Cyclistic Bike-Share: Understanding Member vs. Casual Rider Behavior

Introduction

This case study is a part of the [Google Data Analytics Professional Certificate](). In this project, I have performed the tasks of a junior data analyst at a fictional company that operates a bike-sharing program called Cyclistic, in Chicago. I have followed the steps involved in the six phases of the data analysis process: Ask, Prepare, Process, Analyze, Share, and Act to complete the project end-to-end.

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. Cyclistic's finance analysts have concluded that annual members are much more profitable than casual riders. Although the pricing flexibility helps Cyclistic attract more customers, Moreno believes that maximizing the number of annual members will be key to future growth.

Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a solid opportunity to convert casual riders into members. She notes that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs. Moreno has set a clear goal: Design marketing strategies aimed at converting casual riders into annual members. In order to do that, however, the team needs to better understand how annual members and casual riders differ, why casual riders would buy a membership, and how digital media could affect their marketing tactics. Moreno and her team are interested in analyzing the Cyclistic historical bike trip data to identify trends.

Scenario

I am a junior data analyst working on the marketing analyst team at Cyclistic, a bike-share company in Chicago. The director of marketing believes the company's future success depends on maximizing the number of annual memberships. Therefore, my team wants to understand how casual riders and annual members use Cyclistic bikes differently. From these insights, my team will design a new marketing strategy to convert casual riders into annual members. But first, Cyclistic executives must approve my recommendations, so they must be backed up with compelling data insights and professional data visualizations.

Characters & Team

- Cyclistic: A bike-share program that features more than 5,800 bicycles and 600 docking stations. Cyclistic sets itself apart by also offering reclining bikes, hand tricycles, and cargo bikes, making bike-share more inclusive to people with disabilities and riders who can't use a standard two-wheeled bike. The majority of riders opt for traditional bikes; about 8% of riders use the assistive options. Cyclistic users are more likely to ride for leisure, but about 30% use the bikes to commute to work each day.
- Lily Moreno: The director of marketing and your manager. Moreno is responsible for the development of campaigns and initiatives to promote the bike-share program. These may include email, social media, and other channels.
- Cyclistic Marketing Analytics Team: A team of data analysts who are responsible for collecting, analyzing, and reporting data that helps guide Cyclistic marketing strategy. You joined this team six months ago and have been busy learning about Cyclistic's mission and business goals—as well as how you, as a junior data analyst, can help Cyclistic achieve them.

- Cyclistic Executive Team: The notoriously detail-oriented executive team will decide whether to approve the recommended marketing program.

Ask Phase

Business Task

The marketing director, Lily Moreno, wants to understand how annual members and casual riders use Cyclistic bikes differently. The goal is to design a targeted marketing strategy that converts more casual riders into annual members to support long-term business growth and increase revenue.

As a junior data analyst on the marketing analytics team, I have been tasked with performing a detailed analysis of rider behavior to uncover patterns and differences between these two user groups.

Stakeholders

Stakeholder	Role
Lily Moreno	Director of Marketing; responsible for campaign planning
Marketing Analytics Team	Collects, analyzes, and presents data for decision-making
Executive Team	Approves strategic plans based on insights
Casual Riders	Potential audience for conversion into long-term annual memberships

---

Guiding Questions

- What are the behavioral patterns of annual members vs. casual riders?
- When, where, and how often do each group ride?
- What user types or segments exist within the casual rider group?
- How can Cyclistic use these insights to tailor a marketing strategy?
- What factors might encourage casual riders to become annual members?

Gap Analysis

Current State	Desired State	What's Missing (Gap)
Casual riders use bikes irregularly	More casuals convert to members	No clear understanding of their usage behavior
One-size-fits-all marketing	Targeted campaigns by behavior/needs	No segmentation of user types based on data
Limited retention and recurring revenue	Reliable member base with predictable income	Strategy to drive loyalty and long-term value

---

Summary

Cyclistic is in a strong position with a wide network of bikes and inclusive options. However, to drive membership and long-term value, the company must understand what motivates different users. This analysis will bridge that gap and guide the development of a data-informed marketing plan focused on rider conversion and retention.

Prepare Phase

In the Prepare phase, the objective is to assess the availability, structure, and reliability of the dataset that will be used to analyze user behavior differences between annual members and casual riders in the Cyclistic bike-share program.

Data Source

The dataset used in this project is publicly provided by Motivate International Inc., which operates the Divvy bike-share program in Chicago. It includes detailed ride data for the year 2025. The raw data was downloaded as CSV files from the Divvy System Data Portal and uploaded to Google BigQuery for efficient querying and analysis.

Data Schema & Structure

Upon uploading the dataset to BigQuery, the schema was automatically inferred, and the structure was reviewed. The table contains 13 fields, each representing key aspects of a bike trip. Below is an overview of the fields:

Schema Details Preview Table Explorer Preview Insights Lineage Data Profile Data Quality

Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
ride_id	STRING	NULLABLE	-	-	-	-	-
rideable_type	STRING	NULLABLE	-	-	-	-	-
started_at	TIMESTAMP	NULLABLE	-	-	-	-	-
ended_at	TIMESTAMP	NULLABLE	-	-	-	-	-
start_station_name	STRING	NULLABLE	-	-	-	-	-
start_station_id	STRING	NULLABLE	-	-	-	-	-
end_station_name	STRING	NULLABLE	-	-	-	-	-
end_station_id	STRING	NULLABLE	-	-	-	-	-
start_lat	FLOAT	NULLABLE	-	-	-	-	-
start_lng	FLOAT	NULLABLE	-	-	-	-	-
end_lat	FLOAT	NULLABLE	-	-	-	-	-
end_lng	FLOAT	NULLABLE	-	-	-	-	-
member_casual	STRING	NULLABLE	-	-	-	-	-

- ride_id (STRING): A unique identifier for each bike ride. Essential for tracking and referencing individual records.
- rideable_type (STRING): Indicates the type of bike used (e.g., classic, electric, docked). Useful for analyzing preference and trends.
- started_at / ended_at (TIMESTAMP): The start and end times of the ride. These fields help in calculating ride duration and usage patterns across time.
- start_station_name / end_station_name (STRING): Names of the stations where the ride began and ended. Helps in mapping routes and station popularity.
- start_station_id / end_station_id (STRING): Station identifiers (may have missing or inconsistent values in some datasets).
- start_lat / start_lng / end_lat / end_lng (FLOAT): Geographic coordinates of the start and end points. Useful for mapping and spatial analysis.
- member_casual (STRING): Labels the rider as either a member or casual user. This is the key field for answering the main business question about user behavior differences.

The data types were automatically inferred by BigQuery and appear appropriate for the intended analysis. Some fields may contain null or inconsistent entries (especially station names/IDs), which will be handled in the next phase (Process). Overall, the schema provides a rich foundation for behavioral segmentation and usage analysis.

Data Snapshot

```sql
SELECT * FROM
winter-legend-mg.cyclist.cycle_trip
LIMIT 100;
```

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	ride_id	rideable_type	started_at	ended_at	start_station_name	start_station_id	end_statio	end_statior	start_lat	start_lng	end_lat	end_lng	member_casual
1	D5184CBF80C1D...	classic_bike	2025-03-08 14:53:47.5...	2025-03-09 16:53:...	Altgeld Gardens	20212	null	null	41.6566655296...	-87.5988006591...	null	null	casual
2	A0AF9478B77DB...	classic_bike	2025-03-14 08:42:47.5...	2025-03-15 09:42:...	Kedzie Ave & 110t...	20204	null	null	41.6923947458...	-87.7009606361...	null	null	casual
3	8158AB87A3C24...	classic_bike	2025-03-14 16:12:15.6...	2025-03-15 17:11:...	Vincennes Ave & 1...	20124	null	null	41.7049255649...	-87.6568681095...	null	null	casual
4	FD98A870677AD...	classic_bike	2025-03-15 15:59:51.7...	2025-03-16 16:59:...	Western Ave & 101...	24454	null	null	41.71011	-87.68198	null	null	casual
5	FCDC6D52C70B...	classic_bike	2025-03-16 09:29:27.6...	2025-03-17 10:29:...	State St & 95th St	20104	null	null	41.7218499	-87.6228544	null	null	casual
6	AEE64C5842C0E...	classic_bike	2025-03-20 15:48:38.8...	2025-03-21 16:48:...	Phillips Ave & 83rd...	582	null	null	41.744531	-87.56506	null	null	casual
7	F837FF70199A7...	classic_bike	2025-02-28 14:02:54.5...	2025-03-01 15:02:...	Phillips Ave & 83rd...	582	null	null	41.744531	-87.56506	null	null	casual
8	7875AAA268E90...	classic_bike	2025-03-13 18:48:47.4...	2025-03-14 19:48:...	Ashland Ave & 78t...	20234	null	null	41.7522048494...	-87.6636993885...	null	null	casual
9	1CC44B68FF209...	classic_bike	2025-03-29 11:42:48.9...	2025-03-30 12:42:...	Stony Island Ave & ...	KA15030000...	null	null	41.7664929373	-87.5864608775	null	null	casual
10	8DF95C703D1C5...	classic_bike	2025-03-10 17:28:41.2...	2025-03-11 18:28:...	Central Park Ave & ...	24144	null	null	41.773258	-87.713143	null	null	casual
11	CBF477999F986...	classic_bike	2025-03-21 13:13:11.0...	2025-03-22 14:12:...	Jeffery Blvd & 67th...	KA15030000...	null	null	41.77351755125	-87.5771428255	null	null	casual
12	46B8AE2463DEF...	classic_bike	2025-03-05 12:40:10.1...	2025-03-06 13:40:...	Cottage Grove Ave ...	KA15030000...	null	null	41.7805309407...	-87.6059702038...	null	null	casual
13	D1A06D591EC0D...	classic_bike	2025-03-28 08:14:28.3...	2025-03-28 10:13:...	Midway Orange Line	24289	null	null	41.78692	-87.73892	null	null	casual
14	E12F8B6CFAEFD...	classic_bike	2025-03-09 14:35:17.2...	2025-03-10 15:35:...	Blackstone Ave & ...	22004	null	null	41.787877	-87.590461	null	null	casual
15	ACBCC7243600A...	classic_bike	2025-03-27 16:08:21.0...	2025-03-28 17:08:...	Harper Ave & 59th ...	KA15030000...	null	null	41.78794281287	-87.5883151702	null	null	casual
16	76D63C144FEFA...	classic_bike	2025-03-15 11:11:33.1...	2025-03-16 12:11:...	Harper Ave & 59th ...	KA15030000...	null	null	41.78794281287	-87.5883151702	null	null	member
17	283D24C6A74A4...	classic_bike	2025-03-18 19:34:01.9...	2025-03-19 20:33:...	Kedzie Ave & 57th ...	21346	null	null	41.78983	-87.7031	null	null	casual
18	0117EF4AD0C83...	classic_bike	2025-03-11 17:29:13.5...	2025-03-12 18:29:...	Kedzie Ave & 57th ...	21346	null	null	41.78983	-87.7031	null	null	casual
19	F272DB5D7CF75...	classic_bike	2025-03-27 15:23:07.1...	2025-03-28 16:22:...	University Ave & 57...	KA15030000...	null	null	41.791478	-87.599861	null	null	casual

This is a snapshot of the first few rows of the cyclistic dataset.

Exploring Column Metadata

To ensure the dataset structure aligns with the business task, I examined the table schema using BigQuery's metadata query system. The following SQL query was run against the INFORMATION_SCHEMA.COLUMNS view to list all columns and their data types for the cycle_trip table.

```sql
SELECT
  column_name,
  data_type
FROM
  `winter-legend-mg.cyclist.INFORMATION_SCHEMA.COLUMNS`
WHERE
  table_name = 'cycle_trip';
```

Job information | Results | Chart | JSON | Execution de

Row	column_name	data_type
1	ride_id	STRING
2	rideable_type	STRING
3	started_at	TIMESTAMP
4	ended_at	TIMESTAMP
5	start_station_name	STRING
6	start_station_id	STRING
7	end_station_name	STRING
8	end_station_id	STRING
9	start_lat	FLOAT64
10	start_lng	FLOAT64
11	end_lat	FLOAT64
12	end_lng	FLOAT64
13	member_casual	STRING

Data Quality Checks

A series of SQL queries were performed to identify missing values, data inconsistencies, and category mismatches. These checks help ensure data integrity and reliability.

Finding Missing/NULL Values

```
SELECT
COUNT(*) AS total_rows,
COUNTIF(ride_id IS NULL) AS null_ride_id,
COUNTIF(rideable_type IS NULL) AS null_rideable_type,
COUNTIF(started_at IS NULL) AS null_started_at,
```

```sql
COUNTIF(ended_at IS NULL) AS null_ended_at,
COUNTIF(start_station_name IS NULL)
AS null_start_station_name,
COUNTIF(start_station_id IS NULL)
AS null_start_station_id,
COUNTIF(end_station_name IS NULL)
AS null_end_station_name,
COUNTIF(end_station_id IS NULL)
AS null_end_station_id,
COUNTIF(start_lat IS NULL)
AS null_start_lat,
COUNTIF(start_lng IS NULL) AS null_start_lng,
COUNTIF(end_lat IS NULL) AS null_end_lat,
COUNTIF(end_lng IS NULL) AS null_end_lng,
COUNTIF(member_casual IS NULL)
AS null_member_casual
FROM
Winter-legend-mg.cyclist.cycle_trip;
```

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	total_rows	null_ride_id	null_rideable_type	null_started_at	null_ended_at	null_start_station_na	null_start_station_id	null_end_station_nan	null_end_station_id	null_start_lat	null_start_lng
1	298155	0	0	0	0	55337	55337	57984	57984	0	(

There are total 298155 rows in this dataset and four columns which have null values. These columns are : start_station_name, start_station_id, end_station_name, end_station_id, end_lat, end_lng.

null_start_lat	null_start_lng	null_end_lat	null_end_lng	null_member_casual
0	0	241	241	0

The above screenshots display the result of a SQL query executed in BigQuery to assess the presence of missing (NULL) values across all key columns in the dataset.

Each column was checked individually using the COUNTIF(column IS NULL) condition. This evaluation helps identify potential data quality issues that may require cleaning or special handling in later stages of the analysis. As shown, some fields such as start_station_name and end_station_name contain notable missing values, which may influence route-based insights or mapping. This step is essential to ensure data integrity before proceeding to analysis.

Vertical Format

```
SELECT field, null_count
FROM (
SELECT
COUNT(*) AS total_rows,
[
STRUCT('ride_id' AS field,
COUNTIF(ride_id IS NULL) AS null_count),
STRUCT('rideable_type',
COUNTIF(rideable_type IS NULL)),
STRUCT('started_at',
COUNTIF(started_at IS NULL)),
STRUCT('ended_at',
COUNTIF(ended_at IS NULL)),
STRUCT('start_station_name',
COUNTIF(start_station_name IS NULL)),
STRUCT('start_station_id',
COUNTIF(start_station_id IS NULL)),
STRUCT('end_station_name',
COUNTIF(end_station_name IS NULL)),
STRUCT('end_station_id',
COUNTIF(end_station_id IS NULL)),
STRUCT('start_lat', COUNTIF(start_lat IS NULL)),
STRUCT('start_lng', COUNTIF(start_lng IS NULL)),
STRUCT('end_lat', COUNTIF(end_lat IS NULL)),
STRUCT('end_lng', COUNTIF(end_lng IS NULL)),
STRUCT('member_casual', COUNTIF(member_casual IS NULL)
] AS nulls_array
FROM `winter-legend-mg.cyclist.cycle_trip`),
UNNEST(nulls_array) AS field_struct;
```

Query results

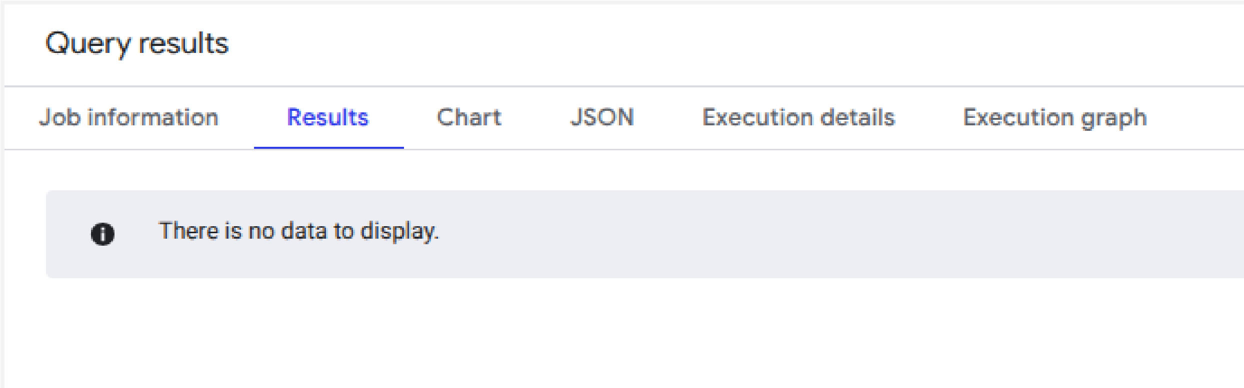
Job information | Results | Chart | JSON | Execution

Row	field	null_count
1	ride_id	0
2	rideable_type	0
3	started_at	0
4	ended_at	0
5	start_station_name	55337
6	start_station_id	55337
7	end_station_name	57984
8	end_station_id	57984
9	start_lat	0
10	start_lng	0
11	end_lat	241
12	end_lng	241
13	member_casual	0

This is a vertical format for better readability where I have used UNNEST query to convert columns to rows.

Checking for Duplicate Values in ride_id

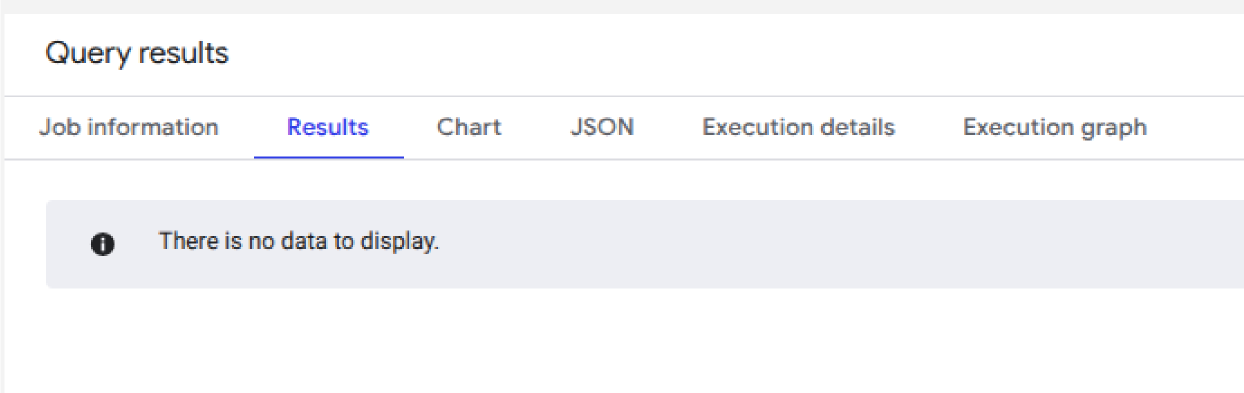
```
SELECT ride_id, COUNT(*) AS count
FROM winter-legend-mg.cyclist.cycle_trip
GROUP BY ride_id
HAVING count > 1;
```



Purpose & Insight: This step ensures each ride is uniquely identified by ride_id. Since no duplicate IDs were found, we can confirm that the dataset maintains integrity in this regard, indicating no immediate concerns with data duplication.

Spatial Outliers

```
SELECT *
FROM winter-legend-mg.cyclist.cycle_trip
WHERE start_lat NOT BETWEEN 41.6 and 42.1
OR start_lng NOT BETWEEN -88 and -87;
```



To ensure data quality in spatial analyses, we checked for geographic outliers by filtering records whose starting latitude and longitude fell outside the general Chicago area. This helps identify any GPS errors or incorrect data entries that could mislead mapping or routing insights. Since no such outliers were found, we can assume the starting point coordinates are within acceptable bounds for the city. End coordinates were not checked in this step because trips typically begin and end within the same geographic area, and start locations are generally more critical for spatial clustering and demand analysis.

Data Inconsistencies & Anomalies

Checking for Anomalies in start_station_name

```
SELECT start_station_name,
LENGTH(start_station_name) AS original_length,
LENGTH(TRIM(start_station_name)) AS new_length
start_station_name != TRIM(start_station_name)
AS has_whitespace,
start_station_name != INITCAP(LOWER(TRIM
(start_station_name) AS has_casing_issue
FROM winter-legend-mg.cyclist.cycle_trip
GROUP BY start_station_name
ORDER BY start_station_name;
```

Job information Results Chart JSON Execution details Execution graph

Row	start_station_name	original_length	new_length	has_whites	has_casing
1	null	null	null	null	null
2	2112 W Peterson Ave	19	19	false	false
3	21st St & Pulaski Rd	20	20	false	false
4	63rd St Beach	13	13	false	false
5	900 W Harrison St	17	17	false	false
6	Aberdeen St & 103rd St	22	22	false	false
7	Aberdeen St & Jackson Blvd	26	26	false	false
8	Aberdeen St & Monroe St	23	23	false	false
9	Aberdeen St & Randolph St	25	25	false	false
10	Ada St & 113th Place	20	20	false	false
11	Ada St & Washington Blvd	24	24	false	false
12	Adler Planetarium	17	17	false	false
13	Albany Ave & 16th St	20	20	false	false
14	Albany Ave & 26th St	20	20	false	false
15	Albany Ave & Belmont Ave	24	24	false	false
16	Albany Ave & Bloomingdale Ave	29	29	false	false
17	Albany Ave & Douglas Blvd	25	25	false	false

To ensure consistency and cleanliness in station names, this query checks for anomalies like leading/trailing spaces and inconsistent capitalization. By comparing the original length of each name with its trimmed version, we can identify names affected by extra spaces that might cause problems during grouping or aggregation. Additionally, by comparing casing patterns, we can detect names that are inconsistently formatted (e.g., "CLARK ST" vs "Clark St"). Addressing these inconsistencies is important for accurate analysis, especially in visualizations, clustering, or matching records with external geographic data. Identifying and standardizing such station names improves overall data integrity.

Check for Station Names with Multiple Coordinate Variants

```
SELECT

start_station_name,
COUNT(DISTINCT ROUND(start_lat, 3)
|| ',' || ROUND(start_lng, 3))
AS coordinate_variants
FROM `winter-legend-mg.cyclist.cycle_trip`
GROUP BY start_station_name
HAVING coordinate_variants > 1;
```

If a station has multiple coordinate entries, it may indicate **inconsistent geolocation data**, **GPS drif**t, or **manual entry errors**. This inconsistency can affect location-based analyses and mapping accuracy. If found, it is often helpful to resolve these by verifying coordinates or using a master station reference for consistency.

To assess consistency between station names and their mapped coordinates, we analyzed how many unique location pairs (latitude and longitude) were linked to each station name. Minor GPS drift was controlled by rounding coordinates to three decimal places (~111 meters of precision). This significantly reduced coordinate variants per station — with the maximum dropping from over 40 to just 6 & minimum being 2.

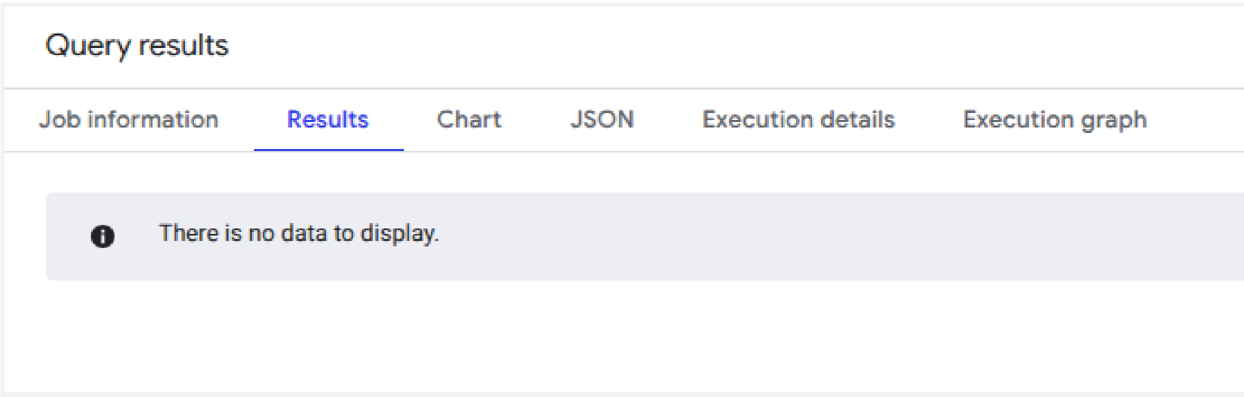
This confirms that most coordinate inconsistencies were not meaningful errors but rather technical noise. A few station names with multiple valid coordinates likely represent physical installations at different points with the same label, such as opposite street corners or public hubs.

Job information | Results | Chart | JSON

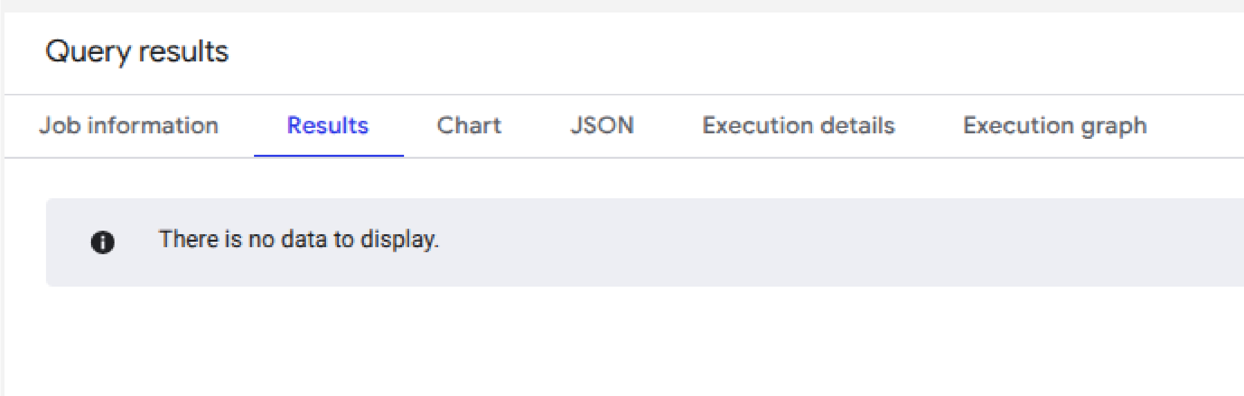
Row	start_station_name	coordinate_variants
1	Blackstone Ave & 59th St	2
2	University Ave & 57th St	2
3	Lake Park Ave & 56th St	2
4	Wentworth Ave & 35th St	2
5	McCormick Place	2
6	Wabash Ave & Cermak Rd	2
7	Halsted St & 18th St	2
8	Shedd Aquarium	2
9	Hermitage Ave & Polk St	2
10	Federal St & Polk St	2
11	Michigan Ave & 8th St	2
12	Peoria St & Jackson Blvd	3
13	Franklin St & Jackson Blvd	4
14	LaSalle St & Adams St	3
15	Dearborn St & Adams St	4
16	Wabash Ave & Adams St	2
17	Franklin St & Monroe St	6

Check for Station Names Missing Station IDs

```
SELECT start_station_name,start_station_ID
FROM `winter-legend-mg.cyclist.cycle_trip`
WHERE start_station_name IS NOT NULL
AND (start_station_id IS NULL OR TRIM(start_station_id)
 = '')
ORDER BY start_station_name;
```




```sql
SELECT end_station_name,end_station_ID
FROM `winter-legend-mg.cyclist.cycle_trip`
WHERE end_station_name IS NOT NULL
AND (end_station_id IS NULL OR TRIM(end_station_id)
= '')
ORDER BY end_station_name;
```

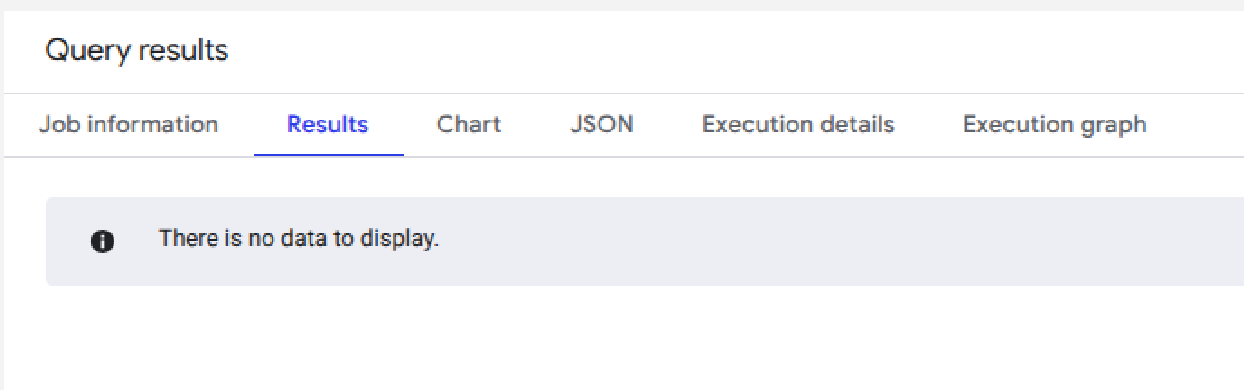


I ran a SQL query that filtered rows with a non-null station name but a missing or blank station ID. However, the query returned no results, indicating that end_station_id and start_station_id are consistently present wherever an end_station_name and start_station_name exists. This suggests a strong level of integrity in this part of the dataset.

Validating Timestamp Consistency

After addressing missing values, the next step is to ensure temporal data integrity by checking for inconsistent timestamps. Specifically, a ride's ended_at timestamp should never precede its started_at timestamp. Such anomalies typically indicate data entry errors or system glitches, which can mislead calculations involving ride duration and usage patterns.

```sql
SELECT *
FROM winter-legend-mg.cyclist.cycle_trip
WHERE ended_at < started_at;
```

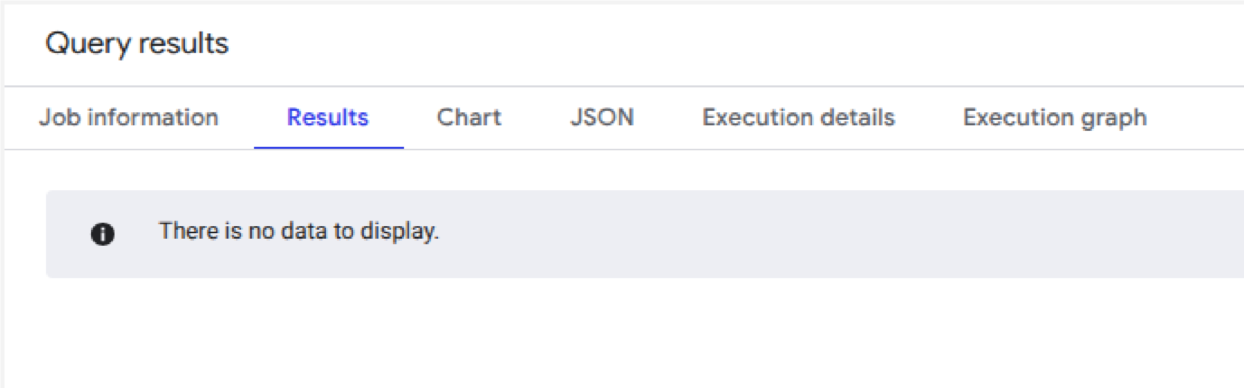


The above result shows that there are no timestamp inconsistencies in this dataset.

Flag Negative Duration

To quantify how severe the issue is, we can calculate the ride duration using TIMESTAMP_DIFF() and filter for rides with negative durations.

```sql
SELECT ride_id,
started_at, ended_at,
TIMESTAMP_DIFF(ended_at, started_at, MINUTE) AS ride_duration_minutes
FROM winter-legend-mg.cyclist.cycle_trip
WHERE TIMESTAMP_DIFF(ended_at, started_at, MINUTE) < 0;
```



This snapshot shows that no negative duration was found in this dataset. The dataset does not have anomalies related to timestamp and duration.

Validating Categorical Fields

After ensuring timestamp consistency, the next focus is to assess the consistency and validity of categorical fields, especially rideable_type and member_casual. These columns are critical for segmenting and comparing user behaviors. It's essential to confirm that they contain only the expected categories and are formatted uniformly.

```
-- Check all values present in rideable_type
SELECT DISTINCT rideable_type
FROM `your_dataset_name.tripdata`;
-- Check all values present in member_casual
SELECT DISTINCT member_casual
FROM `your_dataset_name.tripdata`;
```

The above screenshot shows the result of the query that retrieves unique values present in rideable_type column. These values are *classic_bike* & *electric_bike*.

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	member_casual
1	casual
2	member

The above snapshot shows the result of the query that retrieves distinct values of member_casual column. These values are *casual* & *member*.

Ride Volume and User Distribution

With data quality checks completed, the next step in the Prepare Phase involves understanding the overall size of the dataset and the proportion of rides taken by different user segments. This helps in checking for any imbalances or anomalies in user classification.

```
-- Total Ride Counts and User Segmentation
SELECT COUNT(*) AS total_ride_count,
COUNTIF(member_casual = 'member') AS member_rides,
COUNTIF(member_casual = 'casual') AS casual_rides
FROM winter-legend-mg.cyclist.cycle_trip;
```

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	total_ride_count	member_rides	casual_rides
1	298155	212286	85869

The dataset consists of 298,155 rides in total, of which 212,286 were taken by member users and by casual users. This indicates a higher usage of Cyclistic bikes by annual members compared to casual users, which will be an important point of comparison in later stages of the analysis.

ROCCC Assessment

The ROCCC framework was used to assess the dataset's quality:

- Reliable: Sourced from official system logs of Divvy/Cyclistic.
- Original: Directly generated by the operational system.
- Comprehensive: Includes timestamps, location, and user type for every ride.
- Current: Reflects activity from the most recent full calendar year (2025).
- Cited: Publicly accessible under terms defined by the provider.

Data Limitations

Despite being robust, the dataset has a few limitations:

- Some records contain null station names & IDs.
- No demographic attributes (e.g., gender, age) are available.
- This dataset has limited information, the data is recorded from January till March, 2025.

Process Phase

Tools Chosen

For the Process Phase, I continued using Google BigQuery due to its ability to efficiently query large datasets without manual data handling. I also plan to use Tableau for data visualization in the next phase. BigQuery allows SQL-based transformations which are ideal for cleaning, validating, and preparing data at scale.

Ensuring Data Integrity

The data was previously validated in the Prepare Phase through steps like schema checks, null value checks, consistency verification for timestamps, and validation of categorical fields such as `rideable_type` and `member_casual`. These checks ensured that the dataset structure was reliable and trustworthy before transformations began.

Data Cleaning and Transformation

Filtering Invalid Rows

In the Prepare Phase, rows with missing station names or coordinates, and those with ride duration less than 1 minute or more than 1440 minutes (24 hours), and columns with null values were removed. The cleaned data is stored in a new table: `` `winter-legend-mg.cyclist.cleaned_cycle_trip` ``.

Filter Enter property name or value

☐	Field name	Type	Mode	Key	Collation	Default Value
☐	ride_id	STRING	NULLABLE	-	-	-
☐	rideable_type	STRING	NULLABLE	-	-	-
☐	started_at	TIMESTAMP	NULLABLE	-	-	-
☐	ended_at	TIMESTAMP	NULLABLE	-	-	-
☐	start_station_name	STRING	NULLABLE	-	-	-
☐	start_station_id	STRING	NULLABLE	-	-	-
☐	start_lat	FLOAT	NULLABLE	-	-	-
☐	start_lng	FLOAT	NULLABLE	-	-	-
☐	member_casual	STRING	NULLABLE	-	-	-

```
--DROP AND RECREATE A CLEANED TABLE
DROP TABLE IF EXISTS
winter-legend-mg.cyclist.cleaned_cycle_trip;
CREATE TABLE
winter-legend-mg.cyclist.cleaned_cycle_trip AS
SELECT
ride_id,
rideable_type,
started_at,
ended_at,
start_station_name,
start_station_id,
start_lat,
start_lng,
member_casual
FROM
winter-legend-mg.cyclist.cycle_trip
WHERE
start_station_name IS NOT NULL AND
start_station_id IS NOT NULL AND
start_lat IS NOT NULL AND
start_lng IS NOT NULL AND
started_at IS NOT NULL AND
ended_at IS NOT NULL AND
```

```
--REMOVING ROWS WITH INVALID RIDE DURATION IF ANY
TIMESTAMP_DIFF(ended_at, started_at, MINUTE)
BETWEEN 1 AND 1440;
```

This SQL query performs a crucial data cleaning operation to ensure the reliability and quality of the Cyclistic dataset before conducting any further analysis. The table cleaned_cycle_trip is created by first dropping any existing version to avoid duplicates or outdated data, then selectively retaining rows from the original cycle_trip table based on several criteria. Furthermore, the query eliminates rides with suspicious or erroneous durations by only retaining trips between 1 and 1440 minutes (i.e., 1 minute to 24 hours). This step helps discard outliers like extremely short or excessively long rides that may result from system errors or misuses.

Extracting Month, Days and Time

```
SELECT *,
DATE(started_at) AS start_date,
FORMAT_TIMESTAMP('%B', started_at) AS start_month,
FORMAT_TIMESTAMP('%A', started_at) AS start_day_of_week,
TIME(started_at) AS start_time,
DATE(ended_at) AS end_date,
FORMAT_TIMESTAMP('%B', end_date) AS end_month,
FORMAT_TIMESTAMP('%A', end_date) AS end_day_of_week,
TIME(ended_at) AS end_time
FROM winter-legend-mg.cyclist.cleaned_cycle_trip;
```

Query results

Save results · Open in

Job information · Results · Chart · JSON · Execution details · Execution graph

Row	start_date	start_month	start_day_of_week	start_time	end_date	end_month	end_day_of_week	end_time
1	2025-03-16	March	Sunday	19:32:56.890000	2025-03-16	March	Sunday	19:36:03.157000
2	2025-03-25	March	Tuesday	23:53:59.912000	2025-03-25	March	Tuesday	23:56:24.441000
3	2025-03-16	March	Sunday	01:28:30.394000	2025-03-16	March	Sunday	01:30:58.660000
4	2025-03-26	March	Wednesday	13:00:12.720000	2025-03-26	March	Wednesday	14:10:35.810000
5	2025-03-31	March	Monday	17:30:35.757000	2025-03-31	March	Monday	17:34:11.741000
6	2025-03-12	March	Wednesday	14:52:36.440000	2025-03-12	March	Wednesday	15:02:14.886000
7	2025-03-12	March	Wednesday	14:48:04.049000	2025-03-12	March	Wednesday	15:02:12.272000
8	2025-03-12	March	Wednesday	14:48:12.825000	2025-03-12	March	Wednesday	15:02:41.876000
9	2025-03-26	March	Wednesday	18:55:52.502000	2025-03-26	March	Wednesday	19:06:01.729000
10	2025-03-27	March	Thursday	14:40:02.387000	2025-03-27	March	Thursday	15:02:37.901000
11	2025-03-16	March	Sunday	16:39:46.385000	2025-03-16	March	Sunday	16:41:58.446000
12	2025-03-26	March	Wednesday	12:28:43.434000	2025-03-26	March	Wednesday	12:54:01.916000
13	2025-03-31	March	Monday	17:37:32.783000	2025-03-31	March	Monday	17:51:25.978000
14	2025-03-16	March	Sunday	13:32:33.128000	2025-03-16	March	Sunday	13:43:52.062000
15	2025-03-30	March	Sunday	21:06:48.733000	2025-03-30	March	Sunday	21:12:25.446000
16	2025-03-25	March	Tuesday	15:39:12.937000	2025-03-25	March	Tuesday	16:03:51.993000

Results per page: 50 · 1 – 50 of 239447

This snapshot depicts the newly created columns. I extracted meaningful temporal components from the started_at and ended_at timestamp fields to enhance the dataset's analytical capabilities. Specifically, I broke down the timestamp into the date, day of the week, month, and exact time.

FINAL VERSION OF CLEAN TABLE

This final cleaning step was performed to create a structured, analysis-ready dataset that eliminates redundancy while enhancing time-based insights. By removing the raw timestamp columns started_at and ended_at, and replacing them with easily interpretable components such as start_date, start_month, start_day_of_week, and ride_length_minutes, the table becomes more suitable for reporting, visualizations, and time-series trends.

Unnecessary columns like end_station_name, end_station_id, end_lat, and end_lng — which were entirely null — were removed earlier in the cleaning phase. This final version includes only useful and valid records (ride duration between 1 and 1440 minutes), and ensures all required fields for spatial, temporal, and user-type analysis are present. Below are the screenshots of the final dataset:

Query results

Save results · Open in

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	ride_id	rideable_type	start_station_name	start_station_id	start_lat	start_lng	member_casual
1	49DF6DE1BAF72FA6	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
2	7E16AF2231CACA38	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	casual
3	2F83D8B8218F11FF	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
4	4233194F5DBC262E	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
5	A2A39AC796EC32A3	classic_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
6	8A3BF53753746591	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
7	C3A83A12F23FBE80	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
8	EED0F1AB26F8F608	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	casual
9	9C955B9996694424	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	casual
10	94141DF5B86F7FB4	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	casual
11	F13F95BBD8E4943E	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member
12	F669E7629E3549D1	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	casual
13	1C94971BDC605A72	electric_bike	2112 W Peterson Ave	KA1504000155	41.991178	-87.683593	member

Query results

Save results · Open in

Job information | Results | Chart | JSON | Execution details | Execution graph

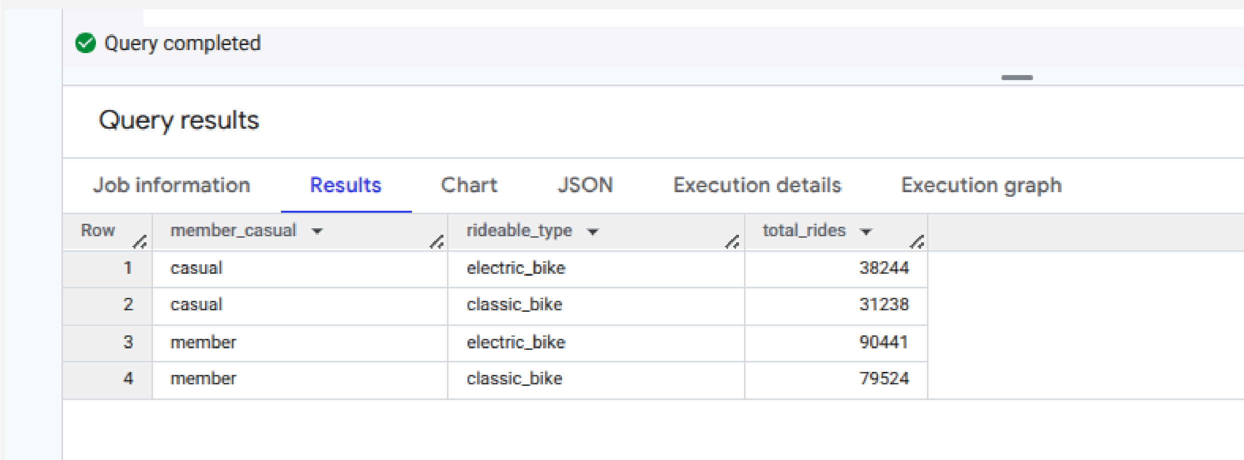
Row	start_date	start_month	start_day_of_week	start_time	end_date	end_month	end_day_of_week	end_
1	2025-03-01	March	Saturday	00:17:15.805000	2025-03-01	March	Saturday	00:3
2	2025-03-02	March	Sunday	20:38:19.299000	2025-03-02	March	Sunday	20:5
3	2025-03-02	March	Sunday	15:53:48.071000	2025-03-02	March	Sunday	15:5
4	2025-03-03	March	Monday	18:03:39.192000	2025-03-03	March	Monday	18:1
5	2025-03-03	March	Monday	15:53:34.068000	2025-03-03	March	Monday	16:0
6	2025-03-06	March	Thursday	23:48:28.768000	2025-03-07	March	Friday	00:0
7	2025-03-07	March	Friday	02:44:38.437000	2025-03-07	March	Friday	02:5
8	2025-03-11	March	Tuesday	21:49:39.422000	2025-03-11	March	Tuesday	21:5
9	2025-03-11	March	Tuesday	13:39:40.036000	2025-03-11	March	Tuesday	13:5
10	2025-03-12	March	Wednesday	10:42:44.040000	2025-03-12	March	Wednesday	10:5
11	2025-03-13	March	Thursday	00:51:31.347000	2025-03-13	March	Thursday	01:0
12	2025-03-14	March	Friday	20:54:48.518000	2025-03-14	March	Friday	21:0
13	2025-03-14	March	Friday	19:57:10.062000	2025-03-14	March	Friday	20:0

In this phase, I focused on transforming raw Cyclistic trip data into a clean, structured, and analysis-ready format. I removed rows with missing or invalid values, filtered out rides with implausible durations, and eliminated redundant or fully null columns. Additionally, I extracted key temporal features like day of the week, month, and ride duration to enhance the dataset's usability for trend analysis. The final cleaned table now includes only essential, validated columns that will support meaningful insights in the next phase.

Analyze Phase

Bike Type Usage by Rider Category

```
SELECT member_casual, rideable_type,
COUNT(*) AS total_rides
FROM winter-legend-mg.cyclist.final_cycle_trip
GROUP BY member_casual,rideable_type
ORDER BY member_casual,total_rides DESC;
```



Row	member_casual	rideable_type	total_rides
1	casual	electric_bike	38244
2	casual	classic_bike	31238
3	member	electric_bike	90441
4	member	classic_bike	79524

This analysis shows how different types of users prefer various types of bikes during their trips. Electric bikes are clearly the most popular among both casual users and members. With casual members having 38,244 electric-bike rides and paid members having 90,441 electric-bike rides, it shows that people prefer speed & convenience offered by electric-bikes. Classic bikes are also heavily used but slightly less than electric bikes with 31,238 & 79,524 for casual and paid members respectively. This may indicate that cost-conscious or fitness-oriented users still value traditional bikes.

Monthly Trip Distribution

```
SELECT start_month, member_casual,
COUNT(*) AS total_rides
FROM winter-legend-mg.cyclist.final_cycle_trip
GROUP BY start_month, member_casual ORDER BY start_month, total_rides DESC;
```

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	start_month	member_casual	total_rides
1	February	member	15
2	February	casual	7
3	March	member	169950
4	March	casual	69475

In February, both casual and member riders had very low ride counts — only 15 for members and 7 for casual users. This could be due to missing data, data errors, or incomplete logs for that month. It's an anomaly that suggests either limited usage (due to weather or availability) or that the dataset doesn’t contain full February records.

In March, there's a dramatic increase in rides: members took 169,950 rides and casual users took 69,475 rides. This indicates that March marks the beginning of higher usage months, possibly due to improving weather conditions or better availability of bikes.

Weekly Usage Trends

```
SELECT start_day_of_week,member_casual,
COUNT(*) AS total_rides
FROM winter-legend-mg.cyclist.final_cycle_trip
GROUP BY start_day_of_week,member_casual
ORDER BY start_day_of_week,total_rides;
```

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	start_day_of_week	member_casual	total_rides
1	Friday	casual	15245
2	Friday	member	26760
3	Monday	casual	9418
4	Monday	member	30491
5	Saturday	casual	16362
6	Saturday	member	23470
7	Sunday	casual	8629
8	Sunday	member	17078
9	Thursday	casual	6851
10	Thursday	member	24443
11	Tuesday	casual	7372
12	Tuesday	member	25326
13	Wednesday	casual	5605
14	Wednesday	member	22397

The analysis of ride frequency across different days of the week reveals distinct patterns of casual riders and members. Casual riders preferred using the services of Cyclistic on weekends, particularly on weekends (Saturday : 16, 362) meanwhile members exhibit more consistent riding pattern throughout the week, with peak on Monday: 30,491.

Hourly Ride Distribution

```
SELECT EXTRACT(HOUR FROM start_time) AS hour_of_day,
member_casual, COUNT(*) AS total_rides
FROM winter-legend-mg.cyclist.final_cycle_trip
GROUP BY hour_of_day, member_casual
ORDER BY hour_of_day, total_rides;
```

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	hour_of_day	member_casual	total_rides
1	0	casual	1002
2	0	member	1213
3	1	casual	620
4	1	member	692
5	2	casual	370
6	2	member	394
7	3	casual	286
8	3	member	308
9	4	casual	250
10	4	member	374
11	5	casual	313
12	5	member	1467
13	6	casual	887
14	6	member	4797

Image 1.1

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	hour_of_day	member_casual	total_rides
15	7	casual	1765
16	7	member	10057
17	8	casual	2792
18	8	member	13845
19	9	casual	2567
20	9	member	8742
21	10	casual	2814
22	10	member	6992
23	11	casual	3789
24	11	member	8142
25	12	casual	4682
26	12	member	9381
27	13	casual	5028
28	13	member	9215

Image 1.2

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	hour_of_day	member_casual	total_rides
29	14	casual	5544
30	14	member	9529
31	15	casual	6052
32	15	member	11714
33	16	casual	6913
34	16	member	16781
35	17	casual	7149
36	17	member	19195
37	18	casual	5669
38	18	member	13781
39	19	casual	3640
40	19	member	8745
41	20	casual	2334
42	20	member	5535

Image 1.3

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	hour_of_day	member_casual	total_rides
35	17	casual	7149
36	17	member	19195
37	18	casual	5669
38	18	member	13781
39	19	casual	3640
40	19	member	8745
41	20	casual	2334
42	20	member	5535
43	21	casual	1962
44	21	member	4139
45	22	casual	1693
46	22	member	3043
47	23	casual	1361
48	23	member	1884

Image 1.4

This SQL query is used to analyze the distribution of bike rides by hour of the day for both casual and member riders. The purpose of this query is to understand what time of day users are most active, which can help in planning bike availability across different hours. Tailoring operational decisions to peak usage hours. Identifying differing usage behaviors between casual and member riders.

Average Ride Duration per Month

```
SELECT
start_month,
member_casual,
ROUND(AVG(ride_lenght_minutes), 2)
AS  avg_ride_length_per_month
FROM
winter-legend-mg.cyclist.final_cycle_trip
GROUP  BY
start_month, member_casual
ORDER  BY member_casual, start_month;
```

Query results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	start_month	member_casual	avg_ride_length_per
1	February	casual	179.64
2	March	casual	19.08
3	February	member	16.93
4	March	member	11.19

In this query, you calculated the average ride duration per month for each user type (member and casual) from the cleaned trip data. You used the AVG() function to summarize ride length (ride_lenght_minutes), grouped the results by start_month and member_casual, and ordered them for clarity. The purpose is to uncover monthly usage patterns and ride duration trends across user types to support behavioral analysis and business strategy.

Most Popular Start Stations

```
SELECT start_station_name,member_casual,
COUNT(*) AS ride_count
FROM winter-legend-mg.cyclist.final_cycle_trip
WHERE start_station_name IS NOT NULL
GROUP BY start_station_name,member_casual
ORDER BY ride_count DESC;
```

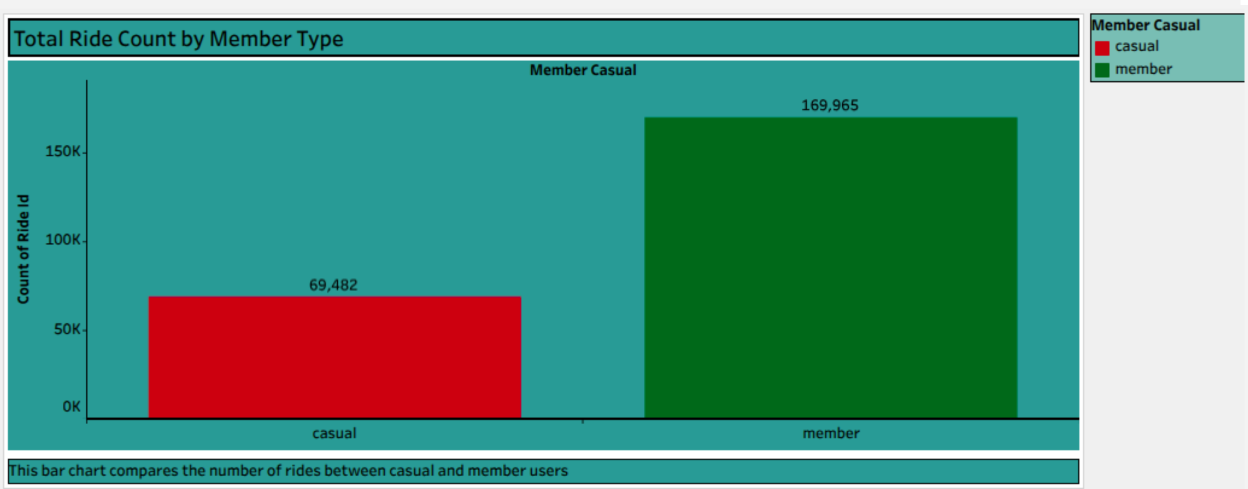
Job information | Results | Chart | JSON | Execution details | Execution graph

Row	start_station_name	member_casual	ride_count
1	Kingsbury St & Kinzie St	member	2233
2	Streeter Dr & Grand Ave	casual	1877
3	Clinton St & Washington Blvd	member	1819
4	Clinton St & Madison St	member	1469
5	Clark St & Elm St	member	1448
6	Canal St & Madison St	member	1444
7	University Ave & 57th St	member	1439
8	DuSable Lake Shore Dr & Monr...	casual	1424
9	Canal St & Adams St	member	1284
10	Wells St & Elm St	member	1269
11	Clinton St & Jackson Blvd	member	1260
12	Ellis Ave & 60th St	member	1236
13	State St & Chicago Ave	member	1201
14	Wells St & Concord Ln	member	1171

In this query, I identified the most popular starting stations for both casual and member riders by counting how many rides began at each station. I excluded rides with missing start station names to ensure accuracy, then grouped the data by start_station_name and member_casual and sorted the results by ride count in descending order. The goal is to find key high-traffic locations, which can inform station placement, resource allocation, and user experience improvements.

Share Phase

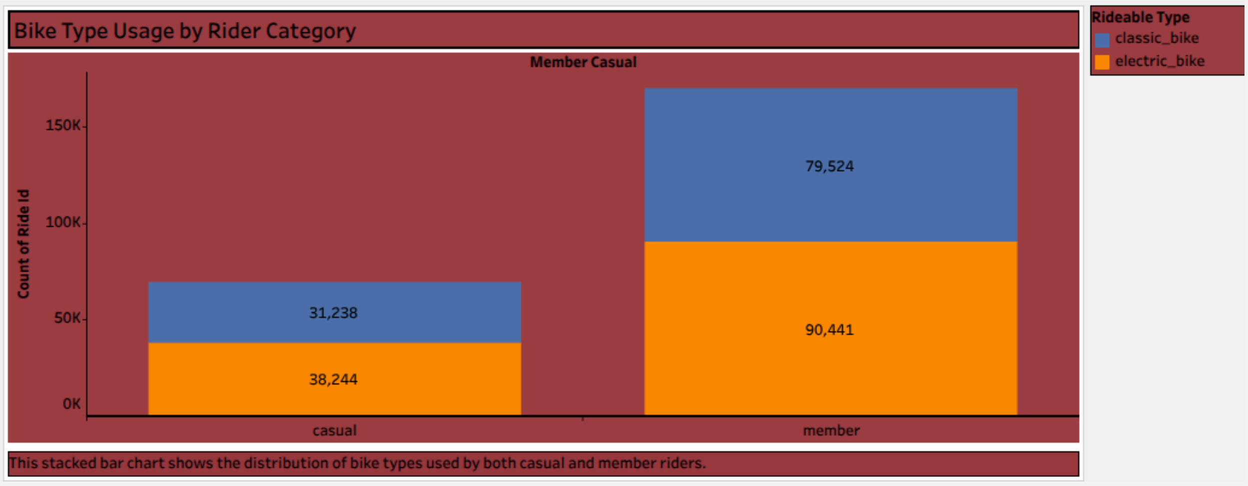
Total Ride Count By Member Type



This chart shows following insights:

- Dominance of Members: The bar chart clearly shows that members take significantly more rides compared to casual users. Member users account for 169,965 rides, while casual users take only 69,482 rides. This indicates that members are more engaged and consistently use the service.
- Potential for Growth: The substantial difference highlights an opportunity to convert casual riders into members. Targeted marketing campaigns offering membership benefits might increase conversion rates.
- Strategic Focus and Business Implications: Since members form the bulk of ride counts, retention strategies for existing members should be prioritized. At the same time, understanding casual users' barriers to membership could help bridge the gap. Increasing the number of members will likely stabilize revenue, as memberships typically provide recurring income. Offering membership incentives, discounts, or exclusive services might encourage casual users to subscribe.

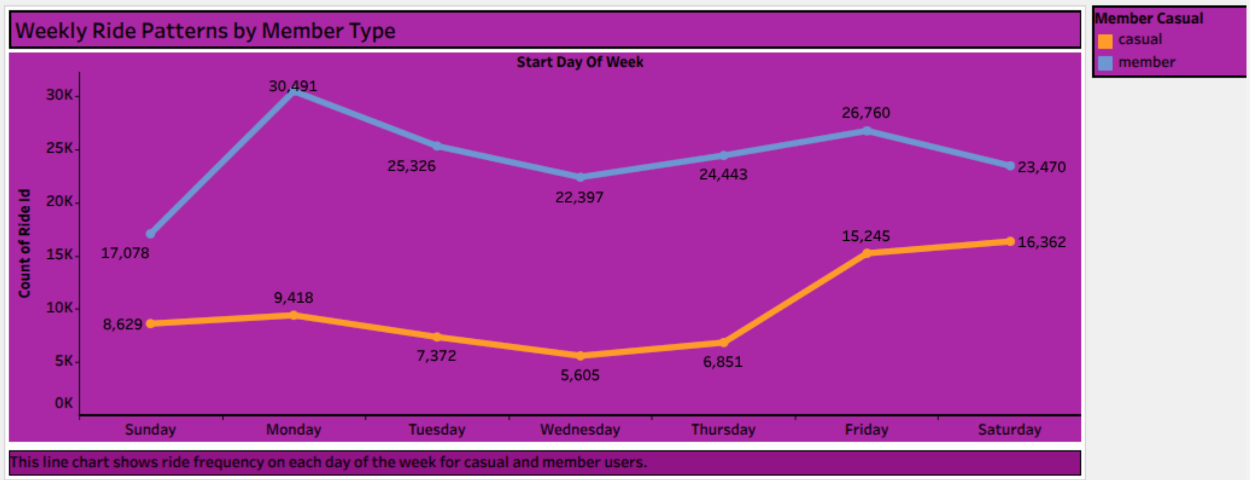
Bike Type Usage by Rider Category



This chart shows :

- Usage Distribution: The stacked bar chart reveals that members prefer classic bikes significantly more than casual users. Among members, 79,524 rides were made on classic bikes, while casual users recorded 31,238 rides on the same type. This indicates that members have a stronger preference for traditional biking.
- Electric Bike Popularity: The chart also shows that electric bikes are more balanced in usage between casual and member riders. Casual users took 38,244 rides on electric bikes, compared to 90,441 rides by members. This suggests that electric bikes might be appealing to both user groups, although members still dominate usage.
- Strategic Insights: Promoting classic bike advantages among casual users could increase engagement, as members show a clear preference for them. Additionally, electric bike promotions could target both casual and member users, leveraging the balanced interest in this category. Analyzing why members prefer classic bikes might also guide future marketing efforts.

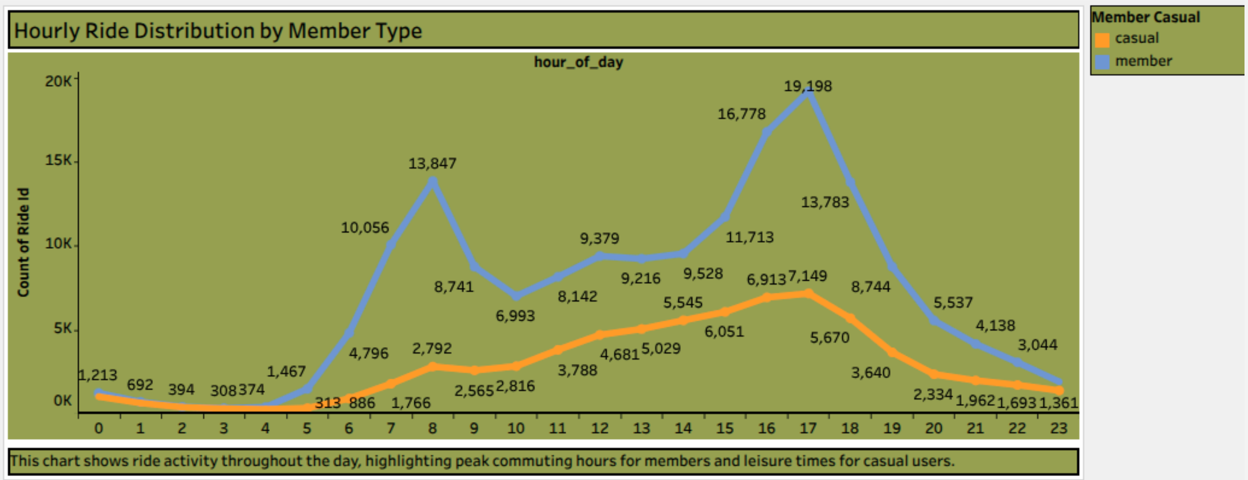
Weekly Ride Patterns by Member Type



This chart helps to give following insights:

- Member Usage Peaks on Weekdays: The line chart shows that members have higher ride counts on weekdays, with the highest peak on Monday (30,491 rides). The trend decreases steadily from Monday through Wednesday, slightly rises on Thursday (26,760), and dips again on Friday. This pattern indicates that members use the service mainly for commuting or routine purposes.
- Casual Users Favor Weekends: Casual riders, on the other hand, show a distinct pattern with higher activity on weekends, especially Saturday (16,362 rides) and Sunday (17,078 rides). Ride counts dip during weekdays, reaching the lowest point on Wednesday (5,605 rides). This pattern suggests that casual users prefer recreational or leisure rides rather than daily commuting.
- Opportunity for Targeted Marketing: Marketing strategies could focus on promoting weekday ride packages to casual users to boost off-peak usage. Additionally, weekend ride promotions for members could help balance the usage pattern. Understanding the distinct riding habits of both user types can aid in designing tailored marketing campaigns.

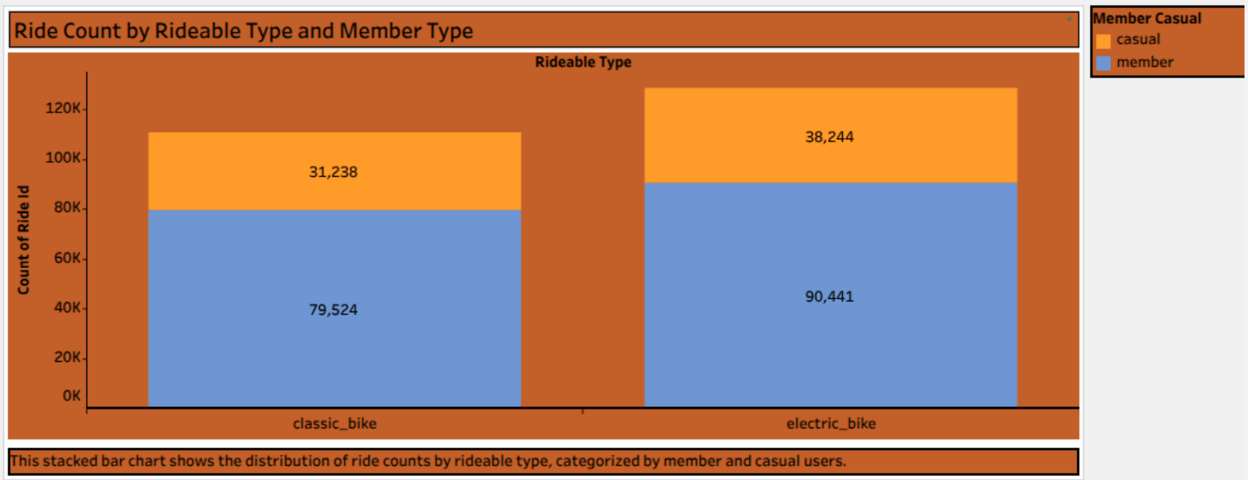
Hourly Ride Distribution by Member Type



This chart provides following insights:

- Peak Usage by Members: Members exhibit two clear peaks: 8 AM (13,847 rides) and 5 PM (19,198 rides), indicating commute-oriented usage. These peaks align with typical work start and end times, emphasizing member dependency on bikes for daily commuting.
- Casual Riders' Steady Usage: Casual users show a more even distribution throughout the day, with a gradual rise from morning to afternoon. The peak usage for casual riders is around 5 PM (7,149 rides), likely reflecting leisure or evening activity.
- Actionable Insights: Enhance commuter benefits during peak hours to retain member satisfaction. Introduce promotions or guided tours for casual users in the afternoon and evening to boost engagement.

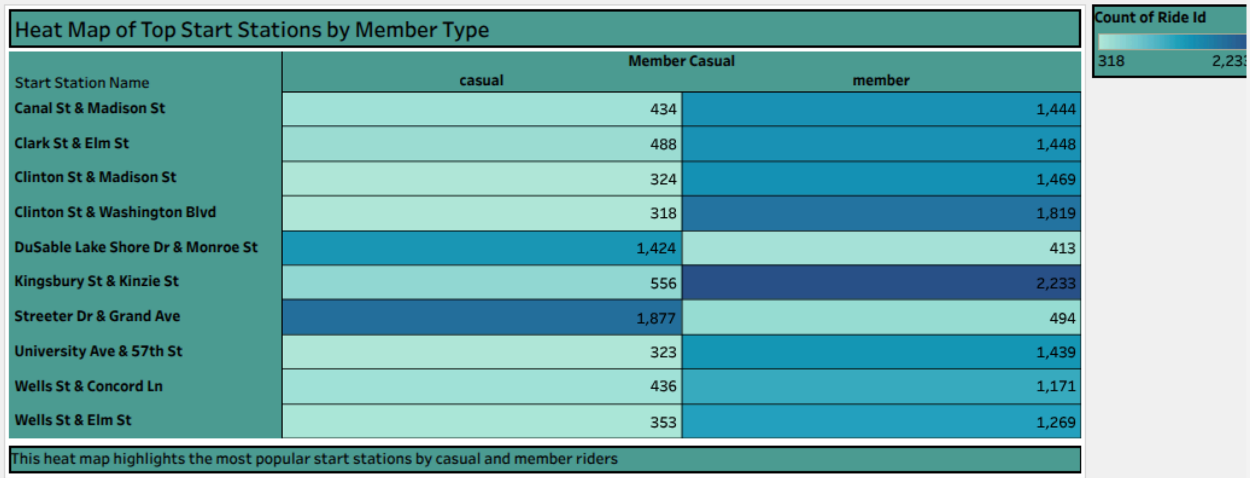
Ride Count by Rideable Type and Member Type



This visualization gives following insights:

- Dominance of Classic Bikes: Both members and casual users favor classic bikes, with members accounting for 79,524 rides and casual users for 31,238 rides. This preference suggests that classic bikes are perceived as more versatile and reliable for everyday use.
- Electric Bike Usage: Members also significantly prefer electric bikes, with 90,441 rides compared to 38,244 rides by casual users. The higher member usage indicates that electric bikes may be favored for longer commutes or quicker transit.
- Strategic Considerations: Enhance the availability of electric bikes during peak commuting hours to meet member demand. Consider promoting electric bike usage to casual users through trial rides or incentives.

Heat Map of Top Start Stations



Heat Map of Top Start Stations by Member Type

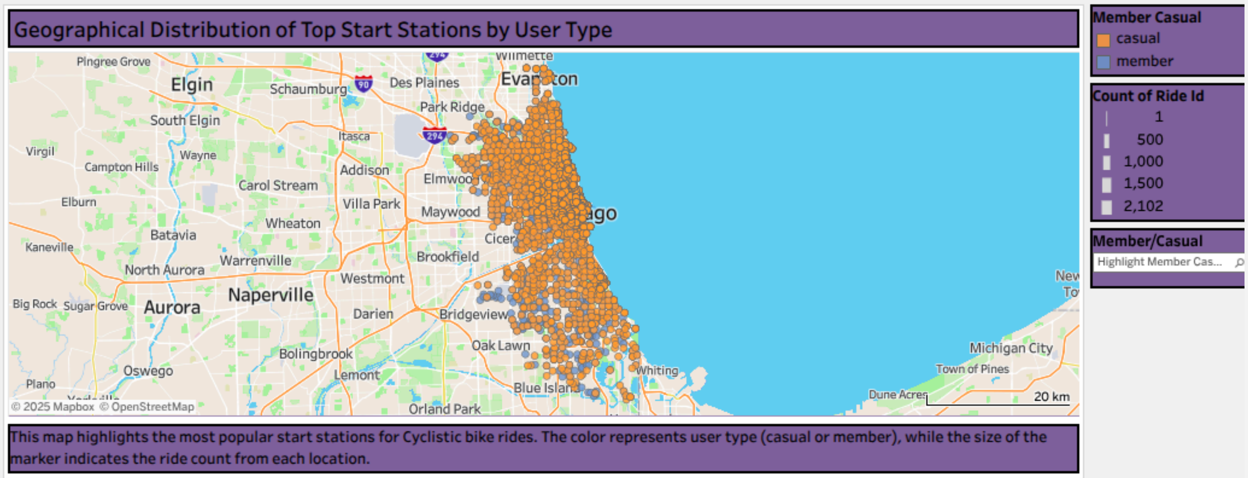
Start Station Name	Member Casual: casual	Member Casual: member
Canal St & Madison St	434	1,444
Clark St & Elm St	488	1,448
Clinton St & Madison St	324	1,469
Clinton St & Washington Blvd	318	1,819
DuSable Lake Shore Dr & Monroe St	1,424	413
Kingsbury St & Kinzie St	556	2,233
Streeter Dr & Grand Ave	1,877	494
University Ave & 57th St	323	1,439
Wells St & Concord Ln	436	1,171
Wells St & Elm St	353	1,269

This heat map highlights the most popular start stations by casual and member riders

The above chart helps in giving following valuable insights:

- Dominant Start Stations: The most popular start station for members is Kingsbury Dr & Kinzie St with 2,233 rides, while Streeter Dr & Grand Ave is the top for casual users with 1,877 rides. This indicates different riding patterns between the two groups, likely influenced by commuting versus leisure activities.
- Shared Popular Stations: Despite differences, some stations like Canal St & Madison St and Clark St & Elm St are popular among both user types. These stations might be strategically significant for targeted promotions.
- Location-Based Strategy: Enhance amenities and bike availability at these high-traffic locations to accommodate both casual and member users. Consider promoting membership options at casual-dominated stations to boost conversions.

Geographical Representation of Start Stations



- Clustered Activity in Urban Areas: The map reveals that most start stations are densely concentrated in downtown Chicago and nearby areas, indicating high cycling activity in the urban core. This clustering suggests that bike-sharing primarily serves commuters and city dwellers.
- Member Versus Casual Usage: While both user types utilize centrally located stations, there appears to be a greater density of member usage in business districts, reflecting a commuter trend. Casual users are more dispersed, hinting at leisure-based rides from parks or scenic areas.
- Strategic Expansion and Promotion: To increase ridership, Cyclistic could focus on expanding stations in underrepresented areas or enhancing visibility at popular tourist spots. Additionally, targeted membership promotions in high-casual-user zones might encourage conversions.

Recommendations

Enhance Member Engagement:

- Since members constitute the majority of ride counts, it is essential to focus on retention strategies. Offer loyalty rewards, exclusive discounts, and community-driven events to strengthen their commitment to the service.
- Additionally, provide personalized recommendations for members based on their ride patterns, such as suggesting optimal routes during peak commuting hours.

Convert Casual Riders to Members:

- Casual users primarily use bikes for leisure rather than commuting. Develop marketing campaigns that highlight the benefits of membership, such as reduced fares, flexible ride durations, and premium access to new bike models.
- Introduce limited-time promotions for casual users, like a "Try Membership for a Week" offer to showcase the value of subscribing.

Optimize Rideable Type Distribution:

- Both casual and member users significantly prefer classic bikes over electric bikes. Consider maintaining a higher ratio of classic bikes at popular stations, especially during peak commuting hours.
- For areas with high casual usage, increase the availability of electric bikes as they may appeal more to leisure riders and tourists.

Leverage Peak Hours and Weekdays:

- Member ridership peaks during weekday commuting hours (7-9 AM and 4-7 PM), indicating a strong association with work commutes. Promote membership plans tailored for daily commuters.
- Since casual users ride more on weekends, launch special weekend passes or family packages to encourage group rides and longer usage durations.

Expand Stations in High-Demand Areas:

- Most rides originate from urban hubs, especially downtown Chicago. Expanding bike stations in these areas would reduce congestion and accommodate more rides.
- Similarly, increase bike availability near popular parks, waterfronts, and tourist spots to cater to casual users.

Targeted Marketing and Promotions:

- Promote the convenience of electric bikes for leisure trips and suggest popular scenic routes to attract tourists and occasional riders.
- Highlight the cost efficiency of memberships, particularly for daily commuters, through social media, email campaigns, and station advertisements.

Data-Driven Service Improvements:

- Use historical ride data to predict high-demand periods and optimize bike distribution accordingly. Implement real-time data tracking for better station management and maintenance scheduling.
- Regularly assess the popularity of different start stations and adjust bike availability to meet demand dynamically.

Final Thoughts: The insights gathered from this case study highlight significant opportunities for Cyclistic to enhance member engagement and increase user conversion rates. By leveraging data-driven strategies, Cyclistic can solidify its market presence and boost overall customer satisfaction. Further research and continuous data monitoring will help sustain growth and adapt to changing user patterns.