

SQL 語法的使用

認識 SQL 語法

結構化查詢語言：SQL

SQL 全名是結構化查詢語言 (Structured Query Language)，是用於資料庫中的標準數據查詢語言。SQL 是目前關聯式資料庫系統所使用查詢語法的標準，使用者可以應用 SQL 語法對資料庫系統進行資料的存取、編輯、刪除及管理...等動作。SQL 語法的內容是利用簡單的英文語句所構成，使用上十分的白話。

在 phpMyAdmin 中使用 SQL 語法



定義資料庫物件語法

SQL 語法在應用上對於 DDL (Data Definition Language)：定義資料庫物件使用的語法是很基礎而重要的，其中重要的功能關鍵字有：

- CREATE：建立資料庫的物件。
- ALTER：變更資料庫的物件。
- DROP：刪除資料庫的物件。

建立資料庫

CREATE 是 SQL 指中建立資料庫或資料表的關鍵字，新增資料庫的語法如下：

```
CREATE DATABASE [IF NOT EXISTS] 資料庫名稱  
[DEFAULT] CHARACTER SET [=] 字元集  
[DEFAULT] COLLATE [=] 連線校對
```

建立資料表

新增資料表的動作更為頻繁而重要，其語法如下：

```
CREATE TABLE [IF NOT EXISTS]資料表名稱  
(欄位名稱 資料類別 [資料屬性]  
[,欄位名稱 資料類別 [資料屬性]]...);
```

ALTER：變更資料庫及資料表內容

資料庫或是資料表在新增後，可以使用 ALTER 指令語法進行修改。

修改資料庫

ALTER DATABASE 指令可以修改存在的資料庫結構，基本語法如下：

```
ALTER DATABASE 資料庫名稱  
[DEFAULT] CHARACTER SET [=] 字元集  
[DEFAULT] COLLATE [=] 連線校對
```

修改資料表欄位

ALTER TABLE 指令可以修改存在的資料表結構，基本語法如下：

```
ALTER TABLE 資料表名稱  
CHANGE 原欄位名稱 新欄位名稱 資料類別 [資料屬性]  
[,原欄位名稱 新欄位名稱 資料類別 [資料屬性]]...)
```

新增資料表欄位

ALTER TABLE 指令可以在已存在的資料表中新增資料表欄位，其語法如下：

```
ALTER TABLE 資料表名稱  
ADD 欄位名稱 資料類別 [資料屬性]  
[,ADD 欄位名稱 資料類別 [資料屬性]]...)  
[FIRST | AFTER 欄位名稱];
```

刪除資料表欄位

ALTER TABLE 指令可以在已存在的資料表中刪除資料表欄位，其語法如下：

```
ALTER TABLE 資料表名稱  
DROP 欄位名稱;
```

DROP：刪除資料庫及資料表內容

刪除資料庫與資料表的語法相似，刪除資料庫的語法如下：

```
DROP DATABASE [IF EXISTS] 資料庫名稱;
```

這個刪除資料庫的動作會連同儲存在資料庫中的所有物件，如資料表都一同刪除。

刪除資料表的語法如下：

```
DROP TABLE [IF EXISTS] 資料表名稱;
```

查詢資料庫資料的內容

SQL 語法在應用上對於 DML (Data Manipulation Language) : 查詢維護資料庫資料內容的語法在使用上是更重要的，無論是查詢資料庫或顯示資料庫的內容，更新或刪除資料庫中的資料，都必須依靠這些指令。其中重要的功能關鍵字有：

- SELECT：查詢選取資料庫中的資料。
- INSERT：新增資料到資料表中。
- UPDATE：更改資料表中的資料。
- DELETE：刪除資料表中的資料。

SELECT：查詢資料

資料的查詢應是資料庫系統最重要的工作了，所以 SELECT 可能是 SQL 語法中最重要的指令，因為所有查詢資料的動作都必須由這個指令開始。

SELECT 指令應用於使用者要向資料庫系統查詢資料的時候，其基本的語法格式如下：

```
SELECT 欄位名稱 FROM 資料表名稱;
```

WHERE：設定篩選條件

在查詢資料時，並不是每一次都要顯示所有的內容。我們可能會為顯示的資料設定一些條件，來篩選顯示的內容，這就是 WHERE 指令的功能。

WHERE 的基本語法格式為：

```
SELECT 欄位名稱 FROM 資料表名稱 WHERE 條件敘述句;
```

BETWEEN ... AND：設定篩選範圍

```
SELECT 欄位名稱 FROM 資料表名稱 WHERE 欄位名稱 BETWEEN 起始值 AND 結束值;
```

IN：指定多個篩選值

```
SELECT 欄位名稱 FROM 資料表名稱 WHERE 欄位名稱 IN (欄位值1,欄位值2,...);
```

指定的值必須放置在「()」左右括號中，每個值之間以「,」逗號區隔。

LIKE：設定字串比對的篩選值

萬用字元的使用

在文字資料中常要找以某些文字開頭、某些文字結尾，或是字串中包含哪些文字的內容，此時

即可以使用 LIKE 運算子，並搭配以下萬用字元來進行查詢：

萬用字元	說明	範例
_ (底線)	任何單一字元，一個中文字也代表一個字元。	條件：「LIKE '文_閣」，「文淵閣」符合，「文藏經閣」即不符合。
%	任何含有零或多個字元的字串。	條件：「LIKE '文%閣」，「文淵閣」符合，「文藏經閣」都符合。

ORDER BY：設定查詢結果的排序

```
SELECT 欄位名稱  
FROM 資料表名稱  
ORDER BY 指定排序的欄位 排序方式
```

其中排序方式有二種：

ASC：遞增排序，由小排到大，也是未指定時預設的排序方法。

DESC：遞減排序，由大排到小。

多欄位排序

```
ORDER BY 指定排序的欄位1 排序方式1, 指定排序的欄位2 排序方式2, ....
```

LIMIT：設定查詢顯示的筆數

LIMIT 可以設定查詢後由哪一筆開始顯示，並顯示多少筆數，它的基本語法如下：

```
SELECT 欄位名稱  
FROM 資料表名稱  
LIMIT 開始顯示的筆數, 顯示多少筆資料
```

LIMIT 是由查詢後的結果再進行擷取資料的動作，如果與 ORDER BY 進行排序搭配可以輕易取得最前的 10 筆資料或是最後的 10 筆資料的結果。也因為如此，LIMIT 在使用時必須放置在 ORDER BY 之後。

MySQL 常用函式

MySQL 資料庫系統本身提供了許多函式可供使用者搜尋資料時進行相關的處理，以下將分成幾個較常使用的函式類別加以介紹。

算術運算子與數學函式

面對數值資料欄位，可以使用算術運算子及數學函式對於欄位的值進行處理成為查詢結果。

算術運算子

+ - * /

常用數學函式

- ABS(數值)
- POW(基數, 次方數)
- SQRT(數值)
- PI()
- ROUND(數值 [,小數位數])

日期時間函式

函式	說明
NOW()	返回目前日期時間，格式為「YYYY-MM-DD HH:MM:SS」，「YYYYMMDDHHMMSS」或時間戳記，依需求返回。
CURDATE();	返回目前系統年月日。 也可寫為 CURRENT_DATE(), CURRENT_DATE。
CURTIME();	返回目前系統時分秒。 CURRENT_TIME(), CURRENT_TIME。
YEAR(日期時間)	返回指定時間的年份，範圍在 1000 到 9999。
MONTH(日期時間)	返回指定時間的月份，範圍在 1 到 12。
DAY(日期時間)	返回指定時間的日期，在 1 到 31 範圍內。
DATE(日期時間)	返回指定時間的年月日。
TIME(日期時間)	返回指定時間的時分秒。
HOUR(日期時間)	返回指定時間的小時，範圍在 0 到 23。
MINUTE(日期時間)	返回指定時間的分鐘，範圍在 0 到 59。
SECOND(日期時間)	返回指定時間的秒數，範圍在 0 到 59。
DAYNAME(日期時間)	返回指定時間的星期名，如 Sunday、Monday 等。
MONTHNAME(日期時間)	返回指定時間的月份名，如 January、February 等。
QUARTER(日期時間)	返回指定時間一年中的季度，範圍 1 到 4。
DAYOFWEEK(日期時間)	返回指定時間的星期索引 (1=週日, 2=週一, ..., 7=週六)。
DAYOFMONTH(日期時間)	與 DAY() 相同，返回指定時間的日期，在 1 到 31 範圍內。
DAYOFYEAR(日期時間)	返回指定時間在一年中的日數，在 1 到 366 範圍內。
LAST_DAY(日期時間)	返回指定時間該月的最後一天，一般可用來判斷是否為閏年。
TO_DAYS(日期時間)	返回指定時間由西元元年到目前的天數。

統計函式

- SUM()

- AVG()
- COUNT()
- MAX()
- MIN()
- GROUP BY
- HAVING : GROUP BY的條件式

新增、更新與刪除資料

查詢雖然是資料庫中重要的功能，但是新增、更新與刪除資料的動作，才是維護資料庫內容的主要核心功能，以下將說明 SQL 語法中新增、更新與刪除資料的指令與語法。

INSERT：新增資料

可以使用 INSERT 語法為資料表新增資料，其基本語法如下：

```
INSERT [INTO] 資料表名稱 [(欄位名稱1, 欄位名稱2, ...)]  
VALUES (值 1, 值 2, ...);
```

UPDATE：更新資料

使用 UPDATE 語法為資料表更新資料，其基本語法如下：

```
UPDATE 資料表名稱  
SET 欄位名稱1 = 值 1, 欄位名稱2 = 值 2, ...  
WHERE 條件式;
```

UPDATE 更新資料的動作可以一次更動多筆資料的內容，所以 WHERE 後加上的條件式十分重要，只要符合條件的資料內容即會進行更新的動作，要特別注意。

DELETE：刪除資料

使用 DELETE 語法為資料表刪除資料，其基本語法如下：

```
DELETE FROM 資料表名稱 WHERE 條件式;
```

DELETE 刪除資料的動作可以一次刪除多筆資料的內容，所以 WHERE 後加上的條件式十分重要，只要符合條件的資料內容即會進行刪除的動作，要特別注意。

多資料表關聯查詢

除了在一個資料表中選取欄位進行查詢，我們也可以在多個資料表中之中選取不同的欄位，進行查詢的動作。這樣的查詢方式是必須有前提的，那就是資料表之間要有一欄可以指定相關或是建立關聯。

若要結合二個資料表的基本語法如下：

```
SELECT 顯示欄位...  
FROM 資料表A, 資料表B  
WHERE 資料表 A.相關欄位 = 資料表 B.相關欄位
```

使用 JOIN 結合資料表

若要 JOIN 語法結合二個資料表的基本語法如下：

```
SELECT 顯示欄位...  
FROM 資料表A [INNER] JOIN 資料表B  
ON A.相關欄位 = 資料表 B.相關欄位
```

另一個方式為：

```
SELECT 顯示欄位...  
FROM 資料表A [INNER] JOIN 資料表B  
USING (相關欄位)
```

LEFT JOIN

LEFT JOIN 可以用來建立左外部連接，查詢的 SQL 敘述句 LEFT JOIN 左側資料表 (table_name1) 的所有記錄都會加入到查詢結果中，即使右側資料表 (table_name2) 中的連接欄位沒有符合的值也一樣。

語法

```
SELECT table_column1, table_column2...  
FROM table_name1  
LEFT JOIN table_name2  
ON table_name1.column_name=table_name2.column_name;
```

客戶資料表 customers

C_Id	Name	City	Address	Phone
1	張一	台北市	XX路100號	02-12345678
2	王二	新竹縣	YY路200號	03-12345678
3	李三	高雄縣	ZZ路300號	07-12345678

產品訂單的資料表 orders

O_Id	Order_No	C_Id
1	2572	3
2	7375	3
3	7520	1
4	1054	1
5	1257	5

我們想查詢所有客戶與其訂單狀況的資料，我們可以作一個 LEFT JOIN 查詢

```
SELECT customers.Name, orders.Order_No
FROM customers
LEFT JOIN orders
ON customers.C_Id=orders.C_Id;
```

LEFT JOIN 會返回左側資料表中所有資料列，就算沒有符合連接條件，而右側資料表中如果沒有匹配的資料值就會顯示為 NULL。

RIGHT JOIN

相對於 LEFT JOIN，RIGHT JOIN 可以用來建立右外部連接，查詢的 SQL 敘述句 RIGHT JOIN 右側資料表 (table_name2) 的所有記錄都會加入到查詢結果中，即使左側資料表 (table_name2) 中的連接欄位沒有符合的值也一樣。

語法

```
SELECT table_column1, table_column2...
FROM table_name1
RIGHT JOIN table_name2
ON table_name1.column_name=table_name2.column_name;
```

我們想查詢所有訂單與相應的客戶之資料，我們可以作一個 RIGHT JOIN 查詢

```
SELECT customers.Name, orders.Order_No
FROM customers
RIGHT JOIN orders
ON customers.C_Id=orders.C_Id;
```

RIGHT JOIN 會返回右側資料表中所有資料列，就算是沒有符合連接條件，而左側資料表中如果沒有匹配的資料值就會顯示為 NULL。