```
## Creating queries answering individual questions with the help of temporary tables. The questions are as follows:

# 1. What is the total revenue contribution by each country?

WITH Revenue_by_country AS
(
  SELECT
    Country,
    CAST(SUM(Revenue) AS INT64) AS Total_revenue
  FROM
    manikanth-sql.customer.customer_spending_behavior
  GROUP BY
    Country
),

# 2. What is the total revenue contribution by age and gender? To make things simple, age is grouped into categories using CASE function.

----Add index as comment to display the results of age group. It is included in this table as a primary key which is to be used later on while using JOIN functions. Also, don't forget to comment entire "Purchasing_behavior_age_group" table due to the presence of index in it while running this query.

Revenue_by_age_gender AS
(
  SELECT
    Index,
    Country,
    CASE
      WHEN Age < 30
      THEN 'Young'
      WHEN Age BETWEEN 30 AND 45
      THEN 'Adult'
      WHEN Age BETWEEN 46 AND 55
      THEN 'Middle aged'
      ELSE 'Old'
    END AS Age_group,
    Gender,
    CAST(SUM(Revenue) AS INT64) AS Total_revenue
  FROM
    manikanth-sql.customer.customer_spending_behavior
  GROUP BY
    Index,
    Country,
    Age_group,
    Gender
  ORDER BY
    Country,
    Total_revenue DESC
),

# 3. What is the total revenue contribution by product category and sub category?

Revenue_by_category AS
(
```

```
  SELECT
    Country,
    Product_category,
    Sub_category,
    CAST(SUM(Revenue) AS INT64) AS Total_revenue,
  FROM
    manikanth-sql.customer.customer_spending_behavior
  GROUP BY
    Country,
    Product_category,
    Sub_category
  ORDER BY
    Country
),

# 4. Understanding how gender, and age group effects purchasing behavior of customers based
on the price and the units of items sold by year and location. In this query, JOIN functions
are used.

Purchasing_behavior_age_gender AS
(
  SELECT
    Original_table.Country,
    Original_table.Year,
    Original_table.Product_category,
    Original_table.Gender,
    Second_table.Age_group,
    CAST(AVG(Original_table.Unit_price) AS INT64) AS Avg_unit_price,
    SUM(Original_table.Quantity) AS Units_purchased
  FROM
    manikanth-sql.customer.customer_spending_behavior AS Original_table
  INNER JOIN
    Revenue_by_age_gender AS Second_table ON
    Second_table.Index = Original_table.Index
  GROUP BY
    1,2,3,4,5 --Here, 1,2,3 & 4 are the names of columns from SELECT in the same order.
  ORDER BY
    Original_table.Country,
    Original_table.Year,
    Original_table.Gender,
    Second_table.Age_group
),

# 5. Creating a new table to investigate sales trend.

Sales_trend AS
(
  SELECT
    Country,
    Date,
    Product_category,
    CAST(Revenue AS INT64) AS Revenue
  FROM
    manikanth-sql.customer.customer_spending_behavior
)
```

```
# 6. Select tables to return values.

SELECT
  *
FROM
  Purchasing_behavior_age_gender
```