

ng-grid

Angular Data Grid written in [AngularJS](#) and [jQuery](#) by the [AngularUI Team](#)

[Download ZIP](#)[Code on GitHub](#)[Download TAR](#)[Star](#)[Fork](#)

News! ng-grid has been rewritten as [UI Grid](#), it's now stable! Check it out.

Getting Started

Steps for getting started (example on right):

1. Add references to [jQuery](#) and [AngularJS](#).
2. Add references to [ngGrid](#)'s javascript and css files.
3. Where you declare your app module, add ngGrid:

```
angular.module('myApp', ['ngGrid']);
```

If you use [angular seed](#), it is in your app.js file.
4. In your html file within the controller where you plan to use ng-grid, add:

```
<div class="gridStyle" ng-grid="gridOptions"></div>
```

gridOptions is the variable we are going to bind to where we will initialize our grid options.
5. We are going to define a basic style for our grid in style.css:

HTML:

```
1. <html ng-app="myApp">
2.   <head lang="en">
3.     <meta charset="utf-8">
4.     <title>Getting Started With ngGrid Example</title>
5.     <link rel="stylesheet" type="text/css" href="ng-grid.css" />
6.     <link rel="stylesheet" type="text/css" href="style.css" />
7.     <script type="text/javascript" src="jquery-1.8.2.js"></script>
8.     <script type="text/javascript" src="angular.js"></script>
9.     <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
10.    <script type="text/javascript" src="main.js"></script>
11.  </head>
```

```
.gridStyle {
  border: 1px solid rgb(212,212,212);
  width: 400px;
  height: 300px
}
```

6. In your javascript file within the controller where you plan to use ng-grid, lets add data that our grid will use:

```
$scope.myData = [{name: "Moroni", age: 50},
  {name: "Tiancum", age: 43},
  {name: "Jacob", age: 27},
  {name: "Nephi", age: 29},
  {name: "Enos", age: 34}];
```

7. Now initialize your grid options under the same controller as data:

```
$scope.gridOptions = { data: 'myData' };
```

8. Below is the outcome of the grid using the example on the right:

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Basic Example

Below is a basic example on how to use ng-grid:

<http://angular-ui.github.io/ui-grid/>

ng-grid

```
12. <body ng-controller="MyCtrl">
13.   <div class="gridStyle" ng-grid="gridOptions">
14.     </div>
15.   </body>
16. </html>
```

CSS:

```
1. /*style.css*/
2. .gridStyle {
3.   border: 1px solid rgb(212,212,212);
4.   width: 400px;
5.   height: 300px
6. }
```

Javascript:

```
1. // main.js
2. var app = angular.module('myApp', ['ngGrid']);
3. app.controller('MyCtrl', function($scope) {
4.   $scope.myData = [{name: "Moroni", age: 50},
5.     {name: "Tiancum", age: 43},
6.     {name: "Jacob", age: 27},
7.     {name: "Nephi", age: 29},
8.     {name: "Enos", age: 34}];
9.   $scope.gridOptions = { data: 'myData' };
10. });
```

View the plunker [here](#).

Below is the outcome of the grid. View plunker [here](#).

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Column Definitions Example

Below is an example on how to define column definitions in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

Name	Age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Grouping Examples

Grouping Example - HTML5

Below is an example using HTML5 drag and drop in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

Drag a column header here and drop it to group by that column.

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Grouping Example - jQueryUI

Below is an example using jQueryUI drag and drop in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
```

Below is the outcome of the grid. View plunker [here](#).

Drag a column header here and drop it to group by that column.

name	age
Moroni	50
Tiancum	43

```

<link rel="stylesheet" type="text/css" href="ng-grid.css" />
<link rel="stylesheet" type="text/css" href="style.css" />
<script type="text/javascript" src="jquery-1.8.2.js"></script>
<script type="text/javascript" src="jquery-ui.js"></script>
<script type="text/javascript" src="angular.js"></script>
<script type="text/javascript" src="ng-grid-1.3.2.js"></script>
<script type="text/javascript" src="main.js"></script>
</head>
<body ng-controller="MyCtrl">
  <div class="gridStyle" ng-grid="gridOptions"></div>
</body>
</html>

```

Jacob	27
Nephi	29
Enos	34

Template Examples

String-based Cell Templates

Below is an example on how to define cell templates using strings in ng-grid:

HTML CSS JS

```

<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>

```

Below is the outcome of the grid. View plunker [here](#).

Name	Age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

File-referenced Cell Templates

Below is an example on how to define cell templates using file references in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

Name	Age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Row Templates

Below is an example on how to define row templates in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
```

Below is the outcome of the grid. View plunker [here](#).

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

```
</body>
</html>
```

Themes Example

Below is an example on how to use themes in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="jquery-ui-theme.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Server-Side Paging Example

Below is an example on how to use server-side paging in ng-grid:

HTML CSS JS

Below is the outcome of the grid. View plunker [here](#).

name	allowance
Moroni	505050505050505000

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Tiancum	53
Jacob	27
Nephi	29
Enos	34
Ether	42
Alma	43

Total Items: 21050

Selected Items: 0

Page Size:

250

⏪ ⏴ 1 / 85 ⏵ ⏩

Master/Details Example

Below is an example on how to make a master/details view using ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
    <div class="selectedItems">{{mySelections}}</div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Cell Selection Examples

Basic Cell Selection Example

Below is a basic example on how to use cell-selection in ng-grid. Use tab/shift tab/enter/shift enter/arrow keys to navigate.

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

Edit On Focus Cell Selection Example

Below is an example on how to edit on focus in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
```

Below is the outcome of the grid. View plunker [here](#).

Name	Age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29

Below is an example on how to edit like excel using ng-grid(double-click or start typing when a cell is focused):

Name	Age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34

<http://angular-ui.github.io/ui-grid/>

Row Selection Example

Below is a basic example on how to use row selection in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
    <script type="text/javascript" src="ng-grid-1.3.2.js"></script>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body ng-controller="MyCtrl">
    <div class="gridStyle" ng-grid="gridOptions"></div>
    <button ng-click="selectEnosRow()">Select Enos Row</button>
  </body>
</html>
```

Below is the outcome of the grid. View plunker [here](#).

name	age
Moroni	50
Tiancum	43
Jacob	27
Nephi	29
Enos	34
Select Enos Row	

Column Pinning Example

Below is a basic example on how to use column pinning in ng-grid:

HTML CSS JS

```
<html ng-app="myApp">
  <head lang="en">
    <meta charset="utf-8">
    <title>Getting Started With ngGrid Example</title>
    <link rel="stylesheet" type="text/css" href="ng-grid.css" />
    <link rel="stylesheet" type="text/css" href="style.css" />
    <script type="text/javascript" src="jquery-1.8.2.js"></script>
    <script type="text/javascript" src="angular.js"></script>
```

Below is the outcome of the grid. View plunker [here](#).

name	age	birthday	sal
Moroni	50	Oct 28, 1970	60,
Tiancum	43	Feb 12, 1985	70,
Jacob	27	Aug 23, 1983	50,
Nephi	29	May 31, 2010	40,
Enos	34	Aug 3, 2008	30,
Moroni	50	Oct 28, 1970	60,

```

<script type="text/javascript" src="ng-grid-1.3.2.js"></script>
<script type="text/javascript" src="main.js"></script>
</head>
<body ng-controller="MyCtrl">
  <div class="gridStyle" ng-grid="gridOptions"></div>
</body>
</html>

```

Tiancum	43	Feb 12, 1985	70,
Jacob	27	Aug 23, 1983	40,
			

API Documentation

Grid Options

id	default value	definition	plunker
primaryKey	undefined	Field of user's data that should be unique. Allows you automatically update selections state for previously loaded data.	Link
aggregateTemplate	<pre><div ng-click="row.toggleExpand()" ng-style="{ 'left': row.offsetleft}" class="ngAggregate">{{row.label CUSTOM_FILTERS}} {{row.totalChildren()}} {{AggItemsLabel}}<div class="ngAggregateText">{{row.aggClass()}}</div></div></pre>	Define an aggregate template to customize the rows when grouped. See github wiki for more details.	Link
afterSelectionChange	function (rowItem, event) {}	Callback for when you want to validate something after selection.	Link
beforeSelectionChange	function (rowItem, event) { return true; }	Callback if you want to inspect something before selection, return false if you want to cancel the selection. return true otherwise. If you need to wait for an async call to proceed with selection you can use rowItem.continueSelection(event) method after returning false initially. Note: when shift+ Selecting multiple items in the grid this will only get called once and the rowItem will be an array of items that are queued to be selected.	Link
checkboxCellTemplate	<pre><div class="ngSelectionCell"><input tabindex="-1" class="ngSelectionCheckbox" type="checkbox" ng-checked="row.selected" /></div></pre>	Checkbox cell template. see wiki for template details	Link
checkboxHeaderTemplate	<pre><input class="ngSelectionHeader" type="checkbox" ng-show="multiSelect" ng-model="allSelected" ng-change="toggleSelectAll(allSelected)"/></pre>	Checkbox header template. see wiki for template details	Link

columnDefs	undefined	definitions of columns as an array [], if not defines columns are auto-generated. See github wiki for more details.	Link
data	[]	Data being displayed in the grid. Each item in the array is mapped to a row being displayed.	Link
enableCellEdit	false	Globally allows all cells to be editable. use the editableCellTemplate option to override the default text input	Link
enableCellSelection	false	Enable or disable cell navigation and selection	Link
enableColumnResize	false	Enable or disable resizing of columns	Link
enableColumnReordering	false	Enable or disable reordering of columns	Link
enableHighlighting	false	Enable or disable cell content selection	Link
enablePaging	false	Enables the server-side paging feature	Link
enableRowReordering	false	Enable drag and drop row reordering. Only works in HTML5 compliant browsers.	Link
enableRowSelection	true	To be able to have selectable rows in grid. (was canSelectRows prior to 2.0)	Link
enableSorting	true	Enables or disables sorting in grid.	Link
filterOptions	{ filterText: "", useExternalFilter: false }	filterText: The text bound to the built-in search box. useExternalFilter: Bypass internal filtering if you want to roll your own filtering mechanism but want to use builtin search box.	Link
footerRowHeight	55	Defining the height of the footer in pixels.	Link
groups	[]	Initial fields to group data by. Array of field names, not displayName.	Link
groupsCollapsedByDefault	true	Collapse entries when grouping is enabled	Link
headerRowHeight	32	The height of the header row in pixels.	Link
headerRowTemplate	undefined	Define a header row template for further customization. See github wiki for more details.	Link
init	function(grid, \$scope) { }	Function to be called as soon as the grid is initialized.	Link
jqueryUIDraggable	false	Enables the use of jquery UI reaggable/droppable plugin. requires jqueryUI to work if enabled. Useful if you want drag + drop but your users insist on crappy browsers.	Link
jqueryUITheme	false	Enable the use jqueryUIThemes	Link
keepLastSelected	true	prevent unselections when ini single selection mode.	Link
maintainColumnRatios	undefined	Maintains the column widths while resizing. Defaults to true when using *'s or undefined widths. Can be overridden by setting to false.	Link
multiSelect	true	Set this to false if you only want one item selected at a time	Link
noKeyboardNavigation	false	Disables keyboard navigation	Link
paginationOptions	{ pageSizes: [250, 500, 1000], pageSize: 250,	pageSizes: list of available page sizes. pageSize: currently selected page size.	Link

	totalServerItems: 0, currentPage: 1 }	totalServerItems: Total items are on the server. currentPage: the uhm... current page.	
plugins	[]	Array of plugin functions to register in ng-grid	Link
rowHeight	30	Row height of rows in grid.	Link
rowTemplate	<code><div ng-style="{ 'cursor': row.cursor }" ng-repeat="col in renderedColumns" ng-class="col.colIndex()" class="ngCell {{col.cellClass}}"><div class="ngVerticalBar" ng-style="{height: rowHeight}" ng-class="{ ngVerticalBarVisible: !\$last }">&nbsp;</div><div ng-cell></div></div></code>	Define a row Template to customize output. See github wiki for more details.	Link
selectedItems	[]	all of the items selected in the grid. In single select mode there will only be one item in the array.	Link
selectWithCheckboxOnly	false	Disable row selections by clicking on the row and only when the checkbox is clicked.	Link
showColumnMenu	false	Enables menu to choose which columns to display and group by. If both showColumnMenu and showFilter are false the menu button will not display.	Link
showFilter	false	Enables display of the filterbox in the column menu. If both showColumnMenu and showFilter are false the menu button will not display.	Link
showFooter	false	Show or hide the footer the footer is disabled by default	Link
showGroupPanel	false	Show the dropzone for drag and drop grouping	Link
sortInfo	{ fields: [], directions: [] }	Define a sortInfo object to specify a default sorting state. You can also observe this variable to utilize server-side sorting (see useExternalSorting). Syntax is sortInfo: { fields: ['fieldName1', 'fieldName2'], directions: ['asc', 'desc']}. Directions are case-insensitive	Link
showSelectionCheckbox	false	Row selection check boxes appear as the first column. (was displaySelectionCheckbox prior to 2.0)	Link
tabIndex	0	Set the tab index of the Vieport.	Link
useExternalSorting	false	Prevents the internal sorting from executing. The sortInfo object will be updated with the sorting information so you can handle sorting (see sortInfo)	Link
virtualizationThreshold	50	The threshold in rows at which to force row virtualization on.	Link
selectRow	function (rowIndex, state)	Function that is appended to the specifec grid options for users to programatically select the row regardless of the related entity.	Link
selectItem	function (itemIndex, state)	Function that is appended to the specifec grid options for users to programatically select the row based on the index of the entity in the data array option.	Link
selectAll	function (state)	Function that is appended to the specifec grid options for users to programatically set the selected value all of the rows to the state passed.	Link

ColumnDefs Options

id	default value	definition	plunker
aggLabelFilter	"	string name for filter to use on the aggregate label ('currency', 'date', etc..) defaults to cellFilter if not set.	Link
cellClass	"	Appends a css class for the column cells	Link
cellFilter	"	string name for filter to use on the cell ('currency', 'date', etc..)	Link
cellTemplate	<div class="ngCellText" ng-class="col.colIndex()">{{COL_FIELD CUSTOM_FILTERS}}</div>	Sets the cell template for the column. See github wiki for more details.	Link
displayName	field	Sets the pretty display name of the column. default is the field given	Link
editableCellTemplate	<input ng-class="colt' + col.index" ng-input="COL_FIELD" />	Sets the editableCellTemplate for the column. Displayed when user uses enableFocusedCellEdit and cell is focused.	Link
enableCellEdit	false	Displays cellTemplate when user is not focused on cell. Displays editableCellTemplate when user focuses on cell.	Link
field	"	The string name of the property in your data model you want that column to represent. Can also be a property path on your data model. 'foo.bar.myField', 'Name.First', etc..	Link
groupable	true	Whether or not the user is allowed to group the column	Link
headerCellTemplate	<div class="ngHeaderSortColumn {{col.headerClass}}" ng-style="{ 'cursor': col.cursor}" ng-class="{ 'ngSorted': !noSortVisible }"><div ng-click="col.sort(\$event)" ng-class="colt' + col.index" class="ngHeaderText">{{col.displayName}}</div><div class="ngSortButtonDown" ng-show="col.showSortButtonDown()"></div><div class="ngSortButtonUp" ng-show="col.showSortButtonUp()"></div><div class="ngSortPriority">{{col.sortPriority}}</div><div ng-class="{ ngPinnedIcon: col.pinned, ngUnPinnedIcon: !col.pinned }" ng-click="togglePin(col)" ng-show="col.pinnable"></div></div><div ng-show="col.resizable" class="ngHeaderGrip" ng-click="col.gripClick(\$event)" ng-mousedown="col.gripOnMouseDown(\$event)"></div>	Sets the template for the column header cell. See github wiki for more details.	Link
headerClass	"	Appends a css class for the column header.	Link

maxWidth	9000	Sets the maximum width of the column.	Link
minWidth	50	Sets the minimum width of the column.	Link
pinnable	true	Whether or not column can be pinned to the left.	Link
resizable	true	Whether or not column is resizable.	Link
sortable	true	Whether or not column is sortable.	Link
sortFn	Based on the detected data type	Sets the sort function for the column. Useful when you have data that is formatted in an unusual way or if you want to sort on an underlying data type. Check the MDN sort docs for examples	Link
visible	true	Whether or not the column is visible by default. Can be overridden by the user at run-time if showColumnMenu is true.	Link
width	***	Sets the width of the column. Can be a fixed width in pixels as an int (42), string px('42px'), percentage string ('42%'), weighted asterisks (width divided by total number of *'s is all column definition widths) See github wiki for more details.	Link