# 資料庫 – 增刪查改(CRUD)

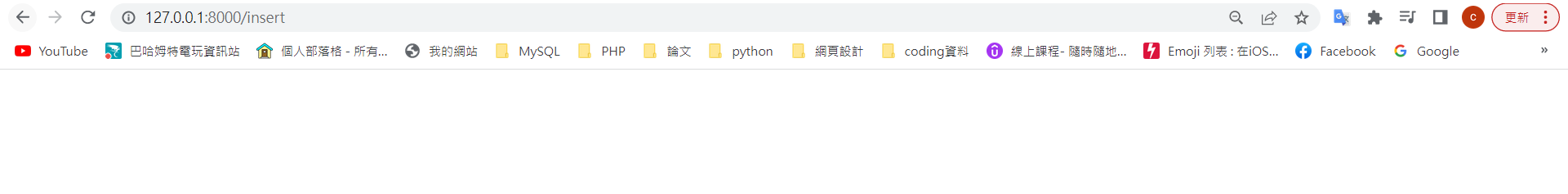
## Inserting Data

Route::get('/insert', function() {

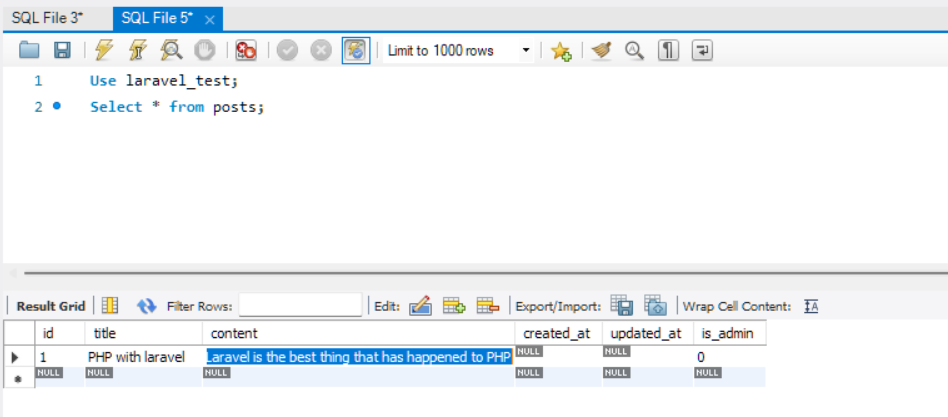
  DB::insert('insert into posts(title, content) value(?, ?)', ['PHP with laravel', 'Laravel is the best thing that has happened to PHP']);

});

在web.php中，新增MySQL的語法如上，可以將資料插入posts表格中，並且到localhost:8000/insert確認是否成功



到insert頁面後，網頁沒有任何東西 => 因為沒有return任何物件到網站中，因此是正常的



到MySQL WorkBench確認後，發現插入的資料有顯示到posts該表格中

## Reading Data

Route::get('/read', function() {

  $result = DB::select('Select \* from posts where id = ?', [1]);

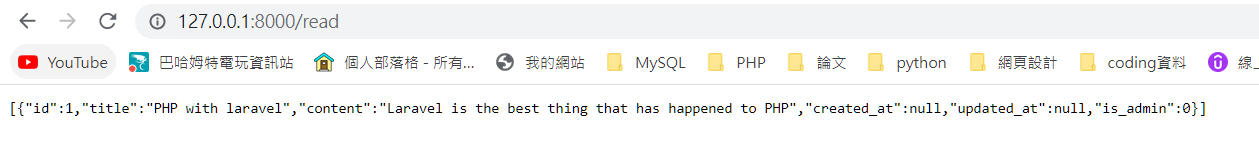
  foreach($result as $post){

    return $post->title.'<br>'.$post->content;

  };

});

使用select的語法，將需要選取的資料搜尋出來



而return $result;的結果為array值，內容包覆著Object

因此需要使用foreach去拆解array，並且使用箭頭指向需要回傳的$key值。



$post->title; 使用Object的內容指向title的key，回傳title的value。

$post->content也做一樣的動作。

## Updating Data

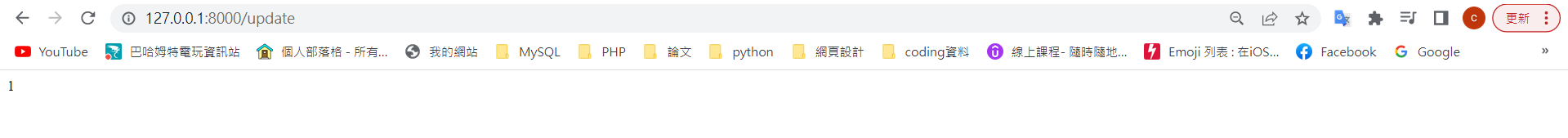
Route::get('/update', function() {

  $updated = DB::update('update posts set title = "Updated title" where id = ?', [1]);

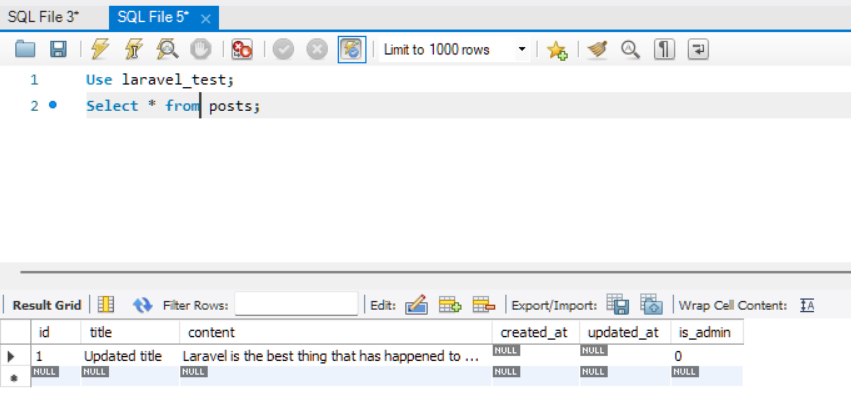
  return $updated;

});

使用update語法，將需要更新的資料的表格選取後，設定該表格欄位的內容和需要修改資料的確切資訊。



當前往update的網頁時，會回傳1的內容。



此時到MySQL WorkBench也可以看到title的內容有被變更

## Deleting Data

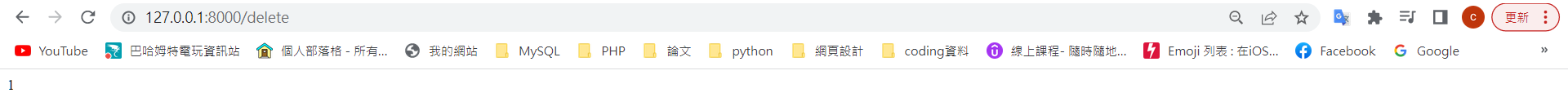
Route::get('/delete', function() {

  $deleted = DB::delete('Delete from posts Where id = ?', [1]);

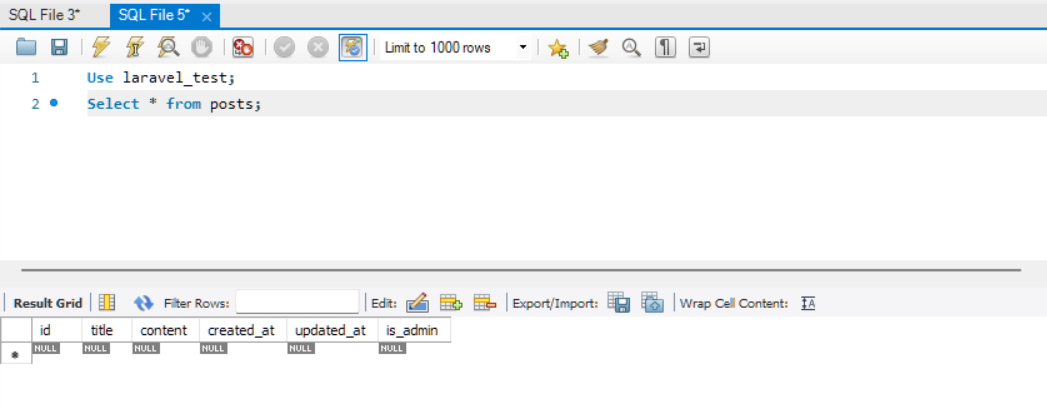
  return $deleted;

});

使用delete的MySQL語法，先選定要刪除的表格是哪個，再定義要刪除的是哪筆資料，最後回傳至網頁中。



這邊的1為影響1筆資料



該表格的內容也空空如也