Cello How-To Guide

Master Data Management

techcello

Contents

1 Master Data Management

Most of the systems possess lot of administration screens and building them consumes time. Master data in CelloSaaS facilitates the creation of admin screens with just few lines of code in short span of time. Master data is used in building a data entry page for a table by simply mentioning the configuration details in configuration file. The data entered through this page will be attached to the specific tenant.

1.1 MasterData Configuration

To configure master data, the user has to specify the details in a separate configuration file.

MasterDataConfiguration.Config.

The user needs to specify the configurable settings using appropriate tags and attributes as follows:

```
<masterData>
<addname="Language"entityName="Language"tableName="Language"viewPrivilege="ViewMasterData"crea
tePrivilege="addMasterData"deactivateRecord="true">
<auditInformationname="Language"tenantColumn="TenantId"statusColumn="LanguageStatus"primaryColu
mn="Language_ID">
</auditInformation>
<column>
<addname="Language_ID"fieldId="Language_ID"displayHeader="Language
ID"parentValueColumn=""displayAs="AsItIs"requiredErrorMessage="Required"mandatory="True"></add>
            <addname="Language_Name"fieldId="Language_Name"displayHeader="Language
Name"displayAs="AsItIs"regularExpression="[^a-zA-Z]"mandatory="True"></add>
<addname="Hi"fieldId="Hi"displayHeader="Hi"parentValueColumn=""displayAs="AsItIs"parentIdColumn=""pa
rentTable=""mandatory="false"></add>
</column>
<childRelations>
<addname="Area"childTable="Area"childForeignKeyColumnName="LanguageId"childTableSoftDelete="true"
></add>
</childRelations>
</add>
<addname="Area"entityName="Area"tableName="Area">
<auditInformationname="Area"tenantColumn="TenantId"primaryColumn="AreaId"statusColumn="AreaStatus
"></auditInformation>
<column>
<addname="AreaName"fieldId="AreaName"displayHeader="Area Name"displayAs="AsItIs"></add>
<addname="AreaDescription"fieldId="AreaDescription"displayHeader="Area
Description"displayAs="AsItIs"></add>
<addname="LanguageId"fieldId="LanguageId"displayHeader="Language"displayAs="DDL"parentIdColumn=
"Language_ID"parentTable="Language"parentValueColumn="Language_Name"parentTenantColumn="Tena
ntId"parentStatusColumn="LanguageStatus"></add>
</column>
</add>
<addname="Department"entityName="Department"tableName="Department">
<auditInformationname="Department"tenantColumn="TenantId"primaryColumn="DepartmentId"></auditInfor
mation>
```

```
<column>
<addname="DepartmentName"fieldId="DepartmentName"displayHeader="Department
Name"displayAs="AsItIs"></add>
<addname="DepartmentDesc"fieldId="DepartmentDesc"displayHeader="Department
Desc"displayAs="AsItIs"maxLength="100"></add>
</column>
<childRelations>
<addname="Designation"childTable="Designation"childForeignKeyColumnName="DepartmentId"></add>
</childRelations>
</add>
</masterData>
```

1.2 MasterData Configuration Tags

In order to configure master data, user should know the list of tags used. <masterData> tag – MasterDataElement: This tag is used to configure the masterdata. The attributes of this tag should be used with the <add> tag. The following are the attributes related to this tag:

Attributes	**Description**
Column	Used to get the columns of the entity. Value of this attribute should be MasterDataColumnElement name
EntityName	Specifies the entityname which is used to bring out CelloSaaS features like Data model extension,tracking, etc.
TableName	Specifies the name of the table
ConnectionStringName	Specifies the name of the connection string. If it does not exist, CelloSaaS considers Application connection string
CreatePrivilege	Sets create privilege to add a masterData
UpdatePrivilege	Sets update privilege to update the masterdata
DeletePrivilege	Sets delete privilege to delete the masterdata
ViewPrivilege	Sets view privilege to view the masterdata list and details

The following elements help in configuring the complete masterdata based on the user requirements. All these elements should be configured under <masterData> tag.

1**.<auditInformation> tag – AuditFieldInformation**:This tag is used to audit masterdata entity. The following are the attributes related to this tag.

Attributes	Description
StatusColumn	Attribute is used to identify the audit status of the column. The value for this attribute is any valid string.For example, statusField=”Status”
TenantColumn	Specifies name of the column where tenantId will be stored
primaryColumn	Specifies name of the primary column
UpdatedBy	Specifies the user who had lastly updated the masterdata entity row. The value for this attribute is the username who had lastly updated the row
UpdatedOn	Specifies the date in which the masterdata entity row lastly updated. The value for this attribute is the date string specifying when the row got updated
AddedBy	Specifies the user who added the masterdata entity row. The value for this attribute is the username who had added the row
AddedOn	Specifies the date in which the masterdata entity row got added. The value for this attribute is the date string specifying when the row got added

2.**<column> tag - MasterDataColumnElement:** This tag is used to define the list of columns that should get bound with the gridview. The attributes of this tag should be used with the <add> tag.Following are the attributes related to this tag.

Attributes	Description
Name	Attribute used to uniquely identify a column from other columns. The value for this attriute is any valid string.
FieldId	Specifies the field Identifier which is used to bring out the customization
Display Header	Specifies the column header name. The value for this attribute is any valid string.
Display Text Column	Specifies what text to be displayed to the user for the column. The value for this attribute is any valid string.
Display As	Specifies how the column should get displayed to the user. The value for this attribute can be any one of the following: AsItIs – To display a textbox control DDL – To display a drop down list control CheckBox – To display a checkbox control Radio – To display a Radio button control Date – To display a date control

	PickupList – To display a pickup list which has beed added in pickup list configuration
Parent Table	Used to specify whether the column got any key relation with some other table.The value for this attribute is any valid database table name which got a key relation with this column
Parent value Column	Specifies to which parent table column the field got the key relation.The value for this attribute is the column name which got the key relation with this column
Parent Tenant column	Specifies name of the parent column where tenantId will be stored
Parent Status column	Specifies name of the parent status column
Mandatory	It specifies whether user have to provide input for the column or can be left empty True: specifies that user have to provide input for the column False: specifies the column can be left empty
MultiLine	Specifies whether the text field is multiline text field or singleline text field True: specifies that the column is a multiline text field False: specifies that the column is a singleline text field
MaxLength	Specifies the max length for text box
Required Error Message	Specifies the error message that should be displayed to the user when the required/mandatory column left blank The value for this attribute is any valid string
Read Only	Specifies whether the column can be modified by the user or not True: specifies that the column cannot be modified by the user False: specifies that the column can be modified by the user
AutoPostBack	Specifies whether the column should posts back to the server each time a user interacts with the column or not True: specifies that the column should post back to the server False: specifies the column should not post back to the server

Soft Delete	Specifies whether to introduce a flag corresponding to the column which indicates whether the column should be logically deleted or should be actually deleted without any introduction of a flag True: specifies that a flag should be introduced in order to do a logical delete False: specifies the column should be deleted actually
PickupListId	Specifies Pickuplist id which has been created in pickuplist management

3**.<childRelations> tag:** This tag is used to configure the child relation of master data. Usually used to represent a hierarchical gridview. The attributes of this tag should be used with the <add> tag. The following are the attributes related to this tag:

Attributes	**Description**
Child Table	Specifies the child table that got a key relation with this column. The value for this attribute is any valid database table name which got a key relation with this column
Child Foreign Key Column Name	Specifies the child column name to which this column got a key relation. The value for this attribute is the column name which got the key relation with this column
Child Table Soft Delete	Specifies whether to introduce a flag corresponding to the child column which indicates whether the child column should be logically deleted or should be actually deleted without any introduction of a flag True: specifies that a flag should be introduced in order to do a logical delete False: specifies that the child column should be deleted actually Based on the test result, the data limit for the tables of master data can be in the range of 500-800 records
ChildTenantColumn	Specifies tenant column for the child table
ChildStatusColumn	Specifies status column for the child table

Contact Information

Any problem using this guide (or) using Cello Framework. Please feel free to contact us, we will be happy to assist you in getting started with Cello.

Email: support@techcello.com

Phone: +1(609)503-7163

Skype: techcello