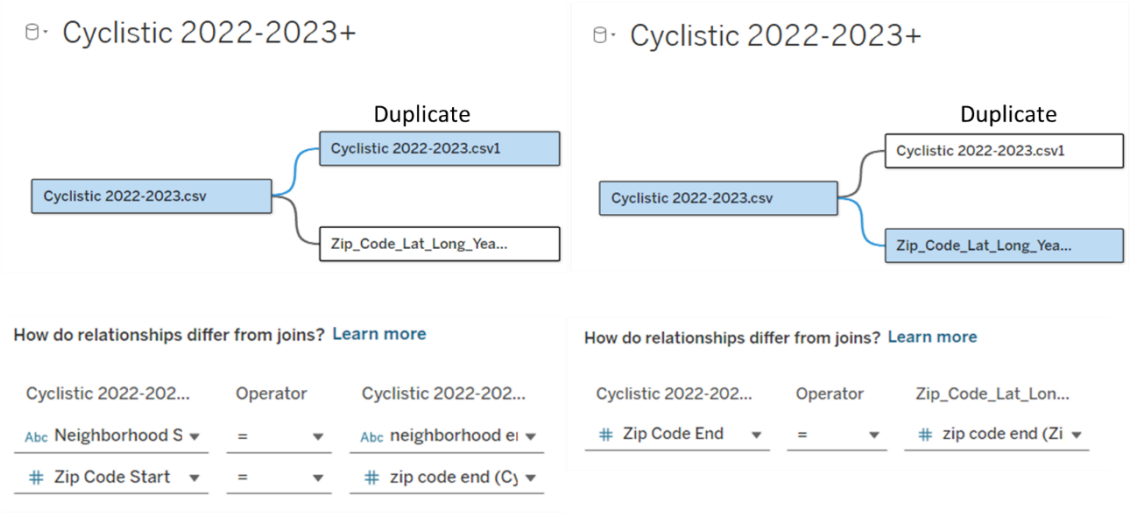
Annual database relationships



The first relationship involves the CSV file for annual trends generated with BigQuery, which is matched (related) with a duplicate file. These datasets were related by matching the initial neighborhood and zip code of each trip with the final neighborhood and zip code of each trip. This was done to calculate congestion, or the difference between bikes leaving and arriving at each neighborhood.

The second relationship involves the CSV file for annual trends generated with BigQuery, matched with a zip code and coordinates CSV file. These datasets were related by matching the zip codes of the final locations.

The second

Calculating total trips per station and congestion

Calculated Fields

- Total trips per initial neighborhood, using variables from the original CSV file.

  Neigh_start_count — Cyclistic 2022-2023+

  ```
  { FIXED [Neighborhood Start]: SUM([Trip Count]) }
  ```

- Total trips per final neighborhood, using variables from the duplicate CSV file.

  Neigh_end_count — Cyclistic 2022-2023+

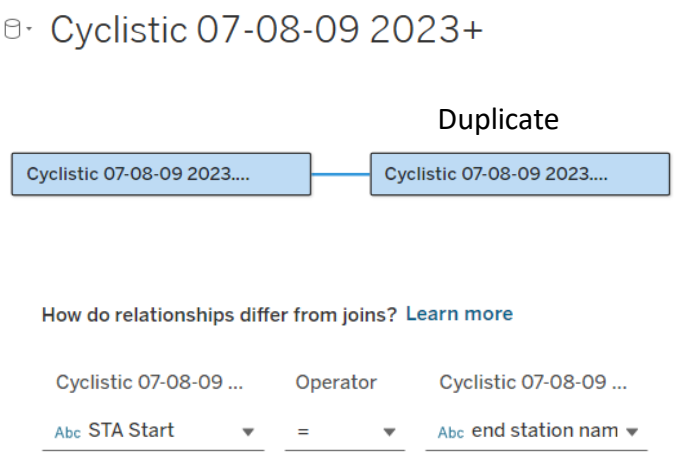
  ```
  { FIXED [Neighborhood End ]: SUM([Trip Count ]) }
  ```

- Congestion

  Congestion — Cyclistic 2022-2023+

  ```
  [Neigh_end_count]-[Neigh_start_count]
  ```

Summer database relationships

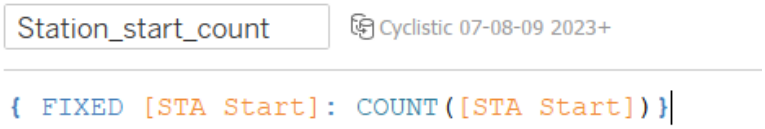


The relationship involves the CSV file for summer trends generated with BigQuery, which is matched (related) with a duplicate file. These datasets were related by matching the initial Station (STA) of each trip with the final station of each trip. This was done to calculate congestion, or the difference between bikes leaving and arriving at each neighborhood.

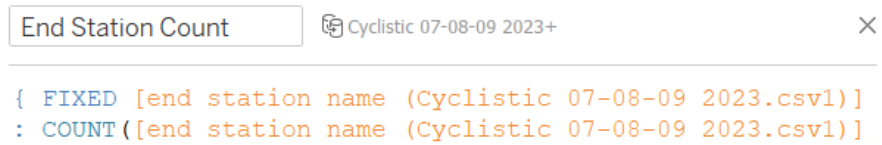
Calculating total trips per station and congestion

Calculated Fields

- Total trips per initial station, using variables from the original CSV file.



- Total trips per final station, using variables from the duplicate CSV file.



- Congestion