Source URL: [Query list action configurations in NITRO Custom Actions - Crow Canyon Software Support](https://www.crowcanyon.help/article/470/)

Query list action configurations in NITRO Custom Actions

4594 views  4 min , 24 sec read [1](https://www.crowcanyon.help/article/470/)

* **Applies to:** SharePoint Online and On-Premises

Introduction

* Query List action is used to retrieve one or more items of a list as per the defined criteria. For example, get all open Tasks related to a Ticket or get all line items for a Purchase Request. Retrieved list items can then be used in other actions, some examples of this usage are given below:
* Cancel all open Tasks of the Ticket when Ticket is cancelled
* Calculate the sum of cost of all line items
* Create an Asset for each of the line items of a Purchase Request
* Check if all Tasks related to a Request are ‘Closed’ and if so then mark the Request as Completed
* Custom Actions are used extensively in NITRO based applications and Query List action is indispensable whenever we have to deal with items that are related to each other. Main input for this action is the site and list from which items have to be fetched and the filter criteria for the items. This action is cross-site capable and can fetch items from any site in the site-collection as long as the user executing the action has the required permissions. Below sections explain the various configuration options in more detail.

Detailed Description

* Graphical user interface, text, application, email

  AI-generated content may be incorrect.
* Please note that at most 50 items are fetched for the Query list action due to performance considerations. For end-user based scenarios this should cover most of the cases.

Run for

* Query List action runs in the context of a item and subsequent options depend on this selection.
* *Current Item***:** It is the item on which user has invoked the custom action from the SharePoint UI. For example, while working with a Purchase Request, we will select this option to get all line items associated with the Purchase Request
* *Lookup***:** It is the item selected in the lookup column of the item on which user has invoked the custom action from the SharePoint UI. For example, Purchase Request has a lookup column for a Purchase Order item. While invoking custom action on Purchase Request, we can select this lookup column to get items associated with the Purchase Order. That way, we can do operations based on items associated with a lookup list item of the current item

Type

* Whether to get a single item or a collection of items. This depends on the business use case.

Query Site

* Select the site of the list from which items have to be fetched. Items can be fetched from any site within the site collection. Based on site selection, next drop-down for the lists will be populated

Query List

* Select list from which items have to be fetched. Please note that next option depends on this selection.

Get Item(s) From

Lookup column

* This option will show a drop-down to select the lookup columns from the list of the item selected in ‘Run for’ option. Note that these lookup columns should be for the list selected in the ‘Query List’ option above. A lookup column contains (or references) items from another list (lookup list specified in the lookup column settings). This option will read these items from the lookup column of the item selected in the ‘Run for’ option:
* Graphical user interface, text, application, email

  AI-generated content may be incorrect.

Related Items

* For this scenario, list selected in the ‘Query List’ should have a lookup to the list of the item selected in the ‘Run for’ option. For example, Tasks list having a lookup to Request list and action being configured on Request list. This option will fetch all Tasks that are related to Request filtered by the selected view (like only open/closed Tasks as per the view conditions).
* Graphical user interface, text, application, email

  AI-generated content may be incorrect.

List View

* Fetch items from the list selected in the ‘Query List’ option based on the selected ‘List View’. In this case it is not necessary that there are any lookup column relations between list of the item selected in the ‘Run for’ option and the list selected in the ‘Query List’ option. Items will be fetched based on the conditions specified in the View conditions
* Graphical user interface, text, application, email

  AI-generated content may be incorrect.

Query Builder

* Fetch items from the list selected in the ‘Query List’ option based on the conditions specified in the Query Builder. In this case it is not necessary that there are any lookup column relations between list of the item selected in the ‘Run for’ option and the list selected in the ‘Query List’ option.
* Graphical user interface, text, application, email

  AI-generated content may be incorrect.
* Column placeholders from the item selected in the ‘Run for’ option can be specified in the conditions.
* Placeholder Syntax:  %%[Column Display Name|Column Internal Name]%%
* E.g. %%[ID|ID])%%

Query

* Fetch items from the list selected in the ‘Query List’ option based on the specified CAML query. In this case it is not necessary that there are any lookup column relations between list of the item selected in the ‘Run for’ option and the list selected in the ‘Query List’ option. Please refer CAML query syntax from [this article](https://docs.microsoft.com/en-us/sharepoint/dev/schema/query-schema).
* Graphical user interface, text, application, email

  AI-generated content may be incorrect.
* Column placeholders from the item selected in the ‘Run for’ option can be specified in the query.
* Placeholder Syntax:  %%[Column Display Name|Column Internal Name]%%
* E.g. %%[ID|ID])%%
* Sample CAML Query:
* <View>  
  <Query>  
  <Where>  
  <Eq>  
  <FieldRef Name='RelatedItems' LookupId='TRUE'/>  
  <Value Type='Lookup'>%%[ID|ID])%%</Value>  
  </Eq>  
  </Where>  
  </Query>  
  </View>

Allow user to select items

* If this is checked, then user will be shown the items fetched from the Query List action. User can select one or more items from the grid and selected items will constitute the result of the Query List action.  This way, user can refine the result of the Query List action.
* If this option is not checked then all items returned by the Query list action will be included in result of the action.

Continue Custom Action Execution

* This option can be used to continue or stop the execution of the Custom Action based on whether items are fetched in the Query List action or not.
* **Always**: This is the default option and execution of subsequent actions is not dependent on whether any items were fetched in the Query List action or not
* **Only if item(s) found**: This option will continue the custom action execution flow only when one or more items are fetched in the Query list action. This is used in common scenario where we are fetching items using Query List action and then there is an Update Item action to update all the items fetched. If there are no items found in Query List then it is not required to run the action to update them. However, note that if there are any other actions after Update Item action then those too will be skipped. So use this option if all subsequent actions should be skipped if no items are found in Query List action
* **Only if no item found**: This option will continue the custom action execution flow only when no items are fetched in the Query List action. Consider a Custom Action configured on Task list to set the status of the Task as Closed. And requirement is to Close the related Request of the Task if all Tasks of the Request are closed (Task has a lookup to Request). In this case, after updating the Task status as Closed, we can add a Query List action to check if all Tasks related to the Request are already Closed. Query List action will try to fetch all ‘Open’ tasks for the Request and if no such tasks are found then it can continue execution. After this Query List action there will be another task to update the Request status as Closed. And this update action will run only if previous Query List action didn’t fetch any open Tasks.
* **Note**: All subsequent actions will be skipped if execution is stopped at the Query List action.