AngularJS Filters

[« Previous](http://www.w3schools.com/angular/angular_scopes.asp)

[Next Chapter »](http://www.w3schools.com/angular/angular_services.asp)

Filters can be added in AngularJS to format data.

AngularJS Filters

AngularJS provides filters to transform data:

* currency Format a number to a currency format.
* date Format a date to a specified format.
* filter Select a subset of items from an array.
* json Format an object to a JSON string.
* limitTo Limits an array/string, into a specified number of elements/characters.
* lowercase Format a string to lower case.
* number Format a number to a string.
* orderBy Orders an array by an expression.
* uppercase Format a string to upper case.

Adding Filters to Expressions

Filters can be added to expressions by using the pipe character |, followed by a filter.

The uppercase filter format strings to upper case:

Example

<div ng-app="myApp" ng-controller="personCtrl">  
  
<p>The name is {{ lastName | uppercase }}</p>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_uppercase)

The lowercase filter format strings to lower case:

Example

<div ng-app="myApp" ng-controller="personCtrl">  
  
<p>The name is {{ lastName | lowercase }}</p>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_lowercase)

Adding Filters to Directives

Filters are added to directives, like ng-repeat, by using the pipe character |, followed by a filter:

Example

The orderBy filter sorts an array:

<div ng-app="myApp" ng-controller="namesCtrl">  
  
<ul>  
  <li ng-repeat="x in names | orderBy:'country'">  
    {{ x.name + ', ' + x.country }}  
  </li>  
</ul>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_orderby)

The currency Filter

The currency filter formats a number as currency:

Example

<div ng-app="myApp" ng-controller="costCtrl">  
  
<h1>Price: {{ price | currency }}</h1>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_currency)

Read more about the currency filter in our [AngularJS currency Filter Reference](http://www.w3schools.com/angular/ng_filter_currency.asp)

The filter Filter

The filter filter selects a subset of an array.

The filter filter can only be used on arrays, and it returns an array containing only the matching items.

Example

Return the names that contains the letter "i":

<div ng-app="myApp" ng-controller="namesCtrl">  
  
<ul>  
  <li ng-repeat="x in names | filter : 'i'">  
    {{ x }}  
  </li>  
</ul>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_filter)

Read more about the filter filter in our [AngularJS filter Filter Reference](http://www.w3schools.com/angular/ng_filter_filter.asp)

Filter an Array Based on User Input

By setting the ng-model directive on an input field, we can use the value of the input field as an expression in a filter.

Type a letter in the input field, and the list will shrink/grow depending on the match:



* Jani
* Carl
* Margareth
* Hege
* Joe
* Gustav
* Birgit
* Mary
* Kai

Example

<div ng-app="myApp" ng-controller="namesCtrl">  
  
<p><input type="text" ng-model="test"></p>  
  
<ul>  
  <li ng-repeat="x in names | filter : test">  
    {{ x }}  
  </li>  
</ul>  
  
</div>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_input)

Sort an Array Based on User Input

Click the table headers to change the sort order::

|  |  |
| --- | --- |
| **Name** | **Country** |
| Jani | Norway |
| Carl | Sweden |
| Margareth | England |
| Hege | Norway |
| Joe | Denmark |
| Gustav | Sweden |
| Birgit | Denmark |
| Mary | England |
| Kai | Norway |

By adding the ng-click directive on the table headers, we can run a function that changes the sorting order of the array:

Example

<div ng-app="myApp" ng-controller="namesCtrl">  
  
<table border="1" width="100%">  
  <tr>  
    <th ng-click="orderByMe('name')">Name</th>  
    <th ng-click="orderByMe('country')">Country</th>  
  </tr>  
  <tr ng-repeat="x in names | orderBy:myOrderBy">  
    <td>{{x.name}}</td>  
    <td>{{x.country}}</td>  
  </tr>  
</table>  
  
</div>  
  
<script>  
angular.module('myApp', []).controller('namesCtrl', function($scope) {  
  $scope.names = [  
    {name:'Jani',country:'Norway'},  
    {name:'Carl',country:'Sweden'},  
    {name:'Margareth',country:'England'},  
    {name:'Hege',country:'Norway'},  
    {name:'Joe',country:'Denmark'},  
    {name:'Gustav',country:'Sweden'},  
    {name:'Birgit',country:'Denmark'},  
    {name:'Mary',country:'England'},  
    {name:'Kai',country:'Norway'}  
  ];  
  $scope.orderByMe = function(x) {  
    $scope.myOrderBy = x;  
  }  
});  
</script>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_orderby_click)

Custom Filters

You can make your own filters by register a new filter factory function with your module:

Example

Make a custom filter called "myFormat":

<ul ng-app="myApp" ng-controller="namesCtrl">  
    <li ng-repeat="x in names">  
        {{x | **myFormat**}}  
    </li>  
</ul>  
  
<script>

var app = angular.module('myApp', []);  
app.filter('**myFormat**', function() {  
    return function(x) {  
        var i, c, txt = "";  
        x = x.split("")  
        for (i = 0; i < x.length; i++) {  
            c = x[i];  
            if (i % 2 == 0) {  
                c = c.toUpperCase();  
            }  
            txt += c;  
        }  
        return txt;  
    };  
});  
app.controller('namesCtrl', function($scope) {  
    $scope.names = ['Jani', 'Carl', 'Margareth', 'Hege', 'Joe', 'Gustav', 'Birgit', 'Mary','Kai'];  
});

</script>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_filters_custom)

The myFormat filter will format every other character to uppercase.