# http service testing

data.service.spec.ts file

// data.service.spec.ts

import { TestBed } from '@angular/core/testing';

import { HttpClientTestingModule, HttpTestingController } from '@angular/common/http/testing';

import { DataService } from './data.service';

import { PostClass } from './PostClass';

describe('DataService', () => {

  let service: DataService;

  let httpMock: HttpTestingController;

  beforeEach(() => {

    TestBed.configureTestingModule({

      imports: [HttpClientTestingModule],

      providers: [DataService]

    });

    service = TestBed.inject(DataService);

    httpMock = TestBed.inject(HttpTestingController);

  });

  afterEach(() => {

    httpMock.verify(); // Ensure all requests are handled

  });

  it('should be created', () => {

    expect(service).toBeTruthy();

  });

});

it('should fetch all posts using GET',()=>{

    service.getAllPosts().subscribe(posts=>{

      expect(posts.length).toBe(100);

    });

    const req= httpMock.expectOne(service.url);

    expect(req.request.method).toBe('GET');

  })

// this is your testCase

Goto the component where the service is injected

Demo.component.spec.ts

import { ComponentFixture, TestBed } from '@angular/core/testing';

import { DemoComponent } from './demo.component';

import { HttpClientTestingModule } from '@angular/common/http/testing';

import { DataService } from '../data.service';

describe('DemoComponent', () => {

  let component: DemoComponent;

  let fixture: ComponentFixture<DemoComponent>;

  beforeEach(async() => {

   await TestBed.configureTestingModule({

      declarations: [DemoComponent],

      imports:[HttpClientTestingModule],

      providers:[DataService]

    }).compileComponents();

    fixture = TestBed.createComponent(DemoComponent);

    component = fixture.componentInstance;

    fixture.detectChanges();

  });

  it('should create', () => {

    expect(component).toBeTruthy();

  });

  it('should render title', () => {

      const fixture = TestBed.createComponent(DemoComponent);

      fixture.detectChanges();

      const compiled = fixture.nativeElement as HTMLElement;

      expect(compiled.querySelector('h1')?.textContent).toContain('my demo');

    });

});