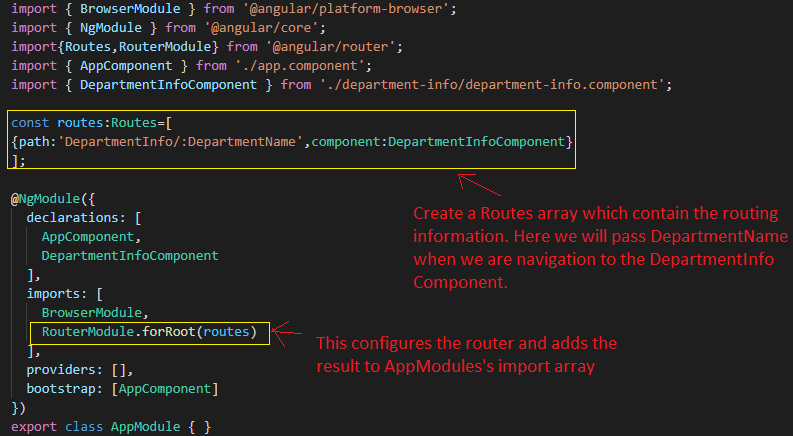
**T12-HOL001\_TestCase [Unit Testing By Mocking ActivatedRoute with Snapshot params]**

In this hands on we will perform unit testing by mocking activated route with snapshot params.

**Tasks:**

1. Create a component called DepartmentInfo.
2. In app.module.ts add the following:



1. Under AppComponent class create a property called Title with a value “ActivedRoute Testing Application” and another property called DepartmentName with a value “HR”.

**Sample Code is given for your reference:**

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

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'ActiveRoute Testing Application';

  DepartmentName='HR';

}

1. Under app.component.html add the [routerLink] to navigate to DepartmentInfoComponent and router-outlet to display routed data.

**Sample Code is given for your reference:**

<h1>{{title}}</h1>

Department Name in App Component:

<a [routerLink]="['DepartmentInfo',DepartmentName]">

 {{DepartmentName}}

</a>

<br/>

  <router-outlet></router-outlet>

1. Under DepartmentInfoComponent do the following:
2. Call the ActivatedRoute service under the constructor
3. Under ngOnInit() method, retrieve the routing value using the snapshot params of activated route and store it under a variable called deptName.

**Sample Code is given for your reference:**

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

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

@Component({

  selector: 'app-department-info',

  templateUrl: './department-info.component.html',

  styleUrls: ['./department-info.component.css']

})

export class DepartmentInfoComponent implements OnInit {

  public deptName:string="";

  constructor(private activeRoute:ActivatedRoute) { }

  ngOnInit() {

    this.deptName = this.activeRoute.snapshot.params.DepartmentName;

  }

}

1. Display the retrieve department name under an h2 tag in department-info.component.html file.

**Sample Code is given for your reference:**

<h2>Department Name Called Under DepartmentInfo Component: {{deptName}}</h2>

1. Use xdescribe() block under app.component.spec.ts
2. Check for the describe() block under depart-info.component.spec.ts
3. Check whether a beforeEach() block that instantiates a new instance of DepartmentInfoComponent for each spec is present. Also under TestBed add the following:
4. imports: [RouterTestingModule,HttpClientTestingModule]
5. providers: [{ provide: ActivatedRoute,

useValue: {

snapshot: {params: {DepaertmentName: 'HR'}}

} } ]

***Note: Here the providers are used to mock the ActivatedRoute service through which it can extract DepartmentName using snapshot params for testing.***

1. Check if there is a test case to check whether the DepartmentInfoComponent is created is present.
2. Create a test case to check if the snapshot params of activated route has extracted the department name “HR” that is passed using routing from AppComponent.
3. Create a test case to check if the extracted department name is displayed in an h2 tag.

**Sample Code is given for your reference:**

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

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

import { of } from 'rxjs';

import {RouterTestingModule} from '@angular/router/testing';

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

import { DepartmentInfoComponent } from './department-info.component';

describe('DepartmentInfoComponent', () => {

  let component: DepartmentInfoComponent;

  let fixture: ComponentFixture<DepartmentInfoComponent>;

  beforeEach(async(() => {

    TestBed.configureTestingModule({

      declarations: [ DepartmentInfoComponent ],

      imports: [

        RouterTestingModule,

        HttpClientTestingModule

      ],

      providers:

      [

        {

          provide: ActivatedRoute,

          useValue: {

            snapshot: {params: {DepaertmentName: 'HR'}}

          }

        }

      ]

    })

    .compileComponents();

}));

  beforeEach(() => {

    fixture = TestBed.createComponent(DepartmentInfoComponent);

    component = fixture.componentInstance;

    fixture.detectChanges();

  });

  it('should create DepartmentInfoComponent', () => {

    expect(component).toBeTruthy();

  });

  it(`should have as department name 'HR'`,  async() => {

    const fixture = TestBed.createComponent(DepartmentInfoComponent);

    let activeRoute=new ActivatedRoute();

    const app = fixture.debugElement.componentInstance;

    fixture.whenStable().then(() => {

    expect(activeRoute.snapshot.params.DepartmentName).toEqual('HR');

    });

  });

  it('should render department name in a h2 tag', async() => {

    const fixture = TestBed.createComponent(DepartmentInfoComponent);

    fixture.detectChanges();

    const compiled = fixture.debugElement.nativeElement;

    let activeRoute=new ActivatedRoute();

    fixture.whenStable().then(() => {

    expect(compiled.querySelector('h2').textContent).toContain('Department Name Called Under DepartmentInfo Component: '+activeRoute.snapshot.params.DepartmentName);

    });

  });

});

1. Use the url generated after running the ng test command to open Karma window in the browser and click Debug button to get Jasmine Test Result.

**Sample output is given below for your reference:**

