

DataTable Plus

App Overview:

DataTable Plus app extends the OOTB lightning datatable with more powerful features that are highly needed and helpful.

Currently the app adds the following two features to the OOTB Lightning Datatable:

- Customize the Lightning Datatable to support using a drop-down menu within the table.
- Pagination feature for the rendering of the datatable.

Installation instruction:

To integrate DataTable Plus into your Salesforce environment, please follow these steps:

1. Install the DataTable Plus app.
2. Create or update your Lightning Web Components (LWC) and make sure to pass the required parameters for the DataTable Plus components.

And below are the details for the application components and how to use them:

App components:

The DataTable Plus app contains the following two components:

- **ITGSoftware-pro-data-table**

This component extends the Lightning Datatable to create a new data type named 'picklist' that provides ability for using the dropdown menu in the datatable. In addition to continue supporting the OOTB data types for the Lightning Datatable component.

Specification:

Attributes:

Support all standard attributes in the lightning datatable.

Methods:

Support all standard methods in the lightning datatable.

Event:

- Support all standard events in the lightning datatable.
- New Event:

onpicklistchanged: The **onpicklistchanged** event is triggered when user selects a value from the dropdown list.

➤ Structure of **onpicklistchanged** event object:

The event contains “detail” object which has properties that allow the developers to access the data in the dropdown.

The detail object has the following attributes:

- **context:** Used to determine the row that has a change in the drop-down value. And it should match the unique id value in the record that will display in the table.
- **value:** It represents the selected value in the drop-down list and allows you to define the default value displayed in the drop-down cell.
- **picklistid:** Used to determine which drop-down triggers the event in case there are more than one drop-down in the datatable.

Notes:

The developer needs to set those attributes when creating and configuring the column object for the datatable.

➤ Handling the **onpicklistchanged** event:

To handle the event, define the handler function in the component's JavaScript class. Inside the handler function, developers can add their own implementation and access the drop-down selected value and the changed row using the data in the event object. Additionally, they can implement a code to save the drop-down selected value in the database.

The below screenshots show how you can use the **onpicklistchanged** event:

Html File

```
<template>
  <div>
    <ITGSoftware-pro-data-table
      key-field="Id"
      data={data}
      columns={columns}
      onpicklistchanged={handlePicklistChanged}>
    </ITGSoftware-pro-data-table>
  </div>
</template>
```

JS File

```
handlePicklistChanged(event) {  
    /*  
        Add your custom implementation  
  
        You can access "detail" properties in this way :  
        event.detail.value  
        event.detail.context  
        event.detail.picklistid  
  
    */  
}
```

Defining a column of type 'picklist' in the datatable's columns:

To customize the behavior and visual aspects for the drop-down columns, pass the attributes of the picklist component via the **typeAttributes** object.

The properties that you need to set in **typeAttributes** object:

- **placeholder**: Text that is displayed before an option is selected, to prompt the user to select an option.
- **options**: A list of options that are available for selection. Each option has the following attributes: label and value.
- **value**: Specifies the value of an input element and you can use it to define the default value displayed in the drop-down cell.
- **context**: it must correspond to the unique identifier field in the record that is displayed in the datatable.
- **picklistid**: Specifies the id for the drop-down column.

The following screenshot shows an example for configuring the datatable's columns to display a Contact object records and this object contains a custom field of type Picklist named 'Region__c':

```
const columns = [  
  { label: 'First Name', fieldName: 'FirstName' },  
  { label: 'Last Name', fieldName: 'LastName' },  
  { label: 'Title', fieldName: 'Title' },  
  { label: 'Phone', fieldName: 'Phone', type: 'phone' },  
  { label: 'Email', fieldName: 'Email', type: 'email' },  
  {  
    label: 'Region', type: 'picklist', fieldName: 'Region__c', typeAttributes: {  
      placeholder: 'Select Region', options: [  
        { label: 'East Asia', value: 'East Asia' },  
        { label: 'Oceania', value: 'Oceania' },  
        { label: 'Europe', value: 'Europe' },  
        { label: 'Caribbean', value: 'Caribbean' },  
        { label: 'Northern Canada', value: 'Northern Canada' },  
        { label: 'Africa', value: 'Africa' },  
        { label: 'West US', value: 'West US' }  
      ]  
    },  
    value: { fieldName: 'Region__c' },  
    context: { fieldName: 'Id' },  
    picklistid: 'selectedRegion'  
  }  
];
```

- [ITGSoftware-datatable-paginator](#)

This component allows displaying the data in the datatable with pagination features.

Below screenshot shows how to use the component in html file:

```
<ITGSoftware-datatable-paginator items={items} page-size={pageSize} onpagination={handlePagination}>
</ITGSoftware-datatable-paginator>
```

Specification:

Attributes:

items: Represent all the records that you need to display in the data table. The developer must set the items property in JavaScript class and pass it to the items attribute in the component.

page-size: Determines the number of rows you need to display per page. The developer must set the pageSize property in JavaScript class and pass it to the page-size attribute in the component.

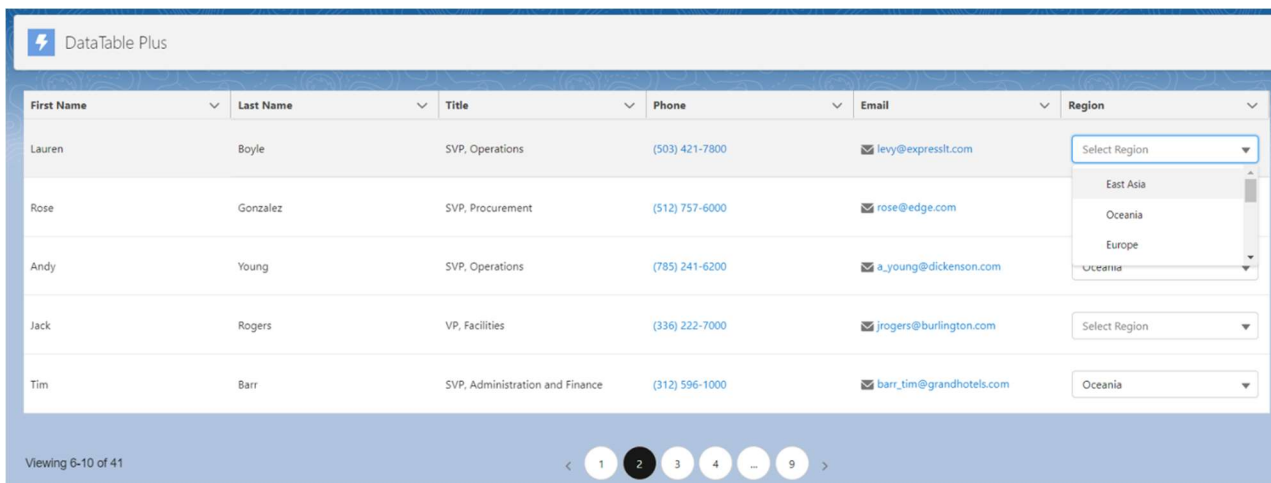
onpagination: This event is triggered when the customer clicks on the pagination buttons.

- The developer must handle the event by defining a handler method in the component's JavaScript class to assign the data that should be displayed for the pressed page.
- The data retrieved from the event should be passed to the data attribute in the data table.

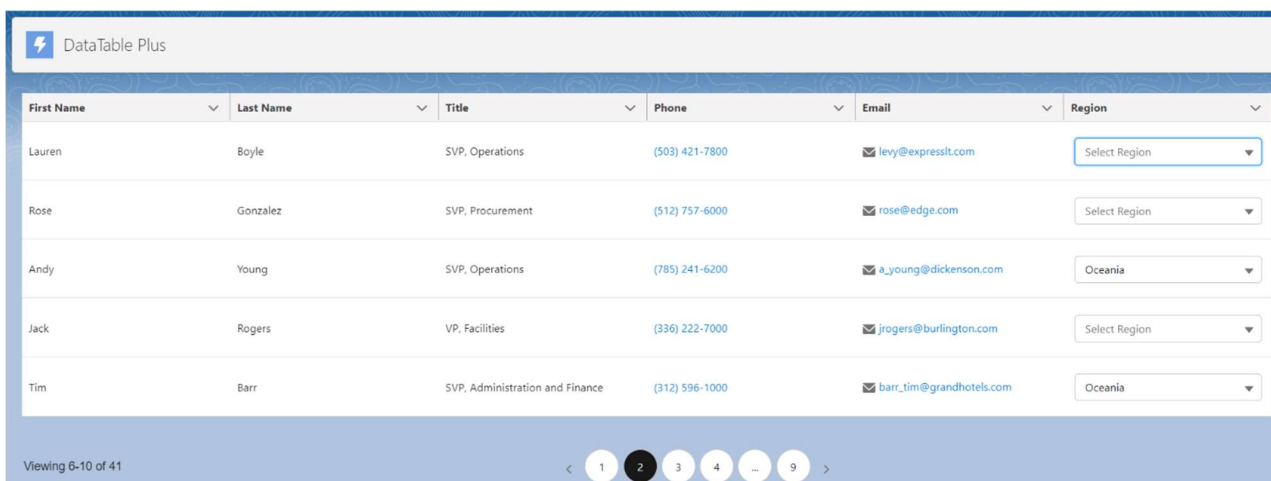
The following screenshot shows the implementation of the handler event:

```
handlePaging(event){
  this.data = event.detail;
}
```

The following screenshots show an example of displaying the Contact records in the table using the DataTable Plus application:



First Name	Last Name	Title	Phone	Email	Region
Lauren	Boyle	SVP, Operations	(503) 421-7800	levy@expressit.com	Select Region
Rose	Gonzalez	SVP, Procurement	(512) 757-6000	rose@edge.com	East Asia
Andy	Young	SVP, Operations	(785) 241-6200	a_young@dickenson.com	Oceania
Jack	Rogers	VP, Facilities	(336) 222-7000	jrogers@burlington.com	Europe
Tim	Barr	SVP, Administration and Finance	(312) 596-1000	barr_tim@grandhotels.com	Oceania



First Name	Last Name	Title	Phone	Email	Region
Lauren	Boyle	SVP, Operations	(503) 421-7800	levy@expressit.com	Select Region
Rose	Gonzalez	SVP, Procurement	(512) 757-6000	rose@edge.com	Select Region
Andy	Young	SVP, Operations	(785) 241-6200	a_young@dickenson.com	Oceania
Jack	Rogers	VP, Facilities	(336) 222-7000	jrogers@burlington.com	Select Region
Tim	Barr	SVP, Administration and Finance	(312) 596-1000	barr_tim@grandhotels.com	Oceania