

Advanced Form Data Binder

Definition

Advanced Form Row Data Binder is an extended version of the default Form Data Binder. It allows you to add in Filter Conditions in a guided and friendly manner.

New Feature

This feature has been enhanced in Joget Workflow v6 to support join, group and aggregate function.

SOURCE DESIGN PROPERTIES PREVIEW SAVE Undo | Redo Advanced Tools

Select Binder > Select Source of Data (Binder) (Advanced Form Data B

Select Binder ?

Select Source of Data (Binder) Advanced Form Data Binder

Select Source of Data (Binder) (Advanced Form Data Binder)

Configure Advanced Form Row Data Binder ?

Form * Expense Claim Entry

Joins Form Data Table

Form Data Table Name *	Field	Join Field Id *
j_expense_claim	j_expense_claim.id	claim

+ - + -

Figure 1: Advanced Form Data Binder Properties

Name	Description
Form	Source form to retrieve data from.

Joins Form Data Table

New Feature

This is a new feature in Joget Workflow v6.

Name	Description
Form Data Table Name	Target table to join with
Field	Target table field to join with
Join Field Id	Parent field Id to join with

Sample

In the screenshot example in Figure 1, such configurations can be presented with the following SQL.

Sample SQL

SELECT * FROM "Claim Entry" entry JOIN hr_expense_claim claim ON claim.id = entry.claim

Advanced

Filter

Filter Conditions

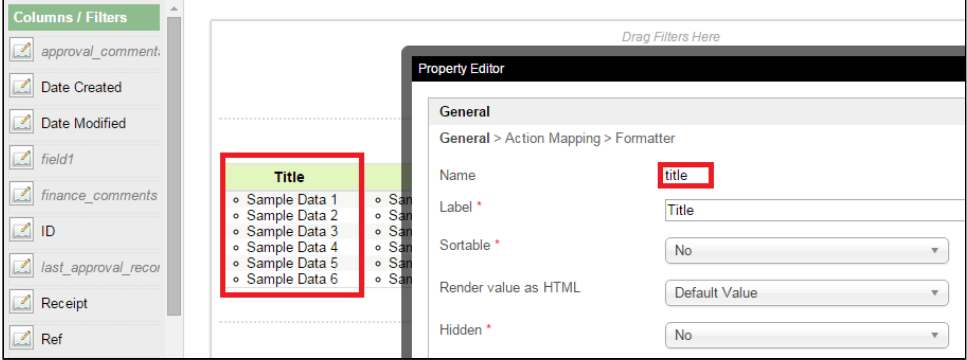
Join Type *	Field *	Operator *	Value	
And	category	Equal	Meal	
Or	category	Equal	Allw	

Figure 2: Advanced Form Data Binder Properties - Advanced - Filter

Name	Description
------	-------------

Filter
Conditio
ns

Filter Conditions

Name	Description
Join Type	<ul style="list-style-type: none">• And• Or
Field	<p>Field ID. (e.g. title)</p> 
Operator	<ul style="list-style-type: none">• Equal• Not Equal• Greater Than• Greater Than Or Equal• Less Than• Less Than Or Equal• Like• Not Like• In• Not In• Is True• Is False• Is Null• Is Not Null
Value	Filter value

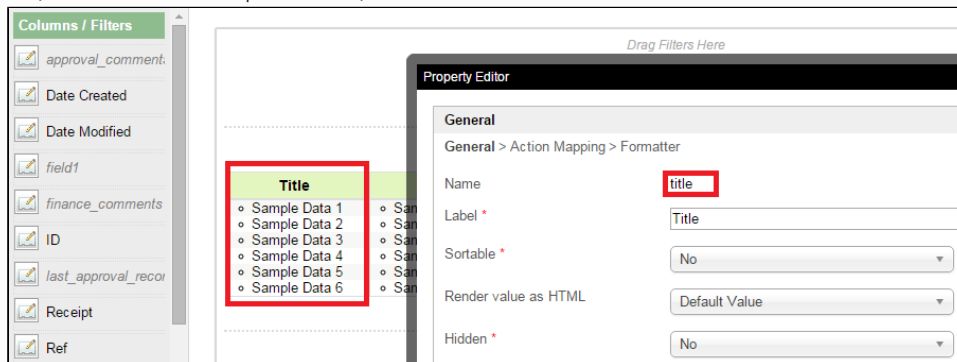
Extra
Conditio
ns

Additional condition(s) for filtering the data set. HQL is the expected here.



Syntax Query

Start your filter name with **e.customProperties.** followed by the field id (i.e. **title**)





HQL is accepted

You may even use operator such as "LIKE" to narrow down your data set.

Select Form
Select Binder > **Select Form**
Form
Extra Filter Condition

localhost:8080/jw/web/cc

localhost:8080/jw/web/console/app/hr_expense/2/datalist/builderPreview/hr_expense_list

10 Title Show

	Title	Ref	Submitted By	Total	Date Created
<input type="checkbox"/>	Team Building 8	REF-000022	Admin Admin	\$ 0.00	19-08-2015 04:29 PM
<input type="checkbox"/>	Team Building	REF-000003	Admin Admin	\$ 0.00	24-08-2015 10:44 AM
<input type="checkbox"/>	Team Building 2	REF-000004	Admin Admin	\$ 0.00	24-08-2015 10:47 AM
<input type="checkbox"/>	Team Building 9	REF-000005	Admin Admin	\$ 0.00	24-08-2015 03:37 PM
<input type="checkbox"/>	Team Building 2	REF-000006	Admin Admin	\$ 0.00	24-08-2015 06:17 PM
<input type="checkbox"/>	Team Building 3	REF-000023	Admin Admin	\$ 0.00	14-09-2015 02:34 PM
<input type="checkbox"/>	Team Building 4	REF-000024	Admin Admin	\$ 0.00	14-09-2015 02:34 PM
<input type="checkbox"/>	Team Building 5	REF-000025	Admin Admin	\$ 0.00	14-09-2015 02:43 PM
<input type="checkbox"/>	Team Building 6	REF-000026	Admin Admin	\$ 0.00	14-09-2015 02:43 PM
<input type="checkbox"/>	Team Building 7	REF-000027	Admin Admin	\$ 0.00	14-09-2015 06:10 PM

Sample

```
e.customProperties.title = 'Trip'
```

Hash variable is accepted here.

Sample

```
e.customProperties.submitted_by = '#currentUser.id#'
```

UserView Key can be used as part of the condition.

Sample

```
e.customProperties.category_id = '#userviewKey#'
```

Aggregate Query

Aggregate Fields

Field *	Function *
amount	Sum
hr_expense_claim.title	Count

Group By

hr_expense_claim receipt
hr_expense_claim remark
hr_expense_claim title
hr_expense_claim field5
hr_expense_claim status
hr_expense_claim username
hr_expense_claim dateCreated
hr_expense_claim dateModified

hr_expense_claim claimant

Having Conditions

Join Type *	Field *	Function *	Operator *	Value
And	hr_expense_claim.total	Sum	Greater Than	1000

Figure 3: Advanced Form Data Binder Properties - Advanced - Aggregate Query



New Feature

This is a new feature in Joget Workflow v6.

Name	Description
Aggregate Fields	<p>Select field is to aggregated.</p> <ul style="list-style-type: none">CountCount DistinctSumMinMaxAvg <p>In the sample screenshot above, the "amount" field will be put into the "Sum" function, and "Count" will be applied to "title".</p>
Group By	<p>Add grouping clause/function to the eventual data set. This can be used together with Aggregate Fields above.</p> <p>In the sample screenshot above, the "amount" field will be summed up by "claimant", shown in per record row.</p>
Having Conditions	<p>The HAVING clause enables you to specify conditions that filter which group results appear in the final results. The WHERE clause places conditions on the selected columns, whereas the HAVING clause places conditions on groups created by the GROUP BY clause. Read more at http://www.dofactory.com/sql/having</p>

The configurations shown in **Figure 3** will produce the following sample result.

joget | **DATALIST BUILDER** HR Expenses Claim App v1: Expense Claim List (Published)

SOURCEDESIGNPROPERTIESPREVIEWSAVE

Columns / Filters

Amount

Amount (SUM)

Calculation Field

Category

claim

Date

Date Created

Date Modified

Drag Filters Here

Drag Columns Here

Drag Row Actions Here

hr_expense_claim.title (COUNT)	Amount (SUM)	hr_expense_claim.username
Sample Data 1	Sample Data 1	Sample Data 1
Sample Data 2	Sample Data 2	Sample Data 2
Sample Data 3	Sample Data 3	Sample Data 3
Sample Data 4	Sample Data 4	Sample Data 4
Sample Data 5	Sample Data 5	Sample Data 5
Sample Data 6	Sample Data 6	Sample Data 6

10

Show

	hr_expense_claim.title (COUNT)	Amount (SUM)	hr_expense_claim.username
<input type="checkbox"/>	2	3000	admin
<input type="checkbox"/>	1	4000	cat

2 items found, displaying all items.

1

CSV

Excel

XML

PDF

Figure 4: Sample result

Expression Columns

Expression Columns
Please refer to [HQL Expression](#)

Alias *	Expression *
price	CAST(price AS long)

+
↑
↓
✖

Custom Checkbox/Radio Button Value
Default to Id

Figure 5: Advanced Form Data Binder Properties - Advanced - Expression Columns

New Feature

This is a new feature in Joget Workflow v6.

Name	Description												
Expression Columns	<p>Additional column can be added in this expression columns. This is especially useful when you need to perform additional computation on multiple columns.</p> <div> <div>Sample</div> <div>CAST(price AS long)</div> </div> <div> <div>Sample</div> <div>CAST(e.customProperties.sales_price AS long) - CAST(e.customProperties.price AS long)</div> </div>												
Custom Checkbox /Radio Button Value	<p>Define custom record ID to be used to pass over to column action. Defaulted to ID.</p> <div> <div>All Expenses History</div> <div> <div>Delete</div> <table> <thead> <tr> <th><input type="checkbox"/></th> <th>#</th> <th>TITLE</th> <th>\$</th> <th>STATUS</th> <th>SUBMIT BY</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>0001</td> <td>Claim A</td> <td>111.00</td> <td>Approved</td> <td>Admin Admin</td> </tr> </tbody> </table> </div> </div>	<input type="checkbox"/>	#	TITLE	\$	STATUS	SUBMIT BY	<input checked="" type="checkbox"/>	0001	Claim A	111.00	Approved	Admin Admin
<input type="checkbox"/>	#	TITLE	\$	STATUS	SUBMIT BY								
<input checked="" type="checkbox"/>	0001	Claim A	111.00	Approved	Admin Admin								

All Expenses History

Delete

<input type="checkbox"/>	#	TITLE	\$	STATUS	SUBMIT BY
<input checked="" type="checkbox"/>	0001	Claim A	111.00	Approved	Admin Admin