

# 10 - Homework (React API)

In this homework you need to finish the Hands-on task from the 10th lecture and modify Header links to render depending on logged user role (isAdmin true/false). Except that, you need to create a protected route for the Admin page and redirect users to the homepage if their **isAdmin** property is **false**.

**Homework must be finished till 14.5.2021. (23:59h).**

Please take a look at the final example where I've explained everything that is important for this homework ([final example](#)).

## Task description

In the last hands-on we have integrated registration and login of users. As you know, we have stored JWT token and isAdmin value of logged user into localStorage. Values in localStorage will be used only for the situations when we close our browser and open it again. In that case localStorage keeps stored values, and after we open our browser (App) again, we can read those values and within useEffect hook inside App.js we can set those values into components state variables, pass them with props and do what we need to do in order to re-render our Header component.

My advice is to start implementation from the App.js where you'll need to define two state variables: isLoggedIn, isAdmin. Then you need to find a way how you can pass values from the localStorage into those two variables.

After that you need to pass state variables to the Header component. Also, you'll need to pass isLoggedIn and isAdmin setters methods to the Login page (in that way you can set the value of those variables when login is successful, and those values will be propagated through all of the components that accept those state variables).

If you've implemented previous tasks, now you just need to go inside your Header component where you should do the following:

- Create a method that will be responsible for user logout, this method must remove JWT token and isAdmin value from localStorage. Except that within this method you'll need to change values of isAdmin and isLoggedIn state variables that come from the App.js.
- Inside the menu you'll need to create a menu item for logout (with the same styling as all other links in the menu - don't forget to do the same for the hamburger menu). When the user clicks on the logout menu item, the logout method should be called.
- If Admin is logged in then following items must be visible inside the desktop and hamburger menu: Home, Events, Admin, Logout.
- If Student is logged in then following items must be visible inside the desktop and hamburger menu: Home, Events, Logout.

If you did that then you can notice that you can access Admin page no matter you are Student or Admin. Just type in the browser 'localhost:3000/admin' and you'll be there. This is very bad because students can access the routes that need to be restricted.

With the help of the [following tutorial](#) you'll need to implement a Protected **route** for Admin page, and **Redirect** to the **Home page** if someone who is not allowed to see Admin page tries to access it directly.

Please note that in the provided tutorial they are using a new page for redirection. Instead of a new page you'll need to redirect to the Homepage!

## Next lecture preparation

In the next lecture we are going to learn what React Context is and how to use it. Except that we'll explain a little bit about the localStorage, sessionStorage and cookies.

Read the following documentation:

- [React Context documentation](#)
- [useContext hook](#)
- [localStorage](#)
- [sessionStorage](#)
- [cookies](#)