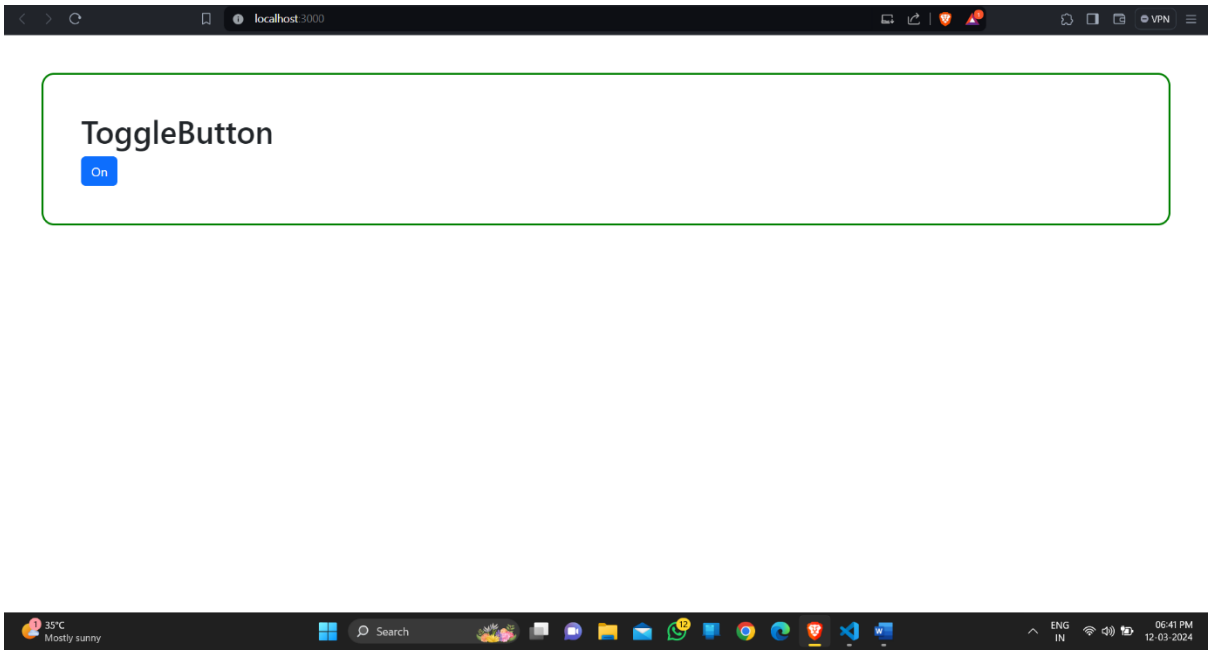
App.js

```
import React from 'react'
import ToggleButton from './ToggleButton'
const App = () => {
  return (
    <ToggleButton />
  )
}
export default App
```

ToggleButton.js

```
import React, { useState } from 'react'
const ToggleButton = () => {
    const [isOn, setIsOn] = useState(false);
    const handleToggle = ()=>{
        setIsOn(!isOn);
    }
  return (
    <div style={{margin: '5%'}}>
        <h1>ToggleButton</h1>
        <button className='btn btn-primary'
onClick={handleToggle}>{isOn ? 'On' : 'Off'}</button>
    </div>
  )
}
export default ToggleButton
```

Output:



Conditional Rendering

App.js

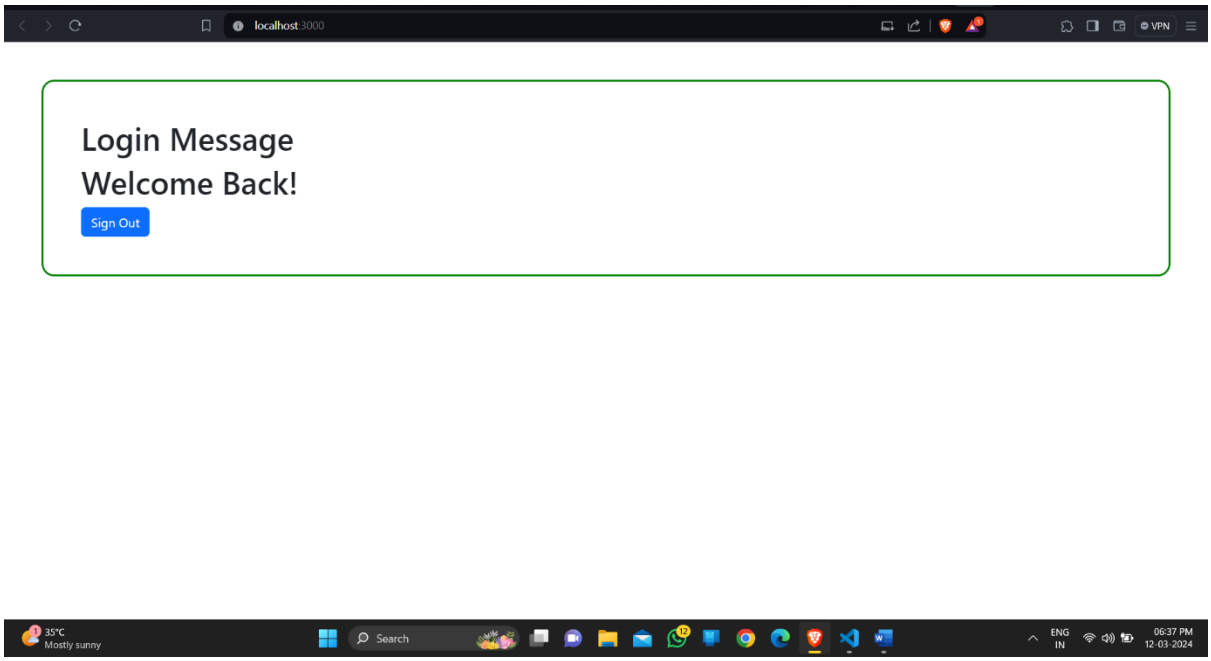
```
import React from 'react'
import LoginMessage from './LoginMessage'
const App = () => {
  return (
    <LoginMessage />
  )
}
```

LoginMessage.js

```
export default App
import React, { useState } from 'react'
const LoginMessage = () => {
    const [isLogged, setIsLogged] = useState(false);
  return (
    <div style={{margin:'4%'}}>
```

```
        <h1>LoginMessage</h1>
        {isLogged ? <h1>Welcome Back!</h1> :
<h1>Please sign in</h1>}
        <button className='btn btn-primary'
onClick={()=>setIsLogged(!isLogged)}>{isLogged ? 'Sign Out' :
'Sign In'}</button>
    </div>
  )
}
export default LoginMessage
```

Output:



Dynamic Content with map()

App.js

```
import React from 'react'
import UserList from './UserList'
const App = () => {
  const user = [
    {
      id: 1,
```

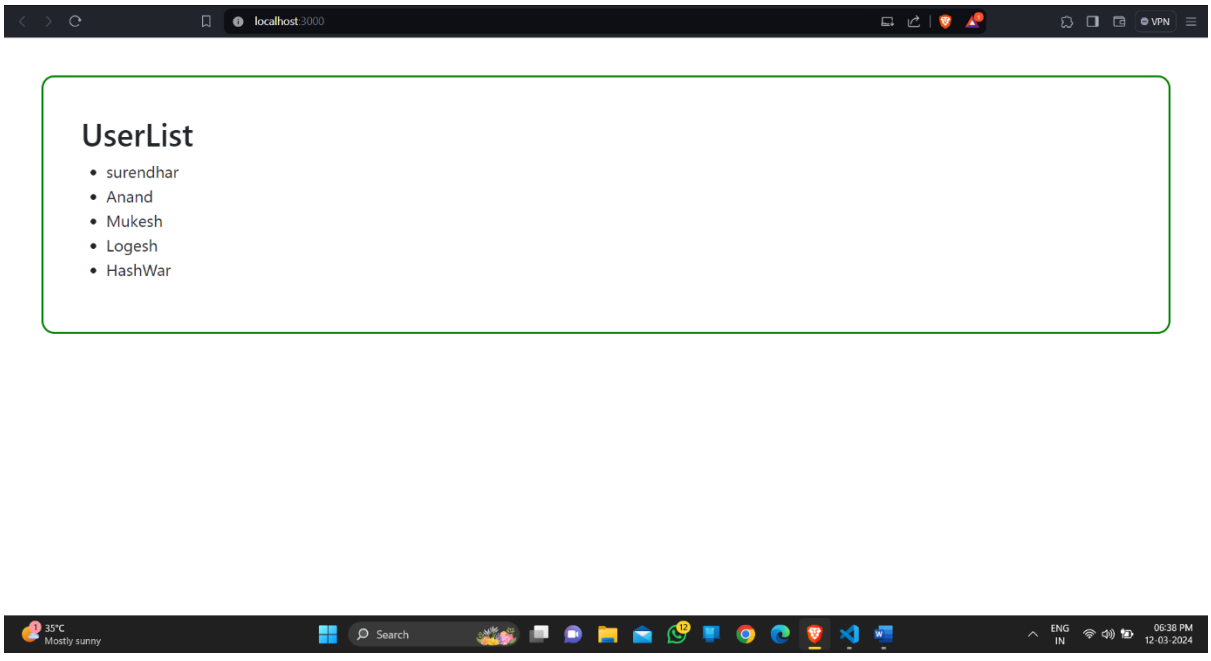
```
      name: "surendhar"
    },
    {
      id:2,
      name: "Anand"
    },
    {
      id:3,
      name: "Mukesh"
    },
    {
      id:4,
      name: "Logesh"
    },
    {
      id:5,
      name: "HashWar"
    }
  ]
  return (
    <UserList
    user = {user}
    />
  )
}
```

UserList.js

```
export default App
import React from 'react'
const UserList = ({user}) => {
```

```
  return (
    <div style={{margin:'3%'}}>
        <h1>UserList</h1>
        <ul>
        {user.map((item) => (
            <li>{item.name}</li>
        ))}
        </ul>
    </div>
  )
}
export default UserList
```

Output:



Form Handling

App.js

```
import React from 'react'
import NameForm from './NameForm'
const App = () => {
  return (
```

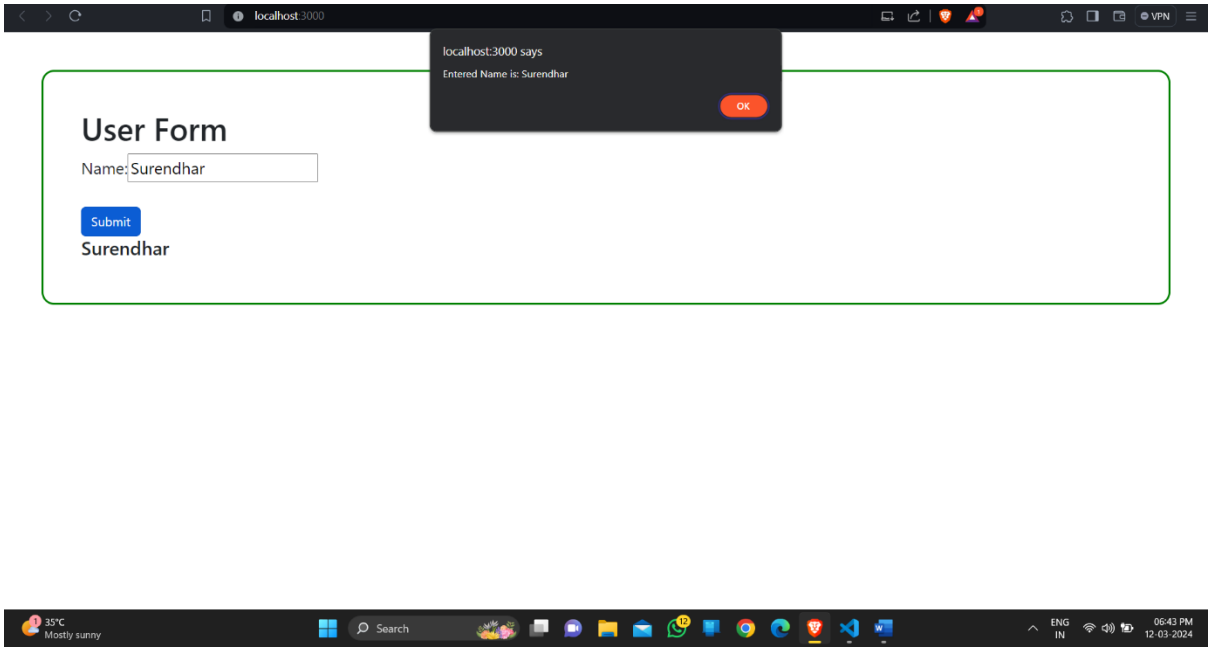
```
    <NameForm />
  )
}
export default App
```

NameForm.js

```
import React, { useState } from 'react'
const NameForm = () => {
    const [name, setName] = useState('');
    const handleChange = (event) =>{
        setName(event.target.value);
    }
    const handleSubmit = (event) =>{
        event.preventDefault();
        alert(`Entered Name is: ${name}`)
    }
  return (
    <div style={{margin: '3%'}}>
        <h1>User Form</h1>
        <form onSubmit={handleSubmit}>
            <label>Name: </label>
            <input type="text" value={name}
onChange={handleChange} />
            <br /> <br />
            <button className='btn btn-
primary'>Submit</button>
        </form>
        <h4>{name}</h4>
    </div>
  )
}
```

```
export default NameForm
```

Output:



Intermediate Concepts

React Router Setup

To Install Router:

Cmd : `npm install react-router-dom`

App.js

```
import React from 'react'
import { BrowserRouter, Route, Routes } from 'react-router-dom'
import Layout from './Layout'
import About from './About'
import Home from './Home'
import Contact from './Contact'
import NotFound from './NotFound'
const App = () => {
  return (
    <BrowserRouter>
```

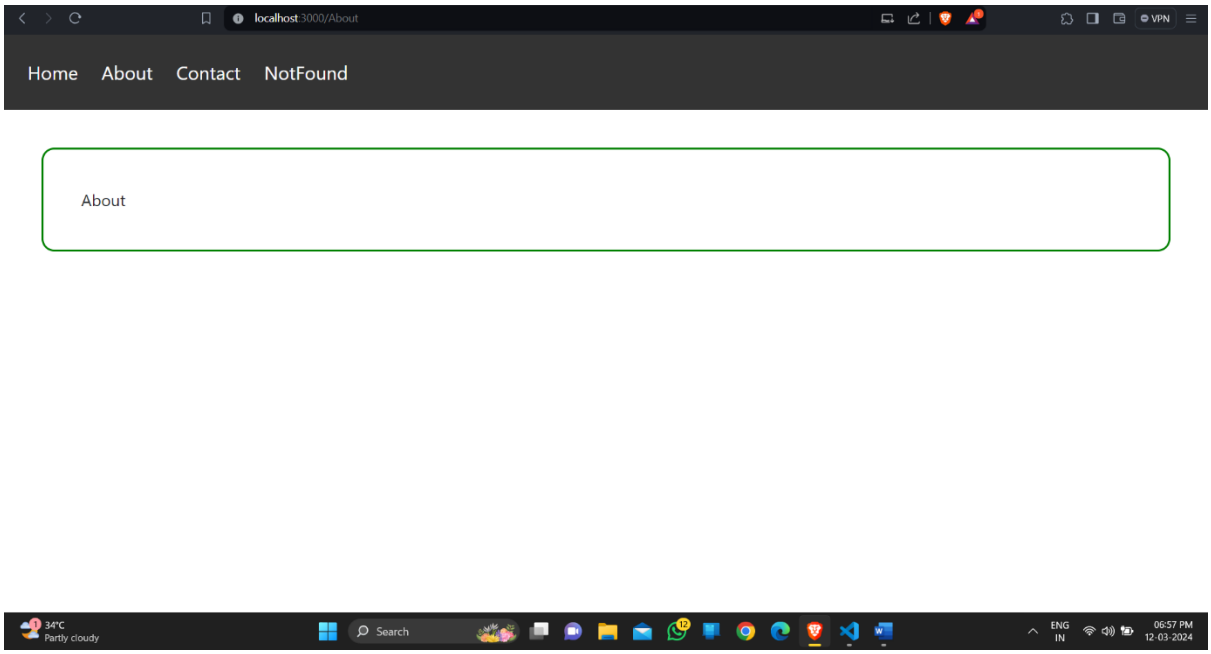
```
    <Routes>
      <Route path='/' element={<Layout />}>
        <Route index element={<Home />}></Route>
        <Route path='about' element={<About />}></Route>
        <Route path='contact' element={<Contact />}></Route>
        <Route path='notfound' element={<NotFound />}></Route>
      </Route>
    </Routes>
    </BrowserRouter>
  )
}
export default App
```

Layout.js

```
import React from 'react'
import { Link, Outlet } from 'react-router-dom'
const Layout = () => {
  return (
    <>
    <nav>
        <ul>
            <li>
                <Link to="/">Home</Link>
            </li>
            <li>
                <Link to="/About">About</Link>
            </li>
            <li>
                <Link to="/Contact">Contact</Link>
            </li>
```

```
            <li>
                <Link
to="/NotFound">NotFound</Link>
            </li>
        </ul>
    </nav>
      <Outlet />
      </>
  )
}
export default Layout
```

Output:



Data Fetching and Routing

To install axios

Cmd: npm install axios

Create a Local API

Cmd: npx json-server -p 3500 -w db.json

db.json:

```
{
    "users":[
        {
            "id": 1, "name": "John Doe", "email":
"john@example.com"
        },
        {
            "id": 2, "name": "Surendhar", "email":
"Surendhar@example.com"
        },
        {
            "id": 3, "name": "Anand", "email":
"Anand@example.com"
        },
        {
            "id": 4, "name": "Mukesh", "email":
"Mukesh@example.com"
        },
        {
            "id": 5, "name": "Surendhar", "email":
"Surendhar@example.com"
        }
    ]
}
```

App.js

```
import React from 'react'
import UserPage from './UserPage'
import { BrowserRouter, Route, Routes } from 'react-router-
dom'
import Layout from './Layout'
import Home from './Home'
```

```
const App = () => {
  return (
    <BrowserRouter>
    <Routes>
      <Route path='/' element={<Layout />}>
        <Route index element={<Home />}></Route>
        <Route path='userpage' element={<UserPage />}></Route>
      </Route>
    </Routes>
    </BrowserRouter>
  )
}
export default App
```

Layout.js

```
import React from 'react'
import { Link, Outlet } from 'react-router-dom'
const Layout = () => {
  return (
    <>
    <nav>
        <ul>
            <li>
                <Link to='/'>Home</Link>
            </li>
            <li>
                <Link to='/UserPage'>User Page</Link>
            </li>
        </ul>
    </nav>
```

```
    <Outlet />
    </>
  )
}
export default Layout
```

Home.js

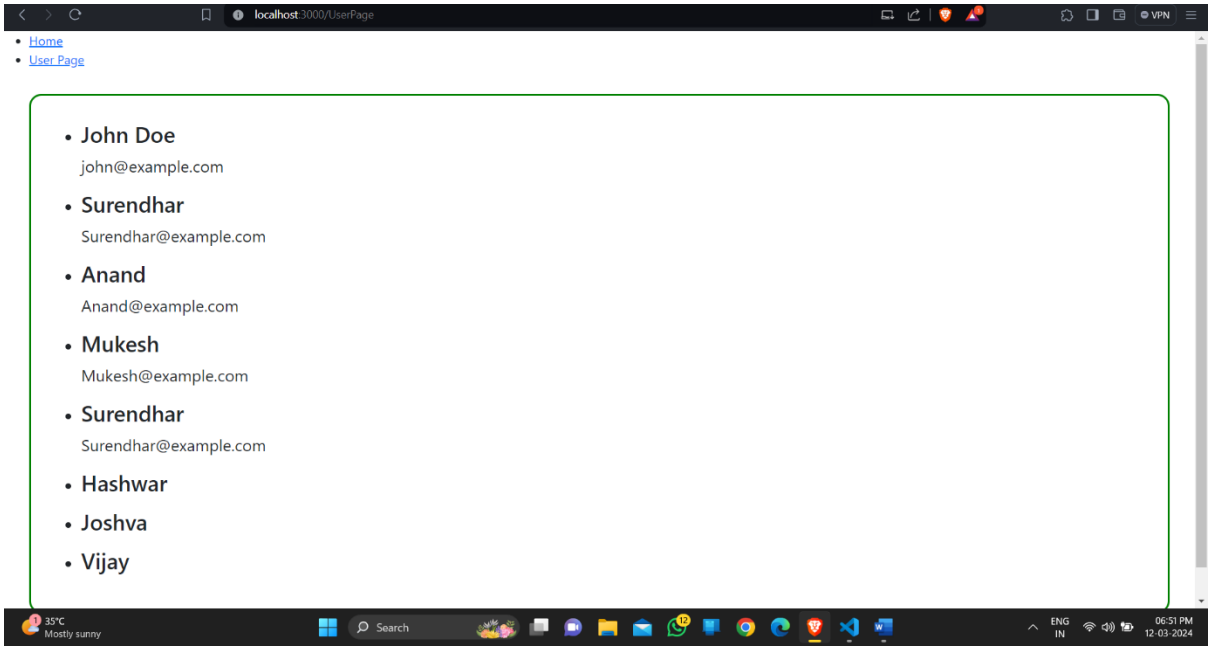
```
import React from 'react'
const Home = () => {
  return (
    <div>Home</div>
  )
}
export default Home
```

UserPage.js

```
import axios from 'axios'
import React, { useEffect, useState } from 'react'
const UserPage = () => {
    const [users, setUsers] = useState([])
    useEffect(()=>{
axios.get('http://localhost:8000/users').then(res=>setUsers(re
s.data)).catch(err=>console.log("Error"))
    },[])
  return (
    <div>
        {users.map(user=>(
            <ul>
                <li>
                    <h3>{user.name}</h3>
                    <p>{user.email}</p>
                </li>
```

```
            </ul>
        ))}
    </div>
  )
}
export default UserPage
```

Output;



API Data Pagination

To install the React-paginate

Cmd: npm install react-paginiate

App.js

```
import './App.css';
import PaginatedItems from './PaginatedItems';
function App() {
  return (
    <div className="main" >
      <PaginatedItems itemsPerPage={3}/>
```

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

PaginatedItems.js

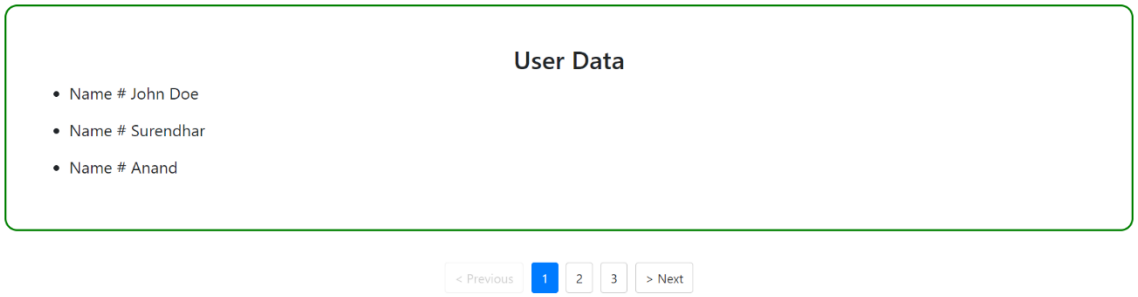
```
import React, { useEffect, useState } from 'react'
import ReactPaginate from 'react-paginate';
import './Pagination.css'
import axios from 'axios';

const PaginatedItems = ({itemsPerPage}) => {
    const [currentIndex, setCurrentIndex] = useState(0);
    useEffect(()=>{

axios.get('http://localhost:8000/users').then(res=>setItems(re
s.data));
    },[])
    const [items, setItems] = useState([]);
    const handleClick = ({selected}) =>{
        setCurrentIndex(selected)
    }
    const startIndex = currentIndex * itemsPerPage;
    const currentItems =items.slice(startIndex,startIndex +
itemsPerPage);
    const pageCount = Math.ceil(items.length / itemsPerPage);
  return (
    <>
    <div
    style={{
        margin: '3rem',
```

```
            border: '3px solid green',
            borderRadius: '1rem',
            padding: '3rem',
            fontSize: '1.3rem',
          }}>
            <h2 style={{textAlign:'center'}}>User Data</h2>
            {currentItems.map(item=>(
                <ul>
                    <li>Name # {item.name}</li>
                </ul>
            ))}
        </div>
        <ReactPaginate
        previousLabel={'< Previous'}
        nextLabel={'> Next'}
        pageCount={pageCount}
        onPageChange={handleClick}
        containerClassName={'pagination'}
        previousClassName={'previous'}
        nextClassName={'next'}
        disabledClassName={'disabled'}
        activeClassName={'active'}
       />
        </>
      )
}
export default PaginatedItems
```

Output:



Reusable Modal Component:

App.js

```
import  Modal  from './Modal'
import React, { useState } from 'react'
const App = () => {
  const [isOpen, setIsOpen] = useState(false);
  const handleOpen = ()=>{
    setIsOpen(true);
  }
  const OnClose =()=>{
    setIsOpen(false);
  }
  return (
    <div>
      <h1>Hi, I'm Surendhar S</h1>
      <button onClick={()=>handleOpen()}>Click Here</button>
      <Modal isOpen={isOpen} OnClose={OnClose}>
```

```
        <h3>From Namakkal</h3>
        <h4>Master of Computer Application</h4>
      </Modal>
    </div>
  )
}
export default App
```

Modal.js

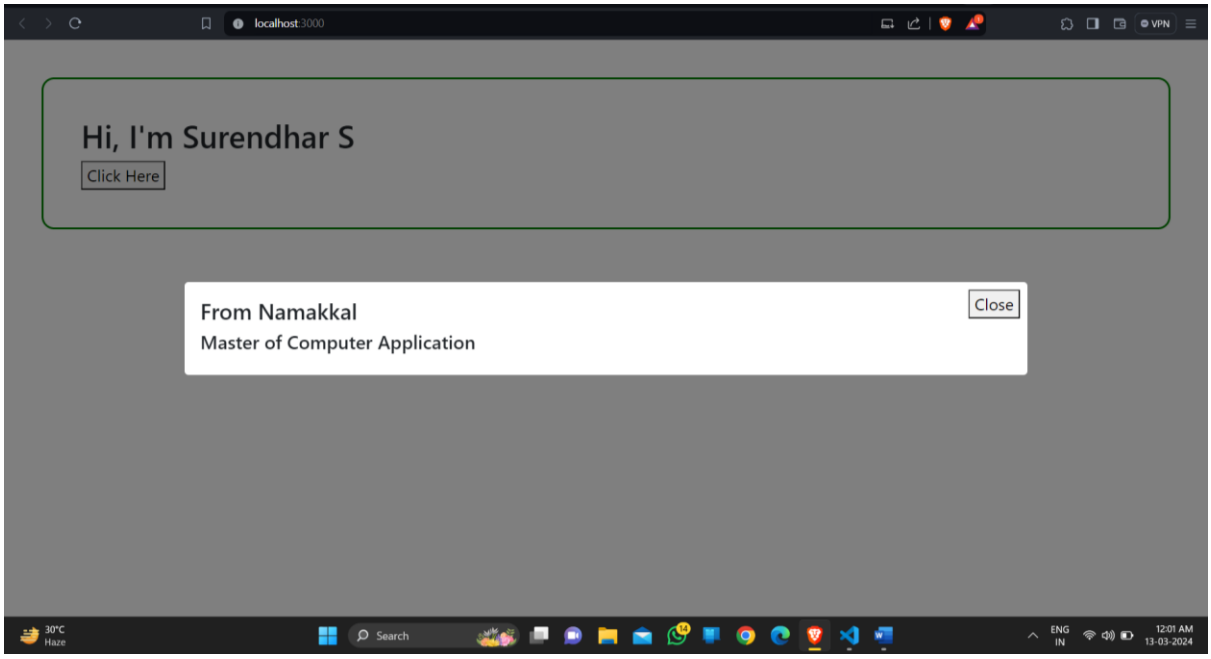
```
import React from 'react'
const Modal = ({isOpen, OnClose, children}) => {
    if(!isOpen) return null;
  return (
    <div className='modal-overlay'>
        <div className='modal-content'>
            <button className='close-btn'
onClick={()=>OnClose()}>Close</button>
            {children}
        </div>
    </div>
  )
}
export default Modal
```

App.css

```
.modal-overlay {
  position: fixed;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background-color: rgba(0, 0, 0, 0.5);
```

```
  display: flex;
  justify-content: center;
  align-items: center;
}
.modal-content {
  background-color: white;
  padding: 20px;
  width: 70%;
  border-radius: 5px;
}
.close-btn {
  position: absolute;
  top: 10px;
  right: 10px;
}
```

Output:



Effect Hook for Data Fetching

App.js

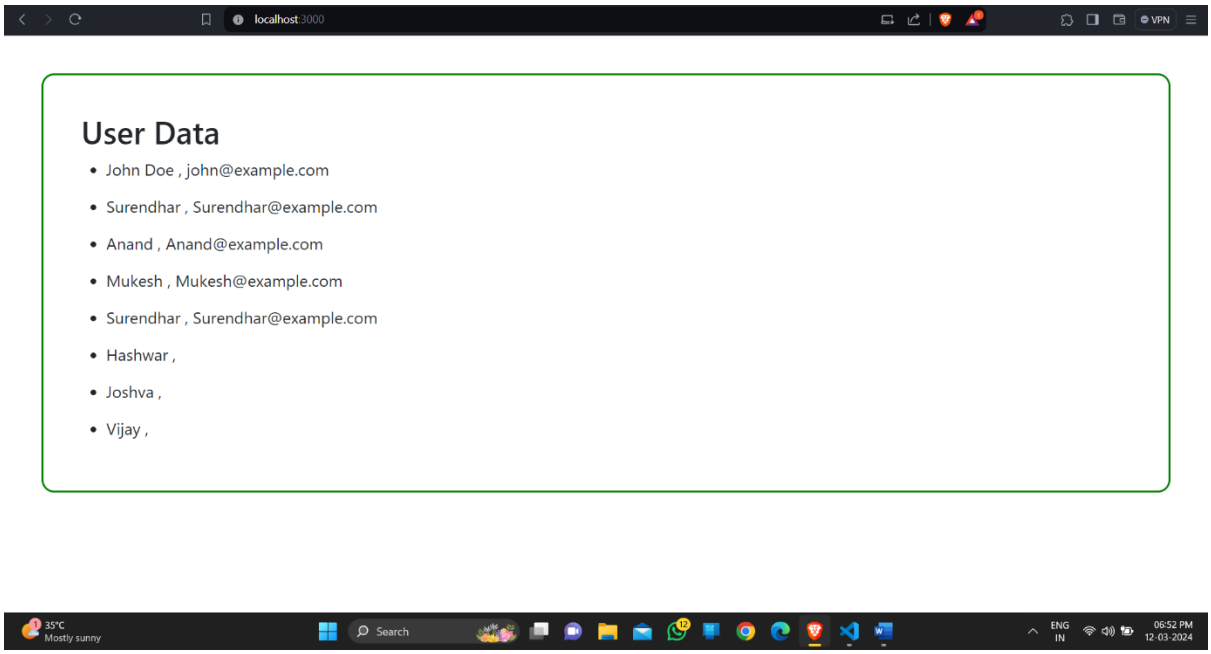
```
import React, { useEffect, useState } from 'react'
```

```
import ApiData from './ApiData'
const App = () => {
  const [data, setData] = useState([]);
  const API_URL = 'http://localhost:8000/users';
  useEffect(()=>{
    const fetchData =  async() =>{
      try{
      const res = await fetch(API_URL)
      const response = await res.json()
      setData(response);
      }
      catch
      {
        console.log("Error");
      }
    }
    fetchData()
  }
  ,[])
  return (
  <>
    <ApiData
    data = {data}
    />
    </>
  )
}
export default App
```

ApiData.js

```
import React from 'react'
const ApiData = ({data}) => {
  return (
    <div
    style={{
        margin: '3rem',
        border: '3px solid green',
        borderRadius: '1rem',
        padding: '3rem',
        fontSize: '1.3rem',
      }}
>
        <h1>User Data</h1>
        {data.map((item) => (
            <ul>
                <li>{item.name} , {item.email}</li>
            </ul>
        ))}
    </div>
  )
}
export default ApiData
```

Output:



Sorting List Items:

App.js

```
import React from 'react'
import UserList from './UserList'
const App = () => {
  const data = [
    {name: 'surendhar', age: 21},
    {name: 'Anand', age: 22},
    {name: 'Mukesh', age:25},
    {name: 'HashWar', age:27},
    {name: 'logesh', age:20}
  ]
  return (
   <UserList
   data ={data}
   />
  )
```

```
}
export default App
```

UserList.js

```
import React, { useState } from 'react'
const UserList = ({data}) => {
    const [sortBy, setSortBy] = useState(null);
    const [sortOrder, setSortOrder] = useState('asc');
    const handleSort=(criteria)=>{
        if(sortBy === criteria)
        {
            setSortOrder(sortOrder === 'asc' ? 'desc' : 'asc')
        }
        else
        {
            setSortBy(criteria);
            setSortOrder('asc');
        }
    }
    const sortedBy = [...data].sort((a, b)=>{
        if(sortBy === 'name')
        {
            return sortOrder === 'asc' ?
a.name.localeCompare(b.name) : b.name.localeCompare(a.name);
        }
        else
        {
            return sortOrder === 'asc' ? a.age - b.age : b.age
- a.age;
        }
        return 0;
```

```
    })
  return (
    <div
    style={{
        margin: '3rem',
        border: '3px solid green',
        borderRadius: '1rem',
        padding: '3rem',
        fontSize: '1.3rem',
      }}
>
        <h1>User Data</h1>
        <button onClick={()=>handleSort('name')}
style={{marginRight:'1.3rem'}}>Sort by Name</button>
        <button onClick={()=>handleSort('age')}>Sort by
Age</button>
        {sortedBy.map((user) => (
            <ul>
                <li>{user.name} - {user.age}</li>
            </ul>
        ))}
    </div>
  )
}
export default UserList
```

Output:

Dynamic Dropdown Menu:

App.js

```
import React from 'react'
import DropdownMenu from './DropdownMenu'
const App = () => {
  const fruits = [
    {
      name : 'Apple', advantage : 'a good source of nutrients'
    },
    {
      name : 'Orange', advantage : 'Forms blood vessels,
muscles'
    },
    {
      name : 'Banana', advantage : 'improve your digestion and
heart health'
    },
    {
```

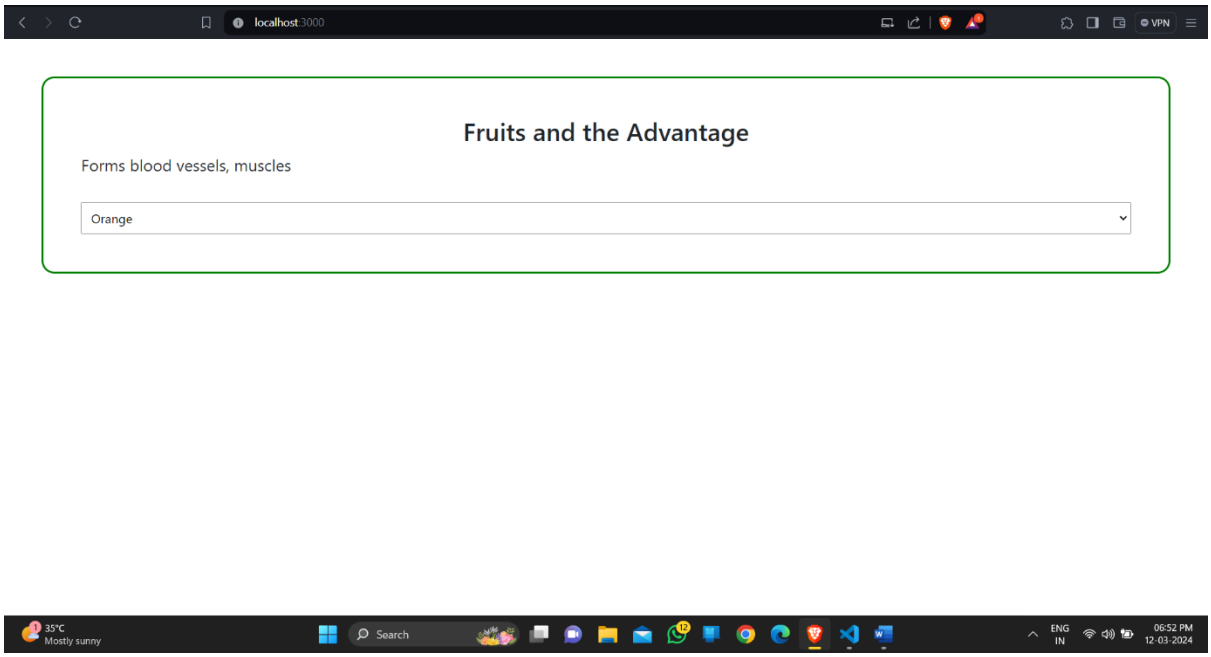
```
      name : 'Mango', advantage :
'strengthening your bones'
    }
  ]
  return (
    <DropdownMenu
    fruits = {fruits}
    />
  )
}
export default App
```

DropdownMenu.js

```
import React, { useState } from 'react';
const DropdownMenu = ({ fruits }) => {
  const [fruit, setFruit] = useState('');
  const handleFruit = (e) => {
    setFruit(e.target.value);
  };
  return (
    <div
      style={{
        margin: '3rem',
        border: '3px solid green',
        borderRadius: '1rem',
        padding: '3rem',
        fontSize: '1.3rem',
      }}
    >
      <h2 style={{ textAlign: 'center' }}>Fruits and the
Advantage</h2>
```

```
      {fruit === '' ? <p>No fruits</p> :
<p>{fruit}</p>}
      <select
        onChange={handleFruit}
        style={{
          padding: '0.5rem',
          fontSize: '1rem',
          marginTop: '1rem',
          width: '100%',
        }}
      >
        <option>--select the Fruits--</option>
        {fruits.map((fruit, index) => (
          <option key={index} value={fruit.advantage}>
            {fruit.name}
          </option>
        ))}
      </select>
    </div>
  );
};
export default DropdownMenu;
```

Output:



Styling with CSS Modules:

App.js

```
import React from 'react';
import Card from './Card';
const App = () => {
  return (
    <div>
      <Card
        title="Surendhar"
        content="Master of Computer Application"
      />
      <Card
        title="Anand"
        content="Master of Computer Science"
      />
      <Card
        title="Mukesh"
```

```
        content="Master of Physics"
      />
    </div>
  );
};
export default App;
```

Card.js

```
import React from 'react';
import styles from './Card.module.css';
const Card = ({ title, content }) => {
  return (
    <div className={styles.card}>
      <h2 className={styles.title}>{title}</h2>
      <p className={styles.content}>{content}</p>
    </div>
  );
};
export default Card;
```

Card.module.css:

```
.card {
    margin: 3rem;
    border: 1px solid #ccc;
    border-radius: 5px;
    padding: 20px;
    background-color: #f9f9f9;
    margin-bottom: 20px;
  }
  .title {
    color: #333;
```

```
    font-size: 1.5rem;
    margin-bottom: 10px;
  }
  .content {
    color: #666;
    font-size: 1.2rem;
  }
```

Output:

Form Validation:

App.js

```
import React from 'react'
import NameForm from './NameForm'
const App = () => {
  return (
    <div>
      <NameForm />
    </div>
  )
```

```
}
export default App
```

NameForm.js

```
import React, { useState } from 'react'
const NameForm = () => {
    const [name, setName] = useState('');
    const [error, setError] = useState('');
    const handleName = (e)=>{
        setName(e.target.value)
    }
    const handleForm =(e) =>{
        e.preventDefault();
        if(name.trim() === '')
        {
            setError("Name cannot be Empty..");
        }
        else if(!/^[a-zA-Z\s]+$/.test(name))
        {
            setError("Invalid Format")
        }
        else
        {
            console.log("Submitted name is : "+name);
            alert("Submitted name is: "+name);
            setError('');
            setName('');
        }
    }
  return (
```

```
    <form onSubmit={handleForm}
style={{margin:'2rem'}}>
        <label>Name  :</label>
        <input
        type='text'
        value={name}
        onChange={handleName}
        />
        {error && <p style={{color:'red'}}>{error}</p>}
        <button type='submit'>Submit</button>
    </form>
  )
}
export default NameForm
```

Output: