

# Copy data from Azure Blob storage to a database in Azure SQL Database by using Azure Data Factory

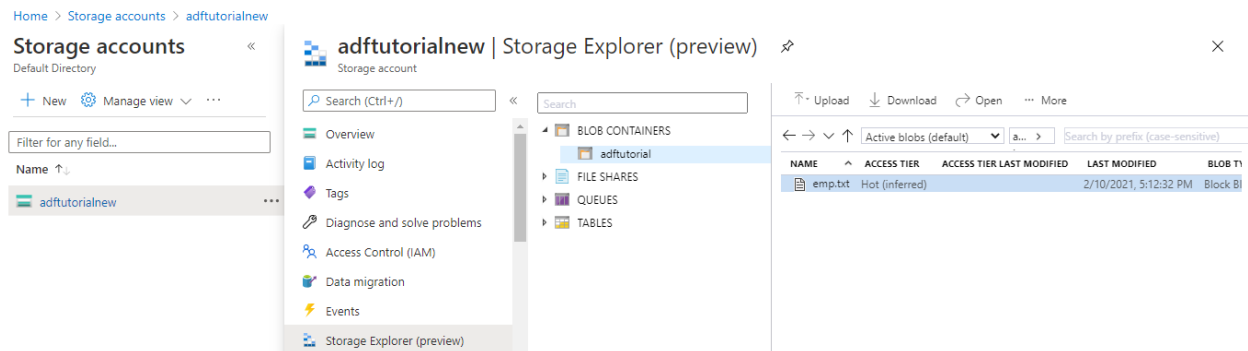
Pipeline in this data factory copies data from Azure Blob storage to a database in Azure SQL Database. This data pipeline copy's data from a file-based data store to a relational data store.

Reference Link:- <https://docs.microsoft.com/en-us/azure/data-factory/tutorial-copy-data-portal>

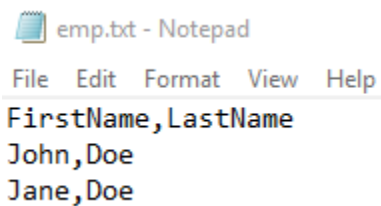
## Create a blob and a SQL table

### Create a source Blob

Create a container 'adftutorial' under which a folder named 'input' is created and the text file containing the employee details are uploaded in the BLOB Container.



### Contents of the text file



## Create a Sink SQL Table

Sink SQL table is the target SQL table in which data is loaded from BLOB storage.

Home > EmployeeDB (emplserver/EmployeeDB)

### EmployeeDB (emplserver/EmployeeDB) | Query editor (preview)

SQL database

Search (Ctrl+/) << Login + New Query ↑ Open query ♥ Feedback

Overview

Activity log

Tags

Diagnose and solve problems

Quick start

Query editor (preview)

Power Platform

Power BI (preview)

Power Apps (preview)

Power Automate (preview)

EmployeeDB (siddhi)

Showing limited object explorer here. For full capability please open SSDT.

Tables

dbo.emp

ID (int, not null)

FirstName (varchar, null)

LastName (varchar, null)

Views

Stored Procedures

Query 1 × Query 2 ×

Run Cancel query Save query Export data as Show only Editor

```
1 CREATE TABLE dbo.emp
2 (
3     ID int IDENTITY(1,1) NOT NULL,
4     FirstName varchar(50),
5     LastName varchar(50)
6 )
7 GO
8
```

Results Messages

Query succeeded: Affected rows: 0Affected rows: 0

# Create a Data Factory

Creating Data Factory to create a pipeline to load the data from BLOB container to Azure SQL Database

Firstly, the pipeline is created and then I created linked services and datasets in order to configure them with the pipeline.






1. Create the linked service.
2. Create input and output datasets.
3. Create a pipeline.

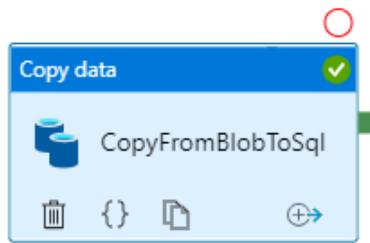
## From Activities toolbox -> Move & transform -> Select 'Copy Data'

The screenshot shows the Azure Data Factory (ADF) interface. On the left, the 'Factory Resources' pane lists 'Pipelines' (1) and 'Datasets' (2). The 'Activities' pane shows 'Copy data' selected under 'Move & transform'. The main canvas displays a 'Copy data' activity named 'CopyFromBlobToSql'. Below the canvas, the 'Output' tab shows the pipeline run ID: 72d5e7bc-b77b-44b0-91b3-d4dc178dfdbd. A table below shows the activity's execution details.

Name	Type	Run start	Duration	Status	Integr
CopyFromBlobToSql	Copy data	2021-02-12T19:27:48.420	00:00:06	✓ Succeeded	Defau

## Configure the source for the copy data





 Save as template    Validate    Validate copy runtime    Debug    Add trigger



General   **Source**   Sink   Mapping   Settings   User properties


Source dataset \*

 DelimitedText1   

 Open    New    Preview data   [Learn more](#) 

File path type

☒ File path in dataset   ☐ Prefix   ☐ Wildcard file path   ☐ List of files 

Filter by last modified 

Start time (UTC)

End time (UTC)

**Click on preview data to view data from the file stored in the BLOB container**

## Preview data

Linked service: AzureStorageLinkedService

Object:      emp.txt

FirstName	LastName
John	Doe
Jane	Doe

**Configure Sink in which the data needs to be loaded**

**Create a 'New Linked Service' to connect with the Azure SQL Database.**

## Set properties

Name

AzureSqlTable1

Linked service \*

AzureSqlDatabase1

Connect via integration runtime \* ⓘ

AutoResolveIntegrationRuntime

Table name

dbo.emp

☐ Edit

Import schema

☒ From connection/store ☐ None

▲ Advanced

[Open this dataset](#) for more advanced configuration with parameterization.

CopyPipeline

OutputSQLDataset

Activities

Search activities

Move & transform

Copy data

Data flow

Azure Data Explorer

Azure Function

Batch Service

Databricks

Data Lake Analytics

General

HDInsight

Iteration & conditionals

Machine Learning

Power Query

Save as template

Validate

Validate copy runtime

Debug

Add trigger

Copy data

CopyFromBlobToSql

General

Source

Sink

Mapping

Settings

User properties

Sink dataset \*

OutputSQLDataset

Open

New

Learn more

Stored procedure name

Select...

Refresh

Edit

Table option

None


Auto create table

Pre-copy script


## Validate the pipeline


Validate the pipeline by clicking on the 'Validate' button present at the top.


Pipeline validation output





**Your pipeline has been validated.**  
No errors were found.

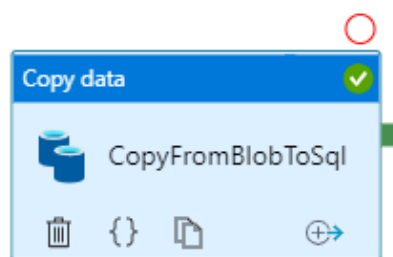
 Save as template

 Validate

 Validate copy runtime

 Debug

 Add trigger

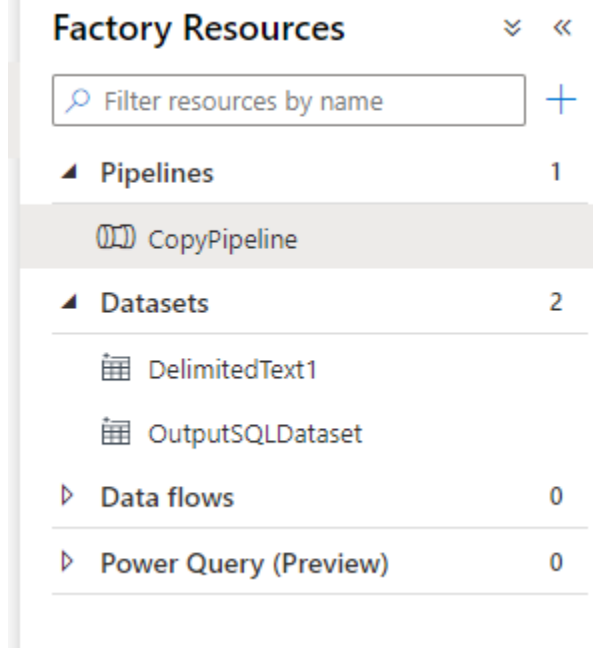


## Debug and publish the pipeline

Debug a pipeline before publishing artifacts (linked services, datasets, and pipeline) to Data Factory.

Triggered	Debug	Rerun	Cancel	Refresh	Edit columns	List	Gantt
<input type="text" value="Search by run ID or name"/>		Local time : Last 24 hours		Pipeline name : All		Status : All	
Showing 1 - 1 items							
<input type="checkbox"/>	Pipeline name	Run start ↑↓	Duration	Status ↑↓	Triggered by		
<input type="checkbox"/>	CopyPipeline	2/12/21, 2:27:46 PM	00:00:08	Succeeded	Manual trigger		

- Select Debug on the toolbar. The status of the pipeline run in the **Output** tab at the bottom of the window.
- Once the pipeline can run successfully, in the top toolbar, select **Publish all**. This action publishes entities (datasets, and pipelines) you created to Data Factory.



- Wait until you see the **Successfully published** message.



## Trigger the pipeline manually

1. Select **Trigger** on the toolbar -> select **Trigger Now** -> On the **Pipeline Run** page -> select **OK**.
2. Go to the **Monitor** tab on the left -> Check for the pipeline run that is triggered by a manual trigger.

**Pipeline runs**

Triggered Debug Rerun Cancel Refresh Edit columns List Gantt

Search by run ID or name Local time : Last 24 hours Pipeline name : All Status : All Runs : Latest runs

Add filter

Showing 1 - 1 items

Pipeline name	Run start ↑↓	Run end	Duration	Triggered by	Status
CopyPipeline	2/12/21, 2:33:26 PM	2/12/21, 2:33:35 PM	00:00:08	Manual trigger	✓ Succeeded

3. You can use links under the **PIPELINE NAME** column to view activity details and to rerun the pipeline.

### CopyPipeline

List Gantt

Rerun Rerun from activity Rerun from failed activity Refresh Edit pipeline

Copy data ✓

CopyFromBlobToSql

+ - 100% [ ]

**Activity runs**

Pipeline run ID [REDACTED]

All status ▾

Showing 1 - 1 of 1 items

Activity name	Activity type	Run start ↑↓	Duration	Status	Integration runtime
CopyFromBlobToSql	Copy data	2/12/21, 2:33:29 PM	00:00:05	✓ Succeeded	DefaultIntegrationRuntime (East US)

4. Verify that two more rows are added to the **emp** table in the database.

EmployeeDB (empsrver/EmployeeDB) | Query editor (preview) SQL database ×

Search (Ctrl+/) << Login + New Query ↑ Open query ♥ Feedback

Overview  
Activity log  
Tags  
Diagnose and solve problems  
Quick start  
Query editor (preview)  
ver Platform  
Power BI (preview)  
Power Apps (preview)  
Power Automate (preview)  
tings  
Configure  
Geo-Replication  
Connection strings  
Sync to other databases

EmployeeDB (siddhi) ↻

Showing limited object explorer here. For full capability please open SSDT.

Tables  
dbo.emp ...  
ID (int, not null)  
FirstName (varchar, null)  
LastName (varchar, null)  
Views  
Stored Procedures

Query 1 × Query 2 ×

Run ☐ Cancel query ↓ Save query ↓ Export data as ▾ ☒ Show only Editor

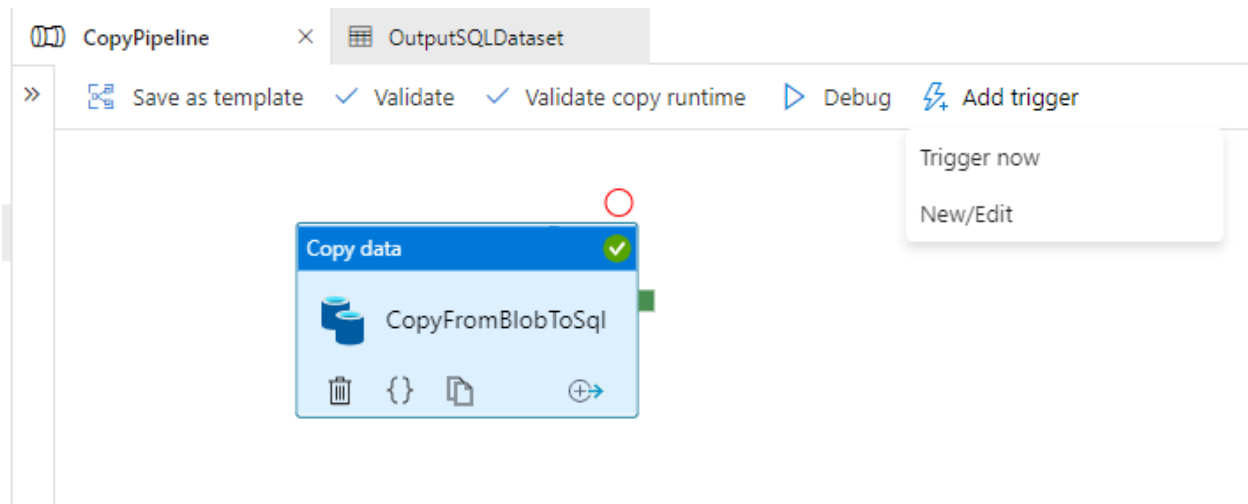
```
1 select * from dbo.emp;
```

Results Messages

ID	FirstName	LastName
1	John	Doe
2	Jane	Doe
3	John	Doe
4	Jane	Doe

Query succeeded | 2s

## Trigger the pipeline on a schedule



1. **Go to Author** tab on the left above the monitor tab ->

## New trigger

Name \*

RunEveryMinute

Description

Type \*

☒ Schedule ☐ Tumbling window ☐ Event

Start date \* ⓘ

02/12/2021 4:46 PM

Time zone \* ⓘ

Coordinated Universal Time (UTC)

Recurrence \* ⓘ

Every

2

Minute(s)


☒ Specify an end date

End On \* ⓘ


02/13/2021 9:49 PM

## Publishing the pipeline

### Publish all


You are about to publish all pending changes to the live environment. [Learn more](#) 


#### Pending changes (1)


NAME	CHANGE	EXISTING
▲ Triggers		
 RunEveryMinute	(New)	-


Publish

Cancel


 Dashboards


 Runs

 Pipeline runs


 Trigger runs

Runtimes & sessions

 Integration runtimes

 Data flow debug

Notifications

 Alerts & metrics


### Trigger runs


All

Schedule

Tumbling window

Event

 Refresh

 Edit columns



Local time : **Last 24 hours**

Trigger name : **All**

Status : **All**

Runs : Latest runs

Showing 1 - 1 items

Trigger name	Trigger type	Trigger time 	Status	Run	Pipelines
RunEveryMinute	ScheduleTrigger	2/12/21, 4:48:00 PM	 Succeeded	Original	1

## Trigger executing every minute

**Trigger runs**

All Schedule Tumbling window Event Refresh Edit columns

Local time : Last 24 hours Trigger name : All Status : All

Showing 1 - 6 items

Trigger name	Scheduled time	Trigger time	Status	Run	Pipeline
RunEveryMinute	2/12/21, 4:58:00 PM	2/12/21, 4:58:00 PM	✓ Succeeded	Original	1
RunEveryMinute	2/12/21, 4:56:00 PM	2/12/21, 4:56:01 PM	✓ Succeeded	Original	1
RunEveryMinute	2/12/21, 4:54:00 PM	2/12/21, 4:54:00 PM	✓ Succeeded	Original	1
RunEveryMinute	2/12/21, 4:52:00 PM	2/12/21, 4:52:00 PM	✓ Succeeded	Original	1
RunEveryMinute	2/12/21, 4:50:00 PM	2/12/21, 4:49:59 PM	✓ Succeeded	Original	1
RunEveryMinute	2/12/21, 4:48:00 PM	2/12/21, 4:48:00 PM	✓ Succeeded	Original	1

## Pipeline running every minute

**Pipeline runs**

Triggered Debug Rerun Cancel Refresh Edit columns List Gantt

Search by run ID or name Local time : Last 24 hours Pipeline name : All Status : All

Runs : Latest runs Add filter

Showing 1 - 7 items

<input type="checkbox"/> Pipeline name	Run start	Run end	Duration	Triggered by
<input type="checkbox"/> CopyPipeline	2/12/21, 4:58:01 PM	2/12/21, 4:58:09 PM	00:00:08	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 4:56:02 PM	2/12/21, 4:56:08 PM	00:00:06	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 4:54:01 PM	2/12/21, 4:54:08 PM	00:00:07	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 4:52:00 PM	2/12/21, 4:52:08 PM	00:00:08	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 4:50:00 PM	2/12/21, 4:50:15 PM	00:00:14	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 4:48:01 PM	2/12/21, 4:48:09 PM	00:00:07	RunEveryMinute
<input type="checkbox"/> CopyPipeline	2/12/21, 2:33:26 PM	2/12/21, 2:33:35 PM	00:00:08	Manual trigger

Employee Data is loaded in the Azure SQL Database.

**EmployeeDB (empsrver/EmployeeDB) | Query editor (preview)**

Search (Ctrl+/) Login New Query Open query Feedback

Overview Activity log Tags Diagnose and solve problems Quick start Query editor (preview)

Power Platform Power BI (preview) Power Apps (preview) Power Automate (preview) Settings

EmployeeDB (si... Showing limited object explorer here. For full capability please open SSDT.

Tables > dbo.emp Views > Stored Procedures

**Query 1**

Run Cancel query Save query Export data as Show only Editor

```
1 select * from dbo.emp
```

**Results** Messages

Search to filter items...

ID	FirstName	LastName
1	John	Doe

Results	Messages
2	Jane Doe
3	John Doe
4	Jane Doe

Results	Messages
5	John Doe
6	Jane Doe
7	John Doe

Query succeeded | 0s

Results	Messages
8	Jane Doe
9	John Doe
10	Jane Doe

Query succeeded | 0s

Results	Messages
11	John Doe
12	Jane Doe
13	John Doe

Query succeeded | 0s

Results	Messages
14	Jane Doe
15	John Doe
16	Jane Doe

Query succeeded | 0s