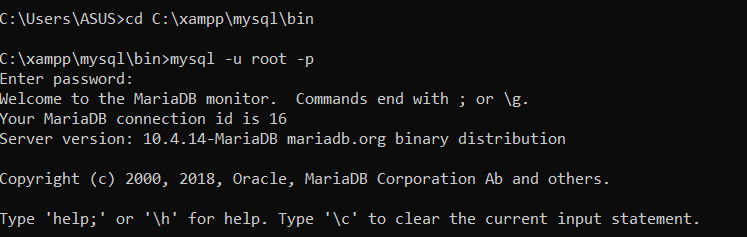
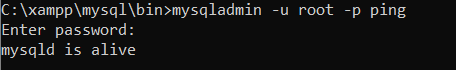
|  |  |
| --- | --- |
| Materi | Nilai |
| Pembelajaran Basis-Data  Video ke 18  (Tugas semester 5 #11) | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

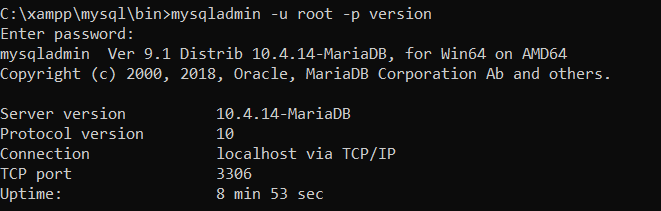
1. Menjalankan Mysql ke dalam cmd setelah menyalakan mysql pada xampp dengan memasukkan lokasi xampp yang sesuai di dalam komputer.



1. Mengecek apakah mysql nyala atau tidak menggunakan cmd.



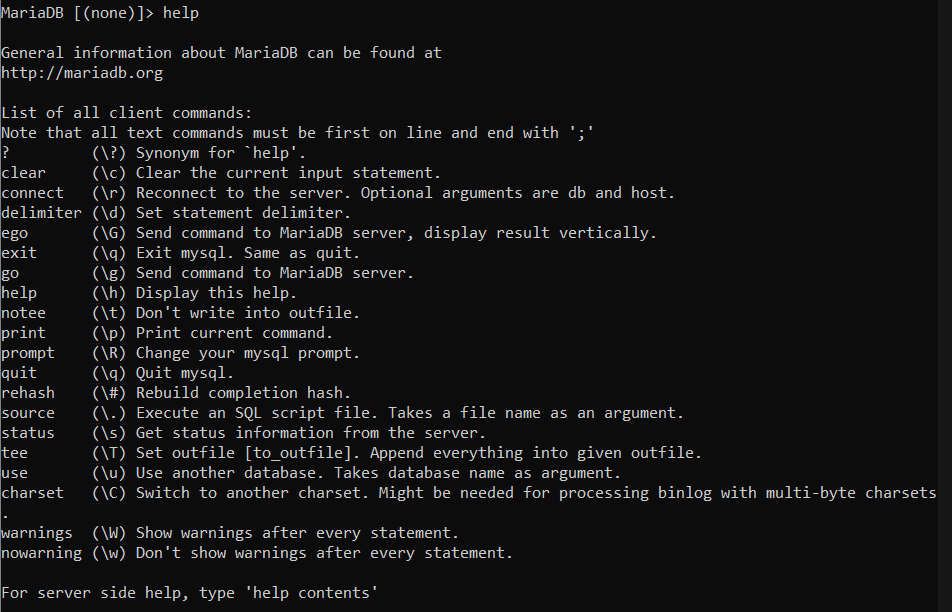
1. Mengecek versi dari mysql yang digunakan menggunakan cmd.



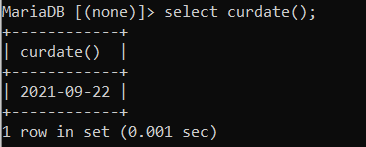
1. Tulisan ketika sudah berhasil masuk ke dalam mysql.



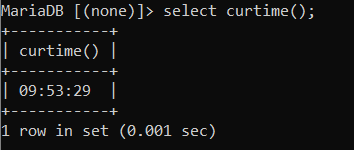
1. Apabila membutuh kan bantuan, maka di dalam cmd kita tinggal menuliskan mariaDB [(none)] > help atau MariaDB[(none)]>?, maka tampilannya akan seperti ini:



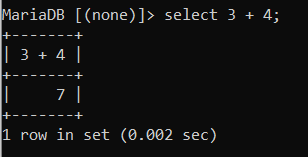
1. Menampilkan tanggal pada mysql menggunakan perintah select curdate();, dimana setiap akhir perintah harus diberi titik koma.



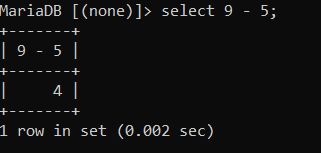
1. Menampilkan waktu atau jam pada mysqk menggunakan perintah select curtime();.



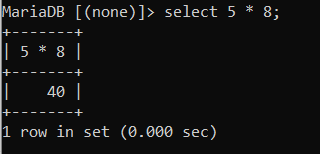
1. Menggunakan kalkulator untuk operasi penjumlahan dimana tanda yang digunakan adalah (+).



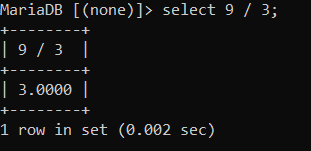
1. Menggunakan kalkulator untuk operasi pengurangan dimana tanda yang digunakan adalah (-).



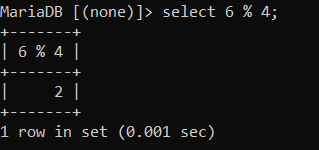
1. Menggunakan kalkulator untuk operasi perkalian dimana tanda yang digunakan adalah (\*).



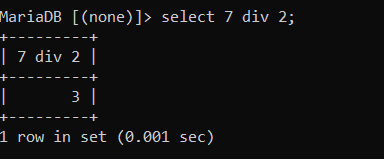
1. Menggunakan kalkulator untuk operasi pembagian dimana tanda yang digunakan adalah (/).



1. Menggunakan kalkulator untuk modulo (sisa hasil pembagian) dengan menggunakan tanda (%).



1. Menggunakan kalkulator untuk hasil pembagian integer utuh dimana menggunakan div untuk operasi ini.



1. DDL (Data Definition Language) merupakan perintah SQL yang berhubungan dengan struktur database dimana DDL adalah proses pembuatan rumah dari data, data akan disimpan pada kolom yang selanjutnya disimpan ke tabel, lalu tabel disimpan ke dalam database.
2. Perintah – perintah yang digunakan dalam DDL :

* CREATE :

1. CREATE DATABASE
2. CREATE FUNCTION
3. CREATE INDEX
4. CREATE VIEW
5. CREATE PROCEDURE
6. CREATE TABLE

* RENAME :

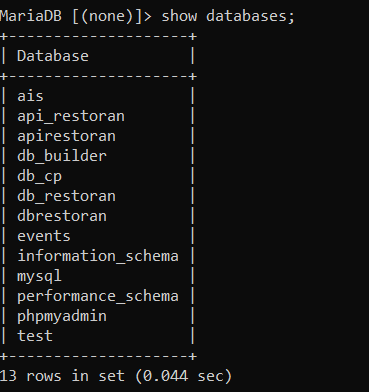
1. RENAME TABLE

* ALTER :

1. ALTER DATABASE
2. ALTER FUNCTION
3. ALTER TABLE
4. ALTER VIEW

* DROP :

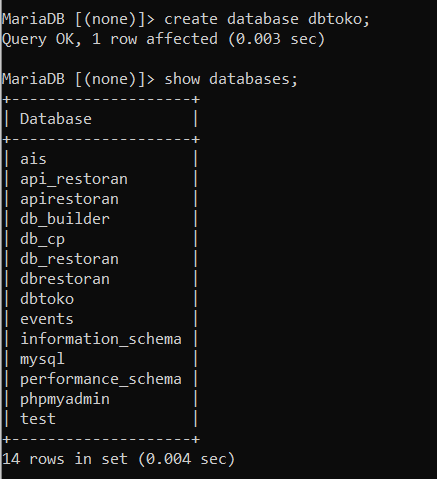
1. DROP DATABASE
2. DROP FUNCTION
3. DROP INDEX
4. DROP PROCEDURE
5. DROP TABLE
6. DROP VIEW
7. SHOW database adalah perintah yang digunakan untuk menampilkan database. Perintah ini akan menampilkan database – database yang ada di dalam computer.



1. SHOW table adalah perintah yang digunakan untuk menampilkan table.
2. SHOW index adalah perintah yang digunakan untuk menampilkan index.
3. DESCRIBE table adalah perintah yang digunakan untuk melihat struktur tabel.
4. Untuk create terdapat beberapa perintah, yaitu :

* Create Database. (digunakan untuk membuat database)
* Create Function. (digunakan untuk membuat function)
* Create Index. (digunakan untuk membuat index)
* Create Procedure. (digunakan untuk membuat procedure)
* Create View. (digunakan untuk mebuat view)
* Create Table. (digunakan untuk membuat table)

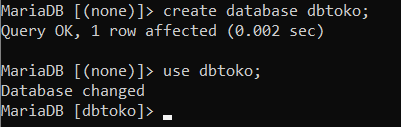
1. CREATE DATABASE dengan nama tabel database adalah dbtoko, dan mengecek menggunakan perintah show databases. Apabila nama db sudah ada, berarti pembuatan dbtoko berhasil.



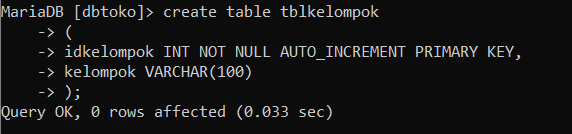
1. RENAME TABLE adalah perintah yang digunakan untuk mengganti nama table.
2. ALTER TABLE … ADD, adalah perintah yang digunakan untuk menambah kolom.
3. ALTER TABLE … MODIFY, adalah perintah yang digunakan untuk merubah tipe data.
4. ALTER TABLE… SET DEFAULT, adalah perintah yang digunakan untuk memberi nilai default atau nilai awal.
5. ALTER TABLE … CHANGE, adalah perintah yang digunakan untuk merubah nama kolom.
6. Index digunakan untuk mempercepat proses pencarian data terutama pada data yang jumlahnya di atas ribuan.
7. CREATE INDEX adalah perintah digunakan untuk membuat index.
8. DELETE adalah perintah yang digunakan untuk mengahpus data, baik database, baik kolom, atau table.
9. DROP database adalah perintah yang digunakan untuk menghapus database yang tidak digunakan.



1. DROP function adalah perintah yang digunakan untuk menghapus function yang tidak digunakan.
2. DROP table adalah perintah yang digunakan untuk menghapus table yang tidak digunakan.
3. DROP view adalah perintah yang digunakan untuk menghapus view yang tidak digunakan.
4. DROP index adalah perintah yang digunakan untuk menghapus index.
5. USE database adalah perintah yang digunakan untuk mengaktifkan database. Setelah dbtoko dihapus, maka buat lagi dbtoko dengan perintah create, lalu aktifkan database dengan menggunakan perintah use. Sehingga setelah diaktifkan, tulisan [none] akan berubah menjadi [dbtoko], yang menunjukkan bahwa database yang aktif untuk digunakan adalah dbtoko.

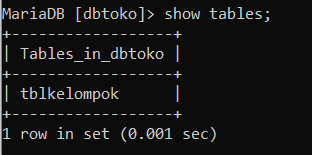


1. CREATE TABLE digunakan untuk membuat table baru dengan perintah seperti ini :



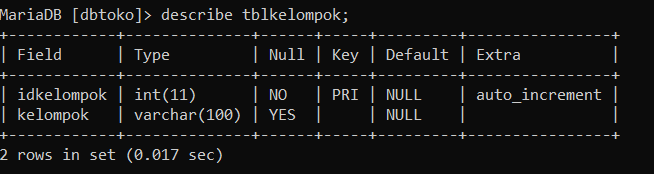
Artinya, membuat table dengan nama tblkelompok dengan kolom idkelompok bertipe data INT dengan idkelompok sebagai AI dan primary key, dan kolom kedua kelompok dengan tipe data varchar yang mempunyai isi maksimal adalah 100.

1. SHOW TABLE digunakan untuk menampilkan table yang sudah dibuat.



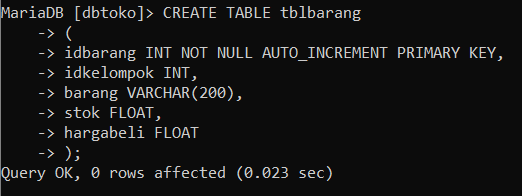
Artinya, di dalam dbtoko mempunyai tabel dengan nama tblkelompok.

1. DESCRIBE nama\_tabel digunakan untuk melihat struktur tabel.



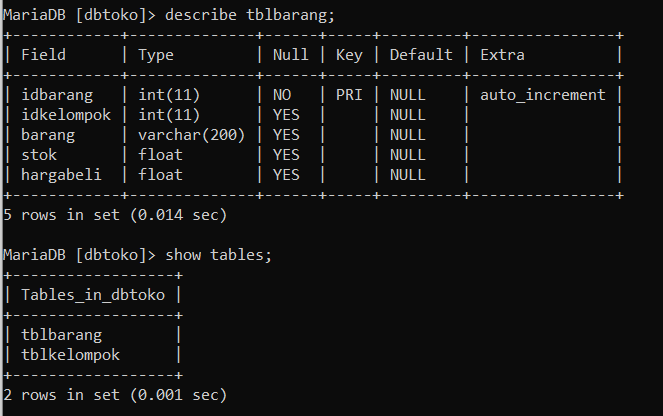
Artinya, di dalam tblkelompok terdapat 2 tabel yaitu idkelompok dan kelompok dimana idkelompok bertipe data int tidak boleh null, key primary key dan auto increment. Dan kolom kelompok bertipe varchar, boleh null. ­

1. CREATE TABLE untuk membuat tblbarang di dalam dbtoko.

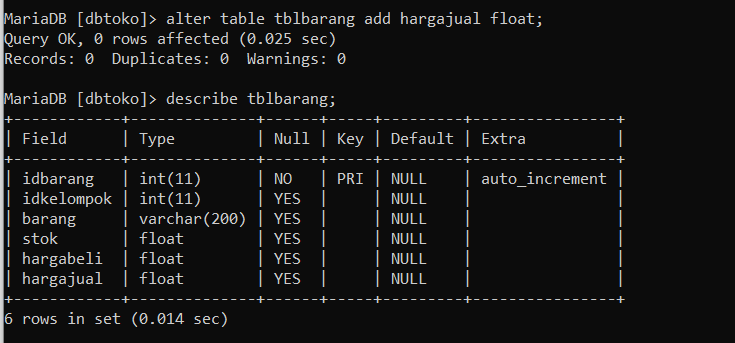


Artinya, di dalam tblbarang terdapat kolom idbarang, idkelompok, barang, stok, dan hargabeli.

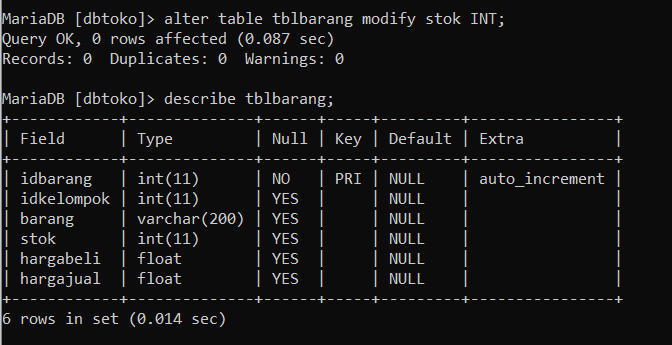
1. Memeriksa struktur tblbarang dengan menggunakan perintah describe tblbarang, dan menampilkan table yang sudah dibuat pada dbtoko menggunakan perintah show tables.



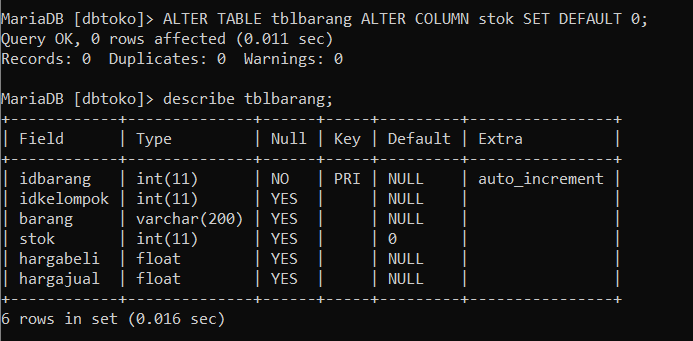
1. Menambah kolom ke dalam tblbarang dengan menggunakan ALTER TABLE tblbarang ADD hargajual float. Setelah menambahkan kolom, mengecek kolom yang baru apakah sudah berhasil ditambahkan atau belom menggunakan perintah describe tblbarang.



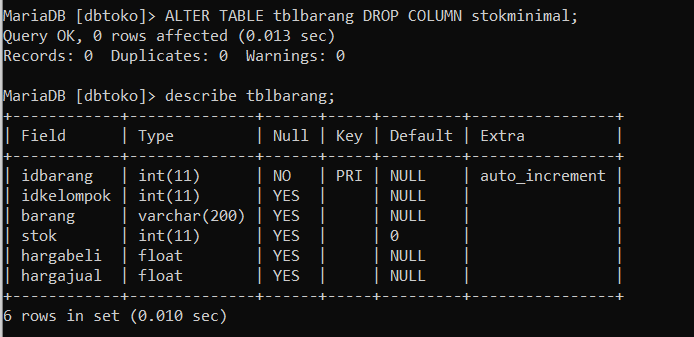
1. Merubah tipe data pada tblbarang dan mengganti tipe data dari stok menjadi int menggunakan perintah ALTER TABLE tblbarang MODIFY stok INT. Setelah itu melakukan pengecekan struktur tabel kembali.



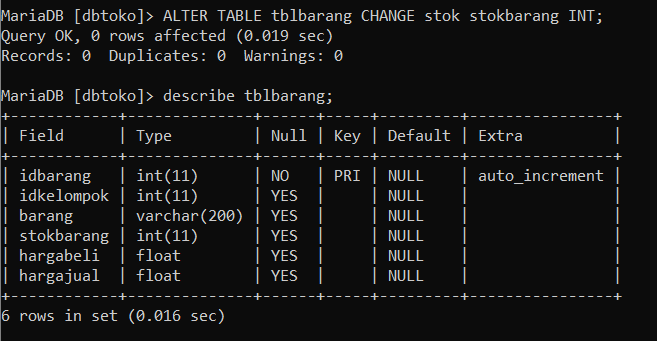
1. Memberi nilai default pada kolom stok di dalam ALTER TABLE dan menambahkan perintah SET DEFAULT. Setelah berhasil menambahkan nilai default, lakukan pengecekan struktur kembali dengan describe tblbarang.



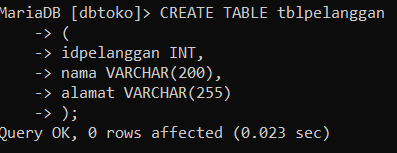
1. Menghapus kolom dengan menggunakan perintaj ALTER TABLE tblbarang DROP COLUMN stokminimal. Setelah menghapus, kembali lakukan pengecekan kolom, apakah kolom berhasil terhapus atau tidak menggunakan perintah describe tblbarang.



1. Mengubah nama kolom dengan menambahkan perintah CHANGE nama\_kolom di dalam ALTER TABLE tblbarang. Setelah melakukan perubahan, cek kembali apakah nama kolom berhasil diubah menggunakan perintah describe tblbarang.

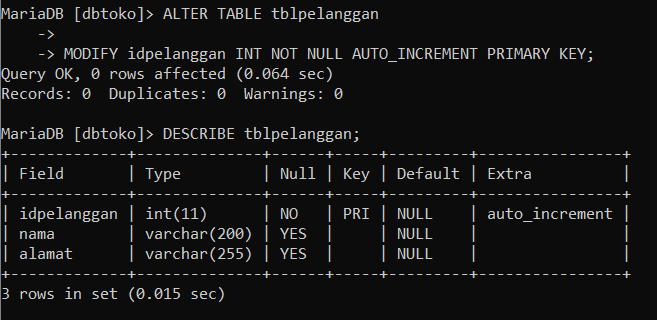


1. Membuat tabel baru yaitu tblpelanggan dengan perintah create table

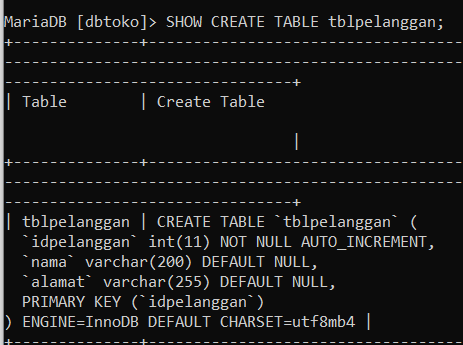


Artinya, di dalam tblpelanggan terdapat idpelanggan bertipe data int, nama bertipe data varchar, dan alamat bertipe data varchar.

1. Menambahkan primary key di dalam tblpelanggan dengan perintah ALTER TABLE tblpelanggan MODIFY. Setelah itu, cek kembali apakah primary key berhasil ditambahkan menggunakan peritah describe tblpelanggan.

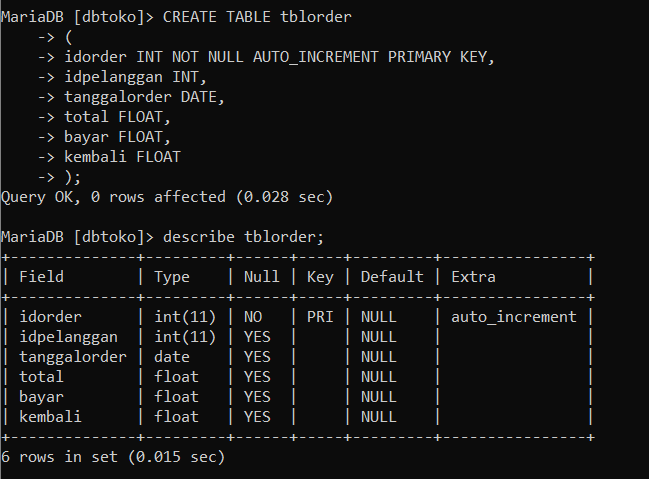


1. Di dalam pembuatan database, terdapat 2 engine, yaitu engine MYISAM dan InnoDB. Pada pembuatan database relasional harus menggunakan InnoDB karena MYISAM tidak bisa menerima penggunaan FOREIGN KEY.



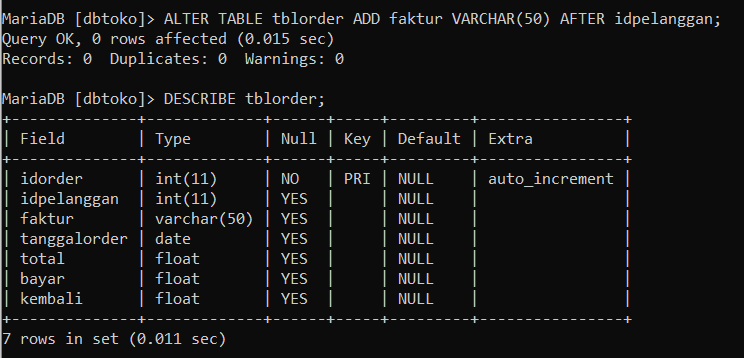
Di dalam tblpelanggan, engine yang digunakan adalah InnoDB.

1. Membuat tblorder dengan perintah CREATE TABLE tblorder.



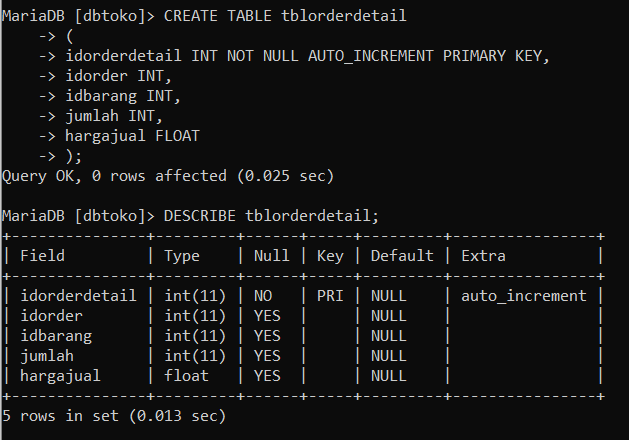
Di dalam tblorder terdapat beberapa kolom, yaitu idorder, idpelanggan, tanggalorder, total, bayar, dan kembali. Setelah itu dilakukan pengecekan struktur dari tblorder dimana idorder menjadi primary key dalam tblorder.

1. Menambahkan kolom setelah kolom menggunakan perintah ALTER TABLE ADD namakolombaru setelah itu menambahkan namakolomsebelumnya.



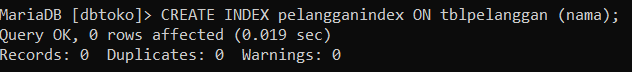
Kolom yang ditambahkan adalah kolom faktur yang terletak di bawah kolom idpelanggan, dimana kolom faktur bertipe data varchar.

1. Membuat table tblorderdetail dengan perintah CREATE TABLE.



Di dalam tblorderdetail terdapat kolom idorderdetail bertipe data int sebagai primarykey, kolom idorder bertipe data int, kolom idbarang bertipe data int, kolom jumlah bertipe data int, dan kolom hargajual bertipe data float.

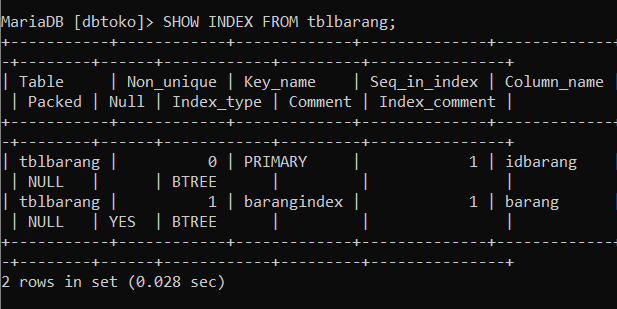
1. Membuat index dengan menggunakan perintah CREATE INDEX.



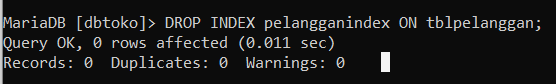


Artinya, membuat index pelangganindex yang ada pada tblpelanggan yang mengambil kolom nama. Dan membuat index barangindex yang ada pada tblbarang yang mengambil kolom barang.

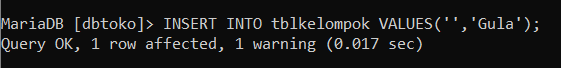
1. Menampilkan index dengan menggunakan perintah SHOW INDEX FROM tblbarang.



1. Menghapus index dengan menggunakan perintah DROP INDEX pelangganindex ON tblpelanggan.

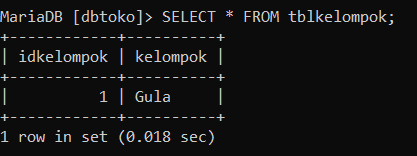


1. DML (Data Manipulation Language) merupakan perintah SQL yang berhubungan dengan pengolahan data dalam table.
2. DML memiliki 4 perintah, yaitu SELECT, INSERT, UPDATE, DELETE.
3. INSERT INTO nama\_table adalah perintah yang digunakan untuk memasukkan data pada semua kolom tabel.
4. INSERT INTO nama\_tabel VALUES isi, adalah perintah yang digunakan untuk memasukkan data hanya pada sebagian kolom yang digunakan.



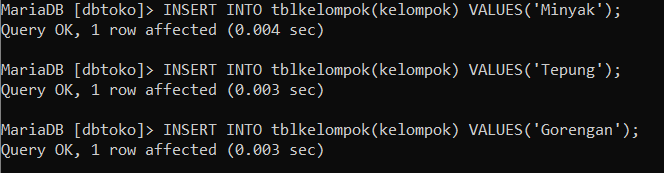
Artinya, perintah INSERT INTO untuk memasukkan data pada tblkelompok dengan VALUES Gula.

1. SELECT \* FROM nama\_tabel adalah perintah yang digunakan untuk mengambil semua data yang ada pada tabel.

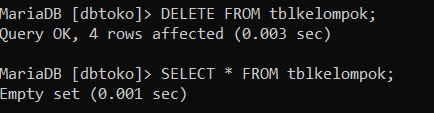


Artinya, setelah memasukkan data ke dalam tblkelompok, untuk mengecek apakah data gula berhasil dimasukkan yaitu dengan perintah SELECT \* FROM tblkelompok.

1. Memasukkan data hanya pada kolom yang disebutkan yaitu dengan menggunakan INSERT INTO namatabel(namakolom) VALUES.



1. DELETE FROM nama\_table adalah perintah yang digunakan untuk menghapus semua baris data.



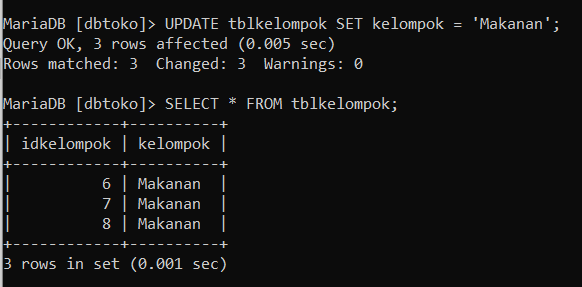
Artinya, menghapus semua data yang ada pada tblkelompok menggunakan perintah DELETE FROM tblkelompok, sehingga setelah data berhasil dihapus, maka isi dari table menjadi kosong.

1. DELETE FROM nama\_table WHERE baris\_dihapus, adalah perintah yang digunakan untuk menghapus sebagian baris.



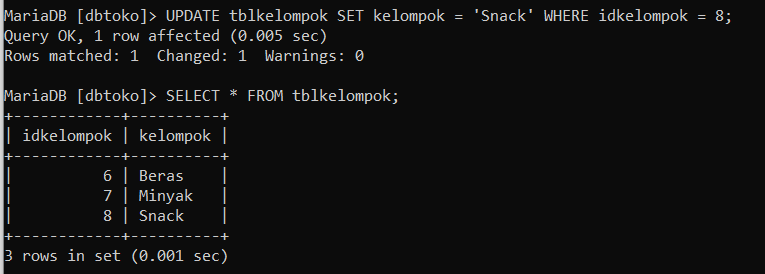
Artinya, menghapus data menggunakan perintah DELETE FROM tblkelompok dimana data yang dihapus adalah yang mempunyai idkelompok 5.

1. UPDATE nama\_table SET nama\_kolom=isi\_kolom adalah perintah yang digunakan untuk mengubah data pada semua kolom.



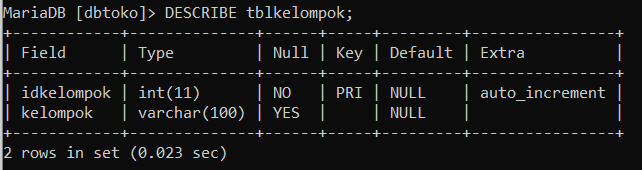
Artinya, mengubah semua isi kolom pada kolom kelompok menjadi makanan dengan perintah UPDATE tblkelompok SET kelompok = ‘Makanan’. Setelah selesai melakukan update, cek kembali apakah data pada tblkelompok kolom kelompok berhasil diupdate menjadi makanan atau tidak.

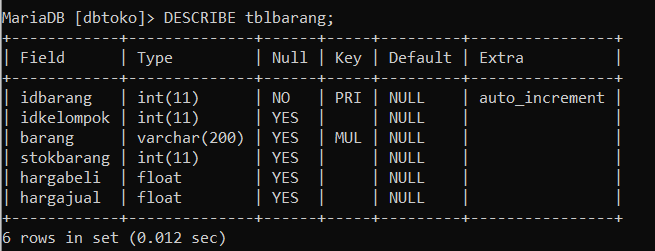
1. UPDATE nama\_tabel SET nama\_kolom=isi\_kolom WHERE nama\_kolom=baris adalah perintah yang digunakan untuk melakukan perubahan pada sebagian kolom.



Artinya, mengupdate data menggunakan perintah UPDATE dengan mengatur isi kolom kelompok adalah snack dimana data dari idkelompok yang diupdate adalah 8. Setelah selesai melakukan update, maka cek kembali apakah data pada idkelompok 8 berhasil diupdate atau tidak.

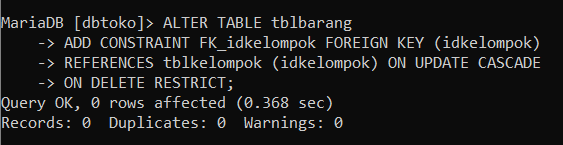
1. SELECT \* FROM nama tabel WHERE namakolom adalah perinah yang digunakan untuk mengambil data dari kolom yang dipilih.
2. SELECT \* FROM nama\_view adalah perintah yang digunakan untuk melihat isi view sama dengan isi tabel.
3. Tabel master adalah tabel yang memasuki tabel lain.
4. Tabel transaksi (detail) adalah tabel yang dimasuki tabel lain.





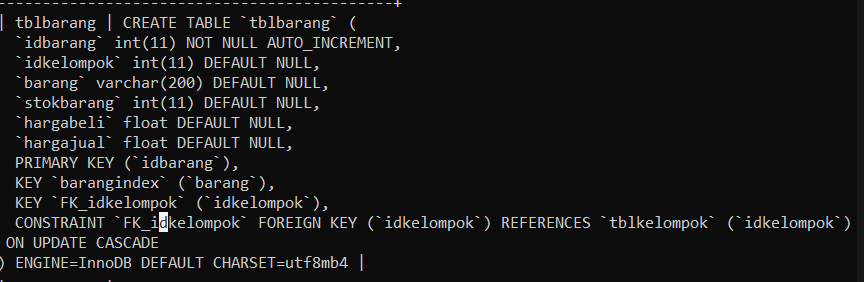
* Di dalam idkelompok pada tblkelompok memasuki tblbarang, jadi tblkelompok menjadi tabel master.
* Di dalam tblbarang dimasuki kolom idkelompok dari tblkelompok, jadi tblbarang menjadi tabel transaksi.

1. Primary key adalah kolom yang berbeda antara baris data yang satu dengan yang lain sehingga setiap baris data mempunyai nilai primary key yang unik.
2. Foregin key atau secondary key adalah atribut yang digunakan untuk pengambilan data, jadi secondary key adalah atribut yang menjadi primary key di dalam tabel lain, dan menjadi secondary key di dalam satu tabel digunakan untuk pengambilan data.
3. Membuat relasi antar tabel dengan menggunakan perintah ALTER TABLE namatabel ADD CONSTRAINT.

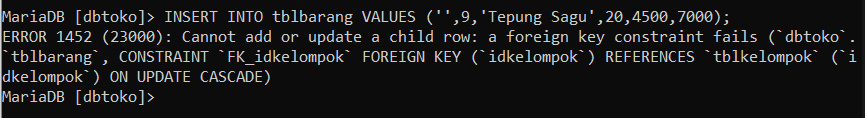


Artinya, membuat relasi pada tblbarang dengan menambahkan foreign key dari tblkelompok, dengan mengambil data dari tabel master dengan perintah ON UPDATE CASCADE apabila data pada tblmaster berubah, maka kolom foregin key pada tabel transaksi juga akan berubah. Dan perintah ON DELETE RESTRICT apabila jika baris kolom master dihapus maka semua baris data yang menggunakan isi tabel master juga akan ikut terhapus.

Tampilan jika relasi berhasil dibuat :

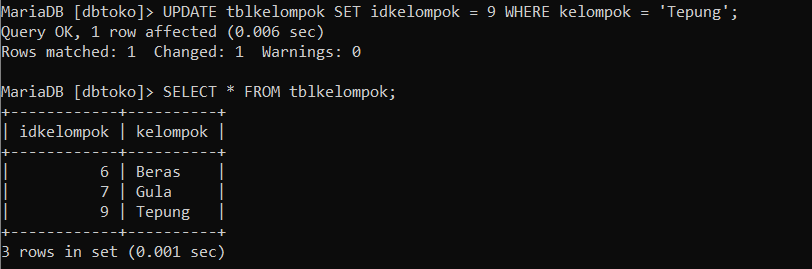


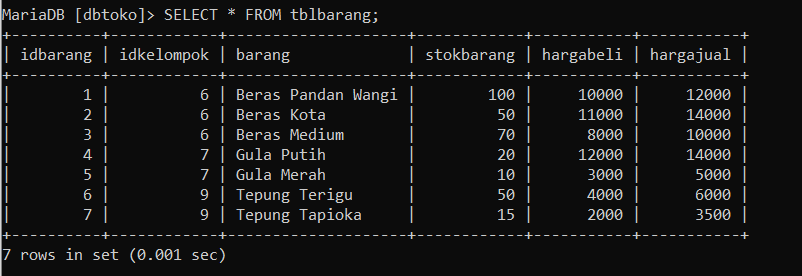
1. ON UPDATE CASCADE adalah pernyataan jika kolom primary key pada tabel master berubah, maka kolom foreign key pada tabel transaksi juga akan berubah.
2. ON DELETE RESTRICT adalah pernyataan jika baris pada kolom master dihapus, maka semua baris data yang menggunakan isi tabel master pada tabel transaksi juga akan terhapus.
3. Melakukan pengujian insert data pada kolom relasi.



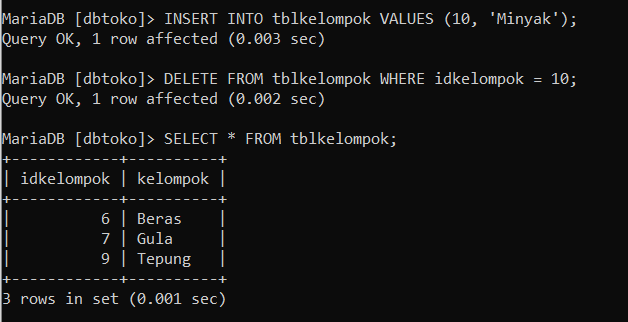
Karena isi dari idkelompok hanya 6,7,8, maka data 9 yang dimasukkan tidak bisa diproses atau menjadi error. Karena data dgn idkelompok 9 tidak tersedia pada tblkelompok.

1. Melakukan update data pada idkelompok 8, dimana diubah menjadi idkelompok 9, sehingga data yang menggunakan idkelompok 8 juga nilai nya akan berganti menjadi idkelompok 9.

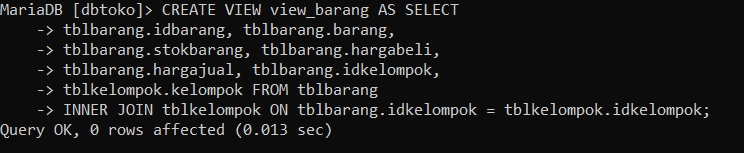




1. Melakukan penghapusan data pada kolom kelompok, karena idkelompok 9 sudah digunakan untuk tabel relasi, maka tidak bisa dilakukan penghapusan. Jadi menambahkan data, lalu melakukan penghapusan dengan perintah DELETE FROM tblkelompok WHERE idkelompok = 10.

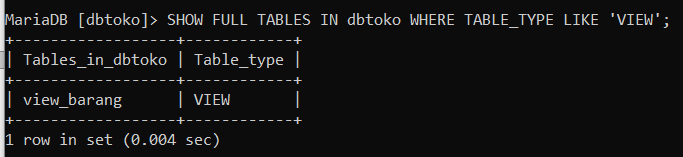


1. View digunakan untuk menggabungkan table master dan tabel transaksi menjadi sebuah tabel baru agar lebih mudah untuk dilihat dan digunakan.
2. CREATE VIEW digunakan untuk membuat view baru.



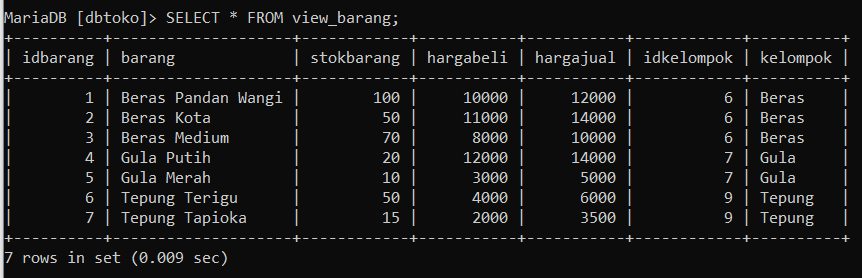
Artinya, view yang dibuat bernama view\_barang, dimana kolom yang dimasukkan adalah idbarang, barang, hargabeli, hargajual, idkelompok, kolom – kolom tersebut diambil dari tblbarang dann mengambil kolom kelompok pada tblkelompok yang mengambil data dari tblbarang, dimana nilai idkelompok pada tblbarang sama dengan nilai idkelompok pada tblkelompok.

1. Menampilkan view yang sudah dibuat dengan menggunakan perintah SHOW FULL TABLES IN namadb WHERE TABLE\_TYPE LIKE ‘VIEW’.



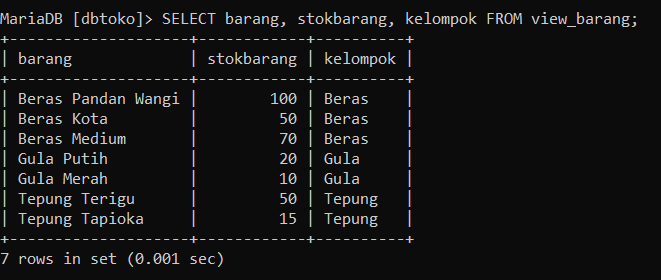
Artinya, di dalam dbtoko terdapat satu view yaitu view yang bernama view\_barang.

1. SELECT \* FROM nama\_view adalah perintah yang digunakan untuk menampilkan isi dari view.

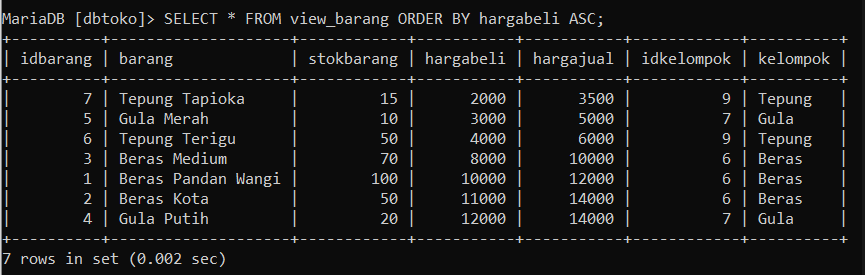


Artinya, di dalam view barang terdapat beberapa kolom, yaitu idbarang, barang, stokbarang, hargabeli, hargajual, idkelompok, dan kelompok.

1. Menampilkan beberapa kolom view menggunakan perintah SELECT namakolom1, 2, 3 FROM namaview;

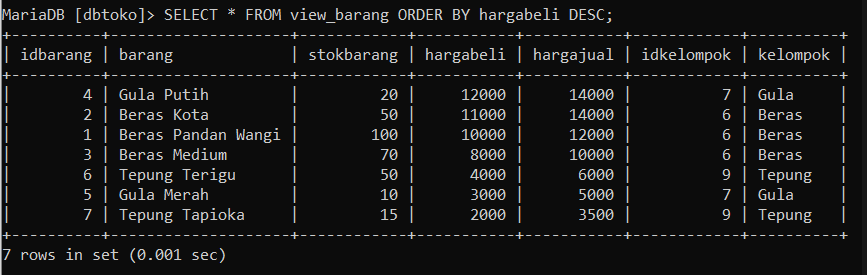


1. SELECT ORDER adalah cara yang digunakan untuk menampilkan data dalam urutan naik atau turun. Apabila urutan data diatur ASC, maka data akan diurutkan dari nilai yang kecil ke nilai yang besar. Tetapi jika urutan data adalah DESC, maka data yang diurutkan adalah dari data bernilai besar ke kecil.
2. SELECT ORDER dengan urutan data ASC.



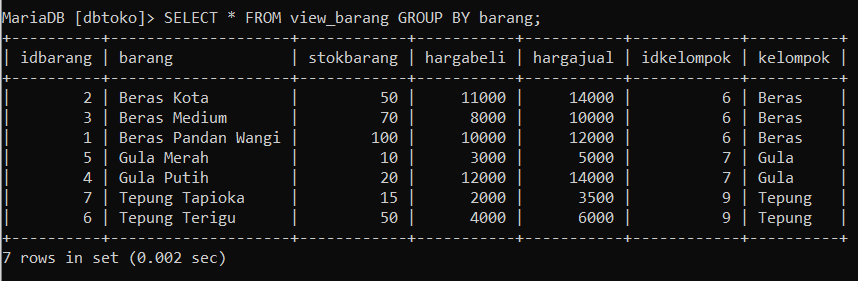
Artinya, menampilkan data pada view\_barang yang diurutkan dari hargabeli dengan mengurutkan dari nilai terkecil ke nilai terbesar, nilai terkecil 2000 dan nilai terbesar 12000.

1. SELECT ORDER dengan urutan data DESC.

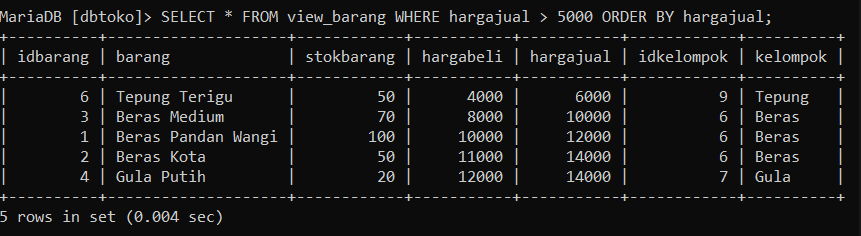


Artinya, menampilkan data pada view\_barang yang diurutkan dari hargabeli dengan mengurutkan dari nilai terbesar ke nilai terkecil, nilai terbesar 12000 dan nilai terkecil 2000.

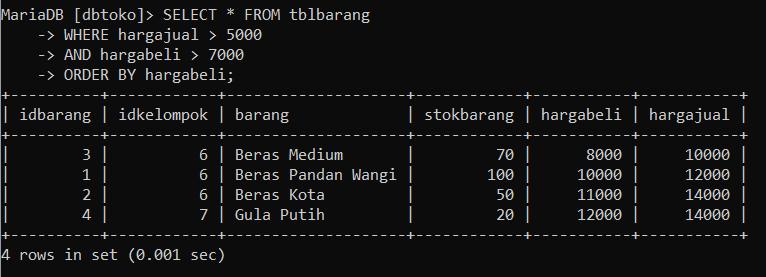
1. SELECT GROUP adalah cara yang digunakan untuk mengelompokkan data sesuai dengan kelompok dari kolom yang dipilih.



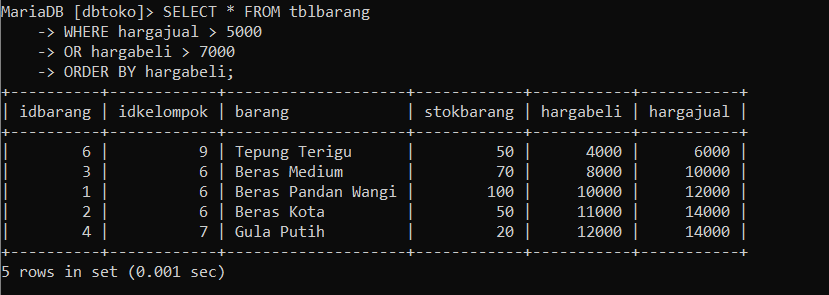
1. Pengujian menggunakan where dimana pengujian dapat menggunakan SELECT, UPDATE, DELETE dan menggunakan operator pembanding, operator logika, dan select.



Menampilkan data dengan pengujian data yang bisa ditampilkan hanya data yang harga jualnya lebih dari 5000.

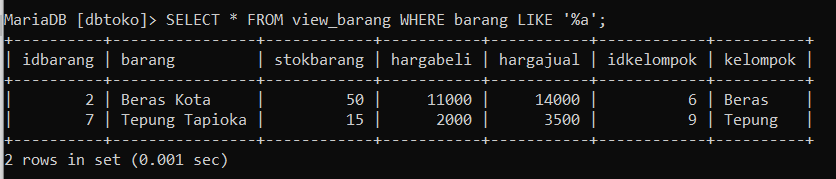


Menampilkan data dengan pengujian data yang bisa ditampilkan hanya data yang harga jualnya lebih dari 5000 dan hargabeli lebih dari 7000.



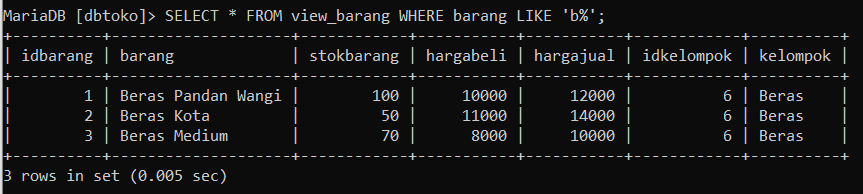
Menampilkan data dengan pengujian data yang bisa ditampilkan hanya data yang harga jualnya lebih dari 5000 atau data yang harga belinya lebih dari 7000.

1. Pengujian menggunakan LIKE % adalah sebutan untuk sembarang atau apapun.
2. Pengujian menggunakan %a artinya awalan sembarang yang penting akhirannya a.



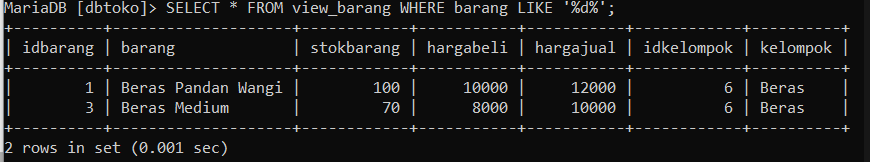
Terdapat 2 barang yang kata akhirannya adalah a, yaitu beras kota dan tepung tapioka.

1. Pengujian menggunakan b% artinya awalan b akhiran sembarang.



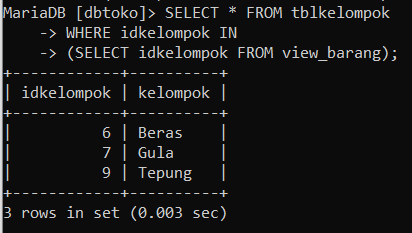
Terdapat 3 barang dengan awalan b dan akhiran sembarang, yaitu beras pandan wangi, beras kota, dan beras medium.

1. Pengujian menggunakan %d% artinya awalan dan akhiran sembarang yang tengahnya ada huruf d.



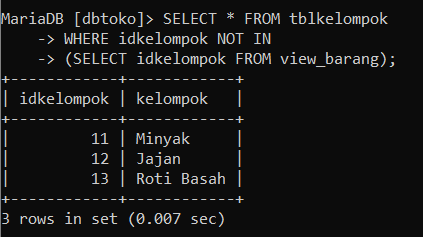
Terdapat 2 barang dengan awalan dan akhiran sembarang, tetapi huruf tengahnya ada huruf d, yaitu beras pandan wangi dan beras medium.

1. SUBQUERY (SELECT IN SELECT), subquery adalah SELECT yang ada di dalam SELECT.
2. Menampilkan data yang dari tblkelompok yang sudah ada di dalam tabel dengan memeriksa nya menggunakan perintah SELECT \* FROM namatabel WHERE namakolom IN.



Terdapat 3 barang yang sudah digunakan pada view\_barang, yaitu beras, gula, dan tepung.

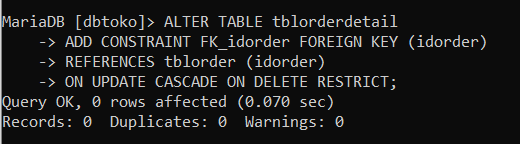
1. Menampilkan data dari tblkelompok yang belom digunakan di dalam tabel dengan memeriksanya menggunakan perintah SELECT \* FROM namatabel WHERE namakolom NOT IN.



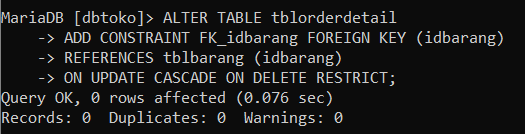
Terdapat 3 barang yang belom digunakan ke dalam view\_barang, yaitu minyak, jajan, dan roti basah.

1. Membuat relasi yang berisi 2 tabel master dan 1 tabel transaksi.

Pertama, membuat relasi antara tabel tblorderdetail dan tblorder

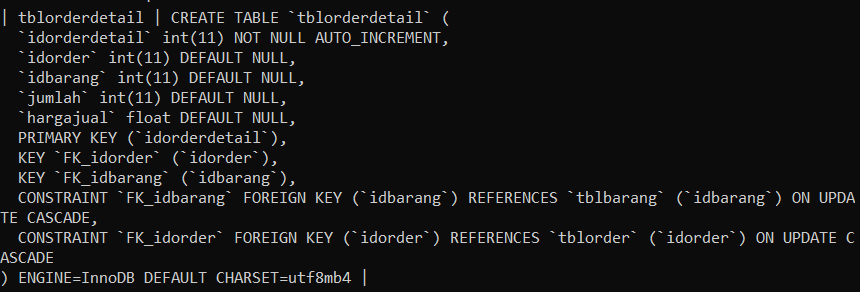


Kedua, membuat relasi antara tblorderdetail dan tblbarang

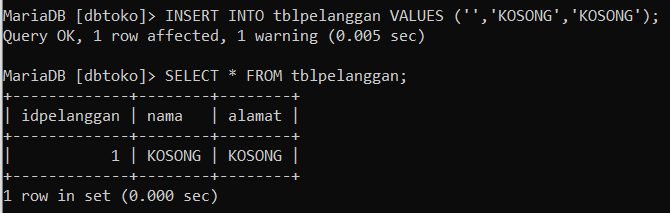


Jadi, tabel masternya adalah tblorder dan tblbarang lalu tabel transaksi nya adalah tblorderdetail.

Menampilkan hasil pembuatan relasi dengan menggunakan perintah SHOW CREATE TABLE tblorderdetail



1. DROP VIEW nama\_view adalah perintah yang digunakan untuk menghapus view yang tidak digunakan.
2. Dummy data adalah data yang tidak sebenarnya, dimana data ini diperlukan jika data sebenarnya tidak bisa diperoleh. Dummy data biasa digunakan selama pengujian database.

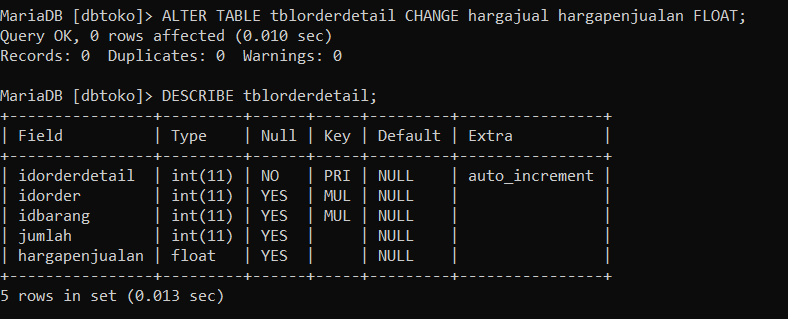


Di dalam tblpelanggan terdapat data pada kolom nama dan kolom alamat yang berisi KOSONG, maksudnya apabila terjadi pembelian dengan nama dan alamat pelanggan yang tidak diisikan, maka aplikasi akan memberi nama dan alamat pelanggan tersebut dengan data KOSONG.

1. Trigger adalah perintah INSERT, UPDATE, DELETE, FUNCTION, PROCEDURE yang ditanam pada MySQL yang akan dijalankan pada kejadian :

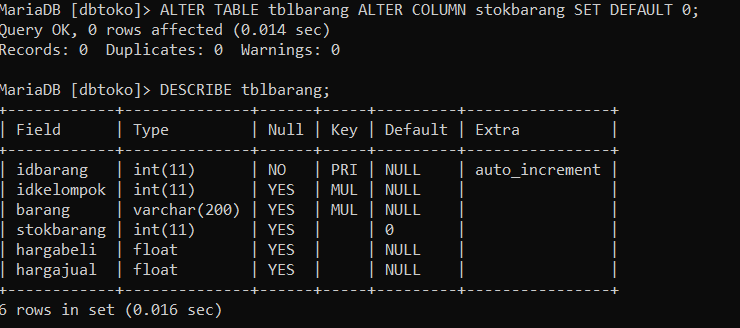
* AFTER INSERT (setelah insert) pada tabel yang dimaksud.
* BEFORE INSERT (sebelum insert) pada tabel yang dimaksud.
* AFTER DELETE (setelah delete) pada tabel yang dimaksud.
* BEFORE DELETE (sebelum delete) pada tabel yang dimaksud.
* AFTER UPDATE (setelah update) pada tabel yang dimaksud.
* BEFORE UPDATE (sebelum delete) pada tabel yang dimaksud.

1. Sebelum pembuatan trigger, kita harus memastikan tidak ada nama kolom yang sama pada setiap tabel, apabila terdapat nama tabel yang sama, maka nama dari salah satu table harus diganti, dengan menggunakan perintah ALTER TABLE namatabel CHANGE nama kolom lama namakolombaru FLOAT.



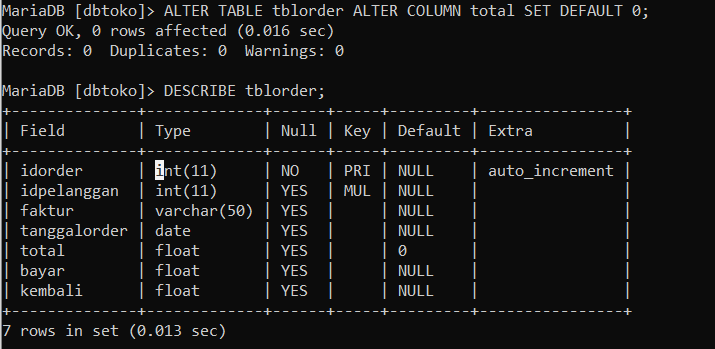
Terdapat perintah untuk mengganti nama dari kolom hargajual menjadi kolom hargapenjualan dengan perintah CHANGE. Setelah melakukan perubahan perintah, cek kembali apakah perubahan pada kolom hargajual sudah berubah menjadi hargapenjualan atau belom. Untuk mengecek menggunakan perintah DESCRIBE tblorderdetail.

1. Memeriksa kolom pada tblbarang yang akan berdampak setelah proses trigger, apabila kolom yang diproses adalah operasi matematika, maka nilai default value adalah 0.



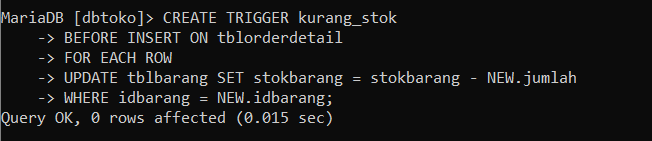
Terdapat perintah untuk mengatur nilai default adalah 0, yaitu dengan peirntah SET DEFAULT 0 pada stokbarang. Setelah mengatur nilai default pada kolom stokbarang, maka cek kembali apakah nilai default sudah berhasil menjadi 0 atau tidak.

1. Memeriksa kolom pada tblorder yang akan berdampak setelah proses trigger, yaitu kolom total akan berisi operasi matematika masih bernilai null, maka harus diganti menjadi 0.

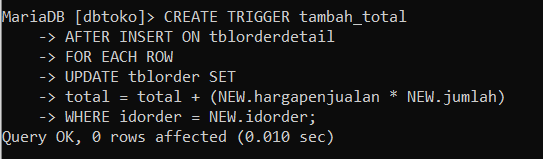


Perintah untuk mengubah nilai default dari null menjadi 0 adalah dengan SET DEFAULT 0. Setelah menambahkan perintah pengaturan nilai default, cek kembali apakah nilai default berhasil dibuah menjadi nilai 0.

1. Membuat trigger kurang\_stok, dimana trigger digunakan untuk mengurangi data dari stokbarang SEBELUM INSERT pada tabel orderdetail.

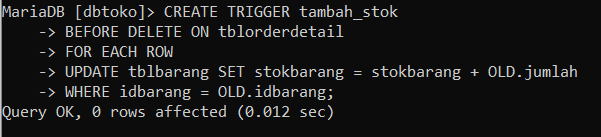


Jadi, nilai dari stokbarang pada tblbarang akan diupdate dan diatur nilai nya dengan operasi pengurangan dari stokbarang dan jumlah saat BEFORE INSERT sebelum data akan dimasukkan ke dalam tblorderdetail. Maksud dari NEW.jumlah adalah nilai yang akan masuk pada kolom jumlah di dalam tblorderdetail. Jadi jumlah stokbarang akan dikurangi ketika sebelum jumlah barang diinsertkan ke dalam tblorderdetail, karena untuk menjalankan operasi pengurangan, dibutuhkan jumlah barang yang dieksekusi sebelum dimasukkan ke dalam tblorderdetail. Maka dari itu, kondisi yang digunakan adalah menggunakan BEFORE INSERT ON. Karena stokbarang akan dieksekusi / dikurangi saat sebelum memasukkan data ke dalam tblorderdetail, sehingga setelah data masuk, maka nilai dari stokbarang sudah berubah.

1. Membuat trigger tambah\_total, dimana trigger ini digunakan untuk menambah nilai total dari kolom total SESUDAH INSERT pada tblorderdetail. 

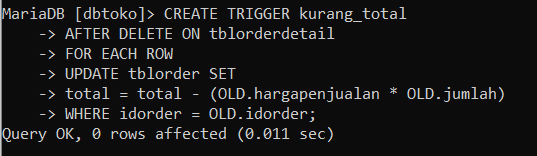
Jadi, nilai dari total pada tblorder akan diupdate dan diatur nilainya dengan operasi penjumlahan dari total yang ditambah dengan (hargapenjualan \* jumlah) saat AFTER INSERT pada tblorderdetail. NEW.hargapenjualan adalah nilai harga barang yang akan dijual dan NEW.jumlah adalah nilai dari jumlah barang yang akan dibeli pada tblorderdetail sehingga akan menambah nilai dari total. Jadi setiap data akan diupdate nilai total nya setelah data dimasukkan ke dalam tblorderdetail, maka dari itu menggunakan kondisi AFTER INSERT ON, karena nilai total akan diperbarui setelah data – data dimasukkan, sehingga setelah data – data dimasukkan, maka operasi perhitungan nilai total akan dijalankan.

1. Membuat trigger tambah\_stok digunakan untuk menambah stokbarang pada tblbarang SEBELUM DELETE pada tblorderdetail.



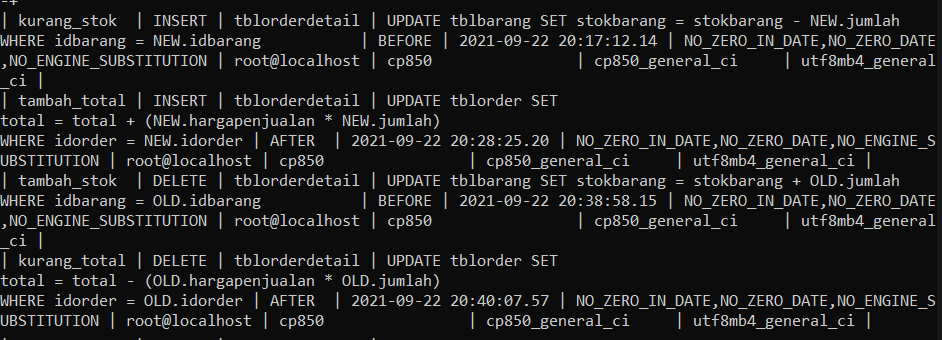
Jadi, sebelum data stokbarang dihapus atau dilakukan pengurangan, data stokbarang akan diupdate dan diatur operasinya yaitu stokbarang = stokbarang + OLD.jumlah. OLD.jumlah ini berisi jumlah barang lama yang sudah ada atau barang sisa yang belom terbeli pada tblorderdetail. Jadi, jumlah stokbarang akan bertambah ketika sebelum jumlah barang dihapuskan ke dalam tblorderdetail, karena untuk menjalankan operasi penambahan stok, dibutuhkan nilai stokbarang yang dikembalikan ke kolom stokbarang. Maka dari itu, kondisi yang digunakan adalah menggunakan BEFORE DELETE ON. Karena data dimasukkan / dieksekusi sebelum melakukan proses delete data pada tblorderdetail, sehingga setelah data dihapus, maka nilai stokbarang sudah ditambah.

1. Membuat trigger kurang\_total yang digunakan untuk mengurangi total pada tblorder SESUDAH DELETE pada tabel tblorderdetail.



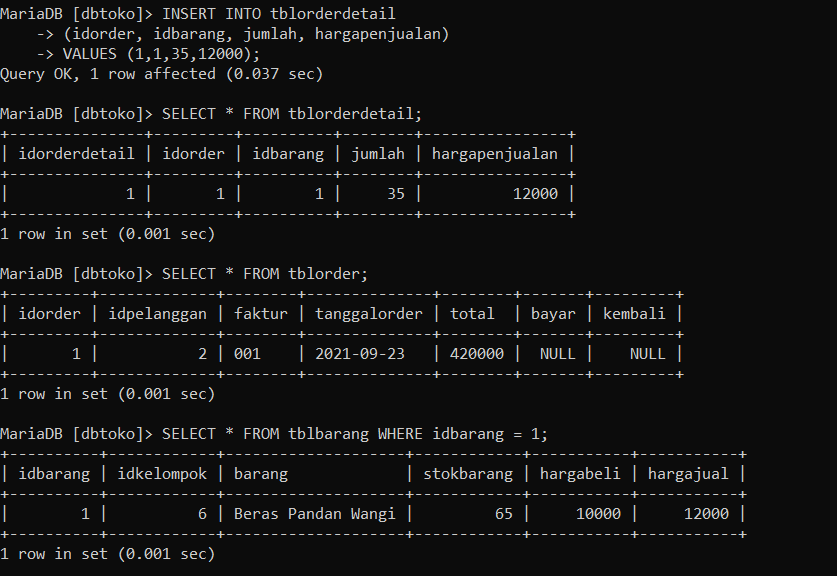
Jadi, setelah data total pada tblorderdetail dihapus, maka data total pada tblorder akan diupdate dengan operasi total = total – (OLD.hargapenjualan \* OLD.jumlah). Sehingga setiap data pada tblorder akan diupdate nilainya setelah nilai total dari tblorderdetail dihapus, maka dari itu menggunakan kondisi AFTER DELETE ON, karena data akan diperbarui setelah data total pada tblorderdetail dihapus.

1. SHOW TRIGGERS digunakan untuk menampilkan trigger yang sudah dibuat.



Terdapat 4 trigger yang sudah dibuat, kurang\_stok, tambah\_total, tambah\_stok, dan kurang\_total.

1. Uji coba berjalannya trigger yang dibuat.

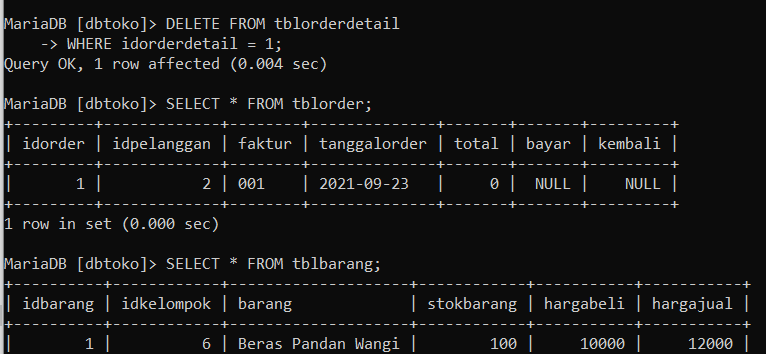


Pertama, dengan menginsertkan data ke dalam tblorderdetail, jumlah barang 35 dengan idbarang 1 dan idorder 1, dan hargapenjualan 12000.

Kedua, setelah data dimasukkan, maka nilai stokbarang pada tblbarang akan berkurang, karena stokbarang sudah dipakai sebanyak 35, jadi menjalankan trigger kurang\_stok.

Ketiga, nilai total pada tblorder juga akan bertambah yaitu hargajual barang dikalikan 35 jumlah barang yang di order, maka menjalankan trigger tambah\_total.

1. Melakukan delete data pada tabell orderdetail dengan menggunakan perintah DELETE FROM tblorderdetail WHERE idorderdetail = 1;

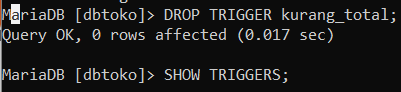


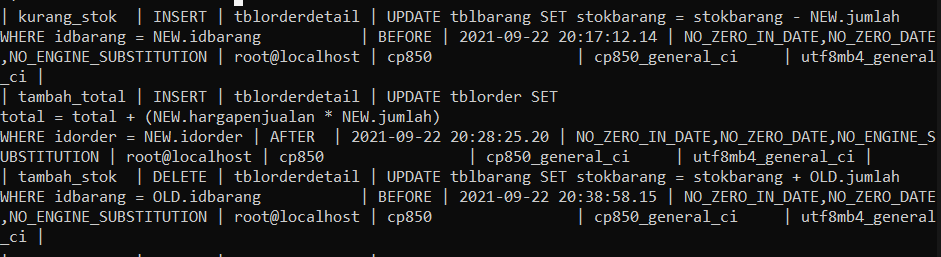
Setelah melakukan perintah DELETE data pada idorderdetail 1, selanjutnya lakukan pengecekan untuk kolom data yang berhubungan dengan idorderdetail, yaitu di dalam tblorder dan di dalam tblbarang.

Pada tblorder kolom total, sebelum data dihapus, kolom total mempunyai nilai 420000, tetapi setelah data pada tblorderdetail dihapus, maka nilai total menjadi 0. Ini menjalankan trigger kurang\_total, karena setelahh data pada tblorderdetail dihapus, maka nilai data pada total akan dikurangi.

Kedua, kolom stokbarang pada tblbarang, sebelum data dihapus, kolom stokbarang mempunyai data 65, setelah data pada tblorderdetail dihapus, kolom total mendapatkan stok kembali yaitu 100. Ini menjalankan trigger tambah\_stok, data stokbarang bertambah karena terdaspat 35 stokbarang yang dimasukkan.

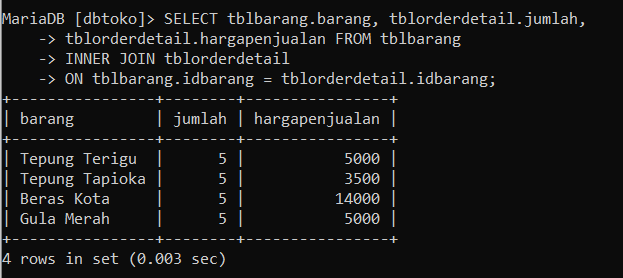
1. DROP TRIGGER adalah perintah yang digunakan untuk menghapus trigger.





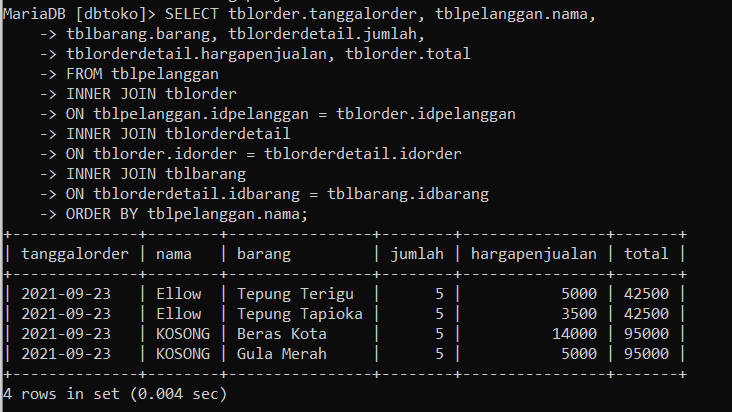
Setelah menghapus trigger kurang\_total, tampilkan kembali trigger yang ada dengan menggunakan perintah SHOW TRIGGER, hanya terdapat 3 trigger, yaitu kurang\_stok, tambah\_total, dan tambah\_stok.

1. JOIN TABEL digunakan untuk memudahkan pembacaan pada tabel.
2. INNER JOIN digunakan untuk menampilkan semua data yang mempunyai nilai sama.



Terdapat perintah SELECT untuk menampilkan data kolom barang pada tblbarang, data kolom jumlah pada tblorderdetail, dan data harga penjualan pada tblorderdetail dari tblbarang dimana tabel yang digabungkan adalah tblorderdetail dan data yang akan ditampilkan adalah data yang sama dan berhubungan antara idbarang pada tblbarang dan idbarang pada tblorderdetail.

1. ON pada inner join digunakan untuk menampilkan data yang sama dan berhubungan antara table master dan tabel transaksi.
2. INNER JOIN pada banyak tabel



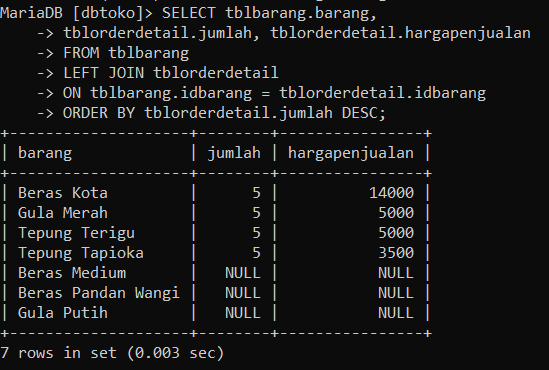
Perintah SELECT untuk menampilkan data tanggalorder pada tblorder, data nama pada tabel pelanggan, data barang pada tblbarang, data jumlah pada tblorderdetail, data hargapenjualan pada tblorderdetail, dan data total pada tblorder, dari tblpelanggan yang digabungkan dengan tblorder, dan data yang akan ditampilkan adalah data yang sama dan berhubungan antara idpelanggan pada tblpelanggan dan idpelanggan pada tblorder.

Menggabungkan tblorderdetail dan tblorder, dan data yang akan ditampilkan adalah data yang sama dan berhubungan antara idorder pada tblorder dan idorder pada tblorderdetail.

Menggabungkan tblbarang dan tblorderdetail, dan data yang akan ditampilkan adalah data yang sama dan berhubungan antara idbarang pada tblbarang dan idbarang pada tblorderdetail.

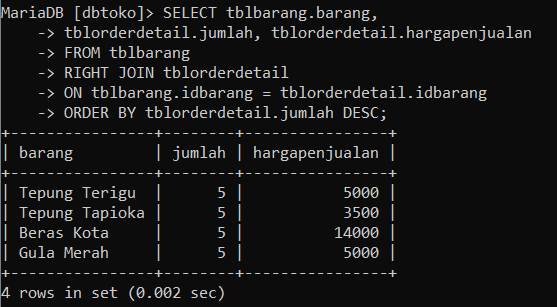
Yang diurutkan oleh kolom nama pada tblpelanggan.

1. LEFT JOIN digunakan untuk menampilkan semua yang ada di tabel master dan mengambil sebagian yang ada di tabel transaksi, baik nilainya berisi atau null.



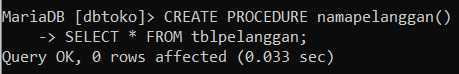
Menampilkan barang yang ada pada tblbarang, jumlah yang ada pada tblorderdetail, dan hargapenjualan yang ada pada tblorderdetail dari tblbarang dan menggabungkan data pada tblorderdetail menggunakan perintah LEFT JOIN dimana data yang akan ditampilkan adalah data yang sama dan berhubungan antara idbarang pada tblbarang dan idbarang pada tblorderdetail.

1. RIGHT JOIN digunakan untuk menampilkan semua yang ada di tabel transaksi dan sebagian yang ada di tabel master, hanya menampilkan nilai data yang berisi, tidak menampilkan nilai data null.



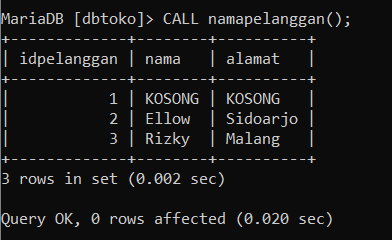
Menampilkan barang yang ada pada tblbarang, jumlah yang ada pada tblorderdetail, dan hargapenjualan yang ada pada tblorderdetail dari tblbarang dan menggabungkan data pada tblorderdetail menggunakan perintah RIGHT JOIN dimana data yang akan ditampilkan adalah data yang sama dan berhubungan antara idbarang pada tblbarang dan idbarang pada tblorderdetail. Data yang ditampilkan akan diurutkan sesuai dengan jumlah pada tblorderdetail dengan urutan desc, dari nilai terbesar ke nilai terkecil.

1. STORE PROCEDURE adalah blok program yang diletakkan pada MySQL, dimana blok yang sudah dibuat bisa dipanggil jika diperlukan dan procedure tidak memiliki return.
2. CREATE PROCEDURE digunakan untuk membuat procedure.



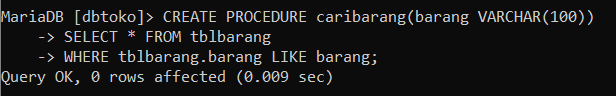
Membuat procedure dengan nama namapelanggan(), dimana procedure tersebut mengambil semua data yang ada pada tblpelanggan.

1. CALL PROCEDURE digunakan untuk memanggil procedure.



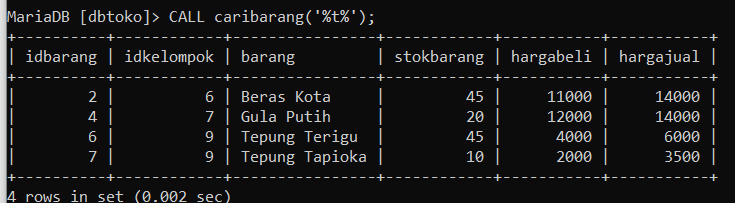
Menampilkan procedure yang mempunyai nama namapelanggan().

1. CREATE PROCEDURE dengan menggunakan parameter.



Membuat procedure dengan nama caribarang dan paremeter berisi barang dengan tipe data varchar. Dimana data diambil dari tblbarang yang memilih data dari barang pada tblbarang.

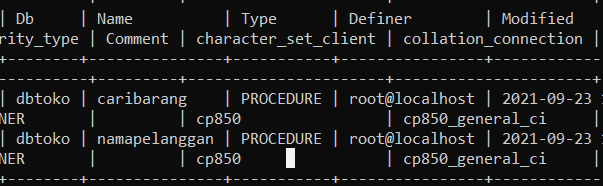
1. Memanggil procedure dengan mengisikan nilai parameternya.



Memanggil procedure caribarang() dengan data yang sesuai pada parameter, yaitu awalan dan

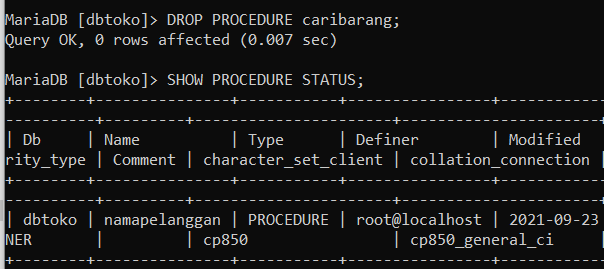
akhiran sembarang, tetapi terdapat huruf t di tengah tengah barang.

1. SHOW PROCEDURE STATUS digunakan untuk menampilkan semua isi procedure



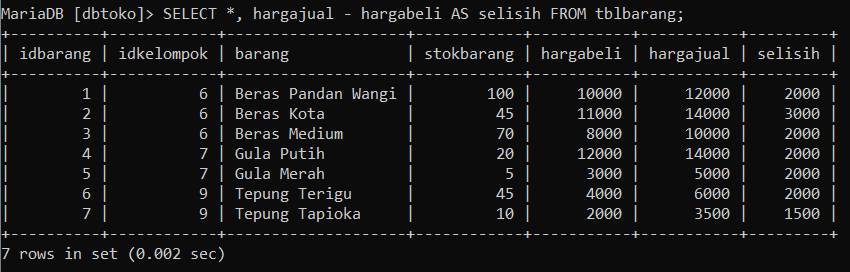
Terdapat 2 procedure, yaitu procedure dengan nama caribarang dan namapelanggan.

1. DROP PROCEDURE digunakan untuk menghapus store procedure.

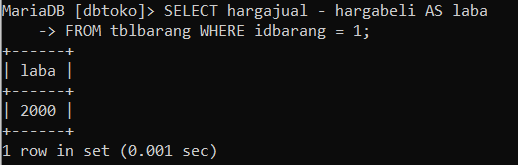


Menghapus procedure caribarang menggunakan perintah DROP PROCEDURE, setelah itu mengecek kembali apakah procedure berhasil dihapus atau tidak.

1. FUNCTION adalah blok program yang disimpan di mysql yang bisa menerima input atau parameter dan memiliki return.
2. Membuat tabel baru dengan nama selisi, dimana selisih berisi nilai dari hargajual – hargabeli.

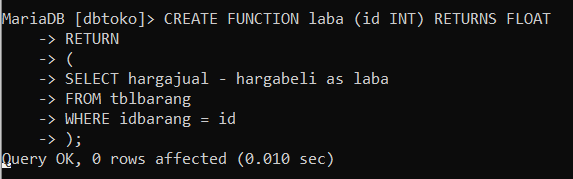


1. Mengambil isi selisih setiap barang sebagai kolom laba dengan mengambil idbarang = 1;



Menampilkan kolom laba dari tblbarang yang akan menampilkan data laba idbarang = 1, dimana kolom ini dibuat dari hasi pengurangan antara hargajual dan hargabeli.

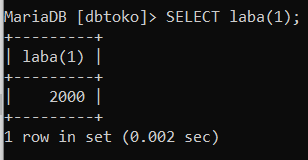
1. CREATE FUNCTION digunakan untuk membuat function.



Membuat function bernama laba dengan perintah CREATE FUNCTION dan parameter function berisi id dengan tipe data int, dimana nilai akan dikembalikan dalam bentuk float.

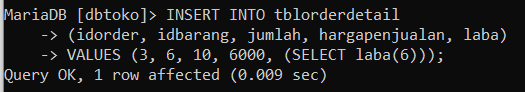
Function laba berisi menampilkan data selisih antara hargajual – hargabeli sebagai laba, dimana data diambil dari tblbarang, dengan nilai idbarang sama dengan id yang ada di parameternya.

1. SELECT FUNCTION digunakan untuk menampilkan function.



Data yang ditampilkan adalah selisih data dengan idbarang 1, yaitu bernilai 2000.

1. Menggunakan function laba() ketika meng insertkan data di dalam tblorderdetail



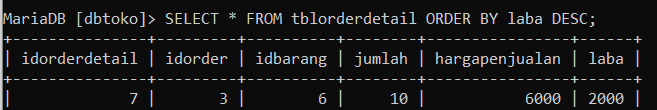
Memasukkan data ke dalam tblorderdetail, dengan memasukkan data pada kolom idorder, idbarang, jumlah, hargapenjualan, dan laba. Dan mengatur values dari kolom idorder 3, idbarang 6, jumlah 10, hargapenjualan 6000, dan values untuk laba menampilkan perintah select function laba dengan idbarang 6.

1. Memeriksa apakah operasi function sudah berjalan dengan benar atau tidak

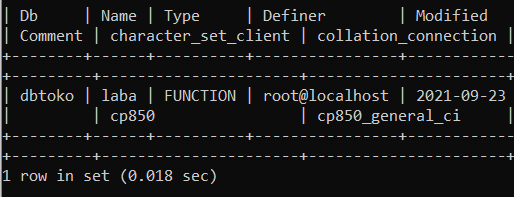
Idbarang 6 mempunyai hargajual 6000 dan hargabeli 4000.



Pemeriksaan laba dengan idbarang 6, dengan nilai laba tertulis 2000. Hasil dari laba sudah benar, karena nilai hargajual 6000 – hargabeli 4000, hasilnya 2000.

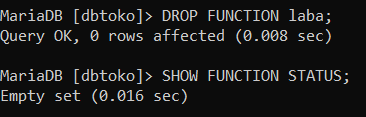


1. SHOW FUNCTION STATUS digunakan untuk menampilkan function – function yang sudah dibuat.



Terdapat 1 function yang sudah dibuat, yaitu function laba.

1. DROP FUNCTION digunakan utnuk menghapus function.

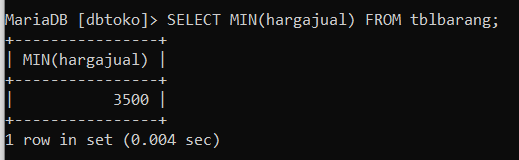


Terdapat perintah untuk menghapus function laba dengan perintah DROP FUNCTION. Setelah function dihapus, cek kembali apakah function berhasil dihapus atau tidak menggunakan function SHOW FUNCTION.

1. SELECT AGGREGATE adalah SELECT yang menampilkan function bawaan dari mysql. Beberapa select aggregate :

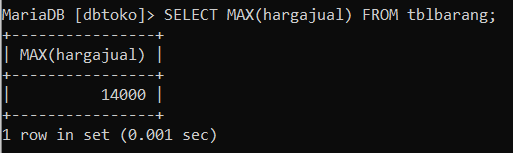
* MIN (menampilkan nilai terkecil pada kolom yang dimaksud).
* MAX (menampilkan nilai terbesar pada kolom yang dimaksud).
* SUM (menampilkan nilai penjumlahan pada kolom yang dimaksud).
* AVG (menampilkan nilai rata – rata pada kolom yang dimaksud).
* COUNT (menampilkan jumlah baris pada kolom yang dimaksud).
* COUNT(\*) (menampilkan jumlah baris pada tabel).

1. Menampilkan data nilai MIN (nilai terkecil).



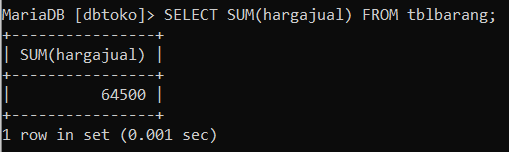
Terdapat perintah untuk menampilkan data dengan nilai minimal pada kolom hargajual yang ada di tblbarang. Selanjutnya data yang ditampilkan adalah 1 data, yaitu dengan nilai minimal adalah 3500.

1. Menampilkan data nilai MAX (nilai terbesar).



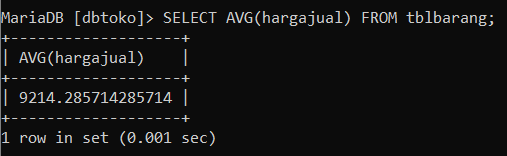
Terdapat perintah untuk menampilkan data dengan nilai maksimal pada kolom hargajual yang ada di tblbarang. Selanjutnya, data yang ditampilkan adalah 1 data, yaitu dengan nilai maksimal adalah 14000.

1. Menampilkan data nilai SUM (nilai hasil penjumlahan pada kolom yang dimaksud).



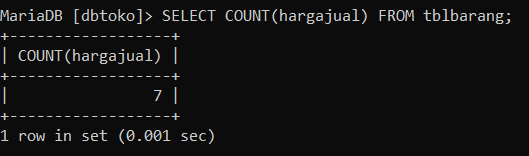
Terdapat perintah untuk menampilkan data dengan nilai hasil penjumlahan pada kolom hargajual yang ada di tblbarang. Kemudian, data yang ditampilkan ada 1 data, yaitu dengan nilai gabungan penjumlahan adalah 64500.

1. Menampilkan data nilai AVG (nilai rata – rata pada kolom yang dimaksud).



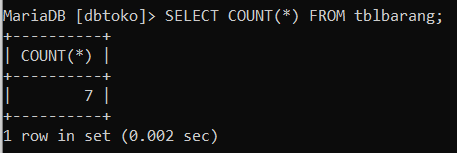
Terdapat perintah untuk menampilkan data dengan nilai rata – rata pada kolom hargajual yang ada di tblbarang. Kemudian, data yang ditampilkan ada 1 data, yaitu dengan nilai rata – rata yaitu 9214.285.

1. Menampilkan data nilai COUNT (jumlah baris pada kolom yang dimaksud).



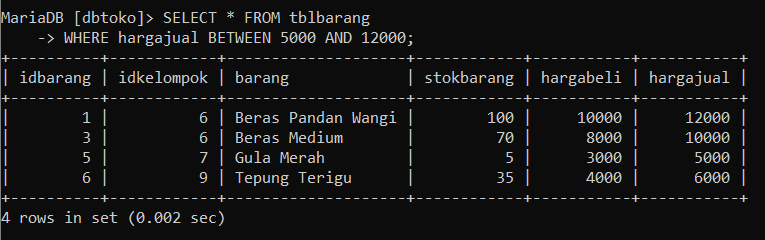
Terdapat perintah untuk nenampilkan data dengan jumlah baris yang ada pada kolom hargajual yang ada di tblbarang. Kemudian, data yang ditampilkan ada 1 data, yaitu dengan jumlah total baris yaitu 7.

1. Menampilkan data nilai COUNT\* (jumlah baris pada tabel yang dimaksud).



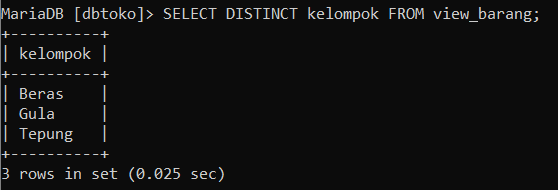
Terdapat perintah untuk menampilkan data dengan jumlah baris pada tabel tblbarang. Kemudian, data yang ditampilkan ada 1 data, yaitu dengan jumlah total baris yaitu 7.

1. SELECT BETWEEN (SELECT ANTARA DUA NILAI) adalah select yang digunakan untuk menampilkan data antara dua nilai terendah dan tertinggi.



Menampilkan data dengan kondisi pemilihan yaitu memilih kolom hargajual dengan nilai antara 5000 dan 12000. Data yang ditampilkan terdapat 4 data, yaitu dengan idbarang 1, 3, 5, dan 6.

1. SELECT DISTINCT digunakan untuk menampilkan data yang sama satu kali.

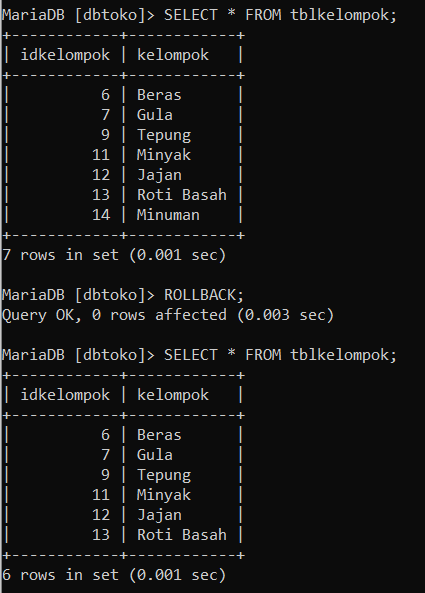


Menampilkan data yang sama satu kali dari kolom kelompok yang ada pada view\_barang, data yang ditampillkan terdapat 3 data, yaitu Beras, Gula, dan Tepung.

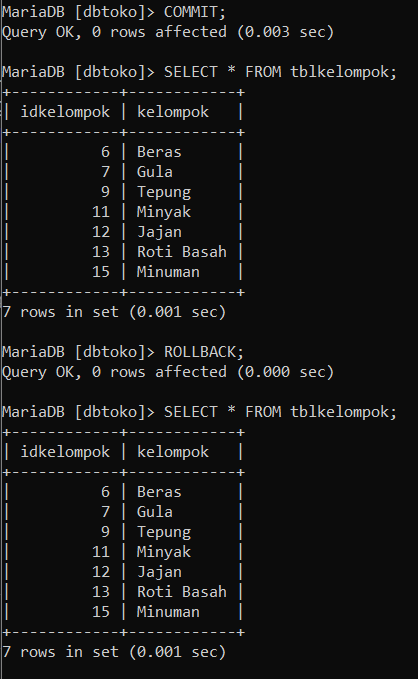
1. Transaksi adalah perintah INSERT, UPDATE, SELECT, DELETE.
2. START TRANSACTION adalah perintah yang digunakan untuk memulai transaksi.



1. ROLLBACK adalah mengembalikan keadaan database ke keadaan sebelumnya.
2. Untuk menjalankan perintah rollback cukup dengan menuliskan ROLLBACK saja setelah transaksi dimulai atau dinyalakan.
3. COMMIT digunakan untuk mengakhiri semua transaksi dan menjadikannya sebagai perubahan permanen.
4. Menambahkan data minuman pada tblkelompok, setelah itu melakukan rollback, hasilnya data yang baru ditambahkan hilang, karena rollback berfungsi untuk mengembalikan keadaan database ke keadaan sebelumnya.



1. Transaksi yang bisa di rollback (undo) adallah transaksi yang dimulai dengan START TRANSACTION dan belum diakhiri dengan commit.
2. Transaksi yang sudah di commit tidak bisa dirollback.
3. Menambahkan data Minuman kembali ke dalam tblkelompok, setelah itu melakukan commit, lalu cek datanya, kemudian lakukan rollback kembali. Hasilnya adalah data tidak terdapat perubahan karena data yang sudah dicommit tidak bisa di rollback.



**Saya Belum Mengerti**