reactjs/react-router-tutorial

Rendering a Route

At its heart, React Router is a component.

```
render(<Router/>, document.getElementById('app'))
```

That's not going to display anything until we configure a route.

Open up `index.js` and

1. import `Router`, `Route`, and `hashHistory`
2. render a `Router` instead of `App`

```
// ...
import { Router, Route, hashHistory } from 'react-router'

render((
  <Router history={hashHistory}>
    <Route path="/" component={App}/>
  </Router>
), document.getElementById('app'))
```

Make sure your server is running with `npm start` and then visit http://localhost:8080

You should get the same screen as before, but this time with some junk in the URL. We're using `hashHistory`--it manages the routing history with the hash portion of the url. It's got that extra junk to shim some behavior the browser has natively when using real urls. We'll change this to use real urls later and lose the junk, but for now, this works great because it doesn't require any server-side configuration.

Adding More Screens

Create two new components at:

- `modules/About.js`
- `modules/Repos.js`

```
// modules/About.js
import React from 'react'

export default React.createClass({
  render() {
    return <div>About</div>
  }
})
```

```
// modules/Repos.js
import React from 'react'

export default React.createClass({
  render() {
    return <div>Repos</div>
  }
})
```

Now we can couple them to the app at their respective paths.

```
// insert into index.js
import About from './modules/About'
import Repos from './modules/Repos'

render((
  <Router history={hashHistory}>
    <Route path="/" component={App}/>
    {/* add the routes here */}
    <Route path="/repos" component={Repos}/>
    <Route path="/about" component={About}/>
  </Router>
), document.getElementById('app'))
```

Now visit http://localhost:8080/#/about and http://localhost:8080/#/repos