Advanced SQL

⏪ Recap

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

⏪ Recap

```sql
select * from person
-- suspect person_id from first witness
where id = 67318
-- suspect person_id from second witness
and id in (67318, 28819)
```

```
select * from interview where person_id=67318
```

> 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.

Clue #1: Find the person who matches the description

> 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.

Clue #1: Find the person who matches the description

```
select *
from drivers_license
where hair_color = 'red'
and car_make = 'Tesla'
and car_model = 'Model S'
and height between 65 and 67
```

`WHERE` column `IN` (list of values)

```
select *
from person
where license_id in (918773, 291182, 202298)
```

Subquery: query inside another query

`WHERE IN` (subquery to return a list of values)

```sql
select *
from person
where license_id in (918773, 291182, 202298)
```

```sql
select *
from person
where license_id in (
    -- subquery that returns the same list of values
    select id
    from drivers_license
    where hair_color = "red"
    and car_make = "Tesla"
    and car_model = "Model S"
    and height between 65 and 67
)
```

The number of columns in subquery must match the number of columns in WHERE clause

```
select *
from person
where (license_id, name) in (
    -- subquery that returns the same list of values
    select id, name -- must match the number of columns in WHERE clause
    from drivers_license
    where hair_color = "red"
    and car_make = "Tesla"
    and car_model = "Model S"
    and height between 65 and 67
)
```

💻 Subquery Exercise

1. Pull up the interview of the first witness, who lives at the last house on "Northwestern Dr."
2. Find the first suspect from person table: membership id starts with "48Z" and license plate includes "H42W"
3. Pull up the interview of the second witness. Her name is Annabel and she lives somewhere on "Franklin Ave".
4. Check in time and check out time of the second witness.

Subquery Exercise (solution)

1. Pull up the interview of the first witness, who lives at the last house on "Northwestern Dr."

```
select * from interview
where person_id = (
    select id from person
    where address_street_name = "Northwestern Dr"
    order by address_number DESC limit 1
    )
```

2. Find the first suspect from person table: membership id starts with "48Z" and license plate includes "H42W"

```
select * from person
where id in (select person_id from get_fit_now_member where id like "48Z%")
and license_id in (select id from drivers_license where plate_number like "%H42W%")
```

💻 Subquery Exercise (solution)

3. Pull up the interview of the second witness. Her name is Annabel and she lives somewhere on "Franklin Ave".

```
select * from interview
where person_id = (select id from person where address_street_name = "Franklin Ave" and name like "%Annabel%")
```

4. Check in time and check out time of the second witness.

```
select check_in_time, check_out_time from get_fit_now_check_in
where membership_id = (
    select id from get_fit_now_member
    where person_id = (
        select id from person where address_street_name = "Franklin Ave" and name like "%Annabel%")
)
```

Find the second suspect

1. Second witness: Annabel, Franklin Ave
2. Membership id of the second witness
3. Check in time and check out time of the second witness
4. Find members who were at the gym during the same time as the second witness

```
select * from get_fit_now_member
where id in (
    select membership_id from get_fit_now_check_in
    where check_in_time <= (
        select check_in_time from get_fit_now_check_in
        where membership_id = (
            select id from get_fit_now_member
            where person_id = (select id from person where address_street_name = "Franklin Ave" and name like "%Annabel%")
            )
        )
    and check_out_time >= (
        select check_out_time from get_fit_now_check_in
        where membership_id = (
            select id from get_fit_now_member
            where person_id = (select id from person where address_street_name = "Franklin Ave" and name like "%Annabel%")
            )
        )
    )
```

Find the killer

1. From first witness: gym membership id starts with "48Z" and license plate includes "H42W"
2. From second witness
3. Cross-reference the person IDs of the suspects in the `person` table

```sql
select * from person
-- from first witness
where id in (select person_id from get_fit_now_member where id like "%48Z%")
and license_id in (select id from drivers_license where plate_number like "%H42W%")
-- from second witness
and id in (
    select person_id from get_fit_now_member
    where id in (
        select membership_id from get_fit_now_check_in
        where check_in_time <= 1700
        and check_out_time >= 1600
        )
    )
```

Clue #2

> I know that she attended the SQL Symphony Concert 3 times in December 2017.

Table: `facebook_event_checkin`

Columns: all

Rows:

- SQL Symphony Concert
- December 2017

```
select *
from facebook_event_checkin
where event_name like "%SQL Symphony Concert%"
and date between 20170101 and 20171231
order by person_id
```

`select` column `from` table `group by` column

`GROUP BY` to aggregate data at a group level

```
select person_id, count(*)
from facebook_event_checkin
where event_name like "%SQL Symphony Concert%"
and date between 20170101 and 20171231
group by person_id
```

Group functions

- `count()`
- `sum()`
- `avg()`
- `max()`
- `min()`
- `stddev()`: standard deviation
- `variance()`: variance

...

https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html

HAVING to filter groups

```
select person_id, count(*)
from facebook_event_checkin
where event_name like "%SQL Symphony Concert%"
and date between 20170101 and 20171231
group by person_id
having count(*)=3
```

Alias AS

```
select person_id, count(*) as num_attendance
from facebook_event_checkin
where event_name like "%SQL Symphony Concert%"
and date between 20170101 and 20171231
group by person_id
having num_attendance=3
```

`ORDER BY` with functions

Who attended SQL Symphony Concert the most in 2017?

```
select person_id, count(*)
from facebook_event_checkin
where event_name like "%SQL Symphony Concert%"
and date between 20170101 and 20171231
group by person_id
order by count(*) desc
```

Facebook Event Checkin

1. When was the first event recorded?
2. how many people attended each event?
3. Who attended the most events in 2017?
4. how many times SQL Symphony Concert was held in 2017?

💻 List all the events that were held more than three times in 2017. What was the most popular event in 2017?

💻 List all the events that were held more than three times in 2017. What was the most popular event in 2017? (solution)

```sql
select event_id, count(distinct date) num_times_held
from facebook_event_checkin
where date between 20170101 and 20171231
group by event_id
having num_times_held > 3
order by num_times_held desc
```

1. Clue #1: People that match the description (`license_id` from `drivers_license`)
2. Clue #2: People that attended SQL Symphony Concert 3 times in 2017 (`person_id` from `facebook_event_checkin`)

Using Subquery

```
select *
from person
where license_id in (
    select id
    from drivers_license
    where hair_color = "red"
    and car_make = "Tesla"
    and car_model = "Model S"
    and height between 65 and 67
)
and id in (
    select person_id
    from facebook_event_checkin
    where event_name like "%SQL Symphony Concert%"
    and date between 20170101 and 20171231
    group by person_id
    having count(*)=3
)
```

Using `JOIN`

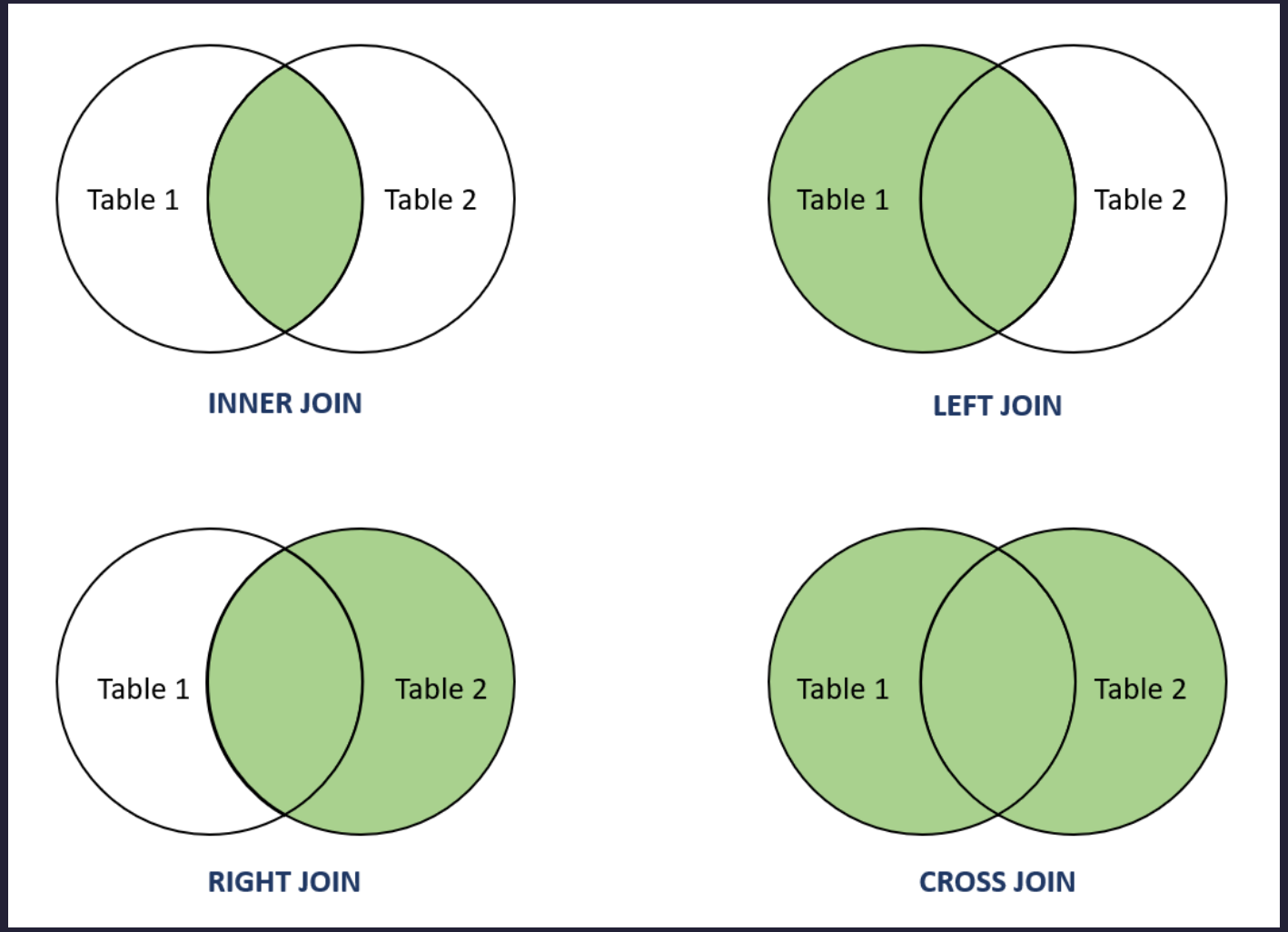
Table 1
Table 2
INNER JOIN
Table 1
Table 2
LEFT JOIN
Table 1
Table 2
RIGHT JOIN
Table 1
Table 2
CROSS JOIN

`JOIN ON` (inner join)

- What tables do you want to join?: `person` and `drivers_license`
- What column(s) do you want to join on?: `license_id` from `person` and `id` from `drivers_license`

```
select person.id, drivers_license.id
from (person join drivers_license on person.license_id = drivers_license.id)
```

Notice `.` to specify which table a column belongs to

> What tables do you want to join?
> What column(s) do you want to join on?

1. Join `person` and `get_fit_now_member`
2. Join `get_fit_now_member` and `get_fit_now_check_in`
3. Join `person` and `get_fit_now_check_in`

Join `person` and `drivers_license`

```
select *
from person
join drivers_license on person.license_id = drivers_license.id
where hair_color = "red"
and car_make = "Tesla"
and car_model = "Model S"
and height between 65 and 67
```

Join `person`, `drivers_license`, and subquery `facebook_event_checkin`

```
select *
from person
join drivers_license on person.license_id = drivers_license.id
where hair_color = "red"
and car_make = "Tesla"
and car_model = "Model S"
and height between 65 and 67
and person.id in (
    select person_id
    from facebook_event_checkin
    where event_name like "%SQL Symphony Concert%"
    and date between 20170101 and 20171231
    group by person_id
    having count(*)=3
    )
```

Other `JOIN` syntaxes

`JOIN USING`:

```
select * from person join drivers_license using (license_id)
```

`NATURAL JOIN`:

```
select * from person natural join drivers_license
```

`WHERE`:

```
select * from person, drivers_license
where person.license_id = drivers_license.id
```

`LEFT JOIN` and `RIGHT JOIN`

left join:

```
select * from person left join get_fit_now_member on person.id = get_fit_now_member.person_id
```

right join:

```
select * from person right join get_fit_now_member on person.id = get_fit_now_member.person_id
```