**Topics to Cover in Today’s Session**

1. CanActivate and CanDeactivate
2. Observable
3. HttpClientModule and HttpClient
4. Performing CRUD operation

Today we will look at how we can make sure the routes we create in Angular are accessed by the right people and that we prevent unauthorized access to routes that are private

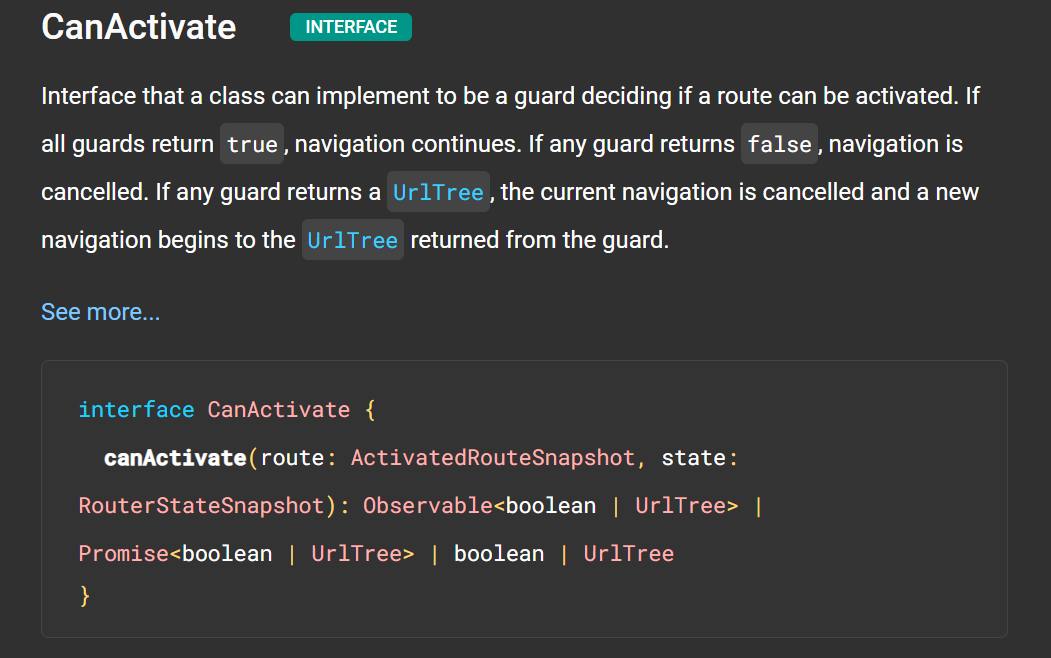
**What Are Route Guards?**

Angular route guards are interfaces provided by Angular which, when implemented, allow us to control the accessibility of a route based on conditions provided in class implementation of that interface.

Guard can return either a boolean or Observable<boolean> or Promise<boolean> which resolve to a boolean at same point of time.

Here are some types of Angular guards: CanActivate, CanActivateChild, CanLoad, CanDeactivate and Resolve.

CanActivate Route: 🡪 **Check whether user can visit.**



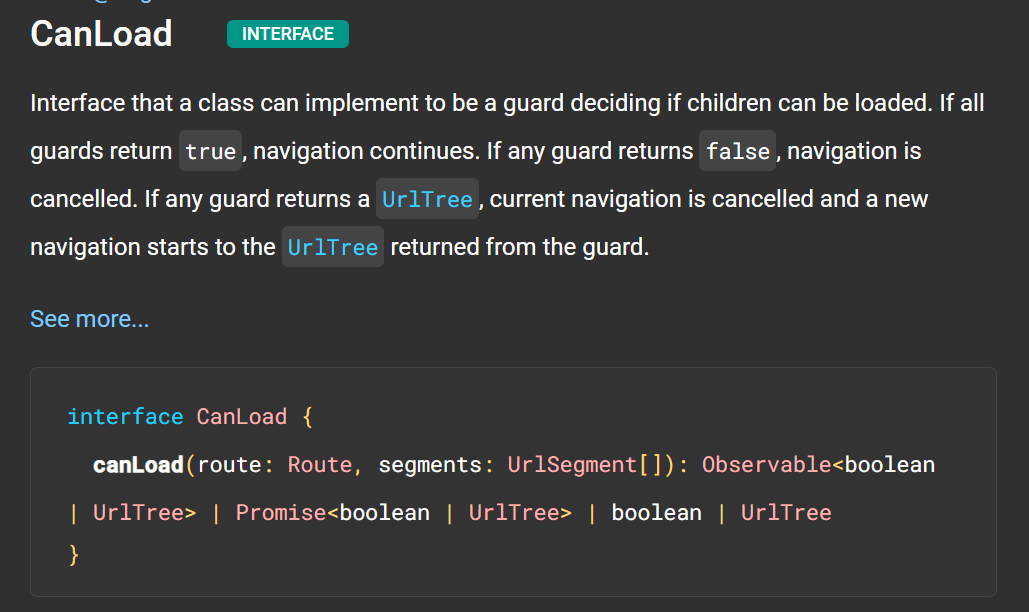
For detain Description visit: <https://angular.io/api/router/CanActivate>.

CanActivateChild:🡪 Check whether user can access children of routes.



For detail Description: <https://angular.io/api/router/CanActivateChild>

CanLoad: 🡪 Use to check whether user can access routes that lazy loaded



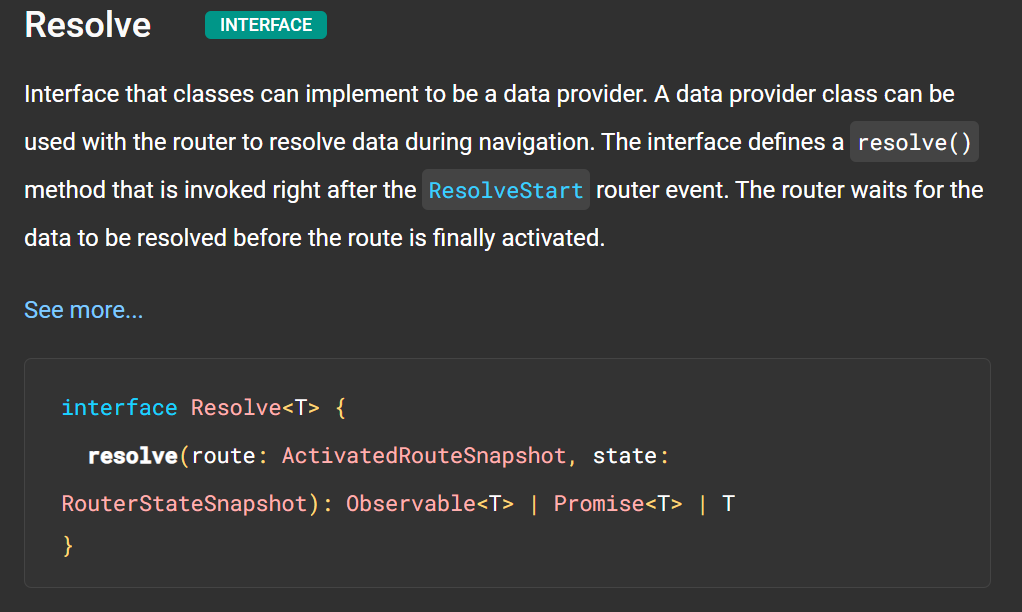
For detail description: <https://angular.io/api/router/CanLoad>

CanDeactivate:🡪 Checks whether user can exit a route



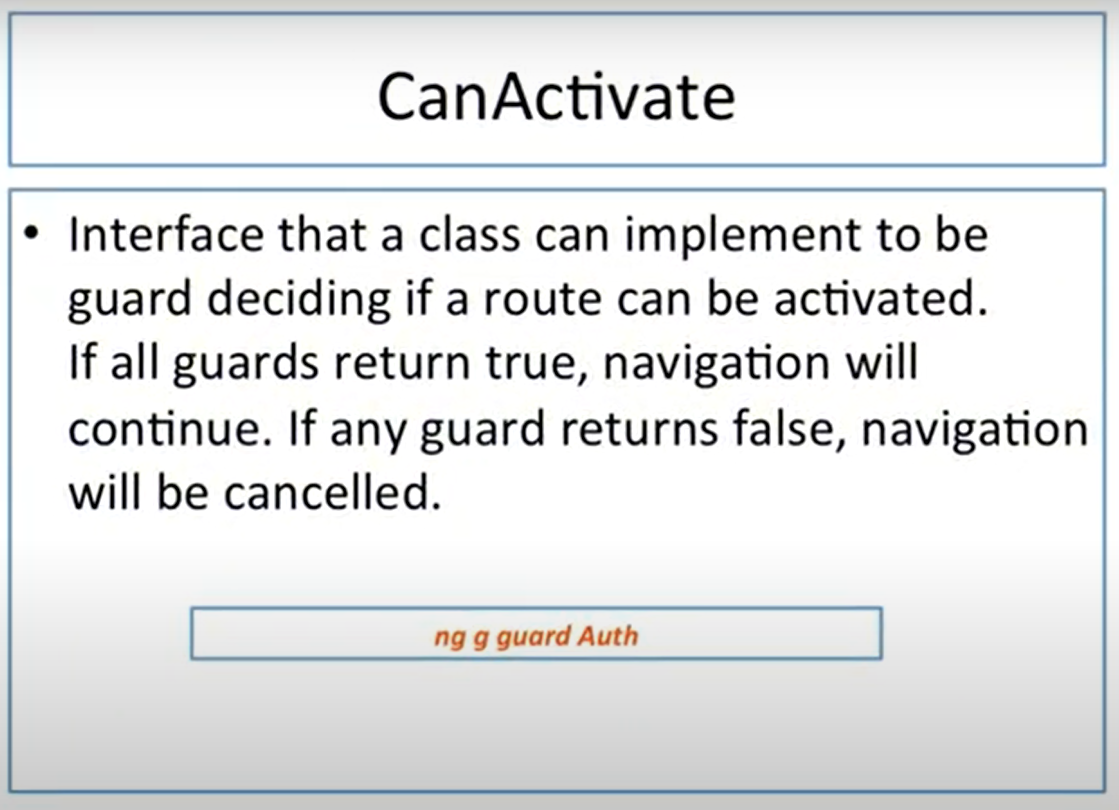
For detail description: <https://angular.io/api/router/CanDeactivate#description>

Resolve:🡪 Performs route data retrieval before route activation.

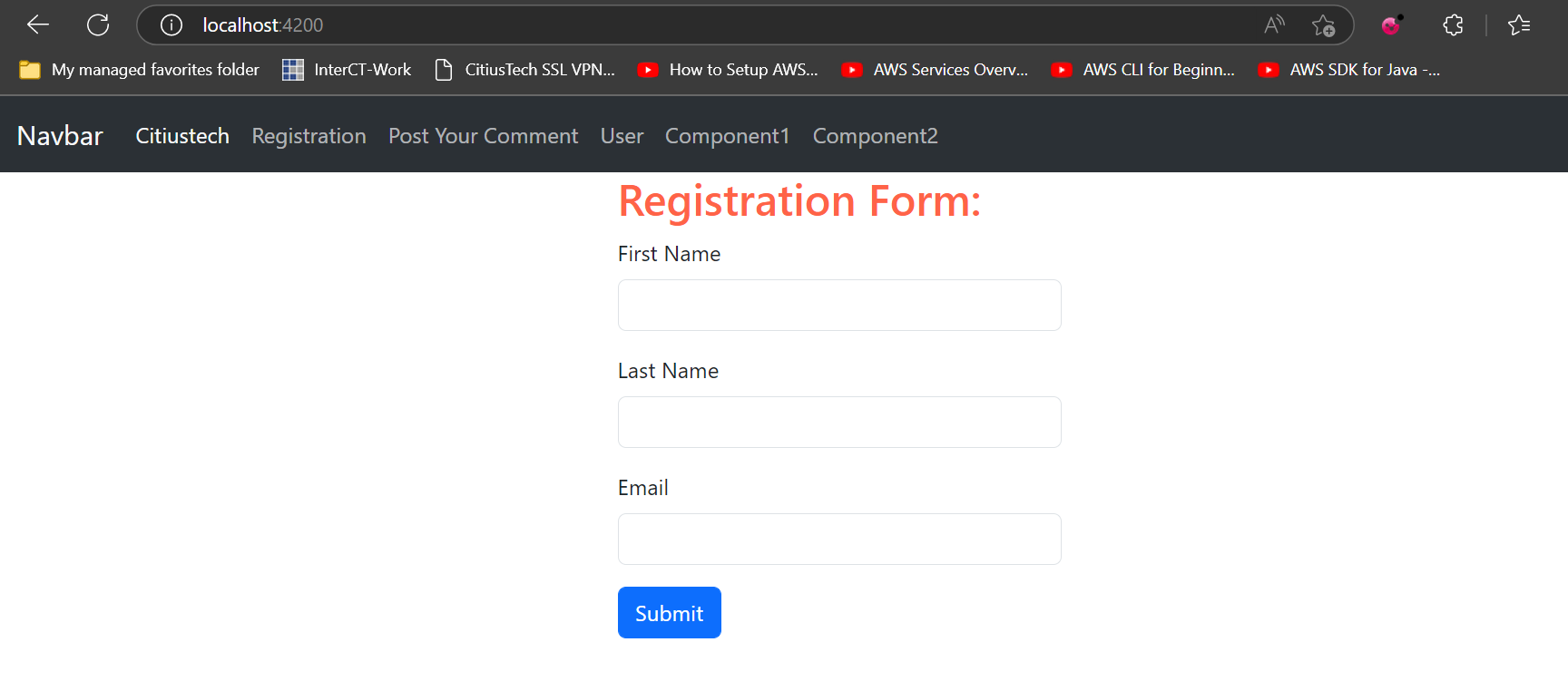


For detail description: <https://angular.io/api/router/Resolve>

CanActivate:



Here we will use same project which we used in last project.



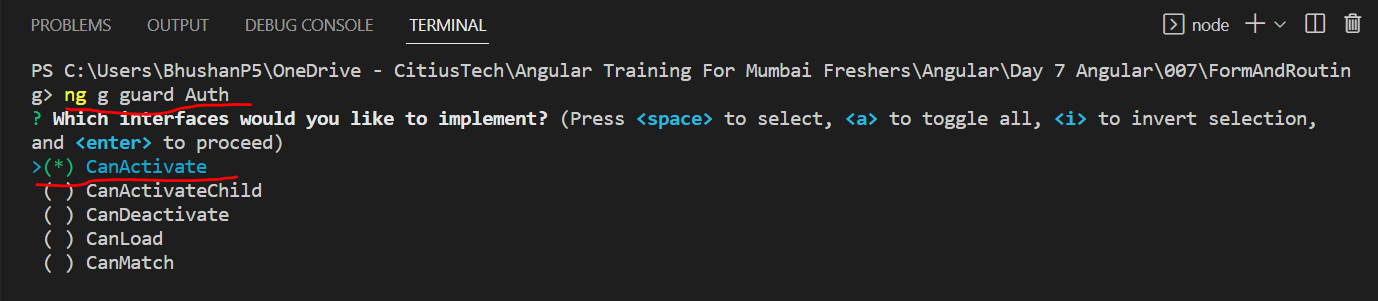
Our goal in this demo is, we can access User component directly but now we will have to provide restriction. Only valid users can only access this component.

So, to achieve our goal, we have to follow given below steps.

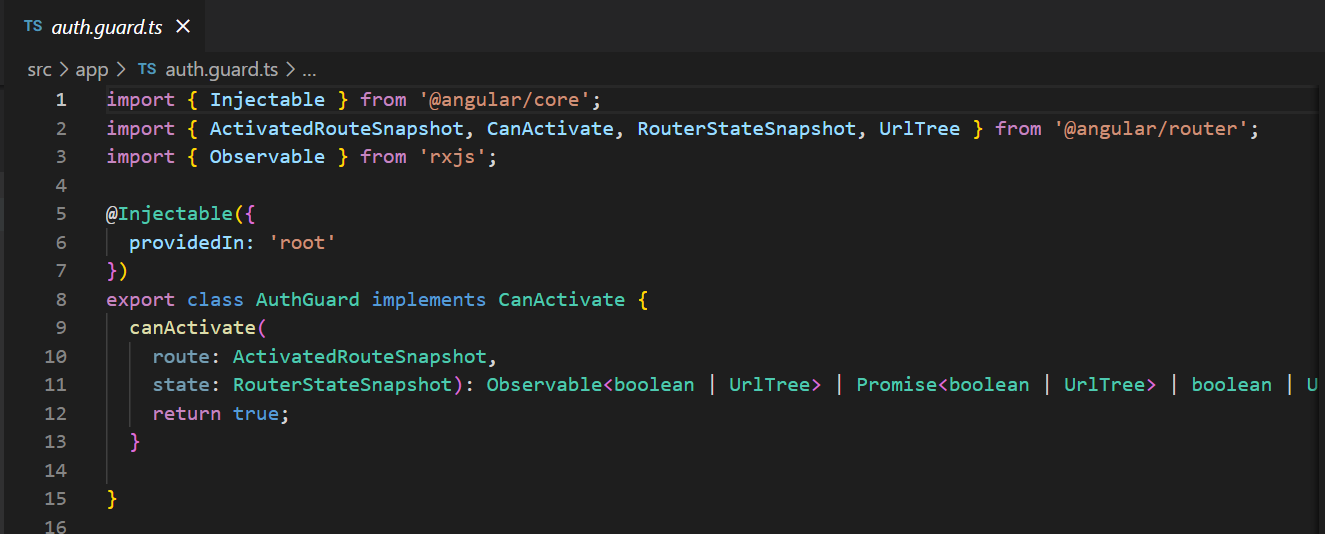
Switch back to VS Code and start implementing our code.

**Step 1:** Create guard for our application. To create guard fire command.

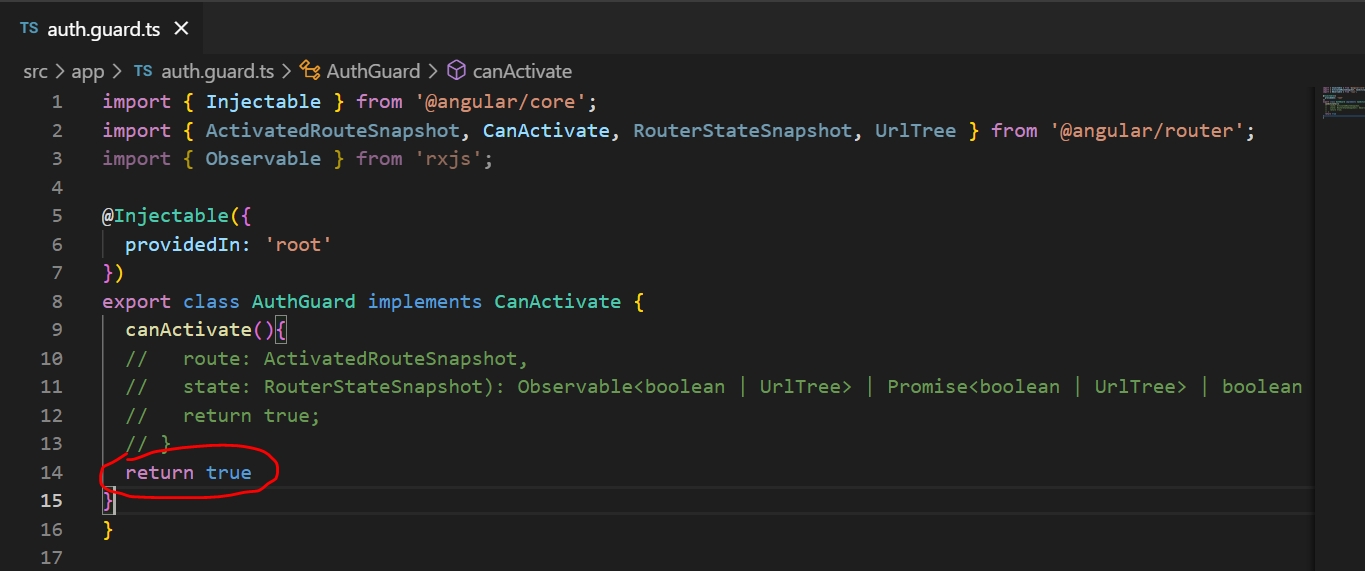
ng g guard Auth



It has created guard by the name Auth



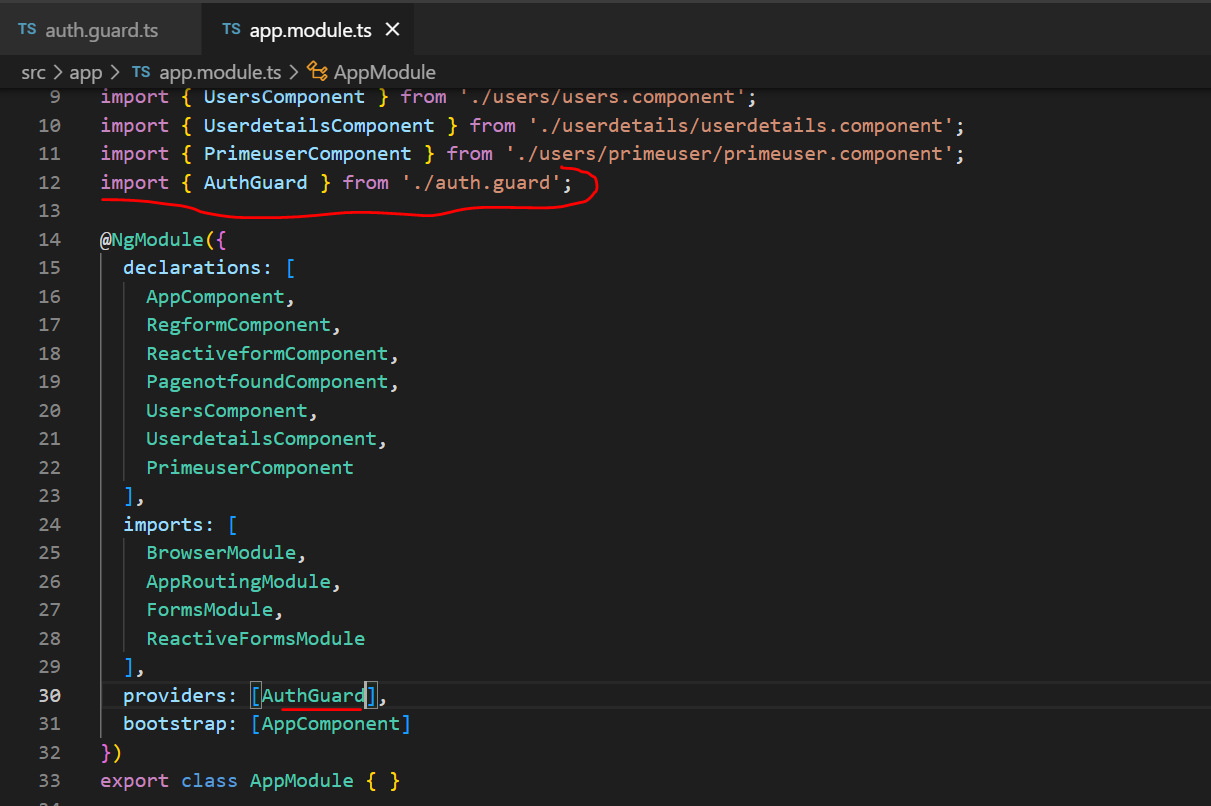
Do one simple change, we will return true statically.



Inside this method we have to define our logic on basis of which it may return true or false. For this purpose we will use service.

Now, move to our app-routing.module.ts file.

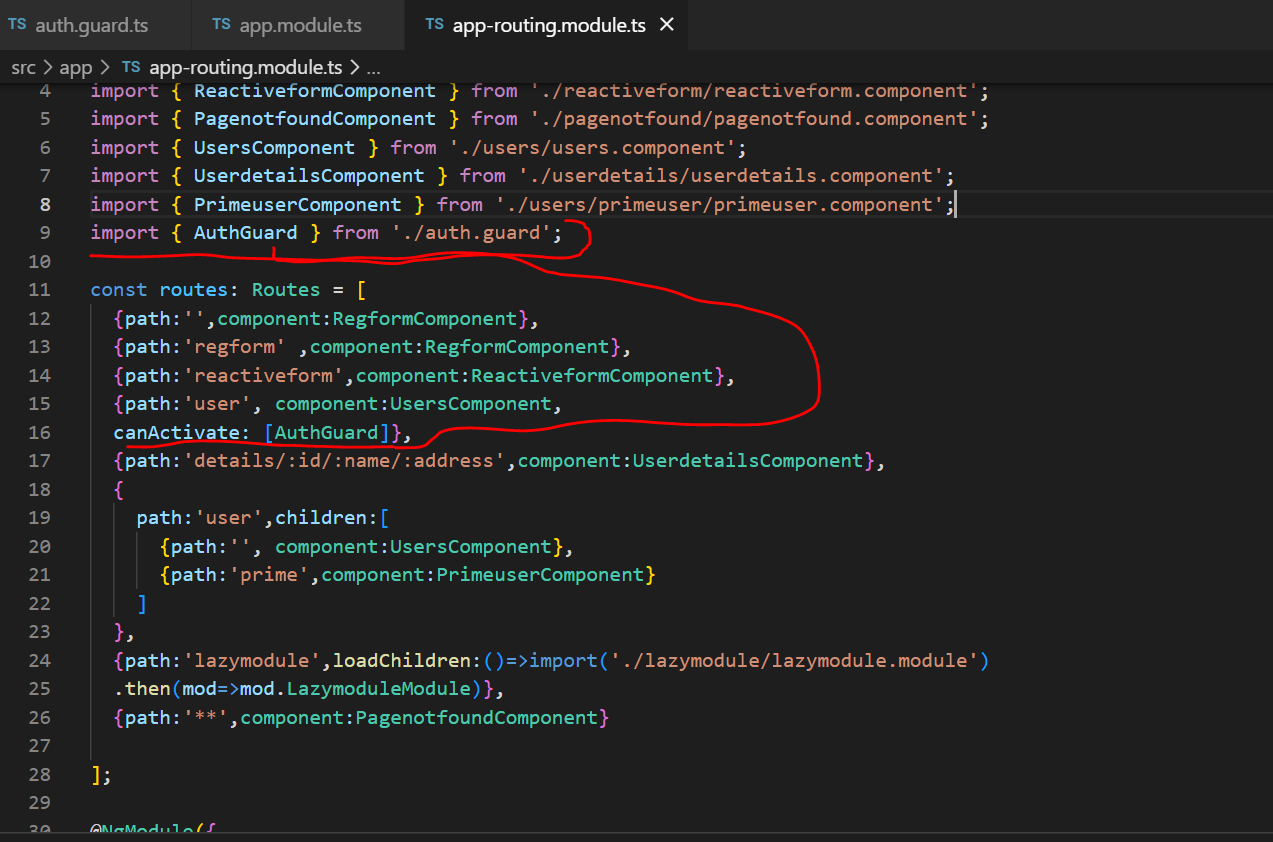
**Step 2:** In app-routing.module.ts file, select the route for which we will have to implement guard. So, in our case we will use user component. If we will open our auth file then it looks like service. It has @Injectable decorator so, like service we will have to import this guard in app.module.ts and add in providers array.



**Step 3:** Next step is, add this guard on route. So, for achieving this we will open app-routing.module.ts file. Here we have to protect our user route. So, whichever guard we have to activate for restriction we have to call our guard which we created.

So, here for user route I have added one more property and that is CanActivate. Which assign with array of Guard. Mean we can have multiple guards before activating this route.

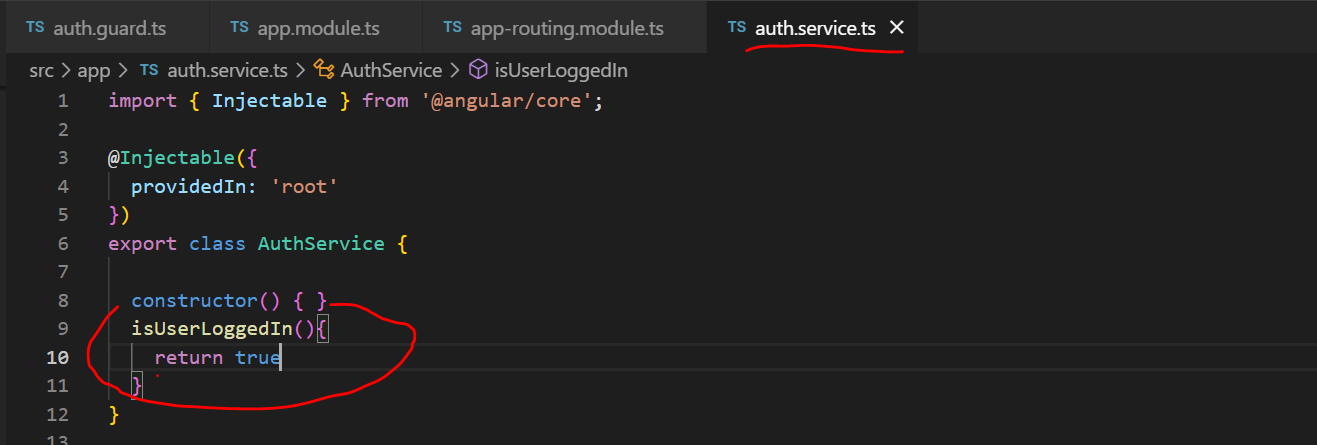
But here we have added only one guard so, we will use only one.



Now, we will use respective service file for creating function to check is user loggedIn or not.

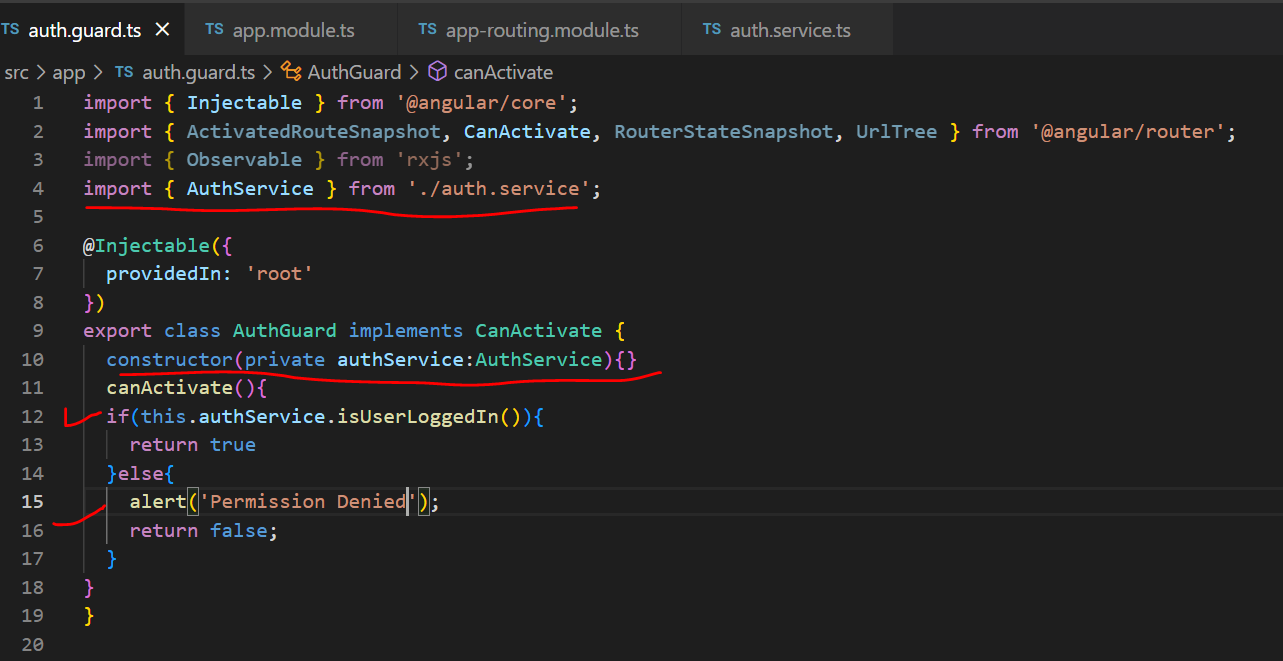
For this we will open guard.service.ts and add a function to checked Login successful or not?

**Step 4:** Open guard.service.ts file create a function to implement logic for login.



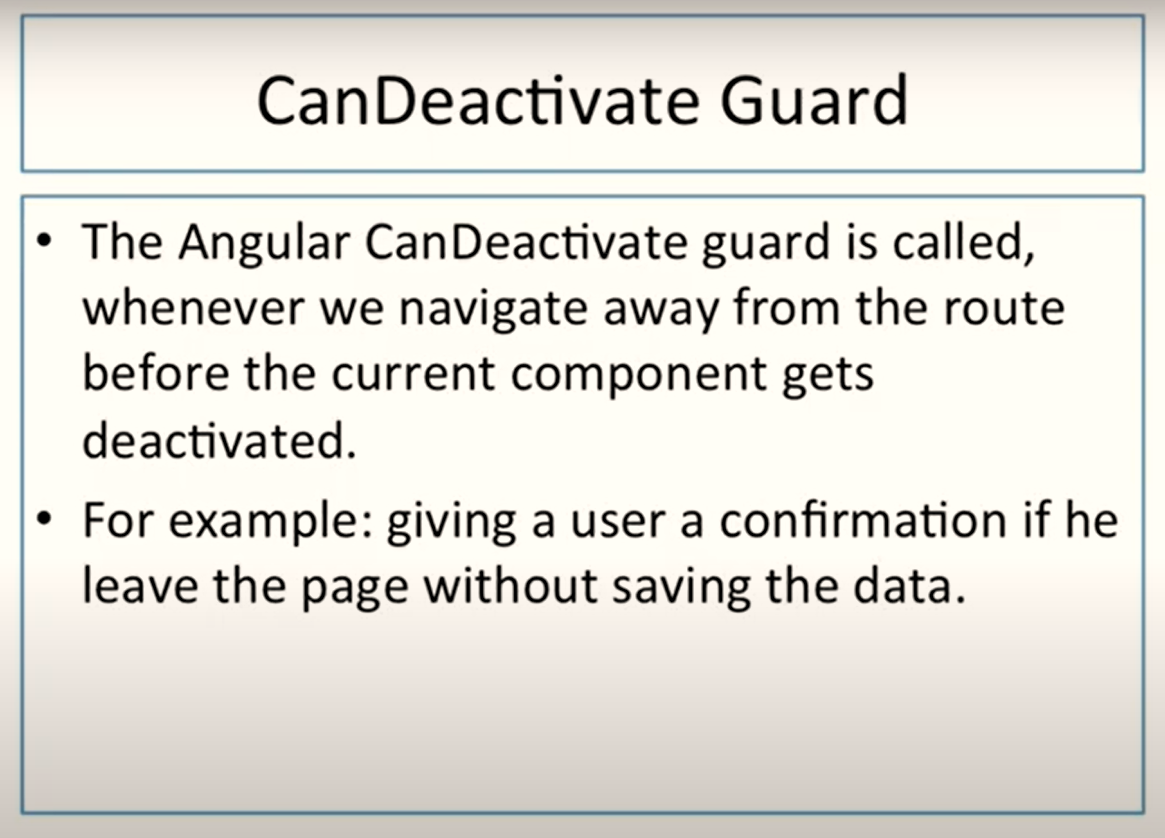
So, in this function we can specify whether user is authenticated or not.

**Step 5.**  Now, we will open auth.guard.ts file and inject service in constructor



If function in service will return true then this CanActivate allow user to access route and if it’s false then it restrict access.

CanDeactivate:



Assume we have a registration form or reactive form; I have filled up text boxes but before saving them I am clicking on other button so other route will be called. But I don’t want it should happen. I want when I will try to route to other page and I am filling the form then I should get notification.

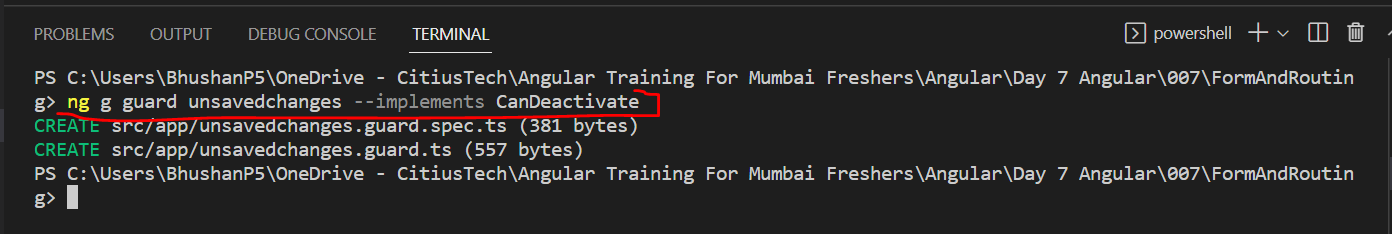
This behaviour we can achieve using **CanDeactivate.**

Here, I am using our previous project only and we will use canDeactivate on regForm.

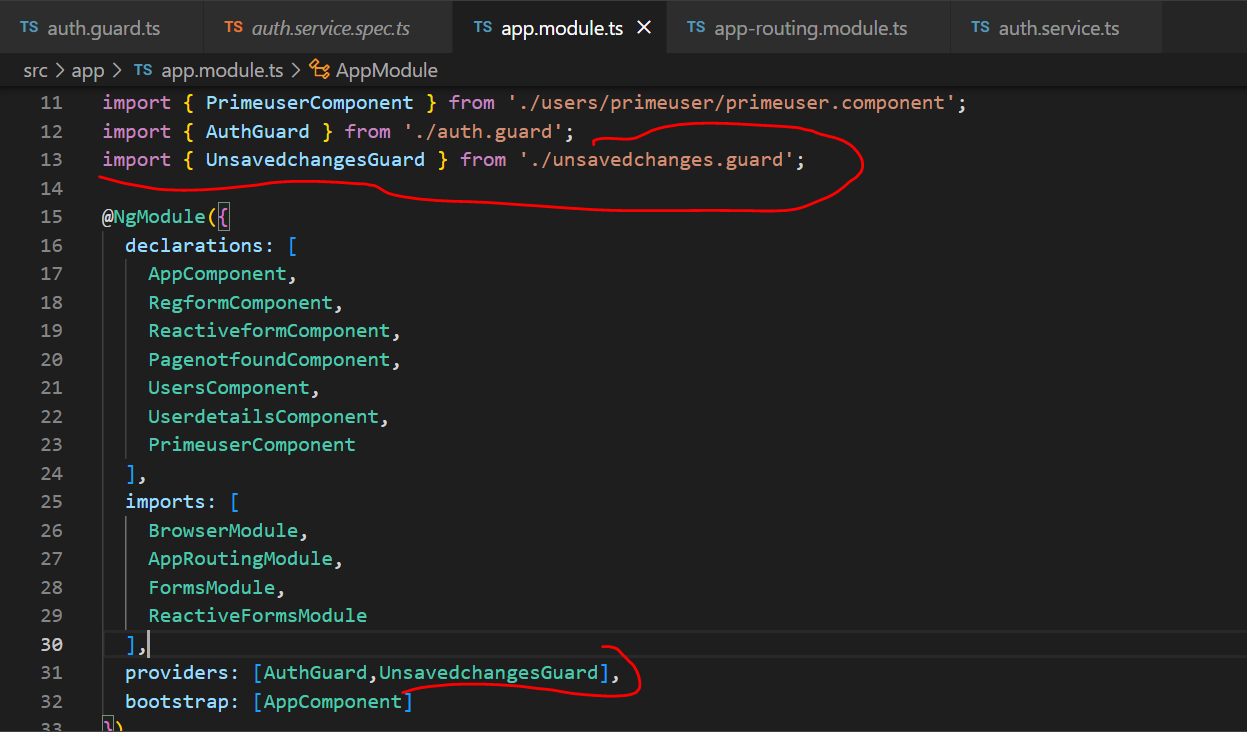
We will use Our previous project in which we have already created reactive form.

We have already created reactive form.

For this purpose we will add guard with implementation of canDeactivate

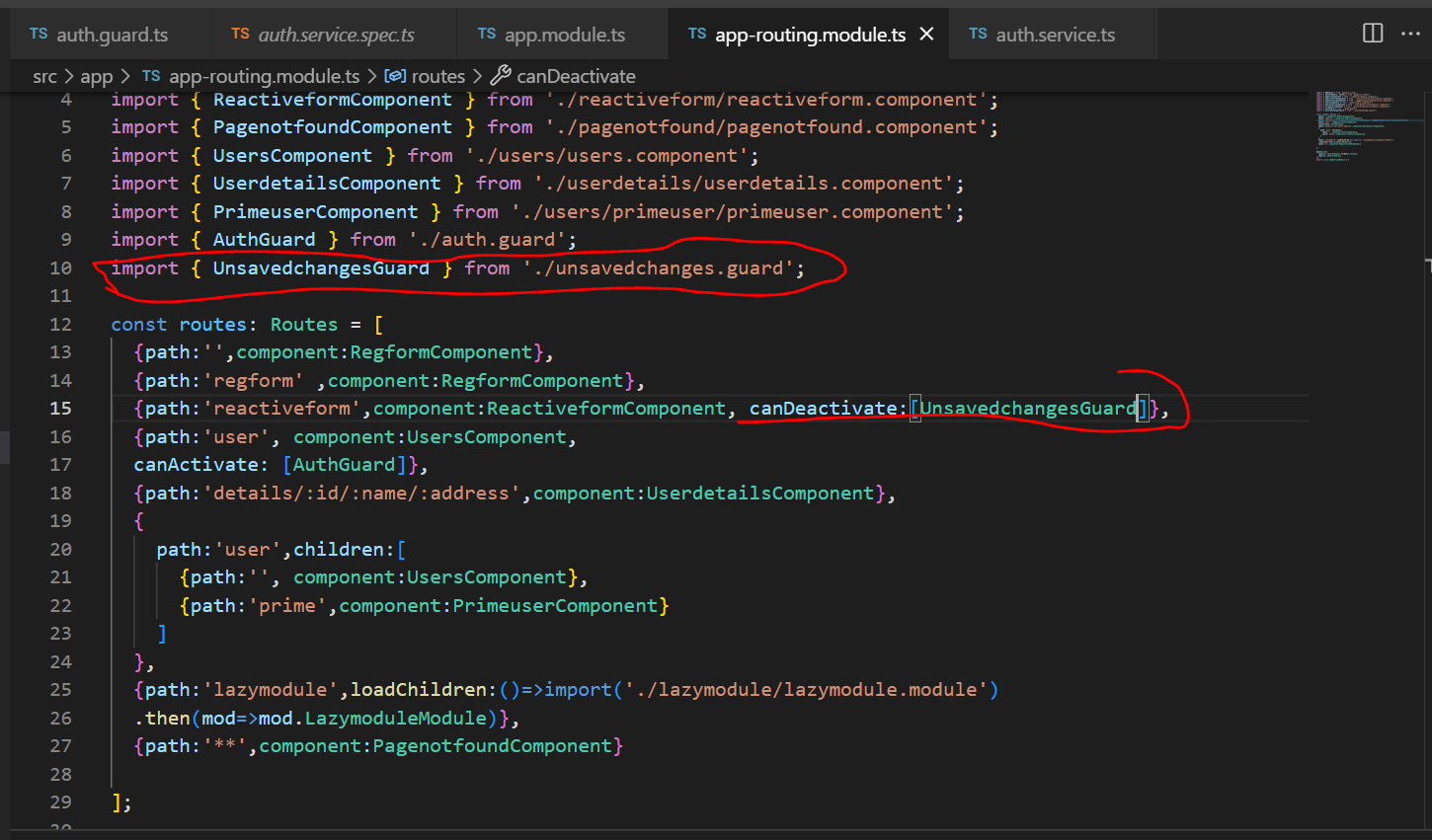


Now, import this guard in app.module.ts



Now, open app-routing.module.ts

And add canDeactivate route



Now, open unsavedchanges.guard.ts

