ADEYEMI ADESHOLA KEHINDE

Retail Sales Analysis – SQL Project

Step 1: Importing the Dataset

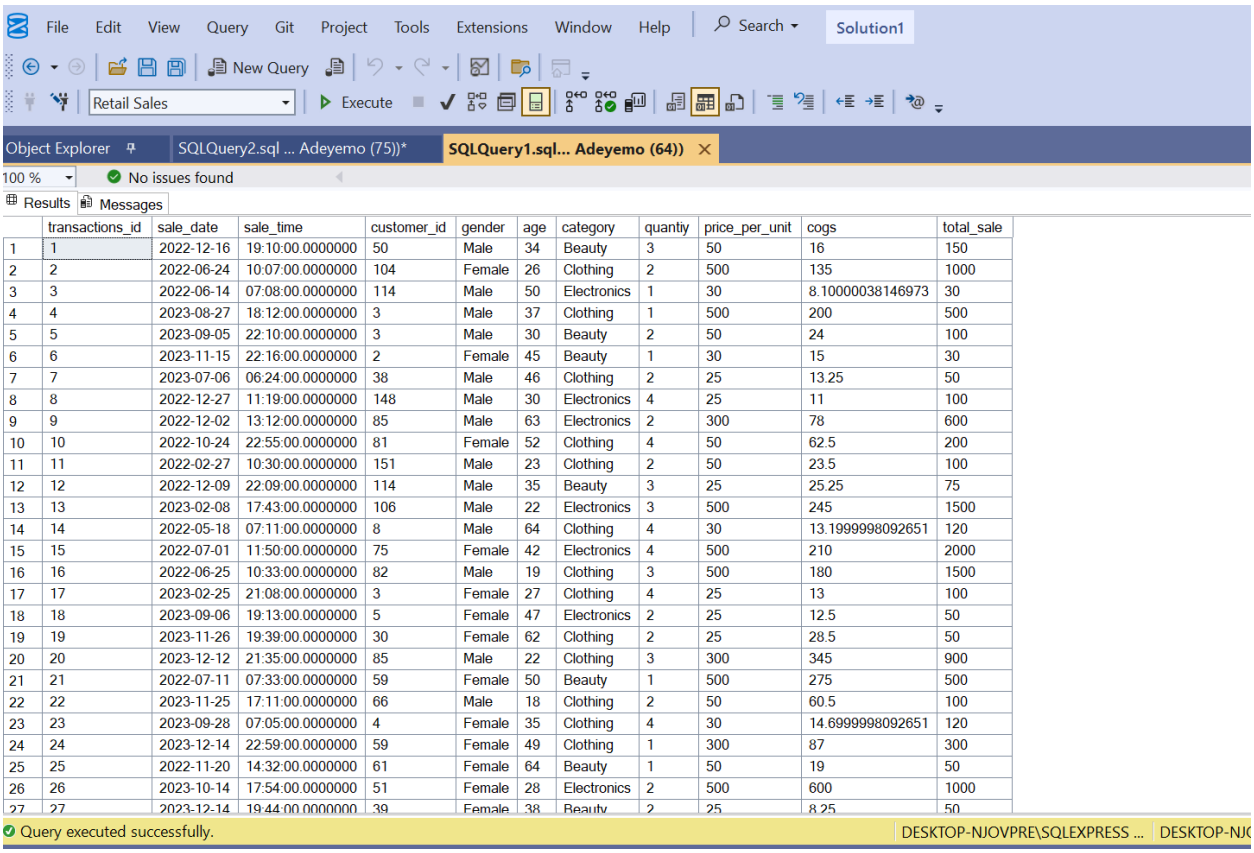
I imported the dataset SQL - Retail Sales Analysis.csv into SQL Server using the **Import Flat File** wizard in SSMS:

1. Connected to my database in SQL Server Management Studio (SSMS).
2. Right-clicked the database → **Tasks** → **Import Flat File**.
3. Selected the CSV file and verified the column data types:
    - Date columns → DATE or DATETIME
    - Numeric columns → INT or DECIMAL
4. Saved the table as RetailSales.

Step 2: Checking for NULL and Cleaning the Data

```
select * from RetailSales where age is null
or quantiy is null or price_per_unit is null or cogs is null
or total_sale is null or transactions_id is null or sale_date is null or sale_time is null or customer_id is null
or gender is null or category is null

delete from RetailSales
where transactions_id in
(150, 432, 679, 746, 797, 845, 921, 1150, 1225, 1367, 1391, 1432, 1845);
```

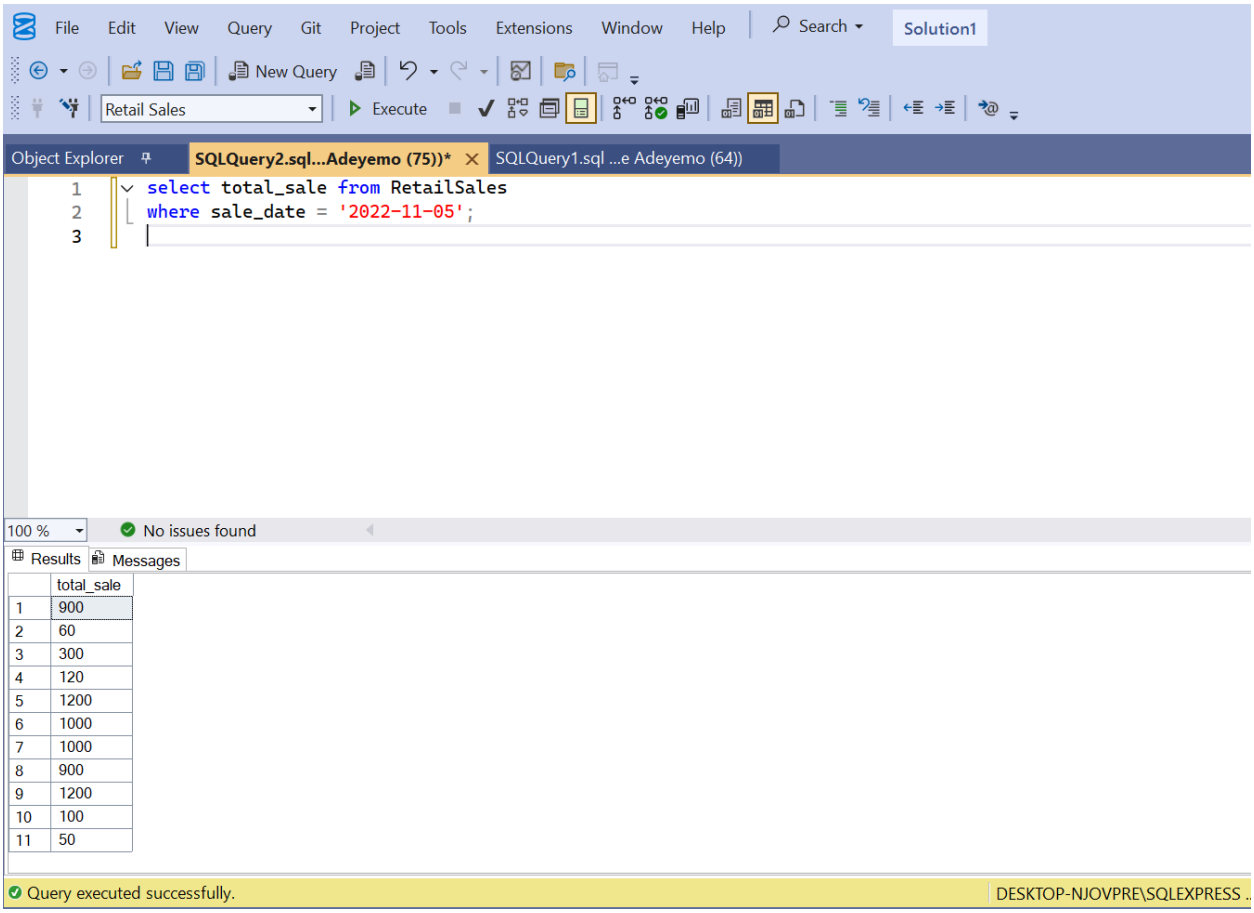


	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	1	2022-12-16	19:10:00.0000000	50	Male	34	Beauty	3	50	16	150
2	2	2022-06-24	10:07:00.0000000	104	Female	26	Clothing	2	500	135	1000
3	3	2022-06-14	07:08:00.0000000	114	Male	50	Electronics	1	30	8.10000038146973	30
4	4	2023-08-27	18:12:00.0000000	3	Male	37	Clothing	1	500	200	500
5	5	2023-09-05	22:10:00.0000000	3	Male	30	Beauty	2	50	24	100
6	6	2023-11-15	22:16:00.0000000	2	Female	45	Beauty	1	30	15	30
7	7	2023-07-06	06:24:00.0000000	38	Male	46	Clothing	2	25	13.25	50
8	8	2022-12-27	11:19:00.0000000	148	Male	30	Electronics	4	25	11	100
9	9	2022-12-02	13:12:00.0000000	85	Male	63	Electronics	2	300	78	600
10	10	2022-10-24	22:55:00.0000000	81	Female	52	Clothing	4	50	62.5	200
11	11	2022-02-27	10:30:00.0000000	151	Male	23	Clothing	2	50	23.5	100
12	12	2022-12-09	22:09:00.0000000	114	Male	35	Beauty	3	25	25.25	75
13	13	2023-02-08	17:43:00.0000000	106	Male	22	Electronics	3	500	245	1500
14	14	2022-05-18	07:11:00.0000000	8	Male	64	Clothing	4	30	13.1999998092651	120
15	15	2022-07-01	11:50:00.0000000	75	Female	42	Electronics	4	500	210	2000
16	16	2022-06-25	10:33:00.0000000	82	Male	19	Clothing	3	500	180	1500
17	17	2023-02-25	21:08:00.0000000	3	Female	27	Clothing	4	25	13	100
18	18	2023-09-06	19:13:00.0000000	5	Female	47	Electronics	2	25	12.5	50
19	19	2023-11-26	19:39:00.0000000	30	Female	62	Clothing	2	25	28.5	50
20	20	2023-12-12	21:35:00.0000000	85	Male	22	Clothing	3	300	345	900
21	21	2022-07-11	07:33:00.0000000	59	Female	50	Beauty	1	500	275	500
22	22	2023-11-25	17:11:00.0000000	66	Male	18	Clothing	2	50	60.5	100
23	23	2023-09-28	07:05:00.0000000	4	Female	35	Clothing	4	30	14.6999998092651	120
24	24	2023-12-14	22:59:00.0000000	59	Female	49	Clothing	1	300	87	300
25	25	2022-11-20	14:32:00.0000000	61	Female	64	Beauty	1	50	19	50
26	26	2023-10-14	17:54:00.0000000	51	Female	28	Electronics	2	500	600	1000
27	27	2023-12-14	19:44:00.0000000	39	Female	38	Beauty	2	25	8.25	50

Step 3: SQL Queries

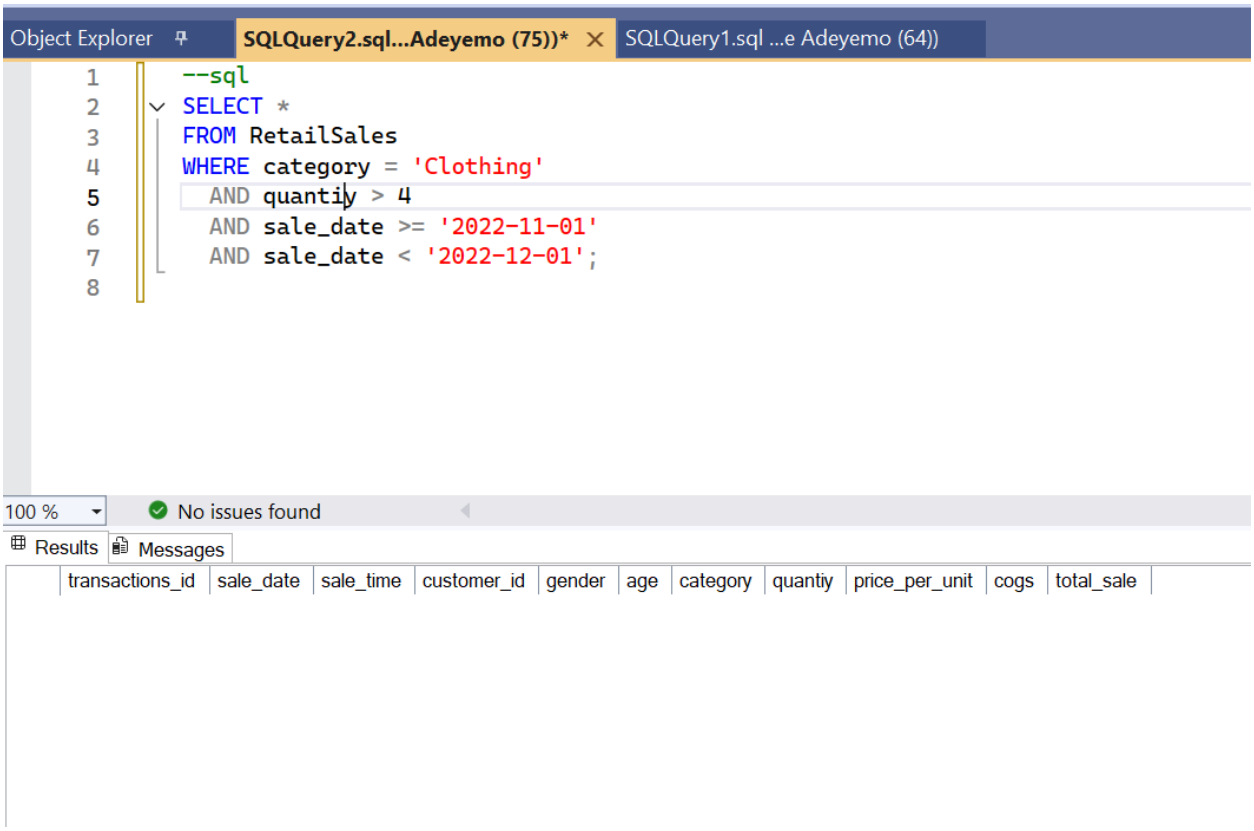
i. Sales made on 2022-11-05

```
select total_sale from RetailSales
where sale_date = '2022-11-05';
```



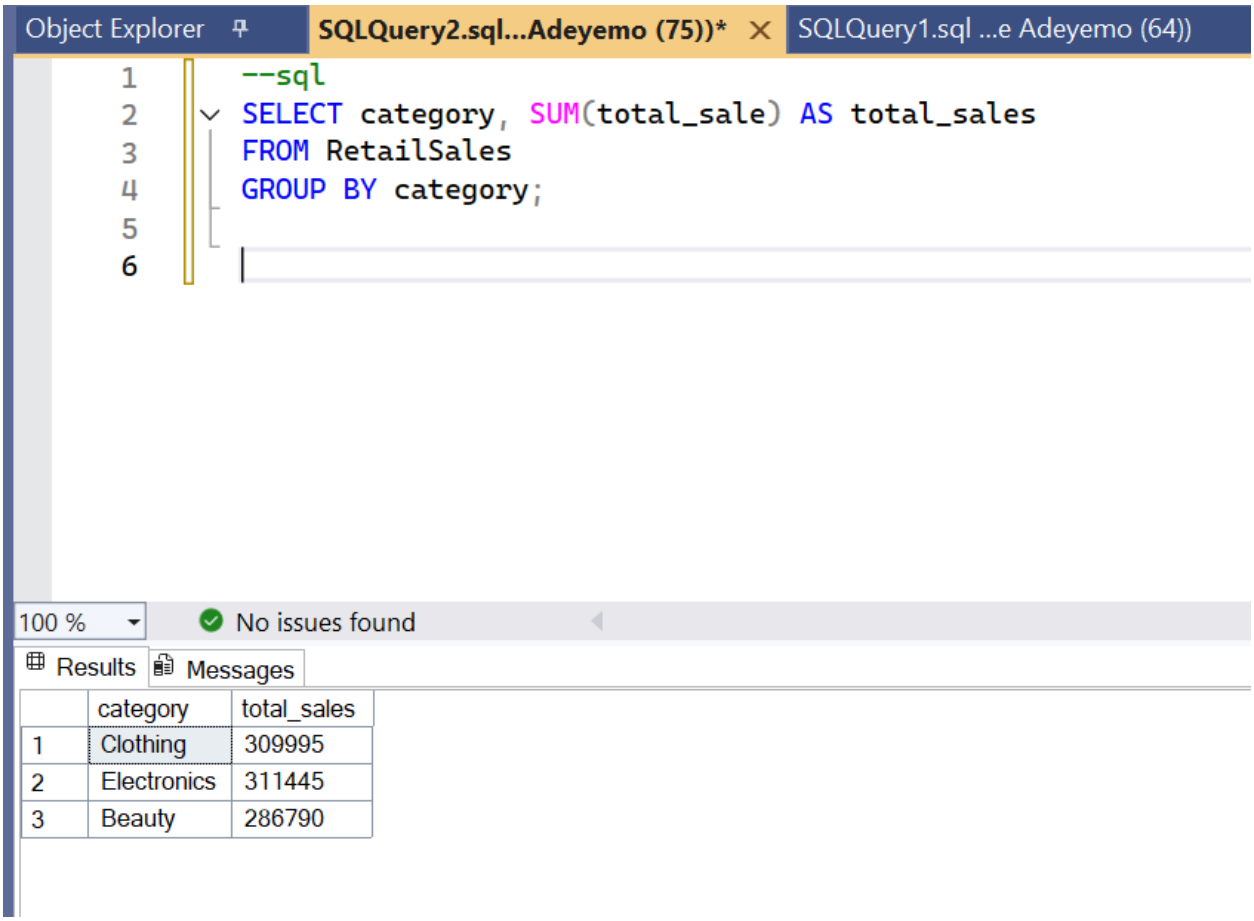
ii. Clothing sales, quantity > 4, Nov-2022

```
--sql
SELECT *
FROM RetailSales
WHERE category = 'Clothing'
  AND quantity > 4
  AND sale_date >= '2022-11-01'
  AND sale_date < '2022-12-01';
```



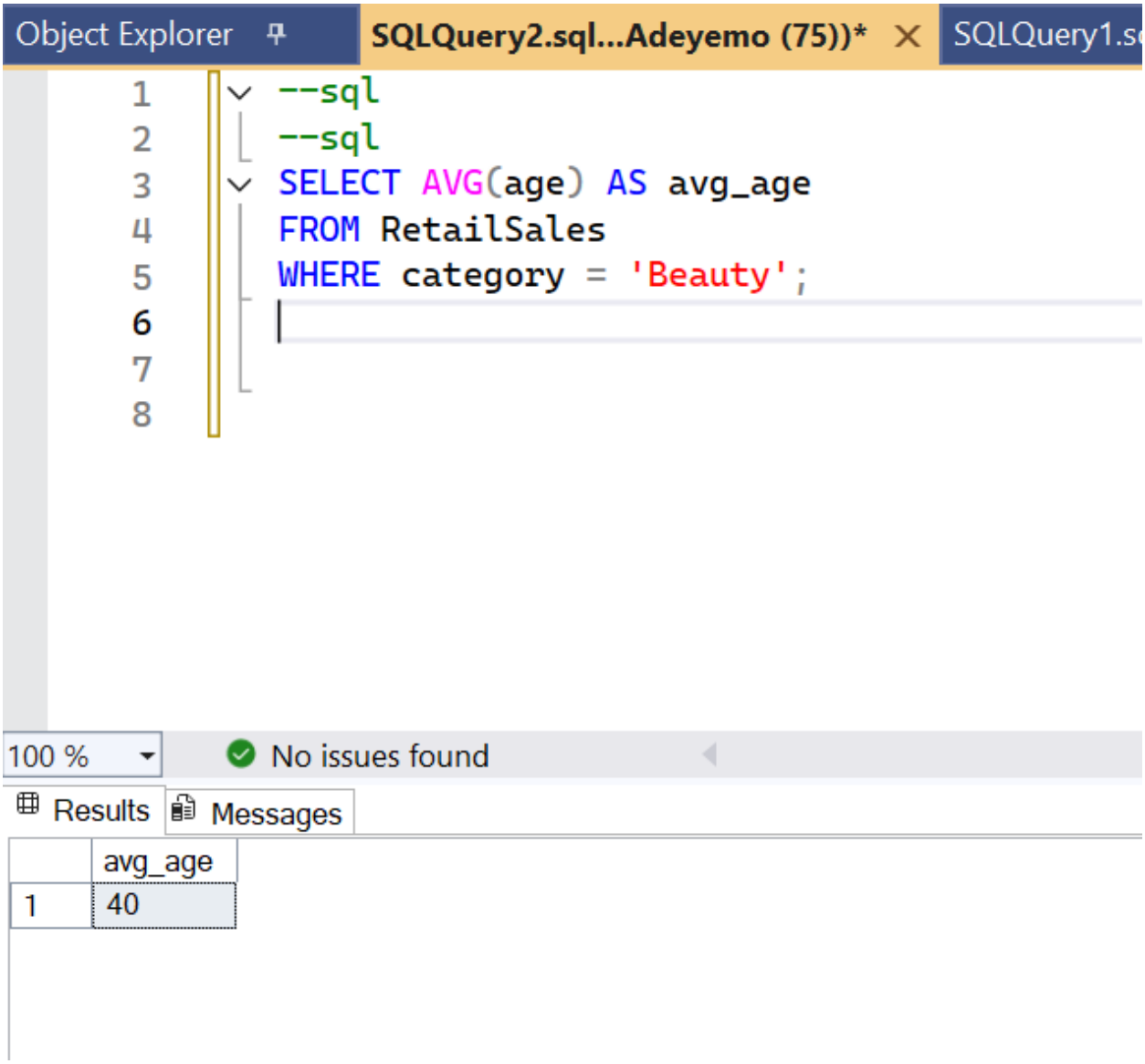
iii. Total sales for each category

--sql
SELECT category, SUM(total_sale) AS total_sales
FROM RetailSales
GROUP BY category;



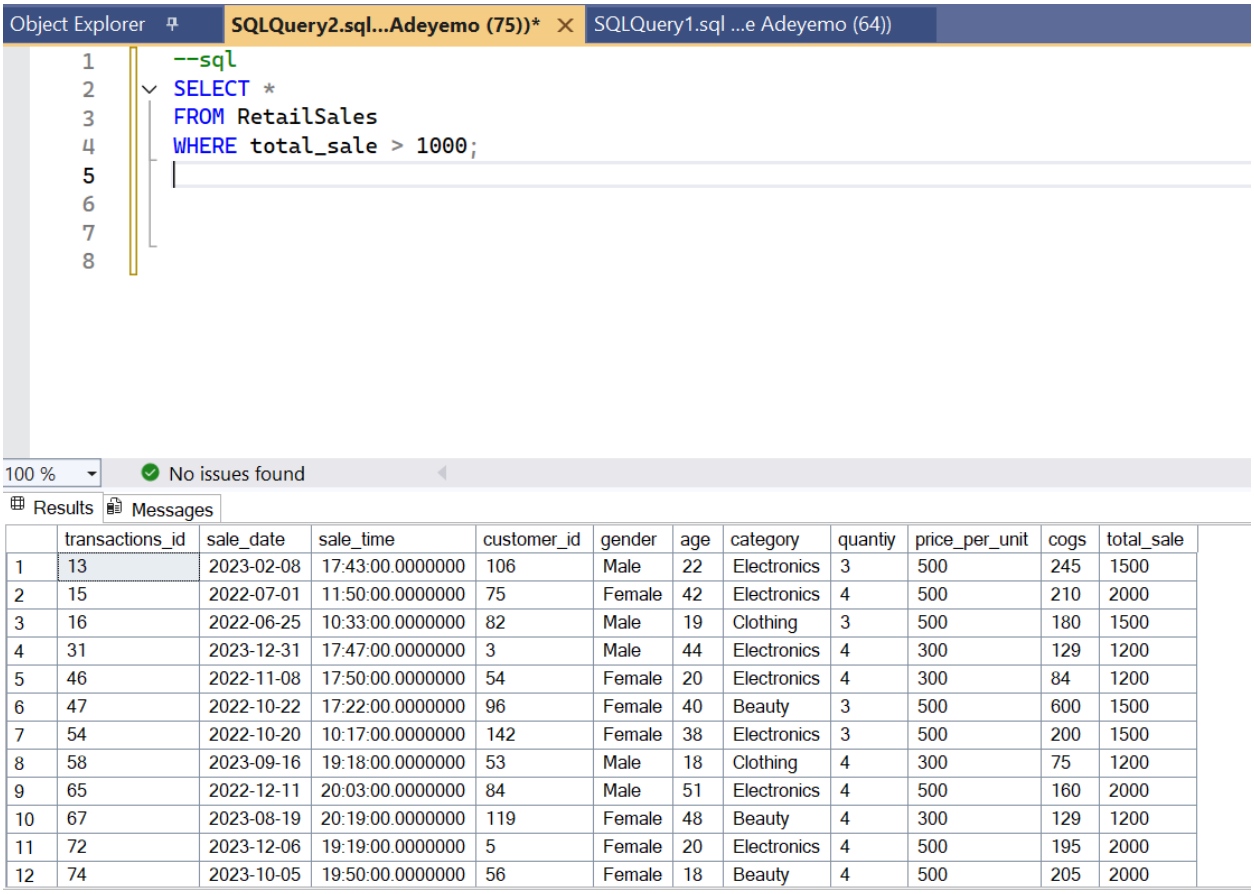
iv. Average age of customers in Beauty category

```
--sql
SELECT AVG(age) AS avg_age
FROM RetailSales
WHERE category = 'Beauty';
```



v. Transactions where total_sale > 1000

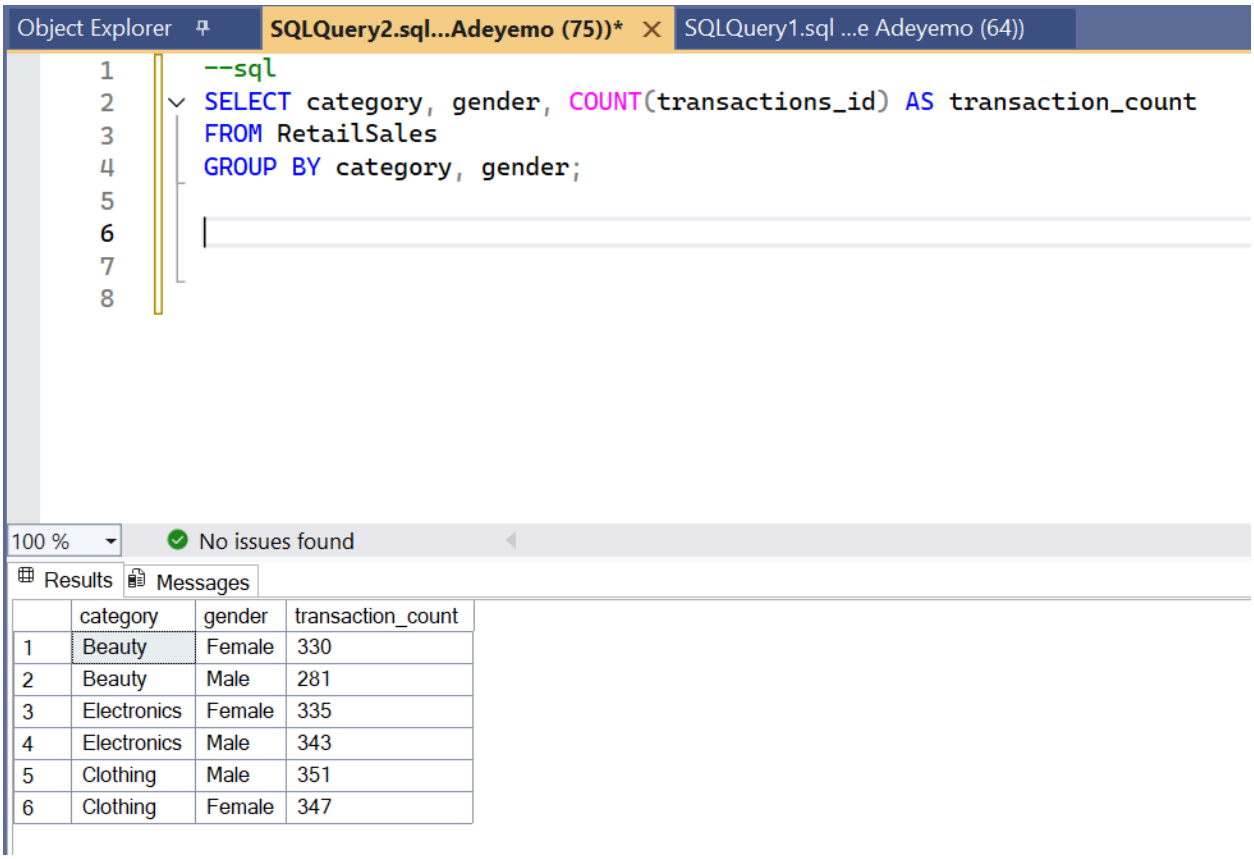
```
--sql
SELECT *
FROM RetailSales
WHERE total_sale > 1000;
```



	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	13	2023-02-08	17:43:00.0000000	106	Male	22	Electronics	3	500	245	1500
2	15	2022-07-01	11:50:00.0000000	75	Female	42	Electronics	4	500	210	2000
3	16	2022-06-25	10:33:00.0000000	82	Male	19	Clothing	3	500	180	1500
4	31	2023-12-31	17:47:00.0000000	3	Male	44	Electronics	4	300	129	1200
5	46	2022-11-08	17:50:00.0000000	54	Female	20	Electronics	4	300	84	1200
6	47	2022-10-22	17:22:00.0000000	96	Female	40	Beauty	3	500	600	1500
7	54	2022-10-20	10:17:00.0000000	142	Female	38	Electronics	3	500	200	1500
8	58	2023-09-16	19:18:00.0000000	53	Male	18	Clothing	4	300	75	1200
9	65	2022-12-11	20:03:00.0000000	84	Male	51	Electronics	4	500	160	2000
10	67	2023-08-19	20:19:00.0000000	119	Female	48	Beauty	4	300	129	1200
11	72	2023-12-06	19:19:00.0000000	5	Female	20	Electronics	4	500	195	2000
12	74	2023-10-05	19:50:00.0000000	56	Female	18	Beauty	4	500	205	2000

vi. Number of transactions by gender in each category

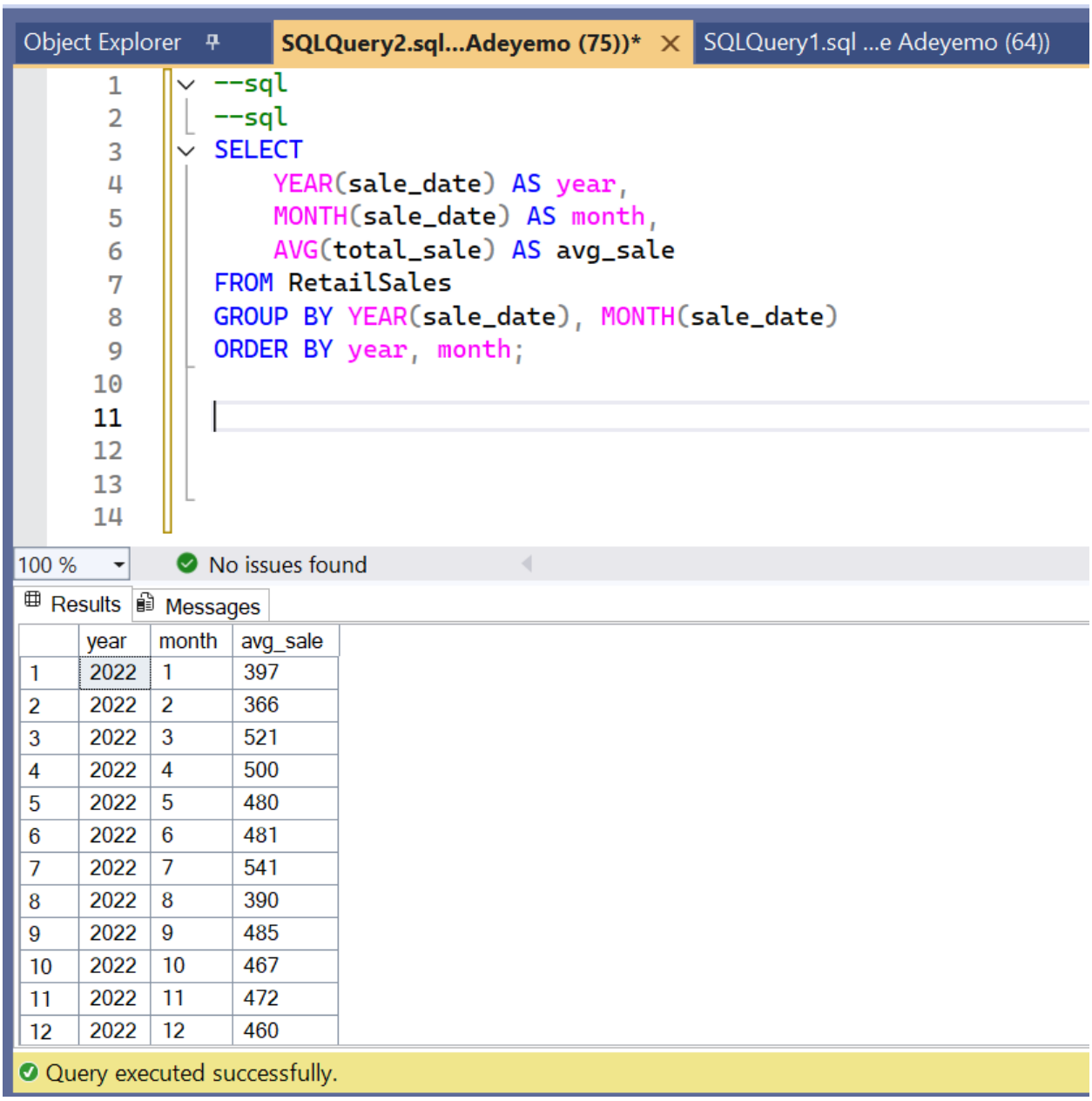
```
--sql
SELECT category, gender, COUNT(transaction_id) AS transaction_count
FROM RetailSales
GROUP BY category, gender;
```



	category	gender	transaction_count
1	Beauty	Female	330
2	Beauty	Male	281
3	Electronics	Female	335
4	Electronics	Male	343
5	Clothing	Male	351
6	Clothing	Female	347

vii. Average sale per month

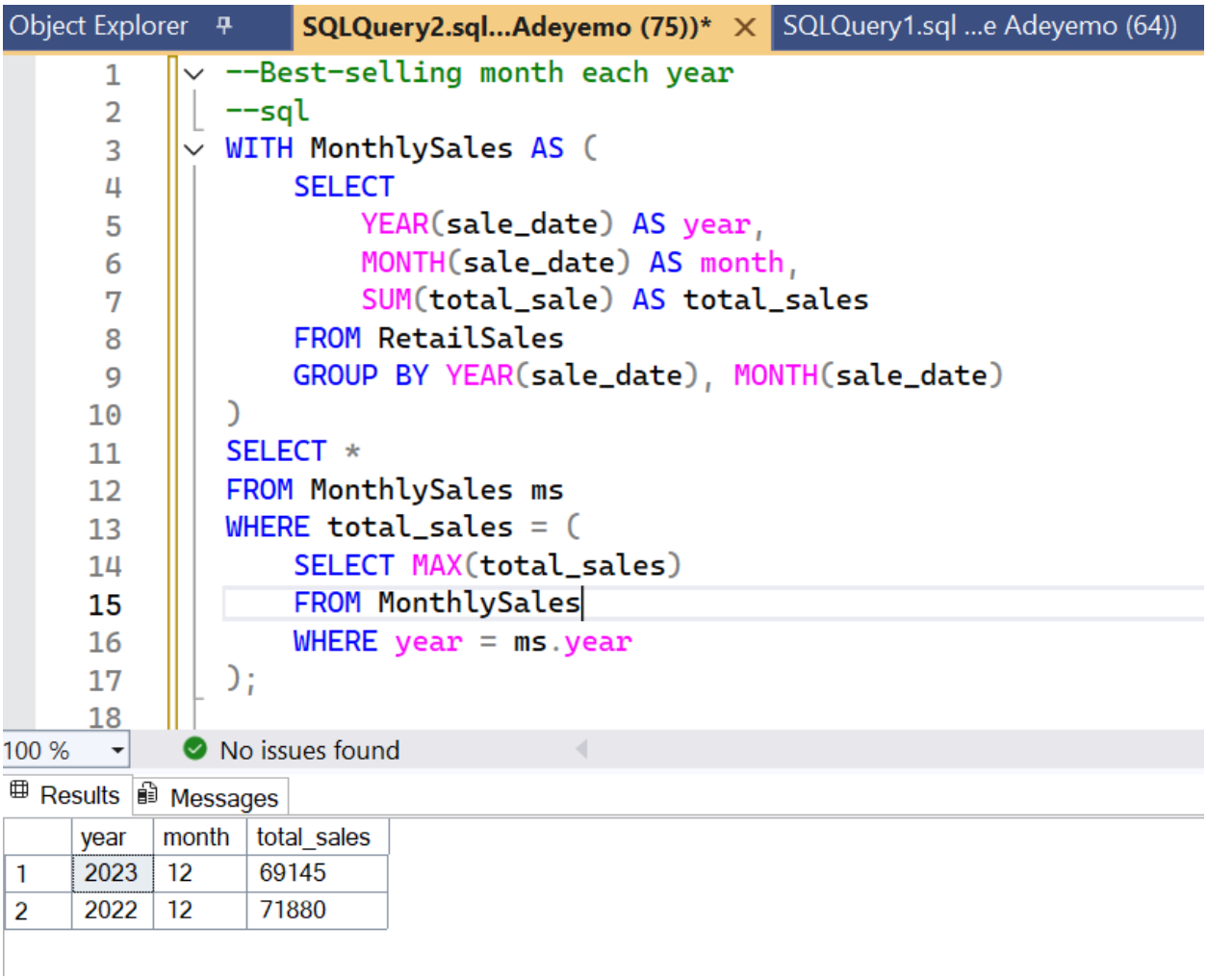
```
--sql
SELECT
   YEAR(sale_date) AS year,
   MONTH(sale_date) AS month,
   AVG(total_sale) AS avg_sale
FROM RetailSales
GROUP BY YEAR(sale_date), MONTH(sale_date)
ORDER BY year, month;
```



--Best-selling month each year

```
--sql
WITH MonthlySales AS (
  SELECT
    YEAR(sale_date) AS year,
    MONTH(sale_date) AS month,
    SUM(total_sale) AS total_sales
  FROM RetailSales
```

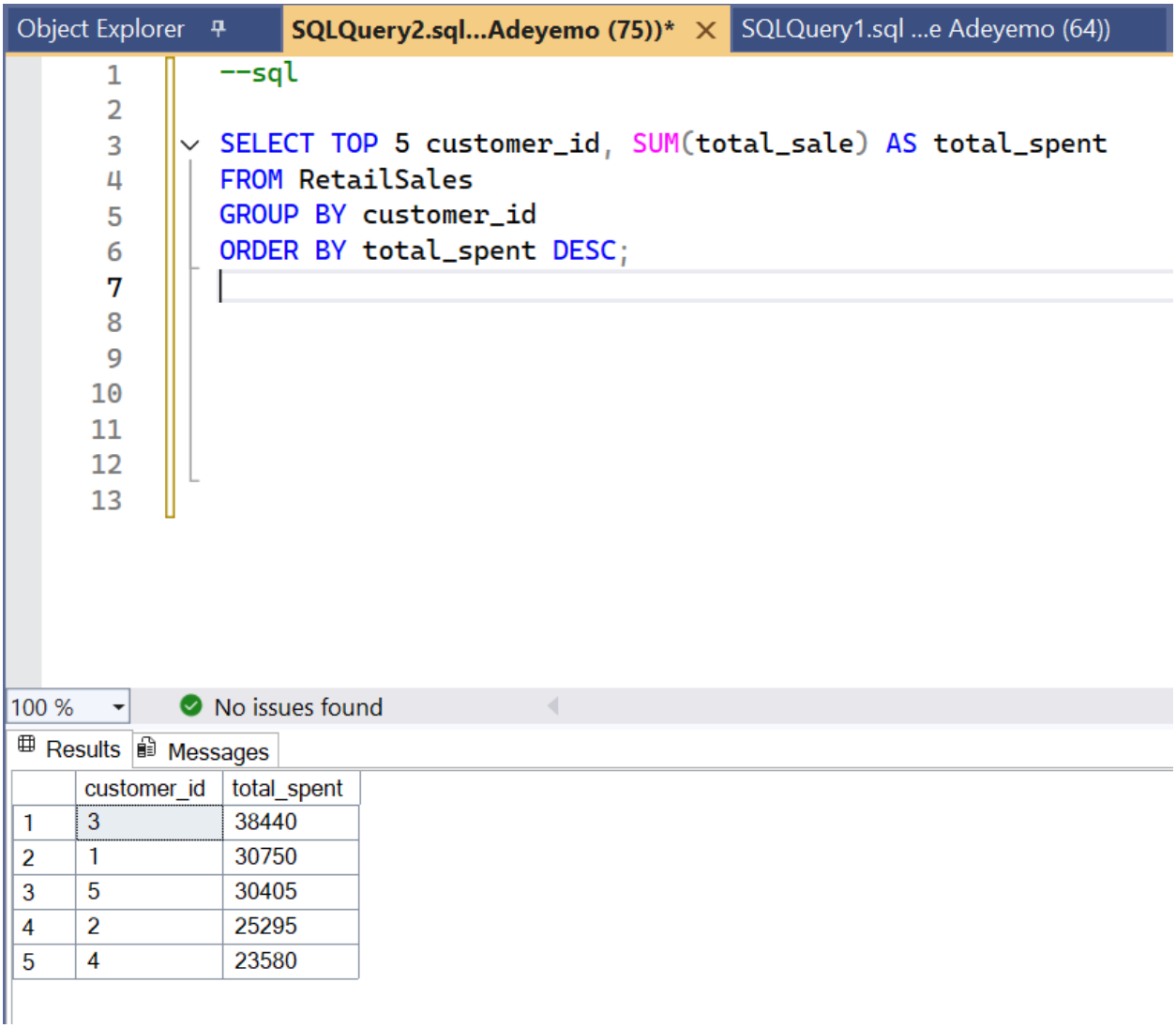
```
   GROUP BY YEAR(sale_date), MONTH(sale_date)
)
SELECT *
FROM MonthlySales ms
WHERE total_sales = (
   SELECT MAX(total_sales)
   FROM MonthlySales
   WHERE year = ms.year
);
```



viii. Top 5 customers by total sales

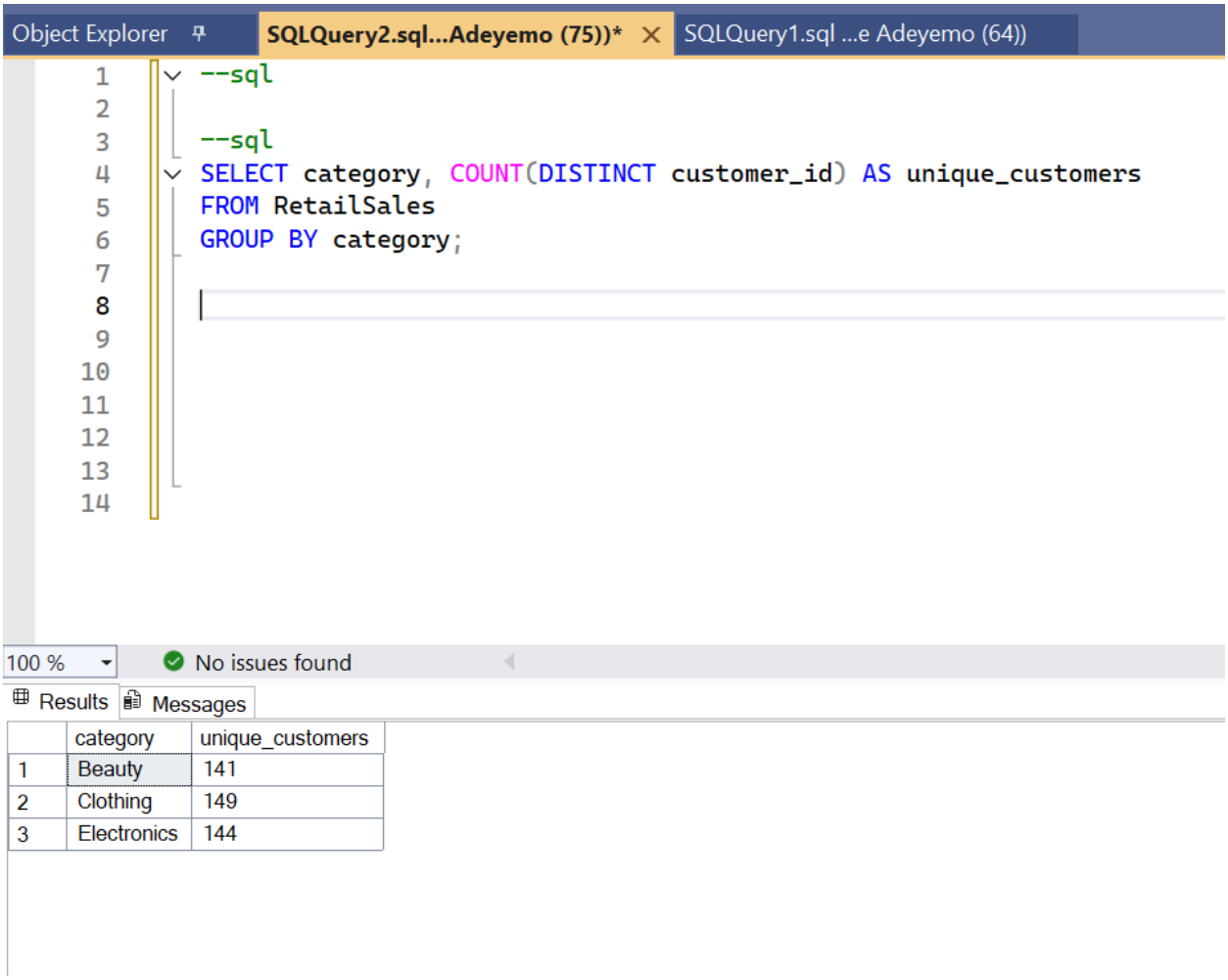
```
--sql

SELECT TOP 5 customer_id, SUM(total_sale) AS total_spent
FROM RetailSales
GROUP BY customer_id
ORDER BY total_spent DESC;
```



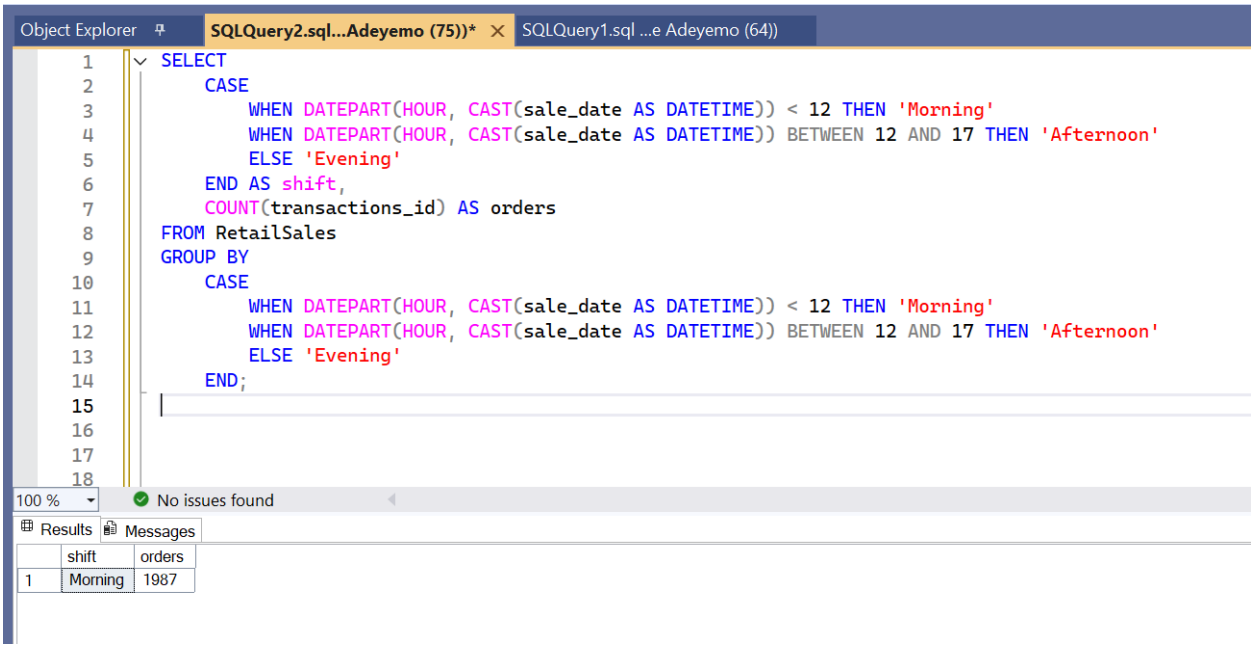
ix. Number of unique customers per category

--sql
SELECT category, COUNT(DISTINCT customer_id) AS unique_customers
FROM RetailSales
GROUP BY category;



x. Number of orders by shift

```
SELECT
  CASE
    WHEN DATEPART(HOUR, CAST(sale_date AS DATETIME)) < 12 THEN 'Morning'
    WHEN DATEPART(HOUR, CAST(sale_date AS DATETIME)) BETWEEN 12 AND 17 THEN 'Afternoon'
    ELSE 'Evening'
  END AS shift,
  COUNT(transactions_id) AS orders
FROM RetailSales
GROUP BY
  CASE
    WHEN DATEPART(HOUR, CAST(sale_date AS DATETIME)) < 12 THEN 'Morning'
    WHEN DATEPART(HOUR, CAST(sale_date AS DATETIME)) BETWEEN 12 AND 17 THEN 'Afternoon'
    ELSE 'Evening'
  END;
```



Step 4: Conclusion

- The dataset was successfully imported into SQL Server.
- Data cleaning handled missing values and removed incomplete records.
- All queries were executed successfully to extract insights, including:
    - Sales by date, category, and quantity
    - Best-selling months
    - Top customers
    - Customer demographics by category
    - Transaction patterns by shift