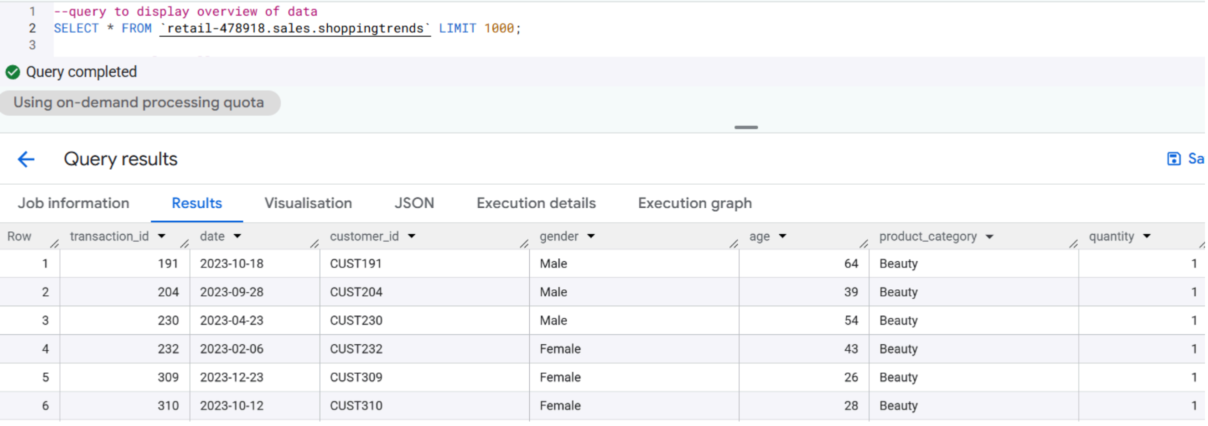
Mikateko Mpapele

Big Query Practical

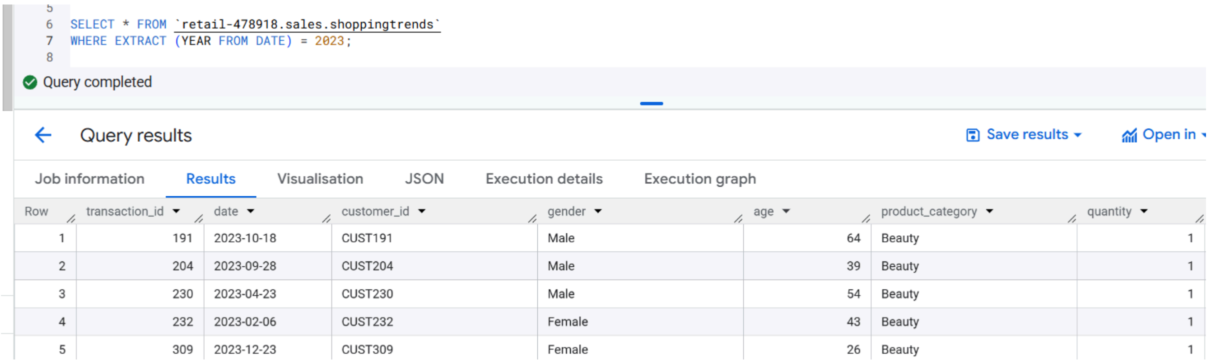
1. Query to display an overview of data



```sql
--query to display overview of data
SELECT * FROM `retail-478918.sales.shoppingtrends` LIMIT 1000;
```

Row	transaction_id	date	customer_id	gender	age	product_category	quantity
1	191	2023-10-18	CUST191	Male	64	Beauty	1
2	204	2023-09-28	CUST204	Male	39	Beauty	1
3	230	2023-04-23	CUST230	Male	54	Beauty	1
4	232	2023-02-06	CUST232	Female	43	Beauty	1
5	309	2023-12-23	CUST309	Female	26	Beauty	1
6	310	2023-10-12	CUST310	Female	28	Beauty	1

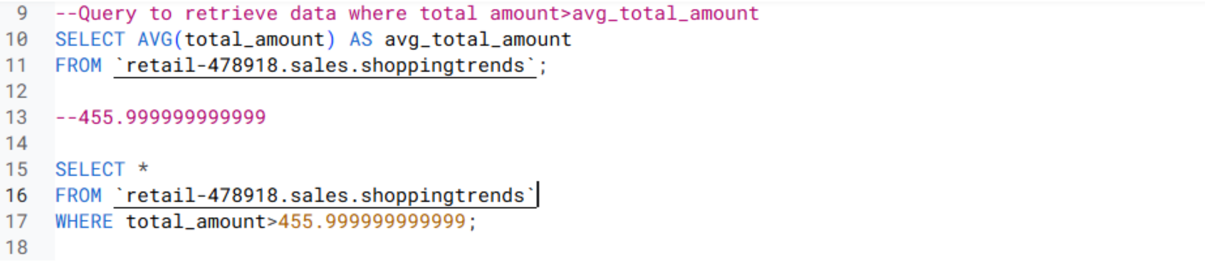
2. Query to extract data on transactions where year=2023



```sql
SELECT * FROM `retail-478918.sales.shoppingtrends`
WHERE EXTRACT (YEAR FROM DATE) = 2023;
```

Row	transaction_id	date	customer_id	gender	age	product_category	quantity
1	191	2023-10-18	CUST191	Male	64	Beauty	1
2	204	2023-09-28	CUST204	Male	39	Beauty	1
3	230	2023-04-23	CUST230	Male	54	Beauty	1
4	232	2023-02-06	CUST232	Female	43	Beauty	1
5	309	2023-12-23	CUST309	Female	26	Beauty	1

3. Query to retrieve data where total amount greater the average total amount.

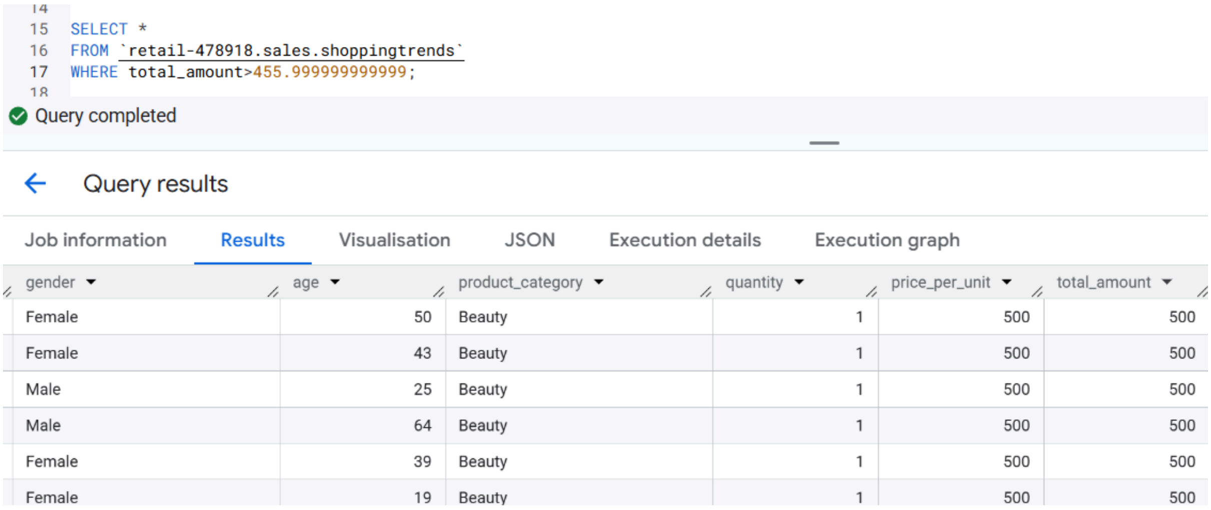


```sql
--Query to retrieve data where total amount>avg_total_amount
SELECT AVG(total_amount) AS avg_total_amount
FROM `retail-478918.sales.shoppingtrends`;

--455.999999999999

SELECT *
FROM `retail-478918.sales.shoppingtrends`
WHERE total_amount>455.999999999999;

```



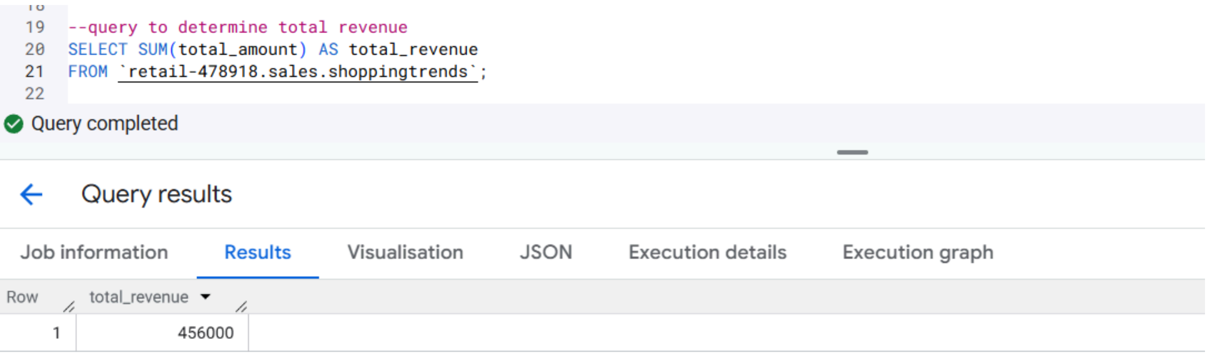
```
SELECT *
FROM `retail-478918.sales.shoppingtrends`
WHERE total_amount>455.999999999999;
```

Query completed

Query results

gender	age	product_category	quantity	price_per_unit	total_amount
Female	50	Beauty	1	500	500
Female	43	Beauty	1	500	500
Male	25	Beauty	1	500	500
Male	64	Beauty	1	500	500
Female	39	Beauty	1	500	500
Female	19	Beauty	1	500	500

4. Query to determine total revenue



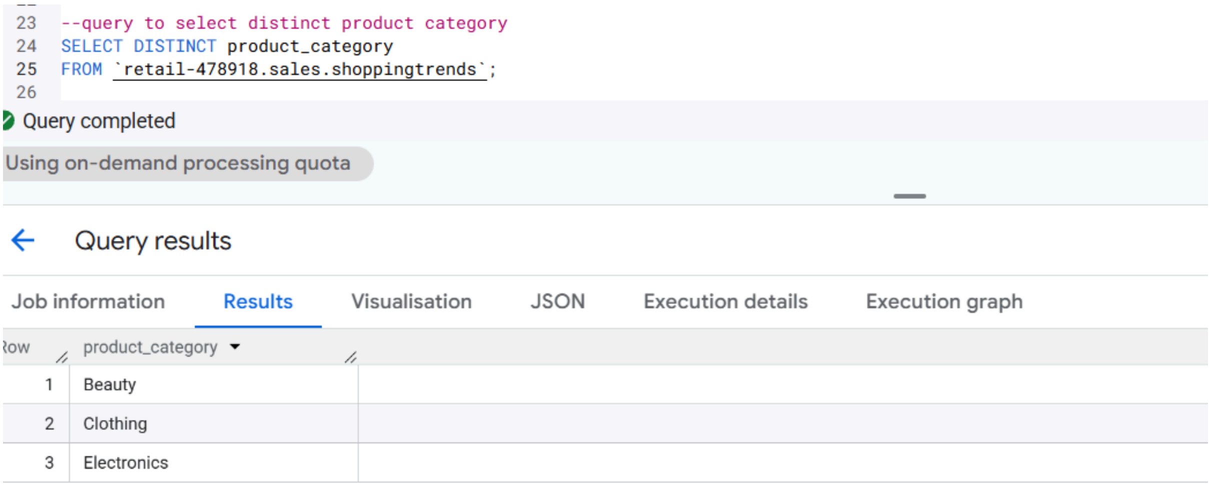
```
--query to determine total revenue
SELECT SUM(total_amount) AS total_revenue
FROM `retail-478918.sales.shoppingtrends`;
```

Query completed

Query results

Row	total_revenue
1	456000

5. Query to select distinct product categories



```
--query to select distinct product category
SELECT DISTINCT product_category
FROM `retail-478918.sales.shoppingtrends`;
```

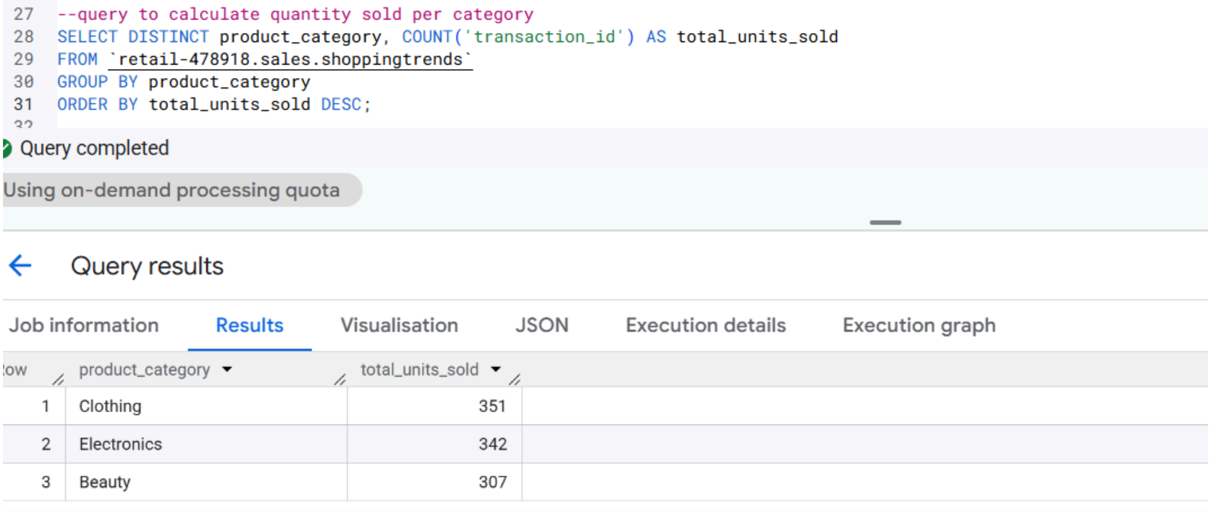
Query completed

Using on-demand processing quota

Query results

Row	product_category
1	Beauty
2	Clothing
3	Electronics

7. Query to calculate quantity sold per category



```
--query to calculate quantity sold per category
SELECT DISTINCT product_category, COUNT('transaction_id') AS total_units_sold
FROM `retail-478918.sales.shoppingtrends`
GROUP BY product_category
ORDER BY total_units_sold DESC;
```

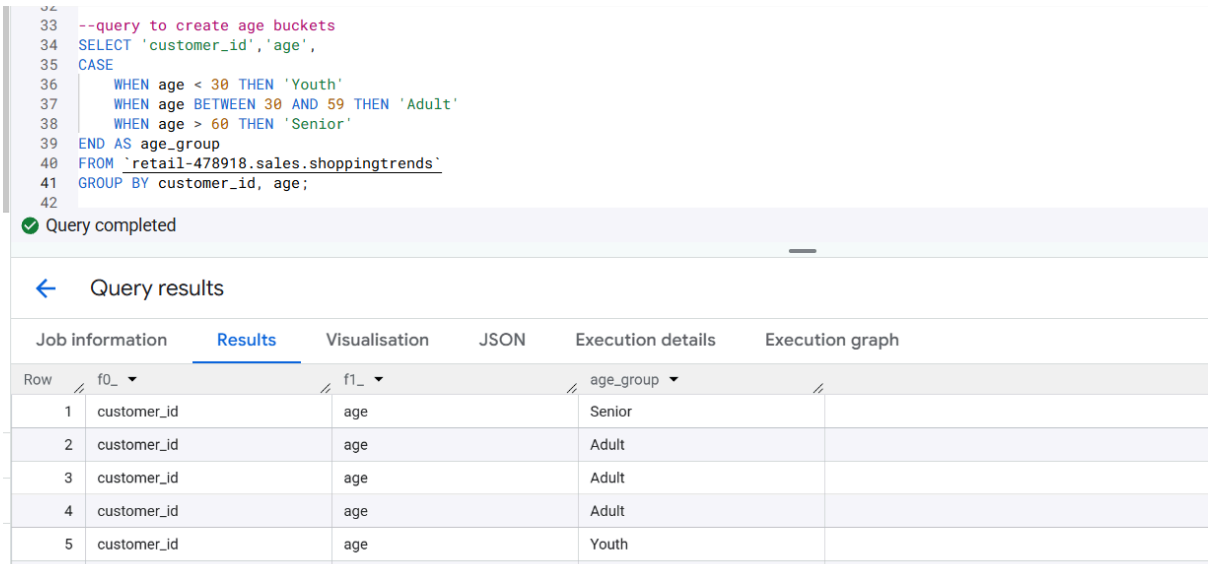
Query completed

Using on-demand processing quota

Query results

ow	product_category	total_units_sold
1	Clothing	351
2	Electronics	342
3	Beauty	307

8. Query to create age groups



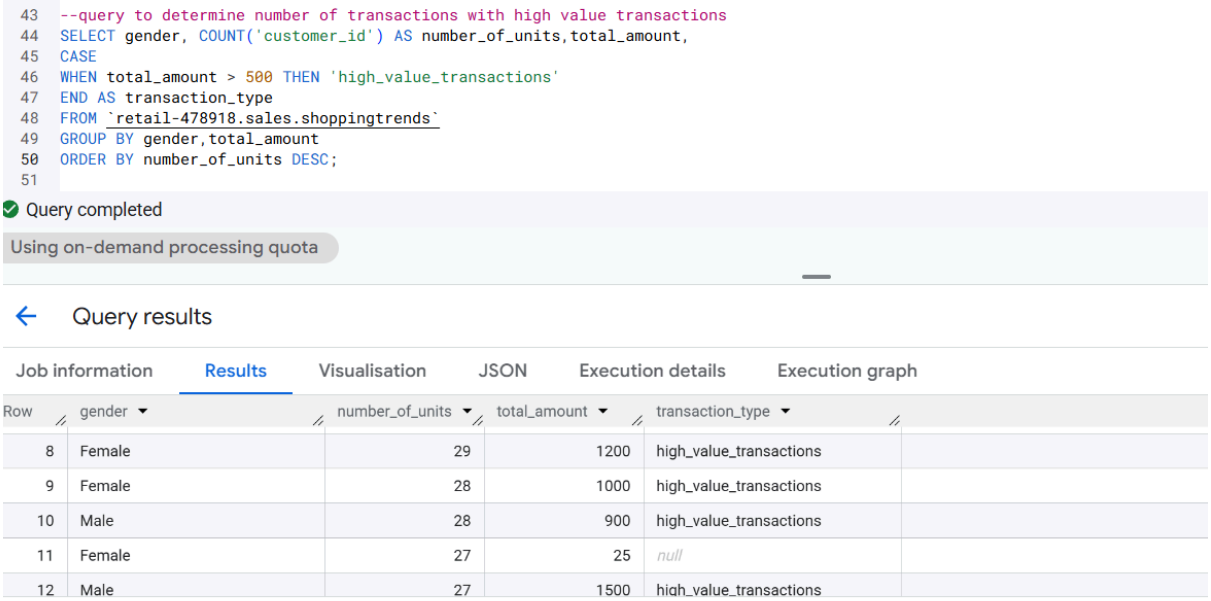
```
--query to create age buckets
SELECT 'customer_id','age',
CASE
    WHEN age < 30 THEN 'Youth'
    WHEN age BETWEEN 30 AND 59 THEN 'Adult'
    WHEN age > 60 THEN 'Senior'
END AS age_group
FROM `retail-478918.sales.shoppingtrends`
GROUP BY customer_id, age;
```

Query completed

Query results

Row	f0_	f1_	age_group
1	customer_id	age	Senior
2	customer_id	age	Adult
3	customer_id	age	Adult
4	customer_id	age	Adult
5	customer_id	age	Youth

9. Query to determine number of transactions with high value transactions



```
--query to determine number of transactions with high value transactions
SELECT gender, COUNT('customer_id') AS number_of_units,total_amount,
CASE
WHEN total_amount > 500 THEN 'high_value_transactions'
END AS transaction_type
FROM `retail-478918.sales.shoppingtrends`
GROUP BY gender,total_amount
ORDER BY number_of_units DESC;
```

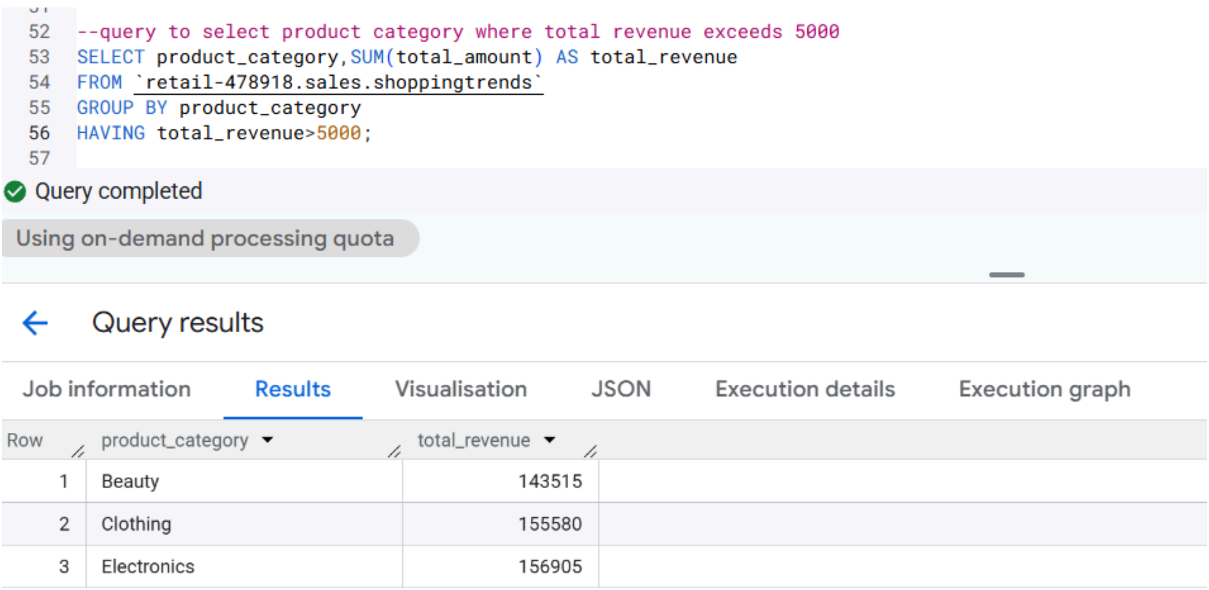
Query completed

Using on-demand processing quota

Query results

Row	gender	number_of_units	total_amount	transaction_type
8	Female	29	1200	high_value_transactions
9	Female	28	1000	high_value_transactions
10	Male	28	900	high_value_transactions
11	Female	27	25	null
12	Male	27	1500	high_value_transactions

10. Query to select product category where total revenue exceeds 1000



```
--query to select product category where total revenue exceeds 5000
SELECT product_category,SUM(total_amount) AS total_revenue
FROM `retail-478918.sales.shoppingtrends`
GROUP BY product_category
HAVING total_revenue>5000;
```

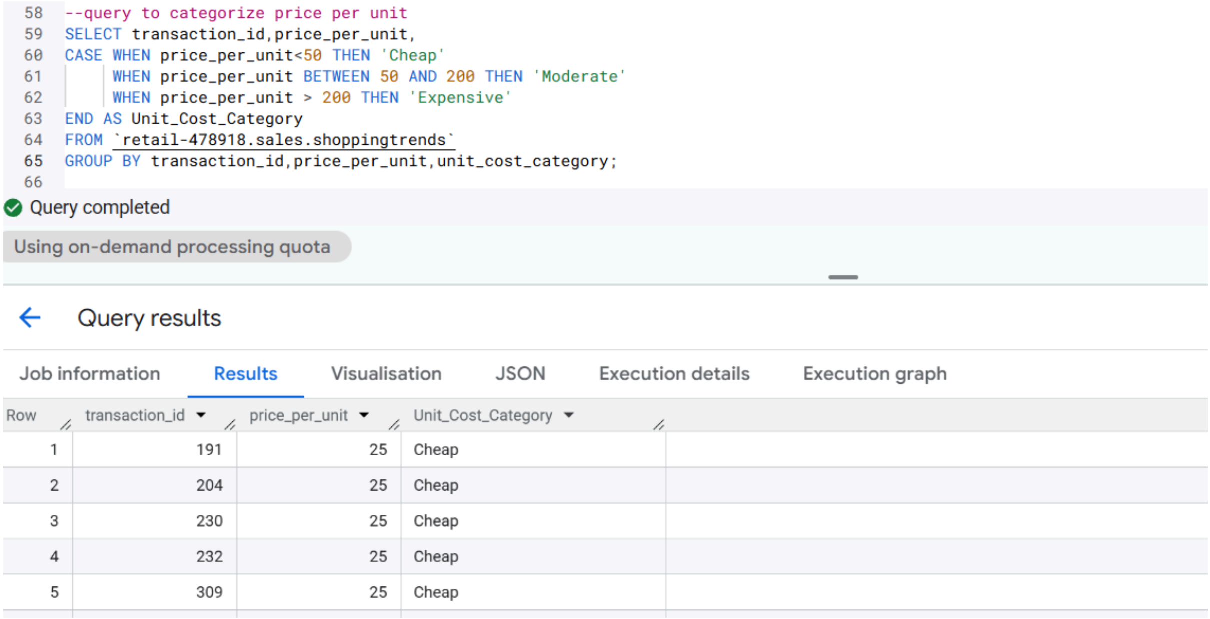
Query completed

Using on-demand processing quota

Query results

Row	product_category	total_revenue
1	Beauty	143515
2	Clothing	155580
3	Electronics	156905

11. Query to categorize price per unit



```
--query to categorize price per unit
SELECT transaction_id,price_per_unit,
CASE WHEN price_per_unit<50 THEN 'Cheap'
     WHEN price_per_unit BETWEEN 50 AND 200 THEN 'Moderate'
     WHEN price_per_unit > 200 THEN 'Expensive'
END AS Unit_Cost_Category
FROM `retail-478918.sales.shoppingtrends`
GROUP BY transaction_id,price_per_unit,unit_cost_category;
```

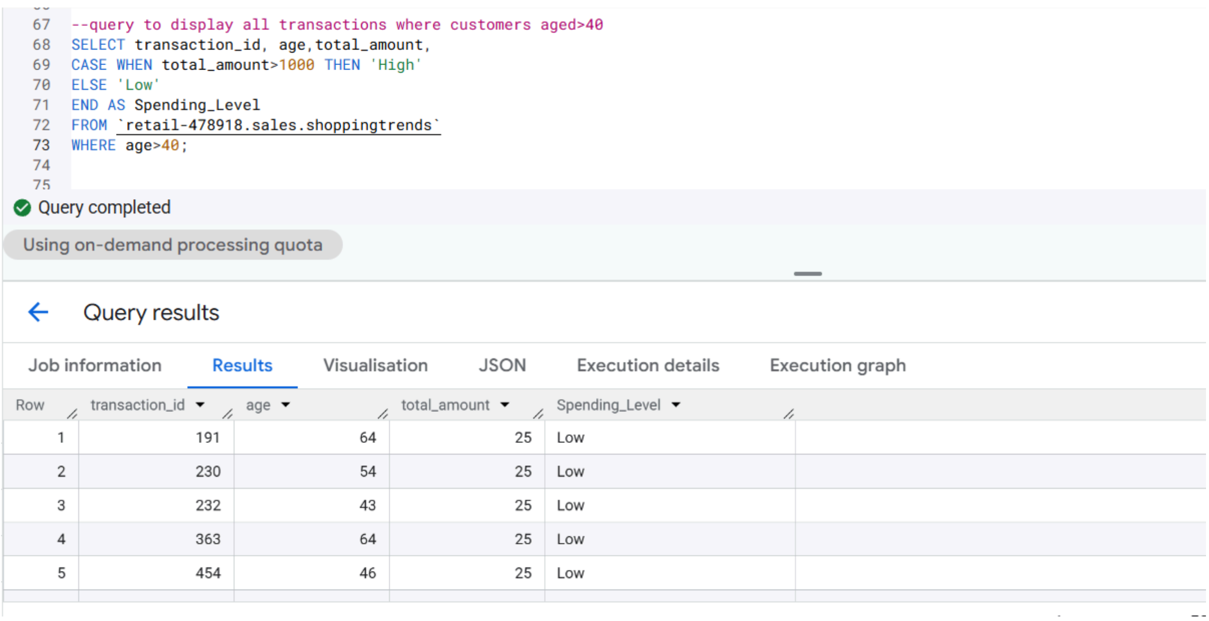
Query completed

Using on-demand processing quota

Query results

Row	transaction_id	price_per_unit	Unit_Cost_Category
1	191	25	Cheap
2	204	25	Cheap
3	230	25	Cheap
4	232	25	Cheap
5	309	25	Cheap

12. Query to display all transactions where customers aged>40



```
--query to display all transactions where customers aged>40
SELECT transaction_id, age,total_amount,
CASE WHEN total_amount>1000 THEN 'High'
ELSE 'Low'
END AS Spending_Level
FROM `retail-478918.sales.shoppingtrends`
WHERE age>40;
```

Query completed

Using on-demand processing quota

Row	transaction_id	age	total_amount	Spending_Level
1	191	64	25	Low
2	230	54	25	Low
3	232	43	25	Low
4	363	64	25	Low
5	454	46	25	Low