CUSTOMER INSIGHTS ANALYSIS

SQL CASE STUDY CHALLENGE

CONTENTS

1.Demographic Analysis

2.Shopping Behavior

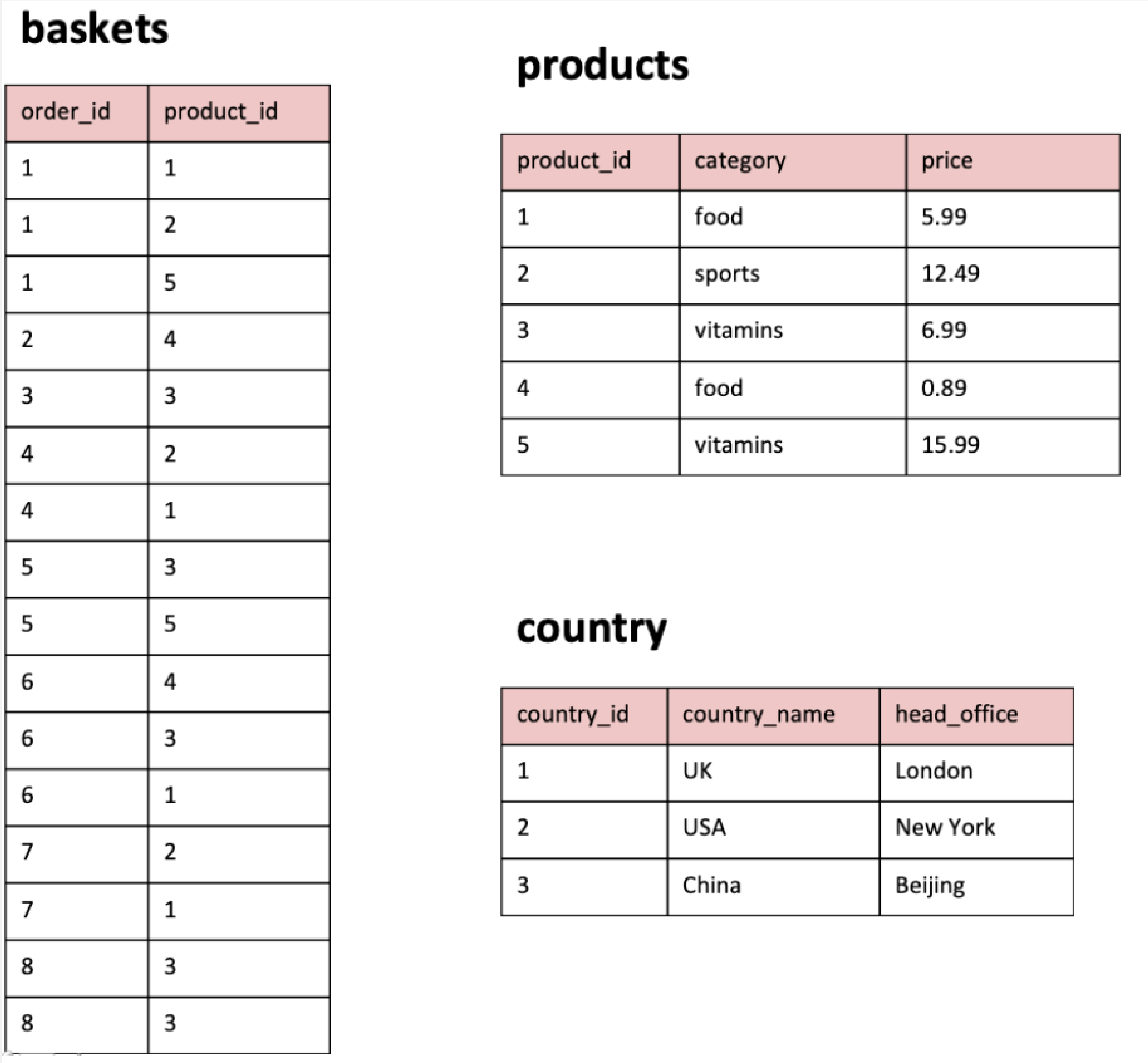
3.Product preferences

4.Geographics Analysis

5.Customer Loyalty

6.Sales performance

Tables



baskets

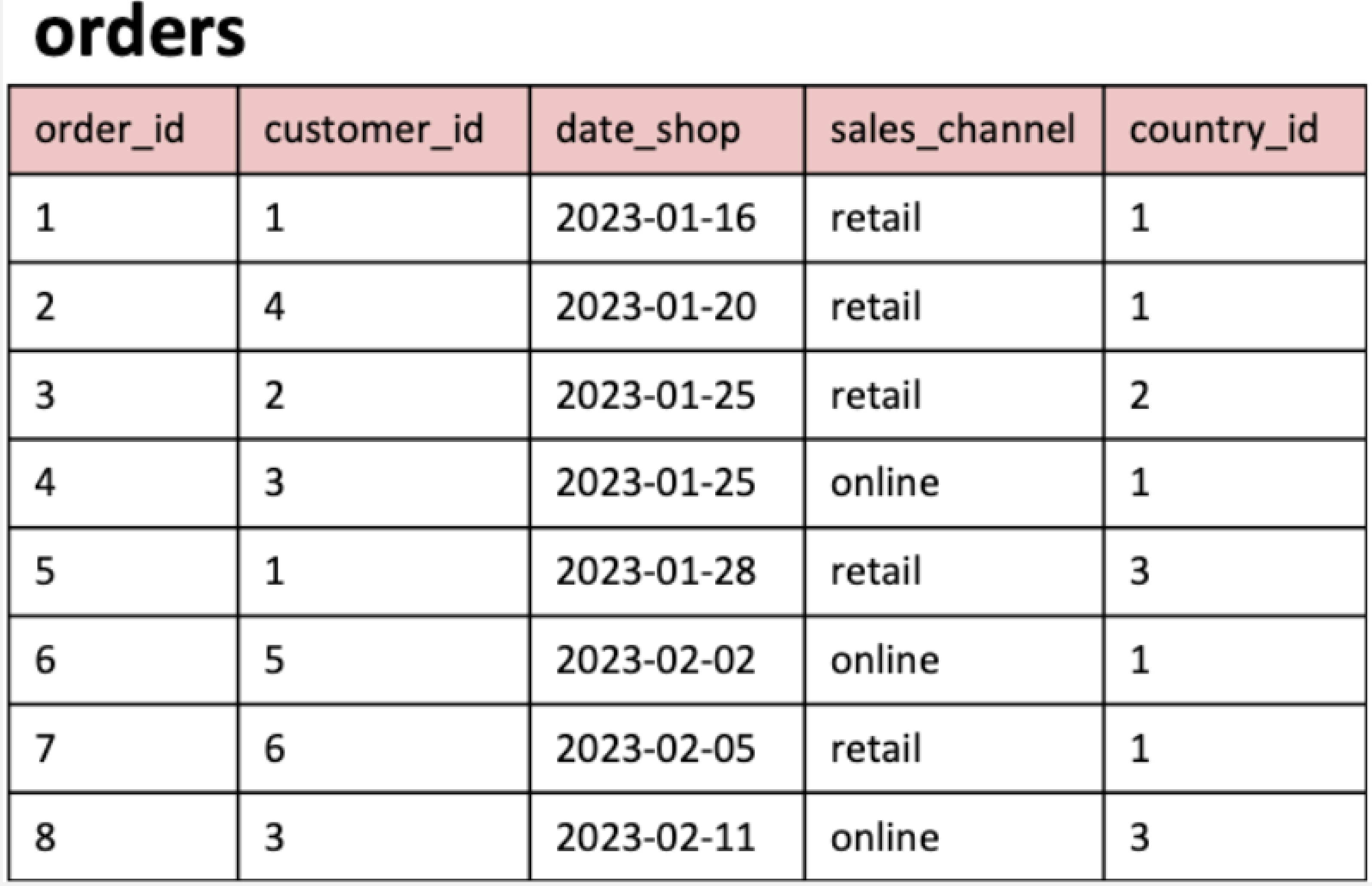
order_id	product_id
1	1
1	2
1	5
2	4
3	3
4	2
4	1
5	3
5	5
6	4
6	3
6	1
7	2
7	1
8	3
8	3

products

product_id	category	price
1	food	5.99
2	sports	12.49
3	vitamins	6.99
4	food	0.89
5	vitamins	15.99

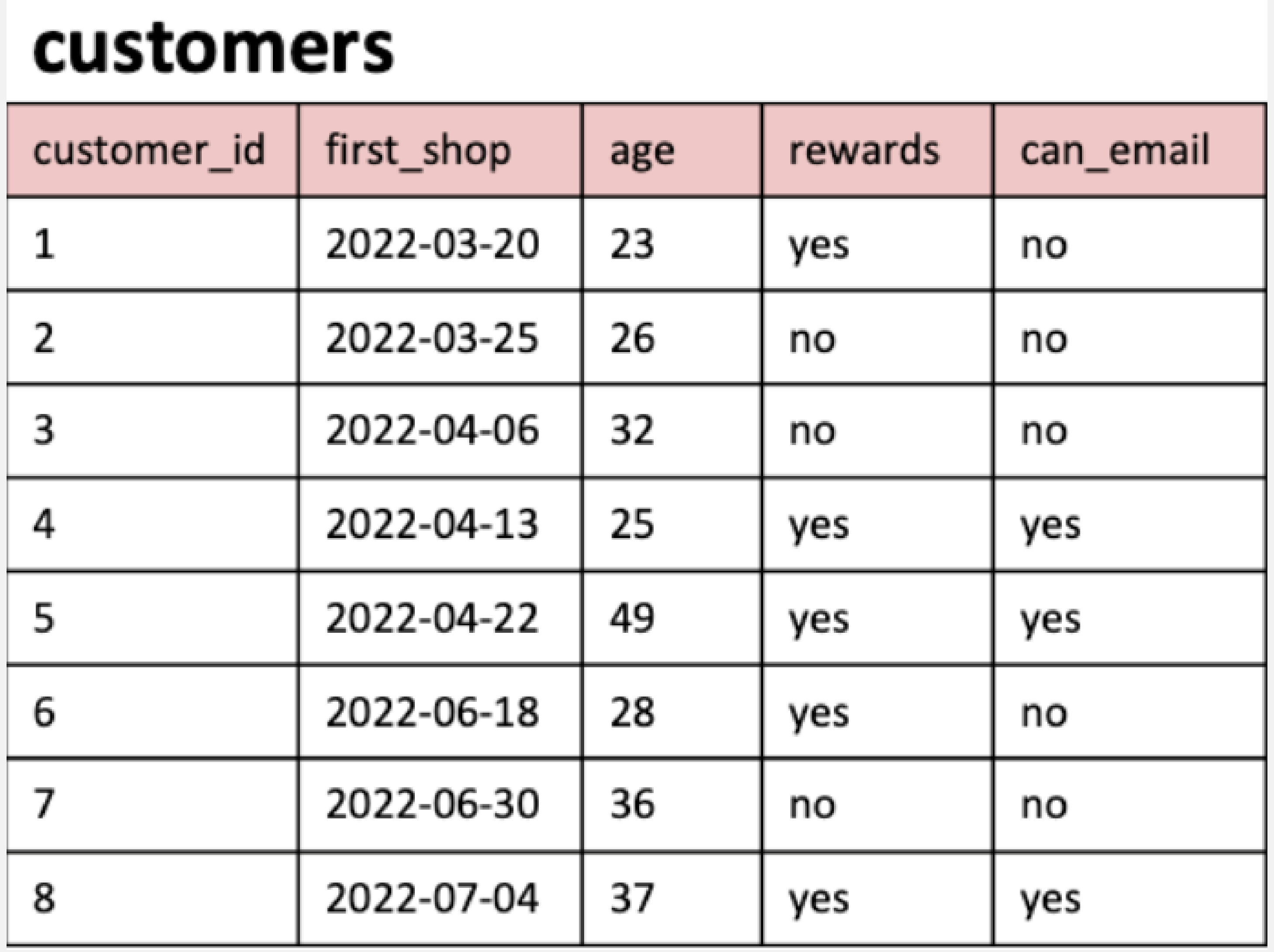
country

country_id	country_name	head_office
1	UK	London
2	USA	New York
3	China	Beijing



orders

order_id	customer_id	date_shop	sales_channel	country_id
1	1	2023-01-16	retail	1
2	4	2023-01-20	retail	1
3	2	2023-01-25	retail	2
4	3	2023-01-25	online	1
5	1	2023-01-28	retail	3
6	5	2023-02-02	online	1
7	6	2023-02-05	retail	1
8	3	2023-02-11	online	3

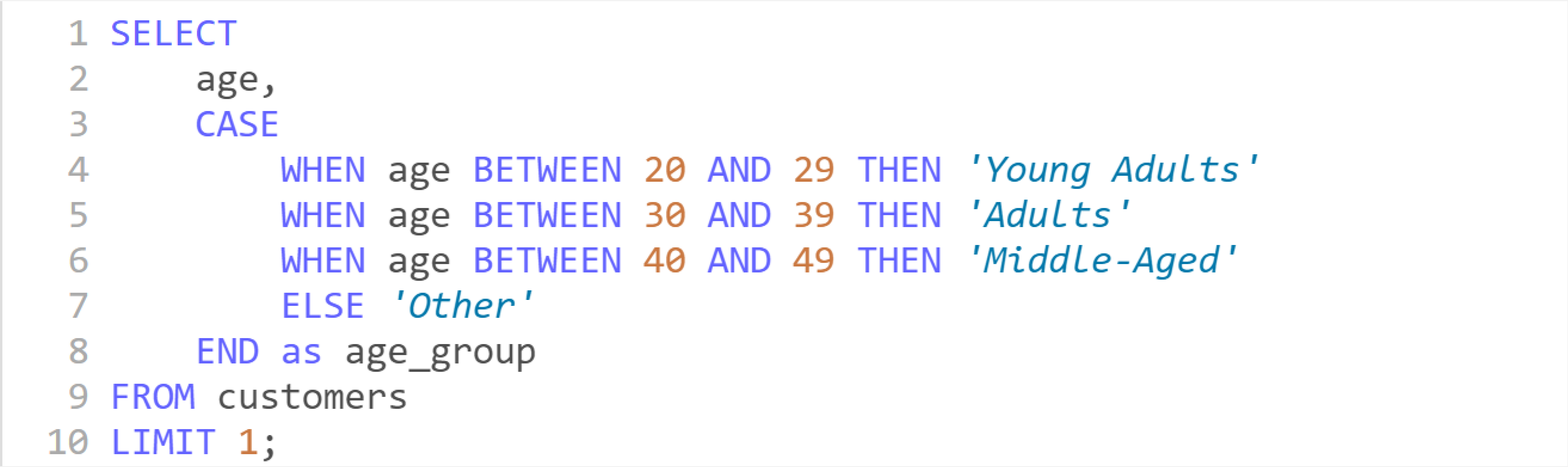


customers

customer_id	first_shop	age	rewards	can_email
1	2022-03-20	23	yes	no
2	2022-03-25	26	no	no
3	2022-04-06	32	no	no
4	2022-04-13	25	yes	yes
5	2022-04-22	49	yes	yes
6	2022-06-18	28	yes	no
7	2022-06-30	36	no	no
8	2022-07-04	37	yes	yes

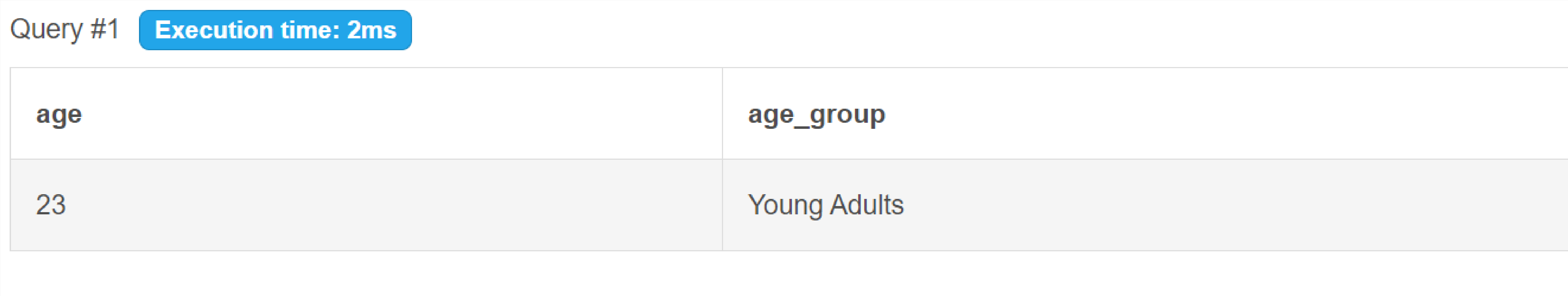
Demographic Analysis:

WHICH AGE GROUP HAS THE HIGHEST NUMBER OF CUSTOMERS?



```sql
SELECT
    age,
    CASE
        WHEN age BETWEEN 20 AND 29 THEN 'Young Adults'
        WHEN age BETWEEN 30 AND 39 THEN 'Adults'
        WHEN age BETWEEN 40 AND 49 THEN 'Middle-Aged'
        ELSE 'Other'
    END as age_group
FROM customers
LIMIT 1;
```

OUTPUT:

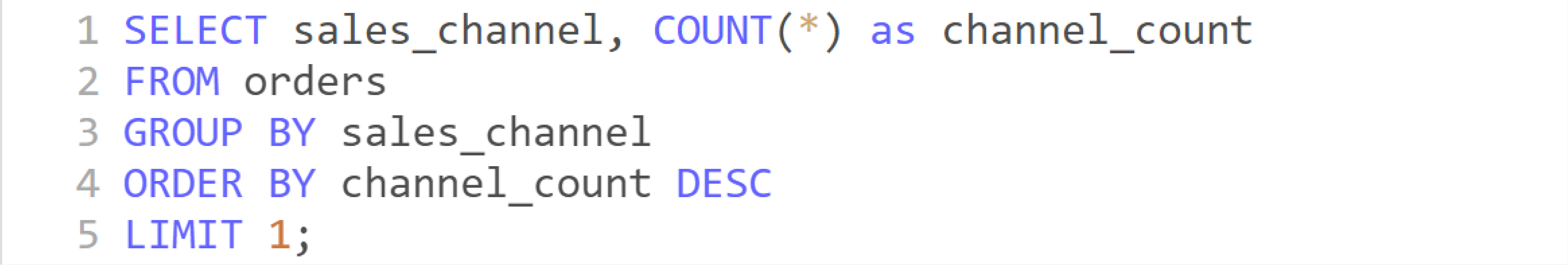


Query #1 Execution time: 2ms

age	age_group
23	Young Adults

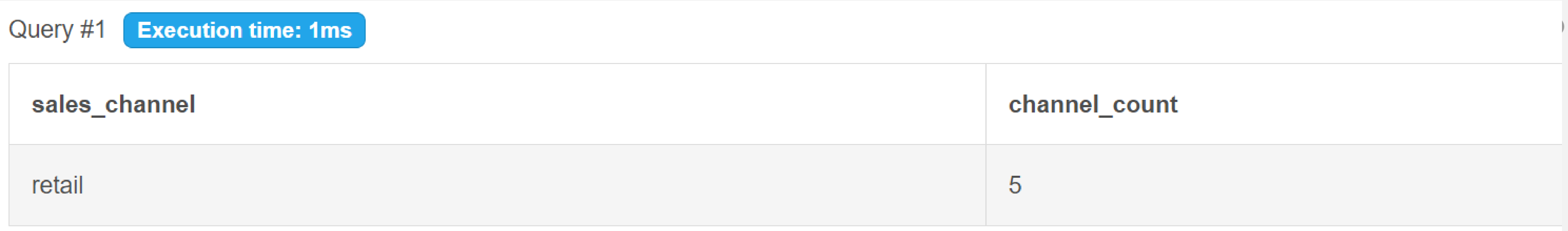
Shopping Behavior :

WHAT IS THE MOST COMMON SALES CHANNEL OUR CUSTOMERS? PREFER?



```sql
SELECT sales_channel, COUNT(*) as channel_count
FROM orders
GROUP BY sales_channel
ORDER BY channel_count DESC
LIMIT 1;
```

OUTPUT:

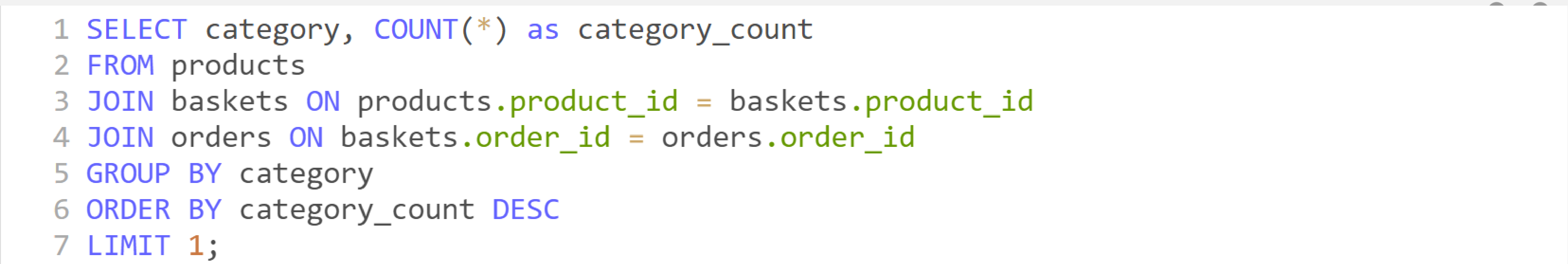


Query #1 Execution time: 1ms

sales_channel	channel_count
retail	5

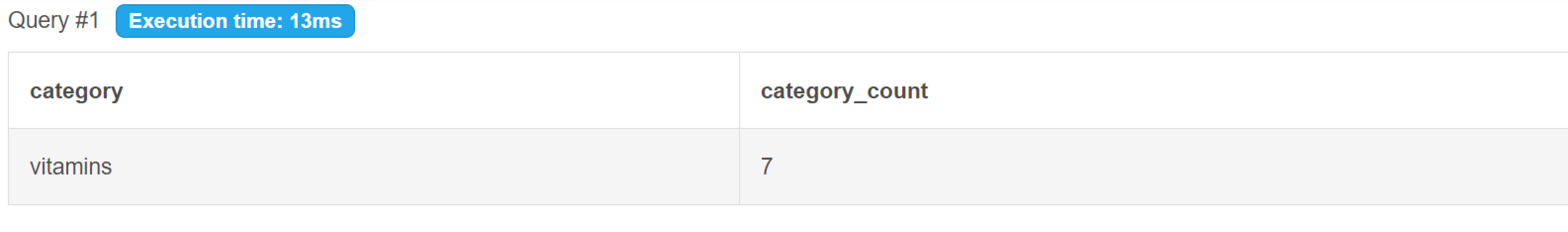
Product Preferences :

WHICH PRODUCT CATEGORY IS THE MOST POPULAR AMONG OUR CUSTOMERS?



```sql
SELECT category, COUNT(*) as category_count
FROM products
JOIN baskets ON products.product_id = baskets.product_id
JOIN orders ON baskets.order_id = orders.order_id
GROUP BY category
ORDER BY category_count DESC
LIMIT 1;
```

OUTPUT:

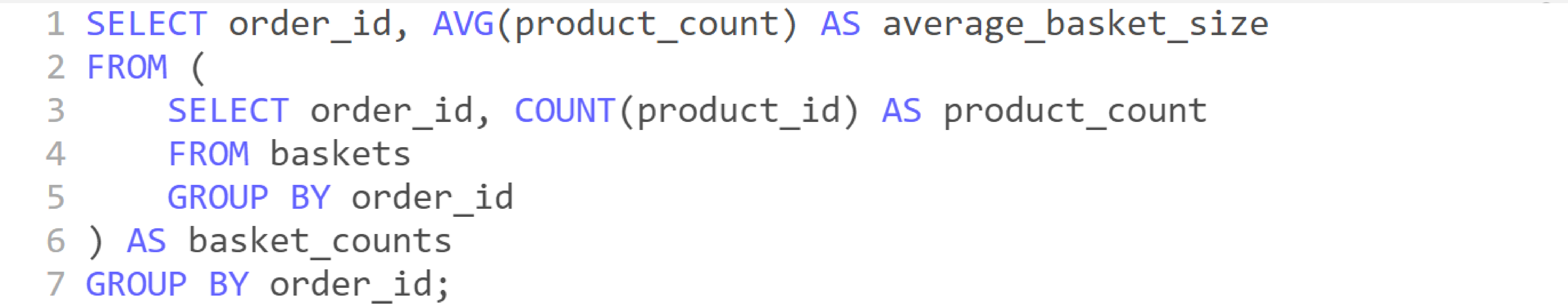


Query #1 Execution time: 13ms

category	category_count
vitamins	7

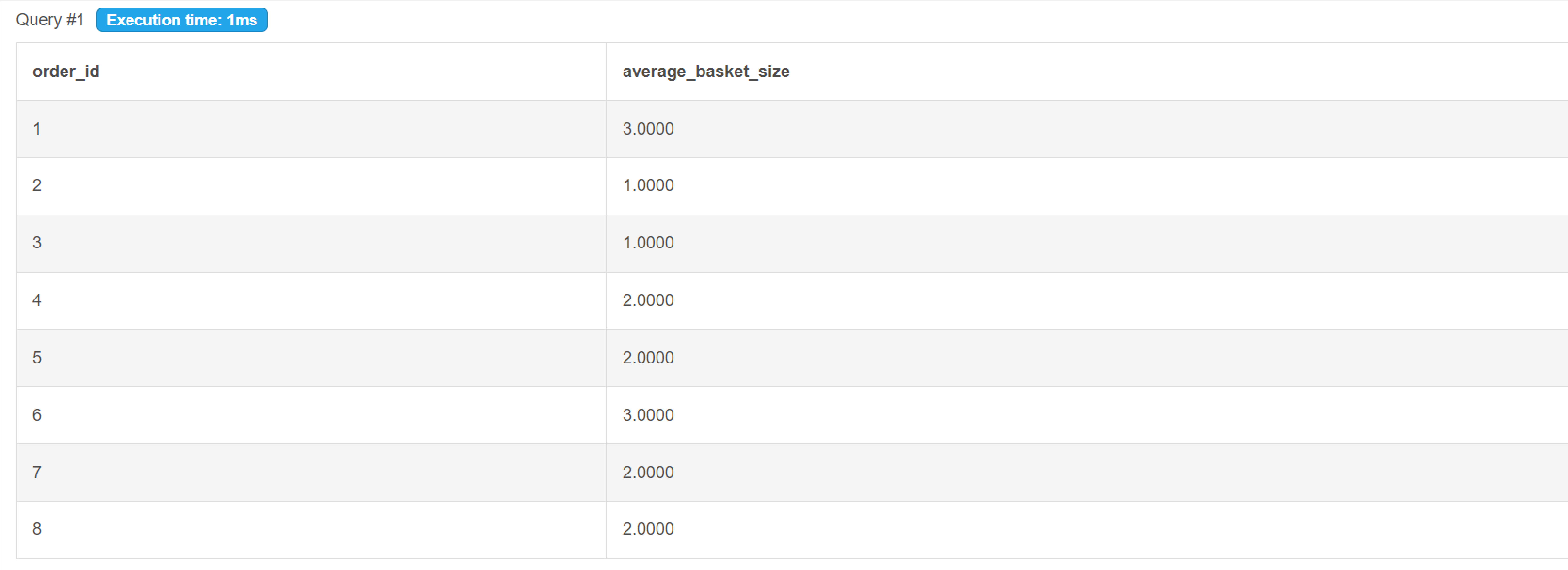
Product Preferences :

What is the average basket size (number of products) for each order?



```sql
SELECT order_id, AVG(product_count) AS average_basket_size
FROM (
    SELECT order_id, COUNT(product_id) AS product_count
    FROM baskets
    GROUP BY order_id
) AS basket_counts
GROUP BY order_id;
```

OUTPUT:

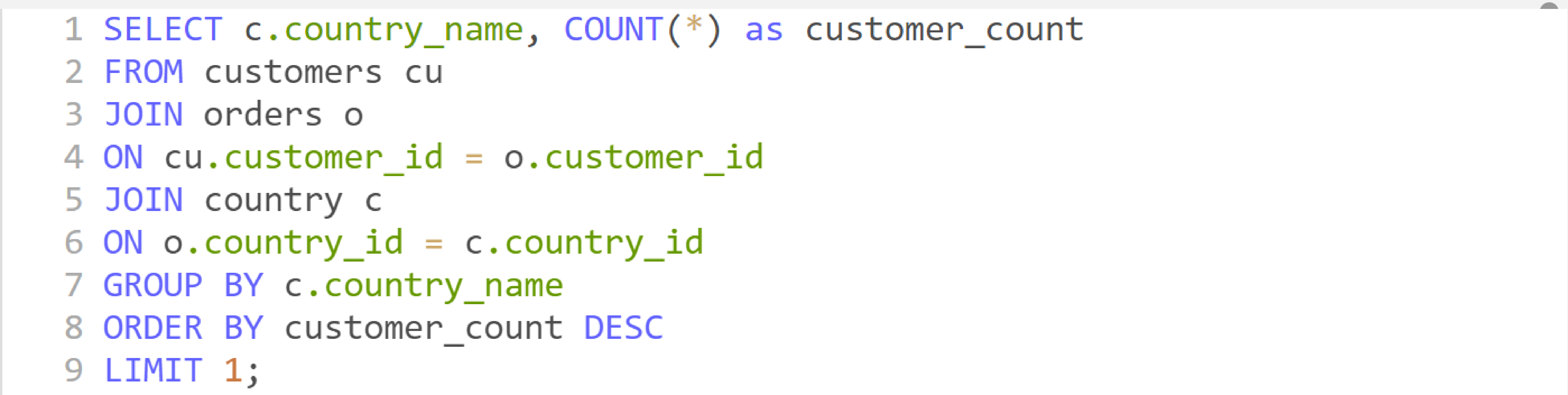


Query #1 Execution time: 1ms

order_id	average_basket_size
1	3.0000
2	1.0000
3	1.0000
4	2.0000
5	2.0000
6	3.0000
7	2.0000
8	2.0000

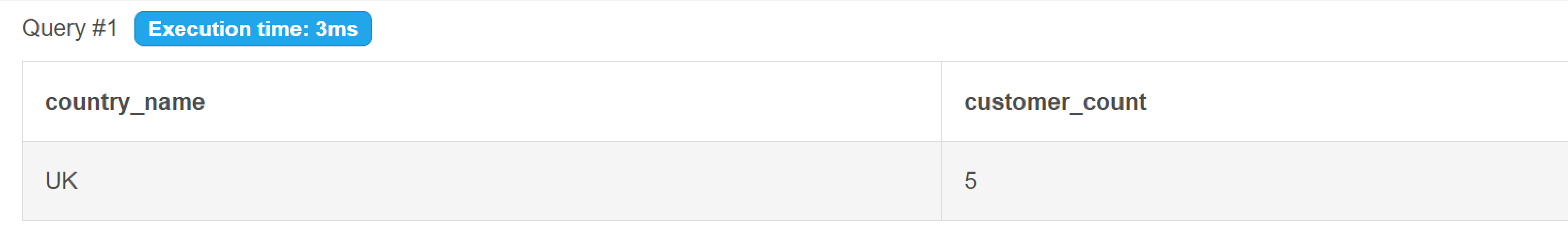
Geographic Analysis :

WHICH COUNTRY HAS THE HIGHEST NUMBER OF CUSTOMERS?



```sql
SELECT c.country_name, COUNT(*) as customer_count
FROM customers cu
JOIN orders o
ON cu.customer_id = o.customer_id
JOIN country c
ON o.country_id = c.country_id
GROUP BY c.country_name
ORDER BY customer_count DESC
LIMIT 1;
```

OUTPUT:

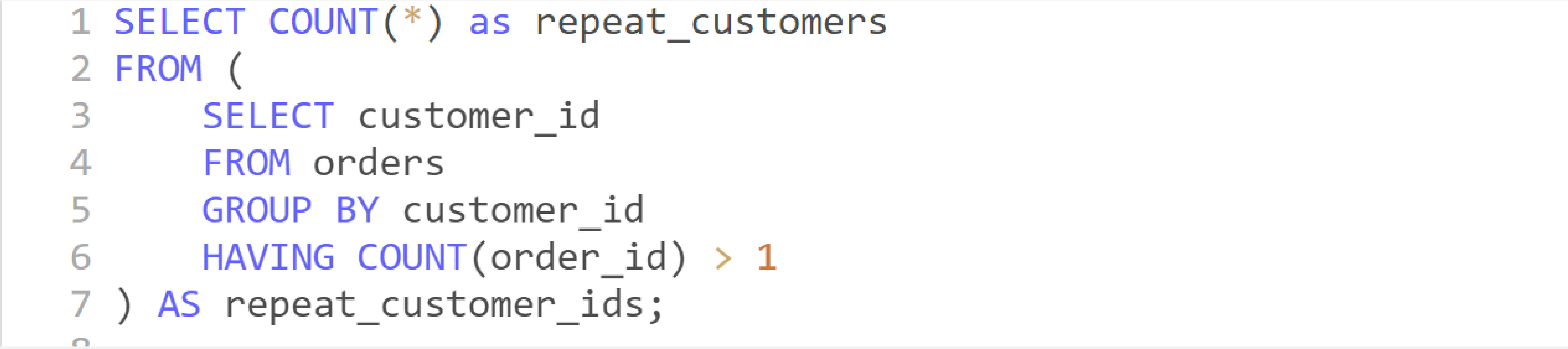


Query #1 Execution time: 3ms

country_name	customer_count
UK	5

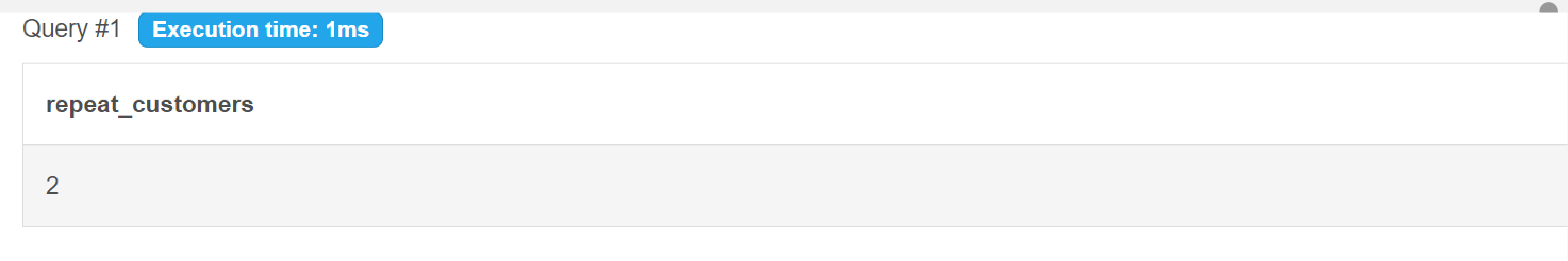
Customer Loyalty :

HOW MANY CUSTOMERS HAVE MADE REPEAT PURCHASES?



```sql
SELECT COUNT(*) as repeat_customers
FROM (
    SELECT customer_id
    FROM orders
    GROUP BY customer_id
    HAVING COUNT(order_id) > 1
) AS repeat_customer_ids;
```

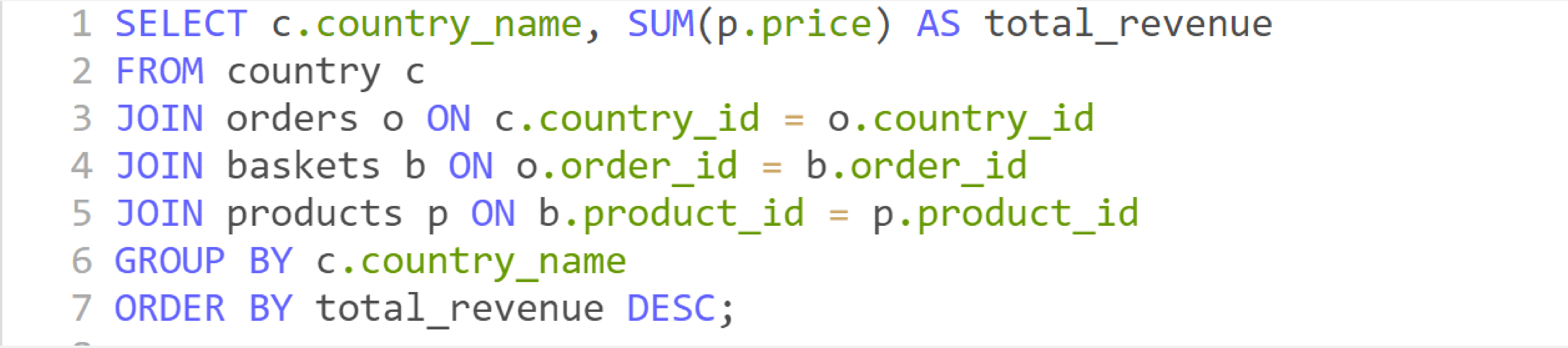
OUTPUT:



repeat_customers
2

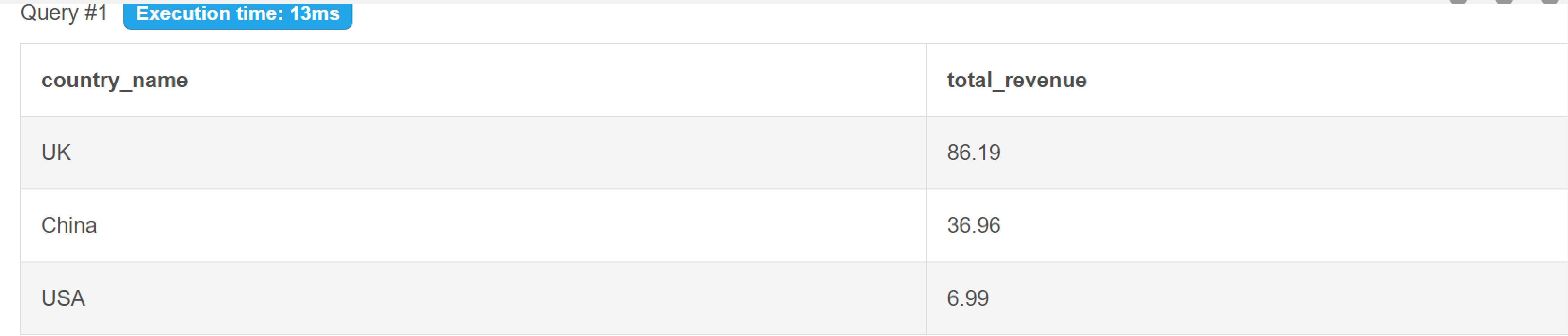
Sales Performance:

WHAT IS THE TOTAL REVENUE GENERATED FROM EACH COUNTRY?



```sql
SELECT c.country_name, SUM(p.price) AS total_revenue
FROM country c
JOIN orders o ON c.country_id = o.country_id
JOIN baskets b ON o.order_id = b.order_id
JOIN products p ON b.product_id = p.product_id
GROUP BY c.country_name
ORDER BY total_revenue DESC;
```

OUTPUT:

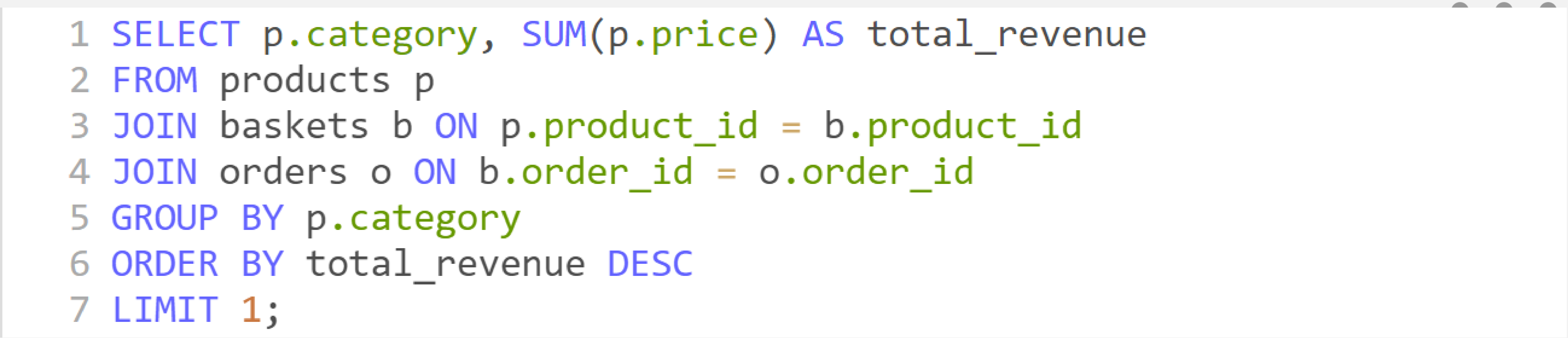


Query #1 Execution time: 13ms

country_name	total_revenue
UK	86.19
China	36.96
USA	6.99

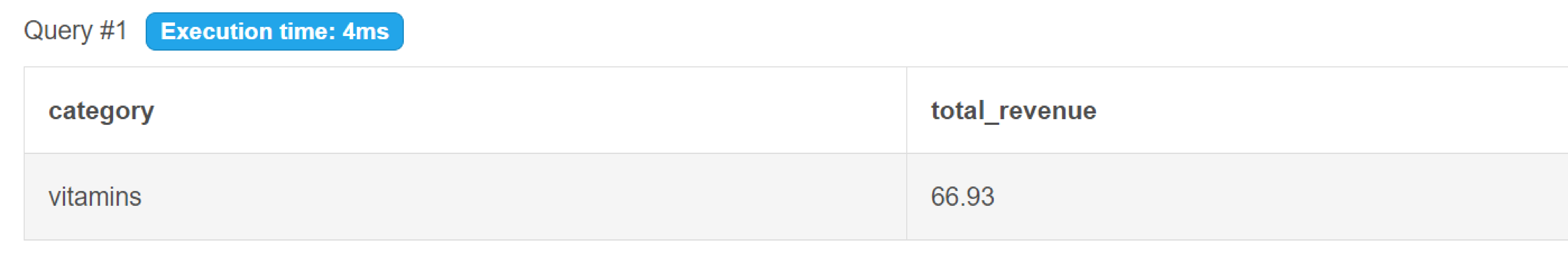
Sales Performance:

Which product category contributes the most to our revenue?



```sql
SELECT p.category, SUM(p.price) AS total_revenue
FROM products p
JOIN baskets b ON p.product_id = b.product_id
JOIN orders o ON b.order_id = o.order_id
GROUP BY p.category
ORDER BY total_revenue DESC
LIMIT 1;
```

OUTPUT:



Query #1 Execution time: 4ms

category	total_revenue
vitamins	66.93

THANK YOU!