**Operator IN dan NOT IN**

untuk menampilkan data yang kriterianya ada pada salah satu nilai didalam IN / NOT IN

**struktur :**

SELECT namafield1, namafield2, .....

FROM namatabel

WHERE namafield IN/NOT IN (nilai1, nilai2, ....);

Soal :

1. Tampilkan nama barang dengan kode B001, B002 dan B005

SELECT \* FROM tbrg

WHERE kode = 'B001' OR kode = 'B002' OR kode = 'B005';

SELECT \* FROM tbrg

WHERE kdbrg IN('B001', 'B002','B005');

1. Tampilkan data mahasiswa yang tidak mendapatkan nilai A atau B.

SELECT \* FROM nilai

WHERE nilai NOT IN ('A','B');

1. Tampilkan data mahasiswa yang mendapatkan nilai A dan B pada kode kuliah TI111 dan TI123

**Operator LIKE dan NOT LIKE**

menampilkan data dengan nilai yang sesuai kriteria tertentu untuk data dengan tipe selain numerik.

**struktur :**

SELECT namafield2, namafield2, ....

FROM namatabel

WHERE namafield LIKE/NOT LIKE '....';

Kriteria --> berupa karakter

ketentuan:

\_ => mewakili 1 karakter

% => mewakili sembarang karakter (0 - dst)

**Contoh :**

A\_ --> 2 karaker, yang diawali dengan huruf A : Ab, Ac, Ad,

A% --> diawali dengan huruf A, dengan jumlah karakter tak hingga. A, Aku, Angga, Ali Syahbana.

%komputer% --> akan mencari data yang mengandung kata-kata. 'komputer'.

A\_u --> Abu, Adu, Aku

R%o --> Ridho, Rimo, Redho

**Contoh :**

1. Tampilkan nama mahasiswa yang diawali dengan huruf A

SELECT nama FROM t\_mahasiswa

WHERE nama LIKE 'A%';

1. Tampilkan nama dosen yang tidak berakhiran a.
2. Pada database classicmodels tampilkan employeenumber, lastname dan first name untuk employees dengan job title yang ada kata 'sales man'
3. Tampilkan identitas mahasiswa yang namanya diawali huruf A tetapi selain Albinanto.
4. Tampilkan nama dan nobp mahasiswa yang karakter ke 3 dan 4 nobp adalah 11.

SELECT nama, no\_bp from t\_mahasiswa

WHERE no\_bp LIKE '\_\_11%';

**Operator BETWEEN dan NOT BETWEEN**

menampilkan kondisi yang memiliki nilai yang ada dalam range tertentu.

Struktur :

SELECT namafield1, namafield2, ....

FROM namatabel

WHERE namafield BETWEEN / NOT BETWEEN nilai1 AND nilai2;

**Catatan :**

- ini hanya untuk field yang memiliki range tertentu

- jika karakter, tetap diapit oleh tanda kutip ('...')

**Contoh :**

1. Tampilkan data mahasiswa yang lahir di tahun 1983.

=> 1983 : 1 Januari 1983 sampai 31 Desember 1983

SELECT \* FROM t\_mahasiswa

WHERE tgl\_lahir BETWEEN '1983-01-01' AND '1983-12-31';

atau :

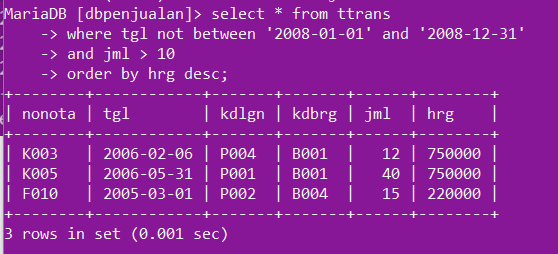
SELECT \* FROM t\_mahasiswa

WHERE tgl\_lahir BETWEEN '19830101' AND '19831231';

1. Tampilkan data transaksi yang terjadi diluar tahun 2008, urutkan berdasarkan tanggal paling akhir.

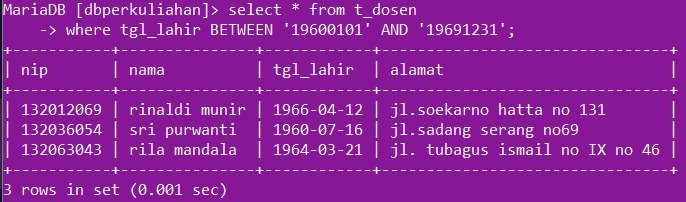
Soal dbjual

1. Tampilkan data transaksi yang terjadi selain Bulan September 2008, tetapi yang jumlah penjualannya diatas 10 buah. Urutkan berdasarkan harga tertinggi

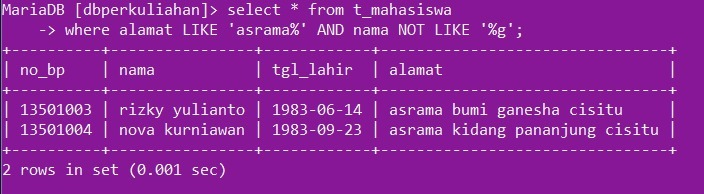


Soal dbperkuliahan :

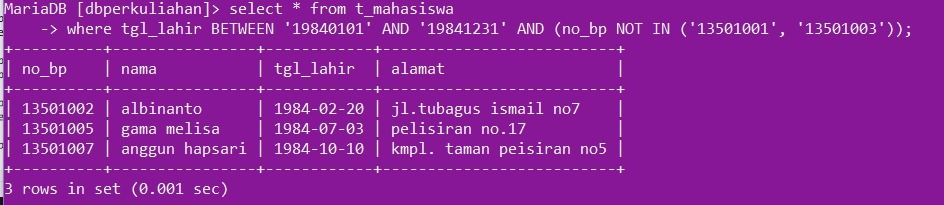
1. Siapa saja dosen yang lahir pada tahun 1960-an?? Tampilkan identitas semua dosen



1. Siapa saja mahasiswa yang tinggal di asrama yang namanya tidak berawalan G



1. Tampilkan mahasiswa yang lahir tahun 1984 dan selain yang memiliki nomor bp 13501001 dan 13501003



soal classicmodels:

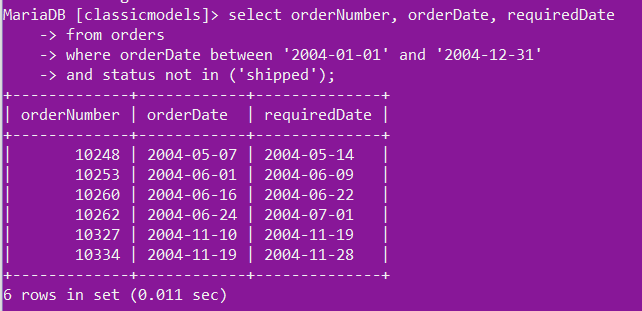
1. Tampilkan officecode, city, phone dan country yang hanya di USA dan France saja



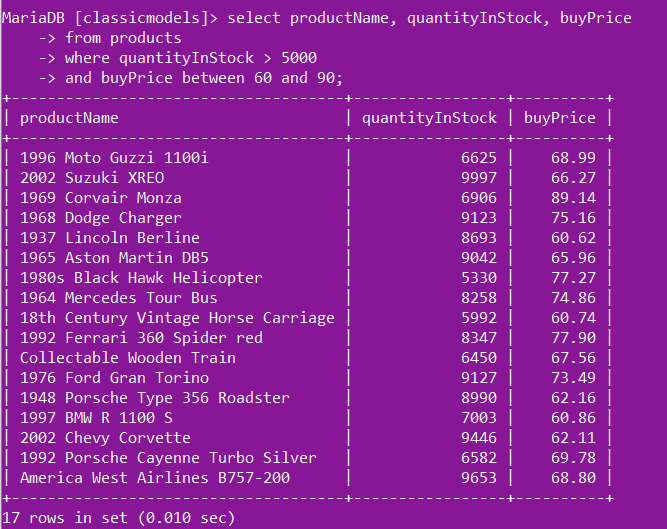
1. Tampilkan customername,country dan state untuk seluruh customer yang tinggal di state CA country USA



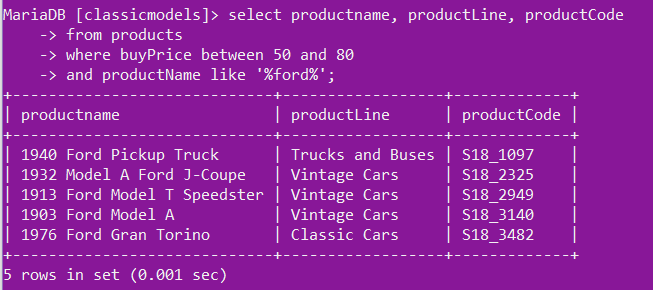
1. Tampilkan seluruh order yang terdiri dari ordernumber,orderdate, requireddate pada tabel orders untuk seluruh order yang terjadi pada tahun 2004 dan statusnya bukan shipped



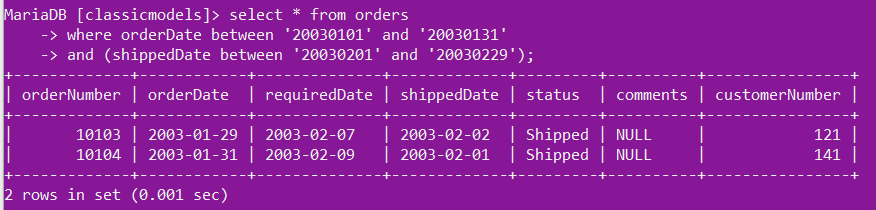
1. Tampilkan nama produk, stok, harga pembelian (buyprice) pada tabel produk untuk produk dengan stok diatas 5000 dan harga pembelian antara 60 - 90



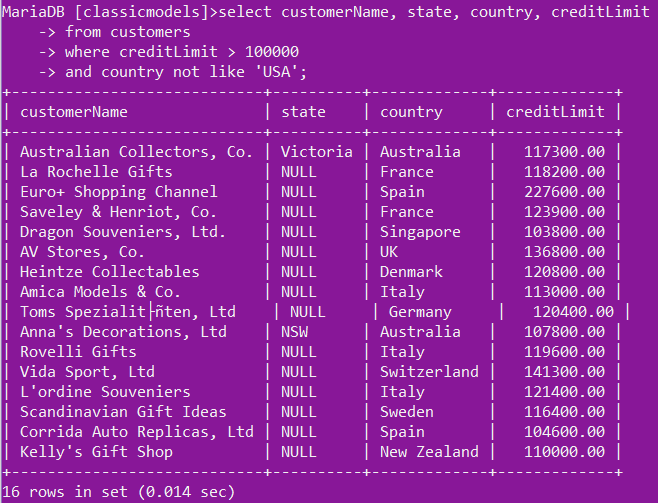
1. Tampilkan nama produk, harga produk dan kode produk untuk produk dengan harga pembelian 50-80 untuk barang yang memiliki nama ford



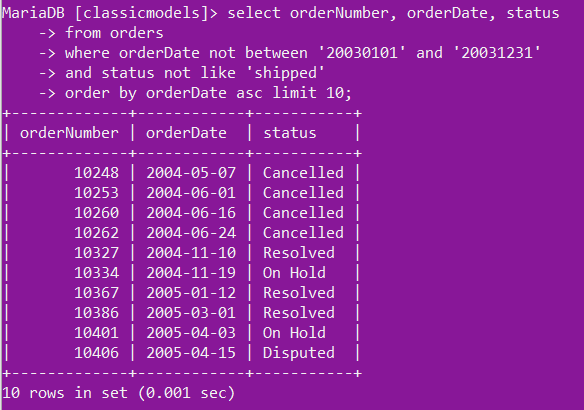
1. Tampilkan data order yang terjadi pada Bulan Januari tahun 2003 dan barangnya dikirim pada bulan Februari 2003



1. Tampilkan customername, state, country, creditlimit untuk customer yang limitkreditnya diatas 100000 tetapi tidak dari USA



1. Tampilkan 10 data orders berupa ordernumber, orderdate dan status untuk tanggal order yang terjadi diluar tahun 2003 dan statusnya bukan shipped. Urutkan berdasarkan tgl order.



**Operator IS dan IS NOT NULL**

operator yang digunakan untuk menyaring data yang kosong (nilainya NULL)

**Contoh :**

dbjual --> tpelanggan

Inputkan 2 buah data pelanggan

INSERT INTO tpelanggan(kdlgn,nmlgn,kota) VALUES

('P010','Rudi','Solok'),

('P015','Citra','Padang');

Kalau ingin menampilkan data-data pelanggan yang tidak memiliki kode pos

Kalau dikerjakan :

SELECT \* FROM tpelanggan WHERE kdpos='NULL';

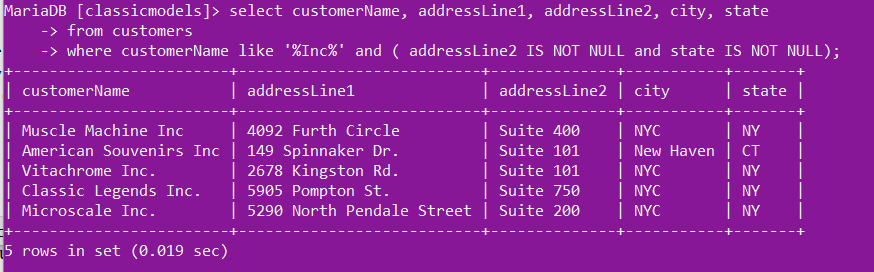
=> empty karena tidak ada yang kodepos bernilai NULL (artinya memang nilai kode pos yang dicari NULL)

SELECT \* FROM tpelanggan WHERE kdpos IS NULL;

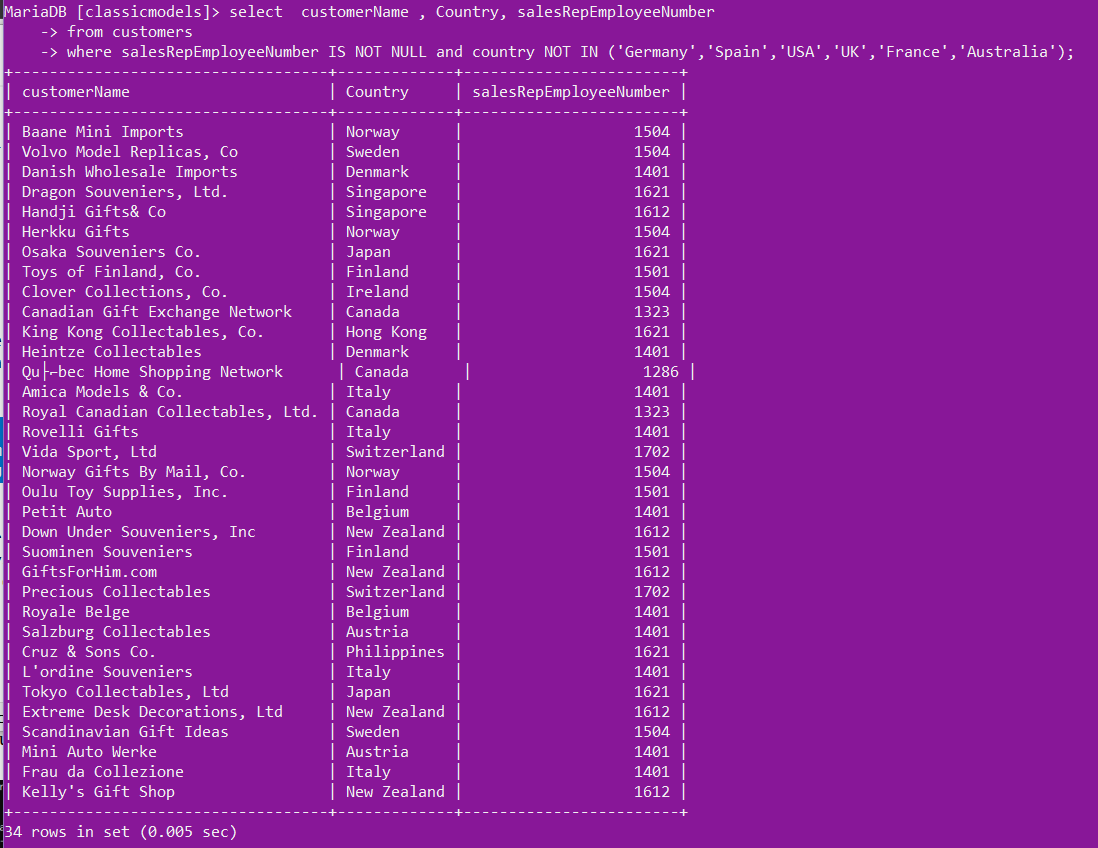
**Contoh :**

classicmodels

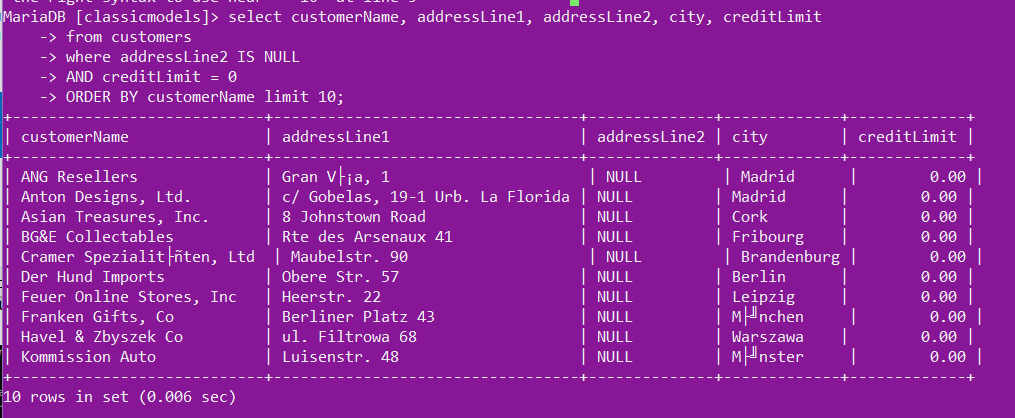
1. Tampilkan data customers yang terdiri dari customername, addressline1, addressline2,city, state yang nama customernya ada unsur Inc, yang addressline2 dan state tidak kosong. --> 5 records



2. Tampilkan data customer terdiri dari nama, negara dan salesrepemployeenumber untuk customer yang salesrepemployeenumber nya tidak kosong dan tdk berasal dari negara Jerman, Spanyol, USA, UK, Perancis dan Australia --> 34



3. Tampilkan 10 data customers (terdiri dari customername, addressline1, addressline2, city dan creditlimit), untuk customer yang tidak punya alamat alternatif (addressline 2 kosong) dan creditlimitnya 0. Tampilkan terurut berdasarkan nama --> 10



**FUNGSI**

1. Fungsi yang sudah merupakan bawaan dari program sehingga tidak perlu didefinisikan tinggal dipergunakan
2. Fungsi yang harus didefinisikan dulu baru bisa digunakan.

Beberapa fungsi bawaan SQL :

- fungsi numerik

- fungsi string

- fungsi date/time

- fungsi kontrol (control flow function) -- IF dan CASE

- fungsi aggregate

**FUNGSI NUMERIK**

=> fungsi bawaan MySQL yang digunakan untuk mengolah data numerik (int, decimal, double, float)

1. CEIL() atau CEILING()

=> digunakan untuk melakukan pembulatan keatas

**struktur :**

SELECT CEIL/CEILING(argumen);

argumen --> berupa nilai numerik atau bisa berupa field dengan tipe numerik

**contoh :**

SELECT CEIL(6.6); --> 7

SELECT CEIL(6.1); --> 7

SELECT CEILING(6.001); -->7

1. FLOOR()=> digunakan untuk pembulatan kebawah

Struktur :

SELECT FLOOR(argumen);

Contoh :

SELECT FLOOR(6.9); --> 6

SELECT FLOOR(6.3); --> 6

SELECT FLOOR(6.99); -->6

3. ROUND()

--> pembulatan matematis

--> digunakan untuk pembulatan keatas atau kebawah tergantung pada nila pecahan. Jika kurang dari 0.5 maka dibulatkan kebawah, jika besar atau sama dngan 0.5 dilakukan pembulatan keatas

Untuk fungsi round --> bisa dimanfaatkan dengan 1 argumen dan 2 argumen. Berbeda dengan fungsi floor dan ceil yang hanya butuh 1 argumen

Struktur :

SELECT ROUND(argumen1,argumen2);

argumen1 : angka pecahan yang akan dibulatkan

argumen2 : jumlah digit desimal yang diinginkan.

Jika argumen2 tidak ditulis (cuma ada 1 argumen dalam fungsi round), berarti tidak ada angka dibelakang koma

Contoh :

Fungsi round dengan 1 argumen

SELECT ROUND(5.9); --> 6

SELECT ROUND(5.4); --> 5

Fungsi round dengan 2 argumen

SELECT ROUND(2.673,1); --> 2.7

SELECT ROUND(2.648,1); --> 2.6

SELECT ROUND(2.648,2); --> 2.65

SELECT ROUND(3.5,3); --> 3.500

4. TRUNCATE()

kalau truncate table digunakan utk ??

truncate table tbrg; --> mengosongkan isi seluruh tabel tanpa mengubah struktur

--> fungsi truncate() digunakan untuk memotong angka desimal sesuai dengan jumlah angka di argumen kedua tanpa melakukan pembulatan keatas maupun kebawah.

Struktur :

SELECT TRUNCATE(argumen1, argumen2);

argumen 1 : nilai desimal yang akan dipotong (field numerik)

argumen 2 : banyak angka dibelakang koma

SELECT TRUNCATE(2.673,1); --> 2.6

SELECT TRUNCATE(2.648,1); --> 2.6

SELECT TRUNCATE(2.648,2); --> 2.64

SELECT TRUNCATE(2.648,0); --> 2

5. DIV

--> menampilkan hasil bagi bulat

Contoh :

SELECT 10 DIV 3; --> hasilnya 3

SELECT 3 DIV 10; --> hasilnya 0

6. MOD

--> menampilkan sisabagi

Contoh :

SELECT 10 MOD 3; --> 1

SELECT 3 MOD 10; --> 3

SELECT 10 % 3;

SELECT 10 MOD 3; --> bisa ditulis SELECT MOD(10,3);

7. SQRT

--> digunakan untuk mencari akar kuadrat.

--> Fungsi ini hanya membutuhkan sebuah argumen berupa angka yang akan dicari nilai kuadratnya

Contoh :

SELECT SQRT(81); --> 81

SELECT ROUND(SQRT(20),2);

8. POW() atau POWER()

--> digunakan untuk mencari nilai pangkat.

--> FUngsi ini butuh 2 argumen berupa bilangan dasar dan bilangan pangkat

Contoh :

SELECT POW(5,3); --> 5 pangkat 3 --125

SELECT ROUND(POW(2.5,5),1); --> 2.5 pangkat 3

9. ABS()

--> digunakan untuk menampilkan nilai absolut/mutlak.

--> fungsi ini hanya butuh 1 argumen. Jika argumen berupa angka positif maka akan langsung dikembalikan. Jika argumen bernilai negatif, maka akan diubah menjadi positif

COntoh :

SELECT ABS(10); -- hasilnya 10

SELECT ABS(-10); -- hasilnya 10

===============================================================

TUGAS

DBJUAL (3 SOAL)

1 .Semua barang mendapatkan diskon sebesar 5% dari total harga(jml\*hrg). Tampilkan nonota,kdbrg, totalharga sebelum diskon, diskon dan total harga setelah diskon. Semua perhitungan gunakan pembulatan keatas

SELECT nonota,kdbrg, CEIL(jml\*hrg) as total,

CEIL(0.05\*jml\*hrg) as diskon,

CEIL((jml\*hrg)-(0.05\*jml\*hrg)) as tot\_bayar

FROM ttrans;

atau

SELECT nonota,kdbrg, CEIL(jml\*hrg) as total,

CEIL(0.05\*jml\*hrg) as diskon,

CEIL(0.95\*jml\*hrg) as tot\_bayar

FROM ttrans;

2. Untuk total pembelian yang lebih dari Rp.1.000.000,- akan mendapatkan diskon sebesar 5%. Tampilkan nonota, kdbrg, total harga sebelum diskon, diskon, total harga setelah diskon.(pembulatan kebawah)

3. untuk pelanggan dengan kode P002 diharuskan membayar pajak barang sebanyak 10% dari total harga. Tapi hanya barang yang dibeli ditahun 2005 dan barang yang harganya diatas Rp.500.000,-. tampilkan data transaksi lengkap dengan besaran pajaknya.(lakukan pembulatan)

---------------------------

CLASSICMODELS (7 SOAL)

1. Untuk pelanggan yang melakukan pembayaran pada bulan januari 2003 dari tabel payments, wajib membayar pajak sebanyak 10% dari amount. Tampilkan semua field dan besaran pajak lakukan pembulatan keatas dan urutkan berdasarkan paymentdate.

2. Carilah berapa total harga yang didapatkan Untuk penjualan product yang bernama Harley dari tabel products, tetapi hanya untuk buyprice yang nilainya dibawah 50. Lakukan pembulatan.

3. Pada tabel orders terdapat orderdate dan shippeddate. Tampilkan lamanya waktu tunggu dari order hingga pengiriman, untuk status yang barangnya telah dikirim (shipped) dan terjadi pada tahun 2004, urutkan dari bulan order desember.

4. MSRP adalah manufacturer suggested retail price atau harga eceran yang disarankan. Total harga masing2 product adalah quantity instock \* msrp. Nilai keuntungan bisa didapatkan dari msrp dikurangi dengan buy price. Kira-kira berapa keuntungan yang didapatkan dari penjualan masing-masing product. Tampilkan msrp, quantityinstock, total, keuntungan, productvendor untuk product yang diproduksi oleh Min Lin Diecast. Lakukan pembulatan kebawah.

5. Profit diperoleh dari selisih msrp dan buyprice. Tampilkan 10 data profit tertinggi untuk produk dengan jenis(productline) Classic cars. Tampilkan productname, productline, msrp, buyprice dan profit. Lakukan pembulatan matematis (round) utk msrp, buyprice dan profit

6. Totalprice adalah hasil kali dari Quantityordered dengan priceeach pada tabel orderdetails. Diskon diberikan 10% dari totalprice. Tampilkan ordernumber,productcode, quantityordered, priceeach, total, diskon,payment (total-diskon) untuk 10 data dengan payment tertinggi yang quantityorderednya diantara 20 sampai 30 dan nomor order nya bukan 10420, 10423, 10425 . Lakukan pembulatan keatas untuk priceeach dan payment

7. Untuk pelanggan yang melakukan pembayaran pada bulan januari 2003 dari tabel payments, wajib membayar pajak sebanyak 10% dari amount. Tampilkan semua field dan besaran pajak lakukan pembulatan keatas dan urutkan berdasarkan paymentdate.

==================================================