



# 網路程式設計實習

Unit13 - SQL SELECT & INSERT

班級：資訊二乙

座號：26

姓名：陳維基



## 單元內容描述

1. Connect to MariaDB
2. SQL
3. SQL Select
4. SQL Select (Advanced)
5. SQL Insert

# 1 Connect to MariaDB

我們可以透過同為在後端的PHP 來讓我們接取資料庫的資料，與資料庫建立一開始連結的方法為 `$con = new mysqli("伺服器", "帳號", "密碼", "資料庫")` 來建立連結物件，接著透過 `$con -> query()` 來設定連接資料庫的相關參數，再透過 `$con -> prepare()` 將 sql 送入資料庫，就可以執行了，執行是透過 `$exe = $con -> execute()` 執行指令，而執行完的結果可以用 `$res = $exe -> get_result()` 取出，再透過 `$res -> fetch_assoc()` 就可以將資料分成一筆一筆的了



# 1 Connect to MariaDB

所以可以整理出以下流程：

1. `new mysqli()` 建立連接物件
2. `-> query()` 設置連接相關設定
3. `-> prepare()` 送入 SQL 指令
4. `-> execute()` 來執行指令
5. `-> get_result()` 取得執行結果
6. `-> fetch_assoc()` 將資料一一取出



## 2 SQL

SQL 是一種操作資料的語言，可以對資料庫進行查詢、編輯、統計等功能，甚至可以新增資料表、變更資料表結構。

SQL 也有內建的函數可以使用，可以使使用者對資料庫的操作更靈活。

SQL Function	說明
AVG	求平均
SUM	求總和
COUNT	求數量

### 3 SQL Select

針對資料庫的查詢，最常使用的就是Select 子句，在搭配Where 就可以依條件來找出符合的資料，Select 中也常搭配\*，這表示選擇所有欄位。

功能	SQL
查詢 table1 的所有資料	Select * from table1
查詢 table1 中 cloumn1 的資料	Select column1 from table1
查詢 table1 中 cloumn1 和 column2 的資料 ， 且 column1 上的資料大於 60	Select column1, column2 from table1 where column1 > 60

## 4 SQL Select (Advanced)

Select 除了一般的查詢外，也可以對查詢出來的資料排序完再輸出結果，或是多補充幾個欄位來統計資料等。

功能	SQL
查詢 table1 的所有資料，並依 column1 從小排到大	Select * from table1 order by column1 asc
查詢 table1 的所有資料，並依 column1 從大排到小，再依 column2 從小排到大	Select * from table1 order by column1 desc, column2 asc
查詢 table1 的所有資料，並增加一個欄位為 column1 與 column2 的總和，並命名為 total	Select *, (column1+column2) as total from table1



## 5 SQL Insert

除了查詢，我們還會很常對資料庫進行新增資料，而SQL新增的指令就是Insert，基本上就是指定資料表、欄位以及數值，不過要注意的一點是如果欄位是主鍵，或是不可為null的話，送入不符合其規則的資料就會導致無法成功執行指令。

Insert 語法：

```
Insert Into [table] (column1, column2...) Values (value1, value2...)
```





## 雜記

1. Get data from database by JavaScript



## 1 Get data from database by JavaScript

資料主要是透過PHP來接取，不過這次我試著透過JavaScript向PHP送request來接取資料，就可以實現JavaScript跟PHP溝通。送request的方法就是透過XMLHttpRequest()這個物件，他可以指定要送request的網址以及資料，就可以從JavaScript向PHP傳遞資料，而傳遞的資料是透過FormData()這個物件，而傳遞出去的資料基本上就是Json格式那樣key對value。而PHP的功能就會變得像API，且接到的資料會是Json格式，所以在資料的處理上相對容易。

# 1 Get data from database by JavaScript

利用 JavaScript 向 PHP 傳遞 request。

```
function getAll() {  
    var data = new FormData()  
    data.append("mode", "all")  
    var req = new XMLHttpRequest()  
    var url = "select.php"  
    req.open("POST", url, true)  
  
    req.onreadystatechange = () => {  
        if (req.readyState == 4 && req.status == 200){  
            console.log(req.responseText)  
            db = JSON.parse(req.responseText)  
            refresh()  
        }  
    }  
    req.send(data)  
}
```



## 1 Get data from database by JavaScript

這種作法基本上就是跟Form 確認時的作用一樣，而優點是他並不會重新導向至其他頁面，所以可以將接取資料的PHP 與畫面分開，使用者體驗上也會比較佳。



## 心得

在這周所作的新功能中, PHP 所扮演的角色基本上就是API, 我就只要向其送資料就可以得到相對應的資料, 在處理龐大的程式碼時也會很方便。在選手訓練的其間也有撰寫API 或是介接API 的經驗, 所以處理這部分算是滿得心應手的。

這次整理出來的方法是想讓處理資料庫的部分便比較簡單, 且可以在不重新整理頁面的方式就達到更新畫面, 在撰寫可以省下不少時間, 且畫面的呈現上也相對乾淨。