

# Create Data Factory

1. Go to Home and click on Data Factories under Azure Services.

The screenshot shows the Microsoft Azure home page. At the top, there's a search bar and several icons for different services like Storage accounts, Data factories, App registrations, Quickstart Center, Virtual machines, App Services, SQL databases, and Azure Cosmos DB. Below the search bar, there's a section titled "Azure services" with a "Create a resource" button and icons for Storage accounts, Data factories, App registrations, Quickstart Center, Virtual machines, App Services, SQL databases, and Azure Cosmos DB. A "More services" link is also present.

2. Next click on Create Data Factory as shown below.

The screenshot shows the "Data factories" blade in Microsoft Azure. It includes a header with "Home > Data factories" and a user profile. Below the header are filter options: "+ Create", "Manage view", "Refresh", "Export to CSV", "Open query", and "Assign tags". There are also buttons for "Filter for any field...", "Subscription equals all", "Type equals all", "Resource group equals all", "Location equals all", and "Add filter". At the bottom, there are grouping and view mode buttons ("No grouping" and "List view"). The main area displays a message: "No data factories to display. Try changing or clearing your filters." with a "Create data factory" button and a "Learn more" link.

3. In the Resource Group give the group name that we created in the previous example.
4. Give the Name and click on the Review+Create option.

## Create Data Factory ...

The screenshot shows the "Create Data Factory" wizard in the "Basics" step. It has tabs for Basics, Git configuration, Networking, Advanced, Tags, and Review + create. Under "Project details", it says "One-click to create data factory with sample pipeline and datasets. Try it". It asks to select a subscription and resource group. The subscription is set to "Microsoft Partner Network" and the resource group is "ADFResource". Under "Instance details", it asks for a name, region, and version. The name is "AzureDataFactoryTraing", the region is "East US", and the version is "V2". At the bottom, there are "Previous", "Next", and "Review + create" buttons.

5. Click on Create.

Basics    Git configuration    Networking    Advanced    Tags    **Review + create**

[View automation template](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offer with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for a detailed description.

**Basics**

Subscription	Microsoft Partner Network
Resource group	ADFResource
Name	AzureDataFactoryTraing
Region	East US
Version	V2

**Networking**

[Previous](#)

[Next](#)

**Create**

6. After some time you will see the below screenshot. It means our data factory is created.

The screenshot shows the Microsoft Data Factory Overview page. At the top, it says "Your deployment is complete". Below that, it provides deployment details: Deployment name: Microsoft.DataFactory-20230720111825, Subscription: Microsoft Partner Network, Resource group: ADFResource. To the right, there is a "Cost management" section with a green icon and a link to "Set up cost alerts >".

7. Now go to the data factory, go to home and click on the data factories.

Azure services



8. You will see the data factory that we created click on it.

**Data factories**

Gisul (gisul.co.in)

[+ Create](#) [Manage view](#) [Refresh](#) [Export to CSV](#) [Open query](#) [Assign tags](#)

[Filter for any field...](#) [Subscription equals all](#) [Type equals all](#) [Resource group equals all](#) [Location equals all](#) [Add filter](#)

Showing 1 to 1 of 1 records.

Name ↑↓

Type ↑↓

Subscription ↑↓

AzureDataFactoryTraing

Data factory (V2)

Microsoft Partner Network

9. Now click on Launch Studio as shown below.

The screenshot shows the Microsoft Azure Data Factory Studio interface. On the left, there's a sidebar with navigation links like 'Create', 'Manage view', 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Settings', 'Networking', 'Managed identities', 'Properties', 'Locks', 'Getting started', 'Quick start', 'Monitoring', 'Alerts', 'Metrics', 'Diagnostic settings', and 'Logs'. The main area displays the 'AzureDataFactoryTraing' data factory with its resource group (move), status (Succeeded), location (East US), and subscription (Microsoft Partner Network). A large blue button labeled 'Launch studio' is prominently displayed at the bottom of the main content area.

10. It will directly take you to Azure Data Factory as shown below.

The screenshot shows the Azure Data Factory studio interface for the 'AzureDataFactoryTraing' data factory. The top navigation bar includes 'Set up code repository'. The main content area features four large cards: 'Ingest' (Copy data at scale once or on a schedule), 'Orchestrate' (Code-free data pipelines), 'Transform data' (Transform your data using data flows), and 'Configure SSIS' (Manage & run your SSIS packages in the cloud). Below these cards is a section titled 'Recent resources' which currently says 'No items to show'.

## Create Pipeline

1. In this example we use the Std\_details file that we upload into the cloud blog storage,

The screenshot shows the Azure Storage Blob container 'f-demo'. The left sidebar includes 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Settings', 'Shared access tokens', 'Access policy', and 'Properties'. The main area shows a table of blobs with one entry: 'Std\_details.txt' (Modified: 7/20/2023, 11:41:27...), 'Access tier: Hot (Inferred)', 'Archive status: Not yet archived', 'Blob type: Block blob', 'Size: 166 B', and 'Lease state: Available'.

2. In the New search tab. Type adf.azure.com
3. Select the Data Factory that we created under the Name section then click on Continue.

Create a new data factory  Select an existing data factory

### Select an existing data factory

Microsoft Azure Data Factory is a cloud-based data integration service that automates data movement and transformation. [Learn more](#)

Azure Active Directory [\(1\)](#)  
Gisul (f91ae283-2d17-4062-aef-1884d70ce0be)

Subscription  
Microsoft Partner Network (df295c6e-136c-4598-9168-19fa9dba7fe1)

Name \*  
AzureDataFactoryTraining

**Continue**

4. Go to Manage table and click on Linket service then click on Create Linked Service.

Microsoft Azure | Data Factory > AzureDataFactoryTraining

Search factory and documentation

Validate all Publish all [\(1\)](#)

Preview experience  OFF

Lochi.Balaji@gisul.co.in

General

Connections

Linked services **New**

Filter by name Annotations: Any

Linked services

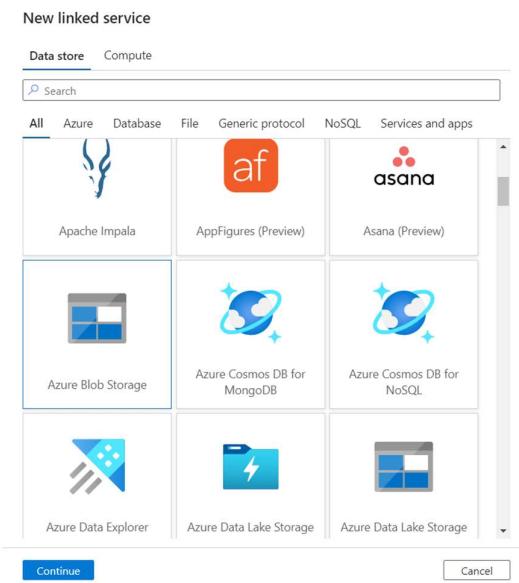
Linked service defines the connection information to a data store or compute. [Learn more](#)

No linked service to show

If you expected to see results, try changing your filters or create a new linked services.

Create linked service

5. Select Azure Blob Storage and click on continue.

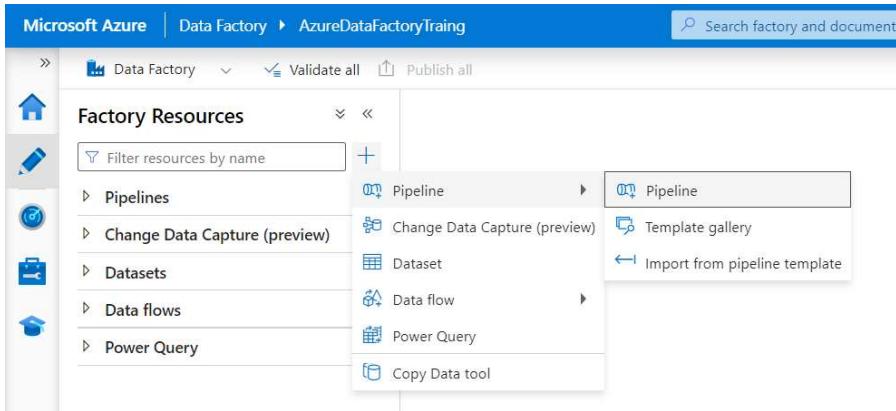


6. Next give the name and storage account name. Then set the properties as shown below.

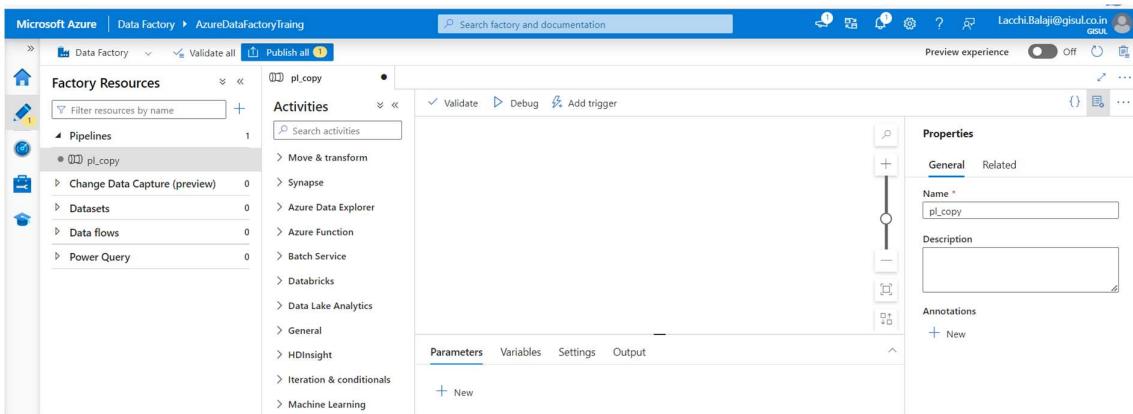
7. Test the connection and click on Create.

The screenshot shows the 'New linked service' configuration for 'Azure Blob Storage'. The 'Name' field contains 'AzureBlobStorage\_txt'. The 'Description' field is empty. The 'Connect via integration runtime' dropdown is set to 'AutoResolveIntegrationRuntime'. The 'Authentication type' dropdown is set to 'Account key'. Under 'Account selection method', the radio button 'From Azure subscription' is selected. The 'Azure subscription' dropdown shows 'Microsoft Partner Network (df295c6e-136c-4598-9168-19fa9dba7fe1)'. The 'Storage account name' dropdown shows 'adftraing'. At the bottom, the 'Create' button is visible, and a message 'Connection successful' is displayed with a checkmark icon.

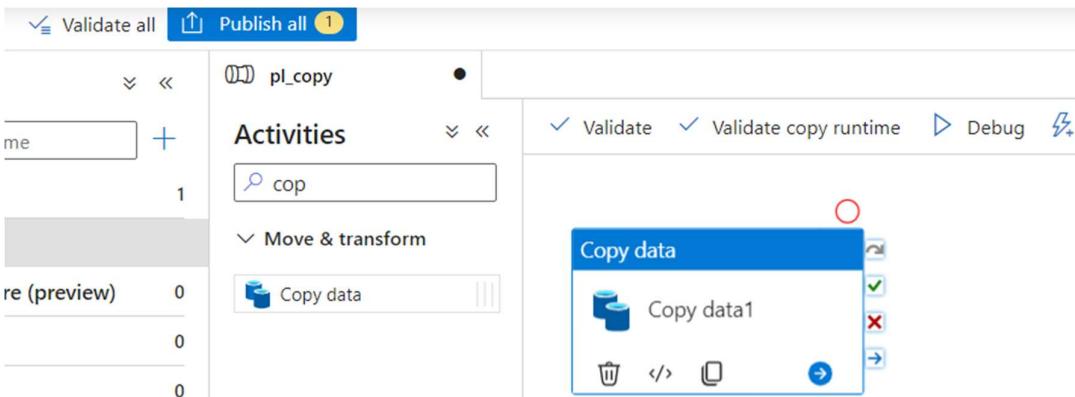
8. Under the Author tab click on the plus symbol and click on the plus symbol then click on the pipeline as shown below.



9. Give the Name as shown below.



10. Under Activities find the copy data and drag and drop it as shown below.



11. Under general give the name.

General   Source <sup>1</sup>   Sink <sup>1</sup>   Mapping   Settings   User properties

Name \*  [Learn more](#)

Description

Activity state (preview)  Active  Inactive

Timeout

12. Under source click on plus symbol to create a dataset.

General   **Source** <sup>1</sup>   Sink <sup>1</sup>   Mapping   Settings   User properties

Source dataset \*  [New](#)

New dataset

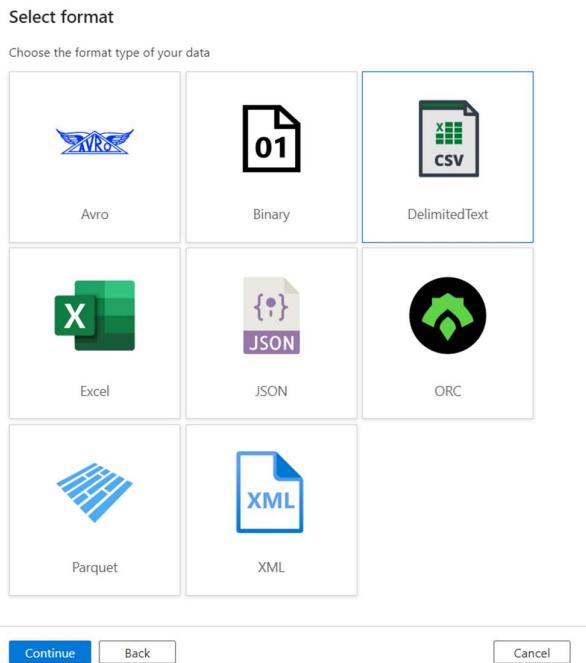
In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. [Learn more](#)

Select a data store

All   Azure   Database   File   Generic protocol   NoSQL   Services and apps

 Amazon Redshift	 Amazon S3	 Amazon S3 Compatible
 Apache Impala	 Azure Blob Storage	 Azure Cosmos DB for MongoDB
		

14. Select Delimited text and click on continue.



15. Give the name and linked service then click on the file symbol and select the file.

16. Check the None option and click on OK.

### Set properties

Name  
ds\_src\_txt

Linked service \*  
AzureBlobStorage\_txt

File path  
f-demo / Directory / Std\_details.txt

First row as header

Import schema  
 From connection/store  From sample file  None

> Advanced

17. Under sink click on the plus symbol to create a destination dataset.

General Source Sink<sup>1</sup> Mapping Settings User properties

Sink dataset \* Select... + New

18. Select the Axure Blog Storage and click on continue. Then click on Delimited text and click on Continue.
19. Give the name and linked service then under the file path give the container name and directory as output and check the below properties.

### Set properties

Name

Linked service \*  
 

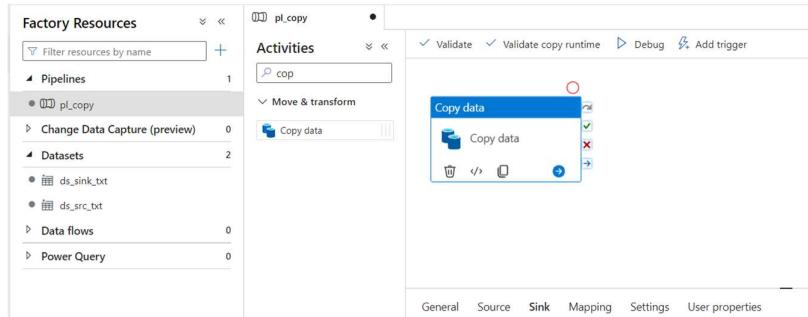
File path  
 /  /   | 

First row as header

Import schema  
 From connection/store  From sample file  None

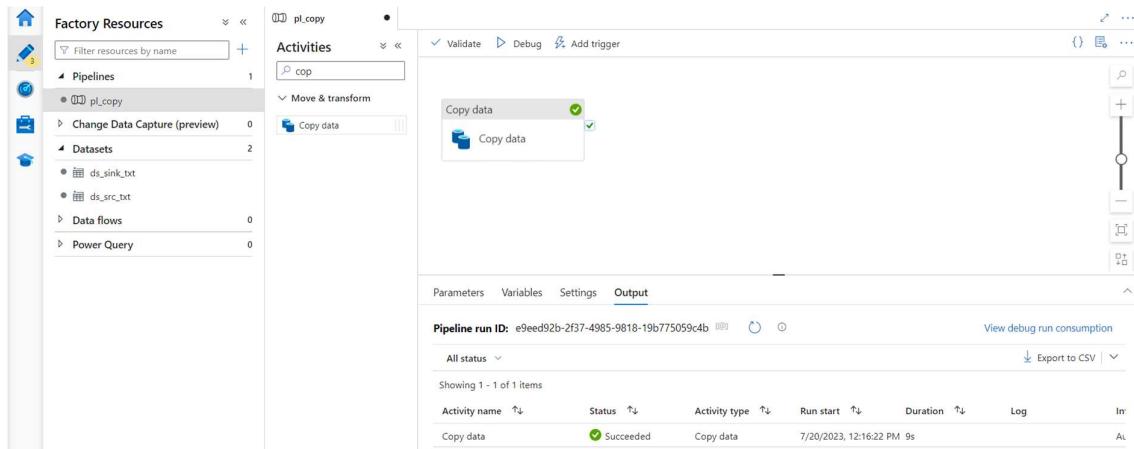


20. Now click on Validate. Next, click on Debug.



The screenshot shows the Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (2), and 'Power Query' (0). The main area shows the 'Activities' pane with a 'Copy data' activity selected. Below it, the 'Move & transform' section also contains a 'Copy data' activity. The top navigation bar includes 'Validate', 'Validate copy runtime', 'Debug', and 'Add trigger' buttons. The bottom navigation bar has tabs for 'General', 'Source', 'Sink', 'Mapping', 'Settings', and 'User properties'.

21. Now pipeline is executed successfully.



The screenshot shows the 'Pipeline run details' page for the 'pl\_copy' pipeline. The left sidebar shows the same resource structure as the editor. The main area displays the 'Copy data' activity with a green checkmark indicating success. Below it, the 'Output' tab is selected, showing the 'Pipeline run ID' (e9eed92b-2f37-4985-9818-19b77509c4b) and a table of activity logs. The log table has columns for 'Activity name', 'Status', 'Activity type', 'Run start', 'Duration', 'Log', and 'In'. One entry is shown: 'Copy data' with status 'Succeeded', activity type 'Copy data', run start '7/20/2023, 12:16:22 PM', duration '9s', and log 'All'.

22. Now go to the container you will see a folder as Output, click on it.

The screenshot shows the Azure Storage Explorer interface for a container named 'f-demo'. On the left, there's a sidebar with options like Overview, Diagnose and solve problems, Access Control (IAM), Settings, Shared access tokens, Access policy, Properties, and Metadata. The 'Overview' tab is selected. In the main area, there's a search bar, upload and refresh buttons, and a delete link. Below that, it says 'Authentication method: Access key (Switch to Azure AD User Account)' and 'Location: f-demo'. There's also a 'Search blobs by prefix (case-sensitive)' input field. A 'Add filter' button is present. The file list shows two items: 'Output' (a folder) and 'Std\_details.txt' (a file). The 'Output' folder is currently selected, indicated by a blue border around its row.

Name	Modified
<input type="checkbox"/> [..]	
<input type="checkbox"/> Std_details.txt	7/20/2023

23. Now we can see our Std\_details file is copied.

Name	Modified
<input type="checkbox"/> [..]	
<input type="checkbox"/> Std_details.txt	7/20/2023

# Parquet to CSV

1. In this example we are loading data from the parquet file to csv file.
2. Upload a parquet file in our container.

The screenshot shows the Azure Blob Storage interface for the 'demo' container. At the top, there are navigation links: Home > adftraing | Containers >. Below this is a search bar and a toolbar with actions like Upload, Change access level, Refresh, Delete, Change tier, Acquire lease, Break lease, and View snapshot. The Authentication method is set to Access key (Switch to Azure AD User Account). The container name is demo. A sidebar on the left lists Settings, Shared access tokens, Access policy, Properties, and Metadata. The main area displays a table of blobs:

	Name	Modified	Access tier	Archive status	Blob type
<input type="checkbox"/>	dept.csv	9/4/2023, 1:01:17 PM	Hot		Block blob
<input checked="" type="checkbox"/>	NYCTripSmall.parquet	9/5/2023, 11:48:14 AM	Hot (Inferred)		Block blob
<input type="checkbox"/>	Std_details.csv	9/4/2023, 12:37:52 PM	Hot		Block blob

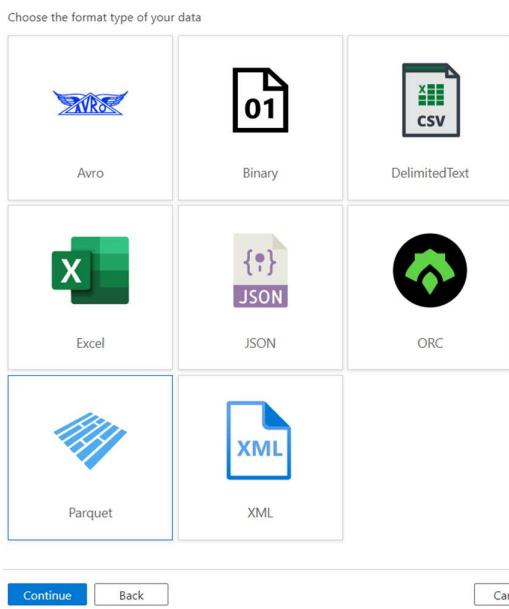
3. Go to Azure Data Factory and click on the plus symbol then click on dataset.

The screenshot shows the Azure Data Factory 'Factory Resources' blade. On the left is a navigation menu with icons for Data Factory, Pipelines, Change Data Capture (preview), Datasets, Data flows, and Power Query. On the right, a dropdown menu is open under 'Dataset' with options: Pipeline, Change Data Capture (preview), Dataset, Data flow, Power Query, and Copy Data tool. The 'Dataset' option is highlighted.

4. Select Azure Blob Storage and click on Continue.

The screenshot shows the 'Select a data store' dialog. At the top, it says 'Select a data store' and has a search bar. Below are tabs for All, Azure, Database, File, Generic protocol, NoSQL, and Services and apps. Under the 'Services and apps' tab, the 'Azure Blob Storage' icon is highlighted with a blue border. Other options include Amazon Redshift, Amazon S3, Amazon S3 Compatible, Apache Impala, Azure Cosmos DB for MongoDB, and other services like MySQL, PostgreSQL, Oracle, and SAP HANA.

5. Select Parquet and click on Continue.



Continue

Back

Cancel

6. Give the name, linked service and select the file then click on create.

#### Set properties

##### Name

DemoParquet

##### Linked service \*

AzureBlobStorage\_txt



##### File path

demo

/ Directory

/ NYCTripSmall.parquet



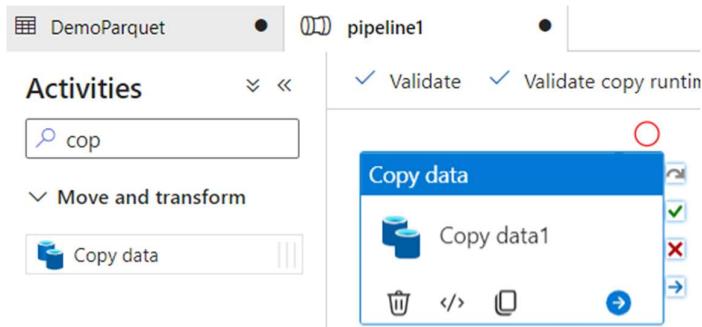
##### Import schema

From connection/store  From sample file  None

7. Create a Pipeline.

The screenshot shows the 'Factory Resources' blade. On the left, there's a sidebar with 'Pipelines', 'Change Data Capture (preview)', and a 'Datasets' section containing 'AzureTable', 'Cast\_src', and 'CitizenDS'. The main area shows a list of resources: 'Pipeline' (highlighted), 'Change Data Capture (preview)', 'Dataset', 'Data flow', 'Power Query', and 'Copy Data tool'. A context menu is open over the 'Pipeline' resource, with the 'Pipeline' option also highlighted.

8. Drag and drop the copy data activity.



9. Select the Dataset that we created before.

A screenshot of the 'Source' tab configuration for a 'Copy data' activity. The 'Source dataset' dropdown is set to 'DemoParquet'. Below it are buttons for 'Open', 'New', 'Preview data', and 'Learn more'. Under 'File path type', 'File path in dataset' is selected. There are sections for filtering by last modified date and time, and a 'Recursively' checkbox which is checked. The 'Sink' tab is also visible at the top.

10. Select the destination dataset that we created in the previous example.

A screenshot of the 'Sink' tab configuration for a 'Copy data' activity. The 'Sink dataset' dropdown is set to 'ds\_sink\_txt'. Below it are buttons for 'Open', 'New', and 'Learn more'. Under 'Copy behavior', there is a dropdown menu labeled 'Select...'. There are also fields for 'Max concurrent connections' and 'Block size (MB)'. At the bottom, there is a 'Metadata' section with a '+ New' button. The 'General' tab is also visible at the top.

11. Validate the Pipeline and click on debug.

12. Here our pipeline executed successfully.

The screenshot shows the 'Parquet\_pipeline' run details. The 'Copy data' step has completed successfully. The pipeline run ID is f764b4e7-d879-4ba7-9eec-85054eaaa15c, and the status is Succeeded. There are links to view debug run consumption, monitor in Azure Metrics, export to CSV, and a dropdown for all status.

13. Go to Destination and open the file. Here you cannot preview the data or edit it.

14. So, click on Download.

The screenshot shows a download interface with Save, Discard, Download, Refresh, and Delete buttons. Below these are tabs for Overview, Versions, Snapshots, Edit (which is selected), and Generate SAS. A warning message says 'File size of '22.7MB' exceeds max supported file size of '2.1MB.'

15. Open the CSV file and check the data.

The screenshot shows a Microsoft Excel spreadsheet titled 'NYCTripSmall'. The table has approximately 27 columns and over 70 rows of data. The columns represent various trip metrics such as DateID, Medallion, HackneyLic, PickupTime, DropoffTime, PickupGeo, DropoffGeo, PickupLat, PickupLon, DropoffLat, DropoffLon, PassengerCount, TripDuration, TripDistance, PaymentType, FareAmount, TipAmount, Surcharge, TaxAmount, TipAmount, TollAmount, and TotalAmount. The data spans from row 1 to row 70.

DateID	Medallion	HackneyLic	PickupTime	DropoffTime	PickupGeo	DropoffGeo	PickupLat	PickupLon	DropoffLat	DropoffLon	PassengerCount	TripDuration	TripDistance	PaymentType	FareAmount	TipAmount	TaxAmount	Surcharge	TipAmount	TollAmount	TotalAmount		
2	2013231	10624	35117	63879	64392	172341	219532	40.8124	-73.9513	40.8124,-7	10,982	-73.9444	40.982,-7	1	513	1.3	CSH	7.5	1	0.5	0	9	
3	2013231	7263	11597	61733	62114	87083	137768	40.7352	-73.9857	40.7352,-7	40,751	-73.9789	40.751,-7	1	380	1.3	CSH	7	1	0.5	0	8.5	
4	2013231	1899	5301	59488	59815	82164	146431	40.7699	-73.9483	40.7699,-7	40,763	-73.962	40.763,-7	1	326	1	CSH	6	1	0.5	0	7.5	
5	2013231	9471	28727	69093	69548	276342	43408	40.7507	-73.9740	40.7507,-7	40,7428	-73.9823	40.7428,-7	1	454	1	CSH	7	1	0.5	0	8.5	
6	2013231	2556	16124	51720	52440	155261	211054	40.7138	-74.0119	40.7138,-7	1	720	6.11	CSH	19.5	0	0.5	0	0	0	20		
7	2013231	1433	33809	31260	32100	263308	80576	40.726	-73.9773	40.726,-7	40,749	-73.9838	40.749,-7	1	840	2.16	CSH	11.5	0	0.5	0	12	
8	2013231	1486	7842	18150	20112	26661	108578	40.7421	-74.0084	40.7421,-7	40,757	-73.8868	40.757,-7	1	1962	9.2	CSH	32	0.5	0.5	0	33	
9	2013231	7225	36224	37020	38460	147003	70493	40.7697	-73.8637	40.7697,-7	40,6519	-74.0064	40.6519,-7	1	1440	14.78	CSH	40.5	0	0.5	0	41	
10	2013231	8726	35099	64740	65700	230937	116813	40.7763	-74.0022	40.763,-7	40,755	-73.9729	40.755,-7	1	960	3.25	CRD	13.5	1	0.5	4.35	0	19.35
11	2013231	573	30912	66600	67260	225983	76697	40.7698	-73.9822	40.7698,-7	40,816	-73.9605	40.816,-7	1	660	3.92	CRD	14	1	0.5	3.75	0	19.25
12	2013231	11778	18618	53157	53626	9876	82971	40.7053	-74.007	40.7053,-7	40,7339	-73.9771	40.7339,-7	1	469	3.5	CRD	12	0	0.5	3.12	0	15.62
13	2013231	12718	40002	51938	52399	137506	114398	40.7803	-73.9838	40.7803,-7	40,7252	-73.9988	40.7252,-7	1	460	1.5	CRD	7.5	0	0.5	1.25	0	9.25
14	2013231	8742	1969	900	1320	185301	290646	40.7292	-73.9975	40.7292,-7	40,754	-73.999	40.754,-7	2	420	2.14	CRD	8	0.5	0.5	1	0	10
15	2013231	3303	17758	73007	73177	11720	178048	40.763	-73.9742	40.763,-7	40,7587	-73.9745	40.7587,-7	2	169	0.4	CRD	4	0.5	0.5	1	0	6
16	2013231	8648	39568	61800	62580	276223	196088	40.779	-73.9548	40.779,-7	40,8199	-73.9468	40.8199,-7	2	780	3.69	CRD	13.5	1	0.5	1	0	16
17	2013231	9044	11325	65555	66297	143603	118847	40.7728	-73.9495	40.7728,-7	40,786	-73.9762	40.786,-7	1	741	2.1	CRD	10.5	1	0.5	2.4	0	14.4
18	2013231	10699	6893	61313	61734	127715	205212	40.7639	-73.9739	40.7639,-7	40,783	-73.9559	40.783,-7	1	420	1.7	CRD	7.5	1	0.5	1.8	0	10.8
19	2013231	1759	6604	58923	59435	37945	40745	40.8224	-73.9422	40.8224,-7	40,8049	-73.955	40.8049,-7	1	512	1.3	CRD	8	1	0.5	2.35	0	11.85
20	2013231	12446	15438	74160	74940	211081	110675	40.7749	-73.9652	40.7749,-7	40,753	-73.984	40.753,-7	1	780	2.34	CRD	11	0.5	0.5	2.3	0	14.3
21	2013231	6056	15282	77100	77820	296972	105491	40.7396	-73.987	40.7396,-7	40,734	-74.0046	40.734,-7	1	720	1.44	CRD	9	0.5	0.5	1.9	0	11.9
22	2013231	9232	13399	76620	76980	103218	188360	40.7373	-74.0051	40.7373,-7	40,7474	-74.0004	40.7474,-7	1	360	0.97	CRD	5.5	0.5	0.5	2	0	8.5
23	2013231	9169	29634	11759	12253	9874	110858	40.7734	-73.9893	40.7734,-7	40,7494	-73.9919	40.7494,-7	1	493	2.2	CRD	9	0.5	0.5	2	0	12
24	2013231	1627	35059	61140	61380	103699	180226	40.781	-73.9815	40.781,-7	40,7861	-73.9774	40.7861,-7	3	240	0.58	CRD	4.5	1	0.5	0	0	6
25	2013231	3615	13532	85740	86100	21537	227387	40.6957	-73.9088	40.6957,-7	40,6979	-73.9275	40.6979,-7	4	360	1.26	CRD	6.5	0.5	0.5	0	0	7.5
26	2013231	3866	23150	18960	19620	184163	184163	40.676	-74.0006	40.676,-7	40,6761	-74.0007	40.6761,-7	1	660	2.1	CRD	10.5	0.5	0.5	0	0	11.5
27	2013231	8529	34782	79988	80751	73243	290373	40.7349	-74.0022	40.7349,-7	40,7095	-74.0054	40.7095,-7	1	762	2.1	CRD	10.5	0.5	0.5	0	0	11.5

# Create SQL Server

1. On the Home page click on SQL servers.



2. Click on Create SQL Server as shown below.

3. Under basic give the properties as shown below.

Basics Networking Additional settings Tags Review + create

SQL database server is a logical container for managing databases and elastic pools. Complete the Basic tab, then go to Review + Create to provision with smart defaults, or visit each tab to customize. [Learn more](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*  Resource group \*  [Create new](#)

**Server details**

Enter required settings for this server, including providing a name and location.

Server name \*  .database.windows.net

Location \*

4. Scroll down and give the login name and password then click on the Next option.

**Authentication**

Select your preferred authentication methods for accessing this server. Create a server admin login and password to access your server with SQL authentication, select only Azure AD authentication [Learn more](#) using an existing Azure AD user, group, or application as Azure AD admin [Learn more](#), or select both SQL and Azure AD authentication.

Authentication method  Use only Azure Active Directory (Azure AD) authentication  Use both SQL and Azure AD authentication  Use SQL authentication

Server admin login \*

Password \*

Confirm password \*

[Review + create](#) | [Next : Networking >](#)

5. Set the below properties and click on Review+Create

## Create SQL Database Server

Microsoft

Basics Networking Additional settings Tags Review + create

Configure networking access for your server.

### Firewall rules

Allow Azure services and resources to access this server

Yes  No

[Review + create](#) [< Previous](#) [Next : Additional settings >](#)

6. Next click on Create.

Basics Networking Additional settings Tags Review + create

### Product details

SQL Database Server  
by Microsoft  
[Terms of use](#) | [Privacy policy](#)

**Estimated cost per month**  
No additional charges

### Terms

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Market frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and third-party offerings. For additional details see [Azure Marketplace Terms](#).

### Basics

Subscription	Microsoft Partner Network
Resource group	ADFResource
Server name	adfsqservertrail
Authentication method	SQL authentication
Server admin login	adminsql
Location	East US

### Networking

[Create](#) [< Previous](#) [Download a template for automation](#)

7. After some time you will see a screen like below.

The screenshot shows the Microsoft Azure Deployment Overview page. At the top, it displays the deployment name: Microsoft.SQLServer.createServer\_fca55494a9eb4c27a5fcf5aee35d70f | Overview. Below this, a green checkmark indicates "Your deployment is complete". The deployment details are listed: Deployment name : Microsoft.SQLServer.createServer\_fca55494a9eb4c27..., Start time : 7/20/2023, 1:25:21 PM, Subscription : Microsoft Partner Network, Correlation ID : 5c1fa646-8fc8-4155-9ebf-9db2786e9654, and Resource group : ADFResource.

## Create SQL database

1. On the Home page click on SQL database.

The screenshot shows the Microsoft Azure Home page. At the top, there is a "Create a resource" button. Below it, there are several service icons: SQL databases, Storage accounts, Data factories, App registrations, Quickstart center, Virtual machines, App Services, Azure Cosmos DB, and More services.

2. Next click on Create SQL database.

The screenshot shows the Microsoft Azure SQL databases blade. It displays a message: "No SQL databases to display" and "Try changing or clearing your filters.". There is a prominent "Create SQL database" button.

3. Give the Database name as sqldb and select the server that we created before then set the properties as shown below. And click on Review+Create.

The screenshot shows the "Create SQL Database" wizard. In the "Database details" step, the following settings are configured:

- Subscription: Microsoft Partner Network
- Resource group: ADFResource
- Database name: sqldb
- Server: adfsqservertrail (East US)
- Want to use SQL elastic pool?: No
- Workload environment: Development

A note at the bottom states: "Default settings provided for Development workloads. Configurations can be modified as needed."

[Review + create](#)

[Next : Networking >](#)

4. Next click on Create.
5. After some time you will see the below screen as shown.

The screenshot shows the 'Deployment' blade for a resource named 'Microsoft.SQLDatabase.newDatabaseExistingServer\_02f8fa5d79fd47e5'. The status is 'Your deployment is complete'. Deployment details include a name, subscription (Microsoft Partner Network), start time (7/20/2023, 1:31:52 PM), correlation ID, and resource group (ADFResource). Navigation links for Overview, Inputs, Outputs, and Template are visible on the left.

6. Now click on the SQL Database that we created just now.
7. Go to Query editor, give login credentials that we created in the sql server click on Ok.
8. If it asks to allow access to an IP address allow it.

The screenshot shows the 'Query editor (preview)' blade for the 'sqldatabase (sampleadmins1/sqldatabase)' database. It displays connection options for 'SQL server authentication' (Login: admins1, Password: redacted) and 'Active Directory authentication' (Continue as Lacchi.Balaji@gisul.co.in). A large 'OK' button is at the bottom right. The left sidebar shows navigation links like Overview, Activity log, Tags, Diagnose and solve problems, Getting started, and Query editor (preview).

9. A Query editor will open so that we can create tables and export tables also.
10. I have created a table using sql query.

The screenshot shows the 'Query editor (preview)' blade for the 'sqldatabase (admins1)' database. A table named 'TotalSale' has been created with the following schema:

```

CREATE TABLE TotalSale (
    id int NOT NULL,
    SalePersonName varchar(100) NULL,
    ProductName varchar(100) NULL,
    ItemsSold int NULL,
    SoldPrice int NULL,
    SoldDate Date NULL,
    Country varchar(100) NULL,
    Region varchar(100) NULL
)
  
```

The left sidebar shows navigation links like Overview, Activity log, Tags, Diagnose and solve problems, Getting started, and Query editor (preview).

11. Below is the table I have created.

The screenshot shows the Azure Data Studio interface. On the left, the Object Explorer displays a database named 'sqldatabase (adminsql1)' with a 'Tables' node expanded, showing the 'dbo.TotalSale' table. This table has columns: id, SalePersonFName, SalePersonLName, ProductName, ItemsSold, SoldPrice, Country, and Region. The 'Results' tab on the right shows the data from the 'TotalSale' table:

id	SalePersonFName	SalePersonLName	ProductName	ItemsSold	SoldPrice	Country	Region
1	Aamir	Shahzad	TV	1	700	USA	North America
2	M	Raza	Cell Phone	2	800	USA	North America
3	Christy	Ladson	TV	3	1600	USA	North America
4	John	Rivers	Laptop	5	2400	USA	North America
5	Najaf	Ali	Computer	1	300	Pakistan	Asia
6	Sukhjeet	Singh	TV	2	900	India	Asia
7	Chirag	Patel	Cell Phone	5	1500	India	Asia
8	Aleena	Aman	Laptop	2	800	Pakistan	Asia
9	Petra	Henry	TV	10	5000	France	Europe
10	Rita	Roger	Laptop	7	2100	France	Europe

## Create a Linked Services

1. Go to Azure Data Factory.
2. Go to the Manage tab and select Linked Services then click on the plus symbol to create.

The screenshot shows the Azure Data Factory 'Linked services' page. The left sidebar has 'General' selected. The main area shows a table with one item:

Name	Type
AzureBlobStorage_txt	Azure

3. Select the SQL Azure Database and click on Continue.

The screenshot shows the 'Data store' selection screen in Azure Data Factory. The 'Compute' tab is selected. Under the 'All' category, several database options are listed:

- Amazon RDS for SQL Server
- Azure Cosmos DB for NoSQL
- Azure Database for MySQL
- Azure Database for PostgreSQL
- Azure SQL Database
- Azure SQL Database Managed Instance
- MySQL
- Oracle

At the bottom, there are 'Continue' and 'Cancel' buttons.

4. Give the Name and Runtime.

New linked service

Azure SQL Database [Learn more](#)

Name \*

Description

Connect via integration runtime \* ⓘ

AutoResolveIntegrationRuntime

5. And give the below properties and test the connection then click on create.

Connect via integration runtime

AutoResolveIntegrationRuntime

**Connection string** **Azure Key Vault**

Account selection method ⓘ

From Azure subscription  Enter manually

Azure subscription

Select all

Server name \*

sampleadmins1sql

Database name \*

sqldatabase

Authentication type \*

SQL authentication

User name \*

admins1sql1

Add dynamic content [Alt+Shift+D]

**Password** **Azure Key Vault**

Password \*

\*\*\*\*\*

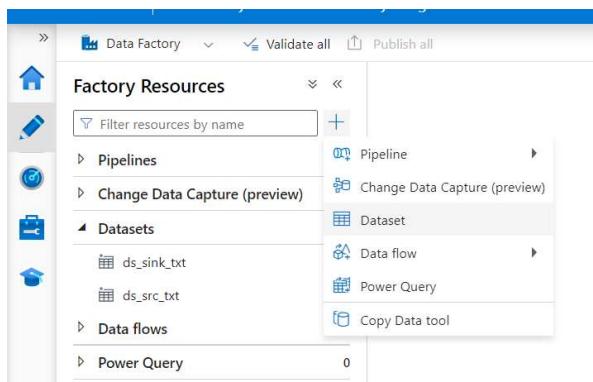
 Connection successful

 Test connection

Create Back Cancel

# Create Dataset

1. Go to the Author tab and click on Plus symbol then click on Dataset.

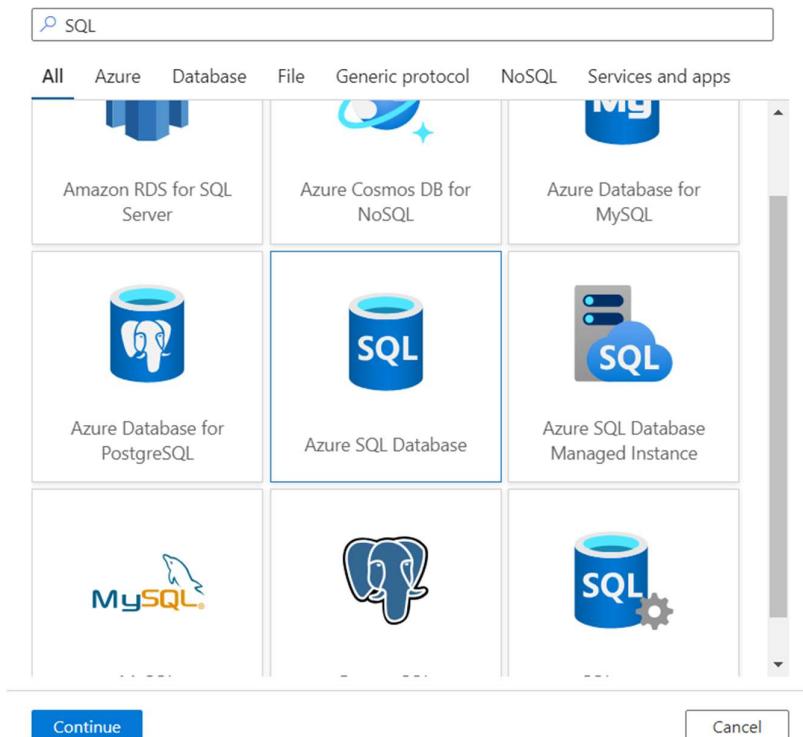


2. Select Azure SQL Database and click on Continue.

## New dataset

In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. [Learn more](#)

Select a data store



SQL

All Azure Database File Generic protocol NoSQL Services and apps

Amazon RDS for SQL Server	Azure Cosmos DB for NoSQL	Azure Database for MySQL
Azure Database for PostgreSQL	Azure SQL Database	Azure SQL Database Managed Instance
MySQL	Oracle	SQL Server

Continue Cancel

- Give the Name and select the linked service and table that we created before. Then check the none option and click on Ok.

#### Set properties

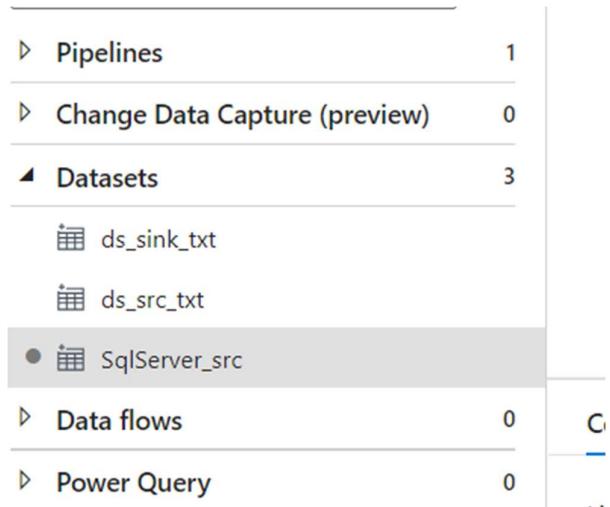
Name  
SqlServer\_src

Linked service \*  
AzureSqlDatabase

Table name  
dbo.TotalSale

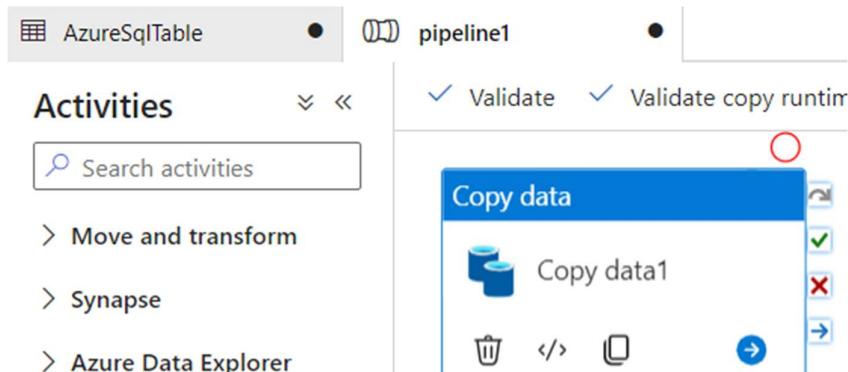
Import schema  
 From connection/store  None

- Under the dataset you can see our sql server dataset as shown below.



## SQL to csv

- Create a pipeline.
- Drag and drop the copy data activity.



3. Next select the Source Dataset that we created before.

General    **Source**    Sink    Mapping    Settings    User properties

Source dataset \*

Use query  Table  Query  Stored procedure

Query timeout (minutes)

Isolation level

Partition option  None  Physical partitions of table  Dynamic range

4. Select the Target dataset that we created in the first ADF pipeline.

General    Source    **Sink**    Mapping    Settings    User properties

Sink dataset \*

Copy behavior

Max concurrent connections

Block size (MB)

5. Validate the pipeline and click on Debug.  
6. Here Our pipeline executed successfully.

Copy data   
 Copy data1

Parameters    Variables    Settings    **Output**

Pipeline run ID: 01905a12-1703-42b1-851b-bf14e4f8d486   Pipeline status  Succeeded

7. Go to Destination and check the Data.

A screenshot of the Azure Storage Explorer interface. At the top, there is a search bar labeled "Search blobs by prefix (case-sensitive)". Below the search bar is a button labeled "Add filter". The main area displays a table with two columns: "Name" and "Modified". The table contains the following data:

Name	Modified
Assert.csv	8/22/2023
badRow.csv	8/17/2023
data_caec6c7e-12d2-4156-bef3-e502c5f1ea97_0f2...	8/18/2023
<input checked="" type="checkbox"/> dbo.TotalSale.txt	9/5/2023,
demo.csv	8/23/2023
Dent Flowlet.csv	8/25/2023

## Azure Key Vault

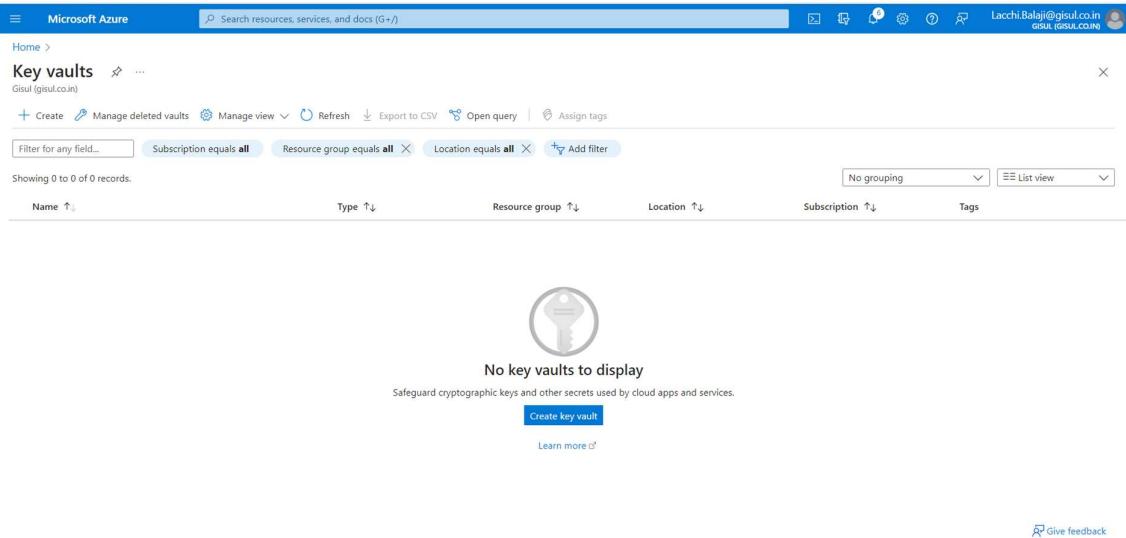
Azure Key Vault is a secure and centralized service for managing sensitive information like keys, secrets, certificates, and more. When creating a Linked Service in Azure Data Factory, you might need to use Azure Key Vault to securely store and retrieve credentials or other sensitive information required for your data integration processes.

By using Azure Key Vault, you can enhance the security of your Azure Data Factory by centralizing and managing sensitive information separately from your data integration pipelines, making it easier to rotate secrets and control access to them.

1. In this example we are going to create a Linked service using Azure Key vault.
2. On the Home page search for Key Vault and click on it.

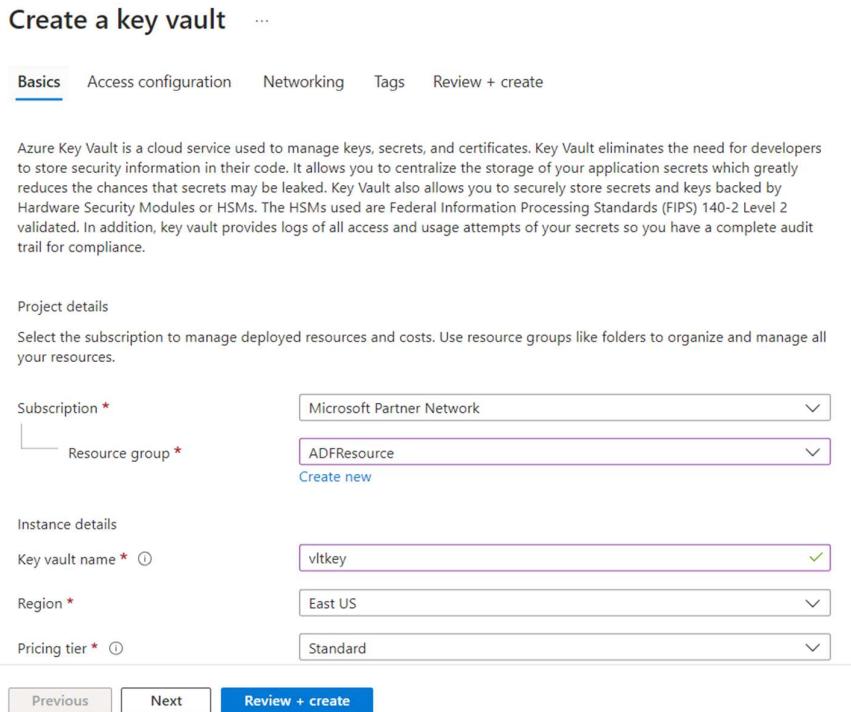
A screenshot of the Microsoft Azure portal. The URL in the address bar is <https://portal.azure.com/#home>. The search bar at the top contains the text "key". The search results show a list of services under the "Services" category. The "Key vaults" item is highlighted with a gray background. Other items listed include "SSH keys", "BitLocker Keys", "Azure Key Vault Managed HSMs", "Language", and "Azure AI services".

### 3. Click on Create Key Vault.



The screenshot shows the Microsoft Azure Key vaults page. At the top, there's a search bar and a user profile. Below it, the title 'Key vaults' is displayed with a 'Gisul (gisul.co.in)' link. A navigation bar includes '+ Create', 'Manage deleted vaults', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', and 'Assign tags'. There are also filter options for 'Subscription equals all', 'Resource group equals all', and 'Location equals all', along with a 'Add filter' button. The main area shows a message 'No key vaults to display' with a key icon, followed by the text 'Safeguard cryptographic keys and other secrets used by cloud apps and services.' A prominent blue 'Create key vault' button is centered. At the bottom right, there's a 'Give feedback' link.

### 4. Give the Resource group and Name then click on Next.



The screenshot shows the 'Create a key vault' wizard on the 'Basics' tab. The title 'Create a key vault' is at the top, followed by a ellipsis. Below it are tabs for 'Basics', 'Access configuration', 'Networking', 'Tags', and 'Review + create'. The 'Basics' tab is selected. A descriptive text explains that Azure Key Vault is a cloud service for managing keys, secrets, and certificates. It highlights features like centralizing secret storage and using Hardware Security Modules (HSMs) backed by FIPS 140-2 Level 2 validation. Under 'Project details', it says to select a subscription and resource group. The 'Subscription' dropdown is set to 'Microsoft Partner Network'. The 'Resource group' dropdown shows 'ADFResource' with a 'Create new' option. Under 'Instance details', the 'Key vault name' is set to 'vltkey', 'Region' is 'East US', and 'Pricing tier' is 'Standard'. At the bottom are 'Previous', 'Next', and 'Review + create' buttons.

5. Set the below properties as shown and click Review + Create.

Basics    Access configuration    Networking    Tags    Review + create

#### Configure data plane access for this key vault

To access a key vault in data plane, all callers (users or applications) must have proper permissions.

##### Permission model

Grant data plane access by using a [Azure RBAC](#) or [Key Vault access policy](#)

- Azure role-based access control (recommended) ⓘ
- Vault access policy ⓘ

##### Resource access

- Azure Virtual Machines for deployment ⓘ
- Azure Resource Manager for template deployment ⓘ
- Azure Disk Encryption for volume encryption ⓘ

##### Access policies

Access policies enable you to have fine grained control over access to vault items. [Learn more](#)

[Previous](#)    [Next](#)    [Review + create](#)

6. Click on Create.

Basics    Access configuration    Networking    Tags    Review + create

[View Automation Template](#)

##### Basics

Subscription	Microsoft Partner Network
Resource group	ADFResource
Key vault name	vltkey1
Region	East US
Pricing tier	Standard
Soft-delete	Enabled
Purge protection during retention period	Disabled
Days to retain deleted vaults	90 days

##### Access configuration

Azure Virtual Machines for deployment	Disabled
Azure Resource Manager for template deployment	Disabled
Azure Disk Encryption for volume encryption	Disabled

[Previous](#)    [Next](#)    [Create](#)

7. After deployment is completed, you will see the below screen.

The screenshot shows the 'Deployment' blade for 'vltkey1'. The main message is 'Your deployment is complete'. Deployment details include: Deployment name: vltkey1, Subscription: Microsoft Partner Network, Resource group: ADFResource. The start time was 9/5/2023, 4:16:02 PM, and the Correlation ID is 81860f2d-5eeb-467e-9816-05281451d91b. Below the message are links for 'Deployment details' and 'Next steps', and a 'Go to resource' button.

8. Go to Azure Storage Account, Under Access Keys, copy the connection string.

9. Paste it somewhere safe.

The screenshot shows the 'Access keys' blade for the storage account 'adftraing'. It displays the storage account name 'adftraing'. Under 'key1', the key was last rotated on 20/7/2023 (47 days ago). The 'Show' button reveals the connection string: DefaultEndpointsProtocol=https;AccountName=adftraing;AccountKey=PKz4Li8ii... .

10. Go to Key Vault, Under Secrets click on the plus symbol.

The screenshot shows the 'Secrets' blade for the key vault 'vltkey1'. The 'Objects' sidebar has 'Secrets' selected. The main area shows a table with columns 'Name' and 'Type'. A message states 'There are no secrets available.'

11. Give the Name and for the secret value give the Connection string that we copied from the storage account.
12. And click on Create.

Upload options: Manual

Name \*: mystorage

Secret value \*: \*\*\*\*\*

Content type (optional):

Set activation date:

Set expiration date:

Enabled: Yes

Tags: 0 tags

**Create** **Cancel**

13. Here we have created a secret.

The secret 'mystorage' has been successfully created.

Name	Type	Status
mystorage		✓ Enabled

14. Under Access policies click on the plus symbol to create Access to ADF.

Access policies enable you to have fine grained control over access to vault items. [Learn more](#)

Name	Email	Key Permissions
Lacchi Balaji	Lacchi.Balaji@gisul.co.in	Get, List, Update,

15. Set the below properties and click on Next.

Create an access policy ...

vltkey1

Permissions Principal Application (optional) Review + create

Configure from a template

Azure Data Lake Storage or Azure Storage

Key permissions	Secret permissions	Certificate permissions
Key Management Operations	Secret Management Operations	Certificate Management Operation:
<input type="checkbox"/> Select all	<input type="checkbox"/> Select all	<input type="checkbox"/> Select all
<input checked="" type="checkbox"/> Get	<input checked="" type="checkbox"/> Get	<input checked="" type="checkbox"/> Get
<input checked="" type="checkbox"/> List	<input checked="" type="checkbox"/> List	<input checked="" type="checkbox"/> List
<input type="checkbox"/> Update	<input type="checkbox"/> Set	<input type="checkbox"/> Update
<input type="checkbox"/> Create	<input type="checkbox"/> Delete	<input type="checkbox"/> Create
<input type="checkbox"/> Import	<input type="checkbox"/> Recover	<input type="checkbox"/> Import
<input type="checkbox"/> Delete	<input type="checkbox"/> Backup	<input type="checkbox"/> Delete
<input type="checkbox"/> Recover	<input type="checkbox"/> Restore	<input type="checkbox"/> Recover
<input type="checkbox"/> Backup		<input type="checkbox"/> Backup
<input type="checkbox"/> Restore	<input type="checkbox"/> Select all	<input type="checkbox"/> Restore
	Privileged Secret Operations	<input type="checkbox"/> Manage Contacts

Previous Next

16. Select your Azure Data Factory name and click on Next.

Create an access policy ...

vltkey1

Permissions Principal Application (optional) Review + create

Only 1 principal can be assigned per access policy.  
Use the new embedded experience to select a principal. The previous popup experience can be accessed here. [Select a principal](#)

Selected item

AzureDataFactoryTraing  
025001e8-382c-4336-a9f9-a245d5280816

Previous Next

17. Click on Next and Click on Create.

✓ Permissions   ✓ Principal   ✓ Application (optional)   4 Review + create

**Key Permissions**

Key Management Operations	Get, List
Cryptographic Operations	Unwrap Key, Wrap Key
Privileged Key Operations	None selected
Rotation Policy Operations	None selected

**Secret Permissions**

Secret Management Operations	Get, List
Privileged Secret Operations	None selected

**Certificate Permissions**

Certificate Management Operations	Get, List
Privileged Certificate Operations	None selected

**Principal**

Principal name	AzureDataFactoryTraining
Object ID	b0a7daf8-817c-47c6-acfd-3ea97dbf619e

[Previous](#) [Create](#)

18. Here we have created an Access policy for our Azure Data Factory.

+ Create   ⌂ Refresh | ⌁ Delete   ⌐ Edit

Access policies enable you to have fine grained control over access to vault items. [Learn more](#)

Search   Permissions : All   Type : All

Showing 1 to 2 of 2 records.

Name	Email	Key Permissions	Secret Permissions
AzureDataFactoryTraining		Get, List, Unwrap Key, Wrap Key	Get, List

19. In the Azure Data Factory, under Linked Services, click on the plus symbol.

Microsoft Azure | Data Factory > AzureDataFactoryTraining

Microsoft recently announced the public preview of Microsoft Fabric, a brand new and exciting way to build

Data Factory   Validate all   Publish all

General   Factory settings

Connections   Linked services   Integration runtimes   Microsoft Purview

**Linked services**

Linked service defines the connection information to a data store

+ New

Filter by name   Annotations : Any

Showing 1 - 8 of 8 items

Name	Type

20. Select the Azure Blob storage and click on continue.

New linked service

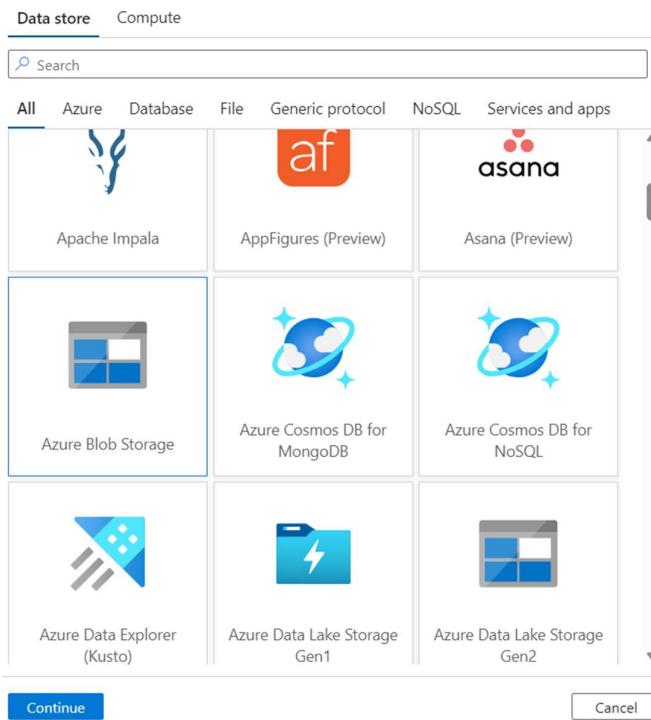
Data store   Compute

Search

All   Azure   Database   File   Generic protocol   NoSQL   Services and apps

Apache Impala	af	asana
Azure Blob Storage	Azure Cosmos DB for MongoDB	Azure Cosmos DB for NoSQL
Azure Data Explorer (Kusto)	Azure Data Lake Storage Gen1	Azure Data Lake Storage Gen2

Continue   Cancel



21. Give the name and click on Azure Key Vault then under AKV linked Service click on New.

New linked service

Azure Blob Storage   Learn more

Name \*  
AzureKeyStorage

Description

Connect via integration runtime \* ⓘ  
AutoResolveIntegrationRuntime

Authentication type  
Account key

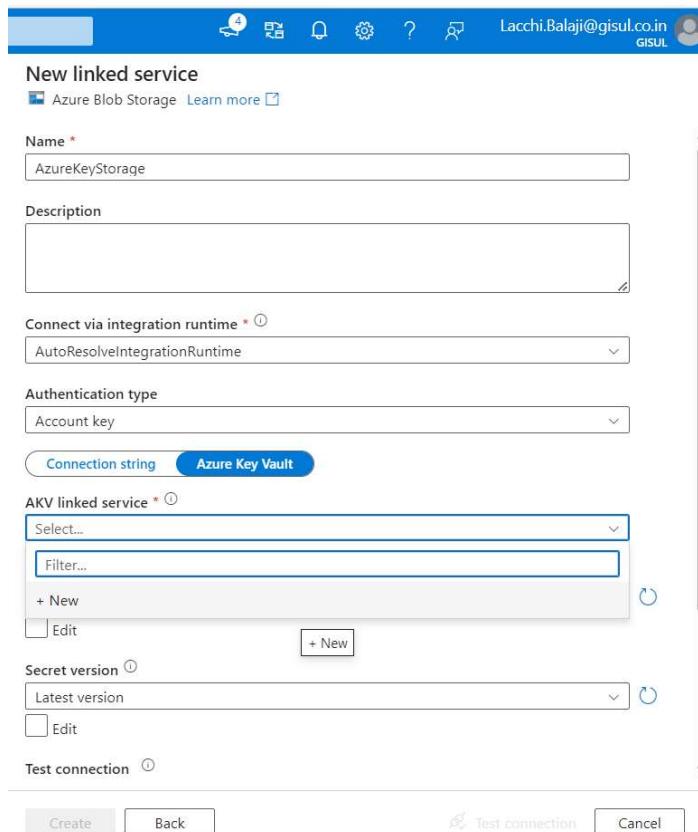
Connection string   Azure Key Vault

AKV linked service \* ⓘ  
Select...   Filter...   + New   + Edit   + New

Secret version ⓘ  
Latest version   + Edit

Test connection ⓘ

Create   Back   Test connection   Cancel



22. Give the Name and Key vault that we created then test the connection.

New linked service

Azure Key Vault

Name \*  
MyKeyVault

Description

Azure key vault selection method ⓘ  
 From Azure subscription  Enter manually

Azure subscription ⓘ  
Microsoft Partner Network (df295c6e-136c-4598-9168-19fa9dba7fe1)

Azure key vault name \*  
vtkey1

Edit key vault

Authentication method  
System Assigned Managed Identity

Managed identity name: **AzureDataFactoryTraining**  
Managed identity object ID: **b0a7daf8-817c-47c6-acfd-3ea97dbf619e**  
Grant Data Factory service managed identity access to your Azure Key Vault. [Learn more](#)

Test connection  
 To linked service  To secret

✓ Connection successful

[Create](#) [Cancel](#) [Test connection](#)

23. Under the secret name select the secret that we created before. And test the connection.

New linked service

Azure Blob Storage [Learn more](#)

Connect via integration runtime \* ⓘ  
AutoResolveIntegrationRuntime

Authentication type  
Account key

[Connection string](#) Azure Key Vault

AKV linked service \* ⓘ  
MyKeyVault

Secret name \* ⓘ  
mystorage

Edit

Secret version ⓘ  
Latest version

Edit

Test connection ⓘ  
 To linked service  To file path

Annotations

✓ Connection successful

[Create](#) [Back](#) [Test connection](#) [Cancel](#)

## Json to csv

1. In this example we are using the below JSON file. I have uploaded it to my storage account.

The screenshot shows the Azure Storage Explorer interface. On the left, there's a sidebar with options like Overview, Diagnose and solve problems, Access Control (IAM), Properties, and Metadata. The main area shows a list of files in a blob container named 'demo'. One file, 'Mobile\_det.json', is selected and its content is displayed on the right. The JSON content is as follows:

```
1  {
2   "id": 1,
3   "Product": [
4     {
5       "name": "mobile",
6       "modelname": "android",
7       "manufacturer": "oneplus"
8     }
9   ],
10  "pricerange": "5000-50000"
11 },
12  {
13   "id": 2,
14   "Product": [
15     {
16       "name": "mobile",
17       "modelname": "iphone",
18       "manufacturer": "apple"
19     }
20   ],
21 }
```

2. Go to ADF and create a Dataset for JSON file.

The screenshot shows the Microsoft Azure Data Factory portal. The top navigation bar says 'Microsoft Azure | Data Factory > AzureDataFactoryTraining'. Below it, there's a message about the public preview of Microsoft Fabric. The main area is titled 'Factory Resources' and contains a sidebar with icons for Pipelines, Change Data Capture (preview), Datasets, Data flows, and Power Query. To the right, there's a list of resources: Pipeline, Change Data Capture (preview), Dataset, Data flow, Power Query, and Copy Data tool.

3. Select Azure Blob Storage and click on continue.
4. Select Json and click on continue.

The screenshot shows the 'Select format' dialog in Azure Data Factory. It asks 'Choose the format type of your data' and lists several options: Avro, Binary, DelimitedText, Excel, JSON, ORC, Parquet, and XML. The 'JSON' option is highlighted with a blue border. At the bottom, there are 'Continue', 'Back', and 'Cancel' buttons.

5. Give the name, select the linked service that we created before selecting the JSON file then click on OK.

#### Set properties

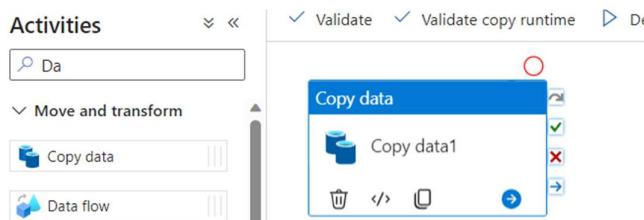
Name  
Jsondemo

Linked service \*  
AzureKeyStorage

File path  
demo / Directory / person.json

Import schema  
 From connection/store    From sample file    None

#### 6. Create a Pipeline.



7. Under source select the JSON dataset.

General   Source   Sink   Mapping   Settings   User properties

Source dataset \*  
Jsondemo

File path type  
 File path in dataset    Prefix    Wildcard file path

Filter by last modified ⓘ  
Start time (UTC) \_\_\_\_\_ End time (UTC) \_\_\_\_\_

8. Under Sink select the target dataset.

General   Source   Sink   Mapping   Settings   User properties

Sink dataset \*  
ds\_sink\_txt

Copy behavior ⓘ  
Select...

Max concurrent connections ⓘ  
\_\_\_\_\_

Block size (MB) ⓘ  
\_\_\_\_\_

9. Under the mapping set the mapping is shown below.

The screenshot shows the 'Mapping' tab in the Azure Data Factory pipeline editor. It displays a table of mappings:

Name	Type	Collection reference	Column name	Type	Include
id	ANY any		id	123 Int32	<input checked="" type="checkbox"/>
[\$'Product'][0]['name']	ANY any		name	abc String	<input checked="" type="checkbox"/>
[\$'Product'][0]['mode...']	ANY any		modelname	abc String	<input checked="" type="checkbox"/>
[\$'Product'][0]['manu...']	ANY any		manufacturer	abc String	<input checked="" type="checkbox"/>
pricerange	ANY any		pricerange	abc String	<input checked="" type="checkbox"/>

10. Validate the pipeline and click on debug.

11. Here our mapping was executed successfully.

The screenshot shows the 'Pipeline status' section of the Azure Data Factory pipeline editor. It displays the following information:

- Pipeline run ID: 52f9b230-c301-41d0-9658-7fa0bb144d33
- Pipeline status: Succeeded
- View details
- Monitor in Azure Metrics

12. Go to Destination and check the Data.

The screenshot shows the 'FilterOutput/Mobile\_det.txt' blob in the Azure Blob storage. It displays the following content:

```
1 id,name,modelname,manufacturer,pricerange
2 1,"mobile","android","oneplus","5000-50000"
3 2,"mobile","iphone","apple","50000-500000"
```

## XML to csv

1. In this example we are using below XML file. I have uploaded the file into the Azure Storage Account.

The screenshot shows the Azure Storage Blob service interface. On the left, there's a sidebar with a search bar and a list of files. The main area displays the content of the 'persondata.xml' file. The XML code is as follows:

```
<?xml version="1.0"?>
<addressbook>
  <contact>
    <name>John Doe</name>
    <email>john.doe@example.com</email>
    <phone>555-123-4567</phone>
  </contact>
  <contact>
    <name>Jane Smith</name>
    <email>jane.smith@example.com</email>
    <phone>555-987-6543</phone>
  </contact>
  <contact>
    <name>Bob Johnson</name>
    <email>bob.johnson@example.com</email>
    <phone>555-555-5555</phone>
  </contact>
</addressbook>
```

2. Create a Dataset.

The screenshot shows the Microsoft Azure Data Factory portal. The left sidebar lists 'Factory Resources' such as Pipelines, Change Data Capture (preview), Datasets, Data flows, and Power Query. A dropdown menu under 'Datasets' is open, showing options like Pipeline, Change Data Capture (preview), Dataset, Data flow, Power Query, and Copy Data tool.

3. Select Azure Blob storage and click on continue.
4. Select XML and click on continue.
5. Give the Name, Linked Service, select the file then click on Ok.

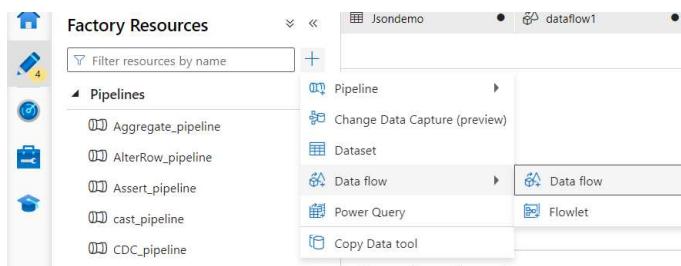
### Set properties

Name  
Xml1

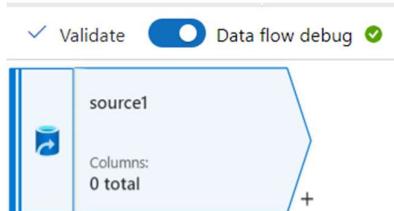
Linked service \*  
AzureKeyStorage

File path  
demo / Directory / persondata.xml

6. Create a data flow.



7. Add a Data source.



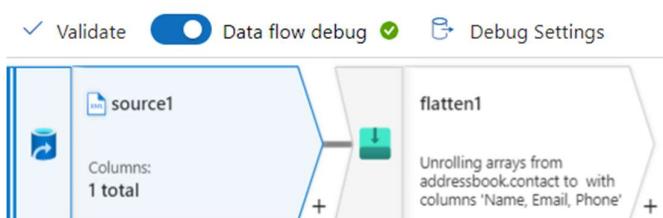
8. Under source select the XML source dataset.

The screenshot shows the 'Source settings' tab for 'source1'. It includes fields for 'Output stream name' (set to 'source1'), 'Description' (set to 'Import data from Xml'), and 'Source type' (set to 'Dataset'). The 'Dataset' button is highlighted. Below these are options for 'Dataset' (set to 'Xml'), 'Test connection', 'Open', and 'New'. Under 'Options', the 'Allow schema drift' checkbox is checked.

9. Under Project click on Import Projection.

The screenshot shows the 'Projection' tab for 'source1'. It includes buttons for 'Define default format', 'Import projection' (which is highlighted in blue), 'Reset schema', and 'Overwrite schema'. Below this is a table for defining columns, with a single row for 'addressbook' where the 'Type' is set to 'complex'.

10. Add Flatten Transformation.

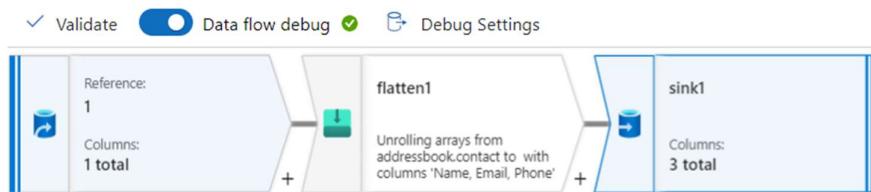


11. Under settings, select the Unroll by option and add below three columns below.

The screenshot shows the 'Flatten settings' blade with the following configuration:

- Incoming stream \***: source1
- Unroll by \***: addressbook.contact
- Unroll root**: {}
- Options**:
  - Skip duplicate input columns
  - Skip duplicate output columns
- Input columns \***:
  - source1's column
  - abc addressbook.contact.name
  - abc addressbook.contact.email
  - abc addressbook.contact.phone
- Output columns**:
  - Name as: Name
  - Name as: Email
  - Name as: Phone

12. Add Sink.



13. Under Sink, select the target dataset.

The screenshot shows the 'Sink' blade with the following configuration:

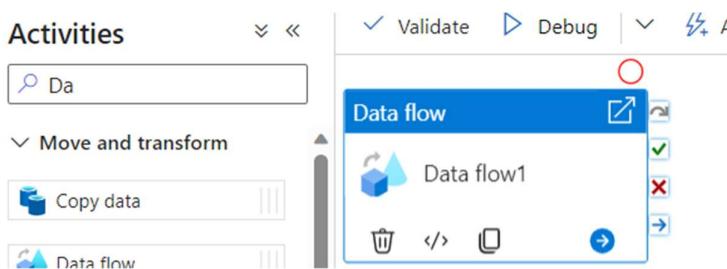
- Output stream name \***: sink1
- Description**: Export data to ds\_sink\_txt
- Incoming stream \***: flatten1
- Sink type \***: Dataset (selected)
- Dataset \***: ds\_sink\_txt

14. Under settings select the file option and give the file name.

The screenshot shows the 'Settings' blade with the following configuration:

- Clear the folder**:
- File name option \***: Output to single file
- Output to single file \***: XML.csv
- Quote All**:

15. Create a Pipeline, Drag and drop the data flow.



16. Under settings select Dataflow.

A screenshot of the 'Settings' tab for a Data flow named 'XML\_dataflow'. The 'Run on' dropdown is set to 'AutoResolveIntegrationRuntime'. The 'Compute size' dropdown is set to 'Small'. At the top right, there are 'Open' and 'New' buttons.

17. Validate the pipeline and click on debug.

18. Here our pipeline was executed successfully.

A screenshot of the pipeline run status page. It shows a successful run with ID 'c92883ca-a041-4191-a1e3-e040f14a03e5' and a status of 'Succeeded'. The pipeline name is 'Data flow' and it contains one step named 'Data flow1'. The 'Output' tab is selected.

19. Go to the destination and check our data.

A screenshot of a blob storage container named 'FilterOutput/XML.csv'. The container has a single file named 'blob'. The file is a CSV table with three columns: 'Name', 'Email', and 'Phone'. The data is as follows:

Name	Email	Phone
John Doe	john.doe@example.com	555-123-4567
Jane Smith	jane.smith@example.com	555-987-6543
Bob Johnson	bob.johnson@example.com	555-555-5555