**1. Create a Simple React App Display “Hello, World!” inside an Tag.**

//app.jsimport "./App.css";

function App() {

return (

<div>

<h1>Hello, World!</h1>

</div>

);

}

export default App;

//**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import "./index.css";

import App from "./App";

import reportWebVitals from "./reportWebVitals";

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(

<React.StrictMode>

<App />

</React.StrictMode>

);

****

**2. Write JSX to Display a Paragraph Inside a , display a tag with your name.**

//app.js

import "./App.css";

function App() {

return (

<div>

<p>Mansi shelar</p>

</div>

);

}

export default App;

//index.js

import React from "react";

import ReactDOM from "react-dom/client";

import "./index.css";

import App from "./App";

import reportWebVitals from "./reportWebVitals";

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(

<React.StrictMode>

<App />

</React.StrictMode>

);



**3. Create a JSX expression that renders the current date and time dynamically**

//app.js

import "./App.css";

function App() {

const currentDate = new Date().toLocaleDateString();

const currentTime = new Date().toLocaleTimeString();

return (

<div>

<p>

{currentDate} - {currentTime}

</p>

</div>

);

}

export default App;

//index.js

import React from "react";

import ReactDOM from "react-dom/client";

import "./index.css";

import App from "./App";

import reportWebVitals from "./reportWebVitals";

const root = ReactDOM.createRoot(document.getElementById("root"));

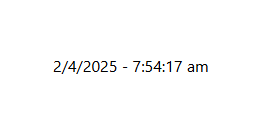
root.render(

<React.StrictMode>

<App />

</React.StrictMode>

);



**4. Create a simple functional component that returns a heading**

**//app.js**

**import React from 'react';**

**import Head from './Head';**

**function App() {**

**return (**

**<div>**

**<Head/>**

**</div>**

**);**

**}**

**export default App;**

**//head.js**

**// File: Heading.js**

**import React from 'react';**

**function Head() {**

**return <h2>This is a heading</h2>;**

**}**

**export default Head;**

//Head.js

import "./Head.css";

function Head() {

return (

<div>

<h2>This is a Heading component.</h2>

</div>

);

}

export default Head;

//index.js

import React from "react";

import ReactDOM from "react-dom/client";

import "./index.css";

import Head from "./heading";

import reportWebVitals from "./reportWebVitals";

const root = ReactDOM.createRoot(document.getElementById("root"));

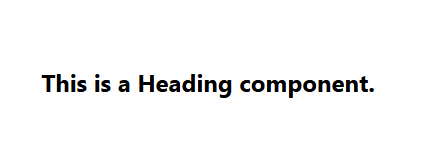
root.render(

<React.StrictMode>

<Head />

</React.StrictMode>

);



**5. Write a program to demonstration of nested functional component.**

Q.5

**🔹 Step-by-step Code**

**1. index.js**

Entry point of the React app.

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

Main component that uses ParentComponent.

jsx

CopyEdit

import React from 'react';

import ParentComponent from './components/ParentComponent';

function App() {

return (

<div>

<h1>Nested Functional Components Demo</h1>

<ParentComponent />

</div>

);

}

export default App;

**3. components/ParentComponent.js**

Parent component that includes ChildComponent.

jsx

CopyEdit

import React from 'react';

import ChildComponent from './ChildComponent';

function ParentComponent() {

return (

<div style={{ border: '2px solid blue', padding: '10px', margin: '10px' }}>

<h2>This is the Parent Component</h2>

<ChildComponent />

</div>

);

}

export default ParentComponent;

**4. components/ChildComponent.js**

Child component nested inside the parent.

jsx

CopyEdit

import React from 'react';

function ChildComponent() {

return (

<div style={{ border: '2px solid green', padding: '10px', margin: '10px' }}>

<h3>This is the Child Component</h3>

</div>

);

}

export default ChildComponent;

6. Create a parent component that passes a name prop to a child component and displays it.

**🔹 Code Files**

**1. index.js**

🔸 Entry point for the React app

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Loads ParentComponent

jsx

CopyEdit

import React from 'react';

import ParentComponent from './components/ParentComponent';

function App() {

return (

<div>

<h1>Props Passing Demo</h1>

<ParentComponent />

</div>

);

}

export default App;

**3. components/ParentComponent.js**

🔸 Passes a name prop to the child

jsx

CopyEdit

import React from 'react';

import ChildComponent from './ChildComponent';

function ParentComponent() {

const name = "Mansi";

return (

<div style={{ border: '2px solid blue', padding: '10px', margin: '10px' }}>

<h2>This is the Parent Component</h2>

<ChildComponent name={name} />

</div>

);

}

export default ParentComponent;

**4. components/ChildComponent.js**

🔸 Receives the name prop and displays it

jsx

CopyEdit

import React from 'react';

function ChildComponent({ name }) {

return (

<div style={{ border: '2px solid green', padding: '10px', margin: '10px' }}>

<h3>This is the Child Component</h3>

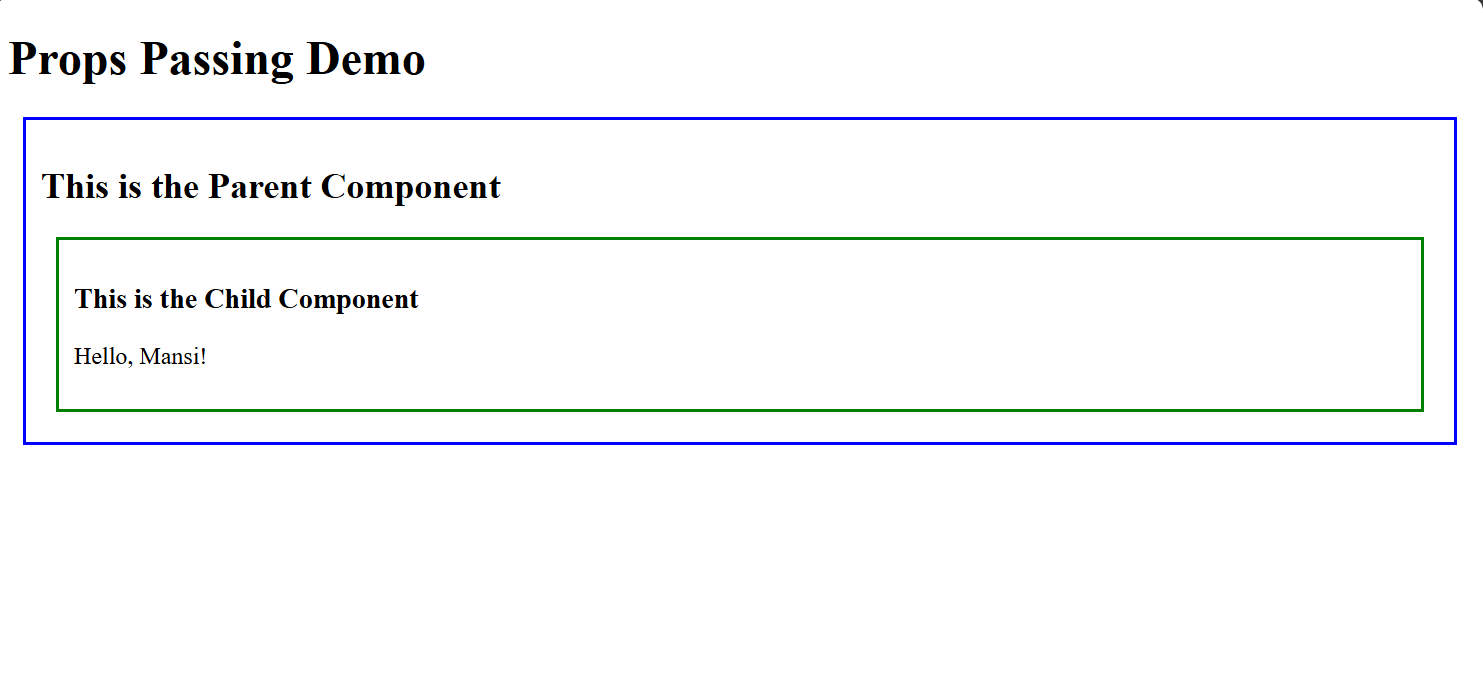
<p>Hello, {name}!</p>

</div>

);

}

export default ChildComponent;



**7. Create an array of students which includes roll number, name and age pass it as a props to**

**functional component and display all information**.

**🔹 Code Files**

**1. index.js**

🔸 Entry point for the app.

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Create array of students and pass it to StudentList as props.

jsx

CopyEdit

import React from 'react';

import StudentList from './components/StudentList';

function App() {

const students = [

{ roll: 101, name: 'Mansi Shelar', age: 22 },

{ roll: 102, name: 'Prashant Pawde', age: 23 },

{ roll: 103, name: 'Sneha Patil', age: 21 },

];

return (

<div>

<h1>Student List</h1>

<StudentList students={students} />

</div>

);

}

export default App;

**3. components/StudentList.js**

🔸 Receives and displays the array using .map().

jsx

CopyEdit

import React from 'react';

function StudentList({ students }) {

return (

<div style={{ border: '2px solid gray', padding: '15px', margin: '10px' }}>

<h2>Student Information</h2>

{students.map((student) => (

<div key={student.roll} style={{ marginBottom: '10px' }}>

<p><strong>Roll No:</strong> {student.roll}</p>

<p><strong>Name:</strong> {student.name}</p>

<p><strong>Age:</strong> {student.age}</p>

<hr />

</div>

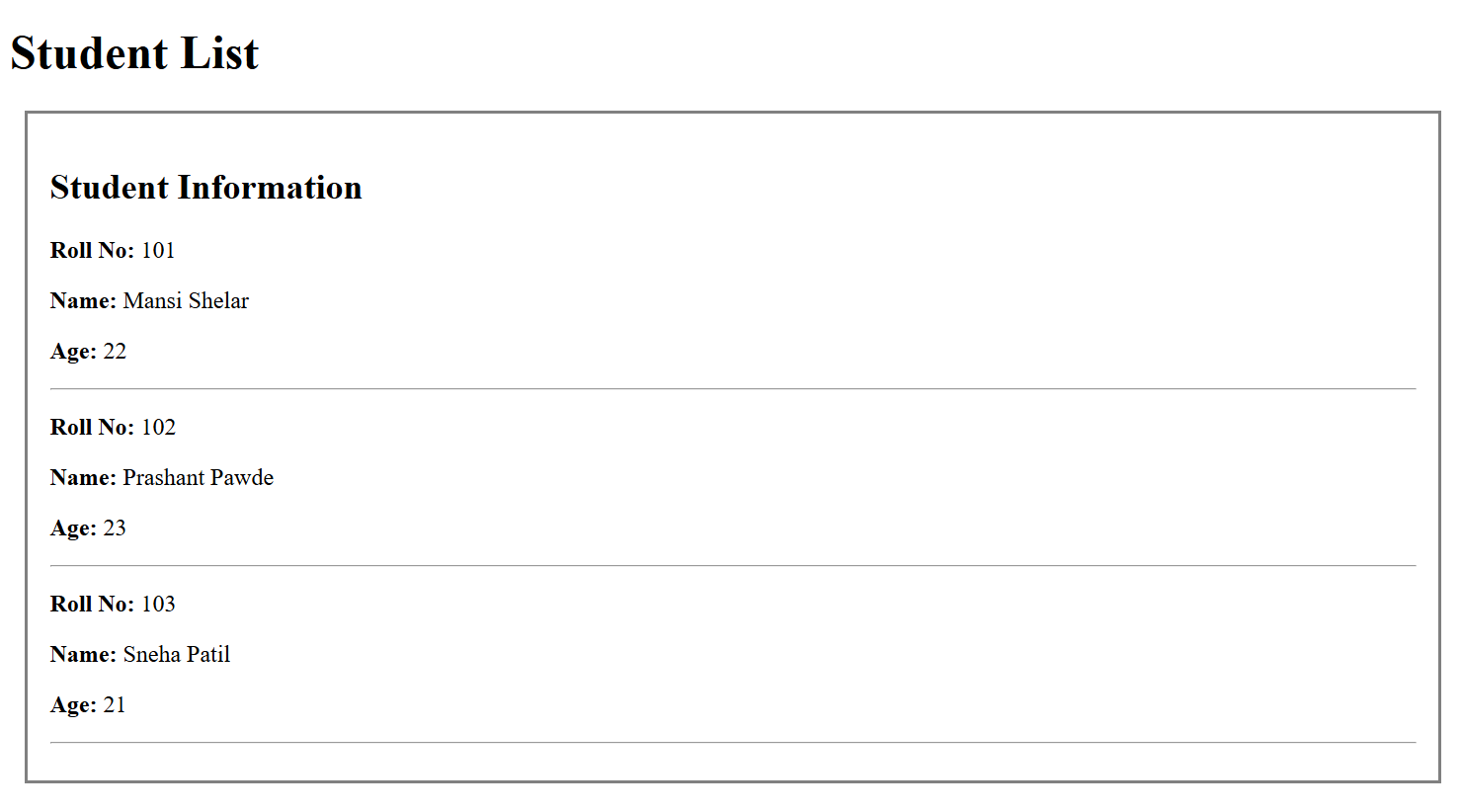
))}

</div>

);

}

export default StudentList;



**8. Pass an array of colors (as props) to a child component and render them as <li> items.**

**🔹 Code Files**

**1. index.js**

🔸 Entry point for the React app.

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Create an array of colors and pass it to the ColorList component.

jsx

CopyEdit

import React from 'react';

import ColorList from './components/ColorList';

function App() {

const colors = ['Red', 'Green', 'Blue', 'Yellow', 'Purple'];

return (

<div>

<h1>Color List</h1>

<ColorList colors={colors} />

</div>

);

}

export default App;

**3. components/ColorList.js**

🔸 Receives the array of colors and displays them as <li> items.

jsx

CopyEdit

import React from 'react';

function ColorList({ colors }) {

return (

<div>

<h2>Colors</h2>

<ul>

{colors.map((color, index) => (

<li key={index}>{color}</li>

))}

</ul>

</div>

);

}

export default ColorList;

****

**9. Build a counter with two buttons (+ and -) to increment/decrement a state variable.**

**🔹 Code Files**

**1. index.js**

🔸 Main entry point of the React app

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Loads the Counter component

jsx

CopyEdit

import React from 'react';

import Counter from './components/Counter';

function App() {

return (

<div>

<h1>Simple Counter App</h1>

<Counter />

</div>

);

}

export default App;

**3. components/Counter.js**

🔸 Functional component with state and two buttons

jsx

CopyEdit

import React, { useState } from 'react';

function Counter() {

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

const increment = () => setCount(count + 1);

const decrement = () => setCount(count - 1);

return (

<div style={{ border: '2px solid #888', padding: '20px', width: '200px', textAlign: 'center' }}>

<h2>Count: {count}</h2>

<button onClick={increment} style={{ marginRight: '10px' }}>+</button>

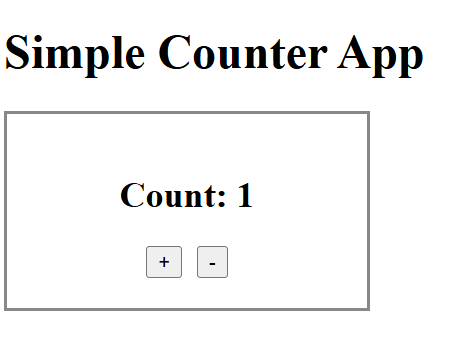
<button onClick={decrement}>-</button>

</div>

);

}

export default Counter;



**10. Create an input field that logs its value to the console on every keystroke (use onChange).**

✅ **Create an input field that logs its value to the console on every keystroke using onChange.**

## 🔹 Code Files

### 1. index.js

🔸 Entry point of the app

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

### 2. App.js

🔸 Loads the InputLogger component

jsx

CopyEdit

import React from 'react';

import InputLogger from './components/InputLogger';

function App() {

return (

<div>

<h1>Input Logger</h1>

<InputLogger />

</div>

);

}

export default App;

### 3. components/InputLogger.js

🔸 Logs input value to the console on each keystroke

jsx

CopyEdit

import React from 'react';

function InputLogger() {

const handleChange = (event) => {

console.log("Input value:", event.target.value);

};

return (

<div>

<input

type="text"

placeholder="Type something..."

onChange={handleChange}

style={{ padding: '8px', fontSize: '16px' }}

/>

</div>

);

}

export default InputLogger;

**11. Create a button that changes its background color when clicked (use onClick and state).**

**🔹 Code Files**

**1. index.js**

🔸 Main entry point for the React app

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Loads the ColorButton component

jsx

CopyEdit

import React from 'react';

import ColorButton from './components/ColorButton';

function App() {

return (

<div>

<h1>Button Color Toggle</h1>

<ColorButton />

</div>

);

}

export default App;

**3. components/ColorButton.js**

🔸 Button that changes color on every click

jsx

CopyEdit

import React, { useState } from 'react';

function ColorButton() {

const [color, setColor] = useState('blue'); // initial color is blue

const changeColor = () => {

// Toggle between blue and green

setColor((prevColor) => (prevColor === 'blue' ? 'green' : 'blue'));

};

return (

<button

onClick={changeColor}

style={{

backgroundColor: color,

color: 'white',

padding: '10px 20px',

fontSize: '16px',

border: 'none',

borderRadius: '5px',

cursor: 'pointer',

}}

>

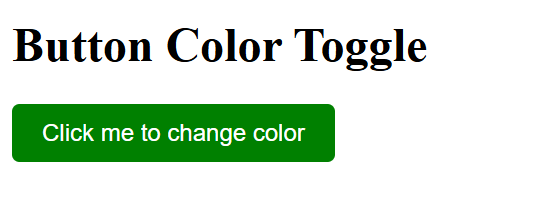
Click me to change color

</button>

);

}

export default ColorButton;



**12. Build a form with name and email fields that alerts the submitted values (use onSubmit).**

**🔹 Code Files**

**1. index.js**

🔸 Main entry point for the React app.

jsx

CopyEdit

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**2. App.js**

🔸 Loads the ContactForm component.

jsx

CopyEdit

import React from 'react';

import ContactForm from './components/ContactForm';

function App() {

return (

<div>

<h1>Form Submission</h1>

<ContactForm />

</div>

);

}

export default App;

**3. components/ContactForm.js**

🔸 A form with name and email fields that alerts the submitted values.

jsx

CopyEdit

import React, { useState } from 'react';

function ContactForm() {

const [name, setName] = useState('');

const [email, setEmail] = useState('');

const handleSubmit = (e) => {

e.preventDefault(); // Prevents the form from refreshing the page

alert(`Name: ${name}, Email: ${email}`);

};

return (

<form onSubmit={handleSubmit} style={{ maxWidth: '400px', margin: '0 auto', padding: '20px', border: '1px solid #ccc' }}>

<div style={{ marginBottom: '10px' }}>

<label htmlFor="name" style={{ display: 'block', marginBottom: '5px' }}>Name:</label>

<input

type="text"

id="name"

value={name}

onChange={(e) => setName(e.target.value)}

placeholder="Enter your name"

style={{ padding: '8px', width: '100%' }}

/>

</div>

<div style={{ marginBottom: '10px' }}>

<label htmlFor="email" style={{ display: 'block', marginBottom: '5px' }}>Email:</label>

<input

type="email"

id="email"

value={email}

onChange={(e) => setEmail(e.target.value)}

placeholder="Enter your email"

style={{ padding: '8px', width: '100%' }}

/>

</div>

<button type="submit" style={{ padding: '10px 20px', backgroundColor: '#4CAF50', color: 'white', border: 'none', cursor: 'pointer' }}>

Submit

</button>

</form>

);

}

export default ContactForm;

