

08. DATATABLES Styling.

8.1. Styling

It is important to style the DataTables enhanced tables in a manner which suits your design, so the tables fit in seamlessly with the rest of your site / app. To this end, DataTables provides a number of options for styling the tables, from the **default DataTables stylesheet** which has a number of built-in optional features, to integration with some of the most widely used **CSS frameworks** available.

Default styling options

The **default DataTables stylesheet** presents a number of different features which you can optionally enable by assigning different class names to your HTML tables, such as row or cell bordering, striped rows and hover row highlighting. The most common set of options can be enabled simply by using the **display** class, but any combination can be used.

<https://datatables.net/manual/styling/classes>

Theme creator

This online style builder will let you customise the colouring of the DataTables stylesheet to fit in seamlessly with your web-site or app. The created stylesheet still provides the class feature selection abilities of the default stylesheet, so your table matches what you need exactly.

<https://datatables.net/manual/styling/theme-creator>

ThemeForest

ThemeForest is a site where HTML/CSS authors can publish their work as reusable templates that you can purchase and use on your site to very quickly and easily create a complex site with a unified style. This is a collection of some of the themes which use DataTables.

<https://datatables.net/manual/styling/themeforest>

Bootstrap 3

Bootstrap 3 is a CSS framework that lets you quickly and easily build a user interface that has a unified look and feel. DataTables as the ability to be styled through any CSS framework allows us to provide direct integration with Bootstrap for DataTables to get beautiful tables very easily.

<https://datatables.net/manual/styling/bootstrap>

Bootstrap 4

Bootstrap 4 is the next generation of the stolid CSS framework that lets you quickly and easily build a user interface that has a unified look and feel. DataTables offers full integration with Bootstrap 4.

<https://datatables.net/manual/styling/bootstrap4>

Foundation

Foundation from Zurb is a CSS framework that allows rapid development of complex sites, featuring a mobile first approach to the design process for HTML pages. With Foundation you can create beautiful, complex and customisable sites / apps very quickly, and with the DataTables integration for Foundation your tables will have all of the options DataTables provides as well.

<https://datatables.net/manual/styling/foundation>

jQuery UI

jQuery UI is a popular suite of Javascript widgets and provides a common styling framework for these widgets through its ThemeRoller component. DataTables integration for jQuery UI provides the ability to have DataTables enhanced tables styled in the same way as other jQuery UI components, providing a common look and feel for your site if you are already using jQuery UI.

<https://datatables.net/manual/styling/jqueryui>

8.2. Default styling options

Styling your tables to fit perfectly into your site / application, is important to ensure a seamless interface for your end users. The DataTables stylesheet provides a base set of styles to position elements such as the search input box, pagination control etc, but it also has a number of features which can be optionally enabled by adding class names to your table tag.

Further to this, in addition to controlling the styled features of the table through class names, the stylesheet's colour scheme can be customised using the **DataTables theme creator**.

<https://datatables.net/manual/styling/theme-creator>

Class options

Table classes

The default DataTables stylesheet has the following class names available to control the different styling features of a DataTable. These class names should be applied to the table element:

Class name	Description
<code>display</code>	Short-hand for the <code>stripe</code> , <code>hover</code> , <code>row-border</code> and <code>order-column</code> classes.
<code>cell-border</code>	Border around all four sides of each cell
<code>compact</code>	Reduce the amount of white-space the default styling for the DataTable uses, increasing the information density on screen (since 1.10.1)
<code>hover</code>	Row highlighting on mouse over
<code>nowrap</code>	Disable wrapping of content in the table, so all text in the cells is on a single line (since 1.10.1)
<code>order-column</code>	Highlight the column that the table data is currently ordered on
<code>row-border</code>	Border around only the top and bottom of each row (i.e. for the rows). Note <code>cell-border</code> and <code>row-border</code> are mutually exclusive and cannot be used together.
<code>stripe</code>	Row striping

You can use any combination of these class names to build up the table style you want. They will each work with the other options available (*with the exception of **cell-border** and **row-border***) tinting and shading the base colours as required.

Each option is demonstrated individually in an example table below, so you can see how each component will style the table. Remember that you can add multiple classes to each table, for example you can have **`class="stripe hover"`** to generate a table which shows both row striping and mouse hover styling.

Example: To make use of these classes, your table element's **class** attribute can be set:

```
<table id="myTable" class="cell-border compact stripe">
  <thead>
    ...
  </thead>
  <tbody>
    ...
  </tbody>
</table>
```

```
</tbody>
</table>
```

Cell classes

In addition to the full table styling options above, the DataTable stylesheet also has basic options for the **styling of cells**. These can be applied to class individually using HTML, or use **columns.className** to apply them to all cells in a column.

Class name	Description
<code>dt[-head -body]-left</code>	Left aligned text
<code>dt[-head -body]-center</code>	Center aligned text
<code>dt[-head -body]-right</code>	Right aligned text
<code>dt[-head -body]-justify</code>	Justify aligned text
<code>dt[-head -body]-nowrap</code>	nowrap aligned text

For convenience when using **columns.className**, optionally you can use **-head** or **-body** in the class name to indicate if the text alignment should be applied to the cells in the **thead** or **tbody** of the table. Omit either option to have it apply to both (as *columns.className* is added to the header and body cells). For example:

- **dt-right** - Right align text in the header and body.
- **dt-head-right** - Right align text in the header only
- **dt-body-right** - Right align text in the body only.

Example column alignment

The following shows the **columns.className** option being used to right align the text in the final column of a table:

Example:

```
$('#myTable').DataTable( {
  columnDefs: [
    {
      targets: -1,
      className: 'dt-body-right'
    }
  ]
} );
```

8.3. Examples

display

Short-hand for stripe, hover, row-border and order-column.

cell-border

Border around all four sides of each cell.

compact

Reduce the amount of white-space the default styling for the DataTable uses, increasing the information density on screen, as shown below.

hover

Row highlighting on mouse hover.

nowrap

Disable line wrapping of content in the table cells, so the text will always appear on one line.

order-column

Highlight the ordering column.

row-border

Border on the rows only.

stripe

Row striping.

8.4. Theme Creator.

This online style builder will let you customise the colouring of the DataTables stylesheet to fit in seamlessly with your web-site or app. Simply select the colours you want for your table using the options below, and click the Create stylesheet button. The page will then reload and show your customised table below, with the CSS available to copy and paste for your site.

<https://datatables.net/manual/styling/theme-creator>

The DataTables stylesheet is written in **SCSS**, making it very easy to customise - providing the ability to create this online style designer! If you want to customise your stylesheet beyond the options of this form, you are very much encouraged to do so - this is where the fun is! The SCSS used to create the table is also shown below.

<https://sass-lang.com/>

Style Selector

- **Table section border:** The table section border is applied between the table header and the table body and between the table body and the table footer. If there is no `dt-tag tfoot` element, it is applied to the bottom of the table.
- **Row / cell border:** The row and cell border is applied to the rows and / or cells depending on the use of the cell-border and row-border classes for the table.
- **Row background:** The row background colour is automatically tinted for mouse hover, row striping and sort column classes.
- **Row selected:** When a row is selected using **TableTools** this is the background colour it will receive. Like the row background colour, it is automatically tinted to fit in with the other features of the stylesheet.
- **Control text:** The control text colour is applied to the elements DataTables adds around the table, including the pagination controls, search input and information display.
- **Paging button active:** The colour of the active paging button (*that which the current page is shown by*) is defined by this colour. Note that non-active paging buttons have a transparent background. A subtle gradient is automatically applied to the button.
- **Paging button hover:** When the mouse is hovered over a paging button, this colour is used. As with the active button, a subtle gradient is automatically applied.

8.5. Bootstrap 3

Bootstrap 3 is a CSS framework that lets you quickly and easily build a user interface that has a unified look and feel. It is super easy to integrate Bootstrap and DataTables so that your fully featured DataTables will match the same look and feel of the rest of your site.

Bootstrap 3 provides a number of options for styling HTML tables, giving you flexibility over the table, while ensuring they look very slick. The DataTables / Bootstrap integration ensures that you can use all of these features as well as the enhancements that DataTables makes to a plain HTML table.

Installation

The simplest way to include the Bootstrap integration for DataTables and its extensions is to **use the DataTables download builder**. This is a point and click interface that lets you select what styling and what software you wish to use. The download builder has the option of hosting the required files on the DataTables **CDN** or **downloading a package** that you can host locally.

DataTables

The main DataTables file has the name *jquery.dataTables.js* using the format that is common to the majority of jQuery plug-ins. There is also a styling integration file for the various styling libraries supported by DataTables, this has the format *dataTables.{style}.js* (*dataTables.bootstrap.js* for *Bootstrap* for example).

Both of these Javascript files should be included - the main file to include the basic DataTables functionality and the styling file to set defaults as appropriate for the styling library.

For CSS only a single file should be included as the styling library will provide the core styling options for the table. The CSS files have the format *dataTables.{style}.css* (*dataTables.bootstrap.css* for *Bootstrap* for example).

Extensions

The file naming conventions described above are also applied to the extensions.

For Javascript we retain the two part name and drop the jquery. part, so now we have: *dataTables.{extension}.js* and *{extension}.{style}.js* - for example for Buttons the files to include are *dataTables.buttons.js* and *buttons.bootstrap.js* - again the core library is included and the styling file to set appropriate defaults.

In the case of the CSS, only a single file should be included - the appropriate file for the styling being used - *{extension}.{style}.css* - *buttons.bootstrap.css* for example. Please note that not all extensions require Javascript and / or CSS for the styling libraries. If the styling libraries are not present in the repo or package, then they are not required.

8.4. Bootstrap 4

Bootstrap 4 is the next generation of the stolid CSS framework that lets you quickly and easily build a user interface that has a unified look and feel. DataTables offers full integration with Bootstrap 4 so that your DataTables will match the same look and feel of the rest of your site.

Bootstrap 4 provides a number of options for styling HTML tables, giving you flexibility over the table, while ensuring they look very slick. The DataTables / Bootstrap integration ensures that you can use all of these features as well as the enhancements that DataTables makes to a plain HTML table.

Installation

The simplest way to include the Bootstrap 4 integration for DataTables and its extensions is to use the DataTables download builder. This is a point and click interface that lets you select what styling and what software you wish to use. The download builder has the option of hosting the required files on the **DataTables CDN** or **downloading a package** that you can host locally.

<https://datatables.net/download>

DataTables

The main DataTables file has the name *jquery.dataTables.js* using the format that is common to the majority of jQuery plug-ins. There is also a styling integration file for the various styling libraries supported by DataTables, this has the format *dataTables.{style}.js* (*dataTables.bootstrap4.js* for Bootstrap 4 for example).

Both of these Javascript files should be included - the main file to include the basic DataTables functionality and the styling file to set defaults as appropriate for the styling library.

For CSS only a single file should be included as the styling library will provide the core styling options for the table. The CSS files have the format *dataTables.{style}.css* (*dataTables.bootstrap4.css* for Bootstrap 4 for example).

Extensions

The file naming conventions described above are also applied to the extensions. For Javascript we retain the two part name and drop the jquery. part, so now we have: *dataTables.{extension}.js* and *{extension}.{style}.js* - for example for Buttons the files to include are *dataTables.buttons.js* and *buttons.bootstrap4.js* - again the core library is included and the styling file to set appropriate defaults.

In the case of the CSS, only a single file should be included - the appropriate file for the styling being used - *{extension}.{style}.css* - *buttons.bootstrap4.css* for example.

Please note that not all extensions require Javascript and / or CSS for the styling libraries. If the styling libraries are not present in the repo or package, then they are not required.