

创建指令

指令也是一种服务，只是这种服务的定义有几个特殊要求：

1. 必须使用模块的directive()方法注册服务
2. 对象工厂必须返回一个指令定义对象

“指令在注入器中的登记名称是：指令名+Directive。例如，ng-app指令的服务名称是：ngAppDirective。

- 每个指令定义的工厂函数，需要返回一个指令定义对象。
- 指令定义对象就是一个具有约定属性的JavaScript对象，编译器\$compile在编译时就根据这个定义对象对指令进行展开。
- 指令定义对象的常用属性如下：

配置项	类型	含义
template	string	使用template指定的HTML标记替换指令内容（或指令自身）
restrict	string	用来限定指令在HTML模板中出现的位置
replace	true,false	是否替换原有的DOM元素
transclude	true,false,'element'	是否保留原有指令的内部元素
scope	true,false,{} 	scope属性为指令创建私有的作用域
link	Function	link属性是一个函数，用来在指令中操作DOM树、实现数

template 定义 替换 模板

- template指明一个HTML片段,可以用来替换指令的内容
- 如果replace = true , 那么用HTML片段替换指令本身
- 如果transclude属性为true则包裹指令的内容

匹配模式restrict

restrict	元素	用法
E	元素	\
A	属性	\
C	样式类	\
M	注释	\

- 推荐使用元素和属性的方式使用指令
- **组件式** 指令使用 **元素** 名称的方式创建指令
- **装饰型** 指令使用 **属性** 的方式创建指令



replace

- replace属性指明使用template时，如何替换指令元素：
- true表示编译时，将使用template替换指令元素
- false表示编译时，将使用template替换指令元素的内容

“replace为true时，要求模板必须有一个根节点”

transclude

- 有些指令需要能够包含其他未知的元素
- transclude属性可以告诉编译器，利用所在DOM元素的内容，替换template中包含 ng-transclude指令的元素的内容

```
<dialog>  
  <div>这里是指令内部的内容。</div>  
</dialog>
```

“对话框的内容在我们开发dialog指令的时候是无法得知的，这部分内容需要被转移到展开的DOM树中适当的位置。”

- 使用transclude有两个要点：
 1. 需要首先声明transclude属性值为true,这将告诉编译器,使用我们这个指令的 DOM元素，其内容需要被复制并插入到编译后的DOM树的某个点。
 2. 需要在template属性值中使用ng-transclude指明插入点



scope的绑定策略

类型	用法
@	把当前属性作为字符串传递。你还可以绑定来自外层scope上的值，在属性中插入{{}}即可
=	与父scope中的属性进行双向绑定
&	传递一个来自父scope中的函数，稍后调用

link:在指令中操作DOM

如果需要在指令中操作DOM，我们需要在对象中定义link属性

```
function link(scope, iElement, iAttrs, controller, transcludeFn) { ... }
```

“注意link函数的参数，AngularJS在编译时负责传入正确的值：

参数说明

参数	含义
scope	指令对应的scope对象。如果指令没有定义自己的本地作用域，那么传入的就是外部的作用域对象
iElement	指令所在DOM对象的jqLite封装。如果使用了template属性，那么iElement对应变换后的DOM对象的jqLite封装
iAttrs	指令所在DOM对象的属性集。这是一个Hash对象，每个键是驼峰规范化后的属性名
controller	控制器的实例,在所有指令间共享,可以作为指令交流的通道

指令的原理

1. 加载 加载angular.js,找到ng-app指令，确定应用的边界
2. 编译 遍历DOM，找到所有指令，根据指令中的template、replace和transclude转换DOM结构，如果存在 `compile` 函数就调用
3. 链接 对每个指令操作 `link` 函数,link函数一般用来操作DOM，绑定事件监听器

compile

- 编译函数用于实现对模板自身的转换，此时无法访问scope对象
- 适合修改节点，插入子模板,最后返回postLink函数
- 参数

参数	含义
element	未link过的原始DOM节点
attrs	原始DOM节点上的属性

compile和link

- compile函数用来对模板 **自身进行转换**，而link函数负责在模型和视图之间进行 **动态** 关联
- scope在 **链接** 阶段才会被绑定到编译之后的link函数上
- compile函数仅在编译阶段运行 **一次**，而对于指令的 **每个** 实例，link函数都会执行一次
- compile函数可以返回 **prelink** 和 **postlink** 函数，而link函数只会返回 **postlink** 函数
- 如果需要修改dom结构，应该在 **postlink** 中做这件事情，如果在 **prelink** 中做这件事会导致错误
- 不成功时会保留原样，并在model上增加一个\$error对象



表单验证

- 定义验证规则，验证有效性
- 显示验证结果
- 禁用html5自带验证 `novalidate="novalidate"`
- 用户输入后，angular会依次调用验证器进行验证，全部成功时model会变成用户输入的值
- 不成功时则保留原值，并在model上增加一个\$error对象

成员变量

变量名	含义
\$dirty	表单中任何一项是否输入过
\$pristine	表单中任何一项尚未输入过
\$error	存放错误信息
\$invalid	表单数据是否无效，只要有一项无效则整个表单无效
\$valid	与\$invalid相反
\$name	表单的名字
email...	表单各个输入框的model

提示和自定义验证

- 显示错误提示
- 友好的错误提示
- 多字段联合验证
- 远程跨域验证