P08: Auto-Populating Forms with Power Automate

Thursday, March 13, 2025 12:32 PM

Context

Item	Detail
Tag	Vertical Patterns | Power Apps
Contributors	Vincent Kimiti | Habab Siddig
Version Control	V1.0
Application Use Case	Auto-populate an "Authorised Individuals" form with account data from Dataverse
Reference Usage	Automating form population using Dataverse and string manipulation in Power Automate

Story Behind The Pattern

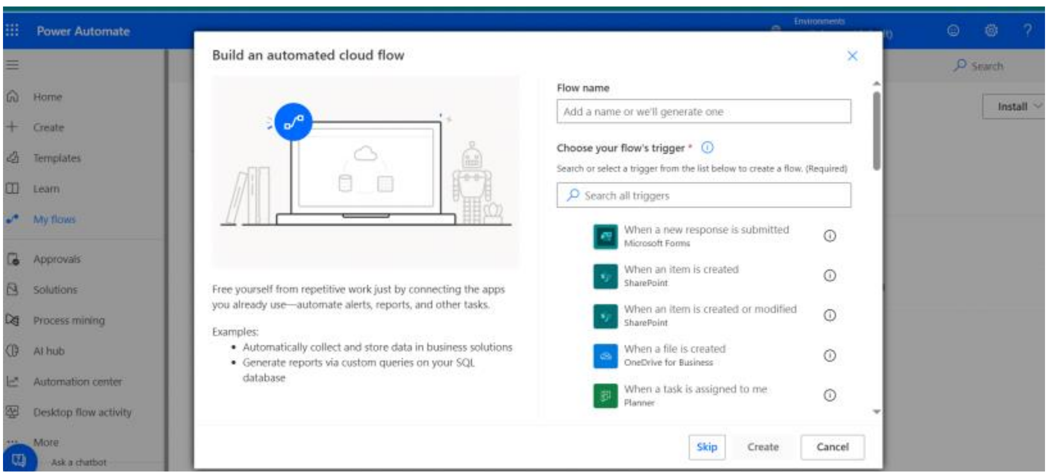
Item	Detail
The Problem	Manually updating the "Authorised Individuals" form with related account data is inefficient and error-prone.
The Solution	Create a Power Automate flow to retrieve row data, initialize a variable, list related accounts, and update the form with processed information.
Dependencies	Power Automate environment, Dataverse with "Authorised Individuals" and "Accounts" tables, permissions to create flows and access tables.

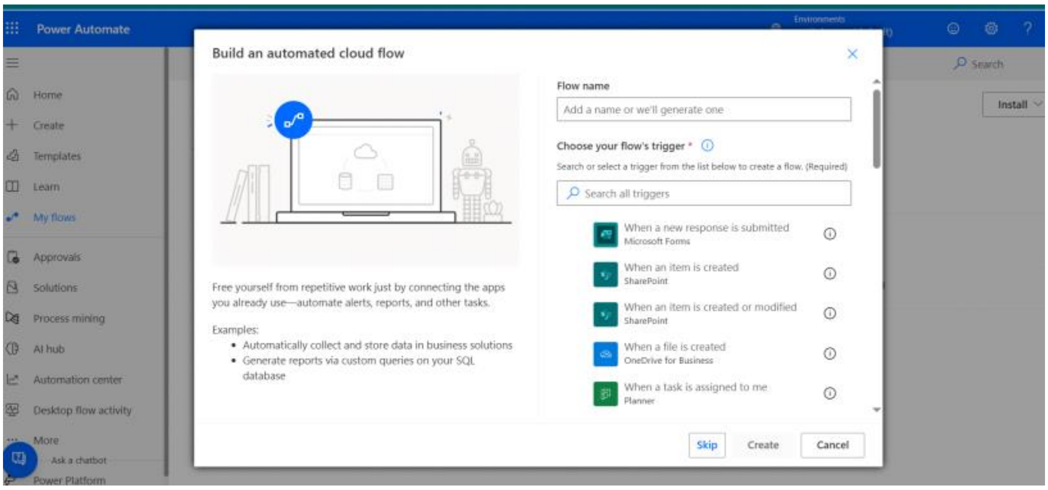
The Pattern

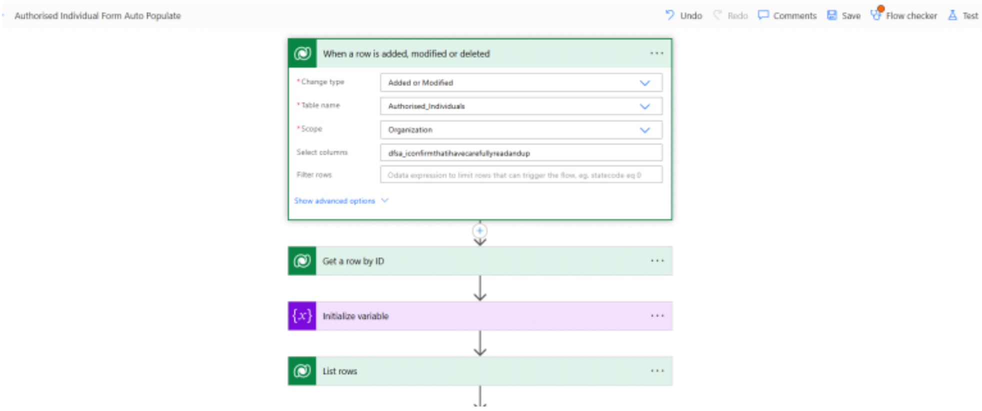
Method 1: Setting Up Authorised Individual Form Auto-Population Flow

Here's a step-by-step guide to create a Power Automate flow that auto-populates an "Authorised Individuals" form by processing account data from Dataverse:

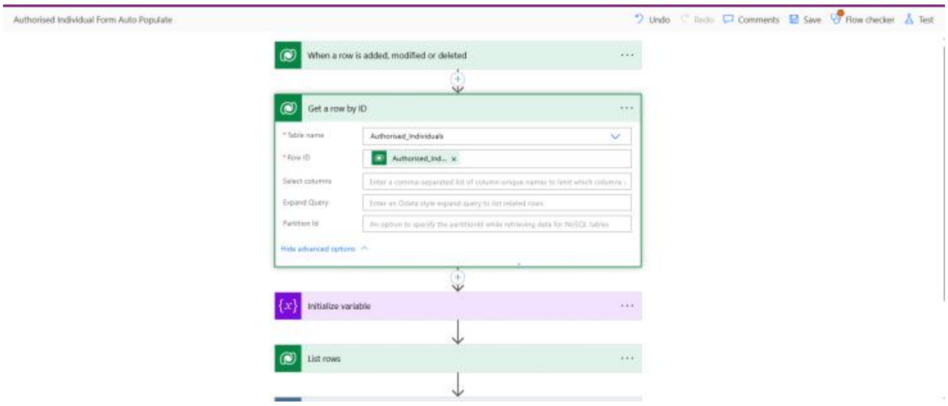
1. Create a New Flow in Power Automate
   - Log in to the Power Automate portal (make.powerautomate.com).
   - Click Create and select Automated cloud flow.
   - Name the flow (e.g., "Authorised Individual Form Auto Populate") and click Create.
   - Choose the trigger When a row is added, modified or deleted from the Dataverse connector.

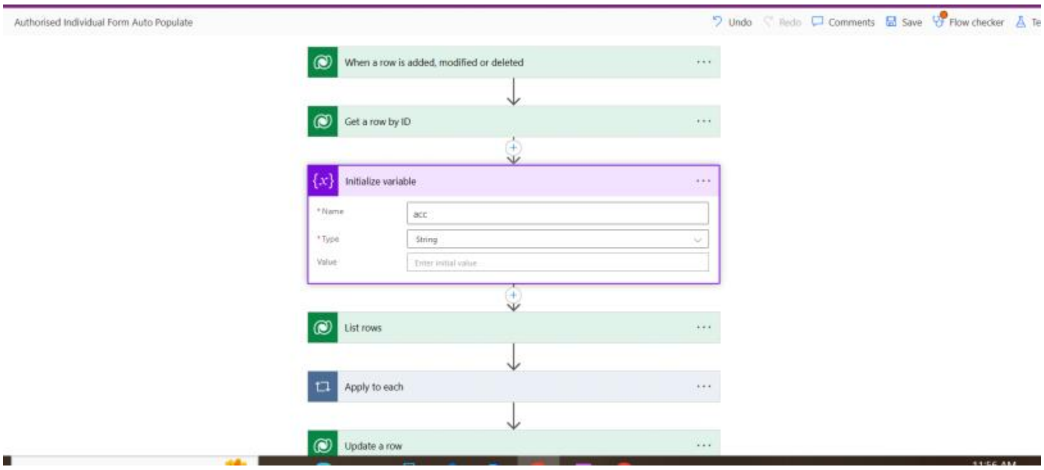




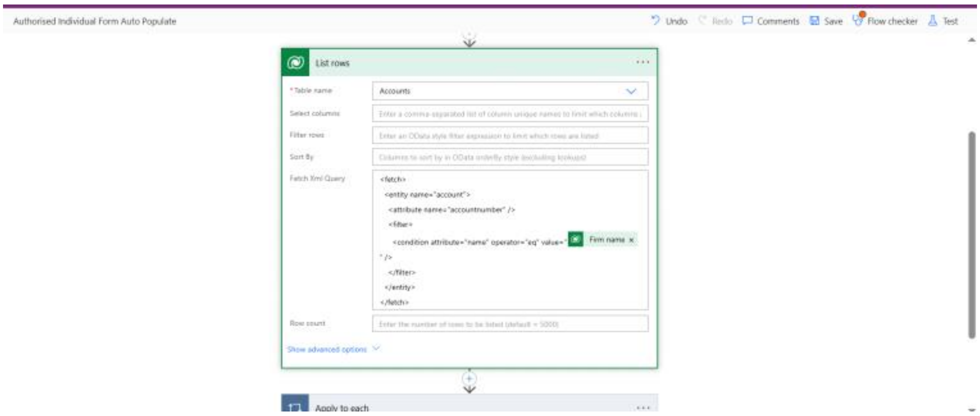
2. Configure the Trigger
   - In the trigger settings, set the following:
     - Change type: Select "Added, Modified" (to trigger on new or updated rows; exclude "Deleted" if not needed).
     - Table name: Select "Authorised Individuals" from the dropdown.
     - Scope: Choose "Organization" to apply the trigger across the organization.
     - Select columns: Leave blank or enter specific columns (e.g., "AuthorisedInd_Id") to optimize performance.
     - Filter rows: Optionally, add an OData expression (e.g., statuscode eq 1) to limit triggers to active records.
   - Click Save to proceed.
   - 

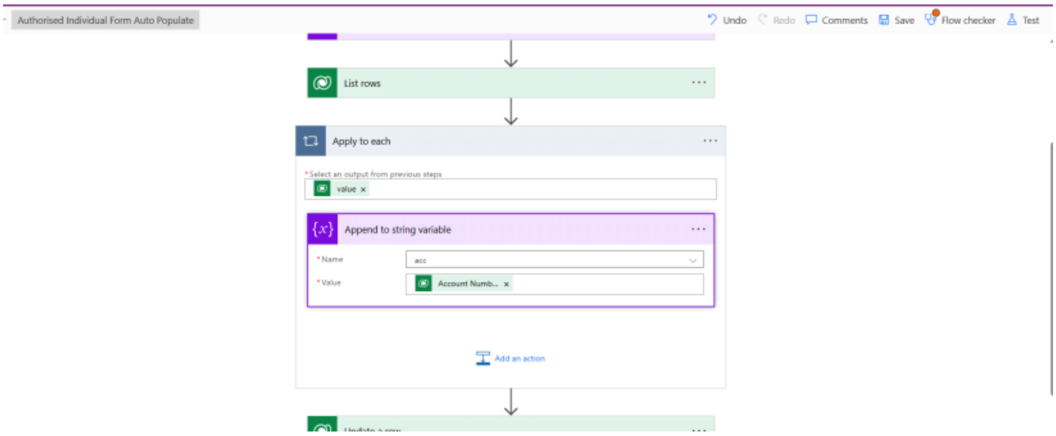
3. Add a Get a Row by ID Action
   - Click + New step below the trigger.
   - Search for and select Get a row by ID (Dataverse connector).
   - Configure the action:
     - Table name: Select "Authorised Individuals."
     - Row ID: Select the "Row ID" dynamic content from the trigger output.
     - Select columns: Leave blank or specify columns (e.g., "AuthorisedInd_Id") to retrieve specific data.
   - This step retrieves the detailed data of the changed row.

- 

4. Initialize a Variable
   - Click + New step.
   - Search for and select Initialize variable (Variables connector).
   - Configure the variable:
     - Name: Enter "acc" (or a meaningful name like "accountList").
     - Type: Select "String."
     - Value: Leave blank or enter an initial value (e.g., an empty string).
   - This step sets up a string variable to store concatenated account data.
   - 

5. Add a List Rows Action
   - Click + New step.
   - Search for and select List rows (Dataverse connector).
   - Configure the action:
     - Table name: Select "Accounts."
     - Select columns: Leave blank or enter specific columns (e.g., "Account Number," "Firm Name").
     - Filter rows: Enter an OData filter (e.g., FirmName eq 'Firm X') to limit rows based on the "Authorised Individuals" data (e.g., match Firm Name).
     - Sort by: Leave blank or specify a column (e.g., "Created On" descending).
     - Fetch XML Query: Optionally, use a custom FetchXML query (e.g., <fetch><entity name="account"><attribute name="accountnumber"/></filter><condition attribute="name" operator="eq" value="Firm X"/></filter></fetch>).
     - Row count: Leave as default (5000) or adjust as needed.
   - This step retrieves a list of accounts matching the criteria.

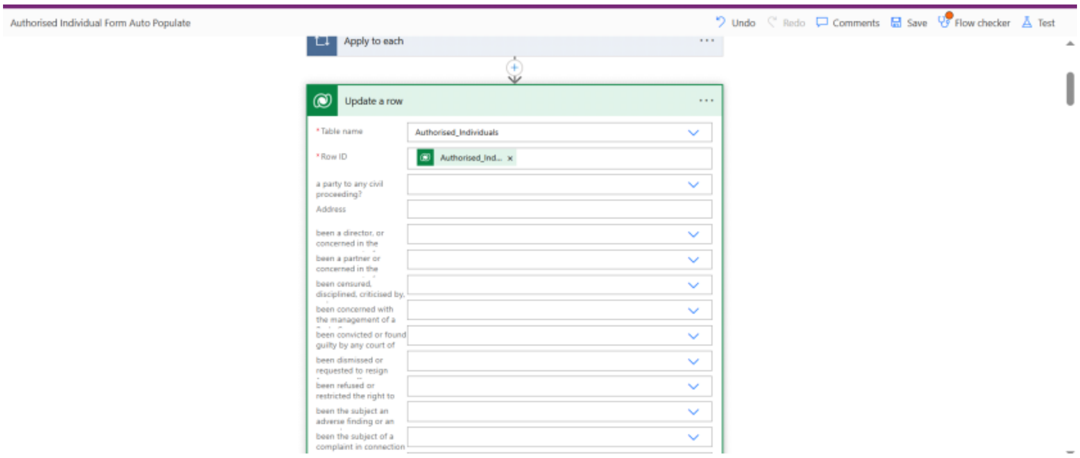
- 

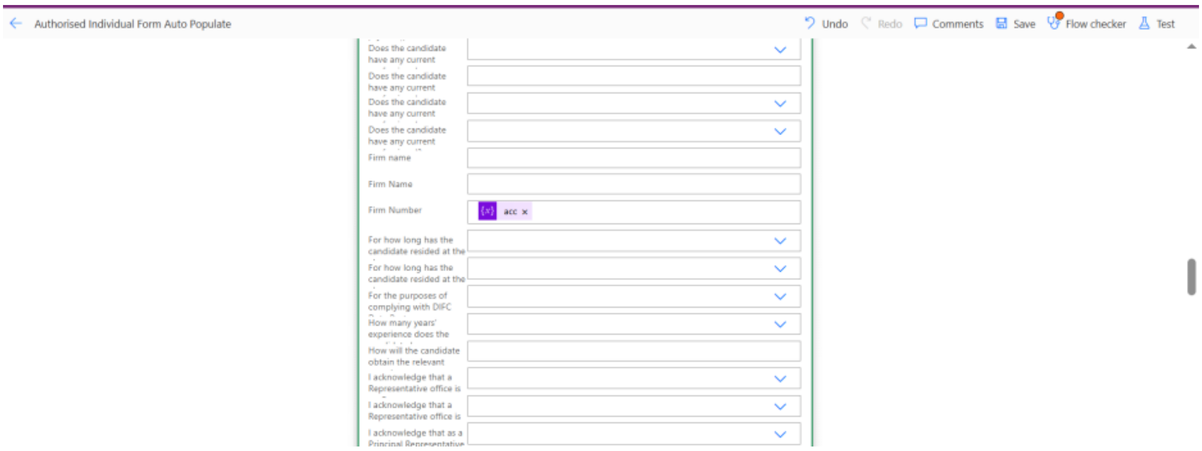
6. Apply to Each Loop
   - The "List rows" output will automatically create an Apply to each loop.
   - In the Select an output from previous steps field, choose the "value" output from the "List rows" action.
   - This step iterates over each account row retrieved.
   - 

7. Append to String Variable
   - Inside the Apply to each loop, click Add an action.
   - Search for and select Append to string variable (Variables connector).
   - Configure the action:
     - Name: Select "acc" (the variable initialized earlier).
     - Value: Select "Account Number" (or another field) from the "List rows" output and append it (e.g., concat(variables('acc'), item()?['accountnumber'], ', ') to add a comma-separated list).
   - This step builds a string of account numbers from the retrieved rows.

8. Update a Row
   - Inside the Apply to each loop, click Add an action.
   - Search for and select Update a row (Dataverse connector).
   - Configure the action:
     - Table name: Select "Authorised Individuals."
     - Row ID: Select the "AuthorisedInd_Id" from the "Get a row by ID" output.
     - Address a party to any civil proceeding?: Leave blank or map a field if applicable.
     - Address been a director, or been concerned in the management of a company?: Leave blank or map a field.
     - Address been a partner or been concerned in the management of a partnership?: Leave blank or map a field.
     - Address been censured, disciplined, criticised?: Leave blank or map a field.
     - Address been convicted of any offence?: Leave blank or map a field.
     - Address been guilty by any court of misconduct?: Leave blank or map a field.
     - Address been dismissed or requested to resign?: Leave blank or map a field.

        ▪ Address been refused or restricted in right to be a director?: Leave blank or map a field.
        ▪ Address been refused or restricted in right to practice or been the subject of an adverse finding or an adverse ruling?: Leave blank or map a field.
        ▪ Address been the subject of a complaint in connection?: Leave blank or map a field.
        ▪ Firm Name: Set to the concatenated "acc" variable (e.g., variables('acc')) to populate with account numbers.
        ▪ Firm Number: Leave blank or map a field if applicable.
    ○ Click Save to update the row with the processed account data.



    ○ 

9. Test and Validate the Flow
    ○ Click Test in the flow editor and select Manually.
    ○ Add, modify, or delete a row in the "Authorised Individuals" table and check if the flow triggers.
    ○ Verify that the "Firm Name" field is updated with the concatenated account numbers from the "Accounts" table.
    ○ Ensure no errors occur during execution.

10. Activate and Monitor the Flow
    ○ Once tested, click Save and then Turn on to activate the flow.
    ○ Monitor the flow runs in the Run history to ensure it executes without errors.
    ○ Adjust the flow (e.g., refine the filter or variable logic) if issues arise.