Scouting App Training

Team 294

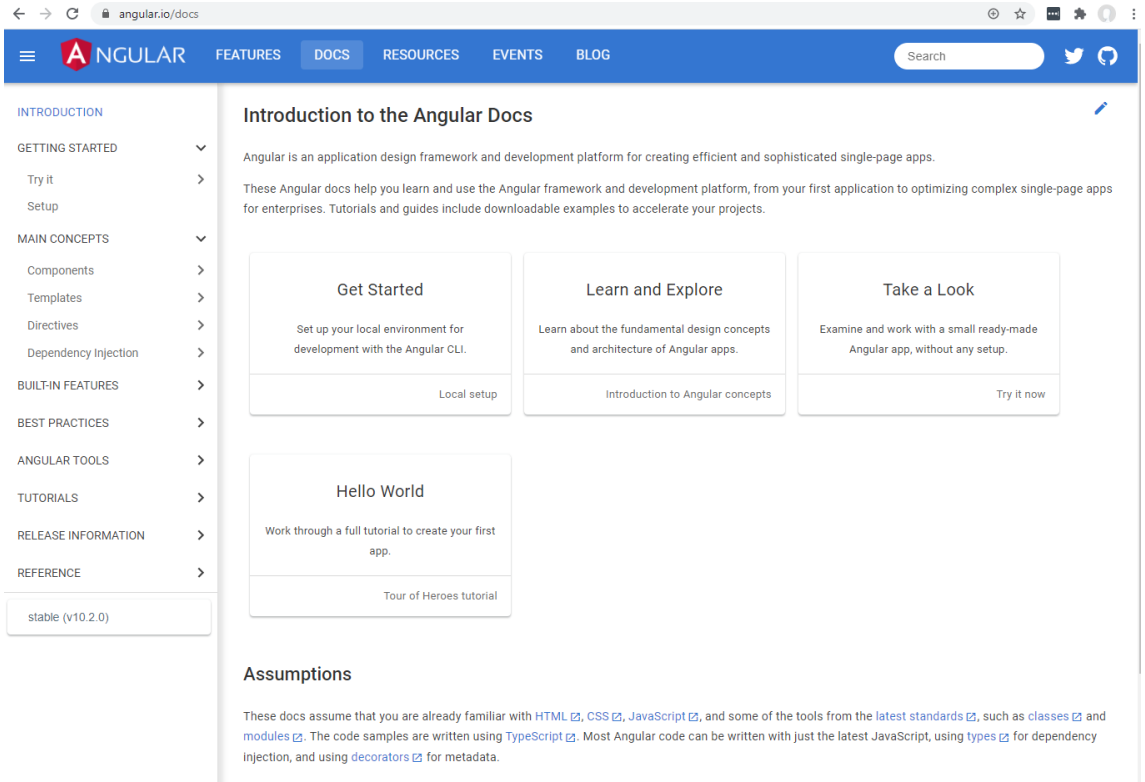
Introduction to Angular and Web Service Programming

Angular Basics

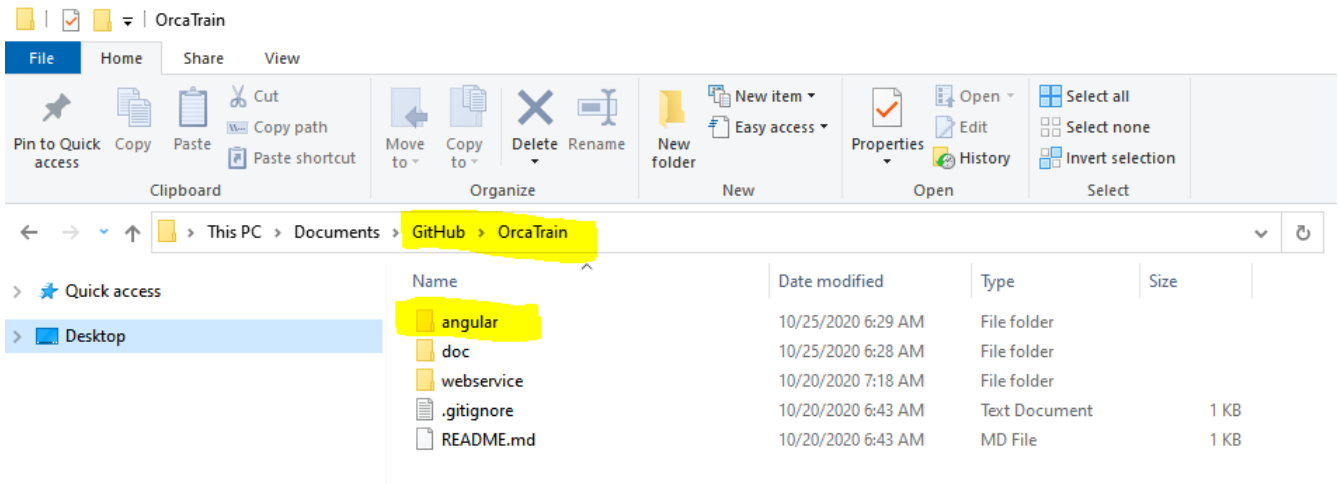
Angular Documentation

https://angular.io/docs

Version 10.2



VS Code



Open the angular folder in VS Code

Run the app from a command line window from the angular directory

ng serve

http://localhost:4200

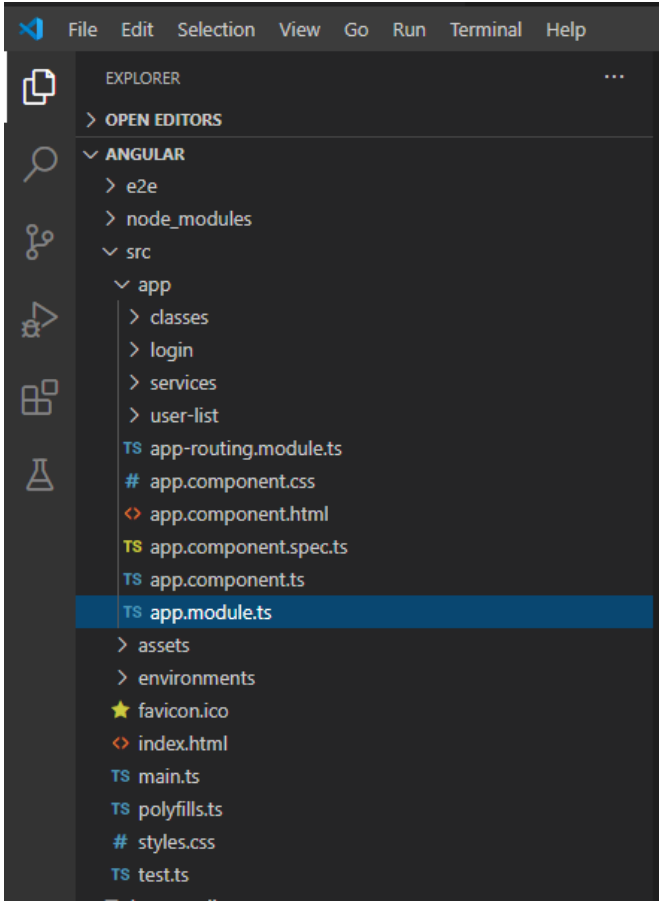
Angular Project

Classes – hold the data

Services – connect to web services

Components – user interface

- Login
- User List



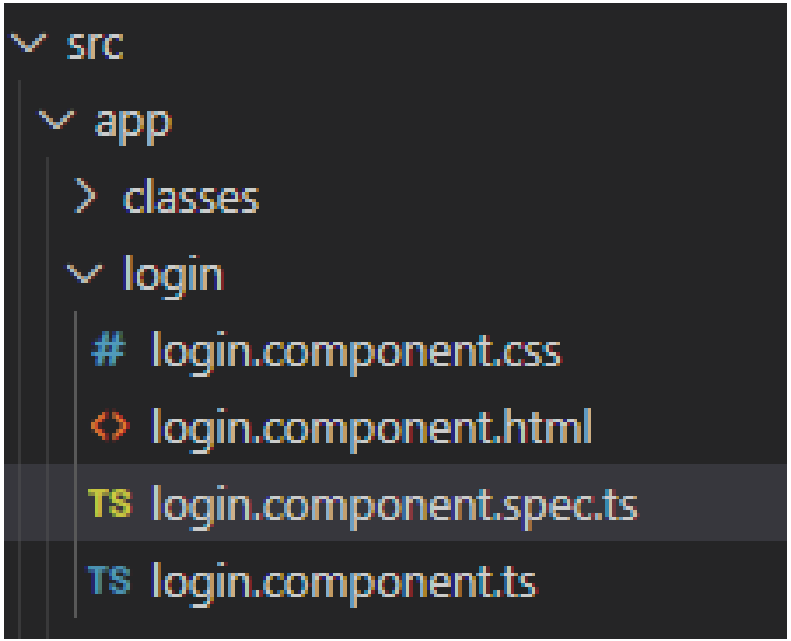
Component

Create using Angular CLI

- ng generate component login

Each component has 4 associated files

- .html
- .css
- .ts
- .spec.ts

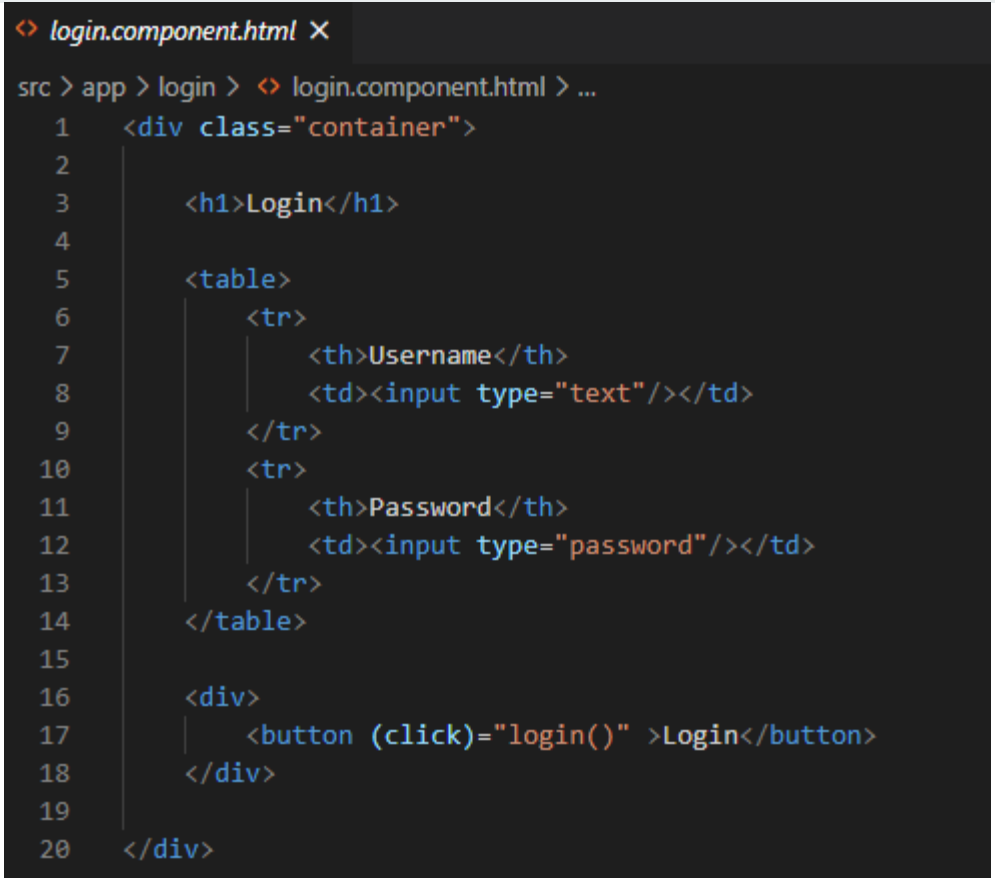


HTML Template

Standard HTML with Angular extensions

Template statements allow you to respond to user events

Line 17 tells Angular to call the login method whenever the user clicks the login button



TypeScript

Each component has a TypeScript file associated with it where you put any component specific code.

Line 23 defines the login function that was referenced in the template.

Code that is shared between components should be in a service.

Use the router to navigate to different pages

Structural Directives

Angular provides built-in directives to bring your html to life and do things like branching and looping

Line 8 defines a loop that will print a new row for every user in the userList array

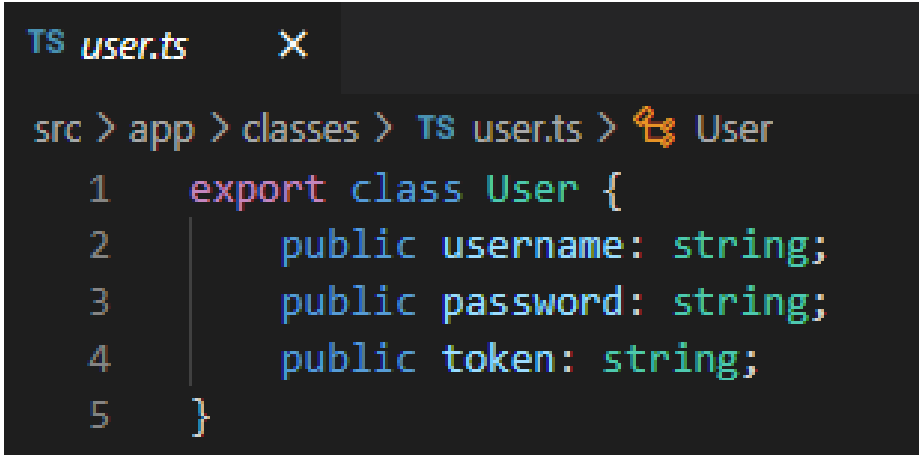
Line 9 will print the value contained in the associated variable

Classes

Angular applications use classes and interfaces to provide structure to data and allow for type checking

Angular applications are designed for the web and are best written using a functional programming style (rather than object-oriented)

Classes should only contain data and behavior should be modeled in functions



Services

Services contain shared logic that is available to all components

Angular uses dependency injection to give each component a reference to the service

Services should hold data that is shared across the application

Services should contain the logic to get the data from a web service

app.service.ts

src > app > services > app.service.ts > AppService

```
1  import { Observable } from 'rxjs';
2  import { HttpClient,  HttpHeaders, HttpErrorResponse} from '@angular/common/http';
3  import { User } from '../classes/user';
4  import { Injectable } from '@angular/core';
5  import { throwError } from 'rxjs';
6  import { catchError, retry } from 'rxjs/operators';
7
8  @Injectable({providedIn: 'root'})
9  export class AppService {
10   public user: User;
11   public userList: User[];
12
13   constructor(private http: HttpClient) {
14     this.userList = [
15       {"username":"paul","password":"abc","token":"testtoken1"},
16       {"username":"michael","password":"def","token":"testtoken2"},
17       {"username":"bryan","password":"ghi","token":"testtoken3"}
18     ];
19   }
```