# 注:

Angular版本:1.4.6

浏览器:360极速浏览器

# 问题汇总:

|  |  |
| --- | --- |
| 问题 | 解决办法 |
| 一个页面多个ng-app的时候只有第一个app会被angular初始化 | 在angular中,一个页面只会加载一个app.所以在页面中添加了controller的概念,一个app内可以引用多个controller来控制angular的定义的模块 |
| 当在页面的Dom元素上添加并命名了ng-app时,angular元素失效 | 因为定义了页面的ng-app被添加并且被命名的时候,一定要添加js:  **var** app = angular.module('myApp1', []);  进行模块化定义. |
| restrict: 'EACM'  自定义指令的时候,M注解的使用方法失效. | / |
| ng-model和ng-bind的区别 | ng-model是用于表单元素的，支持双向绑定。对普通元素无效；  ng-bind用于普通元素，不能用于表单元素，应用程序单向地渲染数据到元素； |
| ng-bind和表达式同时使用 | 当两个同时使用的时候,ng-bind绑定的值会覆盖表达式的值 |

# 主要特性

MVC、模块化、自动化双向数据绑定、语义化标签、依赖注入

# 表达式

表达式使用{{expression}}的格式,添加到html元素的值域中为其添加值.

AngularJS 表达式 与 JavaScript 表达式对比

类似于 JavaScript 表达式，AngularJS 表达式可以包含字母，操作符，变量。

与 JavaScript 表达式不同，AngularJS 表达式可以写在 HTML 中。

与 JavaScript 表达式不同，AngularJS 表达式不支持条件判断，循环及异常。

与 JavaScript 表达式不同，AngularJS 表达式支持过滤器。

# 指令

Angular的指令都是入侵到html标签的属性区上,从而使得对应的html的元素拥有angular的一些特性和功能

**ng-app** 指令初始化一个 AngularJS 应用程序。

**ng-init** 指令应用程序初始化的函数操作、变量定义等等。

**ng-model** 指令把元素值（比如输入域的值）绑定到应用程序。

**ng-repeat** 指令用于操作应用程序中的集合对象,并且会根据集合的个数来克隆HTML元素.

**ng-bing** 用于将应用程序中的参数的值添加到html对象中,效果跟使用表达式差不多.

# 自定义指令

Angular支持自定义指令.通过使用app.directive自定义指令提供已经初始化的app使用.

自定义指令的调用方式:元素名(E),属性(A),类名(C),注解(M)

使用驼峰法来命名一个指令testDirective, 但在使用它时需要以 **-** 分割test-directive

例:

app.directive("testDirective", **function** () {  
 **return** {  
 template: "<h1>this is my custom directive</h1>"  
 };  
});