

How to Delete all files from Folder - Delete Activity in Azure Data Factory ADF Tutorial 2021

Issue: How to Delete all files from Folder - Delete Activity in Azure Data Factory ADF Tutorial 2021

In this article, we are going to learn how to delete all files from a folder, we are going to perform a demonstration in which we will delete some files from different containers, let's start our demo, open the Azure Data Factory Portal, and create a new blob storage, click on Blob Storage and then click +New button, it will open a window in which we have to fill the required fields, which are 1. Subscription, 2. Resource group, 3. Storage account name, 4. region, and then click on Review + create and then click on create as shown in the picture below.

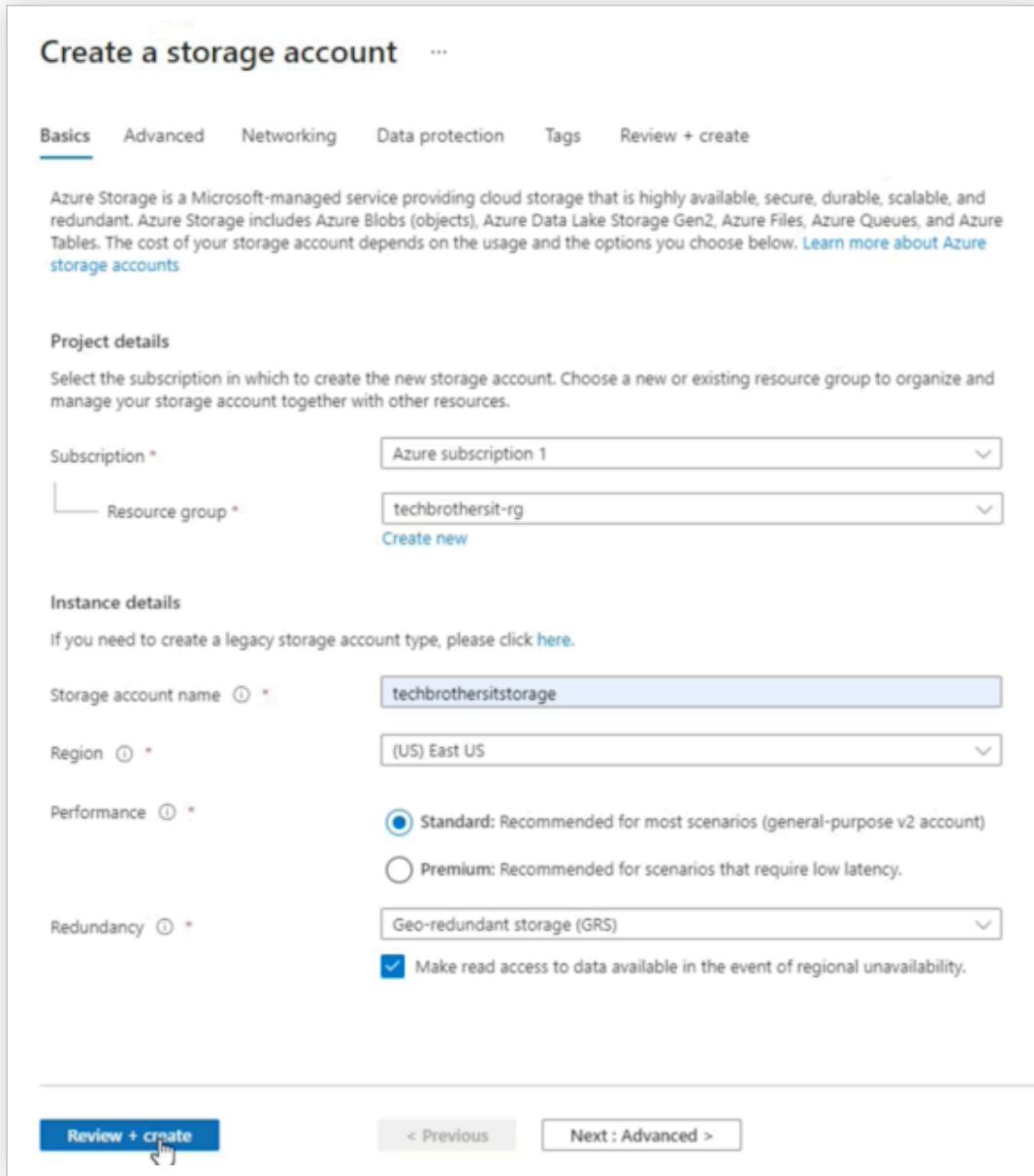


Fig-1: Create a new azure blob storage

Once our blob storage is created we are going to create some containers their and then we will upload some files, let's go to the resource which we have created and click on the containers at the left side of the dashboard under the Data storage tab, click on + New container and give the name and then click on create, in my case I have given the name "input" as my first container then again click on +New and create another container as we created earlier, you can see in the picture below that our containers are created.

The screenshot shows the Azure Storage Explorer interface. On the left, there's a navigation pane with sections like Home, Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage Explorer (preview), Data storage (Containers, File shares, Queues, Tables), Security + networking (Networking, Azure CDN, Access keys, Shared access signature, Encryption, Security), Data management (Geo-replication, Data protection, Object replication), and Advanced filtering. The 'Containers' section is selected. In the main area, there's a search bar for 'Search containers by prefix'. Below it is a table with columns: Name, Last modified, Public access level, and Lease state. Three containers are listed: 'logs' (modified 7/20/2021, 12:10:54 PM, Private, Available), 'input' (modified 7/20/2021, 12:11:29 PM, Private, Available), and 'output' (modified 7/20/2021, 12:11:38 PM, Private, Available). A red box highlights the 'input' container.

Fig-2: Create data containers for performing delete activity

Once our containers are created, we have to upload files in that containers, as you can see in the picture below that we have uploaded some files which are in different formats ie. .txt, .csv, .xlsx etc.

The screenshot shows the Azure Storage Explorer interface, specifically the 'input' container page. The left sidebar includes sections for Overview, Diagnose and solve problems, Access Control (IAM), Settings, Shared access tokens, Access policy, Properties, and Metadata. The main content area displays a table of blobs. The columns are: Name, Modified, Access tier, Blob type, Size, and Lease state. Several files are listed: 'EmployeeAddress_20210712.xlsx' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 75.9 KB, Available), 'Employee_20210712.csv' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 30.8 KB, Available), 'EmployeeTest_20210712.txt' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 38.8 KB, Available), 'mytestcsvfile.csv' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 0.8 KB, Available), 'New Microsoft Excel Worksheet.xlsx' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 6.05 KB, Available), and 'New Rich Text Document.rtf' (Modified 7/20/2021, 12:12:02 PM, Hot (Inferred), Block blob, 7.8 KB, Available). A red box highlights the list of files.

Fig-3: uploaded files in the input container.

As our files are uploaded, now we have to perform the delete activity, so let's go to the Azure data factory and click on the author, and here click on Pipeline, then click on + sign and then click on the pipeline, it will open a window in that window we have given the name of our pipeline, then search for the delete activity and drag it into the working window. double click on delete activity and define the source, so go to the dataset in the source tab and click on the + New button, it will open a window and in that window search for the Azure blob storage and select the Azure blob storage, then click on continue in the new window give the name of your blob storage and then create the linked service, once you click on new linked service it will open a window which is shown in the picture below, give the name of linked service and then click on create. and select the path folder in my case it is the Input folder and then click ok.

New linked service (Azure Blob Storage)

Name *
blobstoragelink

Description

Connect via integration runtime * ⓘ
AutoResolveIntegrationRuntime

Authentication method
Account key

Connection string Azure Key Vault

Account selection method ⓘ
 From Azure subscription Enter manually

Azure subscription ⓘ
Azure subscription 1 (959da5f8-7cdc-4564-8b82-df7a37646d2b)

Storage account name *
techbrothersitstorage

Additional connection properties
+ New

Test connection ⓘ
 To linked service To file path

Annotations
+ New

Parameters

Advanced ⓘ

Connection successful

Create ↗ Test connection Cancel

Fig-4: Create a new linked service.

Once our dataset is created now I have to delete only .txt files, click on the wild card, and in wild card name write *.txt that means it will select only .txt files now click on the logging settings and select the logging account linked service, in my case, it is blobstoragelink, and select the folder path for the logs of the deleted files, then back to the dashboard and click on debug as you can see in the picture below there are 3 .txt files has been deleted.

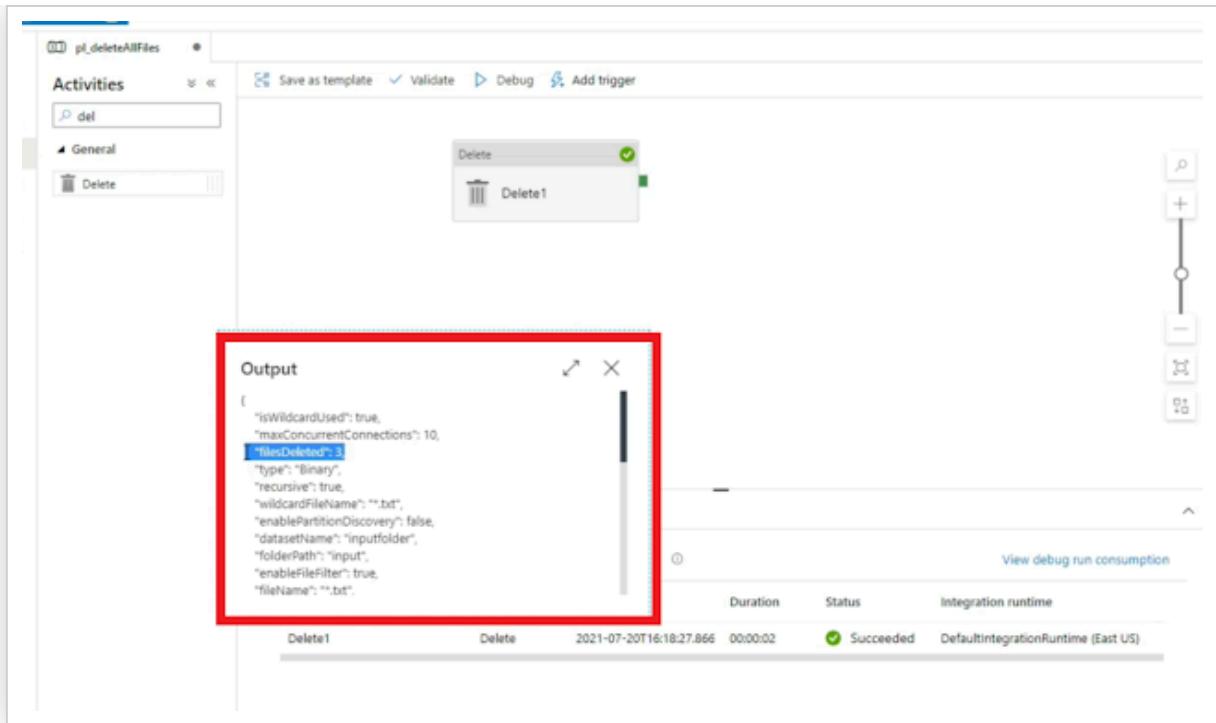


Fig-5: Shown summary of deleted files which were selected by *.txt

As we perform the delete activity and deleted selected files now let's delete all files in the folder, click on the delete activity and go to the source tab now in the source tab if you remember we write *.txt for deleting .txt files, let's remove the *.txt and write *.* and click on debug, now as you can see in the picture below we have deleted all the 6 files.

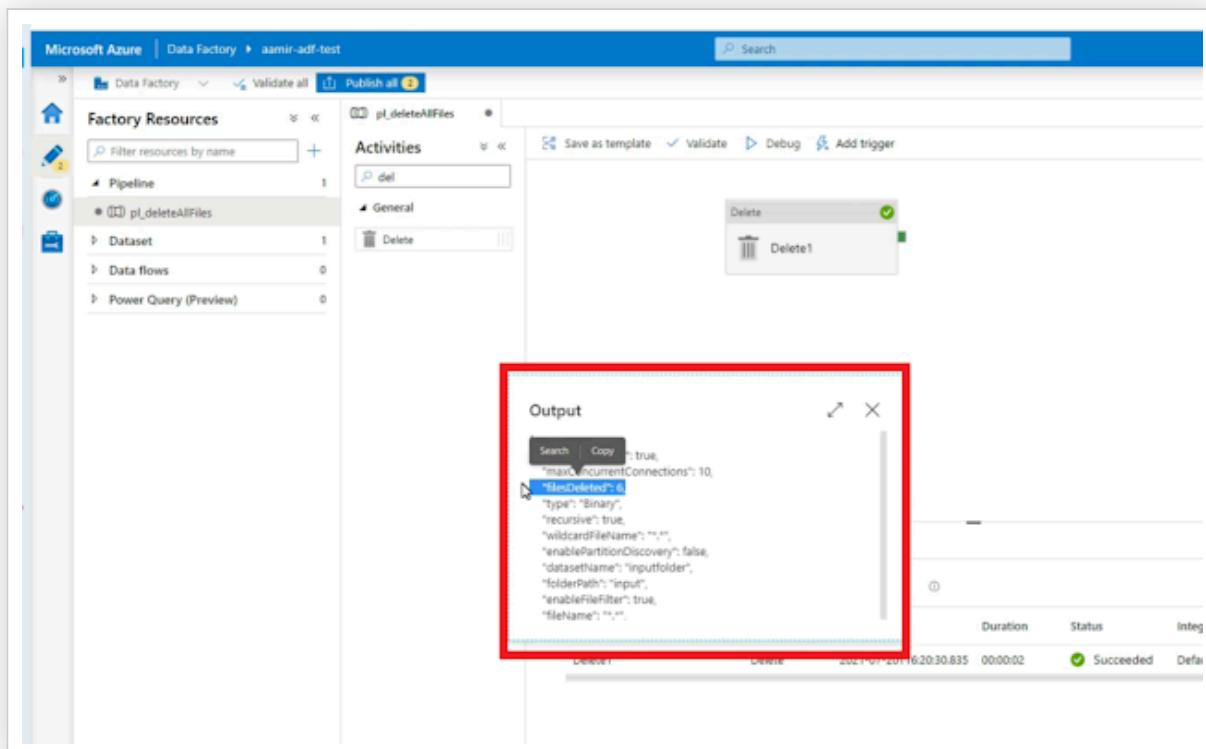


Fig-6: Shown summary of deleted files which were selected by *