**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](#_Toc162638976)

[**1.** **Dasar Teori** 3](#_Toc162638977)

[**2.** **Ulasan** 3](#_Toc162638978)

[**3.** **Hasil dan Jawaban** 6](#_Toc162638979)

[**4.** **Kesimpulan dan Saran** 12](#_Toc162638980)

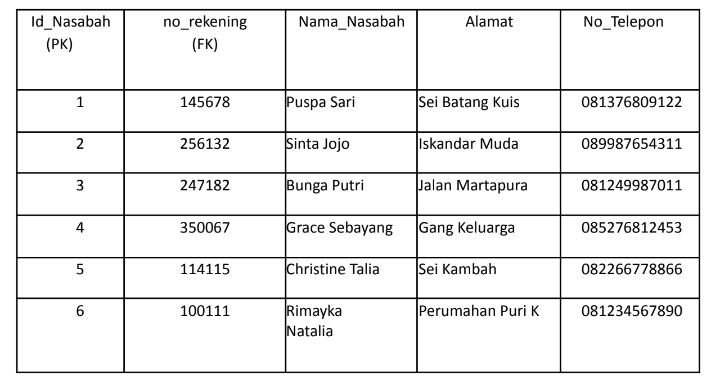
# 

# **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.

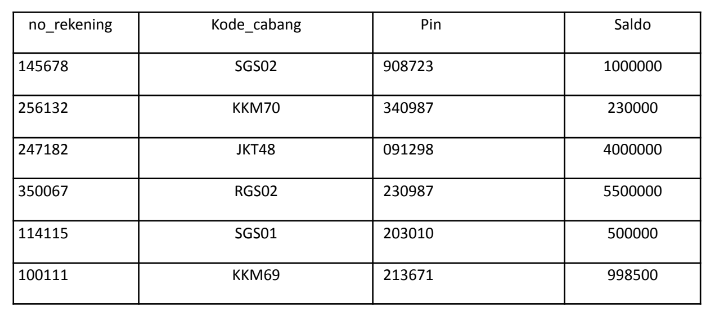
# **Ulasan**

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



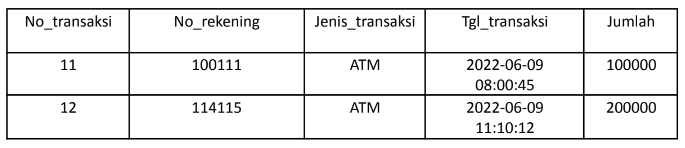
Instruksi tersebut meminta untuk membuat tabel baru bernama "nasabah" dalam Database Bank\_Itera dan mengisi nilai pada setiap kolom sesuai dengan format yang diberikan.

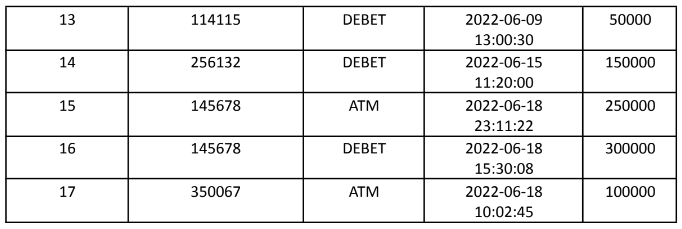
1. Masukkan data berikut ke dalam tabel “rekening”.



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

1. Masukkan data berikut pada tabel “transaksi”.





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

1. 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.

1. 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.

1. 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.

1. 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.

1. 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.

1. 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.

1. 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.

1. 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.

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

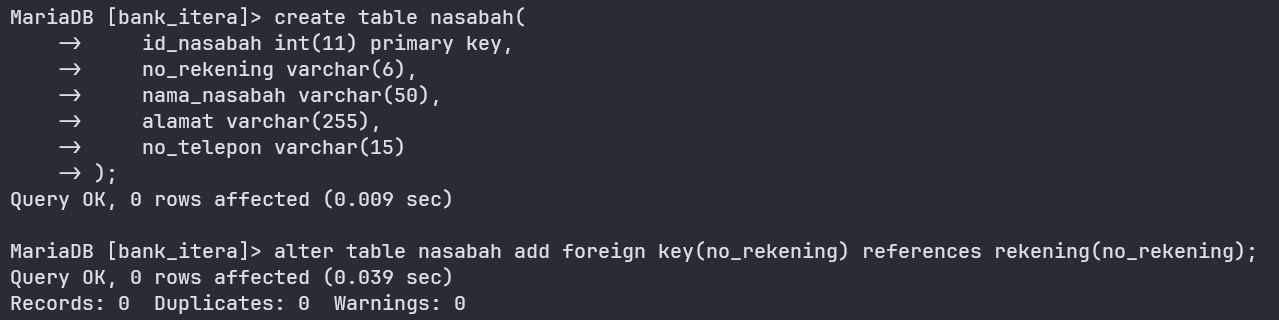
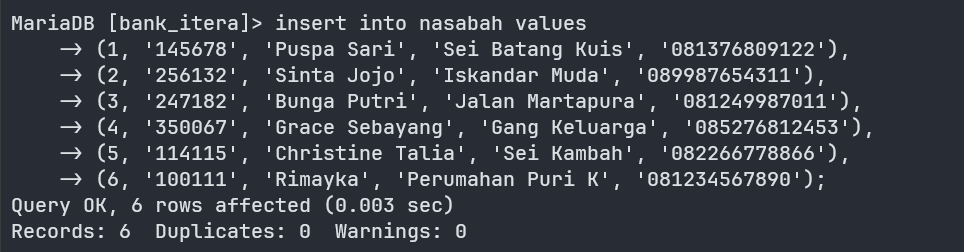
Perintah tersebut dimaksudkan untuk menampilkan data rekening dan transaksi saat nomor rekeningnya adalah "1" dan jenis transaksinya adalah "DEBET", menggunakan salah satu jenis join yang sesuai dengan kriteria tersebut.

1. 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.

# **Hasil dan Jawaban**

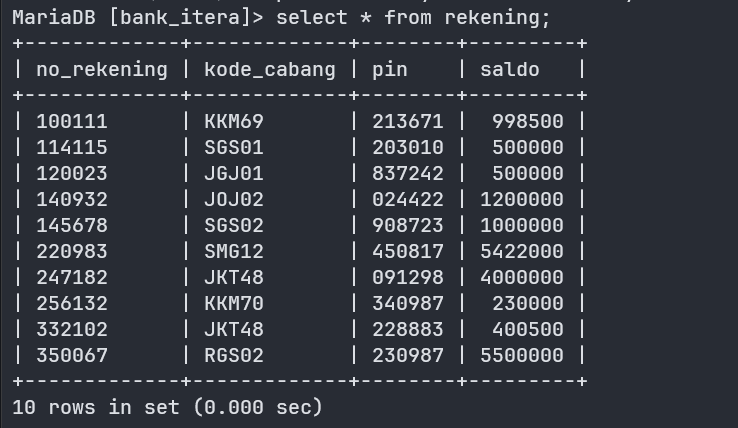
1. Soal 1 Screenshoot hasil dan jawaban dari pengujian



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".

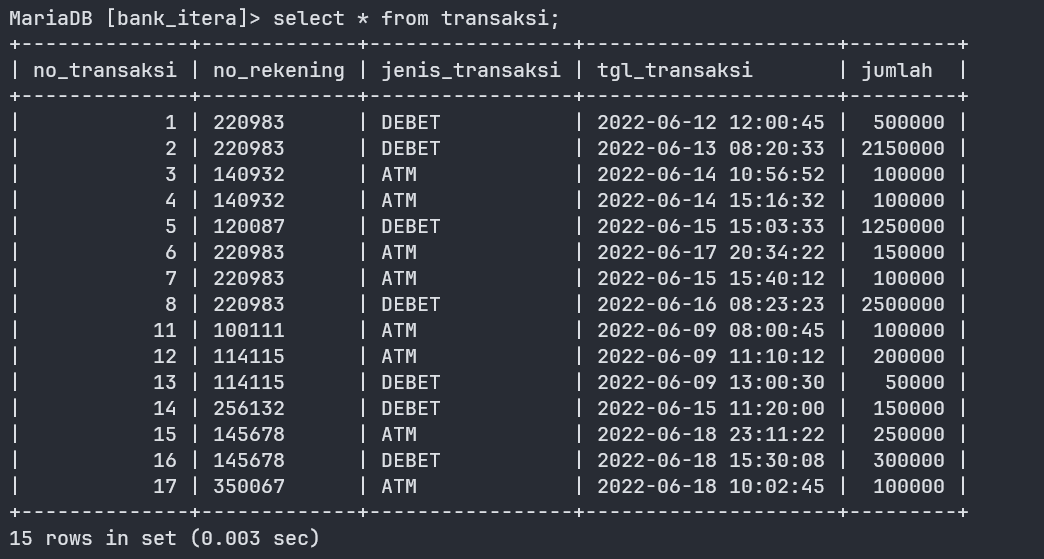
Perintah di atas pada gambar adalah contoh perintah SQL untuk menyisipkan data ke dalam tabel "nasabah" dengan beberapa kolom seperti id\_nasabah, no\_rekening, nama\_nasabah, alamat, dan no\_telepon.

1. Soal 2 Screenshoot hasil dan jawaban dari pengujian



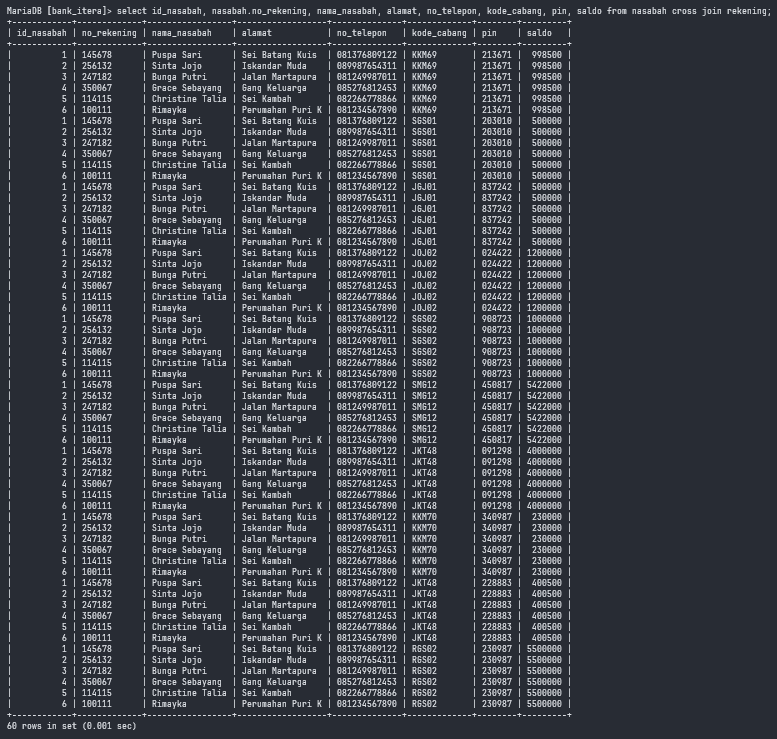
Perintah di atas adalah contoh perintah SQL untuk menampilkan data tabel "rekening" dengan kolom no\_rekening, kode\_bank, no\_pin, dan saldo.

1. Soal 3 Screenshoot hasil dan jawaban dari pengujian



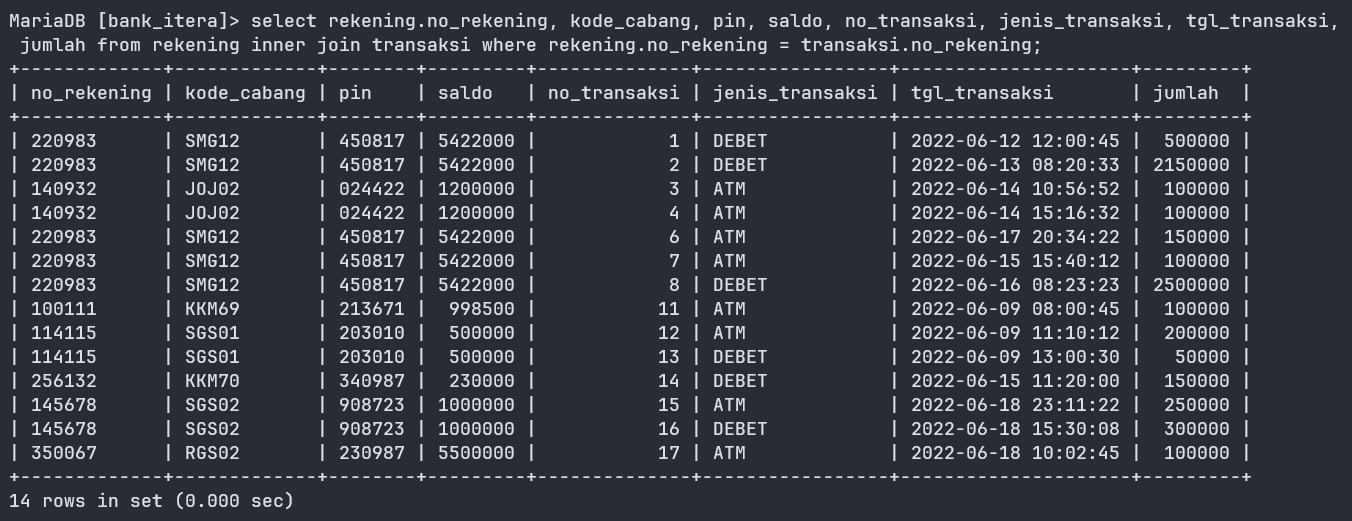
Perintah di atas adalah contoh perintah SQL untuk menampilkan data tabel "transaksi" dengan kolom id\_transaksi, no\_rekening, jenis\_transaksi, waktu\_transaksi, dan jumlah\_transaksi.

1. Soal 4 Screenshoot hasil dan jawaban dari pengujian



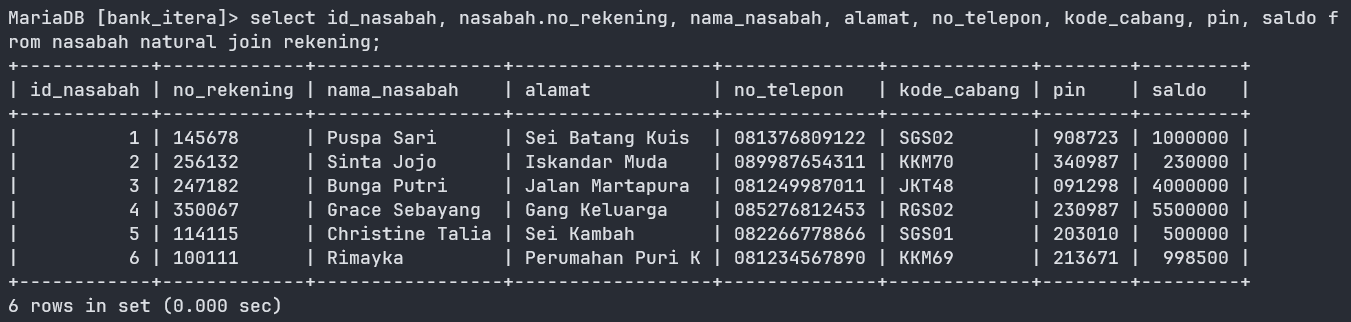
Perintah SQL di atas adalah contoh penggunaan CROSS JOIN untuk menggabungkan data dari tabel "nasabah" dan "rekening" berdasarkan kombinasi setiap baris dalam kedua tabel, dengan hasil yang mencakup kolom yang dipilih seperti id\_nasabah, no\_rekening, nama\_nasabah, alamat, no\_telepon, kode\_cabang, pin, dan saldo.

1. Soal 5 Screenshoot hasil dan jawaban dari pengujian



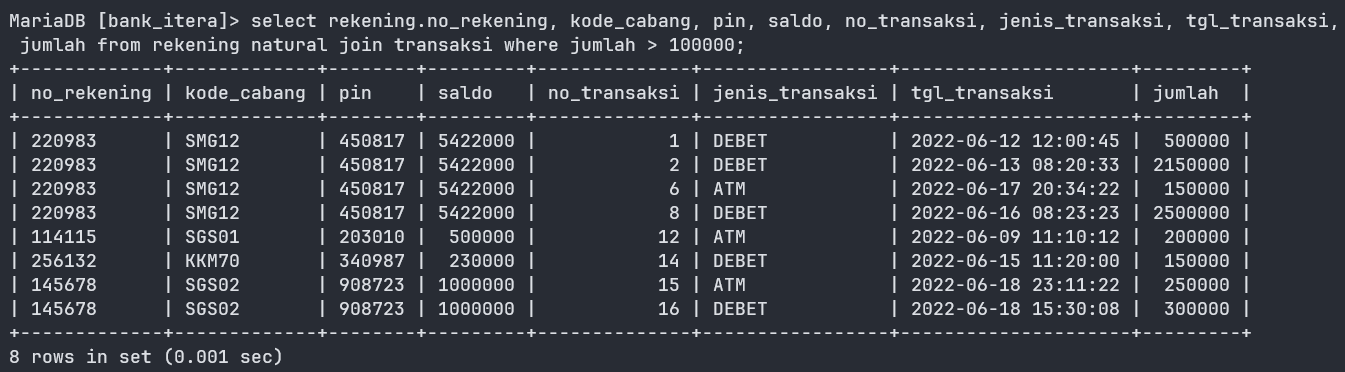
Perintah SQL di atas menggunakan INNER JOIN untuk menggabungkan data dari tabel "rekening" dan "transaksi" berdasarkan kondisi di mana nomor rekening pada kedua tabel harus sama, dengan hasil yang mencakup kolom-kolom yang dipilih seperti no\_rekening, kode\_cabang, pin, saldo, no\_transaksi, jenis\_transaksi, tgl\_transaksi, dan jumlah.

1. Soal 6 Screenshoot hasil dan jawaban dari pengujian



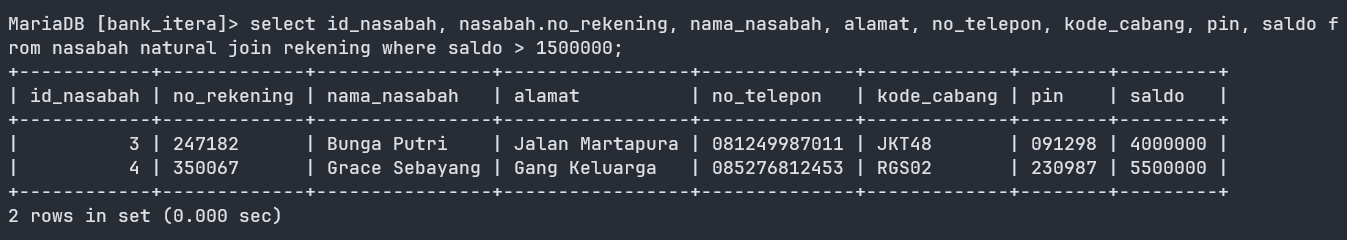
Perintah SQL di atas menggunakan NATURAL JOIN untuk menggabungkan data dari tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis, menghasilkan data yang mencakup kolom-kolom seperti id\_nasabah, no\_rekening, nama\_nasabah, alamat, no\_telepon, kode\_cabang, pin, dan saldo.

1. Soal 7 Screenshoot hasil dan jawaban dari pengujian



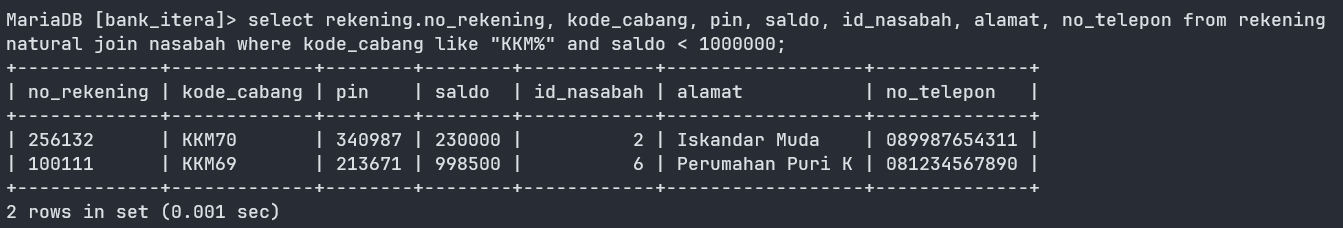
Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "transaksi" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Hasilnya adalah data yang mencakup kolom no\_rekening, kode\_cabang, pin, saldo, no\_transaksi, jenis\_transaksi, tgl\_transaksi, dan jumlah, di mana jumlah transaksi lebih dari 100000.

1. Soal 8 Screenshoot hasil dan jawaban dari pengujian



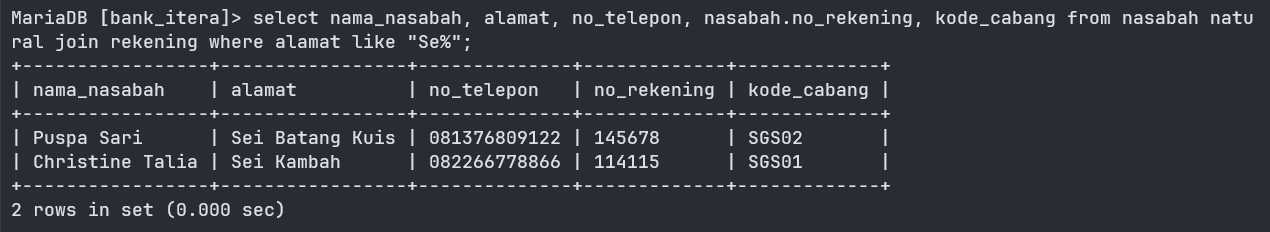
Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom id\_nasabah, no\_rekening, nama\_nasabah, alamat, no\_telepon, kode\_cabang, pin, dan saldo, dengan kondisi bahwa saldo rekening lebih dari 1500000.

1. Soal 9 Screenshoot hasil dan jawaban dari pengujian



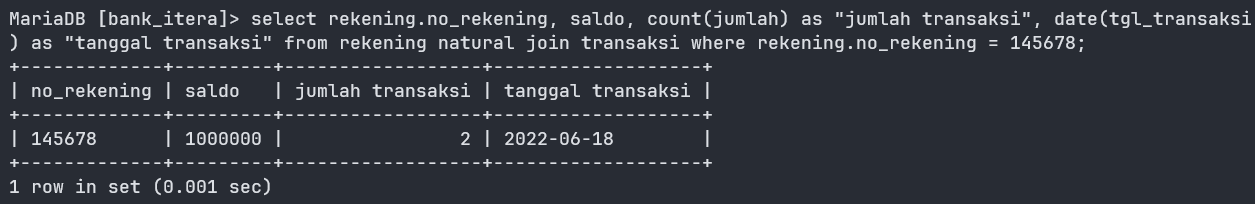
Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "nasabah" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom no\_rekening, kode\_cabang, pin, saldo, id\_nasabah, alamat, dan no\_telepon, dengan kondisi bahwa kode cabang dimulai dengan "KKM%" dan saldo rekening kurang dari 1000000.

1. Soal 10 Screenshoot hasil dan jawaban dari pengujian



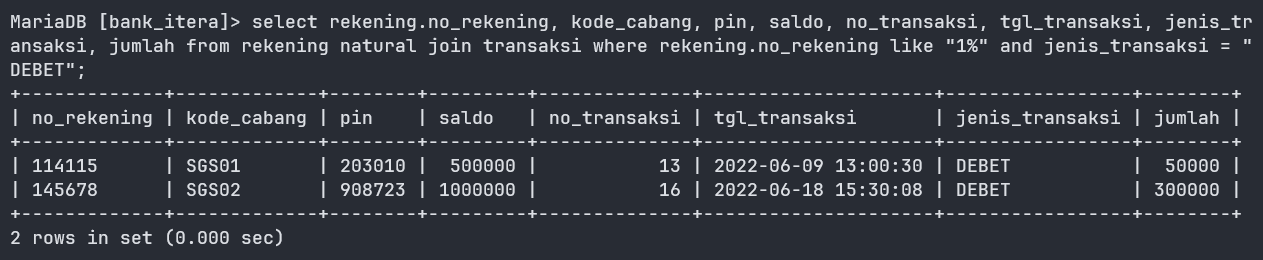
Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "nasabah" dan "rekening" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom nama\_nasabah, alamat, no\_telepon, no\_rekening, dan kode\_cabang, dengan kondisi bahwa alamat dimulai dengan "Se%".

1. Soal 11 Screenshoot hasil dan jawaban dari pengujian



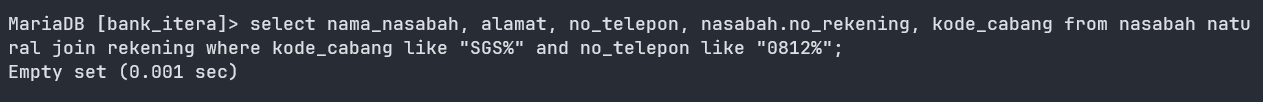
Perintah SQL di atas menggunakan NATURAL JOIN antara tabel "rekening" dan "transaksi" berdasarkan kolom-kolom yang memiliki nama yang sama dan tipe data yang cocok secara otomatis. Data yang dihasilkan mencakup kolom no\_rekening, saldo, jumlah transaksi (dihitung dengan fungsi COUNT), dan tanggal transaksi (diekstraksi dengan fungsi DATE), dengan kondisi bahwa nomor rekening adalah 145678.

1. Soal 12 Screenshoot hasil dan jawaban dari pengujian



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".

1. Soal 13 Screenshoot hasil dan jawaban dari pengujian



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 menunjukan *empty set*.

# **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.