

MySQL-操作練習

先啟動 xampp 的 MySQL 伺服器：建議使用 xampp-control.exe 來開啟 MySQL。

使用 MySQL client 來連接與操作 MySQL 資料庫：

- 1) 啟動"命令列模式"(console mode)：視窗鍵 + R，再輸入 cmd。
- 2) 直接切換至 c:\xampp\mysql\bin 目錄，或者
使用 path=c:\xampp\mysql\bin;%PATH% 來建立路徑連結。

```
mysql\bin> mysql -u UserName -p <== 此處不要輸入密碼以免洩漏！
Enter password: <== 此時再輸入密碼！
```

完成登入MySQL server後，可以先看看目前MySQL中有多少資料庫：

```
mysql> SHOW DATABASES;
```

資料庫與表格的建立順序為：

先建立資料庫(database) => mysql> CREATE DATABASE myDatabaseName;

其次要選擇使用該資料庫 => mysql> USE myDatabaseName;

在myDatabaseName資料庫中建立一個新的資料表 => mysql> CREATE TABLE myTable;

如果要使用當中的某個資料庫，例如：dbxyz，則可以輸入下列指令：

```
mysql> USE dbxyz; // database 的名稱 dbxyz 會區分大小寫！
```

接著大家應該會好奇說這資料庫當中有多少表格，模仿查看資料庫的方式，大家應該能猜出指令的大概樣子：

```
mysql> SHOW TABLES;
```

強烈建議同學先完全仿照下列步驟執行一次，因為這個例子是經過設計的，目的在幫同學建立一個通盤的概念！

底下的示範碼僅作為操作過程的參考，其中的輸出結果可能會因使用的環境不同而有差異！

(*下列資料庫實作參考 "多人連線遊戲程式設計開發手冊", Andrew Mulholland & Teijo Hakala, PG20165, 博碩文化)

// 如果擁有創建資料庫權限的話，例如以 root 登入，請用下列指令：

```
mysql> CREATE DATABASE IF NOT EXISTS mydb;
```

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mydb |
| test |
+-----+
3 rows in set (0.00 sec)
```

```
mysql> USE mydb;
```

// 如果沒有創建資料庫權限的話，請改使用 test 資料庫：

```
mysql> USE test;
```

```
Database changed
```

```
mysql> SHOW TABLES;
Empty set (0.00 sec)
```

底下的指令示範如何從無到有地建立一個新的 playerinfo 資料表：

```
mysql> create table playerinfo (
  -> id INT AUTO_INCREMENT,
  -> name VARCHAR(30),
  -> xpos INT,
  -> ypos INT,
  -> exp INT,
  -> PRIMARY KEY (id) );
```

查看上面所建立資料表的欄位訊息(亦可用 SHOW COLUMNS FROM playerinfo;)：

```
mysql> describe playerinfo;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(30)   | YES  |     | NULL    |                |
| xpos  | int(11)       | YES  |     | NULL    |                |
| ypos  | int(11)       | YES  |     | NULL    |                |
| exp   | int(11)       | YES  |     | NULL    |                |
```

```
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

插入一筆資料，並要依序給予所有欄位的資料值：

```
mysql> INSERT INTO playerinfo VALUES
-> (NULL, 'Jeff Henderson', 14, 35, 1431);    // NULL欄位表示讓系統自動幫我們填上資料，此欄為編號。
Query OK, 1 row affected (0.02 sec)

mysql> select * from playerinfo;
+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

一次插入多筆資料，並依序給予所有欄位的資料值：

```
// 依指表格欄位順序來插入資料：
mysql> INSERT INTO playerinfo VALUES
-> (NULL, 'Sandra Smith', 21, 86, 4563),
-> (NULL, 'John Baker', 10, 5, 231);
Query OK, 2 rows affected, 2 warnings (0.00 sec)
Records: 2  Duplicates: 0  Warnings: 2
```

```
// 以指定不同欄位順序的方式來插入資料：
mysql> INSERT INTO playerinfo (id, name, exp, xpos, ypos) VALUES
-> (NULL, 'Jennifer West', 5400, 73, 13);
```

查看資料庫的全部或部分內容：

```
mysql> select * from playerinfo;
+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
| 2  | Sandra Smith   | 21   | 86   | 4563 |
| 3  | John Baker     | 10   | 5    | 231  |
| 4  | Jennifer West  | 13   | 73   | 5400 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select name, exp from playerinfo;
+-----+-----+
| name           | exp |
+-----+-----+
| Jeff Henderson | 1431 |
| Sandra Smith   | 4563 |
| John Baker     | 231  |
| Jennifer West  | 5400 |
+-----+-----+
4 rows in set (0.00 sec)
```

查詢同時還可以變換資料排列的順序：(上例為：name, exp，本例是：exp, name)

```
mysql> select exp, name from playerinfo;
+-----+-----+
| exp | name           |
+-----+-----+
| 1431 | Jeff Henderson |
| 4563 | Sandra Smith   |
| 231  | John Baker     |
| 5400 | Jennifer West  |
+-----+-----+
4 rows in set (0.00 sec)
```

查詢同時還可以變換資料顯示抬頭(title)的名稱：(exp AS `經驗值`，name AS `姓名`)

```
mysql> select exp AS `經驗值`, name AS `姓名` from playerinfo;
+-----+-----+
| 經驗值 | 姓名           |
+-----+-----+
| 1431   | Jeff Henderson |
| 4563   | Sandra Smith   |
| 231    | John Baker     |
| 5400   | Jennifer West  |
+-----+-----+
```

```
+-----+-----+
4 rows in set (0.00 sec)
```

設定查詢資料的條件：**where exp>1000;**

```
mysql> select * from playerinfo where exp>1000;
+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
| 2  | Sandra Smith   | 21   | 86   | 4563 |
| 4  | Jennifer West  | 13   | 73   | 5400 |
+-----+-----+-----+-----+
3 rows in set (0.03 sec)
```

設定查詢資料的條件：**where ypos=86;**

```
mysql> select id, name from playerinfo where ypos=86;
+-----+-----+
| id | name           |
+-----+-----+
| 2  | Sandra Smith   |
+-----+-----+
1 row in set (0.00 sec)
```

設定查詢資料的條件：**where name='John Baker';**

```
mysql> select name, id, xpos, ypos from playerinfo where name='John Baker';
+-----+-----+-----+-----+
| name           | id | xpos | ypos |
+-----+-----+-----+-----+
| John Baker     | 3  | 10   | 5    |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

設定查詢資料的條件(name 是 J 開頭的，'%'為萬用字符代表任意長度的任意字元)：**where name like 'J%';**

```
mysql> select * from playerinfo where name like 'J%';
+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
| 3  | John Baker      | 10   | 5    | 231  |
| 4  | Jennifer West   | 13   | 73   | 5400 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

設定查詢資料的條件(name 裡有 J 的)：**where name like '%J%';**

```
mysql> select * from playerinfo where name like '%J%';
+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
| 3  | John Baker      | 10   | 5    | 231  |
| 4  | Jennifer West   | 13   | 73   | 5400 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

設定查詢資料的條件(name 的最後一個字母是 J)：**where name like '%J';**

```
mysql> select * from playerinfo where name like '%J';
Empty set (0.00 sec)
```

設定多重查詢資料的條件：**where name like '%S%' AND exp>2000;**

```
mysql> select * from playerinfo where name like '%S%';
+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp |
+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 |
| 2  | Sandra Smith    | 21   | 86   | 4563 |
| 4  | Jennifer West   | 13   | 73   | 5400 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> select * from playerinfo where name like '%S%' AND exp>2000;
+----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp |
+----+-----+-----+-----+
| 2  | Sandra Smith  | 21   | 86   | 4563 |
| 4  | Jennifer West | 13   | 73   | 5400 |
+----+-----+-----+-----+
2 rows in set (0.00 sec)
```

幫已經建立的表格增加一個新欄位(column or field)：

```
mysql> ALTER TABLE playerinfo ADD lastlogin TIMESTAMP;
Query OK, 4 rows affected (0.02 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE playerinfo;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default          | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)       | NO   | PRI | NULL             | auto_increment |
| name       | varchar(30)   | YES  |     | NULL             |                |
| xpos       | int(11)       | YES  |     | NULL             |                |
| ypos       | int(11)       | YES  |     | NULL             |                |
| exp        | int(11)       | YES  |     | NULL             |                |
| lastlogin  | timestamp     | YES  |     | CURRENT_TIMESTAMP |                |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

改變已存在表格欄位的資料型態(data type)：

```
mysql> ALTER TABLE playerinfo MODIFY name VARCHAR(50);
Query OK, 4 rows affected (0.02 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE playerinfo;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default          | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)       | NO   | PRI | NULL             | auto_increment |
| name       | varchar(50)   | YES  |     | NULL             |                |
| xpos       | int(11)       | YES  |     | NULL             |                |
| ypos       | int(11)       | YES  |     | NULL             |                |
| exp        | int(11)       | YES  |     | NULL             |                |
| lastlogin  | timestamp     | YES  |     | CURRENT_TIMESTAMP |                |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 | 0000-00-00 00:00:00 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 0000-00-00 00:00:00 |
| 3  | John Baker     | 10   | 5    | 231  | 0000-00-00 00:00:00 |
| 4  | Jennifer West  | 13   | 73   | 5400 | 0000-00-00 00:00:00 |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

更新已存在資料紀錄中某些欄位的值(data type)：

```
timestamp 為 NULL 時會填入目前的系統時間。
mysql> UPDATE playerinfo SET lastlogin = NULL WHERE name = "Jeff Henderson";
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 | 2006-12-01 10:46:58 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 0000-00-00 00:00:00 |
| 3  | John Baker     | 10   | 5    | 231  | 0000-00-00 00:00:00 |
| 4  | Jennifer West  | 13   | 73   | 5400 | 0000-00-00 00:00:00 |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
timestamp 的格式：西元年月日時分秒 (不足位則補零)
mysql> UPDATE playerinfo SET lastlogin = "20060630140350" WHERE name = 'Sandra Smith';
Rows matched: 1 Changed: 0 Warnings: 0
```

```
mysql> SELECT * FROM playerinfo;
```

```

+---+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp  | lastlogin           |
+---+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 | 2006-12-01 10:46:58 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3  | John Baker     | 10   | 5    | 231  | 0000-00-00 00:00:00 |
| 4  | Jennifer West  | 13   | 73   | 5400 | 0000-00-00 00:00:00 |
+---+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

```

mysql> UPDATE playerinfo SET lastlogin = "20061105083014" WHERE id >= 3;
Query OK, 2 rows affected (0.02 sec)
Rows matched: 2  Changed: 2  Warnings: 0

```

```

mysql> SELECT * FROM playerinfo;
+---+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp  | lastlogin           |
+---+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1431 | 2006-12-01 11:11:21 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3  | John Baker     | 10   | 5    | 231  | 2006-11-05 08:30:14 |
| 4  | Jennifer West  | 13   | 73   | 5400 | 2006-11-05 08:30:14 |
+---+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

```

mysql> UPDATE playerinfo SET exp = 1000 WHERE name LIKE "J%";
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

```

```

mysql> SELECT * FROM playerinfo;
+---+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp  | lastlogin           |
+---+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1000 | 2006-12-01 11:20:51 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3  | John Baker     | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 4  | Jennifer West  | 13   | 73   | 1000 | 2006-12-01 11:20:51 |
+---+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

(*) 如何複製表格：

1) 首先建立與原表格相同的結構：

```
CREATE TABLE new_table LIKE old_table;
```

```
mysql> CREATE TABLE backup LIKE playerinfo;
```

2) 接著將原表格的所有資料，插入新表格

```
INSERT new_table SELECT * FROM old_table;
```

但這個方式有時會漏失一些資訊。

```
mysql> INSERT backup SELECT * from playerinfo;
```

```

mysql> SHOW TABLES;
+-----+
| Tables_in_mydb |
+-----+
| playerinfo      |
| backup          |
+-----+
2 rows in set (0.06 sec)

```

```

mysql> SELECT * FROM backup;
+---+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp  | lastlogin           |
+---+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 1000 | 2006-12-01 11:20:51 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3  | John Baker     | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 4  | Jennifer West  | 13   | 73   | 1000 | 2006-12-01 11:20:51 |
+---+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

```

mysql> UPDATE playerinfo SET exp = 500 WHERE xpos>10 AND xpos<20;
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0

```

```

mysql> SELECT * FROM playerinfo;
+---+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp  | lastlogin           |
+---+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 500  | 2006-12-01 11:27:04 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |

```

```
| 3 | John Baker      | 10 | 5 | 1000 | 2006-12-01 11:20:51 |
| 4 | Jennifer West   | 13 | 73 | 500 | 2006-12-01 11:27:04 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

向資料庫查詢資料，並指定所查到資料記錄顯示時的排列順序：ORDER BY name；
(順序預設為由小到大：ASC，如果要依大到小排列，則在該項目名稱後加 DESC)

```
mysql> SELECT * FROM playerinfo ORDER BY name;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 1 | Jeff Henderson | 14   | 35   | 500 | 2006-12-01 11:27:04 |
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
| 3 | John Baker      | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 2 | Sandra Smith    | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.04 sec)
```

```
mysql> SELECT * FROM playerinfo ORDER BY name DESC;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 2 | Sandra Smith    | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3 | John Baker      | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
| 1 | Jeff Henderson | 14   | 35   | 500 | 2006-12-01 11:27:04 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

為查到的資料記錄指定多個排列順位：先指定的先排。

```
mysql> SELECT * FROM playerinfo ORDER BY exp, xpos;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
| 1 | Jeff Henderson | 14   | 35   | 500 | 2006-12-01 11:27:04 |
| 3 | John Baker      | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 2 | Sandra Smith    | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM playerinfo ORDER BY exp DESC, xpos;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 2 | Sandra Smith    | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3 | John Baker      | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
| 1 | Jeff Henderson | 14   | 35   | 500 | 2006-12-01 11:27:04 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM playerinfo ORDER BY exp DESC, xpos DESC;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 2 | Sandra Smith    | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3 | John Baker      | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 1 | Jeff Henderson | 14   | 35   | 500 | 2006-12-01 11:27:04 |
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

將查詢到的結果先暫存到變數中(變數的名稱前有@)，稍後再用：

```
mysql> SELECT @maxid:=MAX(id) FROM playerinfo;
+-----+
| @maxid:=MAX(id) |
+-----+
| 4 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM playerinfo WHERE id=@maxid;
+-----+-----+-----+-----+-----+-----+
| id | name           | xpos | ypos | exp | lastlogin           |
+-----+-----+-----+-----+-----+-----+
| 4 | Jennifer West   | 13   | 73   | 500 | 2006-12-01 11:27:04 |
+-----+-----+-----+-----+-----+-----+
```

```

1 row in set (0.00 sec)

mysql> SELECT @lastlogin:=MAX(lastlogin) FROM playerinfo;
+-----+
| @lastlogin:=MAX(lastlogin) |
+-----+
| 2006-12-01 11:27:04         |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM playerinfo WHERE lastlogin=@lastlogin;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 500  | 2006-12-01 11:27:04 |
| 4  | Jennifer West  | 13   | 73   | 500  | 2006-12-01 11:27:04 |
+----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

限定查詢到的資料記錄的顯示筆數：**LIMIT 2** (請再試著改成 **LIMIT 1, 2**看看發現什麼規則)

```

mysql> SELECT * FROM playerinfo LIMIT 2;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 1  | Jeff Henderson | 14   | 35   | 500  | 2006-12-01 11:27:04 |
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
+----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

由表單刪除資料：資料刪除後無法復原，所以使用 DELETE 指令時要小心；
雖然刪除表單所有的資料，但該表單仍舊存在，也就是說可以新增新的資料列(記錄)。

```

mysql> DELETE FROM playerinfo WHERE name='Jeff Henderse';
Query OK, 0 rows affected (0.00 sec)

mysql> DELETE FROM playerinfo WHERE name='Jeff Henderson';
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 3  | John Baker     | 10   | 5    | 1000 | 2006-12-01 11:20:51 |
| 4  | Jennifer West  | 13   | 73   | 500  | 2006-12-01 11:27:04 |
+----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> DELETE FROM playerinfo WHERE id=3;
Query OK, 1 row affected (0.00 sec)

```

刪除 playerinfo 表格中的所有資料：

```

mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name          | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 2  | Sandra Smith   | 21   | 86   | 4563 | 2006-06-30 14:03:50 |
| 4  | Jennifer West  | 13   | 73   | 500  | 2006-12-01 11:27:04 |
+----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

也是刪除 playerinfo 表格中的所有資料：

```

mysql> DELETE FROM playerinfo;
Query OK, 2 rows affected (0.00 sec)

mysql> SELECT * FROM playerinfo;
Empty set (0.00 sec)

```

雖然刪除 playerinfo 表格中的所有資料，但 playerinfo 表格仍存在：

```

mysql> show tables;
+-----+
| Tables_in_mydb |
+-----+
| playerinfo      |
+-----+
1 row in set (0.00 sec)

```

的確表格仍存在！

因此可以再插入一列新的資料 (依序給完整資料)：

```
mysql> INSERT INTO playerinfo VALUES (NULL, 'Rich Wang', 30, 40, 500, "20061201121700");
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name      | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 5  | Rich Wang | 30   | 40   | 500  | 2006-12-01 12:17:00 |
+----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

也可以使用部分資料來插入新的一列 (依序給相關的欄位與對應的資料)：

```
mysql> INSERT INTO playerinfo (id, name, xpos, ypos) VALUES (NULL, '王小吉', 30, 40);
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name      | xpos | ypos | exp  | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 5  | Rich Wang | 30   | 40   | 500  | 2006-12-01 12:17:00 |
+----+-----+-----+-----+-----+-----+
| 25 | 王小吉    | 30   | 40   | NULL | 2007-01-11 05:19:24 |
+----+-----+-----+-----+-----+-----+
2 row in set (0.00 sec)
```

改變表格欄位的名稱：CHANGE oldFieldName newFieldName dataType

```
mysql> ALTER TABLE playerinfo CHANGE exp expValue INT;
Query OK, 1 row affected (0.02 sec)
Records: 1 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+-----+
| id | name      | xpos | ypos | expValue | lastlogin          |
+----+-----+-----+-----+-----+-----+
| 5  | Rich Wang | 30   | 40   | 500      | 2006-12-01 12:17:00 |
+----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

刪除表格中的欄位：DROP COLUMN fieldName;

```
mysql> ALTER TABLE playerinfo DROP COLUMN expValue;
Query OK, 1 row affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+
| id | name      | xpos | ypos | lastlogin          |
+----+-----+-----+-----+-----+
| 5  | Rich Wang | 30   | 40   | 2006-12-01 12:17:00 |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> ALTER TABLE playerinfo DROP COLUMN xpos;
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+
| id | name      | ypos | lastlogin          |
+----+-----+-----+-----+
| 5  | Rich Wang | 40   | 2006-12-01 12:17:00 |
+----+-----+-----+-----+
1 row in set (0.00 sec)
```

將xpos欄位加回去，在 name 與 ypos 之間：

```
mysql> ALTER TABLE playerinfo ADD COLUMN xpos AFTER name;
or
mysql> ALTER TABLE playerinfo ADD COLUMN xpos BEFORE ypos;
```

```
mysql> SELECT * FROM playerinfo;
+----+-----+-----+-----+-----+
| id | name      | xpos | ypos | lastlogin          |
+----+-----+-----+-----+-----+
| 5  | Rich Wang | NULL | 40   | 2006-12-01 12:17:00 |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```


更改表格的名稱：**ALTER TABLE** oldTableName **RENAME TO** newTableName;

```
mysql> ALTER TABLE playerinfo RENAME TO byetable;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> show tables;
+-----+
| Tables_in_mydb |
+-----+
| byetable        |
+-----+
1 row in set (0.00 sec)
```

從資料庫中刪除表格：**DROP TABLE** tableName;

```
mysql> DROP TABLE byetable;
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> SHOW TABLES;
Empty set (0.00 sec)
```