

jQuery程式設計-基礎篇1

講師：詹博文

課程大綱

- jQuery 簡介
- 安裝IIS Express 環境
- Visual Studio Code 編輯器介紹
- JavaScript 語法介紹
- JQuery 語法介紹

jQuery 簡介

- 問世於2006年
- jQuery 是一套JavaScript 函式庫
- 少寫些,多做些
- [jQuery](#)官網
- [jQuery](#) 參考網站

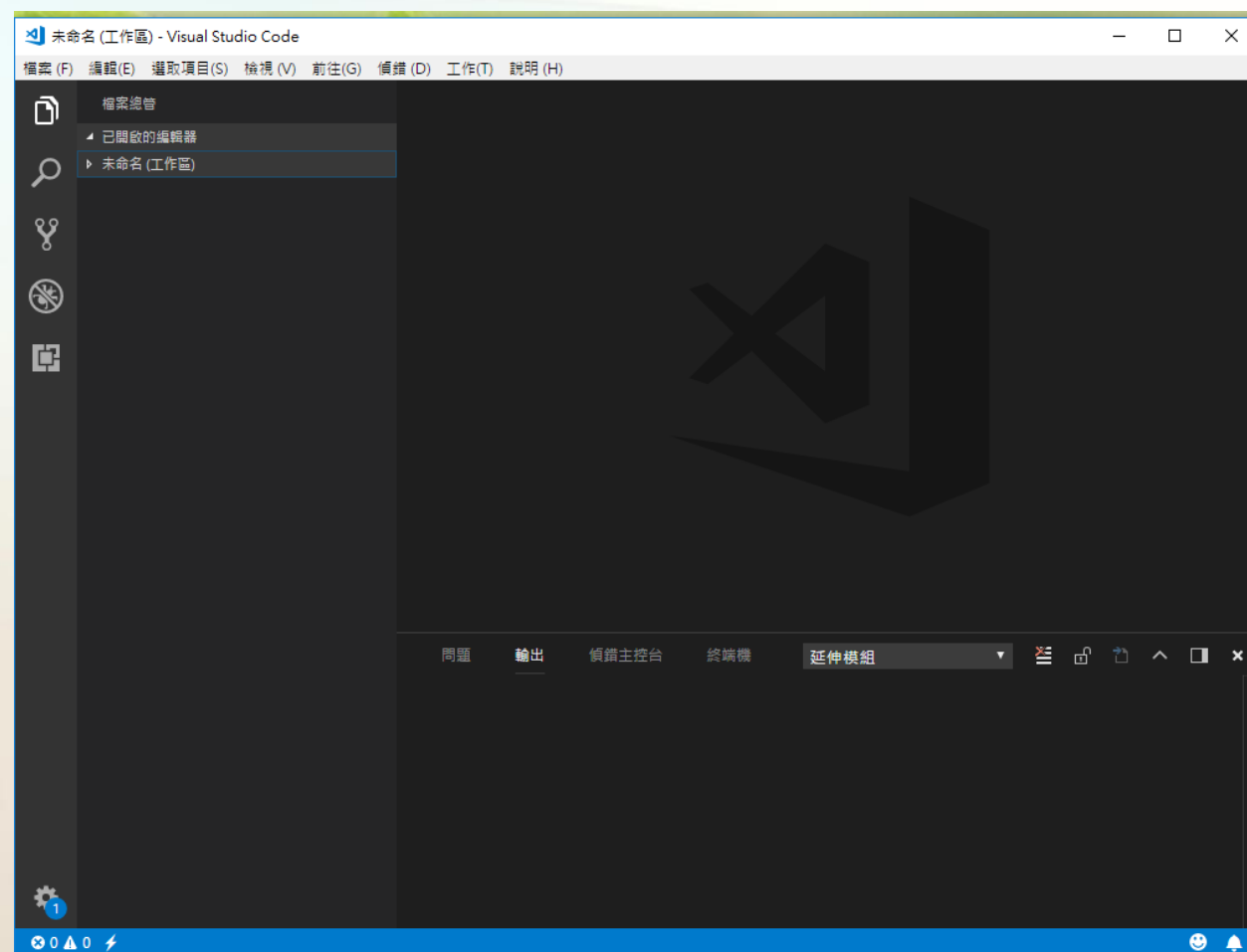
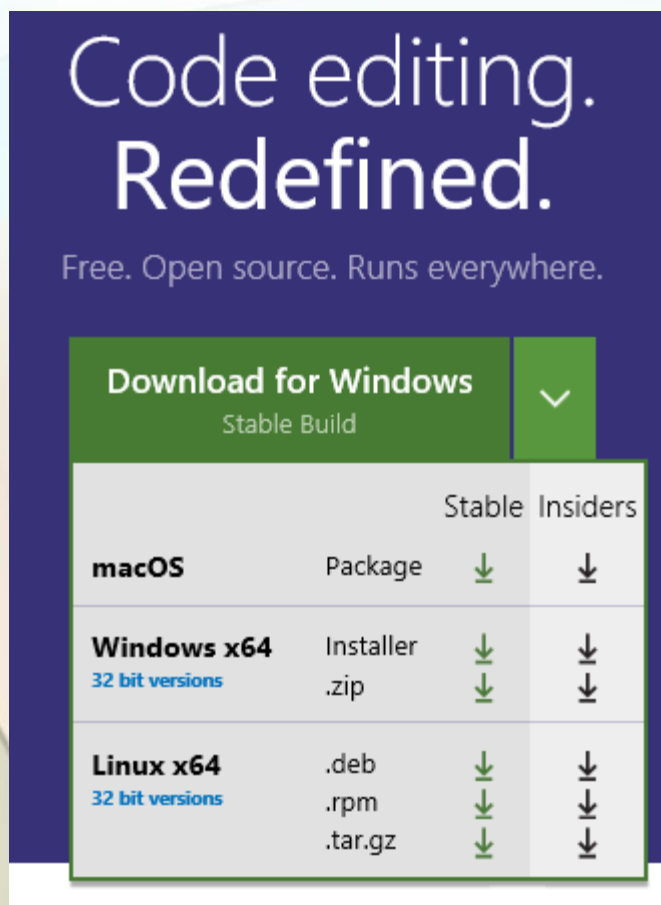
IIS Express 網頁測試環境建置

1. 下載 [IIS Express](#)

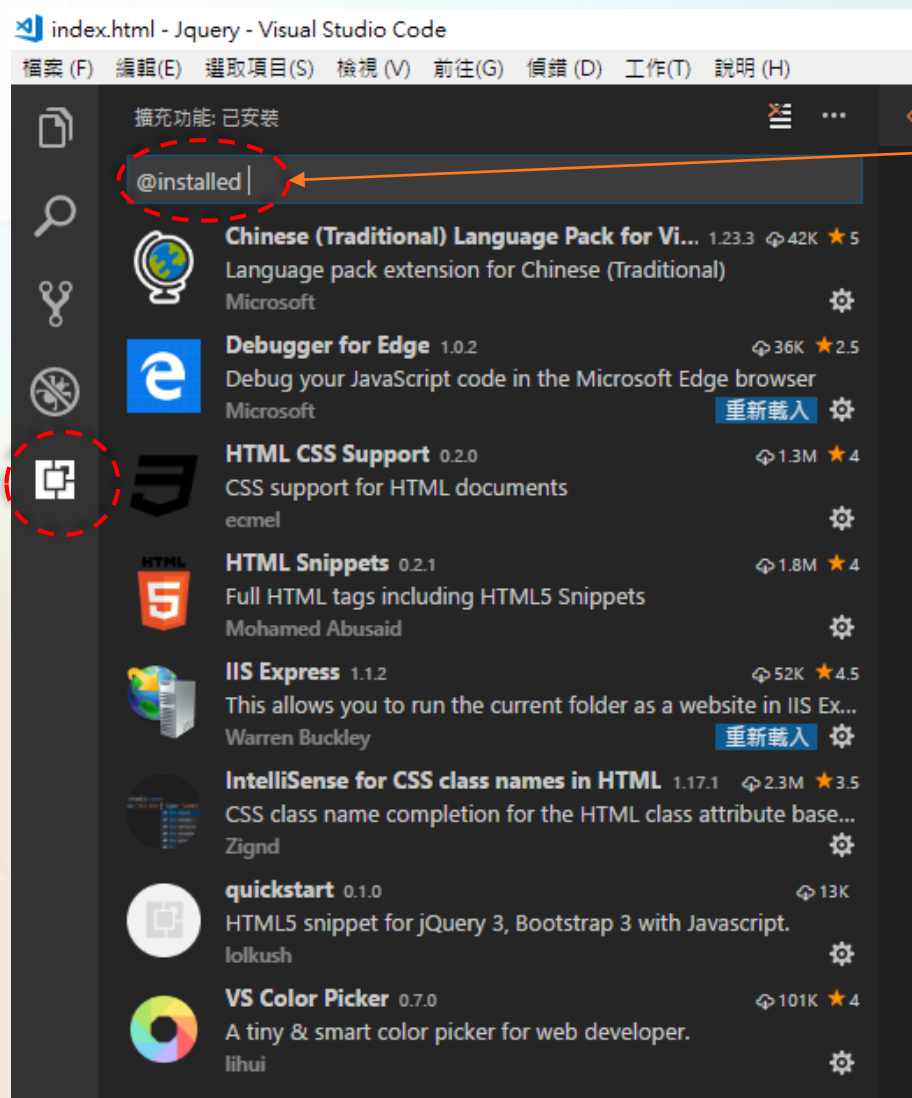
2. 執行 iisexpress_amd64_zh-TW.msi

Visual studio code 編輯器安裝

- 1. 下載 並安裝 [visual studio code](#)



Visual studio code 安裝 擴充元件



輸入關鍵字尋找要
安裝的元件

JavaScript語法介紹-變數與陣列

- 變數：`var` 變數名稱;
`var` 變數名稱=`值`;
- 陣列：`var` 陣列名稱=`new Array`(`索引值`) 索引值從0開始，也可不指定索引值。
 - 直接給值：`var fruit=[ "蘋果" , " 橘子" , " 葡萄" ]`
 - 取值：`var f1=fruit[0];`
 - 開頭加入元素：`fruit.unshift( "西瓜" );`
 - 尾端加入元素：`fruit.push( "鳳梨" );`
 - 修改陣列元素值：`fruit[1]= "柳丁" ;`

JavaScript語法介紹-if條件句

- If(條件){
 條件成立時，要執行的指令
}
- If(條件){
 條件成立時，要執行的指令
}else{
 條件不成立時，要執行的指令
}
- If(條件1){
 條件1成立時，要執行的指令
}else if(條件2){
 條件1不成立,但條件2成立時，要執行的指令
}else{
 條件1和2均不成立時，要執行的指令
}

JavaScript語法介紹-switch,for,while

- switch(n)
{
 case 1:
 要執行的指令;
 break;
 default:
 要執行的指令;
}
- for(起始值; 條件; 值的增減){
 要執行的指令;
}
- while(條件){
 要執行的指令;
}

JavaScript語法介紹-函式

- **Function 函式名稱(){**
 要執行的指令;
}
- **帶參數的函式**
 - **Function 函式名稱(參數1, 參數2....){**
 要執行的指令;
}
- **呼叫函式:**
 - 函式名稱();
 - 函式名稱(參數1, 參數2....);
- **網頁載入後觸發ready事件 : \$(document).ready(function(){ })**
 可簡寫成 : **\$(function(){ })**

jQuery 語法

- `$(選擇器).方法(參數);`
 - `$( "p" ).html( "hello World" );`
- `$(選擇器).方法(參數1, 參數2);`
 - `$( "img" ).css( "color" ,#0000FF" );`
- `$(選擇器).方法(函式);`
 - `$("button").click(function(){
 $(this).html("按下我");
});`

jQuery 選擇器

- ID選擇器

javascript	jquery	效能
getElementById()	\$("#ID")	javascript快10倍
Document.getElementById("id")	\$("#div1").hide()	

實作練習：利用jquery 程式,將div1的內容變更為
"jquery程式班開班開課中"

jQuery 選擇器

- 標籤選擇器

javascript	jquery	效能
getElementsByTagName()	\$("#element")	javascript快10倍
Document.getElementsByTagName("tagName")	\$("#body").css("color","red")	

實作練習 : 撰寫jquery 程式, 將所有DIV中的P標籤文字顏色改成blue

jQuery 選擇器

- 萬用選擇器

javascript	jquery	效能
<code>getElementsByTagName("*")</code>	<code>\$("*")</code>	javascript快10倍
<code>Document.getElementsByTagName("*")</code>	<code>\$("body *").css("color","red")</code>	

實作練習 : 撰寫有表單與的html，
再透過萬用選擇器將文字顏色改成red

jQuery 選擇器

- 類別選擇器：`$( ". " )`
- 無javascript 原型

實作練習：建立三個按鈕，class定為btn，再撰寫jquery程式將每個按鈕顯示“按鈕”文字

jQuery 選擇器

- 階層選擇器

種類	表示法
元素 子元素	元素以下的所有子元素
父元素>子元素	父元素的子元素
當前元素+相鄰元素	同一個父元素的相鄰元素
當前元素~間接元素	同一個父元素的間接元素

實作練習 : 撰寫有表單與的html，
練習四種階層選擇器的用法

jQuery 選擇器-表單選擇器

選擇器	說明
:input	找出所有input、textarea、select、button
:text	找出所有單行文字標籤
:password	找出所有密碼輸入框
:radio	找出所有單選項目按鈕
:checkbox	找出所有核取項目按鈕
:submit	找出所有傳送按鈕
:image	找出所有影像類型元素
:reset	找出重設按鈕
:button	找出所有按鈕元素
:file	找出所有檔案型元素
:hidden	找出所有隱藏元素與type 為hidden的元素

jQuery 虛擬類別選擇器 - 簡易型

種類	表示法
選擇器:first	第一個的元素
選擇器:last	最後一個元素
選擇器:not(選擇器2)	選擇器2以外的元素
選擇器:even	偶數編號的元素
選擇器:odd	奇數編號的元素
選擇器:eq(數值)	等於(數值)的元素
選擇器:gt(數值)	
選擇器:lt(數值)	
選擇器:header	
選擇器:animated	動畫執行元素
選擇器:root	根元素
選擇器:targets	超連結元素

實作練習：

jQuery 虛擬類別選擇器 - 內容型

種類	表示法
選擇器:contains(字串)	包含字串的元素
選擇器:parent	父元素的子元素
選擇器:empty	同一個父元素的相鄰元素
選擇器1:has(選擇器2)	同一個父元素的間接元素

實作練習：

jQuery 虛擬類別選擇器 - 可視型

種類	表示法
選擇器: visible	挑出顯示中的元素
選擇器: hidden	挑出隱藏的元素

實作練習：

jQuery 虛擬類別選擇器 - 屬性類型

種類	表示法
選擇器[屬性名稱]	包含屬性名稱的元素
選擇器[屬性名稱=屬性值]	屬性名稱 等於 屬性值的元素
選擇器[屬性名稱!=屬性值]	屬性名稱 不等於 屬性值的元素
選擇器[屬性名稱^=屬性值]	屬性名稱以 屬性值開頭 的元素
選擇器[屬性名稱\$=屬性值]	屬性名稱以 屬性值結尾 的元素
選擇器[屬性名稱*=屬性值]	屬性名稱 包含有 屬性值的元素
選擇器[屬性名稱 =屬性值]	屬性名稱 符合CSS規範 的元素
選擇器[屬性名稱~=屬性值]	屬性名稱 含有以空隔隔開 的屬性值元素
選擇器[篩選器1][篩選器2]..	找出 全部條件都符合 的元素

實作練習：

jQuery篩選器-過濾方法

種類	說明
eq(index)	取得第N個元素(從0起算)
hasClass(class)	元素中含有某特定類別
filter(expr)	篩選出符合指定運算式的元素集合
filter(fn)	篩選出符合函數內容回傳值的元素集合
map(callback)	將一組元素轉換成其他陣列
has(expr)	保留包含特定後代的元素
Not(expr)	去除符合指定運算式的元素
Slice(start,[end])	截取出元素群中的部分元素

jQuery篩選器-查詢方法1

類型	種類	說明
向下尋找	children([expr])	尋找出下層子元素
	find(expr)	尋找出所有後代元素
	contents()	尋找下層子內容
向上尋找父元素	parents([expr])	尋找上層父元素
	parent([expr])	尋找上層父元素
	parentUntil([selector])	向上尋找直到有指定的選擇器為止
	offsetParent()	尋找最上層的父元素
	closest(expr,[context])	找到第一個符合條件即停止

jQuery篩選器-查詢方法2

類型	種類	說明
兄弟 向下尋找	next([expr])	尋找後一個相鄰的元素
	nextAll([expr])	往後尋找所有相鄰的元素
	nextUntil([selector])	往後尋找直到有指定的選擇器為止
向上找兄弟	prev([expr])	尋找前一個相鄰的元素
	prevAll([expr])	往前尋找所有相鄰的元素
	prevUntil([selector])	往前尋找直到有指定的選擇器為止
	siblings([expr])	尋找所有兄弟元素