BrightLight Data Analytics Coding Practical

Practical 1: SQL Fundamentals (Snowflake-Basic SQL Syntax)

The following questions are designed to help you build a strong foundation in basic SQL syntax. You are provided with a dataset named **retail_sales_dataset.csv**. Upload this dataset to your Snowflake account and use it to answer the questions below.

Please follow the instructions below carefully:

1. Write one SQL query per question.
   Use proper formatting and indentation where necessary.
2. Each question tests a specific SQL concept.
   Read the question carefully and apply only the concept being tested (e.g., SELECT, WHERE, GROUP BY, etc.).
3. Do not combine multiple SQL concepts unless instructed to.
   For example, if the question asks for a SELECT DISTINCT, don't use WHERE unless specified.
4. Use the correct column names as provided in the dataset
5. Expected Output Columns are provided for each question.
   Your query must return exactly those columns in the result.
6. Don't worry about the actual data.
   Focus on getting the SQL syntax right.
7. Submit your completed SQL queries as a .sql

Table 1: Outlines the name and descriptions of the columns in the provided dataset *"retail_sales_dateset.csv"*

Column Name	Description
Transaction_ID	Unique identifier for each transaction.
Date	The date on which the transaction occurred (format: YYYY-MM-DD).
Customer_ID	Unique identifier for the customer making the purchase.
Gender	Gender of the customer (e.g., Male, Female).
Age	Age of the customer at the time of the transaction.
Product_Category	Category of the product purchased (e.g., Beauty, Clothing, Electronics).
Quantity	Number of product units purchased in the transaction.
Price_per_Unit	Cost of a single unit of the product (in the currency used).
Total_Amount	Total amount spent for the transaction (Quantity × Price per Unit).

Questions

1. SELECT Statement

Q1. Display all columns for all transactions.
Expected output: All columns

Q2. Display only the Transaction ID, Date, and Customer ID for all records.
Expected output: Transaction ID, Date, Customer ID

---

2. SELECT DISTINCT Statement

Q3. Display all the distinct product categories in the dataset.
Expected output: Product Category

Q4. Display all the distinct gender values in the dataset.
Expected output: Gender

---

3. WHERE Clause

Q5. Display all transactions where the Age is greater than 40.
Expected output: All columns

Q6. Display all transactions where the Price per Unit is between 100 and 500.
Expected output: All columns

Q7. Display all transactions where the Product Category is either 'Beauty' or 'Electronics'.
Expected output: All columns

Q8. Display all transactions where the Product Category is **not** 'Clothing'.
Expected output: All columns

Q9. Display all transactions where the Quantity is greater than or equal to 3.
Expected output: All columns

---

4. Aggregate Functions

Q10. Count the total number of transactions.
Expected output: Total_Transactions

Q11. Find the average Age of customers.
Expected output: Average_Age

Q12. Find the total quantity of products sold.
Expected output: Total_Quantity

Q13. Find the maximum Total Amount spent in a single transaction.
Expected output: Max_Total_Amount

Q14. Find the minimum Price per Unit in the dataset.
Expected output: Min_Price_per_Unit

---

5. GROUP BY Statement

Q15. Find the number of transactions per Product Category.
Expected output: Product Category, Transaction_Count

Q16. Find the total revenue (Total Amount) per gender.
Expected output: Gender, Total_Revenue

Q17. Find the average Price per Unit per product category.
Expected output: Product Category, Average_Price

---

6. HAVING Clause

Q18. Find the total revenue per product category where total revenue is greater than 10,000.
Expected output: Product Category, Total_Revenue

Q19. Find the average quantity per product category where the average is more than 2.
Expected output: Product Category, Average_Quantity

---

7. CASE Statement

Q20. Display a column called Spending_Level that shows 'High' if Total Amount > 1000, otherwise 'Low'.
Expected output: Transaction ID, Total Amount, Spending_Level

Q21. Display a new column called Age_Group that labels customers as:

- 'Youth' if Age < 30
- 'Adult' if Age is between 30 and 59
- 'Senior' if Age >= 60
  Expected output: Customer ID, Age, Age_Group