Angular

AJ

What is Angular

Angular is a JavaScript **(actually a TypeScript based open-source full-stack web application)** framework which makes you able to create reactive **Single Page Applications** (SPAs). Angular is completely based on components. It consists of several components which forms a tree structure with parent and child components. Angular's versions beyond 2+ are generally known as **Angular** only. The very first version Angular 1.0 is known as **AngularJS.**

"Angular is a complete rewrite of AngularJS by the same team that built AngularJS."

What is Single Page Application (SPA)?
A single page application is a web application or a website which provides users a very fluid, reactive and fast experience similar to a desktop application. It contains menu, buttons and blocks on a single page and when a user clicks on any of them; it dynamically rewrites the current page rather than loading entire new pages from a server. That's the reason behind its reactive fast speed.

Difference between AngularJS and Angular

AngularJS	Angular
AngularJS is common and popular name of the first version of Angular1.0.	Angular is common and popular name of the Angular's version beyond 2+
AngularJS is a JavaScript-based open-source front-end web framework.	Angular is a TypeScript-based open-source full-stack web application framework.
AngularJS uses the concept of scope or controller.	Instead of scope and controller, Angular uses hierarchy of components as its primary architectural characteristic.
AngularJS has a simple syntax and used on HTML pages along with the source location.	Angular uses the different expression syntax. It uses "[ ]" for property binding, and "( )" for event binding.
AngularJS is a simple JavaScript file which is used with HTML pages and doesn't support the features of a server-side programming language.	Angular uses of Microsoft's TypeScript language, which provides Class-based Object Oriented Programming, Static Typing, Generics etc. which are the features of a server-side programming language.
AngularJS doesn't support dynamic loading of the page.	Angular supports dynamic loading of the page.

Angular Features

A list of most important features and benefits of Angular:

Angular supports multiple platforms

Angular is a cross platform language. It supports multiple platforms. You can build different types of apps by using Angular.

1.Desktop applications: Angular facilitates you to create desktop installed apps on different types of operating systems i.e. Windows, Mac or Linux by using the same Angular methods which we use for creating web and native apps.

2.Native applications: You can built native apps by using Angular with strategies from Cordova, Ionic, or NativeScript.

3.Progressive web applications: Progressive web applications are the most common apps which are built with Angular. Angular provides modern web platform capabilities to deliver high performance, offline, and zero-step installation apps.

High Speed, Ultimate Performance

Angular is amazingly fast and provides a great performance due to the following reasons:

1.Universal support: Angular can be used as a front-end web development tool for the programming languages like Node.js, .Net, PHP, Java Struts and Spring and other servers for near-instant rendering in just HTML and CSS. It also optimizes the website for better SEO.

2.Code splitting: Angular apps are fast and loads quickly with the new Component Router, which delivers automatic code-splitting so users only load code required to render the view they request.

3.Code generation: Angular makes your templates in highly optimized code for today?s JavaScript virtual machines which gives the benefits of hand-written code.

Productivity

Angular provides a better productivity due to its simple and powerful template syntax, command line tools and popular editors and IDEs.

1. **Powerful templates:** Angular provides simple and powerful template syntax to create UI view quickly.
2. **IDEs:** Angular provides intelligent code completion, instant errors, and other feedback in popular editors and IDEs.
3. **Angular CLI:** Angular CLI provides command line tools start building fast, add components and tests, and then instantly deploy.

Full Stack Development

Angular is a complete framework of JavaScript. It provides Testing, animation and Accessibility. It provides full stack development along with Node.js, Express.js, and MongoDB.

1. **Testing:** Angular provides Karma and Jasmine for unit testing. B y using it, you can check your broken things every time you save. Karma is a JavaScript test runner tool created by Angular team. Jasmine is the testing framework form unit testing in Angular apps, and Karma provides helpful tools that make it easier to us to call our Jasmine tests whilst we are writing code.
2. **Animation Support:** Angular facilitates you to create high-performance, complex choreographies and animation timelines with very little code through Angular's intuitive API.
3. **Accessibility:** In Angular, you can create accessible applications with ARIA-enabled components, developer guides, and built-in a11y test infrastructure.

How to install Angular

Angular Environment Setup

Install Visual Studio Code IDE or JetBrains WebStorm

Install Node.js

Use npm to install Angular CLI

npm install -g @angular/cli

```
> npm install -g @angular/cli
> ng new my-dream-app
> cd my-dream-app
> ng serve   ng serve --open
```

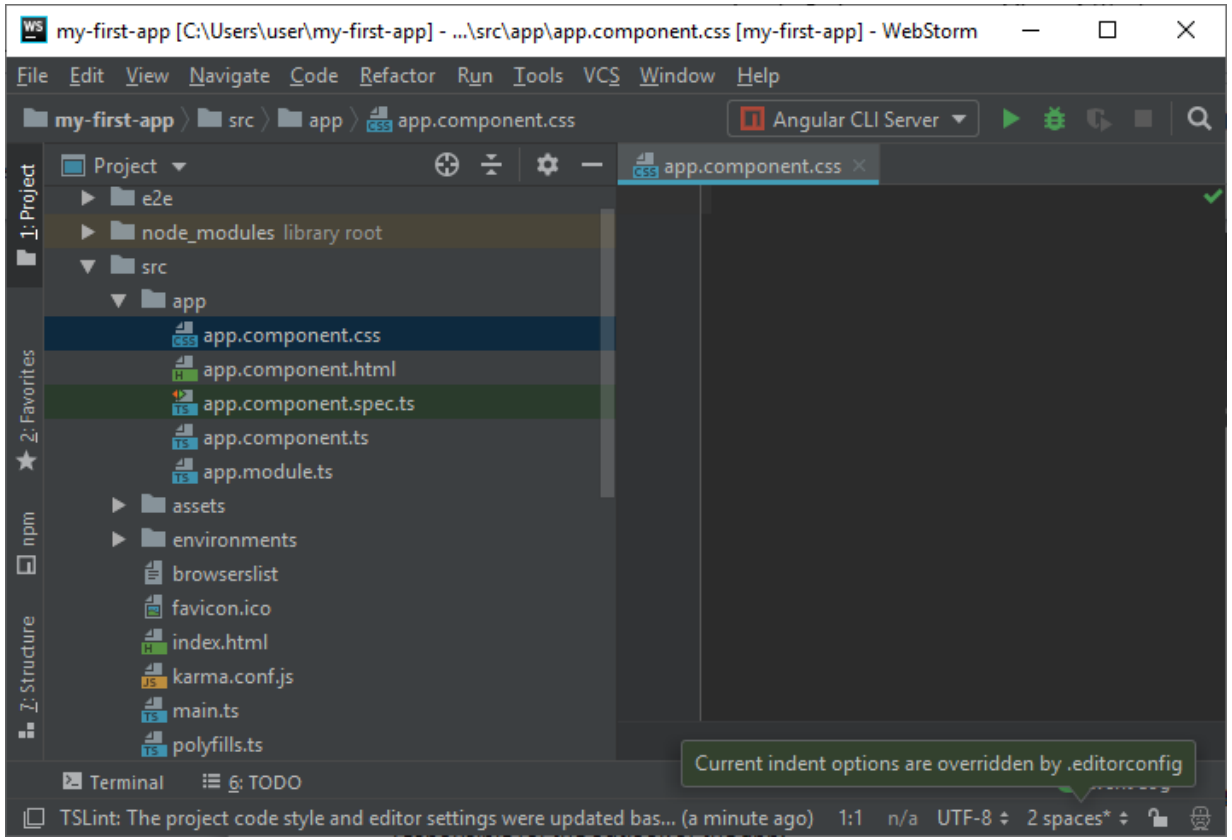
You will see 5 components there:

- app.component.css
- app.component.html
- app.component.spec.ts
- app.component.ts
- app.module.ts

You can see the code within the different components to understand what is going on and which part is responsible for the outlook of the app.

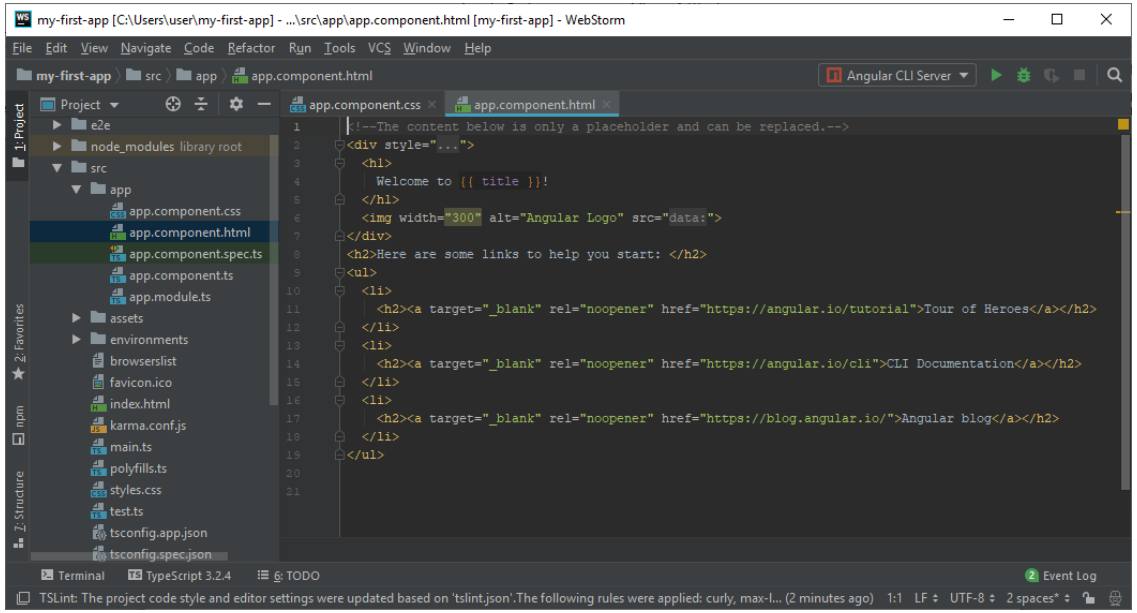
app.component.css

This part is empty because we don't specify any CSS here.



app.component.html

This is the most important component, the front page of your app. Here, you can change the salutation used before your app's name. You can also change the content on the front page and their respective links.



app.component.spec.ts:
This file is used for testing purpose only.

app.component.ts
You can change the name of your app here. You just have to change the title.

app.module.ts it has some libraries which are imported and also a declarative which is assigned the app component

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

import { AppComponent } from './app.component';

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

Angular App files explanation

Files used in Angular 7 App folder

Angular App files which are mainly used in your project are given below:

- **src folder:** This is the folder which contains the main code files related to your angular application.
- **app folder:** The app folder contains the files, you have created for app components.
- **app.component.css:** This file contains the cascading style sheets code for your app component.
- **app.component.html:** This file contains the html file related to app component. This is the template file which is used by angular to do the data binding.
- **app.component.spec.ts:** This file is a unit testing file related to app component. This file is used along with other unit tests. It is run from Angular CLI by the command ng test.
- **app.component.ts:** This is the most important typescript file which includes the view logic behind the component.
- **app.module.ts:** This is also a typescript file which includes all the dependencies for the website. This file is used to define the needed modules to be imported, the components to be declared and the main component to be bootstrapped.

Angular App files explanation

Other Important files

•**package.json:** This is npm configuration file. It includes details about your website's package dependencies along with details about your own website being a package itself.

•**package-lock.json :** This is an auto-generated and modified file that gets updated whenever npm does an operation related to node_modules or package.json

•**angular.json:** It is very important configuration file related to your angular application. **It defines the structure of your app and includes any settings associated with your application.** Here, you can specify environments on this file (development, production). This is the file where we add Bootstrap file to work with Angular 7.

•**.gitignore:** This file is related to the source control git.

•**.editorconfig:** This is a simple file which is used to maintain consistency in code editors to organize some basics such as indentation and whitespaces.

•**assets folder:** This folder is a placeholder for resource files which are used in the application such as images, locales, translations etc.

environments folder: The environments folder is used to hold the environment configuration constants that help when building the angular application. The constants are defined in 2 separate .ts files (environment.ts and environment.prod.ts), where these constants are used within the angular.json file by the Angular CLI. For example, if you run the ng build command, it will build the application using the development environment settings, whereas the command ng build ?prod will build the project using the production environment settings.

- **browserlist:** This file is used by autoprefixer that adjusts the CSS to support a list of defined browsers.
- **favicon.ico:** This file specifies a small icon that appears next to the browser tab of a website.
- **index.html:** This is the entry file which holds the high level container for the angular application.
- **karma.config.js:** This file specifies the config file for the Karma Test Runner, Karma has been developed by the AngularJS team which can run tests for both AngularJS and Angular 2+
- **main.ts:** As defined in angular.json file, this is the main ts file that will first run. This file bootstraps (starts) the AppModule from app.module.ts , and it can be used to define global configurations.
- **polyfills.ts:** This file is a set of code that can be used to provide compatibility support for older browsers. Angular 7 code is written mainly in ES6+ language specifications which is getting more adopted in front-end development, so since not all browsers support the full ES6+ specifications, pollyfills can be used to cover whatever feature missing from a given browser.
- **styles.css:/** This is a global css file which is used by the angular application.
- **tests.ts:** This is the main test file that the Angular CLI command ng test will use to traverse all the unit tests within the application and run them.
- **tsconfig.json:** This is a typescript compiler configuration file.
- **tsconfig.app.json:** This is used to override the tsconfig.json file with app specific configurations.
- **tsconfig.spec.json:** This overrides the tsconfig.json file with app specific unit test configurations.

All Angular CLI commands

Command	Alias	Description
add		It is used to add support for an external library to your project.
build	b	It compiles an Angular app into an output directory named dist/ at the given output path. Must be executed from within a workspace directory.
config		It retrieves or sets Angular configuration values in the angular.json file for the workspace.
doc	d	It opens the official Angular documentation (angular.io) in a browser, and searches for a given keyword.
e2e	e	It builds and serves an Angular app, then runs end-to-end tests using Protractor.
generate	g	It generates and/or modifies files based on a schematic.
help		It provides a list of available commands and their short descriptions.
lint	l	It is used to run linting tools on Angular app code in a given project folder.
new	n	It creates a new workspace and an initial Angular app.
run		It runs an Architect target with an optional custom builder configuration defined in your project.
serve	s	It builds and serves your app, rebuilding on file changes.
test	t	It runs unit tests in a project.
update		It updates your application and its dependencies. See https://update.angular.io/
version	v	It utputs Angular CLI version.
xi18n		It extracts i18n messages from source code.

Angular Components

Components are the key features of Angular. The whole application is built by using different components.
The core idea behind Angular is to build components. They make your complex application into reusable parts which you can reuse very easily.

Creating component with CLI
Syntax
1.ng generate component component_name
2.Or
3.ng g c component_name

Open Command prompt and stop **ng serve** command if it is running on the browser.
Type **ng generate component server2** to create a new component named server2.
You can also use a shortcut **ng g c server2** to do the same task.