DataTables example Row details

This example shows the use of DataTables' ability to show and hide child rows which are attached to a parent row in the host table. This is often used to show additional information about a row, particularly when you wish to convey more information about a row than there is space for in the host table.

The example below shows server-side processing being used with the first column having an event listener attached to it which will toggle the child row's display. This is set up using [columns.data](http://datatables.net/reference/option/columns.data) and [columns.defaultContent](http://datatables.net/reference/option/columns.defaultContent), in combination with CSS to show an empty cell with a background image which can be clicked upon.

The event handler makes use of the [row().child](http://datatables.net/reference/api/row().child) methods to firstly check if a row is already displayed, and if so hide it, if not show it. The content of the child row is, in this example, defined by the format() function, but you would replace that with whatever you wanted to show the content required, possibly including, for example, an Ajax call to the server to obtain the extra information to show. Note that the format details function has access to the full data source object for the row, including information that is not actually shown in the table (the salary parameter for example).

Furthermore, this example shows a small difference from the [client-side row details example](http://docs.google.com/api/row_details.html) in that to have rows automatically reopen when the table is redrawn, we need to track a unique identifier for each row - in this case the row id. This is required because in server-side processing mode rows are automatically destroyed and recreated on each draw.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | First name | Last name | Position | Office |
|  | First name | Last name | Position | Office |

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

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

function format ( d ) { return 'Full name: '+d.first\_name+' '+d.last\_name+'<br>'+ 'Salary: '+d.salary+'<br>'+ 'The child row can contain any data you wish, including links, images, inner tables etc.'; } $(document).ready(function() { var dt = $('#example').DataTable( { "processing": true, "serverSide": true, "ajax": "scripts/ids-objects.php", "columns": [ { "class": "details-control", "orderable": false, "data": null, "defaultContent": "" }, { "data": "first\_name" }, { "data": "last\_name" }, { "data": "position" }, { "data": "office" } ], "order": [[1, 'asc']] } ); // Array to track the ids of the details displayed rows var detailRows = []; $('#example tbody').on( 'click', 'tr td.details-control', function () { var tr = $(this).closest('tr'); var row = dt.row( tr ); var idx = $.inArray( tr.attr('id'), detailRows ); if ( row.child.isShown() ) { tr.removeClass( 'details' ); row.child.hide(); // Remove from the 'open' array detailRows.splice( idx, 1 ); } else { tr.addClass( 'details' ); row.child( format( row.data() ) ).show(); // Add to the 'open' array if ( idx === -1 ) { detailRows.push( tr.attr('id') ); } } } ); // On each draw, loop over the `detailRows` array and show any child rows dt.on( 'draw', function () { $.each( detailRows, function ( i, id ) { $('#'+id+' td.details-control').trigger( 'click' ); } ); } ); } );

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:

td.details-control { background: url('../resources/details\_open.png') no-repeat center center; cursor: pointer; } tr.details td.details-control { background: url('../resources/details\_close.png') no-repeat center center; }

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/basic_init/index.html)

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

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