Fawwaz Abrial Saffa

18221067

**Lembar Kerja Praktikum 2**

**IF2140 Pemodelan Basis Data**

**Waktu: 100 Menit**

**7.10 WIB - 8.50 WIB**

# **Soal**

1. **Bagian 1**

Kerjakan bagian ini terlebih dahulu sebelum pindah ke bagian 2. Query dan SS yang diminta sama dengan praktikum sebelumnya. Pastikan SS yang dilampirkan sesuai dengan ketentuan yang telah tercantum pada file **Panduan Praktikum 2.**

1. Tampilkan seluruh nama lengkap aktor yang tampil pada film “BORN SPINAL”! (**Gunakan Join Statement**)

| Query | SELECT Concat(first\_name, ' ', last\_name)  FROM film NATURAL JOIN film\_actor, actor  WHERE actor.actor\_id = film\_actor.actor\_id  AND title = 'BORN SPINAL'; |
| --- | --- |
| SS |  |

1. Tampilkan kategori film, nama aktor, dan jumlah film yang dimainkan oleh masing-masing aktor di dalam masing-masing kategori dengan ketentuan aktor tersebut pernah bermain lebih dari 5 film pada kategori tersebut.

| Query | CREATE VIEW no2 AS  SELECT actor.actor\_id as ai, name as cat, first\_name, last\_name, count(actor.actor\_id) as cnt  FROM actor  NATURAL JOIN film\_actor  NATURAL JOIN film\_category  NATURAL JOIN category  GROUP BY actor.actor\_id, name  HAVING count(actor.actor\_id) > 5;  SELECT cat, CONCAT(first\_name, ' ', last\_name), cnt  FROM no2  WHERE cnt > 5 |
| --- | --- |
| SS |  |

1. Buatlah sebuah query untuk menunjukkan id film yang memiliki length lebih dari 100 satuan, berkategori Family, dan memiliki *special feature* berupa Behind the Scenes namun tidak pernah dirental antara bulan Juli sampai bulan Desember! Cobalah untuk membuat query ini hanya dengan menggunakan 3 *join statement*!

***Hint: Gunakan fungsi MONTH()***

| Query | SELECT distinct title  FROM film  NATURAL JOIN film\_category  NATURAL JOIN inventory  NATURAL JOIN rental  WHERE category\_id = (SELECT category\_id FROM category WHERE name = 'Family' )  AND special\_features = 'Behind the Scenes'  AND MONTH(rental\_date) < 7 |
| --- | --- |
| SS |  |

1. Nona Nin sedang sedih sekali. Oleh karena itu, Nona Nin berniat menonton film dengan kategori komedi untuk menghibur diri. Tetapi, karena deadline tugas besar yang menumpuk, ia hanya bisa menonton film dengan jangka waktu singkat, yaitu *length* di bawah 100. Ia juga hanya mau meminjam di store dengan ID 2 karena jarak store itu lebih dekat dengan rumahnya. Bantu Nona Nin dengan menampilkan pilihan judul film yang bisa ia tonton!

| Query | SELECT distinct title  FROM film  NATURAL JOIN film\_category  natural join inventory  natural join store  WHERE store\_id = '2' AND  category\_id = (SELECT category\_id FROM category WHERE name = 'Comedy' ) AND  length < 100; |
| --- | --- |
| SS |  |

1. Buatlah sebuah query untuk menampilkan id aktor dan nama lengkap dari aktor yang pernah bermain di film dengan category Family atau Music.

| Query | SELECT distinct actor\_id, CONCAT(first\_name, ' ', last\_name)  FROM film  NATURAL JOIN film\_category  natural join film\_actor  natural join actor  WHERE  category\_id = SOME (SELECT category\_id FROM category WHERE name = 'Family' OR name = 'Music'); |
| --- | --- |
| SS |  |

1. Tampilkan tanggal dan jumlah film terbanyak yang berhasil disewakan oleh perusahaan Sakila dalam 1 hari! (Perusahaan Sakila yang dimaksud adalah secara keseluruhan, bukan cabang tertentu)

***HINT : Gunakan Sub-query***

| Query | CREATE VIEW no6 AS  SELECT rental\_date, count(DAY(rental\_date)) as banyak  FROM rental NATURAL JOIN inventory  GROUP BY DAY(rental\_date);  SELECT rental\_date, max(banyak)  FROM no6; |
| --- | --- |
| SS |  |

1. Buatlah sebuah query untuk menampilkan semua nama lengkap aktor yang memainkan film dengan rental terbanyak.

***Hint : Gunakan fungsi CONCAT()***

| Query | CREATE VIEW no7 AS  SELECT film\_id, count(rental\_id) cnt FROM inventory natural join rental GROUP BY film\_id;  CREATE VIEW max AS  SELECT film\_id, max(cnt) FROM no7;  SELECT CONCAT(first\_name, ' ', last\_name)  FROM max natural join film\_actor natural join actor; |
| --- | --- |
| SS |  |

1. **Bagian 2**

Kerjakan bagin ini **setelah** selesai mengerjakan bagian 1. Terdapat 3 file SS yang diminta pada bagian ini, yaitu **sebelum, saat, dan sesudah** melakukan modifikasi). Bukti SS sebelum dan sesudah harus menunjukkan keadaan sebelum modifikasi dan bukti SS setelah harus menunjukkan keadaan setelah dilakukan modifikasi. SS saat pastikan terdapat tulisan ‘Query OK’ dan berapa rows yang dimodifikasi.

1. Buatlah sebuah query untuk membuat view dengan nama untuk\_nona\_nike yang berisi judul, deskripsi, dan durasi rental dari film-film yang disewa selama lebih dari 8 hari dan memiliki fitur spesial Trailer dan Behind the Scenes. Pastikan view diurutkan berdasarkan durasi rental dari yang paling tinggi dan judul sesuai dengan abjad.

| Query | CREATE VIEW untuk\_nona\_nike AS  SELECT title, description, rental\_duration  FROM film  WHERE rental\_duration > 8  AND special\_features = 'Trailer'  AND special\_features = 'Behind the Scenes'  ORDER BY rental\_duration, title; |
| --- | --- |
| SS sebelum |  |
| SS saat |  |
| SS sesudah |  |

# Setelah ditonton ulang, ternyata aktor Uma Wood juga terlibat dalam film Affair Prejudice. Buatlah query untuk memasukkan data bahwa Uma Wood juga tampil dalam film Affair Prejudice.

| Query |  |
| --- | --- |
| SS sebelum |  |
| SS saat |  |
| SS sesudah |  |

# Belakangan ini, nama aktor Dan Streep kembali viral karena sensasi yang ia buat. Hal ini berdampak positif pada film-film yang pernah ia bintangi. Film-film yang dibintangi oleh Dan Streep kembali dicari-cari oleh masyarakat dan membuat perusahaan rental film dapat meraup keuntungan yang cukup signifikan. Hal ini dimanfaatkan oleh Tuan Jek, manager perusahaan rental, untuk dapat menaikkan harga rental film-film yang dibintangi oleh Dan Streep. Tuan Jek memerintahkan Database Administrator perusahaan untuk dapat menaikkan harga rental film yang dibintangi oleh Dan Street sebesar 20% dari harga awal.

Bayangkan Anda sebagai Database Administrator dari perusahaan Tuan Jek. Bantulah Tuan Jek untuk dapat menaikkan harga rental film pada database perusahaan!

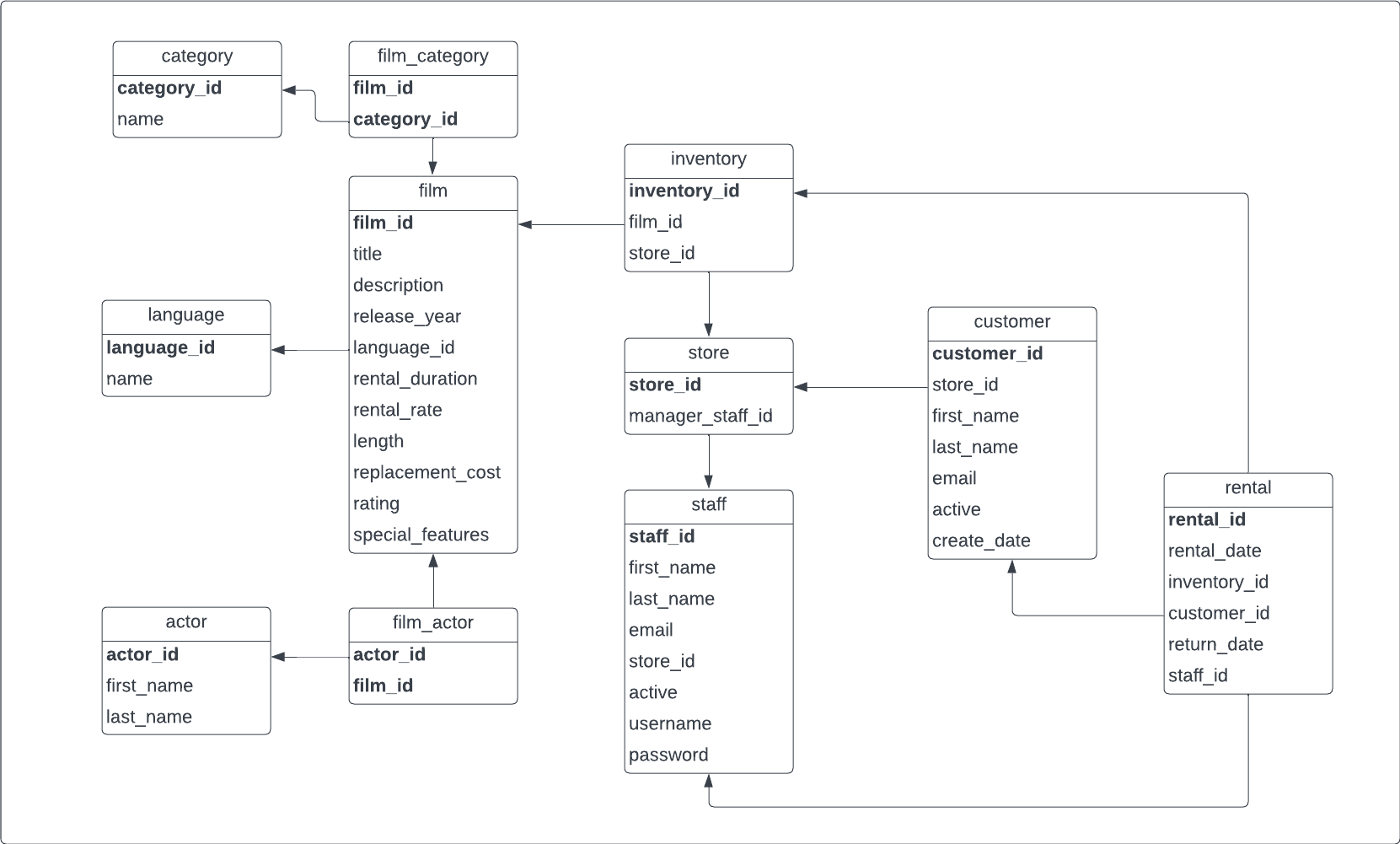
| Query |  |
| --- | --- |
| SS sebelum |  |
| SS saat |  |
| SS sesudah |  |

1. Buatlah query untuk menghapus seluruh bahasa yang ada di tabel language selain bahasa inggris.

| Query | DELETE FROM language  WHERE name != 'English'; |
| --- | --- |
| SS sebelum |  |
| SS saat |  |
| SS sesudah |  |

# 

# **Skema Basis Data**



| actor = (**actor\_id**, first\_name, last\_name)  film\_actor = (**film\_id, actor\_id**)  film = (**film\_id**, title, description, release\_year, language\_id, rental\_duration, rental\_rate, length, replacement\_cost, rating, special\_features)  film\_category = (**film\_id**, **category\_id**)  category = (**category\_id**, name)  inventory = (**inventory\_id**, film\_id, store\_id)  store = (**store\_id**, manager\_staff\_id)  staff = (**staff\_id**, first\_name, last\_name, email, store\_id, active, username, password)  rental = (**rental\_id**, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id)  customer = (**customer\_id**, store\_id, first\_name, last\_name, email, active, create\_date) |
| --- |

# 

# **Hint**

1. Penggabungan pada string, dapat dilakukan dengan fungsi CONCAT(*string1*, *string2, .., string-n*)

**Contoh**:

CONCAT(‘aku’, ‘kamu’) akan mengembalikan “akukamu”.

1. Untuk mengambil bulan dari sebuah atribut bertipe data *date* atau *datetime* dapat menggunakan fungsi MONTH(*datetime1*).

**Contoh**:

MONTH*(‘2022-09-30’)* akan mengembalikan nilai 9.