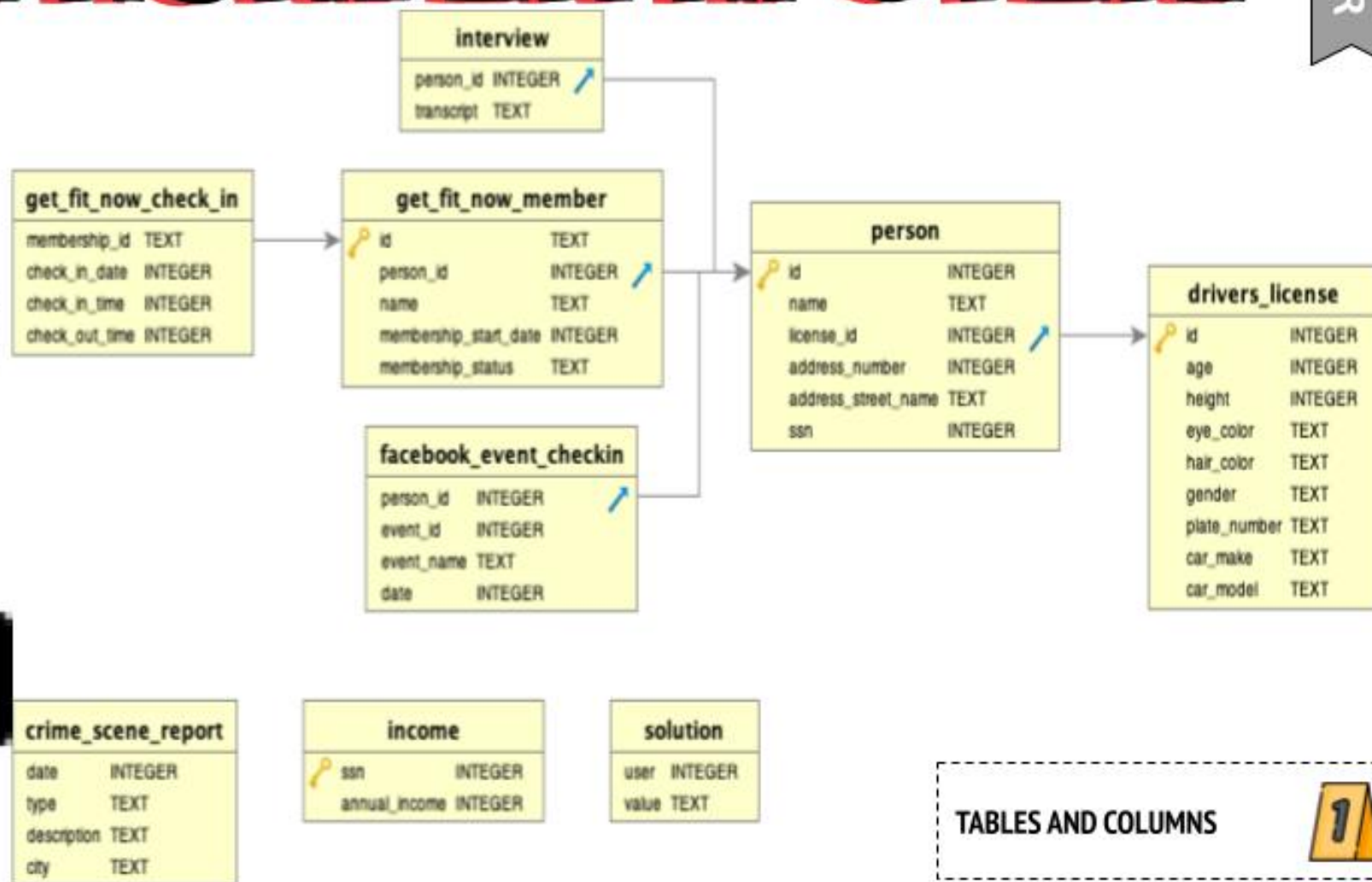# Basic SQL

SQL MURDER MYSTERY

FIRST CLUE... 1

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.

All the clues to this mystery are buried in a huge database, and you need to use SQL to navigate through this vast network of information. Your first step to solving the mystery is to retrieve the corresponding crime scene report from the police department's database.

SQL Murder Mystery took full credit to   Icons made by   and   SQL   from www.flaticon.com

https://mystery.knightlab.com

https://mystery.knightlab.com

To write a query:

1. Table

2. Columns

3. Rows

First clue

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.

SELECT columns **FROM** a table **WHERE** conditions are true

Table: crime_scene_report

Columns: all

Rows: type = "murder"

```
select * from crime_scene_report where type = "murder";
```

Comparison operators

- `=`
- `!=`
- `>=`
...

Logical operators

- `NOT`
- `AND`
- `OR`

Membership operators

- `IN`
- `NOT IN`

SELECT all non-murder cases

```sql
-- using !=
select * from crime_scene_report where type != "murder";
-- using not
select * from crime_scene_report where not type = "murder";
```

Multiple matching criteria (**and** , **or**)

Table: crime_scene_report

Columns: all

Rows:

- type = "murder"

- date = 20180115

- city = "SQL City"

```
select *
from crime_scene_report
where type = "murder"
and date = 20180115
and city = "SQL City";
```

9

Crime report

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".

Matching

Table: person

Columns: all

Rows: address_street_name is northwestern dr

```
select *
from person
where address_street_name = "northwestern dr"
```

Case-insensitive matching

```sql
select *
from person
where lower(address_street_name) = "northwestern dr"
```

`ORDER BY` address_number (descending order)

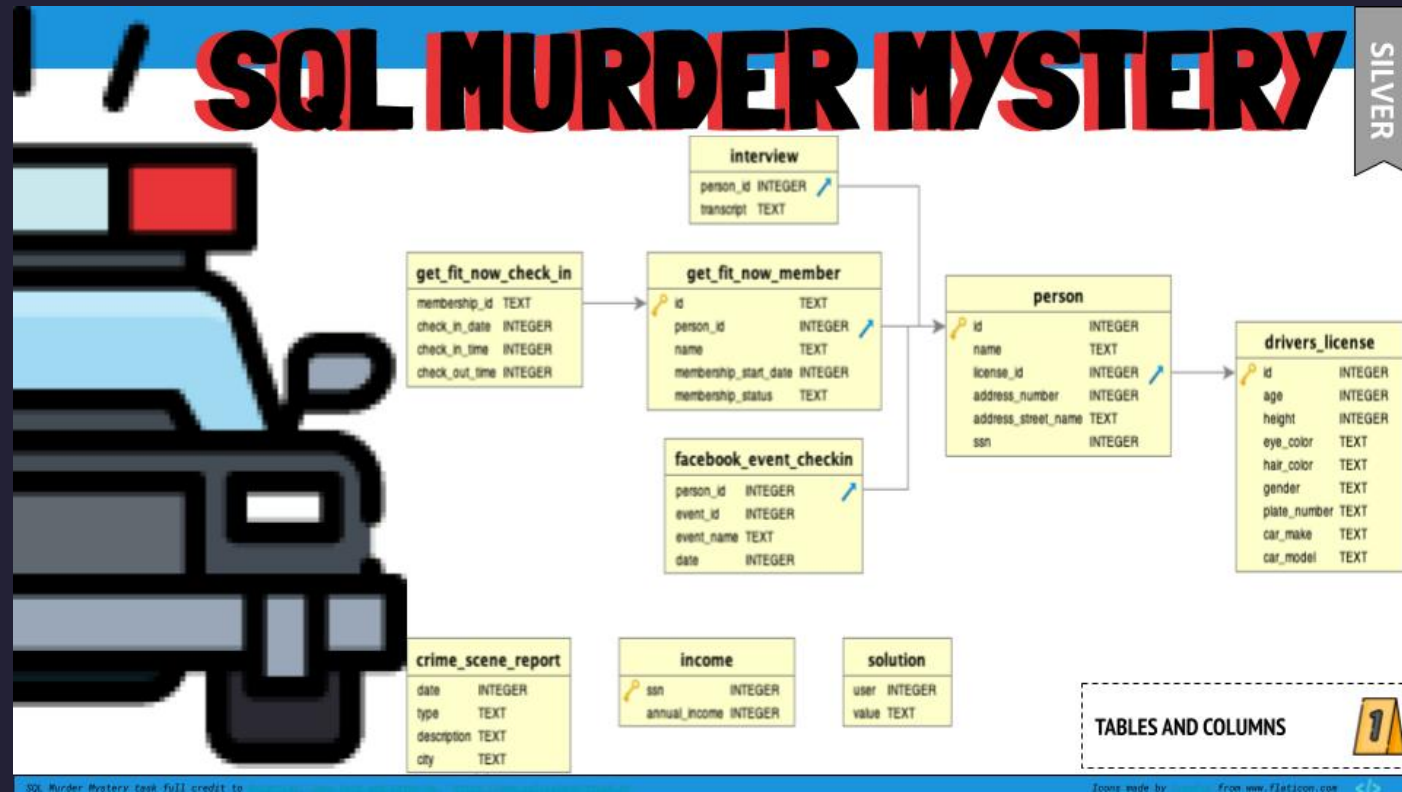**Table: person**

Columns: all

Rows: address_street_name = "Northwestern Dr"

```sql
select *
from person
where address_street_name = "Northwestern Dr"
order by address_number DESC
```

First witness

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

Relationship between tables through (foreign) key

`person_id` in `interview` -> `id` in `person` table

`person_id` in `facebook_event_checkin` -> `id` in `person` table

`person_id` in `get_fit_now_member` -> `id` in `person` table

15

First witness's interview

```sql
select * from interview where person_id = 14887
```

> 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".

Pattern matching `LIKE`

- `%` : Any number of characters

- `_` : Only one character

Exact matching

```sql
-- exact matching
select * from get_fit_now_member where id = "48Z"
select * from get_fit_now_member where id like "48Z"
```

Pattern matching

Starts with:

```
-- followed by any number of characters
select * from get_fit_now_member where id like "48Z%"
-- followed by only one character
select * from get_fit_now_member where id like "48Z_"
```

Ends with:

```
-- any number of preceeding characters
select * from get_fit_now_member where id like "%48Z"
-- only one preceeding character
select * from get_fit_now_member where id like "_48Z"
```

Includes:

```
select * from get_fit_now_member where id like "%48Z%"
```

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".

Gold member whose id starts with "48Z"

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

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

Driver whose license plate includes "H42W"

```
select * from drivers_license
where plate_number like "%H42W%"
```

id	age	height	eye_color	hair_color	gender	plate_number	car_make
183779	21	65	blue	blonde	female	H42W0X	Toyota
423327	30	70	brown	brown	male	0H42W2	Chevrolet
664760	21	71	black	black	male	4H42WR	Nissan

`get_fit_now_member` **<->** `drivers_license`

Using logical operators (AND , OR , NOT)

```sql
select * from person
-- from get_fit_now_member
where id = 67318 or id = 28819
-- from drivers_license
and license_id = 183779 and license_id = 423327 and license_id = 664760
```

Using membership operators (`IN`, `NOT IN`)

```sql
select * from person
where id in (67318, 28819)
and license_id in (183779, 423327, 664760)
```

25

Second witness

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".

🖥️ **What's the person_id of the second witness?**

🖥️ What's the person_id of the second witness? (solution)

Table: person

Columns: all

Rows:

- address_street_name = "Franklin Ave"

- named "Annabel"

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

🖥️ **What's on her interview?**

🖥️ What's on her interview? (solution)

Table: interview

Columns: all

Rows: person_id = 16371

```
select * from interview where person_id = 16371
```

🖥️ **Figure out what time she was at the gym**

🖥️ Figure out what time she was at the gym (solution)

```
select * from get_fit_now_member where person_id = 16371
```

```
select * from get_fit_now_check_in where membership_id = 90081
```

🖥️ **Who else was at the gym during the time she was there?**

🖥️ Who else was at the gym during the time she was there? (solution)

```sql
select * from get_fit_now_check_in
where check_in_time <= 1700
and check_out_time >= 1600
```

BETWEEN, NOT BETWEEN

```sql
select *
from get_fit_now_check_in
where check_in_date = 20180109
and check_out_time between 1600 and 1700
```

```sql
select *
from get_fit_now_check_in
where check_in_date = 20180109
and check_out_time not between 1600 and 1700
```

🖥️ **Who are the members?**

🖥️ Who are the members? (solution)

```sql
select * from get_fit_now_member
-- who were at the gym during the time she was there
where id in ("48Z7A", "48Z55")
```

Wrap up

- `SELECT` columns `FROM` a table `WHERE` conditions are true
  - `LIKE` pattern matching
  - `IN` membership operator
  - `BETWEEN` range operator
- `LIMIT` the number of records returned
- `ORDER BY` columns