# 4.AngularJS ng-model 指令

ng-model 指令用于绑定应用程序数据到 HTML 控制器(input, select, textarea)的值。

## ng-model 指令

ng-model 指令可以将输入域的值与 AngularJS 创建的变量绑定。

[4-1]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="myCtrl">  
 名字: <input ng-model="name">  
</div>  
  
<script>  
 var ***app*** = angular.module('myApp', []);  
 ***app***.controller('myCtrl', function($scope) {  
 $scope.name = "John Doe";  
 });  
</script>  
  
<p>使用 ng-model 指令来绑定输入域的值到控制器的属性。</p>  
  
</body>  
</html>

## 双向绑定

双向绑定，在修改输入域的值时， AngularJS 属性的值也将修改：

[4-2]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="myCtrl">  
 名字: <input ng-model="name">  
 <h1>你输入了: {{***name***}}</h1>  
</div>  
  
<script>  
 var ***app*** = angular.module('myApp', []);  
 ***app***.controller('myCtrl', function($scope) {  
 $scope.name = "John Doe";  
 });  
</script>  
  
<p>修改输入框的值，标题的名字也会相应修改。</p>  
  
</body>  
</html>

## 验证用户输入

[4-3]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<form ng-app="" name="myForm">  
 Email:  
 <input type="email" name="myAddress" ng-model="text">  
 <span ng-show="myForm.myAddress.$error.email">不是一个合法的邮箱地址</span>  
</form>  
  
<p>在输入框中输入你的邮箱地址，如果不是一个合法的邮箱地址，会弹出提示信息。</p>  
  
</body>  
</html>

以上实例中，提示信息会在 ng-show 属性返回 true 的情况下显示。

## 应用状态

ng-model 指令可以为应用数据提供状态值(invalid, dirty, touched, error):

[4-4]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<form ng-app="" name="myForm" ng-init="myText = 'test@runoob.com'">  
  
 Email:  
 <input type="email" name="myAddress" ng-model="myText" required>  
 <p>编辑邮箱地址，查看状态的改变。</p>  
 <h1>状态</h1>  
 <p>Valid: {{myForm.myAddress.$valid}} (如果输入的值是合法的则为 true)。</p>  
 <p>Dirty: {{myForm.myAddress.$dirty}} (如果值改变则为 true)。</p>  
 <p>Touched: {{myForm.myAddress.$touched}} (如果通过触屏点击则为 true)。</p>  
  
</form>  
  
</body>  
</html>

## CSS 类

ng-model 指令基于它们的状态为 HTML 元素提供了 CSS 类：

[4-5]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
 <style>  
 input.ng-invalid {  
 background-color: lightblue;  
 }  
 </style>  
</head>  
<body>  
  
<form ng-app="" name="myForm">  
 输入你的名字:  
 <input name="myName" ng-model="myText" required>  
</form>  
  
<p>编辑文本域，不同状态背景颜色会发生变化。</p>  
<p>文本域添加了 required 属性，该值是必须的，如果为空则是不合法的。</p>  
  
</body>  
</html>

ng-model 指令根据表单域的状态添加/移除以下类：

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