**React-Advance**

* What is react router? how does it handle routing in single-page applications?

Reactjs router is mainly used for developing single page web application. React router is used to define multiple routes in the application. When a user types a specific URL into the browser, and if this URL path matches any ‘route’ inside the router file. The user will be redirected to that particular route.

Hare are some ways react router handles routing in application

* **Dynamic routes**

React router maps URL to components, allowing for dynamic routes that provide a seamless user experience.

* **Declarative approach**

React router uses JSX syntax to define routes and their corresponding components.

* **Nested routing**

React router support nested routing. Which allows for dynamic and hierarchical navigation structures.

* **Browser history**

React router allows users to use browser history features while preserving the application view.

* **link component**

the <link> component creates clickable links that navigate to specified routes.

* Explain the difference between browsrRouter, route, link, and switch components in react router.

In react router, the main differences between the browser router, route, link, and switch components are:

* **browserRouter**

the parent component that store all other route components.it uses the html5 history API to keep the UI in sync with the URL.

* **Route**

A child component that renders a specific UI component when the URL matches the specified path.

* **Link**

Allows a user to navigate to another page by clicking on it. It prevents the default request for the server to render a new HTML page.

* **Switch**

Renders the first route whose path matches the URL. It’s similar to the javascript switch statement.

* **HashRouter**

Uses the hash portion of the URL to keep the UI in sync with the URL. The harsh is never sent to the server, so no special server configuration is needed.

* **Outlet**

Renders the current route selected. It’s part of the layout component.