



Text to SQL Project Microsoft Copilot (Precise Version)

Jackson O'Brien
3/22/2024

Grading Criteria

- Output produced by LLM vs my result
- The CAP SQL Script will be given to the LLM
- For every time LLM has to be corrected it loses a point
- 10/10 if the LLM produces correct output first try
- Question will be sent to LLM the same way user would read the question
- Partial credit will be given if problem is partly correct



It looks like you're setting up a database schema in SQL! Here's what each table is for:

- `People` : This table stores information about people, including their ID, name, city, and date of birth.
- `Customers` : This table stores customer-specific information, such as payment terms and discount percentage. Each customer is also a person, so it references the `People` table.
- `Agents` : Similar to `Customers`, this table stores agent-specific information. Each agent is also a person, so it references the `People` table.
- `Products` : This table stores information about the products, including product ID, name, city, quantity on hand, and price in USD.
- `Orders` : This table stores information about the orders, including order number, date ordered, customer ID, agent ID, product ID, quantity ordered, and total in USD. It references the `Customers`, `Agents`, and `Products` tables.

If you need help with SQL queries or have any questions about these statements, feel free to ask! 😊

Question 1

My Query:

```
select
  city,
  count(distinct prodId),
  rank() over (order by count(distinct prodId) desc)
from Products
group by city
order by count desc;
```

My Output:

	city text	num_products bigint	rank bigint
1	Newark	3	1
2	Dallas	2	2
3	Duluth	2	2
4	Beijing	1	4
5	Montreal	1	4

LLM Query:

You

Display the cities that makes the most different kinds of products. Experiment with the rank() function.

Copilot

Sure, you can use the `RANK()` function in SQL to rank the cities based on the number of different kinds of products they make. Here's an example:

```
SQL
SELECT city, COUNT(DISTINCT name) AS num_products,
       RANK() OVER (ORDER BY COUNT(DISTINCT name) DESC) as rank
FROM Products
GROUP BY city
ORDER BY num_products DESC;
```

AI-generated code. Review and use carefully. [More info on FAQ](#)

LLM Output:

	city text	count bigint	rank bigint
1	Newark	3	1
2	Dallas	2	2
3	Duluth	2	2
4	Beijing	1	4
5	Montreal	1	4

- Grade: 10/10
- No fixing was required, the only difference was the LLM changing the “num_products” alias to “count”.

Question 2

My Query:

```
select Products.name
from Products
where priceUSD < (
  select avg(priceUSD) * 0.01
  from Products)
order by Products.name asc;
```

My Output:

	name text	🔒
1	Denis Wick Valve Oil	
2	Flux Capacitor	
3	HAL 9000 memory chip	
4	LCARS module	

LLM Query:

SQL

```
WITH AvgPrice AS (
  SELECT AVG(priceUSD) AS avg_price
  FROM Products
)
SELECT name
FROM Products
WHERE priceUSD < (SELECT avg_price FROM AvgPrice) / 100
ORDER BY name ASC;
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

LLM Output:

	name text	🔒
1	Denis Wick Valve Oil	
2	Flux Capacitor	
3	HAL 9000 memory chip	
4	LCARS module	




- Grade: 10/10
- No fixing required, the output is the exact same.
- Instead of multiplying by 0.01 it instead divided by 100.
- It used aliases for the priceUSD

Question 3

My Query:

```
select
  People.lastName,
  Orders.prodId,
  Orders.totalUSD
from Orders
join Customers
on Orders.custId = Customers.pid
join People
on Customers.pid = People.pid
where extract(month from Orders.dateOrdered) = 3
order by Orders.totalUSD asc;
```

My Output:

	lastname 	prodid 	totalusd 
	text	character	numeric (12,2)
1	Ferguson	p05	6550.98
2	Ferguson	p06	31236.48
3	Armstrong	p01	56671.55




LLM Query:

SQL

```
SELECT P.lastName, O.prodId, O.totalUSD
FROM Orders O
JOIN Customers C ON O.custId = C.pid
JOIN People P ON C.pid = P.pid
WHERE EXTRACT(MONTH FROM O.dateOrdered) = 3
ORDER BY O.totalUSD ASC;
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

LLM Output:

	lastname 	prodid 	totalusd 
	text	character	numeric (12,2)
1	Ferguson	p05	6550.98
2	Ferguson	p06	31236.48
3	Armstrong	p01	56671.55

- Grade: 10/10
- No fixing was required, the only difference was using aliases the aliases O & P.

Question 4

My Query:

```
select
  People.lastName,
  coalesce(sum(Orders.totalUSD), 0) as total
from People
join Customers
on People.pid = Customers.pid
left join Orders
on Customers.pid = Orders.custId
group by People.lastName
order by People.lastName desc;
```

My Output:

	lastname text	total numeric
1	Severinsen	142768.32
2	Robinson	0
3	Morrison	109053.62
4	Ferguson	146525.21
5	Armstrong	72442.75

LLM Query:

SQL

```
SELECT P.lastName, COALESCE(SUM(O.totalUSD), 0) AS total_ordered
FROM Customers C
JOIN People P ON C.pid = P.pid
LEFT JOIN Orders O ON O.custId = C.pid
GROUP BY P.lastName
ORDER BY P.lastName DESC;
```

AI-generated code. Review and use carefully. [More info on FAQ](#)

LLM Output:

	lastname text	total_ordered numeric
1	Severinsen	142768.32
2	Robinson	0
3	Morrison	109053.62
4	Ferguson	146525.21
5	Armstrong	72442.75

- Grade: 10/10
- No fixing was required, only difference being using aliases again and changing "total" to "total_ordered"

Question 5

My Query:

```
select
  People.firstName as CustomerFirstName,
  People.lastName as CustomerLastName,
  agentPeople.firstName as AgentFirstName,
  agentPeople.lastName as AgentLastName,
  Products.name as Product
from Orders
join Customers on
Orders.custId = Customers.pid
join People
on Customers.pid = People.pid
join products
on Orders.prodId = Products.prodId
join Agents
on Orders.agentId = Agents.pid
join People as agentPeople
on Agents.pid = agentPeople.pid
where agentPeople.homeCity = 'Chilliwack'
```

My Output:

	customerfirstname text	customerlastname text	agentfirstname text	agentlastname text	product text
1	Maynard	Ferguson	Bria	Skonberg	Heisenberg Compensator
2	Maynard	Ferguson	Bria	Skonberg	Universal Translator
3	Maynard	Ferguson	Bria	Skonberg	Denis Wick Valve Oil
4	Louis	Armstrong	Bria	Skonberg	Heisenberg Compensator

LLM Query:

SQL

```
SELECT PCus.firstName || ' ' || PCus.lastName AS Customer_Name,
       Pro.name AS Product_Name,
       PAG.firstName || ' ' || PAG.lastName AS Agent_Name
FROM Orders O
JOIN Customers Cus ON O.custId = Cus.pid
JOIN People PCus ON Cus.pid = PCus.pid
JOIN Products Pro ON O.prodId = Pro.prodId
JOIN Agents Ag ON O.agentId = Ag.pid
JOIN People PAG ON Ag.pid = PAG.pid
WHERE PAG.homeCity = 'Chilliwack';
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

LLM Output:

	customer_name text	product_name text	agent_name text
1	Maynard Ferguson	Heisenberg Compensator	Bria Skonberg
2	Maynard Ferguson	Universal Translator	Bria Skonberg
3	Maynard Ferguson	Denis Wick Valve Oil	Bria Skonberg
4	Louis Armstrong	Heisenberg Compensator	Bria Skonberg

- Grade: 8/10
- Two prompts were required to produce a right output. The first prompt was because it tried to use Cus.firstName, Cus doesn't exist so it produced an error. It however didn't print the first and last name of the customer or agent, so I asked it to do that and then it merged them and put it in one variable after this prompt.

Question 6

My Query:

```
select
  Products.priceUSD,
  Orders.quantityOrdered,
  round(Products.priceUSD * Orders.quantityOrdered, 2) as calculatedUSD,
  Orders.totalUSD
from Orders
join Products
on Orders.prodId = products.prodId
where round(Products.priceUSD * Orders.quantityOrdered, 2) != Orders.totalUSD;
```

My Output:

	priceusd numeric (10,2)	quantityordered integer	calculatedusd numeric	totalusd (12,2)
1	67.76	1100	74536.00	58794.00
2	65.02	1200	78024.00	76096.81
3	16.61	1000	16610.00	15771.20
4	67.76	1000	67760.00	60977.22
5	65.02	500	32510.00	25643.98
6	17.01	600	10206.00	8050.49
7	51.50	400	20600.00	16249.28
8	1.00	600	600.00	585.18
9	67.76	1000	67760.00	66086.33
10	88.00	450	39600.00	31236.48
11	16.61	500	8305.00	6550.98
12	67.76	880	59628.80	56671.55
13	1.00	888	888.00	799.11
14	65.02	808	52536.16	47277.29

LLM Query:

SQL

```
SELECT O.orderNum,
       O.totalUSD AS recorded_total,
       ROUND((O.quantityOrdered * P.priceUSD), 2) AS calculated_total
FROM Orders O
JOIN Products P ON O.prodId = P.prodId
WHERE O.totalUSD <> ROUND((O.quantityOrdered * P.priceUSD), 2);
```

AI-generated code. Review and use carefully. [More info on FAQ](#).

LLM Output:

	ordernum [PK] integer	recorded_total numeric (12,2)	calculated_total numeric
1	1011	58794.00	74536.00
2	1012	76096.81	78024.00
3	1015	15771.20	16610.00
4	1016	60977.22	67760.00
5	1017	25643.98	32510.00
6	1018	8050.49	10206.00
7	1019	16249.28	20600.00
8	1020	585.18	600.00
9	1021	66086.33	67760.00
10	1022	31236.48	39600.00
11	1023	6550.98	8305.00
12	1024	56671.55	59628.80
13	1025	799.11	888.00
14	1026	47277.29	52536.16

- Grade: 9/10
- It provided too much information for the result. I asked it to only provide relevant information and reduce the amount of columns, and then it produced a good final result similar to my output.

Question 7

My Query:

```
select
  firstName,
  lastName
from People
join Agents
on People.pid = Agents.pid
join Customers
on People.pid = Customers.pid;
```

My Output:

	firstname 	lastname 
1	Louis	Armstrong


LLM Query:

SQL

```
SELECT PCus.firstName, PCus.lastName
FROM Customers Cus
JOIN Agents Ag ON Cus.pid = Ag.pid
JOIN People PCus ON Cus.pid = PCus.pid;
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

LLM Output:

	firstname 	lastname 
1	Louis	Armstrong

- Grade: 10/10
- No fixing was required, code was identical except for ai using "Pcus" and aliases again.



Conclusion

- Microsoft Copilot is a very versatile LLM that can handle SQL queries surprisingly well.
- The code is easy to read and the LLM will provide information about how the code works and its thought process behind creating it.
- Midway through giving prompts, it would say session ended
- Would have to give all the SQL script information again
- Would forget everything that happened in the last session
- Used many aliases in almost every prompt, would use full caps for the select keywords
- Had trouble understanding what information to output in the SELECT statement a couple times.



Copilot

Of course, I'm happy to start over. What can I assist you with now?



References

Microsoft Copilot. (2024). Personal communication.