

## Module 7 – Angular Routes and Navigation

---

Demo Document 1 – Build a server management application and make use of routing

edureka!

**edureka!**

© Brain4ce Education Solutions Pvt. Ltd.

## Create server management app to use different routes.

### In this demo, we will see how create server management app.

**Step 1** – Open visual studio code, create new app by using command 'ng new \_\_\_\_\_'.

```
ng new module5Demo1
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? (Use arrow keys)
> CSS
SCSS [ https://sass-lang.com/documentation/syntax#scss ]
Sass [ https://sass-lang.com/documentation/syntax#the-indented-syntax ]
Less [ http://lesscss.org ]
Stylus [ http://stylus-lang.com ]
```

**Step 2** –First we will create all components,

Navigate to /src/app and create Home component using command 'ng g c home'.

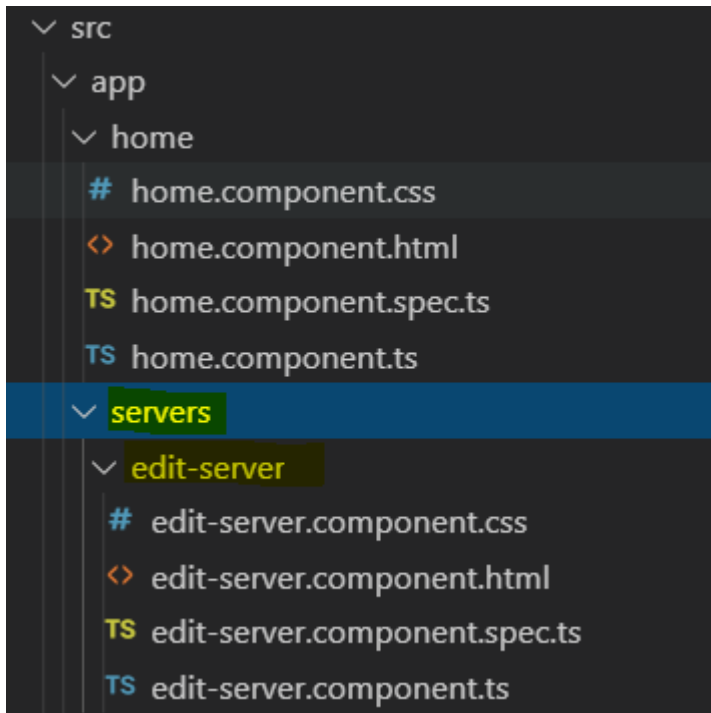
```
> ng g c home
CREATE src/app/home/home.component.html (19 bytes)
CREATE src/app/home/home.component.spec.ts (614 bytes)
CREATE src/app/home/home.component.ts (261 bytes)
CREATE src/app/home/home.component.css (0 bytes)
UPDATE src/app/app.module.ts (467 bytes)
```

Create servers component folder under src/app

```
> ng g c servers
CREATE src/app/servers/servers.component.html (22 bytes)
CREATE src/app/servers/servers.component.spec.ts (635 bytes)
CREATE src/app/servers/servers.component.ts (273 bytes)
CREATE src/app/servers/servers.component.css (0 bytes)
UPDATE src/app/app.module.ts (661 bytes)
```

Create edit-server component by using below command (inside src/app/servers folder)

```
> ng g c edit-server
CREATE src/app/servers/edit-server/edit-server.component.html (26 bytes)
CREATE src/app/servers/edit-server/edit-server.component.spec.ts (657 bytes)
CREATE src/app/servers/edit-server/edit-server.component.ts (288 bytes)
CREATE src/app/servers/edit-server/edit-server.component.css (0 bytes)
UPDATE src/app/app.module.ts (661 bytes)
```



Create server component inside src/app/servers folder

```
> ng g c server
CREATE src/app/servers/server/server.component.html (21 bytes)
CREATE src/app/servers/server/server.component.spec.ts (628 bytes)
CREATE src/app/servers/server/server.component.ts (269 bytes)
CREATE src/app/servers/server/server.component.css (0 bytes)
UPDATE src/app/app.module.ts (751 bytes)
```

Now create users component inside src/app folder

```
> ng g c users
CREATE src/app/users/users.component.html (20 bytes)
CREATE src/app/users/users.component.spec.ts (621 bytes)
CREATE src/app/users/users.component.ts (265 bytes)
CREATE src/app/users/users.component.css (0 bytes)
UPDATE src/app/app.module.ts (829 bytes)
```

Create user component inside src/app/users folder

```
> ng g c user
CREATE src/app/user/user.component.html (19 bytes)
CREATE src/app/user/user.component.spec.ts (614 bytes)
CREATE src/app/user/user.component.ts (261 bytes)
CREATE src/app/user/user.component.css (0 bytes)
UPDATE src/app/app.module.ts (903 bytes)
```

**Step 3** – Now create components for page not found using below command inside src/app folder.

```
..> ng g c page-not-found
CREATE src/app/page-not-found/page-not-found.component.html (29 bytes)
CREATE src/app/page-not-found/page-not-found.component.spec.ts (672 bytes)
CREATE src/app/page-not-found/page-not-found.component.ts (299 bytes)
CREATE src/app/page-not-found/page-not-found.component.css (0 bytes)
UPDATE src/app/app.module.ts (1013 bytes)
```

**Step 4** – Open app-routing.module.ts and add routes as below.

```
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';
import { HomeComponent } from './home/home.component';
import { ServersComponent } from './servers/servers.component';
import { ServerComponent } from './servers/server/server.component';
import { EditServerComponent } from './servers/edit-server/edit-server.component';
import { UsersComponent } from './users/users.component';
import { UserComponent } from './users/user/user.component';

const routes: Routes = [
  {
    path: '',
    component: HomeComponent
  },
  {
    path: 'servers',
    component: ServersComponent,
    children: [
      {
        path: ':id',
        component: ServerComponent
      },
      {
        path: ':id/edit',
        component: EditServerComponent
      }
    ]
  },
  {
    path: 'users',
    component: UsersComponent,
    children: [
      {
        path: ':id/:name',
        component: UserComponent
      }
    ]
  }
]
```

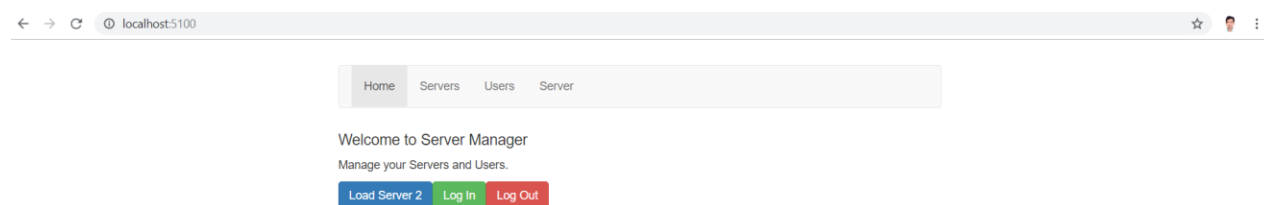
```
    },  
    {  
      path: '**',  
      redirectTo: '/not-found',  
      pathMatch: 'full'  
    }  
  ];  
  
  @NgModule({  
    imports: [RouterModule.forRoot(routes)],  
    exports: [RouterModule]  
  })  
  export class AppRoutingModule { }
```

edureka!

**Step 5** – Open app.component.html and add below code

```
<div class="container">
  <div class="row">
    <div class="col-xs-12 col-sm-10 col-md-8 col-sm-offset-1 col-md-offset-2">
      <ul class="nav nav-tabs">
        <li role="presentation"
          routerLinkActive="active"
          [routerLinkActiveOptions]="{exact: true}">
          <a routerLink="/">Home</a>
        </li>
        <li role="presentation"
          routerLinkActive="active">
          <a routerLink="/servers">Servers</a>
        </li>
        <li role="presentation"
          routerLinkActive="active">
          <a [routerLink]="['/users']">Users</a>
        </li>
      </ul>
    </div>
  </div>
  <div class="row">
    <div class="col-xs-12 col-sm-10 col-md-8 col-sm-offset-1 col-md-offset-2">
      <router-outlet></router-outlet>
    </div>
  </div>
</div>
```

**Step 6** -run ng serve –open command to run application



Now navigate to different routes /links by clicking on link

edureka!