**JAVASCRIPT Assignment**

**MODULE: 18 (REACT – Routing)**

* **Routing**

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

Ans. React Router is a library used in React applications to manage navigation and routing in a single-page application (SPA). It allows you to define and manage multiple routes within a React app, enabling users to navigate between different views or pages without causing a full page reload.

* Below is the simple example of React Router:

import React from 'react';

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

const Home = () => <h2>Home Page</h2>;

const About = () => <h2>About Page</h2>;

const Contact = () => <h2>Contact Page</h2>;

const App = () => (

<Router>

<div>

<nav>

<Link to="/">Home</Link> |

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

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

</nav>

<Route path="/" exact component={Home} />

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

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

</div>

</Router>

);

export default App;

**How it works**:

* The <Router> component wraps the app to enable routing functionality.
* The <Link> components allow navigation without reloading the page.
* The <Route> components render different components based on the URL path (/, /about, /contact).

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

Ans. In React Router, the BrowserRouter, Route, Link, and Switch components are essential for handling routing and navigation in a React application. Here's an explanation of each component and how they differ:

* 1. **BrowserRouter**
* This component is used to wrap your entire application and enable routing using the HTML5 History API. It listens for changes in the URL and allows React Router to dynamically update the UI without a full page reload. It is placed at the top level of your component tree to provide routing functionality across the entire app.
* Example:

import { BrowserRouter as Router } from 'react-router-dom';

const App = () => (

<Router>

{/\* All your routing-related components go here \*/}

</Router>

);

* 1. **Route**
* The Route component is used to define a path and specify the component that should be rendered when that path is matched in the URL. A Route listens for changes in the URL and renders the specified component if the URL matches the path provided. We define a Route for each page or section of our app that we want to be reachable via a URL.
* Example:

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

const App = () => (

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

);

* 1. **Link**
* The Link component is used for navigation. It allows users to navigate between different routes in your app without causing a full page reload. It works like an anchor (<a>) tag, but instead of reloading the page, it updates the URL and renders the corresponding route component. It is used to create clickable elements (such as buttons, links, etc.) to navigate to different parts of the app.
* Example:

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

const Navigation = () => (

<nav>

<Link to="/">Home</Link>

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

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

</nav>

);

* 1. **Switch**
* The Switch component is used to group multiple Route components and ensure that only the first matching route gets rendered. When the URL matches a path, Switch renders the first Route that matches the URL, and no other Route will be rendered, even if there are multiple matching routes. We wrap routes inside a Switch when we want to ensure that only one route is rendered at a time.
* Example:

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

const App = () => (

<Switch>

<Route path="/" exact component={Home} />

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

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

</Switch>

);