WHAT ARE SUBQUERIES

Subqueries are queries nested within other queries.

```
SELECT id, start_time FROM screenings
WHERE film_id IN

(SELECT id FROM films
WHERE length_min > 120)
;
```

subqueries

def - queries nested within other queries - can be in WHERE caluse or FROM
 used in SELECT,INSERT,UPDATE or DELETE query
 NON-correlated and Correlated

NON-CORRELATED SUBQUERY

The inner query can run independently of the outer query.

```
SELECT id, start_time FROM screenings
WHERE film_id IN

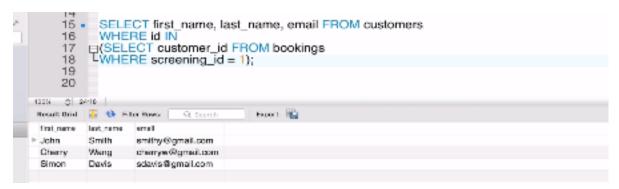
(SELECT id FROM films
WHERE length_min > 120)
;
```

Inner query runs first and produces a result set, which is then used by the outer which are then used by the outer query.

part1

```
1 USE cinema booking system;
      3 • SELECT id, start_time FROM screenings
WHERE film id IN
S (SELECT id FROM films
WHERE length_min > 120);
       8 . SELECT id FROM films
             WHERE length_min > 120;
     10
     12 •
13 •
14
15
16
             SELECT * FROM customers;
SELECT * FROM bookings;
                                        Edit 🔏 🖶 📻 Especiarcort 🛀 🕲
Result third 🍒 😘 Filter Roses | 🔍 Scott h.
                                                                                                                                   iiii
Ressir
     2017-10-02 19:80:00
15 2017-10-02 20:00:00
      2017-10-03 19:00:00
23
     2017-10-04 16:50:00
     2017-10-04 19:80:00
2017-10-06 16:00:00
                           we can see it's returned id and start time
34
      2017-10-07 13:30:00
```

b, customers that made a booking for screening_id =1



part2

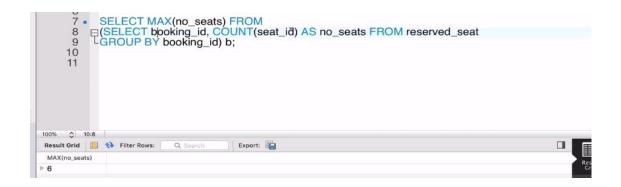
no of seats reserved for booking_id



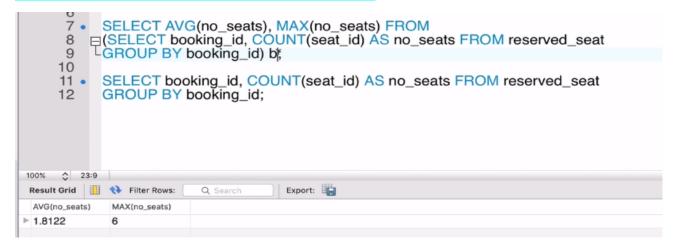
find the max number of seats reserved by a part booking_id

since we are creating a direct table by

SELECT booking_id, COUNT(seat_id) AS no_seats FROM reserved_seat GROUP BY booking_id SO, we have to provide a name too, say b, then just SELECT MAX(no_seats) FROM b



WE can choose multiple columns from this direct table also



CORRELATED SUBQUERY

The inner query can't run independently of the outer query.

SELECT SCREENING_ID, CUSTOMER_ID,
(SELECT COUNT(SEAT_ID)
FROM RESERVED_SEAT WHERE BOOKING_ID = B.ID)
FROM BOOKINGS B;

The inner guery runs for every row in the outer guery.

the inner query is running multiple times



now, running the inner query will give error

```
SELECT COUNT(seat_id)
FROM reserved_seat WHERE booking_id = b.id;
```

as booking as b was declared in the outer query



EX. a. non correlated query

SELECT name, length_min FROM films

WHERE length_min >

(SELECT AVG(length_min) as average FROM films);



verification that it is NCQ

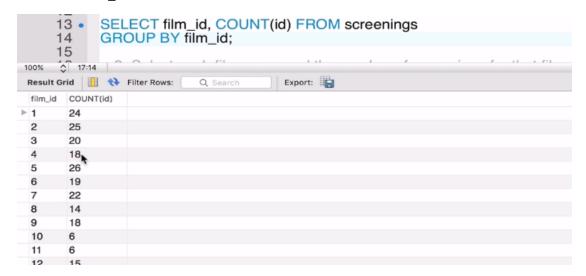


-- 2. Select the maximum number and the minimum number of screenings for a particular film.

AGAIN Ncq

SELECT film_id, COUNT(id) FROM screenings

GROUP BY film_id



gives each film id has how many screenings

→ now just select max and min from this table



-- 3. Select each film name and the number of screenings for that film.

SELECT name

(SELECT COUNT(id) FROM screenings

WHERE film_id=f.id)

FROM films f;

→ we could have use group by screening but films and screenings are two diff table so we cant

