Angular Tutorial (part I : setup)

This article is meant to give a brief orientation on Angular 2+. Part I will show how to set up Angular, and generate a trivial application using the command line interface. Part II will cover building a service class and generating a component. It does not assume any previous knowledge of the framework.

Angular is one of the most popular front-end web frameworks in use today. It is developed by Google and is great for building professional web applications because it allows for modular design and easy communication with a server. When beginning with Angular, it is important to understand two things.

1. Versions (Angular 2+ vs Angular.js)

Angular version can be put into two categories: AngularJS (aka Angular 1) and Angular 2+. They are not cross-compatible.

AngularJS was first developed in 2010 [1]. It is still in use today but is currently being phased out. The last release was in 2018 and they are discontinuing support in 2021 [2].

Angular 2+ refers to versions of Angular 2 or greater. Angular 2 was first released in 2016 and is a total rewrite of Angular 1/AngularJS [3]. This tutorial will use Angular 7, but it should work for any version 2 or greater.

2. Typescript

Angular2+ uses typescript, a transpiled language that produces JavaScript. It is a superset of Javascript ECMAScript 2015 [4]. It allows for greater specification by defining variables and return types. For more information about typescript, visit the website https://www.typescriptlang.org/index.html .

---

[1] https://github.com/angular/angular.js/releases?after=v0.9.4

[2] https://blog.angular.io/stable-angularjs-and-long-term-support-7e077635ee9c

[3] https://blog.angularjs.org/2016/09/angular2-final.html

[4] https://angular.io/guide/upgrade

SETUP

Installation is straightforward. To begin, you need to install NODE.js with npm. If you don't already have it, follow the instructions provided at the node website: https://nodejs.org/en/ .

Next, you need to install Typescript. This is the language that Angular uses to handle its javascript logic. To install, open your console and type:

```
npm install -g typescript
```

Now, you can install the Angular Command Line Interface (CLI). This is an essential tool which allows you to host your app and generate projects, components and service classes. For more info, see https://angular.io/cli. To install the Angular CLI, type:

```
npm install -g @angular/cli
```

Finally, it is time to create an Angular App. Begin by using the CLI to generate a project. Do this by typing:

```
ng new firstAngular
```

This creates a trivial application which serves as a starting point for your project. Now, test it by navigating to your directory…

```
cd firstAngular
```

and type:

```
ng serve
```

After a few moments, the CLI should load the app and host it on port 4200. To confirm, go to your browser and type localhost:4200 into the url bar. This will load your application. It will contain some links to the Angular website, including a tutorial. The tutorial covers a lot of ground, so I created Part II of my tutorial to serve as a bridge.

As you work on your application, keep your browser open and keep ng serve running. It will continue to update as you code, so you'll be able to see your modifications in real time. This is a similar functionality to nodemon, Brackets live mode, or Flask's debug mode.