

1

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object

Opportunity

Configure Trigger

* Trigger the Flow When:

A record is created

A record is updated

A record is created or updated

A record is deleted

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

None

* Optimize the Flow for:

Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

Actions and Related Records

Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed



Edit Decision

Opportunity Account Changed (Opportunity_Account_Changed)

Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	
Account Changed	<p>*Label: Account Changed</p> <p>*Outcome API Name: Account_Changed</p> <p>Condition Requirements to Execute Outcome: All Conditions Are Met (AND)</p> <p>Resource: \$Record > Account ID Operator: Does Not Equal Value: \$Record_Prior > Account ID</p> <p>+ Add Condition</p> <p>When to Execute Outcome</p> <p><input checked="" type="radio"/> If the condition requirements are met</p> <p><input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements</p>	Delete Outcome

[Cancel](#) [Done](#)

Edit Decision

Opportunity Account Changed (Opportunity_Account_Changed)

Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	
New Opportunity	<p>*Label: New Opportunity</p> <p>*Outcome API Name: New_Opportunity</p> <p>Condition Requirements to Execute Outcome: All Conditions Are Met (AND)</p> <p>Resource: \$Record_Prior > Account ID Operator: Is Null Value: True</p> <p>+ Add Condition</p> <p>When to Execute Outcome</p> <p><input checked="" type="radio"/> If the condition requirements are met</p> <p><input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements</p>	Delete Outcome

[Cancel](#) [Done](#)

3

Opportunities #4 - V3

Edit Get Records

Find Salesforce records and store their field values in flow variables.

Get Contacts (Get_Contacts)

Get Records of This Object

* Object
Contact

Filter Contact Records

Condition Requirements
All Conditions Are Met (AND)

Field	Operator	Value
AccountId	Equals	Record > Account ID

+ Add Condition

Sort Contact Records

Sort Order
Not Sorted ⚠ If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store

- Only the first record
- All records

How to Store Record Data

- Automatically store all fields
- Choose fields and let Salesforce do the rest
- Choose fields and assign variables (advanced)

Select Contact Fields to Store in Variable

Field
ID

Field
Search fields...

+ Add Field


Cancel Done





Edit Loop

Start a loop path for iterating over items in a collection variable. For each iteration, the flow temporarily stores the item in the loop variable.

Loop Contacts (Loop_Contacts) 

Select Collection Variable

* Collection Variable

{!Get_Contacts}

Specify Direction for Iterating Over Collection

* Direction

- First item to last item
- Last item to first item



To use the current item in other elements in the loop, use the API name of the Loop element. Example: if your flow iterates over accounts with a Loop element named "My_Account_Loop" you can reference the current item from that loop element. Just start typing "My_Account_Loop" and select "Current Item from Loop My_Account_Loop".


Cancel

Done



5

Edit Assignment

Assign OCR Values (Assign_OCR_Values) 

Set Variable Values

Each variable is modified by the operator and value combination.

Variable	Operator	Value
<input type="text" value="varOCR > Contact ID"/>	<input type="text" value="Equals"/>	<input type="text" value="Current Item from Loop Loop_Conta..."/>
<input type="text" value="varOCR > Opportunity ID"/>	<input type="text" value="Equals"/>	<input type="text" value="SRecord > Opportunity ID"/>
<input type="text" value="varOCR > Role"/>	<input type="text" value="Equals"/>	<input type="text" value="Decision Maker"/>

New Resource

* Resource Type

* API Name

This API name is already used for another element or resource in the flow.

Description

* Data Type
 Allow multiple values (collection)


* Object

Availability Outside the Flow
 Available for input
 Available for output









6

Edit Assignment

Assign OCR to Collection (Assign_OCR_to_Collection) 

Set Variable Values

Each variable is modified by the operator and value combination.

Variable	Operator	Value	
 colOCR 	Add 	 varOCR 	

[+ Add Assignment](#)

[Cancel](#) [Done](#)

New Resource

* Resource Type
Variable

* API Name
colOCR
This API name is already used for another element or resource in the flow.

Description

* Data Type
Record Allow multiple values (collection)

* Object
Opportunity Contact Role

Availability Outside the Flow
 Available for input
 Available for output


[Cancel](#) [Done](#)



7

Edit Create Records

Create Salesforce records using values from the flow.

Create OCR (Create_OCR) 


How Many Records to Create

One

Multiple

Select Values to Create Multiple Records

* Record Collection

For each item in the collection, make sure that ID is blank. After the flow creates the records, each ID is set to the ID of the corresponding record that was created. 



8

Opportunities - V1

Edit Get Records

Find Salesforce records and store their field values in flow variables.

Get OCR (Get_OCR)

Get Records of This Object

* Object
Opportunity Contact Role

Filter Opportunity Contact Role Records

Condition Requirements
All Conditions Are Met (AND)

Field	Operator	Value
OpportunityId	Equals	!Record > Opportunity ID

+ Add Condition

Sort Opportunity Contact Role Records

Sort Order
Not Sorted ⚠ If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store

- Only the first record
- All records

How to Store Record Data

- Automatically store all fields
- Choose fields and let Salesforce do the rest
- Choose fields and assign variables (advanced)

Select Opportunity Contact Role Fields to Store in Variable

Field
ID

Field
Search fields...

+ Add Field

Cancel Done

Assign OCR to Collection Assignment



9

Edit Decision

Null Check (Null_Check)

Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS
1	<p>Is Null</p> <p>*Label: Is Null</p> <p>*Outcome API Name: Is_Null</p> <p>Condition Requirements to Execute Outcome: All Conditions Are Met (AND)</p> <p>Resource: Opportunity Contact Role from Get_OCR X</p> <p>Operator: Is Null</p> <p>Value: True X</p> <p>+ Add Condition</p> <p>When to Execute Outcome</p> <ul style="list-style-type: none"><input checked="" type="radio"/> If the condition requirements are met<input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements

10

Edit Delete Records

Delete Salesforce records.

Delete Existing OCR (Delete_Existing_OCR)

How to Find Records to Delete

- Use the IDs stored in a record variable or record collection variable
- Specify conditions

Select Record(s) to Delete

* Record or Record Collection

Opportunity Contact Role from Get_OCR X

Make sure that each record has an ID. Otherwise the flow can't find the records to delete, and it fails.

