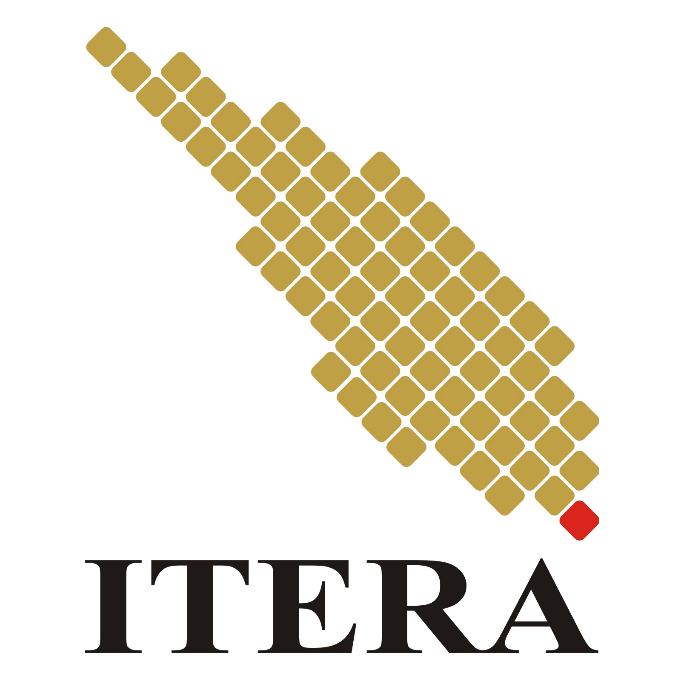
**LAPORAN PRAKTIKUM BASIS DATA RC**

**Muhammad Yusuf  
122140193**

**Latihan**

****

**Teori Dasar**

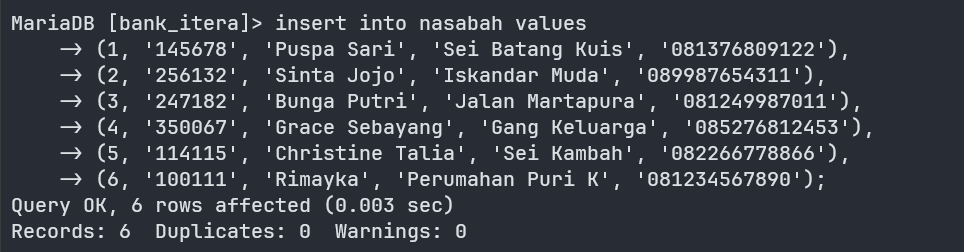
Join adalah konsep penting dalam SQL yang memungkinkan pengguna untuk menggabungkan data dari dua atau lebih tabel dalam satu query.

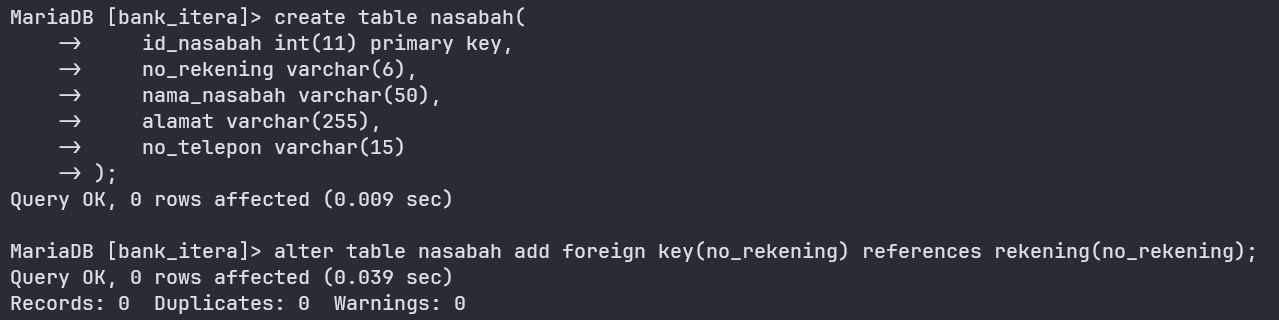
**Inner join** merupakan salah satu jenis join yang mengembalikan baris-baris yang memiliki nilai yang cocok di kedua tabel yang di-join berdasarkan kriteria tertentu. Hal ini berguna untuk mengambil informasi terkait dari tabel yang berbeda.

**Cross join** adalah jenis join lain yang menggabungkan setiap baris dari tabel pertama dengan setiap baris dari tabel kedua, menghasilkan jumlah baris yang sama dengan perkalian jumlah baris di kedua tabel tersebut. Cross join berguna dalam situasi di mana pengguna ingin menghasilkan kombinasi semua baris dari dua tabel tanpa mempertimbangkan nilai-nilai yang cocok.

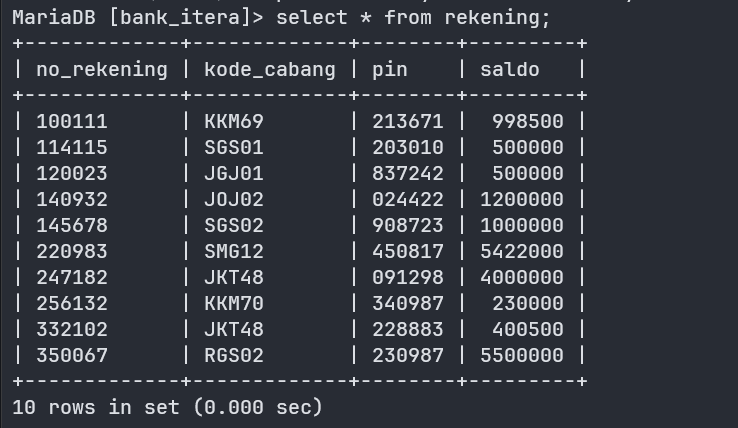
**Natural join** adalah jenis join yang menggabungkan tabel berdasarkan kolom-kolom dengan nama yang sama dan tipe data yang cocok secara otomatis, tanpa perlu menentukan kriteria join secara eksplisit. Natural join dapat mempermudah proses penggabungan data jika tabel memiliki kolom-kolom yang sama.

**Pembahasan**

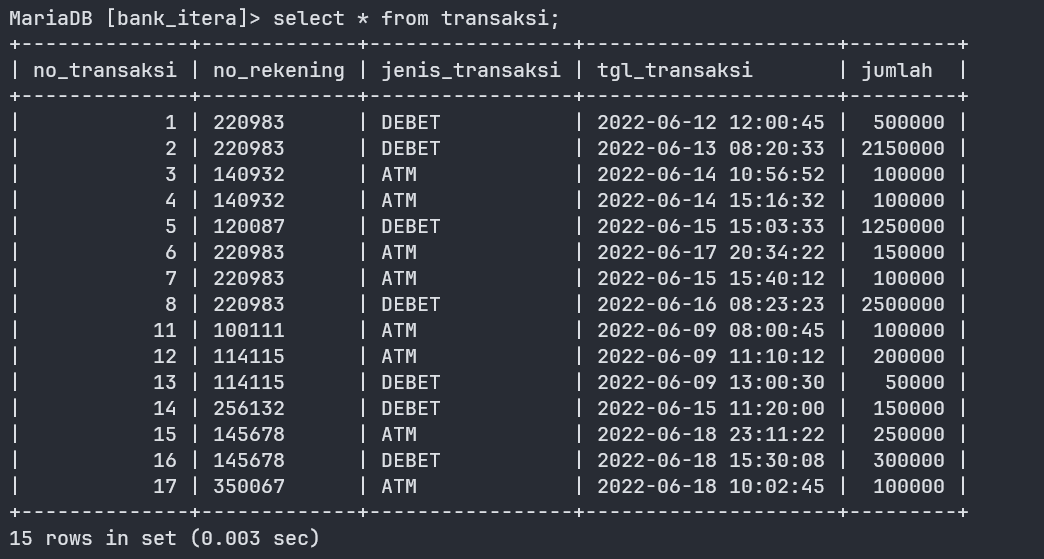
1. Buat dan Tambahkan tabel “nasabah” pada Database Bank\_Itera dan isi nilai pada setiap kolom seperti berikut.



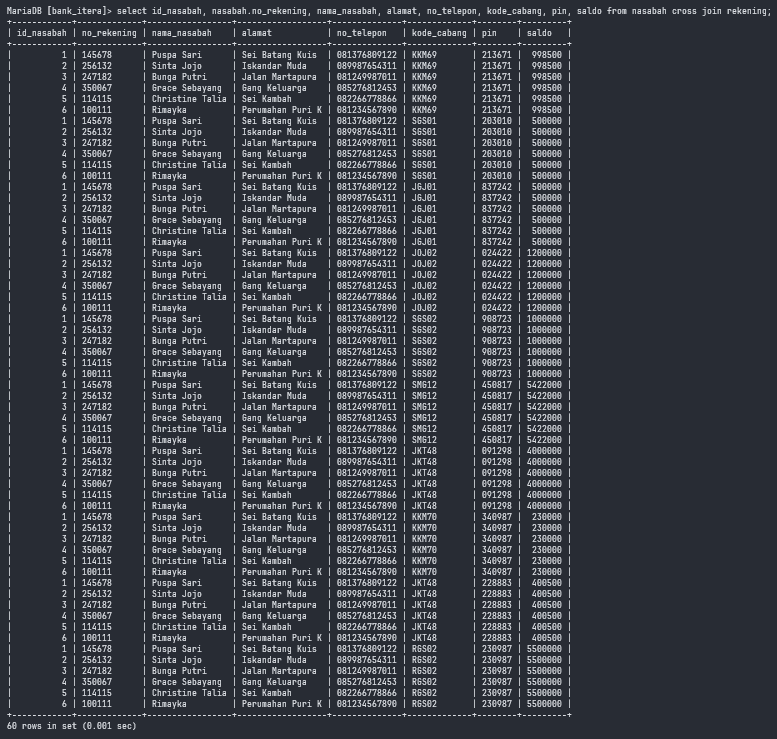
1. Masukkan data berikut ke dalam tabel “rekening”.



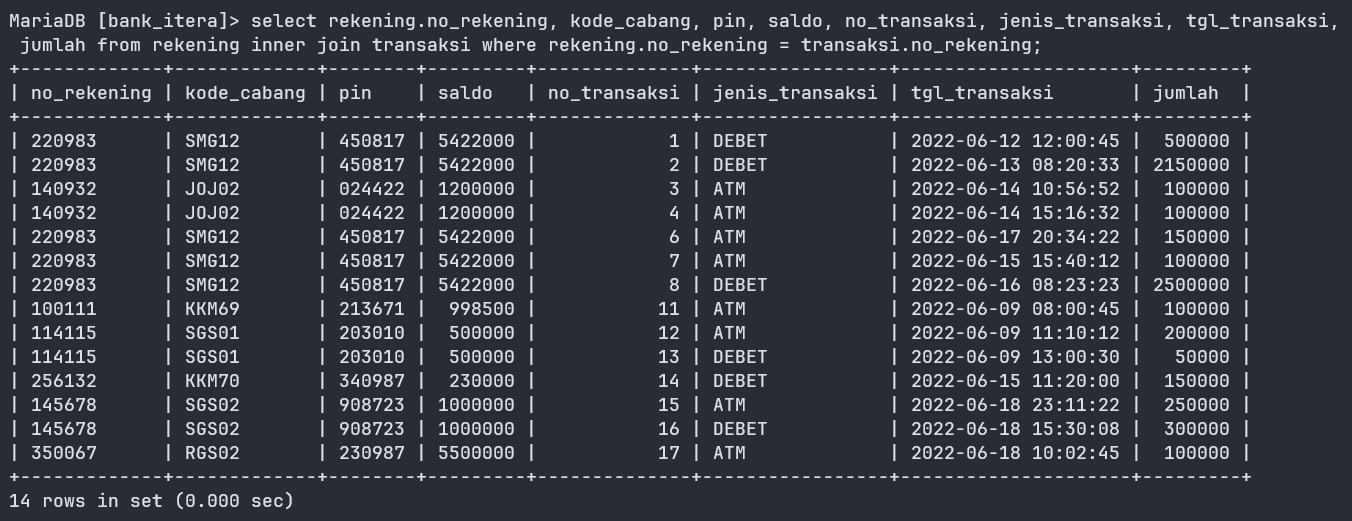
1. Masukkan data berikut pada tabel “transaksi”.



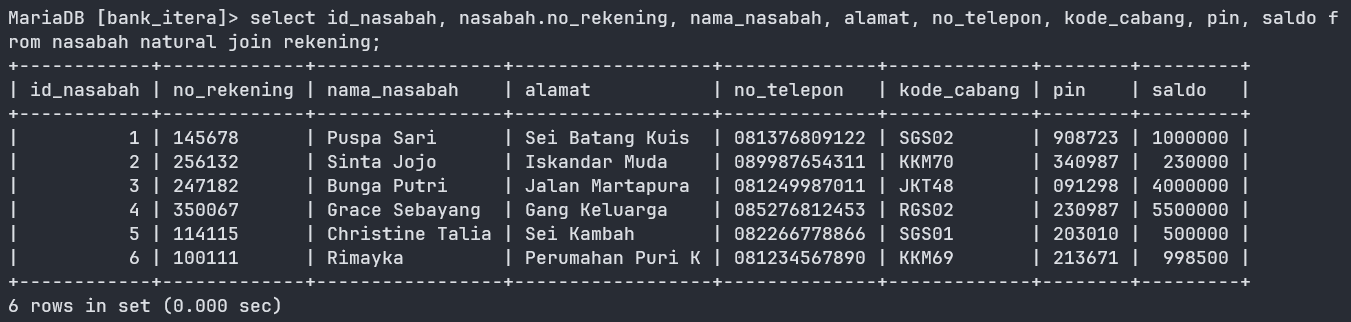
1. Dengan menggunakan cross join, tampilkan data dari nasabah dan rekening yang sesuai (sama).



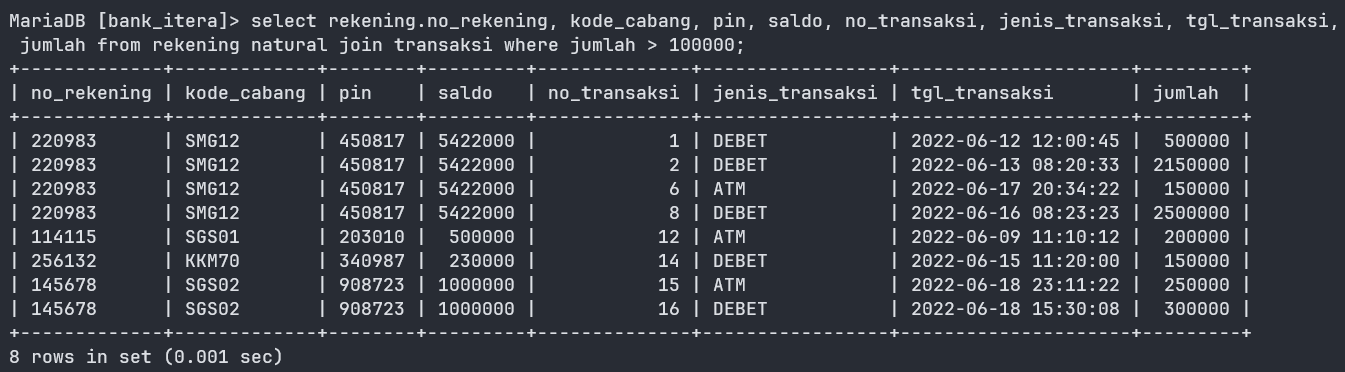
1. Dengan menggunakan Inner join, tampilkan data dari rekening dan transaksi yang sesuai(sama).



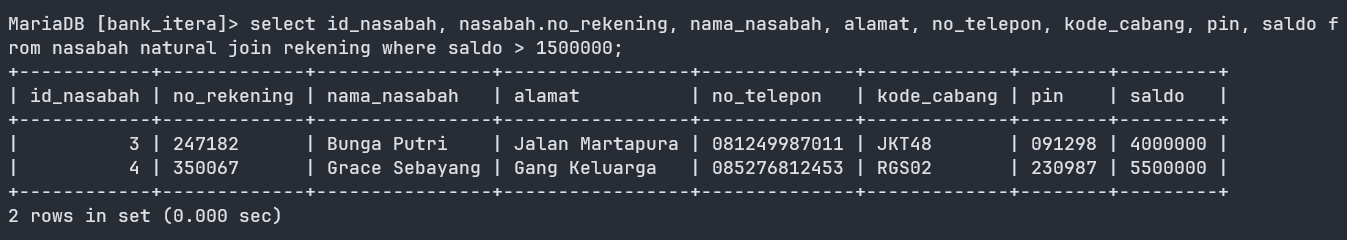
1. Dengan menggunakan natural Join, tampilkan nasabah dan rekening.



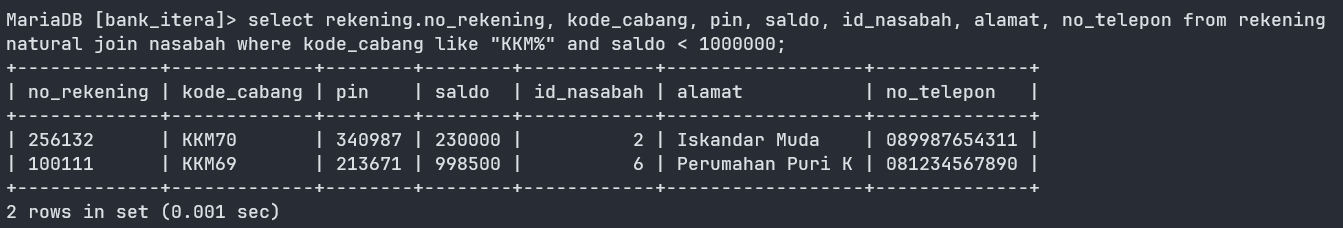
1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan transaksi ketika jumlah transaksi > 100000.



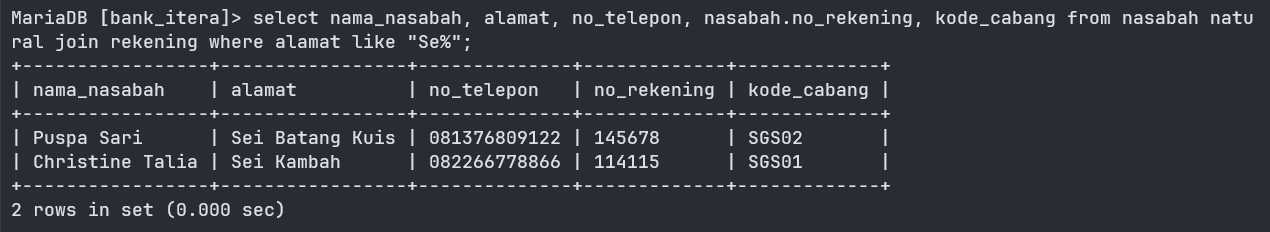
1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data nasabah dan rekening ketika saldo rekening>1500000.



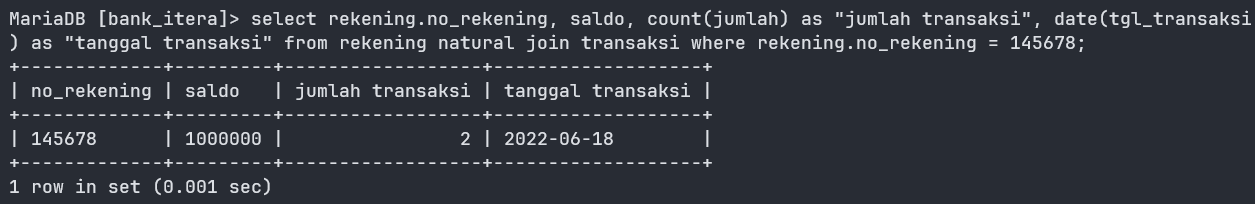
1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan nasabah ketika kode cabang “KKM” dan saldo<1000000.



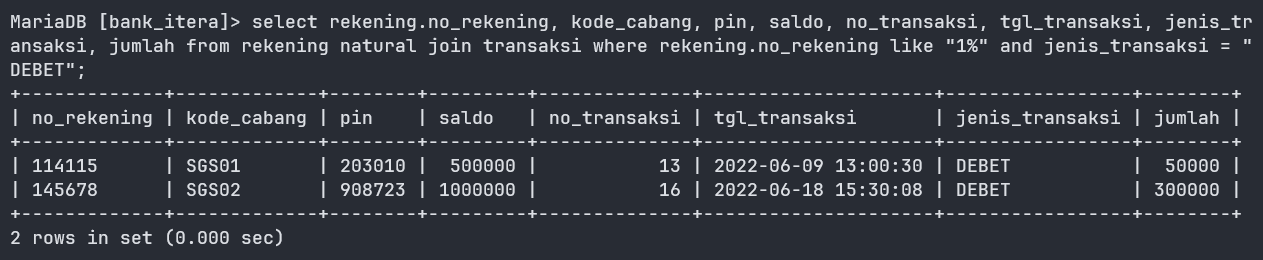
1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika alamat nasabah “Se”.



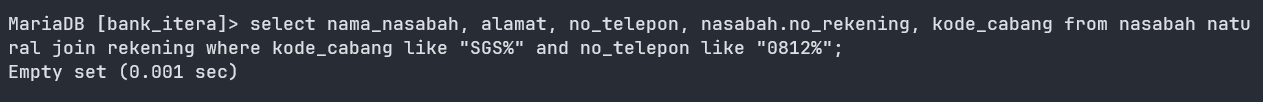
1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nomor rekening, saldo, jumlah transaksi, tanggal transaksi ketika nomor rekening sama dengan 145678.



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



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”.



**Analisis & Kesimpulan**

Saat melakukan praktikum JOIN pada Database Bank\_Itera, saya dapat melihat betapa pentingnya penggunaan berbagai jenis join dalam SQL. Melalui praktik ini, saya dapat menggabungkan data dari tabel "nasabah", "rekening", dan "transaksi" dengan berbagai kriteria yang relevan. Misalnya, dengan menggunakan cross join, saya dapat menampilkan data nasabah dan rekening yang sesuai berdasarkan nomor rekening yang sama. Selain itu, dengan inner join, saya bisa melihat data rekening dan transaksi yang memiliki keterkaitan, seperti menampilkan nomor rekening, saldo, jumlah transaksi, dan tanggal transaksi ketika nomor rekening sama dengan 145678.

Natural join juga memberikan kemudahan dalam menggabungkan data nasabah dan rekening berdasarkan hubungan kunci yang ada. Dalam analisis yang lebih mendalam, saya menggunakan klausa WHERE dalam jenis join tertentu untuk memfilter data sesuai dengan kriteria tertentu. Misalnya, saya dapat menampilkan data rekening dan transaksi ketika jumlah transaksi lebih dari 100000, atau menampilkan data nasabah dan rekening ketika saldo rekening lebih dari 1500000.

Dengan demikian, praktikum ini memberikan wawasan yang mendalam tentang penggunaan SQL dalam analisis data perbankan. Saya dapat melihat bagaimana join dan klausa WHERE dapat digunakan untuk menghasilkan informasi yang relevan dan berguna, seperti profil nasabah, aktivitas transaksi, dan status keuangan rekening. Hal ini menunjukkan pentingnya pemahaman yang baik tentang bahasa SQL dalam konteks pengelolaan data dan analisis informasi di industri perbankan.