

Web Programming

Angular

Eva Oliveira

2022

Angular

is



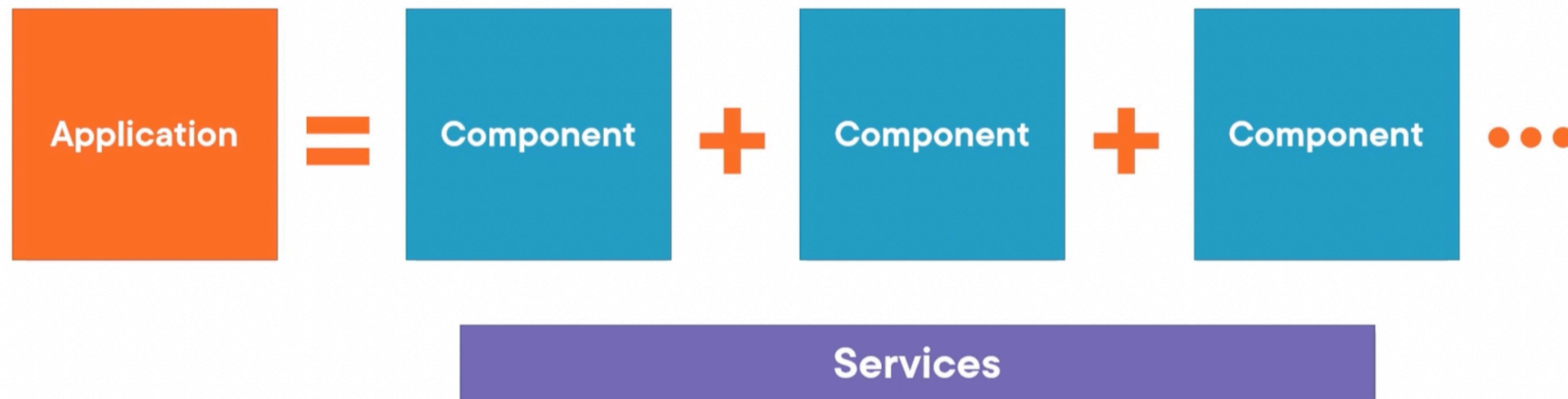
**A JavaScript framework
For building client-side applications
Using HTML, CSS and TypeScript**

Why Angular

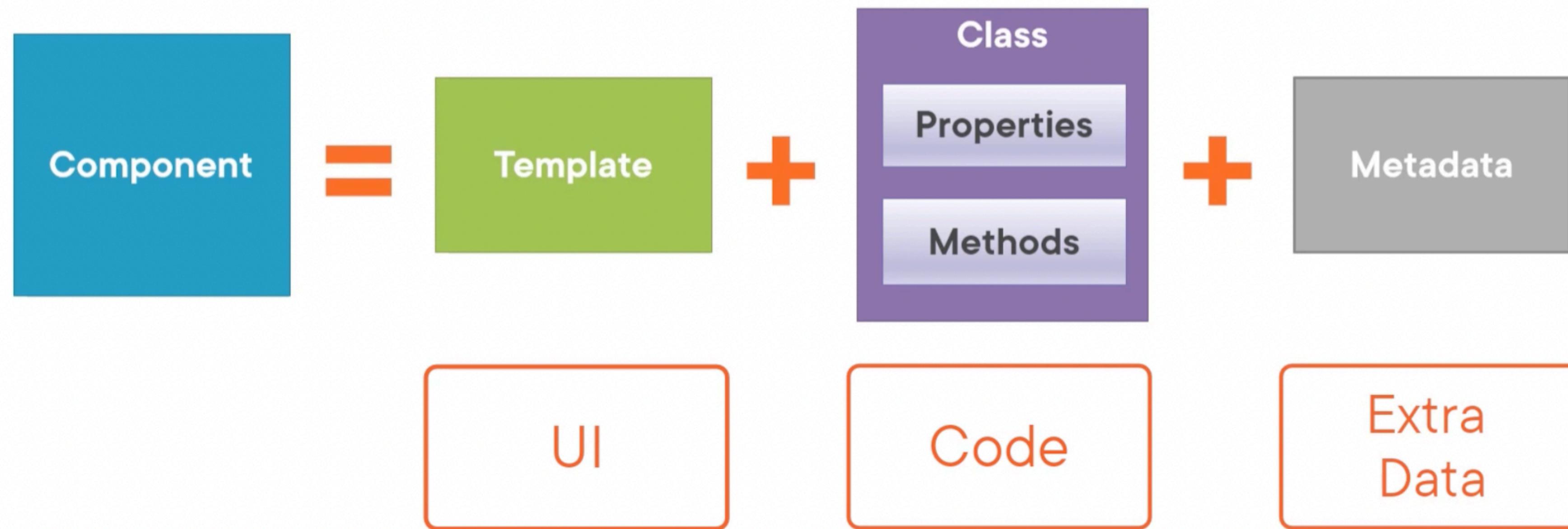
The diagram consists of four colored boxes (blue, black, red, green) arranged horizontally, each containing a white icon above a descriptive text. Above the boxes is a horizontal row of four icons: a blue file icon with '<>' symbols, a grey chain link icon, a red modular structure icon, and a green cloud with double-headed arrows icon. A small circular arrow icon is located at the bottom right of the green box.

- Expressive HTML**
- Powerful Data Binding**
- Modular By Design**
- Built-in Back-End Integration**

Anatomy of an Angular Application



What is a component?



Demo

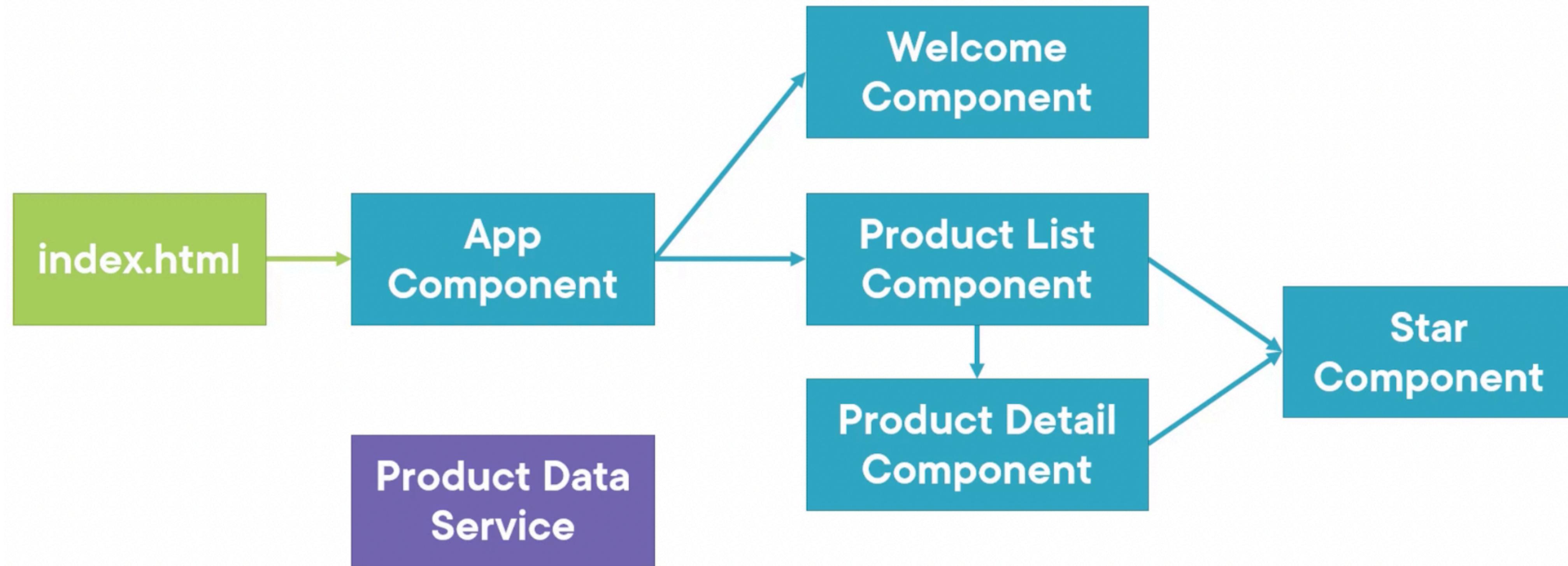
Our app



Products Backend

[visit our store](#)

Sample Application Architecture



TypeScript
is the programming language
we use
when building
Angular applications

TypeScript



- Open-source language**
- Superset of JavaScript**
- Transpiles to plain JavaScript**
- Strongly typed**
- Class-based object-orientation**

TypeScript

Types

```
let total:number = 0
```

```
let isTall:boolean = true
```

```
let name:string = "Eva"
```

```
let list1:number[] = [1,2,3]
```

```
let random:any = 10
```

```
let obj: any = { x: 0 };
```

TypeScript

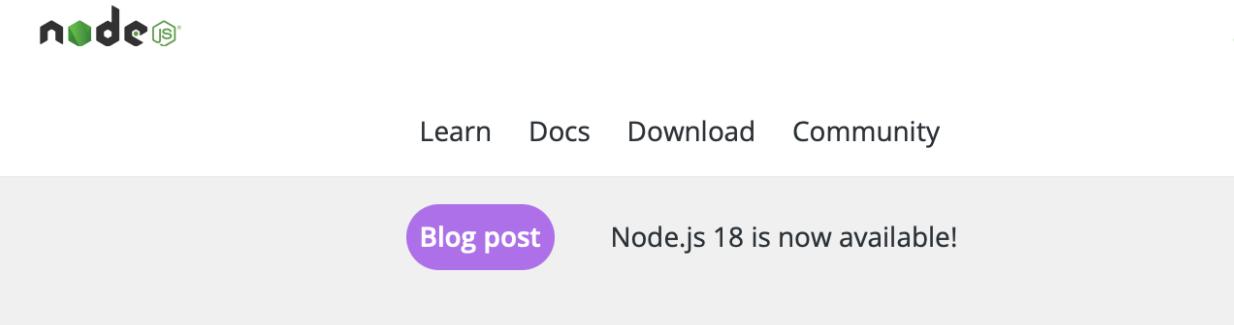
Types

```
// Parameter type annotation
function greet(name: string): void{
    console.log("Hello, " +
name.toUpperCase() + "!!");
}

// Parameter type annotation
function getFavoriteNumber(): number {
    return 26;
}
```

```
// The parameter's type annotation is an
object type
function printCoord(pt: { x: number; y:
number }) {
    console.log("The coordinate's x value is " + pt.x);
    console.log("The coordinate's y value is " + pt.y);
}
```

Installations



Run JavaScript Everywhere.

Node.js is a free, open-sourced, cross-platform JavaScript run-time environment that lets developers write command line tools and server-side scripts outside a browser.

[Download Node \(LTS\)](#)

[Learn Node](#)

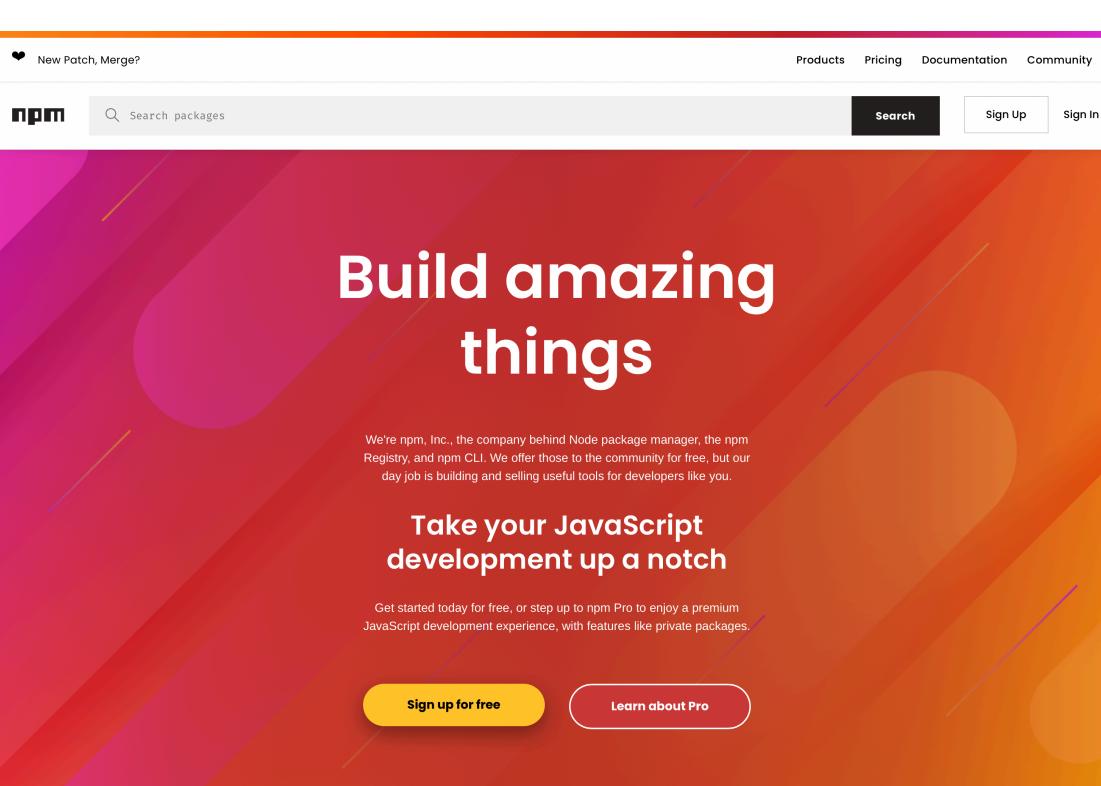
Version 16.14.2 - [Get Current](#)

Install the Angular CLI

You use the Angular CLI to create projects, generate application and library code, and perform a variety of ongoing development tasks such as testing, bundling, and deployment.

To install the Angular CLI, open a terminal window and run the following command:

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



What else do we need?

Angular



Angular

- Framework and libraries

Angular CLI

- Command line interface for Angular

TypeScript

- Programming language

Testing tools, linters, etc.

Package.json

All dependencies

```
12 "dependencies": {
13   "@angular/animations": "~12.0.0",
14   "@angular/common": "~12.0.0",
15   "@angular/compiler": "~12.0.0",
16   "@angular/core": "~12.0.0",
17   "@angular/forms": "~12.0.0",
18   "@angular/platform-browser": "~12.0.0",
19   "@angular/platform-browser-dynamic": "~12.0.0",
20   "@angular/router": "~12.0.0",
21   "rxjs": "~6.6.0",
22   "tslib": "^2.1.0",
23   "zone.js": "~0.11.4"
24 },
25 "devDependencies": {
26   "@angular-devkit/build-angular": "~12.0.0",
27   "@angular/cli": "~12.0.0",
28   "@angular/compiler-cli": "~12.0.0",
29   "@types/jasmine": "~3.6.0",
30   "@types/node": "^12.11.1",
31   "jasmine-core": "~3.7.0",
32   "karma": "~6.3.0",
33   "karma-chrome-launcher": "~3.1.0",
34   "karma-coverage": "~2.0.3",
35   "karma-jasmine": "~4.0.0",
36   "karma-jasmine-html-reporter": "^1.5.0",
37   "typescript": "~4.2.3"
38 }
```

When setting up existing angular code

#1

Navigate down to the project folder

The project folder contains the `package.json` file

#2

Run: `npm install`

To install the packages defined in the `package.json` file

#3

Run: `npm start`

To start the installed Angular application

Go to moodle : week 7 initial files

 src	 >
 angular.json	
 karma.conf.js	
 package-lock.json	
 package.json	
 README.md	
 tsconfig.app.json	
 tsconfig.json	
 tsconfig.spec.json	

Boilerplate Files

By convention all files are
under **src** folder

```
{  
  "name": "firstapp",  
  "version": "0.0.0",  
  ▷ Debug  
  "scripts": {  
    "ng": "ng",  
    "start": "ng serve -o",  
    "build": "ng build",  
    "watch": "ng build --watch --configuration development",  
    "test": "ng test"  
  },  
  "private": true,  
  "dependencies": {  
    "@angular/animations": "~12.0.0",  
    "@angular/common": "~12.0.0",  
    "@angular/compiler": "~12.0.0",  
    "@angular/core": "~12.0.0",  
    "@angular/forms": "~12.0.0",  
    "@angular/platform-browser": "~12.0.0",  
    "@angular/platform-browser-dynamic": "~12.0.0",  
    "@angular/router": "~12.0.0",  
    "bootstrap": "^5.0.1",  
    "font-awesome": "^4.7.0",  
    "rxjs": "~6.6.0",  
    "tslib": "^2.1.0",  
    "zone.js": "~0.11.4"  
  },  
  "devDependencies": {  
    "@angular-devkit/build-angular": "~12.0.0",  
    "@angular/cli": "~12.0.0",  
    "@angular/compiler-cli": "~12.0.0",  
    "@types/jasmine": "~3.6.0",  
    "@types/node": "^12.11.1",  
    "jasmine-core": "~3.7.0",  
    "karma": "~6.3.0",  
    "karma-chrome-launcher": "~3.1.0",  
    "karma-coverage": "~2.0.3",  
    "karma-jasmine": "~4.0.0",  
    "karma-jasmine-html-reporter": "^1.5.0",  
    "typescript": "~4.2.3"  
  }  
}
```

package.json

List of all the app dependencies

Scripts can be runned using npm

To install the app we need to run the **npm install** in the folder of the package.json

Then a folder `node_modules` is created with all the dependencies

References

- Angular site: <https://angular.io>
- Typescript Playground: <https://www.typescriptlang.org/play>