--This is to find out the **peak hours**

SELECT

  member\_casual,

  EXTRACT(HOUR FROM started\_at) AS hour\_of\_day,

  COUNT(ride\_id) AS total\_rides

FROM cyclistic\_data.ride\_data

GROUP BY member\_casual, hour\_of\_day

ORDER BY hour\_of\_day;

--This is to find out favored **ride type**

SELECT

  member\_casual,

  rideable\_type,

  COUNT(ride\_id) AS total\_rides

FROM cyclistic\_data.ride\_data

GROUP BY member\_casual, rideable\_type;

--This is to find out **top locations**

SELECT

  member\_casual,

  start\_station\_name,

  COUNT(ride\_id) AS ride\_count

FROM cyclistic\_data.ride\_data

WHERE start\_station\_name IS NOT NULL

GROUP BY member\_casual, start\_station\_name

ORDER BY ride\_count DESC

LIMIT 10;

--This is to find out favored **days of the week**

SELECT

  member\_casual,

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

  CASE

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

    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'

  END AS day\_of\_week\_name,

  COUNT(ride\_id) AS total\_rides

FROM cyclistic\_data.ride\_data

GROUP BY member\_casual, day\_of\_week\_num, day\_of\_week\_name

ORDER BY day\_of\_week\_num;

--This is to find out the **avg ride duration**

SELECT member\_casual,

COUNT(ride\_id) AS total\_rides,

AVG(TIMESTAMP\_DIFF(ended\_at, started\_at, MINUTE)) AS avg\_ride\_duration

FROM cyclistic\_data.ride\_data

GROUP BY member\_casual;

--To find out **monthly totals**

WITH MonthList AS (

    SELECT '2023-12' AS ride\_month

    UNION ALL

    SELECT '2024-01'

    UNION ALL

    SELECT '2024-02'

    UNION ALL

    SELECT '2024-03'

    UNION ALL

    SELECT '2024-04'

    UNION ALL

    SELECT '2024-05'

    UNION ALL

    SELECT '2024-06'

    UNION ALL

    SELECT '2024-07'

    UNION ALL

    SELECT '2024-08'

    UNION ALL

    SELECT '2024-09'

    UNION ALL

    SELECT '2024-10'

    UNION ALL

    SELECT '2024-11'

    UNION ALL

    SELECT '2024-12'

),

RideCounts AS (

    SELECT

        member\_casual AS rider\_type,

        FORMAT\_TIMESTAMP('%Y-%m', TIMESTAMP(started\_at)) AS ride\_month,

        COUNT(\*) AS rider\_count

    FROM

        cyclistic\_data.ride\_data

    WHERE

        started\_at >= '2023-12-01' AND started\_at < '2025-01-01'

    GROUP BY

        member\_casual, FORMAT\_TIMESTAMP('%Y-%m', TIMESTAMP(started\_at))

)

SELECT

    ml.ride\_month,

    rc.rider\_type,

    COALESCE(rc.rider\_count, 0) AS rider\_count

FROM

    MonthList ml

LEFT JOIN RideCounts rc

    ON ml.ride\_month = rc.ride\_month

ORDER BY

    rc.rider\_type DESC,

    ml.ride\_month;