

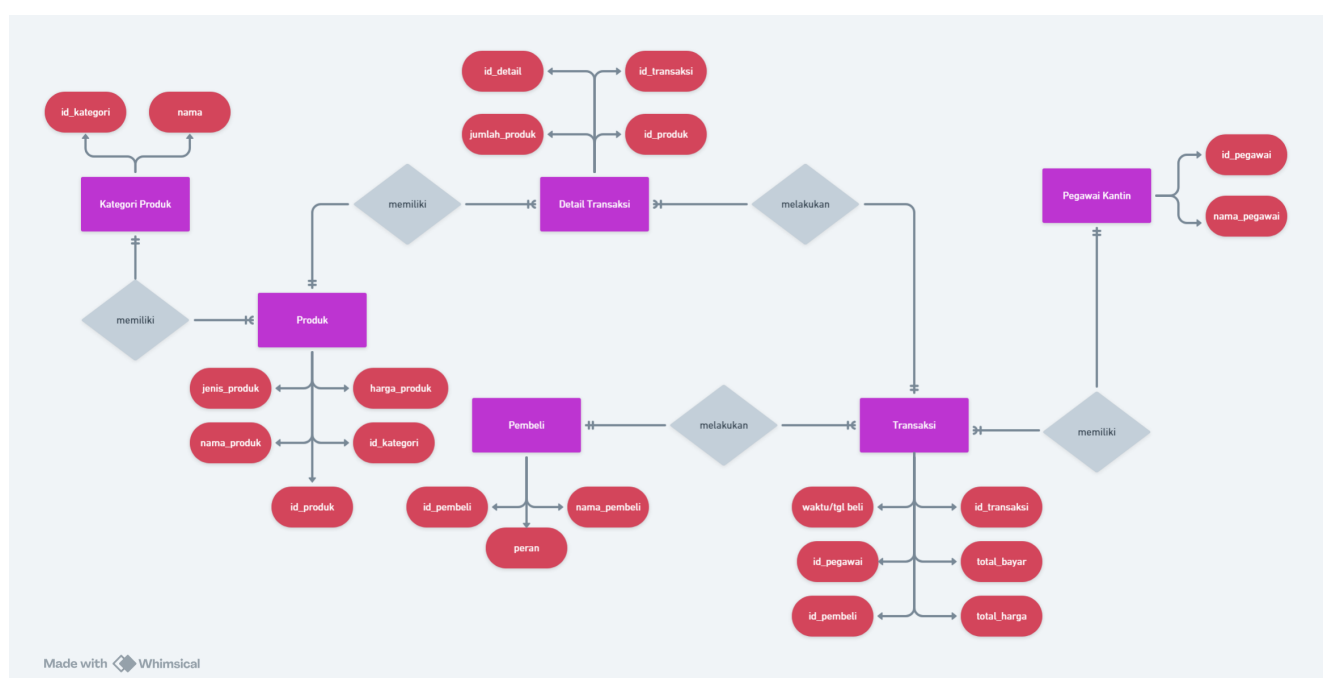
Tabel Keaktifan

Nama	Keaktifan	Keterangan
Ahsan	3	mencari dan mengerjakan relasi antar tabel
Agis	3	Mencari Analisis
Zhafran	3	mencari analisis
Fatur	3	mencari 2 contoh query relasi, group by, dan having secara bersamaan dalam satu query

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



Penjelasan ERD

1. Kategori Produk - Produk

- **Relasi:** Memiliki
- **Tipe:** One-to-Many (1:N)
- **Deskripsi:** Satu kategori produk dapat memiliki banyak produk, tetapi satu produk hanya dimiliki oleh satu kategori.

2. Produk - Detail Transaksi

- **Relasi:** Dimiliki Oleh
- **Tipe:** One-to-Many (1:N)
- **Deskripsi:** Satu produk dapat muncul di banyak detail transaksi, tetapi satu detail transaksi hanya merujuk ke satu produk.

3. Pembeli - Transaksi

- **Relasi:** Melakukan
- **Tipe:** One-to-Many (1:N)
- **Deskripsi:** Satu pembeli dapat melakukan banyak transaksi, tetapi satu transaksi hanya dilakukan oleh satu pembeli.

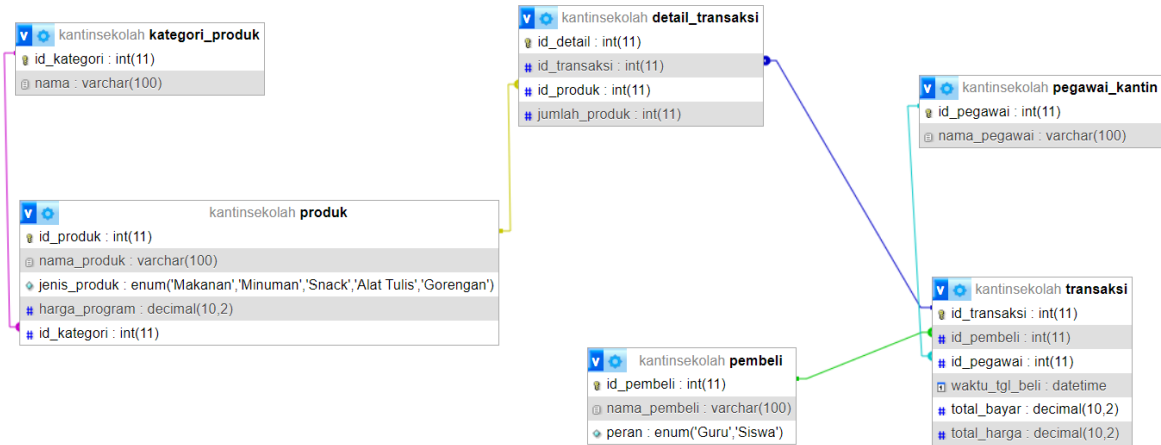
4. Transaksi - Detail Transaksi

- **Relasi:** Memiliki
- **Tipe:** One-to-Many (1:N)
- **Deskripsi:** Satu transaksi dapat memiliki banyak detail transaksi, tetapi satu detail transaksi hanya dimiliki oleh satu transaksi.

5. Kantin Sekolah - Pegawai

- **Relasi:** Memiliki
- **Tipe:** One-to-Many (1:N)
- **Deskripsi:** Satu kantin sekolah dapat memiliki banyak pegawai, tetapi satu pegawai hanya bekerja di satu kantin.

Relasi DataBase



Penjelasan

- **kategori_produk**
 - PK: `id_kategori`
 - Relasi: Digunakan untuk mengelompokkan jenis produk berdasarkan kategori.
- **produk**
 - PK: `id_produk`
 - FK: `id_kategori` (mengacu ke tabel `kategori_produk`)
 - Relasi: Menyimpan data tentang produk yang dijual di kantin.
- **detail_transaksi**
 - PK: `id_detail`
 - FK: `id_transaksi` (mengacu ke tabel `transaksi`), `id_produk` (mengacu ke tabel `produk`)
 - Relasi: Menyimpan informasi detail setiap transaksi, seperti produk dan jumlahnya.
- **pegawai_kantin**
 - PK: `id_pegawai`
 - Relasi: Menyimpan data pegawai yang bertugas di kantin.
- **pembeli**
 - PK: `id_pembeli`
 - Relasi: Menyimpan data pembeli produk.
- **transaksi**
 - PK: `id_transaksi`
 - FK: `id_pembeli` (mengacu ke tabel `pembeli`), `id_pegawai` (mengacu ke tabel `pegawai_kantin`)

- Relasi: Menyimpan informasi transaksi yang dilakukan.

Data Semua Tabel

```

MariaDB [kantinsekolah]> select * from produk;
+-----+-----+-----+-----+-----+
| id_produk | nama_produk | jenis_produk | harga_produk | id_kategori |
+-----+-----+-----+-----+-----+
| 1 | Nasi Goreng | Makanan | 15000.00 | 1 |
| 2 | Mie Ayam | Makanan | 13000.00 | 1 |
| 3 | Soto Ayam | Makanan | 16000.00 | 1 |
| 4 | Es Teh Manis | Minuman | 5000.00 | 2 |
| 5 | Kopi Hitam | Minuman | 7000.00 | 2 |
| 6 | Keripik Singkong | Snack | 8000.00 | 3 |
| 7 | Coklat Batang | Snack | 10000.00 | 3 |
| 8 | Pulpen Biru | Alat Tulis | 3000.00 | 4 |
| 9 | Buku Tulis | Alat Tulis | 7000.00 | 4 |
| 10 | Pisang Goreng | Gorengan | 6000.00 | 5 |
+-----+-----+-----+-----+-----+
10 rows in set (0.095 sec)

MariaDB [kantinsekolah]> select * from kategori_produk;
+-----+-----+
| id_kategori | nama |
+-----+-----+
| 1 | Makanan |
| 2 | Minuman |
| 3 | Snack |
| 4 | Alat Tulis |
| 5 | Gorengan |
+-----+-----+
5 rows in set (0.013 sec)

MariaDB [kantinsekolah]> select * from pegawai_kantin;
+-----+-----+
| id_pegawai | nama_pegawai |
+-----+-----+
| 1 | Pak Joko |
| 2 | Bu Siti |
+-----+-----+
2 rows in set (0.010 sec)

MariaDB [kantinsekolah]> select * from detail_transaksi;
+-----+-----+-----+-----+
| id_detail | id_transaksi | id_produk | jumlah_produk |
+-----+-----+-----+-----+
| 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 1 |
| 3 | 3 | 3 | 1 |
| 4 | 4 | 4 | 1 |
| 5 | 5 | 5 | 10 |
| 6 | 6 | 6 | 5 |
| 7 | 7 | 7 | 1 |
| 8 | 8 | 8 | 1 |
| 9 | 9 | 9 | 1 |
| 10 | 10 | 4 | 2 |
| 11 | 11 | 1 | 2 |
| 12 | 11 | 4 | 3 |
| 13 | 12 | 2 | 1 |
| 14 | 12 | 6 | 2 |
| 15 | 13 | 3 | 1 |
| 16 | 13 | 7 | 4 |
+-----+-----+-----+-----+
16 rows in set (0.011 sec)

MariaDB [kantinsekolah]> select * from transaksi;
+-----+-----+-----+-----+-----+-----+
| id_transaksi | id_pembeli | id_pegawai | waktu_tgl_beli | total_bayar | total_harga |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 2024-09-01 12:00:00 | 15000.00 | 15000.00 |
| 2 | 2 | 2 | 2024-09-01 12:30:00 | 20000.00 | 20000.00 |
| 3 | 3 | 1 | 2024-09-02 13:00:00 | 16000.00 | 16000.00 |
| 4 | 4 | 4 | 2024-09-02 14:00:00 | 10000.00 | 10000.00 |
| 5 | 5 | 5 | 2024-09-03 10:00:00 | 6000.00 | 6000.00 |
| 6 | 1 | 2 | 2024-09-03 11:00:00 | 7000.00 | 7000.00 |
| 7 | 2 | 1 | 2024-09-04 12:00:00 | 8000.00 | 8000.00 |
| 8 | 3 | 3 | 2024-09-04 15:00:00 | 3000.00 | 3000.00 |
| 9 | 4 | 1 | 2024-09-05 16:00:00 | 7000.00 | 7000.00 |
| 10 | 5 | 5 | 2024-09-06 17:00:00 | 13000.00 | 13000.00 |
| 11 | 1 | 1 | 2024-12-01 10:00:00 | 20000.00 | 20000.00 |
| 12 | 3 | 2 | 2024-12-15 14:30:00 | 15000.00 | 15000.00 |
| 13 | 5 | 1 | 2024-12-25 16:00:00 | 18000.00 | 18000.00 |
+-----+-----+-----+-----+-----+-----+
13 rows in set (0.011 sec)

MariaDB [kantinsekolah]> select * from Pembeli;
+-----+-----+-----+
| id_pembeli | nama_pembeli | peran |
+-----+-----+-----+
| 1 | Fatur | Siswa |
| 2 | Ahsan | Siswa |
| 3 | Ibrahim | Guru |
| 4 | Agis | Siswa |
| 5 | Zafhran | Siswa |
+-----+-----+-----+
5 rows in set (0.012 sec)

```

Query Setiap Tabel

Tabel Detail_Transaksi

```
CREATE TABLE `detail_transaksi` (  
  `id_detail` int(11) NOT NULL,  
  `id_transaksi` int(11) DEFAULT NULL,  
  `id_produk` int(11) DEFAULT NULL,  
  `jumlah_produk` int(11) DEFAULT NULLx  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Tabel kategori_produk

```
CREATE TABLE `kategori_produk` (  
  `id_kategori` int(11) NOT NULL,  
  `nama` varchar(100) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Tabel Produk

```
CREATE TABLE `produk` (  
  `id_produk` int(11) NOT NULL,  
  `nama_produk` varchar(100) NOT NULL,  
  `jenis_produk` enum('Makanan', 'Minuman', 'Snack', 'Alat Tulis', 'Gorengan')  
  DEFAULT NULL,  
  `harga_program` decimal(10,2) DEFAULT NULL,  
  `id_kategori` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Tabel Pembeli

```
CREATE TABLE `pembeli` (  
  `id_pembeli` int(11) NOT NULL,  
  `nama_pembeli` varchar(100) NOT NULL,  
  `peran` enum('Guru', 'Siswa') DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Tabel Pegawai_kantin

```
CREATE TABLE `pegawai_kantin` (  
  `id_pegawai` int(11) NOT NULL,  
  `nama_pegawai` varchar(100) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Tabel Transaksi

```
CREATE TABLE `transaksi` (  
  `id_transaksi` int(11) NOT NULL,  
  `id_pembeli` int(11) DEFAULT NULL,  
  `id_pegawai` int(11) DEFAULT NULL,  
  `waktu_tgl_beli` datetime DEFAULT NULL,  
  `total_bayar` decimal(10,2) DEFAULT NULL,  
  `total_harga` decimal(10,2) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

2 Contoh dengan menggunakan query relasi, group by, dan having secara bersamaan dalam satu query dan 1 tambahan soal

Soal 1

Kode Program

```
SELECT  
  pb.nama_pembeli,  
  SUM(t.total_bayar) AS total_belanja  
FROM  
  transaksi t  
JOIN  
  pembeli pb ON t.id_pembeli = pb.id_pembeli  
GROUP BY  
  pb.nama_pembeli  
HAVING  
  total_belanja > 15000;
```

Hasil Program

```

MariaDB [kantinsekolah]> SELECT
  ->     pb.nama_pembeli,
  ->     SUM(t.total_bayar) AS total_belanja
  -> FROM
  ->     transaksi t
  -> JOIN
  ->     pembeli pb ON t.id_pembeli = pb.id_pembeli
  -> GROUP BY
  ->     pb.nama_pembeli
  -> HAVING
  ->     total_belanja > 15000;
+-----+-----+
| nama_pembeli | total_belanja |
+-----+-----+
| Agis         | 17000.00      |
| Ahsan        | 28000.00      |
| Fatur        | 22000.00      |
| Ibrahim      | 19000.00      |
| Zafhran      | 19000.00      |
+-----+-----+
5 rows in set (0.001 sec)

MariaDB [kantinsekolah]> |

```

Analisis

Tujuan Query: Mengetahui total transaksi yang dilakukan setiap pembeli, di mana total belanja mereka lebih dari Rp15.000.

Cara Relasi:

- Tabel `transaksi` direlasikan dengan tabel `pembeli` berdasarkan kolom `id_pembeli`.

Cara Agregasi:

- Menggunakan fungsi `SUM()` untuk menghitung total belanja per pembeli.
- `GROUP BY` digunakan untuk mengelompokkan data berdasarkan nama pembeli.
- `HAVING` digunakan untuk memfilter pembeli dengan total belanja lebih dari Rp15.000.

Soal 2

Kode Program

```

SELECT
p.nama_produk,
SUM(dt.jumlah_produk) AS total_terjual,
SUM(p.harga_program * dt.jumlah_produk) AS total_pendapatan
FROM
detail_transaksi dt

```


JOIN

produk p ON dt.id_produk = p.id_produk

GROUP BY

p.nama_produk

HAVING

total_pendapatan > 10000;

Hasil Program

```
MariaDB [kantinsekolah]> SELECT
->   p.nama_produk,
->   SUM(dt.jumlah_produk) AS total_terjual,
->   SUM(p.harga_program * dt.jumlah_produk) AS total_pendapatan
-> FROM
->   detail_transaksi dt
-> JOIN
->   produk p ON dt.id_produk = p.id_produk
-> GROUP BY
->   p.nama_produk
-> HAVING
->   total_pendapatan > 10000;
+-----+-----+-----+
| nama_produk | total_terjual | total_pendapatan |
+-----+-----+-----+
| Mie Ayam   | 1             | 13000.00         |
| Nasi Goreng | 1             | 15000.00         |
| Soto Ayam   | 1             | 16000.00         |
+-----+-----+-----+
3 rows in set (0.001 sec)

MariaDB [kantinsekolah]> |
```

Analisis

Tujuan Query: Mengetahui total jumlah produk yang terjual dan total pendapatan dari masing-masing produk yang sudah dibeli oleh pembeli.

Cara Relasi:

- Tabel `detail_transaksi` direlasikan dengan tabel `produk` berdasarkan kolom `id_produk`.

Cara Agregasi:

- Menggunakan fungsi `SUM()` untuk menghitung total pendapatan dan total produk terjual.
- `GROUP BY` digunakan untuk mengelompokkan data berdasarkan nama produk.
- `HAVING` digunakan untuk memfilter produk yang pendapatannya lebih dari Rp10.000.

Soal 3 :berikan query menghitung jumlah produk yang dibeli pada bulan desember 2024

Kode Program

```
SELECT
    SUM(t.total_harga) AS total_harga_desember,
    SUM(dt.jumlah_produk) AS total_produk_desember
FROM transaksi t
JOIN detail_transaksi dt ON t.id_transaksi = dt.id_transaksi
WHERE t.waktu_tgl_beli BETWEEN '2024-12-01 00:00:00' AND '2024-12-31 23:59:59';
```

Hasil Program

```
MariaDB [kantinsekolah]> SELECT
->     SUM(t.total_harga) AS total_harga_desember,
->     SUM(dt.jumlah_produk) AS total_produk_desember
-> FROM transaksi t
-> JOIN detail_transaksi dt ON t.id_transaksi = dt.id_transaksi
-> WHERE t.waktu_tgl_beli BETWEEN '2024-12-01 00:00:00' AND '2024-12-31 23:59:59';
+-----+-----+
| total_harga_desember | total_produk_desember |
+-----+-----+
|          106000.00   |             13        |
+-----+-----+
1 row in set (0.001 sec)
```

Analisis

Tujuan Query:

Menghitung total pendapatan (`total_harga`) dan total produk terjual (`jumlah_produk`) selama bulan Desember 2024.

Cara Relasi:

- Tabel `transaksi` direlasikan dengan tabel `detail_transaksi` menggunakan kolom `id_transaksi` untuk menggabungkan data transaksi dengan detail produk yang terjual.

Cara Penyaringan Data:

- Menggunakan klausa `WHERE` untuk membatasi data pada rentang waktu **Desember 2024** (1 Desember hingga 31 Desember, termasuk batas waktu).

Cara Agregasi:

- Fungsi agregasi `SUM()` digunakan untuk menjumlahkan total harga dari tabel `transaksi` dan total produk yang terjual dari tabel `detail_transaksi`.