Here is a comprehensive list of DataTables options, each with its purpose and an example of how to use it. This list covers the main configuration options you can set in DataTables to control its behavior. Each example is provided in the format that would be used inside the DataTables initialization script.

**1. Basic Options**

These options are commonly used to enable basic features like pagination, searching, and sorting.

* **paging**: Enables or disables pagination.

javascript

Copy code

"paging": true // Enables pagination

* **searching**: Enables or disables the search/filter box.

javascript

Copy code

"searching": true // Enables searching

* **ordering**: Allows column ordering by clicking headers.

javascript

Copy code

"ordering": true // Enables ordering

* **lengthChange**: Shows a dropdown to change the number of records per page.

javascript

Copy code

"lengthChange": true // Allows changing records per page

* **info**: Displays table information (e.g., "Showing 1 to 10 of 100 entries").

javascript

Copy code

"info": true // Enables table info display

* **autoWidth**: Automatically adjusts column widths.

javascript

Copy code

"autoWidth": true // Enables automatic column width adjustment

**2. Customizing Pagination**

These options control pagination settings.

* **pageLength**: Sets the default number of records displayed per page.

javascript

Copy code

"pageLength": 10 // Sets default to 10 records per page

* **pagingType**: Controls pagination display style.
  + Options: simple, simple\_numbers, full, full\_numbers

javascript

Copy code

"pagingType": "full\_numbers" // Displays full pagination controls

**3. Column Configuration**

These options define behavior for individual columns.

* **columns**: Configures each column in the table (e.g., sorting, visibility).

javascript

Copy code

"columns": [

{ "name": "Name", "orderable": true, "visible": true },

{ "name": "Category", "orderable": false, "visible": true },

{ "name": "Price", "orderable": true, "visible": true }

]

* **order**: Sets default ordering on a specific column.

javascript

Copy code

"order": [[0, "asc"]] // Orders by the first column in ascending order

* **orderMulti**: Allows multi-column sorting.

javascript

Copy code

"orderMulti": true // Enables multi-column sorting

**4. Data Source Options**

These options help control the data source for DataTables, especially useful for AJAX.

* **data**: Supplies data directly to the table.

javascript

Copy code

"data": myDataArray // Populates table with an array of data

* **ajax**: Loads data via an AJAX request.

javascript

Copy code

"ajax": "https://api.example.com/products" // Loads data from an external source

**5. Language Customization**

These options control the text displayed by DataTables (such as pagination buttons and "No data available" messages).

* **language**: Sets text for various labels.

javascript

Copy code

"language": {

"emptyTable": "No data available",

"info": "Showing \_START\_ to \_END\_ of \_TOTAL\_ entries",

"search": "Filter records:",

"lengthMenu": "Display \_MENU\_ records per page",

"paginate": {

"first": "First",

"last": "Last",

"next": "Next",

"previous": "Previous"

}

}

**6. Buttons for Exporting**

These options enable export buttons (requires the Buttons extension).

* **dom**: Customizes the table layout, including buttons.

javascript

Copy code

"dom": "Bfrtip" // Positions buttons at the top of the table

* **buttons**: Adds buttons for exporting and printing.

javascript

Copy code

"buttons": ["copy", "csv", "excel", "pdf", "print"]

**7. Advanced Search and Filtering**

These options help to control advanced search/filtering features.

* **search**: Allows default search term setting.

javascript

Copy code

"search": {

"search": "initial search text"

}

* **searchCols**: Sets individual column search values.

javascript

Copy code

"searchCols": [

{ "search": "Category A" }, // Filter first column for "Category A"

null, // No filter for the second column

{ "search": ">100" } // Filter third column for values greater than 100

]

**8. Row and Cell Configuration**

These options customize row or cell appearance and behavior.

* **createdRow**: Applies custom styles or classes to rows after they’re created.

javascript

Copy code

"createdRow": function(row, data, index) {

if (data[4] > 100) { // Applies a class if column 5 value > 100

$(row).addClass("highlight");

}

}

* **columnDefs**: Sets column-specific options (e.g., disabling ordering or changing class).

javascript

Copy code

"columnDefs": [

{ "orderable": false, "targets": [0, 2] }, // Disables ordering for columns 1 and 3

{ "className": "text-center", "targets": [1, 4] } // Centers text in columns 2 and 5

]

**9. State Saving**

These options allow saving the state of the table (pagination, search, etc.) across page reloads.

* **stateSave**: Enables state saving.

javascript

Copy code

"stateSave": true // Saves the table state

* **stateDuration**: Controls how long the state is saved (in seconds).

javascript

Copy code

"stateDuration": 86400 // Saves state for one day (in seconds)

**10. Callback Functions**

Callback functions provide hooks at various stages of DataTables operation.

* **initComplete**: Runs when DataTables finishes initializing.

javascript

Copy code

"initComplete": function(settings, json) {

console.log("DataTables has been initialized");

}

* **drawCallback**: Runs every time the table is redrawn.

javascript

Copy code

"drawCallback": function(settings) {

console.log("DataTables has redrawn the table");

}

**Example: Full DataTables Initialization**

Here’s a full example incorporating many of the options listed:

javascript

Copy code

<script>

$(document).ready(function() {

$('#productTable').DataTable({

"paging": true,

"searching": true,

"ordering": true,

"lengthChange": true,

"pageLength": 10,

"pagingType": "full\_numbers",

"order": [[1, "asc"]],

"stateSave": true,

"dom": 'Bfrtip',

"buttons": ["copy", "csv", "excel", "pdf", "print"],

"language": {

"emptyTable": "No records available",

"search": "Filter products:"

},

"createdRow": function(row, data, index) {

if (data[4] > 100) {

$(row).addClass("highlight");

}

},

"columnDefs": [

{ "orderable": false, "targets": [0] },

{ "className": "text-center", "targets": [1, 3] }

]

});

});

</script>

This setup initializes DataTables with features like pagination, searching, custom ordering, export buttons, and specific row styling conditions.