Angular Routing

 Angular CLI to generate a basic Angular app with an app routing module, called AppRoutingModule, which is an NgModule where you can configure your routes.

ng new routing-app --routing

# Adding components for routing

To use the Angular router, an app needs to have at least two components so that it can navigate from one to the other.Create components using CLI

ng generate component first

PS D:\Angular\AngularPreparation> ng generate component first

CREATE src/app/first/first.component.html (20 bytes)

CREATE src/app/first/first.component.spec.ts (621 bytes)

CREATE src/app/first/first.component.ts (271 bytes)

CREATE src/app/first/first.component.css (0 bytes)

UPDATE src/app/app.module.ts (633 bytes)

ng generate component second

PS D:\Angular\AngularPreparation> ng generate component secound

CREATE src/app/secound/secound.component.html (22 bytes)

CREATE src/app/secound/secound.component.spec.ts (635 bytes)

CREATE src/app/secound/secound.component.ts (279 bytes)

CREATE src/app/secound/secound.component.css (0 bytes)

UPDATE src/app/app.module.ts (719 bytes)

App-routing.module.ts

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

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

const routes: Routes = [];//array

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

1.Import [RouterModule](https://angular.io/api/router/RouterModule) and [Routes](https://angular.io/api/router/Routes) into your routing module.

The Angular CLI performs this step automatically. The CLI also sets up a [Routes](https://angular.io/api/router/Routes) array for your routes and configures the imports and exports arrays for @[NgModule](https://angular.io/api/core/NgModule)().

CLI app routing module

import { [NgModule](https://angular.io/api/core/NgModule) } from '@angular/core';

import { [Routes](https://angular.io/api/router/Routes), [RouterModule](https://angular.io/api/router/RouterModule) } from '@angular/router'; // CLI imports router

const routes: [Routes](https://angular.io/api/router/Routes) = []; // sets up routes constant where you define your routes

// configures [NgModule](https://angular.io/api/core/NgModule) imports and exports

@[NgModule](https://angular.io/api/core/NgModule)({

imports: [RouterModule.forRoot(routes)],

exports: [[RouterModule](https://angular.io/api/router/RouterModule)]

})

export class AppRoutingModule { }

2.Define your routes in your [Routes](https://angular.io/api/router/Routes) array.

Each route in this array is a JavaScript object that contains two properties. The first property, path, defines the URL path for the route. The second property, component, defines the component Angular should use for the corresponding path.

AppRoutingModule (excerpt)

const routes: [Routes](https://angular.io/api/router/Routes) = [

{ path: 'first-component', component: FirstComponent },

{ path: 'second-component', component: SecondComponent },

];

import { SecoundComponent } from './secound/secound.component';

import { FirstComponent } from './first/first.component';

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

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

const routes: Routes = [

  { path: 'first-component', component: FirstComponent },

  { path: 'second-component', component: SecoundComponent },

];//array

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

3.Add your routes to your application.

First, add links to the two components. Assign the anchor tag that you want to add the route to the [routerLink](https://angular.io/api/router/RouterLink) attribute. Set the value of the attribute to the component to show when a user clicks on each link. Next, update your component template to include <[router-outlet](https://angular.io/api/router/RouterOutlet)>. This element informs Angular to update the application view with the component for the selected route.

app.component.html

<h1>Angular [Router](https://angular.io/api/router/Router) App</h1>

<!-- This nav gives you links to click, which tells the router which route to use (defined in the routes constant in AppRoutingModule) -->

<nav>

<ul>

<li><a [routerLink](https://angular.io/api/router/RouterLink)="/first-component" [routerLinkActive](https://angular.io/api/router/RouterLinkActive)="active">First [Component](https://angular.io/api/core/Component)</a></li> <li><a[routerLink](https://angular.io/api/router/RouterLink)="/second-component" [routerLinkActive](https://angular.io/api/router/RouterLinkActive)="active">Second [Component](https://angular.io/api/core/Component)</a></li> </ul> </nav>

<!-- The routed views render in the <[router-outlet](https://angular.io/api/router/RouterOutlet)>-->

<[router-outlet](https://angular.io/api/router/RouterOutlet)></[router-outlet](https://angular.io/api/router/RouterOutlet)>

Add below links to index.html for bootstrap styling

(CSS)

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">

(JQuery)

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

app.component.html

<!DOCTYPE html>

<html lang="en">

<head>

</head>

<body>

  <!-- This nav gives you links to click, which tells the router which route to use (defined in the routes constant in  AppRoutingModule) -->

  <nav>

    <ul type="none">

      <li><a routerLink="/first-component" >First Component</a></li>

      <li><a routerLink="/second-component" >Second Component</a></li>

    </ul>

  </nav>

  <!-- The routed views render in the <router-outlet>-->

  <router-outlet></router-outlet>

</body>

</html>

app.component.css

nav{

    background-color: black;

    text:white;

    font-family: calibri;

    padding: 20px;

    font-size: 20px;

}

a{

    text-decoration: none;

    color: white;

}

a:hover{

 color:lightblue;

}

ul li{

    display: inline;

    padding-left: 10px;

}

Brower Output

