**搞懂 AngularJS 預設模組 select 標籤的 ngOptions 參數用法**

时间 2013-05-12 00:37:00 [The Will Will Web](http://www.tuicool.com/sites/fMZVna) *原文*  [http://blog.miniasp.com/post/2013/05/12/AngularJS-ng-module-select-ngOptions-usage-samples.aspx](http://blog.miniasp.com/post/2013/05/12/AngularJS-ng-module-select-ngOptions-usage-samples.aspx?utm_source=tuicool)

今天晚上花了點時間，把 AngularJS 預設模組裡 select 標籤的 ngOptions 參數用法給徹底搞懂，這是我前幾天在公司展示 AngularJS 開發實戰時，一個稍微卡住的地方。今天利用 [JSBin](http://jsbin.com/) 自己弄出一個 [完整範例](http://jsbin.com/ukezuj/5) 後終於大澈大悟。說實在的，AngularJS 這玩意要弄通他，對初學者來說好像還真的有點難度，今天聽同事跟我說，我前幾天再講解 AngularJS 的時候，只覺得我 DEMO 的很帥、程式寫的很快，但自己無法深刻理解，回家想兩天之後才跟我說：「保哥，我想通了，這玩意好棒喔！」XDD

首先，我們來看看 [AngularJS:select](http://docs.angularjs.org/api/ng.directive:select) API 的 ngOptions 參數說明：

|  |
| --- |
| **ngOptions*(optional)* – {comprehension\_expression=} –** in one of the following forms:   * for array data sources:   + label **for** value **in** array   + select **as** label **for** value **in** array   + label **group by** group **for** value **in** array   + select **as** label **group by** group **for** value **in** array * for object data sources:   + label **for (** key **,** value **) in** object   + select **as** label **for (** key **,** value **) in** object   + label **group by** group **for (** key **,** value **) in** object   + select **as** label **group by** group **for (** key **,** value **) in** object   Where:   * array / object : an expression which evaluates to an array / object to iterate over. * value : local variable which will refer to each item in the array or each property value of object during iteration. * key : local variable which will refer to a property name in object during iteration. * label : The result of this expression will be the label for <option> element. The expression will most likely refer to the value variable (e.g. value.propertyName ). * select : The result of this expression will be bound to the model of the parent <select> element. If not specified, select expression will default to value . * group : The result of this expression will be used to group options using the <optgroup> DOM element. |

這說明，看了幾遍還是無法理解，尤其是 label 與 select 與 group 的應用部分，無法完整理解，自然也無法活用。而官方文件又沒有太多範例，並沒有把我有疑問的這幾點都實作出範例，所以這才抽空自己寫了 [一套完整範例](http://jsbin.com/ukezuj/5) ，相信看了之後應該可以理解。有興趣的請參考以下網址： <http://jsbin.com/ukezuj/5>

以下就用文字說明我寫的這幾個範例，以及整理出來的注意事項：

首先，先準備好 Controller 與 Model 物件，這是要提供給 View 用的：

function MySelectCtrl($scope)

{

$scope.Model = [

{

id: 10001,

MainCategory: '男',

ProductName: '水洗T恤',

ProductColor: '白'

},

{

id: 10002,

MainCategory: '女',

ProductName: '圓領短袖',

ProductColor: '黃'

},

{

id: 10002,

MainCategory: '女',

ProductName: '圓領短袖',

ProductColor: '黃'

}];

}

範例 1. **基本下拉選單** ：   
**label** **for value** **in** array

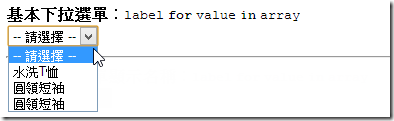
<select ng-model="Select1" ng-options="m.ProductName for m in Model">

<option value="">-- 請選擇 --</option>

</select>

* 這裡的 **value** 也就是程式碼中的 **m** 這部分，而 **m** 這個物件會從 Model 陣列中取出一個元素，一次一個元素。
* 這裡的 **label** 也就是程式碼中的 **m.ProductName** 這部分，這一段其實是一個 **Angular 表達式 (Angular Expression)** ，我們在下一個範例會有另一個複雜範例。

輸出結果：



範例 2. **自訂下拉選單顯示名稱** ：   
**label** **for value** **in** array

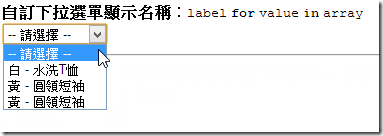
<select ng-model="Select2" ng-options="(m.ProductColor + ' - ' + m.ProductName) for m in Model">

<option value="">-- 請選擇 --</option>

</select>

* 這裡的 **value** 也就是程式碼中的 **m** 這部分，而 **m** 這個物件會從 Model 陣列中取出一個元素，一次一個元素。 看了第 2 次，你應該可以知道，這裡的 m 其實是一個變數，用來存放從陣列中取出的物件，所以這個 m 並不是一個 **Angular 表達式** ，而是一個 **區域變數** 而已。
* 這裡的 **label** 也就是程式碼中的 **(m.ProductColor + ' - ' + m.ProductName)** 這部分，這裡你就可以看出，這其實就是一個 **Angular 表達式 (Angular Expression)** ，所以可以用來顯示較為複雜的下拉選單顯示文字。

輸出結果：



範例 3. **使用 <optgroup> 依據特定欄位群組下拉選單，並自訂顯示名稱** ：   
**label** **group by group** **for value** **in** array

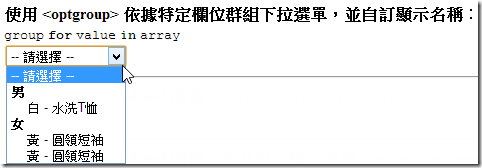
<select ng-model="Select3" ng-options="(m.ProductColor + ' - ' + m.ProductName) group by m.MainCategory for m in Model">

<option value="">-- 請選擇 --</option>

</select>

* 這裡的 **group** 也就是程式碼中的 **m.MainCategory** 這個部分，代表用來設定以這個屬性的內容當成群組的條件。而骨子裡，其實就是用 HTML 的 **<optgroup>** 標籤來做到這點。

輸出結果：



範例 4. **自訂 select 的 ngModel 的值** ： **select** **as label** **for value** **in** array

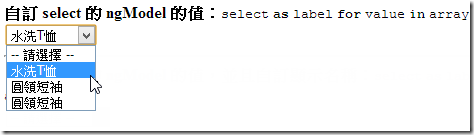
<select ng-model="Select4" ng-options="m.id as m.ProductName for m in Model">

<option value="">-- 請選擇 --</option>

</select>

* 這裡的 **select** 是我當時最不能理解的，實驗後才發現，因為我們有在 **select** 標籤上有指定一個 ng-model 屬性 ( **Select4** )，從本文章的前三個範例來說，當下拉選單被選中時，選中該項的那個 **m** 變數值，預設會自動繫結進 select 的 ng-model 指定的模型物件，在這個範例就是 **Select4** 。不過，在這個範例中，我們的 **select**   指定的是 **m.id** 物件，也代表著最後會繫結進 **Select4** 模型的值，將會是被選中的那個選項的 **m.id** 值。

輸出結果：



**請注意** : 此範例從輸出畫面完全看不出變化，但這時如果你去查 Controller 查詢 **$scope.Select4** 的值，如果你選中的是如上圖的「水洗T恤」，得到的 **$scope.Select4** 就會變成 **10001** 。

另外一個需要注意的是，請不要費心去查看這個 <select> 每一個 <option> 選項的 value 值，我指的是在 DOM 物件中每個 <option value="xxx"> 的 value 部分，因為這邊的值永遠會是 AngularJS 裡面內部物件的陣列元素索引，並不會是你所指定的 **select** ( **m.id** ) 的值。

相信有了上述範例，應該可以更加理解 select directive 裡 ngOptions 的實際用法。

**心得分享**

[AngularJS](http://angularjs.org/) 這玩意不簡單！他創造了另一個世界，顛覆了我們對 HTML 與 JavaScript 之間互動的理解，也將我們傳統 JavaScript / jQuery 的開發習慣給徹底推翻，讓開發 Web 前端應用程式的過程中，

完全 大幅排除手動操作 DOM 物件的機會，徹底做到「關注點分離」，強迫你必須抽象思考，用更加靈活的方式解決問題。

更加難能可貴的地方在於，用這種嶄新的開發架構，並 不會 犧牲效能與增加開發複雜度，反而得到的是 高效能 與 徹底降低開發的複雜度 等優點！

我想，採用 [AngularJS](http://angularjs.org/) 唯一的缺點應該是「抽象思考」這件事不是每個人都能接受的開發方法，但你只要願意採取開放思維，接受一個嶄新的開發架構，花點時間學習之後，相信你也一定會愛上它的！

**相關連結**

* [AngularJS:select](http://docs.angularjs.org/api/ng.directive:select)
* [測試 AngularJS select 的 ng-options 用法](http://jsbin.com/ukezuj/5) (這是我寫的範例)