



## 第四讲 使用表单

# 教学目标

---

1. 使用表单
2. 校验表单
3. 表单指令属性

1

使用表单

1

# HTML控件

input 元素

button 元素

select 元素

textarea 元素

## 2

# AngularJS中的input元素

```
<form novalidate>
  <div class="well">
    <div class="form-group">
      <label>用户名:</label>
      <input type="text" class="form-control">
    </div>
  </div>
</form>
```

`novalidate` 属性在应用中不是必须的，但是你需要在 AngularJS 表单中使用，用于重写标准的 HTML5 验证。

`novalidate` 属性规定当提交表单时不对其进行验证

[demo:ex01.html](#)

## 2

# AngularJS中的button元素

**ng-value** 指令用于设置 input 或 select 元素的 value 属性。

按钮1：

```
<input type="submit" value="确定" class="btn">  
<input type="reset" ng-value="'cancel'" class="btn">  
<input type="reset" ng-value="cancel" class="btn">
```

按钮2：

```
<button type="submit" class="btn btn-primary btn-block">  
    OK  
</button>
```

demo:ex02.html

# 3

## AngularJS中的select元素

HTML中的select元素：

```
<select>  
  <option value="">(选一个)</option>  
  <option value="">苹果</option>  
  <option value="">桔子</option>  
  <option value="">黄瓜</option>  
</select>
```

demo:ex03.html

# 3

## AngularJS中的select元素

为select元素添加数据绑定：

```
<select ng-model="selectValue">
  <option value="">(选一个)</option>
  <option ng-repeat="x in todo"> {{x.fruit}}
</option>
</select>
```

```
<p>你选择的是： {{selectValue || '无'}}</p>
```

demo:ex03.html



# 3

## AngularJS中的select元素

### ng-options 指令

用于使用 `<options>` 填充 `<select>` 元素的选项。

#### 语法:

```
<select ng-options=" expression" > </select>
```

*expression* : 为 select 元素填充选项的表达式

3

# AngularJS中的select元素

数据源的类型有**数组**和**对象**两种

expression表达式用法(使用**数组作为数据源**) :

( 1 ) **label for value in array**

( 2 ) **select as label for value in array**

( 3 ) **label group by group for value in array**

( 4 ) **select as label group by group for value in array track by trackexpr**

# 3

## AngularJS中的select元素

( 1 ) **label for value in array** 用数组中的值做标签

**label** : 标签变量(需同循环变量对应) , option标签的内容,

**value** : 循环变量, option的value属性值

**array** : 数组名,数据源

```
▼<select ng-model="selected" ng-options="item for  
item in fruits" id="fruit-select">  
  <option value="" class="ng-pristine">(选一个)</option>  
  <option value="0">苹果</option>  
  <option value="1">桔子</option>  
  <option value="2">黄瓜</option>  
</select>
```

循环变量

标签变量

0 苹果

1 桔子

2 黄瓜

# 3

## AngularJS中的select元素

( 1 ) **label for value in array**      用数组中的值做标签

```
$scope.arr= [  
  { id: 100, fruit: "苹果", really: true },  
  { id: 200, fruit: "桔子", really: true },  
  { id: 300, fruit: "黄瓜", really: false }  
];
```

```
<select ng-options="item.fruit for item in arr">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex05.html

3

## AngularJS中的select元素

( 2 ) **select as label for value in array**

用数组中的值作为**选中的**标签

**select** : 默认选中的标签变量, 需与标签变量同名

**label** : 标签变量(需同循环变量对应) , option标签的内容,

**value** : 循环变量, option的value属性值

**array** : 数组名,数据源

**select** : 结果绑定到ngModel中 , 如果没有指定 , 则默认绑定value

# 3

## AngularJS中的select元素

### ( 2 ) select as label for value in array

```
▼<select ng-model="fru" ng-options="item.fruit as item.fruit for item in arr"  
class="valid">
```

设置被选中

```
<option>(选一个)</option>  
<option value="0">苹果</option>  
<option value="1" selected="selected">桔子</option>  
<option value="2">黄瓜</option>  
</select>
```

```
<select ng-model="fru" ng-options="item.fruit as  
item.fruit for item in arr">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex06.html

# 3

## AngularJS中的select元素

( 3 ) label group by group for value in array

用数组中的值做标签组

```
$scope.arr= [
  { id: 100, fruit: "苹果", group: "水果" },
  { id: 200, fruit: "桔子", group: "水果" },
  { id: 300, fruit: "黄瓜", group: "蔬菜" },
  { id: 300, fruit: "油菜", group: "蔬菜" }
];
```

```
<select ng-model="fru1" ng-options="item.fruit
group by item.group for item in arr">
  <option value="">(选一个)</option>
</select>
```

demo:ex07.html

③

## AngularJS中的select元素

( 4 ) `select as label group by group for value in array track by trackexpr`

用数组中的值作为选中的标签组

**group** : group by的条件，表示按某条件进行分组

**trackexpr** : 用于唯一确定数组中的迭代项的表达式

[demo:ex08.html](#)



# 3

## AngularJS中的select元素

**expression表达式用法(使用对象作为数据源) :**

( 1 ) **label for** (key , value) **in** object

( 2 ) **select as** label **for** (key , value) **in** object

( 3 ) **label group by** group **for** (key , value) **in** object

( 4 ) **select as** label **group by** group **for** (key , value) **in** object

**object** : 数据源的类型 , 有数组和对象两种

**value** : 迭代过程中 , 引用数组的项或者对象的属性值

**key** : 迭代过程中 , 引用对象的属性名

# 3

## AngularJS中的select元素

( 1 ) **label for (key , value) in object**

用对象的键-值 ( key-value ) 做标签。

```
$scope.name='lily';  
$scope.obj={  
  user: 'lily',  
  pwd: 123456,  
  email: "888@qq.com"  
}
```

```
<select ng-model="name"  
        ng-options="x for (x,y) in obj">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex09.html

③

## AngularJS中的select元素

( 2 ) **select** **as** **label** **for** (**key** , **value**) **in** **object**

用对象的键-值作为选中的标签。

```
<select ng-model="name "  
  ng-options="x as x for (x,y) in obj">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex10.html

③

## AngularJS中的select元素

( 3 ) **label** **group** **by** **group** **for** (**key** , **value**) **in** **object**

用对象的键-值作为标签组

```
<select ng-model="name "  
  ng-options="y group by x for (x,y) in obj">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex11.html

### ③

## AngularJS中的select元素

( 4 ) select as label group by group for (key , value) in object

用对象的键-值作为选中的标签组。

```
<select ng-model="name "  
  ng-options="y as y group by x for (x,y) in obj">  
  <option value="">(选一个)</option>  
</select>
```

demo:ex12.html

# 4

## AngularJS中的textarea元素

AngularJS实现textarea记录只能输入规定数量的字符并显示

如果在同一个表单中有多个“textarea”可以通过定义多个“ng-model”来控制

你还可以输入的字符个数为: 57

你输入的信息: 如果在同一个表单中有多个“textarea”可以通过定义多个“ng-model”来控制

demo:ex13.html

2

校验表单

# 1

## 执行基本的表单验证

AngularJS提供了基本的表单验证方法，实现了对标准HTML元素属性的支持。如 type 和 required 等。

将标准HTML校验属性应用于input元素上既可：

```
<input type="text" required ng-model="newUser.name">
```

**demo:ex14.html**



## ②

# 表单的有效性

名称	描述
<code>\$pristine</code>	如果用户没有与元素/表单产生交互，则返回true
<code>\$dirty</code>	如果用户与元素/表单产生过交互，则返回true
<code>\$valid</code>	当元素/表单内容的校验结果有效时返回true
<code>\$invalid</code>	当元素/表单内容的校验结果无效时返回true
<code>\$error</code>	提供校验错误的详情信息

3

## 提供表单校验反馈信息

### 1、使用CSS提供校验反馈信息

用户名:

邮箱:

☐ 同意协议的相关内容

OK

注册信息: 暂无

表单验证: **false**

demo:ex15.html

# 4

## 提供表单校验反馈信息

### 2、使用特殊变量提供校验反馈信息

邮箱:

241

请输入有效的电子邮件地址

OK

**demo:ex16.html**

# 3

## 表单指令属性

# 属性指令

属性的指令有：

**ng-checked**：在input元素上使用

**ng-disabled**：在所有表单元素上使用

**ng-open**：在details元素上使用

**ng-readonly**：在input和textarea元素上使用

**ng-selected**：在Option元素上使用

# 1

## ng-checked

( 1 ) **ng-checked** : 在input元素上使用

- ◆ **ng-checked** 指令用于设置复选框(checkbox)或单选按钮(radius)的 checked 属性。
- ◆ **ng-checked** 属性返回 true时复选框(checkbox)或单选按钮(radius)将会被选中。

### 语法

```
<input type="checkbox|radio" ng-checked="expression"></input>
```

# 1

## ng-checked

( 1 ) **ng-checked** : 在input元素上使用

```
<div class="form-group">
  <h3>我喜欢的水果:</h3>
  <input type="checkbox" ng-checked="aa">苹果
</div>
<div class="form-group">
  <h3>我喜欢的水果:</h3>
  <input type="radio" ng-checked="true">苹果
  <input type="radio" ng-checked="false">桔子
  <input type="radio" ng-checked="1">香蕉
</div>
```

**demo:ex17.html**

# 1

## ng-checked

( 1 ) **ng-checked** : 实现全选功能

```
<div class="form-group">
  <h3>我喜欢的水果:</h3>
  <input type="checkbox" ng-model="all"> 全选
  <input type="checkbox" ng-checked="all"> 苹果
  <input type="checkbox" ng-checked="all"> 桔子
  <input type="checkbox" ng-checked="all"> 香蕉
</div>
```

**demo:ex18.html**



## ②

# ng-disabled

( 2) **ng-disabled** : 在input和button元素上使用

◆ **ng-disabled** 指令设置表单输入字段的 disabled 属性(input, select, 或 textarea)。

◆ 如果 **ng-disabled** 中的表达式返回 true 则表单字段将被禁用。

## 语法

```
<input ng-disabled=" expression" > </input>
```

<input>, <select>, <textarea>和<button>元素都支持该指令。

# 2

## ng-disabled

( 2) **ng-disabled** : 在input和button元素上使用

```
<input type="checkbox" ng-model="all">  
<input type="text" ng-disabled="all">  
<input type="radio" ng-disabled="all">  
<select ng-disabled="all"></select>  
<input type="submit" ng-disabled="all">  
<button ng-disabled="all">按钮</button>  
<textarea ng-disabled="all" cols="35" rows="3">  
</textarea>
```

**demo:ex19.html**

3

## ng-open

( 3 ) **ng-open** : 在details元素上使用

```
<input type="checkbox" ng-model="showDetails">
```

```
<details ng-open="showDetails">
```

```
<summary>details标签用于描述……</summary>
```

- details 标签是 HTML 5 中的新标签<br>
- open属性 定义 details 是否可见<br>
- summary 标签可以为 details 定义标题<br>
- 用户点击标题时，会显示出 details<br>
- 目前只有 Chrome 和 Safari 6 <br>

```
</details>
```

**demo:ex20.html**

# 4

## ng-redonly

( 4 ) **ng-redonly** : 在input和textarea元素上使用

**ng-redonly** 指令用于设置表单域(input 或 textarea) 的 readonly 属性。

如果 **ng-redonly** 属性的表达式返回 true 则表单域为只读。

```
<input type="text" ng-redonly="true" value="只读">  
<input type="text" ng-redonly="false">
```

**demo:ex21.html**

## 5

# 对表单元素使用双向数据绑定

( 5 ) **ng-selected** : 在Option元素上使用

**ng-selected** 指令用于设置 <select> 列表中的 <option> 元素的 selected 属性。

ng-selected 属性的表达式返回 true 则选项被选中。

```
<input type="checkbox" ng-model="mySel">
<p>我喜欢的水果:</p>
<select>
  <option>苹果</option>
  <option ng-selected="mySel">桔子</option>
  <option>香蕉</option>
</select>
```

**demo:ex22.html**

# 6

## 其他

名称	描述
ng-minlength	设置一个合法元素所需的最小字符数
ng-maxlength	设置一个合法元素所需的最大字符数
ng-pattern	设置一个正则表达式，合法的元素内容必须匹配
ng-required	通过数据绑定设置required值

# AngularJS API

API	描述
<code>angular.lowercase()</code>	转换字符串为小写
<code>angular.uppercase()</code>	转换字符串为大写
<code>angular.isString()</code>	判断给定的对象是否为字符串，如果是返回 true。
<code>angular.isNumber()</code>	判断给定的对象是否为数字，如果是返回 true。

# angular.lowercase()

```
<div ng-app="myApp" ng-controller="myCtrl">
  <p>{{ x1 }}</p>
  <p>{{ x2 }}</p>
</div>
<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
  $scope.x1 = "JOHN";
  $scope.x2 = angular.lowercase($scope.x1);
});
</script>
```



# angular.isString()

```
<div ng-app="myApp" ng-controller="myCtrl">
  <p>{{ x1 }}</p>
  <p>{{ x2 }}</p>
</div>
<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
  $scope.x1 = "JOHN";
  $scope.x2 = angular.isString($scope.x1);
});
</script>
```

# 本课小结

---

1. 使用表单
2. 校验表单
2. 表单指令属性

# TNAKS

主讲：王智娟

QQ: 24132228

Email: wangzhijuan@onest.net