[一、 表达式 1](#_Toc17727)

[二、 指令 1](#_Toc28067)

[三、 模型 2](#_Toc25605)

[四、 作用域和控制器 3](#_Toc1121)

[五、 过滤器 4](#_Toc25138)

[六、 服务 4](#_Toc28609)

# 表达式

AngularJS 表达式写在双大括号内：**{{ expression }}**。

AngularJS 表达式 与 JavaScript 表达式

1. 类似于 JavaScript 表达式，AngularJS 表达式可以包含字母，操作符，变量。
2. 与 JavaScript 表达式不同，AngularJS 表达式可以写在 HTML 中。
3. 与 JavaScript 表达式不同，AngularJS 表达式不支持条件判断，循环及异常。
4. 与 JavaScript 表达式不同，AngularJS 表达式支持过滤器。

<div ng-app="" ng-init="a=1;b=2;person={name:'lixinmiao'};points=[1,2,3]">

<!--表达式{{}}-->

<p>我的第一个表达式: {{ 5 + 5 }}</p>

<p>我的第二个表达式: {{ a\*b }}</p>

<p>我的第三个表达式:{{person.name}}</p>

<p>我的第四个表达式:{{points[2]}}</p>

<span ng-bind="a + ' ' + b"/>

</div>

# 指令

1. AngularJS 指令是扩展的 HTML 属性，带有前缀 ng-
2. ng-app 指令初始化一个 AngularJS 应用程序。
3. ng-init 指令初始化应用程序数据。通常情况下，不使用 ng-init。您将使用一个控制器或模块来代替它。
4. ng-model 指令把元素值（比如输入域的值）绑定到应用程序

* 为应用程序数据提供类型验证（number、email、required）。
* 为应用程序数据提供状态（invalid、dirty、touched、error）。
* 为 HTML 元素提供 CSS 类。
* 绑定 HTML 元素到 HTML 表单

1. ng-repeat 指令会重复一个 HTML 元素

<div ng-app="" ng-init="quantity=1;price=5">

<h2>价格计算器</h2>

数量： <input type="number" ng-model="quantity">

价格： <input type="number" ng-model="price">

<p>

<b>总价：</b> {{ quantity \* price }}

</p>

</div>

<div ng-app="" ng-init="names=['Jani','Hege','Kai']">

<p>使用 ng-repeat 来循环数组</p>

<ul>

<li ng-repeat="x in names"> {{ x }} </li>

</ul>

</div>

# 模型

1. ng-model 指令可以将输入域的值与 AngularJS 创建的变量绑定。
2. 验证用户输入
3. ng-model 指令可以为应用数据提供状态值(invalid, dirty, touched, error)
4. ng-model 指令基于它们的状态为 HTML 元素提供了 CSS 类

* ng-empty
* ng-not-empty
* ng-touched
* ng-untouched
* ng-valid
* ng-invalid
* ng-dirty
* ng-pending
* ng-pristine

# 作用域和控制器

1. Scope(作用域) 是应用在 HTML (视图) 和 JavaScript (控制器)之间的纽带。
2. Scope 是一个对象，有可用的方法和属性。
3. Scope 可应用在视图和控制器上。
4. 所有的应用都有一个 $rootScope，它可以作用在 ng-app 指令包含的所有 HTML 元素中。
5. $rootScope 可作用于整个应用中。是各个 controller 中 scope 的桥梁。用 rootscope 定义的值，可以在各个 controller 中使用。

<div ng-app="myApp" ng-controller="myCtrl">

<input ng-model="name">

<h1>{{ greeting}}</h1>

<button ng-click='sayHello()'>点我</button>

</div>

<script>

var app = angular.module('myApp', []);

app.controller('myCtrl', function($scope) {

$scope.name = "Runoob";

$scope.sayHello = function() {

$scope.greeting = 'Hello ' + $scope.name + '!';

};

});

</script>

# 过滤器

过滤器可以使用一个管道字符（|）添加到表达式和指令中。

|  |  |
| --- | --- |
| currency | 格式化数字为货币格式。 |
| filter | 从数组项中选择一个子集。 |
| lowercase | 格式化字符串为小写。 |
| orderBy | 根据某个表达式排列数组。 |
| uppercase | 格式化字符串为大写。 |

# 服务

1. 内置服务
2. **$location服务：**用于返回当前页面的URL地址

var app = angular.module('myApp', []);

app.controller('customersCtrl', function($scope, $location) {

$scope.myUrl = $location.absUrl();

});

1. **$http服务：用于服务器的数据传输**

var app = angular.module('myApp', []);

app.controller('myCtrl', function($scope, $http) {

$http.get("welcome.htm").then(function (response) {

$scope.myWelcome = response.data;

});

});

1. **$timeout()服务和$interval()服务:setTimeout()和setTimeInterval函数**
2. 自定义服务

app.controller('myCtrl', function($scope, $interval) {

 $scope.theTime = new Date().toLocaleTimeString();

 $interval(function () {

  $scope.theTime = new Date().toLocaleTimeString();

 }, 1000);

});