# # Combine tables

# 1.

SELECT \* FROM `case-study-411420.cyclistic.Jan 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Feb 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Mar 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Apr 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.May 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Jun 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Jul 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Aug 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Sep 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Oct 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Nov 2022`

UNION ALL

SELECT \* FROM `case-study-411420.cyclistic.Dec 2021`

# 2. click save results in the bottom pane

choose BigQuery table

choose case study dataset

give it a name (year)

# # Delete rows with nulls

# 2. delete rows with nulls

DELETE

FROM `case-study-411420.cyclistic.rides2`

WHERE

ride\_id IS NULL OR

rideable\_type IS NULL OR

started\_at IS NULL OR

ended\_at IS NULL OR

start\_station\_name IS NULL OR

start\_station\_id IS NULL OR

end\_station\_name IS NULL OR

end\_station\_id IS NULL OR

start\_lat IS NULL OR

start\_lng IS NULL OR

end\_lat IS NULL OR

end\_lng IS NULL OR

member\_casual IS NULL

# # Check for unwanted punctuation

DELETE

FROM `case-study-411420.cyclistic.rides2`

WHERE start\_station\_name LIKE '%[,- !?;:/]%' OR end\_station\_name LIKE '%[,-!?;:/]%'

OR start\_station\_id LIKE '%[,!?;:/&]%' OR end\_station\_id LIKE '%[,!?;:/&]%'

OR member\_casual LIKE '%[,.-!?;:/&]%'

# # Remove leading and trailing spaces

SELECT TRIM(start\_station\_name) AS start\_station\_name

FROM `case-study-411420.cyclistic.rides2`

# repeat for all columns

# # Change case

UPDATE `case-study-411420.cyclistic.rides2`

SET start\_station\_name = UPPER(start\_station\_name)

WHERE start\_station\_name != UPPER(start\_station\_name)

UPDATE `case-study-411420.cyclistic.rides2`

SET end\_station\_name = UPPER(end\_station\_name)

WHERE end\_station\_name != UPPER(end\_station\_name)

UPDATE `case-study-411420.cyclistic.rides2`

SET member\_casual = UPPER(member\_casual)

WHERE member\_casual != UPPER(member\_casual)

# # Create ride length field

SELECT

TIMESTAMP\_DIFF(ended\_at, started\_at, MINUTE) AS ride\_length

FROM `case-study-411420.cyclistic.rides2`

# # Create ride distance field

SELECT

ACOS(COS(RADIANS(90-start\_lat)) \* COS(RADIANS(90-end\_lat)) + SIN(RADIANS(90-start\_lat)) \* SIN(RADIANS(90-end\_lat)) \* COS(RADIANS(start\_lng-end\_lng))) \* 3959 AS ride\_dist\_mi

FROM `case-study-411420.cyclistic.rides2`

# # Create day of week field

SELECT

DAYNAME(started\_at) as day\_of\_week

FROM `case-study-411420.cyclistic.rides2`

# # Create month field

SELECT

MONTH(started\_at) as month

FROM `case-study-411420.cyclistic.rides2`

# # Analyze ride length

SELECT

member\_casual,

COUNT(ride\_length)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

MAX(ride\_length)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

MIN(ride\_length)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

AVG(ride\_length)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

# # Analyze ride distance

SELECT

member\_casual,

COUNT(ride\_dist\_mi)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

MAX(ride\_ dist\_mi)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

MIN(ride\_ dist\_mi)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

SELECT

member\_casual,

AVG(ride\_ dist\_mi)

FROM `case-study-411420.cyclistic.rides2`

GROUP BY member\_casual

# # Analyze day of week

SELECT

  member\_casual,

  day\_of\_week,

  COUNT(day\_of\_week) AS days

FROM `case-study-411420.cyclistic.rides2`

GROUP BY day\_of\_week, member\_casual

# # Analyze month

SELECT

  member\_casual,

  month,

  COUNT(month) AS months

FROM `case-study-411420.cyclistic.rides2`

GROUP BY month, member\_casual

# # Analyze bike type

SELECT

  rideable\_type,

  member\_casual,

  COUNT(rideable\_type) AS bikes

FROM `case-study-411420.cyclistic.rides2`

GROUP BY rideable\_type, member\_casual

# # Analyze station

SELECT

  start\_station\_name,

  member\_casual,

  COUNT(start\_station\_name) AS bikes

FROM `case-study-411420.cyclistic.rides2`

GROUP BY start\_station\_name, member\_casual