

# 第十一章 表格異動處理

## 11-1 DML 與 DDL 簡介

本書到目前為止，花費了許多章節介紹 Select 查詢命令，因它是由資料庫萃取資訊的主要工具。除此之外，還有兩大類管理命令，本章就來一一介紹：

### (A) 『資料定義語言』(Data Definition Language, DDL)

提供產生(Create)、變更(Alter)與刪除 (Delete)資料表之命令，雖然還有其他命令，但牽涉過於廣泛本書僅介紹這三種較常用的命令。

### (B) 『資料處理語言』(Data Manipulation Language, DML)

提供對資料表內記錄(資料行)做插入 (Insert)、更新(Update)與刪除(Delete) 管理之命令，本章將一一介紹各命令的使用法，與透過 PHP 來靈活運用它。

## 11-2 建立表格 – Create table

### 11-2-1 Create table 語法

Create table 命令是用來建立資料表的。一般資料庫經過邏輯設計，規劃若干的資料表後就用此命令來產生。因此，大多在 SQL 控制台上處理，譬如，在 AppServ 上以選單完成，但我們還是介紹它的語法，如下：

```
Create Table 資料表名稱  
  
    (欄位_1 資料型態 屬性,  
  
    欄位_2 資料型態 屬性,  
  
    ...,
```

```
Primary Key (欄位_1, 欄位_2)
)
```

其中資料型態，常用有：

- a. 字元：CHAR(長度)，CHAR(8)。
- b. 整數：int(長度)，int(8)。
- c. 日期：

另外，屬性有：

- a. 不可空白：Not null
- b. Primary Key：指定主鍵，也許會包含兩個以上欄位。

## 11-2-2 範例研討 – 建立監護人表格

### (A) 系統功能：Ex11\_1

學務處需要建立學生家長相關系統，期望建立一個監護人資料表(guardian)，期望表格結構 {student\_ID, guard\_name, tel, address}，並以 student\_ID 為唯一識別。

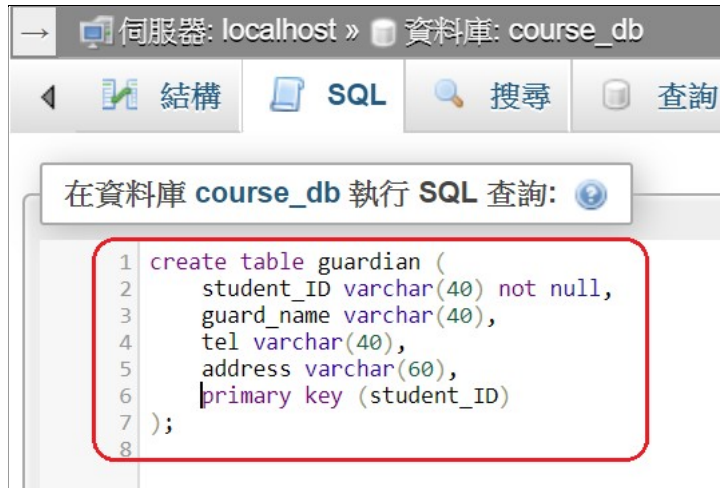
### (B) SQL 系統分析

為了簡化系統設計，我們假設每位學生與監護人之間是一對一關係，因此 guardian 與 students 兩張資料表之間是一對一的關係，之間是由 student\_ID 欄位建立關連，然而 guardian 資料表的 student\_ID 的資料型態須相同。我們規劃如下：

```
create table guardian (
    student_ID varchar(40) not null,
    guard_name varchar(40),
    tel varchar(40),
    address varchar(60),
    primary key (student_ID)
);
```

### (C) 利用 Appserv 控制台建立

在 AppServ 的 SQL 介面下輸入下列命令即可：



執行後 course\_db 資料庫內產生險 guardian 資料表如下：



## 11-3 變更表格 – Alter table

### 11-3-1 Alter table 語法

當表格不滿足所需時，可以利用 Alter Table 變更其型態。可針對現有表格作：增加欄位、修改欄位、刪除欄位，變更主鍵，以及定義屬性、、、等等功能，格式如下：

```
Alter Table 資料表名稱  
[Alter] [欄位] [資料型態][Null | Not null]  
[Add | Drop] [欄位] [屬性][限制]
```

### 11-3-2 範例研討：增加關係欄位

#### (A) 系統功能：Ex10\_2

學務處認為目前監護人資料表(guardian)不能滿足所需，期望增加學生與監護人之間的關係欄位(relation)。

#### (B) 系統分析

描述關係人(relation)可能是父子、父女、養父子、兄妹、、、等等關係，因此，其資料型態還是使用 VARCHAR(40) 較適合，吾人規劃如下：


```
Alter table guardian  
Add relation varchar(40) not null;
```

#### (C) 利用 Appserv 控制台建立

在 AppServ 的 SQL 介面下輸入下列命令即可：



執行後，guardian 表的結構變更如下：

<input type="checkbox"/>	1	<b>student_ID</b> 	varchar(40)	utf8_unicode_ci	否	無
<input type="checkbox"/>	2	<b>guard_name</b>	varchar(40)	utf8_unicode_ci	是	NULL
<input type="checkbox"/>	3	<b>tel</b>	varchar(40)	utf8_unicode_ci	是	NULL
<input type="checkbox"/>	4	<b>address</b>	varchar(60)	utf8_unicode_ci	是	NULL
<input type="checkbox"/>	5	<b>relation</b>	varchar(40)	utf8_unicode_ci	否	無

## 11-4 刪除表格 – Drop table

### 11-4-1 Drop table 語法

當表格不需要時，可以利用 Drop Table 命令將它刪除，格式如下：

```
Drop Table 資料表名稱 [Restrict][Cascade];
```

### 11-4-2 範例研討：刪除監護人表格

#### (A) 系統功能：Ex11\_3

學務處認為不再需要監護人資料表(guardian)，期望將它刪除。

#### (C) 利用 SQL 命令刪除

在 AppServ 的 SQL 介面下輸入下列命令即可：

```
在資料庫 course_db 執行以下指令: ?  
Drop table guardian;
```

## 11-5 資料插入 – Insert into

### 11-5-1 Insert into 語法

將資料插入資料表則使用 Insert 命令達成，但它每次增加一筆記錄，格式如下：

```
Insert into 資料表名稱 <欄位串列>
Value (<資料串列>);
```

## 11-5-2 範例研討 – 增加監護人資料

### (A) 系統功能：Ex11\_4

請新增 3 位學生的監護人資料到 guardian 資料表內。(請先建立 guardian 資料表)

### (B) 利用 SQL 命令

在 AppServ 的 SQL 介面下輸入下列命令即可：**(如已刪除 guardian 表請再新增)**

```
Insert Into guardian
values("90211304", "陳媽媽", "0923111223", "高雄市苓雅區建國路 30 號", "母女");
Insert Into guardian
values("90211305", "吳爸爸", "0923111341", "高雄市新興區民族路 40 號", "父女");
Insert Into guardian
values("90211306", "陳爸爸", "0945675223", "高雄市三民區九如路 10 號", "父子");
```

student_ID	guard_name	tel	address
90211305	吳爸爸	0923111341	高雄市新興區民族路 40 號
90211304	陳媽媽	0923111223	高雄市苓雅區建國路 30 號

## 11-5-3 範例研討 – 新增學生資料

### (A) 系統功能：Ex11\_5

教務處希望有一個可供登錄學生資料的網頁，期望輸入網頁的格式如下：

localhost/chap11/Ex11\_5-form.php

應用程式 單車活動 PBP 活動 ACP Google 北

登錄新生資料

班級名稱： 進四資管二丙 ▾

學生座號(兩位數不可重複)： 57

姓名： 劉美美

性別： 女 ▾

電話： 077310690

電子郵件： stu35@csu.edu.tw

地址： 台北市中山區大明路 223 號

送出

執行網頁的結果如下：

localhost/chap11/Ex11\_5-action.php

應用程式 單車活動 PBP 活動 ACP Google 北海道旅遊 法國旅行計畫 Bookmark

登錄新生資料如下：

學號	姓名	班級代碼	性別	電話	郵件	地址
90211357	劉美美	902113	女	077310690	stu35@csu.edu.tw	台北市中山區大明路 223 號

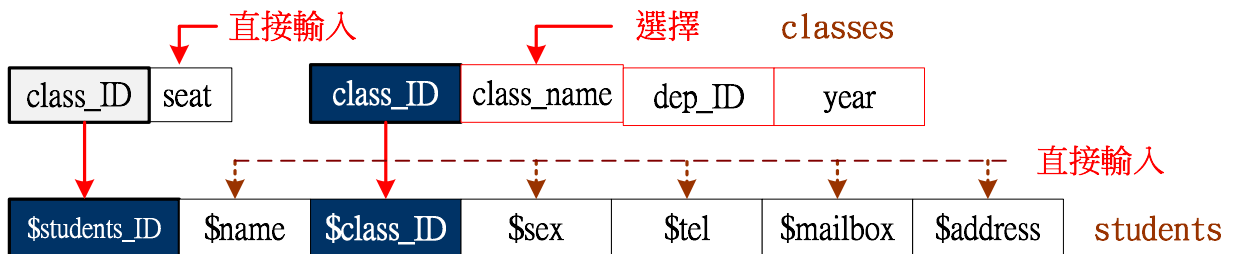
如果重複插入，則會出現此視窗：



## (B) SQL 系統分析

學生資料登錄於 `students` 表內，此題目須對該資料表做 `Insert into` 的動作。下圖是 `students` 與 `classes` 兩張資料表之間的關聯，兩張資料表內的 `class_ID` 欄位建立主鍵與外來鍵之關係，也就是說，當我們輸入 `students` 表內的 `class_ID` 欄位必須對應到 `classes` 表格，不可以隨意輸入，不然關聯就中斷了。因此，此 `class_ID` 與班級名稱(`class_name`) 之間的對應

關係，就由表單網頁來達成，使用者選擇班級名稱後，它傳送 class\_ID 給執行網頁。另外，student\_ID 是 class\_ID 加上座號(seat\_num)，在查詢網頁由使用者輸入座號，執行網頁再將 class\_ID 與 seat\_num 連結成為學號 (student\_ID)(請參閱第三章說明)。表單網頁的關聯圖如下：



『執行網頁』接收訊息後，將該筆資料插入 students 表內，SQL 程式如下：

```
Insert Into students
values('$student_ID', '$name', '$class_ID', '$sex', '$tel', '$mailbox', '$address')
```

### (C) 網頁製作技巧

依照上述 SQL 分析，網頁設計如下：



### (D) 查詢網頁範例：Ex10\_5-form.php

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <head>
3 <title>登錄新生資料</title></head>
4 <body>
5 <p> 登錄新生資料 </p>
6 <form name="表單" method="post" action="Ex11_5-action.php">
7 <input type="text" value="班級名稱" />
8 <input type="text" value="座號" />
9 <input type="text" value="性別" />

```



```
10     <select name="class_ID">
11 <?php
12     $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
13         or die("MySQL 伺服器連結失敗 <br>");
14
15     // 查詢班級名稱
16     $sql_query = "select class_ID, class_name
17                 from classes";
18     $result = mysqli_query($select_db, $sql_query);
19     while($row=mysqli_fetch_row($result)) {
20         echo "<option value='$row[0]'> $row[1] </option><BR>";
21     }
22     echo "</select>";
23     echo "<BR><BR>";
24     mysqli_close($select_db);
25 ?>
26
27     學生座號(兩位數不可重複)： <input name="seat_num" type="text"><BR>
28
29     姓名： <input name="name" type="text"><BR>
30
31     性別： <select name="sex">
32
33         <option>男</option>
34
35         <option>女</option>
36
37     </select><BR><BR>
38
39     電話： <input name="tel" type="text"><BR>
40
41     電子郵件： <input name="mailbox" type="text"><BR>
42
43     地址： <input name="address" type="text"><BR>
44
45     <input type="submit" value="送出">
46
47 </form>
48 </body>
49 </html>
```

程式說明：

- (1) 第 12 ~ 27 行：建立下拉式選單，讓使用者選擇班級名稱，再傳遞班級代碼 (class\_ID) 給執行網頁。
- (2) 第 30~38 行：直接輸入學生資料，包含座號。

#### (D) 執行網頁範例：Ex10\_5-action.php

```
1 <?php
2     $class_ID = $_POST["class_ID"];
3     $seat_num = $_POST["seat_num"];
4     $student_ID = $class_ID.$seat_num; //連結產生學號
5
6     $name = $_POST["name"];
7     $sex = $_POST["sex"];
8     $tel = $_POST["tel"];
9     $mailbox = $_POST["mailbox"];
10    $address = $_POST["address"];
11    $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
12        or die("MySQL 伺服器連結失敗 <br>");
13    $sql_query = "insert into students values(
14                $student_ID, '$name', '$class_ID',
15                '$sex', '$tel', '$mailbox', '$address')";
16    $result = mysqli_query($select_db, $sql_query);
17    // 測試是否登錄成功
18    if (!$result){
19        echo "學生資料登錄失敗(可能學號重複，請進入 AppServ 觀察)! <ER>";
20        exit();
21    }
22    // 查詢新增學生資料
23    $sql_query = "select *
24                from students
25                where student_ID = $student_ID";
26    $result = mysqli_query($select_db, $sql_query);
27    echo "登錄新生資料如下：<BR><BR>";
28
29
```

```
30     $items = array("學號", "姓名", "班級代碼", "性別", "電話", "郵件", "地址");
31     echo "<table border = '2'><tr align='center'>";
32     // 利用 foreach 迴圈印出陣列 $item
33     foreach ($items as $value) {
34         echo "<td> $value </td>";
35     }
36     echo "</tr>";
37     while($row=mysqli_fetch_row($result)) {
38         echo "<tr>";
39         for($j=0; $j<mysqli_num_fields($result); $j++) {
40             echo "<td>$row[$j]</td>";
41         }
42         echo "</tr>";
43     }
44     echo "</table>";
45     mysql_close($db_link);
46     ?>
```

程式說明：

- (1) 第 18~21 行：插入學生資料。
- (2) 第 28~40 行：顯示新增學生資料的內容。

## 11-5-4 範例研討 – 新增課程

### (B) 系統功能：Ex11\_6

教務處需要一個可供新增課程的網頁，期望查詢網頁如下：

localhost/chap11/Ex11\_6-form.php

新增課程資料

系別名稱： 電子工程系

學制： 日間部四技

課程編碼(3 碼數不可重複)： 235

課程名稱： 微電子學(二)

必選修： 必修

學分數： 3

送出

執行網頁的結果如下：

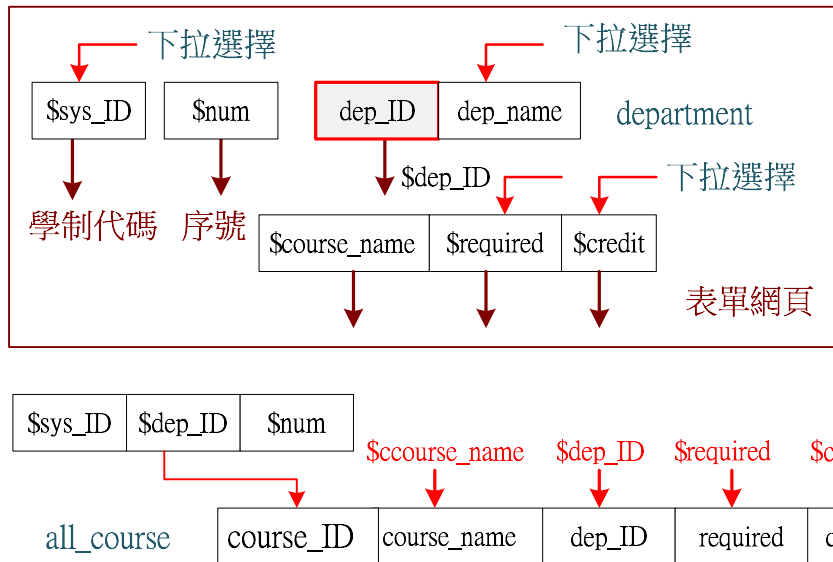
localhost/chap11/Ex11\_6-action.php

登錄新生資料如下：

課程代碼	課程名稱	系別編碼	必選修	學分數
4008235	微電子學(二)	08	必修	3

## (B) SQL 系統分析

課程資料是登錄於 `all_course` 資料表，須對它執行 `Insert into` 命令動作。下圖是 `all_course` 與 `department` 兩張資料表之間關聯，它們是利用 `dep_ID` 來識別哪一系所規劃的課程。我們在查詢網頁上製作下拉式選單，並傳送 `dep_ID` 訊息給直行網頁。在執行網頁方面，其課程代碼(`course_ID`) 編碼方式是：學制(`sys_ID`，日間四技=40、進修部四技=90)+系別代碼(如資管系= 11)+課程編碼(如網路程式設計=002，每一系編碼不可重複) (請參閱第三章說明)。



### (C) 網頁設計技巧

吾人依照上述 SQL 分析，網頁設計如下：

表單網頁 (Ex11\_6-form.php)

由 department 查詢系所名稱名稱提供下拉式選單，再製作各項參數選單或直接輸入

處理網頁 (Ex11\_6-action.php)

組合 \$course\_ID，與接收資料插入 all\_couse 表，並顯示結果

Post : sys\_ID, num, dep\_ID, course\_name, required, credit

### (D) 表單網頁的程式範例：Ex10\_6-form.php

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <head>
3 <title>新增課程資料</title></head>
4 <body>
5 <p> 新增課程資料 </p>
6 <form name="表單" method="post" action="Ex11_6-action.php">
7 <br>
8 <br>
9 <br>
10 <select name = "dep_ID">
11 <?php
12 <br>
13 <br>
14 <br>

```

```
15 // 查詢系別名稱
16
17 $sql_query = "select *
18             from department";
19 $result = mysqli_query($select_db, $sql_query);
20 while($row=mysqli_fetch_row($result)) {
21     echo "<option value='$row[0]'> $row[1] </option><BR>";
22 }
23 echo "</select>";
24 echo "<BR><BR>";
25 mysqli_close($select_db);
26 ?>
27 學制：
28 <select name="sys_ID">
29     <option value = "40"> 日間部四技 </option>
30
31     <option value = "90"> 進修部四技 </option>
32 </select> <BR><BR>
33
34 課程編碼(3 碼數不可重複)： <input name="num" type="text"><BR><BR>
35
36 課程名稱： <input name="course_name" type="text"><BR><BR>
37
38 必選修： <select name="required">
39     <option>必修</option>
40
41     <option>選修</option>
42 </select><BR><BR>
43 學分數： <select name="credit">
44     <option> 1 </option>
45     <option> 2 </option>
46     <option> 3 </option>
47     <option> 4 </option>
48 </select><BR><BR>
49 <input type="submit" value="送出">
50
51 </form>
```

```
52 </body>
    </html>
```

### (D) 執行網頁的程式範例：Ex10\_6-action.php

```
1 <?php
2     $sys_ID = $_POST["sys_ID"];
3     $dep_ID = $_POST["dep_ID"];
4     $num = $_POST["num"];
5     $course_name = $_POST["course_name"];
6     $required= $_POST["required"];
7     $credit = $_POST["credit"];
8     $course_ID = $sys_ID.$dep_ID.$num; //連結產生課程代碼
9
10    $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
11        or die("MySQL 伺服器連結失敗 <br>");
12
13    $sql_query = "insert into all_course values(
14                $course_ID, '$course_name', '$dep_ID',
15                '$required', '$credit)";
16
17    $result = mysqli_query($select_db, $sql_query);
18
19    // 測試是否登錄成功
20    if (!$result){
21        echo "課程登錄登錄失敗(可能編碼重複·請進入 AppServ 觀察)! <ER>";
22        exit();
23    }
24
25    // 查詢新增課程資料
26
27    $sql_query = "select *
28                from all_course
29                where course_ID = $course_ID";
30
31    $result = mysqli_query($select_db, $sql_query);
32
33    echo "新增課程資料如下 : <BR><BR>";
34
35    $items = array("課程代碼", "課程名稱", "系別編碼", "必選修", "學分數");
36
37    echo "<table border = '2'><tr align='center'>";
38
39    // 利用 foreach 迴圈印出陣列 $item
```

```
33     foreach ($items as $value) {
34         echo "<td> $value </td>";
35     }
36     echo "</tr>";
37     while($row=mysqli_fetch_row($result)) {
38         echo "<tr>";
39         for($j=0; $j<mysqli_num_fields($result); $j++) {
40             echo "<td>$row[$j]</td>";
41         }
42         echo "</tr>";
43     }
44     echo "</table>";
45     mysqli_close($db_link);
?>
```

## 11-6 資料刪除 – delete from

### 11-6-1 Delete from 語法

刪除資料表中某一筆或多筆資料，則使用 `delete from` 命令，到底是一筆或多筆，是看是否滿足 `where` 條件多寡，格式如下：

```
Delete from 資料表名稱
Where < 條件敘述 >;
```

只要滿足 `where` 的條件的記錄全部刪除。

### 11-6-2 範例研討 – 刪除學生資料

#### (B) 系統功能：Ex11\_7

學生可能休退學，請製作一個可刪除學生資料的網頁，期望查詢與執行結果的網頁如下：



← → ↻ ⓘ localhost/chap11/Ex11\_7-form.html

應用程式 單車活動 PBP 活動 ACP Google

刪除休退學生

學號：

執行成功則：

← → ↻ ⓘ localhost/chap11/Ex11\_7-action.php

應用程式 單車活動 PBP 活動 ACP Google

該生 40010145 刪除成功

執行失敗則：

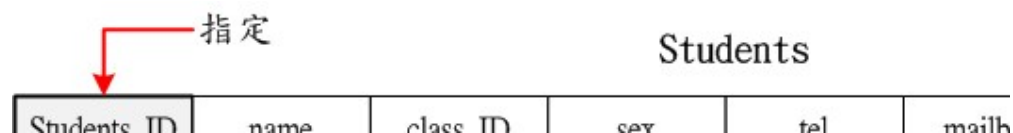
← → ↻ ⓘ localhost/chap11/Ex11\_7-action.php

應用程式 單車活動 PBP 活動 ACP Google

沒有 40010145 學生資料，請至 AppServ 觀察

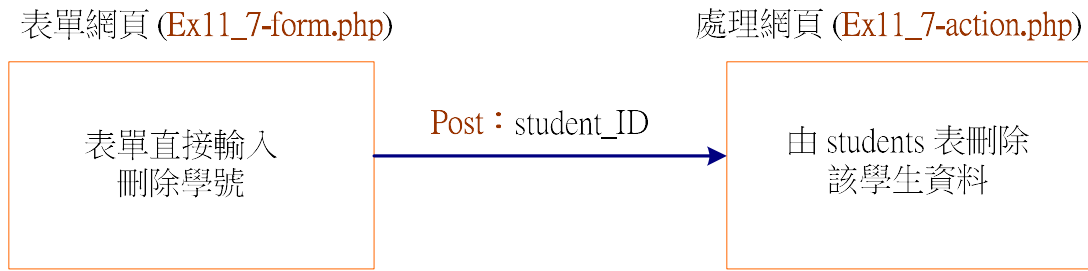
## (B) SQL 系統分析

欲刪除某位學生的資料，最好的方法是依照他的學號 (where student\_ID = “學號”)。因此，在查詢網頁提供輸入學號，執行網頁依照 student\_ID 刪除後，再查詢該學生是否真正被刪除。



## (C) 網頁製作技巧

依照上述 SQL 系統分析，網頁設計如下：



### (D) 表單網頁的程式範例：Ex11\_7-form.html

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <head>
3 <title>刪除休退學生</title></head>
4 <body>
5     <p> 刪除休退學生 </p>
6
7     <form name="表單" method="post" action="Ex11_7-action.php">
8
9         學 號 : <input name = "student_ID" type = "text"> <BR><BR>
10
11        <input type="submit" value="送出">
12
13    </form>
14 </body>
</html>
  
```

### (D) 執行網頁的程式範例：Ex11\_7-action.php

```

1 <?php
2     $student_ID = $_POST["student_ID"];
3     $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
4         or die("MySQL 伺服器連結失敗 <br>");
5     $sql_query = "select *
6                 from students
7                 where student_ID = '$student_ID'";
8     $result = mysqli_query($select_db, $sql_query);
9     if(mysqli_fetch_row($result)) {
10         $sql_query = "delete
11
  
```

```
12         from students
13         where student_ID = '$student_ID';
14     $result = mysqli_query($select_db, $sql_query);
15     echo "該生 $student_ID 刪除成功 <BR>";
16 } else {
17     echo "沒有 $student_ID 學生資料，請至 AppServ 觀察 <BR>";
18 }
19 }
20 mysqli_close($db_link);
21 ?>
```

## 11-7 資料更新 – Update

### 11-7-1 Update 語法

新資料取代現有資料表的內容，稱之為『更新』( Update )，但資料表內有多筆資料，到底要更新哪一筆 ( 或稱紀錄 ) 或多筆資料同時更新，因此需要有一個 Where 敘述句來決定更新哪一筆或哪些資料。格式如下：

```
Update 資料表名稱
Set [<欄位_1>=<數值_1>, ...<欄位_2>=<數值_2>]
Where <條件敘述>;
```

上述語句中，只要某一筆滿足 Where 條件敘述，則更新它指定欄位的內容。

### 11-7-2 範例研討 – 更新學生資料

#### (A) 系統功能：Ex11\_8

系統需要一個可以修改學生資料的工具，期望表單網頁如下：

localhost/chap11/Ex11\_8-form.html

應用程式 單車活動 PBP 活動 ACP Google

修改學生資料

學號：

執行網頁的結果如下：

localhost/chap11/Ex11\_8-action-1.php

應用程式 單車活動 PBP 活動 ACP Google 北海道旅遊

更新 90211304 學生資料

學生姓名：

班級代碼：902113 (轉班另外處理)

性別：

電話：

電子郵件：

地址：

更新網頁的結果如下：

localhost/chap11/Ex11\_8-action-2.php

應用程式 單車活動 PBP 活動 ACP Google 北海道旅遊 法國旅行計畫 Bookmarks

更新學生資料如下：

學號	姓名	班級代碼	性別	電話	郵件	地址
90211304	劉雅玲	902113	女	077313214	stu8@gmail.com	高雄市苓雅區中正路 245 號

執行失敗則顯示如下：

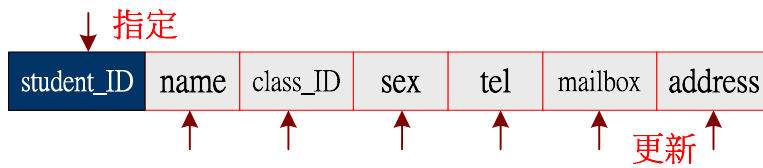
localhost/chap11/Ex11\_8-action-1.php

應用程式 單車活動 PBP 活動 ACP Google

沒有 90211301 學生資料，請至 AppServ 觀察

## (B) SQL 系統分析

修改學生的資料須由 students 表執行，如下：



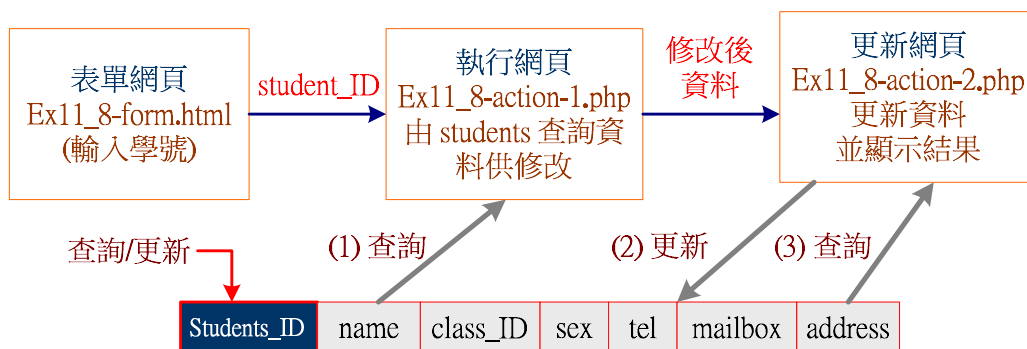
SQL 程式如下：

```
update students
set name = '$name', class_ID = '$class_ID', ...
where student_ID = '$student_ID';
```

### (C) 網頁製作技巧

如欲更新某一位學生資料，首先須先顯示該生資料後，再依其內容修改，修改後再更新其內容，因此需要三個網頁，如下：

- (1) 表單網頁(Ex11\_8-form.html)：**提供輸入欲修改學生的學號(student\_ID)，再傳送給執行網頁。
- (2) 執行網頁(Ex11\_8-action-1.php)：**利用表單網頁傳遞過來的 student\_ID，由 students 表中查詢出該生目前的資料，提供修改參考，修改後將修改後資料傳遞給更新網頁。
- (3) 更新網頁(Ex11\_8-action-2.php)：**接收執行網頁所傳遞過來的訊息，執行更新命令修改其資料，並將結果顯示出來。



**(D) 表單網頁的程式範例：Ex11\_8-form.html**

此網頁僅提供輸入欲修改資料學生的學號，如下：

```
1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <head>
3 <title>修改學生資料</title></head>
4 <body>
5     <p> 修改學生資料 </p>
6
7     <form name="表單" method="post" action="Ex11_8-action-1.php">
8
9         學 號：<input name = "student_ID" type = "text"> <BR><BR>
10
11         <input type="submit" value="送出">
12
13     </form>
14 </body>
</html>
```

**(D) 執行網頁的程式範例：Ex11\_8-action-1.php**

此網頁讀取該生的資料，經修改後，再傳送給更新網頁，如下：

```
1 <?php
2     $student_ID = $_POST["student_ID"];
3     $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
4         or die("MySQL 伺服器連結失敗 <br>");
5     $sql_query = "select *
6                 from students
7                 where student_ID = '$student_ID'";
8     $result = mysqli_query($select_db, $sql_query);
9     $row = mysqli_fetch_row($result);
10    if(!$row) {
11        echo "沒有 $student_ID 學生資料，請至 AppServ 觀察 <BR>";
12        exit();
13    }
14
15    echo "<p> 更新 $student_ID 學生資料 </p>";
16
```

```
17     echo "<form name='表單' method='post' action='Ex11_8-action-2.php'>";
18         echo "<input name='student_ID' value=$row[0] type='hidden'><BR>";
19     echo "學生姓名：<input name='name' value=$row[1] type='text'><BR>";
20
21     echo "班級代碼：$row[2] (轉班另外處理) <BR>";
22
23     echo "性別：<input name='sex' value=$row[3] type='text'><BR>";
24
25     echo "電話：<input name='tel' value=$row[4] type='text'><BR>";
26
27     echo "電子郵件：<input name='mailbox' value=$row[5] type='text'><BR>";
28
29     echo "地址：<input name='address' value=$row[6] type='text' size='60'><BR>";
30
31     echo "<input type='submit' value='送出'>";
32
33     echo "</form>";
34     mysql_close($db_link);
35 ?>
```

### (E) 更新網頁的程式範例：Ex11\_8-action-2.php

接收修改後資料，並寫入該生的記錄內，再讀取顯示其結果，如下：

```
1 <?php
2     $student_ID = $_POST["student_ID"];
3     $name = $_POST["name"];
4     $sex = $_POST["sex"];
5     $tel = $_POST["tel"];
6     $mailbox = $_POST["mailbox"];
7     $address = $_POST["address"];
8     $select_db = mysqli_connect("localhost", "root", "12345678", "course_db")
9         or die("MySQL 伺服器連結失敗 <br>");
10
11     $sql_query = "update students
12                 set name='$name', sex='$sex', tel='$tel',
13                 mailbox='$mailbox', address='$address'
14                 where student_ID = '$student_ID'";
15     $result = mysqli_query($select_db, $sql_query);
16
17     // 測試是否更新成功
18
19     if (!$result){
20         echo "學生資料更新失敗(請進入 AppServ 觀察) !<ER>";
21         exit();
22     }
```

```
20     }
21     // 查詢學生更新資料
22     $sql_query = "select *
23                 from students
24                 where student_ID = $student_ID";
25     $result = mysqli_query($select_db, $sql_query);
26     echo "更新學生資料如下 : <BR><BR>";
27
28     $items = array("學號", "姓名", "班級代碼", "性別", "電話", "郵件", "地址");
29
30     echo "<table border = '2'><tr align='center'>";
31     // 利用 foreach 迴圈印出陣列 $item
32     foreach ($items as $value) {
33         echo "<td> $value </td>";
34     }
35     echo "</tr>";
36     while($row=mysqli_fetch_row($result)) {
37         echo "<tr>";
38         for($j=0; $j<mysqli_num_fields($result); $j++) {
39             echo "<td>$row[$j]</td>";
40         }
41         echo "</tr>";
42     }
43     echo "</table>";
44     mysql_close($db_link);
45     ?>
```

## 11-8 自我挑戰 – DML

### 11-8-1 自我挑戰：新增教師

#### (A) 系統功能：PM11\_1

請在 course\_db 資料庫上建立一只可供登錄教師資料的網頁，期望如下：

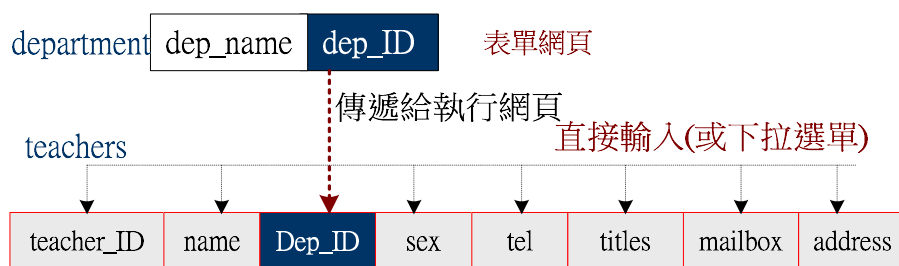
- 『表單網頁』：可供下拉式選擇系所名稱(系所代碼)，直接輸入教師代碼(不可重複)、姓名、性別(下拉選單)、職稱(下拉選單)、電話、電子郵件與地址。



- 『執行網頁』：登錄教師資料並顯示該教師詳細資料。

## (B) SQL 系統分析

只須對 teachers 表做 Insert into 的動作，但教師是屬於哪一系所的 dep\_ID 是須由 department 查詢所，下圖是 teachers 與 department 兩張資料表之間的關聯，教師的 dep\_ID 也是由查詢網頁，由使用者選擇系別名稱產生。另外，吾人希望性別 (sex) 提供男與女的選單，職位(title)也提供教師、系主任、助教的選單。



## 11-8-2 自我挑戰：新增班級

### (A) 系統功能：PM11\_2

請在 `course_db` 資料庫上建立一只可供登錄班級的網頁，期望功能如下：

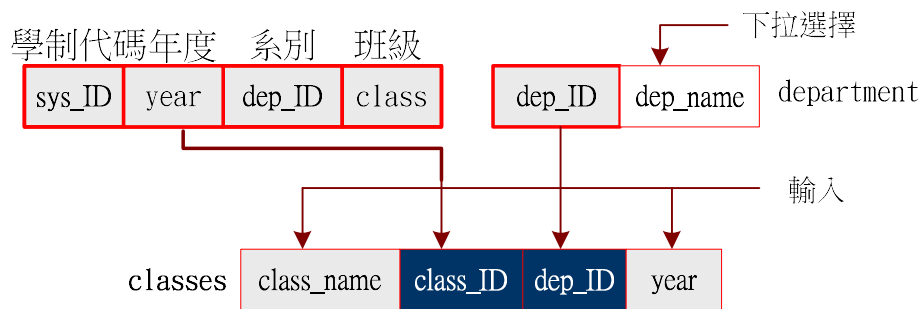
- 『表單網頁』：可供輸入：學制(下拉式選擇)、系所名稱(系所代碼、由 `department` 表查詢，下拉式選擇)、班級編排(甲、乙班，下拉式選擇)、班級名稱(直接輸入)、入學年度(103~109，下拉式選擇)，再傳送給執行網頁。
- 『執行網頁』：登錄班級資料並顯示該班詳細資料。

## (B) SQL 系統分析

班級資料是登錄於 `classes` 資料表內，須對它做 Insert into 的動作。下圖是 `classes` 與 `department` 兩資料表之間的關聯，它們之間是利用 `dep_ID` 來建立關係，也表示每一班級是屬於哪一系。其中比較麻煩的是 `class_ID` 的編碼，它總共有 6 碼，其結構是學制(1 碼)+入學年

度 (2 碼)+系別(2 碼) + 班級(1 碼)·學制是日間部四技(4)·進修部四技(9)·入學年度是 102(02)、103(03)、104(04)、依此類推·班級是甲班(1)、乙班(2)、丙班(3)、依此類推。

我們預定由查詢網頁開啟 department 表，並提供下拉式選擇系所名稱，再傳遞 dep\_ID 給執行網頁，另外也提供輸入班級名稱(class\_name)、入學年度(year)、學制(sys\_ID)、班級(class) 介面，執行網頁再將它連結組合成班級代碼(class\_ID)，再一併插入 classes 資料表。



### 11-8-3 自我挑戰：修改班級資料

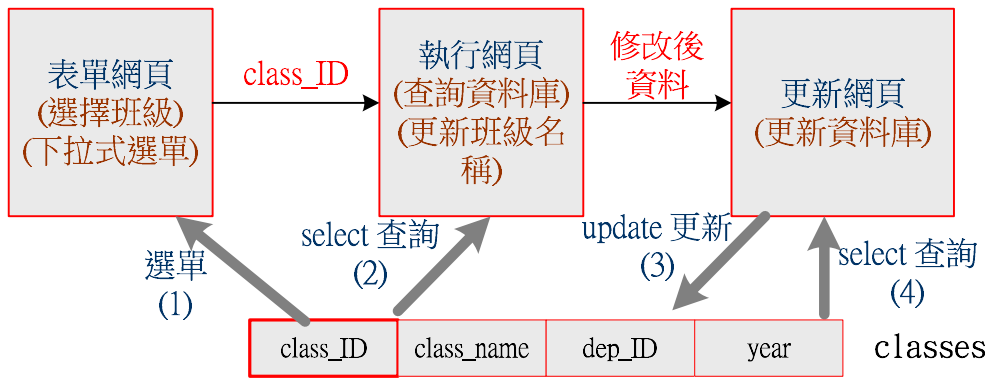
#### (A) 網頁功能：PM11\_3

請在 course\_db 資料庫上建立一只可供登班級名稱的網頁，期望功能如下：

- 『表單網頁』：可供下拉式選擇欲修改的班級名稱(由 classes 表查詢)、並將班級編碼(class\_ID) 傳送給執行網頁。
- 『執行網頁』：由 classes 表讀出該班的班級名稱，以供修改，修改後再傳送給更新網頁。
- 『更新網頁』：更新該班級名稱，並顯示結果。

#### (B) 系 SQL 系統分析

在 classes 表內除了 class\_name 欄位可以任意變更外，其餘欄位都不可隨意變更，因此關聯圖如下：



### 11-8-4 自我挑戰：修改教師資料

#### (A) 網頁功能：PM11-4

請在 course\_db 資料庫上建立一只可供修改教師資料的工具，期望功能如下：

- 『表單網頁』：可供下拉式選擇欲修改的教師姓名(由 teachers 表查詢)、並將教師編碼 (teacher\_ID) 傳送給執行網頁。
- 『執行網頁』：由 teaches 表讀出該教師的資料以供修改(系別代碼不得更改)，修改後再傳送給更新網頁。
- 『更新網頁』：更新該教師資料，並顯示結果。

#### (B) SQL 系統分析

這題只針對 teachers 表做更新動作即可 (dep\_ID 欄位不可更新)，程序如下圖所示：

