

20.1 傳值



web

1. 傳遞參數

利用ajax的技巧將表單輸入的值呼叫php來做相關運算處理以下例子說明此動作



postdata



postdata

頁首

文字輸入傳送至php:

按鈕

由phpt程式傳回的資料: 123

頁尾

postdata.html

```

<script>
$(function(){
    $("#btn_insert").bind("click", insertItem);
});
function insertItem() {
    $.ajax({
        type: "POST",
        url: "http://localhost/EL1065/ch15/test/postdata.php",
        data: {ms1: $('#ms1').val()},
        success: ShowData,
        error: function(data){
            alert("系統錯誤!");
        },
    });
}
function ShowData(data){
    $('#showms1').html(data);
}
</script>
</head>
<body>
<div data-role="page" id="page">
    <div data-role="header">
        <h1>頁首</h1>
    </div>
    <div data-role="content">

        <label for="ms1">文字輸入傳送至php:</label>
        <input type="text" name="ms1" id="ms1" value="" />
        <input name="btn_insert" type="button" id="btn_insert" value="按鈕" />

    <div id="showms1"></div>
    </div>
    <div data-role="footer" data-position="fixed">
        <h4>頁尾</h4>
    </div>
</div>

```

postdata.php

```

<?php
    echo '由php程式傳回的資料: ' . $_POST['ms1'];
?>

```

練習:

由表單取得值利用ajax呼叫php將值寫入資料庫!

2. 傳遞string(json)

當前端有多筆參數需要傳遞給後端API時，可以先將多筆參數存入陣列並利用JSON.stringify()將陣列轉乘json格式的字串後傳遞給後端API，其中為了避免亂碼，ajax須設定 **contentType: "application/json; charset=utf-8"**。

//宣告陣列, 並將參數存放入陣列

```

var dataJSON = {};
dataJSON["username"] = $("#username").val();
dataJSON["password"] = $("#password").val();
dataJSON["email"] = $("#email").val();

```

//將陣列轉換成string必以json的格式輸出

JSON.stringify(dataJSON)

```
//傳遞json資料至後端php
//必須先轉換成字串 JSON.stringify(dataJSON)
//php 接收方法 file_get_contents("php://input",'r');
var dataJSON = {};
$(function(){
    $("#ok_btn").bind("click", function(){
        dataJSON["username"] = $("#username").val();
        dataJSON["password"] = $("#password").val();
        dataJSON["email"] = $("#email").val();

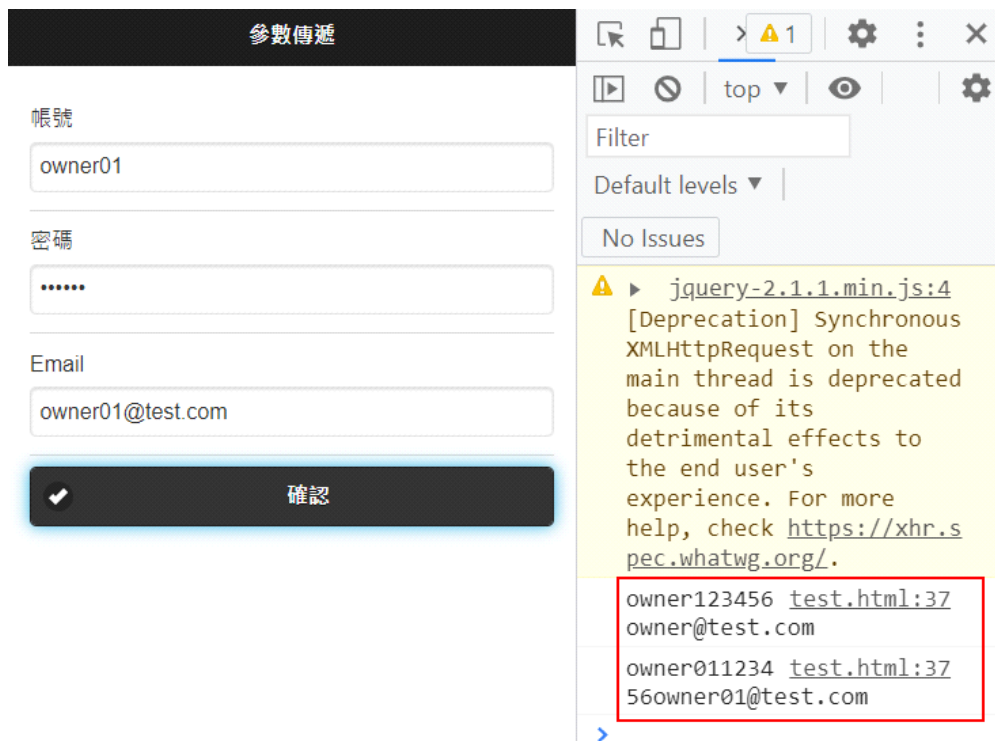
        $.ajax({
            type: "POST",
            url: "test.php",
            data: JSON.stringify(dataJSON),
            contentType: "application/json;charset=utf-8",
            //dataType: "json",
            async: false,
            success: showdata,
            error: function(){
                console.log(JSON.stringify(dataJSON));
                alert("error-test.php");
            }
        });
    });
});
function showdata(data){
    console.log(data);
}
```

後端API接收方式為`file_get_contents("php://input",'r')`
利用`json_decode($data, true)`將接收到的json樣式的字串轉換成陣列，當轉換成陣列後就可以方便的使用(資料庫相關應用)。

在用php寫接口的時候，通常會將請求的數據通過json的形式發送到指定的請求地址處，此時的`file_get_contents('php://input')`主要是用來獲取請求的原始數據。

```
<?php
//php 接收方法 file_get_contents("php://input",'r');
//var_dump()印出變數的相關訊息於螢幕上
$data = file_get_contents("php://input",'r');
$mydata = array();
//將接收的資料轉換成陣列
$mydata = json_decode($data, true);
//var_dump($mydata);
echo $mydata["username"].$mydata["password"].$mydata["email"];
?>
```

執行結果



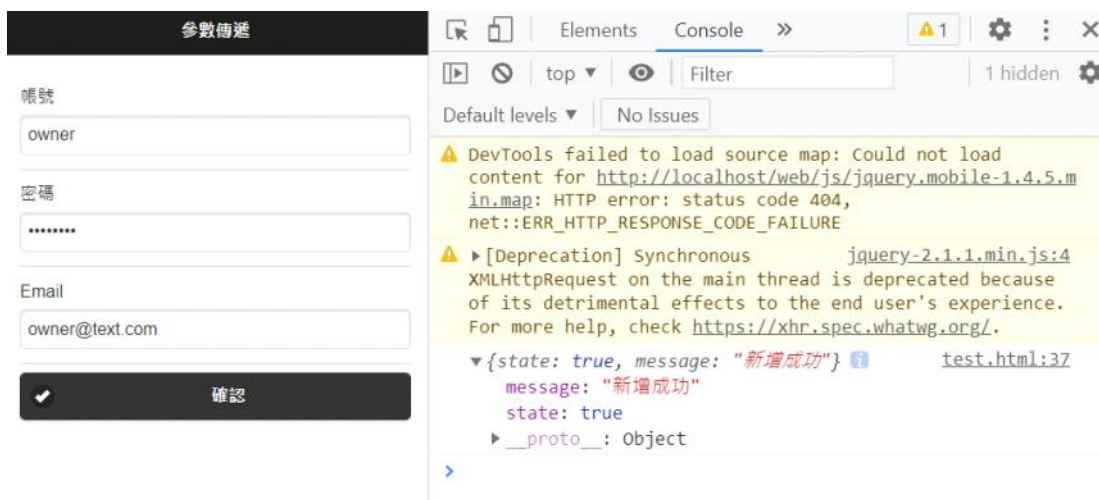
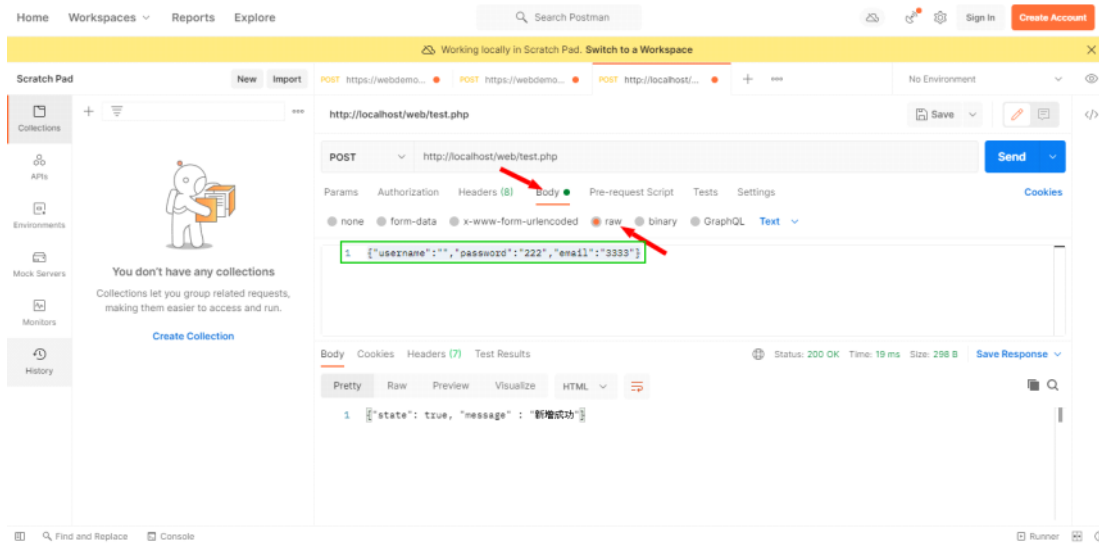
3. 連線資料庫處理

修改API將資料寫入資料庫

```
<?php
require_once("dbtools.php");
//php 接收方法 file_get_contents("php://input",'r');
//var_dump()印出變數的相關訊息於螢幕上
$data = file_get_contents("php://input",'r');
$mydata = array();
//將接收的資料轉換成陣列
$mydata = json_decode($data, true);
//var_dump($mydata);
// echo $mydata["username"].$mydata["password"].$mydata["email"];
$username = $mydata["username"];
$password = $mydata["password"];
$email = $mydata["email"];

$link = create_conn();
$sql = "INSERT INTO testdb(Username, Password, Email) VALUES('$username', '$password', '$email')";
if(exe_sql($link, "owner", $sql)){
    echo '{"state": true, "message": "新增成功"}';
}else{
    echo '{"state": false, "message": "新增失敗"}';
}
mysqli_close($link);
?>
```

使用postman測試結果



	ID	Username	Password	Email
<input type="checkbox"/>	1	aaa	ssss	dddd
<input type="checkbox"/>	2	owner	1111111	owner@text.com

加入判斷參數格式以避免空白欄位與為定義的參數

POST https://webdemo... POST https://webdemo... POST http://localhost/... + ... No Environment

http://localhost/web/test.php Save

POST http://localhost/web/test.php Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL Text

1

Body Cookies Headers (7) Test Results Status: 200 OK Time: 11 ms Size: 326 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "state": false,
3   "message": "不允許未定義的欄位或無參數"
4 }
```