Steps for Route guard:

Step 1: ng new RouteGuard

Step 2:

Create two components login and dashboard.

Step 3: Create new guard using **ng g guard auth**

Step 4: create new service myservice.

Auth.guard.ts

import { elementEventFullName } from '@angular/compiler/src/view\_compiler/view\_compiler';

import { Injectable } from '@angular/core';

import { ActivatedRouteSnapshot, CanActivate, RouterStateSnapshot, UrlTree } from '@angular/router';

import { Observable } from 'rxjs';

import { Router } from '@angular/router';

@Injectable({

  providedIn: 'root'

})

export class AuthGuard implements CanActivate {

  constructor(private \_routes:Router)

  {

  }

  canActivate(

    route: ActivatedRouteSnapshot,

    state: RouterStateSnapshot): boolean

     {

      if(localStorage.getItem('username')!=null)

      {

      return true;

      }

    else

    {

      this.\_routes.navigate(['/login']);

      return false;

    }

  }

}

Myservice.ts

import { Injectable } from '@angular/core';

@Injectable({

  providedIn: 'root'

})

export class MyserviceService {

  constructor() { }

  checkusernameandpassword(uname:string,pwd:string)

  {

    if(uname=="admin" && pwd=="admin123")

    {

      localStorage.setItem('username','admin');

    return true;

    }

    else

    return false;

  }

}

App.routing.module.ts

import { dashCaseToCamelCase } from '@angular/compiler/src/util';

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import { AuthGuard } from './auth.guard';

import { DashboardComponent } from './dashboard/dashboard.component';

import { LoginComponent } from './login/login.component';

const routes: Routes = [

  {path:'login',component:LoginComponent},

  {path:'dashboard',canActivate:[AuthGuard] ,component:DashboardComponent}

];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

Login.component.ts

import { Component, OnInit } from '@angular/core';

import { MyserviceService } from '../myservice.service';

import { Router } from '@angular/router';

@Component({

  selector: 'app-login',

  templateUrl: './login.component.html',

  styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

msg='';

  constructor(private myservice:MyserviceService,private routes:Router) {

    localStorage.clear();

   }

  ngOnInit(): void {

  }

  check(uname:string,password:string)

  {

    var output=this.myservice.checkusernameandpassword(uname,password);

    if(output==true)

    this.routes.navigate(['/dashboard']);

    else{

      this.msg='invalid username or password';

    }

  }

}

Login.component.html

<p>login works!</p>

<div class="container">

    <div class="row">

    <div class="form-group">

      <label for="username">username</label>

      <input type="text" name="username" #username id="" class="form-control"

      placeholder="" aria-describedby="helpId">

    </div>

    <br>

    </div>

    <div class="form-group">

        <label for="password">password</label>

        <input type="password" name="password" #password id="password"

        class="form-control" placeholder="" aria-describedby="helpId">

      </div>

      <button (click)="check(username.value,password.value)"> Login</button>

</div>

<span>{{ msg }}</span>

App.compoent.html

<nav class="nav navbar" style="display: inline-flex; padding: 3PX;margin: 5px;">

  <a [routerLink]="['login']">Login</a>

  <a [routerLink]="['dashboard']">Dashboard</a>

  </nav>

<router-outlet></router-outlet>











