

What I've Learned On Azure Data Factory

 medium.com/analytics-vidhya/a-simple-tutorial-on-azure-data-factory-e92b4a2c1b19

May 28, 2020

Bite-sized bits that will save you a day of learning



Azure functions can now be added as a step in Azure Data Factory (ADF). It makes it easier to synchronize data and to make trigger functions that can help data engineering. I have had to use this technology for an upcoming project and my time was misspent on Google searches, that's why I decided to share with you what I have learned after painstakingly trying to comprehend ADF in 2 days. The tutorial will cover **the specific** things that I needed to implement.

TL;DR You can **Ctrl + F** this:

Creating an Azure Data Factory (ADF)

First, go to create an Azure Data Factory resource on portal.azure.com

The screenshot shows the 'New data factory' form in the Microsoft Azure portal. The form is titled 'New data factory' and has a close button (X) in the top right corner. The form fields are as follows:

- Name ***: A text input field, highlighted with a red box.
- Version**: A dropdown menu with 'V2' selected.
- Subscription ***: A dropdown menu with 'Azure for Students' selected.
- Resource Group ***: A dropdown menu with a 'Create new' link below it, highlighted with a red box.
- Location ***: A dropdown menu with '(US) West US' selected.
- Enable GIT**: A checkbox that is checked, highlighted with a red box.
- GIT URL ***: A text input field.
- Repo name ***: A text input field.

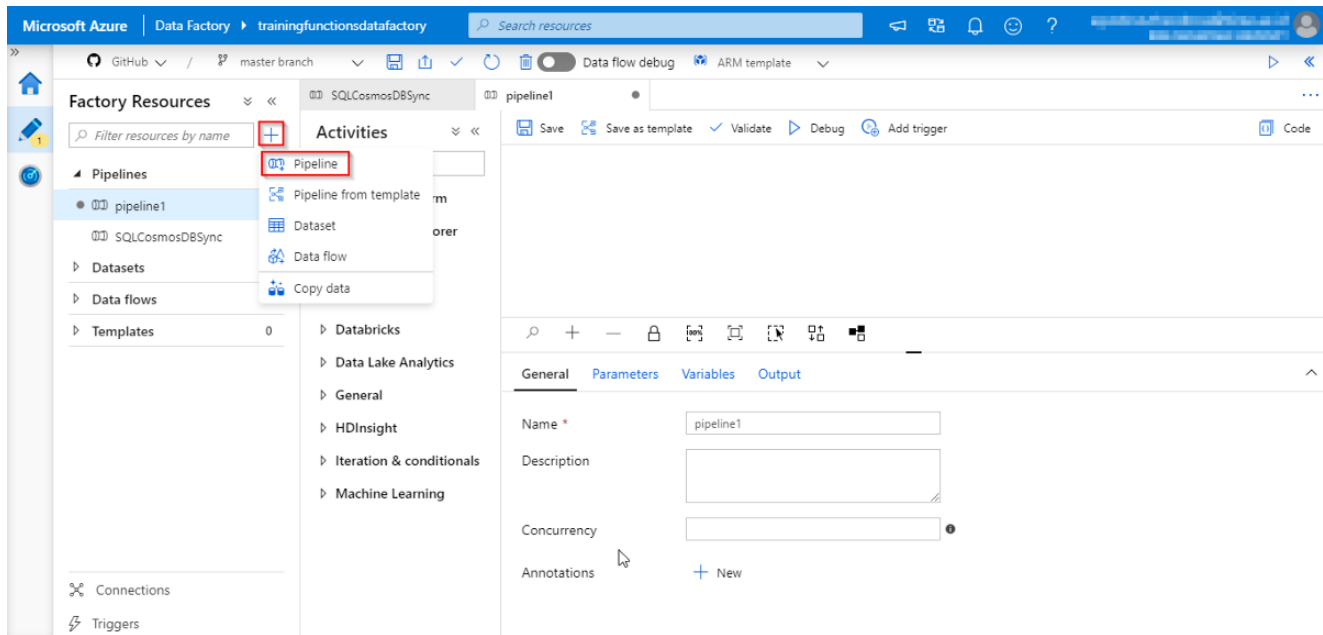
Fill in this New data factory form to generate one:

- Fill in the name of the data factory.
- Choose the resource group, if you haven't made one click the **Create new** button.
- You can connect this to a GIT repository but I prefer to configure it later, thus I unchecked the **Enable GIT** checkbox.

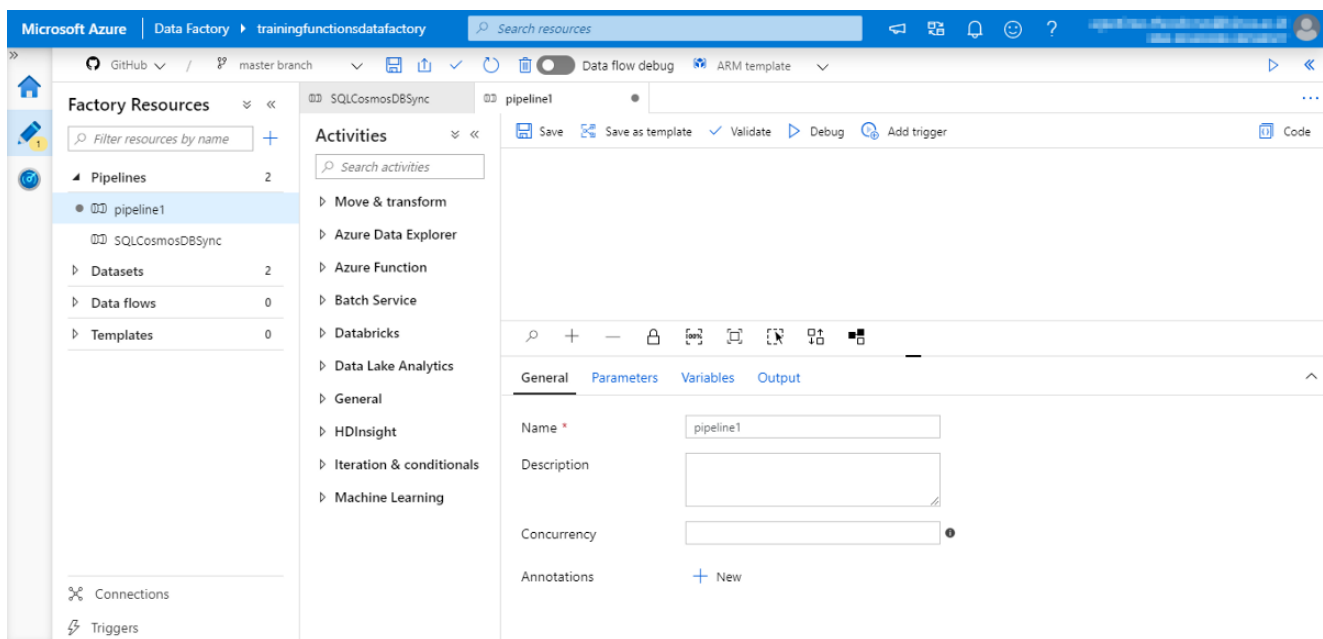
Pipelines in ADF

Adding a new Pipeline in ADF

Then open the ADF author page on adf.azure.com and add a new pipeline



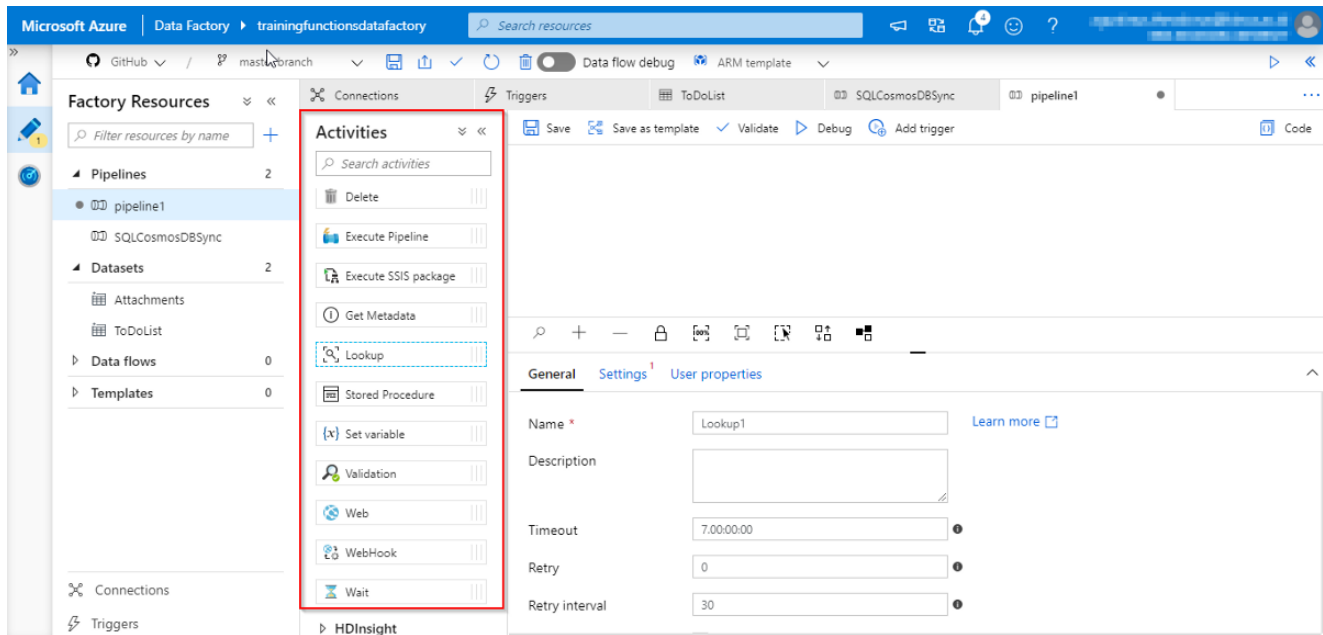
To add a new pipeline click the + sign and click Pipeline.



Voila! There is your pipeline.

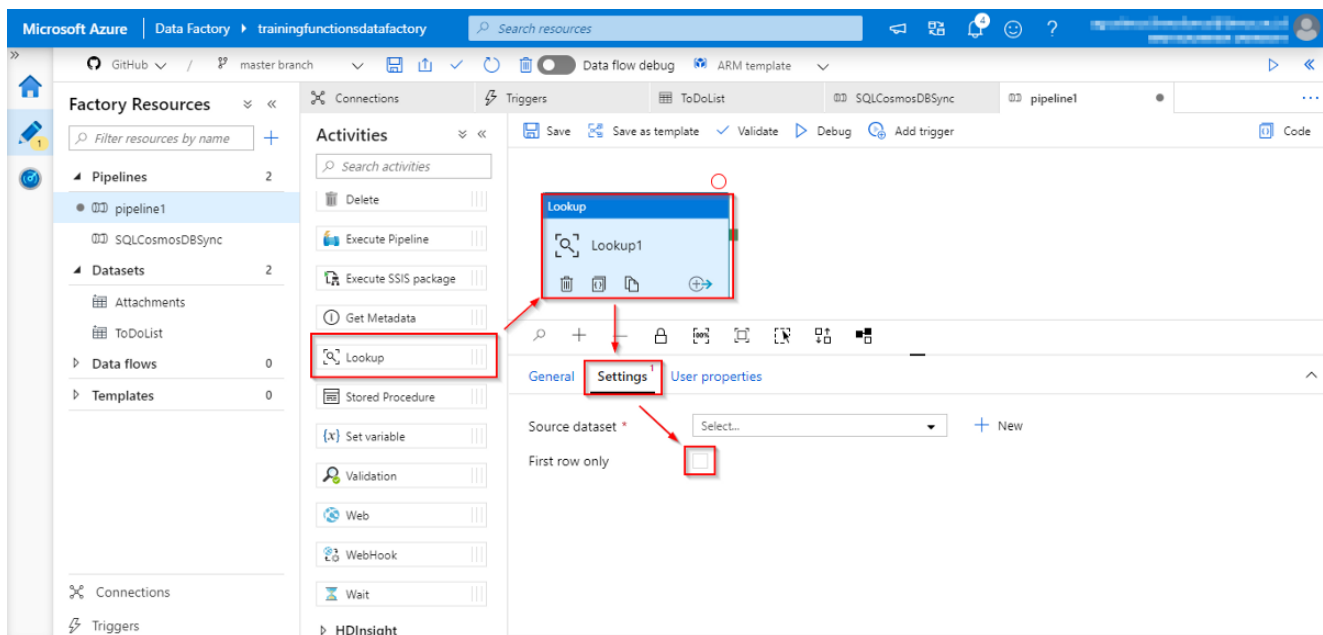
Adding a Lookup block in the Pipeline

There is an **Activities** tab on the left side of the screen, drag Lookup Activities to the center Canvas



At the bottom of the page, there are 3 tabs. Click on the **Settings** tab

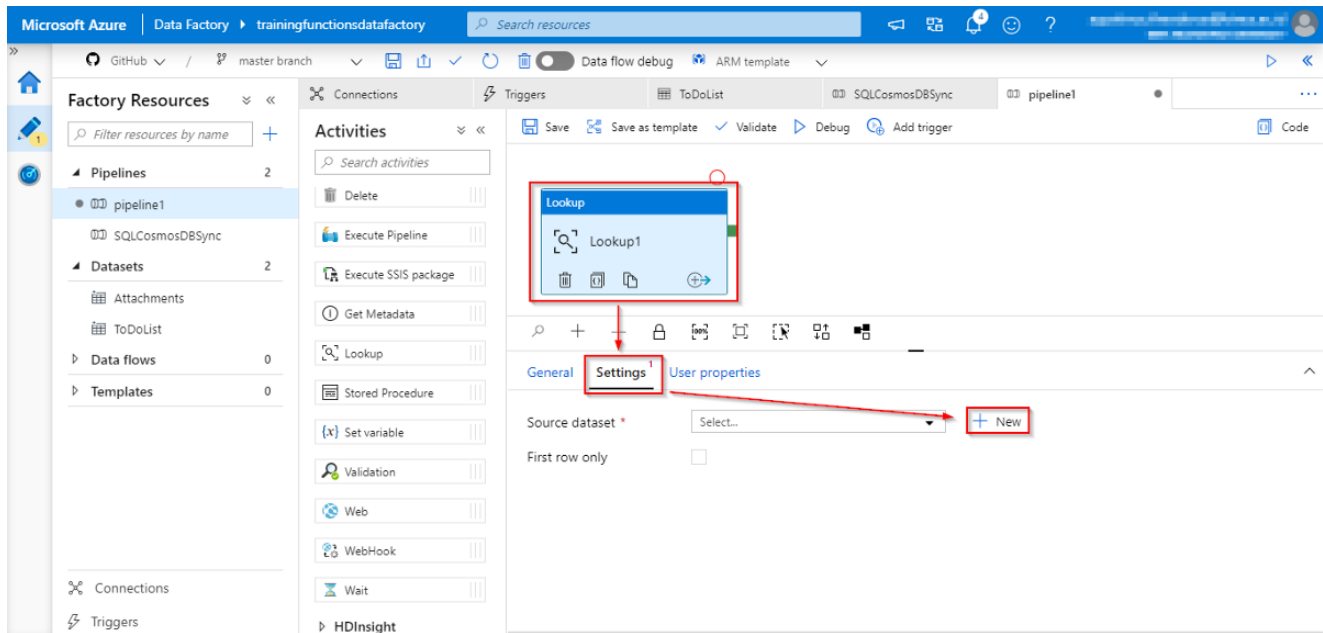
Changing First-row only response to All in Lookup



Uncheck the **First row only** checkbox. **The first row only** will enable us to lookup the first row of data, unchecking this will allow us to get all of the data available.

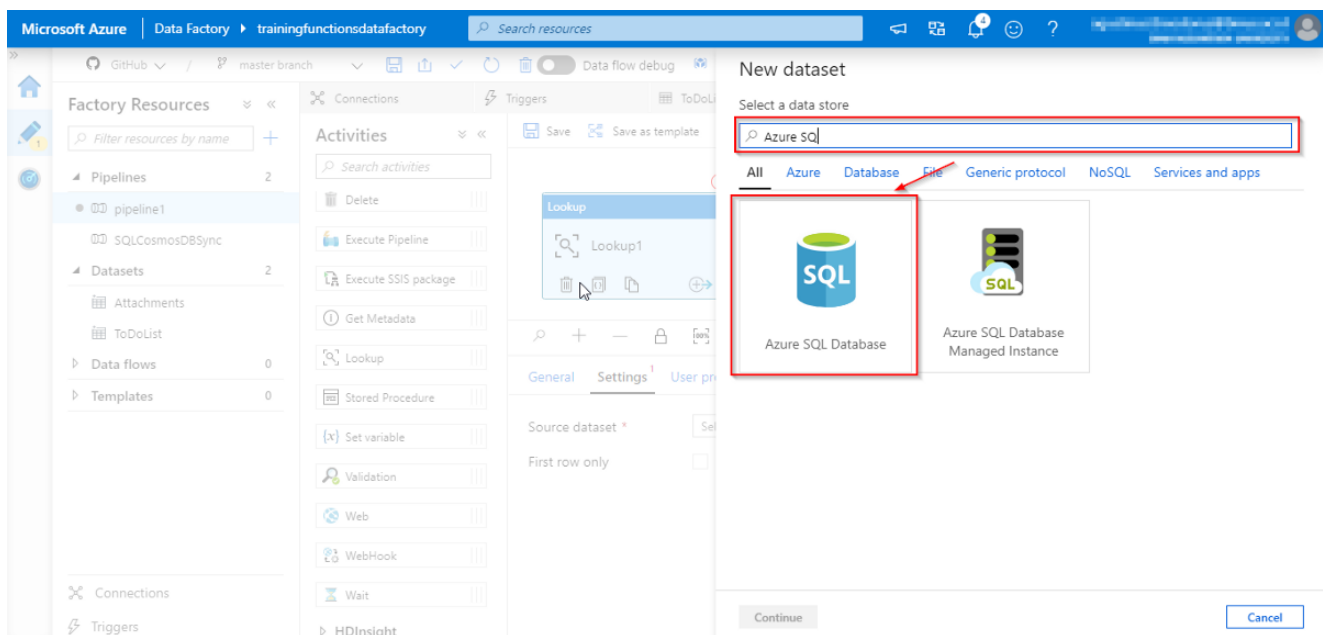
Adding a source dataset to a Lookup

The next thing you have to do is to add or select a dataset. Click the **New** button to create a new dataset source.

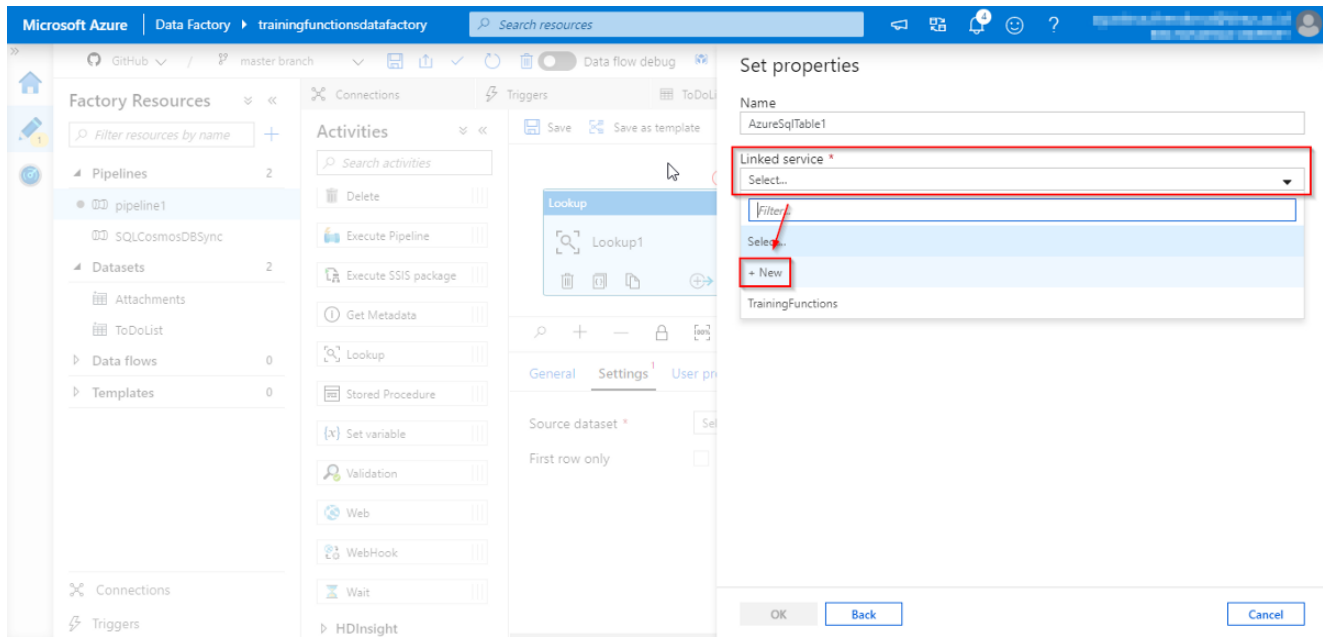


Assuming that you already have pre-existing knowledge of configuring an SQL Database on Azure.

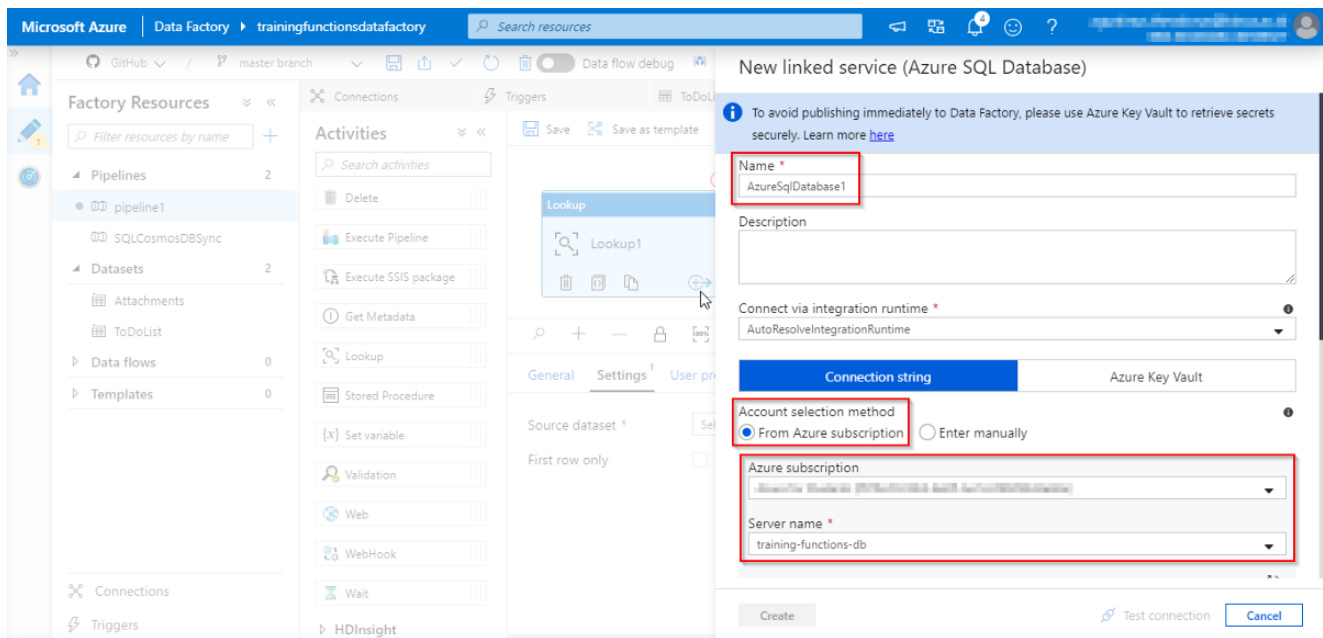
For this tutorial, we will be using an **Azure SQL Database** for simplicity's sake. Click **Continue**.



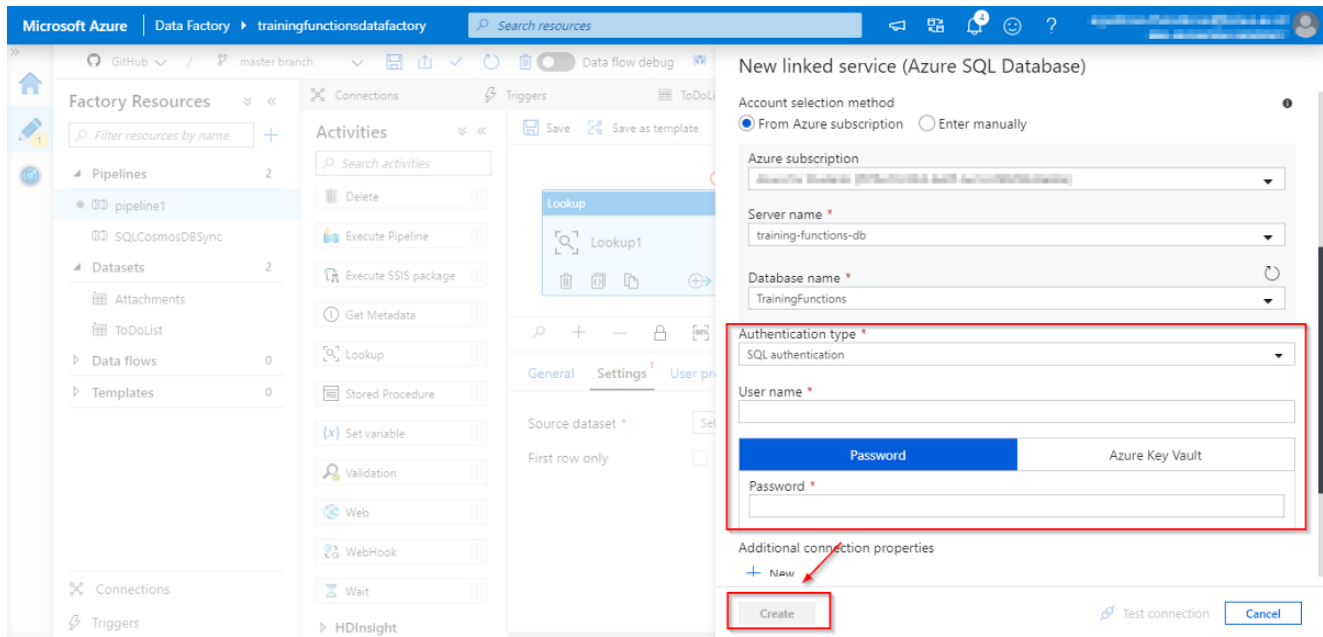
The pipeline requires a dataset to be processed. Even though in this tutorial we are connecting to an Azure SQL Database it is also possible to connect to different databases like Postgres, MySQL, MariaDB, etc.



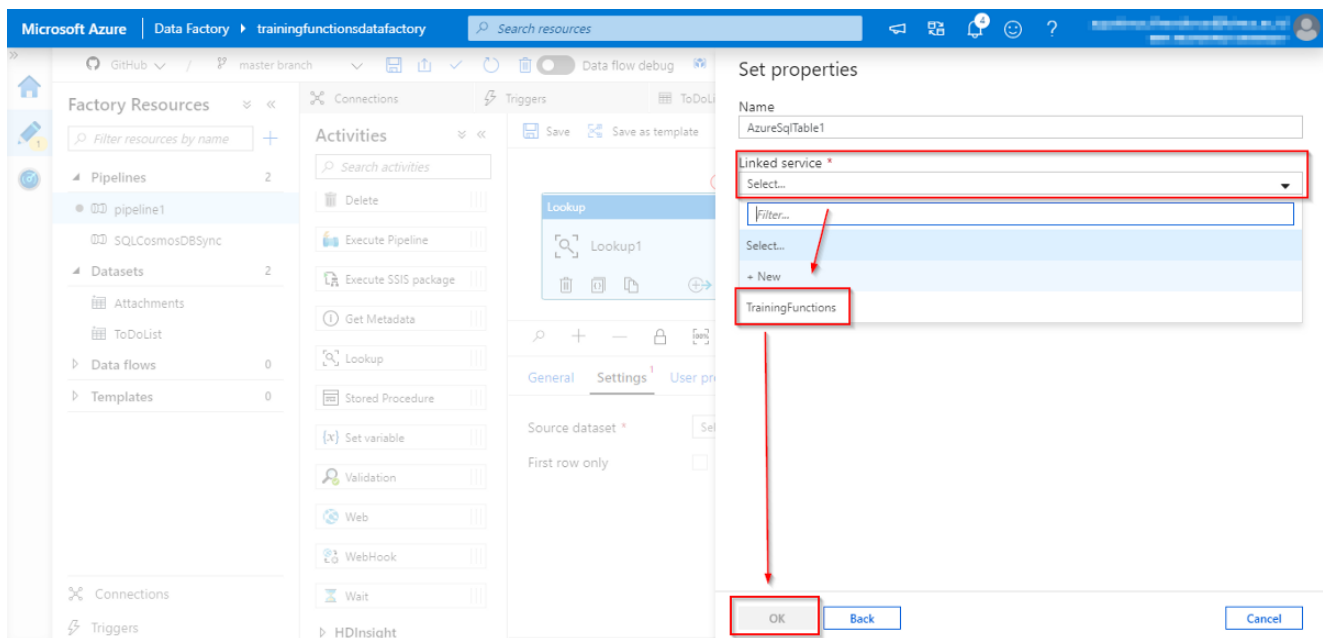
Then click **New** to add a new **Linked service**. **Linked service** is used to sync the Lookup action to a database table.



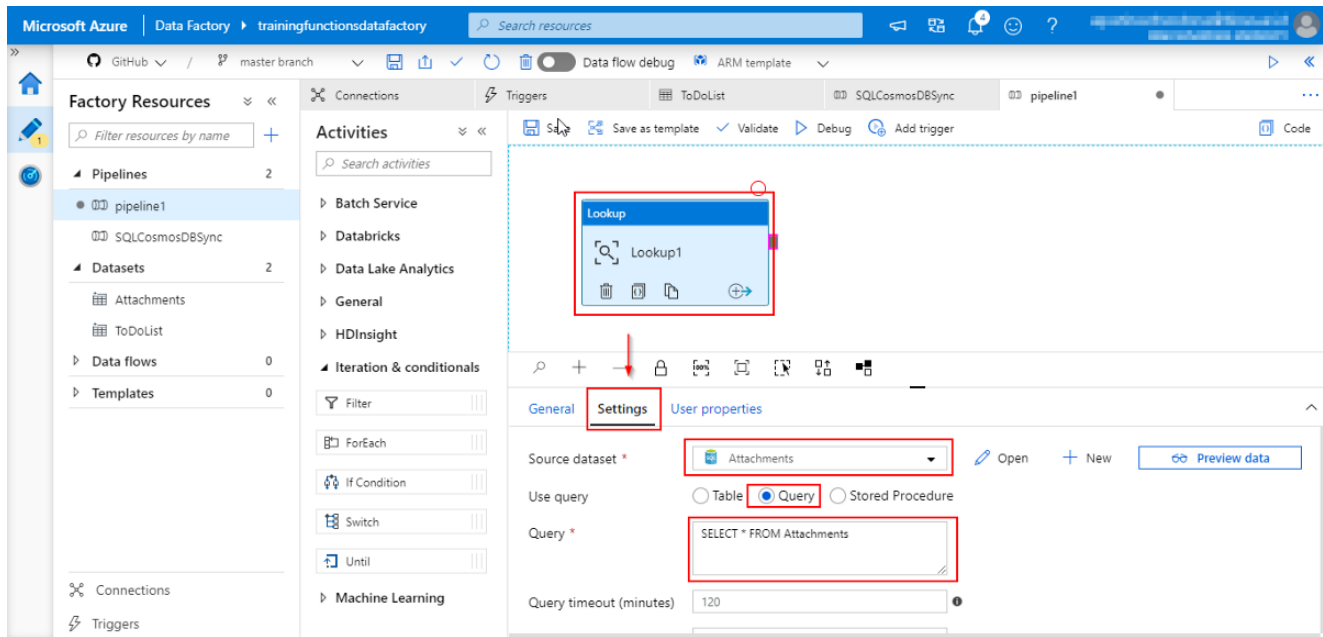
An Azure subscription is required. Choose the subscription you want to use and choose the server name. In my regard, I chose the Azure SQL Server I had and the **TrainingFunctions** database.



Assuming you have filled in the form, click **Create** to connect to a database of your choosing. With this done, all we have to do is to change the Source Dataset directly at the **Settings** tab.

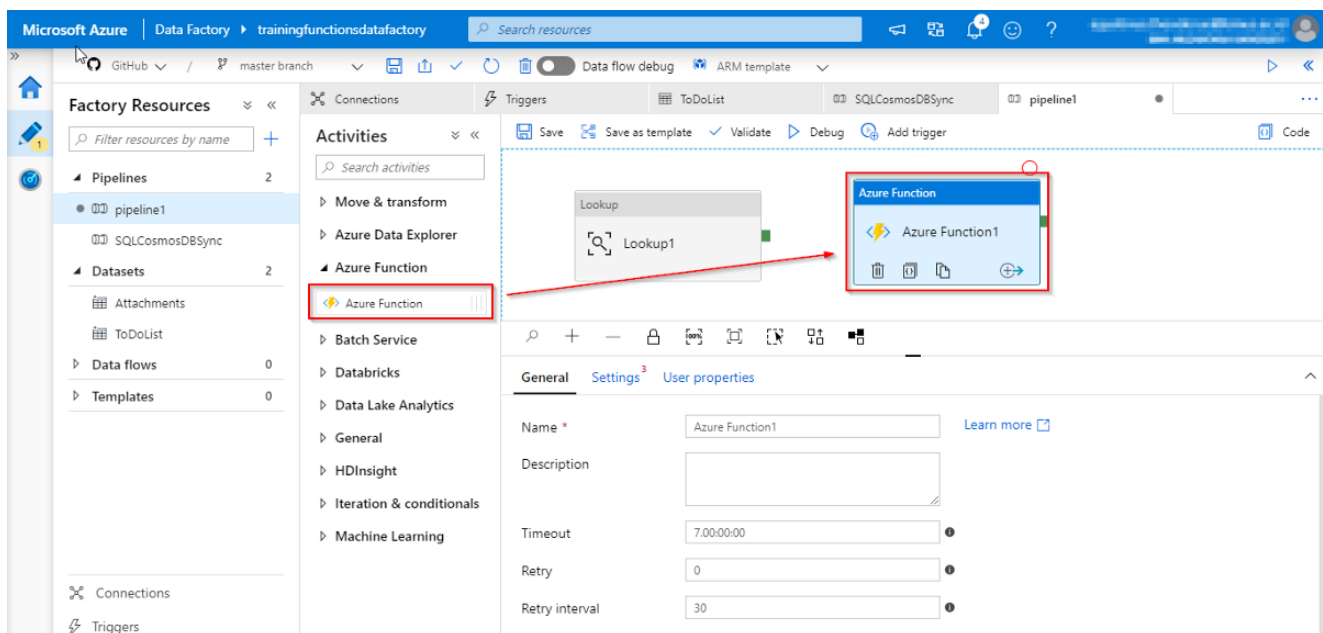


The last thing to do is to determine what kind of action you would like your lookup to do, in my case I want it to select from a table called Attachments.



Thus, the query would be like in the picture above. Otherwise, you can do other complex things such as joins and so on.

Adding an Azure Function block in the Pipeline

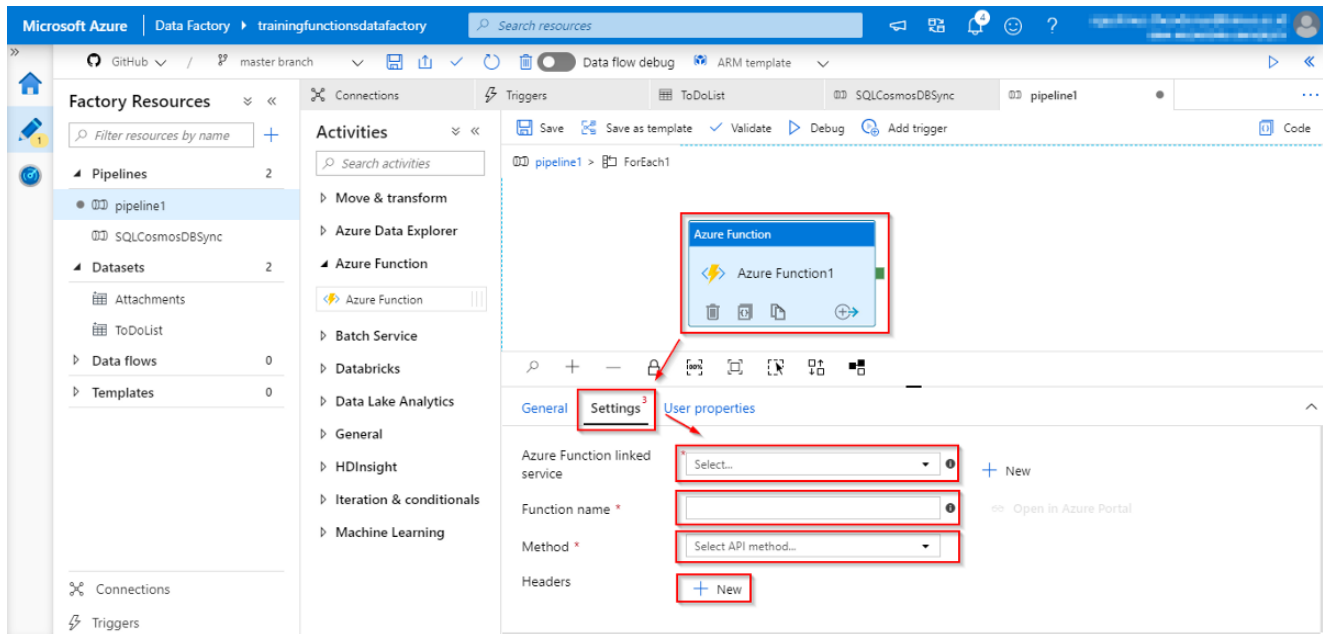


Adding the Azure Function block is just as easy, just drag and drop from the **Activities > Azure Function** tab to the main canvas.

Lookup to Azure Function Best Practice

It is not recommended to send the whole Lookup output to an Azure Function. It is recommended to add a ForEach block beforehand. This is what we are going to do in this tutorial.

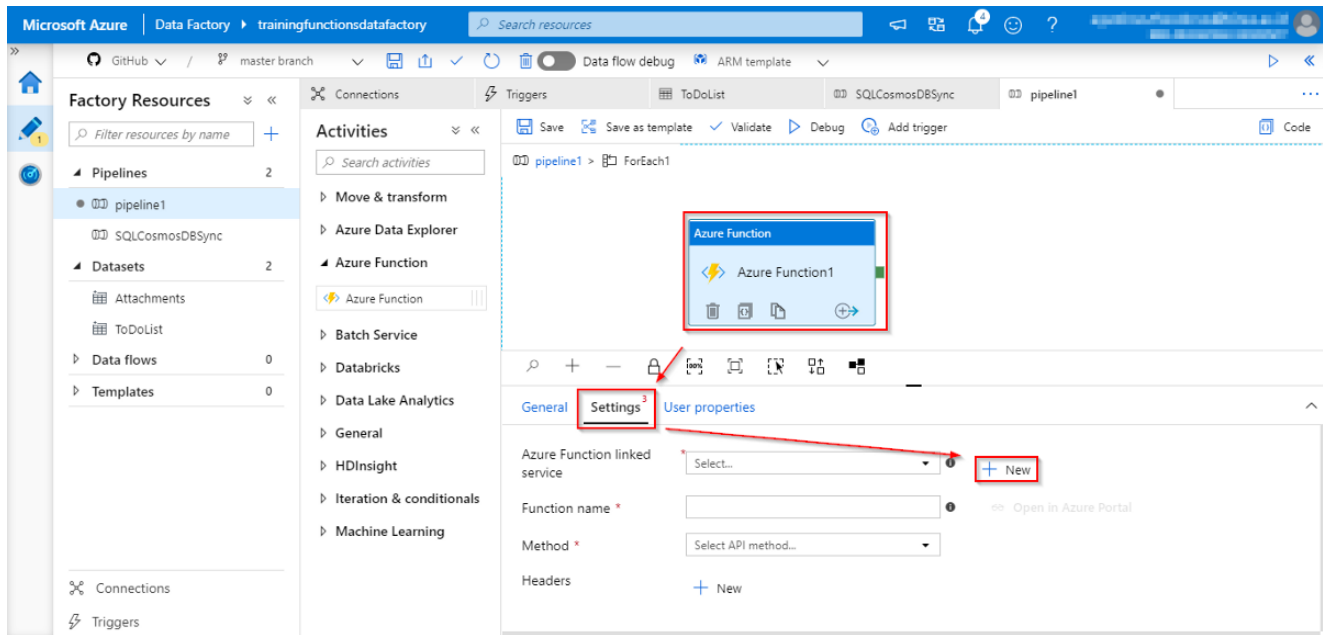
Connecting Azure Function in Pipeline



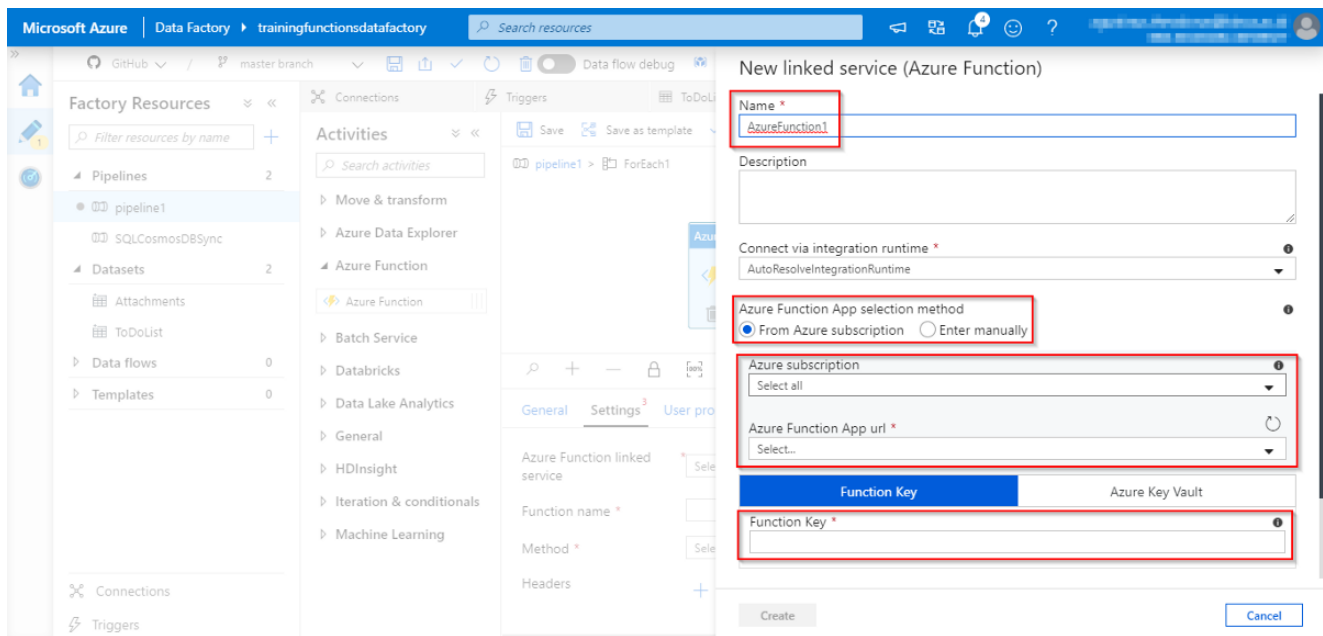
In the Azure Function go to the **Settings** tab.

- Select the linked service to link to the Azure Function. To add a new linked service, click **New** link.
- Fill in the **Azure Function name** in the Function name textbox, make sure the Function name is the same in the linked service.
- Fill in the method used by the selected Azure Function. Usually, HTTP Methods like **GET** and **POST** are used.

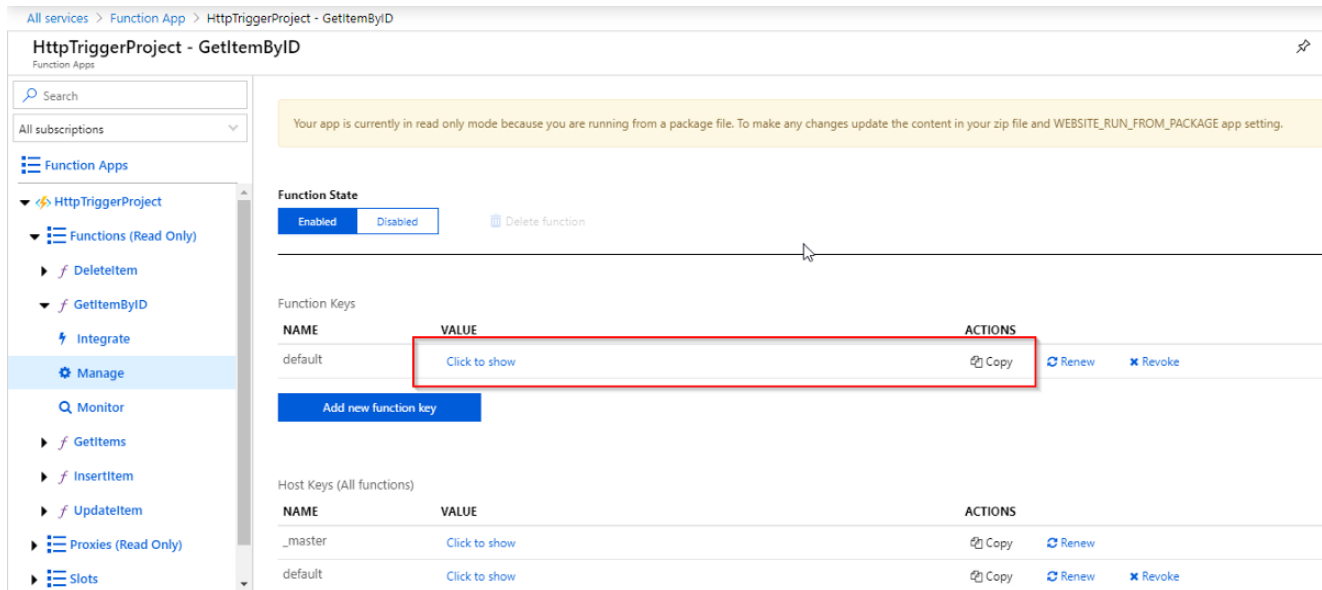
Otherwise, to make a new linked service click the + **New** button



Connecting to an Azure Function through a linked service is similar to connecting to a dataset.

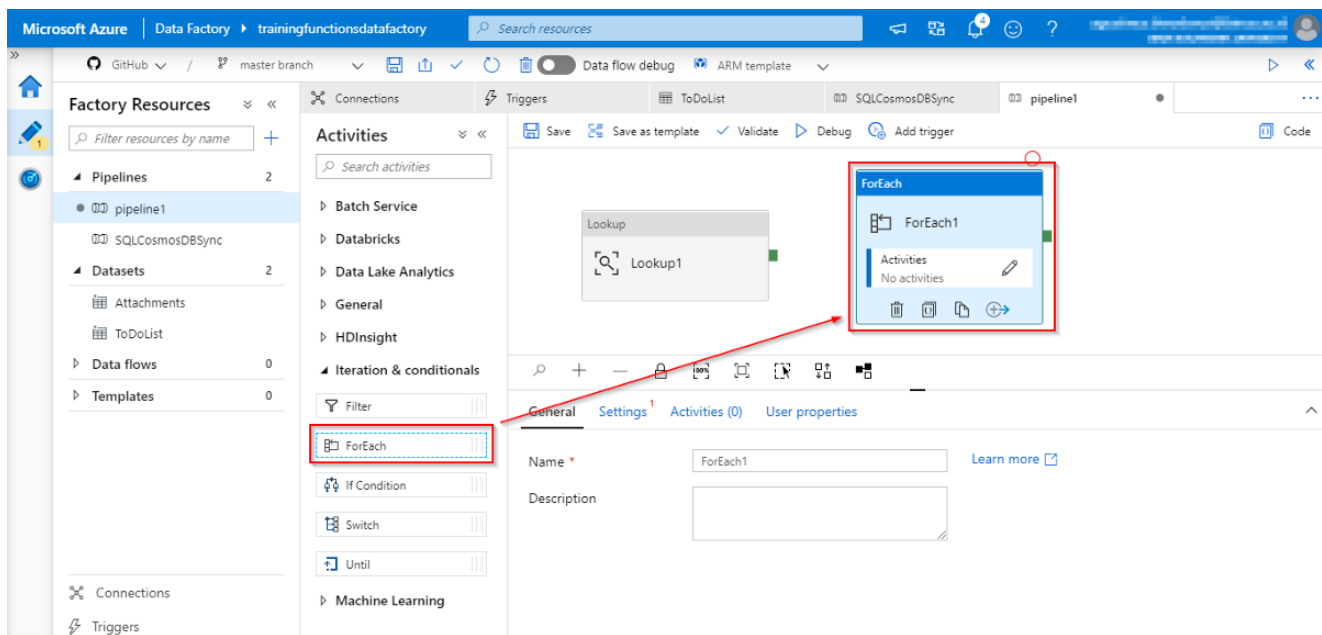


Fill in the linked service form to connect to your Azure Function, select a subscription and then pick the selected function. The next thing is to get your Function key.



You can get your Function Key in **Home > All Services > Function App > Function Project > Functions > {Function Name} > Manage** at portal.azure.com.

Adding a ForEach block in the Pipeline

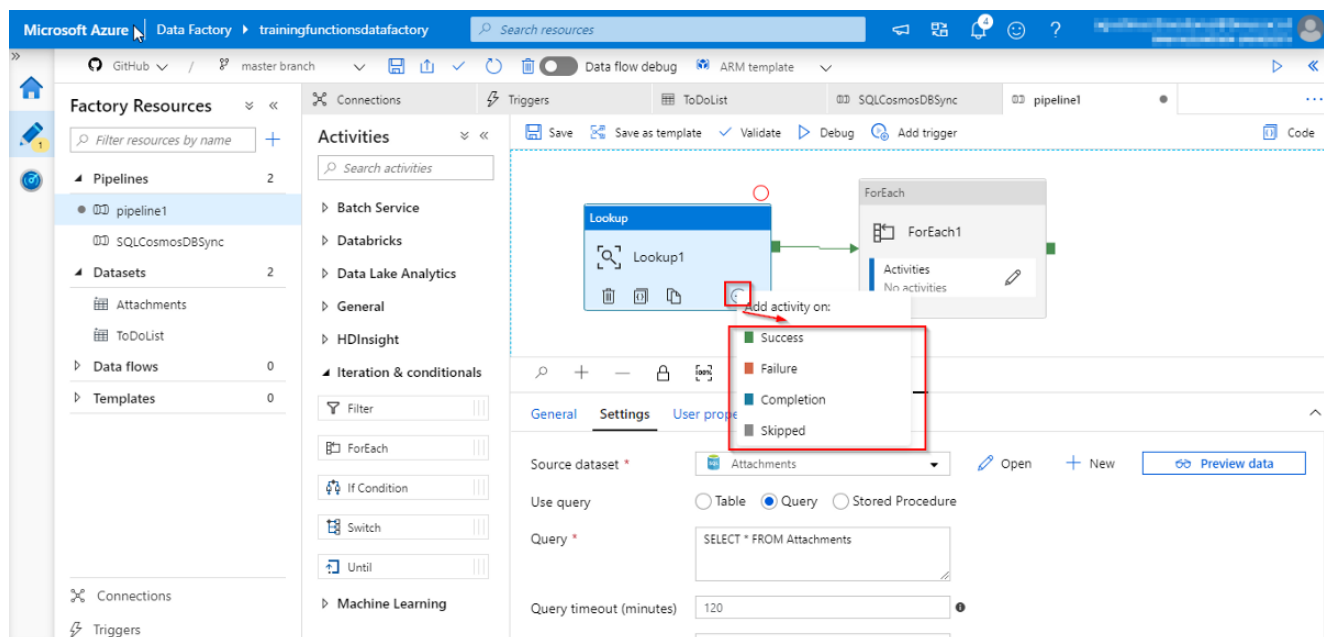


Adding the ForEach block is as simple as just dragging and dropping from the **Activities > Iterations & conditionals** tab to the main canvas.

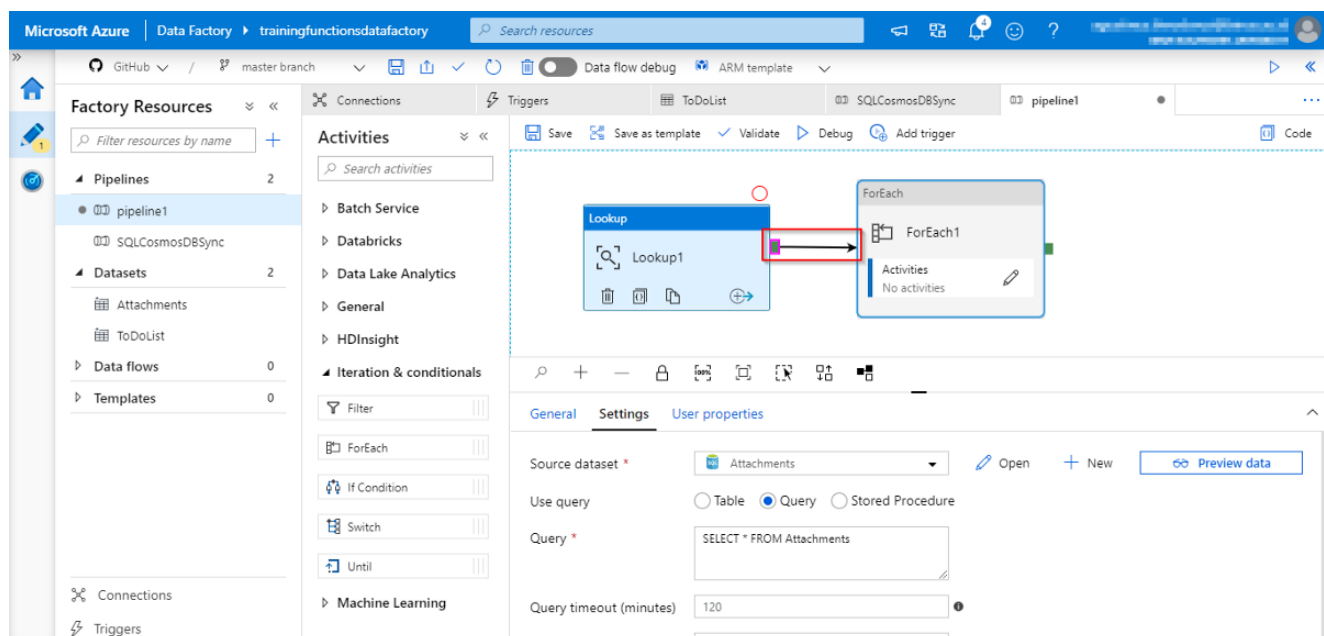
Connecting Pipeline Blocks

Connect a Lookup with ForEach

In this premise, we need to connect the Lookup block to the ForEach block. Sending the Lookup output to be iterated and processed in the ForEach block.



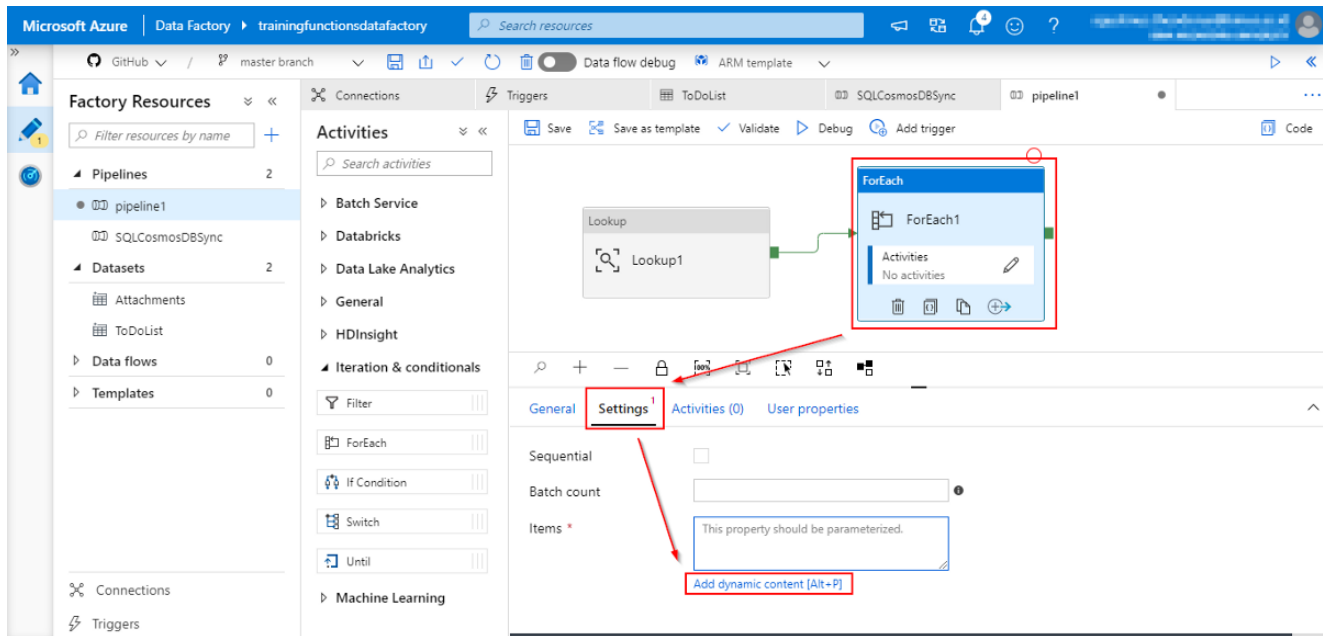
First, we need to connect the two blocks by activity, there are 4 types of activities: Success, Failure, Completion and Skipped. You can see this when you click on the + symbol on the **bottom-right side** of the Lookup box.



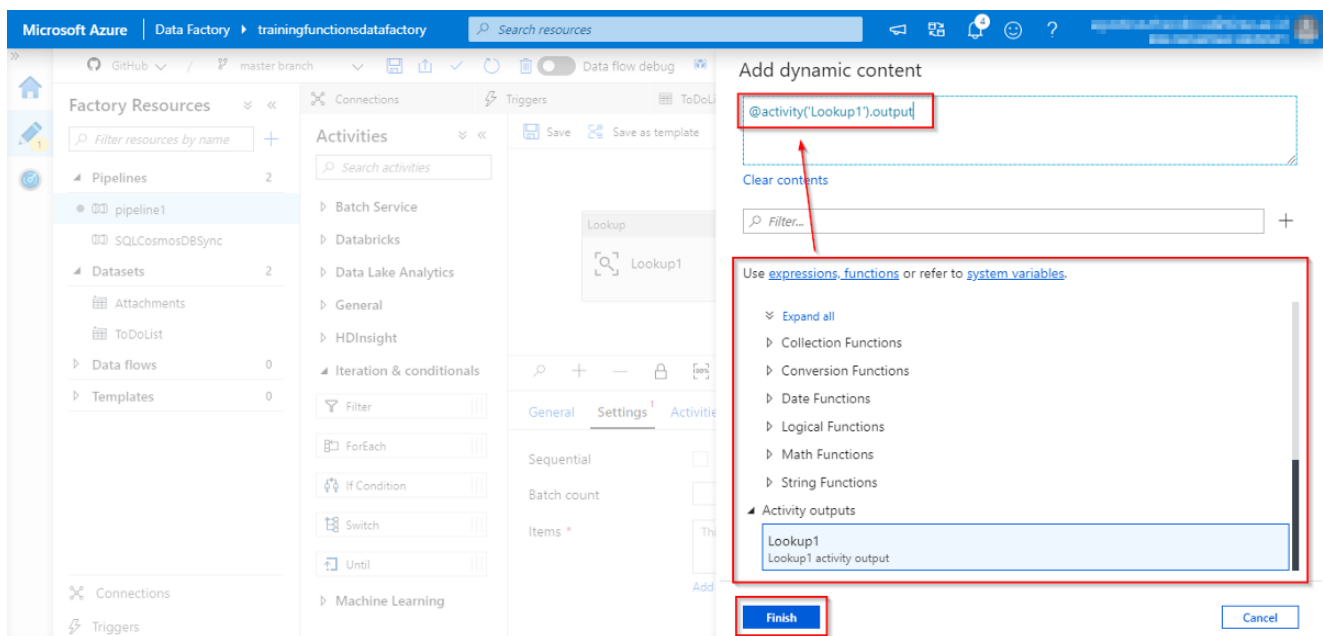
In this example, we will be using Success activity. After connecting, click on the **ForEach** block, go to the **Settings** tab.

Using Dynamic Contents to add Object inside Object in ADF

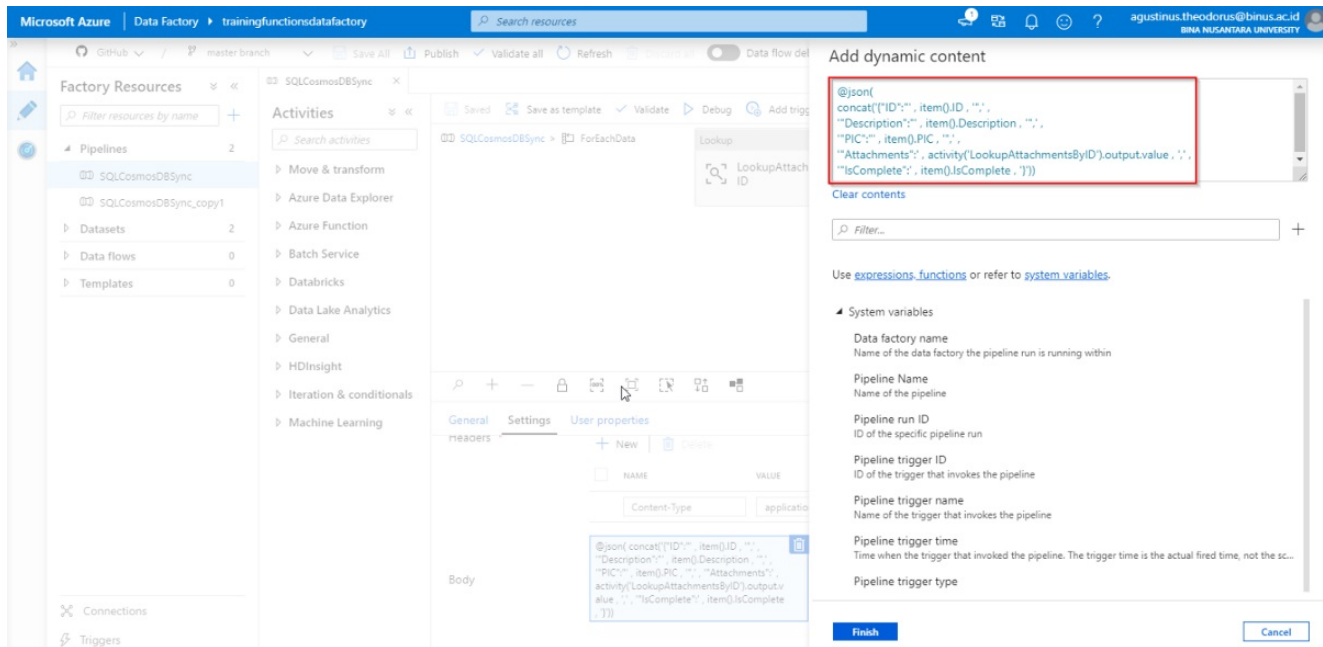
Dynamic contents are used to customize the input.



Click on the Items textbox and click the **Add dynamic content** button.



In the **Add dynamic content** sub-page scroll to the bottom and click on the Lookup output expression, this will add the Lookup output to the ForEach input to be processed.



This example is how you could join two lookup items together as one input using string Concat function, then converting them again to JSON to be then passed to the Azure Function. In the example above, the input will be passing 5 attributes **ID**, **Description**, **PIC**, **Attachments**, and **IsComplete**. With the **Attachment** attribute being a list of JSON objects from the previous lookup.

Final Thoughts

Azure Data Factory is a good tool for data engineering and it is very beneficial in the long term, when your database grows so large until it is almost unmanageable ADF may be a choice for you. With the addition of Azure Functions, the potentials are endless. Even connecting to databases that are not Microsoft related is possible! I hope you found this tutorial useful.

Discover Medium

Welcome to a place where words matter. On Medium, smart voices and original ideas take center stage - with no ads in sight. [Watch](#)

Make Medium yours

Follow all the topics you care about, and we'll deliver the best stories for you to your homepage and inbox. [Explore](#)

Become a member

Get unlimited access to the best stories on Medium — and support writers while you're at

it. Just \$5/month. Upgrade