**Adding Admin Component to React Frontend**

We have already implemented admin capability in our backend. Now a user with the role of admin can update or delete any post or user.

Let's create a separate component for admin in our react frontend, where only a user with the role of admin will get access.

Admin will be able to perform update and delete actions on any post or user.

Create a folder called admin inside src folder. Then create Admin.js inside that admin folder

Let's begin by writing a simple markup with title and two columns. These columns will be used for showing a list of posts and users.

**/src/admin/Admin.js**

import React, { Component } from "react";

class Admin extends Component {

render() {

return (

<div>

<div className="jumbotron">

<h2>Admin Dashboard</h2>

<p className="lead">Welcome to React Frontend</p>

</div>

<div className="container-fluid">

<div className="row">

<div className="col-md-6">

<h2>Posts</h2>

<hr/>

</div>

<div className="col-md-6">

<h2>Users</h2>

<hr/>

</div>

</div>

</div>

</div>

);

}

}

export default Admin;

Now lets add this Admin component to MainRouter

**src/MainRouter.js**

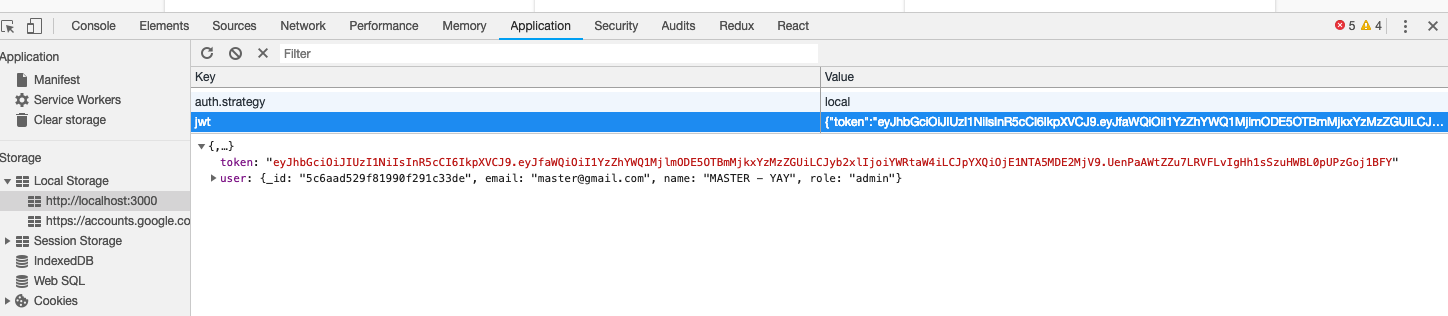
import Admin from './admin/Admin'

// then use inside MainRouter component

<PrivateRoute exact path="/admin" component={Admin} />

**Now login as an Admin user - A user who has the role of admin**

* Open your browser console. Click on Application tab. On your left sidebar
* Click on Local Storage
* Click on localhost:3000. On the right you should see a key with jwt
* Click on that to see the token along with user object
* This user object has role: "admin"
* Based on this role we will conditionally show/hide admin link in the Navbar



**Add an Admin link to /src/core/Menu.js**

This should be visible only if the user is logged in and that user has the role of admin

**src/core/Menu.js**

{isAuthenticated() && isAuthenticated().user.role === "admin" && (

<li className="nav-item">

<Link

to={`/admin`}

style={isActive(history, `/admin`)}

className="nav-link"

>

Admin

</Link>

</li>

)}

Now If admin logs in, he will see the admin link. Others won't!

Any logged in user can still access '/admin' if they type in the url manually

We will implement redirect to other page for non-admin later.

**/admin/Admin.js**

We already have posts component which we are using in Home component to show all posts

Lets re-use this Posts component in Admin.js

import Posts from "../post/Posts";

// show all posts

<div className="col-md-6">

<h2>Posts</h2>

<hr />

<Posts />

</div>

Now you see all posts if you visit '/admin'

Lets do the same with users

**/admin/Admin.js**

import Users from "../user/Users";

// show all users

<div className="col-md-6">

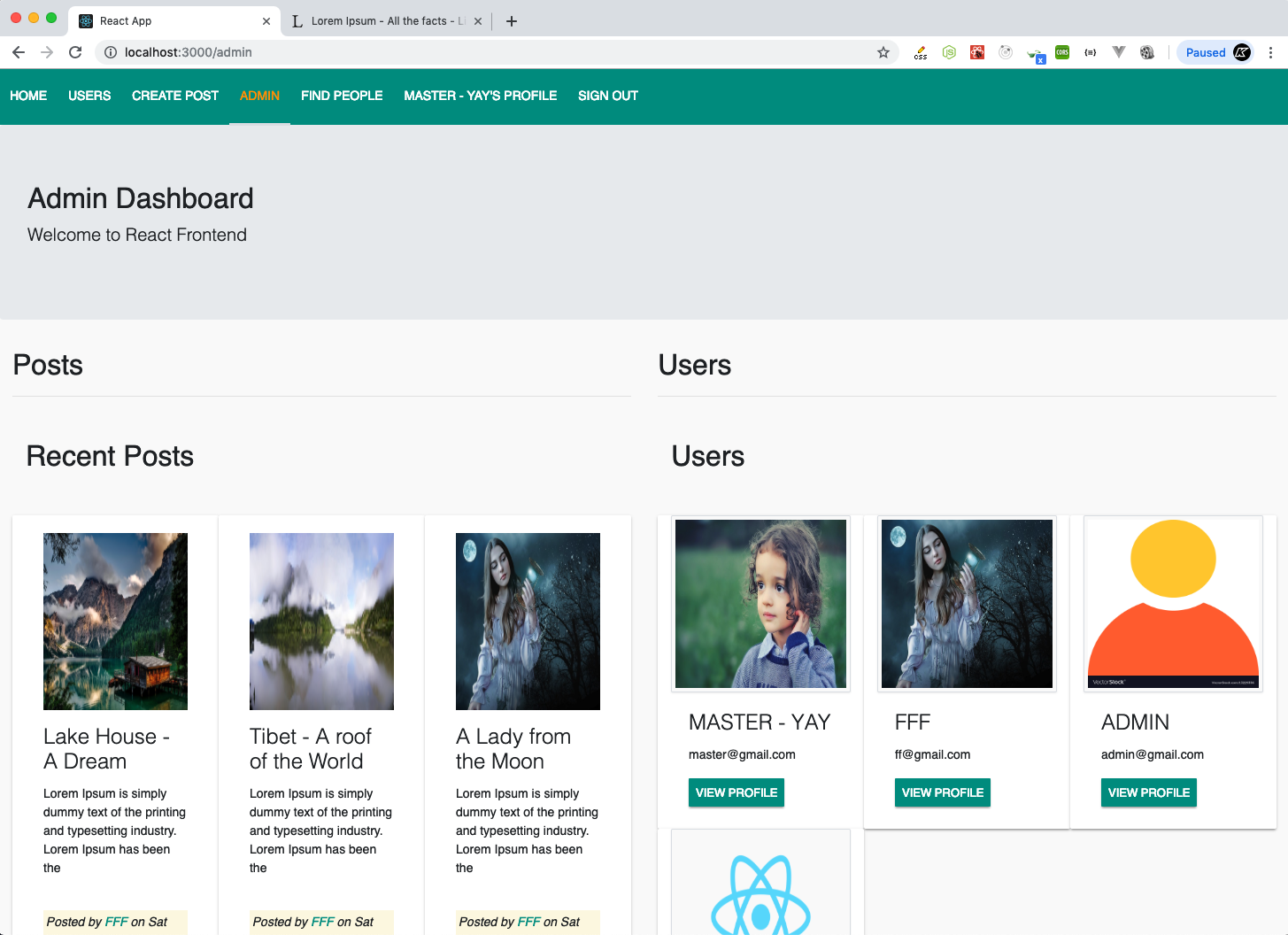
<h2>Users</h2>

<hr />

<Users />

</div>

* Now you see all users and posts in '/admin'
* You can click on Read More button to see entire post
* You can click on View Profile to see entire user profile



At the moment, an Admin cant update or delete other user or their posts. Let's implement that in the next lecture.