**LAPORAN PRAKTIKUM**

**BASIS DATA RD**

**MODUL 5**

**Oleh :**

**Muhammad Fadhil Zurani (122140146)**



**Program Studi Teknik Informatika**

**Fakultas Teknologi Industri**

**Institut Teknologi Sumatera**

**2024**

# **Daftar Isi**

[**Daftar Isi** 2](#_Toc163922734)

[**1.** **Dasar Teori** 3](#_Toc163922735)

[**2.** **Ulasan** 3](#_Toc163922736)

[**3.** **Hasil dan Jawaban** 6](#_Toc163922737)

[**4.** **Kesimpulan dan Saran** 10](#_Toc163922738)

# 

# **Dasar Teori**

Join adalah konsep penting dalam SQL yang memungkinkan pengguna untuk menggabungkan data dari dua atau lebih tabel dalam satu query. Inner join adalah jenis join yang menghasilkan baris-baris dengan nilai yang cocok di kedua tabel yang di-join berdasarkan kriteria tertentu. Hal ini memungkinkan pengguna untuk mengambil informasi terkait dari tabel yang berbeda, misalnya menggabungkan data nasabah dengan data transaksi untuk melihat detail transaksi yang dilakukan oleh setiap nasabah.

Selain inner join, ada juga cross join 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. Di sisi lain, natural join adalah jenis join yang menggabungkan tabel berdasarkan kolom-kolom dengan nama yang sama dan tipe data yang cocok secara otomatis, mempermudah proses penggabungan data jika tabel memiliki kolom-kolom yang sama. Sedangkan outer join, termasuk left outer join (LEFT JOIN), right outer join (RIGHT JOIN), dan full outer join (FULL JOIN), memberikan fleksibilitas lebih besar dalam menyatukan data dari dua tabel atau lebih, dengan mempertahankan baris-baris yang tidak memiliki nilai cocok dengan NULL di hasil query.

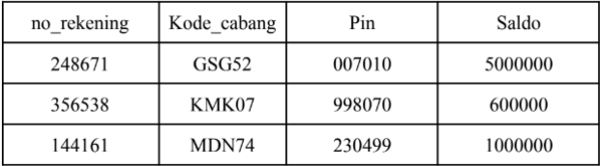
# **Ulasan**

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



Insert data pada tabel nasabah di database bank\_itera sesuai contoh yang terdapat pada modul

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



Insert data pada tabel nasabah di database bank\_itera sesuai contoh yang terdapat pada modul

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

Instruksi ini meminta untuk menggabungkan data dari tabel rekening dan transaksi dengan menggunakan Full Join, sehingga mempertahankan semua baris dari kedua tabel tanpa mempertimbangkan nilai-nilai yang cocok.

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

Ini mirip dengan instruksi sebelumnya, yang meminta penggunaan Full Join untuk menggabungkan data dari tabel rekening dan transaksi dalam satu query.

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

Instruksi ini menunjukkan penggunaan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menampilkan data nasabah dan rekening yang memiliki saldo rekening lebih dari 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.

Ini meminta penggunaan 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.

Instruksi ini meminta penggunaan Left Join untuk menggabungkan data dari tabel rekening dan transaksi, dengan mempertahankan semua baris dari tabel rekening dan baris yang cocok dari tabel transaksi.

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

Instruksi ini menyarankan penggunaan Right Join untuk menggabungkan data dari tabel nasabah dan rekening, dengan 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.

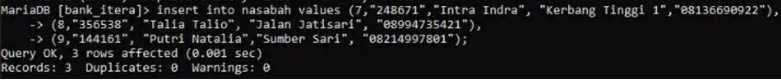
Instruksi ini meminta penggunaan salah satu Jenis Join (seperti Inner Join atau Left Join) untuk menggabungkan data rekening dan transaksi yang sesuai dengan kriteria tertentu.

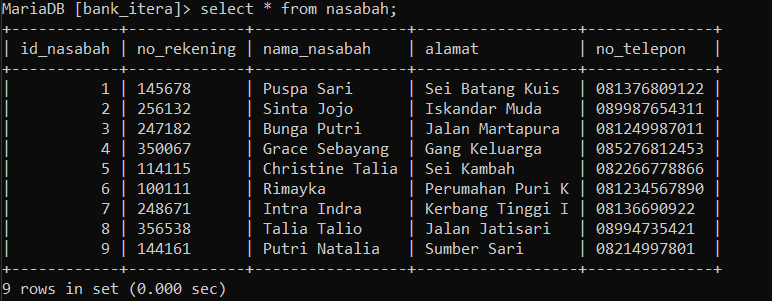
1. Dengan menggunakan salah satu dari Jenis Join, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika nomor rekening kode cabang “KKM” dan nomor telepon“0813”.

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**

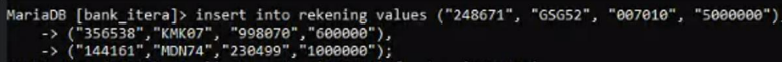
1. Soal 1 Screenshoot hasil dan jawaban dari pengujian

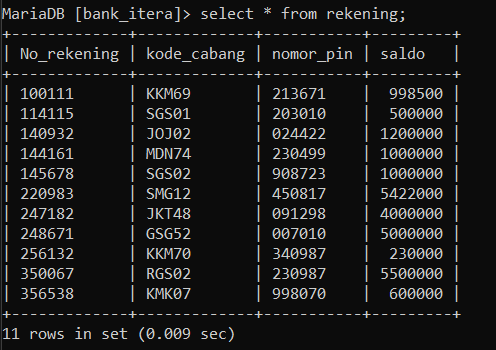




Perintah di atas pada gambar adalah contoh perintah SQL untuk menyisipkan data ke dalam tabel "nasabah" dengan beberapa kolom seperti id\_nasabah, no\_rekening, nama\_nasabah, alamat, dan no\_telepon.

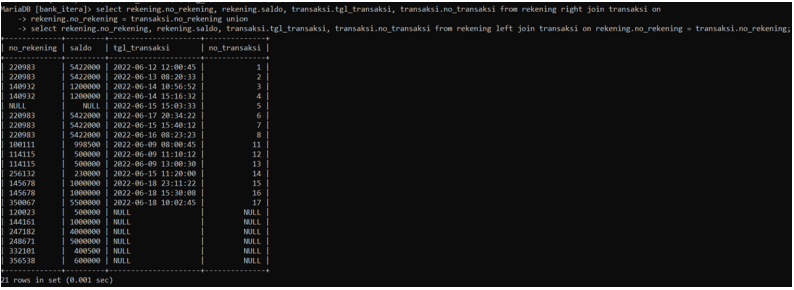
1. Soal 2 Screenshoot hasil dan jawaban dari pengujian





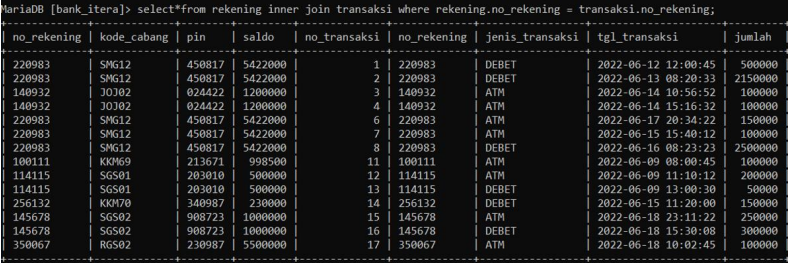
Perintah di atas adalah contoh perintah SQL untuk input data ke tabel "rekening" sesuai dengan kolom no\_rekening, kode\_bank, no\_pin, dan saldo.

1. Soal 3 Screenshoot hasil dan jawaban dari pengujian



Perintah SQL ketiga menggunakan Full Join untuk menggabungkan tabel rekening dan transaksi, mempertahankan semua baris dari kedua tabel dan menampilkan nomor rekening, saldo, tanggal transaksi, dan id transaksi.

1. Soal 4 Screenshoot hasil dan jawaban dari pengujian



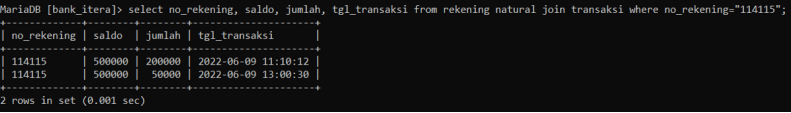
Perintah SQL keempat juga menggunakan Full Join dengan tujuan yang sama seperti nomor 3.

1. Soal 5 Screenshoot hasil dan jawaban dari pengujian



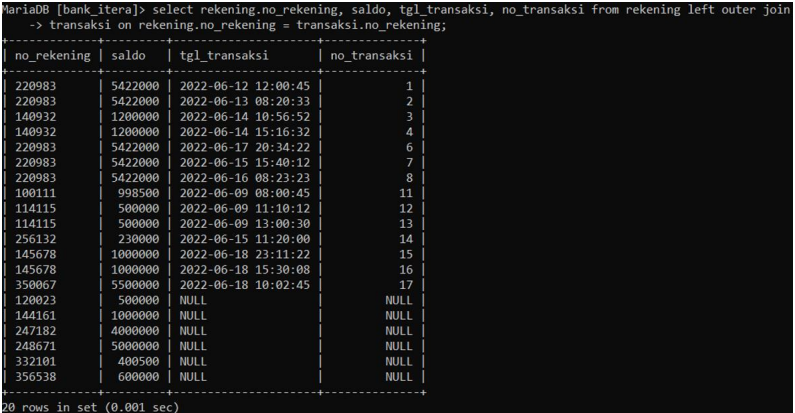
Perintah SQL kelima menggunakan Inner Join atau Left Join untuk menampilkan data nasabah dan rekening dengan saldo rekening lebih dari 200.000.

1. Soal 6 Screenshoot hasil dan jawaban dari pengujian



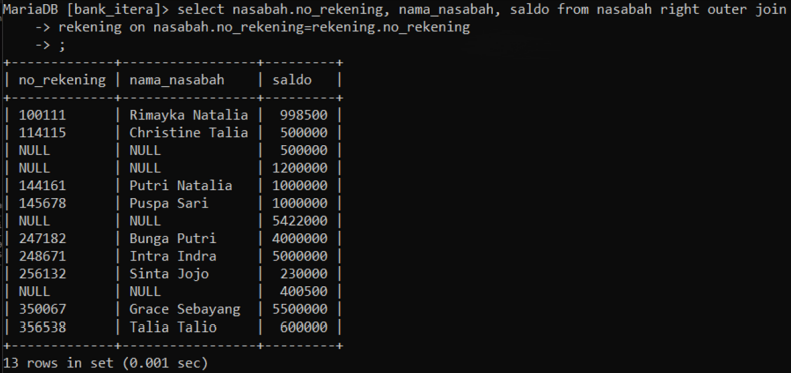
Perintah SQL keenam menggunakan Inner Join atau Left Join untuk menampilkan data rekening dan transaksi saat nomor rekening adalah 114115.

1. Soal 7 Screenshoot hasil dan jawaban dari pengujian



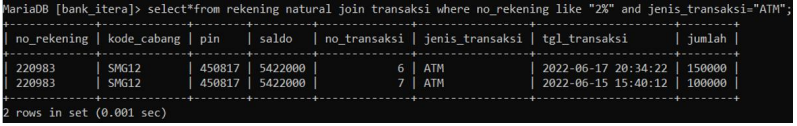
Perintah SQL ketujuh menggunakan Left Join untuk menggabungkan tabel rekening dan transaksi, mempertahankan semua baris dari tabel rekening dan baris yang cocok dari tabel transaksi.

1. Soal 8 Screenshoot hasil dan jawaban dari pengujian



Perintah SQL kedelapan menggunakan Right 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 Inner Join atau Left 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 Inner Join atau Left Join untuk menampilkan data nasabah dan rekening saat nomor rekening memiliki kode cabang "KKM" dan nomor telepon "0813".

# **Kesimpulan dan Saran**

Dari hasil praktik penggunaan berbagai jenis JOIN di Database Bank\_Itera, dapat disimpulkan bahwa keahlian dalam menggunakan LEFT JOIN, RIGHT JOIN, dan FULL JOIN sangat penting untuk menggabungkan data dari tabel yang berbeda dengan kriteria yang sesuai. Penggunaan jenis JOIN yang tepat memungkinkan pengguna untuk melihat hubungan antara data nasabah, rekening, dan transaksi secara lebih jelas. Disarankan untuk terus berlatih menggunakan klausa WHERE dalam JOIN untuk melakukan filter data dengan tepat sesuai dengan kebutuhan analisis, seperti menampilkan data dengan kondisi tertentu atau melakukan seleksi data berdasarkan kriteria yang diberikan. Dengan pemahaman yang baik tentang SQL, dapat meningkatkan efisiensi dalam pengelolaan data dan mendukung pengambilan keputusan yang lebih akurat dalam pengelolaan database.