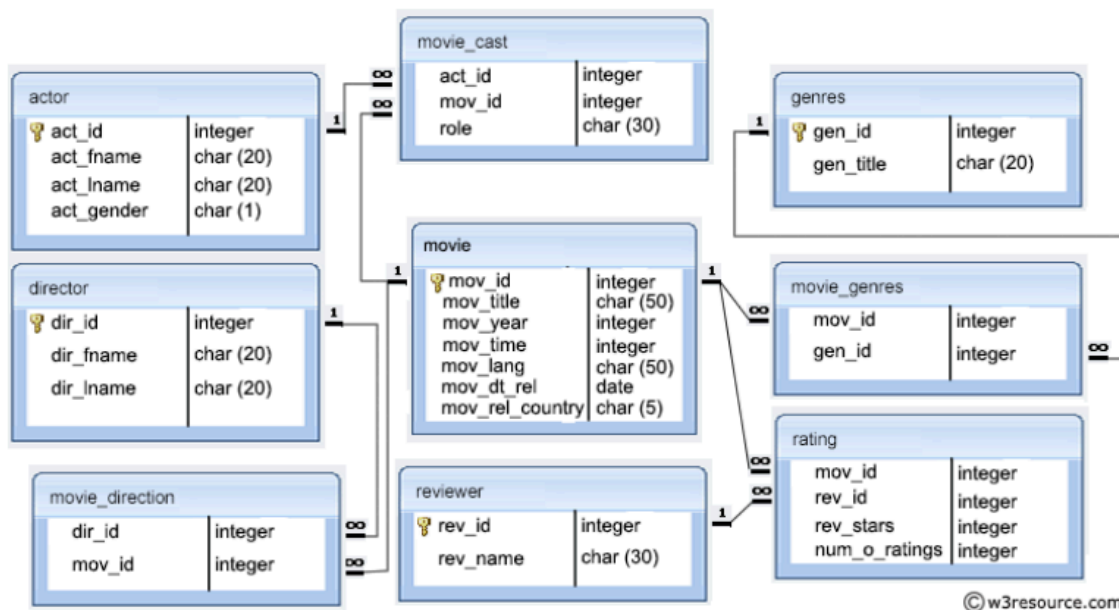


Interview Challenge (Data Engineer Internship) - 3 days

โจทย์



เขียน SQL Script จากฐานข้อมูลภาพยนตร์ตามรูปด้านบน

1. แสดงข้อมูล ผู้กำกับ (director) ที่ได้รับ ความนิยม (rating) สูงที่สุดของแต่ละประเภทหนัง (movie_genres)

```
SELECT
    d.dir_id,
    d.dir_fname,
    d.dir_lname,
    g.gen_title,
    MAX(r.num_o_ratings) AS highest_rating
FROM director AS d
JOIN movie_direction AS md ON d.dir_id = md.dir_id
JOIN movie AS m ON md.mov_id = m.mov_id
JOIN movie_genres AS mg ON m.mov_id = mg.mov_id
JOIN genres AS g ON g.gen_id = mg.gen_id
JOIN rating AS r ON r.mov_id = m.mov_id
GROUP BY d.dir_id, d.dir_fname, d.dir_lname, g.gen_title
ORDER BY highest_rating DESC;
```

2. แสดงข้อมูล นักแสดงแต่ละคนแสดงหนังแต่ละเรื่องรวมเป็นระยะเวลาเท่าใด โดยคิดเฉพาะหนังที่ คะแนนความนิยมไม่ใช่ null

```
SELECT a.act_id, a.act_fname,  
       a.act_lname,  
       m.mov_title,  
       SUM(m.mov_time) AS total_time  
FROM actor AS a  
JOIN movie_cast AS mc ON mc.act_id = a.act_id  
JOIN movie AS m ON m.mov_id = mc.mov_id  
JOIN rating AS r ON m.mov_id = r.mov_id  
WHERE rev_stars IS NOT NULL  
GROUP BY a.act_id, a.act_fname, a.act_lname, m.mov_title  
ORDER BY total_time DESC;
```

3. แสดงข้อมูล จำนวนครั้งที่ ผู้กำกับ (director) และนักแสดงหญิง (actor) เคยร่วมงานกัน โดยแสดง เฉพาะ 5 ลำดับแรกของนักแสดงหญิงที่ร่วมงานกับผู้กำกับคนนั้นบ่อยที่สุด

```
SELECT d.dir_id,  
       d.dir_fname,  
       d.dir_lname,  
       a.act_id,  
       a.act_fname,  
       a.act_lname,  
       COUNT(*) AS collaboration  
FROM director AS d  
JOIN movie_direction AS md ON d.dir_id = md.dir_id  
JOIN movie AS m ON md.mov_id = m.mov_id  
JOIN movie_cast AS mc ON m.mov_id = mc.mov_id  
JOIN actor AS a ON mc.act_id = a.act_id  
WHERE a.act_gender = 'F'  
GROUP BY d.dir_id, d.dir_fname, d.dir_lname, a.act_id, a.act_fname,  
a.act_lname  
ORDER BY collaboration DESC;
```

4. เปลี่ยนประเภทข้อมูล gen_title จาก char (20) เป็น integer โดยเรียงตามตัวอักษร เช่น action แปลงเป็น 1, animation แปลงเป็น 2, biography แปลงเป็น 3, ...

- Add a new column for the integer

```
ALTER TABLE genres  
ADD COLUMN gen_order INTEGER;
```

- Update the new column

```
UPDATE genres  
SET gen_order = subquery.number  
FROM (  
  SELECT  
    gen_id,  
    ROW_NUMBER() OVER (ORDER BY gen_title) AS number  
  FROM genres  
) AS subquery  
WHERE genres.gen_id = subquery.gen_id;
```

- Drop the old column

```
ALTER TABLE genres  
DROP COLUMN gen_title;
```

5. แก้ไขเพศของนักแสดง ที่ชื่อขึ้นต้นด้วยคำว่า Em, Char หรือ ชื่อลงท้ายด้วยคำว่า dy, sy, lia เป็น เพศหญิง โดยไม่สนใจตัวเล็กตัวใหญ่

```
UPDATE actor  
SET act_gender = 'F'  
WHERE  
  LOWER(act_fname) LIKE 'em%'  
  OR LOWER(act_fname) LIKE 'char%'  
  OR LOWER(act_fname) LIKE '%dy'  
  OR LOWER(act_fname) LIKE '%sy'  
  OR LOWER(act_fname) LIKE '%lia';
```