Goal 1 Ingestion

Tuesday, February 3, 2026 8:47 PM

Ingest Data from Two Sources 2 hours

In this part of the hackathon, you'll pull data from **two different sources** into your Fabric workspace. Pick any sources you want — the goal is just to get data moving.

You can ingest your data using **whatever method you prefer**, including:

- Pipelines
- Dataflows
- Notebooks
- Shortcuts
- …or any other ingestion feature in Fabric

The end result:
You'll load both datasets into your **Fabric Lakehouse**, where you'll transform them in the next section.
Let me know if you want this turned into a slide version, expanded into step-by-step tasks, or styled to match the earlier section!

Data Sources:

Student Information System - Azure SQL DB
Connection Strings:

Medical research - JPGs from

Notes:

Microsoft Fabric – Data Ingestion Overview

Microsoft Fabric lets you ingest and orchestrate data from **multiple sources** (databases, files, services) into your Lakehouse or SQL Warehouse using **pipelines, dataflows (Gen2), Spark notebooks, shortcuts, and more**.

• Key Tools & Methods

- **Pipelines** — Managed workflows (ETL/ELT) that connect data sources and perform copy jobs, scheduled runs, and monitoring.
- **Dataflows (Gen2)** — Low-code data ingestion + transformation using Power Query (M language).
- **Notebooks** — Code-centric ingestion & transformation using Spark (e.g., Python/SQL).
- **Shortcuts** — Link external data without copying files; read source data directly from OneLake.

Ingest JPG/Files into Lakehouse

Description (short):
Fabric Dataflows or Notebooks let you ingest file formats (like **JPG, PNG, CSV**) directly from storage (Azure Blob, ADLS Gen2) into your Lakehouse.

- In **Dataflows Gen2**, you point to the file source & define transformations or cleansing steps visually.
- In **Notebooks**, you can use Spark code to read images and save them into OneLake with custom logic (e.g., image metadata extraction).

(Note: JPG ingestion is typically via file containers and then stored as blob/file entries in the lakehouse. Fabric doesn't auto-parse image content by default — use notebooks to process content.)

Reference Links:
Microsoft Learn: *Ingest Data with Dataflows Gen2 in Microsoft Fabric* (module).
https://learn.microsoft.com/en-us/training/modules/use-dataflow-gen-2-fabric/

Microsoft Learn: *How to use Microsoft Fabric notebooks* — includes reading files (jpg, png) from lake storage.
https://learn.microsoft.com/en-us/fabric/data-engineering/how-to-use-notebook

Topic	Link
Microsoft Fabric ingestion learning path	https://learn.microsoft.com/en-us/training/paths/ingest-data-with-microsoft-fabric/ (Microsoft Learn)
Dataflows with Visual ETL	https://learn.microsoft.com/en-us/training/modules/use-dataflow-gen-2-fabric/ (Microsoft Learn)
Decision guide: pipelines vs dataflows vs spark	https://learn.microsoft.com/en-us/fabric/fundamentals/decision-guide-pipeline-dataflow-spark (Microsoft Learn)

Mirroring (Near-Real-Time Ingestion) from Azure SQL DB

Mirroring in **Microsoft Fabric** continuously replicates data from **Azure SQL Database** into **OneLake** with **near real-time change tracking**.
No pipelines, no code, no copy jobs — Fabric keeps the Lakehouse automatically in sync with the source database.

This is ideal when:

- Source is Azure SQL DB
- You want **live sync** instead of scheduled loads
- You want tables instantly available in the Lakehouse for transformation

Why this is perfect for the hackathon

Set up takes ~10 minutes. Data starts appearing automatically in your Lakehouse.

Microsoft Learn References

- Mirroring overview in Fabric:
  https://learn.microsoft.com/fabric/database/mirroring/overview
- Tutorial – Mirror Azure SQL DB into Fabric:
  https://learn.microsoft.com/fabric/database/mirroring/azure-sql-database

Azure SQL Mirror DB Steps

1. Create New Item -> Search for Mirrored Azure SQL Database-> Select Mirrored Azure SQL Database
2. Select Azure SQL DB
3. Enter DB Server/DB Name
4. Select Connect
5. Select Connect
6. Select Create Mirrored database