SQL MURDER MYSTERY

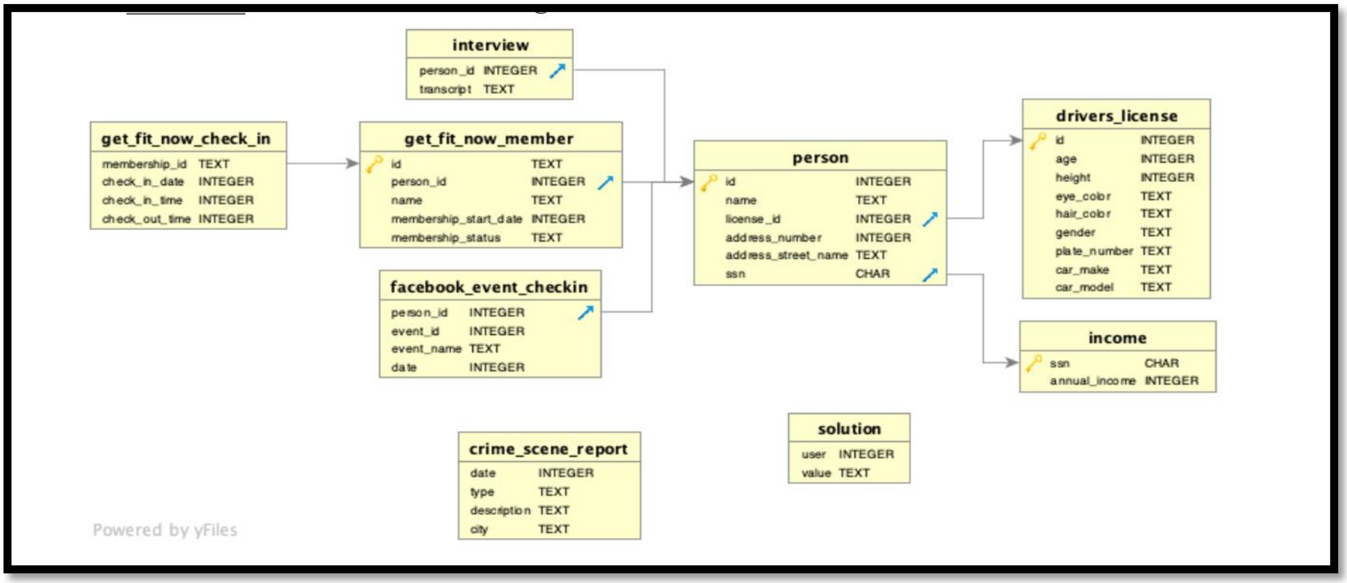
Analyst- Sagar Kalauni
Southern Illinois University Edwardsville
Sagarkalauni7@gmail.com
Sagarkalauni.github.io

Background

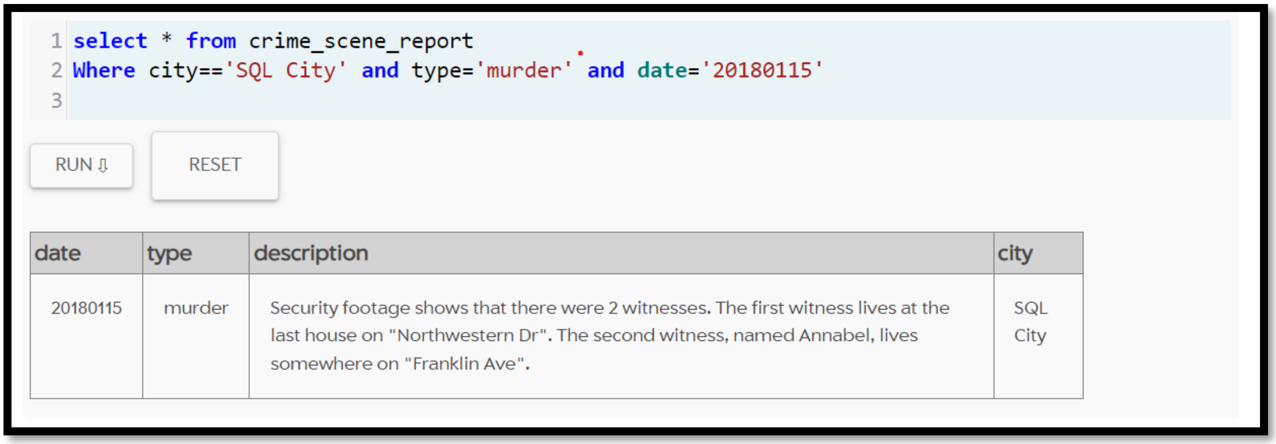
A crime has taken place, and the detective needs your help. The detective gave you the crime scene report, but you somehow lost it. You vaguely remember that the crime was a **murder** that occurred sometime on **Jan.15, 2018** and that it took place in **SQL City**. Start by retrieving the corresponding crime scene report from the police department's database.

INVESTIGATION

As a data analyst, it is essential to familiarize yourself with the data before commencing any analysis. Understanding the metadata significantly enhances the efficiency and accuracy of our analysis. Let's delve into the metadata by examining the schema diagram.



I remember that a murder occurred in SQL City on January 15, 2018. I need to retrieve the description of that murder from the crime_scene_report table. Below is the code and output for that:



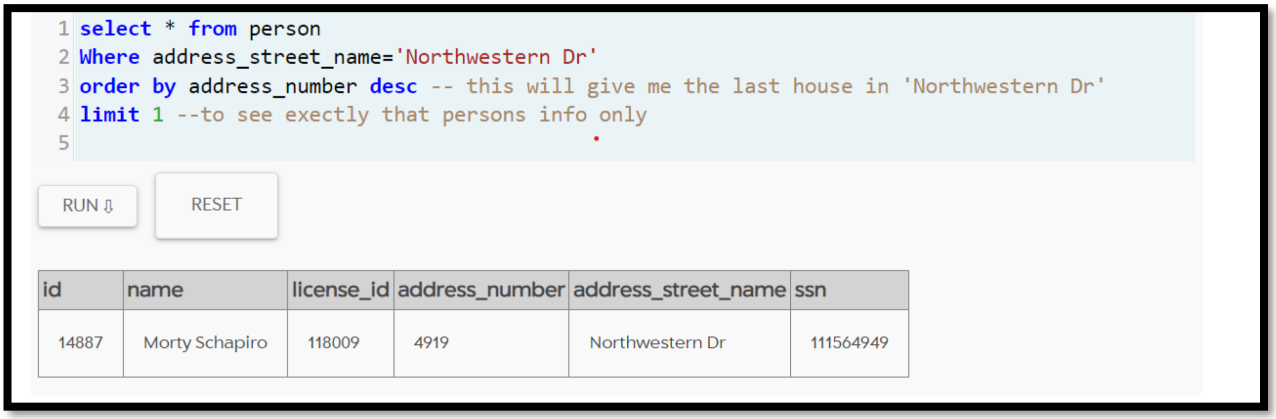
```sql
select * from crime_scene_report
Where city=='SQL City' and type='murder' and date='20180115'
```

date	type	description	city
20180115	murder	Security footage shows that there were 2 witnesses. The first witness lives at the last house on "Northwestern Dr". The second witness, named Annabel, lives somewhere on "Franklin Ave".	SQL City

Based on the output indicating two witnesses, the next step in the investigation is to gather complete information about these witnesses. Below, we'll search for the code to retrieve the full details of the witnesses.

FOR 1ST WITNESS

We're searching for the first witness who resides at the last house on Northwestern Dr. Although we lack the name and gender, we do know the location of their residence. By querying individuals residing on Northwestern Dr. and selecting the one with the highest address number, we can identify this witness. The code to retrieve that information and the output is given below.



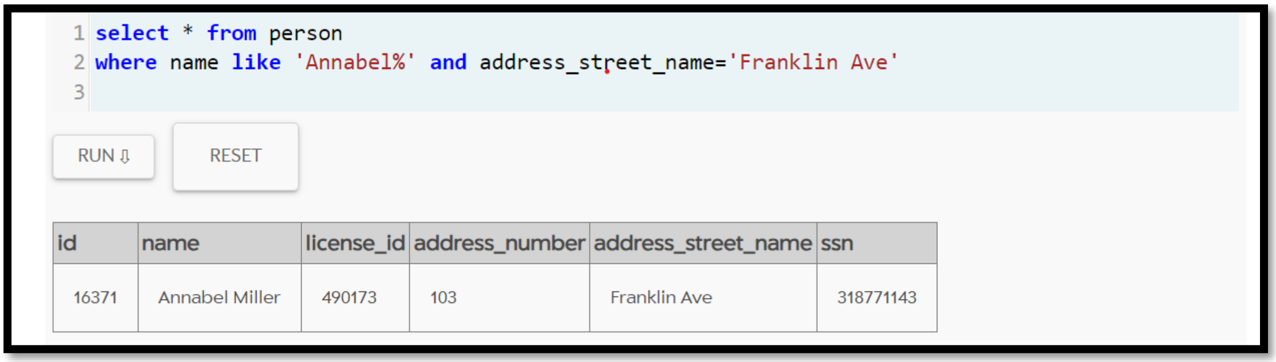
```sql
select * from person
Where address_street_name='Northwestern Dr'
order by address_number desc -- this will give me the last house in 'Northwestern Dr'
limit 1 --to see exectly that persons info only
```

id	name	license_id	address_number	address_street_name	ssn
14887	Morty Schapiro	118009	4919	Northwestern Dr	111564949

Having identified the first witness and obtained their complete information, our next step is to gather details about the second witness.

FOR 2ND WITNESS

We have information about the second witness, named 'Annabel', including name and residence on Franklin Ave. Our next step is to retrieve complete information. The information about second witness is retrieved using the following code.

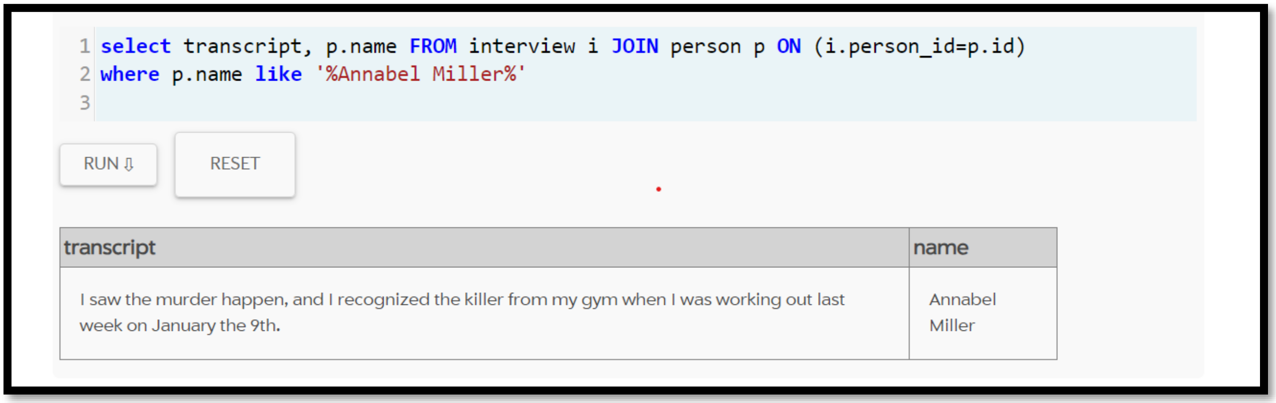


```sql
select * from person
where name like 'Annabel%' and address_street_name='Franklin Ave'
```

id	name	license_id	address_number	address_street_name	ssn
16371	Annabel Miller	490173	103	Franklin Ave	318771143

As an analyst, our next task is to obtain the interview transcripts of both witnesses regarding the witnessed murder. We have their names and IDs stored in the Person table. To retrieve this information, we will use the JOIN command to connect the Person table with the Interview table, linking their IDs. This approach allows us to gather the interview transcripts for both witnesses efficiently. The code to get the interview transcript for both witness is below.

For 1st Witness

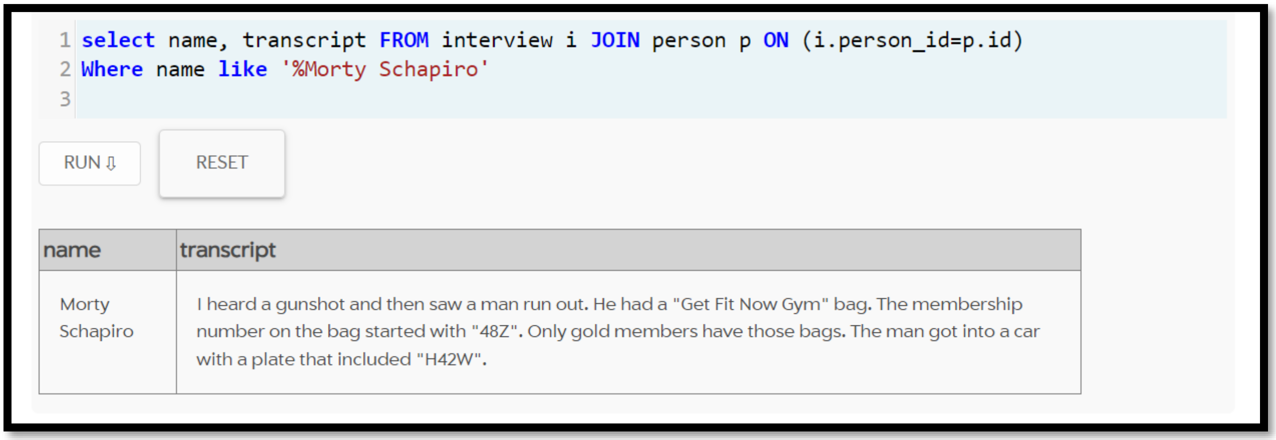


```sql
select transcript, p.name FROM interview i JOIN person p ON (i.person_id=p.id)
where p.name like '%Annabel Miller%'
```

transcript	name
I saw the murder happen, and I recognized the killer from my gym when I was working out last week on January the 9th.	Annabel Miller

On Interview 1st witness says: - "*I saw the murder happen, and I recognized the killer from my gym when I was working out last week on January the 9th.*" The information we got about murderer from is 1st witness is that Actual murder data is Jan 9th and murder happens close to gym in SQL city.

For 2nd Witness

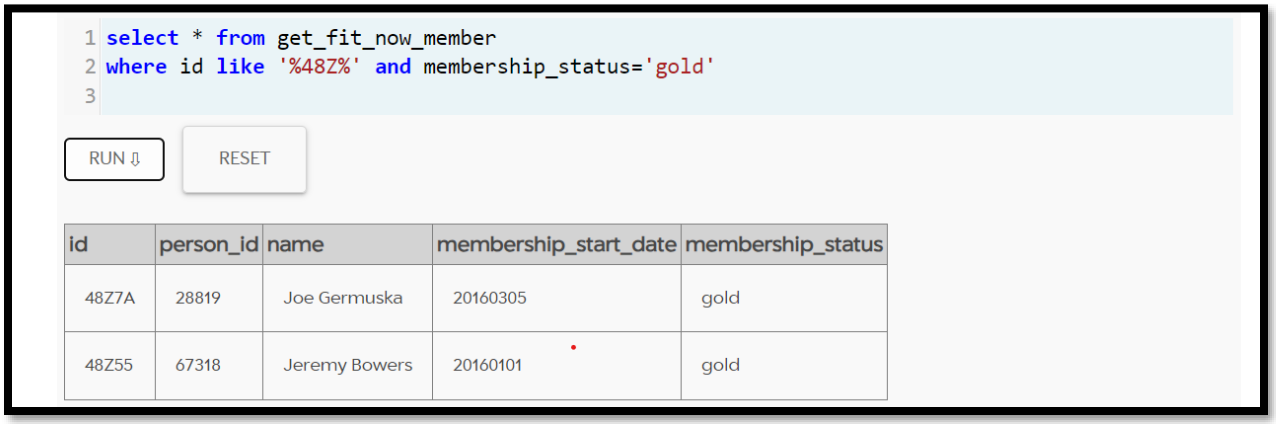


```sql
select name, transcript FROM interview i JOIN person p ON (i.person_id=p.id)
Where name like '%Morty Schapiro'

```

name	transcript
Morty Schapiro	I heard a gunshot and then saw a man run out. He had a "Get Fit Now Gym" bag. The membership number on the bag started with "48Z". Only gold members have those bags. The man got into a car with a plate that included "H42W".

On Interview 2nd witness says: - *I heard a gunshot and then saw a man run out. He had a "Get Fit Now Gym" bag. The membership number on the bag started with "48Z". Only gold members have those bags. The man got into a car with a plate that included "H42W".*

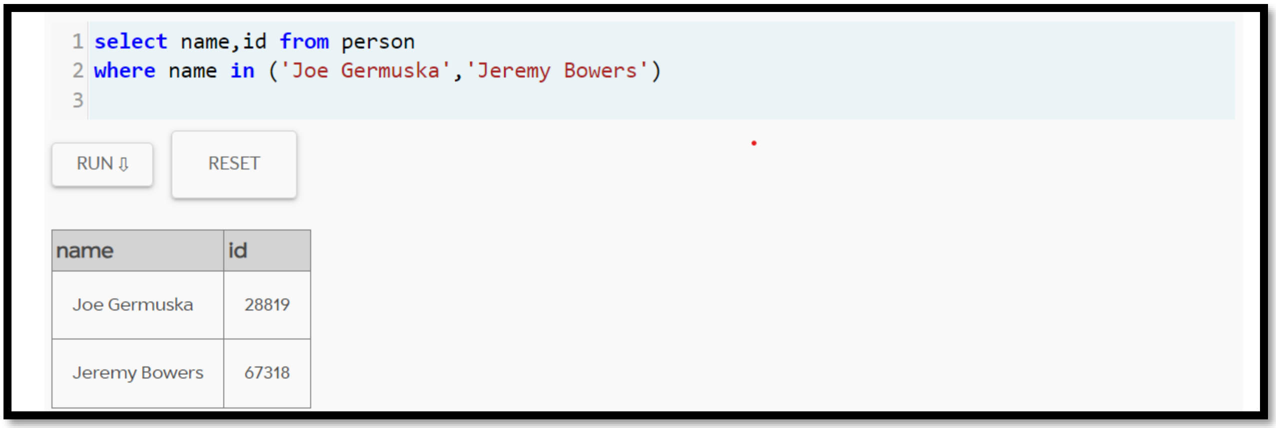
To identify potential suspects, we will focus on individuals who meet specific criteria provided by the second witness regarding the murderer. Firstly, we'll search for gold members of 'Get Fit Now Gym' whose membership ID starts with "48Z". Below, you'll find the code and output for this investigation.



```sql
select * from get_fit_now_member
where id like '%48Z%' and membership_status='gold'

```

id	person_id	name	membership_start_date	membership_status
48Z7A	28819	Joe Germuska	20160305	gold
48Z55	67318	Jeremy Bowers	20160101	gold

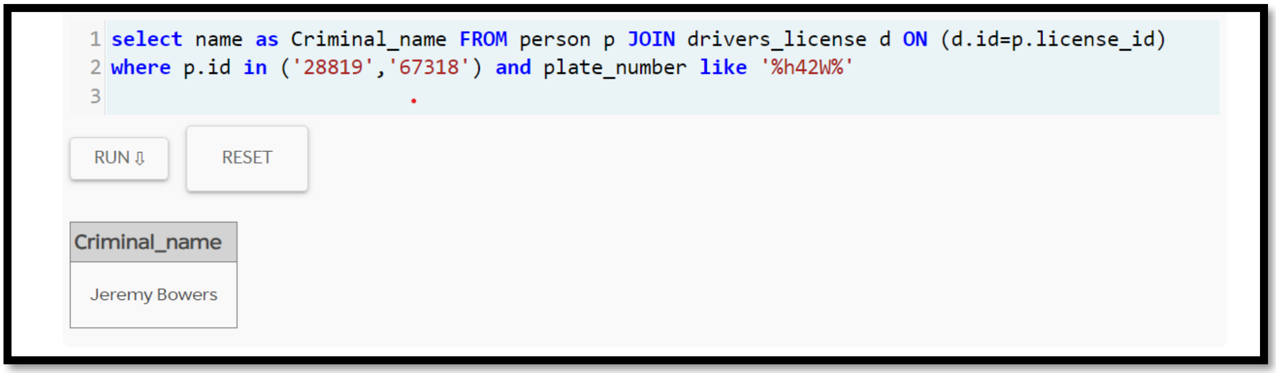
Our investigation now narrows down to Joe Germuska and Jeremy Bowers as possible suspects in the murder case. To determine the actual culprit between these two individuals, we need to inspect their car number plates. This requires combining data from the drivers_license and person tables using a join command. However, we must first obtain their person IDs, as the IDs seen in the output belong to gym memberships, not person IDs. We can retrieve their person IDs by referring to the person table. Here is the code for that



```sql
select name,id from person
where name in ('Joe Germuska','Jeremy Bowers')

```

name	id
Joe Germuska	28819
Jeremy Bowers	67318

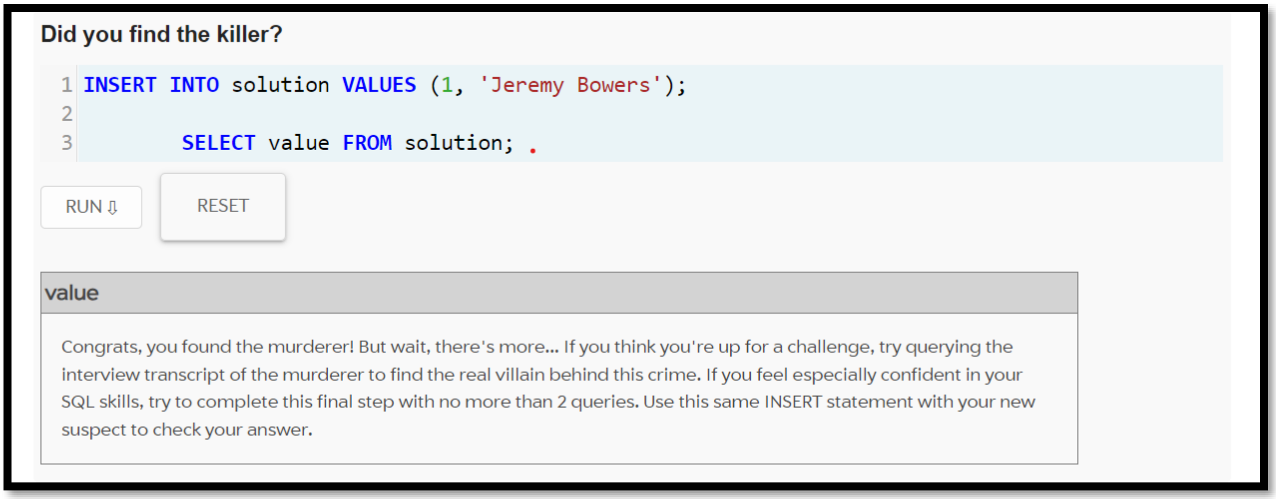
With the IDs of both suspects in hand, our next step is to confirm the identity of the murderer by examining their car number plates. The code for that is below.



```sql
select name as Criminal_name FROM person p JOIN drivers_license d ON (d.id=p.license_id)
where p.id in ('28819','67318') and plate_number like '%h42W%'

```

Criminal_name
Jeremy Bowers

Now that we have identified the murderer as Jeremy Bowers, let's verify whether our conclusion is accurate. The code and output for that is:

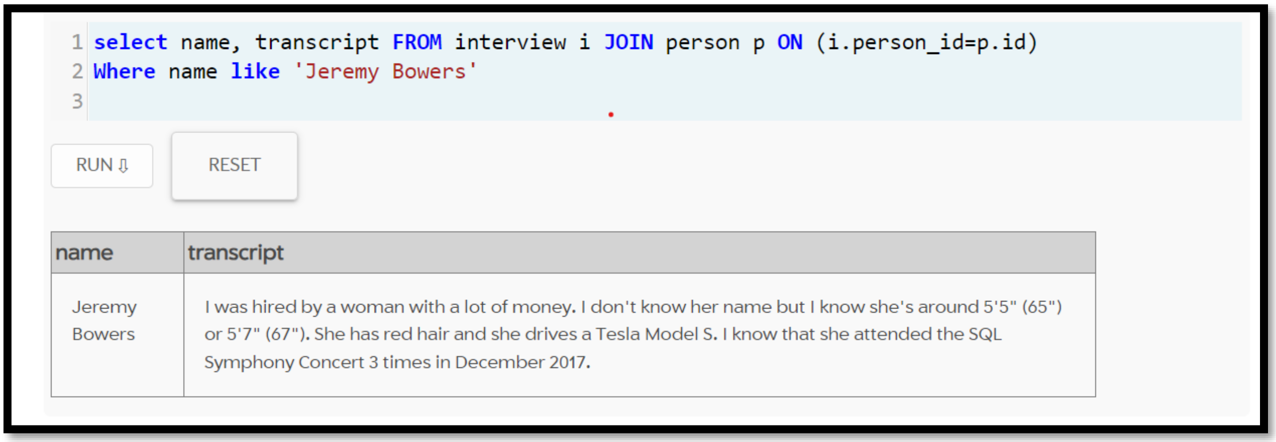


Did you find the killer?

```sql
INSERT INTO solution VALUES (1, 'Jeremy Bowers');

        SELECT value FROM solution;
```

value
Congrats, you found the murderer! But wait, there's more... If you think you're up for a challenge, try querying the interview transcript of the murderer to find the real villain behind this crime. If you feel especially confident in your SQL skills, try to complete this final step with no more than 2 queries. Use this same INSERT statement with your new suspect to check your answer.

CONGRATULATIONS! We have successfully identified the murderer. However, there's a twist in the story. As an analyst, let's examine the interview transcript of the murderer to uncover what Jeremy Bowers has to say about the murder.



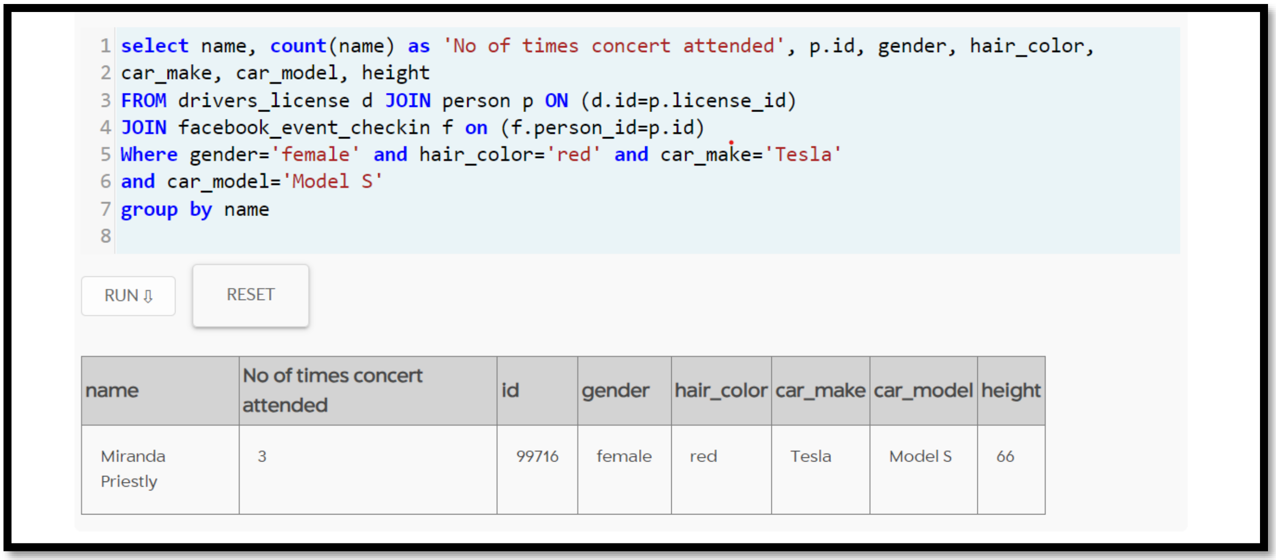
```sql
select name, transcript FROM interview i JOIN person p ON (i.person_id=p.id)
Where name like 'Jeremy Bowers'

```

name	transcript
Jeremy Bowers	I was hired by a woman with a lot of money. I don't know her name but I know she's around 5'5" (65") or 5'7" (67"). She has red hair and she drives a Tesla Model S. I know that she attended the SQL Symphony Concert 3 times in December 2017.

Murderer Jeremy Bowers say:" *I was hired by a woman with a lot of money. I don't know her name but I know she's around 5'5" (65") or 5'7" (67"). She has red hair and she drives a Tesla Model S. I know that she attended the SQL Symphony Concert 3 times in December 2017.*"

Here's the situation: Jeremy Bowers turned out to be just a hired killer, not the mastermind behind the crime. Now, we're on the hunt for the real villain. We have some clues about her: she's a woman with red hair, around 65 to 67 units tall, and drives a Tesla Model S. Also, she went to the SQL Symphony Concert three times in December 2017.

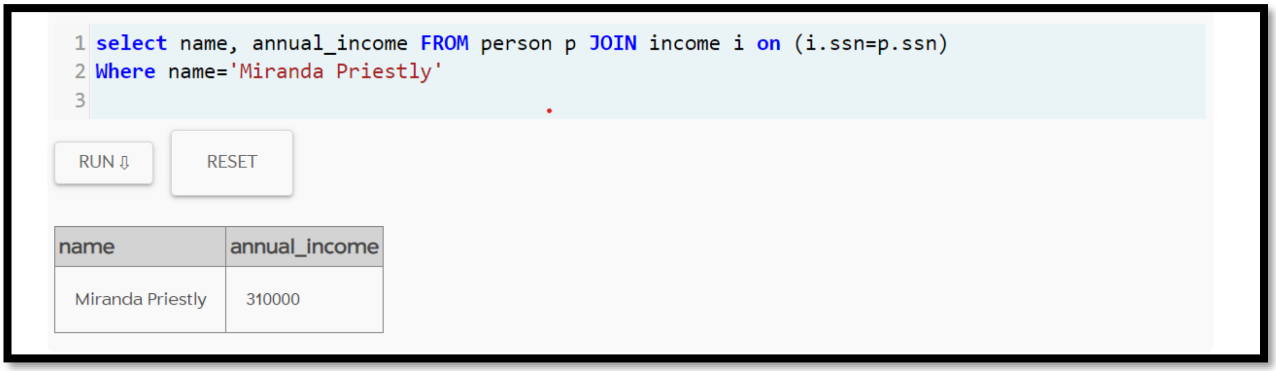
From what we learned in the murderer's interview, it seems focusing on finding women who attended the concert three times in December 2017 should be our first priority. We can also look into her car details and hair color by using different pieces of information we have and joining them together. The code and output for this is below:



```sql
select name, count(name) as 'No of times concert attended', p.id, gender, hair_color,
car_make, car_model, height
FROM drivers_license d JOIN person p ON (d.id=p.license_id)
JOIN facebook_event_checkin f on (f.person_id=p.id)
Where gender='female' and hair_color='red' and car_make='Tesla'
and car_model='Model S'
group by name
```

name	No of times concert attended	id	gender	hair_color	car_make	car_model	height
Miranda Priestly	3	99716	female	red	Tesla	Model S	66

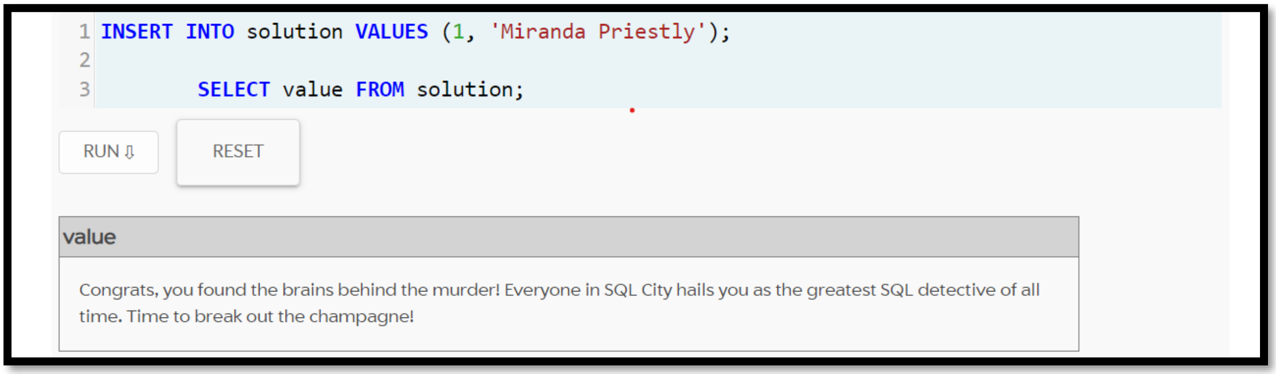
It seems we've identified the true villain behind the crime. However, before confirming her as the final culprit, we need to verify some details. According to the murderer's interview, the main villain should be a wealthy woman. Therefore, our next step is to examine her income and verify if she fits this description.



```sql
select name, annual_income FROM person p JOIN income i on (i.ssn=p.ssn)
Where name='Miranda Priestly'
```

name	annual_income
Miranda Priestly	310000

This gives more proof of she might be the real villain behind the crime. Let’s confirm using the code below.



```sql
INSERT INTO solution VALUES (1, 'Miranda Priestly');

        SELECT value FROM solution;
```

value
Congrats, you found the brains behind the murder! Everyone in SQL City hails you as the greatest SQL detective of all time. Time to break out the champagne!

CONGRATULATIONS We did it. We found the murderer and the mastermind behind it. It was a really nice experience working as a detective in this SQL murder mystery case. Thank you.