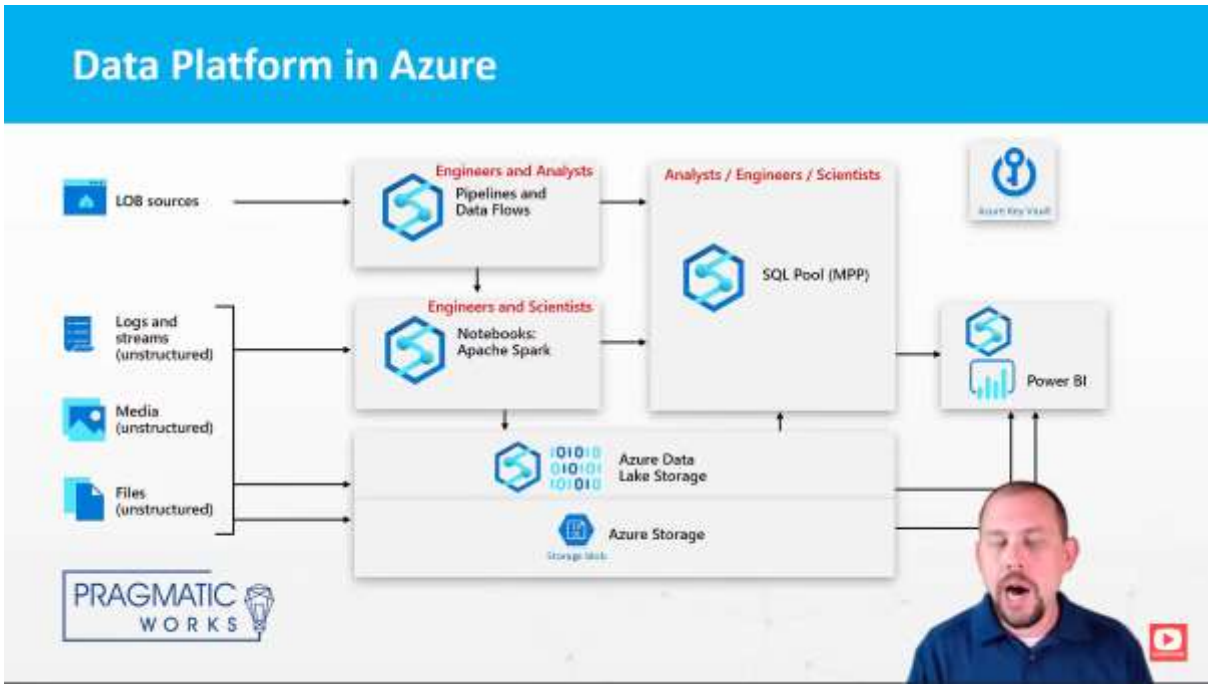
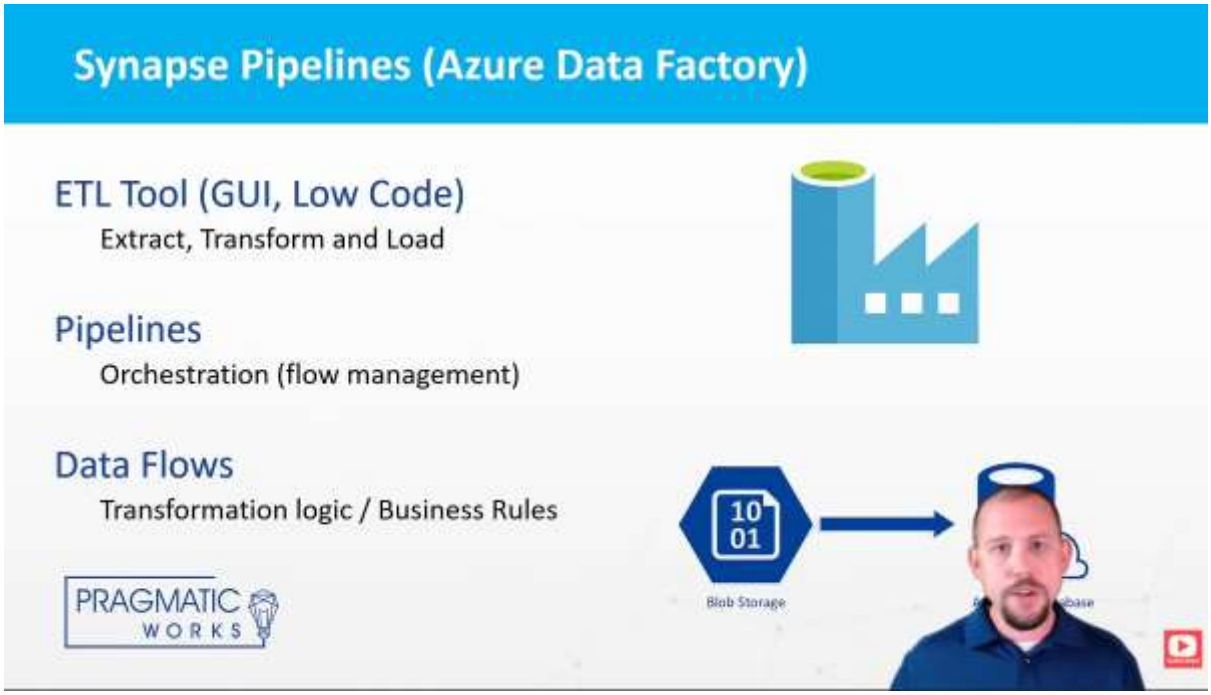


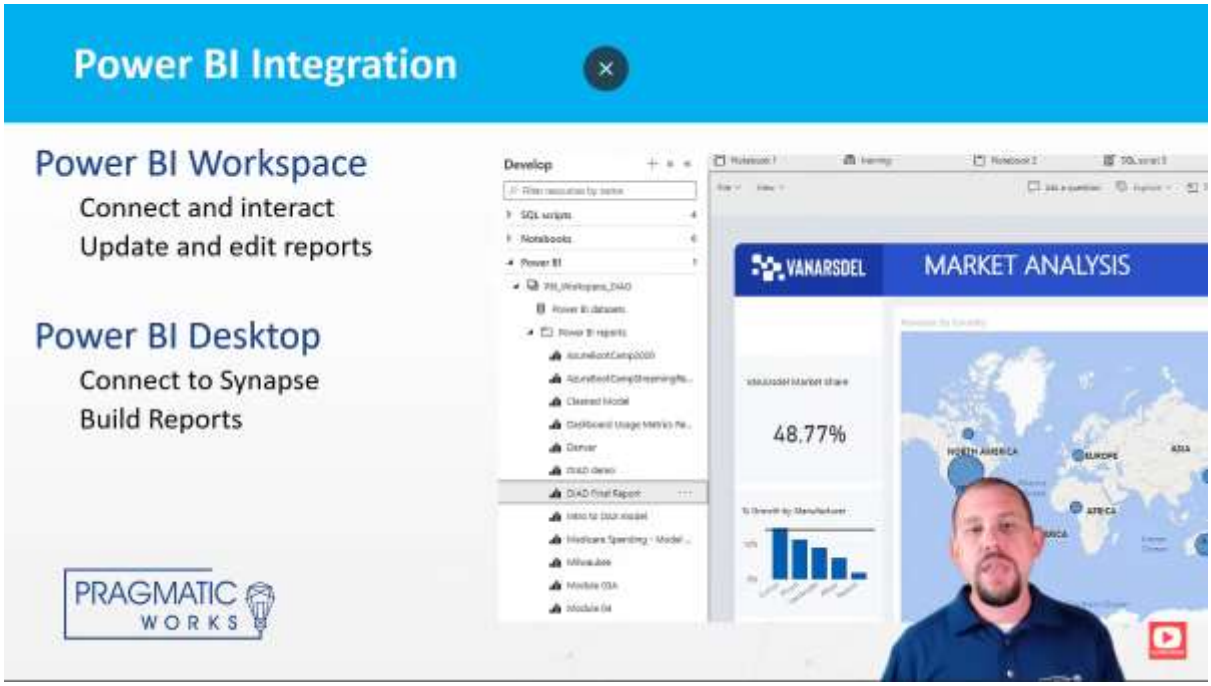
Asa workspace require a datalake configured(should have 1 primary data lake by default)

Azure data lake – general purpose storage account.

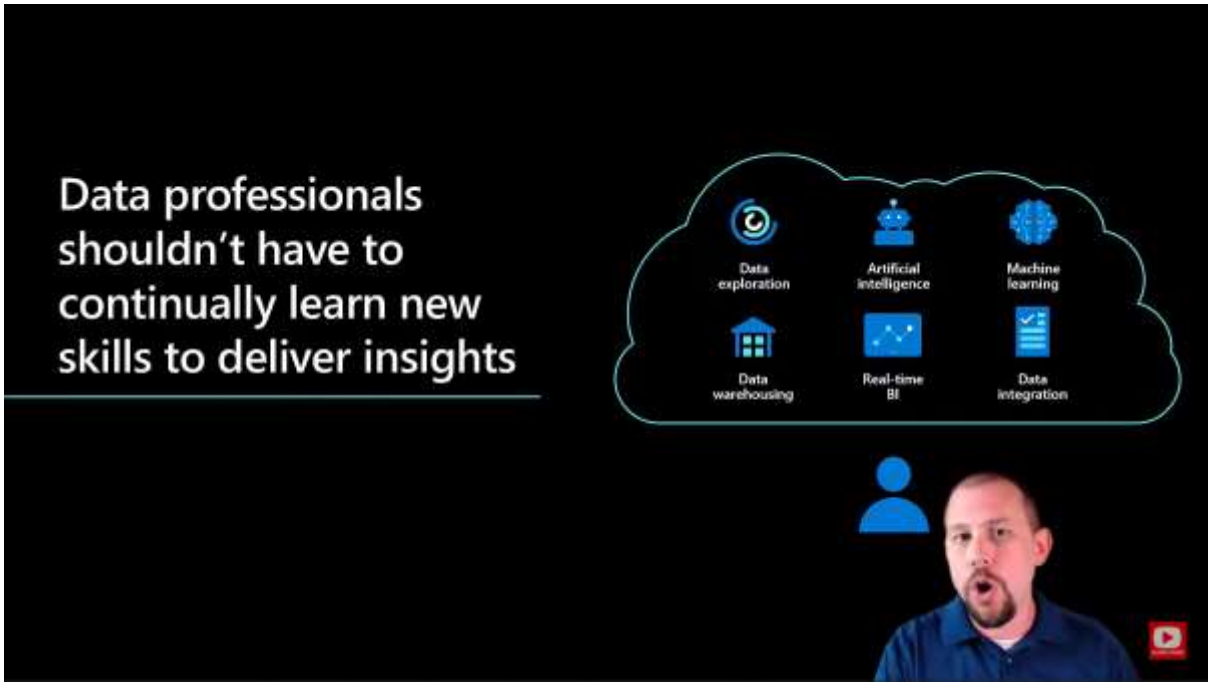
Azure SQL pools(formerly known as SQL Data Warehouse or Parallel Data Warehouse) – helps in doing analysis on big data, relational database(works well on star schema), leverages Masively parallel processing, quickly run complex queries across petabytes of data

Notebooks – Apache spark pools


Synapse Pipelines (Azure Data Factory)
ETL Tool (GUI, Low Code)
Extract, Transform and Load
Pipelines
Orchestration (flow management)
Data Flows
Transformation logic / Business Rules
10
01
Blob Storage
PRAGMATIC
WORKS

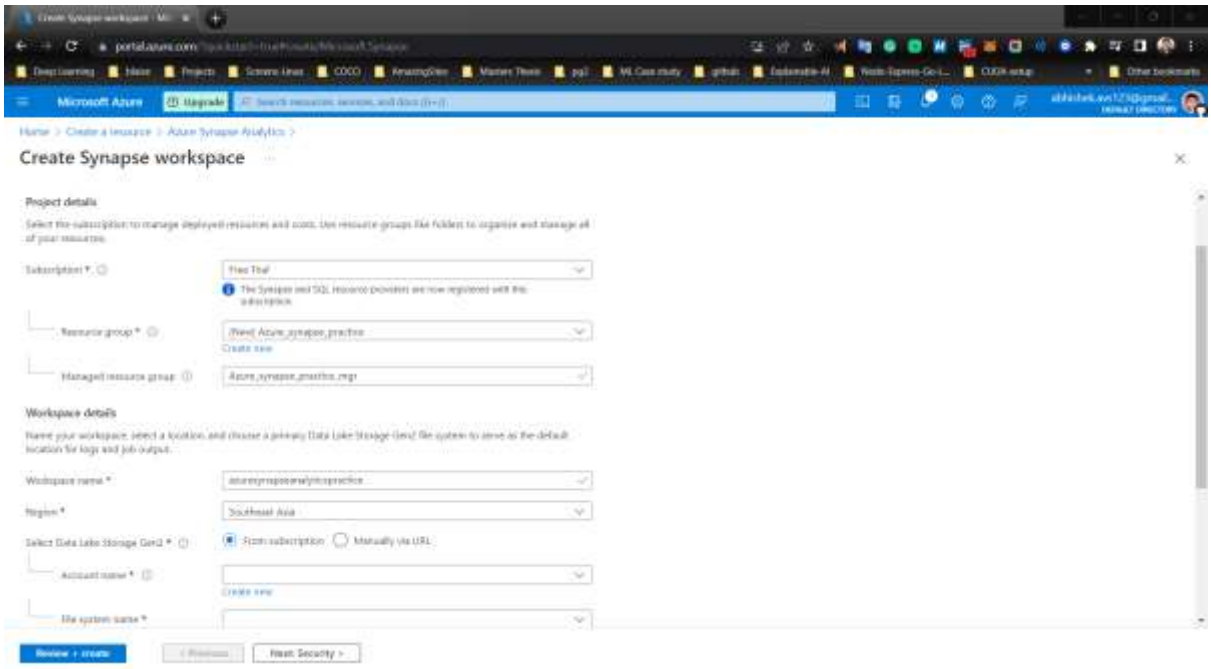

Power BI Integration
Power BI Workspace
Connect and interact
Update and edit reports
Power BI Desktop
Connect to Synapse
Build Reports
Develop
VANARSDEL
MARKET ANALYSIS
48.77%
PRAGMATIC
WORKS

Reason



A **resource group** is a collection of resources that share the same lifecycle, premissions, and policies.

Managed resource group is a container that holds ancillary resources created by Azure Synapse Analytics for your workspace. By default, a managed resource group is created for you when your workspace is created. Optionally, you can specify the name of the resource group that will be created by Azure Synapse Analytics to satisfy your organization's resource group name policies.



- Workspace name must be between 1 and 50 characters long.
- Workspace name must contain only lowercase letters or numbers or hyphens.
- Workspace name must start with a letter or a number.
- Workspace name must end with a letter or a number.
- Workspace name must not contain '-ondemand' word.
- Workspace name must be unique

We need to setup a primary datalake
This account will be the primary storage account for the workspace, holding catalog data and metadata associated with the workspace.

Only Data Lake Storage Gen2 accounts with Hierarchical Name Space (HNS) enabled are listed.

---

first demo i want to do i'm ready we're going to jump right into the first demo here is going to be using serverless pools

serverless on demand - exploration of our data without requiring a dedicated pool

Why serverless pools?

so this dedicated pool that i've provisioned right here for this workshop this is costing me right now 1.50 an hour it's not a lot of money it's the the lowest tier you can absolutely go with but it's going to cost me every hour that it runs i can pause it and i can turn it off but if i do anything that I was using any scripts any you know power bi reports whatever anything that was using that dedicated p1 database is not going to work until it's turned back on so i'm saving money but it's highly inconvenient so instead the way we can kind of work around that problem is we can use serverless

(5$/TB od data processed)

serverlesspools is where we are essentially going to be saying look i want to query data that's in my data

lake using sql and i want to be able to scale that out i want to get great performance even if it's big data but i don't want

to have to go spin up a dedicated pool and so companies shouldn't have to spin

up and leave a dedicated pool running all the time and so if the pool's paused and it's not running and i want to run a

query then i have to go and turn the pool on and that pool is going to be running and it's going to charge me x number of dollars per hour for me just

to go and run a couple of ad hoc queries right that's not very flexible not very helpful and so serverless

sql on demand gives us the capability of kind of just in time reporting it gives

us the capability of running a lot of ad hocs and interacting with our data and doing some you know just it kind of

exploration of our data without requiring a dedicated pool this is game changing in a lot of ways because before

this you would have had to you know had your dedicated pool running which does can cost quite a bit of money per hour

that it runs right so this dedicated pool that i've provisioned right here for this workshop this is costing me

right now 1.50 an hour it's not a lot of money it's the the lowest tier you can absolutely go with

but it's going to cost me every hour that it runs i can pause it and i can turn it off but if i do anything that i

was using any scripts any you know power bi reports whatever anything that was using that dedicated p1 database is not

going to work until it's turned back on so i'm saving money but it's highly

inconvenient so instead the way we can kind of work around that problem is we can use serverless let me show you what

this is every asa environment by default has serverless pools available to you it's

there you can use it if you want don't use it if you don't want serverless pools cost you five dollars

per terabyte of data processed that's can be pretty inexpensive and i'm going to show you that here in a minute

when we go back and look at monitoring okay so five dollars per data per terabyte of data that is processed

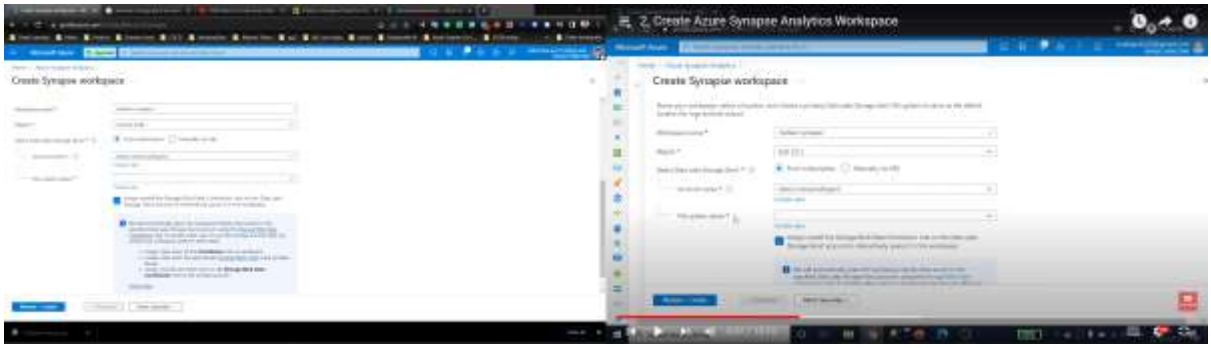
so let's do this let's go out to our data lake and just query some data and take a look at it so i'm going to go

back over to the data hub under the data hub i have some data that i've already uploaded to my data lake right so i'm

What is Azure Synapse Analytics ?

- Azure Synapse is a limitless analytics service that brings together enterprise data warehousing and Big Data analytics. It gives you the freedom to query data on your terms, using either serverless or dedicated resources—at scale.
- Azure Synapse Analytics bring best of below components together as single service
  - SQL technologies used in data warehousing(Synapse SQL)
  - Spark technologies used in Bigdata(Apache Spark)
  - Pipelines for data integration and ETL/ELT(ADF)
- It has deep integration with Azure services such as Power BI, Cosmos DB, Azure ML etc..

Azure Synapse Analytics



Filesystem name – inside datalake storage gen 2 we have containers that has folders and files. We give a name to the container

Synapse Workspace

- Synapse Workspace is a collaboration place for doing cloud-based enterprise analytics in Azure
- A Workspace will be associated with ADLS Gen2 and File System
- A workspace allows you to perform analytics with SQL and Apache spark
- Resources available for SQL and Spark analytics are organized into SQL and Spark pools.

Linked Services

- Linked service, essentially connection strings that define the connection information needed for workspace to connect to external resources

Synapse SQL

- Synapse SQL is the ability to do T-SQL based analytics in Synapse workspace
- Synapse SQL has two consumption models: dedicated and serverless.
- Synapse SQL Pools actually helps you to run SQL scripts

3. Basic Concepts in Azure Synapse Analytics

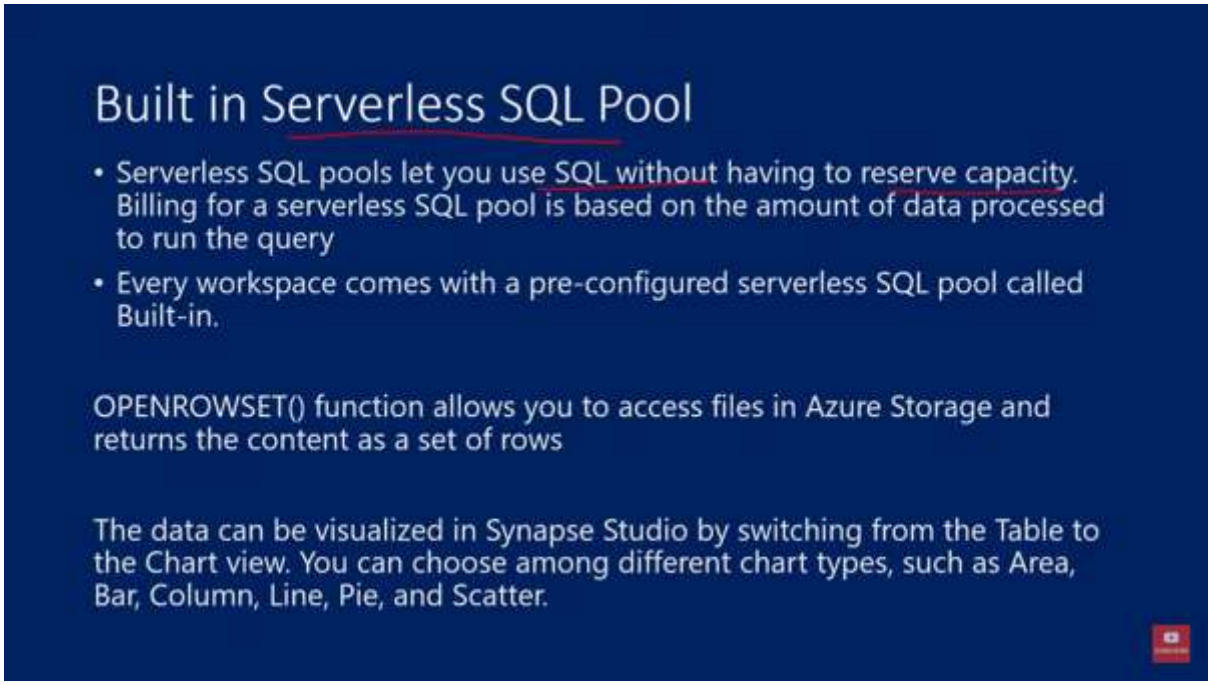
Apache Spark for Synapse

- This gives ability to do Spark based analytics in Synapse Workspace
- You can create Apache spark pools in Workspace. When you start using that Spark Pool, the workspaces creates spark session to handle the resources associated with that session
- There are two ways to use spark in Synapse
  - Spark Notebooks – For doing data science and data engineering using Scala, PySpark, C# and Spark SQL
  - Spark Job definitions – For running batch Spark jobs using jar files

Pipelines

- Pipelines are how Azure Synapse provides Data Integration - allowing you to move data between services and orchestrate activities.
  - Pipeline are logical grouping of activities that perform a task together.
  - Activities defines actions within a Pipeline to perform on data such as copying data, running a Notebook or a SQL script.
  - Data flows are a specific kind of activity that provide a no-code experience for doing data transformation that uses Synapse Spark under-the-covers.
  - Trigger - Executes a pipeline. It can be run manually or automatically (schedule, tumbling window or event-based)
  - Integration dataset - Named view of data that simply points or references the data to be used in an activity as input and output. It belongs to a Linked Service.

SQL SERVERLESS POOL



Synapse workspace>pverview tab> scroll down> open in synapse studio> web.azuresynapse.net will open> manage menu left> sql pool> built-in => comes with pre configured sql pool

In data tab upload some data in linked tab inside the gen2 a container is there, make a data folder and add a data

Then to add sql script, go to develop and add sql script

To add autogenerated one => go to data> new sql script> se;lect top 100 rows

Dedicated SQL pool

Manage>sql pool> +new>choose performance and create

Synapse works superfast

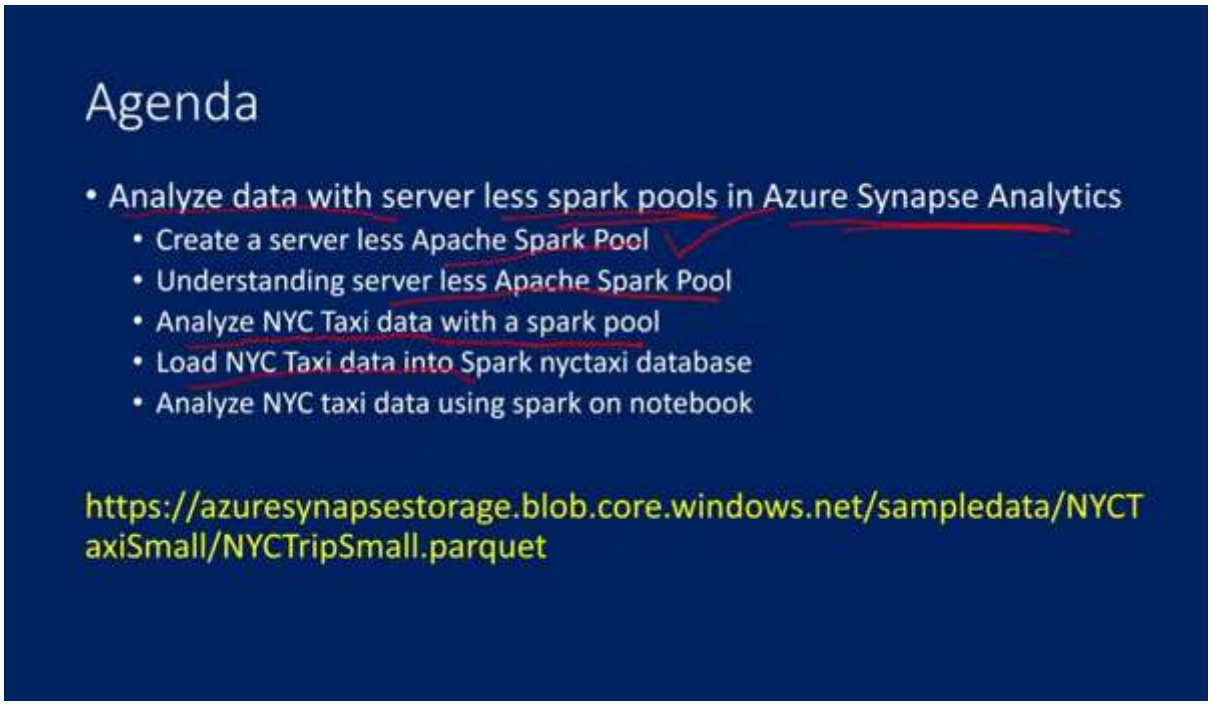
```
IF NOT EXISTS (SELECT * FROM sys.objects O JOIN sys.schemas S ON O.schema_id = S.schema_id WHERE O.NAME = 'NYCTaxiTri
pSmall' AND O.TYPE = 'U' AND S.NAME = 'dbo')
CREATE TABLE dbo.NYCTaxiTripSmall
    (
     [DateID] int,
     [MedallionID] int,
     [HackneyLicenseID] int,
     [PickupTimeID] int,
     [DropoffTimeID] int,
     [PickupGeographyID] int,
     [DropoffGeographyID] int,
     [PickupLatitude] float,
     [PickupLongitude] float,
     [PickupLatLong] nvarchar(4000),
     [DropoffLatitude] float,
     [DropoffLongitude] float,
     [DropoffLatLong] nvarchar(4000),
     [PassengerCount] int,
     [TripDurationSeconds] int,
     [TripDistanceMiles] float,
     [PaymentType] nvarchar(4000),
     [FareAmount] numeric(19,4),
     [SurchargeAmount] numeric(19,4),
     [TaxAmount] numeric(19,4),
     [TipAmount] numeric(19,4),
     [TollsAmount] numeric(19,4),
     [TotalAmount] numeric(19,4)
    )
WITH
    (
    DISTRIBUTION = ROUND_ROBIN,
     CLUSTERED COLUMNSTORE INDEX
     -- HEAP
    )
GO

COPY INTO dbo.NYCTaxiTripSmall
(DateID 1, MedallionID 2, HackneyLicenseID 3, PickupTimeID 4, DropoffTimeID 5,
PickupGeographyID 6, DropoffGeographyID 7, PickupLatitude 8, PickupLongitude 9,
PickupLatLong 10, DropoffLatitude 11, DropoffLongitude 12, DropoffLatLong 13,
PassengerCount 14, TripDurationSeconds 15, TripDistanceMiles 16, PaymentType 17,
FareAmount 18, SurchargeAmount 19, TaxAmount 20, TipAmount 21, TollsAmount 22,
TotalAmount 23)
FROM 'https://maheeradlsgen2.dfs.core.windows.net/synapsedemo/data/NYCTripSmall.parquet'
WITH
```

```
(
    FILE_TYPE = 'PARQUET'
    ,MAXERRORS = 0
    ,IDENTITY_INSERT = 'OFF'
)

SELECT COUNT(*) from dbo.NYCTaxiTripSmall
```

200000 rows in 0.07 sec

We can see the created table in data tab in workspace in tables sub menu in sqlpool and we can use autogenerated script to see the new created table, we can also change the script

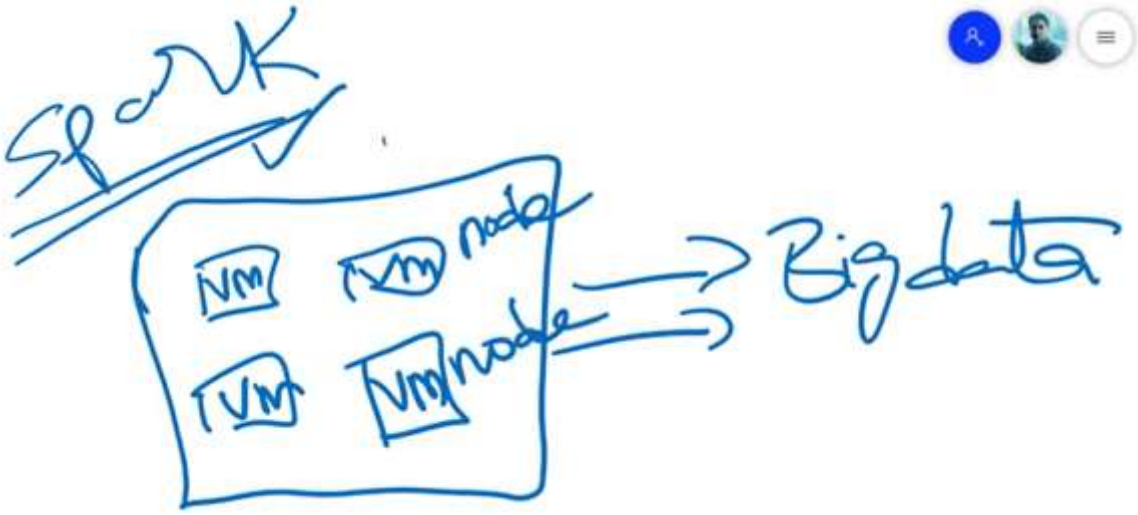
```
SELECT PassengerCount,

    SUM(TripDistanceMiles) as SumTripDistance,

    AVG(TripDistanceMiles) as AvgTripDistance

FROM  dbo.NYCTaxiTripSmall

WHERE TripDistanceMiles > 0 AND PassengerCount > 0

GROUP BY PassengerCount

ORDER BY PassengerCount;
```



Create server less Spark Pool

- Apache Spark is a parallel processing framework that supports in-memory processing to boost the performance of big-data analytic applications.
- We can create Server less Spark Pool under Management hub menu in Synapse Studio

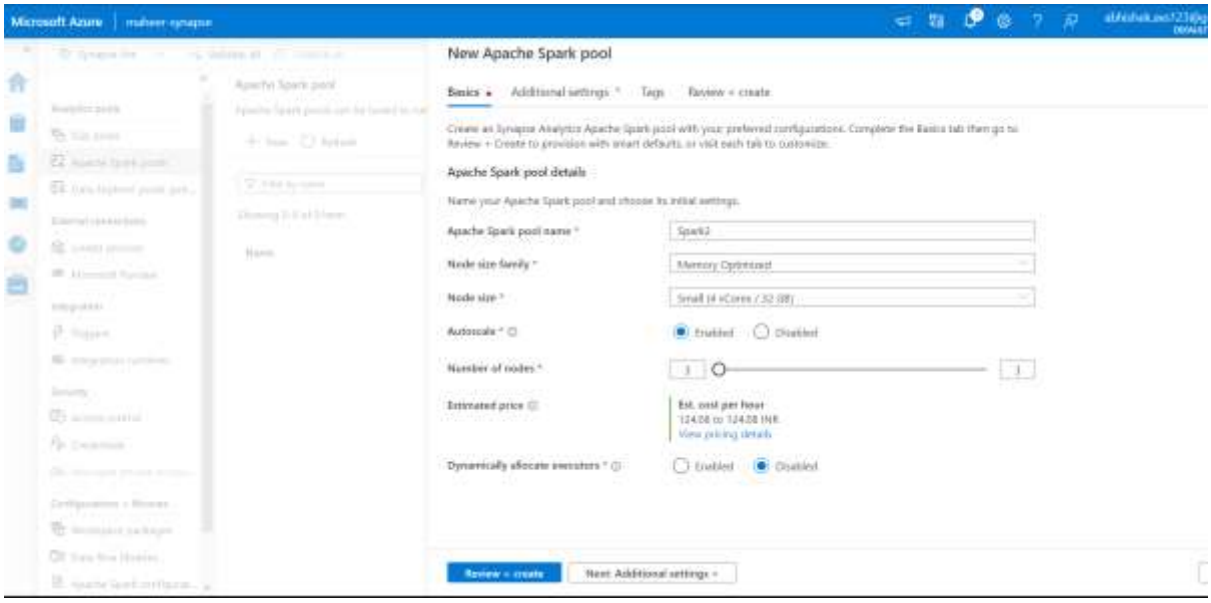
Spark – used for big data processing



Sparks has clusters, which is group of VMs called nodes. These clusters have capability to understand big data and execute. We write spark code in python or scala. These clusters can understand the spark logic and execute that on the big data to generate meaningful data etc.

In Azure data brick, we create spark cluster and that cluster will be running the notebook **but** in synapse we create spark pool that will execute the notebook. The notebook consists of logic.

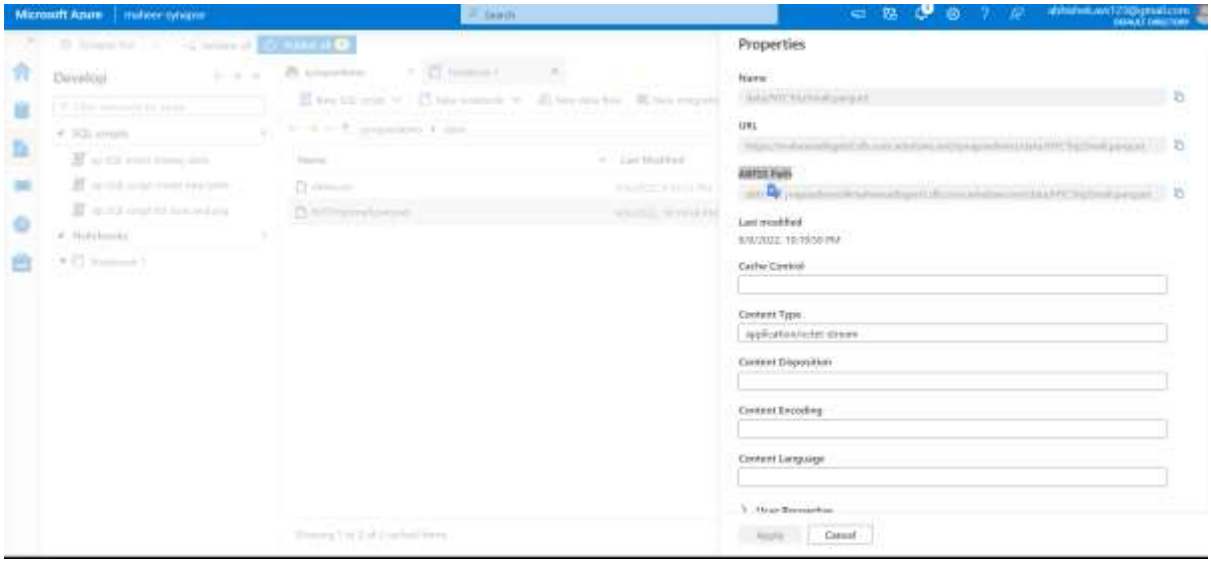
In the spark pool we need to define how many nodes we want and size of each node



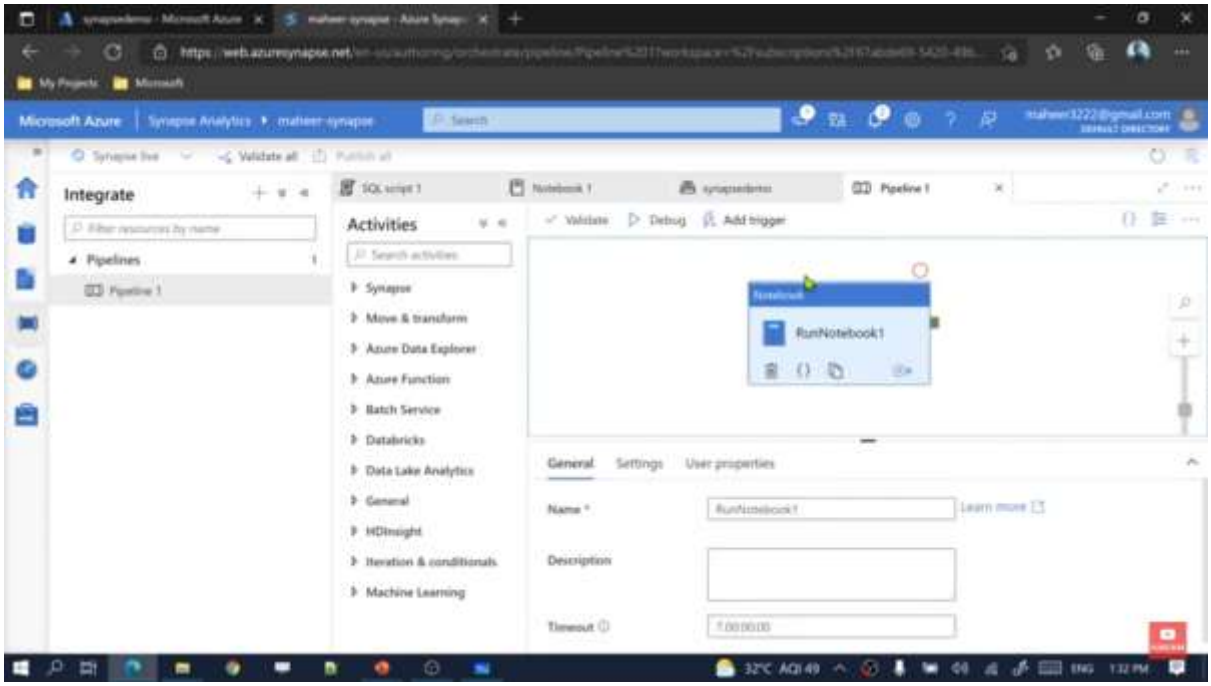
Spark pool decides how many nodes to use based on our data, that is why it asks for min and max number of nodes range.

Serverless spark pools are similar to serverless SQL pool. Whenever we use nodes and we run the logic (notebook), the amount of resources used are only charged for.

We use the path of ABFSS path – azure blob file system

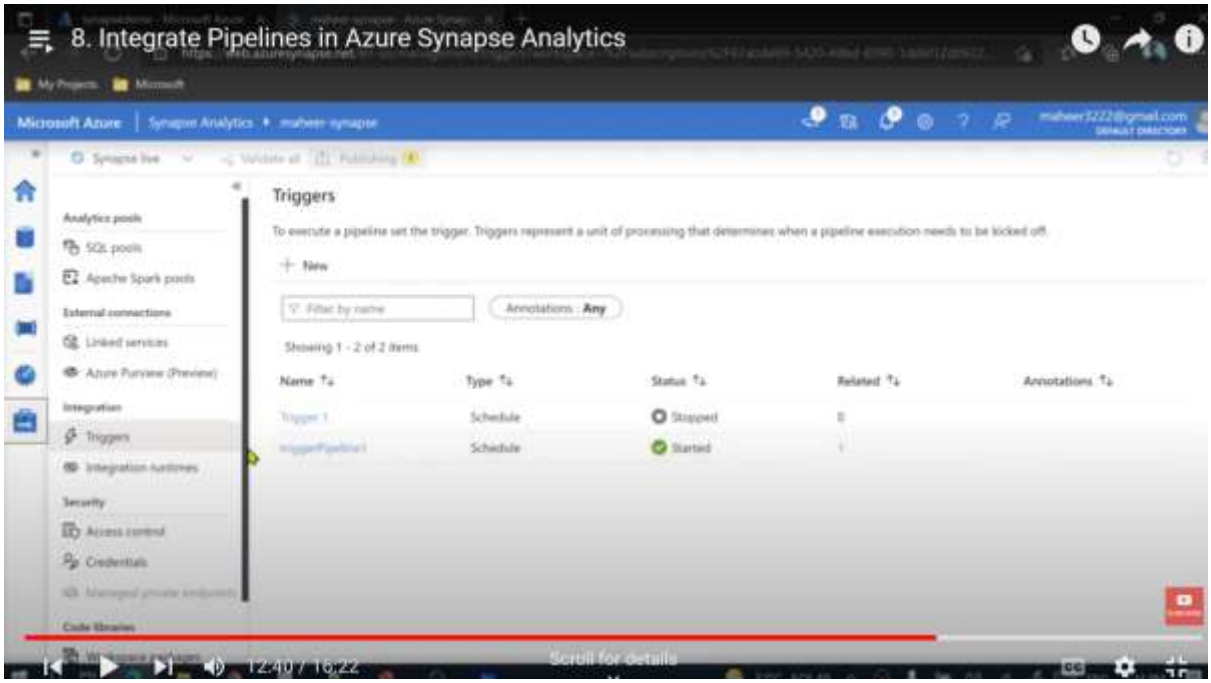


7. Analyze data in Storage Account in Azure Synapse Analytics

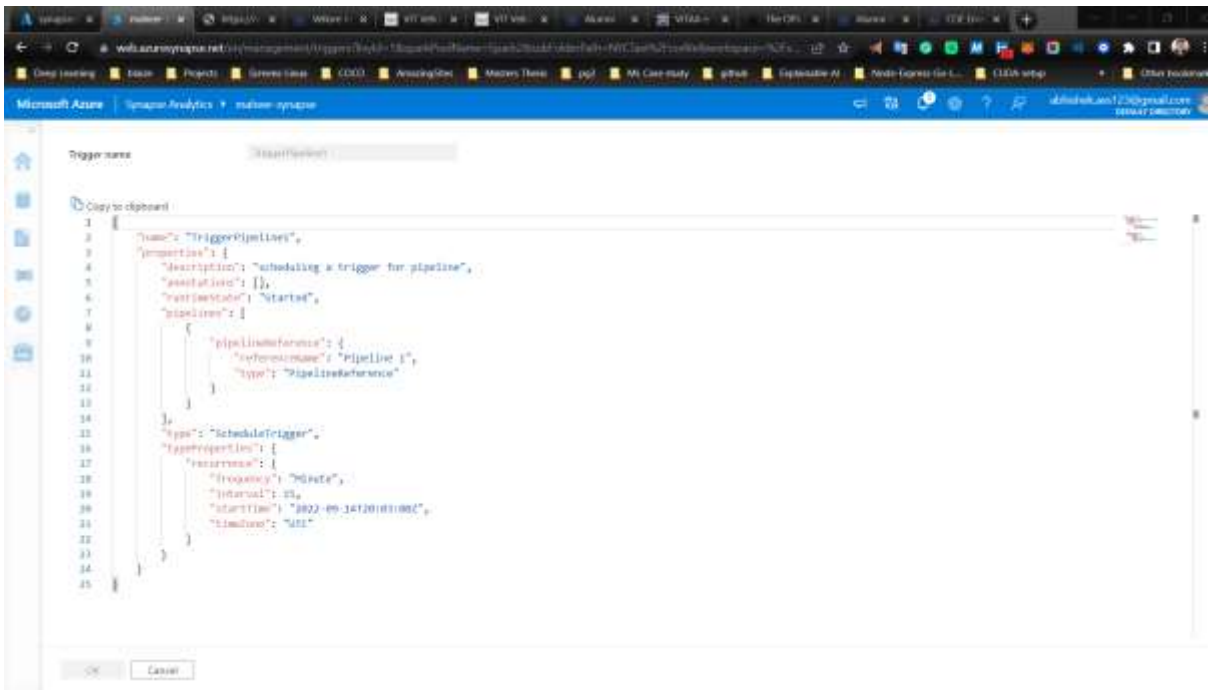


integrate the notebook and run using the pipeline

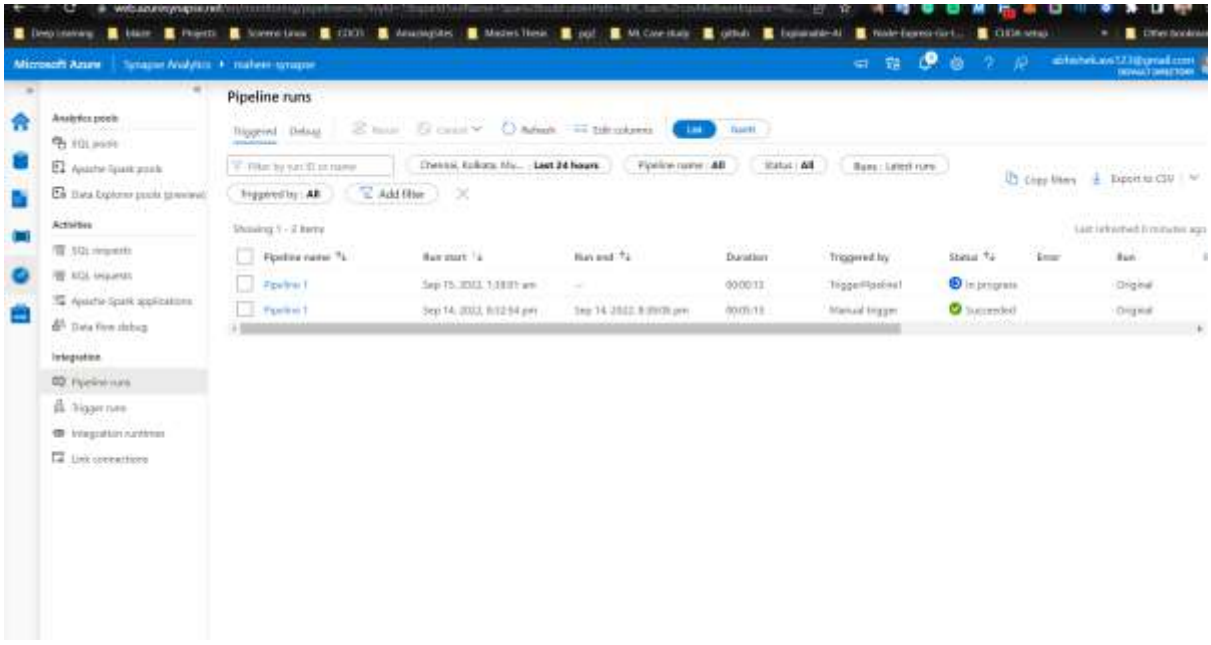
To see the triggers u need to go to manage



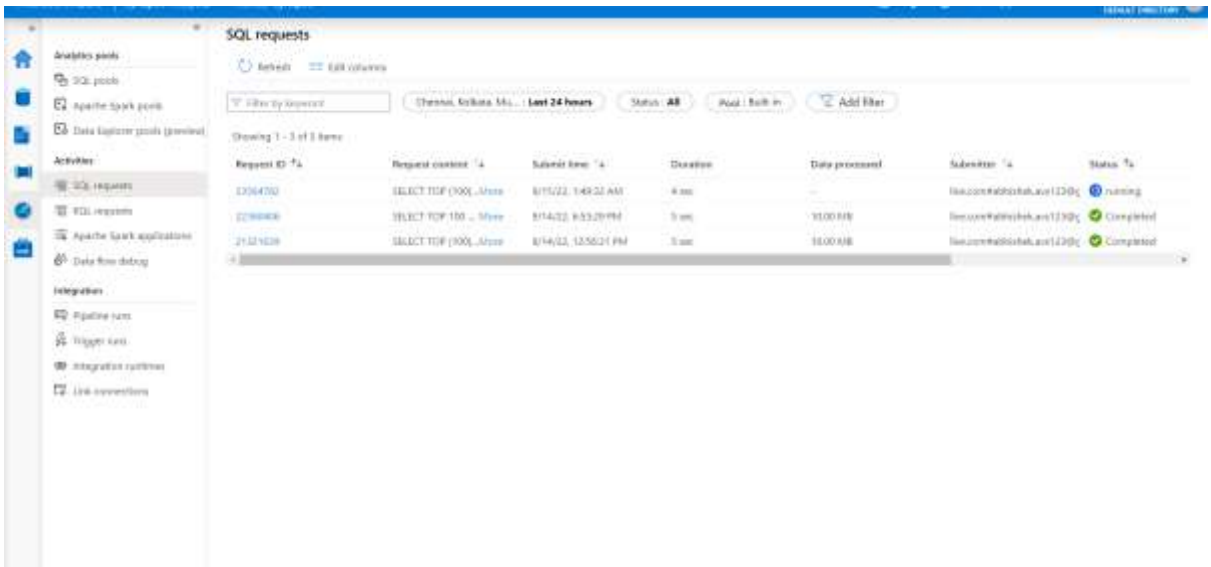
Scheduled trigger, go to manage>triggers> there is a {} symbol on the triogger, if u click it, u can see the json with the trigger schedule etc details



One is the manual trigger, another is a scheduled trigger



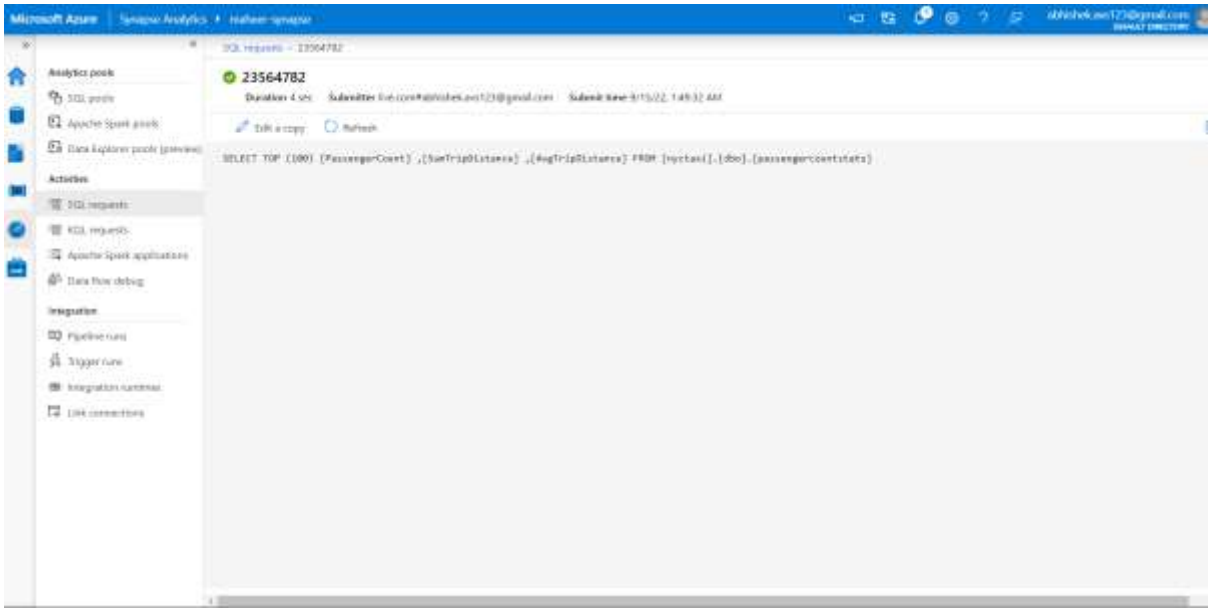
9. Monitor your Azure Synapse Analytics Workspace



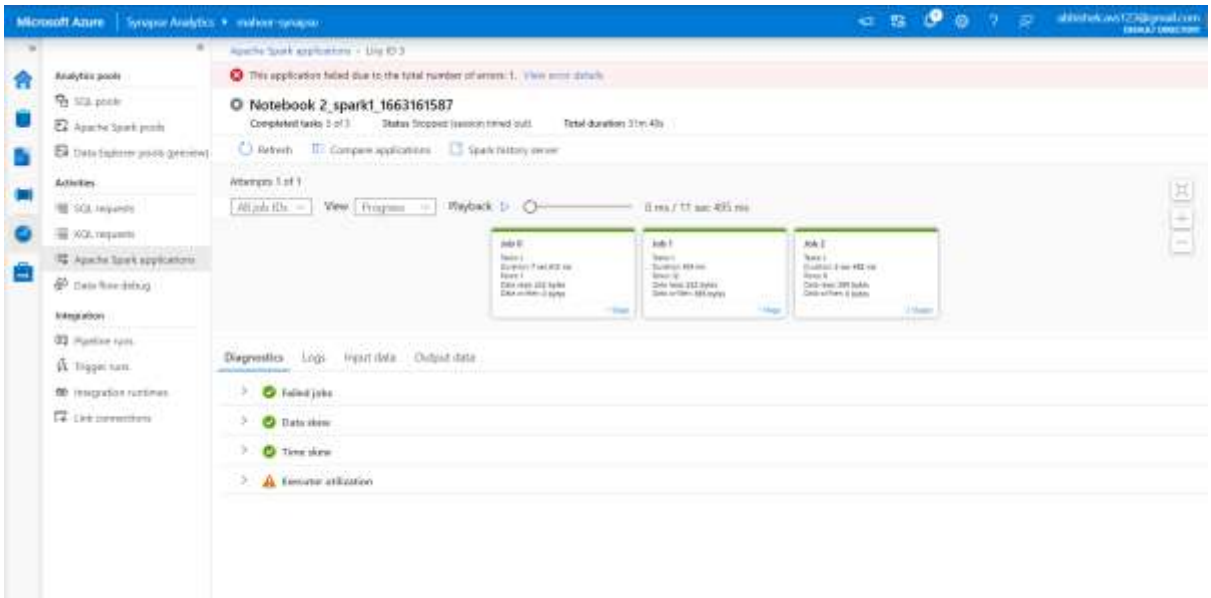
,

In monitor tab u can monitor all the sql requests, apache spark application aetc in activities and in integration u can see the pipline runs and trigger runs

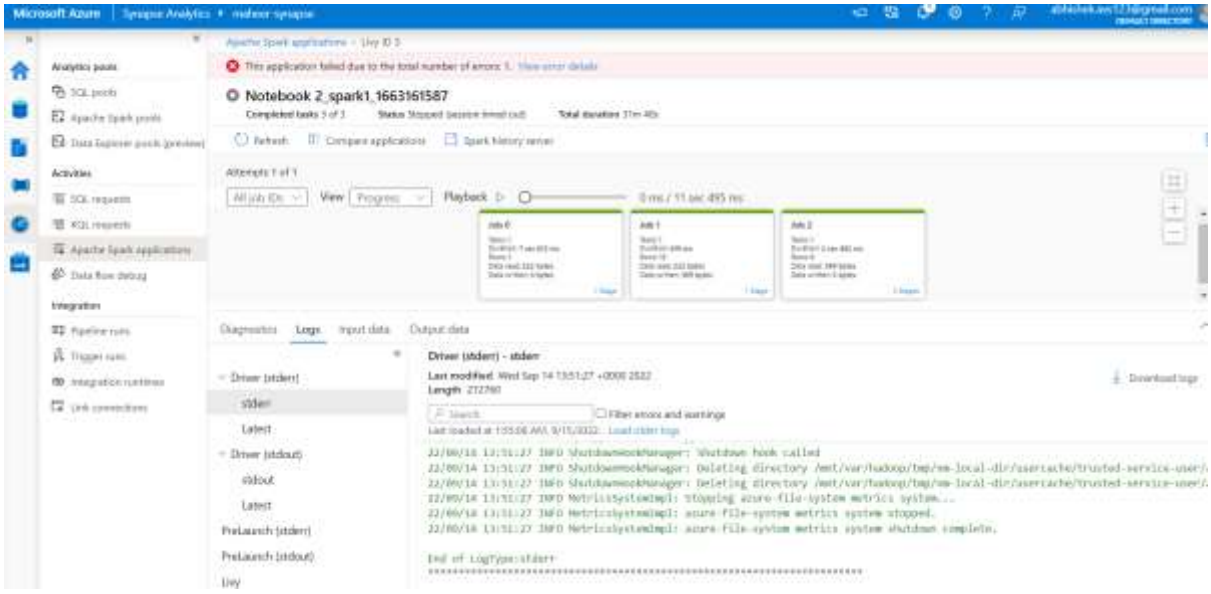
And inside the sql request ID u can see who and when it was submitted and the sql query too



In apache spark application gthere is list of spark session creation from notebooks are here.

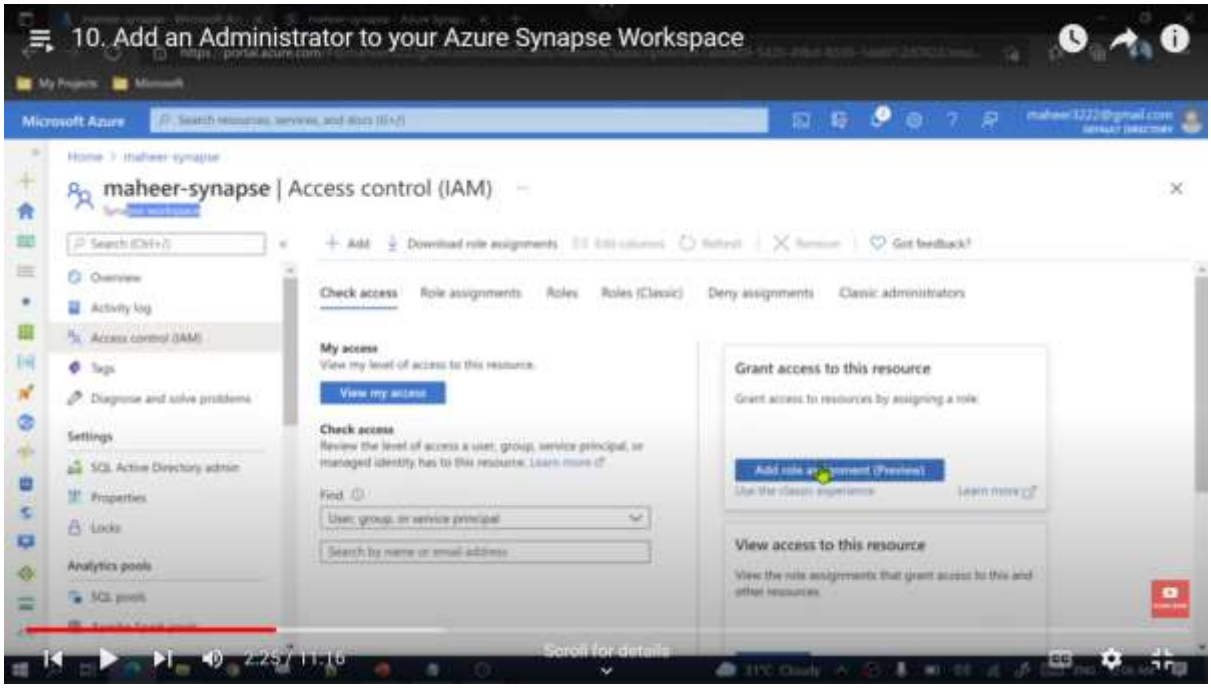


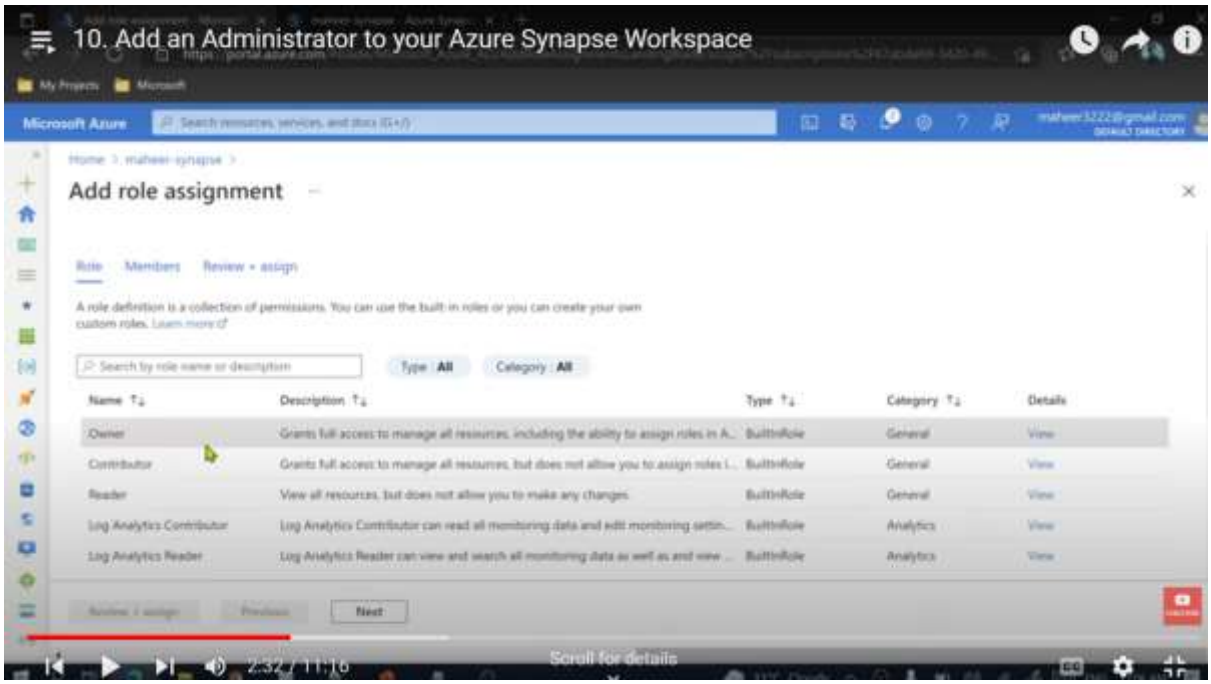
Here logs are available which is beneficial to us



10. Add an Administrator to your Azure Synapse Workspace

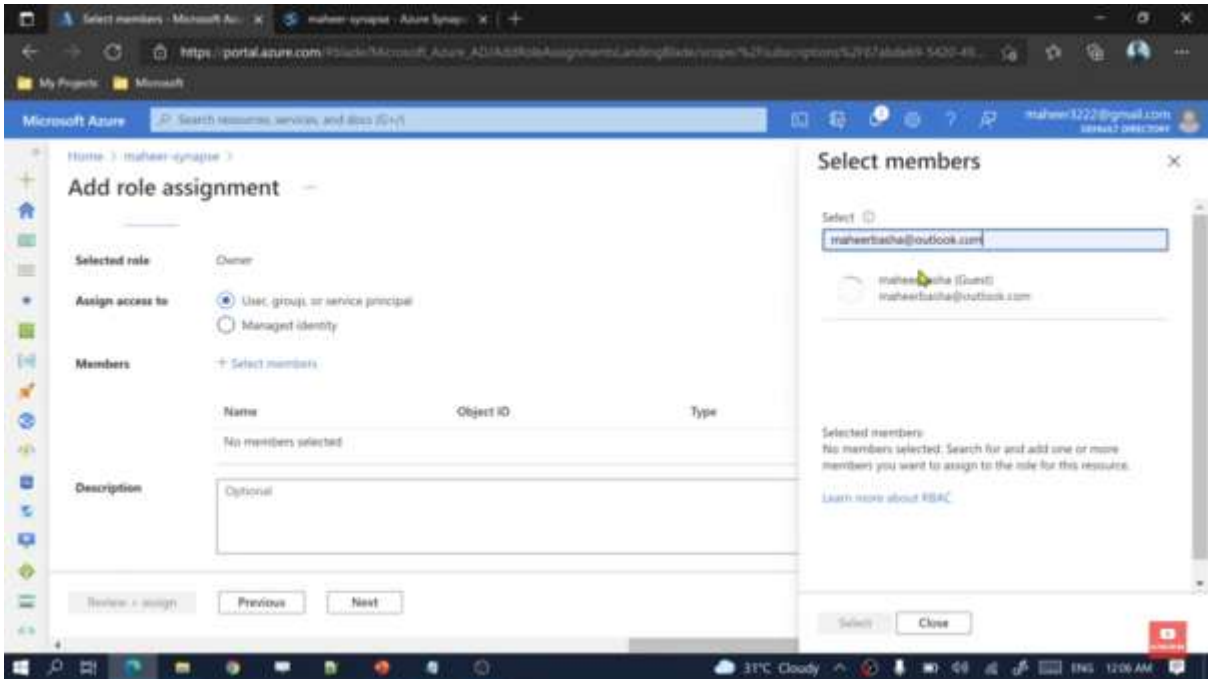
Azure role based access control system





Select owner role

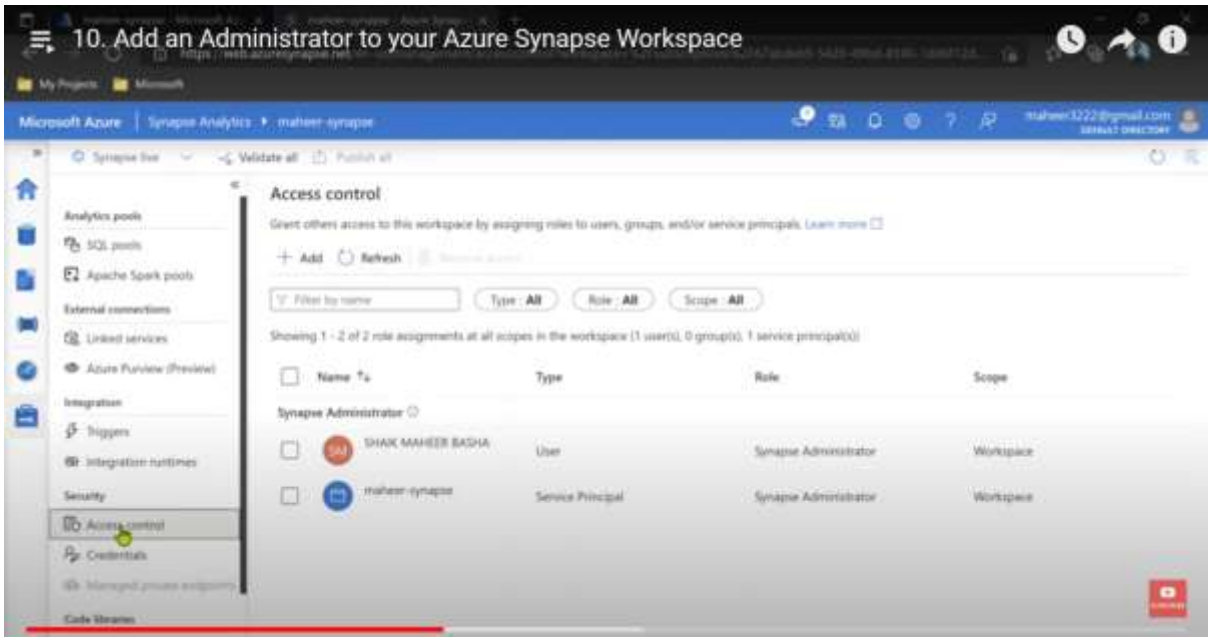
Then click next

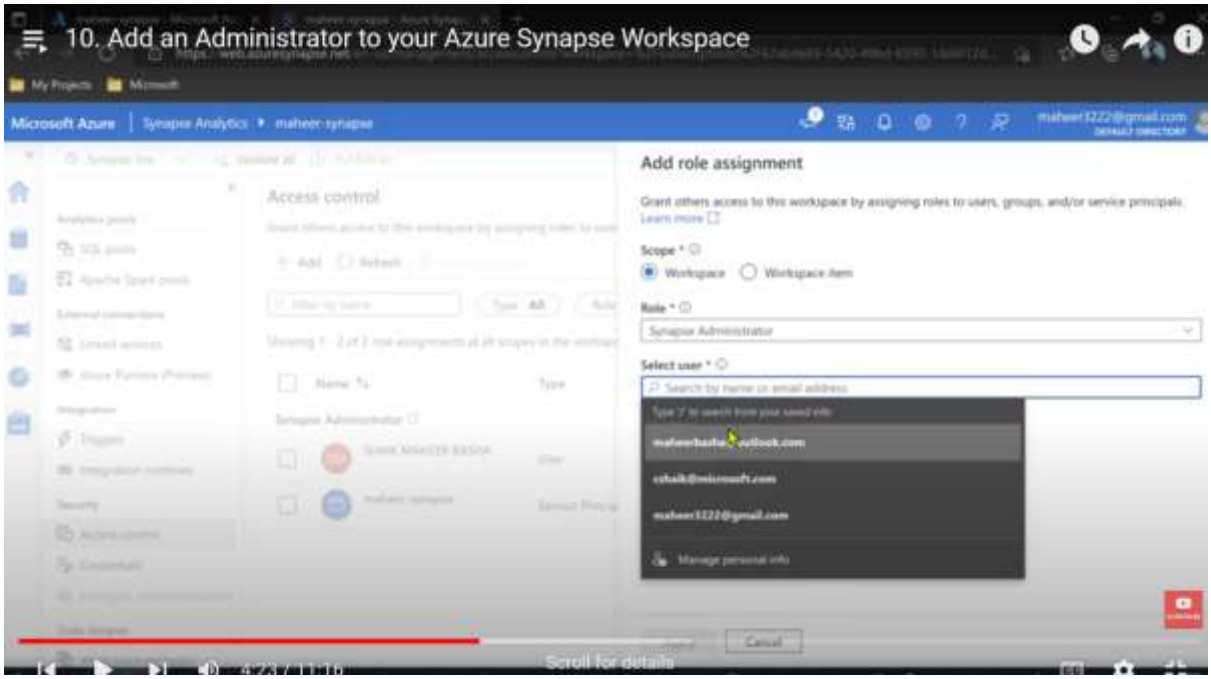


U have added it from the AZURE workspace

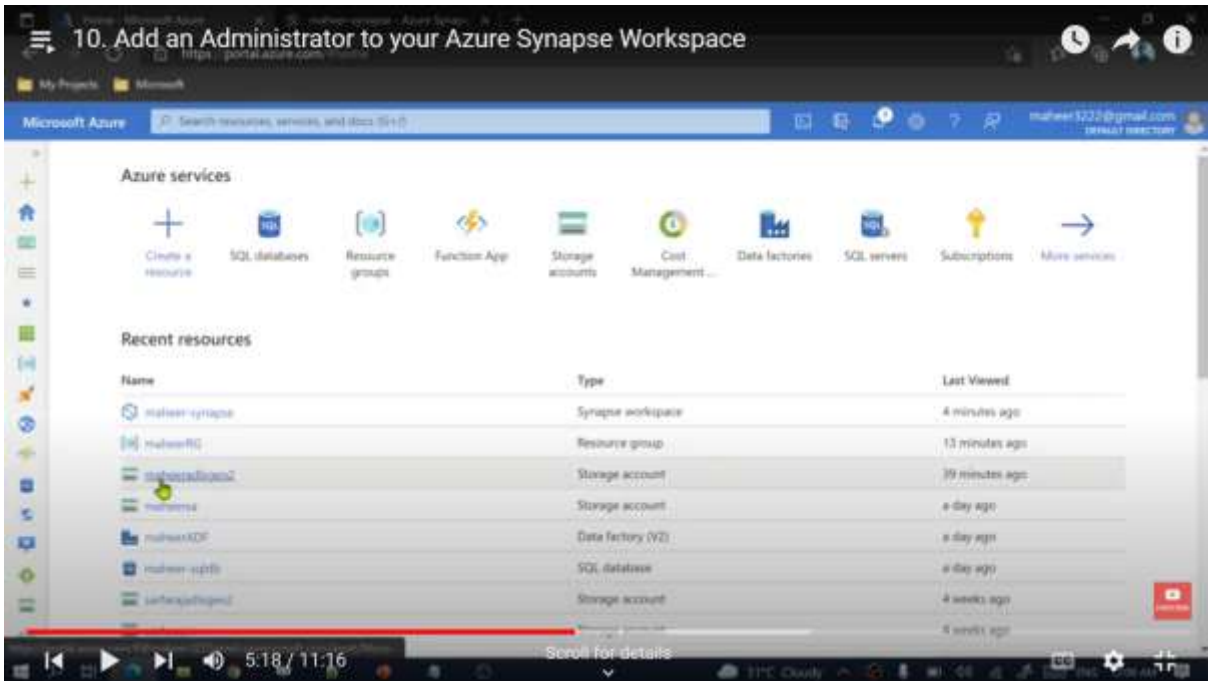
Then u have to also add him as owner from spnapse workspace too

So, go to synapse workspace, > manage tab> access control> add> select the role

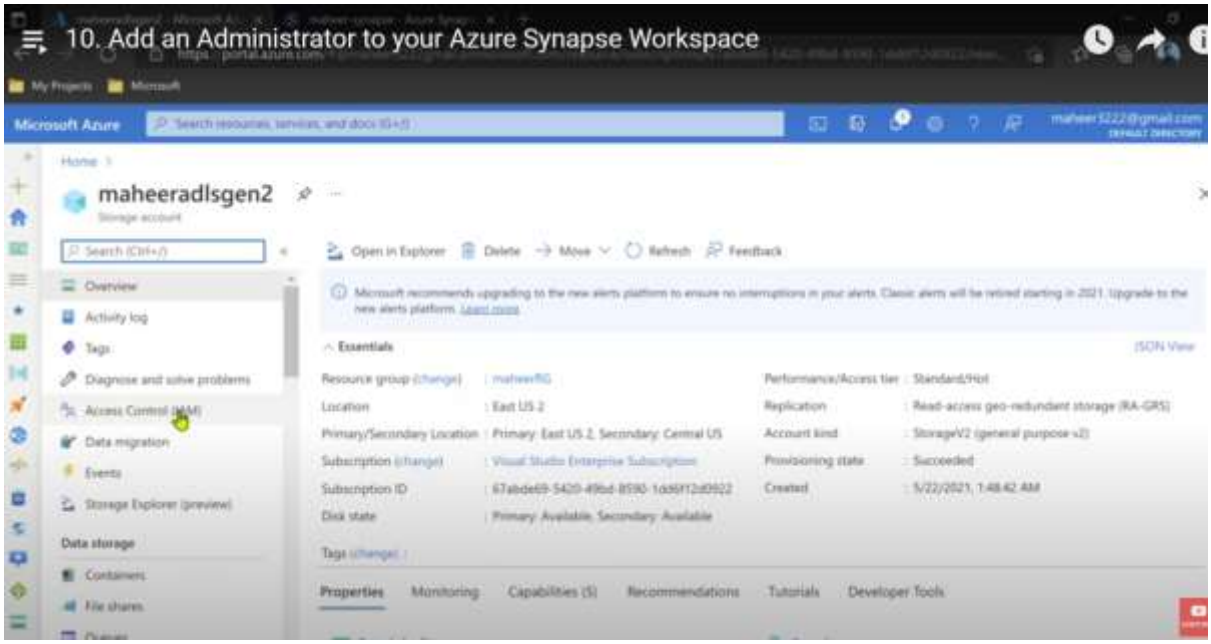


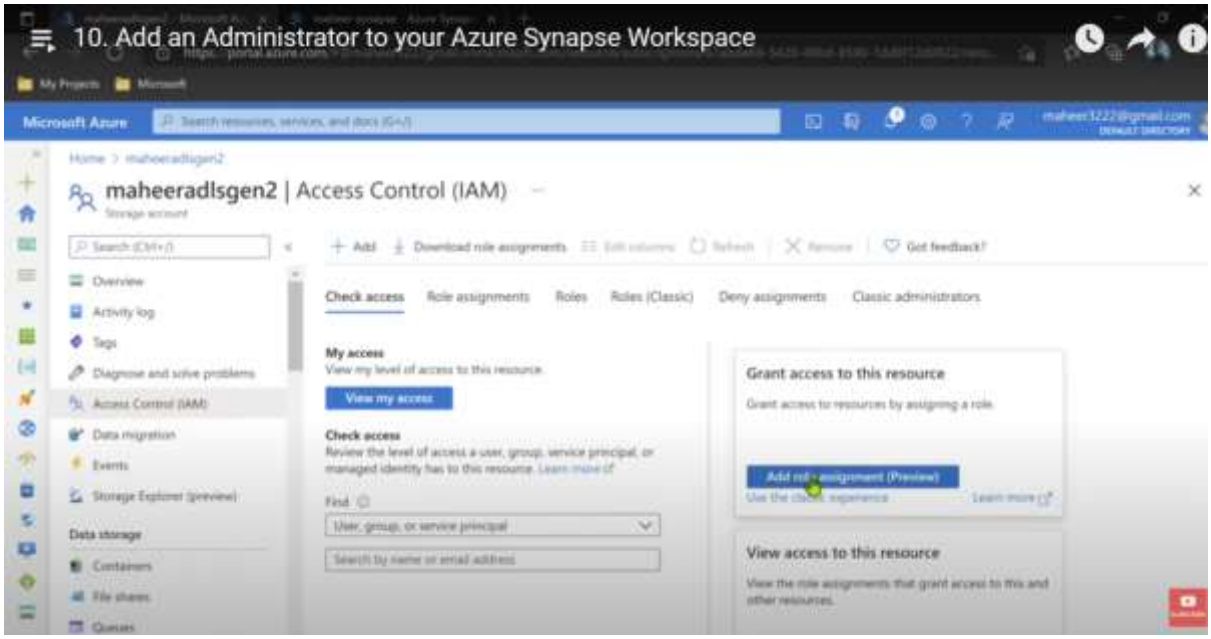


Then we need to give access of gen2 lake



Go to azure portal, open the gen 2 data lake



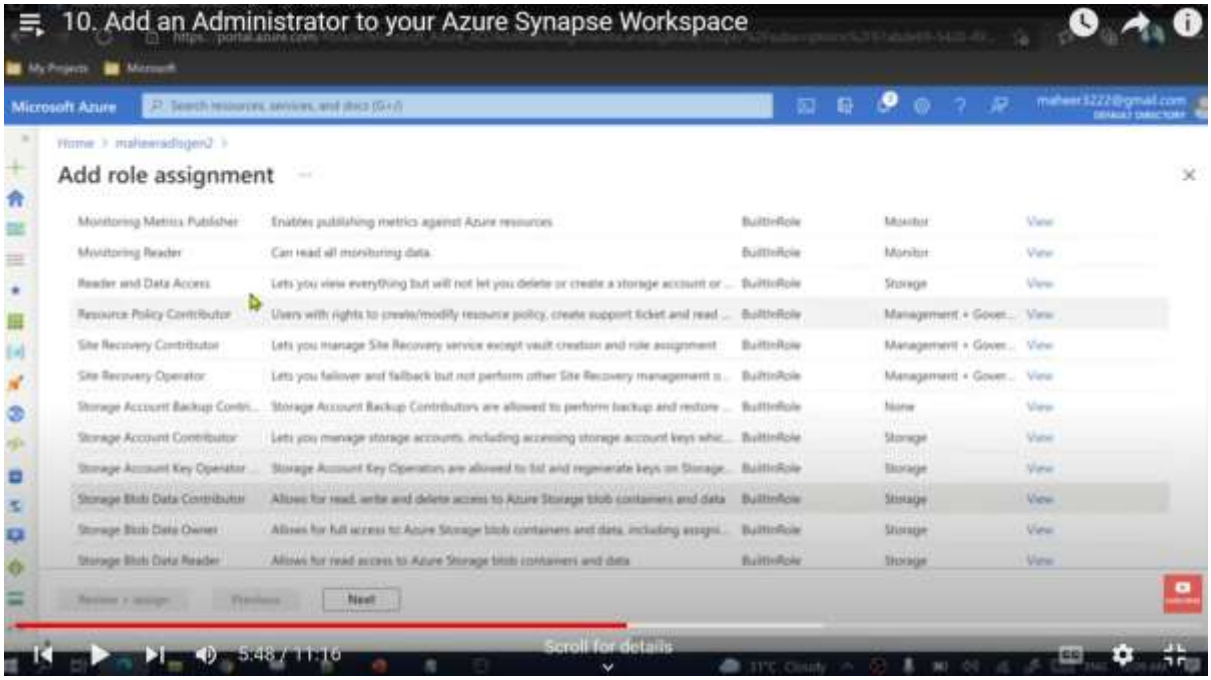


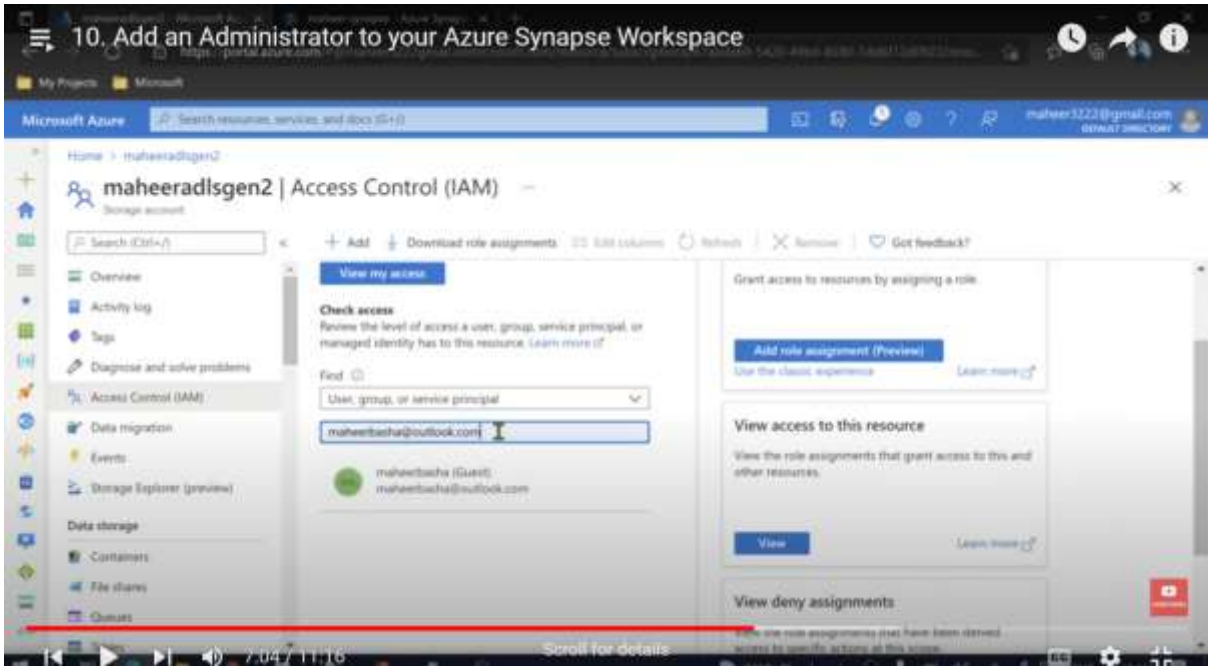
U NEED TO ADD 2 roles –

OWNER

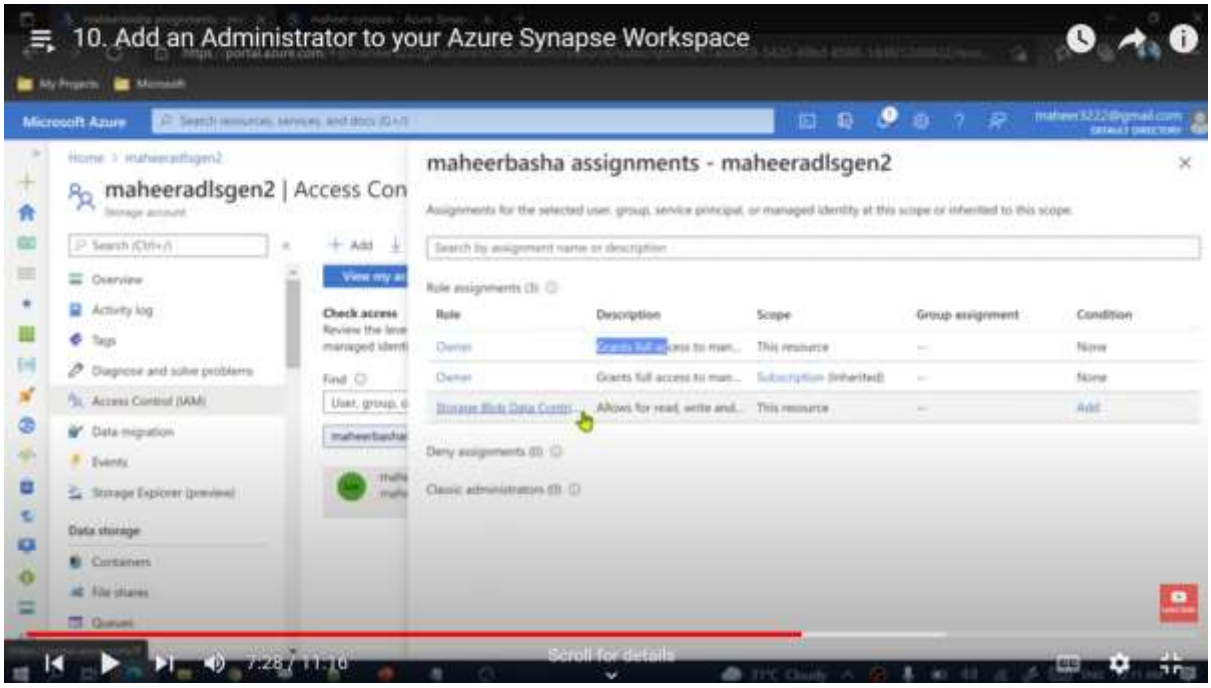
STORAGE BLOB DATA CONTRIBUTOR

U need to do it one by one

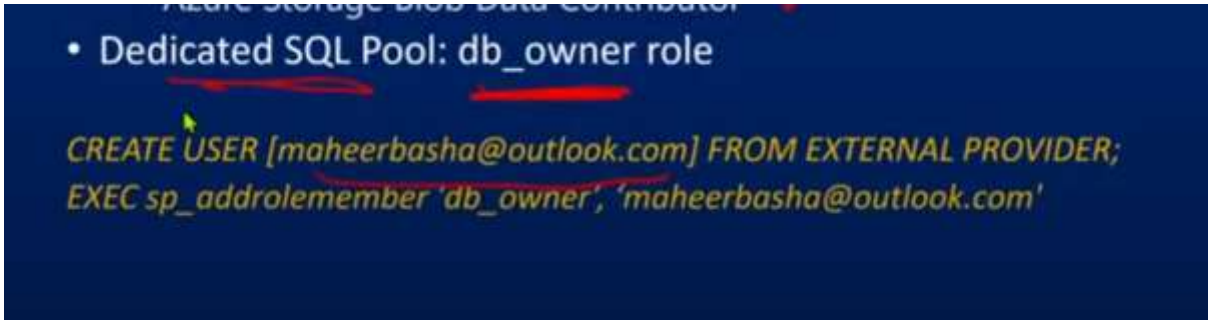




U can check the access of a user too



NEXT WE NEED TO ALSO ADD TO OUR DEDICATED SQL POOL



We create a user named maherbasha@outlook.com from an external provider and we are executing a stored procedure/function. And using that stored procedure, we r going to add this db_owner role for this user - maherbasha@outlook.com

So, this what we did

Add an administrator

- Azure RBAC: Owner role for the workspace
- Synapse RBAC: Synapse Administrator role for the workspace
- Azure RBAC: Below Role assignments on the Workspace's primary storage account
  - Owner
  - Azure Storage Blob Data Contributor
- Dedicated SQL Pool: db_owner role

```
CREATE USER [maheerbasha@outlook.com] FROM EXTERNAL PROVIDER;
EXEC sp_addrolemember 'db_owner', 'maheerbasha@outlook.com'
```

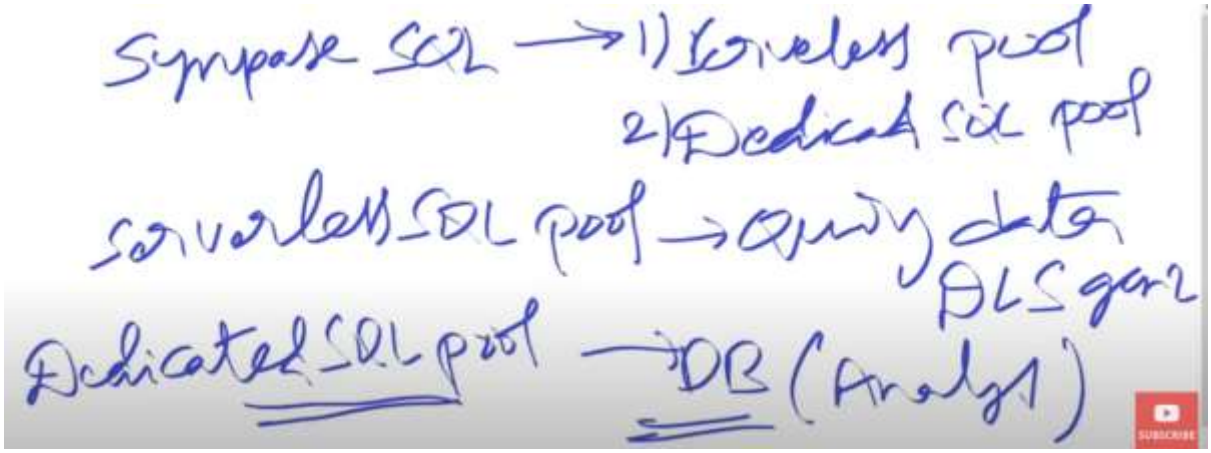
---

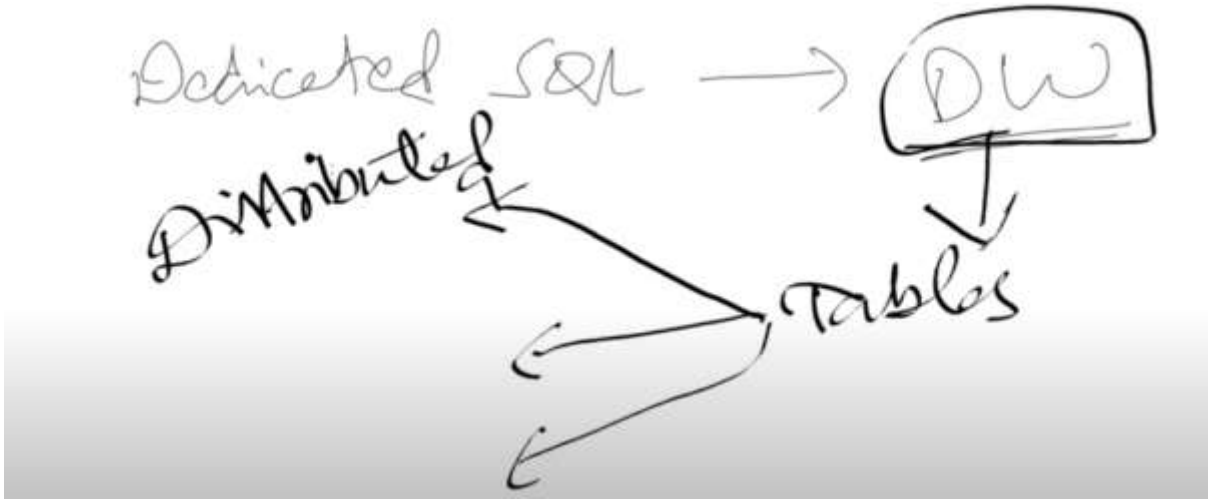
11. Azure Synapse SQL Architecture

Synapse SQL Architecture Components

- Synapse SQL uses a node-based architecture. Applications connect and issue T-SQL commands to a Control node, which is the single point of entry for Synapse SQL.

- When data is ingested into dedicated SQL pool, the data is sharded into distributions to optimize the performance of the system. You can choose which sharding pattern to use to distribute the data when you define the table. These sharding patterns are supported:
  - Hash
  - Round Robin
  - Replicate

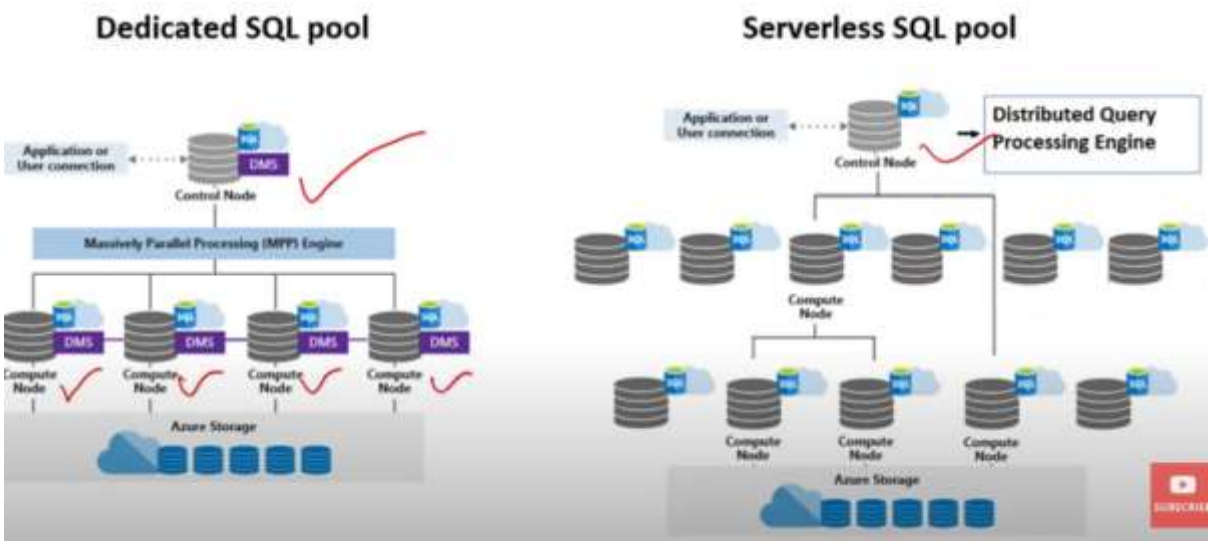




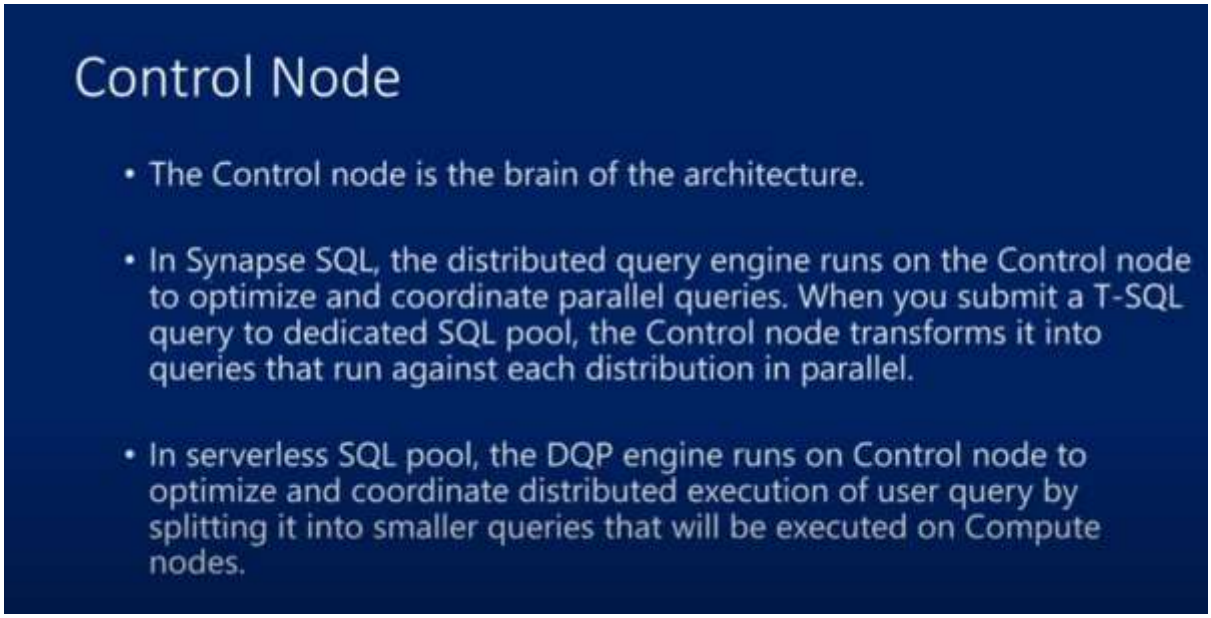
DEDICATED SQL POOL – Data is distributed in order to optimize the performance.

In this the data is stored in partitions or distributions

Synapse SQL uses node based architecture



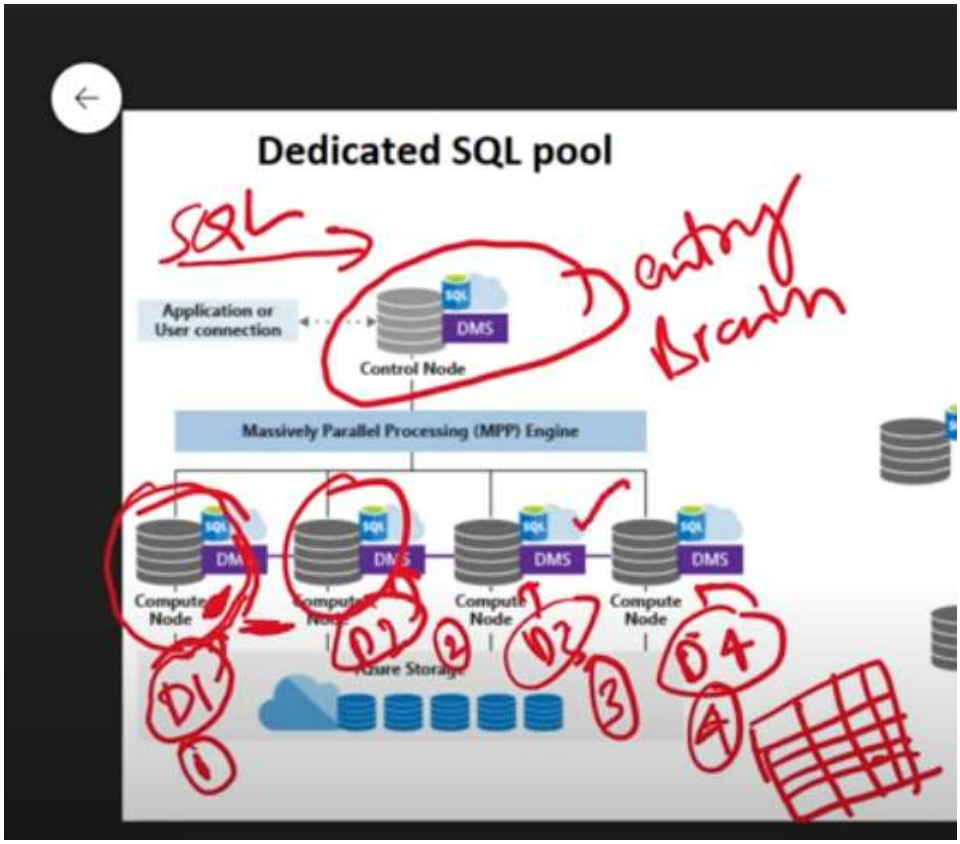
Control Node – Entry point and the Brain



The SQL query gets transformed into parallel queries and each parallel query will land on the control node. SO, each data distribution parallel query is take care by a control node. i.e. each query is going to take each distribution and it will process the distribution

<u>DEDICATED SQL POOL</u>

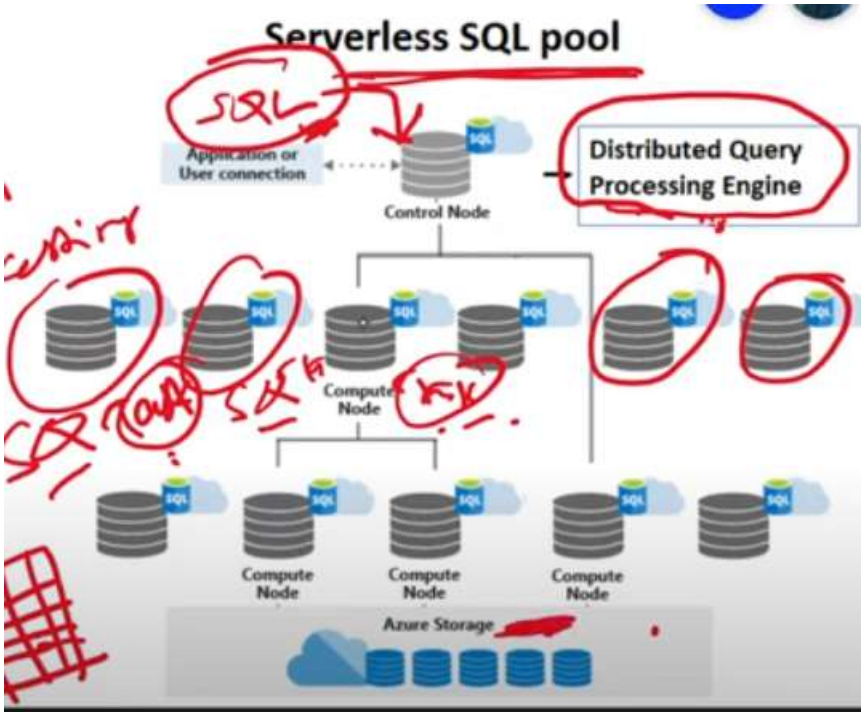
Massively parallel processing engine (MPP)



All this data has to clubbed in the end, so the data has to move in-between the compute nodes for which there is DMS(Data Movement Services)

Compute nodes are basically – processing engines / compute power

SERVERLESS SQL POOL

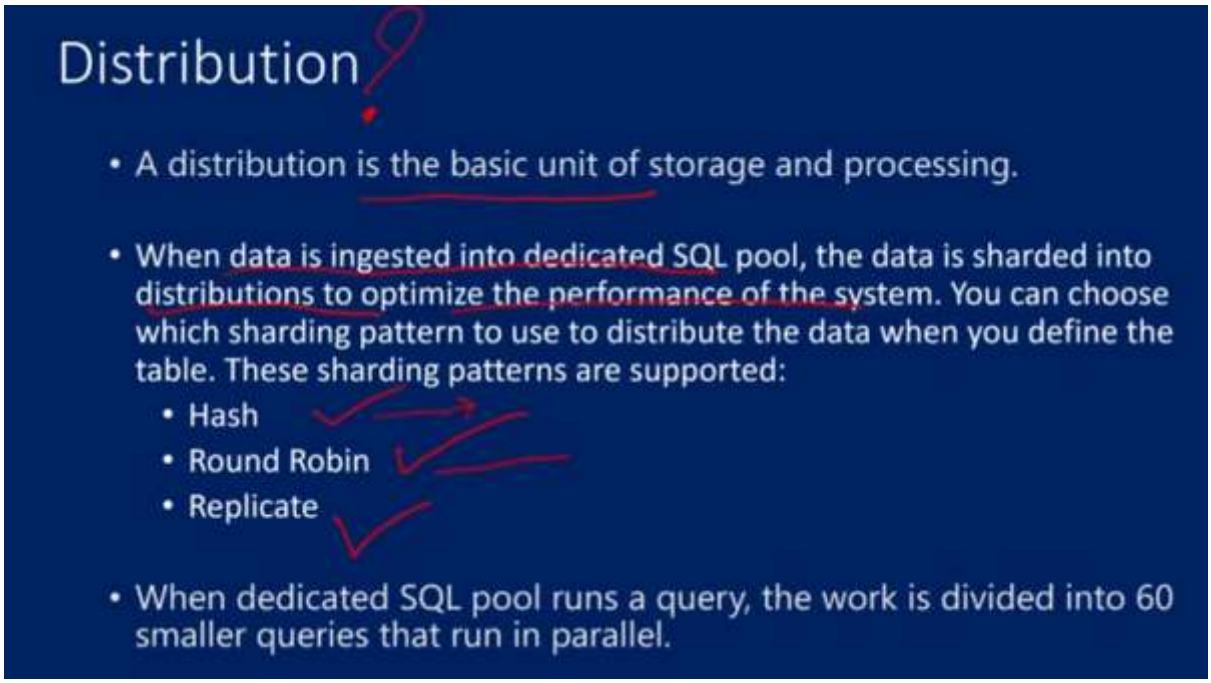


Sql query is passed to the control node and DISTRIBUTED QUERY PROCESSING ENGINE breaks it into small query/ small task. The data goes from our azure storage and it is processed. DISTRIBUTED QUERY PROCESSING ENGINE is responsible to optimize ur query and convert it into small small queries and assign the task to this compute nodes and these compute nodes will utilize azure storage to process the data

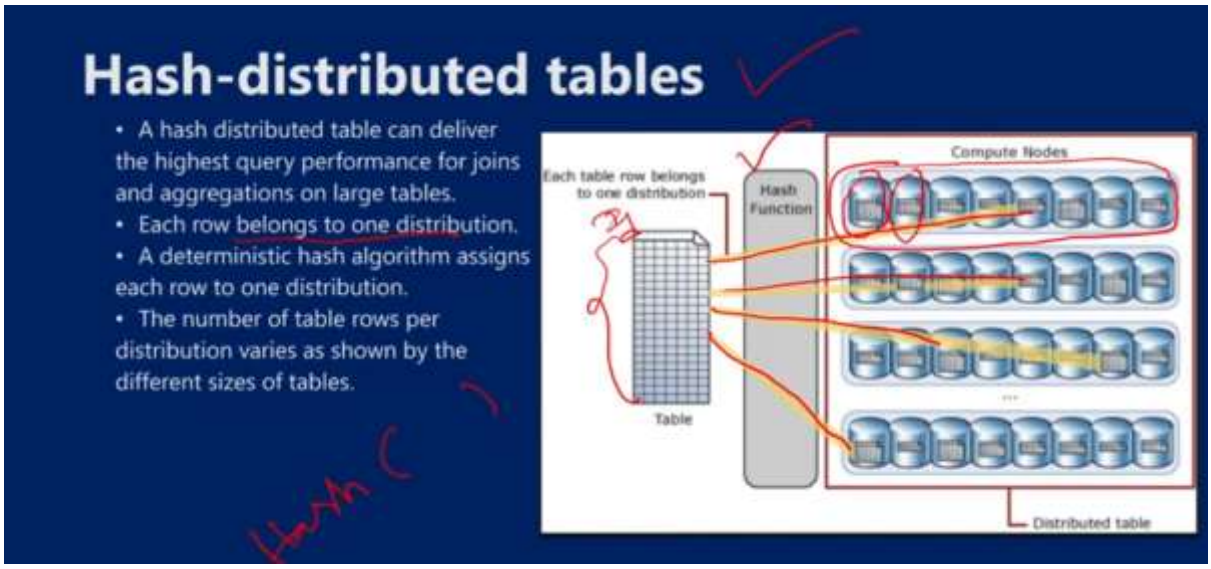
---

12. Distributions(Hash, Round Robbin & Replicate) in Azure Synapse Analytics

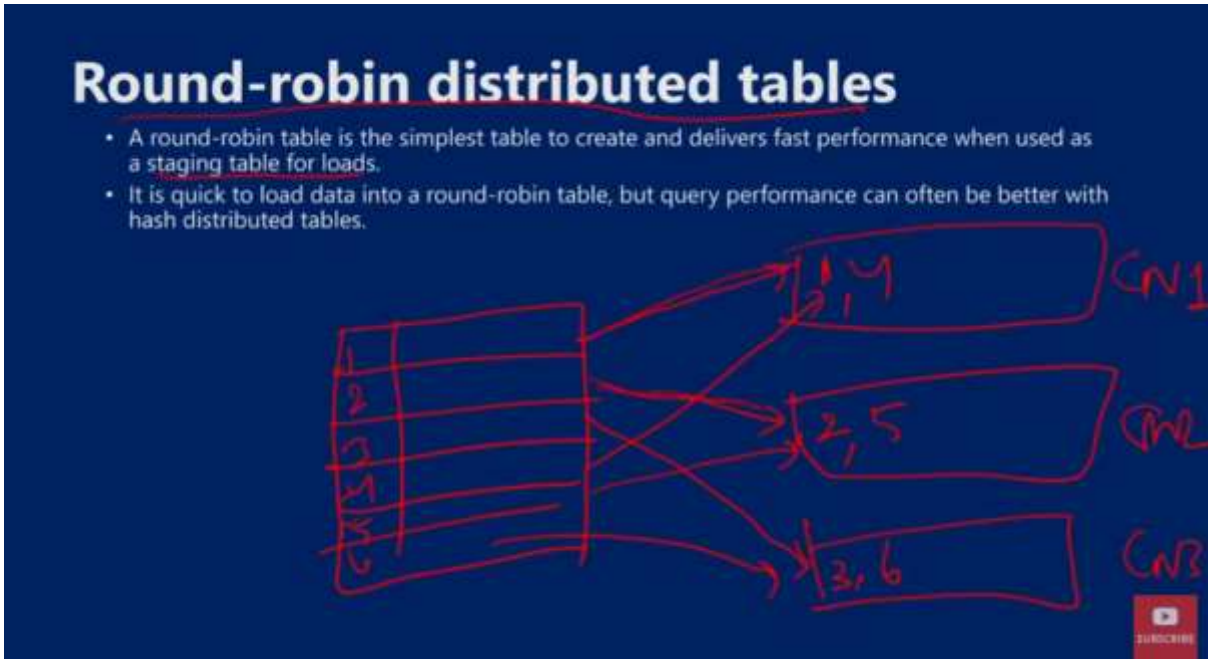
Distribution – basic unit of storage and processing



Hash distribution is used for huge data

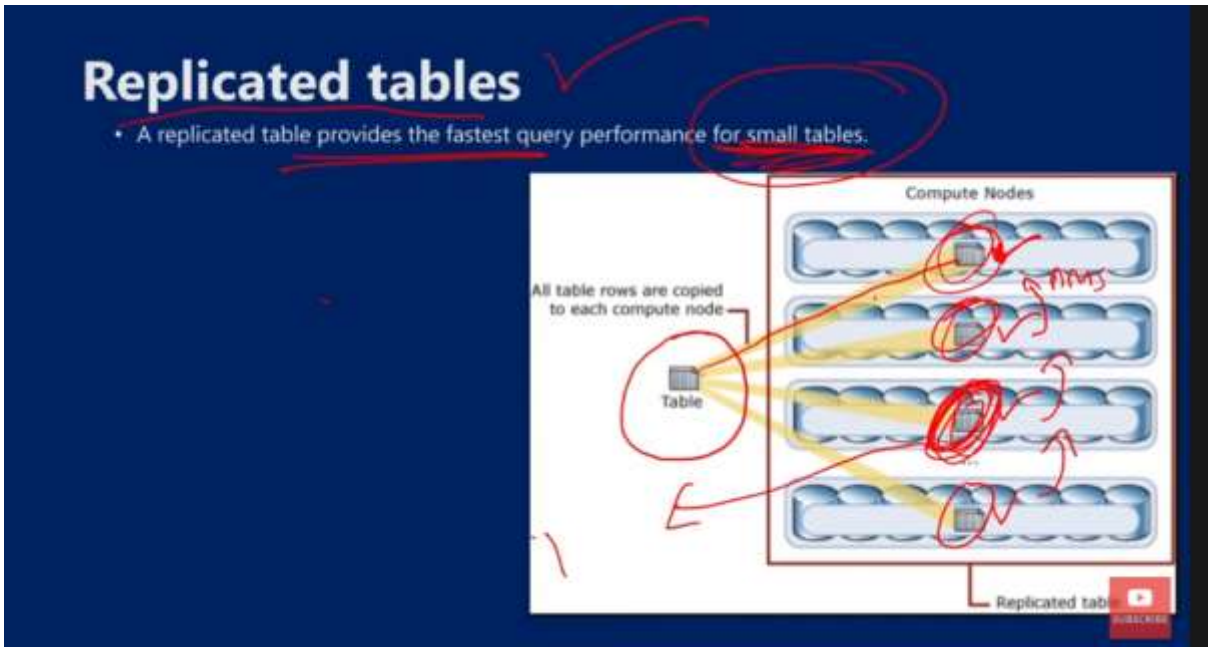


Round robin goes in circular sequence



Replicated tables

When data is in small tables, entire table is added into all the compute nodes. As in single compute node u have all ur data, so it computes fast, ther would be no need of DMS to take data from each compute node etc, so it is fast



---

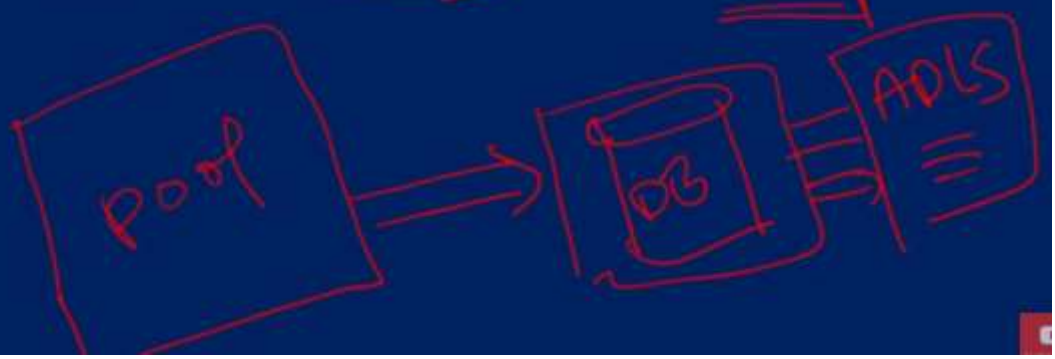
13. Server less SQL Pool Overview in Azure Synapse Analytics

Serverless SQL Pool

- Every Azure Synapse Analytics workspace comes with serverless SQL pool endpoints that you can use to query data in the Azure Data Lake (Parquet, Delta Lake, delimited text formats), Cosmos DB, or Dataverse.
- It allows you query data with T-SQL Syntax directly without need to copy or load data into a specialized store.
- Its true Pay as you go service. You will only pay for query execution

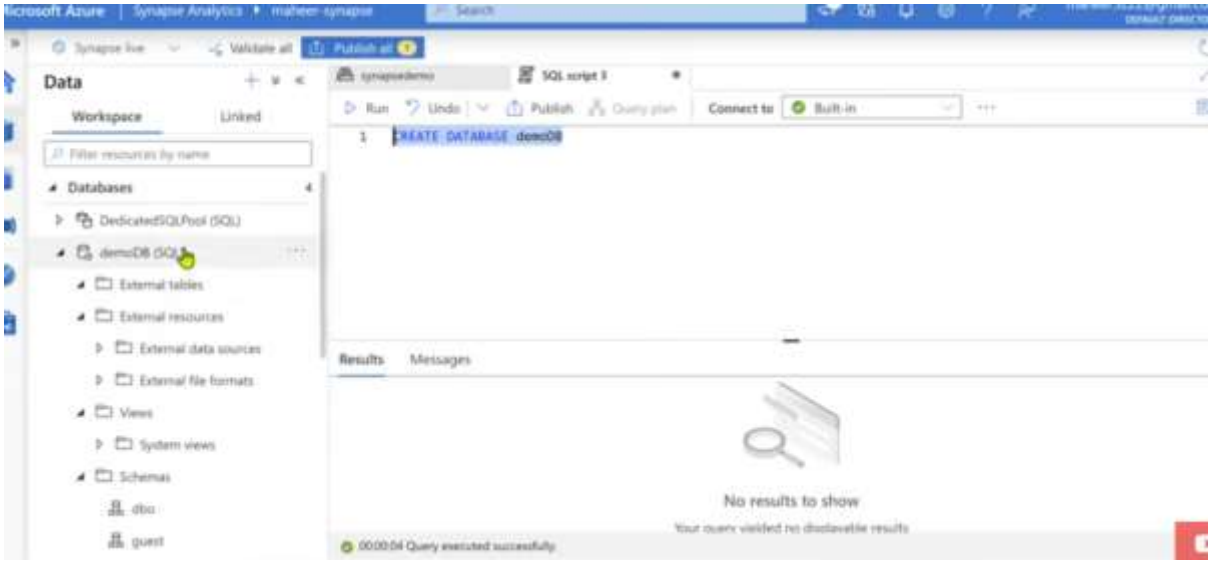
Serverless SQL Pool Benefits

- Basic discovery and exploration → Ad-hoc
- Logical data warehouse(LDW) → Imp
- Data transformation →



Serverless sql pool helps u to create a logical datawarehouse over the data lake etc db. It is just a layer on top of the data.

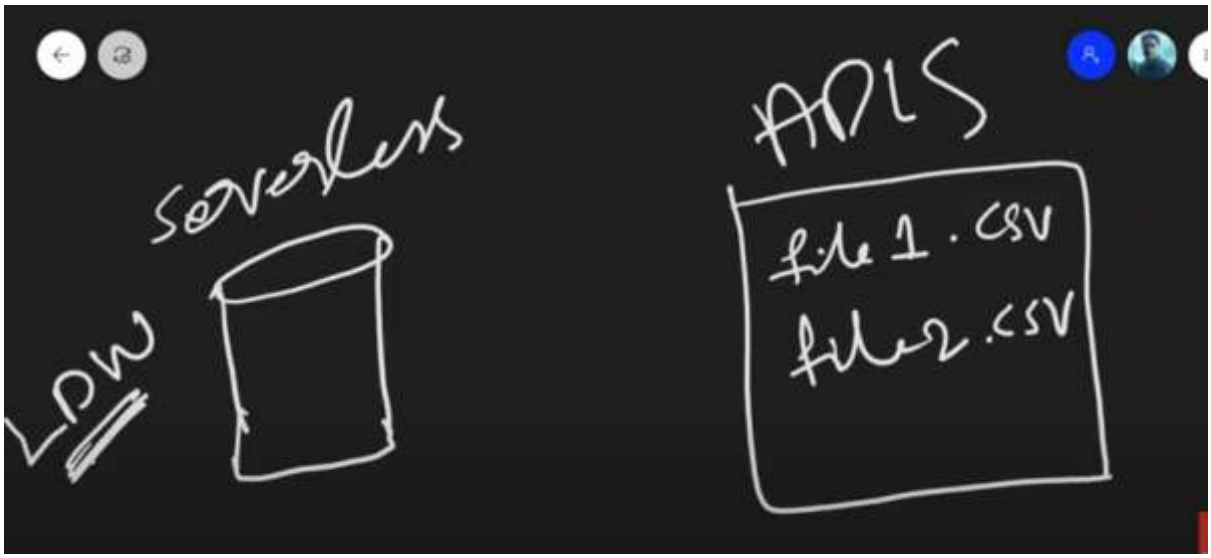
Databse has a power symbol, as it is just a layer, like it has no data , that is y there is external



Using server less sql pool we build the LDW(LOGICAL DATA WAREHOUSE)

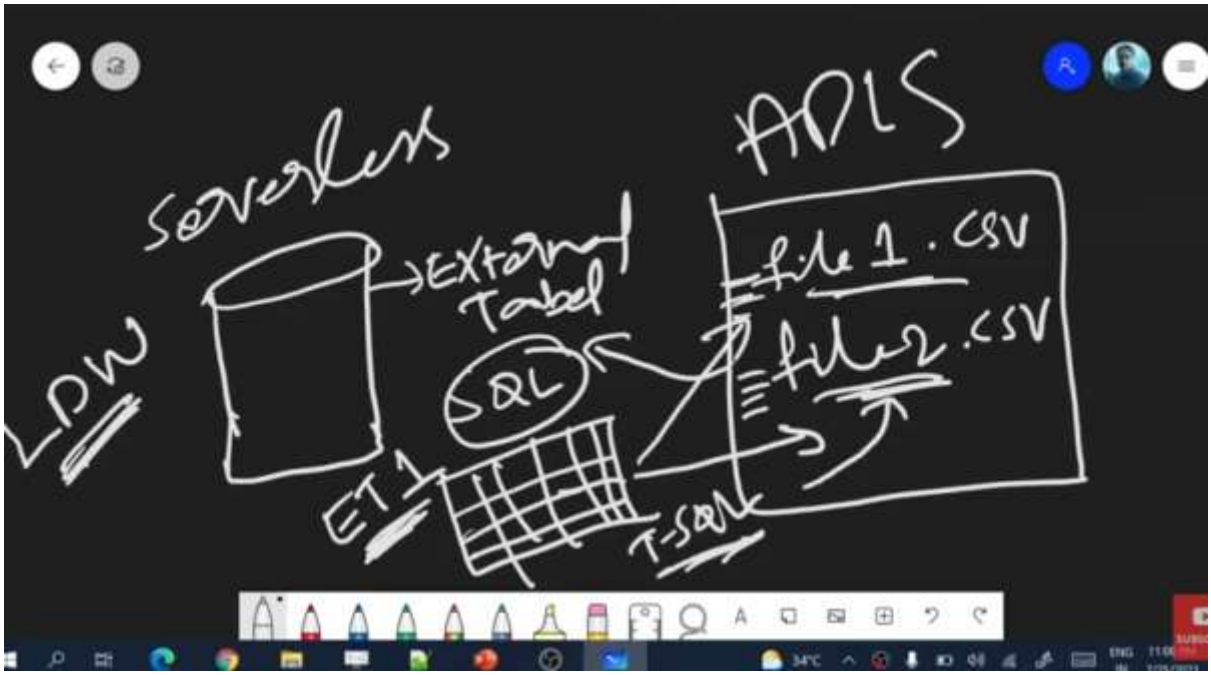
Whatever query we write for the data in azure datalake storage, those are used to store data in the external tables

LDW is a relational layer built on top of Azure data sources such as Azure Data Lake storage (ADLS), Azure Cosmos DB analytical storage, or Azure Blob storage.



Data will always come from the files only(adls) not on the LDW,

In ldw, we only store the metadata of particular external table



T-SQL support

It just offers querying surface area(LDW).

Schema is a group in which u have all the table and procedures etc but inside the schema u can only have external tables views etc

We can create views, stored procedures, inline table value functions, external resouces such as data sources, file formats and tables

Everything is external, nothing is stored. There iss no local storage for ldw. Only a metadata is there about table or view etc u created is there

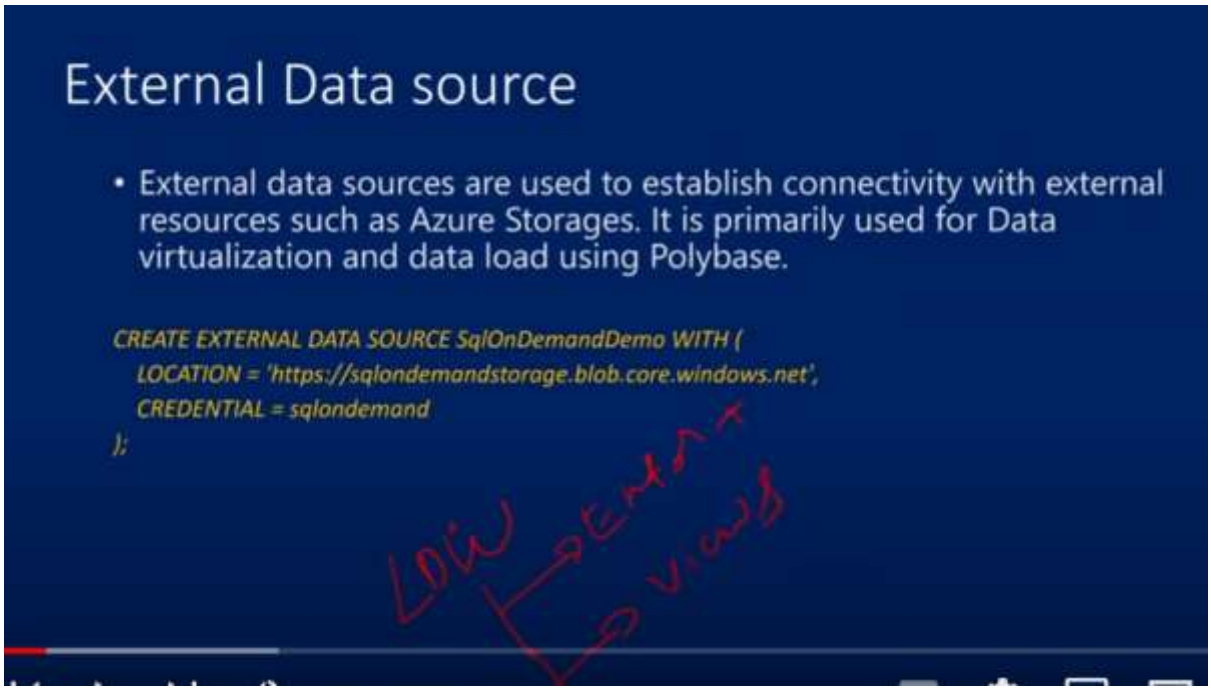
View – it will have sql query only

Materialized views – it will store the data by their own and has own memory alloacation which has data also.

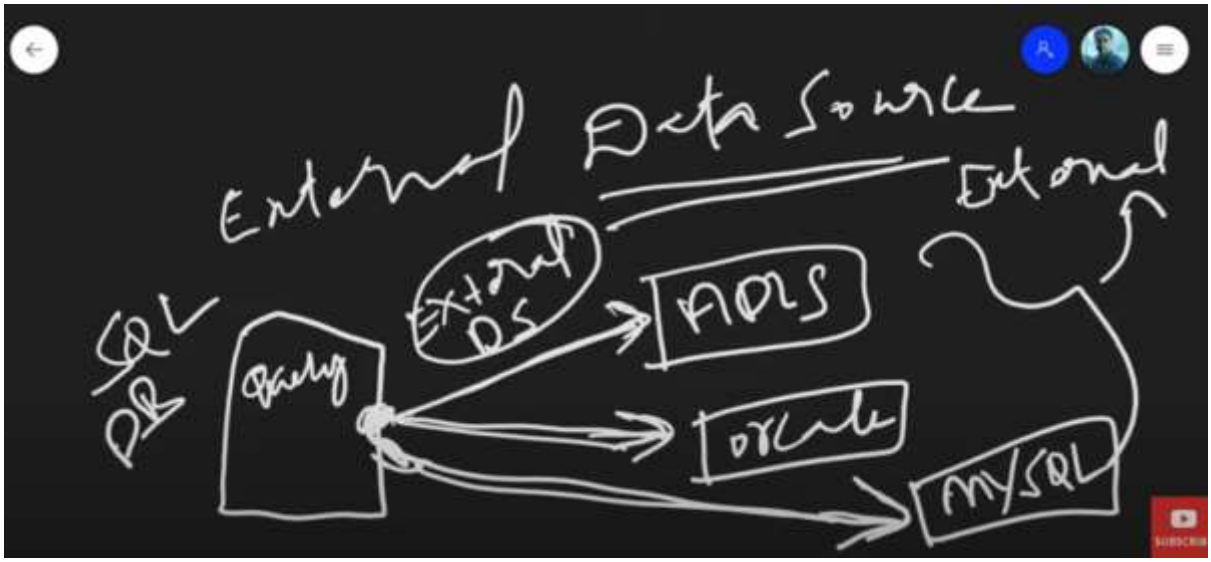
Data definition statements other than views isn't supported and so is Data manipulation statements

---

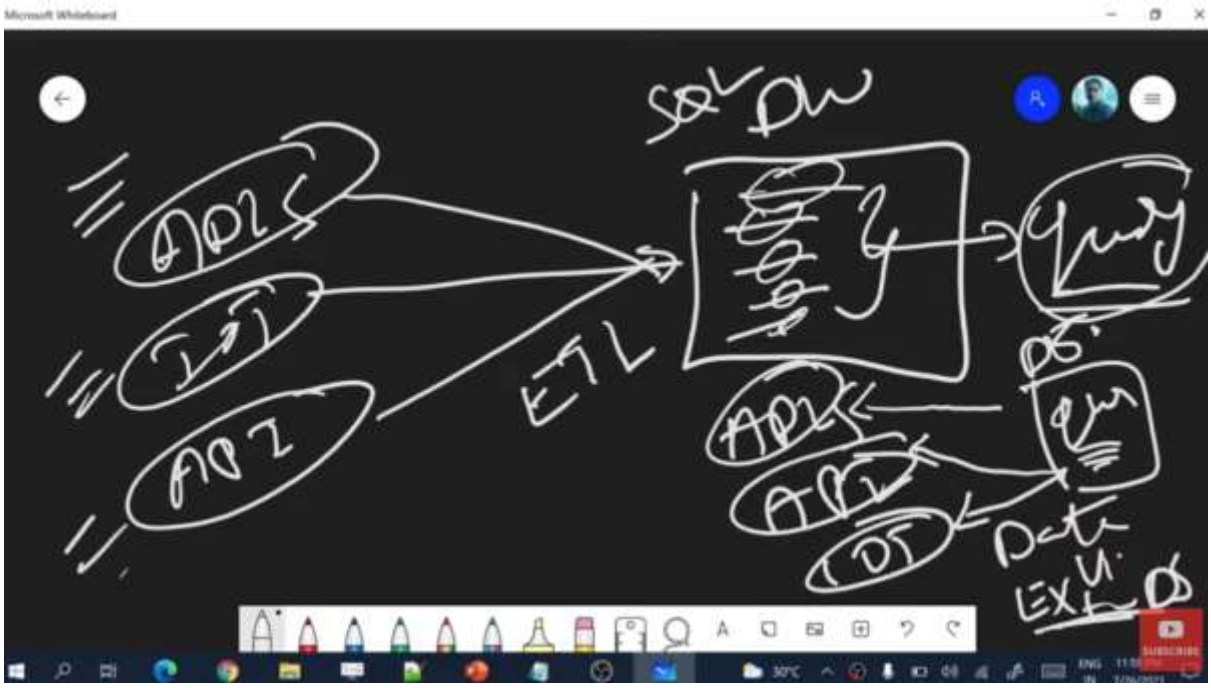
14. Create External Data source in Azure Synapse Analytics



On top of the data lake storage data, we can create external table that is stored in ldw



If we have some data in various locations in adls or oracle or mysql etc etc, we write a query to create a sql db and to get the access from those adls, oracle, mysql etc external data, we need to have external data source to get from external storages. They r like connection streams



We take the data and create a data ware house, then we run a query on the data warehouse data, That query is a local query, this is not data virtualization.

If write a query to interact with the other storages without moving the data from external storage, then, that is data virtualization, for which we need an external data sources

SYNTAX

External Data source

- External data sources are used to establish connectivity with external resources such as Azure Storages. It is primarily used for Data virtualization and data load using Polybase.

```
CREATE EXTERNAL DATA SOURCE SqlOnDemandDemo WITH (
    LOCATION = 'https://sqlondemandstorage.blob.core.windows.net',
    CREDENTIAL = sqlondemand
);
```

We need to CREATE the data source using WITH keyword with location of external storage and credential of the external storage.

For creating external data source for the maheer synapse data lake storage gen2

Credential

- A database scoped credential is a record that contains the authentication information that is required to connect to a resource outside SQL Server.

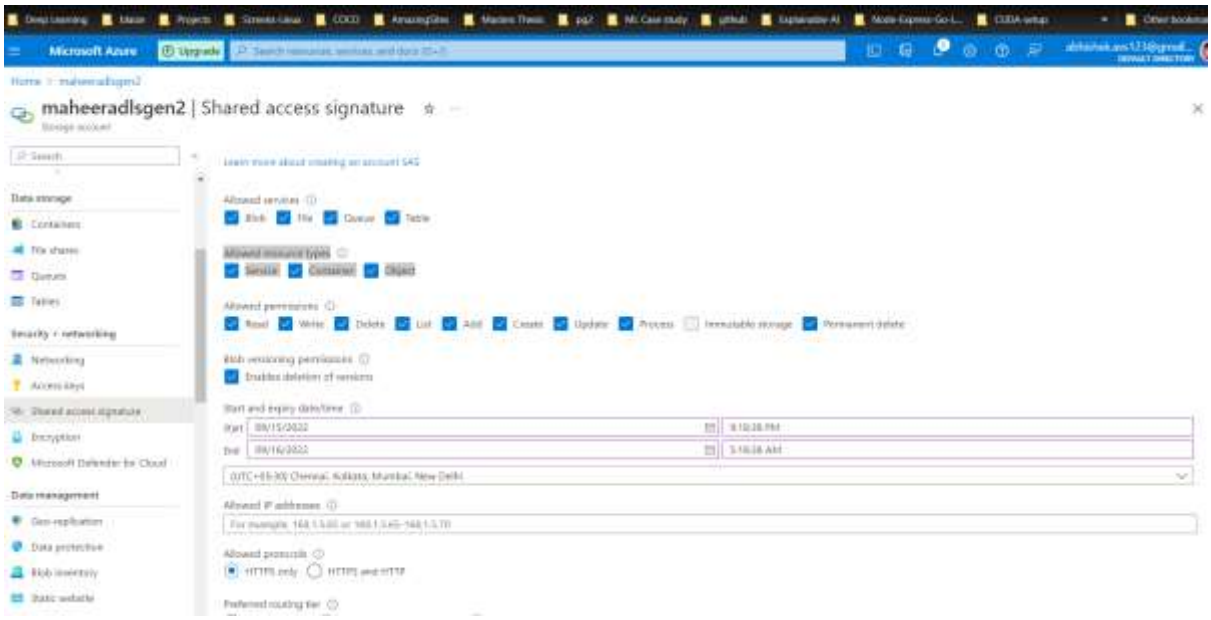
```
CREATE DATABASE SCOPED CREDENTIAL sqlondemand
WITH IDENTITY='SHARED ACCESS SIGNATURE',
SECRET = 'sv=2018-03-28&ss=bf&srt=sco&sp=rl&st=2019-10-
14T12%3A10%3A25Z&se=2061-12-
31T12%3A10%3A00Z&sig=KISU2ullCscyTS0An0nozEpo4tO5JAgGBvw%2FJX2lguw%3D'
```

Note, Before creating a database scoped credential, the database must have a master key to protect the credential.

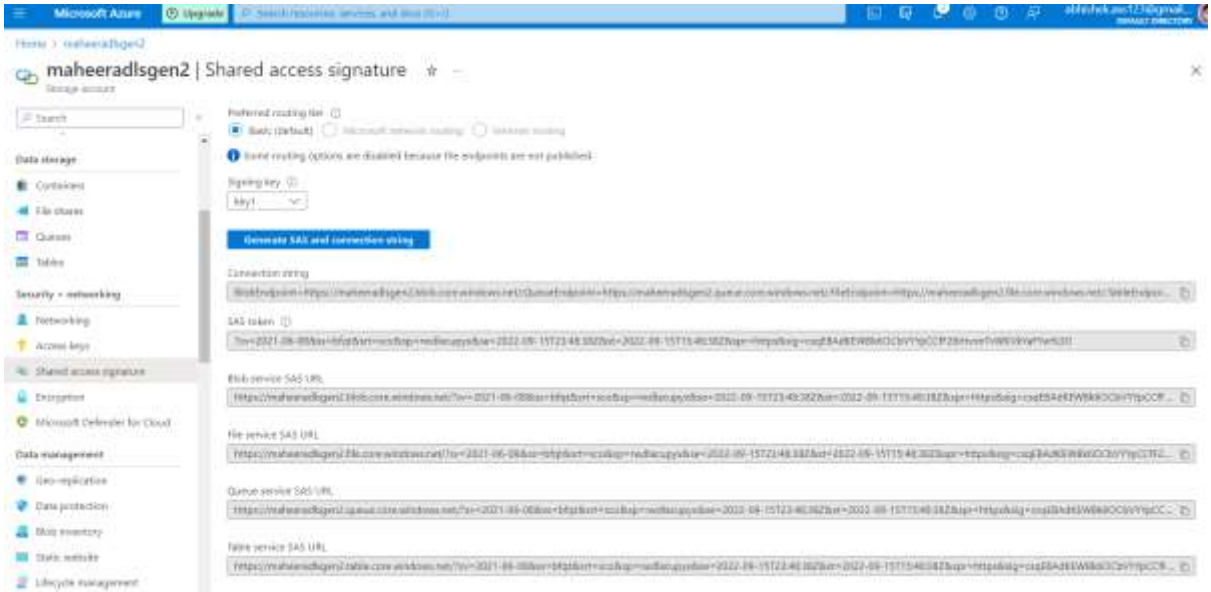
```
Use demoDB
GO
-- create master key that will protect the credentials:
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Abhishek@2000'


CREATE DATABASE SCOPED CREDENTIAL demoCredential
WITH IDENTITY = 'SHARED ACCESS SIGNATURE',
```

```
SECRET = 'sv=2021-06-08&ss=bfqt&srt=sco&sp=rwdlacupyx&se=2022-09-
15T23:48:38Z&st=2022-09-
15T15:48:38Z&spr=https&sig=csqEBAdKEWBk6OCbVYYpCCfF26lrtvonTvWKVkYaFYw%3D',
GO

CREATE EXTERNAL DATA SOURCE demoDataSource WITH(
    LOCATION = "https://maheeradlsgen2.dfs.core.windows.net",
    CREDENTIAL = demoCredential
);
```

To get the shared access signature, go to



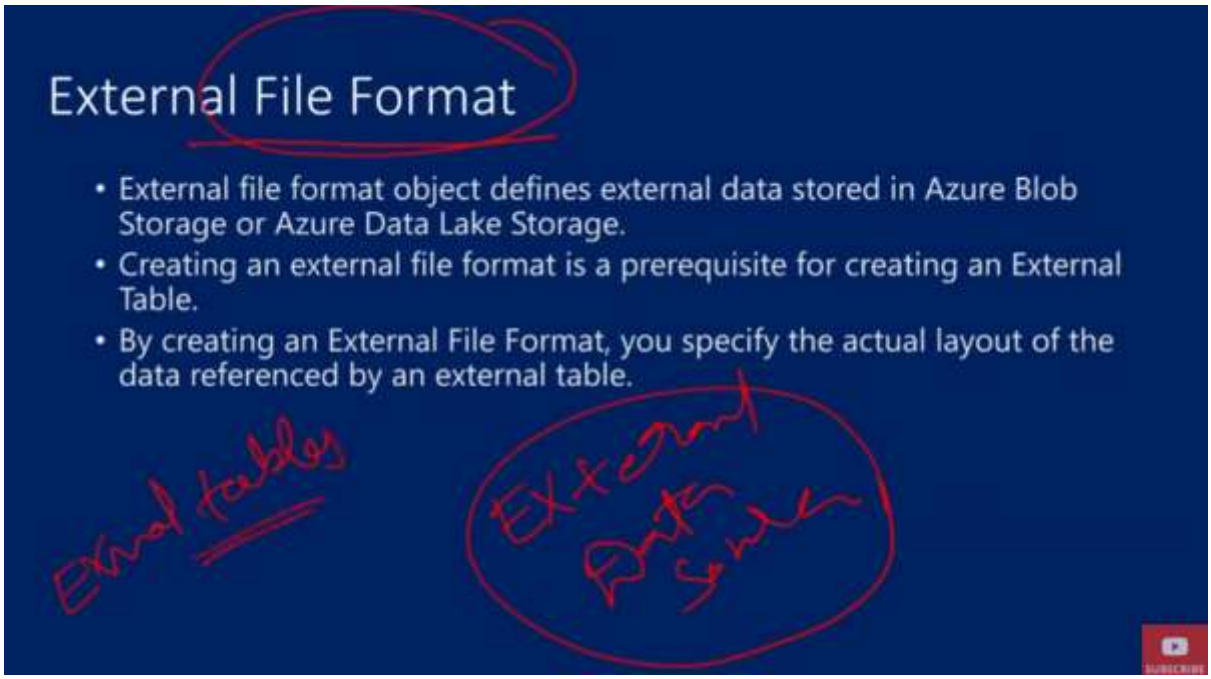
Tick the allowed resources types all 3 boxes- service, container, object> click on Generate SAS and connection string> then copy the SAS token and remove "?" and put it in the SECRET in above code

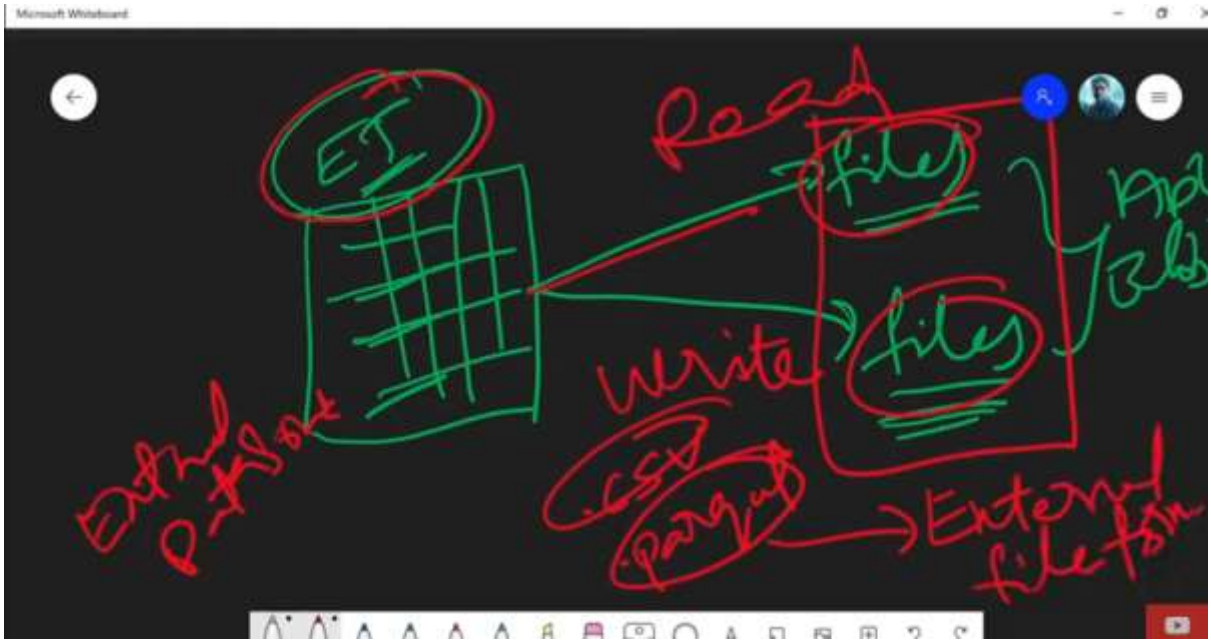


Once master key is created, then we create credentials in the database which will have authentication information of the ADLS. And once the authentication info is stored in the credential object called demoCredentials, we are creating external data source as demoDataSource, using the respective credential I want to access the location of the external data source as the credential that was created was the credential of this particular data lake storage.

---

15. Create External File Format in Azure Synapse Analytics



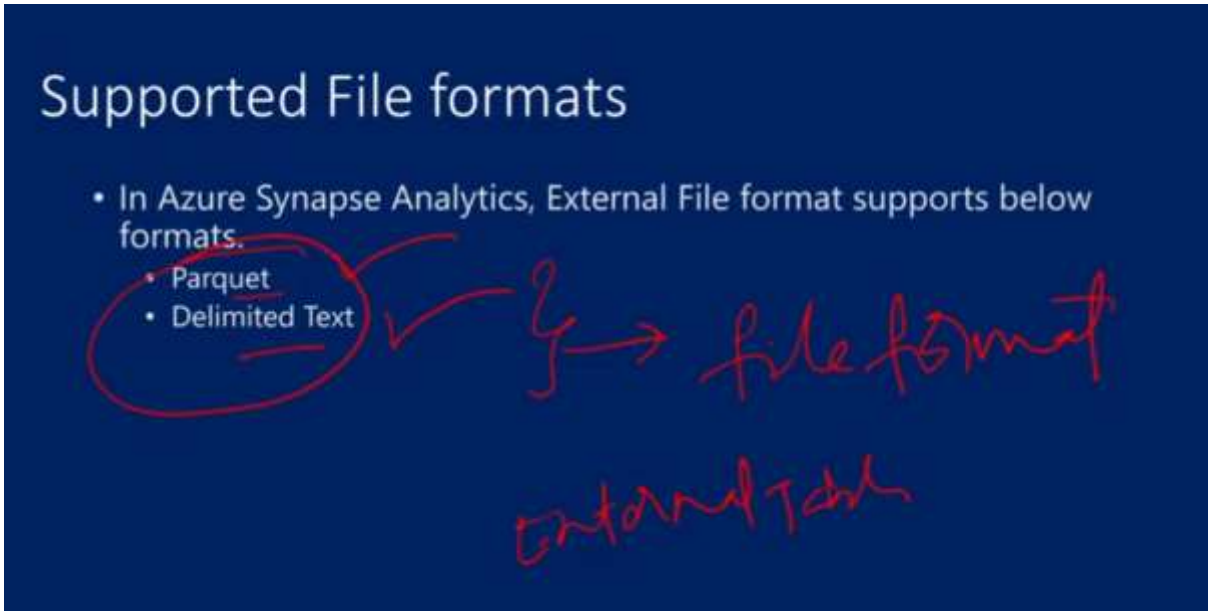


External data source is used to access the files in external storages, to read or write data in that storage. The files in the external storage will have some certain file formats like csv, parquet etc, for them we need to define the external file format object

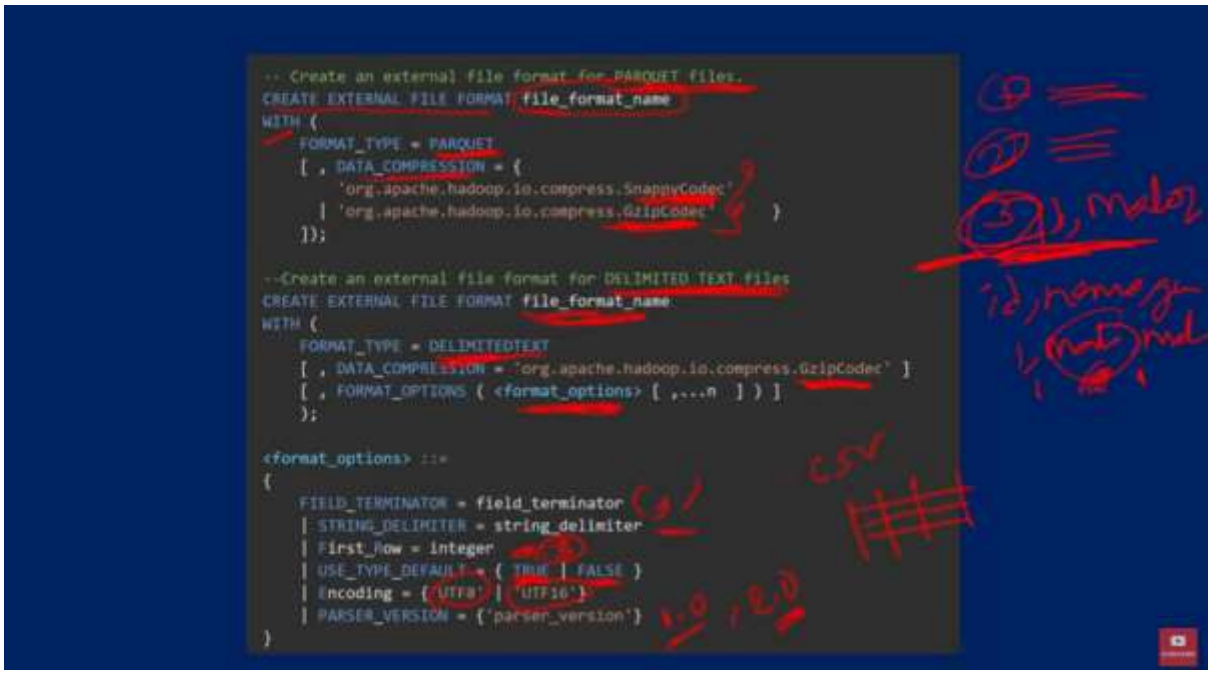
Entire layout of external table is defined in this file format object like

first row i.e starts from first row or some gap is there and from 3rd row the header starts,

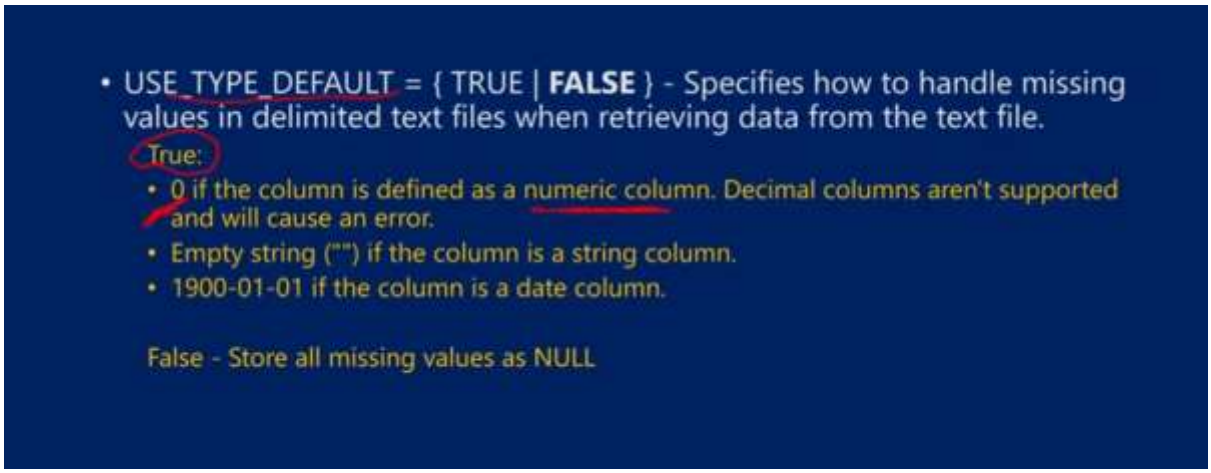
field i.e our data is separated by something like , or | etc



Syntax for creating the external file format



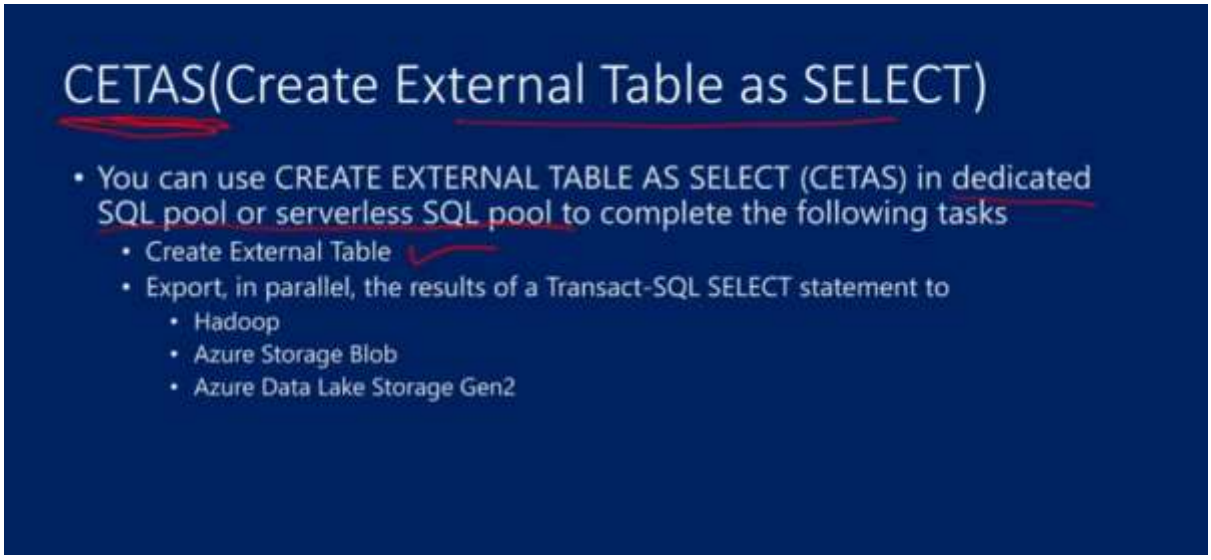
File compression format – SnappyCoded and GzipCodec for parquet file and for delimited text file only GzipCodec is there and we need to define a format options also



---

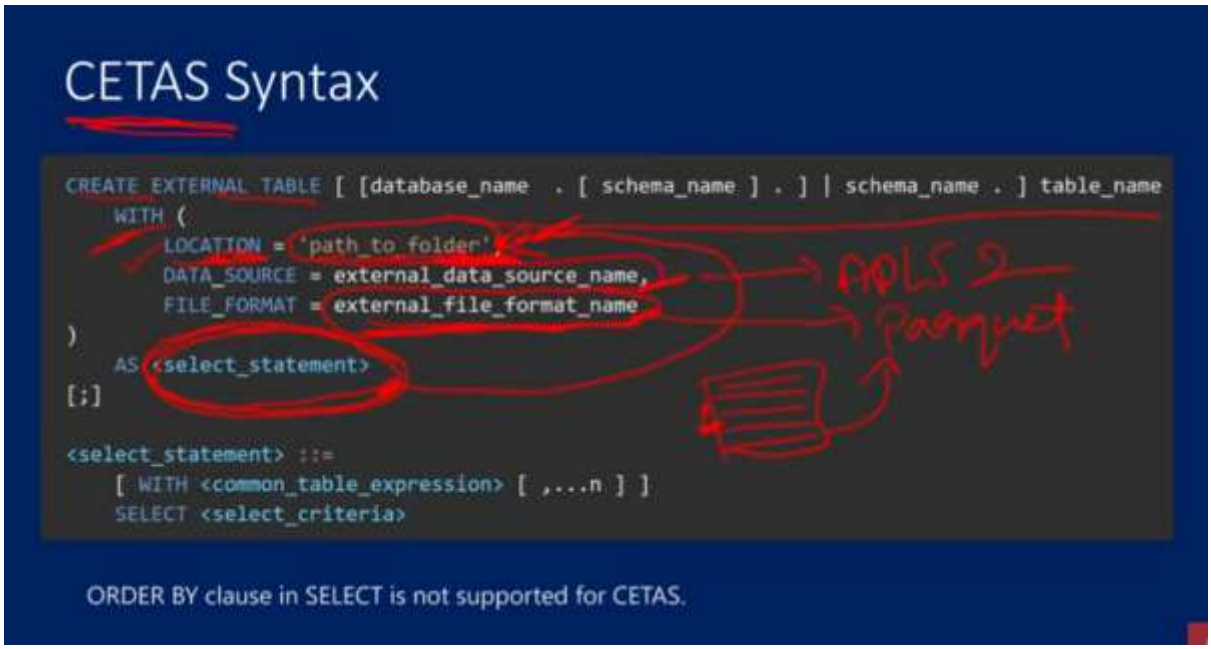
16. CETAS with Synapse SQL in Azure Synapse Analytics

CETAS – create external table as SELECT – this is an external table that is generate based on the output of the SELECT query




```
CREATE EXTERNAL TABLE [ [database_name  . [ schema_name ] . ] | schema_name . ] table_name
   WITH (
      LOCATION = 'path_to_folder',
      DATA_SOURCE = external_data_source_name,
      FILE_FORMAT = external_file_format_name
)
   AS <select_statement>
[;]

<select_statement> ::=
   [ WITH <common_table_expression> [ ,...n ] ]
   SELECT <select_criteria>
```

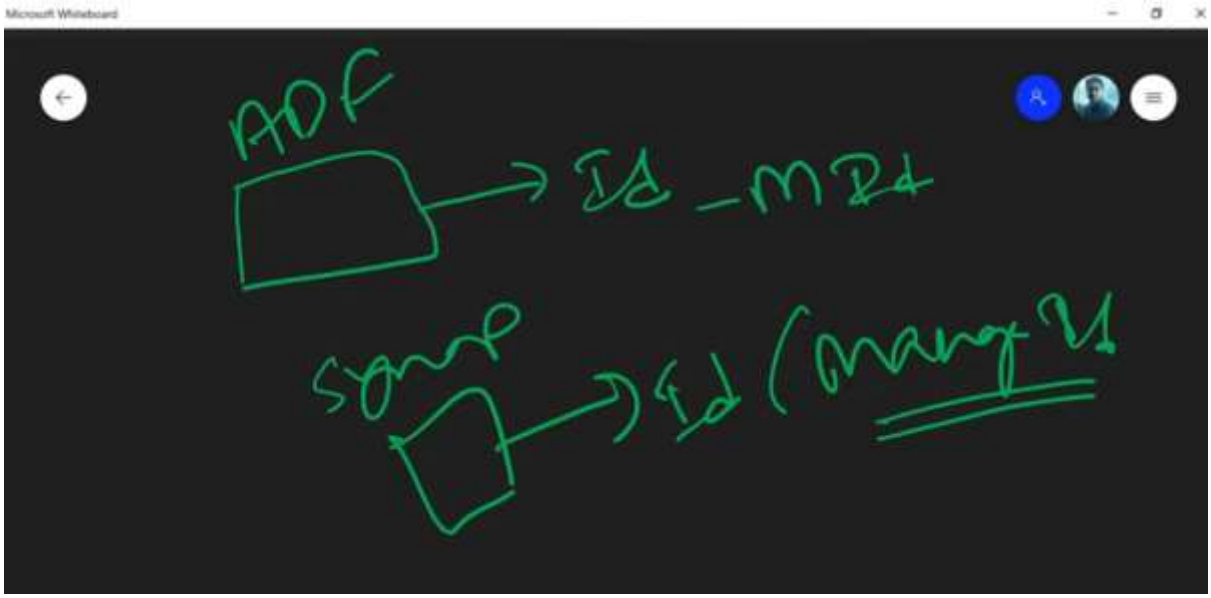
LOCATION is of the external storage and give the data source and file format name



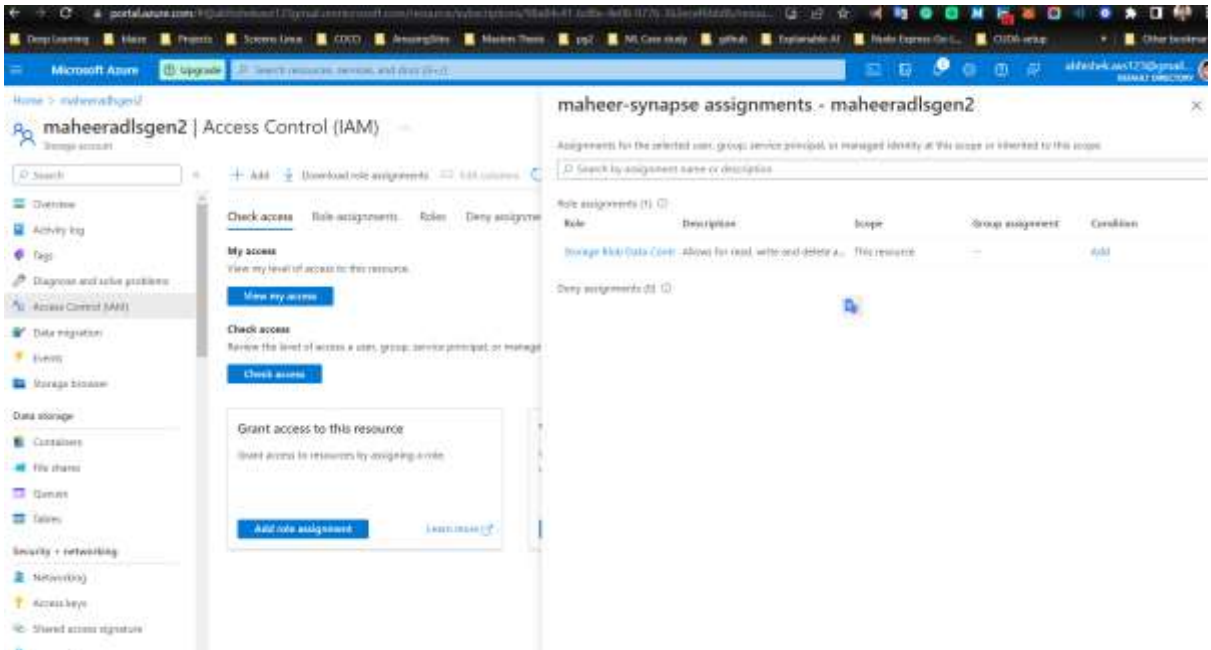
CTE – common table expressions which are temporary datasets

```
CREATE DATABASE demoDB
      COLLATE Latin1_General_100_BIN2_UTF8;

--
inside the database if u have any string data and in which format they are - U
TF8 or UTF 16 compression type
-
- if u r trying to write a query to read or write data from external table, u
will see an error that to add this collate in the db
-- to prevent this error use this collate

Use demo_db
GO
-- create master key that will protect the credentials:
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Abhishek@2000'


-- CREATE DATABASE SCOPED CREDENTIAL demoCredential
-- WITH
--      IDENTITY = 'SHARED ACCESS SIGNATURE',
--      SECRET = 'sv=2021-06-08&ss=bfqt&srt=sco&sp=rwdlacupyx&se=2022-09-
15T23:48:38Z&st=2022-09-
15T15:48:38Z&spr=https&sig=csqEBAdKEWBk6OCbVYYpCCfF26lrtvonTvWKVkYaFYw%3D'
-- GO

CREATE DATABASE SCOPED CREDENTIAL ManagedIdentity
WITH
    IDENTITY = 'Managed Identity'
```

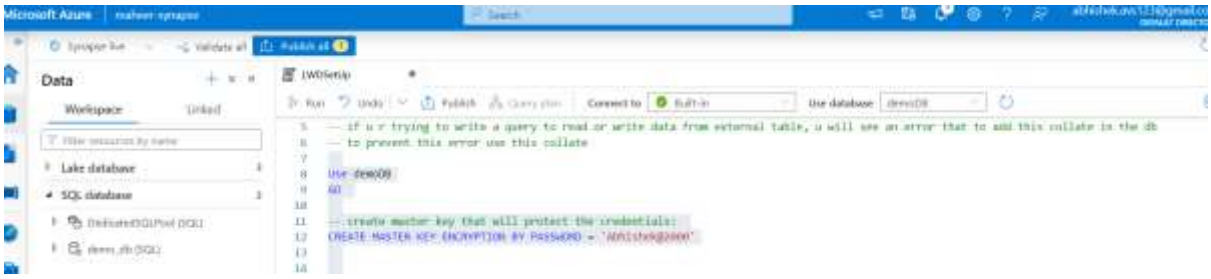
```sql
GO

CREATE EXTERNAL DATA SOURCE demoDataSource
WITH(
    LOCATION = 'https://maheeradlsgen2.dfs.core.windows.net',
    CREDENTIAL = ManagedIdentity
);
```



Managed identity – it Is same as the workspace name. in our case the workspace name is maheer-synapse, so this name is my managed identity. For that ID we are creating the credential object here



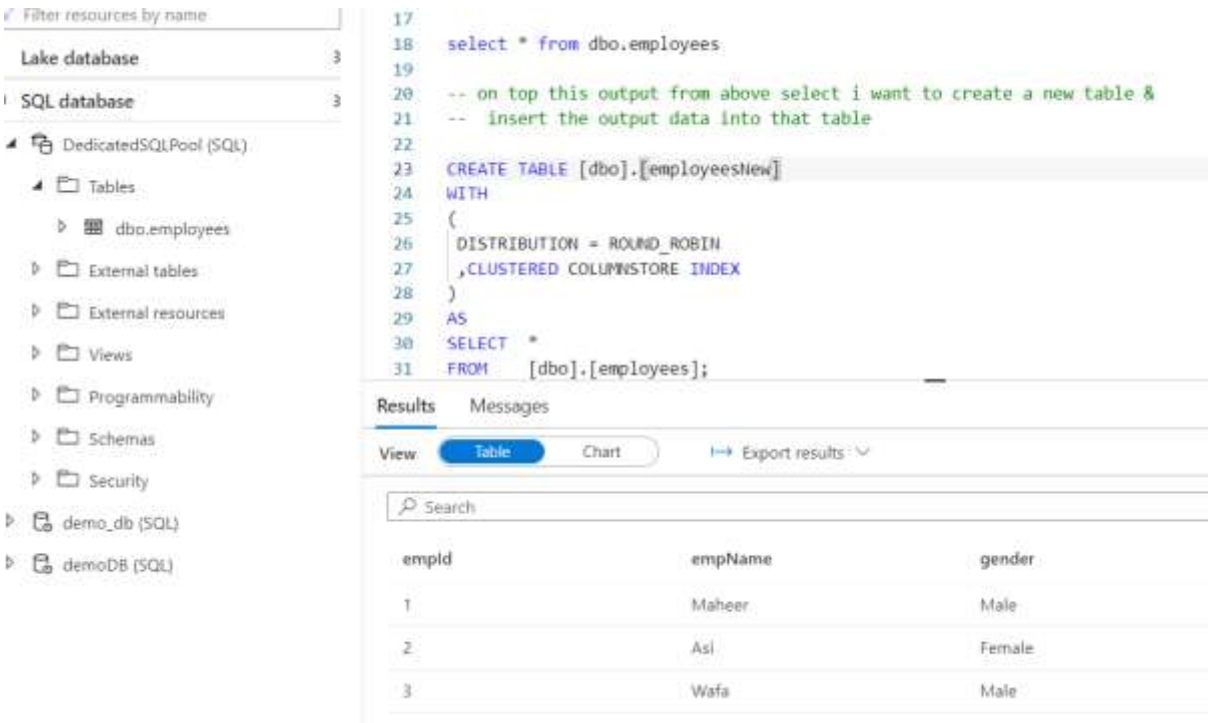
Initially use the master database to create demoDB, then run the rest of the lines with demoDB database



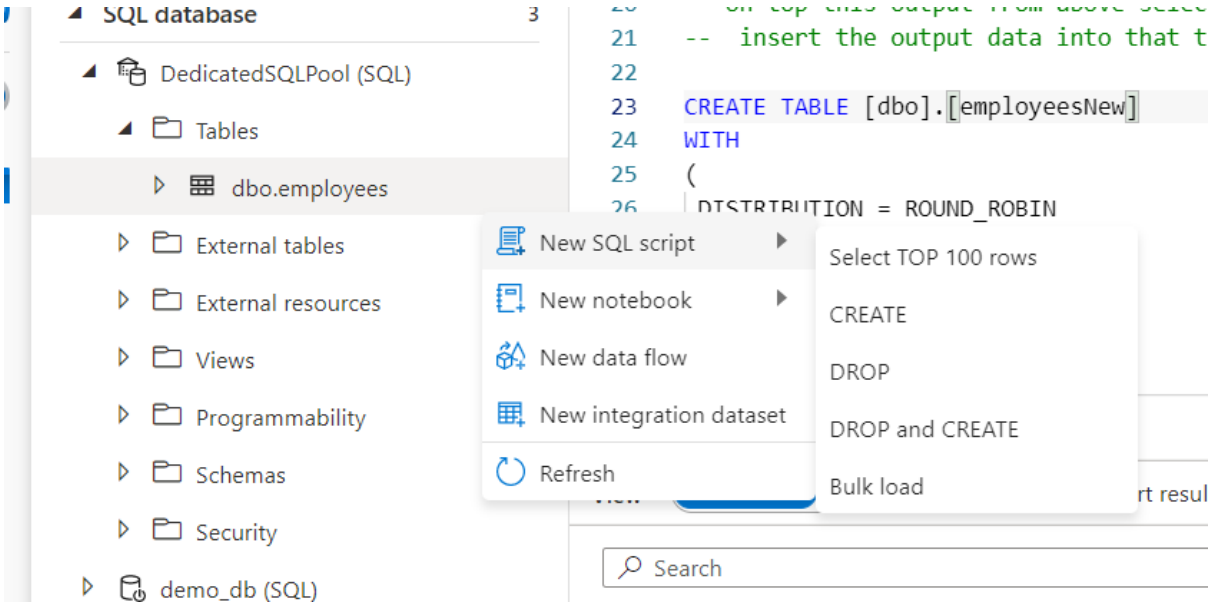
```
-- CREATE SCHEMA NYCTaxi

CREATE EXTERNAL TABLE NYCTaxi.PassengersCountStats
WITH (
    LOCATION = 'synapsedemo/NYCTaxi/Aggdata/',
    DATA_SOURCE = demoDataSource,
    FILE_FORMAT = ParquetFileFormat
)
AS
SELECT PassengerCount,
    SUM(TripDistanceMiles) AS SumTripDistance,
    AVG(TripDistanceMiles) AS AvgTripDistance
FROM
    OPENROWSET
    (
        BULK 'https://maheeradlsgen2.dfs.core.windows.net/synapsedemo/data/NYC
TripSmall.parquet',
        FORMAT='PARQUET'
    )
AS [rows]
    WHERE TripDistanceMiles > 0 AND PassengerCount > 0
    GROUP BY PassengerCount

GO

-- you can query the newly created external table
SELECT * FROM NYCTaxi.PassengersCountStats
```

17. CTAS with Synapse SQL in Azure Synapse Analytics

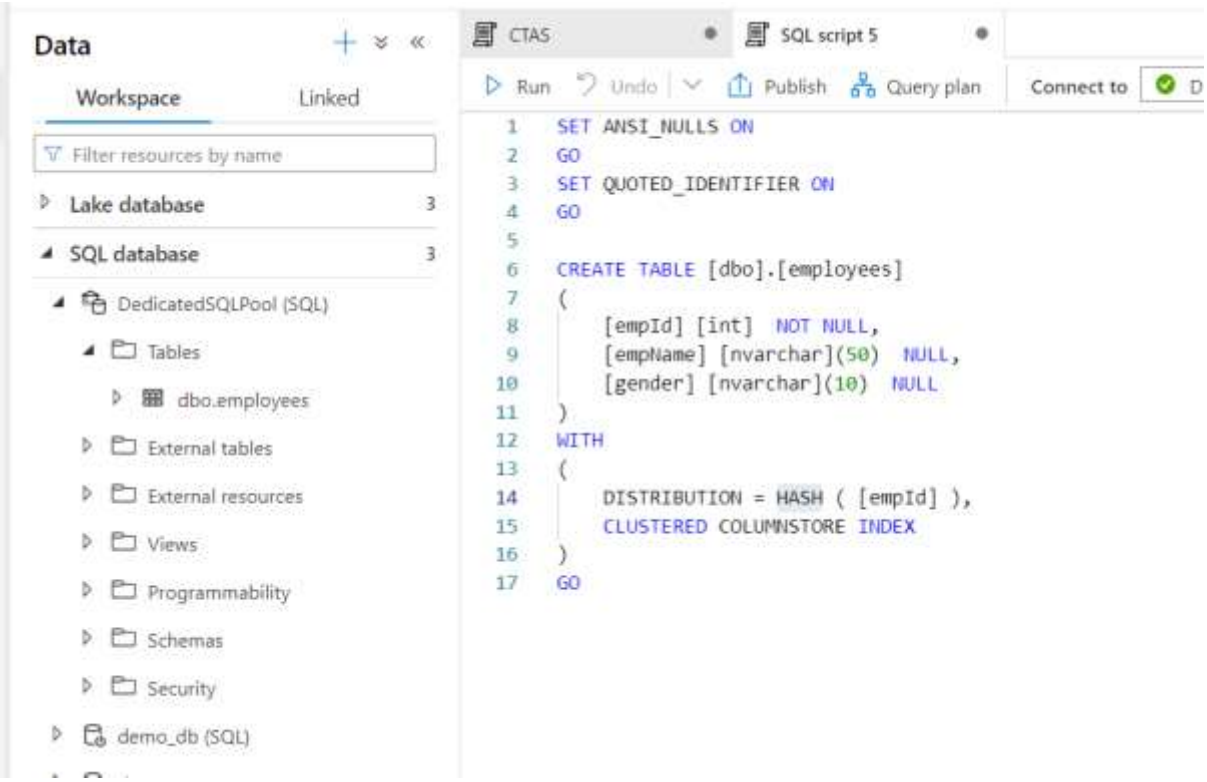
CTAS can be created only in dedicated table



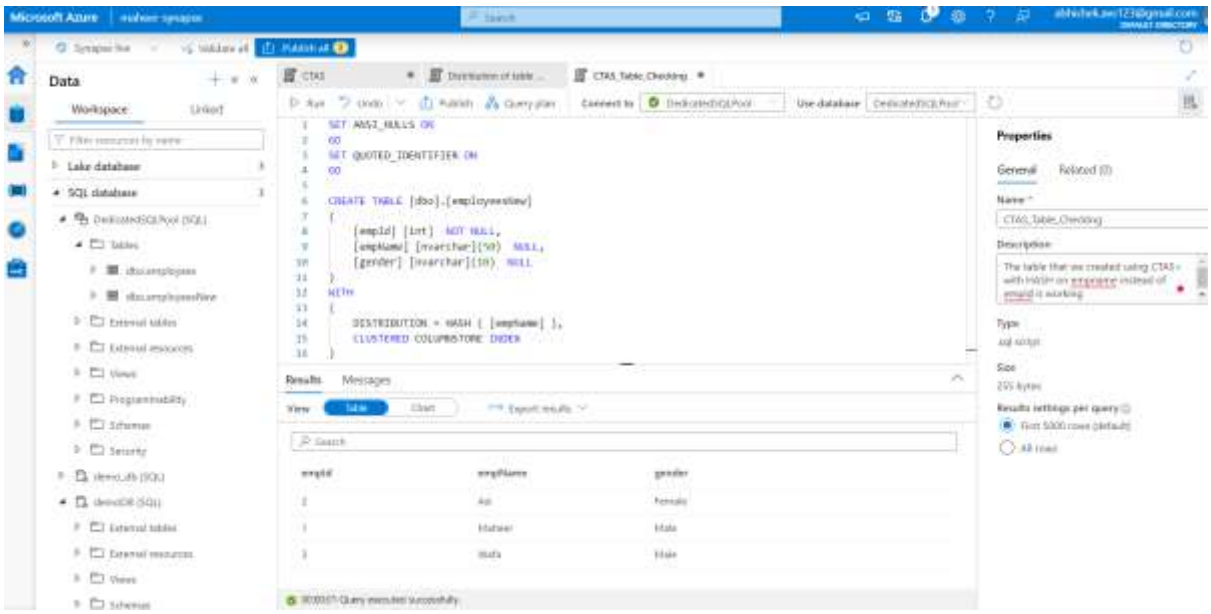
To see the distribution of the table, right click on dbo.employees and new script and click on CREATE



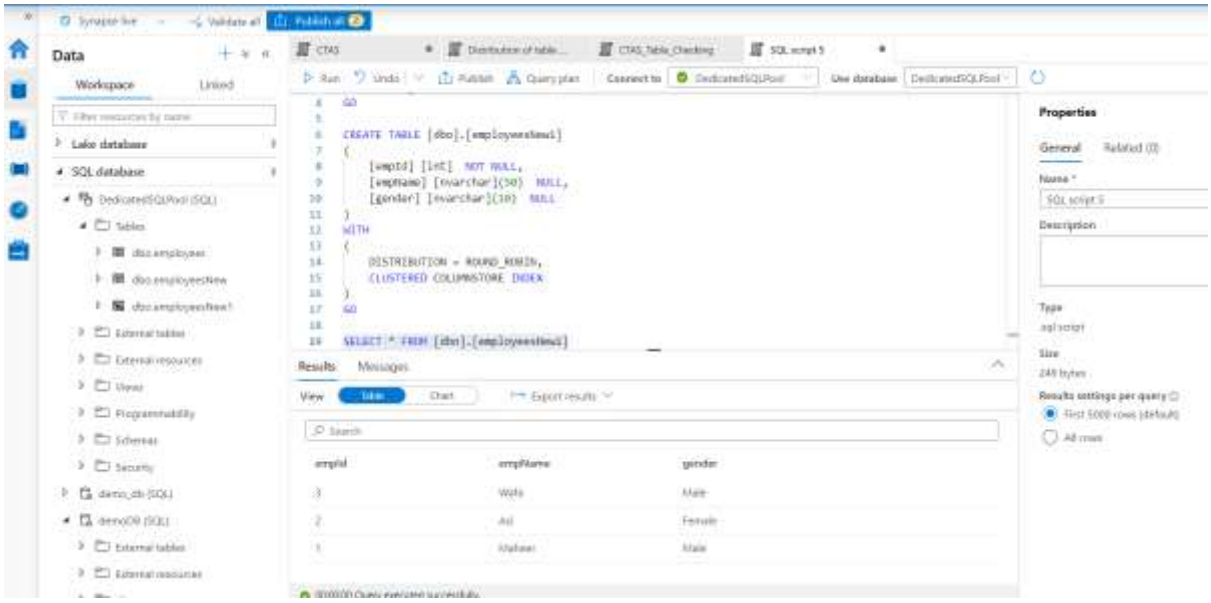
Type of distribution is shown here



Hash with emoname instead if emid



Round robin

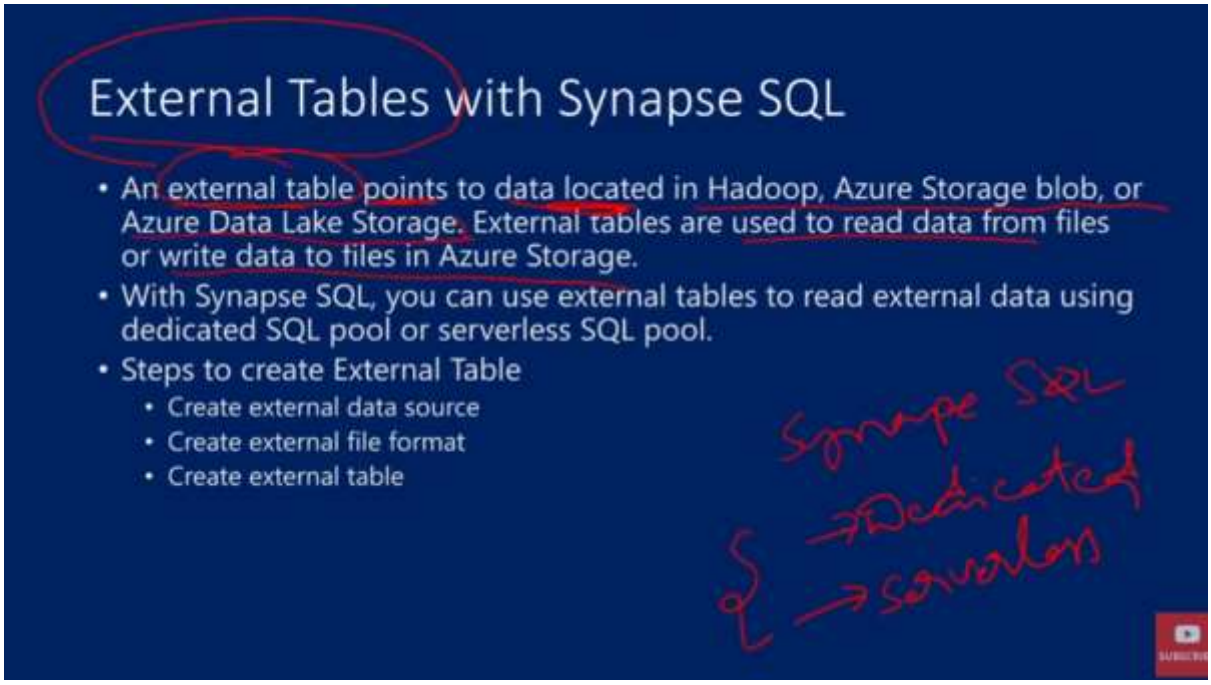


---

18. External Tables with Synapse SQL in Azure Synapse Analytics

CTAS only in Dedicated sql pool

But, External Table is in both




```sql
CREATE EXTERNAL DATA SOURCE maheeradlsgen2
WITH(
    LOCATION = 'abfss://synapsedemo@maheeradlsgen2.dfs.core.windows.net',
    CREDENTIAL = MSI_maheeradlsgen2,
    TYPE = HADOOP
);

-- We use ABFSS as we are creating data source
```

```
-
- 1st we create credential object, once we create it, we use the object in data source and create our external data s
ource
-- We define TYPE as HADOOP. It is datalake gen2 but y am I mentioning as Hadoop, u would be confused, right!
-
- Using this external data source, u will query the data inside the data lake gen 2 using the Hadoop technology. Hado
op -
> Java based technology which has a agility to query the data inside datalake storage gen 2 coz the data lake gen2 is
 built over Hadoop ecosystem only

DROP EXTERNAL DATA SOURCE maheeradlsgen2

CREATE DATABASE SCOPED CREDENTIAL MSI_maheeradlsgen2
WITH
    IDENTITY = 'Managed Identity'
GO

DROP DATABASE SCOPED CREDENTIAL MSI_maheeradlsgen2

CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Abhishek@2000'

CREATE EXTERNAL FILE FORMAT SynapseParquet
WITH (
    FORMAT_TYPE = PARQUET,
    DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
)

DROP EXTERNAL FILE FORMAT SynapseParquet

CREATE EXTERNAL TABLE dbo.NYCTaxi
    (
     [DateID] int,
     [MedallionID] int,
     [HackneyLicenseID] int,
     [PickupTimeID] int,
     [DropoffTimeID] int,
     [PickupGeographyID] int,
     [DropoffGeographyID] int,
     [PickupLatitude] float,
     [PickupLongitude] float,
     [PickupLatLong] nvarchar(4000),
     [DropoffLatitude] float,
     [DropoffLongitude] float,
     [DropoffLatLong] nvarchar(4000),
     [PassengerCount] int,
     [TripDurationSeconds] int,
     [TripDistanceMiles] float,
     [PaymentType] nvarchar(4000),
     [FareAmount] numeric(19,4),
     [SurchargeAmount] numeric(19,4),
     [TaxAmount] numeric(19,4),
     [TipAmount] numeric(19,4),
     [TollsAmount] numeric(19,4),
     [TotalAmount] numeric(19,4)
    )
WITH
    (
        LOCATION = '/data/NYCTripSmall.parquet',
        DATA_SOURCE = maheeradlsgen2,
        FILE_FORMAT = SynapseParquet
    )

SELECT top 100 * FROM dbo.NYCTaxi
```

We use ABFSS as we are creating data source

1st we create credential object, once we create it, we use the object in data source and create our external data source

We define TYPE as HADOOP. It is datalake gen2 but y am I mentioning as Hadoop, u would be confused, right!

Using this external data source, u will query the data inside the data lake gen 2 using the Hadoop technology. Hadoop -> Java based technology which has a agility to query the data inside datalake storage gen 2 coz the data lake gen2 is built over Hadoop ecosystem only

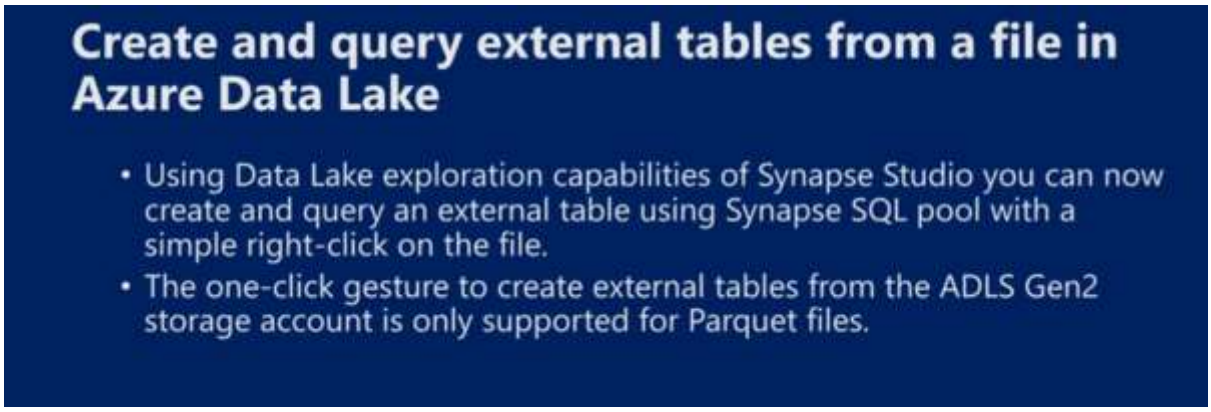
USES of External Table

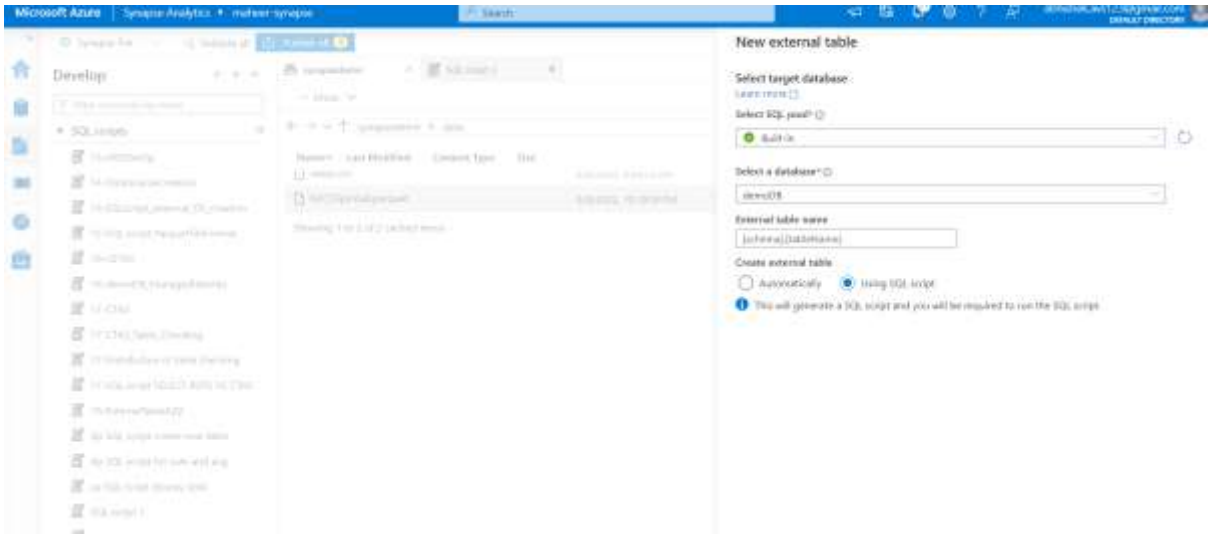
Using CETAS the select query output is stored in the ADLS

Like we can just import data of top 100 rows in table etc

```
-- We create a CTAS using the output of above select query
CREATE TABLE dbo.NYCPhysicalTable
WITH
(
 DISTRIBUTION = HASH(DateID),
 CLUSTERED COLUMNSTORE INDEX
)
AS
SELECT top 100 * FROM dbo.NYCTaxi

SELECT top 100 * FROM dbo.NYCPhysicalTable
```

19. Create and query external tables from a file in ADLS in Azure Synapse Analytics





For csv file it don’t work

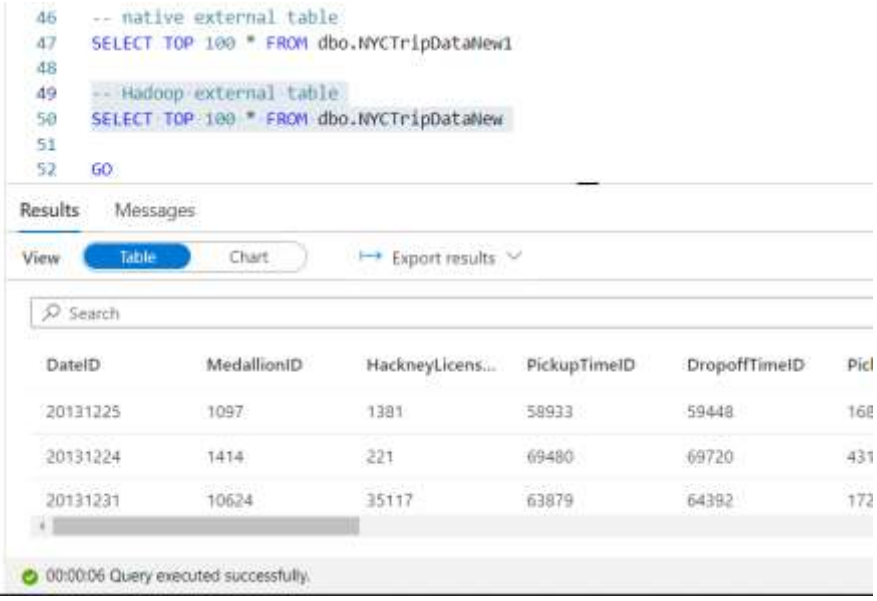
20. Types of External Tables (Hadoop & Native) in Synapse SQL in Azure Synapse Analytics

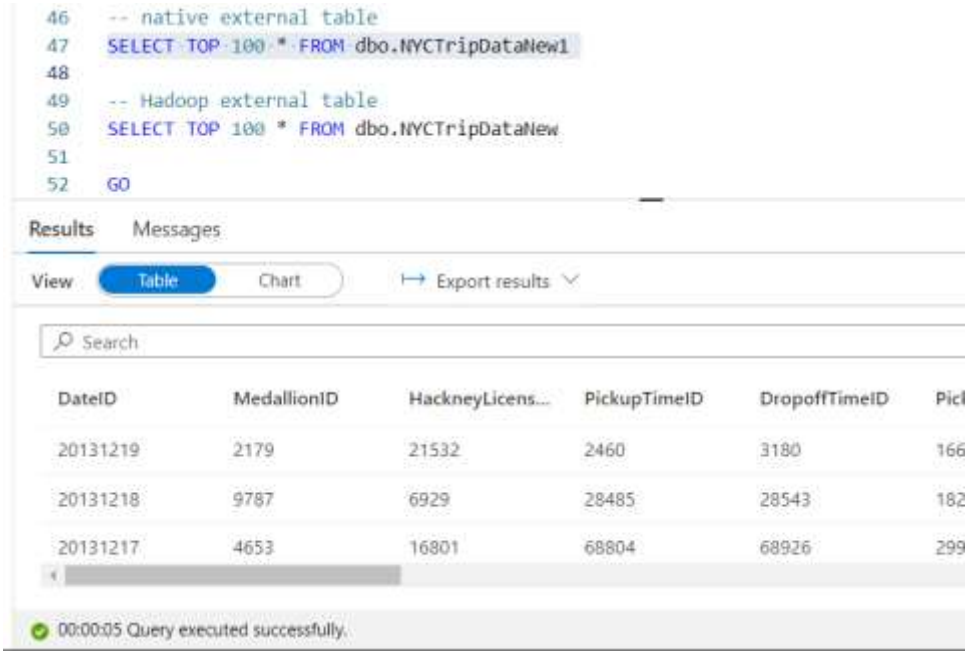
Types of External Tables

- Depending on the type of the external data source, you can use two types of external tables:
  - Hadoop external tables
  - Native external tables

External table type	Hadoop	Native
Dedicated SQL pool	Available	Parquet tables are available in gated preview
Serverless SQL pool	Not available	Available
Supported formats	Delimited/CSV, Parquet, ORC, Hive RC, and RC	Serverless SQL pool: Delimited/CSV, Parquet, and Delta Lake(preview) Dedicated SQL pool: Parquet

- Hadoop External Tables - TYPE = HADOOP
  - It is an external table that uses JAVA technology to read the data from external files
  - It is available in dedicated sql pool, not available in serverless
  - 

- 

- Native External Tables - if u don't mention TYPE = HADOOP
  - Better than Hadoop ones in performance
  - Azure native technology is in C++ technology for reading the data from external storage
  - 

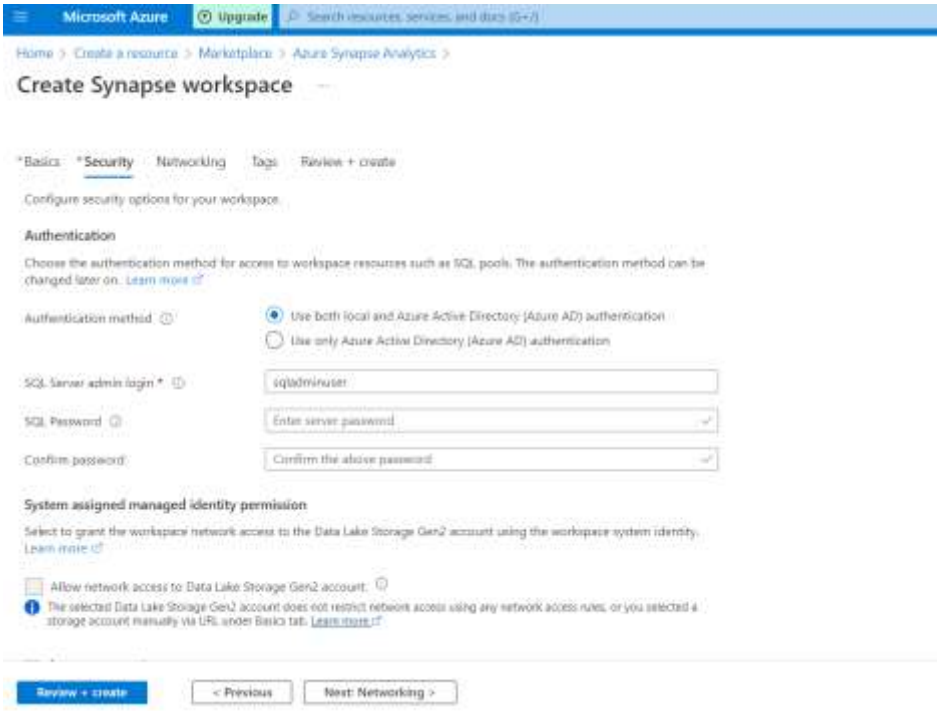
---

21. Administrative accounts in Synapse SQL in Azure Synapse Analytics

Maheer-synapse work space > properties> there are 2 things in it – sql admin nusername and SQL Active Directory Admin

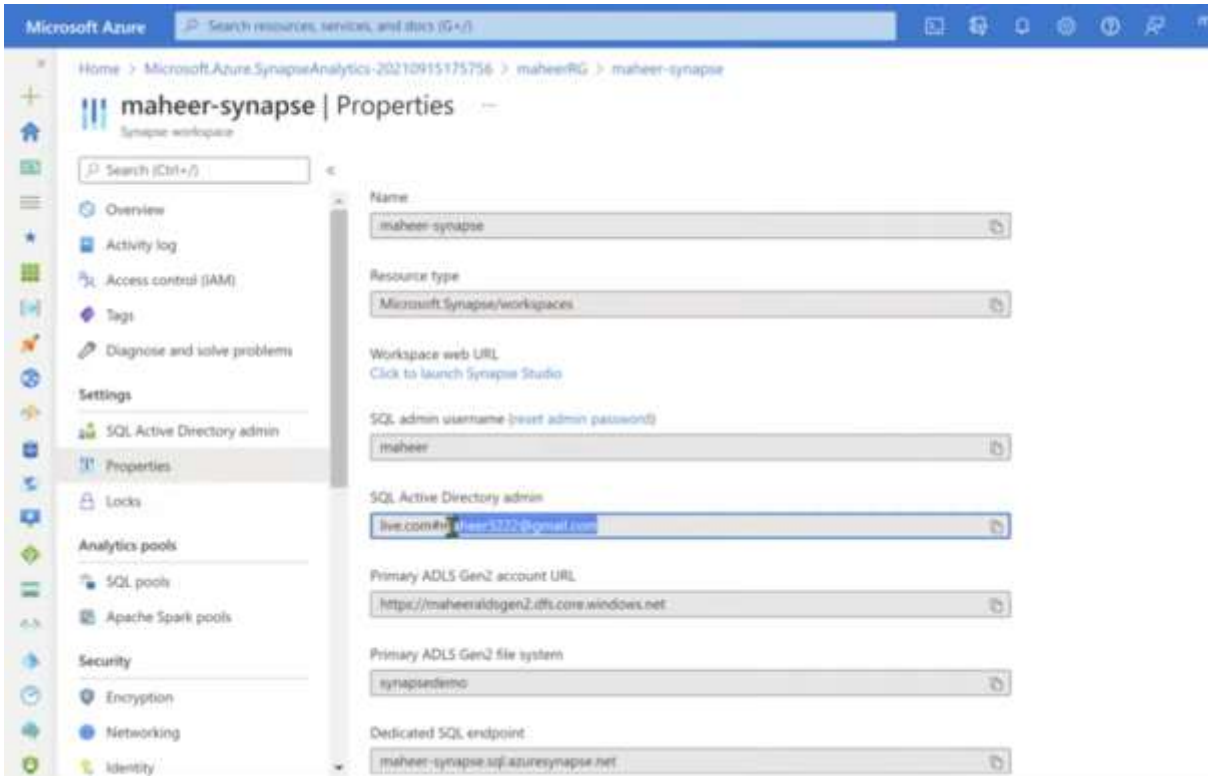
- Server Admin –
  - u can give any name in this
  - created automatically

- uses SQL Authentication, can change by clicking on reset admin password etc

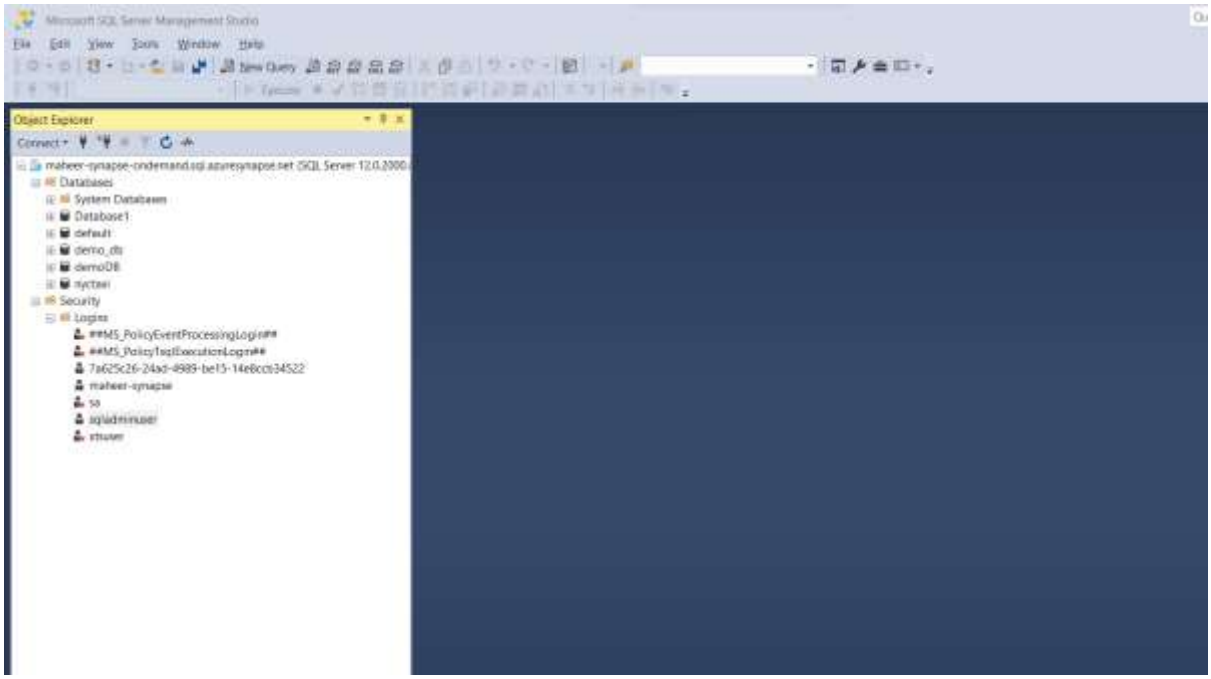


- Active Directory Admin –
  - ur own id will be added to the active directory admin
  - u can change it in settings> sql active directory admin>SET ADMIN> enter the search box
  - u can also create sql directory admin group also

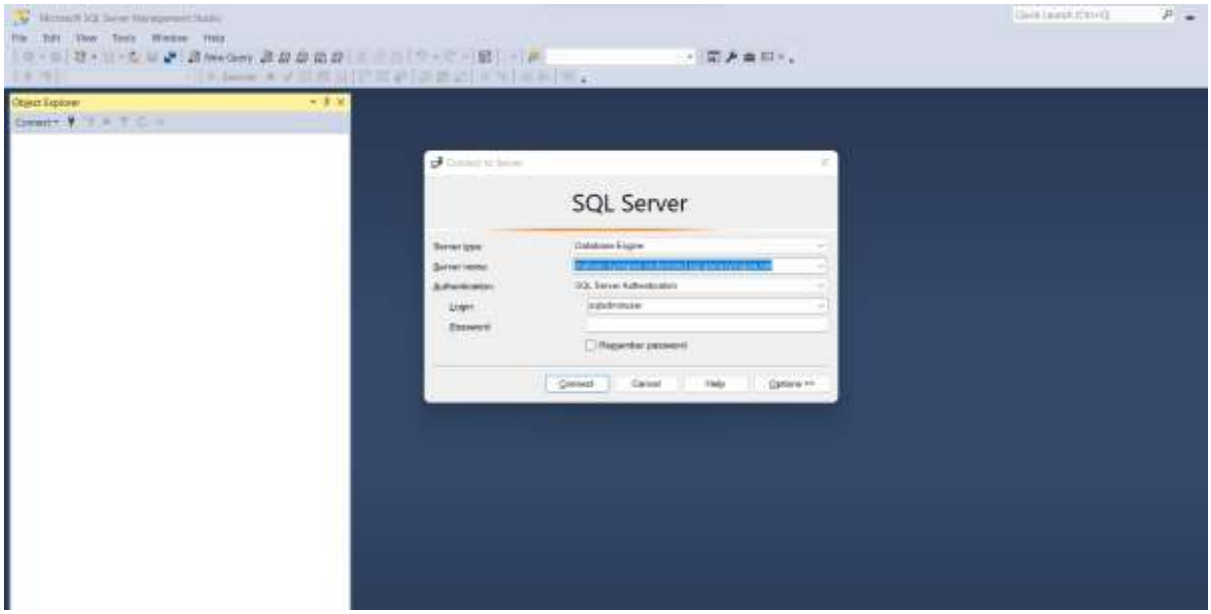
These can be used to connect to any database i.e. serveless and dedicated sql pool an du can do anything u want.

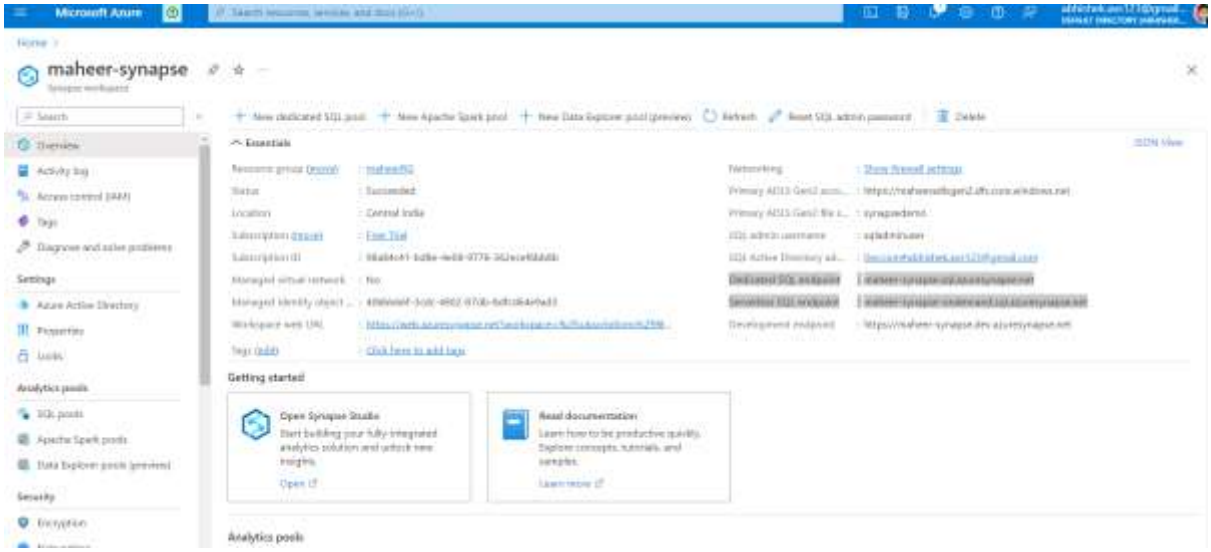


SQL Server Management Studio (SSMS)



2nd icon beside connect in this pic, u can click on it and disconnect


Object Explorer
SQL Server
Connect
Cancel
Help
Options >>


Microsoft Azure
maheer-synapse
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Azure Active Directory
Properties
Locks
Analytics pools
SQL pools
Apache Spark pools
Data Explorer pools (preview)
Security
Encryption
New dedicated SQL pool
New Apache Spark pool
New Data Explorer pool (preview)
Refresh
Reset SQL admin password
Delete
Essentials
JSON View
Networking
Show firewall settings
Central India
Free Trial
Getting started
Open Synapse Studio
Start building your fully integrated analytics solution and unlock new insights.
Open
Read documentation
Learn how to be productive quickly. Explore concepts, tutorials, and samples.
Learn more
Analytics pools

Server admin

- When you create an Azure Synapse Analytics, you must name a **Server admin login**. SQL server creates that account as a login in the master database. This account connects using SQL Server authentication (username and password).

Azure Active Directory admin

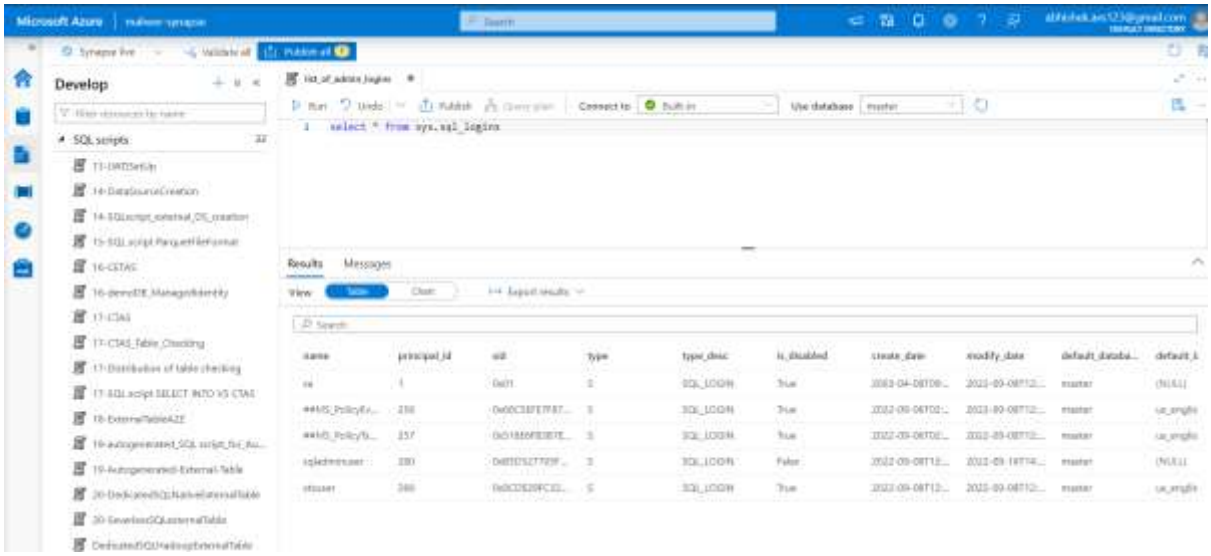
- One Azure Active Directory account, either an individual or security group account, can also be configured as an administrator.
- An Azure AD administrator **must** be configured if you want to use Azure AD accounts to connect to Synapse SQL

Characteristics of Admin accounts

- Are the only accounts that can automatically connect to any SQL Database on the server.
- Can create, alter, and drop databases, logins, users in master, and server-level IP firewall rules.
- Can add and remove members to the **dbmanager** and **loginmanager** roles.
- Can view the **sys.sql_logins** system table.

- Can view the **sys.sql_logins** system table.

list of all logins



---

22. Create Login and User for Server less SQL Pool in Azure Synapse Analytics

To get into the sever, we need login credentials, then to access certain DB, we need a certain username access for each database

Serverless SQL Pool

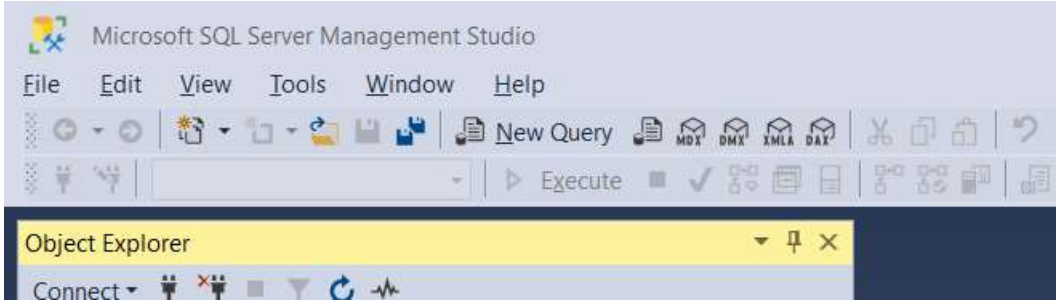
- To create a login to serverless SQL pool, use the following syntax:

```
CREATE LOGIN Mary WITH PASSWORD = '<strong_password>';
-- or
CREATE LOGIN [Mary@domainname.net] FROM EXTERNAL PROVIDER;
```

- Once the login exists, you can create users in the individual databases within the serverless SQL pool endpoint and grant required permissions to these users. To create a use, you can use the following syntax:

```
CREATE USER Mary FROM LOGIN Mary;
-- or
CREATE USER Mary FROM LOGIN Mary@domainname.net;
-- or
CREATE USER [mike@contoso.com] FROM EXTERNAL PROVIDER;
```

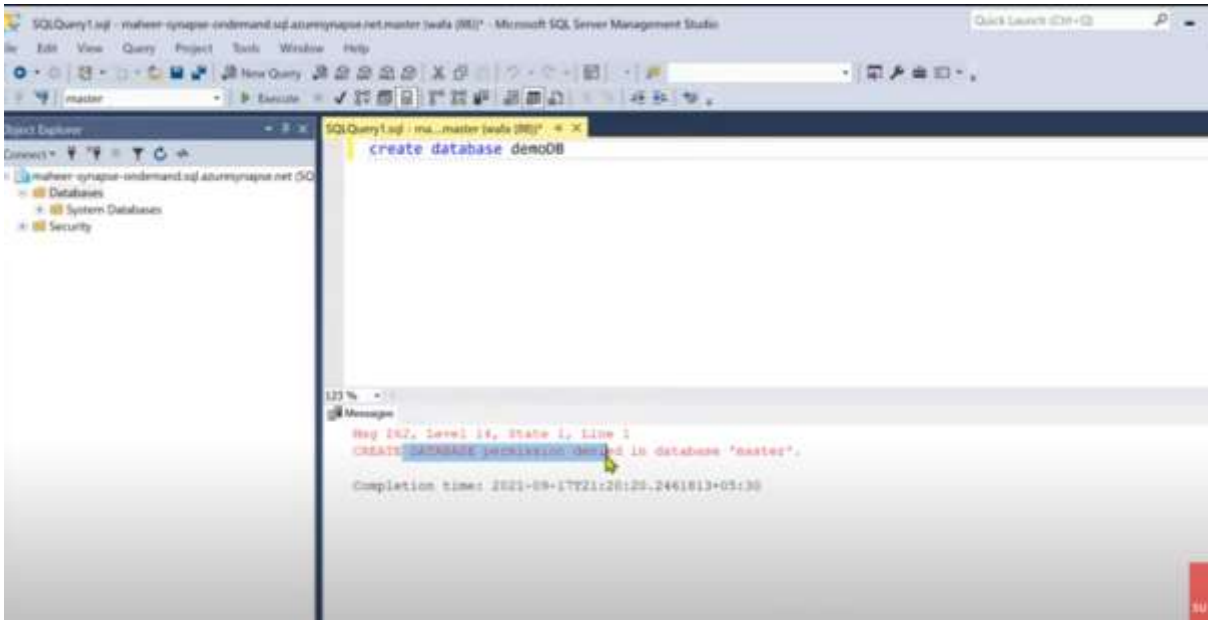
LOGINS are created at server level and USER are created at database level



Click on new query window to write the query

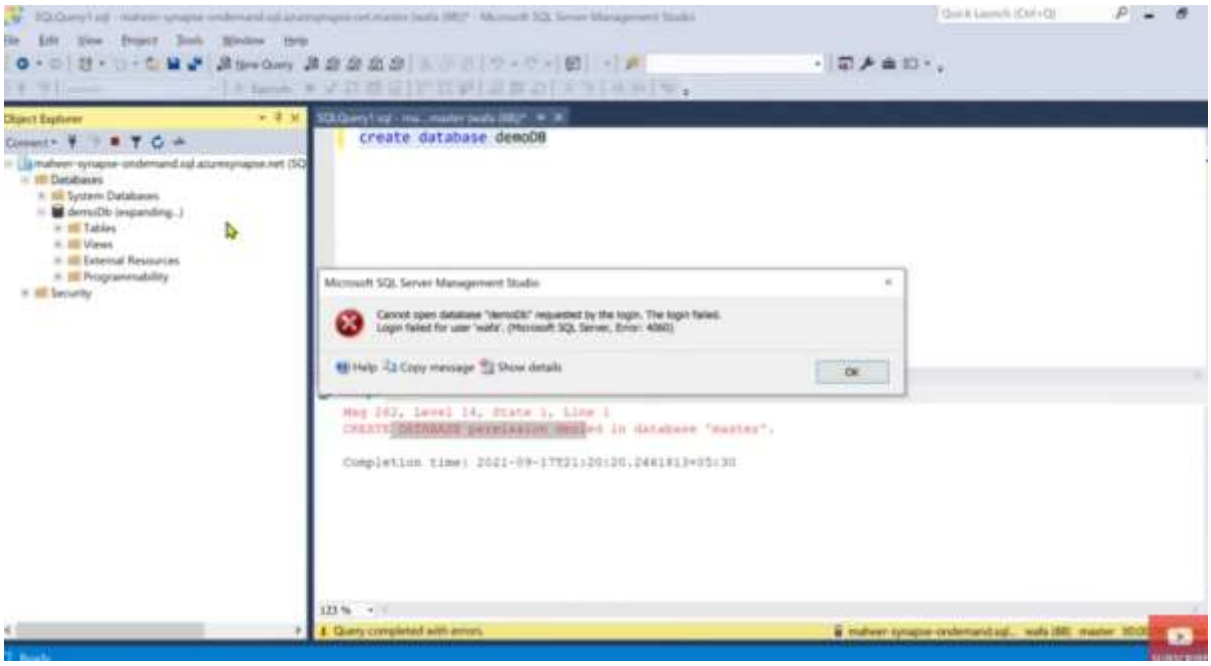
We can login through the new login id that we created

```
create login abhi with password = 'Welcome@1123'
```

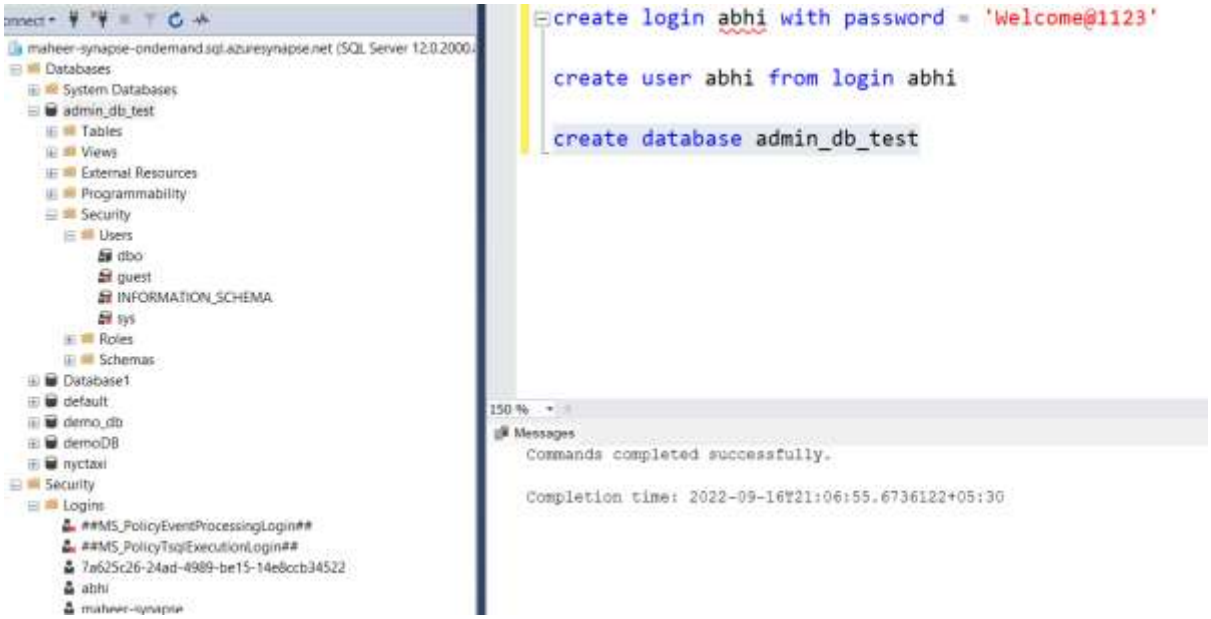


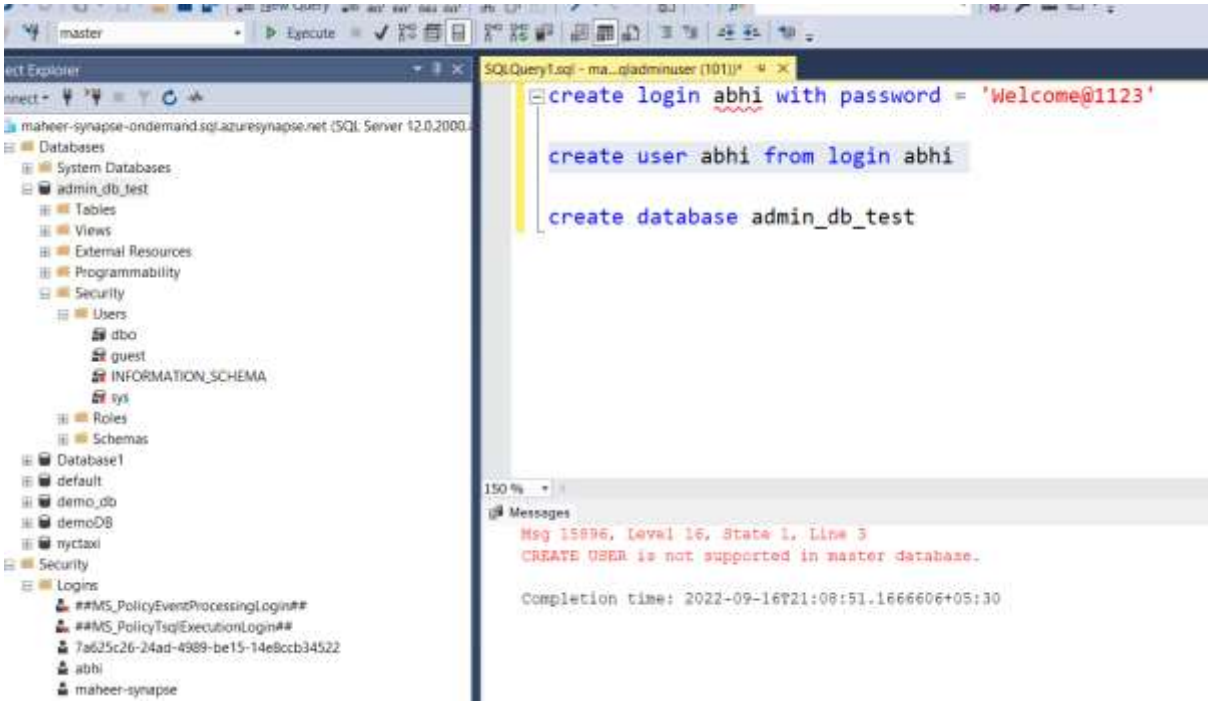
It is denied coz it don't have permission to do that

We only have login, dv access etc not given yet

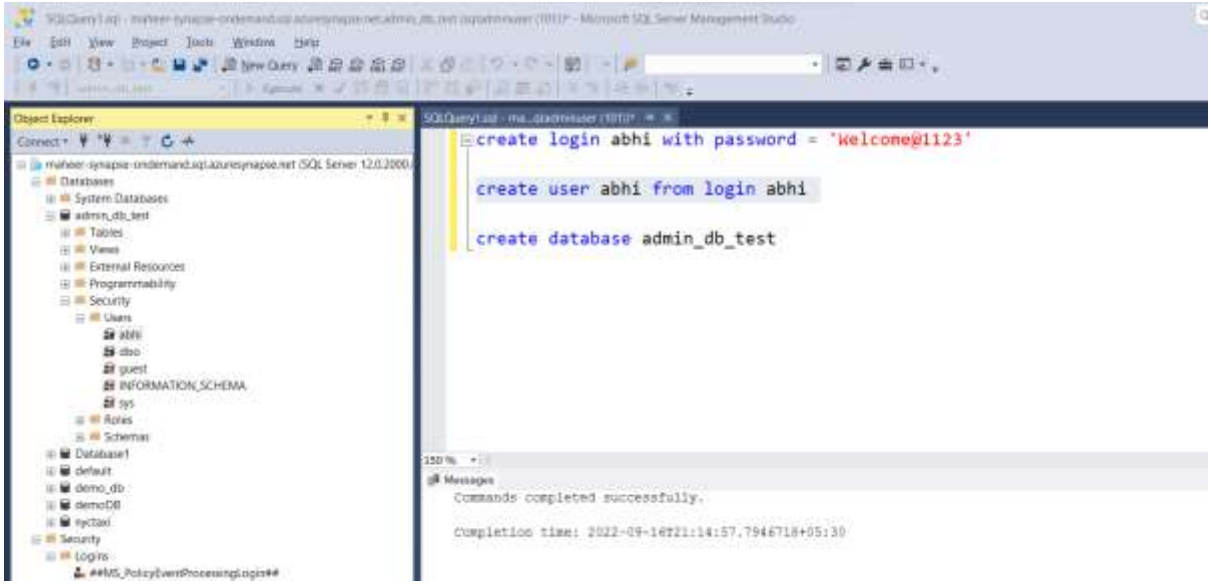


We create a database

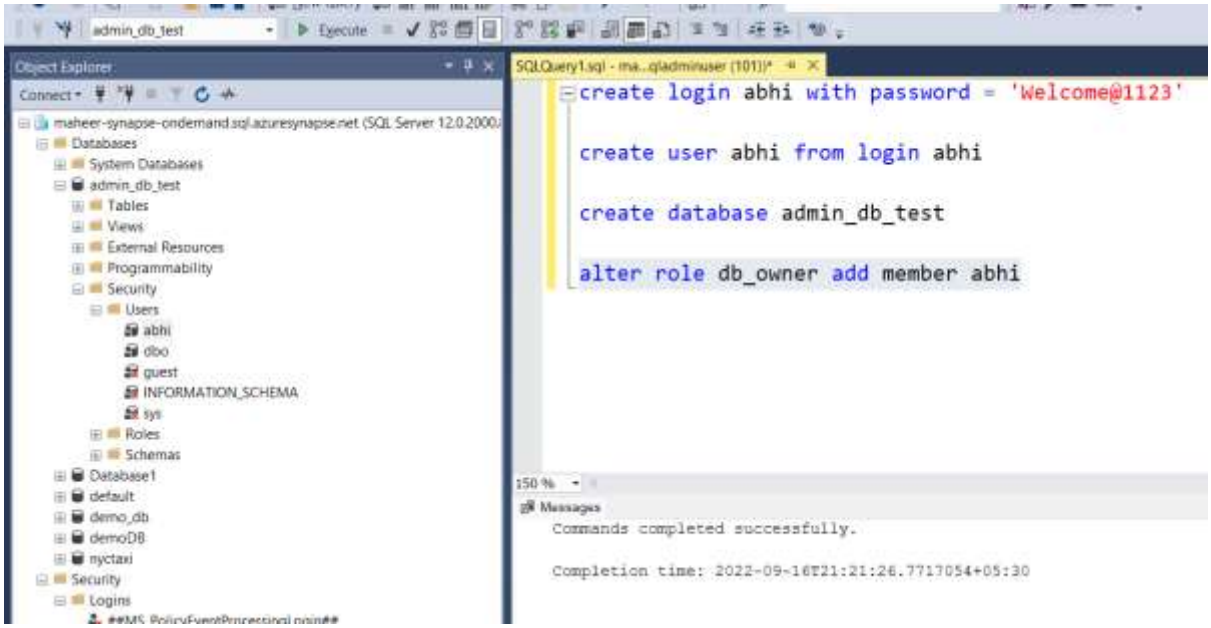




We need to go to the database in which we want to create the user, then we can run that anad the user abhi will be created



After creating the user, we need to also grant the permission etc




```
create login abhi with password = 'Welcome@1123'

create user abhi from login abhi

create database admin_db_test

alter role db_owner add member abhi
```

ALL this will work with sql authentication

In azure active directory admin case

For them the syntax is this

```
-- or
CREATE LOGIN [Mary@domainname.net] FROM EXTERNAL PROVIDER;
```

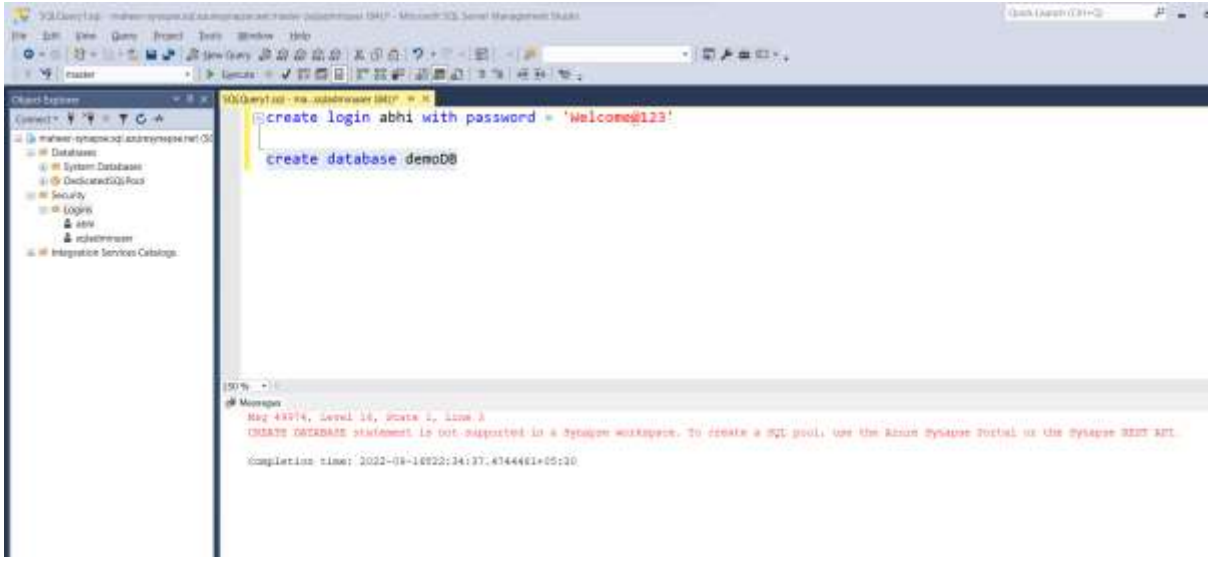
```
CREATE USER Mary FROM LOGIN Mary@domainname.net;
-- or
CREATE USER [mike@contoso.com] FROM EXTERNAL PROVIDER;
```

Instead of Mary, use ur login

---

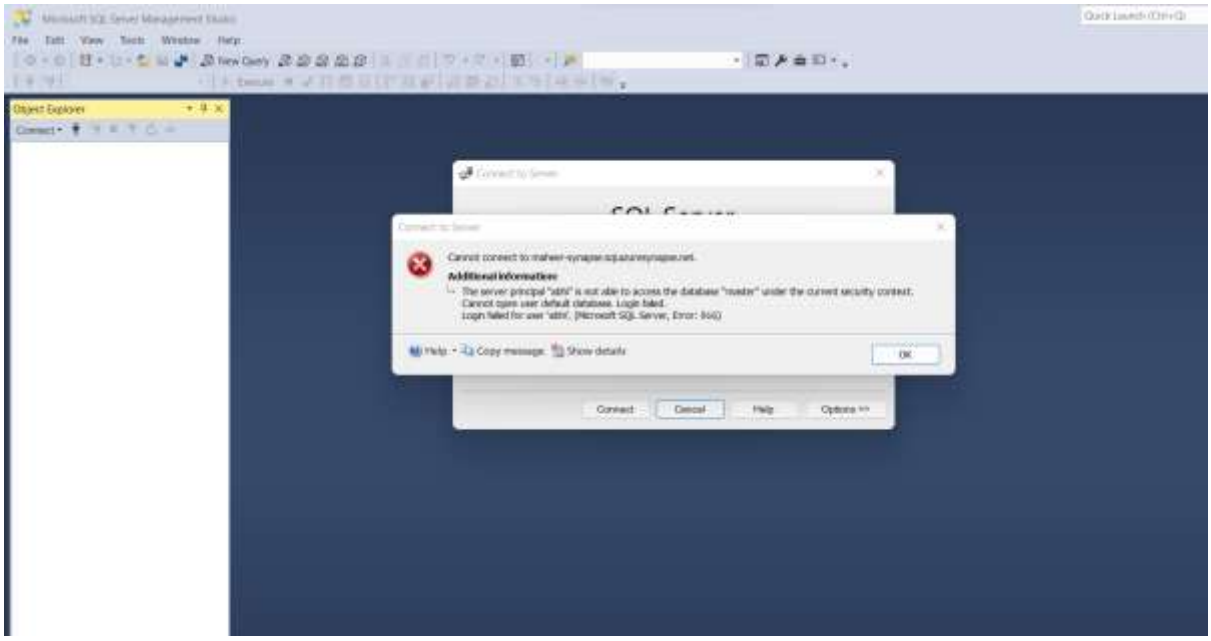
23. Create Login and User for Dedicated SQL Pool in Azure Synapse Analytics

Contained DB user – where u wont create login, only user will be there

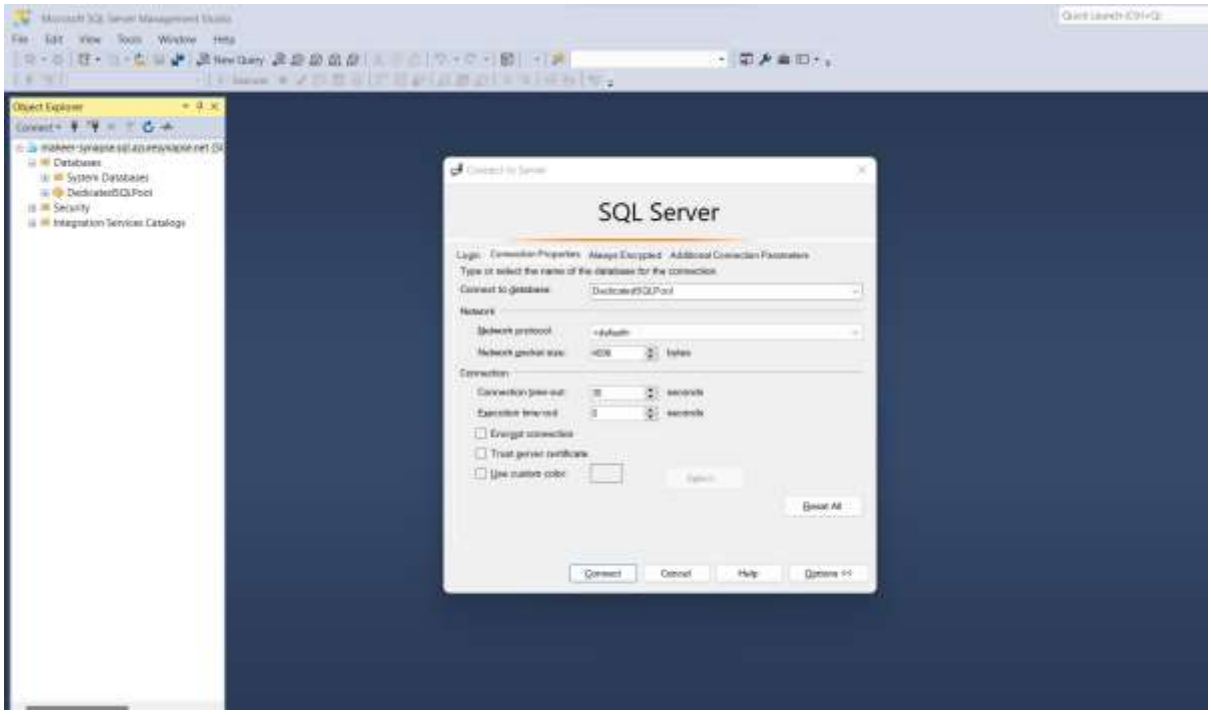


We cant create database from here in case of dedicated sql pool

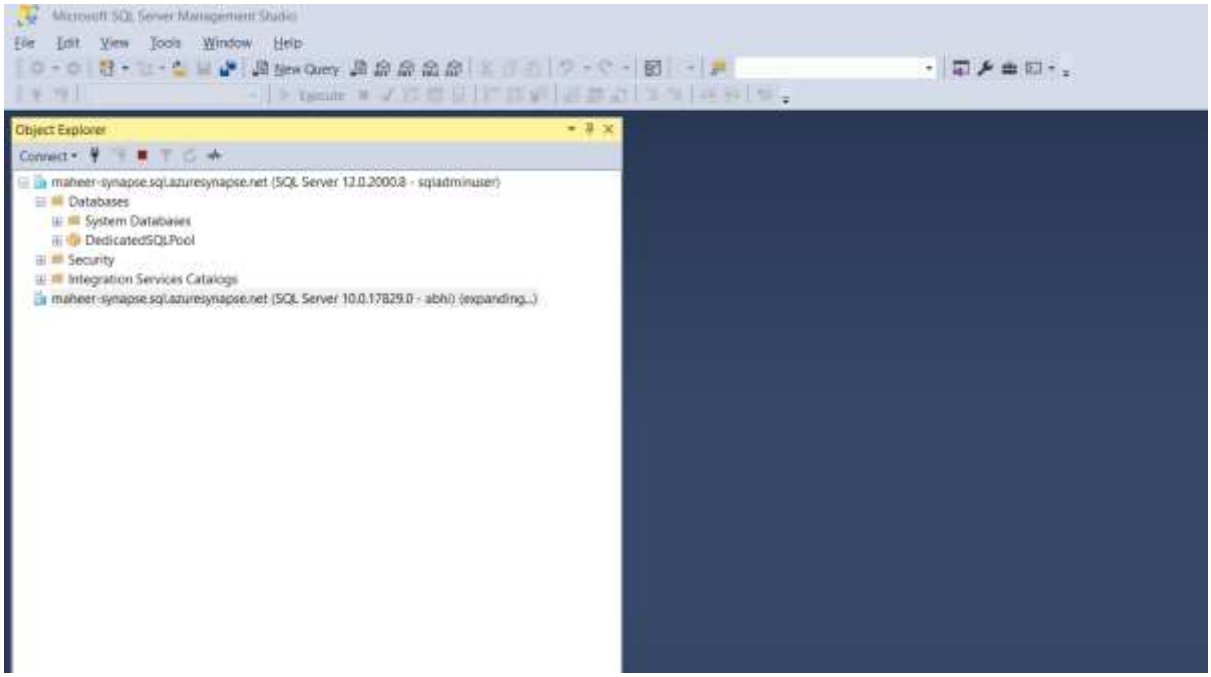
After making the login Still error comes, coz the login is inside the demoDB that we created, not in master



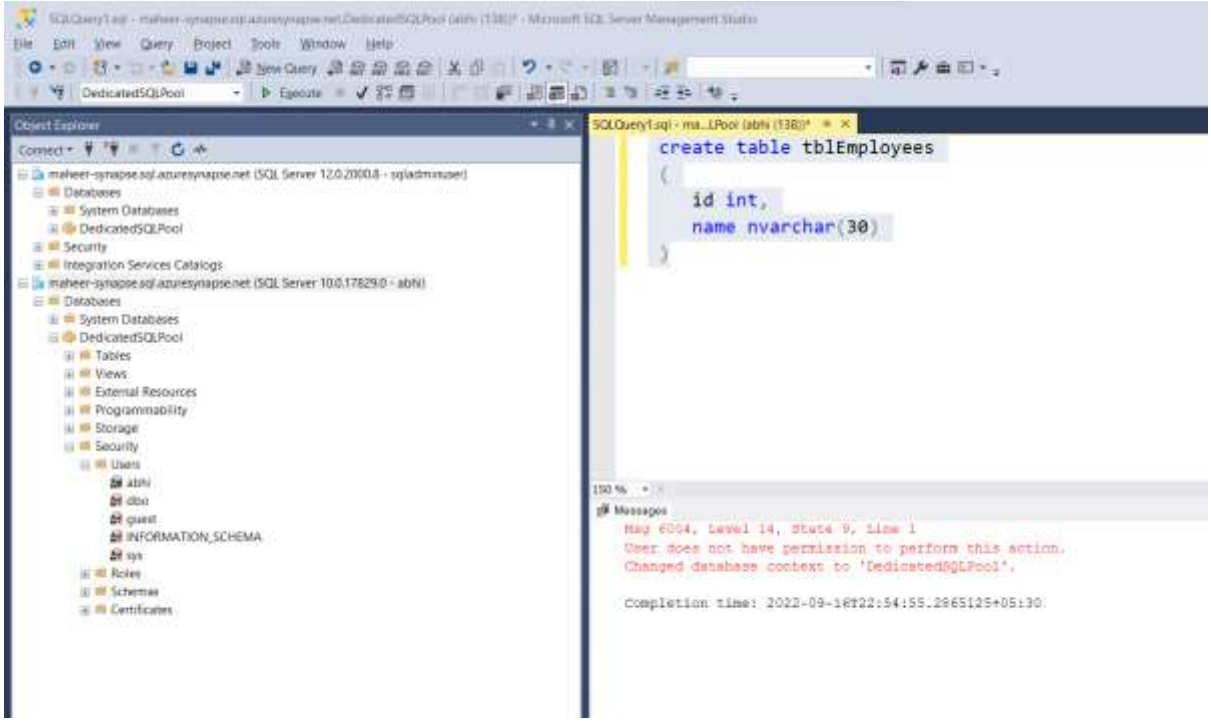
SO, we need to go the options> then click on connect to database, it will take a moment to search(b4 that try to connect with sqladmin then try going to options)



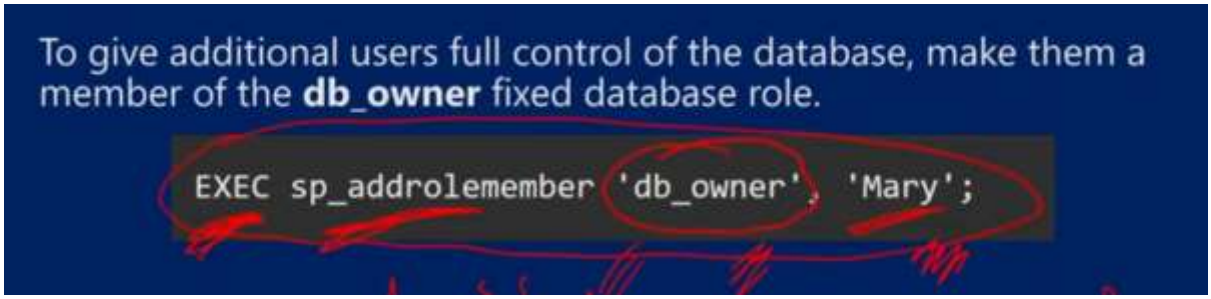

```
create login abhi with password = 'Welcome@123'

-- create database demoDB

create user abhi from login abhi

exec sp_addrolemember 'db_owner','abhi'
```



After connecting, we try to create table from abhi login, but error comes

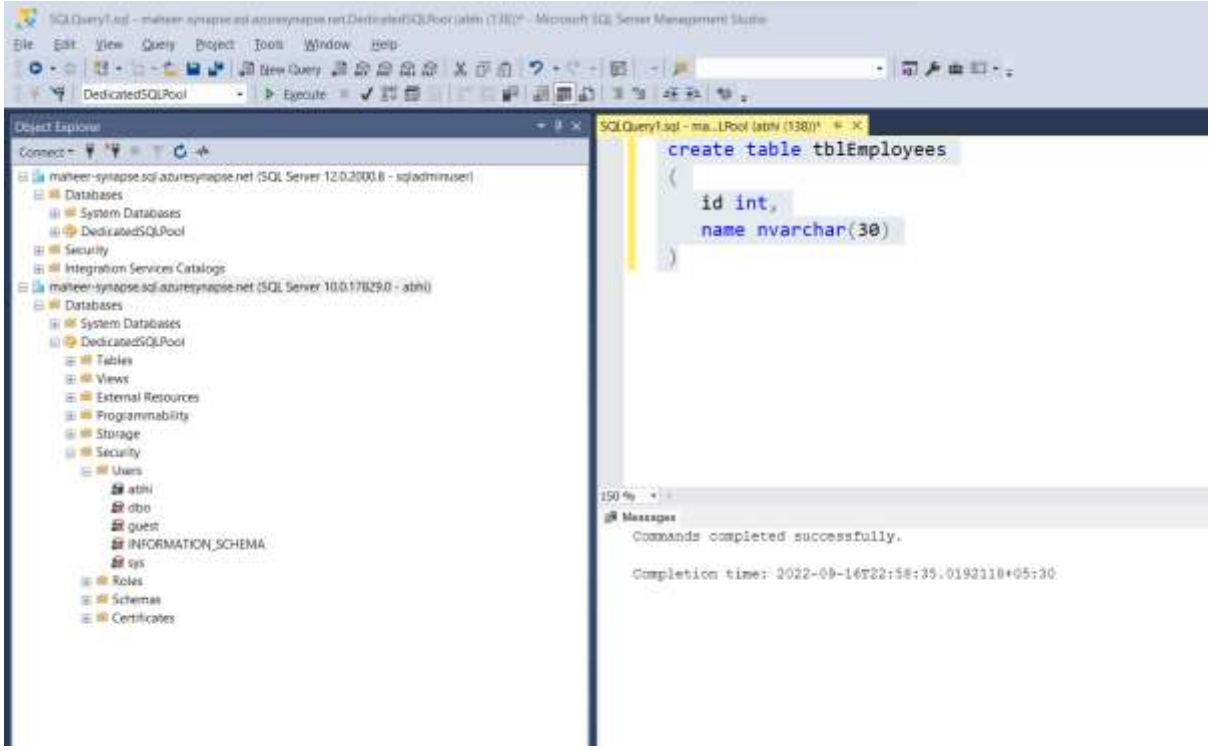


We add the role now




```
exec sp_addrolemember 'db_owner','abhi'
```

after running this from sql admin login, we can now create table in abhi login



Alter permission for other users also

Serverless SQL

To authorize additional users to create new users, grant that selected user the ALTER ANY USER permission, by using a statement such as:

```
GRANT ALTER ANY USER TO Mary;
```

To give additional users full control of the database, make them a member of the **db_owner** fixed database role.

```
EXEC sp_addrolemember 'db_owner', 'Mary';
```

---

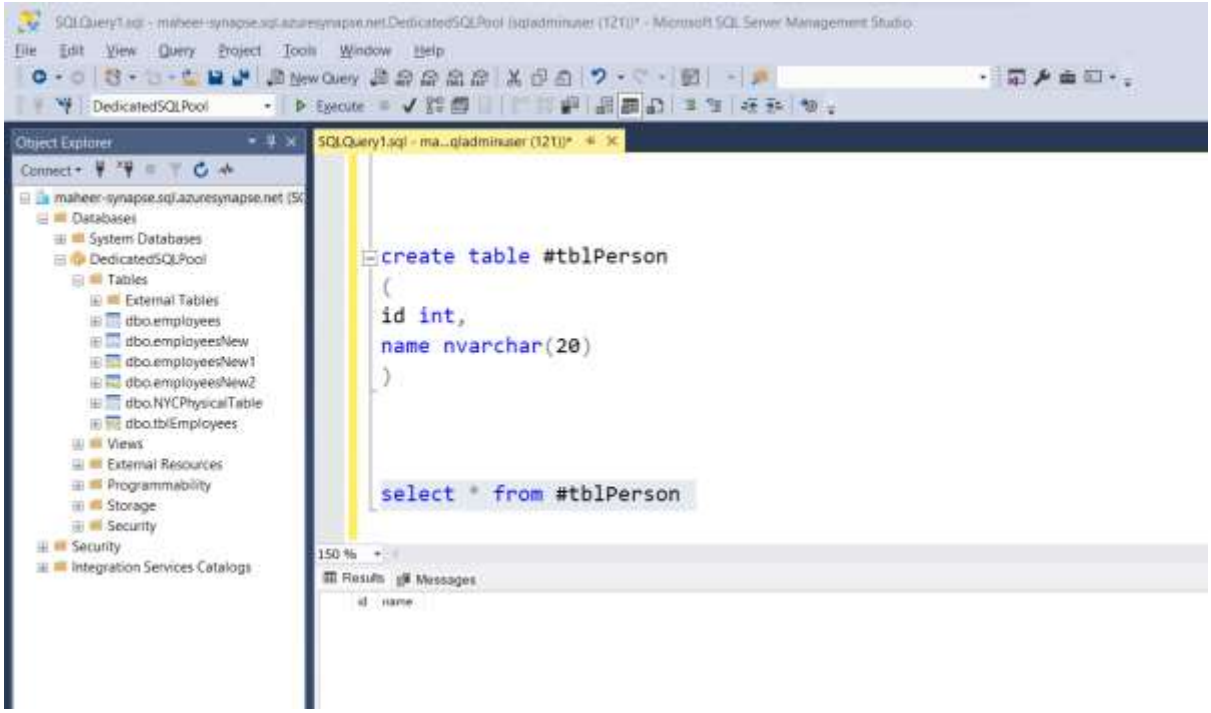
24. Temporary Tables in Synapse SQL in Azure Synapse Analytics

Temporary Tables

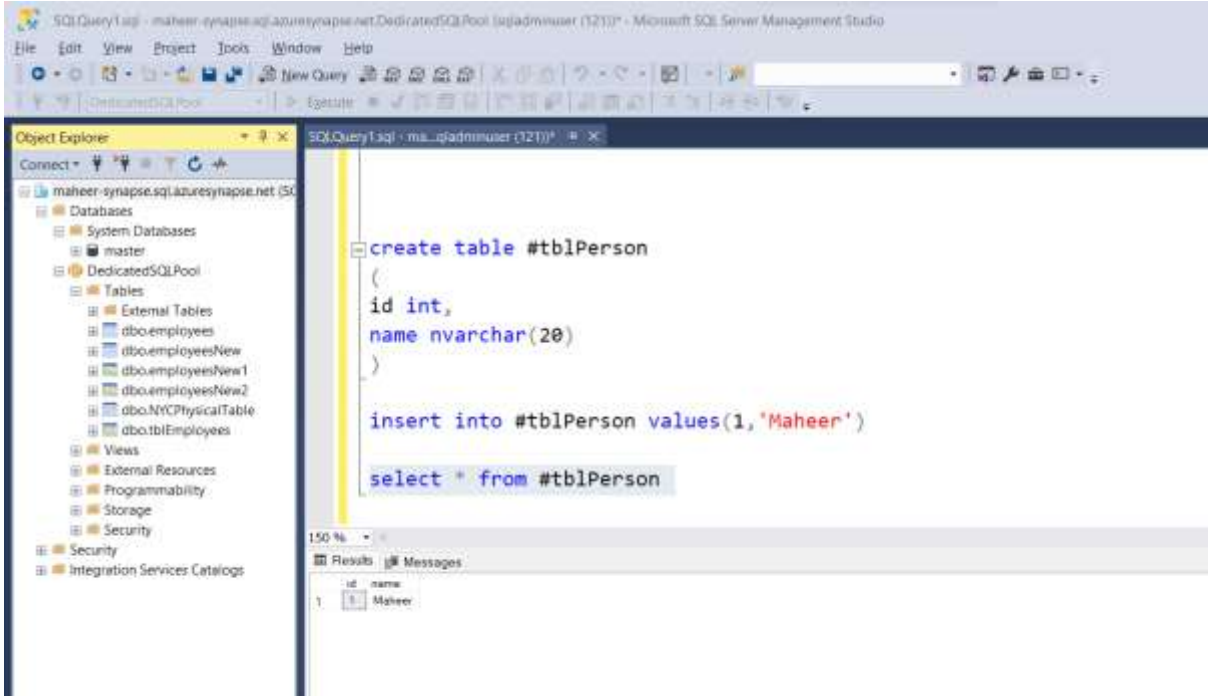
- Temporary tables are useful when processing data, especially during transformation where the intermediate results are transient.
  With Synapse SQL, temporary tables exist at the session level. They're only visible to the session in which they were created.
  As such, they're automatically dropped when that session logs off.

- Temporary tables are created by prefixing your table name with # symbol.

```
CREATE TABLE #stats_ddl
(
    [schema_name]         NVARCHAR(128) NOT NULL
,    [table_name]             NVARCHAR(128) NOT NULL
,    [stats_name]             NVARCHAR(128) NOT NULL
,    [stats_is_filtered]      BIT           NOT NULL
,    [seq_nmbr]               BIGINT        NOT NULL
,    [two_part_name]          NVARCHAR(260) NOT NULL
,    [three_part_name]        NVARCHAR(400) NOT NULL
)
WITH
(
    DISTRIBUTION = HASH([seq_nmbr])
,    HEAP
)
```



In dedicated sql pool, the temporary table doesn't show up



Within the session only the temporary will be available

Session -> new query window opening

Temp table with CTAS

Temp tables with CTAS

- Temporary tables can also be created with a CTAS

```
CREATE TABLE #stats_ddl
WITH
(
    DISTRIBUTION = HASH([seq_nmbr])
,    HEAP
)
AS
(
SELECT   *
FROM     [dbo].[FactInternetSales];
```

Drop temptable

DROP temp tables

- When a new session is created, no temporary tables should exist. However, if you're calling the same stored procedure that creates a temporary with the same name, to ensure that your CREATE TABLE statements are successful, use a simple pre-existence check with DROP:

```
IF OBJECT_ID('tempdb..#stats_ddl') IS NOT NULL
BEGIN
    DROP TABLE #stats_ddl
END
```

Limitation of temp tables

Limitation of Temp Tables

- Dedicated SQL pool does have a few implementation limitations for temporary tables:
  - Only session scoped temporary tables are supported. Global Temporary Tables aren't supported.
  - Views can't be created on temporary tables.
  - Temporary tables can only be created with hash or round robin distribution. Replicated temporary table distribution isn't supported.
- Temporary tables in serverless SQL pool are supported but their usage is limited. They can't be used in queries which target files.
- For example, you can't join a temporary table with data from files in storage. The number of temporary tables is limited to 100, and their total size is limited to 100MB.

In serverless sql pool we cant take the data from the files to make the temp table

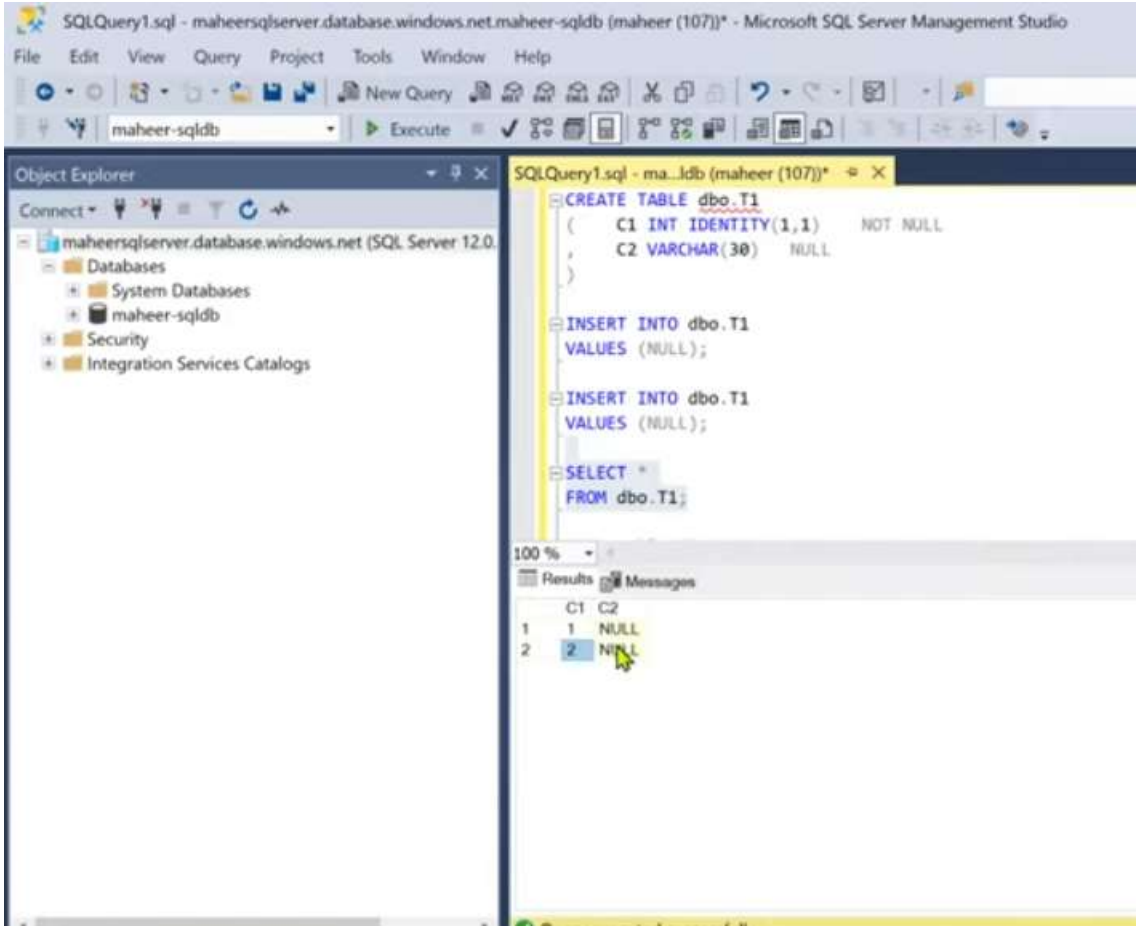
---

25. Using IDENTITY to create surrogate keys using dedicated SQL pool in Azure Synapse Analytics

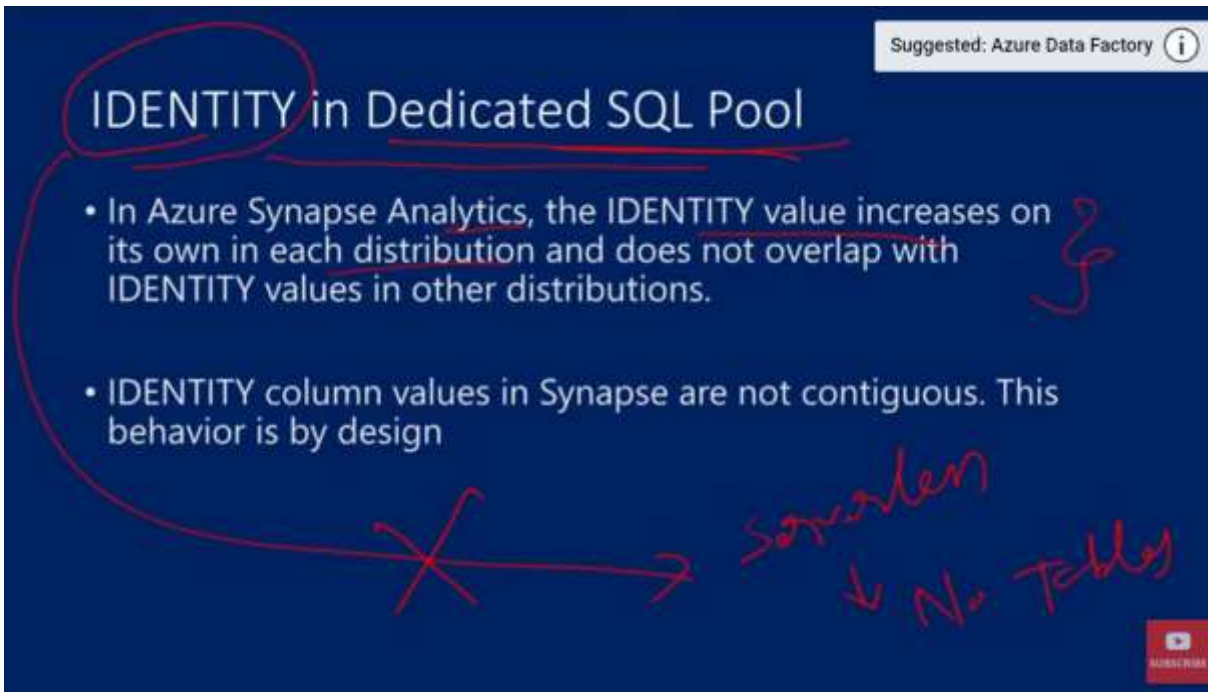
Surrogate Key

- A surrogate key on a table is a column with a unique identifier for each row. The key is not generated from the table data.
- You can use the IDENTITY property to achieve this goal simply and effectively without affecting load performance.

Identity means – if u put (10,1) then table will start from 10th identifier and with increament of 1

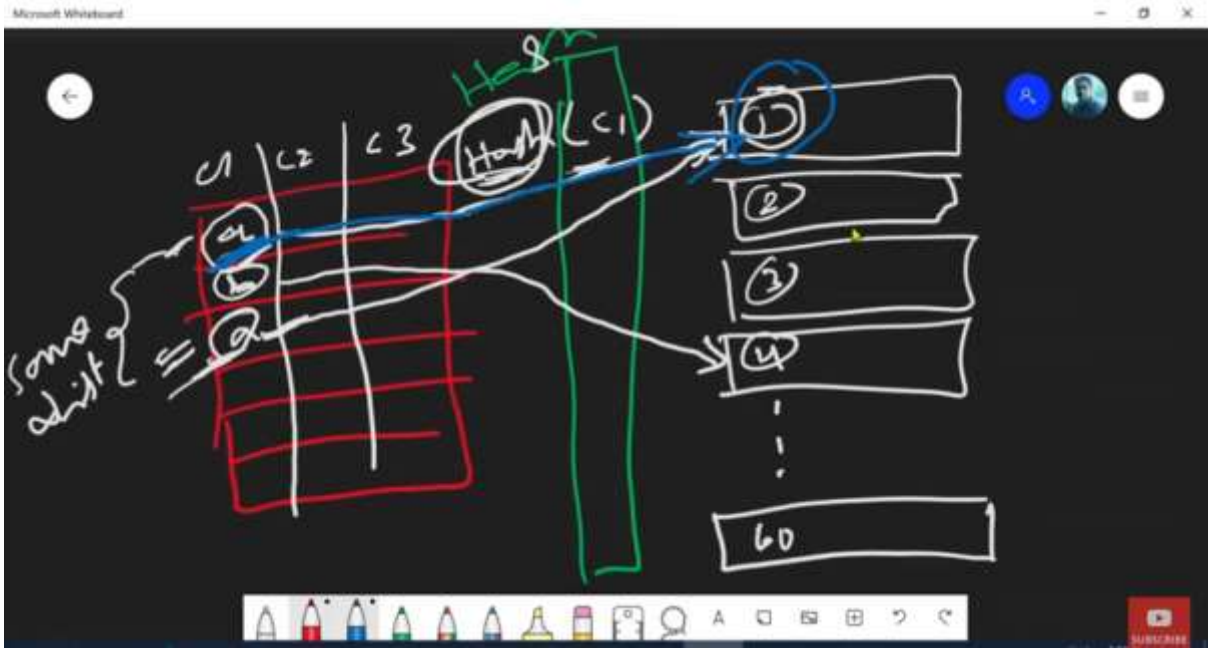


As serverless sql pool don't have tables, hence identity don't work in serverless

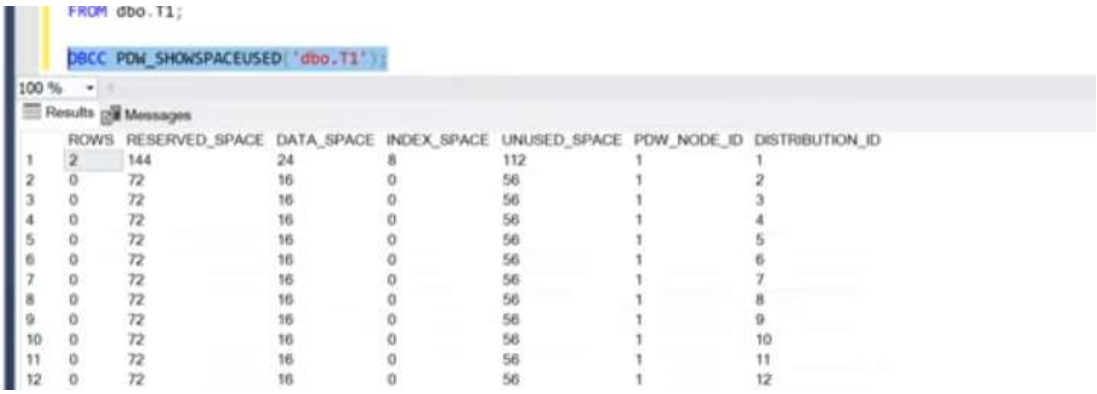


Each compute node we have 60 distributions

So the each column value will have a key and a hash value is calculated, and based on that it is sent to the distribution. If hash value is same, then it will be sent to the same distribution



Both are NULL inserted, as they are same, so both hash value would be same. And 60 distributions are there, so as has values same, so both are in same, so it is 1 & 61



FROM dbo.T1;

DBCC PDW_SHOWSPACEUSED('dbo.T1');

100 %

Results Messages

	ROWS	RESERVED_SPACE	DATA_SPACE	INDEX_SPACE	UNUSED_SPACE	PDW_NODE_ID	DISTRIBUTION_ID
1	2	144	24	8	112	1	1
2	0	72	16	0	56	1	2
3	0	72	16	0	56	1	3
4	0	72	16	0	56	1	4
5	0	72	16	0	56	1	5
6	0	72	16	0	56	1	6
7	0	72	16	0	56	1	7
8	0	72	16	0	56	1	8
9	0	72	16	0	56	1	9
10	0	72	16	0	56	1	10
11	0	72	16	0	56	1	11
12	0	72	16	0	56	1	12

```
INSERT INTO dbo.T1
VALUES (NULL); --- 1
```

100 %

Results | Messages

	ROWS	RESERVED_SPACE	DATA_SPACE	INDEX_SPACE	UNUSED_SPACE	PDW_NODE_ID	DISTRIBUTION_ID
1	2	144	24	8	112	1	1
2	0	72	16	0	56	1	2
3	0	72	16	0	56	1	3
4	0	72	16	0	56	1	4
5	0	72	16	0	56	1	5
6	0	72	16	0	56	1	6
7	0	72	16	0	56	1	7
8	0	72	16	0	56	1	8
9	0	72	16	0	56	1	9
10	0	72	16	0	56	1	10
11	0	72	16	0	56	1	11
12	0	72	16	0	56	1	12
13	0	72	16	0	56	1	13
14	0	72	16	0	56	1	14
15	0	72	16	0	56	1	15
16	0	72	16	0	56	1	16
17	0	72	16	0	56	1	17
18	0	72	16	0	56	1	18
19	0	72	16	0	56	1	19
20	0	72	16	0	56	1	20
21	0	72	16	0	56	1	21
22	0	72	16	0	56	1	22
23	0	72	16	0	56	1	23

That is 1 & 61 bcoz 60 are already reserved to the distribution and in case of identity, after 1 , then 61 will come

Suppose we insert another value

```
INSERT INTO dbo.T1
VALUES ('a');

SELECT *
FROM dbo.T1;
```

100 %

Results | Messages

	C1	C2
1	1	NULL
2	8	a
3	61	NULL

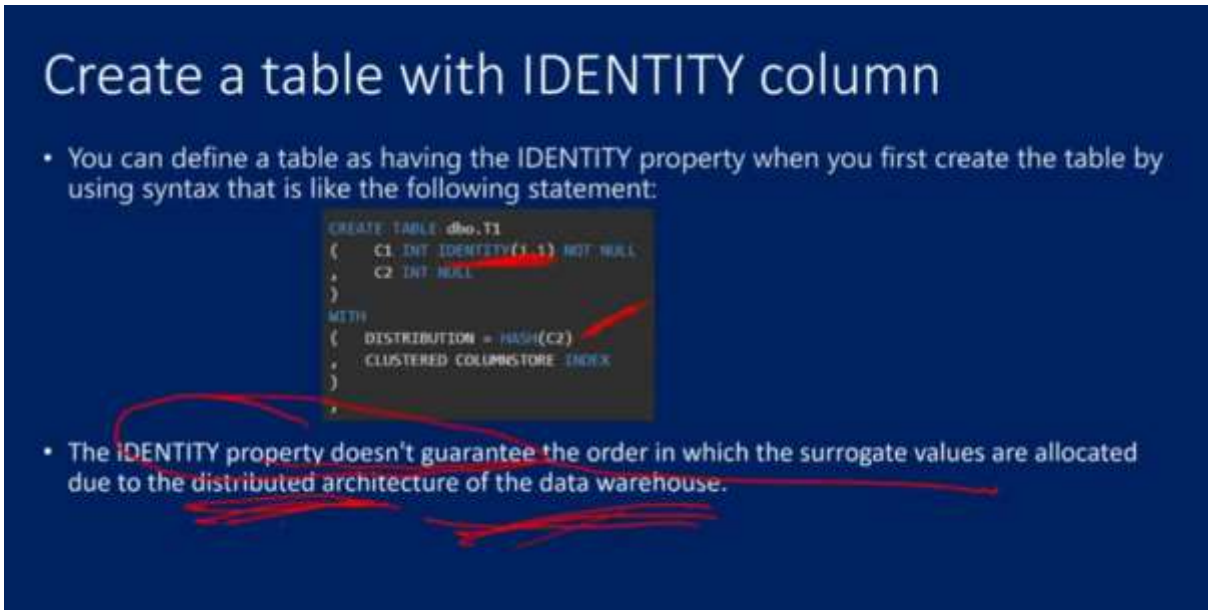
The hash value will be calculated and sent to respective distribution

100 %

Results | Messages

	ROWS	RESERVED_SPACE	DATA_SPACE	INDEX_SPACE	UNUSED_SPACE	PDW_NODE_ID	DISTRIBUTION_ID
1	2	144	24	8	112	1	1
2	0	72	16	0	56	1	2
3	0	72	16	0	56	1	3
4	0	72	16	0	56	1	4
5	0	72	16	0	56	1	5
6	0	72	16	0	56	1	6
7	0	72	16	0	56	1	7
8	1	144	24	8	112	1	8
9	0	72	16	0	56	1	9
10	0	72	16	0	56	1	10
11	0	72	16	0	56	1	11
12	0	72	16	0	56	1	12
13	0	72	16	0	56	1	13
14	0	72	16	0	56	1	14
15	0	72	16	0	56	1	15
16	0	72	16	0	56	1	16
17	0	72	16	0	56	1	17
18	0	72	16	0	56	1	18
19	0	72	16	0	56	1	19
20	0	72	16	0	56	1	20
21	0	72	16	0	56	1	21
22	0	72	16	0	56	1	22
23	0	72	16	0	56	1	23

IMP for interview - IDENTITY PROPERTY DON'T GUARANTEE ORDER



Explicitly inserting values into an IDENTITY column

Dedicated SQL pool supports SET IDENTITY_INSERT <your table> ON|OFF syntax. You can use this syntax to explicitly insert values into the IDENTITY column.

```
SET IDENTITY_INSERT dbo.T1 ON;

INSERT INTO dbo.T1
(   C1
,   C2
)
VALUES (-1,'UNKNOWN')
;

SET IDENTITY_INSERT dbo.T1 OFF;

SELECT     *
FROM    dbo.T1
;
```

Limitation of identity coloumn

Limitations

Can't use

- When the column data type is not INT or BIGINT
- When the column is also the distribution key
- When the table is an external table

---

26. OPENROWSET() function in Synapse SQL in Azure Synapse Analytics

OPENROWSET()

- The OPENROWSET(BULK...) function allows you to access files in Azure Storage. OPENROWSET function reads content of a remote data source (for example file) and returns the content as a set of rows.
- The OPENROWSET function can be referenced in the FROM clause of a query as if it were a table name OPENROWSET. It supports bulk operations through a built-in BULK provider that enables data from a file to be read and returned as a rowset.

OPENROWSET()

```
--OPENROWSET syntax for reading Parquet or Delta Lake (preview) files
OPENROWSET
( { BULK 'unstructured_data_path' , [DATA_SOURCE = <data source name>, ]
    FORMAT= [ 'PARQUET' | 'DELTA'] }
)
[WITH ( {'column_name' 'column_type' }) ]
[AS] table_alias(column_alias,...n)

--OPENROWSET syntax for reading delimited text files
OPENROWSET
( { BULK 'unstructured_data_path' , [DATA_SOURCE = <data source name>, ]
    FORMAT = 'CSV'
    [ <bulk_options> ] }
)
WITH ( {'column_name' 'column_type' [ 'column_ordinal' | 'json_path'] })
[AS] table_alias(column_alias,...n)

<bulk_options> ::=
[ , FIELDTERMINATOR = 'char' ]
[ , ROWTERMINATOR = 'char' ]
[ , ESCAPECHAR = 'char' ]
[ , FIRSTROW = 'first_row' ]
[ , FIELDQUOTE = 'quote_characters' ]
[ , DATA_COMPRESSION = 'data_compression_method' ]
[ , PARSER_VERSION = 'parser_version' ]
[ , HEADER_ROW = { TRUE | FALSE } ]
[ , DATAFILETYPE = { 'char' | 'widechar' } ]
[ , CODEPAGE = { 'ACP' | 'OEM' | 'RAW' | 'code_page' } ]
[ , ROWSET_OPTIONS = '{"READ_OPTIONS":["ALLOW_INCONSISTENT_READS"]}' ]
```

```
29  SELECT *
30  FROM OPENROWSET(
31      BULK 'https://maheersa.blob.core.windows.net/data/holidays.csv',
32      FORMAT = 'CSV',
33      PARSER_version = '2.0',
34      HEADER_ROW = TRUE
35  )
36  as rowFromfile
```

Messages

1:58:24 AM Started executing query at Line 29

File 'https://maheersa.blob.core.windows.net/data/holidays.csv' cannot be opened because it does not exist or it is used by another process.
Visit this article to learn more about this error

Total execution time: 00:00:01.011

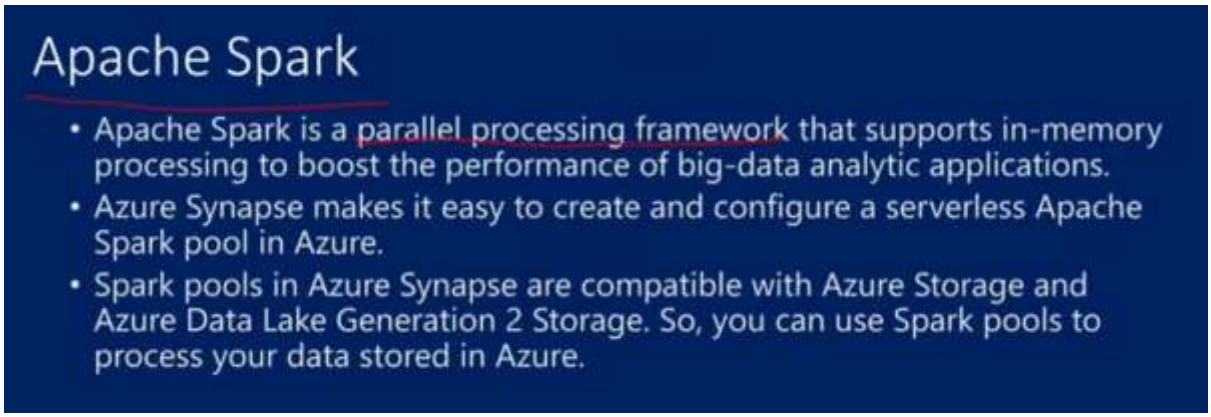
. The easiest way is to grant yourself a Storage Blob Data Contributor role on the storage account you're trying to query.

```
SELECT *
FROM OPENROWSET(
    BULK 'https://maheeradlsgen2.dfs.core.windows.net/synapsedemo/data/dates.c
sv',
    FORMAT = 'CSV',
    PARSER_version = '2.0',
    HEADER_ROW = TRUE
)
WITH
(
    term_id int,
    term_name NVARCHAR(20)
)
as rowFromfile

SELECT TOP 10 *
FROM OPENROWSET(
    BULK 'https://maheeradlsgen2.dfs.core.windows.net/synapsedemo/data/NYCTrip
Small.parquet',
    FORMAT = 'PARQUET'

)
WITH
(
    DateID BIGINT,
    MedallionID INT
)
as rowFromfile
```

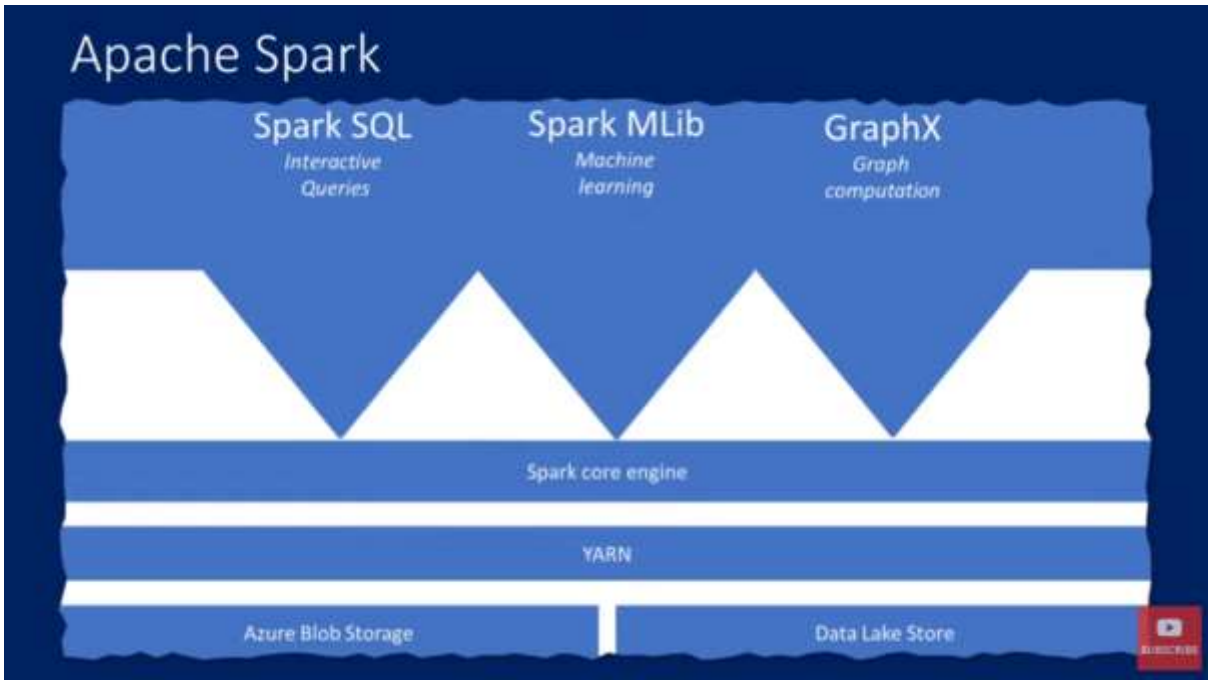
```sql
SELECT *
FROM OPENROWSET(
    BULK 'https://maheersa.blob.core.windows.net/data/holidays.csv',
    FORMAT = 'CSV',
    PARSER_version = '2.0',
    HEADER_ROW = TRUE
)
as rowFromfile
-
- The easiest way is to grant yourself a Storage Blob Data Contributor role on
 the storage account you're trying to query.
-- if u dont wanna do all that, u can make external data source and use it
```

---

27. Apache Spark in Azure Synapse Analytics

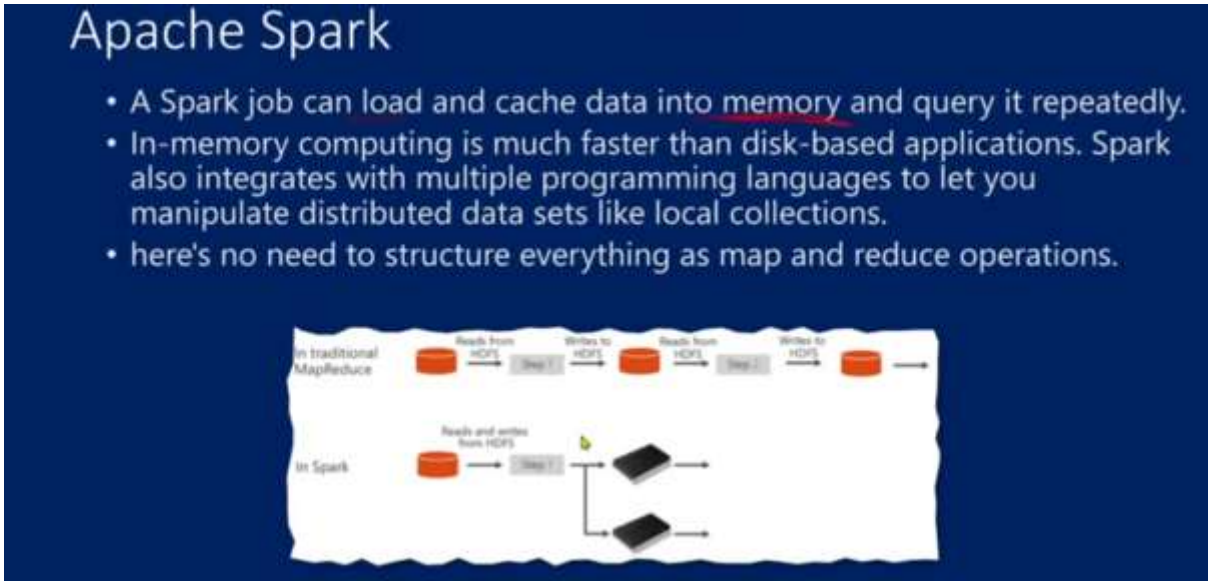


Transformations upon big data, it will take data, partition it, and process parallel and it happens within the memory



Cluster manager is there, master node and worker node is there, the cluster manger ensure that there is proper communication b/w master and worker nodes to make the transformation work.

APACHE YARN is the Cluster manager in case of Azure Synapse. Upon which apache spark engine is built. This engine has – spark sql for intertactive queries, spark MLib for machine learning libraries and Graph X for graph computation



The query execution happens inside the memory no the architecture, that is why it is very much faster than the traditional map reduce technology that helps in data transformation of big data

In MAP REDUCE, there will a data inside the disk(HDFS), reads the data from hdfs, then it performs logical exectution, once the logical execution completes, it writes back into the disk. And again we read, apply logic and write into the disk, so this continues. This is much slower.

In spark, the data will be taken for the first time from the disk into the memory. And all the logic is executed in the memory

Benefits of spark pool

Benefits of Spark Pool

Feature	Description
Speed and efficiency	Spark instances start in approximately 2 minutes for fewer than 60 nodes and approximately 5 minutes for more than 60 nodes. The instance shuts down, by default, 5 minutes after the last job executed unless it is kept alive by a notebook connection.
Ease of creation	You can create a new Spark pool in Azure Synapse in minutes using the Azure portal, Azure PowerShell, or the Synapse Analytics .NET SDK. See Get started with Spark pools in Azure Synapse Analytics.
Ease of use	Synapse Analytics includes a custom notebook derived from Nteract. You can use these notebooks for interactive data processing and visualization.
REST APIs	Spark in Azure Synapse Analytics includes Apache Livy, a REST API-based Spark job server to remotely submit and monitor jobs.
Support for Azure Data Lake Storage Generation 2	Spark pools in Azure Synapse can use Azure Data Lake Storage Generation 2 as well as BLOB storage. For more information on Data Lake Storage, see Overview of Azure Data Lake Storage.
Integration with third-party IDEs	Azure Synapse provides an IDE plugin for JetBrains' IntelliJ IDEA that is useful to create and submit applications to a Spark pool.
Pre-loaded Anaconda libraries	Spark pools in Azure Synapse come with Anaconda libraries pre-installed. Anaconda provides close to 200 libraries for machine learning, data analysis, visualization, etc.
Scalability	Apache Spark in Azure Synapse pools can have Auto-Scale enabled, so that pools scale by adding or removing nodes as needed. Also, Spark pools can be shut down with no loss of data since all the data is stored in Azure Storage or Data Lake Storage.

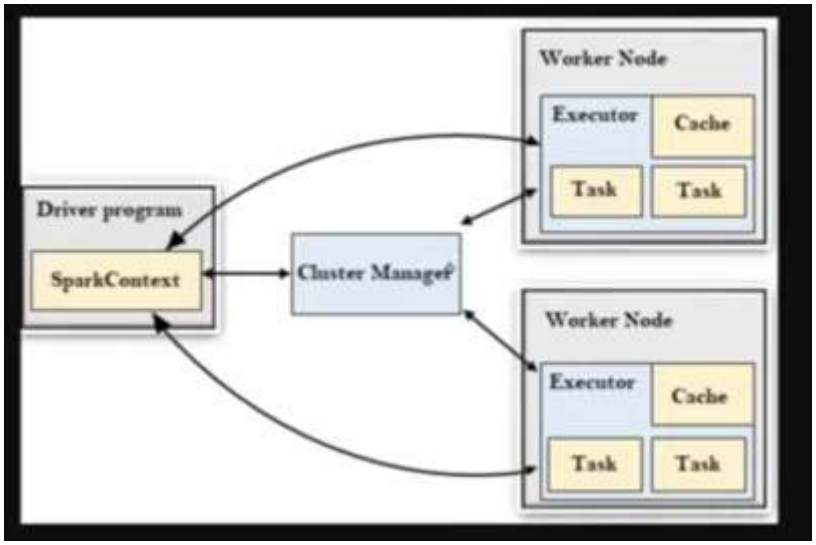
Nodes are like machines

Rest API is supported in Apache Livy.

SPARK POOL ARCHITECTURE

Spark pool architecture

- The SparkContext can connect to the cluster manager. Once connected, Spark acquires executors on nodes in the pool. Next, it sends your application code (defined by JAR or Python files passed to SparkContext) to the executors. Finally, SparkContext sends tasks to the executors to run.



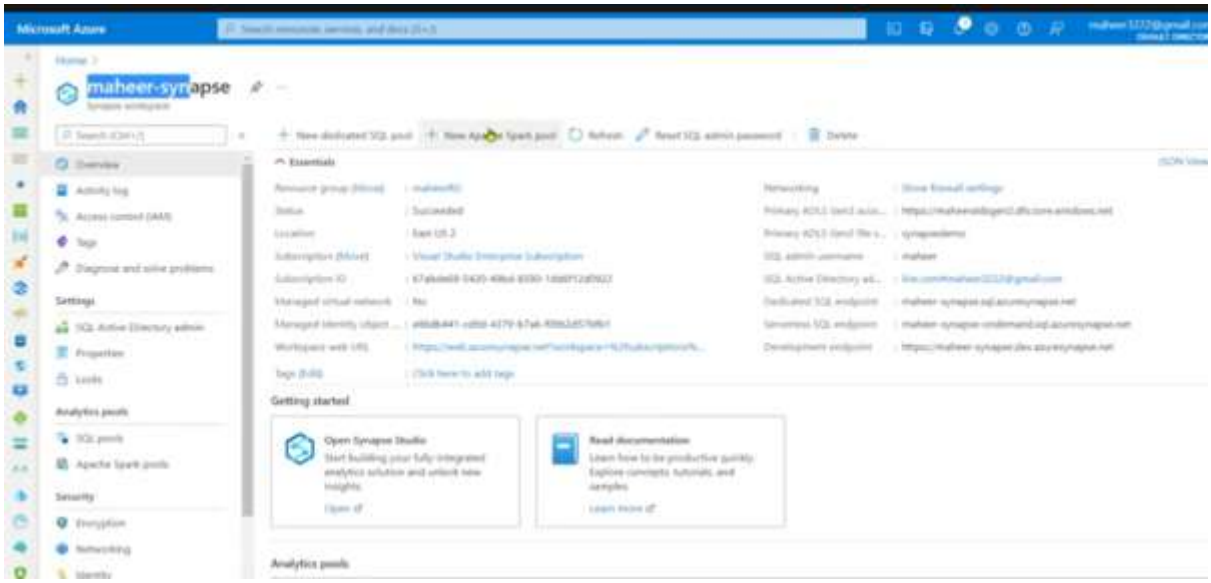
Master node/ Driver program is like a lead and worker nodes are like a teammates. So the lead assign the work to teammates by coordinating with the cluster manager.

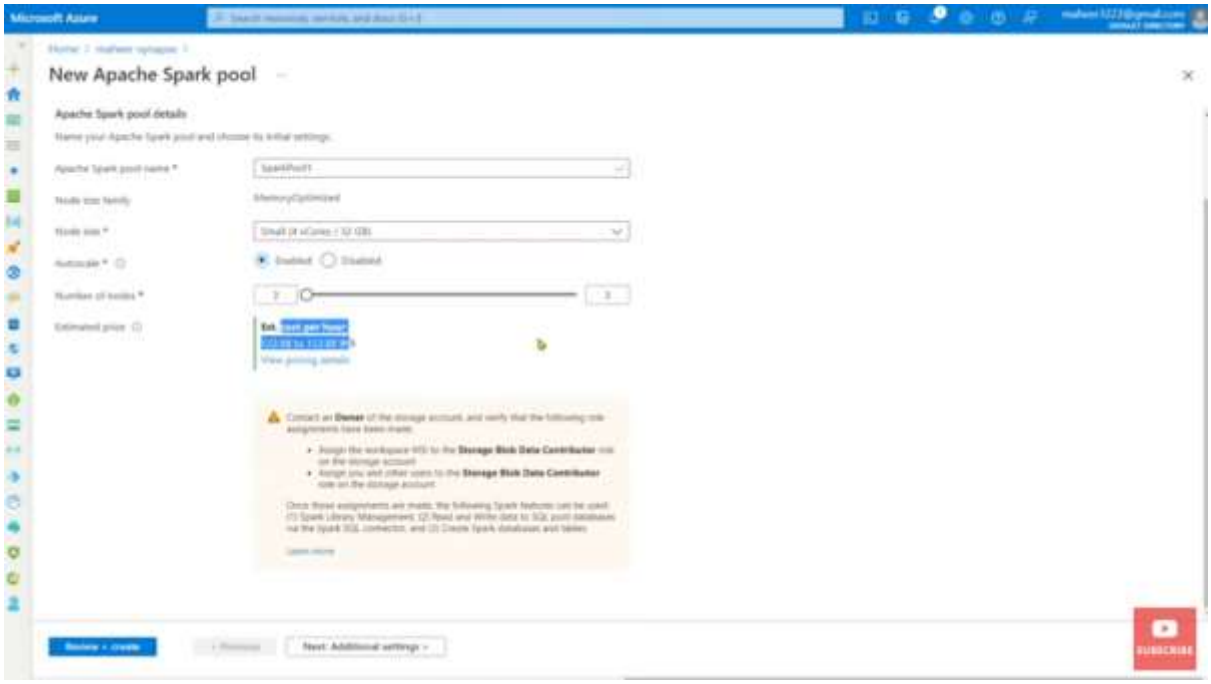
So the driver program contains the spark context, i.e. all the transformation logic. This is going to connect with the cluster manager, and the cluster manager will identify the executor(It performs the tasks) in worker node. Cluster manager takes the entire logic and distributes into the tasks, and the tasks are distributed among the multiple worker nodes

The executor execute the task, that means it applies transformation logic on the data and the output is stored in cache, we r not writing data into harddisk. The output is call RDD objects.

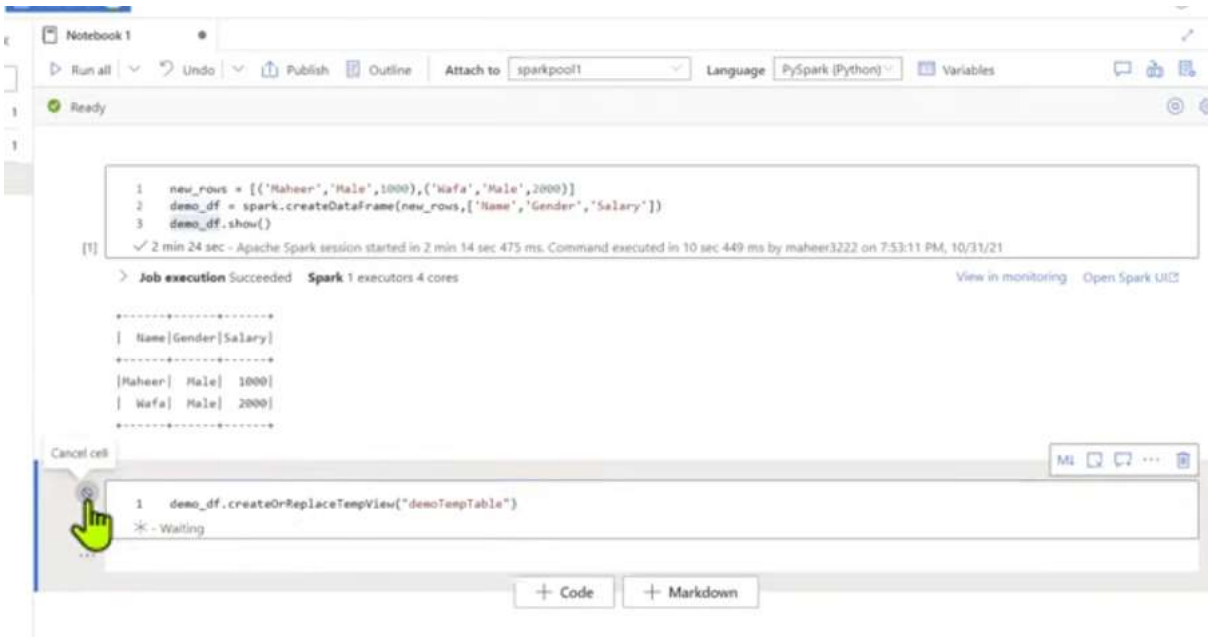
---

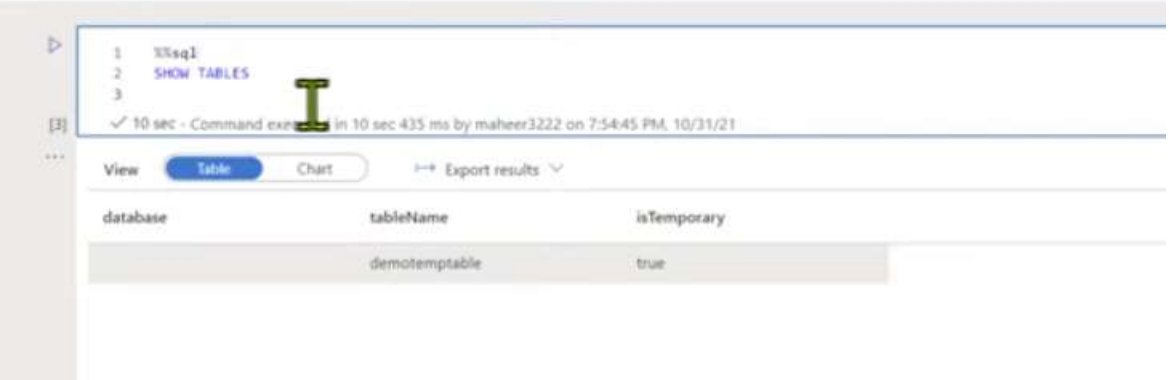
28. Create a Spark Pool with Azure Portal

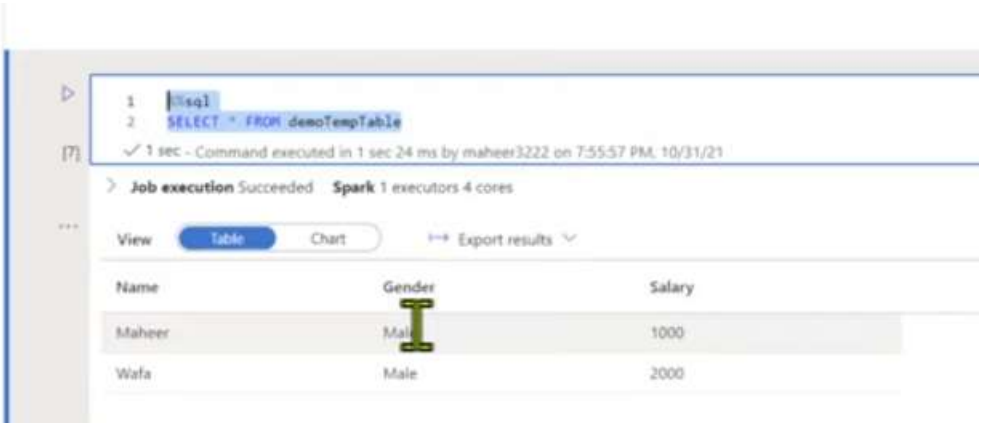




29. Create a Notebook in Azure Synapse Analytics







---

30. Pandas to read/write Azure Data Lake Storage Gen2 data in Apache Spark pool in Synapse Analytics