

Azure Synapse Workspace

Hands on Lab Guide

Lab environment setup

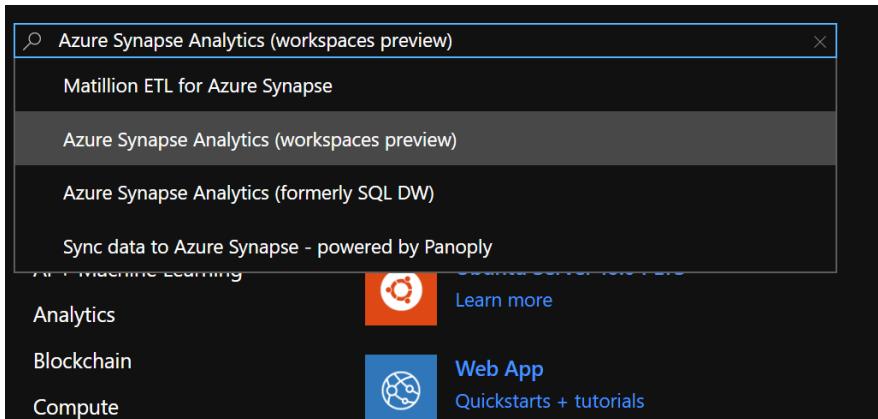
Requirements

1. Microsoft Azure subscription (non-Microsoft subscriptions must be paid subscriptions).
2. **IMPORTANT:** To complete the OAuth 2.0 access component of this lab, you must have permission to register an app and create a service principal within Azure Active Directory within your Azure subscription.

Before practice

Task 1: Azure Synapse Workspace Creation

1. Connect Azure Portal and click create resource and type Azure Synapse Workspace.



2. Choose Southeast Asia for region.

Create Synapse workspace

* Basics * Security + networking Tags Summary

Create a Synapse workspace to develop an enterprise analytics solution in just a few clicks.

Project details

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

Subscription * ⓘ Microsoft Azure Internal Consumption

Resource group * ⓘ (New) Kyowon_HOL Create new

Workspace details

Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.

Workspace name * kyowonsynapse

Region * Southeast Asia

Select Data Lake Storage Gen2 * ⓘ

From subscription Manually via URL

Account name * ⓘ Create new

File system name * ⓘ Create new

3. To create Data Lake Storage Gen2, select 'Create New'.
4. Put file system name and tick check box as below.

Subscription * ⓘ Microsoft Azure Internal Consumption

Resource group * ⓘ (New) Kyowon_HOL Create new

Workspace details

Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.

Workspace name * kyowonsynapse ✓

Region * Southeast Asia

Select Data Lake Storage Gen2 * ⓘ From subscription Manually via URL

Account name * ⓘ (New) kyowonadlsgen2 Create new

File system name * ⓘ (New) kyowonsynapse Create new

Assign myself the Storage Blob Data Contributor role on the Data Lake Storage Gen2 account 'kyowonadlsgen2'.

ⓘ We will automatically grant the workspace identity data access to the specified Data Lake Storage Gen2 account, using the Storage Blob Data Contributor role. To enable other users to use this storage account after you create your workspace, you must grant them the Storage Blob Data Contributor role.

5. Put admin account information in Security + Networking tab.

* Basics * **Security + networking** Tags Summary

Configure security options and networking settings for your workspace.

SQL administrator credentials

Provide credentials that can be used for administrator access to the workspace's SQL pools. If you don't provide a password, one will be automatically generated. You can change the password later.

Admin username * sqladminuser

Password ✓

Confirm password ✓ Pass

Workspace managed identity

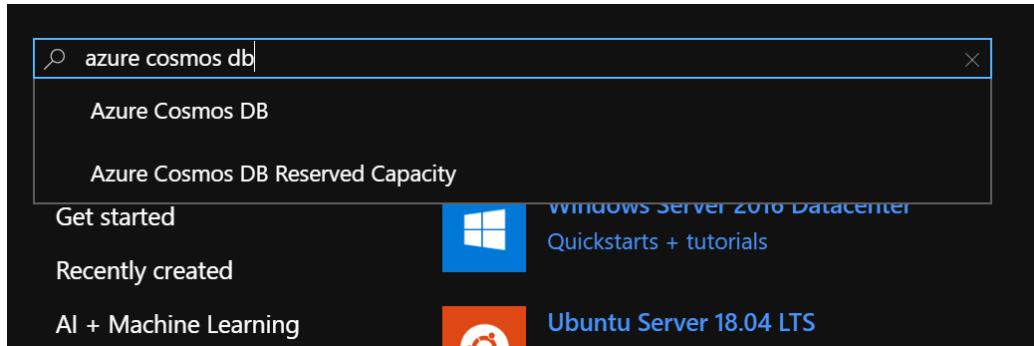
Choose whether you'd like your workspace's managed identity to be automatically granted CONTROL permission (inherits all permissions). This applies when using SQL pools and SQL on-demand within the current workspace.

Grant CONTROL to the workspace's managed identity on all SQL pools and SQL on-demand. ⓘ

6. Review + Create.

Task 2 : Azure Cosmos DB creation

1. Create Resource and type Azure Cosmos DB.



2. Select Azure Cosmos DB.
3. Select Core(SQL) for API, choose Southeast Asia for Region.

Project Details

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

Subscription * Microsoft Azure Internal Consumption

Resource Group * Kyowon_HOL

Instance Details

Account Name * kyowoncosmos

API * Core (SQL)

Notebooks (Preview) On

Location * (Asia Pacific) Korea Central

With Azure Cosmos DB free tier, you will get 400 RU/s and 5 GB of storage for free in an account. You can enable free tier on up to one account per subscription. Estimated \$24/month discount per account.

Apply Free Tier Discount

Account Type Production

Geo-Redundancy Enable

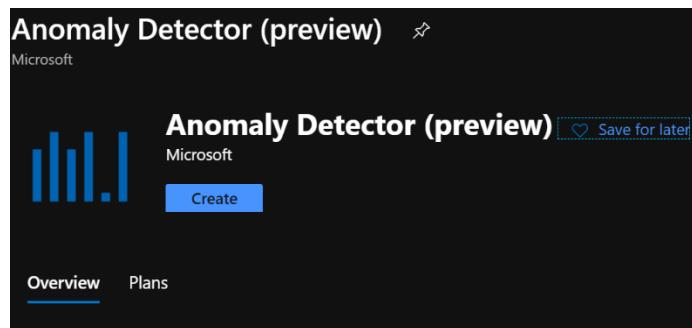
Multi-region Writes Enable

*Up to 33% off multi-region writes is available to qualifying new accounts only. Offer limited to accounts with both account locations and geo-redundancy, and applies only to multi-region writes in those same regions. Both Geo-Redundancy and Multi-region Writes must be enabled in account settings. Actual discount will vary based on number of qualifying regions selected.

4. Review + Create.

Task 3 : Anomaly Detector (Cognitive Service) Creation

1. Create Resource and type Anomaly Detector.



2. Put proper information as below and create.

Name *kyowonanomaly

Subscription *Microsoft Azure Internal Consumption

Location *(US) East US

Pricing tier (View full pricing details) *Free F0 (10 Calls per second, 20K Transactions per month)

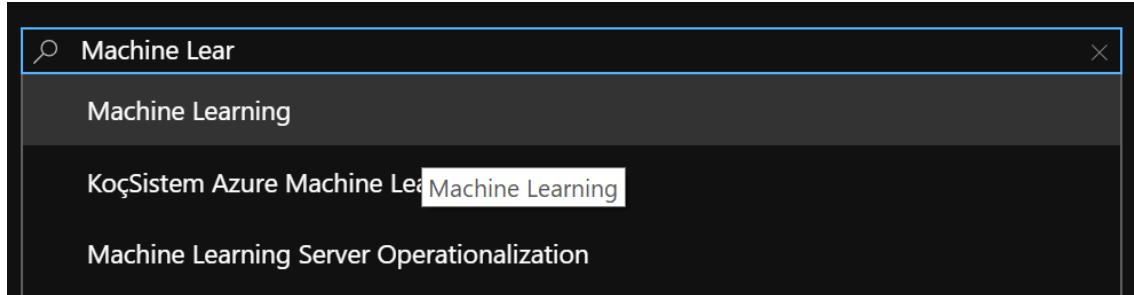
Resource group *Kyowon_HOL
[Create new](#)

I confirm I have read and understood the notice below. *

Previews are made available to you on the condition that you agree to the [Supplemental Terms of Use for](#)

Task 4 : Azure Machine Learning Workspace Creation

1. Create Resource and type Machine Learning.



2. Put proper information as below and create.

Machine Learning
Create a machine learning workspace

Basics Tags Review + create

Project details

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

Subscription * ⓘ Microsoft Azure Internal Consumption

Resource group * ⓘ Kyowon_HOL
Create new

Workspace details

Specify the name, region, and edition for the workspace.

Workspace name * ⓘ kyowonml

Region * ⓘ Southeast Asia

Workspace edition * ⓘ Basic

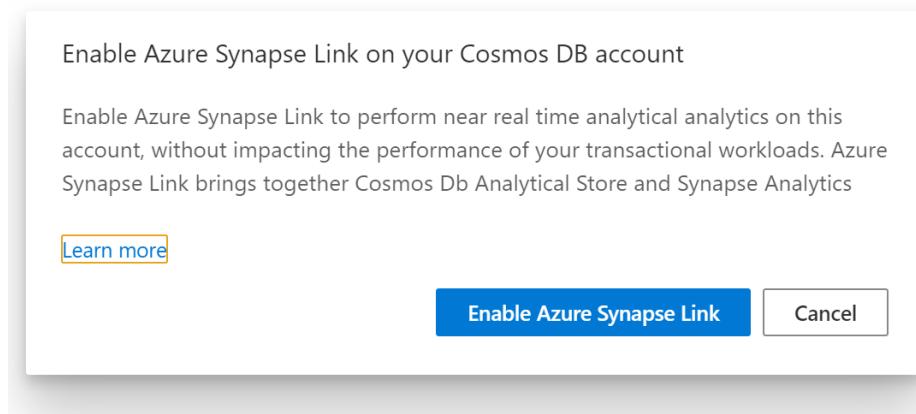
Note: For your convenience, these resources are added automatically to the workspace, if regionally available: Azure Storage, Azure Application Insights, Azure Key Vault

Lab Guide

Task 1: Azure Cosmos DB Enable Synapse Link

1. In [Azure Portal](#), goes to Azure Cosmos DB and select Data Explorer.
2. click Enable Azure Synapse Link(Preview).

The screenshot shows the Azure Cosmos DB Data Explorer interface. At the top, there's a breadcrumb navigation: Dashboard > Resource groups > Kyowon_HOL >. Below it is the title 'kyowoncosmos | Data Explorer' and 'Azure Cosmos DB account'. On the left, a sidebar menu includes: Search (Ctrl+ /), New Container, Enable Azure Synapse Link (Preview) (which is highlighted with a red box), Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Quick start, Notifications, and Data Explorer (which is selected and highlighted with a grey bar). The main content area is titled 'SQL API'.



3. Feature under Setting, you will see Azure Synapse Link is enabled as below.

Feature	Status
Azure Synapse Link	Enrolled

Task 2 : Create Database and Collections

1. In [Azure Portal](#), goes to Azure Cosmos DB and select Data Explorer.
2. Put DB ID as CosmosDemo, choose Autoscale 4000, type IoTDeviceInfo for Container ID and put /id for Partition Key.

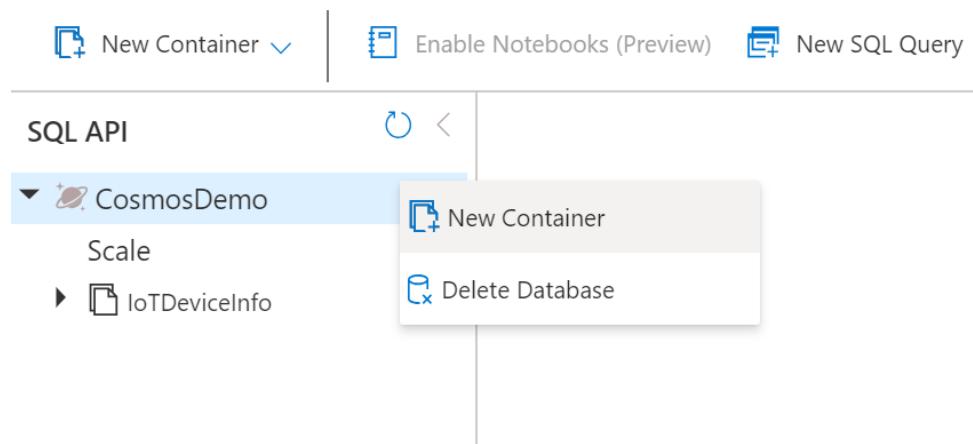
* Database id ⓘ
 Create new Use existing
CosmosDemo

Provision database throughput ⓘ
* Throughput (autoscale) ⓘ
 Autoscale Manual
Provision maximum RU/s required by this resource. Estimate your required RU/s with [capacity calculator](#).
Max RU/s
4000
Your database throughput will automatically scale from **400 RU/s** (**10% of max RU/s**) - **4000 RU/s** based on usage.
After the first 40 GB of data stored, the max RU/s will be automatically upgraded based on the new storage value. [Learn more](#).
Estimated monthly cost (USD): **\$35.04 - \$350.40** (1 region, 400 - 4000 RU/s, \$0.00012/RU)

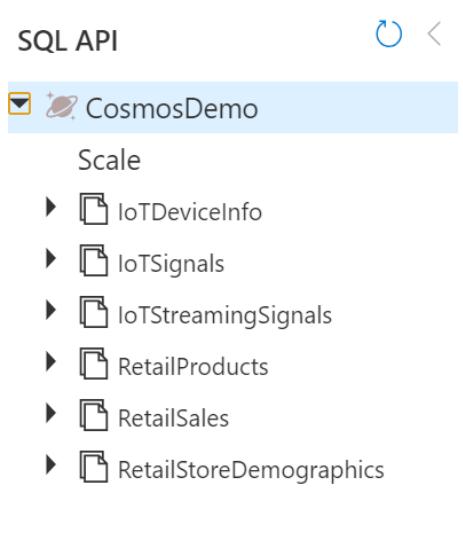
* Container id ⓘ
IoTDeviceInfo

* Partition key ⓘ
/id
 My partition key is larger than 100 bytes

3. Select CosmosDemo DB and click New Container.



4. Put IoTSignals for Collection ID and put /id for Partition Key.
5. Put IoTStreamingSignals for Collection ID and put /id for Partition Key.
6. Put RetailProducts for Collection ID and put /id for Partition Key.
7. Put RetailSales for Collection ID and put /id for Partition Key.
8. Put RetailStoreDemographics for Collection ID and put /id for Partition Key.



Task 3 : ADLS Gen2

1. In [Azure Portal](#), goes to ADLS Gen2 Storage Account.
2. Select Storage Explorer.

Dashboard > Resource groups > Kyowon_HOL >

kyowonadlsgen2 | Storage Explorer (preview)

Storage account

Search (Ctrl+ /) Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Data transfer
- Events
- Storage Explorer (preview)

CONTAINERS

FILE SHARES

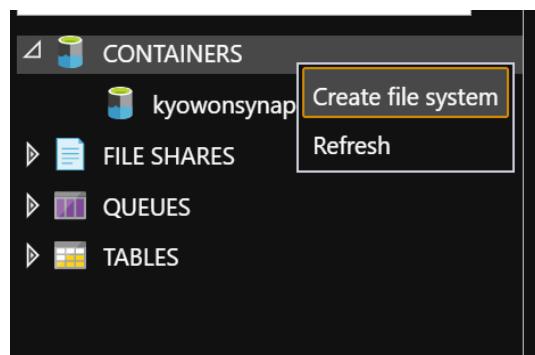
QUEUES

TABLES

Settings

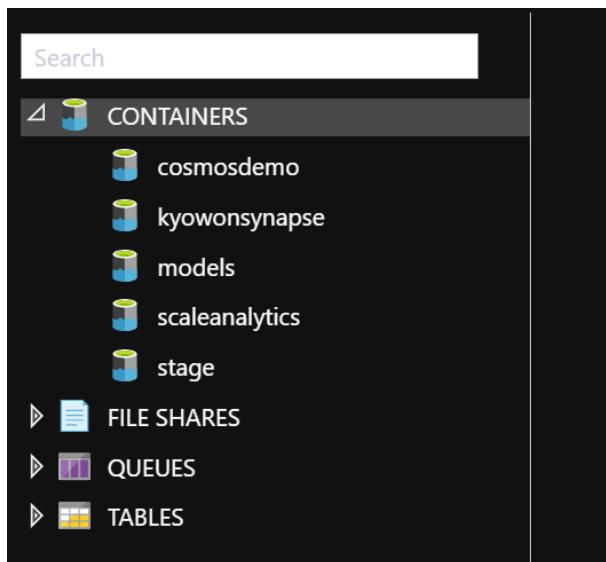
Access keys

3. Choose Container, right click and select Create File System.



4. Type cosmosdemo.
5. Create 'models' container.
6. Create 'scaleanalytics' container.
7. Create 'stage' container.

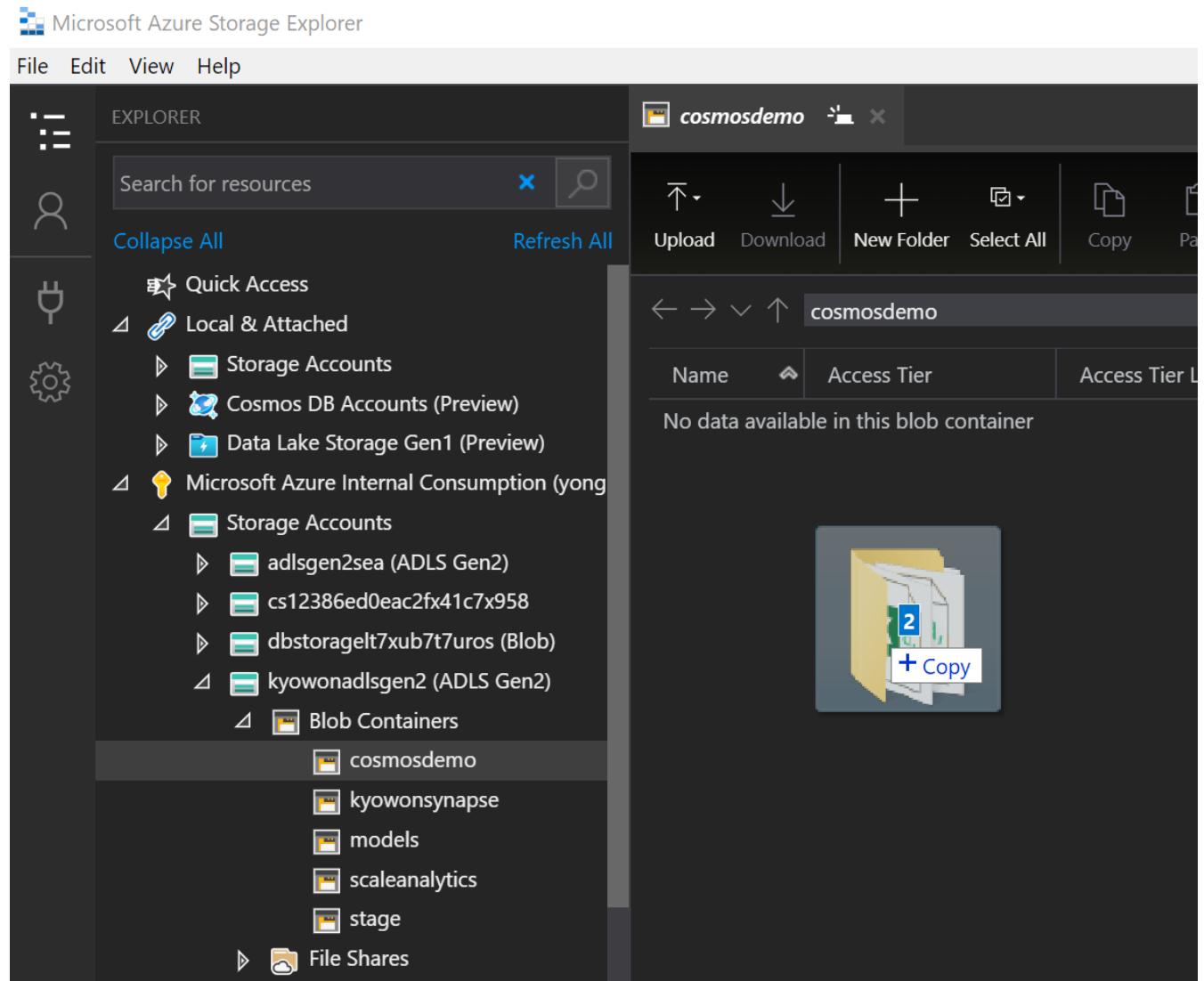
8. Create 'wwi-02' container.



9. Click Upload button and download Storage Explorer.



10. Run Storage Explorer and select ADLS Gen2. Upload all downloaded files from GitHub in following path Synapse_HOL\#Synapse_HOL\#LabFiles\#ADLSSGen2\#cosmosdemo to the cosmosdemo container.



11. Scaleanalytics and 'wwi-02' container also doing same.

Task 4 : Synapse Workspace

1. Goes to Azure Synapse Workspace from Azure Portal.
2. Select SQL Pool.

The screenshot shows the Azure Synapse workspace 'kyowonsynapse'. On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Settings, SQL Active Directory admin, Properties, Locks, Synapse resources (with 'SQL pools' highlighted), Apache Spark pools, Security, Firewalls, and Managed identities. The main content area displays resource group details: Resource group (change) : Kyowon_HOL, Status : Succeeded, Location : Southeast Asia, Subscription (change) : Microsoft Azure Internal Cons, Subscription ID : 2386ed0e-ac2f-41c7-958a-739, Managed virtual network : No, Managed Identity object ... : ecfa88fa-0589-4424-a585-84f, Workspace web URL : https://web.azuresynapse.net/, and Tags (change) : Click here to add tags. Below this is a section for Available resources with a search bar and a note that 'No pools provisioned'.

- Click +New to create a new SQL Pool. Put a name and set DW100c for Performance Level.

The screenshot shows the 'Create SQL pool' wizard on the 'Basics' tab. It has tabs for Basics, Additional settings, Tags, and Review + create. The Basics tab is active. It asks to create a SQL pool with configurations. A note says to complete the Basics tab and go to Review + Create to provision with smart defaults or visit each tab to customize. It shows 'SQL pool details' where the name is 'sqlpool' and the 'Performance level' is set to 'DW100c'. An estimated price of '1349.58 KRW' is shown with a link to 'View pricing details'. A tooltip on the right lists validation rules: Database, The value, Database, and No database.

- Finish the SQL Pool creation.
- Select Apache Spark Pool and click +New.
- Put name and set Performance Level.

* Basics * Additional settings Tags Summary

Create a Synapse Analytics Apache Spark pool with your preferred configurations. Complete the Basics tab then go to Review + create to provision with smart defaults, or visit each tab to customize.

Apache Spark pool details

Name your Apache Spark pool and choose its initial settings.

Apache Spark pool name * sparkpool

Node size family MemoryOptimized

Node size * Medium (8 vCPU / 64 GB)

Autoscale * Enabled

Number of nodes * 3 to 5

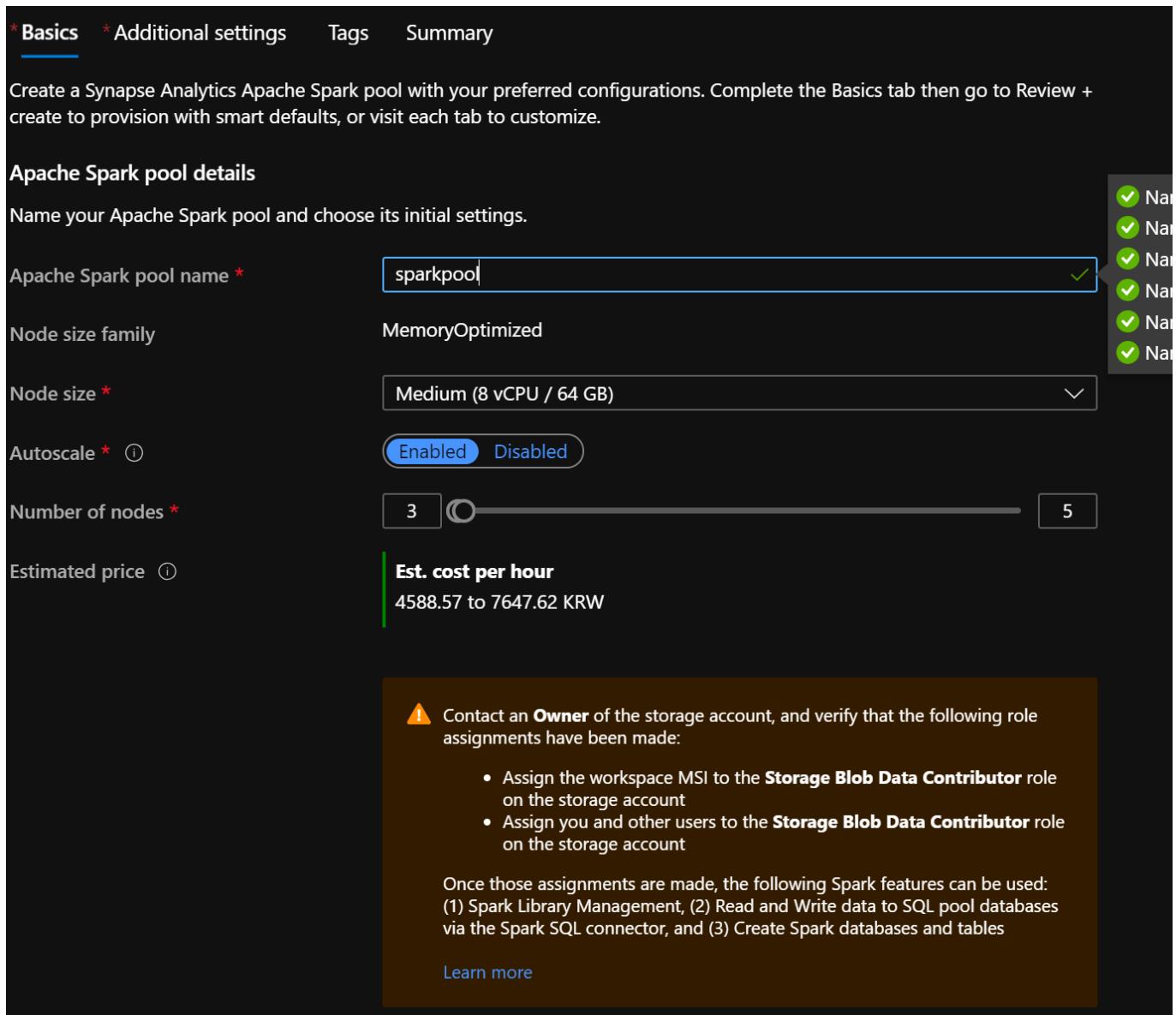
Estimated price ⓘ Est. cost per hour
4588.57 to 7647.62 KRW

⚠️ Contact an Owner of the storage account, and verify that the following role assignments have been made:

- Assign the workspace MSI to the **Storage Blob Data Contributor** role on the storage account
- Assign you and other users to the **Storage Blob Data Contributor** role on the storage account

Once those assignments are made, the following Spark features can be used:
(1) Spark Library Management, (2) Read and Write data to SQL pool databases via the Spark SQL connector, and (3) Create Spark databases and tables

[Learn more](#)



7. Review + Create.

Task 5: Synapse Studio SQL Analytics

In this Task, you are going to migrate DW Data(**OLAP Data**) from MySQL Server VM(On-Premise) to SQL Pool you just created. In Azure Synapse Workspace Overview Page, click below Link to launch Synapse Studio.

The screenshot shows the Azure Synapse Workspace Overview page. At the top, there are buttons for 'New SQL pool', 'New Apache Spark pool', 'Refresh', 'Reset SQL admin password', and 'Delete'. To the right of these is a 'Launch Synapse Studio' button, which is highlighted with a red box. Below the buttons, there's a list of workspace details including Resource group, Status, Location, Subscription, Subscription ID, Managed virtual network, Managed Identity object, and Workspace web URL. The 'Workspace web URL' is also highlighted with a red box. Further down, there's a section for 'Available resources' with tables for 'SQL pools' and 'Apache Spark pools'. The 'sqlpool' under 'SQL pools' is highlighted with a red box.

1. In the Develop, Click + and select import.

The screenshot shows the Microsoft Azure Synapse Analytics 'Develop' interface. On the left, there's a sidebar with icons for Home, Databases, Pipelines, Data Flows, and Jobs. The 'Develop' icon is selected. Above the main area, there are buttons for 'Publish all', 'Validate all', 'Refresh', 'Discard all', and a search bar. A dropdown menu is open, listing 'SQL script', 'Notebook', 'Data flow', 'Spark job definition', and 'Import'. The 'Import' option is highlighted with a red box.

2. Select SQL scripts in Synapse_HOL\#Synapse_HOL\#LabFiles\#Scripts\#SQL which you downloaded from GitHub.
3. Publish All.

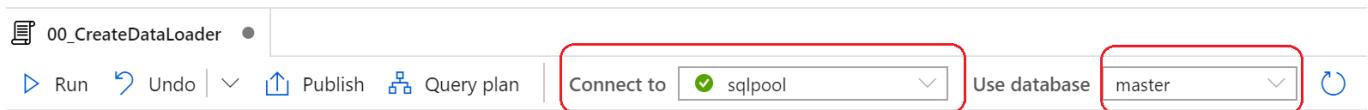
Publish all

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

Pending changes (7)

NAME	CHANGE	EXISTING
▲ SQL script		
00_CreateDataLoader	(New)	-
01_CreateDWTables	(New)	-
02_CreateCredential	(New)	-
03_CreateExternalTables	(New)	-
04_CreateTransactionTables	(New)	-
05_LoadData	(New)	-
06 OLTP_Table_Count	(New)	-

4. **(Optional for SSMS or Azure Data Studio)** To create Data Loader account, run 00_CreateDataLoader script. Connect sqlpool first.

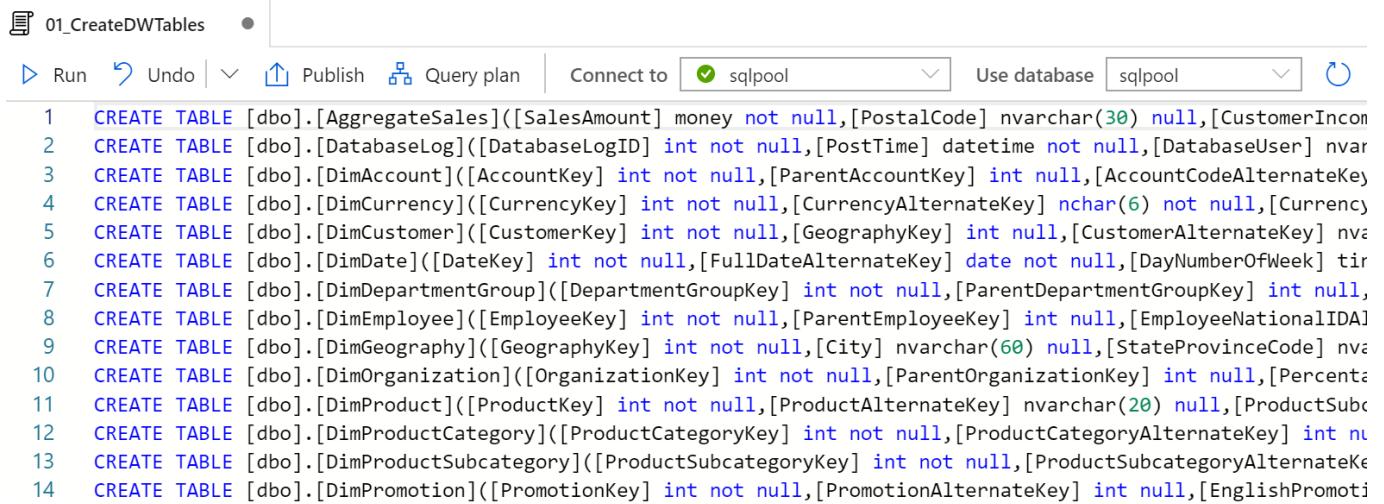


```

1 --Execute this line from the master database
2 CREATE LOGIN dataloader WITH PASSWORD = 'Demo@pass123';
3 CREATE USER dataloader FOR LOGIN dataloader;
4
5 --Execute the remainder of these lines from the sqlpool database
6 CREATE USER dataloader FOR LOGIN dataloader;
7 GRANT CONTROL ON DATABASE::sqlpool TO dataloader;
8 EXEC sp_addrolemember 'largerc', 'dataloader';
9

```

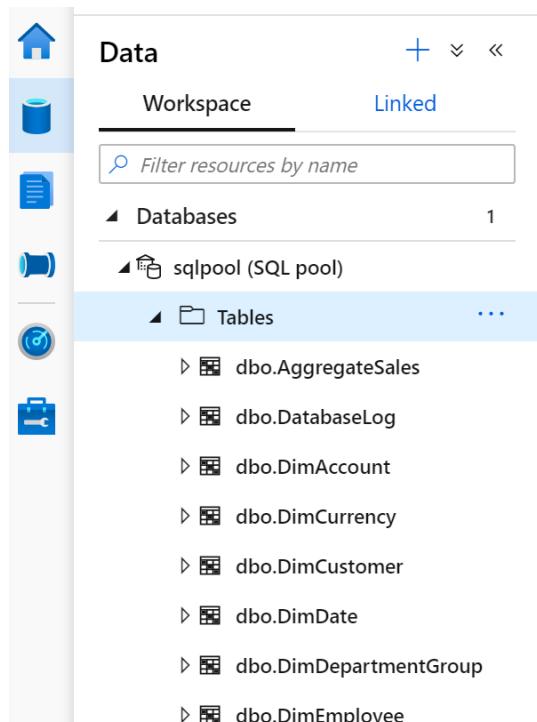
5. Next, For the MySQL Server VM DW Data Migration, create tables which align with MySQL DW Table schema. Run 01_CreateDWTables script.



The screenshot shows a SQL editor window titled "01_CreateDWTables". The toolbar includes "Run", "Undo", "Publish", "Query plan", "Connect to", "sqlpool", "Use database", "sqlpool", and a refresh icon. The code area contains a series of CREATE TABLE statements for a schema named "dbo". The statements define various dimensions and fact tables with their respective columns and data types. The code is as follows:

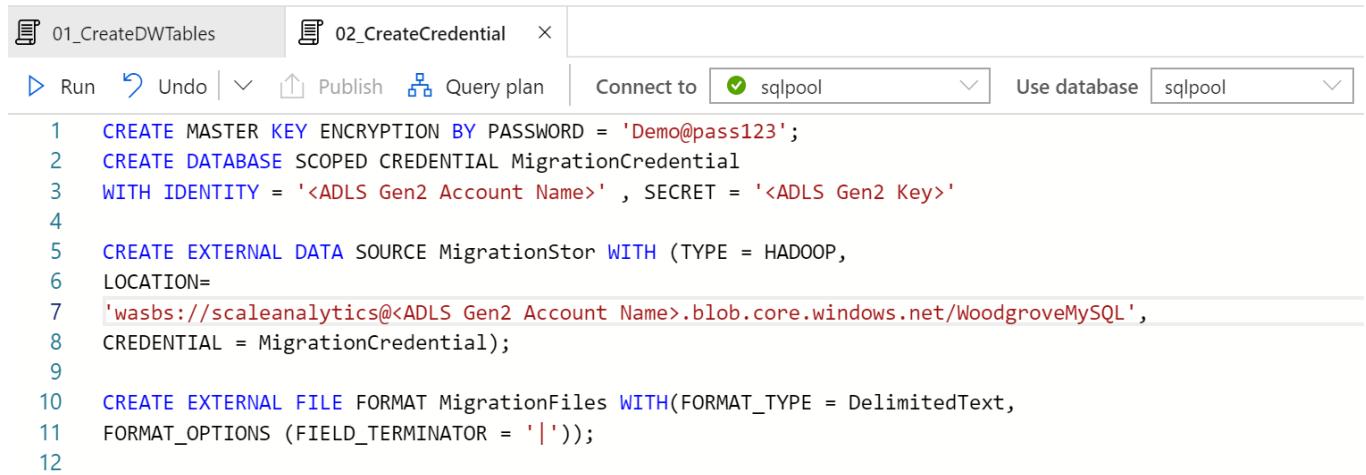
```
1 CREATE TABLE [dbo].[AggregateSales]([SalesAmount] money not null,[PostalCode] nvarchar(30) null,[CustomerIncon
2 CREATE TABLE [dbo].[DatabaseLog]([DatabaseLogID] int not null,[PostTime] datetime not null,[DatabaseUser] nvar
3 CREATE TABLE [dbo].[DimAccount]([AccountKey] int not null,[ParentAccountKey] int null,[AccountCodeAlternateKey]
4 CREATE TABLE [dbo].[DimCurrency]([CurrencyKey] int not null,[CurrencyAlternateKey] nchar(6) not null,[Currency
5 CREATE TABLE [dbo].[DimCustomer]([CustomerKey] int not null,[GeographyKey] int null,[CustomerAlternateKey] nva
6 CREATE TABLE [dbo].[DimDate]([DateKey] int not null,[FullDateAlternateKey] date not null,[DayNumberOfWeek] tir
7 CREATE TABLE [dbo].[DimDepartmentGroup]([DepartmentGroupKey] int not null,[ParentDepartmentGroupKey] int null,
8 CREATE TABLE [dbo].[DimEmployee]([EmployeeKey] int not null,[ParentEmployeeKey] int null,[EmployeeNationalIDA]
9 CREATE TABLE [dbo].[DimGeography]([GeographyKey] int not null,[City] nvarchar(60) null,[StateProvinceCode] nva
10 CREATE TABLE [dbo].[DimOrganization]([OrganizationKey] int not null,[ParentOrganizationKey] int null,[Percenta
11 CREATE TABLE [dbo].[DimProduct]([ProductKey] int not null,[ProductAlternateKey] nvarchar(20) null,[ProductSubc
12 CREATE TABLE [dbo].[DimProductCategory]([ProductCategoryKey] int not null,[ProductCategoryAlternateKey] int nu
13 CREATE TABLE [dbo].[DimProductSubcategory]([ProductSubcategoryKey] int not null,[ProductSubcategoryAlternateKe
14 CREATE TABLE [dbo].[DimPromotion]([PromotionKey] int not null,[PromotionAlternateKey] int null,[EnglishPromoti
```

6. Select Data Tab and check the created tables.



The screenshot shows the Azure Data Explorer interface under the "Data" tab. The left sidebar has icons for Home, Workspace, and Linked. The main area shows a "Workspace" section with a search bar and a "Databases" section containing one item: "sqlpool (SQL pool)". Under "sqlpool", there is a "Tables" section listing the following tables: dbo.AggregateSales, dbo.DatabaseLog, dbo.DimAccount, dbo.DimCurrency, dbo.DimCustomer, dbo.DimDate, dbo.DimDepartmentGroup, and dbo.DimEmployee.

7. In Develop Tab, Select 02_CreateCredential script. This script is creating Master Key, Credential, and File Format to access files which are in 'scaleanalytics' container Woodgrove MySQL folder. We already uploaded files which are exported from Data Source (MySQL DW Tables).

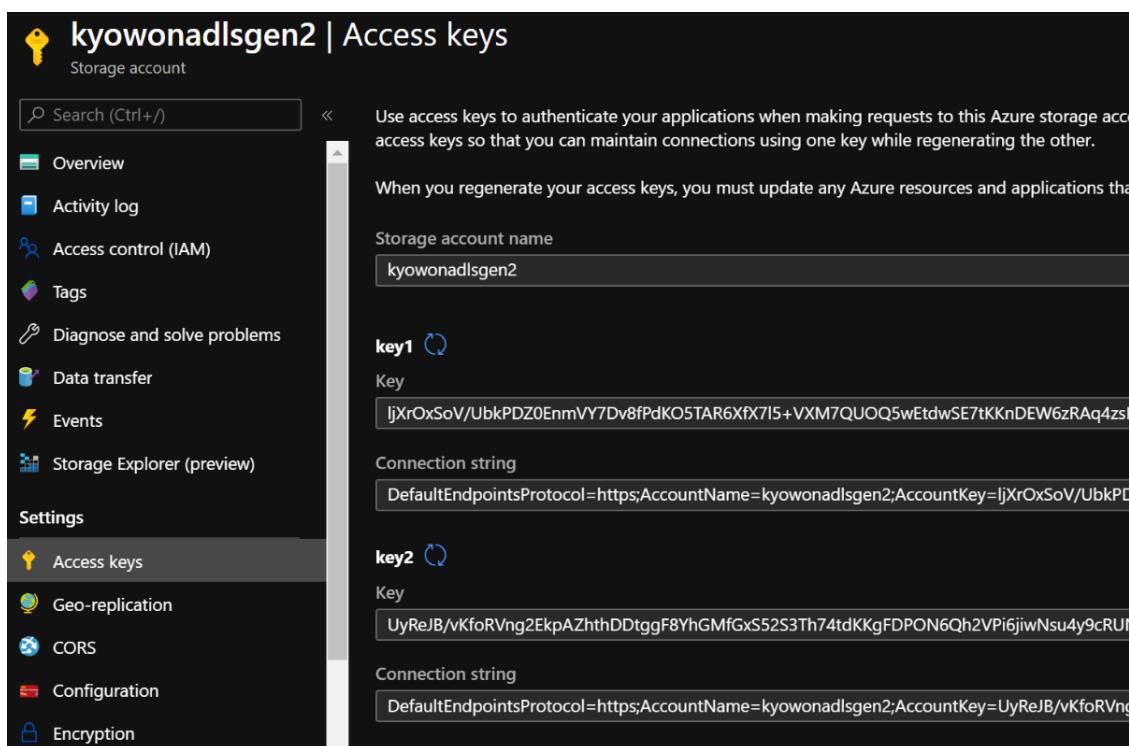


```

1 CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Demo@pass123';
2 CREATE DATABASE SCOPED CREDENTIAL MigrationCredential
3 WITH IDENTITY = '<ADLS Gen2 Account Name>' , SECRET = '<ADLS Gen2 Key>'
4
5 CREATE EXTERNAL DATA SOURCE MigrationStor WITH (TYPE = HADOOP,
6 LOCATION=
7 'wasbs://scaleanalytics@<ADLS Gen2 Account Name>.blob.core.windows.net/WoodgroveMySQL',
8 CREDENTIAL = MigrationCredential);
9
10 CREATE EXTERNAL FILE FORMAT MigrationFiles WITH(FORMAT_TYPE = DelimitedText,
11 FORMAT_OPTIONS (FIELD_TERMINATOR = '|'));
12

```

8. To check ADLS Gen2 Account Name and Key, Select Storage Account and goes to Access Key Tab. Check value and replace in the script.



The screenshot shows the Azure Storage Account 'kyowonadlsgen2' settings page. The 'Access keys' section is highlighted. It displays two sets of access keys: 'key1' and 'key2'. The 'key1' key has a value of 'ljXrOxSoV/UbkPDZ0EnmVY7Dv8fPdKO5TAR6Xfx7I5+VXM7QUOQ5wEtdwSE7tKKnDEW6zRAq4zsF'. The 'key2' key has a value of 'UyReJB/vKfoRVng2EkpAZhthDDtggF8YhGMfGxS52S3Th74tdKKgFDPON6Qh2VPi6jiwNsu4y9cRUN'. Below each key, there is a 'Connection string' field containing the protocol and account name.

9. Next, Run 03_CreateExternalTables script to create external tables which are linked with exported files.
10. Select Data Tab and check external tables are created in Workspace Tab. Run Tp 100 Query to check data.
11. Run 04_LoadDatascript for Migration.
12. Check tables data which you just migrated in sqlpool.

Task 6: Azure Synapse Ingest Wizard

In this Task, you are going to migrate Transaction Table(**OLTP Data**) from MySQL Server VM with Pipeline.

1. Run 05_CreateTransactionTable script to create a target table.
2. Goes to Azure Synapse Studio Home and click Ingest.

The screenshot shows the Azure Synapse workspace interface. At the top left, it says "kyowonsynapse". Below that are four main buttons: "Ingest" (highlighted with a red box), "Explore", "Analyze", and "Visualize". The "Ingest" button has a sub-description: "Use the easy data tool to import data from or via a pipeline". To the right is a large circular graphic with a bar chart and network connections. Below the buttons is a section titled "Resources" with "Recent" and "Pinned" tabs. Under "Recent", there are five items: "01_CreateDWTables", "02_CreateCredential", "04_LoadData", "03_CreateExternalTables", and "05.CreateTransactionTable". To the right is a "Useful links" sidebar with links like "Getting started", "Synapse Analytics overview", "Pricing", "Documentation", and "Give feedback".

3. Put Initial_Copy as a Task Name. Next.

The screenshot shows the "Properties" step of the Ingest Wizard. On the left is a vertical navigation menu with numbered steps: 1 Properties, 2 Source, 3 Destination, 4 Settings, 5 Summary, and 6 Deployment. Step 1 is active. On the right, the "Properties" section has a sub-instruction: "Enter name and description for the copy data task.". It contains fields for "Task name *" (set to "Initial_Copy") and "Task description" (empty). Below these is a "Task cadence or task schedule" section with two radio buttons: "Run once now" (selected) and "Run regularly on schedule".

4. Click + Create New Connection.

Properties
One time copy

Source
Connection
Dataset

Destination
Connection
Dataset

Settings

Summary

Deployment

Source data store
Specify the source data store for the copy task. You can use an existing data store connection or specify a new data store.

All Azure Database File Generic protocol NoSQL Services and apps

All Filter by name + Create new connection

CosmosDemo kyowonsynapse-
 WorkspaceDefaultStorage sqlpool

5. Select MySQL. Next

New linked service

mysql

All Azure Database File Generic protocol NoSQL Services and apps

Azure Database for MySQL MySQL

6. Put proper information as below.

- Name: MySQL_Server
- Server Name: 52.187.2.170
- Port: 3306

- Database Name: cohooltp
- User Name: demouser
- Password: Demo@pass123

Name *MySQL_Server

Description

Connect via integration runtime *
AutoResolveIntegrationRuntime

Connection string Azure Key Vault

Server name *
52.187.2.170

Port
3306

Database name *
cohooltp

User name *
demouser

Password Azure Key Vault

Password *
.....

7. Try Test Connection to check connectivity. Create.

Annotations

+ New

✓ Connection successful

Create

Back

🔗 Test connection

Cancel

8. Select created Data Source. Next.

The screenshot shows the 'Source data store' configuration screen. On the left, a sidebar lists steps 1 through 6: Properties, Source, Destination, Settings, Summary, and Deployment. Step 2, 'Source', is currently selected. The main area displays a list of data stores under 'All' categories: Azure, Database, File, Generic protocol, NoSQL, and Services and apps. A search bar 'Filter by name' and a 'Create new connection' button are also present. The 'MySQL_Server' connection is highlighted in blue, indicating it is selected.

9. Select Cardtransaction table. Next.

The screenshot shows the 'Source' configuration screen. The sidebar on the left shows steps 1 through 6. Step 2, 'Source', is selected. The main area has tabs for 'EXISTING TABLES' and 'USE QUERY'. Under 'EXISTING TABLES', there is a 'Filter by name...' input field and a 'Refresh' button. A list of tables is shown, with the 'cardtransaction' table checked. Below this is a 'Preview' section showing two rows of transaction data:

transactionID	accountID	transactionAmountUSD	transactionAmount	transactionCui
47F3C9CC- 2948-4069- B7BC- 260361539AAD	A914800993377925	86	86	USD
A949C89B- D346-4C42- AF18- 4F75FA0DD82C	A844427439174139	0	0	USD

10. In Apply Filter, go Next.

11. Select sqlpool for Destination Source. Next.

The screenshot shows the 'Copy data' blade in Azure Data Factory. On the left, a sidebar lists steps 1 through 6: Properties, Source, Destination, Settings, Summary, and Deployment. Step 2, 'Source MySQL', is selected. Step 3, 'Destination', is expanded, showing 'Connection' and 'Dataset'. Step 4, 'Settings', is currently active. The main area is titled 'Destination data store' with the sub-instruction 'Specify the destination data store for the copy task. You can use an existing data store connection or specify a new data store.' Below this are tabs for All, Azure, Database, File, Generic protocol, NoSQL, and Services and apps. Under the All tab, there is a dropdown menu set to 'All', a search bar 'Filter by name', and a 'Create new connection' button. A list of data stores is shown, including 'CosmosDemo', 'kyowonsynapse-WorkspaceDefaultStorage', and 'sqlpool'. The 'sqlpool' item is highlighted with a blue background.

12. Select dbo.CardTransaction table in Table Mapping. Next.

Table mapping

For each table you have selected to copy in the source data store, select a corresponding table in the destination data store or specify the table mapping.

The screenshot shows the 'Table mapping' interface. On the left, under 'Source', the table 'cardtransaction' is listed. An arrow points from this table to the 'Destination' column. The 'Destination' column contains a dropdown menu labeled '-Select-' with a filter bar 'Filter...'. A list of destination tables is shown, including 'dbo.AggregateSales', 'dbo.AggregateSales_External', 'dbo.CardTransaction', 'dbo.DatabaseLog', 'dbo.DatabaseLog_External', 'dbo.DimAccount', 'dbo.DimAccount_External', and 'dbo.DimCurrency'. The table 'dbo.CardTransaction' is highlighted with a blue border.

13. In Column Mapping, go Next.

14. In Setting screen, click +New for Staging Account Linked Service.

Settings

More options for data movement

Fault tolerance

 ▼ ⓘ

▲ Performance settings

Enable staging

 ⓘ

▲ Staging settings

Staging account linked service

 Select... ▼ ⓘ+ New

Enable Compression

 ⓘ

▲ Advanced settings

Allow PolyBase

 ⓘ

Reject type

 Value ▼

Reject value

 0

Use type default

Data integration unit

 Auto ▼ ⓘ Edit

You will be charged # of used DIUs * copy duration * \$0.25/DIU-hour. Local currency subscription type. [Learn more](#)

Degree of copy parallelism

 ⓘ Edit

15. Type StagingBlob for name and select ADLS Gen2.

Name *
StagingBlob

Description

Connect via integration runtime *
AutoResolveIntegrationRuntime

Authentication method
Account key

Connection string Azure Key Vault

Account selection method
 From Azure subscription Enter manually

Azure subscription
Microsoft Azure Internal Consumption (2386ed0e-ac2f-41c7-958a-7397006dce05)

Storage account name *
kyowonadlsgen2

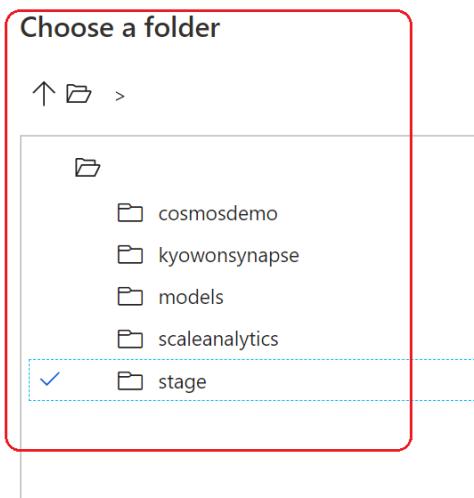
Additional connection properties
+ New

Test connection
 To linked service To file path

Annotations
+ New

+ Advanced ⓘ

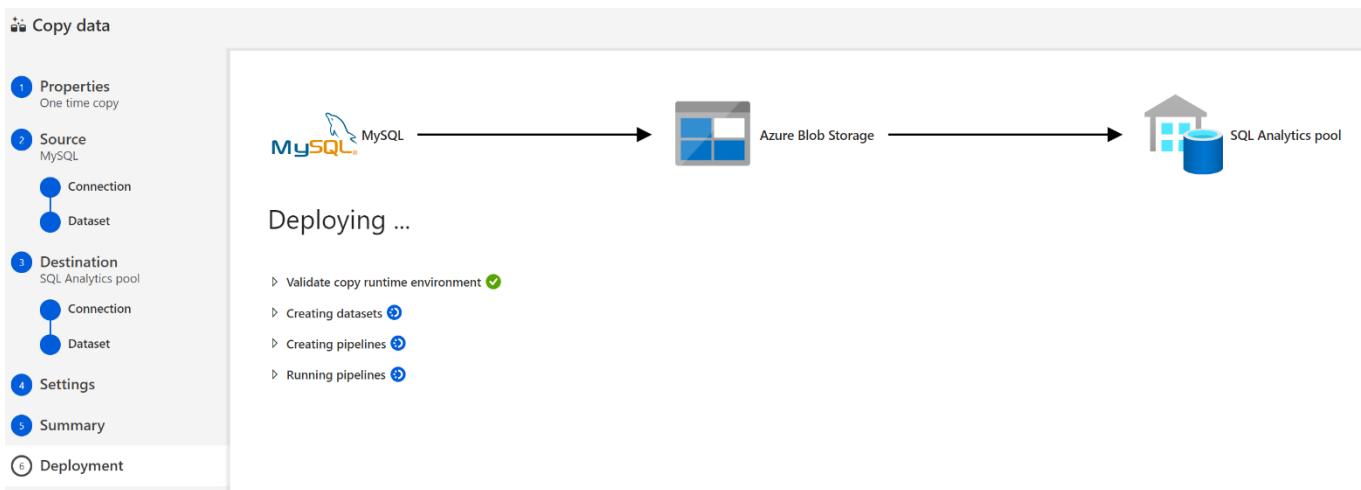
16. Click browse for Storage Path and select Stage.



17. Next.

18. Review Summary Page and go Next.

19. Pipeline will be Deployed and will be automatically triggered.



20. Click Monitor button to go Monitor page and click Detail in Pipeline Activity.

Initial_Copy

The screenshot shows the 'Monitor' page for the 'Initial_Copy' pipeline. It displays the following information:

- Pipeline run ID:** 583b49a6-0fbb-4ea9-b5e8-d5a8ca61c74e
- Status:** All status
- Showing:** 1 - 1 of 1 items

ACTIVITY NAME	ACTIVITY TYPE	RUN START ↑	DURATION	STATUS	INTEGRATION RUNTIME
Copy_fq4	Copy	6/23/20, 6:19:04 AM	00:00:22	✓ Succeeded	DefaultIntegrationRuntime (Southe)

A red box highlights the 'Details' button under the 'Copy_fq4' row.

21. Check Details.

Details ↻ Refresh ↗ ↘ ×

Learn more on copy performance details from here. Provide feedback on performance.

Activity run id: 7791bd5c-5c2a-4276-a0ff-1732f28e533d

```

graph LR
    MySQL["MySQL"] -- "Succeeded" --> Blob["Azure Blob Storage  
Region: Southeast Asia"]
    Blob -- "Succeeded" --> Synapse["Azure Synapse Analytics (formerly SQL DW)  
Region: Southeast Asia"]
  
```

MySQL

Succeeded

Azure IR region: Southeast Asia

Azure Blob Storage

Succeeded

Azure IR region: Southeast Asia

Azure Synapse Analytics (formerly SQL DW)

Region: Southeast Asia

Data read: 12.134 MB

Rows read: 31,364

Peak connections: 1

Data written: 0 byte

Rows written: 31,364

Throughput: 730.882 KB/s

Copy duration 00:00:17

MySQL → Azure Blob Storage

Details	Working duration	Total duration
Queue ⓘ	00:00:00	00:00:02
Transfer ⓘ	[Time to first byte ⓘ Reading from source ⓘ Writing to sink ⓘ]	00:00:03

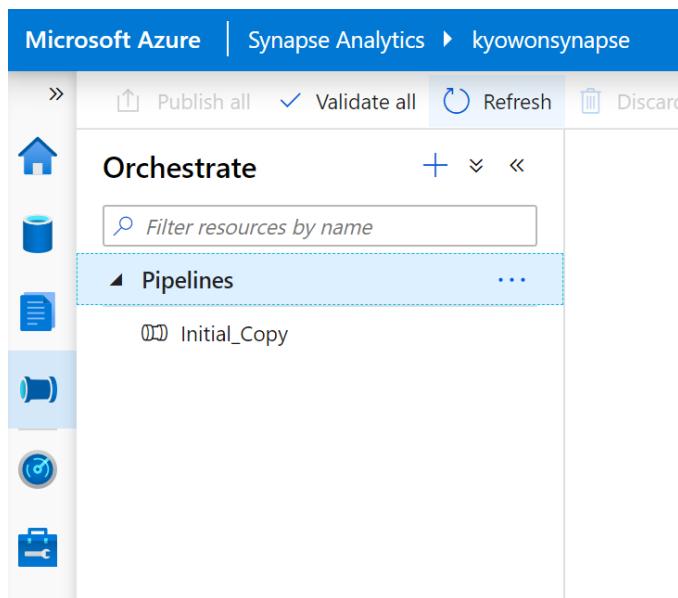
Azure Blob Storage → Azure Synapse Analytics (formerly SQL DW)

22. Come back to Develop tab and run 06_OLTP_Table_Count script to check row count.

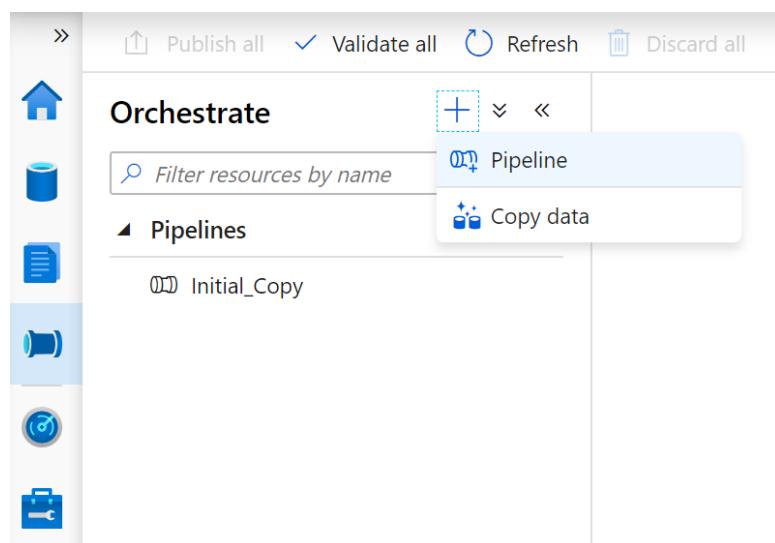
Task 7: Azure Synapse Pipeline

In this Task, you are going to do an Incremental Copy from Transactional Table in MySQL Server VM with Pipeline.

1. In Synapse Studio, Select Orchestrate tab and Refresh. You can see a pipeline which is created from Ingest Wizard.



2. Click + and select Pipeline.



3. Type Incremental_Copy for the name in Property tab.

Properties

General

i Choose a name for your pipeline.
This name can be updated at any time until it is published.

Name *

Description

Concurrency



Annotations

+ New

4. Drag & Drop General -> LookUp in Activities.

The screenshot shows the Azure Data Factory Pipeline designer interface. On the left, there's a sidebar titled 'Activities' with a search bar and a list of activity types under 'General': Append variable, Delete, Execute Pipeline, Execute SSIS package, Get Metadata, Kusto, **Lookup**, Stored procedure, Set variable, and Validation. The 'Lookup' item is highlighted with a red rectangle. In the center, a 'Lookup' activity is selected, shown in a preview window with the name 'Lookup1'. Below the preview, the 'General' tab of the properties panel is active, showing the 'Name' field set to 'Lookup1', a 'Description' field (empty), and a 'Timeout' field set to '7.00:00:00'. Other tabs like 'Settings' and 'User properties' are also visible.

5. In Settin tab in LookUp1, select Destination Data Set(sqlpool) which is created from Ingest Wizard.

The screenshot shows the 'Settings' tab of a data flow component. The 'Source dataset *' dropdown is open, displaying two options: 'DestinationDataset_fq4' and 'SourceDataset_fq4'. The 'DestinationDataset_fq4' option is highlighted with a blue border, indicating it is selected.

6. Select Query in Use Query section and type SELECT MAX([transactionDateTime]) AS transactionDateTime FROM [dbo].[CardTransaction]; .

The screenshot shows the 'Settings' tab of a data flow component. The 'Source dataset *' dropdown is set to 'DestinationDataset_fq4'. Under the 'Use query' section, the 'Query' radio button is selected. A query is typed into the text area: 'SELECT MAX([transactionDateTime]) AS transactionDateTime FROM [dbo].[CardTransaction];'. The 'First row only' checkbox is checked.

7. Click Preview Data to check data.

Preview data

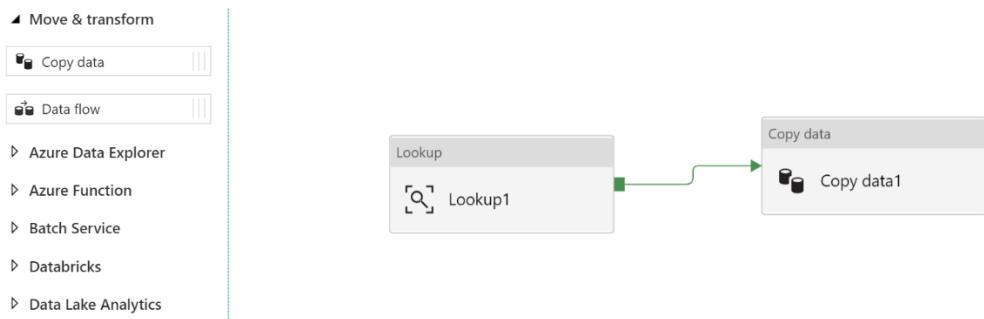
Linked service: sqlpool

Object:

transactionDateTime

2013-05-26T15:44:00

8. Drag & Drop Copy Data in Activities and link together with Lookup1.



9. Select Copy data1 to activate, in Source tab, select SourceDataSet(MySQL) which is created from Ingest Wizard. Type SELECT * FROM cohoOLTP.CardTransaction WHERE transactionDateTime > '@{activity('Lookup1').output.firstRow.transactionDateTime}'; in Query area.

Source dataset * Open + New Preview data

Query *
SELECT * FROM cohoOLTP.CardTransaction WHERE transactionDateTime > '@{activity('Lookup1').output.firstRow.transactionDateTime}';

Add dynamic content [Alt+P]

Additional columns ⓘ + New

10. Go to Sink tab and select destination as sqlpool.

Sink dataset * New

Select...
DestinationDataset_fq4

DestinationDataset_fq4

11. Finally do the same thing with what we have done in Ingest Wizard.

The screenshot shows the 'Settings' tab of a copy activity configuration screen. The 'Enable staging' checkbox is checked and highlighted with a red rectangle. Below it, the 'Staging account linked service' dropdown is set to 'StagingBlob' and the 'Storage Path' input field contains 'stage', both of which are also highlighted with a red rectangle. A 'Test connection' button is visible next to the storage path field, and a 'Browse' button is shown in a dashed box.

General Source Sink¹ Mapping Settings User properties

Data integration unit: Auto Edit

Degree of copy parallelism: Edit

Fault tolerance:

Enable staging ⓘ

▲ Staging settings

Staging account linked service: StagingBlob ⓘ

Storage Path: stage ⓘ

Enable Compression: ⓘ

12. Now click Publish All.

13. Click Add Trigger -> Trigger Now.

The screenshot shows the Azure Data Factory pipeline editor. At the top, there are navigation links: Validate, Debug, and Add trigger. A dropdown menu is open over the 'Add trigger' link, with 'Trigger now' selected. Below the editor area, a pipeline diagram is shown. It consists of two main components: a 'Lookup' activity named 'Lookup1' and a 'Copy data' activity named 'Copy data1'. An output from 'Lookup1' is connected to the 'Copy data1' activity. The 'Copy data1' activity has a green circular status icon in the top right corner.

✓ Validate ⚡ Debug ⚡ Add trigger

Trigger now

New/Edit

Lookup
Lookup1

Copy data
Copy data1

14. Again you can check status with Monitoring tab on left.

Pipeline runs

Time : Last 24 hours (6/22/20 7:12 AM - 6/23/20 7:12 AM) | Time zone : Seoul (UTC+9) | Runs : Latest runs | List | Gantt

All status | Rerun | Cancel | Refresh | Edit columns

Showing 1 - 2 items

Pipeline Name	Run Start	Duration	Triggered By	Status	Parameters
Incremental_Copy	6/23/20, 7:11:43 AM	00:00:28	Manual trigger	In progress	
Initial_Copy	6/23/20, 6:19:00 AM	00:00:26	Manual trigger	Succeeded	

15. Select Incremental_Copy to check Activity Details.

Activity runs

Pipeline run ID b86bb0a7-d01a-42f3-b0e1-90537b748469

All status

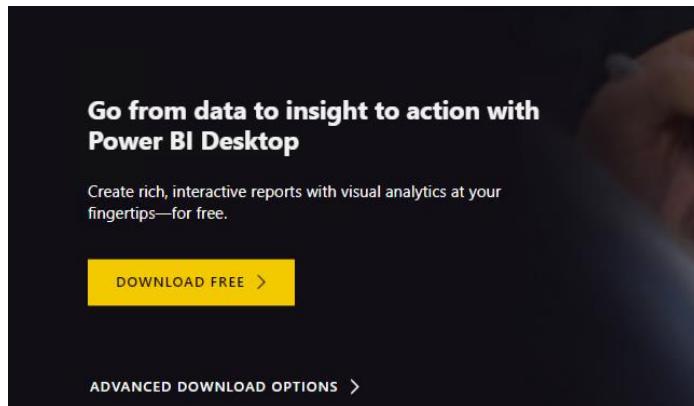
Showing 1 - 2 of 2 items

Activity Name	Activity Type	Run Start ↑	Duration	Status	Integration Runtime	User Properties	Error	Run ID
Copy data1	Copy	6/23/20, 7:11:50 AM	00:00:22	Succeeded	DefaultIntegrationRuntime (Southeast Asia);DefaultInteg			df743632-b
Lookup1	Lookup	6/23/20, 7:11:46 AM	00:00:04	Succeeded	AutoResolveIntegrationRuntime (Southeast Asia)			840b55a9-7

16. Come back to Develop tab and run 06_OLTP_Table_Count script to check row count.

Task 8 : Data Visualization with Power BI.

1. Go to download page of Power BI Desktop (<https://powerbi.microsoft.com/en-us/desktop/>).
2. Click Download Free link.



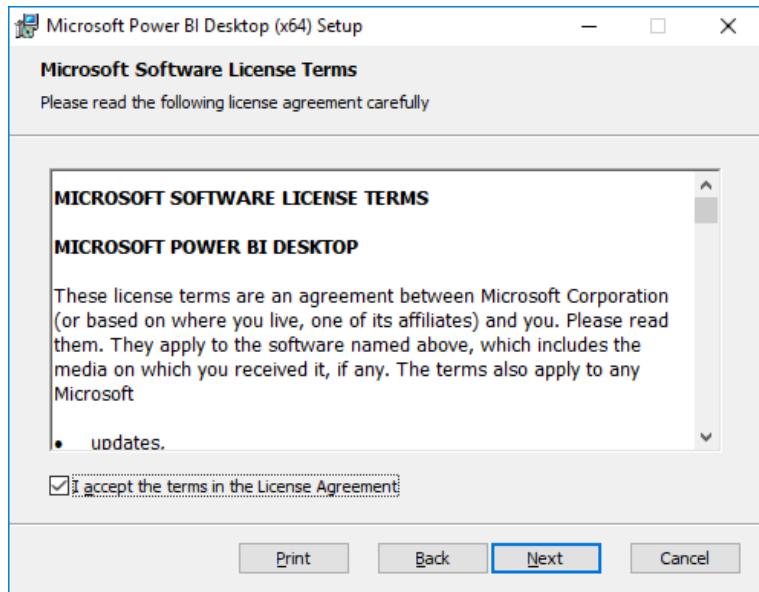
3. Run installer.



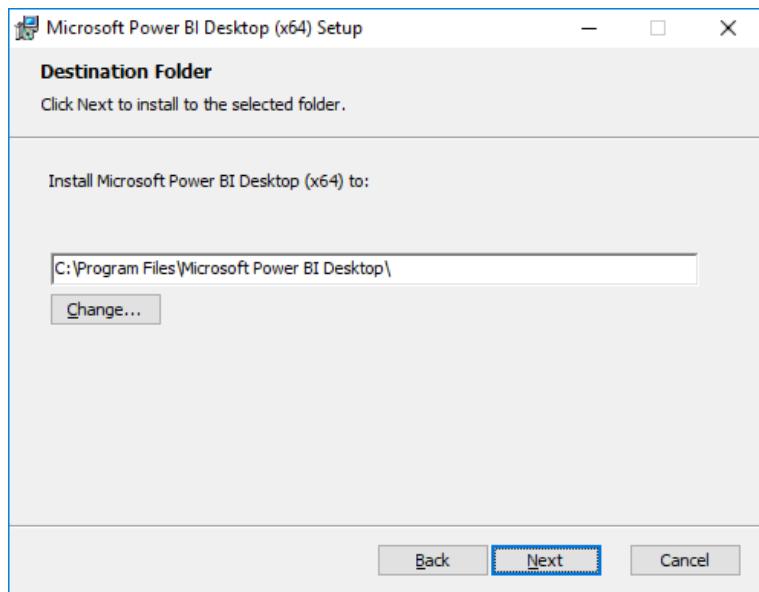
4. Click Next in Welcome page.



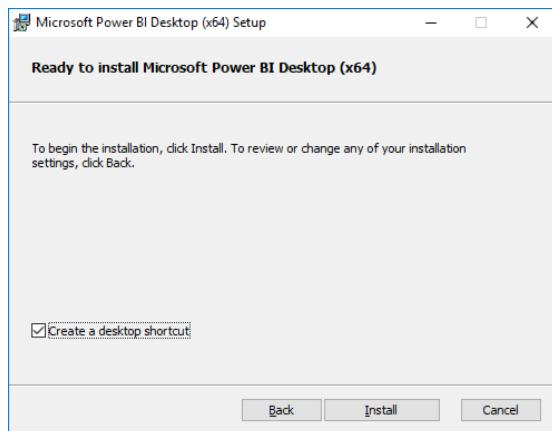
5. Tick agreement and click next.



6. Select folder to install.



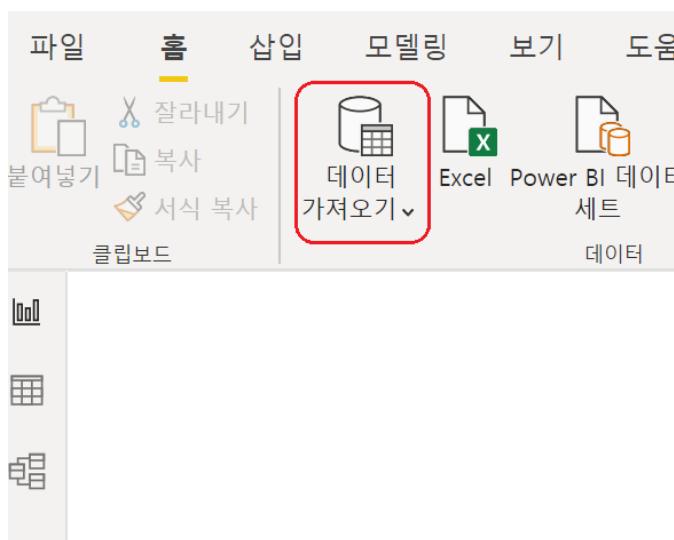
7. Click install.



8. Finish Microsoft Power BI Desktop installation and launch.



9. Click import data in Power BI.



10. Select Azure and select Azure SQL Database. Connect.

데이터 가져오기

The screenshot shows the 'Get Data' interface with a sidebar on the left containing categories: 모두 (All), 파일 (File), 데이터베이스 (Database), Power Platform, Azure (selected), 온라인 서비스 (Online Services), and 기타 (Others). The main area is titled 'Azure' and lists various Azure services with their icons and names. The 'Azure SQL Database' option is highlighted with a gray background. At the bottom right of the main area are two buttons: '연결' (Connect) and '취소' (Cancel).

- Azure SQL Database
- Azure SQL Data Warehouse
- Azure Analysis Services 데이터베이스
- PostgreSQL용 Azure Database
- Azure Blob Storage
- Azure Table Storage
- Azure Cosmos DB
- Azure Data Lake Storage Gen2
- Azure Data Lake Storage Gen1
- Azure HDInsight(HDFS)
- Azure HDInsight Spark
- HDInsight Interactive Query
- Azure Data Explorer(Kusto)
- Azure Cost Management
- Azure Time Series Insights(베타)

11. Go to Synapse Workspace Overview Page in Azure Portal and copy SQL Endpoint.

The screenshot shows the Azure Synapse Workspace Overview page. In the top right corner, there is a 'Copy to clipboard' button next to the 'SQL endpoint' value 'kyowonsynapse.sql.azuresynapse.net'. This value is also highlighted with a red box. Other visible details include the resource group 'Kyown_HOL', status 'Succeeded', location 'Southeast Asia', and subscription information.

12. Paste value in Server area and choose Direct Query.

SQL Server 데이터베이스

서버 ⓘ

kyowonsynapse.sql.azuresynapse.net

데이터베이스(선택 사항)

데이터 연결 모드 ⓘ

가져오기

DirectQuery

▷ 고급 옵션

확인

취소

13. You can log in Synapse Workspace Admin account with Database Login or log in with Microsoft account.

14. Select DimGeography table and load.

탐색 창

The screenshot shows the Power BI Data Load Preview interface. On the left, a sidebar lists various dimension tables with checkboxes. The checkbox for 'DimGeography' is checked and highlighted with a yellow border. The main area displays the 'DimGeography' table with the following data:

GeographyKey	City	StateProvinceCode	StateProvinceName
10	Newcastle	NSW	New South Wales
22	East Brisbane	QLD	Queensland
228	Basingstoke Hants	ENG	England
514	Cheektowaga	NY	New York
95	Toronto	ON	Ontario
295	Baldwin Park	CA	California
487	Saint Louis	MO	Missouri
565	La Vergne	TN	Tennessee
119	Grevenbroich	BY	Bayern
475	Branch	MN	Minnesota
3	Darlinghurst	NSW	New South Wales
192	Paris	75	Seine (Paris)
482	Ferguson	MO	Missouri
138	Berlin	HH	Hamburg
293	Alpine	CA	California
507	Santa Fe	NM	New Mexico
12	North Sydney	NSW	New South Wales
352	Palo Alto	CA	California
474	Zeeland	MI	Michigan
569	Nashville	TN	Tennessee
255	London	ENG	England
500	Winston-Salem	NC	North Carolina
387	Englewood	CO	Colorado

At the bottom, there are buttons for '관련 테이블 선택' (Select Related Table), '로드' (Load) in yellow, '데이터 변환' (Data Transformation), and '취소' (Cancel).

15. Check CountryRegionCode.

The screenshot shows the Power BI Field Explorer. On the left, a tree view of fields under 'DimGeography'. The 'CountryRegionCode' field is selected and highlighted with a yellow border. The tree structure includes:

- DimGeography
 - City
 - CountryRegionCode
 - EnglishCountryRegionName
 - FrenchCountryRegionName
 - GeographyKey
 - PostalCode
 - SalesTerritoryKey
 - SpanishCountryRegionName

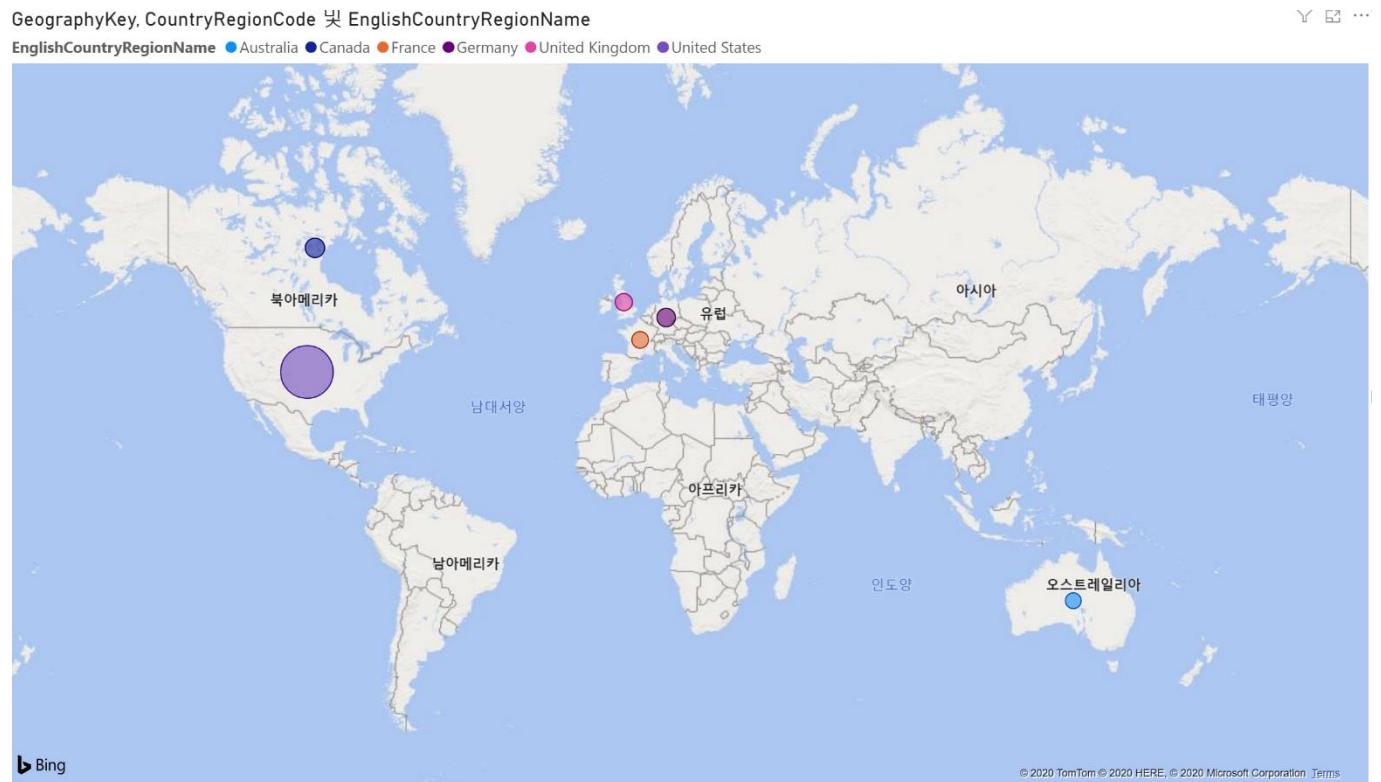
16. To show more specific information, Drag & Drop EnglishCountryRegionName to legend and CountryRegion to size.

The screenshot shows the Power BI Fields pane. On the left, there is a toolbar with various icons for data types and transformations. Below the toolbar, there are sections for Location (위치), Legend (범례), Latitude (위도), Longitude (경도), and Size (크기). The Location section contains fields: CountryRegionCode, EnglishCountryRegionName, and GeographyKey. The Size section contains the same field: GeographyKey. On the right, the Fields pane displays the DimGeography table with the following columns:

필드
City
CountryRegionCode
EnglishCountryRegionName
FrenchCountryRegionName
GeographyKey
PostalCode
SalesTerritoryKey
SpanishCountryRegionName
StateProvinceCode
StateProvinceName

The fields EnglishCountryRegionName and GeographyKey are checked, indicating they are selected for use in the report.

17. Check Map Visualization as below.



Task 9: Create & Populate Customer Information Table

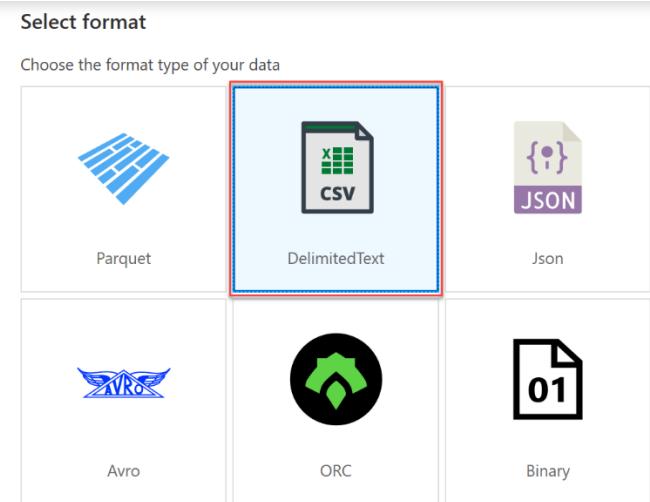
1. Select Develop Tab and open 11_01_Create_Customer_Info_Table script. Run.
2. Go to Data tab and click + to create Data Set.

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, there is a navigation sidebar with icons for Home, Data, Develop, Orchestrate, Monitor, and Manage. The 'Data' icon is highlighted with a red box. The main area is titled 'Data' and contains a search bar with 'Filter resources by name' and a 'Dataset' button. Below the search bar, there are three categories: 'Storage accounts' (4), 'Databases' (7), and 'Datasets' (179). A large red box highlights the 'Dataset' button.

3. Choose ADLS Gen2. Next

The screenshot shows the 'New dataset' creation dialog. At the top, it says 'Select a data store' with a search bar. Below the search bar, there are tabs: All, Azure, Database, File, Generic protocol, NoSQL, and Services and apps. The 'Azure' tab is selected and highlighted with a blue border. The main area displays a grid of data store options. The 'Azure Data Lake Storage Gen2' option is highlighted with a blue box. Other visible options include Azure Blob Storage, Azure Cosmos DB (MongoDB API), Azure Cosmos DB (SQL API), Azure Data Explorer (Kusto), Azure Data Lake Storage Gen1, and two other datasets whose names are partially visible.

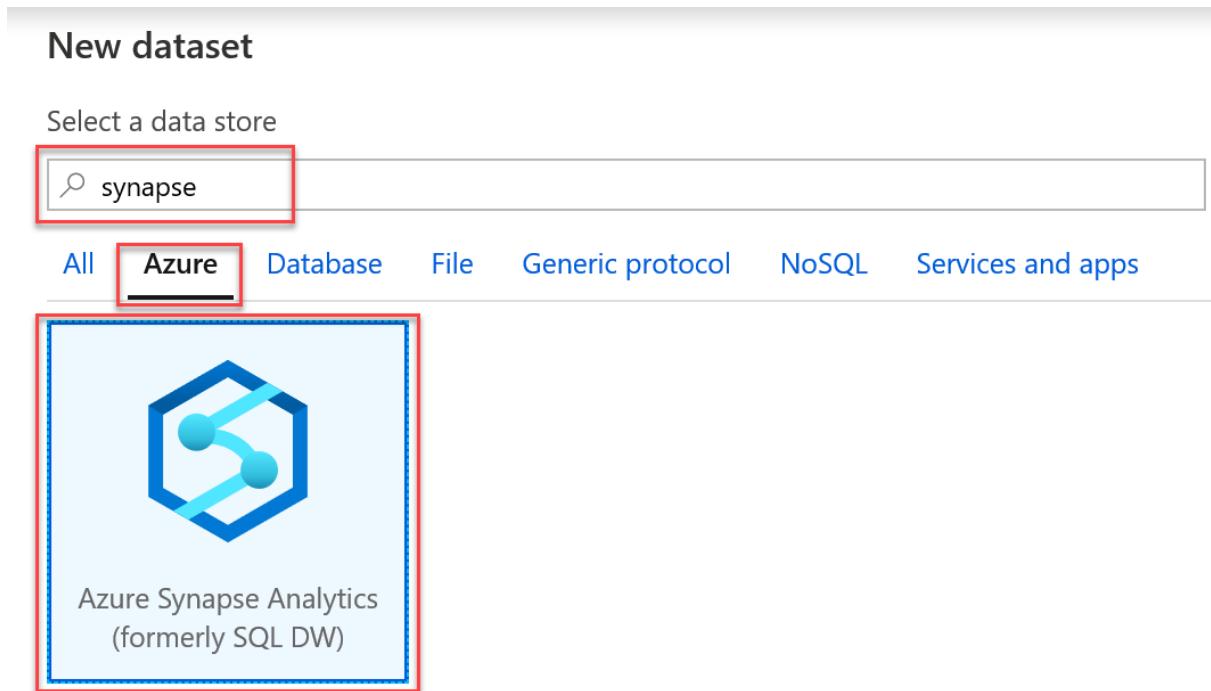
4. Select Delimited Text. Next



5. Set Property as following.

Field	Value
Name	Enter asamcw_customerinfo_csv
Linked service	Select <ADLS Gen2 Account Name>
File Path - Container	Enter wwi-02
File Path - Directory	Enter customer-info
File Path - File	Enter customerinfo.csv
First row as header	Checked
Import schema	Select From connection/store

6. Click + and create Data Set.
7. Select Azure and select **Azure Synapse Analytics (formerly SQL DW)**.



8. Set Property as below.

Field	Value
Name	Enter asamcw_customerinfo_asa
Linked service	Select WorkspaceDefault
Table name	Select WWICustomerInfo .
Import schema	Select From connection/store

9. Tick Edit check box and type table name as below.

Set properties

Name
asamcw_customerinfo_asa

Linked service *
kyowonsynapse-WorkspaceDefaultSqlServer

Table name
dbo . WWICustomerInfo

Edit

Import schema
 From connection/store None

10. Put DB name as sqlpool.

11. Publish All.

12. Go to Orchestrate tab and click + to create new Pipeline.

13. Put name as **Copy Customer Information**.

Properties

General

i Choose a name for your pipeline.
This name can be updated at any time until it is published.

Name *

Copy Customer Information

Description

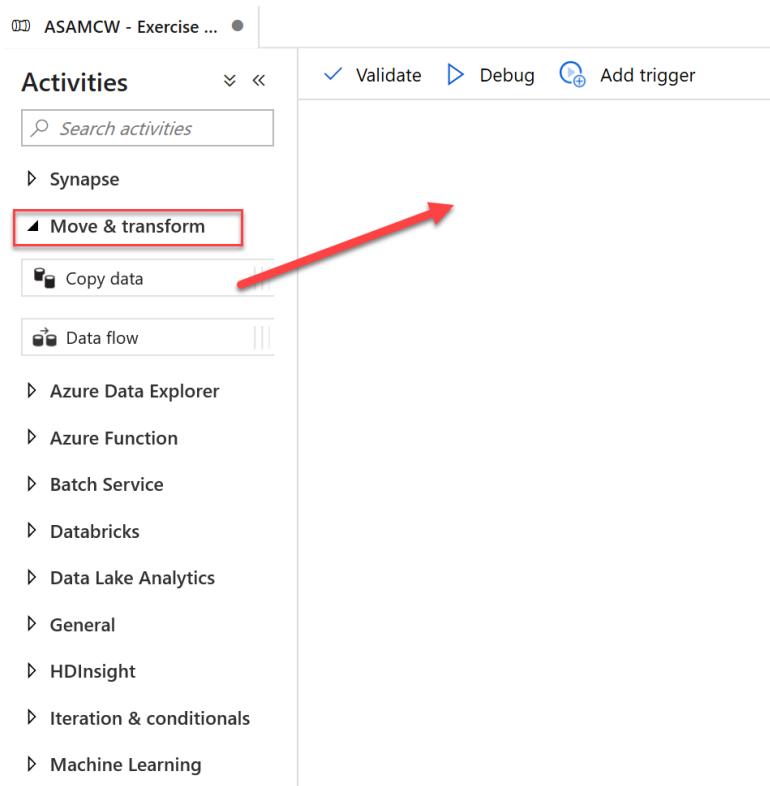
Concurrency

(i)

Annotations

+ New

14. Drag & Drop Copy Data.



15. Put name as below in General tab.

General Source Sink Mapping Settings User properties

Name *	<input type="text" value="Copy Customer Information"/>	Learn more
Description	<input type="text"/>	
Timeout	<input type="text" value="7.00:00:00"/>	
Retry	<input type="text" value="0"/>	
Retry interval	<input type="text" value="30"/>	
Secure output	<input type="checkbox"/>	
Secure input	<input type="checkbox"/>	

16. Choose the data set you just created in Source tab.

General **Source** Sink Mapping Settings User properties

Source dataset * asamcw_customerinfo_csv

File path type File path in dataset Wildcard file path Prefix List of files ⓘ

Start time (UTC) End time (UTC)

Filter by last modified

Recursively

Enable partition discovery

Max concurrent connections

Skip line count

Additional columns ⓘ + New

17. In Sink tab, select Destination Source and select Bulk Insert. Type truncate table dbo.WWICustomerInfo in Pre-Copy Script.

General Source **Sink** Mapping Settings User properties

Sink dataset * asamcw_customerinfo_asa

Copy method PolyBase ⓘ Copy command (Preview) ⓘ Bulk insert

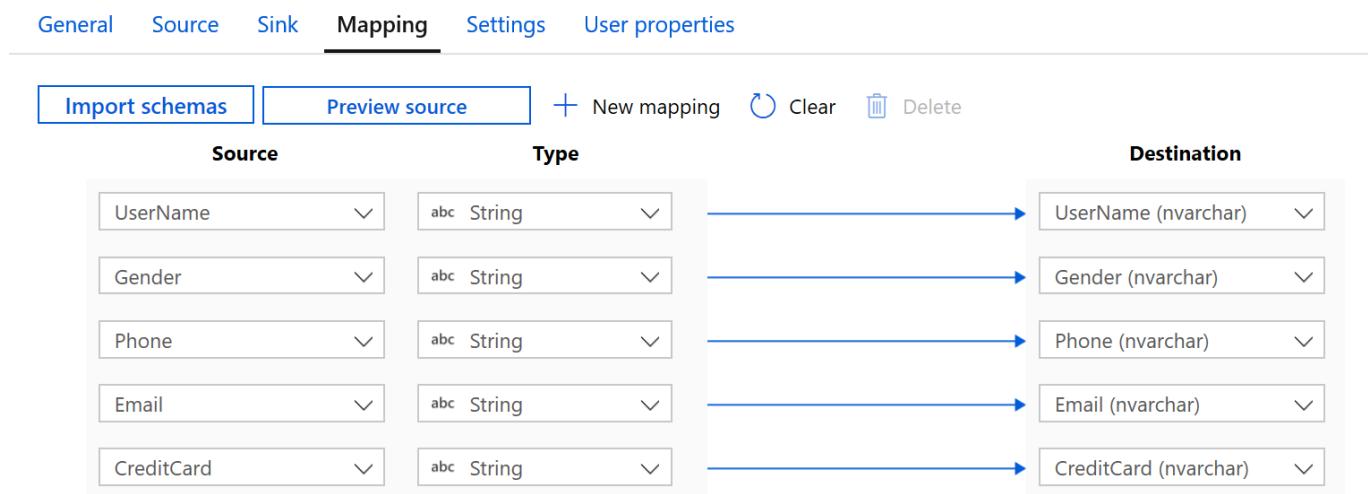
Table option None Auto create table ⓘ

Pre-copy script

truncate table dbo.WWICustomerInfo

Add dynamic content [Alt+P]

18. Go to Mapping tab, click Import Schema and review.



19. Publish All and Trigger now.

20. Check the result from the Monitor tab and run 11_02_Select_Customer_Info_Table to check result.

Task 10: Create & Populate Campaign Analytics Table

1. Select Develop tab and open & run 12_01_Create_Campaign_Analytics_Table script.
2. Go to Data tab and click + to create Data Set.
3. Select ADLS Gen2 and select Delimited Text.
4. Put property as below.

Field	Value
Name	Enter asamcw_campaignanalytics_csv
Linked service	Select <ADLS Gen2 Account Name>
File Path - Container	Enter wwi-02
File Path - Directory	Enter campaign-analytics
File Path - File	Enter campaignanalytics.csv
First row as header	Unchecked
Import schema	Select From connection/store

5. Click + to create Data Set.
6. Select **Azure Synapse Analytics (formerly SQL DW)**.

7. Type Property as below.

Field	Value
Name	Enter asamcw_campaignanalytics_asa
Linked service	Select WorkspaceDefault
Table name	Select WWICampaignAnalytics
Import schema	Select From connection/store

Connection Schema Parameters

Linked service * kyowonsynapse-WorkspaceDefaultS... Test connection Open New

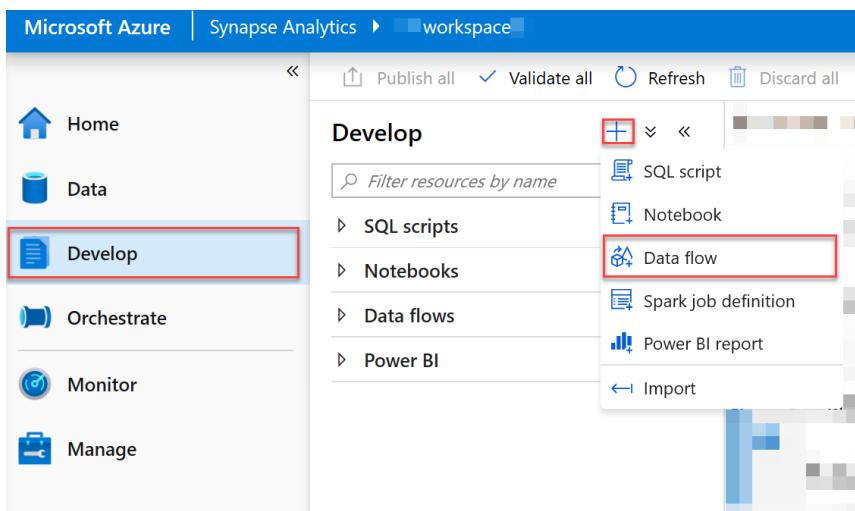
Linked service properties ⓘ

NAME	VALUE
DBName	sqlpool

Table dbo.WWICampaignAnalytics Preview data Edit

8. Publish All.

9. Back to Develop tab and click + to create Data Flow.



10. Put name as **Campaign_Analytics_Data**.

Properties

General

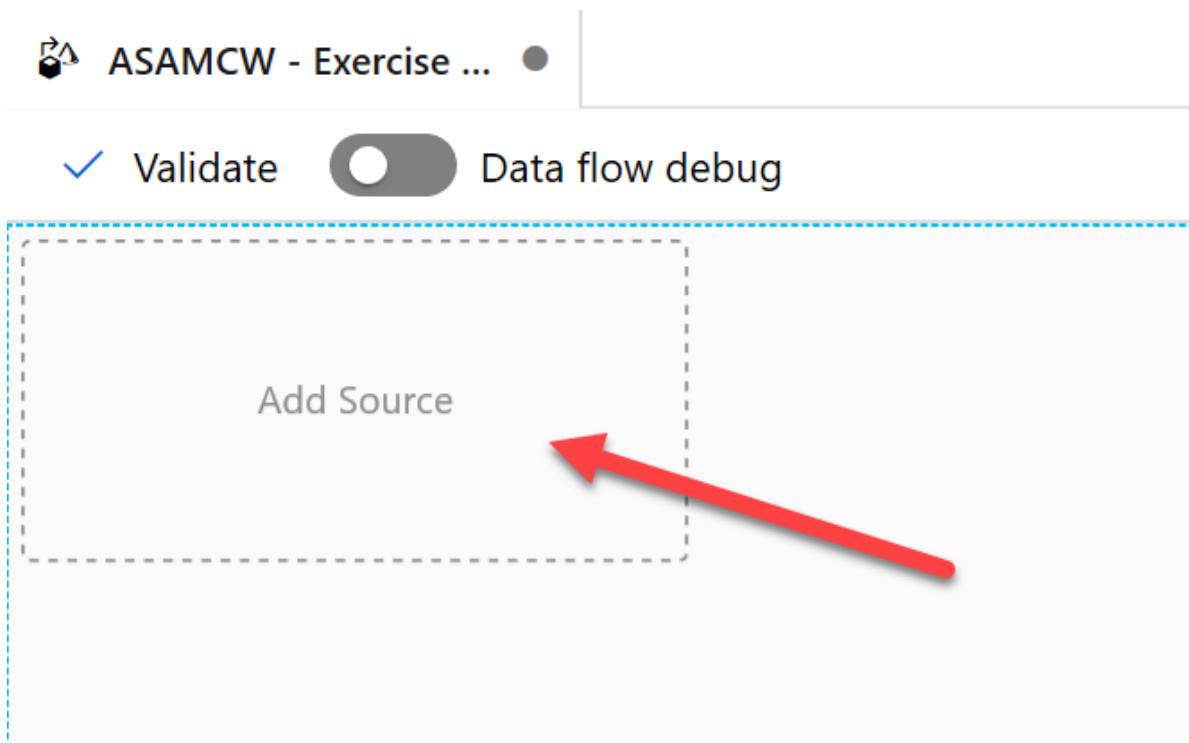
- i** Choose a name for your data flow.
This name can be updated at any time until it is published.

Name *

Campaign_Analytics_Data

Description

11. Click Add Source.



12. Type Property as below.

Field	Value
Output stream name	Enter campaignanalyticscsv
Source type	Select Dataset
Dataset	Select asamcw_campaignanalytics_csv .
Skip line count	Enter 1

The screenshot shows the 'Source settings' tab of a dataset configuration. Key fields highlighted with red boxes are:

- Output stream name ***: campaignanalyticscsv
- Dataset ***: asamcw_campaignanalytics_csv
- Skip line count**: 1

Other visible fields include:

- Source type ***: Dataset
- Options**:
 - Allow schema drift
 - Infer drifted column types
 - Validate schema
- Sampling ***: Disable

Buttons at the top right include **Learn more**, **Test connection**, **Open**, and **New**.

13. When creating Data Flow, you can set up debug such as data preview and schema import (projection) to use certain functions, but due to temporal and environmental constraints, this process is skipped. In the Data Source, there is a Schema that should be defined in general, and it can be configured as a Script in the Designer Toolbar of Data Flow.



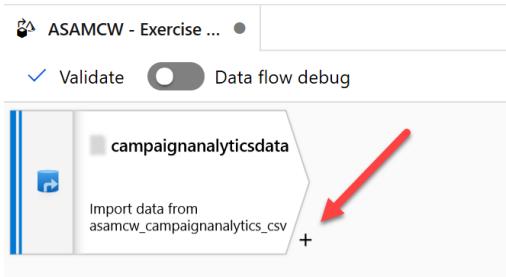
14. Copy below script and paste in the script.

```
source(output(
    {__col0__} as string,
    {__col1__} as string,
    {__col2__} as string,
    {__col3__} as string,
    {__col4__} as string,
    {__col5__} as double,
    {__col6__} as string,
    {__col7__} as double,
    {__col8__} as string,
    {__col9__} as string
),
allowSchemaDrift: true,
validateSchema: false,
skipLines: 1) ~> campaignanalyticscsv
```

15. Go to Projection tab and review.

Source settings	Source options	Projection	Optimize	Inspect	Data preview
Define default format		Detect data type	Import projection		Reset schema
Column name	Type	Format			
col0	abc string	Specify format			
col1	abc string	Specify format			
col2	abc string	Specify format			
col3	abc string	Specify format			
col4	abc string	Specify format			
col5	1.2 double	Specify format			
col6	abc string	Specify format			
col7	1.2 double	Specify format			
col8	abc string	Specify format			
col9	abc string	Specify format			

16. Click + and choose Select.



17. Put Output Stream Name as **mapcampaignanalytics**.

18. Under the Name as, type below as in order.

- Region
- Country
- ProductCategory
- CampaignName
- RevenuePart1
- Revenue
- RevenueTargetPart1
- RevenueTarget
- City
- State

Select settings Optimize Inspect Data preview

Output stream name * Learn more

Incoming stream *

Options Skip duplicate input columns
 Skip duplicate output columns

Input columns *

campaignanalyticscsv's column	Name as
abc_col0_	Region
abc_col1_	Country
abc_col2_	ProductCategory
abc_col3_	CampaignName
abc_col4_	RevenuePart1
1.2_col5_	Revenue
abc_col6_	RevenueTargetPart1
1.2_col7_	RevenueTarget
abc_col8_	City
abc_col9_	State

19. Click + and select Derived Column.
20. Type name as **convertandaddcolumns**.
21. Reference below and type correctly.

Column	Expression	Description
Revenue	toDecimal(replace(concat(toString(RevenuePart1), toString(Revenue)), '\\', ','), 10, 2, '\$###,###.##')	Concatenate the RevenuePart1 and Revenue fields, replace the invalid \ character, then convert and format the data to a decimal type.
RevenueTarget	toDecimal(replace(concat(toString(RevenueTargetPart1), toString(RevenueTarget)), '\\', ','), 10, 2, '\$###,###.##')	Concatenate the RevenueTargetPart1 and RevenueTarget fields, replace the invalid \ character, then convert and format the data to a decimal type.
Analyst	iif(isNull(City), '', replace('DataAnalyst' + City, ' ', ''))	If the city field is null, assign an empty string to the Analyst field, otherwise concatenate DataAnalyst to the City value, removing all spaces.

Derived column's settings [Optimize](#) [Inspect](#) [Data preview](#)

Output stream name * [Learn more](#)

Incoming stream *

Columns * [①](#)

<input type="text" value="Revenue"/>	<input type="text" value="toDecimal(replace(concat(toString(RevenuePart1), toString(Revenue)), '\\', ','), 10, 2, '\$###,###.##')"/>	+ D B
<input type="text" value="RevenueTarget"/>	<input type="text" value="toDecimal(replace(concat(toString(RevenueTargetPart1), toString(RevenueTarget)), '\\', ','), 10, 2, '\$###,###.##')"/>	+ D B
<input type="text" value="Analyst"/>	<input type="text" value="iif(isNull(City), '', replace('DataAnalyst' + City, ' ', ''))"/>	abc + D B

22. Click + and choose Select.

23. Type name as **selectcampaignanalyticscolumns**, then remove **RevenuePart1** and **RevenueTargetPart1**.

Input columns *	Name as
convertandaddcolumns's column	
abc Region	Region
abc Country	Country
abc ProductCategory	ProductCategory
abc CampaignName	CampaignName
e* Revenue	Revenue
e* RevenueTarget	RevenueTarget
abc City	City
abc State	State
abc Analyst	Analyst

24. Finally click + and choose Sink.

25. Put name as **campaignanalyticsasa** and click +New in the Destination Data Set. Now choose ADLS Gen2 with Parquet.

Format
Avro
DelimitedText
Json
Parquet
Binary
ORC

26. Type name as **asamcw_campaignanalytics_parquet** and set Property as below.

Set properties

i Choose a name for your dataset. This name can be updated at any time until it is published.

Name

Linked service *

File path

 / /  | 

Import schema

 From connection/store From sample file None Advanced

[Sink](#) [Settings](#) [Mapping](#) [Optimize](#) [Inspect](#) [Data preview](#)

Output stream name *	<input type="text" value="campaignanalyticsasa"/> Learn more
Incoming stream *	<input type="text" value="selectcampaignanalyticscolumns"/> 
Sink type *	<input type="text" value="Dataset"/> 
Dataset *	 <input type="text" value="asamcw_campaignanalytics_parquet"/>   Test connection  Open  New
Options	<input checked="" type="checkbox"/> Allow schema drift  <input type="checkbox"/> Validate schema 

27. Go to Setting tab, put **Output to one single file** in File Name Option. Type file name as **campaignanalytics.parquet**.

Sink Settings Mapping Optimize Inspect Data preview

Clear the folder Add dynamic content [Alt+P]

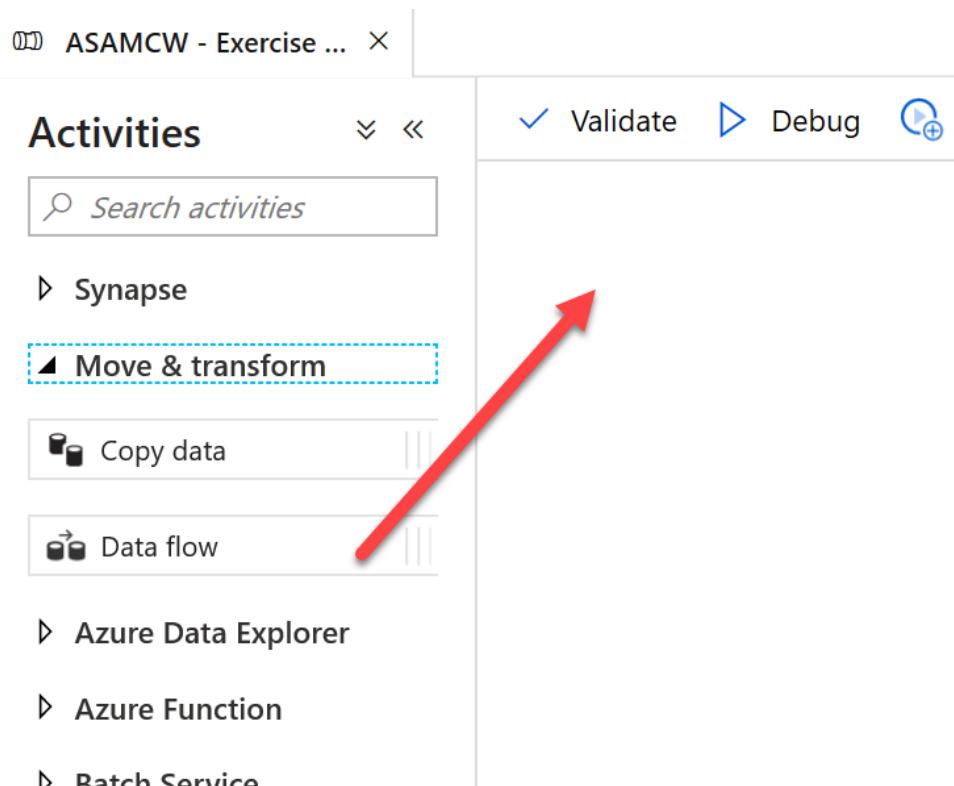
File name option * Default Pattern Per partition As data in column Output to single file

Output to single file * campaignanalytics.parquet ⓘ

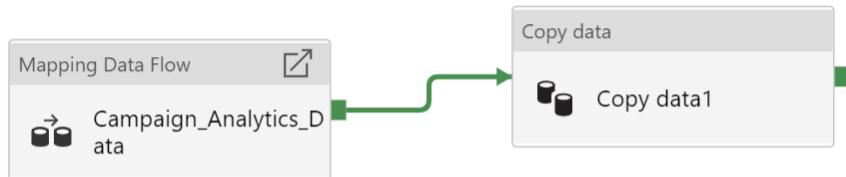
28. Publish All.

29. Create pipeline in the Orchestrate tab and name it as **Copy Campaign Analytics Data**.

30. Drag & Drop Data Flow and choose you just created.



31. Also Drag & Drop Copy Data and name it as below.



32. Select Copy Data and go to Source tab. Select **asamcw_campaignanalytics_parquet** as Source Data Set. Select Wildcard file Path and type campaign-analytics.

General **Source** Sink Mapping Settings User properties

Source dataset *

File path type File path in dataset Wildcard file path List of files ⓘ

Wildcard paths wwi-02 / campaign-analytics / *.parquet

Start time (UTC) End time (UTC)

Filter by last modified

Recursively ⓘ

Enable partition discovery

Max concurrent connections

33. In Sink tab, select **asamcw_campaignanalytics_asa** and select Bulk Insert.

General Source **Sink** Mapping Settings User properties

Sink dataset *

Copy method PolyBase ⓘ Copy command (Preview) ⓘ Bulk insert

Table option None Auto create table ⓘ

Pre-copy script

34. Publish All.

35. Trigger Now and check status in the Monitoring tab.
36. Run 12_02_Count_Campaign_Analytics_Table script when pipeline is finished.

Task 11: Exploring Raw Data

1. 해 새 솔 data tab and right click **sale-**

small/Year=2010/Quarter=Q4/Month=12/Day=20101231 file. Then click Select Top 100 rows.

The screenshot shows the Azure Data Explorer interface. On the left, there's a navigation pane with sections like Storage accounts, Cosmos DB, and Datasets. Under Storage accounts, 'kyowonsynapse' is expanded, showing 'cosmosdemo', 'models', 'scaleanalytics', and 'stage'. A folder named 'wwi-02' is selected and highlighted with a red box. In the main area, a file named 'sale-small-20101231' is selected and also highlighted with a red box. A context menu is open over this file, with the 'Select TOP 100 rows' option highlighted with another red box. The menu also includes options like New SQL script, New notebook, Copy ABFSS path, Manage access..., Rename..., Download, Delete, and Properties... At the top of the interface, there's a breadcrumb navigation bar showing the full path: wwi-02 > sale-small > Year=2010 > Quarter=Q4 > Month=12 > Day=20101231.

2. Check connection with SQL on-demand.

```

Run Undo Publish Query plan Connect to SQL on-demand Use database master
1 SELECT
2     TOP 100 *
3 FROM
4     OPENROWSET(
5         BULK 'https://kyowonadlsgen2.dfs.core.windows.net/wwi-02/sale-small/Year=2010/Quarter=Q4/Month=12/Day=20101231/sale-small-20101231-snappy.parquet',
6         FORMAT='PARQUET'
7     ) AS [r];
8

```

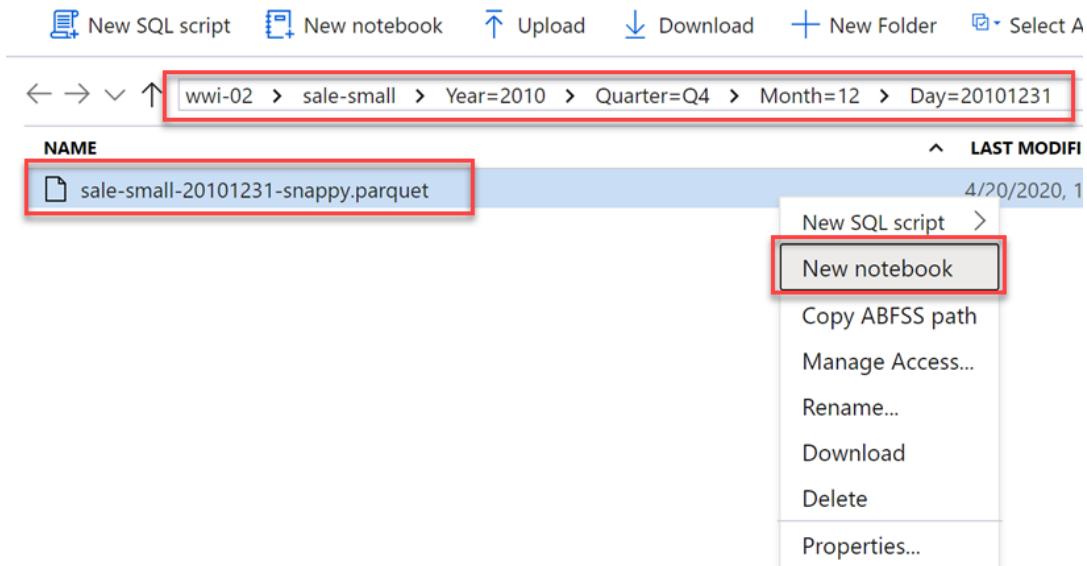
3. Modify query as below and run.

```

SELECT
    TransactionDate, ProductId,
    CAST(SUM(ProfitAmount) AS decimal(18,2)) AS [(sum) Profit],
    CAST(AVG(ProfitAmount) AS decimal(18,2)) AS [(avg) Profit],
    SUM(Quantity) AS [(sum) Quantity]
FROM
    OPENROWSET(
        BULK 'https://<ADLS Gen2 Account Name>.dfs.core.windows.net/wwi-02/sale-
small/Year=2010/Quarter=Q4/Month=12/Day=20101231/sale-small-20101231-snappy.parquet',
        FORMAT='PARQUET'
    ) AS [r] GROUP BY r.TransactionDate, r.ProductId;

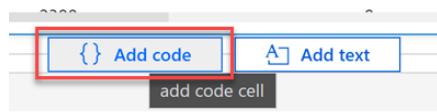
```

4. Right click same file and click New notebook.



5. Run All.

6. Click Add Code and type below to check schema.



```
data_path.printSchema()
```

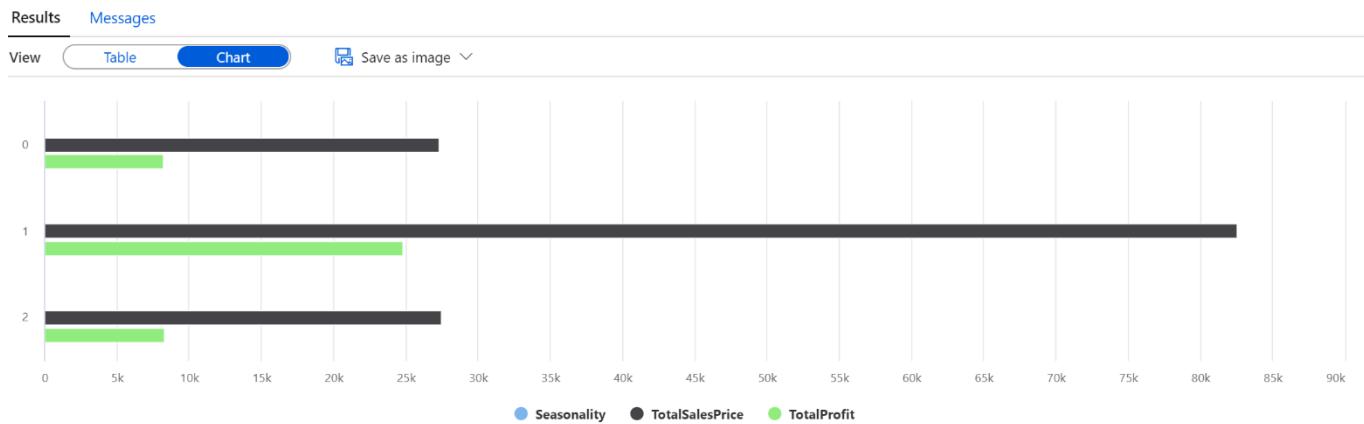
7. Click Add Code and type below to compare result with previous one.

```
from pyspark.sql import SparkSession
from pyspark.sql.types import *
from pyspark.sql.functions import *

profitByDateProduct = (data_path.groupBy("TransactionDate", "ProductId")
    .agg(
        sum("ProfitAmount").alias("(sum)ProfitAmount"),
        round(avg("Quantity"), 4).alias("(avg)Quantity"),
        sum("Quantity").alias("(sum)Quantity"))
    .orderBy("TransactionDate"))
profitByDateProduct.show(100)
```

8. Open 13_01_Select_From_CSV ~ 13_03_Select_From_CSV SQL script and run.

9. Change result to Chart to visualize result as below.



Task 12: Azure Synapse Analytics Security

1. Open 14_01_Column_Level_Security SQL script in Develop Tab and follow the guide in the script.
2. Open 14_02_Row_Level_Security SQL script in Develop Tab and follow the guide in the script.
3. Open 14_03_Dynamic_Data_Masking SQL script in Develop Tab and follow the guide in the script.

Task 13: Spark Pool Library Install

4. [Azure Portal](#)로 돌아가서 Synapse Workspace의 sparkpool을 선택합니다.
5. Sparkpool의 왼쪽 Package 탭에서 Upload Environment Config File을 선택합니다.

sparkpool (kyowonsynapse/sparkpool) | Packages

Apache Spark pool

Search (Ctrl+ /) <> Upload environment config file Refresh

Overview Activity log Access control (IAM) Tags

Settings

Packages Properties Locks

User-provisioned packages

No user-provided packages currently uploaded. You can upload "environment config file".

Upload environment config file

6. 깃허브에서 다운로드 받은 Synapse_HOL\Synapse_HOL\LabFiles\Scripts\Notebooks\Env_Config 안에 있는 Environment_Config.txt 파일을 선택하고, 아래 체크박스에 체크하고 업로드합니다.

Force new settings on the Apache Spark pool (will immediately stop running Apache Spark applications).

Upload

Task 14: Azure Synapse Studio 의 Synapse Link 생성

1. Azure Synapse Workspace Overview Page 에서 아래와 같이 Link 를 클릭하여 Synapse Studio 를 실행합니다.

The screenshot shows the Azure Synapse Workspace Overview Page. At the top right, there is a 'Launch Synapse Studio' button with a red box drawn around it. Below the button, there is a list of workspace details including Resource group, Status, Location, Subscription, Subscription ID, Managed virtual network, Managed Identity object, Workspace web URL (which is also highlighted with a red box), and Tags. Under 'Available resources', there are sections for SQL pools and Apache Spark pools, each listing one pool: 'sqlpool' and 'sparkpool' respectively. The 'Type' column indicates they are both 'SQL pool' and 'Apache Spark pool' respectively.

2. Data 탭을 선택하고 + 를 클릭하여 External Data Source 를 연결합니다.

The screenshot shows the Microsoft Azure Synapse Analytics workspace for the 'kyowonsynapse' workspace. On the left, there is a sidebar with icons for Home, Storage accounts, Pipelines, and Datasets. The main area has a 'Data' tab selected, indicated by a blue bar. A context menu is open at the top right of the 'Data' tab, with the 'Connect to external data' option highlighted. Other options in the menu include 'Publish all', 'Validate all', 'Refresh', and 'Discard all'. Below the 'Data' tab, there are tabs for 'Workspace' and 'Linked'. A search bar says 'Filter resources by name' and a link to 'Storage accounts'.

3. Azure Cosmos DB(SQL API)를 선택합니다.

Connect to external data

Connect to supported data sources to explore them in the Data hub and make them available to pipelines as linked services.



4. 이름에 CosmosDemo로 입력하고 생성한 DB를 선택하여 연결합니다.

New linked service (Azure Cosmos DB (SQL API))

i Choose a name for your linked service. This name cannot be used by another linked service.

Name *
CosmosDemo

Description

Connect via integration runtime *
AutoResolveIntegrationRuntime

Connection string **Azure Key Vault**

Account selection method
 From Azure subscription Enter manually

Azure subscription
Microsoft Azure Internal Consumption (2386ed0e-ac2f-41c7-958a-

Cosmos DB account name *
kyowoncosmos

Database name *
CosmosDemo

Additional connection properties

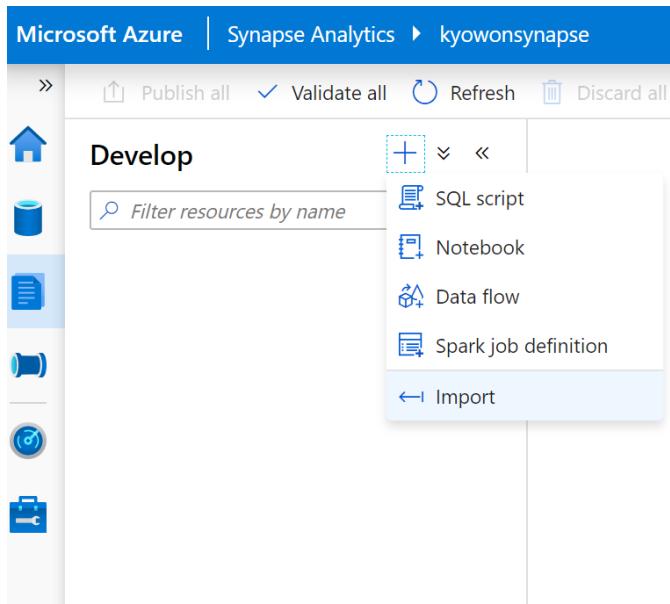
5. Refresh 하여 생성된 Link 를 확인합니다.

The screenshot shows the Azure Data workspace interface. At the top, there are buttons for Publish all, Validate all, Refresh, and Discard all. Below that, a navigation bar has 'Data' selected and tabs for 'Workspace' and 'Linked'. A search bar says 'Filter resources by name'. The main area lists resources under 'Linked': 'Storage accounts' (1 item), 'Cosmos DB' (1 item), and 'CosmosDemo (CosmosDemo)' (6 items). The 'CosmosDemo' item is expanded, showing 'RetailSales', 'IoTSignals', 'IoTDeviceInfo', 'RetailProducts', 'IoTStreamingSignals', and 'RetailStoreDemographics'. The 'Actions' button next to the CosmosDB entry is highlighted with a blue dashed box.

Category	Count
Storage accounts	1
Cosmos DB	1
CosmosDemo (CosmosDemo)	6

Task 15: Azure Synapse Workspace Spark Scenario

10. Go to Develop tab, click + and then click Import.



11. Import all notebooks in Synapse_HOL\#Synapse_HOL\#LabFiles\#Scripts\#Notebooks.

12. Publish All.

13. Open 00_CheckLibraries and run to check installed libraries.

A screenshot of a Jupyter Notebook interface. At the top, there's a toolbar with 'Cell', 'Run all', 'Undo', 'Publish', 'Attach to' (set to 'sparkpool'), 'Language' (set to 'PySpark (Python)'), and a refresh icon. Below the toolbar, a cell is titled 'Cell 1'. The code in the cell is:

```
1 import pip #needed to use the pip functions
2 for i in pip.get_installed_distributions(local_only=True):print(i)
```

The output of the cell shows the following list of installed packages:

```
zipp 3.1.0
zict 1.0.0
xlwt 1.2.0
XlsxWriter 0.9.6
xlrd 1.0.0
wrapt 1.11.2
widgetsnbextension 3.5.1
wheel 0.30.0
Werkzeug 0.16.1
websocket-client 0.57.0
webencodings 0.5.1
wcwidth 0.2.4
```

A note at the bottom of the output says 'Command executed in 7mins 19s 191ms by yongl on 06-23-2020 09:52:44.072 +09:00'.

14. In [Azure Portal](#), go to the Cosmos DB in Resource Group.

Showing 1 to 5 of 5 records. Show hidden types ⓘ

Name ↑↓	Type ↑↓
kyowonadlsgen2	Storage account
kyowoncosmos	Azure Cosmos DB account
kyowonsynapse	Synapse workspace
sparkpool (kyowonsynapse/sparkpool)	Apache Spark pool
sqlpool (kyowonsynapse/sqlpool)	SQL pool

15. Go to Data Explorer and check data of collections in CosmosDemo DB.

kyowoncosmos | Data Explorer

Azure Cosmos DB account

Search (Ctrl+ /) | New Container | Enable Notebooks (Preview)

Overview | Activity log | Access control (IAM) | Tags | Diagnose and solve problems | Quick start | Notifications | Data Explorer | Settings | Features | Replicate data globally | Default consistency

SQL API

CosmosDemo

- Scale
- IoTDeviceInfo
- IoTSignals
- Items
- Settings
- Stored Procedures
- User Defined Functions
- Triggers
- IoTStreamingSignals
- RetailProducts
- RetailSales
- RetailStoreDemographics

16. In Develop tab, open 01_CosmosDB_Configuration and change <ADLS Gen2 Account Name> properly and run all.

17. Come back to Cosmos DB's Data Explorer and check Data Ingestion.

The screenshot shows the Azure Cosmos DB Data Explorer interface. On the left, there is a navigation pane for the 'CosmosDemo' database, with 'IoTSignals' selected. The main area displays a table titled 'Items' with two columns: 'id' and '/id'. One row is highlighted with a yellow box, and its details are shown in a modal window on the right. The modal contains the following JSON data:

```

1  {
2    "dateTime": "2020-04-28 17:40:54",
3    "measureValue": "4322",
4    "unit": "Revolutions per Minute",
5    "unitSymbol": "RPM",
6    "id": "dca4c4d9-d503-4298-a1ce-850469b525d3",
7    "deviceId": "dev-2",
8    "measureType": "Rotation Speed",
9    "_rid": "ePYRAJEonGcFAAAAAAAA==",
10   "_self": "dbs/ePYRAJEonGc=/colls/ePYRAJEonGc=/docs/ePYRAJEonGcFAAAAAAAA==/",
11   "_etag": "\\"1000f824-0000-2500-0000-5ef146c60000\\\"",
12   "_attachments": "attachments/",
13   "_ts": 1592870598
14 }

```

18. Open 02_IoT_Data_Exploration and run until Cell 4.

1. Create Spark tables pointing to the Azure Cosmos DB Analytical Store collections using Azure Synapse Link

Cell 2

```
[ ] 1 %%sql
2 create database CosmosDemoIoT
```

Cell 3

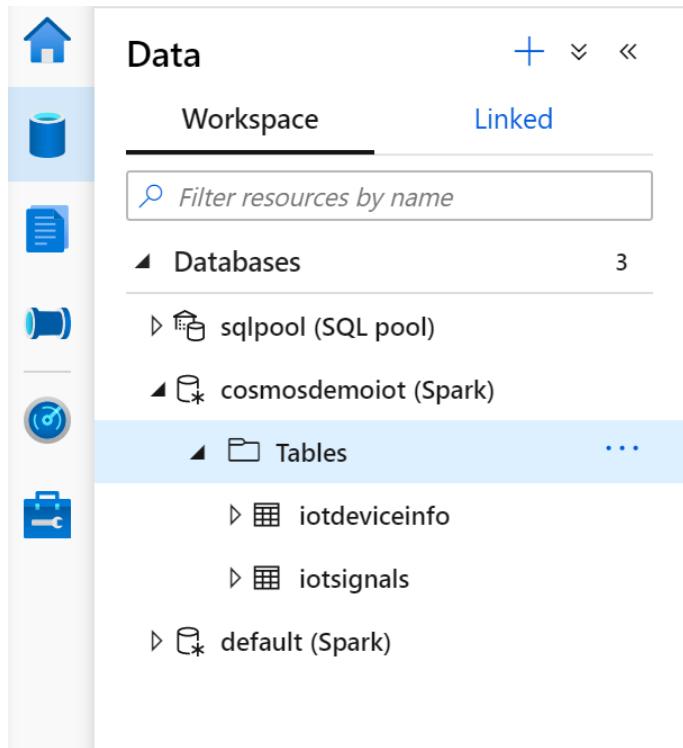
```
[ ] 1 %%sql
2
3 create table if not exists CosmosDemoIoT.IoTSignals
4 using cosmos.olap
5 options(spark.synapse.linkedService 'CosmosDemo',
6 || spark.cosmos.container 'IoTSignals')
```

Cell 4

```
Cell 4 [ ] 1 %%sql
2
3 create table if not exists CosmosDemoIoT.IoTDeviceInfo
4 using cosmos.olap
5 options(spark.synapse.linkedService 'CosmosDemo',
6 || spark.cosmos.container 'IoTDeviceInfo')
```

19. Go to Data tab and check that CosmosDemoloT DB is created in Workspace.

Check tables as well.



The screenshot shows the Databricks Data workspace. On the left, there's a sidebar with icons for Home, Databases, Workspaces, Jobs, and More. The main area has tabs for Data and Workspace, with the Data tab selected. Under the Data tab, there's a search bar labeled "Filter resources by name". Below it, there's a section for "Databases" which lists "sqlpool (SQL pool)", "cosmosdemoliot (Spark)" (which is expanded to show "Tables" like "iotdeviceinfo" and "iotsignals"), and "default (Spark)". To the right, there's a notebook titled "02_IoT_Data_Explora...". The first cell, "Cell 2", contains the following SQL code:

```

1 %%sql
2 create database CosmosDemoIoT

```

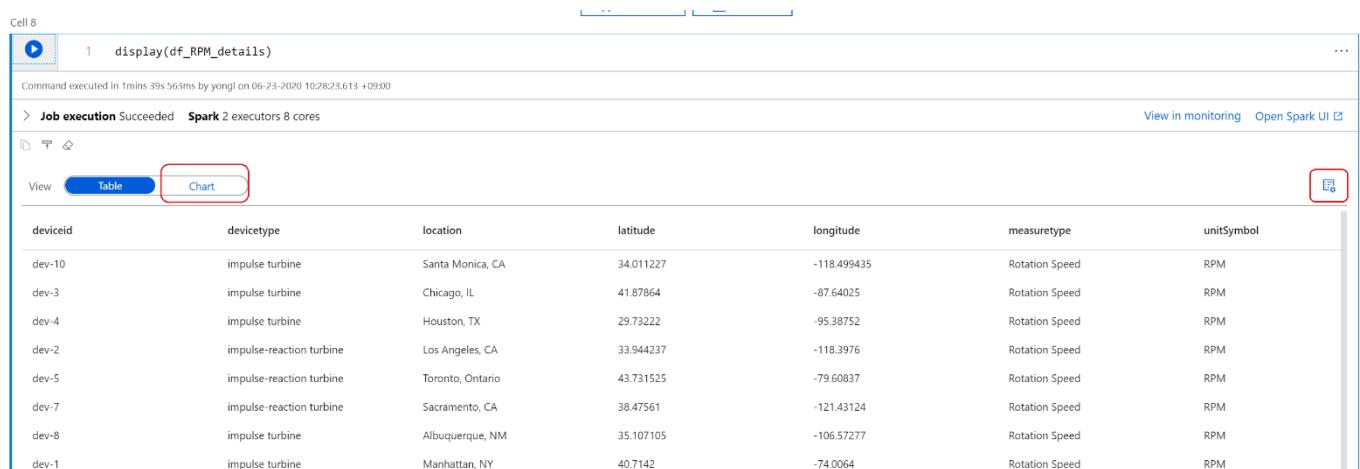
The second cell, "Cell 3", contains the following SQL code:

```

1 %%sql
2
3 create table if not exists (
4     using cosmos.olap

```

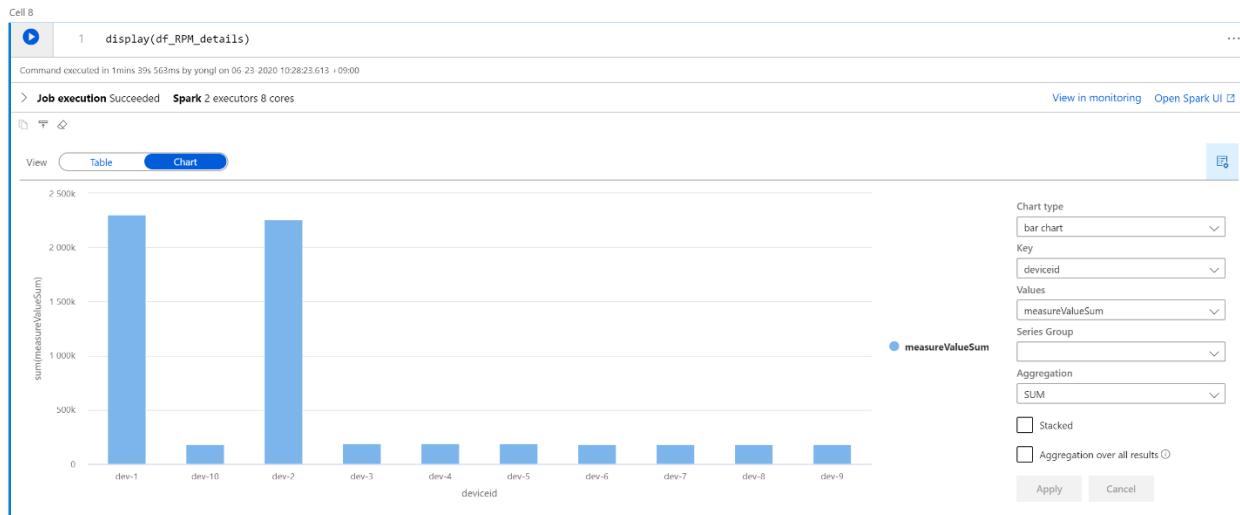
20. Run until Cell 8 and check result. Click chart on the cell result and select measurevaluesum in Value field.



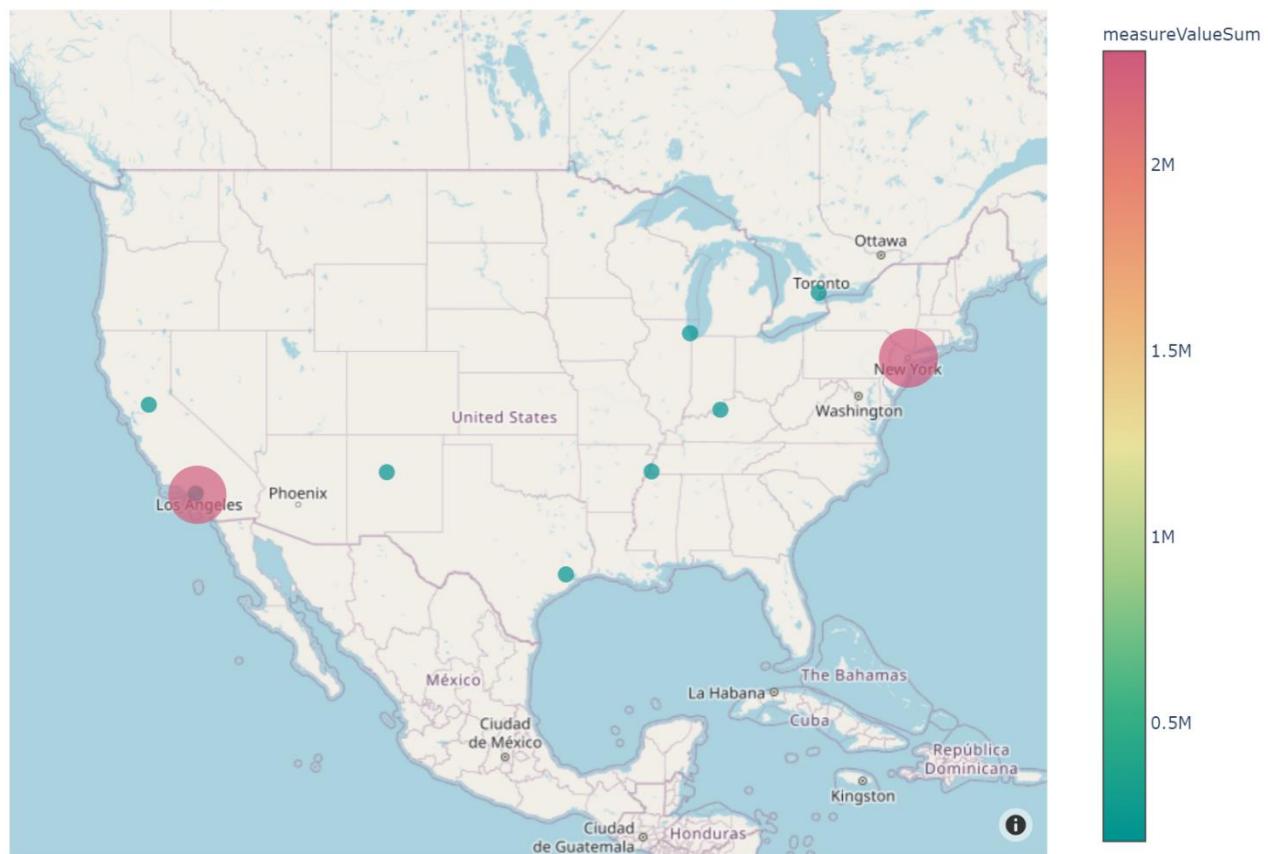
The screenshot shows the results of Cell 8. The cell code was "display(df_RPM_details)". The output shows the command was executed in 1min 39s 563ms by yongli on 06-23-2020 10:28:23.613 +0900. It indicates "Job execution Succeeded" with "Spark 2 executors 8 cores". There are links to "View in monitoring" and "Open Spark UI". Below the cell, there's a table view with a "Chart" button highlighted with a red box. The table data is as follows:

deviceid	devicetype	location	latitude	longitude	measuretype	unitSymbol
dev-10	impulse turbine	Santa Monica, CA	34.011227	-118.499435	Rotation Speed	RPM
dev-3	impulse turbine	Chicago, IL	41.87864	-87.64025	Rotation Speed	RPM
dev-4	impulse turbine	Houston, TX	29.73222	-95.38752	Rotation Speed	RPM
dev-2	impulse-reaction turbine	Los Angeles, CA	33.944237	-118.3976	Rotation Speed	RPM
dev-5	impulse-reaction turbine	Toronto, Ontario	43.731525	-79.60837	Rotation Speed	RPM
dev-7	impulse-reaction turbine	Sacramento, CA	38.47561	-121.43124	Rotation Speed	RPM
dev-8	impulse turbine	Albuquerque, NM	35.107105	-106.57277	Rotation Speed	RPM
dev-1	impulse turbine	Manhattan, NY	40.7142	-74.0064	Rotation Speed	RPM

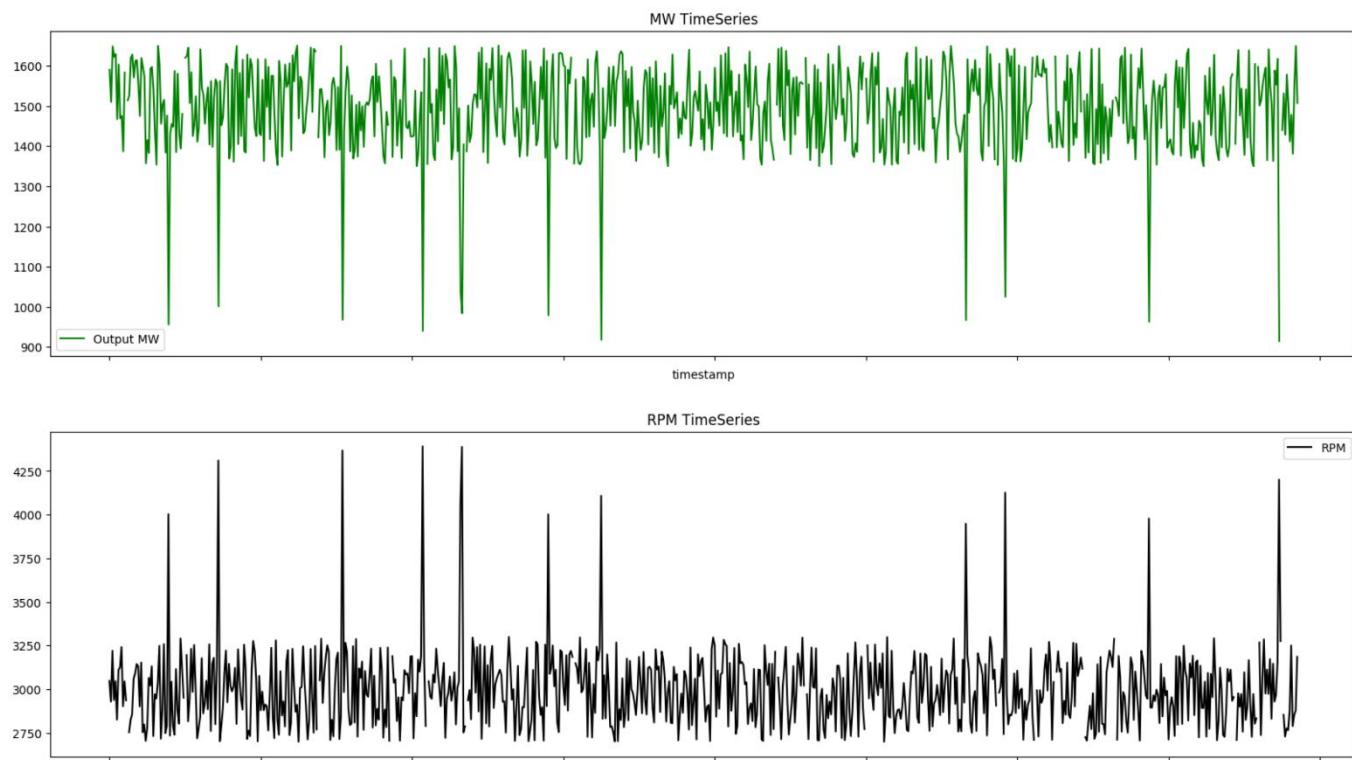
21. Now you can see IoT Device #1 and #2 has high MeasureValueSum.



22. Run Cell 10 to gather Geographic Insight based on device.



23. Open 03_IoT_Data_AnomalyDetection script and run until Cell 4 to one of high value device. You choose Dev-1 and check data based on unitSymbol value which has MW and RPM.

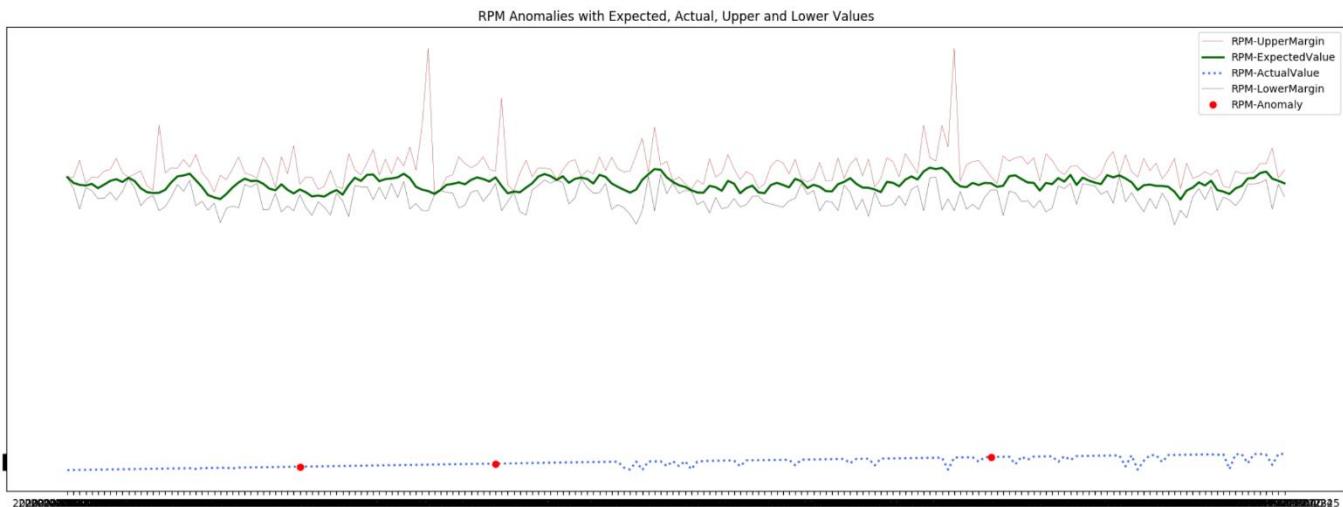


24. Change <Azure Anomaly Detector End Point> properly in Cell 6 and run until Cell 7 to send Anomaly Detector all the device info which has unitSymbol value is RPM.

Now check the response from Anomaly Detector in result.

Cell 7							
	1	display(dt_anomaly)					...
Command executed in 7s 772ms by yongl on 06-23-2020 10:52:20.111 +09:00							
> Job execution Succeeded	Spark	2 executors 8 cores					View in monitoring Open Spark UI
View	Table	Chart					
deviceId	measureType	.etag	value	grouping	SimpleDetectAnomalies_97521124...	anomalies	
"b65c...	dev-2	Rotation Speed	"1000952f-0000-2500-0000-5ef146...	3062	dev-2		▼ {"isAnomaly":false,"isPositiveAnomaly":false,"isNegativeAnomaly":false,"period":0,"expectedValue":3062,"upperMargin":5.980000000000001,"lowerMargin":5.980000000000001}
7415...	dev-2	Rotation Speed	"1000e931-0000-2500-0000-5ef146...	3169	dev-2		▶ {"isAnomaly":false,"isPositiveAnomaly":false,"isNegativeAnomaly":false,"period":0,"expectedValue":3169,"upperMargin":5.980000000000001,"lowerMargin":5.980000000000001}
j3281...	dev-2	Rotation Speed	"10003d32-0000-2500-0000-5ef146...	2795	dev-2		▶ {"isAnomaly":false,"isPositiveAnomaly":false,"isNegativeAnomaly":false,"period":0,"expectedValue":2795,"upperMargin":5.980000000000001,"lowerMargin":5.980000000000001}
lb4b...	dev-2	Rotation Speed	"10009928-0000-2500-0000-5ef146...	2848	dev-2		▶ {"isAnomaly":false,"isPositiveAnomaly":false,"isNegativeAnomaly":false,"period":0,"expectedValue":2848,"upperMargin":5.980000000000001,"lowerMargin":5.980000000000001}

25. Run Cell 9 to Visualize data.



26. Open 04_Retail_Data_AutoML and run until Cell 4.

27. Go to Data tab and check that CosmosDemoRetail DB and tables are created.

Data

Workspace Linked

Filter resources by name

- ▲ Databases 4
 - ▷ 🗁 sqlpool (SQL pool)
 - ▷ 🗁 cosmosdemoint (Spark)
 - ▲ 🗁 cosmosdemoretail (Spark)
 - ◀ Tables
 - ▷ 🗂 retailproducts
 - ▷ 🗂 retailsales
 - ▷ 🗂 retailstoredemographics
 - ▷ 🗁 default (Spark)

04_Retail_Data_Auto... ×

+ Cell ▾ ▶ Run all ⏪ Undo | ▾ ⌟ Publish Att

1. Create Spark tables pointing to the Azure Cosmos

Cell 2

```
[ ] 1 %%sql
      2 create database CosmosDemoRetail
```

Cell 3

```
[ ] 1 %%sql
      2
      3 create table if not exists CosmosD
      4 using cosmos.olap
      5 options(spark.synapse.linkedService
      6 | | spark.cosmos.container 'Re
```

28. Run Cell 5 for data exploration.
29. Put proper information in Cell 7 to configure Azure ML Workspace environment.
30. Run Cell 9 for Data Prep.
31. Run Cell 11to run Azure Auto ML and check the result. It will takes 30mins.
32. Go to the Azure ML Workspace in Azure Portal to check result in there as well.

```

Running on local machine
Parent Run ID: AutoML_141283ae-566c-4259-b588-191b3292f3c4

Current status: DatasetFeaturization. Beginning to featurize the dataset.
Current status: DatasetFeaturizationCompleted. Completed featurizing the dataset.
Current status: DatasetCrossValidationSplit. Generating individually featurized CV splits.
Current status: DatasetFeaturization. Beginning to featurize the CV split.
Current status: DatasetFeaturizationCompleted. Completed featurizing the CV split.
Current status: DatasetFeaturization. Beginning to featurize the CV split.
Current status: DatasetFeaturizationCompleted. Completed featurizing the CV split.
Current status: DatasetFeaturization. Beginning to featurize the CV split.
Current status: DatasetFeaturizationCompleted. Completed featurizing the CV split.

*****
DATA GUARDRAILS:

TYPE: Frequency detection
STATUS: PASSED
DESCRIPTION: The time series was analyzed, all data points are aligned with detected frequency.

TYPE: Missing feature values imputation
STATUS: PASSED
DESCRIPTION: No feature missing values were detected in the training data.
Learn more about missing value imputation: https://aka.ms/AutomatedMLFeaturization

*****
Current status: ModelSelection. Beginning model selection.

*****
ITERATION: The iteration being evaluated.
PIPELINE: A summary description of the pipeline being evaluated.
DURATION: Time taken for the current iteration.
METRIC: The result of computing score on the fitted pipeline.
BEST: The best observed score thus far.

*****
```

ITERATION	PIPELINE	DURATION	METRIC	BEST
0	StandardScalerWrapper ElasticNet	0:00:20	0.0310	0.0310
1	StandardScalerWrapper ElasticNet	0:00:17	0.0324	0.0310
2	StandardScalerWrapper ElasticNet	0:00:19	0.0308	0.0308
3	StandardScalerWrapper ElasticNet	0:00:15	0.0310	0.0308
4	StandardScalerWrapper LassoLars	0:00:19	0.0323	0.0308
5	StandardScalerWrapper ElasticNet	0:00:21	0.0304	0.0304
6	StandardScalerWrapper RandomForest	0:01:37	0.0262	0.0262
7	StandardScalerWrapper ExtremeRandomTrees	0:00:14	0.0184	0.0184
8	StandardScalerWrapper DecisionTree	0:00:14	0.0040	0.0040
9	MinMaxScaler GradientBoosting	0:00:14	0.0215	0.0040
10	MinMaxScaler DecisionTree	0:00:14	0.0078	0.0040
11	RobustScaler LassoLars	0:00:15	0.0325	0.0040
12	MaxAbsScaler LightGBM	0:00:15	0.0121	0.0040
13	StandardScalerWrapper ElasticNet	0:00:14	0.0305	0.0040
14	MinMaxScaler RandomForest	0:00:23	0.0024	0.0024
15	StandardScalerWrapper LassoLars	0:01:41	0.0321	0.0024
16	StandardScalerWrapper ElasticNet	0:00:19	0.0325	0.0024
17	StandardScalerWrapper DecisionTree	0:00:14	0.0048	0.0024
18	MinMaxScaler DecisionTree	0:00:14	0.0157	0.0024
19	RobustScaler DecisionTree	0:00:19	0.0061	0.0024
20	StandardScalerWrapper RandomForest	0:00:31	0.0097	0.0024
21	RobustScaler RandomForest	0:00:23	0.0057	0.0024
22	MinMaxScaler LightGBM	0:00:32	0.0176	0.0024
23	MaxAbsScaler GradientBoosting	0:00:29	0.0115	0.0024
24	RobustScaler ExtremeRandomTrees	0:00:35	0.0106	0.0024
25	RobustScaler RandomForest	0:00:21	0.0160	0.0024
26	MaxAbsScaler LightGBM	0:00:20	0.0220	0.0024
27	StandardScalerWrapper ExtremeRandomTrees	0:00:20	0.0167	0.0024
28	MaxAbsScaler LightGBM	0:00:15	0.0214	0.0024
29	StandardScalerWrapper LightGBM	0:00:20	0.0102	0.0024
30	VotingEnsemble	0:03:12	0.0024	0.0024

33. Run Cell 13 to get result of Auto ML and run Prediction with Test Data.
34. Run Cell15 for Plotting.

[Test data scores]

```
root_mean_squared_error: 11.958
normalized_root_mean_squared_error: 0.007
normalized_root_mean_squared_log_error: 0.003
r2_score: 0.998
root_mean_squared_log_error: 0.015
explained_variance: 0.998
median_absolute_error: 0.426
mean_absolute_error: 3.304
mean_absolute_percentage_error: 0.862
normalized_mean_absolute_error: 0.002
spearman_correlation: 1.000
normalized_median_absolute_error: 0.000
```

