Navigation

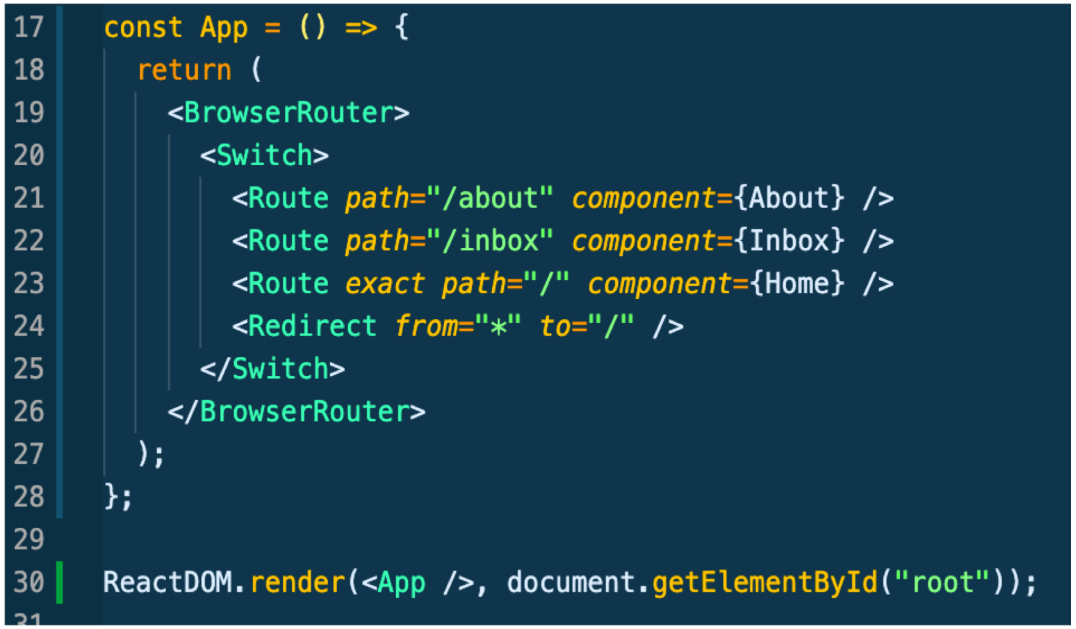
The React Router

Introduction

- **Allows** multiple views **and** flows **in an app.**
- **Keeps the URL in sync with the UI.**
- **Supports traditional web principles:**
  1. Addressability.
  2. Information sharing.
  3. Deep linking.
  – **1st generation AJAX apps violated these principles.**
- **A separate library.**

Basic routing configuration

	URL	Components
1	/	Home
2	/about	About
3	/inbox	Inbox



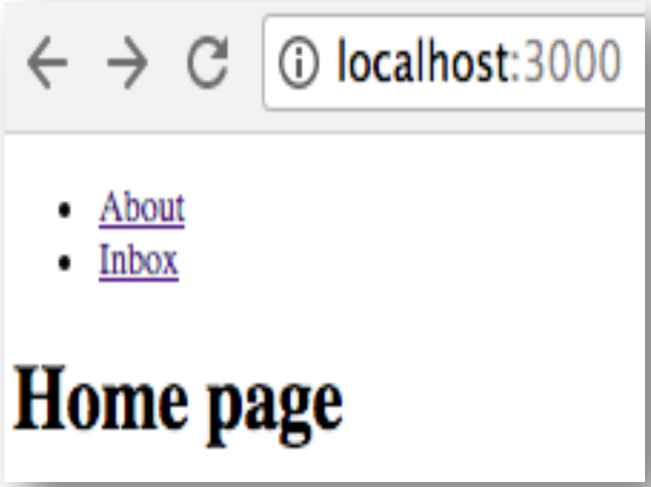
```
const App = () => {
  return (
    <BrowserRouter>
      <Switch>
        <Route path="/about" component={About} />
        <Route path="/inbox" component={Inbox} />
        <Route exact path="/" component={Home} />
        <Redirect from="*" to="/" />
      </Switch>
    </BrowserRouter>
  );
};

ReactDOM.render(<App />, document.getElementById("root"));
```

- **Declarative routing.**
- <BrowserRouter**> match browser's URL with** <Route> paths**.**
- <Route> component **mounted when match occurs.**
- <Route> path **supports regular expression pattern matching.**
  - **Use** exact **argument for precision.**
- **Use** <Redirect> to **avoid 404-type error.**
- <Switch> - **only one of the nested Routs can be active.**
- ReactDOM.render**() passed an app's Router component**.
- **Ref.** src/sample1/

Hyperlinks

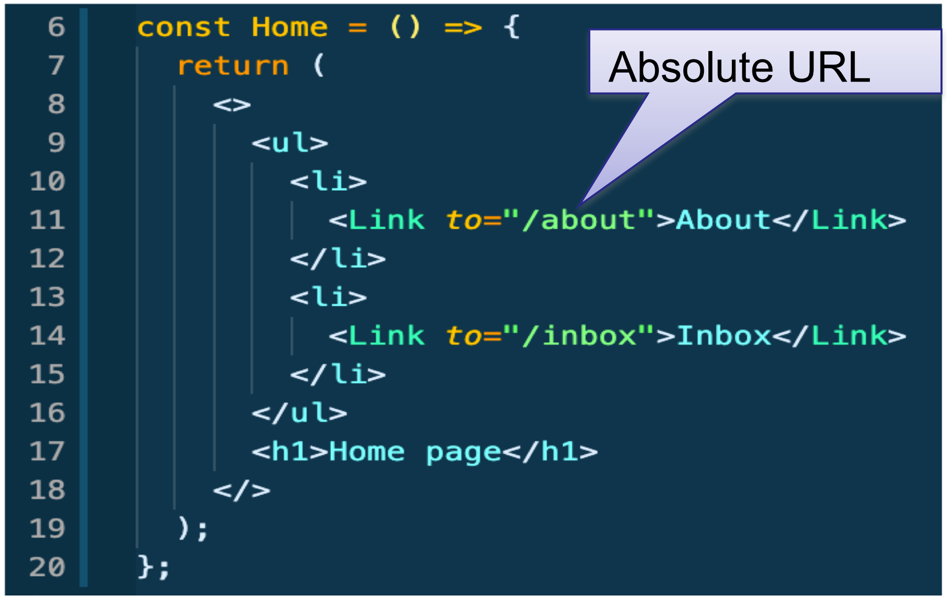
- **Use the** <Link> **component for internal links.**
  - **Use anchor tag for external links -** <a href . . . . . >
- **EX. Ref**. src/sample2/



localhost:3000

- About
- Inbox

Home page



```
const Home = () => {
  return (
    <>
      <ul>
        <li>
          <Link to="/about">About</Link>
        </li>
        <li>
          <Link to="/inbox">Inbox</Link>
        </li>
      </ul>
      <h1>Home page</h1>
    </>
  );
};
```

- <Link> **gives access to other useful router properties.**
- **Use** <LinlContainer> **when link wraps other 3rd party component, e.g. Bootstrap-React <Buttom />**

Dynamic segments.

- Parameterized URLs**, e.g. /users/***22***, /users/***12***/purchases**
  - **How do we declare a parameterized path in the routing configuration?**
  - **How does a custom component access the parameter value?**
- **Ex: Ref** src/sample3/**.**

  Suppose the Inbox **component shows messages for a specific user, where the user's id is part of the browser URL**

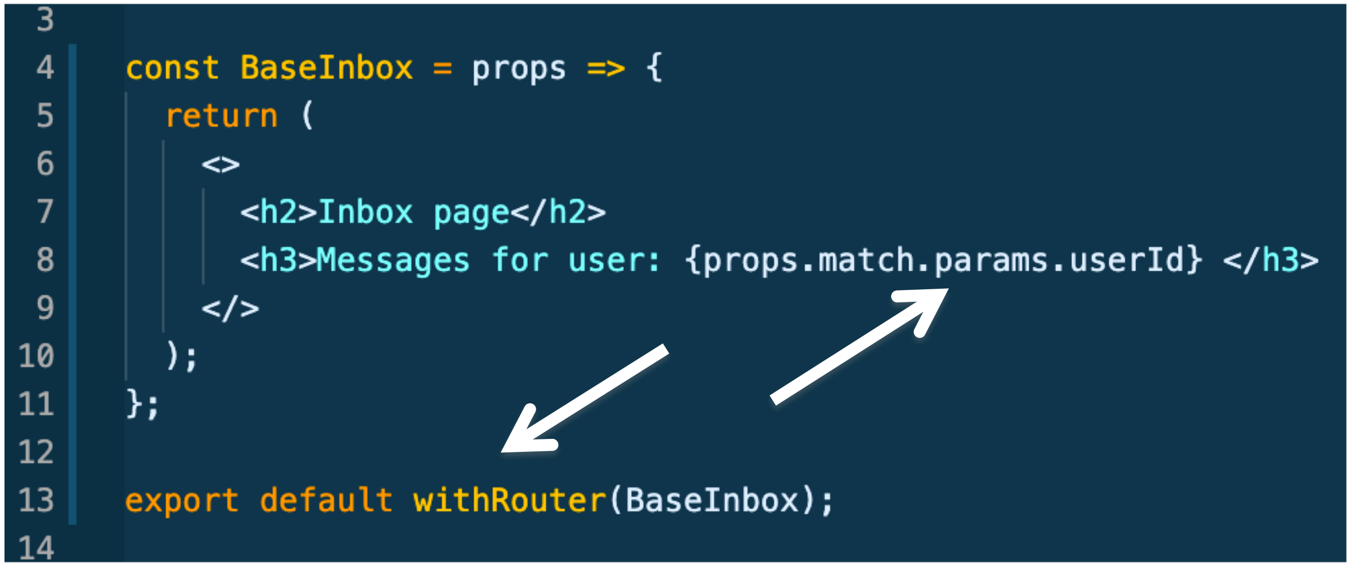
  e.g /inbox/123 where 123 is the user's id.

  ```
  <Route path='/inbox/:userId' component={ Inbox } />
  ```

  The colon (:) prefixes a parameter in the path.

  Parameter name is arbitrary.

Dynamic segments.



```
const BaseInbox = props => {
  return (
    <>
      <h2>Inbox page</h2>
      <h3>Messages for user: {props.match.params.userId} </h3>
    </>
  );
};

export default withRouter(BaseInbox);
```

- withRouter() **function:**
  - **Injects routing props into a component:**
    - props.match.params.(parameter-name)
    - props.history
  - **Returns a new, enriched component.**

Routing.

- **More routing features next week.**