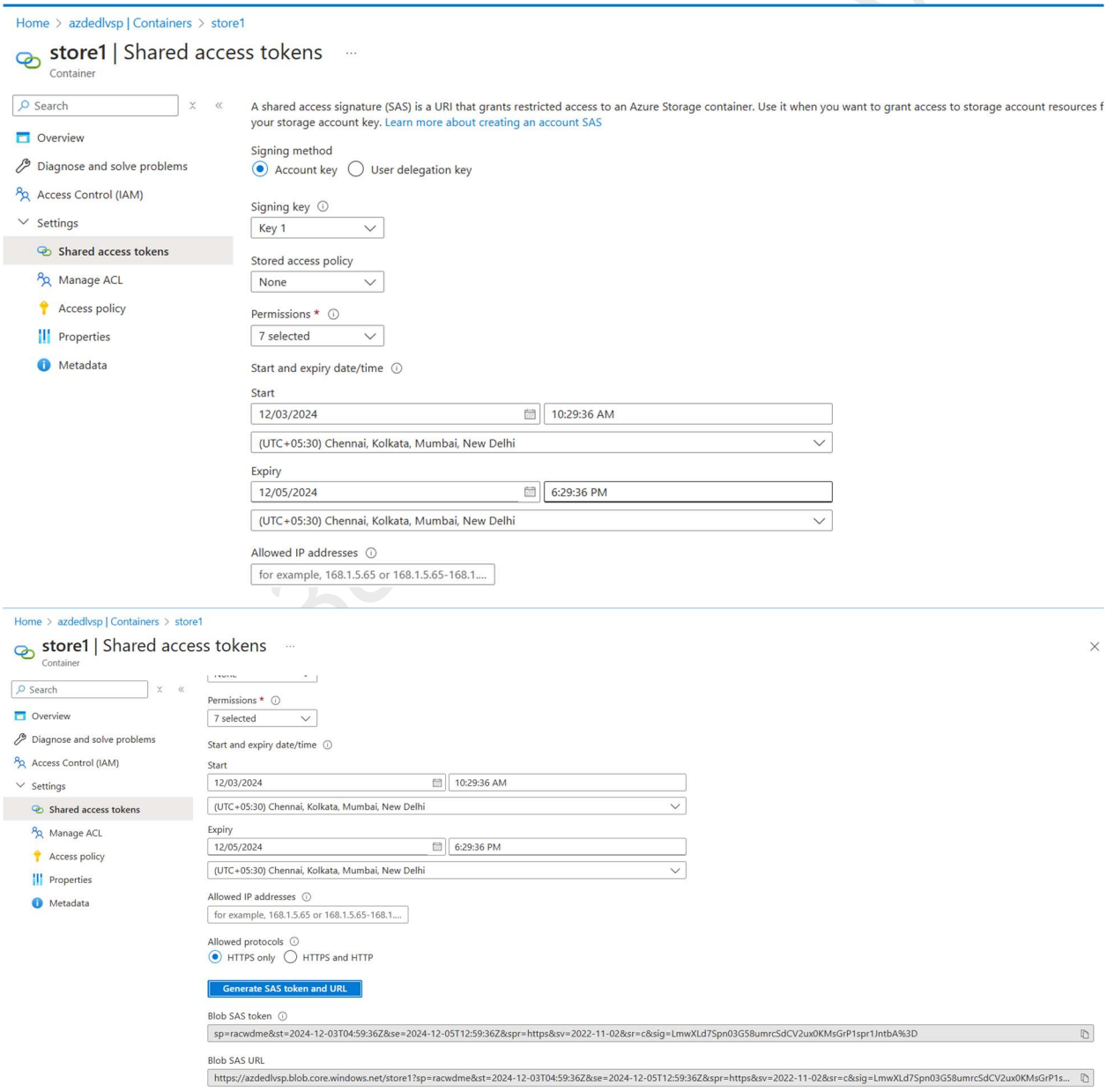
Accessing storage container via Azure storage explorer.

Prerequisites: Ensure Azure storage explorer is installed and active in the local.

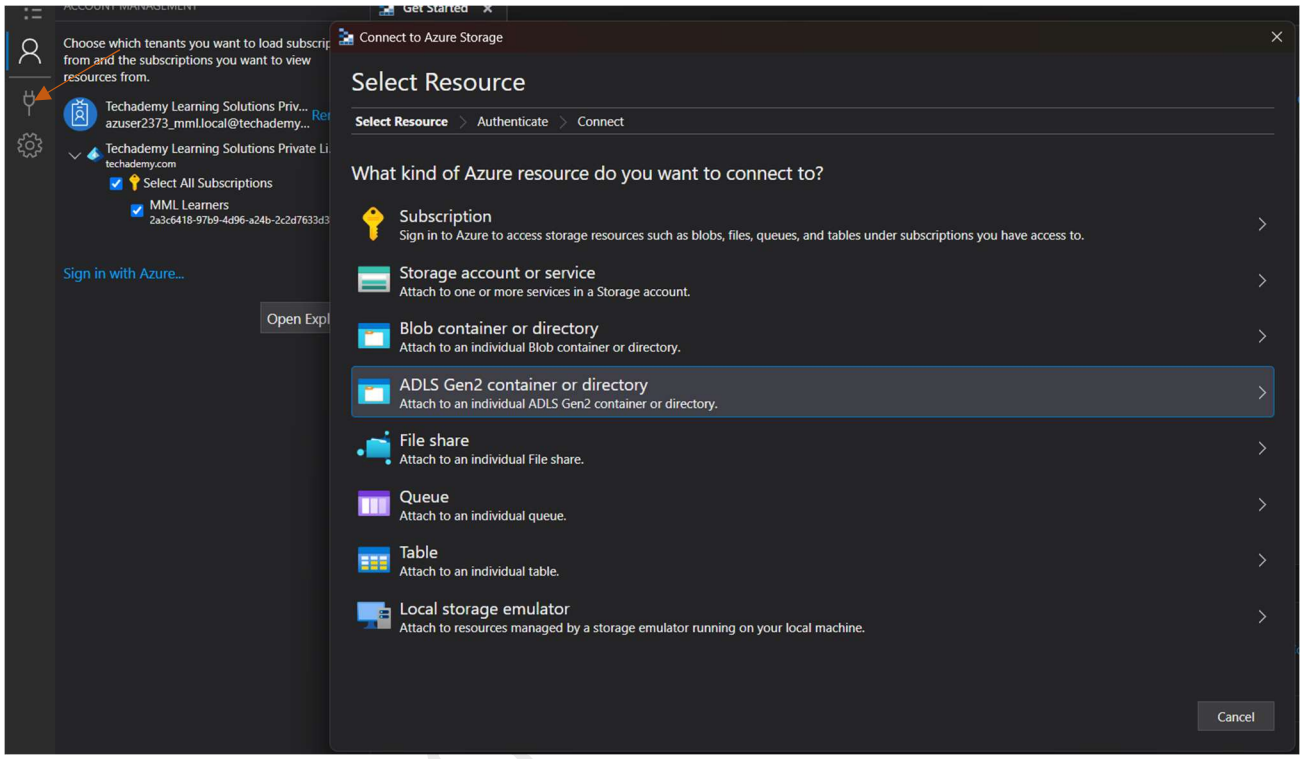
Steps to create SAS token:

1. Create storage account
2. Create a container
3. Upload some sample files to the container
4. In that container go to Shared Access Tokens under settings.
5. Provide the necessary permission and expiry date.
6. Generate SAS token and copy the second URL and paste it in notepad.

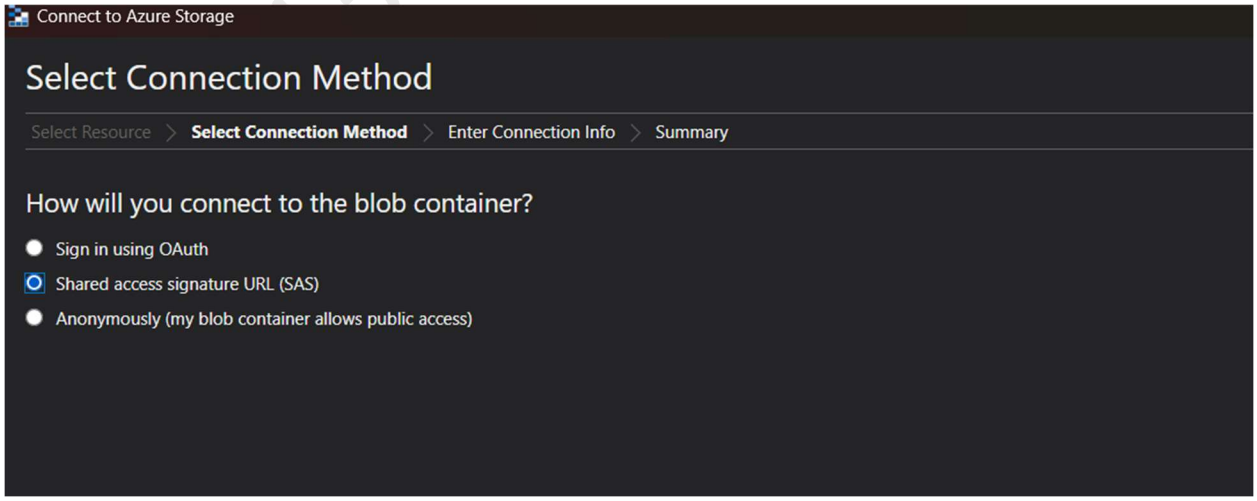


Steps to connect:

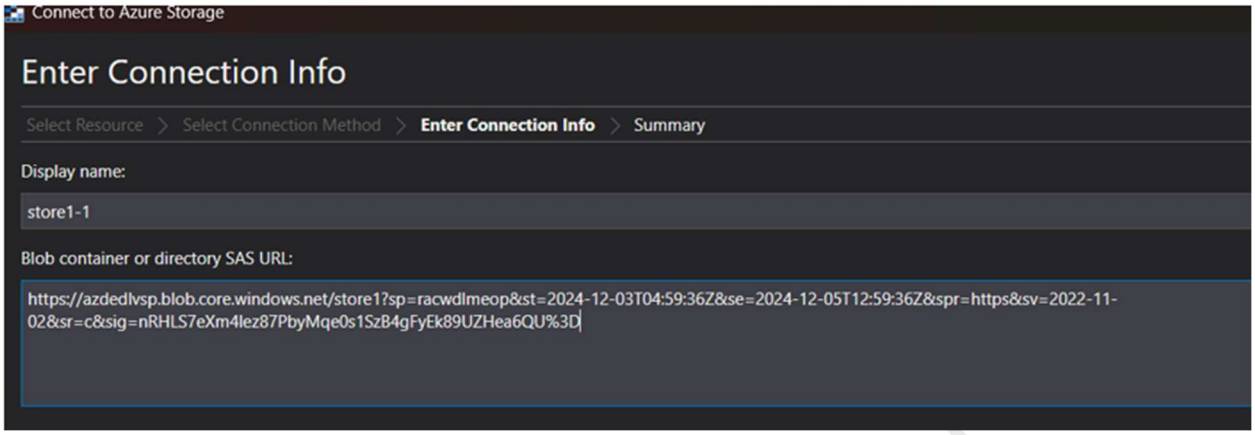
1. Go to Storage explorer
2. Click on Plug symbol
3. Click on ADLS Gen2 container or directory (for Azure data lake storage). If it is Blob storage then choose Blob container or directory.



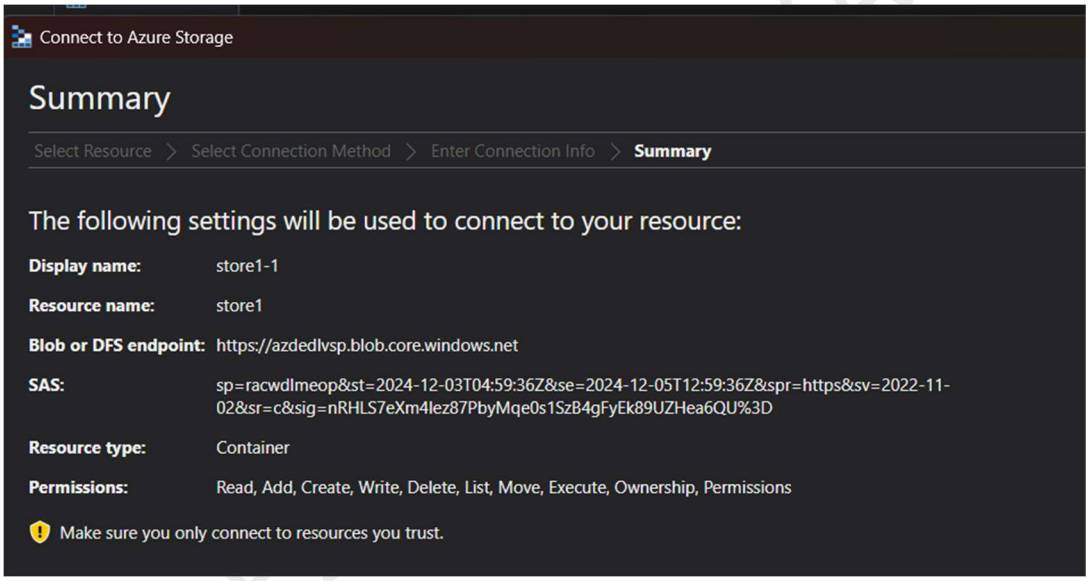
4. Select the Shared access signature URL (SAS) and click next.



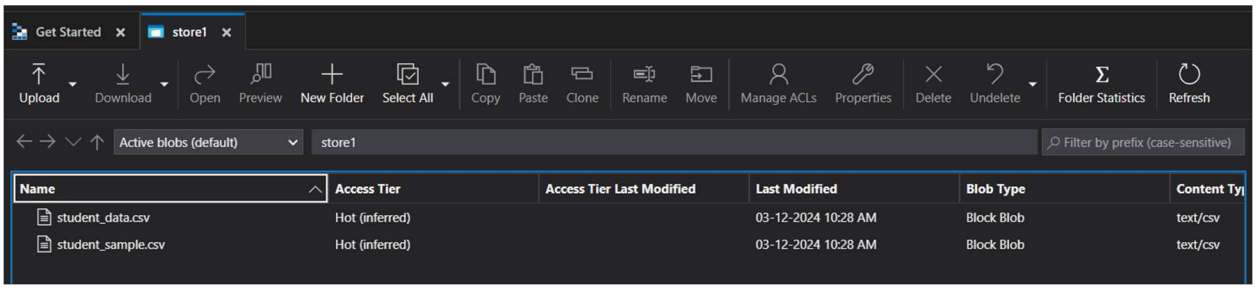
5. Enter the display name and paste the URL which we have in notepad. And click next.



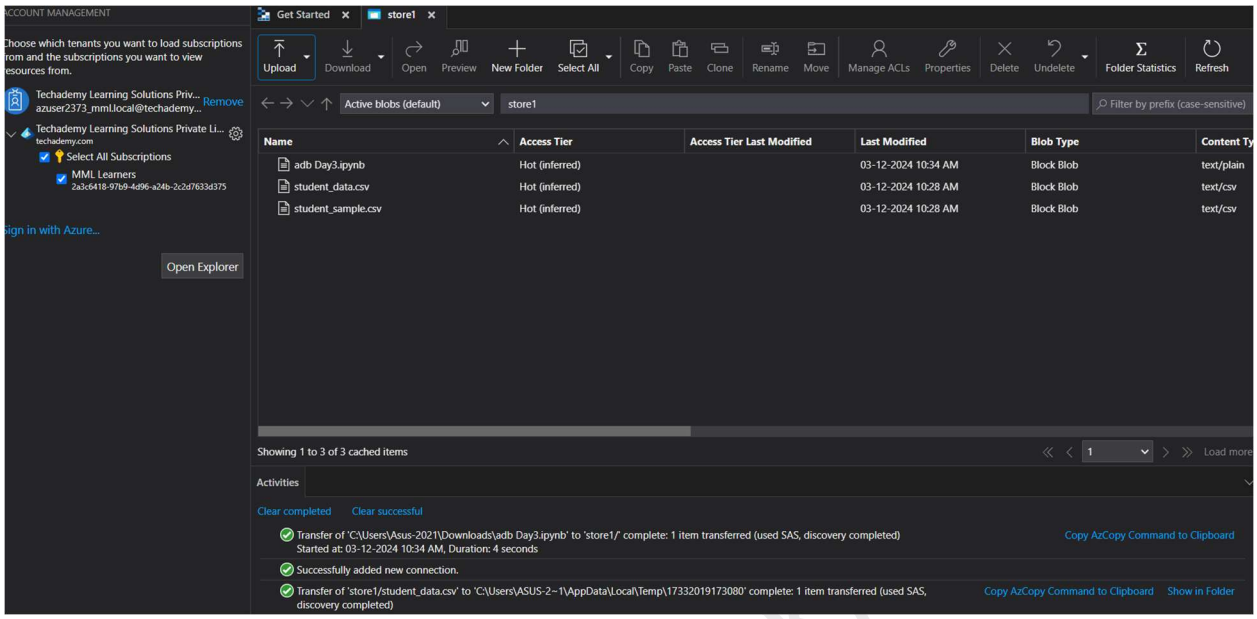
6. Check the summary and click next



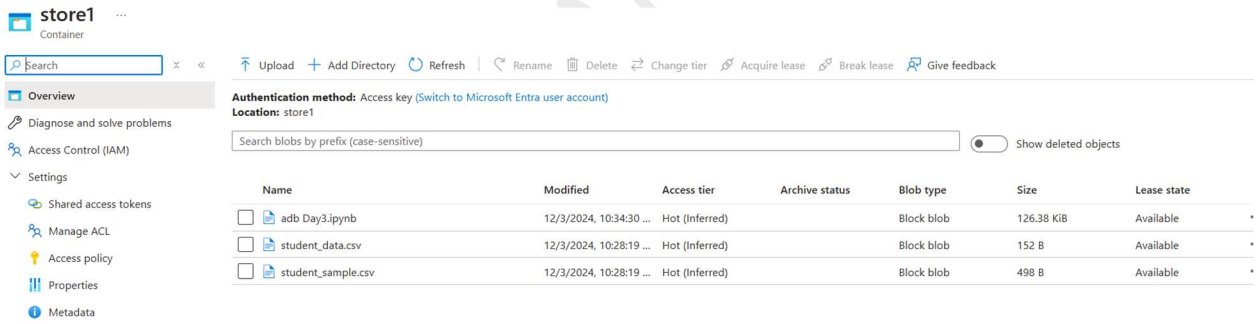
7. Now we get access to the storage container which we have created in Azure web.

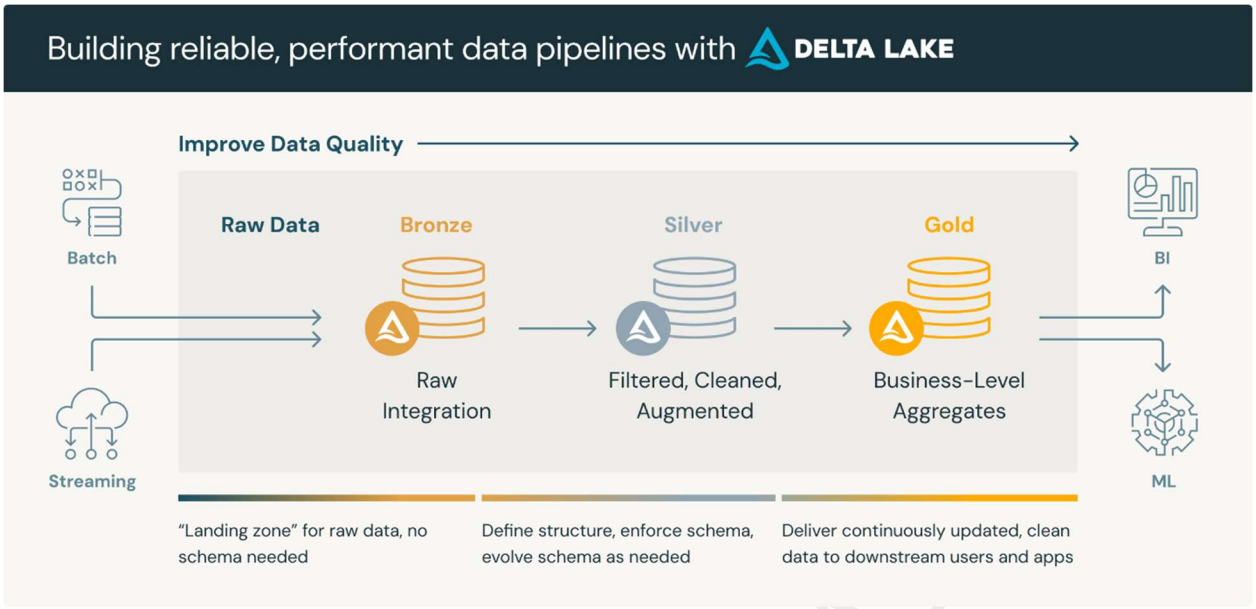


8. We can add/ update/delete the files in the local via storage explorer (for given permissions).



9. Refresh the Azure web, the file which we have uploaded in the local via storage explorer is visible.





Delta Live Tables, Materialized views

4 minutes ago (12s) 2

```
%sql
CREATE OR REFRESH STREAMING TABLE taxi_raw_records
AS
SELECT *
FROM
STREAM(samples.nyctaxi.trips);
```

taxi_raw_records is defined as a **Delta Live Tables** dataset with schema:

Name	Type
tpep_pickup_datetime	timestamp
tpep_dropoff_datetime	timestamp
trip_distance	double
fare_amount	double
pickup_zip	int
dropoff_zip	int

To populate your table you must either:

- Run an existing pipeline using the **Delta Live Tables** menu
- Create a new pipeline: Create Pipeline

```sql
%sql
CREATE OR REFRESH STREAMING TABLE taxi_raw_recordsonadb
(CONSTRAINT valid_distance EXPECT (trip_distance > 0.0) ON VIOLATION DROP ROW )
AS SELECT
  *
FROM
  STREAM(samples.nyctaxi.trips);
```

taxi_raw_recordsonadb is defined as a **Delta Live Tables** dataset with schema:

Name	Type
tpep_pickup_datetime	timestamp
tpep_dropoff_datetime	timestamp
trip_distance	double
fare_amount	double
pickup_zip	int
dropoff_zip	int

To populate your table you must either:

- Run an existing pipeline using the **Delta Live Tables** menu
- Create a new pipeline: Create Pipeline

```sql
%sql
CREATE OR REFRESH STREAMING TABLE flagged_rides
AS SELECT
  date_trunc('week', tpep_pickup_datetime) as week,
  pickup_zip as zip,
  trip_distance, fare_amount
FROM
 STREAM(samples.nyctaxi.trips)
WHERE   ((pickup_zip = dropoff_zip AND fare_amount > 50) OR
        (trip_distance < 5 AND fare_amount > 50));
```

flagged_rides is defined as a **Delta Live Tables** dataset with schema:

Name	Type
week	timestamp
zip	int
trip_distance	double
fare_amount	double

To populate your table you must either:

- Run an existing pipeline using the **Delta Live Tables** menu
- Create a new pipeline: Create Pipeline

```sql
%sql
CREATE
OR REFRESH MATERIALIZED VIEW weekly_stats
AS SELECT
  date_trunc("week", tpep_pickup_datetime) as week,
  AVG(fare_amount) as avg_amount,
  AVG(trip_distance) as avg_distance
FROM
 live.taxi_raw_records
GROUP BY
  week
ORDER by week ASC;
```

weekly_stats is defined as a **Delta Live Tables** dataset with schema:

Name	Type
week	timestamp
avg_amount	double
avg_distance	double

To populate your table you must either:

- Run an existing pipeline using the **Delta Live Tables** menu
- Create a new pipeline: Create Pipeline

```sql
%sql
CREATE OR REPLACE MATERIALIZED VIEW top_m
AS SELECT
  weekly_stats.week,
  ROUND(avg_amount,2) as avg_amount,
  ROUND(avg_distance,3) as avg_distance,
  fare_amount,trip_distance, zip

  FROM live.flagged_rides
  LEFT JOIN live.weekly_stats ON flagged_rides.week = weekly_stats.week
  ORDER BY fare_amount DESC;
```

top_m is defined as a **Delta Live Tables** dataset with schema:

Name	Type
week	timestamp
avg_amount	double
avg_distance	double
fare_amount	double
trip_distance	double
zip	int

To populate your table you must either:

- Run an existing pipeline using the **Delta Live Tables** menu
- Create a new pipeline: Create Pipeline