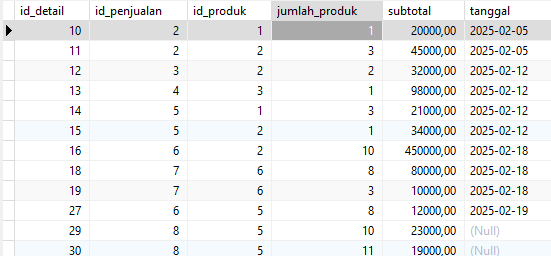
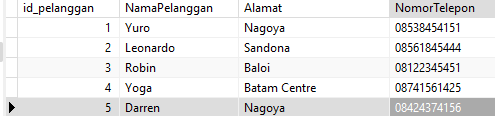
Langkah awal adalah membuat tabel detailpenjualan, pelanggan, penjualan, dan produk

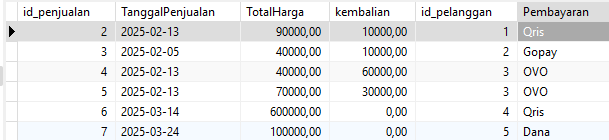
1. Detailpenjualan



1. Pelanggan



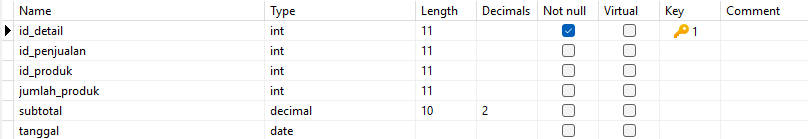
1. Penjualan



1. Produk

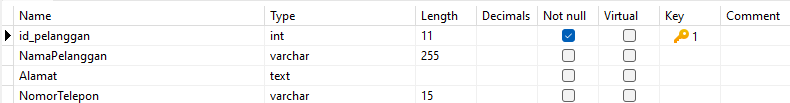


Design Table (Detailpenjualan)

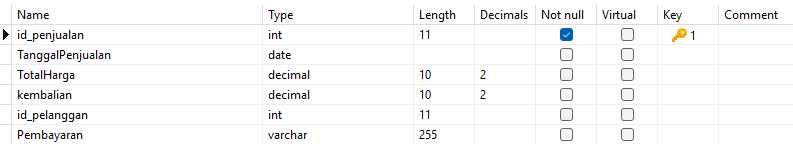


Setiap nama tabel yang menggunakan id, seperti contoh id\_detail, harus menggunakan INT (angka) dan menggunakan Auto\_increment supaya jika ingin menambahkan atau mengisi maka akan terisi id sesuai urutan dari angka 1 (untuk table lain juga seperti itu).

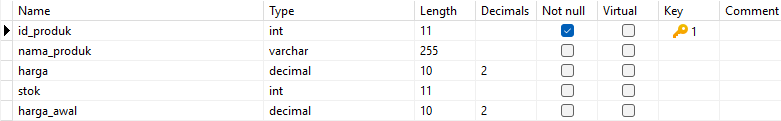
Design Table (Pelanggan)



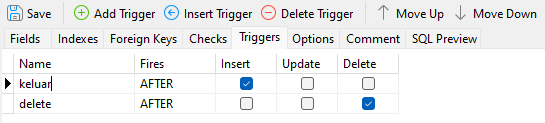
Design Table (Penjualan)



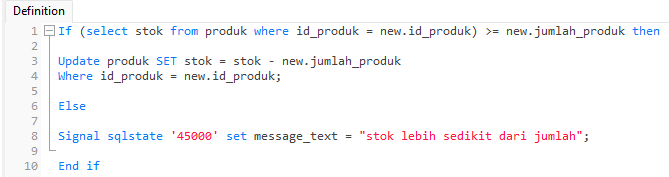
Design Table (Produk)



1. Membuat Trigger untuk Insert dan Delete



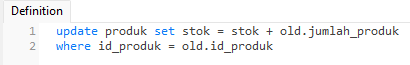
Pertama, setelah membuat Table dan Design Table langkah selanjutnya untuk membuat Trigger Insert adalah dengan cara “Add Trigger” kemudian di Fires pilih opsi “After” kemudian centang bagian “Insert”. Berikut adalah query untuk membuat Trigger Insert



Penjelasan: contoh data baru itu masuk atau telah dipebarui melalui sebuah insert dan update di tabel transaksi, jadi trigger ini akan bisa di check atau akan memeriksa apakah stok barang yang ada pada tabel produk mencukupi untuk dikurangi. Pemeriksaan dilakukan dengan membandingkan nilai dari stok pada tabel produk dengan new.jumlah\_produk (jumlah barang yang akan dikurangi)

Kemudian jika stok cukup (stok lebih besar sama dengan dari jumlah produk yang diminta), maka sistem tersebut akan mengurangi stok di tabel produk sesuai jumlah yang diminta.

Kemudian kalau stok itu tidak cukup (stok lebih sedikit dari jumlah produk yang diminta), sistem akan membatalkan proses dan dapat menampilkan pesan error: “stok lebih sedikit dari jumlah”.



Penjelasan: query ini untuk menambahkan kembali stok barang pada dalam tabel produk saat terjadi penghapusan atau sebiah pembaruan terhadap transaksi. Fungsi nya adalah “mengembalikan stok”, proses nya dengan dari perintah update atau memperbarui sebuah table produk kemudian, pada bagian set stok = stok + old.jumlah\_produk akan menambahkan lagi stok yang ada pada dengan jumlah produk dari data yang lama (old.jumlah\_produk). Dari kata kunci yang paling penting nya adalah “old” yang digunakan untuk data sebelum perubahan seperti contoh biasanya pada trigger after delete atau before update.

Bagian where id\_produk = old.id\_produk untuk memastikan bahwa hanya produk dengan id\_produk yang sesuai di perbarui.

1. Bagaimana cara melihat stok didalam produk yang stok nya itu 0?



Penjelasan: pertama pada table produk kita mengambil atayu select, (\*) bintang yang artinya semuanya from (dari) produk (nama tabel) where stok (stok itu yang ada pada tabel produk) = 0 (0 itu dari soal yang kita akan cari stok didalam produk yang stok nya itu 0).



(Untuk melihat tabel nya, bisa lihat di paling atas pada bagian design tabel dan tabel tabel di pengelolaan basis data).

1. Produk apa yang paling banyak menghasilkan keuntungan



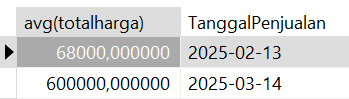
Penjelasan: Query ini digunakan untuk **menampilkan total penjualan berdasarkan metode pembayaran** pada tanggal **13 Februari 2025**. Perintah ini mengambil data dari tabel penjualan dengan cara menghitung jumlah total harga dari semua transaksi pada tanggal tersebut. Data kemudian dikelompokkan berdasarkan metode pembayaran seperti Gopay, Dana, dan Ovo.



1. berapa rata2 penjualan dalam 1 bulan

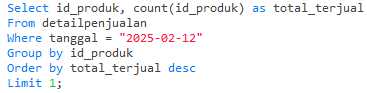


Penjelasan: Query select AVG(totalharga), tanggalpenjualan from penjualan group by month (TanggalPenjualan); digunakan untuk **menghitung rata-rata penjualan dalam satu bulan**. Query ini mengambil nilai rata-rata dari kolom totalharga dan mengelompokkannya berdasarkan bulan dari TanggalPenjualan.



(tabel untuk penjualan bisa dilihat di atas)

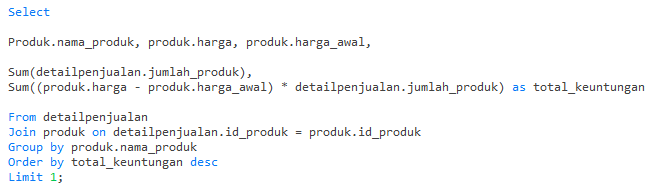
1. item yang paling banyak terjual



Penjelasan: select itu mengambil dari id\_produk, count berfungsi untuk menghitung dalam id\_produk as dari total\_terjual from (dari) tabel detailpenjualan where di bagian tanggal dengan tanggal nya adalah 2025-02-12, group by untuk mengelompokan id\_produk, fungsi order by adalah untuk mengurutkan total\_terjual, kalau desc limit 1; itu berfungsi untuk mengambil satu data teratas dari hasil query yang sudah diurutkan secara menurun dengan menggunakan order by.



1. Untuk mencari bagian yang menguntungkan



Penjelasan: Query ini digunakan untuk mencari produk yang menghasilkan keuntungan paling besar. Prosesnya dimulai dengan mengambil data dari dua tabel, yaitu tabel detailpenjualan dan produk, yang digabungkan melalui kolom id\_produk. Data yang ditampilkan meliputi nama produk, harga jual, harga awal, jumlah total produk yang terjual, dan total keuntungan dari setiap produk.

Total jumlah produk terjual dihitung dengan fungsi sum(detailpenjualan.jumlah\_produk), sedangkan total keuntungan dihitung dari selisih antara harga jual dan harga awal, kemudian dikalikan dengan jumlah produk yang terjual. Nilai ini diberi alias sebagai total\_keuntungan.

Hasil data dikelompokkan berdasarkan nama produk menggunakan group by produk.nama\_produk, kemudian diurutkan berdasarkan total\_keuntungan secara menurun (order by total\_keuntungan desc) untuk menampilkan produk dengan keuntungan terbesar di urutan teratas. Dengan penggunaan limit 1, query ini hanya menampilkan satu baris data yang merupakan produk dengan total keuntungan tertinggi.

