

**IF2240 - Basis Data**

# **Praktikum 1**

**Operasi Dasar SQL**

19 Maret 2025

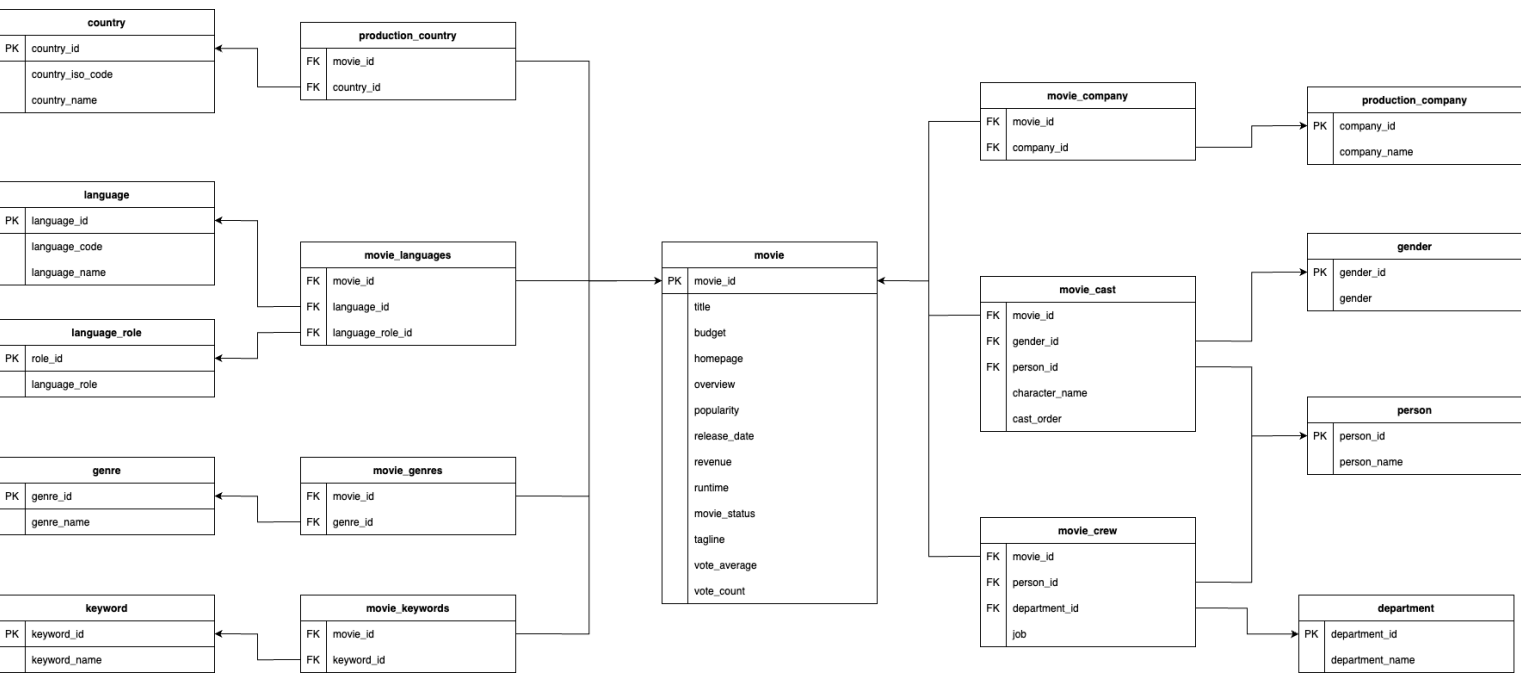


M. Zahran Ramadan A. - Razi Rachman Widyadhana

13523104 - 13523004

**Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung  
2025**

# IMDb (IMDatabase)



## Legenda

No	Nama Tabel	Keterangan
1	movie	Informasi setiap film.
2	production_country	Negara tempat film diproduksi. Tabel pemetaan negara dengan film.
3	movie_languages	Bahasa yang digunakan pada film. Tabel pemetaan bahasa dengan film.
4	movie_genres	Genre film. Tabel pemetaan genre dengan film.
5	movie_keywords	Keyword film. Tabel pemetaan kata kunci dengan film.
6	movie_company	Perusahaan produksi film. Tabel pemetaan perusahaan produksi dengan film.
7	movie_cast	Pemeran film. Tabel pemetaan pemeran dengan film.
8	movie_crew	Kru film. Tabel pemetaan kru pekerja dengan film.
9	country	Negara.
10	language	Bahasa.
11	language_role	Jenis bahasa yang digunakan (Original atau Spoken).
12	genre	Genre.
13	keyword	Keyword.
14	production_company	Perusahaan pemroduksi.
15	gender	Gender.
16	person	Orang.
17	department	Departemen dari para kru.

## **Petunjuk**

1. Praktikum ini wajib dikerjakan di dalam Laboratorium Teknik Informatika ITB.
2. Keyword yang diperbolehkan pada praktikum ini hanyalah kombinasi dari keyword berikut, SELECT (DISTINCT), FROM, WHERE, LIKE, BETWEEN, AND, OR, UNION, INTERSECT, dan EXCEPT. Penggunaan fungsi dasar SQL hanya diperbolehkan sesuai dengan petunjuk soal. DILARANG menggunakan keyword selain yang tertera di atas.

# 1. Kru Kolaborasi

Pak Red, seorang penggemar film sejati, baru saja menonton film di mana seorang aktor juga bekerja sebagai kru di balik layar. Hal ini membuatnya penasaran siapa saja orang-orang multi-talenta di dunia perfilman yang memiliki peran ganda tersebut. Bantu Pak Red untuk menampilkan film, beserta nama karakter yang diperankan, pekerjaan di kru, dan nama orangnya.

Query	<pre>SELECT m.title, mc.character_name, mcr.job, p.person_name FROM movie m, movie_cast mc, movie_crew mcr, person p WHERE m.movie_id = mc.movie_id AND m.movie_id = mcr.movie_id AND mc.person_id = mcr.person_id AND mc.person_id = p.person_id;</pre>
-------	--

SS Query dan Hasil

Query ditambah LIMIT 10 *for the sake* untuk melakukan *screenshot* pada 10 rows pertama.

```
MariaDB [movies]> SELECT m.title, mc.character_name, mcr.job, p.person_name
-> FROM movie m, movie_cast mc, movie_crew mcr, person p
-> WHERE m.movie_id = mc.movie_id
-> AND m.movie_id = mcr.movie_id
-> AND mc.person_id = mcr.person_id
-> AND mc.person_id = p.person_id
-> LIMIT 10;
```

title	character_name	job	person_name
Four Rooms	Chester Rush	Director	Quentin Tarantino
Four Rooms	Chester Rush	Executive Producer	Quentin Tarantino
Four Rooms	Chester Rush	Writer	Quentin Tarantino
Four Rooms	Long Hair Yuppy Scum	Producer	Lawrence Bender
Star Wars	Stormtrooper (special edition) (uncredited)	Producer	Rick McCallum
Star Wars	Cantina Alien (uncredited)	Animation	Phil Tippett
Finding Nemo	Crush (voice)	Screenplay	Andrew Stanton
Finding Nemo	Crush (voice)	Director	Andrew Stanton
Finding Nemo	Crush (voice)	Original Story	Andrew Stanton
Finding Nemo	Mr. Ray (voice)	Screenplay	Bob Peterson

Restoration	Harold	Director	Zack Ward	
Restoration	Harold	Screenplay	Zack Ward	
Solitude	Carl Colson	Director	Livingston Oden	
Solitude	Dick Hancock	Director	Taylor Scott Olson	
연천상륙작전	Lt. Col. Edward L. Rowny	Writer	Sean Dulake	
Perfect Cowboy	Jimmy	Director	Ken Roht	
Puss in Boots	Little Boy Blue / Friar Miller / Prison Guard / Manuel / Rafael (voice)	Director	Chris Miller	
Puss in Boots	Conandante / Hombre del bigote (voice)	Executive Producer	Guillermo del Toro	
Puss in Boots	Bartender / Mean Boy / Wagon Driver / Hotel Owner / Rodrigo (voice)	Screenplay	Tom Wheeler	
Puss in Boots	Bartender / Mean Boy / Wagon Driver / Hotel Owner / Rodrigo (voice)	Story	Tom Wheeler	
Puss in Boots	Crazy Woman / Mean Girl / Milk Lady / Little Boy (voice)	Producer	Latifa Ouau	

```
2929 rows in set (0.252 sec)
```

## 2. Si Paling Research Dulu

Pak Anzy adalah seorang *hopeless romantic* yang ingin mempelajari film agar memiliki topik untuk dibicarakan saat *blind date*. Pak Anzy mendengar bahwa para kaum wanita sangat menyukai film yang dimainkan oleh Channing Tatum 🥰👉 dan ingin menonton film-film yang dimainkannya. Tetapi, karena Pak Anzy tidak memiliki banyak waktu untuk menonton semuanya, Pak Anzy hanya akan menonton film-film dengan rating rata-rata di atas 7. Bantu Pak Anzy untuk mendapatkan semua judul unik yang perlu ia tonton beserta rating rata-ratanya.

Query	<pre>SELECT m.title, m.vote_average FROM movie m, movie_cast mc, person p WHERE m.movie_id = mc.movie_id AND mc.person_id = p.person_id AND m.vote_average &gt; 7 AND p.person_name = "Channing Tatum";</pre>
SS Query dan Hasil	
<pre>MariaDB [movies]&gt; SELECT m.title, m.vote_average -&gt; FROM movie m, movie_cast mc, person p -&gt; -&gt; WHERE m.movie_id = mc.movie_id -&gt; AND mc.person_id = p.person_id -&gt; AND m.vote_average &gt; 7 -&gt; AND p.person_name = "Channing Tatum"; +-----+-----+   title             vote_average   +-----+-----+   Coach Carter     7.30             The Lego Movie    7.50             The Book of Life   7.30             The Hateful Eight   7.60           +-----+-----+ 4 rows in set (0.046 sec)  MariaDB [movies]&gt;</pre>	

## 3. I'm just a girl, standing in front of a boy, asking him to love her.

Pak Anzy baru saja menyelesaikan film *Notting Hill*. Ia sekarang terobsesi dengan film yang menggabungkan genre *romance* dan juga *comedy*. Bantu Pak Anzy untuk mendapatkan semua judul unik dengan genre *romance* dan juga *comedy*.

[HINT] Untuk soal ini **WAJIB** menggunakan salah satu atau lebih dari keyword **INTERSECT/UNION/EXCEPT**.

Query	<pre>(SELECT DISTINCT m.title FROM movie m, movie_genres mg, genre g WHERE m.movie_id = mg.movie_id AND g.genre_id = mg.genre_id AND g.genre_name = "Comedy") INTERSECT (SELECT DISTINCT m.title FROM movie m, movie_genres mg, genre g WHERE m.movie_id = mg.movie_id AND g.genre_id = mg.genre_id AND g.genre_name = "Romance");</pre>
SS Query dan Hasil	
<pre>MariaDB [movies]&gt; (SELECT DISTINCT m.title -&gt; FROM movie m, movie_genres mg, genre g -&gt; WHERE m.movie_id = mg.movie_id -&gt; AND g.genre_id = mg.genre_id -&gt; AND g.genre_name = "Comedy") -&gt; INTERSECT -&gt; (SELECT DISTINCT m.title -&gt; FROM movie m, movie_genres mg, genre g -&gt; WHERE m.movie_id = mg.movie_id -&gt; AND g.genre_id = mg.genre_id -&gt; AND g.genre_name = "Romance"); +-----+-----+   title   +-----+-----+   Forrest Gump   Pretty Woman   Groundhog Day   Jackie Brown   Le fabuleux destin d'Amélie Poulain   Volver   Some Like It Hot   High Fidelity   About a Boy   The Apartment +-----+-----+   Da Sweet Blood of Jesus   Anomalisa   There's Always Woodstock   My Big Fat Greek Wedding 2   Accidental Love   Anderson's Cross   Christmas Eve   Fifty Shades of Black   Me You and Five Bucks   The Perfect Match +-----+-----+ 484 rows in set (0.045 sec)  MariaDB [movies]&gt; </pre>	

## 4. Cinematic Extremes

Dr. D sedang meneliti dua kategori film yang mewakili dua sisi berlawanan dalam spektrum sinematik:

1. Film-film produksi USA ("United States of America") yang mendapat pujian kritis dengan vote rata-rata di atas 8.0 dan telah ditonton oleh banyak penonton yaitu dengan jumlah vote lebih dari 1000
2. Film-film produksi Prancis ("France") yang dianggap gagal dengan vote rata-rata di bawah 5.0 namun masih memiliki jumlah penonton yang cukup signifikan yaitu dengan jumlah vote lebih dari 100

Bantu Dr. D menyusun penelitiannya dengan mengambil ID film dan judul film yang termasuk kategori-kategori tersebut.

*[HINT] Untuk soal ini **WAJIB** menggunakan salah satu atau lebih dari keyword INTERSECT/UNION/EXCEPT.*

Query	<pre>(SELECT m.movie_id, m.title FROM movie m, production_country pc, country c WHERE m.movie_id = pc.movie_id AND pc.country_id = c.country_id AND c.country_name = "United States of America" AND m.vote_average &gt; 8 AND m.vote_count &gt; 1000)  UNION  (SELECT m.movie_id, m.title FROM movie m, production_country pc, country c WHERE m.movie_id = pc.movie_id AND pc.country_id = c.country_id AND c.country_name = "France" AND m.vote_average &lt; 5 AND m.vote_count &gt; 100);</pre>
SS Query dan Hasil	



```

MariaDB [movies]> (SELECT m.movie_id, m.title
-> FROM movie m, production_country pc, country c
-> WHERE m.movie_id = pc.movie_id
-> AND pc.country_id = c.country_id
-> AND c.country_name = "United States of America"
-> AND m.vote_average > 8
-> AND m.vote_count > 1000)
-> UNION
-> (SELECT m.movie_id, m.title
-> FROM movie m, production_country pc, country c
-> WHERE m.movie_id = pc.movie_id
-> AND pc.country_id = c.country_id
-> AND c.country_name = "France"
-> AND m.vote_average < 5
-> AND m.vote_count > 100);

```

movie_id	title
11	Star Wars
13	Forrest Gump
73	American History X
77	Memento
122	The Lord of the Rings: The Return of the King
155	The Dark Knight
207	Dead Poets Society
238	The Godfather
240	The Godfather: Part II
274	The Silence of the Lambs
278	The Shawshank Redemption
311	Once Upon a Time in America
335	C'era una volta il West
380	12 Angry Men
424	Schindler's List
429	Il buono, il brutto, il cattivo
497	The Green Mile
510	One Flew Over the Cuckoo's Nest
539	Psycho
550	Fight Club
629	The Usual Suspects
680	Pulp Fiction
694	The Shining
769	GoodFellas
807	Se7en
1891	The Empire Strikes Back
27205	Inception
157336	Interstellar
244786	Whiplash
8011	Highlander III: The Sorcerer
8645	The Happening
10002	Showgirls
56715	Just Visiting
86331	Q

34 rows in set (0.011 sec)

## 5. Maple Leaf Exclusion

Ms. D adalah seorang elitist yang hanya akan menonton film dengan rata - rata vote di atas 8.0. Dikarenakan pernah mengalami pengalaman buruk sebelumnya, Miss D melakukan blacklist pada seluruh film yang diproduksi dari Kanada ("Canada").

Bantu Ms. D menyusun daftar film yang sesuai dengan preferensinya dengan mengambil ID film dan judul yang memenuhi kriterianya!

[HINT] Untuk soal ini **WAJIB** menggunakan salah satu atau lebih dari keyword **INTERSECT/UNION/EXCEPT**.

Query	<pre> (SELECT m.movie_id, m.title FROM movie m, production_country pc, country c </pre>
-------	---

```
WHERE m.movie_id = pc.movie_id
AND pc.country_id = c.country_id
AND c.country_name != "Canada")
```

**INTERSECT**

```
(SELECT m.movie_id, m.title
FROM movie m
WHERE m.vote_average > 8);
```

## SS Query dan Hasil

```


MariaDB [movies]> (SELECT n.movie_id, n.title
-> FROM movie n, production_country pc, country c
-> WHERE n.movie_id = pc.movie_id
-> AND pc.country_id = c.country_id
-> AND c.country_name != 'Canada')
-> INTERSECT
-> (SELECT m.movie_id, m.title
-> FROM movie m
-> WHERE m.vote_average > 8);
+-----+-----+
| movie_id | title |
+-----+-----+
| 11 | Star Wars |
| 13 | Forrest Gump |
| 73 | American History X |
| 77 | Memento |
| 122 | The Lord of the Rings: The Return of the King |
| 128 | ものけね |
| 129 | 千と千尋の神隠し |
| 155 | The Dark Knight |
| 287 | Dead Poets Society |
| 238 | The Godfather |
| 240 | The Godfather: Part II |
| 274 | The Silence of the Lambs |
| 278 | The Shawshank Redemption |
| 284 | The Apartment |
| 311 | Once Upon a Time in America |
| 335 | C'era una volta il West |
| 346 | 七人の侍 |
| 389 | 12 Angry Men |
| 424 | Schindler's List |
| 429 | Il buono, il brutto, il cattivo |
| 497 | The Green Mile |
| 510 | One Flew Over the Cuckoo's Nest |
| 539 | Psycho |
| 550 | Fight Club |
| 598 | Cidade de Deus |
| 629 | The Usual Suspects |
| 680 | Pulp Fiction |
| 684 | The Shining |
| 769 | GoodFellas |
| 887 | Se7en |
| 1891 | The Empire Strikes Back |
| 3862 | Modern Times |
| 4935 | ハウスの動く屋 |
| 17603 | Anne of Green Gables |
| 27285 | Inception |
| 27322 | Love Jones |
| 30973 | The Visual Bible: The Gospel of John |
| 40963 | Little Big Top |
| 43807 | The Prisoner of Zenda |
| 69848 | One Man's Hero |
| 78373 | Dancer, Texas Pop. 81 |
| 80541 | There Goes My Baby |
| 89801 | Stiff Upper Lips |
| 107336 | Interstellar |
| 242373 | Guten Tag, Nanon |
| 244780 | Whiplash |
| 264644 | Room |
| 322745 | Counting |
| 346081 | Sardegna |
+-----+-----+
49 rows in set (0.067 sec)

MariaDB [movies]>

```

## 6. Tolong! Saya Ketagihan Moke! :(

Kelkelkelkelkel Cicerokel Moke! Ambatokel merupakan seorang mahasiswa Teknik Informatika yang sedang menjalankan ibadah puasa. Konon katanya dia merasa kecewa dengan dirinya sendiri karena dia seringkali membatalkan puasanya pada siang hari akibat lapar (anak muda sering menamakan fenomena

tersebut sebagai *moke/*). Suatu hari, dia merumuskan strategi untuk menahan hawa nafsunya hingga maghrib dengan menonton dua jenis film, yaitu film-film oleh aktor favoritnya **atau** film-film yang berhubungan dengan makanan. Dia ingin melihat daftar film yang dimainkan oleh **Lebron James** atau dimainkan oleh **Kanye West**. Selain itu, dia juga ingin melihat daftar film yang **mengandung kata *food*** dalam *keyword*-nya. Karena dia ingin mencari distraksi yang membantu dia menahan hawa nafsu hingga maghrib, film-film tersebut harus memiliki ***runtime* lebih dari 90 menit**. Selain itu, dia merupakan penggemar film mahal dan berkelas sehingga dia hanya ingin menonton **film dengan keyword *food* yang memiliki *budget* lebih dari sejuta (1000000)**. Film yang dimainkan oleh aktor favoritnya **tidak perlu** memiliki *budget* lebih dari sejuta karena dia yakin film tersebut pasti sudah mahal dan berkelas. Bantulah Kelkelkelkelkel Cicerokel Moke/ Ambatokel mendapat daftar ***title, runtime, dan budget*** film yang mampu membantunya menghindari *moke/* dan melewati ramadhan! Semoga kalian mendapat pahala yang lebih atas bantuan kalian .

*HINT: keyword UNION dapat membantu kalian dalam soal ini.*

Query	<pre> (SELECT M.title, M.runtime, M.budget FROM movie M, movie_keywords MK, keyword K WHERE M.runtime &gt; 90 AND M.movie_id = MK.movie_id AND MK.keyword_id = K.keyword_id AND keyword_name LIKE "%food%" AND M.budget &gt; 1000000) UNION (SELECT M.title, M.runtime, M.budget FROM movie M, movie_cast MC, person P WHERE M.runtime &gt; 90 AND M.movie_id = MC.movie_id AND MC.person_id = P.person_id AND (P.person_name = "Lebron James" OR P.person_name = "Kanye West")); </pre>
SS Query dan Hasil	

```
MarlaDB [movies]> (SELECT M.title, M.runtime, M.budget
-> FROM movie M, movie_keywords MK, keyword K
-> WHERE M.runtime > 90
-> AND M.movie_id = MK.movie_id
-> AND MK.keyword_id = K.keyword_id
-> AND keyword_name LIKE "afoods"
-> AND M.budget > 1000000)
->
-> UNION
-> (SELECT M.title, M.runtime, M.budget
-> FROM movie M, movie_cast MC, person P
-> WHERE M.runtime > 90
-> AND M.movie_id = MC.movie_id
-> AND MC.person_id = P.person_id
-> AND (P.person_name = "Lebron James"
-> OR P.person_name = "Kanye West"));
-----
| title | runtime | budget |
-----
| The Terminal | 128 | 60000000 |
| Rambo III | 102 | 63000000 |
| Little Shop of Horrors | 94 | 25000000 |
| Home Fries | 93 | 15000000 |
| Les petits nouchoirs | 154 | 25000000 |
| Cloudy with a Chance of Meatballs 2 | 95 | 78000000 |
| Dave Chappelle's Block Party | 100 | 3000000 |
| Anchorman 2: The Legend Continues | 119 | 50000000 |
| Trainwreck | 125 | 35000000 |
| Zoolander 2 | 100 | 50000000 |
-----
10 rows in set (0.258 sec)

MarlaDB [movies]>
```

## 7. Harta yang Paling Berharga adalah Keluarga

Jumbilo adalah seorang produser film yang sangat mencintai tema keluarga dan *romance*. Ia percaya bahwa nilai-nilai kekeluargaan serta cinta adalah elemen paling penting dalam kehidupan, dan selalu berusaha menyampaikan pesan tersebut dalam setiap film yang ia produksi. Namun, belakangan ini, ia merasa penasaran karena jarang menemukan produser lain yang memiliki pandangan serupa.

Untuk mengatasi rasa penasarannya, Jumbilo ingin mencari produser lain yang pernah bekerja di film yang memiliki tagline dengan kata *family* atau *love* di dalamnya. Dengan begitu, ia bisa berdiskusi dan berbagi pandangan dengan orang-orang yang juga memahami keindahan keluarga dan cinta dalam film. Namun, Jumbilo tidak ingin berhubungan dengan produser yang terlibat dalam film bergenre Horror atau memiliki tagline yang mengandung kata *fear*, karena menurutnya film semacam itu kurang mencerminkan kehangatan dan kedekatan emosional yang ia cari.

Bantu Jumbilo menemukan produser lain yang memenuhi kriterianya agar ia bisa berdiskusi dan berbagi pandangan dengan orang-orang yang juga menuangkan nilai-nilai keluarga dan cinta dalam film yang mereka buat!

*[HINT] Gunakan salah satu atau lebih dari keyword INTERSECT/ UNION/ EXCEPT jika diperlukan (tidak wajib).*

Query	<pre> ((SELECT P.person_id, P.person_name FROM movie M, movie_crew MC, person P WHERE M.tagline LIKE "%family%" AND M.movie_id = MC.movie_id AND MC.job = "Producer" AND MC.person_id = P.person_id) UNION (SELECT P.person_id, P.person_name FROM movie M, movie_crew MC, person P WHERE M.tagline LIKE "%love%" AND M.movie_id = MC.movie_id AND MC.job = "Producer" AND MC.person_id = P.person_id)) EXCEPT ((SELECT P.person_id, P.person_name FROM movie M, movie_crew MC, person P WHERE M.tagline LIKE "%fear%" AND M.movie_id = MC.movie_id AND MC.job = "Producer" AND MC.person_id = P.person_id) UNION (SELECT P.person_id, P.person_name FROM movie M, movie_crew MC, person P, movie_genres MG, genre G WHERE M.movie_id = MG.movie_id AND MG.genre_id = G.genre_id AND G.genre_name = "Horror" AND M.movie_id = MC.movie_id AND MC.job = "Producer" AND MC.person_id = P.person_id)); </pre>
SS Query dan Hasil	

```

MariaDB [movies]> ((SELECT P.person_id, P.person_name
-> FROM movie M, movie_crew MC, person P
-> WHERE M.tagline LIKE "Kantiky"
-> AND M.movie_id = MC.movie_id
-> AND MC.job = "Producer"
-> AND MC.person_id = P.person_id)
-> UNION
-> (SELECT P.person_id, P.person_name
-> FROM movie M, movie_crew MC, person P
-> WHERE M.tagline LIKE "KloveR"
-> AND M.movie_id = MC.movie_id
-> AND MC.job = "Producer"
-> AND MC.person_id = P.person_id))
-> EXCEPT
-> ((SELECT P.person_id, P.person_name
-> FROM movie M, movie_crew MC, person P
-> WHERE M.tagline LIKE "KfearK"
-> AND M.movie_id = MC.movie_id
-> AND MC.job = "Producer"
-> AND MC.person_id = P.person_id)
-> UNION
-> (SELECT P.person_id, P.person_name
-> FROM movie M, movie_crew MC, person P, movie_genres MG, genre G
-> WHERE M.movie_id = MG.movie_id
-> AND MG.genre_id = G.genre_id
-> AND G.genre_name = "Horror"
-> AND M.movie_id = MC.movie_id
-> AND MC.job = "Producer"
-> AND MC.person_id = P.person_id));
+-----+-----+
| person_id | person_name |
+-----+-----+
| 5741      | Matt Groening |
| 3388      | James L. Brooks |
| 57572     | Richard Sakat |
| 162931    | Al Jean |
| 165810    | Mike Scully |
| 193307    | Jay Kleckner |
| 1128337   | Peter Gave |
| 1128338   | Amanda Moshay |
| 1128339   | Felicia Nalivansky |
| 1128340   | Matt Orefice |
+-----+-----+

1141683 | Theodore Melfi
60963  | Alison Greenspan
43147  | Ed Guiney
1375896 | David Gross
1041394 | Anthony McCarten
1129513 | Chris Clark
1486231 | Quentin Curtis
51684  | Mehcad Brooks
55084  | Sean Faris
1502483 | Mark E. Walker
+-----+-----+
474 rows in set (0.053 sec)

```

## 8. Hobi yang Aneh

Seorang *database administrator*, John, adalah orang dengan hobi yang aneh.

Dalam hobinya, John ingin kamu membantunya mencari data dengan spesifikasi:

1. Sebuah *movie* dengan *movie\_id* ganjil, *release\_date* sebelum tahun 2000, dan *vote\_average* lebih besar dari 7,
2. Dengan bahasa selain *English* yang memiliki 2 role (Original dan Spoken).

Bantu John menampilkan **movie\_id**, **title**, **release\_date**, **vote\_average**, **language\_name**, dan 2 **language\_role** tersebut.

[HINT]: Fungsi-fungsi dibawah ini mungkin dapat membantu kalian mengerjakan soal:

1. MOD(x, y) untuk mendapatkan nilai *modulo* dari x terhadap y selayaknya bentuk matematika (x mod (y)). Fungsi ini menghasilkan nilai yang sama dengan operator *modulo* (%) pada x % y

Query	<b>SELECT</b> M.movie_id, M.title, M.release_date, M.vote_average, L.language_name, L1.language_role, L2.language_role
-------	---

	<pre> <b>FROM</b> movie M, movie_languages ML1, movie_languages ML2, language L, language_role L1, language_role L2  <b>WHERE</b> MOD(M.movie_id, 2)  <b>AND</b> M.release_date &lt; "2000-01-01"  <b>AND</b> M.vote_average &gt; 7  <b>AND</b> M.movie_id = ML1.movie_id <b>AND</b> M.movie_id = ML2.movie_id  <b>AND</b> ML1.language_id = L.language_id <b>AND</b> ML2.language_id = L.language_id  <b>AND</b> L.language_code != "en"  <b>AND</b> ML1.language_role_id = L1.role_id <b>AND</b> ML2.language_role_id = L2.role_id  <b>AND</b> L1.language_role = "Original" <b>AND</b> L2.language_role = "Spoken"  <b>AND</b> L1.language_role != L2.language_role; </pre>
--	--

## SS Query dan Hasil

```

MariaDB [movies]> SELECT M.movie_id, M.title, M.release_date, M.vote_average, L.language_name, L1.language_role, L2.language_role
-> FROM movie M, movie_languages ML1, movie_languages ML2, language L, language_role L1, language_role L2
->
-> WHERE MOD(M.movie_id, 2)
-> AND M.release_date < "2000-01-01"
-> AND M.vote_average > 7
-> AND M.movie_id = ML1.movie_id
-> AND M.movie_id = ML2.movie_id
-> AND ML1.language_id = L.language_id
-> AND ML2.language_id = L.language_id
-> AND L.language_code != "en"
-> AND ML1.language_role_id = L1.role_id
-> AND ML2.language_role_id = L2.role_id
-> AND L1.language_role = "Original"
-> AND L2.language_role = "Spoken"
-> AND L1.language_role != L2.language_role;

```

movie_id	title	release_date	vote_average	language_name	language_role	language_role
149	アキラ	1988-07-16	7.80	Japanese	Original	Spoken
309	Festen	1998-05-20	7.70	Dansk	Original	Spoken
387	Das Boot	1981-09-16	7.90	Deutsch	Original	Spoken
391	Per un pugno di dollari	1964-09-12	7.60	Italiano	Original	Spoken
429	Il buono, il brutto, il cattivo	1966-12-23	8.10	Italiano	Original	Spoken
593	Соларис	1972-03-20	7.70	Russian	Original	Spoken
905	Die Büchse der Pandora	1929-01-30	7.60	Deutsch	Original	Spoken
11687	Les visiteurs	1993-01-27	7.10	French	Original	Spoken
12207	醉拳二	1994-02-03	7.20	Chinese	Original	Spoken
13889	まあだだよ	1993-04-17	7.50	Japanese	Original	Spoken
17139	Karakter	1997-04-17	7.70	Nederlands	Original	Spoken
36095	キューア	1997-11-06	7.40	Japanese	Original	Spoken
48035	Ordet	1955-01-09	7.80	Dansk	Original	Spoken

13 rows in set (0.007 sec)

## Pembagian Tugas

Nomor Soal	Driver (NIM)	Navigator (NIM)
1	13523004	13523104
2	13523004	13523104

3	13523004	13523104
4	13523004	13523104
5	13523104	13523004
6	13523104	13523004
7	13523104	13523004
8	13523104	13523004