

Day-22

Introduction of Flow

- Flow provides declarative process automation.
- Lightning flow provides this point and click automation tool i.e., Flow builder.
- Lightning flow is the name of product.
- Process builder and flow builder are name of the tools.
 - Use process builder to create processes.
 - Use flow builder to create flow.

When to use?

- Automated a guided visual experience.
 - Screen flow
- To build complex functionality which can run automatically in the background.
 - Record Triggered flow.
 - Schedule Triggered flow.
 - Auto launched flow.
- Need to call flows from apex or lightning components.

Flow Elements:

- Interaction
 - Screen
 - Action
 - Subflow
- Logic
 - Assignment
 - Decision
 - Loop
 - Collection sort
 - Collection filter
- Data
 - Create Records
 - Update Records
 - Get Records
 - Delete Records

Resources that can be created:

- Variables
- Constant
- Formula

- Text template
- Choice
- Record choice set
- Picklist choice set
- Stage

Flow must be:

- Activated
 - So that it works.
- Deactivated
 - So that it won't work.

Types of flow:

- Screen flow.
- Record triggered flow.
- Schedule triggered flow.
- Platform event triggered flow.
- Auto launched flow.

1) Screen flow:

- Screen flow guides through a business process.
- It can be launched through,
 - lighting page,
 - experience cloud sites,
 - quick actions
 - and more
- screen flow is the only flow which is visible to user and other type of flow is not visible to users its run-in background.
- We can place screen flow in any type of lighting page like, App page, Record page and home page.

2) Record triggered flow:

- Record triggered flow launches when a record is,
 - Created
 - Updated or
 - Deleted
- This auto launched flow runs in the background.

3) Schedule – triggered flow:

- Schedule- triggered flow launches at a specific time and frequency for each record in a batch.
- This auto launched flow also runs in the background.

4) Platform event – triggered flow:

- Platform event – triggered flow launched when a platform event message is received.
- It used rarely.
- This auto launched flow also runs in the background.

5) Auto launched flow:

- It is invoked or call by any external features like,
 - Apex
 - Processes (process builder)
 - REST API
 - And more
- This auto launched flow also runs in the background.

Sanjay Gupta Tech School

	Process Builder	Flow Builder	Workflow	Approvals
Complexity	Multiple if/then Statements	Complex	A single if/then Statement	A single if/then Statement
Visual Designer	Yes	Yes	No	No
Supports Time-based Actions	Yes	Yes	Yes	No
Supports User Interaction	No	Yes	No	

Sanjay Gupta Tech School

	Process Builder	Flow Builder	Workflow	Approvals
Starts when	1. Record is changed 2. Invoked by another process 3. Platform event message is received	1. User click button or link 2. User accesses Lightning Page, Experience Site Page, VF Page or Custom Tab 3. User Accesses item in a utility bar 4. Process starts 5. Apex is called 6. Record is Created, Updated or Deleted	Record is changed	1. User clicks button or link 2. Process or flow starts that includes a Submit for Approval action 3. Apex is

Day 22 | Salesforce Bootcamp 2023 | Introduction to Flow | Types of Flows | Understand Flow

34:03 / 1:22:35

Difference between process builder, flow builder, workflow and approvals based on supported action:

Sanjay Gupta Tech School

	Process Builder	Flow Builder	Workflow	Approvals
Call Apex Code	Yes	Yes	No	No
Create Records	Yes	Yes	Task only	Task only
Invoke Sub Processes	Yes	Yes	No	No
Delete Records	No	Yes	No	No



Sanjay Gupta Tech School

	Process Builder	Flow Builder	Workflow	Approvals
Launch a flow	Yes	Yes	No	No
Post to Chatter	Yes	Yes	No	No
Send email	Yes (Email alerts only)	Yes	Yes (Email alerts only)	Yes (Email alerts only)
Send outbound messages without code	No	Yes	Yes	Yes



Sanjay Gupta Tech School

	Process Builder	Flow Builder	Workflow	Approvals
Submit for Approvals	Yes	Yes	No	No
Update fields	The record or its parent or child	Any record	The record or its parent	The record or its parent
Custom Notification	Yes	Yes	No	No
Quick Action	Yes	Yes	No	No
Quip	Yes	Yes	No	No



Flow builder demo (steps):

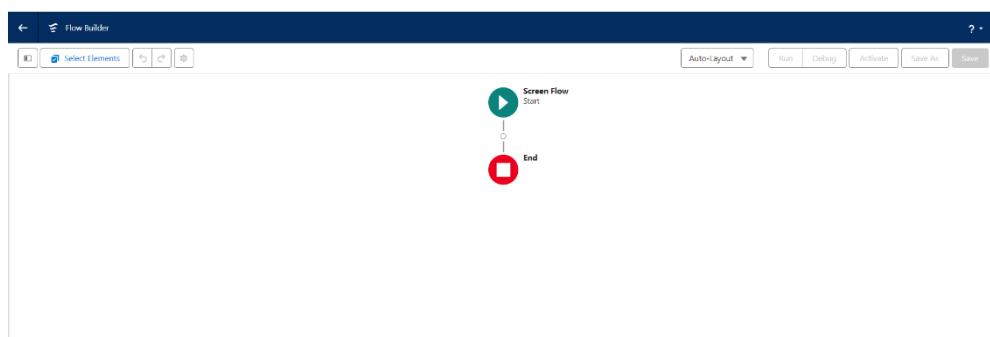
- To create a flow 1st, go to setup after that go to home and search for **flows** in quick search. After that click on flows. After that click on new flow.

The screenshot shows the Salesforce Setup interface. In the left sidebar, under 'Process Automation', the 'Flows' link is highlighted with a red box. At the top right, there is a 'New Flow' button, which is also highlighted with a red box. The main area displays a list of 'Flow Definitions' with various columns like 'Flow Label', 'Process Type', 'Active', 'Template', etc. A message at the top says 'Didn't find what you're looking for? Try using Global Search.'

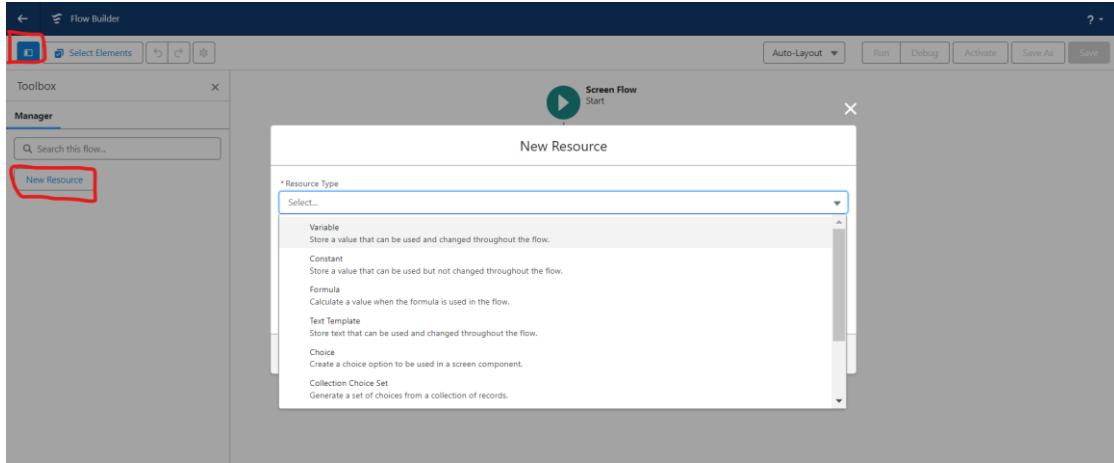
- After that different type of flow screen will open and we have to choose one flow. Here, we are selecting screen flow. And click on create.

This screenshot shows the 'New Flow' creation screen in the Flow Builder. On the left, there's a 'Toolbox' with tabs for 'Elements' and 'Manager'. The 'Core' tab is selected, showing various flow types: 'Screen Flow' (highlighted with a red box), 'Record-Triggered Flow', 'Schedule-Triggered Flow', 'Autolaunched Flow (No Trigger)', and 'Record-Triggered Orchestration'. At the bottom right of the toolbox, there is a 'Create' button.

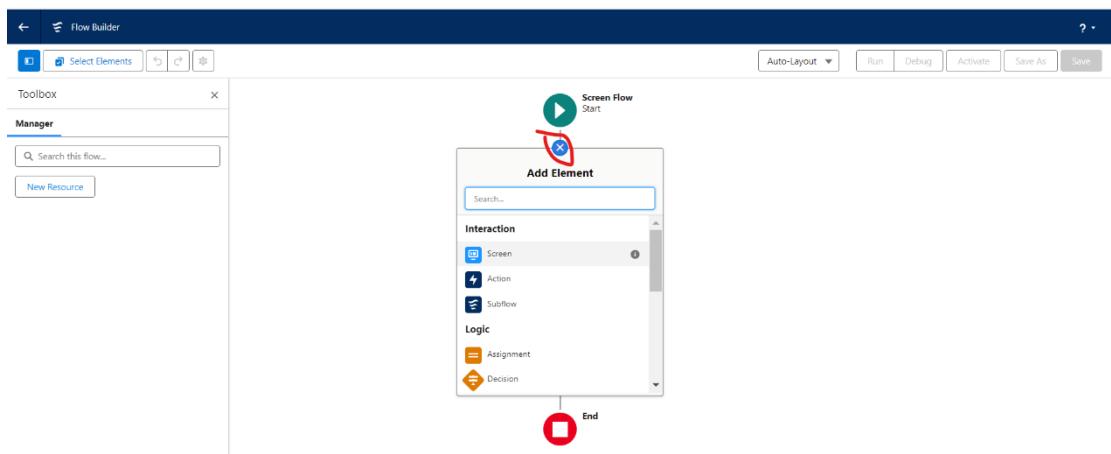
- After that flow builder will open.



- As we know that we are making flow with 2 things elements and resources.
- To take the resource we have to click on toggle button on left hand side and click on add resources button so add the resources.



- To take elements we have to click on + sign on flow builder canvas.



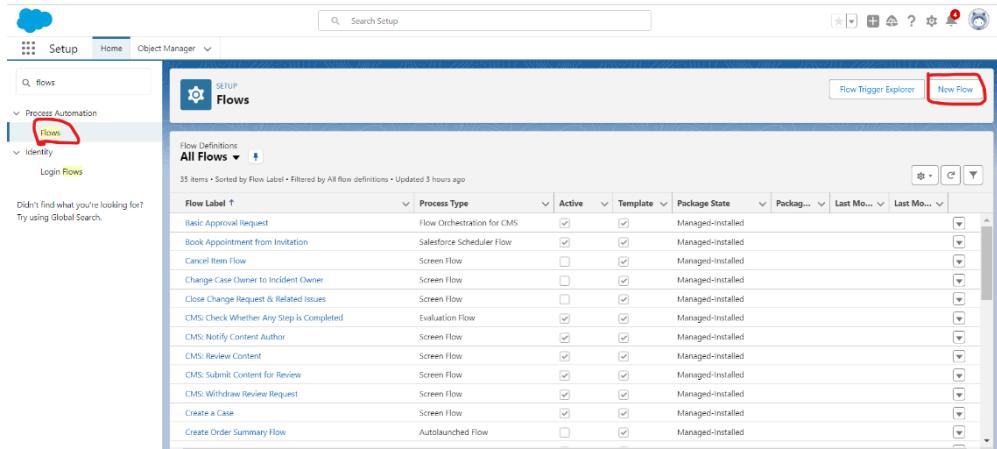
Screen flow

Scenario: -

1. create a screen flow to create opportunity related to an Account.
2. Debug the flow.
3. Launch screen flow through quick action and test.
4. Handle flow error.
5. Place screen flow on account page and test

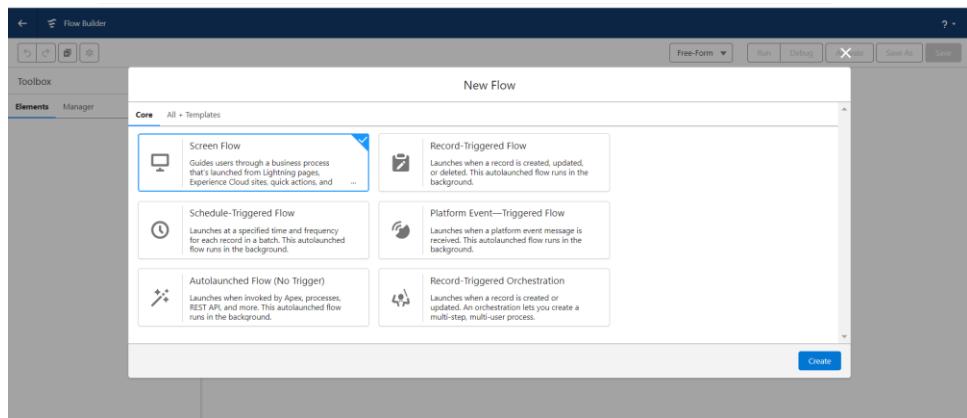
1. Steps to create screen flow:

- Go to setup and click on home page and search for flows in quick box.

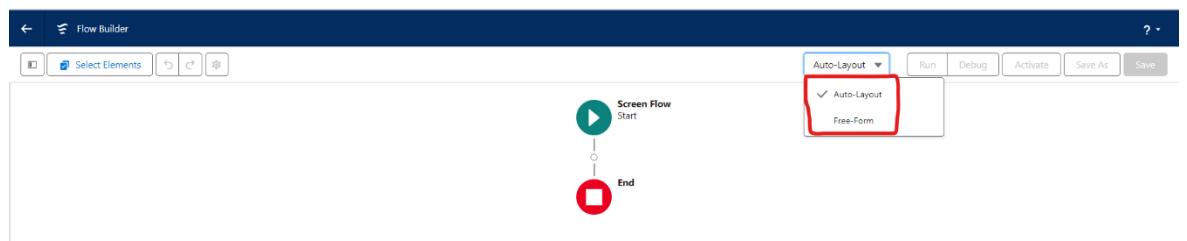


After that click on flows and click on new flow.

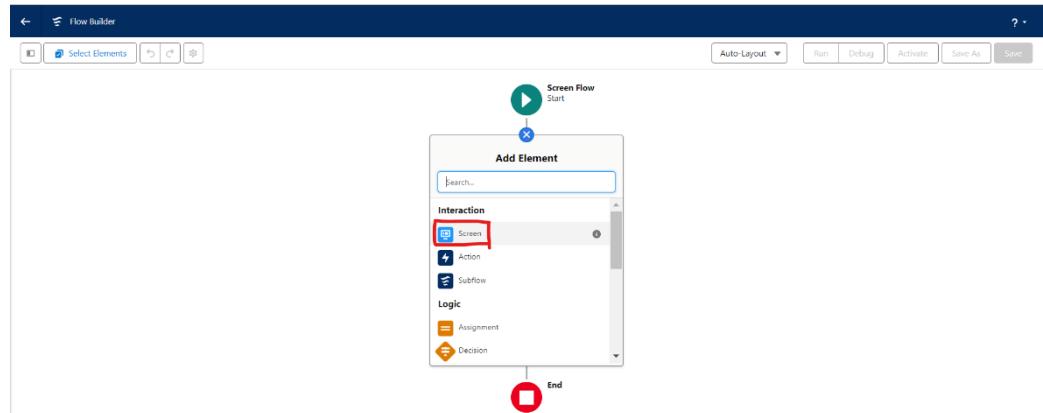
- After that one pop up will open to select the flow here, we are selecting screen flow. And click on create.



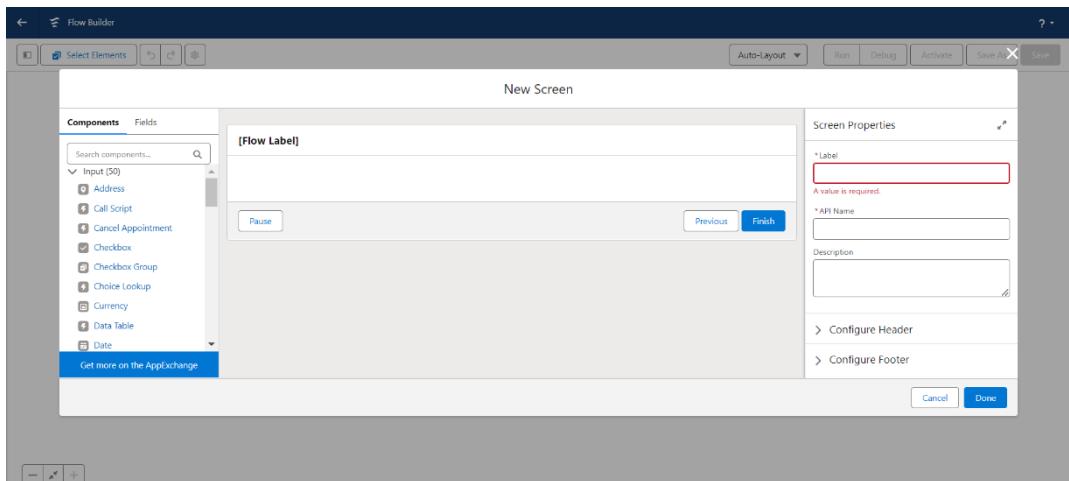
- As we know that we have 2 layout auto layout and free flow here, we are selecting auto-layout.



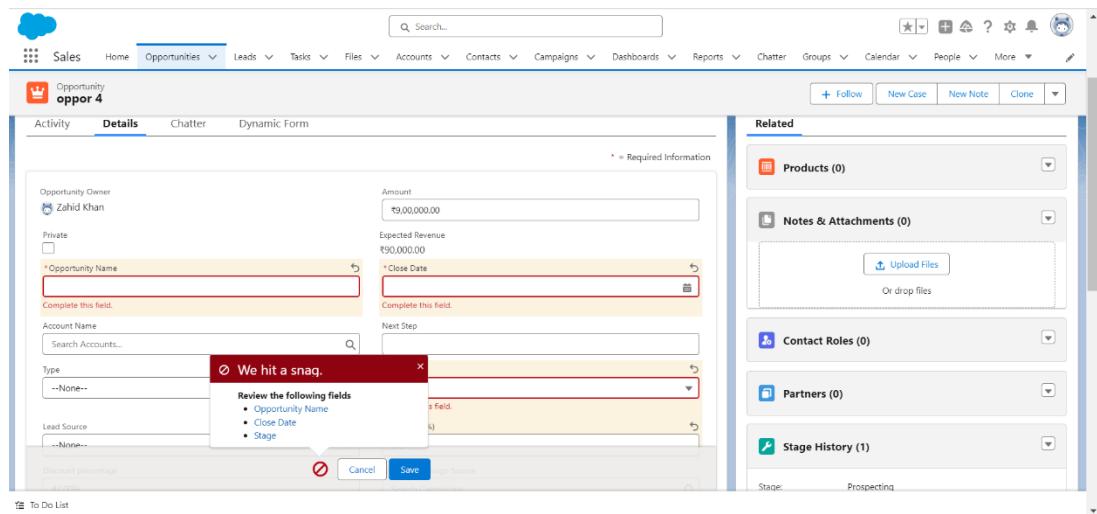
- Now click on the + sign we will see screen in flow this screen only visible only in screen flow not in triggered and record flow.



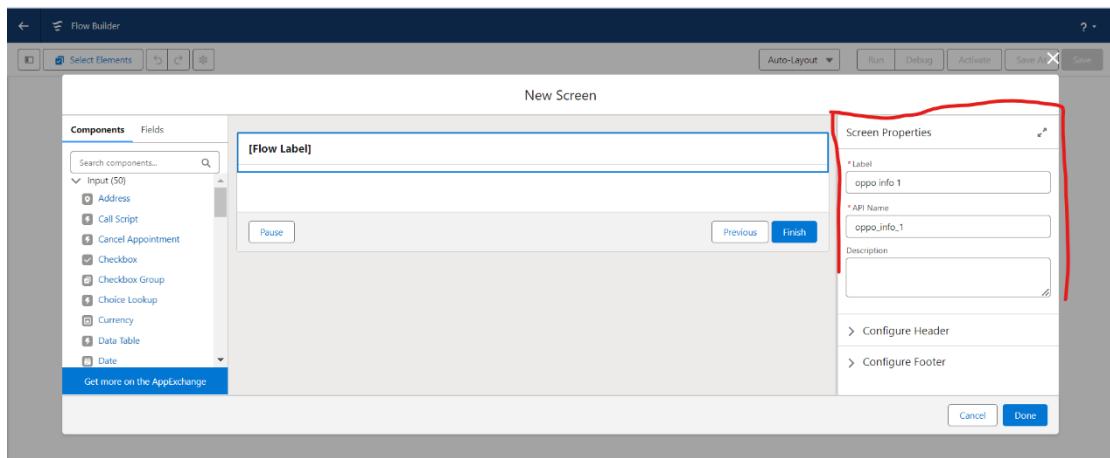
- Now click on screen, one new screen window will open. This is screen builder through this we can create screen and get input from user.



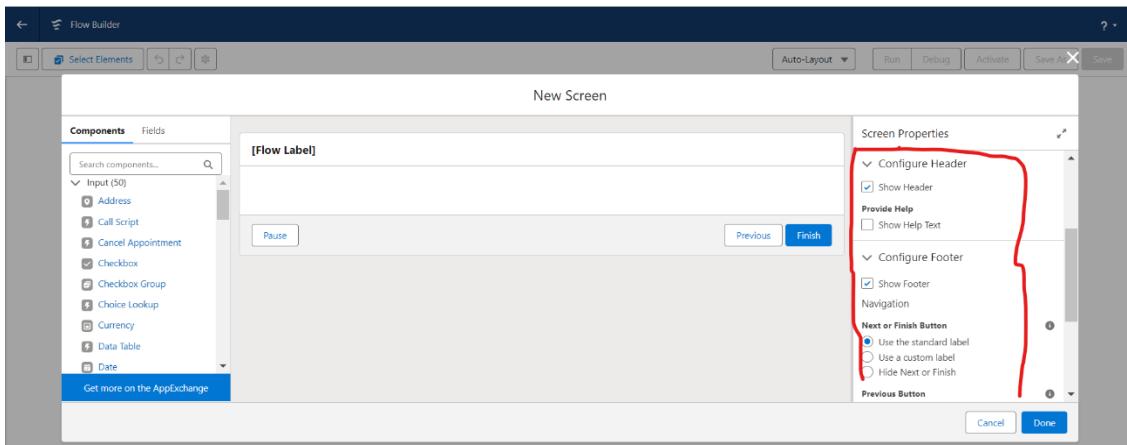
- Now we have to decide what are required field to create an opportunity record. For example, we are going to create opportunities record so go to sales application and click on opportunities tab and click on new to create new record and click on save without entering any value so that required field will be visible.
- Here, opportunity name, stage and closing date are the required fields. Its mean when we want these information from user through screen flow user have to enter this information otherwise we can't get information from user through screen flow.



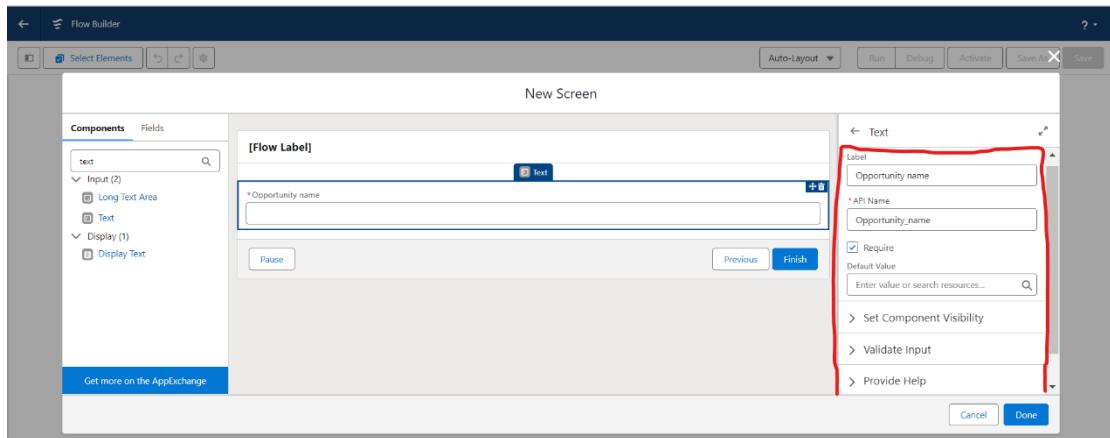
- Now go the screen flow builder, in left hand side we have screen properties in which we have to enter label and api name. screen is also an element in the flow which need identification.



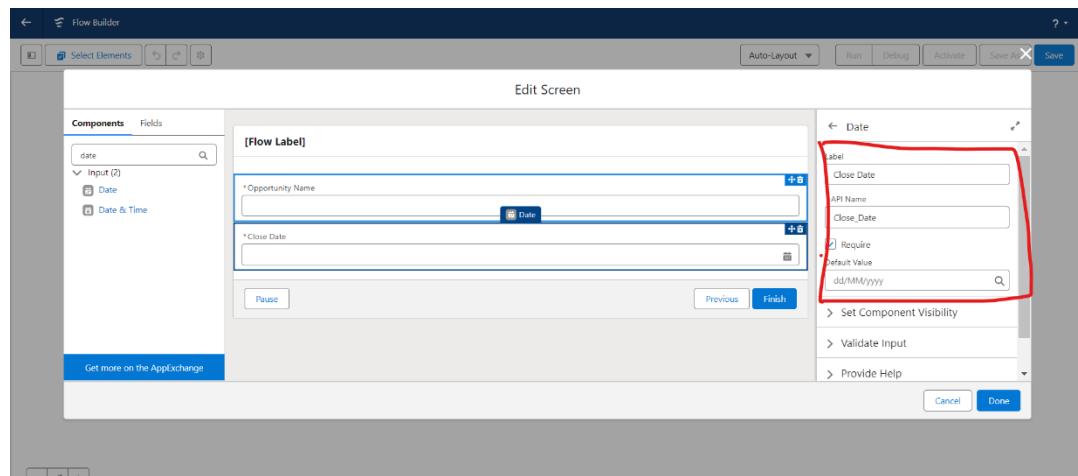
- In middle of the screen, we have header and footer using configure header and configure footer. Using configure header and footer option we can remove the header and footer (Flow label)



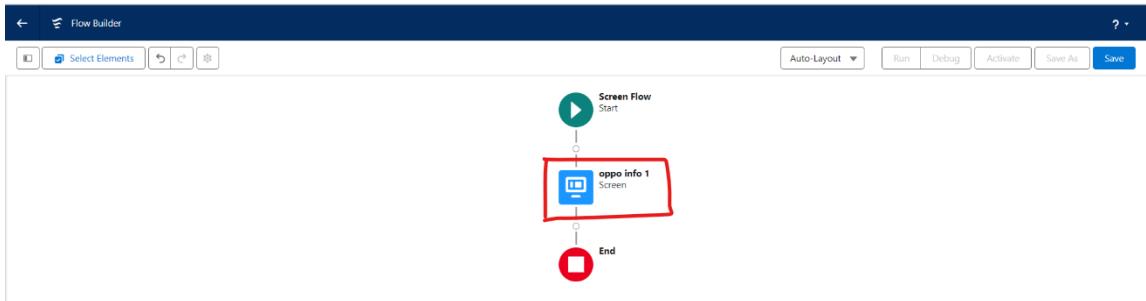
- In footer we have 3 buttons pause, previous and finish. We can control these buttons using configure footer.
- At left hand side we have components and fields. Here, we are going to add some components like opportunity name (text). Search text in components and click on it so that text comes in field.
- On right hand we have screen properties of the text components then required box. Apart from this we have component visibility (based on condition who can see this component). Then validate input (to show error message)



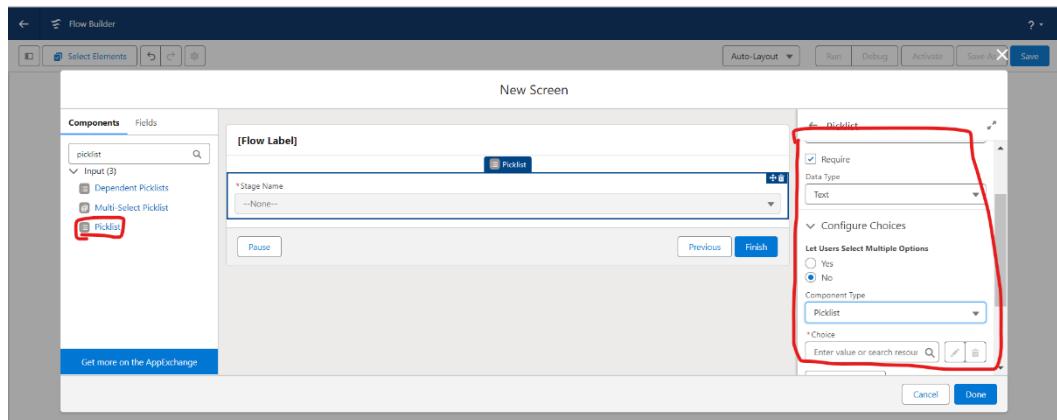
- In same way we have to enter close date. For this search for date in components and click on it so that it will show in middle screen. If we want to give default value so we can give using default value field.



- In this way we add two components text and date. Now we have to insert 3rd element stage.
- Now click on done so that 1 screen will created.

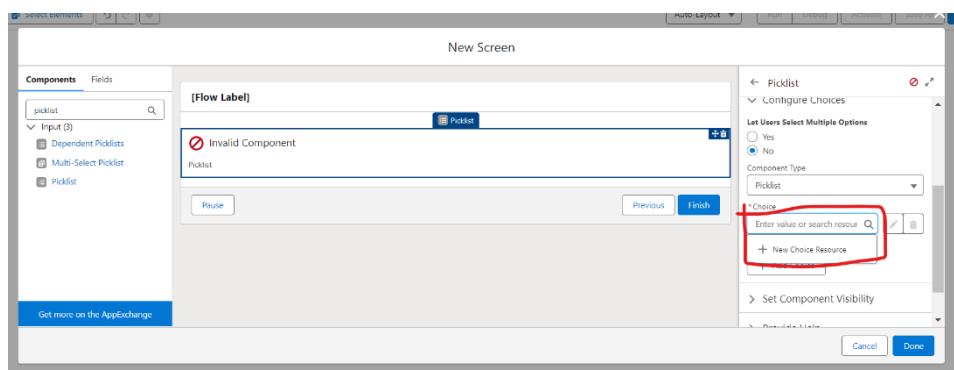


- after that click on + sign below 1st screen so that we can create another screen for stage. After that enter label for 2nd screen.
- Here, we have to make picklist because stage is a picklist in opportunity records. And it contains lots of picklist value. So, we need to make screen flow which populates all value.
- For this search for picklist in components and click on it so it will insert into the middle screen. After that enter label and api name, data type is text, let

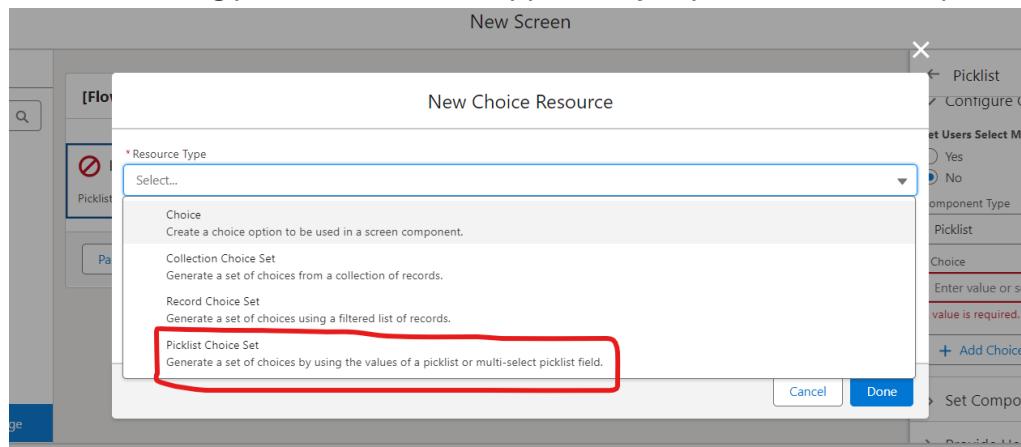


user select multiple options and components type picklist.

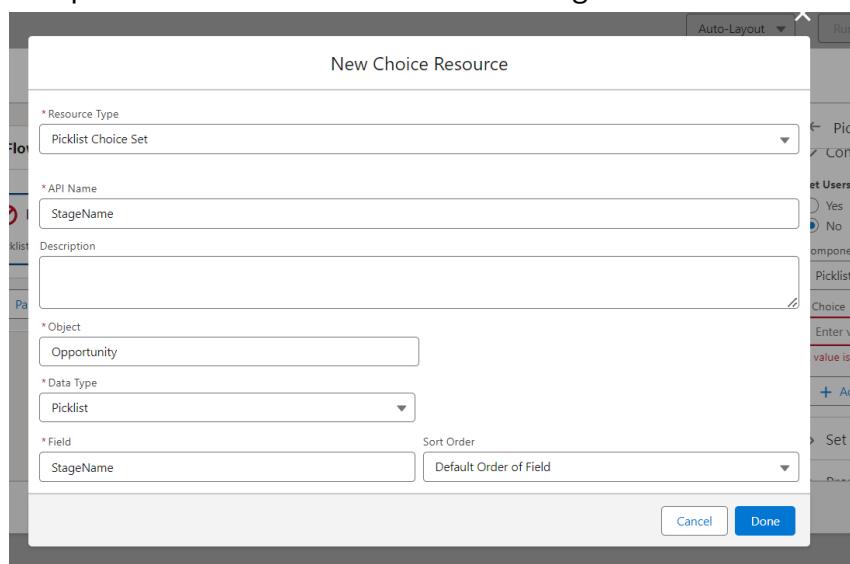
- Now we have to insert the picklist option same as opportunity object stage record options. Now we have 2 options either we can enter picklist data manually or we can extract picklist value directly from opportunity object's stage records.
- Here, we are choosing 2nd options. Now click on choice in right hand side of screen properties and click on new **+New choice Resources**.



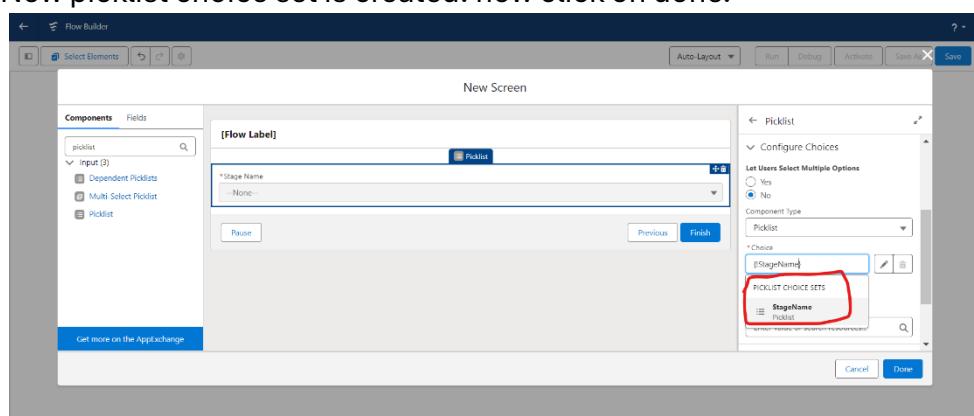
- After that new resource choice page will open. In which we have 4 options here we are taking picklist value from opportunity object so select 4th option.



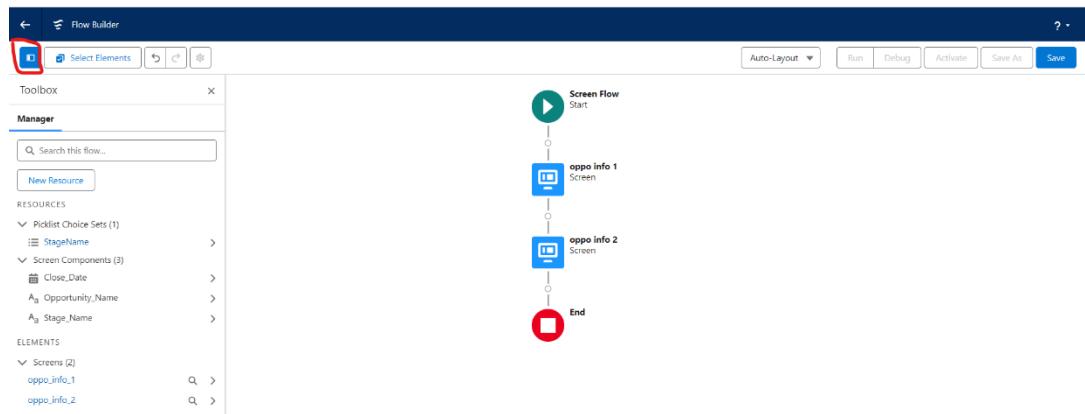
- After that enter Api name then search object as opportunity. After that select data type as picklist. After that select the field stageName and click on done.



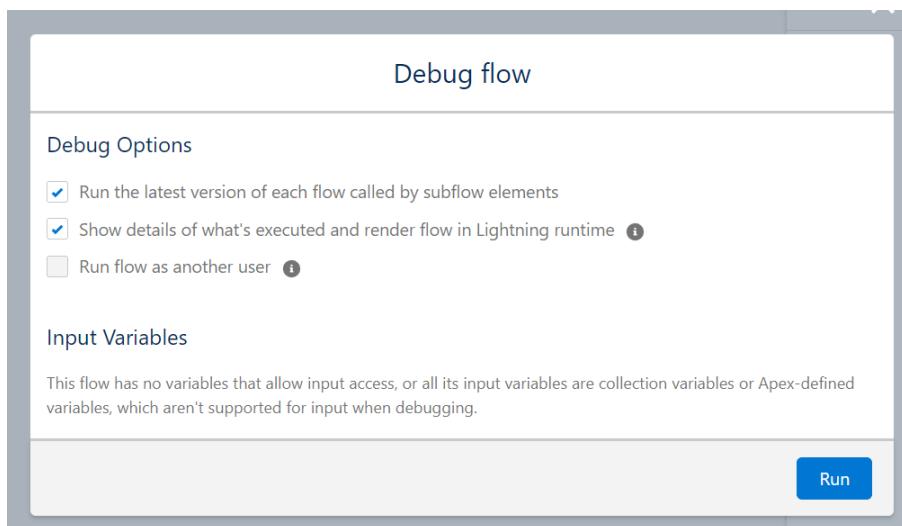
- Now picklist choice set is created. now click on done.



- In this way we created 2 screen flow. Click on the small icon on right hand side we will see the overview if components, resources, and screens.



- Now click on save button and enter label and api name.
- Now before activation we have to **debug** the flow as per scenario.
- Click on **debug** button so that we can assure that our flow is working fine.
- After that click on run to run the flow.

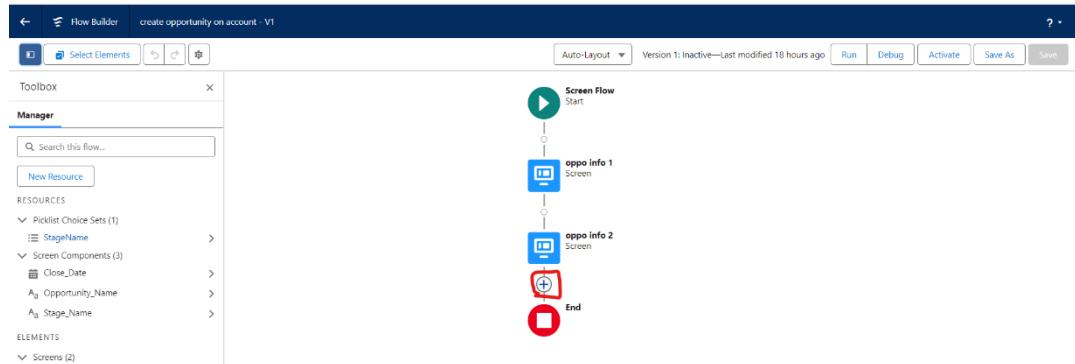


- In next page we will verify the flow like its working fine or not. 1st screen run first after that clicking on next button 2nd screen will run.
- After all, done just click on finish button.

1st screen

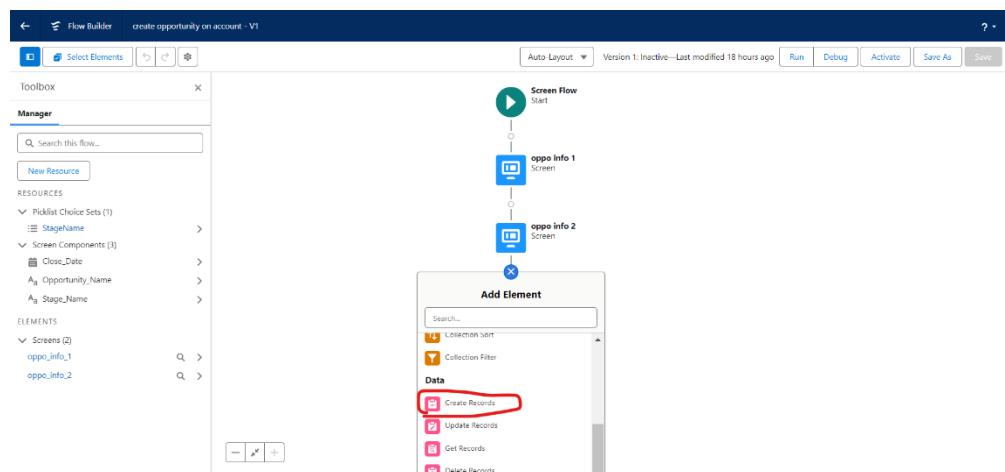
2nd screen

- After successfully run (debug) the flow returns to the canvas.
- Now after verifying that all the flow working fine like get opportunities name, close date, and stage. In the background we have to create an opportunity record.
- To create opportunities, record in background click on + sign below 2nd screen

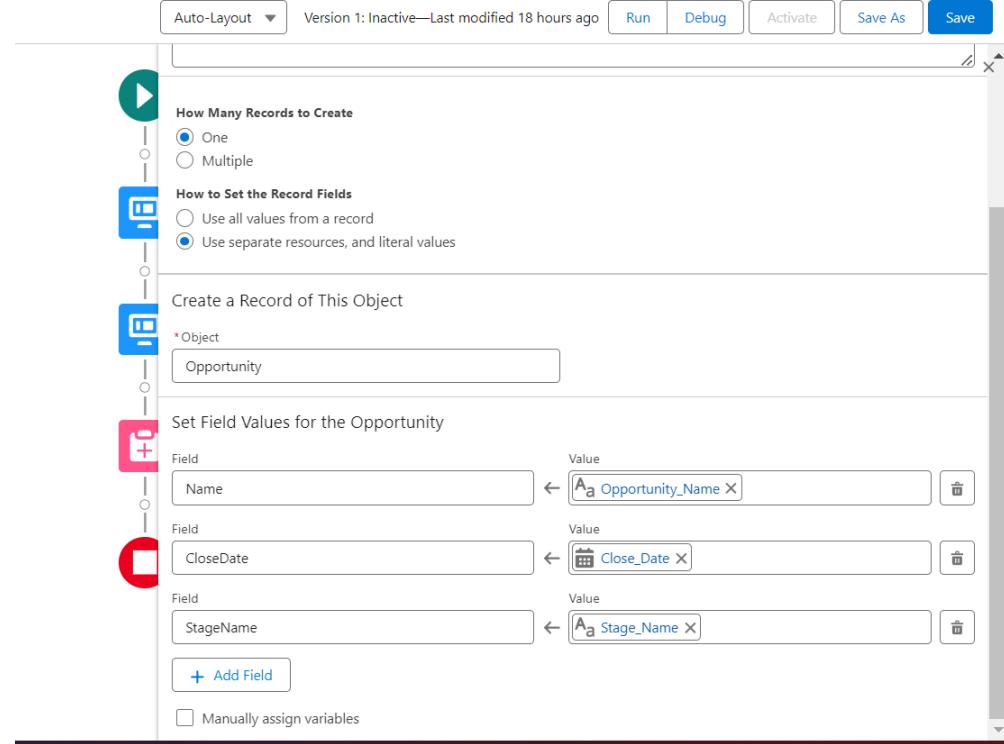


flow.

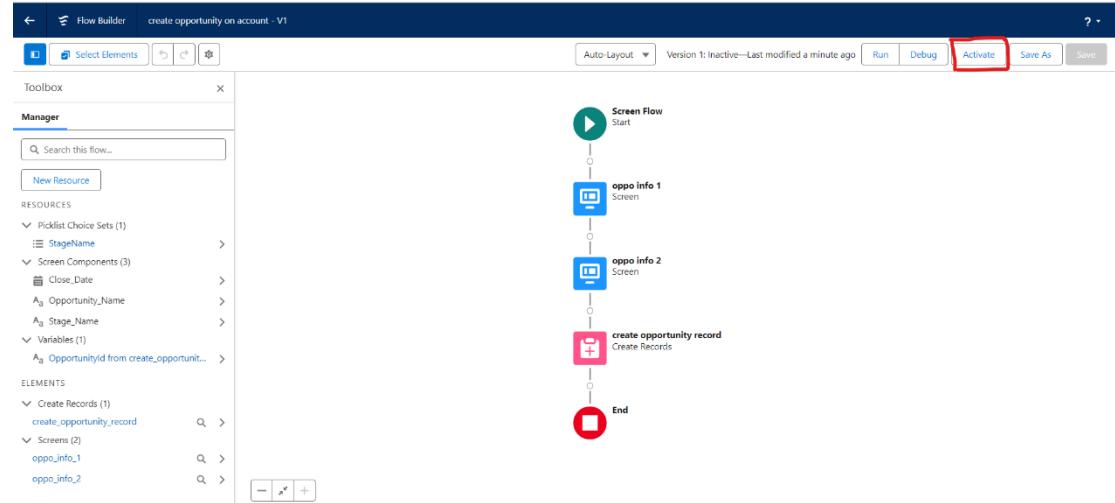
- After that select **create record** elements under **Data**.



- After that enter label and API name,
- In option **how many records to create** we are selecting **One** for this scenario,
- In **how to set record fields** we are **selecting Use separate resources, and literal values**. If we select **multiple** in **how many records to create** so we need to select **Use all values from a record**.
- After that select object as **opportunity**.
- After that we have to map the opportunity field with field components which we created in flow. Like **Name** with **opportunity name**, **close date** with **close date** and **stagename** with **stage**. And by clicking on **Add field button** we can add the numbers of field.
- After that click on done.

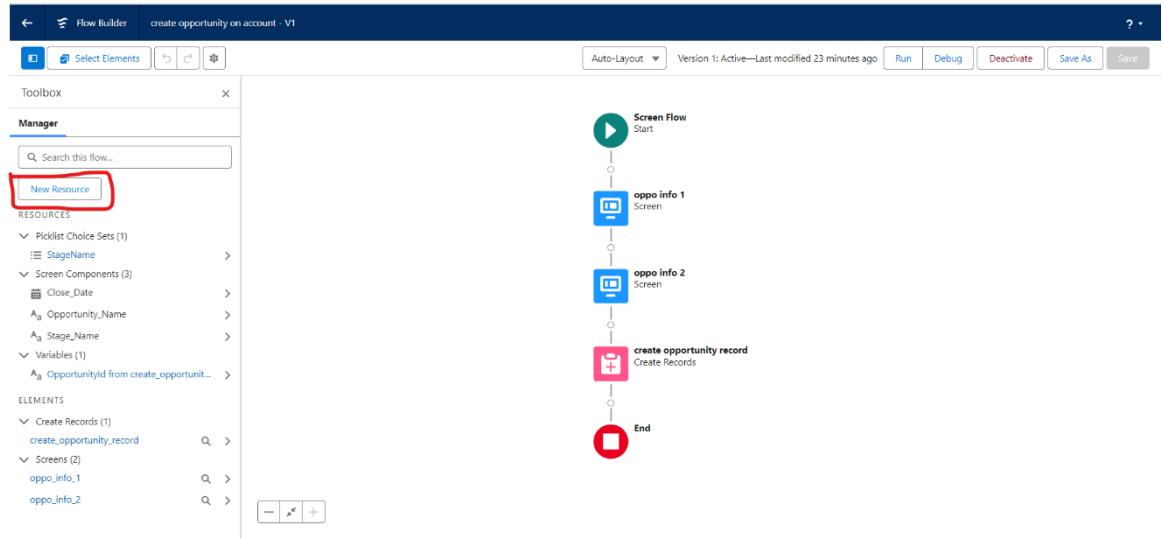


- After that create a record is created.
- After that click on save again.
- After that activate the flow by clicking on activate button.

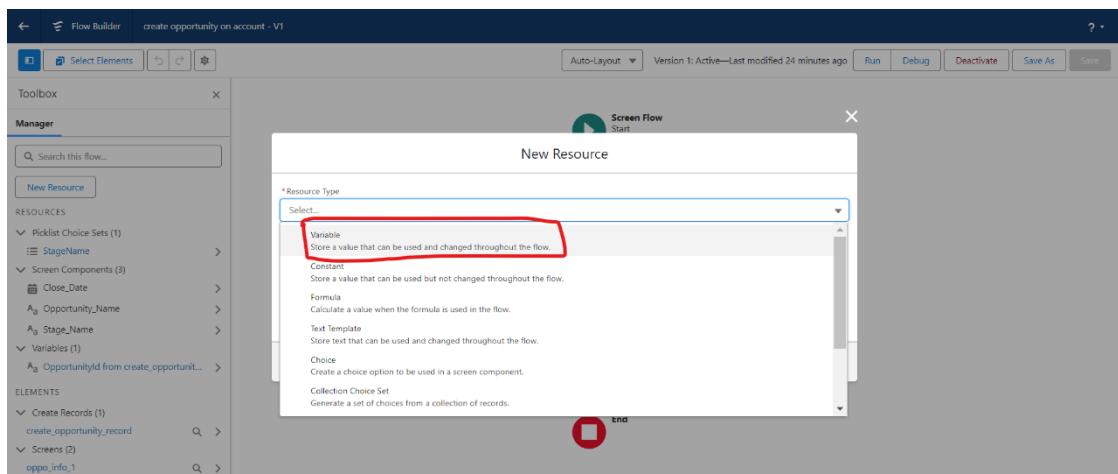


- Now we can debug and enter the value which is visible in opportunity records, but we can see that its not linked with account object, so we have to do some steps to add this.
- Now go back to screen flow canvas so that we can create a link between opportunities object flow with account object.

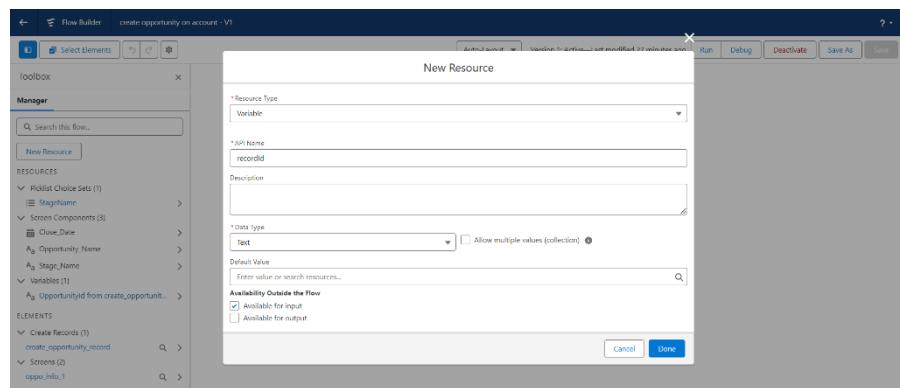
- To link the opportunities object with account object we have to create resources **variable**. For this click on Add resource button on at left hand side.



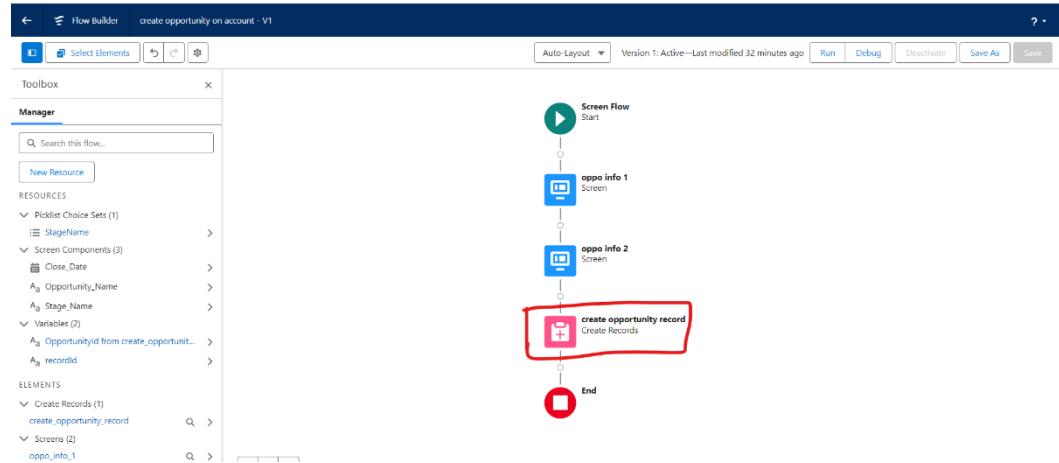
- After that select Variable from the drop-down menu to record the account id.



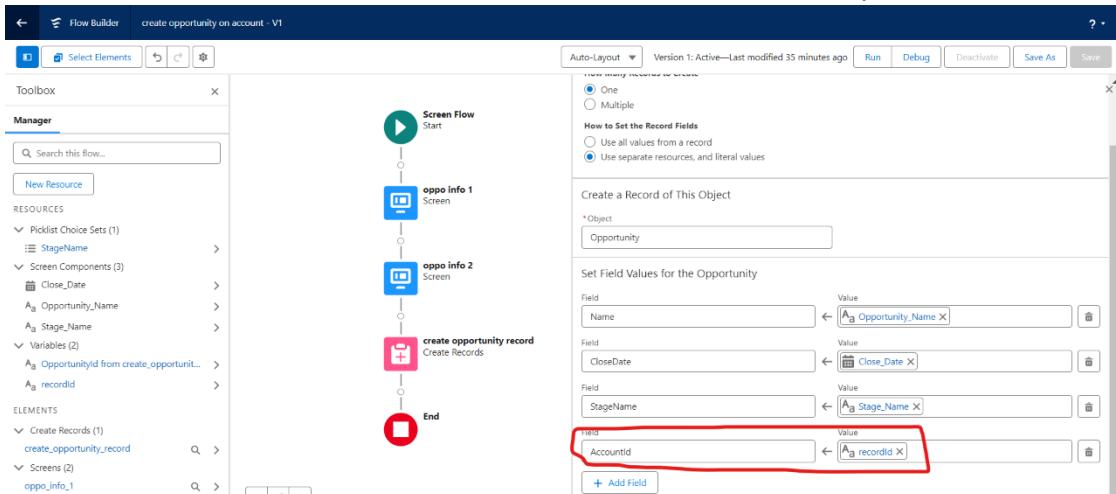
- After that enter the API name and api name should be **recordId** and data type is text and always select **available for input** in **availability outside the flow** option. And click on done.



- After that click on create record field which we created earlier to map the Id field. And click on edit elements.

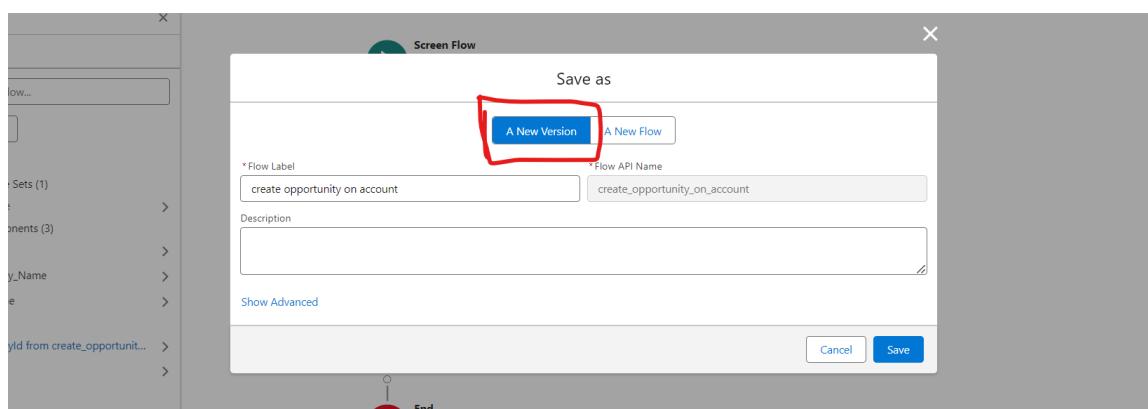


- After that click on Add field button and add the recordId variable. In field type account id and value will be recordId so that I will be lookup to account object.

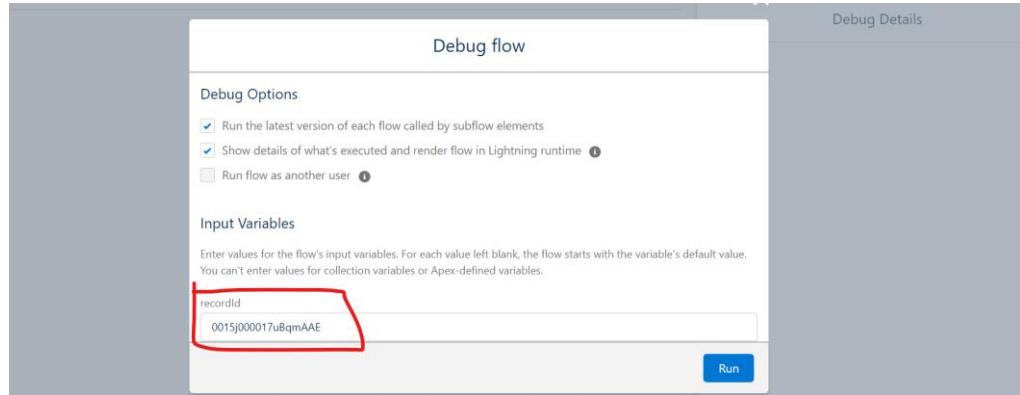


object.

- After that click on save as to save the flow and select **A new version**. And click on save. When do save as always click on activate button to activate current version.



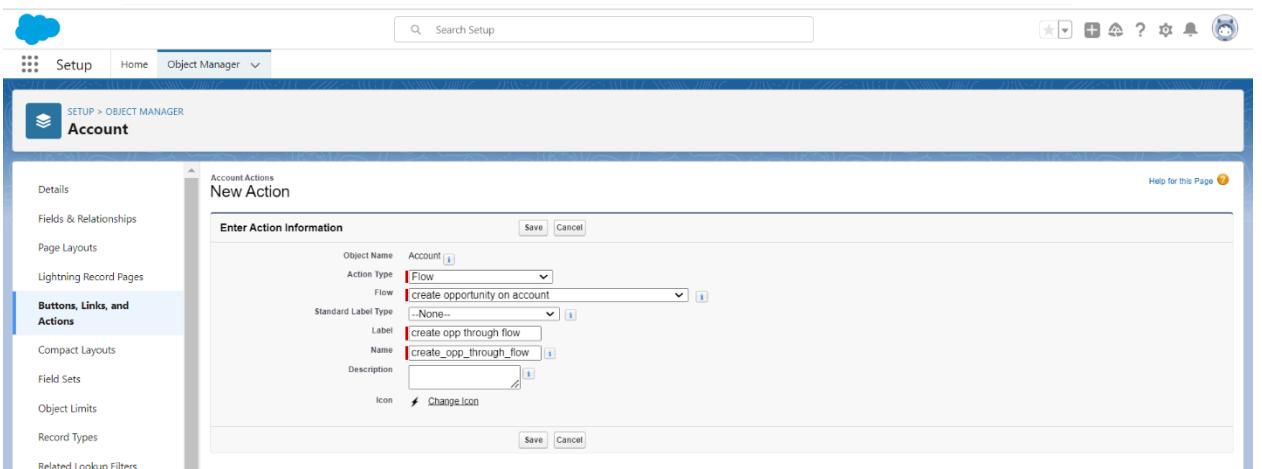
- After that again click on activate.
- After that again click on debug and check. When we click on debug button recordId variable also available, so we have to enter the account id of any records for this open record and copy the id from the url and paste it in debug window.



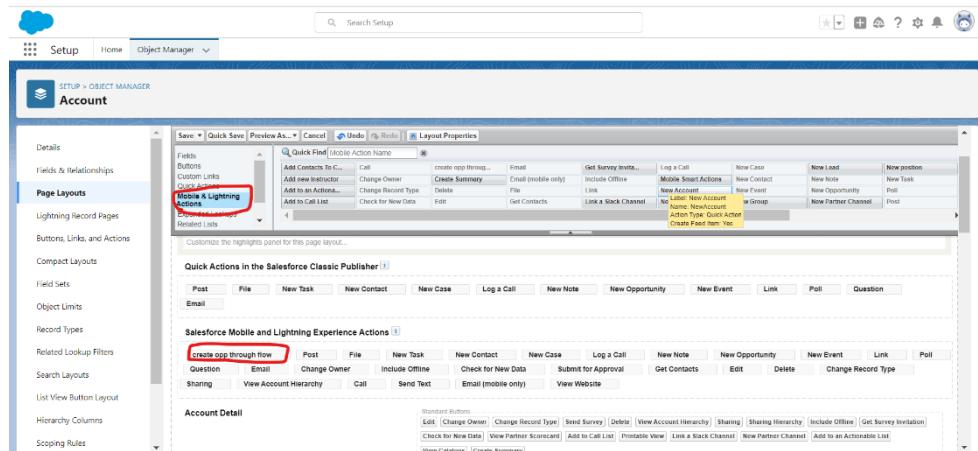
- Now we enter any value, so it relates to the account record of that account id.

Screen flow placing in quick action:

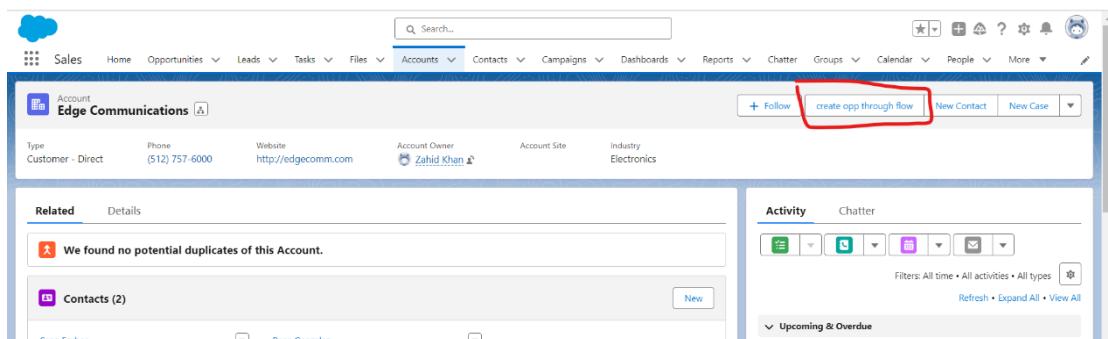
- Now we are going to place this flow in quick action.
- For this, go to object manager and select the account object we are created opportunity flow, which is related to account, so we have to add that flow to account object.
- After that click on button, links & actions.
- After click on new actions.
- After that select flow in Action type.
- After that select flow like create opportunity on account which we created. only active flow will be available here.
- After that enter label and name.
- Click on save.



- After that we have to add this action to page layout and select account layout and click on mobile and lighting action. Click the action we created and drag and drop inside the salesforce mobile and lighting experience.
- And click on save.

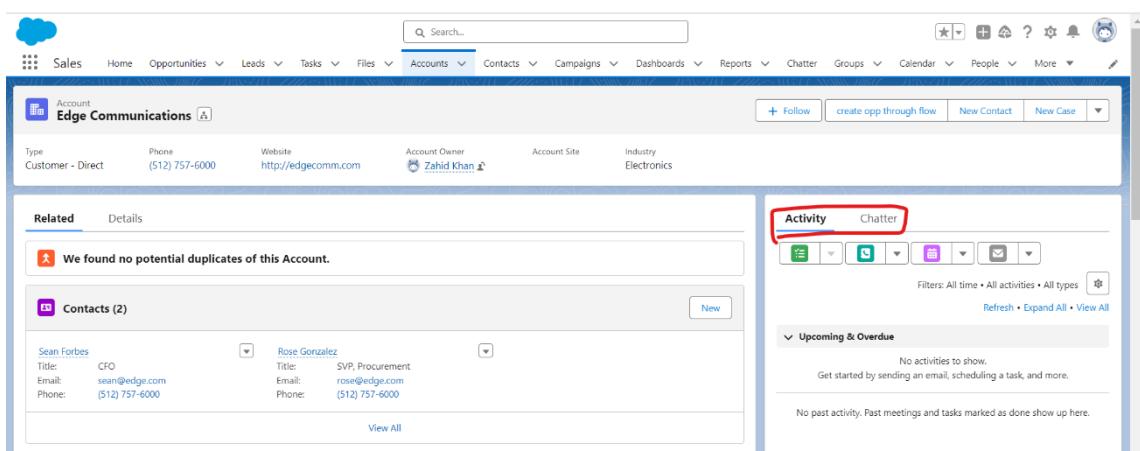


- After that go to account object page and open any record and check that button will available.

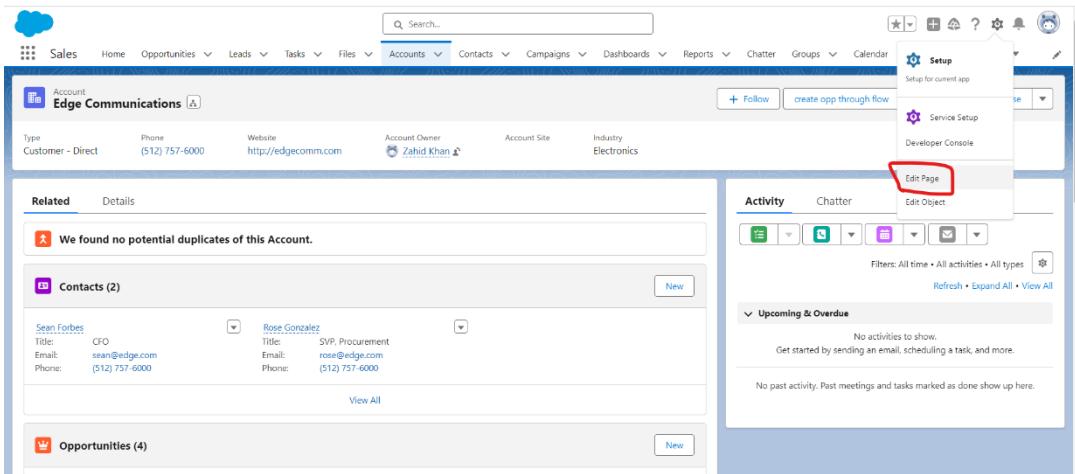


Place flow on record page (account object) record page:

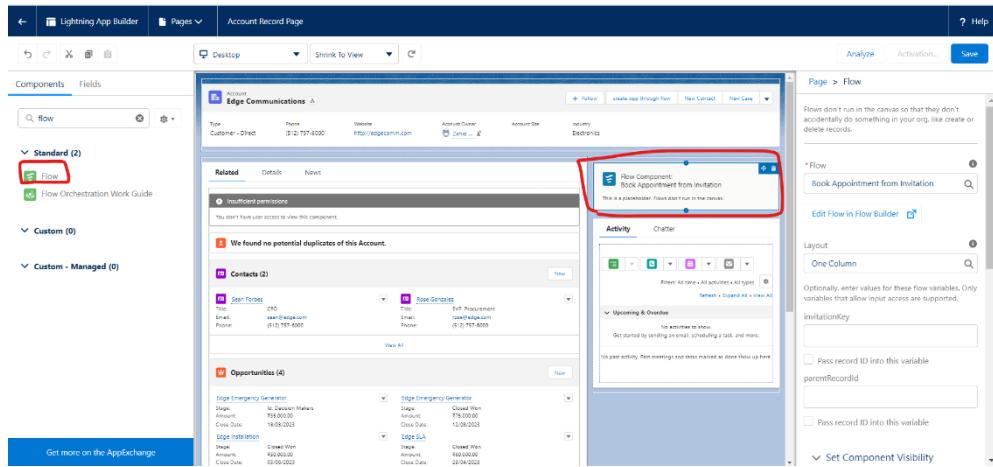
- Opportunity flow will be placed before activity and chatter component.



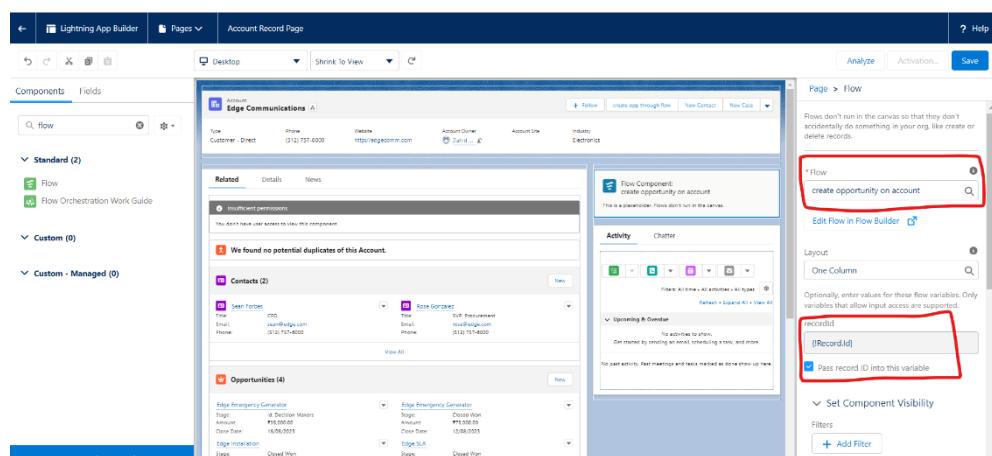
- Click on set-up icon and click on edit page. I will open lightning app builder.



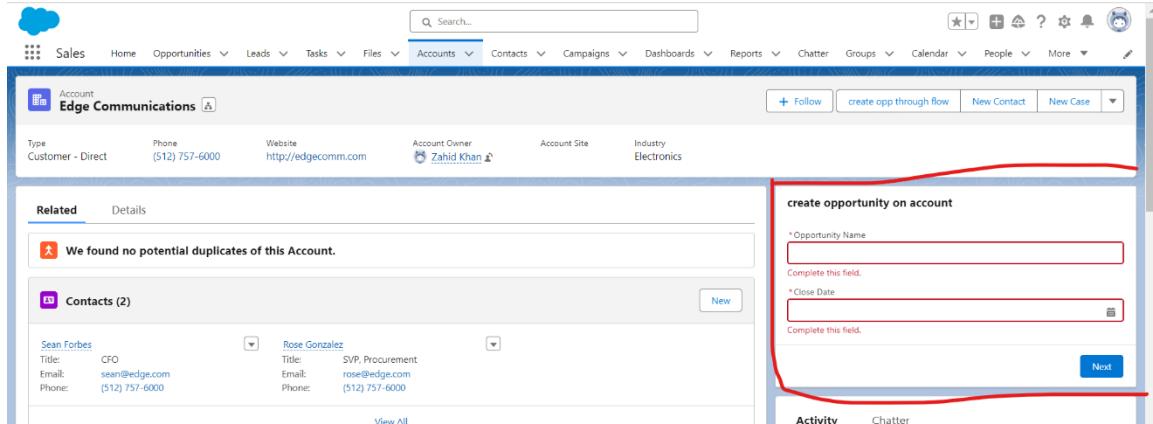
- After that search for flow on left hand side components. After that click drag and drop flow above activity tab.



- After that we can see the flow field at right hand side just select the flow which we created **create opportunity on account**. And enable the check box of pass recordId into this variable. After that record id will be stored in this variable.

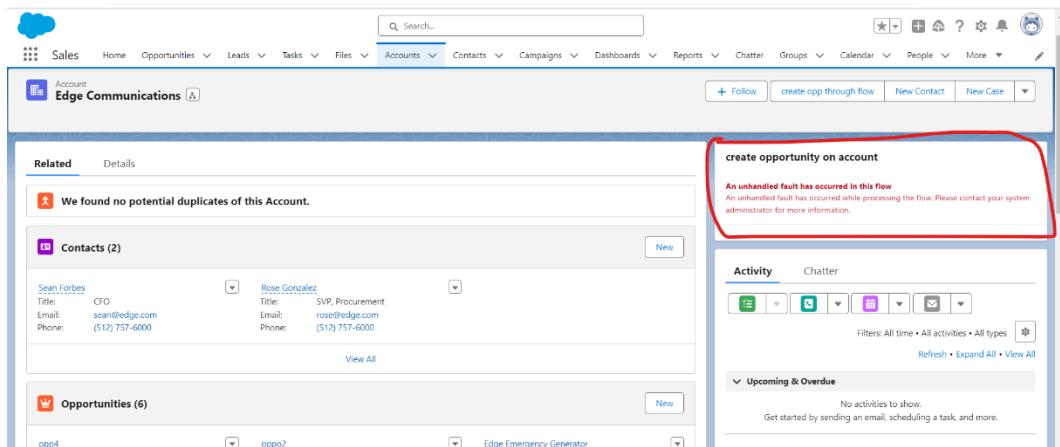


- After that click on save.
- After that go back to the record page and see the changes.

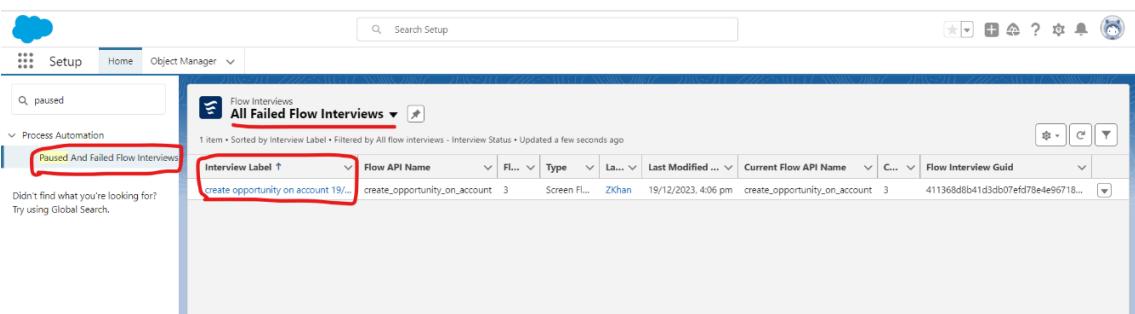


Handle flow error:

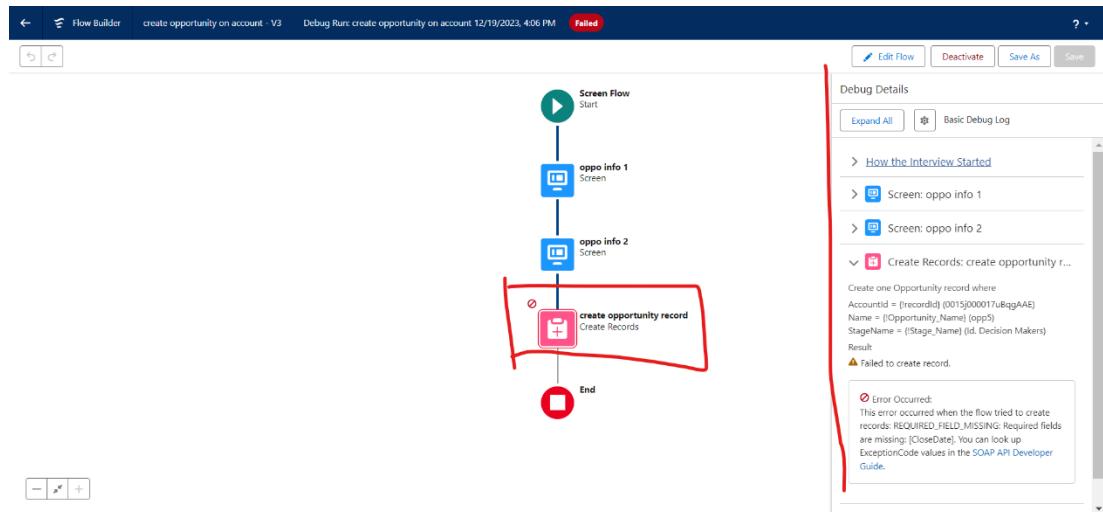
- Whenever we create flow and forgot to map any field then an error will come.



- To locate the error, go to set-up and search for paused and failed flow interviews. All failed flow interviews will show here.
- If we click on the interview label so that we can see the details of this error.



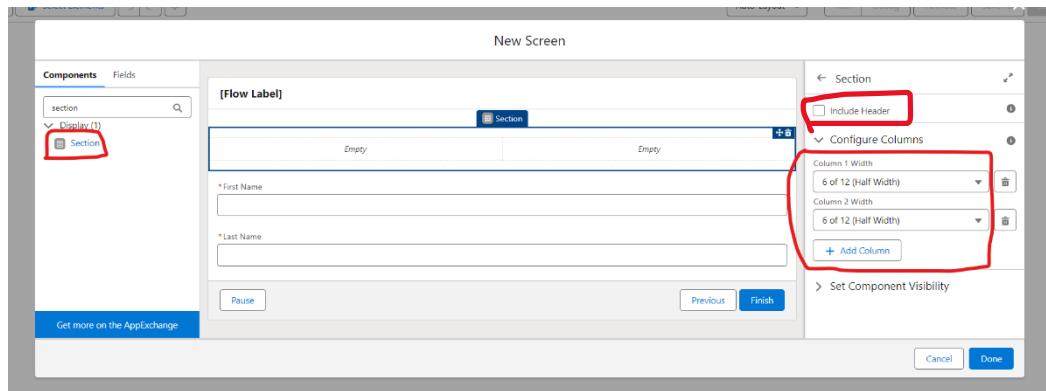
- We can see the error on right hand side.



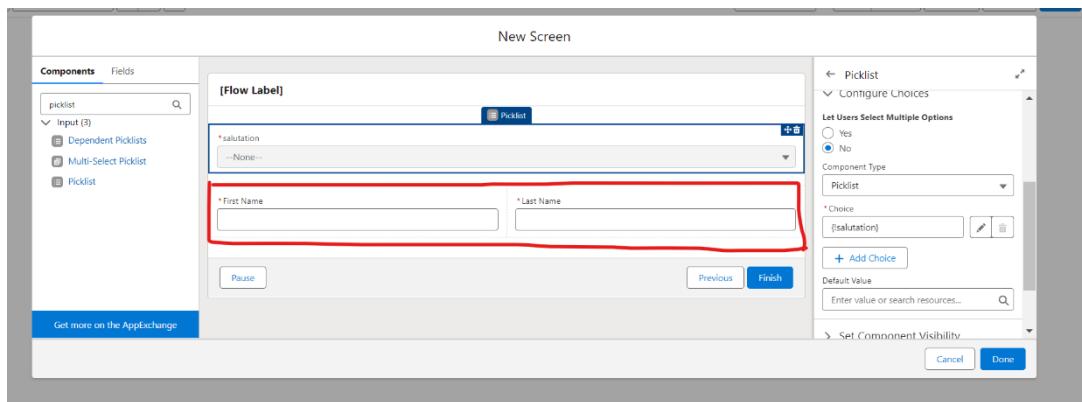
Flow continuation – Screen flow 2

Scenario:

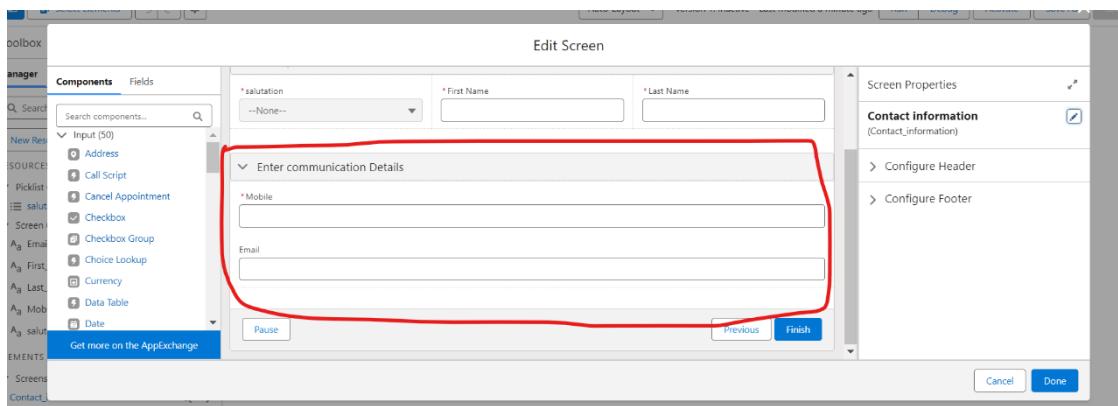
1. Create a contact by placing screen flow on home page of “zahid khan tech school”.
 2. Option to attach a file and file should be related to contact.
 3. Option to link contact with an account [lookup flow].
 4. Upon contact creation send an email.
 5. Try to place flow on app page and test.
1. **Create a contact by placing screen flow on home page of “zahid khan tech school”.**
 - 1st of all check the required field of contact in sales application which is last name.
 - After that we have to create a new flow for contact. For this go to home and search for flows in quick action and click on it. After that click on new flow button.
 - After that select **screen flow** because we want to take data from user.
 - After that click on + sign in canvas and select screen.
 - Then enter the screen flow label and Api name.
 - After that search text in components and place it in middle screen of screen flow.
 - Enter the label First name and Last Name. here, we are making two column layouts. For this search for section in components.
 - After that click on Add column button on right hand side so it creates 2 columns.
 - If we want to add header to this 2-column layout so we can add by check the box **includes header**.



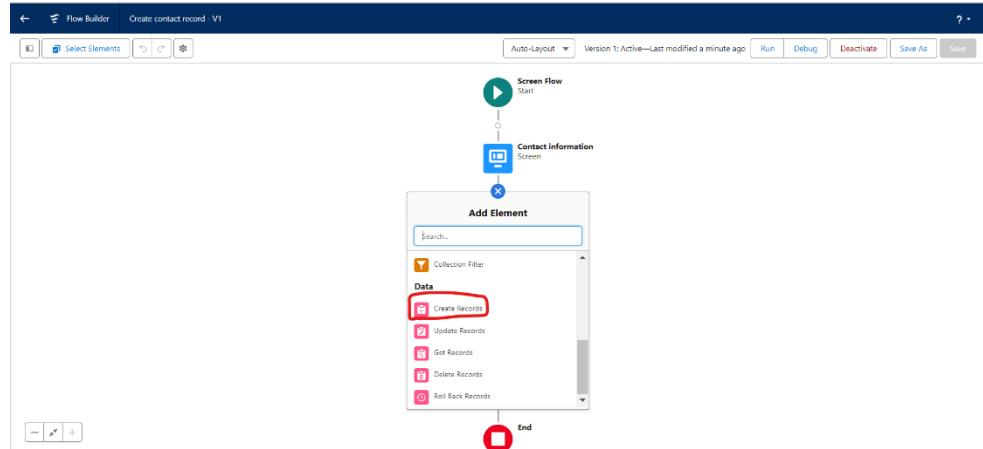
- Now drag the 1st name and last name into this 2-column section.



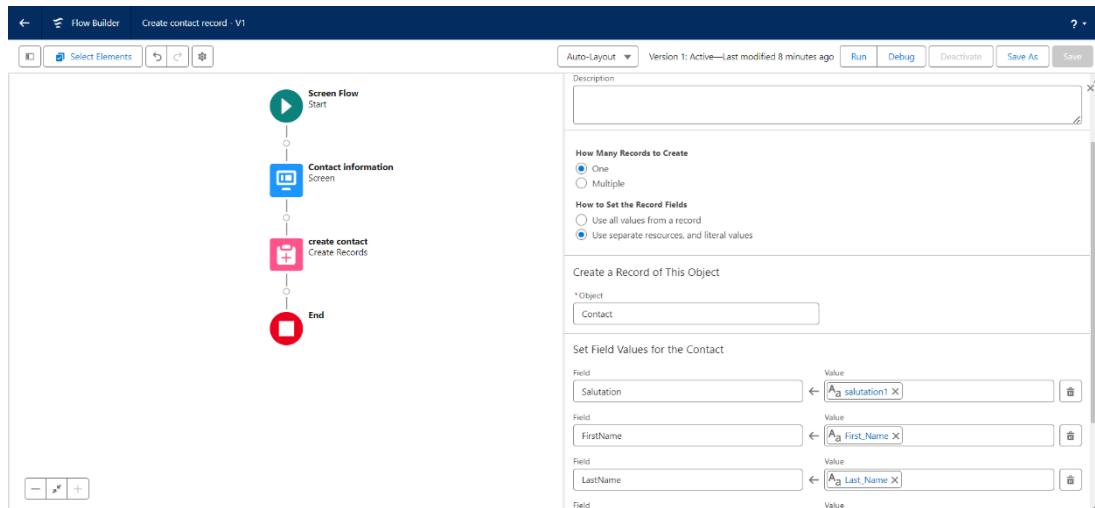
- In same way create mobile field and email field using text components and also includes header. And click on done



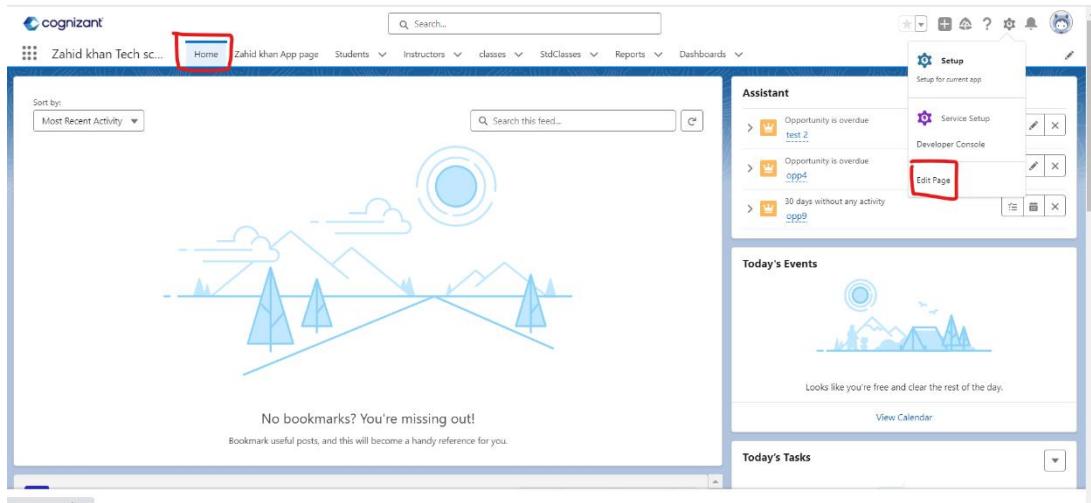
- After that save the flow and debug the flow.
- Now we have to create record from Data.



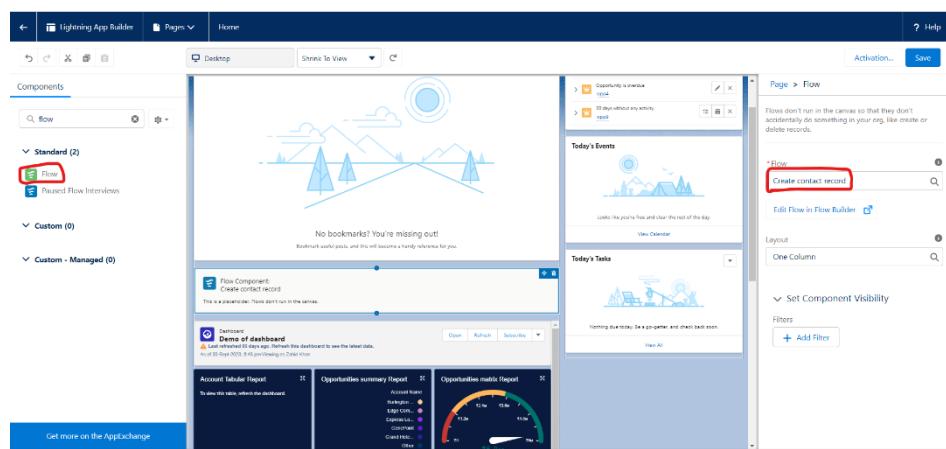
- After that enter the label name and api name.
- How Many Records to Create as **one**.
- How to Set the Record Fields as **Use separate resources, and literal values**.
- After that select the object as contact.
- After that enter the field and match with the value so that it takes data from user.



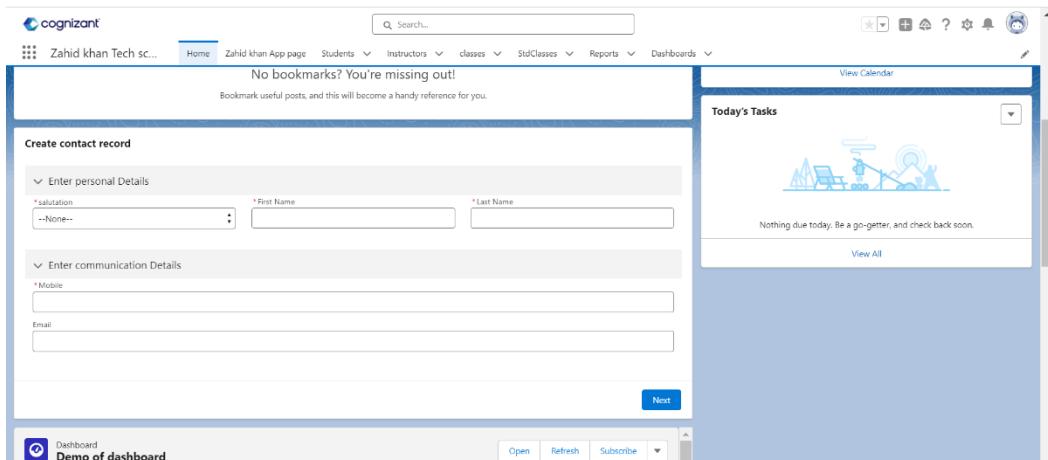
- Now we are going to place this contact flow in the home page of ‘zahid khan rech school’.
- Go to the ‘zahid khan tech school’ app page.
- After that click on home tab.
- After that click on set-up icon and click on edit page.



- After that search for flow in components and drag it in middle page.
- After that choose to **create a contact flow** which we created earlier. And click on save.



- After that go to home page and check the results.

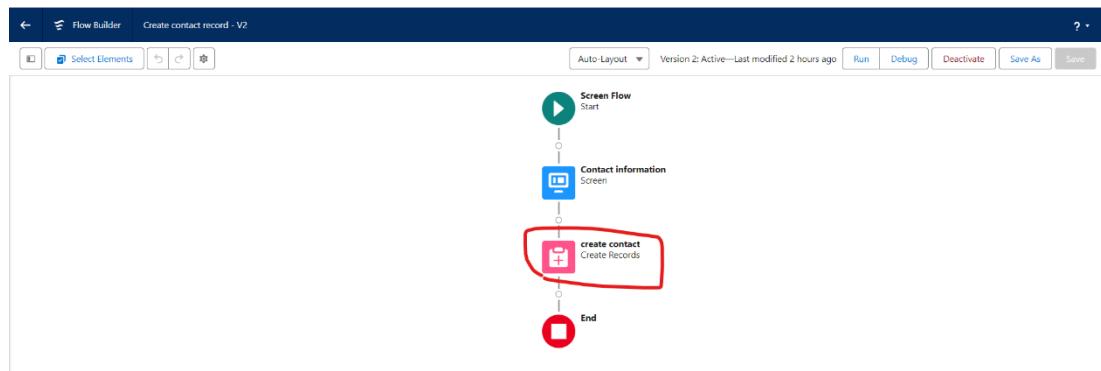


Option to attach a file and file should be related to contact:

- First, go to the contact flow which we created earlier.

The screenshot shows the Salesforce Setup interface with the 'Flows' tab selected. The 'All Flows' section displays a list of various flows. One flow, 'Create contact record', is highlighted with a red box. The list includes other flows like 'CMS: Submit Content for Review', 'CMS: Withdraw Review Request', 'Create a Case', and 'Create opportunity on account'. Each flow entry shows its label, type, status, package state, and last modified date.

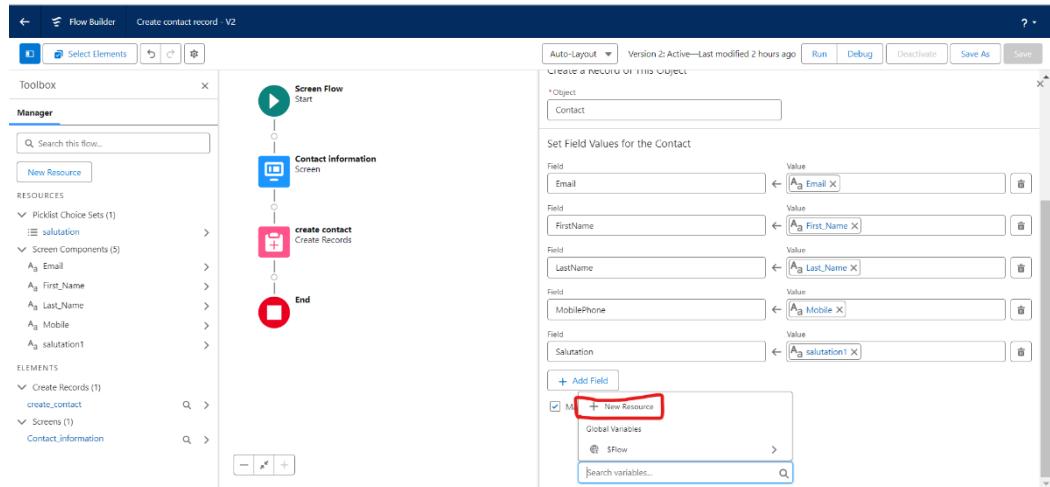
- Before creating file upload, we have to give a **related record id**. For this open the create record screen by double clicking on it.



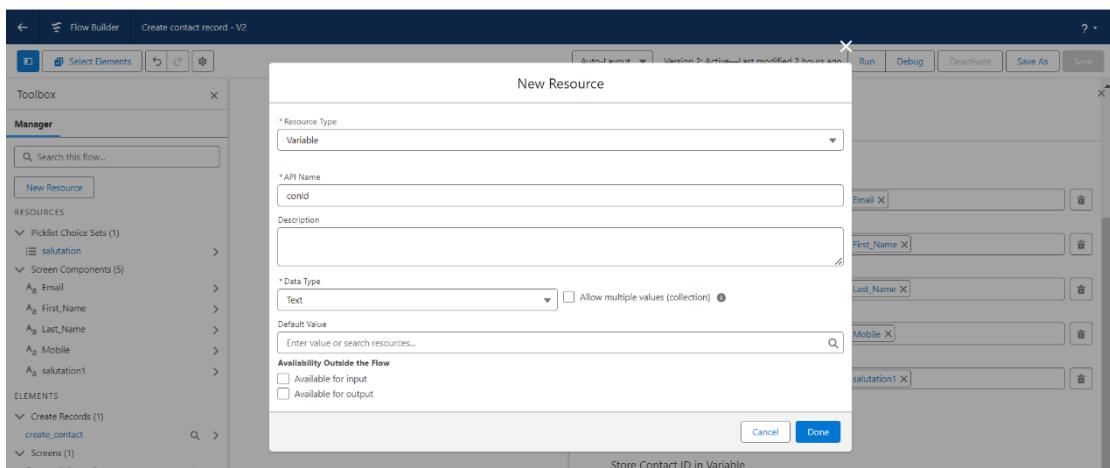
- After that check the box **Manually assign variable**.

The screenshot shows the Flow Builder interface with the 'Create contact record - V2' flow. The 'create contact Create Records' step is expanded, showing the 'To Set the Record Fields' configuration. It specifies 'Object: Contact'. Under 'Set Field Values for the Contact', there are five fields: Email, FirstName, LastName, MobilePhone, and Salutation, each with a value assigned from a variable. At the bottom of the configuration pane, there is a checkbox labeled 'Manually assign variables' (highlighted with a red box).

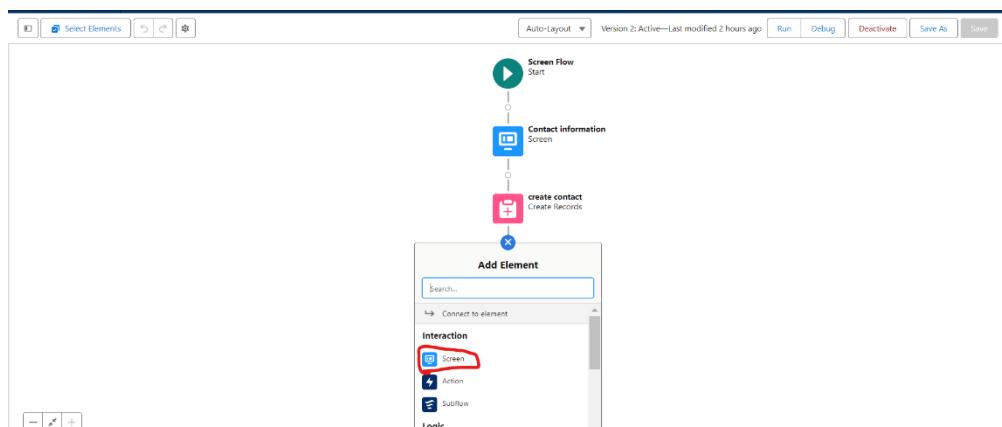
- Now we got a search box of Store Contact ID in Variable.
- After that we have to create new variable for this click on **+new resources** option.



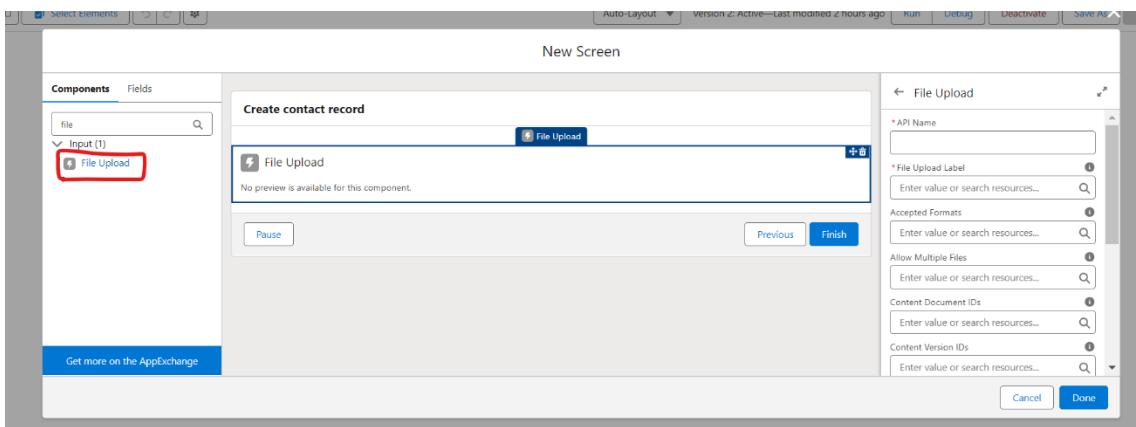
- Select **variable** from resources. After that enter api name, data type text and click on done.



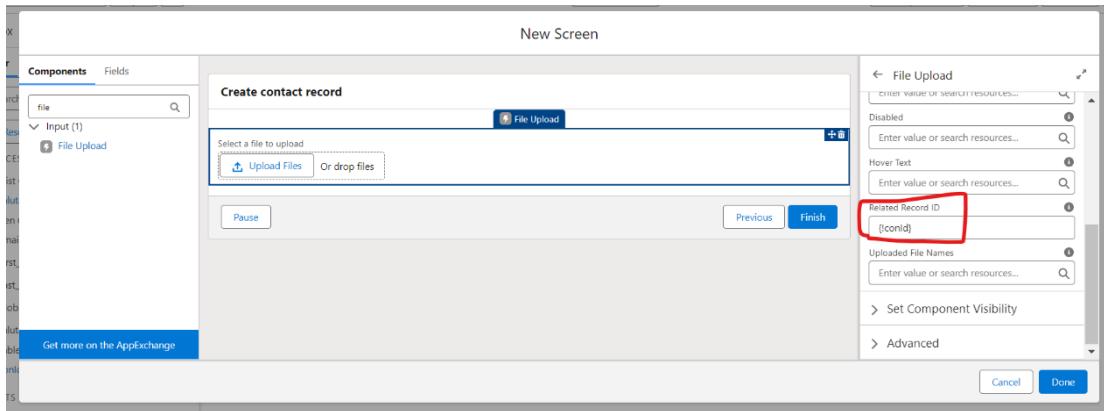
- Here what happen that if any contact record is created new created record id store in these variables (comId).
- After the create record add click on + icon to add one screen.



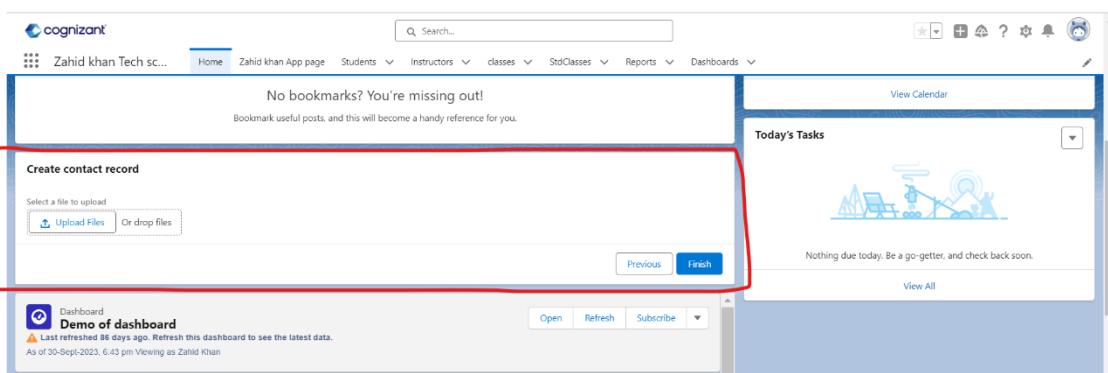
- After that enter the screen label and api name.
- After that search for **file upload** in components and place it in middle scree.



- After that enter the API name and file upload label of the file upload, now we have to provide related record id. add that variable **conId** which we created earlier.



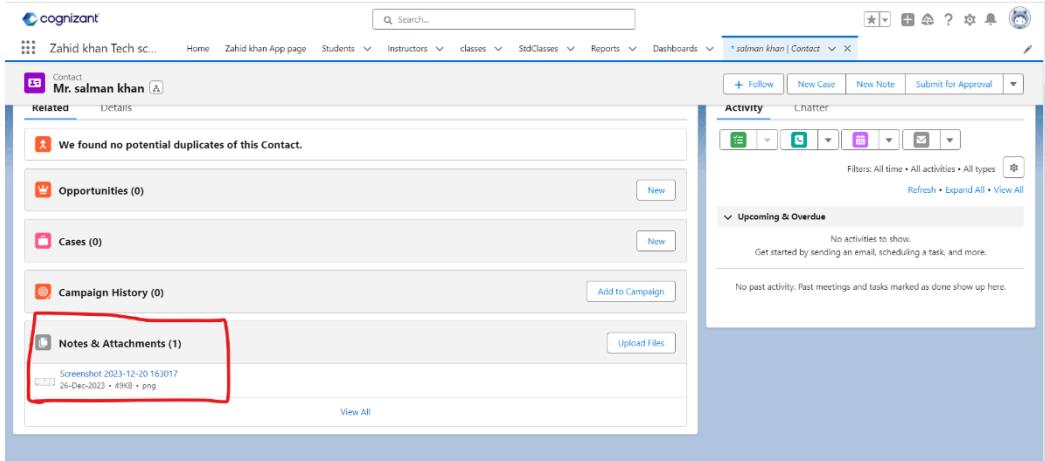
- After that click on done.
- After that save as the flow and click on activate.
- Now go to 'Zahid khan tech school' home page and check that file is uploaded or



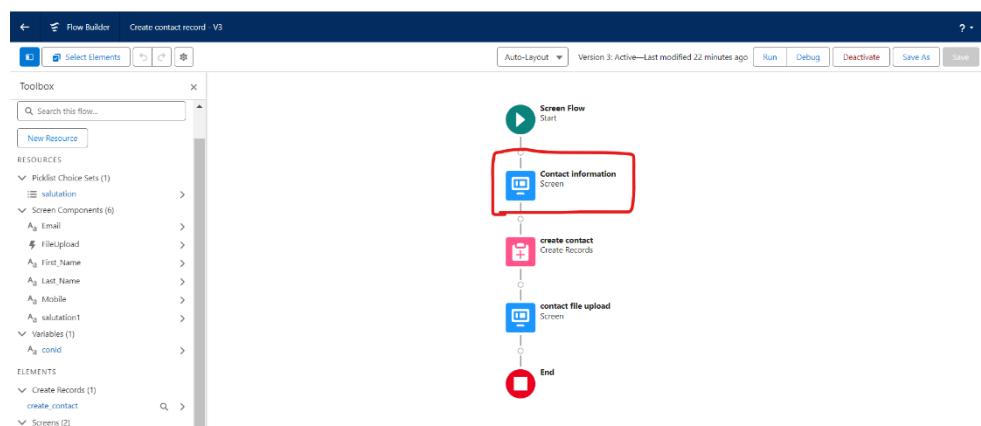
not.

- Here we can see that we have an option to upload a file related to contact. Upload any file.

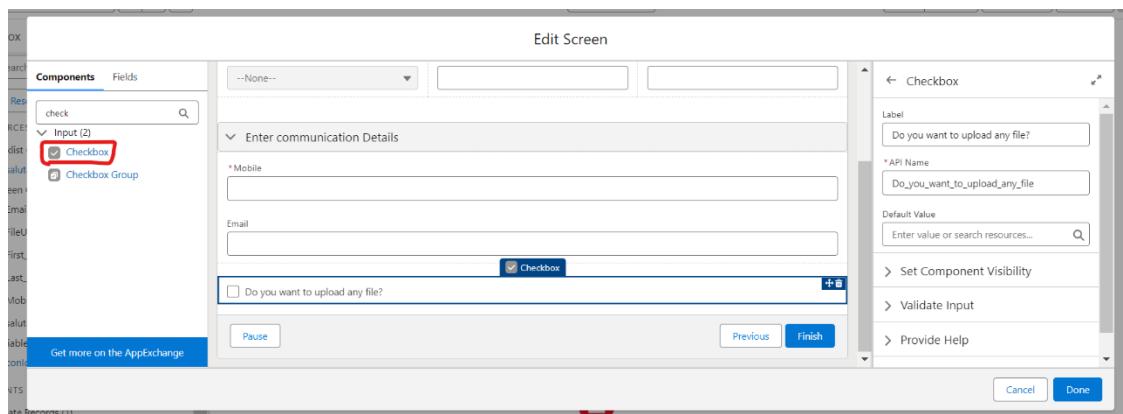
- After that go to contact object page in sales application and see that this file will be uploaded.



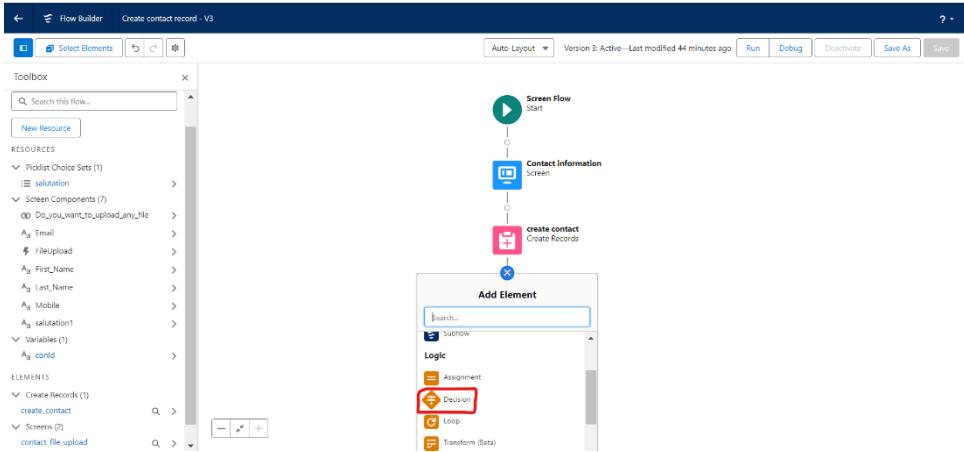
- We can also add check box in our 1st screen. If user want to upload file then they have to check the box after that they can upload file if unchecked then upload file steps will skips.
- To create this, we have to make decision logic.
- Open the contact flow which we created and double click 1st screen.



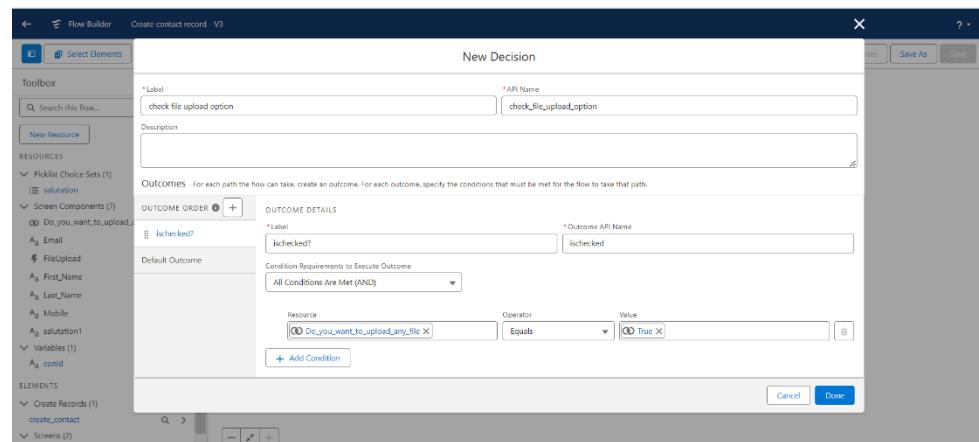
- After entering the label and API name. and click on done.



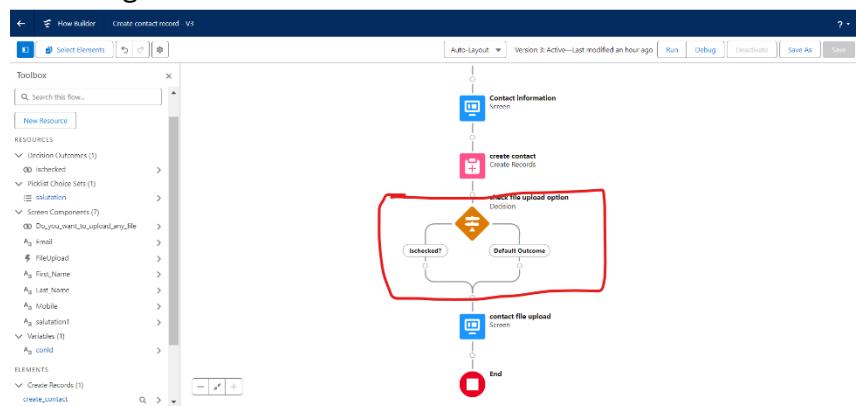
- After that click on + icon below create records and select decisions under logic.



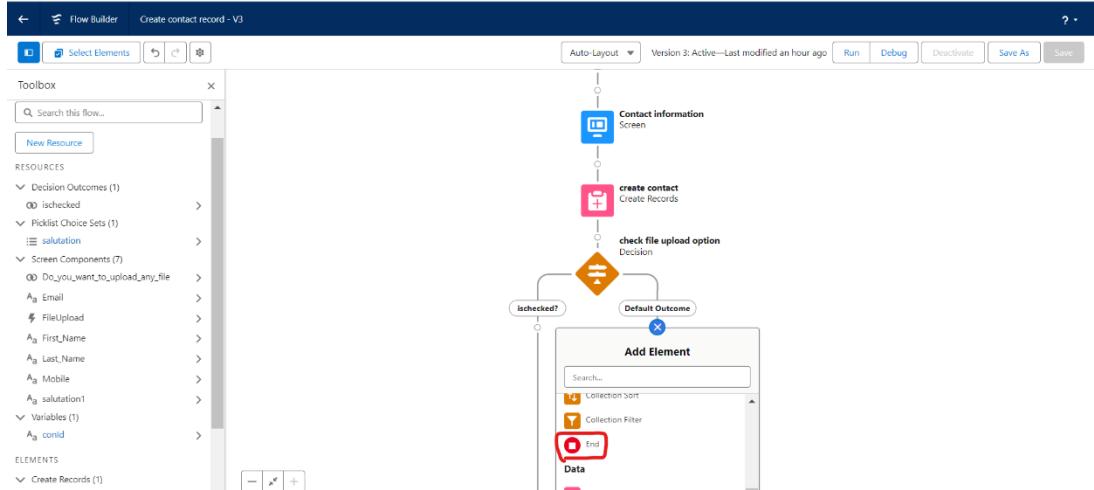
- After that enter the label and API name.
- After that enter the label and API name in outcome details.
- After that select resources that checkbox which we created in 1st screen. And operator is equals and value is true. Its means if the check box checked this decision will trigger otherwise default outcome will come. If we want to add more outcome, they we can do that.



- after that click on done.
- After that we got a flow that ischecked and default outcome.



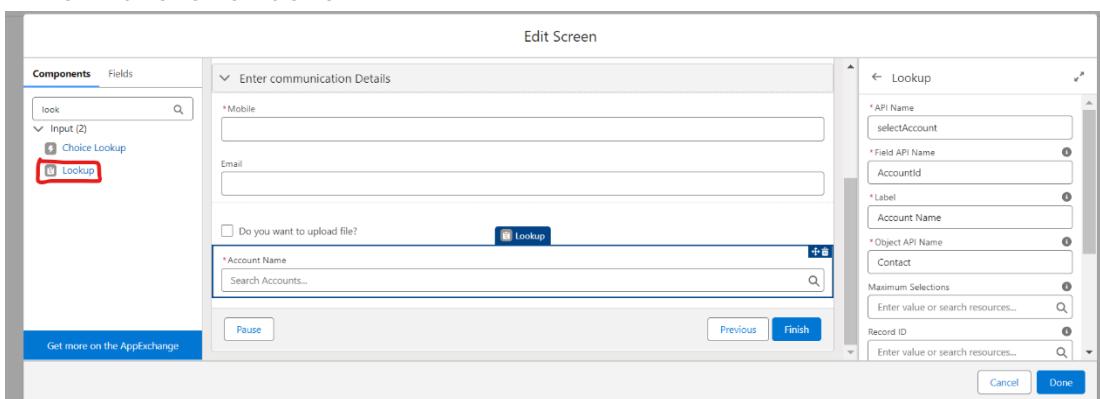
- Now we have 2 options if checkbox is checked so the file upload option will be visible. If not checked so the flow will end.
- To end the flow just click on the + sign below default outcome and select End



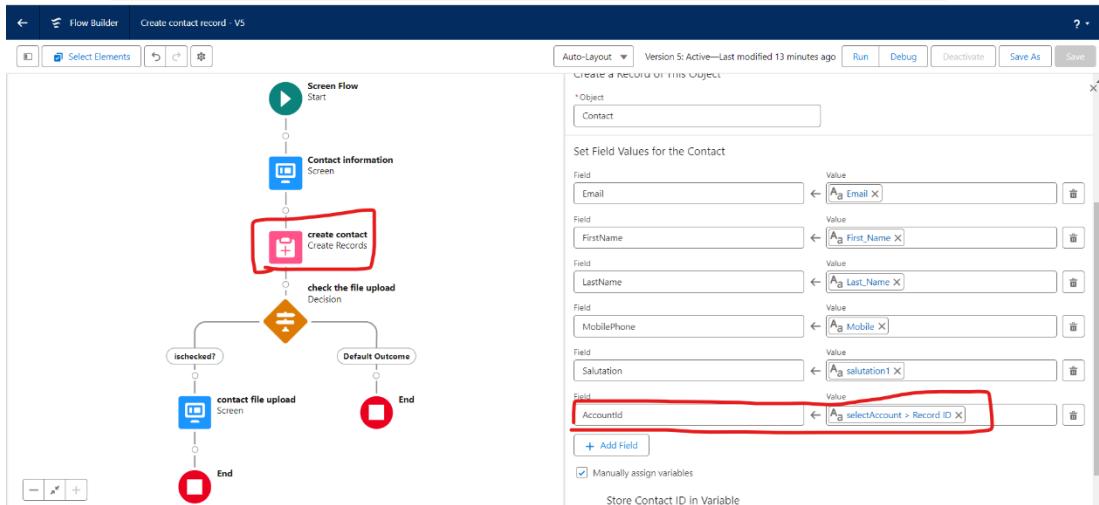
from logic.

Option to link contact with an account [lookup flow]:

- To link the contact with an account record, double click on 1st screen and add **lookup** components.
- After that enter label, API name, field API name and object API name.
- Field API name should be the source API same here, we enter AccountId.
- Now we have to enter the object API name which should be source object here, contact.
- After that click on done.

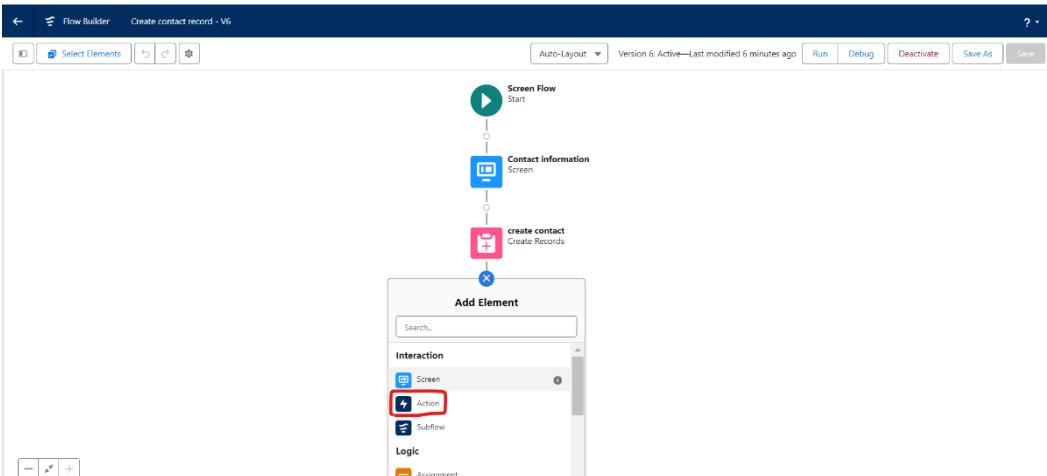


- After that double click on create record screen to add the account lookup.
- Now save as the flow and activate it.
- Now go to the home page of 'zahid khan tech school' and check the result.

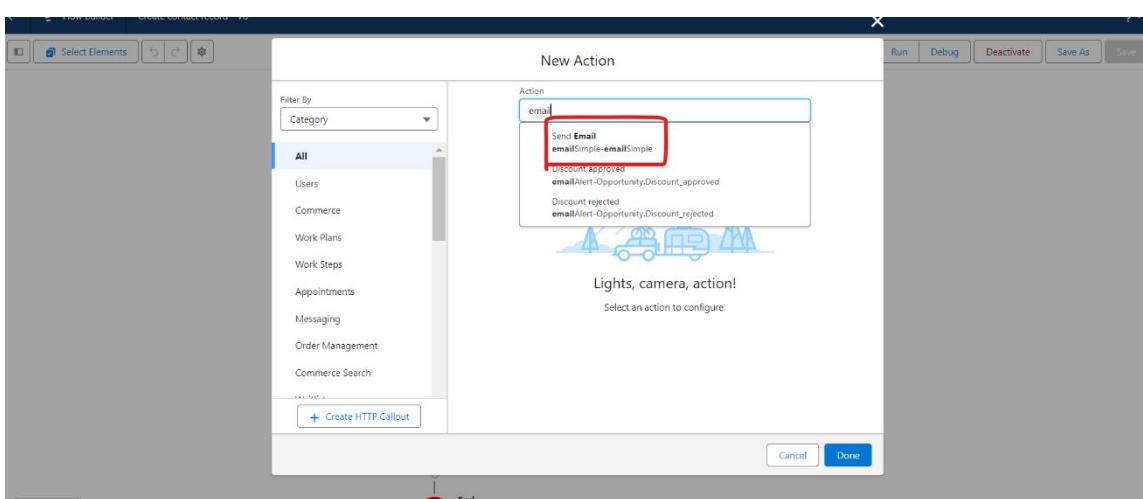


Upon contact creation send an email.

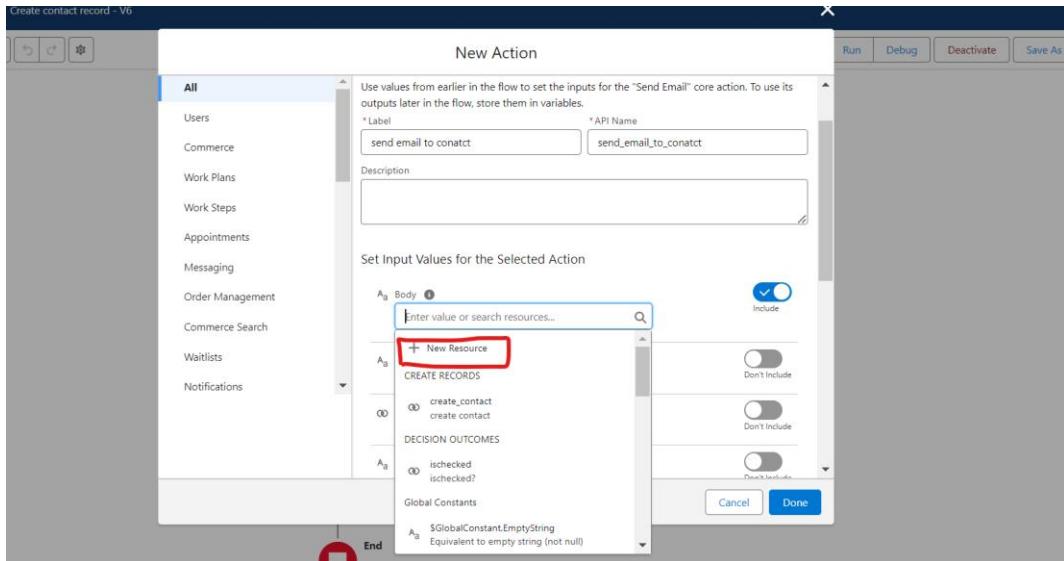
- After the **create record** click on + icon and choose action from interaction.



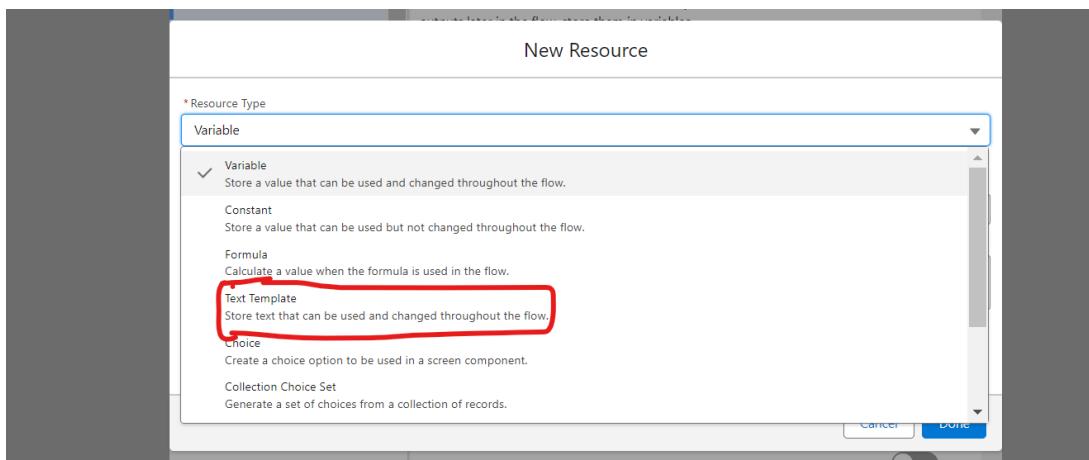
- After that search **email** in action and choose to send email options.



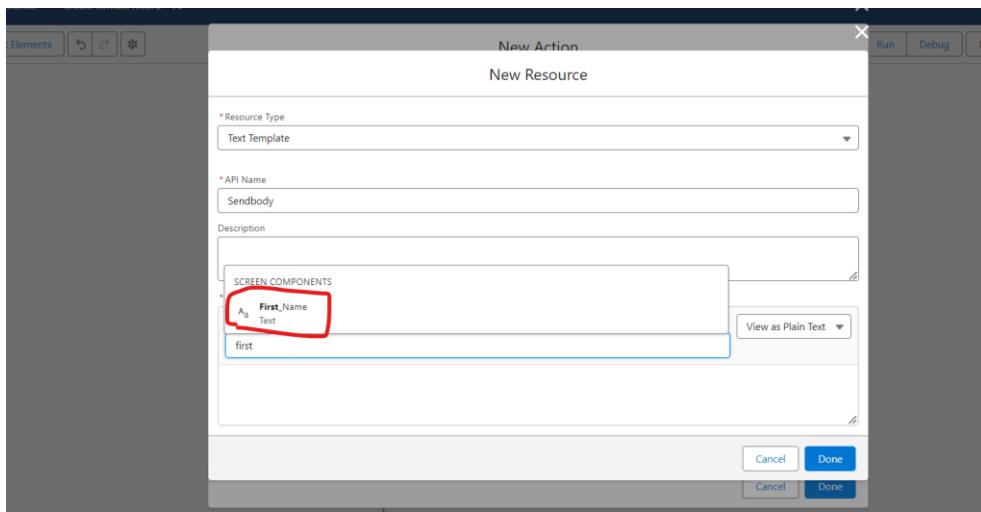
- After that enter label and API name. and we have to create body and subject for this on the toggle switch of the body and subject.
- After that inside body click on +new resource option and go to the resource page.



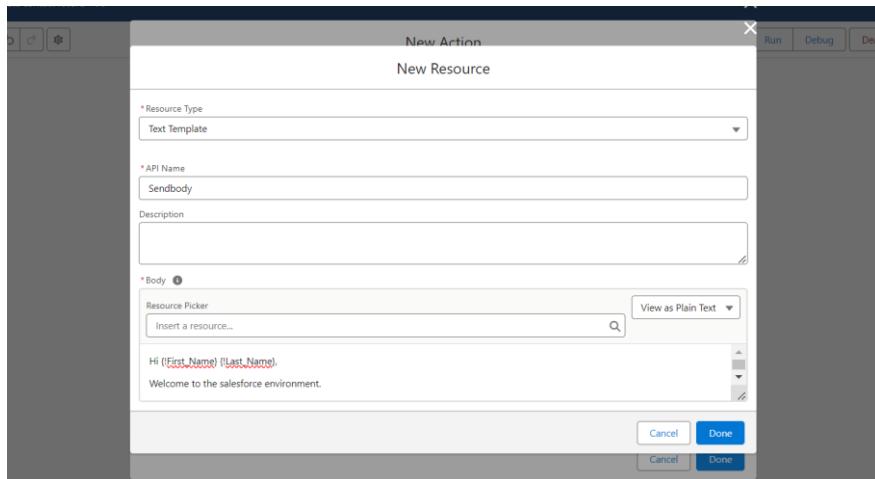
- Now select template text from drop down.



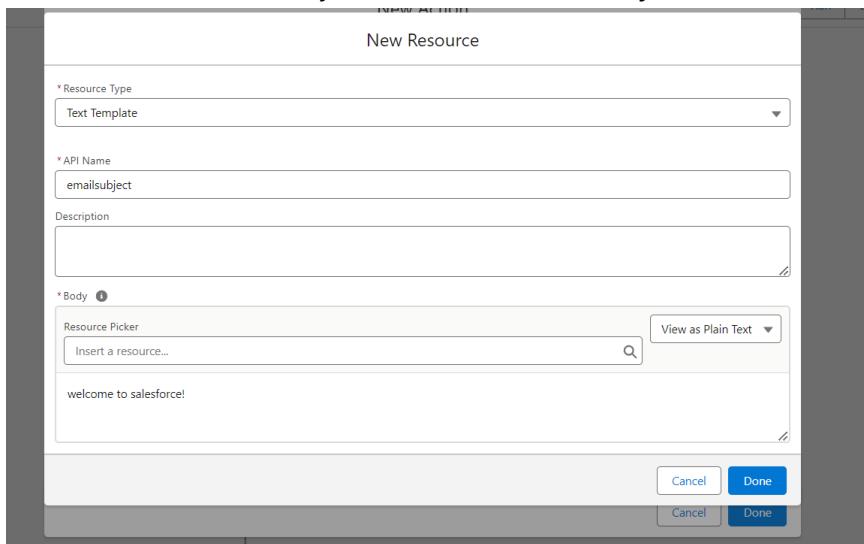
- After that API name and body. After that select view as plain text.
- After select the resources like first name or last name any resources which is created in the screen flow.



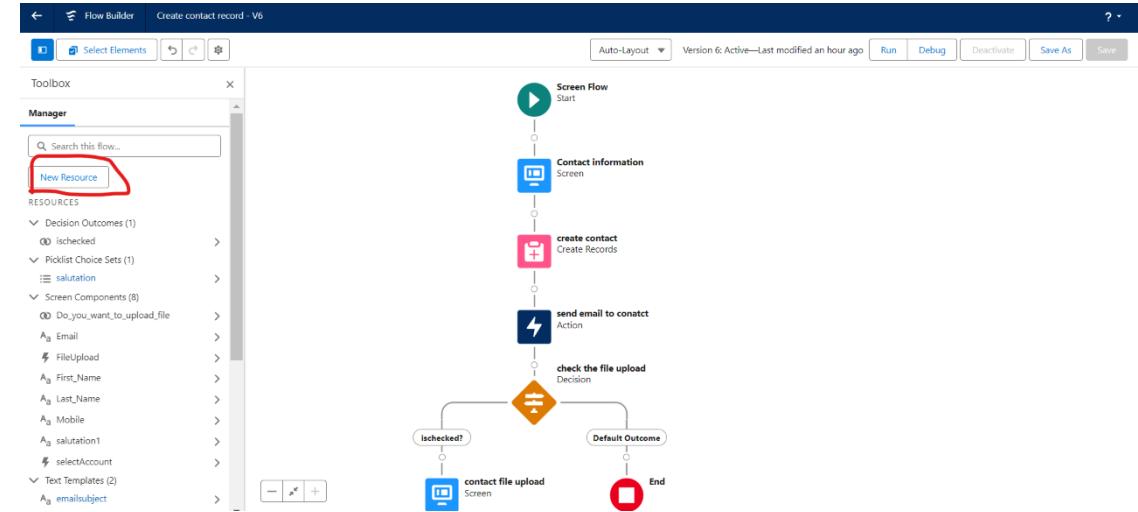
- After that enter the body of the email.



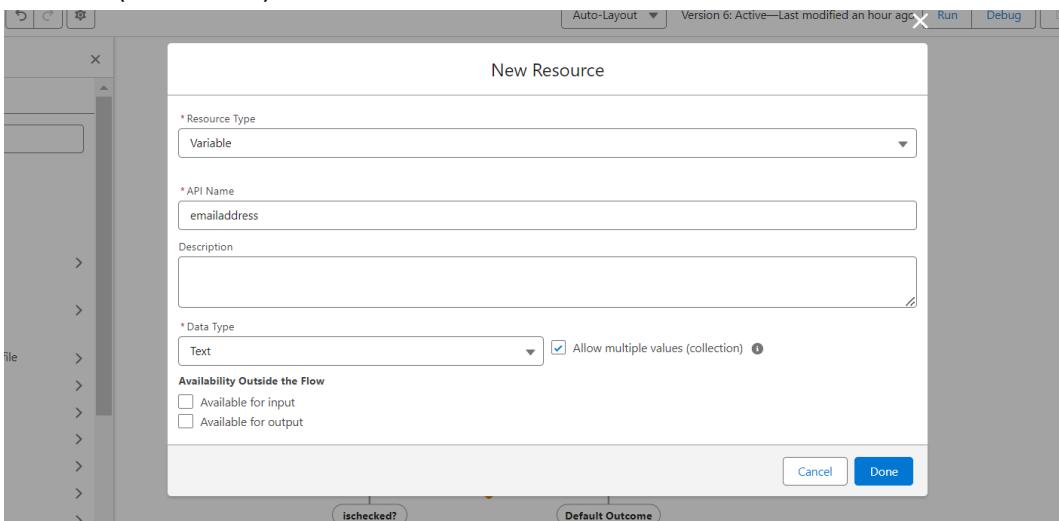
- After that click on done.
- Now we have to create a subject same as email body.



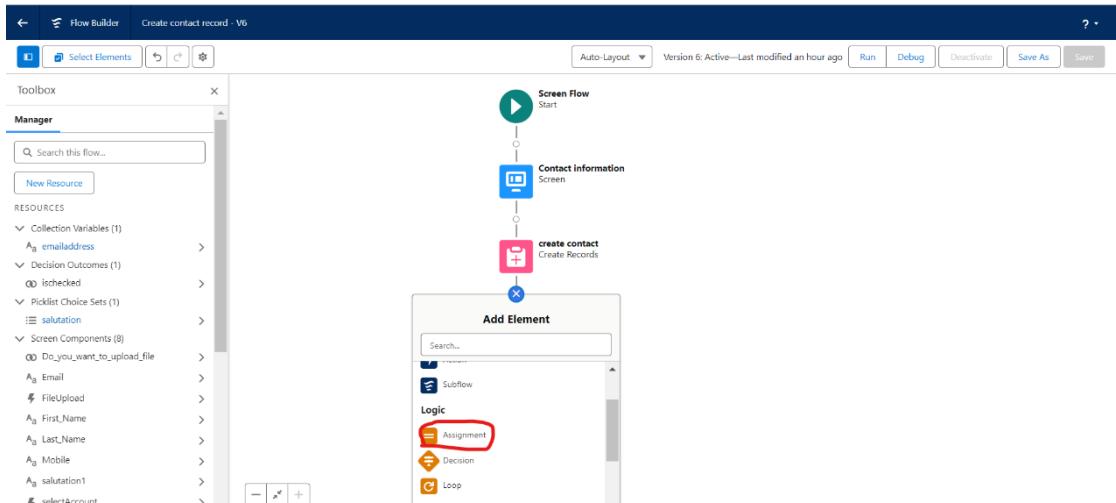
- After that click on done.
- Now we have to specify to whom we want to send email. Before specifying the email of collection, we have to make new resource of **variable**. For this click on new resource button on left hand side.



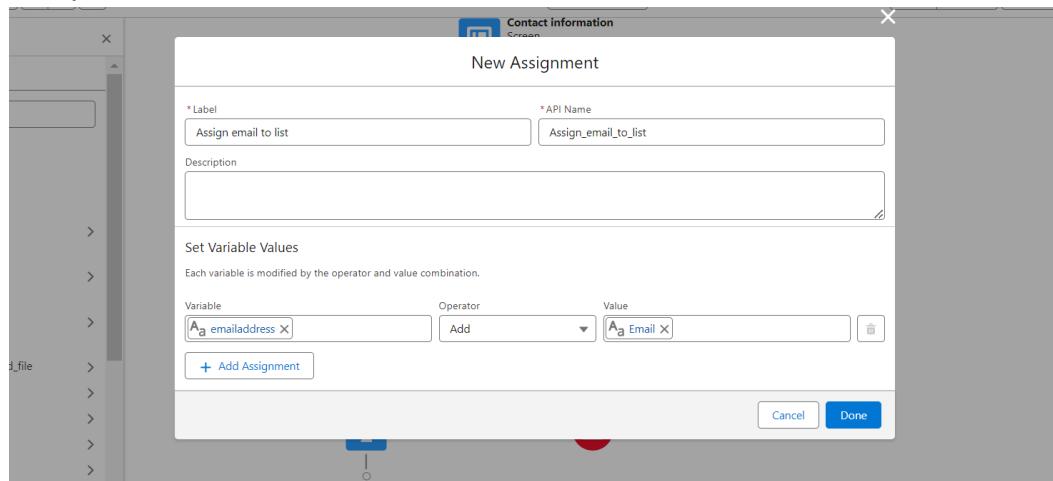
- After that's elect variables. Enter API name and check the box Allow multiple values(collection). And click on done.



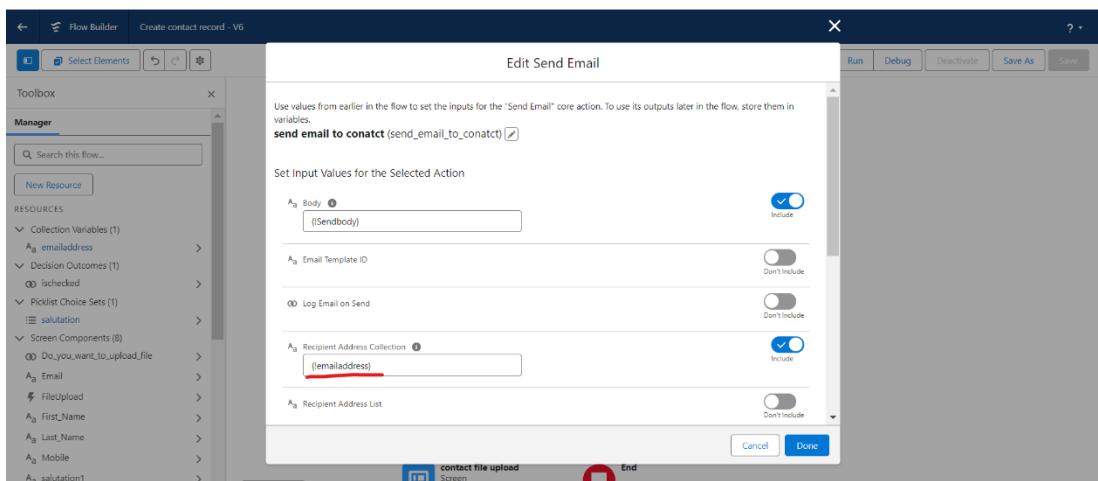
- after that we have to make an assignment to store the email address on email field in the emailaddress variable which we created earlier.
- For this click on + icon below **create record** and above email temaplates. And select **assignment** from logic.



- after that enter the label and API name. after that choose variable **emailaddress** in variable, in operator choose **Add** and in value choose email field from screen components. And click on done.



- After that double click on send email to contact action. Enable the recipient address collection. And select emailaddress variable inside the collection variable. And click on done.

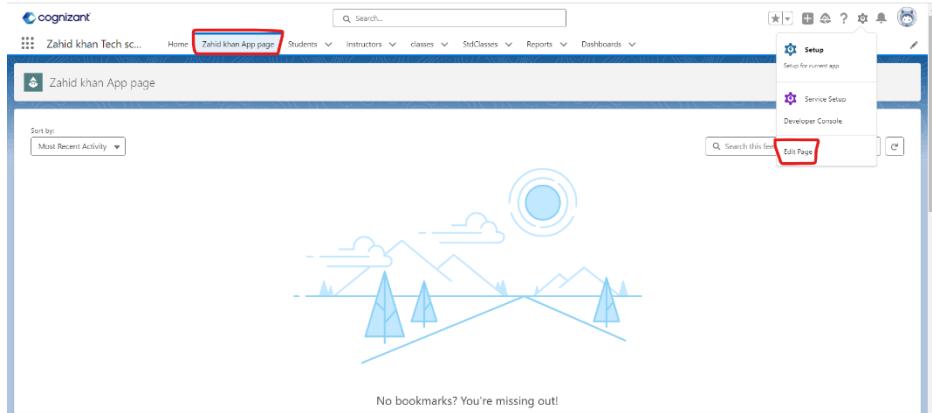


- After that click on save as and activate.

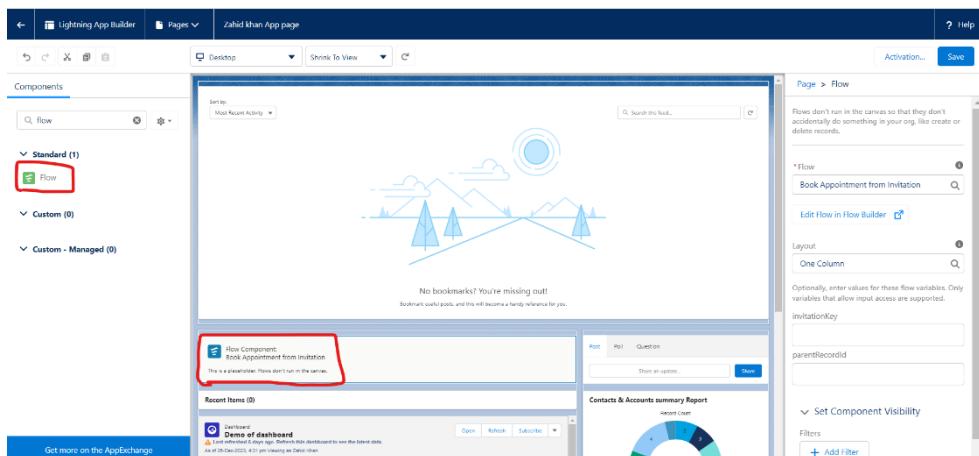
- Go to homepage and test the results.

Try to place flow on app page and test:

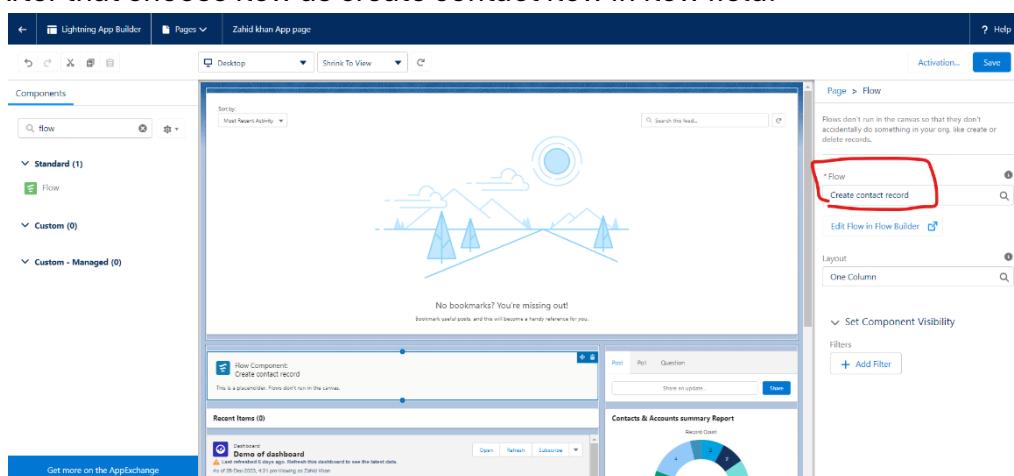
- First, open the app page which we created in ‘zahid khan tech school’ and click on se-up icon and click on edit page.



- After that search for flow in components and drag it into the middle screen.



- After that choose flow as create contact flow in flow field.



- After that click on done and see the results.

The screenshot shows a CRM interface. On the left, a red box highlights the 'Create contact record' form. This form contains sections for 'Enter personal Details' (with fields for Salutation, First Name, and Last Name) and 'Enter communication Details' (with fields for Mobile and Email). There's also a checkbox for uploading files and a search bar for account names. A 'Next' button is at the bottom right of the form. On the right, there's a summary report titled 'Contacts & Accounts summary Report'. It features a donut chart with the number 26 in the center, representing a total record count. The chart segments show various account names with their respective counts: Burlington Textiles Corp of America (4), Dickenson plc (2), Edge Cc (4), and others (2, 2, 1, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1).

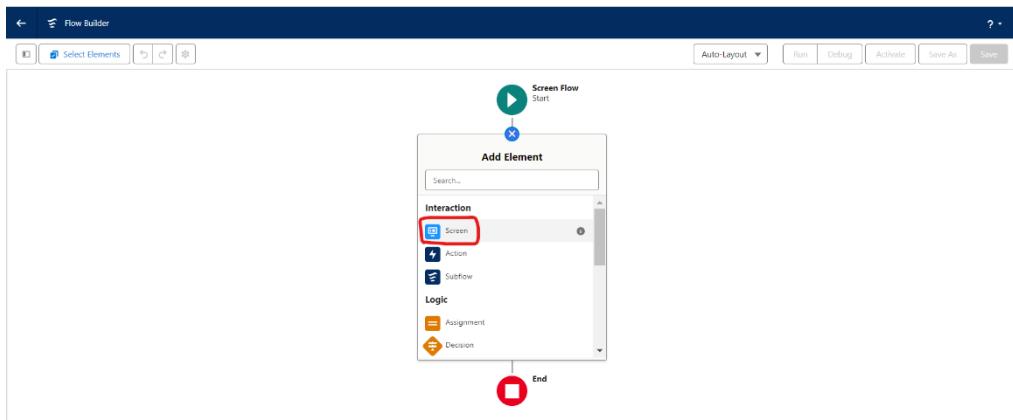
Day-25

Scenario

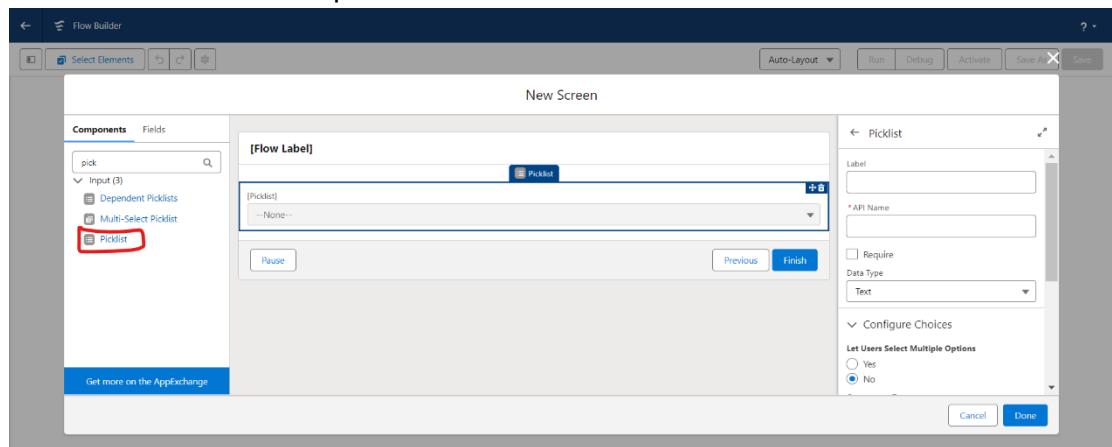
- Create a contact or an opportunity or case related to account based on users' choice through one flow.
- Create a contact or an opportunity or case related to account based on user choice through one sub flow.
- Modify above flow to ask how many contacts or opportunity or case should be created related to account based on user choice.

Create a contact or an opportunity or case related to account based on users through one flow:

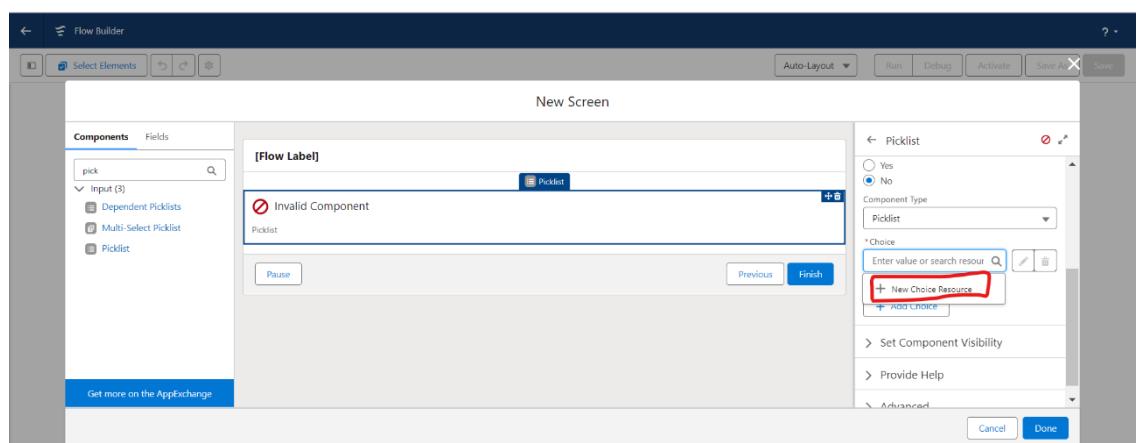
- Firstly, go to the set-up and search for flow in quick search and click on new flow.
- After that select screen flow and click on create.
- Before creating any flow, we have to ask or check that which record should be in flow.
- Now we have to create a picklist to select the that which flow wants to open. Here, we are going to make manual picklist value of three objects.
- For this, click on + icon below start and choose screen.



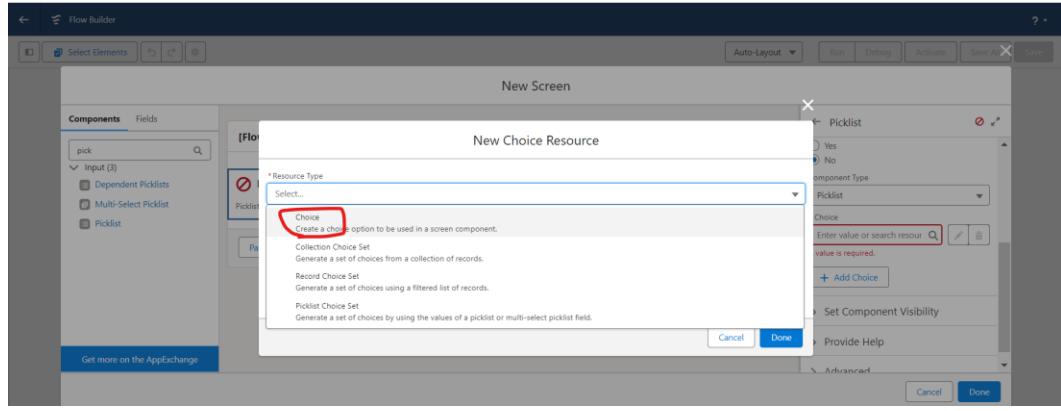
- After that search for picklist and insert it into the middle screen.



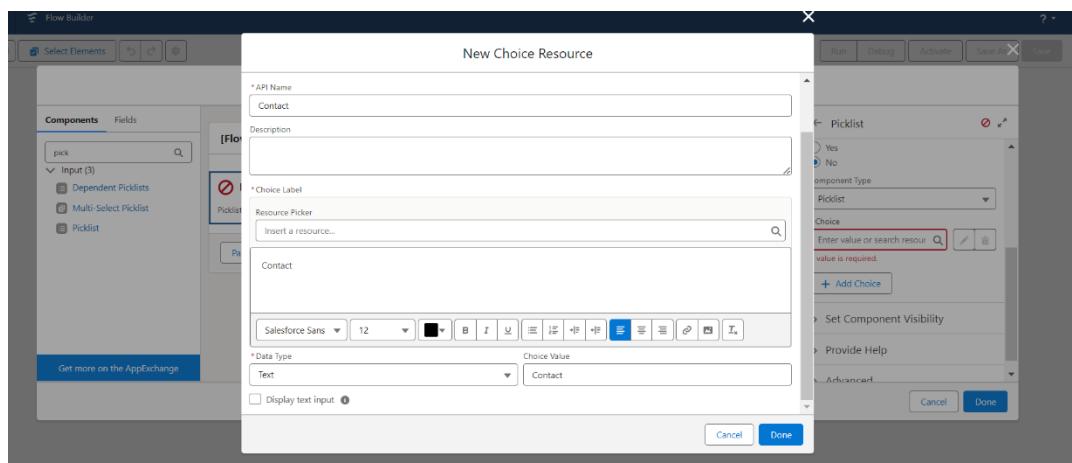
- After that enter the label and Api name. check the required box, data type text, and **let users select multiple options** should be no and components type should be picklist.
- After that we have to add options manually. Here, we are making picklist manually for contact, opportunity, and case so that user can choose any one.
- For this, click on **choice** field and click on **+ new resource choice**.



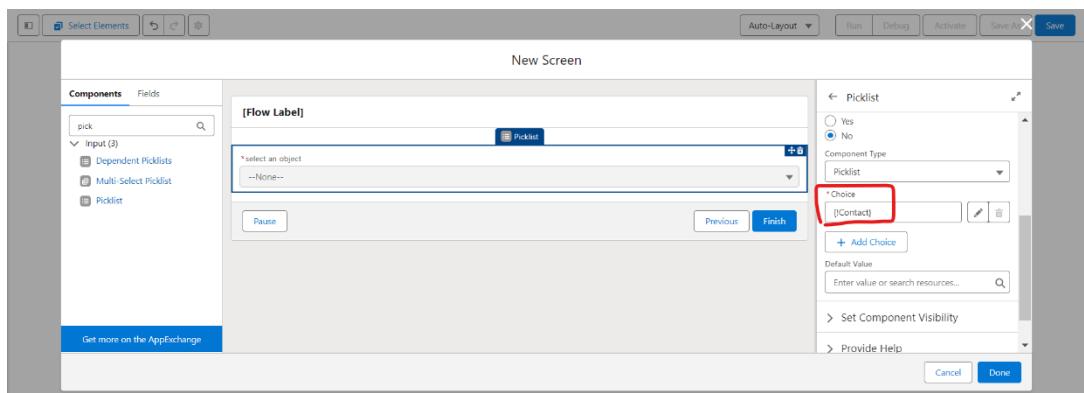
- After that select choice in resource types.



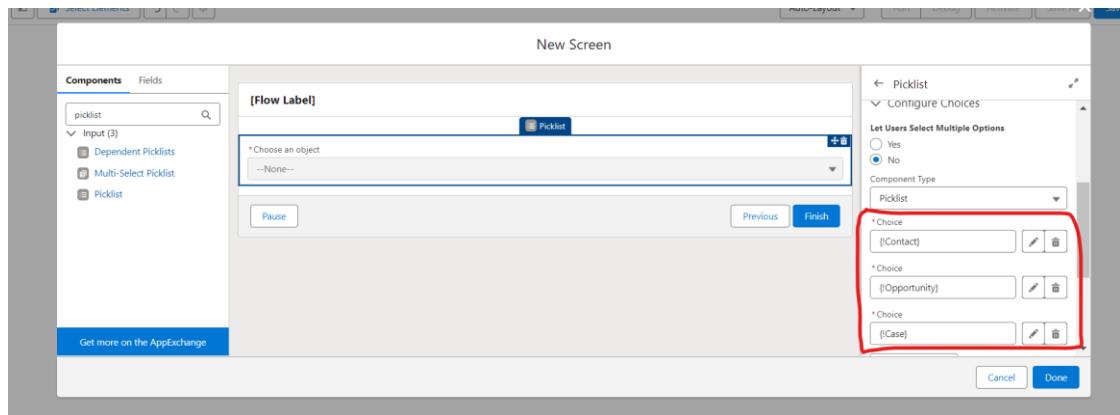
- After that enter the API name as contact. Choice label should be contact, data type should be text and choice value are contact and click on done.



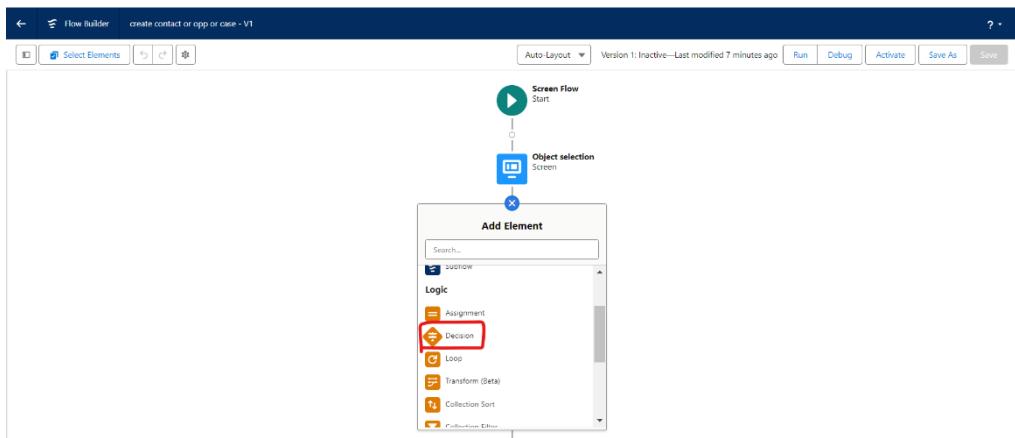
- Now contact choice is created.



- Same way we have to create picklist for account and case by clicking on Add Choice.
- After that click on done.
- After that save the flow and check in debug that all picklist value visible or not.



- Now we have 3 options in picklist contact, opportunity, and case. If user choose to contact so they can create contact if case so user can create case.
- To create this feature, we have to apply **decision** from **logic**.
- Just click on + below screen and select decision from logic.



- After that enter label and API name.
- After that enter the outcome details label and API name.
- 1st we have to create outcome for contact so enter label and API name should be isContact.
- After that enter the resource as **picklist** which we created earlier in screen. And in value select **contact** from choice.

New Decision

Label: check_user_selection **API Name:** check_user_selection

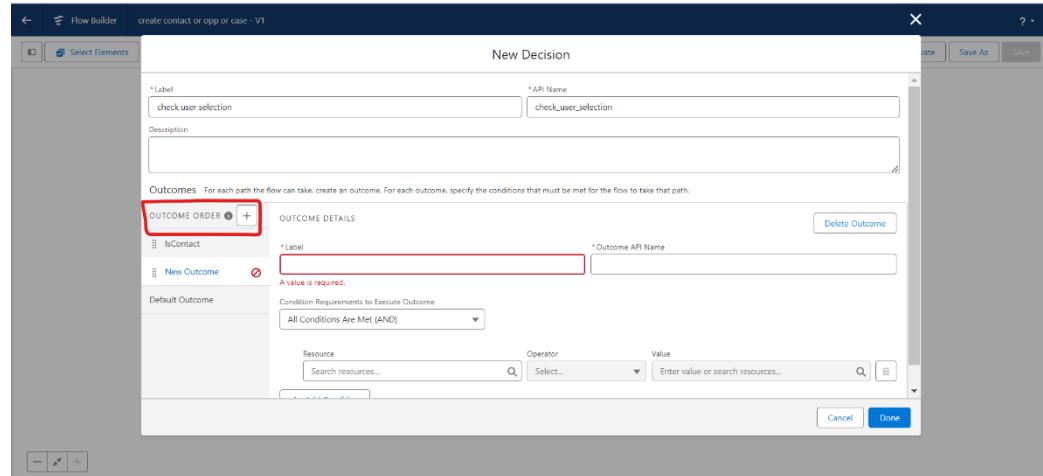
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER + **OUTCOME DETAILS**

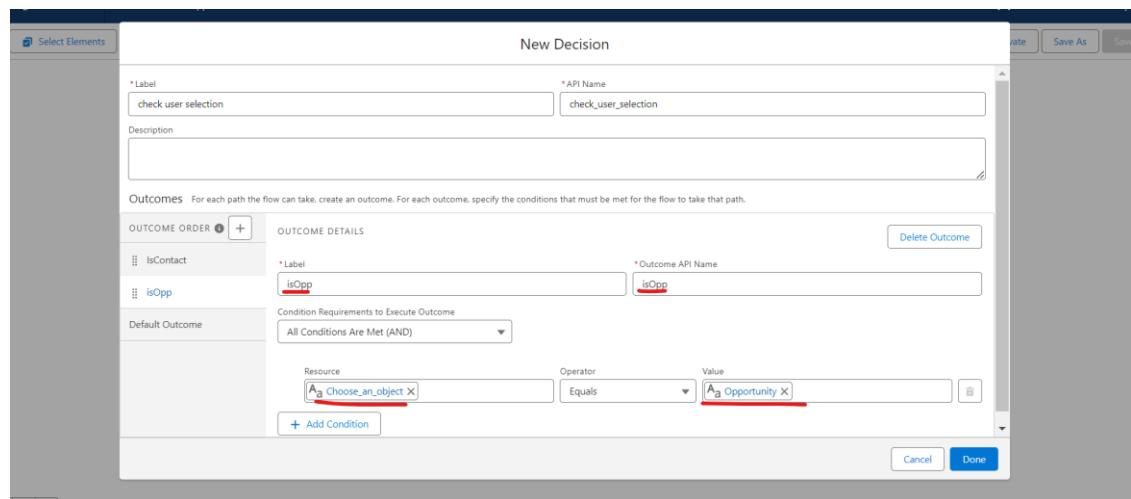
Label: IsContact	Outcome API Name: IsContact
Condition Requirements to Execute Outcome: All Conditions Are Met (AND)	
Resource: {A3 Choose_an_object}	Operator: Equals
Value: {A3 Contact}	

+ Add Condition

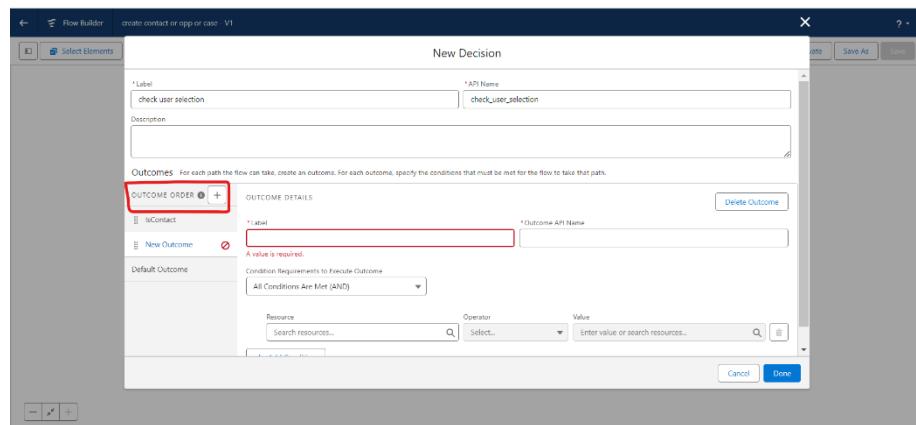
- In same way we have to create outcome for opportunities and case.
- For this click on **Outcome order +** and create outcome for opportunities.



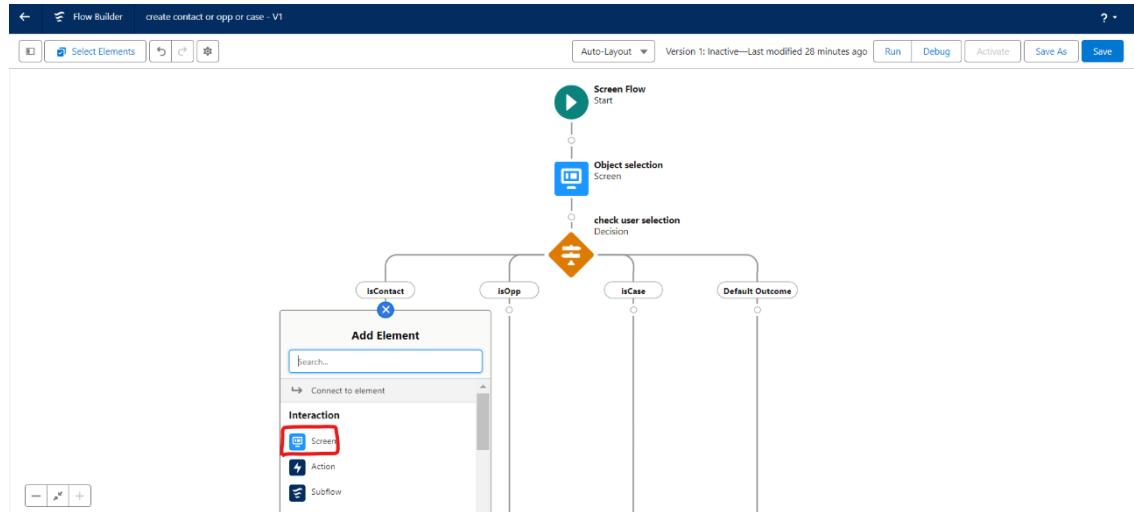
- Label and API name of the Outcome details should be **isOpp**. Resource should be picklist we created, and value is **opportunity** from choice.



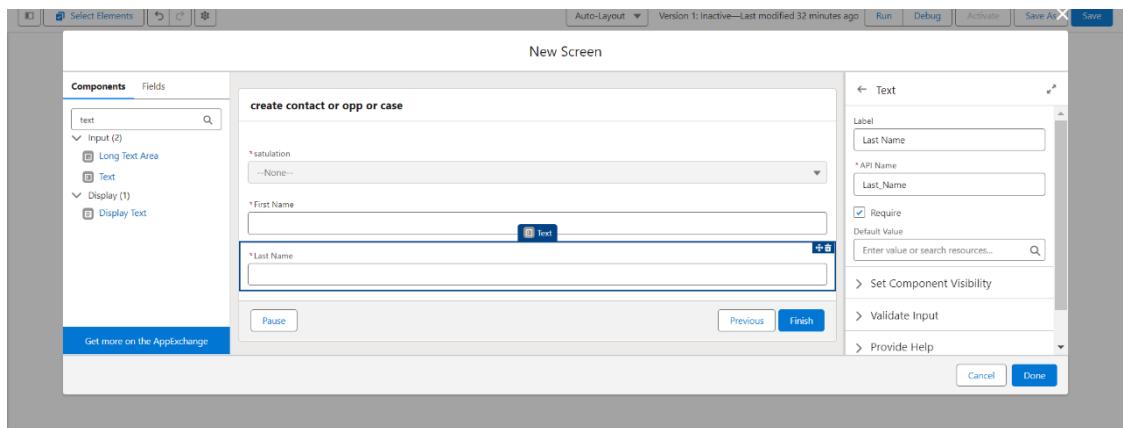
- In same way we have to create outcome for case.
- For this click on **Outcome order +** and create outcome for case.



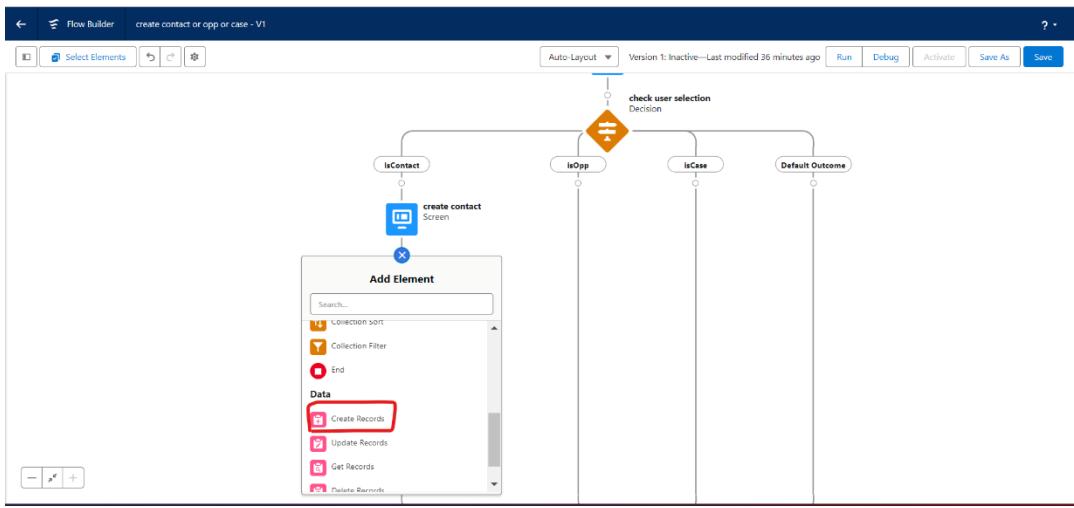
- Label and API name of the Outcome details should be **isCase**. Resource should be picklist we created, and value is **case** from choice.
- In this way we created 3 outcomes. Now click on done.
- Now click on + sign below the **isContact** so that we can create screen for create contact record if user select contact so contact record will open. Same way we have to create screen for **isOpp** and **isCase**.



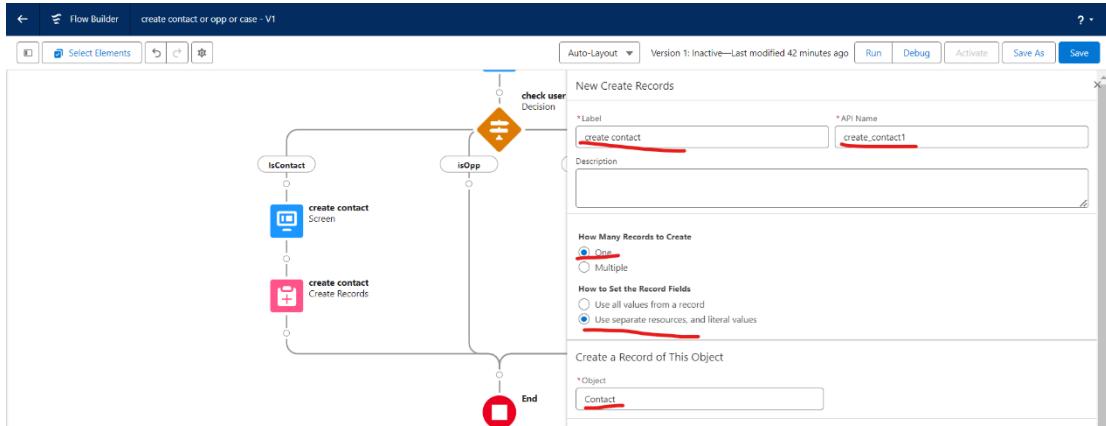
- Now we have to add field and mandatory field of contact. As we know that in contact last name is mandatory, so we are going to create First name, last name.



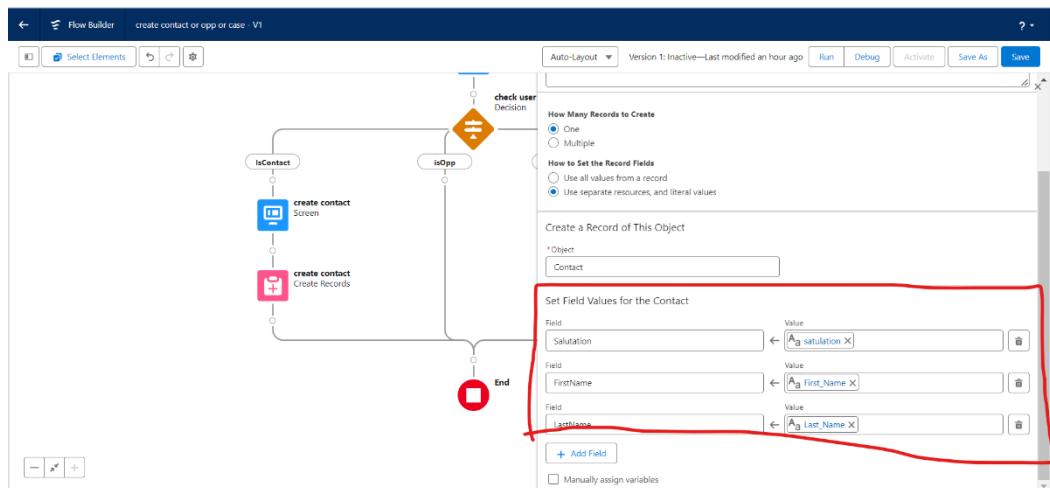
- click on done.
- After that click on + icon below screen to choose **create record** from Data.



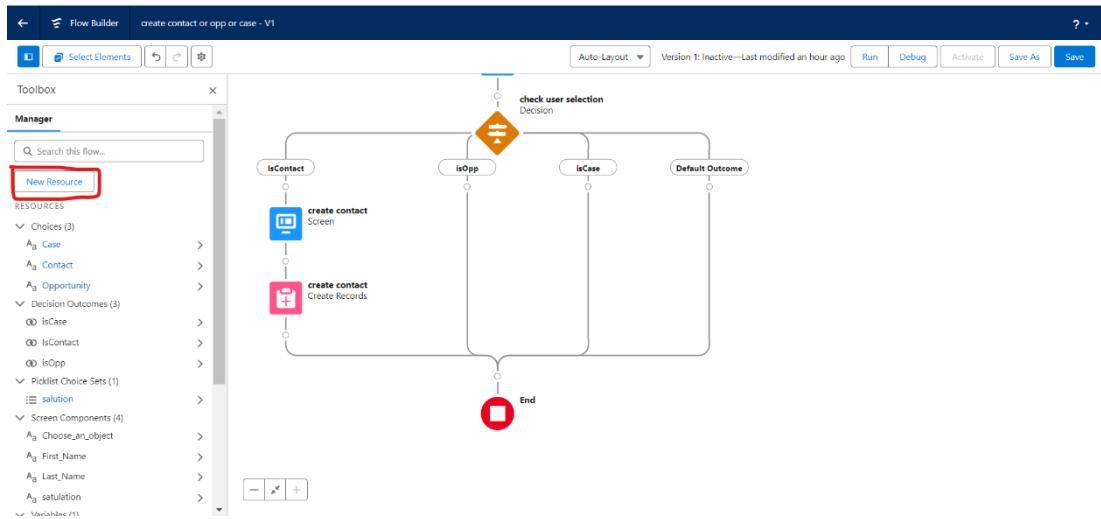
- Enter label and API name. **How Many Records to Create** is **one**, **How to Set the Record Fields** is Use separate resources, and literal values.
- After that select object as contact.



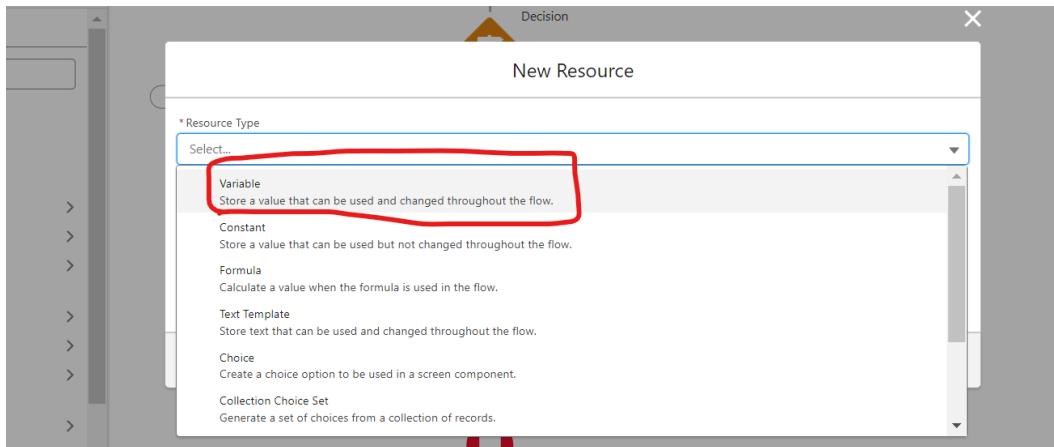
- After that enter **field** and **value**. Such as **salutation** to **salutation**, **First name** to **first name**, and **last name** to **last name**.



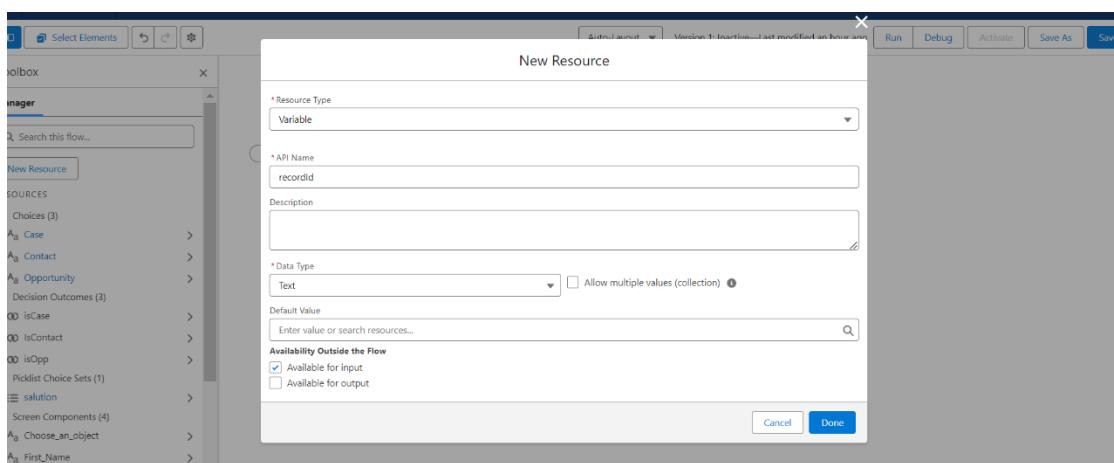
- now we have to relate this contact with account object. For this 1st create new resources as variable for **recordId**. For this click on **New Resources**.



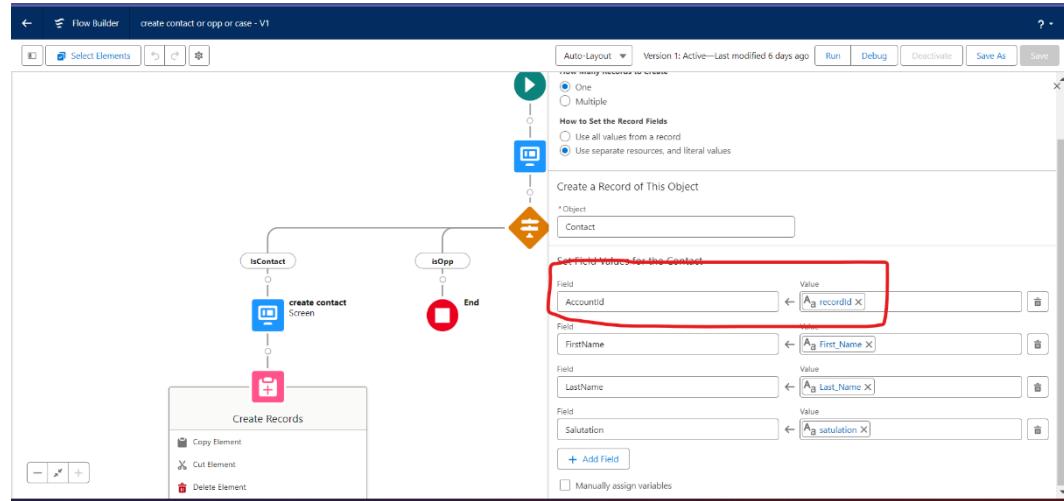
- After that select the variable from drop down.



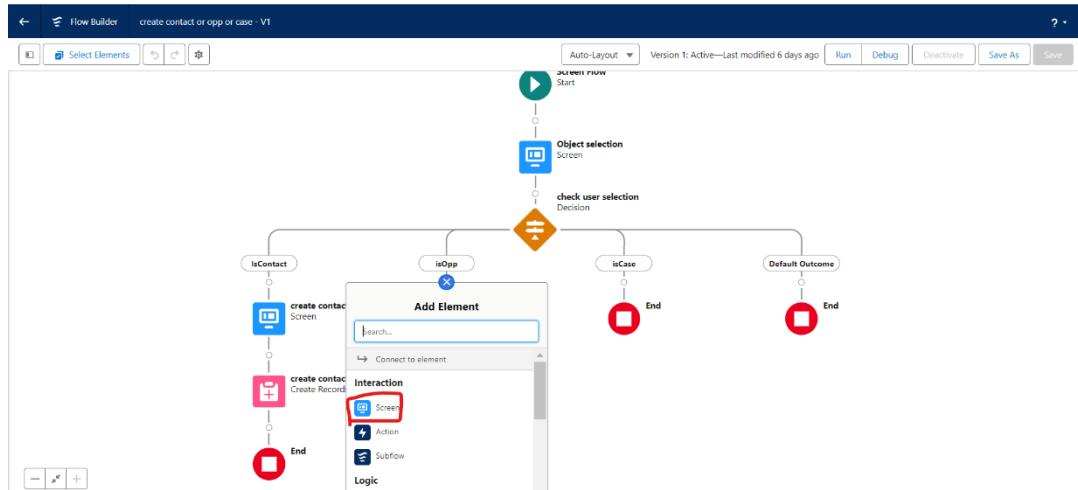
- After that enter label and API name. data type as **text**. **Availability Outside the Flow** as available for input.



- After that double click on **create record** of **isContact** to add the **recordId** variable. Field should be **AccountId** and value should be **recordId** which we created as variable.



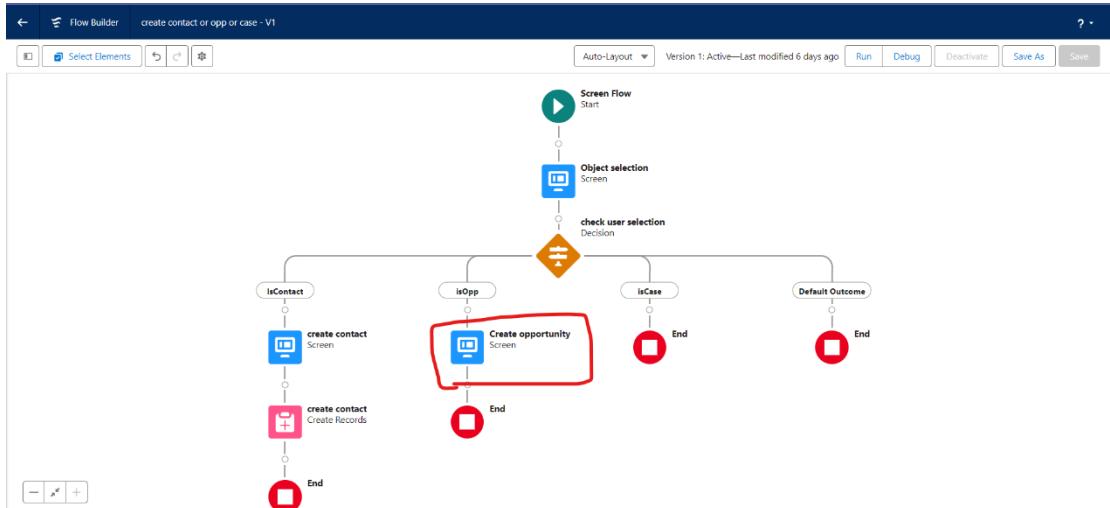
- In same way we have to create screen for opportunity object. For this click on + sign below **isOpp**.



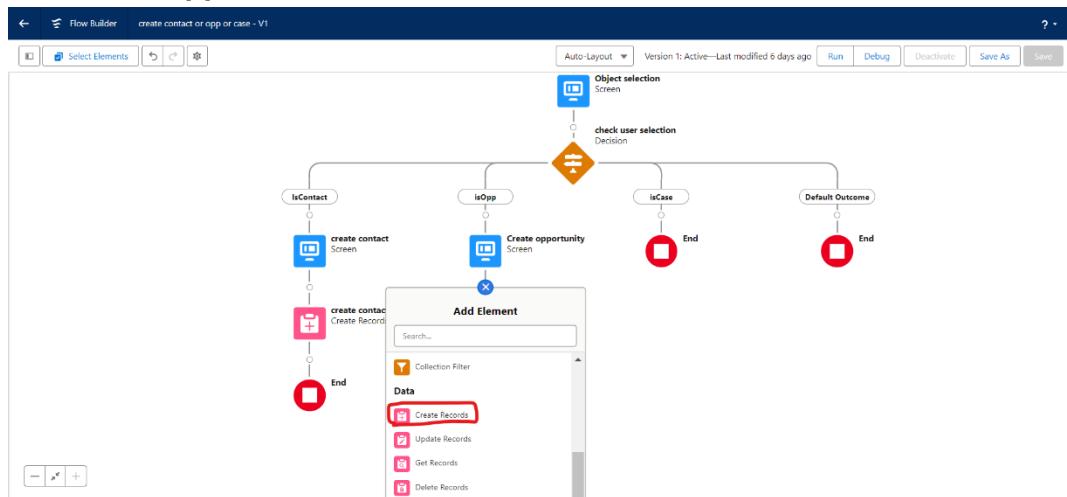
- After that create field of name, close date and stage (picklist choice set). And done.

The screenshot shows the 'New Screen' configuration page. The main area displays a form with fields: 'Name', 'close date', and 'Stage'. The 'Stage' field is a picklist component. The right sidebar shows the configuration for this picklist, including a 'Choice' field with the value '(select_stage)' and a 'Default Value' input field. Other configuration options like 'Set Component Visibility', 'Provide Help', and 'Advanced' are also visible.

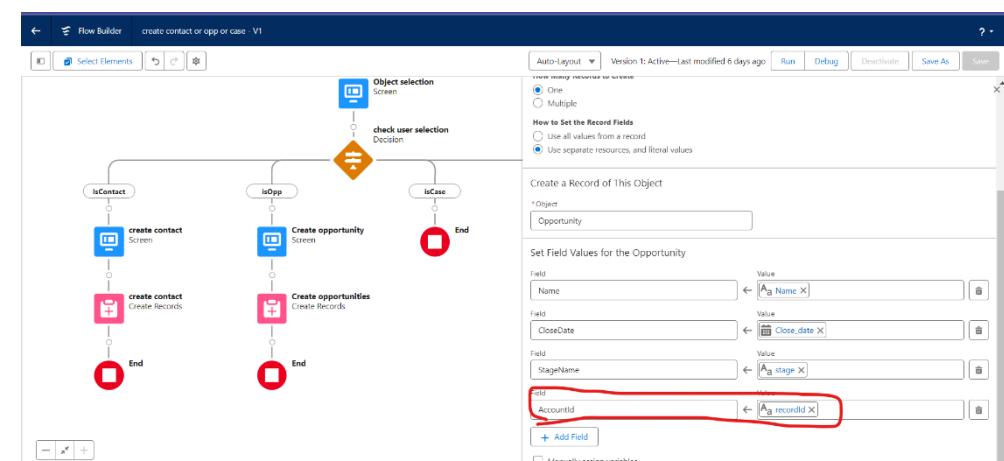
- This way opportunities screen is created.



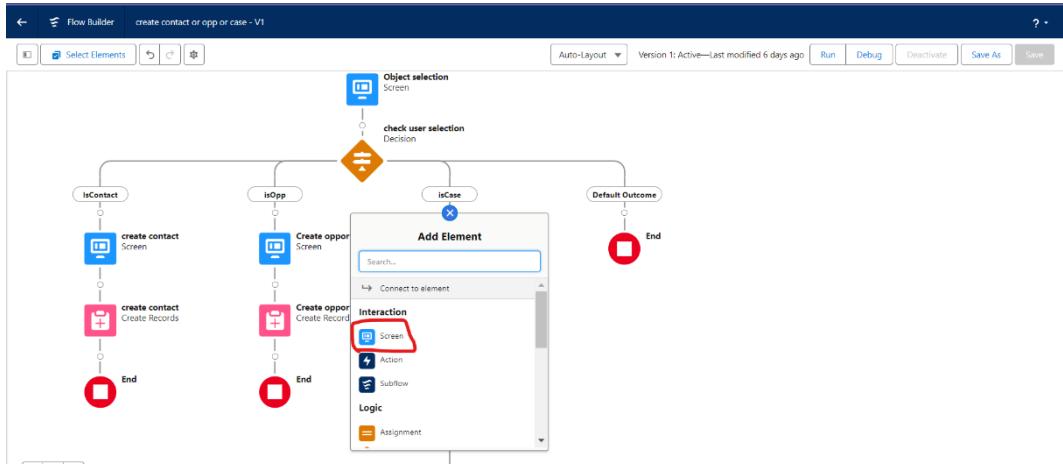
- Now we have to create record for opportunities. For this click + sign below screen in **isOpp** and **create record from data**.



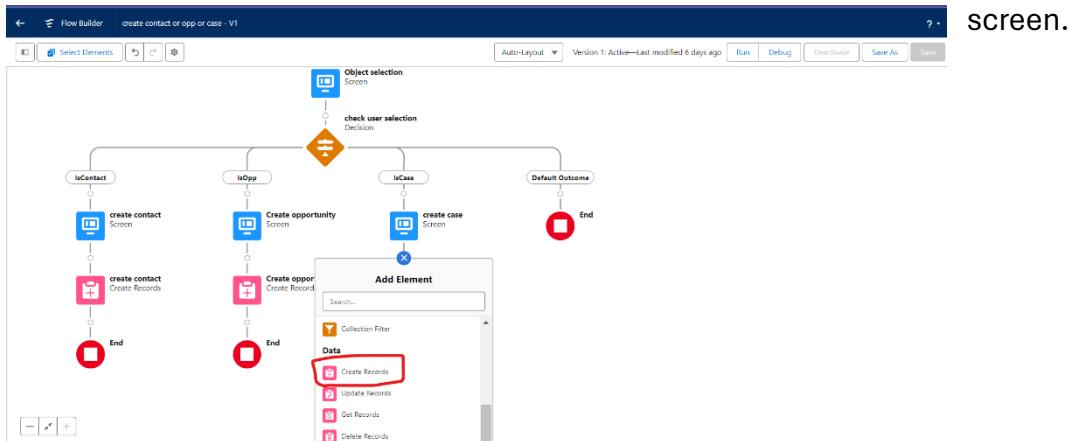
- Now map the field of opportunities object. And also add **accountId** with **recordId** variable to relate opportunities with account.



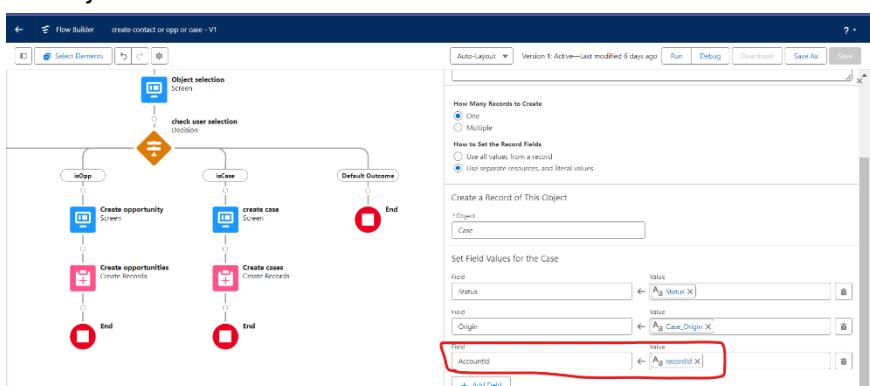
- Now opportunities object flow is created and relate with account.
- Now same way we are going to create screen for case object.
- For case object we are going to create screen for **status** and **case origin**, and both are picklist, so we need to create picklist choice set.
- Click on + icon in isCase and select screen.



- Create required field and click on done.
- After that make create screen. For this choose create screen from data below screen.

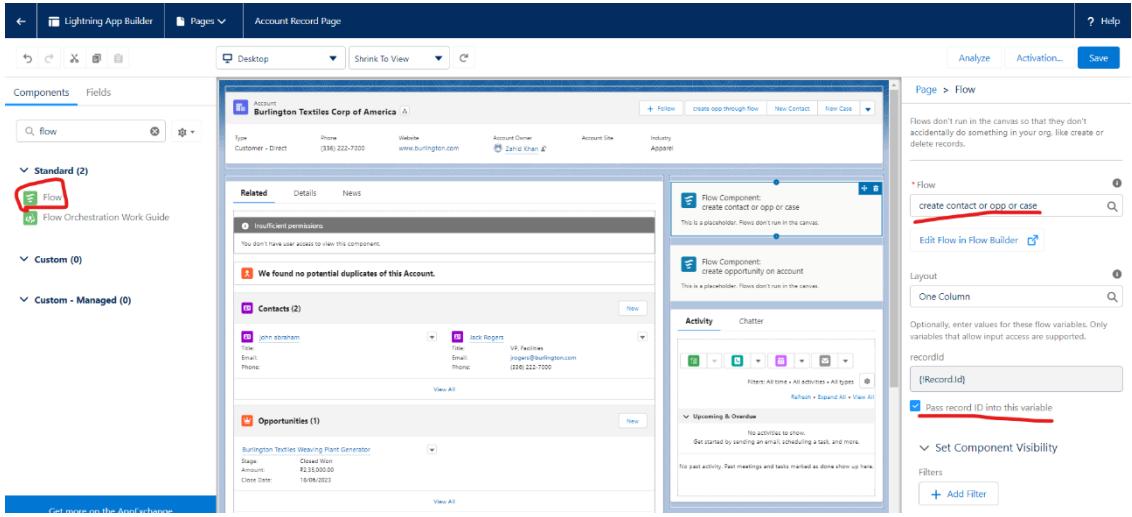


- Map field with the case object and also add accountId with recordId so that case with relate to account.



- Now all flow is created just click on save as and activate it.

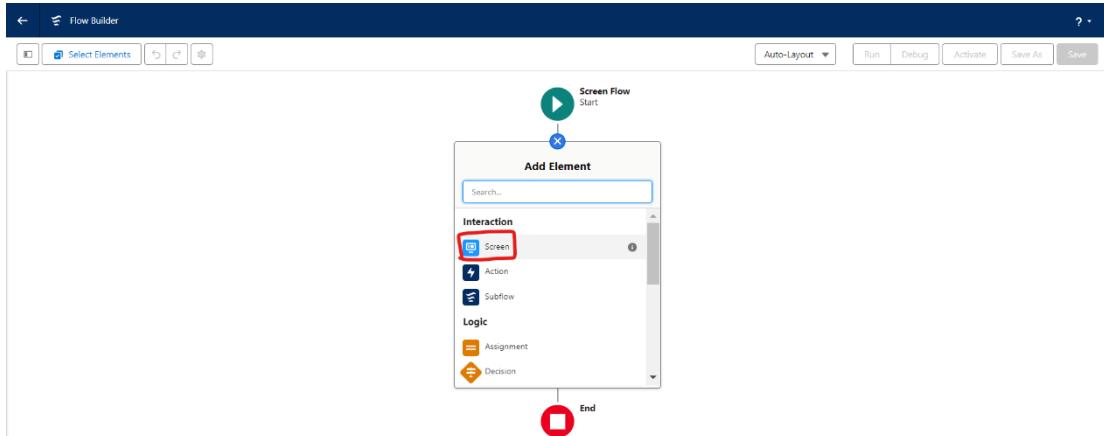
- After that we are going to add this flow inside account object records.
- Go to account and open any records. After that click on set-up icon then click on edit page option. And check the check box **Pass record ID into variable**.



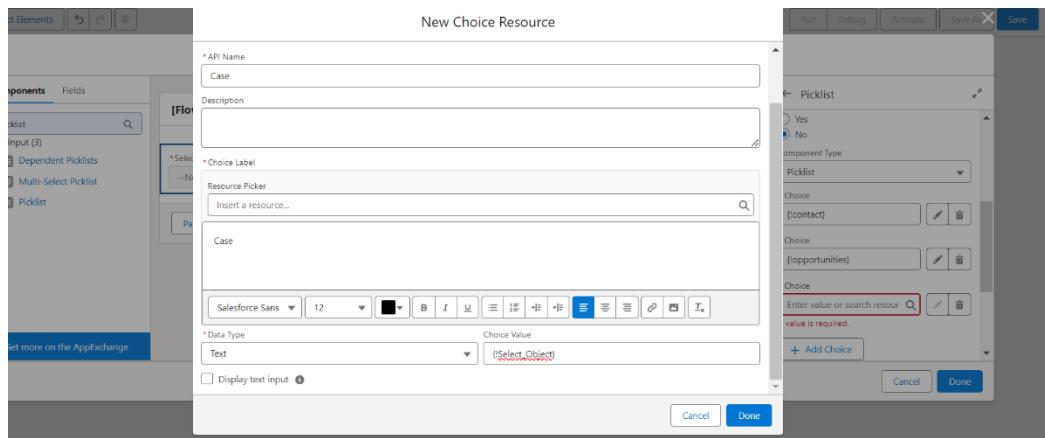
- After that go to the records of account and check the result of the flow.

Create a contact or an opportunity or case related to account based on user choice through one sub flow:

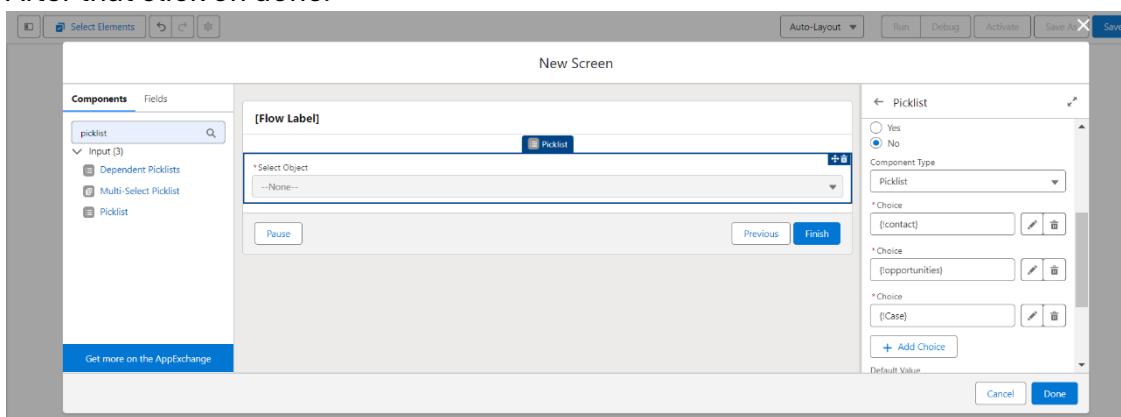
- For creating sub flow we have to create new flow. For this search flow in quick search and click on it after that click on new flow.
- After that select screen and click on create. This flow is for object selection (parents).
- After that click on + sign and select screen.



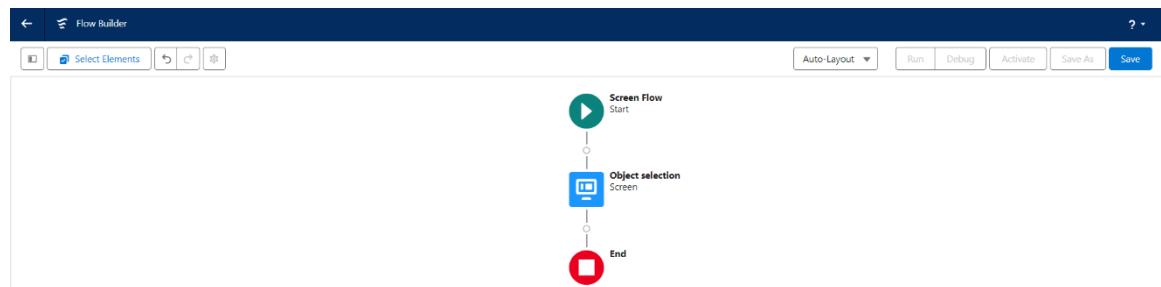
- After that enter the screen label and API.
- After that search picklist and insert into screen to make object selection.
- Create picklist value of contact, opportunity, case.



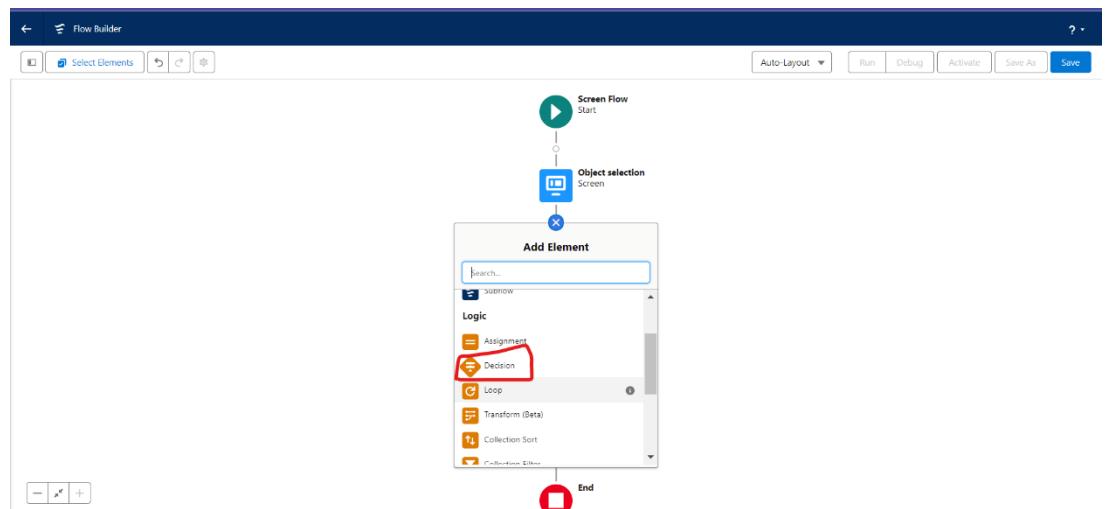
- After that click on done.



- This way the object selection screen is created.

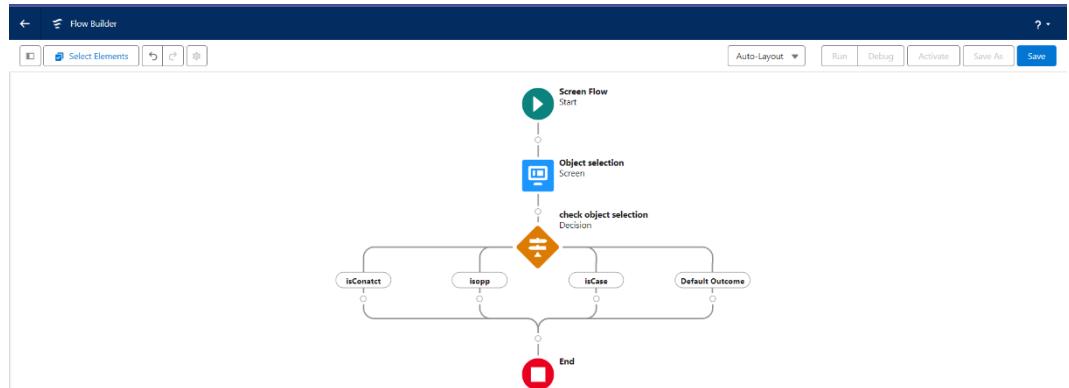


- After that we are going to create decision. For this click on + icon below screen and select **decision** under **logic**.



- After that enter Label and API name.
- After that enter outcome details for contact. Label, API name, Resource as select_object(picklist) from screen components operator is equal and value is contact from choice. This way contact outcome is created.
- Same way we have to create outcome for opportunity and case by clicking on + sign.

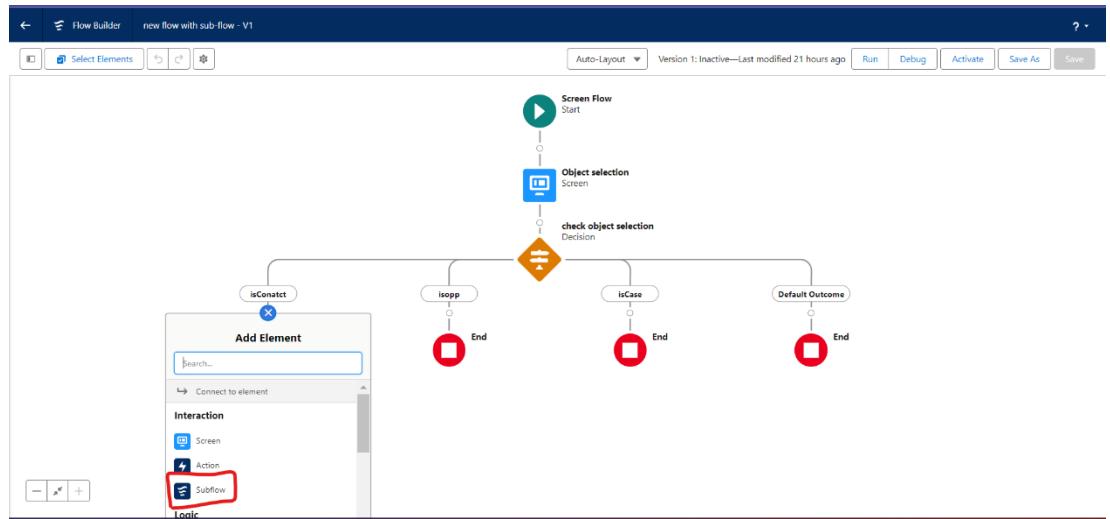
- After creating click on done. Now decision is created.



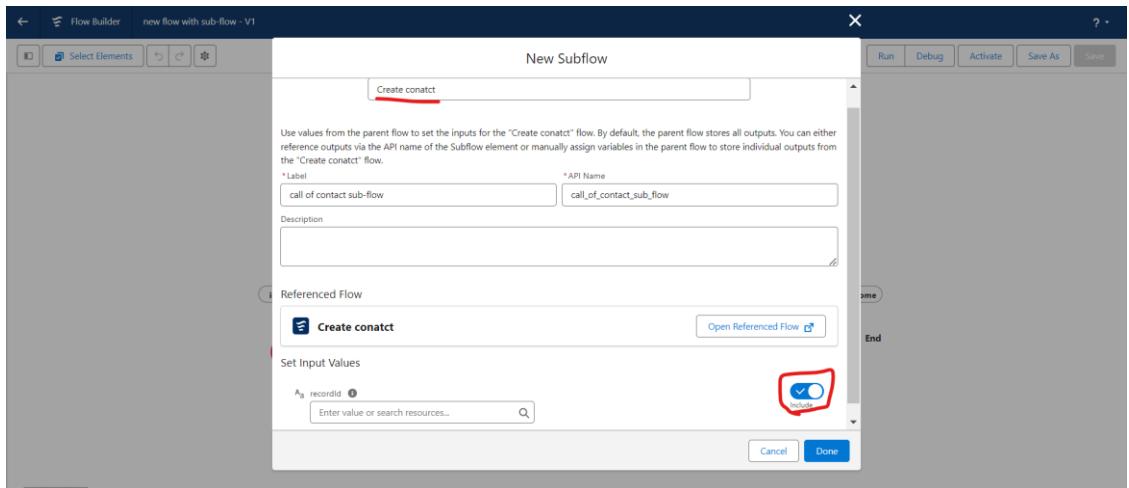
- After that we are going to create sub-flow for each object and linked this with account.
- To use sub-flow, we have to create new flow for each object and then we have to add them into parent flow.
- Firstly, we are going to create flow for contact with account linked.
- After that go to the parent flow and click + sign under iscontact.

Flow Label	Process Type	Active	Template	Package State	Packaged By	Last Modified	Last Modified Date
HIND Contact Associated with messaging session	Inbound-Object Linking Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Generate Appointment Invitation	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Inbound Cancel Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Inbound Modify Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Inbound New Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Inbound New Guest Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
new flow with sub-flow	Screen Flow	<input type="checkbox"/>	<input type="checkbox"/>	Unmanaged	Zahid Khan	10/01/2024, 5:30 pm	
Orchestration flow for Recurrence Scheduler	Autolaunched Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Order on Behalf Of Flow	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Outbound Modify Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Outbound New Appointment	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Pay Now with Products	Screen Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Recurrence Schedule Flow	Autolaunched Flow	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Managed-Installed			

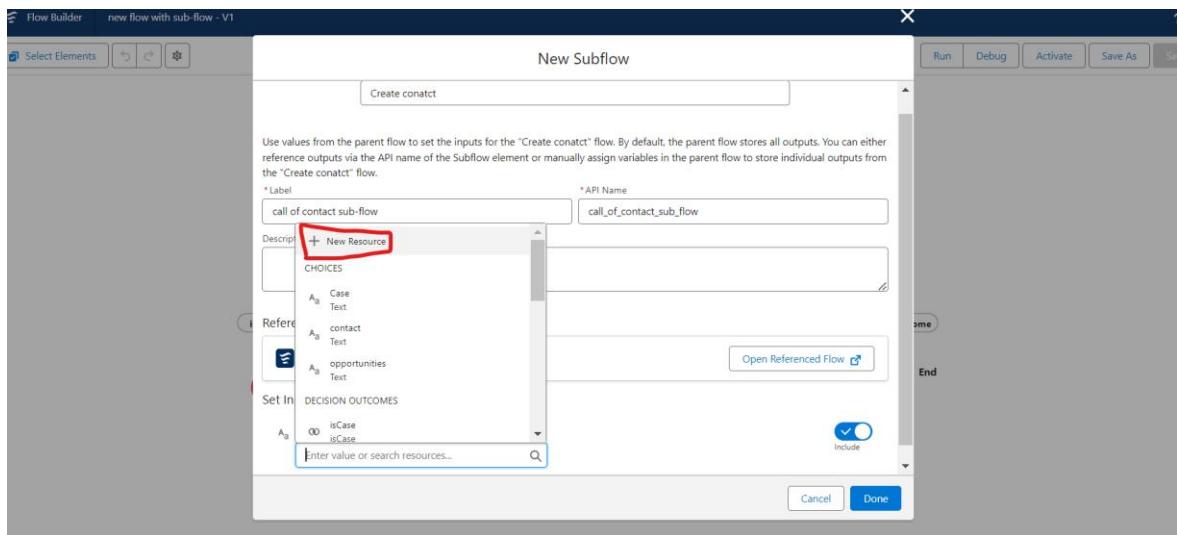
- After that select sub-flow.



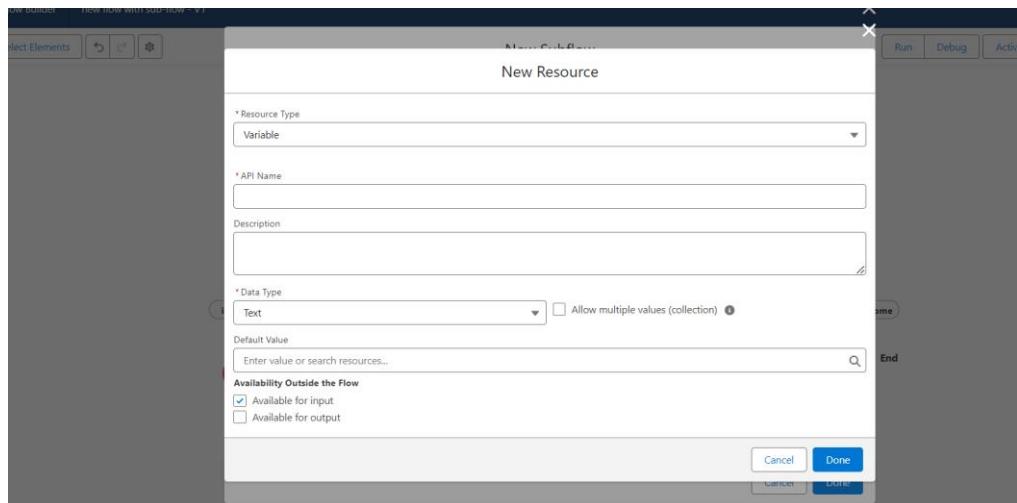
- After that select reference flow which is created for contact.
- After that enter label name and API. recordId under set input value toggle button should be green.



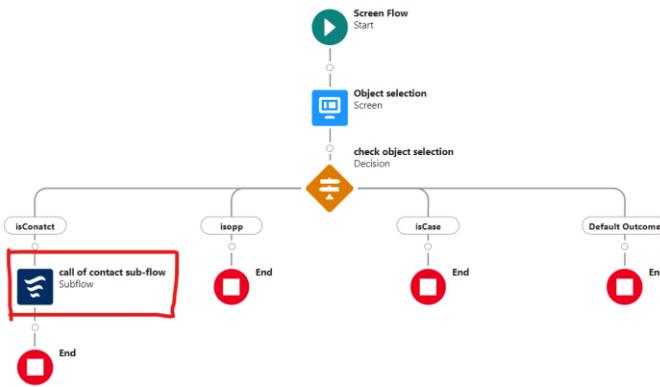
- After that in recordId select new resources .



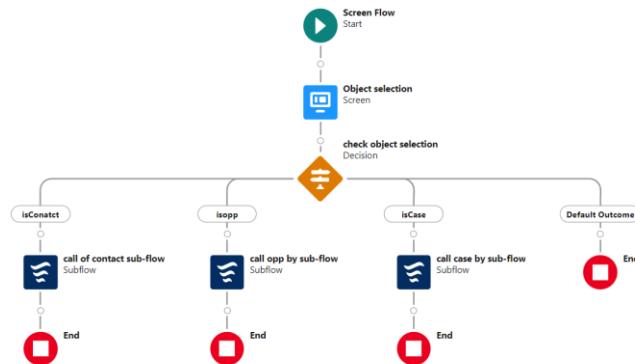
- Here, we need to create new variable for account id with name recordId.
- After that select variable in resource type and API name should be record id and data type text. Check box available for input. Click on done.



- After that click on done.



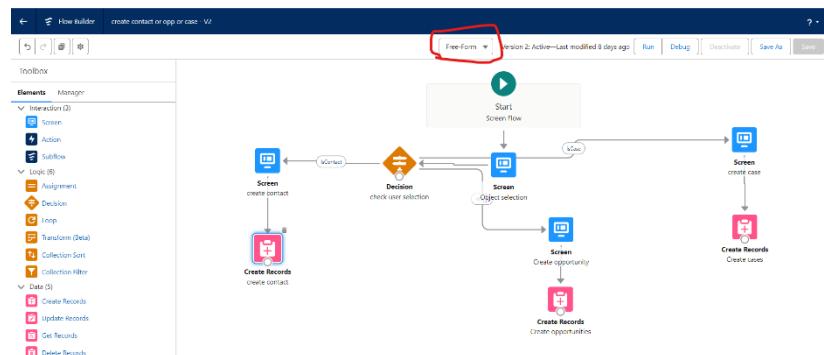
- In same way we have to create new flow for opportunity and case and add them in parent flow as sub-flow.
- After adding all flow our flow will look like this.



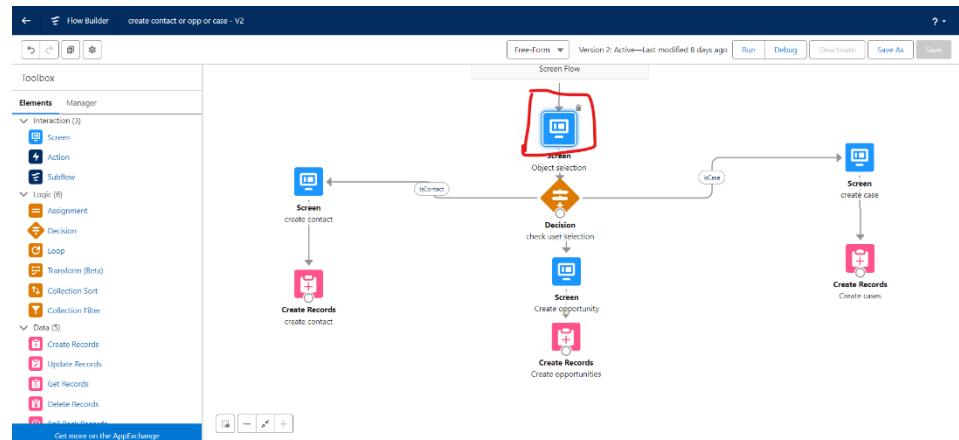
- After that go to account record page and place this flow.

Modify above flow to ask how many contacts or opportunity or case should be created related to account based on user choice:

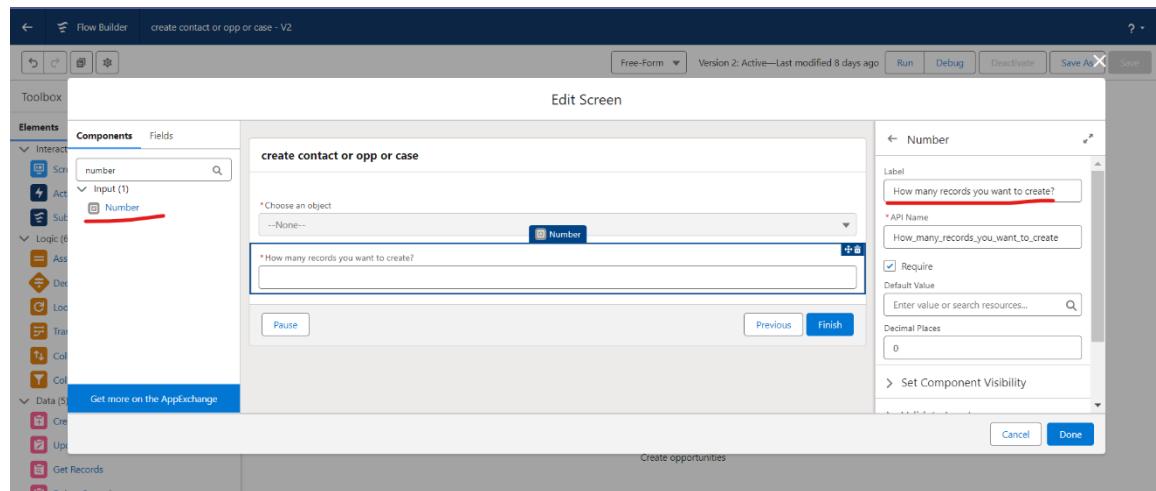
- For creating this scenario, we are going to modify the 1st scenario flow like we created a picklist for each object then decision after that make create screen flow.
- Firstly, convert the flow to free-form show that we can adjust the flow according to our requirements.



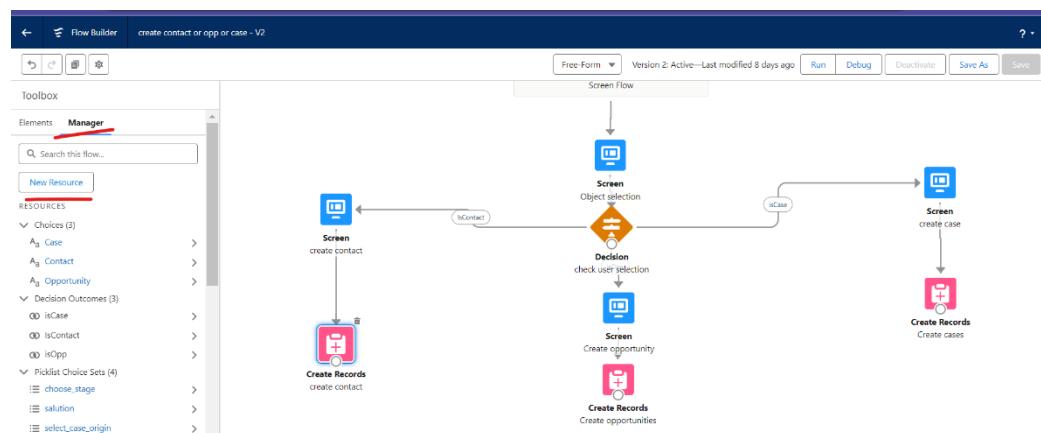
- After that open the **object select** screen by double click on it.



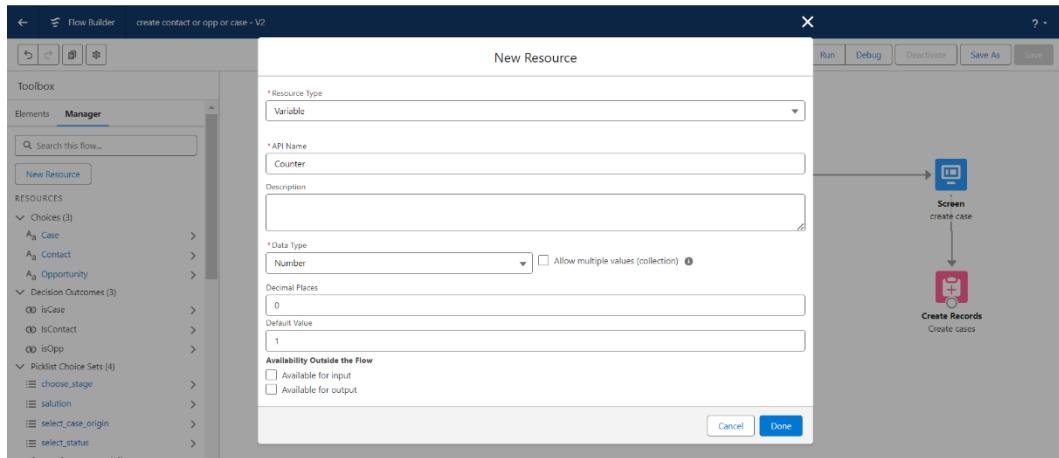
- After that add the **number** components to ask from user like **how many records you want to create?** make it required and click on done.



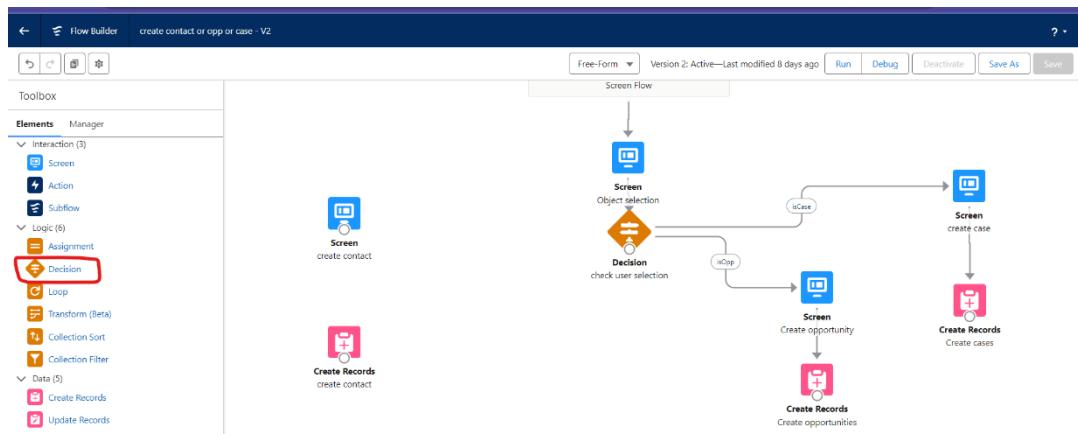
- Suppose that user select contact object and records 5 times so the isContact screen should be repeat 5 times. Now, we have to create loop for this scenario.
- For this we have to create a new resource. Go to manage and click on new resource and select variable.



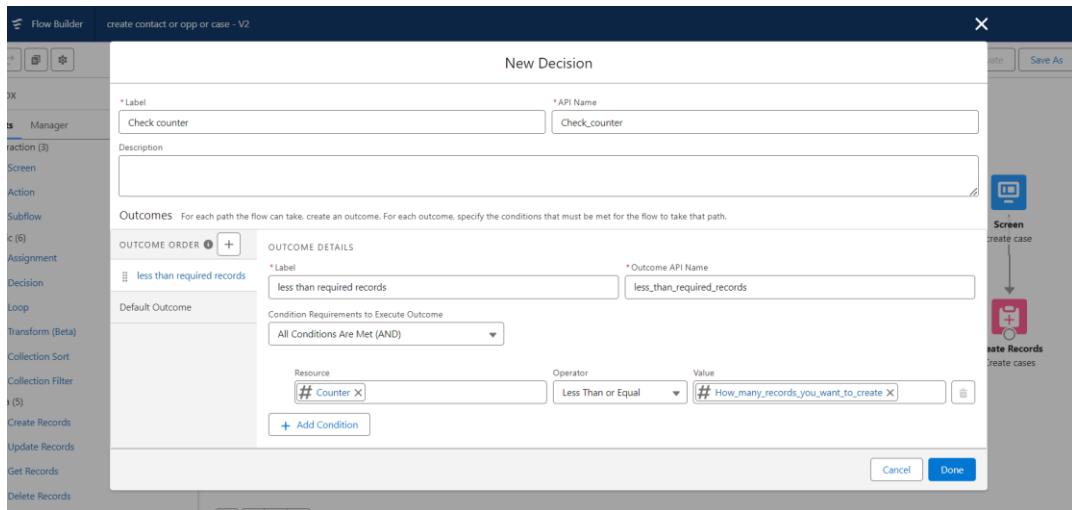
- Enter the label and Api name, data type should be number, decimal places is 0 and default value is 1.



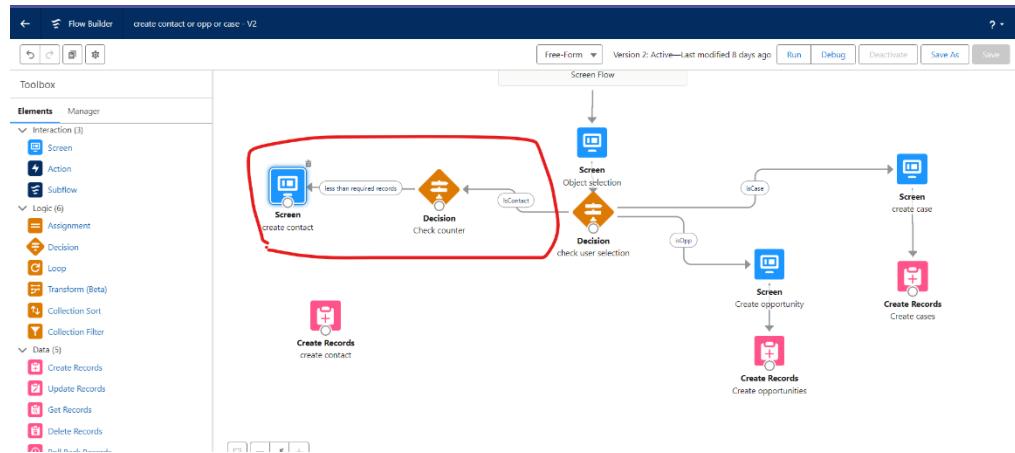
- Here, we are creating a counter which default value is 1 it works like a loop. Now click on done.
- After that delete the line which connects to the isContact screen to the decision show that we can add another element. To delete line just click on it and press delete button.
- Now, go to elements tab and select the decision to create loop.



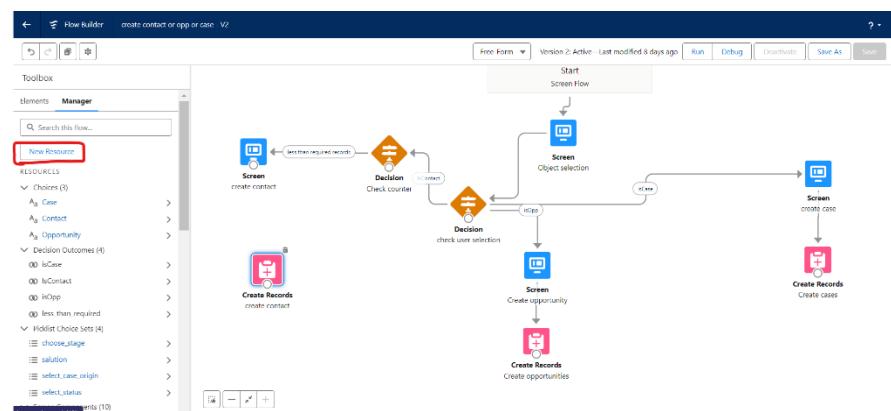
- After that enter the label name and api name. then enter details outcome label and api name.
- After that enter the Resource as **counter** variable.
- Operator as **less than or equal**.
- Value is **How many records you want to create?** Number component which we created in object select screen.



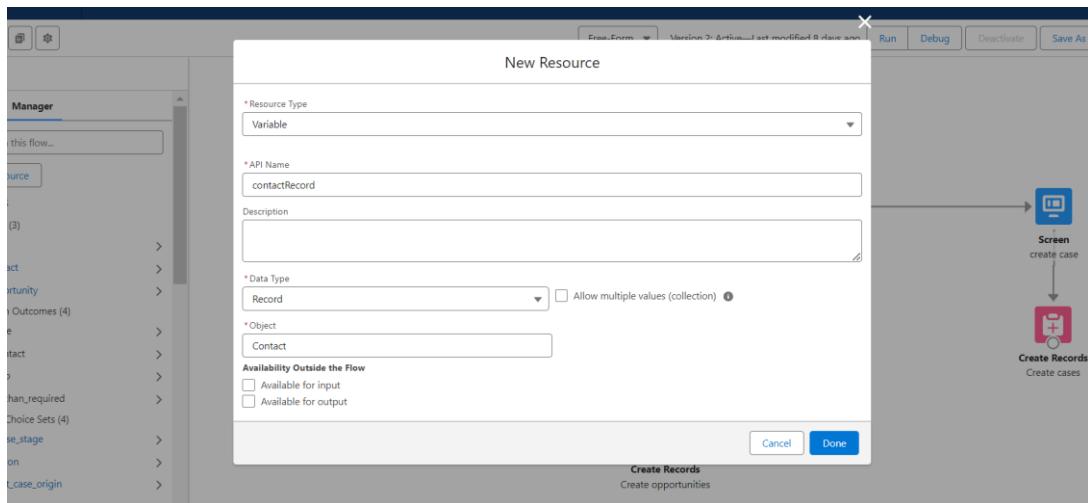
- After that click on done. Now the decision loop is created.
- Now, first connect the connect the **check user select decision** to **check counter decision** after that to the **isContact** screen. As shown in below image.



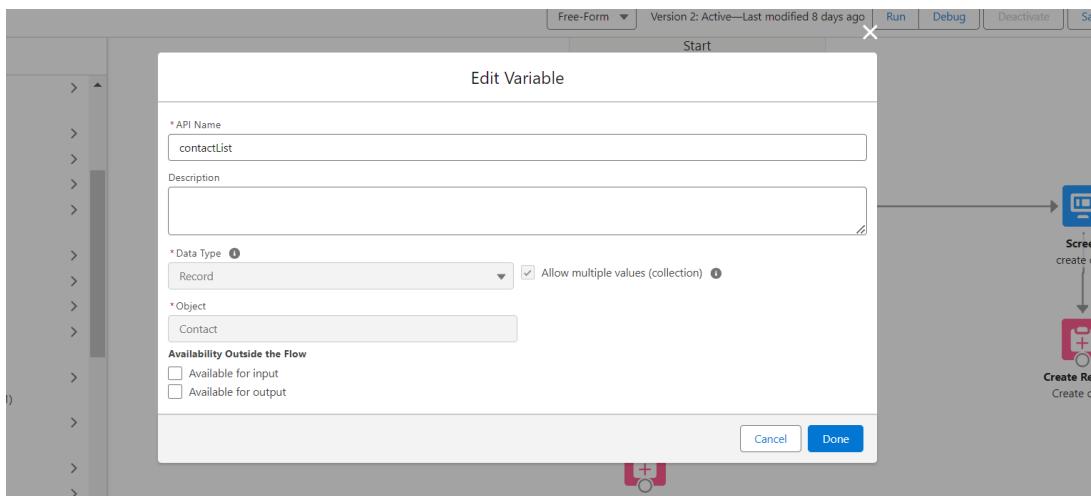
- we are not going to add the create records of isContact to this loop, so we disconnect the line.
- Here, we are going to create 2 variable 1st variable recording contact related information. 2nd variable for **contactList** as collection.
- To create these 2 variables, go to manager and select new resources.



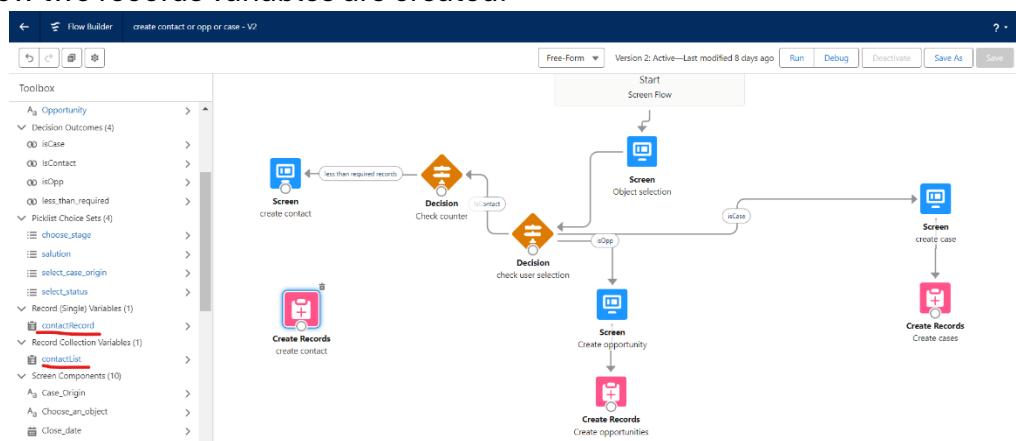
- After that select variables. Then enter API name as contactRecord and data type is record. And object will be contact. And click on done.



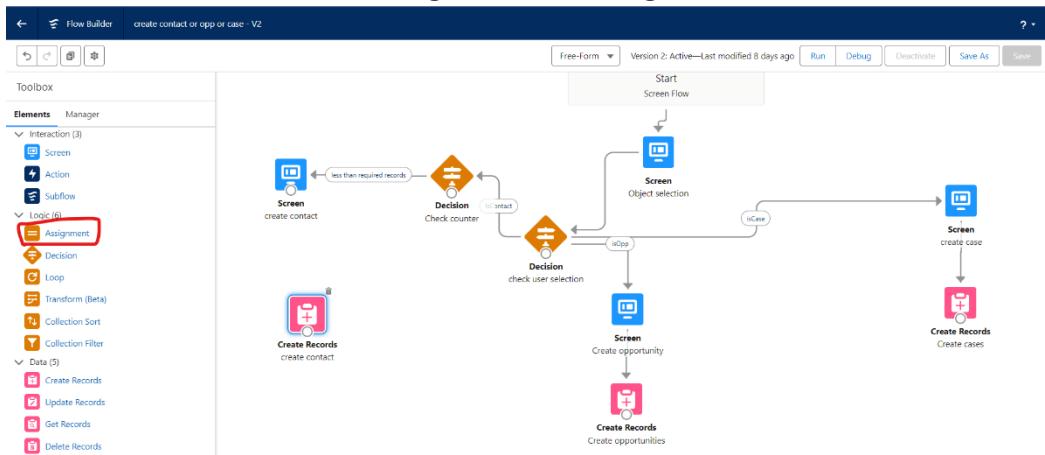
- Now again click on new resources to create variable as **contactList**.
- After that enter the API name as **contactList**, data type should be Record, object is contact. And check the box **allow multiple values**. And click on done.



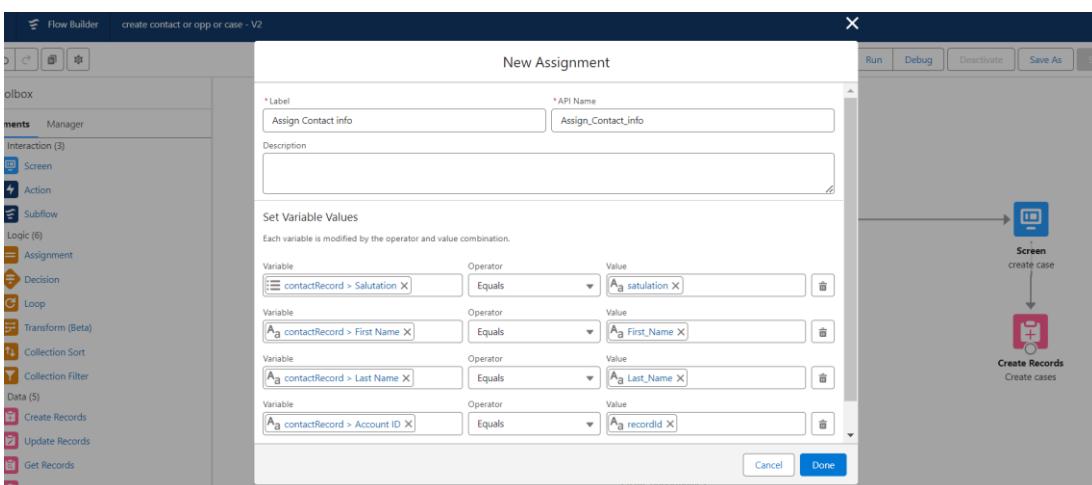
- Now two records variables are created.



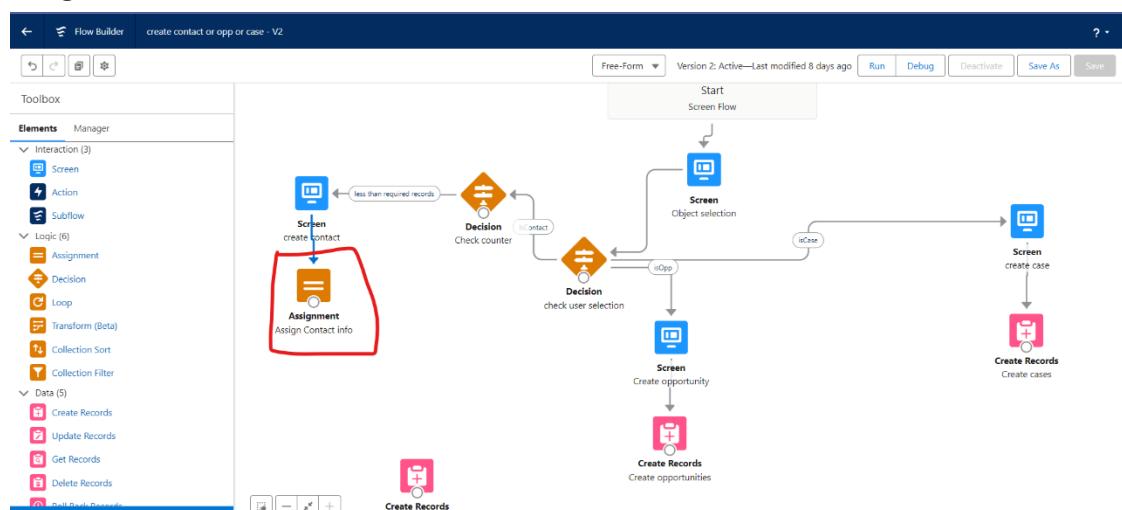
- After that we have to add assignment from elements.
- Go to elements and select assignment and drag it to the canvas.



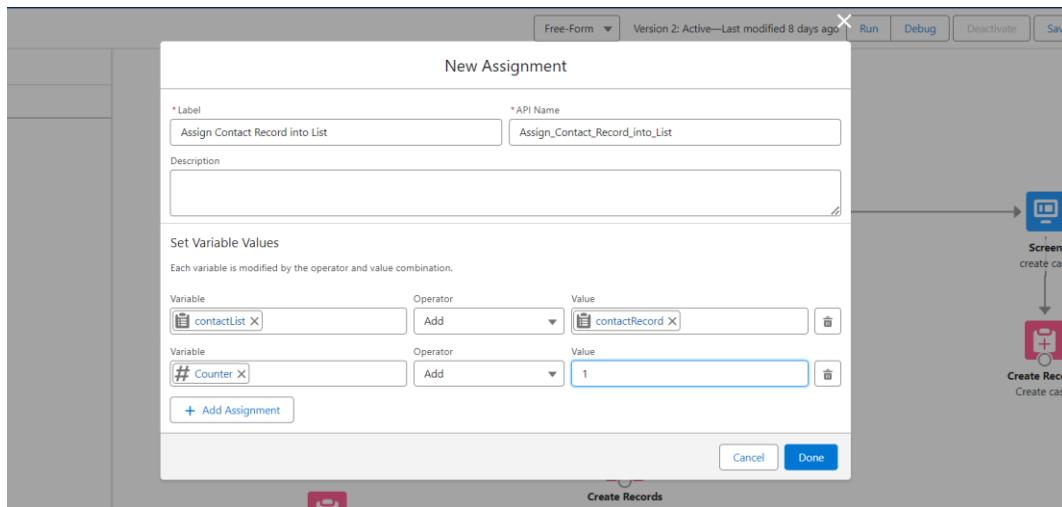
- After that enter the label and API name. now we are going to map the fields of contact object inside the contactRecord variable. Like below image and click on done.



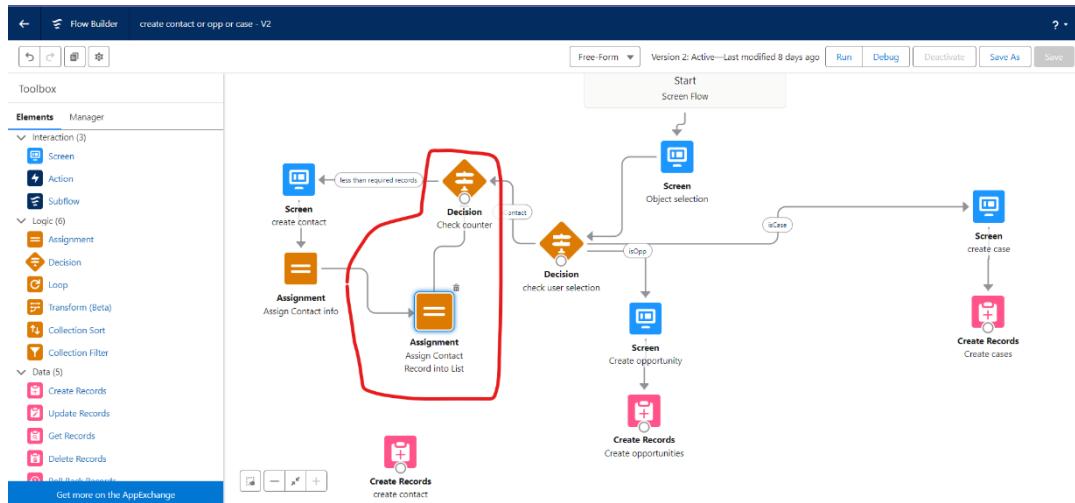
- Now assignment is created. after that connect the screen of contact with this assignment.



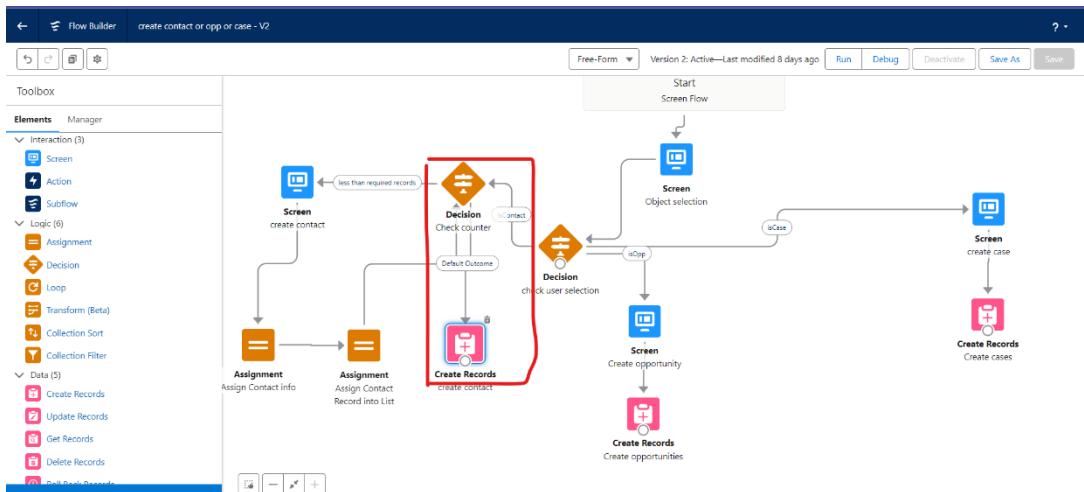
- After that we have to create one more assignment for **contactList** which add the records of contact which is created in **contactRecord** variable.
- Again, click on assignment inside elements and place it in canvas.
- After that enter label and API name, variable is **contactList**, operators is ADD and value is **contactRecords**.
- After that click on Add assignment button to add the counter. Variable is **counter**, operator is ADD and Values is 1. And click on done.



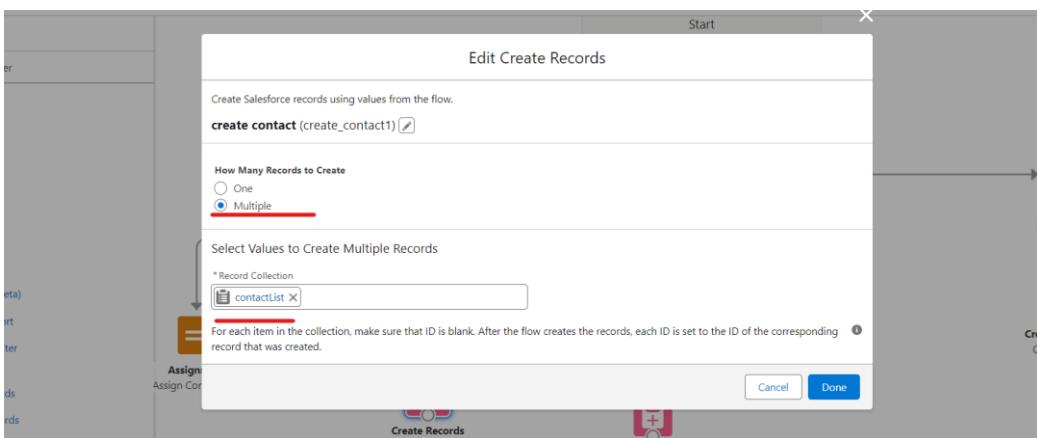
- After that connect the **Assign contact info assignment** with **assign contact Record into List assignment**.
- After that connect the **assign contact Record into List assignment** with the **check counter decision**. This way a loop is created.



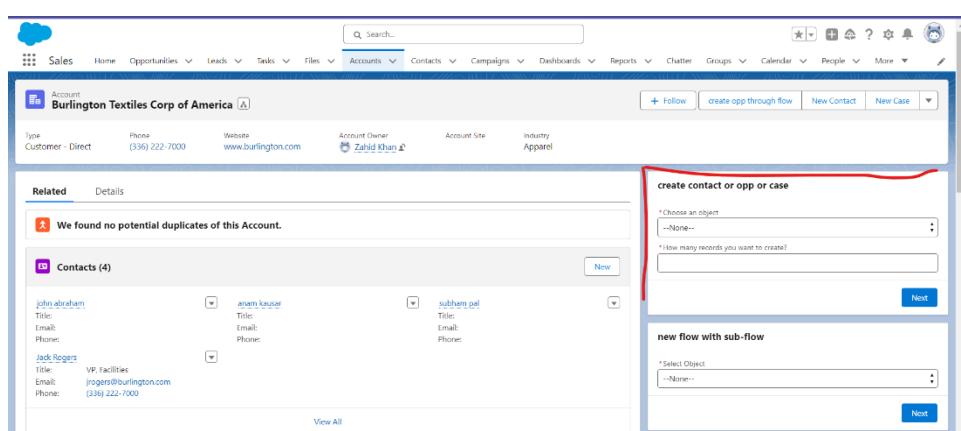
- Once this loop is fulfilling the check counter decision will be directed to create records of contact. So, connect the check counter decision with create records.



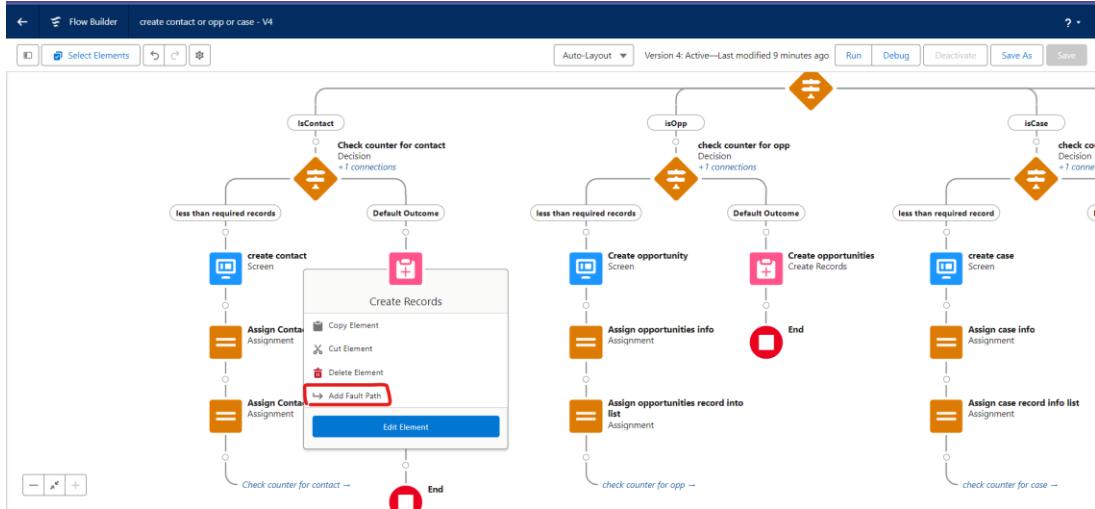
- now, we are creating multiple records, so we have to choose multiple in create records of contact.
- Just double click on create records and choose multiple in **How Many Records to Create**, and for record collection enter the contactList variable which we created. and click on done.



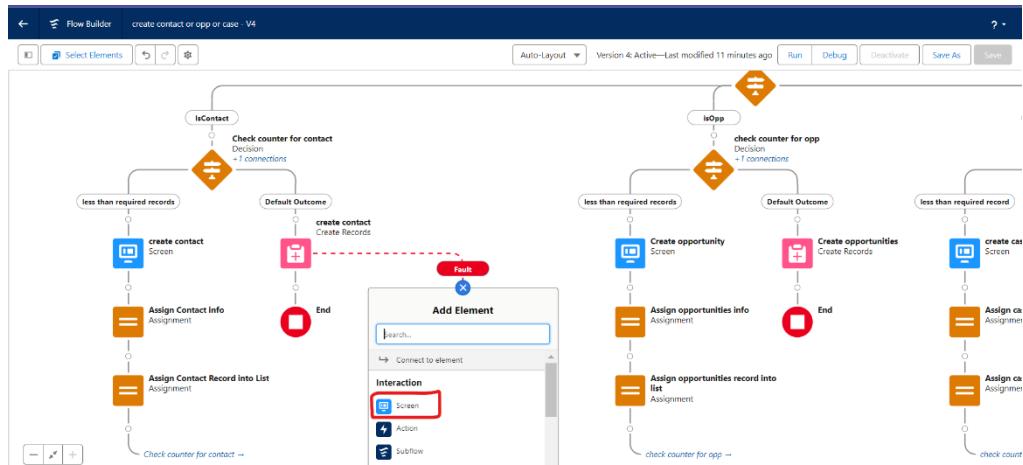
- This way the contact part is done in same way we have to create this for opportunities and case. But here we are using same variable counter only apart from this we have to repeat all steps.
- Now place this flow on account record page. And check the results.



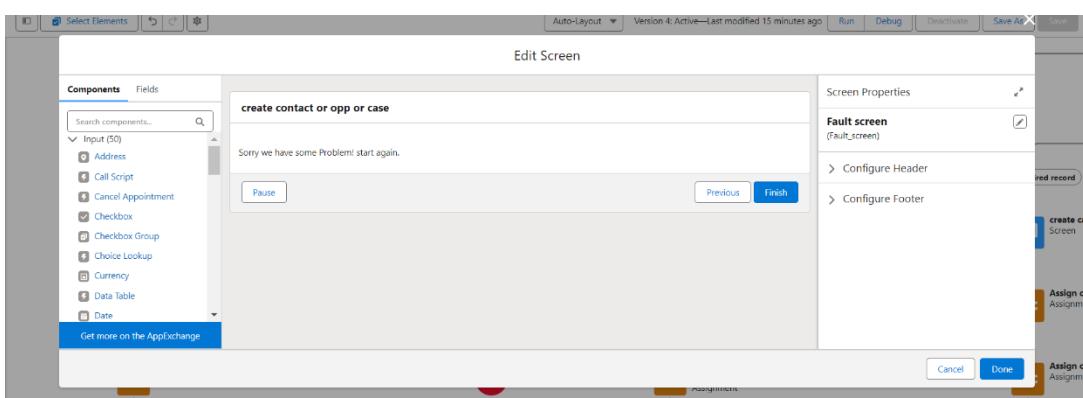
- We can also add fault path for the flow so if the create records is not working so that we can use fault path and show some text.
- To use fault path first convert the flow into auto-layout after that click on the create records and select Add fault path.



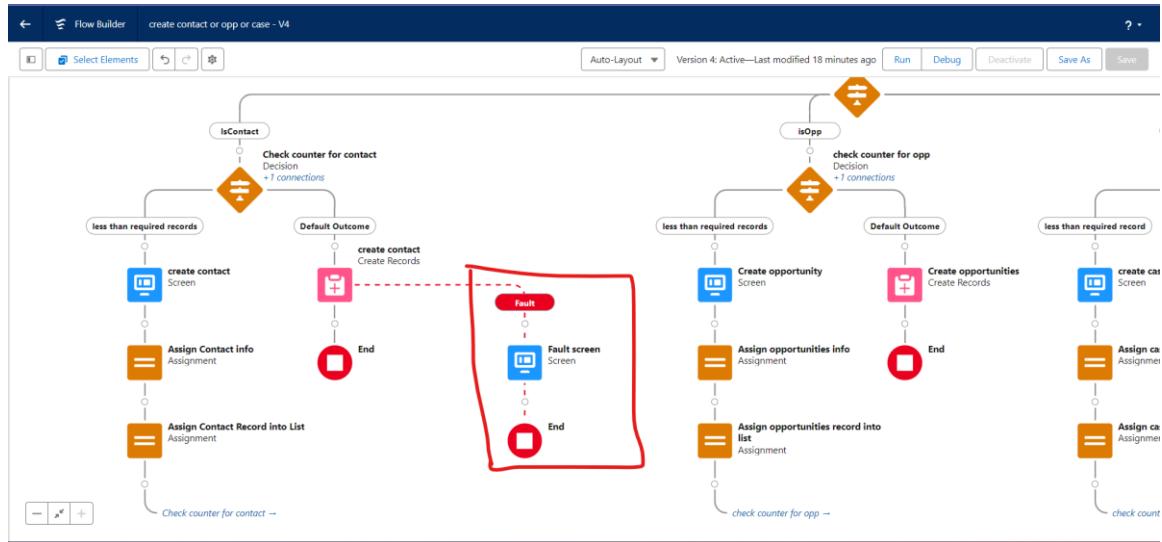
- After that click on + sign below fault and we can use anything like screen, sub-flow. Here we are using screen to show text.



- Now select text from display and add it to the screen.



- Now fault is created. so when ever the create screen is fail so fault screen will show the text.



DAY-27

TRIGGERED FLOW

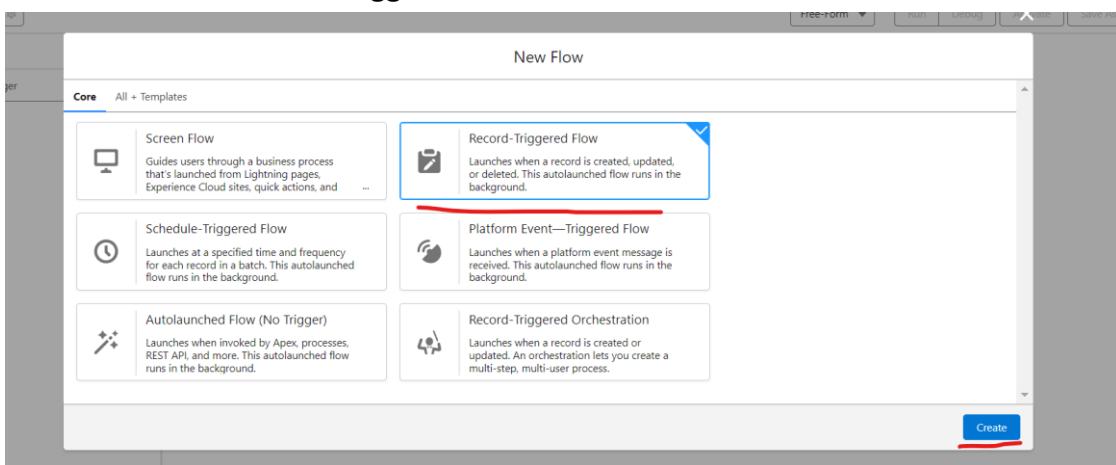
- Record triggered flow can't be visible in front-end or UI. It will run in background.
- Record triggered flow will be trigger when record is created, updated, or deleted.
- Screen flow is used to show the flow on UI.

Scenario:

- On account creation if industry is = 'media' then popular rating as 'hot'.
 - Upon creation of an account, a related opportunity should be created automatically. [no criteria].
Opp name = opportunity name
Close date = close date
Stage = prospecting
 - Now add a criterion if Active = Yes on account then newly opportunity should be created.
- When a contact is created the update the 'Total contact' field

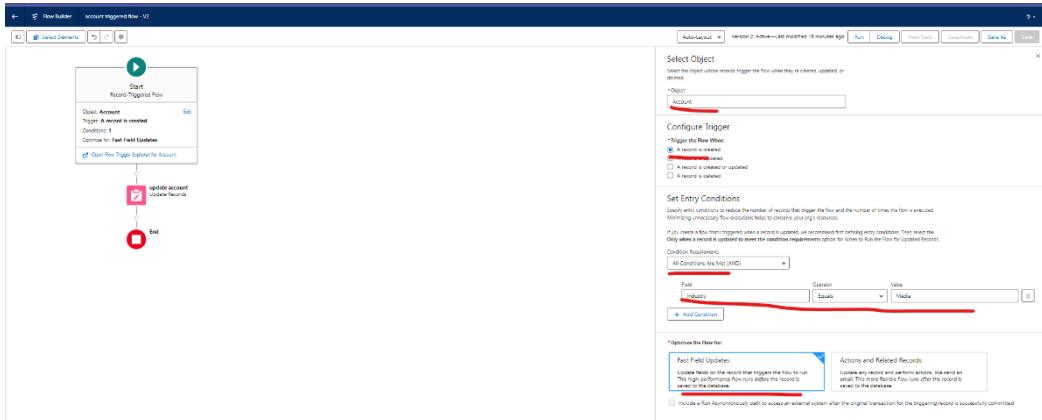
On account creation if industry is = 'media' then popular rating as 'hot'.

- Go to the set-up and search for flow on quick search and click on new flow.
- After that select record-triggered flow and click on create.

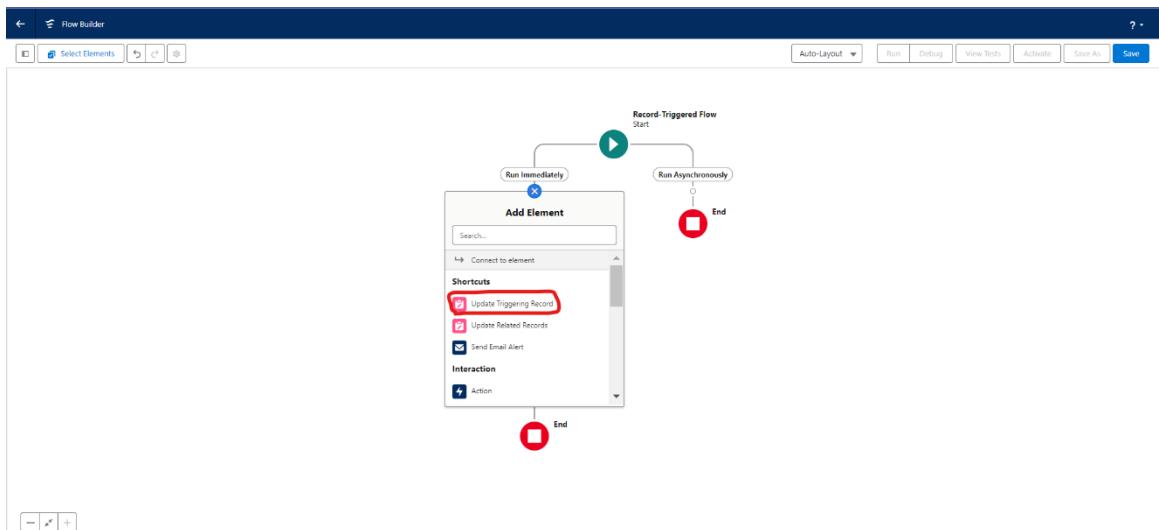


- After that we have to select object in triggered flow. In our requirements we are using account object so select account object.
- Inside configure trigger, **A record in created** because we are creating account records.
- In set entry **conditions requirement** choose AND and conditions is **industry is equal to media**.

- In optimize the flow for we have 2 option **Fast Field updates** and **Actions and related records**.
- If we want to do something on the same record through which flow is triggered, then choose Fast Field updates.
- If we want to do something on the related record, then go with actions and related records.
- Here, we are selecting Fast field updates.

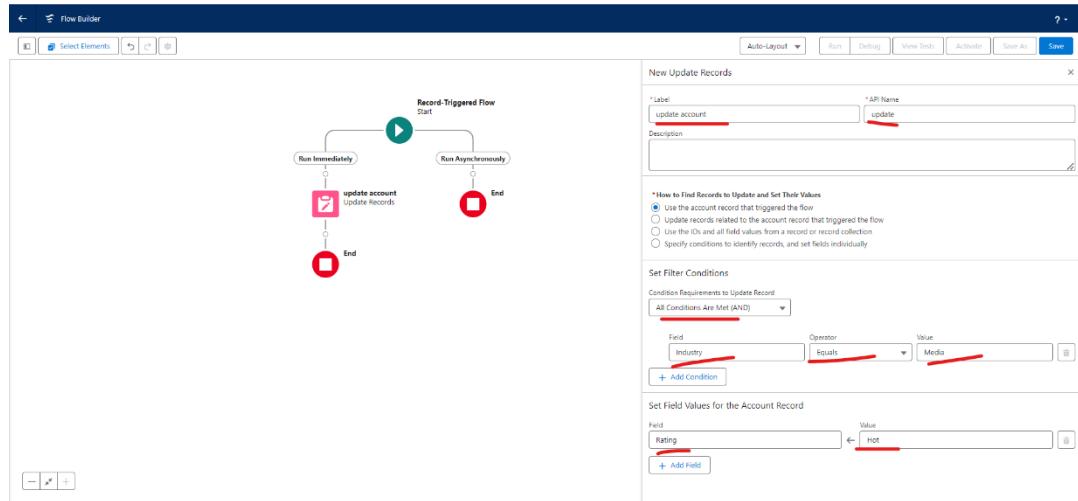


- After that click on + icon on run immediately and select **update triggering record**. Select this option when we want to create account records and through which this flow will triggered.

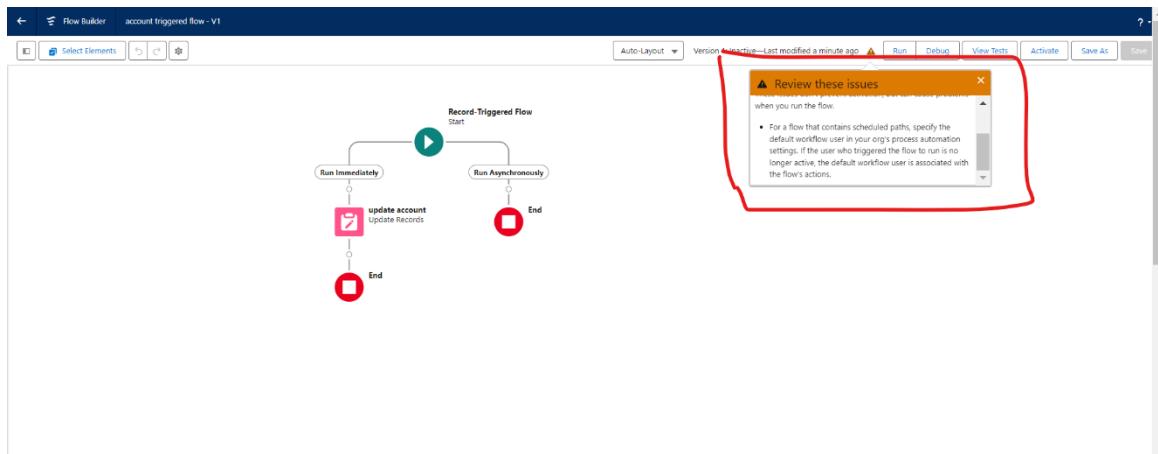


- After that enter label and API name. in **How to Find Records to Update and Set Their Values** choose 1st option.
- In set filter conditions we are going to make criteria according to our scenario if industry is = 'media' then popular rating as 'hot'.

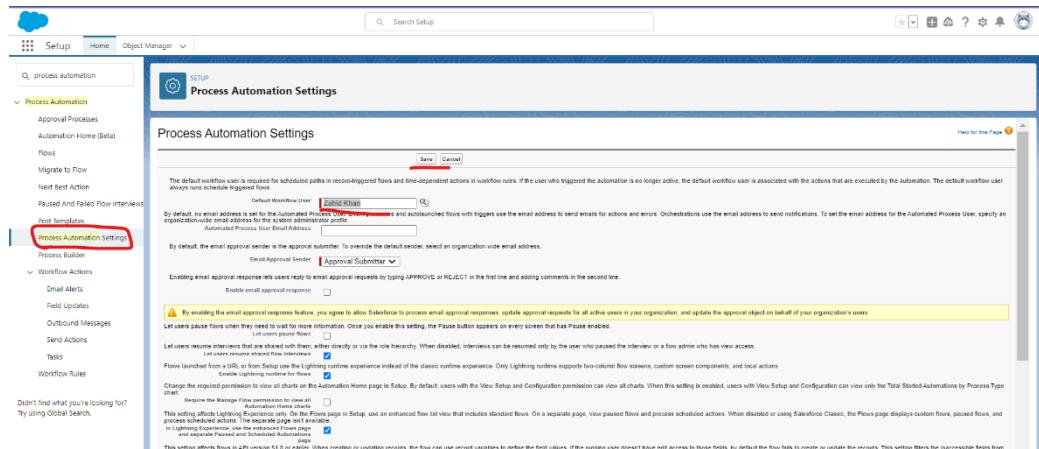
- So, in Set Filter Conditions select AND and condition is when industry is equal to media the rating is hot.



- now when the industry = 'media' so rating will be hot otherwise this flow will not be triggered.
- After that save the flow.
- After save we got a warning.

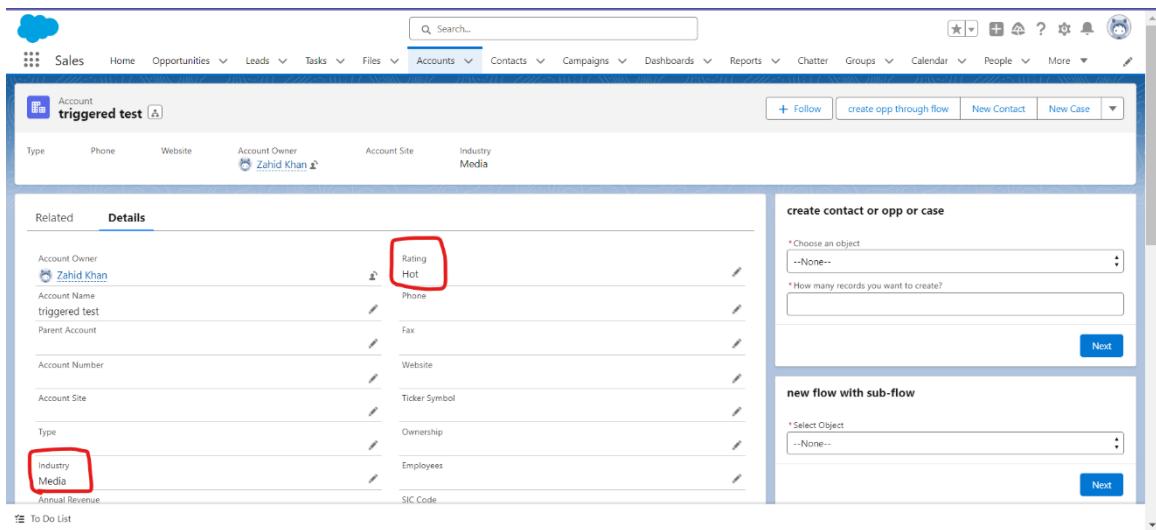


- To solve this warning, we have to set default workflow user. For this go to home and search for process automation settings. After that enter system admin name in default workflow user. And click on save.



- After that refresh the page so warning will be gone.
- Now activate the flow.

Now to account object and create a record and choose media in industry so that rating will become hot automatically because of this triggered



2nd-scenario:

1.Upon creation of an account, a related opportunity should be created automatically. [no criteria].

Opp name = opportunity name

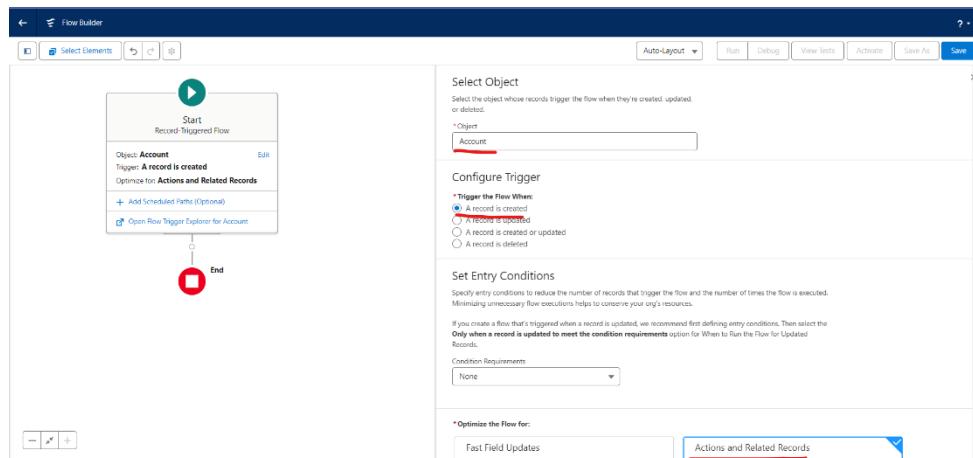
Close date = close date

Stage = prospecting

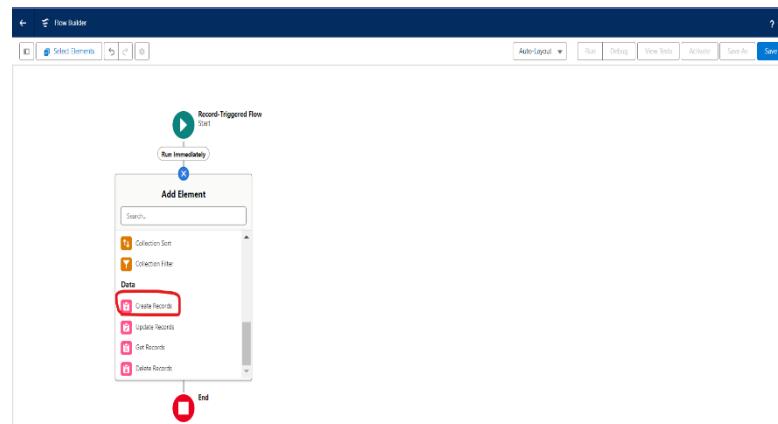
2.Now add a criterion if Active = Yes on account then newly opportunity should be created

1.Upon creation of an account, a related opportunity should be created automatically. [no criteria]:

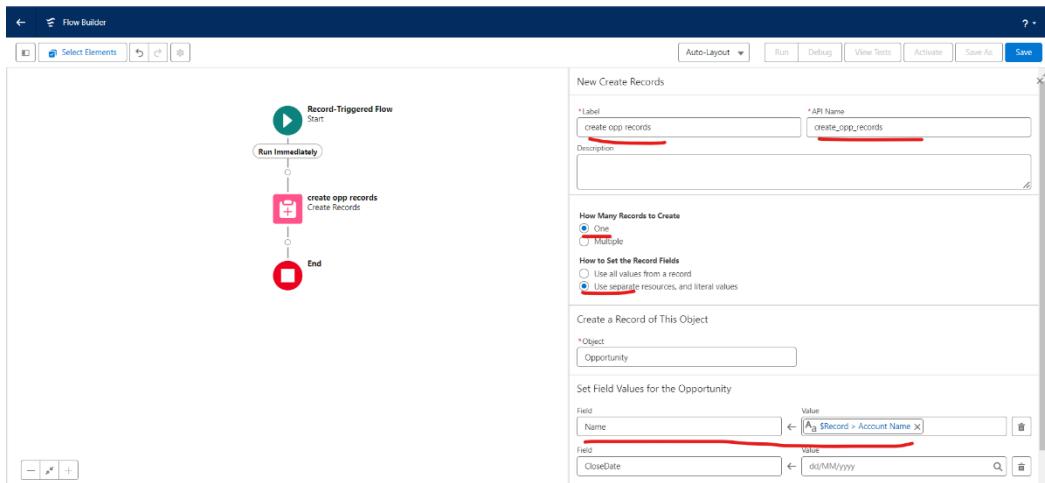
- Go to the home and search for flows and click on new flow and select the record-triggered flow.
- After that select account object and create a record and select Action and related records. And criteria are none.



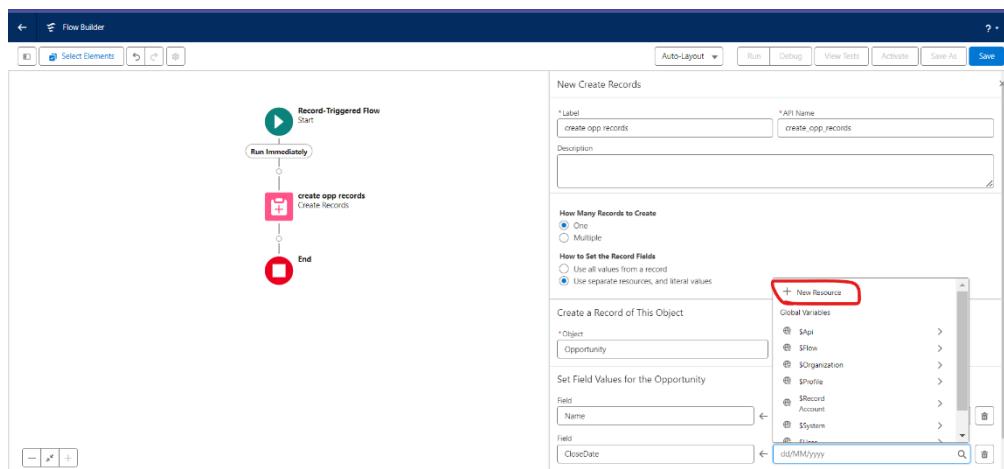
- If we have a validation rule in our org like few fields can't be empty so we have to add that field in our conditions otherwise we won't be able to create record.
- After that click on + icon and select **create Records** from **data** so that we can create record for opportunity.



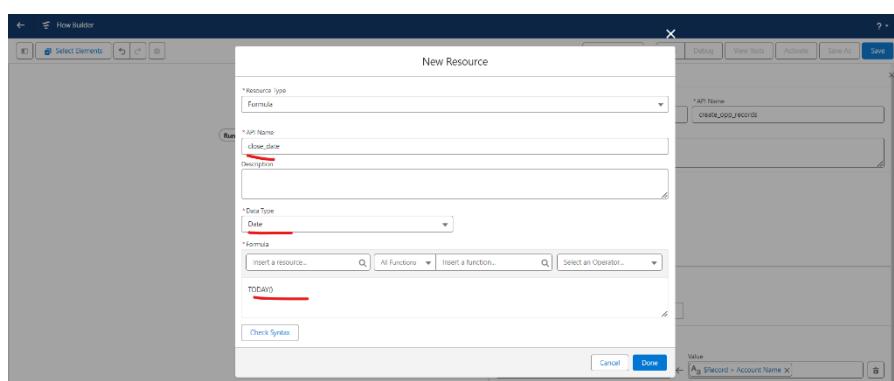
- After that enter the label and api name, how many records to create select one, how to set the records field select use separate resource. Select object as opportunity. Now we have to write the field name **as name, close date and stage**.
- In field type select **Name** and in value select **\$Record** inside this all information will available. So, the account name will be equal to opportunity name.



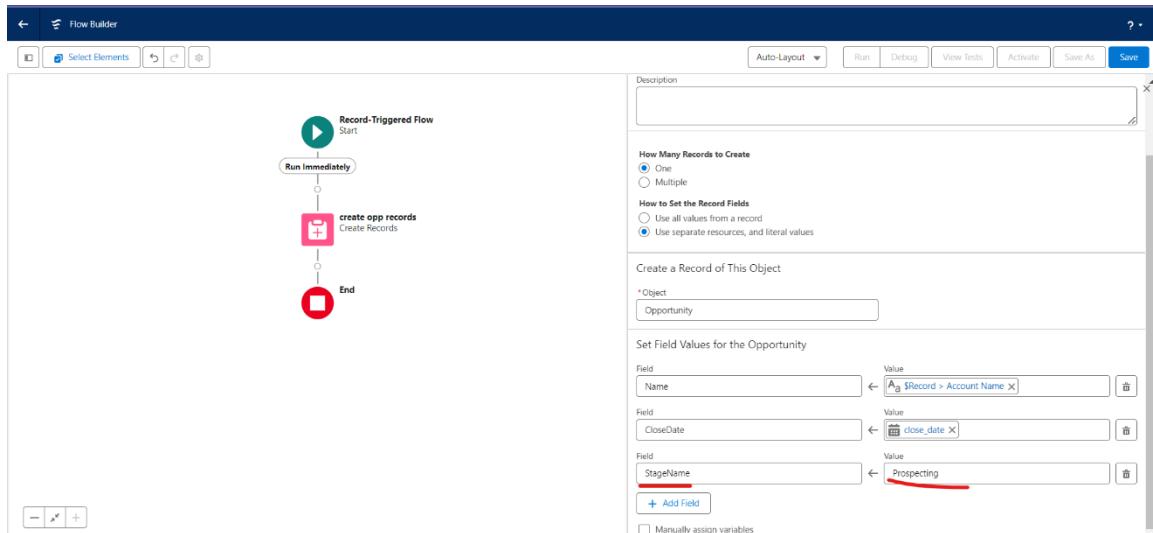
- After that enter close date in field. For value we have to create formula. For this click on new resources and select formula.



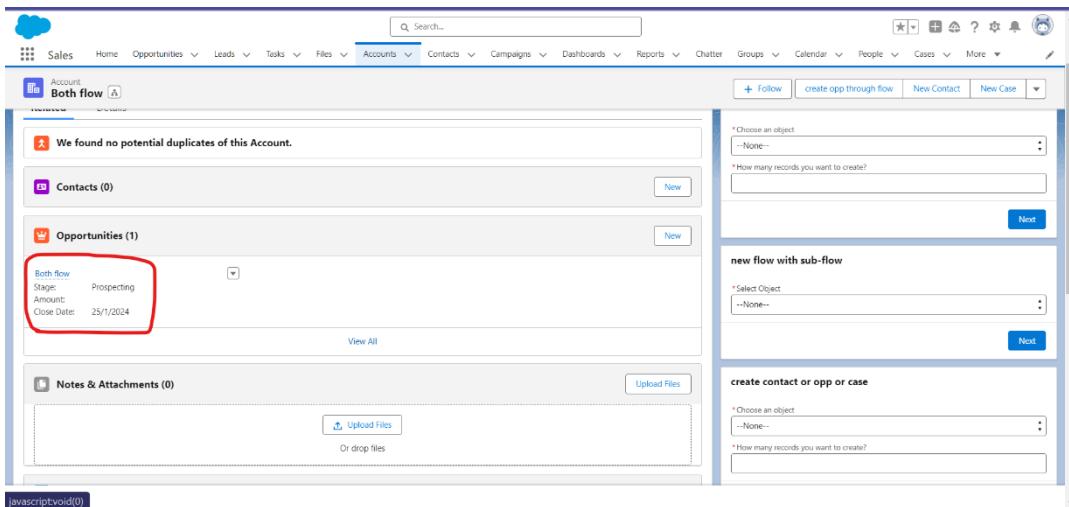
- After that type api name, data type as date and in formula type **TODAY()**. And click on done.



- After that we have to create stage. So, write stage in field and in value select prospecting.

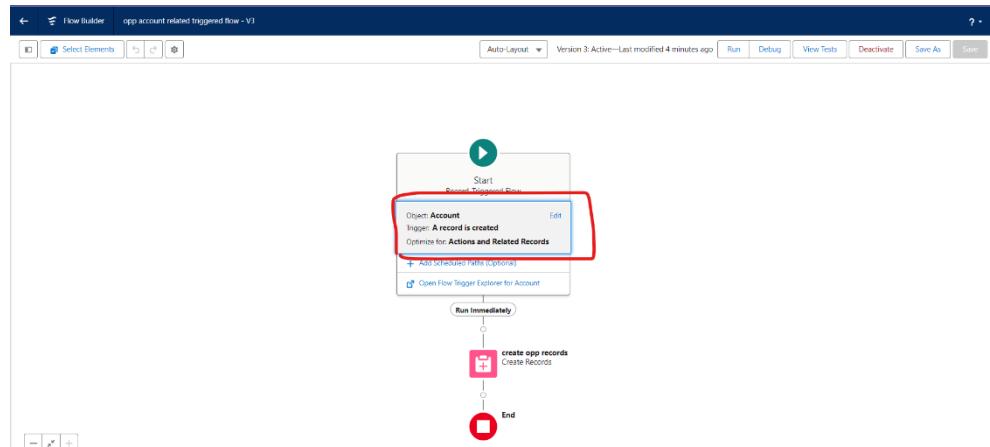


- After that add accountId in field and in value select \$Record then accountid so that this opportunity record will be related to account.
- Now save the flow and activate it.
- Now go to account object and create any record to see that working of flows.
- Opportunity is created by creating account records.

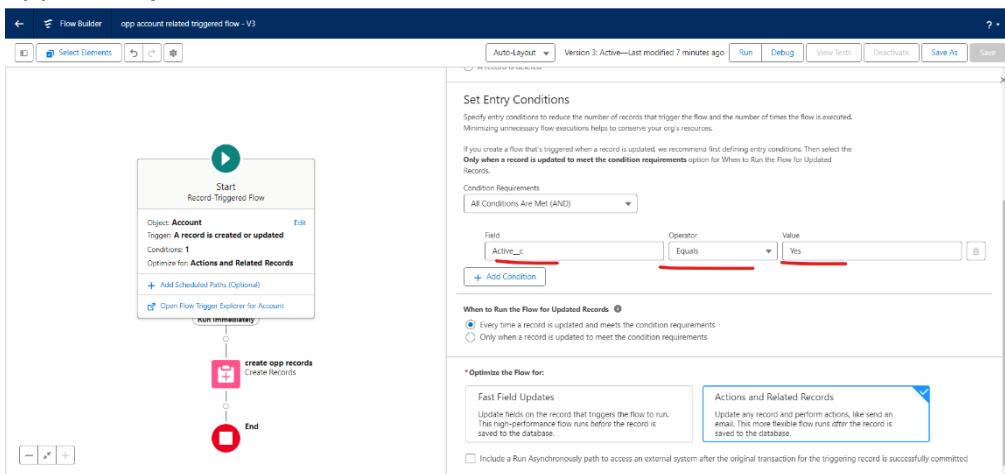


2.Now add a criterion if Active = Yes on account then newly opportunity should be created:

- Now we have to add the criteria in same flow. So, open the flow click on start button and flow and edit the record triggered flow.



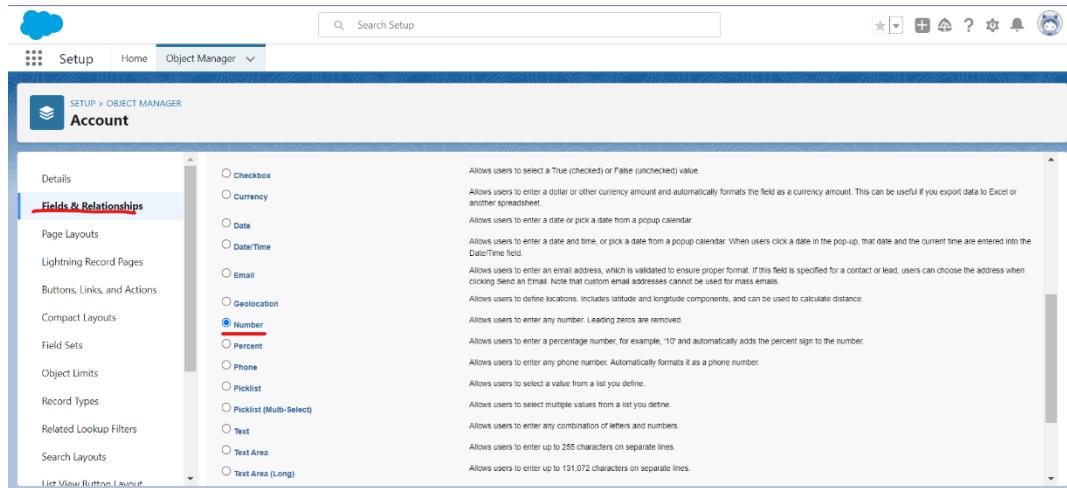
- After that enter the criteria in set entry conditions that active = 'yes' so that opportunity record created otherwise not created.



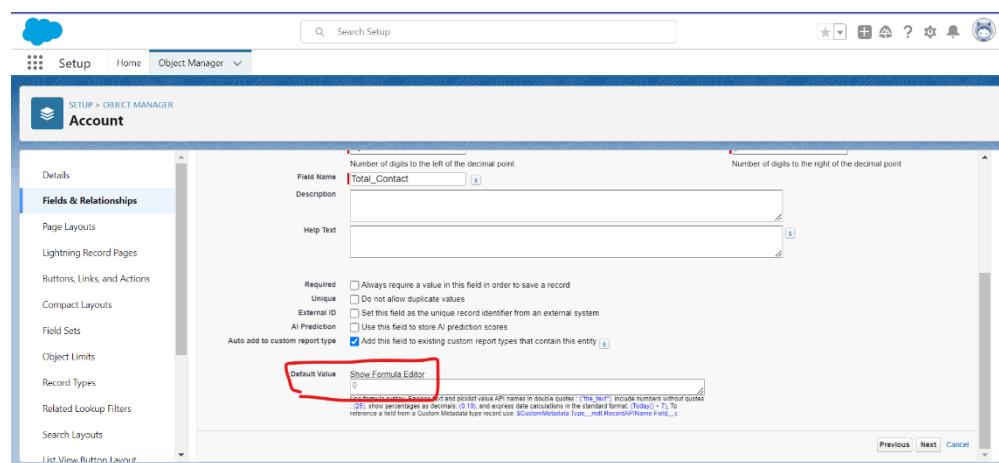
- Now save the flow and check that result.
- As we select active is no so the opportunity record is not created. if active is yes so that opportunity record in created.

3-Scenario-When a contact is created then update the ‘Total contact’ field on account using record triggered flow:

- Before making this flow, we have to create a field of total contact in account object. For this, go to set-up and click on object-manager after that select account object.
- After that click on field & relationship and click on new.
- After that select number in datatype. And create a field with name **Total Contact**.



- After that enter the default value is 0.



- Now total contact field is created.
- Now we can make a triggered flow which count the total contact on account when account is created.
- Go to the set-up and search flows in quick search. And click on new flow.

The screenshot shows the Salesforce Setup Flows interface. On the left, there's a sidebar with categories like Apps, Lightning Bolt, Flow Category, Process Automation, and more. The main area displays a table of flow definitions. The 'New Flow' button in the top right is highlighted with a red box.

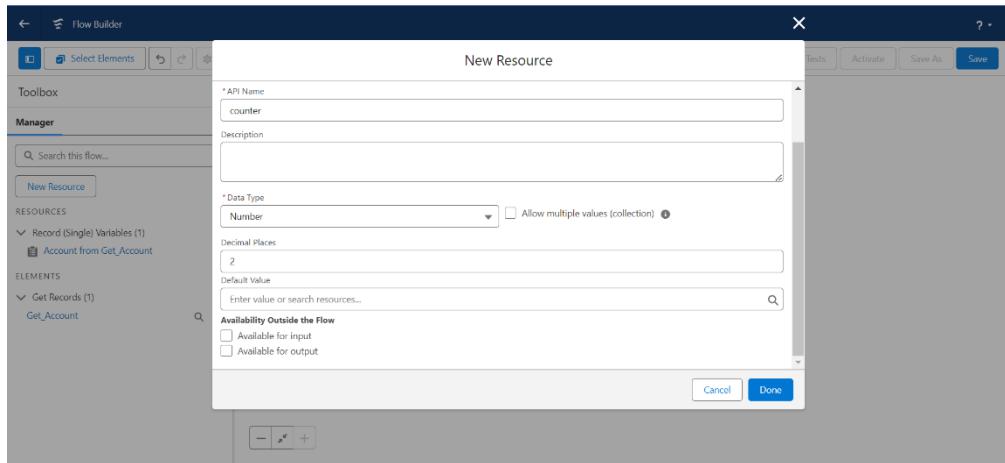
- after that select record-triggered flow and click on create.
- After that enter the object as **contact**, in configure trigger choose **A record is created**. then make entry criteria that accountid is null. And select actions and related records. And click on done.

This screenshot shows the Flow Builder for a contact-triggered flow. In the 'Configure Trigger' section, 'A record is created' is selected. In the 'Set Entry Conditions' section, there is a condition 'AccountId Is Null'. The 'Actions and Related Records' tab is also visible.

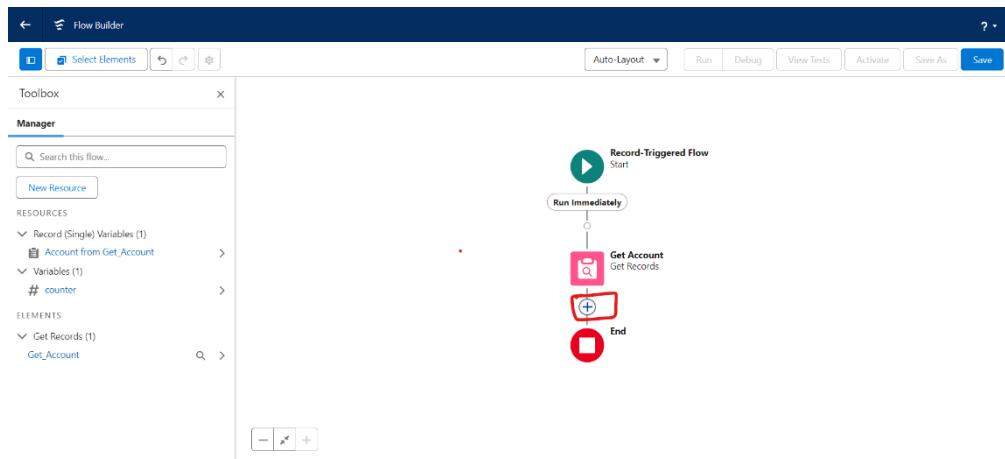
- After that click on + icon and select **Get Record** under **Data**.
- After that enter Label and API name and select account is object. And filter account records as id equal to account id. after that select only the first record and automatically store all field.

This screenshot shows the Flow Builder canvas. A flow starts with a 'Record-Triggered Flow Start' element, followed by an 'Add Element' step which contains a 'Get Account' action. The 'Get Account' action is configured with a 'New Get Records' section (Label: Get Account, API Name: Get_Account) and a 'Filter Account Records' section (Object: Account, Condition Requirements: All Conditions Are Met (AND), Field: Id, Operator: Equals, Value: \$Record > Account ID).

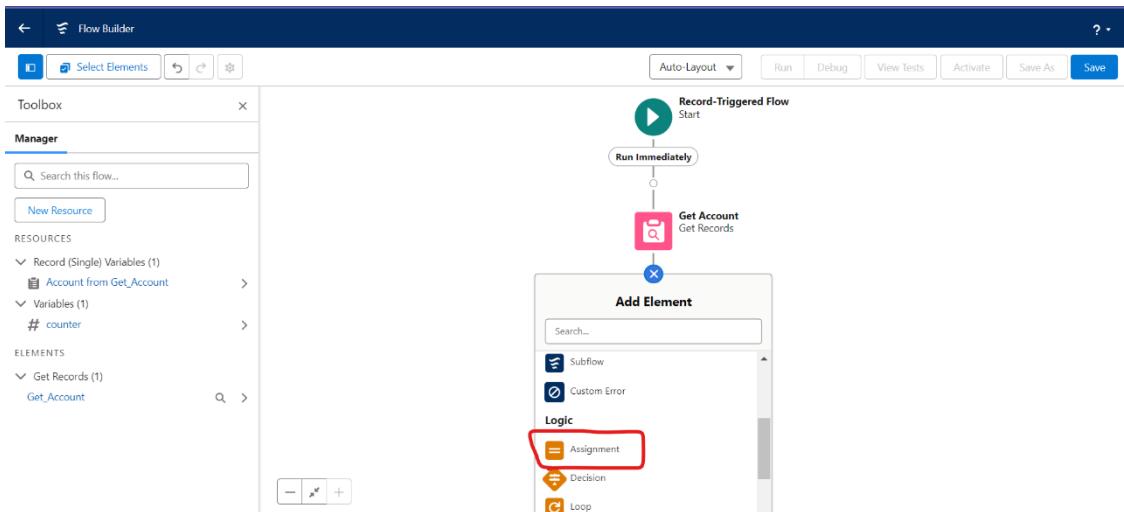
- Make a resource and resource type should be variable. Enter the API and data type should be number. And click on done.



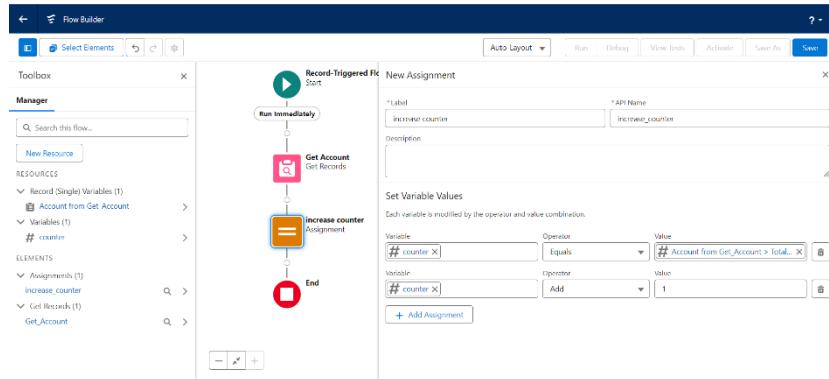
- After that click on + icon below Get Account.



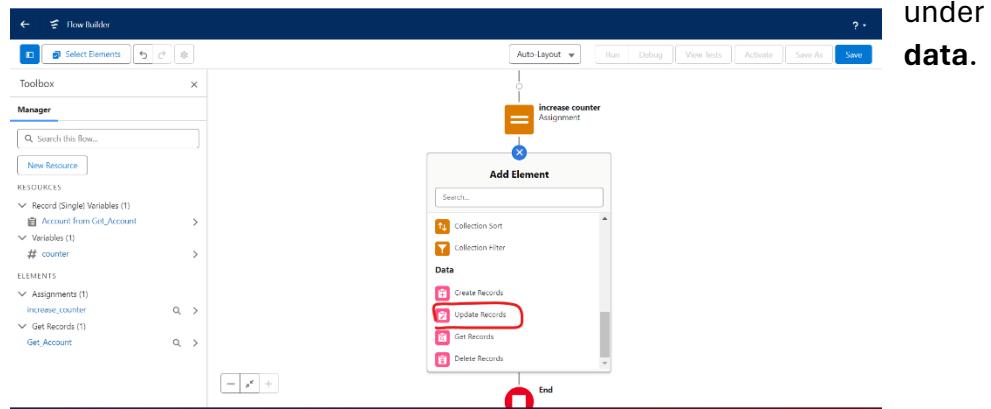
- After that select assignment under logic.



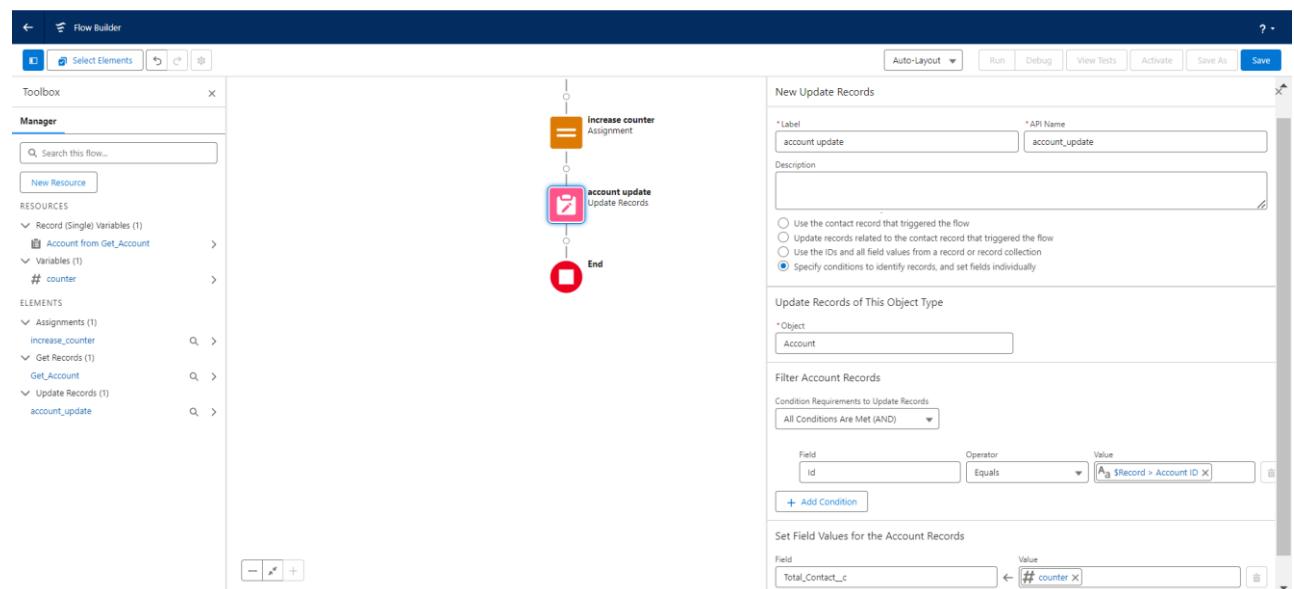
- After that enter name and api name. and set the variable.



- After that click on + icon below increase counter and select **update record**



- After that enter label and api name. choose specify conditions to identify records and set fields individually. After that select object account. Then enter filters for account records. Now set the field value for the account records.



- Now save the flow and activate it. Now go to front-end of the app and check.

Day – 28

Record-Trigged Flow -2

Scenario:

- Upon Account Phone update, update all related contacts phone number as well.
 - Without using loop elements.
 - Using loop elements.
- When an opportunity is set as closed won/lost and Account is populated the post below message as chatter on the opportunity: [using chatter post action]

@[account owner] opportunity is closed won/lost.
Closed date: show close date of opportunity.
Amount: show amount of opportunity

1. Upon Account Phone update, update all related contacts phone number as well.

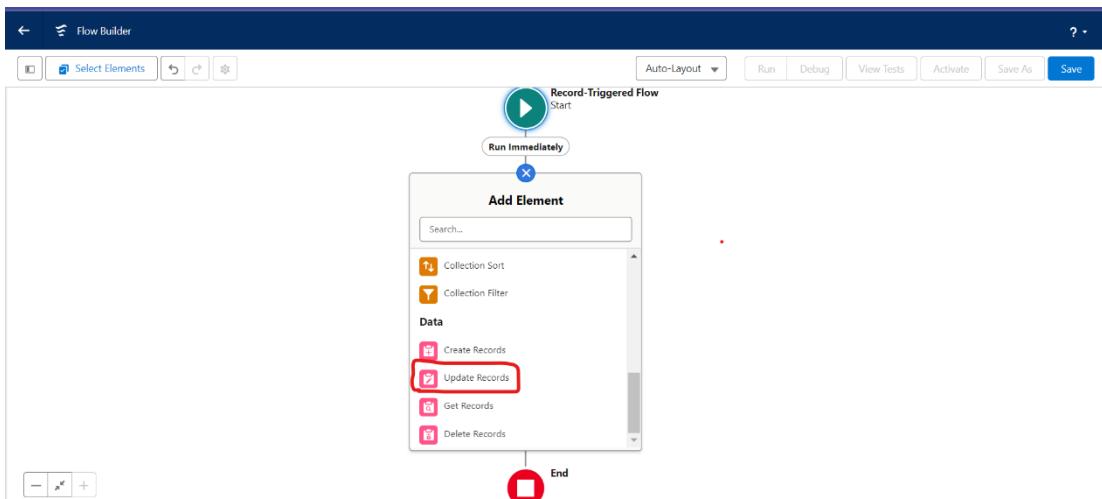
- Without using loop elements.
- Using loop elements.

- Steps for Without using loop elements.**
- Click on set-up and search flows in quick search after that click on new flow.
- After that select record triggered flow and click on create.
- After select object account and configure trigger is **a record is updated** because we update the phone number.
- In entry conditions select And and phone number is changed to true. And choose account and related accounts.

The screenshot shows the configuration steps for a Record Triggered Flow:

- Select Object:** Account
- Configure Trigger:** A record is updated
- Set Entry Conditions:** Field: Phone, Operator: Equals, Value: True
- When to Run the Flow for Updated Records:** Every time a record is updated and meets the condition requirements
- Optimize the Flow for:** Fast Field Updates, Actions and Related Records
- Include a Run Asynchronously path:** (checkbox)

- After that click on + icon and select update records from the **Data** section.



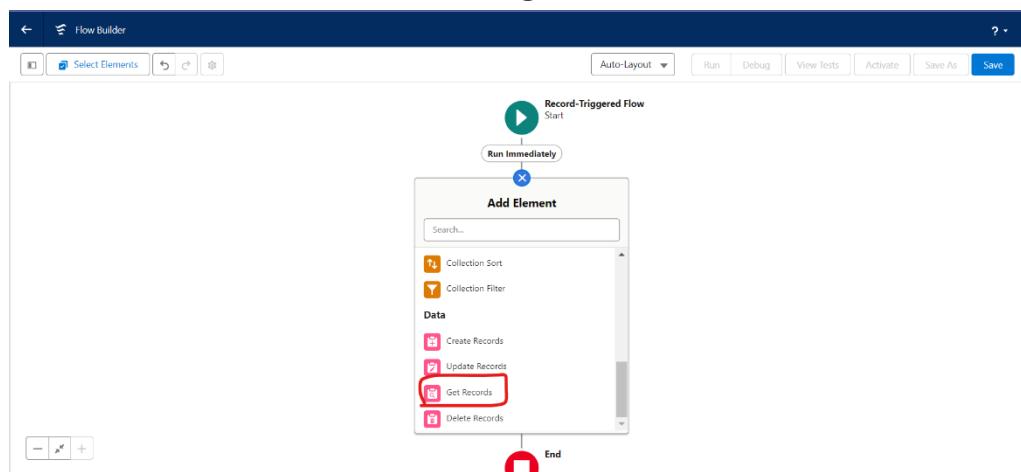
- After that enter label and Api name and select Specify conditions to identify records and set fields individually.
- Now select the contact object because we want to update the contacts phone number updated.
- After that set filter contact record. Which is accountid equal to \$Record>account id.
- After that set field values for the contact records. field is mobilephone and value is \$Record and the account phone.

This screenshot shows the configuration details for the 'update contacts' update record flow. At the top, it displays the label 'update contacts' and the API name 'update_contacts'. Below this, the 'Object' is set to 'Contact'. A condition is defined: 'Specify conditions to identify records, and set fields individually' with the filter 'Accountid Equals \$Record > Account ID'. In the 'Set Field Values for the Contact Records' section, the 'MobilePhone' field is mapped to the value '\$Record > Account Phone'.

- Now save the flow and activate it.
- Now go the app and see the result that change the phone of any account records its related contact phone also updated.

- **Using loop elements.**

- 1st create a triggered flow same as before.
- After that click on + icon and select **get record** from data **section**.

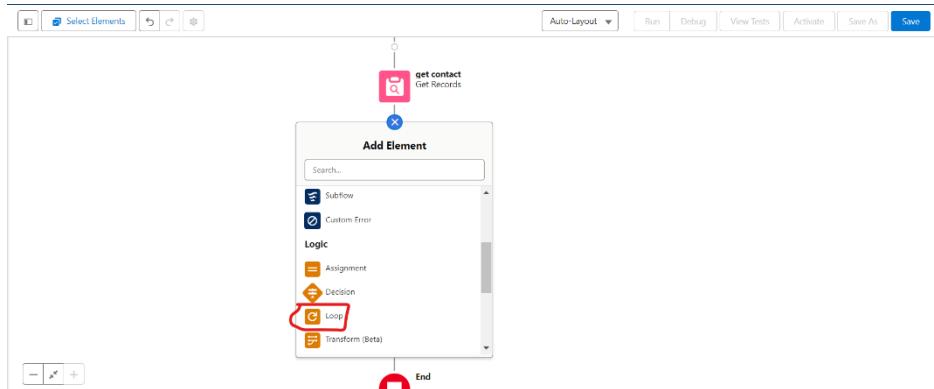


- After that enter label and api name and select contact object. After that we have to specify the condition. Condition should be accountid equal to \$record and accountid.
- In how many records to store select **All records**.

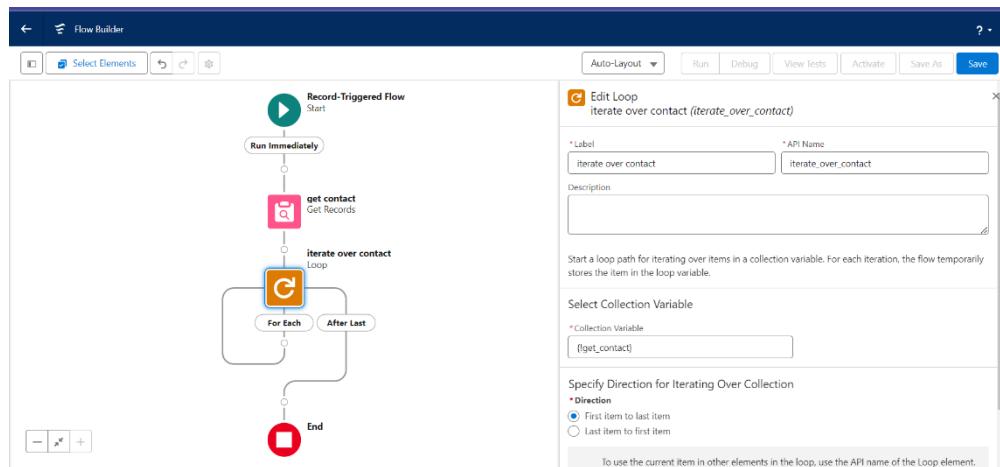
The screenshot shows the 'New Get Records' configuration page. It includes the following sections:

- New Get Records**: Fields for Label (get contact) and API Name (get_contact).
- Get Records of This Object**: Field for Object (Contact).
- Filter Contact Records**: Condition Requirements dropdown set to "All Conditions Are Met (AND)". A condition row is shown with Field: AccountId, Operator: Equals, Value: \$Record > Account ID.
- Sort Contact Records**: Sort Order dropdown set to "Not Sorted". A note says: "⚠ If you store only the first record, filter by a unique field, such as ID."
- How Many Records to Store**: Radio button selected for "Only the first record".
- How to Store Record Data**: Radio button selected for "Automatically store all fields". Other options include "Choose fields and let Salesforce do the rest" and "Choose fields and assign variables (advanced)".

- Now we are going to implement loop. For this click on + icon below get contact and select **loop** under **logic** section.



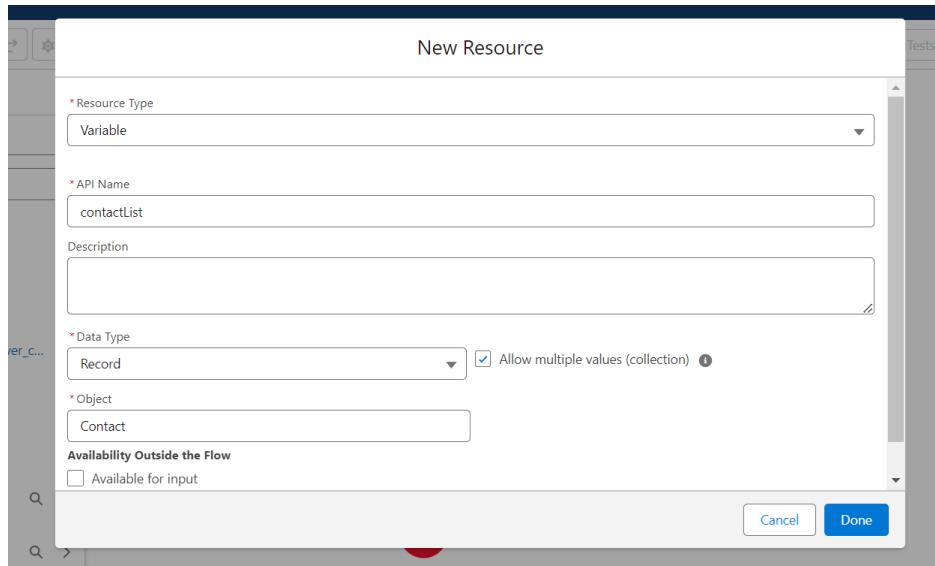
- After that enter label and Api name. after that enter the collection variable get contact which we create earlier.



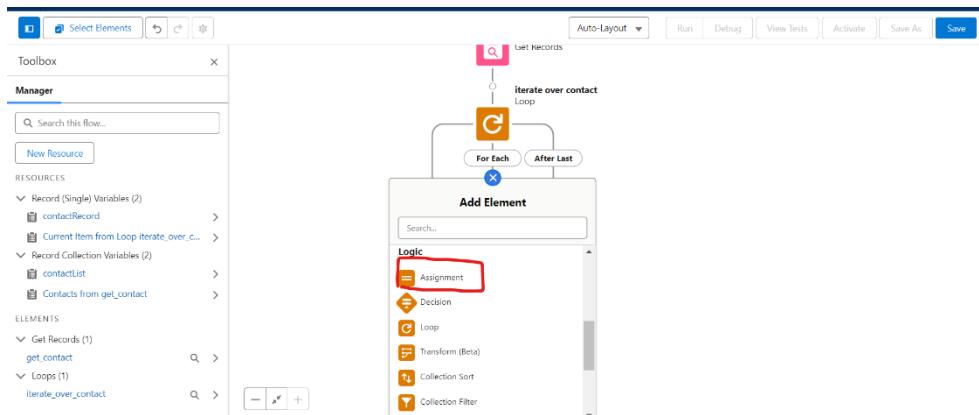
- After that create 2 variables. For this click on new resources and select variable.
- After that enter the api name contactRecord and data type is record and object is contact. and click on done.

The screenshot shows the 'New Resource' dialog. It has fields for 'Resource Type' (set to 'Variable'), 'API Name' (set to 'contactRecord'), 'Description' (empty), 'Data Type' (set to 'Record'), 'Object' (set to 'Contact'), and a checkbox for 'Allow multiple values (collection)' which is unchecked. At the bottom, there are 'Cancel' and 'Done' buttons.

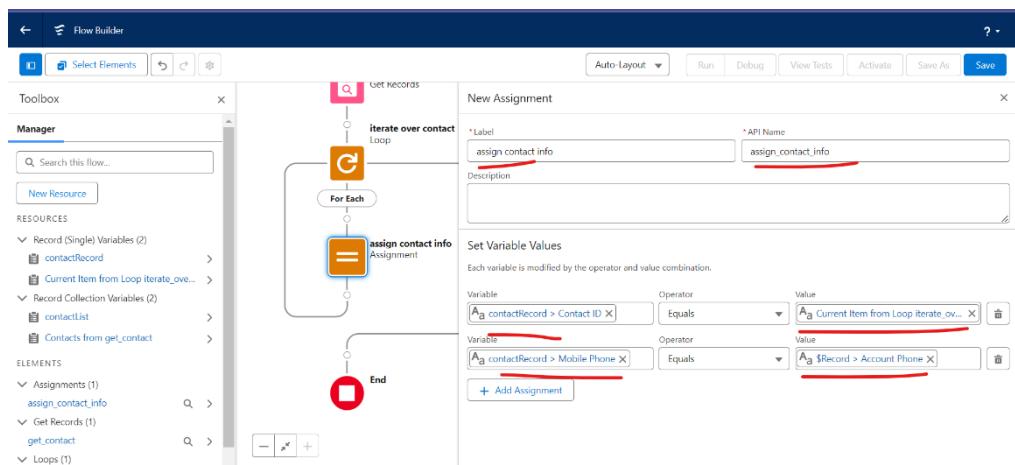
- After that create one more variable. Name contactList data type is record and object will be contact and check the box allow multiple value. And click on done.



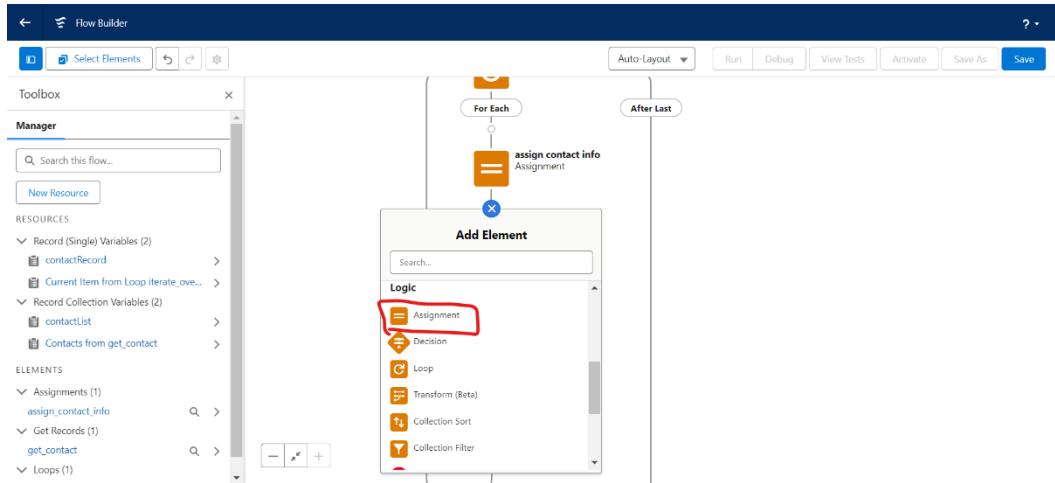
- After that click on + icon below For Each and select **assignment** under logic section.



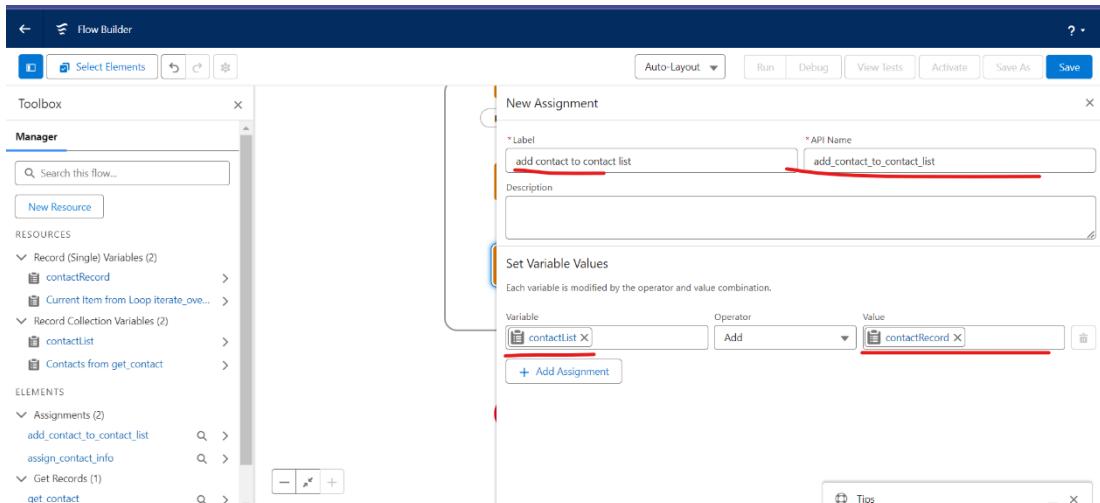
- after that enter label and api and enter below details and click on done.



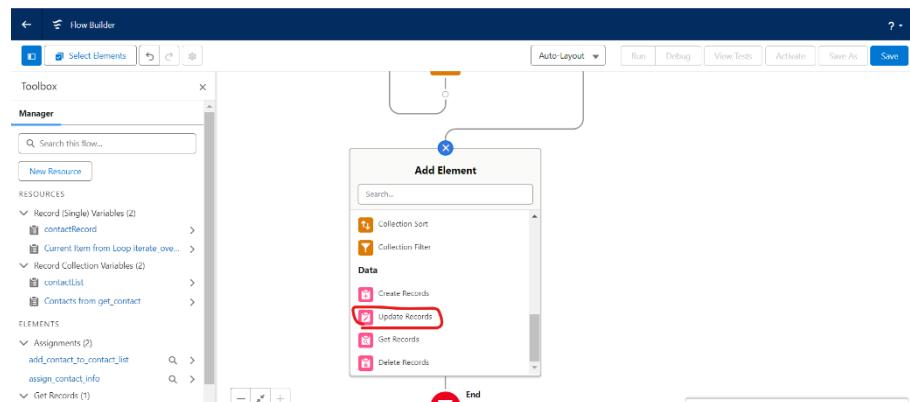
- again, click on + icon below the + icon of 1st assignment and choose assignment from logic section.



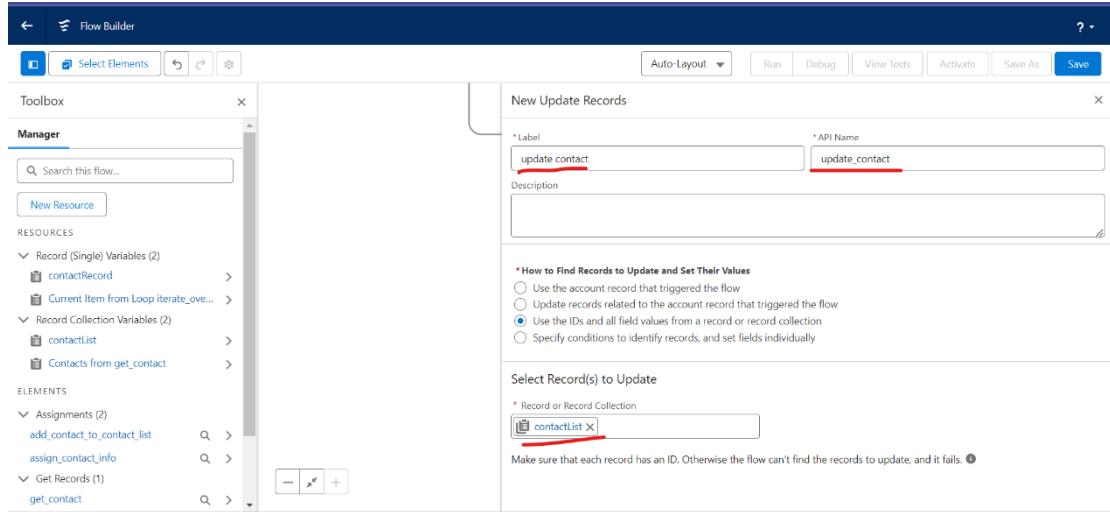
- After that enter label and api name and enter below details as shown in image.



- Now flow is created and this loop will iterate based on the contact in account object if 3 contact so it iterate 3 time.
- Now click on + icon just above the end and select **update record** from **data** section.



- Enter label and api. After that enter details as shown in below.



- Now click on save and activate the flow.
- Now go to front-end and check the results that when phone number of account is changed so the related contact number is also changed.
- When an opportunity is set as closed won/lost and Account is populated the post below message as chatter on the opportunity: [using chatter post action]**

@ [account owner] opportunity is closed won/lost.
Closed date: show close date of opportunity.
Amount: show amount of opportunity.
- Go to the set-up and search for flows and click on new flow. After that select record-triggered flow.
- After that select object as opportunity.
- In configure trigger choose A record is created or updated.

Or were you:

*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

Custom Condition Logic Is Met

*Condition Logic (1 OR 2) AND 3

Field	Operator	Value
1 StageName	Equals	Closed Won
2 StageName	Equals	Closed Lost
3 AccountId	Is Null	False

+ Add Condition

When to Run the Flow for Updated Records

- Every time a record is updated and meets the condition requirements
- Only when a record is updated to meet the condition requirements

*Optimize the Flow for:

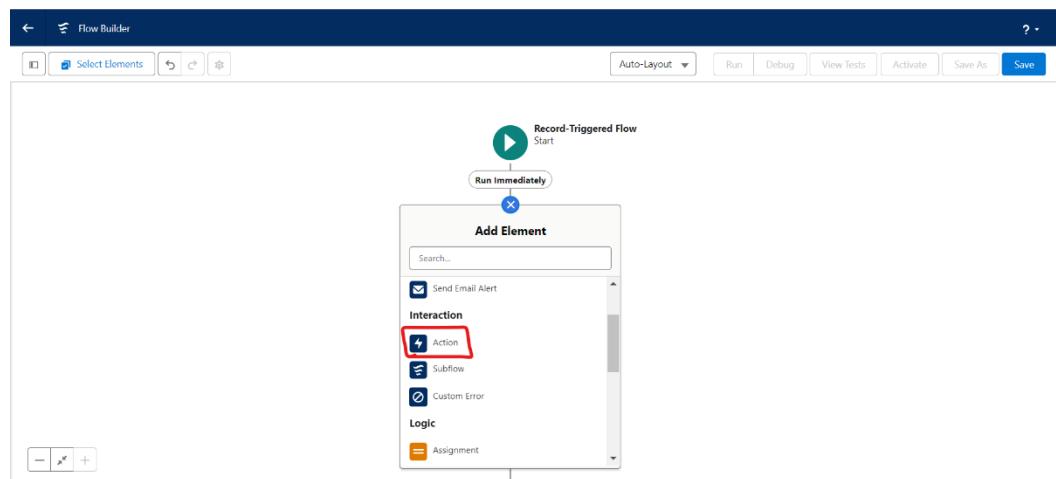
Fast Field Updates

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

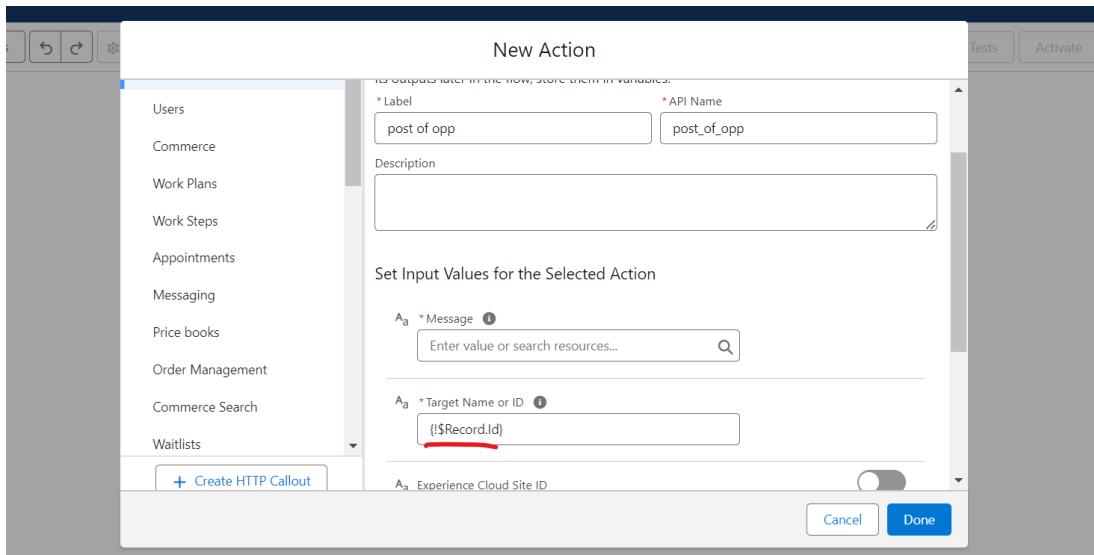
Actions and Related Records

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

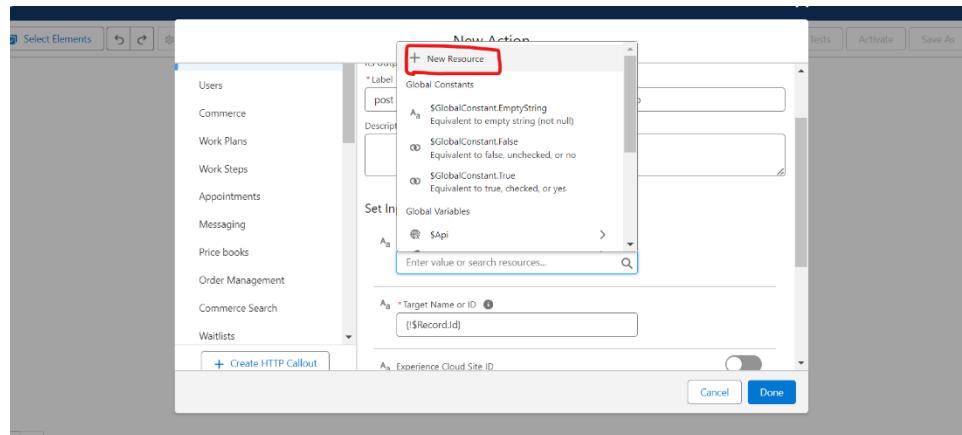
- After that click on + icon and select action from interaction section.



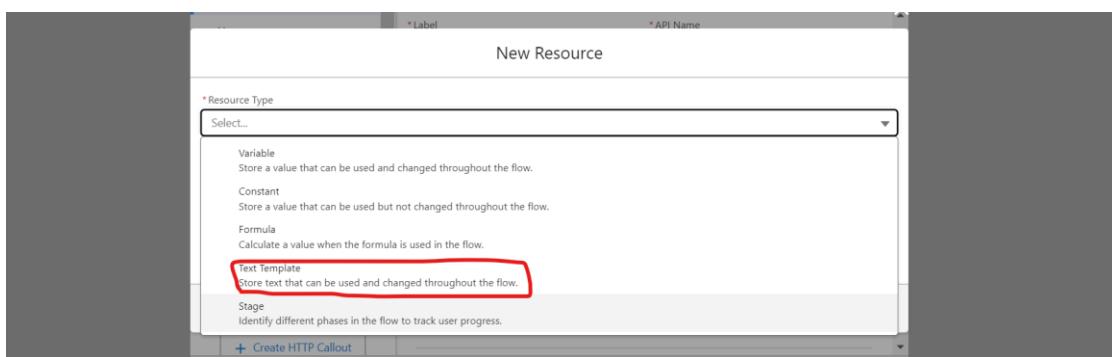
- Search for post to chatter in action search box.
- After that enter label and Api name.
- After that enter the target id from where we want to post that chatter message so choose opportunity id. (\$record then id).



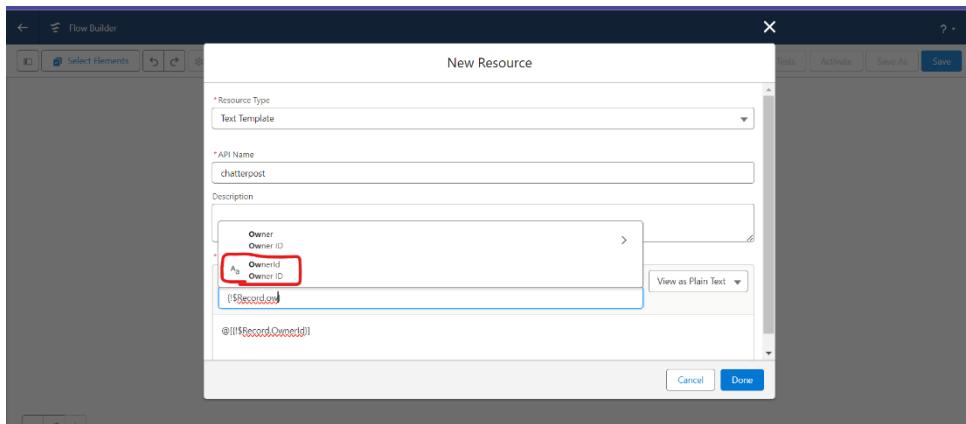
- After that in message we have to create message. For this click new resources inside the message.



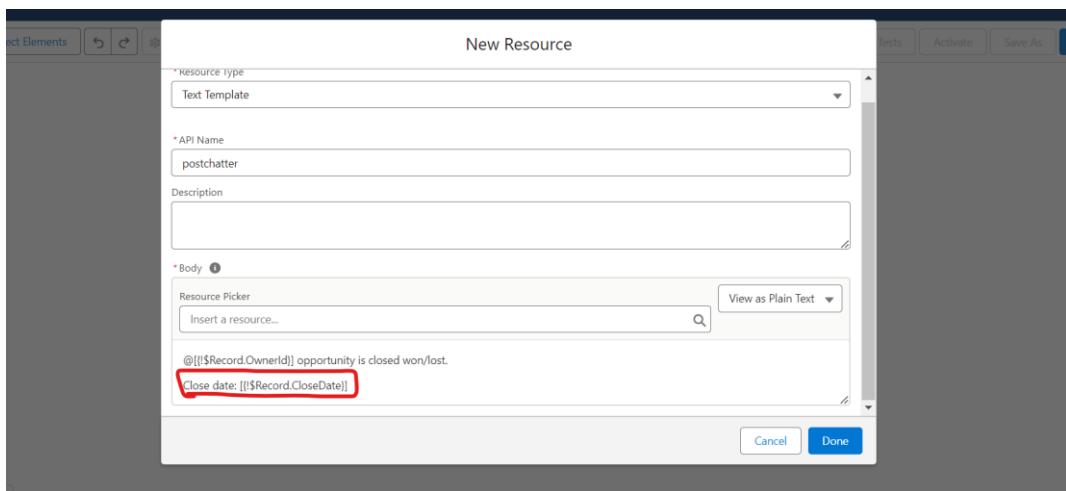
- After that select text templates from the drop down.



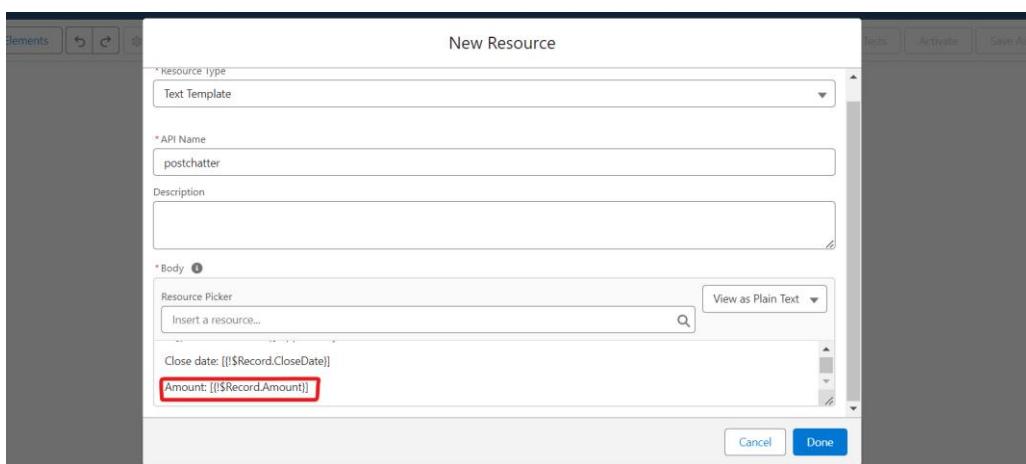
- After that enter the Api name and select view as plain text. After that enter the msg in body.
- To add account owner simply enter @*[search \$record(opportunity) then ownerId].



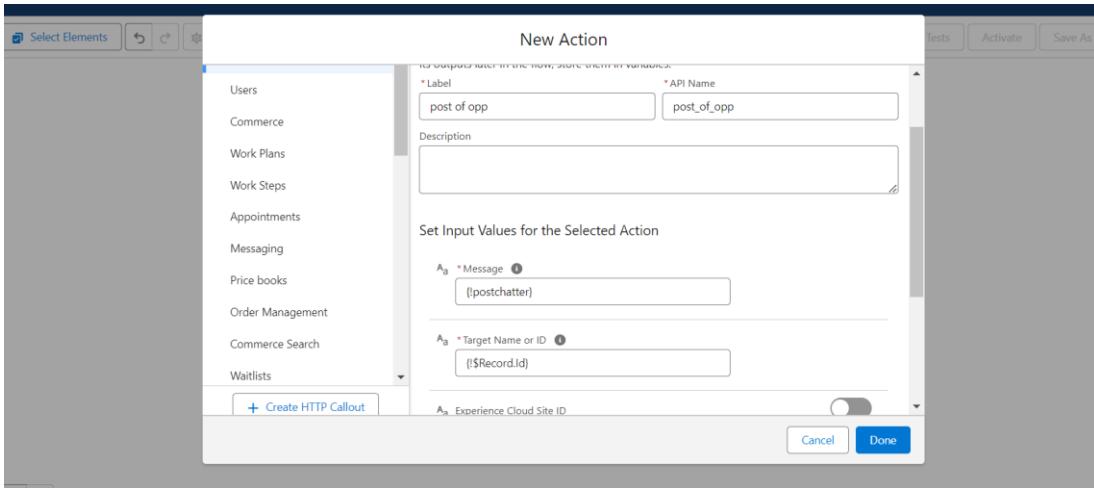
- After that enter the msg **opportunity is closed won/lost**.
- After that enter close date from the opportunity [`[$record>opportunity then close date]`].



- After that we have to enter amount and fetch amount from opportunity. [`[$Record>opportunity then amount]`].



- Now message is created click on done.



- Now save the for and activate it and text the result on App.

Opportunity
active will be yes

Share an update... Share

Sort by: Most Recent Activity

Zahid Khan Just now

@Zahid Khan opportunity is closed won/lost.
Close date: [25 January 2024]
Amount: [1,00,000]

Like Comment

1 view

Day – 29

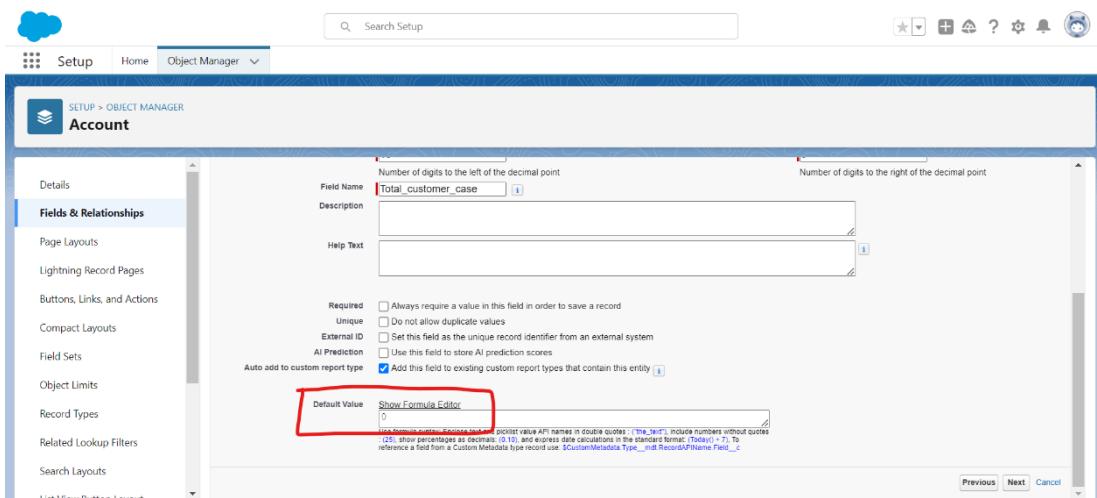
Triggered flow – 2.

Scenario:

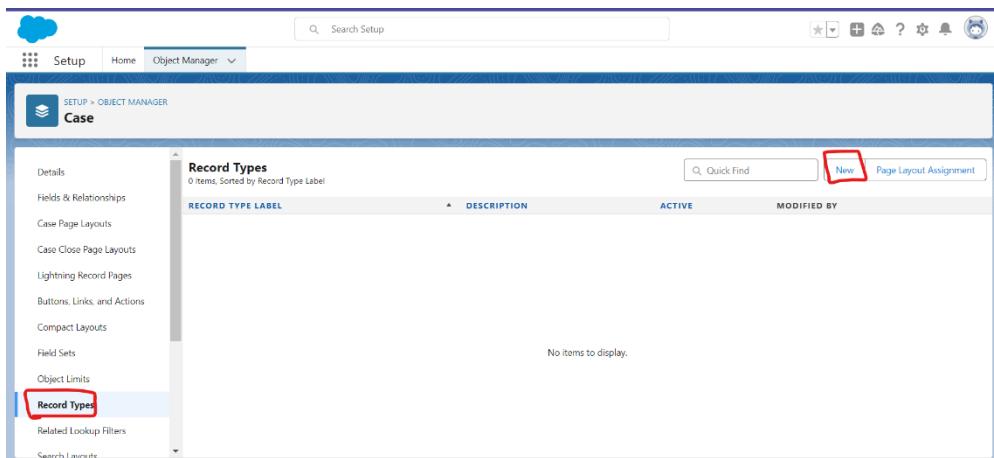
- Create two record types names as “partner case” and “customer case” on case object. On creation case populate the total number of partner case and customer on account object. create custom fields labeled “total customer case” and total partner case” on account.
- Delete the opportunity when its stage is set to closed-lost.
- Apply validation rule based on existing record data using record triggered flow.

1. Create two record types names as “partner case” and “customer case” on case object. On creation case populate the total number of partner case and customer on account object. create custom fields labeled “total customer case” and total partner case” on account.

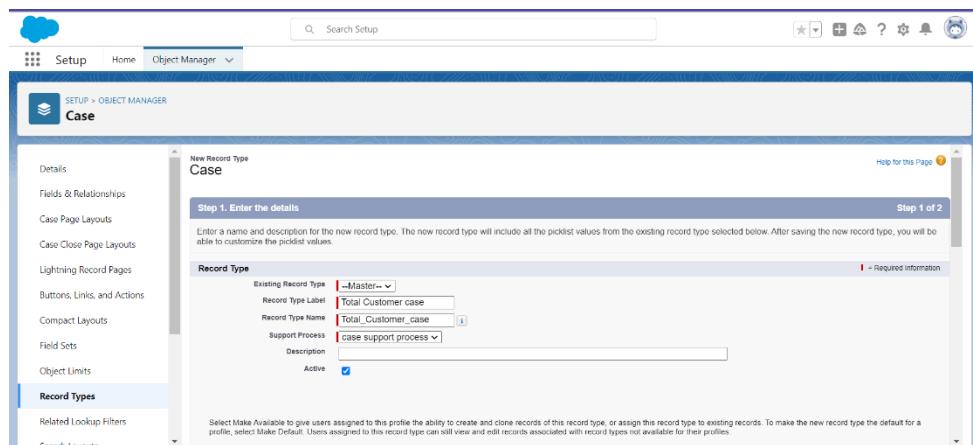
- 1st create total customer case and total partner case field in account object.
- Make sure that default value should be 0 for these two fields.



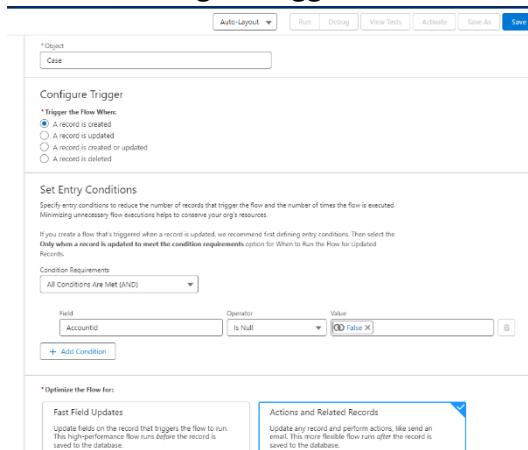
- After that go the case object from object manager and click on record types to create two records and click on new.



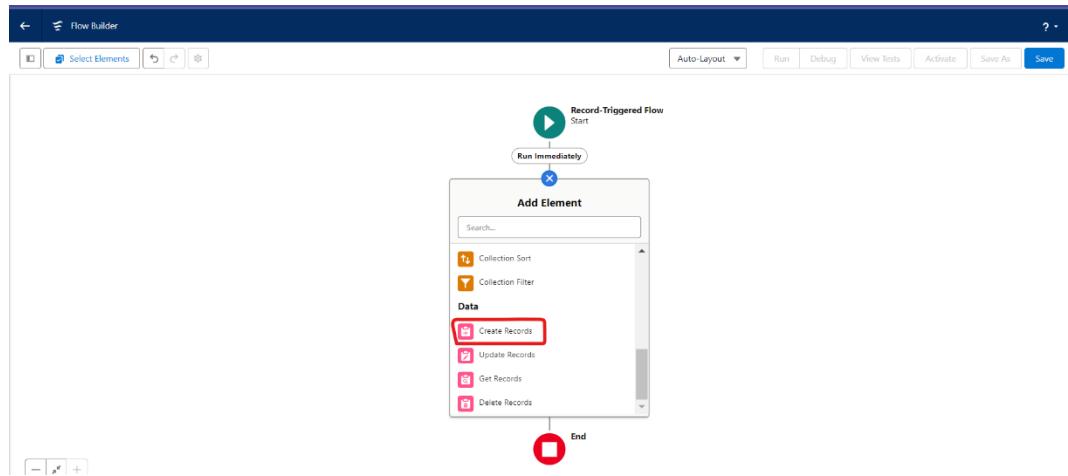
- Create support process before creating record type.
- After that enter the label and support process and click on next.



- after that select case layout and click on save.
- Same way creates select total partner case record types.
- After that go to home and search flows in quick find and click on new flow and select record-triggered flow.
- After that select object case, configure trigger “A record is created”



- After that click on + icon and select **create record** from Data.



Apex

- Object oriented programming language.
 - Supports classes, interfaces and inheritance.
 - Uses java like syntax.
- Strongly types.
 - Validates references to objects at compile time.
- Integrated with the databases.
 - Provides direct access to records and their fields.
- Enable developers to add business logic to system events, including button clicks, related record updates, visual Force pages and lightning components.
- One can call apex code through web service request and triggers the object