# Angular JS 2 安装

## 1、安装，通过npm命令安装，命令如下

npm install –g @angular/cli

其中-g参数表示全局安装， @angular/cli表示angular在npm上注册的名称。

安装之后在C:\Users\acerE\AppData\Roaming\npm\node\_modules目录下可以查看到安装的项目

## 2、通过npm安装出现错误，可通过淘宝镜像来实现安装

在C:\Users\acerE目录下新建文件【.npmrc】，添加以下镜像

sass\_binary\_site=https://npm.taobao.org/mirrors/node-sass/

phantomjs\_cdnurl=http://cnpmjs.org/downloads

electron\_mirror=https://npm.taobao.org/mirrors/electron/

registry=https://registry.npm.taobao.org

第一次安装失败之后，先卸载已经安装的angularjs，执行命令

npm uninstall –g @angular/cli

清除npm缓存

npm cache clean

然后在进行安装

npm install –g @angular/cli

## 3、安装成功之后，开始ng2的学习之旅

学习文档地址：https://angular.cn/docs/ts/latest/

# Angular JS 2 学习

## 1、微语法

### 1、NgFor

NgFor是一个重复器指令——自定义数据显示的一种方式。

例子：



**意思为去除heroes中的每个元素存放到hero中，并在每次迭代时对模板html可用。**

NgFor也可以应用到一个**组件元素**上，例如



这里需要重点注意的是，一个是ngFor前面有个【\*】，另一个是关键字【of】，不是【in】

hero前面的let关键字创建了名叫hero的模板输出变量，**模板输出变量和模板引用变量不是一回事。**

## 2、命名规则

### 1、模块：Module

NgModule是一个装饰器函数，它接收一个用来描述模块属性的元数据对象。其中最重要的属性是：

declarations - 声明本模块中拥有的**视图类**。 Angular 有三种视图类：组件，指令和管道；

exports - declarations 的子集，可用于其它模块的组件模板；

imports - **本**模块声明的组件模板需要的类所在的其它模块；

providers - [服务](https://angular.cn/docs/ts/latest/guide/architecture.html#services)的创建者，并加入到全局服务列表中，可用于应用任何部分。

bootstrap - 指定应用的主视图（称为**根组件**），它是所有其它视图的宿主。只有**根模块**才能设置bootstrap属性。

### 2、组件：Component

### 3、模板：Template

### 4、元数据：MetaData

### 5、数据绑定：Binding

数据绑定的语法有四种形式。每种形式都有一个方向 —— 绑定到 DOM 、绑定自 DOM 以及双向绑定。

1、<li>{{hero.name}}</li>

{{hero.name}}[**插值表达式**](https://angular.cn/docs/ts/latest/guide/displaying-data.html#interpolation)在<li>标签中显示组件的hero.name属性的值

2、<hero-detail [hero]="selectedHero"></hero-detail>

[hero][**属性绑定**](https://angular.cn/docs/ts/latest/guide/template-syntax.html#property-binding)把父组件HeroListComponent的selectedHero的值传到子组件HeroDetailComponent的hero属性中。

3、<li (click)="selectHero(hero)"></li>

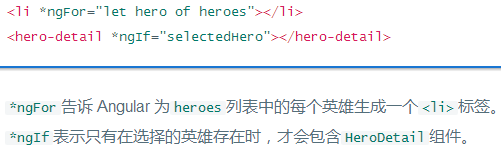
(click) [**事件绑定**](https://angular.cn/docs/ts/latest/guide/user-input.html#click)在用户点击英雄的名字时调用组件的selectHero方法。

4、<input [(ngModel)]="hero.name">

使用ngModel指令组合了属性绑定和事件绑定的功能。

### 6、指令：Directive

1、**结构型**指令通过在 DOM 中添加、移除和替换元素来修改布局



2、**属性型** 指令修改一个现有元素的外观或行为



### 7、服务：Service

我们遵循的文件命名约定是：服务名称的小写形式（基本名），加上.service后缀。 如果服务名称包含多个单词，我们就把基本名部分写成中线形式 ([dash-case](https://angular.cn/docs/ts/latest/guide/glossary.html#dash-case))。 例如，SpecialSuperHeroService服务应该被定义在special-super-hero.service.ts文件中。

### 8、依赖注入：Injector

当 Angular 创建组件时，会首先为组件所需的服务请求一个**注入器 (injector)**。

注入器维护了一个服务实例的容器，存放着以前创建的实例。 如果所请求的服务实例不在容器中，注入器就会创建一个服务实例，并且添加到容器中，然后把这个服务返回给 Angular。 当所有请求的服务都被解析完并返回时，Angular 会以这些服务为参数去调用组件的构造函数。 这就是**依赖注入** 。