# 7.AngularJS 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。

## AngularJS 过滤器

AngularJS 过滤器可用于转换数据：

|  |  |
| --- | --- |
| **过滤器** | **描述** |
| 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。 |

## 表达式中添加过滤器

过滤器可以通过一个管道字符（|）和一个过滤器添加到表达式中。.

(下面的两个实例，我们将使用前面章节中提到的 person 控制器)

**uppercase** 过滤器将字符串格式化为大写：

[7-1]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="personCtrl">  
  
 *<!--uppercase:格式化字符串为大写-->* <p>姓名为 {{ lastName | uppercase }}</p>  
  
</div>  
  
*<!--引用外部脚本文件的controller-->*<script src="js/personController.js"></script>  
  
</body>  
</html>

**lowercase** 过滤器将字符串格式化为小写：

[7-2]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="personCtrl">  
  
 *<!--lowercase:格式化字符串为小写-->* <p>姓名为 {{ lastName | lowercase }}</p>  
  
</div>  
  
*<!--引用外部脚本文件的controller-->*<script src="js/personController.js"></script>  
  
</body>  
</html>

## currency 过滤器

**currency** 过滤器将数字格式化为货币格式：

[7-3]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="costCtrl">  
  
 数量: <input type="number" ng-model="quantity">  
 价格: <input type="number" ng-model="price">  
  
 *<!--currency:格式化数字为货币格式-->* <p>总价 = {{ (quantity \* price) | currency }}</p>  
  
</div>  
  
<script>  
 var ***app*** = angular.module('myApp', []);  
 ***app***.controller('costCtrl', function($scope) {  
 $scope.quantity = 1;  
 $scope.price = 9.99;  
 });  
</script>  
  
</body>  
</html>

## 向指令添加过滤器

过滤器可以通过一个管道字符（|）和一个过滤器添加到指令中。

**orderBy** 过滤器根据表达式排列数组：

[7-4]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="namesCtrl">  
  
 <p>循环对象:</p>  
 <ul>  
 *<!--orderBy:根据某个表达式排列数组-->* <li ng-repeat="x in names | orderBy:'country'">  
 {{ ***x***.name + ', ' + ***x***.country }}  
 </li>  
 </ul>  
  
</div>  
  
*<!--引用外部脚本文件的controller-->*<script src="js/namesController.js"></script>  
  
</body>  
</html>

## 过滤输入

输入过滤器可以通过一个管道字符（|）和一个过滤器添加到指令中，该过滤器后跟一个冒号和一个模型名称。

**filter** 过滤器从数组中选择一个子集：

[7-5]

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <script src="js/angular.min.js"></script>  
</head>  
<body>  
  
<div ng-app="myApp" ng-controller="namesCtrl">  
  
 <p>输入过滤:</p>  
  
 <p><input type="text" ng-model="test"></p>  
  
 <ul>  
 *<!--filter:从数组中选择一个子集-->* <li ng-repeat="x in names | filter:test | orderBy:'country'">  
 {{ (***x***.name | uppercase) + ', ' + ***x***.country }}  
 </li>  
 </ul>  
  
</div>  
  
*<!--引用外部脚本文件的controller-->*<script src="js/namesController.js"></script>  
  
</body>  
</html>

## 自定义过滤器

以下实例自定义一个过滤器 **reverse**，将字符串反转：

[7-6]

<!DOCTYPE html>  
<html>  
<meta charset="utf-8">  
<script src="js/angular.min.js"></script>  
  
<body>  
  
<div ng-app="myApp" ng-controller="myCtrl">  
   
 姓名: {{ msg | reverse }}  
  
</div>  
  
<script>  
 var ***app*** = angular.module('myApp', []);  
 ***app***.controller('myCtrl', function($scope) {  
 $scope.msg = "Runoob";  
 });  
 */\*自定义过滤器reverse，将字符串反转\*/* ***app***.filter('reverse', function() { *//可以注入依赖* return function(text) {  
 return text.split("").reverse().join("");  
 }  
 });  
</script>  
  
</body>  
</html>