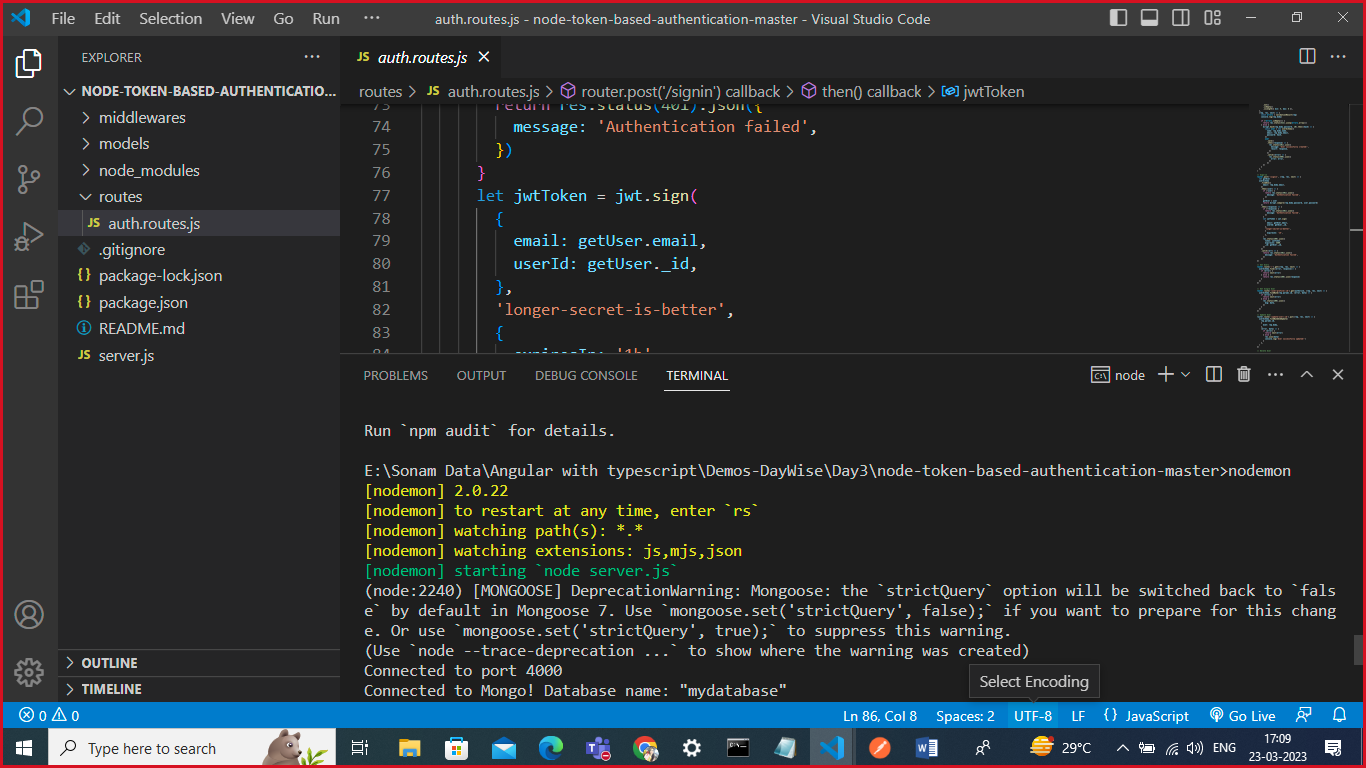
Let’s implement JWT Authentication

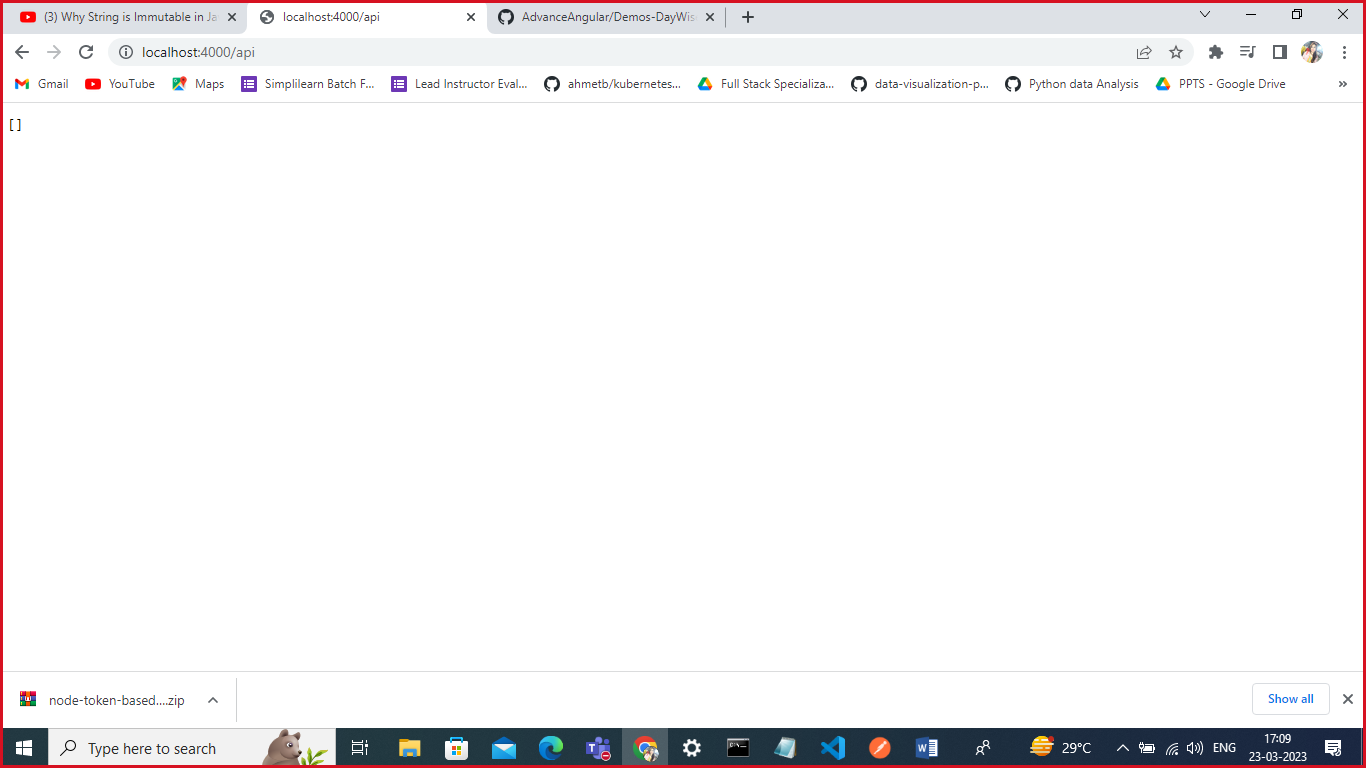
To set up a server you can down load the code: <https://github.com/sonam-niit/AdvanceAngular/tree/main/Demos-DayWise/Day3/node-token-based-authentication-master>

After downloading go to the folder and excute NPM install command

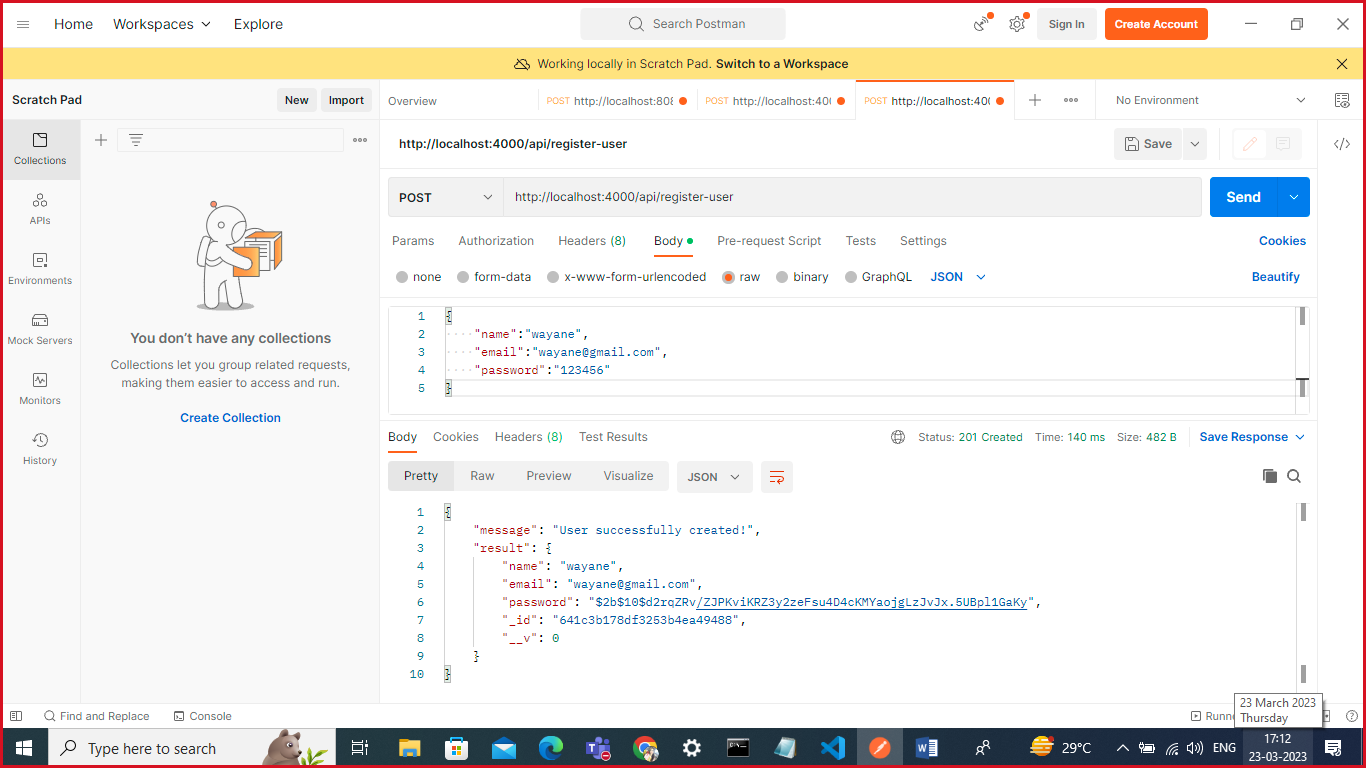
Execute nodemon to run server



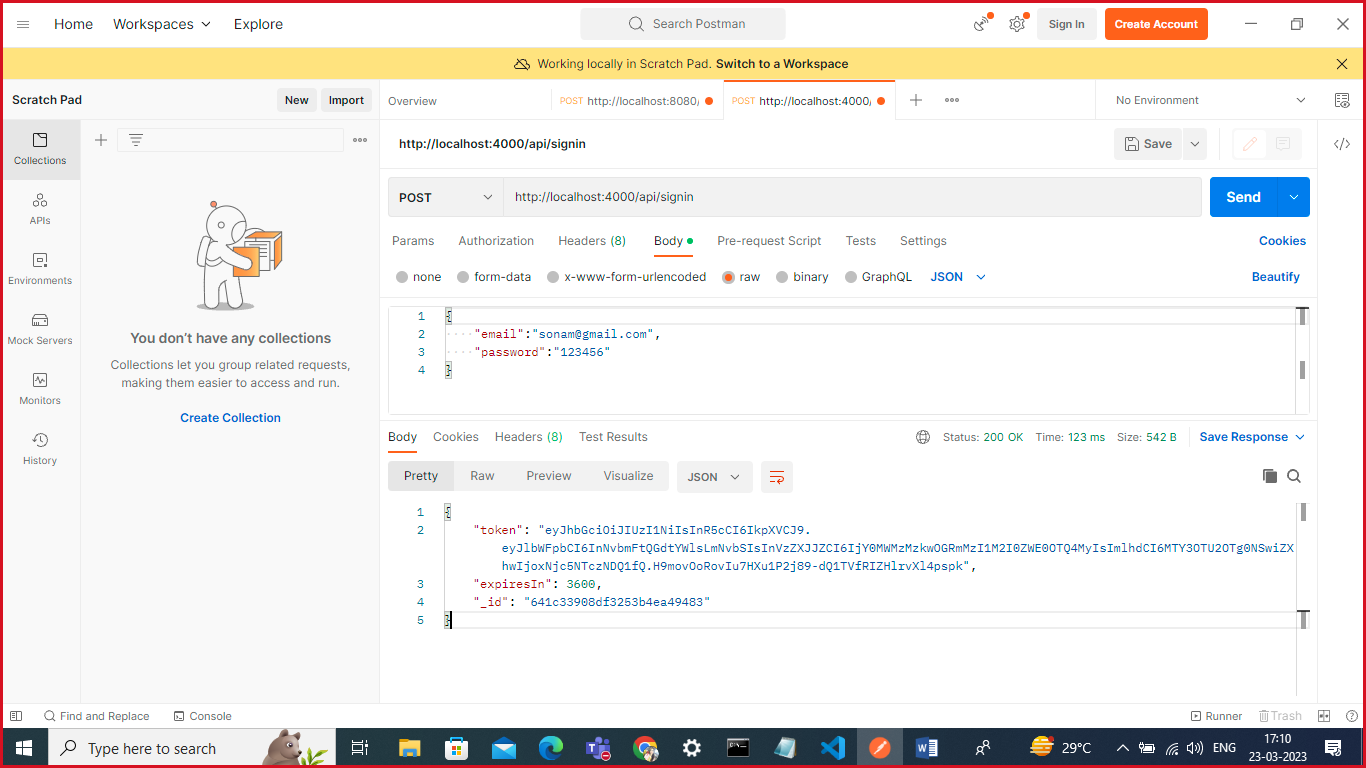
Check get request



Register



Signin request



You can see the generated token

Let’s create Angular Application JWT Auth.

Created 3 components

Ng g c jwt/signin –skip-tests

Ng g c jwt/signup –skip-tests

Ng g c jwt/user-profile –skip-tests

Created service: ng g s services/auth –skip-tests

Edit auth.service.ts

import { HttpClient, HttpHeaders } from '@angular/common/http';

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

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

import { Observable } from 'rxjs';

import { Userclass } from '../models/userclass';

@Injectable({

  providedIn: 'root'

})

export class AuthService {

  currentuser={}

  endpoint:string="http://localhost:4000/api";

  headers= new HttpHeaders().set('Content-Type','application/json');

  constructor(private http:HttpClient,private router:Router) { }

  //signin

  signIn(user:Userclass){

    return this.http.post(`${this.endpoint}/signin}`,user)

    .subscribe((res:any)=>{

      localStorage.setItem('access\_token',res.token);

      this.getUserProfile(res.\_id).subscribe((res)=>{

        this.currentuser=res;

      })

    })

  }

  getToken(){

    return localStorage.getItem('access\_token');

  }

  getIsLoggedIn():boolean{

    let token=localStorage.getItem('access\_token');

    return token!==null?true :false;

  }

  doLogOut(){

    let token=localStorage.removeItem('access\_token');

    if(token==null){

      this.router.navigate(['login'])

    }

  }

  //get UserProfile

  getUserProfile(id:any):Observable<Userclass>{

    return this.http.get<Userclass>(`${this.endpoint}/user-profile/${id}}`)

  }

}

Create file authconfig.interceptor.ts

import { HttpHandler, HttpInterceptor, HttpRequest } from "@angular/common/http";

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

import { AuthService } from "./auth.service";

@Injectable()

export class AuthInterceptor implements HttpInterceptor{

    constructor(private authService:AuthService){}

    intercept(req:HttpRequest<any>,next:HttpHandler){

        const authToken= this.authService.getToken();

        req= req.clone({

            setHeaders:{

                Authorization: "Bearer "+ authToken

            }

        });

        return next.handle(req);

    }

}

Register interceptor in app.module.ts

 providers: [

    {

      provide:HTTP\_INTERCEPTORS,

      useClass:AuthInterceptor,

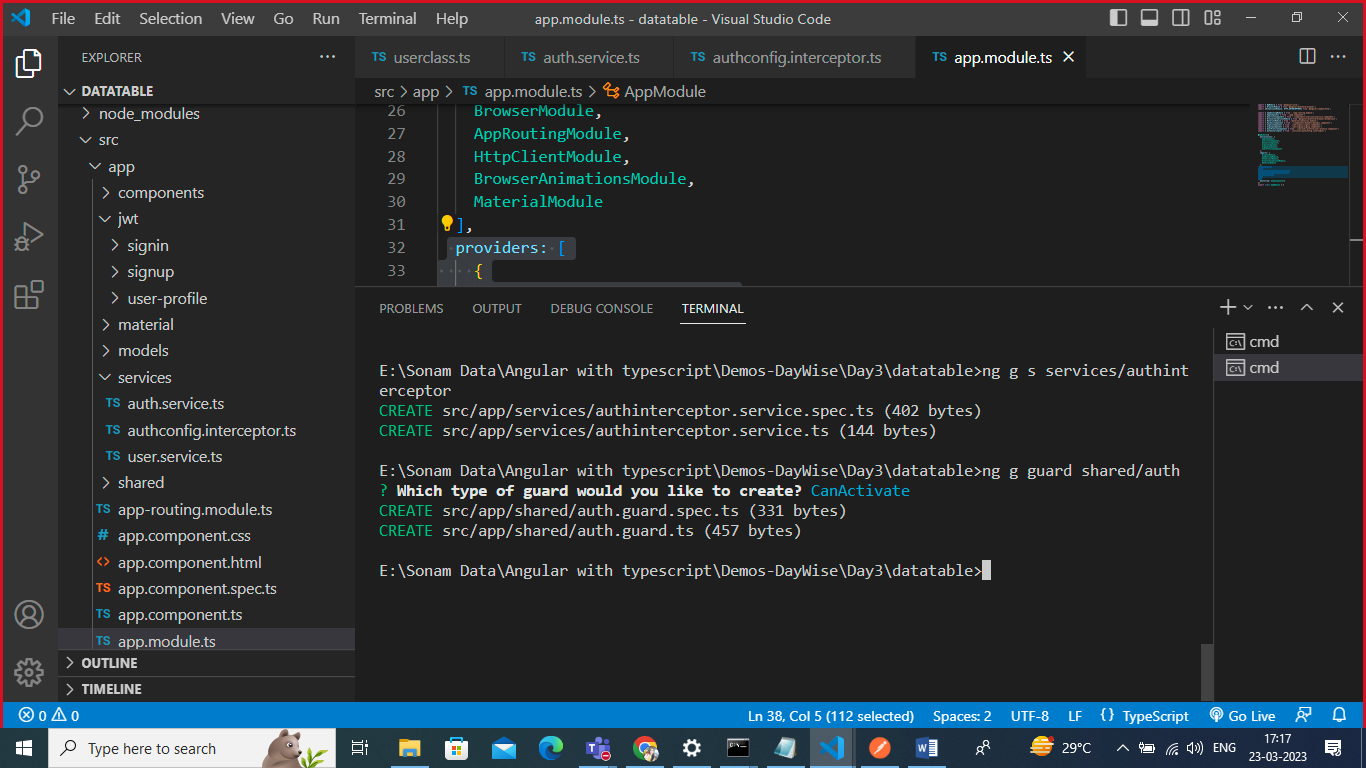
      multi:true

    }

  ],

For creating a protected route: ng g guard shared/auth

Give option of canActivate



Edit auth.guard.ts file

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

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

import { Observable } from 'rxjs';

import { AuthService } from '../services/auth.service';

@Injectable({

  providedIn: 'root'

})

export class AuthGuard implements CanActivate {

  constructor(

     public authService:AuthService,

     public router:Router

  ){}

  canActivate(

    route: ActivatedRouteSnapshot,

    state: RouterStateSnapshot): Observable<boolean | UrlTree> | Promise<boolean | UrlTree> | boolean | UrlTree {

      if(!this.authService.getIsLoggedIn){

          window.alert("Access not allowed");

          this.router.navigate(['login'])

      }

    return true;

  }

}

Set up routes in routing file

const routes: Routes = [

  {path:'',redirectTo:'login',pathMatch:'full'},

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

  {path:'signup',component:SignupComponent},

  {path:'userprofile/:id',component:UserProfileComponent,

canActivate:[AuthGuard]}

];

Let’s create a simple Login form

Sigin.component.ts

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

import { FormBuilder, FormGroup } from '@angular/forms';

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

import { AuthService } from 'src/app/services/auth.service';

@Component({

  selector: 'app-signin',

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

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

})

export class SigninComponent {

  signinForm!: FormGroup;

  constructor(

    private fb: FormBuilder,

    private authService: AuthService,

    public router: Router

  ) { }

  ngOnInit() {

    this.signinForm = this.fb.group({

      email: [''],

      password: ['']

    })

  }

  loginUser() {

    this.authService.signIn(this.signinForm.value);

  }

}

Singin-html

<div>

    <form [formGroup]="signinForm" (ngSubmit)="loginUser()">

        <h3>SignIn Here</h3>

        <div class="form-group">

            <label>Email</label>

            <input type="email" class="form-control"

            formControlName="email" placeholder="Enter Email" required/>

        </div>

        <div class="form-group">

            <label>Password</label>

            <input type="password" class="form-control"

            formControlName="password" placeholder="Enter Password" required/>

        </div>

        <button type="submit" class="btn btn-primary btn-block">SignIn</button>

    </form>

</div>

