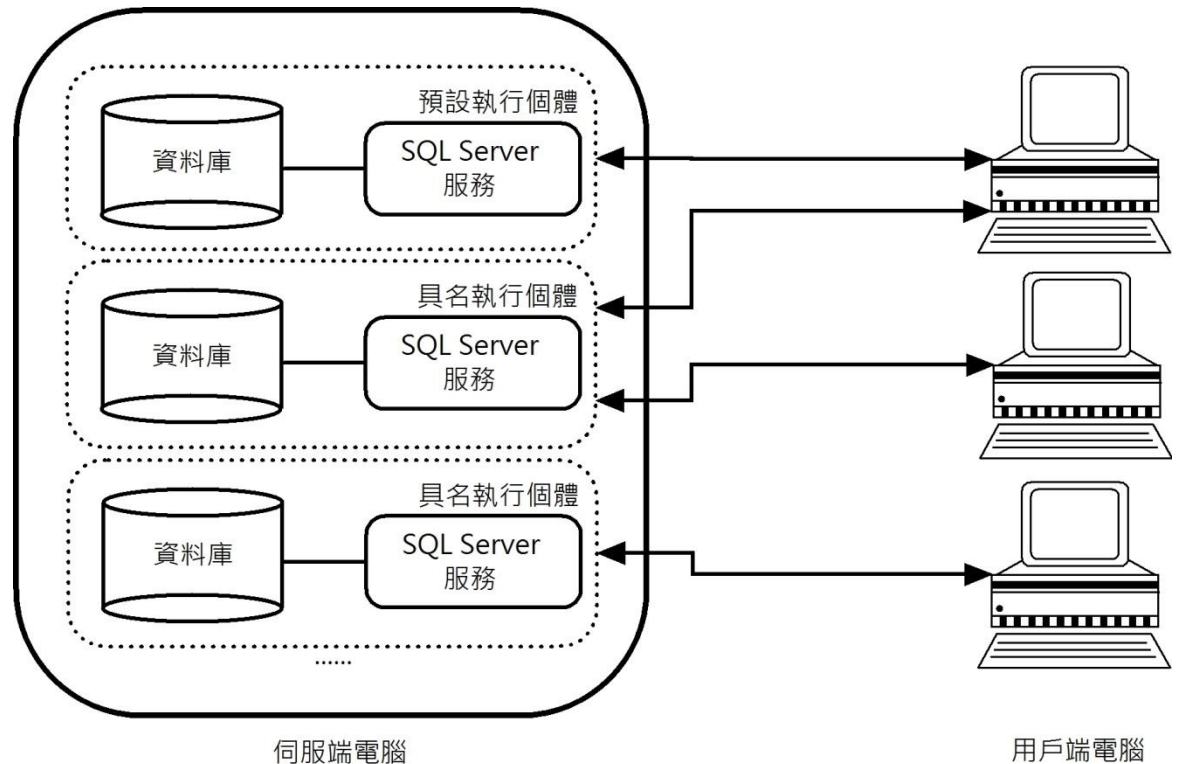


SQL Server資料庫管理系統

SQL Server的執行個體(圖例)

- SQL Server可以在同一台電腦安裝多個執行個體 (Instances)，如右圖所示：



SQL Server的執行個體(種類)

- 預設執行個體（Default Instance）：預設執行個體是一個不需名稱的執行個體，在每一台電腦只能安裝一個預設執行個體。因為是預設執行個體，用戶端只需指名電腦名稱，就可以連接預設執行個體。
- 具名執行個體（Named Instance）：在電腦安裝的SQL Server不是預設或不具名的執行個體，就是具名執行個體。我們需要替具名執行個體命名。因為同一台電腦可以安裝多個具名執行個體，所以連接SQL Server時，除了電腦名稱外，還需指明執行個體名稱。

建立資料庫

SQL Server 的內建資料庫

- **master**：master 資料庫負責記錄所有有關 SQL Server 的系統資訊, 包括：登入帳戶、系統組態設定以及目前系統中有哪些資料庫...等等，master 資料庫可以說是儲存整個 SQL Server 執行個體可以正常運作的重要資訊。如果 master 資料庫損壞，SQL Server 將無法正常的運作。
- **model**：model 是一個 "範本" 資料庫, 當新增資料庫時, SQL Server 便會以 model 資料庫為範本, 將其內容複製到新增的資料庫中, 所以 model 資料庫可說是所有資料庫最初的模型。如果更改了 model 資料庫的內容, 那麼以後新增的資料庫也會擁有相同的內容。

SQL Server 的內建資料庫

- **msdb**：msdb 是專供 SQL Server 代理程式使用的資料庫, 它存放有關警告、作業、資料備份、資料複製與資料維護等排程事宜。
- **tempdb**：tempdb 資料庫用來存放所有暫時的資料表和預存程序, 並提供 SQL Server 存放目前使用中的資料表。tempdb 是一個全域的資源, 其內的暫存資料表和預存程序可供所有使用者使用。它的容量會依需要自動成長, 但在每次 SQL Server 啟動時都會將內容全部清除, 並以預設大小重新建立。

結構化查詢語言(SQL)

- 「SQL」(Structured Query Language)的全名是結構化查詢語言，本書簡稱SQL語言。SQL語言在1980年成為「ISO」(International Organization for Standardization)和「ANSI」(American National Standards Institute)的標準資料庫語言，其版本分為1989年的ANSI-SQL 89和1992年制定的ANSI-SQL 92，也稱為SQL 2，這是目前關聯式資料庫的標準語言，ANSI-SQL 99稱為SQL 3，適用在物件關聯式資料庫的SQL語言。SQL Server 2012版的T-SQL也支援最新的ANSI-SQL 2008的特點。

SQL語言的基本語法

- SQL語言的基礎是關聯式代數和計算，SQL語法可以視為是一種關聯式計算的版本，屬於一種非程序式（Non-procedural）查詢語言，這是一種宣告語言，並不用一步一步描述執行的過程，如下所示：

```
SELECT * FROM 員工  
WHERE 薪水 >= 30000
```

- 上述SQL指令敘述查詢員工資料表中，薪水超過3萬元的員工資料，SQL指令敘述是直接告訴資料庫管理系統需要什麼樣的查詢結果，而不用詳細說明取得查詢結果的步驟。

Transact-SQL的語法元素

- 識別名稱（Identifiers）：SQL Server執行個體的資料庫物件名稱，例如：資料庫、資料表、檢視、預存程序和觸發程序等。
- 資料類型（Data Types）：指定欄位、T-SQL變數或參數可以儲存的資料內容。
- 函數（Functions）：SQL Server內建或自訂函數，例如：GETDATE()可以傳回目前的系統日期時間。
- 運算式（Expressions）：使用SQL指令敘述運算式來取得單一值，例如：欄位相加。
- 關鍵字（Keywords）：在SQL Server具有特殊意義的保留字（Reserved Words），例如：SELECT、FROM和WHERE等都是關鍵字。

SQL語言的命名原則

- 名稱長度不可超過128個字元。
- 名稱是Unicode所定義的字元，包含a到z和從A到Z的字元、數字0~9、底線或其他語系的字元，例如：中文字元。
- 名稱除第1個字元之後的字元可以是數字、@、\$、#或底線，換句話說，我們不可以使用數字、@、\$、#或底線作為識別名稱的第1個字元。
- 名稱不可以是T-SQL關鍵字，因為T-SQL不區分大小寫，所以不可以包含任何大小寫的關鍵字。
- 名稱不允許內嵌空格、關鍵字或特殊字元開頭，否則需要使用雙引號「"」或方括號「[]」將名稱括起，例如：[My School]、[1CD]和[where]等。

資料庫物件的完整名稱

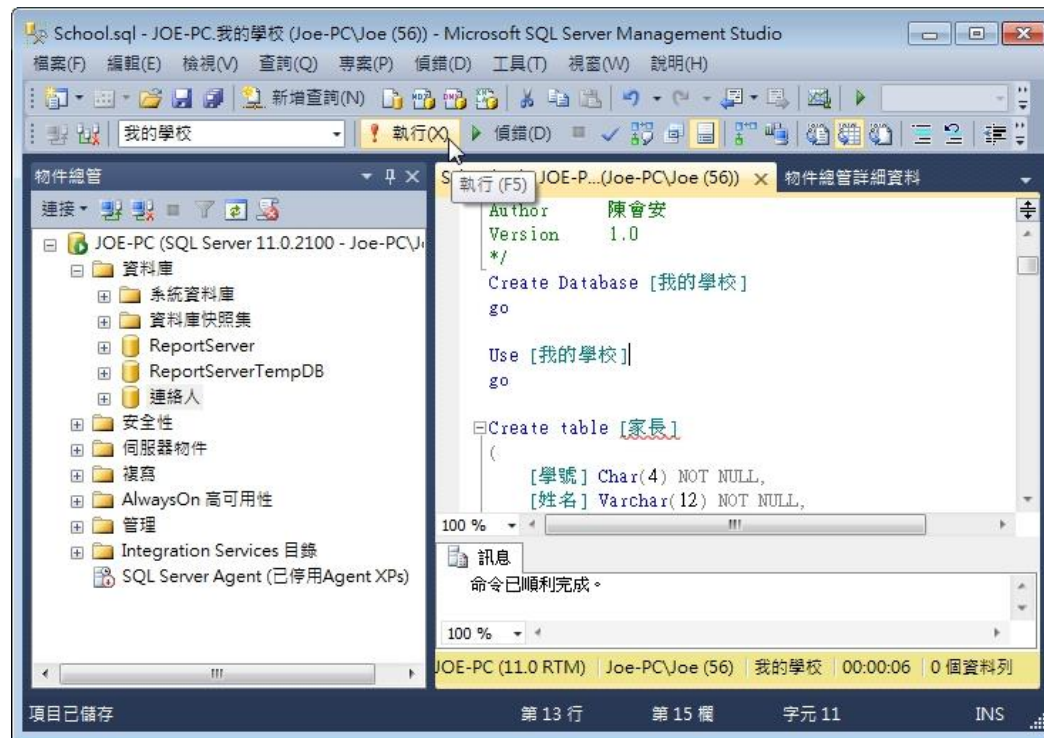
- T-SQL資料庫物件的完整名稱是由四個部分所組成，其語法如下所示：
伺服器名稱.資料庫名稱.結構描述名稱.物件名稱
- 上述名稱是使用「.»句號運算子來連接，例如：
【員工】資料表的完整名稱為【JOE-PC.我的學校.dbo.員工】。
- 在SQL Server參考指定物件時，不一定需要使用完整名稱，物件名稱只要足以讓SQL Server資料庫引擎識別即可。

SQL語言的指令種類

- 資料定義語言DDL（Data Definition Language）：DDL指令是用來建立、修改、刪除資料庫物件的資料表、檢視表、索引、預存程序、觸發和函數等。
- 資料操作語言DML（Data Manipulation Language）：DML指令是針對資料表的指令，可以插入、刪除、更新和查詢記錄資料。
- 資料控制語言DCL（Data Control Language）：資料庫安全管理的權限設定指令，主要有GRANT、DENY和REVOKE指令。

執行SQL指令碼檔案

- SQL指令碼檔案其副檔名預設為.sql。對於現存的SQL指令碼檔案，我們可以在Management Studio開啟指令碼檔案來執行SQL指令。



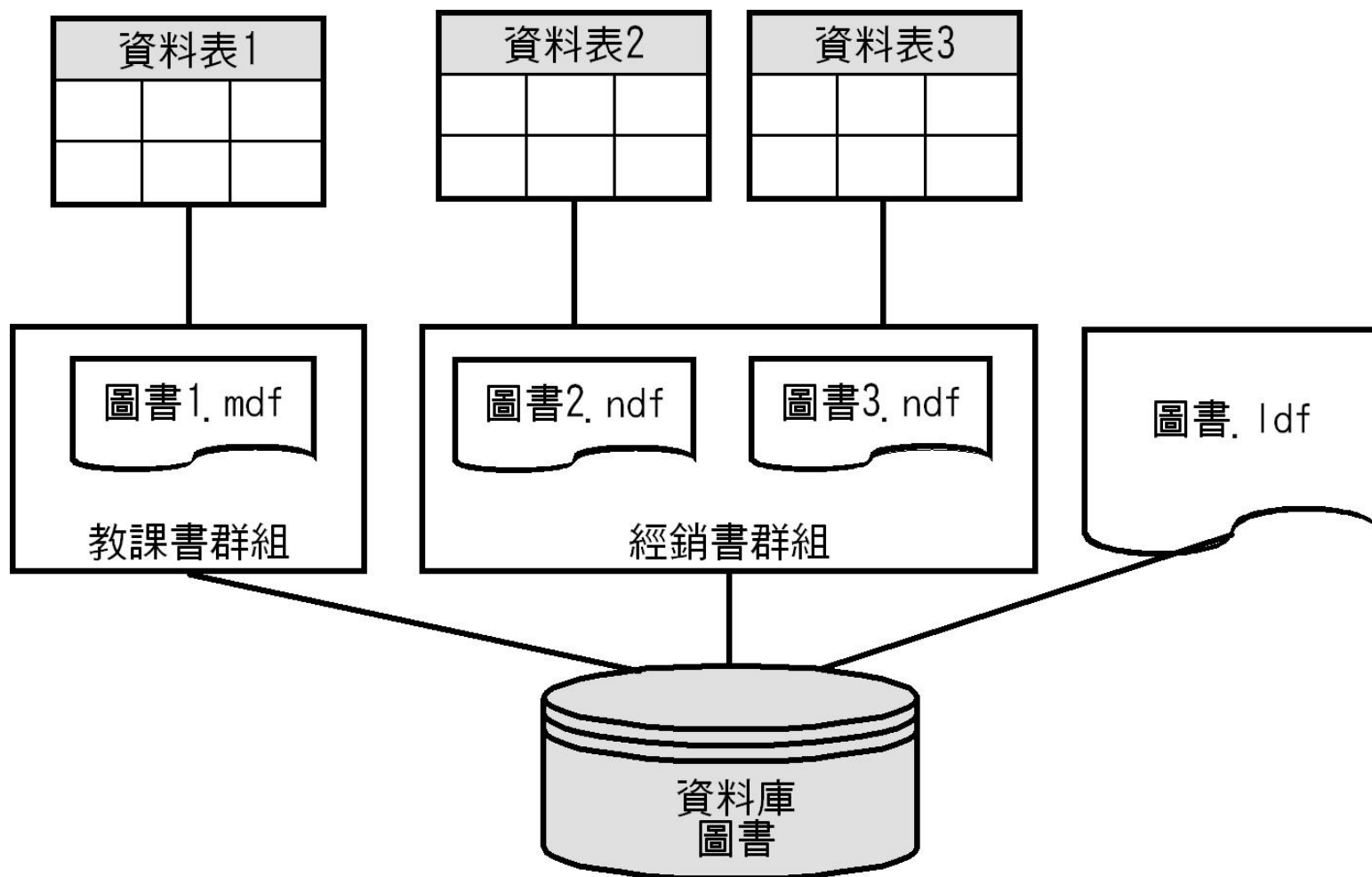
SQL Server資料庫檔案結構

- SQL Server資料庫檔案結構是安排記錄如何儲存在檔案中，不同檔案結構佔用不同大小的空間，且因結構的不同，將擁有不同的存取方式。
- SQL Server資料庫結構可以分為兩種：
 - 邏輯資料庫結構：使用者觀點的資料庫結構，SQL Server的邏輯資料庫結構是由資料表、檢視表、索引和限制條件等物件所組成。
 - 實體資料庫結構：實際儲存觀點的資料庫結構，也就是如何將資料儲存在磁碟的結構，以作業系統來說，資料庫是以檔案為單位來儲存在磁碟，檔案內容是由分頁（Pages）和範圍（Extents）所組成，為了方便管理，我們可以將檔案分類成檔案群組（Filegroups）。

資料庫檔案與檔案群組

- SQL Server資料庫是由多個作業系統檔案所組成的集合，資料庫儲存的資料（Data）和交易記錄（Log）分別位在不同檔案。在資料部分基於存取效率、備份和還原的管理上考量，可以進一步將大型資料檔（Data Files）分割成多個小型資料檔。
- 檔案群組（Filegroups）是用來組織資料庫的多個資料檔，以方便資料庫管理師來管理多個資料檔。

資料庫檔案與檔案群組



資料庫檔案與檔案群組

- 主資料檔（Primary Data Files）：資料庫儲存的資料一開始就是存入主資料檔，在主資料檔除了能夠儲存資料外，它還包含資料庫的啟動資訊，每一個資料庫都有一個且只有一個主資料檔，其建議的副檔名為**.mdf**。
- 次資料檔（Secondary Data Files）：不是主資料檔的其他資料檔稱為次資料檔，一個資料庫可能沒有任何次資料檔，也可能擁有多個次資料檔，其建議的副檔名為**.ndf**。
- 交易記錄檔（Log Files）：儲存交易記錄的檔案，每一個資料庫至少擁有一個交易記錄檔，也可能擁有多個交易記錄檔，建議副檔名是**.ldf**。

資料庫檔案與檔案群組

- 一個資料庫至少要包括一個**主資料檔**儲存資料, 以及一個**記錄檔**記錄異動資訊, 至於**次資料檔**則不一定需要。

檔 案	說 明	建議副檔名
主資料檔 Primary Data File	此檔案儲存資料庫的啓始資訊、系統資料、和一般資料。每個資料庫都必須要有一個主資料檔, 而且也只能有一個。	.MDF
次資料檔 Secondary Data File	次資料檔是用來輔助主資料檔的不足。我們可為資料庫建立多個次資料檔, 然後將資料分散存放在主資料檔與次資料檔中。	.NDF
記錄檔 Log File	此檔案會儲存資料庫中異動的日誌資訊。當資料庫發生問題時, 可利用此檔復原資料庫。一個資料庫至少要有一個記錄檔 (可以有多个), 其最小容量是 512 KB。	.LDF

資料庫檔案與檔案群組

- SQL Server資料庫如果只有一個資料檔時，我們並不需要考量檔案群組的問題。但是，對於大型資料庫，或基於管理或配置磁碟空間的考量（例如：將部分資料置於不同磁碟），而將資料庫建立成多個資料檔時，我們就可以將它們分成不同檔案群組（Filegroups），來方便資料庫檔案的管理。
- 當使用檔案群組來群組多個資料檔時，將資料存入資料檔時，就是以檔案群組為單位，而不是個別資料檔。

資料庫檔案與檔案群組

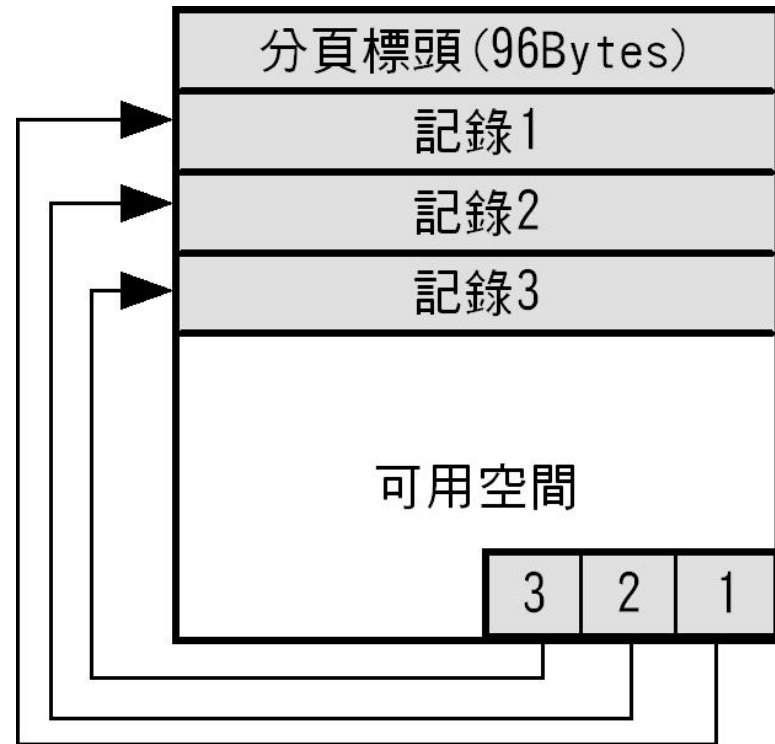
- 主檔案群組（Primary Filegroups）：這是內含主資料檔的檔案群組，它是在建立資料庫時，SQL Server預設建立的檔案群組，如果資料庫建立其他次資料檔時，沒有指定所屬檔案群組的資料檔，就是屬於主檔案群組。
- 使用者定義檔案群組（User-defined Filegroups）：使用者自行建立的檔案群組，這是使用FILEGROUP關鍵字，在T-SQL指令CREATE DATABASE或ALTER DATABASE指令所建立的檔案群組。
- 預設檔案群組（Default Filegroups）：這是資料庫預設使用的檔案群組，可以是主檔案群組或使用者定義檔案群組，如果沒有指定，預設就是主檔案群組。

分頁

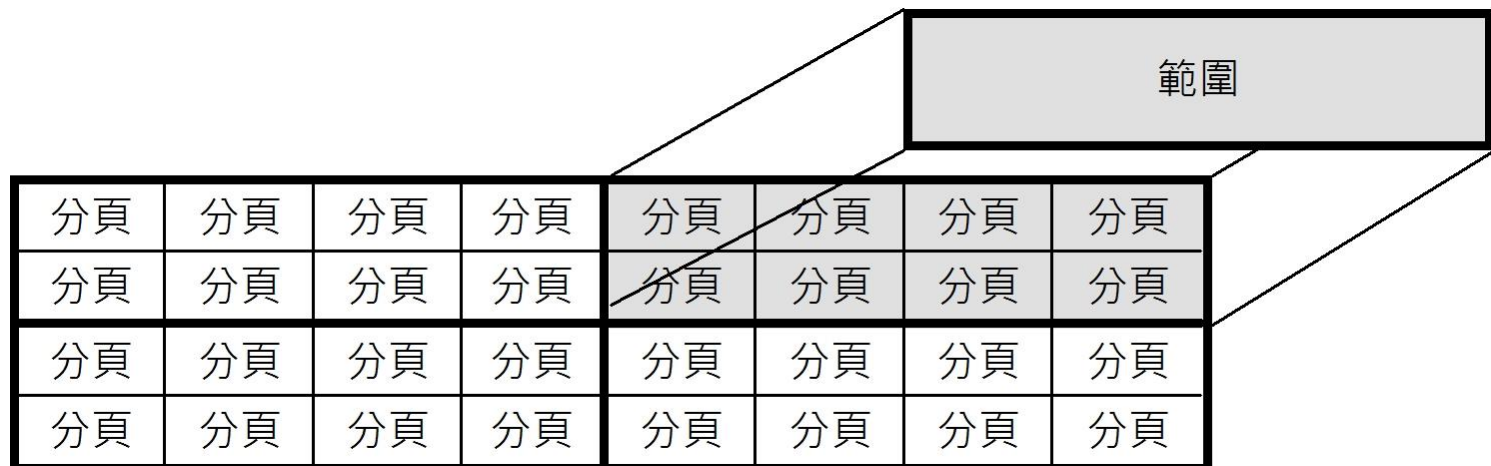
- SQL Server資料檔的內容在邏輯上是分成連續的分頁（Pages），它是SQL Server最基本的儲存單位，當資料庫配置資料檔的磁碟空間（即副檔名.mdf或ndf）時，就是配置0至n頁的連續分頁。
- 分頁（Pages）是SQL Server儲存資料的基本單位，其大小是8KB，128頁分頁等於1MB空間。當在資料檔（即副檔名.mdf或ndf）新增記錄時，如果是在空資料檔新增第1筆記錄時，不論記錄大小，SQL Server一定配置一頁分頁給資料表來儲存這筆記錄，其他記錄則會依序存入分頁配置的可用空間中。

分頁-圖例

- 分頁開始是96位元組的標頭資訊，用來儲存系統所需的相關資訊，之後依序是存入的記錄資料，在分頁的最後擁有資料列位移（Row Offsets）的指標，可以指向分頁中各記錄的開始位址。



- 範圍（Extends）是由8個連續分頁所組成，其目的是讓SQL Server可以更有效率的來管理資料檔的眾多分頁，如下圖所示：



範圍-種類

- 制式範圍（Uniform Extends）：在制式範圍中的分頁都是儲存同一個物件的資料，即完全由一個物件所使用，例如：都是配置給資料表或都配置給索引。當我們建立存在資料表的索引時，如果一建立就需要配置超過8頁分頁的索引資料，此時就是使用制式範圍來儲存索引資料。
- 混合範圍（Mixed Extends）：混合範圍中的分頁是儲存不同物件的資料，例如：部分分頁屬於資料表，部分屬於索引。一般來說，新建立的資料表或索引都是儲存在混合範圍，等到資料表或索引成長至超過8頁分頁時，就會轉成使用制式範圍來儲存。

建立資料庫

- 建立方式：
 - 使用SQL Server Mangement Studio建立
 - 使用Create DataBase敘述建立
- 注意事項
 - 資料庫命名、檔案命名、Log檔命名
 - 使用次要資料檔(檔案群組)
 - 資料庫定序
 - 資料庫初始化大小、自動成長、資料庫檔案路徑

Create DataBase敘述

```
CREATE DATABASE database_name
[ ON      [ PRIMARY ]
          [ < filespec > [ ,...n ] ]
          [ , <filegroup> [ ,...n ] ]
]
[ LOG ON { < filespec > [ ,...n ] } ]
[ COLLATE collation_name ]
[ FOR ATTACH ]
```

← 設定資料庫名稱

← 設定資料庫的檔案資料

← 設立資料庫的記錄檔

← 設定資料庫的定序

← 附加資料庫

< filespec > ::=

← < filespec > 的語法內容

```
(
    NAME = logical_file_name ,
    FILENAME = 'os_file_name'
    [ , SIZE = size ]
    [ , MAXSIZE = { max_size | UNLIMITED } ]
    [ , FILEGROWTH = growth_increment ]
) [ ,...n ]
```

< filegroup> ::=

← < filegroup> 的語法內容

```
FILEGROUP filegroup_name < filespec > [ ,...n]
```

SQL語法的符號意義

- 大寫代表關鍵字(如CREATE DATABASE)，小寫代表使用者要代換的值(如database_name)。
- 『[]』代表可省略的項目或參數。
- 『|』多個選項擇一使用，如[arg1 | arg2]，表示可用arg1或arg2。
- 『{}』表示不可省略，例如{max_size | UNLIMITED}
- 『<>』表示某一項的內容太多，必需要另外說明，例如<filespec>、<filegroup>
- 『<>::=』表示某一項需另外說明的語法如後。
- 『[,...n]』表示此項目可以設定多個，然後用逗點做分隔，例如<filespec>[,...n]表示<filespec1>,<filespec2>,.....

Create DataBase檔案規格參數

- **PRIMARY**：表示其後所定義的檔案是主資料檔。若省略此參數，則第一個定義的檔案即為主資料檔。
- **NAME = logical_file_name**：設定資料檔的邏輯檔案名稱。
- **FILENAME = 'os file name'**：設定資料檔在作業系統下的存放路徑及實際檔案名稱，存放路徑只要是在安裝 SQL Server 的那部電腦上即可。例如：**FILENAME = 'C:\DATA\ 訂單資料檔_1.MDF'**。
- **SIZE = size**：將 size 換成資料檔的起始大小，可加上單位，包括 KB、MB (預設單位)、GB、TB。次資料檔與紀錄檔的起始大小預設值為 1 MB。

Create DataBase檔案規格參數

- **MAXSIZE = max_size | UNLIMITED**：若要設定檔案的最大容量上限, 請將 max_size 換成實際的上限值, 可加上單位, 包括 KB、MB (預設單位)、GB、TB, 例如：
MAXSIZE = 10 MB。若不想設定上限, 則可省略 MAXSIZE 參數, 或設定為 MAXSIZE = UNLIMITED。
- **FILEGROWTH = grow_increment**：當檔案容量不足且尚未超過最大容量上限時, 檔案會自動成長, 此參數即用來設定每次成長的數量。成長數量可用數值或百分比來指定, 使用數值時, 可加上 KB、MB、GB、TB 單位 (預設單位為 MB)。若不想讓檔案自動成長, 請設為 0。如果省略 FILEGROWTH 參數, 則資料檔預設會以 1 MB 成長, 而記錄檔預設以 10% 來成長。若此值小於 64 KB, 則會以 64 KB 來成長。

Create DataBase檔案群組

- SQL Server 的檔案群組可分為兩種：
 - 主檔案群組：在建立資料庫時, SQL Server 會自動產生主檔案群組, 名稱就叫 PRIMARY, 而主資料檔則固定屬於主檔案群組, 不可更改。在為資料庫加入其它的次資料檔時, 若未特別指定所屬的檔案群組, 預設也會放入 PRIMARY 檔案群組中。
 - 使用者自訂檔案群組：使用者自行建立的檔案群組, 稍後即會說明建立的方法。
- 資料庫預設的檔案群組稱為預設檔案群組。
- 建立新的資料庫物件，若未指定檔案群組，該資料庫物件將建立在預設檔案群組中。

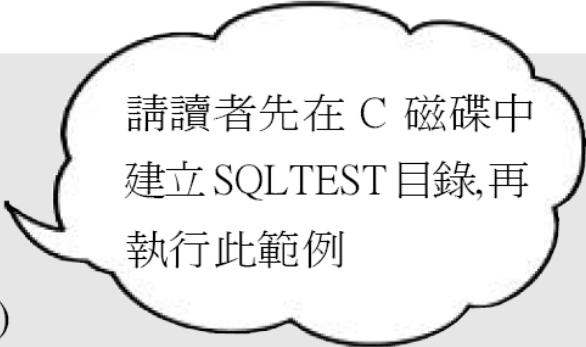
Create DataBase檔案群組

- 利用 **FILEGROUP** 參數來建立使用者自訂檔案群組。

```
FILEGROUP 訂單檔案群_1      ←——建立一個使用者自訂檔案群組
(
    NAME = 訂單資料_1,      ←——此資料檔將歸屬訂單檔案群_1 檔案群組
    FILENAME = 'E:\Program Files\訂單資料_1.NDF'
)
```

Create DataBase範例

```
CREATE DATABASE 產品資料庫  
ON  
( NAME = 產品資料庫,  
  FILENAME = 'C:\SQLTEST\產品資料庫.MDF' )
```



請讀者先在 C 磁碟中
建立 SQLTEST 目錄,再
執行此範例

```
CREATE DATABASE 機密產品資料庫  
ON  
( NAME = 機密產品資料 _1,  
  FILENAME = 'C:\SQLTEST\機密產品資料 _1.MDF',  
  SIZE = 10MB,  
  MAXSIZE = 50MB,  
  FILEGROWTH = 5 )
```


Create DataBase範例

```
CREATE DATABASE 銷售資料庫  
ONPRIMARY
```

```
( NAME = 銷售資料檔 _1, _____  
  FILENAME = 'C:\SQLTEST\銷售資料檔 _1.MDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 15% ),
```

這兩個資料檔案
屬於 PRIMARY
檔案群組

```
( NAME = 銷售資料檔 _2, _____  
  FILENAME = 'C:\SQLTEST\銷售資料檔 _2.NDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 15% ),
```

```
FILEGROUP 銷售資料庫檔案群 _1
```

```
( NAME = 銷售資料庫檔案群 _1_ 檔案 _1, _____  
  FILENAME = 'C:\SQLTEST\銷售資料庫檔案群 _1_ 檔案 _1.NDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 5 ),
```

這兩個資料檔
案屬於銷售資
料庫檔案群 _1

```
( NAME = 銷售資料庫檔案群 _1_ 檔案 _2, _____  
  FILENAME = 'C:\SQLTEST\銷售資料庫檔案群 _1_ 檔案 _2.NDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 5 ),
```

```
FILEGROUP 銷售資料庫檔案群 _2
```

```
( NAME = 銷售資料庫檔案群 _2_ 檔案 _1, _____  
  FILENAME = 'C:\SQLTEST\銷售資料庫檔案群 _2_ 檔案 _1.NDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 5 ),
```

這兩個資料檔案
屬於銷售資料庫
檔案群 _2

```
( NAME = 銷售資料庫檔案群 _2_ 檔案 _2, _____  
  FILENAME = 'C:\SQLTEST\銷售資料庫檔案群 _2_ 檔案 _2.NDF',  
  SIZE = 5, MAXSIZE = 10, FILEGROWTH = 5 )
```

設定交易記錄檔

- 指定交易記錄檔的檔案規格, 請在 LOG ON 參數中設定, 設定的語法和資料檔案一樣的, 例如：

LOGON

(NAME = 產品資料日誌 ,

FILENAME = 'C:\DATA\ 產品資料日誌.LDF' ,

SIZE = 5 MB ,

MAXSIZE = 10MB ,

FILEGROWTH = 5MB)

設定定序(Collation)

- 定序是設定資料庫所要使用的字元集 (code page) 以及字元資料排序的方式。
 - Chinese_Taiwan：表示台灣繁體中文。
 - Stroke：表示按照筆畫排序；若是 Bobomofo 則表示按注音排序。
 - CI：表示 Case Insensitive, 與大小寫無關；CS 則表示要區分大小寫。
 - AS：表示 Accent Sensitive, 要區分含重音節符號的字元；AI 則表示不區分。
- 除非有特殊需求, 例如想改用注音來排序、要區分大小寫、欄位中有他國語言的資料...等等, 否則建議保留預設的 <伺服器預設值> 設定

設定定序(Collation)

- 利用 COLLATE 參數可為資料庫個別指定定序 (若省略, 則資料庫會使用 SQL Server 的預設值), 例如：

```
COLLATE Chinese_Taiwan_Bopomofo_CI_AI
```

- 可以使用下列敘述, 列出所有的定序, 查詢要使用的定序。

```
SELECT *  
FROM ::fn_helpcollations()
```

卸離(Detach)資料庫

- 使用SQL Server Management Studio卸離資料庫
- 使用預儲程式sp_detach_db卸離資料庫

```
exec sp_detach_db [database_name]
```

附加(Attach)資料庫

- 使用SQL Server Management Studio附加資料庫。
- 使用預儲程式附加資料庫，最多只能附加16的檔案。

```
sp_attach_db [ dbname= ], filename1 [ ,...  
filename16 ]
```

範例：

```
exec sp_attach_db NorthwindC ,  
    'E:\DataBase\NorthwindC.mdf' ,  
    'E:\DataBase\NorthwindC_log.ldf'
```

附加(Attache)資料庫

- 利用Create DataBase 時的For Attache語句來附加資料庫

```
CREATE DATABASE [NorthwindC] ON  
( FILENAME = ' E:\DataBase\northwindC.mdf ' ),  
( FILENAME = ' E : \DataBase\NorthWindC_log.ldf ' )  
FOR ATTACH
```

- Microsoft建議使用Create DataBase...For Attach指令來附加卸離的資料庫，因為未來版本，將可能不再支援sp_attach_db語法。

Alter DATABASE(修改資料庫)

- ALTER DATABASE詳述請參照線上叢書
- ALTER DATABASE 敘述一次只能修改一件事。

例如：

Create DataBase TestDB

Alter DataBase TestDB Modify Name=TTDB

註：修改資料庫名稱，另可以使用sp_renamedb
例如：exec sp_renamedb 'Testdb','TTDB'

Alter DATABASE敘述

```
ALTER DATABASE database_name
{
    <add_or_modify_files>
    | <add_or_modify_filegroups>
    | <set_database_options>
    | MODIFY NAME = new_database_name
    | COLLATE collation_name
}

<add_or_modify_files>::=
{
    ADD FILE <filespec> [ ,...n ]
        [ TO FILEGROUP { filegroup_name } ]
    | ADD LOG FILE <filespec> [ ,...n ]
    | REMOVE FILE logical_file_name
    | MODIFY FILE <filespec>
}

<filespec>::=
(
    NAME = logical_file_name
    [ , NEWNAME = new_logical_name ]
    [ , FILENAME = 'os_file_name' ]
    [ , SIZE = size [ KB | MB | GB | TB ] ]
    [ , MAXSIZE = { max_size [ KB | MB | GB | TB ] | UNLIMITED } ]
    [ , FILEGROWTH = growth_increment [ KB | MB | GB | TB | % ] ]
    [ , OFFLINE ]
)
```

Alter DataBase 參數說明

- **ALTER DATABASE database** : 指定欲修改的資料庫, 請將 database 換成要修改的資料庫名稱。
- **ADD FILE < filespec > [,...n] [TO FILEGROUP filegroup_name]**

ADD FILE 參數可以為資料庫新增資料檔案, 檔案規格 < filespec > 的語法如下：

```
<filespec> ::=  
( NAME = logical_file_name  
  [ , FILENAME = 'os_file_name' ]  
  [ , SIZE = size ]  
  [ , MAXSIZE = { max_size | UNLIMITED } ]  
  [ , FILEGROWTH = growth_increment ] )
```

若不要讓新增的資料檔案放在預設檔案群組中, 則可加上 TO FILEGROUP 參數另外指定檔案群組, 將 filegroup_name 換成實際的檔案群組名稱即可。

Alter DataBase 參數說明

- **ADD LOG FILE** < filespec > [,...n] : 為資料庫新增記錄檔, 請將 < filespec > 換成實際的檔案規格。
- **REMOVE FILE** logical_file_name : 刪除資料庫的資料檔案或記錄檔, 請將 logical_file_name 換成欲刪除的邏輯檔案名稱。刪除檔案有個先決條件, 即檔案內不能包含任何資料, 否則無法刪除。
- **ADD FILEGROUP** filegroup_name : 為資料庫新增自訂的檔案群組, 請將filegroup_name 換成實際的檔案群組名稱。
- **REMOVE FILEGROUP** filegroup_name : 刪除資料庫現有的自訂檔案群組, 請將 filegroup_name 換成欲刪除的檔案群組名稱。刪除檔案群組的先決條件是, 該檔案群組內不能包含任何檔案, 否則無法刪除。

Alter DataBase 參數說明

- **MODIFY FILE** `< filespec >` : 修改資料庫檔案 (資料檔案或記錄檔) 的屬性設定, 例如 NAME、SIZE、MAXSIZE ... 等, 但一次只能更改一項。其 `< filespec >` 的內容如下：

`<filespec> ::=`

<code>(NAME = logical_file_name</code>	← 指定欲更改的檔案名稱
<code>[, NEWNAME = new_logical_name]</code>	← 設定新的邏輯檔案名稱
<code>[, SIZE = size]</code>	← 重新設定檔案大小, 其值須大於檔案目前的大小
<code>[, MAXSIZE = { max_size UNLIMITED }]</code>	← 設定新的成長上限
<code>[, FILEGROWTH = growth_increment])</code>	← 設定新的每次成長數量

查詢資料庫檔案及屬性

- 修改資料庫的檔案屬性之前, 通常會先查看資料庫目前有哪些檔案及屬性, 此時可利用預存程序 `sp_helpfile` 來查詢, 其語法如下：

```
sp_helpfile [ [ @filename = ] 'name' ]
```

←——可指定欲查詢哪一個檔案,
若未指定則表示要全部列出

查詢資料庫及檔案群組

- 查詢SQL SERVER目前有那些資料庫

```
sp_helpdb [ [ @dbname= ] 'name' ]
```

- 查詢資料庫檔案群組：
 - 執行方式和 sp_helpfile 相同, 但請記得, 需先切換為檔案所在的資料庫。

```
sp_helpfilegroup [ [ @filegroupname = ] 'name' ] ←——可指定欲查詢哪一個檔案群組
```

刪除資料庫

- 當不再需要某個資料庫的時候, 我們可以將它刪除。
 - 使用 SQL Server Management Studio 管理工具刪除資料庫
 - 刪除資料庫 DROP DATABASE 敘述