

Front-end Using React, Angular JS

1. Create a React component to display “Hello, World!” using JSX.

```
function HelloWorld() {  
  return <h1>Hello, World!</h1>;  
}
```

2. Create a React component to display your name and age using props.

```
function UserInfo(props) {  
  return <p>Name: {props.name}, Age: {props.age}</p>;  
}
```

3. Write a React component to display a simple unordered list of three hobbies.

```
function HobbiesList() {  
  return (  
    <ul>  
      <li>Reading</li>  
      <li>Gaming</li>  
      <li>Traveling</li>  
    </ul>  
  );  
}
```

4. Create a component with a button that increments a counter on click.

```
function Counter() {  
  const [count, setCount] = useState(0);  
  return (  
    <div>  
      <p>Count: {count}</p>  
      <button onClick={() => setCount(count + 1)}>Increment</button>  
    </div>  
  );  
}
```

5. Implement a simple toggle button using React state.

```
function ToggleButton() {  
  const [isOn, setIsOn] = useState(true);  
  return (  
    <button onClick={() => setIsOn(!isOn)}>  
      {isOn ? 'ON' : 'OFF'}  
    </button>  
  );  
}
```

6. Create a component that takes user input and displays it below the input field.

```
function InputDisplay() {
  const [input, setInput] = useState('');
  return (
    <div>
      <input onChange={(e) => setInput(e.target.value)} />
      <p>{input}</p>
    </div>
  );
}
```

7. Set up React Router and create routes for "Home" and "About" pages.

```
import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';
```

```
function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
      </Routes>
    </Router>
  );
}
```

8. Create a navigation bar using Link from React Router.

```
import { Link } from 'react-router-dom';
```

```
function Navbar() {
  return (
    <nav>
      <Link to="/">Home</Link>
      <Link to="/about">About</Link>
    </nav>
  );
}
```

9. Implement a 404 page for undefined routes in React.

```
<Route path="*" element={<h1>404 Not Found</h1>} />
```

10. Write a simple React app demonstrating the use of useEffect to fetch and display placeholder data.

```
function DataFetcher() {
  const [data, setData] = useState([]);
```

```

useEffect(() => {
  fetch('https://jsonplaceholder.typicode.com/posts')
    .then(response => response.json())
    .then(data => setData(data));
}, []);

return (
  <ul>
    {data.slice(0, 5).map(item => <li key={item.id}>{item.title}</li>)}
  </ul>
);
}

```

11. Create a component using useRef to focus on an input field on button click.

```

function FocusInput() {
  const inputRef = useRef();

  return (
    <div>
      <input ref={inputRef} />
      <button onClick={() => inputRef.current.focus()}>Focus
    </button>
    </div>
  );
}

```

12. Implement React Context to pass a theme value between two components.

```

const ThemeContext = createContext();

function App() {
  return (
    <ThemeContext.Provider value="dark">
      <ChildComponent />
    </ThemeContext.Provider>
  );
}

function ChildComponent() {
  const theme = useContext(ThemeContext);
  return <p>Current theme: {theme}</p>;
}

```

13. Create a basic Angular component that displays "Welcome to Angular".

```

// app.component.ts
export class AppComponent {
  title = 'Welcome to Angular';
}

```

```
// app.component.html
<h1>{{ title }}</h1>
```

14. Use Angular interpolation to display the current date.

```
// app.component.ts
export class AppComponent {
  today = new Date();
}
```

```
// app.component.html
<p>{{ today }}</p>
```

15. Create an Angular template with a button that changes a text value when clicked.

```
// app.component.ts
export class AppComponent {
  message = 'Hello';
  changeMessage() {
    this.message = 'You clicked the button!';
  }
}

// app.component.html
<p>{{ message }}</p>
<button (click)="changeMessage()">Click Me</button>
```