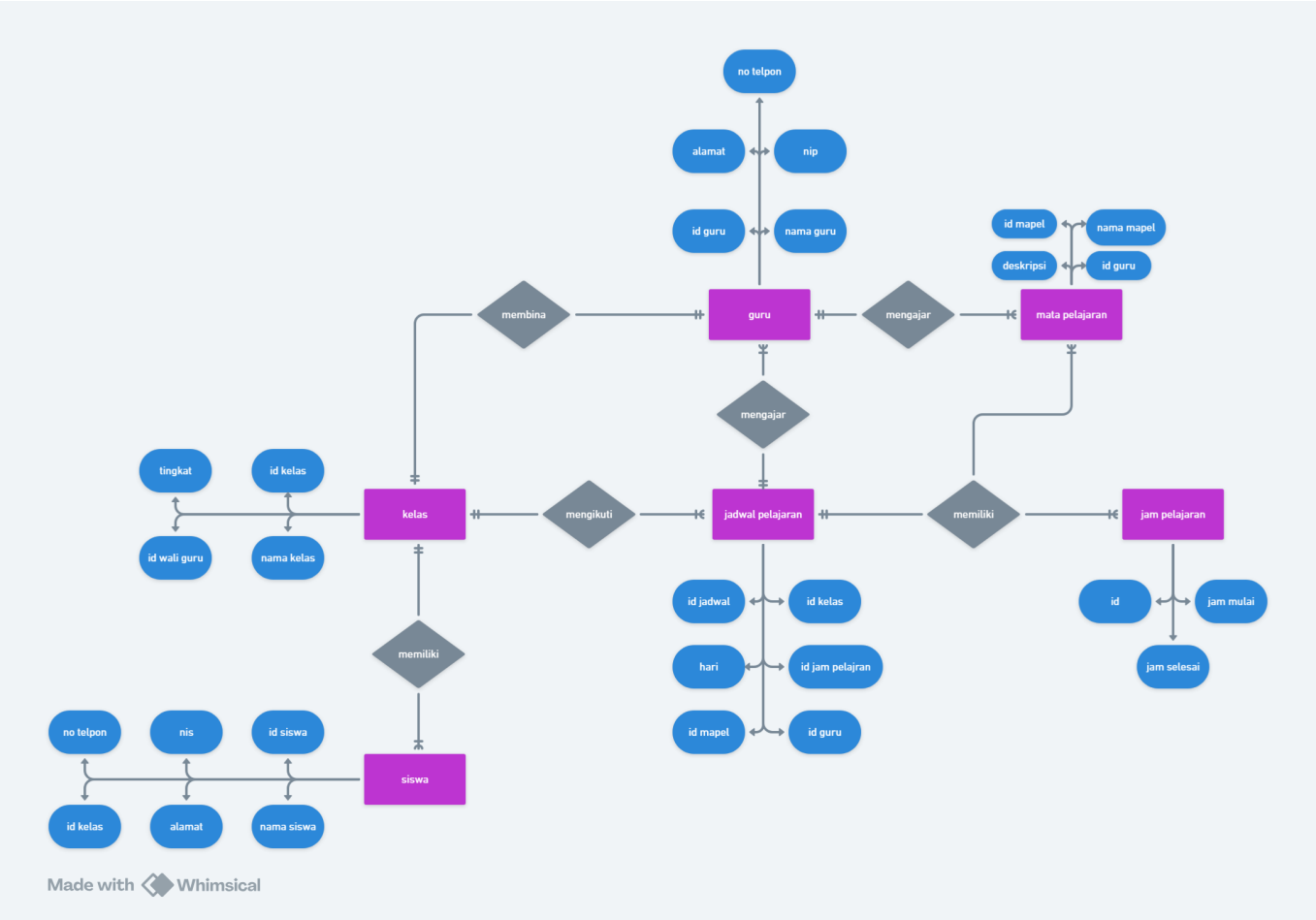
Revisi Ujian

Nama	Score	Peran
Ahmad Anugrah Satya	3	Kerja Soal 2 Query 2
Nur Rahmat Ramadan	3	Membantu Pengerjaan
Muh Ardiansya	3	Kerja Soal 2 Query 1
Raihan Juli Saputra	3	Membantu Pengerjaan

Soal

1. Setiap kelompok merancang database di MySQL dari hasil perencanaan ERD-nya masing-masing. Di dalam database tersebut wajib menjadikan tabel berelasi, dengan menambah foreign key.
2. Selanjutnya tampilkan datanya secara kontekstual dengan menggunakan query relasi, group by, dan having secara bersamaan dalam satu query. Buatlah minimal sebanyak 2 contoh.
3. Sertakan pula penjelasan dan analisis kalian pada soal nomor 1 dan 2.

ERD (Entity Relationship Diagram)



Penjelasan ERD

- **Guru ke Mata Pelajaran**
  - Kardinalitas: 1-N
  - Penjelasan: Satu guru (guru) dapat mengajar beberapa mata pelajaran (mata_pelajaran), tetapi setiap mata pelajaran hanya memiliki satu guru yang mengajar.
- **Guru ke Kelas**
  - Kardinalitas: 1-1
  - Penjelasan: Satu guru (guru) hanya bisa menjadi 1 walikelas (kelas), dan setiap kelas (kelas) hanya memiliki 1 walikelas
- **Kelas ke Siswa**
  - Kardinalitas: 1-N
  - Penjelasan: Satu kelas (kelas) dapat berisi banyak siswa (siswa), tetapi setiap siswa hanya terdaftar di satu kelas.
- **Kelas ke Jadwal Pelajaran**
  - Kardinalitas: 1-N
  - Penjelasan: Satu kelas dapat memiliki beberapa jadwal pelajaran (jadwal_pelajaran), tetapi setiap jadwal pelajaran hanya berkaitan dengan satu kelas.
- **Mata Pelajaran ke Jadwal Pelajaran (Melalui Guru)**
  - Kardinalitas: N-N

    - Penjelasan: Melalui tabel guru anyak jadwal pelajaran (jadwal_pelajaran) dapat berhubungan dengan satu mata pelajaran (mata_pelajaran).
- **Jam Pelajaran ke Jadwal Pelajaran**
    - Kardinalitas: N-1
    - Penjelasan: Satu jadwal bisa memiliki banyak jam pelajaran.
- **Jadwal Pelajaran ke Siswa (melalui kelas)**
    - Kardinalitas: N-N
    - Penjelasan: Tabel guru menghubungkan jadwal_pelajaran dan siswa dengan kardinalitas banyak ke banyak, memungkinkan satu jadwal pelajaran diikuti oleh banyak siswa, dan satu siswa mengikuti banyak jadwal pelajaran.

Tabel Yang Digunakan (6 Tabel)

Tabel Jam_Pelajaran

```
MariaDB [jadwal_pelajaran]> select * from jam_pelajaran
    -> ;
+----+-----------+-------------+
| id | jam_mulai | jam_selesai |
+----+-----------+-------------+
|  1 | 07:15:00  | 08:00:00    |
|  2 | 08:00:00  | 08:45:00    |
|  3 | 08:45:00  | 09:30:00    |
|  4 | 09:30:00  | 10:15:00    |
|  5 | 10:15:00  | 10:30:00    |
|  6 | 10:30:00  | 11:15:00    |
|  7 | 11:15:00  | 12:00:00    |
|  8 | 12:00:00  | 13:00:00    |
|  9 | 13:00:00  | 13:45:00    |
| 10 | 13:45:00  | 14:30:00    |
| 11 | 14:30:00  | 15:15:00    |
| 12 | 15:15:00  | 16:00:00    |
+----+-----------+-------------+
12 rows in set (0.000 sec)
```

Tabel Mata Pelajaran

```
MariaDB [jadwal_pelajaran]> select * from mata_pelajaran;
+----------+-----------------------------------+-----------+---------+
| id_mapel | nama_mapel                        | deskripsi | id_guru |
+----------+-----------------------------------+-----------+---------+
|        1 | Matematika                        |           |       1 |
|        2 | Bahasa Indonesia                  |           |       3 |
|        3 | Basis Data                        |           |       2 |
|        4 | Pemograman Web                    |           |       2 |
|        5 | Produk Kreatif dan Kewirausahaan  |           |       9 |
|        6 | PPKn                              |           |       5 |
|        7 | Pemodelan Perangkat Lunak         |           |       6 |
|        8 | PBO                               |           |       7 |
|        9 | Pendidikan Agama Islam            |           |       4 |
|       10 | Bahasa Inggris                    |           |       8 |
|       11 | PJOK                              |           |      10 |
+----------+-----------------------------------+-----------+---------+
```

Tabel Jadwal Pelajaran

```
MariaDB [jadwal_pelajaran2]> select * from jadwal_pelajaran;
+-----------+----------+--------+------------------+----------+---------+
| id_jadwal | id_kelas | hari   | id_jam_pelajaran | id_mapel | id_guru |
+-----------+----------+--------+------------------+----------+---------+
|         1 |        3 | Senin  |                1 |        4 |       2 |
|         2 |        3 | Senin  |                2 |        5 |       9 |
|         3 |        3 | Selasa |                3 |        9 |       4 |
|         4 |        3 | Selasa |                4 |        1 |       1 |
|         5 |        3 | Selasa |                5 |        5 |       9 |
|         6 |        4 | Senin  |                1 |        4 |       2 |
|         7 |        4 | Senin  |                2 |        2 |       3 |
|         8 |        4 | Selasa |                8 |        1 |       1 |
|         9 |        4 | Selasa |                9 |        5 |       9 |
|        10 |        4 | Selasa |               10 |        7 |       6 |
+-----------+----------+--------+------------------+----------+---------+
10 rows in set (0.001 sec)
```

Tabel Guru

```
MariaDB [jadwal_pelajaran]> select * from guru;
+---------+-----------------------------------+-------+----------+------------+
| id_guru | nama_guru                         | nip   | alamat   | no_telepon |
+---------+-----------------------------------+-------+----------+------------+
|       1 | A.TENRITTE, S.Pd                  | 12345 | Alamat A | 0811111111 |
|       2 | Ibrahim Mallombassang, S.Pd       | 54321 | Alamat B | 0811111112 |
|       3 | DRA. WAROYAH, M.Pd                | 56789 | Alamat C | 0811111113 |
|       4 | Moh.Saleh Burhan, S.Pd, M.Pdi     | 98765 | Alamat D | 0811111114 |
|       5 | Dra. Anis Dwi Kartika Wati        | 11122 | Alamat E | 0811111115 |
|       6 | Adrianty. S.Kom                   | 22333 | Alamat F | 0811111116 |
|       7 | Muhammad Fajar Shiddiq AD, S.Kom  | 33444 | Alamat G | 0811111117 |
|       8 | Hasnidar, S.Ag                    | 44555 | Alamat H | 0811111118 |
|       9 | Andi Asrawati Sut, S.TP, S.Pd     | 55666 | Alamat I | 0811111119 |
|      10 | Andi Muh.Agussalim, S.Pd          | 66777 | Alamat J | 0811111120 |
|      11 | Hasannuddin S.Pd                  | NULL  | NULL     | NULL       |
|      12 | Aswar S.Pd                        | NULL  | NULL     | NULL       |
|      13 | Ardiansyah, S.Pd                  | NULL  | NULL     | NULL       |
+---------+-----------------------------------+-------+----------+------------+
```

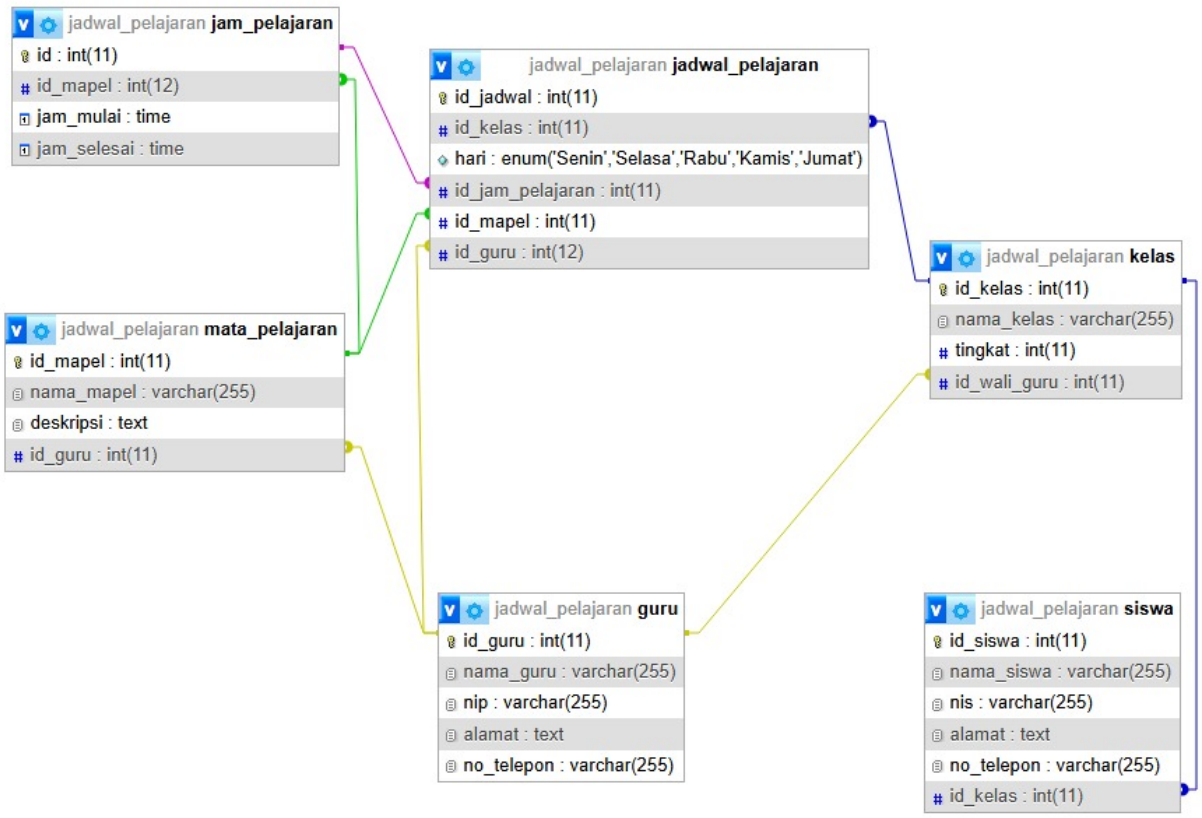
Tabel Kelas

```
MariaDB [jadwal_pelajaran]> select * from kelas;
+----------+------------+---------+--------------+
| id_kelas | nama_kelas | tingkat | id_wali_guru |
+----------+------------+---------+--------------+
|        1 | PPLG       |      10 |            2 |
|        2 | PPLG       |      11 |            4 |
|        3 | RPL 1      |      12 |            1 |
|        4 | RPL 2      |      12 |            8 |
|        5 | MPLB 1     |      10 |            5 |
|        6 | MPLB 1     |      11 |            6 |
|        7 | AP 1       |      12 |            7 |
|        8 | AK 1       |      10 |            3 |
|        9 | AK 1       |      11 |            9 |
|       10 | AK 1       |      12 |           10 |
|       11 | PS 1       |      10 |           11 |
|       12 | PS 1       |      11 |           12 |
|       13 | PS 1       |      12 |           13 |
+----------+------------+---------+--------------+
```

Tabel Siswa

```
MariaDB [jadwal_pelajaran]> select * from siswa;
+----------+-----------------------------+--------+-----------+------------+----------+
| id_siswa | nama_siswa                  | nis    | alamat    | no_telepon | id_kelas |
+----------+-----------------------------+--------+-----------+------------+----------+
|        1 | ABD.Rahman                  | NIS001 | Alamat 1  | 0812111111 |        1 |
|        2 | Ahmad Anugrah Satya         | NIS002 | Alamat 2  | 0812111112 |        1 |
|        3 | Ahsan Putra Ahyar           | NIS003 | Alamat 3  | 0812111113 |        2 |
|        4 | Andi Ashadela Maharani Anil | NIS004 | Alamat 4  | 0812111114 |        2 |
|        5 | Andi Muh Raihan Alkawsar    | NIS005 | Alamat 5  | 0812111115 |        3 |
|        6 | Chairil Abizali             | NIS006 | Alamat 6  | 0812111116 |        3 |
|        7 | Fachri Ramadhan             | NIS007 | Alamat 7  | 0812111117 |        4 |
|        8 | Fatsa Akhwani               | NIS008 | Alamat 8  | 0812111118 |        4 |
|        9 | Jordan                      | NIS009 | Alamat 9  | 0812111119 |        5 |
|       10 | Hansar                      | NIS010 | Alamat 10 | 0812111120 |        5 |
+----------+-----------------------------+--------+-----------+------------+----------+
```

Hasil Relasi Pada Database

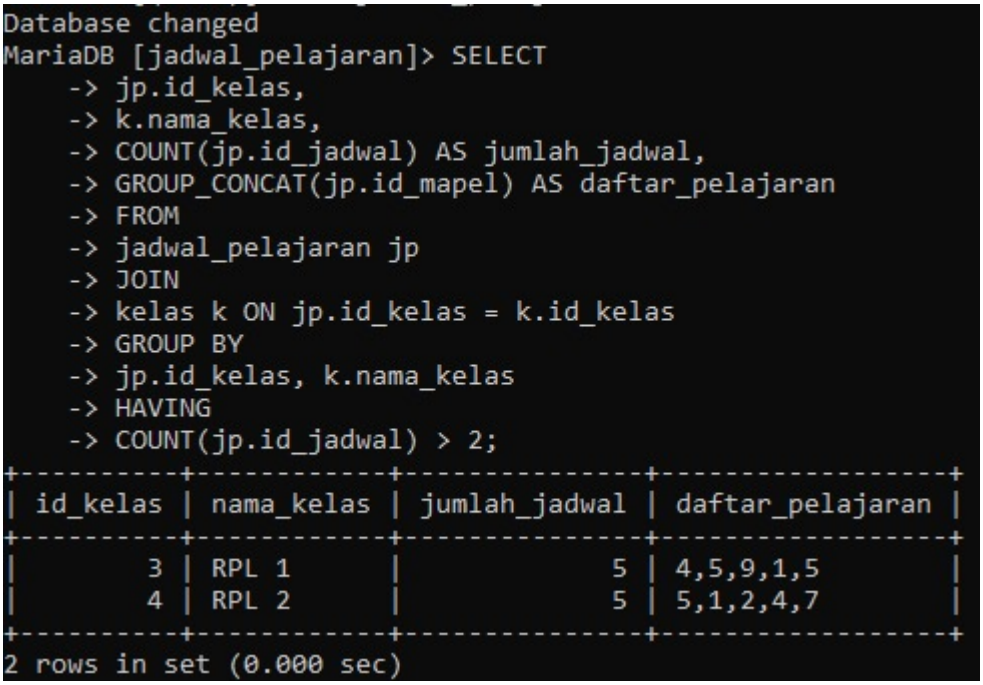


Query Relasi, Group By and Having

Contoh Query 1

```sql
SELECT
        jp.id_kelas,
        k.nama_kelas,
        COUNT(jp.id_jadwal) AS jumlah_jadwal,
        GROUP_CONCAT(jp.id_mapel) AS daftar_pelajaran
        FROM
    jadwal_pelajaran jp
        JOIN
    kelas k ON jp.id_kelas = k.id_kelas
        GROUP BY
    jp.id_kelas, k.nama_kelas
        HAVING
    COUNT(jp.id_jadwal) > 2;
```

Hasil Query 1



```
Database changed
MariaDB [jadwal_pelajaran]> SELECT
    -> jp.id_kelas,
    -> k.nama_kelas,
    -> COUNT(jp.id_jadwal) AS jumlah_jadwal,
    -> GROUP_CONCAT(jp.id_mapel) AS daftar_pelajaran
    -> FROM
    -> jadwal_pelajaran jp
    -> JOIN
    -> kelas k ON jp.id_kelas = k.id_kelas
    -> GROUP BY
    -> jp.id_kelas, k.nama_kelas
    -> HAVING
    -> COUNT(jp.id_jadwal) > 2;
+----------+------------+---------------+------------------+
| id_kelas | nama_kelas | jumlah_jadwal | daftar_pelajaran |
+----------+------------+---------------+------------------+
|        3 | RPL 1      |             5 | 4,5,9,1,5        |
|        4 | RPL 2      |             5 | 5,1,2,4,7        |
+----------+------------+---------------+------------------+
2 rows in set (0.000 sec)
```

Tujuan Query

- **Menampilkan informasi kelas** berdasarkan jadwal pelajaran yang dimiliki.
- Menampilkan **ID kelas** (`id_kelas`), **nama kelas** (`nama_kelas`), jumlah jadwal pelajaran yang dimiliki setiap kelas, dan daftar ID mata pelajaran yang diajarkan di setiap kelas.
- Menyaring hanya kelas yang memiliki **lebih dari 2 jadwal pelajaran**.

Cara Relasi

Relasi ini menggunakan operasi **JOIN** antara tabel `jadwal_pelajaran` (`jp`) dan tabel `kelas` (`k`) melalui kolom `id_kelas`. yang dimana kita merelasikannya dikarekan kedua table tersebut memiliki

konteks relasi yaitu satu kelas bisa mengikuti banyak jadwal_pelajaran dan satu jadwal_pelajaran bisa diikuti banyak kelas

Cara Agregasi

Fungsi agregasi yang bisa kita gunakan untuk menyaring hanya kelas yang memiliki **lebih dari 2 jadwal pelajaran** dan fungsi tersebut adalah `count` dan `group_count` agar kita bisa menyaring nama kelas yang memiliki 2 jadwal. yang fungsi agregasi tersebut berfungsi

- `count` : Menghitung jumlah seluruh baris dalam sebuah tabel atau grup, terlepas dari nilai kolomnya. dan kolom yang menggunakan fungsi tersebut `id_jadwal` dalam tabel `jadwal_pelajaran`
- `group_count` : Menggabungkan nilai-nilai dari kolom yang dipilih, dan hasilnya akan dipisahkan oleh pemisah yang ditentukan (default adalah koma `,`). dan kolom yang menggunakan fungsi tersebut `id_mapel` dalam tabel `jadwal_pelajaran`.

Penjelasan

- **jadwal_pelajaran**: Berisi data jadwal pelajaran, termasuk ID jadwal, ID kelas, hari, waktu mulai, dan waktu selesai.
- **kelas**: Berisi data tentang kelas, seperti ID kelas dan nama kelas.
  Kedua tabel ini direlasikan untuk menyimpan dan mengelola jadwal pelajaran untuk setiap kelas. Sebagai seorang administrator database, Anda diminta untuk mencari kelas yang memiliki lebih dari 2 jadwal pelajaran. Tugas ini bisa diselesaikan dengan mengelompokkan data berdasarkan kelas, lalu **menghitung jumlah jadwal tiap kelas menggunakan fungsi agregasi COUNT, serta menyaring kelas yang memiliki lebih dari 2 jadwal menggunakan klausa HAVING**.

Analisis

- `SELECT`:Digunakan untuk menampilkan kolom yang ingin diambil dari tabel
- `jp.id_kelas`: Mengambil **ID kelas** dari tabel `jadwal_pelajaran` (`jp`). Ini akan digunakan untuk mengelompokkan data berdasarkan kelas.
- `k.nama_kelas`: Mengambil **nama kelas** dari tabel `kelas` (`k`) untuk memberikan informasi lebih deskriptif terkait kelas yang dimaksud.
- `COUNT(jp.id_jadwal) AS jumlah_jadwal`: Menggunakan **fungsi agregat** `COUNT` untuk menghitung berapa banyak jadwal (`id_jadwal`) yang ada untuk masing-masing kelas. Hasilnya diberi alias `jumlah_jadwal`.
- `GROUP_CONCAT(jp.id_mapel) AS daftar_pelajaran`: Menggunakan **fungsi agregat** `GROUP_CONCAT` untuk menggabungkan semua ID mata pelajaran (`id_mapel`) yang terkait dengan kelas tersebut dalam satu string. Hasilnya diberi alias `daftar_pelajaran`
- `jadwal_pelajaran jp`: Data utama diambil dari tabel `jadwal_pelajaran` yang diberi alias `jp`. Tabel ini berisi informasi tentang jadwal pelajaran yang dihubungkan dengan kelas dan mata pelajaran.

- `JOIN kelas k`: Melakukan **join** antara tabel `jadwal_pelajaran` dan tabel `kelas` untuk menghubungkan informasi jadwal dengan nama kelas.
- `ON jp.id_kelas = k.id_kelas`: Kondisi **join** berdasarkan kecocokan kolom `id_kelas` antara kedua tabel. Ini memastikan bahwa informasi jadwal pelajaran yang ditampilkan sesuai dengan kelas yang ada.
- `GROUP BY jp.id_kelas, k.nama_kelas`: Mengelompokkan data berdasarkan **ID kelas** dan **nama kelas**. Ini memastikan bahwa setiap kelas hanya muncul satu kali dalam hasil, dengan informasi jumlah jadwal dan daftar mata pelajaran untuk setiap kelas.
- `HAVING COUNT(jp.id_jadwal) > 2`: Kondisi **HAVING** digunakan untuk menyaring grup yang memiliki lebih dari 2 jadwal (`id_jadwal`). Dengan demikian, hanya kelas yang memiliki lebih dari dua jadwal yang akan ditampilkan dalam hasil query.

Contoh Query 2

SQL

```sql
SELECT
        g.id_guru,
    g.nama_guru, -- Kolom dari tabel guru untuk menampilkan nama guru
        COUNT(mp.id_mapel) AS jumlah_mapel
        FROM
        mata_pelajaran mp
        JOIN
        guru g ON mp.id_guru = g.id_guru -- Relasi dengan tabel guru
        GROUP BY
        g.id_guru, g.nama_guru
        HAVING
        COUNT(mp.id_mapel) > 1;
```

Hasil Query 2

```
MariaDB [jadwal_pelajaran]> SELECT
    ->     g.id_guru,
    ->     g.nama_guru, -- Kolom dari tabel guru untuk menampilkan nama guru
    ->     COUNT(mp.id_mapel) AS jumlah_mapel
    -> FROM
    ->     mata_pelajaran mp
    -> JOIN
    ->     guru g ON mp.id_guru = g.id_guru -- Relasi dengan tabel guru
    -> GROUP BY
    ->     g.id_guru, g.nama_guru
    -> HAVING
    ->     COUNT(mp.id_mapel) > 1;
+---------+---------------------------+--------------+
| id_guru | nama_guru                 | jumlah_mapel |
+---------+---------------------------+--------------+
|       2 | Ibrahim Mallombassang, S.Pd |            2 |
+---------+---------------------------+--------------+
1 row in set (0.001 sec)
```

Tujuan Query

- **Mengambil data guru yang mengajar lebih dari satu mata pelajaran**, dengan informasi ID guru, nama guru, dan jumlah mata pelajaran yang diajarkan.
- **Memfilter** hasilnya untuk menampilkan hanya guru-guru yang mengajar lebih dari satu mata pelajaran.

Cara Relasi

Relasi ini menggunakan operasi **JOIN** antara tabel `mata_pelajaran` (`mp`) dan tabel `guru` (`g`) melalui kolom `id_guru`. yang dimana kita merelasikannya dikarekan kedua table tersebut memiliki konteks relasi yaitu satu guru bisa mengajarkan banyak mata_pelajaran

Cara Agregasi

Fungsi agregasi yang bisa kita gunakan untuk mengetahi guru yang mengajar lebih dari satu mata pelajaran dan fungsi tersebut adalah `count` yang berfungsi sebagai berikut

- `count` : Menghitung jumlah seluruh baris dalam sebuah tabel atau grup, terlepas dari nilai kolomnya. dan kolom yang menggunakan fungsi tersebut `id_mapel` dalam tabel `mata_pelajara`.

Penjelasan

- **mata_pelajaran**: Berisi data mata pelajaran, seperti nama mata pelajaran, ID guru (FK), dan deskripsi mata pelajaran.

- **guru**: Berisi data guru, seperti nama guru, NIP, dan kontak.
  Kedua tabel ini direlasikan untuk mencatat mata pelajaran yang diajarkan oleh setiap guru. Sebagai seorang administrator database, Anda diminta untuk mencari guru yang mengajar lebih dari satu mata pelajaran. Masalah ini dapat diselesaikan dengan mengelompokkan data berdasarkan guru, **menghitung jumlah mata pelajaran tiap guru dengan COUNT, serta menyaring guru yang mengajar lebih dari satu mata pelajaran menggunakan klausa HAVING**.

Analisis

- `SELECT`:Digunakan untuk menampilkan kolom tertentu.
- `g.id_guru`: Mengambil **ID guru** dari tabel `guru` (`g`). Ini digunakan untuk mengidentifikasi setiap guru yang terlibat dalam mengajar mata pelajaran.
- `g.nama_guru`: Mengambil **nama guru** dari tabel `guru` untuk menampilkan nama guru yang terhubung dengan mata pelajaran.
- `COUNT(mp.id_mapel) AS jumlah_mapel`: Menggunakan **fungsi agregat** `COUNT` untuk menghitung jumlah mata pelajaran (`id_mapel`) yang diajarkan oleh setiap guru. Hasilnya diberi alias `jumlah_mapel` untuk mempermudah pemahaman.
- `mata_pelajaran mp`: Data utama diambil dari tabel `mata_pelajaran` yang diberi alias `mp`. Tabel ini berisi informasi tentang mata pelajaran dan guru yang mengajar.
- `JOIN guru g`: Melakukan **join** antara tabel `mata_pelajaran` (`mp`) dan tabel `guru` (`g`). Ini digunakan untuk menghubungkan setiap mata pelajaran dengan guru yang mengajarkannya.
- `ON mp.id_guru = g.id_guru`: Kondisi **join** dilakukan dengan mencocokkan kolom `id_guru` di kedua tabel. Ini memastikan setiap mata pelajaran terhubung dengan guru yang sesuai.
- `GROUP BY g.id_guru, g.nama_guru`: Mengelompokkan data berdasarkan **ID guru** dan **nama guru**. Ini memastikan bahwa setiap grup mewakili seorang guru, dan kita bisa menghitung jumlah mata pelajaran yang diajarkan oleh setiap guru.
- `HAVING COUNT(mp.id_mapel) > 1`: Kondisi **HAVING** digunakan untuk menyaring hasil agregasi. Di sini, hanya guru yang mengajar **lebih dari satu mata pelajaran** yang akan muncul dalam hasil query.

Contoh Query 3

SQL

```sql
SELECT
    g.nama_guru,
    jp.hari,
    jp2.jam_mulai,
    jp2.jam_selesai,
    COUNT(DISTINCT k.id_kelas) AS total_kelas_terlibat,
    GROUP_CONCAT(DISTINCT k.nama_kelas) AS kelas_terlibat
FROM
```

```
    jadwal_pelajaran jp
JOIN
    kelas k ON jp.id_kelas = k.id_kelas
JOIN
    guru g ON jp.id_guru = g.id_guru
JOIN
    jam_pelajaran jp2 ON jp.id_jam_pelajaran = jp2.id
JOIN
    jadwal_pelajaran jp_conflict ON jp.id_guru = jp_conflict.id_guru
JOIN
    kelas k_conflict ON jp_conflict.id_kelas = k_conflict.id_kelas
JOIN
    jam_pelajaran jp2_conflict ON jp_conflict.id_jam_pelajaran = jp2_conflict.id
WHERE
    jp.hari = jp_conflict.hari
    AND jp2.jam_mulai = jp2_conflict.jam_mulai
    AND jp2.jam_selesai = jp2_conflict.jam_selesai
    AND jp.id_kelas <> jp_conflict.id_kelas
GROUP BY
    g.nama_guru, jp.hari, jp2.jam_mulai, jp2.jam_selesai;
```

Hasil Query

```
MariaDB [jadwal_pelajaran2]> SELECT
    ->     g.nama_guru,
    ->     jp.hari,
    ->     jp2.jam_mulai,
    ->     jp2.jam_selesai,
    ->     COUNT(DISTINCT k.id_kelas) AS total_kelas_terlibat,
    ->     GROUP_CONCAT(DISTINCT k.nama_kelas) AS kelas_terlibat
    -> FROM
    ->     jadwal_pelajaran jp
    -> JOIN
    ->     kelas k ON jp.id_kelas = k.id_kelas
    -> JOIN
    ->     guru g ON jp.id_guru = g.id_guru
    -> JOIN
    ->     jam_pelajaran jp2 ON jp.id_jam_pelajaran = jp2.id
    -> JOIN
    ->     jadwal_pelajaran jp_conflict ON jp.id_guru = jp_conflict.id_guru
    -> JOIN
    ->     kelas k_conflict ON jp_conflict.id_kelas = k_conflict.id_kelas
    -> JOIN
    ->     jam_pelajaran jp2_conflict ON jp_conflict.id_jam_pelajaran = jp2_conflict.id
    -> WHERE
    ->     jp.hari = jp_conflict.hari
    ->     AND jp2.jam_mulai = jp2_conflict.jam_mulai
    ->     AND jp2.jam_selesai = jp2_conflict.jam_selesai
    ->     AND jp.id_kelas <> jp_conflict.id_kelas
    -> GROUP BY
    ->     g.nama_guru, jp.hari, jp2.jam_mulai, jp2.jam_selesai;
+---------------------------+-------+-----------+-------------+---------------------+----------------+
| nama_guru                 | hari  | jam_mulai | jam_selesai | total_kelas_terlibat | kelas_terlibat |
+---------------------------+-------+-----------+-------------+---------------------+----------------+
| Ibrahim Mallombassang, S.Pd | Senin | 07:15:00  | 08:00:00    |                   2 | RPL 1,RPL 2    |
+---------------------------+-------+-----------+-------------+---------------------+----------------+
1 row in set (0.017 sec)
```

Tujuan Query

Query ini bertujuan untuk mendeteksi tabrakan jadwal pada waktu dan hari yang sama, dengan guru yang sama, tetapi di kelas yang berbeda. Output memberikan informasi tentang guru, waktu tabrakan,

dan nama-nama kelas yang terlibat dalam konflik.

Cara Relasi

Relasi ini menggunakan operasi JOIN antara tabel jadwal_pelajaran (jp1) dan tabel kelas (k1 dan k2) melalui kolom id_kelas, tabel jadwal_pelajaran (jp1) dan tabel guru(g) melalui kolom id_guru, tabel jadwal_pelajaran (jp1) dan tabel jam_pelajaran (jp2) melalui kolom id

Cara Agregasi

Fungsi agregasi yang bisa kita gunakan untuk menyaring hanya kelas yang memiliki lebih dari 2 jadwal pelajaran dan fungsi tersebut adalah count dan group_count agar kita bisa menyaring nama kelas yang jadwalnya bertabrakan. yang fungsi agregasi tersebut berfungsi
count : Menghitung jumlah seluruh baris dalam sebuah tabel atau grup, terlepas dari nilai kolomnya. dan kolom yang menggunakan fungsi tersebut id_kelas dalam tabel kelas
group_count : Menggabungkan nilai-nilai dari kolom yang dipilih, dan hasilnya akan dipisahkan oleh pemisah yang ditentukan (default adalah koma ,). dan kolom yang menggunakan fungsi tersebut nama_kelas dalam tabel kelas.

Penjelasan

jadwal_pelajaran: Berisi data jadwal pelajaran, termasuk ID jadwal, ID kelas, hari, waktu mulai, dan waktu selesai. Tabel ini digunakan untuk mengelola jadwal pelajaran setiap kelas dan guru.

jam_pelajaran: Berisi data tentang jam pelajaran, seperti ID jam, waktu mulai, dan waktu selesai. Tabel ini digunakan untuk menentukan durasi setiap pelajaran.

guru: Berisi data tentang guru, seperti ID guru dan nama guru. Tabel ini digunakan untuk mencatat guru yang bertanggung jawab mengajar di kelas tertentu.

kelas: Berisi data tentang kelas, seperti ID kelas dan nama kelas. Tabel ini digunakan untuk mencatat informasi tentang kelas-kelas yang ada.

Analisis

- `SELECT`: Digunakan untuk menampilkan kolom-kolom yang diambil dari tabel.
- `g.nama_guru`: Nama guru yang dijadwalkan.
- `jp1.hari`: Hari di mana jadwal berlangsung.
- `jp2.jam_mulai, jp2.jam_selesai`: Waktu dimulainya dan berakhirnya pelajaran.
- `k1.nama_kelas dan k2.nama_kelas`: Nama kelas yang bertabrakan.
- `COUNT(DISTINCT k.id_kelas) AS total_kelas_terlibat` : Menggunakan fungsi agregat COUNT untuk menghitung berapa banyak kelas (id_kelas) yang bertabrakan . Hasilnya diberi alias total_kelas_terlibat .

- `GROUP_CONCAT(DISTINCT k.nama_kelas) AS kelas_terlibat`: Menggunakan fungsi agregat GROUP_CONCAT untuk menggabungkan semua nama_kelas (nama_kelas) yang jadwal nya bertabrakan. Hasilnya diberi alias kelas_terlibat
- `FROM`: Menentukan tabel utama yang digunakan dalam query. jadwal_pelajaran jp1: Tabel ini digunakan untuk mengakses data jadwal pelajaran.
- `JOIN`: Menggabungkan tabel jam_pelajaran dengan jadwal_pelajaran. `ON jp1.id_jam_pelajaran = jp2.id`: Kondisi penghubung adalah ID jam pelajaran dari kedua tabel.
- `JOIN`: Menggabungkan tabel kelas untuk mendapatkan nama kelas pertama. `ON jp1.id_kelas = k1.id_kelas`: Menghubungkan jadwal dengan ID kelas.
- `JOIN`: Menggabungkan tabel guru untuk mendapatkan nama guru. `ON jp1.id_guru = g.id_guru`: Menghubungkan data jadwal dengan guru melalui ID guru.
- `JOIN`: Menggabungkan tabel kelas untuk mendapatkan nama kelas pertama. `ON jp1.id_kelas = k1.id_kelas`: Menghubungkan jadwal dengan ID kelas.
- `JOIN`: Menggabungkan tabel jadwal_pelajaran dengan salinan tabel itu sendiri untuk mencari konflik jadwal guru. `ON jp1.id_guru = jp1_conflict.id_guru`: Kondisi menghubungkan adalah guru yang sama.
- `JOIN`: Menghubungkan tabel kelas kedua untuk menemukan kelas lain yang bertabrakan. `ON jp1_conflict.id_kelas = k2.id_kelas`: Kondisi menghubungkan adalah ID kelas dari jadwal kedua.
- `JOIN`: Menghubungkan tabel jam_pelajaran kedua untuk menemukan waktu jadwal kedua. `ON jp1_conflict.id_jam_pelajaran = jp2_conflict.id`: Menghubungkan ID jam pelajaran untuk mencocokkan waktu.
- `WHERE`: Menentukan kondisi untuk mendeteksi tabrakan jadwal.
- `jp1.hari` = jp1_conflict.hari: Hari dari kedua jadwal harus sama.
- `jp2.jam_mulai` = jp2_conflict.jam_mulai: Jam mulai dari kedua jadwal harus sama.
- `jp2.jam_selesai` = jp2_conflict.jam_selesai: Jam selesai dari kedua jadwal harus sama.
- `jp1.id_kelas <> jp1_conflict.id_kelas`: Kelas dari kedua jadwal harus berbeda (untuk memastikan tabrakan terjadi).
- `GROUP BY`: Mengelompokkan hasil query untuk memastikan setiap kombinasi guru, hari, waktu, dan kelas yang bertabrakan ditampilkan satu kali. Kolom-kolom yang digunakan untuk pengelompokan adalah: nama_guru, hari, jam_mulai, jam_selesai, dan kedua nama kelas (kelas_1 dan kelas_2).