1. Viewing the Complete Customer Table

```
SELECT *
FROM `ekartdataset.customer_t`
```

2. Counting Total Rows in the Customer Table

```
SELECT COUNT(*) AS total_rows
FROM `ekartdataset.customer_t`
```

3. Counting Customers by Occupation (Sorted Descending)

```
SELECT OCCUPATION, COUNT(*) AS count
FROM `ekartdataset.customer_t`
GROUP BY OCCUPATION
ORDER BY count DESC;
```

4. Viewing Distinct Occupation Categories

```
SELECT DISTINCT OCCUPATION
FROM `ekartdataset.customer_t`;
```

5. Viewing Sample Records from the Sales Table (First 20 Rows)

```
SELECT *
FROM `ekartdataset.sales_t`
LIMIT 20;
```

6. Counting Total Rows in the Sales Table

```
SELECT COUNT(*) AS total_rows
FROM `ekartdataset.sales_t`
```

7. Joining Customer and Sales Tables to View Key Info

```
SELECT
  c.customer_id,
  c.first_name,
  c.last_name,
  c.occupation,
```

```
  s.revenue,
  s.total_orders
FROM `ekart-assignment.ekartdataset.customer_t` c
JOIN `ekart-assignment.ekartdataset.sales_t` s
  ON c.customer_id = s.customer_id
LIMIT 10;
```

8. Cleaning Occupation Field in the Joined Data

```
SELECT
  c.customer_id,
  c.first_name,
  c.last_name,
  CASE
    WHEN c.occupation = 'Blue Collar' THEN 'Blue-Collar'
    ELSE c.occupation
  END AS cleaned_occupation,
  s.revenue,
  s.total_orders
FROM `ekart-assignment.ekartdataset.customer_t` c
JOIN `ekart-assignment.ekartdataset.sales_t` s
  ON c.customer_id = s.customer_id
LIMIT 10;
```

9. Creating Age Categories Based on Date of Birth

```
SELECT
  c.customer_id,
  CASE
    WHEN DATE_DIFF(CURRENT_DATE(), c.DOB, YEAR) <= 19 THEN '19 or below'
    WHEN DATE_DIFF(CURRENT_DATE(), c.DOB, YEAR) BETWEEN 20 AND 29 THEN '20-29'
    WHEN DATE_DIFF(CURRENT_DATE(), c.DOB, YEAR) BETWEEN 30 AND 39 THEN '30-39'
    WHEN DATE_DIFF(CURRENT_DATE(), c.DOB, YEAR) BETWEEN 40 AND 49 THEN '40-49'
    ELSE '50 and above'
  END AS age_category
FROM `ekart-assignment.ekartdataset.customer_t` c
JOIN `ekart-assignment.ekartdataset.sales_t` s
  ON c.customer_id = s.customer_id
LIMIT 10;
```

10. Final Cleaned Dataset with Demographic and Behavioral Insights

- Combines customer and sales data

- Cleans occupation values

- Computes age and age category
- Adds detailed order and revenue breakdowns

```
SELECT
  c.CUSTOMER_ID,
  CONCAT(c.FIRST_NAME, ' ', c.LAST_NAME) AS CUSTOMER_NAME,
  c.EMAIL_ID,

  -- Cleaned OCCUPATION field
  CASE
    WHEN c.OCCUPATION = 'Blue Collar' THEN 'Blue-Collar'
    ELSE c.OCCUPATION
  END AS CLEANED_OCCUPATION,

  c.DOB,

  -- Calculated AGE
  DATE_DIFF(CURRENT_DATE(), DATE(c.DOB), YEAR) AS AGE,

  -- Age Categories
  CASE
    WHEN DATE_DIFF(CURRENT_DATE(), DATE(c.DOB), YEAR) <= 19 THEN '19 & below'
    WHEN DATE_DIFF(CURRENT_DATE(), DATE(c.DOB), YEAR) BETWEEN 20 AND 29
THEN '20-29'
    WHEN DATE_DIFF(CURRENT_DATE(), DATE(c.DOB), YEAR) BETWEEN 30 AND 39
THEN '30-39'
    WHEN DATE_DIFF(CURRENT_DATE(), DATE(c.DOB), YEAR) BETWEEN 40 AND 49
THEN '40-49'
    ELSE '50 and above'
  END AS AGE_CATEGORY,

  -- Sales info
  s.TOTAL_ORDERS,
  s.REVENUE,
  s.AVERAGE_ORDER_VALUE,
  s.CARRIAGE_REVENUE,
  s.FIRST_ORDER_DATE,
  s.LATEST_ORDER_DATE,

  -- Additional fields
  s.AVERAGESHIPPING,
  s.AVGDAYSBETWEENORDERS,
  s.DAYSSINCELASTORDER,
  s.MONDAY_ORDERS,
```

```
  s.TUESDAY_ORDERS,
  s.WEDNESDAY_ORDERS,
  s.THURSDAY_ORDERS,
  s.FRIDAY_ORDERS,
  s.SATURDAY_ORDERS,
  s.SUNDAY_ORDERS,
  s.MONDAY_REVENUE,
  s.TUESDAY_REVENUE,
  s.WEDNESDAY_REVENUE,
  s.THURSDAY_REVENUE,
  s.FRIDAY_REVENUE,
  s.SATURDAY_REVENUE,
  s.SUNDAY_REVENUE,
  s.WEEK1_DAY01_DAY07_ORDERS,
  s.WEEK2_DAY08_DAY15_ORDERS,
  s.WEEK3_DAY16_DAY23_ORDERS,
  s.WEEK4_DAY24_DAY31_ORDERS,
  s.WEEK1_DAY01_DAY07_REVENUE,
  s.WEEK2_DAY08_DAY15_REVENUE,
  s.WEEK3_DAY16_DAY23_REVENUE,
  s.WEEK4_DAY24_DAY31_REVENUE,
  s.TIME_0000_0600_ORDERS,
  s.TIME_0601_1200_ORDERS,
  s.TIME_1200_1800_ORDERS,
  s.TIME_1801_2359_ORDERS,
  s.TIME_0000_0600_REVENUE,
  s.TIME_0601_1200_REVENUE,
  s.TIME_1200_1800_REVENUE,
  s.TIME_1801_2359_REVENUE
FROM `ekart-assignment.ekartdataset.customer_t` c
JOIN `ekart-assignment.ekartdataset.sales_t` s
  ON c.customer_id = s.customer_id
```