

Lesson 03 Demo 7

Implementing Child and Parent route

Objective: To implement child and parent route

Tools required: Ubuntu, Visual Studio, Node.js

Prerequisites: Knowledge of JavaScript, Typescript, Node.js

Steps to be followed:

1. Create an angular app
2. Run it on the browser
3. Create child and parent route
4. Check the output on browser

Step 1: Create an angular app

1.1 Open Visual Studio

1.2 Create a new project using the command **ng new angularexamples** and enter **yes:**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

npm WARN using --force Recommended protections disabled.
● labuser@ubuntu2204:~$ npm update

up to date, audited 5 packages in 830ms

found 0 vulnerabilities
● labuser@ubuntu2204:~$ npm i @angular/core -s
○ labuser@ubuntu2204:~$ ng new angularexamples
? Would you like to add Angular routing? (y/N) █
```

1.3 Open the created project folder and go to **File -> open folder -> angularexamples**

1.4 Now, go to the terminal and type the commands:

ng g c clients

ng g c services

ng g c testimonials

```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL GITLENS SQL CONSOLE

● labuser@ubuntu2204:~/angularexamples$ ng g c clients
CREATE src/app/clients/clients.component.css (0 bytes)
CREATE src/app/clients/clients.component.html (22 bytes)
CREATE src/app/clients/clients.component.spec.ts (606 bytes)
CREATE src/app/clients/clients.component.ts (206 bytes)
UPDATE src/app/app.module.ts (704 bytes)
● labuser@ubuntu2204:~/angularexamples$ ng g c services
CREATE src/app/services/services.component.css (0 bytes)
CREATE src/app/services/services.component.html (23 bytes)
CREATE src/app/services/services.component.spec.ts (613 bytes)
CREATE src/app/services/services.component.ts (210 bytes)
UPDATE src/app/app.module.ts (794 bytes)
● labuser@ubuntu2204:~/angularexamples$ ng g c testimonials
CREATE src/app/testimonials/testimonials.component.css (0 bytes)
CREATE src/app/testimonials/testimonials.component.html (27 bytes)
CREATE src/app/testimonials/testimonials.component.spec.ts (641 bytes)
CREATE src/app/testimonials/testimonials.component.ts (226 bytes)
UPDATE src/app/app.module.ts (900 bytes)
○ labuser@ubuntu2204:~/angularexamples$ █
```

Note: These are names of the tabs where we will be performing the routing on.

1.5 Go to `src -> app -> app-routing.module.ts` file and write the code below:

```
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';
import { ClientsComponent } from './clients/clients.component';
import { ServicesComponent } from './services/services.component';
import { TestimonialsComponent } from './testimonials/testimonials.component';

const routes: Routes = [
  {
    path: 'clients',
    component: ClientsComponent
  },
  {
    path: 'services',
    component: ServicesComponent
  },
  {
    path: 'testimonials',
    component: TestimonialsComponent
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

1.6 Go to the `app.component.html` file and write the code below:

```
<h1> Routing </h1>
<a routerLink="clients">Clients</a>
<br>
<a routerLink="services">Services</a>
<br>
```

```
<a routerLink="testimonials">Testimonials</a>
```

```
<router-outlet></router-outlet>
```

1.7 Go to the **app.module.ts** file and see that the code should be automatically updated as below:

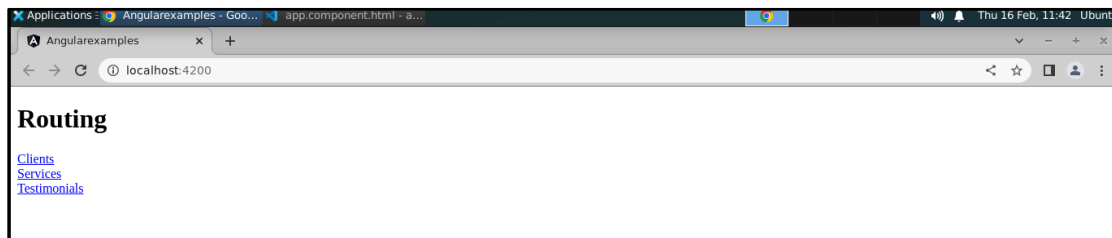
```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ClientsComponent } from './clients/clients.component';
import { ServicesComponent } from './services/services.component';
import { TestimonialsComponent } from './testimonials/testimonials.component';

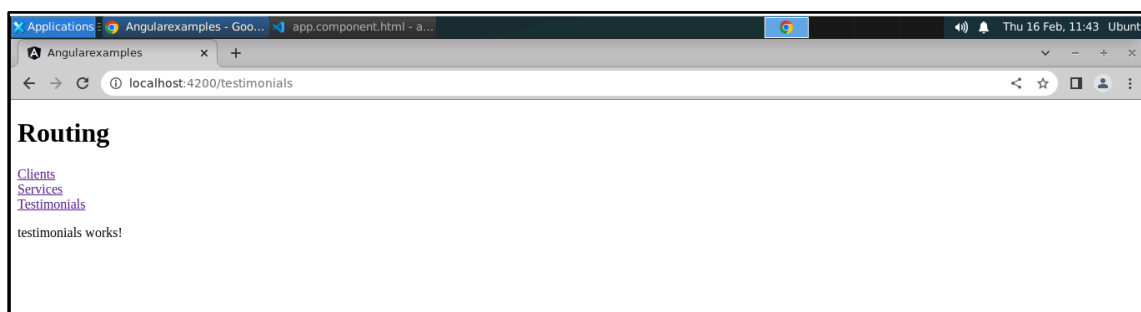
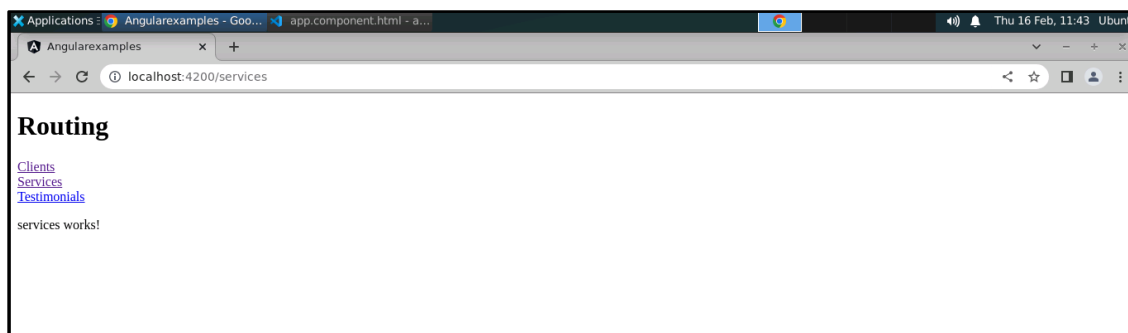
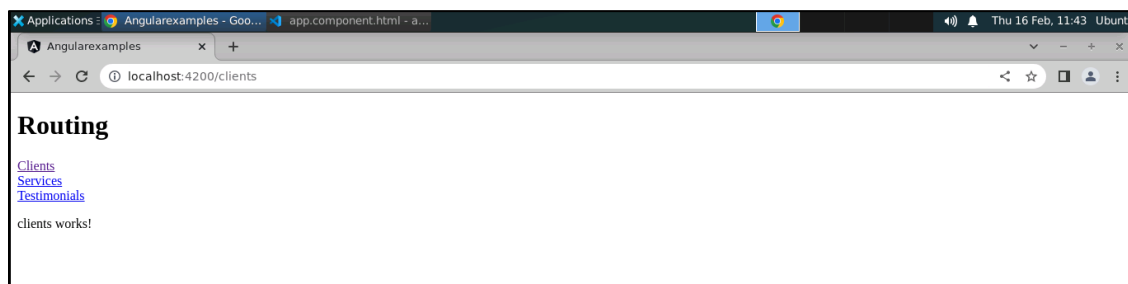
@NgModule({
  declarations: [
    AppComponent,
    ClientsComponent,
    ServicesComponent,
    TestimonialsComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Step 2: Run it on the browser

2.1 Run the **ng serve** command in the terminal and copy the **localhost address** and paste it opens in the browser:



2.2 Click on any of the service and you will see that it is redirected to that service, as shown below:



Step 3: Create child and parent route

3.1 Create a child of any route named client, and the childs are Simplilearn and Angular

```
ng g c simplilearn
```

```
ng g c angular
```

```

✓ Compiled successfully.
^C
• labuser@ubuntu2204:~/angularexamples$ ng g c simplilearn
CREATE src/app/simplilearn/simplilearn.component.css (0 bytes)
CREATE src/app/simplilearn/simplilearn.component.html (26 bytes)
CREATE src/app/simplilearn/simplilearn.component.spec.ts (634 bytes)
CREATE src/app/simplilearn/simplilearn.component.ts (222 bytes)
UPDATE src/app/app.module.ts (776 bytes)
• labuser@ubuntu2204:~/angularexamples$ 
connect  Tests ✓ 0 × 0 coverage: progress

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS  SQL CONSOLE

CREATE src/app/simplilearn/simplilearn.component.ts (222 bytes)
UPDATE src/app/app.module.ts (776 bytes)
• labuser@ubuntu2204:~/angularexamples$ ng g c angular
CREATE src/app/angular/angular.component.css (0 bytes)
CREATE src/app/angular/angular.component.html (22 bytes)
CREATE src/app/angular/angular.component.spec.ts (606 bytes)
CREATE src/app/angular/angular.component.ts (206 bytes)
UPDATE src/app/app.module.ts (862 bytes)
• labuser@ubuntu2204:~/angularexamples$ 

```

3.2 Go to the **app-routing-module.ts** file and edit the client route as below, where we have created the child component:

```

{
  path: 'clients',
  component: ClientsComponent,
  children: [{
    path: 'simplilearn',
    component: SimplilearnComponent
  }],
}

```

```

    path: 'angular',
    component: AngularComponent
  }}
},

```

```

const routes: Routes = [
  {
    path: 'clients',
    component: ClientsComponent,
    children: [{
      path: 'simplilearn',
      component: SimplilearnComponent
    }],
  },
  {
    path: 'angular',
    component: AngularComponent
  }
],
},
[
  {
    path: 'services',
    component: ServicesComponent
  },
  {
    path: 'testimonials',
    component: TestimonialsComponent
  }
]

```

3.3 Go to **clients.component.html** file. You can find this under the clients folder. Write the code as below:

```

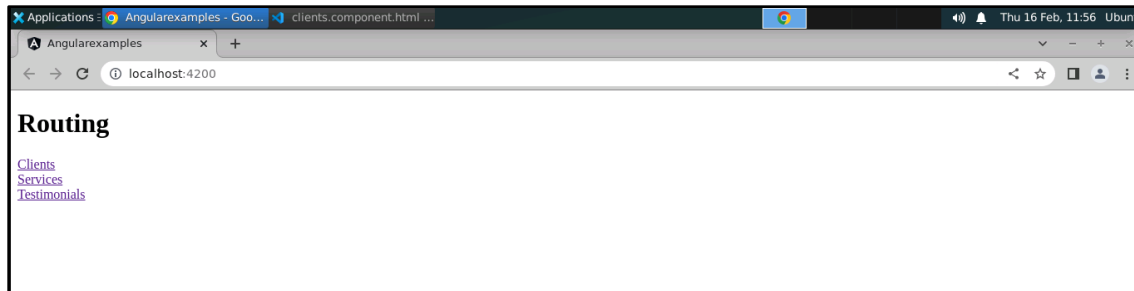
<p>clients works!</p> <br>
<a routerLink="simplilearn">Redirected to Simplilearn</a>
<br>
<a routerLink="Angular">Redirected to Angular</a>

<router-outlet></router-outlet>

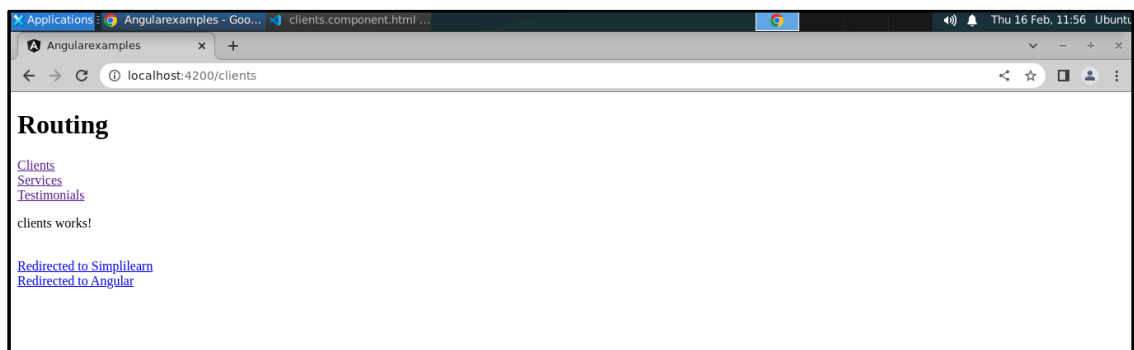
```

Step 4: Check the output on browser

4.1 Run the code and view the output



4.2 Click on clients and you should see the two child added



4.3 Click on any of them and you should be redirected to the child route

