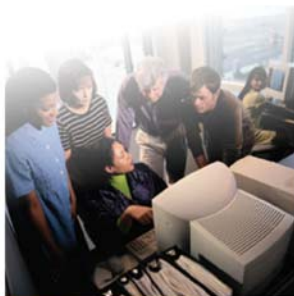


STMIK AMIKOM YOGYAKARTA



Modul Praktikum Sistem Basis Data dengan SQL Server 2000



Semester 3 Gasal
T.A. 2007/2008
S1 Teknik Informatika
2 SKS



- Introduction to SQL
- Introduction to SQL Server Objects
- Registering Database
- Attaching Database
- Detaching Database
- DDL Statement
- Constraints
- Data Diagram
- DML Statement

Copyright : M.Rudyanto Arief,S.T.,M.T.

PHOTOCOPIABLE

rudy@amikom.ac.id [www.rudy.amikom.ac.id]



MODUL 3

DATA DEFINITION LANGUAGES (DDL)

Bagian 1

Tujuan

- ☐ Mampu mendefinisikan (membuat, menghapus, merubah) obyek database
- ☐ Mampu menggunakan perintah-perintah DDL SQL untuk mengelola database

Teori Singkat

Seperti telah dijelaskan di modul sebelumnya, DDL adalah sebuah bahasa yang digunakan untuk mendefinisikan data dan obyek pada *database*.

Berikut ini sintaks-sintaks yang termasuk dalam DDL didalam SQL Server :

| Perintah | Keterangan |
|------------------|---------------------------------------|
| CREATE DATABASE | Untuk membuat <i>database</i> |
| CREATE TABLE | Untuk membuat tabel |
| CREATE VIEW | Untuk membuat <i>view</i> |
| CREATE PROCEDURE | Untuk membuat <i>Stored Procedure</i> |
| ALTER DATABASE | Mengubah database |
| DROP DATABASE | Menghapus database |

Membuat Database

Database merupakan obyek yang utama dalam DBMS SQL Server. Semua obyek-obyek database yang di jelaskan di modul 1 tersimpan di dalam obyek database. Di dalam SQL Server terdapat beberapa database default yang otomatis di buat ketika SQL Server pertama kali di install, yaitu: master, model, msdb, Northwind, pubs, tempdb. Selain obyek-obyek database tersebut terdapat juga database-database lain yang di buat oleh seorang database administrator (DBA) dengan menggunakan perintah-perintah SQL. Untuk membuat database di SQL Server maka di butuhkan perintah SQL yang termasuk kategori data definition language (DDL). Berikut adalah perintah dasar untuk membuat database menurut SQL Server Books Online.

Perintah dasar:

```
CREATE DATABASE database_name
[ ON
    [ < filespec > [ ,...n ] ]
    [ , < filegroup > [ ,...n ] ]
]
[ LOG ON { < filespec > [ ,...n ] } ]
[ COLLATE collation_name ]
[ FOR LOAD | FOR ATTACH ]
< filespec > ::=
[ PRIMARY ]
( [ NAME = logical_file_name , ]
    FILENAME = 'os_file_name'
    [ , SIZE = size ]
    [ , MAXSIZE = { max_size | UNLIMITED } ]
    [ , FILEGROWTH = growth_increment ] ) [ ,...n ]
< filegroup > ::=
FILEGROUP filegroup_name < filespec > [ ,...n ]
```

Keterangan:

Database Name: Nama *Database* dan harus bersifat unik, panjang maksimal 128 karakter

Log On : Menyatakan file yang dipakai untuk menyatakan log.

On Primary : Menyatakan *file-file* disk untuk penyimpanan data dari sebuah *database*.

Keterangan lain silahkan baca di **SQL SERVER BOOKS ONLINE**.

Atau dapat di sederhanakan menjadi:

```
CREATE DATABASE nama_database
```

Merubah Database

Untuk merubah database seperti menambahkan file atau menghapus file database dan kelompok file dari database menggunakan perintah `ALTER DATABASE`. Selain merubah nama file database, `ALTER DATABASE` juga dapat merubah nama database, nama kelompok file, dan nama logik dari data file dan log file.

Berikut adalah perintah dasar menurut SQL Server Books Online:

```
ALTER DATABASE database
{ ADD FILE < filespec > [ ,...n ] [ TO FILEGROUP filegroup_name ]
| ADD LOG FILE < filespec > [ ,...n ]
| REMOVE FILE logical_file_name
| ADD FILEGROUP filegroup_name
| REMOVE FILEGROUP filegroup_name
| MODIFY FILE < filespec >
| MODIFY NAME = new_dbname
| MODIFY FILEGROUP filegroup_name {filegroup_property | NAME =
new_filegroup_name }
| SET < optionspec > [ ,...n ] [ WITH < termination > ]
| COLLATE < collation_name >
}
```

Menghapus Database

Untuk menghapus satu atau lebih dari satu database dari SQL Server menggunakan perintah `SQL DROP DATABASE`. Menghapus database berarti menghapus database dan file-file yang digunakan oleh database tersebut di harddisk komputer.

Berikut adalah perintah dasarnya:

```
DROP DATABASE database_name [ ,...n ]
```

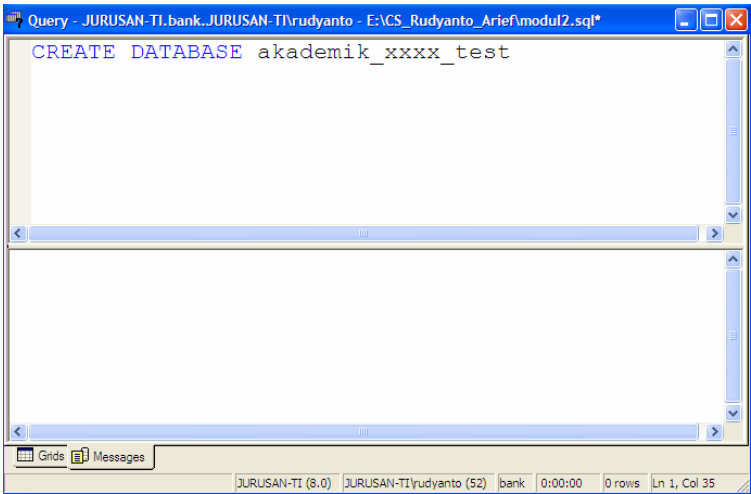
Latihan 3.1.

Membuat Database tanpa parameter

☐ Aktifkan query analyzer, kemudian pada jendela editor ketikkan perintah berikut:

```
CREATE DATABASE akademik_xxxx_test;
```

Dimana xxxx = 4 (empat) digit terakhir nim anda.



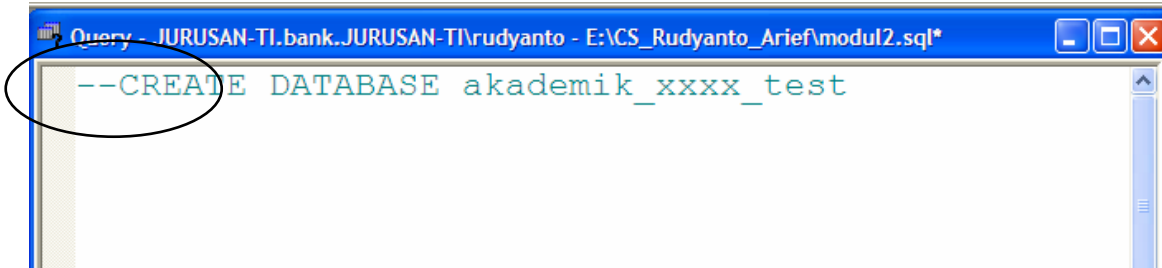
- Kemudian eksekusilah perintah tersebut dengan menekan tombol **F5**. Perintah diatas adalah perintah SQL untuk membuat sebuah database baru dengan nama **akademik_xxxx_test**. Jika perintah tersebut di eksekusi maka SQL Server akan membuat database **akademik_xxxx_test** dan **data filenya** beserta **log filenya** di simpan di direktori default penyimpanan data SQL Server.

Latihan 3.2.

Membuat database dengan parameter tambahan

Berikut contoh untuk membuat database baru dengan beberapa penambahan parameter di dalamnya.

- Masih di jendela editor yang sama dengan perintah **Latihan 3.1.**, jadikan komentar perintah pada **Latihan 3.1.** (tidak perlu di hapus) sebelumnya dengan menambahkan tanda *double dash* (--) di depan baris perintah SQL tersebut.

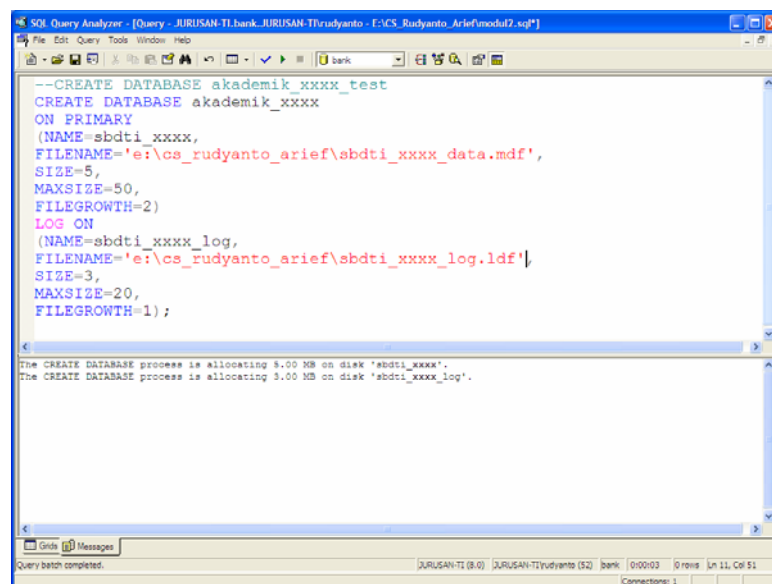


- Ketikkan perintah SQL berikut di baris berikutnya di jendela editor. Perintah berikut adalah untuk membuat database dengan nama **akademik_xxxx**, **xxxx** merupakan 4 (empat) digit nim anda.

```
CREATE DATABASE akademik_xxxx
ON PRIMARY
(NAME=sbdti_xxxx,
FILENAME='d:\nama folder anda\sbdti_xxxx_data.mdf',
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='d:\nama folder anda\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
```

```
--CREATE DATABASE akademik_xxxx_test
CREATE DATABASE akademik_xxxx
ON PRIMARY
(NAME=sbdti_xxxx,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_data.mdf',
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='d:\cs_rudyanto_arief\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
|
```

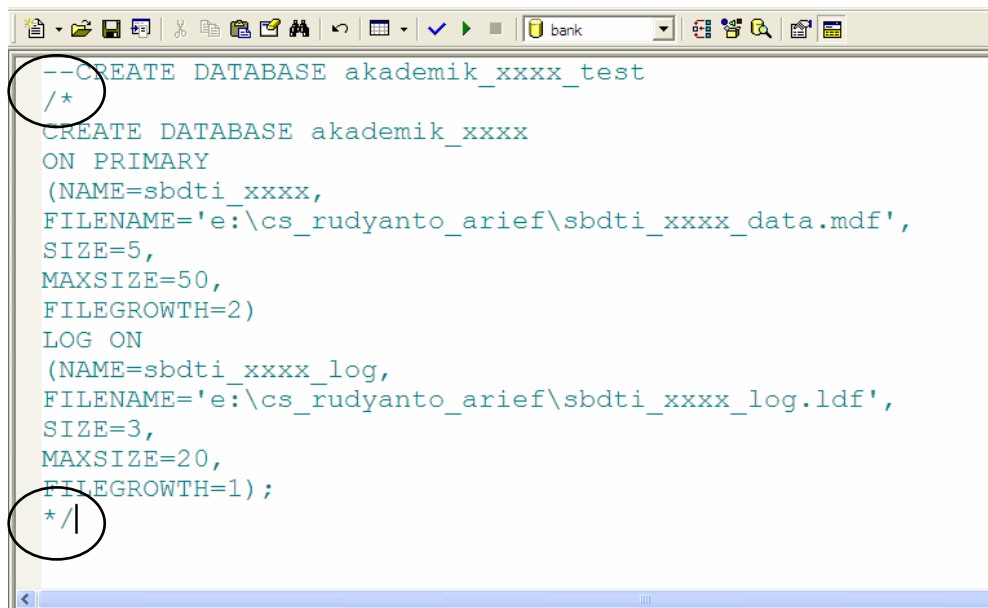
- Pada contoh nama folder tempat penyimpanan data file dan log filenya di simpan di direktori **e:\cs_rudyanto_arief**.
- Eksekusilah perintah tersebut dengan menekan tombol **F5**. Jika sukses maka akan muncul konfirmasi di bagian jendela hasil query di bawah jendela editor yang menyatakan bahwa dataabase telah berhasil dibuat dengan lokasi direktori penyimpanan untuk data file dan log filenya.



```
--CREATE DATABASE akademik_xxxx_test
CREATE DATABASE akademik_xxxx
ON PRIMARY
(NAME=sbdti_xxxx,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_data.mdf',
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
```

The screenshot shows the SQL Query Analyzer interface. The main window displays a SQL script to create a database named 'akademik_xxxx'. The script includes a comment '--CREATE DATABASE akademik_xxxx_test', followed by 'CREATE DATABASE akademik_xxxx'. It then defines the primary data file with '(NAME=sbdti_xxxx, FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_data.mdf', SIZE=5, MAXSIZE=50, FILEGROWTH=2)' and the log file with 'LOG ON (NAME=sbdti_xxxx_log, FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_log.ldf', SIZE=3, MAXSIZE=20, FILEGROWTH=1);'. Below the script, a status message indicates that the CREATE DATABASE process is allocating space on disk for both files. The bottom status bar shows 'Query batch completed.' and 'Connections: 1'.

- ☐ Perhatikan perbedaan antara contoh perintah SQL di **Latihan 3.1.** dan **Latihan 3.2.**
- ☐ File Primer yang berisi data disimpan dengan nama **sbdti_xxxx_data.Mdf**. Folder penyimpanannya di sesuaikan dengan nama folder penyimpanan kelas Anda.
- ☐ File Awal meminta *space* penyimpanan sebesar 5 MB, dengan ukuran maksimum 50 MB dengan penambahan 2 MB.
- ☐ File Log disimpan dengan nama **sbdti_xxxx_log.Ldf**. Folder penyimpanannya di sesuaikan dengan nama folder penyimpanan kelas Anda. Ukuran 3 MB dengan maksimal 20 MB dengan penambahan 1 MB.
- ☐ Untuk mengecek database tersebut apakah sudah dibuat ketikkan perintah berikut melalui jendela Query Analyzer. Sebelumnya non aktifkan terlebih dahulu perintah SQL **Latihan 3.2.** yang sebelumnya di ketik di jendela editor dengan menjadikan komentar untuk perintah tersebut.

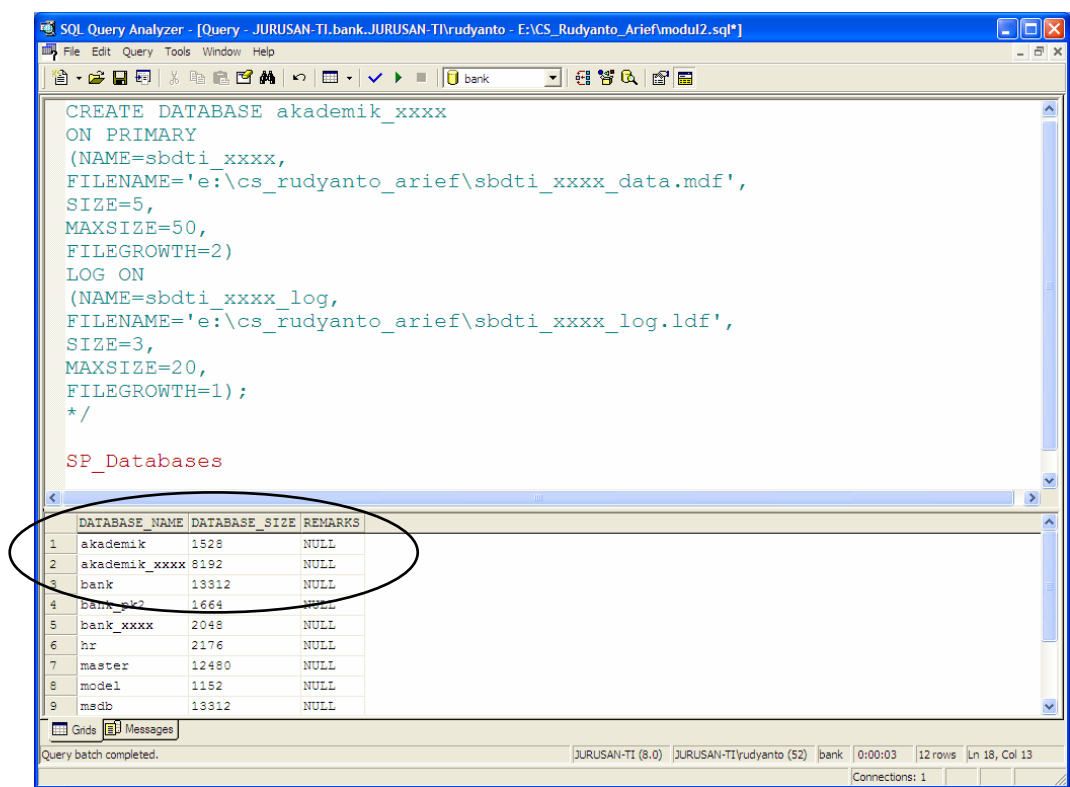


```
--CREATE DATABASE akademik_xxxx_test
/*
CREATE DATABASE akademik_xxxx
ON PRIMARY
(NAME=sbdti_xxxx,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_data.mdf',
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
*/
```

sp_databases

Atau

SELECT name FROM master.dbo.sysdatabases



Latihan 3.3.

Merubah nama database

- ☐ Masih di jendela editor yang sama, non aktifkan perintah sp_databases. Kemudian ketikkan perintah berikut di baris berikutnya di jendela editor query analyzer.

```
ALTER DATABASE akademik_xxxx_test
MODIFY NAME = akademik_xxxx_test2
```

```
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
*/

--sp_databases
ALTER DATABASE akademik_xxxx_test
MODIFY NAME = akademik_xxxx_test2
```

- Eksekusilah perintah tersebut dengan menekan tombol F5.
- Perintah diatas adalah untuk melakukan perubahan nama database yang sebelumnya akademik_xxxx_test menjadi akademik_xxxx_test2.

```
SIZE=5,
MAXSIZE=50,
FILEGROWTH=2)
LOG ON
(NAME=sbdti_xxxx_log,
FILENAME='e:\cs_rudyanto_arief\sbdti_xxxx_log.ldf',
SIZE=3,
MAXSIZE=20,
FILEGROWTH=1);
*/

sp_databases
--ALTER DATABASE akademik_xxxx_test
--MODIFY NAME = akademik_xxxx_test2
```

| | DATABASE_NAME | DATABASE_SIZE | REMARKS |
|---|---------------------|---------------|---------|
| 1 | akademik | 1528 | NULL |
| 2 | akademik_xxxx | 8192 | NULL |
| 3 | akademik_xxxx_test2 | 1144 | NULL |
| 4 | bank | 13312 | NULL |
| 5 | bank_pk2 | 1664 | NULL |
| 6 | bank_xxxx | 2048 | NULL |
| 7 | hr | 2176 | NULL |

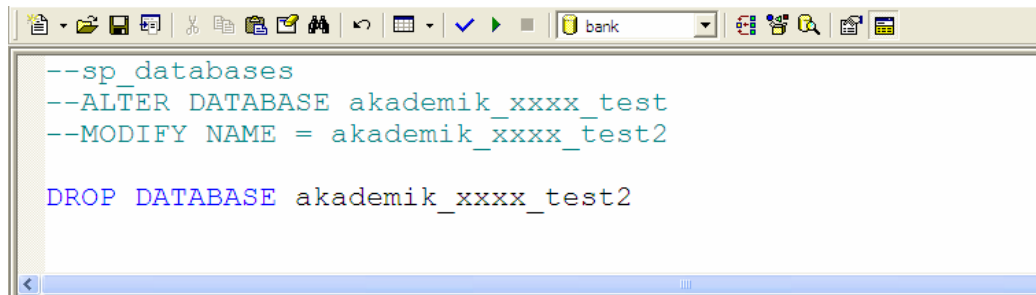
- Untuk melihat kembali hasil perubahan, maka aktifkan kembali perintah `sp_databases` di jendela editor dengan menghapus kembali tanda *double dash* (`--`) di depan baris perintah tersebut dan non aktifkan terlebih dahulu perintah **Latihan 3.3**.

Latihan 3.4.

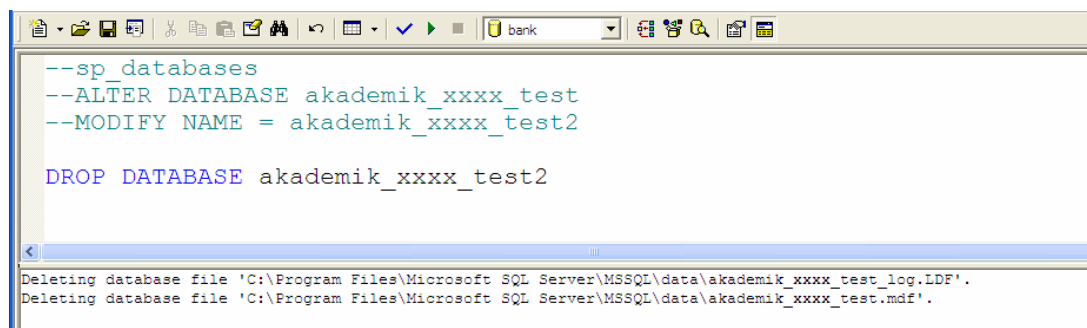
Menghapus database

- ☐ Non aktifkan terlebih dahulu semua perintah SQL yang ada di jendela editor, kemudian ketikkan perintah berikut:

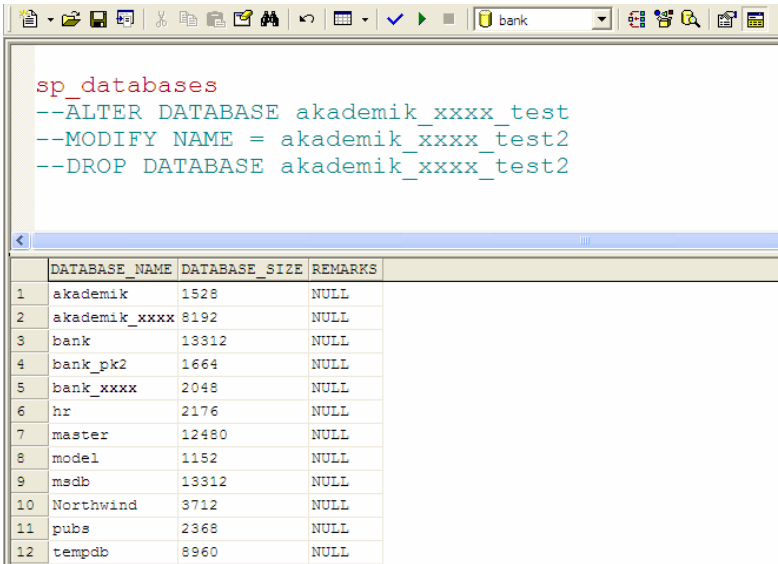
```
DROP DATABASE akademik_xxxx_test2
```



- ☐ Eksekusilah perintah tersebut dengan menekan tombol F5. Jika sukses maka jendela hasil query akan menampilkan pesan konfirmasi di dalamnya.



- ☐ Untuk melihat apakah database **akademik_xxxx_test2** berhasil di hapus, maka aktifkan kembali perintah **sp_databases** dengan me-non aktifkan terlebih dahulu perintah **DROP DATABASE akademik_xxxx_test2**. Eksekusilah perintah tersebut dengan menekan tombol F5.



- ❑ Simpanlah semua perintah SQL yang sudah di ketikkan di jendela editor mulai dari **Latihan 3.1.** sampai **Latihan 3.4.** dengan nama file: **modul3.sql** yang di simpan di dalam direktori kerja anda masing-masing.

Penugasan

1. Jelaskan perbedaan hasil antara perintah SQL Latihan 3.1. dan Latihan 3.2.
2. Tuliskan perintah untuk membuat database baru dengan nama `akademik_xxxx_test3` yang maksimal ukuran data filenya 100 MB, pertumbuhannya tiap 20 MB dan lokasi data file dan log filenya otomatis di atur oleh SQL Server begitu pula dengan nama data file dan nama log filenya di tentukan otomatis oleh SQL Server.
3. Tuliskan perintah SQL untuk merubah nama database di soal nomor 2 (dua) menjadi `akademik_latihan_xxxx`.
4. Tuliskan perintah untuk menghapus database lebih dari satu dalam satu perintah SQL. (asumsi nama database yang akan di hapus: `test1`, `test2`, `test3`).
5. Apa yang terjadi jika perintah `DROP DATABASE` di lakukan pada sebuah database?