Requirement1: *the total amount spent and the country for the Pending delivery status for each country.*

Assumption: Assuming that the customers having at least one shipping with Pending status is included in the reporting

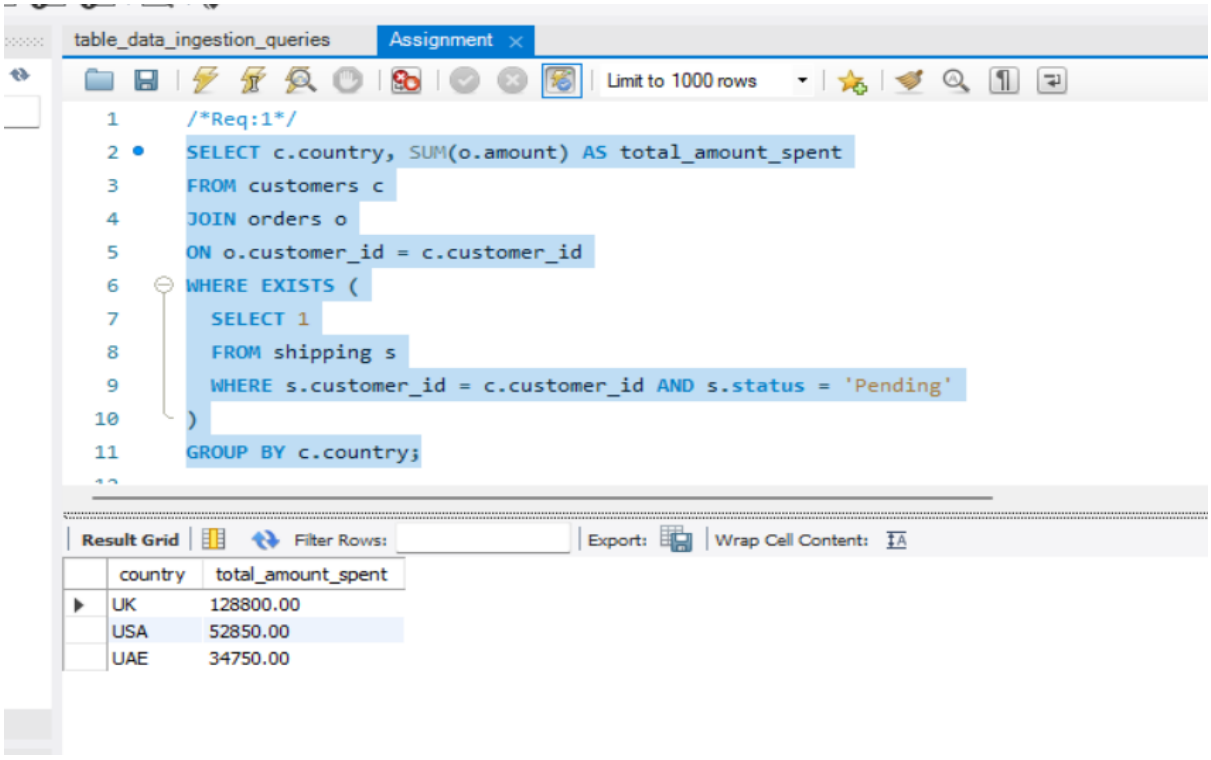
***Test Cases**:*

Detailed test scenarios and test cases for this requirement are documented in the attached Excel file Anwita_take_home_Assignment_test_cases.xlsx, Sheet: Req1.

SQL query to test the Expected value:

```
SELECT c.country, SUM(o.amount) AS total_amount_spent
FROM customers c
JOIN orders o
ON o.customer_id = c.customer_id
WHERE EXISTS (
 SELECT 1
 FROM shipping s
 WHERE s.customer_id = c.customer_id AND s.status = 'Pending'
)
GROUP BY c.country;
```

Test result:



Requirement Gap:

This requirement does not clearly define whether Pending delivery status applies at order level or customer level. Since shipping status is available only at customer level and not mapped to Order_ID, the calculation includes all orders of a customer when at least one Pending delivery record exists. This may not reflect the accurate amounts for pending order.

Requirement2: *the total number of transactions, total quantity sold, and total amount spent for each customer, along with the product details.*

Assumptions:

Assuming the report is validated at customer and product level, i.e. one row per customer per product

Assuming each order row represents one unit sold.

Transaction is not specifically defined in the data tables so it is being counted using distinct Order IDs.

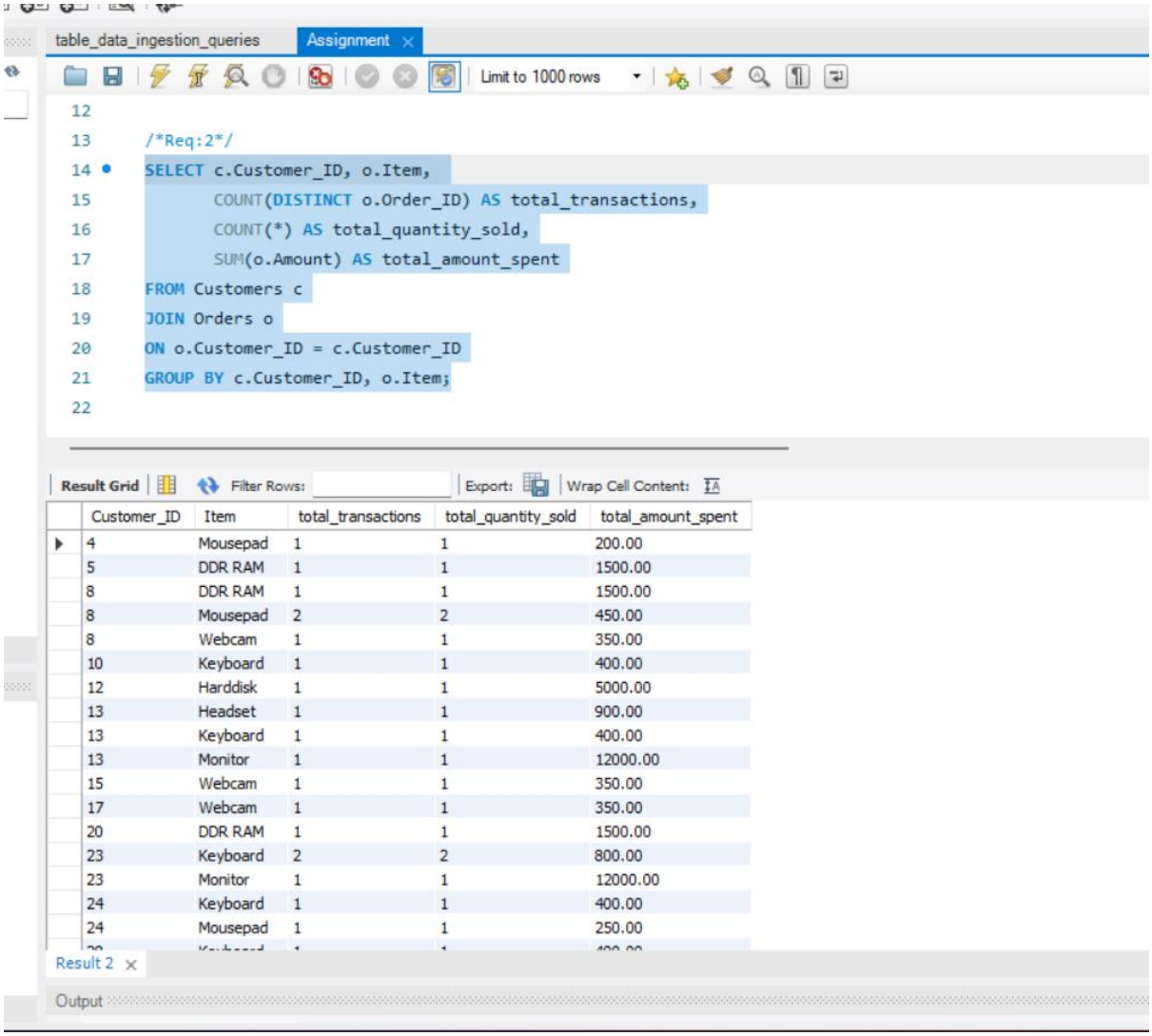
***Test Cases**:*

Detailed test scenarios and test cases for this requirement are documented in the attached Excel file Anwita_take_home_Assignment_test_cases.xlsx, Sheet: Req2.

SQL query to test the Expected value:

```
SELECT c.Customer_ID, o.Item,
    COUNT(DISTINCT o.Order_ID) AS total_transactions,
    COUNT(*) AS total_quantity_sold,
    SUM(o.Amount) AS total_amount_spent
FROM Customers c
JOIN Orders o
ON o.Customer_ID = c.Customer_ID
GROUP BY c.Customer_ID, o.Item;
```

Test result:



Requirement Gaps:

The requirement does not clearly specify whether totals should be calculated at customer level or customer-product level

Also, transaction is not defined as to how the total number of transactions should be calculated.

Quantity column is not present in any data table

Requirement3: *the maximum product purchased for each country*.

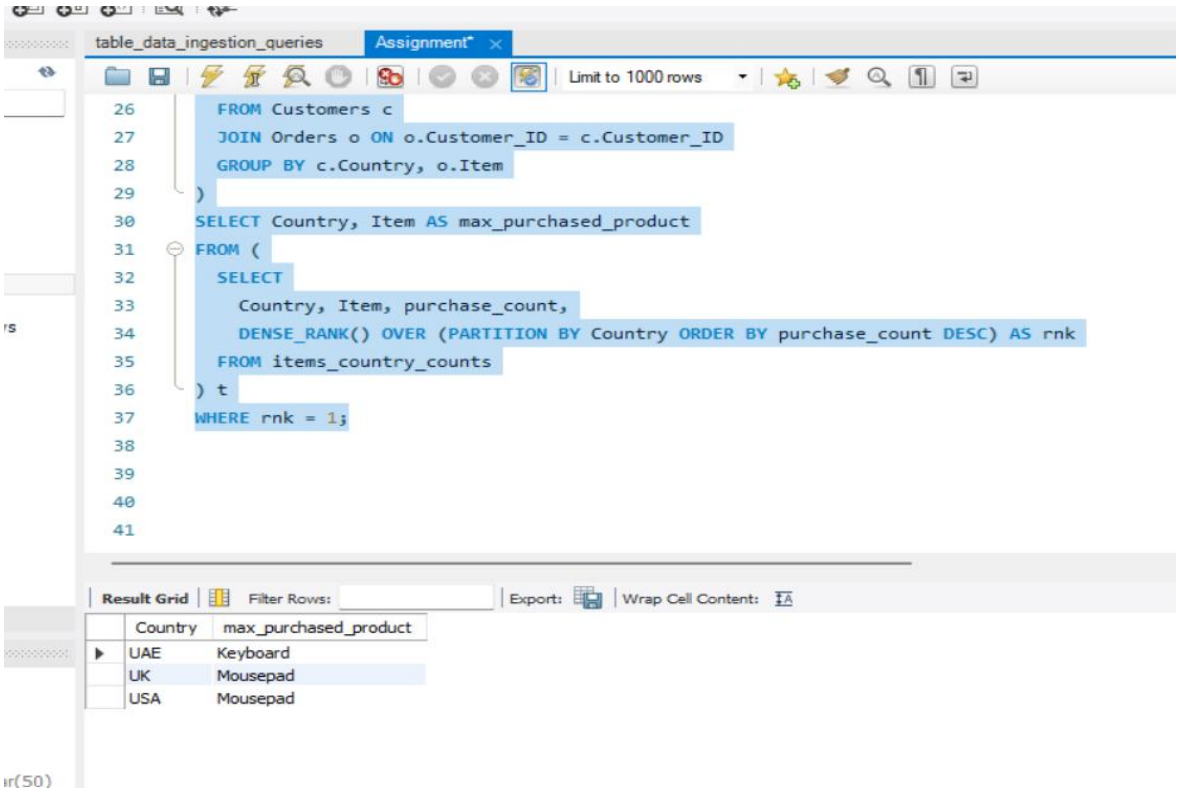
***Test Cases**:*

Detailed test scenarios and test cases for this requirement are documented in the attached Excel file Anwita_take_home_Assignment_test_cases.xlsx, Sheet: Req3.

SQL query to test the Expected value:

```
WITH items_country_counts AS (
  SELECT c.Country, o.Item, COUNT(*) AS purchase_count
  FROM Customers c
  JOIN Orders o ON o.Customer_ID = c.Customer_ID
  GROUP BY c.Country, o.Item
)
SELECT Country, Item AS max_purchased_product
FROM (
  SELECT
    Country, Item, purchase_count,
    DENSE_RANK() OVER (PARTITION BY Country ORDER BY purchase_count
DESC) AS rnk
  FROM items_country_counts
) t
WHERE rnk = 1;
```

Test result:



Requirement Gap:

The requirement does not define what “maximum product purchased” means (maximum by quantity or number of orders) and since the data source does not have a quantity field, the report can only determine the maximum product using count of order records. This may result in incorrect value reporting

Requirement4: *the most purchased product based on the age category less than 30 and above 30.*

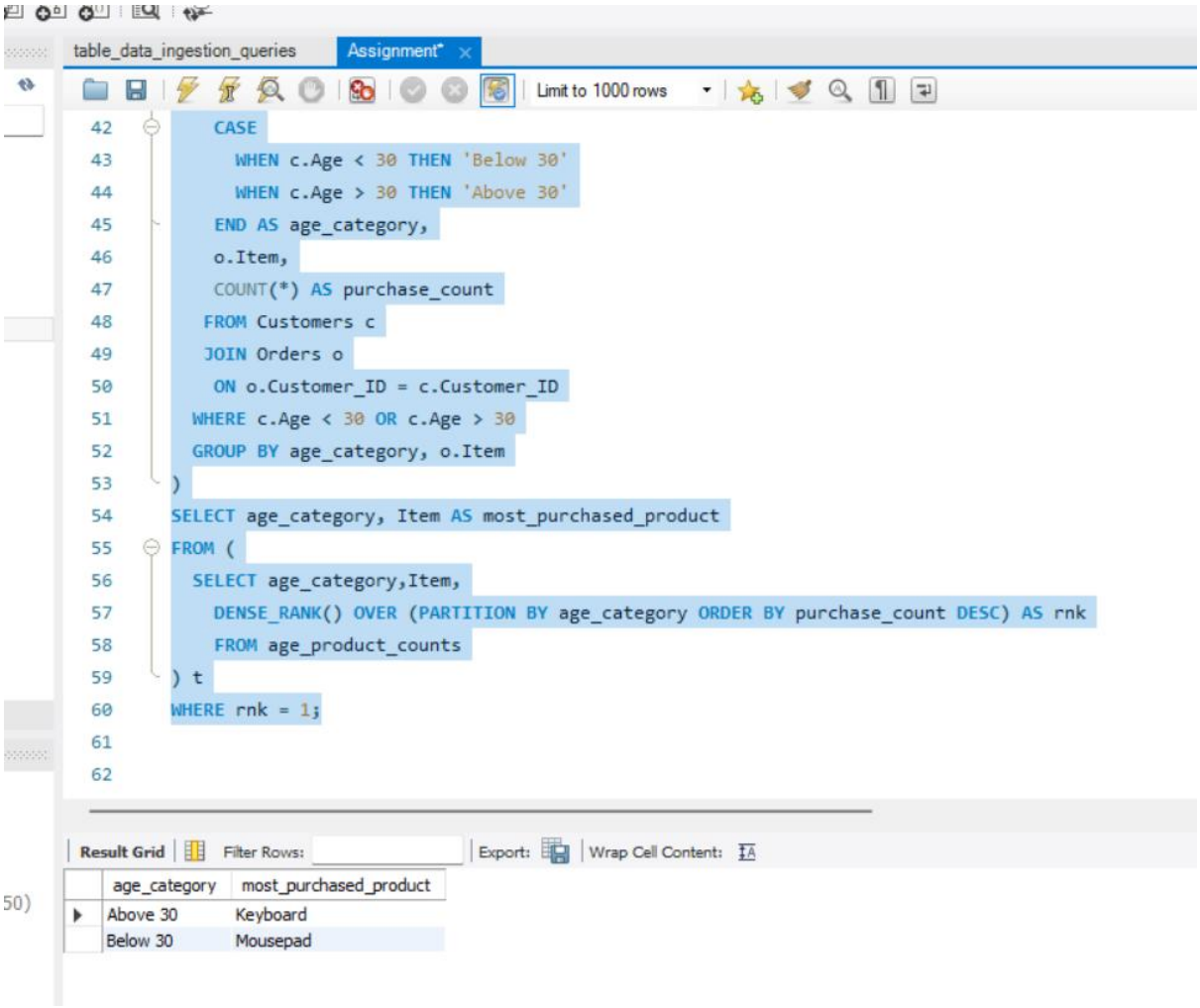
***Test Cases**:*

Detailed test scenarios and test cases for this requirement are documented in the attached Excel file Anwita_take_home_Assignment_test_cases.xlsx, Sheet: Req4.

SQL query to test the Expected value:

```
WITH age_product_counts AS (
  SELECT
    CASE
      WHEN c.Age < 30 THEN 'Below 30'
      WHEN c.Age > 30 THEN 'Above 30'
    END AS age_category,
    o.Item,
    COUNT(*) AS purchase_count
  FROM Customers c
  JOIN Orders o
    ON o.Customer_ID = c.Customer_ID
  WHERE c.Age < 30 OR c.Age > 30
  GROUP BY age_category, o.Item
)
SELECT age_category, Item AS most_purchased_product
FROM (
  SELECT age_category,Item,
    DENSE_RANK() OVER (PARTITION BY age_category ORDER BY purchase_count DESC) AS rnk
    FROM age_product_counts
) t
WHERE rnk = 1;
```

Test result:



Requirement Gap:

The requirement does not specify how customers aged exactly 30 should be handled

Requirement5: *the country that had minimum transactions and sales amount*

***Test Cases**:*

Detailed test scenarios and test cases for this requirement are documented in the attached Excel file Anwita_take_home_Assignment_test_cases.xlsx, Sheet: Req5.

SQL query to test the Expected value:

```
SELECT c.country
FROM Customers c
JOIN Orders o
ON o.Customer_ID = c.Customer_ID
GROUP BY c.Country
ORDER BY COUNT(DISTINCT o.Order_ID) ASC, SUM(o.Amount) ASC
LIMIT 1;
```

Test result: