Cyclistic Data Cleaning & Preparation Report

Data Source: 12 Months of Divvy/Cyclistic Trip Data

Tool: Microsoft Excel Power Query

Prepared by: Daniel Darzi

1. Overview

This document outlines the full data-cleaning and preparation workflow applied to the Cyclistic bike-share dataset using Power Query. The goal of this process is to produce a clean, reliable, analysis-ready dataset suitable for answering the business question:

“How do annual members and casual riders use Cyclistic bikes differently?”

The steps below describe the extraction, transformation, and loading (ETL) procedures performed on the raw data.

2. Data Extraction

2.1 Importing Monthly CSV Files

- Created a folder named **DataForExcel** containing 12 monthly CSV files.
- Used **Get Data → From Folder** to import all files simultaneously.
- Verified that all files contained consistent column structures.

2.2 Removing System Files

- Deleted non-data files (e.g., desktop.ini) automatically detected by Power Query.
- Ensured only valid CSV files remained in the folder query.

3. Data Transformation

3.1 Verified and Corrected Column Data Types

- Confirmed started_at and ended_at were typed as **Date/Time**.
- Ensured latitude/longitude fields were numeric.
- Adjusted any incorrect types to maintain consistency.

3.2 Created Ride Duration Field

- Added a calculated column:
  ride_length_minutes = (ended_at – started_at) expressed in minutes
- Set the data type to **Decimal Number**.

3.3 Rounded Ride Duration

- Applied rounding to **0 decimal places** using:
  Transform → Round → Round
- Ensured the column remained numeric after rounding.

3.4 Removed Invalid Ride Durations

Applied a single multi-condition filter:

- **ride_length_minutes > 0**
- **ride_length_minutes < 1440** (24 hours)

This removed:

- Negative durations
- Zero-minute rides
- Extremely long rides caused by system errors

3.5 Handled Missing Station Information

For each of the following columns:

- start_station_name
- start_station_id
- end_station_name
- end_station_id

Replaced missing values (null) with:

"Unknown"

This prevents grouping and aggregation errors during analysis.

3.6 Added Date-Based Columns

Using the started_at field:

- **ride_date** (Date Only)
- **day_of_week** (Name of Day)
- **month** (Name of Month)
- **year** (Year)

These fields support weekday, monthly, and seasonal trend analysis.

3.7 Created Weekday Sort Column

Added a custom column to enforce correct weekday ordering:

```
let d = Text.Trim([day_of_week]) in if d = "Monday" then 1 else if d = "Tuesday" then 2
else if d = "Wednesday" then 3 else if d = "Thursday" then 4 else if d = "Friday" then 5
else if d = "Saturday" then 6 else 7
```

- Converted day_sort to **Whole Number**.
- This column will be used in PivotTables to sort weekdays chronologically (Monday → Sunday).

4. Data Loading

4.1 Load Configuration

Using **Load To...** from the Excel Queries pane:

- Selected **Only Create Connection**
- Enabled **Add this data to the Data Model**

This approach:

- Prevents Excel from loading millions of rows into a worksheet
- Enables fast PivotTables
- Supports DAX and large-scale analysis

- Keeps the workbook responsive

5. Current Status

The dataset is now:

- Fully cleaned
- Structurally consistent
- Free of invalid durations
- Free of missing station identifiers
- Enriched with date-based fields
- Ready for PivotTable analysis or Power BI modeling

6. Creating Pivot Table

To sort weekdays correctly:

- Add day_of_week to Rows.
- Add day_sort to Rows (under day_of_week).
- Right-click day_sort → Sort Smallest to Largest.
- Hide the day_sort field.

6.1 “The Grand Total in the PivotTable represents a weighted average of all rides, not the average of the two group averages. Because members take significantly more rides than casual riders, the overall average is pulled toward the member average. This is expected and mathematically correct.”