Accessing field values and properties

ConfiForms has a lot of various field types to help you with building your data forms. And it has a very sophisticated system to retrieve the values stored, including accessing rich objects such as Page, User and other referenced Records when used together with "smart fields"

> ⚠ Accessing values using **Velocity syntax is deprecated since ConfiForms 1.16** and not supported in 100% of cases. Please try to accomplish what you have to do using [entry.] notation

- Accessing ConfiForms fields
- Additional values available in the context

> ⚠ If you are **upgrading from 1.30** and you have configurations using "virtual functions" which collect the data from multi-select fields then please be aware, that the standard behaviour has chnaged to return ID's, and NOT labels of the fields
>
> If you need to have labels then a **transform operation is needed.** See "transform" function in Virtual functions for details
>
> But, basically, you will need to call on your multiselect field something like this (for example, and assuming your multiselect field is called "mymultiselectfield"):
>
> ```
> mymultiselectfield.transform(label).asArrayMultiSelect
> ```
>
> Before 1.30
>
> ```
> mymultiselectfield.asArrayMultiSelect
> ```
>
> was returning field labels, now (without a transformation) **it returns ID's**

> ⓘ Accessing rich properties is also supported in ConfiForms Field macros. Giving myfield.subproperty as a field name is totally valid and supported. For smart fields this can be of any level deep, but please keep in mind the performance issues this might cause (as values are lazy loaded for each sub-field for "smart" fields).

> ⚠ Make sure the [entry.myfield] is actually stored the way you see it, as sometimes Confluence editor injects some "invisible to user" blocks of HTML, such as spans... and the text becomes actually something like [entry.myfield] which makes it not to work correctly with ConfiForms parser.
>
> The easiest way to see that text is clean is by looking at storage format of the page (using Tools menu and then clicking on "View Storage format" or by installing a free add-on from Atlassian which enables you to see raw storage format directly in the Confluence editor - https://marketplace.atlassian.com/plugins/com.atlassian.confluence.plugins.editor.confluence-source-editor/server/overview)
>
> Please also see this: https://wiki.vertuna.com/display/CONFIFORMS/ConfiForms+FAQ#ConfiFormsFAQ-WhyintheeditorIgetautolinksinsteadof [entry.field_name]notations

> ⓘ See what fields are available on **ConfiForms CLOUD** - Differences between ConfiForms server/data center and ConfiForms cloud

> ⚠ **CLOUD users**
>
> **Please see the fields list that is available on ConfiForms cloud if you are looking for the documentation on accessing field properties in ConfiForms cloud**
>
> Differences between ConfiForms server/data center and ConfiForms cloud

Accessing ConfiForms fields

Here is the table showing you how to reference field values for different field types. Our field name is: "myfield"

Field type	Accessing in ConfiForms Field macro	Accessing though [entry.] notation	Using in filters	Sorting	Remarks
Text	myfield	[entry.myfield]	myfield:*test myfield:test* myfield:*test* myfield:*	as text	
Action Button	myfield	N/A	N/A	N/A	N/A
Autolink	myfield	[entry.myfield]	same as text field	as text	only dynamic part is taken when shown as [entry.] notation
Autonumber	myfield	[entry.myfield]	same as text field	as text	autonumber with previx
Autopage	myfield	See page type	See page type	See page type	See page type
Calculated	myfield	[entry.myfield]	Same as numeric field	as number	non-numeric values are set as 0 in sorting
Captcha	myfield	N/A	N/A	N/A	N/A
Checkbox	myfield myfield.label	[entry.myfield] [entry.myfield.label]	myfield:true myfield:false		Returns true or false By using .label you can access checkbox field label you set
Checkbox group	myfield myfield.label	[entry.myfield] [entry.myfield.label]	myfield:*fieldvalue* to ensure it is matched	as text	
Comment **SINCE 2.13.9**	mycommentfield.user mycommentfield.timestamp mycommentfield.content ⚠ Important! A multi-value field Which means to access any property you will need to transform it first mycommentfield.transform(user).asList - will give you a list of usernames for users who added a comment mycommentfield.transform(content).asList - will get you a comma separated list of comments				
Currency	myfield	[entry.myfield]	Same as numeric field	as number	
Date	myfield myfield.timestamp	[entry.myfield] [entry.myfield.timestamp]	Same as numeric field	as number	timestamp value Date and Datetime field types allow the following metaparams to be given as input [now], [tomorrow], [yesterday], [today], [today]-days(n), [datetimeyyyyMMdd HH:mm], [dateyyyyMMdd] Basically, same as in ConfiForms Filters
Datetime	myfield myfield.timestamp	[entry.myfield] [entry.myfield.timestamp]	Same as numeric field	as number	timestamp value See above

Datetime interval	myfield myfield.startDateFormatted myfield.endDateFormatted myfield.startDate myfield.endDate myfield.startDate.timestamp myfield.endDate.timestamp myfield.asPeriodFormatted myfield.interval - returns interval in minutes between start and end dates **SINCE 2.28.0** **CLOUD** myfield.periodInSeconds myfield.periodInMinutes myfield.periodInHours myfield.periodInDays myfield.periodInWorkDays	[entry.myfield] [entry.myfield.startDateFormatted] [entry.myfield.endDateFormatted] [entry.myfield.startDate] [entry.myfield.endDate] [entry.myfield.startDate.timestamp] [entry.myfield.endDate.timestamp] [entry.myfield.asPeriodFormatted] [entry.myfield.interval] **SINCE 2.28.0** **CLOUD** [entry.myfield.periodInSeconds] [entry.myfield.periodInMinutes] [entry.myfield.periodInHours] [entry.myfield.periodInDays] [entry.myfield.periodInWorkDays]	myfield.startDate: myfield.endDate: conditions as in numeric field	as number by start date	timestamp value
DB dropdown	myfield myfield.id myfield.label	see remarks ->			Same as normal dropdown
DB multi-select	myfield	see remarks ->			Same as normal multi-select
DB radio group	myfield	see remarks ->			Same as normal radio group
DB checkbox	myfield	see remarks ->			Same as normal checkbox
Drawing canvas	myfield	N/A	N/A	N/A	internal format is returned
Dropdown	myfield myfield.id myfield.label	[entry.myfield] [entry.myfield.id] [entry.myfield.label]	myfield:ID myfield.label:LABEL	as text	Each dropdown choice has ID and LABEL [entry.myfield] returns ID
Dynamic Dropdown	myfield myfield.id myfield.label	[entry.myfield] [entry.myfield.id] [entry.myfield.label]	myfield:*fieldvalue* to ensure it is matched	as text	Each dropdown choice has ID and LABEL [entry.myfield] returns ID
Dynamic Multilabel	myfield myfield.id myfield.label	[entry.myfield] [entry.myfield.id] [entry.myfield.label]	myfield:*fieldvalue* to ensure it is matched	as text	Each choice has ID and LABEL [entry.myfield] returns ID
Enroll button	myfield	N/A	N/A	N/A	
File	myfield	[entry.myfield] - will show as link	same as text, see remarks	as text	only attachment/file name is in search
Formula	myfield	[entry.myfield]	same as numeric field	as number	non-numeric values are set as 0 in sorting
Html	myfield	[entry.myfield]	same as text type	as text	
Insight	myfield myfield.attributes. objectTypeAttribute (objectTypeAttributeId=46). objectAttributeValues.value	[entry.myfield] [entry.myfield.attributes. objectTypeAttribute (objectTypeAttributeId=46). objectAttributeValues.value] See more details here: Accessing JSON structures. The tutorial is for ConfiDoc plugin but is perfectly validfor ConfiForms (easpecially from teh perspective of accessing the Insight object attributes)			
Link	myfield	[entry.myfield]	same as text type	as text	
Hidden	myfield	[entry.myfield]	same as text type	as text	
Masked Text	myfield	[entry.myfield]	same as text type	as text	
Multi Select	myfield	[entry.myfield]	same as dropdown (ID and LABEL properties)	as text	
Numeric field	myfield	[entry.myfield]	myfield:>= myfield:> myfield:<= myfield:< myfield:	as number	

JIRA issue	myfield myfield.key myfield.title myfield.summary (same as title) myfield.description myfield.link myfield.status **V 1.35.5** Any JIRA field which is provided via JIRA REST API For example: myfield.fields.assignee.emailAddress myfield.fields.reporter.displayName See also: Using Field Definition Rule to request data from JIRA and set it to ConfiForms fields **3.10.0** **CLOUD** myfield.exists - returns true or false to verify if the value stored in this field corresponds to Jira issue	[entry.myfield] [entry.myfield.key] [entry.myfield.title] [entry.myfield.summary] (same as title) [entry.myfield.description] [entry.myfield.link] [entry.myfield.status]	same as text type and only by key	as text	
JQL	myfeld myfeld.jql myfeld.count	[entry.myfield] [entry.myfield.jql] [entry.myfield.count]	Renders as link to JIRA with number of matching issues 2 additional properties available: jql and count		
Page/Blog post	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	same as text, for properties list see remarks	as text, by default pageId	all properties of AbstractPage class (see Confluence API docs). For cloud see https://developer.atlassian.com/cloud/confluence/rest/api-group-content/#api-wiki-rest-api-content-get Examples: **method in class** / **How to access** getTitle() / myfield.title getSpaceKey() / myfield.spaceKey getCreatorName() / myfield.creatorName getVersion() / myfield.version getCreationDate() / myfield.creationDate ... any other **get** method from Page class /
Password	myfield	[entry.myfield]	same as text type	as text	
Radio group	myfield myfield.id myfield.label	[entry.myfield] [entry.myfield.id] [entry.myfield.label]	myfield:ID myfield.label:LABEL	as text	Each choice has ID and LABEL [entry.myfield] returns ID
Readonly	myfield	[entry.myfield]	same as text type	as text	
Section	myfield	N/A	N/A	N/A	
Simple Date	myfield	[entry.myfield]	Same as numeric field	as number	timestamp value

Smart Classifier	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	Depends on a field referenced. If [entry.myfield] or just "myfield" is referenced then in most cases it acts as text. But with "smart fields" you reference a record in another form and that means through this reference you can get ANY field of that record, including other smart fields. That said, you can build quite a property graph to get any value, which is lazy loaded for you	depends on a field referenced	Properties you can access totally depend on a structure of the record in the form you reference. You can access ANY field value through this notation, including other smart fields which will be lazy loaded for you. Example: ``` [entry.myfield. field_in_another _form] [entry.myfield. another_smartfie ld_in_another_fo rm.somefield] ```
Smart Dropdown	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	See Smart Classifier	depends on a field referenced	See Smart Classifier
Smart Multiselect	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	See Smart Classifier	depends on a field referenced	See Smart Classifier
Smart Checkbox	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	See Smart Classifier	depends on a field referenced	See Smart Classifier
Smart Radio buttons	myfield myfield.* - see remarks	[entry.myfield] [entry.myfield.*] - see remarks	See Smart Classifier	depends on a field referenced	See Smart Classifier
Status	myfield myfield.id myfield.label	[entry.myfield] [entry.myfield.id] [entry.myfield.label]	Same as dropdown field, with IDs and LABELs. But has predefined values for IDs (depending on a colour)	as text	See remarks for dropdown field
Textarea	myfield	[entry.myfield]	same as text type	as text	
User	myfield myfield.username myfield.fullName myfield.picture myfield.email myfield.lastName myfield.firstName myfield.key **FROM 2.12.5** myfield.accountId **CLOUD**	[entry.myfield] [entry.myfield.username] [entry.myfield.fullName] [entry.myfield.picture] [entry.myfield.email] [entry.myfield.lastName] [entry.myfield.firstName]	same as text type, but you can query for field properties: myfield: myfield.username: myfield.fullName: myfield.picture: myfield.email: myfield.lastName: myfield.firstName:	as text, see remarks	sorting is by username field lastName and firstName are synthetic properties, which we try to extract from a given fullName. We consider las word as lastName and what is given before is taken as firstName. This solution does not work for the whole world, but this is the best option we can offer you at the moment
User (multiselect)	myfield	[entry.myfield] [entry.myfield.asUserFullNames] [entry.myfield.asUserEmails] * - see remarks	Same as User	as text	You can show user full names in a list, or emails or (when no helper function is used) as list of usernames See Virtual functions
Voting control	myfield	N/A	N/A	N/A	N/A

Webservice backed fields	As with "JIRA issue" fields you can access ANY property returned by the web service for the option you map with the field.
	
	For example:
	
	``
	[
	    {
	        id: 1,
	        name:
	"Alex",
	        sn: "E123"
	    },
	    {
	        id: 2,
	        name:
	"Vlad",
	        sn: "C100"
	    }
	]
	``
	
	You can map the structures of an array to be used as options for the web services backed fields in ConfiForms
	
	Assuming your field is called "myfield"
	
	then to access the properties you will need to put the following expressions
	
	``
	myfield.id
	myfield.name
	myfield.sn
	``
	
	There is also a synthetic property called "label" for the field what you use as "label" in your web services backed field
	
	Important note: when your web service returns a structure with ID and you map another field to be used as ID then original data will not be accessible, as mapping overrides it.				

WYSIWYG simple editor	myfield	[entry.myfield]	same as text type	as text	

Accessing through *[entry.]* notation can be used in ListView, in PlainView, in CalendarView and in IFTTT macro body, as well as when referencing ConfiForms record fields in macro parameters

See also Virtual functions to help you with formatting values in the format you want

> ⊘ Accessing values using **Velocity syntax is deprecated** and not supported in 100% of cases. Please try to accomplish what you have to do using [entry.] notation

Additional values available in the context

Variable			
[owner]	Deprecated, use [entry._user]	current record owner (ownedBy field) or Anonymous (when user is "Anonymous")	
[modifier] [user]	Deprecated, use [entry._user]	current record modifier (User's full name)	
[now]		Current timestamp as expression. **NB! Not evaluated to the value when in pre-sets and stays as [now] which results ALWAYS in current date. Use [entry._now] for the value** Only on server/dc version	
[today]	Deprecated	Current timestamp as expression. Does not get evaluated to actual timestamp, but shows "current time" when set on the field Use [entry._today] instead or [entry._now]	
[user.name]	Deprecated, use [entry._user.username]	current user's username	
[user.fullName]	Deprecated, use [entry._user.fullName]	current user's full name	
[user.email]	Deprecated, use [entry._user.email]	current user's email	
[entry._user]	_user	current user object (can access email, name, and fullname, see accessing properties for user fields)	User field
[entry._now]	_now	current timestamp, which can be converted to date using Virtual functions or used for comparison against other date /datetime fields On **cloud** this returns timestamp in user's TZ	Datetime
[entry._page]	_page	current page id (present in most cases, but not always)	Page
[entry._storagePage]	_storagePage	form's page id	Page
[entry._today]	_today	current timestamp (for date, without time) On **cloud** this returns value for today in user's TZ	Date
[entry._count]	_count	number of records in the dataset	Numeric
[entry._total]	_total	total number of records in the dataset	Numeric
[entry._formName]	_formName	Name of the form is now available in the context **SINCE 2.2.0**	String

When used with ConfiForms TableViewMerger macro you have access to the following variables

		Type
_sourcePage	Name of the page the form is located	Page
_sourceForm	Name of the form where this data row is taken from	Text

When ConfiForms IFTTT macro is processed it does temporarily make these variable accessible

Available in **SINCE 2.21.0** and **CLOUD**

		Type
_iftttBody	IFTTT macro body	Text
_iftttBodyPrepared	Processed/Prepared IFTTT macro body	Text

See also

[Configuring ConfiForms IFTTT actions and rules](#)

Important note about IFTTT macros with macro bodies which are evaluated as velocity templates - there is additional context variables available

> ⓘ   Standard set of objects inside the context (Velocity templates)
>
> ```
> context.put("entry", entry); <- ConfiForms Entry (raw)
> context.put("user", user); <- Confluence user object
> context.put("page", contentObject); <- AbstractPage object
>
>
> From ConfiForms version 1.49.3 there are additional objects in the context:
>
>
> context.put("generalUtil", newcom.atlassian.confluence.util.GeneralUtil());
>
> context.put("res", ServletActionContext.getResponse());
> context.put("req", ServletActionContext.getRequest());
> context.put("action", com.atlassian.confluence.renderer.radeox.macros.MacroUtils.
> getConfluenceActionSupport());
> ```

which you can referece as any other variable in Velocity using velocity syntax, for example: ${page.id} to reference page id of current page