

Chương 2 (phần 2)

Tạo và quản lý database

Giáo trình & Tài liệu tham khảo:

1. **Microsoft SQL Server 2008 R2 Unleashed**, Ray Rankins, Paul Bertucci, Chris Gallelli, Alex T. Silverstein, 2011, Pearson Education, Inc
2. **MS SQL Server 2012 T-SQL fundamentals**, Tizik Ben-Gan
3. <https://docs.microsoft.com/>

Nội dung

- Tổ chức Lưu trữ của SQL Server
- Các database file
- Tạo database và thiết lập thuộc tính cho database
- Quản lý database

Lưu trữ data trong SQL Server

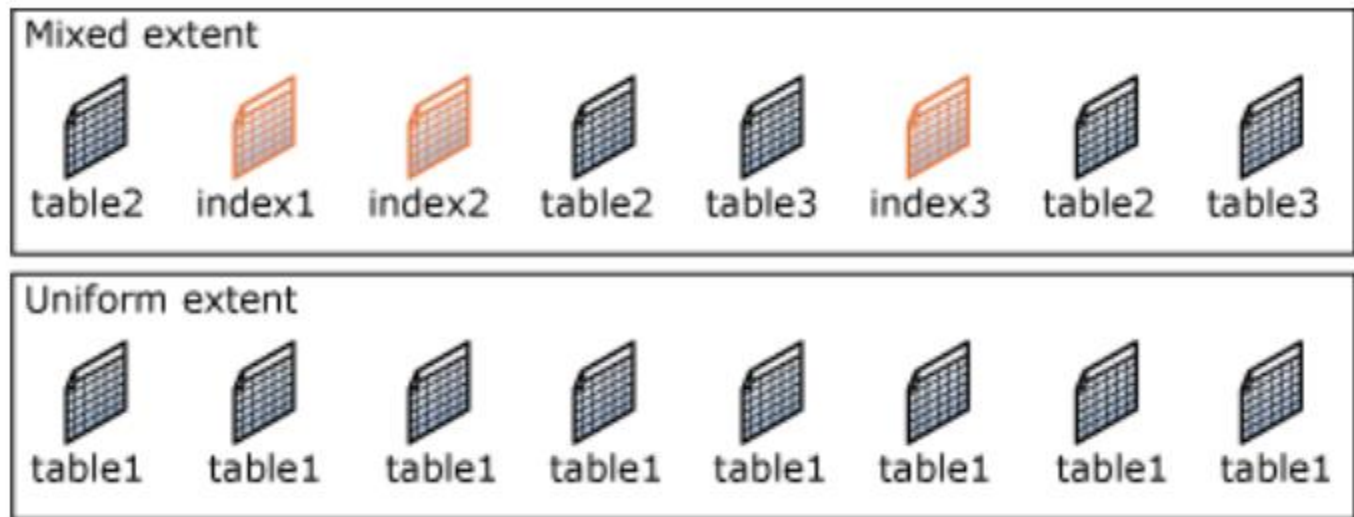
- Đơn vị lưu trữ data trong SQL Server là **page**
 - size page : 8KB (hay 8192 bytes)
 - SQL Server đọc/ghi data từ đĩa theo đơn vị page
- Một data file (.mdf hay .ndf) của một database được chia thành các page , đánh số từ 0->n

Lưu trữ data trong SQL Server

- Các data row của một table được chứa trong các page
 - Một data row chỉ nằm trong 1 page (không nằm trên nhiều page), và có size giới hạn là size của page = 8060 byte (Mỗi page có header chứa thông tin chung 96byte...)
- Những column chứa loại dữ liệu text/image có kích thước vượt quá page size : cấp page theo nhu cầu ,và dùng pointer để quản lý
 - Large object data types: (text, ntext, image, nvarchar(max), varchar(max), varbinary(max), and xml data)
 - Variable length columns when the data row exceeds 8 KB: (varchar, nvarchar, varbinary, and sql_variant)

Lưu trữ data trong SQL Server

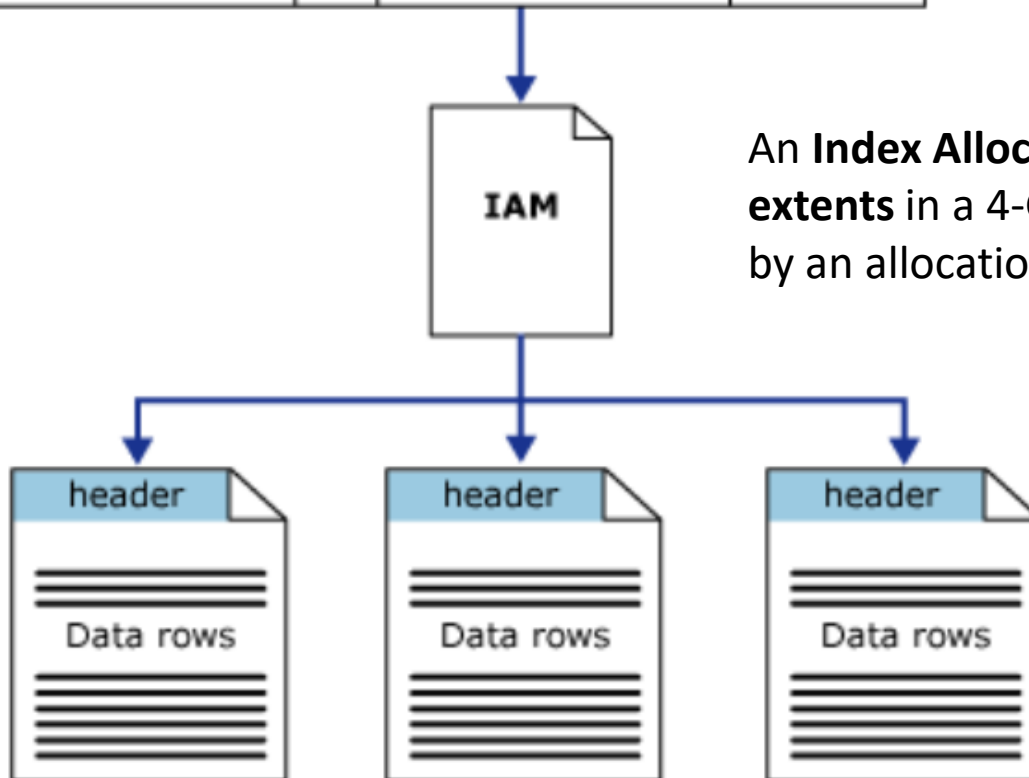
- Để tăng hiệu suất I/O, SQL Server cấp phát không gian cho table theo **Extents**
 - Extent là một dãy gồm 8 page vật lý liên tiếp trên đĩa , 64KB



Lưu trữ data trong SQL Server

- Cấu trúc của một **table** ko có clustered index

id	index_id = 0	first_iam_page	
----	--------------	----------------	--



An **Index Allocation Map (IAM)** page maps the **extents** in a 4-GB part of a database file used by an allocation unit

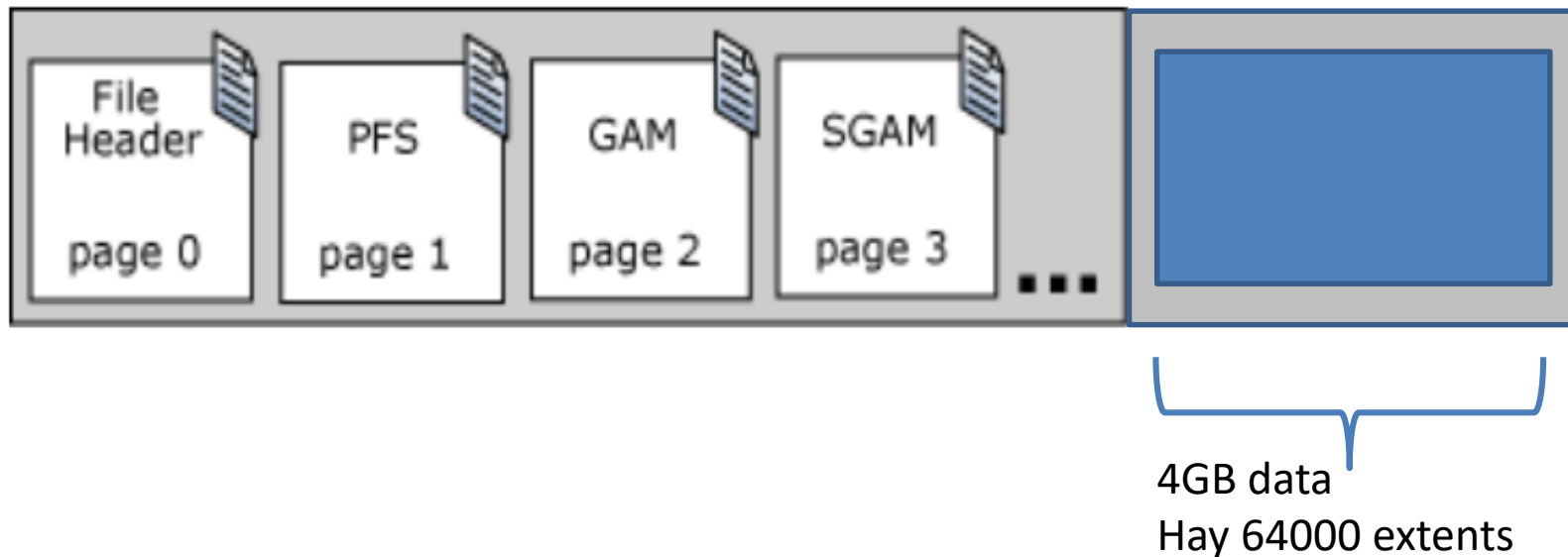
Lưu trữ data trong SQL Server

- Các loại page có trong một **data file**
 - Data pages và text/image pages
 - Index pages
 - System pages : chứa metadata về các page của file (PFS, GAM, SGAM, IAM, DCM, BCM pages)

Global Allocation Map (GAM)

GAM pages record what extents have been allocated. Each GAM covers 64,000 extents, or almost 4 gigabytes (GB) of data. The GAM has 1-bit for each extent in the interval it covers. If the bit is 1, the extent is free; if the bit is 0, the extent is allocated.

Lưu trữ data trong SQL Server



Page Free Space (PFS)

Global Allocation Map (GAM)

Shared Global Allocation Map (SGAM)

Tham khảo : [pages-and-extends-architecture-guide](https://docs.microsoft.com/en-us/sql/relational-databases/pages-and-extends/pages-and-extends-architecture-guide) trong docs.microsoft.com

Các database file

- Mỗi database bao gồm các file
 - Một hay nhiều **data file**
 - Một database phải có **1 Primary data file (.mdf)** : chứa thông tin tham chiếu đến các file khác của database và data của database
 - Có thể có **một/nhiều** secondary data file (.ndf)
 - Một hay nhiều **transaction log file (.ldf)**
 - Transaction là đơn vị chứa một chuỗi các thao tác liên quan logic
 - SQL Server lưu vết các transaction xảy ra trong log file
=> giúp database recovery

Các database file

performance and management of database files. **Consider, for example,** a situation in which a database server has four physical drives available for the data file(s). Each drive is 1GB in size, but the database you are creating is 2GB. In this example, the database will not fit on one drive. A solution to this problem is to create a primary data file on one of the drives and a secondary data file on each of the three remaining drives. SQL Server automatically spreads the 2GB database across the four data files located on four separate drives.

This is most apparent with large databases. **For example,** let's say you have a 100GB database, and it contains only a primary data file. If you want to move this database to another environment, you must have a drive that is at least 100GB to store the primary data file. If you want to copy the database to a server that has 10 50GB drives, you cannot do it. You have the space across all 10 drives, but you do not have a single drive that can hold the primary data file. If, however, you create the database with several secondary files, you have the option of placing each of the secondary files on a separate drive.

Filegroups

- Filegroup cho phép ta nhóm **các data file** nằm trên các ổ đĩa khác nhau
 - Các table, index hay large object data khi được chỉ định lưu trữ trong một filegroup : sẽ được lưu trữ ngang qua các file trong filegroup
- ⇒ Tăng hiệu năng truy suất dữ liệu

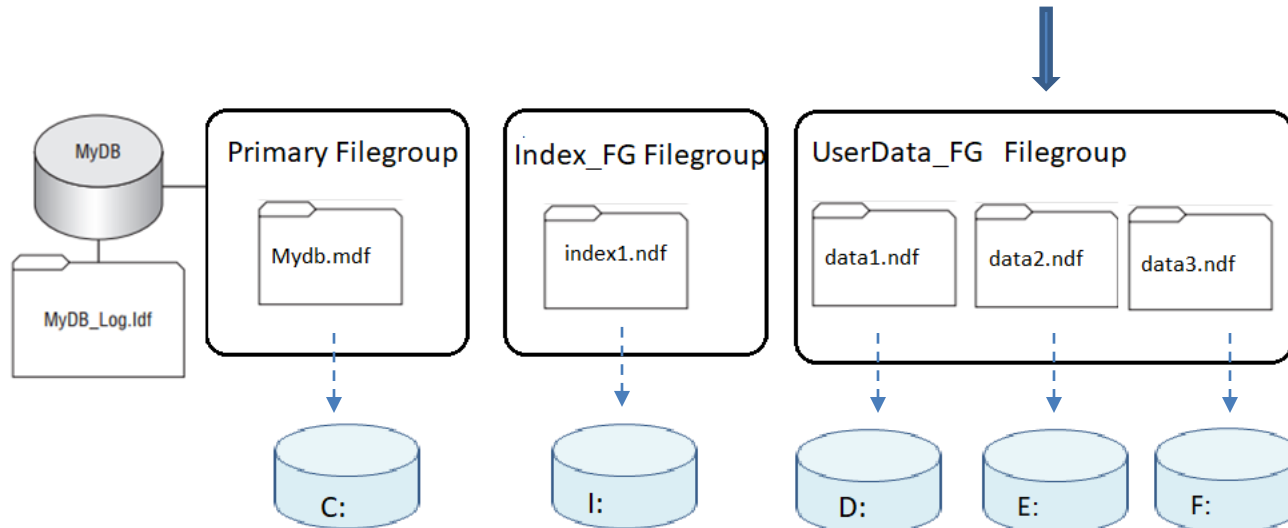
- Mặc định mỗi database có một filegroup là **primary filegroup**
- Mặc định Primary filegroup là **default filegroup** => có thể thay đổi default filegroup

Filegroups

- Vd trang 713

For example, you could create a filegroup called UserData_FG, consisting of three files spread over three physical drives. You could create another filegroup named Index_FG, with a single file, on a fourth drive. Then, when you create the tables, you can create them on the UserData_FG filegroup. You can create indexes on the Index_FG filegroup. This reduces contention between tables because the data is spread over three disks and can be accessed independently of the indexes. If more storage is required in the future, you can easily add additional files to the index or data filegroup, as appropriate.

```
CREATE TABLE dbo.Table1  
(TableId int NULL,  
TableDesc varchar(50) NULL)  
ON [UserData_FG]
```



Filegroups

- **Một số nguyên tắc**

When creating filegroups, you should keep in mind the following restrictions:

- ▶ You can't move a data file to another filegroup after it has been added to the database.
- ▶ Filegroups apply only to data files and not to log files.
- ▶ A data file can be part of only one filegroup and cannot be spread across multiple filegroups.
- ▶ You can have a maximum of 32,767 filegroups for each database.

- **Ưu điểm**

The same performance improvement can be accomplished by using a single file created on a RAID (redundant array of independent disks) stripe set. However, files and filegroups let you easily add new files to new disks.

Filegroups

CREATE DATABASE Sales

ON

```
( NAME = Sales1_dat, FILENAME = 'D:\BTSQL\Sales_dat.mdf',  
    SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),  
( NAME = Sales2_dat, FILENAME = 'E:\BTSQL\Sales2_dat.ndf',  
    SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),  
( NAME = Sales3_dat, FILENAME = 'G:\BTSQL\Sales3_dat.ndf',  
    SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )  
( NAME = Sales4_dat,  
    FILENAME = 'H:\BTSQL\Sales4_dat.ndf',  
    SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )
```

LOG ON

```
( NAME = 'Sales_log',  
    FILENAME = 'D:\BTSQL\salelog.ldf',  
    SIZE = 5MB,  
    MAXSIZE = 25MB,  
    FILEGROWTH = 5MB )
```

Sales DB gồm : bao nhiêu data file ?
Log file ? Các data file thuộc
filegroup nào ? Tên filegroup ?

Filegroups

CREATE DATABASE Sales

ON PRIMARY

```
( NAME = Sales1_dat, FILENAME = 'D:\BTSQ\Sales_dat.mdf',  
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),  
( NAME = Sales2_dat, FILENAME = 'E:\BTSQ\Sales2_dat.ndf',  
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),
```

FILEGROUP SalesGroup1

```
( NAME = Sales3_dat, FILENAME = 'D:\BTSQ\Sales3_dat.ndf',  
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )  
( NAME = Sales4_dat,  
  FILENAME = 'E:\BTSQ\Sales4_dat.ndf',  
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )
```

LOG ON

```
( NAME = 'Sales_log',  
  FILENAME = 'D:\BTSQ\salelog.ldf',  
  SIZE = 5MB,  
  MAXSIZE = 25MB,  
  FILEGROWTH = 5MB )
```

Sales DB gồm : bao nhiêu data file ?
Log file ? Các data file thuộc
filegroup nào ? Tên filegroup ?

Create, alter, drop database

- Thực hiện :
 - **Thiết lập** tên database, các data file và log file , filegroup
 - **Thiết lập** các thuộc tính cho database
 - **Thay đổi** các thiết lập đã tạo
 - **Xóa** database
- Công cụ
 - SSMS
 - Lệnh T-SQL

Create database

- **Tạo database _ Ví dụ**

```
CREATE DATABASE [mydb]
```

```
ON PRIMARY
```

```
( NAME = N'mydb',
```

```
FILENAME = N'C:\mssql2008\data\mydb.mdf' ,
```

```
SIZE = 2048KB ,
```

```
FILEGROWTH = 1024KB )
```

```
LOG ON
```

```
( NAME = N'mydb_log',
```

```
FILENAME = N'C:\mssql2008\log\mydb_log.ldf',
```

```
SIZE = 1024KB ,
```

```
FILEGROWTH = 10%)
```

```
GO
```

Create database

Các thông số

CREATE DATABASE Sales

ON PRIMARY

(**NAME** = Sales1_dat,
 FILENAME = 'D:\BTSQL\Sales_dat.mdf',
 SIZE = 10,
 MAXSIZE = 50,
 FILEGROWTH = 15%)

LOG ON

(**NAME** = 'Sales_log',
 FILENAME = 'D:\BTSQL\salelog.ldf',
 SIZE = 5MB,
 MAXSIZE = 25MB,
 FILEGROWTH = 5MB)

Create database

Các thông số

- Mở rộng file
 - Mỗi file khi đầy (kể từ size ban đầu), file sẽ được tăng kích thước theo chỉ định trong **FILEGROWTH**
- File size
 - Kích thước của file khởi tạo ban đầu theo giá trị trong **SIZE**, và có thể tăng tối đa đến hết kích thước đĩa , hoặc theo **MAXSIZE**

Alter database

- Thêm/xóa files , filegroups
- Thay đổi các thông số của data file, log file

Cú pháp:

ALTER DATABASE database_name

ADD FILE filespec [TO FILEGROUP filegroup_name]

ADD LOG FILE filespec

| REMOVE FILE logical_filename

| ADD FILEGROUP filegroup_name

| REMOVE FILEGROUP filegroup_name

| MODIFY FILE filespec

| MODIFY FILEGROUP filegroup_name

filegroup_property

|SET optionspec [WITH termination]

Thay đổi thuộc tính database

- Thay đổi các thiết lập của database _dùng T-SQL , ví dụ

-- *thiết lập db về offline mode*

```
ALTER DATABASE [AdventureWorks2008] SET OFFLINE WITH NO_WAIT
```

-- *Thiết lập db về single-user mode*

```
EXEC sp_dboption 'AdventureWorks2008', 'single user', 'TRUE'
```

-- *hoặc dùng*

```
ALTER DATABASE [AdventureWorks2008] SET SINGLE_USER WITH NO_WAIT
```

-- *Đổi tên database*

```
ALTER DATABASE [AdventureWorks2008] MODIFY NAME =  
[AdventureWorks2008_New]
```

-- *hoặc dùng*

```
sp_renamedb 'oldname', 'newname'
```

Thay đổi thuộc tính database

Thay đổi các thiết lập cho database – dùng SSMS (Trang 722)

Chọn Database -> Properties -> Options

Collation:	SQL_Latin1_General_CP1_CI_AS
Recovery model:	Full
Compatibility level:	SQL Server 2012 (110)
Containment type:	None
Other options:	
▼ Automatic	
Auto Close	False
Auto Create Statistics	True
Auto Shrink	False
Auto Update Statistics	True
Auto Update Statistics Asynchronously	False
▼ Containment	
Default Fulltext Language LCID	1033
Default Language	English
Nested Triggers Enabled	True
Transform Noise Words	False
Two Digit Year Cutoff	2049
▼ Cursor	
Close Cursor on Commit Enabled	False
Default Cursor	GLOBAL
▼ FILESTREAM	
FILESTREAM Directory Name	
FILESTREAM Non-Transacted Access	Off

▼ Miscellaneous	
Allow Snapshot Isolation	False
ANSI NULL Default	False
ANSI NULLS Enabled	False
ANSI Padding Enabled	False
ANSI Warnings Enabled	False
Arithmetic Abort Enabled	False
Concatenate Null Yields Null	False
Cross-database Ownership Chaining Enabled	False
Date Correlation Optimization Enabled	False
Is Read Committed Snapshot On	False
Numeric Round-Abort	False
Parameterization	Simple
Quoted Identifiers Enabled	False
Recursive Triggers Enabled	False
Trustworthy	False
VarDecimal Storage Format Enabled	True
▼ Recovery	
Page Verify	CHECKSUM
Target Recovery Time (Seconds)	0
▼ Service Broker	
Broker Enabled	False
Honor Broker Priority	False
Service Broker Identifier	3417cea0-fbc0-49c6-a3e3-fde8ebcb8ad3
▼ State	
Database Read-Only	False
Database State	NORMAL
Encryption Enabled	False

Drop database

- Xóa database : lệnh sẽ xóa toàn bộ các data file và log file của database

DROP DATABASE *database_name*

- Không thể xóa các system database

Quản lý Database

- Bao gồm
 - Mở rộng database
 - Thu nhỏ database
 - Di chuyển database
 - Xem thông tin
 - Phát sinh scripts

Quản lý Database

- **Mở rộng database – tăng kích thước file**
 - SQL Server sẽ mở rộng size của file **tự động** khi database đã sử dụng hết không gian đã cấp phát
 - Cần cân nhắc giá trị FILEGROWTH : nếu nhỏ thì ảnh hưởng hiệu năng do mở rộng có thể diễn ra liên tục
 - **Người quản trị** có thể giám sát và tăng size trước khi SQL Server rơi vào trạng thái sử dụng hết không gian đã được cấp
 - Bằng cách thêm file cho DB hoặc tăng size của file đang sử dụng (p.730)

Quản lý Database

- **Thu nhỏ Database (Shrinking DB)**
 - Có thể chủ động thu nhỏ kích thước các file của DB bằng DBCC SHRINKDATABASE, DBCC SHRINKDATAFILE, hay SSMS
 - Phù hợp khi di chuyển file
 - Ước lượng DB không tăng size
 - Dùng SSMS
 - Chọn database -> Tasks -> Shrink -> chọn Database hay files

Quản lý Database

- **Di chuyển Database – di chuyển vị trí file :** có thể thực hiện bằng một trong các cách sau
 - Tạo một database backup, rồi restore database ở vị trí mới
 - Thực hiện Alter database, thay đổi vị trí mới cho database file
 - Detach database , rồi reattach database ở một vị trí mới

Quản lý Database

- **Di chuyển Database – di chuyển vị trí file**
 - Di chuyển database (không thay đổi server)
 - Di chuyển database sang server khác (hay instance khác)

Quản lý Database

- Di chuyển Database (tiếp)
 - **Dùng lệnh Alter Database** với các bước sau (p.736)
 1. Take the database offline.
 2. Manually move the file(s) to the new location.
 3. Run the ALTER DATABASE command to set the FILENAME property to the new file location (các file data và log)

```
ALTER DATABASE Northwind_copy  
MODIFY FILE (NAME = Northwind,  
              FILENAME = 'C:\new_path\northwind.mdf')
```

```
ALTER DATABASE Northwind_copy  
MODIFY FILE (NAME = Northwind_Log,  
              FILENAME = 'C:\new_path\northwind_log.ldf')
```

4. Bring the database online

Quản lý Database

- Di chuyển Database (tiếp)
 - **Sử dụng Detaching và Attaching Database** dùng **SSMS**
 - Chọn Detach database (nhấn phải mouse vào tên DB cần detach)
 - Sao chép hay di chuyển các file của database tới vị trí mới
 - Chọn Attach Database (nhấn phải mouse vào Database node trong Object Explorer)=> chỉ định file .mdf của database

Detaching a database removes it from the instance of SQL Server but leaves the database intact within its data files and transaction log files. These files can then be used to attach the database to any instance of SQL Server, including the server from which the database was detached.

Quản lý Database

- Di chuyển database (tiếp)
 - Sử dụng **Detaching và Attaching Databases** dùng **t-sql** . Ví dụ :

--(1)Detach the database

EXEC master.dbo.sp_detach_db

@dbname = N'AdventureWorks2008',

@keepfulltextindexfile=N'false'

GO

Quản lý Database

- Di chuyển database (tiếp)
 - Sử dụng Detaching và Attaching Databases dùng t-sql . Ví dụ :
- (2) Di chuyển/copy các data và log file tới vị trí mới
- (3) Attach the database

```
CREATE DATABASE [AdventureWorks2008] ON  
( FILENAME = 'C:\current_path\AdventureWorks2008_Data.mdf' ),  
( FILENAME = 'C:\current_path\AdventureWorks2008_log.LDF' )  
FOR ATTACH
```


Quản lý Database

Xem thông tin

- **Sử dụng các system view:**

sysdatabases, sysObjects, systypes, syslogins, sysusers, sysmessages, syspermissions...

- Ví dụ

```
select * from sys.databases
```

```
select * from sys.sysdatabases
```

Quản lý Database

Xem thông tin

- **Sử dụng các system stored procedure**
 - **Xem thông tin** : sp_help, sp_helpdb, sp_helpfile, sp_spaceused , sp_helpfilegroup, sp_helpconstraint, sp_tables,
 - **Thực hiện tác vụ** : sp_rename, sp_renamedb,, sp_addlogin, sp_addmessage, sp_addrole ...

Ví dụ : sp_helpdb *database_name*

Quản lý database

Phát sinh scripts

- SSMS hỗ trợ phát sinh scripts
 - Với các object đang có trong database , hay là toàn bộ database , SSMS hỗ trợ phát sinh scripts (tạo mã lệnh T-SQL)
- Thao tác ?
- Sử dụng scripts khi nào ?

Tóm tắt

Bài tập 1a

- Đọc hiểu : docs.microsoft.com#**Estimate the Size of a Heap**

<https://docs.microsoft.com/en-us/sql/relational-databases/databases/estimate-the-size-of-a-heap?view=sql-server-ver15>

(*) A **heap** is a table without a clustered index

Bài tập 1b

- Đọc hiểu 2 nội dung sau. Rút ra :
 - Các nguyên tắc với file và filegroup
 - Các gợi ý để tăng hiệu năng

1

The following rules pertain to files and filegroups:

- A file or filegroup cannot be used by more than one database. For example, file sales.mdf and sales.ndf, which contain data and objects from the sales database, can't be used by any other database.
- A file can be a member of only one filegroup.
- Transaction log files are never part of any filegroups.

Bài tập 1b (tt)

Recommendations when working with files and filegroups:

- Most databases will work well with a single data file and a single transaction log file.
- If you use multiple data files, create a second filegroup for the additional file and make that filegroup the default filegroup. In this way, the primary file will contain only system tables and objects.
- To maximize performance, create files or filegroups on different available disks as possible. Put objects that compete heavily for space in different filegroups.
- Use filegroups to enable placement of objects on specific physical disks.
- Put different tables used in the same join queries in different filegroups. This step will improve performance, because of parallel disk I/O searching for joined data.
- Put heavily accessed tables and the nonclustered indexes that belong to those tables on different filegroups. Using different filegroups will improve performance, because of parallel I/O if the files are located on different physical disks.
- Don't put the transaction log file(s) on the same physical disk that has the other files and filegroups.

Bài tập 2

- Đọc hiểu và rút ra cách SQL Server ghi dữ liệu vào filegroup

<https://docs.microsoft.com/en-us/sql/relational-databases/databases/database-files-and-filegroups?view=sql-server-ver15#file-and-filegroup-fill-strategy>

Bài tập 3

- Di chuyển / sao chép một table sang filegroup khác ?

Tham khảo trang sau và tạo ví dụ minh họa ?

<https://www.mssqltips.com/sqlservertip/5832/move-sql-server-tables-to-different-filegroups/>

Bài tập 4

- Thực hiện phát sinh script cho một database tùy ý ?
- Sử dụng file script này để tạo một database bản sao (đổi tên và vị trí file lưu trữ) ?
- Thực hiện phát sinh script cho một table có data ? Sử dụng script này để tạo lại table này trong database khác ?

Bài tập 5

- Để sao chép một database sang một vị trí khác, bạn chọn phương án nào ? Giải thích tại sao (ưu/nhược điểm)?
 1. Copy database (bằng attach)
 2. Sử dụng script file để tạo lại database ở vị trí mới

Bài tập 6

- <https://www.sqlshack.com/move-a-sql-database-using-the-detach-and-attach-method/>