```
--By J Kusnendar
--Membuat database
CREATE DATABASE inventory
ON
( NAME = inventory_data,
  FILENAME = 'c:\program files\microsoft sql
server\mssql\data\inventory_data.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5)
LOG ON
( NAME = 'inventory_log',
  FILENAME = 'c:\program files\microsoft sql
server\mssql\data\inventory_log.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB)

-- Menghapus database

drop database inventory

------------------------------------------------------------------------------
------------------
-- Membuat dan menghapus Tabel-tabel :

use inventory

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

-- 1. Tabel Supplier

create table Supplier
        (
        KodeSup Char(5) CHECK (KodeSup LIKE 'S[0-9][0-9][0-9][0-9]')not null,
        NamaSup Varchar(20) not null,
        AlamatSup Varchar(30) not null,
        TelpSup Varchar(15) check (TelpSup like
'[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
        KontakSup Varchar(20),
        constraint PKSupplier primary key(KodeSup)
        )

------------------------------------------------------------------------------
------------------
-- 2. Tabel Barang

create table Barang
        (
        KodeBrg Char(5) CHECK (KodeBrg LIKE 'B[0-9][0-9][0-9][0-9]')not null,
        NamaBrg Varchar(20) not null,
        SatuanBrg Varchar(10) not null,
        StokBrg Int not null,
        HargaBrg int not null,
        constraint PKBarang primary key(KodeBrg)
        )

------------------------------------------------------------------------------
------------------
-- 3. Tabel Pelanggan

create table Pelanggan
        (
        KodePlg Char(5) CHECK (KodePlg LIKE 'P[0-9][0-9][0-9][0-9]')not null,
        NamaPlg Varchar(20) not null,
        AlamatPlg Varchar(30) not null,
        KotaPlg Varchar(20),
        TelpPlg Varchar(15)check (TelpPlg like
```

```
'[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
        constraint PKPlg primary key(KodePlg)
        )

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

-- 4. Tabel TransaksiBarangMasuk (Pembelian)

create table TBMasuk
        (
        NoTBM Char(5) CHECK (NoTBM LIKE 'M[0-9][0-9][0-9][0-9]') not null,
        KodeSup Char(5) CHECK (KodeSup LIKE 'S[0-9][0-9][0-9][0-9]')not null,
        TglTransaksi datetime null,
        TglDatang datetime null,
        TotalHarga int not null,
        constraint PKTBM primary key(NoTBM),
        constraint FKTBM foreign key(KodeSup) references Supplier(KodeSup)
        )

-------------------------------------------------------------------------------
------------------
-- 5. Tabel RincianTransaksiBarangMasuk (RincianPembelian)

create table RincianTBMasuk
        (
        NoRincianTBM char(6) CHECK (NoRincianTBM LIKE
'RM[0-9][0-9][0-9][0-9]')not null,
        NoTBM Char(5) CHECK (NoTBM LIKE 'M[0-9][0-9][0-9][0-9]')not null,
        KodeBrg Char(5) CHECK (KodeBrg LIKE 'B[0-9][0-9][0-9][0-9]')not null,
        JumlahTBM Int not null,
        constraint PKRincianTBMasuk primary key(NoRincianTBM, NoTBM, KodeBrg),
        constraint FKRincianTBM1 foreign key(NoTBM) references TBMasuk(NoTBM),
        constraint FKRincianTBM2 foreign key(KodeBrg) references Barang(KodeBrg)
        )

-------------------------------------------------------------------------------
------------------
-- 6. Tabel TransaksiBarangKeluar (Penjualan)

create table TBKeluar
        (
        NoTBK Char(5) CHECK (NoTBK LIKE 'K[0-9][0-9][0-9][0-9]')not null,
        KodePlg Char(5) CHECK (KodePlg LIKE 'P[0-9][0-9][0-9][0-9]')not null,
        TglTransaksi Datetime null,
        TotalHarga int not null,
        constraint PKTBK primary key(NoTBK),
        constraint FKTBK foreign key(KodePlg) references Pelanggan(KodePlg)
        )

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

-- 7. Tabel RincianTransaksiBarangKeluar (RincianPenjualan)

create table RincianTBKeluar
        (
        NoRincianTBK char(6) CHECK (NoRincianTBK LIKE
'RK[0-9][0-9][0-9][0-9]')not null,
        NoTBK Char(5)CHECK (NoTBK LIKE 'K[0-9][0-9][0-9][0-9]') not null,
        KodeBrg Char(5) CHECK (KodeBrg LIKE 'B[0-9][0-9][0-9][0-9]')not null,
        JumlahTBK Int not null,
        constraint PKRincianTBKeluar primary key(NoRincianTBK, NoTBK, KodeBrg),
        constraint FKRincianTBK1 foreign key(NoTBK) references TBKeluar(NoTBK),
        constraint FKRincianTBK2 foreign key(KodeBrg) references Barang(KodeBrg)
        )

-------------------------------------------------------------------------------
------------------
```

DATA MANIPULATION LANGUAGE (DML)

Mengisi Tabel-Tabel (insert into)

```
-- 1. Tabel Supplier

insert into Supplier values
        ('S0001','Maju','Jl.Cancer 07','02270084322','Rio')
insert into Supplier values
        ('S0002','Makmur','Jl.Virgo 05','02270048233','Ana')
insert into Supplier values
        ('S0003','Bintang','Jl.Sagitarius 12','02270054321','Robert')
insert into Supplier values
        ('S0004','Top','Jl.Capricorn 02','02270045678','Toni')
insert into Supplier values
        ('S0005','Sentosa','Jl.Leo 08','70065433','Fitri')

select * from supplier

-- 2. Tabel Barang

insert into Barang values
        ('B0001','CPU','Biji','20',850000)
insert into Barang values
        ('B0002','Monitor','Biji','30',500000)
insert into Barang values
        ('B0003','Hardisk','Biji','50',400000)
insert into Barang values
        ('B0004','Keyboard','Biji','50',50000)
insert into Barang values
        ('B0005','Mouse','Biji','50',25000)

select * from Barang

-- 3. Tabel Pelanggan

insert into Pelanggan values
        ('P0001','Amir','Jl.Jeruk 10','Bandung','08133085001')
insert into Pelanggan values
        ('P0002','Eko','Jl.Durian 01','Cimahi','08563085445')
insert into Pelanggan values
        ('P0003','Rudi','Jl.Salak 212','Bandung','02230865001')
insert into Pelanggan values
        ('P0004','Selvi','Jl.Mangga 45','Cimahi','08179085234')
insert into Pelanggan values
        ('P0005','Tina','Jl.Anggur 98','Bandung','02253085022')

select * from Pelanggan

-- 4. Tabel TBMasuk (Tabel Pembelian)

insert into TBMasuk values
        ('M0001','S0001','11/08/2007','11/08/2007', 17000000)
insert into TBMasuk values
        ('M0002','S0002','11/9/2007','11/10/2007', 15000000)
insert into TBMasuk values
        ('M0003','S0003','12/9/2007','12/9/2007',20000000)
insert into TBMasuk values
        ('M0004','S0004','12/10/2007','12/11/2007',2500000)
insert into TBMasuk values
        ('M0005','S0005','12/12/2007','12/12/2007',1250000)

select * from TBMasuk
```

```
-- 5. Tabel RincianTBMasuk (Tabel RincianPembelian)

insert into RincianTBMasuk values
        ('RM0001','M0001','B0001','20')
insert into RincianTBMasuk values
        ('RM0002','M0001','B0002','30')
insert into RincianTBMasuk values
        ('RM0003','M0003','B0003','50')
insert into RincianTBMasuk values
        ('RM0004','M0004','B0004','50')
insert into RincianTBMasuk values
        ('RM0005','M0005','B0005','50')

select * from RincianTBMasuk


-- 6. Tabel TBKeluar (Tabel Penjualan)

insert into TBKeluar values
        ('K0001','P0001',11/9/2007,8500000)
insert into TBKeluar values
        ('K0002','P0002',11/10/2007,10000000)
insert into TBKeluar values
        ('K0003','P0003',12/11/2007,12000000)
insert into TBKeluar values
        ('K0004','P0004',12/12/2007,2000000)
insert into TBKeluar values
        ('K0005','P0005',12/15/2007,750000)

select * from TBKeluar

-- 7. Tabel RincianTBKeluar (Tabel RincianPenjualan)

insert into RincianTBKeluar values
        ('RK0001','K0001','B0001','10')
insert into RincianTBKeluar values
        ('RK0002','K0002','B0001','20')
insert into RincianTBKeluar values
        ('RK0003','K0003','B0001','30')
insert into RincianTBKeluar values
        ('RK0004','K0003','B0002','40')
insert into RincianTBKeluar values
        ('RK0005','K0005','B0001','30')

select * from RincianTBKeluar


--------------------------------------------------------------------------------
-------------------
-- Create RULE
--------------------------------------------------------------------------------
-------------------
create rule rulHarga
as @hargabrg > 0

sp_bindrule 'rulHarga', 'Barang.hargabrg'


create rule rulTotHarga
as @TotalHarga > 0

sp_bindrule 'RulTotHarga', 'TBMasuk.Totalharga'


create rule rulTBM
as @jumlahTBM >0

sp_bindrule 'RulTBM', 'RincianTBMasuk.jumlahTBM'
```

```
create rule rulTBK
as @TotalHarga > 0

sp_bindrule 'rulTBK', 'TBKeluar.TotalHarga'


create rule rulRTBK
as @JumlahTBK > 0

sp_bindrule 'rulRTBK', 'RincianTBKeluar.JumlahTBK'


-- untuk mencoba rule

insert into barang
values ('B0007','Laptop','Biji','25','-1200')

insert into barang
values ('B0006','Printer','Biji','50','500000')

select * from barang

-------------------------------------------------------------------------------
-------------------
-- STORED PROCEDURE

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

-- MEMBUAT STORED PROCEDURE MELALUI QUERY ANALYZER

CREATE PROCEDURE SP_Supplier_TB_Masuk
AS
Select TBMasuk.NoTBM,TBMasuk.KodeSup,Supplier.NamaSup
from Supplier, TBMasuk
where TBMasuk.KodeSup=Supplier.KodeSup

exec SP_Supplier_TB_masuk

drop procedure SP_Supplier_TB_masuk
-------------------------------------------------------------------------------
-------------------

-- STORED PROCEDURE DENGAN PARAMETER

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

CREATE PROCEDURE SP_Tampilbarang5 @kodebrg varchar(5),@Satuan varchar(20)
AS
Select * from Barang
where kodeBrg=@KodeBrg and SatuanBrg=@Satuan

Exec SP_TampilBarang5 'B0002','Biji'

-------------------------------------------------------------------------------
-------------------
-- CURSOR
-------------------------------------------------------------------------------
-------------------

-- Membuat Cursor

declare curTerlambat cursor for
select KodeSup, TglTransaksi, tglDatang
from TBMasuk
open curTerlambat
```

```
declare @KodeSup varchar(20)
declare @TglTransaksi datetime
declare @TglDatang datetime
declare @Ket varchar (20)
fetch curTerlambat into @KodeSup, @TglTransaksi, @TglDatang
PRINT 'Kode Sup  Tanggal Transaksi  Tanggal Datang    Keterangan'
print'-----------------------------------------------------------------'
while (@@fetch_status =0)
begin
     if (@TglTransaksi < @TglDatang)
            select @ket = 'Terlambat'
     else

         select @ket = 'Tidak Terlambat'
       print @KodeSup +'        '+ convert
       (char(11),@TglTransaksi)+'      '+convert
       (char(11),@TglDatang)+ '      ' +@Ket
   fetch curTerlambat into @KodeSup, @TglTransaksi, @TglDatang
end

close curTerlambat

deallocate curTerlambat

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

-- FUNCTION
-------------------------------------------------------------------------------
------------------
-- Mencari sebagian NamaBrg menggunakan function

create FUNCTION fn_carinamaBrg (@BagianNamaBrg varchar(15))
RETURNS TABLE
AS
      return (SELECT NamaBrg
                  FRom barang
                  WHERE NamaBrg like '%' + @BagianNamaBrg + '%')


-- Menjalankan funcion fn_cari NamaBrg

select * from fn_carinamaBRg('Mon')

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

-- TRANSACTION
-------------------------------------------------------------------------------
------------------
-- Memasukkan data dalam Tabel RincianTBMasuk dan mengupdate stok barang   dalam
table barang
select * from barang
select * from RincianTBMasuk

BEGIN TRANSACTION
INSERT INTO RincianTBMasuk values ('RM0006','M0001','B0001','50')
UPDATE Barang
SET StokBrg=StokBRg + 50
WHERE KodeBrg='B0001'

select * from RincianTBMasuk
select * from barang

-- Memasukkan data dalam Table RincianTBKeluar dan mengupdate stokBrg dalam
table barang

select * from barang
select * from RincianTBKeluar
```

```
BEGIN TRANSACTION
INSERT INTO RincianTBKeluar values ('RK0002','K0001','B0002','5')
UPDATE barang
SET StokBrg=Stokbrg - 5
WHERE KodeBrg='B0002'
IF (SELECT stokBrg FROM Barang WHERE kodeBrg ='B0002') < 0
        BEGIN
        PRINT 'Stok Kurang!'
        ROLLBACK TRANSACTION
        END
ELSE
BEGIN
PRINT 'Transaksi berhasil'
COMMIT TRANSACTION
END

select * from RincianTBKeluar
Select * from Barang
-------------------------------------------------------------------------------
-------------------
-- TRIGGER
-------------------------------------------------------------------------------
-------------------

-- Membuat Trigger dalam table TBKeluar

create trigger [trg_waktu] ON [dbo].[TBKeluar]
for insert
as
declare @NoTBK varchar (5)
select @NoTBK = NoTBK from TBKeluar
update TBKeluar set TglTransaksi = getdate () where NoTBK = @NoTBK

insert into TBKeluar (NoTBK, kodeplg, TotalHarga)
values ('K0007','P0001','999999')

select * from TBKeluar


-- Membuat Trigger dalam table TBMasuk

create trigger [trg_waktu2] ON [dbo].[TBMasuk]
for insert
as
declare @NoTBM varchar (5)
select @NoTBM = NoTBM from TBMasuk
update TBMasuk set TglTransaksi = getdate () where NoTBM = @NoTBM

insert into TBMasuk (NoTBM, kodesup, TotalHarga)
values ('M0006','S0002','999999')

select * from TBMasuk

-------------------------------------------------------------------------------
-------------------
-- Menambahkan kolom baru

alter table Pelanggan
        add EmailPlg varchar(20)

select * from Pelanggan

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

-- Menghapus kolom

alter table Pelanggan
```

Drop column EmailPlg

```
select * from Pelanggan

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

-- Mengganti nama tabel

exec sp_rename 'Pelanggan','Customer'

select * from customer


exec sp_rename 'Customer','Pelanggan'

select * from Pelanggan

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

-- Mengganti nama kolom

exec sp_rename 'Pelanggan.[EmailPlg]','EmailCustomer','column'

select * from Pelanggan


exec sp_rename 'Pelanggan.[EmailCustomer]','EmailPlg','column'

select * from Pelanggan

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

-- Mengganti ukuran dan tipe data

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

-- Ukuran data

alter table Pelanggan
alter column NamaPlg Varchar(30)

alter table Pelanggan
alter column NamaPlg Varchar(20) -- > mengembalikan aslinya lagi

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

-- Tipe data

alter table Pelanggan
alter column NamaPlg Varchar(30)


alter table Pelanggan
alter column NamaPlg char(30)


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

-- Perintah UPDATE

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

-- Contoh pertama, misalnya  akan  dilakukan  perubahan  nilai pada   kolom
```

JumlahTBK tabel TBkeluar yang memiliki NoTBK K0001 (perhatikan tabel data RincianTBK keluar pada record pertama nilai kolom JumlahTBK = 1 akan dibah menjadi 3). Maka perintah yang digunakan adalah :

```
Update RincianTBKeluar set JumlahTBK=3 where NoTBK = 'K0001'

Select * from RincianTBKeluar

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

-- Perintah UPDATE
-------------------------------------------------------------------------------
------------------
```

-- Contoh kedua, misalnya akan dilakukan perubahan data pada tabel barang yang semula dengan kode barang B0001 memiliki nama CPU, satuan biji, Stok Brg berjumlah 20 diganti dengan nama CD dengan satuan keping dan jumlah stoknya 1000, maka perintahnya:

```
Update Barang set namaBrg='CD',satuanBrg='Keping',stokBrg='1000' where
kodeBrg='B0001'

Select * from Barang

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

-- Perintah DELETE

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

-- 1. Perintah Delete dengan syarat

Delete from TBKeluar where NoTBK = 'K0001'

Select * from TBKeluar


-- 2. Perintah Delete tanpa syarat

Delete from TBKeluar

Select * from TBKeluar

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

-- Membuat VIEW

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

create view tampil_barang as (select * from barang) --> membuat view (dgn

nama tampil_barang)

Select * from tampil_barang --> menampilkan view

Drop view tampil_barang --> menghapus view

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


-- SORTIR DAN FILTER

-------------------------------------------------------------------------------
```

```
------------------
-- A. Menggunakan ORDER BY
-------------------------------------------------------------------------------
------------------
-- A.1. Perintah Order By
Select NamaBrg, StokBrg from Barang order by NamaBrg --> berdasarkan nama
        kolom (NamaBrg)
Select NamaBrg, StokBrg from Barang order by 2 --> berdasarkan letak kolom
(kolom kedua yaitu StokBrg)

-- A.2. Perintah Order By dan Asc
Select NamaBrg, StokBrg from Barang order by NamaBrg Asc

-- A.3. Perintah Order By dan Desc
Select NamaBrg, StokBrg from Barang order by NamaBrg Desc
-------------------------------------------------------------------------------
------------------
-- B.  Menggunakan klausa WHERE
-------------------------------------------------------------------------------
------------------
-- B.1. Klause  where dan sama dengan ( = )
Select * From barang where  NamaBrg = 'Monitor'
Select * From barang where  KodeBrg = 'B0004'

-- B.2. Klause  where dan lebih besar( > )
Select * From barang where  stokBrg > 30

-- B.3. Klause  where dan lebih besar sama  dengan ( >= )
Select * From barang where  stokBrg >=30
-- B.4. Klause  where dan lebih kecil( < )
Select * From barang where  kodeBrg < 'B0003'

-- B.5. Klause  where dan lebih kecil Sama  dengan ( <= )
Select * From barang where  kodeBrg <= 'B0003'

-- B.6. Klause  where tidak  sama dengan ( <> )
Select * From barang where  stokBrg <> '1000'
Select * From barang where  satuanBrg <>'biji'

-- B.7. Klause  where dan Between
```

```
Select * From barang where  stokBrg <>'1000'
Select * From barang where  stokBrg  between 30 and 50

-- B.8. Klause  where dan In
Select * From barang where  stokBrg <>'1000'
Select * From barang where  stokBrg  in (50,1000)

-- B.9. Klause  where dan Not
Select * From barang where not kodeBrg = 'B0001'

-- B.10. Klause  where dan Is Null
Select * From pelanggan where kotaplg is null

-- B.11. Klause  where dan Operator  OR
Select * From Barang where satuanbrg = 'Biji' or stokBrg>50

-- B.12. Klause  where dan Operator AND
Select * From barang  where satuanBrg = 'Biji' and stokBrg=30

-- B.13. Klause  where dan Operator LIKE
Select * From pelanggan  where kotaPlg like 'c%'
Select * From pelanggan  where kotaPlg like 'bandung%'
Select * From supplier  where namasup like  '%a'
Select * From supplier  where telpsup like '%33'
Select * From supplier  where namasup like '%ma%'
------------------------------------------------------------------------------
------------------
-- C. Menggunakan ALIAS
------------------------------------------------------------------------------
------------------
Select kodebrg as [kode barang] from barang
-- atau
Select KodePlg as [kode pelanggan], NamaPlg as [nama pelanggan] from Pelanggan
------------------------------------------------------------------------------
------------------
-- D. Menggunakan Fungsi AGREGAT
------------------------------------------------------------------------------
------------------
-- D.1. Perintah Count
```

```
Select count (*) from barang

Select count (kodebrg) as [jumlah baris kode], count (stokbrg) as [jumlah baris
stok] from barang


-- D.2. Perintah AVG ()

Select avg (stokbrg) as [rata-rata stok] from barang

-- D.3. Perintah Max ()

Select * from RincianTBMasuk

Select max (jumlahTBM) as [jumlah mak barang masuk] from RincianTBMasuk

Select max (stokbrg) as [jumlah mak STOK barang] from barang


-- D.4. Perintah Min ()

Select min (stokbrg) as [jumlah min stok barang] from Barang

Select min (jumlahTBM) as [jumlah terkecil barang masuk] from RincianTBMasuk

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

-- Bekerja dengan Multitabel (JOIN)

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

Select RincianTBMasuk.NoTBM,RincianTBMasuk.KodeBrg,Barang.NamaBrg
from RincianTBMasuk JOIN Barang
ON RincianTBmasuk.KodeBrg=Barang.KodeBrg


Select TBmasuk.notbm,TBmasuk.kodesup,Supplier.namasup
from tbmasuk, supplier
where TBmasuk.kodesup = supplier.kodesup


Select RincianTBmasuk.kodebrg, Barang.NamaBrg, TBmasuk.Kodesup,Supplier.namasup
from Rinciantbmasuk, supplier, barang, Tbmasuk
where TBmasuk.kodesup = supplier.kodesup and Supplier.KodeSup=TBMasuk.KodeSup


Select RincianTBMasuk.kodebrg,Barang.Namabrg,TBmasuk.KodeSup,Supplier.Namasup
from RincianTBMasuk, supplier, TBMasuk,barang
Where TBmasuk.kodesup = supplier.kodesup and
Rinciantbmasuk.kodebrg=barang.kodebrg and Barang.kodebrg='B0001'


Select  COUNT (*)
from tbmasuk, supplier, barang, rincianTbMasuk
Where TBmasuk.kodesup = supplier.kodesup and
rinciantbmasuk.kodebrg=barang.kodebrg

------------------------------------------------------------------------------
------------------
-- Membuat Index
------------------------------------------------------------------------------
------------------
CREATE INDEX idxKategoriBarang
ON Barang (NamaBrg)

CREATE INDEX idxKategoriSupplier
ON Supplier (NamaSup)
```

```
-- Mengecek/ Memverify Index
 sp_helpindex barang
 sp_helpindex supplier
```