

Scenario:

Data Scientist at USDA (United States Department of Agriculture)

Context:

You are a Data Scientist working at the USDA. Your department has been tracking the production of various agricultural commodities across different states.

Your datasets include:

`milk_production`, `cheese_production`, `coffee_production`, `honey_production`, `yogurt_production`, and a `state_lookup` table.

The data spans multiple years and states, with varying levels of production for each commodity.

Your manager has requested that you generate insights from this data to aid in future planning and decision-making. You'll need to use SQL queries to answer the questions that come up in meetings, reports, or strategic discussions.

Objectives:

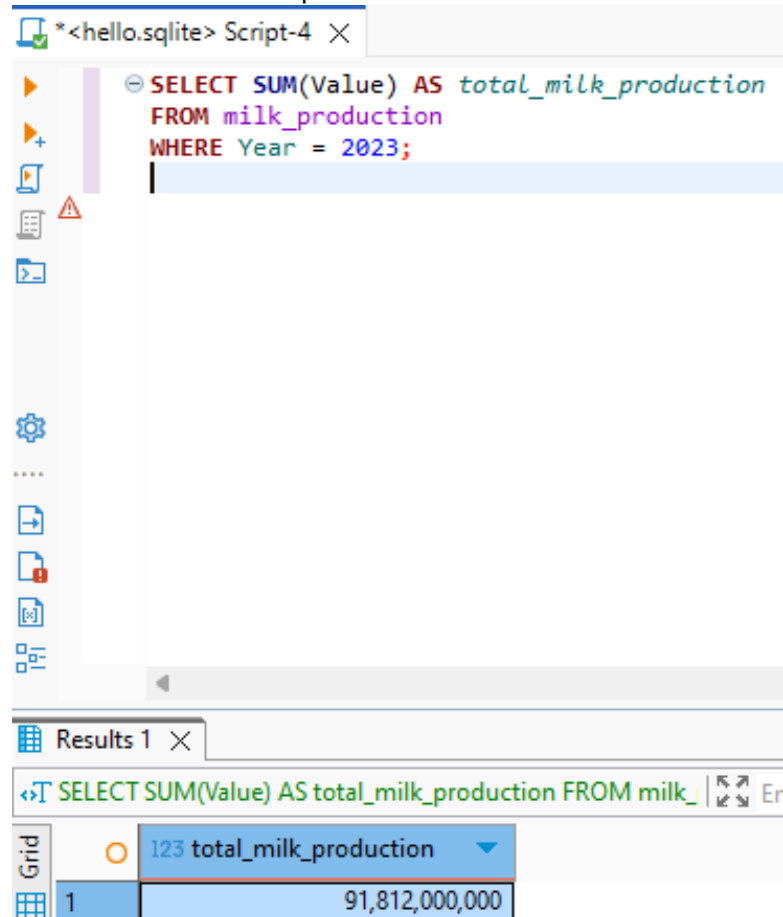
- Assess state-by-state production for each commodity.
- Identify trends or anomalies.
- Offer data-backed suggestions for areas that may need more attention.

NOTE: All answer entries are numeric and only numbers and periods. The autograder does not accept commas for the final project.

1. Question 1

Can you find out the total milk production for 2023? My manager wants this information for the yearly report.

What is the total milk production for 2023?



The screenshot shows a SQLite IDE interface. The top pane, titled '*<hello.sqlite> Script-4', contains the following SQL query:

```
SELECT SUM(Value) AS total_milk_production
FROM milk_production
WHERE Year = 2023;
```

The bottom pane, titled 'Results 1', displays the query results in a table. The table has one column, 'total_milk_production', and one row with the value 91,812,000,000.

	total_milk_production
1	91,812,000,000

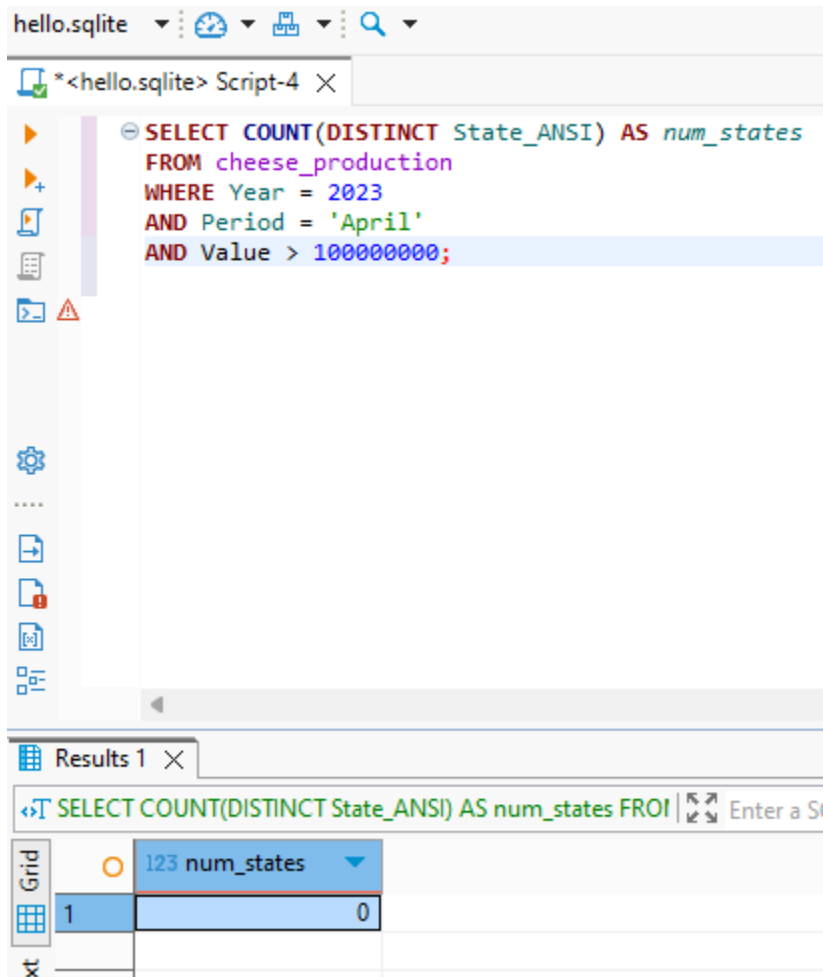
Analyst Explanation:

- SUM(Value): Calculates the **total milk production**.
- WHERE Year = 2023: Filters the records to only include production data from **2023**.

2. Question 2

Which states had cheese production greater than 100 million in April 2023? The Cheese Department wants to focus their marketing efforts there.

How many states are there?



The screenshot shows a SQLite IDE interface. The top toolbar includes icons for file operations, search, and execution. The main editor displays a SQL query: `SELECT COUNT(DISTINCT State_ANSI) AS num_states FROM cheese_production WHERE Year = 2023 AND Period = 'April' AND Value > 100000000;`. Below the editor, the 'Results 1' tab is active, showing a table with one column, 'num_states', and one row with the value '123'. The table is displayed in a grid view.

	num_states
1	123

Analyst Explanation:

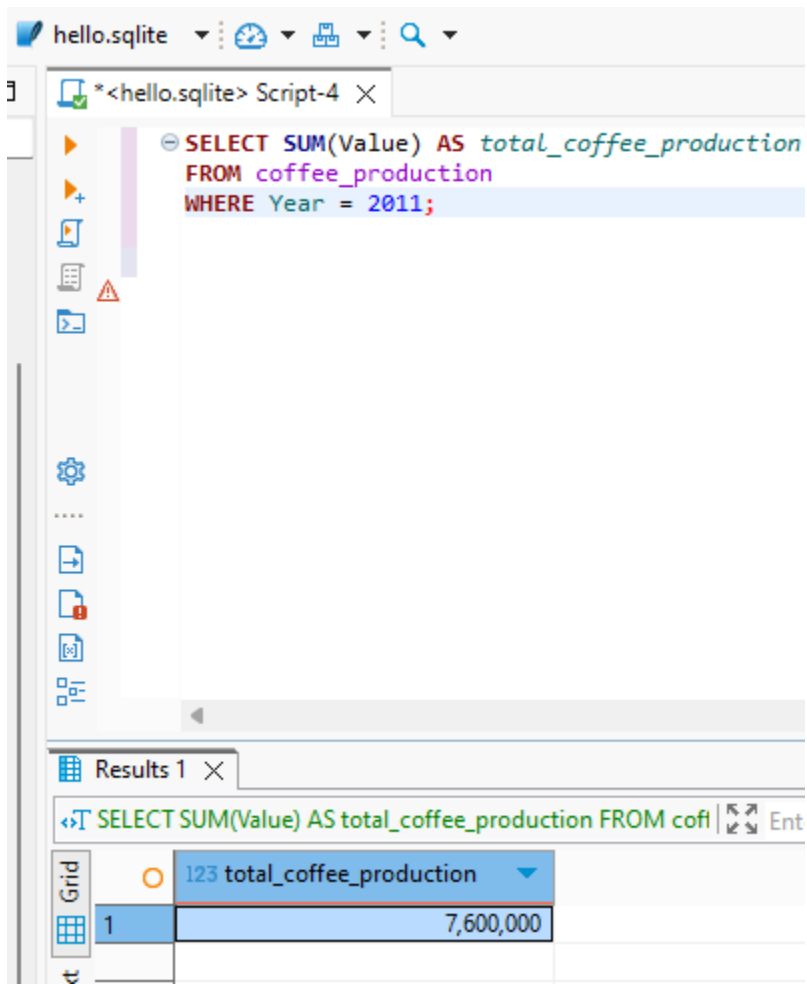
- `COUNT(DISTINCT State_ANSI)`: Counts the **number of unique states** meeting the criteria.
- `WHERE Year = 2023 AND Period = 'April'`: Filters data for **April 2023**.
- `AND Value > 100000000`: Ensures only states where cheese production **exceeded 100 million** are included.

--

3. Question 3

Your manager wants to know how coffee production has changed over the years.

What is the total value of coffee production for 2011?

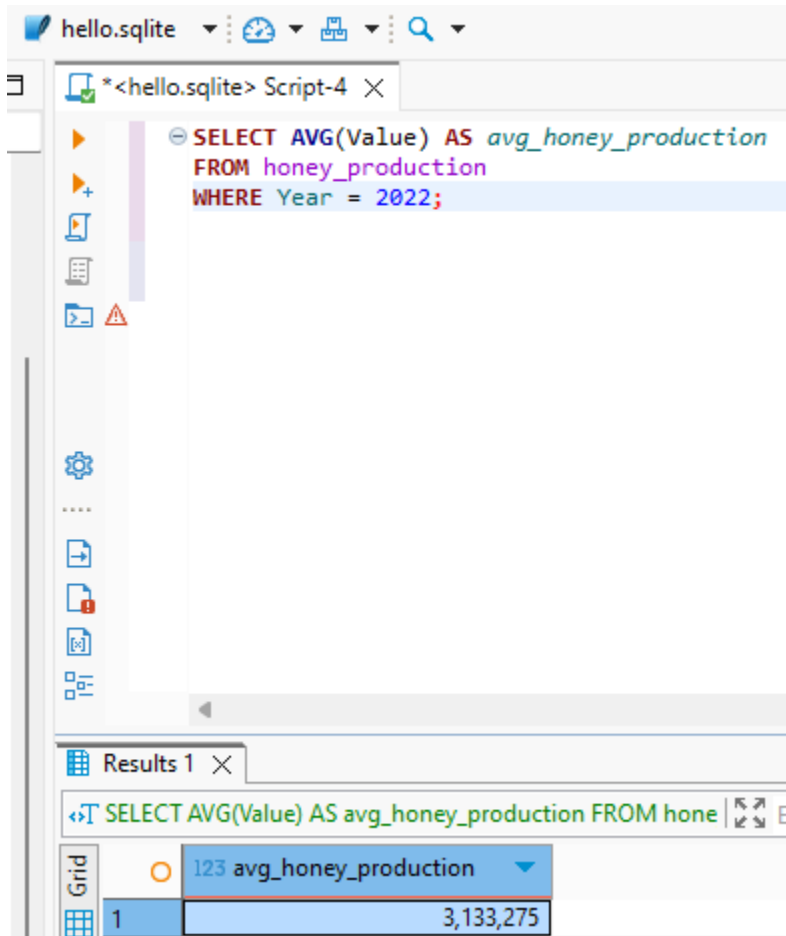


The screenshot shows a SQLite IDE interface. The top toolbar includes icons for file operations and search. The main editor window displays a SQL query: `SELECT SUM(Value) AS total_coffee_production FROM coffee_production WHERE Year = 2011;`. Below the editor, the 'Results 1' pane shows the query executed on a table named 'coffee_production'. The results are displayed in a grid format with one row and one column.

	total_coffee_production
1	7,600,000

4. Question 4

There's a meeting with the Honey Council next week. Find the average honey production for 2022 so you're prepared.



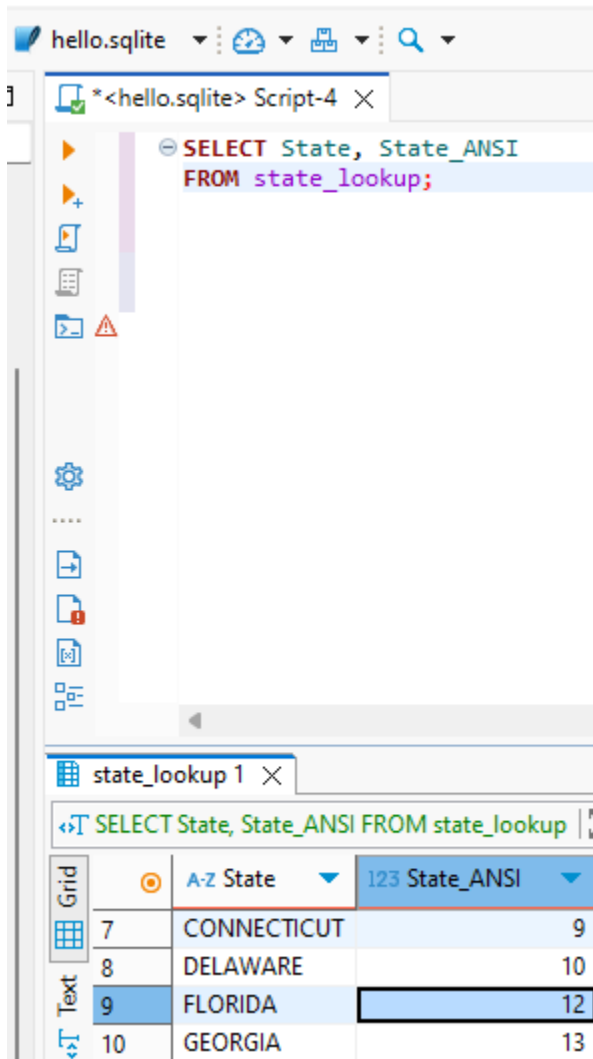
Analyst Explanation:

- `AVG(Value)`: Calculates the **average honey production**.
- `WHERE Year = 2022`: Filters records to only include **2022** data.

Question 5

The State Relations team wants a list of all states names with their corresponding ANSI codes. Can you generate that list?

What is the State_ANSI code for Florida?



The screenshot shows a SQLite IDE interface. The top pane displays a SQL script named "Script-4" with the following query:

```
SELECT State, State_ANSI
FROM state_lookup;
```

The bottom pane shows the results of the query in a table view. The table has two columns: "State" and "State_ANSI". The results are as follows:

	A-Z State	123 State_ANSI
7	CONNECTICUT	9
8	DELAWARE	10
9	FLORIDA	12
10	GEORGIA	13

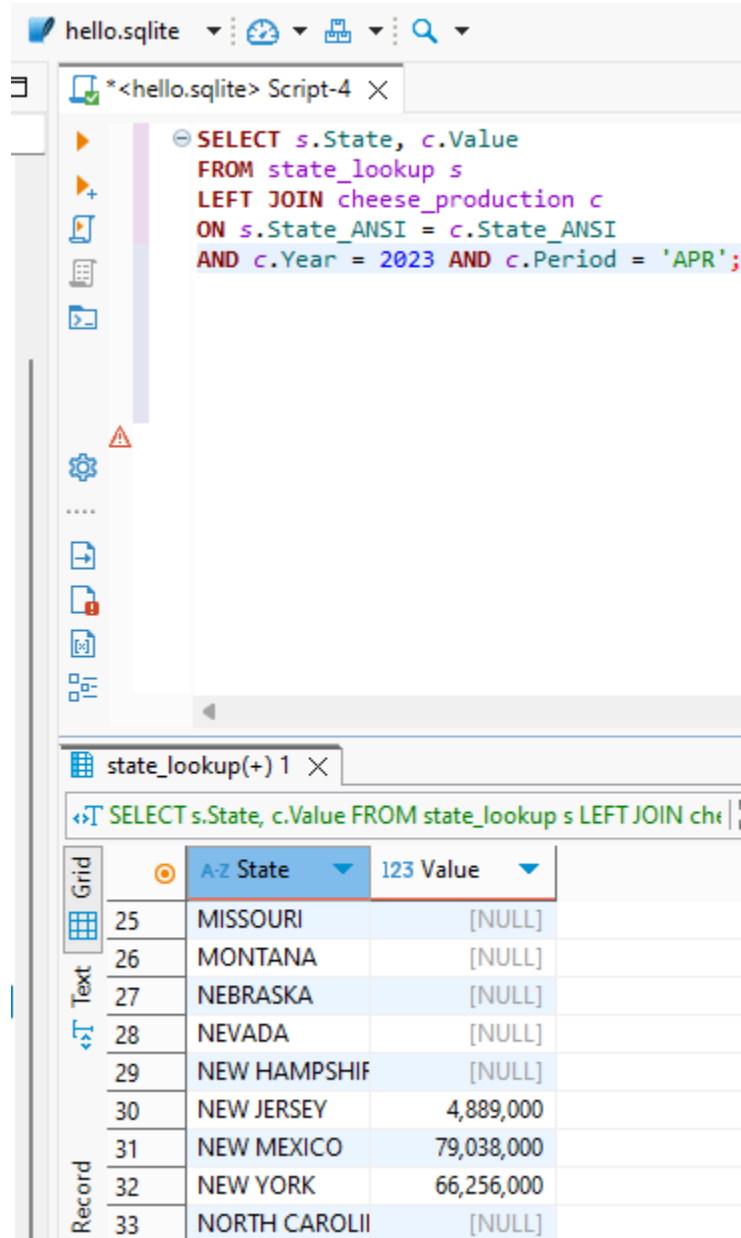
Analyst Explanation:

- **SELECT State, State_ANSI** → Selects the columns containing the state name and ANSI code.
- **FROM state_lookup** → Ensures the data is pulled from the correct reference table that stores state information.

6. Question 6

For a cross-commodity report, can you list all states with their cheese production values, even if they didn't produce any cheese in April of 2023?

What is the total for NEW JERSEY?



The screenshot shows a SQLite IDE interface. The top pane displays a SQL query in a script editor:

```
SELECT s.State, c.Value
FROM state_lookup s
LEFT JOIN cheese_production c
ON s.State_ANSI = c.State_ANSI
AND c.Year = 2023 AND c.Period = 'APR';
```

The bottom pane shows the results of the query in a table view. The table has two columns: "A-Z State" and "Value". The results are as follows:

	A-Z State	Value
25	MISSOURI	[NULL]
26	MONTANA	[NULL]
27	NEBRASKA	[NULL]
28	NEVADA	[NULL]
29	NEW HAMPSHIRE	[NULL]
30	NEW JERSEY	4,889,000
31	NEW MEXICO	79,038,000
32	NEW YORK	66,256,000
33	NORTH CAROLINA	[NULL]

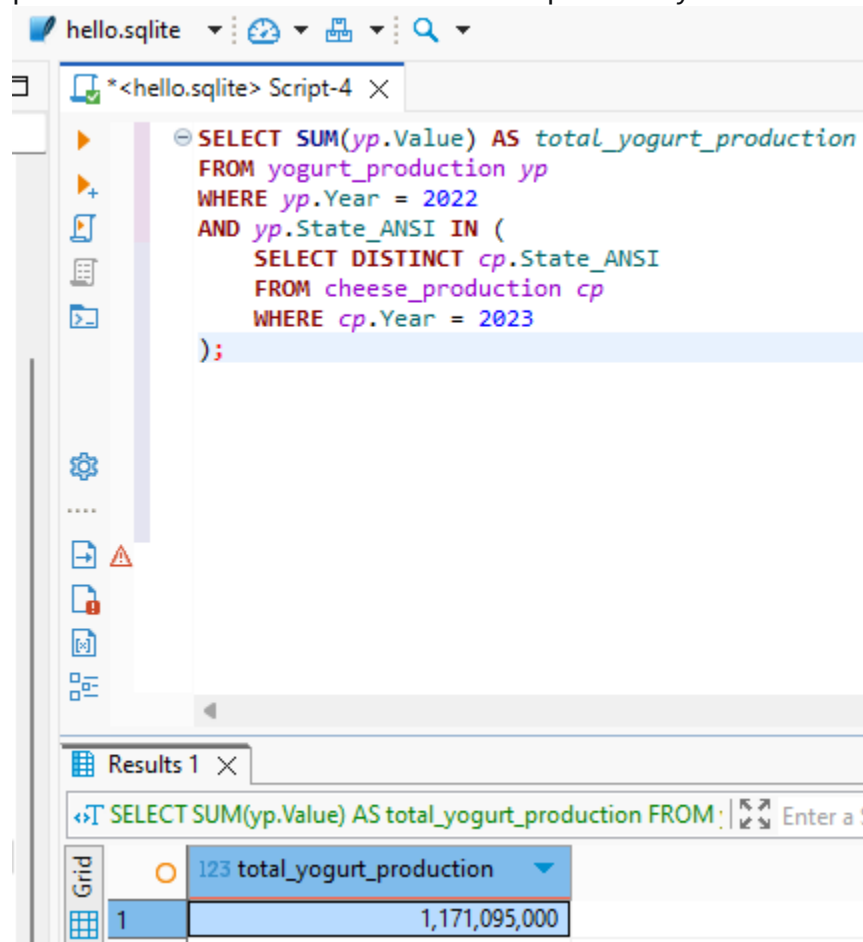
Analyst Explanation:

- `LEFT JOIN state_lookup s ON cheese_production c` → Ensures all states are included, even if they **didn't produce cheese** in April 2023.
- `AND c.Year = 2023 AND c.Period = 'APR'` → Filters only **2023 April cheese production** records.

- `s.State, c.Value` → Displays **state names** alongside **cheese production values**.

7. Question 7

Can you find the total yogurt production for states in the year 2022 which also have cheese production data from 2023? This will help the Dairy Division in their planning.



The screenshot shows a SQLite IDE interface. The top pane displays a SQL query in a script editor. The query is as follows:

```
SELECT SUM(yp.Value) AS total_yogurt_production
FROM yogurt_production yp
WHERE yp.Year = 2022
AND yp.State_ANSI IN (
  SELECT DISTINCT cp.State_ANSI
  FROM cheese_production cp
  WHERE cp.Year = 2023
);
```

The bottom pane shows the results of the query. The results are displayed in a table with one column, `total_yogurt_production`, and one row with the value `1,171,095,000`.

total_yogurt_production
1,171,095,000

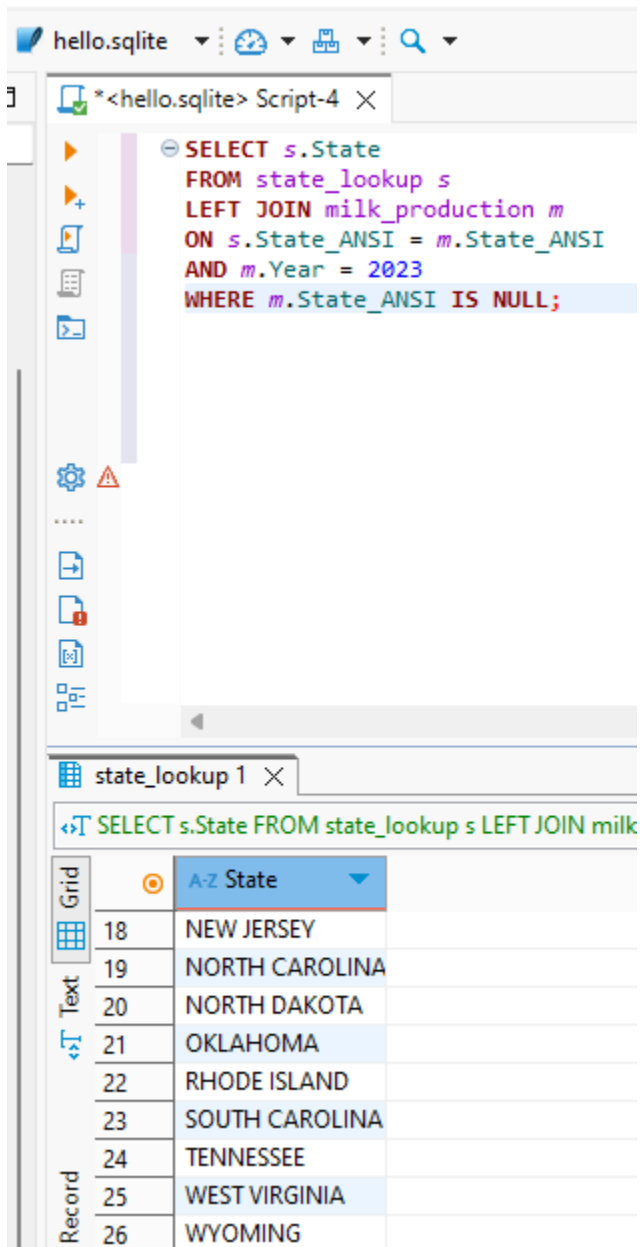
Analyst Explanation:

- `SUM(yp.Value)`: Calculates the **total yogurt production** for qualifying states.
- `WHERE yp.Year = 2022`: Filters yogurt production records to only include **2022**.
- **Subquery (IN (...))**: Retrieves **distinct** states (`State_ANSI`) that have **cheese production data from 2023**, ensuring only those states are included in the calculation.

8. Question 8

List all states from state_lookup that are missing from milk production in 2023.

How many states are there?



The screenshot shows a SQLite IDE interface. The top pane displays a SQL query in a script editor:

```
SELECT s.State
FROM state_lookup s
LEFT JOIN milk_production m
ON s.State_ANSI = m.State_ANSI
AND m.Year = 2023
WHERE m.State_ANSI IS NULL;
```

The bottom pane shows the results of the query in a table view. The table has two columns: an index and the state name. The results are as follows:

	A-Z State
18	NEW JERSEY
19	NORTH CAROLINA
20	NORTH DAKOTA
21	OKLAHOMA
22	RHODE ISLAND
23	SOUTH CAROLINA
24	TENNESSEE
25	WEST VIRGINIA
26	WYOMING

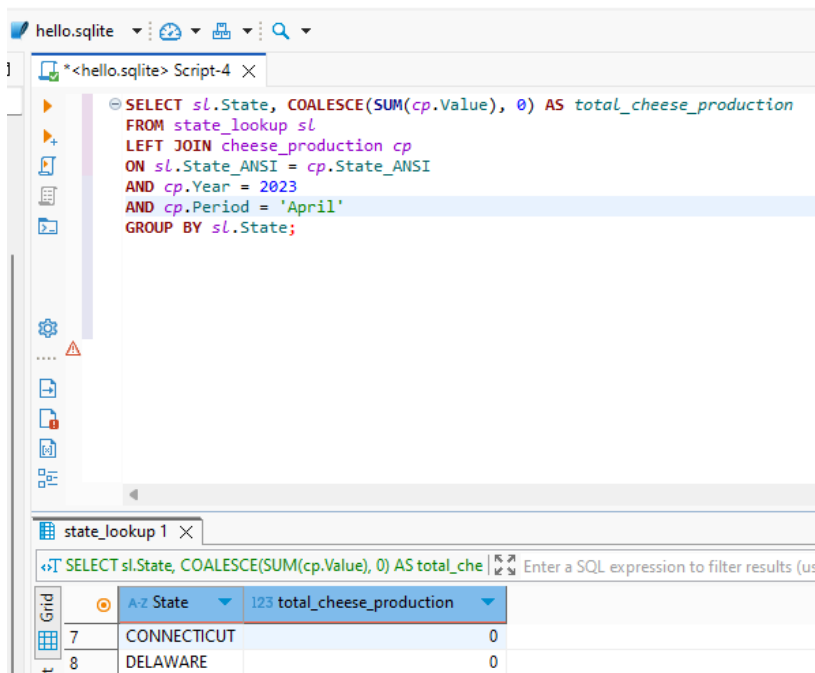
Question 9

List all states with their cheese production values, including states that didn't produce any cheese in April 2023.

Did Delaware produce any cheese in April 2023?

Analyst Explanation:

- `LEFT JOIN` ensures all states from `state_lookup` are included, even if they **didn't produce cheese** in April 2023.
- `COALESCE(SUM(cp.Value), 0)` replaces `NULL` values with 0 for states that didn't report cheese production.
- `GROUP BY sl.State` ensures each state appears only once in the results.

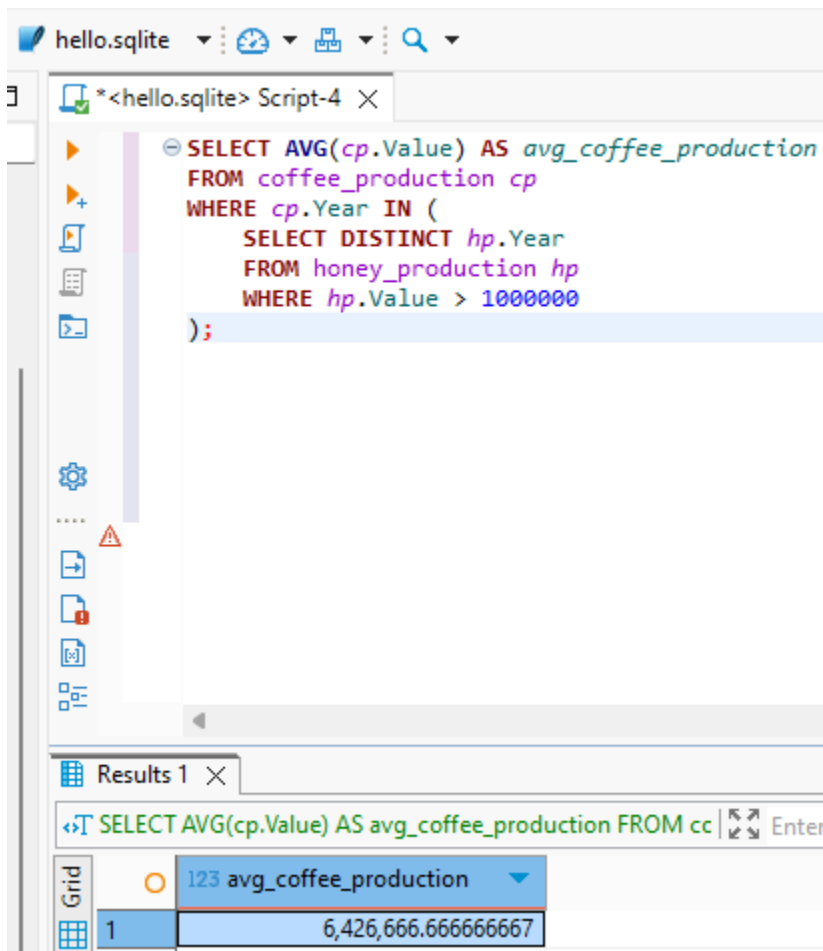


The screenshot shows a SQLite IDE window titled 'hello.sqlite'. The main editor displays a SQL query: `SELECT sl.State, COALESCE(SUM(cp.Value), 0) AS total_cheese_production FROM state_lookup sl LEFT JOIN cheese_production cp ON sl.State_ANSI = cp.State_ANSI AND cp.Year = 2023 AND cp.Period = 'April' GROUP BY sl.State;`. Below the editor, a results pane titled 'state_lookup 1' shows a table with two columns: 'A-z State' and '123 total_cheese_production'. The table contains two rows: 'CONNECTICUT' with a value of 0, and 'DELAWARE' with a value of 0.

A-z State	123 total_cheese_production
CONNECTICUT	0
DELAWARE	0

Question 10

Find the average coffee production for all years where the honey production exceeded 1 million.



The screenshot shows a SQLite IDE window titled 'hello.sqlite'. The main editor displays a SQL query: `SELECT AVG(cp.Value) AS avg_coffee_production FROM coffee_production cp WHERE cp.Year IN (SELECT DISTINCT hp.Year FROM honey_production hp WHERE hp.Value > 1000000);`. Below the editor, the 'Results 1' tab is active, showing a single row of results in a grid. The column is labeled 'avg_coffee_production' and the value is '6,426,666.666666667'.

```
SELECT AVG(cp.Value) AS avg_coffee_production
FROM coffee_production cp
WHERE cp.Year IN (
    SELECT DISTINCT hp.Year
    FROM honey_production hp
    WHERE hp.Value > 1000000
);
```

	avg_coffee_production
1	6,426,666.666666667

Analyst Explanation:

- `AVG(cp.Value)`: Computes the **average coffee production**.
- `WHERE cp.Year IN (...)`: Filters coffee production to only include years where honey production exceeded **1 million**.
- **Subquery** (`IN (...)`): Retrieves **distinct years (Year)** from `honey_production` where `Value > 1000000`.