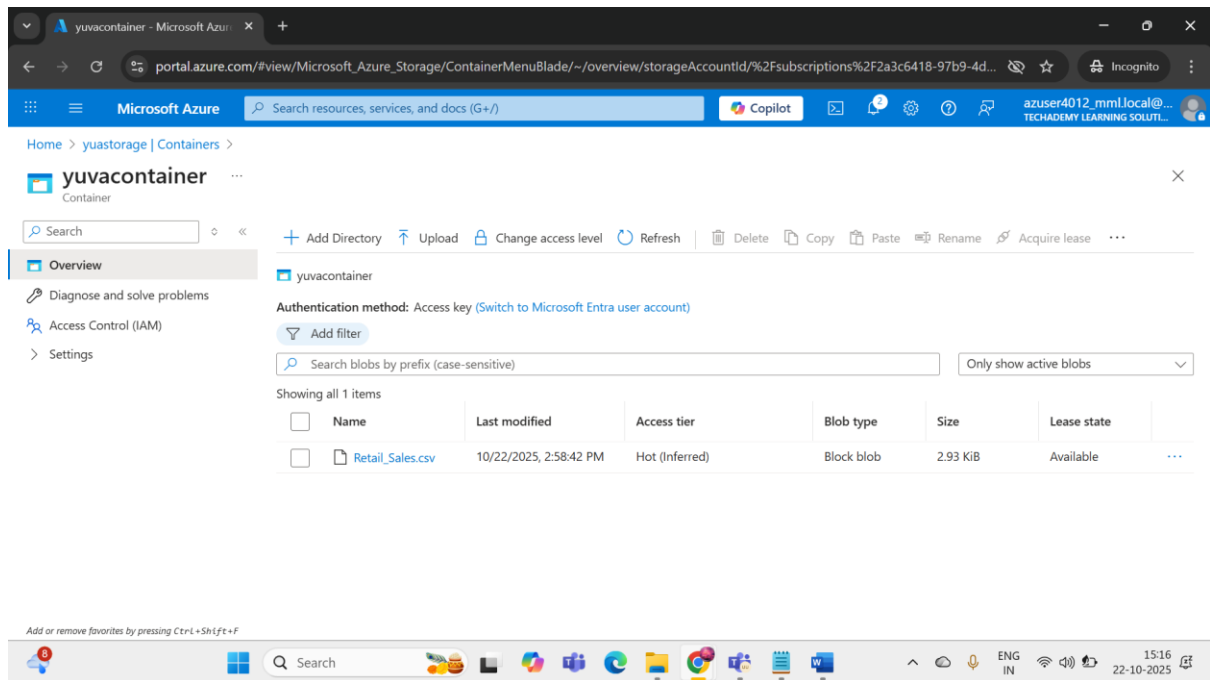


## Case Study

Azure → Snowflake with Snowpark, then Power BI Scenario You're the data engineer at ItTechGenie Retail. Sales teams drop monthly CSVs into an Azure Storage container. You must: upload the CSV to Azure, ingest it into Snowflake using Snowpark, model it into proper database/schema/table, and build a quick Power BI report for business users.

### Step 1: Upload CSV to Azure Blob Storage



### Step 2: Connect Azure to Snowflake (External Stage)

CREATE OR REPLACE DATABASE RETAIL\_DB;

USE DATABASE RETAIL\_DB;

CREATE SCHEMA SALES\_SCHEMA;

USE SCHEMA SALES\_SCHEMA;

CREATE OR REPLACE STAGE azure\_stage\_sales

URL='azure://yuastorage.blob.core.windows.net/yuvacontainer'

CREDENTIALS=(AZURE\_SAS\_TOKEN='sv=2024-11-04&ss=bfqt&srt=sc&sp=rwdlacupiytfx&se=2025-10-25T17:37:29Z&st=2025-10-22T09:22:29Z&spr=https&sig=hrEpzsXQ2buMqseVSEltq8QKC9%2FAs7LOM13sw5usrmw%3D');

LIST @azure\_stage\_sales;

The screenshot shows the Snowflake web interface. The top navigation bar includes the Snowflake logo and a search bar. The main workspace displays a SQL query editor with the following code:

```
1 CREATE OR REPLACE STAGE azure_stage_sales
2 URL='azure://yuastorage.blob.core.windows.net/yuvacontainer'
3 CREDENTIALS=(AZURE_SAS_TOKEN='sv=2024-11-04&ss=bfgt&srt=sc&sp=rwdLacuplytfx&se=2025-10-25T17:37:29Z&st=2025-10-22T09:22:29Z&spr=https&sig=hrEpsX02buMqseVSEItq8QKC9%2FAs7LOM13sw5usrmw%3D');
4
5 LIST @azure_stage_sales;
```

The query results are displayed in a table with the following columns: name, size, md5, and last\_modified. The table contains one row of data:

name	size	md5	last_modified
azure://yuastorage.blob.core.window	3004	a38163e0b0e75a2fc23a0509023fd	Wed, 22 Oct 2025 09:28:42 GMT

The interface also shows a sidebar with file explorer, a database explorer, and a query history section.

### Step 3: Ingest Data Using Snowpark (Python)

```
from snowflake.snowpark import Session
```

```
import pandas as pd
```

```
connection_parameters = {
    "account": "sb81969.me-central2.gcp",
    "user": "yuvasri310",
    "password": "Yuvasri3102004",
    "role": "SYSADMIN",
    "warehouse": "COMPUTE_WH",
    "database": "RETAIL_DB",
    "schema": "SALES_SCHEMA"
}
```

```
session = Session.builder.configs(connection_parameters).create()
```

```
df = pd.read_csv("Retail_Sales.csv")
```

```
snow_df = session.create_dataframe(df)
```

```
snow_df.write.save_as_table("retail_sales", mode="overwrite")

print("Retail_Sales table created in Snowflake!")
```

The screenshot shows a Databricks workspace with a Python notebook. The code in the cell reads a CSV file from a URL and displays the first five rows of the data. The URL is a SAS file stored in Azure Blob Storage. The data preview shows columns: OrderID, OrderDate, MonthOfSale, Discount, Sales, and Profit.

```
url = (
    "https://yuastorage.blob.core.windows.net/yuvacontainer/Retail_Sales.csv?sp=r&st=2025-10-22T10:56:10Z&se=2025-10-22T19:11:10Z&spr=https&sv=2024-11-04&sr=b&sig=YTWNPSYK0t6LuzXT0r2DmWDEVeW8dXUaP2g7uIOYY%3D"
)

df = pd.read_csv(url)
print(df.head())
```

	OrderID	OrderDate	MonthOfSale	Discount	Sales	Profit
0	ORD-5F8D6F8C	2024-10-08	2024-10	0.00	2700.0	780.43
1	ORD-BF0078E4	2024-08-11	2024-08	0.15	27200.0	4135.60
2	ORD-86CD58A3	2024-06-12	2024-06	0.10	31500.0	5676.96
3	ORD-FB0CD2D9	2024-12-18	2024-12	0.00	36000.0	11783.22
4	ORD-EF35596B	2024-10-27	2024-10	0.00	24000.0	4189.98

The screenshot shows a Databricks workspace with a Python notebook. The code in the cell establishes a connection to a Snowflake database, writes the data from the previous cell to a table named 'RETAIL\_SALES', and prints a success message. The connection details are provided in a dictionary.

```
conn = snowflake.connector.connect(
    user="yuvasri310",
    password="Yuvasri3102004",
    account="sb81969.me-central12.gcp",
    warehouse="COMPUTE_WH",
    database="RETAIL_DB",
    schema="SALES_SCHEMA"
)

success, nchunks, nrows, _ = write_pandas(conn, df, "RETAIL_SALES")

print(f"Uploaded {nrows} rows successfully!")
conn.close()
```

Uploaded 25 rows successfully!

## Step 4: Create Database, Schema & Model Tables in Snowflake

CREATE OR REPLACE TABLE retail\_sales (

OrderID STRING,

OrderDate STRING,

MonthOfSale STRING,

CustomerID STRING,

CustomerName STRING,

Country STRING,

Region STRING,

City STRING,

Category STRING,

Subcategory STRING,

Quantity INT,

Discount FLOAT,

Sales FLOAT,

Profit FLOAT

);

SELECT \* FROM retail\_sales;

The screenshot shows the Snowflake web interface. The query editor contains the following SQL code:

```
SELECT * FROM retail_sales;
```

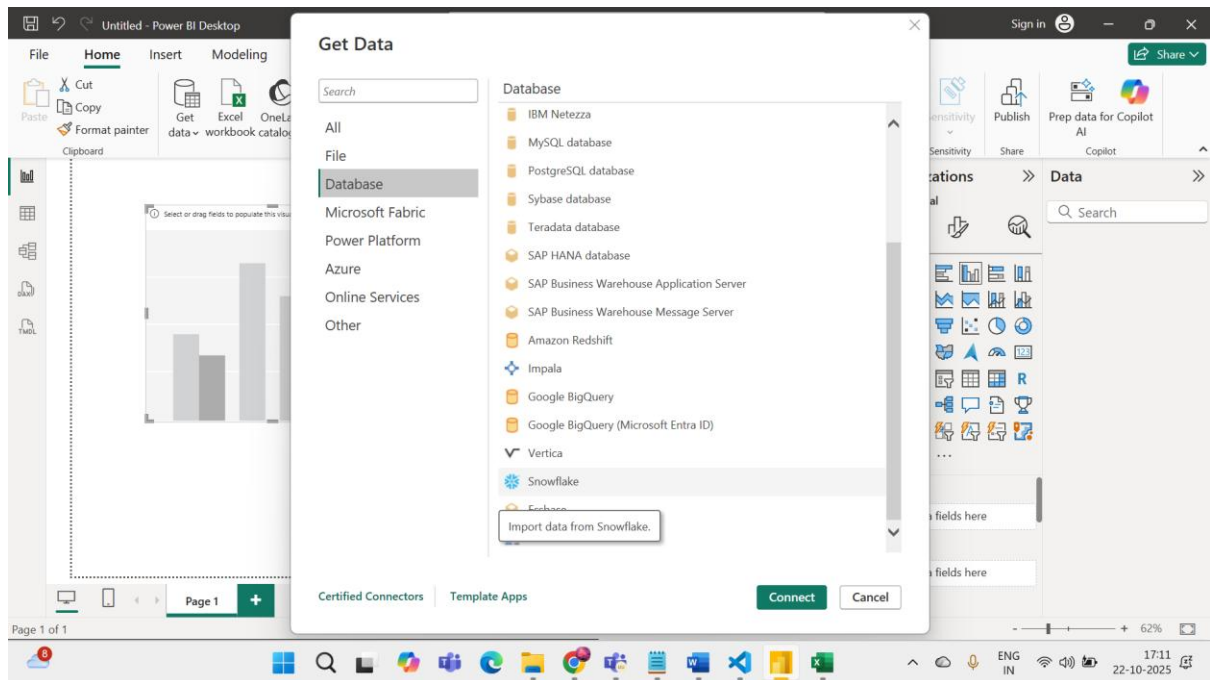
The results pane displays a table with 25 rows. The columns are: ORDERID, ORDERDATE, MONTHOFSALE, CUSTOMERID, CUSTOMERNAME, COUNTRY, REGION, and a partially visible column. The first two rows are visible:

ORDERID	ORDERDATE	MONTHOFSALE	CUSTOMERID	CUSTOMERNAME	COUNTRY	REGION	
ORD-BF0078E4	2024-08-11	2024-08	CUST1001	Aarav Iyer	India	Central	Luc
ORD-86CD58A3	2024-06-12	2024-06	CUST1002	Arjun Sharma	USA	East	Kol

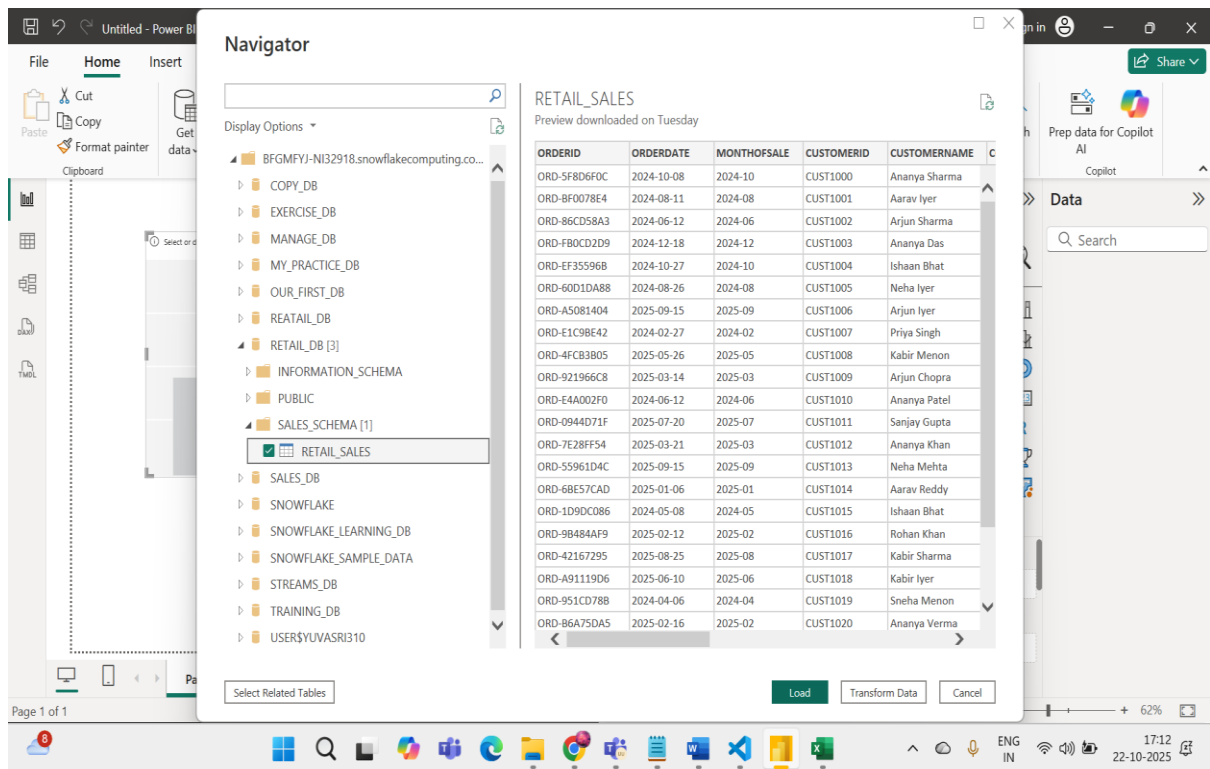
The Query History pane shows the following queries:

- just now 1.1s SELECT \* FROM retail\_sales; 01bfe10d-0000-c0e6-0001-17b600055032
- 7 minutes ago 194ms CREATE OR REPLACE TABLE retail\_sales ( OrderID STRING, OrderDate ST... 01bfe106-0000-c08d-0001-17b600054196

## Step 5: Connect Power BI to Snowflake



## Choose the database



**Untitled - Power BI Desktop**

File Home View Help **Table tools** **Column tools**

Name: ORDERID Format: Text Summarization: Don't summarize Data category: Uncategorized Sort by column: Sort Groups: Manage relationships New column: Calculations

ORDERID	ORDERDATE	MONTHOFSALE	CUSTOMERID	CUSTOMERNAME	COUNTRY	REGION	CITY	CATEGORY
ORD-SF8D6FOC	2024-10-08	2024-10	CUST1000	Ananya Sharma	India	South	Mumbai	Office Supp
ORD-BF0078E4	2024-08-11	2024-08	CUST1001	Aarav Iyer	India	Central	Lucknow	Technology
ORD-86CD58A3	2024-06-12	2024-06	CUST1002	Arjun Sharma	USA	East	Kolkata	Furniture
ORD-FB0CD2D9	2024-12-18	2024-12	CUST1003	Ananya Das	India	North	Kolkata	Office Supp
ORD-EF35596B	2024-10-27	2024-10	CUST1004	Ishaan Bhat	UK	Central	Chennai	Furniture
ORD-60D1DA88	2024-08-26	2024-08	CUST1005	Neha Iyer	UAE	West	Chennai	Furniture
ORD-A5081404	2025-09-15	2025-09	CUST1006	Arjun Singh	India	Central	Jaipur	Office Supp
ORD-E1C9BE42	2024-02-27	2024-02	CUST1007	Priya Singh	India	North	Lucknow	Furniture
ORD-4FCB3B05	2025-05-26	2025-05	CUST1008	Kabir Menon	India	West	Jaipur	Technology
ORD-921966C8	2025-03-14	2025-03	CUST1009	Arjun Chopra	UAE	West	Mumbai	Furniture
ORD-E4A002F0	2024-06-12	2024-06	CUST1010	Ananya Patel	UK	West	Ahmedabad	Office Supp
ORD-0944D71F	2025-07-20	2025-07	CUST1011	Sanjay Gupta	UAE	North	Pune	Office Supp
ORD-7E28FF54	2025-03-21	2025-03	CUST1012	Ananya Khan	India	North	Delhi	Office Supp
ORD-55961D4C	2025-09-15	2025-09	CUST1013	Neha Mehta	India	East	Ahmedabad	Technology
ORD-6BE57CAD	2025-01-06	2025-01	CUST1014	Aarav Reddy	UAE	East	Ahmedabad	Technology
ORD-I9D8DC086	2024-05-08	2024-05	CUST1015	Ishaan Bhat	India	West	Jaipur	Office Supp
ORD-9B484AF9	2025-02-12	2025-02	CUST1016	Rohan Khan	Singapore	East	Delhi	Furniture
ORD-42167295	2025-08-25	2025-08	CUST1017	Kabir Sharma	India	Central	Mumbai	Technology
ORD-A91119D6	2025-06-10	2025-06	CUST1018	Kabir Iyer	India	West	Ahmedabad	Technology
ORD-951CD788	2024-04-06	2024-04	CUST1019	Sneha Menon	India	West	Mumbai	Furniture

Data pane: RETAIL\_SALES > CATEGORY > CITY > COUNTRY > CUSTOMERID > CUSTOMERNAME > DISCOUNT > MONTHOFSALE > ORDERDATE > **ORDERID** > PROFIT > QUANTITY > REGION > SALES > SUBCATEGORY

Table: RETAIL\_SALES (25 rows) Column: ORDERID (25 distinct values)

## Visual Representation:

