

## 第十二章 專題研討 - 課務管理系統

### 12-1 系統需求

我們建構一套『教務開課/選課系統』，來探討建構一套資料庫系統的過程如何，也依照前面幾章所介紹方法分別實現之。

#### 12-1-1 系統需求情境

教務處希望建立一套可供老師開課，與讓學生選課系統。期望此系統可讓各系老師開課，學生也可依照自己的興趣到各系選課。真實情境如下：

- 另外學校設有若干個系（包含各學制），每一個系裡有若干位專任老師。
- 每系主任由各系老師中挑選一位擔任，系所名稱可能會依照環境變化而改名。
- 每一個系招收若干班，每個班級都有一個班級名稱，譬如：日四資三甲，表示日間部四技甲班、進四資二丙，表示進修部四技丙班，這些名稱會隨年變更，譬如，明年上述班級就成為日四資四甲、進四資三丙。
- 學校每年招收新學生入學，並依照學生分發到那一系、那一班而編定學號，絕對不會與其他同學相衝突。學生的學號到畢業後都不會變更，無論轉班、轉系或休退學。
- 學校裡有若干位老師，分發到各系，並給予教師編號。無論教師轉調到那一系，他的編號都不會改變。
- 各系依照其特性規劃了若干課程，並繳交給教務處。教務處歸納各系所計畫的課程，編製一份『課程資料表』，並給予每一課程一個代碼。
- 每學期各系將所預計開設的課程繳交給教務處，其中包含課程名稱、開課老師、開課班級。教務處歸納各系所開的課程建立了『開課總表』，其中同一門課可能由不同老師開在不同班級或系所上。同一位老師可能在本系所或其他系開設多門課程，
- 學生依照開課總表選課，所選課程並不限定於那一系開設，但不可同時選讀同一課程。
- 學生選讀課程可以登錄分數。

## 12-1-2 資料收集

我們收集到的資料如下：

- (1) 學生資料：姓名、學號、性別、電話、地址、電子郵遞、、、、。
- (2) 系所資料：系名、主任。
- (3) 班級：班級名稱、、、。
- (4) 教師：姓名、員工代碼、地址、電話、職稱、、、。
- (5) 課程：課程名稱、課程代碼、學分數、必選修。

## 12-1-3 系統驗證範例

我們希望建構成功的系統可讓老師開課、學生選課。為了驗證系統功能，吾人選定下列查詢題目，如果能成功查詢的話，初步驗證系統符合需求：(輸入資料時，也依照題目需求輸入)

1. 請顯示所有學生的學號、姓名、性別、地址、電話與電子郵件。
2. 請列出 102 學年度入學『女』學生，且住在高雄市的姓名、電話與地址。
3. 請列出資管系老師的姓名、性別、電話、電子郵件與地址。
4. 請列印出各系系主任的系別名稱、姓名、電話與地址。
5. 請查詢這學期每位老師各開幾門課。
6. 請列印出『進四資二丙』的學生名冊。
7. 請查詢『進四資二丙』這學期開了哪些課。
8. 找出全校內姓名相同的學生。
9. 請列印出『資訊工程系』規劃有哪些課程，依課程名稱、必選修、學分數印出。
10. 請查詢這學期有哪幾位老師有開『資料庫管理系統』這門課。

11. 請查詢『粘添壽』老師這學期開了哪些課程。
12. 請列印出這學期修讀『粘添壽』老師所開的『資料庫管理系統』的學生名冊。

## 12-2 資料庫邏輯設計

我們利用由外往內拆解方法，來分析此系統的邏輯架構，分別說明如下：

### 12-2-1 第一層分析：系統基本需求

實際狀況：

1. 學生可透過該系統選課。
2. 老師也可透過該系統開課。

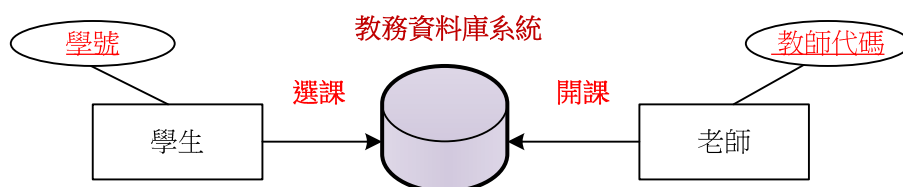


圖 12-1 系統基本需求

### 12-2-2 第二層分析：班別與系所組織

實際狀況：

1. 每一位學生都被歸屬於某一班級。
2. 每一位老師也被歸屬於某一系所。

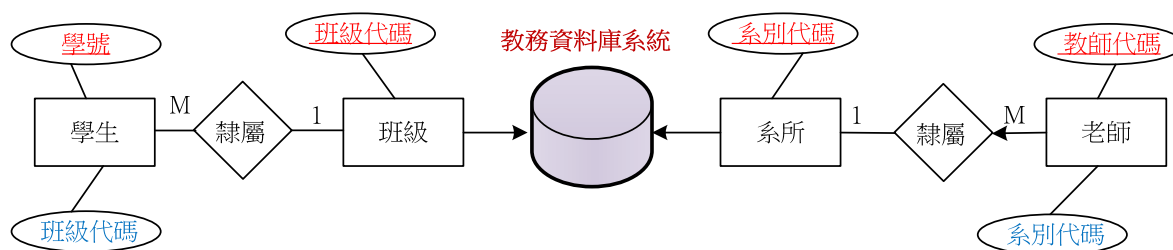


圖 12-2 班別與系所組織

圖 12-2 是學生與班級之間的 E-R 圖，每一位學生僅隸屬於某一個班級，但該班級可擁有多位學生，因此他們之間是多對一的關係。兩者之間連結係利用班級代碼，在學生資料中

加入一個外來鍵『班級代碼』，連結到班級資料的主鍵『班級代碼』。教師與系所關係也是如此，我們在系所資料加入『系別代碼』當主鍵，也在老師資料內加入外來鍵『系別代碼』，以標示老師是隸屬哪一個系所的。

### 12-2-3 第三層分析：課程概況

實際狀況：

1. 各系將所規劃課程繳交給教務處(課程名稱、必選修、學分數)。
2. 教務處將各系所繳交課程歸納成一個課程總表(加入系別代碼、課程代碼)。

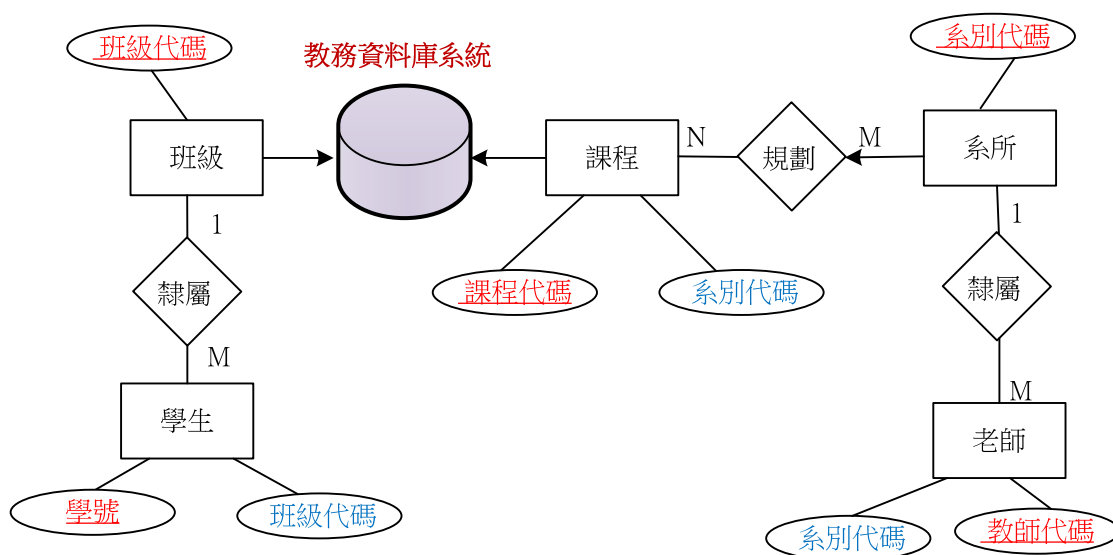


圖 12-3 課程概況

我們增加了『課程』實體來顯示各系所規劃的課程，它與系統之間的關係是多個系所都可以繳交所欲開設的課程，且每一系所可繳交多門課，同一門課也可以由多個系所開設，因此，課程與系所之間是多對多的關係。為了辨識各個課程，我們增加了『課程代碼』做為主鍵，再增加『系別代碼』為外來鍵，以聯繫它是屬於哪系所開設。

### 12-2-4 第四層分析：學期開課概況

實際狀況：

1. 每一學期開始，老師由課程總表內選擇開設課程(或被指定開設)。
2. 老師可以跨系開課，表示所開課程不一定是他所屬系所規劃的。

3. 教務處將所有老師開設的課程彙整成一總表。

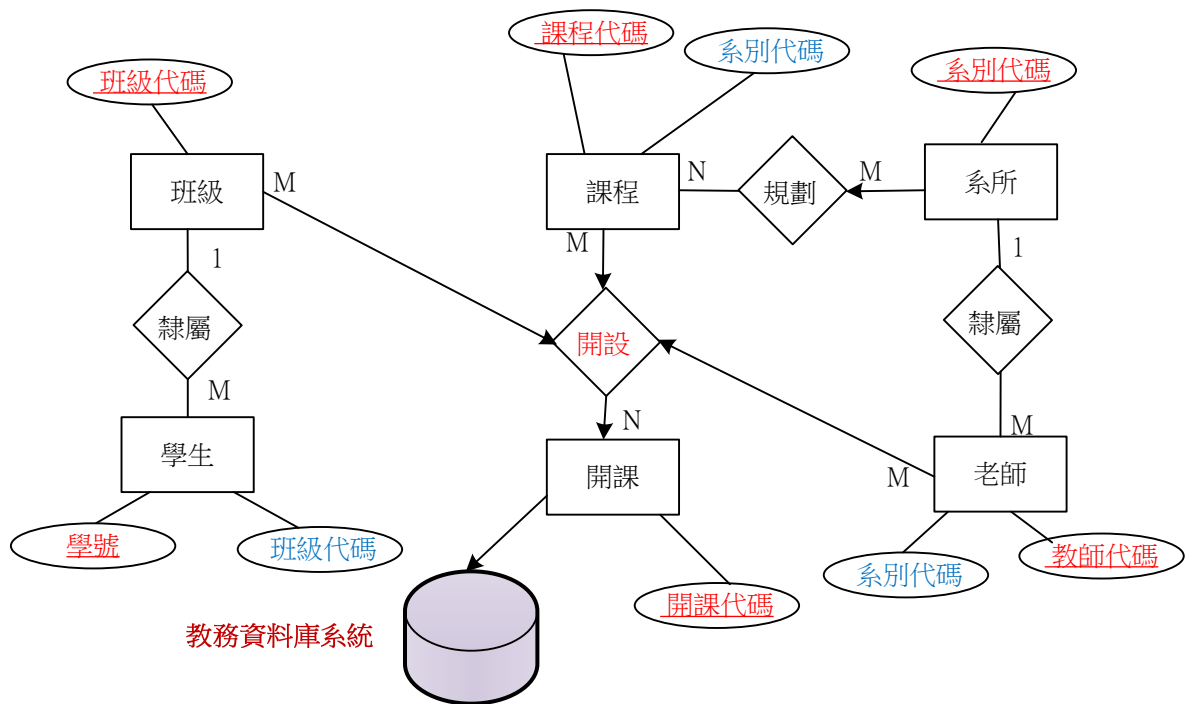


圖 12-4 開課狀況

開設課程關係到老師、課程與班別，而且之間都是多對多的關係，我們還是另外增加『開課代碼』做為該實體的主鍵，在操作方面會比較容易。

### 12-2-4 第五層分析：學生選課概況

實際狀況：

1. 學生每一學期可選讀若干門課。
2. 學生可以跨系選課。
3. 同一門課可讓多位學生修讀。
4. 學生同一學期、同一門課只能選讀一次。
5. 教務處將所有學生所選讀課程彙整，並允許老師輸入學生成績。

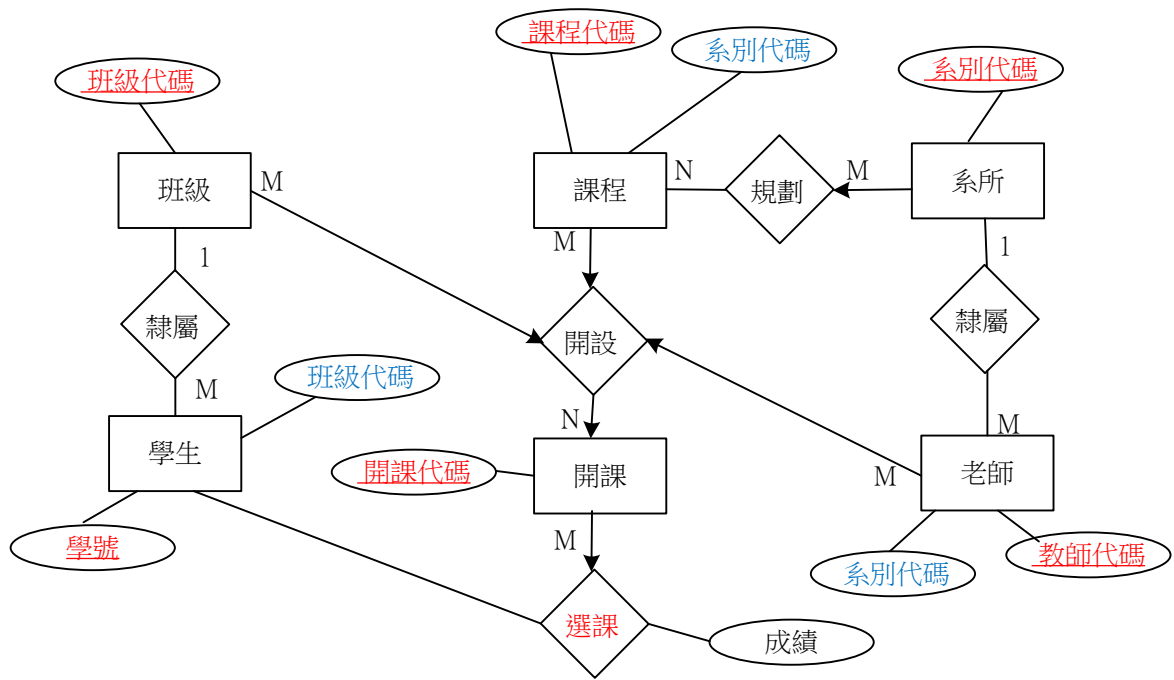


圖 12-5 選課狀況

學生可由開課表內選擇課程，每一課程也可讓多位學生選讀，因此它們之間是多對多的關係，我們只要增加『學號』與『開課代碼』等兩個外來鍵，當作複合主鍵，再加入一個『成績』的欄位即可。

### 12-2-4 第六層分析：加入所需屬性

分析到這裡幾乎完成了，接下來依照系統需求(驗證題目)加入個實體或關係的屬性，結果如下：(請參考附件)

Course\_db系統 E-R 關係圖

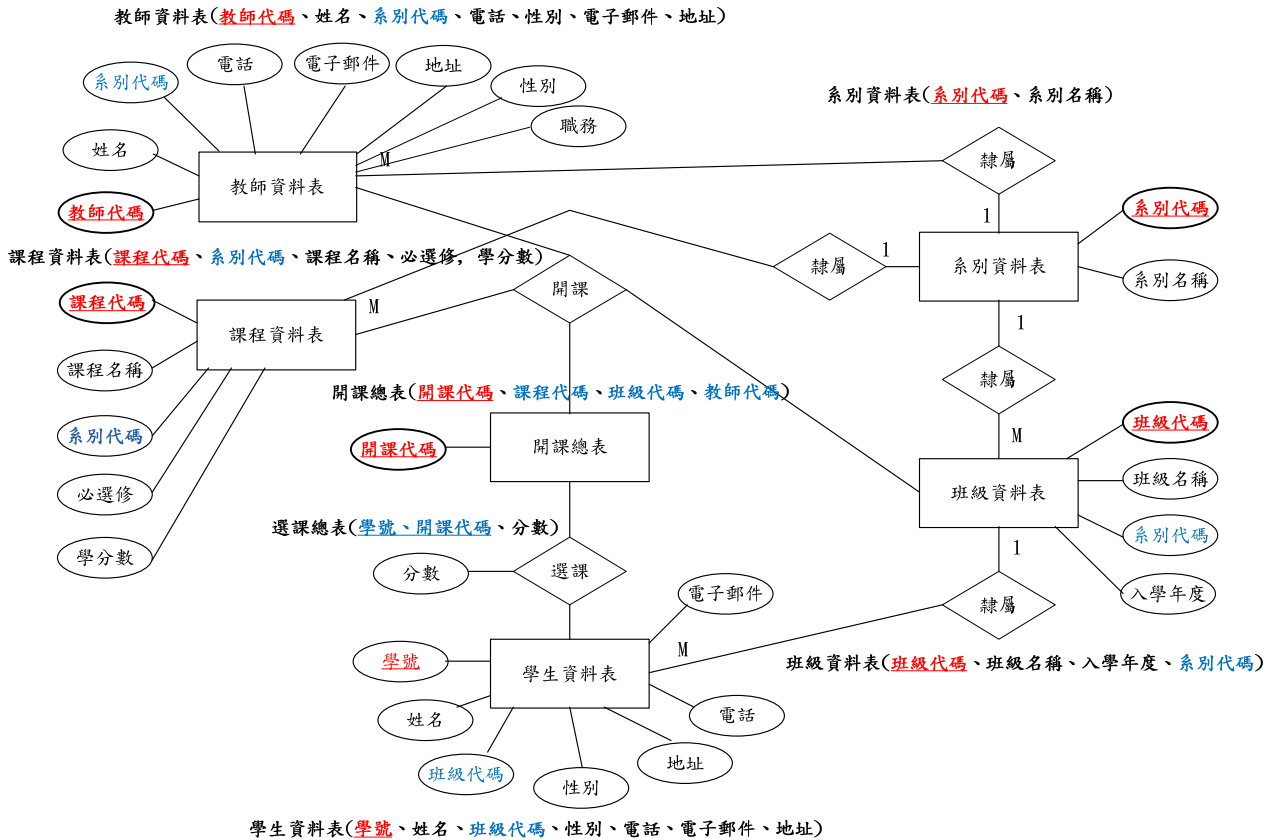


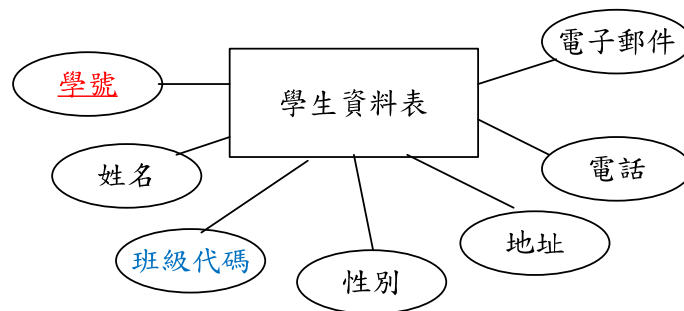
圖 12-6 課務管理系統的 E-R 關係圖

## 12-3 E-R 圖轉換成資料表

完成系統分析並繪製 E-R 關係圖完成之後，接下來必須將各個實體或關係轉換為資料表，在轉換過程中也順便分析它是否滿足 3NF 正規化。

### 12-3-1 學生資料表 - students

學生實體的 E-R 圖如下：



取名為『學生資料表』(students)，並選擇學號為主鍵。各個欄位名稱與資料型態如下所示：

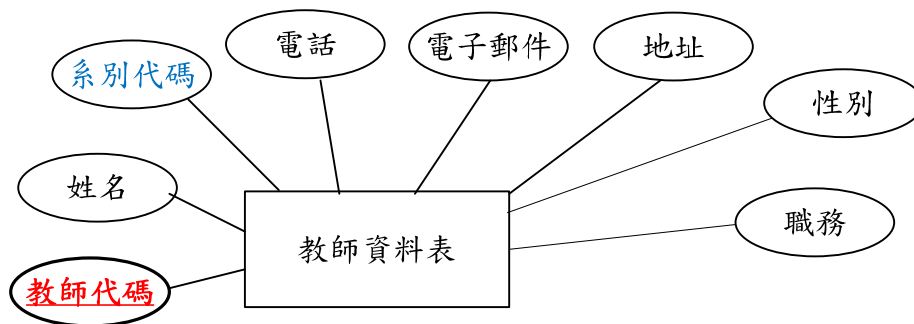
學生資料表						
<u>學號</u>	姓名	班級代碼	性別	地址	電話	電子郵件
Student_ID	name	Class_ID	sex	address	tel	mailbox
CHAR(10)	CHAR(10)	CHAR(10)	CHAR(5)	CHAR(50)	CHAR(10)	CHAR(20)

正規化分析：

- 1NF：所有欄位都是最小單元(Atomic value)。符合。
- 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
- 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-2 教師資料表 - teachers

教師實體的 E-R 圖如下：



取名為『**教師資料表**』(teachers)，並選擇教師代碼為主鍵。各個欄位名稱與資料型態如下所示：

學生資料表							
<u>教師代碼</u>	姓名	系別代碼	性別	職務	地址	電話	電子郵件
<u>Teacher_ID</u>	name	Dep_ID	sex	title	address	Tel	mailbox
CHAR(10)	CHAR(10)	CHAR(10)	CHAR(5)	CHAR(20)	CHAR(50)	CHAR(10)	CHAR(20)

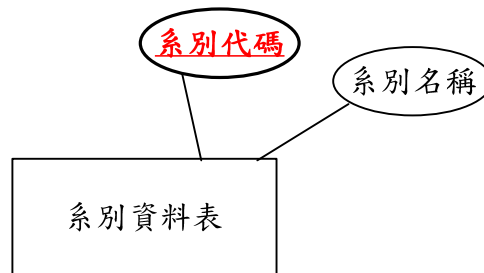
正規化分析：

- 1NF：所有欄位都是最小單元(Atomic value)。符合。
- 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
- 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。



### 12-3-3 系所資料表 - department

系所實體的 E-R 圖如下：



取名為『系所資料表』(department)，並選擇系所代碼為主鍵。各個欄位名稱與資料型態如下所示：

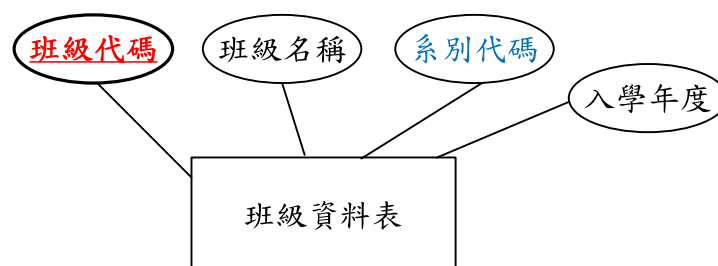
系所資料表	
<u>系所代碼</u>	系別名稱
<u>Dep_ID</u>	Dep_name
CHAR(10)	CHAR(10)

正規化分析：

1. 1NF：所有欄位都是最小單元(Atomic value)。符合。
2. 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
3. 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-4 班級資料表 - classes

班級實體的 E-R 圖如下：



取名為『班級資料表』(classes)，並選擇『班級代碼』(class\_ID)為主鍵。各個欄位名稱與資

料型態如下所示：

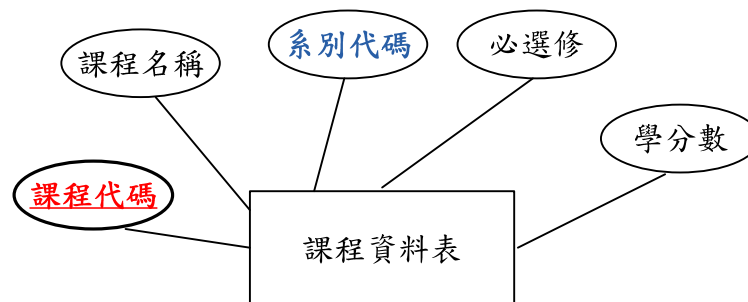
班級資料表			
<u>班級代碼</u>	班級名稱	系別代碼	入學年度
<u>Class_ID</u>	Class_name	Dep_ID	year
CHAR(10)	CHAR(10)	CHAR(10)	int

正規化分析：

1. 1NF：所有欄位都是最小單元(Atomic value)。符合。
2. 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
3. 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-5 課程資料表 – all\_course

課程實體的 E-R 圖如下：



取名為『課程資料表』(all\_course)，並選擇『課程代碼』(course\_ID)為主鍵。各個欄位名稱與資料型態如下所示：

課程資料表				
<u>課程代碼</u>	課程名稱	系別代碼	必選修	學分數
<u>Course_ID</u>	Course_name	Dep_ID	required	credit
CHAR(10)	CHAR(10)	CHAR(10)	CHAR(5)	int

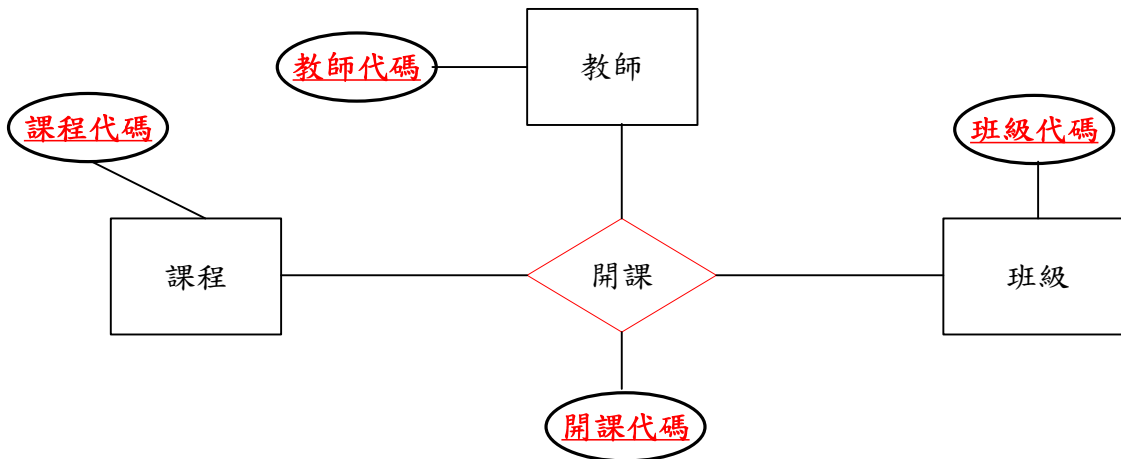
正規化分析：

1. 1NF：所有欄位都是最小單元(Atomic value)。符合。
2. 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。

3. 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-6 開課資料表 – open\_course

開課實體的 E-R 圖如下：



取名為『開課總表』(open\_course)，並選擇『開課代碼』(open\_course\_ID)為主鍵。各個欄位名稱與資料型態如下所示：

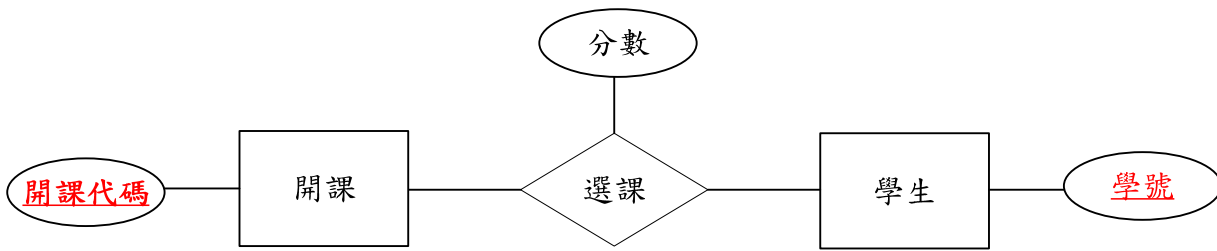
開課總表			
<u>開課代碼</u>	課程代碼	班級代碼	教師代碼
<u>Open course ID</u>	Course_ID	Class_ID	Teacher_ID
CHAR(10)	CHAR(10)	CHAR(10)	CHAR(10)

正規化分析：

1. 1NF：所有欄位都是最小單元(Atomic value)。符合。
2. 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
3. 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-7 選課資料表 – select\_course

選課實體的 E-R 圖如下：



取名為『選課總表』(select\_course)，並選擇『學號』(student\_ID)與『開課代碼』(open\_course\_ID)為複合主鍵。各個欄位名稱與資料型態如下所示：

選課總表		
<u>學號</u>	<u>開課代碼</u>	分數
<u>Student_ID</u>	<u>Open_course_ID</u>	final
CHAR(10)	CHAR(10)	int

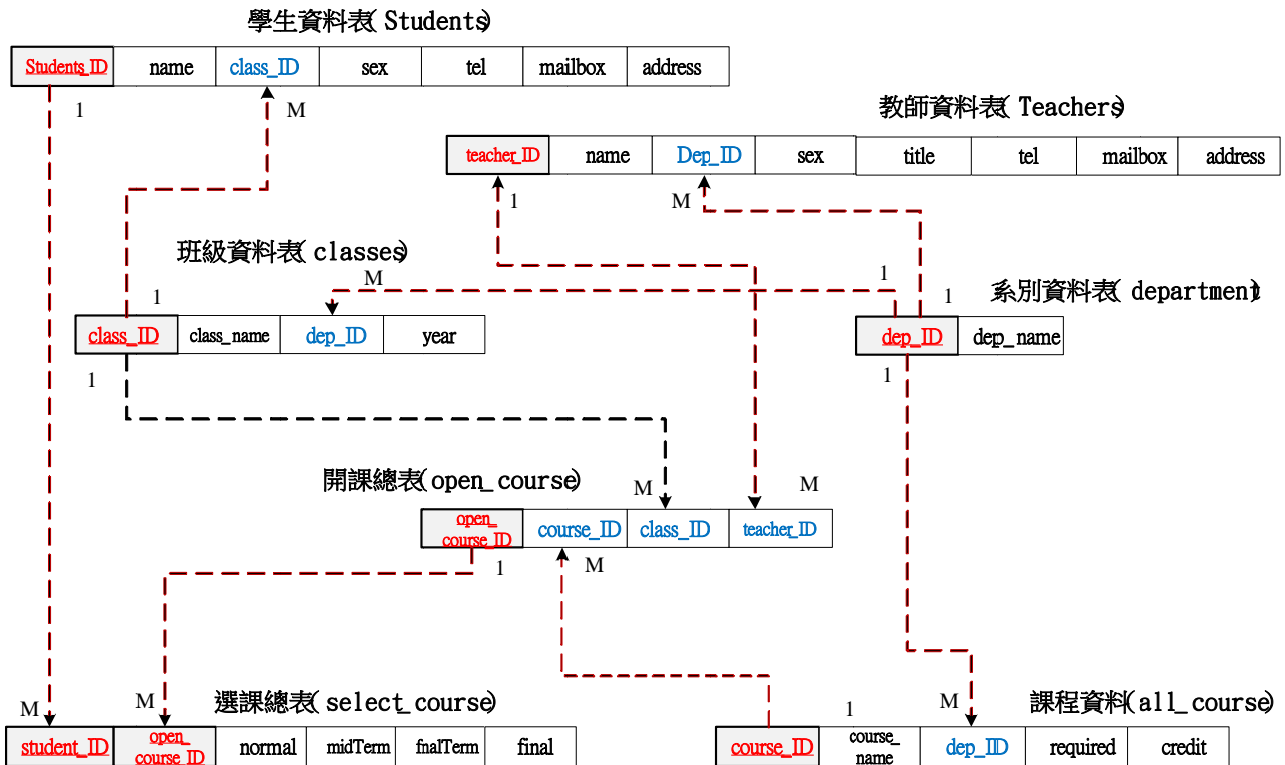
正規化分析：

1. 1NF：所有欄位都是最小單元(Atomic value)。符合。
2. 2NF：除了主鍵外，所有欄位都與主鍵相依。符合。
3. 3NF：除了主鍵外，所有欄位之間都沒有功能性相依。符合。

### 12-3-8 所有表格關聯圖

完成各資料表設計之後，接下來繪出所有資料表之間的關聯圖，如下：

### 課程管理系統 course\_db 表格關聯圖

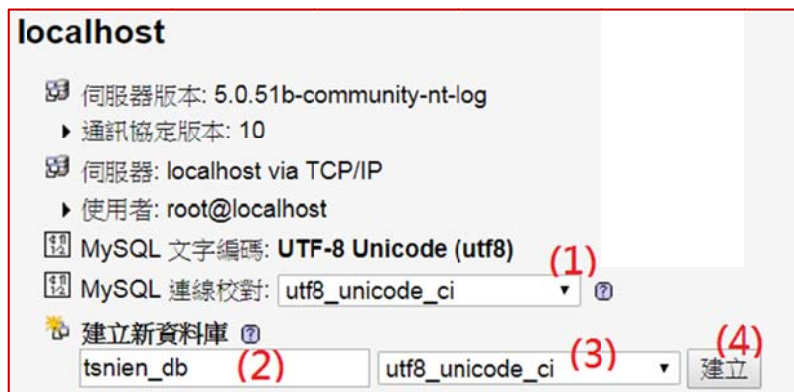


## 12-4 實體建置

完成各個資料表設計之後，接下來必須實體建置起來，目前我們係利用 AppServ 套件實習，就將它建立在 AppServ 上。

### 12-4-1 建立新資料庫

進入 AppServ 後建立資料庫，如下：



### 12-4-2 建立新表格

(A) 建立表格：

資料庫中沒有資料表

建立新資料表於資料庫 `tsnien_db`

名稱:  欄位數目:

欄位數

表格名稱

執行

(B) 建立表格欄位屬性：

伺服器: localhost > 資料庫: tsnien\_db > 資料表: students

欄位	型態	長度/集合 <sup>1</sup>	校對	屬性	Null	預設值 <sup>2</sup>	附加				
student_ID	VARCHAR	10	utf8_unicode_ci	UNSIGNED	not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	VARCHAR	50	utf8_unicode_ci	UNSIGNED	not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
class_ID	VARCHAR	10	utf8_unicode_ci	UNSIGNED	not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sex	VARCHAR	10	utf8_unicode_ci	UNSIGNED	not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tel	VARCHAR	50	utf8_unicode_ci	UNSIGNED	null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
mailbox	VARCHAR	50	utf8_unicode_ci	UNSIGNED	null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
address	VARCHAR	50	utf8_unicode_ci	UNSIGNED	null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>