Azure Data Factory – Challenge Lab

Objectives

- Basic use of Azure Data Factory activities

Technologies

This lab post assumes you have a Microsoft Azure subscription and pre-requisite resources like Azure Data Factory

Step-by-Step Instructions

Instantiate Resources if you haven't already

First, we'll quickly run through creation of the basic resources we'll need to complete this exercise. If you have existing Azure resources in your subscription, consider using them if not follow below steps to create required resources.

Resource Group

https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/manage-resource-groups-portal

On the "Create a resource group" blade, populate the following:

- Subscription … self-explanatory
- Resource Group … choose a name that is meaningful for you (and aligned with your naming standards)
- Region … select a region appropriate for your situation; take into consideration that some regions {e.g. West US and East US} see higher demand than others

Click the "Review + create" button and after validation, click the "Create" button. Allow time for processing.

Azure Data Factory

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-portal

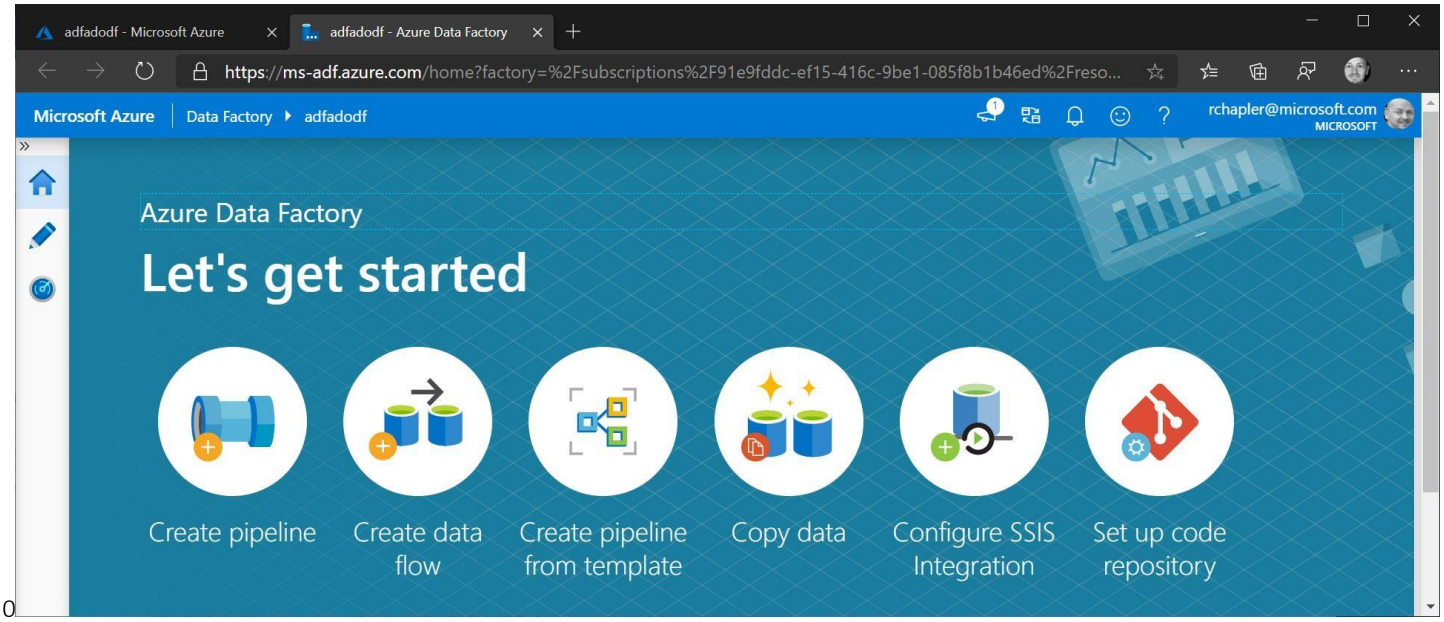
On the “New data factory” blade, populate the following:

- Name … choose a name that is meaningful for you (and aligned with your naming standards)
- Version … confirm default selection, “V2”
- Subscription … self-explanatory
- Resource Group … select the resource group created in the prior step
- Location … select a region appropriate for your situation; take into consideration that some regions {e.g. West US and East US} see higher demand than others
- Enable Git … we will, for now, un-check this box

Click the “Create” button. Allow time for processing.

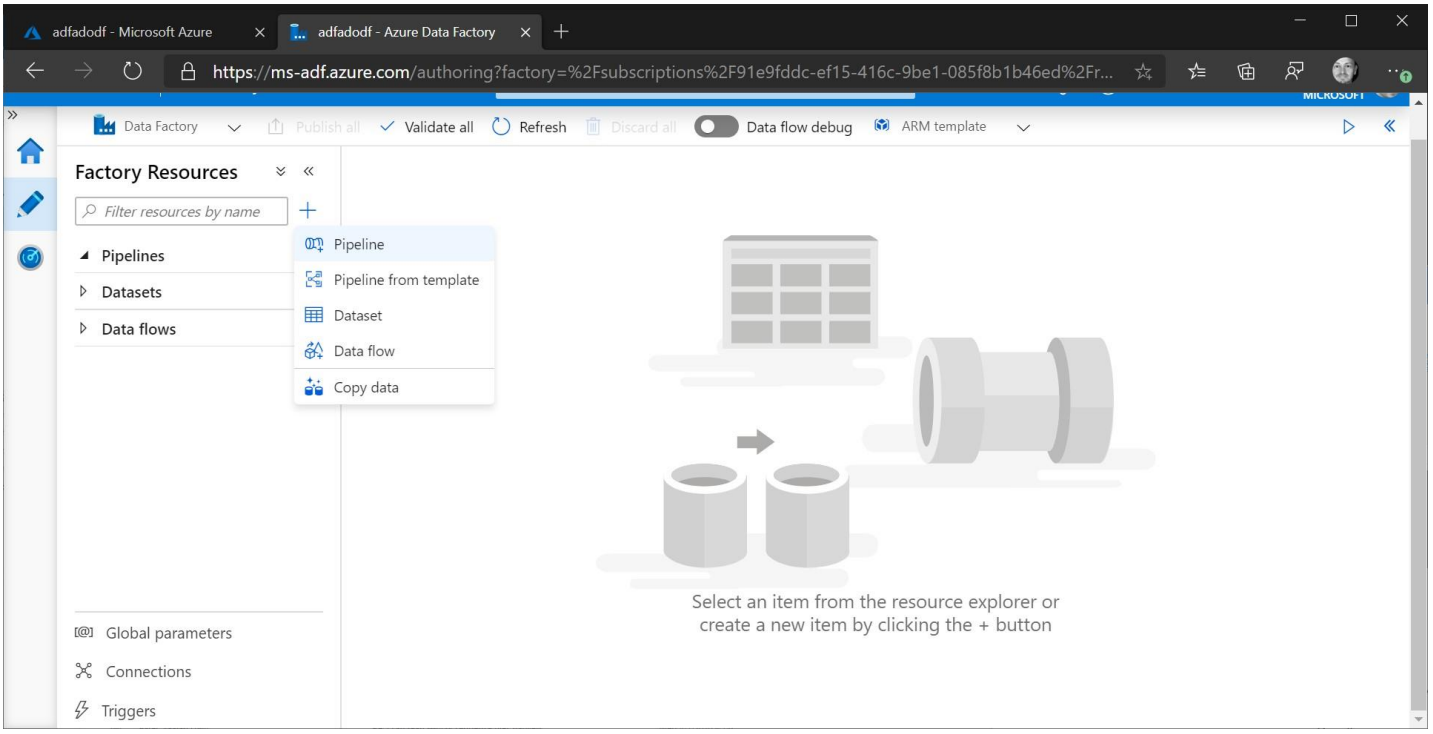
Create Sample Application

Navigate to your data factory.



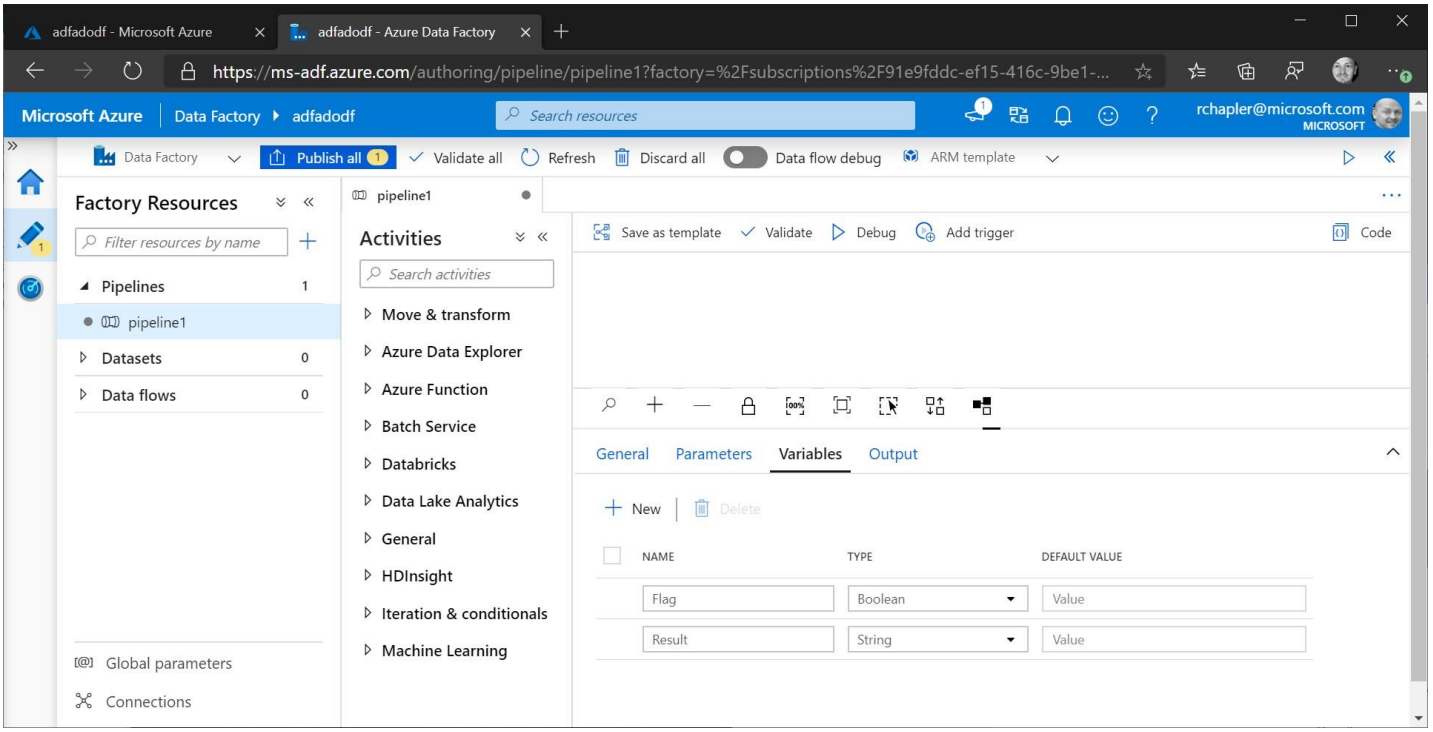
0

Click the Author button {i.e. pencil icon} on the left sidebar.



Pipeline

Click the “+” icon to the right of the “Filter resources by name” input. Select Pipeline from the resulting dropdown.

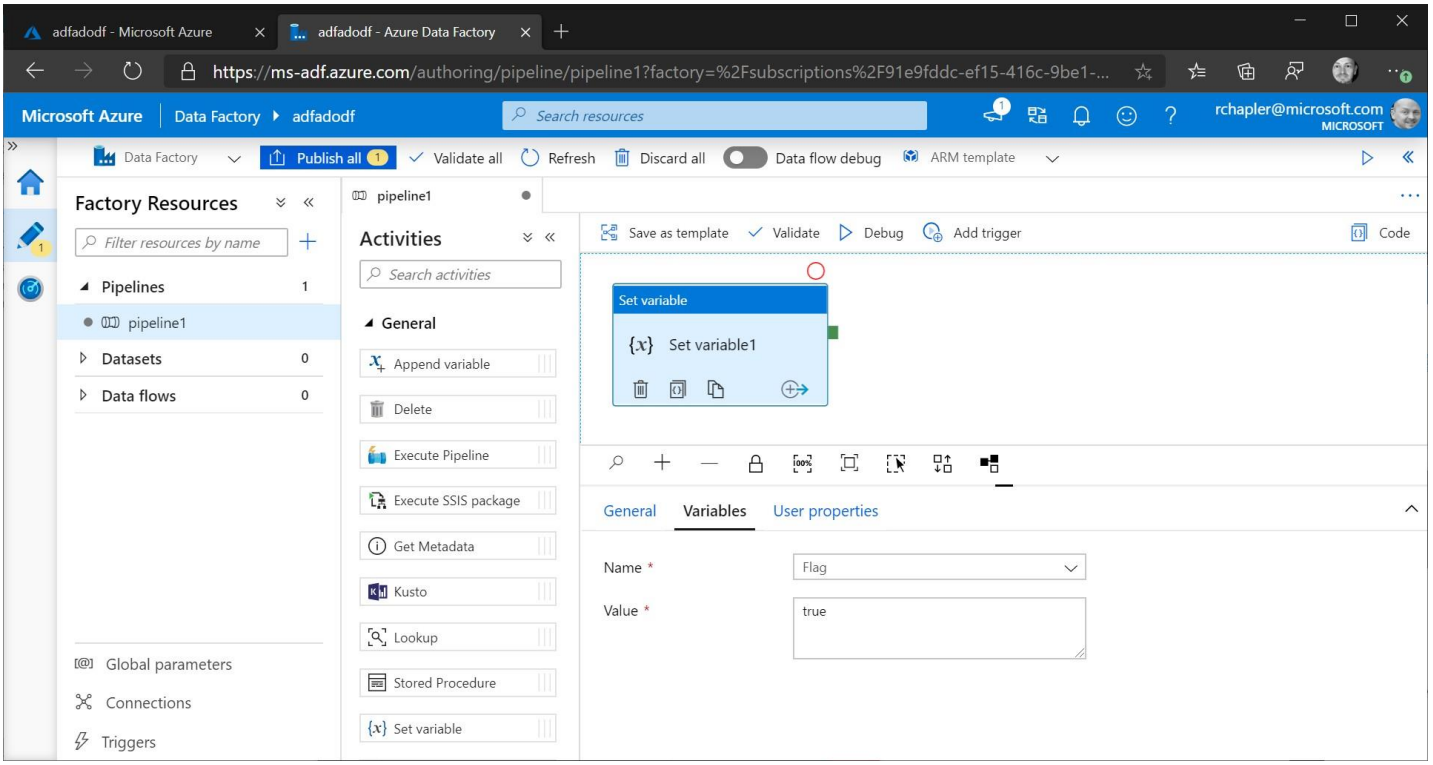


Click on the “Variables” tab. Click “+ New” and add the following two variables:

- Flag (Boolean) … will serve as a TRUE / FALSE trigger for the conditional result
- Result (String) … for capturing an anecdotal result {e.g. “Success!”}

Activity #1, Set Variable

Expand “General” in the Activities bar. Drag-and-drop a “Set variable” component into the pipeline.

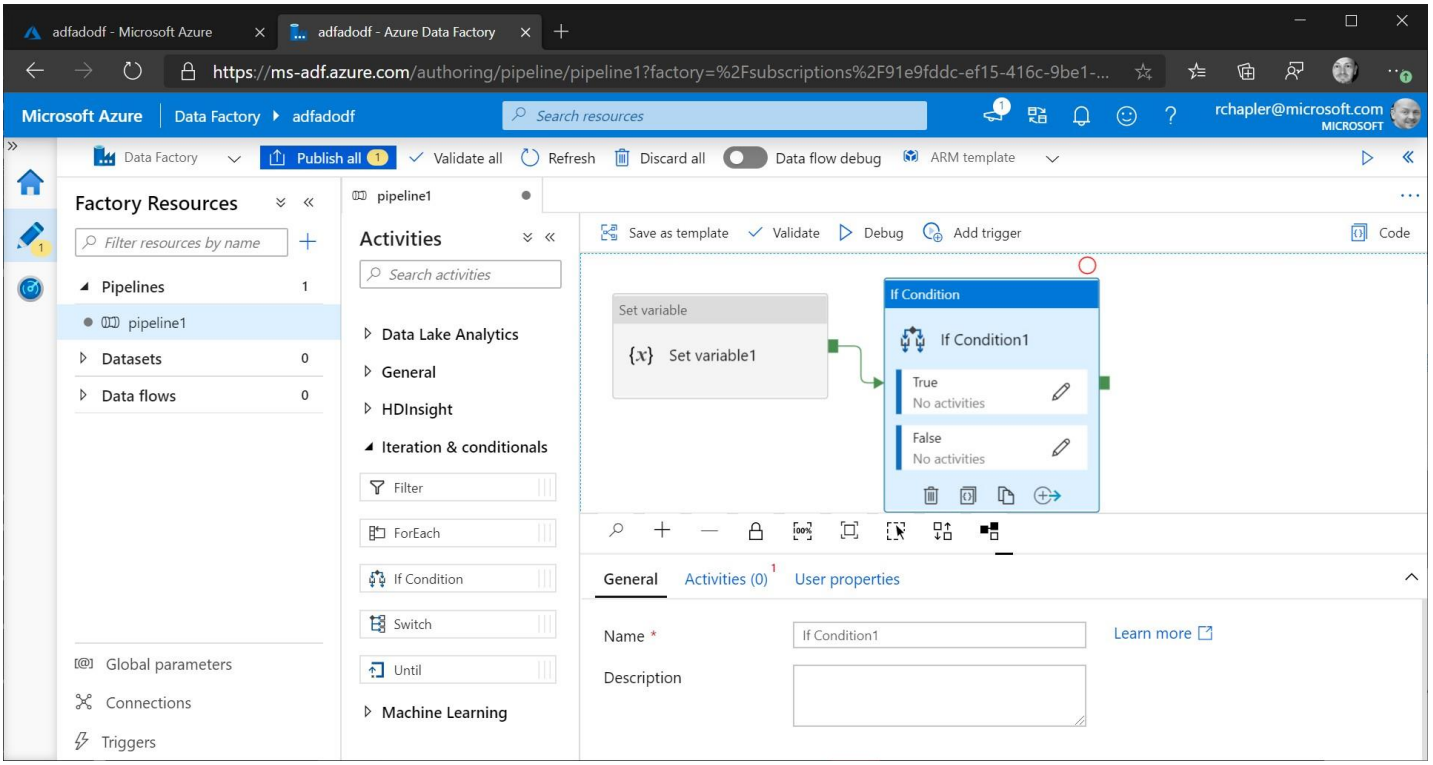


On the “Variables” tab of the “Set variable” component, populate the following values:

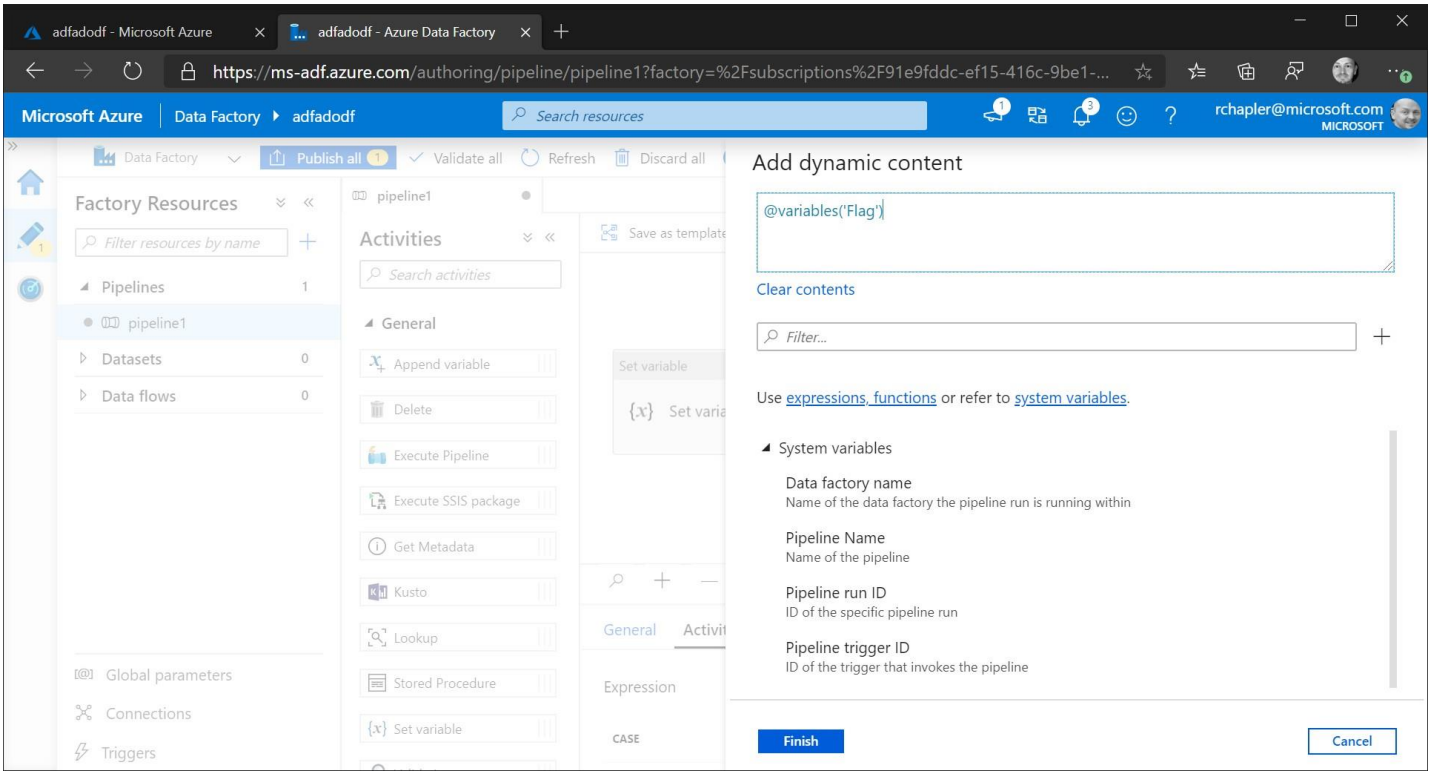
- Name … select “Flag” from the dropdown
- Value … enter “true”

Activity #2, If Condition

Expand “Iteration & conditionals” in the Activities bar. Drag-and-drop an “If Condition” component into the pipeline.

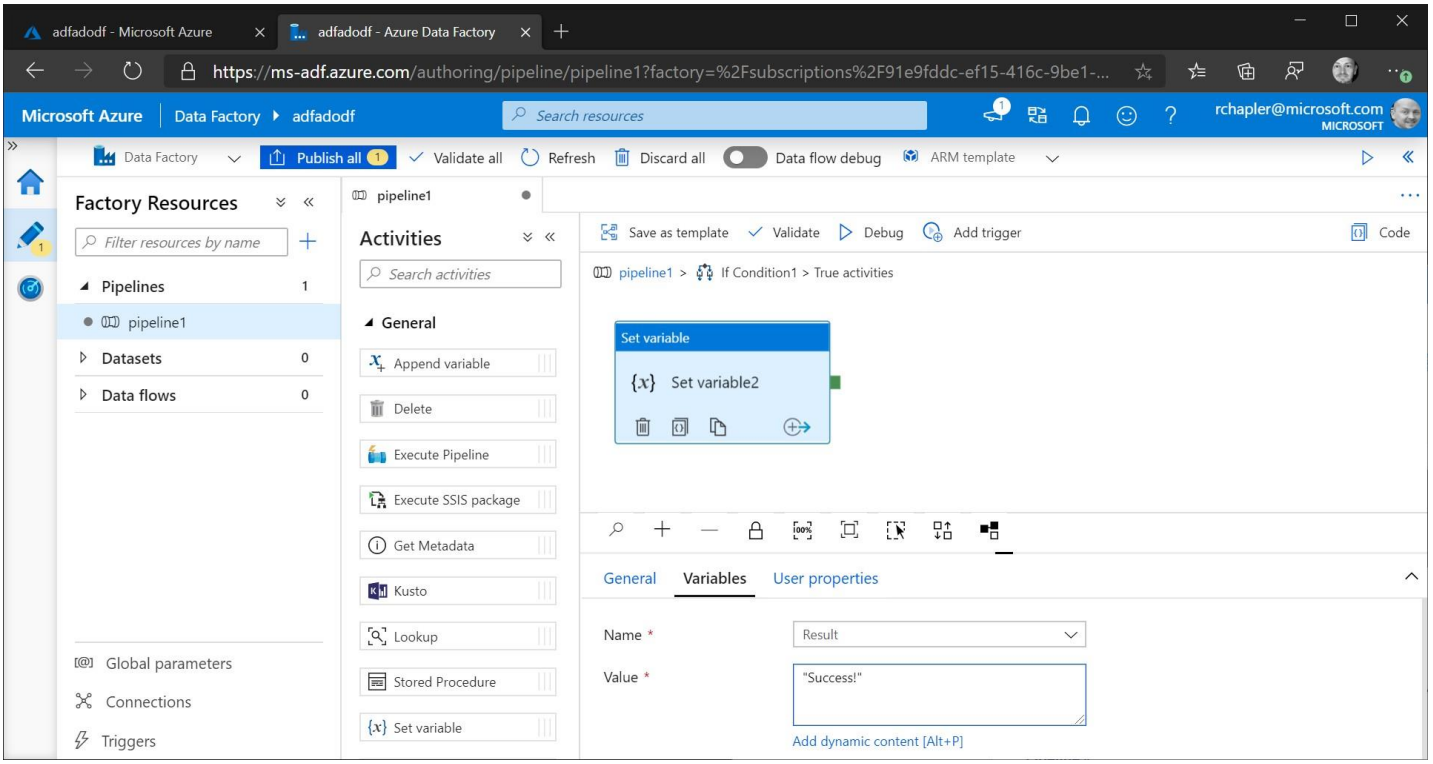


Create a dependency between the “Set variable” and “If Condition” components (by drawing an arrow from one to the other).



On the “Activities” tab, enter the following dynamic content in the Expression textbox: `@variables('Flag')`

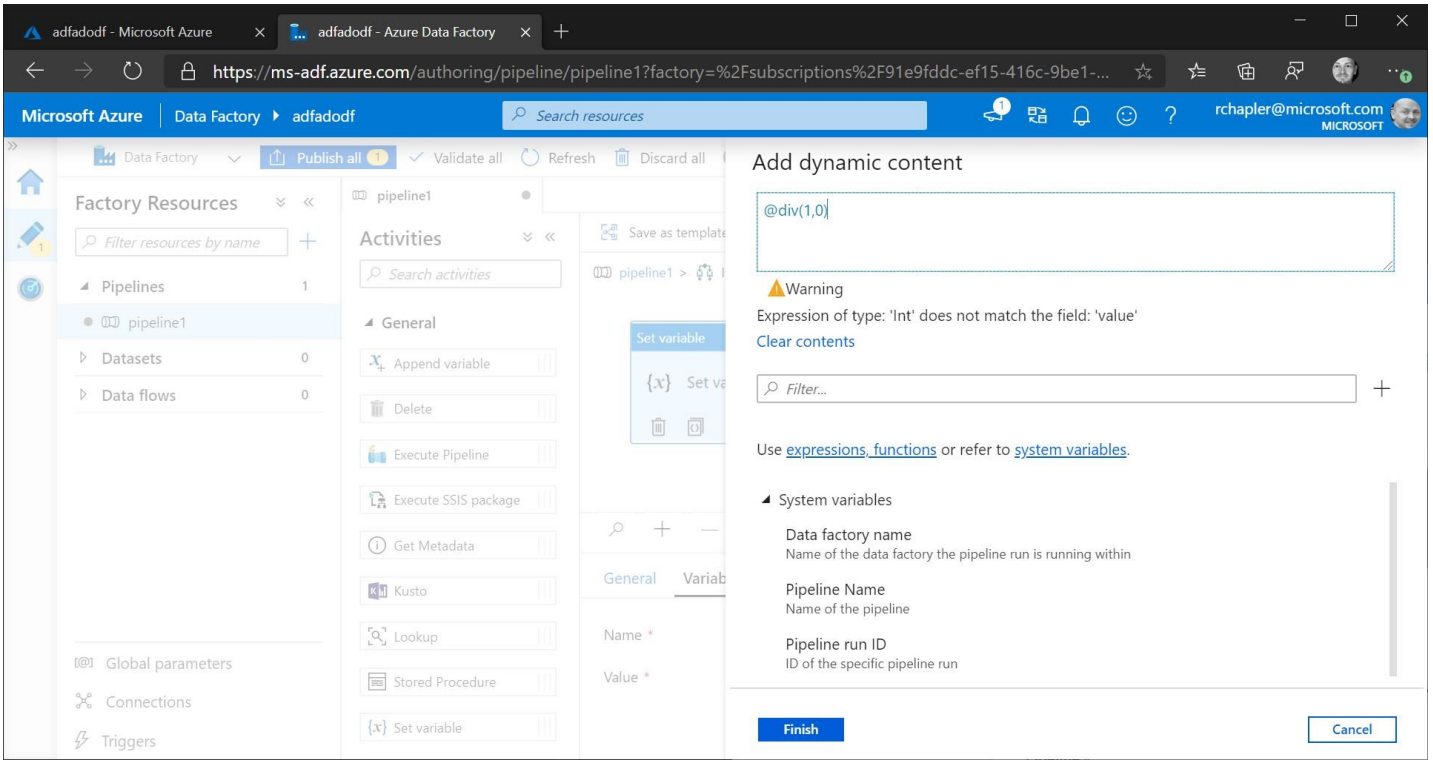
Click “Finish” to close the “Add dynamic content” popout, then click on the pencil icon in the “If Condition” component, “True” selection.



Expand “General” in the Activities bar. Drag-and-drop a “Set variable” component into the activity window.
On the “Variables” tab of the “Set variable” component, populate the following values:

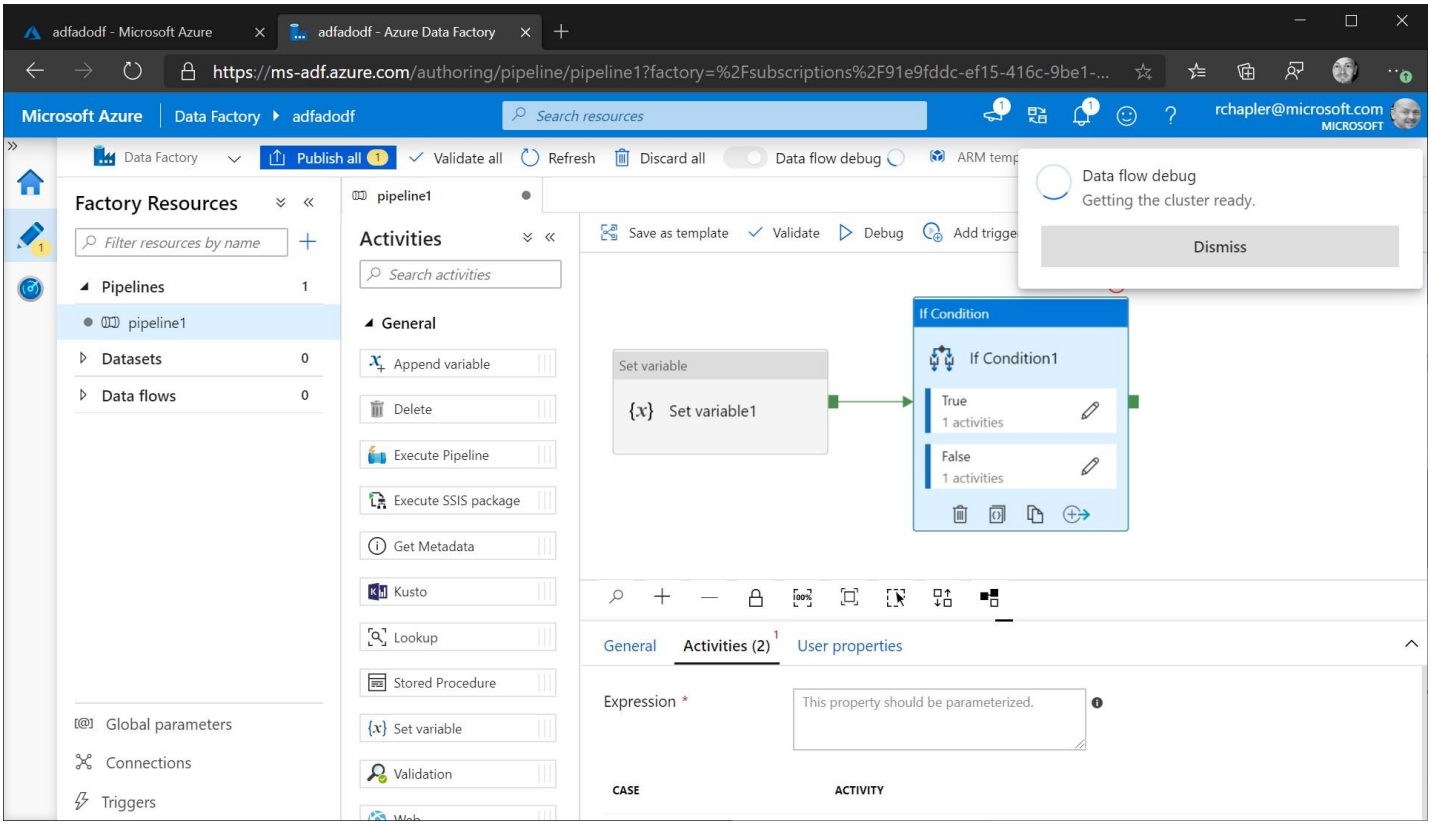
- Name … select “Result” from the dropdown
- Value … enter “Success!”

Repeat for the “If Condition” component, “Fail” selection. In this component, we’ll enter dynamic content “@div(1,0)” for Value.

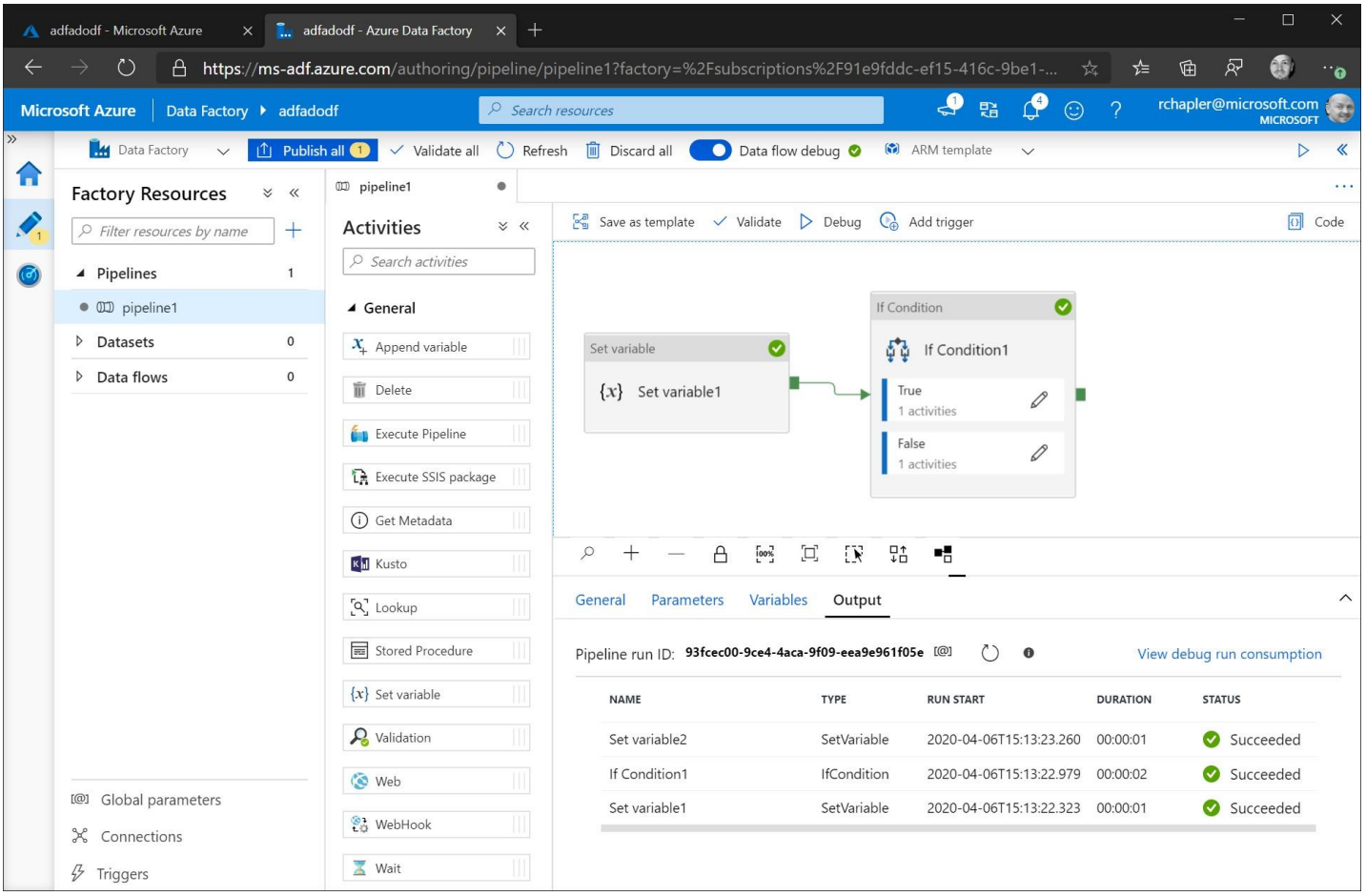


Debug

Click to activate “Data flow debug”. Allow time for processing.



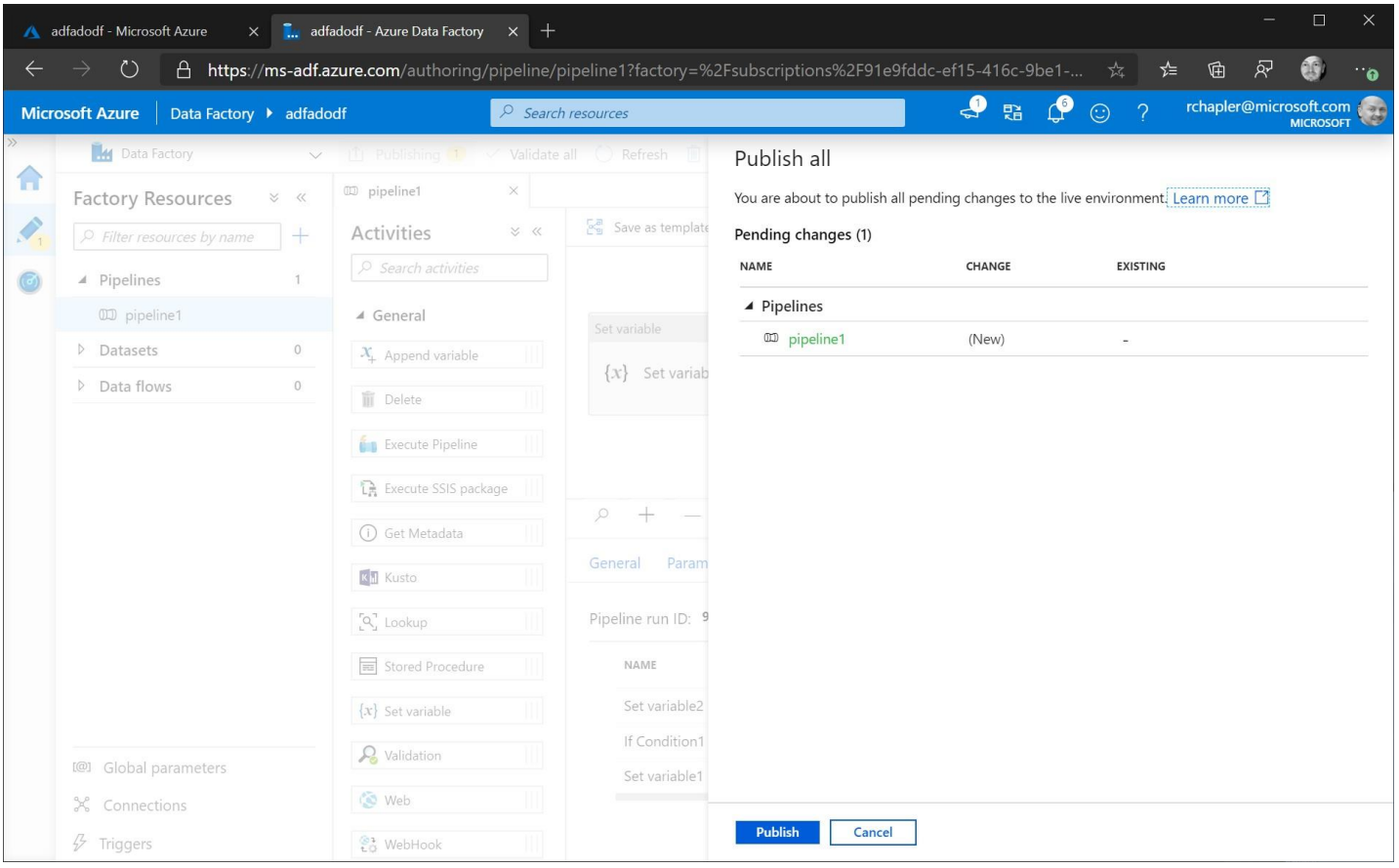
Once processing is complete, click the “Debug” button.



You should see “Succeeded” messages given current variable settings.

Publish

Click the “Publish all” button in the upper-leftish of the window.



Click the “Publish” button in the resulting popout. Allow time for processing.