Linx-AS SAPUI5 Custom Controls

Excel Export Button

Contents

Overview

The Linx-AS Excel Export Button control allows you to provide users the ability to quickly and easily download data from any sap.m.ListBase control. The control can be configured to provide an automatic download or can be configured to provide users the ability to customize what is downloaded.

Capabilities

The ExcelExportButton is linked to a sap.m.ListBase element in the same view by it's id. It uses both the sap.m.ListBase items as well as the associated model to retrieve data and provide that data into an Excel Spreadsheet. At it's most basic configuration the listBaseID value must be defined, and it must be able to find a sap.m.ListBase in the same view with that id, or the button will not be displayed.

Base Parameters

Below are the most basic parameters:

listBaseID {String}

> This is the only property that is required and has no default. This must be the id of a sap.m.ListBase element (List, Table, Tree, GridList, NotificationList) in the same view. If an element with the provided id cannot be found, or that element is not an extension of sap.m.ListBase then the ExcelExportButton will not be rendered and an error will be reported in the console.

fileName {String} default: 'Export.xlsx'

> The file name to use *for* the generated Excel workbook.

sheetName {String} default: "Export Data"

> The sheet name to use *in* the generated Excel workbook.

Defining Properties

The properties aggregation allows you to define 0..n ExcelExportProperty elements. These elements allow you to:

1. Define which properties are exported, or if the showAllColumns property is set to true, define which properties are the defaults. In this scenario, properties defined in the properties aggregation are considered default or defined properties, any other discovered non-array based properties are considered undefined properties.
2. Specify a specific column label for a property that is different than the human readable label that is generated.
3. Specify a data type for the property, rather than the default String type.

The ExcelExportProperty element has the following properties:

label {String}

> The column label to use for the property

value {String}

> The property name in the object to enter in the column

type {sap.ui.export.EdmType}

> The expected data type for the property

NOTE: Any properties displayed in the selection dialog that are not defined in the properties aggregation (i.e. if showAllColumns is set to true) will have a label that is an attempt at a human readable word case label based on the property name. The basic algorithm used is to:

1. Replace any '_x' sections with 'X'
2. Capitalize the first character
3. Replace an 'xX' sections with ' X'

For example:

> userId would become User Id

> reported_by would become Reported By

Modes of operation

The Excel Export Button has two modes of operation:

1. Pre-configured setup that provides an automatic download of data from a sap.m.ListBase model to Excel.
2. A user-customizable download experience, where the user can select which properties to download, and whether to download all available data or just currently loaded data.

When using the user-customizable configuration, the user is presented with a dialog box that allows them to decide what they want to download. There are some settings you can provide to control what is available to the user to select:

- Allow the user to select from all non-array based properties in the model object's data or only from a pre-defined list of properties.
- Define a list of "default" properties that are automatically selected in the dialog through the properties aggregation.
- Allow the user the option to download all data from the parent model, or only the rows that are currently loaded in the associated sap.m.ListBase element.

Using the automatic download configuration

To use the Excel Export Button to automatically download data to Excel, without providing a customizable dialog, you must either not define the showSelectionDialog property or set it to false.

When the user clicks the Excel Export Button in this configuration, download of the Excel spreadsheet will happen instantly.

Relevant Properties

When using the automated configuration, the following properties are relevant:

exportAllRows {Boolean} default: false

> If set to true, then all the available data from the model will be exported. If set to false, then only the data that is currently loaded into the associated sap.m.ListBase will be exported.

exportAllColumns {Boolean} default: false

> If set to true, then all non-array based properties in the model's objects will be exported. If set to false, then only the properties defined in the properties aggregation will be exported.

> NOTE: If this property is set to false, and the properties aggregation is empty, then all properties will be exported as if set to true.

All other properties area ignored when using this configuration.

Examples

Basic example

In this example, we will use the most basic setup. It will read the model that is bound to the sap.m.ListBase element with id 'list', will export all the properties defined in the objects of that model, and will export only the rows that have been loaded into the associated sap.m.ListBase element:

```
<comp:ExcelExportButton listBaseId="list"/>
```

To export all rows, just set the exportAllRows property to true:

```
<comp:ExcelExportButton listBaseId="list" exportAllRows="true"/>
```

Examples with defined properties

In this example, we will define which properties we want to have exported, providing custom column labels. It will read the model that is bound to the sap.m.ListBase element with id 'list' and export only the properties defined in the properties aggregation (id, title, description and reportedBy). It will export only the rows that have been loaded into the sap.m.ListBase element:

```
<comp:ExcelExportButton listBaseId="list">
      <comp:properties>
            <comp:ExcelExportProperty label="Incident ID"
                  value="id" type="Number"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Reported By"
                  value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

To export all rows, just set the exportAllRows property to true:

```
<comp:ExcelExportButton listBaseId="list" exportAllRows="true">
      <comp:properties>
            <comp:ExcelExportProperty label="Incident ID"
                  value="id" type="Number"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Reported By"
                  value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

To export all properties, not just the ones defined in the properties aggregation, set exportAllColumns to true:

```
<comp:ExcelExportButton listBaseId="list" exportAllRows="true"
 exportAllColumns="true">
      <comp:properties>
            <comp:ExcelExportProperty label="Incident ID"
                  value="id" type="Number"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Reported By"
                  value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

Using the selection dialog configuration

To use the selection dialog configuration, you must first set the showSelectionDialog property to true.

When the user clicks the Excel Export Button in this configuration, they will be presented with a selection dialog that allows them to make choices about what and how much data is exported based on the other properties defined in the ExcelExportButton constructor. Excel export will begin when the Export button is clicked in this dialog.

Relevant Properties

When using this configuration, the following properties are relevant:

showOnlyDefinedProperties {Boolean} default: true

> If set to true, the only the properties defined in the properties aggregation will be available for the user to select for export. If set to false, then all the properties defined in the model's objects will be available to select for export.
>
> NOTE: If set to false, and the properties aggregation is empty, then all properties will be displayed.

showDefinedPropertiesFirst {Boolean} default: false

> If set to true, then the properties listed in the properties aggregation will appear at the top of the selection list. If set to false, then they will appear in their natural alphabetical order within the list.
>
> NOTE: If set to true and the properties aggregation is empty, this setting is ignored.

selectDefinedProperties {Boolean} default: true

> If set to true, then the properties defined in the properties aggregation will be initially selected each time the dialog is opened. If set to false, then they will be unselected.
>
> NOTE: If set to true and the properties aggregation is empty, this setting is ignored.

highlightDefinedProperties {Boolean} default: false

> If set to true, then the properties displayed in the dialog will be highlighted. Properties defined in the properties aggregation will be highlighted in green, those that are not defined will be highlighted in blue.
>
> NOTE: If set to true and the properties aggregation is empty, this setting is ignored.

showPropertyNames {Boolean} default: false

> If set to true, then each property displayed in the selection dialog will show the label and the actual property below. For example, it will display "User ID" and below that user_id.

sortProperties {Boolean} default: true

If set to true properties will be sorted in alphabetical order by their label. If false, they will appear as defined in the objects.

NOTE: If showDefinedPropertiesFirst is set to true and showOnlyDefinedProperties is set to false, then the defined properties will still be displayed at the top of the list (in alphabetical order), followed by the remaining undefined properties (in alphabetical order).

Linx/AS™ Envision PLM

Examples

Basic Example

The most basic example using the selection dialog is to simply set the showSelectionDialog property to true and let the component handle the rest. With this configuration, the user will be presented with a list of all the discovered properties in the bound model's objects. It will attempt to generate a human readable Label based on the property name. The user will have the option to select as many of these properties as they like, as well as the option to download all available data, or just the data that is currently displayed in the associated sap.m.ListBase control (if applicable).

```
<comp:ExcelExportButton listBaseID="list" showSelectionDialog="true"/>
```

Select Values to Export

☐ Select All (19)

☐ Action Type

☐ Actiontaken

☐ Category Code

☐ Date

☐ Days To Process

☐ Description

☐ Id

☐ Investigator Id

☐ Investigator Name

☐ Location Description

Export All Rows (126)   Export Visible Rows (20)   Close

Examples with defined properties

In this example, we will define which properties we want to have exported, providing custom column labels.

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true">
    <comp:properties>
        <comp:ExcelExportProperty label="Date Created"
            value="date" type="Date"/>
        <comp:ExcelExportProperty label="Title" value="title"
            type="String"/>
        <comp:ExcelExportProperty label="Description"
            value="description" type="String"/>
        <comp:ExcelExportProperty label="Lead Investigator"
            value="reportedBy" type="String"/>
    </comp:properties>
</comp:ExcelExportButton>
```

This will result in a selection dialog that displays only the provided properties, will select those properties by default, and will give the user the ability to download all data, or just the data currently loaded in the sap.m.ListBase component:

Select Values to Export

- ☑ Select All (4)
- ☑ Date Created
- ☑ Description
- ☑ Lead Investigator
- ☑ Title

Export All Rows (126)   Export Visible Rows (20)   Close

Now we will modify the dialog to show all available properties, not just the ones we have defined in the properties aggregation by setting showOnlyDefinedProperties to false.

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false">
       <comp:properties>
             <comp:ExcelExportProperty label="Date Created"
                   value="date" type="Date"/>
             <comp:ExcelExportProperty label="Title" value="title"
                   type="String"/>
             <comp:ExcelExportProperty label="Description"
                   value="description" type="String"/>
             <comp:ExcelExportProperty label="Lead Investigator"
                    value="reportedBy" type="String"/>
       </comp:properties>
</comp:ExcelExportButton>
```

This will provide an additional checkbox to allow the user to select/de-select just the defined properties or all properties. It will also by default select only the defined properties when opening:

We will modify this again to put the default properties at the top by setting showDefinedPropertiesFirst to true:

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false" showDefinedPropertiesFirst="true">
      <comp:properties>
            <comp:ExcelExportProperty label="Date Created"
                  value="date" type="Date"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Lead Investigator"
                        value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

Select Values to Export

☐ Select All (19)   ☑ Select Defaults (4)

☑ Date Created

☑ Description

☑ Lead Investigator

☑ Title

☐ Action Type

☐ Actiontaken

☐ Category Code

☐ Days To Process

☐ Id

☐ Investigator Id

⊞ Export All Rows (126)   ⊞ Export Visible Rows (20)   ⊗ Close

Next, let's modify this again so that the default properties are NOT initially selected by setting selectDefinedProperties to false:

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false" showDefinedPropertiesFirst="true"
selectDefinedProperties="false">
      <comp:properties>
            <comp:ExcelExportProperty label="Date Created"
                  value="date" type="Date"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Lead Investigator"
                  value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

Select Values to Export

☐ Select All (19)   ☐ Select Defaults (4)

☐ Date Created

☐ Description

☐ Lead Investigator

☐ Title

☐ Action Type

☐ Actiontaken

☐ Category Code

☐ Days To Process

☐ Id

☐ Investigator Id

▦ Export All Rows (126)   ▦ Export Visible Rows (20)   ⊗ Close

Now let's highlight the properties that are defaults versus the ones that are not by setting highlightDefinedProperties to true:

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false" showDefinedPropertiesFirst="true"
selectDefinedProperties="false" highlightDefinedProperties="true">
      <comp:properties>
            <comp:ExcelExportProperty label="Date Created"
                  value="date" type="Date"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Lead Investigator"
                  value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

Select Values to Export

☐ Select All (19)   ☐ Select Defaults (4)

☐ Date Created

☐ Description

☐ Lead Investigator

☐ Title

☐ Action Type

☐ Actiontaken

☐ Category Code

☐ Days To Process

☐ Id

☐ Investigator Id

⊞ Export All Rows (126)   ⊞ Export Visible Rows (20)   ⊗ Close

Now we will modify the dialog to show the property names, not just the labels by setting showPropertyNames to true:

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false" showDefinedPropertiesFirst="true"
selectDefinedProperties="false" highlightDefinedProperties="true"
showPropertyNames="true">
      <comp:properties>
            <comp:ExcelExportProperty label="Date Created"
                  value="date" type="Date"/>
            <comp:ExcelExportProperty label="Title" value="title"
                  type="String"/>
            <comp:ExcelExportProperty label="Description"
                  value="description" type="String"/>
            <comp:ExcelExportProperty label="Lead Investigator"
                        value="reportedBy" type="String"/>
      </comp:properties>
</comp:ExcelExportButton>
```

Select Values to Export

☐ Select All (19)   ☐ Select Defaults (4)

☐ **Date Created**
date

☐ **Description**
description

☐ **Lead Investigator**
investigatorName

☐ **Title**
title

☐ **Action Type**
actionType

☐ **Actiontaken**
actiontaken

⊞ Export All Rows (126)   ⊞ Export Visible Rows (20)   ⊗ Close

We can turn off property sorting by setting sortProperties to false. Note that in this example because we have showDefinedProperties set to true, they will still appear on top.

```
<comp:ExcelExportButton listBaseId="list" showSelectionDialog="true"
showOnlyDefinedProperties="false" showDefinedPropertiesFirst="true"
selectDefinedProperties="false" highlightDefinedProperties="true"
showPropertyNames="true" sortProperties="false">
    <comp:properties>
        <comp:ExcelExportProperty label="Date Created"
            value="date" type="Date"/>
        <comp:ExcelExportProperty label="Title" value="title"
            type="String"/>
        <comp:ExcelExportProperty label="Description"
            value="description" type="String"/>
        <comp:ExcelExportProperty label="Lead Investigator"
                value="reportedBy" type="String"/>
    </comp:properties>
</comp:ExcelExportButton>
```

Select Values to Export

☐ Select All (19)   ☑ Select Defaults (4)

☑ **Date Created**
date

☑ **Description**
description

☑ **Title**
title

☑ **Lead Investigator**
investigatorName

☐ **Location Name**
locationName

☐ **Near Miss**
nearMiss

▦ Export All Rows (126)   ▦ Export Visible Rows (20)   ⊗ Close

API Documentation

ExcelExportButton

Constructor

Constructor for a new ExcelExportButton.

Accepts an object literal mSettings that defines the initial property values, aggregated and associated objects as well as event handlers. See sap.ui.base.ManagedObject for a general description of the syntax of the settings object.

```
new ExcelExportButton(sId?, mSettings?)
```

Parameter	Type	Default Value
sId?	String	
ID for the new control, generated automatically if no ID is given		
mSettings?	Object	
Initial settings for the new control		

Properties

Name	Type	Default Value
listBaseID	String	
The ID of the sap.m.ListBase from which this control will read model data, items and properties.		
fileName	String	Export.xlsx
The name to use for the generated Excel document.		
sheetName	String	Export Data
The name to use for the sheet in the generated Excel document.		
exportAllRows	Boolean	false
If true all the data found is the bound model will be exported. If false only the data from the items currently displayed in the sap.m.ListBase will be exported. NOTE: If showSelectionDialog is set to true, this property is ignored.		
exportAllColumns	Boolean	false
If true, all non-array based properties discovered in the objects of the bound model objects will be exported. If false, only the properties defined in the properties aggregation will be exported. NOTE: If showSelectionDialog is set to true or the properties aggregation is empty, this property is ignored.		
showSelectionDialog	Boolean	false
If true, the user is presented with a selection dialog where they can select the data to be exported. If false then export is initiated automatically based on the other properties defined.		
showOnlyDefinedProperties	Boolean	true

If true, only properties in the properties aggregation will be available for selection. If false, then all discovered non-array based properties will be available for selection and two selection checkboxes will be provided, one to select all default properties, and another to select all properties. NOTE: If showSelectionDialog is set to false, this property is ignored.		
showDefinedPropertiesFirst	Boolean	false
If true, then properties in the properties aggregation will be displayed in the top of the selection. If false they will be displayed in their normal order. NOTE: If showSelectionDialog is set to false, this property is ignored.		
selectDefinedProperties	Boolean	true
If true, then properties defined in the properties aggregation will automatically be selected when the dialog opens. If false, no properties will be selected. NOTE: If showSelectionDialog is set to false, this property is ignored.		
highlightDefinedProperties	Boolean	false
If true, properties will be highlighted to show which are defaults (green) and which are not (blue). NOTE: If showSelectionDialog is set to false, this property is ignored.		
showPropertyNames	Boolean	false
If true, then property names as well as labels will be displayed. If false, then only labels will be displayed. NOTE: If showSelectionDialog is set to false, this property is ignored.		
sortProperties	Boolean	false
If true, properties are sorted alphabetically. If false, they appear in their discovered order. If showDefinedPropertiesFirst is set to true, then they will be displayed first (in order) then non-defined properties will be displayed (in order). NOTE: If showSelectionDialog is set to false, this property is ignored.		

Aggregations

Name	Cardinality	Type
properties (default)	0..n	ExcelExportProperty
Defines the properties that are to be exported or considered default values.		

www.linxas.net

Constructor

Constructor for a new ExcelExportProperty.

Accepts an object literal mSettings that defines the initial property values, aggregated and associated objects as well as event handlers. See sap.ui.base.ManagedObject for a general description of the syntax of the settings object.

```
new ExcelExportProperty(sId?, mSettings?)
```

Parameter	Type	Default Value
sId?	String	
ID for the new control, generated automatically if no ID is given		
mSettings?	Object	
Initial settings for the new control		

Properties

Name	Type	Default Value
label	String	
The column label to use for this property column in the generated Excel document.		
value	String	
The property to use for the data row for the column in the generated Excel document.		
type	sap.ui.export.EdmType	EdmType.String
The expected data type in the property.		
wrap	Boolean	false
If true, wrap long text values. if false, display test in one line.		