Angular 5 CT Specialization

Angular Material Lab 7

January 2019

Document Control

Version	1.0
Created by	Karthikeyan J
Updated by	Karthikeyan J
Reviewed by	
Date	January, 2019
Earlier versions	

Table of Contents

Creating and registering *injectable* Angular Service

1. Run the following command to create an **injectable** Angular service class:

```
D:\angular5ctspecialization\angularmaterialapp>ng g s contact-manager\services\user --skipTests
```

2. Modify the **@Injectable()** decorator to make sure that the service is available only when the **ContactManager** module is lazily loaded. In other words, we are **limiting** the scope of the service only to the **ContactManager** module:

```
@Injectable()
```

3. Open **contact-manager.module.ts** file & the following import statement:

```
import {UserService} from './services/user.service';
```

4. Open **contact-manager.module.ts** file & register the service class as a provider so that it is available only when the **ContactManger** module is lazily loaded. Add the following **providers** array to the **NgModule's** decorator:

```
providers:[UserService]
```

5. Add the following code inside the **UserService** class:

```
constructor()
{
    console.log("UserService instance created");
}
```

Creating the model classes

6. Run the following commands to create the model classes respectively:

```
D:\angular5ctspecialization\angularmaterialapp>ng g class contact-manager\models\user --skipTests
```

```
D:\angular5ctspecialization\angularmaterialapp>ng g class contact-manager\models\note --skipTests
```

7. Open **user.ts** file & add the following import statement into it:

```
import {Note} from './note';
```

8. Add the following code inside the **User** class:

```
id:number;
birthDate:Date;
name:string;
avatar:string;
bio:string;
notes:Note[] = [];
```

9. Open **note.ts** file & add the following code inside the **Note** class:

```
id:number;
title:string;
date:Date;
```

Using *HttpClient* inside the service for server communication

10. Open **contact-manager.module.ts** file & add the following import statement:

```
import {HttpClientModule} from '@angular/common/http';
```

11. Add the following to the **imports** array of the **NgModule** decorator:

```
, HttpClientModule
```

12. Open **user.service.ts** file & add the following import statements:

```
import {HttpClient} from '@angular/common/http';
import {Observable} from 'rxjs';
import { User } from '../models/user';
```

13. Open **user.service.ts** file & add the following as a constructor parameter of the **UserService** class:

```
private http:HttpClient
```

14. Add the following method inside the **UserService** class:

```
public LoadAllUsers():Observable<User[]>
{
    return this.http.get<User[]>('https://angular-material-
                                  api.azurewebsites.net/users');
}
```

Subscribing to the *Observable* inside the component to get data

15. Open **toolbar.component.ts** file & add the following import statements:

```
import {UserService} from './services/user.service';
import {Observable} from 'rxjs';
import {User} from './models/user';
```

16. Add the following inside the **ToolbarComponent** class:

```
public users:Observable<User[]>;
```

17. Add the following parameter to the constructor of the **ToolbarComponent** class:

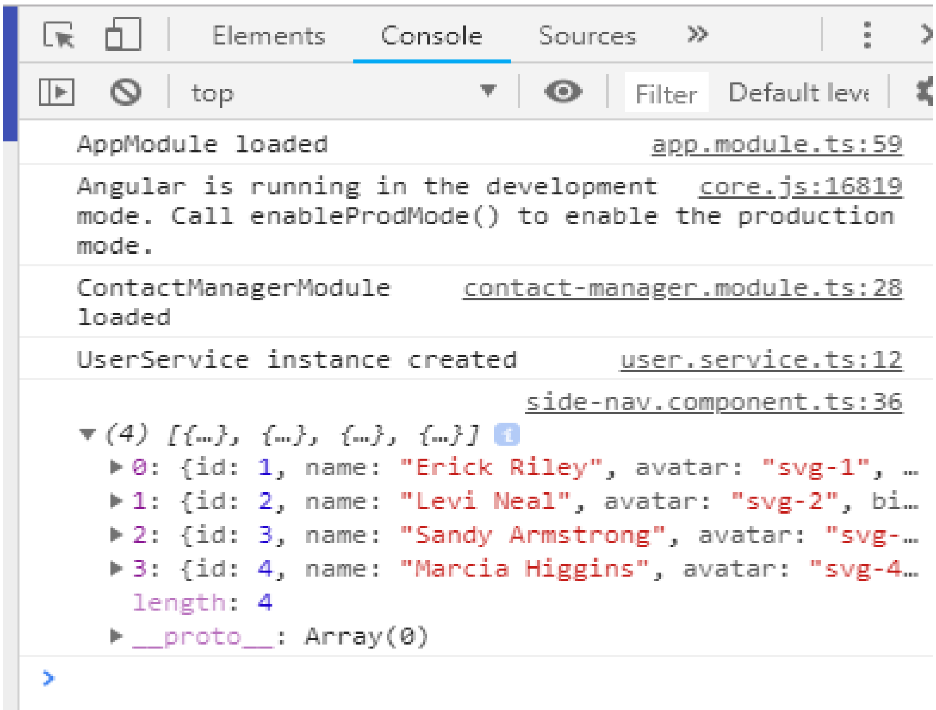
```
private svc:UserService
```

18. Add the following code inside the **ngOnInit()** method of the **ToolbarComponent** class:

```
this.users = this.svc.LoadAllUsers();

this.users.subscribe(
   (data) => console.log(data),
   (error) => console.log(error)
);
```

19. Serve the application & navigate to http://localhost:4200/contactmanager. Open the browser console & observe that when the **ToolbarComponent** is loaded, the console displays the number of records fetched from the REST API *(see snapshot below)*:



Using Angular material's *<mat-nav-list>* component to create a navigation list & Angular's *async* pipe to subscribe to observables

20. **Open toolbar.component.html** & comment out the **<mat-list>** element's markup.

21. Add the following markup instead:

```
<mat-nav-list>
  <a mat-list-item href="#" *ngFor="let user of users | async">
      {{user.name}}
  </a>
</mat-nav-list>
```

22. Serve the application & navigate to http://localhost:4200/contactmanager. Observe that the navigation menu now displays the contact names as shown in the snapshot below: