Advanced Grid

Introduction

Advanced Grid is best described as a Grid with rich inline editing transformation.

Depending on the target field, the Advanced Grid element would mimic the original target type for accurate and richer data entry capability.

The Advanced Grid is only available on **Professional** and **Enterprise** Edition.

Get Started

The easiest way to see how the Advanced Grid works is to use the existing built-in App Expenses Claims and create a new form to mimic an existing form in the app. Here are the steps:

1. Start the **Joget Server** and open the **App Center**.

2. Log in as **admin** and click on the pencil icon on the **Expenses Claim** to open the Design App. (see Figure 1)
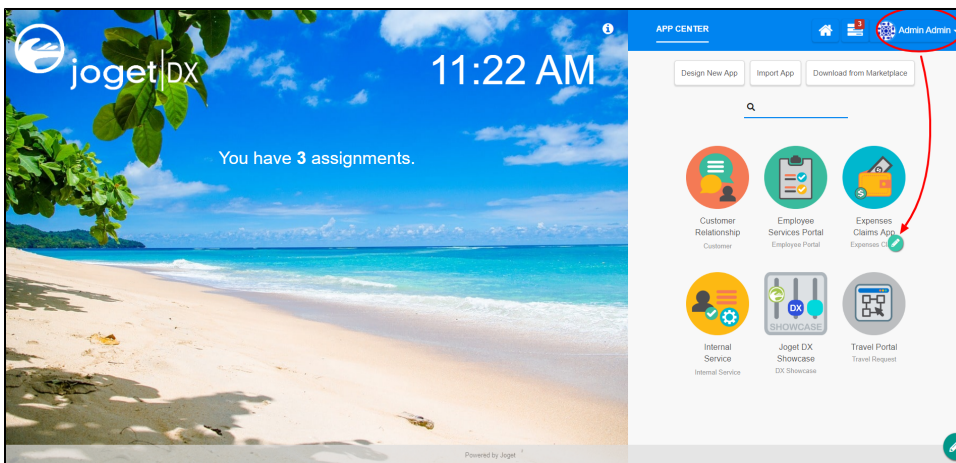

Figure 1

3. Click on "**Create new form**" and fill up the Form Details as follows (see Figure 2):
    i. Form ID: **advGrdTest**
    ii. Form Name: **Advanced Grid Test**
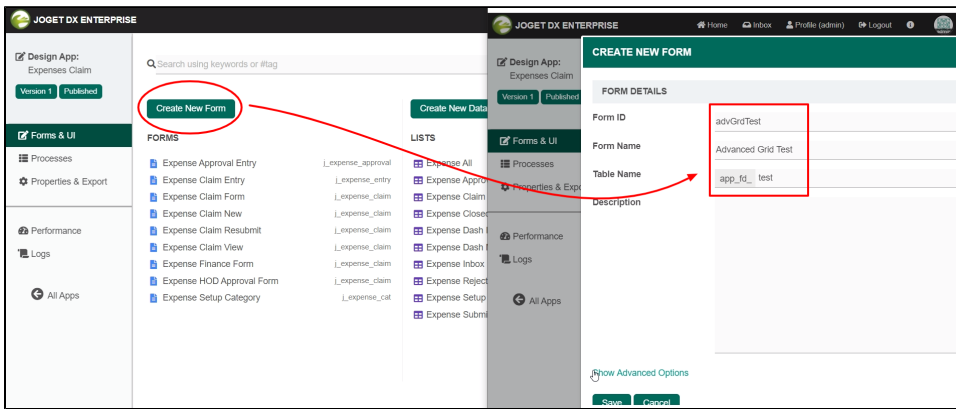   iii. Table Name: **test**

Figure 2

4. Click on the **Save** button and you will be directed to the **Form Builder**.

5. Drag and drop the **Advanced Grid** element on to the canvas.

6. **Hover** the mouse over the Advanced Grid element on the canvas and click on **Edit** to open up the Edit Advanced Grid properties.

7. Fill up the **Edit Advanced Grid** properties as follows and click on **OK** (see Figure 3):
   a. ID : **entries**
   b. Form: **Expense Claim Entry**
   c. Columns :

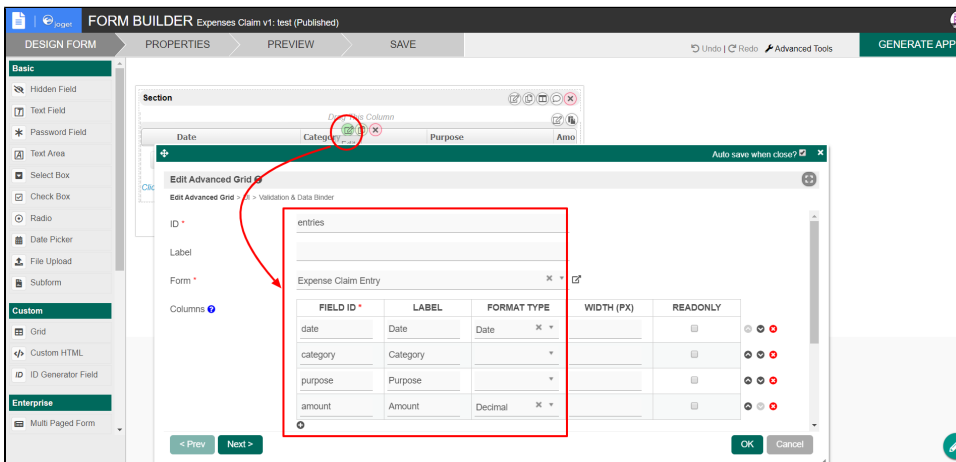| Field ID*	Label	Format Type
date	Date	Date
category	Category	
purpose	Purpose	
amount	Amount	Decimal

Figure 3

8. Click on **Preview** to open up the Preview tab, try clicking on each column in the Advanced Grid and notes each column's behavior. (see Figure 4)

Figure 4

9. Open up Expense Claim Entry and note the Field ID's are the same as written in the Edit Advanced Grid. This is how the Advanced Grid mimics the targeted fields we listed in the step 7c from said target form. (see Figure 5)

Figure 5

Advanced Grid Properties

Edit Advanced Grid

Name	Description
ID	Element ID. (By declaring as "entries", a corresponding database table column "c_entries" will be created) Please see Form Element for more information about defining the ID and list of reserved IDs.
Label	Element Label to be displayed to the end-user.
Form	Target form for record editing.
Columns	The columns defined here must be correspondent to the Form chosen above.

Name	Description
Value	Field ID of the target form. **Example** In "Expense Claim Entry" form, there are Form Elements with the ID "date", "category", "purpose", and "formattedAmount"
Label	Column Label.
Format Type	If you would like to format the returned value, you may make use of the format type. **Available Format Types** • **Date** - formats the field as date. • **Decimal** - formats the field as decimal.
Width	Column width in characters. **Sample** 200px
Readonly	Determines if the column is editable.

UI

Grid

Name	Description
Search	Enable search capability within the grid data itself.
Readonly	Determines if the element is editable.
Disable Add Feature	Determines if a new row can be added.
Disable Delete Feature	Determines if a row can be removed.
Default Sorting Column Number (Start from 1)	Defines the column index to sort by default. This field accepts integers starting from 1 onwards only. Example: If the grid has 5 columns, inputting a "4" will sort the 4th column from the left in ascending order.
Delete confirmation message	Confirmation message when deleting a row.
Show Row Numbering?	Shows numbering on the grid.

Editing

Name	Description
Key to save cell	Keyboard key to use to determine when to save and move on to the next cell.
Edit hint	Edit hint message.

Paging

Name	Description
Record Per Page	Paging Parameter - Record to show per Page
Paging Options	Paging Parameter - Paging Options

Validation & Data Binder

Validation

Validation & Data Binder

Edit Advanced Grid > UI > **Validation & Data Binder**

Validation

Validator	▼
Unique Column ❓	▼
Min Number of Row Validation (Integer)	
Max Number of Row Validation (Integer)	
Error Message	Invalid number of rows

Name	Description
Validator	Attach a Validator plugin to validate the input value. Please see Form Validator. ⓘ **When will validation takes place?** Validation will take place whenever a form is submitted except when it is submitted as "Save as Draft".
Unique Column	Column/Field ID to identify record ID.
Min Number of Row Validation (Integer)	Min Number of Row Validation (Integer)
Max Number of Row Validation (Integer)	Max Number of Row Validation (Integer)
Error Message	Error message to be shown when row requirements set above is not met.

Data Binder

Validation & Data Binder

Edit Advanced Grid > UI > **Validation & Data Binder**

Data Binder

Load Binder

Store Binder

Name	Description
Load & Save Binder	**Load Binder** allows you to customize the method for data retrieval to populate the advance grid in the form.
	Save Binder allows you to customize the method on how the advance grid row records are saved to.
	This option is empty by default. An empty binder means that the advance grid records will be saved/loaded as a JSON format in the parent form & database table. See the list of available Form Binders.
	The recommended binder to use is the **Multirow Form Binder** so that each record is saved into a child database table via a subform definition. The binder will update the foreign key "parent id" into each child record to point to the parent database table..

Related Tutorials

- Calculating Time Between Two Dates using Advanced Grid