

ng-app

- ng-app指令来指定一个Angular应用程序的入口，并通过AngularJS完成自动初始化和创建应用根作用域
- 同时载入和指令内容相关的模块，并通过拥有ng-app指令的标签为根节点开始编译其中的DOM。

引用方法如下所示：

```
<div ng-app></div>
```

- 如上引用，一个AngularJS应用程序初始化就完成了并标记了作用域，也就是div元素就是AngularJS应用程序的"所有者"，在它里面的指令也就会被Angular编译器所编译、解析了。



ng-init

- ng-init指令 **初始化** 应用程序数据，也就是为AngularJS应用程序定义初始值
- 如下所示，我们为应用程序变量name赋定初始值。

```
<div ng-app="" ng-init="name='zfpx'">  
  </div>
```

- 不仅可以赋值字符串，也可以赋值为数字、数组、对象，而且可以为多个变量赋初始值

- 数据绑定，由两个花括号{{}}组成，可以把 **数据绑定到HTML**，类似 Javascript代码片段，可以包含文字、运算符和变量
- 输出变量的值

```
<div ng-app ng-init="name='zfpx'">
  {{name}}
</div>
```

- 进行表达式计算

ng-model

- 在AngularJS中，只需要使用ng-model指令就可以把应用程序数据绑定到HTML元素，实现model和view的 **双向绑定**。
- ng-model把相关处理事件绑定到指定标签上，这样我们就可以不用在手工处理相关事件(比如change等)的条件下完成对数据的展现需求。
- 如下示例，使用ng-model指令对数据进行绑定。

```
<div ng-app>  
请输入任意值: <input type="text" ng-model="name">  
你输入的为: {{ name }}  
</div>
```

ng-model原理

1. angular加载完成之后会启动，首先找 `ng-app` 指令
2. 找到后认为 `ng-app` 里面的所有的内容都归angular来管
3. 找子层标签里所有的指令，然后就可以找到 `ng-model`
4. 找到后会生成数据模型，然后挂在 **作用域** 上面
5. 使用此数据模型的 **变量** 和 **视图** 进行 **绑定**

ng-bind

- `ng-bind` 和 `AngularJS` 表达式功能相似，但 `ng-bind` 是在 `angular` 解析渲染完毕后才将数据显示出来的。
- 如下使用 `ng-bind` 指令绑定把应用程序数据。

```
<div ng-app="">  
  请输入一个名字: <input type="text" ng-model="name">  
  Hello <span ng-bind="name"></span>  
</div>
```

“使用花括号语法时，因为浏览器需要首先加载页面，渲染它，然后AngularJS才能把它解析成你期望看到的内容，所以对于首个页面中的数据绑定操作，建议采用`ng-bind`，以避免其未被渲染的模板被用户看到。

ng-click

- AngularJS也有自己的HTML事件指令,比如说通过ng-click定义一个AngularJS单击事件。
- ng-click指令将DOM元素的鼠标点击事件(即mousedown)绑定到一个方法上,当浏览器在该DOM元素上鼠标触发点击事件时,Angular就会调用相应的方法。
- 对按钮、链接等,我们都可以用ng-click指令属性来实现绑定,如下简单示例:

```
<div ng-app="" ng-init="click=false">
  <button ng-click="click= !click">隐藏/显示</button>
  <div ng-hide="click">
    请输入一个名字: <input type="text" ng-model="name">
    Hello <span ng-bind="name"></span>
  </div>
</div>
```

“ PS : ng-hide="true", 设置HTML元素不可见。

ng-repeat

- ng-repeat指令，遍历一个数据集中的每个数据元素，并且加载HTML模版把数据渲染出来

```
<div class="container-fluid" ng-init="phones=[{name:'iphone'}, {name:'xiaomi'}]">
  <div class="row-fluid">
    Search: <input ng-model="query">
    <ul class="phones">
      <li ng-repeat="phone in phones | filter:query">
        {{phone.name}}
      </li>
    </ul>
  </div>
</div>
```


ng-class

- ng-class用于动态自定义dom元素的css
- scope变量绑定

```
function ctr($scope){  
    $scope.test ="classname";  
}  
<div class="{{test}}"></div>
```

- 对象key/value处理

```
<div ng-class="{ 'has-error':username.length>6}"></div>
```

ng-include

- `ng-include` 就是将多个页面的公共页面提取出来，如 `header.html`，`footer.html` 等，在每个页面公用

```
<div ng-include="'header.html'"></div>
```