

Retail Sales Dataset

Sonali Matadin

--1. SELECT Statement

--Q1. Display all columns for all transactions.

```
SELECT *
FROM RETAIL.SALES.DATASET;
```

```
2 --Q1. Display all columns for all transactions.
3 SELECT *
4 FROM RETAIL.SALES.DATASET;
5
```

Results (just now)

TableChart

1,000 rows33ms

	# TRANSACTION_ID	📅 DATE	🔗 CUSTOMER_ID	👤 GENDER	# AGE	🔗 PRODUCT_CATEGORY	# QUANTITY	# PRICE_PER_UNIT	# TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000
3	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
4	4	2023-05-21	CUST004	Male	37	Clothing	1	500	500
5	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
6	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
7	7	2023-03-13	CUST007	Male	46	Clothing	2	25	50
8	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
9	9	2023-12-13	CUST009	Male	63	Electronics	2	300	600

--Q2. Display Transaction ID, Date, Customer ID.

```
SELECT TRANSACTION_ID,
       DATE,
       CUSTOMER_ID
FROM RETAIL.SALES.DATASET;
```

```
8 SELECT TRANSACTION_ID,
9       DATE,
10      CUSTOMER_ID
11 FROM RETAIL.SALES.DATASET;
12
13
```

Results (just now)

TableChart

1,000 rows36ms

	# TRANSACTION_ID	📅 DATE	🔗 CUSTOMER_ID
1	1	2023-11-24	CUST001
2	2	2023-02-27	CUST002
3	3	2023-01-13	CUST003
4	4	2023-05-21	CUST004
5	5	2023-05-06	CUST005
6	6	2023-04-25	CUST006
7	7	2023-03-13	CUST007

-- 2. SELECT DISTINCT Statement

--Q3. Distinct product categories

```
SELECT DISTINCT PRODUCT_CATEGORY
```

```
FROM RETAIL.SALES.DATASET;
```

```
15      -- 2. SELECT DISTINCT Statement
16      --Q3. Distinct product categories
17      SELECT DISTINCT PRODUCT_CATEGORY
18      FROM RETAIL.SALES.DATASET;
19
```

Results (just now)

Table Chart

	PRODUCT_CATEGORY
1	Clothing
2	Beauty
3	Electronics

--Q4. Distinct gender values

```
SELECT DISTINCT GENDER
```

```
FROM RETAIL.SALES.DATASET;
```

```
21
22      --Q4. Distinct gender values
23      SELECT DISTINCT GENDER
24      FROM RETAIL.SALES.DATASET;
25
```

Results (just now)

Table Chart

	GENDER
1	Male
2	Female

--3. WHERE Clause

--Q5. Age greater than 40

SELECT *

FROM RETAIL.SALES.DATASET

WHERE AGE > 40;

```
26
27 --3. WHERE Clause
28 --Q5. Age greater than 40
29 SELECT *
30 FROM RETAIL.SALES.DATASET
31 WHERE AGE > 40;
32
```

Results (just now)

Table Chart 534 rows 482ms

	# TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
2	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
3	7	2023-03-13	CUST007	Male	46	Clothing	2	25	50
4	9	2023-12-13	CUST009	Male	63	Electronics	2	300	600
5	10	2023-10-07	CUST010	Female	52	Clothing	4	50	200
6	14	2023-01-17	CUST014	Male	64	Clothing	4	30	120

Q6. Price per Unit between 100 and 500

SELECT *

FROM RETAIL.SALES.DATASET

WHERE PRICE_PER_UNIT BETWEEN 100 AND 500;

```
34 --Q6. Price per Unit between 100 and 500
35 SELECT *
36 FROM RETAIL.SALES.DATASET
37 WHERE PRICE_PER_UNIT BETWEEN 100 AND 500;
```

Results (just now)

Table Chart 396 rows 76ms

	# TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000
2	4	2023-05-21	CUST004	Male	37	Clothing	1	500	500
3	9	2023-12-13	CUST009	Male	63	Electronics	2	300	600
4	13	2023-08-05	CUST013	Male	22	Electronics	3	500	1500
5	15	2023-01-13	CUST015	Female	48	Electronics	4	500	2000

Q7. Product Category is Beauty OR Electronics

SELECT *

FROM RETAIL.SALES.DATASET

WHERE PRODUCT_CATEGORY IN ('Beauty', 'Electronics');

```
39 --Q7. Product Category is Beauty OR Electronics
40 SELECT *
41 FROM RETAIL.SALES.DATASET
42 WHERE PRODUCT_CATEGORY IN ('Beauty', 'Electronics');
```

Results (just now)

Table Chart 649 rows 69ms

	# TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
3	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
4	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30

--Q8. Product Category NOT Clothing

SELECT *

FROM RETAIL.SALES.DATASET

WHERE PRODUCT_CATEGORY <> 'Clothing';

```
45 --Q8. Product Category NOT Clothing
46 SELECT *
47 FROM RETAIL.SALES.DATASET
48 WHERE PRODUCT_CATEGORY <> 'Clothing';
49
```

results (just now)

Table Chart

649 rows 34ms

	# TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
3	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
4	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
5	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100

Q9. Quantity ≥ 3

SELECT *

FROM RETAIL.SALES.DATASET

WHERE QUANTITY ≥ 3 ;

```
51 --Q9. Quantity ≥ 3
52 SELECT *
53 FROM RETAIL.SALES.DATASET
54 WHERE QUANTITY >= 3;
```

results (just now)

Table Chart

504 rows 464ms

	# TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
3	10	2023-10-07	CUST010	Female	52	Clothing	4	50	200
4	12	2023-10-30	CUST012	Male	35	Beauty	3	25	75
5	13	2023-08-05	CUST013	Male	22	Electronics	3	500	1500
6	14	2023-01-17	CUST014	Male	64	Clothing	4	30	120
7	15	2023-01-16	CUST015	Female	42	Electronics	4	500	2000
8	16	2023-02-17	CUST016	Male	19	Clothing	3	500	1500
9	17	2023-04-22	CUST017	Female	27	Clothing	4	25	100
10	20	2023-11-05	CUST020	Male	22	Clothing	3	300	900
11	23	2023-04-12	CUST023	Female	25	Clothing	4	25	100

--4. Aggregate Functions

--Q10. Count total transactions

SELECT COUNT(*) AS Total_Transactions

FROM RETAIL.SALES.DATASET;

```
56
57 --4. Aggregate Functions
58 --Q10. Count total transactions
59 SELECT COUNT(*) AS Total_Transactions
60 FROM RETAIL.SALES.DATASET;
61
```

results (just now)

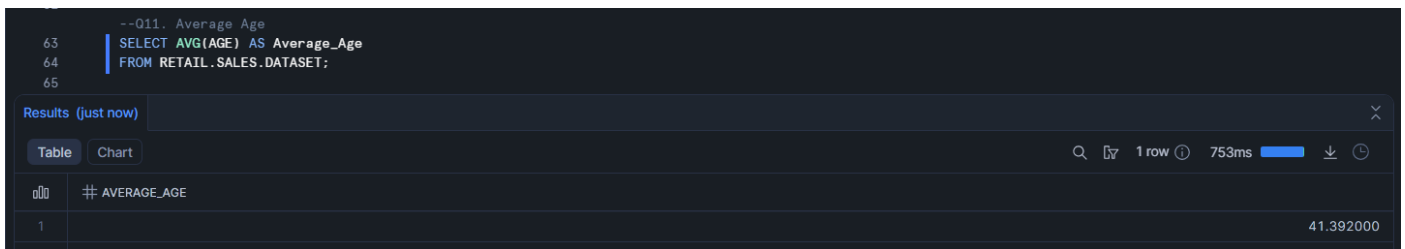
Table Chart

1 row 24ms

	# TOTAL_TRANSACTIONS
1	1000

Q11. Average Age

```
SELECT AVG(AGE) AS Average_Age  
FROM RETAIL.SALES.DATASET;
```

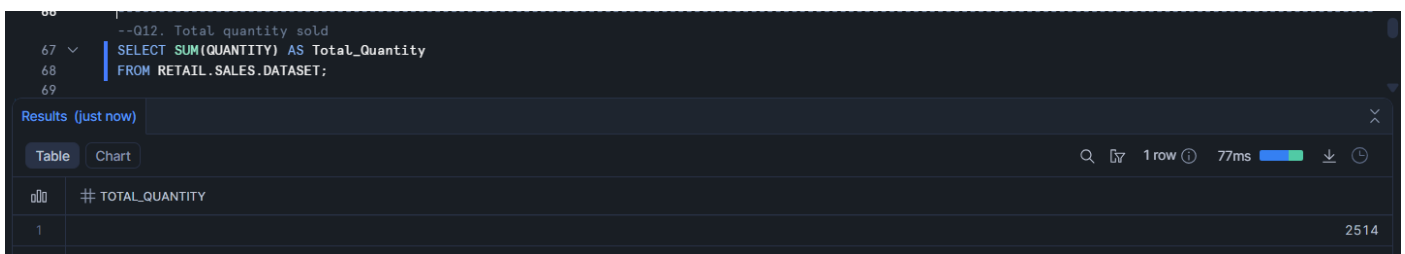


```
--Q11. Average Age  
SELECT AVG(AGE) AS Average_Age  
FROM RETAIL.SALES.DATASET;
```

Results (just now)	
Table	Chart
1	41.392000

Q12. Total quantity sold

```
SELECT SUM(QUANTITY) AS Total_Quantity  
FROM RETAIL.SALES.DATASET;
```

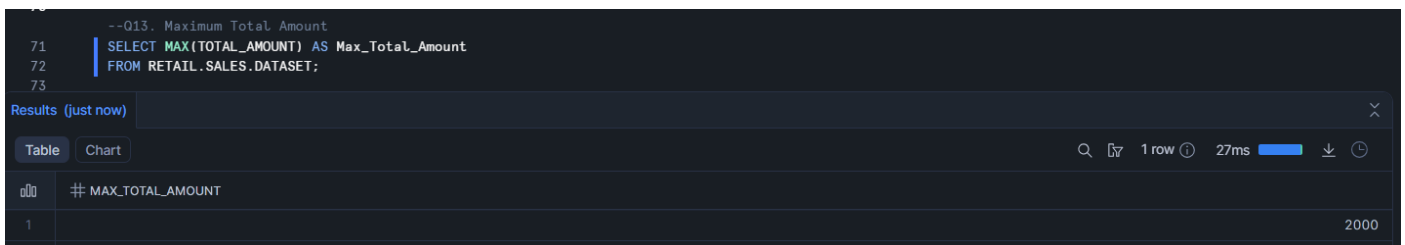


```
--Q12. Total quantity sold  
SELECT SUM(QUANTITY) AS Total_Quantity  
FROM RETAIL.SALES.DATASET;
```

Results (just now)	
Table	Chart
1	2514

Q13. Maximum Total Amount

```
SELECT MAX(TOTAL_AMOUNT) AS Max_Total_Amount  
FROM RETAIL.SALES.DATASET;
```

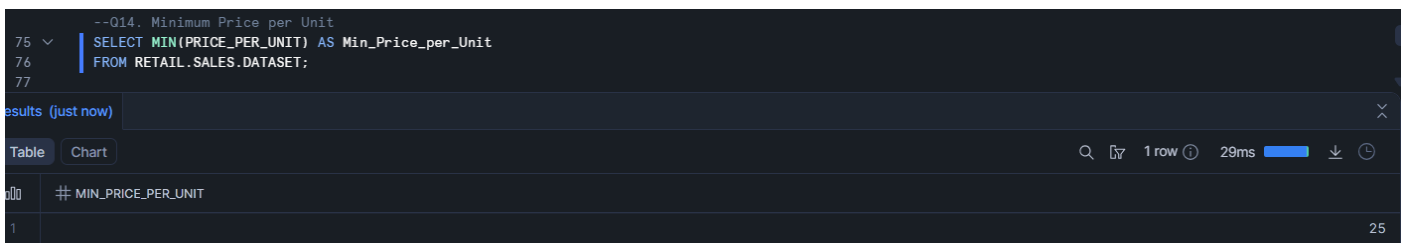


```
--Q13. Maximum Total Amount  
SELECT MAX(TOTAL_AMOUNT) AS Max_Total_Amount  
FROM RETAIL.SALES.DATASET;
```

Results (just now)	
Table	Chart
1	2000

Q14. Minimum Price per Unit

```
SELECT MIN(PRICE_PER_UNIT) AS Min_Price_per_Unit  
FROM RETAIL.SALES.DATASET;
```



```
--Q14. Minimum Price per Unit  
SELECT MIN(PRICE_PER_UNIT) AS Min_Price_per_Unit  
FROM RETAIL.SALES.DATASET;
```

Results (just now)	
Table	Chart
1	25

5. GROUP BY Statement

--Q15. Number of transactions per product category

```
SELECT PRODUCT_CATEGORY,  
       COUNT(*) AS Transaction_Count
```

```
FROM RETAIL.SALES.DATASET

GROUP BY PRODUCT_CATEGORY;
```

```
78
79 -- 5. GROUP BY Statement
80 --Q15. Number of transactions per product category
81 SELECT PRODUCT_CATEGORY,
82        COUNT(*) AS Transaction_Count
83 FROM RETAIL.SALES.DATASET
84 GROUP BY PRODUCT_CATEGORY;
```

Results (1 minute ago)

Table Chart 3 rows 88ms

	PRODUCT_CATEGORY	TRANSACTION_COUNT
1	Clothing	351
2	Beauty	307
3	Electronics	342

Q16. Total revenue per gender

```
SELECT GENDER,
       SUM(TOTAL_AMOUNT) AS Total_Revenue
FROM RETAIL.SALES.DATASET
GROUP BY GENDER;
```

```
85
86 --Q16. Total revenue per gender
87 SELECT GENDER,
88        SUM(TOTAL_AMOUNT) AS Total_Revenue
89 FROM RETAIL.SALES.DATASET
90 GROUP BY GENDER;
```

Results (just now)

Table Chart 2 rows 92ms

	GENDER	TOTAL_REVENUE
1	Male	223160
2	Female	232840

Q17. Average price per unit per product category

```
SELECT PRODUCT_CATEGORY,
       AVG(PRICE_PER_UNIT) AS Average_Price
FROM RETAIL.SALES.DATASET
GROUP BY PRODUCT_CATEGORY;
```

```
91
92 --Q17. Average price per unit per product category
93 SELECT PRODUCT_CATEGORY,
94        AVG(PRICE_PER_UNIT) AS Average_Price
95 FROM RETAIL.SALES.DATASET
96 GROUP BY PRODUCT_CATEGORY;
```

Results (just now)

Table Chart 3 rows 82ms

	PRODUCT_CATEGORY	AVERAGE_PRICE
1	Beauty	184.055375
2	Clothing	174.287749
3	Electronics	181.900585

6. HAVING Clause

```
--Q18. Total revenue per product category > 10,000

SELECT PRODUCT_CATEGORY,
```

SUM(TOTAL_AMOUNT) AS Total_Revenue

FROM RETAIL.SALES.DATASET

GROUP BY PRODUCT_CATEGORY

HAVING SUM(TOTAL_AMOUNT) > 10000;

```
105 --6. HAVING Clause
106 --Q18. Total revenue per product category > 10,000
107
108 SELECT PRODUCT_CATEGORY,
109        SUM(TOTAL_AMOUNT) AS Total_Revenue
110 FROM RETAIL.SALES.DATASET
111 GROUP BY PRODUCT_CATEGORY
112 HAVING SUM(TOTAL_AMOUNT) > 10000;
113
```

Results (just now)

	PRODUCT_CATEGORY	TOTAL_REVENUE
1	Beauty	143515
2	Clothing	155580
3	Electronics	156905

Q19. Average quantity > 2

SELECT PRODUCT_CATEGORY,

AVG(QUANTITY) AS Average_Quantity

FROM RETAIL.SALES.DATASET

GROUP BY PRODUCT_CATEGORY

HAVING AVG(QUANTITY) > 2;

```
105 --Q19. Average quantity > 2
106
107 SELECT PRODUCT_CATEGORY,
108        AVG(QUANTITY) AS Average_Quantity
109 FROM RETAIL.SALES.DATASET
110 GROUP BY PRODUCT_CATEGORY
111 HAVING AVG(QUANTITY) > 2;
112
```

Results (just now)

	PRODUCT_CATEGORY	AVERAGE_QUANTITY
1	Beauty	2.511401
2	Clothing	2.547009
3	Electronics	2.482456

7. CASE Statement

--Q20. Spending Level (High/Low)

```
SELECT TRANSACTION_ID,  
  
       TOTAL_AMOUNT,  
  
       CASE  
  
         WHEN TOTAL_AMOUNT > 1000 THEN 'High'  
  
         ELSE 'Low'  
  
       END AS Spending_Level  
  
FROM RETAIL.SALES.DATASET;
```

112
113
114
115
116
117
118
119
120
121

-- 7. CASE Statement
--Q20. Spending Level (High/Low)
SELECT TRANSACTION_ID,
 TOTAL_AMOUNT,
 CASE
 WHEN TOTAL_AMOUNT > 1000 THEN 'High'
 ELSE 'Low'
 END AS Spending_Level
FROM RETAIL.SALES.DATASET;

results (just now)

TableChart

1,000 rows71ms

#	TRANSACTION_ID	TOTAL_AMOUNT	SPENDING_LEVEL
1	1	150	Low
2	2	1000	Low
3	3	30	Low
4	4	500	Low
5	5	100	Low
6	6	30	Low
7	7	50	Low
8	8	100	Low

Q21. Age Group Classification

```
SELECT CUSTOMER_ID,  
  
    AGE,  
  
    CASE  
  
        WHEN AGE < 30 THEN 'Youth'  
  
        WHEN AGE BETWEEN 30 AND 59 THEN 'Adult'  
  
        ELSE 'Senior'  
  
    END AS Age_Group  
  
FROM RETAIL.SALES.DATASET;
```

--Q21. Age Group Classification

123 SELECT CUSTOMER_ID,
124 AGE,
125 CASE
126 WHEN AGE < 30 THEN 'Youth'
127 WHEN AGE BETWEEN 30 AND 59 THEN 'Adult'
128 ELSE 'Senior'
129 END AS Age_Group
130 FROM RETAIL.SALES.DATASET;
131

Results (just now)

TableChart

1,000 rows66ms

	CUSTOMER_ID	AGE	AGE_GROUP
1	CUST001	34	Adult
2	CUST002	26	Youth
3	CUST003	50	Adult
4	CUST004	37	Adult
5	CUST005	30	Adult
6	CUST006	45	Adult