



Front-End Essentials

Lab Guides

Document Code	25e-BM/HR/HDCV/FSOFT
Version	1.1
Effective Date	8/1/2019


Hanoi, 04/2019

RECORD OF CHANGES

No	Effective Date	Change Description	Reason	Reviewer	Approver
1	25/Jun/2018	Create a new Lab	Create new	DieuNT1	VinhNV
2	01/May/2019	Update Fsoft Template	Update	DieuNT1	VinhNV
3	01/Aug/2019	Update for relase	Release	DieuNT1	VinhNV

Contents

Day 2. Unit 2 – Angular Overview.....	4
Objectives.....	4
Technical Requirements:.....	4
Specifications.....	4
Guidelines.....	4
Step 1: Install NodeJS.....	4
Step 2: Install NPM.....	4
Step 3: Install Angular CLI.....	4
Step 4: Create Angular app using Angular CLI.....	4
Step 5: Start Angular app.....	4
Step 6: Create new Component.....	4
Step 7: Use newly created Component.....	4
Step 8: Data Binding using Interpolation.....	4
Step 9: Data Binding using Property Binding.....	4
Step 10: Data Binding using Event Binding.....	5
Step 11: Data Binding using 2 Way data Binding.....	5
Step 12: Using ngStyle.....	5
Step 13: Using ngClass.....	5
Step 14: Using ngFor.....	5
Step 15: Using ngIf.....	5
Step 16: Create sample Animation.....	5
Step 17: Review.....	5

	CODE:	FEFW_Angular.M.L0201
	TYPE:	Medium
	LOC:	200
	DURATION:	90 MINUTES

Day 2. Unit 2 – Angular Overview

Objectives

- Understand the overview architecture of Angular
- Understand what is Single Page Application
- Able to create angular project using Angular CLI
- Understand basic concepts in Angular: Component, Directive
- Understand 4 ways of Data Binding
- Able to use built-in Angular Directive

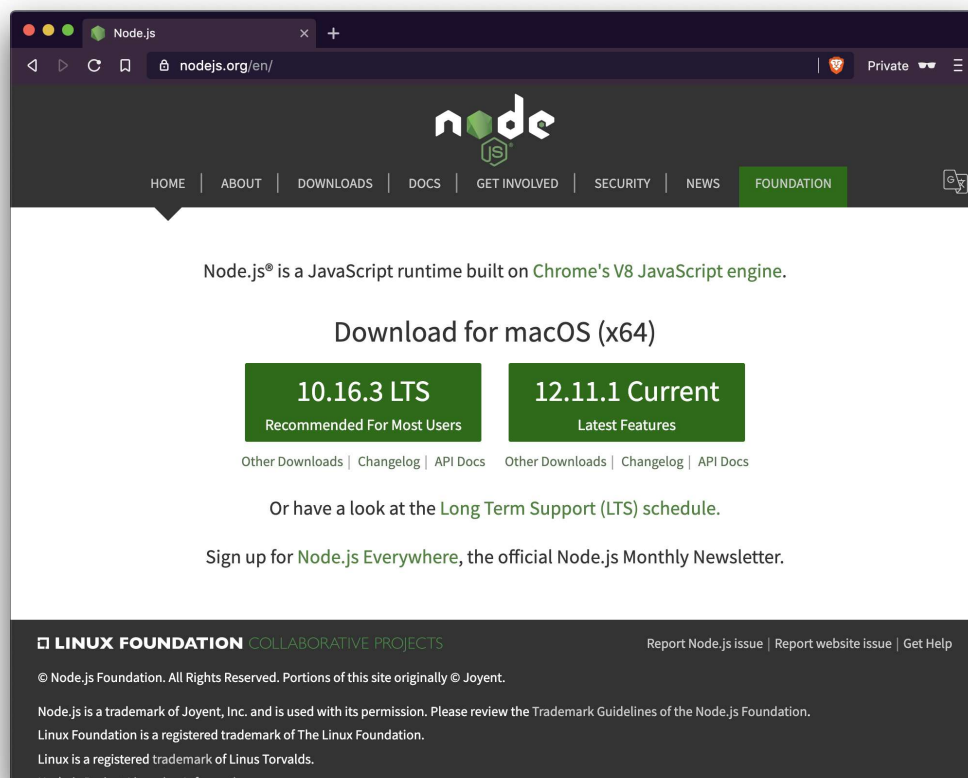
Specifications

Create a sample Angular application.

Guidelines

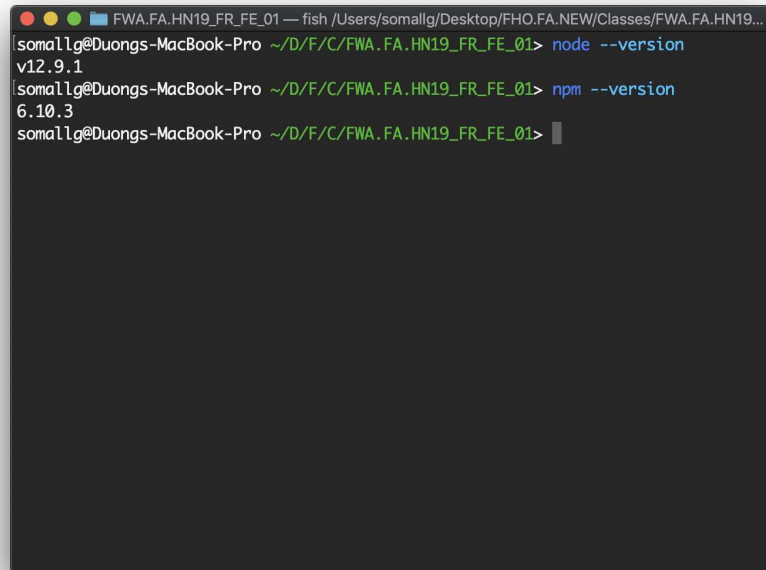
Step 1: Install NodeJS

Navigate to <https://nodejs.org/en/> then download and install NodeJS latest version corresponding to your OS.



Step 2: Verify NodeJS

Open Terminal and verify that NodeJS is properly installed. If yes, you should have **node** and **npm** command line like figure below:

A terminal window on a macOS system. The prompt is 'somallg@Duongs-MacBook-Pro ~'. The user enters 'node --version' and the output is 'v12.9.1'. Then the user enters 'npm --version' and the output is '6.10.3'.

```
FWA.FA.HN19_FR_FE_01 — fish /Users/somallg/Desktop/FHO.FA.NEW/Classes/FWA.FA.HN19...
[somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01] node --version
v12.9.1
[somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01] npm --version
6.10.3
[somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01]
```

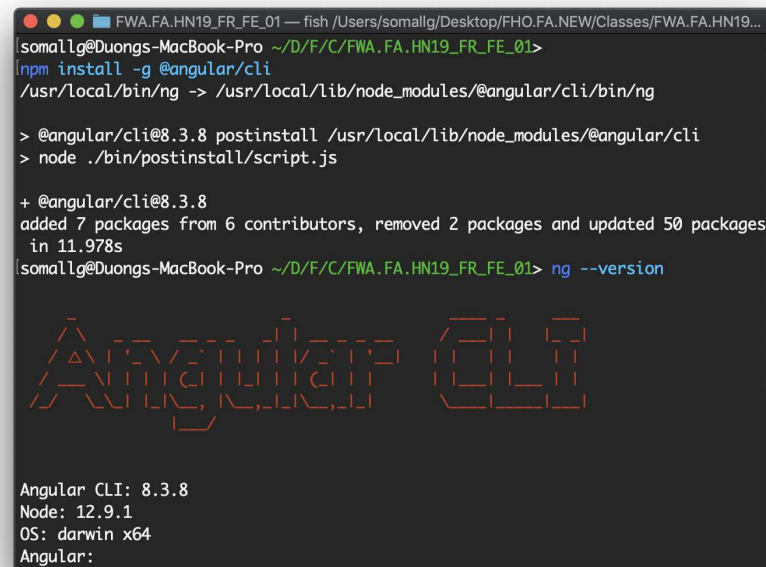
Step 3: Install Angular CLI

Install Angular CLI by typing the following command line:

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

Verify that Angular CLI is installed by typing:

```
1. ng --version
```

A terminal window showing the installation of Angular CLI. The user enters 'npm install -g @angular/cli'. The output shows the installation progress, including downloading packages and postinstall scripts. After installation, the user enters 'ng --version' and the output shows 'Angular CLI: 8.3.8', 'Node: 12.9.1', 'OS: darwin x64', and 'Angular:'.

```
FWA.FA.HN19_FR_FE_01 — fish /Users/somallg/Desktop/FHO.FA.NEW/Classes/FWA.FA.HN19...
[somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01] npm install -g @angular/cli
/usr/local/bin/ng -> /usr/local/lib/node_modules/@angular/cli/bin/ng

> @angular/cli@8.3.8 postinstall /usr/local/lib/node_modules/@angular/cli
> node ./bin/postinstall/script.js

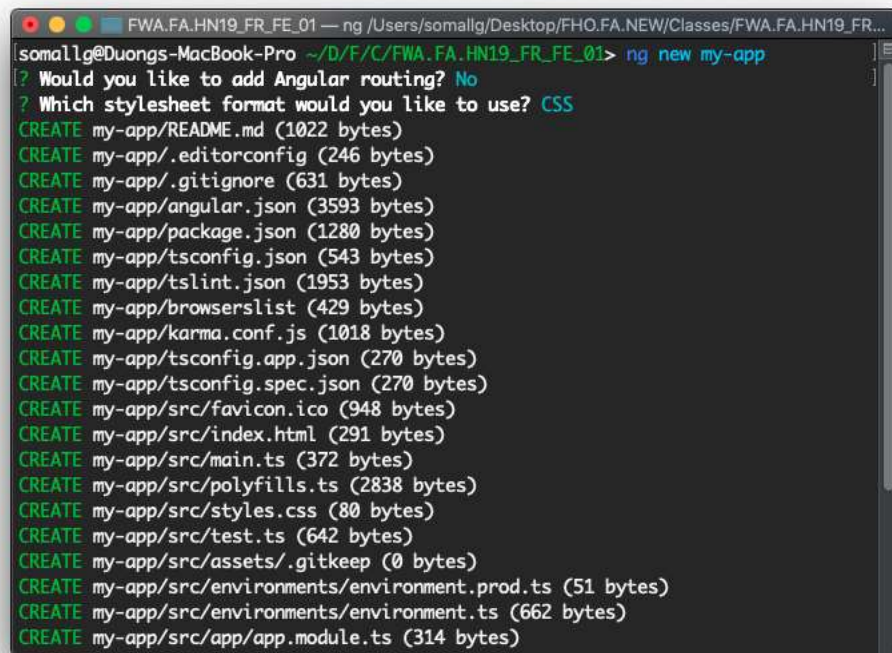
+ @angular/cli@8.3.8
added 7 packages from 6 contributors, removed 2 packages and updated 50 packages
in 11.978s
[somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01] ng --version

Angular CLI: 8.3.8
Node: 12.9.1
OS: darwin x64
Angular:
```

Step 4: Create Angular app using Angular CLI

Use ng command line to create new Angular app:

```
1. ng new my-app
```



```
FWA.FA.HN19_FR_FE_01 — ng /Users/somallg/Desktop/FHO.FA.NEW/Classes/FWA.FA.HN19_FR...
somallg@Duongs-MacBook-Pro ~/D/F/C/FWA.FA.HN19_FR_FE_01> ng new my-app
? Would you like to add Angular routing? No
? Which stylesheet format would you like to use? CSS
CREATE my-app/README.md (1022 bytes)
CREATE my-app/.editorconfig (246 bytes)
CREATE my-app/.gitignore (631 bytes)
CREATE my-app/angular.json (3593 bytes)
CREATE my-app/package.json (1280 bytes)
CREATE my-app/tsconfig.json (543 bytes)
CREATE my-app/tslint.json (1953 bytes)
CREATE my-app/browserslist (429 bytes)
CREATE my-app/karma.conf.js (1018 bytes)
CREATE my-app/tsconfig.app.json (270 bytes)
CREATE my-app/tsconfig.spec.json (270 bytes)
CREATE my-app/src/favicon.ico (948 bytes)
CREATE my-app/src/index.html (291 bytes)
CREATE my-app/src/main.ts (372 bytes)
CREATE my-app/src/polyfills.ts (2838 bytes)
CREATE my-app/src/styles.css (80 bytes)
CREATE my-app/src/test.ts (642 bytes)
CREATE my-app/src/assets/.gitkeep (0 bytes)
CREATE my-app/src/environments/environment.prod.ts (51 bytes)
CREATE my-app/src/environments/environment.ts (662 bytes)
CREATE my-app/src/app/app.module.ts (314 bytes)
```

Step 5: Start Angular app

Change working directory to newly created project:

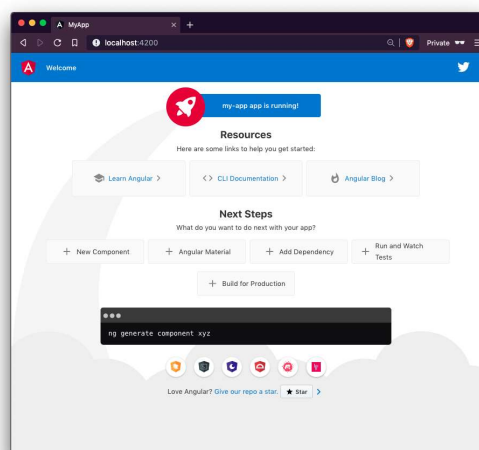
```
1. cd my-app/
```

and type command below:

```
1. ng serve
```

Angular will build the app and host it in port **4200** (default port). Open Chrome browser and navigate to <http://localhost:4200>

You should have the similar result like figure below:

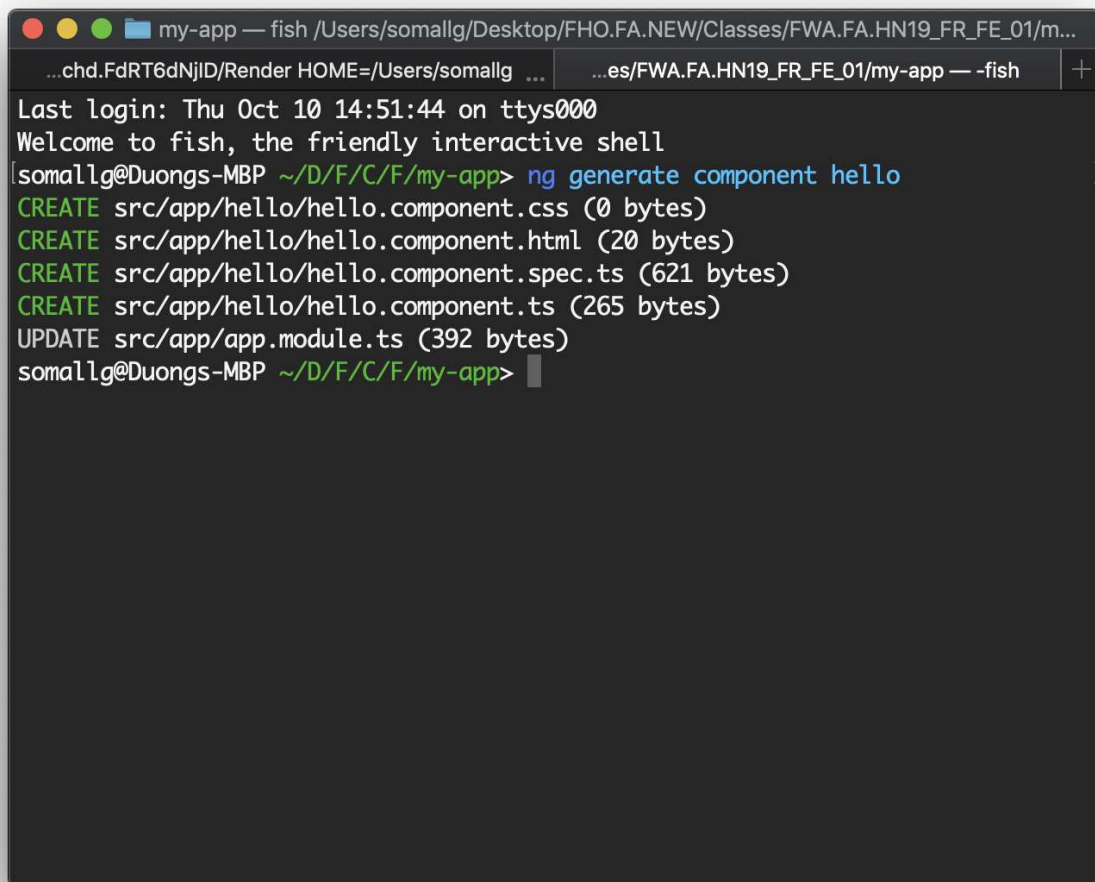


Step 6: Create new Component

Create a new Component by using the following command:

```
1. ng generate component hello
```

You should see similar output:



```
my-app — fish /Users/somallg/Desktop/FHO.FA.NEW/Classes/FWA.FA.HN19_FR_FE_01/m...
...chd.FdRT6dNjID/Render HOME=/Users/somallg ...  ...es/FWA.FA.HN19_FR_FE_01/my-app — -fish +
Last login: Thu Oct 10 14:51:44 on ttys000
Welcome to fish, the friendly interactive shell
[somallg@Duongs-MBP ~/D/F/C/F/my-app] ng generate component hello
CREATE src/app/hello/hello.component.css (0 bytes)
CREATE src/app/hello/hello.component.html (20 bytes)
CREATE src/app/hello/hello.component.spec.ts (621 bytes)
CREATE src/app/hello/hello.component.ts (265 bytes)
UPDATE src/app/app.module.ts (392 bytes)
somallg@Duongs-MBP ~/D/F/C/F/my-app
```

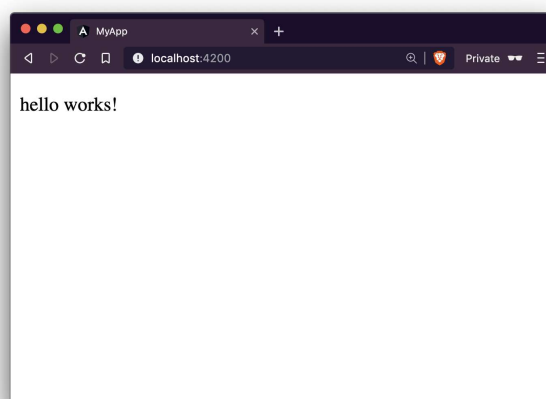
Step 7: Use newly created Component

Open Visual Studio Code, and open **src/app/app.component.html** file.

Remove all the content and put in the following content:

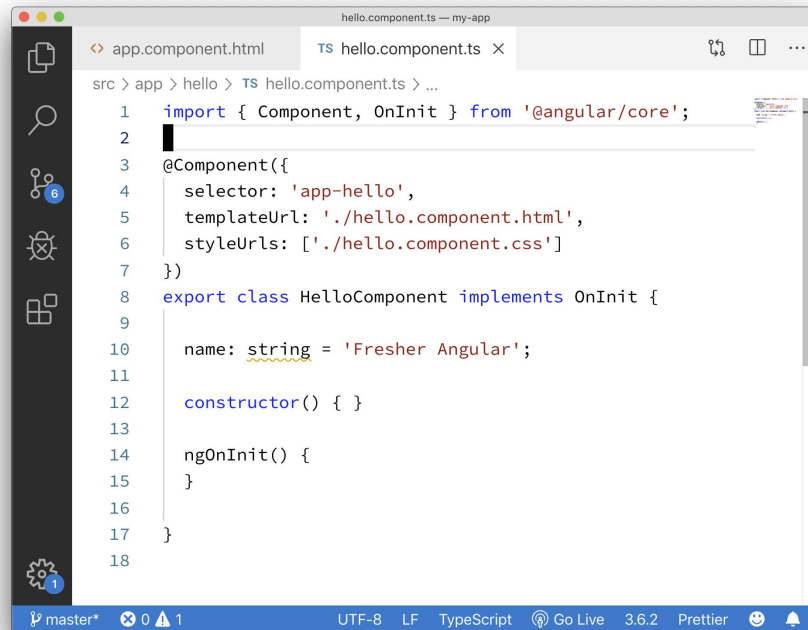
```
1. <app-hello></app-hello>
```

Save the file, check Chrome browser, you should see something like figure below:



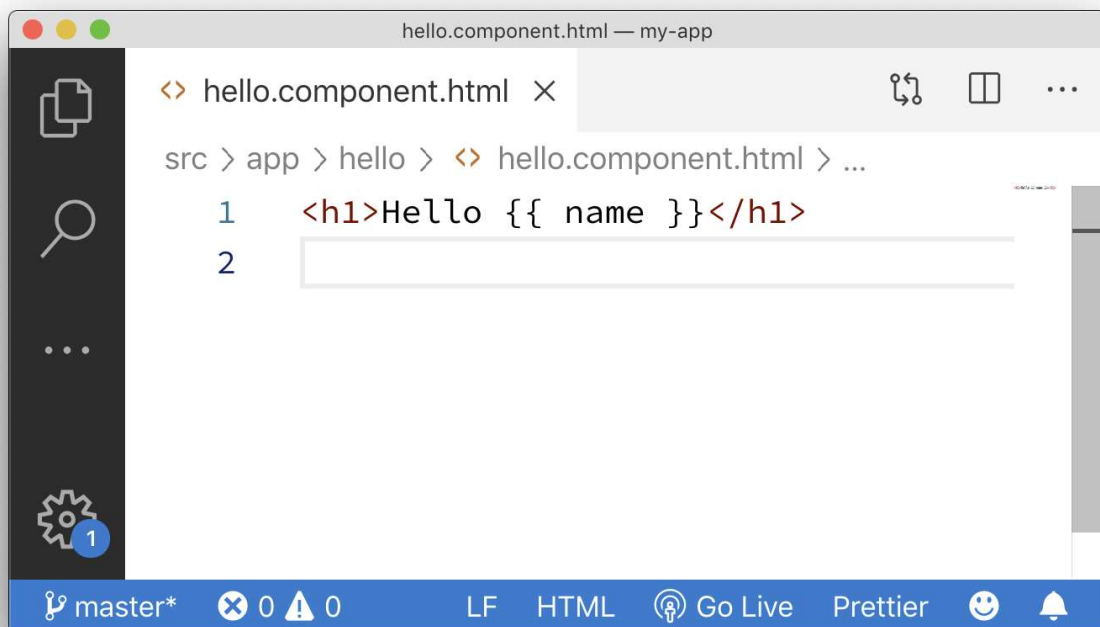
Step 8: Data Binding using Interpolation

Open **src/app/hello/hello.component.ts** file, inside HelloComponent class declare a new field called name with value 'Fresher Angular'



```
1 import { Component, OnInit } from '@angular/core';
2
3 @Component({
4   selector: 'app-hello',
5   templateUrl: './hello.component.html',
6   styleUrls: ['./hello.component.css']
7 })
8 export class HelloComponent implements OnInit {
9
10   name: string = 'Fresher Angular';
11
12   constructor() { }
13
14   ngOnInit() {
15   }
16
17 }
18
```

We want to display value of variable name into HTML, open **hello.component.html** file and type the below content:



```
1 <h1>Hello {{ name }}</h1>
2
```

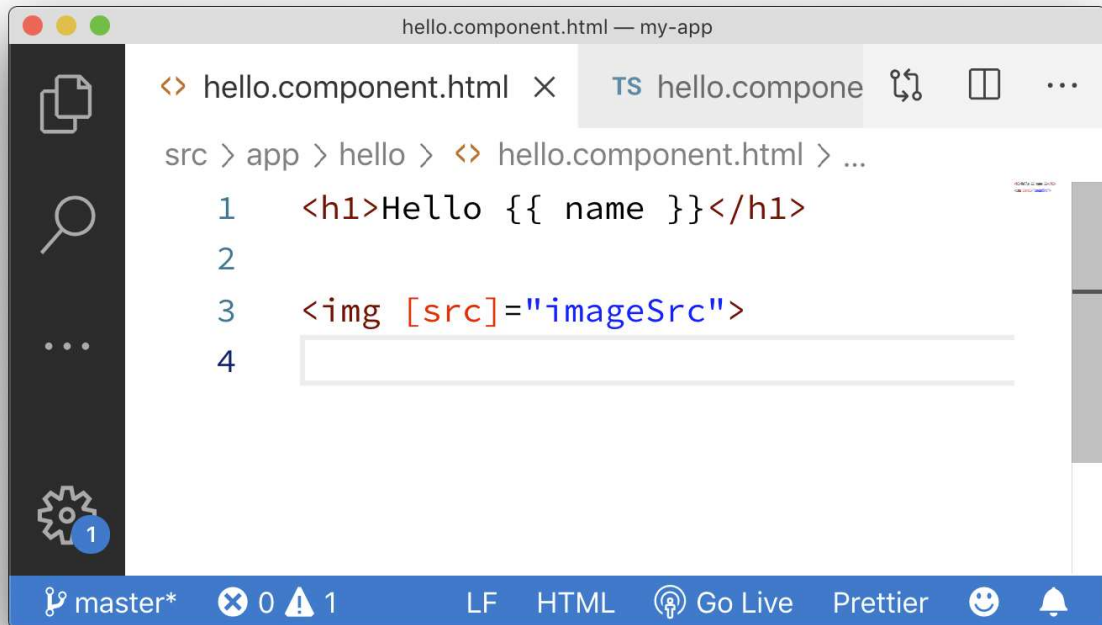
Check Chrome Browser, you should see **Hello Fresher Angular**

Step 9: Data Binding using Property Binding

Same like previous Step, now you must declare a field named **imageSrc** with value:

<http://placekitten.com/200/300>

Then in **hello.component.html** file, create a new **img** tag like so:



```
hello.component.html  x  TS hello.compone  ↺  📄  ...
src > app > hello > <> hello.component.html > ...
1  <h1>Hello {{ name }}</h1>
2
3  <img [src]="imageSrc">
4  
```

Check Chrome browser, you should see an image of cat.

Step 10: Data Binding using Event Binding**Step 11: Data Binding using 2 Way data Binding****Step 12: Using ngStyle****Step 13: Using ngClass****Step 14: Using ngFor****Step 15: Using ngIf****Step 16: Create sample Animation****Step 17: Review**