

Angular02

WEB 三大框架

框架名	出品方	适用范围	手机端
vue	开源社区	中小型网站+手机端	mintUI等, 纯H5风格. 制作的手机端样式的网页 UniApp 开发的程序可以编译成手机安装软件
react	Facebook	能做网站,但偏重手机端	ReactNative框架, 做出的程序可以编译成真正的原生App 与原生对比, 效率一样但是可定制性不强
angular	Google	大型网站+手机端	ionic框架

本阶段的安排

- angular: 6天
- react: 7天
- python: 5天
 - python火, python和js语法十分相似, 能做服务器
- vue项目: 2天
 - 使用git上线 vue项目 到 新浪云

Angular

开发时, 需要通过脚手架来 生成 项目包.

要求:

- node版本: 10.9 以上
- npm下载脚手架, 默认地址是外网的: `http://registry.npm.org`
 - 感谢阿里, 外网在中国访问非常慢, 甚至无法下载. 所以阿里免费提供了国内的服务器, 实时拉取外网的所有资源到国内服务器
 - 把下载地址切换成 中国服务器 即可提高下载速度

```
npm config set registry https://registry.npm.taobao.org/
```

安装angular脚手架

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

-g: 代表全局安装

常见报错:

达内提供的电脑 登录账号的权限给的很低: 怕不小心把系统文件删除掉.

登录用户的权限低, 导致启动的cmd命令行工具 无法进行替换操作: 先删->粘贴.

只能采用手动方式进行删除 已下载的包

npm下载的所有脚手架存放在固定的位置:

C:\Users\web\AppData\Roaming\npm

C:\Users\web\AppData\Roaming\npm\node_modules

删除上方文件夹中的 与 angular 和 ng 有关的即可!

注意:

- * 不同电脑用户名不同, 例如上方的 \Users\web; 具体用户名可以通过打开cmd 来查看默认路径确定
- * AppData是隐藏文件夹, 必须开启 查看隐藏文件 才能看到

生成项目包

- 注意项目生成的位置: 项目一定是在 cmd 打开的那个目录下生成
- 生成命令

普通写法: ng new 包名

简化写法: ng n 包名

生成过程中两个询问:

- 要不要路由系统? y/N 大写字母是默认值, 代表No 不使用. 回车即可!
- 使用哪种样式来书写? 选css

项目启动, 必须在项目的目录下执行命令行

基础写法: ng serve

简化写法: ng s

自动启动浏览器: ng serve --open

自动启动浏览器的简化写法: ng s -o

项目默认路径: `http://localhost:4200/`

- localhost: 域名. 计算机在网络中的名字, localhost和127.0.0.1 都是代表当前计算机, 即 我 字
- 4200: 端口号. 计算机上程序的唯一标识, 名字
 - node 的 express服务器, 习惯端口号: 5050
 - mysql 的端口号: 3306
 - vue端口号: 8080
- localhost:4200 代表访问 当前计算机上 名字是 4200的程序

index.html: 服务器约定俗成的规矩, 默认访问的是 index.html 这个命名的文件

webpack工具: 全自动打包工具, 在项目运行时, 把项目进行打包.

默认设定: 运行时把 main.ts 文件引入到 index.html 中

ts文件: TypeScript语言. 是微软公司出品. 弥补了JS语言的先天不足.

main.ts文件中, 引入的主配置文件: app.module.ts

配置文件中可以注册组件, 然后全局使用

angular的组件:

- 由3个文件组件: html, css, ts
- ts文件是主文件, 其中会引入 html 和 css

```
@Component({
  //组件名, 例如此处 <app-root></app-root>
  selector: 'app-root',
  //引入html文件
  templateUrl: './app.component.html',
  //引入css文件, 数组类型. 一个组件可以拥有多个css
  styleUrls: ['./app.component.css']
})
```

创建组件:

全写格式: `ng generate component 组件名`

- * `generate`: 生成
- * `component`: 组件

简化: `ng g c 组件名`

作业1: 购物车cell

```
ng g c myc01
```

- * 生成文件夹 + 4个文件, 其中有一个 `spec.ts` 是测试文件, 与开发无关.
- * 更新操作: 把组件注册到主配置中 `app.module.ts`

关于命令行:

windows提供了两款命令行终端工具,

- powershell: 新. 默认**不能执行脚本**
- cmd: 旧

```
<p>myc01 works!</p>
```

```
<div class="cell">
```

```

<!--
    vue中: :src="js代码"
    ng 中: [src]="js代码"
-->
<img [src]="goods.pic" alt="" class="cell-pic" />
<span class="cell-title">{{ goods.title }}</span>
<span class="cell-desc">{{ goods.desc }}</span>
<span class="cell-price">¥{{ goods.price.toFixed(2) }}</span>
<div class="cell-count">
    <button
        (click)="goods.count = goods.count - 1"
        [disabled]="goods.count == 1"
    >
        -
    </button>
    <!-- 双向绑定: v-model 此处 ng-model -->
    <!-- form模块默认不加载, 必须手动加载 -->
    <input type="text" [(ngModel)]="goods.count" />
    <button (click)="goods.count = goods.count + 1">+</button>
    <button (click)="goods.count = goods.count * 1 + 1">+</button>
    <!-- goods.count * 1 转数字 然后+1 就不会拼接 -->
</div>
<span class="cell-total">¥{{ (goods.price * goods.count).toFixed(2) }}</span>
</div>

```

```

import { Component, OnInit } from '@angular/core';

@Component({
    selector: 'app-myc01',
    templateUrl: './myc01.component.html',
    styleUrls: ['./myc01.component.css'],
})
export class Myc01Component implements OnInit {
    goods = {
        pic:

        'https://img10.360buyimg.com/cms/s80x80_jfs/t1/45249/15/4307/121563/5d1f1033E53
e2293a/3a3c6921fbfe170a.jpg',
        title: '灵蛇（LINGSHE）有线单键盘USB笔记本台式电脑一体机通用办',
        desc: 'K200防水单键盘',
        price: 19.9,
        count: 1,
    };

    constructor() {}

    ngOnInit(): void {}

    // 传入的值保留2位小数
    money(m) {
        return m.toFixed(2);
    }

    doAdd() {
        this.goods.count++;
    }
}

```

```
}
```

练习2: 轮播图

生成组件:

```
ng g c myc02
```

```
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-myc02',
  templateUrl: './myc02.component.html',
  styleUrls: ['./myc02.component.css'],
})
export class Myc02Component implements OnInit {
  images = ['111.jpg', '222.jpg', '333.jpg', '444.jpg'];
  //           0           1           2           3

  curP = 0; //当前页序号

  nextPage() {
    this.curP++;

    //序号超出最大，则返回0
    if (this.images.length == this.curP) this.curP = 0;
  }

  prevPage() {
    this.curP--;

    if (this.curP < 0) this.curP = this.images.length - 1;
  }

  constructor() {}

  ngOnInit(): void {}
}
```

```
<p>myc02 works!</p>
```

```
<div>
```

```
<!-- 相对路径写法 -->
```

```


<div>
  <button (click)="prevPage()">上一页</button>
  <button (click)="nextPage()">下一页</button>
</div>
</div>
```