**Disable Filtering for the ListOfBusIds Column**

Since filtering is not required for ListOfBusIds, we can turn it off to prevent Kendo from trying to attach a filter widget to that column.

Update your **Grid column configuration** for ListOfBusIds in Index.cshtml:

csharp

CopyEdit

columns.Bound(c => c.ListOfBusIds)

.Hidden(true) // Hide this column since it’s just for editor purposes

.Filterable(false) // Disable filtering for this column

.EditorTemplateName("VehicleSeriesEditor");

Here:

* .Hidden(true) hides the column from the grid since it's just used for editing.
* .Filterable(false) disables filtering for the column, avoiding the kendoFilterCell error.

**2. Ensure Kendo Grid Doesn't Try to Handle Filtering Internally**

If filtering is enabled globally in the grid, Kendo will still attempt to filter this column. To avoid this, make sure to explicitly disable filtering where required.

Update the grid configuration to ensure proper filtering behavior:

csharp

CopyEdit

@(Html.Kendo().Grid<RebuiltPartsViewModel>()

.Name("RebuiltPartsGrid")

.Columns(columns =>

{

columns.Bound(c => c.RebuiltStockNum)

.Title("Rebuilt Stock Code")

.Filterable(ftb => ftb.Cell(cell => cell.Operator("contains").SuggestionOperator(FilterType.Contains)));

columns.Bound(c => c.ListOfBus)

.Title("Vehicle Series")

.ClientTemplate("# if (ListOfBus && ListOfBus.length > 0) { # #= ListOfBus.map(bus => bus.Description).join(', ') # # } else { # - # } #");

columns.Bound(c => c.ListOfBusIds)

.Hidden(true) // Keep this column hidden from the grid view

.Filterable(false) // Explicitly disable filtering for this column

.EditorTemplateName("VehicleSeriesEditor");

columns.Command(command =>

{

command.Edit().Text("Update");

command.Destroy().Text("Delete");

}).Title("&nbsp;").Width(200);

})

.ToolBar(toolbar => toolbar.Create().Text("Add New Part"))

.Editable(editable => editable.Mode(GridEditMode.PopUp)) // Use PopUp editing mode

.Pageable()

.Sortable()

.Scrollable()

.Filterable() // Enable filtering for other columns globally

.DataSource(dataSource => dataSource

.Ajax()

.Read(read => read.Url(Url.Action("GetRebuiltPartsView", "RebuiltParts")).Type(HttpVerbs.Get))

.Update(update => update.Url(Url.Action("UpdateRebuiltPart", "RebuiltParts")).Type(HttpVerbs.Post))

.Destroy(destroy => destroy.Url(Url.Action("DeleteRebuiltPart", "RebuiltParts")).Type(HttpVerbs.Post))

.Model(model =>

{

model.Id(p => p.RebuiltStockNum);

model.Field(p => p.ListOfBusIds).DefaultValue(new List<int?>());

})

)

)

**3. Correct jQuery Version**

Ensure you’re using a jQuery version that’s compatible with Kendo UI 2021.2.511. For this version of Kendo, the following versions of jQuery are supported:

* **jQuery 1.12.x** (older, but stable with Kendo)
* **jQuery 3.5.x** (modern, compatible with Kendo)

If you’re using jQuery 3.7.1, there could be compatibility issues. Try rolling back to 3.5.1 to avoid potential issues.

Include the compatible jQuery version in your layout file or HTML:

html

CopyEdit

<script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>

<script src="https://kendo.cdn.telerik.com/2021.2.511/js/kendo.all.min.js"></script>

<link href="https://kendo.cdn.telerik.com/2021.2.511/styles/kendo.default-v2.min.css" rel="stylesheet" />

**4. Manually Initialize the MultiSelect**

To ensure the plain HTML <select> element is properly initialized in the editor, manually trigger any required behavior after the editor opens.

Update the editor script in VehicleSeriesEditor.cshtml:

html

CopyEdit

@model List<int?>

<select id="VehicleSeries" name="ListOfBusIds" class="form-control" multiple>

<option value="">-- Select Vehicle Series --</option>

</select>

<script>

$(document).ready(function () {

// Populate the MultiSelect with vehicle series options

$.getJSON('@Url.Action("GetVehicleSeries", "RebuiltParts")', function (data) {

var select = $("#VehicleSeries");

select.empty(); // Clear any existing options

$.each(data, function (index, item) {

select.append($("<option>").val(item.listId).text(item.description));

});

// Pre-select the current values

var selectedValues = @Html.Raw(Json.Serialize(Model));

select.val(selectedValues);

});

// Re-initialize after popup opens

$(document).on("kendo:editor-opened", function () {

$("#VehicleSeries").trigger("change");

});

});

</script>

**Additional Debugging**

If the error persists, enable Kendo debug mode and check the stack trace for more specific details.

1. Replace kendo.all.min.js with the unminified version:

html

CopyEdit

<script src="https://kendo.cdn.telerik.com/2021.2.511/js/kendo.all.js"></script>

1. Check the **browser console** for detailed error messages and trace the issue to a specific line of code.

**Summary of Fixes**

1. **Disable filtering** for the ListOfBusIds column to prevent Kendo from attaching a filter widget.
2. **Hide the ListOfBusIds column** in the grid since it’s only used for editing.
3. Use a **compatible jQuery version** (e.g., 3.5.1) to avoid runtime errors.
4. **Manually initialize the MultiSelect** in the editor template and re-trigger it when needed.