

## 第二章 MySQL 教學範本

<a href="#">2.1 伺服器連線與離線</a>	3
<a href="#">2.1.1 連線</a>	3
<a href="#">2.1.2 離線</a>	4
<a href="#">2.2 指令編輯方式</a>	4
<a href="#">2.3 資料庫管理指令</a>	5
<a href="#">2.3.1 建立資料庫</a>	5
<a href="#">2.3.2 顯示資料庫明細</a>	5
<a href="#">2.3.3 選用資料庫</a>	5
<a href="#">2.3.4 刪除資料庫</a>	5
<a href="#">2.3.5 備份資料庫</a>	6
<a href="#">2.3.6 重載資料庫</a>	6
<a href="#">2.4 資料表管理</a>	6
<a href="#">2.4.1 建立資料表</a>	6
<a href="#">2.4.2 顯示資料表明細</a>	7
<a href="#">2.4.3 顯示資料表結構</a>	7
<a href="#">2.4.4 建立暫時性資料表</a>	8
<a href="#">2.4.5 刪除資料表</a>	8
<a href="#">2.4.6 修改資料表</a>	8
<a href="#">2.4.7 索引表管理</a>	8
<a href="#">2.5 載入資料</a>	8
<a href="#">2.5.1 批次載入：限定固定格式文字檔</a>	8
<a href="#">2.5.2 逐筆輸入</a>	9
<a href="#">2.6 查閱資料</a>	9
<a href="#">2.6.1 查看全部資料</a>	10
<a href="#">2.6.2 查看部份資料</a>	10
<a href="#">2.6.3 紀錄排序</a>	13
<a href="#">2.6.4 函數型SQL指令</a>	14
<a href="#">2.6.5 統計函數</a>	15
<a href="#">2.6.6 限定輸出筆數</a>	16
<a href="#">2.7 關聯查詢</a>	16
<a href="#">2.7.1 鍵值關聯</a>	16
<a href="#">2.7.2 Where關聯</a>	16
<a href="#">2.8 子查詢</a>	17
<a href="#">2.8.1 依第 2 表之結果來過濾第 1 表</a>	17
<a href="#">2.8.2 將挑選第 2 表資料新增至第 1 表</a>	17
<a href="#">2.8.3 依第 2 表挑出之資料將第 1 表對應資料刪除</a>	17

<a href="#">2.8.4 以IN運算子比對資料</a>	17
<a href="#">2.9 刪除資料</a>	17
<a href="#">2.10 更新資料</a>	18
<a href="#">回目錄</a>	

## 第二章 MySQL 教學範本

本章將以本書第十七章進銷存管理系統當範例，介紹 MySQL 資料庫常用語法和操作指令，透過這些範例之實作，初次接觸 MySQL 之讀者可以很快的認識 MySQL。基本上，這些範例都是運用【mysql】主控端應用程式，以交談方式和伺服器溝通。所謂交談式，是指使用者在 DOS 環境下直接下命令，伺服器接到命令後接著處理對應資料，然後將處理結果送回主控端之螢幕上，處理過程是一個命令對應一個結果。本章將以教學導向，指令力求簡單，若需了解各指令之詳細語法，請參考本書附錄之 MySQL 資料庫操作指令，若需了解相關資料表結構，請讀者參閱本書第十七章。

### 2.1 伺服器連線與離線

操作資料庫，首要工作就是連上伺服器。假如資料庫是多人共用者，不再處理資料時，或是需要長時間離開座位的話，請記住養成隨時離線的習慣，除了可減輕連線負荷外，也可提高資料庫之安全。本節將介紹連線與離線之操作命令。

#### 2.1.1 連線

連線指令常用型式有下列三種方式：

指令一：`mysql -h Host -u User -pPassword`

指令二：`mysql -h Host -u User -p`

Enter password: \*\*\*\*\*

指令三：`mysql`

前述指令之意義，其中“-h”代表指定連線主機參數，Host 代表主機名稱，可用數字碼，或是網域名稱。“-u”代表帳號參數，User 為使用者帳號。“-p”表示密碼參數，Password 為使用者自己設定之密碼，“-p”和密碼間不能有空格。指令一之密碼直接在“-p”之後輸入，輸入之密碼將直接顯示在螢幕上。指令二之密碼在輸入“-p”之後，按 Enter 鍵，然後輸入密碼，輸入之密碼將被“\*”遮蔽，因此較為安全。若使用者是在伺服器之主機上操作，則“-h Host”部分可省略，伺服器自動認定為近端主機（localhost）。指令三適用於伺服器允許以匿名方式連線者。

連線成功後，螢幕將回應下列信息：

[↑ H](#)

```
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 1 to server version: 4.1.9-nt.  
Type 'help' or '\h' for help. Type '\c' to clear the buffer.  
mysql>
```

圖 2.1 連線成功畫面

連線失敗時，將出現類似下列信息：

```
ERROR 1045 (28000) Access denied for 'user'@'localhost'  
(using password No)  
mysql>
```

圖 2.2 連線失敗畫面

### 2.1.2 離線

離線指令有下列兩種型式：

指令一： QUIT

指令二： \q

離線後，螢幕將出現下列信息：

```
mysql>QUIT  
Bye
```

圖 2.3 離線畫面

## 2.2 指令編輯方式

指令編輯型式有下列兩種：

1. 一行指令一道命令：指令編輯完成後，直接按【Enter】鍵執行命令。
2. 數行指令一道命令：指令佔滿一行時，按【Enter】鍵換行（可不按），繼續編輯後續指令。

指令列若為標準 SQL 指令，必須加上 “;”，然後按【Enter】鍵結束。不管指令列跨越幾行，一道命令只能包含一個 “;”。指令列若為 MySQL 內部指令，則不用加上 “;”，例如 Use, Quit 等。圖 2.4 提示各種有效指令之編輯形式。

[↑ H](#)

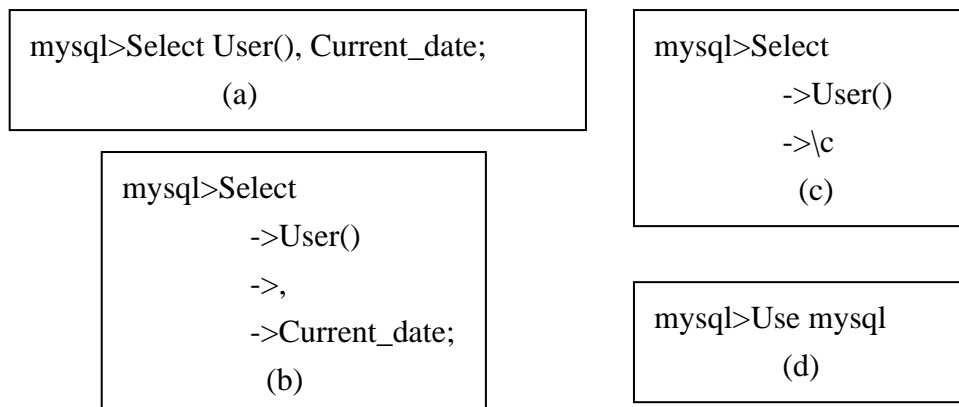


圖 2.4 各種有效指令型式：(a) 一行指令一道命令，(b) 數行指令一道命令，(c)終止指令執行，(d) MySQL 內部指令（無“;”號）

讀者若需處理中文字元，請在連線後執行下列指令：

```
Set character_set_client = big5;  
Set character_set_results=big5;  
Set character_set_connection =big5;
```

## 2.3 資料庫管理指令

### 2.3.1 建立資料庫

語法：Create Database 資料庫名稱;

範例：Create Database mysal; 建立一個名稱為【mysal】之資料庫。

說明：前述指令亦可搭配檢查語法，以防資料庫被覆寫，指令如下：

```
Create Database If Not Exists mysal;
```

### 2.3.2 顯示資料庫明細

語法：Show Databases;

執行結果：如圖 2.5。

### 2.3.3 選用資料庫

語法：Use 資料庫名稱

範例：Use Mysql; 螢幕會提示：Database Changed 之信息，如圖 2.6。

### 2.3.4 刪除資料庫

語法：Drop Database 資料庫名稱;

範例： Drop Database Temp;

```
mysql> show databases;
+-----+
| Database |
+-----+
| mycto    |
| myeqp    |
| myerp    |
| mymtl    |
| mysal    |
| mysql    |
| mysys    |
| temp     |
| test     |
| ??s -mysql |
+-----+
10 rows in set (0.05 sec)

mysql>
```

圖 2.5 顯示資料庫明細

```
mysql> use mysql
Database changed
mysql>
```

圖 2.6 選用資料庫範例

### 2.3.5 備份資料庫

可使用主控端應用程式 - mysqldump 來備份資料庫 (在 DOS 主控台執行)，指令語法如下：

指令 1:

mysqldump 目標資料庫 -u 帳號 -p > 備份檔案名稱

指令 2 (不備份資料)

mysqldump 目標資料庫 -d -u 帳號 -p > 備份檔案名稱

指令 3 (不加註解):

mysqldump 目標資料庫 --comment=0 -u 帳號 -p > 備份檔案名稱

指令 4 (包含資料庫建檔指令):

mysqldump 目標資料庫 -d -u 帳號 --databases -p > 備份檔案名稱

### 2.3.6 重載資料庫

登入 MySQL 伺服器後，將備份檔重新載入資料庫，指令語法如下：

#### \. 備份檔案名稱

說明：備份檔必須包含完整路徑，且路徑符號必須使用“/”。

## 2.4 資料表管理

### 2.4.1 建立資料表

語法：Create Table 資料表名稱 (欄位定義);

[↑ H](#)

範例：CREATE TABLE cuinfo (

```
CU_No varchar(10) NOT NULL DEFAULT "",
CU_Name varchar(26) NOT NULL DEFAULT "",
CU_Tel varchar(12) NOT NULL DEFAULT "",
CU_Mtel varchar(12) DEFAULT NULL,
CU_Email varchar(35) DEFAULT NULL,
CU_Staf varchar(20) NOT NULL DEFAULT "",
CU_Adrs varchar(60) NOT NULL DEFAULT "",
CU_Txno varchar(10) DEFAULT NULL,
ML_Type varchar(3) DEFAULT NULL,
PRIMARY KEY (CU_No)
```

) ENGINE=MyISAM DEFAULT CHARSET=big5;

說明：欄位定義之格式為，欄位名稱 資料型態 (寬度) [Null | NOT Null] [Default 值]，欄位間須以“,”間隔。此範例用於建立客戶資料表。

#### 2.4.2 顯示資料表明細

語法：Show Tables;

執行結果：如圖 2.7。

#### 2.4.3 顯示資料表結構

語法：Describe 資料表名稱;

範例：Describe cuinfo;

執行結果：如圖 2.8。

```
mysql> use mysal
Database changed
mysql> show tables
-> ;
+-----+
| Tables_in_mysal |
+-----+
| actionproject   |
| alarmgroup      |
| backtable       |
| barcodefmt      |
| bbkiclog        |
| bkichl          |
| bkiclog         |
| bkspbl          |
| bksplog         |
| blobfile        |
| bom             |
| classinfo       |
| comport         |
| cuinfo          |
| cuquoat         |
```

圖 2.7 顯示資料表範例

```
mysql> describe cuinfo;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CU_No | varchar(10) | NO | PRI | | |
| CU_Name | varchar(26) | NO | | | |
| CU_Tel | varchar(12) | NO | | | |
| CU_Mtel | varchar(12) | YES | | NULL | |
| CU_Email | varchar(35) | YES | | NULL | |
| CU_Staf | varchar(20) | NO | | | |
| CU_Adrs | varchar(60) | NO | | | |
| CU_Txno | varchar(10) | YES | | NULL | |
| ML_Type | varchar(3) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.01 sec)
```

圖 2.8 顯示資料表結構範例

### 2.4.4 建立暫時性資料表

語法：Create Temporary Table 資料庫名稱 (欄位定義)；

說明：資料表將建立在記憶體內，資料處理完後可予以刪除，離線後該表將自動被刪除。暫存表可用於整理過渡資料。

### 2.4.5 刪除資料表

語法：Drop Table 資料表名稱；

### 2.4.6 修改資料表

此部份請參閱本書 4.5.6 節說明。

### 2.4.7 索引表管理

#### 1. 建立索引表

語法：Create Index 索引表名稱 On 資料表名稱 (欄位名稱)；

範例：CREATE Index cuinfo1 On cuinfo (CU\_No)；

#### 2. 顯示索引表

語法：Show Index From 資料表名稱；

範例：Show Index From bom；

結果如圖 2.9。

MySQL定義									
	Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub par	Packed
▶	bom	0	PRIMARY	1	PD_No	A			
	bom	0	PRIMARY	2	PT_No	A	3		
	bom	1	bom1	1	PD_No	A			
	bom	1	bom2	1	PT_No	A			
*									

圖 2.9 顯示索引表資訊

## 2.5 載入資料

### 2.5.1 批次載入：限定固定格式文字檔

事前將對應檔案上傳至伺服器端，然後使用下述指令將文字檔內容載入資料表。

指令：Load Data Local Infile “外部檔案來源” Into Table 資料表名稱 Fields Terminated By “,” [Line Terminated By “\r\n”]；

[↑H](#)



範例：Load Data Local Infile “C:\\mysql\\data\\test\\cuinfo.csv” Into Table  
cuinfo Fields Terminated By “,”;

說明：文字檔內容必須是以列為單位之固定格式，欄位間須以逗號或空格間隔開來，若以空格間隔，需記住空格個數。編輯檔案時，若每一列是以'\r\n'符號換行的話，則輸入命令需加上[Line Terminated By '\r\n']參數。一筆紀錄若有部分欄位為空白資料時，對應欄位須加上['\N']記號，MySQL 會將此記號解譯為 Null。文字檔之欄數須和資料表欄數相同，否則會產生錯誤。若資料是從主控端上傳伺服器端，則須加上[Local]參數，檔案可置於其它路徑，命令列之檔案來源須加上檔案路徑。

### 2.5.2 逐筆輸入

指令一：Insert Into 資料表名 Values (欄位資料串);

指令二：Insert Into 資料表名 (欄位串列) Values (欄位資料串);

範例：Insert Into cuinfo Values ('A001', '聯興公司', '037-123456',  
'0937-123456', 'carlu@nuu.com.tw', '張先生', '台北市', '123456',  
'RPT');

說明：若所有欄位都需要輸入資料時，使用指令一，否則使用指令二。指令二需要指定欄名，欄名間以逗號間隔。除了數值資料外，欄位資料須以雙引號或單引號夾住，欄位資料若為空值，請填入 Null。

## 2.6 查閱資料

從資料表取出資料之形式千變萬化，但基本的指令格式如下：

```
SELECT 欄位串列  
FROM 資料來源  
WHERE 資料範圍  
ORDER BY 欄位串列  
GROUP BY 欄位串列;
```

其中，

SELECT：用來指定輸出欄位名稱，可用\*表示全部欄位，欄名間使用逗號間隔。

FROM：用來指定資料來源，可同時查看兩個以上資料表。

WHERE：用來設定輸出資料範圍，常見格式為指定欄名，再搭配關係運算子（諸如=、>、>=、<、<=、<>(代表不等於)），然後輸入待比對之欄位條件值。若有需要，還可以設定第二個比較欄位，但兩個欄位間要用邏輯運算子結合，諸如 AND、OR。

ORDER BY：用來指定資料要以升冪（修飾辭為ASC）或降冪 [↑ H](#)

(修飾辭為 DESC) 順序輸出。需要排序之欄位，請在 BY 後面定義，資料排序方式內定為升冪形式，ASC 修飾辭可省略。若為降冪排列，DESC 修飾辭不能省略。同時指定兩個以上欄位排序時，欄位間須以逗號間隔，排序順序可不相同。

GROUP BY：用來指定群組欄位，若資料表內相同鍵值紀錄有多筆情況，而輸出資料須予以加總、平均、計算筆數等時，須指定群組欄位。群組欄位可指定一個以上，欄位間須以逗號間隔。

以下資料測試畫面，是作者使用 EasyDoMySQL 連線至 MySQL 之測試結果，因此會和讀者在 DOS 視窗看到之畫面有所差異。

### 2.6.1 查看全部資料

指令：SELECT \* FROM 單一資料表；

範例：SELECT \* FROM cuinfo;

執行結果：如圖 2.10。

	客戶編號	公司名稱	電話	行動電話	E-mail	聯絡人
▶	A001	聯興公司	037-123456	0937-123456	carlu@nuu.edu.tw	張先生
	A002	聯茂公司	02-12345678	0937-123456	A001@mail	張先生
	A003	交通大學	035-123456	0937-123456	carlu@nuu.edu.tw	李先生
	A004	聯合大學	037-381500	381501	nuu@nuu,email,tw	李先生
*						

圖 2.10 查看全部資料

### 2.6.2 查看部份資料

#### 1. 簡單過濾條件

指令：SELECT \* FROM 單一資料表 WHERE 資料範圍；

範例：SELECT \* FROM cuinfo WHERE cu\_no='A001'；

執行結果：如圖 2.11。

	客戶編號	公司名稱	電話	行動電話	E-mail	聯絡人
▶	A001	聯興公司	037-123456	0937-123456	carlu@nuu.edu.tw	張先生
*						

圖 2.11 查看部份資料(1)

#### 2. 界限型過濾條件

範例：SELECT \* FROM cuinfo WHERE cu\_no >="A003"；

執行結果：如圖 2.12。

[↑H](#)

	客戶編號 ▲	公司名稱	電話	行動電話	E-mail
▶	A003	交通大學	035-123456	0937-123456	carlu@nuu.edu.tw
	A004	聯合大學	037-381500	381501	nuu@nuu,email,tw
*					

圖 2.12 查看部份資料(2)

### 3. 複式過濾條件(And)

範例：SELECT \* FROM bkiclog WHERE bk\_blno='951101' And pd\_no='PD001';

執行結果：如圖 2.13。

	單號 ▲	產品編號	數量	金額	稅額	處理註記	退貨註記
▶	951101	PD001	200	3000	200		N
*							

圖 2.13 查看部份資料(3)

### 4. 複式過濾條件(Or)

範例：SELECT \* FROM bkiclog WHERE bk\_blno='951101' Or pd\_no='PD001';

執行結果：如圖 2.14。

	單號 ▲	產品編號	數量	金額	稅額	處理註記	退貨註記
▶	951101	PD001	200	3000	200		N
	980201	PD001	95	1000	100	N	Y
	980202	PD001	200	3000	200		N
	991101	PD001	200	3000	200		N

圖 2.14 查看部份資料(4)

### 5. 查看指定欄位資料

範例：Select BK\_Blno, PD\_No, BK\_Qty From bkiclog;

執行結果：如圖 2.15。

### 6. 查看特定欄位且排除重覆資料

範例：Select Distinct PD\_No From bkiclog;

執行結果：如圖 2.16。

BK_Bln	PD_No	BK_Qty
951101	PD001	200
951102	PD002	300
951103	PD003	100
951108	PD01	110
980201	PD001	95
980202	PD001	200
991101	PD001	200
*		

圖 2.15 查看指定欄位資料

PD_No
PD001
PD002
PD003
PD01
*

圖 2.16 查看唯一資料

## 7. 比對樣板值-前置樣板

範例：Select \* From cuinfo Where CU\_Name LIKE "聯%";

執行結果：如圖 2.17，

說明：LIKE 為條件值比較運算子，字母"聯%"表示所有以聯開頭之資料均符合。

CU_No	CU_Name	CU_Tel	CU_Mtel
A001	聯興公司	037-123456	0937-123456
A002	聯茂公司	02-12345678	0937-123456
A004	聯合大學	037-381500	381501
*			

圖 2.17 以樣板值當過濾條件(1)

## 8. 比對樣板值-後置樣板

範例：Select \* From cuinfo Where CU\_Name LIKE "%學";

執行結果：如圖 2.30，

說明：%學表示所有字串尾有學字之資料均符合。

CU_No	CU_Name	CU_Tel	CU_Mtel
A003	交通大學	035-123456	0937-123456
A004	聯合大學	037-381500	381501
*			

圖 2.18 以樣板值當過濾條件(2)

## 9. 比對樣板值-中置樣板

[↑ H](#)

範例：Select \* From cuinfo Where CU\_Name LIKE "%合%";

執行結果：如圖 2.19，

說明：%合%表示所有字串中有合字之資料均符合。

	CU_No	CU_Name	CU_Tel	CU_Mtel
▶	A004	聯合大學	037-381500	381501
*				

圖 2.19 以樣板值當過濾條件(3)

### 2.6.3 紀錄排序

#### 1. 升冪

範例：Select \* From bkiclog Order By BK\_Blno;

執行結果：如圖 2.20。

	BK_Blno	PD_No	BK_Qty
▶	951101	PD001	200
	951102	PD002	300
	951103	PD003	100
	951108	PD01	110
	980201	PD001	95
	980202	PD001	200
	991101	PD001	200
*			

圖 2.20 輸出資料排序—升冪

	BK_Blno	PD_No	BK_Qty
▶	991101	PD001	200
	980202	PD001	200
	980201	PD001	95
	951108	PD01	110
	951103	PD003	100
	951102	PD002	300
	951101	PD001	200
*			

圖 2.21 輸出資料排序—降冪

#### 2. 降冪

範例：Select \* From bkiclog Order By BK\_Blno DESC;

執行結果：如圖 2.21。

#### 3. 混合排序

範例：Select \* From bkiclog Order By BK\_Blno,PD\_No DESC;

執行結果：如圖 2.22。

	BK_Blno	PD_No	BK_Qty
▶	951101	PD001	200
	951102	PD002	300
	951103	PD003	100
	951108	PD01	110
	980201	PD001	95
	980202	PD001	200
	991101	PD001	200
*			

[↑H](#)

圖 2.22 輸出資料排序—混合排序

## 2.6.4 函數型 SQL 指令

### 1. 日期欄位計算 (1)

範例：Select BK\_Blno, cu\_no, BK\_Date, DL\_Name, curdate()AS Now,  
(year(curdate())- year(BK\_Date))- (right(curdate(),5)  
<right(BK\_Date,5)) As Year from bkicbl;

執行結果：如圖 2.23。

說明：計算結果另列一欄，可使用 AS 修飾辭指定欄名，本例欄名為  
“Now”及“Year”。指令中之 curdate(),year(), right() 為 MySQL 內  
建日期與字串函數。

	BK_Blno	cu_no	BK_Date	DL_Name	Now	Year
▶	951107	A007	2010/2/18	Da Jong	2010/10/4	0
	951101	A001	2009/2/19	Da Jong	2010/10/4	1
	951106	A001	2010/2/18	Da Jong	2010/10/4	0
	951103	A003	2010/2/18	Da Jong	2010/10/4	0
	951102	A002	2009/2/19	Da Jong	2010/10/4	1
	951108	A003	2010/4/18	Da Jong	2010/10/4	0
	991101	A001	2010/7/12	Da Jong	2010/10/4	0
*						

圖 2.23 日期欄位經計算後輸出資料(1)

### 2. 日期欄位計算 (2)

範例：Select \* From Bkicbl where Month(BK\_Date) =2;

執行結果：如圖 2.24，

說明：此例以函數取值並據以當過濾條件。除日期欄位外，數值欄位也經  
常透過運算式來取得所需資料。

	BK_Blno	cu_no	BK_Date	DL_Name
▶	951107	A007	2010/2/18	Da Jong
	951101	A001	2009/2/19	Da Jong
	951106	A001	2010/2/18	Da Jong
	951103	A003	2010/2/18	Da Jong
	951102	A002	2009/2/19	Da Jong
*				

圖 2.24 以函數計算欄資料(2)

### 3. 紀錄筆數計算 (1)

範例：Select Count(\*) AS 筆數 From bkiclog ;

執行結果：如圖 2.25，

[↑ H](#)

說明：Count()為計算紀錄數之函數，\*為統計對象。

	筆數
▶	7
*	

圖 2.25 統計資料筆數(1)

#### 4. 紀錄筆數計算 (2)

範例：Select PD\_No, Count(\*) AS 筆數 From bkiclog Group By PD\_No;

執行結果：如圖 2.26，

說明：Count()函數搭配其它欄位使用時，必須加上 GROUP BY 修飾辭來使用。

	PD_No	筆數
▶	PD001	4
	PD002	1
	PD003	1
	PD01	1
*		

圖 2.26 統計資料筆數－  
搭配 GROUP BY(2)

#### 2.6.5 統計函數

範例：Select Count(\*) As RecNo, Sum(BK\_Qty) As SUM, Avg(BK\_Qty) As AVG, Std(BK\_Qty) As STD, StdDev\_Samp(BK\_Qty) As SSTD, Max(BK\_Qty) As MAX, Min(BK\_Qty) As MIN From bkiclog

執行結果：如圖 2.27。

	Item	Value
▶	筆數	7
	合計	1205
	平均	172.1429
	標準差	69.3777
	樣本標準差	74.9365
	最大值	300
	最小值	95
*		

圖 2.27 統計資料

說明：

Count：計算筆數，

Sum：計算總和，

AVG：計算平均，

Std：計算母體標準差，

StdDev\_Samp：計算樣本標準差，

Max：計算最大值，

Min：計算最小值。

### 2.6.6 限定輸出筆數

語法：SELECT \* FROM 資料表 Limit m, n;

說明：m 表起始紀錄編號，從 0 起算（若缺設則表從 0 開始）；n 表輸出筆數。

範例 1：SELECT \* FROM cuinfo Limit 1;（輸出第一筆紀錄）

範例 2：SELECT \* FROM cuinfo Limit 5,5;（從第六筆開始，共輸出五筆紀錄）

## 2.7 關聯查詢

### 2.7.1 鍵值關聯

語法：Select 欄位串列 From 主表

Inner Join 子表

On 主表鍵值欄 1=子表鍵值欄 1 [ And 主表鍵值欄 2=子表鍵值欄 2]

Where ...;

範例：Select bkspl.BK\_Blno, bkspl.CU\_No, bkspl.BK\_Date,  
bkspl.PD\_No, bkspl.BK\_Qty, bkspl.DL\_Name From bkspl  
Inner Join (bkspl) On (bkspl.BK\_Blno = bkspl.BK\_Blno);

執行結果：如圖 2.28。

說明：Inner Join 用於主表與子表鍵值完全吻合，Inner 可省略，

Left Join 主表輸出全部紀錄，子表為鍵值吻合者，

Right Join 子表輸出全部紀錄，主表為鍵值吻合者。

退貨單號	客戶編號	退貨日期	產品編號	數量	送貨公司
951101	A001	2006/11/14	PD003	1000	
951101	A001	2006/11/14	PD004	500	
951102	A002	2006/11/15	PD001	500	
951102	A002	2006/11/15	PD002	500	
951103	A003	2006/11/14	PD001	200	
951103	A003	2006/11/14	PD002	200	
951104	A004	2006/11/14	PD001	325	
951104	A004	2006/11/14	PD002	1000	

圖 2.28 關聯查詢- 透過鍵值欄

### 2.7.2 Where 關聯

[↑ H](#)



語法：Select 欄位串列 From 主表

Inner Join 子表

Where 主表鍵值欄 1=子表鍵值欄 1 [ And...]

範例：Select bkspbl.BK\_Blno, bkspbl.CU\_No, bkspbl.BK\_Date,  
bksplog.PD\_No, bksplog.BK\_Qty, bkspbl.DL\_Name From bkspbl  
Inner Join (bksplog)  
Where bkspbl.BK\_Blno = bksplog.BK\_Blno;

執行結果：如圖 2.28。

## 2.8 子查詢

所謂子查詢是將一道查詢指令之輸出結果，直接引用在 Where 當欄位比對條件，此種用法在 MySQL 4.1 版以後才加上的，是一個非常具有彈性的指令。常見的用法摘述如下：

### 2.8.1 依第 2 表之結果來過濾第 1 表

指令格式：Select \* From t1 Where column1 = (Select column1 From t2);

### 2.8.2 將挑選第 2 表資料新增至第 1 表

指令格式：Insert Into t1 (欄位串列) Select (欄位串列) From t2 Where (t2 之過濾條件);

說明：兩個資料表之欄位串列之個數必須相同，對應型態也須相同。

### 2.8.3 依第 2 表挑出之資料將第 1 表對應資料刪除

Delete From t1 Where column1 = (Select column1 From t2);

### 2.8.4 以 IN 運算子比對資料

假如主資料表（在子查詢前指定之資料表）需要以兩個或以上之欄位與子查詢比對資料時，可使用下述指令：

指令格式： Select column1,column2,column3 From t1  
Where (column1,column2,column3) IN  
(Select column1,column2,column3 From t2);

說明：前述指令之 "IN" 限定辭，也可改用 "NOT IN"，表示不在子查詢內的資料。

## 2.9 刪除資料

刪除資料之基本指令型式如下：

DELETE

[↑ H](#)

FROM 資料來源

WHERE 資料範圍；

刪除資料不需指定欄位串列，因為刪除資料時，是將整筆紀錄刪除，而不能刪除部份欄位。通常刪除資料之時機為，例如學生之繳費檔，已繳費者可予以刪除，尚未繳費者則予以保留，此時刪除指令必須搭配 WHERE 來使用。若一個資料表之資料有期限者，當期限終了就須將整個資料表資料刪除，此時不需搭配 WHERE 過濾條件。請讀者注意，刪除資料和刪除資料表是不同性質的，刪除資料時，資料表結構還留存在資料庫內，但刪除資料表卻是將整個資料表從資料庫移除。

請讀者自行練習下列刪除資料範例，每執行一道刪除指令，接著執行後續之查詢指令，查看資料變動情形。

1. Delete From bkiclog Where bk\_blno <'951101';  
Select \* From bkiclog;
2. Delete From bkiclog;  
Select \* From bkiclog;

假如你需要透過第二個資料表來挑選待刪除資料，此時需將指令改成下述型式。DELETE 之後須接資料表名稱，指定刪除資料對象。

3. DELETE bkicbl FROM bkicbl, bkiclog WHERE bkicbl.bk\_blno =  
bkiclog.bk\_blno ;

一道刪除指令可同時刪除兩個資料表的資料，指令格式如下述：

4. DELETE bkicbl, bkiclog FROM bkicbl, bkiclog WHERE bkicbl.bk\_blno =  
bkiclog.bk\_blno ;

## 2.10 更新資料

更新資料之基本指令型式如下：

UPDATE 資料表名稱

SET 欄位名稱 = 更新資料, [後續更新運算式]

WHERE 資料範圍；

更新資料之時機，常見情況有下列幾種：

1. 原輸入資料錯誤，事後才發現錯誤，因此需要更新。
2. 時效過期。
3. 異動檔資料轉入主控檔（例如會計作業之過帳）。

請讀者自行練習下列更新資料範例，每執行一道更新指令，接著 [↑H](#)

執行後續之查詢指令，查看資料變動情形。

1. 將代號為 A001 之公司名稱清空

```
Update cuinfo Set cu_name=NULL Where cu_no='A001';
```

2. 將報價全部調高 10%。

```
Update cuquoat Set un_price = un_price*1.1;
```

3. 出貨轉應收帳

```
Create Temporary Table updTable
```

```
Select  spbill.CU_No, Sum(splist.SP_Amt + splist.TX_Amt) AS Amount,  
        splist.TR_Note From spbill Join splist Where spbill.SP_Blno = splist.SP_Blno  
        And splist.TR_Note<>"T" Group By spbill.CU_No;
```

說明：此指令用於建立暫存表：updTable，欄位由後續SQL指令產生。

```
Lock Tables updTable write, spbill write, splist write,rcpay Write, rcpay As  
objTable Write;
```

說明：轉檔前需將相關資料表上鎖。

```
Insert Into rcpay (CU_No) Select CU_No From updTable
```

```
Where (CU_No) Not In (Select CU_No From rcpay As objTable);
```

說明：此指令用於查閱應收帳款之鍵值欄是否有遺漏，有則補登之。

```
Update rcpay, updTable
```

```
Set rcpay.CR_Spamt = rcpay.CR_Spamt + updTable.Amount Where  
rcpay.CU_No= updTable.CU_No;
```

說明：此指令用於將出貨金額轉入應收帳款。

```
Update splist Set splist.TR_Note ="T" Where splist.TR_Note<>"T";
```

說明：此指令用於將已轉檔出貨紀錄加上註記。

```
Unlock Tables;
```

說明：此指令用於將資料表解鎖。

```
Drop Temporary Table updTable;
```

說明：此指令用於刪除暫存表。

## 習題

1. 簡述 mysql 應用程式之用途。
2. 寫出 MySQL 伺服器連線與離線指令。
3. 任舉 MySQL 處理資料庫、資料表和資料之指令各三個。
4. 任意修改某資料表結構，修改內容為：在指應欄位後加入一個新欄位，型態為字串，寬度為 20 位元組。列出新結構後，將新增欄刪除。
5. 任意增加一個索引定義在某資料表，列出索引明細後，將新增索引刪除。
6. 簡述輸出兩個資料表紀錄之指令語法，兩種以上。
7. 說明資料需要更新之時機。

[↑ H](#)

[回目錄](#)