Frontend Authentication Exercises

How to auth

- protect the private pages
    - redirect to login
    - once logged in, redirect to the private page user was going to
- also, all private APIs should be protected

ex1

Create a private page (eg: an address page). Now show the Address page only if a user is logged (if the `login` variable is true - no need to add any state) else redirect to the login page.

challenge

https://codesandbox.io/s/react-router-navlink-solution-o9k0ew

solution

https://codesandbox.io/s/react-router-navlink-login-solution-forked-ixls8h

understanding

In order for authentication to work, when someone tries to access a protected page, they will be redirected to a login page. You can only access private pages after the key has been extracted & is set to `true`.

ex2

challenge

Using state create a login button and accordingly show the address page and login page.

challenge

https://codesandbox.io/s/react-router-navlink-login-solution-forked-ixls8h

solution

https://codesandbox.io/s/react-router-auth-02-loxjdu

understanding

To summarize, you set the login using a `useState`. This is all about authentication. From here we will just refactor our code to make it better.

ex3

Create a `RequiresAuth` component and show the element based on login status

challenge

https://codesandbox.io/s/frontend-auth-challenge-ex-03-forked-p6k961

solution

https://codesandbox.io/s/frontend-auth-solution-ex-03-forked-6qwfhb

understanding

```
// RequiresAuth.js
export function RequiresAuth({ children, login }) {
  return login ? children : <Navigate to="/login" />;
}

// App.js
<Route
      path="/address"
          element={
            <RequiresAuth login={login}>
```

```
              <Address />
            </RequiresAuth>
          }
        />

// Routing understanding
if you are not logged in
/address -> /login -> /address
```

COPY

ex4

Convert useState to useContext and check if a user is logged in or not.

challenge

https://codesandbox.io/s/frontend-auth-challenge-ex-04-forked-n0x91m

solution

https://codesandbox.io/s/frontend-auth-solution-ex-04-forked-lixe2w

ex5

Navigate the user to the page where he was going, after the successful login. User clicked on "address" and you redirected her to the "login" page so now it's your duty to redirect her to the "address" page.

challenge

https://codesandbox.io/s/frontend-auth-solution-ex-04-forked-lixe2w

solution

https://codesandbox.io/s/frontend-auth-redirection-ex05-solution-forked-hql6e2

understanding

```
// read location and supply in requireAuth
let location = useLocation();
<Navigate to='/login' state={{ from: location }} />

// use navigate in login to redirect after login
let navigate = useNavigate()
let from = location.state?.from?.pathname || '/'
navigate(from)
```