**LAPORAN PRAKTIKUM**

**BASIS DATA RD**

**MODUL 5**

**Oleh :**

**Lois Novel E Gurning (122140098)**



**Program Studi Teknik Informatika**

**Fakultas Teknologi Industri**

**Institut Teknologi Sumatera**

**2024**

# **Daftar Isi**

[**Daftar Isi** 2](#_Toc164284529)

[**1.** **Dasar Teori** 3](#_Toc164284530)

[**2.** **Ulasan** 3](#_Toc164284531)

[**3.** **Hasil dan Jawaban** 6](#_Toc164284532)

[**4.** **Kesimpulan dan Saran** 10](#_Toc164284533)

# 

# **Dasar Teori**

Konsep utama dalam SQL yang penting adalah join, yang memungkinkan penggabungan data dari tabel-tabel yang berbeda dalam satu kueri. Inner join, sebagai contoh, menghasilkan baris-baris dengan nilai yang sama di kedua tabel yang digabungkan, berdasarkan kriteria tertentu. Ini memungkinkan pengguna untuk menampilkan informasi terkait, seperti menggabungkan data pelanggan dengan data transaksi untuk melihat rincian transaksi per pelanggan. Di sisi lain, cross join menghasilkan gabungan setiap baris dari kedua tabel, berguna ketika semua kombinasi baris diperlukan tanpa mempertimbangkan kesesuaian nilai. Natural join, dengan menggabungkan tabel berdasarkan kolom-kolom dengan nama dan tipe data yang sama, menyederhanakan proses penggabungan jika kolom-kolom tersebut ada di kedua tabel. Kemudian, outer join, seperti left outer join (LEFT JOIN) atau right outer join (RIGHT JOIN), memberikan fleksibilitas tambahan dengan menyatukan data dari dua tabel dan mempertahankan baris yang tidak memiliki nilai yang cocok dengan NULL di hasil querry.

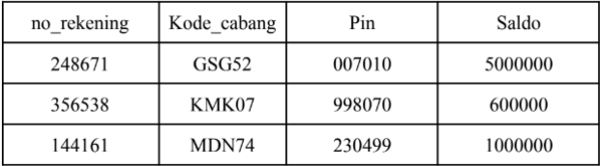
# **Ulasan**

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



Isi tabel nasabah dalam basis data bank\_itera mengikuti contoh yang ada dalam modul.

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



Input data ke tabel nasabah di basis data bank\_itera sesuai dengan contoh dalam modul.

1. Dengan menggunakan Full Join untuk tabel rekening dan transaksi tampilkan nomor rekening , saldo dan tanggal transaksi, id transaksi.

Permintaan ini menginstruksikan untuk melakukan Full Join antara tabel rekening dan transaksi, mempertahankan semua baris dari kedua tabel tanpa memperhatikan nilai-nilai yang sama.

1. Dengan menggunakan Full Join untuk tabel rekening dan transaksi tampilkan nomor rekening , saldo dan tanggal transaksi, id transaksi.

Sama dengan permintaan sebelumnya, yang mengharuskan penggunaan Full Join untuk menggabungkan data dari tabel rekening dan transaksi dalam satu kueri.

1. Dengan menggunakan salah satu dari Jenis Join, tampilkan data nasabah dan rekening ketika saldo rekening>200.000.

Instruksi ini menampilkan penggunaan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menampilkan data nasabah dan rekening yang memiliki saldo rekening di atas 200.000.

1. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nomor rekening, saldo, jumlah transaksi, tanggal transaksi ketika nomor rekening sama dengan 114115.

Disini diminta untuk menggunakan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menggabungkan data rekening dan transaksi berdasarkan nomor rekening tertentu.

1. Dengan menggunakan Left Join untuk tabel rekening dan transaksi tampilkan nomor rekening , saldo dan tanggal transaksi, id transaksi.

Meminta penggunaan Left Join untuk menggabungkan data dari tabel rekening dan transaksi, mempertahankan semua baris dari tabel rekening dan baris yang sesuai dari tabel transaksi.

1. Dengan menggunakan Right Join untuk tabel nasabah dan rekening tampilkan nomor rekening, nama nasabah , kode cabang dan saldo nasabah.

Permintaan ini menyarankan penggunaan Right Join untuk menggabungkan data dari tabel nasabah dan rekening, mempertahankan semua baris dari tabel nasabah dan baris yang cocok dari tabel rekening.

1. Dengan menggunakan salah satu dari Jenis Join, tampilkan data rekening dan transaksi ketika nomor rekening “2” dan jenis transaksi menggunakan ATM.

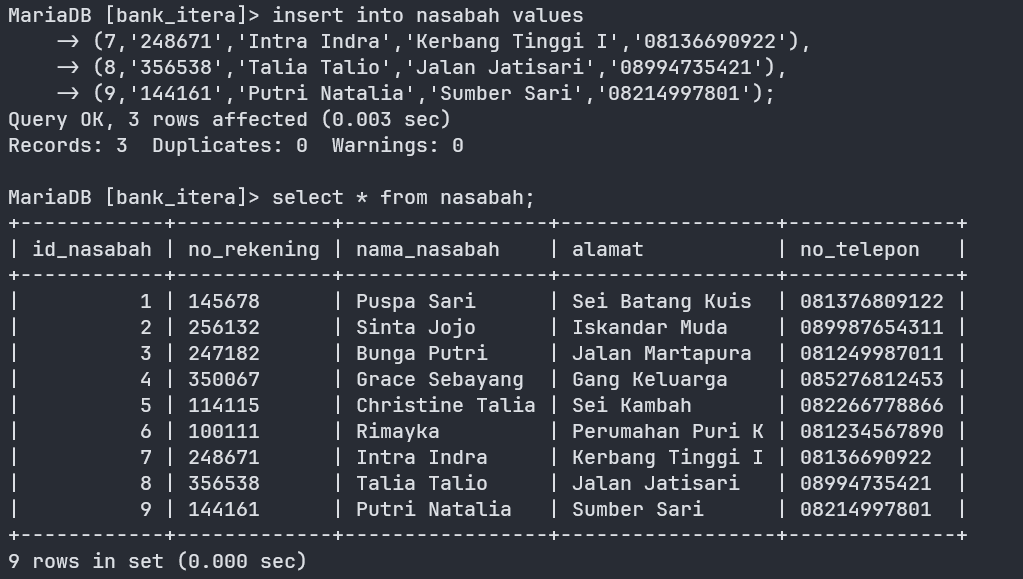
Instruksi ini meminta penggunaan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menggabungkan data rekening dan transaksi sesuai dengan kriteria tertentu.

1. Dengan menggunakan salah satu dari Jenis Join, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika nomor rekening kode cabang “KKM” dan nomor telepon“0813”.

Instruksi ini menunjukkan penggunaan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menampilkan data nasabah dan rekening yang memenuhi kriteria nomor rekening kode cabang "KKM" dan nomor telepon "0813".

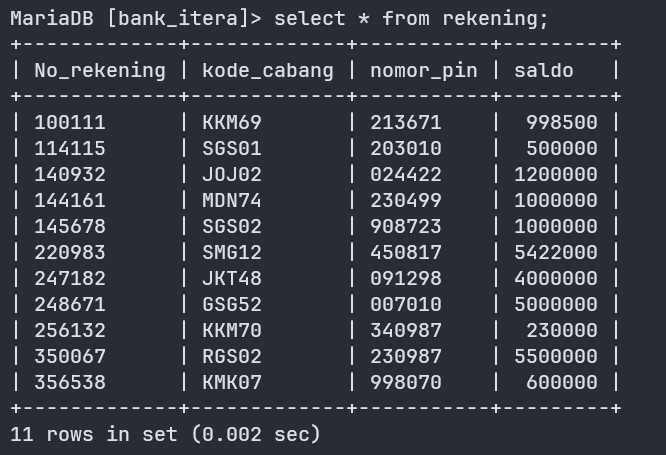
# **Hasil dan Jawaban**

1. Soal 1 Screenshoot hasil dan jawaban dari pengujian



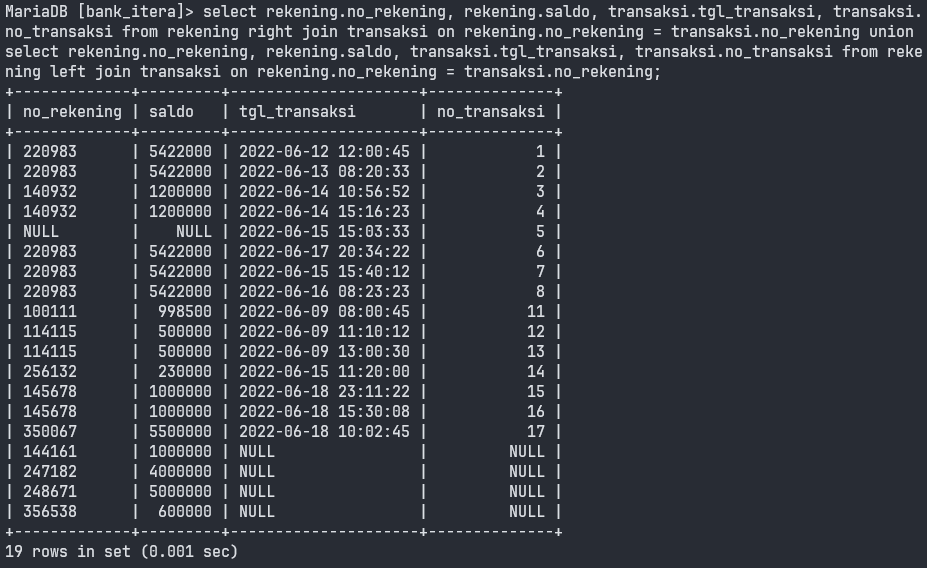
Pernyataan SQL yang ditunjukkan dalam gambar adalah contoh perintah untuk memasukkan data ke dalam tabel "nasabah", dengan mencakup beberapa kolom seperti id\_nasabah, no\_rekening, nama\_nasabah, alamat, dan no\_telepon.

1. Soal 2 Screenshoot hasil dan jawaban dari pengujian



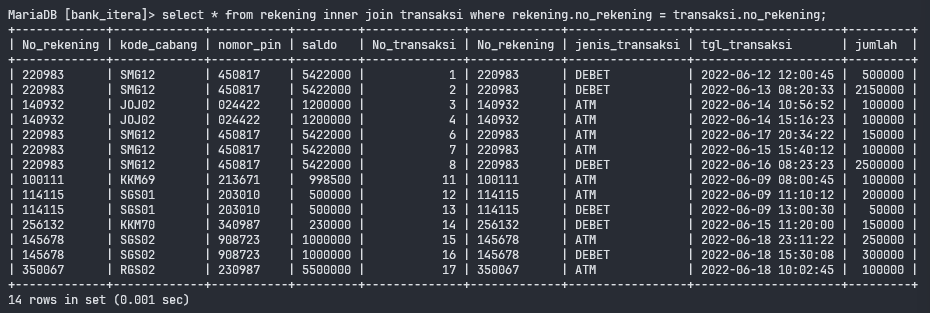
Diatas adalah gambaran tabel rekening setelah memasukan data sesuai dengan yang ada di modul

1. Soal 3 Screenshoot hasil dan jawaban dari pengujian



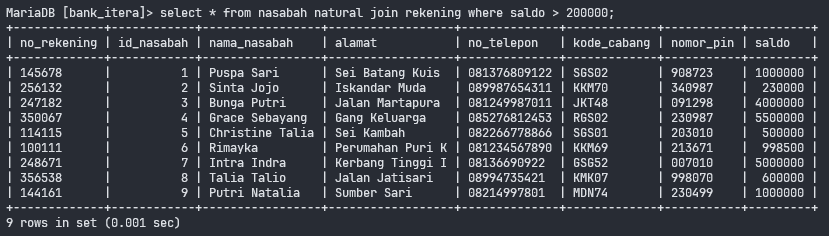
Query SQL ketiga menggunakan Right Join untuk menggabungkan informasi dari tabel rekening dan transaksi. Pendekatannya adalah mempertahankan semua data dari kedua tabel tersebut tanpa kehilangan entri apa pun. Hasil akhir dari query ini akan menampilkan nomor rekening, saldo, tanggal transaksi, dan id transaksi yang terkait.

1. Soal 4 Screenshoot hasil dan jawaban dari pengujian



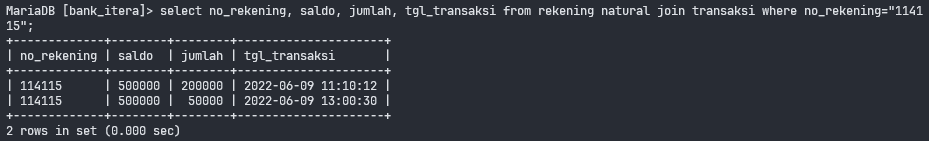
Querry SQL keempat juga menggunakan Inner Join dengan tujuan yang sama seperti nomor 3.

1. Soal 5 Screenshoot hasil dan jawaban dari pengujian



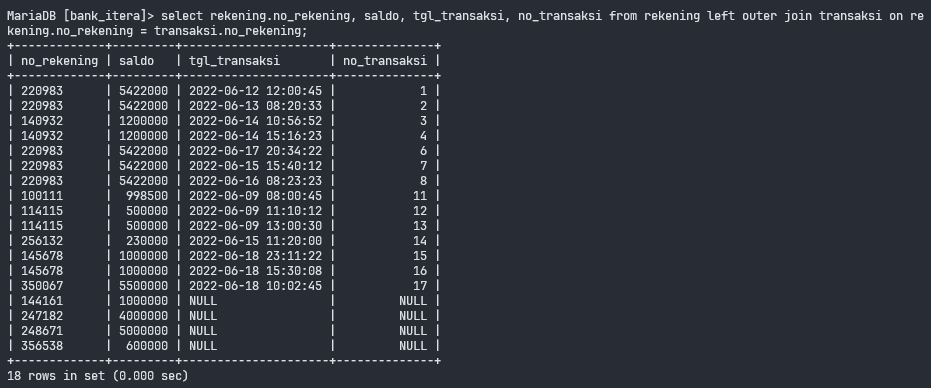
Perintah SQL kelima menggabungkan data nasabah dan rekening dengan menggunakan Natural Join, dengan syarat bahwa saldo rekening harus lebih dari 200.000, dan tujuan akhirnya adalah menampilkan informasi dari kedua tabel tersebut yang memenuhi kriteria tersebut.

1. Soal 6 Screenshoot hasil dan jawaban dari pengujian



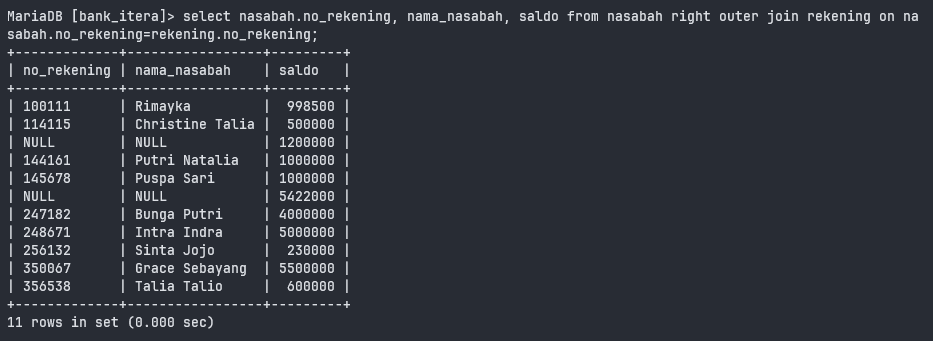
SQL statement keenam menggabungkan data antara tabel rekening dan transaksi, memanfaatkan Natural Join, dengan fokus pada nomor rekening spesifik, yaitu 114115. Hasil akhirnya adalah menampilkan informasi terkait dari kedua tabel yang berkaitan dengan nomor rekening tersebut.

1. Soal 7 Screenshoot hasil dan jawaban dari pengujian



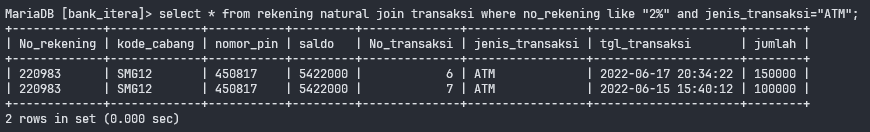
Perintah SQL ketujuh menggunakan Left Outer Join untuk menggabungkan tabel rekening dan transaksi, mempertahankan semua baris dari tabel rekening dan baris yang cocok dari tabel transaksi.

1. Soal 8 Screenshoot hasil dan jawaban dari pengujian



Perintah SQL kedelapan menggunakan Right Outer Join untuk menggabungkan tabel nasabah dan rekening, mempertahankan semua baris dari tabel nasabah dan baris yang cocok dari tabel rekening.

1. Soal 9 Screenshoot hasil dan jawaban dari pengujian



Perintah SQL kesembilan menggunakan Natural Join untuk menampilkan data rekening dan transaksi saat nomor rekening adalah "2" dan jenis transaksi adalah ATM.

1. Soal 10 Screenshoot hasil dan jawaban dari pengujian



Perintah SQL kesepuluh menggunakan Natural Join untuk menampilkan data nasabah dan rekening saat nomor rekening memiliki kode cabang "KKM" dan nomor telepon "0813".

# **Kesimpulan dan Saran**

Dari pengalaman praktis dalam menggunakan berbagai jenis JOIN di Database Bank\_Itera, dapat disimpulkan bahwa keahlian dalam memanfaatkan LEFT JOIN, RIGHT JOIN, dan FULL JOIN sangatlah penting untuk mengintegrasikan data dari tabel-tabel yang berbeda dengan ketepatan kriteria. Pemilihan jenis JOIN yang tepat memungkinkan pengguna untuk memahami interaksi data nasabah, rekening, dan transaksi secara lebih terperinci. Disarankan untuk terus mengasah kemampuan dalam menggunakan klausa WHERE dalam proses JOIN untuk menyaring data dengan presisi sesuai dengan keperluan analisis, seperti menampilkan data dengan kondisi spesifik atau melakukan pengecualian data berdasarkan kriteria yang ditetapkan. Dengan pemahaman yang mendalam tentang SQL, hal ini akan meningkatkan efisiensi dalam mengelola data dan memberikan dukungan yang lebih kuat dalam pengambilan keputusan yang lebih akurat dalam administrasi basis data.