

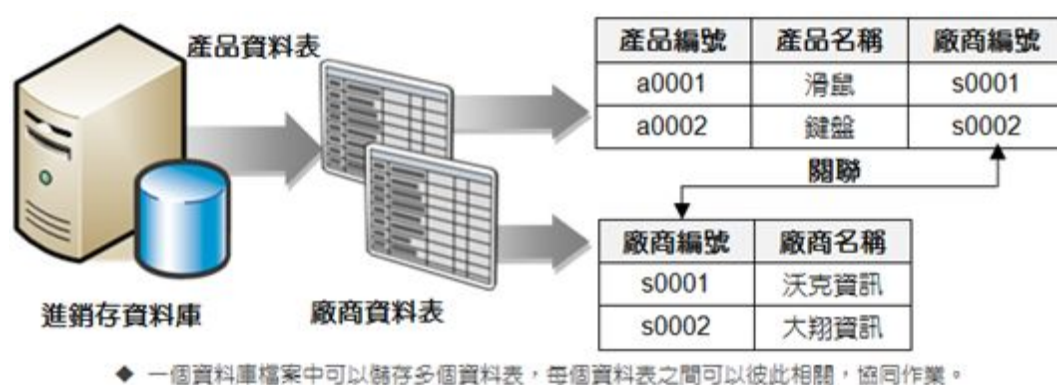
MySQL 資料庫的使用與管理

關於資料庫

要使一個網站達到互動的效果，不是讓網頁充滿了動畫、音樂，而是當瀏覽者對網頁提出要求時能出現回應的結果。而這樣的效果，大多必須搭配資料庫的使用，讓網頁讀出儲存在資料庫中的資料，顯示在網頁上，也因為每個瀏覽者對於某一個相同的網頁所提出的要求不同，顯示出的結果即不同，這才是真正的互動網站。

認識資料庫

資料庫 (Database) 可以說是一些相關資料的集合並進行儲存的地方，我們可以一定的原則與方法新增、編輯、刪除資料的內容，進而搜尋、分析、比對所有資料，取得可用的資訊，產生所需的結果。



資料庫管理系統

- DBMS
- Database Server
- Database Engine

關聯式資料庫

MySQL 資料庫的特色

早期的資料庫大都屬於操作在單機或是區域網路的系統，雖然在操作上方便，管理上也比較輕鬆，但是談到資料庫本身的效能或是安全性都是相當不利的弱點。

隨著 Internet 的興起，開始興起 Internet 資料庫的概念，此時資料庫的角色已經化為一個在國際網路上提供資料存取編輯、應用查詢的伺服器了。MySQL 資料庫就是這個概念的一個具體表現，在與網站伺服器結合作業後，MySQL 資料庫就成為了一個網路型的資料庫系統。

資料庫的字元集與連線校對

我們將採用 phpMyAdmin 管理程式來執行，以便能有更簡易的操作環境與使用效果，讓所有學習 MySQL 的初學者可以輕鬆入門，也可以讓已經相當熟悉 SQL 指令的朋友能得心應手。

在 phpMyAdmin 的畫面左方是選取要顯示的資料庫、資料表及常用功能連結的地方。

認識字元集與連線校對

字元集 (character set) 是指資料庫中資料文字的編碼方式，而 連線校對 (collation) 是資料中字元的排序方式。

中文衝碼問題

在中文的環境中，我們可以選取 big5 繁體中文與 utf8 的編碼方式做為操作 MySQL 資料庫的字元集。

建議中文使用的字元集與連線校對

在 MySQL 資料庫最完美的解決方式還是選擇使用 utf8 的方式來進行文字編碼，如此即可將中文字以 Unicode 的方式進行儲存，徹底解決衝碼問題。



為什麼MYSQL要設定用UTF8MB4編碼 UTF8MB4_UNICODE_CI

utf8 跟 utf8mb4具有相同的儲存特性：相同的代碼值，相同的編碼，相同的長度。不過 utf8mb4擴展到一個字符最多可有4位元，所以能支持更多的位元集。utf8mb4兼容utf8，且比utf8能表示更多的字串，將編碼改為utf8mb4外不需要做其他轉換。為了要跟國際接軌，原本的 utf8 編碼在儲存某些國家的文字(或是罕見字)已經不敷使用，因此在mysql 5.5.3版以上，可以使用4-Byte UTF-8 Unicode的編碼方式。

到底要用 utf8mb4_general_ci 還是 utf8mb4_unicode_ci 呢？

建議使用：utf8mb4_unicode_ci

MySQL 使用的字元集與連線校對

MySQL 由資料庫、資料表、資料欄位各個層級中，都可以單獨設定採用的字體集與連線校對方式。若在某一層操作時沒有設定使用字體集與連線校對方式為何，就會繼承上一層的設定來使用，若都沒有設定就會以 MySQL 資料庫的預設值來做使用標準。

資料庫與資料表的新增

我們將要在 MySQL 中建置一個學校班級的資料庫：「class」，並新增一個同學通訊錄資料表：「students」。



認識資料表的欄位

在資料表新增前，首要的動作是要先規劃資料表中所要使用的欄位。其中設定資料欄位的類型非常重要，使用正確的資料型態才能正確的儲存、應用資料。

數值型態

可運用來儲存、計算的數值資料欄位，例如會員的編號或是產品的價格等。在 MySQL 中的數

值欄位依照所儲存的資料所需空間大小有以下的區別：

數值型態	儲存空間	資料的表示範圍
TINYINT	1 byte	SIGNED : -128 ~ 127 , UNSIGNED : 0 ~ 255 。
SMALLINT	2 bytes	SIGNED : -32768 ~ 32767 , UNSIGNED : 0 ~ 65535 。
MEDIUMINT	3 bytes	SIGNED : -8388608 ~ 8388607 , UNSIGNED : 0 ~ 16777215 。
INT 或 INTERGER	4 bytes	SIGNED : -2147483648 ~ 2147483647 。 UNSIGNED : 0 ~ 4294967295 。
BIGINT	8 bytes	SIGNED: -9223372036854775808 ~ 9223372036854775807 。 UNSIGNED: 0 ~ 18446744073709551615 。
FLOAT[(M,D)]	4 bytes	單精度浮點數，能夠記錄小數點。M 為顯示位數，不得大於 255。D 為小數位數，其值範圍 M-2 ~ 30。 SIGNED: $\pm 1.175494351\text{E}-38$ 。 UNSIGNED: $\pm 3.402823466\text{E}+38$ 。
DOUBLE[(M,D)]	8 bytes	雙精度浮點數，用途與 FLOAT 雷同，但是儲存空間是二倍。除非特別需要高精度或範圍極大的值，一般來說用 FLOAT 來儲存資料應該是夠了。 Signed: $\pm 1.7976931348623157\text{E}+308$ 。 UNSIGNED: $\pm 2.2250738585072014\text{E}-308$ 。
DECIMAL[(M[,D])]	M+2	DECIMAL 的範圍與 DOUBLE 一樣，但是其有效的取值範圍由 M 和 D 的值決定。如果改變 M 而固定 D，則其取值範圍將隨 M 的變大而變大。

文字型態

可用來儲存文字類型的資料，如學生姓名、地址等。在 MySQL 中文字型態資料有下列幾種格式：

文字型態	最大儲存空間	資料的特性
CHAR(M)	M bytes，最大為 255 bytes 。	必須指定欄位大小，資料不足時以空白字元填滿。
VARCHAR(M)	M bytes，最大為 255 bytes 。	必須指定欄位大小，但以實際填入的資料內容來儲存。
TINYTEXT	255 bytes	基本上是 TEXT 的資料型態，依儲存容量大小而有名稱上的區分。在使用設定時要依可能使用到的空間大小來選擇設定。當不確定要儲存文字內容長度的狀況時最建議使用，因為在儲存時可依內容彈性調整儲存大小。
TEXT	65,535 bytes	
MEDIUMTEXT	16,777,215 bytes	
LONGTEXT	4,294,967,295 bytes	
TINYBLOB	255 bytes	基本上是 BLOB 的資料型態，依儲存容量大小而有名稱上的區分。用於儲存圖片、聲音、影片等二進位檔案的資料，在儲存時可依內容彈性調整儲存大小。
BLOB	65,535 bytes	
MEDIUMBLOB	16,777,215 bytes	
LONGBLOB	4,294,967,295 bytes	

日期及時間型態

日期及時間型態：可用來儲存日期或是時間類型的資料，例如會員的生日、留言的時間等。

MySQL 中的日期及時間型態有下列幾種格式：

日期時間型態	儲存空間	範圍及格式																
DATE 日期	3 bytes	"1000-01-01" 到 "9999-12-31" <table><tr><th>4 位數西元年格式</th><th>2 位數西元年格式</th></tr><tr><td>YYYY-MM-DD</td><td>YY-MM-DD</td></tr><tr><td>YYYYMMDD</td><td>YYMMDD</td></tr></table>	4 位數西元年格式	2 位數西元年格式	YYYY-MM-DD	YY-MM-DD	YYYYMMDD	YYMMDD										
4 位數西元年格式	2 位數西元年格式																	
YYYY-MM-DD	YY-MM-DD																	
YYYYMMDD	YYMMDD																	
TIME 時間	3 bytes	"-838:59:59" 到 "838:59:59" 資料格式為：hh:mm:ss 或 hhmmss。																
DATETIME 日期時間	8 bytes	"1000-01-01 00:00:00" 到 "9999-12-31 23:59:59" <table><tr><th>4 位數西元年格式</th><th>2 位數西元年格式</th></tr><tr><td>YYYY-MM-DD hh:mm:ss</td><td>YY-MM-DD hh:mm:ss</td></tr><tr><td>YYYYMMDDhhmmss</td><td>YYMMDDhhmmss</td></tr></table>	4 位數西元年格式	2 位數西元年格式	YYYY-MM-DD hh:mm:ss	YY-MM-DD hh:mm:ss	YYYYMMDDhhmmss	YYMMDDhhmmss										
4 位數西元年格式	2 位數西元年格式																	
YYYY-MM-DD hh:mm:ss	YY-MM-DD hh:mm:ss																	
YYYYMMDDhhmmss	YYMMDDhhmmss																	
TIMESTAMP[(M)] 時間戳記	4 bytes	"1970-01-01 00:00:00" 到 "2037" 在使用 TIMESTAMP 型態時，M 為顯示位數，但與儲存所需空間無關，而是與顯示的格式有關，請見下表： <table><tr><th>型態</th><th>顯示格式</th></tr><tr><td>TIMESTAMP(14)</td><td>YYYYMMDDhhmmss</td></tr><tr><td>TIMESTAMP(12)</td><td>YYMMDDhhmmss</td></tr><tr><td>TIMESTAMP(10)</td><td>YYMMDDhhmm</td></tr><tr><td>TIMESTAMP(8)</td><td>YYYYMMDD</td></tr><tr><td>TIMESTAMP(6)</td><td>YYMMDD</td></tr><tr><td>TIMESTAMP(4)</td><td>YYMM</td></tr><tr><td>TIMESTAMP(2)</td><td>YY</td></tr></table>	型態	顯示格式	TIMESTAMP(14)	YYYYMMDDhhmmss	TIMESTAMP(12)	YYMMDDhhmmss	TIMESTAMP(10)	YYMMDDhhmm	TIMESTAMP(8)	YYYYMMDD	TIMESTAMP(6)	YYMMDD	TIMESTAMP(4)	YYMM	TIMESTAMP(2)	YY
型態	顯示格式																	
TIMESTAMP(14)	YYYYMMDDhhmmss																	
TIMESTAMP(12)	YYMMDDhhmmss																	
TIMESTAMP(10)	YYMMDDhhmm																	
TIMESTAMP(8)	YYYYMMDD																	
TIMESTAMP(6)	YYMMDD																	
TIMESTAMP(4)	YYMM																	
TIMESTAMP(2)	YY																	
YEAR[(2 4)] 年	1 bytes	參數 4 為 1901 到 2155 (預設)。 參數 2 為 1970 到 2069。																
備註：表格內格式 Y 為年、M 為月、D 為日、h 為時、m 為分、s 為秒。																		

特殊資料型態

還有二個特殊的資料型態，嚴格來說它們都屬於文字型態，但是因為它們的內容只能由固定的選項內挑選，又有人稱它們為「列舉資料型態」，內容如下：

資料型態	儲存空間	最多選項	資料的特性
ENUM	1 或 2 bytes。	65535	儲存資料為單選選項的結果。當儲存的資料欄位有多個選項，但是只能擇一使用時可以使用這種資料欄位。例如性別 (男、女)、季節 (春、夏、秋、冬) 等選項。
SET	1、2、3、4、8 bytes。	64	儲存資料為複選選項的結果。當儲存的資料欄位有多個選項，可有多個選擇時可以使用這種資料欄位。例如會員資料的興趣、由何處得知本站等選項。

重要的欄位屬性

在建置資料表時，除了要依不同性質的資料選擇適合的欄位型態，有些重要的欄位屬性定義也能在不同的型態欄位中發揮其功能，常用的設定如下：

特性定義名稱	儲存空間	定義內容
SIGNED,UNSIGNED	數值類型	定義數值資料中是否允許有負值，SIGNED 表示允許。
auto_increment	數值類型	自動編號，由 0 開始以 1 來累加。
BINARY	文字類型	儲存的字元有大小寫的區別。
NULL,NOT NULL	全部	是否允許不填入資料於欄位中。
預設值	全部	若是欄位中沒有資料，即以預設值代入。
主鍵 PrimaryKey	全部	主鍵，每個資料表中只能允許一個主鍵，而且該欄資料不能重複，加強資料表的檢索功能。

新增資料表

欄位規劃

以下我們要新增一個同學個人資料表：「students」，以下是這個資料表欄位的規劃：

名稱	欄位名稱	資料型態	屬性	NULL	其他
座號	clD	TINYINT(2)	UNSIGNED ZEROFILL	否	主鍵，auto_increment
姓名	cName	VARCHAR(20)		否	
性別	cSex	ENUM('F','M')		否	預設為 F (female, 女性)
生日	cBirthday	DATE		否	
電子郵件	cEmail	VARCHAR(100)		是	
電話	cPhone	VARCHAR(50)		是	
住址	cAddr	VARCHAR(255)		是	

資料的新增

以下是這一次要新增到資料庫中的學生資料：

座號	姓名	性別	生日	電子郵件	電話	住址
1	簡奉君	女	1987-04-04	elven@superstar.com	0922988876	台北市濟洲北路 12 號
2	黃靖倫	男	1987-07-01	jinglun@superstar.com	0918181111	台北市敦化南路 93 號 5 樓
3	潘四敬	男	1987-08-11	sugie@superstar.com	0914530768	台北市中央路 201 號 7 樓
4	賴勝恩	男	1984-06-20	shane@superstar.com	0946820035	台北市建國路 177 號 6 樓
5	黎楚寧	女	1988-02-15	ivy@superstar.com	0920981230	台北市忠孝東路 520 號 6 樓
6	蔡中穎	男	1987-05-05	zhong@superstar.com	0951983366	台北市三民路 1 巷 10 號
7	徐佳蔓	女	1985-08-30	lala@superstar.com	0918123456	台北市仁愛路 100 號
8	林雨婕	女	1986-12-10	crystal@superstar.com	0907408965	台北市民族路 204 號
9	林心儀	女	1988-12-01	peggy@superstar.com	0916456723	台北市建國北路 10 號
10	王燕博	男	1993-08-10	albert@superstar.com	0918976588	台北市北環路 2 巷 80 號

資料瀏覽

資料的編輯

資料的刪除

資料庫的備份

使用 phpMyAdmin 的管理程式選取資料庫中的所有資料表，匯成一個單獨的文字檔。當資料庫受到損壞或是要將資料搬移到新的 MySQL 資料庫時，只要將這個文字檔匯入即可完成。

資料庫的還原

關於 MySQL 資料庫的安全問題

對於 MySQL 資料庫的安全考量

MySQL 資料庫是一個存在於網際網路上的資料庫系統，換句話說只要是網際網路上的使用者都可以連接到這個資源，如果沒有權限或其他措施的控管，任何人都可以對 MySQL 資料庫予取予求，為所欲為。

對於 phpMyAdmin 資料庫的安全考量

phpMyAdmin 是一套網頁介面的 MySQL 管理程式，有許多 PHP 的程式設計師都會將這套工具直接上傳到他的 PHP 網站資料夾裡，那麼管理者只要從遠端透過瀏覽器登入 phpMyAdmin 來管理的資料庫了！

防堵安全漏洞的好建議

無論是 MySQL 資料庫本身的權限設定或是 phpMyAdmin 管理程式的安全漏洞，為了避免他人透過網路入侵您的資料庫，有幾件事必須要先做的：

1. 修改 phpMyAdmin 管理程式的資料夾名稱
2. 為MySQL資料庫的管理帳號加上密碼
3. 養成備份 MySQL 資料庫的習慣

檢視帳號設定



The screenshot shows the phpMyAdmin interface with the 'Users' tab selected. The title is '使用者帳號一覽'. Below it is a table with columns: 帳號 (Username), 主機名稱 (Host), 密碼 (Password), 全域權限 (Global Privileges), 允許授權 (Grant Privileges), and 動作 (Actions). Two users are listed: 'mysql.sys' and 'root'. The 'root' user is highlighted with a red box. At the bottom, there are checkboxes for '全選' (Select All) and '已選擇項目' (Selected Items) with a '匯出' (Export) button.

帳號	主機名稱	密碼	全域權限	允許授權	動作
<input type="checkbox"/> mysql.sys	localhost	是	USAGE	否	編輯權限 匯出
<input type="checkbox"/> root	localhost	是	ALL PRIVILEGES	是	編輯權限 匯出

↑ ☐ 全選 已選擇項目: 匯出

全域權限

☒ 全選

注意: MySQL 權限名稱會以英文表示。

☒ 資料

☒ SELECT
☒ INSERT
☒ UPDATE
☒ DELETE
☒ FILE

☒ 結構

☒ CREATE
☒ ALTER
☒ INDEX
☒ DROP
☒ CREATE TEMPORARY TABLES
☒ SHOW VIEW
☒ CREATE ROUTINE
☒ ALTER ROUTINE
☒ EXECUTE
☒ CREATE VIEW
☒ EVENT
☒ TRIGGER

☒ 管理

☒ GRANT
☒ SUPER
☒ PROCESS
☒ RELOAD
☒ SHUTDOWN
☒ SHOW DATABASES
☒ LOCK TABLES
☒ REFERENCES
☒ REPLICATION CLIENT
☒ REPLICATION SLAVE
☒ CREATE USER

資源限制

注意: 設定為 0 即代表無限制。

MAX QUERIES PER HOUR

MAX UPDATES PER HOUR

MAX CONNECTIONS PER HOUR

MAX USER_CONNECTIONS

修改帳號的權限

每個帳號可以設定對於資料編輯、結構調整，甚至是系統管理的權限。

建立或修改密碼

更改登入資訊 / 複製使用者

為單一資料庫新增使用者

新增一個只能使用該資料庫的帳號。接著我們將要新增一個帳號：「class」來使用資料庫 class。

刪除使用者

設定 phpMyAdmin 的登入方式

phpMyAdmin 是許多人在管理、操作 MySQL 資料庫最常使用的工具。因為 phpMyAdmin 是以網頁的方式存在，無論是管理者或是一般使用者都可以藉由瀏覽器開啟，登入機制的啟用便是防堵沒有帳號權限的人入侵資料庫的最後一道關卡。phpMyAdmin 有三種登入認證模式：config、cookie 及 http，以下將說明這三種認證模式的差異與設定方法。

- config 登入認證模式

若是您的程式是處於本機開發階段，並沒有安全性上的考量，不希望每次使用 phpMyAdmin 來管理操作 MySQL 資料庫時都要一再輸入帳號、密碼，造成開發時的困擾，您可以將資料庫的帳號、密碼直接寫到 phpMyAdmin 的設定檔 `<config.inc.php>` 中，如此在進入 phpMyAdmin 時就會直接跳過登入步驟進入管理畫面中。

- http 登入認證模式

若是您的網站伺服器是使用 httpd 的方式來編譯 PHP 的程式頁面，即可使用 http 登入認證模式，如 Apache 伺服器。但若是使用 CGI 的方式來使用 PHP 程式頁面就不能使用，例如 IIS 伺服器。

- cookie 登入認證模式

這是一種在設定上限制最少，使用上也很方便的認證模式，也是一開始說明 phpMyAdmin 時所採用的認證模式。只要是使用者的瀏覽器允許 cookie 的使用，即可使用 cookie 登入認證模式。