GOOGLE DATA ANALYTICS CAPSTONE

darshan

2025-02-25

Introduction

In this case study, I will be assuming the role of a junior data analyst from a fictional company named "Cyclistic". In order to answer business questions, I will follow the steps of the data analysis process: Ask, Prepare, Process, Analyze, Share, Act. Owing to the data having over 5.8 million rows, I chose to go ahead with the Combining, Exploration, Cleaning and Analysis using SQL on BigQuery, while performing Visualizations on Tableau over spreadsheets.

Dataset

The data was obtained from the divvy-tripdata dataset, accessible from https://divvy-tripdata.s3.amazonaws.com/index.html

About the company

In 2016, Cyclistic launched a successful bike-share offering. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system anytime. Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes, and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

To be answered:

1. How do annual members and casual riders use Cyclistic bikes differently?
2. Why would casual riders buy Cyclistic annual memberships?
3. How can Cyclistic use digital media to influence casual riders to become members?

Data Combining

Following is the SQL query to combine all 12 files

```sql
-- Drop the table if it exists
DROP TABLE IF EXISTS `arcane-silo-451505-g2.biking_1.combined_data_1`;

-- Create and populate the table in a single step
CREATE TABLE `arcane-silo-451505-g2.biking_1.combined_data_1` AS
```

```sql
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_1`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_2`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_3`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_4`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_5`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_6`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_7`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_8`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_9`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_10`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_11`
UNION ALL
SELECT * FROM `arcane-silo-451505-g2.biking_1.biking_data_12`;


-- Count the number of rows in the newly created table = 5860568
SELECT COUNT(*) AS total_rows
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`;
```

Observations: The SQL code combines 12 divvy-tripdata files from the same year, creating a combined file with 5860568 rows.

Data Exploration

The following code is the SQL query for data exploration of the combined dataset:

```sql
SELECT COUNT(*) - COUNT(ride_id) AS ride_id,
       COUNT(*) - COUNT(rideable_type) AS rideable_type,
       COUNT(*) - COUNT(started_at) AS started_at,
       COUNT(*) - COUNT(ended_at) AS ended_at,
       COUNT(*) - COUNT(start_station_name) AS start_station_name,
       COUNT(*) - COUNT(start_station_id) AS start_station_id,
       COUNT(*) - COUNT(end_station_name) AS end_station_name,
       COUNT(*) - COUNT(end_station_id) AS end_station_id,
       COUNT(*) - COUNT(start_lat) AS start_lat,
       COUNT(*) - COUNT(start_lng) AS start_lng,
       COUNT(*) - COUNT(end_lat) AS end_lat,
       COUNT(*) - COUNT(end_lng) AS end_lng
       FROM `arcane-silo-451505-g2.biking_1.combined_data_1;
       --no of null values--
```

```sql
SELECT COUNT(ride_id) - COUNT (DISTINCT ride_id) AS duplicate_rows
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`;
--211 duplicate rows--

SELECT LENGTH(ride_id) as length_ride_id, COUNT(ride_id) as no_of_rows
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
GROUP BY ride_id;
--checking if length is 16 for ride ids--

SELECT DISTINCT(rideable_type) AS rideable_type, COUNT(rideable_type) AS no_of_trips
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
GROUP BY rideable_type;
--3 unique types of bikes--

SELECT started_at, ended_at
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
LIMIT 50;

SELECT COUNT(ride_id) as more_than_one_day
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE (
  EXTRACT(HOUR FROM ended_at - started_at)*60 +
  EXTRACT(MINUTE FROM ended_at - started_at) +
  EXTRACT(SECOND FROM ended_at - started_at)/60
) >= 1440;
--7596 trips longer than a day--


SELECT COUNT(ride_id) AS less_than_a_minute
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE(
  EXTRACT(HOUR from ended_at - started_at)*60 +
  EXTRACT(MINUTE FROM ended_at - started_at) +
  EXTRACT(SECOND FROM ended_at - started_at)/60
) <= 1;
--132644 trips less than a minute--


SELECT DISTINCT start_station_name
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
ORDER BY start_station_name;
--49 different start stations--

SELECT COUNT(ride_id) AS start_station_null
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE start_station_name IS NULL OR start_station_id IS NULL;
--no of null start stations is 1073951--

SELECT DISTINCT end_station_name
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
ORDER BY end_station_name;
--49 different end stations--
```

```sql
SELECT COUNT(ride_id) AS end_station_null
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE end_station_name IS NULL OR end_station_id IS NULL;
--no of null end stations is 1104653--

SELECT COUNT(ride_id) AS start_loc_null
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE start_lat IS NULL OR start_lng IS NULL;
--no of null start locations is 0

SELECT COUNT(ride_id) AS end_loc_null
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE end_lat IS NULL OR end_lng IS NULL;
--no of null end locations is 7232

SELECT COUNT(ride_id) AS no_casual
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE member_casual = 'casual';
--no of casual riders is 2151658

SELECT COUNT(ride_id) AS no_member
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE member_casual = 'member';
--no of members is 3708910

SELECT COUNT(ride_id) AS member_null
FROM `arcane-silo-451505-g2.biking_1.combined_data_1`
WHERE member_casual IS NULL;
--no null assignments for member/casual
```

Nulls:

1. There are 0 null values in the column ride_id
2. There are 0 null values in the column rideable_type
3. There are 0 null values in the column started_at
4. There are 0 null values in the column ended_at
5. There are 1073951 null values in the column start_station_name
6. There are 1073951 null values in the column start_station_id
7. There are 1104653 null values in the column end_station_name
8. There are 1104653 null values in the column end_station_id
9. There are 0 null Values in the columns start_lat and start_lng
10. There are 7232 null values in the columns end_lat and end_lng

There are 211 Duplicate rows

Length for all ride IDs is 16

There are 3 unique types of bikes - Classic Bike, Electric Bike and Electric Scooter.

There are 7596 trips that take longer than a day.

There are 132644 trips that take less than a minute.

There are 49 different start stations.

There are 49 different end stations.

2151658 casual riders.

3708910 members.

0 null values in the member_casual column.

Data Cleaning

The SQL queries to clean data are:

```
DROP TABLE IF EXISTS `arcane-silo-451505-g2.biking_1.cleaned_combined`;

CREATE TABLE IF NOT EXISTS `arcane-silo-451505-g2.biking_1.cleaned_combined` AS (
SELECT a.ride_id, rideable_type, started_at, ended_at, ride_length,
       CASE EXTRACT(DAYOFWEEK FROM started_at)
       WHEN 1 THEN'SUN'
       WHEN 2 THEN'MON'
       WHEN 3 THEN'TUES'
       WHEN 4 THEN 'WED'
       WHEN 5 THEN 'THUR'
       WHEN 6 THEN'FRI'
       WHEN 7 THEN 'SAT'
       END AS day,
       CASE EXTRACT(MONTH FROM started_at)
       WHEN 1 THEN 'JAN'
       WHEN 2 THEN 'FEB'
       WHEN 3 THEN 'MAR'
       WHEN 4 THEN 'APR'
       WHEN 5 THEN 'MAY'
       WHEN 6 THEN 'JUN'
       WHEN 7 THEN 'JUL'
       WHEN 8 THEN 'AUG'
       WHEN 9 THEN 'SEPT'
       WHEN 10 THEN 'OCT'
       WHEN 11 THEN 'NOV'
       WHEN 12 THEN 'DEC'
       END AS month, start_station_name, start_station_id, end_station_name,end_station_id, start_lat, s
       end_lat, end_lng, member_casual
       FROM `arcane-silo-451505-g2.biking_1.combined_data_1` as a --entering data as a
       JOIN(SELECT ride_id,
            (EXTRACT(HOUR FROM (ended_at-started_at)*60) +
             EXTRACT(MINUTE FROM (ended_at-started_at)) +
             EXTRACT(SECOND FROM (ended_at-started_at)/60)) AS ride_length
        FROM `arcane-silo-451505-g2.biking_1.combined_data_1`)
```

```
      AS b --having ride length as b

      ON a.ride_id = b.ride_id  --joining a and b
      WHERE start_station_name IS NOT NULL AND
      start_station_id IS NOT NULL AND
      end_station_name IS NOT NULL AND
      end_station_id IS NOT NULL AND
      start_lat IS NOT NULL AND
      start_lng IS NOT NULL AND
      end_lat IS NOT NULL AND
      end_lng IS NOT NULL AND
      member_casual IS NOT NULL AND
      ride_length > 1 AND ride_length < 1440);


SELECT COUNT(*) as no_of_rows
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
--returned 4168618 rows
```

Process:

Creates new table named "cleaned_combined"

Creates a subquery to create new table with columns ride_id, rideable_type, started_at, ended_at, ride_length, start_station_name, start_station_id, end_station_name,end_station_id, start_lat, start_lng, end_lat, end_lng, member_casual and creating 2 new columns day and month, with values extracted from started_at.

Assigned the table to a.

Using JOIN statement to join ride_id from the dataset with new column ride_length, containing duration of rides, calculated by extracting values from differences of ended_at and started_at, and converting to minutes.

Assigned joined columns to b.

Using ON statement for a.ride_id and b.ride_id to merge all data, with WHERE statement ensuring no null values are entered, and ride duration is more than a minute but less than a day.

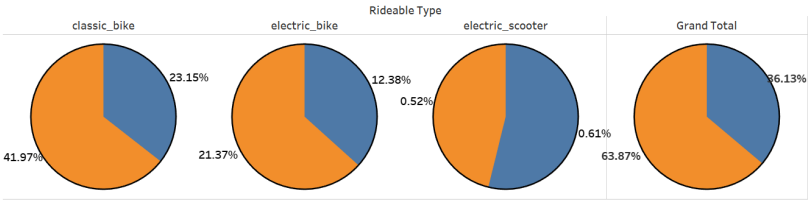
4168618 rows have been returned. 1691950 rows have been removed.

Data Analysis and Visualisation

The SQL queries for data analysis are:

```
SELECT member_casual, rideable_type, COUNT(*) AS total_trips
 FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
 GROUP BY member_casual, rideable_type
 ORDER BY member_casual, total_trips;
```

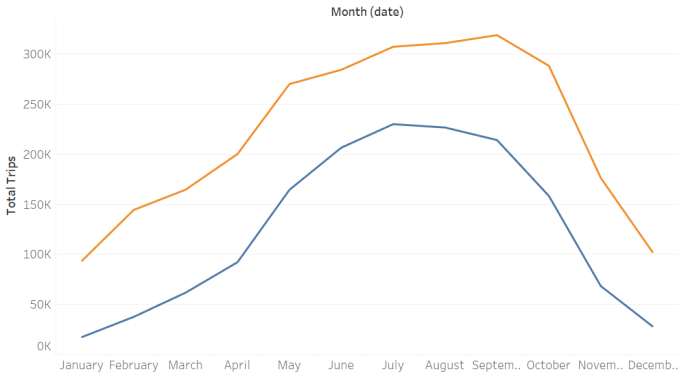
Total Casual/Members and different bike types



63.87% of riders are members. 36.13% percent of riders are casual riders. The most used bike is the classic bike, followed by the electric bike. Percentages based on bike types are present in the chart.

```sql
SELECT month, member_casual, COUNT(ride_id) AS total_trips
 FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
 GROUP BY month, member_casual
 ORDER BY member_casual;
```

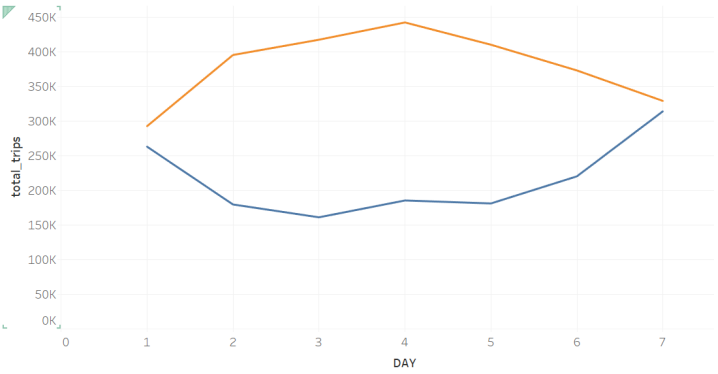
RIDES PER MONTH



The total number of trips is highest in July for Casual riders, with a peak of 230145. For members, the peak is in September with a number of 319089. The total number of trips increases once summer begins, and gradually goes down past September. The data indicates that members have higher bike usage than casual riders throughout the year.

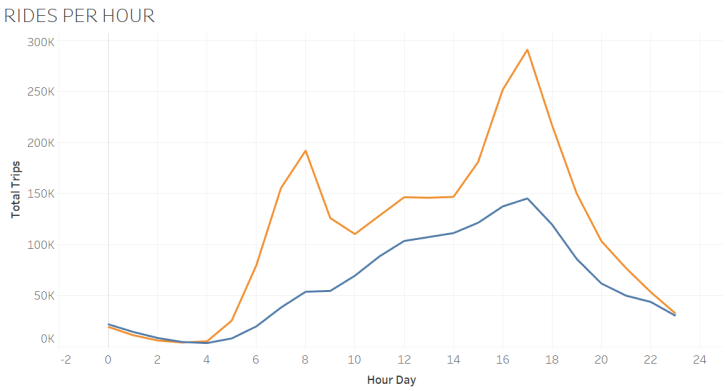
```sql
SELECT day, member_casual, COUNT(ride_id) AS total_trips
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY day, member_casual
ORDER BY member_casual;
```

RIDES PER DAY



Comparing weekly data, Members seem to have higher usage on weekdays when compared to the weekends in contrast to Casual riders, who tend to use bikes more on the weekends than they do on weekdays. The number of trips reaches a peak of 442737 on Wednesdays for Members and 314445 on Saturdays for Casual riders. Members have higher usage throughout the week, with the gap being the least on Saturdays.

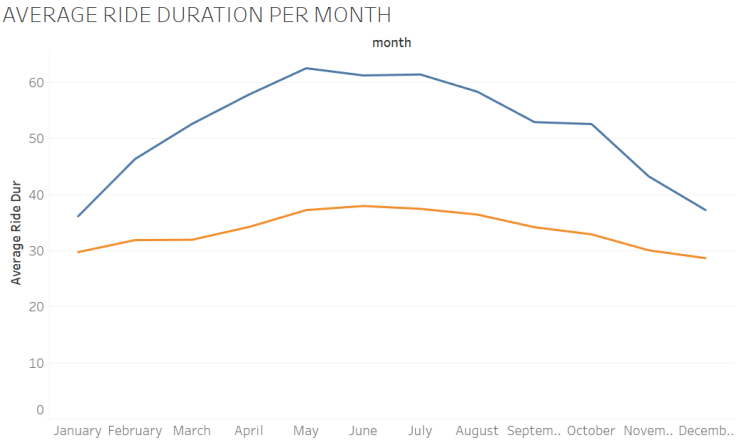
```
SELECT EXTRACT(HOUR FROM started_at) AS hour_day, member_casual, COUNT(ride_id) AS total_trips
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY hour_day, member_casual
ORDER BY member_casual;
```



Comparing hourly data, it can be seen that the members have 2 peaks throughout the day, one at around 8 am and one at around 5 pm, after which the number of rides decreases. For Casuals, the number of rides increases throughout the day, reaching a peak at 5 pm, and then decreasing afterwards.

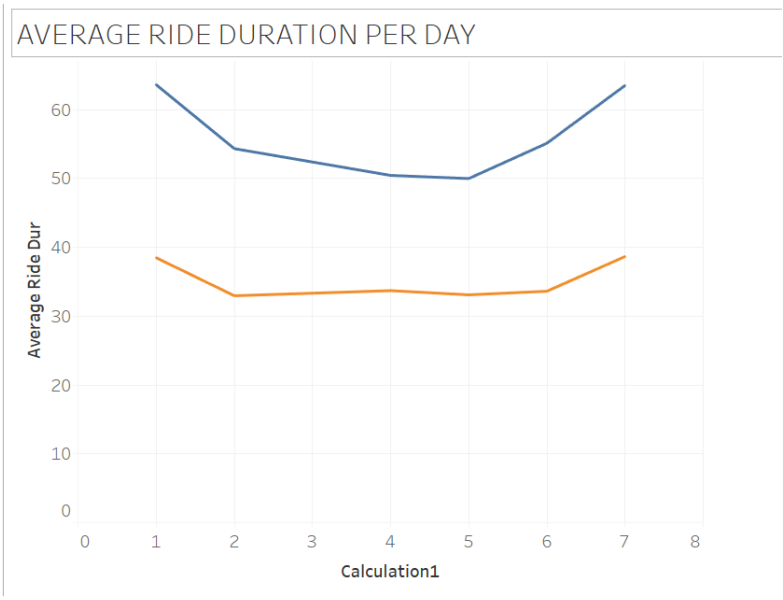
From this data, it can be inferred that the members are using bikes to travel to their workplaces and back owing to the high usage in the mornings and evenings, and to the higher ride count on weekdays.

```
SELECT month, member_casual, avg(ride_length) AS average_ride_dur
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY month, member_casual;
```



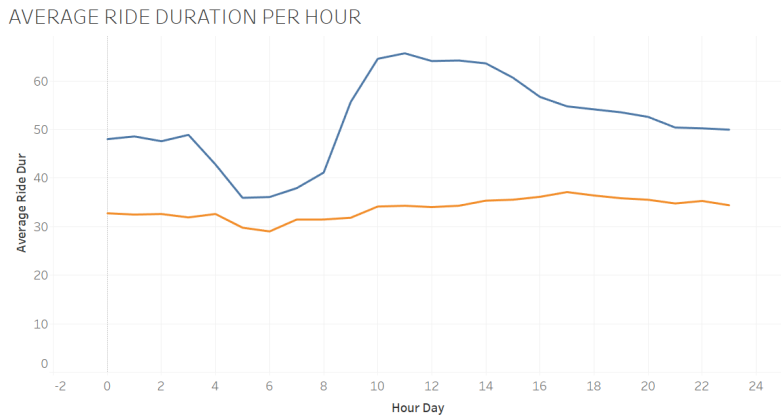
The data shows that despite the lesser number of rides for Casuals, they tend to have a longer ride duration. The average ride duration for Casuals remains comparatively higher than that of Members' throughout the year. It can be seen that the ride duration for Casuals peaks in the Summer, in the months of May, June and July, after which it gradually decreases. The ride duration for Members remains mostly steady throughout the year, with a slight bump in the summer. The average ride duration peaked in May with a value of 62.56 minutes for Casuals and in June with a value of 37.99 minutes for Members.

```sql
SELECT day, member_casual, avg(ride_length) AS average_ride_dur
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY day, member_casual;
```



The Casuals have higher bike usage throughout the week, with the longest ride duration on weekends. Members have steady bike usage throughout the week, without much variation in their ride duration but with a slight bump on the weekend.

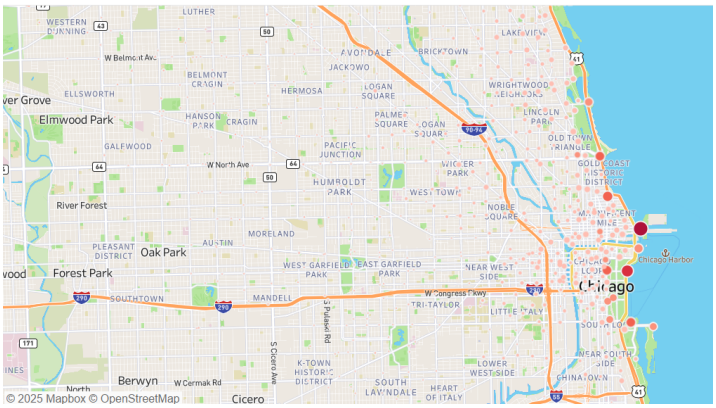
```sql
SELECT EXTRACT(HOUR FROM started_at) AS hour_day, member_casual, avg(ride_length) AS average_ride_dur
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY hour_day, member_casual;
```



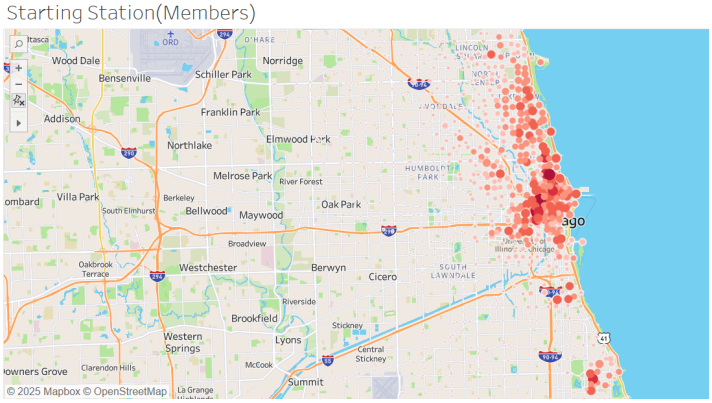
The Casuals tend to use bikes more during the day, with their longest trips happening between 10 AM to 2 PM. Members have steady bike usage throughout the day, without much variation in their ride duration. Due to the higher ride duration of Casuals on weekends and in the day outside of commuting hours, it can be inferred that they're using bikes for recreational purposes.

```sql
SELECT start_station_name, member_casual, avg(start_lat) AS lat, avg(start_lng) as lng, COUNT(ride_id)
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY start_station_name, member_casual;
```

Starting Station(Casuals)



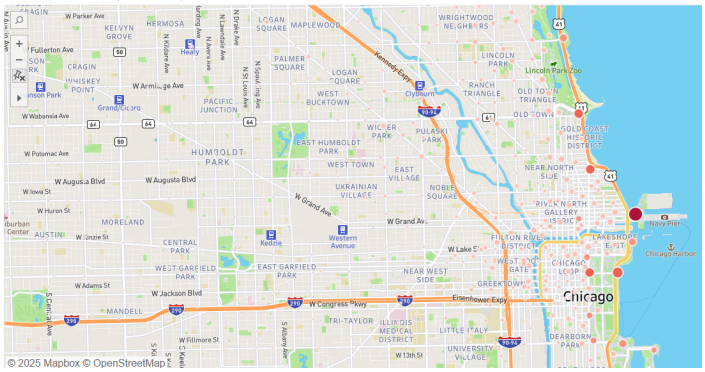
Starting Station(Members)



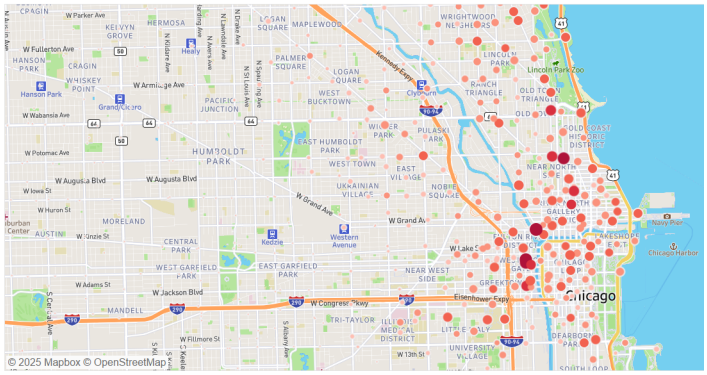
It can be seen from the map that Casuals usually start their trips close to the harbor or parks whereas Members start their trips closer to areas with company offices. The most popular starting station for Members is Kingsbury St and Kinzie St, with 26,601 trips, whereas the most popular station for Casuals is Streeter Dr and Grand Ave, with 47,852 trips.

```
SELECT end_station_name, member_casual, avg(end_lat) AS lat, avg(end_lng) AS lng, COUNT(ride_id) AS tota
FROM `arcane-silo-451505-g2.biking_1.cleaned_combined`
GROUP BY end_station_name, member_casual;
```

End Station(Casual)



End Station(Member)



The end stations follow a similar trend, confirming that Casuals use bikes for recreational activities whereas Members use bikes for daily commute to their workplaces/institutions. The most popular end station for Members is Kingsbury St and Kinzie St, with 26,720 trips. The most popular end station for Casuals is Streeter Dr and Grand Ave, with 51,979 trips.

Summary

Casuals:

1. Tend to be more active on the weekends, in Summer and outside of commute hours.
2. Focus more on recreational activities.
3. Ride for longer periods of time, but with a lesser number of rides.

Members:

1. Tend to be more active on weekdays, during commute hours in the Summer.
2. Tend to focus more on commuting to workplaces.
3. Have a higher number of rides, with shorter ride duration.

Suggestions

Offer weekend-only membership plans that will attract more casual riders.

Offer discounts for people with higher ride duration, encouraging Casual riders and pushing Members to ride longer.

Free trial runs offered at stations closest to harbors/parks will increase popularity amongst Casuals.

Offer members priority access to bikes in high traffic locations.

Create short term membership plans for tourists.

Add a Summer membership plan to make rides cost efficient and convenient for Casuals.