European
Power Platform
Conference

PowerUp!

WELCOME
TO

VIENNA 2025

JOIN THE
CONVERSATION

#EPPC25

Enterprise-Ready Flow Management: Error Handling, Monitoring, and Optimization

Shahid Mahmood Hussain

Lead Technical Advisor, Power Platform

Abakion A/S, Denmark

Enterprise-Ready Flow Management: Error Handling, Monitoring, and Optimization

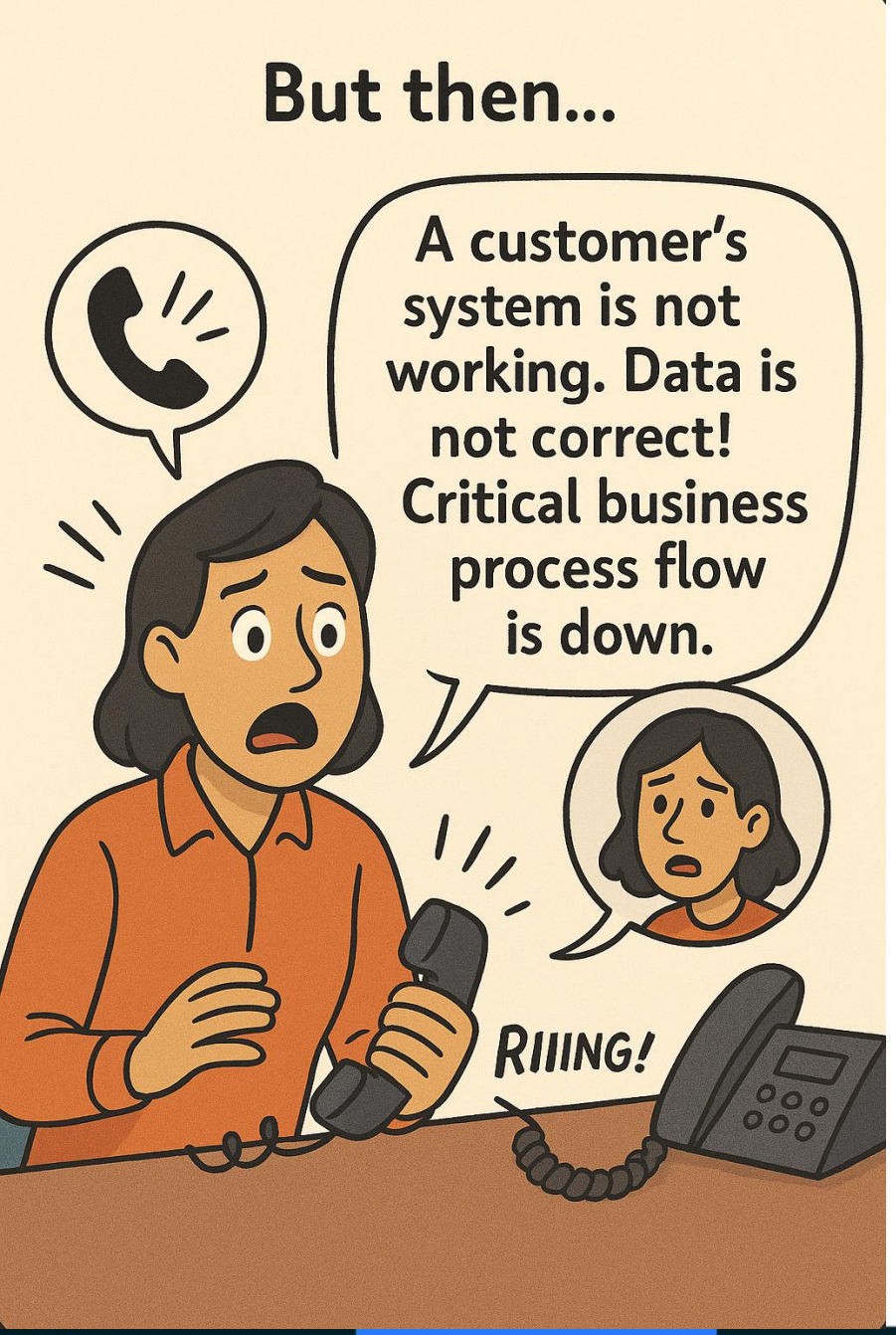
Michael Nielsen

Senior Developer,
Abakion A/S, Denmark

Let us try to imagine a scenario together…

✅ Successful deployment

Friday afternoon and successful deployment has been completed 🎯



⚠️ Critical Issue occur

"A critical business process flow has failed"

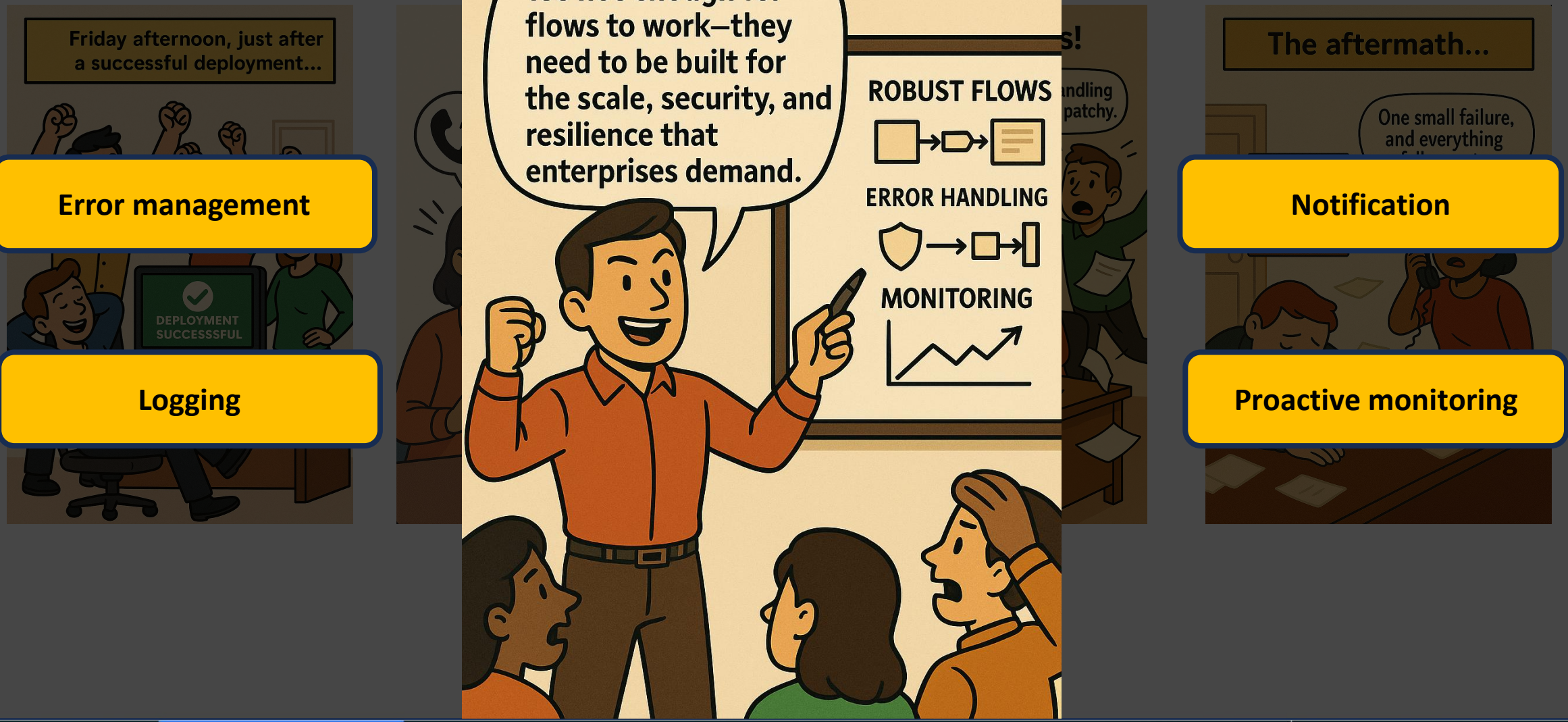
What would you do in this situation? 🤷

Everyone’s scrambling to find the root cause.

- 📄 The logs are incomplete
- ⚠️ Error handling? Patchy at best.
- 🔍 Monitoring?
- 👁️ Practically nonexistent

Catch Errors Before They Cause Chaos

Better error handling would’ve caught this early



Error management

Logging

Notification

Proactive monitoring

In this session, we'll explore how to elevate your flow management game:

🔍 How to design robust error handling so your flows can heal themselves—or at least tell you what went wrong.

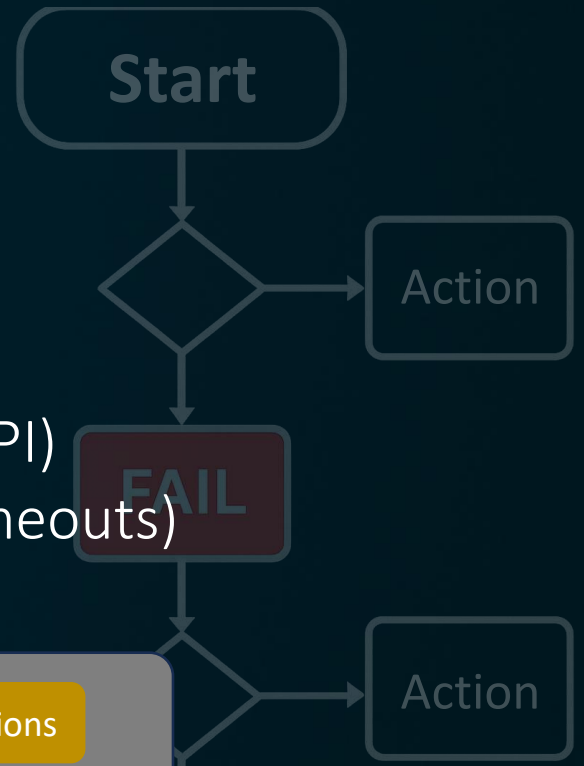
📊 How to implement real-time monitoring that keeps you ahead of failures instead of reacting after the damage is done.

⚙️ And finally, how to optimize your flows for performance and scalability, so they run smoothly no matter how complex your business grows.
Let's dive in and turn our flows into the reliable backbone of the modern enterprise.

Quick Raise of hands ✋

Why Error Handling Matters

- 2016 Power Automate GA
- Power Automate executes business-critical processes (ordre processing, Approval flow, moving data, Integrating with API)
- No process is completely error-free(API failures, data changes, timeouts)
- Unhandled errors lead to:



Business disruptions

- 🚧 Workflow chaos
- ⏳ Approval delays
- 🛑 Unseen Interruptions

Data inconsistency

- ⚖️ Asymmetrical Sync
- 🔀 Data incomplete
- ❓ Poor data

User frustration

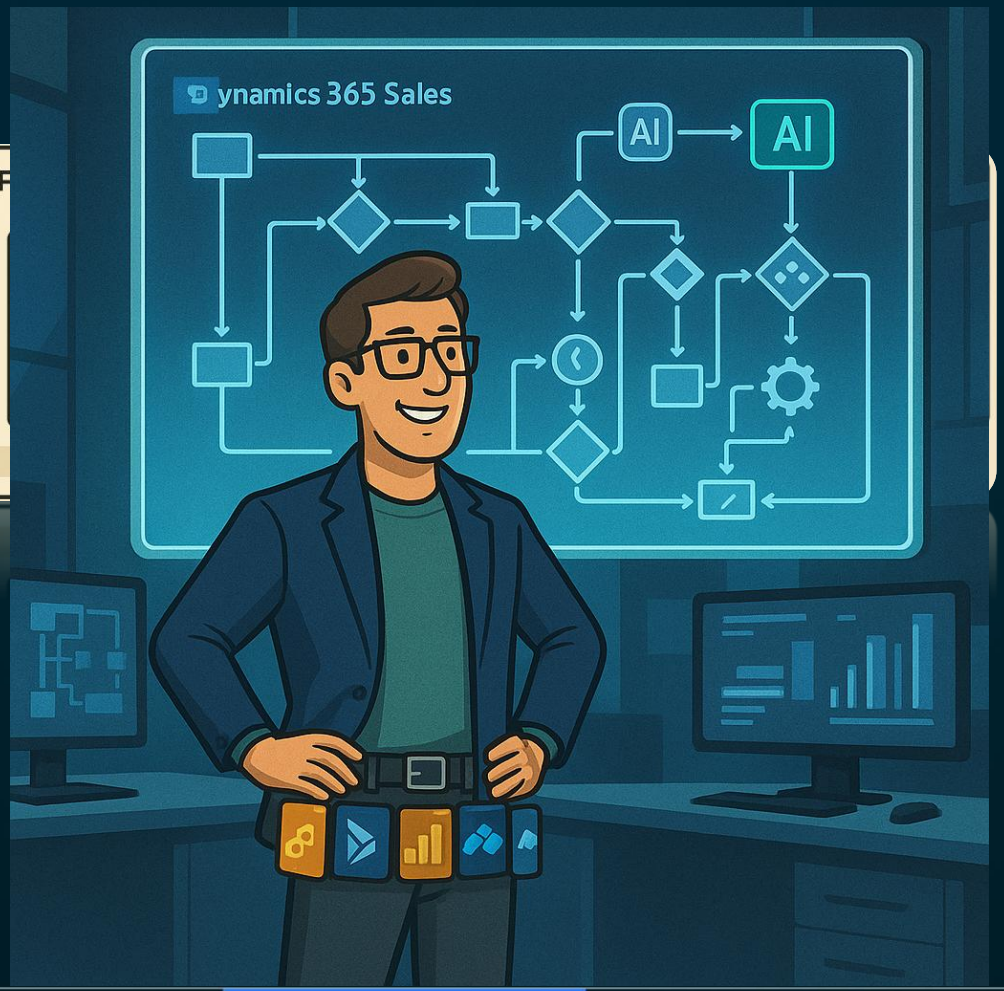
- 😕 Confused
- 💬 Where's My Notification?
- 🛠️ Manual work

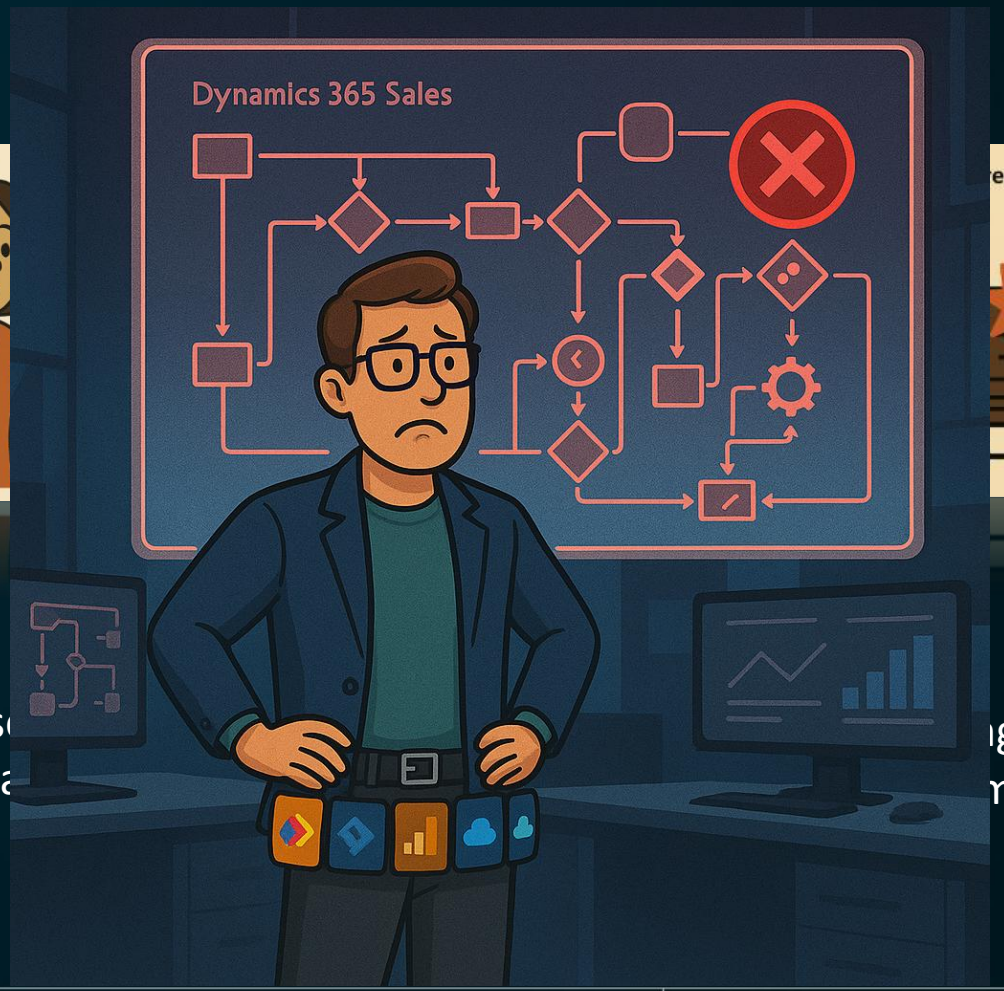
Hidden failures

- ❌ No Logs, No Clues
- 🔔 Failures in the Shadows
- 🧨 Quiet Breakage

Let’s imagine you’ve built a great flow...

What Happens Without Error Handling?





Common Pitfalls

Even the best flows can fail if you overlook common pitfalls.

Avoid failure by spotting these common mistakes early:

- No Configure Run After Conditions 🚫⚙️
- Assuming APIs or Connectors Always Succeed ✅🔗
- No Retry Policies Configured 🔁❌
- No Centralized Monitoring or Alerting 📡🔔

Solution

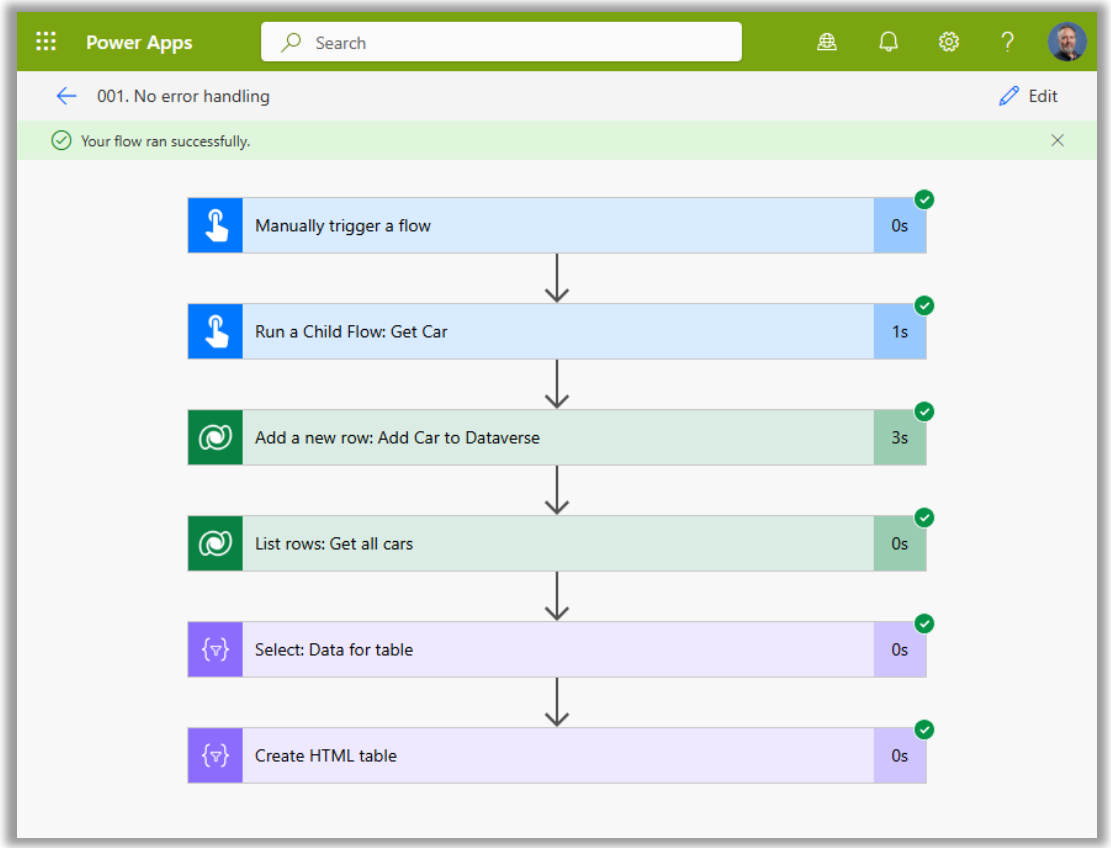
Use Try ➡️ Catch ➡️ Finally patterns using Scope actions and Configure run after settings

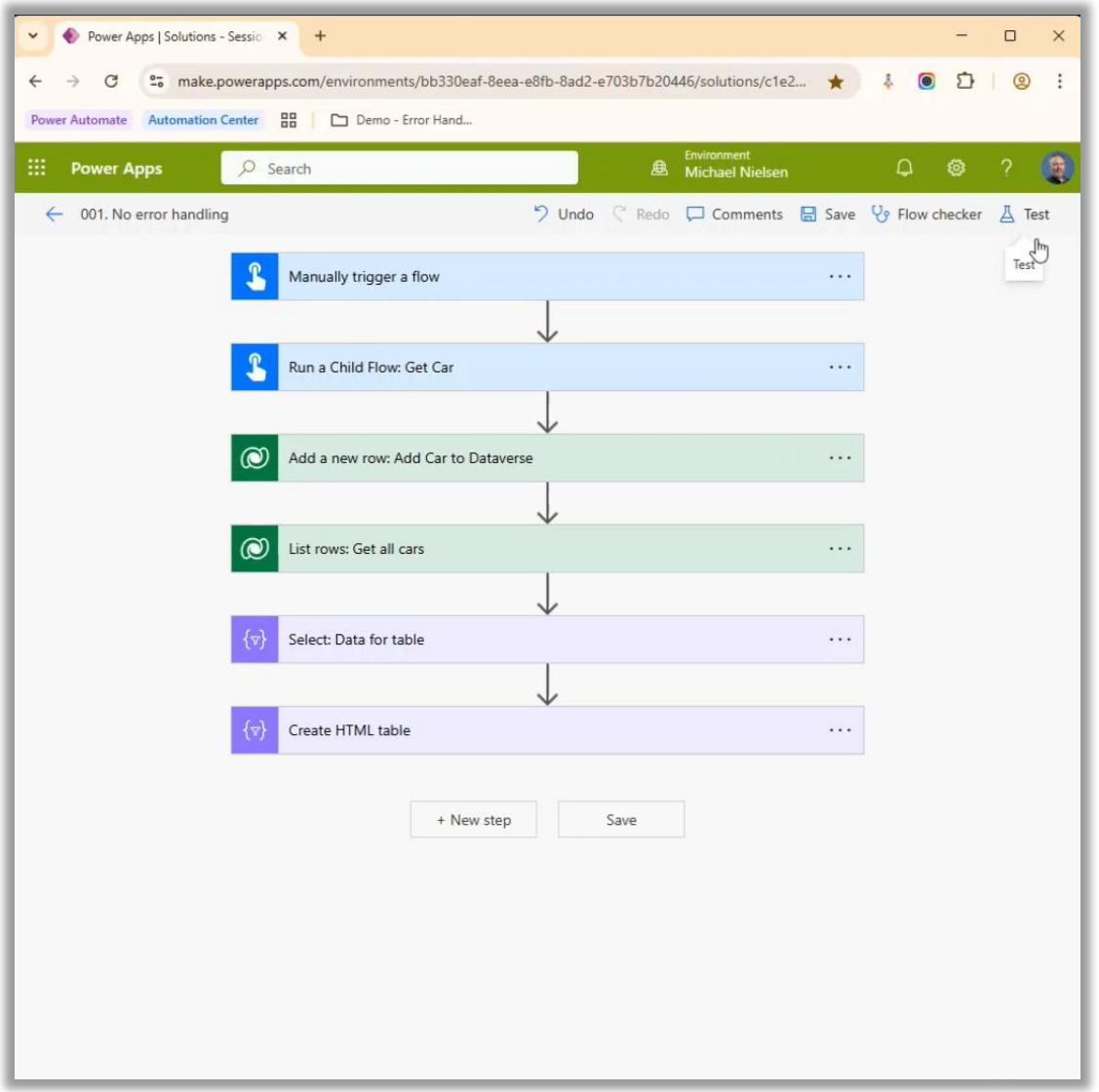
- Set up notifications on failure paths for immediate alerting
- Configure Retry policies with exponential backoff for transient errors
- Use Terminate actions to gracefully end flows with custom status and messages
- Design parallel branches for compensation or rollback mechanisms
- Add structured logging (Dataverse, SharePoint list, Azure Application Insights)
- Leverage variables to capture and pass error details for downstream diagnostics

Interactive Demo time

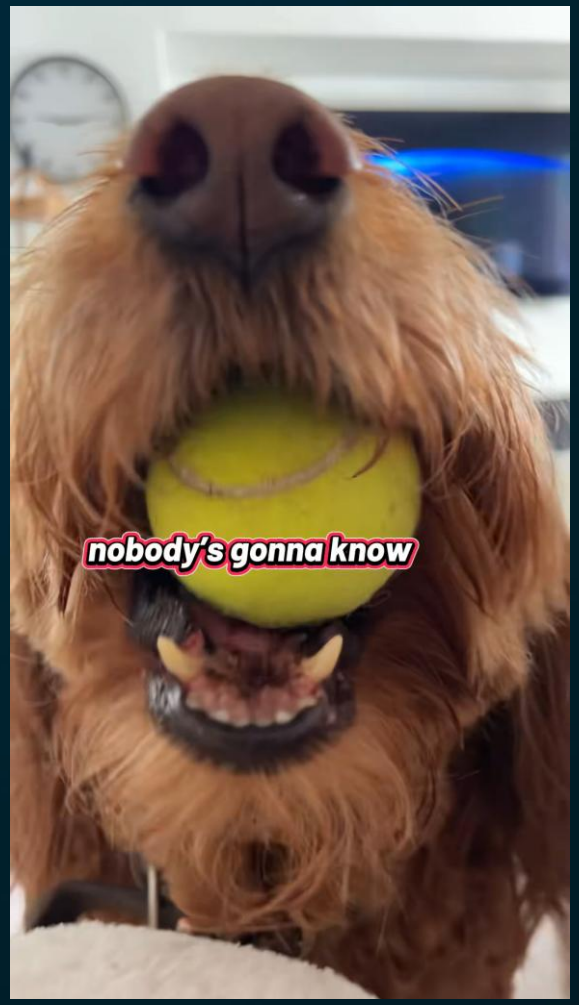
- Understand the importance of error handling
- Learn how to implement Try-Catch-Finally logic
- Implement custom error messages and notifications

Flow without error handling



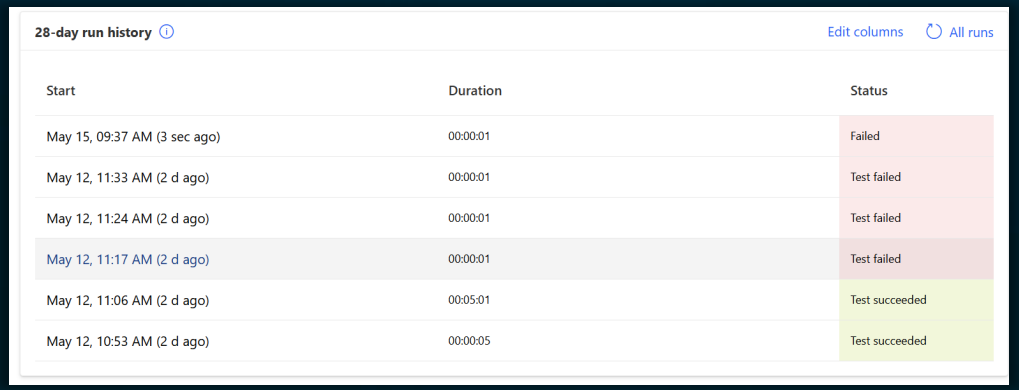


Who’s gonna know?



How will you know if a flow fails

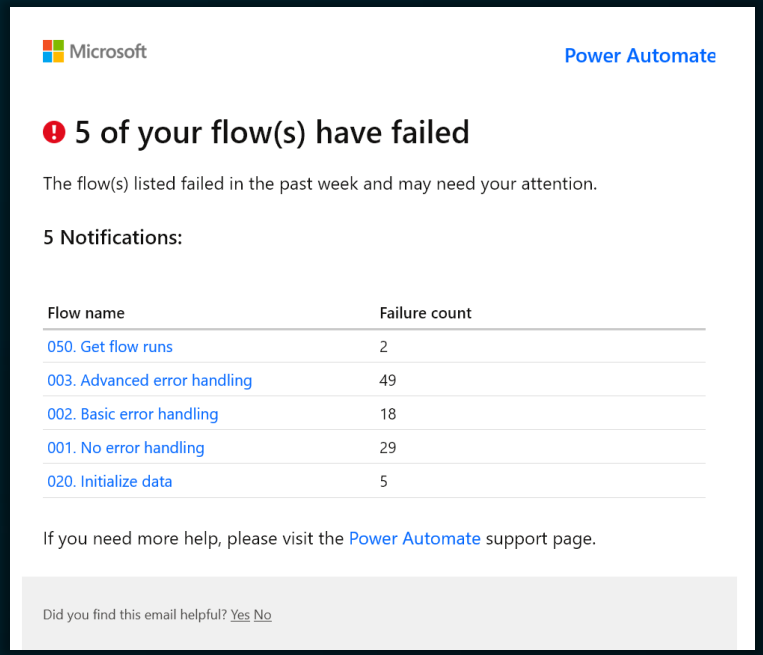
You can see it in the flow run history:



28-day run history ⓘ Edit columns All runs

Start	Duration	Status
May 15, 09:37 AM (3 sec ago)	00:00:01	Failed
May 12, 11:33 AM (2 d ago)	00:00:01	Test failed
May 12, 11:24 AM (2 d ago)	00:00:01	Test failed
May 12, 11:17 AM (2 d ago)	00:00:01	Test failed
May 12, 11:06 AM (2 d ago)	00:05:01	Test succeeded
May 12, 10:53 AM (2 d ago)	00:00:05	Test succeeded

You will receive an email:



Microsoft Power Automate

5 of your flow(s) have failed

The flow(s) listed failed in the past week and may need your attention.

5 Notifications:

Flow name	Failure count
050. Get flow runs	2
003. Advanced error handling	49
002. Basic error handling	18
001. No error handling	29
020. Initialize data	5

If you need more help, please visit the Power Automate support page.

Did you find this email helpful? Yes No

But only once per day.

How will you know if a flow fails

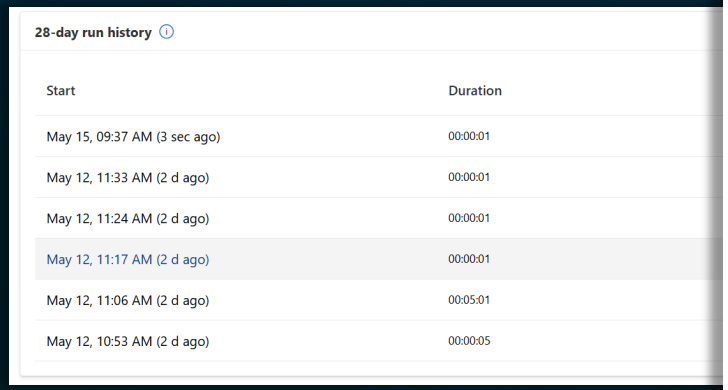
28-day run history ⓘ

Edit columns | All runs

Start	Duration	Status
May 15, 09:37 AM (3 sec ago)	00:00:01	Failed
May 12, 11:33 AM (2 d ago)	00:00:01	Test failed
May 12, 11:24 AM (2 d ago)	00:00:01	Test failed
May 12, 11:17 AM (2 d ago)	00:00:01	Test failed
May 12, 11:06 AM (2 d ago)	00:05:01	Test succeeded
May 12, 10:53 AM (2 d ago)	00:00:05	Test succeeded

How will you know

You can see it in the flow ru



28-day run history

Start	Duration
May 15, 09:37 AM (3 sec ago)	00:00:01
May 12, 11:33 AM (2 d ago)	00:00:01
May 12, 11:24 AM (2 d ago)	00:00:01
May 12, 11:17 AM (2 d ago)	00:00:01
May 12, 11:06 AM (2 d ago)	00:05:01
May 12, 10:53 AM (2 d ago)	00:00:05

Microsoft Power Automate

5 of your flow(s) have failed

The flow(s) listed failed in the past week and may need your attention.

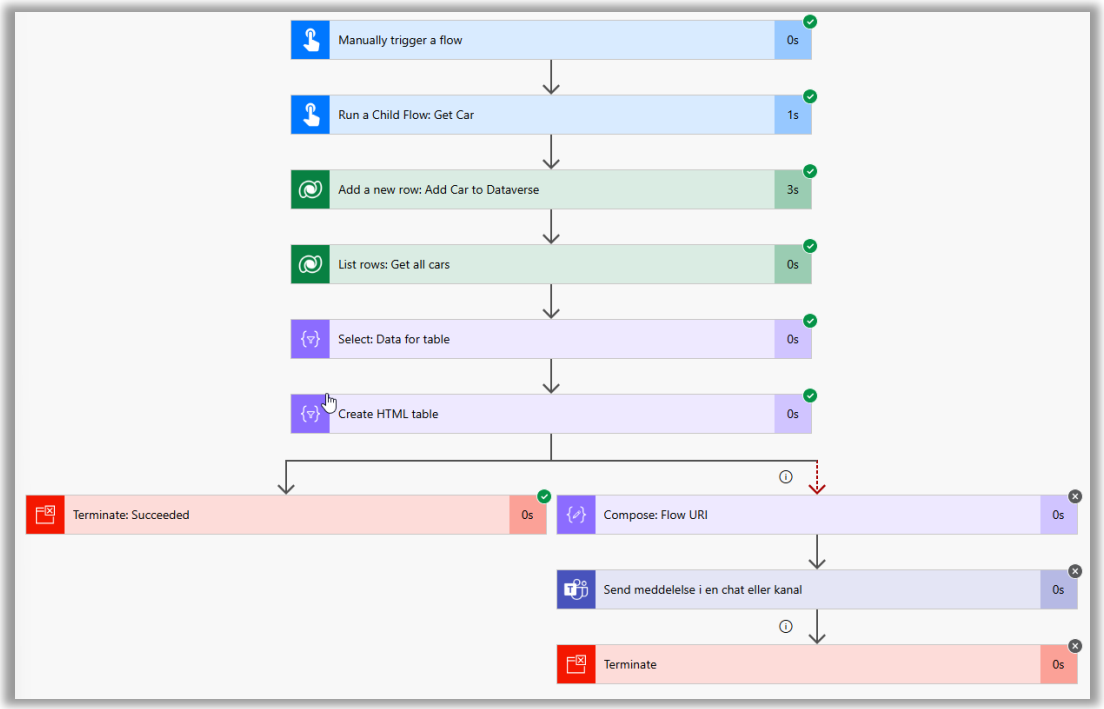
5 Notifications:

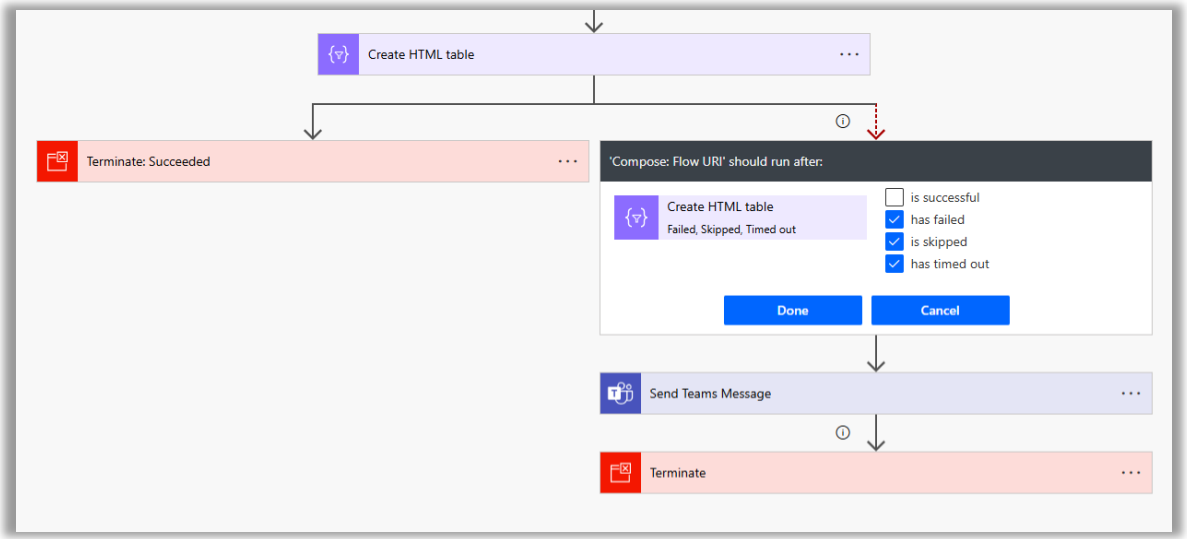
Flow name	Failure count
050. Get flow runs	2
003. Advanced error handling	49
002. Basic error handling	18
001. No error handling	29
020. Initialize data	5

If you need more help, please visit the Power Automate support page.

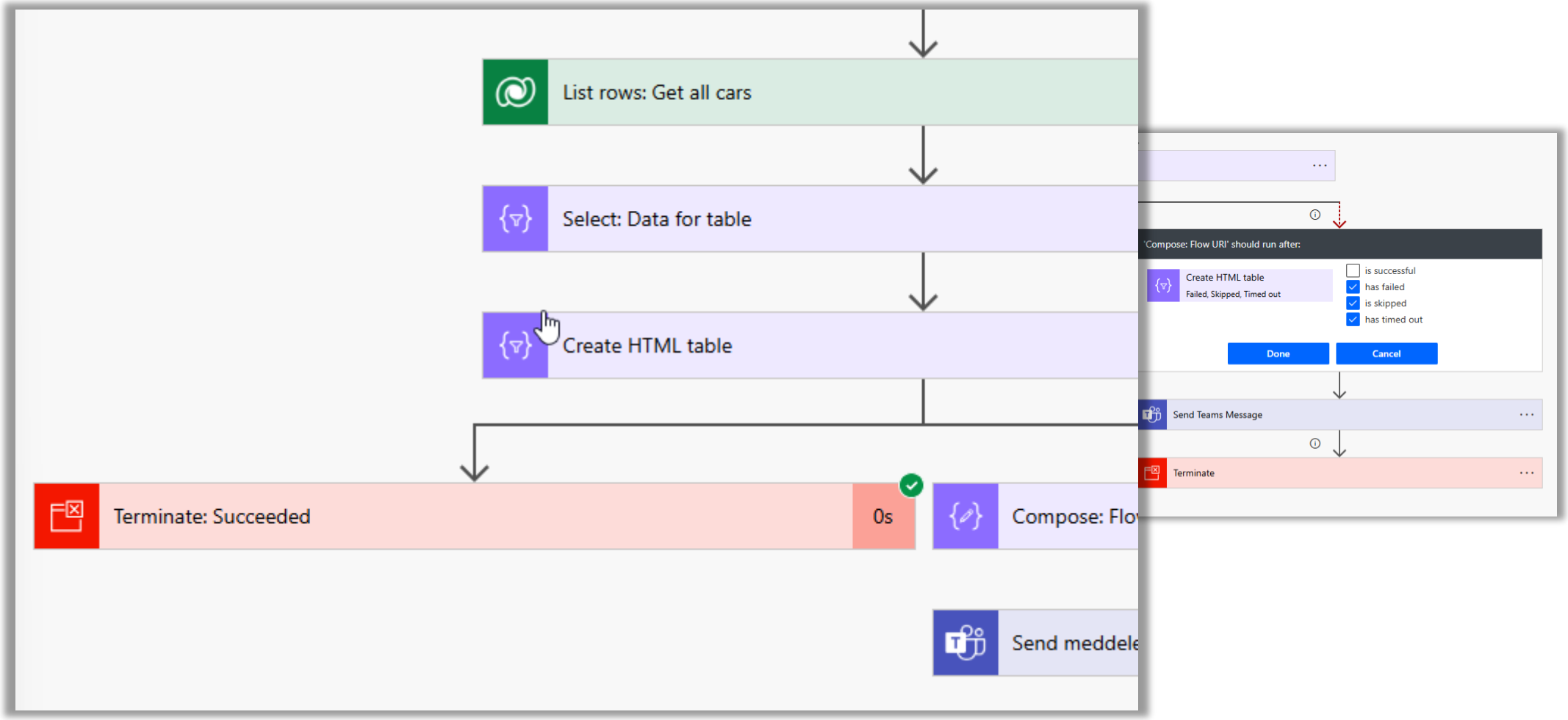
Did you find this email helpful? Yes No

Basic error handling – Parallel Actions

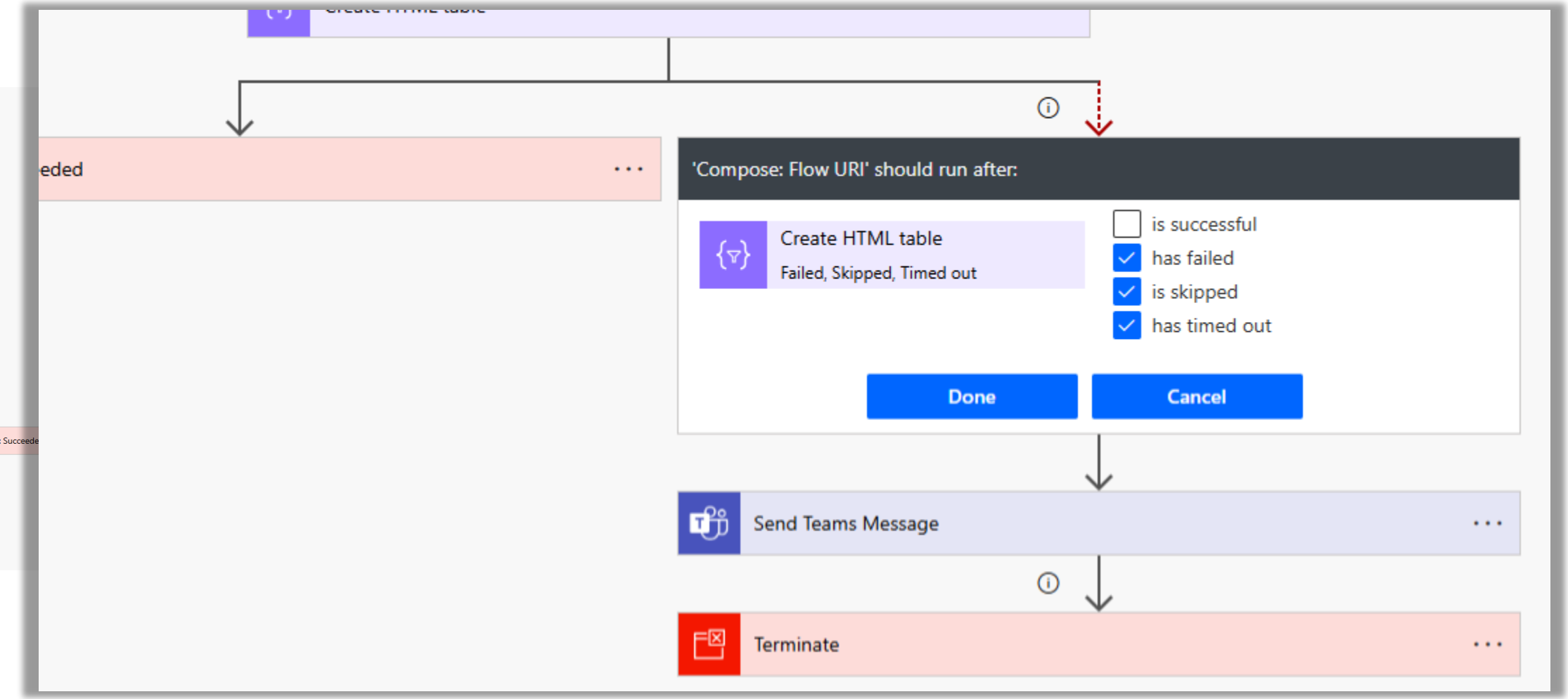




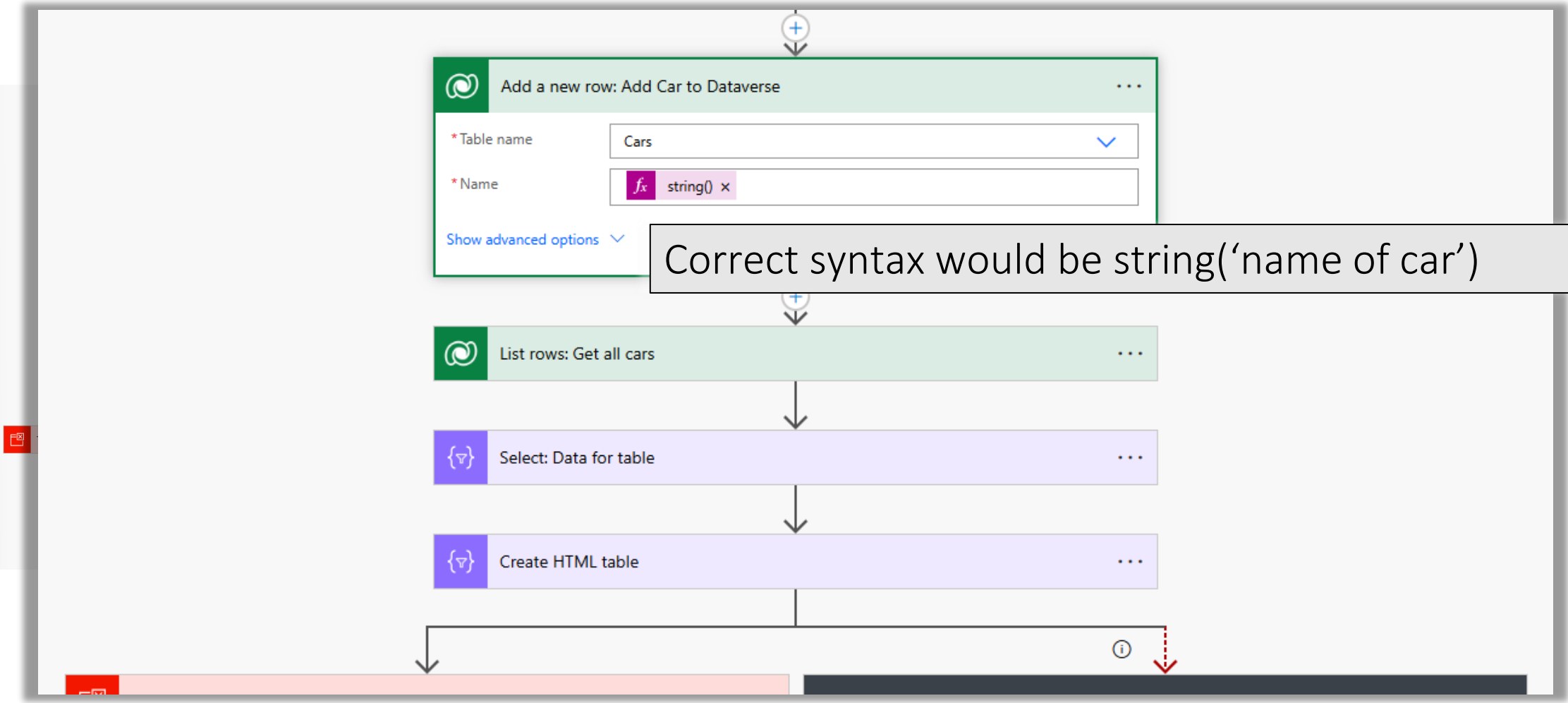
Left Branch – Success Path

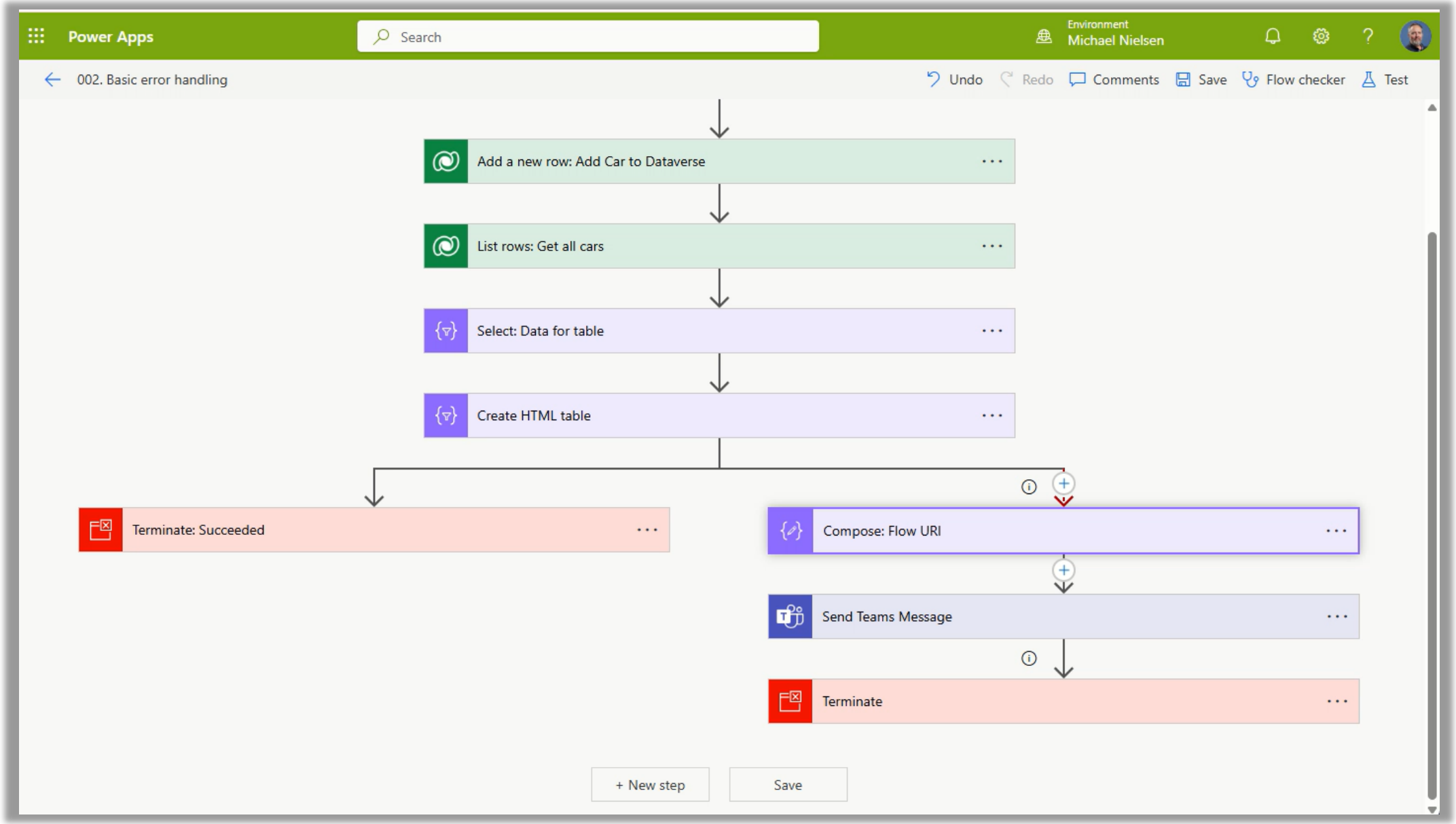


Right Branch – Failure Path

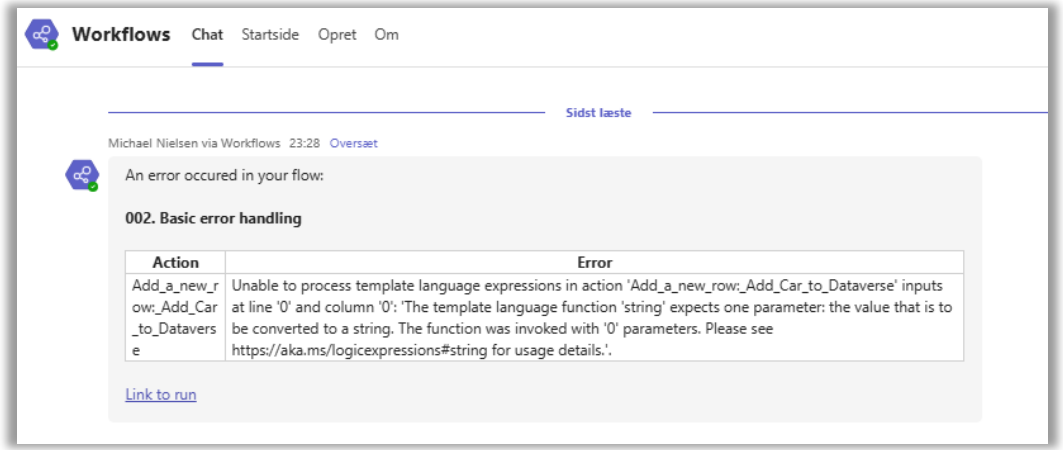


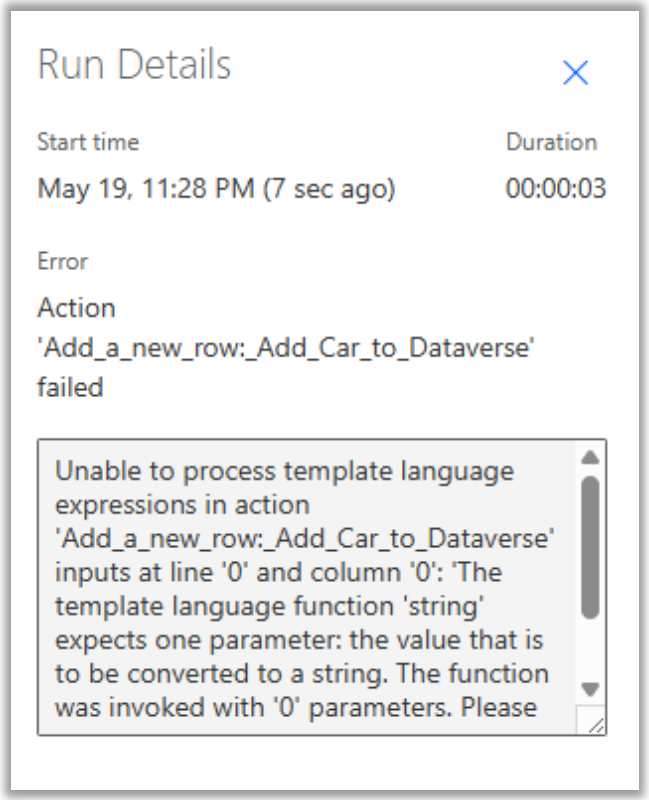
Syntax Error Example




Power Apps
Search
Environment
Michael Nielsen
002. Basic error handling
Undo
Redo
Comments
Save
Flow checker
Test
Add a new row: Add Car to Dataverse
List rows: Get all cars
Select: Data for table
Create HTML table
Terminate: Succeeded
Compose: Flow URI
Send Teams Message
Terminate
+ New step
Save

Messages received after error handling





Teams message received

Michael Nielsen via Workflows 23:28 Oversæt

An error occured in your flow:

002. Basic error handling

Action	Error
Add_a_new_row:_Add_Car_to_Dataverse	Unable to process template language expressions in action 'Add_a_new_row:_Add_Car_to_Dataverse' inputs at line '0' and column '0': 'The template language function 'string' expects one parameter: the value that is to be converted to a string. The function was invoked with '0' parameters. Please see https://aka.ms/logicexpressions#string for usage details.'.

Link to run

Teams message received

Michael Nielsen via Workflows 23:28 Oversæt

An error occured in your flow:

002. Basic error handling

Action	Error
Add_a_new_row:_Add_Car_to_Dataverse	Unable to process template language expressions in action 'Add_a_new_row:_Add_Car_to_Dataverse' inputs at line '0' and column '0': 'The template language function 'string' expects one parameter: the value that is to be converted to a string. The function was invoked with '0' parameters. Please see https://aka.ms/logicexpressions#string for usage details.'.

Link to run

Teams message received

Michael Nielsen via Workflows 23:28 Oversæt

An error occured in your flow:

002. Basic error handling

Action	Error
Add_a_new_row:_Add_Car_to_Dataverse	Unable to process template language expressions in action 'Add_a_new_row:_Add_Car_to_Dataverse' inputs at line '0' and column '0': 'The template language function 'string' expects one parameter: the value that is to be converted to a string. The function was invoked with '0' parameters. Please see https://aka.ms/logicexpressions#string for usage details.'.

Link to run

Teams message received

Michael Nielsen via Workflows 23:28 Oversæt

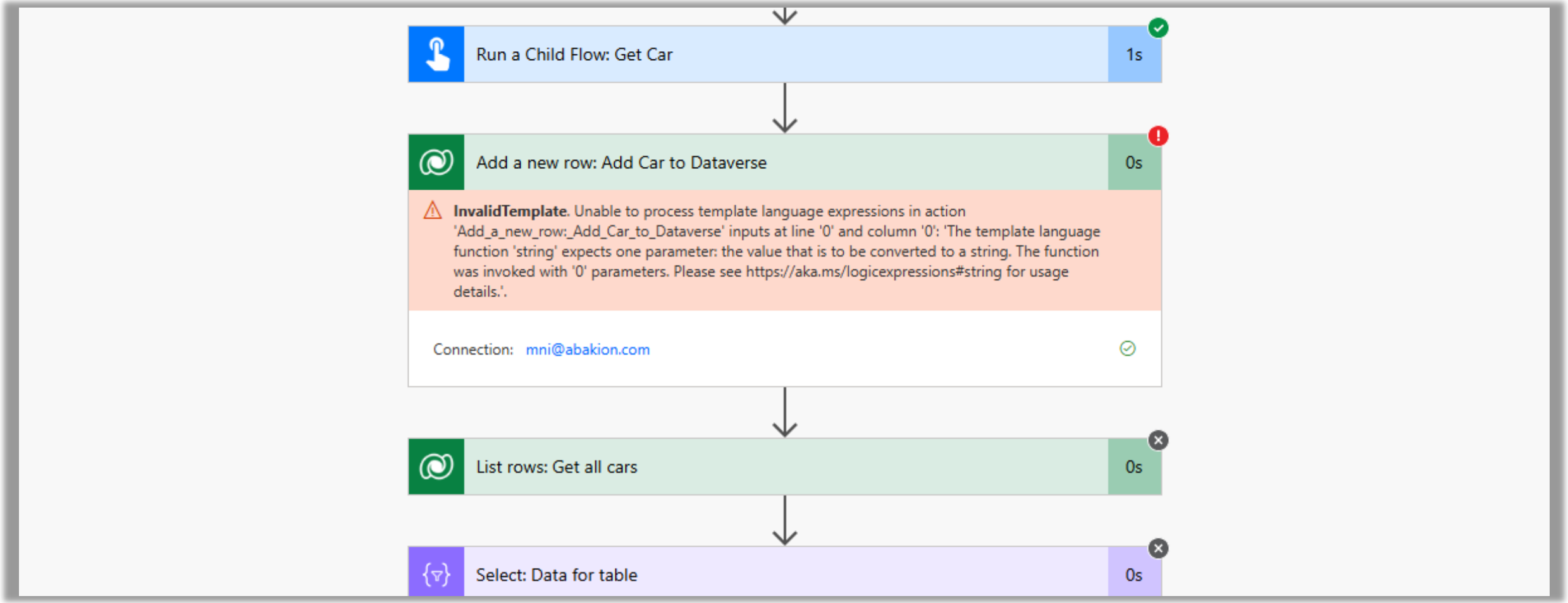
An error occured in your flow:

002. Basic error handling

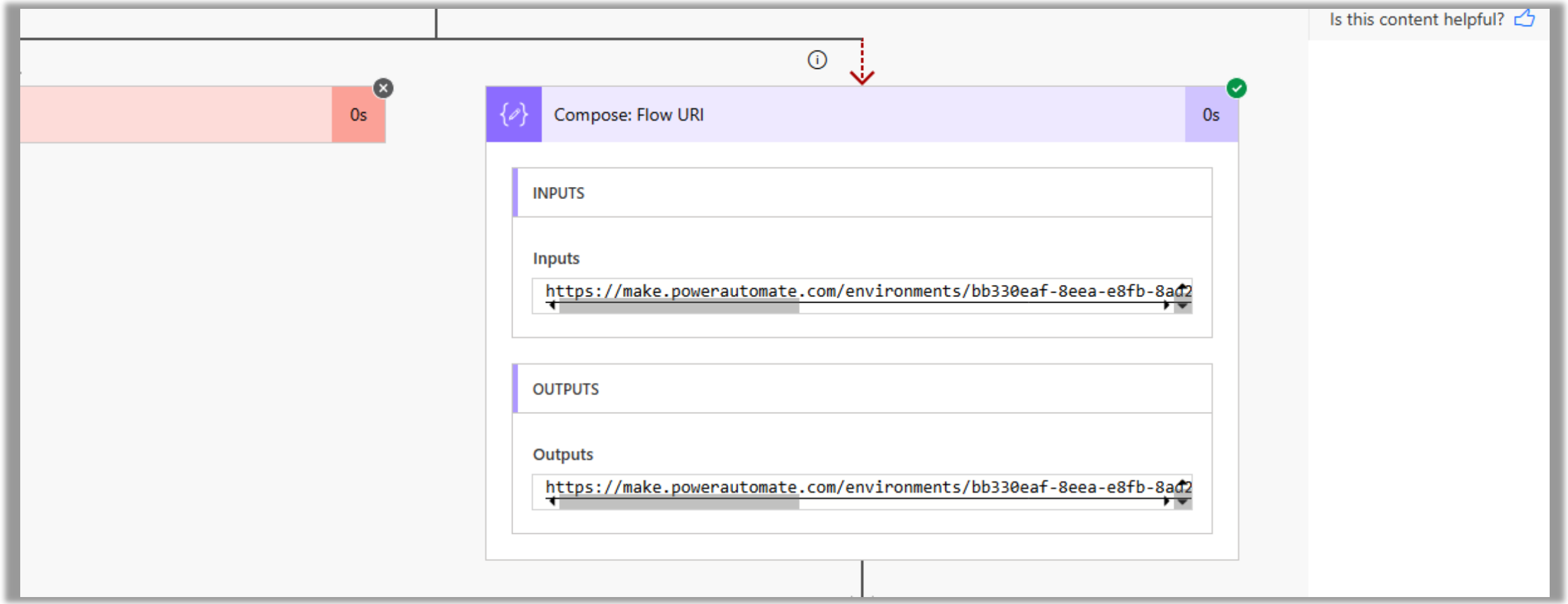
Action	Error
Add_a_new_row:_Add_Car_to_Dataverse	Unable to process template language expressions in action 'Add_a_new_row:_Add_Car_to_Dataverse' inputs at line '0' and column '0': 'The template language function 'string' expects one parameter: the value that is to be converted to a string. The function was invoked with '0' parameters. Please see https://aka.ms/logicexpressions#string for usage details.'.

Link to run

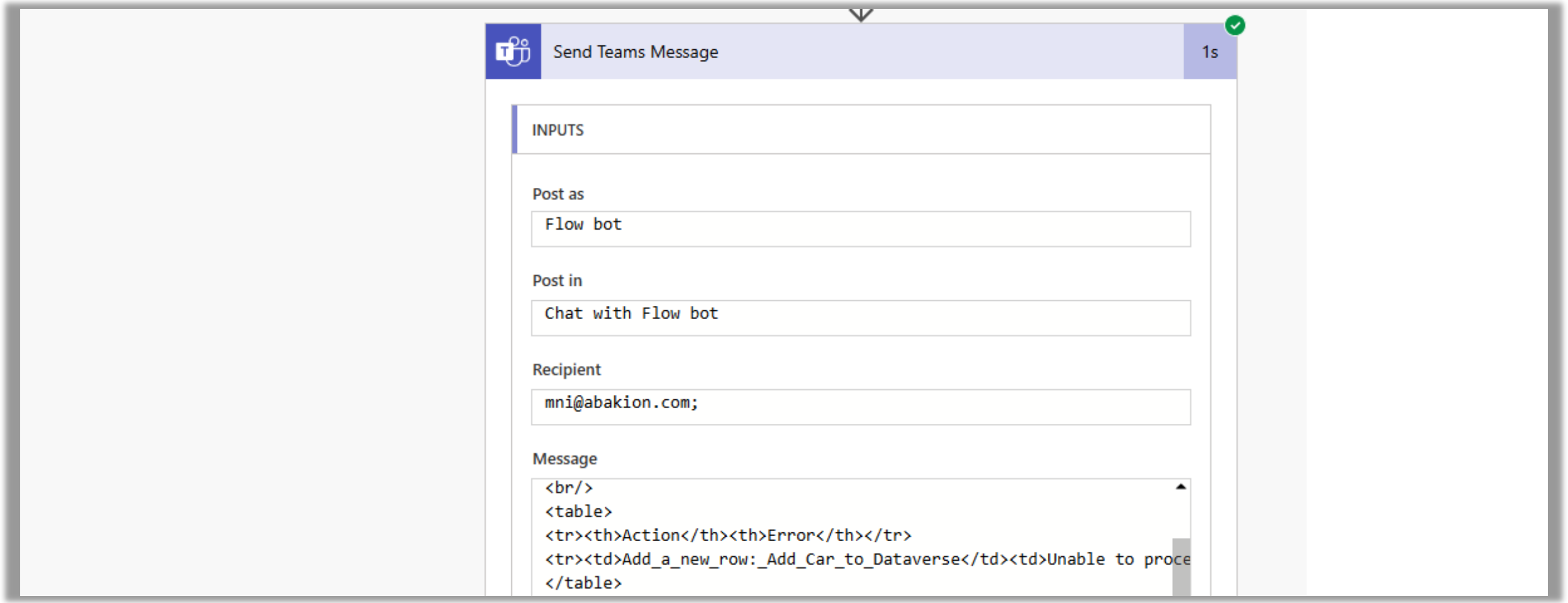
Direct link into my failed flow



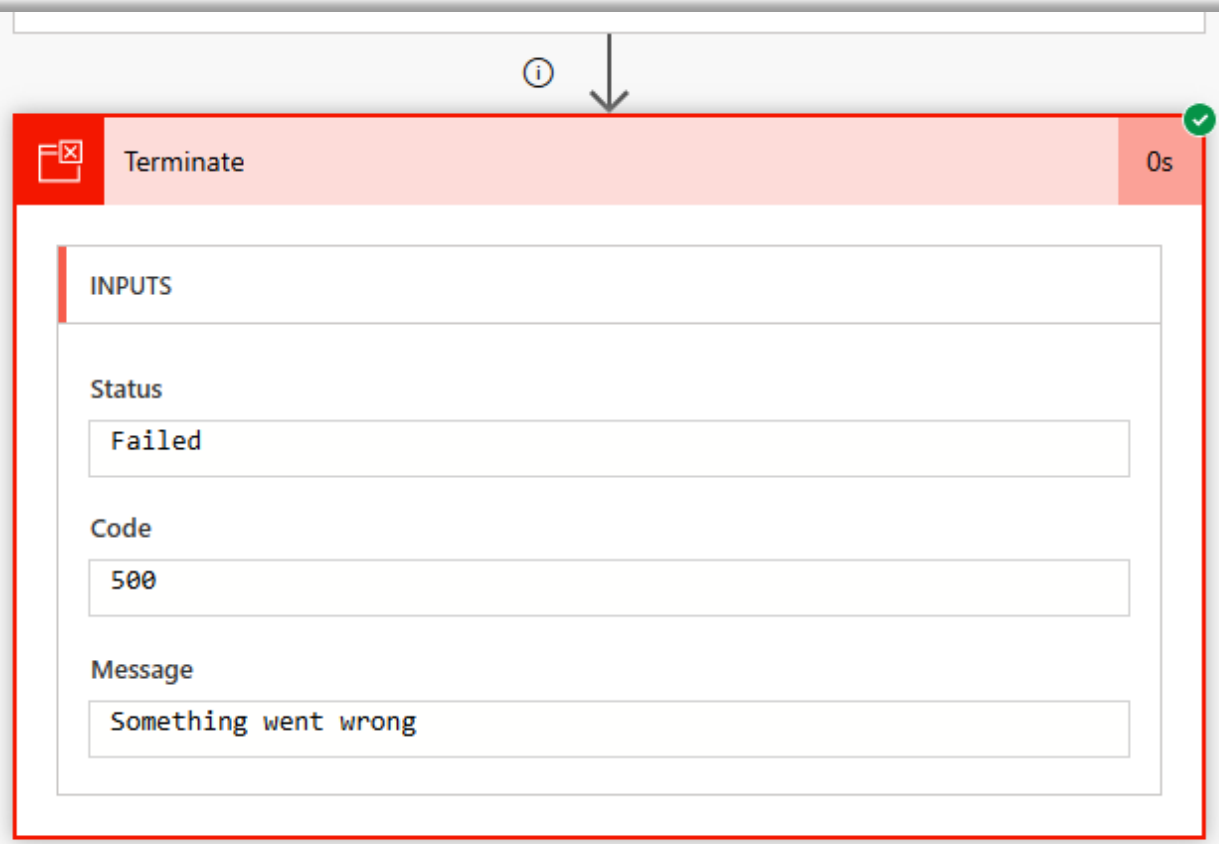
Get the URI for the actual flow run



Send the Teams message

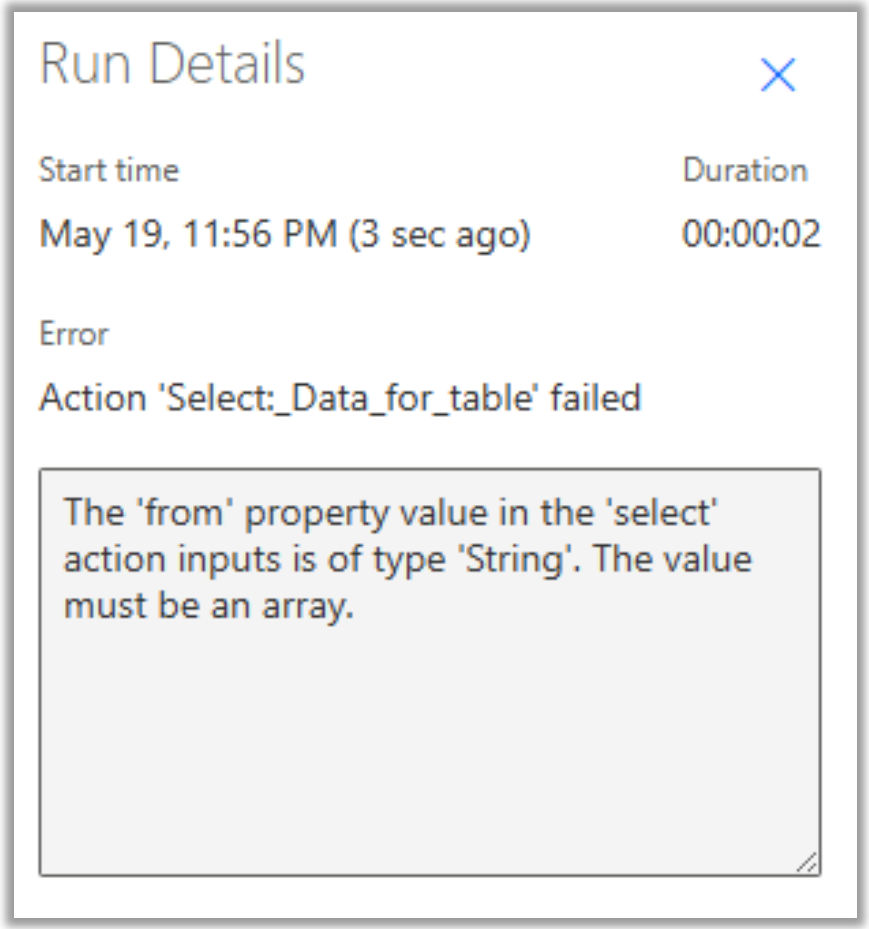


Terminate the flow as failed

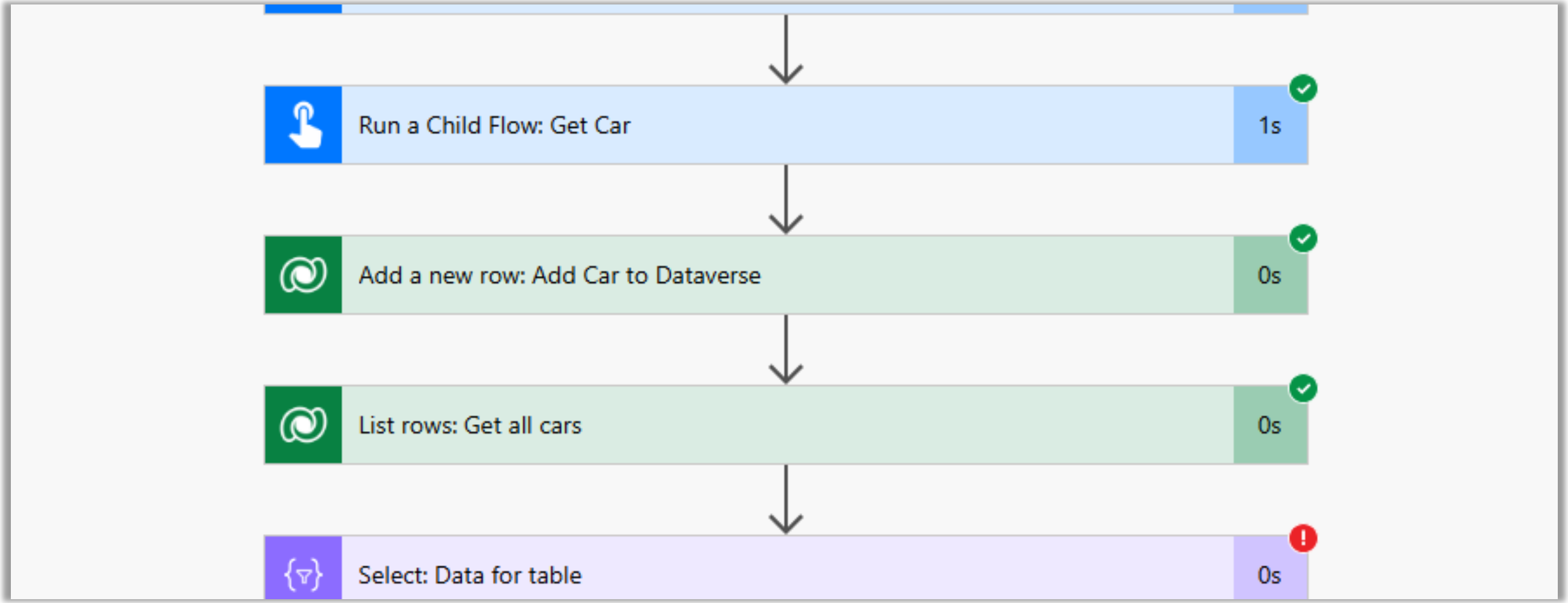


Unexpected Failure

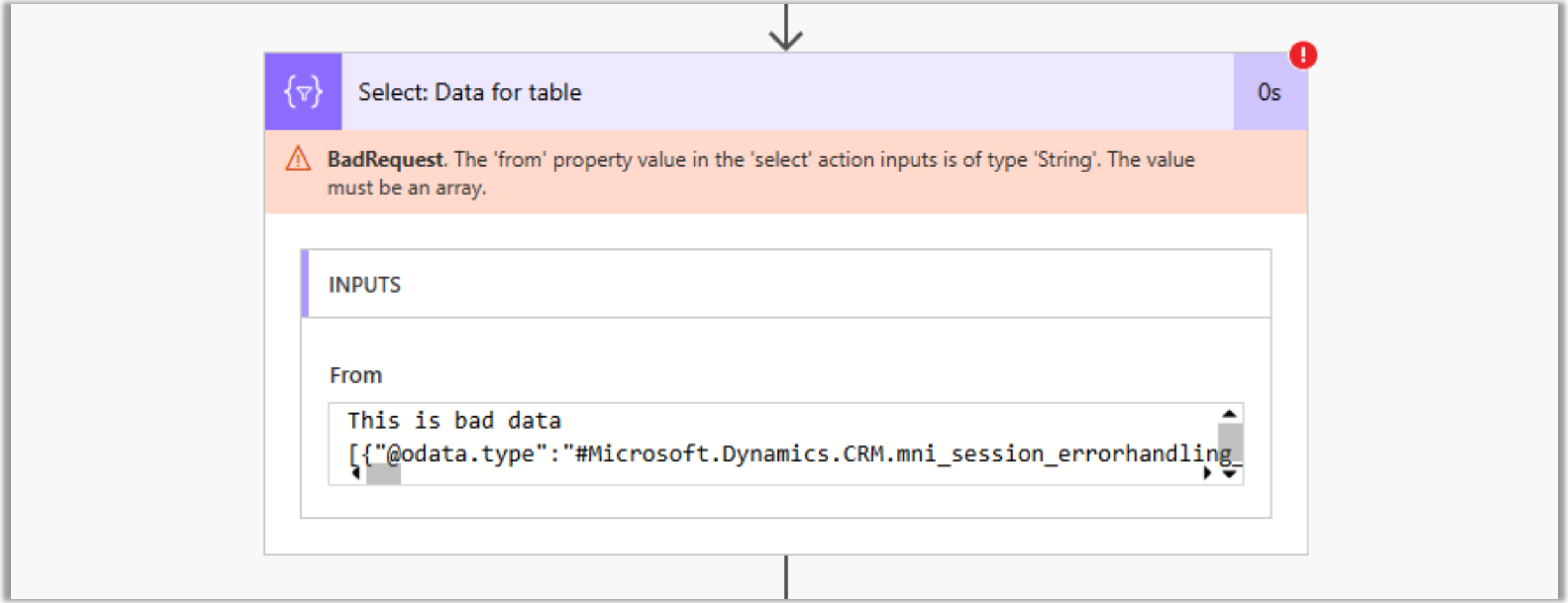
But what if another action fails than the one you expected?



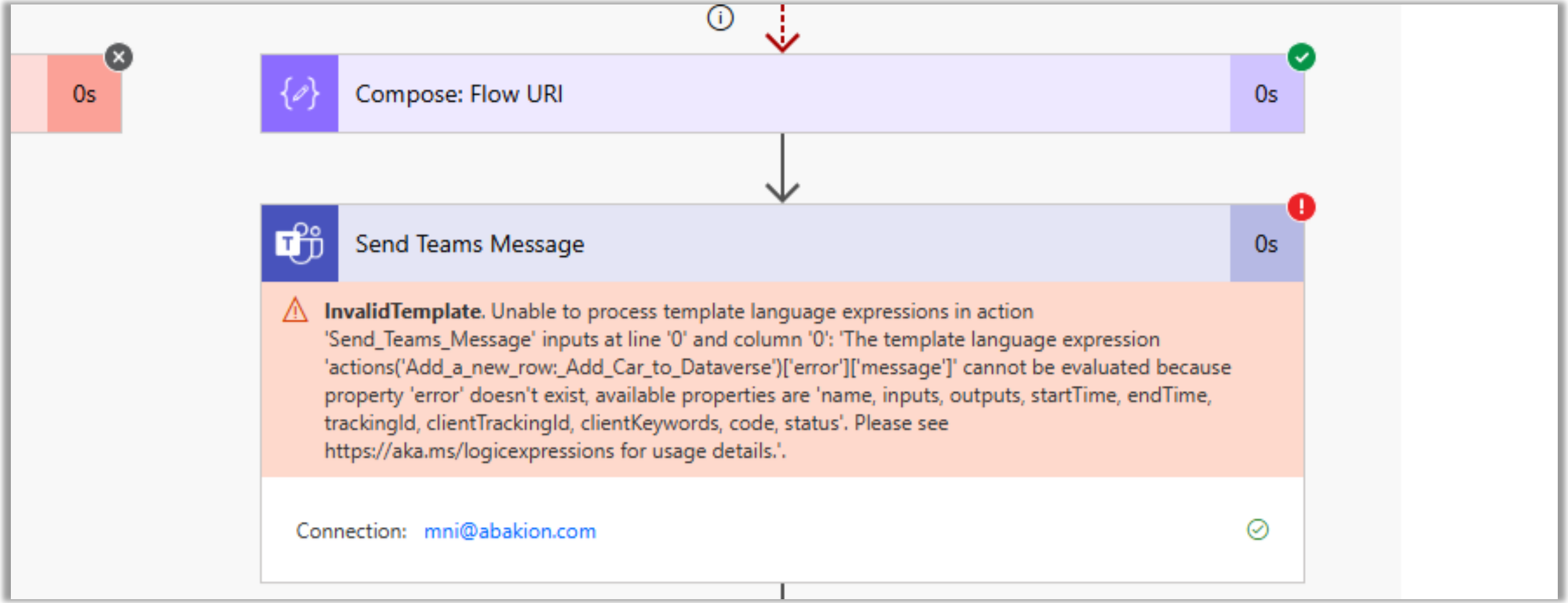
Failure is in another action



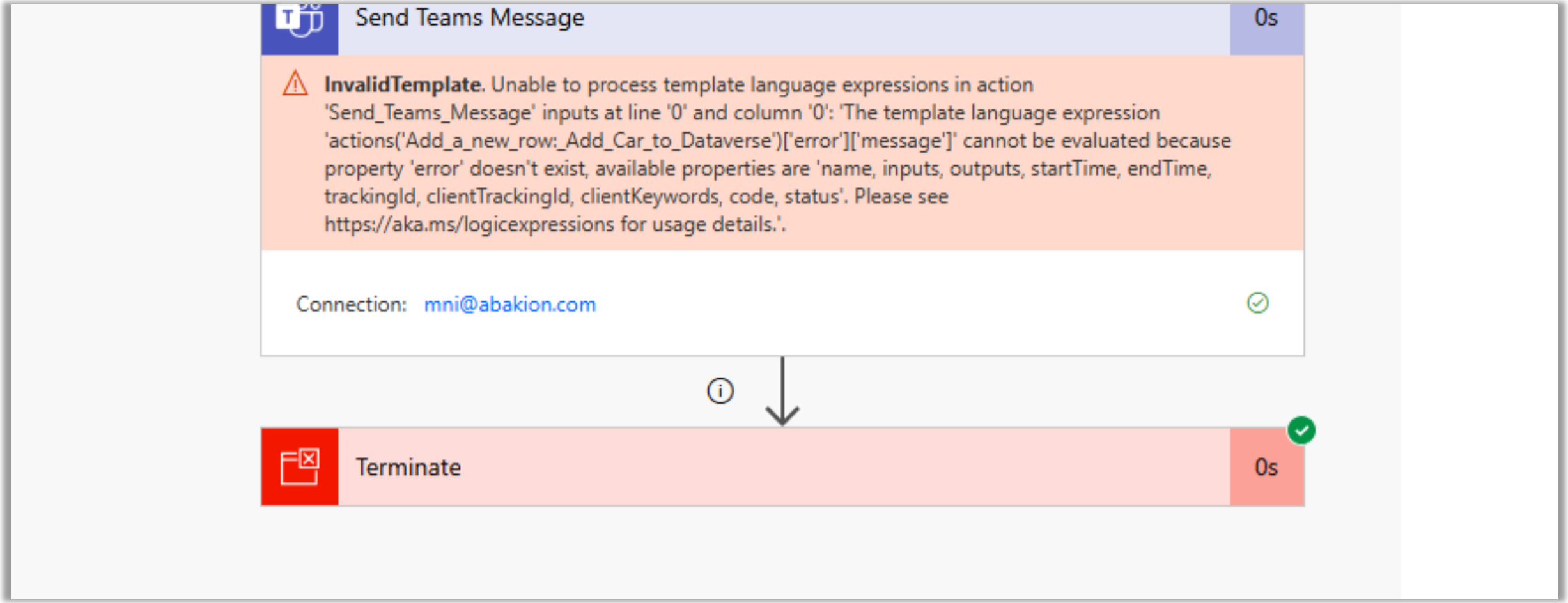
Bad Data was introduced to Select: Data for table



Basic error handling



Basic error handling



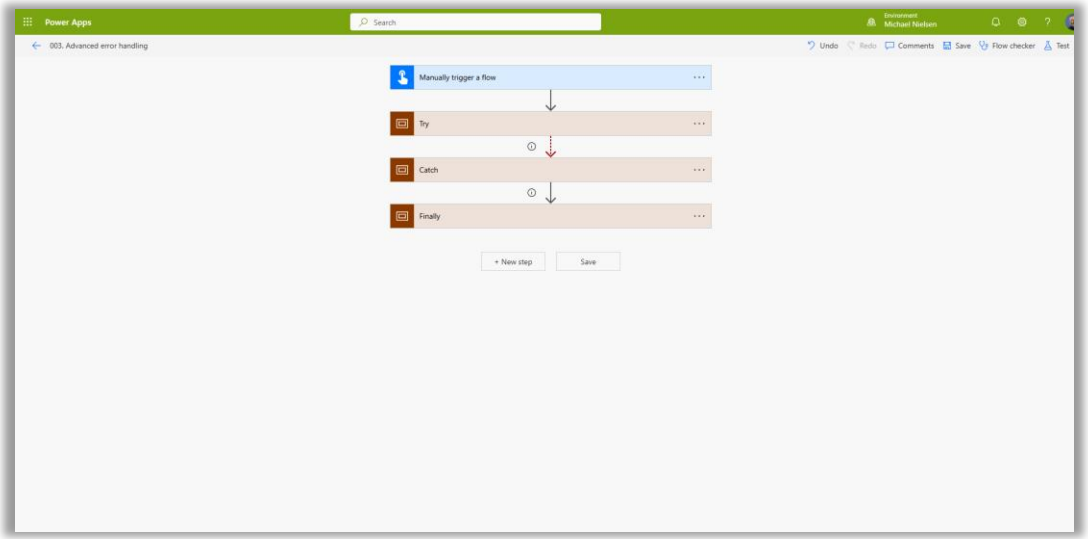
Implement Try-Catch-Finally

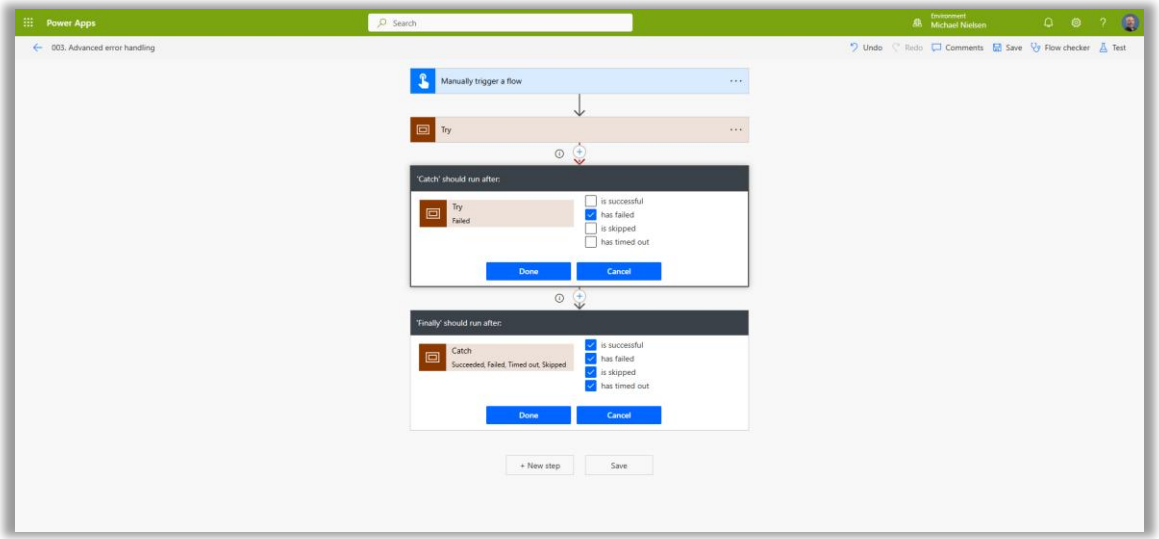
- **Try Scope**
  Encloses the logic where errors can occur.

- **Catch Scope**
  Catches and retrieves any errors that occur in the Try scope.

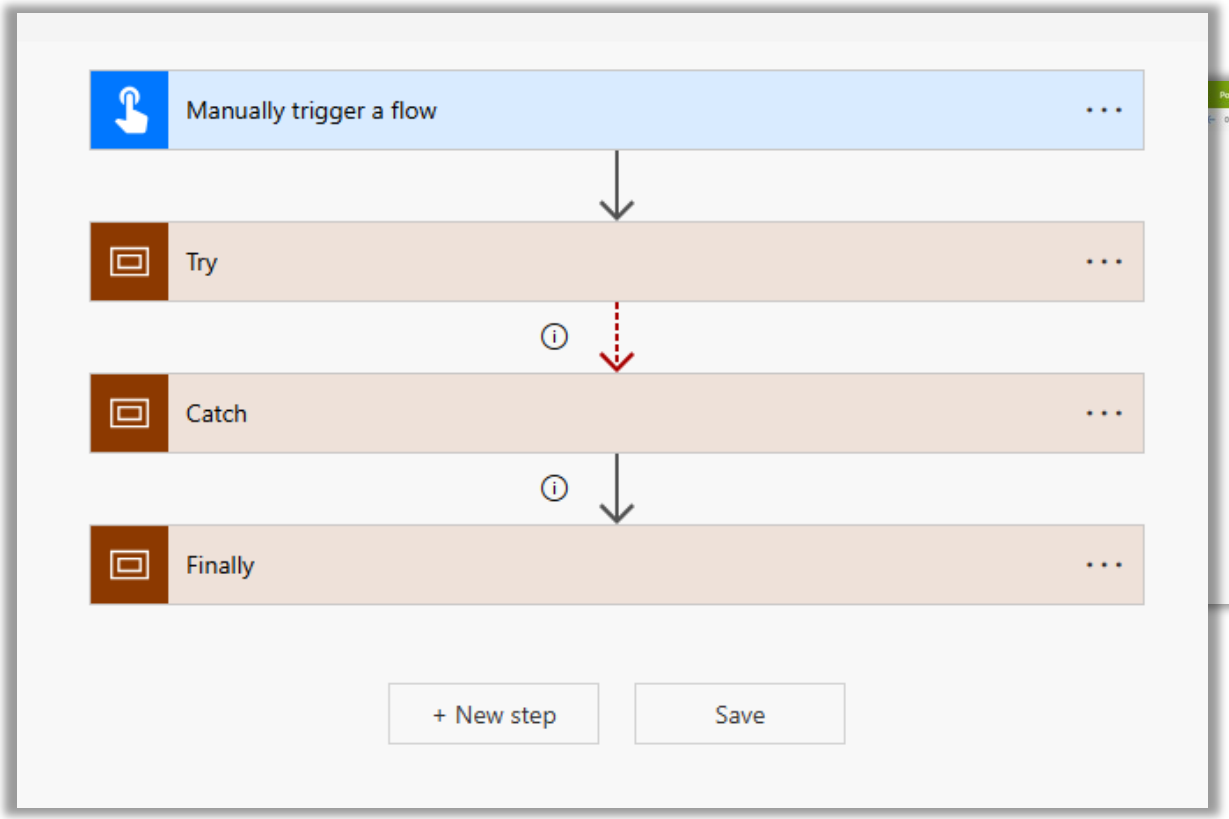
- **Finally Scope**
  Runs regardless of whether an error occurred or not, useful for cleanup operations.

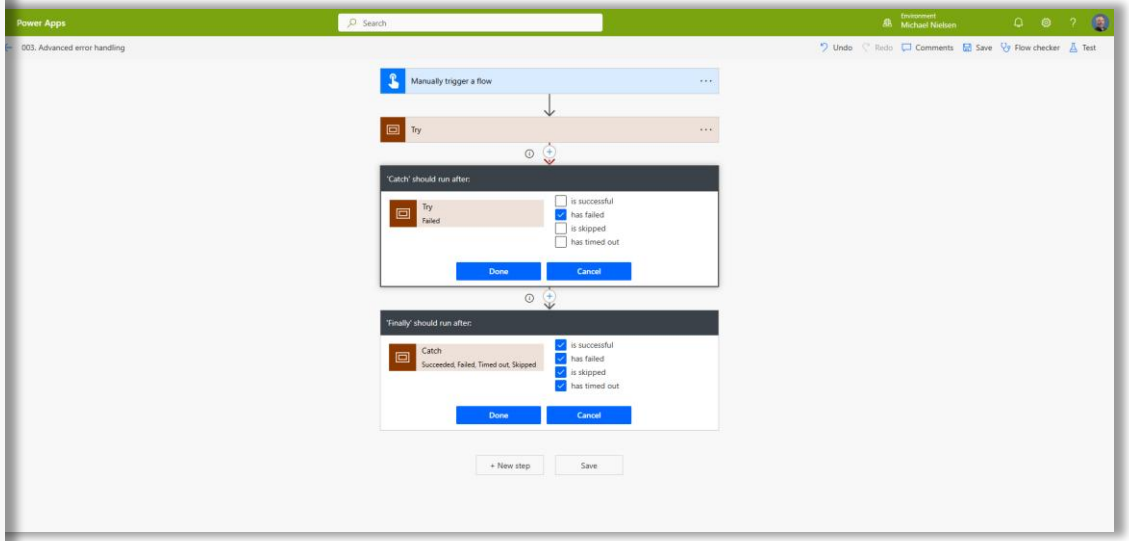
FLOW WITH TRY-CATCH FINALLY



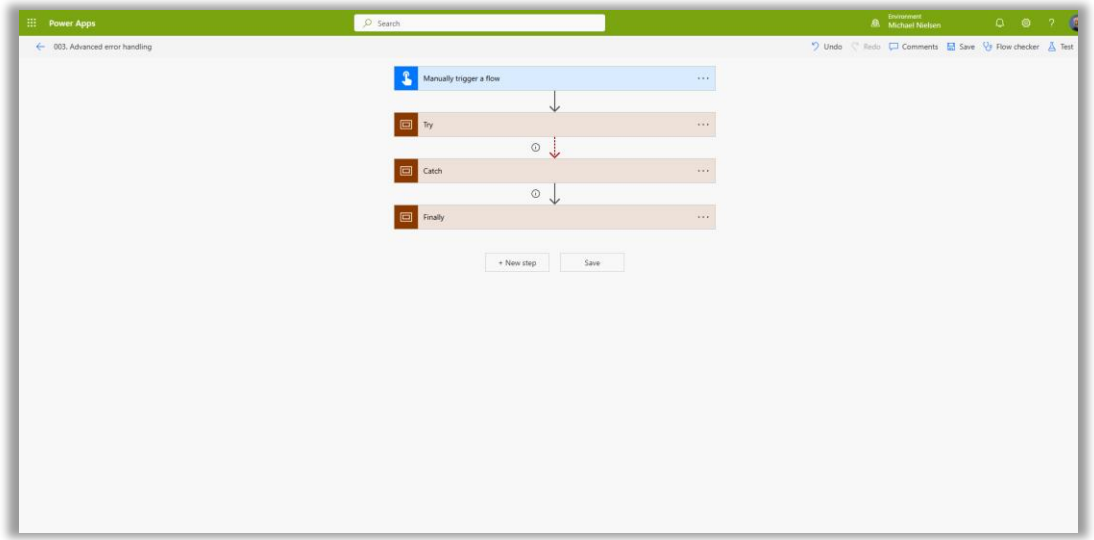


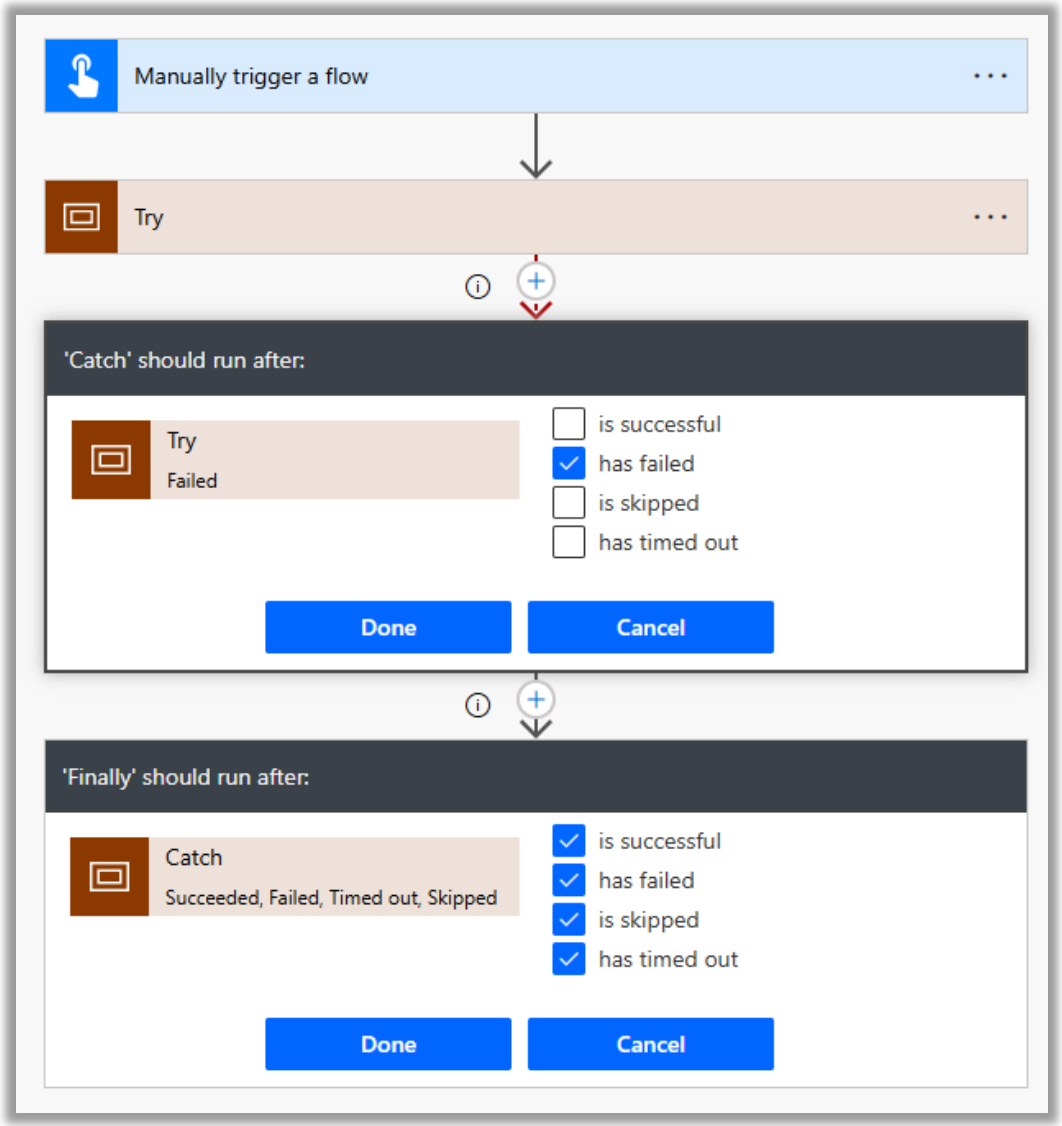
FLOW WITH TRY-CATCH FINALLY



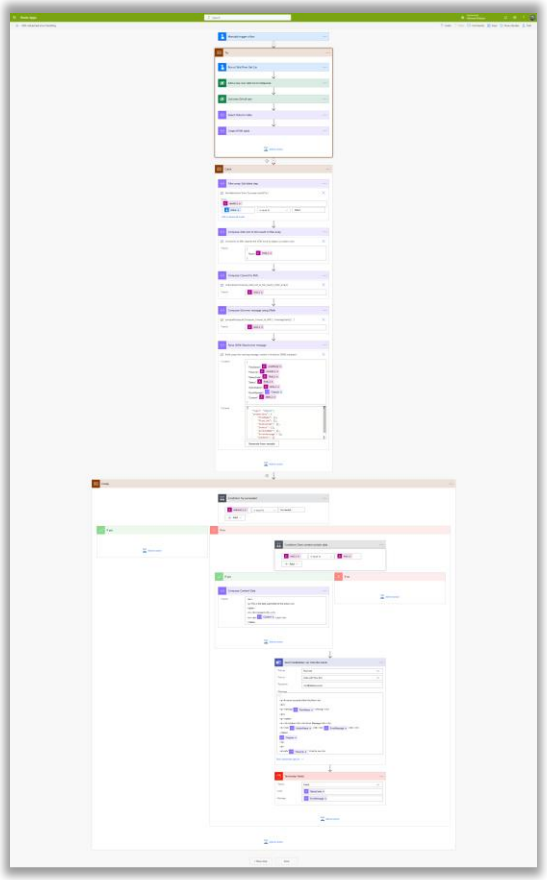


FLOW WITH TRY-CATCH FINALLY

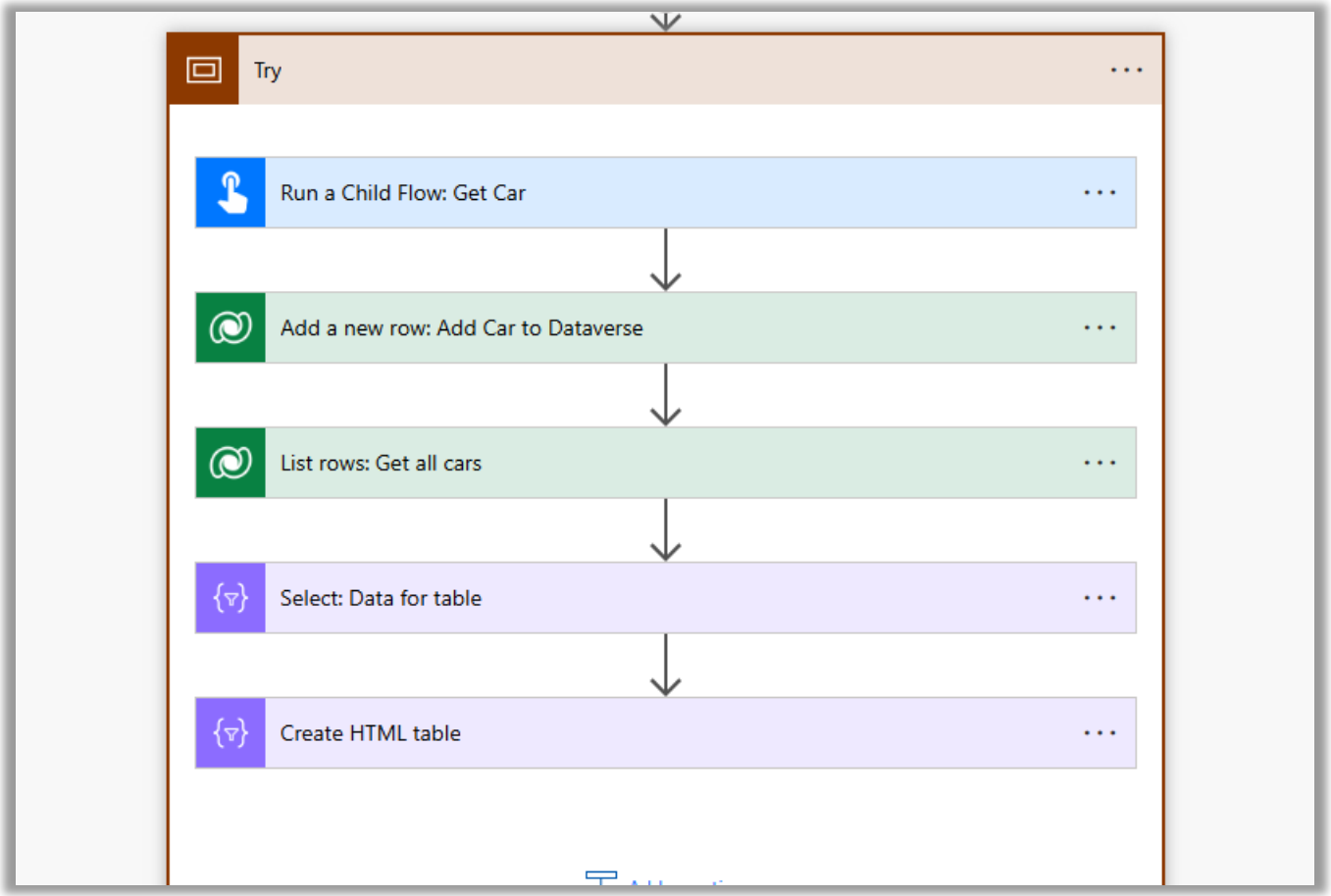




FLOW WITH TRY-CATCH FINALLY

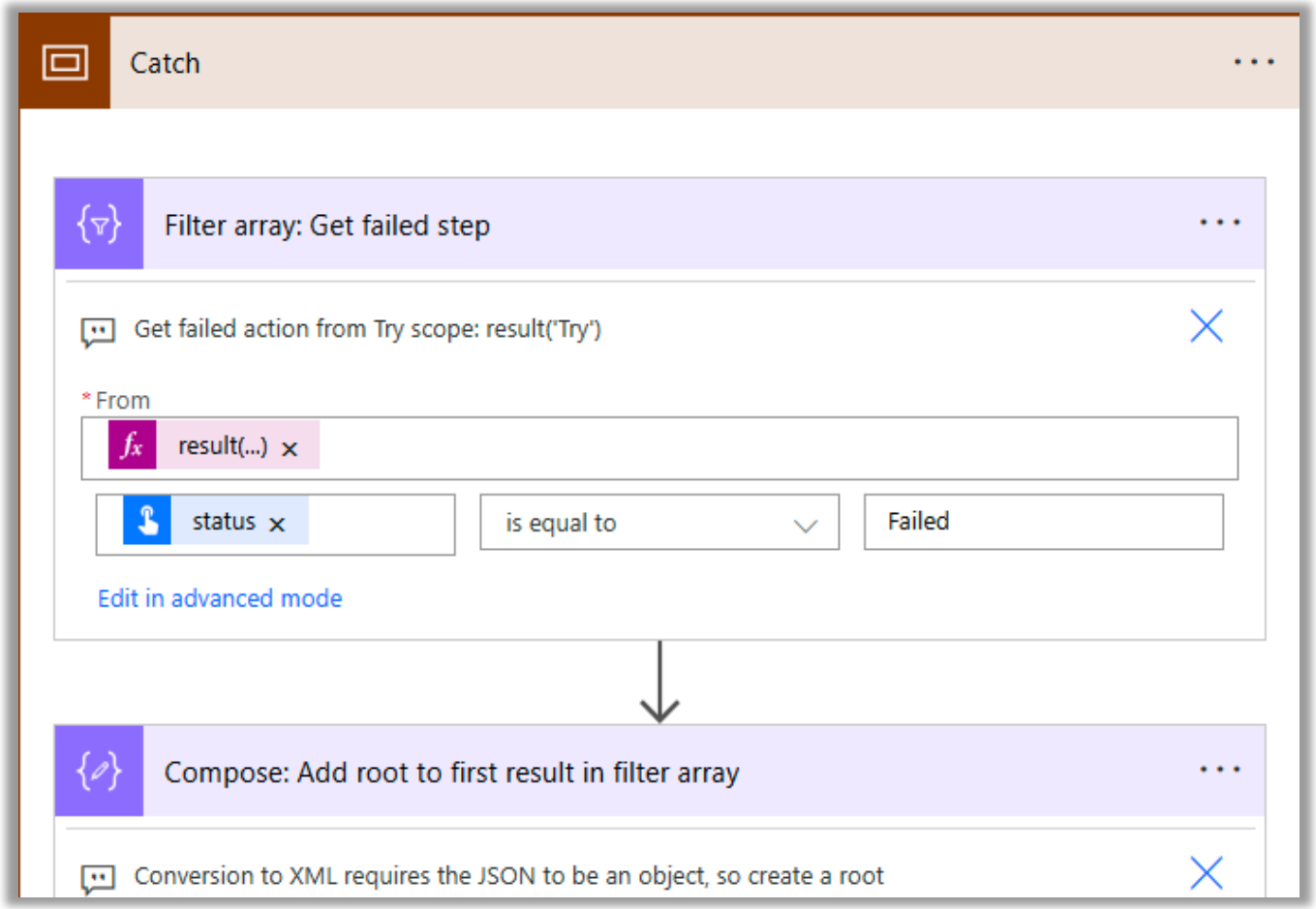


Try Scope



No changes to logic

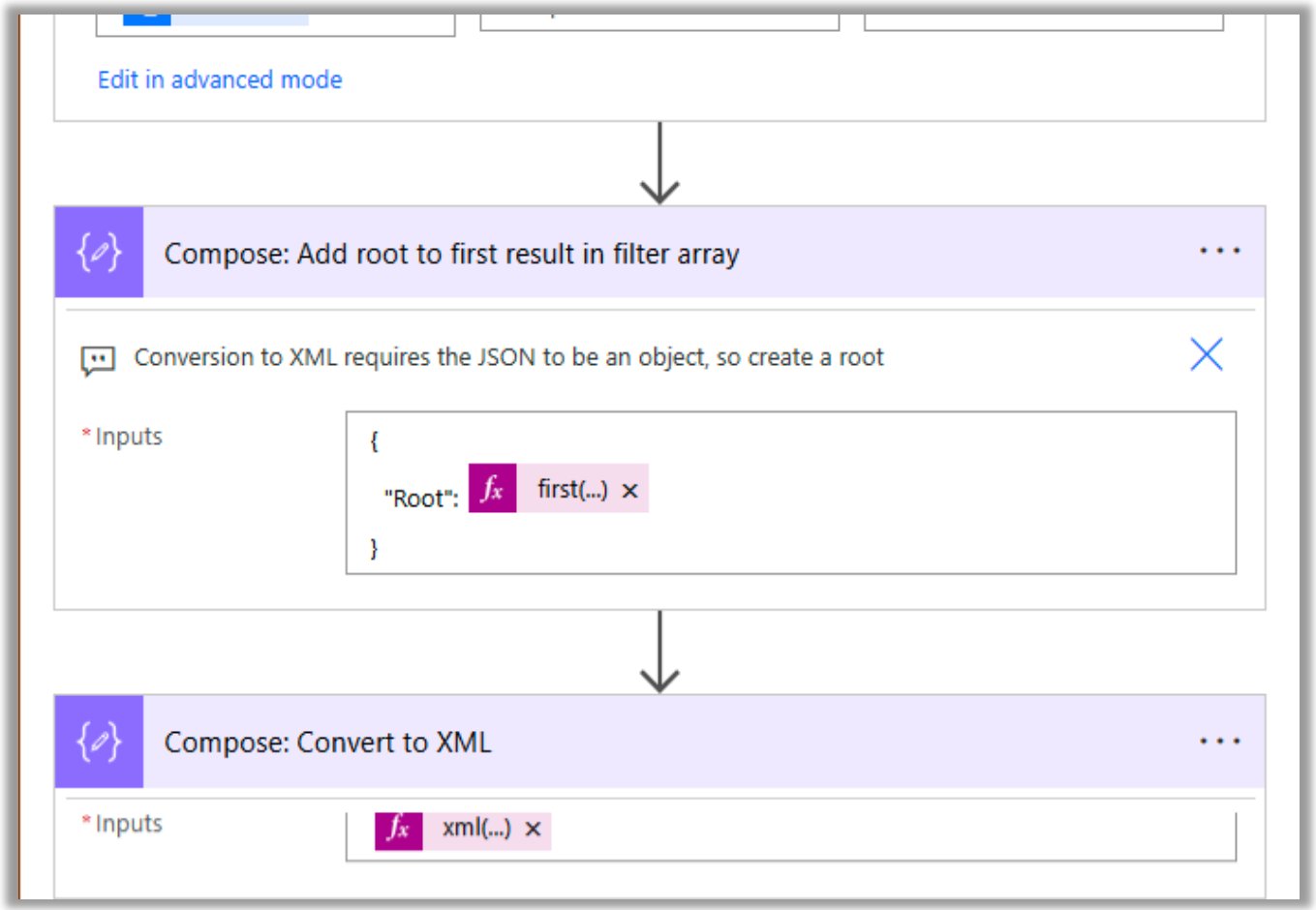
Catch Scope - Filter array



Filter actions within the Try scope container to identify failed steps.

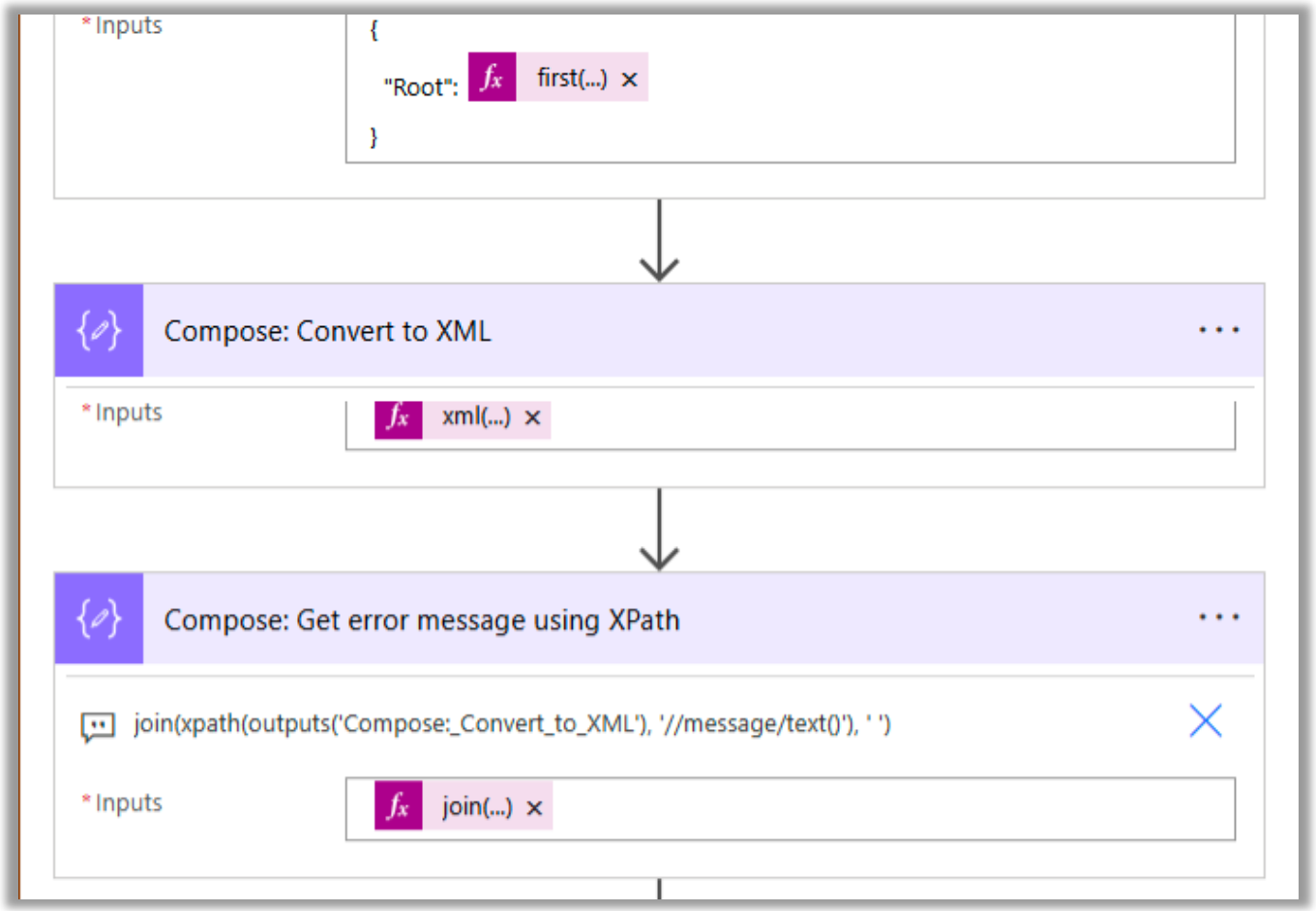
This process filters based on a status of 'Failed'.

Make new JSON object



Next, we need to format the data into an object to ensure proper conversion to XML.

Convert to XML

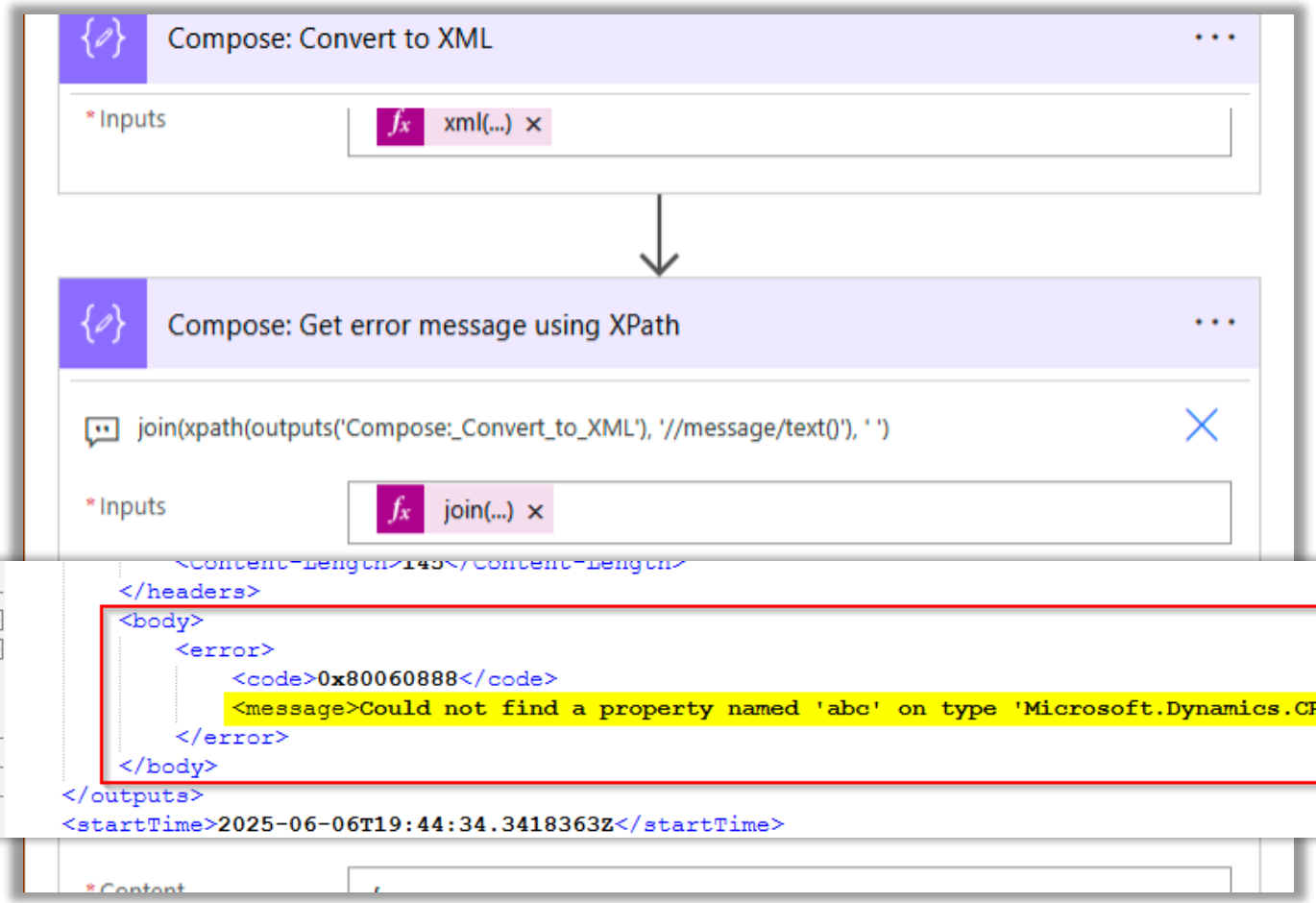


Convert the structured JSON object to XML using the xml() function.

XPATH Magic

Use XPath to extract all <message> elements from the XML and join them into a single string.

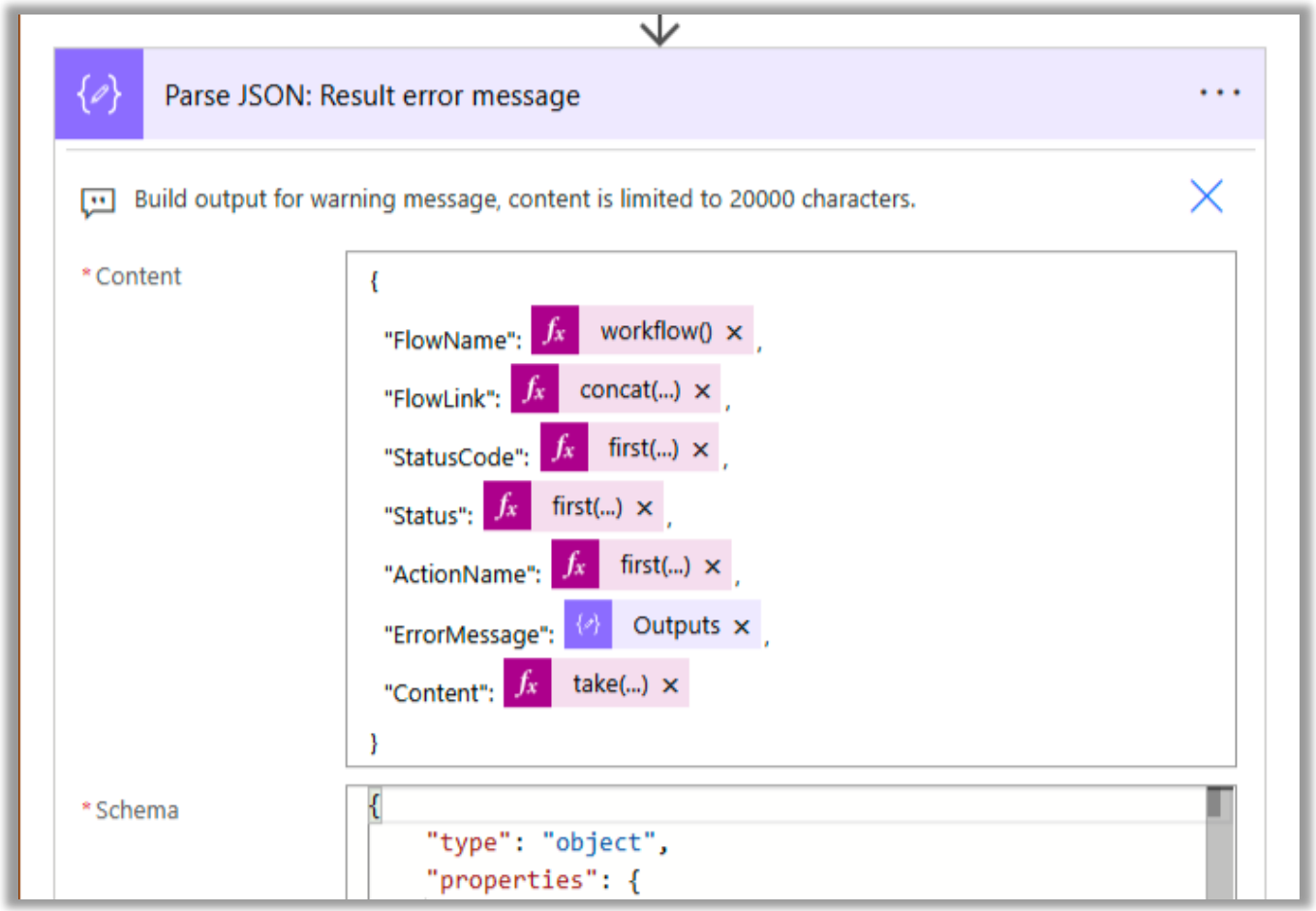
XPATH Magic




```
    </headers>
    <body>
        <error>
            <code>0x80060888</code>
            <message>Could not find a property named 'abc' on type 'Microsoft.Dynamics.CRM.mni_session_errorhandling_car'.</message>
        </error>
    </body>
</outputs>
<startTime>2025-06-06T19:44:34.3418363Z</startTime>
```

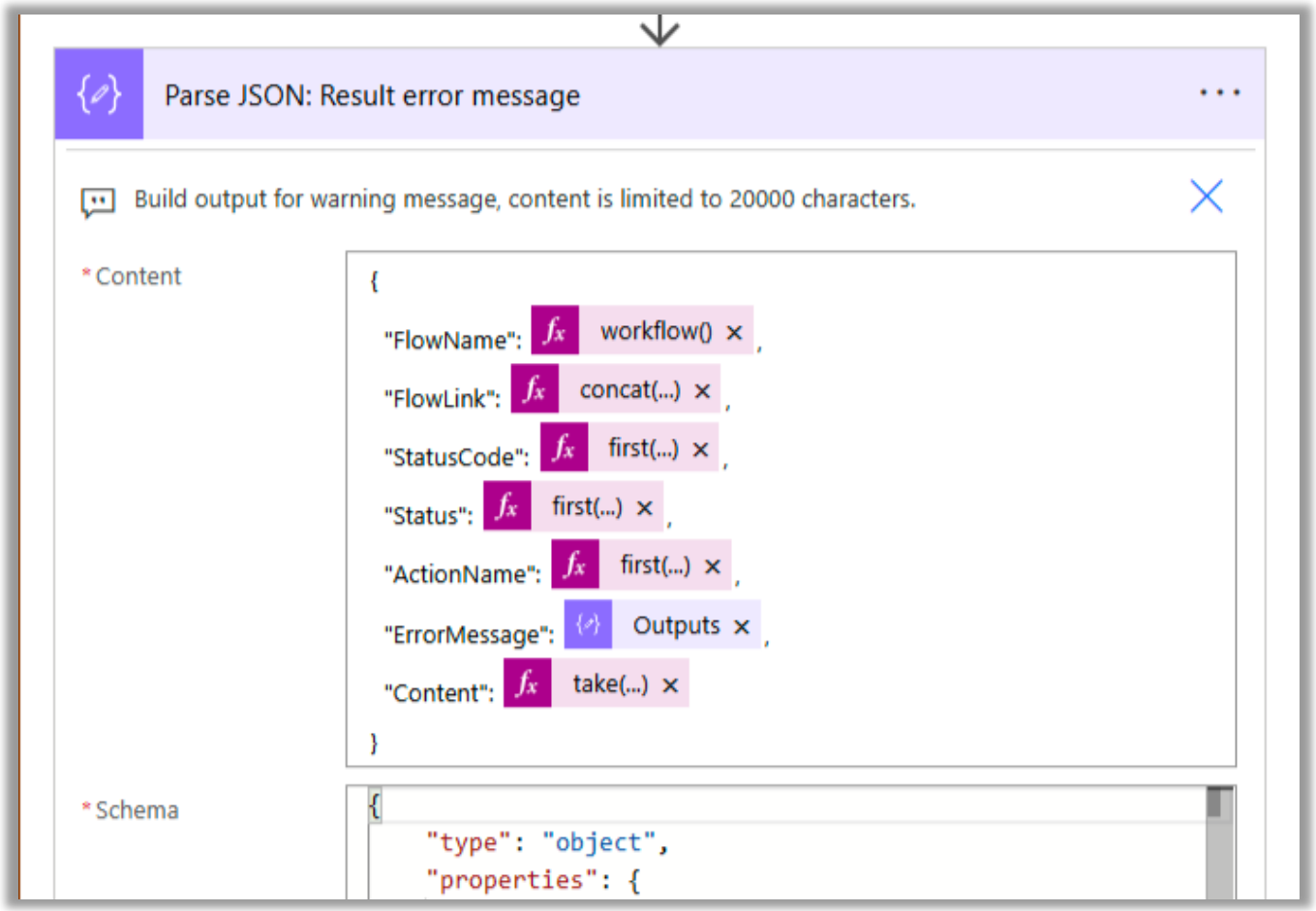
Use XPath to extract the contents of all <message> elements from the XML. This allows us to retrieve detailed error messages for logging or notifications.

Result error object



A custom object containing all necessary details for error message.

CATCH SCOPE – Commands (1)



FlowName

```
workflow().tags.flowDisplayName
```

FlowLink

```
concat('https://make.powerautomate.com/environments/',workflow().tags.environmentName,'/flows/',workflow().name, '/runs/',workflow().run.name)
```

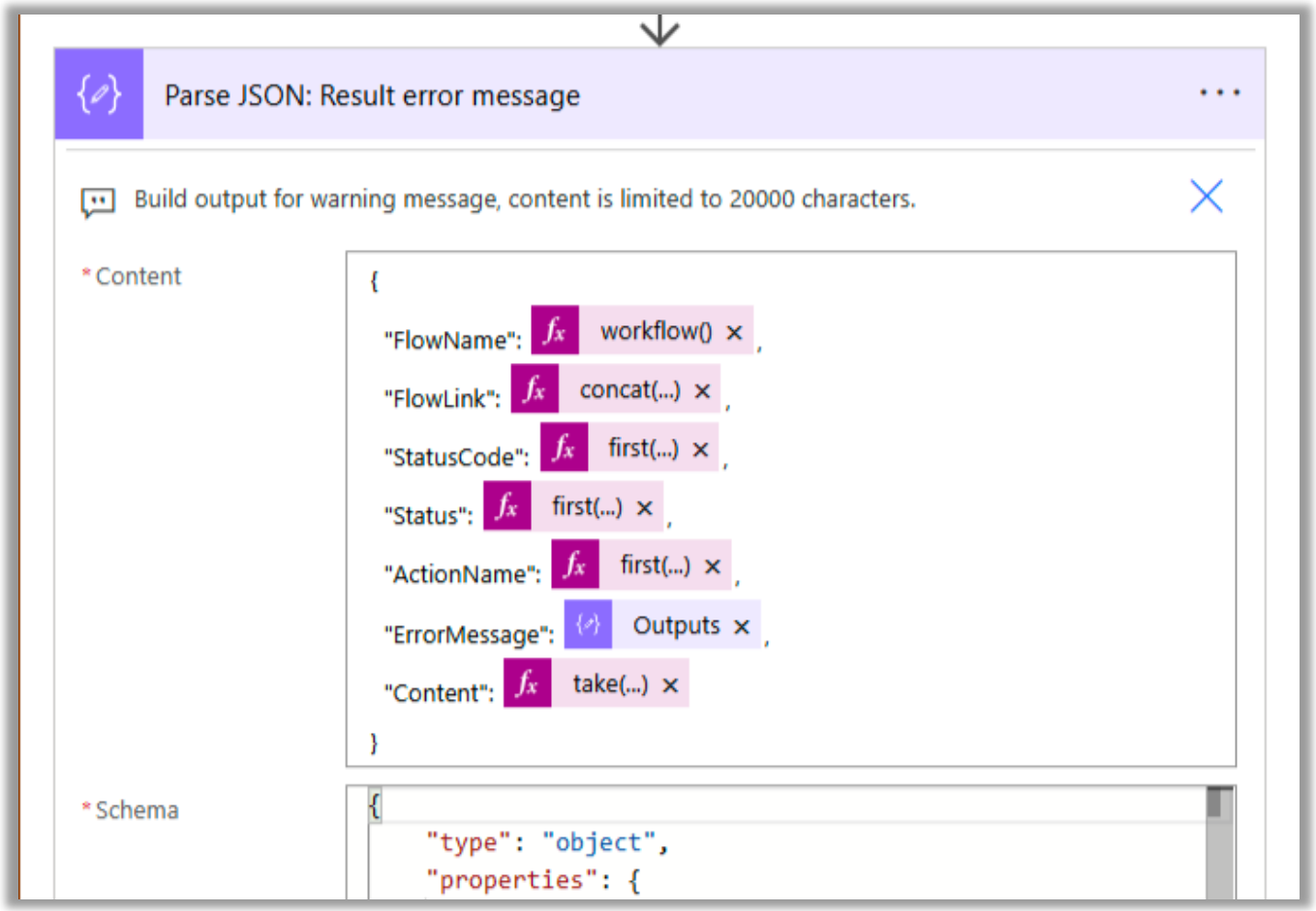
StatusCode

```
first(xpath(outputs('Compose:_Convert_to_XML'), '//code/text()'))
```

Status

```
first(xpath(outputs('Compose:_Convert_to_XML'), '//status/text()'))
```

CATCH SCOPE – Commands (2)



ActionName

```
first(xpath(outputs('Compose:_Convert_to_XML'), '//name/text()'))
```

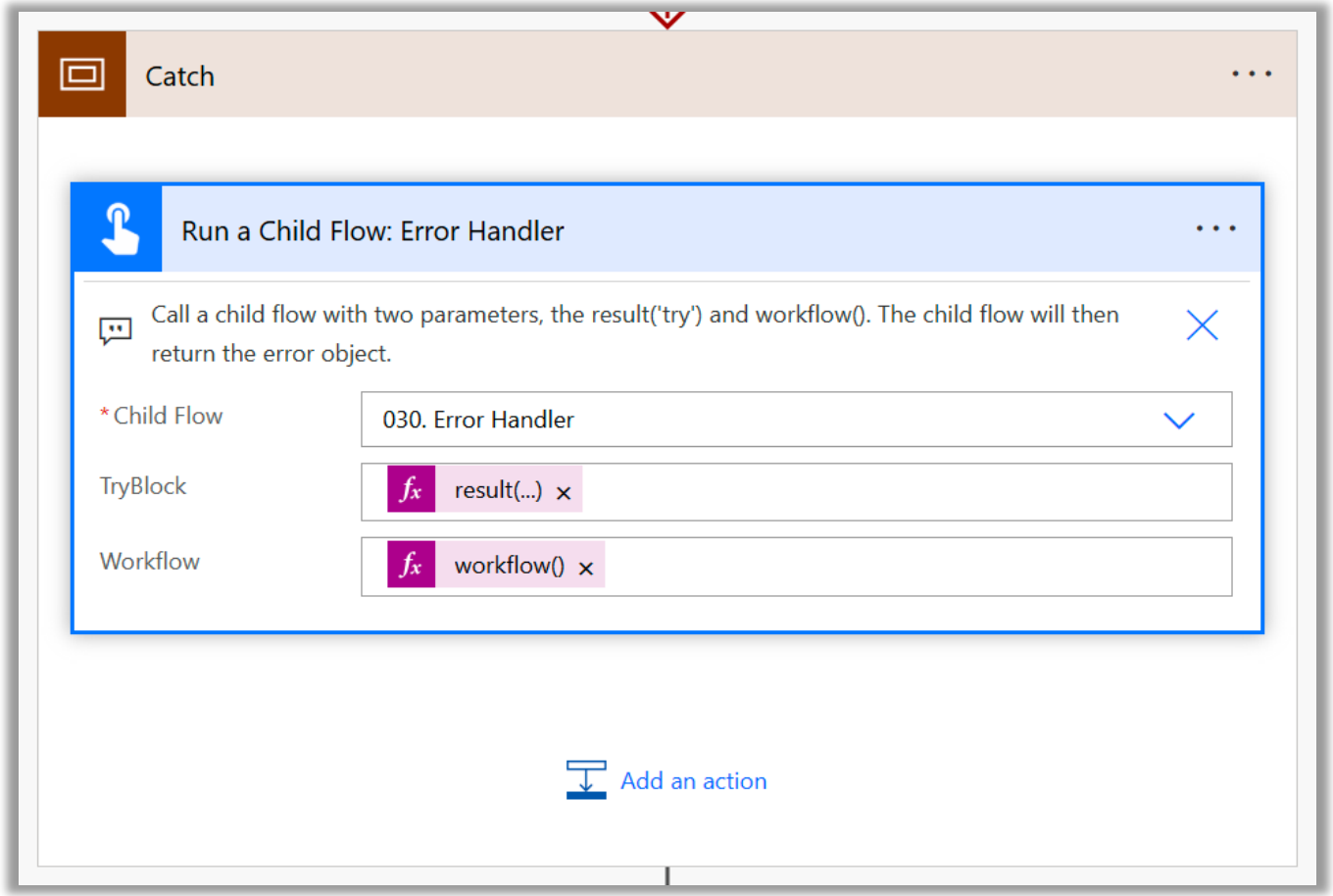
ErrorMessage

```
outputs('Compose:_Get_error_message_using_XPath')
```

Content

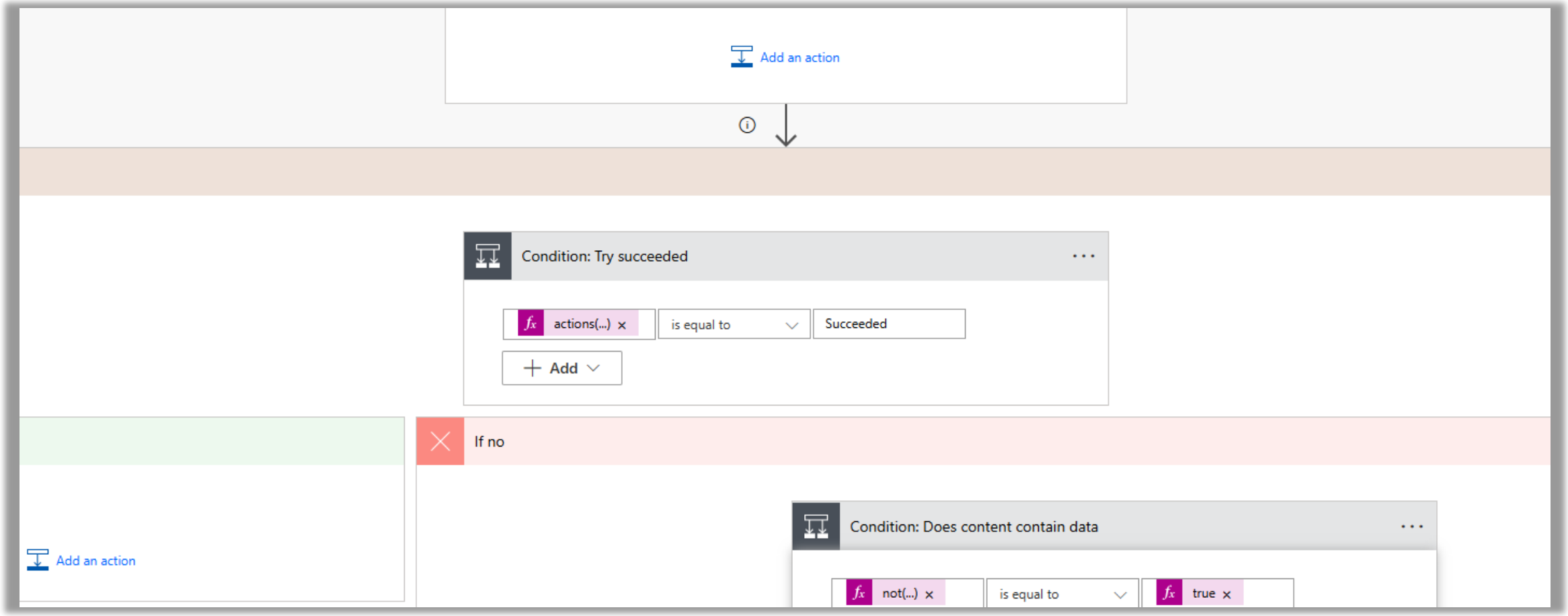
```
take(string(first(body('Filter_array:_Get_failed_step'))), 20000)
```

CATCH SCOPE – Child flow

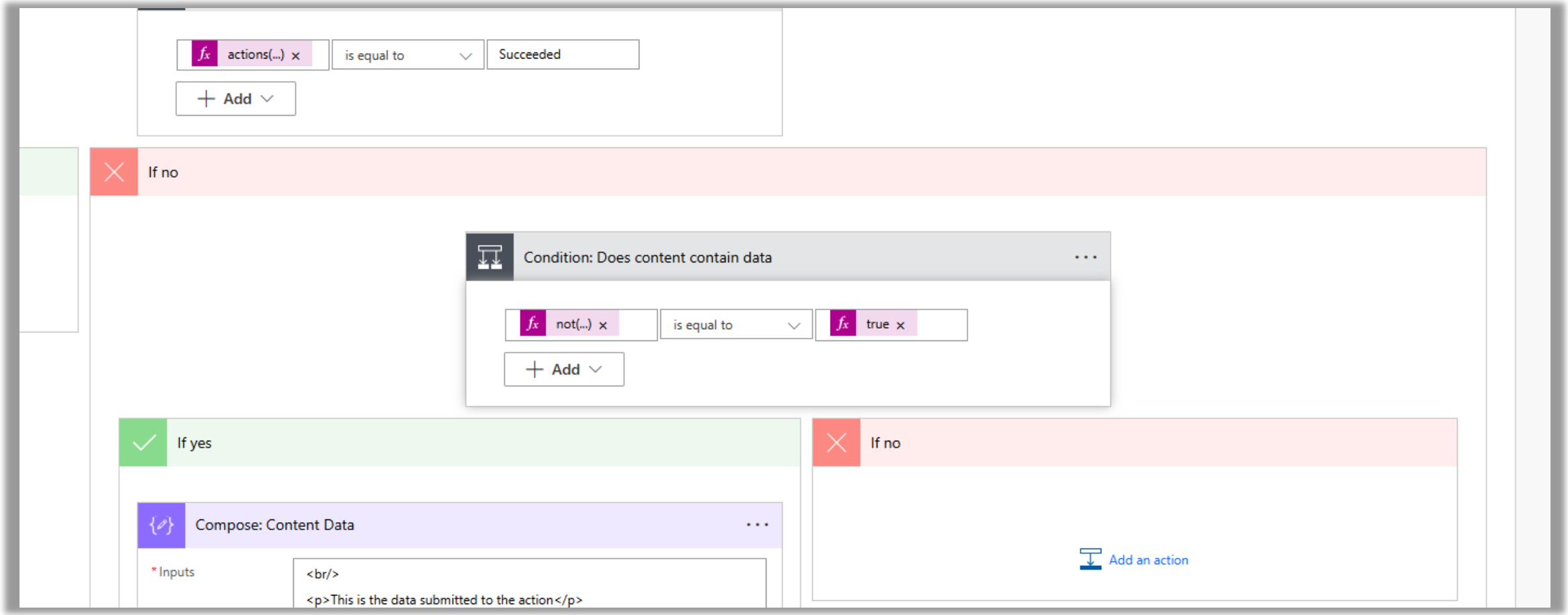


Make the Catch logic into a child flow that can be re-usable across all of your flows.

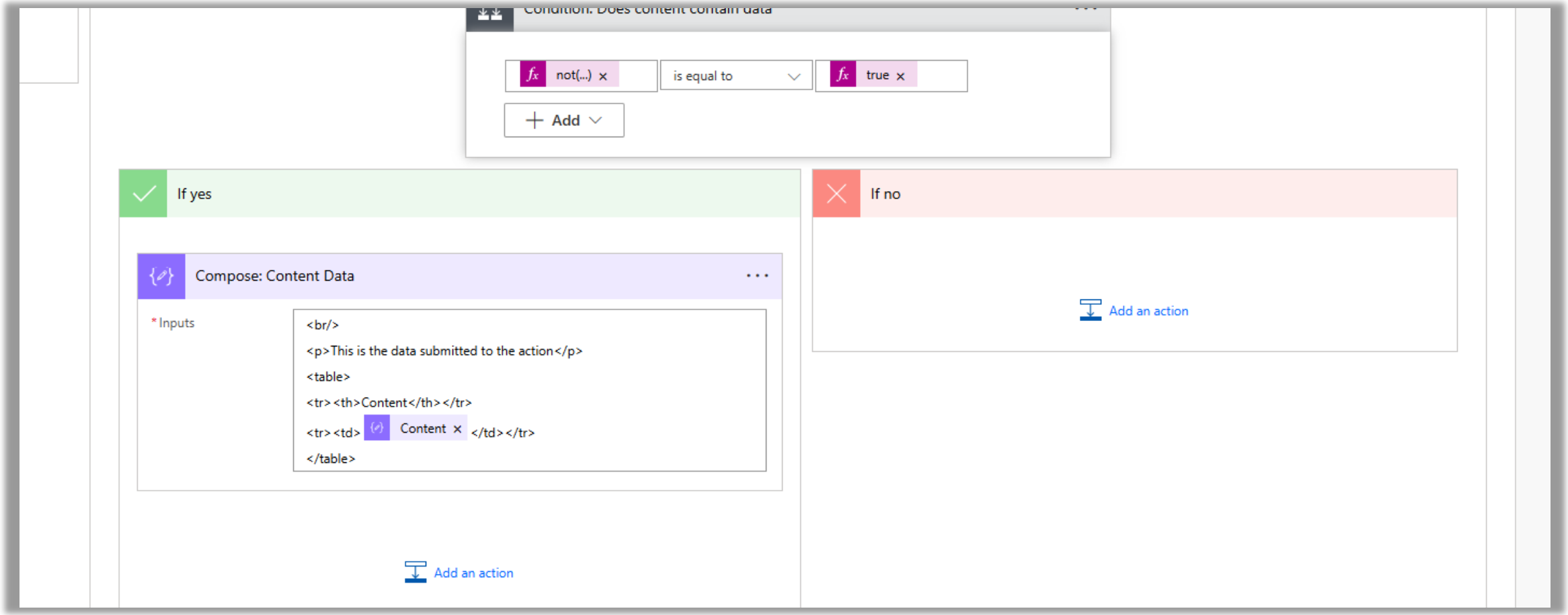
FINALLY SCOPE – Was Try successful?



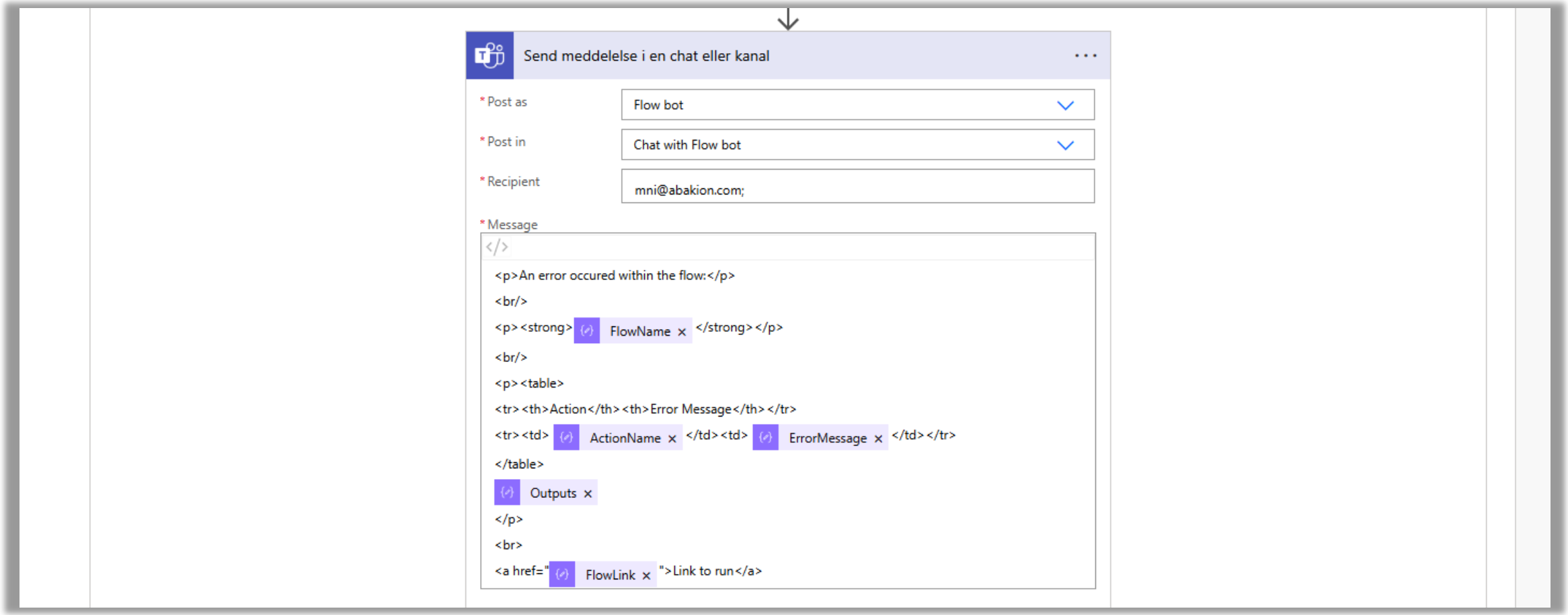
FINALLY SCOPE – Try failed, is there content



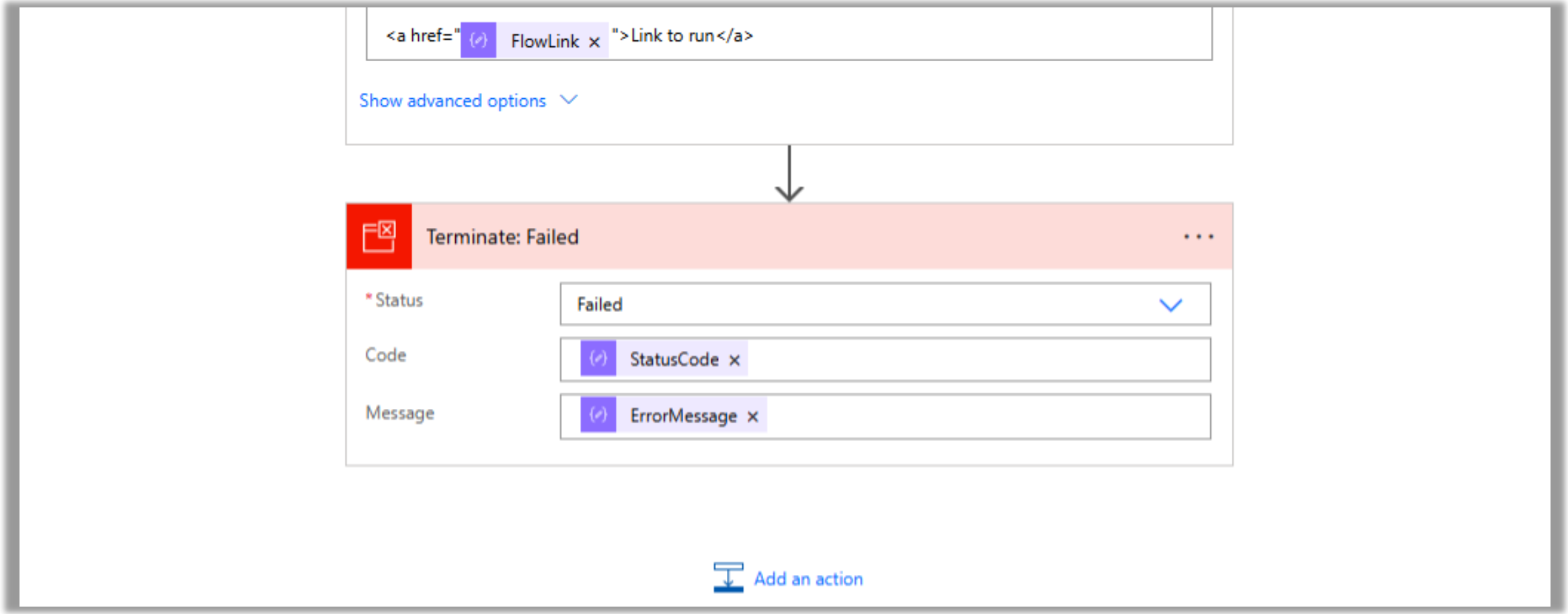
FINALLY SCOPE – Format content



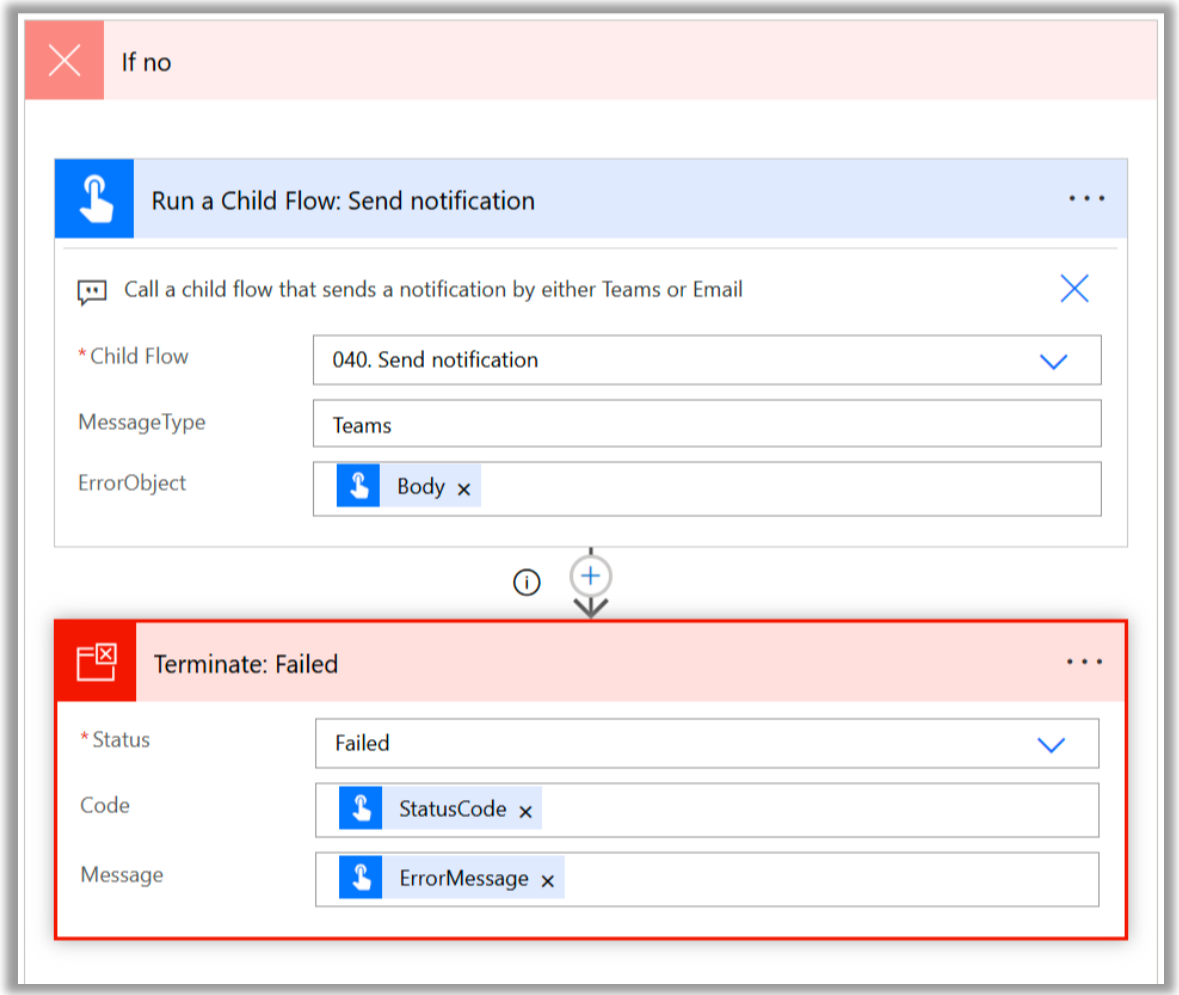
FINALLY SCOPE – Send Teams message



FINALLY SCOPE – Terminate as failed



Child flow for sending notification



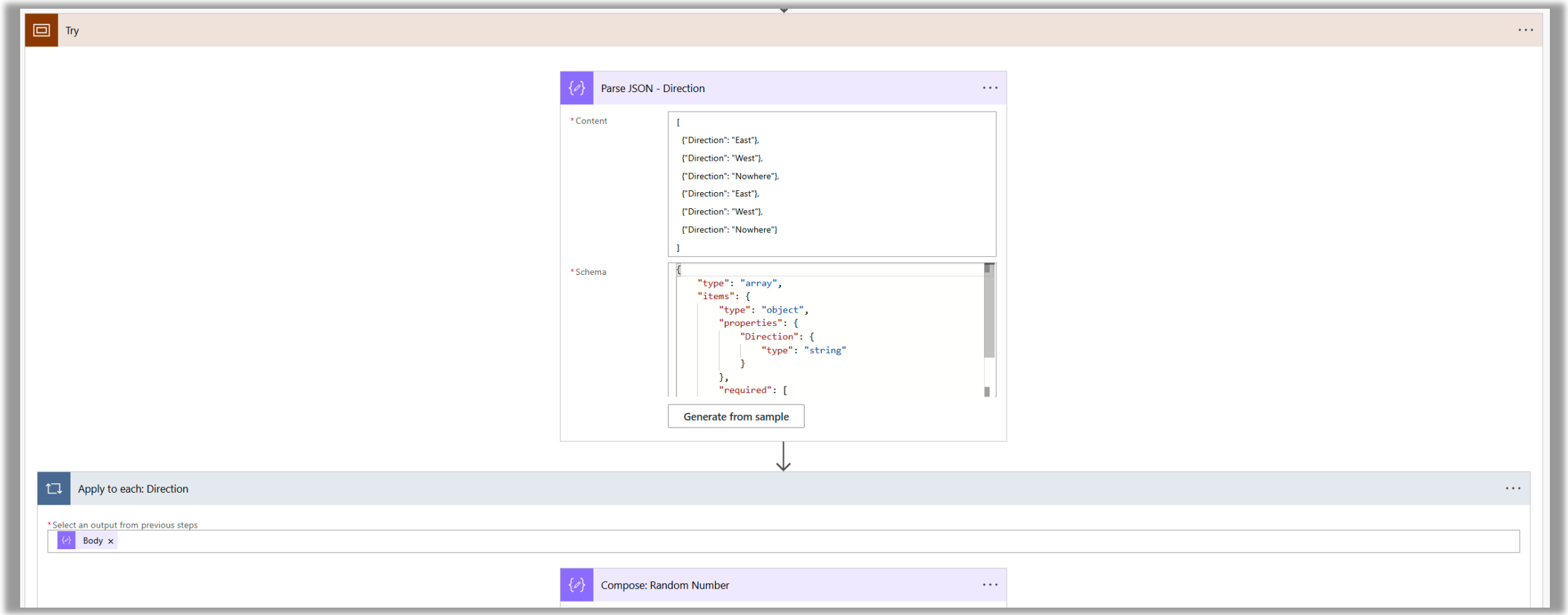
This far

- How to setup Try-Catch-Finally
- How to extract the error message and related details
- Use dynamic content to create custom error messages

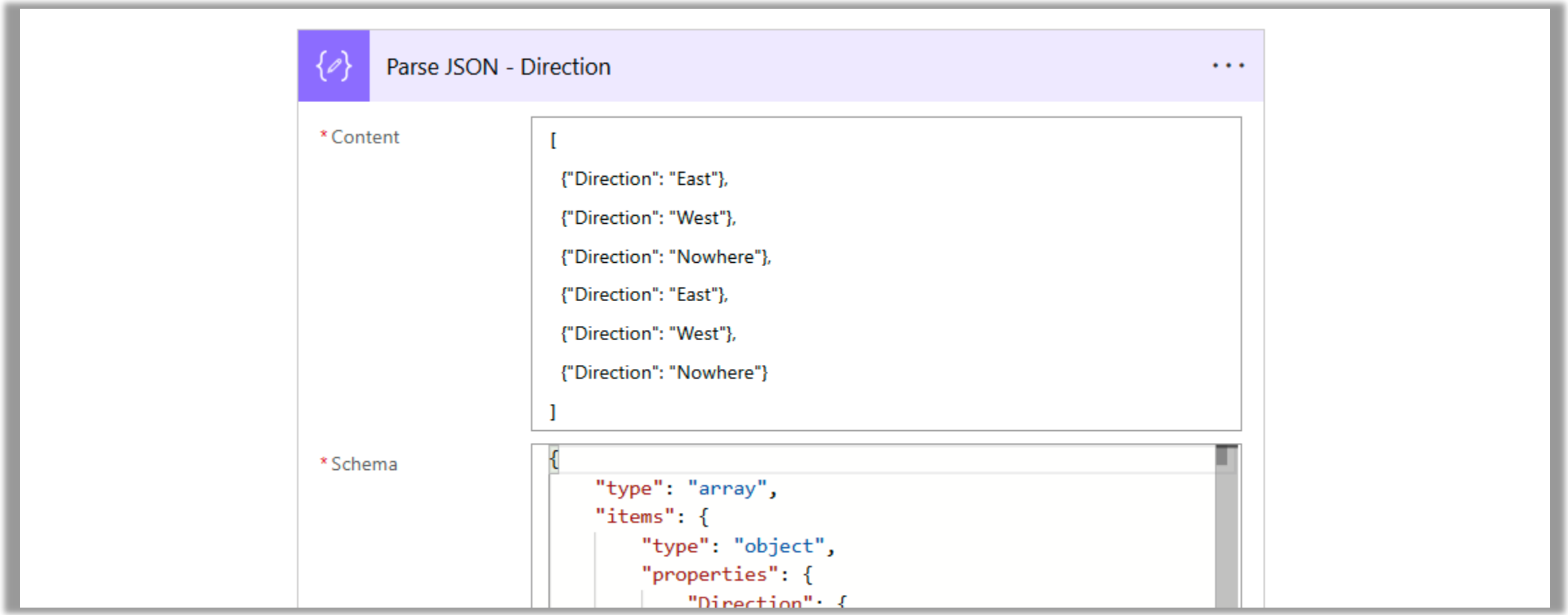
What is not covered in this kind of error handling

- Loops and conditions are difficult to handle

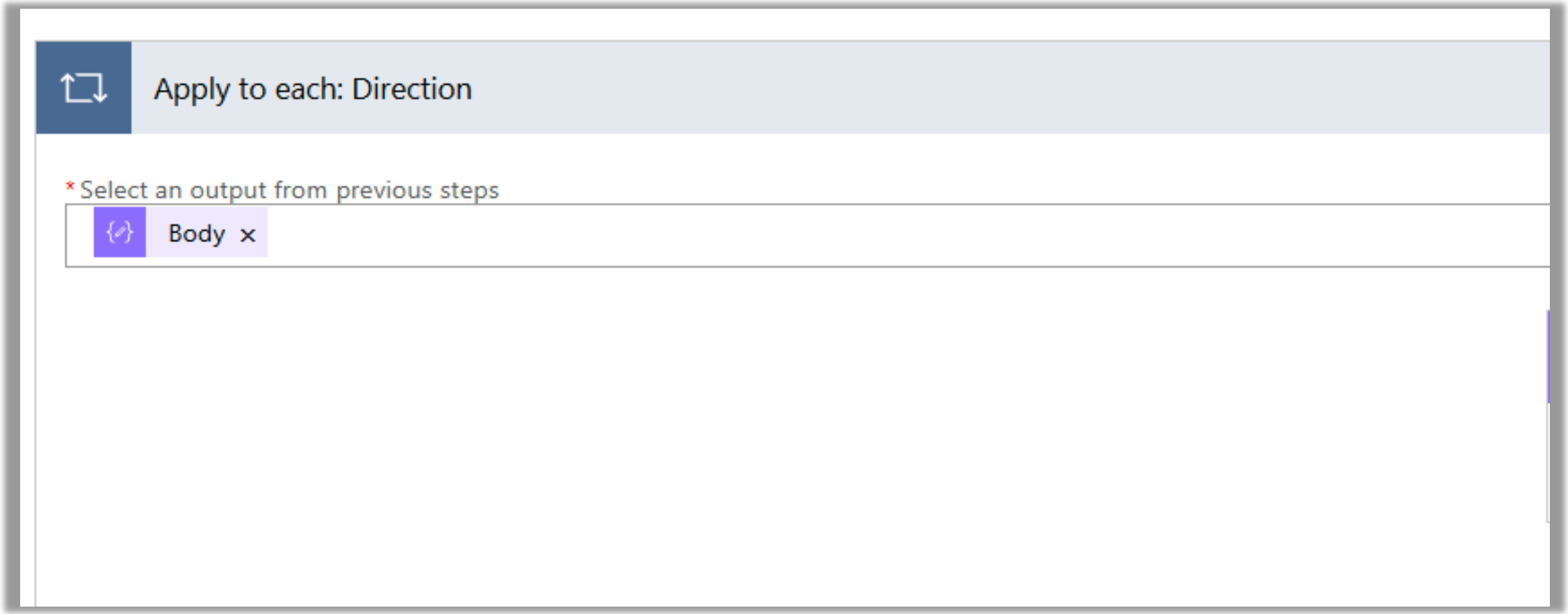
Conditions and nested actions



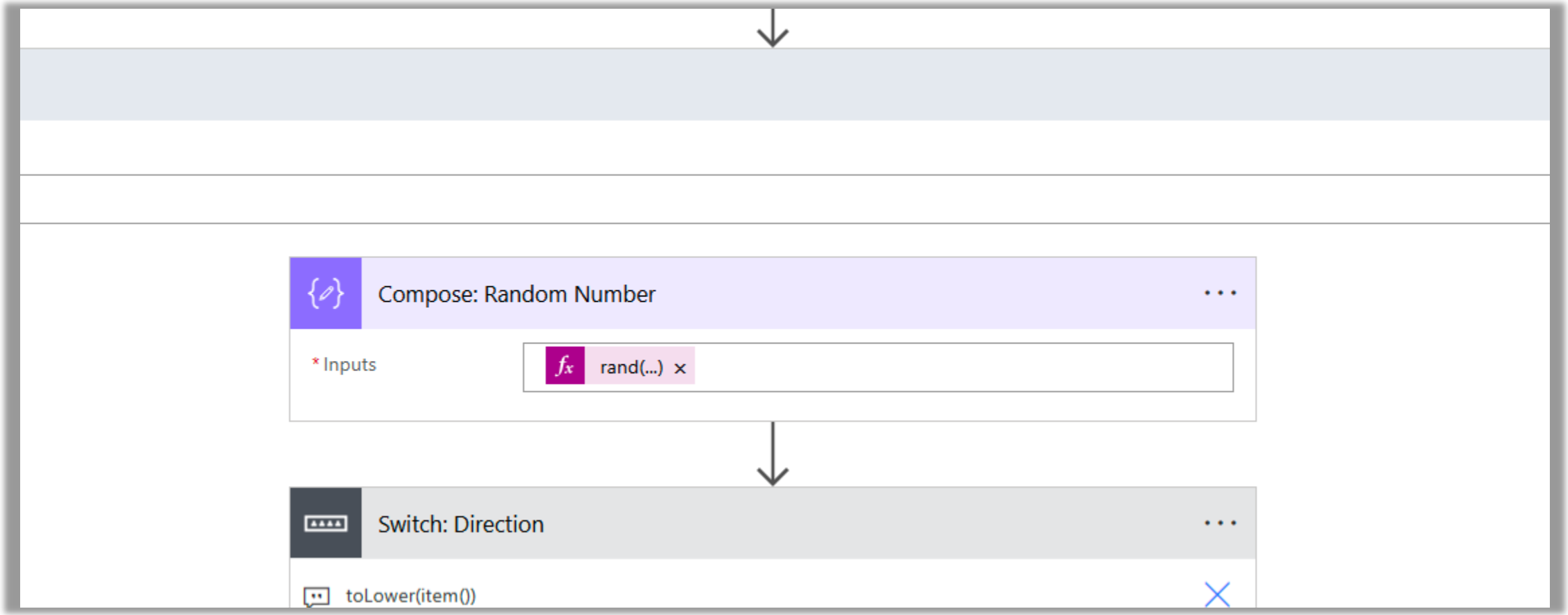
Add a JSON array



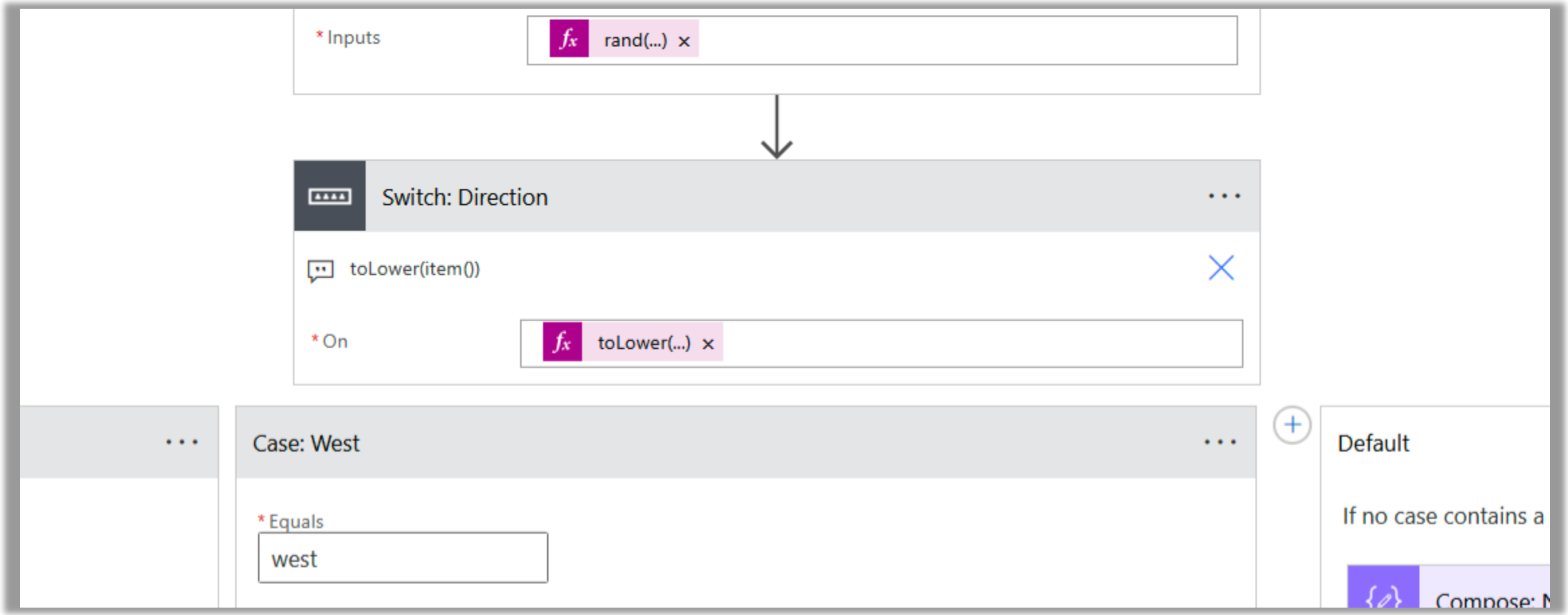
Apply to each element of the JSON array



Get a random number



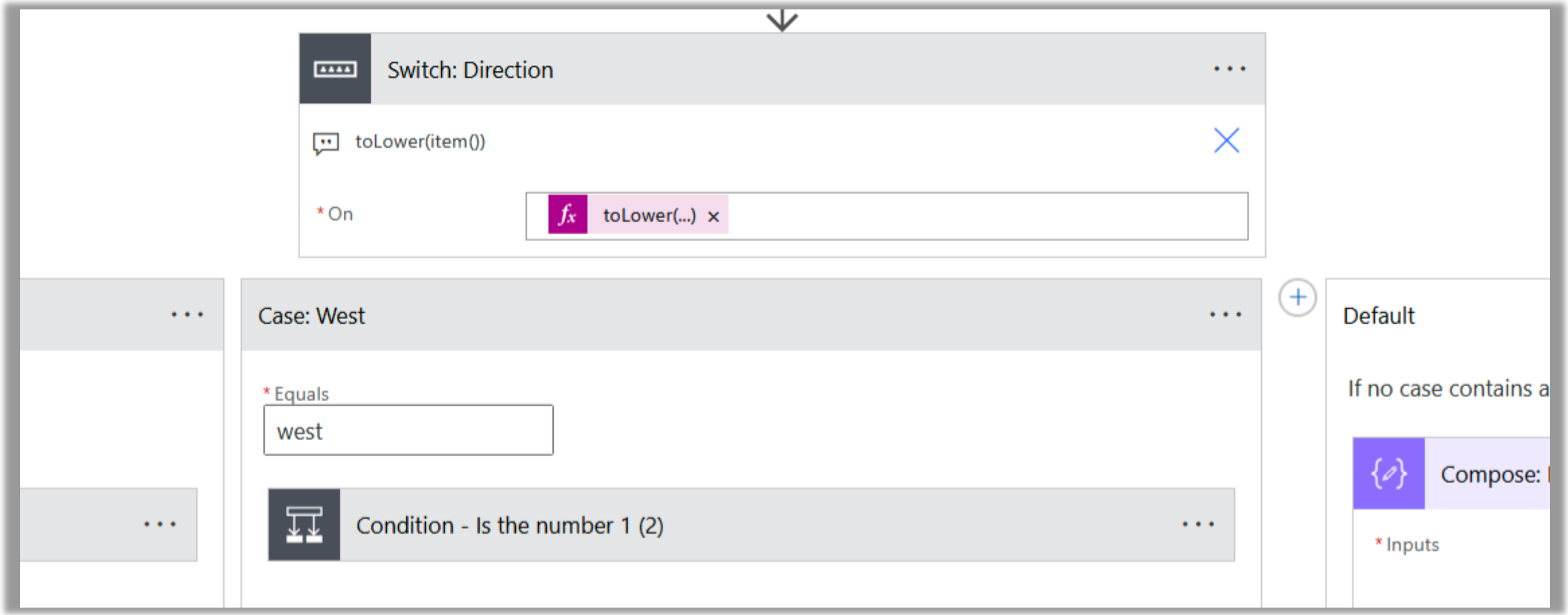
Switch on the current item of the for each loop



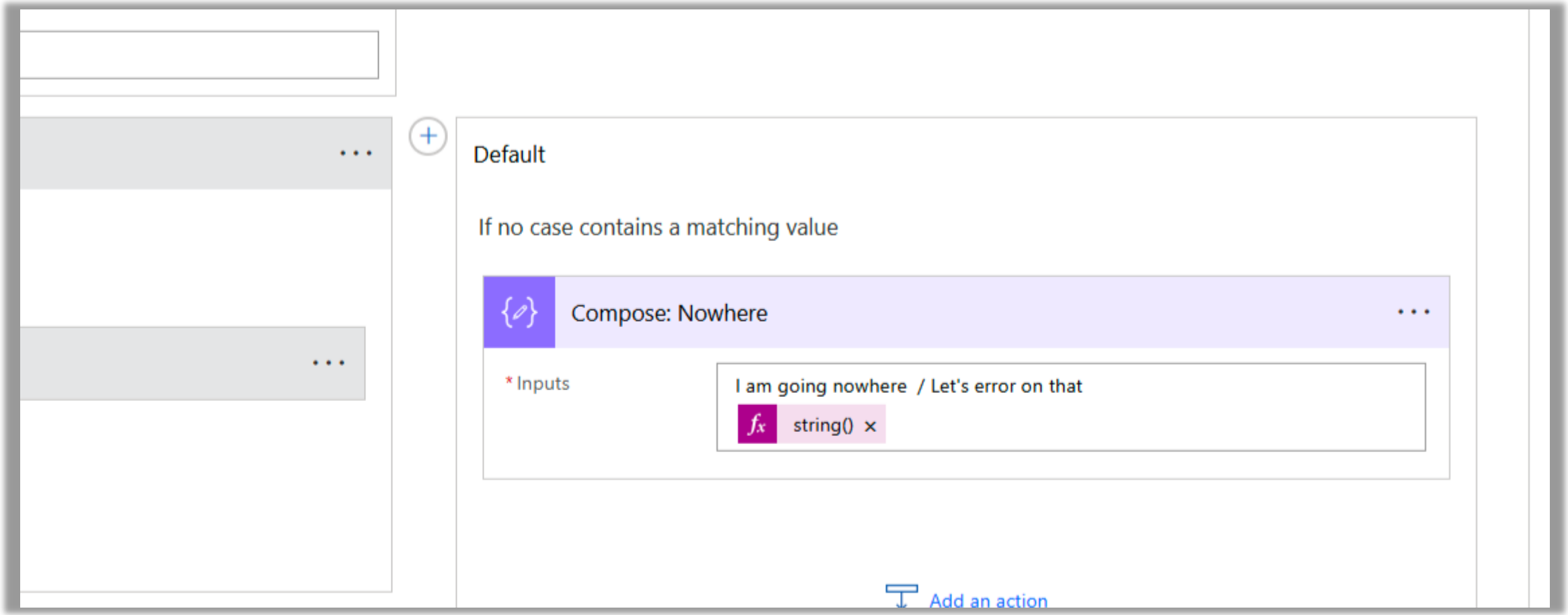
First east


Switch: Direction
toLower(item())
* On
toLower(...)
Case: East
* Equals
east
Condition - Is the number 1 (1)
Case: West
* Equals
west
Condition - Is the number 1 (2)

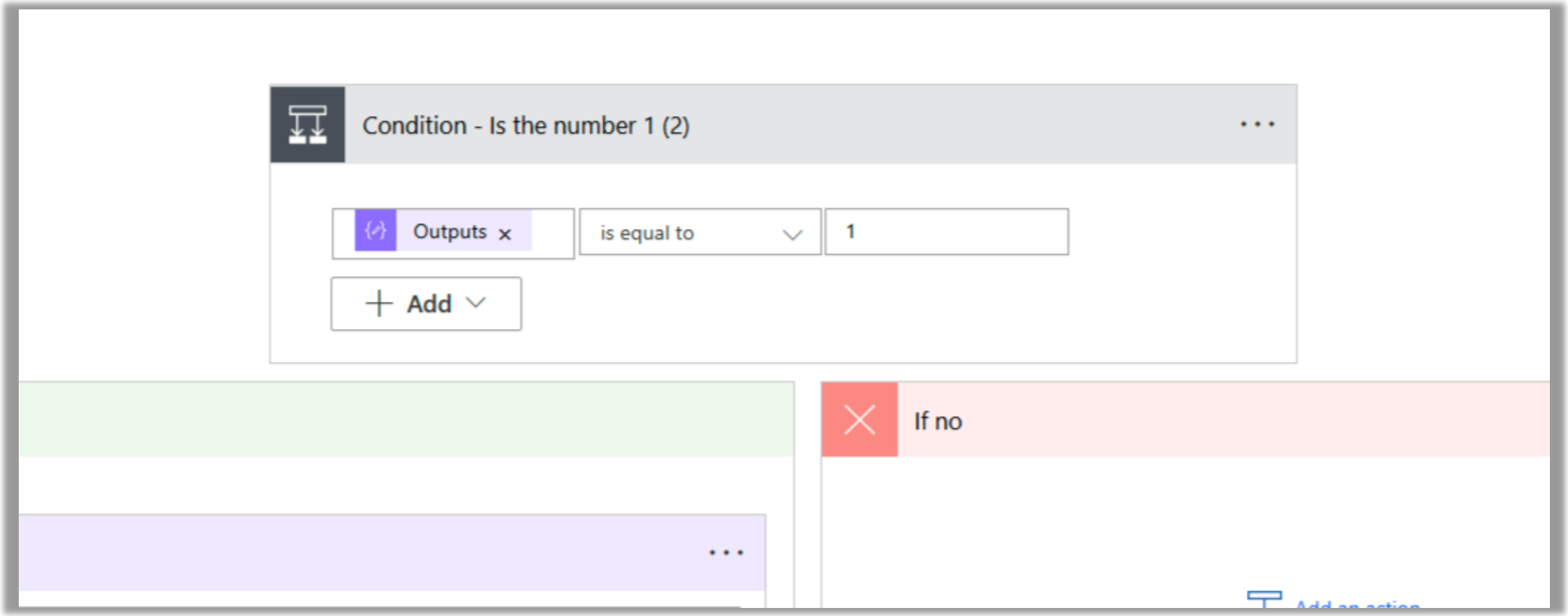
Then west


Switch: Direction
toLower(item())
* On
toLower(...)
Case: West
* Equals
west
Condition - Is the number 1 (2)
Default
If no case contains a
Compose:
* Inputs

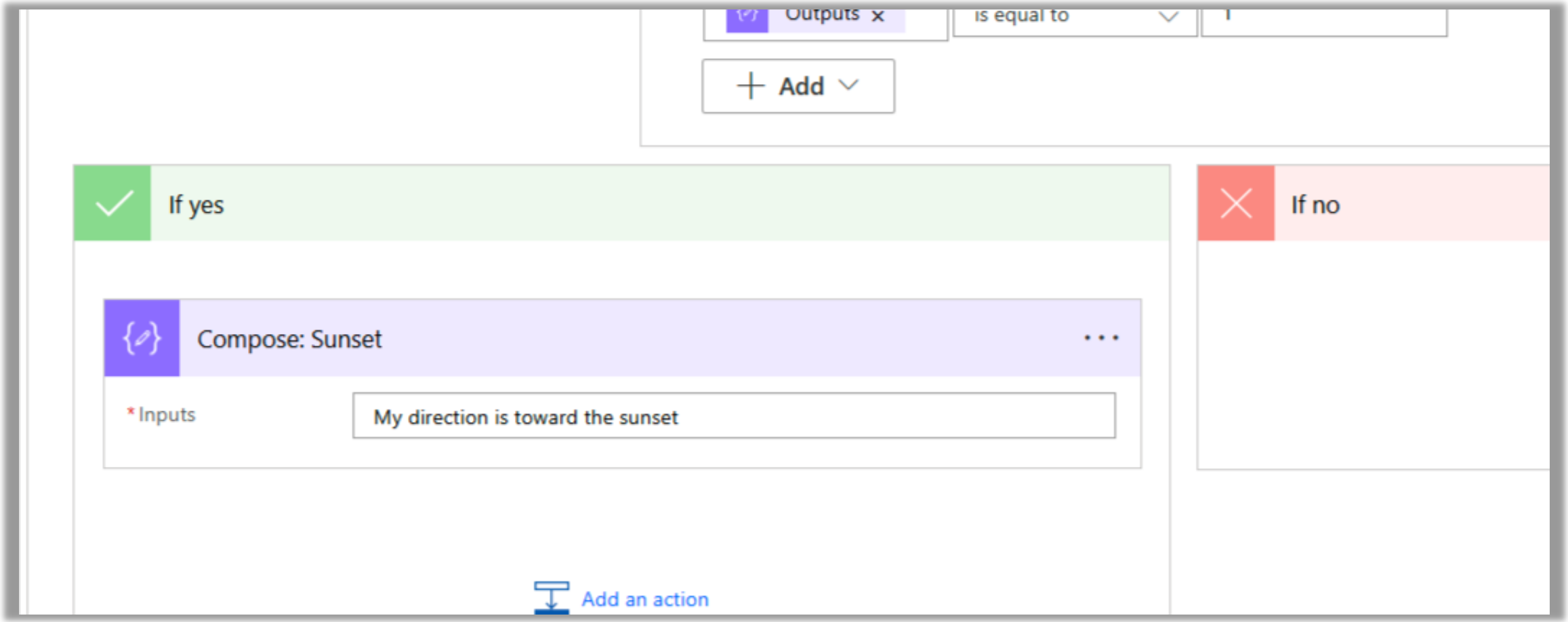
Or nowhere


Default
If no case contains a matching value
Compose: Nowhere
* Inputs
I am going nowhere / Let's error on that
string()
Add an action

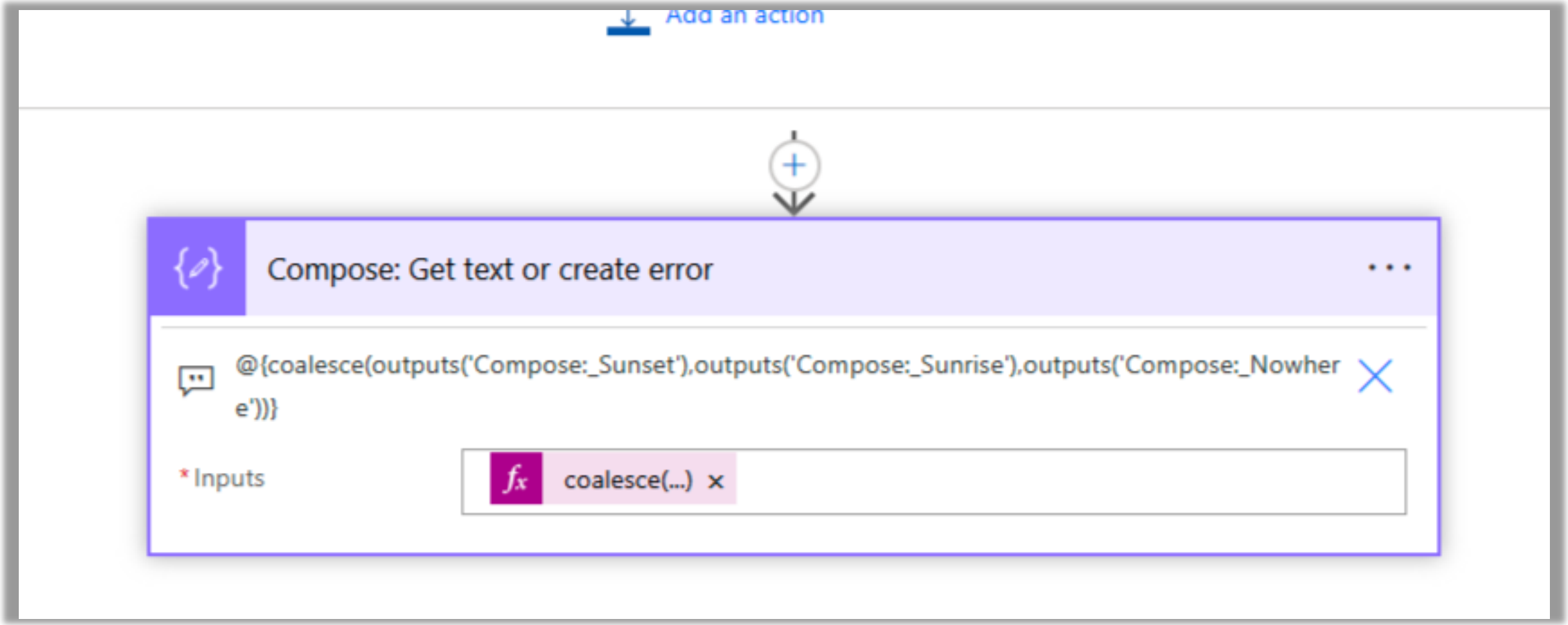
Inside the switch is a condition



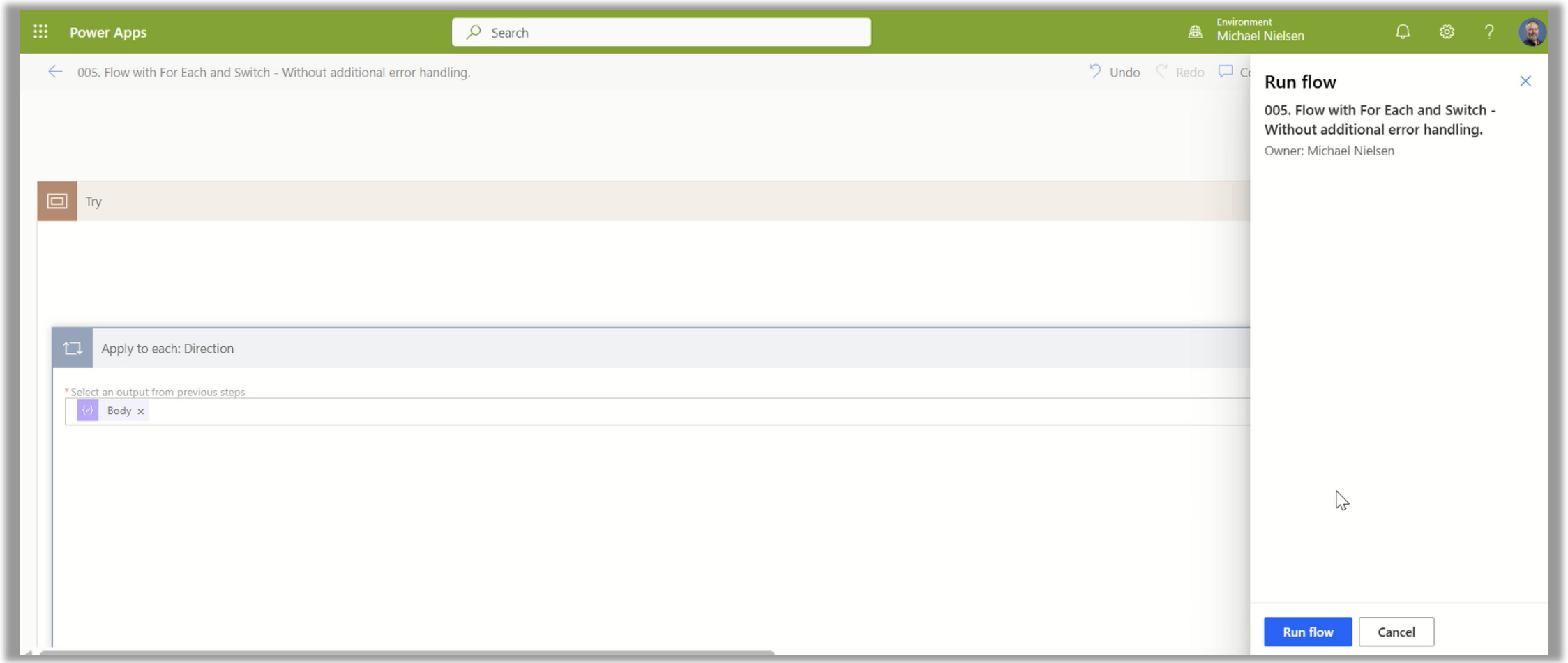
And inside the condition is a Compose



And finally a compose to get the text from either branch



Lets try to run the flow with nested actions



An action failed. No dependent actions succeeded.

An error occured within the flow:

005. Flow with For Each and Switch - Without additional error handling.

Action	Error Message
Apply_to_each:_Direction	An action failed. No dependent actions succeeded.

This is the data submitted to the action

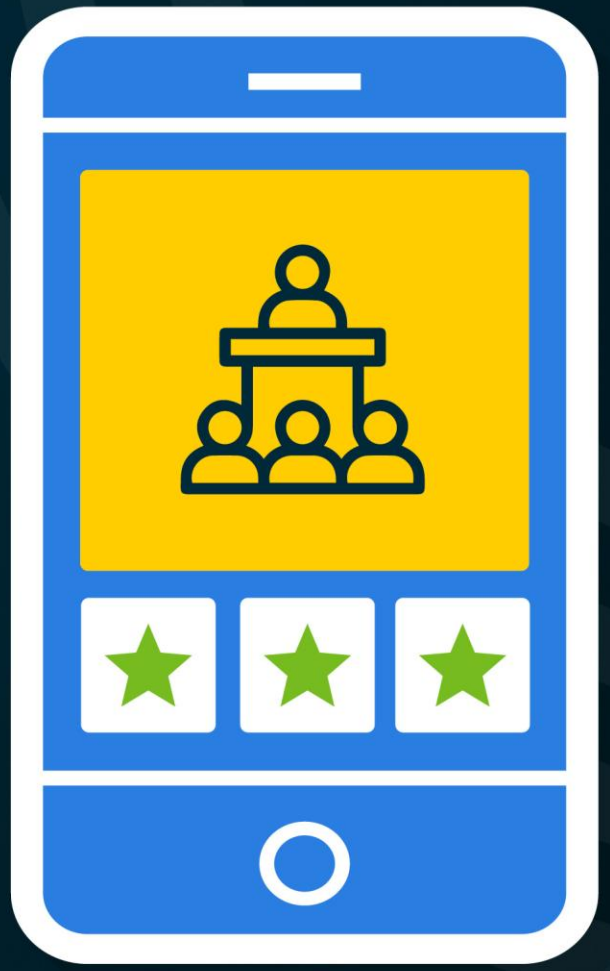
Content
{"name":"Apply_to_each:_Direction","inputs":{"foreachItems":[{"Direction":"East"},{"Direction":"West"},{"Direction":"Nowhere"},{"Direction":"East"},{"Direction":"West"},{"Direction":"Nowhere"}]},"inputsMetadata":{"foreachItemsCount":6},"startTime":"2025-06-18T22:51:23.3137558Z","endTime":"2025-06-18T22:51:26.1082844Z","trackingId":"f3f6430d-da8c-44d5-a160-c4281fe962b9","clientTrackingId":"08584513198026585602344733126CU216","clientKeywords":["testFlow"],"code":"ActionFailed","status":"Failed","error":{"code":"ActionFailed","message":"An action failed. No dependent actions succeeded."}}

Link to run

Lets look at it live

Lets connect

Please rate this session on the app



cvent

GET IT ON Google Play

Download on the App Store