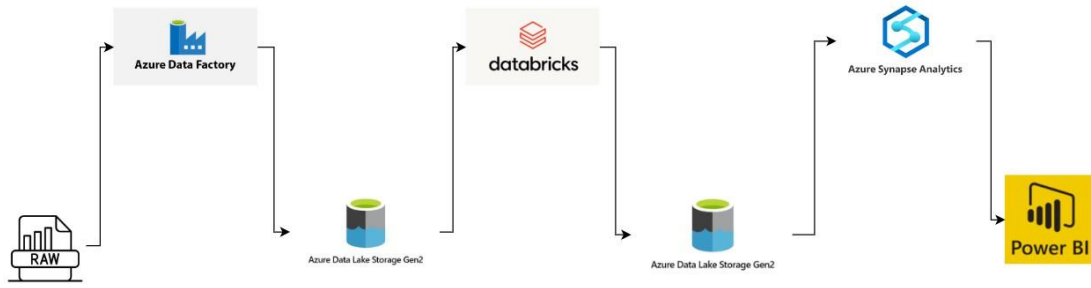


AZURE DATA PIPELINE



Azure End-to-End Data Pipeline

In this project I have tried to build an End-to End Data Pipeline using Microsoft Azure Services. The resources deployed in this project are

Raw data (Provided on GitHub)

Azure Data Factory: For streamlining the pipeline

Azure Data Lake Gen2: To store the data

Databricks: To perform data transformations using PySpark

Azure Synapse Analytics: For data warehousing and querying

Power BI: For data visualization and dashboarding

The GitHub Link to the Project- <https://github.com/pranav1001/Azure-End-to-End-Data-Pipeline->

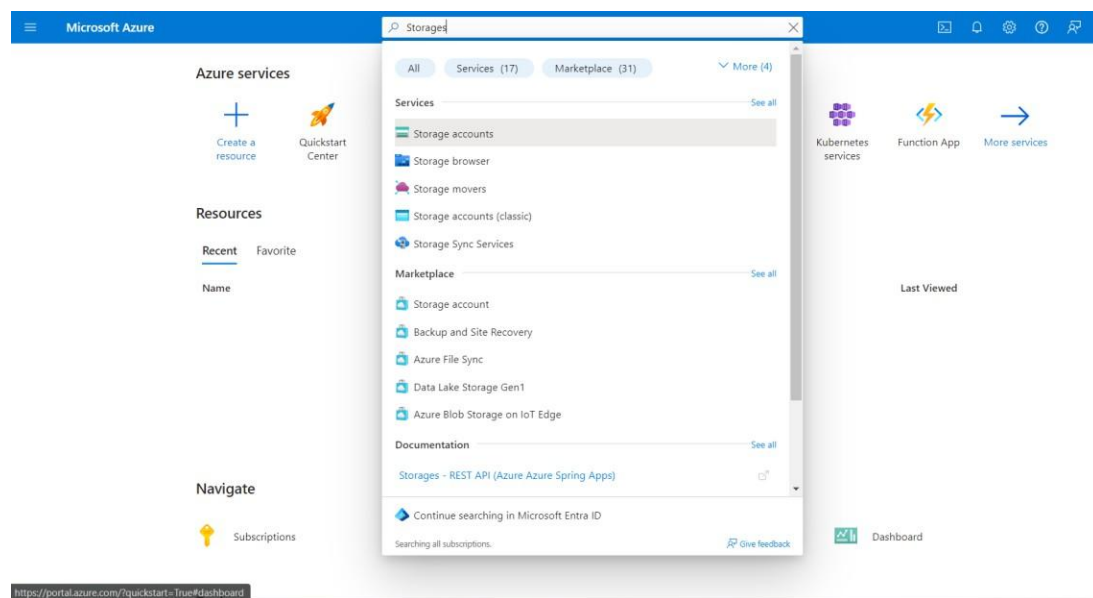
Dataset -

This dataset contains the details of over 11,000 athletes, with 47 disciplines, along with 743 Teams taking part in the 2021(2020) Tokyo Olympics. This dataset contains the details of the Athletes, Coaches, Medals, Teams participating as well as the Entries by gender. It contains their names, countries represented, discipline, gender of competitors, name of the coaches, medals(gold, silver, bronze).

Let's dive into the project and get started

1. Creating Blob storage account

Once you are on your Azure Portal, you can search for a Storage account on the search bar. Go to the storage account page and create a new one .



Give a name to your resource group to access all the resources deployed in this project via resource group. Give a name to your storage account and select a region (select the region that is closest to your location)

Project details

Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.

Subscription *	<input type="text" value="Azure subscription 1"/>
Resource group *	<input type="text" value="(New) olympic-rg"/>

[Create new](#)

Instance details

Storage account name ⓘ *	<input type="text" value="tokyoolympicdatasumer"/>
Region ⓘ *	<input type="text" value="(US) East US"/>

[Deploy to an edge zone](#)

You will also get to choose options about performance and redundancy. I went for Locally-redundant storage because that is the cheapest one and the data is not of much importance to me .

Basics

Advanced

Networking

Data protection

Encryption

Tags

Review

Resource group *

(New) olympic-rg

Create new

Locally-redundant storage (LRS):

Lowest-cost option with basic protection against server rack and drive failures. Recommended for non-critical scenarios.

Geo-redundant storage (GRS):

Intermediate option with failover capabilities in a secondary region. Recommended for backup scenarios.

Zone-redundant storage (ZRS):

Intermediate option with protection against datacenter-level failures. Recommended for high availability scenarios.

Geo-zone-redundant storage (GZRS):

Optimal data protection solution that includes the offerings of both GRS and ZRS. Recommended for critical data scenarios.

Geo-redundant storage (GRS)

☒

 Make read access to data available in the event of regional unavailability.

Instance details

Storage account name ⓘ *

Region ⓘ *

Performance ⓘ *

Redundancy ⓘ *

Under the Advanced option -> Hierarchical Namespace select the “Enable hierarchical namespace checkbox”. This will allow you to store your files in hierarchical format and not flat files. This hierarchical format will make storing and querying for files easier.

Hierarchical Namespace

Hierarchical namespace, complemented by Data Lake Storage Gen2 endpoint, enables file and directory semantics, accelerates big data analytics workloads, and enables access control lists (ACLs) [Learn more](#)

Enable hierarchical namespace

☒

Here you can see that you have created your storage Blob

tokyoolympicdatasumer

Storage account

Overview

Activity log

Tags

Diagnose and solve problems

Access Control (IAM)

Data migration

Events

Storage browser

Data storage

Containers

File shares

Queues

Tables

Security + networking

Networking

Access keys

Shared access signature

Upload

Open in Explorer

Delete

Move

Refresh

Open in mobile

CLI / PS

Feedback

Essentials

Resource group (move) : olympic-rg

Location : eastus

Primary/Secondary Location : Primary: East US, Secondary: West US

Subscription (move) : Azure subscription 1

Subscription ID : 7bbeb435-df60-453c-bfcf-851eaf0ad0e3

Disk state : Primary: Available, Secondary: Available

Performance : Standard

Replication : Read-access geo-redundant storage (RA-GRS)

Account kind : StorageV2 (general purpose v2)

Provisioning state : Succeeded

Created : 2/20/2024, 1:07:11 AM

Tags (edit) : Add tags

Properties

Monitoring

Capabilities (5)

Recommendations (0)

Tutorials

Tools + SDKs

Data Lake Storage

Hierarchical namespace : Enabled

Default access tier : Hot

Blob anonymous access : Enabled

Blob soft delete : Enabled (7 days)

Container soft delete : Enabled (7 days)

Versioning : Disabled

Change feed : Disabled

Security

Require secure transfer for REST API operations : Enabled

Storage account key access : Enabled

Minimum TLS version : Version 1.2

Infrastructure encryption : Disabled

Networking

Allow access from : All networks

Then create a container by selecting the container option on the left side pane of the storage account. This container will allow you to add directories and add directories to keep the raw-data and transformed data separately.

Home > tokyoolympicdatasumer_1708412819110 | Overview > tokyoolympicdatasumer

tokyoolympicdatasumer | Containers

Storage account

Overview

Activity log

Tags

Diagnose and solve problems

Access Control (IAM)

Data migration

Events

Storage browser

Data storage

Containers

File shares

Queues

Tables

Security + networking

Networking

+ Container

Change access level

Restore containers

Refresh

Delete

Give feedback

Search containers by prefix

Show

Name

Last modified

Anonymous access level

☐

 \$logs

20/02/2024, 01:07:34

Private

New container

Name *

tokyo-olympic-data

Anonymous access level ⓘ

Private (no anonymous access)

Advanced

Click on add directories to create two directories

1. raw-data-To store the raw files
2. transformed-data-To store the data files after we perform the Pyspark transformations

⏮

⬆️ Upload

➕ Add Directory

🔄 Refresh

|

🔄 Rename

🗑️ Delete

↔️ Change tier

🔗 Acquire lease

🔗 Break lease

🗨️ Give feedback

Authentication method: Access key (Switch to Microsoft Entra user account)

Location: tokyo-olympic-data

Search blobs by prefix (case-sensitive)

🔍

	Name	Modified	Access tier	Archive status	Blob type
<input type="checkbox"/>	📁 raw-data				
<input type="checkbox"/>	📁 transformed-data				

Then let’s go on to create our data factory. You can search for Azure Data Factory in the search box and then click on “Create data factory” option to create a new data factory.

Data factories

Default Directory (tan805987@gmail.onmicrosoft.com)

+

 Create

⚙️

 Manage view

🔄

 Refresh

📄

 Export to CSV

🔗

 Open query

🏷️

 Assign tags

Filter for any field...

Subscription equals all

Type equals all

Resource group equals all

Location equals all

Add filter

Showing 0 to 0 of 0 records.

No grouping

List view

Name	Type	Subscription	Resource group	Location
<div><div>🏭</div><div>No data factories to display</div><div>Try changing or clearing your filters.</div><div>Create data factory</div><div>Learn more</div></div>				

We can keep the same resource group as the storage account as these services are deployed in the same project. Give a name to your azure data factory and select a region.

Basics

Git configuration

Networking

Advanced

Tags

Review + create

One-click to create data factory with sample pipeline and datasets. [Try it](#)

Project details

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

Subscription *

Azure subscription 1

Resource group *

olympic-rg

Create new

Instance details

Name *

sumer-olympic-data-df

Region *

East US

Version *

V2

Go-on clicking next-next in the creation dialog box and you will get an option to validate the resource and create it .

Create Data Factory

🔗 View automation template

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Basics

Subscription

Azure subscription 1

Resource group

olympic-rg

Name

sumer-olympic-data-df

Region

East US

Version

V2

Networking

Connect via

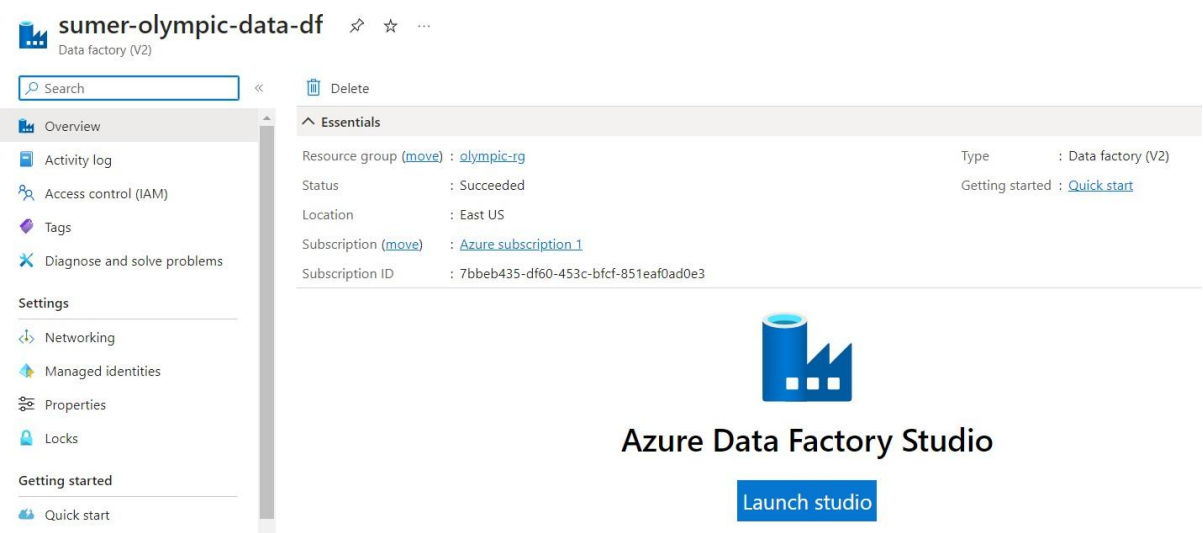
Public endpoint

Previous

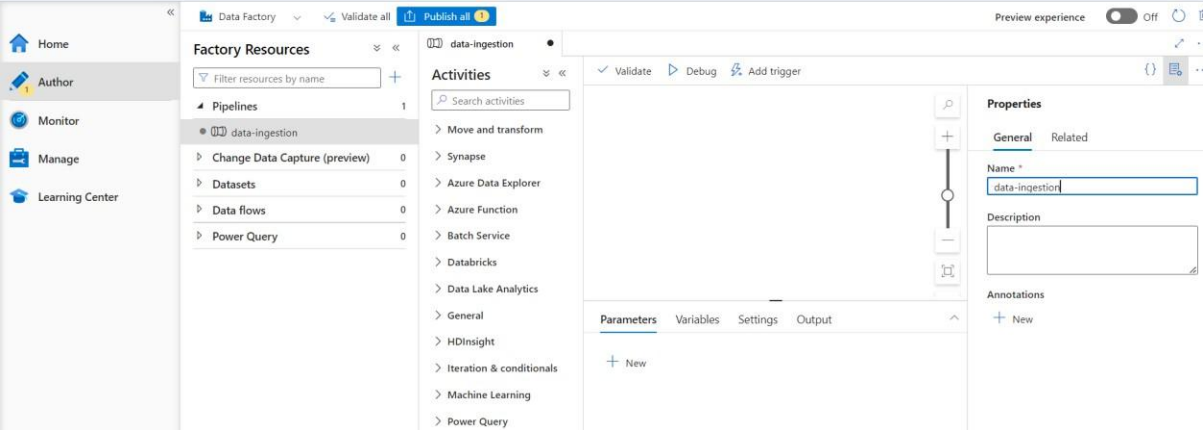
Next

Create

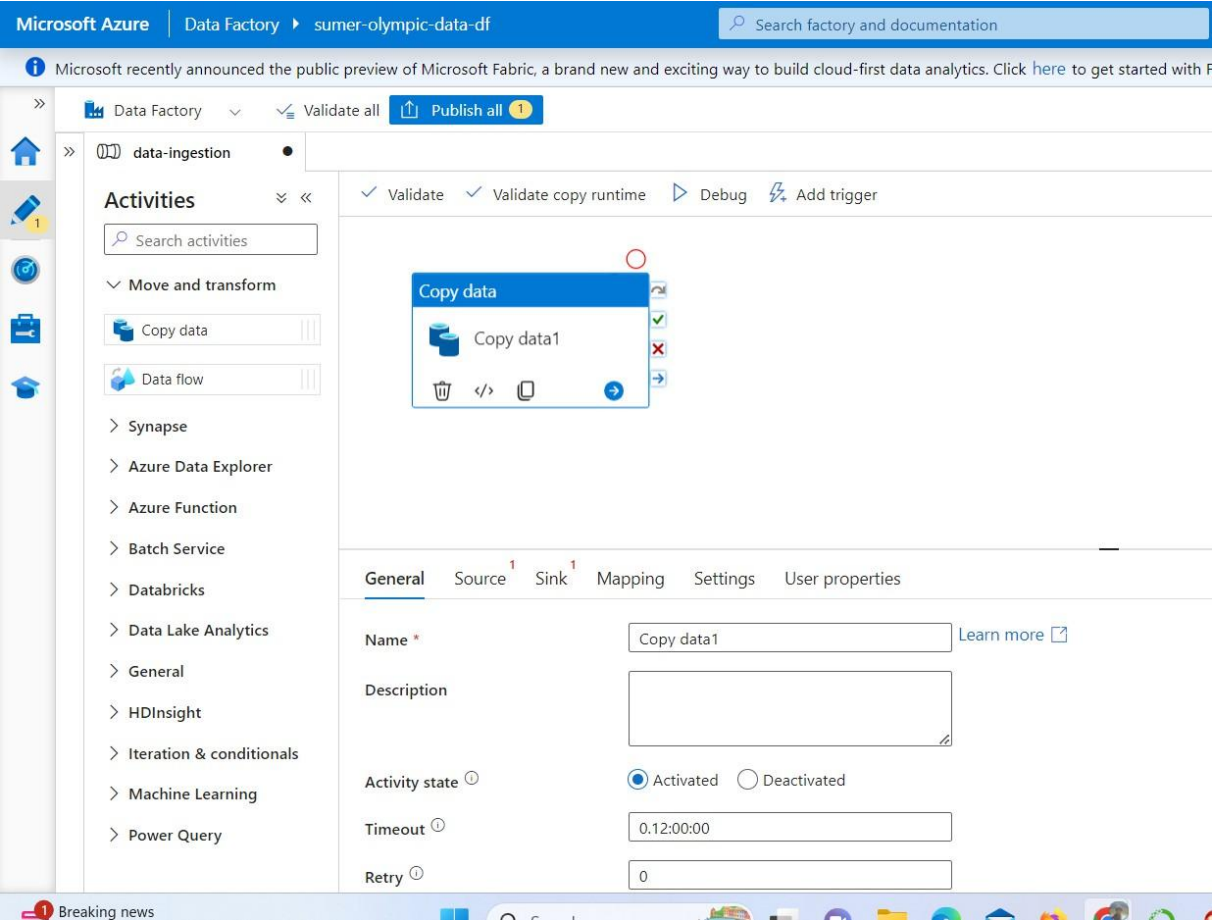
Here you can see that Azure Data Factory is Deployed .



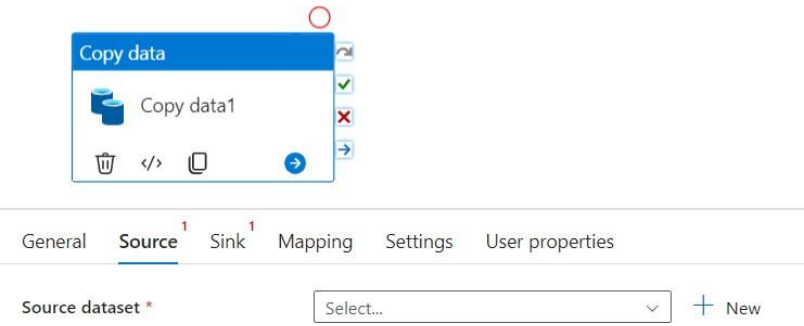
Now we have to create a pipeline to streamline our flow of data .For this create a new pipeline by selecting options Author->Pipelines->Move and transform. Give name to your data pipeline to avoid any kind of confusion later.



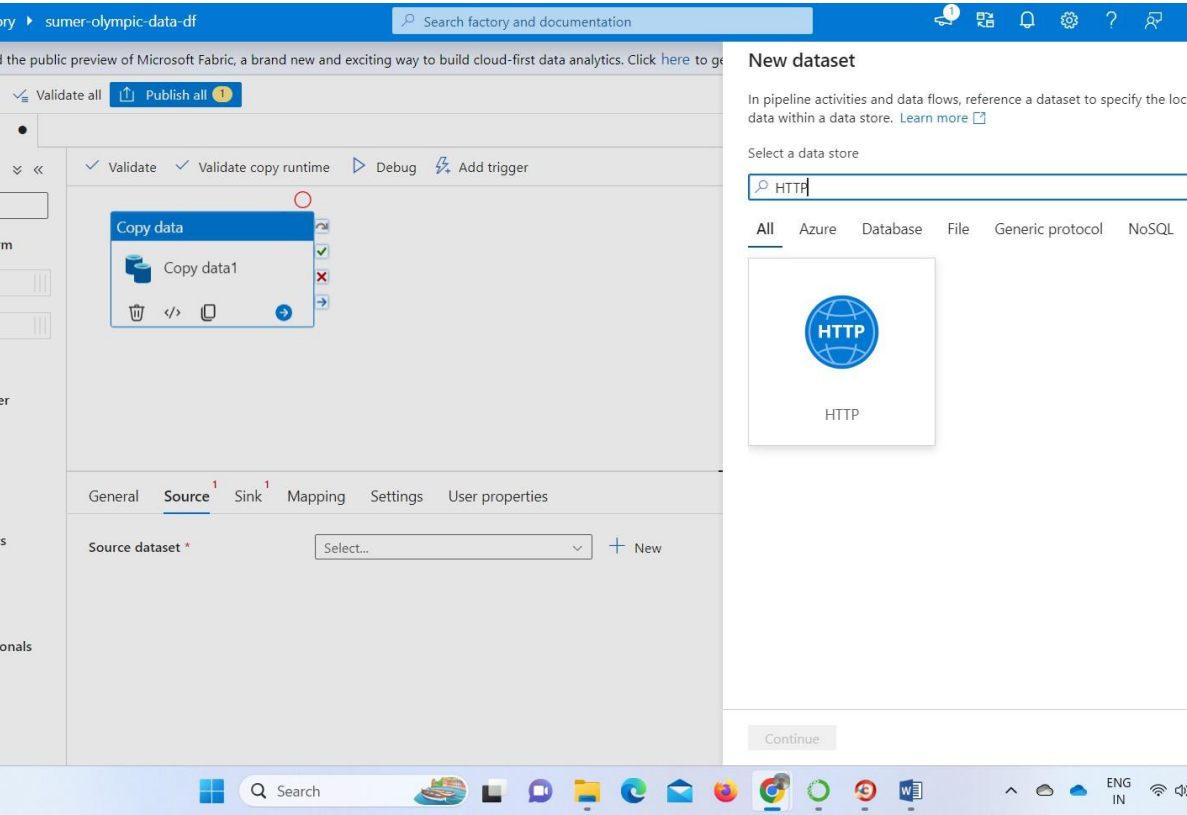
Drag the Copy data option to your window to start your data ingestion process. You can give a name to copying your data resource under the “General” tab.



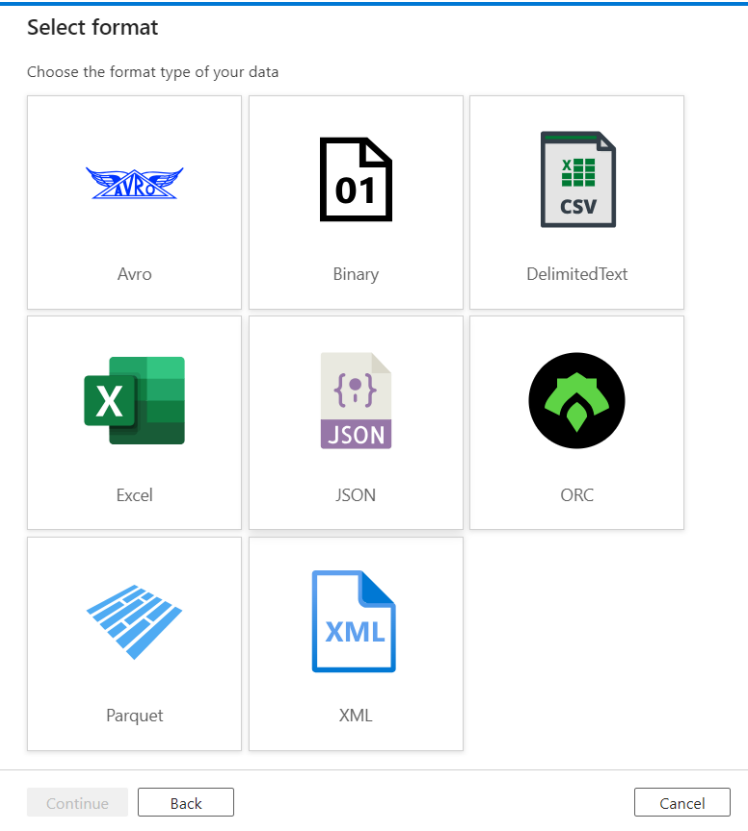
Select the Source tab and you define a location from where you have to ingest your data .Once you create a new source ,the portal will ask you to name the source and mention the path.



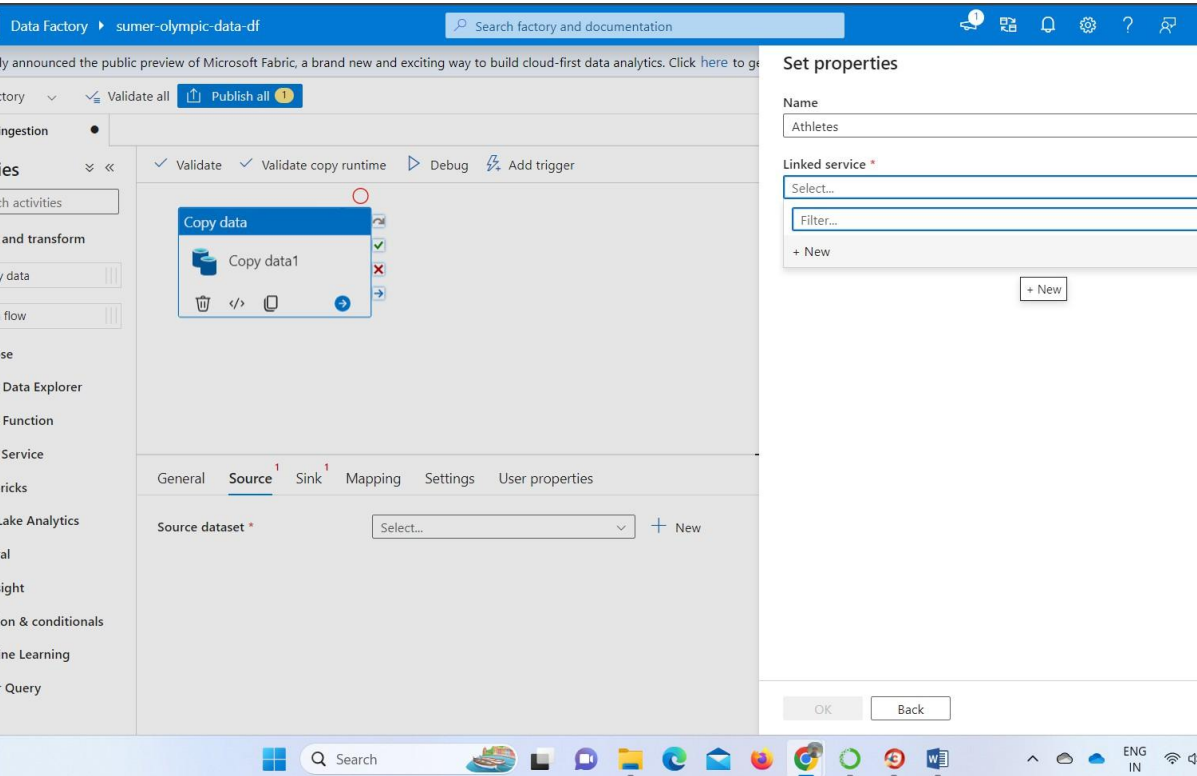
Here we are selecting the source from where we will ingest our data .I have selected through HTTP request because the data is stored on my GitHub repository. You can choose a different option based on the option available and your data location.



Select the format of the data you are ingesting .In my case it is a CSV file .



Give a name to your resource. In my case I am ingesting the athletes table from my repository so I gave the name as “Athletes”. Create a new linked service.



Then let’s give a name to our new linked service. I have named it Athletes HTTP to make things easier. I have kept the authentication as “anonymous” . It's not the best practice but you will have options to choose where you can have a secret key in your key vault through which you can access your linked service.

New linked service

HTTP

Learn more

Name *

AthletesHTTP

Description

Connect via integration runtime * ⓘ

AutoResolveIntegrationRuntime

Base URL *

https://raw.githubusercontent.com/SumerPariani/azure_cloud/main/olympic-azure-dataen

⚠ Information will be sent to the URL specified. Please ensure you trust the URL entered.

Server Certificate Validation ⓘ

Enable

Disable

Authentication type * ⓘ

Anonymous

Auth headers ⓘ

+ New

Annotations

+ New

Create

Cancel

Test connection

In this project I am ingesting the data from my GitHub repository so for this .I will go to my GitHub repository ,dataset .Select the Raw form and copy the URL of the table and paste it on the Base URL box there .

azure_cloud / olympic-azure-dataengineering-project / data / Athletes.csv

SumerPariani

0c341c4 - 7 minutes ago

History

Preview

Code

Blame

11086 lines (11086 loc) - 409 KB

Code 55% faster with GitHub Copilot

Raw

Q Search this file

1	Name	NOC	Discipline
2	AALERUD Katrine	Norway	Cycling Road
3	ABAD Nestor	Spain	Artistic Gymnastics
4	ABAGNALE Giovanni	Italy	Rowing
5	ABALDE Alberto	Spain	Basketball
6	ABALDE Tamara	Spain	Basketball
7	ABALO Luc	France	Handball
8	ABAROA Cesar	Chile	Rowing
9	ABASS Abobakr	Sudan	Swimming
10	ABBASALI Hamideh	Islamic Republic of Iran	Karate
11	ABBASOV Islam	Azerbaijan	Wrestling
12	ABBINGH Lois	Netherlands	Handball
13	ABBOT Emily	Australia	Rhythmic Gymnastics
14	ABBOTT Monica	United States of America	Baseball/Softball
15	ABDALLA Abubaker Haydar	Qatar	Athletics
16	ABDALLA Maryam	Egypt	Artistic Swimming

This is how your data looks in Raw form .

raw.githubusercontent.com/SumerPariani/azure_cloud/main/olympic-azure-dataengineering-project/data/Athletes.csv

Name,NOC,Discipline

AALERUD Katrine,Norway,Cycling Road

ABAD Nestor,Spain,Artistic Gymnastics

ABAGNALE Giovanni,Italy,Rowing

ABALDE Alberto,Spain,Basketball

ABALDE Tamara,Spain,Basketball

ABALO Luc,France,Handball

ABAROA Cesar,Chile,Rowing

ABASS Abobakr,Sudan,Swimming

ABBASALI Hamideh,Islamic Republic of Iran,Karate

ABBASOV Islam,Azerbaijan,Wrestling

ABBINGH Lois,Netherlands,Handball

ABBOT Emily,Australia,Rhythmic Gymnastics

ABBOTT Monica,United States of America,Baseball/Softball

ABDALLA Abubaker Haydar,Qatar,Athletics

ABDALLA Maryam,Egypt,Artistic Swimming

ABDALLAH Shahd,Egypt,Artistic Swimming

ABDALRASOOL Mohamed,Sudan,Judo

ABDEL LATIF Radwa,Egypt,Shooting

ABDEL RAZEK Samy,Egypt,Shooting

ABDELAZIZ Abdalla,Egypt,Karate

ABDELAZIZ Farah,Egypt,Table Tennis

ABDELAZIZ Feryal,Egypt,Karate

ABDELMAWGOUD Mohamed,Egypt,Judo

ABDELMOTTALEB Diaaeldin Kamal Gouda,Egypt,Wrestling

ABDELRAHMAN Ihab,Egypt,Athletics

ABDELSALAM Mohamed,Egypt,Football

ABDELSALAM Nour,Egypt,Taekwondo

ABDELWAHED Ahmed,Italy,Athletics

ABDI Bashir,Belgium,Athletics

ABDIRAHMAN Abdi,United States of America,Athletics

ABDUL HADI Farah Ann,Malaysia,Artistic Gymnastics

ABDUL RAHMAN Kiria Tikanah,Singapore,Fencing

ABDUL RAZZAQ Fathimath Nabaaha,Maldives,Badminton

ABDULHAMID Saud,Saudi Arabia,Football

ABDULJABBAR Ammar Riad,Germany,Boxing

ABDULLAEV Gulomjon,Uzbekistan,Wrestling

ABDULLAEV Muminjon,Uzbekistan,Wrestling

ABDULLAH Rahmat Erwin,Indonesia,Weightlifting

ABDULLIN Ilfat,Kazakhstan,Archery

ABDULREDHA Mohamed,Bahrain,Handball

ABDURAIMOV Elnur,Uzbekistan,Boxing

let’s select the option to keep the first row as header and import schema as none .By clicking Okay we have successfully created the source of the data resource to be ingested

Set properties

Name

Athletes

Linked service *

AthletesHTTP

Relative URL

First row as header

Import schema

From connection/store

From sample file

None

> Advanced

OK

Back

Cancel

You can preview the data by going on the source tab and clicking the “preview data” option

GeneralSourceSink¹MappingSettingsUser properties

Source dataset *

Athletes

Open

New

Preview data

Learn more

Request method * ⓘ

GET

Additional headers ⓘ

Request body ⓘ

Here I have previewed the data

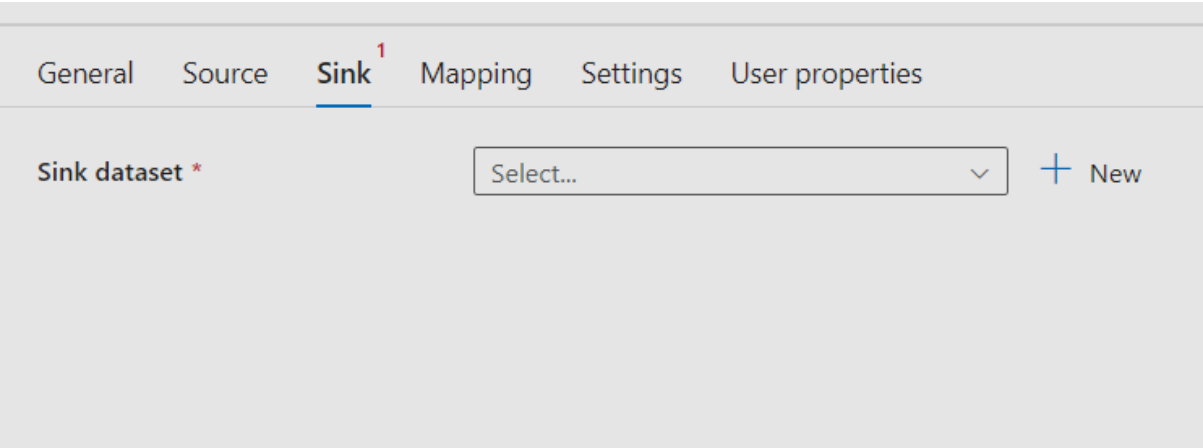
Preview data

Linked service: AthletesHTTP

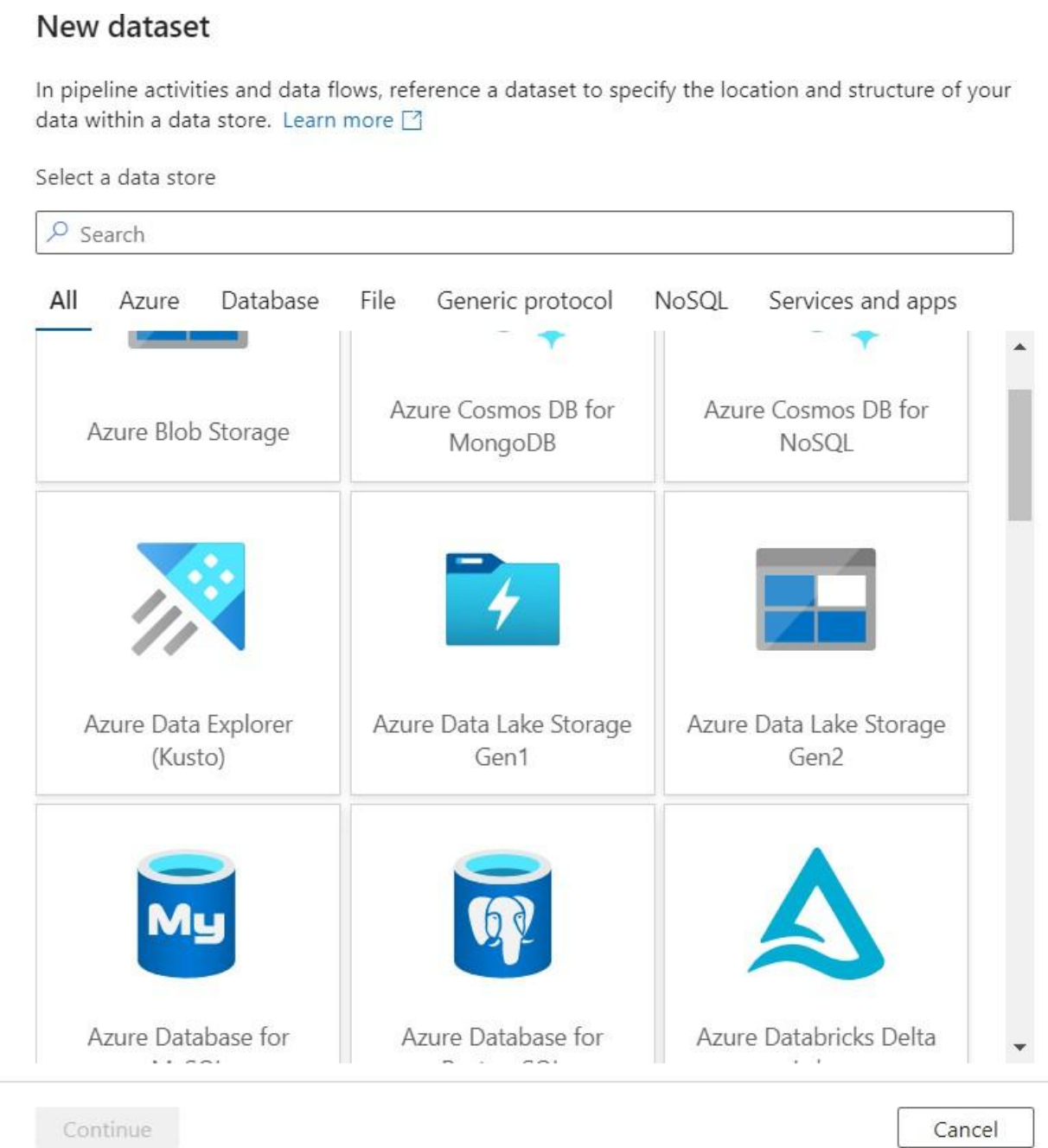
Object:

	Name	NOC	Discipline
1	AALERUD Katrine	Norway	Cycling Road
2	ABAD Nestor	Spain	Artistic Gymnastics
3	ABAGNALE Giovanni	Italy	Rowing
4	ABALDE Alberto	Spain	Basketball
5	ABALDE Tamara	Spain	Basketball
6	ABALO Luc	France	Handball
7	ABAROA Cesar	Chile	Rowing
8	ABASS Abobakr	Sudan	Swimming
9	ABBASALI Hamideh	Islamic Republic of Iran	Karate
10	ABBASOV Islam	Azerbaijan	Wrestling

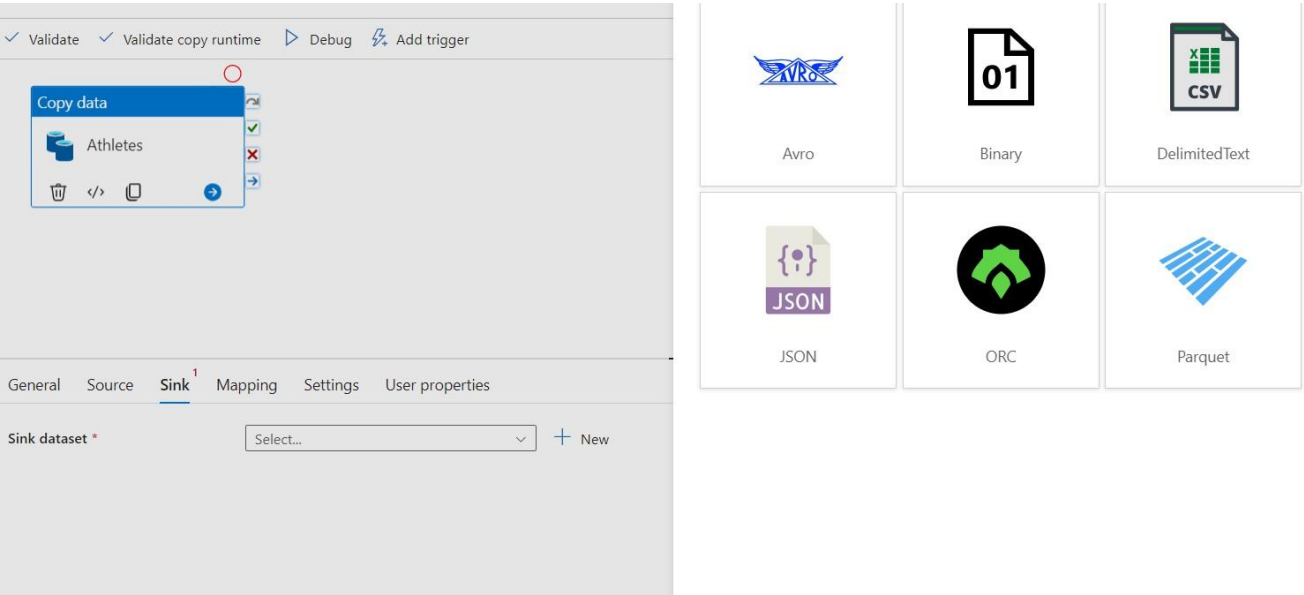
Now the next step is to create Sink for the data .Let’s create a sink for the source data by selecting the source tab and clicking the create new option.



Let’s select the Sink location for the data source we have created .I have selected Data Lake Gen2 (ADLS).You can select any other option your prefer.



Now we have to select the format in which we have to keep our data in our ADLS .I have selected the CSV format .



Set the properties of your sink service ,Give name and create a new linked service

Set properties

Name

DelimitedText1

Linked service *

Select...

Filter...

+ New

+ New

I have given the name AzureDataLakeStorage1 and then selected your subscription and the storage account (tokyoolympicdatasumer in my case).

New linked service

Azure Data Lake Storage Gen2

Learn more

Name *

AzureDataLakeStorage1

Description

Connect via integration runtime * ⓘ

AutoResolveIntegrationRuntime

Authentication type

Account key

Account selection method ⓘ

From Azure subscription

Enter manually

Azure subscription ⓘ

Azure subscription 1 (7bbeb435-df60-453c-bfcf-851eaf0ad0e3)

Storage account name *

tokyoolympicdatasumer

Test connection ⓘ

To linked service

To file path

Annotations

Create

Cancel

Test connection

select the file path by browsing the “raw-data” directory we have created in our storage Blob->container->directory.

Set properties

Name

DelimitedText1

Linked service *

AzureDataLakeStorage1

File path

File system / Directory / File name

First row as header

Filename doesn't support wildcard in dataset

Import schema

From connection/store

From sample file

None

Advanced

Root folder > tokyo-olympic-data

- raw-data
- transformed-data

I have not imported the schema and then clicked okay.

File path

tokyo-olympic-data

/

raw-data

/

athletes.csv

First row as header☒

Import schema

☐ From connection/store

☐ From sample file

☒ None

> Advanced

And ring ding ding ding we have successfully ingested our athletes data.

✓ Validate

✓ Validate copy runtime

▶ Debug

⚡ Add trigger

Copy data

Athletes

General

Source

Sink

Mapping

Settings

...

Sink dataset *

DelimitedText1

Open

New

Learn more

Copy behavior

Select...

Max concurrent connections

Properties

General

Related

Name *

data-ingestion

Description

Annotations

New

✓

data-ingestion

Your pipeline has been validated.

No errors were found.

Validate the change you have made and then if the changes are validated click on publish option to save the changes

✓ Validate

▶ Debug

⚡ Add trigger

Copy data

Athletes

Now we can see that we have successfully ingested(copied) our data to ADLS(raw-storage).

Parameters




Variables

Settings


Output

Pipeline run ID:

d9b1aa41-20ea-45b0-865f-4b2638b9c7d3




Pipeline status


 Succeeded

[View debug run consumption](#)


All status






[Monitor in Azure Metrics](#)



[Export to CSV](#)



Showing 1 - 1 of 1 items

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime
Athletes	   Succeeded	Copy data	2/20/2024, 2:14:14 AM	15s	AutoResolveIntegrati

you can see the athletes CSV file by going to your raw-data directory in Azure BLOB

tokyo-olympic-data

Container

Search

«

Upload

Add Directory

Refresh

Rename

Delete

Change tier

Acqui

Overview

Diagnose and solve problems

Access Control (IAM)

Settings

Shared access tokens

Manage ACL

Access policy

Properties

Metadata

Authentication method:

Access key (Switch to Microsoft Entra user account)

Location:

tokyo-olympic-data / raw-data

Search blobs by prefix (case-sensitive)

Name	Modified	Access tier
<div><div></div><div></div></div> [...]		
<div><div></div><div></div></div> athletes.csv	20/02/2024, 02:14:27	Hot (Inferred)

let’s repeat the same process for all the five data files we have

Athletes	✔ Succeeded	Copy data	2/20/2024, 2:26:29 AM	15s	AutoResolveIntegration	467101f0-b95a-4205-9d9
Coaches	✔ Succeeded	Copy data	2/20/2024, 2:26:29 AM	15s	AutoResolveIntegration	fb56e2d2-3827-457f-830:

Once we are done with this we can see that we have ingested your data and created our ingestion pipeline .

Validate

Debug

Add trigger

Copy data

Athletes

Copy data

Coaches

Copy data

EntriesGender

Copy data

Medals

Copy data

Teams

Parameters

Variables

Settings

Output

Showing 1 - 5 of 5 items

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	User properties	Activity run ID
Teams	✔ Succeeded	Copy data	2/20/2024, 2:40:34 AM	14s	AutoResolveIntegration		ba6d5f61-4430-4c30-8
Medals	✔ Succeeded	Copy data	2/20/2024, 2:40:17 AM	14s	AutoResolveIntegration		9c31c43e-8f61-49b0-9
EntriesGender	✔ Succeeded	Copy data	2/20/2024, 2:39:57 AM	13s	AutoResolveIntegration		6deb88fa-9a61-48bd-5
Coaches	✔ Succeeded	Copy data	2/20/2024, 2:39:40 AM	15s	AutoResolveIntegration		48a8f3a0-c62f-4c45-bc
Athletes	✔ Succeeded	Copy data	2/20/2024, 2:39:26 AM	14s	AutoResolveIntegration		2f2f0001-6f24-4b6b-9c

All the five CSV files are present in our raw-data storage container .

Upload

Add Directory

Refresh

Rename

Delete

Change tier

Acquire lease

Break lease

Give feedback

Authentication method:

Access key (Switch to Microsoft Entra user account)

Location:

tokyo-olympic-data / raw-data

Search blobs by prefix (case-sensitive)

Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
<div><div></div><div></div></div> athletes.csv	20/02/2024, 02:39:38	Hot (Inferred)		Block blob	408.67 KiB	Available ***
<div><div></div><div></div></div> coaches.csv	20/02/2024, 02:39:54	Hot (Inferred)		Block blob	16.49 KiB	Available ***
<div><div></div><div></div></div> entriessgender.csv	20/02/2024, 02:40:07	Hot (Inferred)		Block blob	1.1 KiB	Available ***
<div><div></div><div></div></div> medals.csv	20/02/2024, 02:40:30	Hot (Inferred)		Block blob	2.35 KiB	Available ***
<div><div></div><div></div></div> teams.csv	20/02/2024, 02:40:46	Hot (Inferred)		Block blob	34.44 KiB	Available ***

Now let us set up the Azure Databricks workspace to perform the necessary transformation in our data using Pyspark and then store the data in your transformed folder in a storage blob.create your Azure Databricks workspace by selecting the same resource group and region, give a name and click create .

Create an Azure Databricks workspace

Basics

Networking

Encryption

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 *

Azure subscription 1

Resource group *

olympic-rg

Create new

Instance Details

Workspace name *

olympic-databricks

Region *

East US

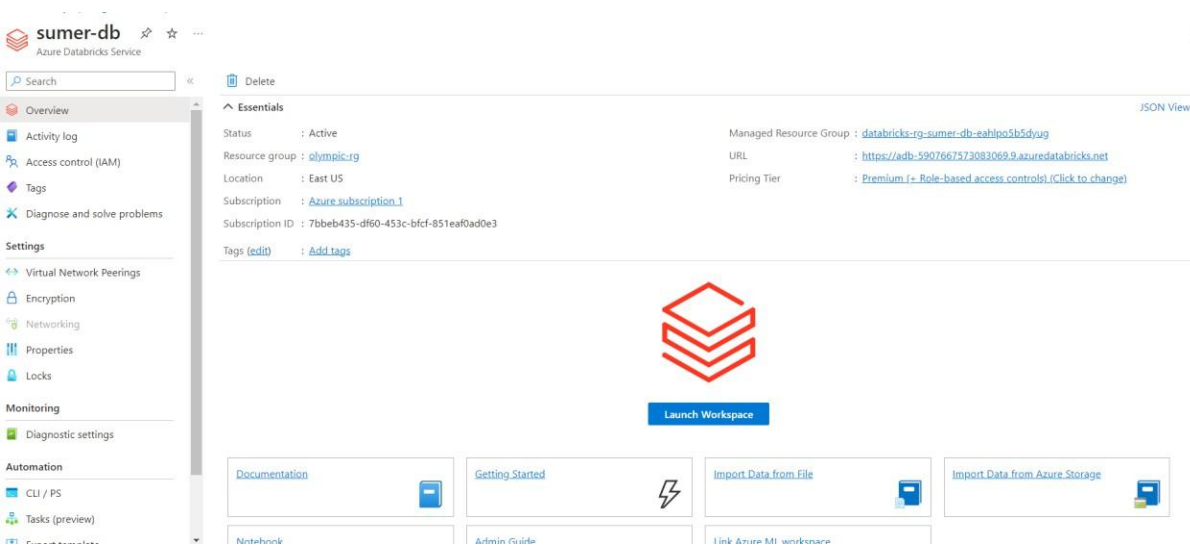
Pricing Tier *

Trial (Premium - 14-Days Free DBUs)

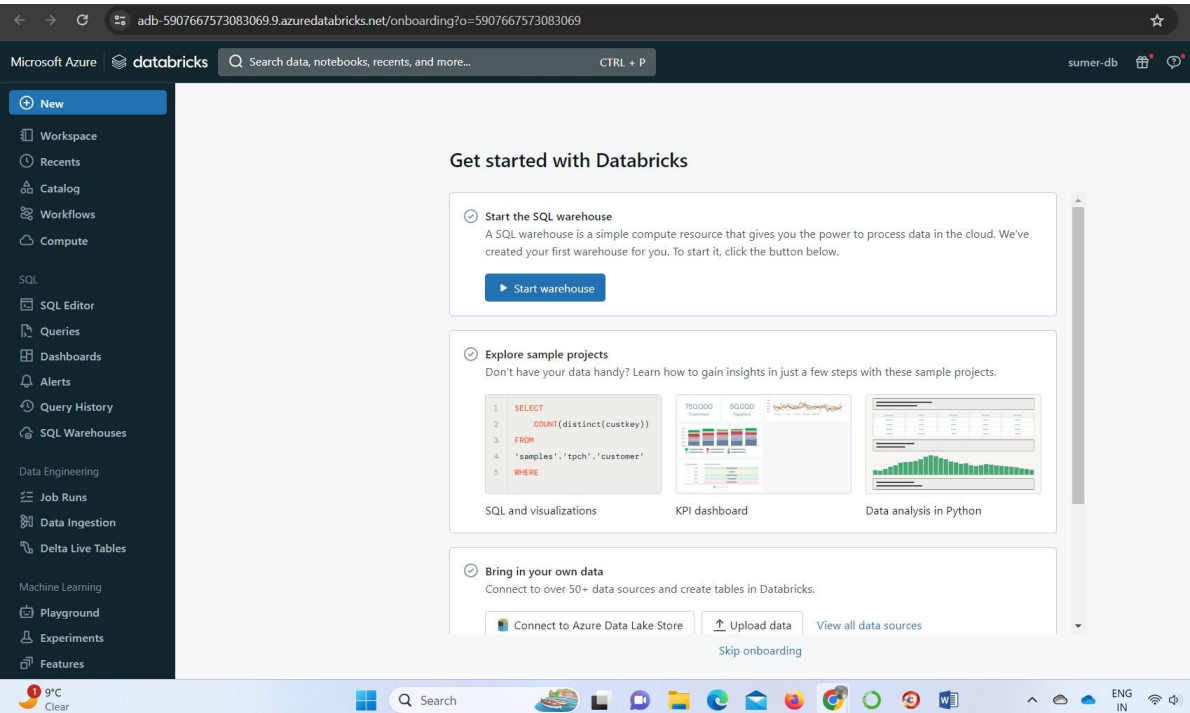
Managed Resource Group name

Enter name for managed resource group


Launch your workspace





Select the Compute option to create your work cluster



Once you are in your workspace (Sumer’s Cluster),select the machine you want to work with by selecting the policy, number of nodes, node type, runtime version and click create compute .

Sumer's Cluster 

Policy 


Unrestricted 


Multi node


Single node


Performance


Type to search...


Databricks runtime version 

Runtime: 12.2 LTS (Scala 2.12, Spark 3.3.2) 

Use Photon Acceleration 


Node type 


Standard_DS3_v2 14 GB Memory, 4 Cores 



Terminate after

120

minutes of inactivity 

Tags 

Add tags

Key

Value

Add

> Automatically added tags

▶ Advanced options

Create compute

Cancel

we can see that we have successfully created our cluster now on which we can perform our transformation in Pyspark

Compute

All-purpose computeJob computeSQL warehousesVector SearchPreviewPoolsPolicies ⓘ

Q Filter compute you have access to

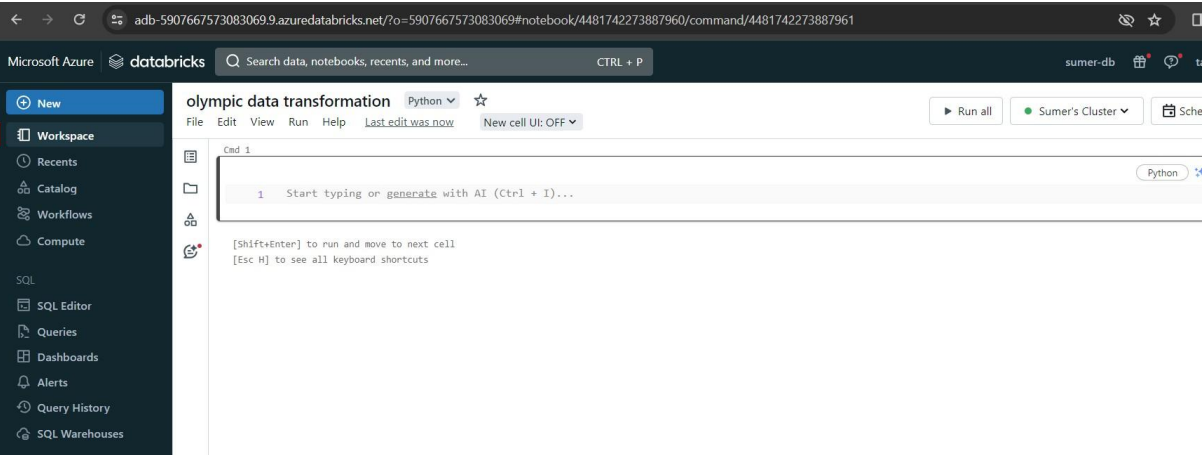
Created by ▾

☐ Only pinned

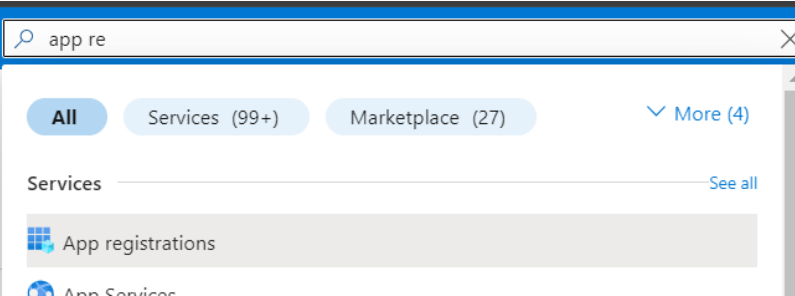
Create

State ↕	Name	Policy	Runtime	Active mem...	Active cores	Active DBU ...	Source
	Sumer's Cluster	-	12.2	14 GB	4 cores	0.75	UI

Let’s give a name to our data transformation code and start our data transformation process. Before we write our data transformation code we need to give permission to data brick to access our data.



For that we need to go to the App registration service by type App registration in search box



Click on new registration
Register your application by selecting the supported account types.

Register an application

* Name

The user-facing display name for this application (this can be changed later).

app01

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory (Default Directory only - Single tenant)

☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)

☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

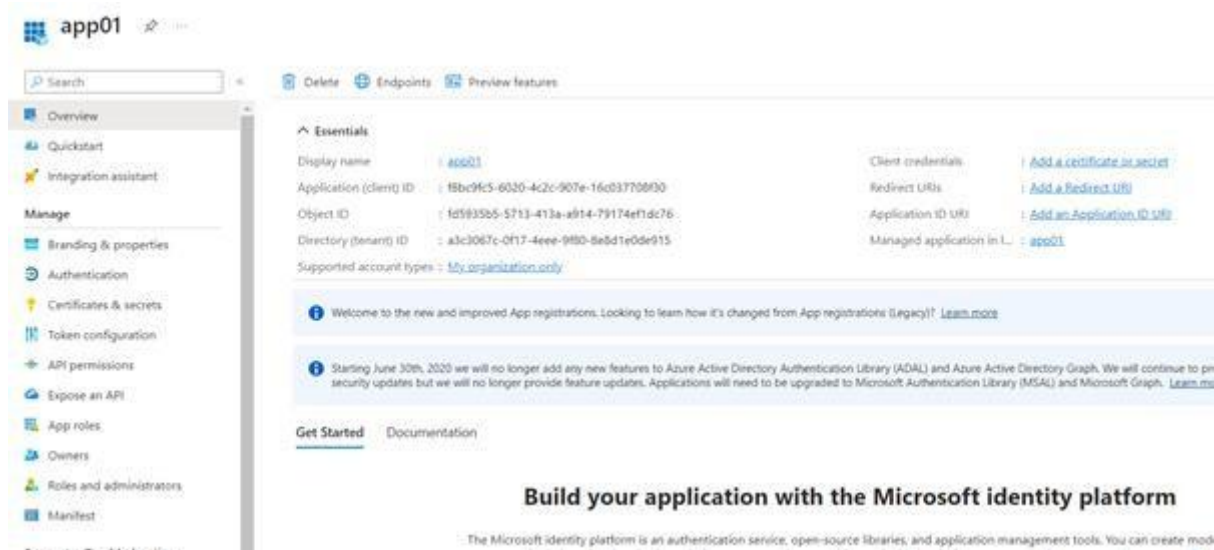
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

By proceeding, you agree to the Microsoft Platform Policies

Register

Once the app is created you will have your
Application (client ID)
Directory(Tenant ID)

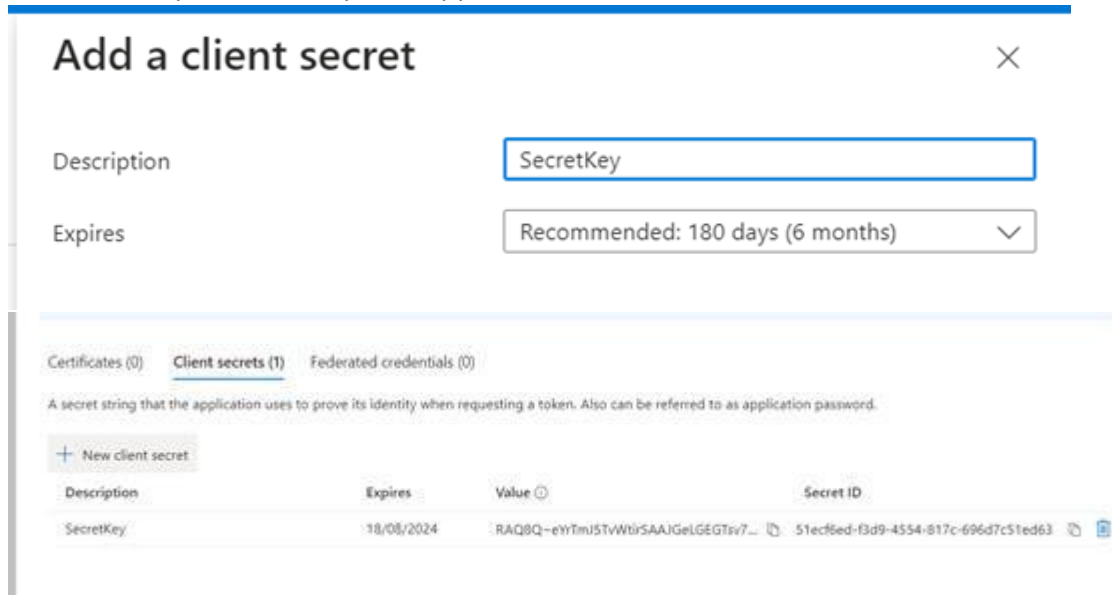
Copy paste these ID’s on a notepad



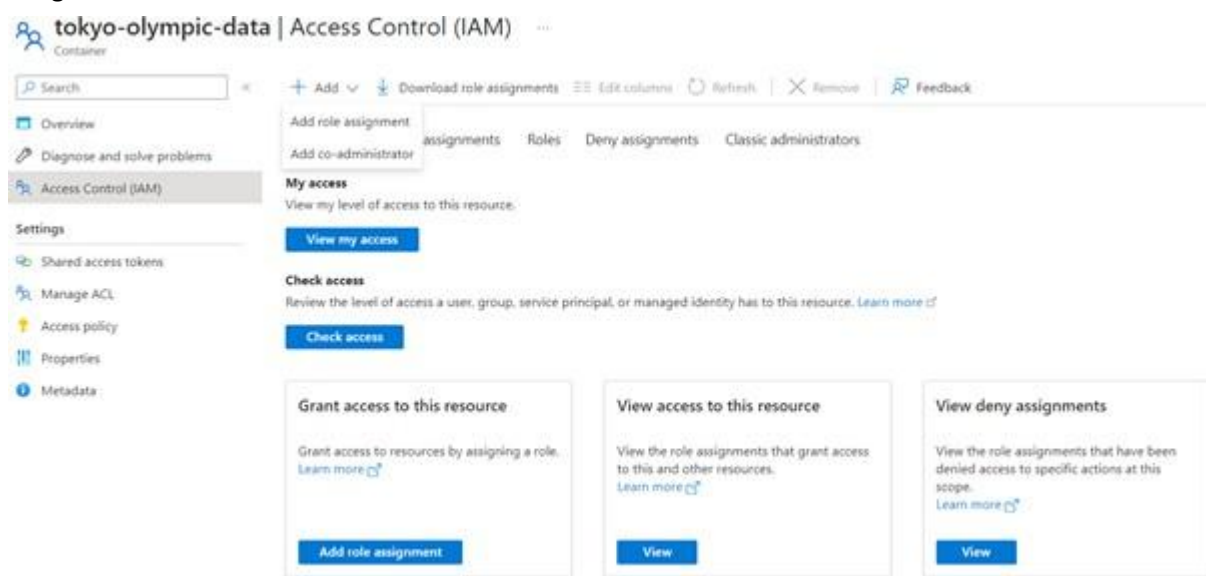
click Certificate and secrets option on the left side pane and create a new secret key



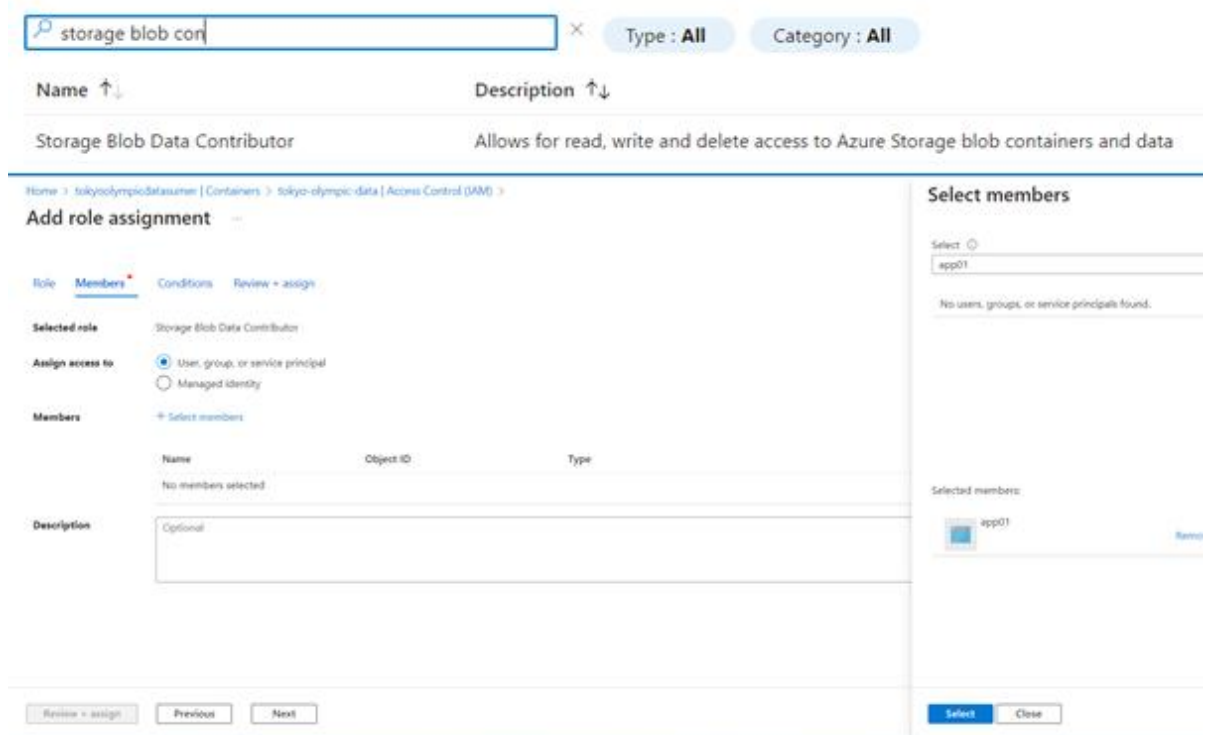
Give name to your secret key and copy the value



Now we have to give permission to our to be able to access the data kept in our raw storage .For that go to your storage blob->containers->Access control->Add->Add role assignment



Select the role of Storage Blob data contributor and select the members as your app .



App registrations

[+ New registration](#) [Endpoints](#) [Troubleshooting](#) [Refresh](#) [Download](#) [Preview features](#) | [Got feedback?](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

All applications

Owned applications

Deleted applications

Applications from personal account

Start typing a display name or application (client) ID to filter these r...

Add filters

You have successfully given the permission to your app to access the storage account

Add role assignment

Role

Members

Conditions

Review + assign

Role

Storage Blob Data Contributor

Scope

/subscriptions/7bbeb435-df60-453c-bfcf-851ea0ad0e3/resourceGroups/olympic-rg/providers/Microsoft.Storage/storageAccounts/tokyoolympicdatasumer/blobServices/default/containers/tokyo-olympic-data

Members

Name	Object ID	Type
app01	208119de-c6db-49a2-92e9-aa604793feb9	App

Description

No description

Condition

None

Review + assign

Previous

Next

Once we have successfully given the permission to our app we now have to connect with our app using the databrick workspace and mount our data on data bricks .

Below is the code to connect with your app and mount the data on databricks .

```
configs = {"fs.azure.account.auth.type": "OAuth",
```

```
"fs.azure.account.oauth.provider.type": "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
```

```
"fs.azure.account.oauth2.client.id": "f96d1bdb-fb31-4af5-96bf-2c7c68afa0d1",
```

```
"fs.azure.account.oauth2.client.secret": 'nh08Q~2_QV5P1lcuI8OoNKshcPpJAIaMhGsFxccF',
```

```
"fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/a3c3067c-0f17-4eee-9f80-8e8d1e0de915/oauth2/token"} 
```

```
dbutils.fs.mount(
```

```
source = "abfss://tokyo-olympic-data@tokyoolympicdatasumer.dfs.core.windows.net", # contrainer@storageacc
```

```
mount_point = "/mnt/tokyoolymicsumer",
```

```
extra_configs = configs)
```

For the data transformation you need to start your spark session and explore your dataset for any anomalies .

I have inspected the dataset and changed the data types of a few columns by exploring their schema’s. You can also set the first row as header here and deal with duplicates,missing values ,data type transformation ,new column creation and the transformation you need in your dataset here .

Link to Transformation notebook-[Transformation notebook](#)

Now we have to load our transformed dataset to the transformed-data container in our storage block .Below is the code for that .

```
athletes.repartition(1).write.mode("overwrite").option("header",'true').csv("/mnt/tokyoolymicsumer/transformed-data/athletes")
coaches.repartition(1).write.mode("overwrite").option("header","true").csv("/mnt/tokyoolymicsumer/transformed-data/coaches")
```

```
entriesgender.repartition(1).write.mode("overwrite").option("header","true").csv("/mnt/tokyoolymicsumer/transformed-data/entriesgender")
```

```
medals.repartition(1).write.mode("overwrite").option("header","true").csv("/mnt/tokyoolymicsumer/transformed-data/medals")
```

```
teams.repartition(1).write.mode("overwrite").option("header","true").csv("/mnt/tokyoolymicsumer/transformed-data/teams")
```

Home > tokyoolympicdatasumer > Containers >

tokyo-olympic-data

Container

Search

Upload

+ Add Directory

Refresh

Rename

Delete

Change tier

Acquire lease

Break lease

Give feedback

Overview

Diagnose and solve problems

Access Control (IAM)

Settings

Shared access tokens

Manage ACL

Access policy

Properties

Metadata

Authentication method: Access key (Switch to Microsoft Entra user account)

Location: tokyo-olympic-data / transformed-data / athletes

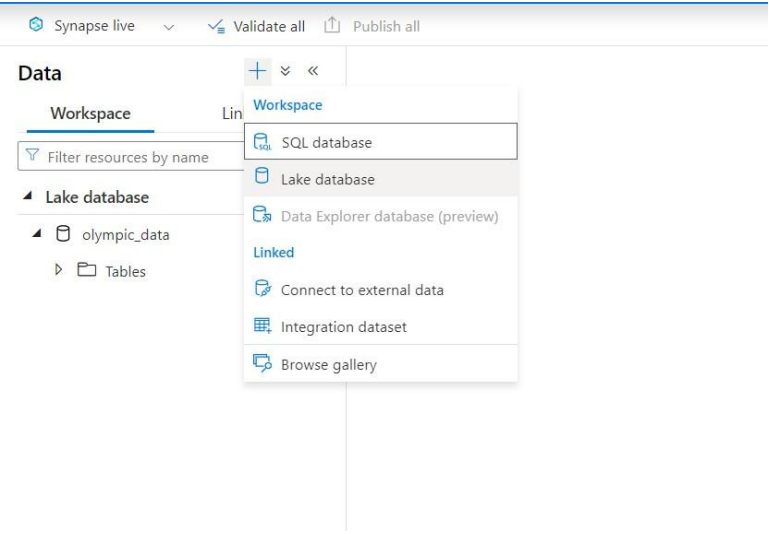
Search blobs by prefix (case-sensitive)

Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
<div><div></div><div></div></div> [-]						...
<div><div></div><div>_committed_3216881274779219328</div></div>	20/02/2024, 04:18:09	Hot (Inferred)		Block blob	112 B	Available
<div><div></div><div>_started_3216881274779219328</div></div>	20/02/2024, 04:18:09	Hot (Inferred)		Block blob	0 B	Available
<div><div></div><div>_SUCCESS</div></div>	20/02/2024, 04:18:09	Hot (Inferred)		Block blob	0 B	Available
<div><div></div><div>part-0000-tid-3216881274779219328-3fd112ab-4...</div></div>	20/02/2024, 04:18:09	Hot (Inferred)		Block blob	397.9 KiB	Available

Now that we have our transformed data in the storage blob we need to create an environment to query this data using Azure Synapse analytics .For this follow the same process of searching for Azure Synapse analytics in the search box and open the workspace .

To ingest the data we need to click on the Add option under the Data pane and select the option of lake data .



Give a name to our new Database, select the linked services and browse to the transformed data file

Properties

General

Related (0)

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

Name *

Database1

Description

Storage settings for database

Linked service * ⓘ

olympic-data-sumer-sa-Workspac...

Input folder * ⓘ

tokyo-olympic-data/Database1

Data format *

Delimited Text

Once the database is created we need to import the table by adding table option ,give name to our table and attach the related linked services .Validate the data you are importing and publish it .

Create external table from data lake

External table details

Select the storage location where the files containing the data is staged. Currently Azure Data Lake Storage (ADLS) Gen2 and Azure Blob Storage are supported. [Learn more](#)

External table name *

athletes

Linked service * ⓘ

Select a linked service

olympic-data-sumer-sa-WorkspaceDefaultStorage(tokyoolympicdatasumer)

You will have all the data in your Azure Synapse analytics ,you can run queries on the data to get insights of the dataset .

The last and final step in the pipeline is to create visualization based on the insights which you got by connecting Azure Synapse analytics to Power BI or any other dashboarding tool.

Below is the Dashboard which I have created .

