

致理技術學院

資訊網路技術系 實務專題報告

網路 E 化課程輔助 暨管理資訊系統

指導教師：高楊達 老師

學生：劉靜雯 (69434107)

林億綾 (69434111)

陳佑宗 (69434146)

中華民國 97 年 12 月

致理技術學院

資訊網路技術系 實務專題報告

網路E化課程輔助 暨管理資訊系統

學生：劉靜雯 (69434107)

林億綾 (69434111)

陳佑宗 (69434146)

本成果報告書經審查及口試合格特此證明。

指導教師：_____

中華民國 97 年 12 月

專題研究授權書

本授權書所授權之專題研究為網路E化課程輔助暨管理資訊系統

共3人，在致理技術學院資訊網路技術系 97 學年度第1 學期完成資網實務專題。

專題名稱：網路E化課程輔助暨管理資訊系統

同意 不同意

本組同學共3人，皆同意著作財產權之論文全文資料，授予教育部指定送繳之圖書館及本人畢業學校圖書館，為學術研究之目的以各種方法重製，或為上述目的再授權他人以各種方法重製，

不限地域與時間，惟每人以一份為限。

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未勾選，該組同學皆同意視同授權。

指導教師姓名：高楊達 老師

專題學生簽名：

學號：

中華民國 97 年 11 月 27 日

誌 謝

轉眼之間，我們在致理技術學院從懵懵懂懂的青澀少年時期到現在已經要畢業了，想當初剛進入致理技術學院的資訊網路技術系時還不甚清楚將要面對的是哪一方面的領域，伴隨著時間一點一滴的過去，我們了解資訊網路技術是多麼廣大又艱深的知識領域。想到這四年中，我們從一大群不認識的人相偕在這緣份安排之下而互相認識進而結下深刻的友誼，雖然這四年的過程中，陸陸續續有些夥伴相繼離開，但也有一些優秀的生力軍加入，真的很感謝老天給我們緣分讓我們在此相聚，有很多的歡笑也有很多的淚水，但我們都一起攜手彼此相扶相持互相鼓勵安慰陪伴的度過這大學四年。

也很感謝所有指導教師，總是耐心好脾氣的教導我們這些不知死活的小鬼，讓我們從天真無知進而慢慢認識資訊網路技術的這一塊廣大深奧領域，不藏私的將所有知識技術指導傳授於我們。其中最最最感謝的就是我們的指導老師及班導師暨本系的系主任-高主任楊達，常常給予我們彈性的發展空間，適時的給予關心與鼓勵，當我們專題遇到瓶頸或困擾時，總是能精準的針對問題給予專業的建議，使我們的專題系統從雜亂不堪一直修改到頗具專業的感覺，所以我們專題如果有所成就或規模的話，都要感謝高主任不放棄有耐心地一直給予細心指導教誨，有高主任的灌溉，才有我們現在能顯現出來的成果，在此，我們要大聲的說，謝謝您！

最後也要好好謝謝各位同學的支持與鼓勵，更感謝我們所有組員的互相配合，在這一年來大伙們在暑假放棄私人休息時間齊聚一起討論、在過年過節都要心繫專題進度、天冷的季節還要捨棄溫軟

的被窩，為了專題全力以赴互相合作，所以專題有此進度成果我們也要互相謝謝自己，我們做到了！

摘 要

以往傳統的課程管理作業都是紙本作業，教師常常不清楚到底有哪些學生修他的課，如果這學期開的課比較多，登錄成績的相關紙本就需要常常帶在身邊，學生的書面作業也容易造成遺失或是不同課程作業資料混在一起，而學生寄發電子作業容易不同課程或年級混在一起。所以設計出這套簡易系統，可以將學生資料整合，可以上網直接登錄學生的成績，電子作業有完善的分類規劃，加上也可以達到節能減碳，資訊 e 化的目的。

我們利用 JSP 來達到後端程式執行與資料庫溝通部份網頁的呈現，JavaScript 讓畫面更為方便，使用也更為方便，用 HTML 將網頁內容呈現出來，再使用 Dreamweaver 來達到網頁版面美觀設計等資源打造出適性化專屬教師的課程輔助管理系統。

關鍵詞：JSP、JavaScrip、HTML、Dreamweaver、資料庫、課程輔助系統、課程管理系統

目 錄

授權書	i
誌謝	ii
摘要	iv
目錄	v
圖目錄	vii
表目錄	xi
第一章 緒論	1
第一節 管理資訊系統之發展演進	1
第二節 研究動機與目的	2
第三節 研究範圍	4
第四節 研究架構	5
第二章 理論與技術探討	6
第一節 HTML	6
第二節 Dreamweaver	9
第三節 資料庫系統—SQL 2005	11
第四節 JavaScript 與 JSP	14
第五節 實體關聯模型 (ER Model)	19
第六節 正規化	30
第三章 網路 e 化課程輔助暨管理資訊系統	42
第一節 系統架構	42
第二節 啟用 SQL Server 2005 資料庫	44
第三節 本系統資料庫的 ERM	59

第四節 系統資料庫	60
第五節 啟用 JSP 環境	64
第六節 系統技術與分析	71
第四章 系統呈現	76
第一節 起始畫面及註冊	76
第二節 學生端使用操作	75
第三節 教師端使用操作	76
第五章 結論	100
參考文獻	101

圖 目 錄

圖 1-1、研究概念架構圖	5
圖 2-1、資料庫系統 (Database System)	12
圖 2-2、實體與實例 (instance) 之間的關係	20
圖 2-3、實體子型態與超型態	21
圖 2-4、實體與屬性的圖形表示	22
圖 2-5、各種屬性的圖形表示法	24
圖 2-6、各種屬性的圖形表示法	25
圖 2-7、三種不同的實體關係 (m 表示多)	27
圖 2-8、實體關聯圖使用的圖形符號	29
圖 2-9、關聯是資料表圖	38
圖 2-10、與 Primary Key 完全無關的	39
圖 2-11、相依關係	40
圖 2-12、表 2-10、2-11 資料表關聯圖	41
圖 3-1、系統架構	43
圖 3-2、微軟更新安裝畫面	44
圖 3-3、SQL Server 2005 資料庫安裝畫面 1	45

圖 3-4、SQL Server 2005 資料庫安裝畫面 2	46
圖 3-5、SQL Server 2005 資料庫安裝畫面 3	47
圖 3-6、SQL Server 2005 資料庫安裝畫面 4	48
圖 3-7、SQL Server 2005 資料庫安裝畫面 5	49
圖 3-8、SQL Server 2005 資料庫安裝畫面 6	49
圖 3-9、SQL Server 2005 資料庫安裝畫面 7	50
圖 3-10、SQL Server 2005 資料庫安裝畫面 8.....	51
圖 3-11、SQL Server 2005 資料庫安裝畫面 9.....	52
圖 3-12、SQL Server 2005 資料庫安裝畫面 10	53
圖 3-13、SQL Server 2005 資料庫安裝畫面 11	53
圖 3-14、SQL Server 2005 資料庫安裝畫面 12	54
圖 3-15、SQL Server 2005 資料庫安裝畫面 13	55
圖 3-16、SQL Server 2005 資料庫安裝畫面 14	56
圖 3-17、SQL Server 2005 資料庫安裝畫面 15	57
圖 3-18、SQL Server 2005 資料庫安裝畫面 16	58
圖 3-19、資料庫分析圖	59
圖 3-20、JDK 安裝 1.....	64
圖 3-21、JDK 安裝 2.....	65

圖 3-22、JDK 安裝 3	66
圖 3-23、JDK 安裝 4	67
圖 3-24、JDK 安裝 5	67
圖 3-25、TOMCAT 安裝畫面 1	68
圖 3-26、TOMCAT 安裝畫面 2	69
圖 3-27、TOMCAT 安裝畫面 3	70
圖 3-28、TOMCAT 安裝畫面 4	70
圖 4-1、管理系統首頁	76
圖 4-2、最新訊息存放地點	77
圖 4-3、最新訊息修改檔	77
圖 4-4、學生註冊畫面	78
圖 4-5、學生註冊輸入資料畫面	79
圖 4-6、學生登入畫面	80
圖 4-7、學生管理首頁	81
圖 4-8、學生選課畫面	82
圖 4-9、學生課程查詢畫面	83
圖 4-10、學生修改個人資料	84
圖 4-11、學生成績查詢	85

圖 4-12、學生查詢成績期中考課程選單	86
圖 4-13、學生查詢課程期中考成績	87
圖 4-14、學生上傳作業課程選單	88
圖 4-15、學生上傳作業	89
圖 4-16、學生上傳作業	89
圖 4-17、教師登入畫面	90
圖 4-18、教師登入後的畫面	91
圖 4-19、教師開課畫面	92
圖 4-20、老師端課程查詢畫面	93
圖 4-21、審核資料畫面	94
圖 4-22、登入成績 1.....	95
圖 4-23、登入成績 2.....	96
圖 4-24、登入成績 3.....	97
圖 4-25、作業存放點 1.....	98
圖 4-26、作業存放點 2.....	99

表 目 錄

表 2-1、員工資料表	32
表 2-2、非正規化資料表	33
表 2-3、非正規化資料表	34
表 2-4、正規化資料表	34
表 2-5、部分相依	36
表 2-6、出貨記錄表	37
表 2-7、書籍資料表	37
表 2-8、訂單資料表	38
表 2-9、訂單資料表	39
表 2-10、訂單資料表	40
表 2-11、員工資料表	40
表 2-12、對照表	40
表 3-1、系統資料表-老師開課資料表	60
表 3-2、系統資料表-學生資料資料表	60
表 3-3、系統資料表-學生選課資料表	60
表 3-4、系統資料表-學生平時成績資料表	60

表 3-5、系統資料表-學生期中/末成績資料表	61
表 3-6、資料庫欄位-老師開課資料表	61
表 3-7、資料庫欄位-學生資料資料表	62
表 3-8、資料庫欄位-學生選課資料表	62
表 3-9、資料庫欄位-學生平時成績資料表	63
表 3-10、資料庫欄位-學生期中/末成績資料表	63

第一章 緒論

第一節 管理資訊系統之發展演進

現代生活隨著科技技術日新月異的快速發展下，資料管理方面已經從過去的紙上作業漸漸演進成網際網路遠端電子化管理所取而代之。

透過網際網路利用電子化資訊管理存取更新紀錄資料不僅變得簡單容易，而且資料管理的管道也變的更加多元化及方便。例如：客戶向業務人員下訂單買書，以往要先找出客戶的資料，再填寫一份訂單資料，而客戶又不單單只有一、兩個，必須在一堆客戶資料紙群中搜尋。造成人工搜尋耗時或沒辦法馬上查詢到該客戶的歷史資料甚致容易發生遺失、紙張保存不易…等問題，造成許多後續的困擾。

之後雖出現了電子資料管理，輕易解決了以上此類問題。只要利用電腦建檔，不只搜尋容易，新增建檔也方便許多，也將更節省時間與人力上的消耗浪費。但是在現在這個充斥著網際網路的 E 世代，光是只有電子資料管理是無法滿足管理者與使用者的需求，必須可以透過網際網路遠端管理的電子資料系統才能跟的上現代 E 世代的腳步。

所以現在利用 SQL 資料庫系統與網頁做連結，管理者可以輕易在遠端控管資料，而使用者也可以方便利用網際網路下單，也不需要再”親自”跑那麼一趟，只要利用網際網路就可以即時更新即時查詢即時管理，節省了人力與時間上的花費，使用性上也大大提升方便度，更提高了管理者的管理效率呢！

第二節 研究動機與目的

目前國內已經有很多架設在網際網路上的資訊教學網站，也已經漸漸受到國人的接受與肯定，成為虛擬教室的新寵，所以全球資訊網已成為不可忽視的新教學輔助媒體。

教學都可以利用現在 E 世代網際網路的腳步架設出虛擬輔助教學教室，那麼代表網際網路在學校的輔助使用是可以有更多變化的，例如可以利用在學生選課的管理，開放學生們可以利用網際網路在家即可連線上網來做課程之查詢與選課之管理。但目前針對教師們的課程學生管理卻尚未普及，教師授課需要面對許多班級及許多學生，卻只靠著傳統紙上名單來聯繫管理，本組認為似乎可以利用網際網路結合資料庫電子化來輔助教師們方便管理，讓教師也能向企業一樣隨時跟著科技世代的腳步前進。

目前一般傳統教師在管理學生流程有以下困擾：

- ◎ 學生課程名單：不同課程選課的學生名單各有一份紙張名單，教師總是要另外規劃一套管理方式，相當不方便。
- ◎ 教學進度：必須在書上、紙上塗塗寫寫著名，眼花撩亂，也容易搞混。
- ◎ 繳交作業：除了繳交書面報告外，電子檔總是靠著一堆磁碟片、隨身碟或是電子信箱傳送，容易造成遺失或是信件被擋，甚致收到無心的病毒傳播。
- ◎ 教師與學生之間橋樑：只能靠課堂中或是下課休息時間溝通，時間過短，且有時面對面反而讓學生更難言之隱。
- ◎ 成績：必須在紙上塗塗寫寫，還要小心保管；學生想知道成績也

必須等教師公佈或是等學校公佈，等待時間過長。

◎ 教學：就只有上課時間，時間過於緊迫。

◎ 測驗：傳統紙筆測驗，紙張消耗過多，現在是講求節能減碳的時代，似乎顯得有些不環保。

因此，本組希望可以研發出一套版面與架構都頗為簡單，專屬於教師們能很容易也很輕鬆的管理學生課程資訊管理系統，輔助教師也可以輕鬆利用網際網路管理並輕易解決以上之問題。希望讓教師們可以輕易的利用網際網路遠端就能做到這些事情，提高教學效率，以教師為出發點打造專屬於教師的網路E化結合資料庫之輔助教師課程管理資訊系統。

第三節 研究範圍

在現今電子資訊發達的時代中，市面上已經有很多套電子資訊管理系統，例如：便利商店商品管理系統、書店租賃管理系統、醫院病例管理系統、各商家會員管理系統、公務人員出勤管理系統.....等等，都是針對客戶需求而花錢外聘請工程師外包工程所製作。因學校經費稍嫌拮据，並無法針對每位老師製作軟體，而外包工程不僅費用頗高且整體上太過於華麗、架構上也多少埋伏許多商業內容，容易讓老師們感受不到專屬用於課程的實用性。

因此本組將以老師們的課程管理需求為目的去探討研究，希望能研究出一套方便簡單的課程管理系統提供給每一位元需要的授課老師輕鬆使用。

本組欲利用 JSP、Java Script、HTML 等程式作為主要的研究範圍撰寫出可以輔助教師管理系統之服務項目，當然網頁與程式、資料之間的結合也是本組所研究的範圍之一，例如：SQL 2005 資料庫系統、DreamWeaver 網頁設計軟體等等方面的技術與理論都是本組所要研究探討的範圍。

第四節 研究架構

本組基本研究概念架構如下：

- (一) 教師不需要另外安裝特殊軟體也不需要帶著系統程式到處跑，就可以透過網際網路登入該輔助系統利用網際網路連線到資料庫做資料之管理。
- (二) 學生也可以在世界各個角落隨時利用網際網路連線繳交作業線上測驗或是成績查詢等等。

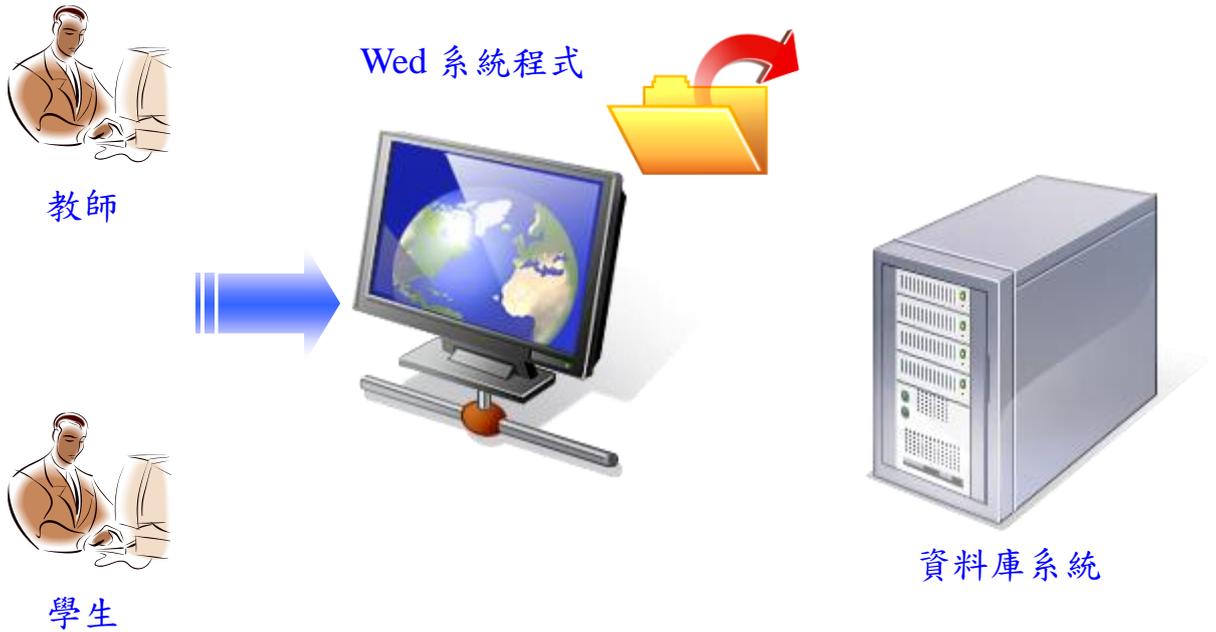


圖 1-1 研究概念架構圖

第二章 理論與技術探討

第一節 HTML

在 1960 年代時，IBM 開發了一種稱為 GML 的電腦語言，是用來描述內部檔與排版的用途，之後 IBM 又將 GML 改良成為 SGML，在那時 SGML 也廣泛地被當時的工業界所採用，之後在 1986 年 SGML 被列為國際標準規範，成為所謂的電腦檔的標準格式化語言。

後來在 1989 年代時，兩位 CERN 的研究人員 Tim Berners-Lee 與 Andres Berglund 又將 SGML 再度簡化成為一種以標示文字為基礎只要透過 HTTP 網路通訊協定，就能夠在世界各地透過 WWW 網際網路上利用電腦語言的架構做跨平臺之通訊交流，這就是所謂的 HTML [1]。

HTML 是 Hyper Text Markup Language 的縮寫，按照英文字面上的意思解讀為「超文字標示語言」，顧名思義就是具備有超文字 (HyperText)、超連結 (HyperLink)、超媒體 (HyperMedia) 等等之特性，是屬於標示語言。它讓資料利用各式各樣的控制碼來描述，然後再將這些資料依照控制碼來做轉換，最後形成一份完整的檔。這是它跟一般的文書處理器最大不同的地方。

HMTL 一開始只是用來描述靜態的文字或圖像，但是經過了許多努力的研發改良，今天我們所看到的 HTML 已經能做出許多的動態效果。

而 HTML 最重要的一個標籤就是表單 (Forms)。一般來說，HTML 的標籤，就是為了用來呈現描述網頁內容的電子資訊所用之電腦語言，接收伺服器端傳過來的超文字標示語言資料解讀的結果顯示在您的電腦螢幕上，以方便使用者在客戶端的瀏覽。而表單不僅是只是呈現電子資

訊的功能，它最重要的是還包含了多種表單元素（Form Elements），就可以讓使用者利用網頁進行電子資料填寫或選取的功能，並且將此電子資料傳送到伺服器端，讓管理者可以進行必要的處理。所以，表單可以說是客戶端對伺服器端進行電子資訊交流的第一個重要橋樑。而表單將使用者所輸入之電子資料傳送到伺服器後，伺服器端將會有相對的程式來對電子資料進行一定的處理，例如將資料送入資料庫或是進行統計分析等等。

HTML 中的標籤主要是由「<」和「>」這兩個符號所組成的，例如說<title>就是 HTML 中的一個標籤，它是由英文字「title」和「<」、「>」符號所組成。

依照標籤的位置的不同，我們可以把 HTML 中的符號分成兩種，一種為「起始標籤」，另一種為「結束標籤」。起始標籤和結束標籤幾乎完全一樣的，只不過結束標籤比起始標籤再多了一個「/」。

為了網頁中有更豐富的變化與效果，HTML 中的標籤還有「屬性」與「值」。例如我們在網頁使用<a>這個標籤來產生超連結，那超連結要有連結才行，所以我們要在<a>標籤中加上「href」這個屬性來產生連結。

要編寫 HTML 的檔案就必須要遵循 HTML 的語法與規則。一個完整的 HTML，必須由標題、段落、列表、表格、文字及嵌入的各種物件所組成。這些邏輯上統一的物件我們稱之為元素，而 HTML 使用標記來分割並描述這些元素，其實整個 HTML 檔就是由這些元素與標記所組織而成的。

HTML 原始碼分為 3 部份，其中：<HTML>……</HTML>就是宣告瀏覽器 HTML 檔案的開始和結束，所以一個 HTML 檔總是以<HTML>開始，以</HTML>結束的。HTML 檔中所有的內容都應該在<HEAD>和

<BODY>這兩個標記之間；<HEAD>……</HEAD>是 HTML 檔案的標題標記，可以放置頁面的標題以及檔案資訊等之內容，通常在這標籤中間的內容統稱做 HTML 標題；而<BODY>……</BODY>則是 HTML 檔的主體標記，絕大多數 HTML 內容都放置在這個區域裡面，通常位於</HEAD>標記之後</HTML>標記之前的位置〔2〕。

在傳統的網頁中使用者選擇好觀看的頁面，系統隨即將此頁面的所有內容簡單的顯示在電腦螢幕上，這種方式就是所謂的靜態式網頁，主要就是利用超連結的功能來達到此目的，雖然有些網頁會加上一些 Gif、Flash 或是 JavaScript 的動態效果，但是這些也只能算是「視覺上」的動態而已。

而真正的動態網頁不只是只有視覺上的動態而已，是還要做到將網頁與網站伺服器的資料庫建立連結，讓使用者端與網路伺服器端具有即時互動的效果，並且可以即時更新網頁上的內容，這才能所謂真正的動態網頁。

除了互動性的功能之外，動態網頁的維護與管理的便利性也是靜態網頁所不能比的。例如網站網頁資料需要新增或刪除的時候，就不需要像傳統的靜態網頁那樣再大費周章的重新製作另一個新的頁面；而是只要更新資料庫中的內容即可，這樣不僅可以大大的節省網站維護的時間與成本，還可以在使用者端動態展示資料庫中的內容，甚至可以讓使用者端輸入的資料寫入伺服器端的資料庫中，以達到雙向互動的效果。

第二節 Dreamweaver

在早期，說到要做網頁，一般人第一個想到的，通常就是 FrontPage，然而，隨著電子商務的發展，能夠儲存龐大資料的資料庫網頁，已經漸漸成為一種主流。其中，Macromedia 公司推出的 Dreamweaver，即是為了製作資料庫網頁，而量身打造的一套軟體。

Dreamweaver 的誕生就是為了幫助這些不懂的寫程式的使用者，能以視覺化的方式製作出專業的動態網頁。Dreamweaver 與 Flash、Fireworks 等軟體一樣，都是 Macromedia 公司的重量級產品，將這幾套軟體整合搭配應用後，可以說是網頁設計的完全解決方案。

而 Dreamweaver 與 FrontPage 這兩套的主要的差別在於，FrontPage 主要是用來製作「靜態式」網頁，而 Dreamweaver 則是可以製作靈活的「動態式」網頁。雖然說 FrontPage 也是可以與資料庫做結合，但是 FrontPage 的功能畢竟沒有 Dreamweaver 來的那麼的完整〔3〕。

Dreamweaver 的操作環境是採用圖形的介面處理畫面，並且只要搭配 HTML 的原始碼就可以自動產生，此軟體同時可以即時交互編輯修改，有編輯文字、插入分隔線、表格製作、超連結、圖形編輯、圖層應用等的功能，便可以輕輕鬆鬆的產生一般與動態甚至包含程式組等多樣式的網頁型態。而且 Dreamweaver 還可以插入 Flash 的.swf 動畫文件和 Firework 設計的 HTML 文件，讓網頁更加活潑生動。

Dreamweaver 具有「設計互動式網頁」的功能。我們可以利用 Dreamweaver 的「圖層」(Layer) 和「時間軸」(Time Line) 功能，來讓圖片或文字隨意地在網頁中自由移動。移動的方式可以是繞著你所設定的路徑 (path) 移動，也是可以左右或直線的移動。

另外，你可以設計讓文字或圖片配合鼠標的點選有更多的變化。就是我們網路上看到網頁當遊標經過文字或圖片，文字或圖片會產生變化的動態效果。

Dreamweaver 大致具有下面幾項特色：

- (一) 直覺式的操作介面：Dreamweaver 可以直覺式的配置網頁、管理網頁、發佈網頁，而且可以快速開發 Web 應用程式，修改程式的內容。
- (二) 可隨時顯示或隱藏的工具列及功能面板：Dreamweaver 的工具列及面板都是定置式，而且可以視需要隨時隱藏或是開啟，這樣編輯的時候就不需要將工具列或面板搬來搬去，增加編輯的方便性。
- (三) 可製作出跨平臺的應用程式：以前要製作動態資料庫的網頁，要再不同的伺服器平臺就得使用該平臺所特定的開發工具來製作。而 Dreamweaver 可以在相同的編輯環境下，製作出 ASP、ASP.NET、JSP、PHP 及 CFML 等伺服器模組的網頁。
- (四) 視覺化的資料庫連結設定：建立資料集後，只要用拖移滑鼠指標的方式，就可以將網頁上的文字或圖形與資料庫建立連結。
- (五) 即時的資料庫連結預覽：建立連結後的網頁，不需使用瀏覽器或上傳到伺服器上即可預覽結果，方便又有效率。

第三節 資料庫系統—SQL 2005

SQL server 是一關聯式資料庫，除了支援傳統關聯式資料庫元件（如資料庫、資料表）和特性（如資料表的 join）外，另外也支援關聯式資料庫常用的物件（如預存程式、檢視）、標準查詢語言—SQL(Structured Query Language)、資料庫複寫的功能等。

SQL server 最早是在 1989-1990 年出現，適用於 OS/2，而 Windows NT 上的 SQL server 初版為 SQL server 4.2 在 1993 年出現稱為桌上型資料庫，是一個低功能性的桌上型資料庫，能儲存和處理一個小部門的資料需求，且能與 Windows 整合。在 1995 年一個核心資料庫引擎的主要改版 SQL server 6.5—一個小型的商業資料庫，也是 SQL server 第一個「重大」的版本。1998 年 SQL server 又將核心資料庫引擎作另一次重大改版 SQL server 7.0—網路資料庫，是一個關鍵性的版本。2000 年 SQL server 2000 大幅改善效能、擴充還有可靠度讓 SQL server 變成企業級資料庫，成為市場主要成員之一。一直到現在 SQL server 2005 已經帶來一系列的新特徵、圖形使用者介面（GUIs）與管理工具〔4〕。

在以前的操作方面，使用者是利用終端機或終端機模擬程式來使用資料庫系統，其操作畫面設計是文字導向。現今的應用架構則大部分是使用個人電腦和視窗作業系統作為前端的平臺，所設計出來的操作畫面都是視窗化的畫面。

前端開發工具用來設計輸入和查詢畫面，使用者透過這個畫面輸入資料，再由前端程式透過網路傳給後端的資料庫引擎將資料儲存在資料庫。當使用者要查詢資料時，前端程式將查詢命令傳給後端的資料庫執行、前端程式則等待接收資料結果，然後再將結果顯示在畫面上。

SQL server 2005 資料庫內含.NET 框架共用語言執行環境 (CLR)，所以可在資料庫中用 Visual Basic 2005 和 C#設計組件程式。對於 SQL server 資料庫程式設計師而言，這是一個不錯的結果，因為以前他們只能受限於 SQL 和 T-SQL，而現在這樣的發展已經可以使用更彈性的方式來架構應用系統。

透過成熟的 HTML 資料類型與相關資料類型的連結，可完善地支援 HTML。可直接插入一份 HTML 檔到資料庫中，並讓後者驗證、萃取出部份的文件內容；這意謂著我們能以關聯性資料方式處理半結構性的資料、儲存在相同的地方，並以同樣的方式對待他們。另外，伺服器也支援 HTML 查詢及語言定義的標準〔5〕。

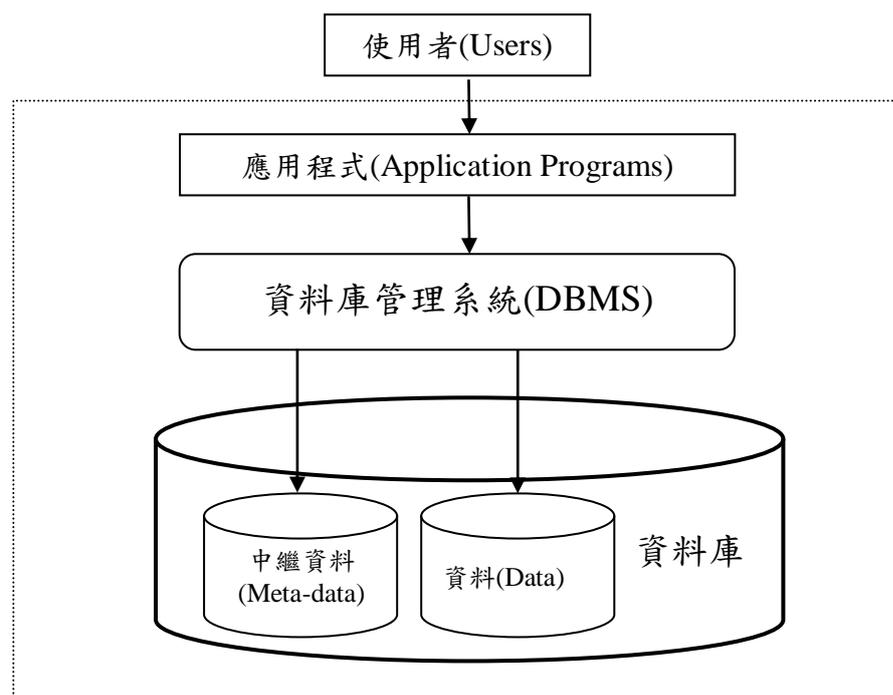


圖 2-1、資料庫系統 (Database System)

資料庫管理系統是一些程式模組如圖 2-1 所示，分別負責定義、建立和維護資料庫，並且控制資料庫的資料存取和管理使用者權限。

資料是儲存在電腦儲存裝置的磁碟機內，使用者只需執行應用程式，下達查詢條件指令，就可以存取查詢條件相關的資料〔6〕。

第四節 JavaScript 與 JSP

JavaScript 是一種電腦語言，而且是直譯式的電腦語言。所謂的直譯式電腦語言就是解析器一行一行地解譯程式碼，將程式碼轉換成為電腦可以瞭解的機器碼。JavaScript 是目前 script 語言中最受歡迎的。原因就是因為所有瀏覽器都會支援 JavaScript，不管您解析器（就是您的網頁瀏覽器）的是使用 Internet Explorer 或是 Netscape Neavigator。甚致在不同的電腦平臺。如 Windows、Unix、Linux 等，JavaScript 都是標準的 script 語言。

JavaScript 它只能夠存在於網頁中的<script>標籤內，或是儲存在副檔名為*.js 的檔案中。JavaScript 沒有資料型態的檢驗，不管您使用的資料型態是整數、浮點數、還是字串，我們通通使用 var 關鍵字來宣告，或者是根本就不需要宣告資料型態。甚致我們可以將原本是數字的變數，在途中改成為字串。當我們在瀏覽器中載入一個網頁時，瀏覽器除了會解譯網頁中的 HTML 程式碼以外，它還會解譯網頁中的 script 程式碼。HTML 程式碼是靜態顯示標籤與文字，而 script 程式碼則能讓您的網頁做出與使用者互動的功能。例如存取資料庫的紀錄、檢查使用者的帳號與密碼輸入值的正確與否、以及設計資料的表格等等。

由於 JavaScript 在不同瀏覽器上有不同的格式，所以 Microsoft，Netscape，以及其他公司聯合起來制定了統一的 JavaScript 規格，並且稱為 ECMA Script。（ECMA 是 European Compute Manufacturers Association 的縮寫，是總部位於瑞士日內瓦的國際組織。）〔1〕

ECMA Script 標準只涵蓋 JavaScript 的核心特性，即數字，字串，與布林值的資料型態，Date、Array 與 Math 等自然物件，if、else 條件式，

以及 for 與 while 迴圈等。ECMA Script 並沒有制定用戶端物件，例如 window、document、form 等物件的規格。所以雖然 ECMA Script 讓 Script 與 Jscript 的核心語言相容，但是用戶端物件在不同的瀏覽器上仍然有不相容的問題。

E 世代在現在的資訊科技產業快速發展下，隨著科技進步，程式語言也演變的越來越複雜。從一開始的非結構化程式語言到結構化程式語言，再發展到物件導向程式語言。Java Server Page (JSP) 是一種用來產生動態網頁內容的規格，以 Java 的技術為主的跨平臺 web 所開發的語言與跨網頁伺服器支援的特性。這樣的技術在網路上的應用相當廣泛，舉凡網路購物、網路銀行和線上訂票等等，皆是其可以發揮之處。傳統的網頁建構技術，如 Common Gate way Interface (CGI) 等，必須以程式產生整個網頁，包括靜態和動態內容的部份。而 JSP 可以嵌入在靜態的 HTML 網頁中，僅用來產生動態的內容，使網站的開發更容易。JSP 也可以將網路應用程式的前端介面和後端程式邏輯隔開來，使用網路易於維護。它是使用類似 HTML 的標籤以及 Java 程式代碼段所構成的，可以用來架構網站，而不是常用到的 VBScript。JSP 使用 Java 的程式代碼編譯成 Java 的 byte code (可以需要時才編譯，也可以先編譯好)。

JSP 技術是 J2EE 的一個關鍵技術，它在更高一級的層次上抽象 Servlet，它可以讓常規靜態 HTML 與動態產生內容結合。現在有許多商業應用伺服器都支持 JSP 技術，如 BEA WebLogic、IBM WebSphere、JRun 等。如果你有 JSP 的 Web 伺服器，你可以把它放到任何靜態 HTML 檔可以放置的位置，不用編譯、也不用進行 ClassPath 的設置，就可以像一般網頁一樣使用它，這個伺服器會自動幫你做好其他的工作。

JSP 的編譯器指示是針對 JSP 引擎，它不會直接產生任何看得見的

輸出；相反的，它是在告訴引擎如何處理其他的 JSP 網頁。它永遠包含在（<%……………%>）標籤裡，兩個主要的指引是 page 與 include。

在 JSP 網頁的最上面可以找到 page 的編譯器指示，如：<% page errorPage="errorPage.jsp"%> 發生問題時，會把訊息傳 <% page errorPage="errorPage.jsp" %>。Include 編譯器指示讓你將你的內容分成可管理的組件，就像那些有表頭或註腳的網頁，如：<%@ include file="filename.jsp" %>。

我們已知靜態 HTML 如何存取，只要輸入（Uniform Resource Locator）如 <http://dormouse.instantjsp.com/SimplePage.html>。瀏覽器透過 HTTP（Hypertext Transport Protocol）傳輸協定，針對 SimplePage.html 的內容，送出一個請求到網域 instantjsp.com 中的主機 dormouse 上執行的網頁伺服器，網頁伺服器又得以 HTML 標籤解釋展現方式的檔案後，傳送給瀏覽器。

對 JSP 的請求很類似，只要指定延伸檔名為.jsp 而非.html。上述的 URL 就變成：<http://dormouse.instantjsp.com/SimplePage.jsp>。將請求傳送到支援 JSP 的網頁伺服器或網頁應用程式伺服器，辨識出.jsp 的延伸檔案，轉送請求到 JSP container，呼叫編譯過的 JSP 中的_jspService（）方法。_jspService（）方法傳送回應到 JSP container,再傳送回客戶端。不一定是由瀏覽器提出 JSP 的請求，大部份的情況是由瀏覽器提出，但亦可能由 servlet 或其他的 JSP 程式提出。〔7〕

JSP 所提供最重要的優點：將程式邏輯與視覺及感覺效果分開。這是早年困擾網站的主要問題，影響的範圍也是很大。由網頁設計師所製作出來的網頁，雖然很吸引人，但缺乏功能；另一方面，程式設計師所製作出的網頁，擁有功能卻乏吸引人的外觀。由程式設計師及美工設計師合

作出來的網頁解決了這個問題，但在維護上又出現問題。美工設計師有時會發現改變程式邏輯會影響視覺及感覺效果。使用 JSP，兩者就能夠真正地獨立工作。

另外一種重要的優點，就如 Java 簡短的口號中所號稱-Write Once,Run Anywhere™。將 JSP 程式編譯為 byte code 使其跨越平臺，可以在任何平臺上開發，並在任何伺服器上部署。

JSP 程式也將嵌入可重複使用的元件變得容易。JavaBeans 及 Enterprise JavaBeans 等特殊功能元件，可以在開發後，供任何 JSP 程式使用。想像一下，你被指派一個開發網頁應用程式的工作，主要目的是為舊有系統提供現代化的使用介面。你可能花費相當多的時間評估要使用市面上的 Enterprise JavaBeans 來讓 JSP 與 CICS 溝通，或執行 3270 終端機模擬，或者決定自行開發。如果你的下一個指派工作是為另一個舊有系統做同樣的事情，你只要在 JSP 中，包含必要指令，使用先前購買或開發 Enterprise JavaBeans 就行了。除了可重複使用的元件外，JSP 程式可靜態地或動態包含其他 JSP 程式〔8〕。

Java2 Enterprise Edition (J2EE) 已將 JSP 技術整合進來，將其帶進企業運算的領域。利用 JSP 程式來建構網站，可以建立一個前端元件，結合 J2EE 所提供強大的 N-層 (N-tier) 應用程式架構。

另一個優點是自訂標籤庫所帶來的延展性。除了標準的語言必須由 JSP container 實行的動作 (action) 種類以外，JSP 技術規格第一版提供一個 taglib 指令，以容納新的 action 種類。此法減少 JSP container 開發人員的力氣，並縮短學習 JSP 的時間。技術規格更進一步提到，未來的技術規格版本將包含一個標籤延伸架構，用來說明所有非標準 action 語法。JSP 技術規格 1.1 版提供更多細節，說明如何實作 taglib 指令。所有的 JSP

container 保證都可支援 taglib，它可以動態地載入，不必依靠提供 JSP container 廠商的支援。標籤庫中定義的 JSP 元素可以在設計時使用的網頁編輯工具中看到，也可以由 JSP 授權的工具來使用。

第五節 實體關聯模型 (ER Model)

實體關聯模型 (Entity Relationship Model, 簡稱 ER Model 或 ERM), 是在於 1975 年由美籍的華人陳品山 (Peter Chen) 先生所開發提出最普遍的概念資料模型, 您需要用很多的方式來整理組織需求的資料以及如何將它結構化。通常這類的檔要經過一些人的審查, 這些人包括資料模型的建立者、終端使用者、實際負責建置資料的人。是一項能增強資料塑模 (data modeling) 的清晰與明瞭性的工具。在分析與設計階段要進行資料表的設計是用來協助資料管理師描述出實體 (entity) 以及實體和實體之間的各種關係 (relationship)。而透過 ERM 資料模型所描繪出的 ERD 資料關聯圖便是對應到資料庫架構三層次中概念層的概念模式 (conceptual model) [9]。

ERM 採用圖形方式來描述資料, 對簡單的系統與複雜系統的高層透視, 利用 ERD 的方格、線條與其他符號會比用文字描述的方式更容易瞭解, 對公司系統而言它是一個很有價值的資料模型工具。然而 ERM 並非都適用於每一種資料模型, 知道何時不用 ERM 以及該如何使用它是很重要的。

實體關聯模型組成元件有關之元素及其相關之性質包括: 實體、屬性、關係。實體關聯模型是目前資料庫系統分析和設計時最常使用的方法, 可以將商業領域的公司或組織的資料以邏輯方式呈現。

實體有時稱實體類別 (Entity Class) 或簡稱實體關聯式資料庫的最基本抽象概念, 用以描述實際世界中的物件事情與實體。一些具有共同性質 (Properties) 或特徵 (Characteristics) 之實體案例 (Entity instance) 或稱案例 (Instance) 的集合。每個實體類型有一個名稱為其辨別物

(Identifier) 且常以矩形表示，實體的名稱標示於矩形內。實體就是在真實世界中我們有興趣的人、事、物。由於，在真實世界中有太多的人、事、物是我們所關心的，因此在決定實體之前，我們必須界定範圍。同時必須以系統使用者的角度，去思考哪些人、事、物會是他們所關心的。舉例來說，當我們幫學校建立一套紀錄學生資料庫時，我們必需界定這套系統將來會是供誰使用？如果是供學務處使用，由於學務處主掌學生各類事務，使用者所關心的是學生的基本資料、緊急連絡人姓名以及方式。因此，這時的實體便是學生及其緊急連絡人。如果資料庫未來供課務組使用，課務組長記錄學生的選課資料、各課程班級 (class) 的資料 (包括上課時間、地點以及上課人數)、學生的修課成績。這時的實體便是學生、課程班級以及成績三項。因此，實體將因為資料庫未來的使用範圍而有不同的差異。每個實體都會包含一個或一個以上的實體 (instance)，要紀錄每個實際實例可以透過關聯式模型中的表格 (table) 來紀錄 [10]。

SNO	SNAME	ADDRESS	TEL
694341	陳小宗	臺北縣板橋致理路 1 號	0252210
694342	劉小雯	臺北縣板橋致理路 2 號	0252277
...			
694343	林小綾	臺北縣板橋致理路 3 號	0252220

圖 2-2 實體與實例 (instance) 之間的關係

如圖 2-2 所示，您可以發現一個『學生』實體，包含多個實體，右邊每一筆記錄都是一組存在於『學生』實體的實例。此外，實體又可細分為強實體 (strong entity) 以及弱實體 (weak entity)，以及實體子型態和

超型態分別說明如下：

(一) 強實體：是指不需要依附其他實體而存在的實體。例如：即使學生無法提供緊急連絡人資料，學生資料仍需要被紀錄。因此，就學生以及緊急連絡人而言，學生便是強實體。

(二) 弱實體：是指需要依附其他實體而存在的實體。例如：如果學生轉學，則該學生的緊急連絡人資料便不需要被紀錄。因此，就學生以及緊急連絡人而言，緊急連絡人便是弱實體。

強實體以長方形表示，而弱實體是以雙邊框的長方形表示，又以圖書館所要紀錄的書籍資料為例，有些書本有套件，因為，套件依附於書本，若該書本已經因為年代過久而被圖書館理員決議淘汰，則其套件便不需要保留。在此例中『書本』便是強實體，而『套件』便是弱實體。

(三) 實體型態 (entity subtype) 與超型態 (entity supertype)：實體子型態與超型態是相對的，如圖 2-3 為例：

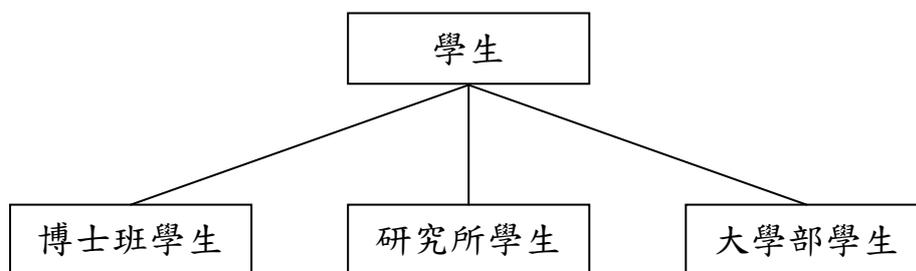


圖 2-3 實體子型態與超型態

在學校中，學生這個實體還可以再細分為博士班學生、研究所學生以及大學部學生如圖 2-3 所示。因此，我們稱博士班學生、研究所學生以及大學部學生為學生的子型態 (subtype)；而學生則是他們的超型態 (supertype)。實體子型態與超型態的分類方法並不包含在 Chen 所提出

的 ER 資料模型中，而是由 Teorey、Ynag 和 Fry 於 1986 年所補充的。

一個實體會需要的屬性 (Attributes) 對於一些屬性有它專屬屬性值 (Attribute values) 實體關聯模型相信實體與關聯性是真實世界最自然的資料塑模方式。可以使用實體和關聯性來描述真實世界的資料。換個角度來說，實體關聯模型是將真實世界的資料塑模成邏輯關聯資料。每個實體類型都具有一些屬性，每個屬性是實體類型的一個性質或特徵。

對於每個實體，使用都會進一步想紀錄或瞭解的資訊，而這些資料便成為該實體的屬性 (attribute)。由於，使用都對於每個實體所要紀錄的資訊不只一種，因此，每個實體常擁有多個屬性。以學生實體而言，讓課務組人員有興趣的是學生的學號、姓名以及班級。因此，『學生』實體所包含的屬性便包含：學號、姓名以及班級。

因為在 ERD 資料模型是一最注重圖形表示的方法，因此，對於實體與屬性各有指定的表示方法。實體以長方形表示，一個屬性有一名稱以茲辨別且常以橢圓形表示，並將屬性名稱標示於橢圓形中且以一條直線與其實體類型相連接，如圖 2-4 所示。

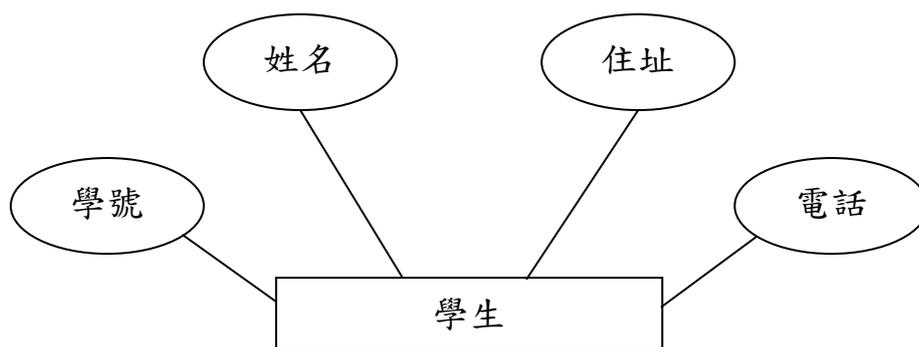


圖 2-4 實體與屬性的圖形表示

我們可以知道有一個『學生』的實體，其相關的屬性分別是『學號』、

『姓名』、『住址』以及『電話』如圖 2-4 所示。對應到圖 2-1，可以發現每一個屬性都對應到圖 2-1 右邊的欄位。以『學生』實體中的第一筆實例，該實例對應到『學號』屬性的值為 694341，對應到『姓名』屬性的值為陳小宗，對應到『住址』屬性值為臺北縣板橋致理路 1 號，對應到『電話』屬性的值為 0252210。換言之，每個實體的屬性都有它的值域，每個屬性的值域相當於所有實例對應到該屬性欄位元中所有值的集合。舉例來說，學號屬性對應到圖 2-1 右邊表格的 SNO 欄位元，因此『學號』屬性的值域為{69434101, 69434102, 69434103, …, 69434301}其他屬性的值域同理可推。

屬性值可能不只一個時，我們稱該屬性為多值屬性 (Multivalued attributes) 屬性可由數個屬性所組成時，我們稱該屬性為複合屬性 (Composite attributes) 非多值屬性我們稱為單值屬性，非複合屬性則稱為簡單屬性，有些實體是屬於同一類的，每一類實體就稱為實體型態 (Cont.) 在一個實體型態裡，有一些屬性，用來描述這些實體的性質，使得不同實體的該屬性值必然不同，我們就稱該屬性為關鍵屬性。

除上述所介紹的屬性概念之外，Chen 並進一步將屬性作以下的區分：

(一) 簡單屬性 (simple attribute) 與複合屬性 (compsite attribute)：

複合屬性代表未來該屬性可進一步作切割，例如：地址屬性是由區域號碼、縣市、鄉鎮、路、巷、弄、號等各個屬性所組成。至於學號屬性，已經無法繼續切割錢其他有意義的單位，因此，學號屬性便是簡單屬性。

(二) 關鍵屬性：是指該屬性的值在某個環境內具有唯一性，例如：

在學校環境內每個學生具有不同的學號，因此，學號屬性的值在學校環境內每個實例其學號屬性值都不一樣、具唯一性；

因此，學號屬性稱為關鍵屬性。如果一個實體有兩個或兩個以上的屬性具有關鍵的特性，則只擇一擔任關鍵屬性。其觀念與關聯式資料模型中的關鍵概念是相同的。

(三) 單值屬性 (single-valued attribute) 與多值屬性 (multi-valued attribute): 單值屬性是指每筆實例在該屬性中只會存在一個單一值，例如每個學生只會有一個學號，因此學號是單值屬性。所謂多值屬性是指每筆實例在該屬性中會存在多個數值，假設每個學生會有多支電話，因此，在電話屬性中每筆實例可能存在多支電話的號碼，如此一來，電話便是多值屬性。

(四) 衍生屬性 (derived attribute): 假設在圖 2-1 學生實例中多加入年齡與生日兩個屬性，因為生日屬性包含年、月、日三個屬性，因此，生日屬性是複合屬性。如果要計算學生年齡可以由目前的系統時間減去生日屬性的值，便可換算出年齡屬性的值；因此，年齡屬性便屬衍生屬性。

至於圖形表示方式，單值屬性與簡單屬性都採用橢圓形方表示，鍵屬性則是對橢圓形內的屬性名稱加底線，複合屬性則是以雙邊線的橢圓形表示；至於多值屬性則是橢圓形以雙直線與實體相連托，衍生屬性則是以虛線橢圓形表示如圖 2-5 所示。

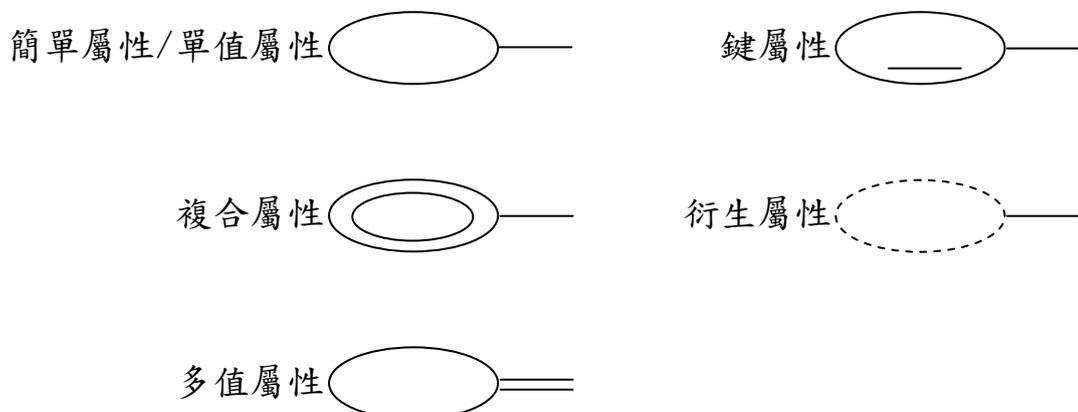


圖 2-5 各種屬性的圖形表示法

參閱圖 2-5 所示的屬性圖形表示方式，如果經過訪談我們得知，在圖 2-4 中各個學生屬性有以下進一步的描述，例如：每個同學的『學號』屬性值都是唯一的、彼此不會互相重複；每個同學的『電話』屬性中記錄兩之電話號碼，『位址』屬性中可以進一步區分出『區域號碼』以及『其他的地址資料』。則圖 2-4 的圖，便可進一步修改為如圖 2-6 所示。

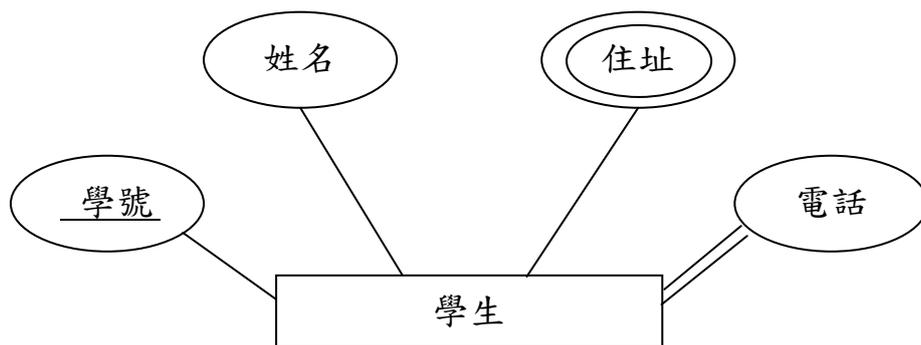


圖 2-6 各種屬性的圖形表示法

關係 (Relationship) 是指存在實體與實體之間的關聯，可能是連接一個實體、兩個實體或是多個實體。接續我們在前幾頁實體所舉的例子：當資料庫未來的使用者是學務處人員時，由於他們所關心的是學生的基本資料、緊急連絡人姓名以及聯絡方式。因此，這時的實體便是學生以及其緊急連絡人。這時存在學生與緊急連絡人中間的關係就是『對應』。整體解釋起來，就是哪一位緊急連絡人的資料是『對應』到哪一位學生。如果，資料庫應用到課務組，因為使用者關心的實體為學生、課程班級以及成績三項。因此，學生與課程兩個實體之間便存在『修習』的關係，學生與成績兩個實體之間便存在於『屬於』的關係，同時，課程與成績兩個實體之間也存在『屬於』的關係。整體解釋起來如下：某個學生『修習』某班課程，某筆成績資料是『屬於』某個同學『修習』某班課程的

成績。

對應於關聯式資料模型，ER 資料模型中的關係也有三種：分別是 1 對 1、1 對多以及多對多。茲分別說明如下：

- (一) 1 對 1：是指實體與實體之間存在 1 對 1 的關係。舉例來說：如果學務組在設計資料庫時，學務處人員告訴資料庫管理師，他們將要求學生每個人提供一個緊急連絡人資料。如此一來，每個學生只會有一組而且是唯一一組緊急連絡人資訊，而每個緊急連絡人也只會對應到一位學生，因此『學生』與『緊急連絡人』之間存在的關係便是 1 對 1。
- (二) 1 對多：是指實體與實體之間存在 1 對多的關係。舉例來說：如果我們要表達學生與指導老師之間的關係。如果對於指導老師指導學生的規則是一個學生只會有一位而且是唯一一位指導老師，而每個老師可以指導多位學生。則存在『學生』與『老師』之間的『指導』關係便是一對多。
- (三) 多對多：是指實體與實體之間存在多對多的關係。舉例來說：在課務組的資料庫中想要紀錄每個學生『修習』課程的情況，由於學校沒有規定一個同學可以修多少課，因此每個同學可以修習多門課程，換言之，每個學生至少會跟一個班級課程有關係。而個修課班級的人數一定有多位學生。因此，『學生』與『課程班級』之間的關係便是多對多，因為一位學生可以修『多』門課，而每個課程班級都會有『多』位學生修課。

ER 資料模型中是以菱形來表示關係，因為共有三種關係如圖 2-7 所示，因此個別的表達方式。

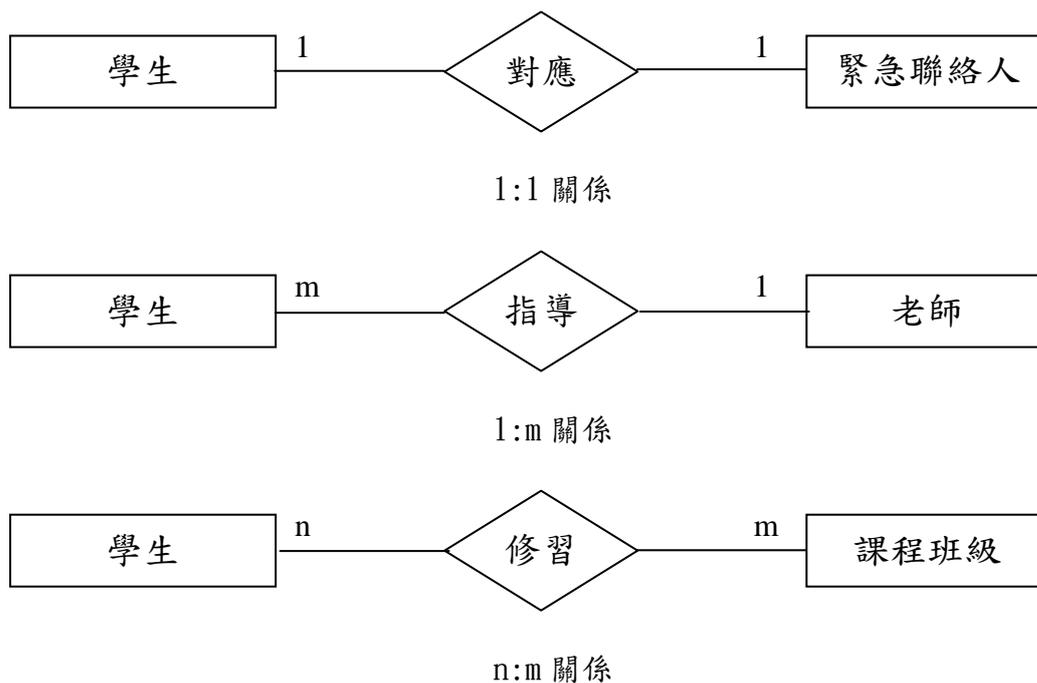


圖 2-7 三種不同的實體關係 (m 表示多)

實體關聯圖示使用圖形符號所建立的實體關聯模型，以資料庫設計來說，通常是使用在概念或邏輯資料庫設計。實體關聯圖的基本建立步驟，如下所示：

Step 1：從系統需求找出實體型態

Step 2：找出實體型態與其他實體型態間的關聯性

Step 3：定義實體型態間的關聯型態種類是：一對一、一對多或多對多關聯型態。

Step 4：定義實體型態的屬性型態與主鍵

ERD 最好要有一個命名的慣例，成為寫的人和讀的人的一個共同的默契：實體型態及關係型態的名稱的第一個字元為大寫字母、屬性的名稱第一個字元為小寫字母、角色名稱第一個字元要底線 (underscore) _、實體型態的名稱用單數名詞、關係型態的名稱用單數動詞、屬性的名稱

盡量用不同的名詞。在畫 ERD 時內容文字要盡量符合人的讀書習慣，例如：A 實體 → 關係 → B 實體，要讓看的人由左而右或由上而下很像在看一句話一樣。當一個 ER Model 建立完成之後，除了可瞭解資料庫的概念性架構外，最主要的是可以根據一定的轉換規則將實體關係圖轉換成關聯表（或稱 Table）。

在實體關聯圖使用的圖形符號，整理如圖 2-8 所示〔6〕：

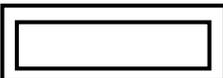
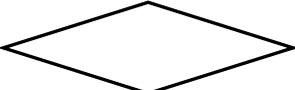
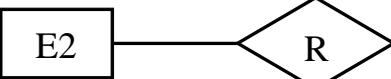
實體關聯圖的種類	圖形符號
實體	
弱實體	
關聯性	
識別關聯性	
屬性	
鍵屬性	
複合屬性	
多重值屬性	
導出屬性	
E1 全部參與 R	
E2 部分參與 R	

圖 2-8 實體關聯圖使用的圖形符號

第六節 正規化

E.F. Codd 博士的關聯式資料庫正規化理論，將正規化的步驟歸納成幾個階段，讓我們有具體可循的方法來建全資料表的結構。(註：E.F. Codd 博士是關聯式資料庫理論之父，正規化分析則是由他所發展出來的方法。)

正規劃的定義就是對於一個關聯式資料而言，將所有需要的記錄儲存於同一表格中，會有資料重複的現象，因此根據功能相依的情況進行表格分割，以減少資料重複現象的過程，就稱之為正規化。相依性：若有一欄位 A 的值可唯一決定欄位 B 的值時，則稱 B 功能相依於 A。

關聯式資料庫不管設計得好壞，都可以儲存資料，但是存取效率上可能會有很大的差別。想提升關聯式資料庫的效率，在設計資料庫的時候，可以利用正規化的方法來協助我們修改資料表的結構〔10〕。

我們所要介紹另一個關聯式資料庫的主題就是有考慮到資料重覆性的正規化格式(normal form)。按照定義，資料庫關聯又稱第一正規化(first normal form:1NF)，其含義就是沒有集合的屬性值，因為關聯是屬於1NF，故所有的關聯式模式之參考與操作方式適用。然而，1NF 關聯並非是最理想的資料表式方式，1NF 關聯可能出現重覆，假如資料庫檔案或資料檔案或資料表是以這種方式來實做，實務上重覆的資料在修改時可能產生不一致的情況。

將具有資料重覆的關聯分割成二個或更多個不具資料重覆關聯的動作叫正規化(normalization)。關聯有五個主要的正規化格式(1NF 到 5NF)，每一個正規化是針對每種特殊的資料重覆型態。建構實體關係圖及將實體關係圖轉成關聯表的設計步驟必須包含正規化的處理，否則關聯表中

可能存在一些重複的資料。正規化是將資料屬性組合成為一個具有良好結構的關聯表的過程。雖然正規化常與關聯式模式相結合，但它也是一種邏輯設計的技術，可以獨立於關聯式資料庫管理系統之外而單獨被使用。

第一正規化型式 (First Normal Form, 1NF)，主要除去關聯表中任何的重複群，使關聯表中任一行與任一列的交叉格 (Cell) 上均只有一個值。每個欄位不可有空值 (null)、主索引可查詢出唯一的資料值。

第二正規化型式 (Second Normal Form, 2NF)，符合第一正規化型式，再除去資料的部分功能相依。且不允許有部分相依。

第三正規化型式 (Third Normal Form, 3NF)，符合第二正規化型式，再除去資料的遞移相依；Boyce-Codd 正規化型式 (Boyce-Codd Normal Form, BCNF)，符合第三正規化型式，再除去任何因功能相依所造成的異常結果。條件：需符合 2NF，且不允許有遞移相依。

第四正規化型式 (Fourth Normal Form, 4NF)，符合 Boyce-Codd 正規化型式，再除去所有的多值相依。

第五正規化型式 (Fifth Normal Form, 5NF)，符合第四正規化型式，再除去剩餘的所有異常情況等〔11〕。

一般來說，在實務上常應用至第三正規化型式，所以本組也就只介紹到第三皆正規化。

第一階正規化：

正規化的過程是循序漸進的，資料表必須在滿足第一階正規化的條件之下，才能進行第二階正規化。也就是說，第二階正規化必須建立在符合第一階正規化的資料表上，依此類推。因此，第一階正規化是所有正規化的基礎。第一階正規化 (1NF) 有以下幾個規則：

(一) 資料表中必須有 Primary Key，而其他所有的欄位都「相依」於 Primary Key。「相依」是指一個資料表中，若欄位 B 的值必須搭配欄位 A 才有意義，就是「B 欄位相依於 A 欄位」。舉例來說，某一員工資料表如表 2-1 所示：

表 2-1 員工資料表

*員工編號	姓名	地址
1052	陳佑宗	臺北縣板橋市文化路一段 313 號
1074	劉靜雯	臺北縣板橋市文化路一段 313 號

上表 2-1 所示的員工編號欄為 Primary Key，做為唯一辨識該筆記錄的欄位。對此資料表來說，地址欄必須要相依於員工編號欄才有意義。否則“陳佑宗”和“劉靜雯”的地址都相同，若是以地址做為 Primary Key，就無法從地址來區別是哪一個人了！同樣地，姓名欄也必須相依於員工編號欄。

- (二) 每個欄位中都只儲存單一值，例如同一筆記錄的姓名欄位中不能存放 2 個人的姓名。
- (三) 資料表中沒有意義相同的多個欄位，例如姓名 1、姓名 2... 等重複的欄位。反之，若資料表的欄位不符合以上規則，則稱為「非正規化」的資料表。

不符合 1NF 資料表的缺點，首先我們來看一個非正規化的訂單資料表如表 2-2 所示：

表 2-2 非正規化資料表

訂單編號	客戶名稱	員工編號	負責業務員	書號	書籍名稱	數量
CH101	金瓜書局	1052	陳佑宗	L5301	JAVAGO	20
				L5120	SQLGO	60
				L5662	HTMLGO	30
CH103	誠信書局	1072	劉靜雯	L5662	HTMLGO	80

這個資料表表 2-2 所示，乍看之下清楚明瞭，但卻有以下兩個缺點：

- (一)「書號」、「書籍名稱」及「數量」欄的長度無法確定：由於訂購的書籍種類可多可少，不同的客戶訂購的種類也不相同，所以必須預留很大的空間給這些欄位，如此反而造成儲存空間的浪費。
- (二)降低存取資料的效率：例如要找出“HTMLGO”的訂單數量，必須先在書籍名稱欄中找出“HTMLGO”所在的位置，然後才能從數量欄中擷取出對應的數量資料，這不僅減緩了資料處理的速度，而且也增加了程式出錯的機會。

顯然的，此資料表違反了 1NF 的第 2 個規則。另外，兩家客戶同時購買了「HTMLGO」這本書，也可能同一家客戶在不同定單中購買同一本書，因此這個資料表缺少具有唯一性的 Primary Key，也違反了 1NF 的第 1 個條件。

接著再看一個不符合 1NF 第 3 個規則的例子如表 2-3 所示：

表 2-3 非正規化資料表

訂購客戶	書籍 1	書籍 2	書籍 3	數量 1	數量 2	數量 3
金瓜書局	JAVAGO	SQLGO	HTMLGO	20	60	30
誠信書局	HTMLGO			80		

像書籍 1、書籍 2、書籍 3 這樣一群意義相同的欄位，其問題同樣是無法確定要有多少個重複的欄位，而且存取效率低。例如要找“JAVAGO”的數量，必須在書籍群組的每一個欄位中搜尋，找到後還得到數量群組中的相同位置欄位中讀取，相當麻煩。

建構 1NF 資料表的方法：對於不具 1NF 形式的訂單資料表，我們可將重複的資料項分別儲存到不同的紀錄中，並加上適當的 Primary Key (標示*符號者為 Primary Key) 如表 2-4 所示，產生如下的訂單資料如表 2-4 所示：

表 2-4 正規化資料表

*訂單編號	*書號	員工編號	負責業務員	客戶名稱	書籍名稱	數量
CH101	L5301	1052	陳佑宗	金瓜書局	JAVAGO	20
CH101	L5120	1052	陳佑宗	金瓜書局	SQLGO	60
CH101	L5662	1052	陳佑宗	金瓜書局	HTMLGO	30
CH103	L5662	1074	劉靜雯	誠信書局	HTMLGO	80

如此一來，雖然增加了許多筆記錄，但每一個欄位的長度及數

目都可以固定，而且我們可以用訂單編號欄加上書號欄做為 Primary Key，那麼在查詢某家客戶訂購某本書的數量時，就非常地方便快捷了。

第二階正規化：

將上述訂單資料表執行 1NF 之後，應該很容易察覺：我們輸入許多重複的資料。如此，不但浪費儲存的空間，更容易造成新增、刪除或更新資料時的異常狀況。所以，我們必須接著進行第二階正規化，來消除這些問題。第二階正規化（2NF）有以下幾個規則：

（一）必須符合 1NF 的格式。

（二）各欄位與 Primary Key 間沒有「部分相依」的關係。

「部分相依」只有在 Primary Key 是由多個欄位組成時才會發生，它是指某些欄位只與 Primary Key 中的部分欄位元有相依性，而與另一部分的欄位沒有相依性。

以前例的訂單資料表來說，其 Primary Key 為訂單編號+書號欄位，但客戶名稱欄只和訂單編號欄有相依性（一筆訂單只對應一家客戶），而書籍名稱欄只和書號欄有相依性（一個書號只對應一本書）如表 2-5 所示：

表 2-5 部分相依

書籍名稱只和書號有相依性

訂單資料表

* 訂 單 編號	*書號	員 工 編 號	負 責 業 務 員	客 戶 名 稱	書 籍 名 稱	數 量
CH101	L5301	1052	陳佑宗	金瓜書 局	JAVAGO	20
CH101	L5120	1052	陳佑宗	金瓜書 局	SQLGO	60
CH101	L5662	1052	陳佑宗	金瓜書 局	HTMLGO	30
CH103	L5662	1074	劉靜雯	誠信書 局	HTMLGO	80

客戶名稱只和訂單編號有相依性

部分相依會造成下列問題：

- (一) 新增資料時可能會無法輸入：若有一新出的書籍“JAVAGO”，但還沒有任何客戶訂購，那麼它的資料將無法輸入（因為 Primary Key 中的欄位質不允許 NULL 值，但此時根本沒有訂單編號可輸入）。
- (二) 更改資料時沒有效率：當客戶名稱“金瓜書局”更改為“南

瓜書局”時，必須搜尋整個資料庫並一一更改，非常沒有效率。

(三) 刪除資料時可能會同時刪除仍有用的資料：由於“SQLGO”僅在“CH101”這筆訂單中被訂購，如果將該筆記錄刪除，那麼“SQLGO”的資料也就跟著消失了。

另外，部分相依也會造成資料重複出現的問題，例如“CH101，L5301”這組資料每次都必須同時輸入，不但浪費時間及儲存空間，而且也容易疏忽而造成資料不一致的錯誤。

建構 2NF 資料表的方法：要除去資料表中的部分相依性，只需將部份相依的欄位分割成另外的資料表即可。例如我們將訂單資料表分割成 3 個較小的資料表如表 2-6、表 2-7、表 2-8 所示(表示“*”號的欄位為 Primary Key)：

表 2-6 出貨記錄表

*訂單編號	*書號	數量
CH101	L5301	20
CH101	L5120	60
CH101	L5662	30
CH103	L5662	80

表 2-7 書籍資料表

*書號	書籍名稱
L5301	JAVAGO
L5120	SQLGO
L5662	HTMLGO

表 2-8 訂單資料表

*訂單編號	客戶名稱	員工編號	負責業務員
CH101	金瓜書局	1052	陳佑宗
CH103	誠信書局	1074	劉靜雯

分成 3 個資料表後，便去除了原本資料表的“部分相依性” 如

圖 2-9 所示，我們來看看關聯式資料表可以更容易明白：

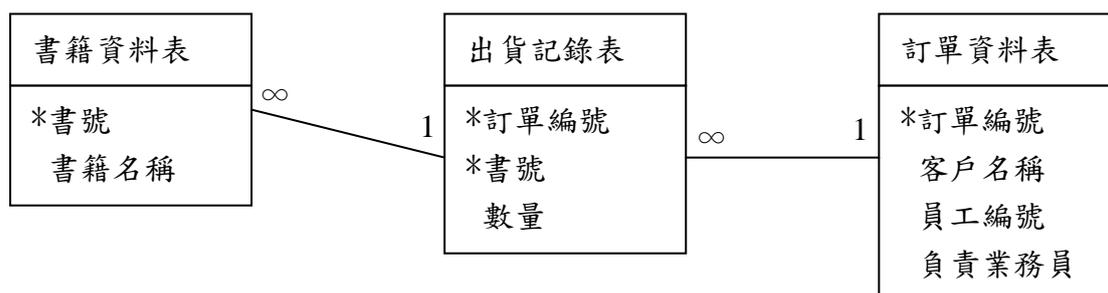


圖 2-9 關聯是資料表圖

第三階正規化：

經過 2NF 後的資料表，其實還存在一些問題：

- (一) 在訂單資料表中，如果有新進業務同仁“林億綾”，在該員尚未安排負責客戶之前，我們無法輸入該員的資料。
- (二) 若要刪除金瓜書局的這家客戶，勢必會將負責該客戶的業務同仁“陳佑宗”一併刪除。
- (三) 若要更改 HTMLGO 這本書的名稱，則必須同時更改多筆記錄（同一本書，會有多筆訂單），造成不便。

基於上述理由，我們必須在執行第三階正規化，第三階正規化（3NF）有以下幾個要件：

- (一) 符合 2NF 的格式
- (二) 各欄位與 Primary Key 間沒有“間接相依”的關係，“間接相依”是指二個欄位間並非直接相依，而是借住第三個欄位來達成資料相依的關係，例如 A 相依於 B；而 B 又相依於 C，如此 A 與 C 之間就是間接相依的關係。

要找出各欄位與 Primary Key 間の間接相依性，最簡單的方式就是看看資料表中有沒有“與 Primary Key 無關的相依性”存在如表 2-9 所示。例如在訂單資料表 2-9 中：

表 2-9 訂單資料表

*訂單資料表	客戶名稱	員工編號	負責業務員
CH101	金瓜書局	1052	陳佑宗
CH103	誠信書局	1074	劉靜雯

由於每筆訂單都會有一位業務員負責，所以員工編號欄和負責業務員欄都相依於訂單編號欄。但負責業務員又同時相依於員工編號欄，而這個相依性是與 Primary Key 完全無關的如圖 2-10 所示：

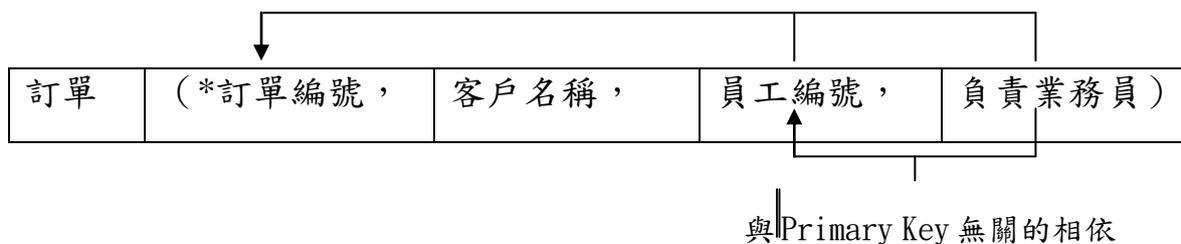


圖 2-10 與 Primary Key 完全無關的

事實上，它們之間的相依關係為如圖 2-11 所示：

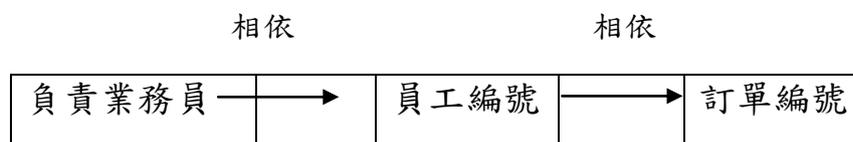


圖 2-11 相依關係

由此可知，負責業務員與 Primary Key 存在著無關的相依性，也就是有“間接相依”的關係存在。

建構 3NF 資料表的方法：要除去資料表中的“間接相依性”，其方法和除去“部分相依性”完全相同。例如訂單資料表可在分隔成兩個資料表如表 2-10、表 2-11 所示：

表 2-10 訂單資料表

*訂單編號	客戶名稱	負責員工編號
CH101	金瓜書局	1052
CH103	誠信書局	1074

表 2-11 員工資料表

*員工編號	員工姓名
1052	陳佑宗
1074	劉靜雯

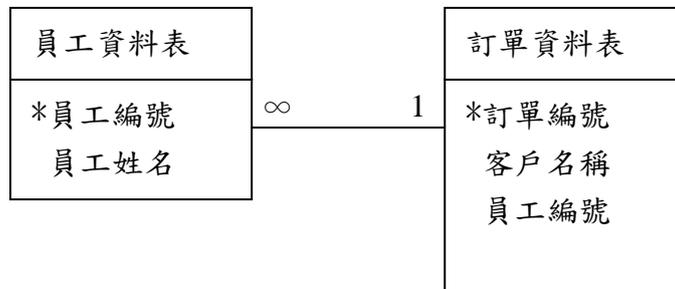


圖 2-12 表 2-10、2-11 資料表關聯圖

我們來看看這 2 個資料表的關聯如圖 2-12 所示，這樣負責業務員的“間接相依性”便被去除了。

與直覺式的分割技巧做比較：當您設計資料庫一段時間，累積了經驗及技術後，您便可依照自己的經驗，以直覺的方式對資料表執行最佳化，如表 2-12 所示是兩種方法再功能上的對照：

表 2-12 對照表

正規化	功能相同的直覺式分割法
1NF：有主鍵 欄位中只有一個單值 沒有意義相同的重複欄位	無
2NF：除去“部分相依性”	分割“欄位值一再重複”的欄位
3NF：除去“間接相依性”	分割“與主鍵無關”的欄位

第三章 網路 e 化課程輔助暨管理資訊系統

第一節 系統架構

本組研究此系統是希望能夠幫助教師可以利用電腦方便管理學生方面的業務，所以本系統有教師端與學生端，教師可以透過登錄教師端的教師主畫面來進行新增開課、查詢或刪除、審核學生資料、登錄平時成績、登錄期中/期末成績等功能；學生可以透過登錄學生端的學生畫面來進行選課、查詢選課、成績查詢、上傳作業等功能如圖 3-1 所示。其基本系統架構詳：

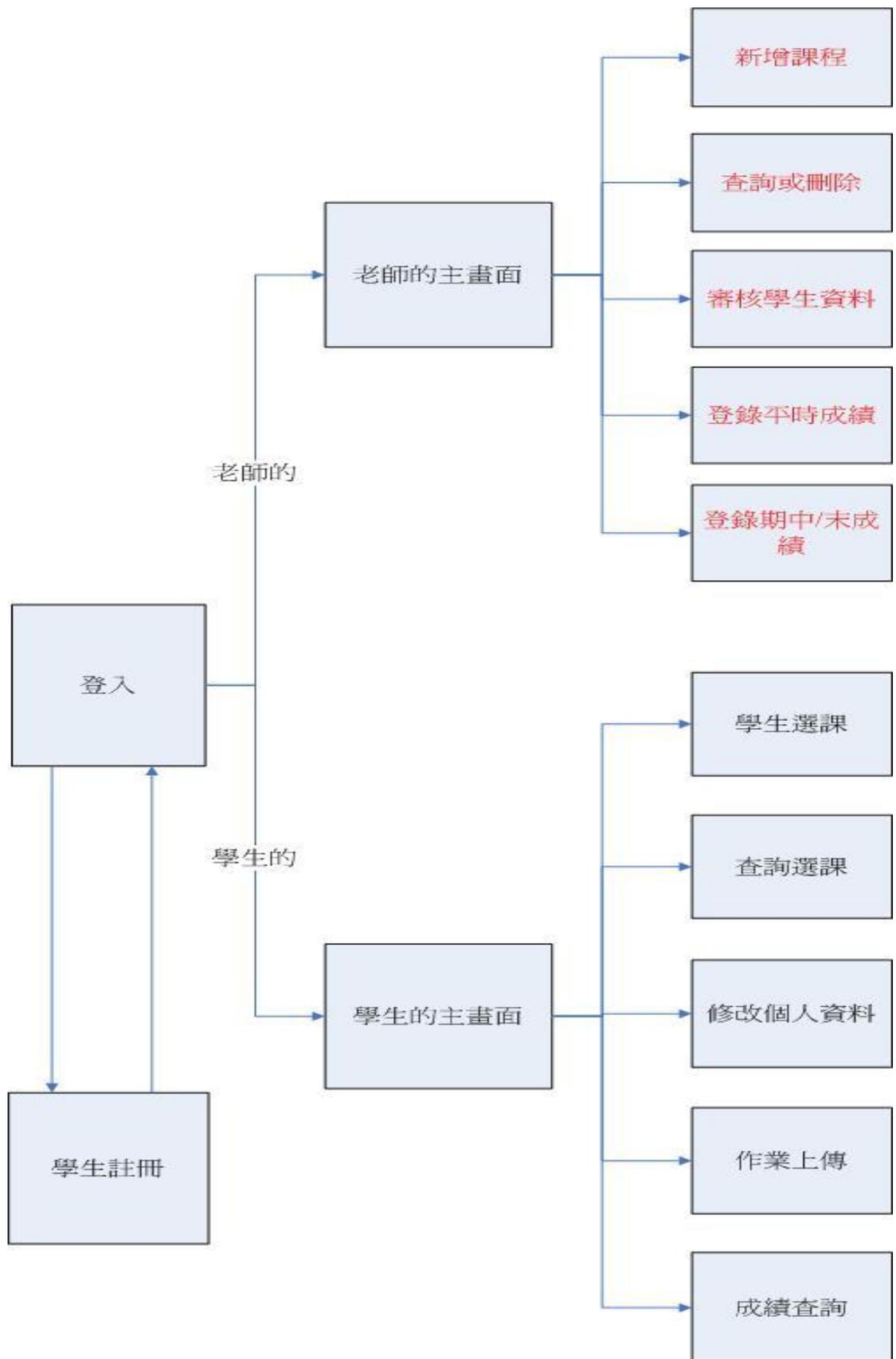


圖 3-1 系統架構

第二節 啟用 SQL Server2005 資料庫

系統網站做好後當然需要一個資料庫來儲存教師端或學生端的資訊、資料或檔案，就像賣場一樣，要有一個儲存貨物的倉庫。所以本組就選擇使用了 SQL Server2005 資料庫。

而 SQL Server2005 資料庫是一套多使用者、多執行的資料庫伺服器，不但簡單易懂方便，還有免費版可以使用。以下是安裝 SQL Server2005 資料庫步驟：

1. 首先要先確定該電腦是否已經先更新微軟 KB893803-v2-x86 的更新程式，否則要先上微軟下載「WindowsInstaller-KB893803-v2-x86.exe」如圖 3-2 所示，更新程式安裝。



圖 3-2 微軟更新安裝畫面

2. 微軟 KB893803 更新完後就可以安裝 SQL Server 2005 資料庫了，首先我們先安裝 SQL Server 2005 資料庫的「SQLEXPRESS_CHT.EXE」如圖 3-3 所示。

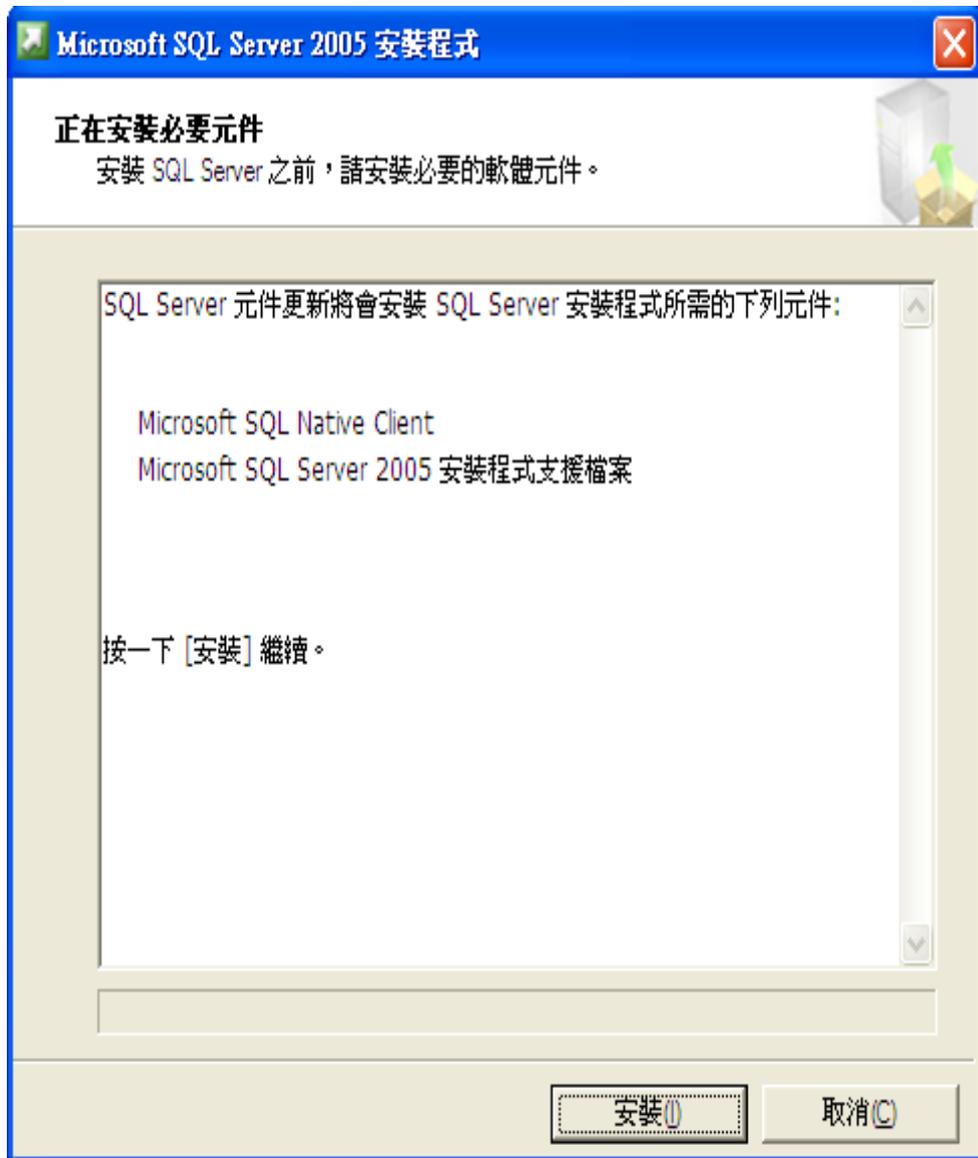


圖 3-3 SQL Server 2005 資料庫安裝畫面 1

SQL Server2005 資料庫安裝過程中的系統組態檢查，如圖 3-4 所示。



圖 3-4 SQL Server2005 資料庫安裝畫面 2

SQL Server2005 資料庫安裝過程中的功能選擇，可根據使用者的需求決定安裝哪些功能如圖 3-5 所示，每個功能右邊都有文字敘述解釋。

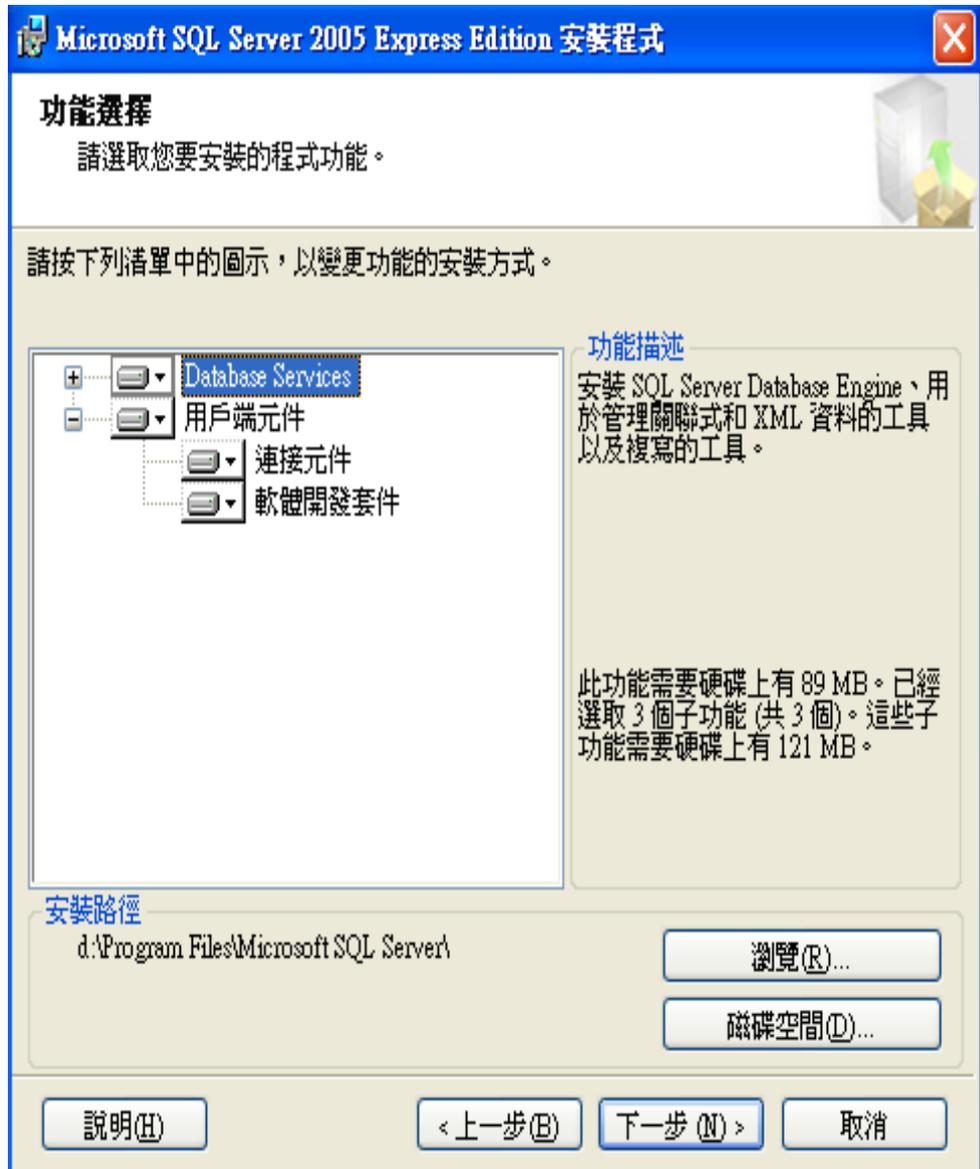


圖 3-5 SQL Server2005 資料庫安裝畫面 3

SQL Server2005 資料庫安裝過程中的驗證模式選擇如圖 3-6 所示，一般都是選擇第一項「Windows 驗證模式 (W)」，在按「下一步」。



圖 3-6 SQL Server2005 資料庫安裝畫面 4



圖 3-7 SQL Server2005 資料庫安裝畫面 5



圖 3-8 SQL Server2005 資料庫安裝畫面 6

3.安裝完 SQL Server2005 資料庫的「SQLEXPRESS_CHT.EXE」，再來我們就要再安裝「SQLServer2005_SSMSEE.msi」如圖 3-9 所示這樣才算是完成唷！



圖 3-9 SQL Server2005 資料庫安裝畫面 7

同樣的，安裝過程中也有功能選擇如圖 3-10 所示，右邊也有中文介紹。

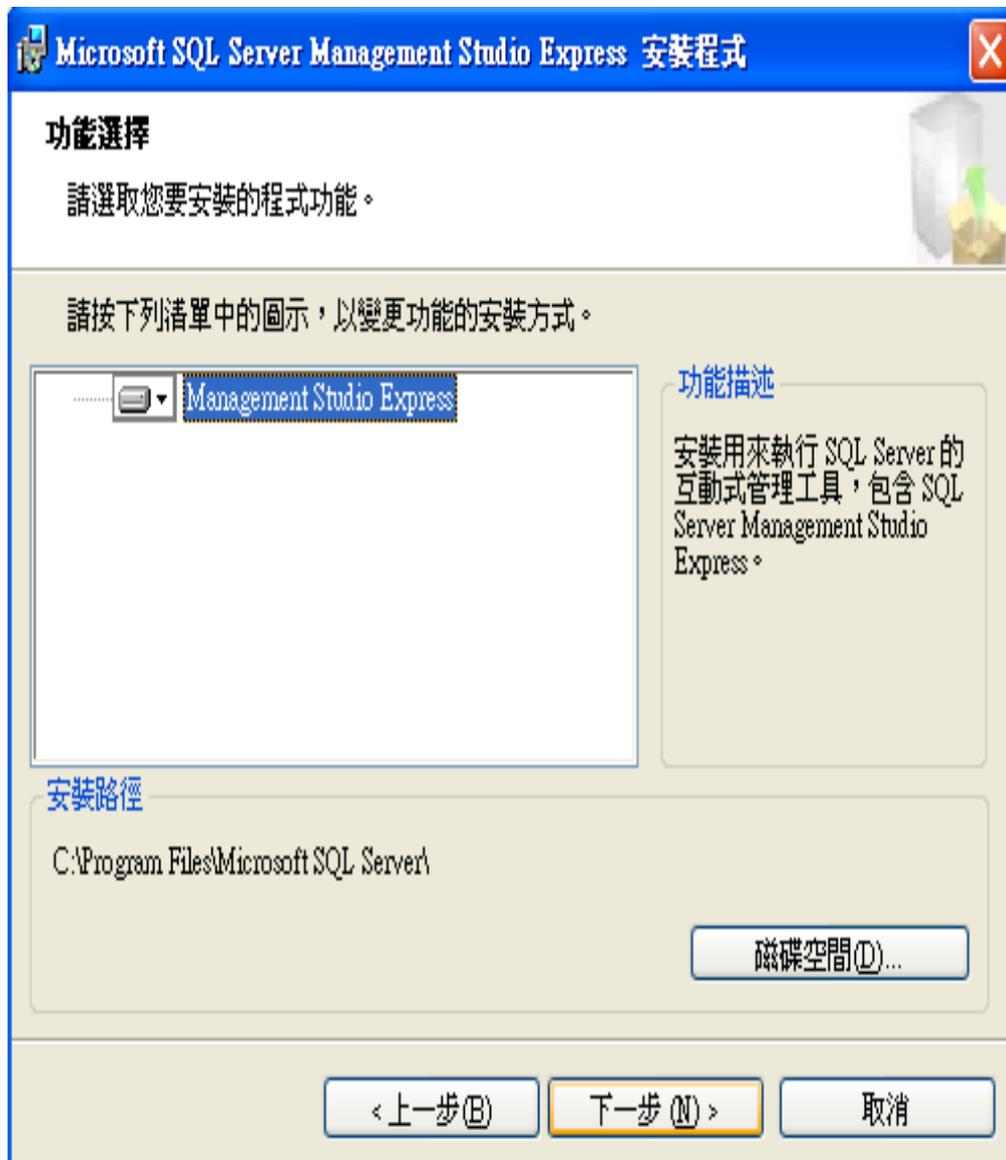


圖 3-10 SQL Server2005 資料庫安裝畫面 8

安裝完 SQL Server2005 資料庫「SQLServer2005_SSMSEE.msi」如圖 3-11 所示，這樣才算是完整的把 SQL Server2005 資料庫安裝完畢唷！

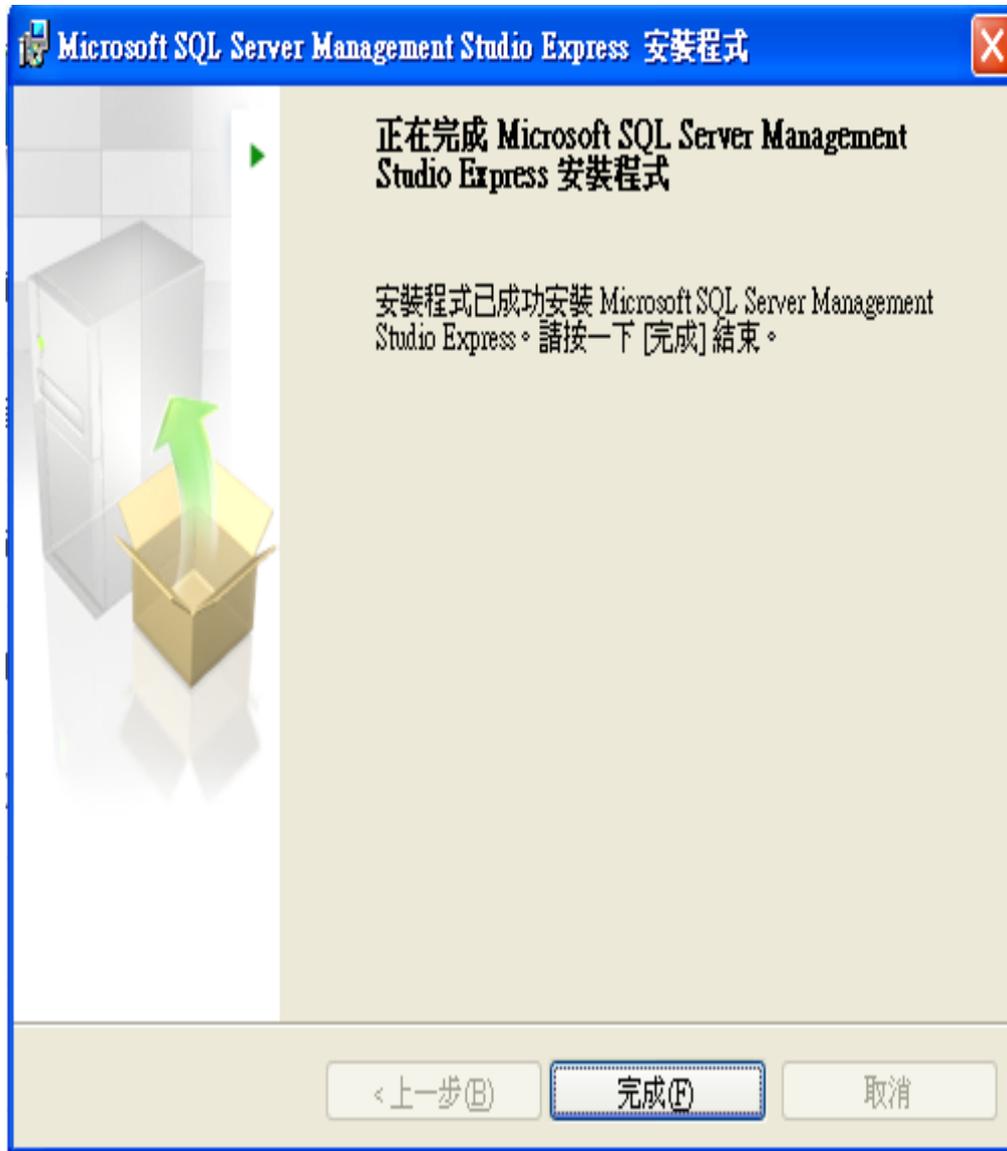


圖 3-11 SQL Server2005 資料庫安裝畫面 9

4.再來就是放置資料如圖 3-12 所示，首先我們要先打開剛剛安裝圖 SQL Server2005 資料庫的資料夾尋找出「Microsoft SQL Server\MSSQL.1\MSSQL\Data」資料夾的位置並打開，將「work_log.ldf」跟「work.mdf」複製上去。



圖 3-12 SQL Server2005 資料庫安裝畫面 10

複製如圖 3-13 所示到「Microsoft SQL Server\MSSQL.1\MSSQL\Data」資料夾內。

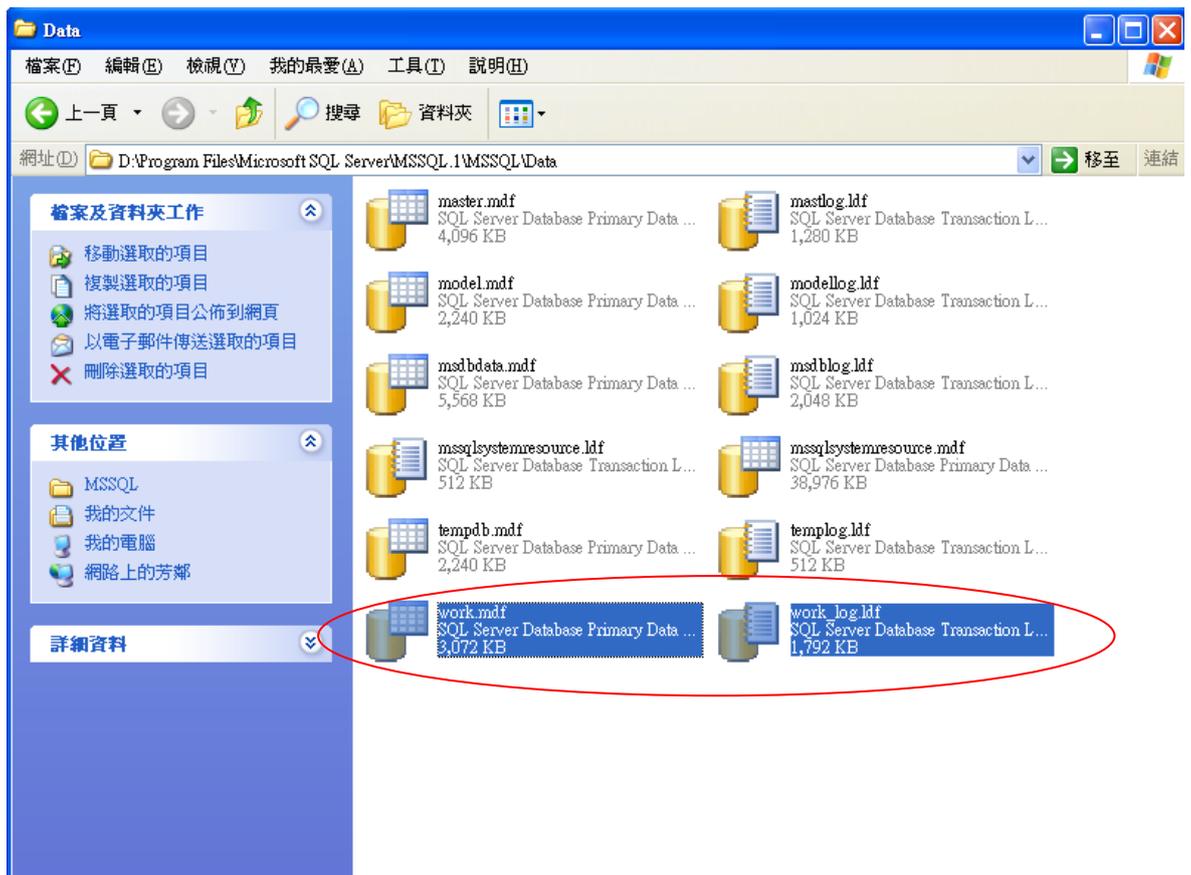


圖 3-13 SQL Server2005 資料庫安裝畫面 11

再來我們要先登入 SQL Server2005 資料庫程式如圖 3-14 所示，打開「SQL Server Management Studio Express」登入後會出現連接伺服器畫面，驗證那邊，一樣選擇 Windows 驗證模式。

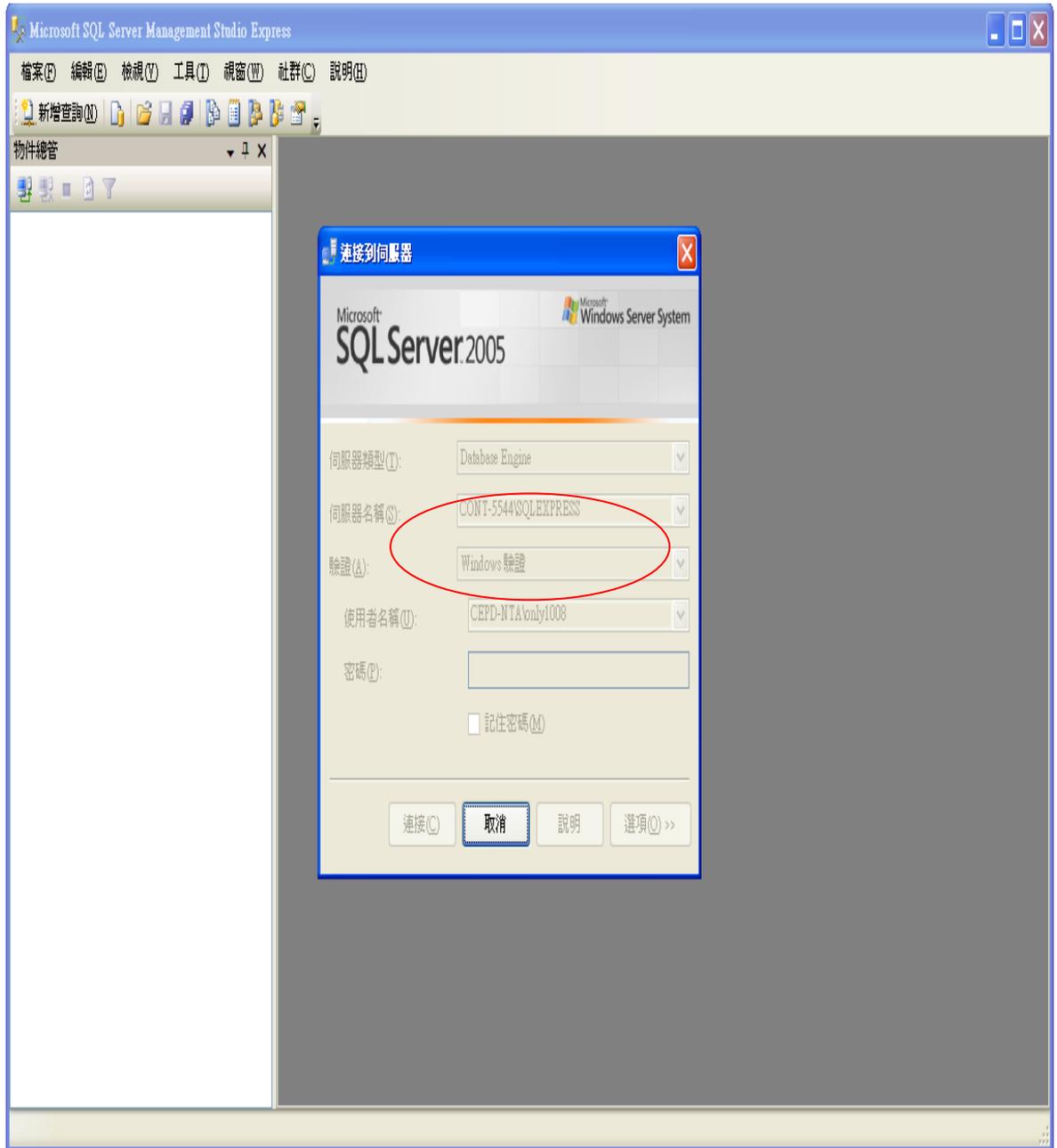


圖 3-14 SQL Server2005 資料庫安裝畫面 12

登入 SQL Server2005 資料庫後如圖 3-15 所示，我們先尋找左邊選項的「資料庫」，並用滑鼠點右鍵，點選「附加」。

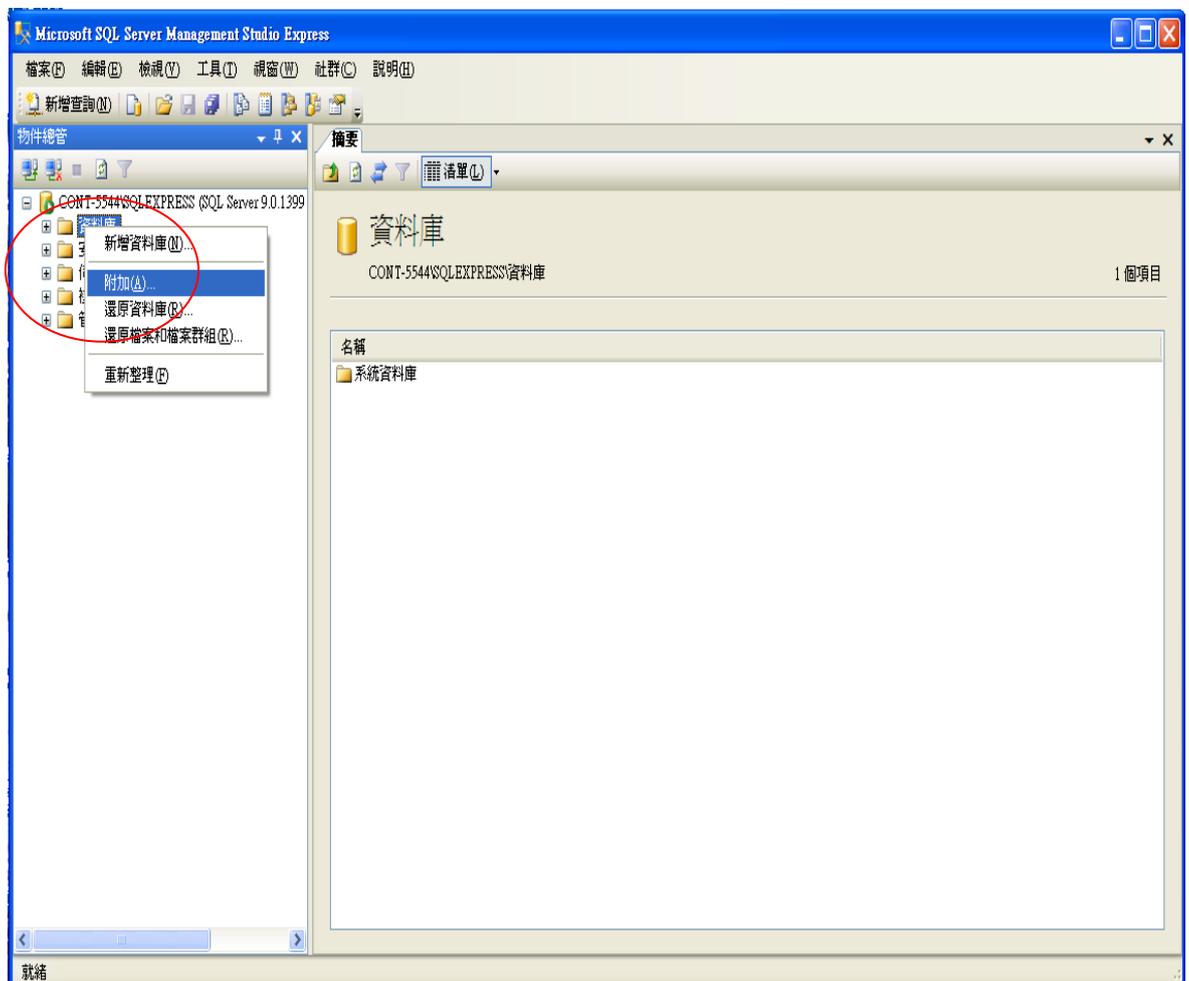


圖 3-15 SQL Server2005 資料庫安裝畫面 13

進入 SQL Server2005 資料庫「附加資料庫」的畫面後如圖 3-16 所示，選擇右邊中間選項的「加入」。

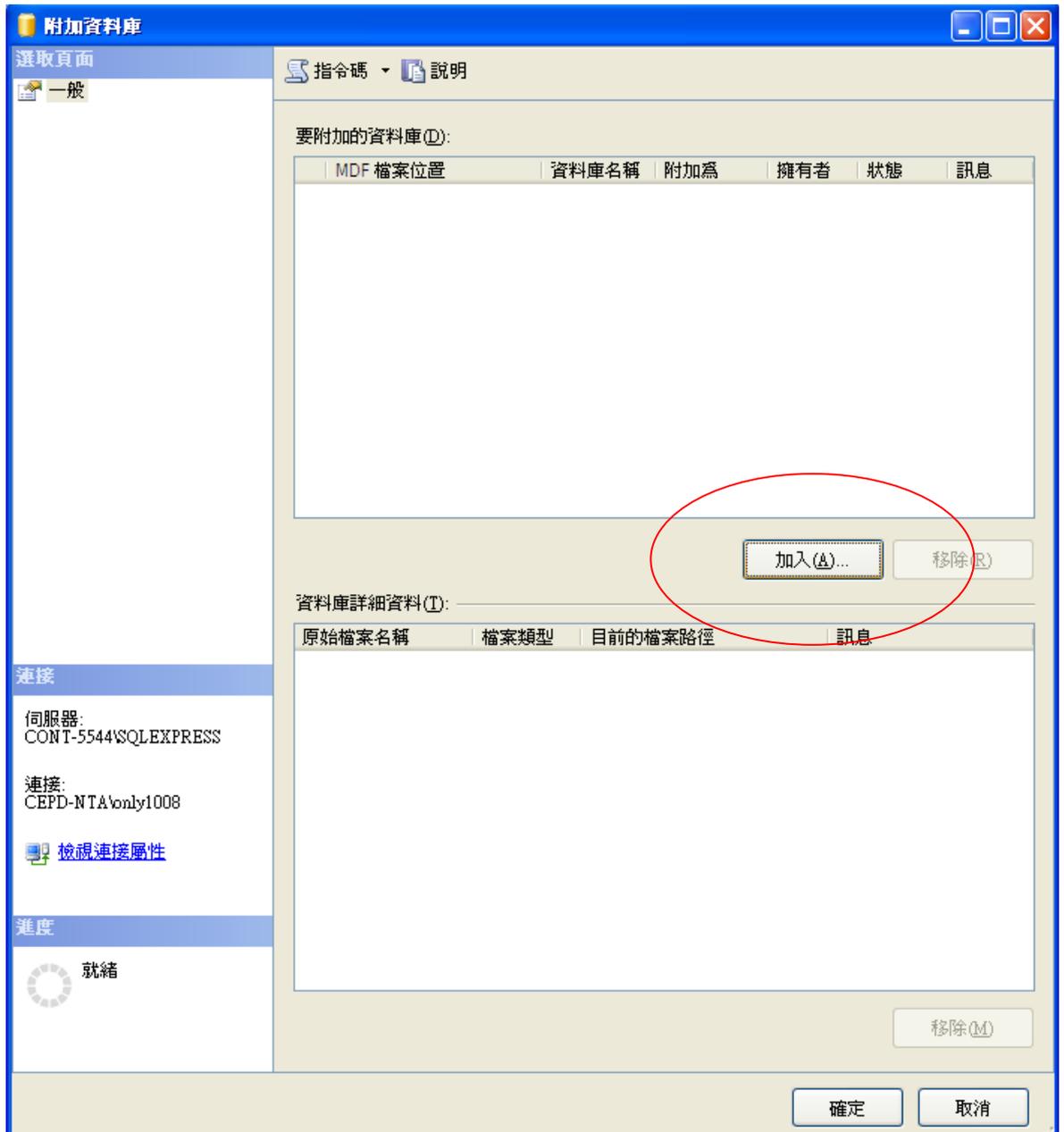


圖 3-16 SQL Server2005 資料庫安裝畫面 14

點選加入後，會出現一個選取檔案的畫面如圖 3-17 所示，我們必須要尋找出，之前放置「work.mdf」的「Microsoft SQL Server\MSSQL.1\MSSQL\Data」資料夾位置，並點選「work.mdf」，按下「確定」。

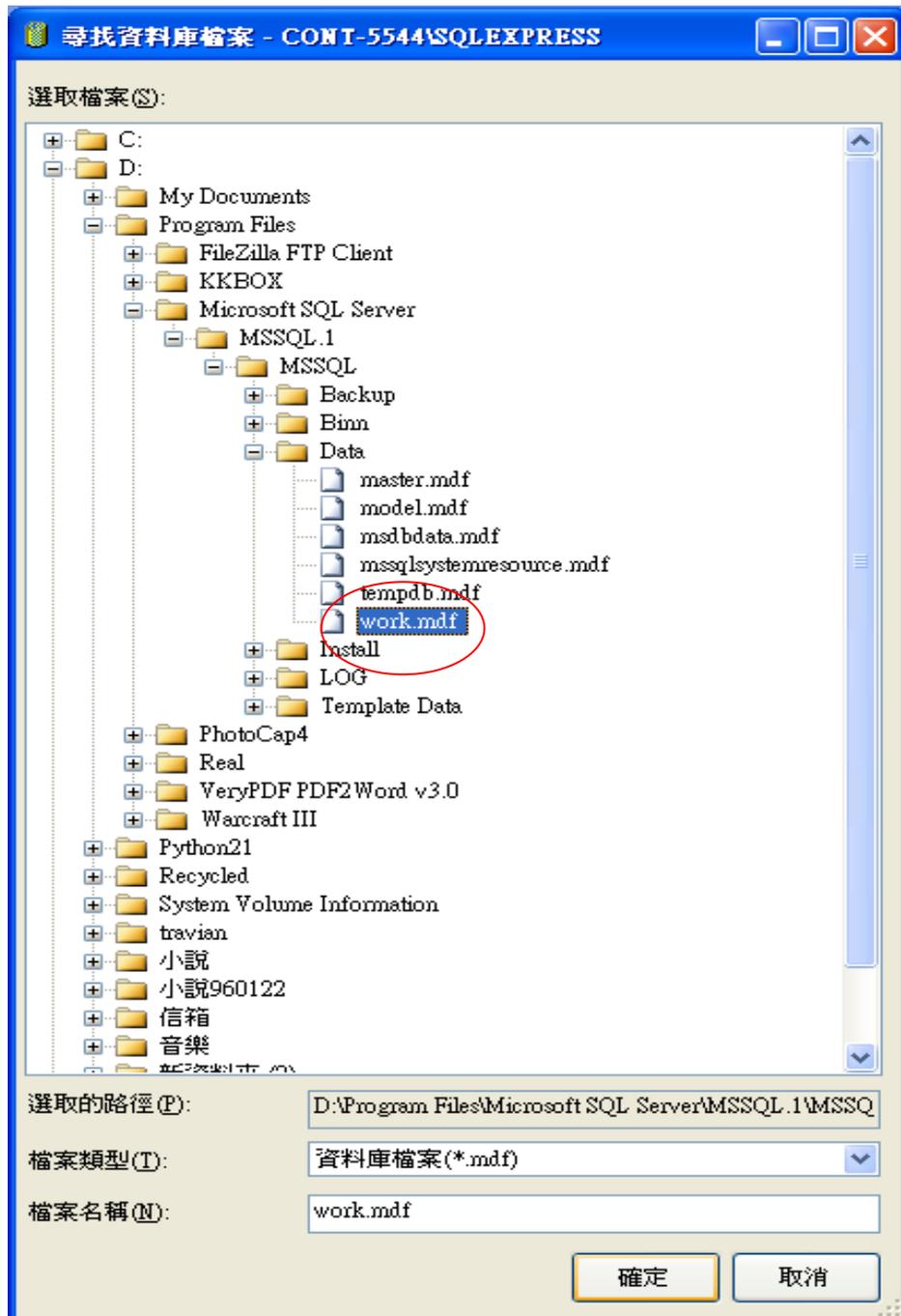


圖 3-17 SQL Server2005 資料庫安裝畫面 15

選取確定之後會回到「附加資料庫」的畫面如圖 3-18 所示，右下框內就會出現之前放置的「work_log.ldf」跟「work.mdf」並按下確定，這樣就算完整的薑我們所需的資料放置完畢。

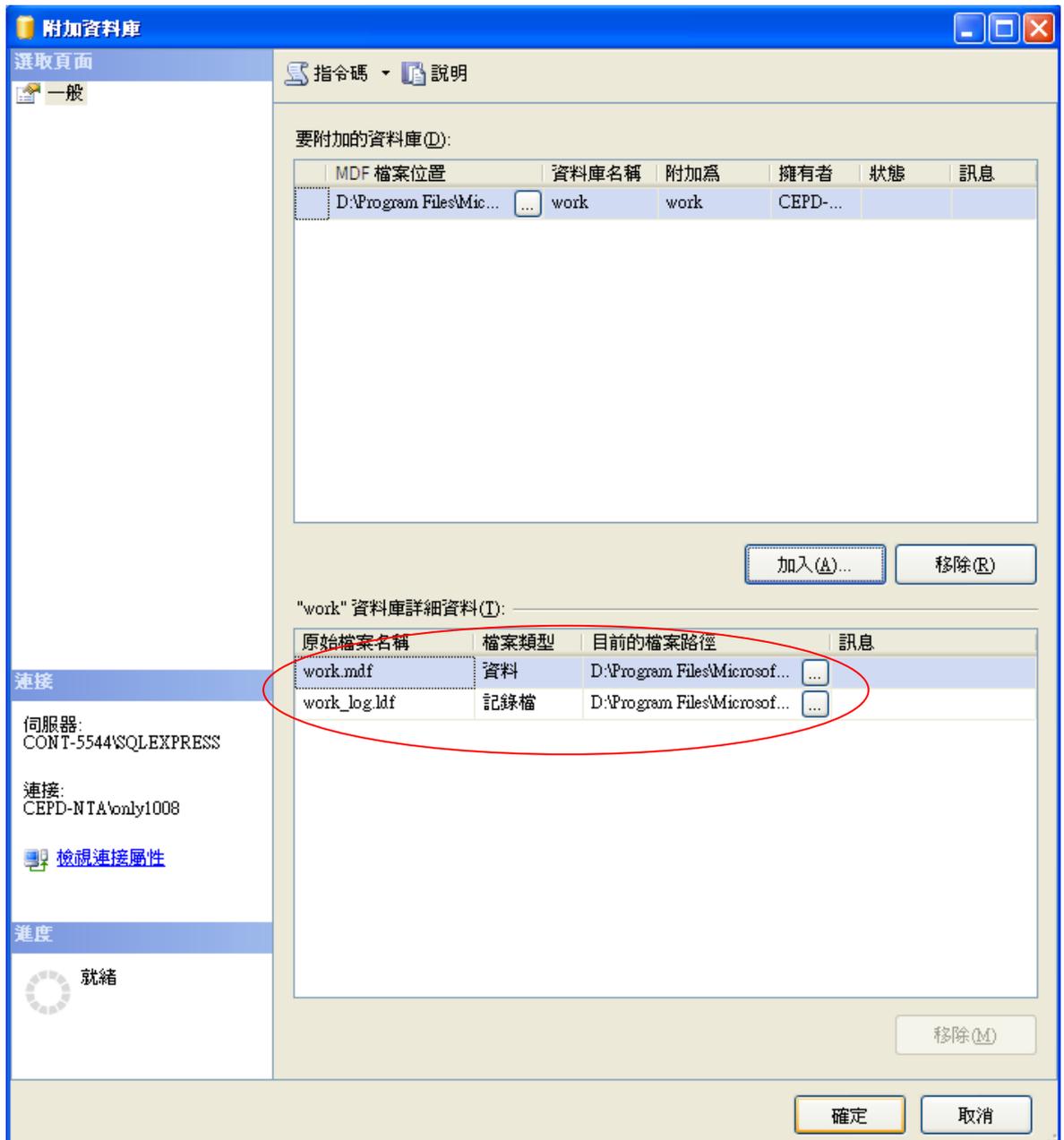


圖 3-18 SQL Server2005 資料庫安裝畫面 16

第三節 本系統資料庫的 ERM

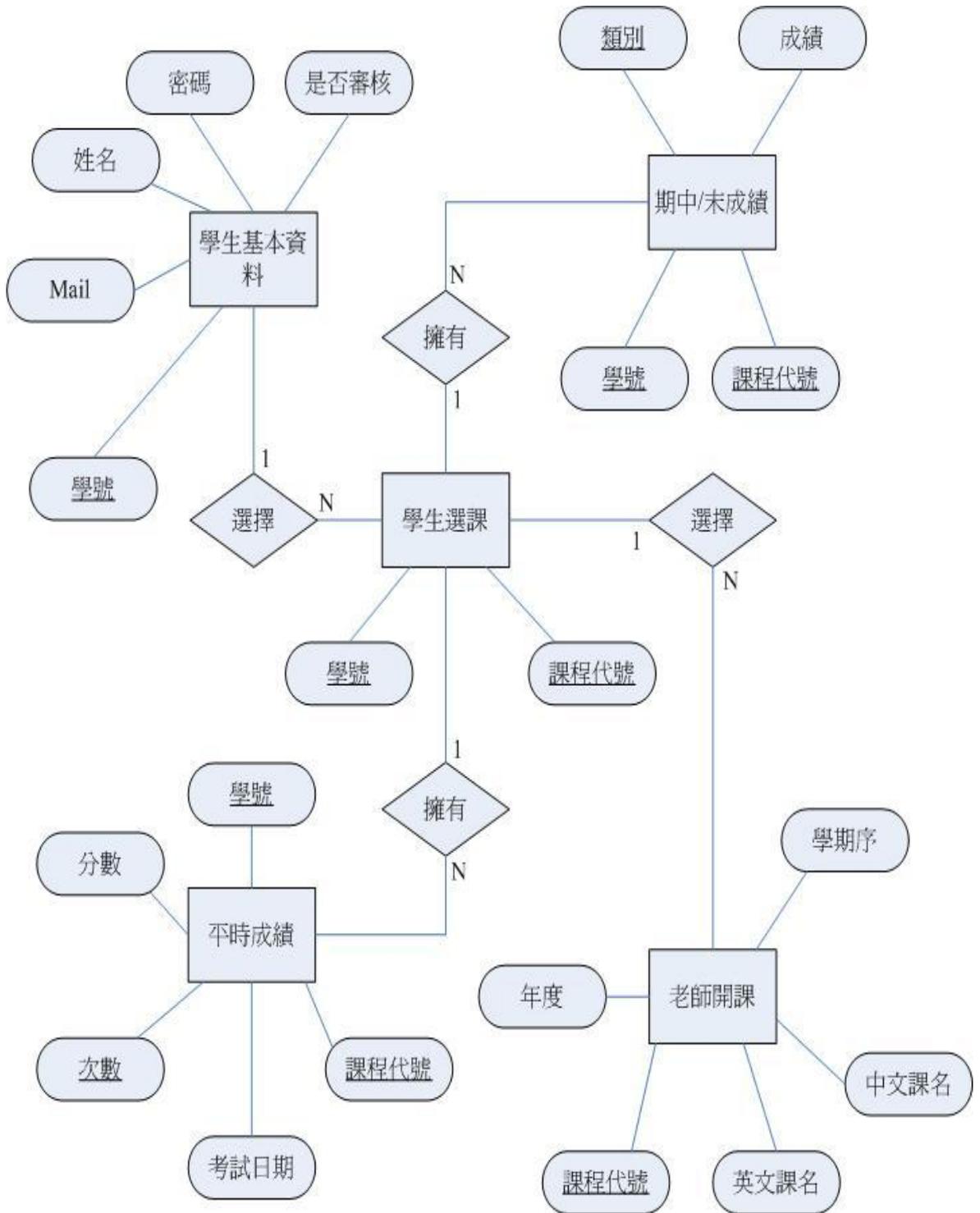


圖 3-19 資料庫分析圖

第四節 系統資料庫

系統中共有 5 個資料表如表 3-1、表 3-2、表 3-3、表 3-4 及表 3-5 所示，各資料表：

表 3-1 系統資料表-老師開課資料表

LE 老師開課資料表

學年	年度	中文課名	英文課名	課程代號	備註
----	----	------	------	------	----

表 3-2 系統資料表-學生資料資料表

ST 學生資料資料表

學號	姓名	密碼	e-mail	是否審核
----	----	----	--------	------

表 3-3 系統資料表-學生選課資料表

STLE 學生選課資料表

學號	課程代號
----	------

表 3-4 系統資料表-學生平時成績資料表

SCORE 學生平時成績資料表

學號	課程代號	次數	分數
----	------	----	----

表 3-5 系統資料表-學生期中/末成績資料表

IM_SCORE 學生期中/末成績資料表

學號	課程代號	類別	分數
----	------	----	----

資料庫欄位如表 3-6、表 3-7、表 3-8、表 3-9、表 3-10 所示：

表 3-6 資料庫欄位-老師開課資料表

LE 老師開課資料表

欄位名稱	屬性	長度	允許 null	說明
LE_year	varchar	4	否	學年
LE_term	nvarchar	3	否	年度
LE_name	nvarchar	20	否	中文課名
LE_Ename	nvarchar	10	否	英文課名
LE_SN	varchar	20	否	課程代號
mark	nvarchar	50	是	備註

表 3-7 資料庫欄位-學生資料資料表

學生資料資料表

ST

欄位名稱	屬性	長度	允許 null	說明
ST_id	varchar	8	否	學號
ST_name	nvarchar	20	否	姓名
ST_pwd	varchar	10	否	密碼
ST_mail	nvarchar	40	否	e-mail
status	char	2	否	是否審核

0 為未審核，1 為已審核

表 3-8 資料庫欄位-學生選課資料表

學生選課資料表

STLE

欄位名稱	屬性	長度	允許 null	說明
ST_id	varchar	8	否	學號
ST_lesson	nvarchar	20	否	課程代號

表 3-9 資料庫欄位-學生平時成績資料表

學生平時成績資料表

SCORE

欄位名稱	屬性	長度	允許 null	說明
ST_id	varchar	8	否	學號
LE_SN	nvarchar	20	否	課程代號
status	int	2	是	次數
score	int	3	是	分數

表 3-10 資料庫欄位-學生期中/末成績資料表

學生期中/末成績資料表

IM_SCORE

欄位名稱	屬性	長度	允許 null	說明
ST_id	varchar	8	否	學號
LE_SN	nvarchar	20	否	課程代號
status	int	2	是	類別
score	int	3	是	分數

1 為期中考，2 為期末考

第五節 啟動 JSP 環境

本組主要是利用 JSP 對資料庫跟網頁之間做為聯結的橋樑。首先我們要先安裝 JDK6.10 如圖 3-20 所示，所以我們直接從官方網站所提供的程式下載「<http://java.sun.com/javase/downloads/index.jsp>」。

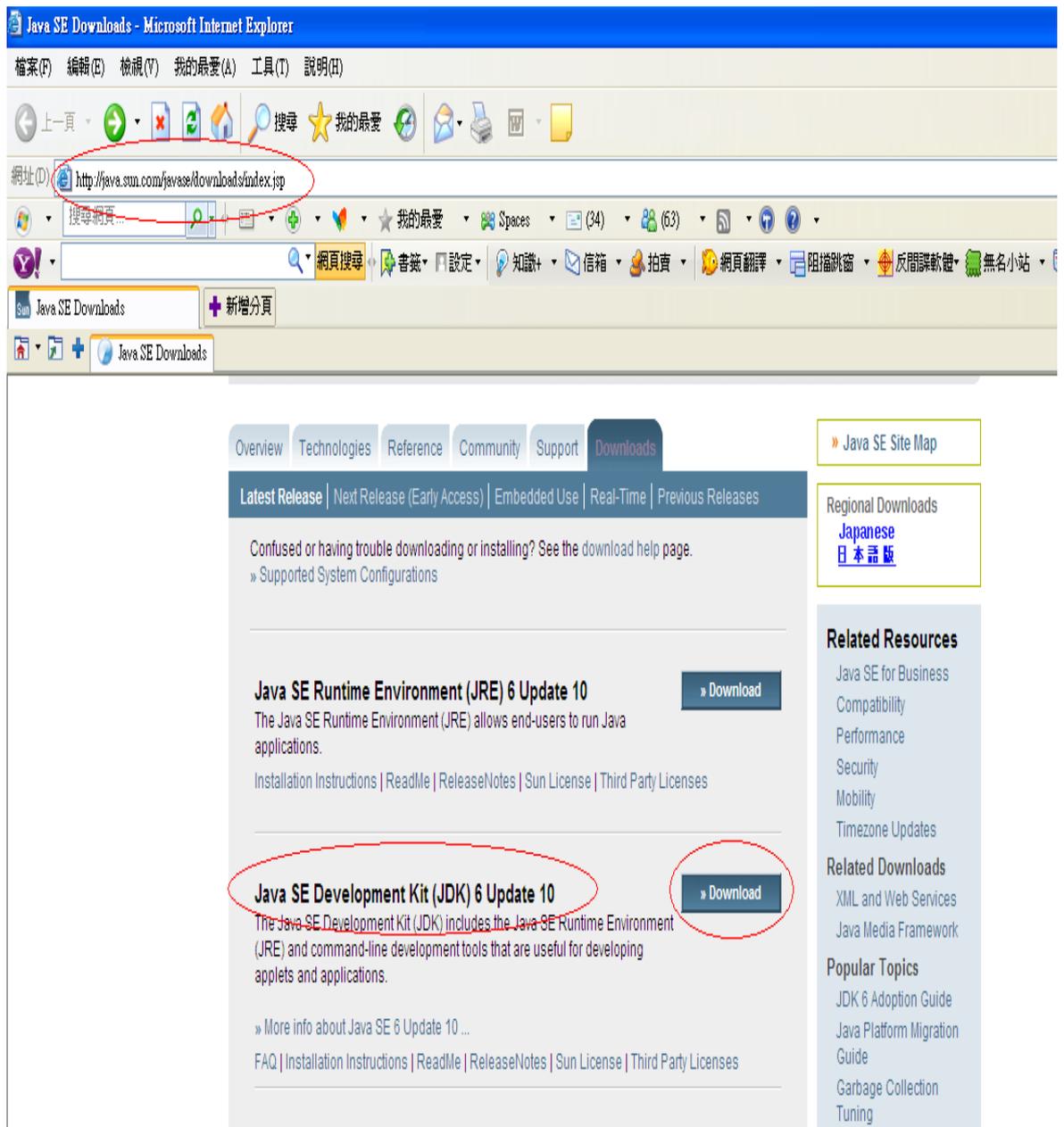


圖 3-20 JDK 安裝 1

尋找到 Java SE Development Kit (JDK) 6 Update 10 並點下載如圖 3-21 所示，畫面上會要求您選擇使用平臺以及語言，我們選擇 Windows，語言其實也只有一個選項不用理會，記得要打勾。

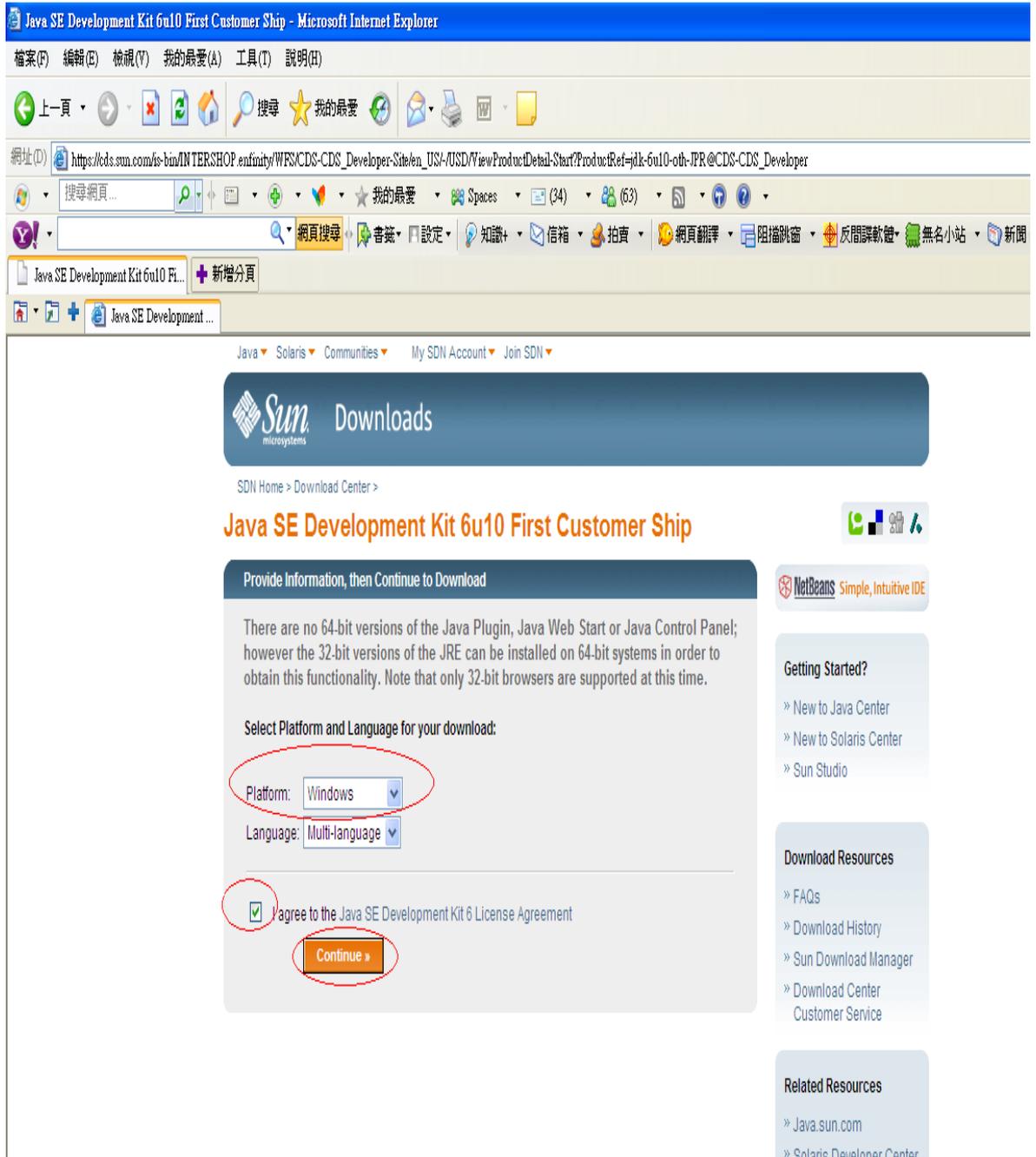


圖 3-21 JDK 安裝 2

當上述步驟選完後，就可以下載安裝檔到電腦了，如圖 3-22 所示下載完後我們就可以開始安裝 JDK。

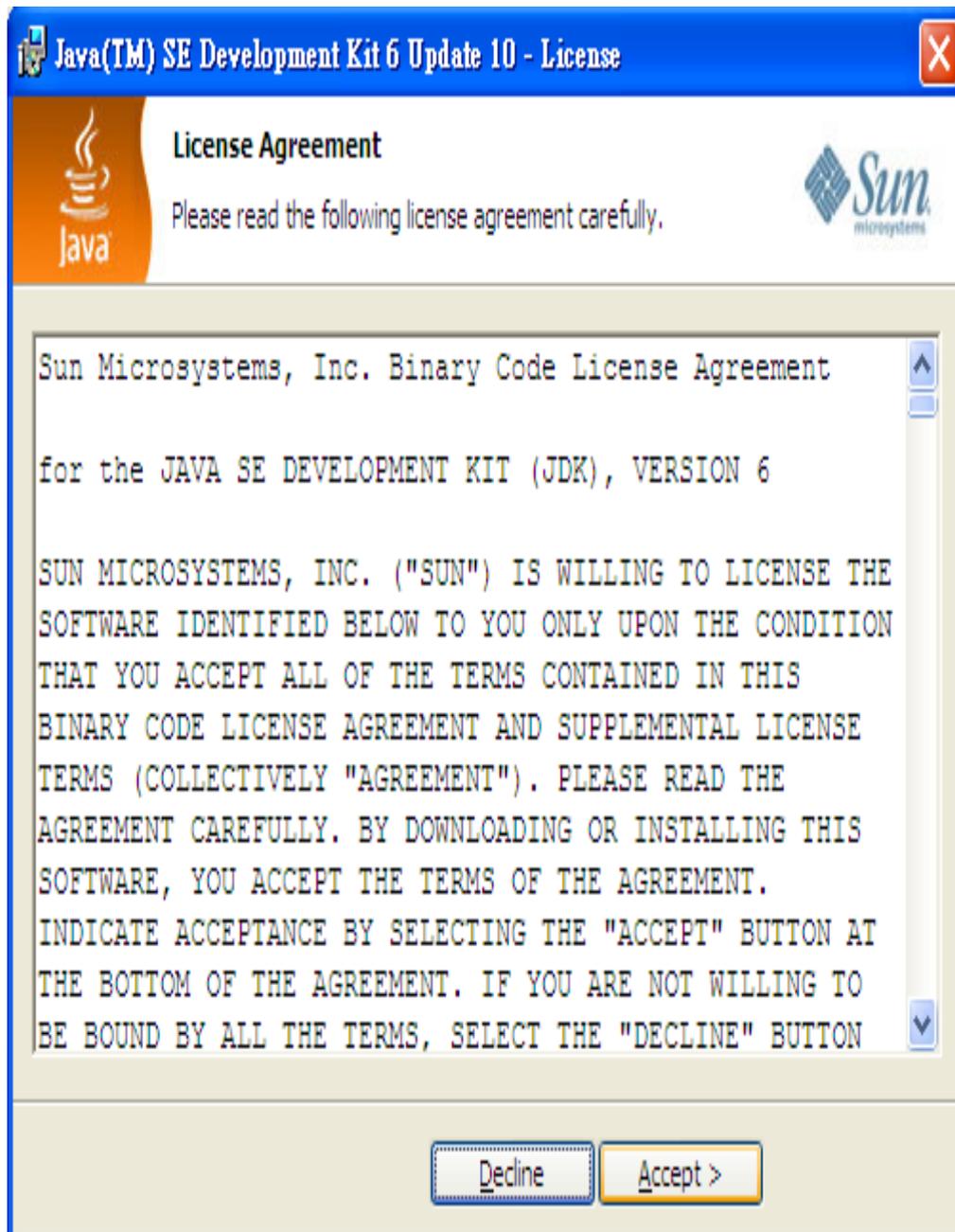


圖 3-22 JDK 安裝 3

同樣的 JDK 安裝過程中也有安裝功能選單如圖 3-23、圖 3-24 所示，可以依照使用的的需求設定，不過通常都是全部安裝。

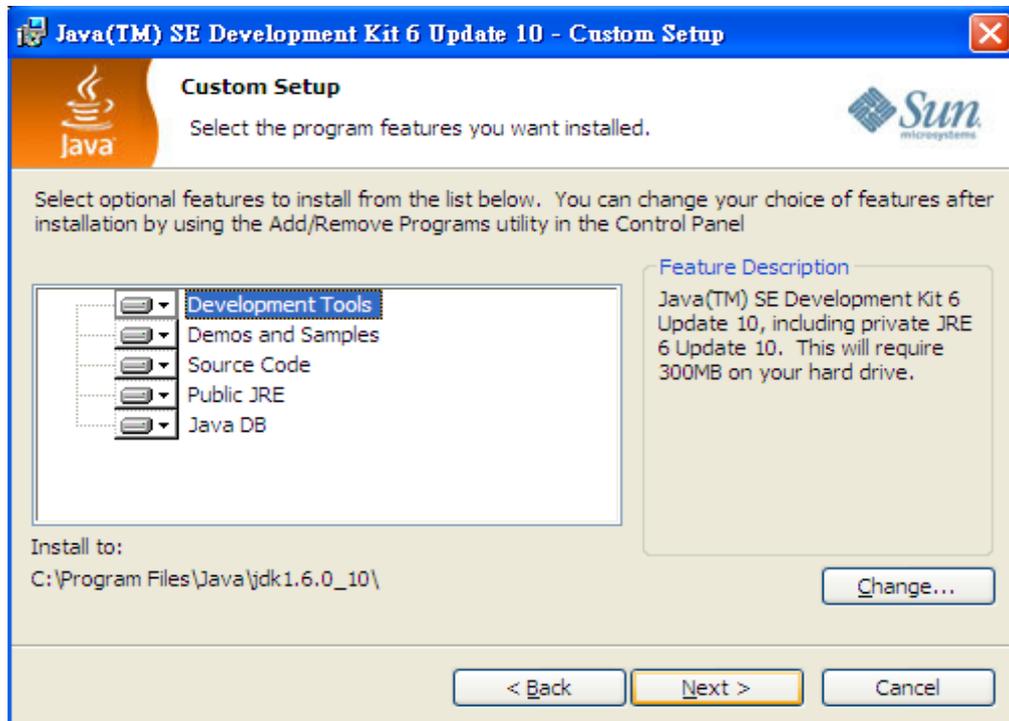


圖 3-23 JDK 安裝 4



圖 3-24 JDK 安裝 5

再來呢，我們要到「<http://tomcat.apache.org/download-60.cgi>

」tomcat 官方網站下載 tomcat6.x 如圖 3-25 所示，開啟網頁後選擇左邊的 TOMCAT6.x 版本。

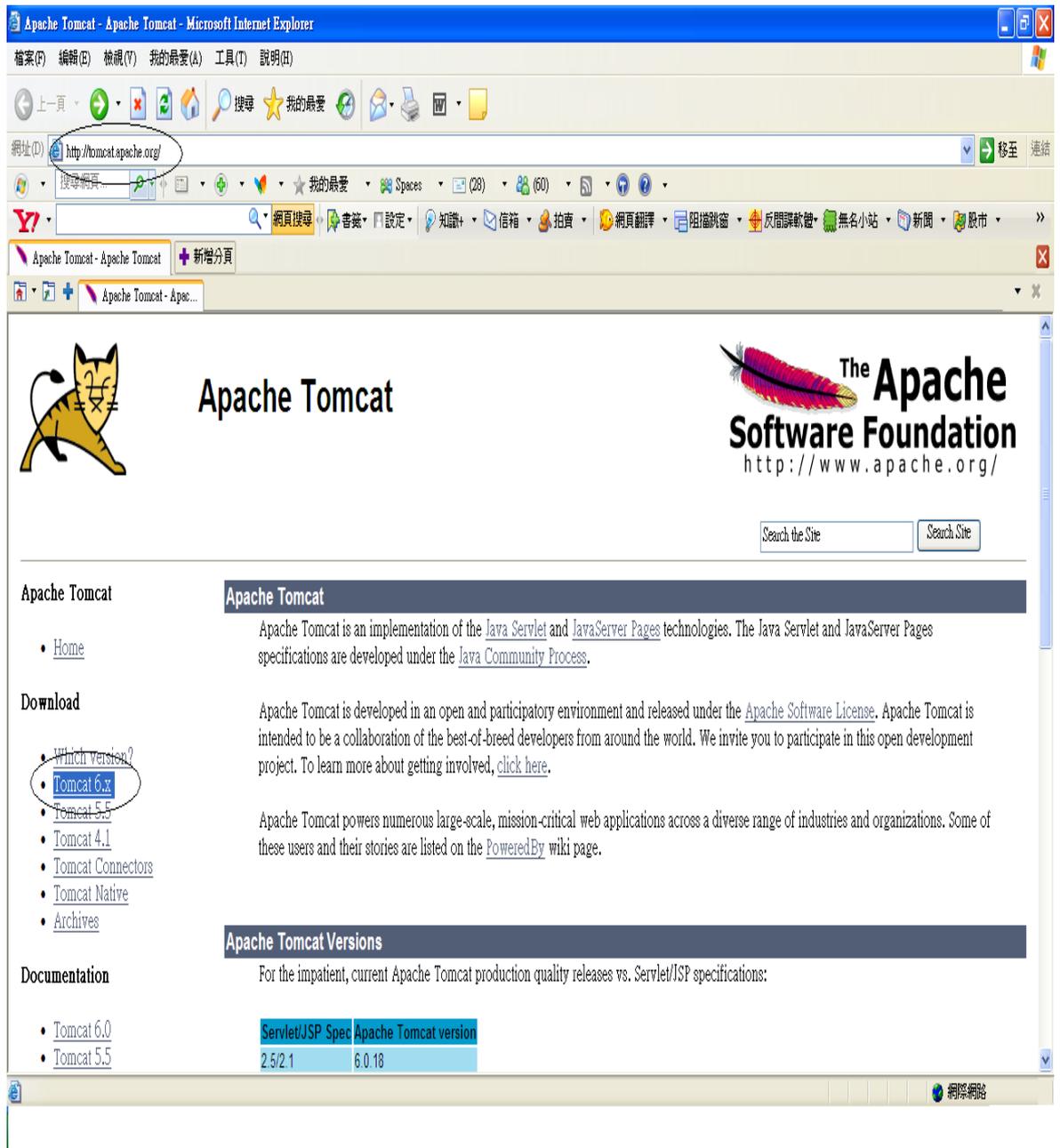


圖 3-25 TOMCAT 安裝畫面 1

進入 tomcat6.x 版本畫面後如圖 3-26、圖 3-27、圖 3-28 所示，拉置最下方 6.0.18 版本點選下載檔案到電腦並開始安裝。

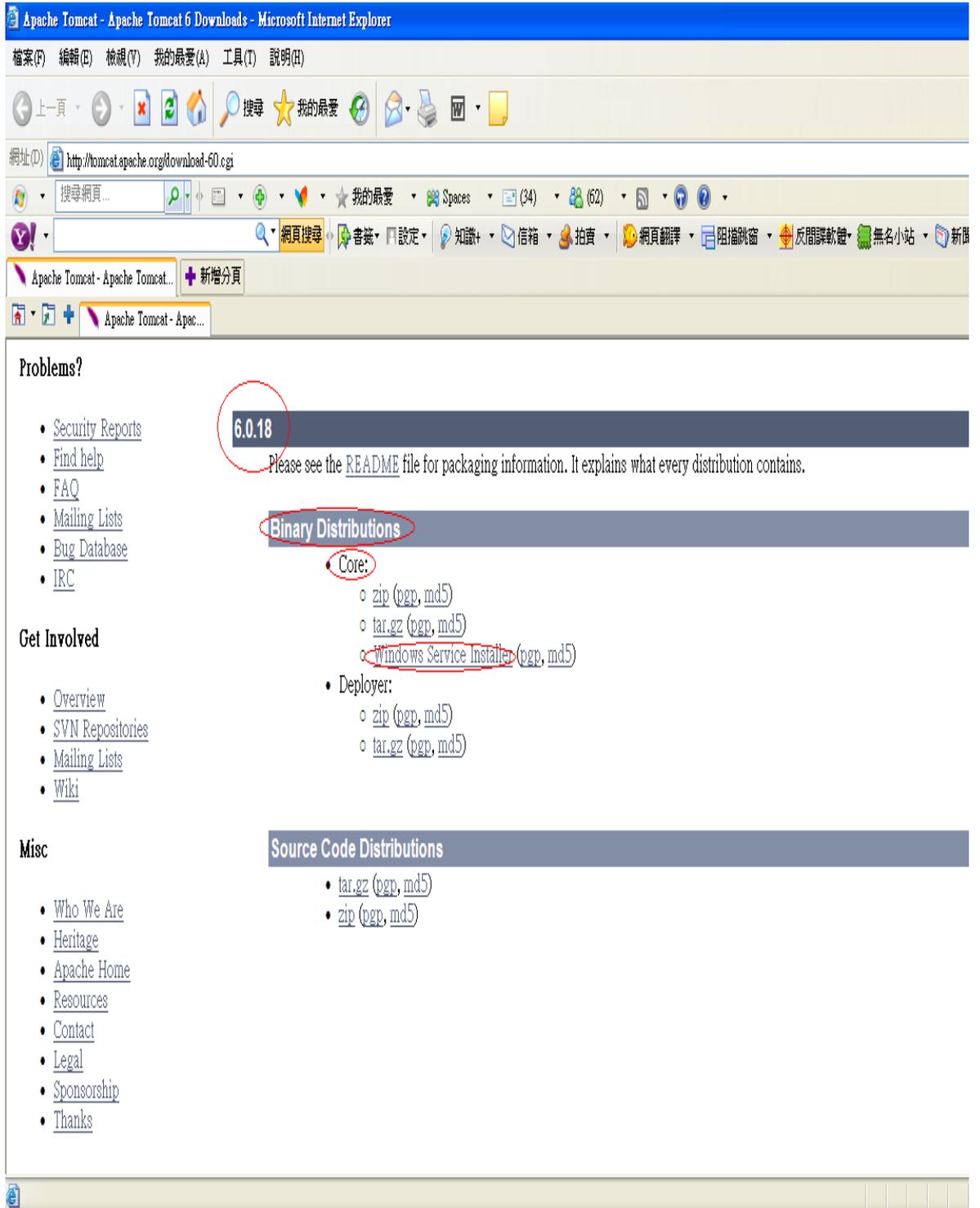


圖 3-26 TOMCAT 安裝畫面 2

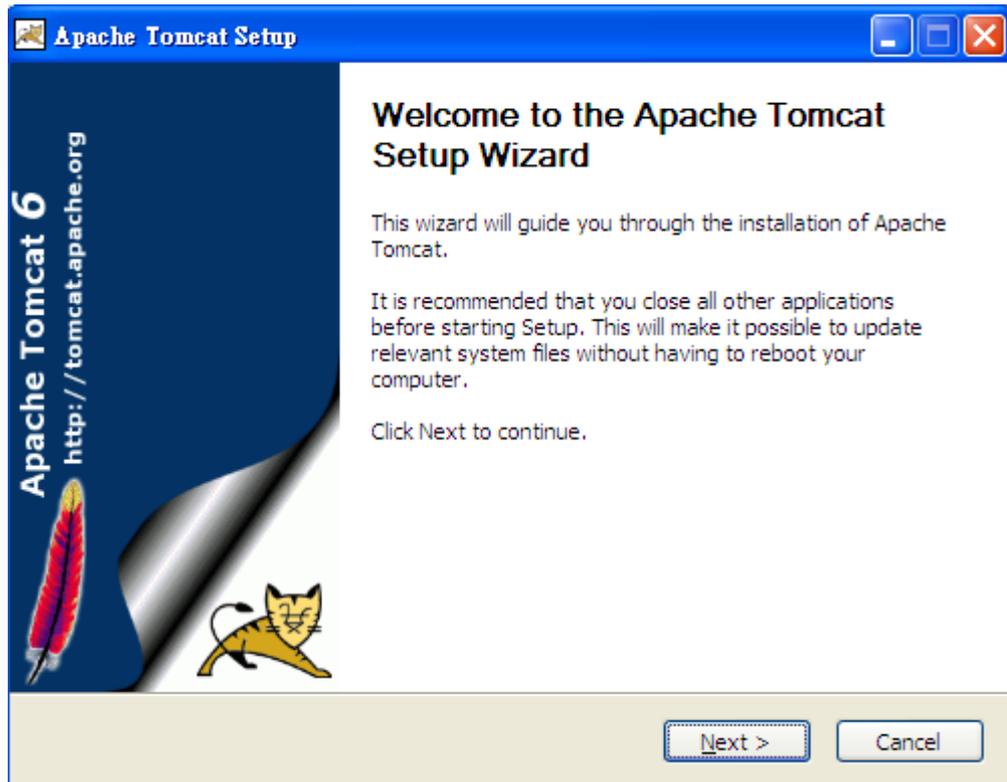


圖 3-27 TOMCAT 安裝畫面 3

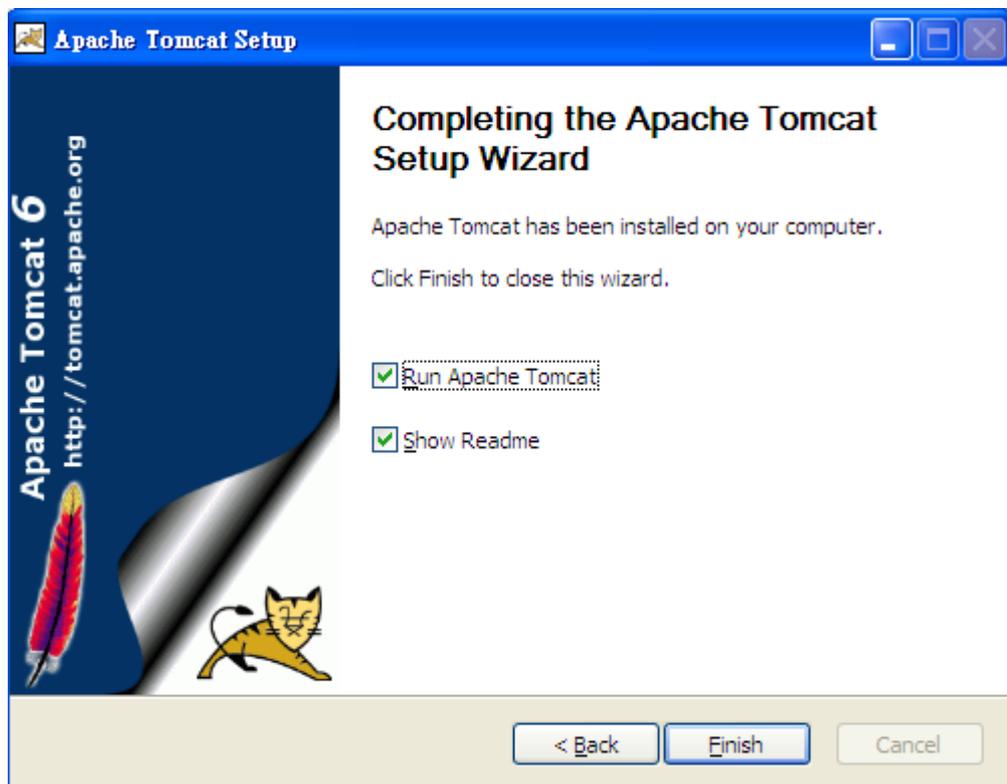


圖 3-28 TOMCAT 安裝畫面 4

第六節 系統技術與分析

在連接 SQL SERVER 2005 時，需要先下載 sqljdbc.jar，將此檔案放置在 C:\Program Files\Java\jdk1.6.0_06\lib 中，然後才能與 SQL SERVER 連線，連線之程式如下：

```
1 request.setCharacterEncoding("Big5");
2 Connection conn = null;
3 Statement smt = null;
4 Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
5 conn = DriverManager.getConnection("jdbc:sqlserver://localhost:1433;" +
6 "databaseName=work;" + "user=sa;" + "password=1234;");
7 smt = conn.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
8 ResultSet.CONCUR_READ_ONLY);
```

程式第 1 為設定中文編碼規則，第 4 為註冊驅動程式，第 5~6 為取得連線及設定連線之參數，第 7~8 為建立操作物件。建立完成後，即可執行新增、修改、刪除、查詢之功能，相關程式如下：

```
1 String sql = "insert into LE (LE_year , LE_term , LE_name , LE_Ename ,
2 LE_SN) values ('" + year + "','" + term + "','" + name + "','" + Ename +'"
3 ,'" + SN + "')";
4 x = smt.executeUpdate(sql);
5 if (x > 0) {
6 out.print("新增成功");
7 } else {
```

```
8 out.print("新增失敗"); }
```

程式 1~3 為新增資料至 SQL SERVER 資料庫之語法，第 4 為完成新增的動作，回傳值為布林值，第 5~8 確定資料是否有新增至資料庫，並顯示第 6~8 之訊息至螢幕上。

```
1 String sqla = "Select * from LE Where LE_year = '" + year + "' and
2 LE_term = '" + term + "' ";
3 ResultSet myResultSet = smt.executeQuery(sqla);
4 out.println("<table border='2' >");
5 out.println("<tr><td>編號</td><td>年度</td><td>學期序</td><td>
6 課程名稱</td><td>英文課名</td></tr>");
7 int y = 1;
8 while (myResultSet.next()) {
9 out.print("<tr><td>" + y + "</td><td>" +
10 myResultSet.getString("LE_year") + "</td><td>" +
11 myResultSet.getString("LE_term") + "</td><td>" +
12 myResultSet.getString("LE_name") + "</td><td>" +
13 myResultSet.getString("LE_Ename") + "</td></tr>");
14 y++; }
15 out.println("</table>");
```

程式 1~2 為查詢資料庫之語法，第 3 為將查詢出來的資料紀錄在變數中，第 4~15 為將要查詢的資料，用表格的方式呈現在螢幕上。

在網頁中，我們需要儲存某些使用者輸入的資料，在其他頁面當做查詢的條件，所以要使用 session 的技術，session 可以儲存需要記錄的資

料，且 session 的生命週期為開啟網頁後，直到關閉此網頁，session 都會記錄需要的資料，程式碼如下：

```
1 session.setAttribute("year", year);
2 session.getAttribute("year")
```

程式第 1 為設定要儲存什麼資料到 session 的變數中，第 2 為將 session 變數中的值取出來使用。

在此系統中，登錄平時成績或是期中期末成績，因為同時有許多學生的成績需要登錄，所以就需把該學生之成績和學號，將兩者的關係建立起來，然後在一次將所有同學的成績寫入資料庫中，相關程式碼如下：

```
1 <input type='text' name='score' size='5'>
2 <input type='hidden' name='stid'
3 value="+myResultSet.getString("ST_id")+ " />
```

程式第 1 行是讓使用者輸入的欄位，第 2~3 為紀錄該學生之學號，學號的欄位型態為隱藏欄位。

利用下列的程式碼，將學號和成績取出來，並寫入到資料庫中，程式碼如下：

```
1 String[] str = request.getParameterValues("score");
2 String[] stid = request.getParameterValues("stid");
3 for (int i = 0; i < str.length; i++) {
4 String sql = "insert into SCORE (ST_id,LE_SN,score,status) values('" +
```

```
5  std[i] + 5  "," + session.getAttribute("SN") + "," + str[i] + "," + a +  
6  ""); }
```

程式第 1~2 是將學生的學號和成績，用字串陣列來記錄，程式第 3~6 是利用 FOR 迴圈，逐一將兩個陣列的值寫入至資料庫中。

為了防止教師少登錄學生的成績，在寫入資料庫前，會先做個判斷，只要少登錄任何一個學生的成績，就不會將所有的成績進行儲存的動作，相關程式碼如下：

```
1  for (int i = 0; i < str.length; i++) {  
2  if (str[i] == "") {  
3  out.print("尚有成績未登錄!!，請按上一頁返回並輸入完整的成績。");  
4  break; }  
5  }
```

程式 1~5 為一個 FOR 迴圈，去讀取陣列裡面的值，並且經由程式第 2~3，去判斷陣列裡面是否有空值，如果有的話，就會顯示程式第 3 行的訊息，且程式也會中段執行。

學生在選課的時候，會先顯示當年度老師開的課程，然後學生去點選有修哪些科目，相關程式碼如下：

```
1  Calendar cal = Calendar.getInstance();  
2  int y = cal.get(Calendar.YEAR);  
3  int m = cal.get(Calendar.MONTH) + 1;  
4  y = y - 1911;  
5  if(m > 8){
```

```

6   m = 1;
7   }else
8   m = 2;
9   String sqla = "Select * from LE where LE_year='"+y+"'
10  and LE_term='"+m+"'";
11  ResultSet rs=smt.executeQuery(sqla);
12  out.println("<table border='2' name='test'>");
13  out.println("<tr><td></td><td>年度</td><td>學期序
14  </td><td>課程名稱</td><td>英文課名</td></tr>");
15  int x = 1;
16  while(rs.next()){
17  out.print("<tr><td>"+<input type='checkbox' name='check' value='"+
18  rs.getString("LE_SN") +"'>"+</td><td>"+rs.getString("LE_year")+
19  "</td><td>"+rs.getString("LE_term")+</td><td>"+
20  rs.getString("LE_name")+</td><td>"+rs.getString("LE_Ename")+
21  </td></tr>");
22  x++;
23  }
24  out.println("</table>");

```

程式第 1~3 為取得目前系統時間，取得年和月，程式第 4~8 是將西元年轉成民國年，並去判斷是上學期還是下學期，程式第 9~12 是抓取資料庫中相關的資料，並透過程式第 13~24 將抓取出來的資料，用表格的方式呈現在網頁上。

第四章 系統呈現

第一節 起始畫面及註冊

登入系統首頁之畫面及功能簡介如圖 4-1 所示：1.最新訊息；2.學生註冊及登入；3.老師登入選單；4.本系統簡單使用說明書；5.學校的一些常用超連結。

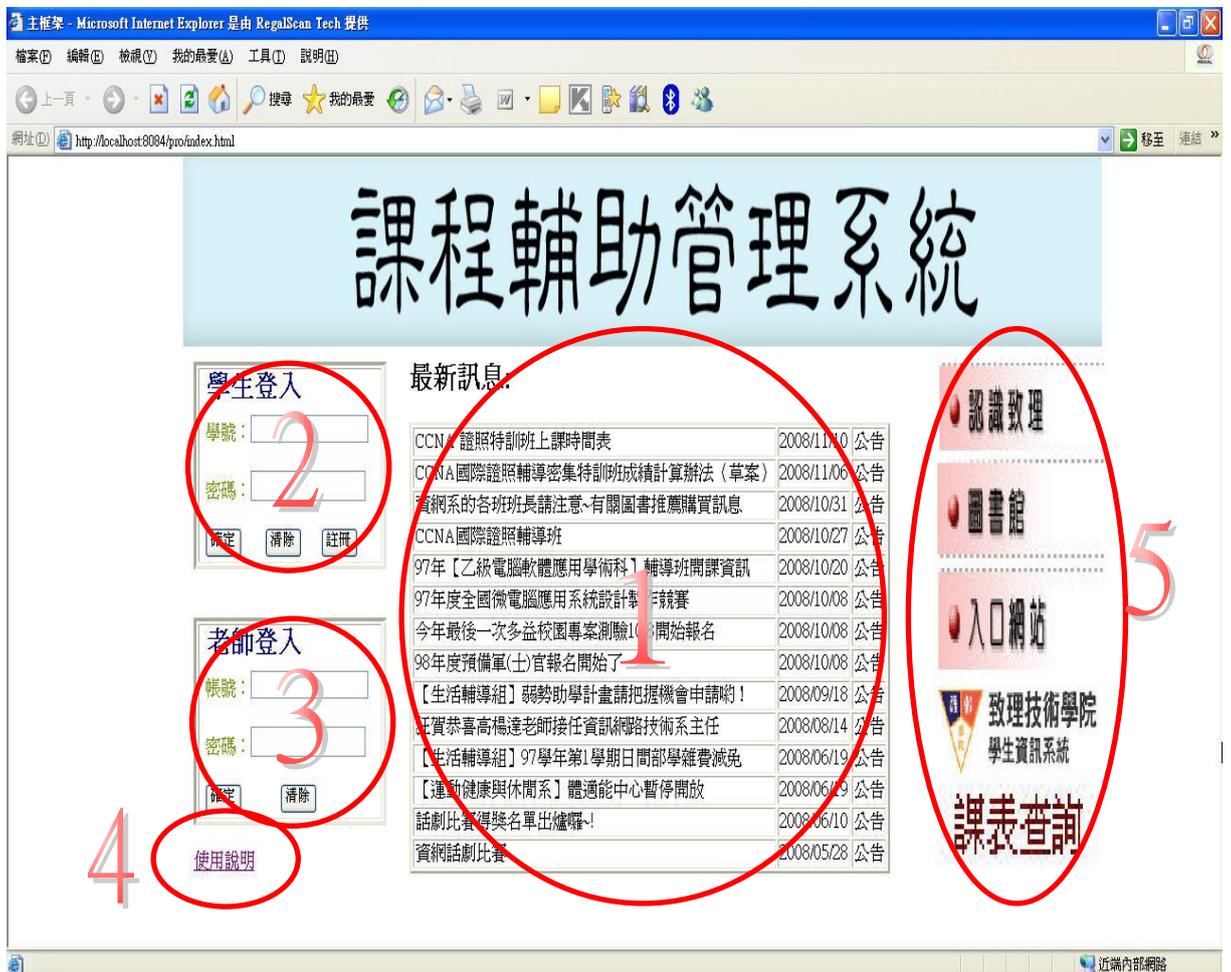


圖 4-1 管理系統首頁

系統首頁中間的最新訊息，是教師想要透露給學生們的資訊，如下圖 4-2 所示，教師可在「C:\」下的「news.txt」叫出此檔案修改。

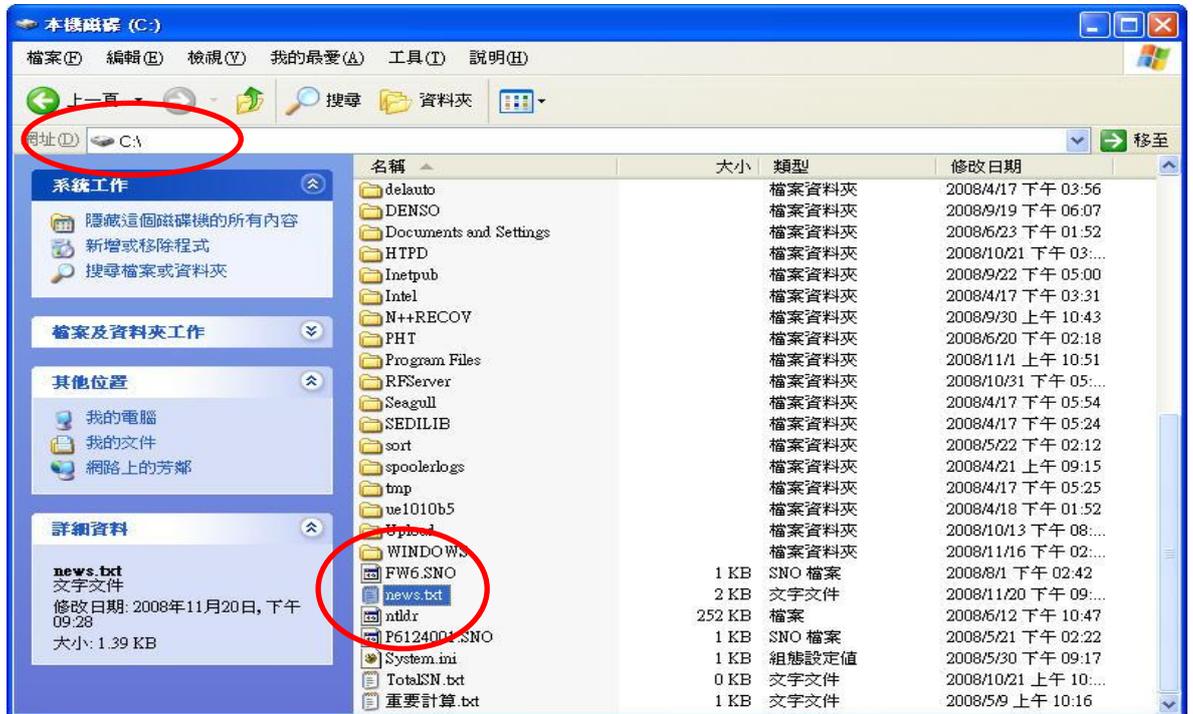


圖 4-2 最新訊息存放地點

設定檔「news.txt」如圖 4-3 所示，直接修改：

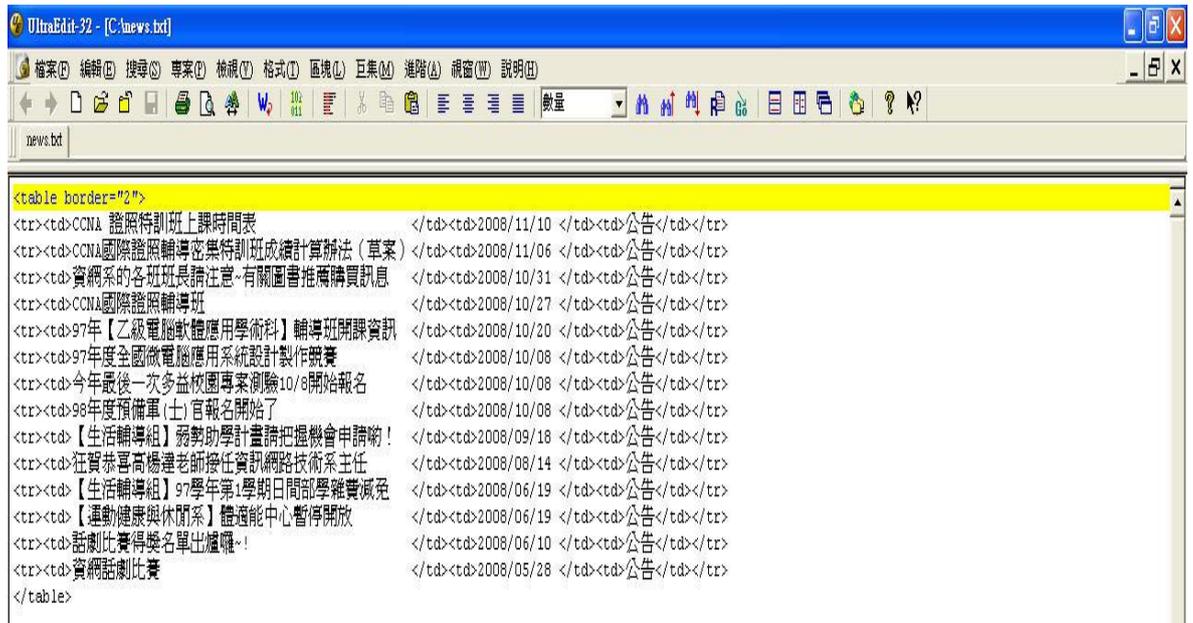


圖 4-3 最新訊息修改檔

由於教師的帳號跟密碼採用安裝時就先給予設定，所以只需要學生執行註冊的動作如圖 4-4 所示，教師並不需要。學生第一次登錄起始畫面後點選左邊學生登入欄內的「註冊」選單便可註冊帳號與密碼。



圖 4-4 學生註冊畫面

點選註冊選單後，就會跳到註冊畫面如圖 4-5 所示，請學生輸入學號、姓名、密碼及信箱，本系統的帳號即為學生的學號，密碼無限制，但密碼與密碼確認兩欄必須輸入一樣，否則會出現註冊錯誤之畫面，資料輸入完畢後點選確定即可。

The screenshot shows a Microsoft Internet Explorer browser window displaying a registration form. The browser's address bar shows the URL 'http://localhost:8084/pro/login.html'. The form is titled '註冊' and contains the following fields and buttons:

- 學號: [input field]
- 姓名: [input field]
- 密碼: [input field]
- 密碼確認: [input field]
- E-Mail: [input field]
- Buttons: 確定, 清除, 關閉視窗

The entire registration form is circled in red.

圖 4-5 學生註冊輸入資料畫面

第二節 學生端使用操作

註冊完畢後，等待教師審核通過後，即可在啟始畫面如圖 4-6 所示，輸入帳號（即學號）與密碼後按下確定後登入使用。



圖 4-6 學生登入畫面

學生登入成功後如圖 4-7 所示，即可看到學生所可以使用的選單畫面、學生操作首頁畫面。

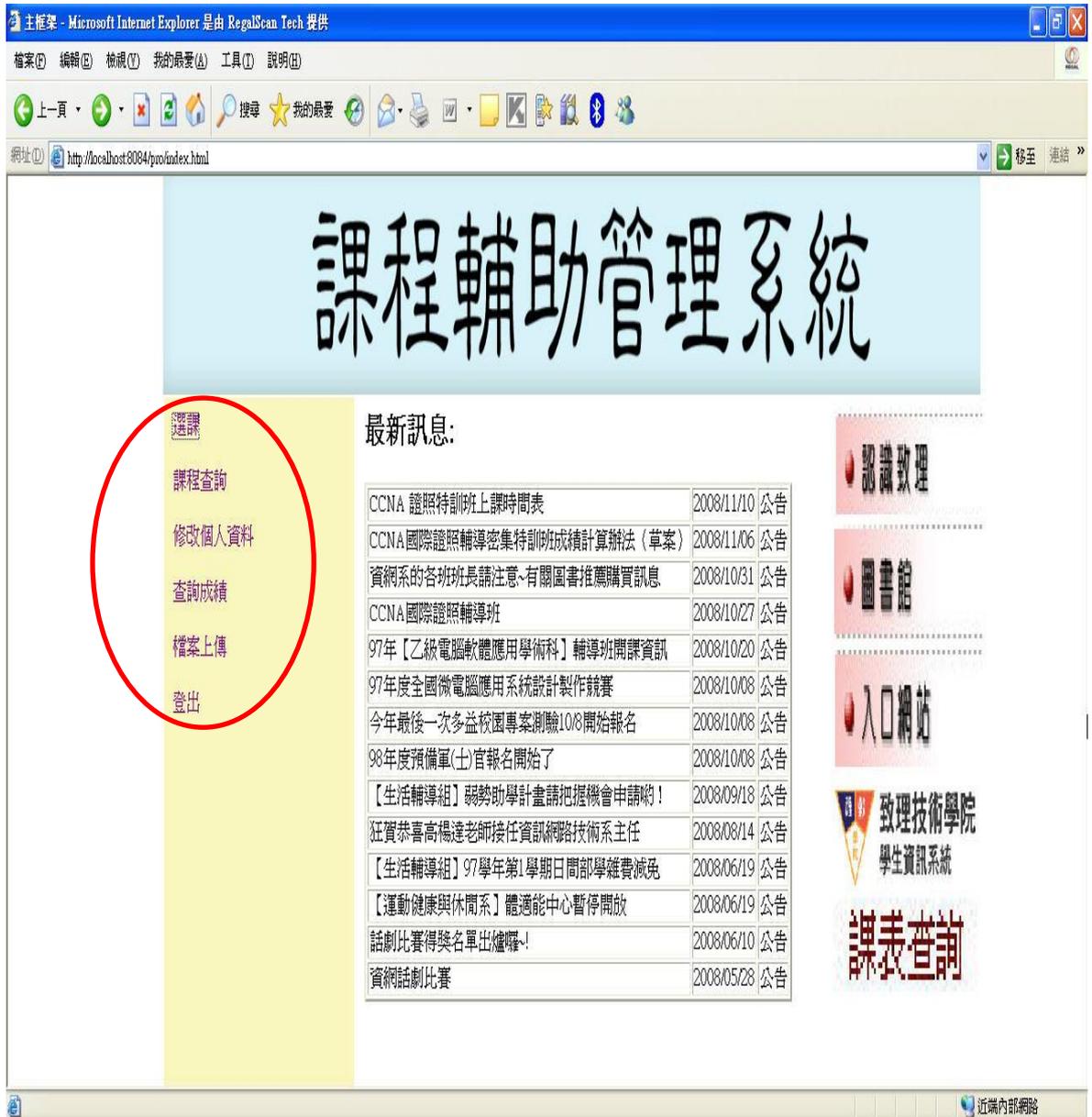


圖 4-7 學生管理首頁

學生可藉由左邊選單來進行選課、課程查詢、修改個人資料密碼、查詢成績甚至作業上傳。

當然學生一登入後最重要的事情就是要先選課如圖 4-8 所示，所以要先點選左邊選單的「選課」進入此畫面後，就會出現該教師這一學期所開課的選項，學生們要勾選自己所修的課程，按下確定後會出現新增成功之畫面，教師端就會有開學生的資料。

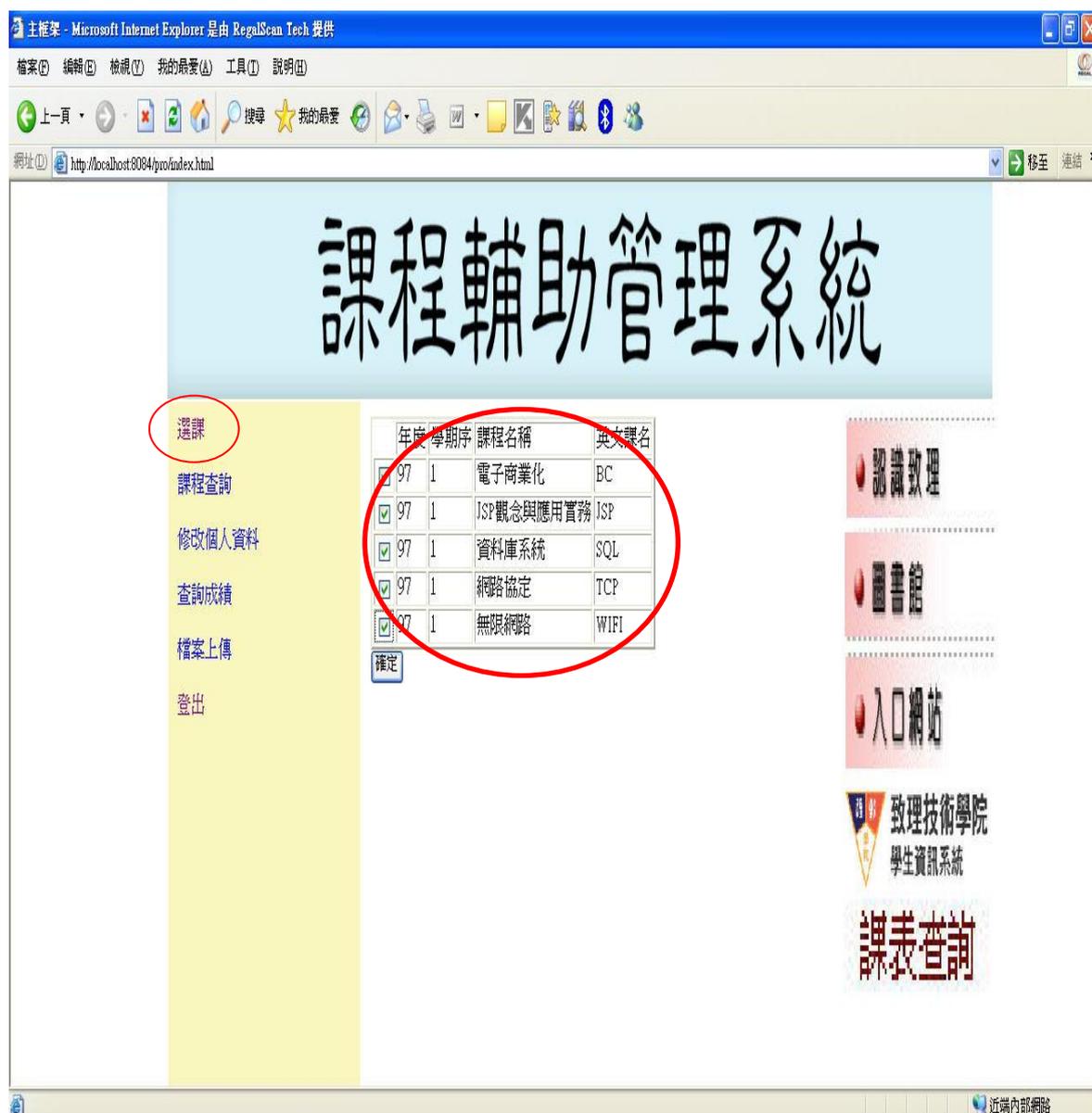


圖 4-8 學生選課畫面

當然有選課畫面就會有課程查詢畫面，學生們可以利用左邊選項點選「課程查詢」如圖 4-9 所示，即可知道自己選修該教師哪些課程，如果有選錯，也可以把上通知教師將學生資料刪除，在重新選課。

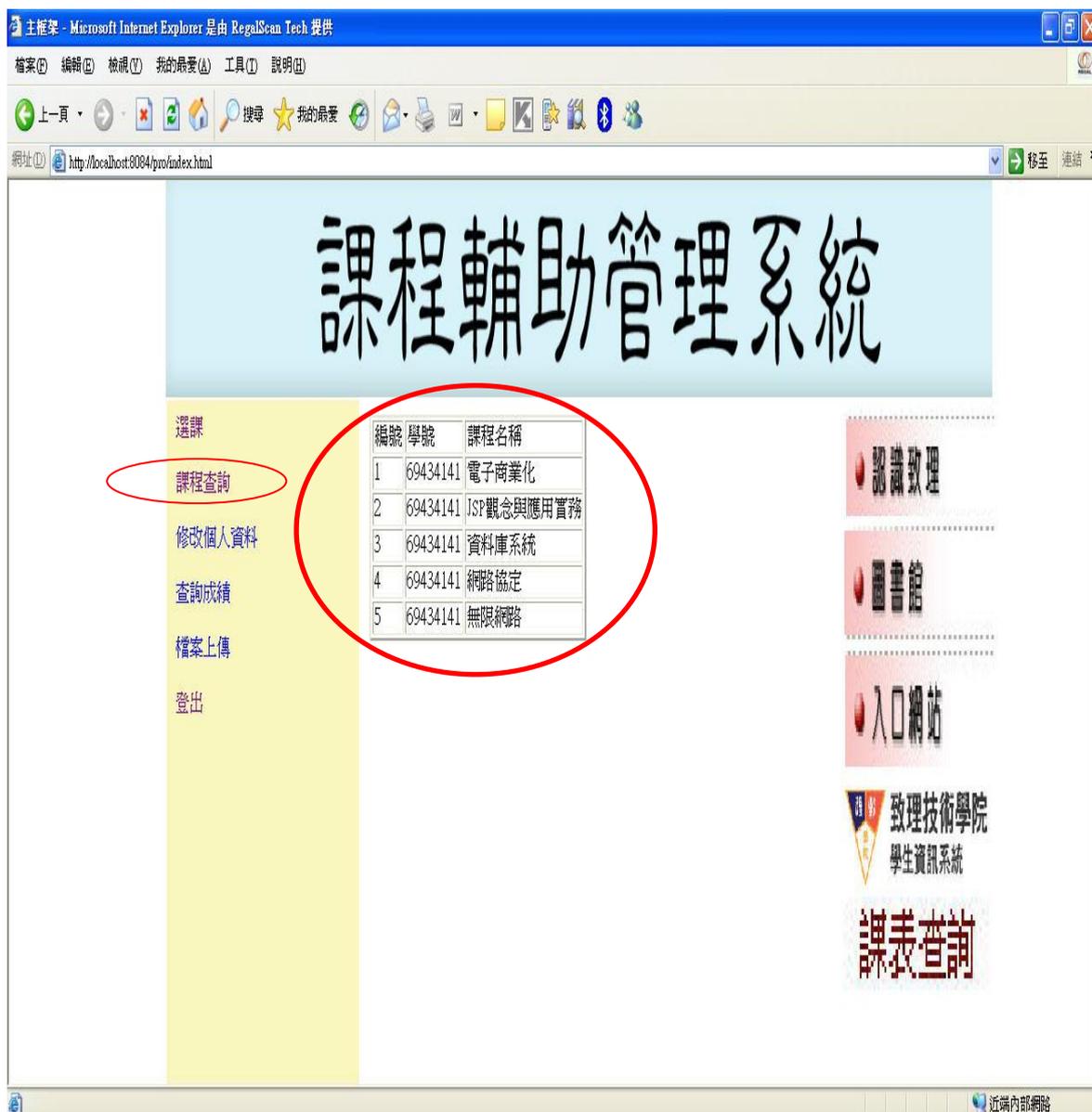


圖 4-9 學生課程查詢畫面

如果學生想要修改密碼或是其他個人資料如圖 4-10 所示，可以點選左邊選項的「修改個人資料」來修改個人資料。

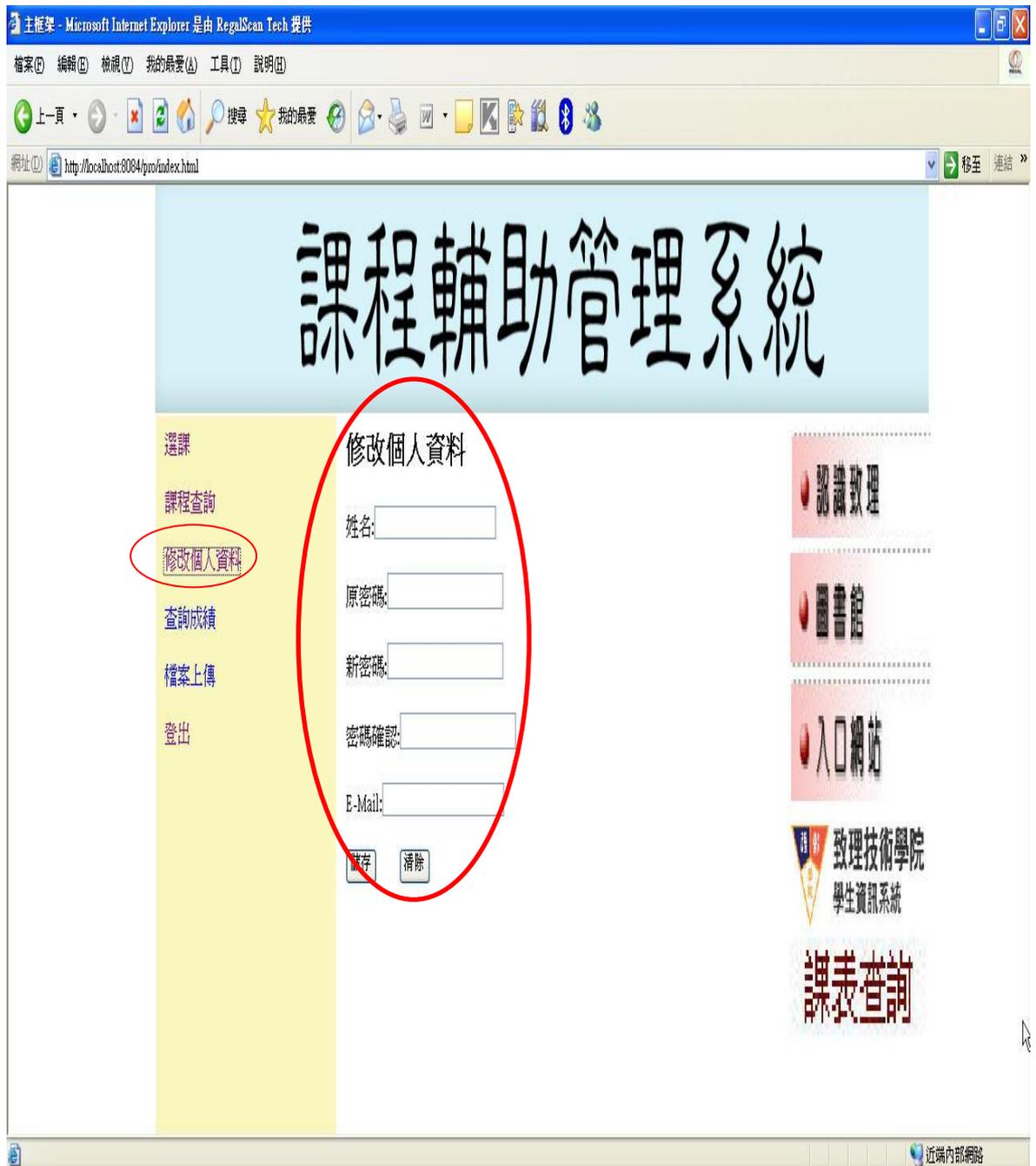


圖 4-10 學生修改個人資料

學生只要在修改資料輸入想要修改的資料按下儲存，就會自動把新的資訊覆蓋舊的資料。

一般大專院校成績都有平時成績、期中考成績及期末考成績等，所以學生們想要查詢教師的評如圖 4-11 所示，可以在左邊欄位點選「成績查詢」進行查詢之動作。

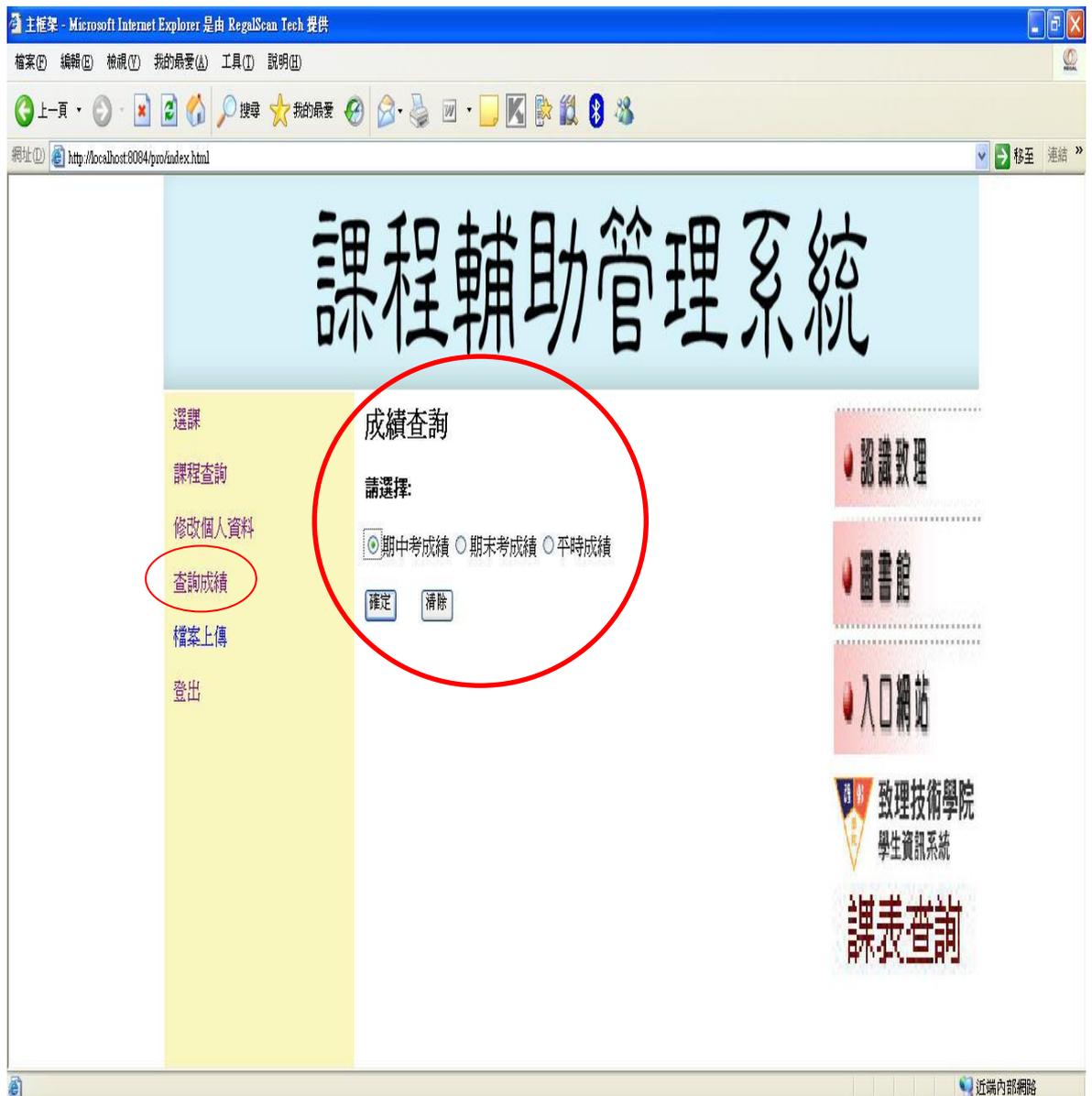


圖 4-11 學生成績查詢

點選「查詢成績」後，中間畫面就會出現平時成績、期中考成績及期末考成績等選單提供選擇，看學生們想查詢哪一個，在點選其選項後按下確定即可。

由於可能學生不只有這位老師一堂課程，所以點選想要查詢的成績後如圖 4-12 所示，還是會出現學生所修的課程選單請學生點選，(我們以查詢期中考成績為例)。

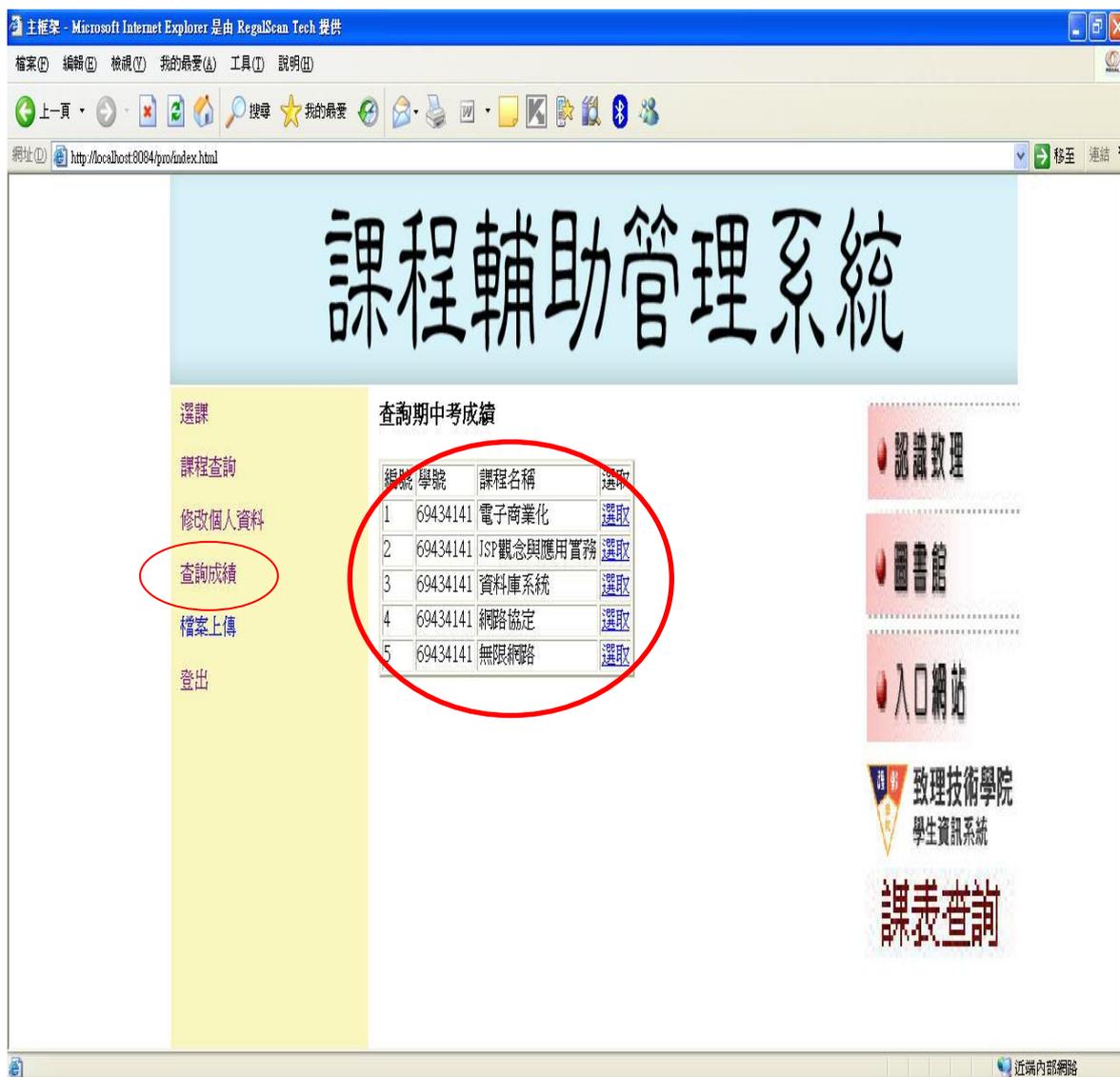


圖 4-12 學生查詢成績期中考課程選單

學生選了 5 堂該教師的課程，所以有 5 個選單。點選要查看該課程的選取後，如圖 4-13 所示，就會出現該科目的期中考成績了。(我們以查詢網路協定的期中考成績為例)。

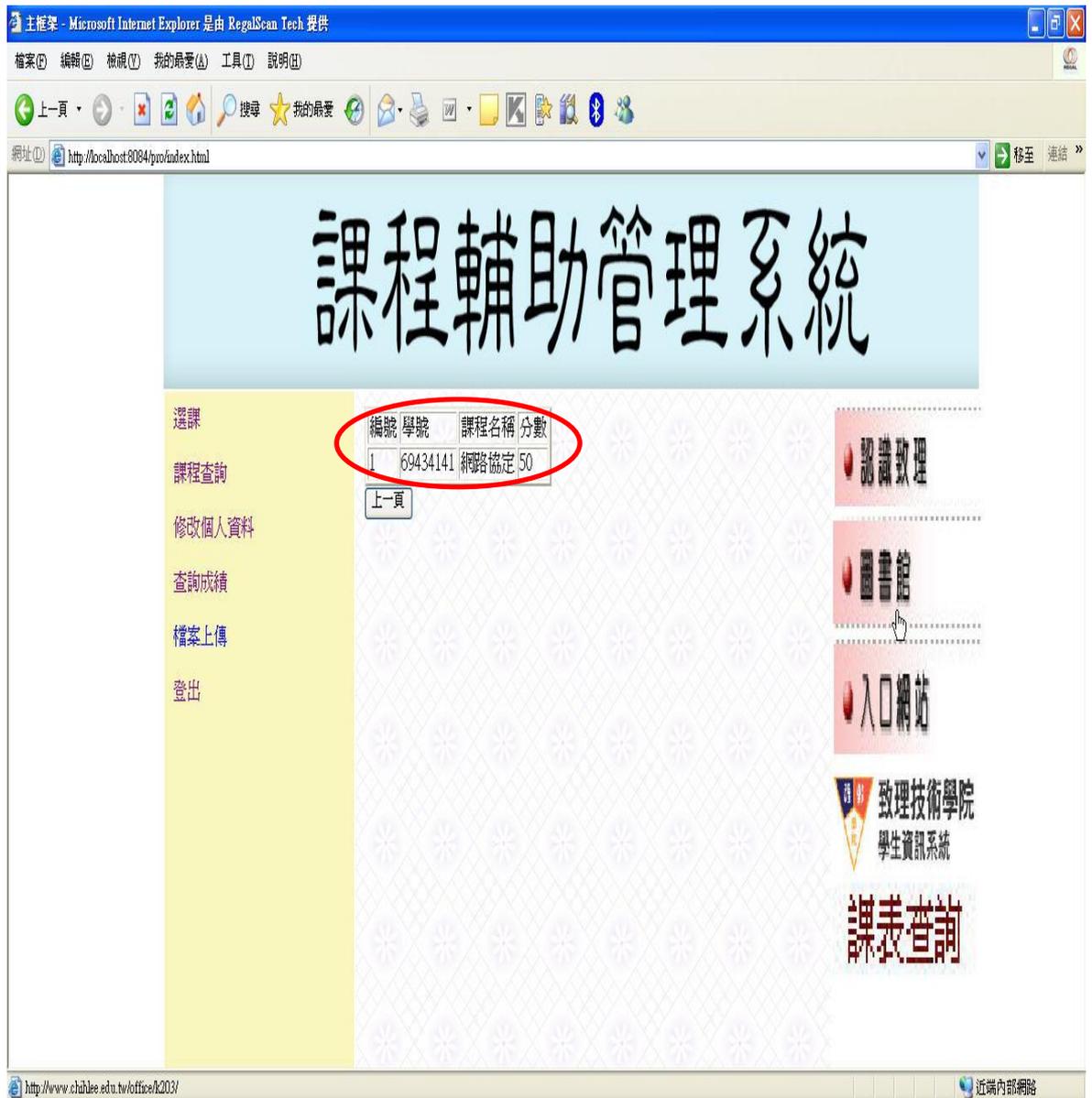


圖 4-13 學生查詢課程期中考成績

當然可以在利用「上一頁」之按鍵繼續查詢其他課程之期中成績。

除此之外上傳作業的功能也是必須考慮的功能，有了上傳作業的功能，學生們就可以在上課時間點選左邊選項的「上傳作業」功能將作業傳送到教師的電腦如圖 4-14 所示，一樣也要先選擇此作業是哪一科目，才不會把作業傳錯課程。

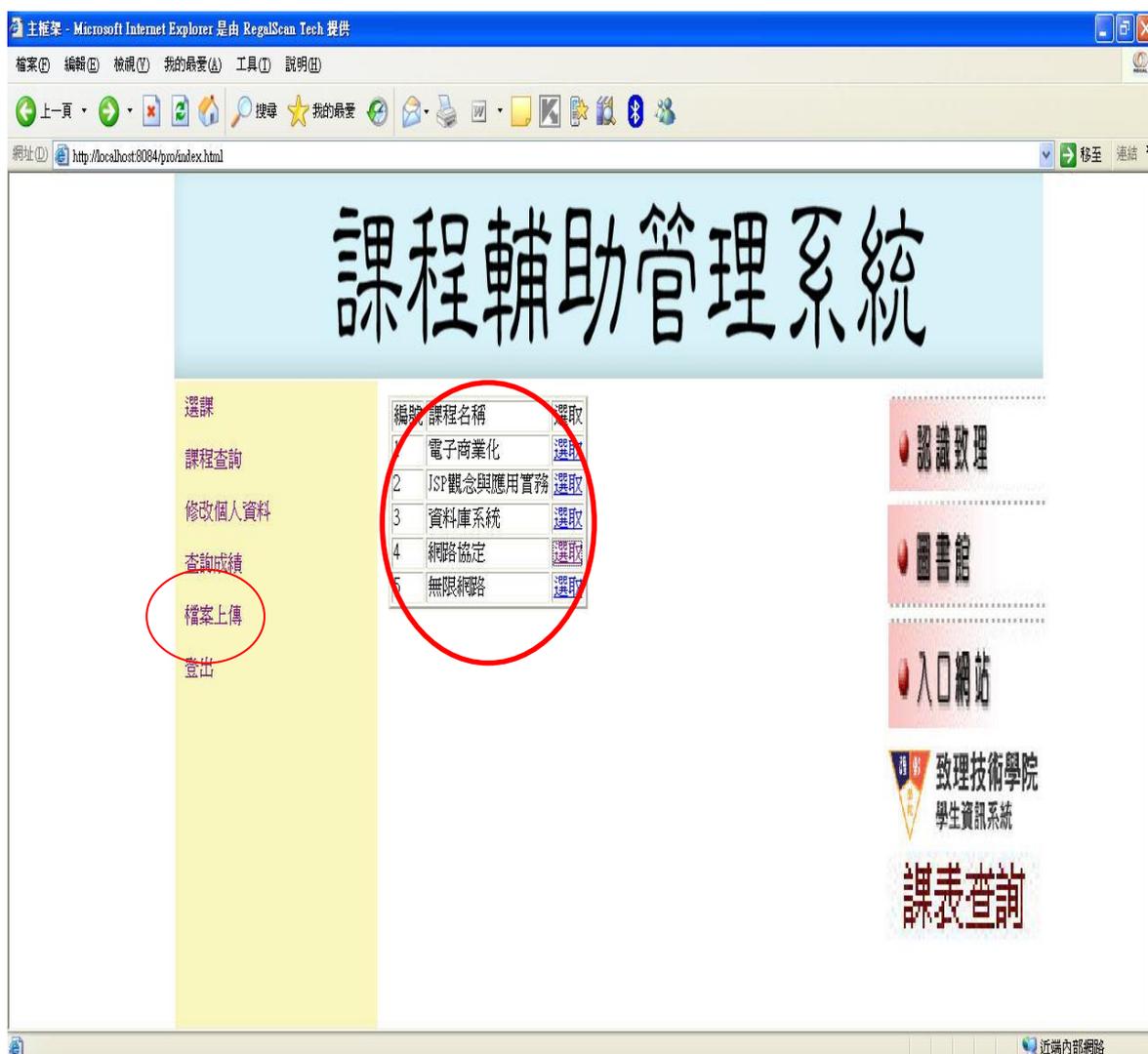


圖 4-14 學生上傳作業課程選單

選好課程之後，畫面就會出現請學生們瀏覽作業資料存放地點如圖 4-15 所示，將作業路徑尋找出來後，即可上傳作業到老師的電腦內。

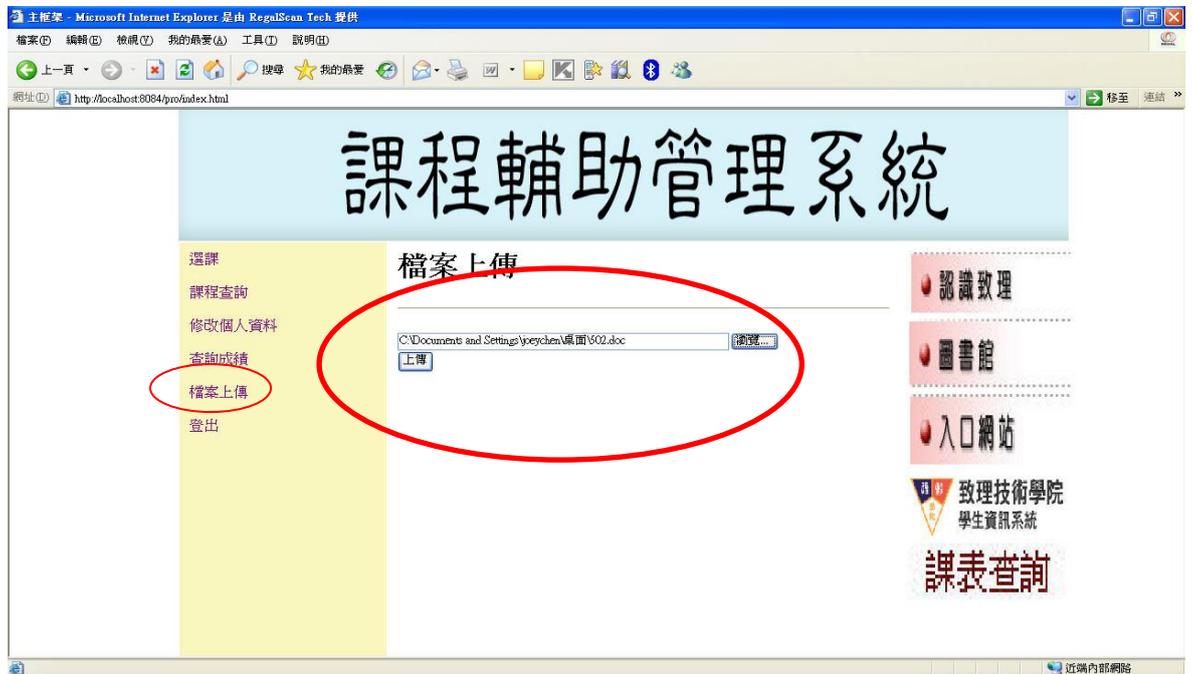


圖 4-15 學生上傳作業

作業檔案上傳成功後如圖 4-16 所示，就會出現上傳成功的字樣畫面。

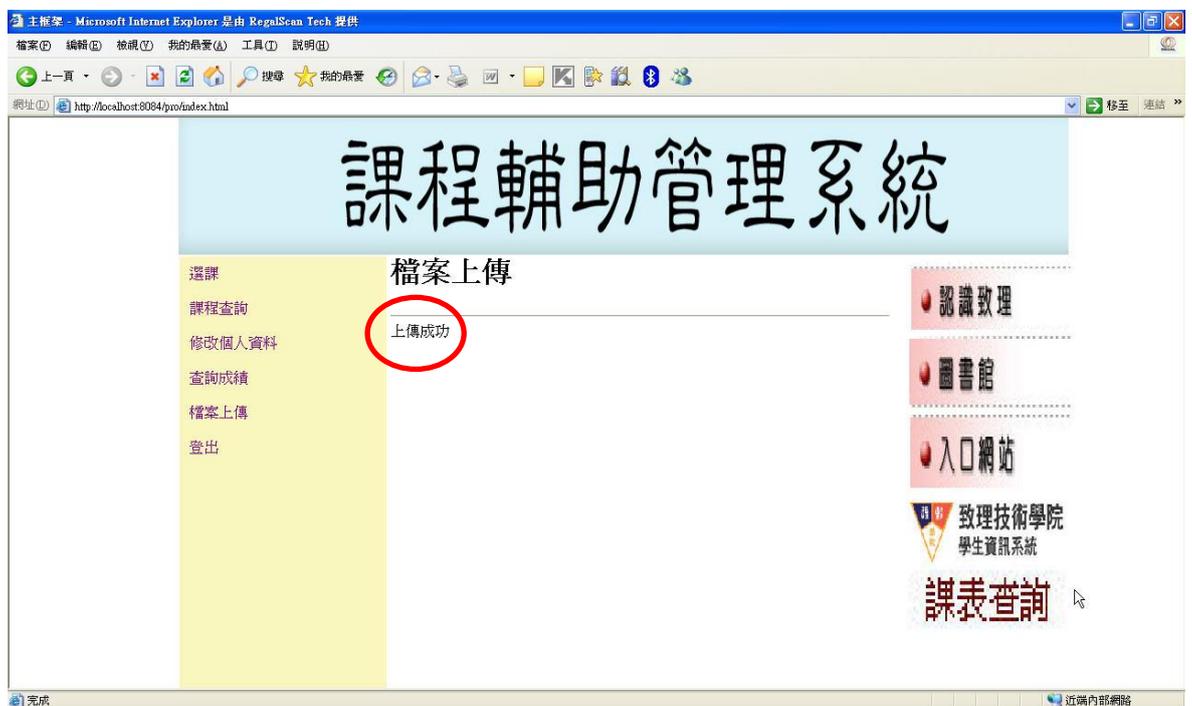


圖 4-16 學生上傳作業

第三節 教師端使用操作

教師的帳號密碼是一開始安裝時就先預設好的（所以教師們一開始就要想好帳號密碼給安裝人員設定），進入畫面如圖 4-17 所示，教師只要直接登入即可。



圖 4-17 教師登入畫面

登入後左邊選單如圖 4-18 所示，就是教師可以使用的管理的選單。

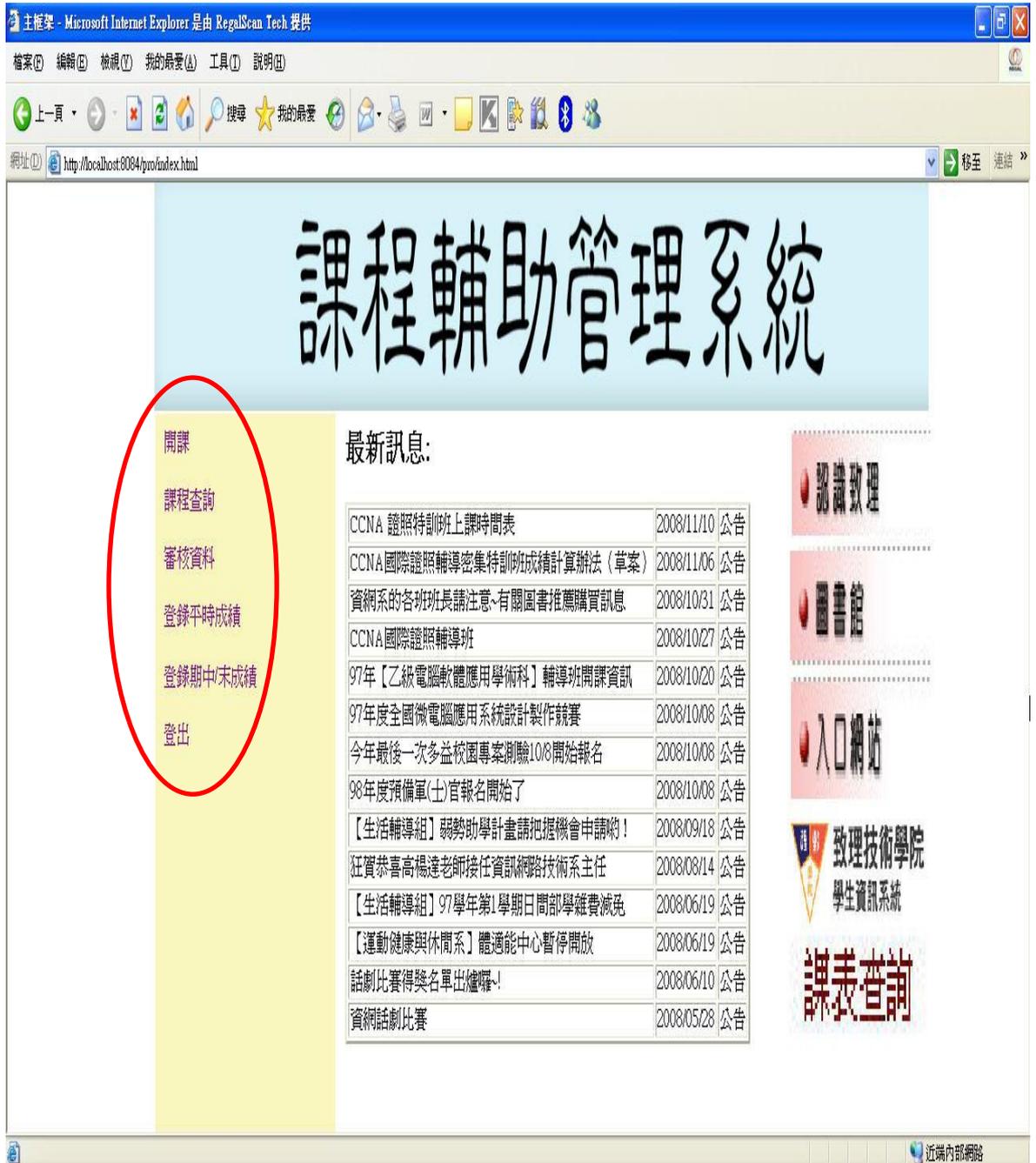


圖 4-18 教師登入後的畫面

教師端登入後，左邊選單可以使用開課、課程查詢、審核資料、登錄平時成績及登錄期中期末考成績等功能。

當然教師登入第一件事情就是開課，點選左邊選單的「開課」選項進行開課程序如圖 4-19 所示，當新增開課資訊完成後，按下確定就開課成功了，而下方也會顯示出教師這一學期所開過的課程。

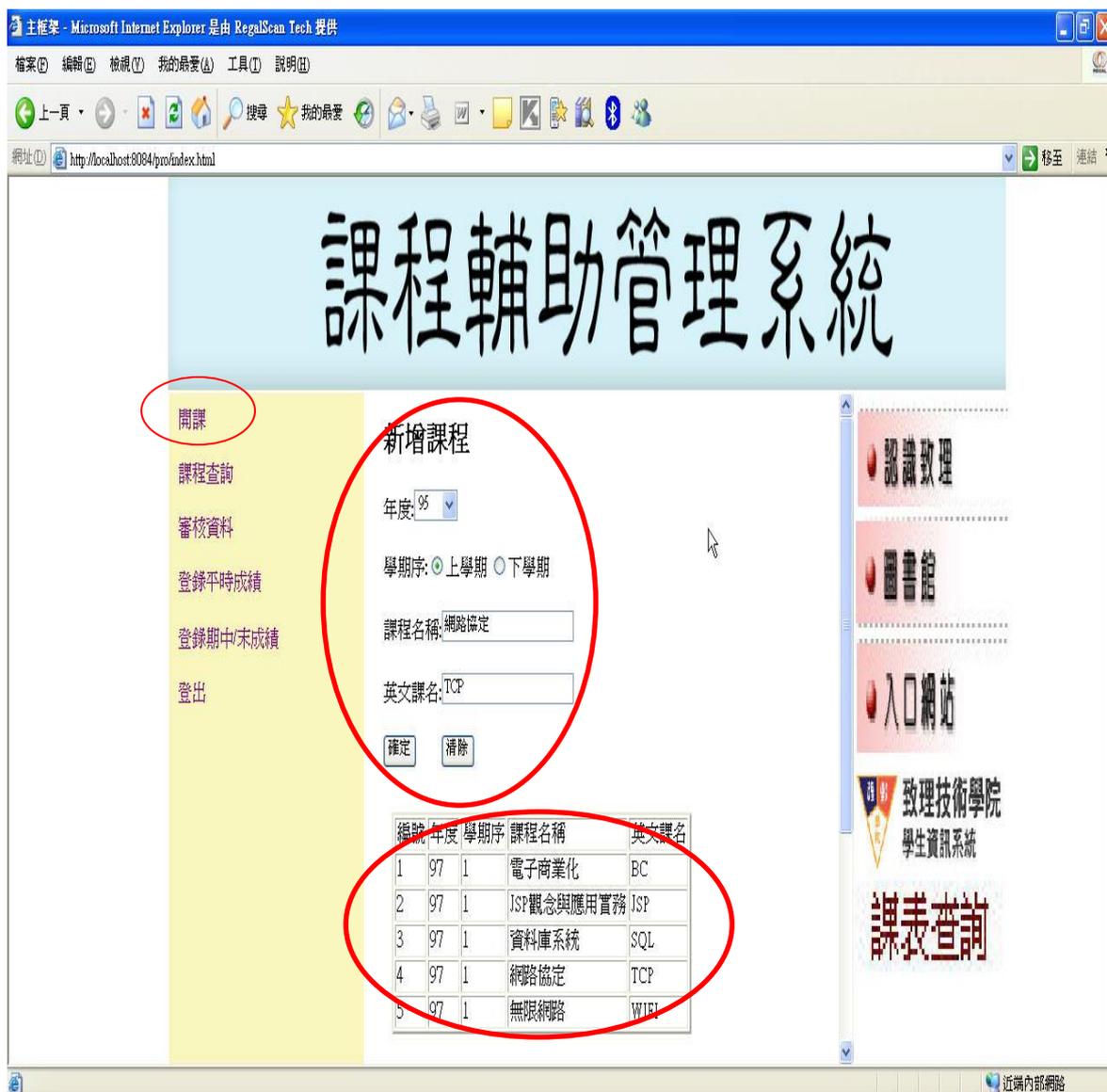


圖 4-19 教師開課畫面

當然我們也很貼心的，設立了可以查詢課程的功能如圖 4-20 所示，教師端跟學生端最大不同的就是，教師端的課程查詢是可以查詢教師個人歷年來所有開課的資料，所以只要輸入一些條件即可查詢教師想要看到的課程資訊。

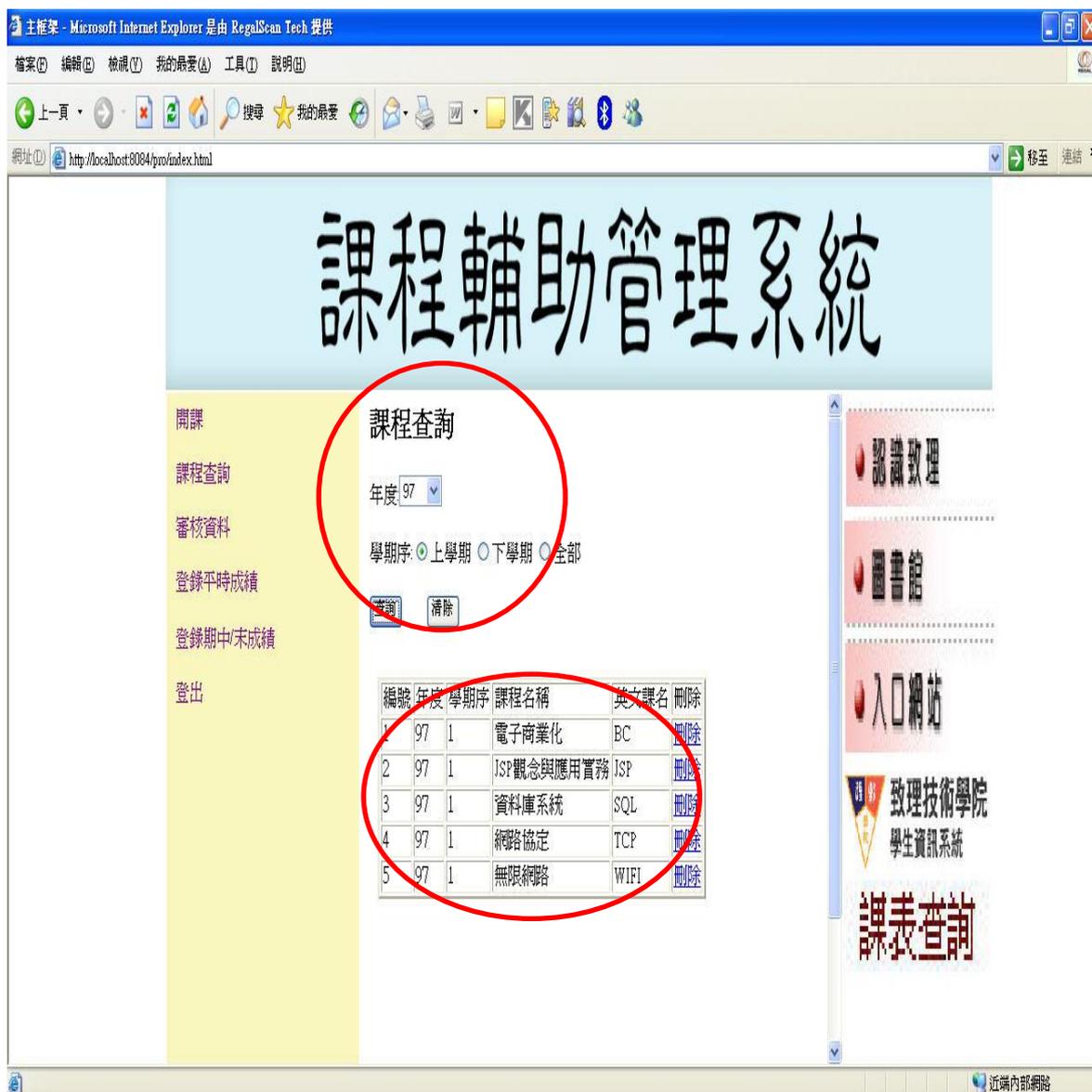


圖 4-20 老師端課程查詢畫面

當然教師端的課程查詢不止可以查到歷年來所有開課資訊，當教師不小心在新增開課時輸入錯誤，也可以從這裡刪除該錯誤的課程，便可從新開課。

開完課後，在學生註冊後，教師還有一項很重要的動作要處理，就是幫學生審核資料如圖 4-21 所示，要先幫學生審核通過資料，學生才可以登入此管理系統唷!當然如果學生註冊資料輸入錯誤時，教師可以直接從這裡刪除後請學生重新註冊。

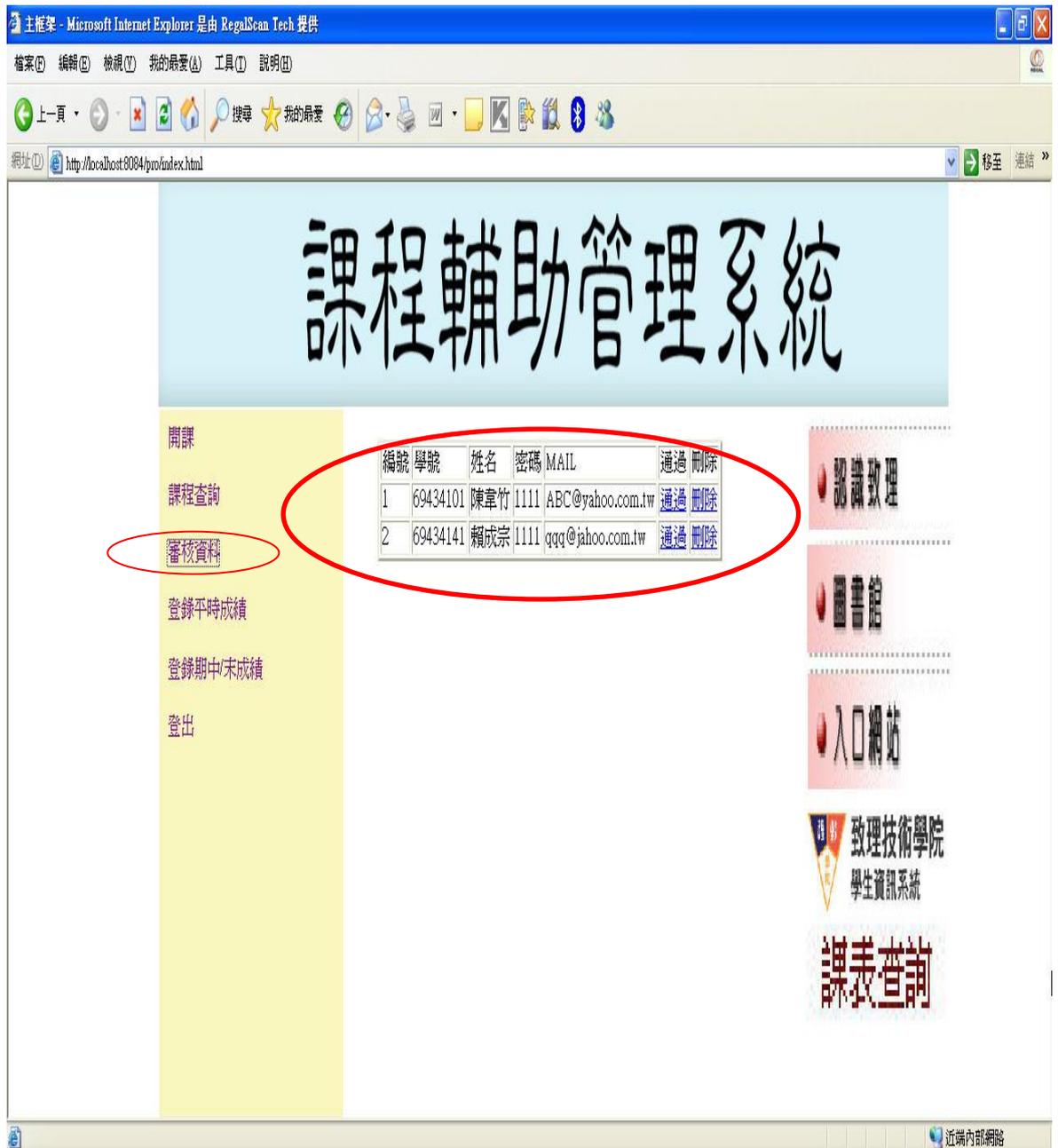


圖 4-21 審核資料畫面

關於學生評分的動作如圖 4-22 所示，平時成績與期中期末考成績使用方法相同，首先點選左邊選單的「登錄期中期末考成績」，選擇要登錄哪一種成績，以期中考為例。

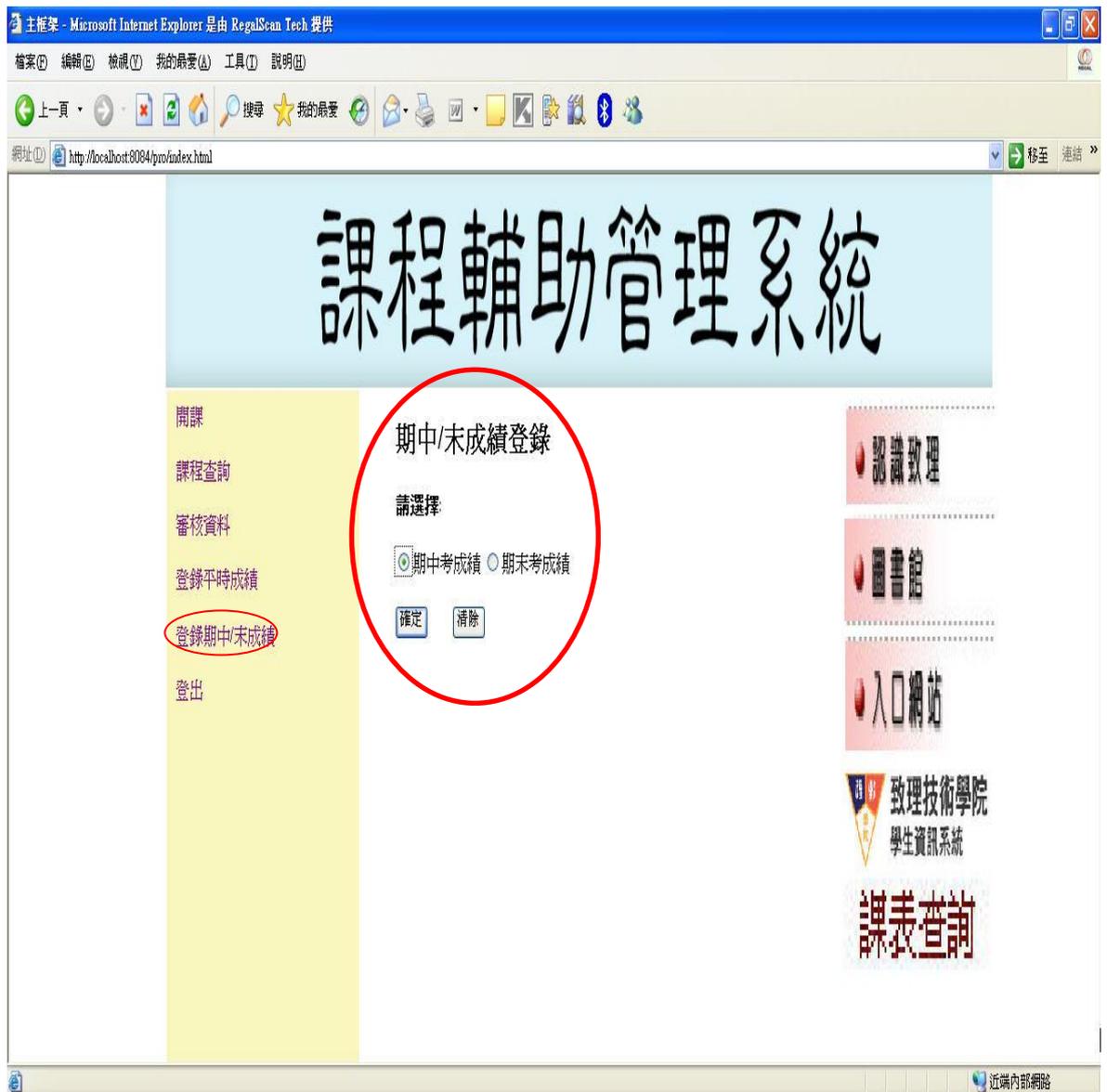


圖 4-22 登入成績 1

此畫面是唯一跟「登錄平時成績」不一樣的，平時成績登入直接進入選課畫面不需要在選擇期中或期末。

選擇期中考成績之後，會再出現請教師選擇對哪一個課程評期中考成績如圖 4-23 所示，這裡只會出現這一學期內老師全部所開的課程為選單。

課程輔助管理系統

登錄期中考成績

編號	年度	學期序	課程名稱	選取
1	97	1	電子商業化	選取
2	97	1	JSP觀念與應用實務	選取
3	97	1	資料庫系統	選取
4	97	1	網路協定	選取
5	97	1	無限網路	選取

開課
課程查詢
審核資料
登錄平時成績
登錄期中/末成績
登出

認識教理
圖書館
入口網站
致理技術學院
學生資訊系統
課表查詢

近端內部網絡

圖 4-23 登入成績 2

教師選擇好要評分之課程後如圖 4-24 所示，就會出現該課程所有學生的簡單資料，只需要在空格框框中輸入分數按下確定即可。

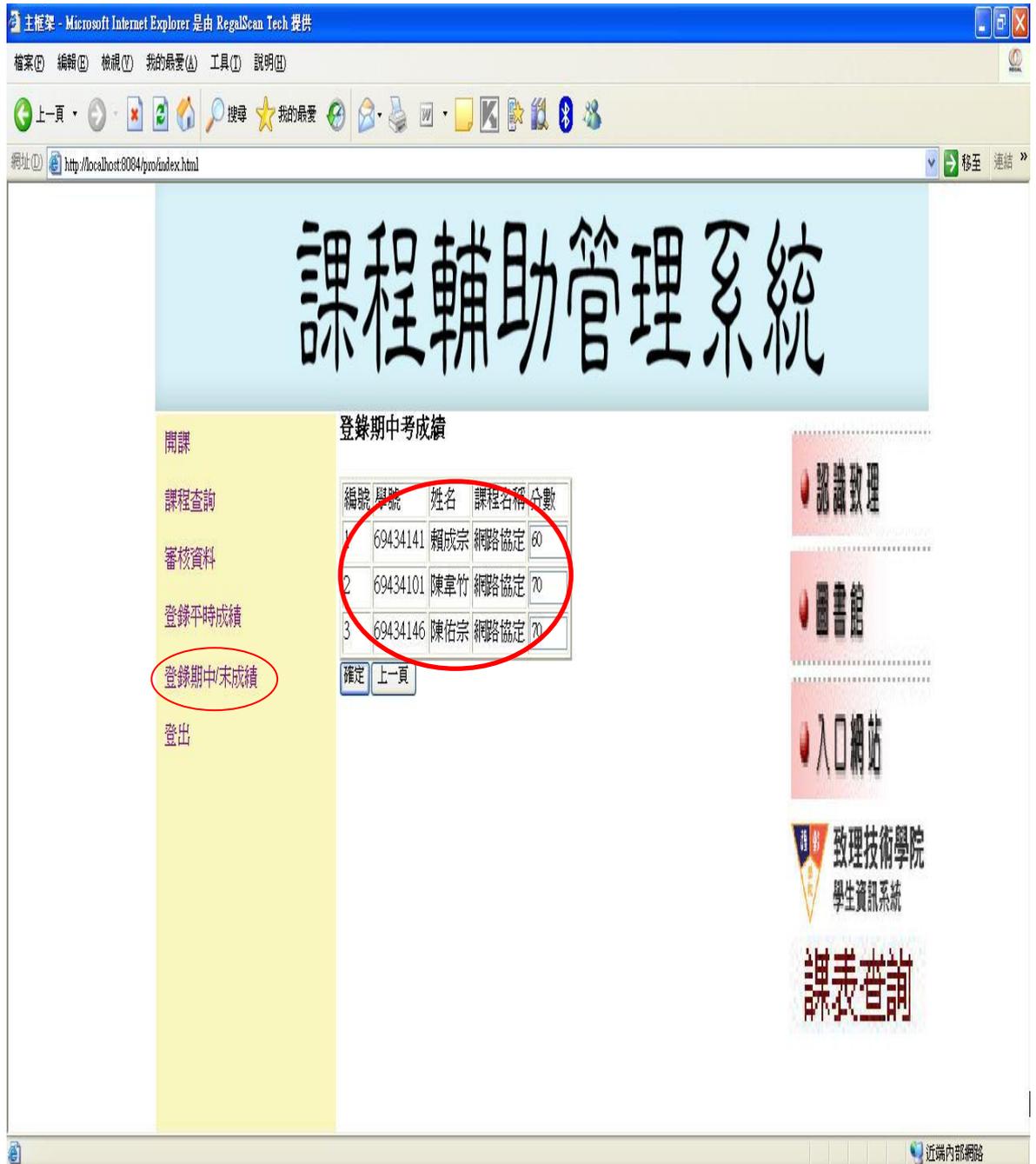
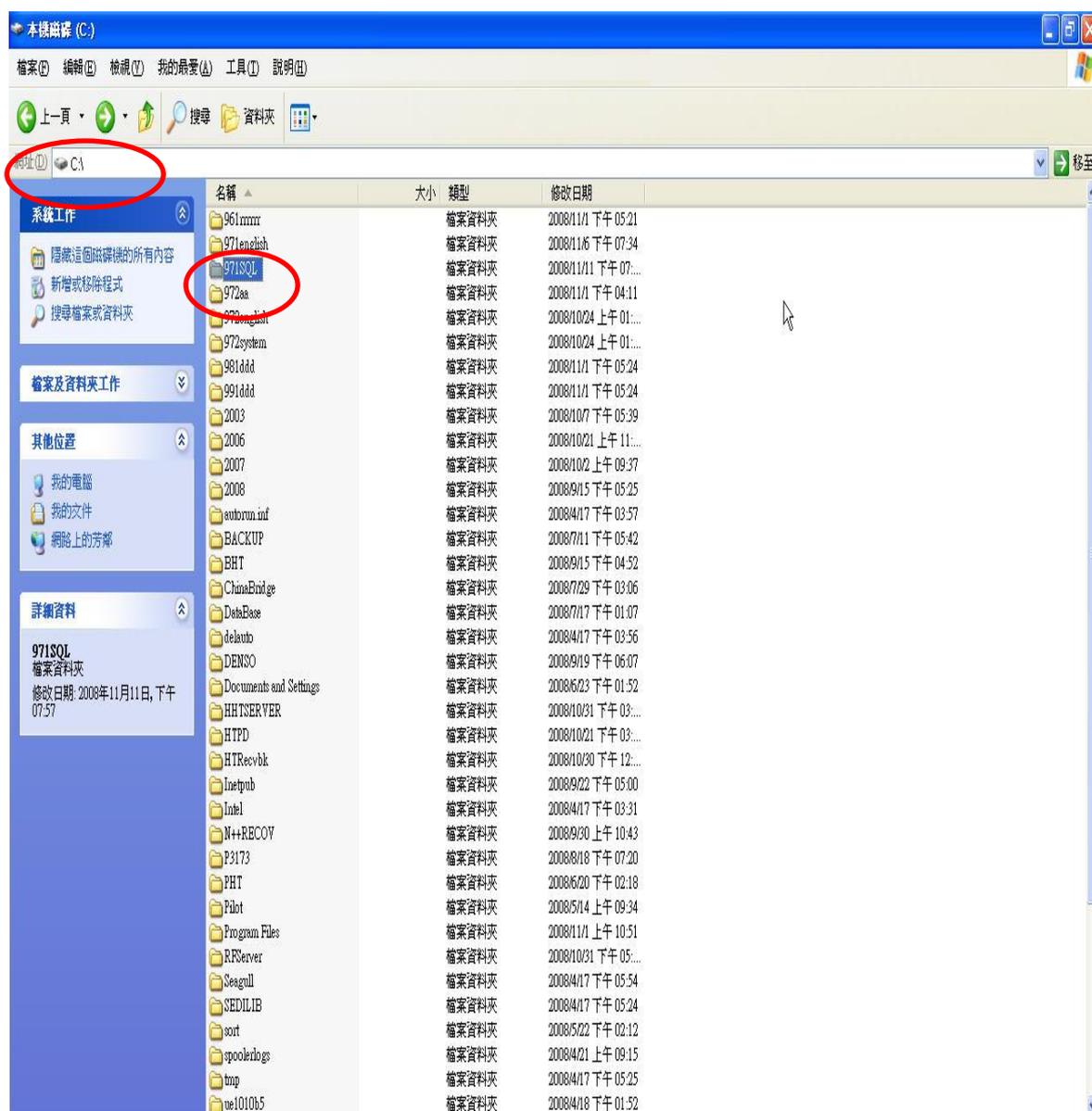


圖 4-24 登入成績 3_

當學生上傳作業後，教師應該要在哪裡尋找作業呢？我們在設計上做了小小的巧思，當教師在開課時，會輸入哪一年度哪一學期還有英文課程名稱，這些就是此課程之資料夾名稱如圖 4-25 所示，而所有資料夾統一都放置在「我的電腦」的「C:\」內，例如：我們開了一門 97 年度第一學期英文名稱為 SQL 的課程，那麼在 C:\磁碟區內，就可以找到命名為「971SQL」的資料夾，此資料夾就是該課程學生上傳作業的存放點。



4-25 作業存放點 1

打開資料夾後如圖 4-26 所示，就可看到學生們所上傳的資料了。例如我們在學生端已經上傳一份架構的資料（學生選課 97 年上學期的 SQL 課程），當老師端在主機端就可以在 C:\971SQL 的資料夾內看到這份文件。

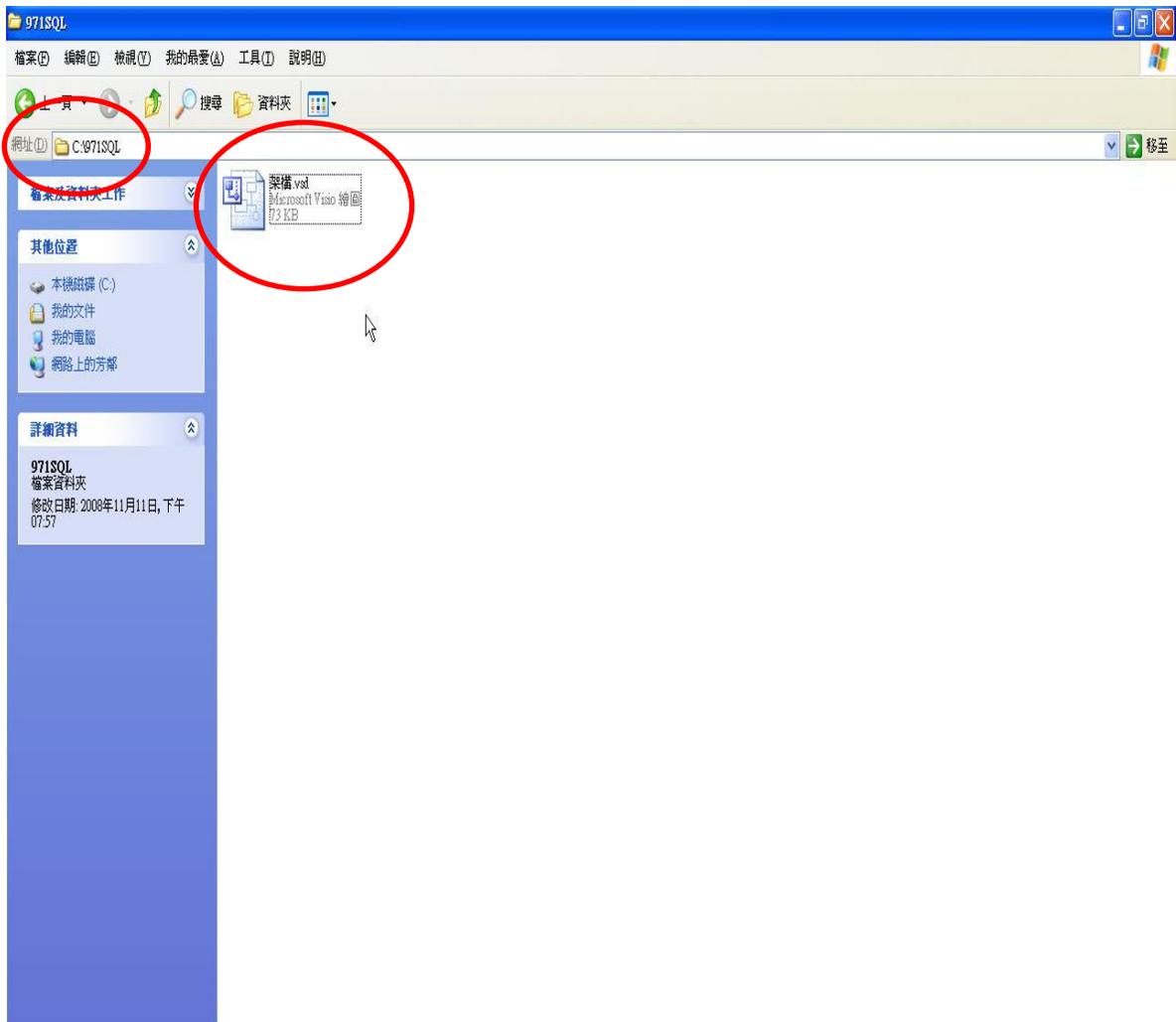


圖 4-26 作業存放點 2

第五章 結論及後續發展方向

本系統是以網頁結合資料庫系統研發出符合教師需求的課程管理系統，系統以簡單直接做為主軸，幫助教師利用網際網路，進行管理開課以及管理學生成績及作業，節省時間成本、提高效率，並降低錯誤發生機率。

在介面上，我們採取較單純的方式，沒有華麗而複雜的介面，使用者不需額外的教學就可直接使用。在功能上，本系統提供了教師開課資訊以及成績登錄，學生查詢以及學生作業上傳。本組所研發的教師課程輔助暨管理資訊系統，可有效的提升教學效率。在後續發展上，本組還打算繼續將系統在擴充與應用，如下所列：

1. 建立留言系統，便於教師及學生雙向溝通。
2. 滙入教學計畫，便於學生了解課程方向及內容。
3. 提供學生課程時間、教室查詢。
4. 擴充研發線上測驗之功能。
5. 老師可將課程講義上傳讓學生下載。
6. 老師在開課時，可加入課程大綱，開出書籍清單供學生參考。

參考文獻

- 〔1〕蕭世文，2006，HTML/JavaScript 網頁程式設計：XML+CSS，第一版，松崗文魁。
- 〔2〕胡崧，HTML 玩美設計範例語典，2007，第一版，PCUSER 電腦人。
- 〔3〕吳目誠，劉麗伶，精彩 Dreamweaver MX 2004 動態網站，2004，第一版，網奕。
- 〔4〕方盈，Microsoft SQL Server 2005 資料庫管理實務，2006，第一版，博碩。
- 〔5〕廖信彥，SQL Server 2005 深度管理與應用 (Pro SQL Server 2005)，第一版，2007，博碩。
- 〔6〕陳會安，SQL Server 2005 資料庫系統設計與開發實務，第一版，2007，學貫行銷股份有限公司。
- 〔7〕html 概論，<http://dormouse.instantjsp.com/SimplePage.html>。
- 〔8〕施威銘研究室，Microsoft SQL Server 2005 設計實務，2006，第一版，旗標出版社。
- 〔9〕蕭明城，周岱琳，SQL 設計與運用範例集，2003，初版。
- 〔10〕張真誠，蔡文輝，林敏惠，挑戰資料庫管理系統，2003，第一版，旗標出版社。
- 〔11〕Paul Tremblet，Java Server Pages 完全引導手冊，2001，第一版，麥格羅.希爾國際出版公司。
- 〔12〕吳仁和，林信惠，系統分析與設計：結論與實務應用，2004，第三版，智勝文化事業有限公司。
- 〔13〕陳會安，JSP2.0 網頁設計範例，2005，第一版，學貫行銷股份有限公司。
- 〔14〕黃三益，資料庫的核心理論與實務，2007，第三版，華東。

〔15〕余昀，精通 JSP 網頁程式設計，2005，初版。

〔16〕張心怡，陶幼慧，2002，資料庫正規劃與 E-R Model 簡介，義守大學，
資訊管理系。