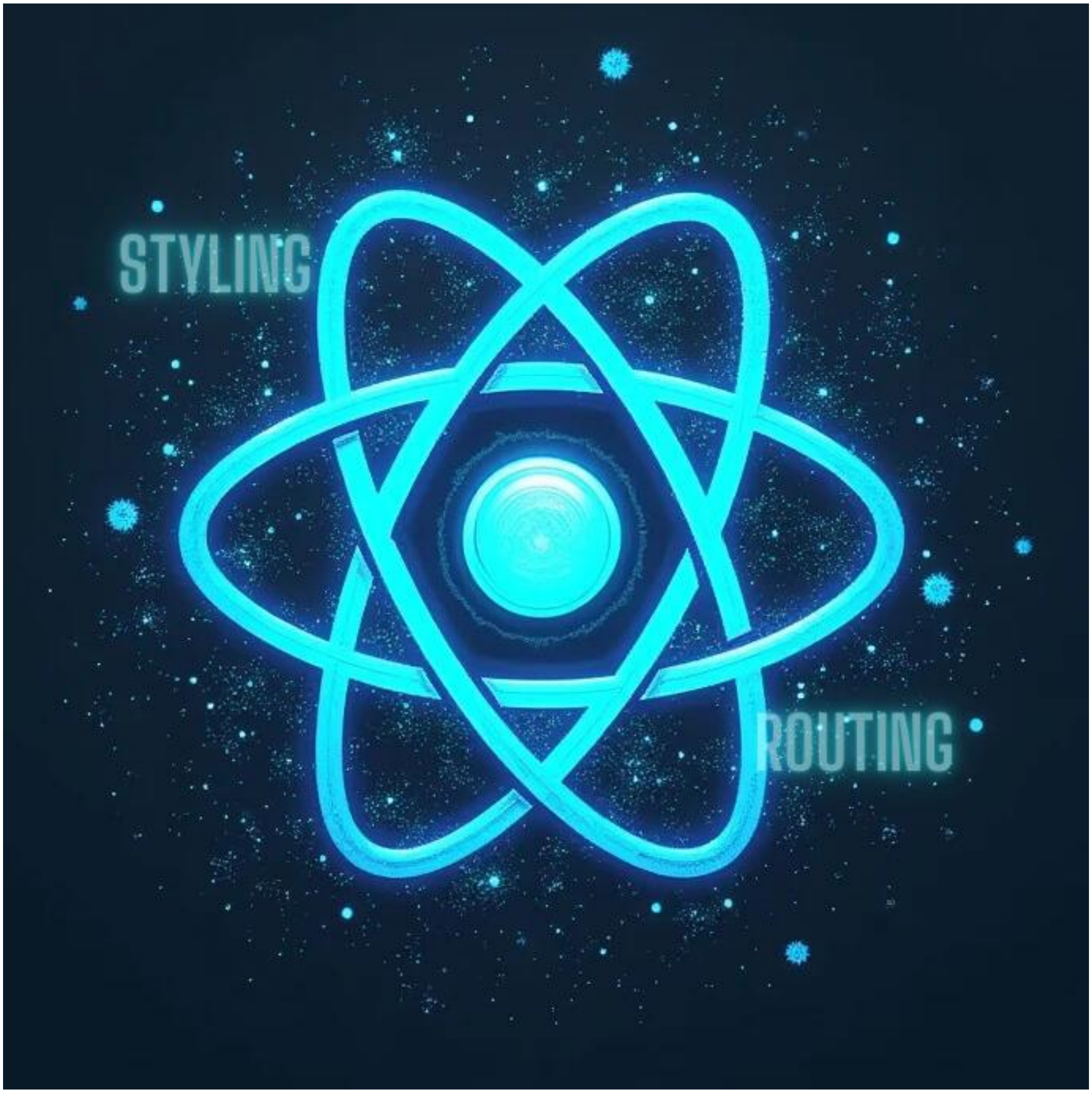


React – Advance

React – Styling, Routing

React-Styling

Question 1: How do you apply inline styles dynamically in React based on props or state?

Ans.

inline styles dynamically by using the **`style`** attribute and providing an object where the keys are camelCase versions of the CSS properties.

```
const style = {
    backgroundColor: isActive ? 'green' : 'red',
};
<div style={style}>Hello</div>;
```

Question 2 : Can you use both inline and external styles together in a React component?

Ans:

Yes, you can use both inline and external styles together in a React component.

```
import './App.css';

const style = { color: 'blue' };

<h1 className="title" style={style}>Hello</h1>;
```

React – routing

Question 1 : What is React Router? How does it handle routing in single-page applications?

Ans.

React Router is a library for managing navigation and routing in React applications. It allows you to define routes to map URLs to specific components, enabling seamless transitions between pages without reloading the browser.

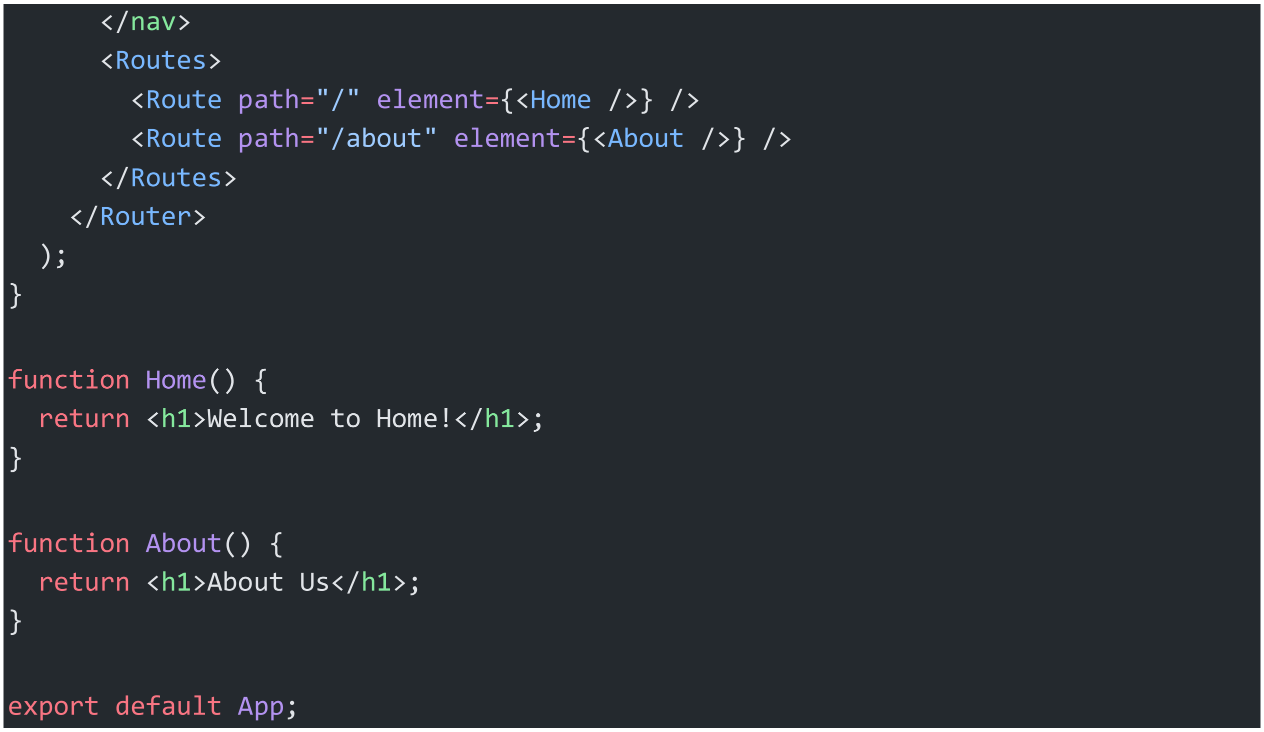
Handle routing in single-page applications

In single-page applications (SPAs), React Router handles routing by using the **History API** and **JavaScript** to dynamically update the URL and render the corresponding component without reloading the page.

Example:

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

function App() {
  return (
    <Router>
      <nav>
        <Link to="/">Home</Link>
        <Link to="/about">About</Link>
```



```
      </nav>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
      </Routes>
    </Router>
  );
}

function Home() {
  return <h1>Welcome to Home!</h1>;
}

function About() {
  return <h1>About Us</h1>;
}

export default App;
```

Question 1 : Explain the difference between BrowserRouter, Route, Link, and Switch components in React Router.

Ans.

1. **BrowserRouter**
    - Purpose: Wraps the entire app to enable React Router functionality..
    - Role: Manages the history stack and listens for changes in the URL.

Example:

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

function App() {
  return (
    <BrowserRouter>
      {/* Your routes go here */}
    </BrowserRouter>
  );
}
```

2. Route

- **Purpose**: Defines the mapping between a URL path and a component.
- **Role**: Renders a specific component when the current URL matches the `path` prop.

Example:

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

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

3. Link

- **Purpose**: Provides navigation between routes without reloading the page.
- **Role**: Generates clickable links that update the URL and render the corresponding component.

Example:

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

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

4. Switch (React Router v5)

- Purpose: Renders the first <Route> that matches the URl.
- Role: Ensures only one route is rendered at a time.

Example:

```
import { Switch, Route } from 'react-router-dom';

<Switch>
  <Route path="/" exact component={Home} />
  <Route path="/about" component={About} />
</Switch>
```

Note: In React Router v6, `Switch` is replaced by `Routes`.