# 10.AngularJS Select(选择框)

AngularJS 可以使用数组或对象创建一个下拉列表选项。

## 使用 ng-options 创建选择框

在 AngularJS 中我们可以使用 **ng-option** 指令来创建一个下拉列表，列表项通过对象和数组循环输出，如下实例:

### **实例**

<div ng-app="myApp" ng-controller="myCtrl"> <select ng-init="selectedName = names[0]" ng-model="selectedName" ng-options="x for x in names"> </select> </div> <script> var app = angular.module('myApp', []); app.controller('myCtrl', function($scope) { $scope.names = ["Google", "Runoob", "Taobao"]; }); </script>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select" \t "https://www.runoob.com/angularjs/_blank)**

ng-init 设置默认选中值。

## ng-options 与 ng-repeat

我们也可以使用**ng-repeat** 指令来创建下拉列表：

### **实例**

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

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_repeat" \t "https://www.runoob.com/angularjs/_blank)**

**ng-repeat** 指令是通过数组来循环 HTML 代码来创建下拉列表，但 **ng-options** 指令更适合创建下拉列表，它有以下优势：

使用 **ng-options** 的选项是一个对象， **ng-repeat** 是一个字符串。

## 应该用哪个更好?

假设我们使用以下对象:

$scope.sites = [

{site : "Google", url : "http://www.google.com"},

{site : "Runoob", url : "http://www.runoob.com"},

{site : "Taobao", url : "http://www.taobao.com"}

];

**ng-repeat** 有局限性，选择的值是一个字符串:

### **实例**

使用 **ng-repeat**:

<select ng-model="selectedSite">  
<option ng-repeat="x in sites" value="{{x.url}}">{{x.site}}</option>  
</select>  
  
<h1>你选择的是: {{selectedSite}}</h1>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_repeat_selected" \t "https://www.runoob.com/angularjs/_blank)**

使用 **ng-options** 指令，选择的值是一个对象：

### **实例**

使用 **ng-options**:

<select ng-model="selectedSite" ng-options="x.site for x in sites">  
</select>  
  
<h1>你选择的是: {{selectedSite.site}}</h1>  
<p>网址为: {{selectedSite.url}}</p>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_selected" \t "https://www.runoob.com/angularjs/_blank)**

当选择值是一个对象时，我们就可以获取更多信息，应用也更灵活。

## 数据源为对象

前面实例我们使用了数组作为数据源，以下我们将数据对象作为数据源。

$scope.sites = {

site01 : "Google",

site02 : "Runoob",

site03 : "Taobao"

};

**ng-options** 使用对象有很大的不同，如下所示：

### **实例**

使用对象作为数据源, **x** 为键(key), **y** 为值(value):

<select ng-model="selectedSite" ng-options="**x for (x, y) in sites**">  
</select>  
  
<h1>你选择的值是: {{selectedSite}}</h1>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_object" \t "https://www.runoob.com/angularjs/_blank)**

你选择的值为在 key-**value** 对中的 **value**。

**value** 在 key-**value** 对中也可以是个对象：

### **实例**

选择的值在 key-**value** 对的 **value** 中, 这是它是一个对象:

$scope.cars = {  
car01 : {brand : "Ford", model : "Mustang", color : "red"},  
car02 : {brand : "Fiat", model : "500", color : "white"},  
car03 : {brand : "Volvo", model : "XC90", color : "black"}  
};

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_object_object" \t "https://www.runoob.com/angularjs/_blank)**

在下拉菜单也可以不使用 **key**-value 对中的 **key** , 直接使用对象的属性：

### **实例**

<select ng-model="selectedCar" ng-options="**y.brand** for (x, y) in cars">  
</select>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_select_object_y" \t "https://www.runoob.com/angularjs/_blank)**