SQL code for Capstone Project- Google Data Analysis Course Certificate

Author: Yesim Purutcuoglu Gelgen

Date: September 2021

Data Preparation and cleansing of 12 months of divvy rides raw data.

Data set location[: https://divvy-tripdata.s3.amazonaws.com/index.html](:%20https:/divvy-tripdata.s3.amazonaws.com/index.html)

Link to Tableau Dashboard: <https://public.tableau.com/app/profile/yesim.purutcuoglu.gelgen/viz/Cyclisticbikeridesanalysis/CapstoneProject>

Link to Data Analysis Project Paper:

---------------------

WITH

  combined\_rides AS (

      #July 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202007\_tripdata`

  UNION ALL

    # August 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202008\_tripdata`

  UNION ALL

    #September 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

   timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202009\_tripdata`

  UNION ALL

    #October 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202010\_tripdata`

  UNION ALL

    # November 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

 FROM `dataanalyst2021.cyclistic\_bike\_share.202011\_tripdata`

  UNION ALL

    #December 2020

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202012\_tripdata`

  UNION ALL

    #January 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202101\_tripdata`

  UNION ALL

    #February 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

   timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202102\_tripdata`

  UNION ALL

    #March 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202103\_tripdata`

  UNION ALL

    #April 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202104\_tripdata`

  UNION ALL

    #May 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202105\_tripdata`

  UNION ALL

    #June 2021

  SELECT

    ride\_id,

    rideable\_type,

    CAST(started\_at AS TIMESTAMP ) AS started\_at,

    CAST(ended\_at AS TIMESTAMP ) AS ended\_at,

    start\_station\_name,

    CAST(start\_station\_id AS STRING ) AS start\_station\_id,

    end\_station\_name,

    CAST(end\_station\_id AS STRING ) AS end\_station\_id,

    start\_lat,

    start\_lng,

    end\_lat,

    end\_lng,

    member\_casual,

    timestamp\_diff (ended\_at,started\_at, minute) AS trip\_duration,

    EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_week

  FROM

    `dataanalyst2021.cyclistic\_bike\_share.202106\_tripdata` )

SELECT \* FROM combined\_rides

 WHERE

    NOT( start\_station\_id IS NULL

      OR end\_station\_id IS NULL

      OR start\_lat IS NULL

      OR end\_lat IS NULL

      OR start\_station\_name LIKE '%CHECKING%'

      OR end\_station\_name LIKE '%CHECKING%'

      OR trip\_duration < 1)

ORDER BY started\_at DESC

# Temp table “combined\_rides” was saved as a permanent table

**ANALYSIS**

#Calculate the min, max, sum,mean of ride\_length

SELECT

  rideable\_type,

  member\_casual,

MIN(trip\_duration) AS min\_trip\_duration,

MAX(trip\_duration) AS max\_trip\_duration,

SUM(trip\_duration) AS sum\_of\_trip\_duration,

ROUND (AVG(trip\_duration),2)  AS average\_trip\_duration

FROM

  `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY

  rideable\_type,

  member\_casual

ORDER BY average\_trip\_duration DESC;

#Calculate trip duration and percentages

SELECT

  rideable\_type,

  member\_casual,

  COUNT(\*) AS number\_of\_trips,

  ROUND(COUNT(\*) \* 100 /SUM (COUNT(\*)) OVER (),2) AS percent\_rides,

  SUM(trip\_duration) AS sum\_of\_trip\_duration,

  ROUND(SUM(trip\_duration) \* 100 /SUM (SUM(trip\_duration)) OVER (),2) AS percent\_trip\_duration

FROM

  `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY

  rideable\_type,

  member\_casual

ORDER BY

  percent\_trip\_duration DESC;

#Analysis by mode of day\_of\_week

SELECT

COUNT(\*) as number\_of\_rides,

(SELECT CASE WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=2 THEN 'Monday'

             WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=3 THEN 'Tuesday'

WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=4 THEN 'Wednesday'

             WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=5 THEN 'Thursday'

WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=6 THEN 'Friday'

             WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=7 THEN 'Saturday'

             WHEN EXTRACT(DAYOFWEEK   FROM   started\_at)=1 THEN 'Sunday' END)

  as day\_of\_week

FROM

 `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY day\_of\_week

ORDER BY

  number\_of\_rides DESC;

#Analysis by season (number of rides)

SELECT

COUNT(\*) as number\_of\_rides,

 (SELECT CASE

    WHEN EXTRACT(MONTH  FROM   started\_at)= 12 OR EXTRACT(MONTH FROM started\_at)= 1 OR EXTRACT(MONTH  FROM   started\_at)= 2 THEN 'Winter'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 3  OR EXTRACT(MONTH FROM started\_at)= 4 OR EXTRACT(MONTH  FROM   started\_at)= 5 THEN 'Spring'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 6  OR EXTRACT(MONTH FROM started\_at)= 7 OR EXTRACT(MONTH  FROM   started\_at)= 8 THEN 'Summer'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 9  OR EXTRACT(MONTH FROM started\_at)= 10 OR EXTRACT(MONTH  FROM  started\_at)= 11 THEN 'Fall'

    ELSE '-' END) as Season

FROM

 `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY Season

ORDER BY

  number\_of\_rides DESC;

#Analysis by season (trip duration, number of rides)

SELECT

COUNT(\*) as number\_of\_rides, SUM(trip\_duration) as total\_trip\_duration\_minutes, ROUND(AVG(trip\_duration)) AS  mean\_trip\_duration\_minutes, rideable\_type, member\_casual,

 (SELECT CASE

    WHEN EXTRACT(MONTH  FROM   started\_at)= 12 OR EXTRACT(MONTH FROM started\_at)= 1 OR EXTRACT(MONTH  FROM   started\_at)= 2 THEN 'Winter'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 3  OR EXTRACT(MONTH FROM started\_at)= 4 OR EXTRACT(MONTH  FROM   started\_at)= 5 THEN 'Spring'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 6  OR EXTRACT(MONTH FROM started\_at)= 7 OR EXTRACT(MONTH  FROM   started\_at)= 8 THEN 'Summer'

    WHEN EXTRACT(MONTH  FROM   started\_at)= 9  OR EXTRACT(MONTH FROM started\_at)= 10 OR EXTRACT(MONTH  FROM  started\_at)= 11 THEN 'Fall'

    ELSE '-' END) as Season

FROM

 `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY Season, rideable\_type, member\_casual

ORDER BY

   number\_of\_rides DESC;

#Analysis by time of day (Morning, Afternoon, Evening, Night)

SELECT

member\_casual,ROUND(AVG(trip\_duration)) AS  mean\_trip\_duration\_minutes,COUNT(ride\_id) AS number\_of\_rides,

 (SELECT CASE

    WHEN EXTRACT(TIME  FROM   started\_at) BETWEEN '06:00:00' AND '11:59:59' THEN 'Morning'

    WHEN EXTRACT(TIME  FROM   started\_at) BETWEEN '12:00:00' AND '16:59:59' THEN 'Afternoon'

    WHEN EXTRACT(TIME  FROM   started\_at) BETWEEN '17:00:00' AND '21:59:59' THEN 'Evening'

    ELSE'Night' END) as time\_of\_day

FROM

 `dataanalyst2021.cyclistic\_bike\_share.combined\_rides`

GROUP BY member\_casual,time\_of\_day

ORDER BY number\_of\_rides DESC