DataTables example Multi-column ordering

DataTables allows ordering by multiple columns at the same time, which can be activated in a number of different ways:

* User shift click on a column (added the clicked column as a secondary, tertiary etc ordering column).
* On a per-column basis (i.e. order by a specific column and then a secondary column if the data in the first column is identical), through the [columns.orderData](http://datatables.net/reference/option/columns.orderData) option.
* Using the [columns.orderData](http://datatables.net/reference/option/columns.orderData) option to specify a multiple column order by default (for example [ 0, 1 ]).
* Through the [order()](http://datatables.net/reference/api/order()) API method.

Note that, the ability for the user to shift click to order multiple columns can be disabled through the [orderMulti](http://datatables.net/reference/option/orderMulti) option.

The example below shows the first column having a secondary order applied to the second column in the table, vice-versa for the second column being tied directly to the first and the salary column to the first name column.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| First name | Last name | Position | Office | Salary |
| Tiger | Nixon | System Architect | Edinburgh | $320,800 |
| Garrett | Winters | Accountant | Tokyo | $170,750 |
| Ashton | Cox | Junior Technical Author | San Francisco | $86,000 |
| Cedric | Kelly | Senior Javascript Developer | Edinburgh | $433,060 |
| Airi | Satou | Accountant | Tokyo | $162,700 |
| Brielle | Williamson | Integration Specialist | New York | $372,000 |
| Herrod | Chandler | Sales Assistant | San Francisco | $137,500 |
| Rhona | Davidson | Integration Specialist | Tokyo | $327,900 |
| Colleen | Hurst | Javascript Developer | San Francisco | $205,500 |
| Sonya | Frost | Software Engineer | Edinburgh | $103,600 |
| Jena | Gaines | Office Manager | London | $90,560 |
| Quinn | Flynn | Support Lead | Edinburgh | $342,000 |
| Charde | Marshall | Regional Director | San Francisco | $470,600 |
| Haley | Kennedy | Senior Marketing Designer | London | $313,500 |
| Tatyana | Fitzpatrick | Regional Director | London | $385,750 |
| Michael | Silva | Marketing Designer | London | $198,500 |
| Paul | Byrd | Chief Financial Officer (CFO) | New York | $725,000 |
| Gloria | Little | Systems Administrator | New York | $237,500 |
| Bradley | Greer | Software Engineer | London | $132,000 |
| Dai | Rios | Personnel Lead | Edinburgh | $217,500 |
| Jenette | Caldwell | Development Lead | New York | $345,000 |
| Yuri | Berry | Chief Marketing Officer (CMO) | New York | $675,000 |
| Caesar | Vance | Pre-Sales Support | New York | $106,450 |
| Doris | Wilder | Sales Assistant | Sidney | $85,600 |
| Angelica | Ramos | Chief Executive Officer (CEO) | London | $1,200,000 |
| Gavin | Joyce | Developer | Edinburgh | $92,575 |
| Jennifer | Chang | Regional Director | Singapore | $357,650 |
| Brenden | Wagner | Software Engineer | San Francisco | $206,850 |
| Fiona | Green | Chief Operating Officer (COO) | San Francisco | $850,000 |
| Shou | Itou | Regional Marketing | Tokyo | $163,000 |
| Michelle | House | Integration Specialist | Sidney | $95,400 |
| Suki | Burks | Developer | London | $114,500 |
| Prescott | Bartlett | Technical Author | London | $145,000 |
| Gavin | Cortez | Team Leader | San Francisco | $235,500 |
| Martena | Mccray | Post-Sales support | Edinburgh | $324,050 |
| Unity | Butler | Marketing Designer | San Francisco | $85,675 |
| Howard | Hatfield | Office Manager | San Francisco | $164,500 |
| Hope | Fuentes | Secretary | San Francisco | $109,850 |
| Vivian | Harrell | Financial Controller | San Francisco | $452,500 |
| Timothy | Mooney | Office Manager | London | $136,200 |
| Jackson | Bradshaw | Director | New York | $645,750 |
| Olivia | Liang | Support Engineer | Singapore | $234,500 |
| Bruno | Nash | Software Engineer | London | $163,500 |
| Sakura | Yamamoto | Support Engineer | Tokyo | $139,575 |
| Thor | Walton | Developer | New York | $98,540 |
| Finn | Camacho | Support Engineer | San Francisco | $87,500 |
| Serge | Baldwin | Data Coordinator | Singapore | $138,575 |
| Zenaida | Frank | Software Engineer | New York | $125,250 |
| Zorita | Serrano | Software Engineer | San Francisco | $115,000 |
| Jennifer | Acosta | Junior Javascript Developer | Edinburgh | $75,650 |
| Cara | Stevens | Sales Assistant | New York | $145,600 |
| Hermione | Butler | Regional Director | London | $356,250 |
| Lael | Greer | Systems Administrator | London | $103,500 |
| Jonas | Alexander | Developer | San Francisco | $86,500 |
| Shad | Decker | Regional Director | Edinburgh | $183,000 |
| Michael | Bruce | Javascript Developer | Singapore | $183,000 |
| Donna | Snider | Customer Support | New York | $112,000 |

* Javascript
* HTML
* CSS
* Ajax
* Server-side script

The Javascript shown below is used to initialise the table shown in this example:

$(document).ready(function() { $('#example').DataTable( { columnDefs: [ { targets: [ 0 ], orderData: [ 0, 1 ] }, { targets: [ 1 ], orderData: [ 1, 0 ] }, { targets: [ 4 ], orderData: [ 4, 0 ] } ] } ); } );

In addition to the above code, the following Javascript library files are loaded for use in this example:

* [//code.jquery.com/jquery-1.12.3.min.js](http://code.jquery.com/jquery-1.12.3.min.js)
* [../../media/js/jquery.dataTables.js](http://docs.google.com/media/js/jquery.dataTables.js)

The HTML shown below is the raw HTML table element, before it has been enhanced by DataTables:

This example uses a little bit of additional CSS beyond what is loaded from the library files (below), in order to correctly display the table. The additional CSS used is shown below:

The following CSS library files are loaded for use in this example to provide the styling of the table:

* [../../media/css/jquery.dataTables.css](http://docs.google.com/media/css/jquery.dataTables.css)

This table loads data by Ajax. The latest data that has been loaded is shown below. This data will update automatically as any additional data is loaded.

The script used to perform the server-side processing for this table is shown below. Please note that this is just an example script using PHP. Server-side processing scripts can be written in any language, using [the protocol described in the DataTables documentation](http://datatables.net/manual/server-side).

Other examples

[**Basic initialisation**](http://docs.google.com/index.html)

* [Zero configuration](http://docs.google.com/zero_configuration.html)
* [Feature enable / disable](http://docs.google.com/filter_only.html)
* [Default ordering (sorting)](http://docs.google.com/table_sorting.html)
* [Multi-column ordering](http://docs.google.com/multi_col_sort.html)
* [Multiple tables](http://docs.google.com/multiple_tables.html)
* [Hidden columns](http://docs.google.com/hidden_columns.html)
* [Complex headers (rowspan and colspan)](http://docs.google.com/complex_header.html)
* [DOM positioning](http://docs.google.com/dom.html)
* [Flexible table width](http://docs.google.com/flexible_width.html)
* [State saving](http://docs.google.com/state_save.html)
* [Alternative pagination](http://docs.google.com/alt_pagination.html)
* [Scroll - vertical](http://docs.google.com/scroll_y.html)
* [Scroll - vertical, dynamic height](http://docs.google.com/scroll_y_dynamic.html)
* [Scroll - horizontal](http://docs.google.com/scroll_x.html)
* [Scroll - horizontal and vertical](http://docs.google.com/scroll_xy.html)
* [Language - Comma decimal place](http://docs.google.com/comma-decimal.html)
* [Language options](http://docs.google.com/language.html)

[**Advanced initialisation**](http://docs.google.com/advanced_init/index.html)

* [DOM / jQuery events](http://docs.google.com/advanced_init/events_live.html)
* [DataTables events](http://docs.google.com/advanced_init/dt_events.html)
* [Column rendering](http://docs.google.com/advanced_init/column_render.html)
* [Page length options](http://docs.google.com/advanced_init/length_menu.html)
* [Multiple table control elements](http://docs.google.com/advanced_init/dom_multiple_elements.html)
* [Complex headers with column visibility](http://docs.google.com/advanced_init/complex_header.html)
* [Read HTML to data objects](http://docs.google.com/advanced_init/object_dom_read.html)
* [HTML5 data-\* attributes - table options](http://docs.google.com/advanced_init/html5-data-options.html)
* [HTML5 data-\* attributes - cell data](http://docs.google.com/advanced_init/html5-data-attributes.html)
* [Language file](http://docs.google.com/advanced_init/language_file.html)
* [Setting defaults](http://docs.google.com/advanced_init/defaults.html)
* [Row created callback](http://docs.google.com/advanced_init/row_callback.html)
* [Row grouping](http://docs.google.com/advanced_init/row_grouping.html)
* [Footer callback](http://docs.google.com/advanced_init/footer_callback.html)
* [Custom toolbar elements](http://docs.google.com/advanced_init/dom_toolbar.html)
* [Order direction sequence control](http://docs.google.com/advanced_init/sort_direction_control.html)

[**Styling**](http://docs.google.com/styling/index.html)

* [Base style](http://docs.google.com/styling/display.html)
* [Base style - no styling classes](http://docs.google.com/styling/no-classes.html)
* [Base style - cell borders](http://docs.google.com/styling/cell-border.html)
* [Base style - compact](http://docs.google.com/styling/compact.html)
* [Base style - hover](http://docs.google.com/styling/hover.html)
* [Base style - order-column](http://docs.google.com/styling/order-column.html)
* [Base style - row borders](http://docs.google.com/styling/row-border.html)
* [Base style - stripe](http://docs.google.com/styling/stripe.html)
* [Bootstrap 3](http://docs.google.com/styling/bootstrap.html)
* [Foundation](http://docs.google.com/styling/foundation.html)
* [jQuery UI ThemeRoller](http://docs.google.com/styling/jqueryUI.html)
* [Bootstrap 4 (Tech. preview)](http://docs.google.com/styling/bootstrap4.html)
* [Semantic UI (Tech. preview)](http://docs.google.com/styling/semanticui.html)
* [Material Design (Tech. preview)](http://docs.google.com/styling/material.html)
* [UIKit (Tech. preview)](http://docs.google.com/styling/uikit.html)

[**Data sources**](http://docs.google.com/data_sources/index.html)

* [HTML (DOM) sourced data](http://docs.google.com/data_sources/dom.html)
* [Ajax sourced data](http://docs.google.com/data_sources/ajax.html)
* [Javascript sourced data](http://docs.google.com/data_sources/js_array.html)
* [Server-side processing](http://docs.google.com/data_sources/server_side.html)

[**API**](http://docs.google.com/api/index.html)

* [Add rows](http://docs.google.com/api/add_row.html)
* [Individual column searching (text inputs)](http://docs.google.com/api/multi_filter.html)
* [Individual column searching (select inputs)](http://docs.google.com/api/multi_filter_select.html)
* [Highlighting rows and columns](http://docs.google.com/api/highlight.html)
* [Child rows (show extra / detailed information)](http://docs.google.com/api/row_details.html)
* [Row selection (multiple rows)](http://docs.google.com/api/select_row.html)
* [Row selection and deletion (single row)](http://docs.google.com/api/select_single_row.html)
* [Form inputs](http://docs.google.com/api/form.html)
* [Index column](http://docs.google.com/api/counter_columns.html)
* [Show / hide columns dynamically](http://docs.google.com/api/show_hide.html)
* [Using API in callbacks](http://docs.google.com/api/api_in_init.html)
* [Scrolling and Bootstrap tabs](http://docs.google.com/api/tabs_and_scrolling.html)
* [Search API (regular expressions)](http://docs.google.com/api/regex.html)

[**Ajax**](http://docs.google.com/ajax/index.html)

* [Ajax data source (arrays)](http://docs.google.com/ajax/simple.html)
* [Ajax data source (objects)](http://docs.google.com/ajax/objects.html)
* [Nested object data (objects)](http://docs.google.com/ajax/deep.html)
* [Nested object data (arrays)](http://docs.google.com/ajax/objects_subarrays.html)
* [Orthogonal data](http://docs.google.com/ajax/orthogonal-data.html)
* [Generated content for a column](http://docs.google.com/ajax/null_data_source.html)
* [Custom data source property](http://docs.google.com/ajax/custom_data_property.html)
* [Flat array data source](http://docs.google.com/ajax/custom_data_flat.html)
* [Deferred rendering for speed](http://docs.google.com/ajax/defer_render.html)

[**Server-side**](http://docs.google.com/server_side/index.html)

* [Server-side processing](http://docs.google.com/server_side/simple.html)
* [Custom HTTP variables](http://docs.google.com/server_side/custom_vars.html)
* [POST data](http://docs.google.com/server_side/post.html)
* [Automatic addition of row ID attributes](http://docs.google.com/server_side/ids.html)
* [Object data source](http://docs.google.com/server_side/object_data.html)
* [Row details](http://docs.google.com/server_side/row_details.html)
* [Row selection](http://docs.google.com/server_side/select_rows.html)
* [JSONP data source for remote domains](http://docs.google.com/server_side/jsonp.html)
* [Deferred loading of data](http://docs.google.com/server_side/defer_loading.html)
* [Pipelining data to reduce Ajax calls for paging](http://docs.google.com/server_side/pipeline.html)

[**Plug-ins**](http://docs.google.com/plug-ins/index.html)

* [API plug-in methods](http://docs.google.com/plug-ins/api.html)
* [Ordering plug-ins (with type detection)](http://docs.google.com/plug-ins/sorting_auto.html)
* [Ordering plug-ins (no type detection)](http://docs.google.com/plug-ins/sorting_manual.html)
* [Custom filtering - range search](http://docs.google.com/plug-ins/range_filtering.html)
* [Live DOM ordering](http://docs.google.com/plug-ins/dom_sort.html)

Please refer to the [DataTables documentation](http://www.datatables.net) for full information about its API properties and methods.

Additionally, there are a wide range of [extensions](http://www.datatables.net/extensions) and [plug-ins](http://www.datatables.net/plug-ins) which extend the capabilities of DataTables.

DataTables designed and created by [SpryMedia Ltd](http://www.sprymedia.co.uk) © 2007-2016

DataTables is licensed under the [MIT license](http://www.datatables.net/mit).