第一节：借助命令行创建angular2 app

概要：

Angular-CLI ,即**C**ommand **L**ine **I**nterface简写，是一个官方推荐的命令行界面工具，angular官方出品。

Angular-CLI可以创建一个项目,添加文件,并执行各种持续的开发任务,如测试、发布等。

本节内容如下:

1. 搭建开发环境
2. 创建一个新项目（带脚手架的单页面应用程序）
3. 运行应用程序
4. 编辑应用程序
5. 发布应用
6. 角色架简介
7. Angular2 系统架构

1. 搭建开发环境

在线尝鲜（plunker）：

<https://embed.plnkr.co/?show=preview&show=app%2Fapp.component.ts>

1. 开发IDE

强烈推荐微软出品 VSC （Visual Studio Code）

另外较出名的几款：Webstorm、Atom、sublime

1. 安装Node.js 和 npm

版本要求：Node.js >6.9.x npm >4.2.x

查看Node、npm版本方法：node -v npm -v

如果之前安装过NODE及NPM，需要更新，更新node及npm步骤

1. 卸载nodejs安装最新版
2. 更新NPM

npm update -g

1. 全局安装/更新Angular-CLI

npm install -g @angular/cli

（~~教程编写时版本为1.0.0-beta.24， 使用~~ ~~npm install -g angular-cli~~ ）

更新Angular-CLI

如果用之前的版本创建的项目需更新按下面方法：

清理公共区域

npm uninstall -g angular-cli @angular/cli  
npm cache clean  
npm install -g @angular/cli@latest

清理项目区域（转到项目路径）  
rmdir /s/q node\_modules dist

npm install --save-dev @angular/cli@latest  
npm install

Angular-CLI github:

<https://github.com/angular/angular-cli>

1. 创建一个新项目（带脚手架的单页面应用程序）

打开一个CMD窗口或终端窗口

在VSC中终端窗口中（ctrl+`）

ng new my-app

执行后等待数分钟，这段时间主要用于给应用安装NPM包

1. 运行应用程序
   1. 转到应用目录

cd my-app

1. 运行程序

npm start

或者

ng serve

默认将以4200端口启动，通过--port 参数可指定端口启动

如下将以4000端口启动

ng serve --port 4000

1. 打开浏览器 打开http://localhost:4200/

将在浏览器中显示：

**app works!**

1. 编辑应用程序

找到 ./src/app/app.component.ts 修改 title值为：我的单页应用-ng2app

无需重新运行应用，浏览器将自动刷新

尝试修改 ./src/app/app.component.css 可观察样式变化

1. 发布应用

ng build

执行命令后将自动编译到dist/目录

1. 脚手架简介
   1. 根路径

目录结构如下：

*my-app*

*e2e*

*app.e2e-spec.ts*

*app.po.ts*

*tsconfig.json*

*node\_modules/...*

*src/...*

*.editorconfig*

*.gitignore*

*angular-cli.json*

*karma.conf.js*

*package.json*

*protractor.conf.js*

*README.md*

*tslint.json*

用途描述:

|  |  |
| --- | --- |
| 文件/文件夹 | 用途 |
| e2e/\* | 端到端测试相关文件夹 |
| node\_modules/... | Node.js包存放文件夹,包括了在package.json中的所有三方包 |
| Src/… | 应用源码目录，详细见下一小节 |
| .editorconfig | 编辑器配置，大部分编辑器支持.editorconfig,但VSC默认不支持需要安装扩展，查看官网 <http://editorconfig.org/> 可了解更多 |
| .gitignore | git忽略配置文件，用于源代码管理，确保配置的项不加入源码管理 |
| angular-cli.json | Angular-CLI配置文件。 |
| karma.conf.js | 单元测试配置文件。 |
| package.json | Npm 配置文件。 |
| protractor.conf.js | 端到端测试配置。当运行ng e2e是启用，端到端采用的是 Protractor，可打开 <http://www.protractortest.org/> 查看详细 |
| README.md | 项目说明文件。 |
| tslint.json | 代码检测工具配置文件（TSLint\codelyzer）,运行ng lint将使用此配置，有助于团队代码风格一致。 |

1. src文件夹

APP源码文件夹，包括了所有的模块、组件、模板、样式等

目录结构如下：

*src*

*app*

*app.component.css*

*app.component.html*

*app.component.spec.ts*

*app.component.ts*

*app.module.ts*

*assets*

*.gitkeep*

*environments*

*environment.prod.ts*

*environment.ts*

*favicon.ico*

*index.html*

*main.ts*

*polyfills.ts*

*styles.css*

*test.ts*

*tsconfig.json*

用途描述：

|  |  |
| --- | --- |
| 文件/文件夹 | 用途 |
| app/app.component.{ts,html,css,spec.ts} | 定义了一个根组件AppComponent，包括了HTML模板,CSS样式表和一个单元测试。 |
| app/app.module.ts | 定义了一个根模块AppModule,主要用于告知angular如何去组织组件，一个angular-cli创建的新应用只声明了AppComponent，随着需求业务的实现，在这将声明更多组件及其他。 |
| assets/\* | 一个源文件夹，可以把图片和其他任何需要复制到发布后的应用中的特定文件夹。 |
| environments/\* | 环境配置文件夹，包含了开发环境和生产环境的配置 |
| favicon.ico | 浏览器中地址栏中显示的 LOGO |
| index.html | 单页应用页，主页，当我们需要引入JS或CSS时，传统的做法我们将手工修改此页，但在CLI下，在编译时它将自动帮助我们创建 |
| main.ts | 应用程序的入口 |
| polyfills.ts | 不同的浏览器有不同级别的web标准的支持。Polyfills帮助规范化这些差异，但IE低版本也是无法支持的     |  |  |  |  |  | | --- | --- | --- | --- | --- | | Chrome | Firefox | Edge | IE | Safari | | latest | latest | 14 | 11 | 10 | |  |  | 13 | 10 | 9 | |  |  |  | 9 | 8 | |  |  |  |  | 7 | |
| styles.css | 公共样式文件 |
| test.ts | 单元测试的主要入口点。  一些自定义的配置,不明白怎么配置,但我们不需要编辑。 |
| tsconfig.json | TypeScript 编译配置. |

1. VSC插件推荐
   1. Angular Files

可以让VSC支持右键菜单中使用angular-cli命令

1. Bootstrap 4 & Font awesome snippets
2. NgBootstrap Snippets
3. VSCode Simpler Icons with angular icons

更改VSC中 angular文件图标外观

1. Angular2 系统架构

摘抄官网图如下：



首先Angular2是一个完整的单页应用开发框架。

很多人拿它跟React比，相比来说，React是一个基础框架，更像是一个库，你需要很多第三方的库才能方便的开发一个完整的应用。而Angular2则提供了很多组件，或者叫库，比如Directive(指令)、组件框架、模板、依赖注入、绑定、路由等，在这些库的帮助下，你更多的只需要关注具体业务的开发。