**Vetty Data Engineer (assignment solutions)**

**Transactions table:**

**A screen shot of a computer

Description automatically generated**

**Items table:**

**A screen shot of a computer program

Description automatically generated**

**Q.1 What is the count of purchases per month (excluding refunded purchases)?**

**Logi**c**: I Filtered transactions where refund\_item IS NULL. Group by the month of purchase\_time and counted rows.**

**Approach: Use DATE\_FORMAT on purchase\_time and COUNT().**

**A screen shot of a computer

Description automatically generated**

**Q.2 . How many stores receive at least 5 orders/transactions in October 2020?**

**Logic: I Filtered transactions in October 2020 using DATE\_FORMAT. Group by store\_id and count orders. Apply HAVING to keep only stores with ≥5 transactions.**

**Approach: Use WHERE for filtering, GROUP BY for aggregation, and HAVING for threshold filtering because I used Group BY here.**

**A screen shot of a computer code

Description automatically generated**

**Q.3 . For each store, what is the shortest interval (in min) from purchase to refund time?**

**Logic: For this, I calculated the difference between purchase\_time and refund\_item using TIMESTAMPDIFF. I grouped by store\_id to ensure the calculation was done store-wise, and I picked the minimum time difference for each store.**

**A screenshot of a computer

Description automatically generated**

**Q.4 What is the gross\_transaction\_value of every store’s first order?**

**Logic: To solve this, I first found the earliest purchase (MIN(purchase\_time)) for each buyer using a CTE. Then, I joined this result with the transactions table to get the items from those first purchases. After that, I joined with the items table to fetch the item names. Finally, I counted how many times each item was bought and sorted them to find the most popular item, returning only the top one.**

**A screenshot of a computer program

Description automatically generated**

**Q.5 . What is the most popular item name that buyers order on their first purchase?**

**Logic: I started by finding each buyer’s first purchase using the earliest purchase\_time. Then, I joined this data with the items table to get item names and counted how often each item appeared. Finally, I sorted the result to find the most popular item.**

**A computer screen with white text

Description automatically generated**

**Q.6 Create a flag in the transaction items table indicating whether the refund can be processed or not. The condition for a refund to be processed is that it has to happen within 72 of Purchase time. Expected Output: Only 1 of the three refunds would be processed in this case**

**Logic: To check if a refund can be processed, I calculated the time difference in hours between purchase\_time and refund\_item using TIMESTAMPDIFF. If the difference was less than or equal to 72 hours, I marked it as "Processed"; otherwise, "Not Processed."**

**A screen shot of a computer

Description automatically generated**

**Q.7 Create a rank by buyer\_id column in the transaction items table and filter for only the second purchase per buyer. (Ignore refunds here) Expected Output: Only the second purchase of buyer\_id 3 should the output**

**Logic: I ranked each transaction for every buyer based on the order of purchase\_time. Using ROW\_NUMBER(), I ensured each buyer's transactions were ordered. Then, I filtered out the rows where the rank was 2 to get the second purchase.**

**A black screen with white text

Description automatically generated**

**Q.8 How will you find the second transaction time per buyer (don’t use min/max; assume there were more transactions per buyer in the table) Expected Output: Only the second purchase of buyer\_id along with a timestamp**

**Logic: Similar to Question 7, I ranked the transactions of each buyer by their purchase\_time using ROW\_NUMBER(). Then, I filtered for rows with a rank of 2 to get the second transaction for each buyer.**

**A screen shot of a computer

Description automatically generated**