# To preview the whole data in the dataset

SELECT ride\_id, started\_at, ended\_at, ride\_length, day\_of\_week, start\_station\_name, end\_station\_name, member\_casual

FROM Cyclistic

ORDER BY ride\_id DESC



# Total Trips: Members vs Casual

# Looking at overall, annual member and casual rider totals

SELECT

TotalTrips,

TotalMemberTrips,

TotalCasualTrips,

ROUND(CAST(TotalMemberTrips AS FLOAT) / TotalTrips, 2) \* 100 AS MemberPercentage,

ROUND(CAST(TotalCasualTrips AS FLOAT) / TotalTrips, 2) \* 100 AS CasualPercentage

FROM

(

SELECT

COUNT(ride\_id) AS TotalTrips,

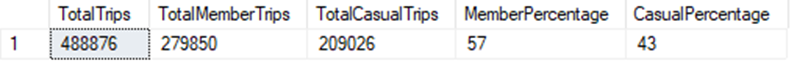
COUNT(CASE WHEN member\_casual = 'member' THEN 1 END) AS TotalMemberTrips,

COUNT(CASE WHEN member\_casual = 'casual' THEN 1 END) AS TotalCasualTrips

FROM

Cyclistic

) AS subquery;



# Average Ride Length

SELECT

AVG(DATEPART(MINUTE, CONVERT(datetime, '00:' + ride\_length)) \* 60 +

DATEPART(SECOND, CONVERT(datetime, '00:' + ride\_length))) / 60 AS AvgRideLength\_Overall

FROM Cyclistic;

SELECT

AVG(DATEPART(MINUTE, CONVERT(datetime, '00:' + ride\_length)) \* 60 +

DATEPART(SECOND, CONVERT(datetime, '00:' + ride\_length))) / 60 AS AvgRideLength\_Member

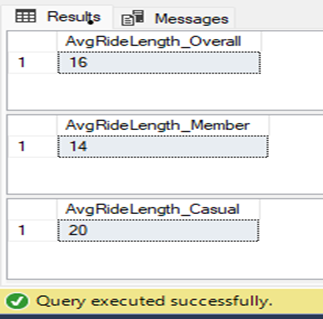
FROM Cyclistic where member\_casual = 'member'

SELECT

AVG(DATEPART(MINUTE, CONVERT(datetime, '00:' + ride\_length)) \* 60 +

DATEPART(SECOND, CONVERT(datetime, '00:' + ride\_length))) / 60 AS AvgRideLength\_Casual

FROM Cyclistic where member\_casual = 'casual'



# Maximum Ride Length - Members vs Casual

-- Looking at max ride lengths to check for outliers

SELECT TOP 2

member\_casual,

MAX(ride\_length) AS ride\_length\_MAX

FROM

Cyclistic

GROUP BY

member\_casual

ORDER BY

ride\_length\_MAX DESC;

# Maximum Ride Length - Members vs Casual

# Looking at max ride lengths to check for outliers

SELECT TOP 2

member\_casual,

MAX(ride\_length) AS ride\_length\_MAX

FROM

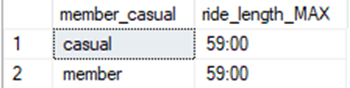
Cyclistic

GROUP BY

member\_casual

ORDER BY

ride\_length\_MAX DESC;



# Looking at top 100 longest trips for casual riders

SELECT TOP 100

member\_casual,

ride\_length

FROM

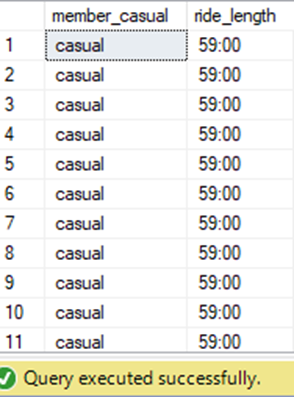
Cyclistic

WHERE

member\_casual = 'casual'

ORDER BY

ride\_length DESC;



# Busiest day for rides

# Looking at which days have the highest number of rides

SELECT TOP 2 WITH TIES

member\_casual,

day\_of\_week AS mode\_day\_of\_week -- Top number of day\_of\_week

FROM

(

SELECT

DISTINCT member\_casual, day\_of\_week, ROW\_NUMBER() OVER (PARTITION BY member\_casual ORDER BY COUNT(day\_of\_week) DESC) rn

FROM

Cyclistic

GROUP BY

member\_casual, day\_of\_week

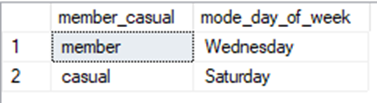
) subquery

WHERE

rn = 1

ORDER BY

member\_casual DESC;



# Total Rides Per Day

# Looking at total number of trips per day

SELECT TOP 7

day\_of\_week,

COUNT(DISTINCT ride\_id) AS TotalTrips,

SUM(CASE WHEN member\_casual = 'member' THEN 1 ELSE 0 END) AS MemberTrips,

SUM(CASE WHEN member\_casual = 'casual' THEN 1 ELSE 0 END) AS CasualTrips

FROM

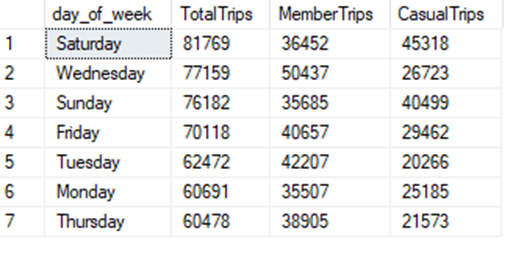
Cyclistic

GROUP BY

day\_of\_week

ORDER BY

TotalTrips DESC;



# Most popular start stations for trips

SELECT

DISTINCT start\_station\_name,

SUM(

CASE WHEN ride\_id = ride\_id AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS total,

SUM(

CASE WHEN member\_casual = 'member' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS member,

SUM(

CASE WHEN member\_casual = 'casual' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS casual

FROM

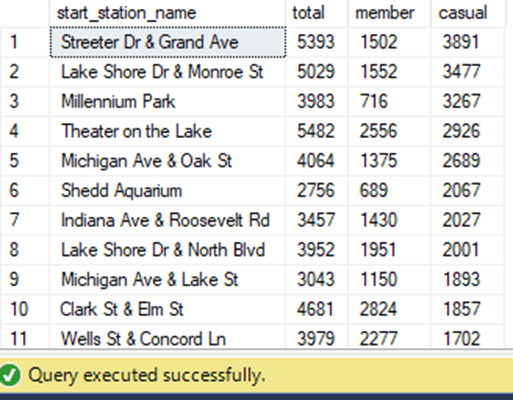
Cyclistic

GROUP BY

start\_station\_name

ORDER BY

total DESC, casual DESC, member DESC



# Top start Station name for casual riders

SELECT

DISTINCT start\_station\_name,

SUM(

CASE WHEN ride\_id = ride\_id AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS total,

SUM(

CASE WHEN member\_casual = 'member' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS member,

SUM(

CASE WHEN member\_casual = 'casual' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS casual

FROM

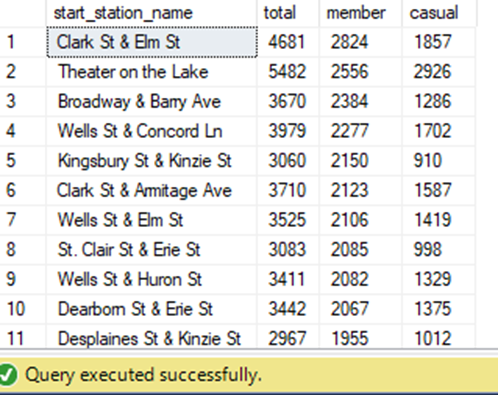
Cyclistic

GROUP BY

start\_station\_name

ORDER BY

casual DESC



# Top Start Station Name for Members

SELECT

DISTINCT start\_station\_name,

SUM(

CASE WHEN ride\_id = ride\_id AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS total,

SUM(

CASE WHEN member\_casual = 'member' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS member,

SUM(

CASE WHEN member\_casual = 'casual' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS casual

FROM

Cyclistic

GROUP BY

start\_station\_name

ORDER BY

member DESC

# Popular End Station Name

SELECT

DISTINCT start\_station\_name,

SUM(

CASE WHEN ride\_id = ride\_id AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS total,

SUM(

CASE WHEN member\_casual = 'member' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS member,

SUM(

CASE WHEN member\_casual = 'casual' AND start\_station\_name = start\_station\_name THEN 1 ELSE 0 END

) AS casual

FROM

Cyclistic

GROUP BY

start\_station\_name

ORDER BY

casual DESC

