

TOKYO OLYMPIC DATA ANALYZATION USING AZURE SERVICES

PROJECT REPORT DONE BY :-

Ms.Mansi Jadhav

Mr.Jayesh Waingankar

Mr.Yogesh Pawar

TABLE OF CONTENTS

SR.NO	CONTENTS	PAGE NO.
1	Synopsis	3 - 4
2	Description of services	5
3	Usage of services	6-7
2	Data Flow	8
3	Diagram	9
4	Project Design	10-53
5	Advantages and Conclusion	54-55
6	Bibliography	56

SYNOPSIS

❖ **Introduction :-**

We have done analysis of data set using azure services .Here we have our data source and dataset of Tokyo Olympic. We will copy the data from data source to our Azure Data Storage using Data Factory service so we will build pipeline where we will upload raw data on data storage which is Data Lake Gen2.Once we have our data available onto raw storage then we will use azure data picks and write our code and do some transformation. Main Goal is to build a data file plan and show how you can build project so it just simple pipeline and try to remove duplicates and also drop columns rename it and then upload our data back to data lake storage which is transform data so this is initial pipeline where we will load our data do transformation and then again load that data onto data lake and then we will use synapse analytics to understand our data so we can easily write SQL query on top of these data that we have transform and extract the insight from it. So we will perform each and every steps.

❖ **Services used:-**

1. Resource Group
2. Azure Data Factory
3. Azure Synapse Analyze
4. Azure Data Bricks
5. App registration

❖ **Requirements :-**

1. Laptop and stable internet connection
2. Basics of SQL query
3. Azure Services

❖ **Scope :-**

1. Data Source and Dataset: The project focuses on analyzing data related to the Tokyo Olympics. This includes data from various sources such as athletes, teams, medals, etc.
2. Azure Services Utilization: The project leverages various Azure services like Data Factory, Data Lake Storage Gen2, Databricks, Synapse Analytics, and Power BI.
3. Data Ingestion and Storage: The project involves ingesting raw data from a GitHub repository and storing it in Azure Data Lake Storage Gen2. This allows for efficient data management and retrieval.
4. Data Transformation: Azure Databricks is used for data transformation. This includes tasks like removing duplicates, dropping unnecessary columns, and renaming columns.

5. Data Analysis: Azure Synapse Analytics is employed for data analysis. This involves writing SQL queries to extract insights from the transformed data.
6. Data Visualization: Power BI is used for creating visually appealing and informative reports and dashboards based on the analyzed data.

❖ **Objective :-**

1. Data Pipeline Establishment: Create a robust data pipeline for ingesting, storing, transforming, and analyzing Tokyo Olympic data.
2. Data Quality Assurance: Ensure the quality of data through transformation steps like removing duplicates and dropping unnecessary columns.
3. Insight Extraction: Extract meaningful insights from the data through SQL queries in Azure Synapse Analytics.
4. Interactive Reporting: Generate visually engaging reports and dashboards in Power BI to facilitate easy understanding and decision-making.

❖ **Problems faced:-**

1. Access Control Configuration: Setting up proper access control in Azure Data Lake Storage Gen2 required careful attention to ensure secure data handling.
2. Data Integration Complexities: Managing different data formats and ensuring seamless integration from a variety of sources presented initial challenges.
3. Data Quality Issues: Dealing with inconsistent data quality in the raw dataset necessitated careful handling to ensure accurate insights
4. Power BI Customization: Designing a visually appealing and informative dashboard in Power BI required experimentation and learning to leverage its full potential.

DESCRIPTION OF SERVICES

1. Resource Group :

A Resource Group in Azure is a logical container that holds related Azure resources (such as virtual machines, databases, and websites) together. It helps you manage, organize, and monitor these resources as a single entity. Resource Groups allow you to apply policies, access control, and billing to multiple resources at once.

2. Azure Data Factory:

Azure Data Factory is a cloud-based data integration service that allows you to create, schedule, and manage data pipelines. It enables the extraction, transformation, and loading (ETL) of data from various sources into data stores and analytics platforms, both in the cloud and on-premises. Data Factory supports a wide range of data connectors and provides tools for data orchestration and monitoring.

3. Azure Synapse Analytics:

Azure Synapse Analytics is an analytics platform that integrates enterprise data warehousing, big data processing, and data integration. It allows you to analyze large volumes of data using both serverless and provisioned resources. It provides features for data integration, data warehousing, big data processing, and machine learning capabilities.

4. Azure Databricks:

Azure Databricks is an Apache Spark-based analytics platform optimized for Azure. It provides a collaborative environment for building, training, and deploying machine learning models and performing data analysis at scale. It includes features for data engineering, data science, and business intelligence.

5. App Registration:

App Registration in Azure Active Directory (Azure AD) is a way to define the characteristics and permissions of an application that wants to integrate with Azure AD. It involves creating an identity for the application, configuring authentication settings, and specifying which resources the application can access. This is a fundamental step in enabling secure access to various Azure services and APIs.

USAGE OF SERVICES

Azure Data Factory:

Usage: Data Factory was used for creating data pipelines to ingest, transform, and load data from different sources into Azure Data Lake Storage Gen2.

Reasoning: Data Factory is designed for building and managing data pipelines. It supports data movement and transformation activities, making it an ideal choice for handling data workflows in this project.

Azure Data Lake Storage Gen2:

Usage: Data Lake Storage Gen2 was used as the target storage location for the raw and transformed data.

Reasoning: Data Lake Storage Gen2 is optimized for big data analytics workloads and provides a hierarchical namespace, which makes it easier to organize and manage large amounts of data.

Azure Databricks:

Usage: Databricks was used for data transformation tasks, including removing duplicates, dropping unnecessary columns, and performing other data preparation steps.

Reasoning: Databricks is a powerful tool for big data processing and analytics. It provides a collaborative environment for data scientists and engineers to work on data transformation tasks at scale.

Azure Synapse Analytics:

Usage: Synapse Analytics was used for data analysis using SQL queries to extract insights from the transformed data.

Reasoning: Azure Synapse Analytics is an enterprise-grade analytics platform that integrates data warehousing, big data processing, and data integration. It's well-suited for performing complex analytical queries on large datasets.

Azure Power BI:

Usage: Power BI was used for data visualization, creating visually appealing reports and dashboards based on the analyzed data.

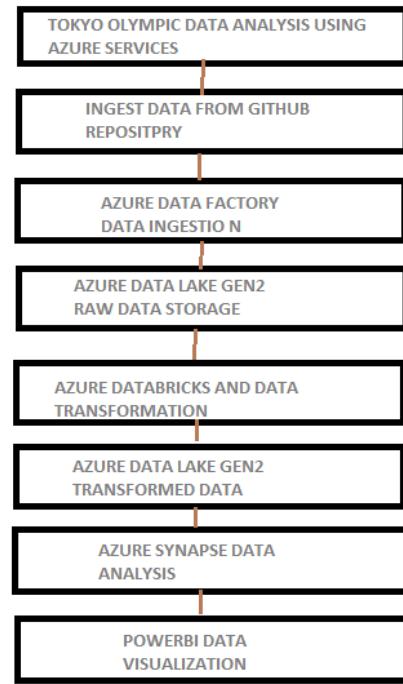
Reasoning: Power BI is a leading business analytics tool that enables users to visualize and share insights from their data. It's highly effective for creating interactive and informative reports.

App Registration (Azure Active Directory):

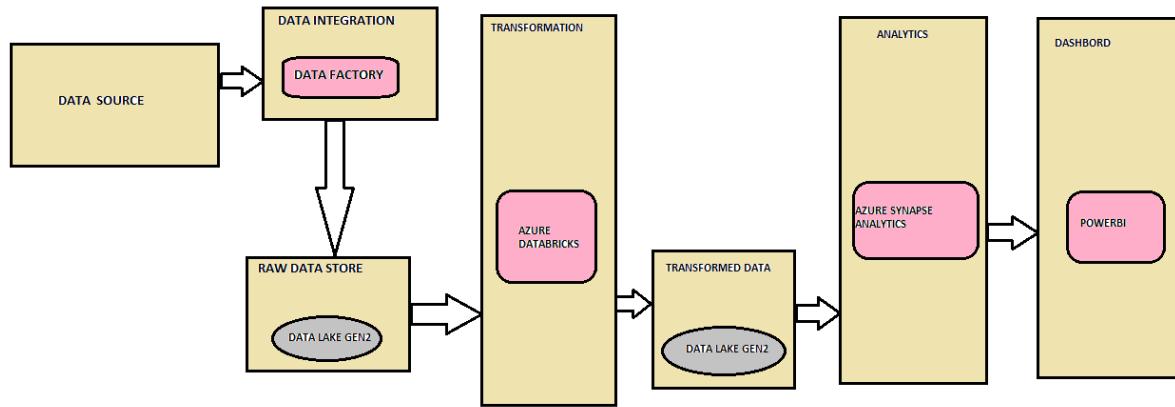
Usage: App Registration was used to define the characteristics and permissions of an application that integrates with Azure AD. It's a crucial step in enabling secure access to various Azure services and APIs.

Reasoning: App Registration ensures that only authorized applications have access to the resources in Azure AD, providing an additional layer of security for the project.

DATA FLOW

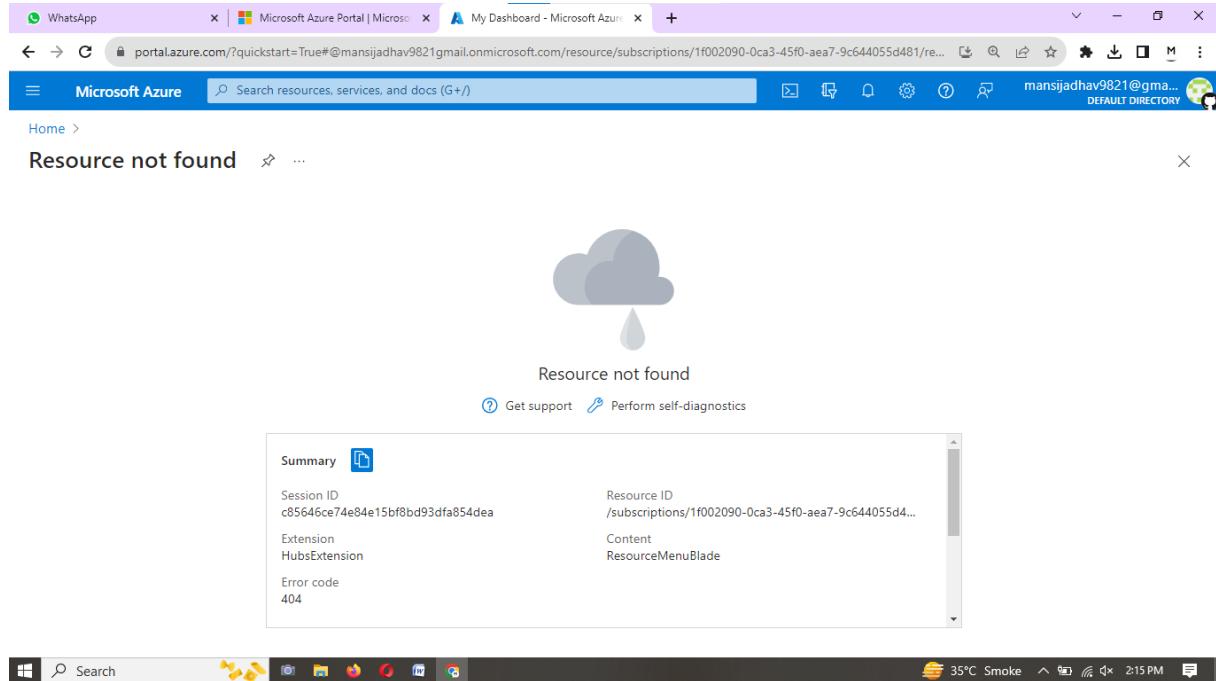


DIAGRAM

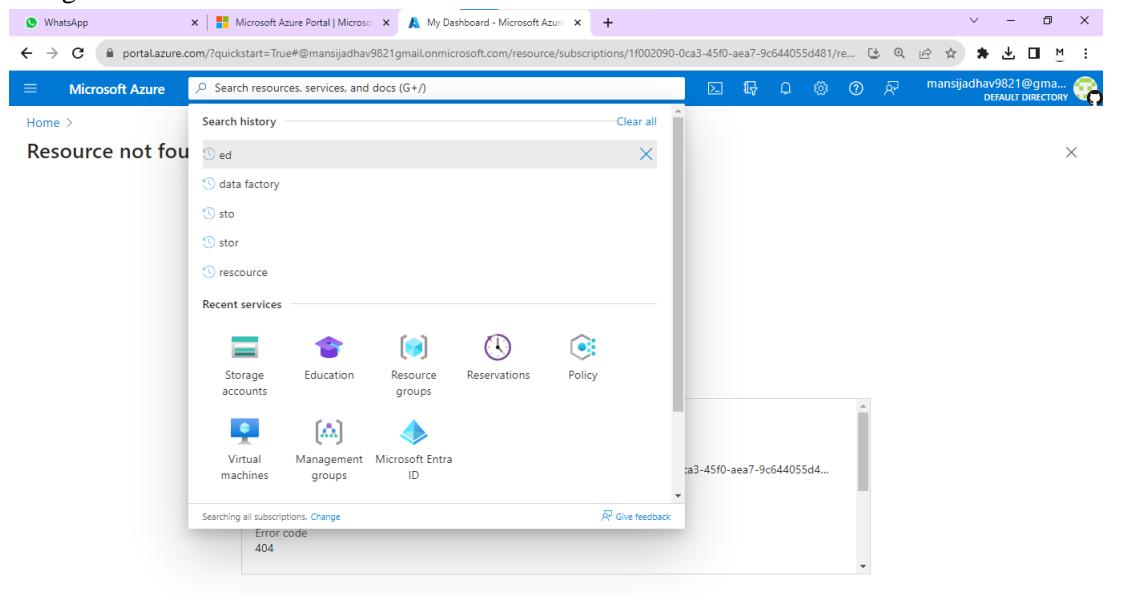


PROJECT DESIGN

So first thing we will open azure portal and login. Open azure portal <https://azure.microsoft.com/en-in/get-started/azure-portal> and once you login in it will be as shown below.



So First we have to do is ingest our data to data external. So we need two things one is Data Factory to create simple pipeline and second is Gen2 Which is where we will store raw data so lets start with it so we will these two things individually so first we will create Storage Account So in search bar search storage account so click on it and create account.



Storage account is created. Name of the storage account should be globally unique. Here performance is Standard as if is for General purpose v2 account and redundancy is Geo-Redundant Storage means if you want to replicate your data across different data centers or regions.

In advanced we will enable hierarchical Namespace means against all of the files that you store inside the container of your storage account will be available in hierarchical format the way it stores any data to any abject storage such as S3 or Google or just Azure Block Storage they store the entire data as an object but once you enable it you can access your data just like simple directory the way you do it on local computer so it is important to select the hierarchical namespace when we are creating storage account. And then review and create

Storage account created

The screenshot shows the Azure Storage accounts overview page. At the top, there are navigation links for Home, Storage accounts, and a search bar. Below the header is a toolbar with actions like Create, Restore, Manage view, Refresh, Export to CSV, Open query, Assign tags, and Delete. There are also filter and grouping options. The main table displays one record:

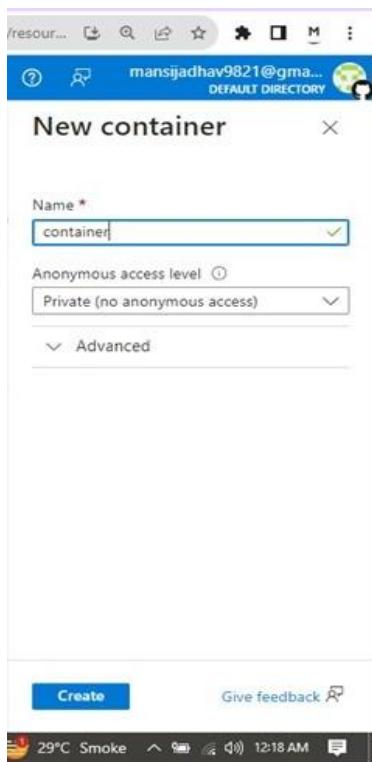
Name	Type	Kind	Resource group	Location	Subscription
tokyoaac	Storage account	StorageV2	olympic-data	Central India	Azure for Students

At the bottom of the page, there are pagination controls (Page 1 of 1) and a feedback link.

So these is the panel of storage account .here in overview we have all details about storage account its ressource group, subscription and many more. In azure storage we have container where you can store data as an object. Now we will create container in storage account.

The screenshot shows the Azure Storage account 'tokyoaac' details page. The left sidebar includes links for Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage browser, Data storage (Containers, File shares, Queues, Tables), and Security + networking. The main content area has tabs for Properties, Monitoring, Capabilities (5), Recommendations (0), Tutorials, and Tools + SDKs. The 'Containers' tab is selected under Data storage. The 'Data Lake Storage' and 'Security' sections are also visible.

Click on container and write name of the container and then create it



Click on Container and there we will create 2 folders so we will click on add directory .Now we will add 2 directory for raw where there will be raw data as it is from data source that we extract and transformed data where we will have transform data using Database transformation

A screenshot of the Microsoft Azure portal showing the storage account overview page. The URL is 'portal.azure.com/?quickstart=True#view/Microsoft_Azure_Storage/ContainerMenuBlade~/overview/storageAccountId/%2Fsubscriptions%2F9042566f-1...'. The 'Containers' tab is selected. The interface includes a search bar for blobs and a table with columns: Name, Modified, Access tier, Archive status, and Blob type. The table shows 'No results'.

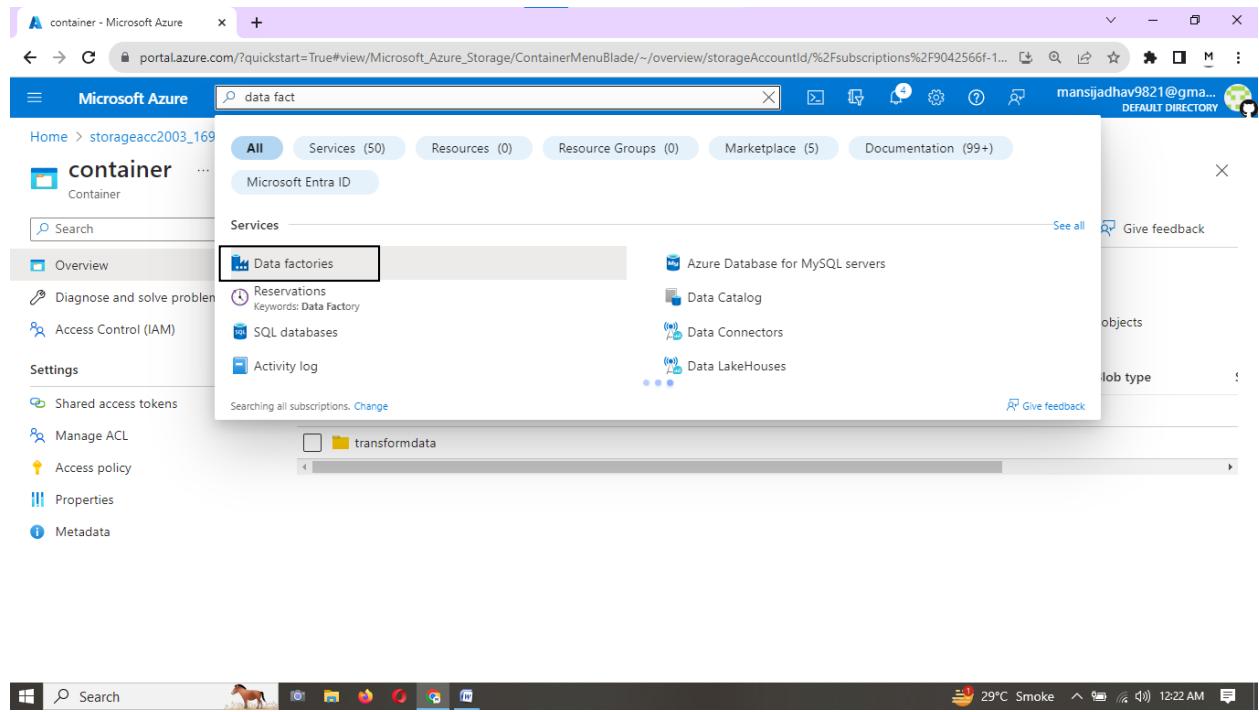
Adding of raw data

The screenshot shows the Microsoft Azure Storage Container settings page. On the left, there's a sidebar with options like Overview, Diagnose and solve problems, Access Control (IAM), Settings (Shared access tokens, Manage ACL, Access policy, Properties, Metadata), and a search bar. The main area shows a table with columns Name, Modified, and Access tier, which is currently empty with a message 'No results'. On the right, a modal window titled 'Add Directory' is open, prompting for a 'Name' (with 'rawdata' entered) and other details. At the bottom right of the modal are 'Save' and 'Give feedback' buttons.

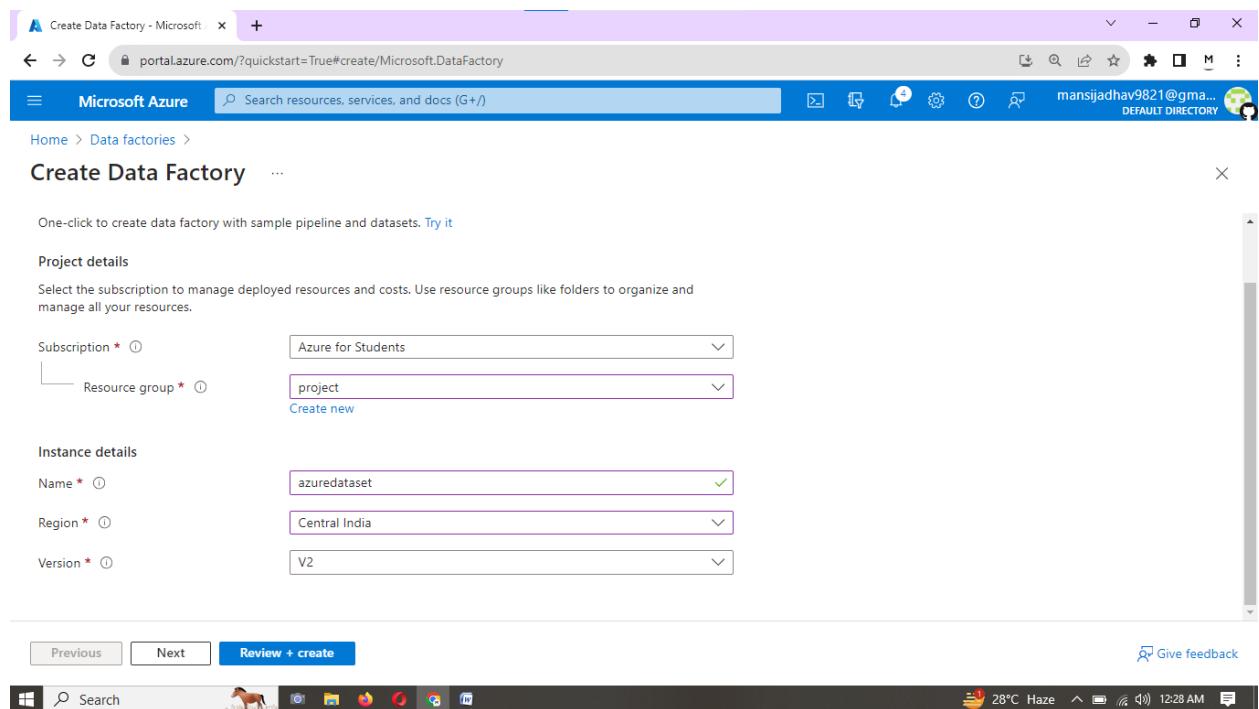
For transformed data.

This screenshot is similar to the previous one but shows the state after the 'rawdata' directory has been added. The table now lists a single entry: 'rawdata'. The rest of the interface, including the sidebar, authentication method, location, and the 'Add Directory' modal, are identical to the first screenshot.

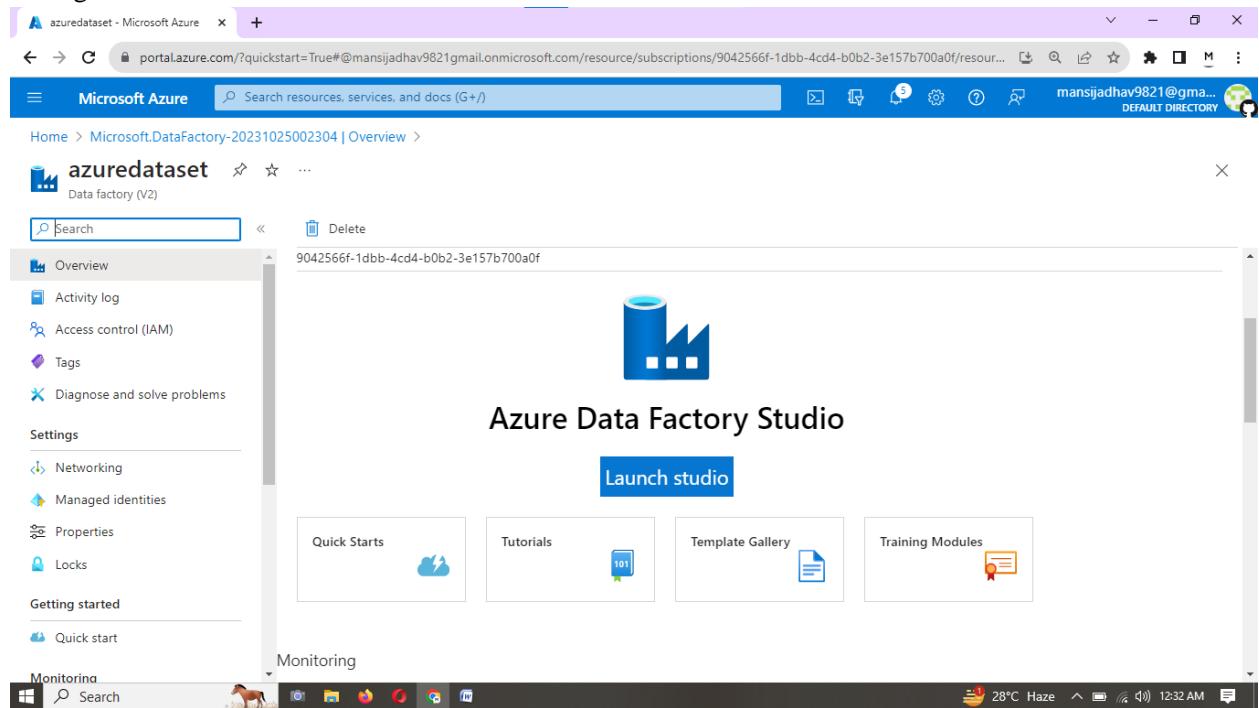
Now we have set our Azure data Lake now we have to copy the data from data source and put data on Target location and for that we need to use Azure Data Factory so search for Data Factory



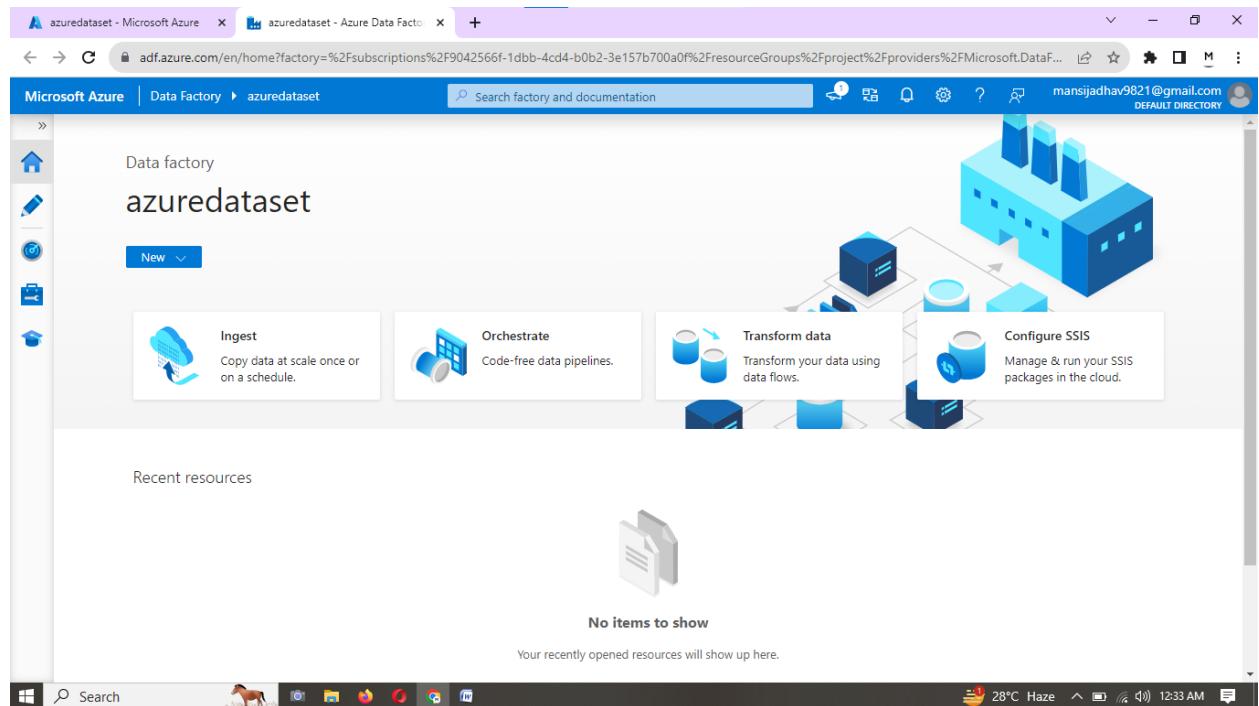
This is the console and click on Create and it will redirect to page. Now resource group will be same and then review and create



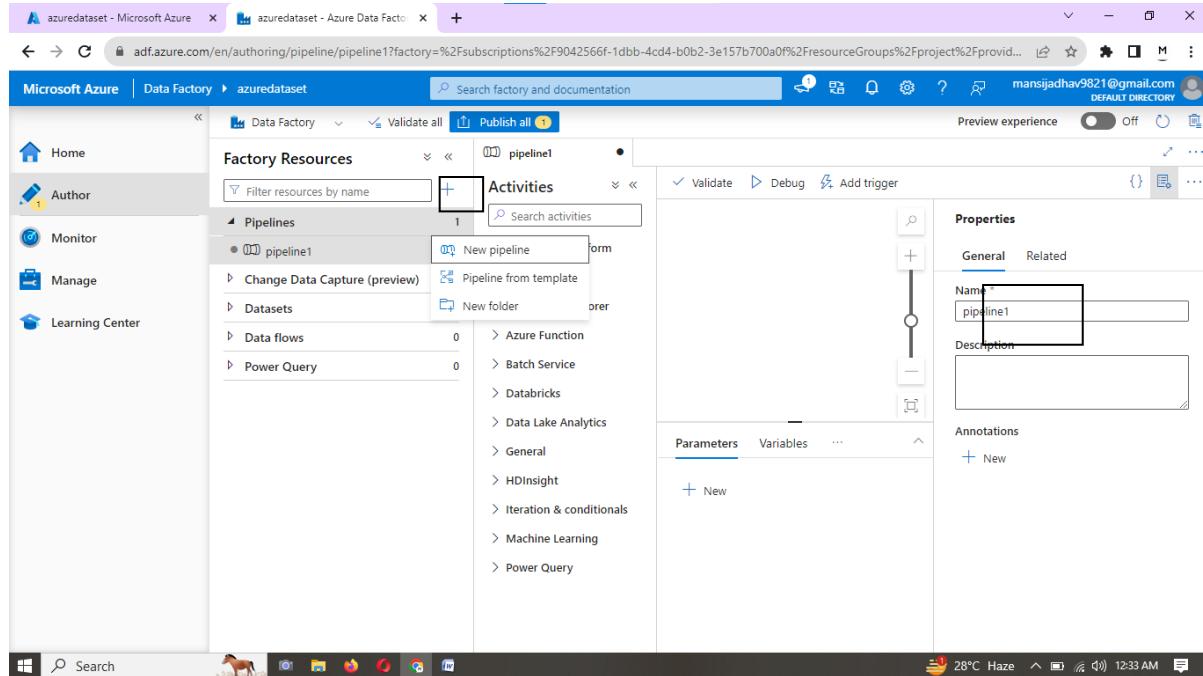
Then go to resource and Launch Studio



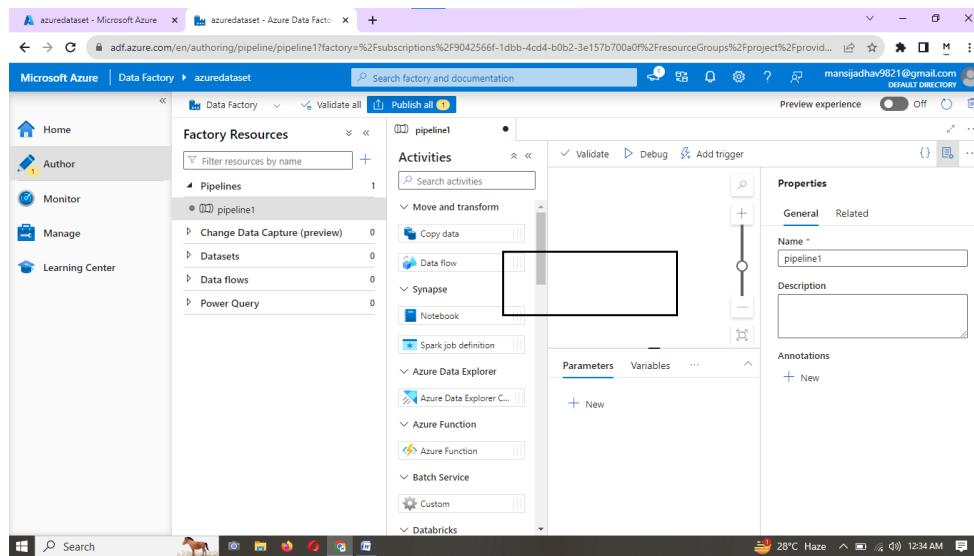
This is the Azure Data set ie Data factory panel. Here we can create Pipeline to extract data from different sources and upload that data on their Target Location so we have ingest option and other options also. In home all data factory are shown .



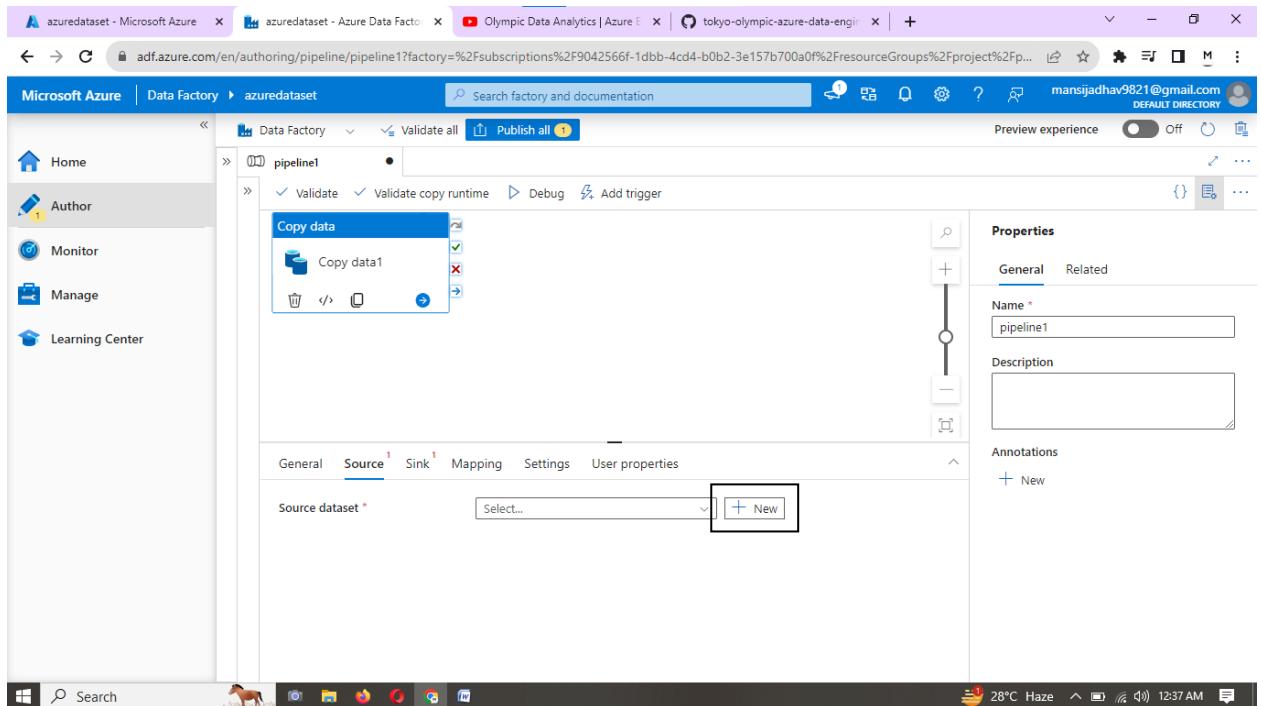
So we will click on Author where we actually create pipelines. Then click on plus sign and then select pipeline and we will create simple pipeline. Give the name for pipeline



So this is the pipeline window. here we have activities and there we will select move and transform as we have to copy the data and drag copy data in window. So first we will copy the data from our API or this data source to our location which is data storage this is our copy activity inside the copy activity we want to give the source where our data is actually stored and then sink where we want to load our data so lets do it. All the data is uploaded on github so you can refer that. Using Github repository as our data source we will extract data from repository and load our data on our azure location so we have five files available.

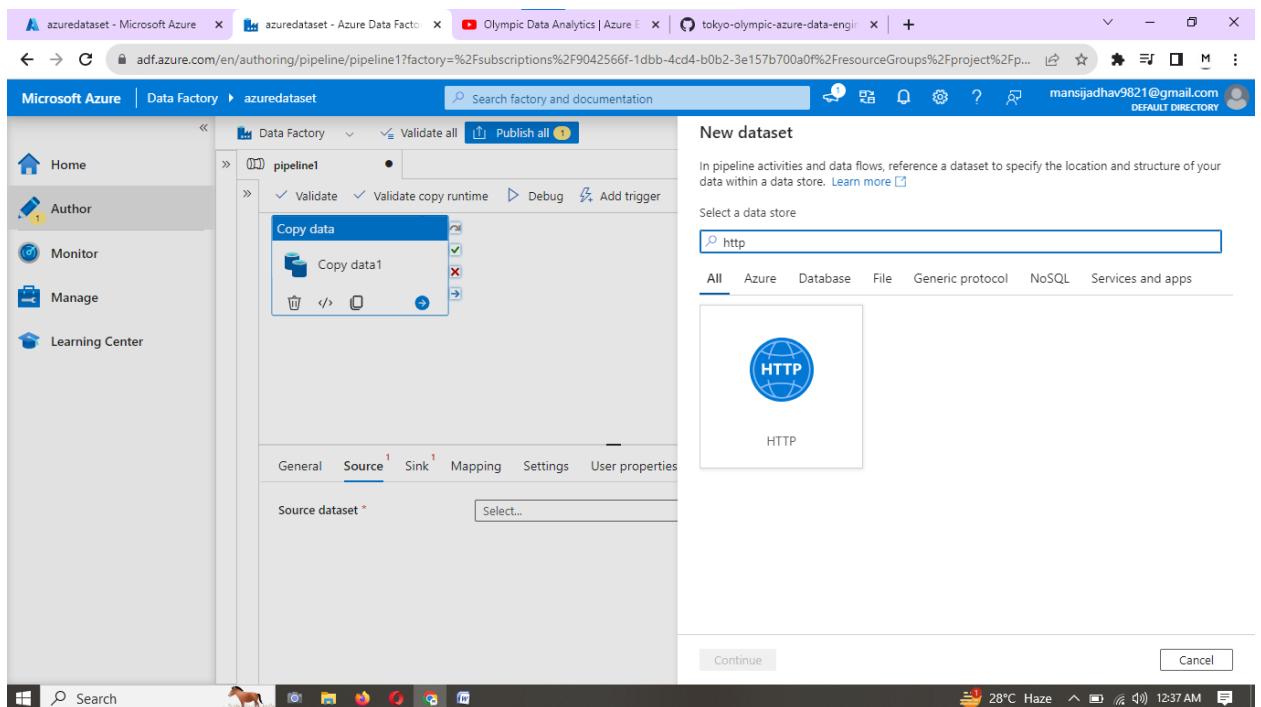


Now here select Source and click on New



The screenshot shows the Microsoft Azure Data Factory pipeline editor. A pipeline named 'pipeline1' is open, containing a single 'Copy data' activity named 'Copy data1'. The 'Source' tab is selected in the activity configuration pane. A red box highlights the 'Select...' button in the 'Source dataset' dropdown. The properties pane on the right shows the pipeline name as 'pipeline1'.

Here we select dataset as HTTP. We have multiple datasets to access data we select HTTP because this raw data is accessed through HTTP server.



The screenshot shows the 'New dataset' dialog in the Microsoft Azure Data Factory pipeline editor. The 'Select a data store' dropdown is set to 'http'. The 'HTTP' option is selected and highlighted with a red box. Other options like 'All', 'Azure', 'Database', etc., are visible at the bottom. The 'Continue' button is at the bottom right.

The file format is csv file so select DelimitedText and continue

The screenshot shows the Azure Data Factory pipeline configuration interface. On the left, the navigation bar includes 'Home', 'Author' (selected), 'Monitor', 'Manage', and 'Learning Center'. In the center, under 'pipeline1', there is a 'Copy data' activity named 'Copy data1'. The 'Source' tab is selected. To the right, a 'Select format' dialog box is open, titled 'Choose the format type of your data'. It displays various file formats with icons: Avro, Binary, DelimitedText (selected), Excel, JSON, ORC, XML, and others. At the bottom of the dialog are 'Continue' and 'Back' buttons.

Now this is the GitHub account where we have our dataset and we have 5 files so we have to copy all the data of 5 files so first we will do for Athletes.

The screenshot shows a GitHub repository page for 'tokyo-olympic-azure-data-engineering-project'. The repository name is 'data'. The left sidebar shows a file tree with 'main' and 'data' directories. The 'data' directory contains 'Athletes.csv', 'Coaches.csv', 'EntriesGender.csv', 'Medals.csv', 'Teams.csv', 'README.md', and 'Tokyo Olympic Transformation.ip...'. The main content area displays a table of files in the 'main' directory:

Name	Last commit message	Last commit date
..		
Athletes.csv	Add files via upload	3 months ago
Coaches.csv	Add files via upload	3 months ago
EntriesGender.csv	Add files via upload	3 months ago
Medals.csv	Add files via upload	3 months ago
Teams.csv	Add files via upload	3 months ago

So click on Athletes and then at the right bottom up we have raw so click on it

Azure Data Studio - Microsoft Azure | azuredataset - Azure Data Factory | Olympic Data Analytics | Azure | tokyo-olympic-azure-data-engineering-project | GitHub - darshilparmar/tokyo-olympic-azure-data-engineering-project/blob/main/data/Athletes.csv

github.com/darshilparmar/tokyo-olympic-azure-data-engineering-project/blob/main/data/Athletes.csv

Files

main

Go to file

data

Athletes.csv

Coaches.csv

EntriesGender.csv

Medals.csv

Teams.csv

README.md

Tokyo Olympic Transformation.ip...

Preview | Code | Blame | 11086 lines (11086 loc) · 409 KB

darshilparmar Add files via upload

a1a38a8 · 3 months ago · History

Raw

Search this file

1	PersonName	Country	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
		Australia	Rhythmic Gymnastics

https://github.com/darshilparmar/tokyo-olympic-azure-data-engineering-project/raw/main/data/Athletes.csv

28°C Haze 12:40 AM

After clicking on it we get raw data so the URL is the base link. So copy that link and paste it in Base link

Azure Data Studio - Microsoft Azure | azuredataset - Azure Data Factory | Olympic Data Analytics | Azure | raw.githubusercontent.com/darshilparmar/tokyo-olympic-azure-data-engineering-project/main/data/Athletes.csv

raw.githubusercontent.com/darshilparmar/tokyo-olympic-azure-data-engineering-project/main/data/Athletes.csv

PersonName,Country,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

ABBOTT Emily,Australia,Rhythmic Gymnastics

ABBOTT Monica,United States of America,Baseball/Softball

ABDALLA Abubaker Haydar,Qatar,Athletics

ABDALLA Maryam,Egypt,Artistic Swimming

ABDALLAH Shabd,Egypt,Artistic Swimming

ABDALRASOUD 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

ABDELAHGOUD Mohamed,Egypt,Judo

ABDELMOHTALEB Diaeldin Kamal Gouda,Egypt,Wrestling

ABDELRAHMAN Ihab,Egypt,Athletics

ABDELSALAH Mohamed,Egypt,Football

ABDELSALAH Nour,Egypt,Taekwondo

ABDELAHED 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 Fatimah Nabaha,Maldives,Badminton

ABDULRAHID Saudi,Saudi Arabia,Football

ABDULRAHMAN Khaled,Germany,Boxing

ABDULRAEV Gulomjon,Uzbekistan,Wrestling

ABDULRAEV Muminjon,Uzbekistan,Wrestling

ABDULRAHMAN Ervin,Indonesia,Weightlifting

ABDULILIM Ilfat,Kazakhstan,Archery

ABDULREHDA Mohamed,Bahrain,Handball

ABDURAIHON Elmur,Uzbekistan,Boxing

ABDURAKHIMOV Resuljon,Uzbekistan,Artistic Gymnastics

28°C Haze 12:40 AM

After copy create the linked service

The screenshot shows the Microsoft Azure Data Factory pipeline editor. A 'Copy data' activity named 'Copy data1' is selected in the pipeline. On the right, a 'New linked service' dialog is open. The 'Name' field is set to 'Athleteshttp'. The 'Connect via integration runtime' dropdown is set to 'AutoResolveIntegrationRuntime'. The 'Base URL' field contains the value 'https://raw.githubusercontent.com/darshilparmar/tokyo-olympic-azure-data-engineering-'. Below it, a warning message states: 'Information will be sent to the URL specified. Please ensure you trust the URL entered.' Under 'Server Certificate Validation', the 'Enable' radio button is selected. The 'Authentication type' is set to 'Anonymous'. The 'Auth headers' section has a '+ New' button. At the bottom of the dialog are 'Create' and 'Cancel' buttons, along with a 'Test connection' link.

Now we will give the path for data so click on ok.

The screenshot shows the Microsoft Azure Data Factory pipeline editor. The 'Copy data' activity 'Copy data1' is selected. On the right, a 'Set properties' dialog is open for the 'Source' tab. The 'Name' field is 'DelimitedText1'. The 'Linked service' dropdown is set to 'Athleteshttp'. The 'Relative URL' field is empty. Under 'Import schema', the 'From connection/store' radio button is selected. The 'OK' button is visible at the bottom of the dialog.

Now click on preview data if the data occurs means you are on right track

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the navigation menu includes Home, Author (selected), Monitor, Manage, and Learning Center. The main area displays a table titled "Preview data" under the heading "Object". The table has columns: PersonName, Country, and Discipline. The data consists of 10 rows:

	PersonName	Country	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

On the right, the "Properties" panel shows the pipeline name as "pipeline1". The status bar at the bottom indicates "28°C Haze" and the time "12:42 AM".

Now we will sink so click on new

The screenshot shows the Microsoft Azure Data Factory interface. The pipeline1 dataset is selected. In the center, a "Copy data" step is configured. The "Sink" tab is active, with a dropdown menu showing "Select..." and a "+ New" button. The "Properties" panel on the right shows the pipeline name as "pipeline1". The status bar at the bottom indicates "28°C Haze" and the time "12:43 AM".

And here we will select Azure Data Lake Storage Gen2

The screenshot shows the Microsoft Azure Data Factory pipeline editor. A pipeline named 'pipeline1' is open, containing a single 'Copy data' activity named 'Copy data1'. The 'Sink' tab is selected. On the right, a 'New dataset' dialog is displayed, prompting the user to 'Select a data store'. A grid of data store icons is shown, with 'Azure Data Lake Storage Gen2' highlighted. Below the grid are 'Continue' and 'Cancel' buttons.

Same here also we select csv file and then continue

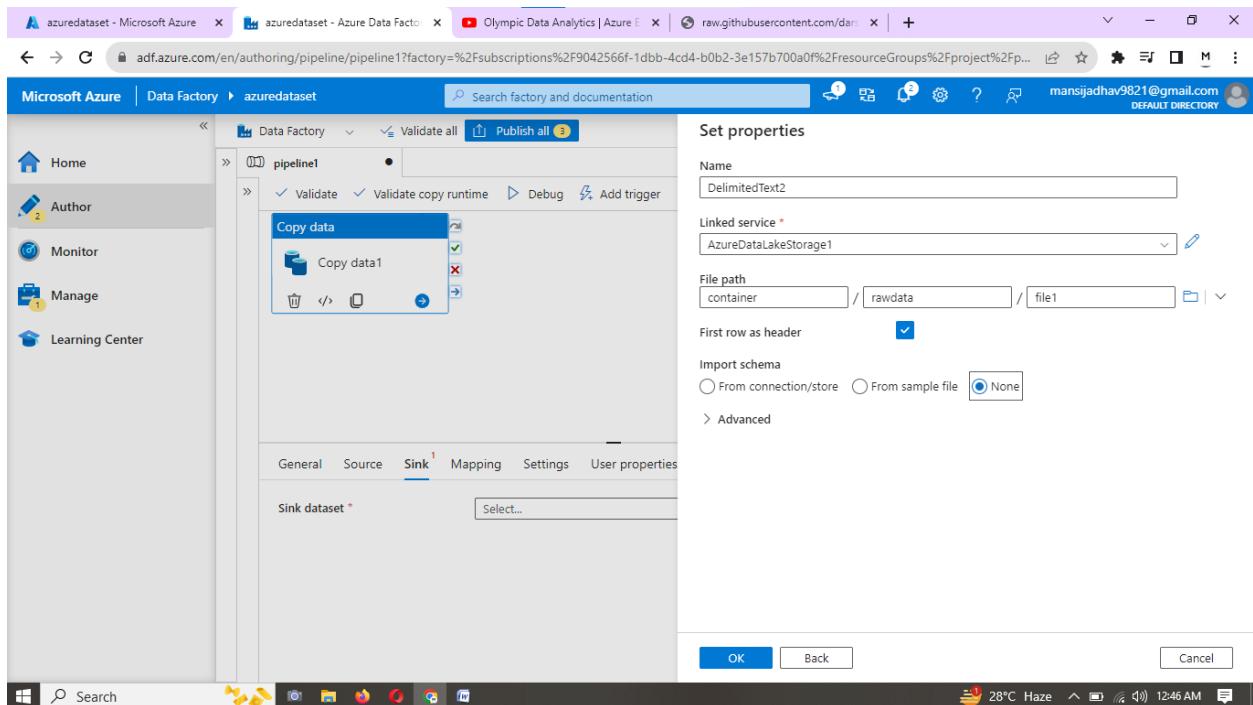
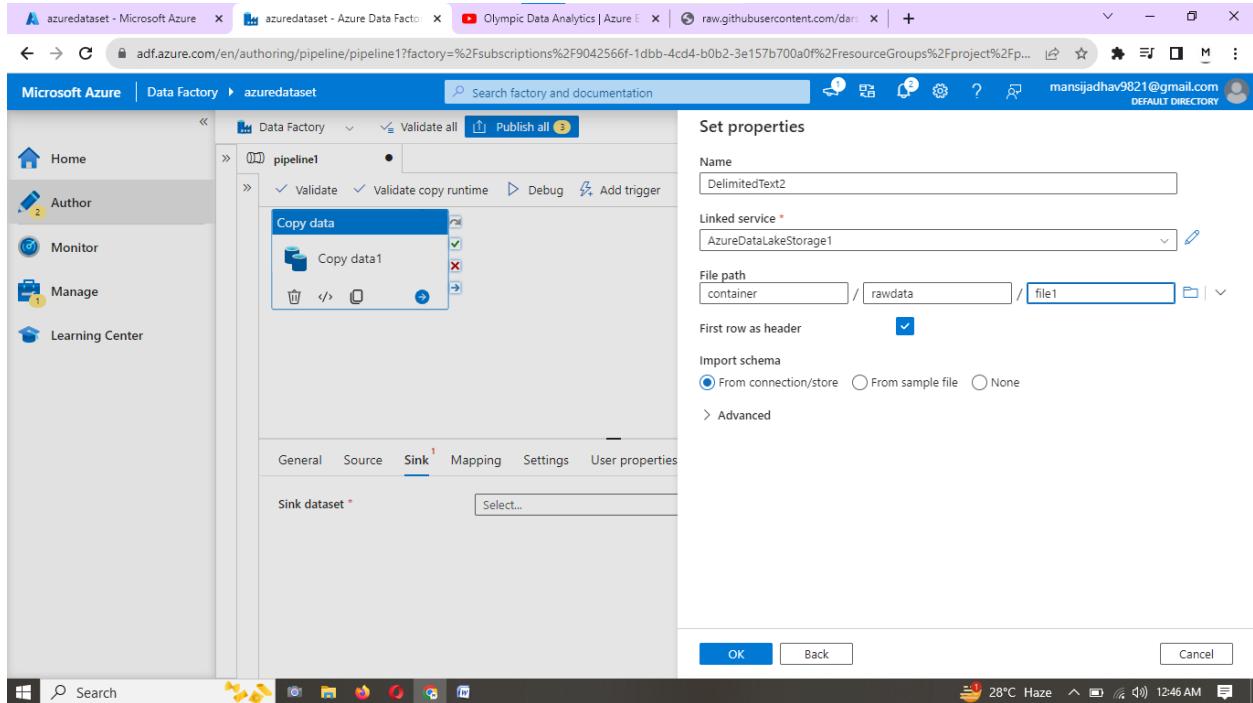
The screenshot shows the Microsoft Azure Data Factory pipeline editor. The 'Sink' tab is selected for the 'Copy data1' activity. On the right, a 'Select format' dialog is open, titled 'Choose the format type of your data'. It displays six options: Avro, Binary, DelimitedText, JSON, ORC, and Parquet. The 'CSV' icon is highlighted. Below the grid are 'Continue' and 'Back' buttons.

Here select the subscription and storage account we have created and then create it

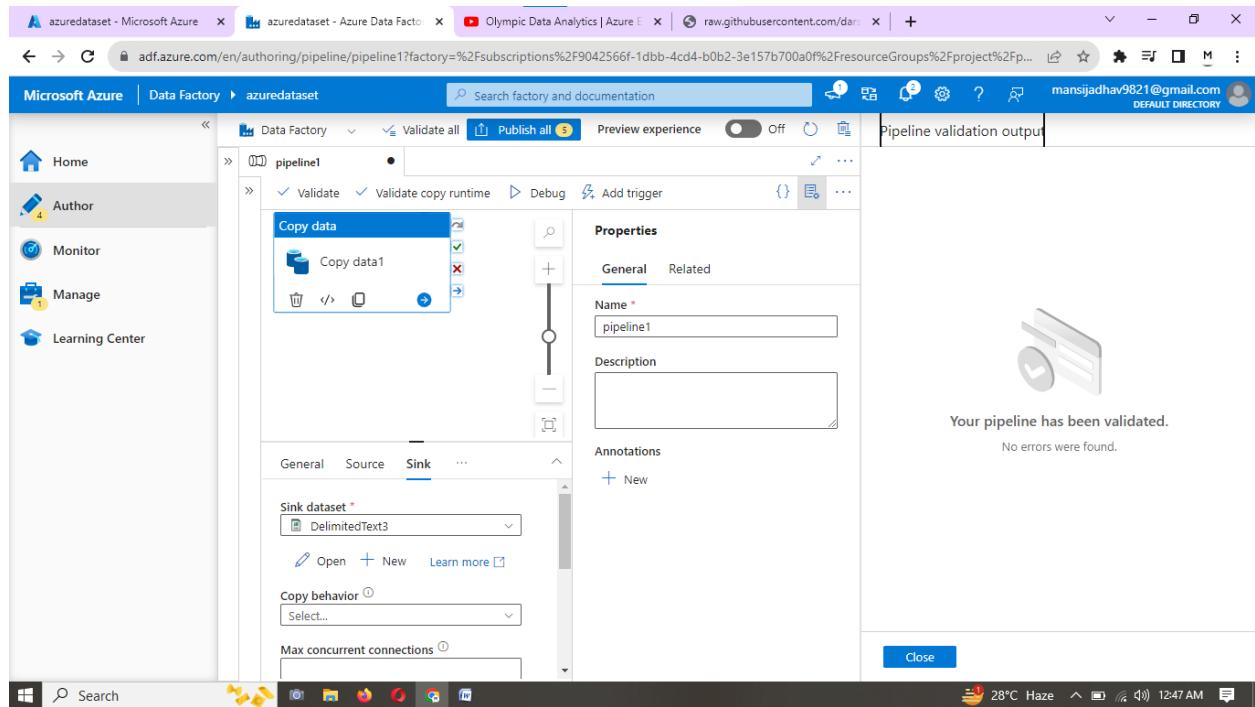
The screenshot shows the Microsoft Azure Data Factory pipeline editor. A 'Copy data' activity named 'Copy data1' is selected in the pipeline. On the right, a 'New linked service' dialog is open. The 'Name' field is set to 'AzureDataLakeStorage1'. Under 'Connect via integration runtime', 'AutoResolveIntegrationRuntime' is selected. The 'Authentication type' is set to 'Account key'. The 'Account selection method' is set to 'From Azure subscription', with 'Azure for Students (9042566f-1dbb-4cd4-b0b2-3e157b700a0f)' selected. The 'Storage account name' is 'storageacc2003'. A 'Test connection' button is visible at the bottom right of the dialog.

The screenshot shows the Microsoft Azure Data Factory pipeline editor. A 'Copy data' activity named 'Copy data1' is selected in the pipeline. On the right, a 'Browse' dialog is open, showing a file tree under 'Root folder > container'. It lists two items: 'rawdata' and 'transformdata'. The 'rawdata' item is highlighted with a red border. An 'OK' button is visible at the bottom right of the dialog.

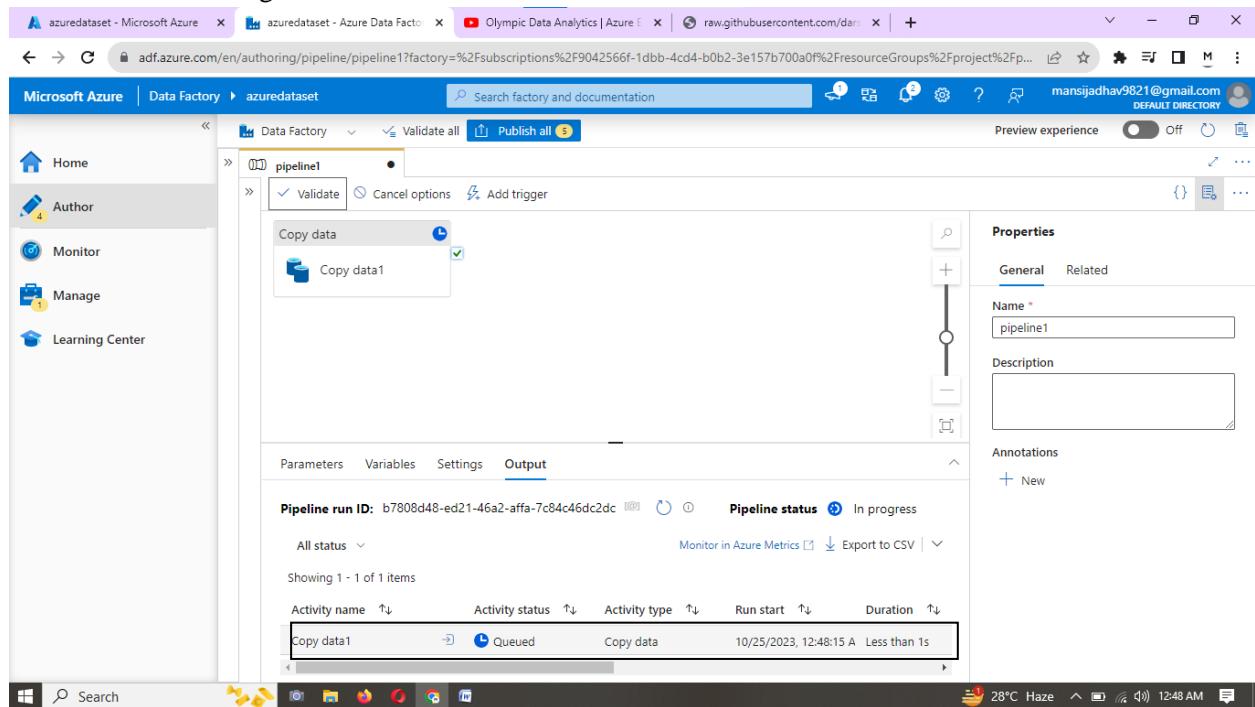
Now here we will browse the location ie container and there raw data and click on ok and give the file name.



After that click on Validate and if no error occurs means we have done all right.



So now we will Debug



Before the raw data was empty no file was there

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, a sidebar lists 'Overview', 'Diagnose and solve problems', and 'Access Control (IAM)'. The main area displays a table with columns: Name, Modified, Access tier, Archive status, and Blob type. A single entry, '[.]', is listed under the 'Name' column. At the top, there are buttons for 'Upload', 'Add Directory', 'Refresh', 'Rename', 'Delete', 'Change tier', 'Acquire lease', 'Break lease', and 'Give feedback'. A search bar at the top right contains the prefix 'Search blobs by prefix (case-sensitive)' and a toggle switch for 'Show deleted objects'.

Now it is successful and go and check in azure portal Athletes file will be shown there

The screenshot shows the Microsoft Azure Data Factory pipeline status page. The left sidebar lists 'Factory Resources' with sections for Pipelines (1), Datasets (5), Data flows (0), and Power Query (0). The main area shows a pipeline named 'pipeline12' with a status of 'Succeeded'. The 'Output' tab is selected, displaying a table with columns: Activity name, Activity status, Activity type, and Run start. One item is listed: 'data1' with 'Succeeded' status, 'Copy data' activity type, and a run start time of '10/25/2023, 1:06:43 AM'. The 'Properties' panel on the right shows the pipeline's name as 'pipeline12' and its description as empty. It also includes tabs for 'General' and 'Related' and an 'Annotations' section with a '+ New' button.

See we have athletes file in container

The screenshot shows the Microsoft Azure Storage account interface. In the center, there is a table listing blobs. The first row shows a folder named '[.]'. Below it, a single blob named 'Athletes' is listed with the following details:

Name	Modified	Access tier	Archive status	Blob type
Athletes	10/25/2023, 1:06:55 ...	Hot (Inferred)		Block blob

Now we will do same steps for other files perform same step

The screenshot shows the Microsoft Azure Data Factory pipeline configuration. On the left, the 'Factory Resources' sidebar lists 'Pipelines' with one item named 'pipeline12'. The main area displays the 'pipeline12' pipeline. A 'Copy data' activity is selected, showing its configuration. The 'Sink' tab is active, and the 'Sink dataset' dropdown is set to 'Select...'. To the right, a 'Set properties' panel is open, showing the following configuration:

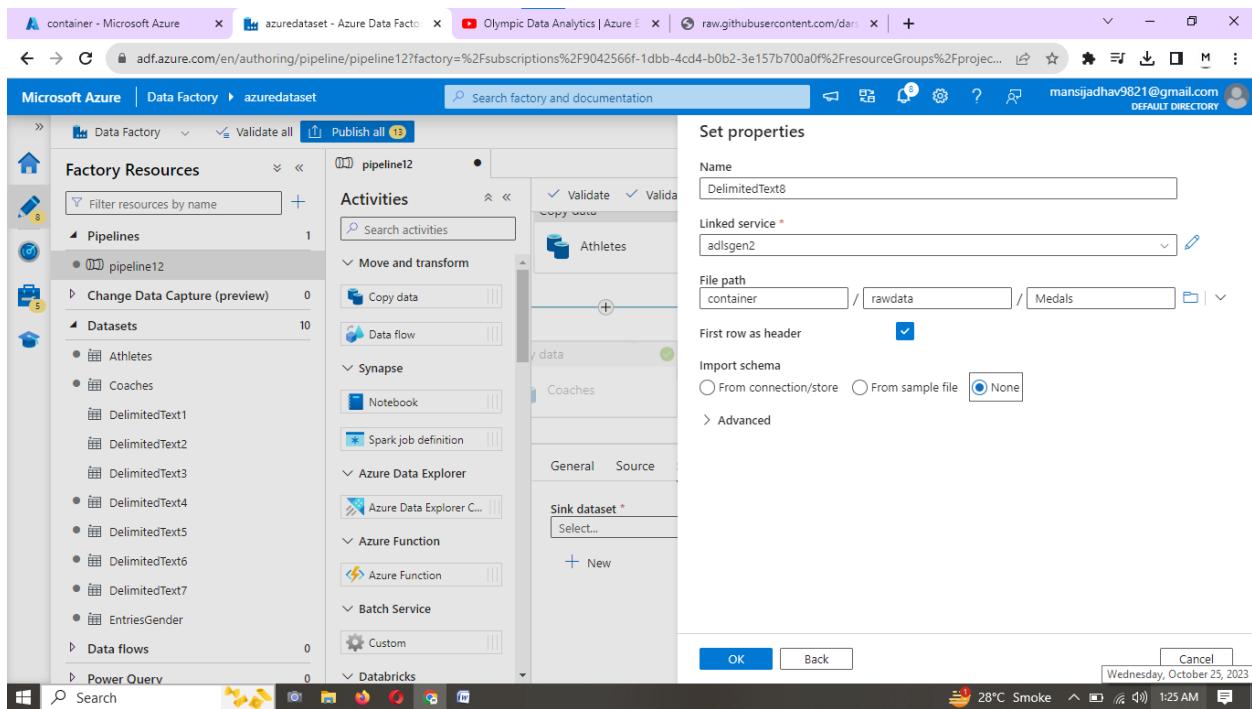
- Name:** DelimitedText5
- Linked service:** adlsgen2
- File path:** container / rawdata / File name
- First row as header:** checked
- Import schema:** From connection/store (radio button selected)

At the bottom of the panel are 'OK', 'Back', and 'Cancel' buttons.

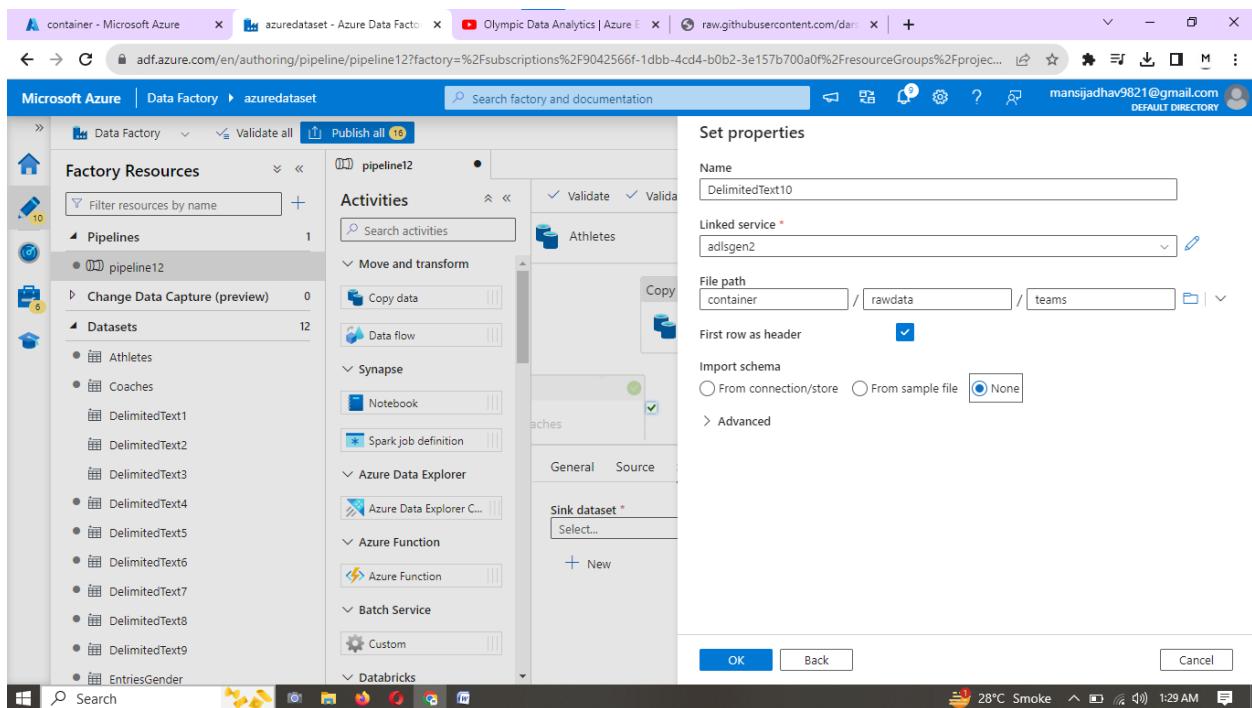
The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (pipeline12), 'Change Data Capture (preview)', 'Datasets', 'Data flows', and 'Power Query'. The main workspace displays 'pipeline12' with two 'Copy data' activities: 'data1' and 'data2'. The 'Properties' pane on the right shows the pipeline name as 'pipeline12' and its status as 'Succeeded'. The 'Annotations' pane has a '+ New' button.

The screenshot shows the Microsoft Azure Storage container blade for 'container'. The left sidebar includes 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Settings' (with options like 'Shared access tokens', 'Manage ACL', 'Access policy', 'Properties', and 'Metadata'), and a search bar. The main area shows a list of blobs: 'Athletes' and 'Coaches'. The 'Athletes' blob was modified on 10/25/2023 at 1:14:26 PM and is in the 'Hot (Inferred)' access tier. The 'Coaches' blob was modified on 10/25/2023 at 1:14:25 PM and is also in the 'Hot (Inferred)' access tier. Both are 'Block blob' types. A 'Search blobs by prefix (case-sensitive)' input field and a 'Show deleted objects' toggle are also present.

For Medals



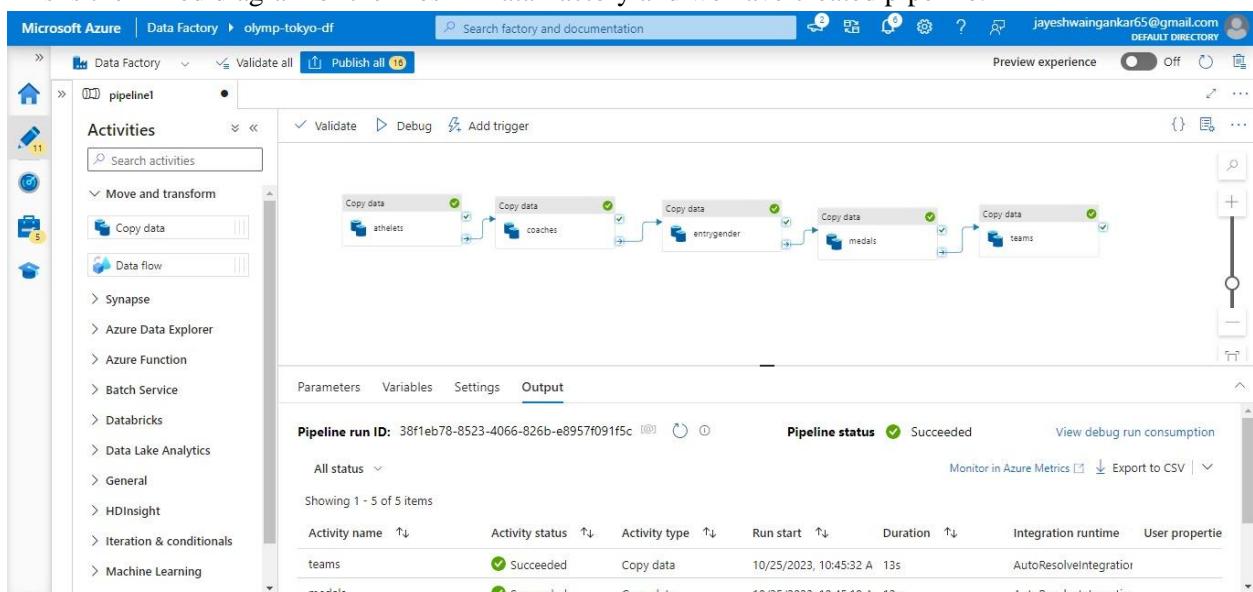
For Teams

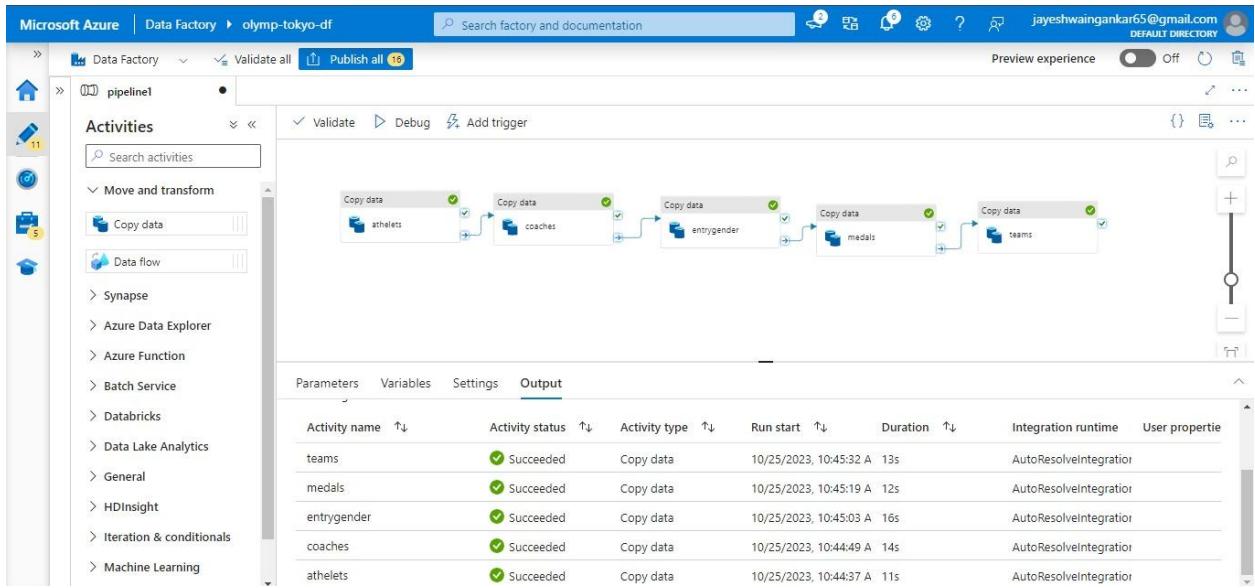


Here we have all our 5 files in azure container

The screenshot shows the Azure Storage Explorer interface for the 'tokyoolymp' storage account. The 'Containers' section is selected, and the 'raw-data' container is shown. The table lists five CSV files: 'athletes.csv', 'coaches.csv', 'entrygen.csv', 'medals.csv', and 'teams.csv'. Each file has its name, modified date (10/25/2023), access tier (Hot (Inferred)), archive status (Not yet archived), blob type (Block blob), size (e.g., 408.68 KiB for athletes.csv), and lease state (Available). A search bar at the top allows filtering by prefix.

This is the linked diagram of the files in Data Factory and we have created pipeline.





Now we will do Azure Databricks part where we will write transformation code. So search databricks in azure portal. Give the and and then review and create.

The screenshot shows the 'Create an Azure Databricks workspace' wizard. The current step is 'Set up workspace'. It requires selecting a subscription ('Subscription *') and a resource group ('Resource group *'). The selected subscription is 'Azure for Students' and the resource group is 'olympic-data'. Below these, there are fields for 'Instance Details': 'Workspace name *' (set to 'tokyoldb'), 'Region *' (set to 'Central India'), and 'Pricing Tier *' (set to 'Premium (+ Role-based access controls)'). A note indicates that the recommended pricing tier was selected. There is also a field for 'Managed Resource Group name' with a placeholder 'Enter name for managed resource group'. At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next : Networking >'.

After deployment click on go to resource and launch workspace

The screenshot shows the Microsoft Azure Databricks Service Overview page for a resource named 'tokyoldb'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Virtual Network Peerings, Encryption, Networking, Properties, Locks, Monitoring, Diagnostic settings, and Automation. The main content area displays the 'Essentials' section with the following details:

Setting	Value
Status	Active
Resource group	olympic-data
Location	Central India
Subscription	Azure for Students
Subscription ID	8e31541e-daff-4840-855e-c1e8edaafbc
Tags (edit)	Add tags

Managed Resource Group: [databricks-rg-tokyoldb-q7d22ma44cyvm](#)
URL: <https://adb-8257815783853483.3.azuredatabricks.net>
Pricing Tier: [Premium \(+ Role-based access controls\) \(Click to change\)](#)

A large red 'Databricks' logo icon is centered below the essentials section, with a 'Launch Workspace' button underneath it.

Now go to Compute and then create compute

The screenshot shows the Databricks Compute page. The left sidebar is identical to the one in the previous screenshot, with the 'Compute' option selected. The main content area is titled 'Compute' and shows a table header for 'All-purpose compute' with columns: State, Name, Policy, Runtime, Active m..., Active co..., Active DB..., Source, Creator, Notebooks, and a gear icon. A search bar at the top allows filtering by 'Filter compute you have...' and 'Created by'. A 'Create with Personal Compute' dropdown menu is open, and a prominent blue 'Create compute' button is highlighted with a red box. Below the table, there is a large plus sign icon and the text 'No compute'. A sub-instruction reads: 'Create compute to run workloads from your notebooks and jobs. Learn more about best practices for compute configuration'. A final 'Create compute' button is located at the bottom of this section.

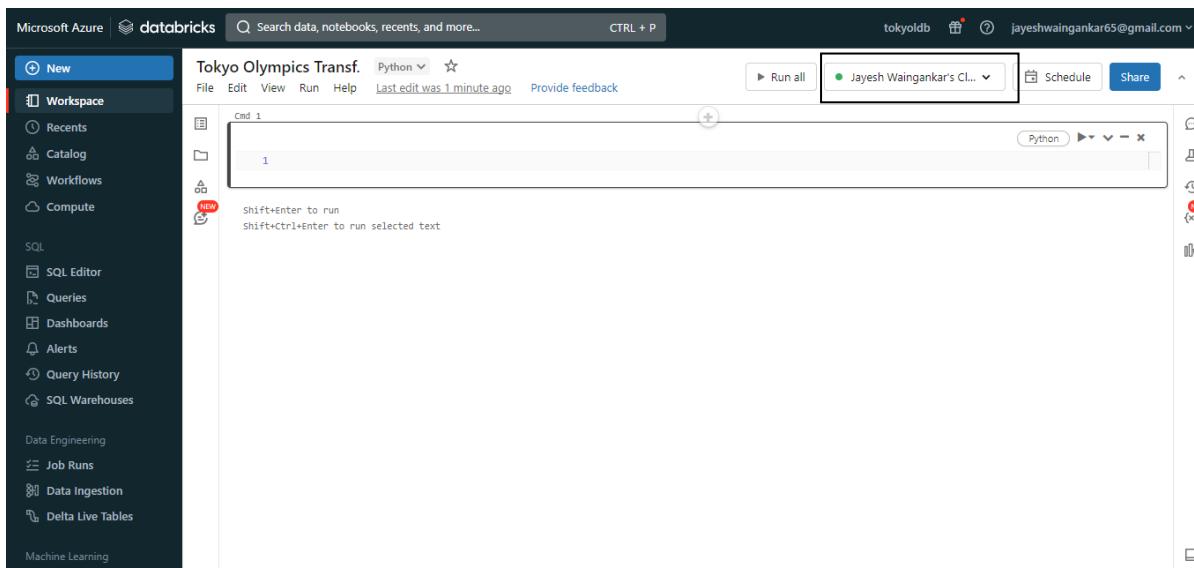
Now if we have multiple machines we can select MultiNode and if single we can select single node and then create compute.

The screenshot shows the Databricks Compute creation interface. On the left sidebar, under the 'Compute' section, 'Single' is selected. In the main panel, 'Single node' is chosen for access mode. The 'Summary' section indicates 1 Driver, 14 GB Memory, 4 Cores, Runtime 12.2.x-scala2.12, and Standard_DS3_v2 with 0.75 DBU/h. The 'Performance' section shows the runtime version as 12.2 LTS (Scala 2.12, Spark 3.3.2), and the 'Node type' is set to Standard_DS3_v2. A checkbox for 'Terminate after 120 minutes of inactivity' is checked. At the bottom are 'Create compute' and 'Cancel' buttons.

Here then go to new and then notebook and give the name for notebook.

The screenshot shows the Databricks workspace. On the left sidebar, 'New' is selected. In the center, a new notebook titled '10-25 11:09:25 Python' is shown. The notebook interface includes a code editor with a Python tab, run controls (Run all, Connect, Schedule, Share), and a preview area. The sidebar also lists other workspace sections like Workspace, Catalog, Workflows, Compute, SQL, and Machine Learning.

In Connect attach your cluster.



Now we have to create connection from this azure databricks to our azure data storage so that we can easily access the data so the steps to create connection between this is you need to mount this azure data lake storage to azure data factory so that you can access the file from it so mounting is basically attaching so just like we use USB cable to attach the hard disk physically we have to mount service to azure data factory. go to azure portal.

So in azure portal search app registration.

A screenshot of the 'Register an application' page in the Azure portal. The page title is 'Register an application ...'. It has a 'Name' field containing 'app1'. Below it is a section for 'Supported account types' with four options: 'Accounts in this organizational directory only (Default Directory only - Single tenant)' (selected with a radio button), '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)', and 'Personal Microsoft accounts only'. There is also a 'Help me choose...' link. At the bottom, there is a note about agreeing to 'Microsoft Platform Policies' and a 'Register' button.

After creating it Go to resource and then in certificates and secrets click on new certificate

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (0) Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value ⓘ	Secret ID
No client secrets have been created for this application.			

Give the name for client server and add it.

Add a client secret

Description: secretkey

Expires: Recommended: 180 days (6 months)

Add Cancel

Here copy the client id , tenant id in notepad .

Search

Overview Quickstart Integration assistant

Manage

- Branding & properties
- Authentication
- Certificates & secrets
- Token configuration
- API permissions
- Expose an API
- App roles
- Owners
- Roles and administrators

Search

Delete Endpoints Preview features

Essentials

Display name app1

Application (client) ID 7123a42e-3f22-4db7-be12-ae998cff5664

Object ID e0b1b44d-99ff-4163-84a6-e040c8395f8e

Directory (tenant) ID 91e74d04-e238-4aee-a2f3-8a47d23ee52c

Supported account types My organization only

Client credentials Add a certificate or secret

Redirect URLs Add a Redirect URI

Application ID URI Add an Application ID URI

Managed application in local directory app1

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

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

23°C Partly cloudy 12:12 AM

After creation of secret key copy the value in notepad

Search

Got feedback?

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (1) Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

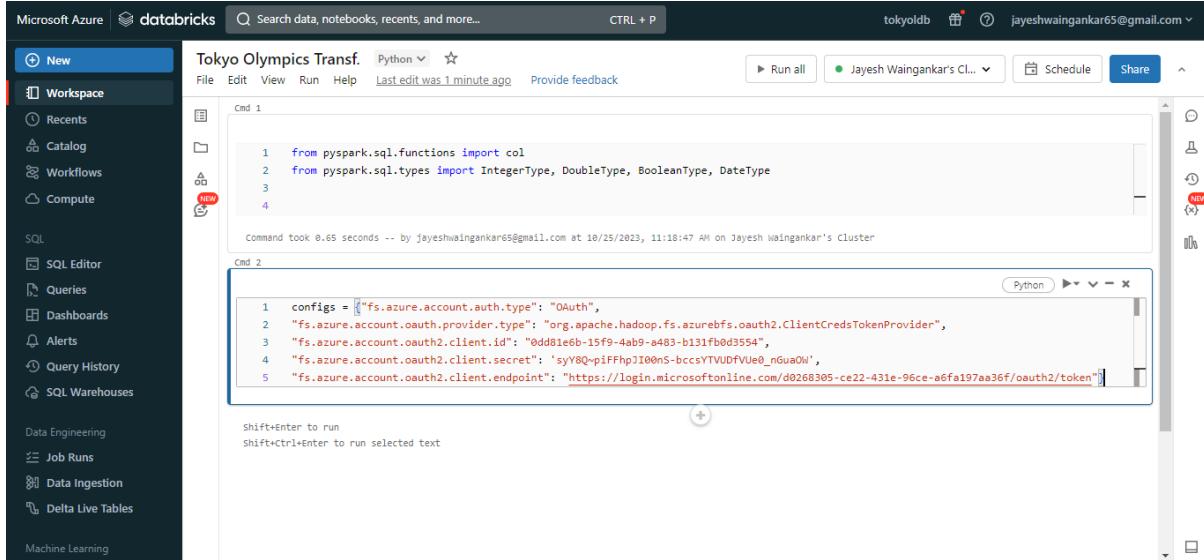
New client secret

Description	Expires	Value	Secret ID
secretkey	4/23/2024	Tob8Q~ScT-zkTCA5ZYHU1WvoKlajQ... 120c3d1-a780-4fdd-b7d3-c4d3573...	

Update application credentials Successfully updated application app1 credentials

23°C Partly cloudy 35 min to full charge

This is the code to run in databricks in the where we have tenant id copy that id here and where we have secret key copy the key here and then run. All the code are uploaded on Github link



The screenshot shows the Databricks workspace interface. On the left, there's a sidebar with various options like Recents, Catalog, Workflows, Compute, SQL, and Machine Learning. The main area has tabs for 'Cmd 1' and 'Cmd 2'. In 'Cmd 2', there is Python code:

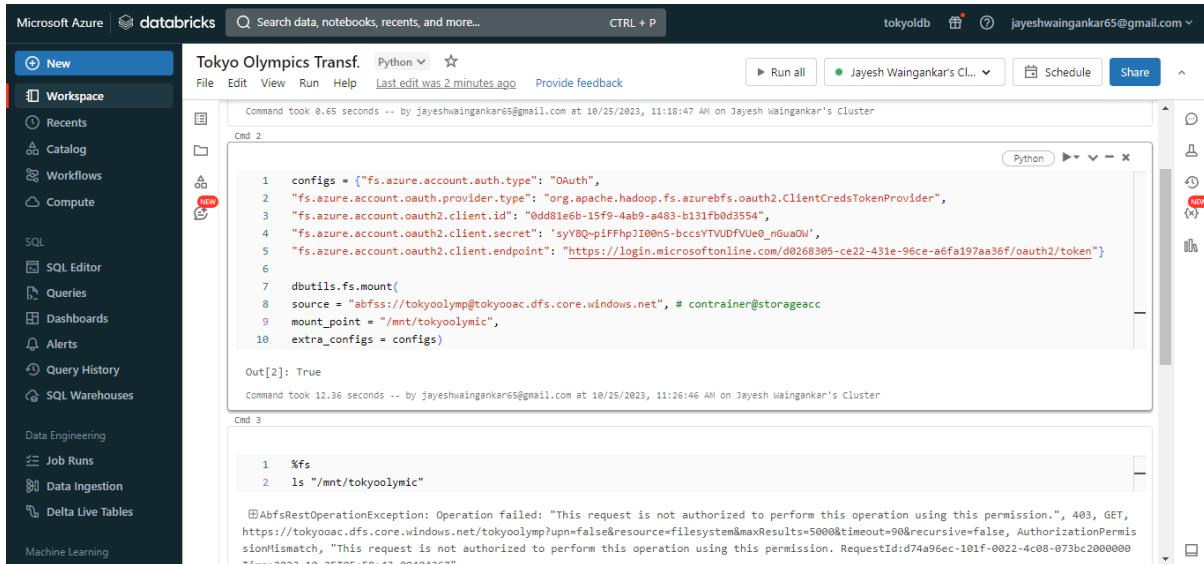
```

1 configs = {"fs.azure.account.auth.type": "OAuth",
2             "fs.azure.account.oauth.provider.type": "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
3             "fs.azure.account.oauth2.client.id": "0dd81e6b-15f9-4ab9-a483-b131fb0d3554",
4             "fs.azure.account.oauth2.client.secret": "syYQ0-pIfFhpJ100nS-bccsYTUVfUe0_nGuAOw",
5             "fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/d0268305-ce22-431e-96ce-a6fa197aa36f/oauth2/token"}

```

Below the code, it says 'Shift+Enter to run' and 'Shift+Ctrl+Enter to run selected text'. The status bar at the bottom indicates the command took 0.65 seconds.

before @ write the name of container ie [container@storageaccname.Now](#) operation fail as access is not give so that we will give access.



The screenshot shows the Databricks workspace interface. The sidebar and tabs are identical to the previous screenshot. In 'Cmd 2', the Python code is the same as before, but the 'source' line now includes the container name:

```

1 configs = {"fs.azure.account.auth.type": "OAuth",
2             "fs.azure.account.oauth.provider.type": "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
3             "fs.azure.account.oauth2.client.id": "0dd81e6b-15f9-4ab9-a483-b131fb0d3554",
4             "fs.azure.account.oauth2.client.secret": "syYQ0-pIfFhpJ100nS-bccsYTUVfUe0_nGuAOw",
5             "fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/d0268305-ce22-431e-96ce-a6fa197aa36f/oauth2/token"}
6
7 dbutils.fs.mount(
8   source = "abfss://tokyoolymp@tokyoac.dfs.core.windows.net", # contrainer@storageacc
9   mount_point = "/mnt/tokyoolympic",
10  extra_configs = configs)

```

The output of the command is 'Out[2]: True'. In 'Cmd 3', there is another Python command:

```

1 %fs
2 ls "/mnt/tokyoolympic"

```

The output of this command is a detailed error message from AbfsRestOperationException:

```

AbfsRestOperationException: Operation failed: "This request is not authorized to perform this operation using this permission.", 403, GET,
https://tokyoac.dfs.core.windows.net/tokyoolymp?upn=false&resource=filesystem&maxResults=5000&timeout=90&recursive=false, AuthorizationPermissionMismatch,
"This request is not authorized to perform this operation using this permission. RequestId:d74a96ec-101f-0022-4c08-073bc2000000
Time:2023-10-25T04:58:43Z"

```

Go to azure portal container >> Access Control

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information (jayeshwaingankar65@gmail.com, DEFAULT DIRECTORY). Below the navigation is a breadcrumb trail: Home > Storage accounts > tokyooac | Containers > tokyoolymp. The main title is "tokyoolymp | Access Control (IAM)". On the left, a sidebar menu lists: Overview, Diagnose and solve problems, Access Control (IAM), Settings, Shared access tokens, Manage ACL, Access policy, Properties, and Metadata. The "Access Control (IAM)" item is selected and highlighted in grey. The main content area has a header "Add role assignment". Below it are three sections: "Grant access to this resource", "View access to this resource", and "View deny assignments". Each section contains a brief description and a "View" or "Add role assignment" button.

Here select storage blob data contributor

The screenshot shows the "Add role assignment" page. At the top, there are tabs for "Role" (selected), "Members" (selected), and "Review + assign". A note states: "A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. Learn more". The "Assignment type" dropdown is set to "Job function roles". Under "Job function roles", "Privileged administrator roles" are listed. A note says: "Grant access to Azure resources based on job function, such as the ability to create virtual machines." A search bar contains "storage blob data contributor". The results table shows one entry: "Storage Blob Data Contributor" (Name), "Allows for read, write and delete access to Azure Storage blob containers and data" (Description), "BuiltInRole" (Type), "Storage" (Category), and a "View" link (Details). Below the table, it says "Showing 1 - of 1 results". Navigation buttons at the bottom include "Review + assign", "Previous", "Next", and "Feedback".

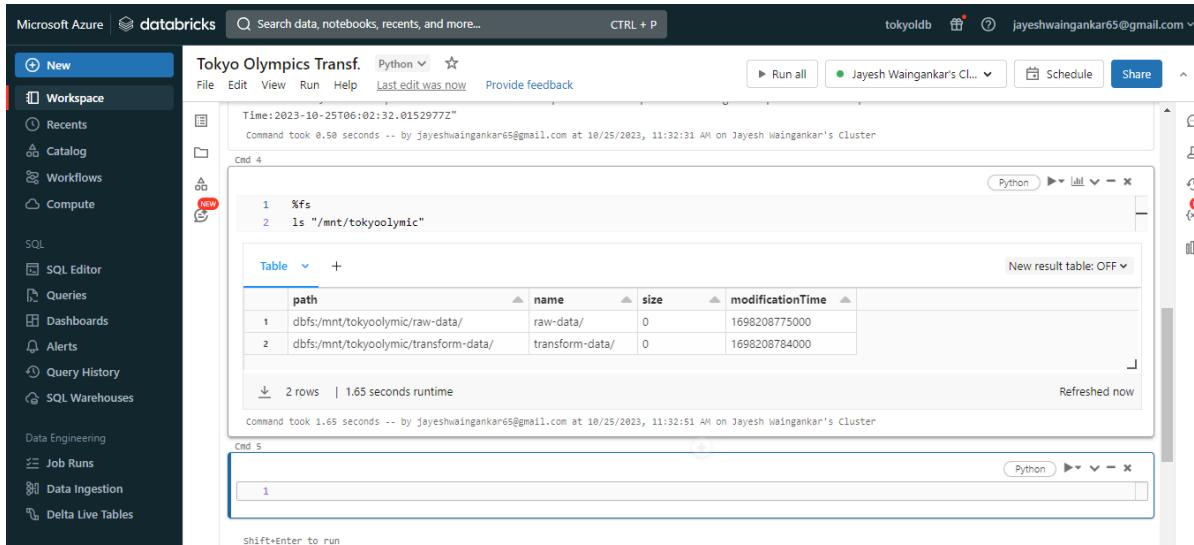
Here select members and select the app we have created

The screenshot shows the 'Add role assignment' dialog in Microsoft Azure. The 'Members' tab is selected. Under 'Selected role', 'Storage Blob Data Contributor' is chosen. Under 'Assign access to', 'User, group, or service principal' is selected. The 'Members' section contains a button '+ Select members'. A modal window titled 'Select members' is open, showing a search bar with 'app' and a list of results: 'Application Insights API' and 'Application Insights Configuration Service'. Below the list is a 'Selected members:' section containing 'app1' with a 'Remove' link. At the bottom of the modal are 'Select' and 'Close' buttons.

Created access control

The screenshot shows the 'Access Control (IAM)' blade for the 'tokyoolymp' container. The 'Role assignments' tab is selected. The top navigation includes 'Check access', 'Role assignments' (selected), 'Roles', 'Deny assignments', and 'Classic administrators'. On the left, a sidebar lists 'Access Control (IAM)', 'Settings', 'Shared access tokens', 'Manage ACL', 'Access policy', 'Properties', and 'Metadata'. The main area displays 'Number of role assignments for this subscription' (2) and '4000'. It includes a search bar 'Search by name or email' and filters for 'Assignment type : All', 'Type : All', 'Role : All', 'Scope : All scopes', and 'Group by : Role'. A table lists '1 items (1 Service Principals)'. The table columns are Name, Type, Role, Scope, and Condition. One item is shown: 'Storage Blob Data Contributor' (Type: App, Role: Storage Blob Data Contributor, Scope: This resource). An 'Add' button is at the bottom right of the table.

Now run the same code no error will come as now we have assigned role



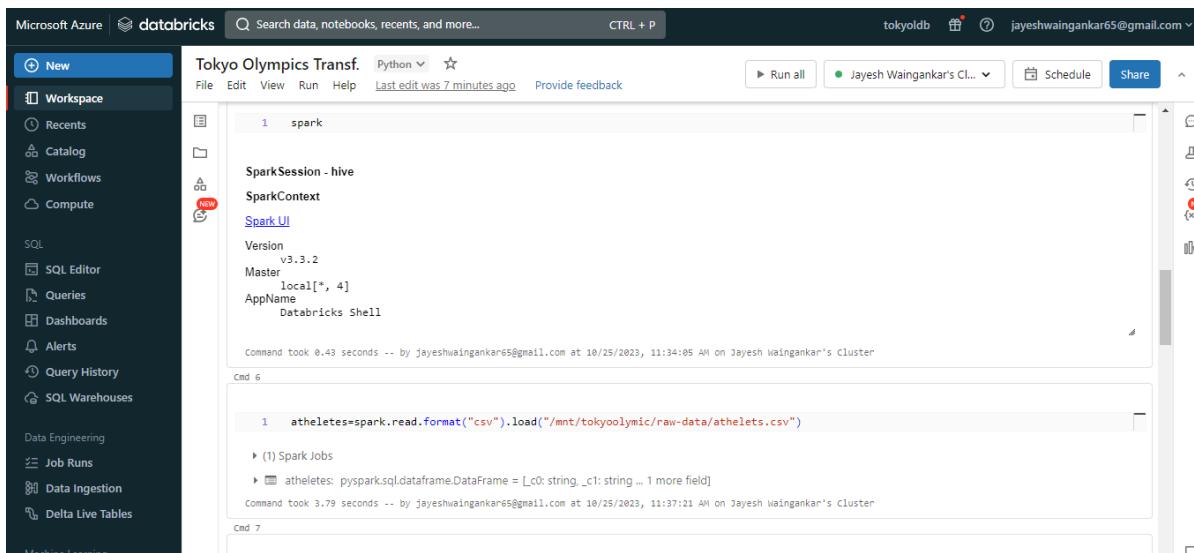
The screenshot shows the Microsoft Azure Databricks workspace interface. On the left, the sidebar includes options like New, Workspace, Recents, Catalog, Workflows, Compute, SQL, and Data Engineering. The main area displays a notebook titled "Tokyo Olympics Transf." in Python. The notebook contains two commands:

```
1 %fs
2 ls "/mnt/tokyoolymic"
```

The output of the first command is a table showing the contents of the directory:

path	name	size	modificationTime
dbfs:/mnt/tokyoolymic/raw-data/	raw-data/	0	1698208775000
dbfs:/mnt/tokyoolymic/transform-data/	transform-data/	0	1698208784000

The second command is empty, indicated by a single digit '1'.



The screenshot shows the Microsoft Azure Databricks workspace interface. On the left, the sidebar includes options like New, Workspace, Recents, Catalog, Workflows, Compute, SQL, and Data Engineering. The main area displays a notebook titled "Tokyo Olympics Transf." in Python. The notebook contains two commands:

```
1 spark
```

The output of the first command is a detailed view of the SparkSession configuration:

```
SparkSession - hive
SparkContext
Spark UI
Version v3.3.2
Master local[*, 4]
AppName Databricks Shell
```

The second command is:

```
1 atheletes=spark.read.format("csv").load("/mnt/tokyoolymic/raw-data/athelets.csv")
```

Tokyo Olympics Transf. Python

```

1 athletes.show()

+-----+-----+-----+
| _c0 | _c1 | _c2 |
+-----+-----+-----+
| PersonName| Country| Discipline|
| AALERUD Katrine| Norway| Cycling Road|
| ABAD Nestor| Spain| Artistic Gymnastics|
| ABAGNALE Giovanni| Italy| Rowing|
| ABALE Alberto| Spain| Basketball|
| ABALDE Tamara| Spain| Basketball|
| ABALO Luc| France| Handball|
| ABAROS Cesar| Chile| Rowing|
| ABASS Abobakr| Sudan| Swimming|
| ABASALI Hamideh| Islamic Republic ...| Karate|
| ABBASOV Islam| Azerbaijan| Wrestling|
| ABBINGH Lois| Netherlands| Handball|
| ABBOTT Emily| Australia| Rhythmic Gymnastics|
| ABBOTT Monica|United States of ...| Baseball/Softball|
| ABDALLA Abubaker ...| Qatar| Athletics|
| ABDALLA Maryam| Egypt| Artistic Swimming|
| ABDALLAH Shahid| Egypt| Artistic Swimming|
| ABDALRASOOL Mohamed| Sudan| Judo|

```

Here we have get the data for all 5 files

```

1 athletes=spark.read.format("csv").option("header","true").load("/mnt/tokyoolympic/raw-data/athletes.csv")
2 coaches=spark.read.format("csv").option("header","true").load("/mnt/tokyoolympic/raw-data/coaches.csv")
3 entrygen=spark.read.format("csv").option("header","true").load("/mnt/tokyoolympic/raw-data/entrygen.csv")
4 medals=spark.read.format("csv").option("header","true").load("/mnt/tokyoolympic/raw-data/medals.csv")
5 teams=spark.read.format("csv").option("header","true").load("/mnt/tokyoolympic/raw-data/teams.csv")

(5) Spark Jobs
athletes: pyspark.sql.dataframe.DataFrame = [PersonName: string, Country: string ... 1 more field]
coaches: pyspark.sql.dataframe.DataFrame = [Name: string, Country: string ... 2 more fields]
entrygen: pyspark.sql.dataframe.DataFrame = [Discipline: string, Female: string ... 2 more fields]
medals: pyspark.sql.dataframe.DataFrame = [Rank: string, Team_Country: string ... 5 more fields]
teams: pyspark.sql.dataframe.DataFrame = [TeamName: string, Discipline: string ... 2 more fields]

Command took 2.08 seconds -- by jayeshwaingankars@gmail.com at 10/25/2023, 11:41:19 AM on Jayesh Waingankar's Cluster

```

Created so now in azure portal go and check in each file all the files are created

Microsoft Azure Search resources, services, and docs (G+)

Home > Storage accounts > tokyooolymp | Containers >

tokyooolymp Container

Search

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: tokyooolymp / transformed-data / athletes

Search blobs by prefix (case-sensitive) Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[...]						...
_committed_719497...	10/25/2023, 11:56:23...	Hot (Inferred)		Block blob	112 B	Available
_started_7194977578...	10/25/2023, 11:56:22...	Hot (Inferred)		Block blob	0 B	Available
_SUCCESS	10/25/2023, 11:56:23...	Hot (Inferred)		Block blob	0 B	Available
part-00000-tid-7194...	10/25/2023, 11:56:23...	Hot (Inferred)		Block blob	397.91 KiB	Available

Home > Storage accounts > tokyooolymp | Containers >

tokyooolymp Container

Search

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: tokyooolymp / transformed-data

Search blobs by prefix (case-sensitive) Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[...]						...
athletes						...
coaches						...
entrygen						...
medals						...
teams						...

Now we will do Analysis of the data so search Synapse Analytics in azure portal and create it

The screenshot shows the Azure Synapse Analytics dashboard. At the top, there are navigation links for Home, Azure Synapse Analytics, and a Default Directory. Below the header is a toolbar with options like Create, Manage view, Refresh, Export to CSV, Open query, and Assign tags. There are also filter buttons for Subscription equals all, Resource group equals all, Location equals all, and a button to Add filter. A search bar labeled 'Filter for any field...' is present. On the left, there are sorting options for Name (up), Type (up), Resource group (up), Location (up), and Subscription (up). On the right, there are grouping and list view options. The main content area displays a large hexagonal icon with a stylized 'S' inside. Below the icon, the text 'No Azure Synapse Analytics to display' is centered. A descriptive paragraph explains that Synapse Analytics is a fully-managed service for building modern data warehouses. It highlights features like SQL, Apache Spark, Orchestration, and Ingestion into a single workspace, which reduces time to build an analytics solution. A prominent blue 'Create Synapse workspace' button is located in the center of the page. At the bottom right, there is a 'Give feedback' link.

Then review and create

The screenshot shows the 'Create Synapse workspace' wizard, Step 1: Workspace details. At the top, there is a breadcrumb trail: Home > Azure Synapse Analytics > Create Synapse workspace. The page title is 'Create Synapse workspace'. On the left, there are sections for 'Resource group *' (set to 'olympic-data') and 'Managed resource group' (with a placeholder 'Enter managed resource group name'). The main area is titled 'Workspace details' with the sub-instruction: '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.' It includes fields for 'Workspace name *' (set to 'tokyooly'), 'Region *' (set to 'Central India'), and 'Select Data Lake Storage Gen2 *'. The 'From subscription' radio button is selected, and the account 'tokyoaac' is listed. The file system 'tokyoolymp' is also listed. A note at the bottom says 'Assign myself the Storage Blob Data Contributor role on the Data Lake'. At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next: Security >'.

Here we have all our details about our project

The screenshot shows the Microsoft Azure portal interface for a resource group named 'olympic-data'. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, Deployments, Security, Deployment stacks, Policies, Properties, and Locks. The main content area displays a table of resources with columns for Name, Type, and Location. The resources listed are:

Name	Type	Location
olymp-toko-df	Data factory (V2)	Central India
tokyoldb	Azure Databricks Service	Central India
tokooc	Storage account	Central India
tokyooly	Synapse workspace	Central India

Go to overview and then open synapse studio

The screenshot shows the Microsoft Azure portal interface for a Synapse workspace named 'tokyooly'. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Microsoft Entra ID, Properties, and Locks. The main content area displays workspace settings and a 'Getting started' section. The workspace settings include:

Setting	Value
Location	Central India
Subscription (move)	Azure for Students
Subscription ID	0e31541e-daff-4840-855e-c1e0edaafbc
Managed virtual network	No
Managed Identity object ID	a6519649-9bd2-4502-bc00-cf49d937f476
Workspace web URL	https://web.azure-synapse.net?workspace=%2bsub%20
Tags (edit)	Add tags

The 'Getting started' section contains two cards: 'Open Synapse Studio' and 'Read documentation'.

Here go to Data and then click on plus sign and there we will create data

The screenshot shows the Microsoft Azure Synapse Analytics Data workspace interface. The left sidebar has a 'Data' section with 'Workspace' selected. The main area is titled 'Select an item' with the sub-instruction 'Use the resource explorer to select or create a new item'. There are icons for databases and tables.

Here we will create table for all 5 files so in folder we will browse the part file here

Container >> transformed data >> part file . Do these for all the 5 files

The screenshot shows the 'Create external table from data lake' dialog box overlaid on the Azure Synapse Analytics Data workspace. The workspace sidebar shows a 'Lake database' named 'tokyodb' containing a 'Tables' folder. The dialog fields include:

- External table name: athletes
- Linked service: tokyooly-WorkspaceDefaultStorage(tokyooac)
- Input file or folder: tokyolymp/transformed-data/athletes/part-00000-tid-7194977578449721388-746b975e...

Buttons at the bottom are 'Continue' and 'Cancel'.

After creating tables for all we will write the SQL query so we will open SQL script and write code. Here we have all the athletes

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, the Data pane displays a 'Lake database' named 'tokyodb' containing a 'Tables' folder with a single table named 'athletes'. The main area shows a 'SQL script 1' tab with the following query:

```
1 SELECT * FROM athletes;
```

The results pane displays the following data:

PersonName	Country	Discipline
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

The status bar at the bottom indicates "00:00:06 Query executed successfully."

These is the athletes in descending order

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, the Data pane displays a 'Lake database' named 'tokyodb' containing a 'Tables' folder with a table named 'athletes' having columns 'PersonName', 'Country', and 'Discipline'. Below the table are other tables: 'coaches', 'entrygen', 'medals', and 'teams'. The main area shows a 'SQL script 1' tab with the following query:

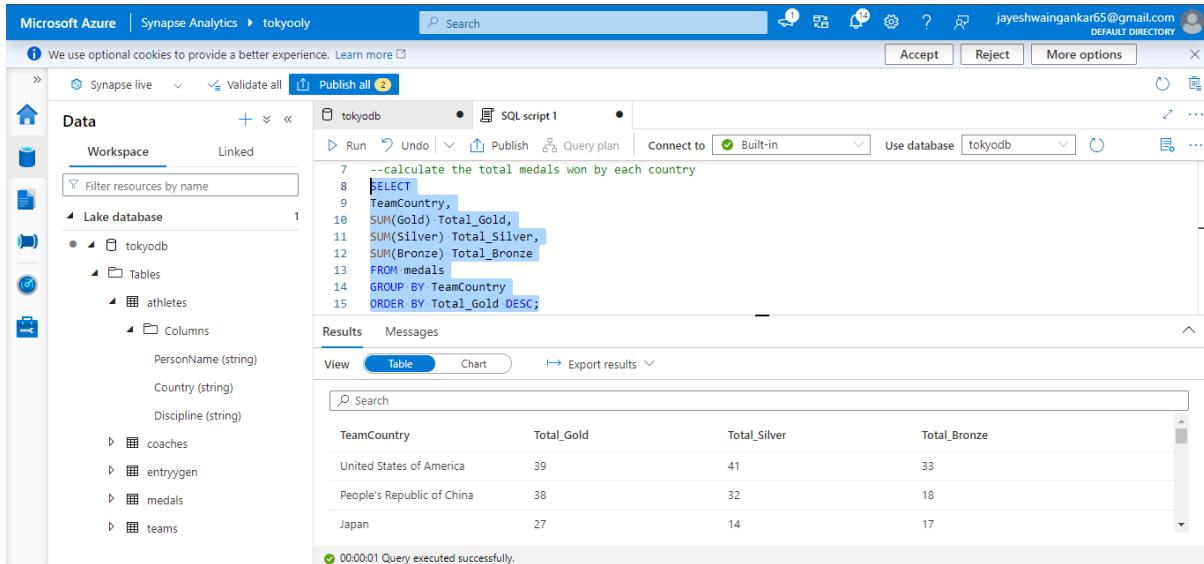
```
1 SELECT COUNTRY,COUNT(*) AS TOTALATHLETES
2 FROM athletes
3 GROUP BY Country
4 ORDER BY TOTALATHLETES DESC;
```

The results pane displays the following data:

COUNTRY	TOTALATHLETES
United States of America	615
Japan	586
Australia	470

The status bar at the bottom indicates "00:00:02 Query executed successfully."

This is the code for medals where we have how many medals are there in total in descending order.



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, the Data workspace sidebar shows a Lake database named 'tokyodb' containing tables like 'athletes', 'coaches', 'entryygen', 'medals', and 'teams'. In the main area, a SQL script titled 'SQL script 1' is displayed:

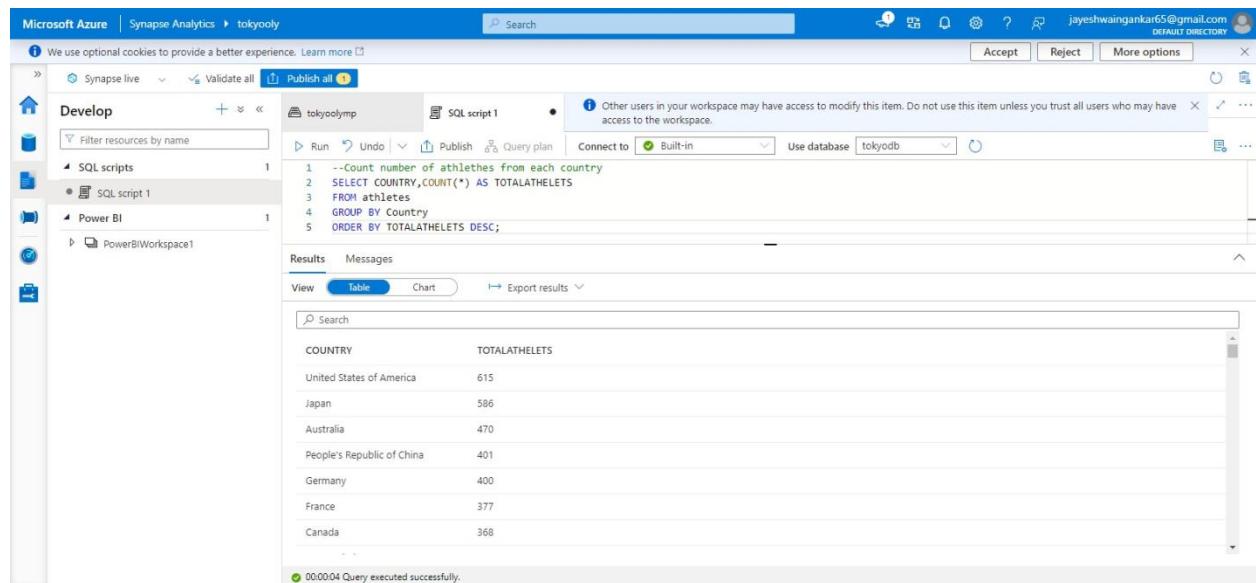
```
--calculate the total medals won by each country
SELECT
TeamCountry,
SUM(Gold) Total_Gold,
SUM(Silver) Total_Silver,
SUM(Bronze) Total_Bronze
FROM medals
GROUP BY TeamCountry
ORDER BY Total_Gold DESC;
```

The results table shows the following data:

TeamCountry	Total_Gold	Total_Silver	Total_Bronze
United States of America	39	41	33
People's Republic of China	38	32	18
Japan	27	14	17

A message at the bottom indicates: 00:00:01 Query executed successfully.

This is count number of athletes from each country



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, the Develop workspace sidebar shows a SQL script named 'SQL script 1' under 'SQL scripts':

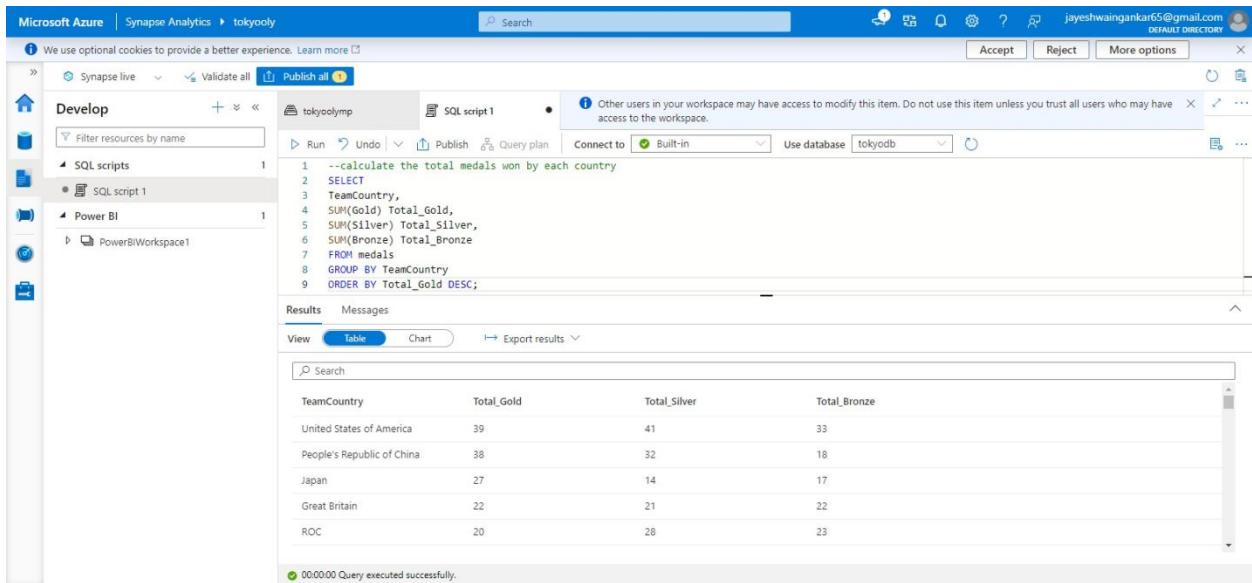
```
--Count number of athletes from each country
SELECT COUNTRY,COUNT(*) AS TOTALATHLETS
FROM athletes
GROUP BY Country
ORDER BY TOTALATHLETS DESC;
```

The results table shows the following data:

COUNTRY	TOTALATHLETS
United States of America	615
Japan	586
Australia	470
People's Republic of China	401
Germany	400
France	377
Canada	368

A message at the bottom indicates: 00:00:04 Query executed successfully.

Total number of medals won by each country



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar shows resources like 'Synapse live', 'Validate all', and 'Power BI'. The main area has a 'Develop' tab selected. A 'SQL script 1' tab is open, containing the following SQL code:

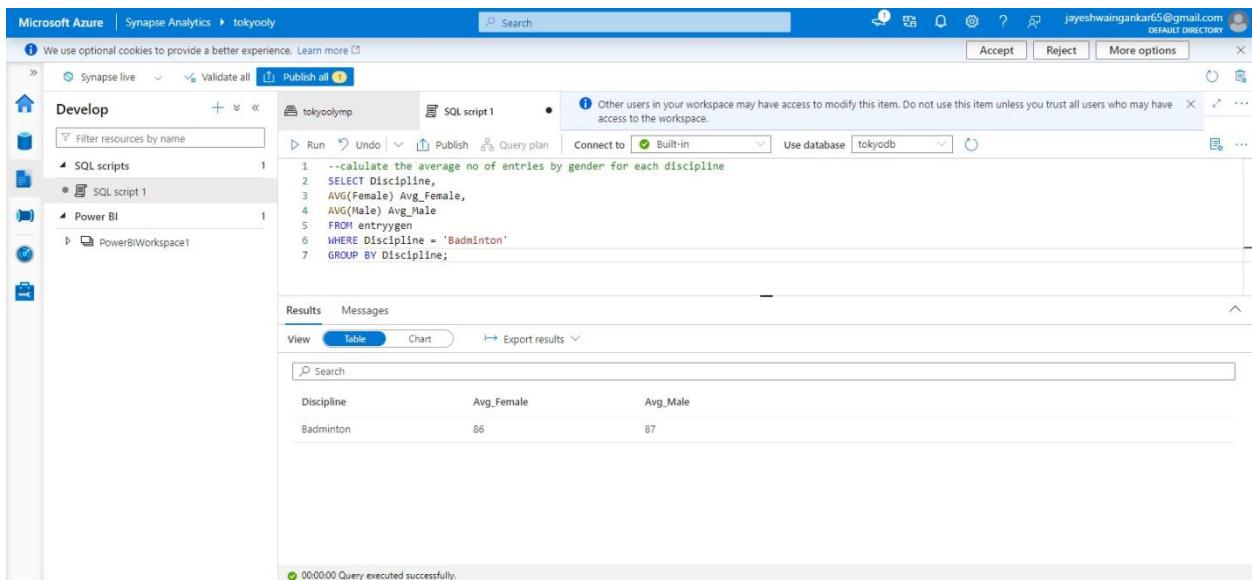
```
--calculate the total medals won by each country
SELECT
    TeamCountry,
    SUM(Gold) Total_Gold,
    SUM(Silver) Total_Silver,
    SUM(Bronze) Total_Bronze
FROM medals
GROUP BY TeamCountry
ORDER BY Total_Gold DESC;
```

The results pane shows a table with the following data:

TeamCountry	Total_Gold	Total_Silver	Total_Bronze
United States of America	39	41	33
People's Republic of China	38	32	18
Japan	27	14	17
Great Britain	22	21	22
ROC	20	28	23

A message at the bottom indicates: "00:00:00 Query executed successfully."

This is the average of entries by gender in badminton. So we can analyze the data using SQL query



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar shows resources like 'Synapse live', 'Validate all', and 'Power BI'. The main area has a 'Develop' tab selected. A 'SQL script 1' tab is open, containing the following SQL code:

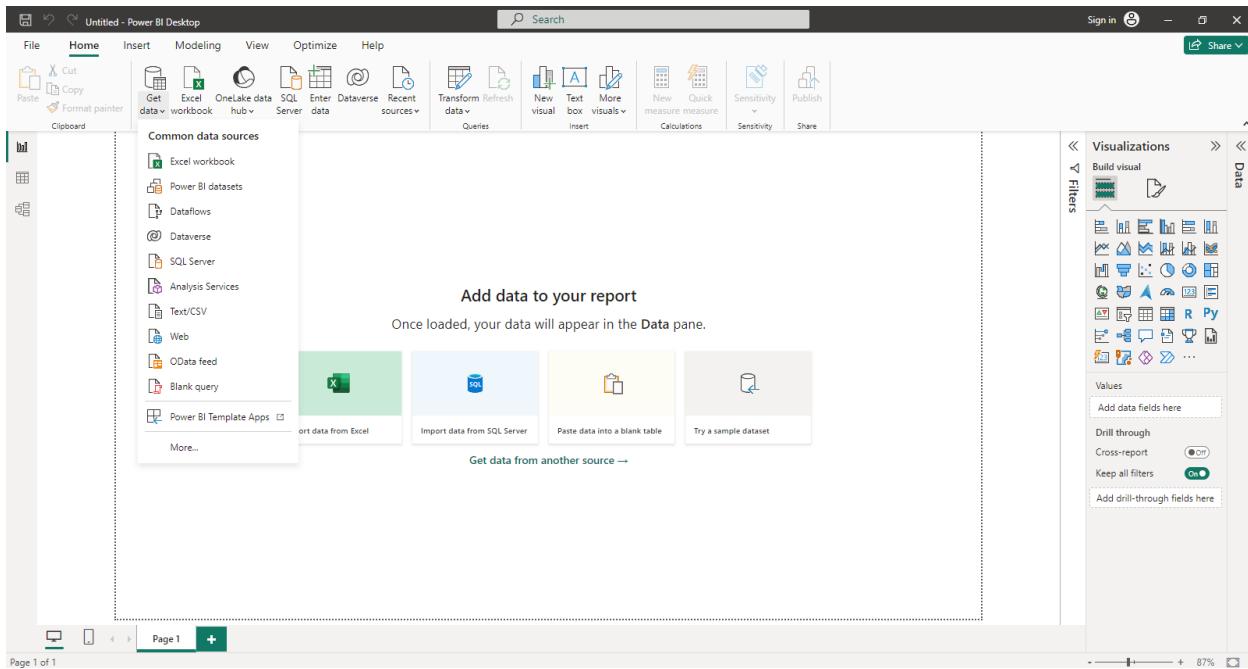
```
--calculate the average no of entries by gender for each discipline
SELECT Discipline,
    AVG(Female) Avg_Female,
    AVG(Male) Avg_Male
FROM entrygen
WHERE Discipline = 'Badminton'
GROUP BY Discipline;
```

The results pane shows a table with the following data:

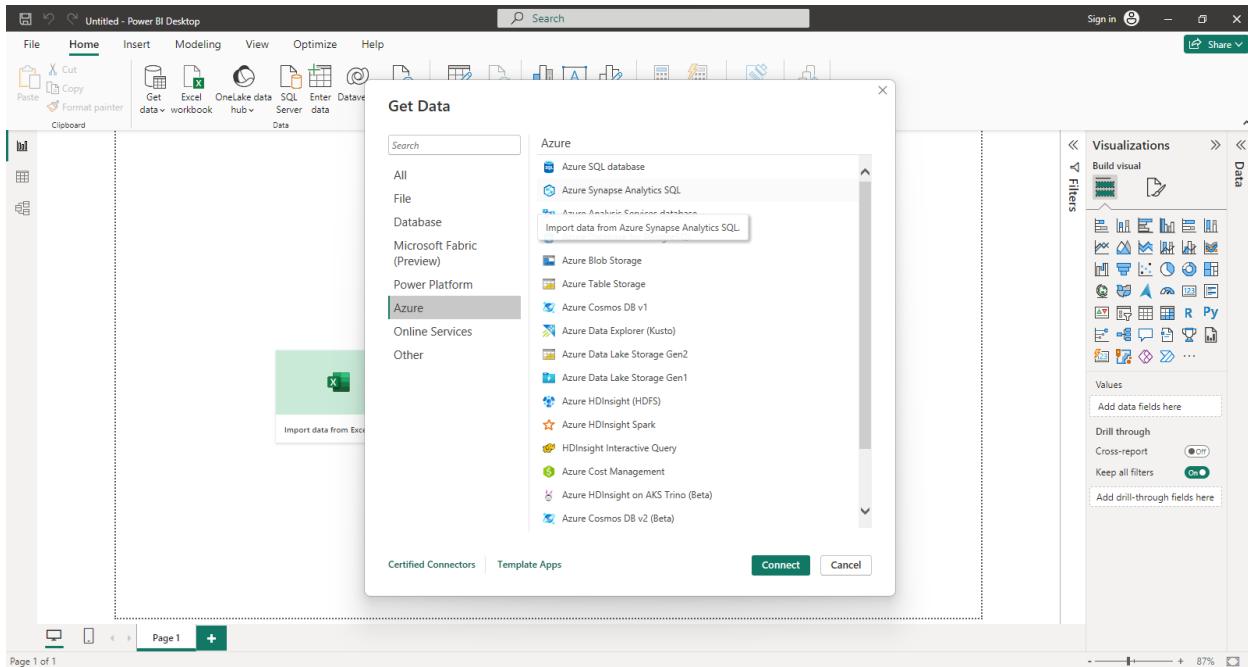
Discipline	Avg_Female	Avg_Male
Badminton	86	87

A message at the bottom indicates: "00:00:00 Query executed successfully."

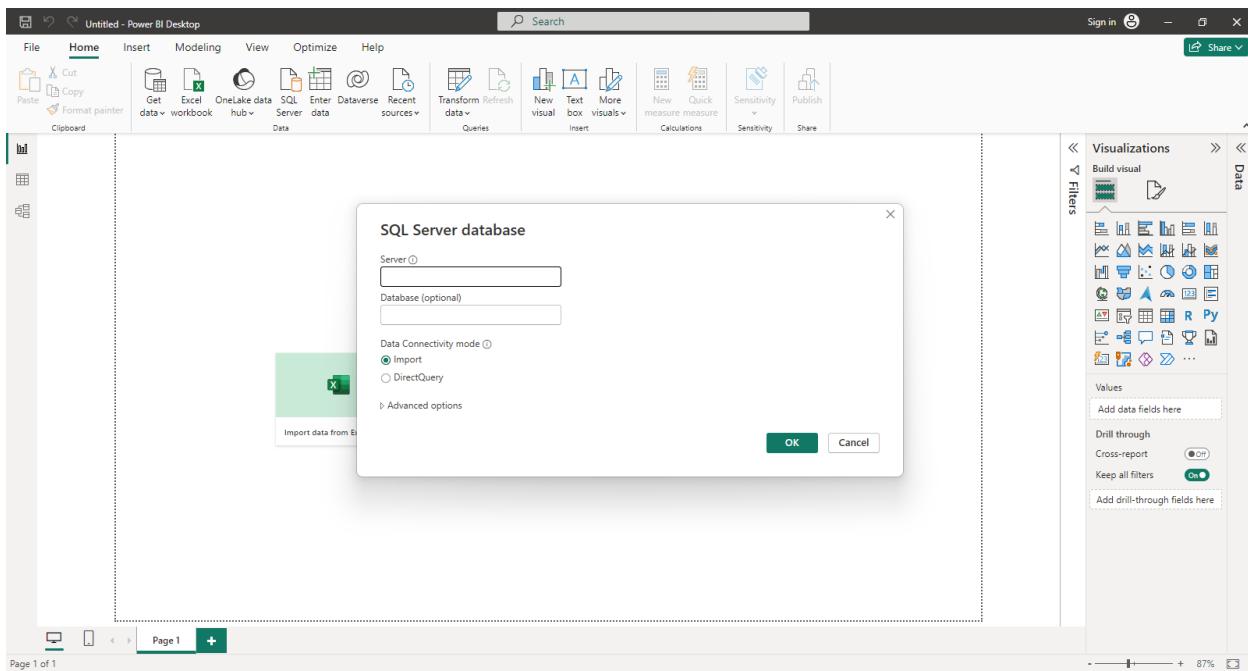
Now we will Do the visualization using Powerbi tool. Select the option get data and click on more



And then select azure synapse analytics SQL and then connect



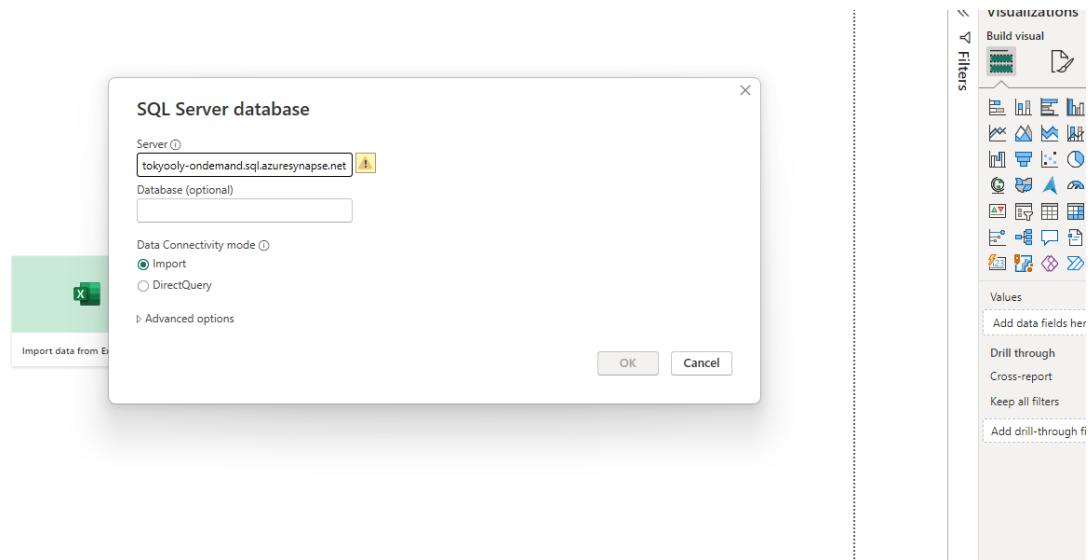
And then enter the credentials



For credentials go to the portal azure synapse analytics and go to properties and copy the serverless SQL endpoint

A screenshot of the Azure portal showing the properties of a workspace named "tokyooly". The left sidebar lists "Default Directory", "Create", "Manage view", and a search bar. The main area shows the workspace name "tokyooly" and its properties. Under "Properties", the "Serverless SQL endpoint" field is highlighted and contains the value "tokyooly-ondemand.sql.azuresynapse.net". Other visible properties include "Primary ADLS Gen2 file system" (tokyolymp), "Dedicated SQL endpoint" (tokyooly.sql.azuresynapse.net), "Development endpoint" (https://tokyooly.dev.azuresynapse.net), "Managed identity object ID" (a6519649-9bd2-4502-bc00-cf49d937f476), "Region" (Central India), "Location ID" (centralindia), and "Resource ID" (/subscriptions/8e31541e-daff-4840-855e-c1e8edaafbc/resourceGroups/olympic-data/providers/M...). The URL https://tokyoolyac.dfs.core.windows.net is also shown in the address bar.

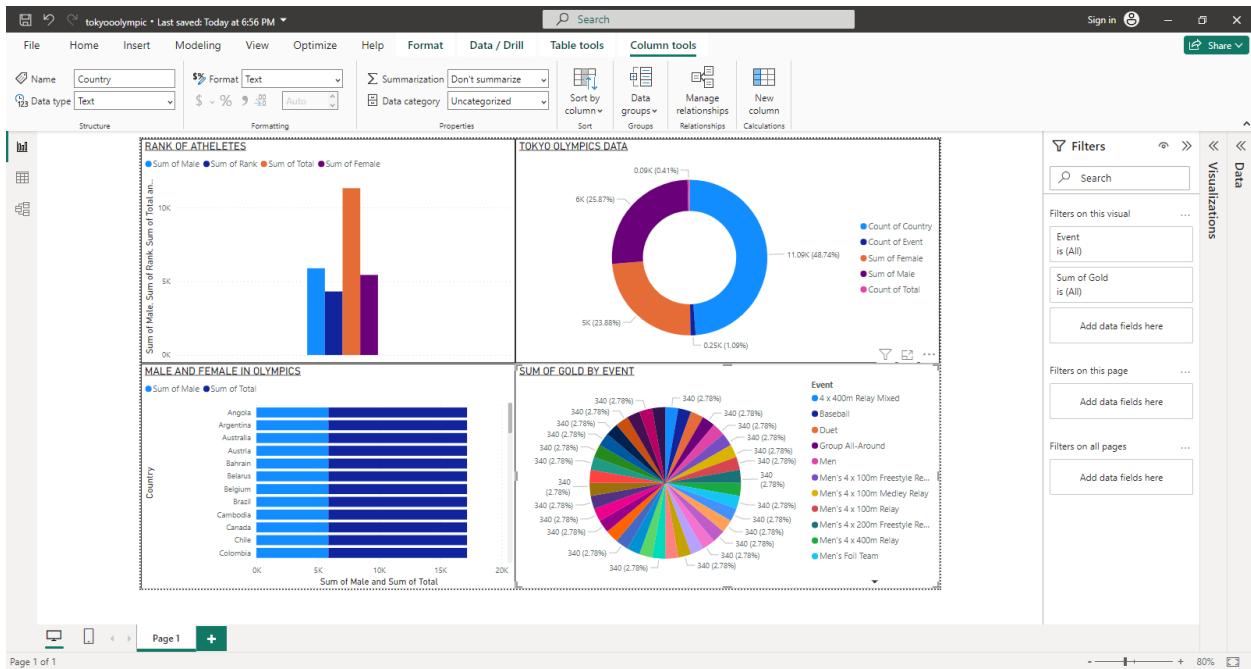
And go to power bi and then enter the copied text and then select import and ok



And then the data will get imported successfully and then create relationship through relationship manager and then make a report through report view

The screenshot shows the Power BI ribbon with the 'Format' tab selected. The 'Name' field is set to 'Country' and the 'Data type' is 'Text'. The 'Formatting' tab is active, displaying controls for currency (\$), percentage (%), and auto formats. The left sidebar shows the navigation pane with 'Report view' selected, followed by 'Table view' and 'Model view'.

And visualize the data as you want



ADVANTAGES AND CONCLUSION

Advantages:-

1. Comprehensive Data Analysis: The project provides a thorough analysis of Tokyo Olympic data, offering valuable insights into athlete performance, team achievements, and medal distributions.
2. Efficient Data Handling: By leveraging Azure services like Data Factory, Databricks, and Synapse Analytics, the project demonstrates an efficient pipeline for data ingestion, transformation, analysis, and visualization.
3. Scalability: Azure services are designed to handle large volumes of data, making this project easily adaptable to future Olympic events or similar large-scale datasets.
4. Learning Opportunities: The project offers valuable lessons in effective collaboration, data quality assurance, and the integration of Azure services, providing a learning platform for future endeavors.
5. Security and Access Control: Properly configuring access control in Azure Data Lake Storage Gen2 ensures secure data handling, protecting sensitive information.
6. Demonstration of Azure Capabilities: The project showcases the capabilities of Azure services, illustrating their role in handling and analyzing large datasets efficiently.

PROBLEMS FACED:-

1. Access Control Configuration: Setting up proper access control in Azure Data Lake Storage Gen2 required careful attention to ensure secure data handling.
2. Data Integration Complexities: Managing different data formats and ensuring seamless integration from a variety of sources presented initial challenges.
3. Data Quality Issues: Dealing with inconsistent data quality in the raw dataset necessitated careful handling to ensure accurate insights.
5. Power BI Customization: Designing a visually appealing and informative dashboard in Power BI required experimentation and learning to leverage its full potential.

Lesson Learned

1. Effective Collaboration: The project highlighted the importance of clear communication and collaborative efforts between team members, ensuring tasks were completed efficiently.
2. Azure Services Integration: Understanding the capabilities of Azure services and their seamless integration greatly enhanced the efficiency of data processing and analysis.
3. Continuous Learning: Regularly exploring new features and best practices in Azure services and Power BI can lead to more sophisticated and efficient solutions.

Future Enhancement :-

1. Integration with External Data Sources:Include data from external sources like social media analytics, weather data, or economic indicators to provide additional context for the analysis.
2. Cost Optimization Strategies:Explore cost-saving measures for Azure services, such as utilizing serverless computing options or optimizing resource allocation based on usage patterns.
3. Feedback Mechanism:Implement a feedback system to gather user feedback on the dashboard's usability, features, and any additional functionalities they would find valuable.

Conclusion

The "Tokyo Olympic Data Analysis using Azure Services" project successfully demonstrated the power of leveraging Azure services for comprehensive data analysis. By employing a well-structured pipeline, the team efficiently ingested, transformed, and analyzed data, providing valuable insights into the Tokyo Olympic dataset. The resulting Power BI dashboard offers a visually intuitive representation of the data, allowing for easy exploration and interpretation. This project serves as a solid foundation for future data analyses and showcases the capabilities of Azure services in handling large-scale datasets.

BIBLIOGRAPHY

We express our sincere gratitude to all those who helped us in gathering the information while preparing this project. To prepare this project we required information regarding how to use services and implement it in proper way.

References:-

1. <https://learn.microsoft.com/en-us/azure/?product=popular>
2. <https://learn.microsoft.com/en-us/power-bi/>
3. <https://learn.microsoft.com/en-us/azure/data-factory/>
4. <https://learn.microsoft.com/en-us/azure/databricks/>
5. <https://learn.microsoft.com/en-us/azure/synapse-analytics/>
6. You tube videos