ConfiForms Field Definition Rules

Looking for CLOUD documentation? This page has information for both versions

Action types available in ConfiForms Field Definition Rules (aka **ConfiForms Rules for Field Definition**)

Rules are the best way to add dynamics to your forms, make validation rules conditional or customise form UI behaviour

Available rules

- Apply CSS on a field
- Apply Filter on a field
- Apply Filter based on matching criteria
- Hide field
- Hide container
- Lookup and set DB value
- Lookup and set WS value
- Lookup and set JIRA issue data
- Lookup and set Insight object data
- Lookup and set value
- Run custom JavaScript
- Set expression
- Set field readonly
- Set form title
- Set field label
- Set value
- Set value if empty
- Set submit button label
- Show field
- Show container
- Reset value
- Validate if exists in other Form
- Validate uniqueness
- Validate on delete
- Validation rule
- Validation rule (dataset)

Availability	Action type	
SERVER/DC CLOUD	Apply CSS on a field	Applies CSS styles on a given field, based on the condition set
SERVER/DC CLOUD	Apply Filter on a field	You can apply specific filters on "select"-type fields. This is especially powerful, when your fields are backed by smart fields, web-service or database fields. Your filter is applied ON THE DATASET this field relies on. Which means, the filter is applied on the source dataset Few examples: • Using ConfiForms Field Definition Rules macro to reduce number of choices in a dropdown field based on the value selected in another field • Using Field Definition Rule to filter choices in another field • Using apply filter on field rule • Working with Dependent DB dropdowns
SERVER/DC CLOUD	Apply Filter based on matching criteria	Please see the following tutorial for details Using filter based on criteria rule

SERVER/DC CLOUD	Hide field	Conditionally hides the field and the "wrapping" container (if given) • How to show and hide fields in the form conditionally
SERVER/DC CLOUD	Hide container	Allows you to hide by CSS selector • How to show or hide blocks of fields in ConfiForms conditionally
SERVER/DC	Lookup and set DB value	Helps you to have a lookup against the query you create from your database • Using Lookup and set a DB value action
SERVER/DC CLOUD	Lookup and set WS value	Allows you to make a request to a webservice and set the received values back to the form's fields
SERVER/DC CLOUD	Lookup and set JIRA issue data	Allows you to request issue details from your Jira (by issue key) and set the received values back to the • Lookup the values from JIRA ticket with Definition Rules
SERVER/DC	Lookup and set Insight object data	Same as with Jira issue data, but allows you to query Insight data storage
SERVER/DC CLOUD	Lookup and set value	This rule allows you to lookup the data from another ConfiForms form and set received data back to the form's fields • Form to lookup and preset values from existing records • form to lookup and preset values from existing records with radio buttons

SERVER/DC	Run custom JavaScript	ConfiForms can execute a custom JavaScript function via ConfiForms Field Definition Rules, and your function can expect the following variables to be present
		▪ formName - is the "form" with HTML field elements. This allows you to access fields very easily via AJS.$(formName).find('#i_**FIELD_NAME**')
		▪ formId - is the "div" element in HTML holding the whole form and fields. This allows you to access fields very easily via AJS.$('#' + formId).find('#i_**FIELD_NAME**')
		which means your function can have a signature like:
		`function myfunction(formName, formId) {`
		`}`
		See demo: Using ConfiForms Field Definition rules with JavaScript action to get and set field values
		Since version 2.21.5 ConfiForms offers some helper functions to use

JavaScript functions

Available since version	Function	Description	Demo
2.21.5	function **cfAddFieldRequired**(formId, fieldName)	Adds required field mark / asterisk. This does not make the field required - you need to have an additional validation in place to validate the field's value	Your browser does not support the HTML5 video element
2.21.5	function **cfRemoveFieldRequired**(formId, fieldName)	Removes required field mark / asterisk from a field. Important to know that if the field is set as required in ConfiForms Field Definition then it still remains as required and users must provide a value before they can submit the form. This function only changes the visual appearance of a field label	
2.21.5	function **cfAddTabStop**(formId, fieldName)	Adds "tab stop", meaning that the field will be excluded from form's set of fields when moving around using the "tab" key	Your browser does not support the HTML5 video element
2.21.5	function **cfRemoveTabStop**(formId, fieldName)	Removes the "tab stop" marker from a field	
2.23	function **cfSetValue**(formId, fieldName, fieldValue, isAdvanced = false)	Set's value on a field. Set "*isAdvanced* = true" if you want to set the value on "advanced" fields (such as dropdown fields with lookup, etc)	
3.8	function **cfSetValueWithEventPropagation**(formId, fieldName, fieldValue, eventType, isAdvanced = false)	Same as "**cfSetValue**" function but triggers an *eventType* event eventType can be of the following type: • change • input • blur • keyup This depends on the field you want to trigger this for. Text fields expect 'input', dropdowns and other choice based fields - "change"; Jira, user and page lookup fields expect a "blur"	
2.23	function **cfResetValue**(formId, fieldName, isAdvanced = false)	Resets value on a field (sets it to *empty* state).	
2.23	function **cfSetValueByIndex**(formId, fieldName, index, isAdvanced = false)	Selects value by index in a dropdown (radio) field. Options start with 0 index (so, the first option will be with index = 0)	
3.8	function **cfToggleReadOnly**(formId, fieldName, isReadOnly, isAdvanced = false)	Toggles readonly attribute (either adds it or removes it, depending on a *isReadOnly* parameter value)	
3.8	function **cfToggleDisabled**(formId, fieldName, isDisabled, isAdvanced = false)	Toggles disabled attribute (either adds it or removes it, depending on a *isDisabled* parameter value)	

More on JavaScript functions

SERVER/DC CLOUD	Set expression	Same as set value, but you can set expression (Supported math operators, formulas and functions), which will be calculated for you Useful to wrap expressions in () to help ConfiForms with parsing
SERVER/DC CLOUD	Set field readonly	Sets field as readonly, setting reverse condition will do the opposite. Please note that you will need 2 rules to set/unset the readonly status - you cannot have one (as with hide/show, for example) Please also note that not all the fields can be set as readonly - dropdowns are ALREADY readonly, and setting the as readonly makes no sense and no impact. Also, not all the fields are supported (work in progress).
SERVER/DC CLOUD	Set form title	Rule to help you to changes form's title
SERVER/DC CLOUD	Set field label	Sets label text for the field
SERVER/DC CLOUD	Set value	Set's value to the fields. Please be aware, that internal value for the field is expected. See ConfiForms Field Types Set the value on dropdown choice change
SERVER/DC CLOUD	Set value if empty	Same as above, but has additional check applied to ensure the value is set ONLY when the field has not value set
SERVER/DC CLOUD	Set submit button label	Changes label on submit form button
SERVER/DC CLOUD	Show field	Conditionally shows the field and the "wrapping" container (if given) • How to show and hide fields in the form conditionally
SERVER/DC CLOUD	Show container	The opposite what "Hide container" does, and allows you to show elements matching the CSS selector you put • How to show or hide blocks of fields in ConfiForms conditionally
SERVER/DC CLOUD	Reset value	Resets the field value
SERVER/DC CLOUD	Validate if exists in other Form	Useful action type to check if the record matching configured condition / filter exists in the other form • Using Field Definition Rule to validate the data across different forms
SERVER/DC CLOUD	Validate uniqueness	On form submit, the action checks if the record is unique within a dataset. Uniqueness criteria is calculated based on the fields given
SERVER/DC CLOUD	Validate on delete	This action is executed on "record delete", and could prevent the record being deleted based on the condition you set
SERVER/DC CLOUD	Validation rule	Validates record(conditionally) • Conditional validation for required field • Conditional validation rules and date fields

SERVER/DC CLOUD	Validation rule (dataset)	Validates current record against the rows in the dataset, For example, when you want to ensure the max number of records stored in the dataset • Creating conditional validation rules against the dataset