Cyclistic Bike-Share Case Study (2019)
Customer vs Subscriber Usage Patterns

*Tool stack: **R (RStudio) • PostgreSQL (DBeaver) • Excel (Microsoft 365 Web)***
*Author: **Elnur Huseynov***
*Date: **23.12.2025***

Executive Summary

This case study analyzes 2019 Cyclistic (Divvy) bike-share trips to understand how Customers and Subscribers behave differently. After cleaning and standardizing four quarterly datasets, the analysis was replicated across R, PostgreSQL, and Excel to demonstrate end-to-end analytics workflow and tool fluency. Key differences were found in ride frequency, trip duration, time-of-day patterns, and seasonality, which informed recommendations to increase Customer-to-Subscriber conversion.

Business Task

Primary question: How do Customers and Subscribers use Cyclistic differently, and what actions could increase conversion to membership?

Data

- **Files**: 2019 Q1–Q4 trip data (combined into one 2019 dataset).
- **Unit of analysis**: individual bike trips.
- **Key fields used**: start/end time, trip duration, user type, station names/IDs (plus optional demographics when available).

Tools and Deliverables

- **R (RStudio)**: data cleaning, feature engineering, analysis, and visualizations.
- **PostgreSQL** + **DBeaver**: schema, cleaning view, and analysis queries (CTEs, GROUP BY).
- **Excel Web** (Microsoft 365): Power Query import, pivot tables, charts, and a one-page dashboard.

Data Preparation (Cleaning and Feature Engineering)

Main issues addressed: inconsistent column names across quarters, non-numeric duration values, and invalid/implausible trip durations.

Cleaning rules applied:

- Standardize column names across all quarters (e.g., Trip_ID, Start_Time, End_Time, Trip_Duration_Seconds, User_Type).
- Convert Start_Time and End_Time to timestamps.

- Parse trip duration as numeric seconds (remove thousands separators where present), then compute trip_duration_min = seconds / 60.
- Filter to realistic rides: trip_duration_min between 1 and 1440 (1 minute to 24 hours).
- Keep valid user categories only: Customer and Subscriber.
- Create time features for analysis: hour (0–23), day_of_week, month, plus order columns for correct sorting in charts (weekday_order, month_order).

Analysis Approach

- KPI summary by user type: total rides, average duration, and median duration.
- Rides by weekday to compare leisure vs commute patterns.
- Rides by hour to detect daily peaks.
- Rides by month to describe seasonality.

Key Results (2019)

Summary KPIs from the cleaned dataset:

User type	Rides	Avg duration (min)	Median duration (min)
Customer	879,290	39.43	25.78
Subscriber	2,936,866	12.93	9.80

Insights

- Subscribers account for the majority of rides, consistent with routine/commute usage.
- Customers take longer trips on average and have a higher median duration, consistent with leisure or tourist use.
- Hourly patterns show commute peaks for Subscribers (morning and late afternoon), while Customers are more evenly distributed around daytime hours.
- Ridership is seasonal for both groups, peaking in summer months.

Recommendations

1. Target weekend and leisure riders with conversion offers (e.g., first-month discount after a weekend ride).
2. Introduce or promote a flexible membership tier (weekend or seasonal pass) to match Customer behavior (longer rides).
3. Emphasize commuter value for Subscribers: reliability, cost savings over frequent single rides, and time efficiency.
4. Time campaigns in spring/early summer to capture Customers before peak season.

How This Project Was Replicated Across R, SQL, and Excel

R (RStudio):

- Renamed columns to a consistent schema across quarters.
- Combined quarterly tables into DF_2019 (bind_rows).
- Created trip_duration_min and time-based features.
- Generated summary tables and exported charts.

SQL (PostgreSQL):

- Created quarterly tables and combined them using UNION ALL into df_2019.
- Built a cleaned view (e.g., v_df_2019_clean) to cast/clean duration and filter invalid records.
- Wrote GROUP BY queries to create exportable aggregates for Excel (KPIs, rides by weekday/hour/month).

Excel (Excel Web):

- Imported SQL aggregate CSV outputs via Power Query (Queries & Connections).
- Created PivotTables and PivotCharts for weekday/hour/month patterns.
- Built a Dashboard sheet with KPI cards and charts for a one-page summary.