# 自動寫入當前日期時間

1. 在 INSERT 資料時，讓 MySQL 自動將當前的日期時間寫入，如下圖：



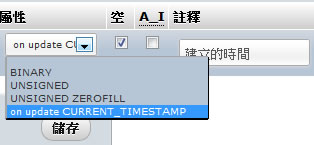
1. 觀察資料表結構:

a.類型選擇 TIMESTAMP (時間戳)

b.預設選擇 CURRENT\_TIME (當前時間)



1. 屬性選擇 on update CURRENT\_TIMESTAMP (當新增資料時使用當前時間戳)



1. 接受空值，可改為不接受



1. 語法設定方式:

|  |
| --- |
| **create** **table** **manage\_template** (  manage\_template\_name **int**,  manage\_template\_value **varchar**(**20**),  manage\_template\_datetime **timestamp** **NOT** **NULL** **DEFAULT** **CURRENT\_TIMESTAMP** **ON** **UPDATE** **CURRENT\_TIMESTAMP**  ) |

1. 如此一下，在 PHP 程式中，不再需要去抓取目前時間，直接新增資料即可，寫入的時間 MySQL 會自行處理。

|  |
| --- |
| <?php  $dbh = Dbo::getInstance()->get\_dbh();  $sql = sprintf("INSERT INTO `manage\_template`(manage\_template\_name,manage\_template\_value) VALUES(?,?)");  $sth = $dbh->prepare($sql);  $sth->bindValue(**1**,$add\_template\_name,PDO::PARAM\_STR);  $sth->bindValue(**2**,$template,PDO::PARAM\_STR);  $sth->execute();  ?> |

1. 注意，屬性為 on update CURRENT TIMESTAMP，意思為當你「修改」這筆資料時，日期時間也會重新被寫入！所以若是要記錄「建立」的時間，可以使用。  
   若是有可能會去 update 某筆資料，而不想更動原先的「建立」時間，則不要將屬性設為 on update CURRENT TIMESTAMP，改由程式抓取並寫入。

# AUTO\_INCREMENT 自動累加

1. 使用資料庫存放記錄時，經常會想給該筆記錄一個獨一無二的編號，像是給它一個ID或是一個Index等等，方便日後顯示資料、搜尋或是作為兩張資料表 (Table) 連結在一起的key。

但每次都手動給他一個值，這樣實在太不方便了！因此我們可以使用

AUTO\_INCREMENT 來幫助我們。

1. 建立一個記錄用戶資訊的資料表，欄位包含編號、人名、電話及年紀，將ID設為Primary key，並給予自動累加的功能。：

|  |
| --- |
| **CREATE** **TABLE** **User** (  ID **int** **NOT** **NULL** **AUTO\_INCREMENT**,  Name **varchar**(**50**) **NOT** **NULL**,  Telephone **varchar**(**20**),  Age **int**,  **PRIMARY** **KEY**(ID)  ); |

1. 接著加入資料進入這張資料表，只要這樣做：

|  |
| --- |
| **INSERT** **INTO** **User** (Name, Telephone, Age)  **VALUES** ('Ian', '0987654321', '24'),  ('Rex', '0912345678', '20'),  ('Ash', '0988776655', '18'); |

不需要對 ID 給值，他會自動從一開始設定的值 (預設是1) 一直累加上去。

1. 資料庫會長這樣：

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Telephone | Age |
| 1 | Ian | 0987654321 | 24 |
| 2 | Rex | 0912345678 | 20 |
| 3 | Ash | 0988776655 | 18 |

1. 當把記錄刪除，設定AUTO\_INCREMENT的欄位也不會回復，會再繼續加上去，舉個例子，刪掉1筆資料，再加入1筆資料：

|  |
| --- |
| **DELETE** **FROM** User **WHERE** Name = 'Ash';  **INSERT** **INTO** **User** (Name, Telephone, Age)  **VALUES** ('Roy', '0911223344', '30'); |

1. 資料庫會長這樣：

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Telephone | Age |
| 1 | Ian | 0987654321 | 24 |
| 2 | Rex | 0912345678 | 20 |
| 4 | Roy | 0911223344 | 30 |

ID 3 就再也看不見。