**Analyze data with dedicated SQL pools**

**Create a dedicated SQL pool**

1. In Synapse Studio, on the left-side pane, select **Manage** > **SQL pools** under **Analytics pools**.
2. Select **New**
3. For **Dedicated SQL pool name** select **SQLPOOL1**
4. For **Performance level** choose **DW100C**
5. Select **Review + create** > **Create**. Your dedicated SQL pool will be ready in a few minutes.

Your dedicated SQL pool is associated with a SQL database that's also called **SQLPOOL1**.

1. Navigate to **Data** > **Workspace**.
2. You should see a database named **SQLPOOL1**. If you do not see it, click **Refresh**.

A dedicated SQL pool consumes billable resources as long as it's active. You can pause the pool later to reduce costs.

**Note**

When creating a new dedicated SQL pool (formerly SQL DW) in your workspace, the dedicated SQL pool provisioning page will open. Provisioning will take place on the logical SQL server.

**Load the NYC Taxi Data into SQLPOOL1**

1. In Synapse Studio, navigate to the **Develop** hub, click the **+** button to add new resource, then create new SQL script.
2. Select the pool 'SQLPOOL1' (pool created in STEP 1 ) in **Connect to** drop down list above the script.
3. Enter the following code:

IF NOT EXISTS (SELECT \* FROM sys.objects O JOIN sys.schemas S ON O.schema\_id = S.schema\_id WHERE O.NAME = 'NYCTaxiTripSmall' 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://contosolake.dfs.core.windows.net/users/NYCTripSmall.parquet'

WITH

(

FILE\_TYPE = 'PARQUET'

,MAXERRORS = 0

,IDENTITY\_INSERT = 'OFF'

)

1. Click the **Run** button to execute the script.
2. This script will finish in less than 60 seconds. It loads 2 million rows of NYC Taxi data into a table called dbo.NYCTaxiTripSmall.

**Explore the NYC Taxi data in the dedicated SQL pool**

1. In Synapse Studio, go to the **Data** hub.
2. Go to **SQLPOOL1** > **Tables**.
3. Right-click the **dbo.NYCTaxiTripSmall** table and select **New SQL Script** > **Select TOP 100 Rows**.
4. Wait while a new SQL script is created and runs.
5. Notice that at the top of the SQL script **Connect to** is automatically set to the SQL pool called **SQLPOOL1**.
6. Replace the text of the SQL script with this code and run it.

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;

This query shows how the total trip distances and average trip distance relate to the number of passengers.

1. In the SQL script result window, change the **View** to **Chart** to see a visualization of the results as a line chart.