DataTables example Retain selection on reload

DataTables has the ability to use a property in the data source for each row as that row's unique identifier through the [rowId](http://datatables.net/reference/option/rowId) option. This is typically used in DataTables to assign an id attribute to the tr elements in the table, but it can also be used by Select and other libraries to retain a unique identifier for each row over data reloads.

Select uses this information to be able to retain row selection over a data reload. Generally, if a table is reloaded, the state of each row is lost (since the old rows are deleted and new rows added), but this can be undesirable, particularly if performing frequent data updates. Setting the [rowId](http://datatables.net/reference/option/rowId) option of DataTables will address this.

This example uses the extn property of the row as the [rowId](http://datatables.net/reference/option/rowId) option (normally a database primary key might be used). A reload button is also available through the [Buttons extension](https://datatables.net/extensions/buttons) which will call the [ajax.reload()](http://datatables.net/reference/api/ajax.reload()) method when activated.

To demonstrate Select's ability to retain the selected rows over a data reload, select a number of rows and then click the *Reload table* button.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Position | Office | Extn. | Start date | Salary |
| Name | Position | Office | Extn. | Start date | Salary |

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

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

$(document).ready(function() { var table = $('#example').DataTable( { ajax: '../../../../examples/ajax/data/objects.txt', deferRender: true, columns: [ { data: 'name' }, { data: 'position' }, { data: 'office' }, { data: 'extn' }, { data: 'start\_date' }, { data: 'salary' } ], rowId: 'extn', select: true, dom: 'Bfrtip', buttons: [ { text: 'Reload table', action: function () { table.ajax.reload(); } } ] } ); } );

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)
* [../../js/dataTables.select.js](http://docs.google.com/js/dataTables.select.js)
* [../../../Buttons/js/dataTables.buttons.js](http://docs.google.com/Buttons/js/dataTables.buttons.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)
* [../../css/select.dataTables.css](http://docs.google.com/css/select.dataTables.css)
* [../../../Buttons/css/buttons.dataTables.css](http://docs.google.com/Buttons/css/buttons.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

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

* [Simple initialisation](http://docs.google.com/simple.html)
* [Single item selection](http://docs.google.com/single.html)
* [Multi item selection](http://docs.google.com/multi.html)
* [Cell selection](http://docs.google.com/cells.html)
* [Checkbox selection](http://docs.google.com/checkbox.html)
* [Internationalisation](http://docs.google.com/i18n.html)
* [Blur selection](http://docs.google.com/blurable.html)
* [Defer rendering](http://docs.google.com/deferRender.html)
* [Buttons](http://docs.google.com/buttons.html)
* [Retain selection on reload](http://docs.google.com/reload.html)

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

* [Bootstrap](http://docs.google.com/styling/bootstrap.html)
* [Bootstrap 4](http://docs.google.com/styling/bootstrap4.html)
* [Foundation](http://docs.google.com/styling/foundation.html)
* [Semantic UI](http://docs.google.com/styling/semanticui.html)
* [jQuery UI](http://docs.google.com/styling/jqueryui.html)

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

* [Get selected items](http://docs.google.com/api/get.html)
* [Select items](http://docs.google.com/api/select.html)
* [Events](http://docs.google.com/api/events.html)
* [Events (cancellable)](http://docs.google.com/api/cancellableEvents.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).