

SQL 結構化查詢語言

第二堂 SQL 的基礎與基本操作

聯成電腦

張明泰 mtchang.tw@gmail.com

複習及補充

- 上次上課內容review



MS-SQL 與 MySQL

- MS-SQL 與 MySQL 為目前國內較常見被使用的關連式資料庫管理系統(RDBMS)。
 - MS-SQL 是微軟公司出品的。
目前最新版為 sql 2005
分成4種版本 dev, ent, std, express
 - MySQL 是 MySQL AB 公司出的。
目前最新版為 mysql 5.16
分成兩種主要版本 MAXDB 及 MySQL server
-
-

MSSQL express

- <http://www.microsoft.com/downloads/details.aspx?FamilyID=220549b5-0b07-4448-8848-dcc397514b41&DisplayLang=en>
- 請參考...

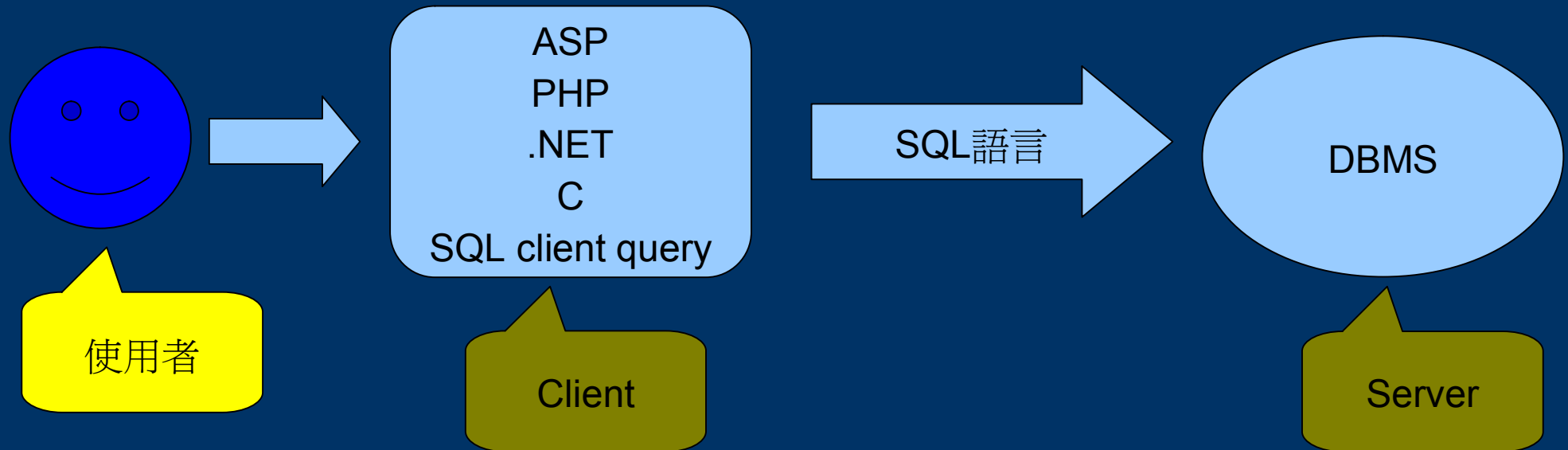


MySQL

- MySQL Administrator → 管理server
- MySQL query → 傳送sql query
- MySQL server → DBMS
- MySQL migration → 轉移工具



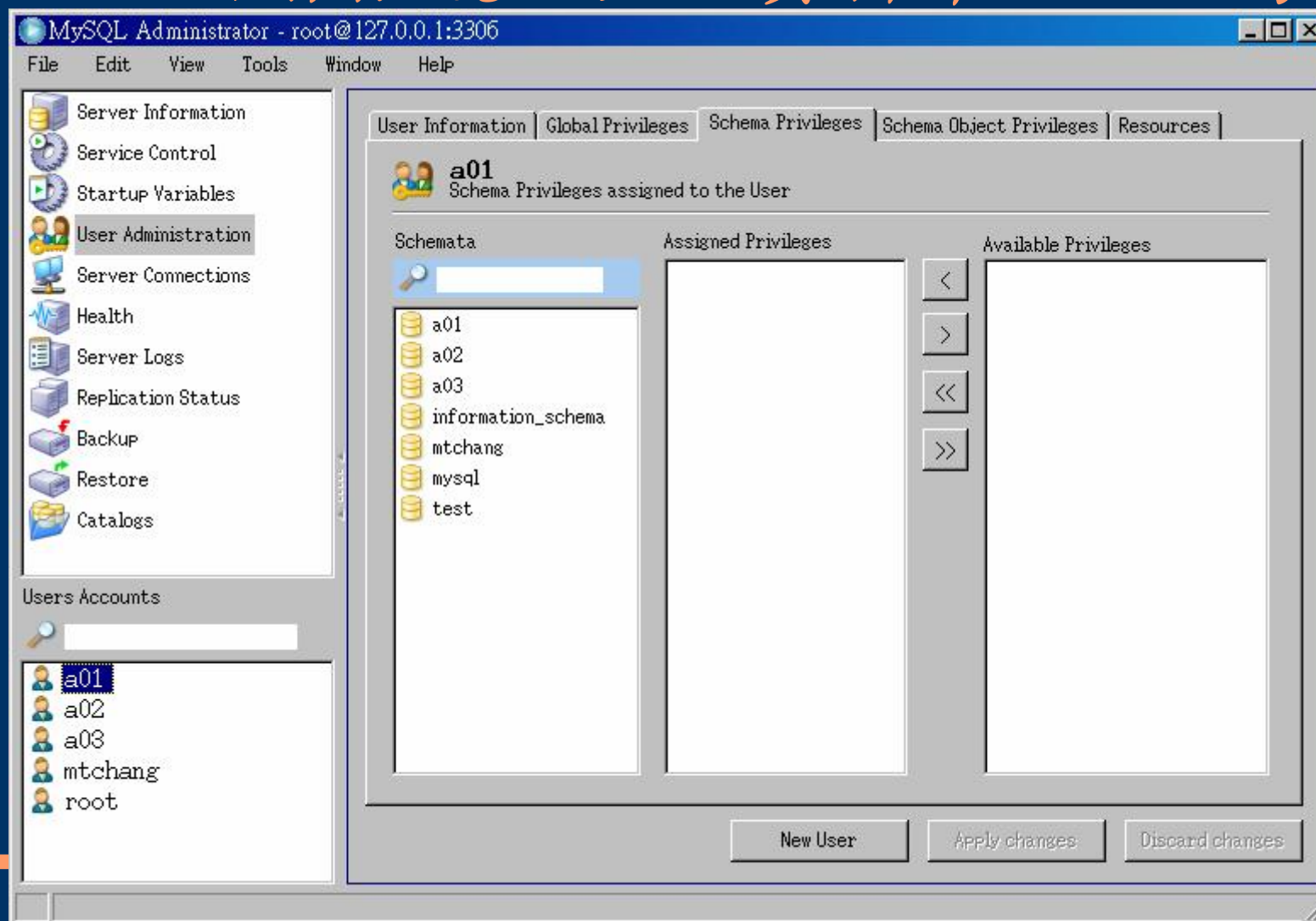
SQL 架構



MySQL 預設帳號

- 預設帳號root
 - 密碼：null
 - 預設登入來源：127.0.0.1
 - 如何觀看本身的IP：
命令提示字元
ipconfig
 - 更詳細請見
[http://linux.tnc.edu.tw/techdoc/mysql/mysql_doc
/manual_toc.html](http://linux.tnc.edu.tw/techdoc/mysql/mysql_doc/manual_toc.html)
-
-

使用MySQL Administrator建立使用者帳號及匯入資料庫-以a03為例



統一上課時的連線方式

- 各位的帳號及密碼
 - 使用XSQL tools連到我的MySQL server
 - 主機：192.168.3.xx
 - 帳號：axx
 - 密碼：axx
 - 資料庫：axx
-
-

休息一下(10 min)



提醒：請記得上網到討論區註冊



第二堂

SQL的基礎與基本操作



MySQL 的命名法則

- 在 MySQL 裡，database, table, index, column 與 alias 都依循以下的命名原則：
 - **Identifier** **Max length** **Allowed characters**
 - Database 64 除了「/」、「\」與「.»以外的其它字元
 - Table 64 除了「/」與「.»以外的其它字元
 - Column 64 所有字元
 - Alias 255 所有字元
-
-

MySQL 的命名法則

- MySQL 對關鍵字與函數名稱是不分大小寫的，SELECT、select 與 sElEcT 的作用均相同。此外，index 與 column 也是不分大小寫。
 - Alias 則是大小寫分明的。即使以相同的字母來為 alias 命名時，大小寫相異便會被視為不同的 alias。
 - database 與 table 的名稱就要視情況而定。在 Windows 作業系統下，不區分大小寫；在 UNIX 系列的作業系統下，則會區分大小寫。
-
-

試著使用SQL

- 製作資料庫或表格時，使用CREATE的命令
- 表格需指定「資料型態」，如文字、數字等...
- 為了讓在表格內的資料正確我們可以用「限制(constraints)」的方式作限制。
- CREATE-->DCL

如何操作資料庫

- 登錄資料使用INSERT敘述-DML
 - 取出資料可以使用SELECT敘述-DML
 - SELECT有附加選項(option)，可以讓使用者變更從表格中取出的資料，將資料群組化或刪除資料。
 - 此種資料庫語言更嚴謹的區分成為三種：
資料定義語言(DDL)、資料操作語言(DML)、
資料控制語言(DCL)
-
-

SQL 的分類

- 資料定義語言（Data Definition Language，DDL）
 - 可以用來建立、更改或刪除 table、schema、domain、index 與 view。
 - 主要指令有三：CREATE、ALTER 與 DROP。
- 資料操作語言（Data Manipulation Language，DML）
 - DML係用來操作資料。
 - 主要指令有四：SELECT、INSERT、UPDATE 和 DELETE。
- 資料控制語言（Data Control Language，DCL）
 - DCL 提供資料庫的安全性。
 - 主要指令有四：COMMIT、ROLLBACK、GRANT 和 REVOKE。

製作資料庫

- 語法：create database 資料庫名稱;
 - 範例：create database db_ehon;
 - 結束敘述記得加上「;」
 - 資料庫欄位及名稱有分大小寫，SQL指令並無區分。
-
-

製作表格

- 如果要製作一個有code、title、price這三列的表格，需定義：
 - 語法：create table 表格名(欄位1 資料型別,欄位2 資料型別,欄位3 資料型別.....);
 - 範例：`create table tb_ehon(code int,title varchar(30),price int);`
-
-

指定資料庫

- 語法：use 資料庫名;
- 範例：use tb_ehon;
- 說明：在使用資料前，需要先指定資料庫
- 顯示表格
- Select * from tb_ehon;
- 由於目前無資料，所以結果只顯示上方的欄位列。

限制(*constraint*)

- 為了讓登錄的資料落在正常的範圍而加上去的一些條件。
 - 範例：create table tb_ehon(code INT **UNIQUE**,title VARCHAR(30),price INT,**PRIMARY KEY**(title,price));
 - 局部範例：code **INT UNIQUE NOT NULL**
 - 說明：也可以對一欄位設定一個以上的限制。
-
-

主要的限制種類

- 限制：意義
 - **PRIMARY KEY(主鍵)**：禁止資料重複、禁止NULL值的出現。
 - **UNIQUE**：資料不得重複。
 - **CHECK(條件式)**：事先準備條件式，禁止使用不符合條件的資料。
 - **NOT NULL**：資料不得為NULL。
 - **DEFAULT=值**：將指定的值設定為預設值。
-
-

休息一下(10 min)



提醒：請記得上網到討論區註冊



限制的範例：

- 範例：`use db_ehon;`
`create table tb_height(id INT PRIMARY KEY,name VARCHAR(20),height FLOAT);`
 - 說明：限制的結果並不會顯示出來，但當妳使用新增或是修改指令時，就會知道她的功用。
 - 觀看：`select * from tb_height;`
-
-

登錄資料

- 語法：insert into 表格名 (欄位1,欄位2,欄位3)
values (資料1,資料2,資料3);
 - 範例：insert into tb_ehon(code,title,price)
values (111,'SQL繪本',300);
 - 說明：欄位1對應資料1，如果資料有字串應用'(單引號)刮起來。
-
-

取出資料

- 語法：select 欄位 from 表格名稱(資料庫.表格名稱)
 - 範例：
`select * from tb_ehon;`
`select * from db_ehon.tb_ehon;`
 - 說明：請先使用insert指令插入資料，再執行範例(此範例後面例題會用到)。
`Insert into tb_height(id,name,height) value(1,'相波',149);`
-
-

取出資料-只取出一欄位

- 語法：select 欄位1,欄位2... from 表格名;
- 範例：
select title from tb_ehon;
select title,price from tb_ehon;
- 說明：只取出title此欄位的資料。
在欄位的選項裡，顯示會照這順序來顯示。
如果以*代表欄位，所表示的是全部欄位。

ORDER BY-遞增排序

- 指定欄為基準來排序資料並顯示。
 - 語法：select 欄位 from 表格 order by 指定排序的基準欄名 ascend;
 - 範例：select * from tb_ehon order by price ascend;
 - 說明：以price欄位為基準做ASCEND遞增排序，可簡寫為ASC。
-
-

ORDER BY-遞減排序

- 指定欄為基準來排序資料並顯示。
 - 語法：select 欄位 from 表格 order by 指定排序的基準欄名 descend;
 - 範例：select * from tb_ehon order by price descend;
 - 說明：以price欄位為基準做DESCEND遞增排序，可簡寫為DESC。
-
-

比較複雜的排序

- 語法：select 欄位 from 表格 order by 基準欄位1,基準欄位2;
 - 範例：`select * from tb_ehon order by price,id;`
 - 說明：在此範例中，首先以price為基準作遞增排序，若值相同的話，則以id欄位為第二基準。
-
-

將資料群組化

- 使用GROUP BY的語法，如果欄位的資料相同時，把這些行的內容整理在一塊。通常會和合成函數一起使用。
 - 範例：`SELECT 班級座號,年月日,COUNT(公假) FROM db1.records GROUP BY 班級座號;`
 - 說明：count為合成函數功能，是會傳會所指令欄位中存在的行數。
-
-

刪除重複的資料

- 這不是指真正的刪除，是在select時將指定欄位重複項去除。
 - 範例：**SELECT DISTINCT 班級座號,年月日,公假 FROM db1.records;**
 - 說明：distinct 可以將指令欄位中重複的資料只顯示第一筆找出的資料。指令欄可以顯示一個上，但必須使用,(逗點)來區隔欄名。
-
-

變更欄位名來顯示

- 用不同的名稱顯示原來的欄位名稱。
- 範例：**SELECT 班級座號 AS class_no,年月日,公假 FROM db1.records;**
- 說明：班級座號這個欄位名稱被class_no取代了。



休息一下(10 min)



提醒：請記得上網到討論區註冊



整數型

- 資料型別(sql 99) 可使用資料範圍
使用方式(sql server)
- Integer 整數值 **INT**
- Smallint 比integer還要小的整數值
SMALLINT
- 詳細請見

[http://yes.nctu.edu.tw/SQL/DataType/Data
Type.htm](http://yes.nctu.edu.tw/SQL/DataType/Data
Type.htm)

實數型

- DECIMAL[(m[,n])] *1 使用者自訂精確度的小數值。以m表示整數位數，n表示小數點以下為數。 DECIMAL[(m[,n])]
- NUMERIC[(m[,n])] 同上 NUMERIC[(m[,n])]
- REAL 單精度浮點數 REAL
- FLOAT[(n)] 倍精度浮點數 FLOAT[(n)] *2
- DOUBLE PRECISION 倍精度浮點數
FLOAT(53)

• *1:[]的部分可以省略。 *2:n≤53。若n≤24和REAL同義。

字串型別(1)

- character[(n)]*₃ 長度固定(n位元組以內)的字串
char[(n)]*₃
- character varying(n)*₃ 長度不定(n位元組以內)的字串
varchar(n)*₃
- character large object 文章等大量的字串
TEXT
- *₃:n=<8000。[]的部分可省略。此時n=1。

字串型別(2)

- National character[(n)]*4 長度固定(n字以內)的unicode字串

nchar[(n)]*4

- National character varying(n)*4 長度不定(n字以內)的unicode字串

nvarchar[(n)]*4

- National character large object 文章等大量的unicode字串 NTEXT

- *4:n=<4000。[]的部分可省略。此時n=1。

如何取出特定的資料

- 在select敘述中可以加入where條件來取得符合條件的的資料。
- 範例：**SELECT * FROM students WHERE 學號=911013;**

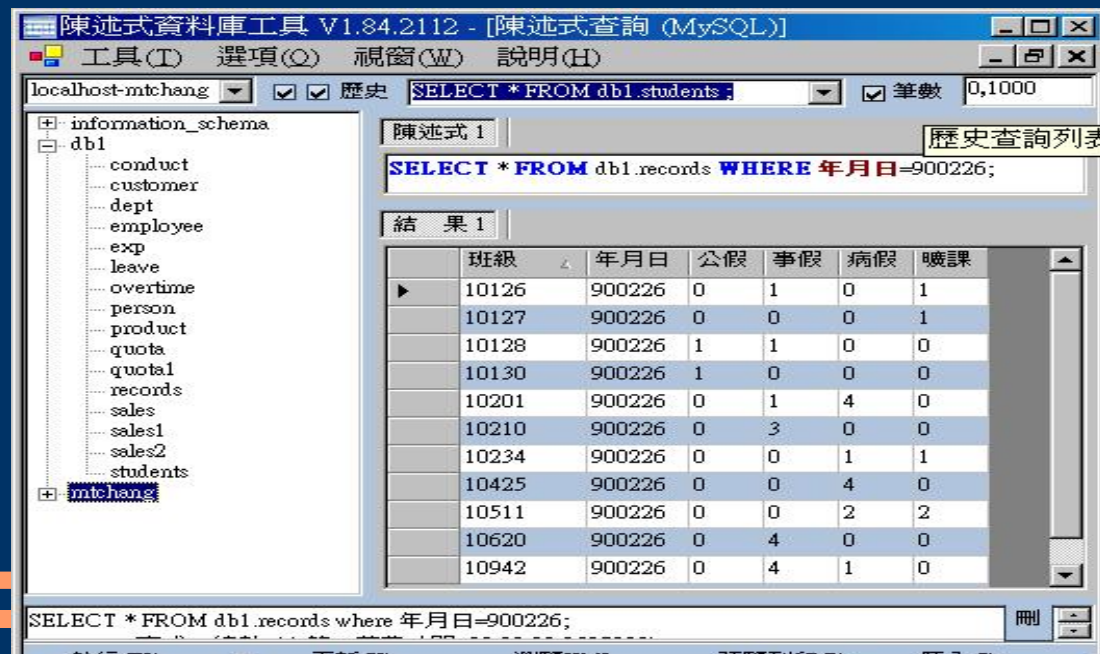
SELECT  FROM  WHERE

各種運算子

- 算術運算子(+ - x /)
- 比較運算子(> < = !=)
- 邏輯運算子(AND OR NOT)

取出符合條件的資料

- 範例：SELECT * FROM db1.records WHERE 年月日=900226;
- 語法：select 欄位名 from 表格名 where 條件式;



把符合條件的資料排序後再取出

- 範例：SELECT * FROM db1.records WHERE 年月日='900226' ORDER BY 班級座號 DESC;
- 練習1：取出沒有請過病假的學生，並依請假次數由少到多排序。

在做數學運算時的運算子

- 「+」 $a+b$
 - 「-」 $a-b$
 - 「*」 $a*b$
 - 「/」 a/b
 - 「%」 $a\%b$ a除以b取餘數
 - 範例：SELECT 班級座號,(公假+事假+病假+曠課) AS sum_day FROM db1.records;
 - 統計每個學生的請假總數
-
-

運算子指定優先順序

- 「(運算式)/2」在這範例中，刮號中的運算式具有優先計算的權力

比較運算子

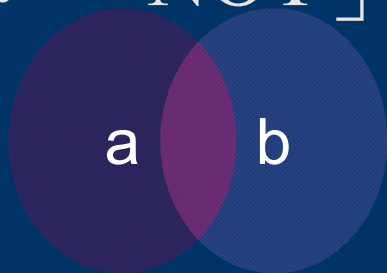
- 比較運算的結果會傳回ture或是false
 - 「=」 $a=b$ a等於b
 - 「<」 $a<b$ a小於b
 - 「<=」 $a<=b$ a小於等於b
 - 「>」 $a>b$ a大於b
 - 「>=」 $a>=b$ a大於等於b
 - 「<>」 $a<>b$ a不等於b
 - 「!=」 $a!=b$ a不等於b
-
-

條件式結果

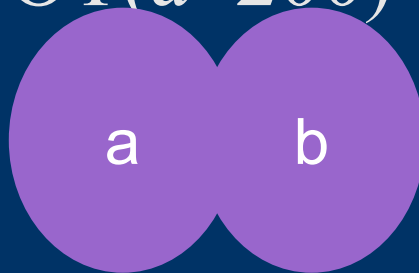
- 在經過運算後，產生的結果稱為條件式結果，一定只有ture或false兩種結果。
 - 範例：SELECT * FROM db1.records WHERE 病假 > 0;
 - 說明：列出(病假>0)的資料。ture->列出，false->不列出。
-
-

邏輯運算子

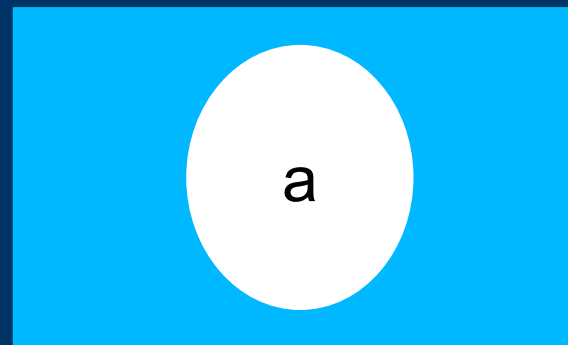
- 運用邏輯運算子可以組合多個條件式，表達更複雜的邏輯條件。
- 「AND」 且 $(a \geq 10) \text{ AND } (a \leq 20)$
- 「OR」 或 $(a = 5) \text{ OR } (a = 10)$
- 「NOT」 非 $\text{NOT}(a = 200)$



A and B



A OR B



NOT a

較複雜的運算式

- 範例：SELECT * FROM db1.records WHERE
病假 = 0 AND 公假=0 AND 事假=0；
 - 說明：找出沒有請過病、公、事假的學生。
 - 練習：找出沒有請過事假、病假及曠課的學生，或是在900201以前請假的人。
-
-

操作文字的運算子(mysql不支援)

- 此功能會因為sql版本不同，而語法不同。
- 若要連結文字的話可以使用「||」運算子。
- 範例：`SELECT 姓名 || 家長 AS name FROM db1.students;`
- SQL server範例：`select 姓名+' '+家長 from dbo.STUDENTS;`

文字運算的執行結果(SQL server)

The screenshot displays the Microsoft SQL Server Management Studio interface. The 'Object Explorer' on the left shows the database structure for '163.32.141.8 (SQL Server 8.0.194 - m...)', including 'Databases' (System Databases, Northwind, SHDB, SHWEB, mtchang) and 'Tables' (System Tables, dbo.CONDUCT, dbo.RECORDS, Columns, Keys, Constraints). The 'Query Editor' window shows the following SQL query:

```
use mtchang;  
select 姓名+' '+家長 from dbo.STUDENTS;
```

The 'Results' tab shows the execution results in a table format:

	(No column name)
1	陳忠仁 陳世傑
2	王慧如 王文淵
3	王琇榆 王進豐
4	朱宏武 朱水順
5	江新高 江明順
6	余家男 余長壽
7	余鳳如 余學義
8	吳乾君 吳 冠

The status bar at the bottom indicates the current position: 'Ready', 'Ln 1', 'Col 13', 'Ch 13', and 'INS'.

字串的比較

- 「like」運算子可以作字串比較，同常與底下符號合併使用作模糊的條件判斷。
- 「%」相當於任意字串 %明% 找出字串中有「明」的列。
- 「_」相當於任意一個字
- 範例：SELECT * FROM db1.students WHERE 姓名 LIKE '吳%';
- 練習：請找出家長姓名為「陳榮」開頭的資料。

*BETWEEN*運算子

- 使用 between~and 可以指定值的範圍
 - 範例：SELECT * FROM db1.students
WHERE 學號 BETWEEN 913024 AND
913035;
 - 練習：請使用 AND 寫成用 between 的描述。
-
-

IS NULL 運算子

- 用來檢查欄位的值是否為NULL
 - 欄位 IS NULL
 - 欄位 IS NOT NULL
 - 範例：SELECT * FROM db1.students
WHERE 學號 IS NOT NULL;
 - 範例：SELECT * FROM db1.students
WHERE 學號 IS NULL;
-
-

IN運算子

- IN用來取出()內指定的值的資料。
- 範例：SELECT * FROM db1.students
WHERE 學號 IN (911002,911005);
- 練習：請用OR或AND運算子達到一樣的功能。

指定件數取出

- 當我們的查詢一次列出很多，但是我們只要取其中列1到列10的話，可以用下列方式。
 - 每個RDBMS都有其不同的作法
 - MySQL範例：SELECT * FROM db1.students
LIMIT 0,10;
 - SQL server、Access範例：
 - PostgreSQL範例：
-
-

課後練習

- 請複習本日所教的指令用法。
- 請熟練每個範例。
- 參閱參考資料，加深印象。



參考資料

- <http://yes.nctu.edu.tw/SQL/index.htm>
 - <http://chensh.loxa.edu.tw/php/>
 - <http://ftp.tn.edu.tw/Study/SQL%ac%e3%b2%df/>
 - SQL99 手冊
-
-