Activity Exemplar: Create your target table for Cyclistic

In this activity, you created target tables to consolidate and store the data you pulled from the Cyclistic datasets. These tables will allow you to develop a dashboard using Tableau in the upcoming end-of-course project activities in the next course. As a BI professional, you will need to be able to use programs such as BigQuery and Dataflow to move and analyze data with SQL. This end-of-course project showcases your ability to do just that.

The exemplar you are about to review will help you evaluate whether you completed the activity correctly. In this case, you might have discovered a solution that works just as well as the exemplar. That's great! This exemplar is an example of how a BI professional might have approached this challenge. As long as your process achieved the same results, you can move on to the next phase of the project.

If you find that the result you received is different from the exemplar provided, use the exemplar to iterate and adjust your own code.

Exploring the exemplar code

For this activity, you could run the following SQL query to create a summary table for the entire year:

```sql
SELECT
TRI.usertype,
 ZIPSTART.zip_code AS zip_code_start,
 ZIPSTARTNAME.borough borough_start,
 ZIPSTARTNAME.neighborhood AS neighborhood_start,
  ZIPEND.zip_code AS zip_code_end,
  ZIPENDNAME.borough borough_end,
 ZIPENDNAME.neighborhood AS neighborhood_end,
  DATE_ADD(DATE(TRI.starttime), INTERVAL 5 YEAR) AS start_day,
  DATE_ADD(DATE(TRI.stoptime), INTERVAL 5 YEAR) AS stop_day,
  WEA.temp AS day_mean_temperature, -- Mean temp
 WEA.wdsp AS day_mean_wind_speed, -- Mean wind speed
  WEA.prcp day_total_precipitation, -- Total precipitation
 -- Group trips into 10 minute intervals to reduces the number of rows
  ROUND(CAST(TRI.tripduration / 60 AS INT64), -1) AS trip_minutes,
  COUNT(TRI.bikeid) AS trip_count
FROM
 `bigquery-public-data.new_york_citibike.citibike_trips` AS TRI
INNER JOIN
  `bigquery-public-data.geo_us_boundaries.zip_codes` ZIPSTART
 ON ST_WITHIN(
 ST_GEOGPOINT(TRI.start_station_longitude, TRI.start_station_latitude),
ZIPSTART.zip_code_geom)
INNER JOIN
  `bigquery-public-data.geo_us_boundaries.zip_codes` ZIPEND
  ON ST_WITHIN(
ST_GEOGPOINT(TRI.end_station_longitude, TRI.end_station_latitude),
ZIPEND.zip_code_geom)
INNER JOIN
`bigquery-public-data.noaa_gsod.gsod20*` AS WEA
  ON PARSE_DATE("%Y%m%d", CONCAT(WEA.year, WEA.mo, WEA.da)) = DATE(TRI.starttime)
INNER JOIN
  -- Note! Add your zip code table name, enclosed in backticks: `example_table`
  `(insert your table name) zipcodes` AS ZIPSTARTNAME
  ON ZIPSTART.zip_code = CAST(ZIPSTARTNAME.zip AS STRING)
INNER JOIN
  -- Note! Add your zipcode table name, enclosed in backticks: `example_table`
  `(insert your table name) zipcodes` AS ZIPENDNAME
  ON ZIPEND.zip_code = CAST(ZIPENDNAME.zip AS STRING)
WHERE
```

The result of this query is a merged target table that JOINs the public datasets and the zip code table you uploaded.

Additionally, you needed to execute a query that captured data from just the summer season:

```sql
SELECT
 TRI.usertype,
 TRI.start_station_longitude,
 TRI.start_station_latitude,
 TRI.end_station_longitude,
 TRI.end_station_latitude,
 ZIPSTART.zip_code AS zip_code_start,
 ZIPSTARTNAME.borough borough_start,
 ZIPSTARTNAME.neighborhood AS neighborhood_start,
 ZIPEND.zip_code AS zip_code_end,
  ZIPENDNAME.borough borough_end,
  ZIPENDNAME.neighborhood AS neighborhood_end,
 -- Since we're using trips from 2014 and 2015, we will add 5 years to make it look recent
  DATE_ADD(DATE(TRI.starttime), INTERVAL 5 YEAR) AS start_day,
 DATE_ADD(DATE(TRI.stoptime), INTERVAL 5 YEAR) AS stop_day,
  WEA.temp AS day_mean_temperature, -- Mean temp
 WEA.wdsp AS day_mean_wind_speed, -- Mean wind speed
  WEA.prcp day_total_precipitation, -- Total precipitation
  -- We will group trips into 10 minute intervals, which also reduces the number of rows
ROUND(CAST(TRI.tripduration / 60 AS INT64), -1) AS trip_minutes,
 TRI.bikeid
FROM
 `bigquery-public-data.new_york_citibike.citibike_trips` AS TRI
INNER JOIN
`bigquery-public-data.geo_us_boundaries.zip_codes` ZIPSTART
ON ST_WITHIN(
ST_GEOGPOINT(TRI.start_station_longitude, TRI.start_station_latitude),
 ZIPSTART.zip_code_geom)
INNER JOIN
`bigquery-public-data.geo_us_boundaries.zip_codes` ZIPEND
ON ST_WITHIN(
 ST_GEOGPOINT(TRI.end_station_longitude, TRI.end_station_latitude),
ZIPEND.zip_code_geom)
INNER JOIN
 -- https://pantheon.corp.google.com/bigquery?p=bigquery-public-data&d=noaa_gsod
 `bigquery-public-data.noaa_gsod.gsod20*` AS WEA
 ON PARSE_DATE("%Y%m%d", CONCAT(WEA.year, WEA.mo, WEA.da)) = DATE(TRI.starttime)
INNER JOIN
-- Note! Add your zipcode table name, enclosed in backticks: `example_table`
`legalbi.sandbox.zipcodes` AS ZIPSTARTNAME
```

This query results into a similar table as the previous query, except it focuses on trends from July through September.

Key takeaways

Storing data from multiple sources in target tables allows you to access and use consolidated data for reporting purposes. In the Course 3 end-of-course project, you will use the table you've created in this activity to design a dashboard and share insights with the Cyclistic product development team in order to help guide their process and make informed decisions.