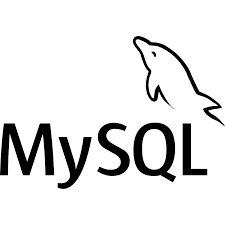
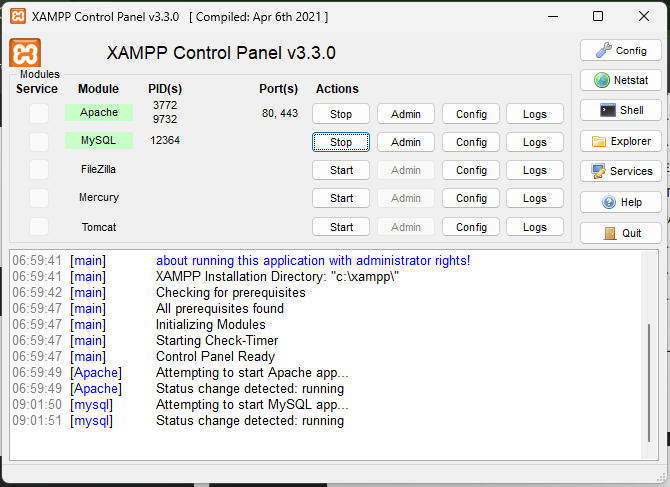
# TUGAS

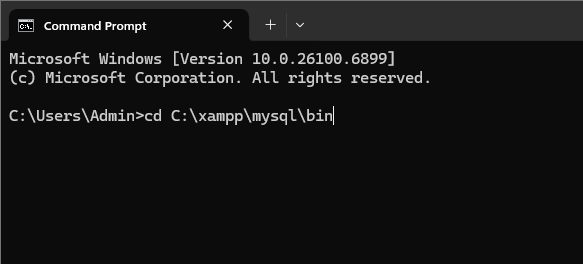


## **By : Andaru Ilham Amerta**

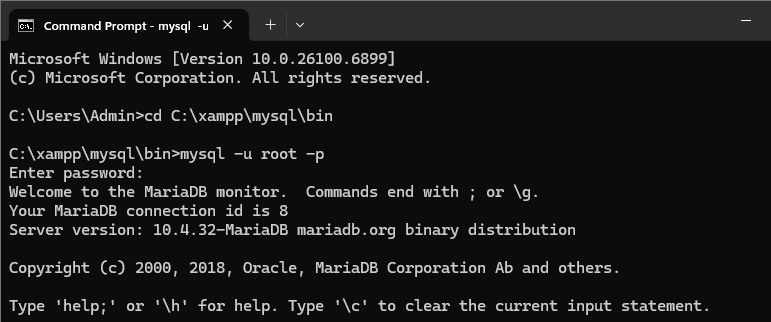
# Nyalakan MySql & Apache



Setelah terbuka jalankan MySQL dengan mengetik

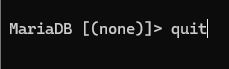


**Karena password default atau bawaan XAMPP nya kosong, langsung ketik enter saja. Jika MySQL berjalan maka akan tampil**



# KELUAR DARI MYSQL

Gunakan command :



# MEMERIKSA MYSQL JALAN ATAU TIDAK

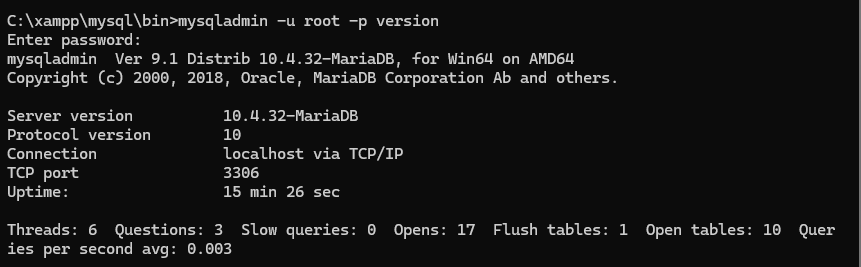
Untuk mengecek MySQL berjalan atau tidak, pastikan keluar dari MySQL dulu kemudian gunakan perintah dibawah. Tekan enter jika passwordnya kosong



# CEK VERSI MYSQL Kita bisa mengecek versi MySQL

yang digunakan menggunakan perintah; Pastikan sudah keluar dari MySQL sebelum mengecek versinya. Gunakan command

“ mysqladmin -u root -p version ”



# MENAMPILKAN HELP MYSQL

Sangat sulit mengingat semua perintah MySQL, karena itu dibutuhkan bantuan agar kita bisa mengetahui perintah MySQL yang akan digunakan. Untuk melihat bantuan MySQL, anda harus masuk dulu ke MySQL dengan perintah



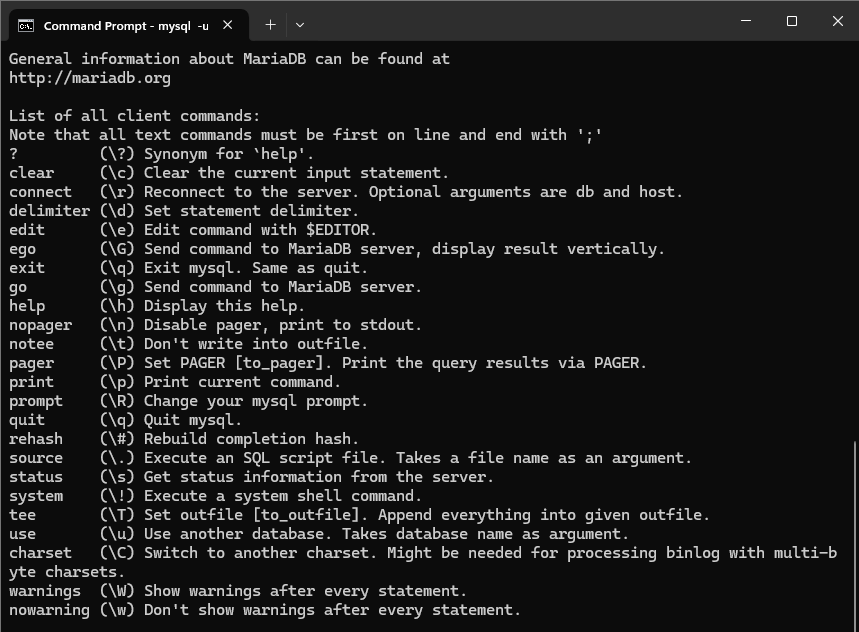
Tanda jika anda sudah ada dalam MySQL adalah



Untuk menampilkan HELP bisa menggunakan perintah

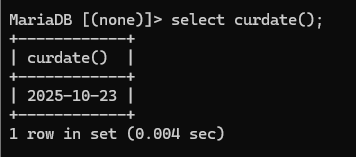
Atau gunakan 

Dan akan tampil seperti ini :



# MENAMPILKAN TANGGAL

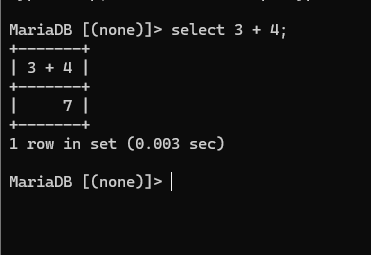
Gunakan perintah berikut untuk menampilkan tanggal. Pastikan akhir perintah di beri tanda TITIK KOMA (;) kemudian tekan enter.



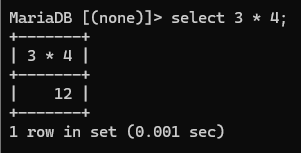
# MENGGUNAKAN KALKULATOR

Untuk menggunakan kalkulator di MySQL gunakan perintah sebagai berikut

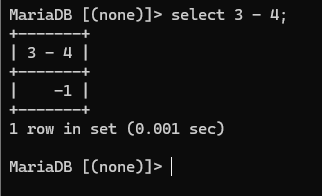
1. Penjumlahan



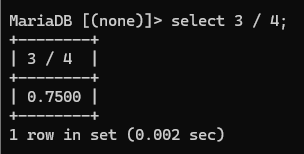
1. Perkalian



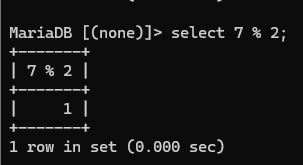
1. Pengurangan



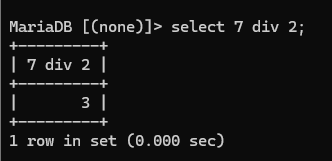
1. Pembagian



1. Modulus



1. Hasil pembagian integer (utuh)



**DDL**

# Data Definition Language

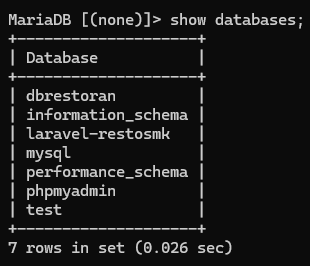
# Pembuatan DB, Tabel, & Index

# MENAMPILKAN DATABASE

Untuk menampilkan semua database yang ada di MySQL komputer anda. Gunakan perintah



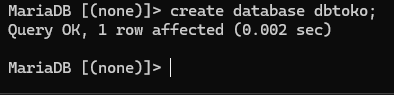
Dan hasilnya adalah :



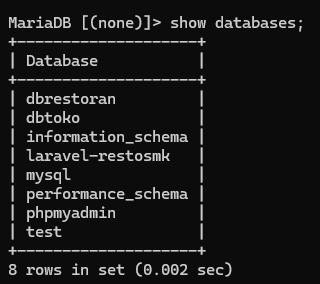
# MEMBUAT DATABASE

Pada latihan ini, akan membuat database dengan nama dbtoko. Gunakan perintah berikut





Periksa database yang sudah dibuat dengan menggunakan perintah

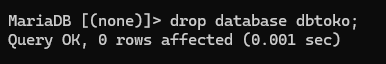


Database yang bernama “ dbtoko “

# MENGHAPUS DATABASE

Jika ada database yang tidak digunakan bisa dihapus dengan perintah



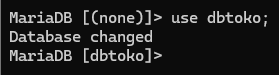


# MENGAKTIFKAN DATABASE

Jika database sebelumnya dihapus buat lagi dengan nama [dbtoko], gunakan perintah diatas. Sebelum membuat tabel anda harus memilih atau mengaktifkan database yang akan digunakan sebagai tempat penyimpanan tabel. Untuk memilih atau mengaktifkan gunakan perintah sebagai berikut



Jadi kita akan menggunakan dbtoko dengan command :

Dan hasilnya seperti itu

Perhatikan terjadi perubahan dari [none] menjadi [dbtoko] ini menunjukan bahwa database yang aktif yang akan digunakan adalah [dbtoko]

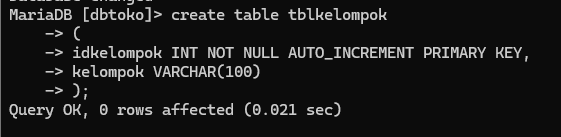
# MEMBUAT TABEL

tabel yang pertama dibuat adalah tabel tblkelompok, dan cara membuat sebuah table adalah seperti berikut



# MEMBUAT TABEL tblkelompok

tblkelompok Pastikan sebelum membuat tabel database sudah dipilih dengan benar gunakan perintah pada penjelasan sebelumnya. Jika tipe yang digunakan adalah “text” maka pada MySQL menggunakan VARCHAR

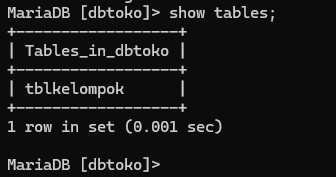


# MENAMPILKAN TABEL

Untuk menampilkan tabel yang sudah dibuat, gunakan perintah berikut



Dan hasilnya seperti berikut

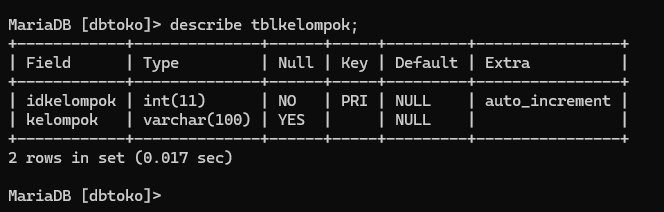


# MENAMPILKAN STRUKTUR TABEL

Untuk melihat struktur tabel gunakan perintah sebagai berikut

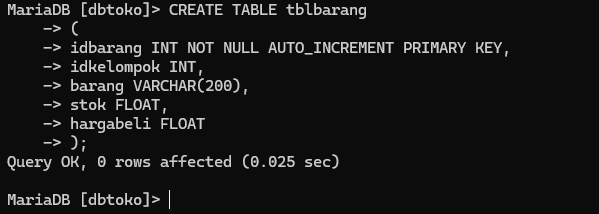


Dan hasilnya seperti berikut



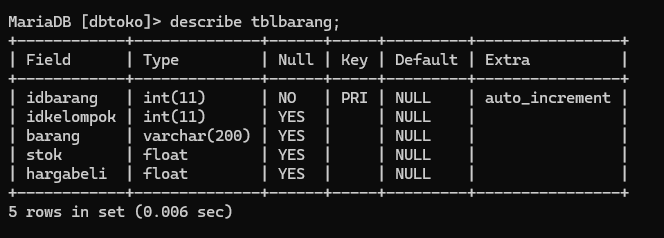
# Membuat Tabel tblbarang

Untuk membuat tabel tblbarang gunakan perintah sebagai berikut



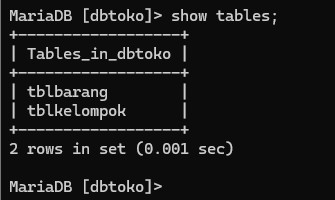
Karena pada MySQL tidak ada tipe data “ Currency “ maka kita bisa menggunakan tipe data “FLOAT”

Periksa apakah struktur yang dibuat sudah benar



# Menampilkan Tabel

Tabel yang sudah dibuat pada database bisa dilihat dengan menggunakan perintah berikut

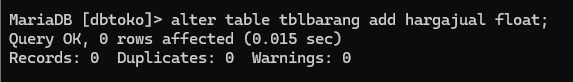


# MENAMBAH KOLOM

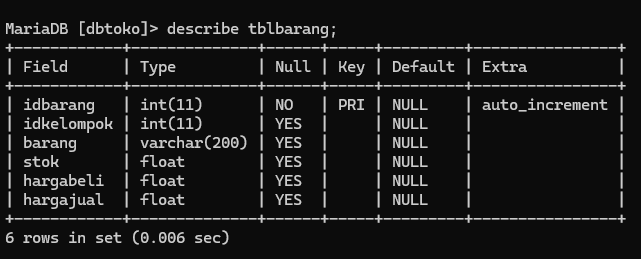
Jika dilihat pada desain terdapat kekurangan kolom pada tabel yang dibuat, kita bisa menambahkan kolom dengan perintah sebagai berikut



Kita akan tambahkan “hargajual” pada tblbarang



Setelah penambahan kolom periksa “tblbarang” dengan perintah sebagai berikut

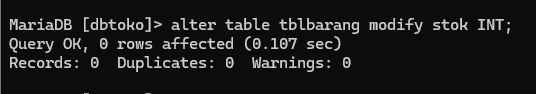


# MERUBAH TIPE DATA

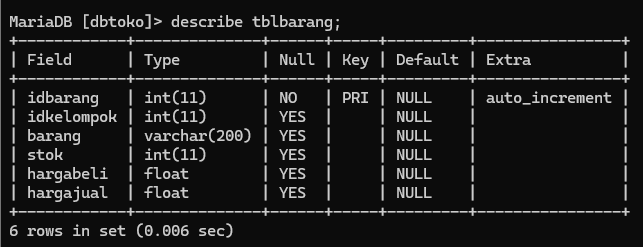
Pada tblbarang terdapat tipe data FLOAT pada kolom stok, kolom stok tipe data akan diganti menjadi INT. Gunakan perintah sebagai berikut



Dan kita kan ubah/modifikasi “stok” yang ada di dalam “tblbarang”



Periksa kembali tipe data yang digunakan menggunakan perintah berikut



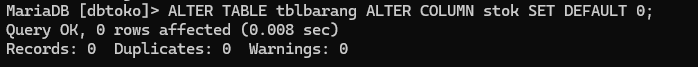
# MEMBERI NILAI DEFAULT PADA KOLOM

Pada kolom [stok] tblbarang akan diberi nilai DEFAULT [0] atau nilai awal [0]. Gunakan perintah sebagai berikut

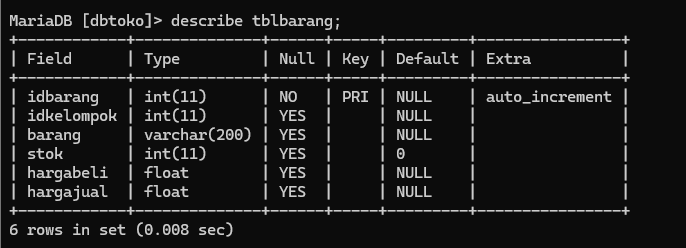


Isi nilai DEFAULT, jika VARCHAR maka beri tanda petik

Dan kita beri nilai pada DEFAULT yang ada pada column “stok” yang ada di dalam “tblbarang”



Periksa dengan menggunakan perintah berikut



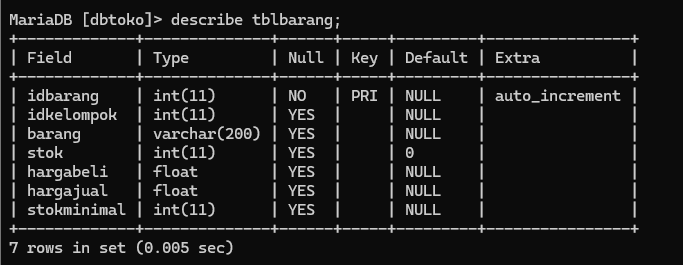
Dan seperti yang dilihat “Default” pada “stok” bukan lagi “NULL” tapi berubah menjadi 0

# MENGHAPUS KOLOM

Pada latihan untuk menghapus kolom, tambahkan kolom baru terlebih dahulu dengan menggunakan perintah sebagai berikut



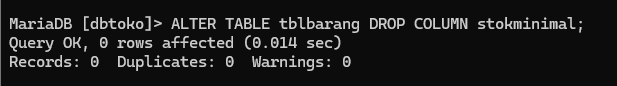
Periksa struktur tabel yang sudah dibuat



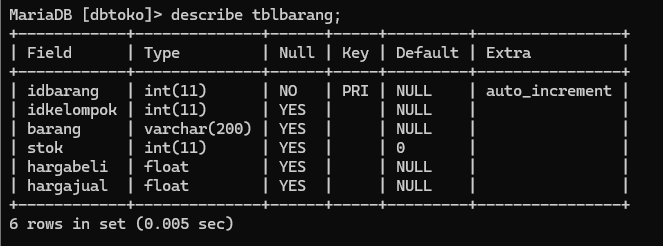
Pada struktur diatas kolom stokminimal akan dihapus. Untuk menghapus gunakan perintah berikut



Dan kita akan coba untuk men-DROP kolom “stokminimal”



Periksa struktur tabel dengan perintah berikut



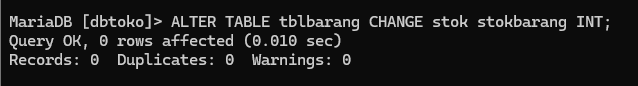
Dan yup, column “stokminimal” berhasil di hapus

# MERUBAH NAMA KOLOM

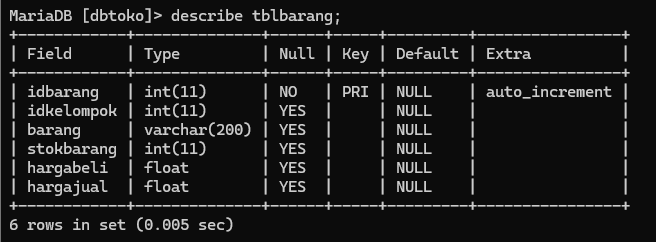
Untuk merubah nama kolom jika terdapat kesalahan bisa menggunakan perintah sebagai berikut



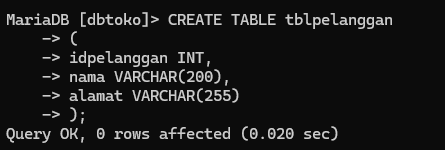
Kita coba akan ubah column “stok” menjadi “stokbarang”



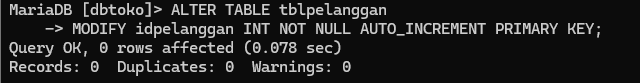
Periksa struktur tabelnya



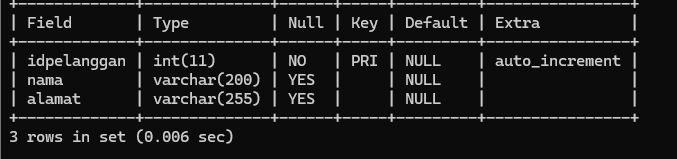
# MEMBUAT TABEL tblpelanggan



Pada pembuatan tabel diatas belum ada PRIMARY KEY. Untuk menambahkan PRIMARY KEY gunakan perintah ubah tipe data.



Periksa struktur tabel dengan perintah berikut



Dan akhirnya pada “idpelanggan” memiliki PRIMARY KEY

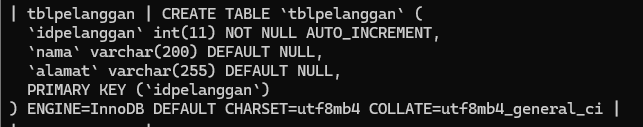
# MENAMPILKAN ENGINE YANG DIGUNAKAN

Pada pembuatan database terdapat 2 engine atau mesin yang digunakan yaitu MYISAMdan InnoDB. Untuk proses pembuatan database relasional harus menggunakan InnoDB. MYISAM tidak bisa menerima penggunaan FOREIGN KEY.

Gunakan perintah berikut



Kita akan coba untuk menampilkan engine yang digunakan oleh “tblpelanggan”

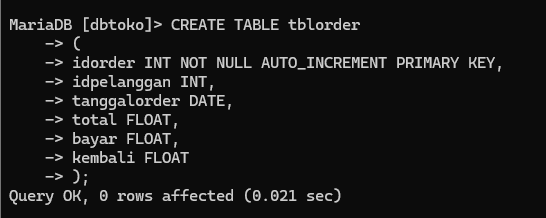


Dan engine yang digunakan adalah :

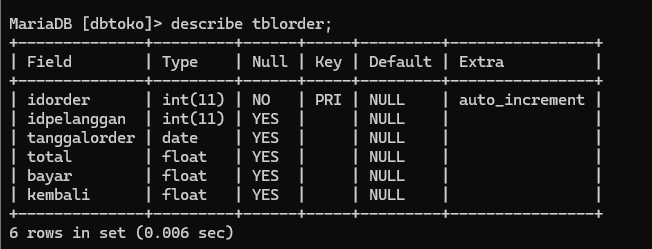
**ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci**

# MEMBUAT TABEL tblorder

Perintah untuk membuat tblorder



Struktur tabel

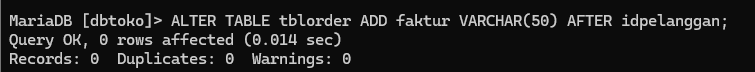


Perhatikan pada desain pembuatan tabel diatas, kita kekurangan satu kolom yaitu kolom [faktur] yang letaknya dibawah kolom [idpelanggan].

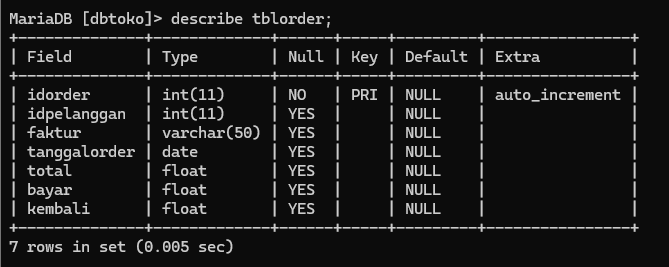
# MENAMBAH KOLOM SETELAH KOLOM



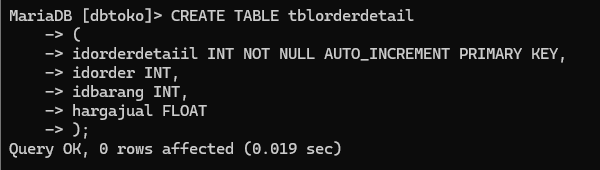
Kita akan buat kolom [faktur] yang letaknya dibawah kolom [idpelanggan]



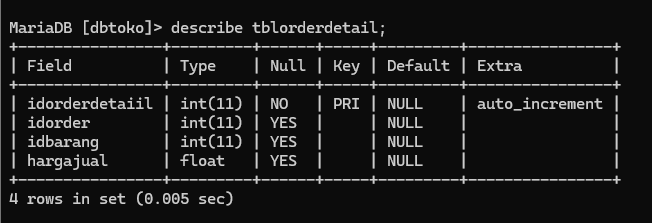
Hasilnya adalah



# MEMBUAT TABEL tblorderdetail



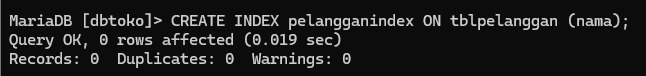
Struktur tabel yang sudah dibuat



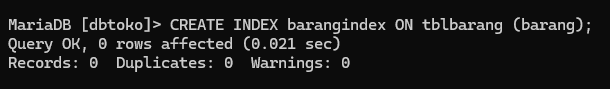
# MEMBUAT INDEX

Index digunakan untuk mempercepat proses pencarian data terutama pada data yang jumlahnya diatas ribuan. Pada tabel diatas yang biasa digunakan untuk pencarian data adalah nama barang dan nama pelanggan.

  
Buat index untuk tabel tblpelanggan dengan perintah sebagai berikut



Buat index untuk tabel tblbarang dengan perintah sebagai berikut



# MENAMPILKAN INDEX

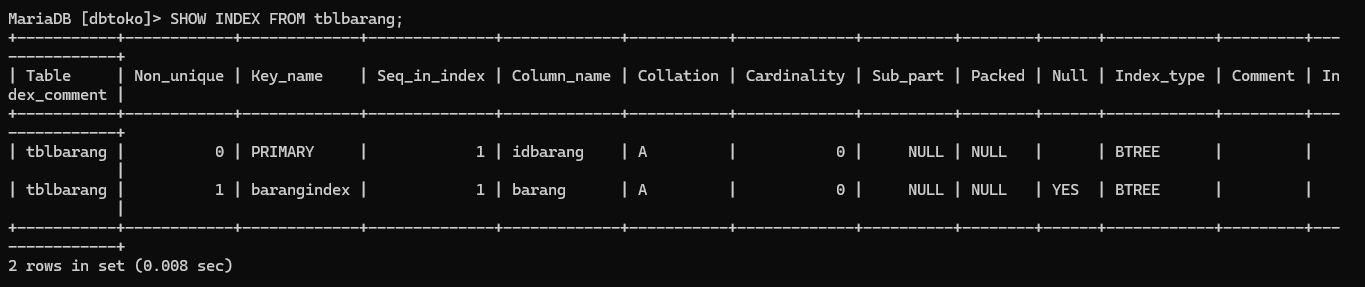
Index yang sudah dibuat bisa ditampilkan dengan perintah



Kita akan coba melihat index dari tblbarang



Dan hasilnya adalah



# MENGHAPUS INDEX

Jika INDEX tidak digunakan bisa dihapus dengan perintah



Kita akan coba men\_DROP “pelangganindex” yang ada di “tblpelanggan”



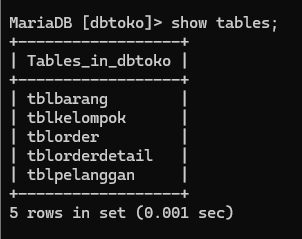
# DML

# Data Manipulation Language

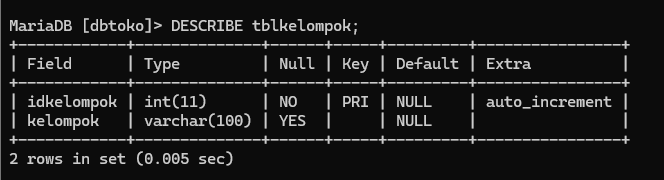
Insert, Delete, Update, Select, View, Trigger, Procedure, & Function

# INSERT SEMUA KOLOM

Insert semua kolom digunakan untuk memasukan data pada semua kolom tabel. Sebelum melakukan INSERT data periksa dulu tabel yang akan digunakan.



Sebagai latihan akan menggunakan tblkelompok, lihat dulu kolom tblkelompok



Pada tblkelompok terdapat 2 kolom yang akan di INSERT datanya, yaitu [idkelompok], [kelompok].



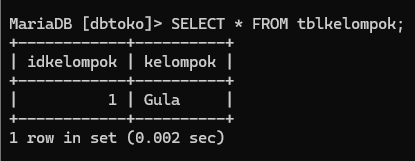
Kita akan coba insert di tblkelompok



Untuk memeriksa apakah data yang di INSERT kan berhasil gunakan perintah



Kita akan lihat data yang sudah kita insert di tblkelompok



# INSERT SEBAGIAN KOLOM INSERT

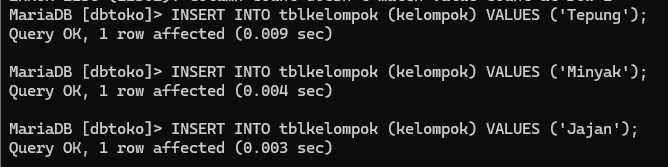
sebagian kolom digunakan untuk melakukan INSERT HANYA PADA kolom yang disebutkan.



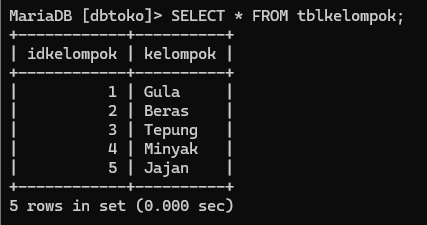
Kita akan mencoba di tblkelompok



Kita akan latihan, dan akan melakukan INSERT data



sehingga diperoleh data sebagai berikut



# DELETE SEBAGIAN RECORD (BARIS DATA)

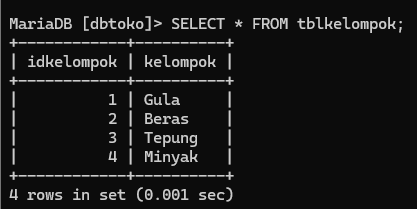
Untuk menghapus sebagian baris gunakan perintah



Kita akan coba untuk delete “Jajan”



Sekarang akan kita cek setelah kita hapus

 Dan yupp, Jajan sudah terhapus.

# DELETE SEMUA RECORD

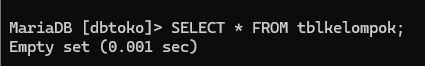
Untuk menghapus semua baris data bisa dilakukan dengan perintah



Kita akan coba record tblkelompok



Karena data sudah dihapus maka yang tampil adalah



# UPDATE SEBAGIAN RECORD

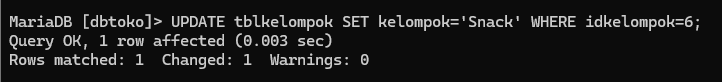
Karena semua data sudah dihapus semua pada materi delete, isi dulu data pada tblkelompok sebagai latihan dengan data sebagai berikut



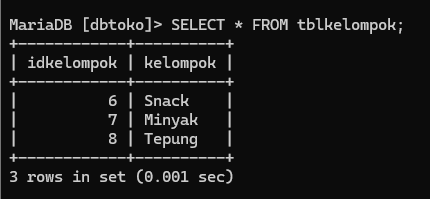
Pada tabel diatas UBAH data ‘Jajan’ menjadi ‘Snack’ menggunakan perintah sebagai berikut



Kita akan coba UPDATE “Jajan” menjadi “Snack”



Kita akan periksa data yang sudah kita ubah

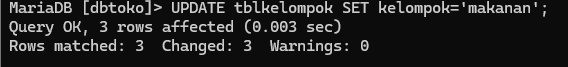


# UPDATE SEMUA RECORD

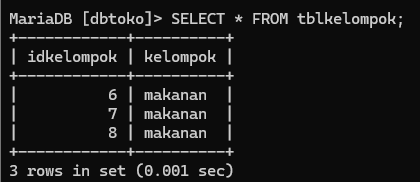
Untuk mengubah semua data bisa dilakukan dengan perintah sebagai berikut



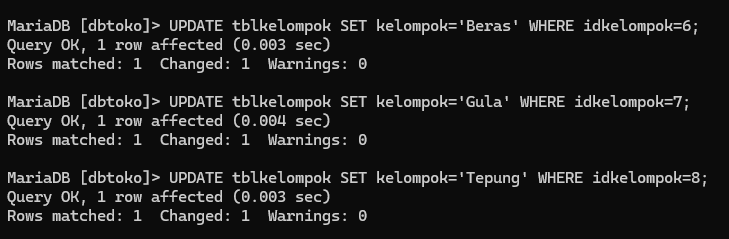
Kita coba UPDATE yang ada di kelompok menjadi “Makanan”



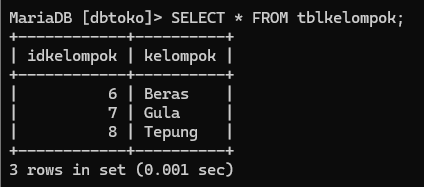
Kita coba periksa setelah kita UPDATE



Kita akan kembalikan seperti semula dengan menggunakan update



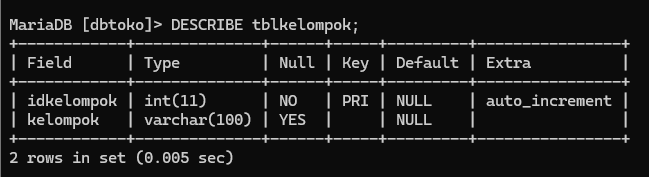
Kita akan cek setelah kita UPDATE



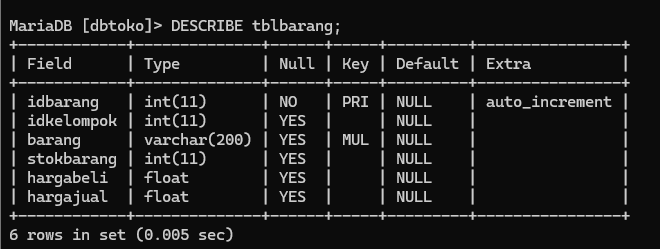
# TABEL MASTER DAN TABEL TRANSAKSI (DETAIL)

Lihat struktur tblkelompok dan tblbarang

tblkelompok

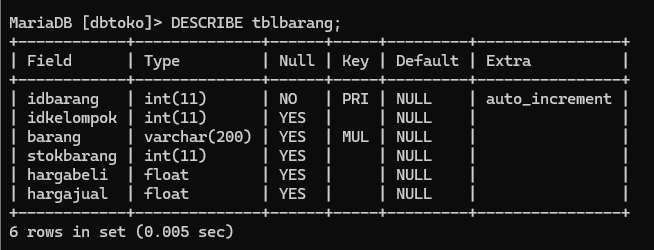


tblbarang

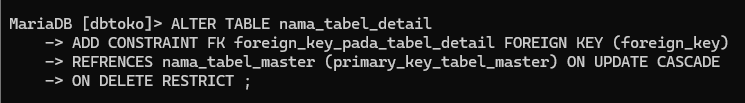


# MEMBUAT RELASI ANTAR TABEL

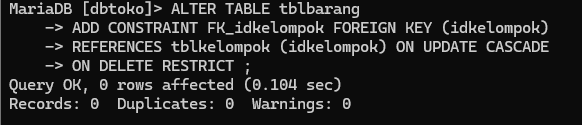
Lihat dulu struktur tabel yang akan dibat relasinya. Pada tblbarang terlihat struktur seperti pada gambar



Perintah untuk membuat relasi adalah sebagai berikut



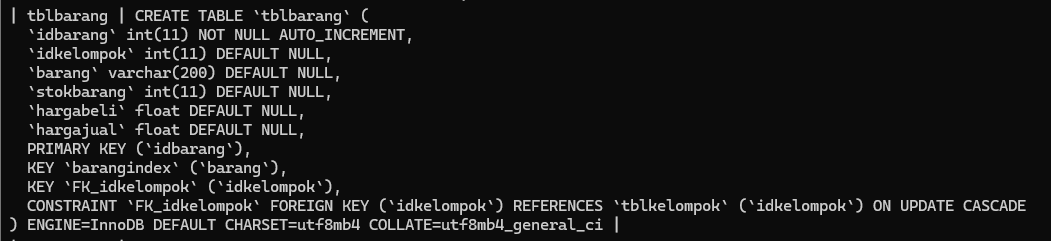
Kita akan coba pada tblbarang



Untuk melihat hasil pembuatan relasi gunakan perintah berikut



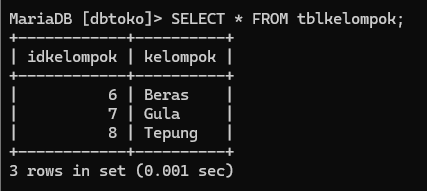
Kita akan lihat tblbarang



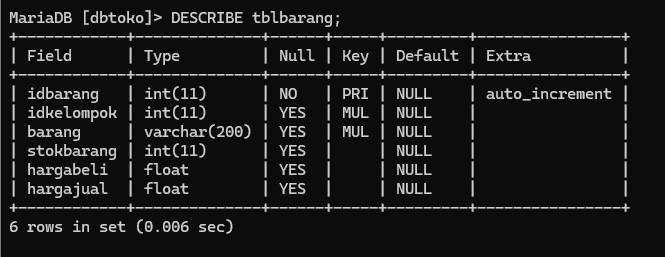
Dan diatas adalah hasilnya

# MENGUJI HASIL PEMBUATAN RELASI

Untuk menguji hasil pembuatan relasi lakukan INSERT data pada tabel master terlebih dahulu baru kemudian ke tabel transaksi.



Tampilkan struktur tabel untuk memudahkan proses insert data

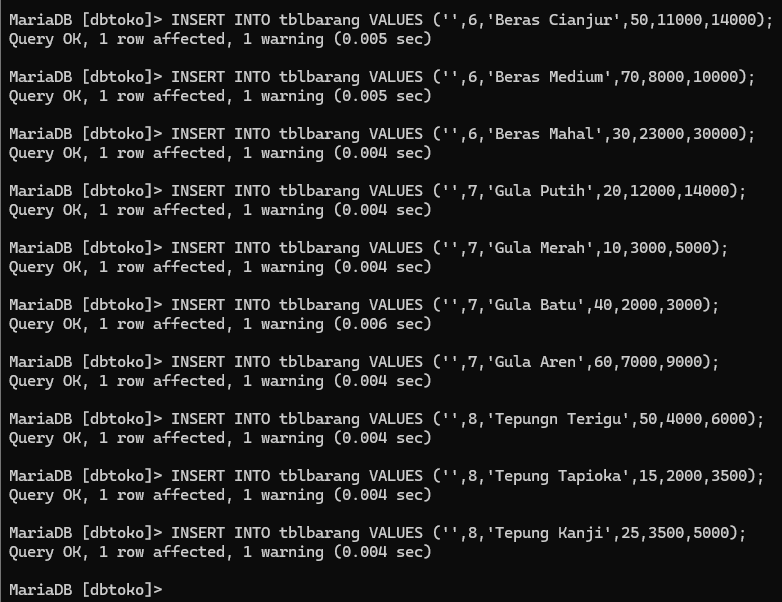


Isi tabel transaksi sesuai dengan idkelompok yang akan digunakan

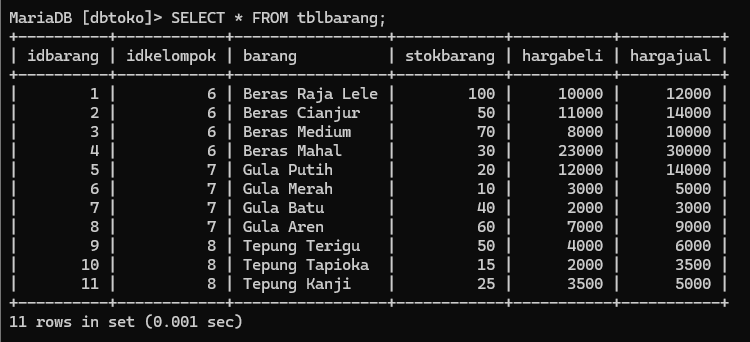
Karena idbarang AUTO\_INCREMENT maka kolom bisa diganti dengan ‘’ (petik kosong)



Kita lakukan insert beberapa jenis beras, gula, dan tepung

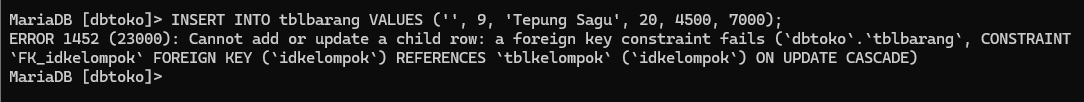


Dan kita akan lihat hasilnya



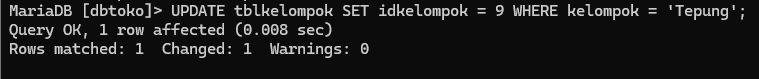
# PENGUJIAN INSERT

Insert pada tabel [tblbarang] HANYA BISA dilakukan menggunakan data yang tersedia pada tabel master. [Idkelompok] pada tabel master yang tersedia hanya [6,7,8].. Perhatikan hasil pengujian berikut; Pada pengujian INSERT idkelompok yang digunakan adalah 9 yang TIDAK TERSEDIA pada tabel master [tblkelompok].

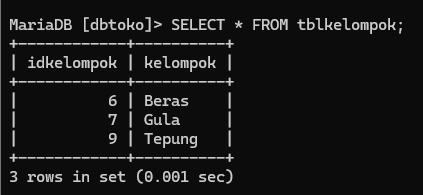


# PENGUJIAN UPDATE

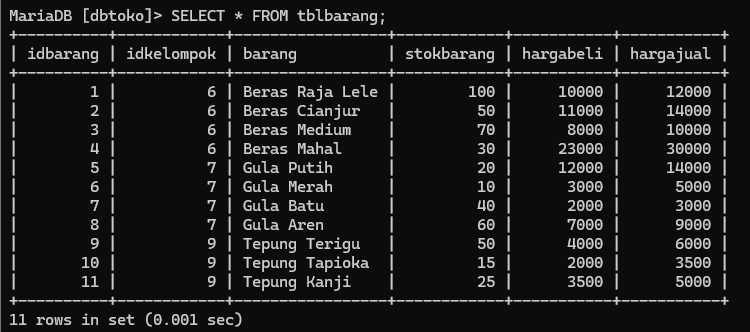
Coba lakukan update pada tabel master kemudian periksa di tabel transaksi. Ubah idkelompok tepung dari 8 menjadi 9 dengan perintah berikut



Periksa perubahan pada tblkelompok

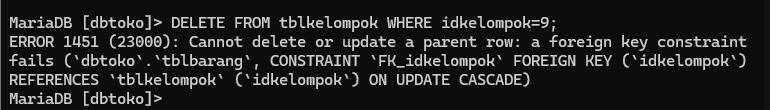


Periksa pada tblbarang



# PENGUJIAN DELETE

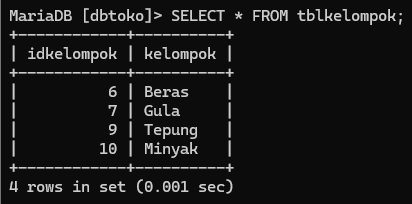
Pengujian delete dilakukan dengan menghapus tabel master.



Karena [ idkelompok] 9 sudah digunakan pada tabel transaksi maka data tersebut tidak bisa dihapus seuai dengan relasi yang dibuat yaitu ON DELETE RESTRICT. Data yang bisa dihapus hanya yang belum digunakan pada tabel transaksi. coba INSERT kan data baru pada tabel master

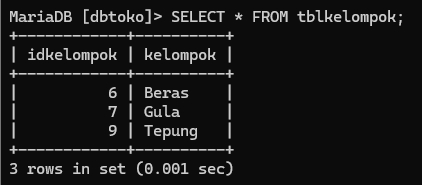


Periksa data yang sudah dimasukan



Karena kolom [idkelompok] 10 belum digunakan pada tabel transaksi maka kolom tersebut bisa dihapus. Lakukan penghapusan menggunakan perintah berikut

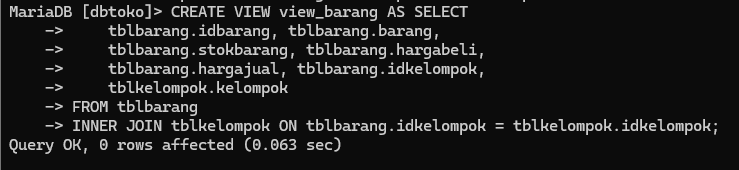


Periksa tabel yang sudah dihapus

# PEMBUATAN VIEW

View digunakan untuk menggabungkan tabel master dan tabel transaksi menjadi sebuah tabel baru agar lebih mudah dilihat dan digunakan.

Kita akan langsung mencobanya

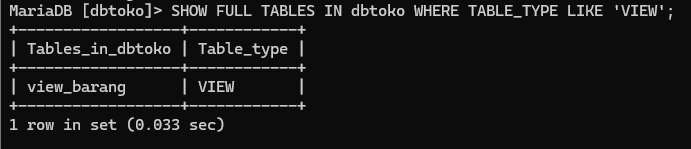


# MENAMPILKAN SEMUA VIEW YANG SUDAH DIBUAT

View yang sudah dibuat bisa dilihat dengan perintah



Kita akan lihat view dari dbtoko

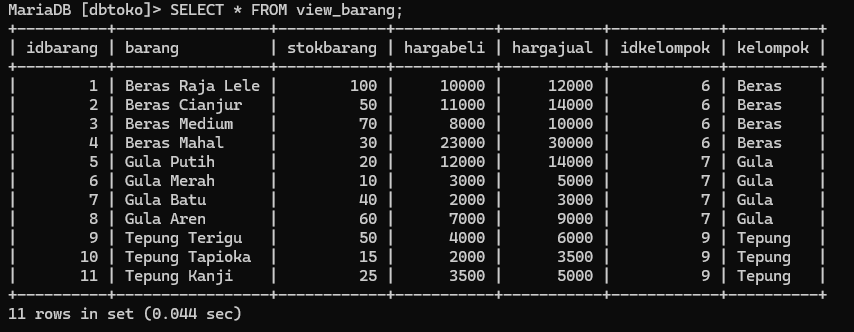


# MELIHAT ISI VIEW Untuk melihat isi VIEW

sama dengan melihat isi table



Kita akan lihat view\_barang



# MENGHAPUS VIEW Jika VIEW

yang sudah dibuat tidak digunakan lagi bisa di hapus dengan menggunakan perintah berikut



Kita akan coba untuk DROP view\_barang

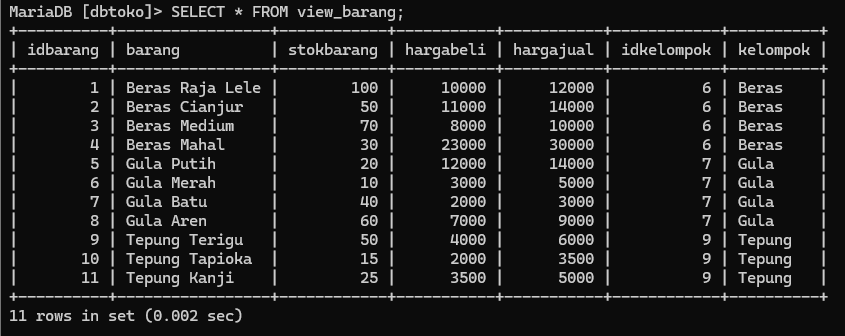


# SELECT SEMUA KOLOM (\*)

Select semua adalah menampilkan semua kolom dari tabel atau VIEW



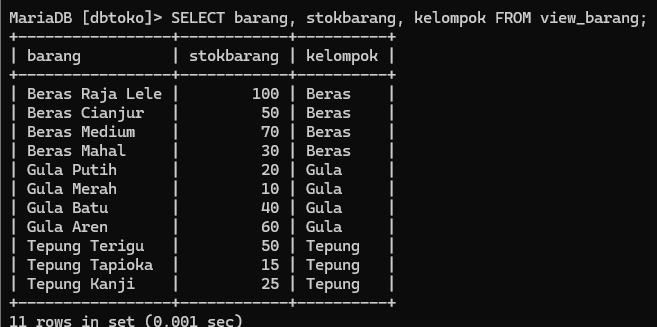
Kita akan buat view lagi dan kita lihat lagi view\_barang



# SELECT SEBAGIAN KOLOM

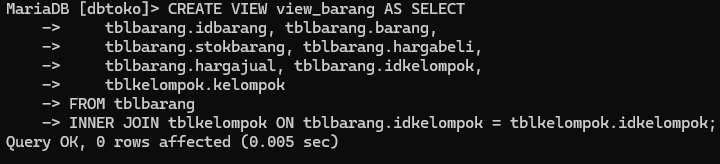
Select sebagian kolom adalah menampilkan hanya pada kolom yang dipilih





# SELECT ORDER Jika VIEW

pada latihan diatas sudah dihapus, silahkan buat ulang untuk latihan materi selanjutnya

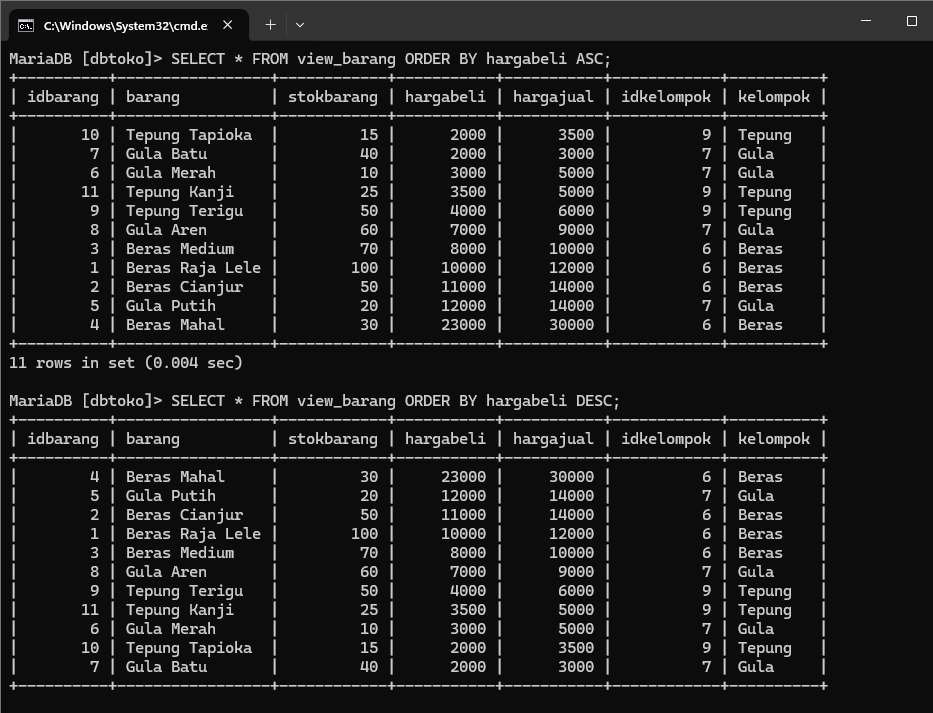


Kita sudah buat, dan kita akan melanjutkan materinya

SELECT ORDER adalah cara untuk menampilkan data dalam urutan naik atau turun Jika NAIK menggunakan ASC Jika TURUN menggunakan DESC



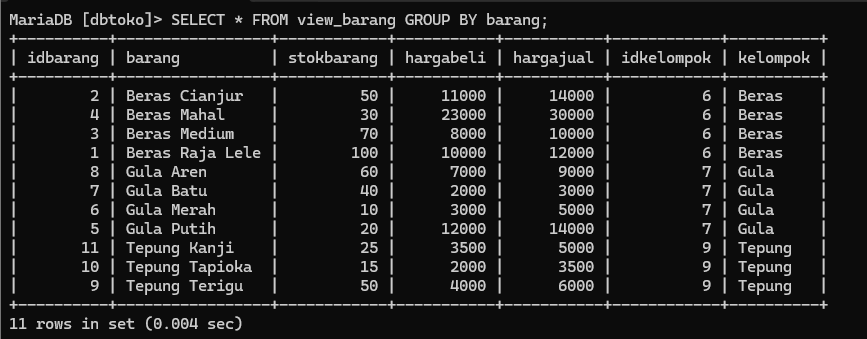
Hasil select ORDER BY nama\_kolom ASC dan DESC



# SELECT GROUP

Perintah SELECT GROUP digunakan untuk mengelompokan data sesuai dengan kelompok dari kolom yang dipilih.

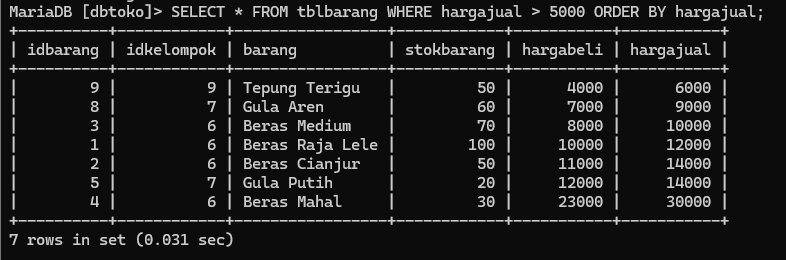


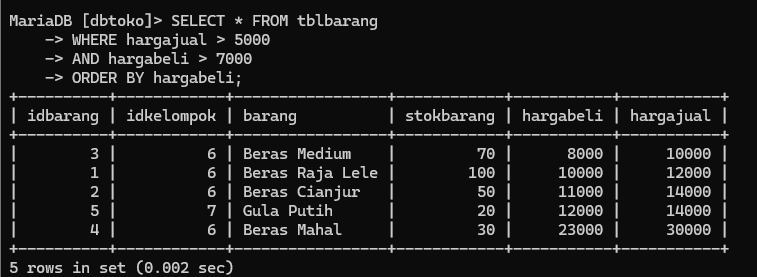


# PENGUJIAN WHERE

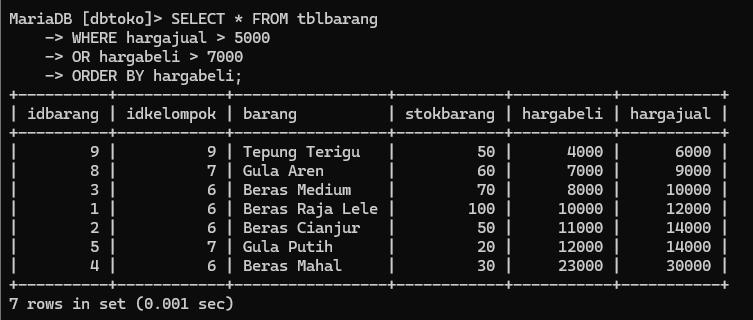


Kita akan mencoba pengujiannya menggunakakn tblbarang





Hasil di atas menampilkan data dari tabel **tblbarang** yang disaring menggunakan kondisi **hargajual > 5000** dan **hargabeli > 7000**, kemudian diurutkan berdasarkan **hargabeli**.  
Dari hasil tersebut, terdapat 5 barang yang memenuhi kriteria, yaitu berbagai jenis beras dan gula dengan harga beli antara **8000 hingga 23000**, serta harga jual antara **10000 hingga 30000**.



hasil di atas menampilkan data dari **tblbarang** dengan kondisi **hargajual > 5000** **atau** **hargabeli > 7000**, lalu diurutkan berdasarkan **hargabeli**.  
Dari hasilnya terdapat **7 barang**, mulai dari **Tepung Terigu** hingga **Beras Mahal**, dengan rentang **harga beli 4000–23000** dan **harga jual 6000–30000**



%a artinya awalan bebas yang penting akhiranya a

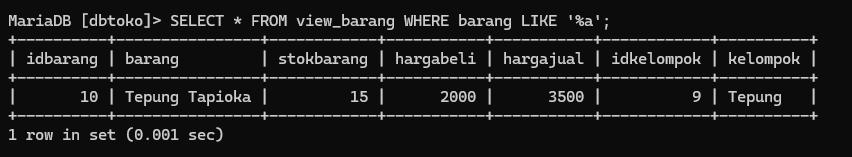


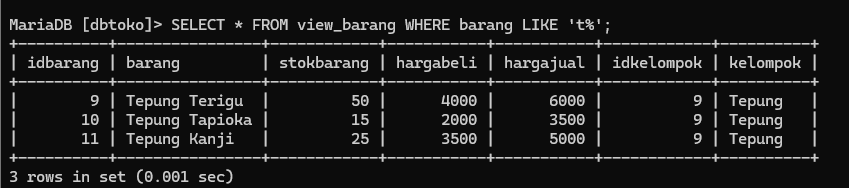
a% artinya inisial a akhiran bebas

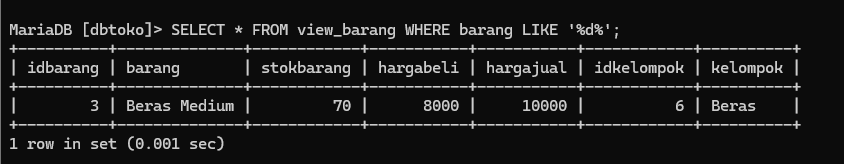


%a% artinya awalan sembarang dan akhiran sembarang yang tengahnya ada huruf a

contoh :





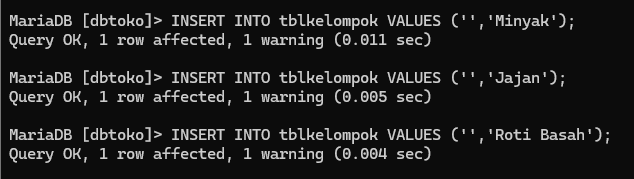


Dan jika tidak ada / kosong, maka



# SUBQUERY (SELECT IN SELECT)

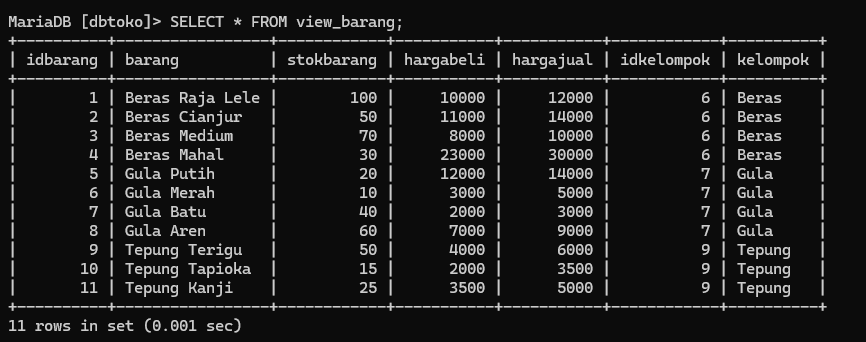
Subquery adalah SELECT yang ada di dalam SELECT. Sebelum berlatih menggunakan subquery, kita akan tambahkan data pada [tblkelompok].



sehingga datanya akan tampil seperti berikut



# Periksa [view\_barang]



Pada [view\_barang] ada kelompok yang sudah menggunakan data dari tabel [tblkelompok]. Ada pertanyaan yang muncul dari data diatas adalah:

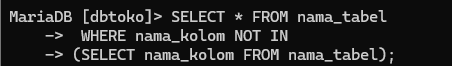
1. Ada berapa data dari tabel [tblkelompok] yang SUDAH digunakan pada VIEW?
2. Ada berapa data yang BELUM digunakan pada VIEW?

Untuk menjawab pertanyaan diatas bisa menggunakan SUBQUERY.

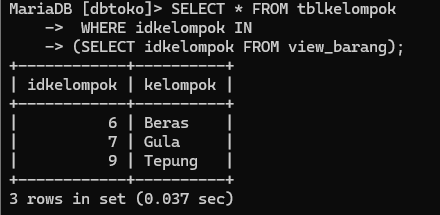
Gunakan perintah



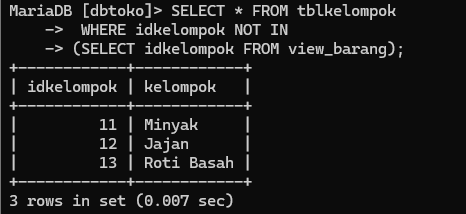
**IN** digunakan untuk **mengambil data** dari tabel utama yang **nilainya cocok** dengan hasil dari subquery.



**NOT IN** digunakan untuk **mengambil data** dari tabel utama yang **tidak ada** dalam hasil subquery.

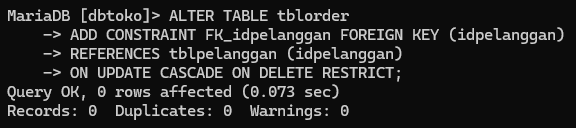


Menampilkan data **yang ada di view\_barang**, yaitu:  
**Beras (6)**, **Gula (7)**, dan **Tepung (9)**.



Menampilkan data **kelompok yang tidak ada di view\_barang**, yaitu:  
**Minyak (11)**, **Jajan (12)**, dan **Roti Basah (13)**.

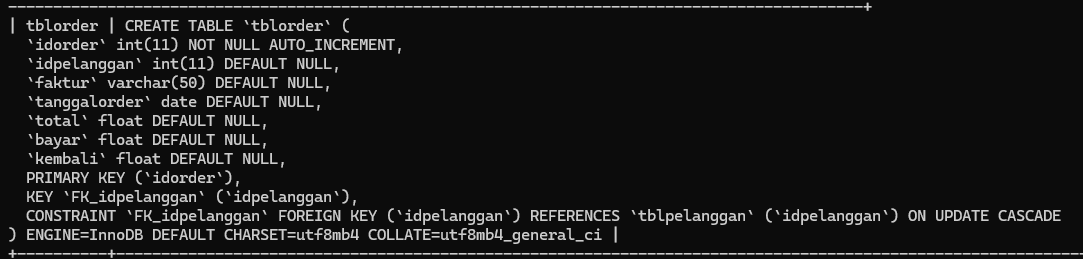
# MEMBUAT RELASI ANTAR TABEL



Periksa hasil pembuatan RELASI dengan perintah berikut

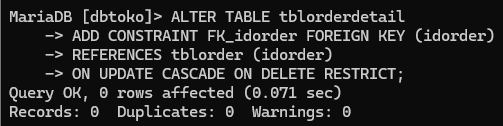


Hasilnya adalah

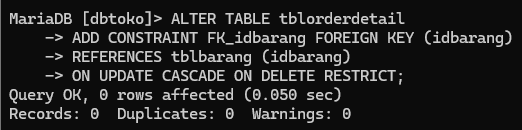


# RELASI 2 TABEL MASTER DAN 1 TABEL TRANSAKSI

Buat relasi antara tabel [tblorderdetail] dan tabel [tblorder]



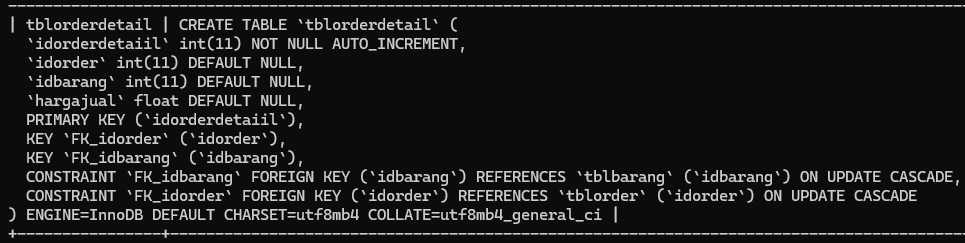
Buat relasi antara tabel [tblorderdetail] dan tabel [tblbarang]



Periksa hasil pembuatan relasi antara 2 tabel master [tblorder] [tblbarang] dan 1 tabel transaksi [tblorderdetail]



Dan ini hasilnya

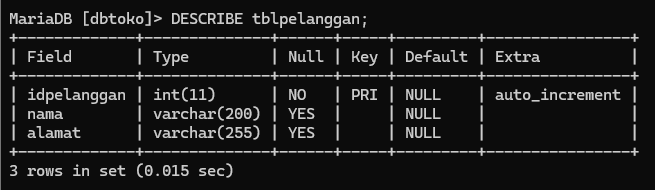


# DUMMY DATA

Dummy data adalah data yang tidak sebenarnya. Data ini diperlukan jika data sebenarnya tidak bisa diperoleh. Dummy data biasa digunakan selama pengujian database.

Pada tabel [tblpelanggan] aplikasi akan di isi dengan dummy data dengan nama pelanggan sebegai berikut

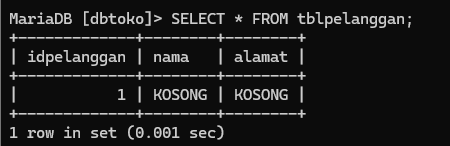
Tampilkan struktur tabel terlebih dahulu



Insert nama pelanggan dengan dummy data



Periksa hasil INSERT data



Pada tabel terdapat nama pelanggan dengan nama [KOSONG], jika terjadi pembelian dengan nama pelanggan yang tidak disebutkan maka aplikasi akan memberi nama pelanggan tersebut dengan nama [KOSONG]. Nama pelanggan dengan nama [KOSONG] ini disebut dengan dummy data.

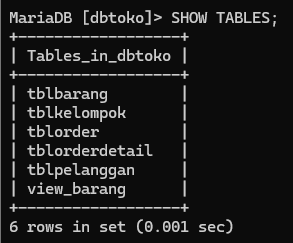
# KONSEP TRIGGER

Trigger adalah perintah INSERT, UPDATE, DELETE, FUNCTION, PROCEDURE yang ditanam pada MySQL yang akan DIJALANKAN pada kejadian berikut:

* AFTER INSERT (setelah INSERT) pada tabel yang dimaksud
* BEFORE INSERT (sebelum INSERT) pada tabel yang dimaksud
* AFTER DELETE (setelah DELETE) pada tabel yang dimaksud
* BEFORE DELETE (Sebelum DELETE) pada tabel yang dimaksud
* AFTER UPDATE (Setelah UPDATE) pada tabel yang dimaksud
* BEFORE UPDATE (Sebelum UPDATE) pada tabel yang dimaksud

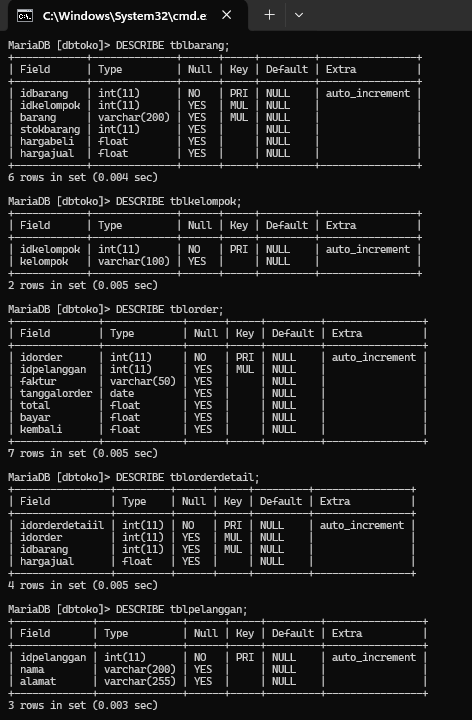
Sebelum membuat trigger pastikan anda memahami proses bisnis dari pembuatan database yang anda lakukan. Sebagai latihan buka semua database yang sudah dibuat pada materi sebelumnya, ikuti langkah berikut

1. TAMPILKAN SEMUA TABEL



# LIHAT STRUKTUR SEMUA TABEL

* [tblbarang]
* [tblkelompok]
* [tblorder]
* [tblorderdetail]
* [tblpelanggan]

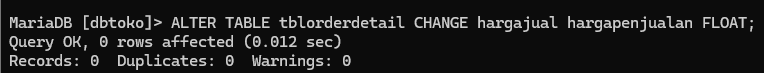


# PENYUSUNAN PROSES BISNIS

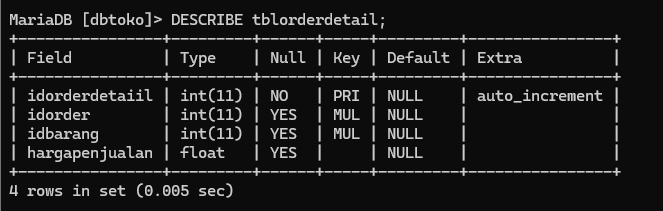
* kolom [stokbarang] pada tabel [tblbarang] akan BERKURANG jika terjadi INSERT data pada tabel [tblorderdetail]
* kolom [stokbarang] pada tabel [tblbarang] akan BERTAMBAH jika terjadi DELETE data pada tabel [tblorderdetail]
* kolom [total] pada tabel [tblorder] akan BERTAMBAH jika terjadi INSERT data pada tabel [tblorderdetail]
* kolom [total] pada tabel [tblorder] akan BERKURANG jika terjadi DELETE data pada tabel [tblorderdetail]

# PEMERIKSAAN NAMA KOLOM YANG SAMA

Sebelum membuat TRIGGER pastikan tidak ada nama kolom yang sama pada setiap tabel. Jika melihat tabel diatas terdapat nama kolom yang sama. Yaitu [hargajual] pada tabel [tblbarang] dan [hargajual] pada tabel [tblorderdetail]. Ubah nama kolom [hargajual] di tabel [tblorderdetail] menjadi [hargapenjualan]



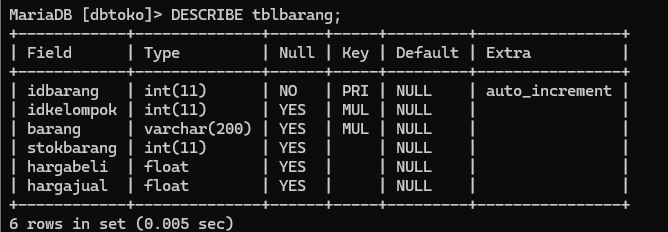
Hasil :



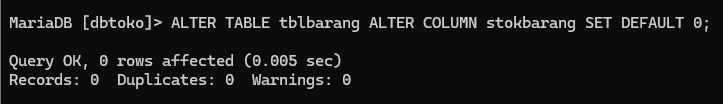
# PEMERIKSAAN KOLOM UNTUK PERUBAHAN HASIL TRIGGER

Lakukan pemeriksaan pada pada kolom yang akan terkena dampak dari proses trigger. Jika yang diproses oleh trigger adalah operasi matematika (penjumlahan, pengurangan, pembagian, perkalian). Pastikan SET DEFAULT VALUE pada kolom tabel tersebut di isi dengan ANGKA NOL. Jika belum di isi angka NOL lakukan ALTER untuk merubah kolom tersebut. Lihat contoh berikut

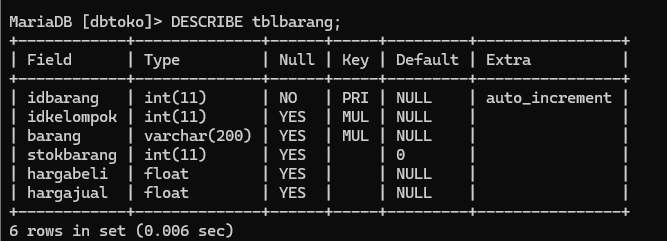
Kolom [stokbarang] belum di SET DEFAULT dengan ANGKA NOL



Command ALTER pada tblbarang

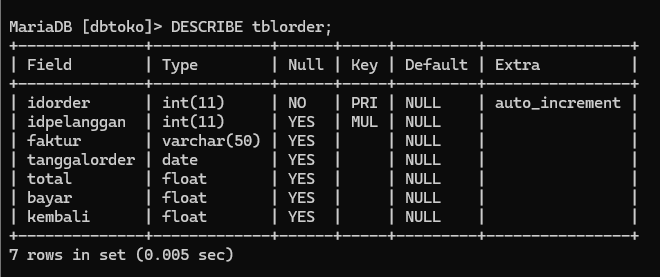


Hasil :

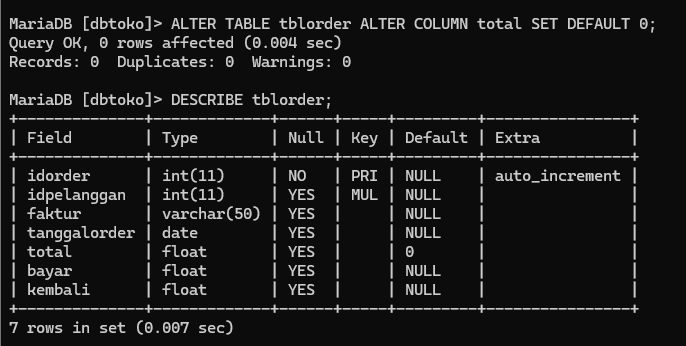


Default pada stokbarang menjadi 0

Pemeriksaan kolom [total] pada tabel [tblorder]. Belum di set DEFAULT ANGKA NOL



Lakukan perubahan dengan ALTER



# PEMBUATAN TRIGGER

Setelah semua konsep, proses bisnis, dan pemeriksaan tabel dilakukan sekarang waktunya membuat TRIGGER. Ada 4 trigger yang akan dibuat.