Assignment No 2 (C)

Aim: Create an Angular application which will do following actions: Register User, Login User, Show User Data on Profile Component.

Performance (3)	Understanding (1)	Regularity (1)	Total (5)	Sign of Staff

Assignment-4. Create an Angular application which will do following actions: Register User, Login User, Show User Data on Profile Component.

Useful link - https://www.tutorialsteacher.com/angular/install-angular

1. Angular requires a current, active LTS(long term support) or maintenance LTS version of Node.js and NPM.

install node.js https://nodejs.org/
It will automatically install NPM - node package manager

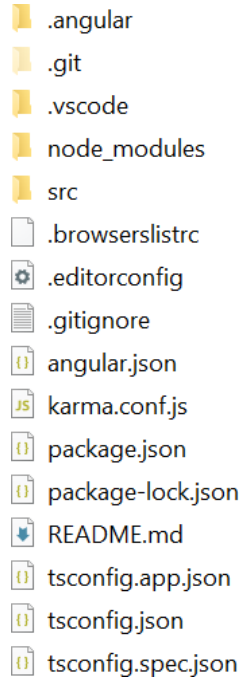
2. Install Angular CLI

npm install -g @angular/cli@latest
To Create Angular 2 Application Angular CLI is required

3. To create new project

through CLI go to folder of the new projectGive command as -
ng new project-name press ENTER

The project will be created as directory structure below –



Open folder src/app
Modify app.module.ts for form application

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

import { AppRoutingModule } from'./app-routing.module';
import { AppComponent } from'./app.component';
```

```
import {FormsModule} from'@angular/forms'

@NgModule({
  declarations: [
  AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    FormsModule,
  ],
  providers: [],
  bootstrap: [AppComponent]
})
exportclassAppModule { }
```

Open app.component.html
Write html code for form (representative code is mentioned here, modify for multiple inputs)

```
<h1>Simple Form</h1>
<form#simpleForm = "ngForm"(ngSubmit) = "getValues(simpleForm.value)">
  <inputtype ="text"ngModelname = "user"placeholder = "Enter Name">
  <br><br>
  <inputtype ="text"ngModelname = "age"placeholder = "Enter age">
  <br><br>
  <inputtype ="text"ngModelname = "city"placeholder = "Enter city">
  <br><br>
  <button>Get user value</button>
</form>
```

Make changes in app.component.ts

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

@Component({
  selector:'app-root',
  templateUrl:'./app.component.html',
  styleUrls: ['./app.component.css']
})
exportclassAppComponent {
  title = 'AngProj1';
  getValues(val:any)
  {
    console.log(val);
  }
}
```

Here getValue() function which is called in form file is defined.
You can check inputted values through form in console.
Step 1: Create an Angular application using the Angular CLI

ng new my-app

Step 2: Create a user service to handle user-related functionality. This service will be responsible for making API calls to your server to register, login, and retrieve user data. You can generate a service using the

Angular CLI:

ng generate service user

Step 3: Define a user model to represent the user data. The user model will typically contain properties like username, email, password, etc.

Step 4: Create a registration component to allow users to register for an account. This component will have a form that accepts user input and submits it to the user service for processing. The registration component will also handle any error messages returned by the user service.

Step 5: Create a login component to allow users to log in to their account. This component will have a form that accepts user input and submits it to the user service for processing. The login component will also handle any error messages returned by the user service.

Step 6: Create a profile component to display the user data. This component will use the user service to retrieve the user data and display it on the screen. The profile component should only be accessible to users who are logged in.

Step 7: Create a routing module to define the routes for your application. The routing module should include routes for the registration, login, and profile components.

Step 8: Create a navigation bar to allow users to navigate between the different components. The navigation bar should be included in a shared component that can be used across the application.

Step 9: Update the app module to include the routing module and the user service.

Step10:Update the app component to include the navigation bar and the router outlet. Once you have completed these steps, you should have an Angular application that allows users to register, log in, and view their user data on a profile component.

Build application

Use Angular CLI command ng serve -o to build an application.The -o indicates to open it automatically in the default browser.

1. Use NPM command 'npm start' to build an application
http://localhost:4200 to see the application home page.
2.
3.Open the terminal in VS Code from menu Terminal -> New Terminal,
and type ng serve -o command and press enter,

You can send the form contents from console to other page.
On the basis of above implementation, you can design login user, show user data.