

**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

Daftar Isi	2
1. Dasar Teori	3
2. Ulasan	3
3. Hasil dan Jawaban.....	6
4. Kesimpulan dan Saran.....	12

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

1. Buat dan Tambahkan tabel “nasabah” pada Database Bank_Itera dan isi nilai pada setiap kolom seperti berikut.

Id_Nasabah (PK)	no_rekening (FK)	Nama_Nasabah	Alamat	No_Telepon
1	145678	Puspa Sari	Sei Batang Kuis	081376809122
2	256132	Sinta Jojo	Iskandar Muda	089987654311
3	247182	Bunga Putri	Jalan Martapura	081249987011
4	350067	Grace Sebayang	Gang Keluarga	085276812453
5	114115	Christine Talia	Sei Kambah	082266778866
6	100111	Rimayka Natalia	Perumahan Puri K	081234567890

Instruksi tersebut meminta untuk membuat tabel baru bernama "nasabah" dalam Database Bank_Itera dan mengisi nilai pada setiap kolom sesuai dengan format yang diberikan.

2. Masukkan data berikut ke dalam tabel “rekening”.

no_rekening	Kode_cabang	Pin	Saldo
145678	SGS02	908723	1000000
256132	KKM70	340987	230000
247182	JKT48	091298	4000000
350067	RGS02	230987	5500000
114115	SGS01	203010	500000
100111	KKM69	213671	998500

Instruksi tersebut meminta untuk memasukkan data yang diberikan sesuai contoh pada modul ke dalam tabel "rekening" dalam Database Bank_Itera.

3. Masukkan data berikut pada tabel "transaksi".

No_transaksi	No_rekening	Jenis_transaksi	Tgl_transaksi	Jumlah
11	100111	ATM	2022-06-09 08:00:45	100000
12	114115	ATM	2022-06-09 11:10:12	200000
13	114115	DEBET	2022-06-09 13:00:30	50000
14	256132	DEBET	2022-06-15 11:20:00	150000
15	145678	ATM	2022-06-18 23:11:22	250000
16	145678	DEBET	2022-06-18 15:30:08	300000
17	350067	ATM	2022-06-18 10:02:45	100000

Instruksi tersebut meminta untuk memasukkan data yang diberikan sesuai contoh pada modul ke dalam tabel "transaksi" dalam Database Bank_Itera.

4. Dengan menggunakan cross join, tampilkan data dari nasabah dan rekening yang sesuai (sama).

Perintah tersebut meminta untuk menampilkan data dari tabel "nasabah" dan "rekening" yang sesuai (sama) dengan menggunakan cross join.

5. Dengan menggunakan Inner join, tampilkan data dari rekening dan transaksi yang sesuai(sama).

Perintah tersebut meminta untuk menampilkan data dari tabel "rekening" dan "transaksi" yang sesuai (sama) dengan menggunakan inner join.

6. Dengan menggunakan natural Join, tampilkan nasabah dan rekening.

Perintah tersebut meminta untuk menampilkan data dari tabel "nasabah" dan "rekening" yang memiliki kolom-kolom dengan nama yang sama secara otomatis menggunakan natural join.

7. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan transaksi ketika jumlah transaksi > 100000.

Perintah tersebut bertujuan untuk menampilkan data dari tabel "rekening" dan "transaksi" yang memiliki jumlah transaksi lebih dari 100000 menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

8. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data nasabah dan rekening ketika saldo rekening>1500000.

Perintah tersebut bertujuan untuk menampilkan data dari tabel "nasabah" dan "rekening" yang memiliki saldo rekening lebih dari 1500000 menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

9. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan nasabah ketika kode cabang "KKM" dan saldo<1000000.

Perintah tersebut bertujuan untuk menampilkan data rekening dan nasabah yang memiliki kode cabang "KKM" dan saldo kurang dari 1000000 menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

10. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan nasabah ketika kode cabang "KKM" dan saldo<1000000.

Perintah tersebut bertujuan untuk menampilkan data rekening dan nasabah yang memiliki kode cabang "KKM" dan saldo kurang dari 1000000 menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

11. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nomor rekening, saldo, jumlah transaksi, tanggal transaksi ketika nomor rekening sama dengan 145678.

Perintah tersebut bertujuan untuk menampilkan nomor rekening, saldo, jumlah transaksi, dan tanggal transaksi saat nomor rekeningnya sama dengan 145678, menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

12. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan transaksi ketika nomor rekening “1” dan jenis transaksi menggunakan “DEBET”.

Perintah tersebut dimaksudkan untuk menampilkan data rekening dan transaksi saat nomor rekeningnya adalah "1" dan jenis transaksinya adalah "DEBET", menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

13. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika nomor rekening kode cabang “SGS” dan nomor telepon “0812”.

Perintah tersebut bertujuan untuk menampilkan data nama nasabah, alamat, nomor telepon, nomor rekening, dan kode cabang ketika nomor rekening memiliki kode cabang "SGS" dan nomor telepon dimulai dengan "0812", menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

3. Hasil dan Jawaban

1. Soal 1 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> create table nasabah(
  → id_nasabah int(11) primary key,
  → no_rekening varchar(6),
  → nama_nasabah varchar(50),
  → alamat varchar(255),
  → no_telepon varchar(15)
  → );
Query OK, 0 rows affected (0.009 sec)

MariaDB [bank_itera]> alter table nasabah add foreign key(no_rekening) references rekening(no_rekening);
Query OK, 0 rows affected (0.039 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> insert into nasabah values
  → (1, '145678', 'Puspa Sari', 'Sei Batang Kuis', '081376809122'),
  → (2, '256132', 'Sinta Jojo', 'Iskandar Muda', '089987654311'),
  → (3, '247182', 'Bunga Putri', 'Jalan Martapura', '081249987011'),
  → (4, '350067', 'Grace Sebayang', 'Gang Keluarga', '085276812453'),
  → (5, '114115', 'Christine Talia', 'Sei Kambah', '082266778866'),
  → (6, '100111', 'Rimayka', 'Perumahan Puri K', '081234567890');
Query OK, 6 rows affected (0.003 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

Tulisan di atas merupakan perintah SQL untuk membuat tabel "nasabah" dengan kolom id_nasabah sebagai primary key dan kolom no_rekening sebagai foreign key yang merujuk ke tabel "rekening".

Perintah di atas pada gambar adalah contoh perintah SQL untuk menyisipkan data ke dalam tabel "nasabah" dengan beberapa kolom seperti id_nasabah, no_rekening, nama_nasabah, alamat, dan no_telepon.

2. Soal 2 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select * from rekening;
```

no_rekening	kode_cabang	pin	saldo
100111	KKM69	213671	998500
114115	SGS01	203010	500000
120023	JGJ01	837242	500000
140932	JOJ02	024422	1200000
145678	SGS02	908723	1000000
220983	SMG12	450817	5422000
247182	JKT48	091298	4000000
256132	KKM70	340987	230000
332102	JKT48	228883	400500
350067	RGS02	230987	5500000

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

Perintah di atas adalah contoh perintah SQL untuk menampilkan data tabel "rekening" dengan kolom no_rekening, kode_bank, no_pin, dan saldo.

3. Soal 3 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select * from transaksi;
```

no_transaksi	no_rekening	jenis_transaksi	tgl_transaksi	jumlah
1	220983	DEBET	2022-06-12 12:00:45	500000
2	220983	DEBET	2022-06-13 08:20:33	2150000
3	140932	ATM	2022-06-14 10:56:52	100000
4	140932	ATM	2022-06-14 15:16:32	100000
5	120087	DEBET	2022-06-15 15:03:33	1250000
6	220983	ATM	2022-06-17 20:34:22	150000
7	220983	ATM	2022-06-15 15:40:12	100000
8	220983	DEBET	2022-06-16 08:23:23	2500000
11	100111	ATM	2022-06-09 08:00:45	100000
12	114115	ATM	2022-06-09 11:10:12	200000
13	114115	DEBET	2022-06-09 13:00:30	50000
14	256132	DEBET	2022-06-15 11:20:00	150000
15	145678	ATM	2022-06-18 23:11:22	250000
16	145678	DEBET	2022-06-18 15:30:08	300000
17	350067	ATM	2022-06-18 10:02:45	100000

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

Perintah di atas adalah contoh perintah SQL untuk menampilkan data tabel "transaksi" dengan kolom id_transaksi, no_rekening, jenis_transaksi, waktu_transaksi, dan jumlah_transaksi.

4. Soal 4 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select id_nasabah, nasabah.no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, saldo from nasabah cross join rekening;
```

	id_nasabah	no_rekening	nama_nasabah	alamat	no_telepon	kode_cabang	pin	saldo
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	KKM69	213671	998500	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	KKM69	213671	998500	
3	247182	Bunga Putri	Jalan Martapura	081249987011	KKM69	213671	998500	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	KKM69	213671	998500	
5	114115	Christine Talia	Sei Kambah	082266778866	KKM69	213671	998500	
6	100111	Rimayka	Perumahan Puri K	081234567890	KKM69	213671	998500	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	SSS01	203010	500000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	SSS01	203010	500000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	SSS01	203010	500000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	SSS01	203010	500000	
5	114115	Christine Talia	Sei Kambah	082266778866	SSS01	203010	500000	
6	100111	Rimayka	Perumahan Puri K	081234567890	SSS01	203010	500000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	JKJ01	837242	500000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	JKJ01	837242	500000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKJ01	837242	500000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	JKJ01	837242	500000	
5	114115	Christine Talia	Sei Kambah	082266778866	JKJ01	837242	500000	
6	100111	Rimayka	Perumahan Puri K	081234567890	JKJ01	837242	500000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	JKJ02	024422	1200000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	JKJ02	024422	1200000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKJ02	024422	1200000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	JKJ02	024422	1200000	
5	114115	Christine Talia	Sei Kambah	082266778866	JKJ02	024422	1200000	
6	100111	Rimayka	Perumahan Puri K	081234567890	JKJ02	024422	1200000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	SSS02	908723	1000000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	SSS02	908723	1000000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	SSS02	908723	1000000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	SSS02	908723	1000000	
5	114115	Christine Talia	Sei Kambah	082266778866	SSS02	908723	1000000	
6	100111	Rimayka	Perumahan Puri K	081234567890	SSS02	908723	1000000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	SMG12	450817	5422000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	SMG12	450817	5422000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	SMG12	450817	5422000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	SMG12	450817	5422000	
5	114115	Christine Talia	Sei Kambah	082266778866	SMG12	450817	5422000	
6	100111	Rimayka	Perumahan Puri K	081234567890	SMG12	450817	5422000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	JKT48	091298	4000000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	JKT48	091298	4000000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKT48	091298	4000000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	JKT48	091298	4000000	
5	114115	Christine Talia	Sei Kambah	082266778866	JKT48	091298	4000000	
6	100111	Rimayka	Perumahan Puri K	081234567890	JKT48	091298	4000000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	KKM70	340987	230000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	KKM70	340987	230000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	KKM70	340987	230000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	KKM70	340987	230000	
5	114115	Christine Talia	Sei Kambah	082266778866	KKM70	340987	230000	
6	100111	Rimayka	Perumahan Puri K	081234567890	KKM70	340987	230000	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	JKT48	228883	400500	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	JKT48	228883	400500	
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKT48	228883	400500	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	JKT48	228883	400500	
5	114115	Christine Talia	Sei Kambah	082266778866	JKT48	228883	400500	
6	100111	Rimayka	Perumahan Puri K	081234567890	JKT48	228883	400500	
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	RSS02	230987	5500000	
2	256132	Sinta Jojo	Iskandar Muda	089987654311	RSS02	230987	5500000	
3	247182	Bunga Putri	Jalan Martapura	081249987011	RSS02	230987	5500000	
4	350067	Grace Sebayang	Gang Keluarga	085276812453	RSS02	230987	5500000	
5	114115	Christine Talia	Sei Kambah	082266778866	RSS02	230987	5500000	
6	100111	Rimayka	Perumahan Puri K	081234567890	RSS02	230987	5500000	

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

Perintah SQL di atas adalah contoh penggunaan CROSS JOIN untuk menggabungkan data dari tabel "nasabah" dan "rekening" berdasarkan kombinasi setiap baris dalam kedua tabel, dengan hasil yang mencakup kolom yang dipilih seperti id_nasabah, no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, dan saldo.

5. Soal 5 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select rekening.no_rekening, kode_cabang, pin, saldo, no_transaksi, jenis_transaksi, tgl_transaksi,
jumlah from rekening inner join transaksi where rekening.no_rekening = transaksi.no_rekening;
```

no_rekening	kode_cabang	pin	saldo	no_transaksi	jenis_transaksi	tgl_transaksi	jumlah
220983	SMG12	450817	5422000	1	DEBET	2022-06-12 12:00:45	500000
220983	SMG12	450817	5422000	2	DEBET	2022-06-13 08:20:33	2150000
140932	JOJ02	024422	1200000	3	ATM	2022-06-14 10:56:52	100000
140932	JOJ02	024422	1200000	4	ATM	2022-06-14 15:16:32	100000
220983	SMG12	450817	5422000	6	ATM	2022-06-17 20:34:22	150000
220983	SMG12	450817	5422000	7	ATM	2022-06-15 15:40:12	100000
220983	SMG12	450817	5422000	8	DEBET	2022-06-16 08:23:23	2500000
100111	KKM69	213671	998500	11	ATM	2022-06-09 08:00:45	100000
114115	SGS01	203010	500000	12	ATM	2022-06-09 11:10:12	200000
114115	SGS01	203010	500000	13	DEBET	2022-06-09 13:00:30	50000
256132	KKM70	340987	230000	14	DEBET	2022-06-15 11:20:00	150000
145678	SGS02	908723	1000000	15	ATM	2022-06-18 23:11:22	250000
145678	SGS02	908723	1000000	16	DEBET	2022-06-18 15:30:08	300000
350067	RGS02	230987	5500000	17	ATM	2022-06-18 10:02:45	100000

14 rows in set (0.000 sec)

Perintah SQL di atas menggunakan INNER JOIN untuk menggabungkan data dari tabel "rekening" dan "transaksi" berdasarkan kondisi di mana nomor rekening pada kedua tabel harus sama, dengan hasil yang mencakup kolom-kolom yang dipilih seperti no_rekening, kode_cabang, pin, saldo, no_transaksi, jenis_transaksi, tgl_transaksi, dan jumlah.

6. Soal 6 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select id_nasabah, nasabah.no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, saldo f
rom nasabah natural join rekening;
```

id_nasabah	no_rekening	nama_nasabah	alamat	no_telepon	kode_cabang	pin	saldo
1	145678	Puspa Sari	Sei Batang Kuis	081376809122	SGS02	908723	1000000
2	256132	Sinta Jojo	Iskandar Muda	089987654311	KKM70	340987	230000
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKT48	091298	4000000
4	350067	Grace Sebayang	Gang Keluarga	085276812453	RGS02	230987	5500000
5	114115	Christine Talia	Sei Kambah	082266778866	SGS01	203010	500000
6	100111	Rimayka	Perumahan Puri K	081234567890	KKM69	213671	998500

6 rows in set (0.000 sec)

Perintah SQL di atas menggunakan NATURAL JOIN untuk menggabungkan data dari tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis, menghasilkan data yang mencakup kolom-kolom seperti id_nasabah, no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, dan saldo.

7. Soal 7 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select rekening.no_rekening, kode_cabang, pin, saldo, no_transaksi, jenis_transaksi, tgl_transaksi,
jumlah from rekening natural join transaksi where jumlah > 100000;
```

no_rekening	kode_cabang	pin	saldo	no_transaksi	jenis_transaksi	tgl_transaksi	jumlah
220983	SMG12	450817	5422000	1	DEBET	2022-06-12 12:00:45	500000
220983	SMG12	450817	5422000	2	DEBET	2022-06-13 08:20:33	2150000
220983	SMG12	450817	5422000	6	ATM	2022-06-17 20:34:22	150000
220983	SMG12	450817	5422000	8	DEBET	2022-06-16 08:23:23	2500000
114115	SGS01	203010	500000	12	ATM	2022-06-09 11:10:12	200000
256132	KKM70	340987	230000	14	DEBET	2022-06-15 11:20:00	150000
145678	SGS02	908723	1000000	15	ATM	2022-06-18 23:11:22	250000
145678	SGS02	908723	1000000	16	DEBET	2022-06-18 15:30:08	300000

8 rows in set (0.001 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "transaksi" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Hasilnya adalah data yang mencakup kolom no_rekening, kode_cabang, pin, saldo, no_transaksi, jenis_transaksi, tgl_transaksi, dan jumlah, di mana jumlah transaksi lebih dari 100000.

8. Soal 8 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select id_nasabah, nasabah.no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, saldo from nasabah natural join rekening where saldo > 1500000;
```

id_nasabah	no_rekening	nama_nasabah	alamat	no_telepon	kode_cabang	pin	saldo
3	247182	Bunga Putri	Jalan Martapura	081249987011	JKT48	091298	4000000
4	350067	Grace Sebayang	Gang Keluarga	085276812453	RGS02	230987	5500000

2 rows in set (0.000 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom id_nasabah, no_rekening, nama_nasabah, alamat, no_telepon, kode_cabang, pin, dan saldo, dengan kondisi bahwa saldo rekening lebih dari 1500000.

9. Soal 9 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select rekening.no_rekening, kode_cabang, pin, saldo, id_nasabah, alamat, no_telepon from rekening natural join nasabah where kode_cabang like "KKM%" and saldo < 1000000;
```

no_rekening	kode_cabang	pin	saldo	id_nasabah	alamat	no_telepon
256132	KKM70	340987	230000	2	Iskandar Muda	089987654311
100111	KKM69	213671	998500	6	Perumahan Puri K	081234567890

2 rows in set (0.001 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "nasabah" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom no_rekening, kode_cabang, pin, saldo, id_nasabah, alamat, dan no_telepon, dengan kondisi bahwa kode cabang dimulai dengan "KKM%" dan saldo rekening kurang dari 1000000.

10. Soal 10 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select nama_nasabah, alamat, no_telepon, nasabah.no_rekening, kode_cabang from nasabah natural join rekening where alamat like "Se%";
```

nama_nasabah	alamat	no_telepon	no_rekening	kode_cabang
Puspa Sari	Sei Batang Kuis	081376809122	145678	S6S02
Christine Talia	Sei Kambah	082266778866	114115	S6S01

2 rows in set (0.000 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom nama_nasabah, alamat,

no_telepon, no_rekening, dan kode_cabang, dengan kondisi bahwa alamat dimulai dengan "Se%".

11. Soal 11 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select rekening.no_rekening, saldo, count(jumlah) as "jumlah transaksi", date(tgl_transaksi) as "tanggal transaksi" from rekening natural join transaksi where rekening.no_rekening = 145678;
```

no_rekening	saldo	jumlah transaksi	tanggal transaksi
145678	1000000	2	2022-06-18

1 row in set (0.001 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "transaksi" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom no_rekening, saldo, jumlah transaksi (dihitung dengan fungsi COUNT), dan tanggal transaksi (diekstraksi dengan fungsi DATE), dengan kondisi bahwa nomor rekening adalah 145678.

12. Soal 12 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select rekening.no_rekening, kode_cabang, pin, saldo, no_transaksi, tgl_transaksi, jenis_transaksi, jumlah from rekening natural join transaksi where rekening.no_rekening like "1%" and jenis_transaksi = "DEBET";
```

no_rekening	kode_cabang	pin	saldo	no_transaksi	tgl_transaksi	jenis_transaksi	jumlah
114115	SGS01	203010	500000	13	2022-06-09 13:00:30	DEBET	50000
145678	SGS02	908723	1000000	16	2022-06-18 15:30:08	DEBET	300000

2 rows in set (0.000 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "transaksi" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom-kolom yang dipilih (no_rekening, kode_cabang, pin, saldo, no_transaksi, tgl_transaksi, jenis_transaksi, jumlah), dengan kondisi bahwa nomor rekening dimulai dengan angka 1 (LIKE "1%") dan jenis transaksi adalah "DEBET".

13. Soal 13 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [bank_itera]> select nama_nasabah, alamat, no_telepon, nasabah.no_rekening, kode_cabang from nasabah natural join rekening where kode_cabang like "SGS%" and no_telepon like "0812%";
```

Empty set (0.001 sec)

Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup nama nasabah, alamat, nomor telepon, nomor rekening nasabah, dan kode cabang, dengan kondisi bahwa kode cabang dimulai dengan "SGS" (LIKE "SGS%") dan nomor telepon dimulai dengan "0812" (LIKE "0812%"). Namun saat di enter, database menunjukkan *empty set*.

4. Kesimpulan dan Saran

Dari pengalaman praktikum JOIN di Database Bank_Itera, saya menyimpulkan bahwa penggunaan beragam jenis join dalam SQL sangatlah vital dalam menggabungkan data dari berbagai tabel dengan kriteria yang relevan. Saya merekomendasikan untuk terus mempraktikkan penggunaan cross join, inner join, dan natural join agar semakin mahir dalam mengelola data nasabah, rekening, dan transaksi. Lebih lanjut, sangat disarankan untuk terus mengasah kemampuan menggunakan klausa WHERE dalam jenis join tertentu untuk melakukan filter data sesuai dengan kebutuhan analisis, seperti menampilkan data transaksi dengan jumlah tertentu atau data rekening dengan saldo di atas batas tertentu. Dengan pemahaman yang kuat tentang bahasa SQL akan membantu dalam mengelola dan menganalisis data dengan lebih efisien dan akurat, mendukung pengambilan keputusan yang lebih baik dalam pembuatan database.