

## SQL CHALLENGE 3: CUSTOMER INSIGHTS

You are a customer insights analyst for 'The General Store'. Can you analyse the following tables to find out crucial information about your customers to provide to your marketing team.

JERMAINE SANGIWA

## 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

## 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

## 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

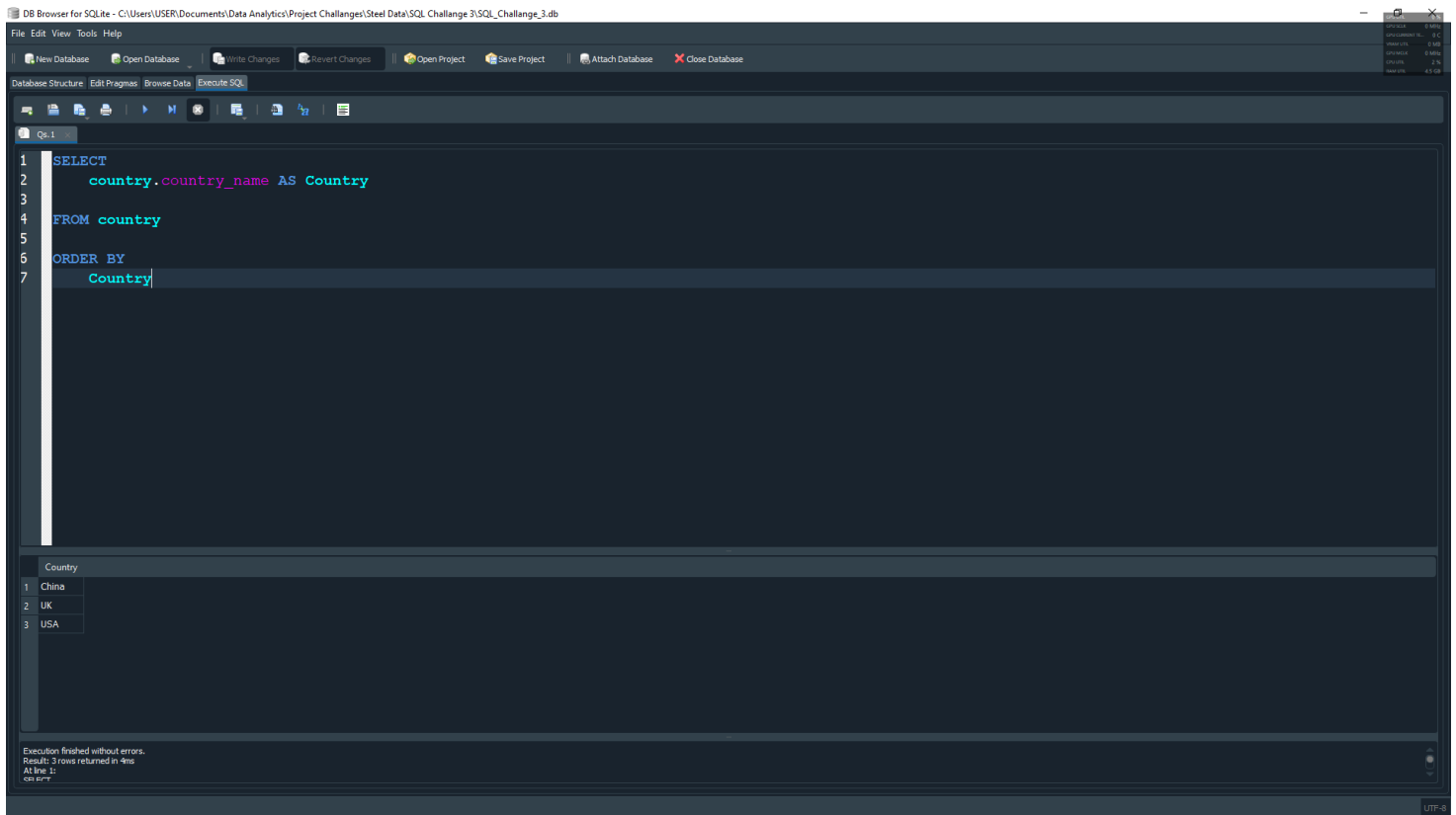
## products

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

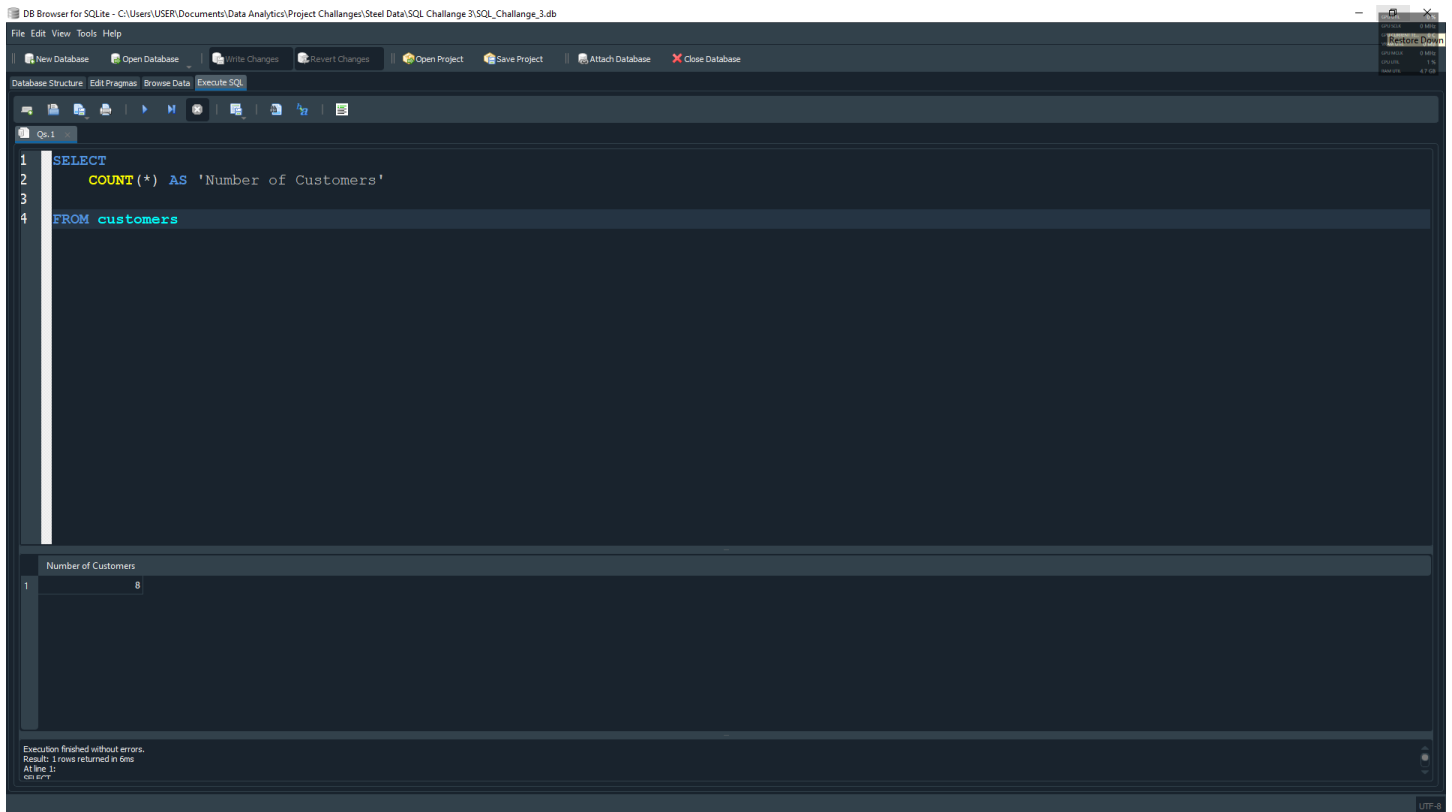
## QUESTION 1: WHAT ARE THE NAMES OF ALL THE COUNTRIES IN THE COUNTRY TABLE?



The name of all the countries in the country table are:

- **China**
- **United Kingdom**
- **United States of America**

## QUESTION 2: WHAT IS THE TOTAL NUMBER OF CUSTOMERS IN THE CUSTOMER TABLE?



The screenshot shows the DB Browser for SQLite application. The main window displays a SQL query in the editor:

```
1 SELECT
2   COUNT(*) AS 'Number of Customers'
3
4 FROM customers
```

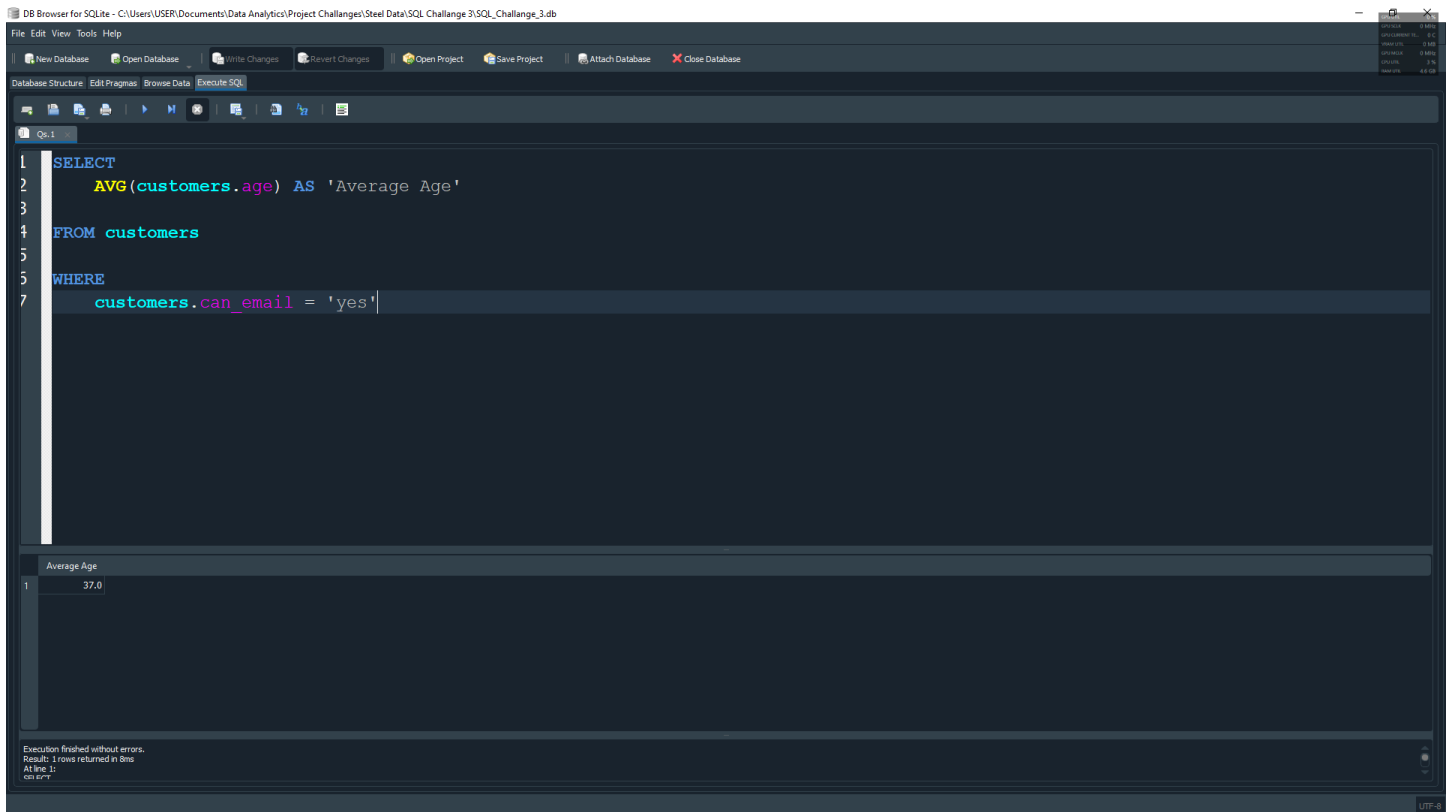
Below the editor, the results pane shows a single row with the value 8 under the column header 'Number of Customers'.

	Number of Customers
1	8

At the bottom of the window, a status bar indicates: "Execution finished without errors. Results: 1 rows returned in 6ms. At line 1: 0/0 OK".

The number of customers in the customer table is **8**.

QUESTION 3: WHAT IS THE AVERAGE AGE OF CUSTOMERS WHO CAN RECEIVE MARKETING EMAIL ('can\_email' is set to 'yes')?



The screenshot shows the DB Browser for SQLite interface. The SQL editor contains the following query:

```
1 SELECT
2     AVG(customers.age) AS 'Average Age'
3
4 FROM customers
5
6 WHERE
7     customers.can_email = 'yes'
```

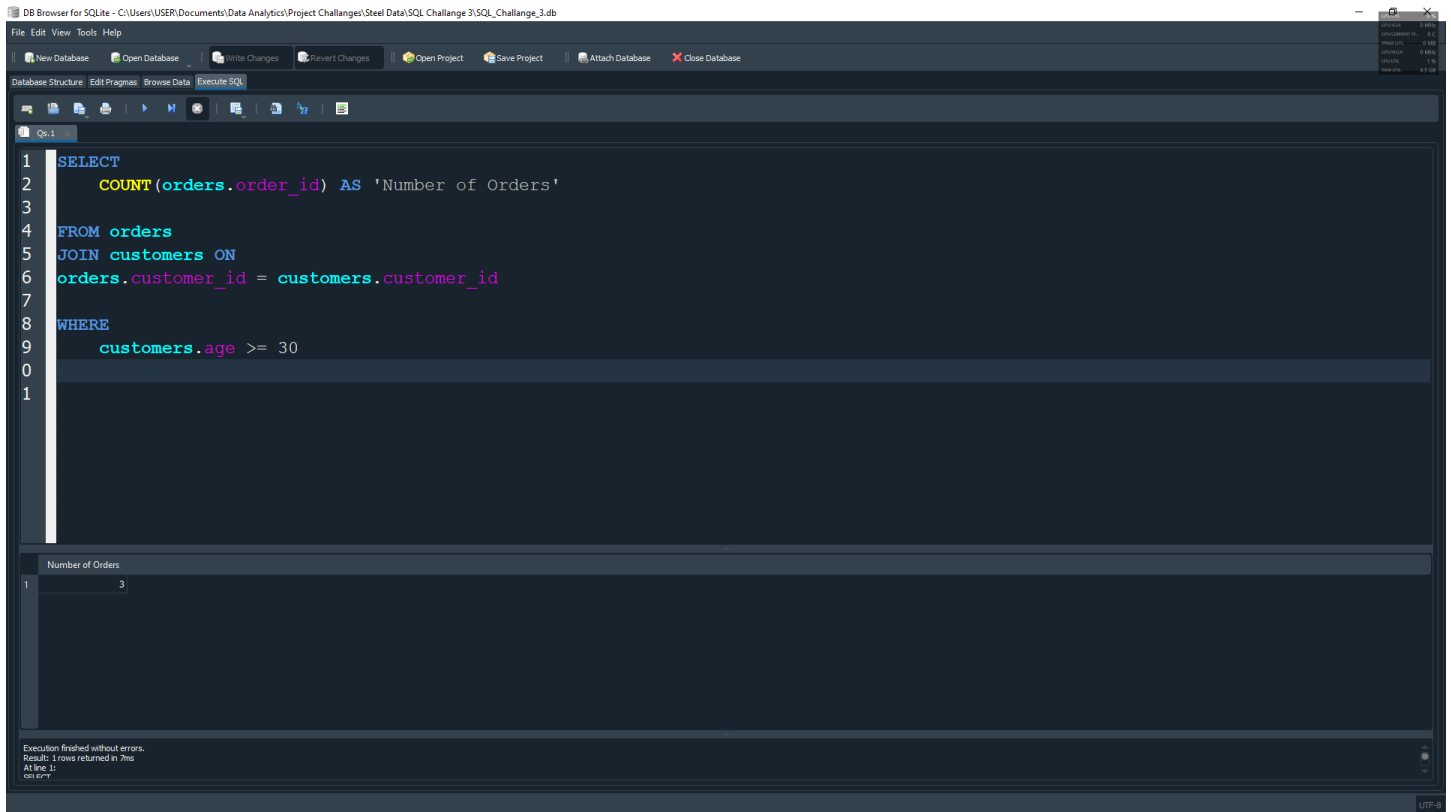
Below the editor, the results are displayed in a table with the column 'Average Age'.

	Average Age
1	37.0

At the bottom, a status bar indicates: "Execution finished without errors. Results: 1 rows returned in 0ms. At line 1: 0/0 OK".

The average age of customers who can receive marketing email is **37 years old**.

## QUESTION 4: HOW MANY ORDERS WERE MADE BY CUSTOMERS AGED 30 OR OLDER?



The screenshot shows a SQL database browser interface. The main window displays a SQL query for counting orders made by customers aged 30 or older. The query is as follows:

```
1 SELECT
2     COUNT(orders.order_id) AS 'Number of Orders'
3
4 FROM orders
5 JOIN customers ON
6     orders.customer_id = customers.customer_id
7
8 WHERE
9     customers.age >= 30
10
11
```

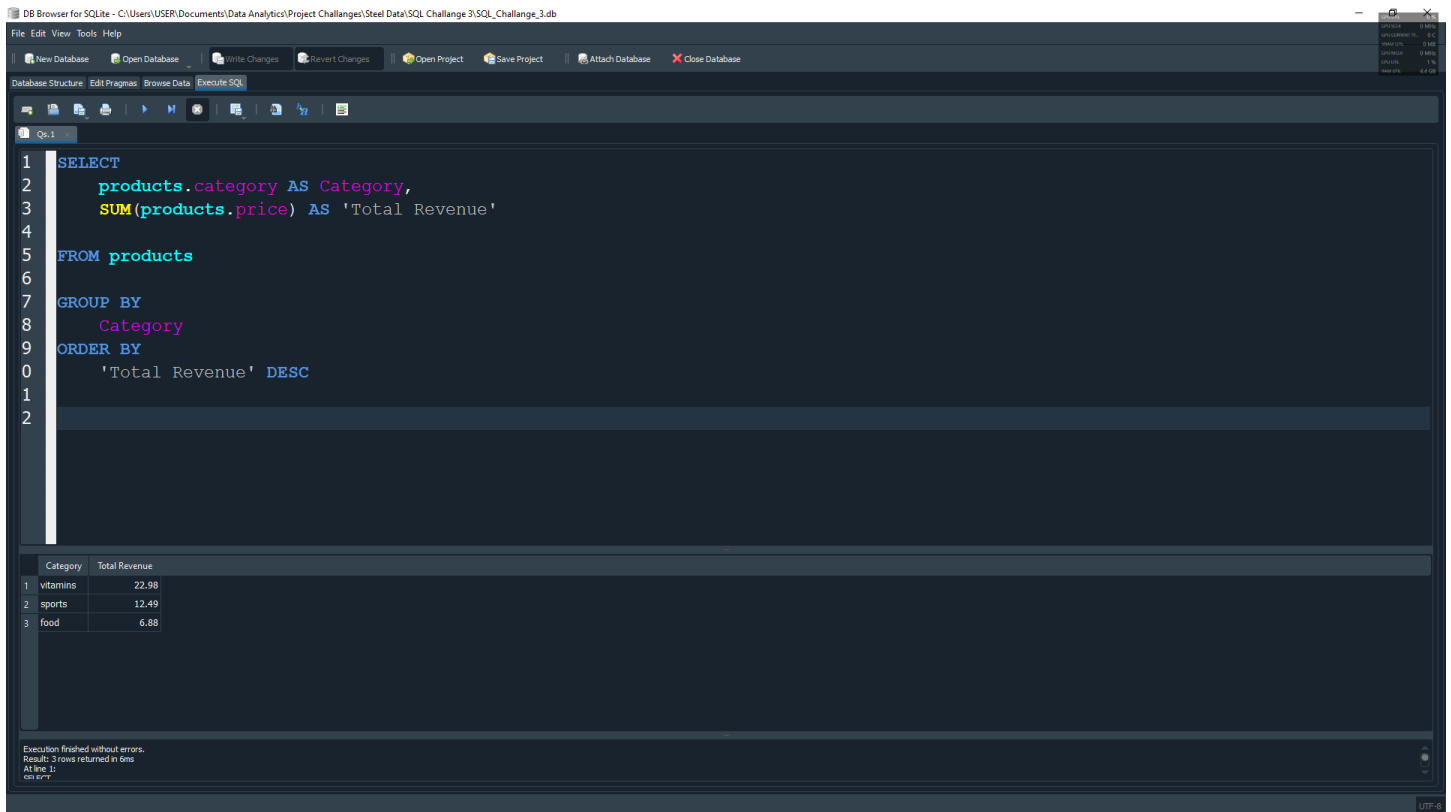
Below the query editor, the results are displayed in a table with the following structure:

	Number of Orders
1	3

At the bottom of the interface, a status bar indicates: "Execution finished without errors. Result: 1 rows returned in 7ms. At line 11, col 87".

The number of orders that were made by customers aged 30 or older are **3**

## QUESTION 5: WHAT IS THE TOTAL REVENUE GENERATED BY EACH PRODUCT CATEGORY?



The screenshot shows the DB Browser for SQLite interface. The SQL query is as follows:

```
1 SELECT
2   products.category AS Category,
3   SUM(products.price) AS 'Total Revenue'
4
5 FROM products
6
7 GROUP BY
8   Category
9 ORDER BY
10  'Total Revenue' DESC
11
12
```

The results are displayed in a table with the following data:

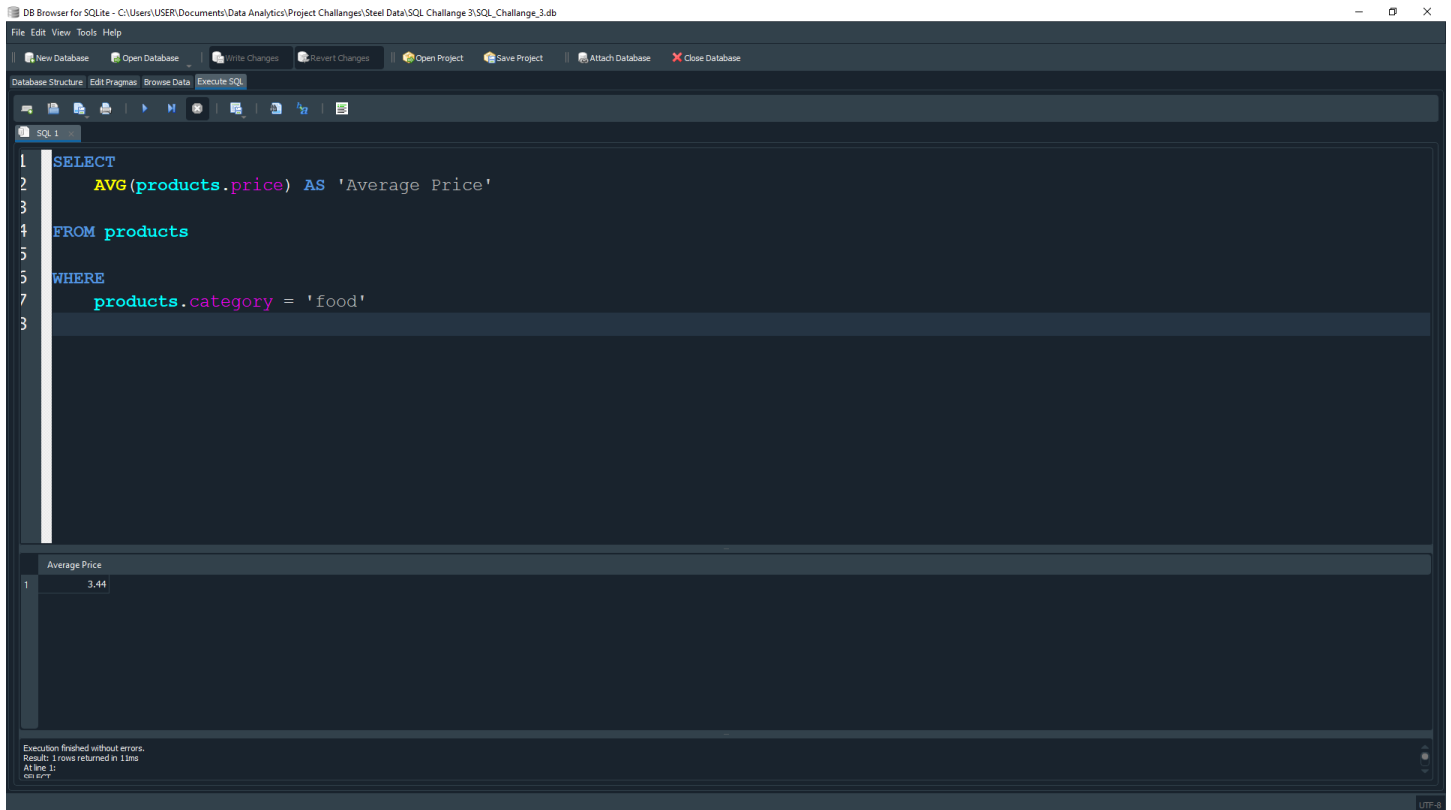
	Category	Total Revenue
1	vitamins	22.98
2	sports	12.49
3	food	6.88

Execution finished without errors.  
Results: 3 rows returned in 6ms.  
At line 1:  
OK OK Y

The total revenue generated by each product category are:

- **Vitamins: \$22.98**
- **Sports: \$12.49**
- **Food: \$6.88**

## QUESTION 6: WHAT IS THE AVERAGE OF PRODUCTS IN THE 'FOOD' CATEGORY?



The screenshot shows the DB Browser for SQLite application. The title bar indicates the file path: C:\Users\USER\Documents\Data Analytics\Project Challenge\Steel Data\SQL Challenge 3\SQL\_Challenge\_3.db. The menu bar includes File, Edit, View, Tools, and Help. The toolbar contains icons for New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, and Close Database. The main window is titled 'SQL 1' and contains the following SQL query:

```
1 SELECT
2     AVG(products.price) AS 'Average Price'
3
4 FROM products
5
6 WHERE
7     products.category = 'food'
8
```

Below the query editor, the results pane shows a single row with the column header 'Average Price' and the value '3.44'.

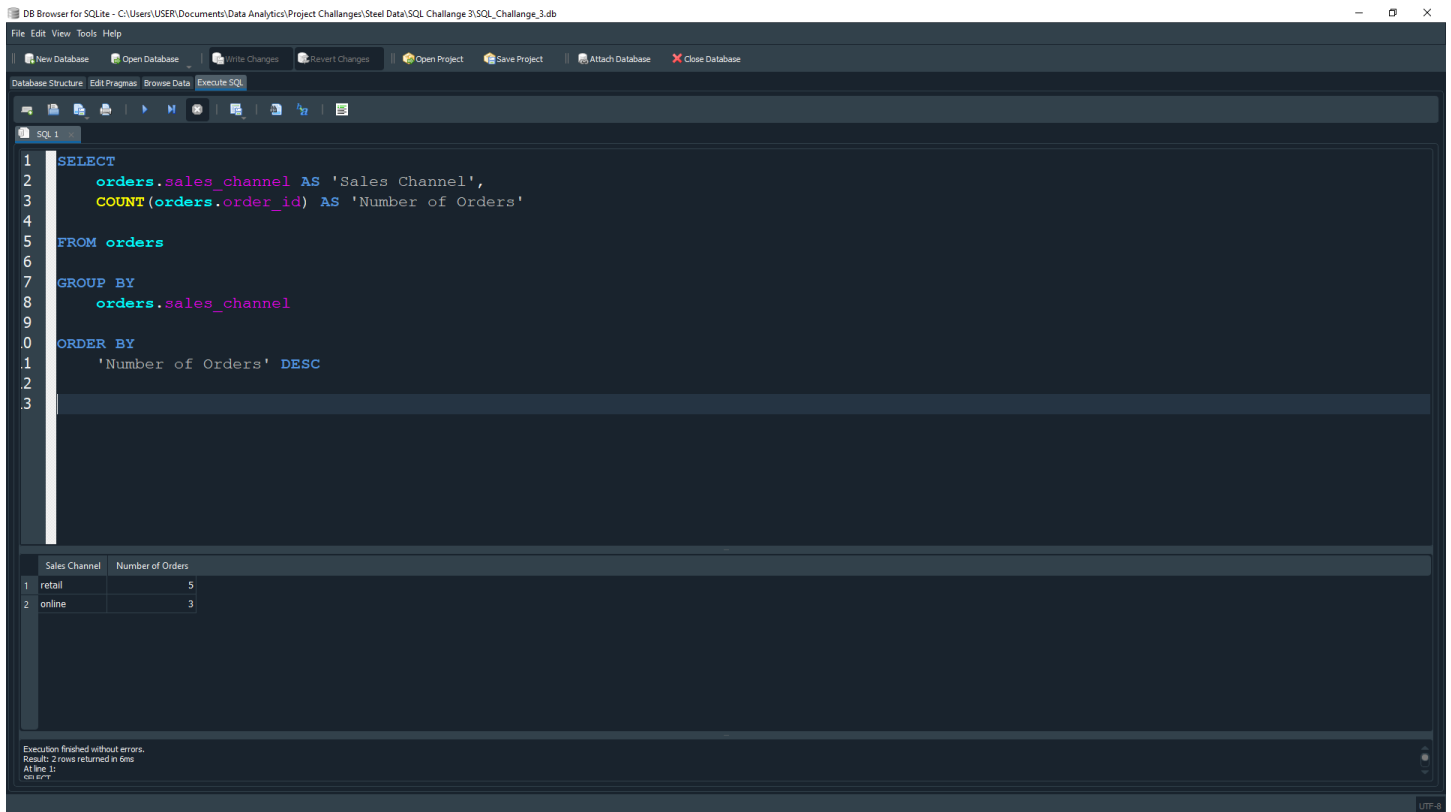
Average Price
3.44

At the bottom of the application, a status bar indicates: 'Execution finished without errors. Result: 1 rows returned in 11ms. At line 11, col 80'.

The average price of products in the 'food' category group is **\$3.44**.



QUESTION 7: HOW MANY ORDERS WERE MADE IN EACH SALES CHANNEL (sales\_channel COLUMN) IN THE ORDERS TABLE?



The screenshot shows the DB Browser for SQLite application. The SQL editor contains the following query:

```
1 SELECT
2   orders.sales_channel AS 'Sales Channel',
3   COUNT(orders.order_id) AS 'Number of Orders'
4
5 FROM orders
6
7 GROUP BY
8   orders.sales_channel
9
10 ORDER BY
11   'Number of Orders' DESC
12
13
```

The results pane displays the following data:

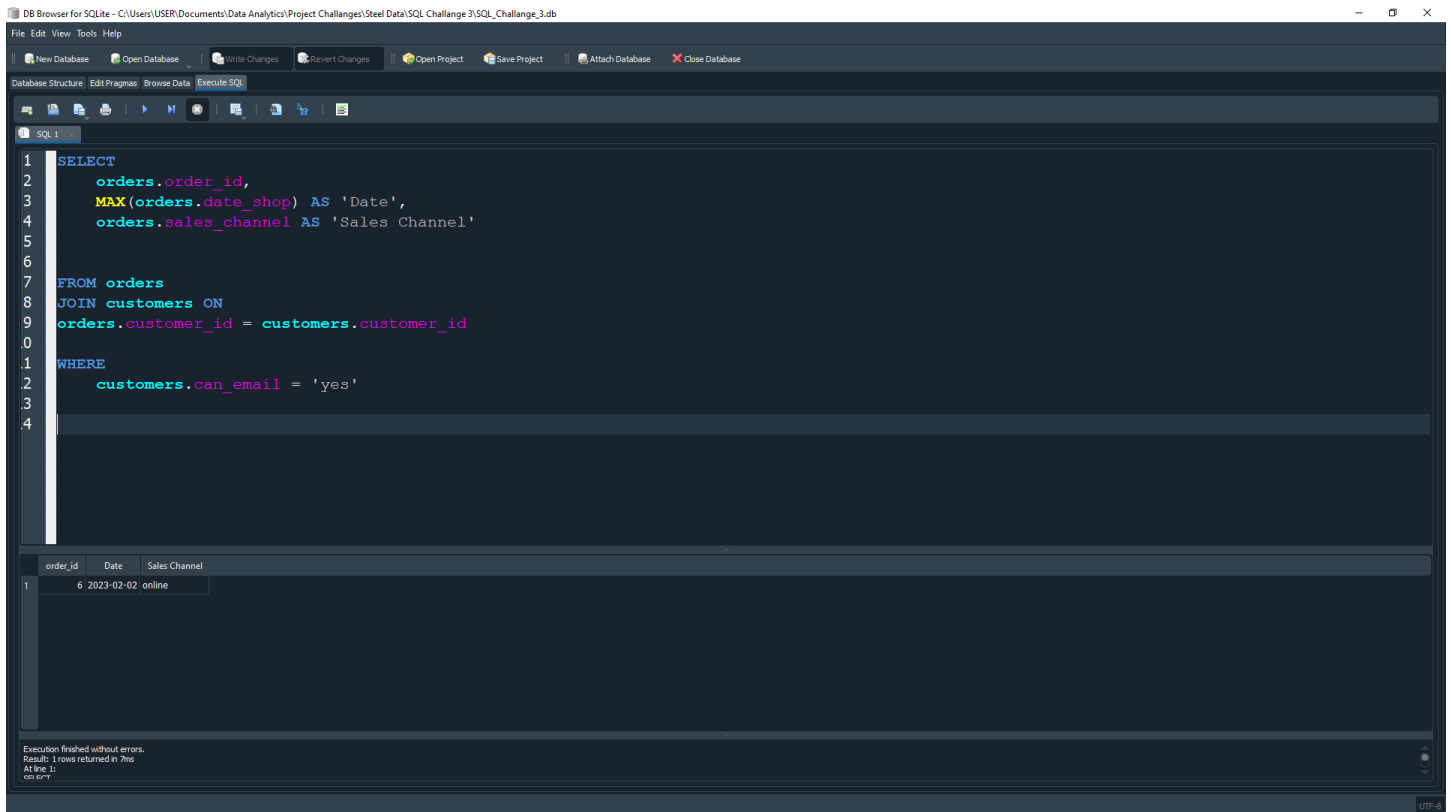
Sales Channel	Number of Orders
1 retail	5
2 online	3

At the bottom, a status bar indicates: "Execution finished without errors. Results: 2 rows returned in 6ms. At line 11: 0/0 OK."

Number of orders made in each sales channel are:

- **Retail: 5**
- **Online: 3**

## QUESTION 8: WHAT IS THE DATE OF THE LATEST ORDER MADE BY A CUSTOMER WHO CAN RECEIVE MARKETING EMAILS?



The screenshot shows a SQL query executed in a DB Browser for SQLite window. The query is as follows:

```
1 SELECT
2     orders.order_id,
3     MAX(orders.date_shop) AS 'Date',
4     orders.sales_channel AS 'Sales Channel'
5
6
7 FROM orders
8 JOIN customers ON
9     orders.customer_id = customers.customer_id
10
11 WHERE
12     customers.can_email = 'yes'
```

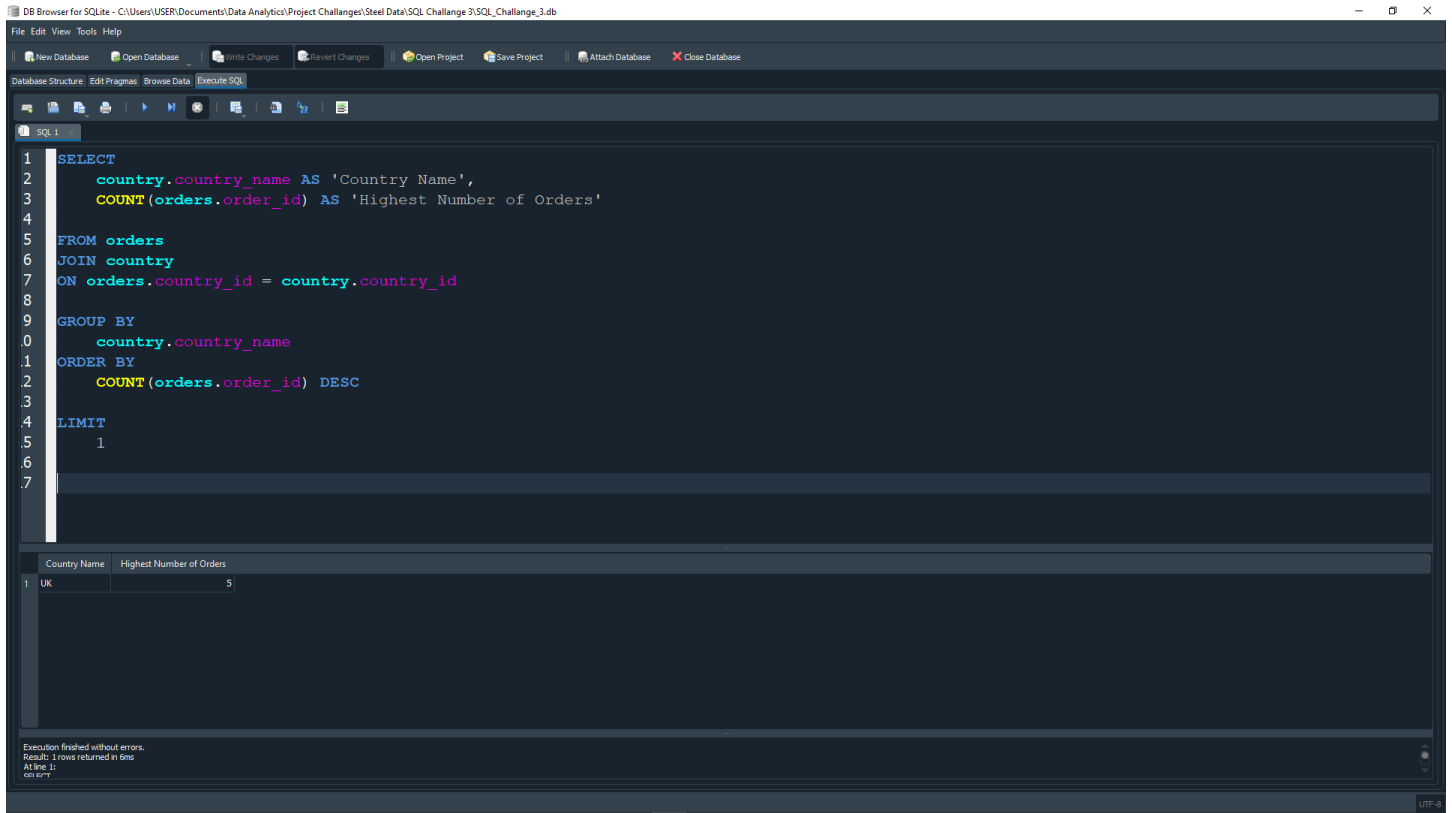
The result of the query is displayed in a table below the query editor:

order_id	Date	Sales Channel
6	2023-02-02	online

Execution finished without errors.  
Results: 1 rows returned in 7ms.  
At line 1:  
OK OK Y

The date of the latest order made by a customer who can receive marketing emails are **2<sup>nd</sup> February, 2023**

## QUESTION 9: WHAT IS THE NAME OF THE COUNTRY WITH THE HIGHEST NUMBER OF ORDERS?



The screenshot shows a DB Browser for SQLite window. The SQL editor contains the following query:

```
1 SELECT
2   country.country_name AS 'Country Name',
3   COUNT(orders.order_id) AS 'Highest Number of Orders'
4
5 FROM orders
6 JOIN country
7 ON orders.country_id = country.country_id
8
9 GROUP BY
10  country.country_name
11 ORDER BY
12  COUNT(orders.order_id) DESC
13
14 LIMIT
15  1
16
17
```

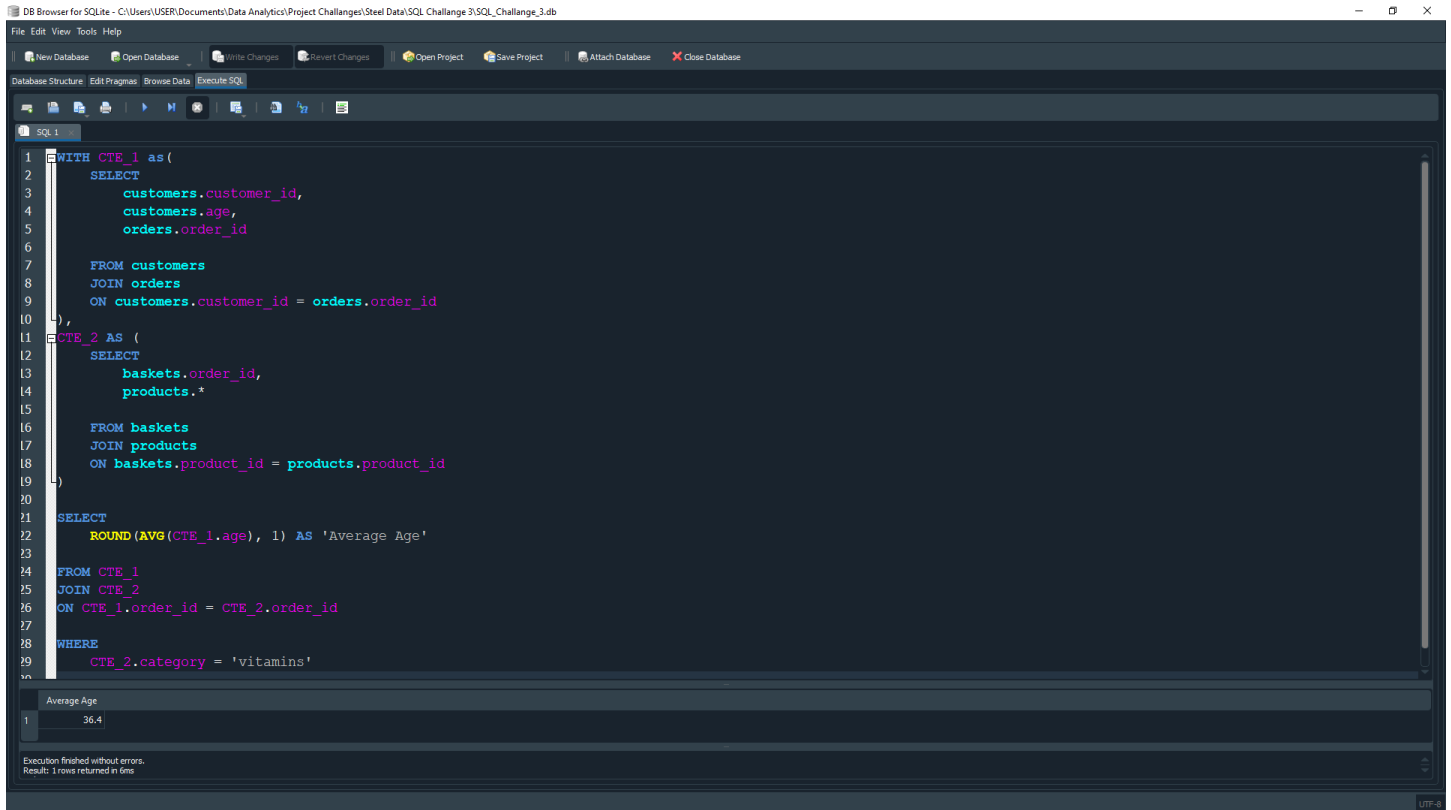
Below the editor, the results are displayed in a table:

	Country Name	Highest Number of Orders
1	UK	5

At the bottom of the window, a status bar indicates: "Execution finished without errors. Results: 1 rows returned in 6ms. At line 11: OK (0/1)".

The name of the country with the highest number of orders is **United Kingdom**.

## QUESTION 10: WHAT IS THE AVERAGE AGE OF CUSTOMERS WHO MADE ORDERS IN THE 'vitamins' PRODUCT CATEGORY?



The screenshot shows the DB Browser for SQLite interface. The SQL editor contains the following query:

```
1 WITH CTE_1 as (  
2     SELECT  
3         customers.customer_id,  
4         customers.age,  
5         orders.order_id  
6  
7     FROM customers  
8     JOIN orders  
9     ON customers.customer_id = orders.order_id  
10 )  
11 CTE_2 AS (  
12     SELECT  
13         baskets.order_id,  
14         products.*  
15  
16     FROM baskets  
17     JOIN products  
18     ON baskets.product_id = products.product_id  
19 )  
20  
21 SELECT  
22     ROUND(AVG(CTE_1.age), 1) AS 'Average Age'  
23  
24 FROM CTE_1  
25 JOIN CTE_2  
26 ON CTE_1.order_id = CTE_2.order_id  
27  
28 WHERE  
29     CTE_2.category = 'vitamins'
```

The result set shows the average age of customers who made orders in the 'vitamins' product category.

Average Age
36.4

Execution finished without errors.  
Result: 1 rows returned in 0ms

The average age of customers who made orders in the 'vitamins' product category is **36 years old**.