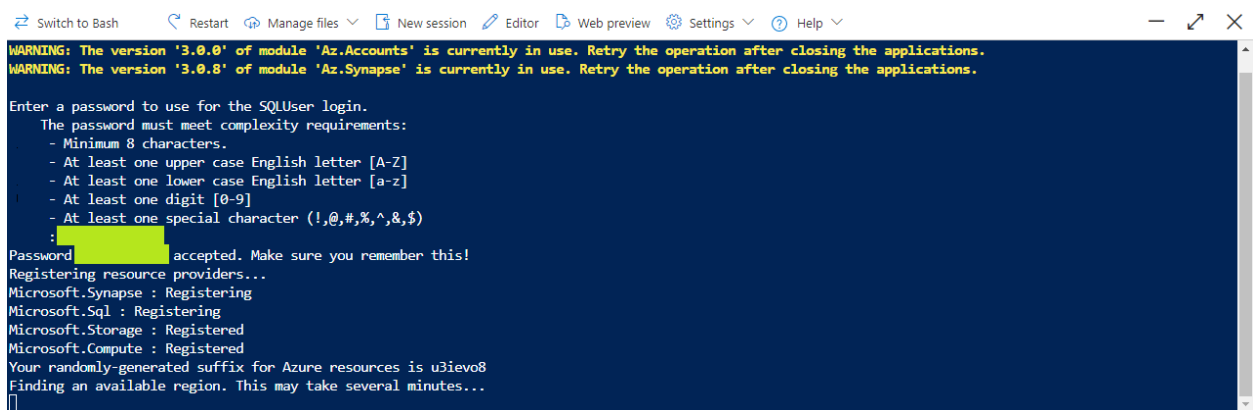


# Azure Synapse Exercise and Discussion

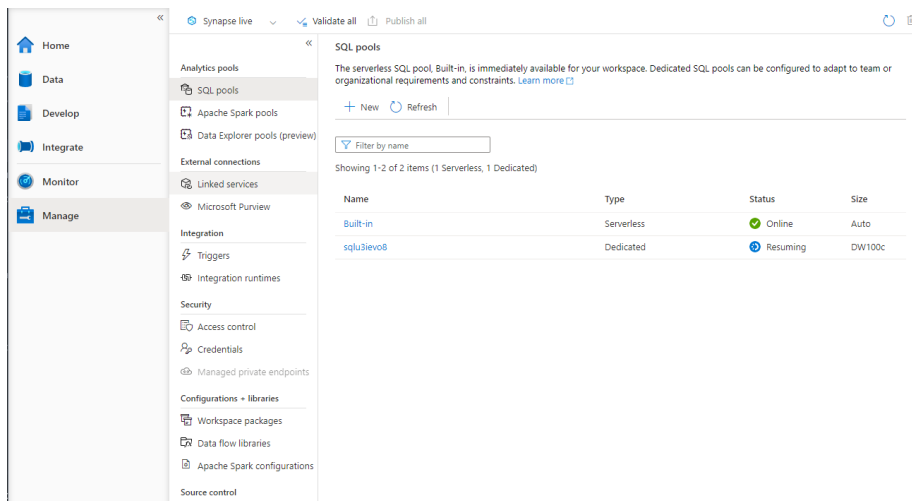
For this project, an example pipeline was built using some of the functionality provided in Azure Data Factory. This brief document highlights some of the results seen during implementation (a full outline of the steps taken is also available in this repository).

An Azure Synapse Analytics workspace was provisioned as outlined in the instructional material:



```
Switch to Bash Restart Manage files New session Editor Web preview Settings Help
WARNING: The version '3.0.0' of module 'Az.Accounts' is currently in use. Retry the operation after closing the applications.
WARNING: The version '3.0.8' of module 'Az.Synapse' is currently in use. Retry the operation after closing the applications.

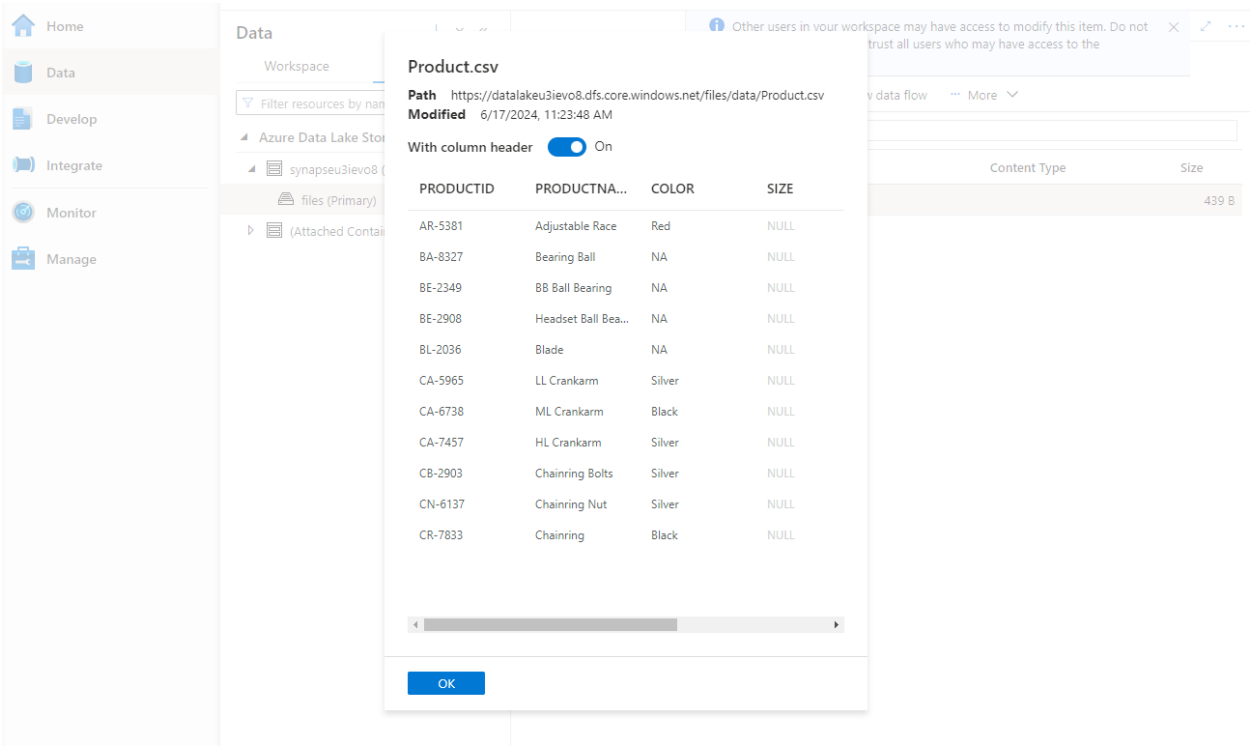
Enter a password to use for the SQLUser login.
The password must meet complexity requirements:
- Minimum 8 characters.
- At least one upper case English letter [A-Z]
- At least one lower case English letter [a-z]
- At least one digit [0-9]
- At least one special character (!, @, #, %, ^, &, $)
:
Password accepted. Make sure you remember this!
Registering resource providers...
Microsoft.Synapse : Registering
Microsoft.Sql : Registering
Microsoft.Storage : Registered
Microsoft.Compute : Registered
Your randomly-generated suffix for Azure resources is u3ievo8
Finding an available region. This may take several minutes...
[]
```



The screenshot shows the Azure Synapse Studio interface. On the left is a navigation pane with options: Home, Data, Develop, Integrate, Monitor, and Manage. The 'Manage' option is selected, and the 'SQL pools' section is expanded. The main area displays the 'SQL pools' configuration page. It includes a 'Filter by name' search bar and a table showing the status of SQL pools. The table has columns for Name, Type, Status, and Size. Two pools are listed: 'Built-in' (Serverless, Online, Auto) and 'sqlu3ievo8' (Dedicated, Resuming, DW100c).

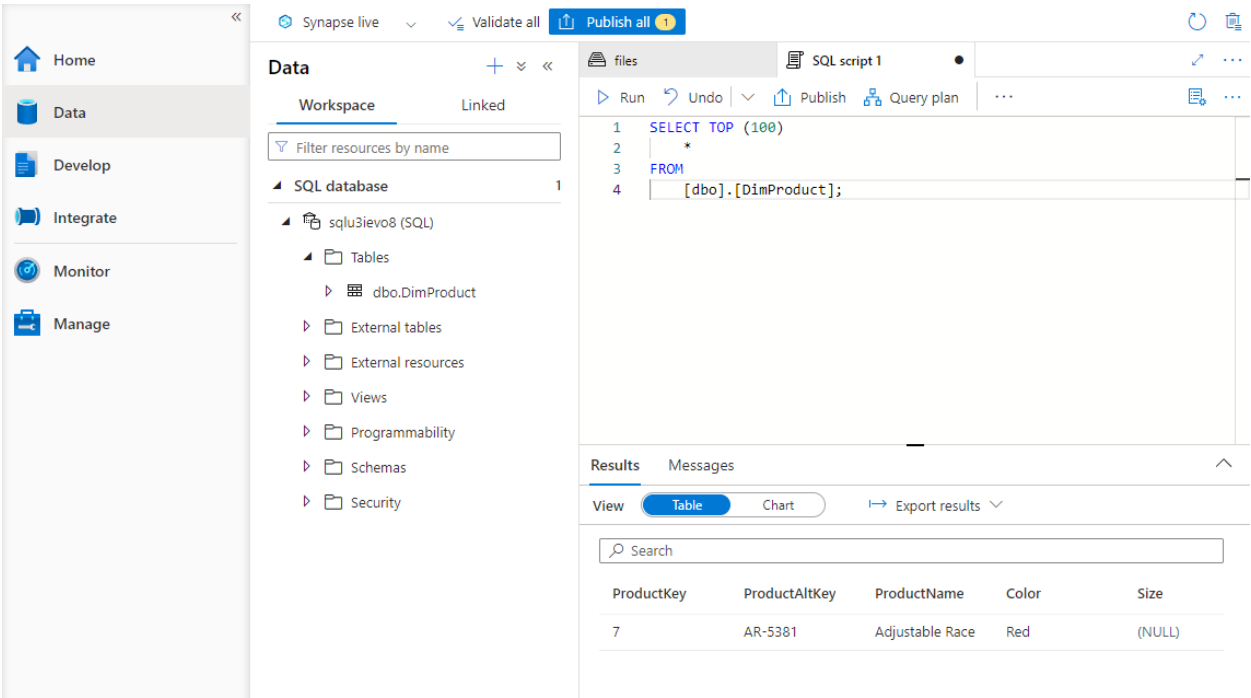
Name	Type	Status	Size
Built-in	Serverless	Online	Auto
sqlu3ievo8	Dedicated	Resuming	DW100c

This process creates a sample flat file in .csv format along with a destination table with several matching columns:



The screenshot shows the Synapse Data Explorer interface. On the left is a navigation pane with 'Home', 'Data', 'Develop', 'Integrate', 'Monitor', and 'Manage'. The main area displays a 'Workspace' view of 'Azure Data Lake Storage' for 'synapseu3ievo8'. A file named 'Product.csv' is selected, and a preview window is open. The preview shows the file's path, modification date, and a table of data with columns: PRODUCTID, PRODUCTNAME, COLOR, and SIZE. The 'With column header' toggle is turned on. An 'OK' button is at the bottom of the preview window.

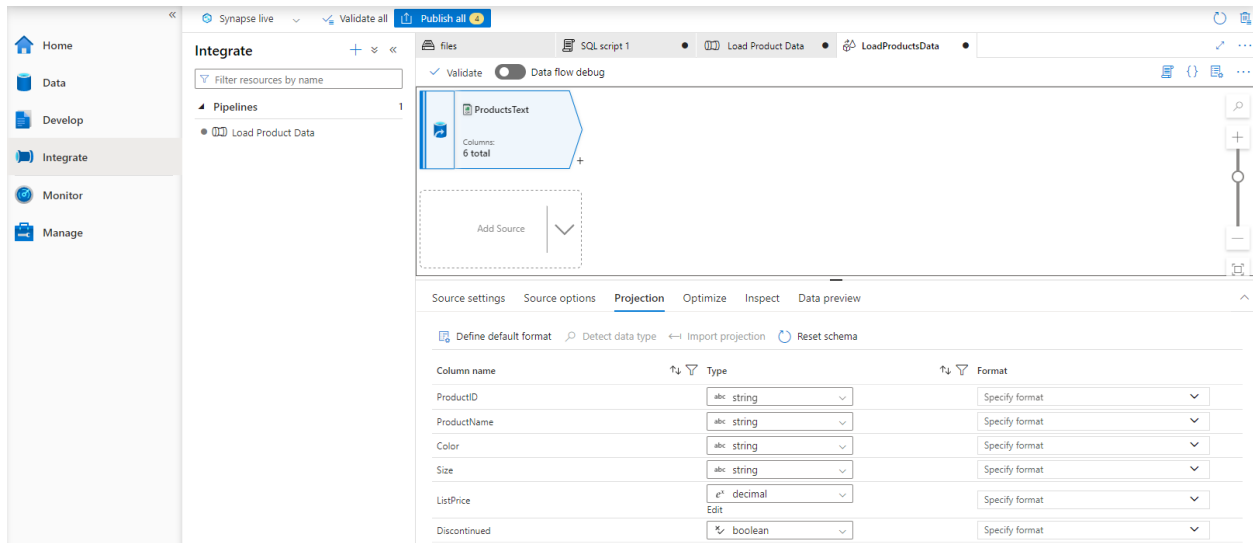
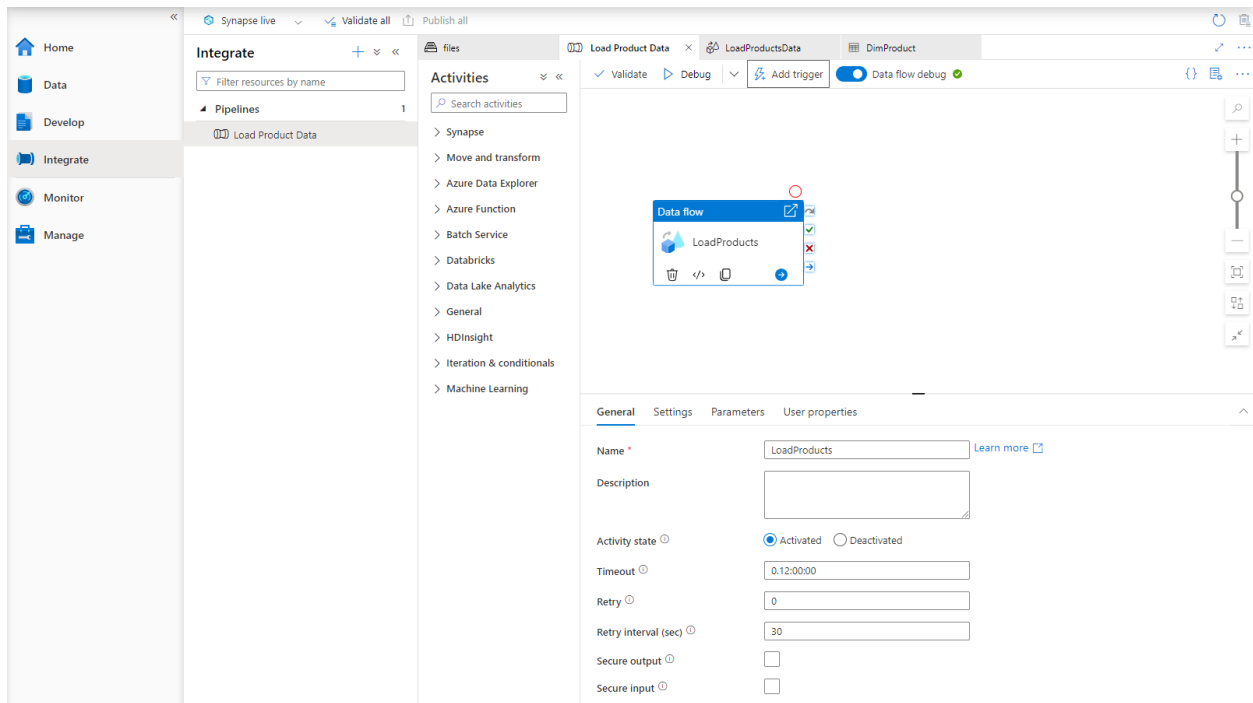
PRODUCTID	PRODUCTNAME	COLOR	SIZE
AR-5381	Adjustable Race	Red	NULL
BA-8327	Bearing Ball	NA	NULL
BE-2349	BB Ball Bearing	NA	NULL
BE-2908	Headset Ball Bea...	NA	NULL
BL-2036	Blade	NA	NULL
CA-5965	LL Crankarm	Silver	NULL
CA-6738	ML Crankarm	Black	NULL
CA-7457	HL Crankarm	Silver	NULL
CB-2903	Chainring Bolts	Silver	NULL
CN-6137	Chainring Nut	Silver	NULL
CR-7833	Chainring	Black	NULL



The screenshot shows the Synapse SQL Editor interface. On the left is a navigation pane with 'Home', 'Data', 'Develop', 'Integrate', 'Monitor', and 'Manage'. The main area displays a 'Workspace' view of 'SQL database' for 'sqlu3ievo8 (SQL)'. A table named 'dbo.DimProduct' is selected. The 'SQL script 1' tab is active, showing a query: `SELECT TOP (100) * FROM [dbo].[DimProduct];`. The 'Results' pane at the bottom shows the query output in a table view.

ProductKey	ProductAltKey	ProductName	Color	Size
7	AR-5381	Adjustable Race	Red	(NULL)

A data flow and initial pipeline was then created. We begin by first establishing a node for the flat file to be ingested and verifying its data types:



We similarly add a node for the destination table, then merge the two datasets via an upsert job. The results are then landed in this destination table:

The image displays two screenshots of the Synapse live interface, showing the 'Integrate' and 'Develop' tabs.

**Integrate Tab:** The 'Integrate' tab shows a pipeline named 'Load Product Data'. The pipeline consists of the following steps:

- ProductsText** (Source): Products text data
- MatchedProducts** (Join): Matched product data
- SetLoadAction** (Action): Insert new, 'upsert' existing
- DimProductTable** (Sink): Columns: 6 total

The 'Mapping' tab is selected, showing the following options:

- ☒ Skip duplicate input columns
- ☒ Skip duplicate output columns
- ☐ Auto mapping
- ☐ Reset
- ☐ Add mapping
- ☐ Delete
- ☐ Output format: 5 mappings: 1 column(s) from the output schema left unmapped

The 'Mapping' tab also shows the following columns:

Input columns	Output columns
abc: ProductID	abc: ProductAltKey
abc: ProductsText@ProductName	abc: ProductName
abc: ProductsText@Color	abc: Color
abc: ProductsText@Size	abc: Size
e <sup>4</sup> : ProductsText@ListPrice	e <sup>4</sup> : ListPrice
ProductsText@Discontinued	Discontinued

**Develop Tab:** The 'Develop' tab shows the same pipeline. The 'Data preview' tab is selected, showing the following data:

ProductAltKey	ProductName	Color	Size	ListPrice	Discontinued
AR-5381	Adjustable...	Red	NULL	2.0000	✓
BA-8327	Bearina Ball	NA	NULL	1.0000	×
BE-2349	BB Ball Be...	NA	NULL	1.0000	×
BE-2908	Headset B...	NA	NULL	0.0000	×
BL-2036	Blade	NA	NULL	2.0000	×
CA-5965	LL Crankarm	Silver	NULL	8.0000	×
CA-6738	ML Cranka...	Black	NULL	8.0000	×
CA-7457	HL Crankar...	Silver	NULL	8.0000	×
CB-2903	Chainring ...	Silver	NULL	2.0000	×
CN-6137	Chainring ...	Silver	NULL	3.0000	×
CR-7833	Chainring	Black	NULL	2.0000	×

For this test, the pipeline is executed via a manual trigger, though this can be scheduled by other means. We can verify that the landing table has been updated with data from the flat file.

Home

Data

Develop

Integrate

Monitor

Manage

Analytics pools

SQL pools

Apache Spark pools

Data Explorer pools (preview)

Activities

SQL requests

KQL requests

Apache Spark applications

Data flow debug

Integration

Pipeline runs

Trigger runs

Integration runtimes

Link connections

Pipeline runs

Triggered Debug Rerun Cancel options Refresh Edit columns List Gantt

Filter by run ID or name Local time : Last 24 hours Pipeline name : All Status : All Runs : Latest runs Copy filters Export to CSV

Triggered by : All Add filter

Showing 1 - 3 items Last refreshed 0 minutes ago

Pipeline name	Run start	Run end	Duration	Triggered by	Status	Run	Pa
Load Product Data	6/17/2024, 2:06:10 PM	--	3s	Manual trigger	In progress	Original	
Load Product Data	6/17/2024, 2:03:35 PM	6/17/2024, 2:05:34 PM	2m 0s	Manual trigger	Canceled	Original	
Load Product Data	6/17/2024, 2:02:17 PM	6/17/2024, 2:05:36 PM	3m 19s	Manual trigger	Canceled	Original	

Home

Data

Develop

Integrate

Monitor

Manage

Synapse live Validate all Publish all

Data

Workspace Linked

Filter resources by name

SQL database 1

sqlu3ievo8 (SQL)

Tables

dbo.DimProduct

Columns

External tables

External resources

Views

Programmability

Schemas

Security

files

Load Product Data LoadProductsData DimProduct

Run Undo Publish Query plan Connect to sqlu3ievo8

1 SELECT TOP (100) [ProductKey]

2 , [ProductAltKey]

3 , [ProductName]

4 , [Color]

5 , [Size]

6 , [ListPrice]

7 , [Discontinued]

8 FROM [dbo].[DimProduct]

Results Messages

View Table Chart Export results

Search

ProductKey	ProductAltKey	ProductName	Color	Size	ListPrice	Discontinued
5	CA-7457	HL Crankarm	Silver	(NULL)	8.0000	False
7	AR-5381	Adjustable Race	Red	(NULL)	2.0000	True
10	CB-2903	Chainring Bolts	Silver	(NULL)	2.0000	False
11	BA-8327	Bearing Ball	NA	(NULL)	1.0000	False
13	CN-6137	Chainring Nut	Silver	(NULL)	3.0000	False
22	CR-7833	Chainring	Black	(NULL)	2.0000	False
32	CA-6738	HL Crankarm	Black	(NULL)	8.0000	False

00:00:01 Query executed successfully.