

**LAPORAN PRAKTIKUM  
BASIS DATA RD  
MODUL 4**

**Oleh :**

**Muhammad Fadhil Zurani      (122140146)**



**Program Studi Teknik Informatika**

**Fakultas Teknologi Industri**

**Institut Teknologi Sumatera**

**2024**

## Daftar Isi

<b>Daftar Isi .....</b>	<b>2</b>
<b>1. Dasar Teori .....</b>	<b>3</b>
<b>2. Ulasan .....</b>	<b>3</b>
<b>3. Hasil dan Jawaban.....</b>	<b>7</b>
<b>4. Kesimpulan dan Saran.....</b>	<b>13</b>

## 1. Dasar Teori

Join adalah elemen penting dalam bahasa SQL yang memungkinkan pengguna untuk menggabungkan data dari dua atau lebih tabel dalam satu query. Inner join adalah jenis join yang mengembalikan baris-baris yang memiliki nilai yang cocok di kedua tabel yang di-join berdasarkan kriteria tertentu, memungkinkan pengguna untuk mengambil informasi terkait dari tabel yang berbeda. Cross join, di sisi lain, menghasilkan kombinasi setiap baris dari tabel pertama dengan setiap baris dari tabel kedua, berguna ketika pengguna ingin menghasilkan semua kombinasi baris tanpa memperhatikan nilai-nilai yang cocok. Natural join adalah jenis join yang secara otomatis menggabungkan tabel berdasarkan kolom-kolom dengan nama yang sama dan tipe data yang cocok, mempermudah proses penggabungan data jika ada kolom-kolom yang sama di kedua tabel.

## 2. Ulasan

Database Galeri\_itera

1. Tambahkan data berikut pada tabel Produk.

Produk_id	Produk_nama	Jumlah_Stok	Supplier_id
P552	Aqua 1 L	300 pcs	S001
P882	Indomilk 25 ml	200 pcs	S004
P333	Sari Roti 100 gram	30 pcs	S005
P123	Gulaku 1 Kg	100 pcs	S005
P453	Garam 30 gram	20 pcs	S006
P109	The Kotak 300 ml	40 pcs	S002

Instruksi tersebut meminta kita untuk menambahkan data pada tabel "produk" sesuai contoh diatas.

2. Tambahkan data berikut pada tabel Supplier.

Supplier_id	Company_nama	Nama_Kontak
S004	Pelita Baru	Puspa
S005	Surya Kun	Siti
S006	Ceria Kasih	Topan

Instruksi tersebut meminta kita untuk menambahkan data pada tabel "supplier" sesuai contoh diatas.

3. Tambahkan data pada tabel Pegawai.

Id_pegawai	Pegawai_nama	Jabatan
Pg_002	Siska	Casier
Pg_003	Nuri	Casier
Pg_004	Jamal	Casier

Instruksi tersebut meminta kita untuk menambahkan data pada tabel "pegawai" sesuai contoh diatas.

4. Tabel Pembeli tetap seperti minggu sebelumnya.

Tidak ada perubahan data dari tabel "pembeli"

5. Buatlah tabel Transaksi sesuai keterangan pada tabel berikut dan isilah data ke dalam tabel transaksi.

Id_transaksi (PK)	Id_pembeli (FK)	Id_pegawai (FK)	Produk_Id (FK)	Tgl_transaksi	Jumlah_beli
----------------------	--------------------	--------------------	-------------------	---------------	-------------

1	C_901	Pg_002	P552	2022-09-12	3 pcs
2	C_901	Pg_003	P109	2022-09-15	10 pcs
3	C_810	Pg_001	P114	2022-09-15	2 pcs
4	C_991	Pg_004	P333	2022-09-18	3 pcs
5	C_800	Pg_001	P552	2022-09-18	1 pcs
6	C_810	Pg_002	P_552	2022-09-20	10 pcs

Instruksi tersebut meminta kita untuk menambahkan data pada tabel "transaksi" sesuai contoh diatas.

6. Tampilkan data Produk dan supplier menggunakan Natural Join.

Perintah tersebut bertujuan untuk menampilkan data produk dan supplier menggunakan natural join, yang akan menggabungkan data berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis.

7. Tampilkan data Produk dan supplier yang sesuai (sama) menggunakan Join.

Perintah tersebut bertujuan untuk menampilkan data produk dan supplier yang sesuai (sama) menggunakan join, yang menggabungkan data berdasarkan kriteria tertentu yang ditentukan, seperti kesesuaian nilai pada kolom tertentu.

8. Dengan melakukan salah satu dari jenis Join tabel, tampilkan Jabatan pegawai, id pegawai, tanggal transaksi untuk pegawai dengan id "Pg\_001".

Perintah tersebut bertujuan untuk menggunakan salah satu jenis join pada tabel untuk menampilkan jabatan pegawai, ID pegawai, dan tanggal transaksi untuk pegawai dengan ID "Pg\_001".

9. Dengan melakukan salah satu dari jenis Join tabel, tampilkan Company nama, ProdukId, Supplier id ketika Jumlah stok > 50pcs.

Perintah tersebut bertujuan untuk menggunakan salah satu jenis join pada tabel untuk menampilkan nama perusahaan, ID produk, dan ID supplier saat jumlah stok lebih dari 50 pcs.

10. Dengan melakukan salah satu dari jenis Join tabel, tampilkan Produk id, id transaksi , jumlah dan tanggal transaksi ketika jumlah yang dibeli > 6pcs.

Perintah tersebut bertujuan menggunakan jenis join pada tabel untuk menampilkan ID produk, ID transaksi, jumlah pembelian, dan tanggal transaksi saat jumlah yang dibeli lebih dari 6 pcs.

11. Tampilkan data dari tabel Transaksi dan pegawai yang sesuai(sama).

Perintah tersebut bertujuan untuk menampilkan data dari tabel Transaksi dan pegawai yang memiliki keterkaitan atau kesesuaian berdasarkan kriteria tertentu.

12. Tampilkan id\_pembeli, Pembeli\_nama, tanggal transaksi, Produk\_id dan jumlah\_beli yang sama (sesuai) dengan melakukan Join pada tabel pembeli dan transaksi.

Perintah tersebut digunakan untuk menampilkan id\_pembeli, nama pembeli, tanggal transaksi, Produk\_id, dan jumlah\_beli yang sama atau sesuai dengan melakukan join antara tabel pembeli dan transaksi berdasarkan kriteria tertentu.

13. Dengan melakukan salah satu dari jenis Join tabel, tampilkan Produk\_id, supplier\_id, Produk\_nama, Nama kontak ketika Produk\_nama "G".

Perintah tersebut digunakan untuk menampilkan Produk\_id, supplier\_id, Produk\_nama, dan Nama kontak dari supplier yang terkait, ketika Produk\_nama adalah "G", dengan melakukan salah satu jenis join antara tabel Produk dan supplier.

14. Dengan melakukan salah satu dari jenis Join tabel, tampilkan Produk\_id, Produk\_nama, id\_transaksi, jumlah stok, tanggal transaksi untuk Produk\_id "P1".

Perintah tersebut bertujuan untuk menampilkan Produk\_id, Produk\_nama, id\_transaksi, jumlah stok, dan tanggal transaksi untuk produk dengan Produk\_id "P1", menggunakan salah satu jenis join antara tabel Produk dan transaksi.

15. Tampilkan Produk\_id, Produk\_nama, tanggal transaksi, id\_transaksi ketika Jumlah stok > 60pcs dan < 150pcs.

Perintah tersebut bertujuan untuk menampilkan Produk\_id, Produk\_nama, tanggal transaksi, dan id\_transaksi dari transaksi yang memiliki Jumlah stok antara lebih dari 60pcs dan kurang dari 150pcs.

16. Tampilkan id\_pegawai, jabatan, id\_transaksi untuk transaksi dengan jumlah yang dibeli < 6pcs tanggal transaksi antara "2022-09-15 sampai dengan 2022-09-18".

Perintah tersebut bertujuan untuk menampilkan id\_pegawai, jabatan, dan id\_transaksi untuk transaksi dengan jumlah yang dibeli kurang dari 6pcs, yang tanggal transaksinya berada di antara rentang "2022-09-15" sampai "2022-09-18".

### 3. Hasil dan Jawaban

1. Soal 1 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> select * from produk;
```

Produk_id	Produk_nama	Jumlah_stok	Supplies_id
P109	The Kotak 300 ml	40	S002
P114	Milo 100 ml	800	S001
P115	Milo 150 ml	50	S003
P123	Gulaku 1 Kg	100	S005
P235	Aqua 250 ml	300	S001
P311	Grand 320 ml	400	S003
P333	Sari Roti 100 gram	30	S005
P441	Rojo Lele 5 kg	60	S002
P453	Garam 30 gram	20	S006
P552	Aqua 1 L	300	S001
P882	Indomilk 25 ml	200	S004

```
11 rows in set (0.000 sec)
```

Gambar diatas adalah kondisi tabel setelah kita menambahkan data tambahan yang ada pada soal pertama, yang sebelumnya kita tambahkan lewat command sql seperti berikut:

```
insert into Produk values
('P552', 'Aqua 1 L', 300, 'S001'),
('P882', 'Indomilk 25 ml', 200, 'S004'),
('P333', 'Sari Roti 100 gram', 30, 'S005'),
('P123', 'Gulaku 1 Kg', 100, 'S005'),
('P453', 'Garam 30 gram', 20, 'S006'),
('P109', 'The Kotak 300 ml', 40, 'S002');
```

2. Soal 2 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> select * from suppliers;
```

Suppliers_id	Company_nama	Nama_kontak
S001	Semua Terang	Ali
S002	Suka Maju	Rahmat
S003	Maju Terus	Dayono
S004	Pelita Baru	Puspa
S005	Surya Kun	Siti
S006	Ceria Kasih	Topan

```
6 rows in set (0.003 sec)
```

Gambar diatas adalah kondisi tabel suppliers setelah kita menambahkan data tambahan yang ada pada soal kedua, yang sebelumnya kita tambahkan lewat command sql seperti berikut:

```
insert into Suppliers values
('S004', 'Pelita Baru', 'Puspa'),
('S005', 'Surya Kun', 'Siti'),
('S006', 'Ceria Kasih', 'Topan');
```

3. Soal 3 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> select * from pegawai;
+-----+-----+-----+
| id_pegawai | Pegawai_nama | Jabatan |
+-----+-----+-----+
| pg_001     | Santi        | Cashier |
| Pg_002     | Siska Casier | Casier  |
| Pg_003     | Nuri Casier  | Casier  |
| Pg_004     | Jamal Casier | Casier  |
| pg_201     | Santo        | Cashier |
| pg_300     | yaya         | Manager |
+-----+-----+-----+
6 rows in set (0.003 sec)
```

Gambar diatas adalah kondisi tabel pegawai setelah kita menambahkan data tambahan yang ada pada soal ketiga, yang sebelumnya kita tambahkan lewat command sql seperti berikut:

```
insert into Pegawai values
('Pg_002', 'Siska Casier', 'Casier'),
('Pg_003', 'Nuri Casier', 'Casier'),
('Pg_004', 'Jamal Casier', 'Casier');
```

4. Soal 4 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> select * from pembeli;
+-----+-----+-----+
| id_pembeli | Pembeli_nama | Pembeli_kontak |
+-----+-----+-----+
| C_800      | Egi          | 0812521221     |
| C_810      | Ardi         | 0862145121     |
| C_890      | prassetya    | 08521116464    |
| C_901      | Rudi         | 081231511      |
| C_991      | Andi         | 085212021111   |
+-----+-----+-----+
5 rows in set (0.002 sec)
```



Gambar diatas adalah kondisi tabel pembeli yang tidak berubah datanya, yang dimana datanya sudah dibuat di praktikum minggu sebelumnya.

5. Soal 5 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> desc transaksi;
```

Field	Type	Null	Key	Default	Extra
Id_transaksi	int(11)	NO	PRI	NULL	
id_pembeli	char(6)	YES	MUL	NULL	
Id_pegawai	char(6)	YES	MUL	NULL	
Produk_Id	char(4)	YES	MUL	NULL	
Tgl_transaksi	date	YES		NULL	
Jumlah_beli	int(4)	YES		NULL	

6 rows in set (0.009 sec)

Gambar diatas kondisi tabel baru bernama "transaksi" yang dibuat dengan menggunakan perintah sql:

```
create table Transaksi(
    Id_transaksi int(11) primary key,
    Id_pembeli char(5),
    Id_pegawai char(6),
    Produk_Id char(4),
    Tgl_transaksi date,
    Jumlah_beli int(4)
);

alter table transaksi add foreign key(id_pembeli) references
pembeli(id_pembeli);
alter table transaksi add foreign key(id_pegawai) references
pegawai(id_pegawai);
alter table transaksi add foreign key(produk_id) references produk
(produk_id);
```

Setelah membuat tabel kita masukan datanya dengan perintah sql:

```
insert into Transaksi values
(1, 'C_901', 'Pg_002', 'P552', '2022-09-12', 3),
(2, 'C_901', 'Pg_003', 'P109', '2022-09-15', 10),
(3, 'C_810', 'Pg_001', 'P114', '2022-09-15', 2),
(4, 'C_991', 'Pg_004', 'P333', '2022-09-18', 3),
(5, 'C_800', 'Pg_001', 'P552', '2022-09-18', 1),
(6, 'C_810', 'Pg_002', 'P552', '2022-09-20', 10);
```

## 6. Soal 6 Screenshoot hasil dan jawaban dari pengujian

```

MariaDB [galeri_iteks]> SELECT * FROM Produk NATURAL JOIN Suppliers;

```

Produk_id	Produk_nama	Jumlah_stok	Supplies_id	Suppliers_id	Company_name	Messe_kontak
P189	The Kotak 398 ml	48	S982	S981	Semus Terang	Ali
P189	The Kotak 398 ml	48	S982	S982	Suka Maju	Rahmat
P189	The Kotak 398 ml	48	S982	S983	Maju Terus	Deyono
P189	The Kotak 398 ml	48	S982	S984	Pelita Baru	Puspa
P189	The Kotak 398 ml	48	S982	S985	Surya Kun	Siti
P189	The Kotak 398 ml	48	S982	S986	Ceris Kasih	Topan
P114	Milo 188 ml	888	S981	S981	Semus Terang	Ali
P114	Milo 188 ml	888	S981	S982	Suka Maju	Rahmat
P114	Milo 188 ml	888	S981	S983	Maju Terus	Deyono
P114	Milo 188 ml	888	S981	S984	Pelita Baru	Puspa
P114	Milo 188 ml	888	S981	S985	Surya Kun	Siti
P114	Milo 188 ml	888	S981	S986	Ceris Kasih	Topan
P113	Milo 138 ml	38	S983	S981	Semus Terang	Ali
P113	Milo 138 ml	38	S983	S982	Suka Maju	Rahmat
P113	Milo 138 ml	38	S983	S983	Maju Terus	Deyono
P113	Milo 138 ml	38	S983	S984	Pelita Baru	Puspa
P113	Milo 138 ml	38	S983	S985	Surya Kun	Siti
P113	Milo 138 ml	38	S983	S986	Ceris Kasih	Topan
P123	GuTaku 1 Kg	188	S983	S981	Semus Terang	Ali
P123	GuTaku 1 Kg	188	S983	S982	Suka Maju	Rahmat
P123	GuTaku 1 Kg	188	S983	S983	Maju Terus	Deyono
P123	GuTaku 1 Kg	188	S983	S984	Pelita Baru	Puspa
P123	GuTaku 1 Kg	188	S983	S985	Surya Kun	Siti
P123	GuTaku 1 Kg	188	S983	S986	Ceris Kasih	Topan
P233	Aque 238 ml	388	S981	S981	Semus Terang	Ali
P233	Aque 238 ml	388	S981	S982	Suka Maju	Rahmat
P233	Aque 238 ml	388	S981	S983	Maju Terus	Deyono
P233	Aque 238 ml	388	S981	S984	Pelita Baru	Puspa
P233	Aque 238 ml	388	S981	S985	Surya Kun	Siti
P233	Aque 238 ml	388	S981	S986	Ceris Kasih	Topan
P311	Grand 328 ml	488	S983	S981	Semus Terang	Ali
P311	Grand 328 ml	488	S983	S982	Suka Maju	Rahmat
P311	Grand 328 ml	488	S983	S983	Maju Terus	Deyono
P311	Grand 328 ml	488	S983	S984	Pelita Baru	Puspa
P311	Grand 328 ml	488	S983	S985	Surya Kun	Siti
P311	Grand 328 ml	488	S983	S986	Ceris Kasih	Topan
P333	Sari Roti 188 gram	38	S983	S981	Semus Terang	Ali
P333	Sari Roti 188 gram	38	S983	S982	Suka Maju	Rahmat
P333	Sari Roti 188 gram	38	S983	S983	Maju Terus	Deyono
P333	Sari Roti 188 gram	38	S983	S984	Pelita Baru	Puspa
P333	Sari Roti 188 gram	38	S983	S985	Surya Kun	Siti
P333	Sari Roti 188 gram	38	S983	S986	Ceris Kasih	Topan
P441	Rojo Lela 3 kg	68	S982	S981	Semus Terang	Ali
P441	Rojo Lela 3 kg	68	S982	S982	Suka Maju	Rahmat
P441	Rojo Lela 3 kg	68	S982	S983	Maju Terus	Deyono
P441	Rojo Lela 3 kg	68	S982	S984	Pelita Baru	Puspa
P441	Rojo Lela 3 kg	68	S982	S985	Surya Kun	Siti
P441	Rojo Lela 3 kg	68	S982	S986	Ceris Kasih	Topan
P433	Garam 38 gram	28	S986	S981	Semus Terang	Ali
P433	Garam 38 gram	28	S986	S982	Suka Maju	Rahmat
P433	Garam 38 gram	28	S986	S983	Maju Terus	Deyono
P433	Garam 38 gram	28	S986	S984	Pelita Baru	Puspa
P433	Garam 38 gram	28	S986	S985	Surya Kun	Siti
P433	Garam 38 gram	28	S986	S986	Ceris Kasih	Topan
P332	Aque 1 L	388	S981	S981	Semus Terang	Ali
P332	Aque 1 L	388	S981	S982	Suka Maju	Rahmat
P332	Aque 1 L	388	S981	S983	Maju Terus	Deyono
P332	Aque 1 L	388	S981	S984	Pelita Baru	Puspa
P332	Aque 1 L	388	S981	S985	Surya Kun	Siti
P332	Aque 1 L	388	S981	S986	Ceris Kasih	Topan
P882	Indomilk 23 ml	288	S984	S981	Semus Terang	Ali
P882	Indomilk 23 ml	288	S984	S982	Suka Maju	Rahmat
P882	Indomilk 23 ml	288	S984	S983	Maju Terus	Deyono
P882	Indomilk 23 ml	288	S984	S984	Pelita Baru	Puspa
P882	Indomilk 23 ml	288	S984	S985	Surya Kun	Siti
P882	Indomilk 23 ml	288	S984	S986	Ceris Kasih	Topan

66 rows in set (0.081 sec)

Perintah SQL di atas digunakan untuk menampilkan data dari tabel Produk dan Suppliers dengan menggunakan Natural Join, menggabungkan data berdasarkan kolom yang memiliki nama yang sama di kedua tabel secara otomatis.

## 7. Soal 7 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT * FROM Produk JOIN Suppliers ON Produk.Supplies_id = Suppliers.Suppliers_id;
```

Produk_id	Produk_nama	Jumlah_stok	Supplies_id	Suppliers_id	Company_nama	Nama_kontak
P109	The Kotak 300 mL	40	S002	S002	Suka Maju	Rahmat
P114	Milo 100 mL	800	S001	S001	Semua Terang	Ali
P115	Milo 150 mL	50	S003	S003	Maju Terus	Dayono
P123	Gulaku 1 Kg	100	S005	S005	Surya Kun	Siti
P235	Aqua 250 mL	300	S001	S001	Semua Terang	Ali
P311	Grand 320 mL	400	S003	S003	Maju Terus	Dayono
P333	Sari Roti 100 gram	30	S005	S005	Surya Kun	Siti
P441	Rojo Lele 5 kg	60	S002	S002	Suka Maju	Rahmat
P453	Garam 30 gram	20	S006	S006	Ceria Kasih	Topan
P552	Aqua 1 L	300	S001	S001	Semua Terang	Ali
P882	Indomilk 25 mL	200	S004	S004	Pelita Baru	Puspa

```
11 rows in set (0.000 sec)
```

Perintah SQL di atas digunakan untuk menampilkan data dari tabel Produk dan Suppliers dengan menggunakan Join, menggabungkan data berdasarkan kolom Supplies\_id dari tabel Produk dan Suppliers\_id dari tabel Suppliers.

## 8. Soal 8 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Pegawai.Jabatan, Transaksi.Id_pegawai, Transaksi.Tgl_transaksi FROM Pegawai INNER JOIN Transaksi ON Pegawai.id_pegawai = Transaksi.Id_pegawai WHERE Pegawai.id_pegawai = 'pg_001';
```

Jabatan	Id_pegawai	Tgl_transaksi
Cashier	Pg_001	2022-09-15
Cashier	Pg_001	2022-09-18

```
2 rows in set (0.001 sec)
```

Perintah SQL di atas digunakan untuk menggabungkan tabel Pegawai dan Transaksi dengan Inner Join berdasarkan kolom id\_pegawai, kemudian memilih data jabatan pegawai, id pegawai, dan tanggal transaksi yang sesuai dengan id pegawai 'pg\_001' dari tabel Pegawai

## 9. Soal 9 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Suppliers.Company_nama, Produk.Produk_id, Produk.Supplies_id FROM Produk JOIN Suppliers ON Produk.Supplies_id = Suppliers.Suppliers_id WHERE Produk.Jumlah_stok > 50;
```

Company_nama	Produk_id	Supplies_id
Semua Terang	P114	S001
Surya Kun	P123	S005
Semua Terang	P235	S001
Maju Terus	P311	S003
Suka Maju	P441	S002
Semua Terang	P552	S001
Pelita Baru	P882	S004

```
7 rows in set (0.000 sec)
```

Perintah SQL di atas digunakan untuk menggabungkan tabel Produk dan Suppliers dengan jenis Join menggunakan kolom Supplies\_id, kemudian memilih data Company\_nama dari Suppliers, Produk\_id, dan Supplies\_id dari Produk ketika Jumlah\_stok produk lebih dari 50.

## 10. Soal 10 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Transaksi.Produk_Id, Transaksi.Id_transaksi, Transaksi.Jumlah_beli, Transaksi.Tgl_transaksi FROM Transaksi WHERE Transaksi.Jumlah_beli > '6 pcs';
```

Produk_Id	Id_transaksi	Jumlah_beli	Tgl_transaksi
P109	2	10	2022-09-15
P552	6	10	2022-09-20

```
2 rows in set, 1 warning (0.000 sec)
```

Perintah SQL di atas digunakan untuk memilih data dari tabel Transaksi di mana Jumlah\_beli lebih dari 6 pcs, dan menampilkan Produk\_Id, Id\_transaksi, Jumlah\_beli, dan Tgl\_transaksi.

## 11. Soal 11 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT * FROM Transaksi JOIN Pegawai ON Transaksi.Id_pegawai = Pegawai.id_pegawai;
```

Id_transaksi	id_pembeli	Id_pegawai	Produk_Id	Tgl_transaksi	Jumlah_beli	id_pegawai	Pegawai_nama	Jabatan
1	C_901	Pg_002	P552	2022-09-12	3	Pg_002	Siska Casier	Casier
2	C_901	Pg_003	P109	2022-09-15	10	Pg_003	Nuri Casier	Casier
3	C_810	Pg_001	P114	2022-09-15	2	pg_001	Santi	Cashier
4	C_991	Pg_004	P333	2022-09-18	3	Pg_004	Jamal Casier	Casier
5	C_800	Pg_001	P552	2022-09-18	1	pg_001	Santi	Cashier
6	C_810	Pg_002	P552	2022-09-20	10	Pg_002	Siska Casier	Casier

```
6 rows in set (0.000 sec)
```

Perintah SQL di atas digunakan untuk melakukan join antara tabel Transaksi dan tabel Pegawai berdasarkan kolom Id\_pegawai. Hasil join tersebut akan menampilkan semua kolom dari kedua tabel.

## 12. Soal 12 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Pembeli.id_pembeli, Pembeli.Pembeli_nama, Transaksi.Tgl_transaksi, Transaksi.Produk_Id, Transaksi.Jumlah_beli FROM Pembeli JOIN Transaksi ON Pembeli.id_pembeli = Transaksi.Id_pembeli;
```

id_pembeli	Pembeli_nama	Tgl_transaksi	Produk_Id	Jumlah_beli
C_901	Rudi	2022-09-12	P552	3
C_901	Rudi	2022-09-15	P109	10
C_810	Ardi	2022-09-15	P114	2
C_991	Andi	2022-09-18	P333	3
C_800	Egi	2022-09-18	P552	1
C_810	Ardi	2022-09-20	P552	10

```
6 rows in set (0.000 sec)
```

Perintah SQL di atas digunakan untuk melakukan join antara tabel Pembeli dan tabel Transaksi berdasarkan kolom id\_pembeli. Hasil join tersebut akan menampilkan id\_pembeli, Pembeli\_nama, Tgl\_transaksi, Produk\_Id, dan Jumlah\_beli dari kedua tabel.

## 13. Soal 13 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Produk.Produk_id, Produk.Supplies_id, Produk.Produk_nama, Suppliers>Nama_kontak FROM Produk JOIN Suppliers ON Produk.Supplies_id = Suppliers.Suppliers_id WHERE Produk.Produk_nama LIKE '%%';
```

Produk_id	Supplies_id	Produk_nama	Nama_kontak
P123	S005	Gulaku 1 Kg	Siti
P311	S003	Grand 320 mL	Dayono
P453	S006	Garam 30 gram	Topan

```
3 rows in set (0.000 sec)
```

Perintah SQL di atas digunakan untuk melakukan join antara tabel Produk dan tabel Suppliers berdasarkan kolom Supplies\_id, dengan menampilkan Produk\_id, Supplies\_id, Produk\_nama, dan Nama\_kontak dari kedua tabel, di mana Produk\_nama di tabel Produk dimulai dengan huruf 'G'.

14. Soal 14 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Produk.Produk_id, Produk.Produk_nama, Transaksi.Id_transaksi, Produk.Jumlah_stok, Transaksi.Tgl_transaksi FROM Produk JOIN Transaksi ON Produk.Produk_id = Transaksi.Produk_Id WHERE Produk.Produk_id LIKE 'P1%';
```

Produk_id	Produk_nama	Id_transaksi	Jumlah_stok	Tgl_transaksi
P109	The Kotak 300 mL	2	40	2022-09-15
P114	Milo 100 mL	3	800	2022-09-15

2 rows in set (0.001 sec)

Perintah SQL di atas menggabungkan data antara tabel Produk dan Transaksi berdasarkan Produk\_id dan Produk\_Id, lalu menampilkan Produk\_id, Produk\_nama, Id\_transaksi, Jumlah\_stok, dan Tgl\_transaksi di mana Produk\_id di tabel Produk dimulai dengan 'P1'.

15. Soal 15 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Produk.Produk_id, Produk.Produk_nama, Transaksi.Tgl_transaksi, Transaksi.Id_transaksi FROM Produk JOIN Transaksi ON Produk.Produk_id = Transaksi.Produk_Id WHERE Produk.Jumlah_stok > 60 AND Produk.Jumlah_stok < 150;
```

Empty set (0.000 sec)

Perintah SQL di atas menggabungkan data dari tabel Produk dan Transaksi berdasarkan Produk\_id, kemudian menampilkan Produk\_id, Produk\_nama, Id\_transaksi, dan Tgl\_transaksi di mana jumlah\_stok dari produk berada di antara 60 dan 150. Namun saat di enter menunjukan *empty set*.

16. Soal 16 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [galeri_itera]> SELECT Pegawai.id_pegawai, Pegawai.Jabatan, Transaksi.Id_transaksi FROM Pegawai JOIN Transaksi ON Pegawai.id_pegawai = Transaksi.Id_pegawai WHERE Transaksi.Jumlah_beli < '6 pcs' AND Transaksi.Tgl_transaksi BETWEEN '2022-09-15' AND '2022-09-18';
```

id_pegawai	Jabatan	Id_transaksi
pg_001	Cashier	3
Pg_004	Casier	4
pg_001	Cashier	5

3 rows in set, 1 warning (0.000 sec)

Perintah SQL di atas menggabungkan data dari tabel Pegawai dan Transaksi berdasarkan id pegawai, kemudian menampilkan id pegawai, jabatan, dan id transaksi di mana jumlah beli dalam transaksi kurang dari 6 pcs dan tanggal transaksi berada di antara '2022-09-15' dan '2022-09-18'.

## 4. Kesimpulan dan Saran

Database Galeri\_itera mencakup tabel Produk, Suppliers, Pegawai, dan Transaksi yang terkait. Dari query-query yang dijalankan, kita dapat melihat bagaimana data di tabel-tabel tersebut digabungkan menggunakan berbagai jenis JOIN seperti NATURAL JOIN, JOIN dengan kriteria, dan INNER JOIN berdasarkan kunci asing. Ini memberikan wawasan tentang bagaimana data

terkait dihubungkan dalam database, serta cara mengambil informasi yang spesifik dari kombinasi data tersebut.

Saran untuk pengembangan lebih lanjut adalah memperhatikan konsistensi data, seperti mengecek kesesuaian antara kunci asing dan nilai yang direferensikan dalam tabel yang dihubungkan. Selain itu, bisa dijelajahi lebih lanjut bagaimana menggunakan fungsi-fungsi SQL lainnya seperti GROUP BY, ORDER BY, dan agregasi data untuk analisis yang lebih mendalam.