**Props**

function App() {

  const arr = [ { title: "The Dark Knight", image: '1.jpg' },   { title: "Batman", image: '2.jpg' },

    { title: "Spiderman", image: '3.jpg' },  { title: "Superman", image: '4.jpg' }]

  const handleButton = () => {

    console.log("Clicked");

  }

  return (

    <div className="App">

      {arr.map((item, index) => {

        return (

          <Card key={index} title={item.title} image={item.image} clicked={handleButton} >

            <h2 style={{ color: 'white' }}>ninos</h2>

            <h2 style={{ color: 'white' }}>nahrain</h2>

          </Card>

        ) })}

    </div>

  );}

export default App;

**Condition**

import './App.css';

import Card from './component/Card';

function App() {

**const arr = []**

  const handleButton = () => {

    console.log("Clicked");

  }

  return (

    <div className="App">

      {

**arr.length > 0 ?**

          (arr.map((item, index) => {

            return (

              <Card key={index} title={item.title} image={item.image} clicked={handleButton} >

                <h2 style={{ color: 'white' }}>ninos</h2>

                <h2 style={{ color: 'white' }}>nahrain</h2>

              </Card>

            )

          }))

**:**

**<h1 style={{ color: 'white' }}>No data</h1>**

      }

    </div>

  );

}

export default App;

**Handle Event**

import './App.css';

function App() {

**const handleClick = () => {**

**console.log("clicked");**

**}**

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

**console.log(e.target.value);**

**}**

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

**console.log(e.nativeEvent.data);**

**}**

  return (

    <div className="App">

      <>

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

**<button onClick={handleClick}>Submit</button>**

**<input type="text" onChange={displayLastLetter} />**

      </>

    </div>

  );

}

export default App;

**Routing**

**Install routing**

* npm i react-router-dom
* import { Link } from 'react-router-dom'
* const Home = () => {
* return (
* <div>
* <h2>Home</h2>
* <Link to="/header">Header</Link>
* <Link to="/footer">Footer</Link>
* </div>
* )
* }
* export default Home
* return (
* <div className="App">
* <Main />
* <BrowserRouter>
* <Routes>
* <Route path="/" element={<Home />} />
* <Route path="/header" element={<Header />} />
* <Route path="/footer" element={<Footer />} />
* <Route path="\*" element={<Error />} />
* </Routes>
* </BrowserRouter>
* </div>
* );

**Route example 2**

function App() {

  return (

    <div className="App">

      <h1>Welcome to My App</h1>

      <BrowserRouter>

        <Navbar />

        <Routes>

          <Route path="/" element={<Home />} />

          <Route path="/about" element={<About />} />

          <Route path="/contact" element={<Contact />} />

        </Routes>

      </BrowserRouter>

    </div>

  );

}

const Navbar = () => {

    return (

        <nav>

            <Link to="/" >Home</Link>

            <Link to="/about" >About</Link>

            <Link to="/contact" >Contact</Link>

        </nav>

    )}

**useState**

1. **Primitive types**

\*-Number

import { useState } from "react";

function App() {

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

  return (

    <div className="App">

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

**{count}**

**<button onClick={() => setCount(count - 1)}>-</button>**

    </div>

  );

}

export default App;

\*-String

function App() {

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

  return (

    <div className="App">

**<button onClick={() => setName("John")}>Change Name</button>**

**<p>Hello, {name}!</p>**

    </div>

  );

}

export default App;

\*-boolean

function App() {

  const [isVisible, setIsVisible] = useState(true);

  return (

    <div className="App">

      <p>Visible: {isVisible.toString()}</p>

      <button onClick={() => setIsVisible(!isVisible)}>Toggle Visibility</button>

    </div>

  );

}

export default App;

1. **Array**
2. function App() {
3. const [items, setItems] = useState(["apple", "banana"]);
4. const [inputValue, setInputValue] = useState(""); // store input text
5. const handleAddItem = () => {
6. if (inputValue.trim() !== "") {
7. setItems([...items, inputValue]);
8. setInputValue(""); // clear input after adding
9. }
10. };
11. const handleInput = (e) => {
12. setInputValue(e.target.value);
13. };
14. return (
15. <div className="App">
16. <ul>
17. {items.map((item, index) => (
18. <li key={index}>{item}</li>
19. ))}
20. </ul>
21. <input
22. value={inputValue}
23. onChange={handleInput}
24. type="text"
25. placeholder="Enter an item"
26. />
27. <button onClick={handleAddItem}>Add Item</button>
28. </div>
29. );
30. }
31. export default App;

function App() {

**const [user, setUser] = useState({**

**name: "Alice",**

**age: 25**

**});**

**const handleAge = () => {**

**setUser({**

**...user,**

**age: user.age + 1**

**})**

**}**

  return (

    <div className="App">

      <h2>Object</h2>

**<p>User: {user.name}, Age: {user.age}</p>**

**<button onClick={handleAge}>Increase Age</button>**

    </div>

  );}

1. **Object**
2. function App() {
3. const [user, setUser] = useState({
4. name: "Alice",
5. age: 25
6. });
7. const handleAge = () => {
8. setUser({
9. ...user,
10. age: user.age + 1
11. })
12. }
13. return (
14. <div className="App">
15. <h2>Object</h2>
16. <p>User: {user.name}, Age: {user.age}</p>
17. <button onClick={handleAge}>Increase Age</button>
18. </div>
19. );
20. }
21. export default App;

**useEffect**

function App() {

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

**const [x, setX] = useState(0);**

**// This useEffect will run after every render**

**// useEffect(() => {**

**//   console.log("test")**

**// })**

**// This useEffect will run only once**

**// useEffect(() => {**

**//   console.log("test")**

**// }, [])**

**// This useEffect will run when 'count' changes**

**useEffect(() => {**

**console.log("test")**

**}, [count])**

  return (

    <div className="App">

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

      {count}

      <button onClick={() => setCount(count - 1)}>-</button>

      <br /><br />

      <button onClick={() => setX(x + 3)}>+3</button>

      {x}

      <button onClick={() => setX(x - 3)}>-3</button>

    </div>

  );

}

export default App;

**useRef**

function App() {

**const r = useRef(null);**

**useEffect(() => {**

**r.current.focus();**

**})**

  return (

    <div className="App">

**<input ref={r} type="text" placeholder="Enter your name" />**

    </div>

  );

}

export default App;

**useContext**

1. **create a context**

**NinoosContext.ts**

import { createContext } from "react";

**export const Ninoos = createContext<string | null>(null);**

1. **create a provider**

**One.tsx**

import { Ninoos } from "./NinoosContext";

import Two from "./Two";

export default function One() {

    const **message** = "from One";

    return (

        <div className="one">

            <h2>component One</h2>

**<Ninoos.Provider value={message}>**

                <Two />

**</Ninoos.Provider>**

        </div>

    )

}

**Two.tsx**

import Three from "./Three";

export default function Two() {

    return (

        <div className="two">

            <h2>component Two</h2>

            <Three />

        </div>

    )

}

**Three.tsx**

import { useContext } from "react"

import { Ninoos } from "./NinoosContext"

export default function Three() {

**const valueMessage = useContext(Ninoos)**

    return (

        <div className="three">

            <h2>component Three</h2>

**{valueMessage}**

        </div>

    )

}