

DBSQL Workshop Instructions

a. Create Catalog / Schema

```
SELECT current_user();
```

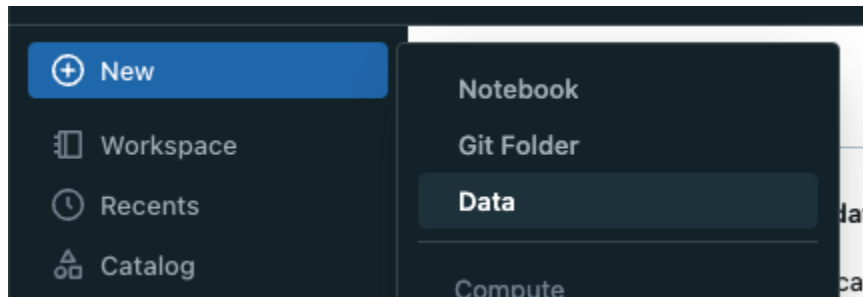
```
CREATE CATALOG MYCATALOG;
```

```
USE CATALOG MYCATALOG;
```

```
CREATE SCHEMA MYSCHEMA;
```

b. Upload csv file and create table

- i. File is the “nyc_boroughs.csv” in the zip file
- ii. Upload file and create table in your catalog and schema



From local files (2)



Create or modify table

Upload tabular data files to create a new table or replace an existing one


```
on a.pickup_zip = b.zip
```

Write a Common Table Expression and leverage SQL Functions to create a complex query. Copy and paste the below SQL and modify for your Catalog and Schema in the first query.

```
with trips_detailed as (  
  select a.*,  
    hour(tpep_pickup_datetime) as trip_hour,  
    date(tpep_pickup_datetime) as trip_date,  
    cast(date_format(tpep_pickup_datetime, 'yyyyMM') as int)  
    as trip_month,  
    year(tpep_pickup_datetime) as trip_year,  
    date_format(tpep_pickup_datetime, 'EEEE') as  
    trip_day_of_week,  
    b.borough as pickup_borough,  
    b.neighborhood as pickup_neighborhood,  
    c.borough as dropoff_borough,  
    c.neighborhood as dropoff_neighborhood  
  from samples.nyctaxi.trips a  
  join <YOUR CATALOG>.<YOUR SCHEMA>.nyc_boroughs b  
  on a.pickup_zip = b.zip_code  
  join <YOUR CATALOG>.<YOUR SCHEMA>.nyc_boroughs c  
  on a.dropoff_zip = c.zip_code),
```

```
trips_agg as (  
  select  
    trip_date,  
    trip_hour,  
    trip_month,  
    trip_year,  
    trip_day_of_week,  
    pickup_borough,  
    pickup_neighborhood,  
    dropoff_borough,  
    dropoff_neighborhood,  
    count(*) as trips,  
    sum(trip_distance) as total_miles_traveled,  
    avg(trip_distance) as avg_distance,  
    sum(fare_amount) as total_fare  
  from trips_detailed  
  group by  
    trip_date,  
    trip_hour,  
    trip_month,  
    trip_year,
```

```

trip_day_of_week,
pickup_borough,
pickup_neighborhood,
dropoff_borough,
dropoff_neighborhood)

select
trip_date,
trip_hour,
trip_month,
trip_year,
trip_day_of_week,
pickup_borough,
pickup_neighborhood,
dropoff_borough,
dropoff_neighborhood,
trips,
total_miles_traveled
avg_distance,
total_fare,
lag(trips) over (partition by
trip_hour,
trip_month,
trip_year,
trip_day_of_week,
pickup_borough,
pickup_neighborhood,
dropoff_borough,
dropoff_neighborhood
order by trip_date) as last_month_trips
from trips_agg

```

Create a simple visualization.

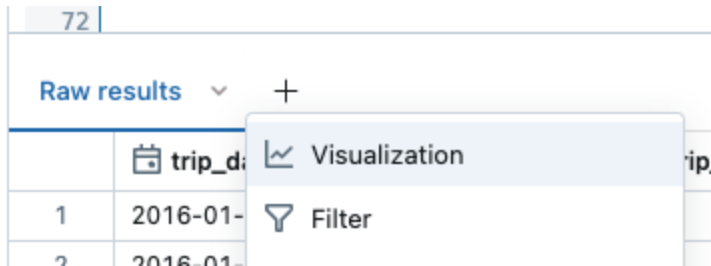
```

69 | order by trip_date) as l
70 | from trips_agg
71 |
72 |

```

Raw results ▾ +

trip_date	trips
2013-01-01	1



Choose -> Save

Line 1

Visualization type

Line

General X axis Y axis Series Colors Data labels

☐ Horizontal chart

X column

trip_date None

Y columns

trips Sum

Add column

Group by

Choose column...

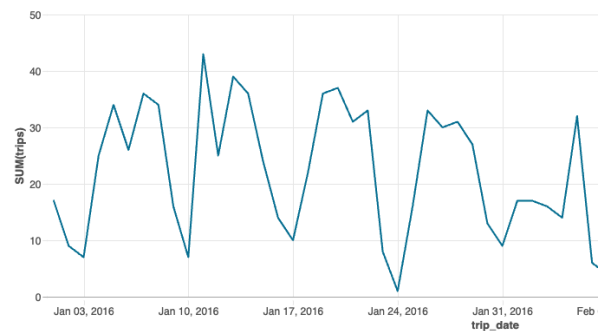
Error column

Add column

☐ Normalize values to percentage

Missing and NULL values

Convert to 0 and display in chart



How to Build a Lakeview Dashboard - 10 Mins

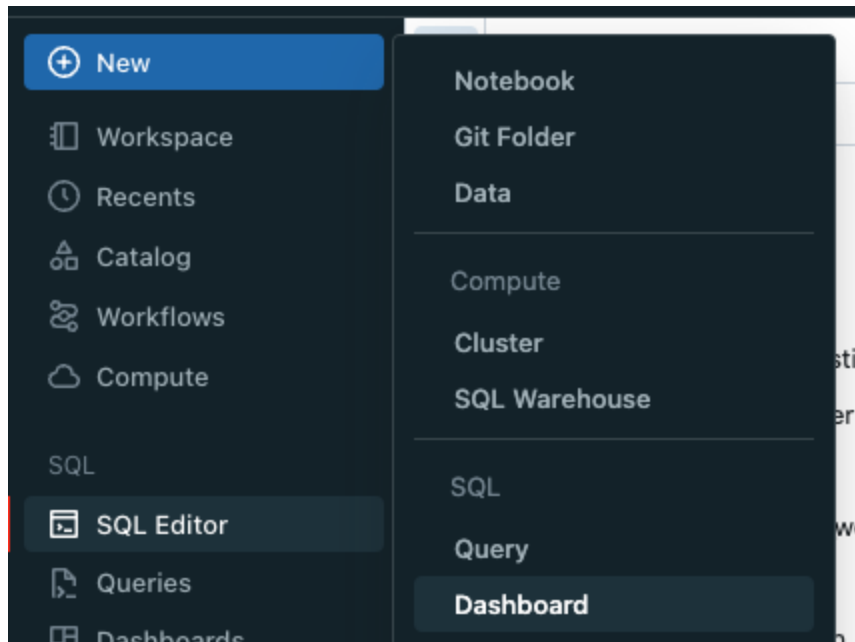
- ii. Materialized View
- iii. Data sets - Parameters
- iv. Visualizations - Filters
- v. Charts with AI

Hands On - Build an Example Dashboard with Taxi Demo Data - 30 Mins

Add the following line to the top of the Query we created in the last session.

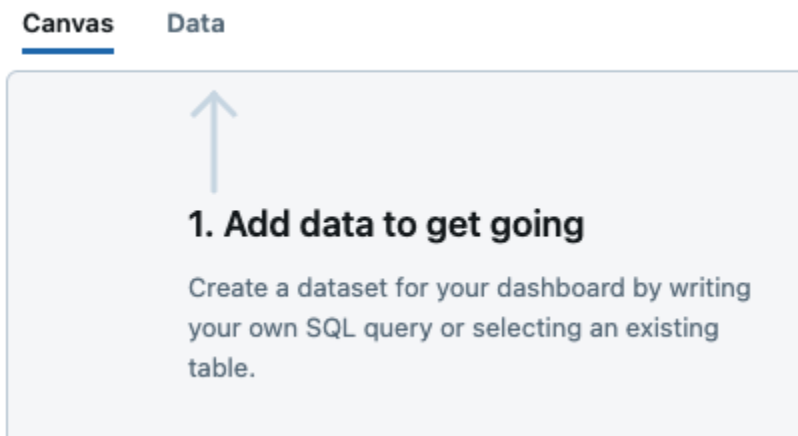
```
create materialized view <YOUR CATALOG>.<YOUR SCHEMA>.nyc_taxi_trips_mv as
with trips_detailed as (
```

This will create a Materialized View in the same schema you created the nyc_boroughs table.




Make sure you use a new query window or remove the visualization, or you may get an error on the visualization.

Select the Data link at the top.



On the data tab, choose "Select a table"

Canvas **Data**

 **Datasets**




+ Create from SQL



+ Select a table

Choose your catalog and schema and then select the table for the materialized view (nyc_taxi_trips_mv if you named it the same as mine).

Select table

 Search is available for Unity Catalog tables only. To use a Hive Metastore table in a dashboard, please choose the ' option.


cmoore_customer_... X

✕ ▼


customer_demos X

✕ ▼


Clear filters

 trips

cmoore_customer_demos.customer_demos • Cary Moore

 rptmgr_com_service_data_raw

cmoore_customer_demos.customer_demos • Cary Moore

 nyc_taxi_trips_mv

cmoore_customer_demos.customer_demos • Cary Moore

You should see something like this:


▶ Run

just now

1

SELECT * FROM cmoore_customer_demos.customer_demos.nyc_taxi_trips_mv

+ Add parameter

	 trip_date	¹ ₃ trip_hour	¹ ₃ trip_month	¹ ₃ trip_year	^A _C trip_day_of_week	^A _C pickup
1	2016-01-22	0	201601	2016	Friday	Brooklyn
2	2016-01-15	0	201601	2016	Friday	Manhattan
3	2016-01-29	0	201601	2016	Friday	Manhattan
4	2016-01-01	0	201601	2016	Friday	Manhattan
-	-----	-	-----	----	---	-----

Now we can select the Canvas back at the top.

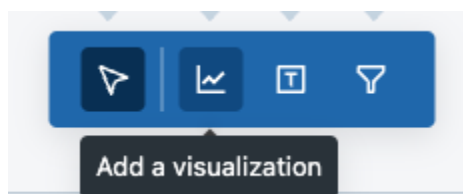
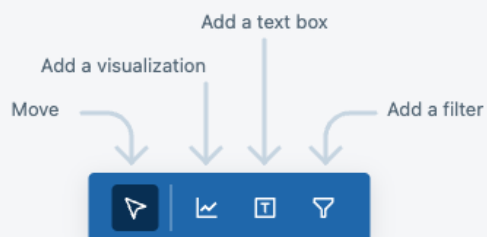
Canvas

Data

Now we'll add a visualization.

Add a visualization or filter to your canvas

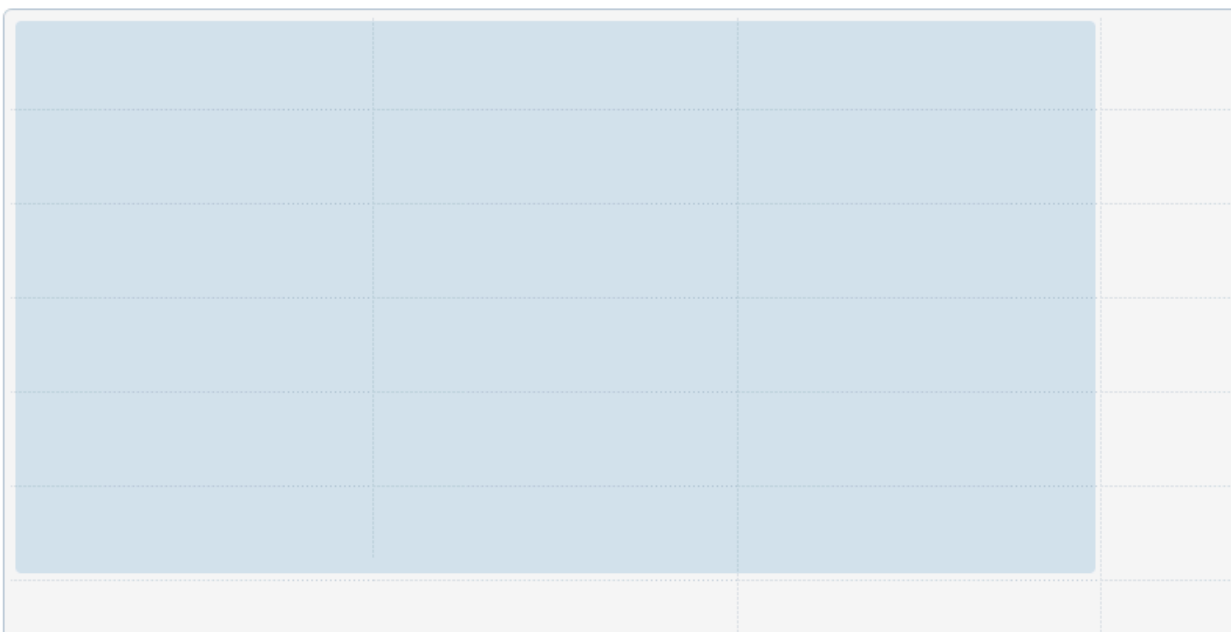
Select a tool then drag and draw to create your first dashboard widget.



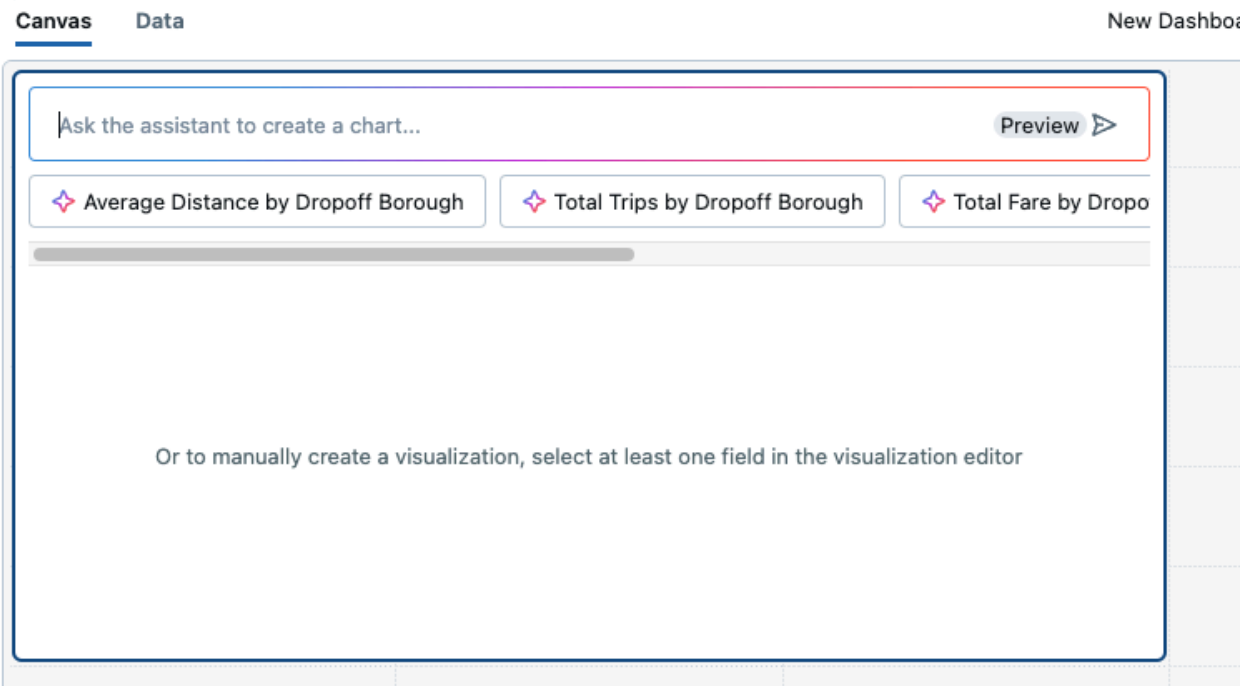
Canvas

Data

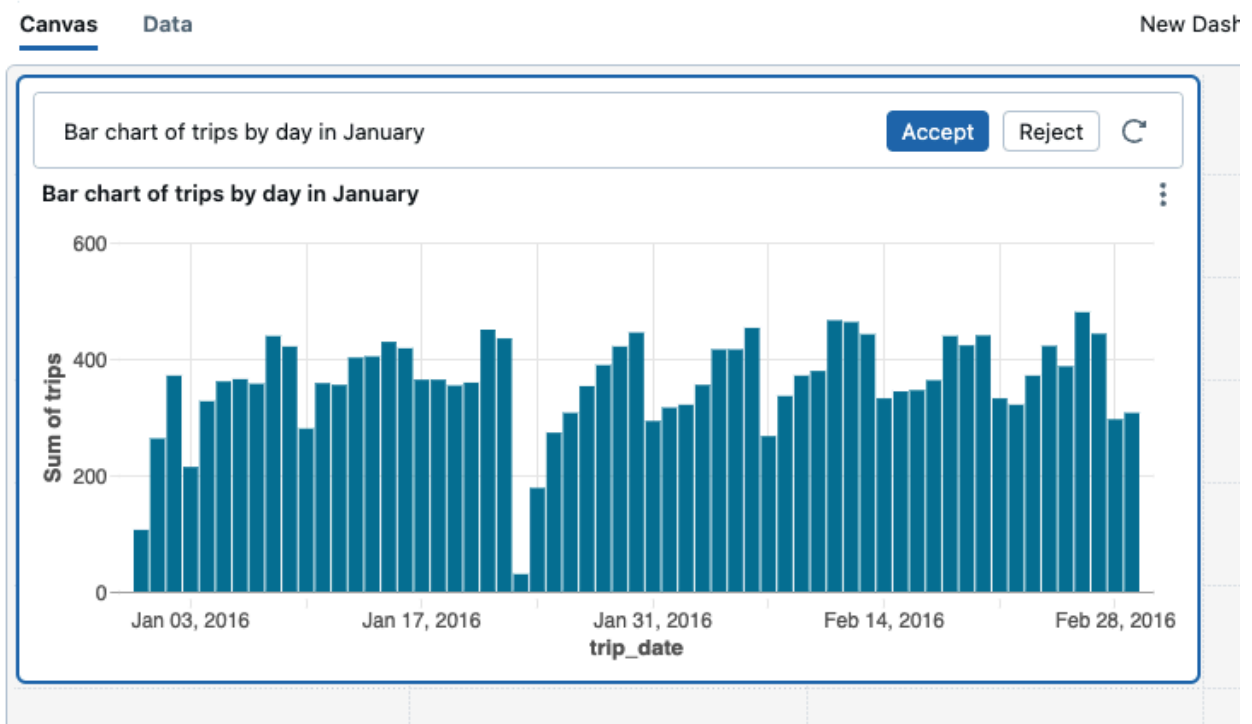
New Dashboard 2024



And after you click



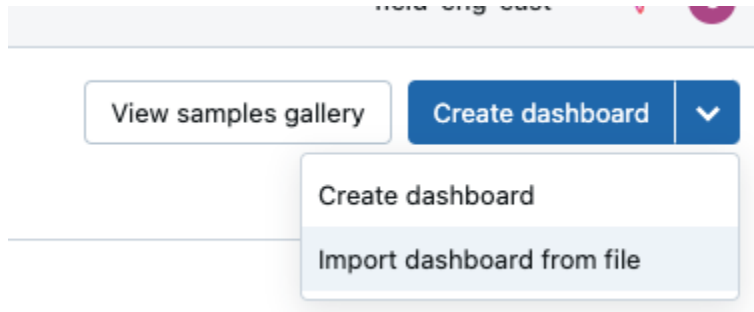
Type: "Bar chart of trips by day in January"



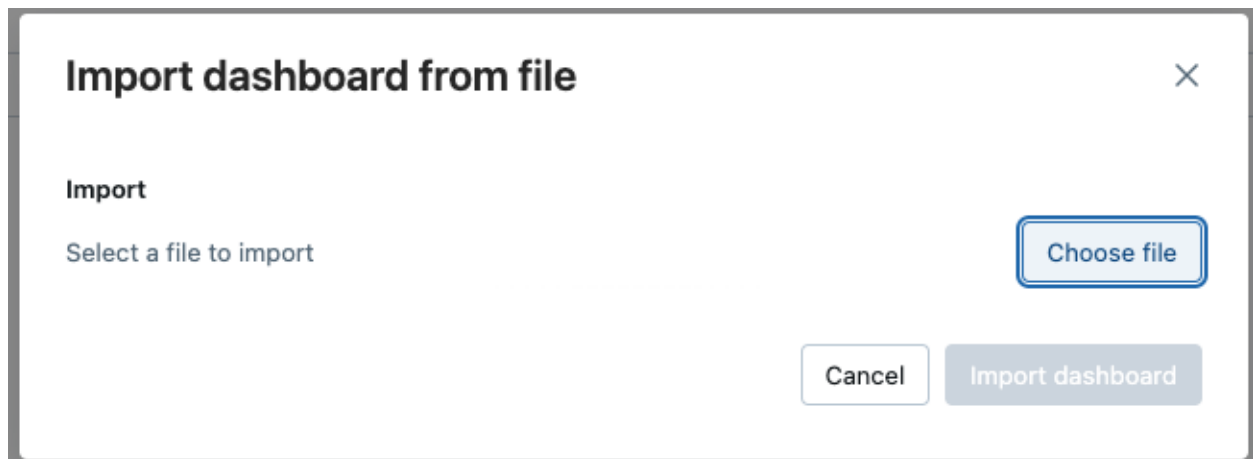
Then choose Accept. Your done!

Import Dashboard if you want to.

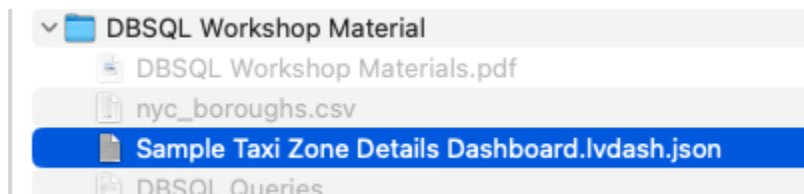
Follow these instructions.



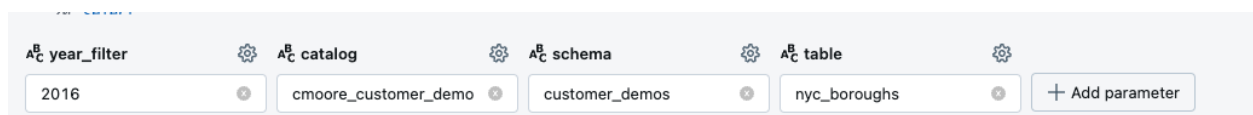
Select "Choose File"



Select the "Sample Taxi Zone Details Dashboard.lvdash.json" file



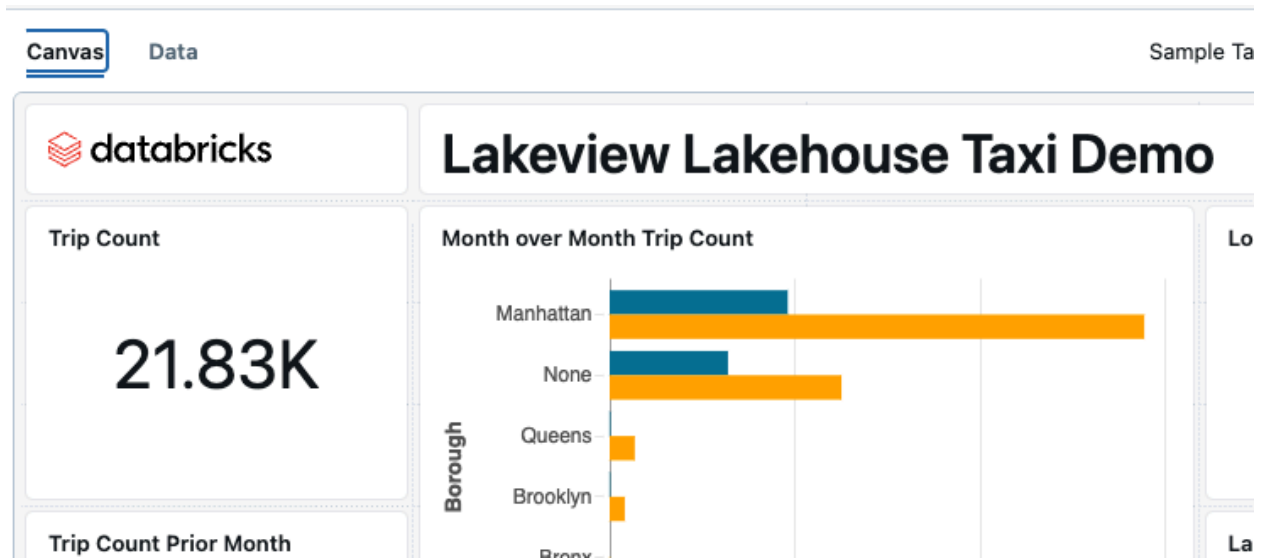
Alter the parameters in the Data Tab, run the query and confirm.



```
▶ Run just now

1 with trips_detailed as (
2   select a.*,
3   hour(to_timestamp(pickup_datetime))
```

Click on Canvas.



And your Dashboard should be live.