

關聯式資料庫

關聯式資料庫(Relation Database)

假設學校行政系統中有一個尚未分割的「學籍資料表」，如下表所示：

	學號	姓名	系碼	系名	系主任
#1	S0001	一心	D001	資工系	李春雄
#2	S0002	二聖	D001	資工系	李春雄
#3	S0003	三多	D002	資管系	李碩安
#4	S0004	四維	D002	資管系	李碩安
#5	S0005	五福	D002	資管系	李安

大量資料重複現象

由上表中，我們可以清楚看出**多筆資料重複現象**，如果有某一筆資料打錯，將會**導致資料不一致現象**。例如：在上表中的第5筆記錄的系主任，應該是「李碩安」卻打成「李安」。

因此，我們就必須要將原始的「**學籍資料表**」分割成**數個不重複的資料表**，再利用「**關聯式資料庫**」的方法來進行資料表的關聯。

何謂「**關聯式資料庫**」呢？它是由**兩個或兩個以上的資料表組合而成**。其**目的**：

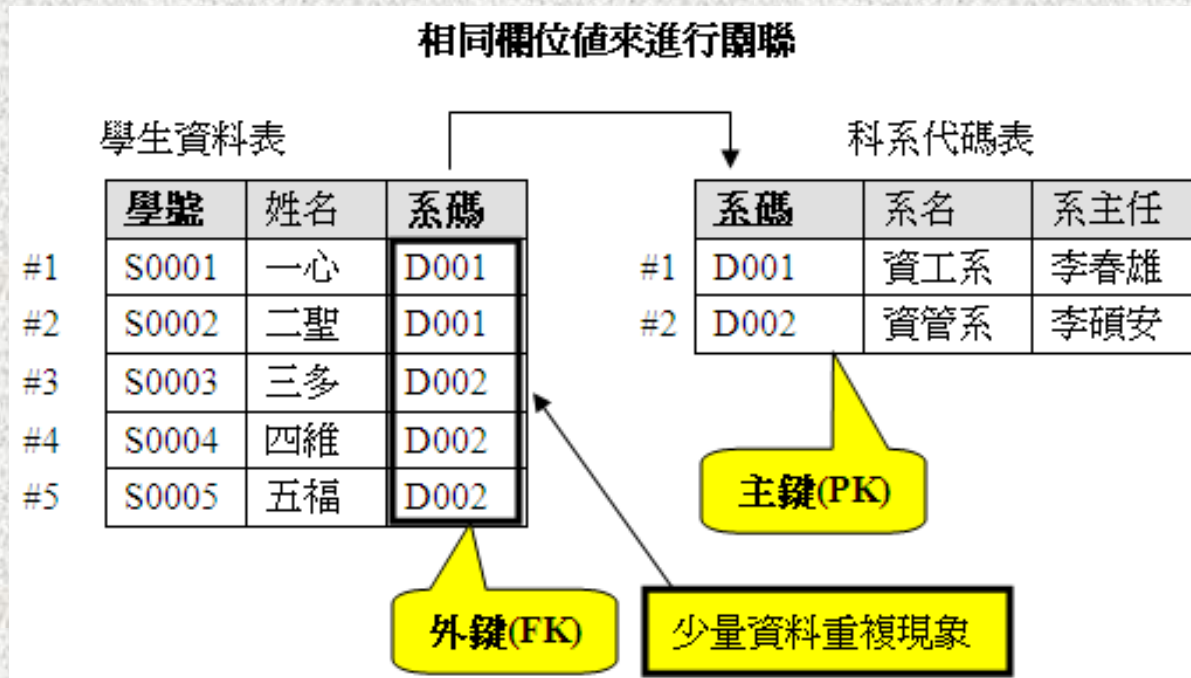
1. **節省重複輸入的時間與儲存空間**。
2. **確保異動資料(新增、修改、刪除)時的一致性**及**完整性**。

因此，我們必須將各種資料依照性質的不同(如：學籍資料、選課資料，課程資料，學習歷程資料等.....)，分別存放在幾個不同的表格中，表格與表格之間的關係，則以**共同的欄位值**(如：「**學號**」欄位...)相互連結，以這種方式來存放資料的資料庫，在電腦術語中，稱為「**關聯式資料庫(Relational Database)**」。

【定義】

- 1.是由一群相互關係的正規化關聯（表格）所組成。
- 2.關聯（表格）之間是透過相同的欄位值(即外鍵參考主鍵)來連繫。
- 3.關聯（表格）中的所有屬性內含值都是基元值(Atomic Value)。

因此，我們可以將上表中的「學籍資料表」分割為「學生資料表」與「科系代碼表」，如何產生關聯式資料庫呢？它是透過兩個資料表的相同欄位值(即系碼)來進行連結。如下所示：



註：「主鍵」與「外鍵」專有名詞會有後面章節中詳細介紹。

【優點】

1.節省記憶體空間

相同的資料記錄不須要再重複輸入。

2.提高行政效率

因為資料不須再重複輸入，故可以節省行政人員的輸入時間。

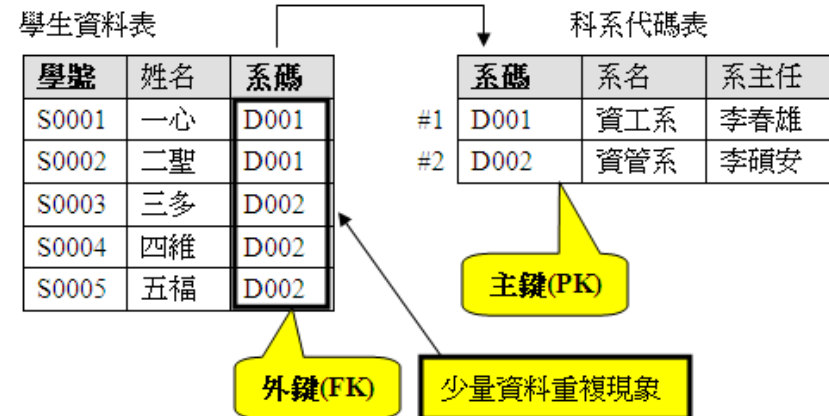
3.達成資料的一致性

因為資料不須再重複輸入，故可以減少多次輸入產生人為的錯誤。

	學號	姓名	系碼	系名	系主任
#1	S0001	一心	D001	資工系	李春雄
#2	S0002	二聖	D001	資工系	李春雄
#3	S0003	三多	D002	資管系	李碩安
#4	S0004	四維	D002	資管系	李碩安
#5	S0005	五福	D002	資管系	李安

大量資料重複現象

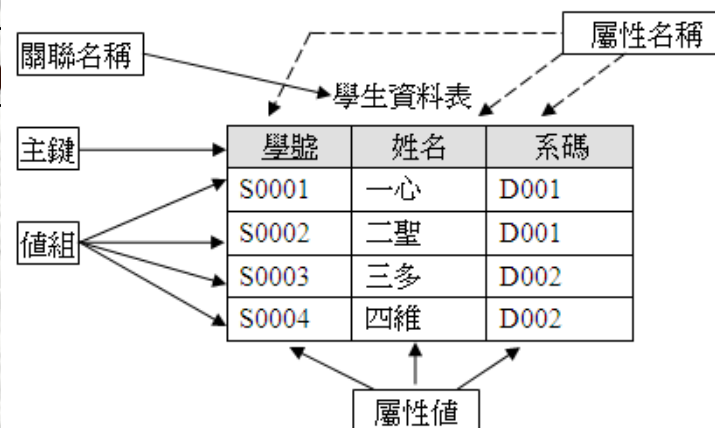
相同欄位值來進行關聯



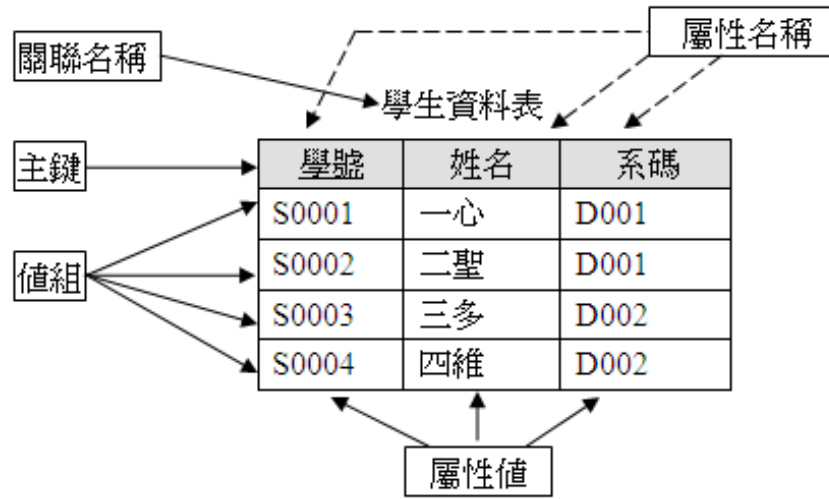
【關聯名詞】

關聯式資料庫模型的相關術語通常是用來說明資料庫系統的相關理論，而SQL Server或Access等資料庫管理系統所使用的資料庫相關名詞是利用另成一套術語，不過這些名詞或術語都代表相同意義，如表2-1所示：

關聯式資料模型	SQL Server或Access
關聯(Relation)	表格(Table)
值組(Tuple)	橫列(Row)或記錄(Record)
屬性(Attribute)	直欄(Column)或欄位(Field)
基數(Cardinality)	記錄個數(number of Record)
主鍵(Primary Key)	唯一識別(unique identifier)
定義域(Domain)	合法值群(pool legal values)



圖示說明：



【重要專有名詞】

1.資料表(Table)：又稱為**表格**，它是**真正儲存資料的地方**。它可視為特定主題的資料集合。並且它是由「**資料行**」與「**資料列**」的**二維表格**組合而成。

例如：上表中的「**學生資料表**」。

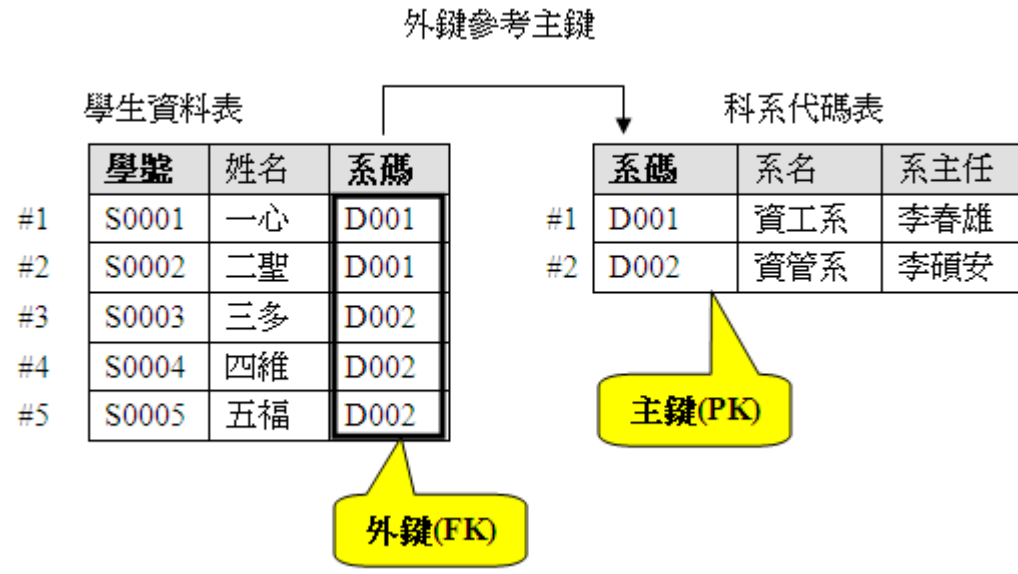
2.資料行(Column)：是指資料表中的某些「**欄位**」，它是以「**垂直**」方式來呈現。

例如：上表中的「**學號**」、「**姓名**」等。

3.資料列(Row)：是指資料表中某些「**記錄**」，它是以「**水平**」方式來呈現。

例如：上表中的**第一筆記錄**。#1 S0001，一心，D001。

圖示說明：



【重要專有名詞】

4.主鍵(Primary Key; PK)：是指用來識別記錄的**唯一性**，它**不可以重複及空值(Null)**。

例如：上表中的學生資料表中的「**學號**」及科系代碼表中的「**系碼**」

5.外鍵(Foreign Key; FK)：是指用來**建立**資料表之間的**關係**，其外鍵內含值必須要與另一個資料表的**主鍵**相同。

例如：上表中的學生資料表中的「**系碼**」。

6.關聯性(Relationship)：在資料表之間，透過**外鍵**來參考另一個資料表的主鍵，如果**具有相同欄位值**就可以進行關聯。

例如：上表中的學生資料表中的「**系碼**」與科系代碼表中的「**系碼**」都具有**相同欄位值**，因此，就可以進行關聯。

鍵值屬性

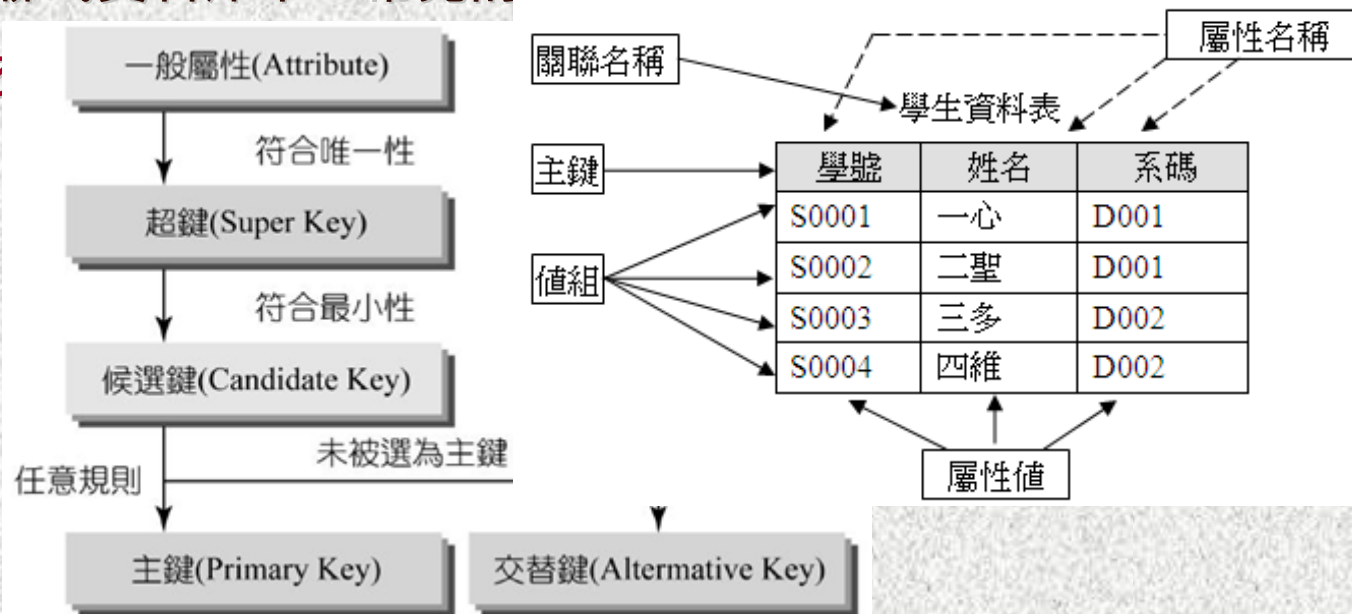
在關聯式資料庫中，每一個關聯(表格)會有許多不同的鍵值屬性 (Key Attribute)，因此，我們可以分成兩個部份來探討：

一、屬性(Attribute)：是指一般屬性或欄位。如下圖所示。

二、鍵值屬性(Key Attribute)：是指由一個或一個以上的屬性所組成，並且在一個關聯中，必須要具有「唯一性」的屬性來當作「鍵 (Key)」。

例如：在關聯式資料庫中，常見的鍵(Key)可分為：超鍵、候選鍵、主

鍵及



屬性(Attribute)

【定義】用來描述**實體**的**性質(Property)**。

【例如】

學號、**姓名**、「**性別**」都是用來描述**學生實體**的**性質**，並且每一個屬性一定要有一個**定義域(Domain)**(亦即**資料類型**、**範圍大小**等)。其中，

「**性別**」屬性的內含值，必須是「**男生**」或「**女生**」，而不能超出定義域(Domain)的合法值群。

【分類】

1. **簡單**屬性(Simple Attribute)
2. **複合**屬性(Composite Attribute)
3. **衍生**屬性(Derived Attribute)

1. 簡單屬性(Simple Attribute)

【定義】已經無法再繼續切割成其他有意義的單位，亦即該屬性為基元值(Atomic Value)。

【例如】「學號」屬性便是「簡單屬性」。

「地址」就不算是簡單屬性，因為它還可以再切割成區域號碼、縣市、鄉鎮、路、巷、弄、號等小單位。

2. 複合屬性(Composite Attribute)

【定義】由兩個或兩個以上的其他屬性的值所組成。

【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。

【適用時機】戶政事務查詢，房屋仲介網站...

那些屬性是屬於「複合屬性」呢？必須要視需求而定。一般使用者在設定客戶資料表或學生資料表時，「地址」屬性是視為「簡單屬性」。

【優點】大量查詢時，較快速。

where 地址 Like '*苓雅區*' → 速度較慢

where 區域 = '苓雅區' → 速度較快

3. 衍生屬性(Derived Attribute)

【定義】 `SELECT DATEDIFF("YYYY",#1971/10/9#,NOW());`

指可以經由某種方式的計算或推論而獲得的。

【例如1】 「年齡」屬性便屬於「衍生屬性」。

以實際的年齡為例，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。

年齡 = 目前的系統時間 - 生日

衍生屬性

題目：年齡=目前的系統時間-生日

請輸入您的生日(如：西元年1971)

1991

計算 系統時間

輸出「年齡」

您今年 20 歲

下一題

```
Dim Age As Integer
```

```
Age = Year(Now()) - Val(TextBox1.Text)
```

```
Label3.Text = "您今年 " & Age & " 歲"
```

程式名稱： `ch2\ch2-2\ch2-2.sln`

超鍵(Super Key)

基本上，我們會在**每一個資料表**中，**選出一個具有唯一性的欄位**來當作「**主鍵**」，但是，在一個資料表中，**如果找不到具有唯一性的欄位**時，我們也可以選出**兩個或兩個以上的欄位**組合起來，以作為唯一識別資料的欄位。

【定義】

是指在一個資料表中，選出**兩個或兩個以上的欄位**組合起來，以作為**唯一識別資料的欄位**，因此，我們可以稱這種組合出來的欄位，就是「**超鍵**」。在一個**關聯(表格)**中**至少有一個「超鍵」**，就是**所有屬性的集合**。

【例如】

以「**學生資料表**」為例，若是全班的學生姓名中，若有人**同名同姓**時(**重複**)，則我們可以**搭配學生的學號**，讓「學生的學號」與「學生的姓名」兩欄位結合起來(亦即「**學號 + 姓名**」)來**產生新的鍵**。所以，{ **姓名，學號** } 是一個**超鍵**。因為**不可能**有兩個學生的**姓名與學號**皆相同。{ **身份證字號** } 也是一個**超鍵**。

學號	姓名
S0001	一心
S0002	一心
S0003	三多
S0004	四維
S0005	一心

同名同姓

設定 { 姓名，學號 } 為超鍵

欄位名稱	資料類型
學號	文字
姓名	文字

【同理】

{ 姓名, 學號, 身份證字號, 年齡, 系別 }, { 姓名, 學號, 身份證字號, 年齡 } 也都是**超鍵**。因為它可以造成**唯一性**的限制。

【分析】

1. { 年齡 } 或 { 姓名 } 都不是「超鍵」。
2. 最大的「超鍵」是所有屬性的集合；

學生資料表		
	欄位名稱	資料類型
🔑▶	學號	文字
🔑	姓名	文字
🔑	身份證字號	文字
🔑	年齡	數字
🔑	系別	文字

最小的「超鍵」則是關聯的**主鍵**。

學生資料表		
	欄位名稱	資料類型
🔑▶	學號	文字
	姓名	文字
	身份證字號	文字
	年齡	數字
	系別	文字

主鍵(Primary Key)

在關聯式資料庫模型中，將每一個資料表視為一個「實體」，而每一個實體利用「屬性」描述之，而這些屬性就稱為「鍵值」。其中用來識別資料表中記錄的唯一的鍵值，稱為「主鍵」。

【定義】

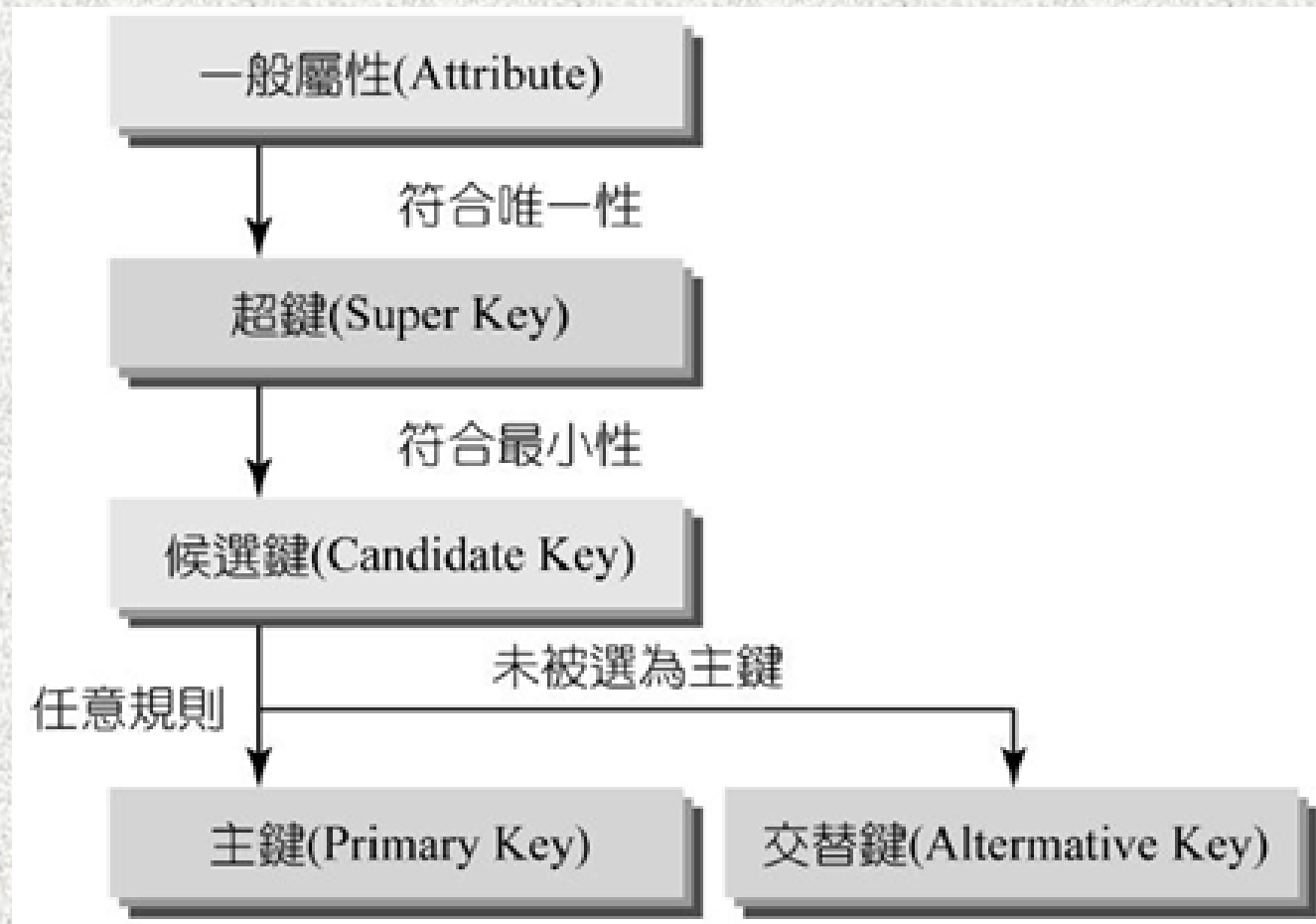
1. 從候選鍵中選擇一個用來唯一識別值組（記錄）的鍵，稱為主鍵。
2. 在關聯網要裡，我們會在主鍵的屬性名稱加一個底線。
3. 在一個關聯中，只有一個主鍵，若候選鍵未被選為主鍵時，則稱為「交替鍵(Alternate Key)」。
4. 主鍵之鍵值不可為虛值(Null Value)。
5. 在建立資料表時一般都是以「P.K.」來代表主鍵。

【舉例】學生資料表 (學號 , 姓名 , 生日 , 身分證字號 , 科系)

(1) 候選鍵 : (學號) 或 (身分證字號)

(2) 主鍵 : 學號

(3) 交替鍵 : 身分證字號



【如何挑選主鍵?】

基本上，我們要從**多個鍵值**中挑選「主鍵」時，會依循以下**三個原則**：

1. 固定不會再變更的值

在挑選「主鍵」時，必須要找永遠不會被變更的欄位，否則會增加爾後的管理和維護資料的困難度與複雜性。

例如：「學號」與「身份證字號」在決定之後，幾乎不會再改變。

2. 單一的屬性

在一個資料表中，最好只選取「單一屬性」的候選鍵作為主鍵，因為可以**節省記憶體空間及提高執行效率**。

例如：{姓名+學號}與{學號}，雖然二者都具有唯一性，但是後者{學號}是單一屬性。

3. 不可以為空值或重覆

依照「關聯式資料完整性規則」，主鍵的鍵值**不可以重覆**，也**不可以為空值(NULL)**。

例如：{姓名}欄位就不適合當作主鍵欄位。因為可能會重複。

複合鍵(Composite Key)

【定義】是指資料表中的**主鍵**，是由**兩個或兩個欄位以上**所組成，這種主鍵稱為**複合鍵(Composite Key)**。

【使用時機】

當表格中**某一欄位的值無法區分資料記錄**時，可以使用這種方法。

【例如】

在表2-2a中「縣市」的欄位值有重複，無法區分出每一筆記錄，所以「**縣市**」欄位**不能當作主鍵欄位**。因此，必須要把「**縣市**」與「**區域**」**兩個欄位組合在一起**，當作**主鍵欄位**。如表2-2b所示。

表 2.2a

	縣市	區域
重複	高雄市	三民區
	高雄市	苓雅區
重複	台北市	大安區
	台北市	信義區

複合鍵

表 2.2b

縣市	區域
高雄市	三民區
高雄市	苓雅區
台北市	大安區
台北市	信義區

候選鍵(Candidate Key)

【定義】**候選鍵**就是**主鍵**的候選人，並且也是關聯表的屬性子集所組成。

【條件】

一個屬性(欄位)是要成為**候選鍵**，則必須同時要符合下列**兩項條件**：

1.具有唯一性

是指在一個關聯表中，用來**唯一識別資料記錄的欄位**。

例如：**超鍵(Super Key)**。但可以由**多個欄位組合{縣市+區域}**而成。

2.具有最小性

是指除了符合「**唯一性**」的條件之外，還必須要在該「屬性子集」中**移除**任一個屬性之後，不再符合唯一性。亦即**鍵值欄位個數為最小**。

例如：**{縣市+區域}**組合成來符合「**唯一性**」的條件。並且在移除任一個屬性**{區域}**之後，**{縣市}**不再符合唯一性。因此，**{縣市}**

【特性】

1. 候選鍵可以**唯一識別值組**（記錄），大部份關聯都**只有一個候選鍵**。
2. 若**候選鍵**只包含一個屬性時，稱為**簡單(simple)候選鍵**。

例如：{學號}

若包含**兩個或兩個以上屬性**時，稱為**複合(composite)候選鍵**。

例如：{縣市+區域}

【舉例1】假設現在有一個「學生資料表」，其相關的欄位如下所示：

學生資料表 (學號, 姓名, 生日, 身分證字號, 科系)

請找出此資料表中的兩個「候選鍵」。

【解答】第一個候選鍵

(1)找出「具有唯一性」的欄位

{學號+姓名}共同組成時，滿足唯一性。

(2)檢查是否「具有最小性」

但是{學號+姓名}不滿足最小性，因為在移去「姓名」屬性之後，

「學號」仍然具有唯一性。

因此，我們必須要縮減為最小欄位為{學號}，所以找到第一個候選鍵。

學生資料表 (學號 , 姓名 , 生日 , 身分證字號 , 科系)

【解答】第二個候選鍵

(1) 找出「具有唯一性」的欄位

{身分證字號+科系}共同組成時，滿足唯一性。

(2) 檢查是否「具有最小性」

但是{身分證字號+科系}不滿足最小性，因為在移去「科系」屬性之後，「身分證字號」仍然具有唯一性。

因此，我們必須要縮減為最小欄位為{身分證字號}，
所以找到第二個候選鍵。

所以，{學號}或{身分證字號}皆為「候選鍵」

【舉例2】假設現在有一個「通訊錄資料表」，其相關的欄位如下所示：

通訊錄資料表 (姓名，生日，電話，地址)

請找出此資料表中的一個「候選鍵」。

【解答】

(1)找出「具有唯一性」的欄位

{姓名+生日}共同組成時，滿足唯一性。

(2)檢查是否「具有最小性」

並且{姓名+生日}也滿足最小性，因為在移去「生日」屬性之後，「姓名」就不具有唯一性。

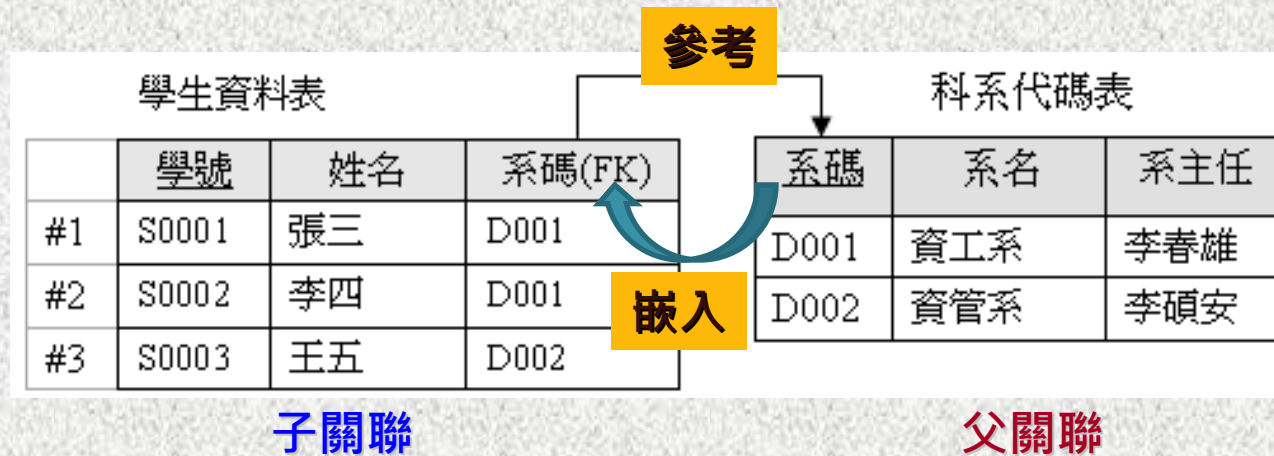
因此，{姓名+生日}兩個欄位組合，缺一不可，所以這種候選鍵又稱為「複合式候選鍵」。

所以，{姓名+生日}為「複合式候選鍵」

【外來鍵的特性】

1. 「子關聯」的外鍵必須對應「父關聯」的主鍵。
2. 外鍵是用來建立「子關聯」與「父關聯」的連結關係。

例如：張三同學可以找到對應的系主任。



說明：在SQL語言中，通常是「主鍵值=外來鍵值」當作條件式

例如：在SELECT之WHERE子句中撰寫如下：

```
學生資料表.系碼=科系代碼表.系碼
```

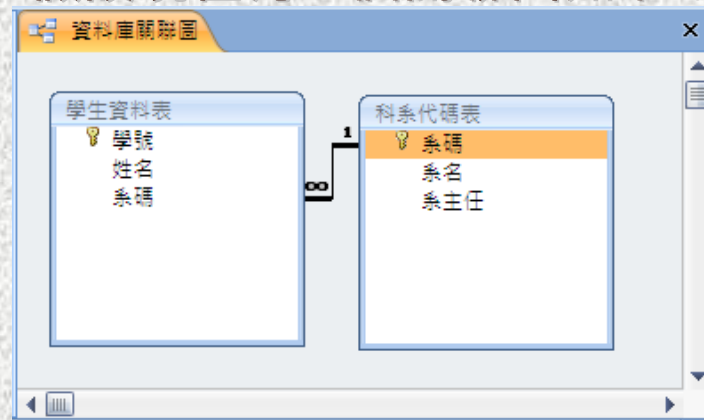
說明：以上SQL指令是用來聯結「學生資料表」和「科系代碼表」兩個資料表。

【外來鍵的特性】 <續>

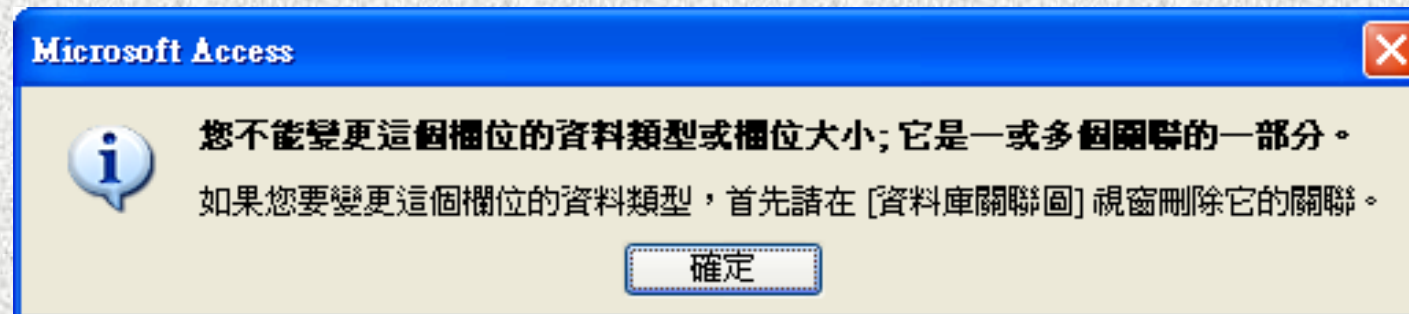
3.外來鍵和「父關聯」的主鍵欄位必須要具有相同定義域，亦即相同的資料型態和欄位長度，但名稱則可以不相同。

【舉例1】相同的資料型態和欄位長度

假設現在有一個關聯圖如下：

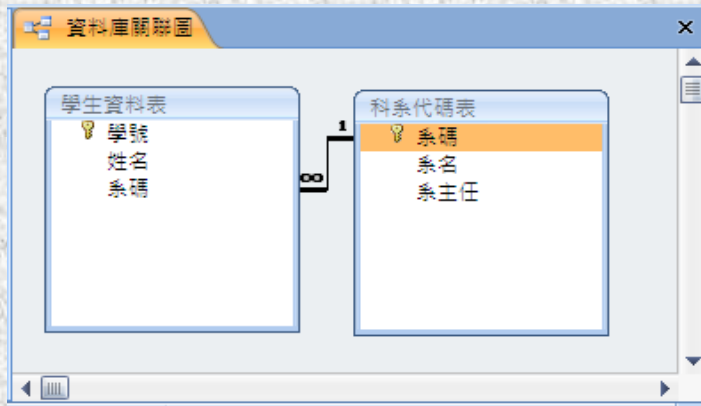


其中，「科系代碼表」的「系碼」欄位的資料類型為「文字」，現在欲改為「備忘」的資料類型，則會出現以下的錯誤產生：

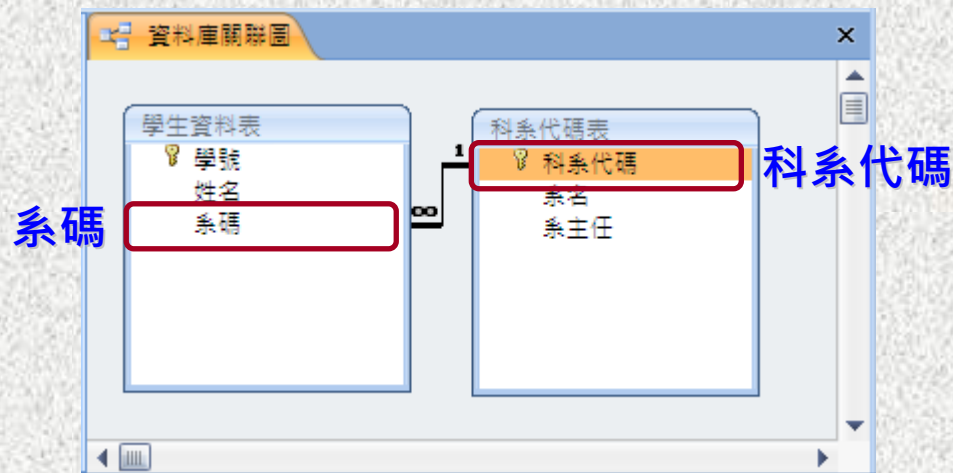


【舉例2】外來鍵和「父關聯」的主鍵欄位名稱則可以不相同

假設現在有一個關聯圖如下：



其中，「科系代碼表」的「系碼」欄位名稱，現在欲改為「科系代碼」欄位名稱，則是可以的。如下圖所示：

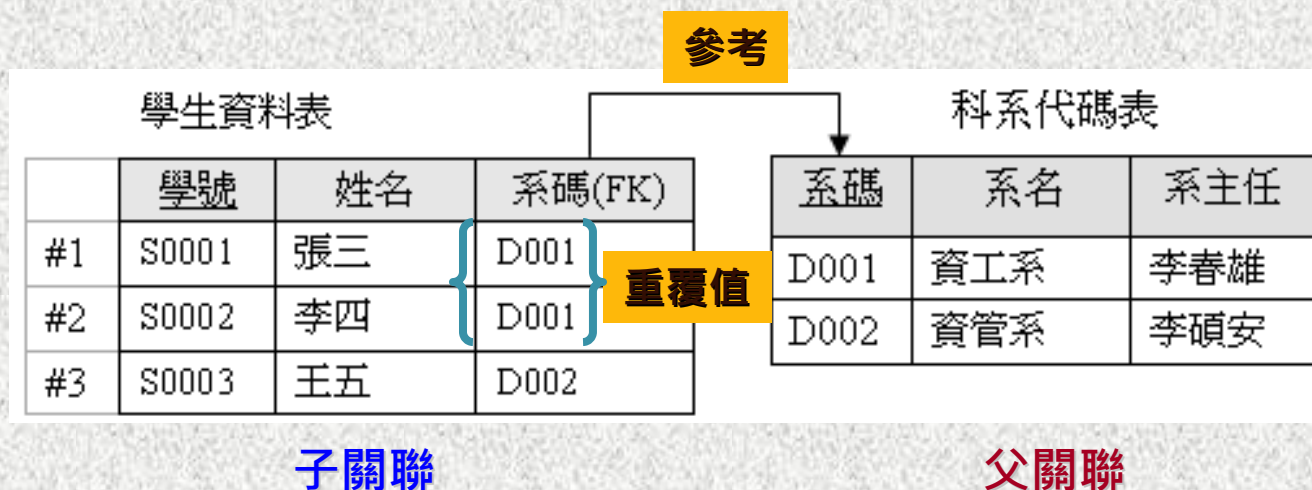


註：因此，我們可以清楚得知，「子關聯」的外來鍵參考「父關聯」

【外來鍵的特性】 <續>

4.外來鍵的欄位值可以是**重覆值**或**空值(NULL)**。

(1)「重覆值」的例子

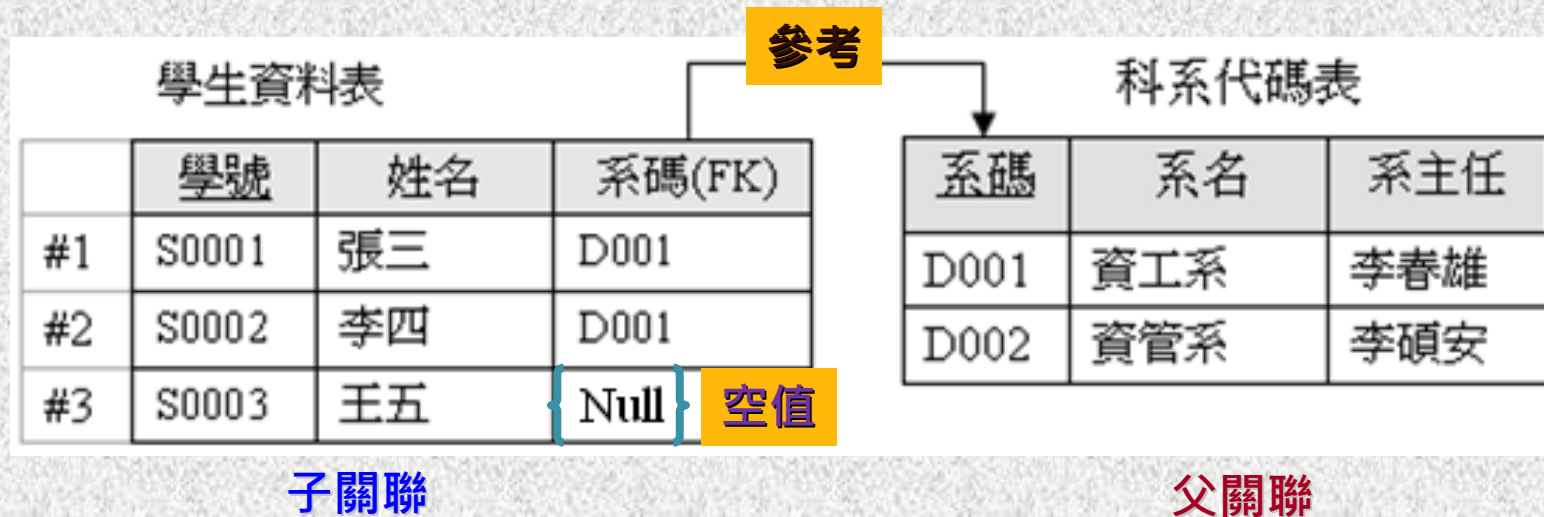


說明：在上表中，代表張三與李四都是就讀「資工系」

【外來鍵的特性】 <續>

4.外來鍵的欄位值可以是**重覆值**或**空值(NULL)**。

(2)空值(NULL)的例子



說明：在上表中，代表王五尚未決定要就讀那一個科系

關聯式資料庫的種類

假設現在有甲與乙兩個資料表，其「關聯式資料庫」中資料表的關聯種類可以分為下列三種：

1. 一對一的關聯(1 : 1)：

甲資料表中的一筆記錄，只能對應到乙資料表中的一筆記錄，並且乙資料表中的一筆記錄，只能對應到甲資料表中的一筆記錄。

2. 一對多的關聯(1 : M)：

甲資料表中的一筆記錄，可以對應到乙資料表中的多筆記錄，但是乙資料表中的一筆記錄，卻只能對應到甲資料表中的一筆記錄。

3. 多對多的關聯(M : N)：

甲資料表中的一筆記錄，能夠對應到乙資料表中的多筆記錄，並且乙資料表中的一筆記錄，也能夠對應到甲資料表中的多筆記錄。

一對一關聯(1 : 1)

【定義】

假設現在有甲與乙兩個資料表，在一對一關聯中，甲資料表中的一筆記錄，只能對應到乙資料表中的一筆記錄，並且乙資料表中的一筆記錄，只能對應到甲資料表中的一筆記錄。

【舉例】

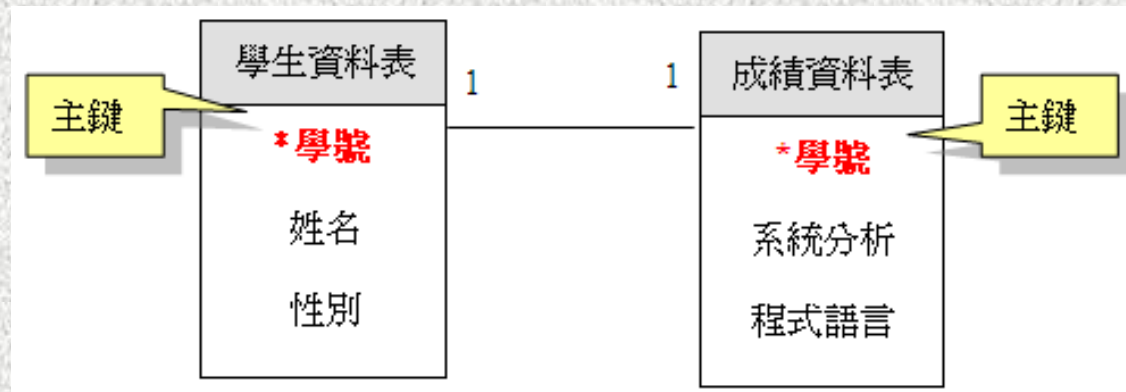
以「成績處理系統」為例，當兩個資料表之間做一對一的關聯時，表示「學生資料表」中的每一筆記錄，只能對應到「成績資料表」的一筆記錄，而且「成績資料表」的每一筆記錄，也只能對應到「學生資料表」的一筆記錄，這就是所謂的1對1關聯。

【適用時機】

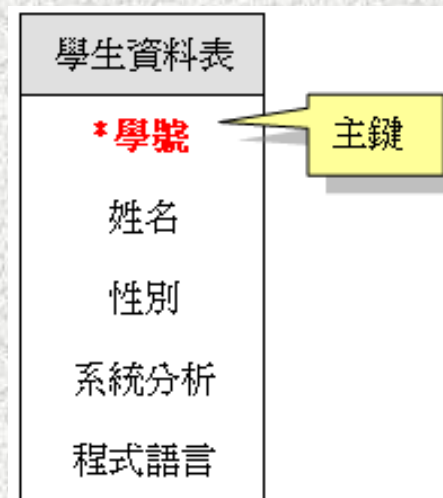
通常是基於安全上的考量(資料保密因素)，將某一部份的欄位分割到另一個資料表中。

【一對一的關聯圖】

在下圖中，「學生資料表」與「成績資料表」是一對一的關係。因此，「學生資料表」的**主鍵**必須**對應**「成績資料表」的**主鍵**，才能設定為**1:1**的關聯圖。



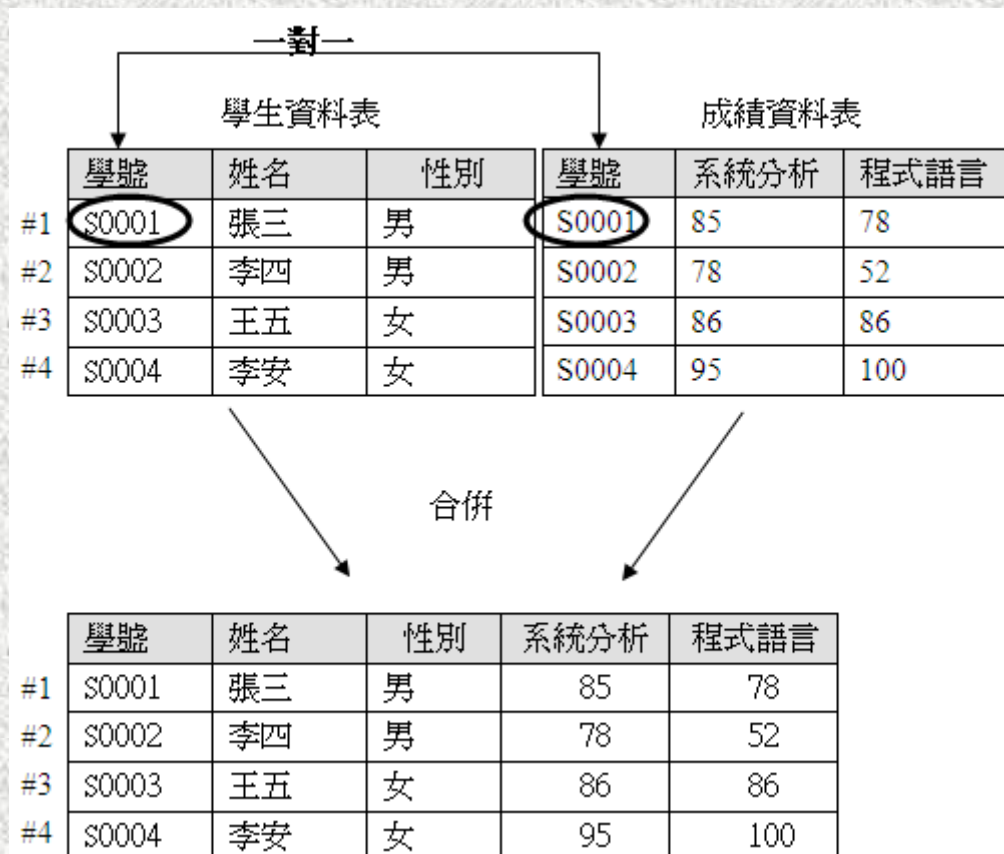
在實務上，我們也可以將**這兩個資料表合併**成**一個資料表**，其合併結果如下圖所示。



【一對一的實例】

欲將「學生資料表」與「成績資料表」的這兩個資料表合併成一個資料表時，必須要先完成以下兩個條件，否則就無法進行「合併」：

- 1.先檢查「學生資料表」中「學號」欄位值是否與「成績資料表」中「學號」欄位值完全相同。
- 2.再建立「1:1的資料庫關聯圖」，下圖所示：



一對多關聯(1 : M)

【定義】

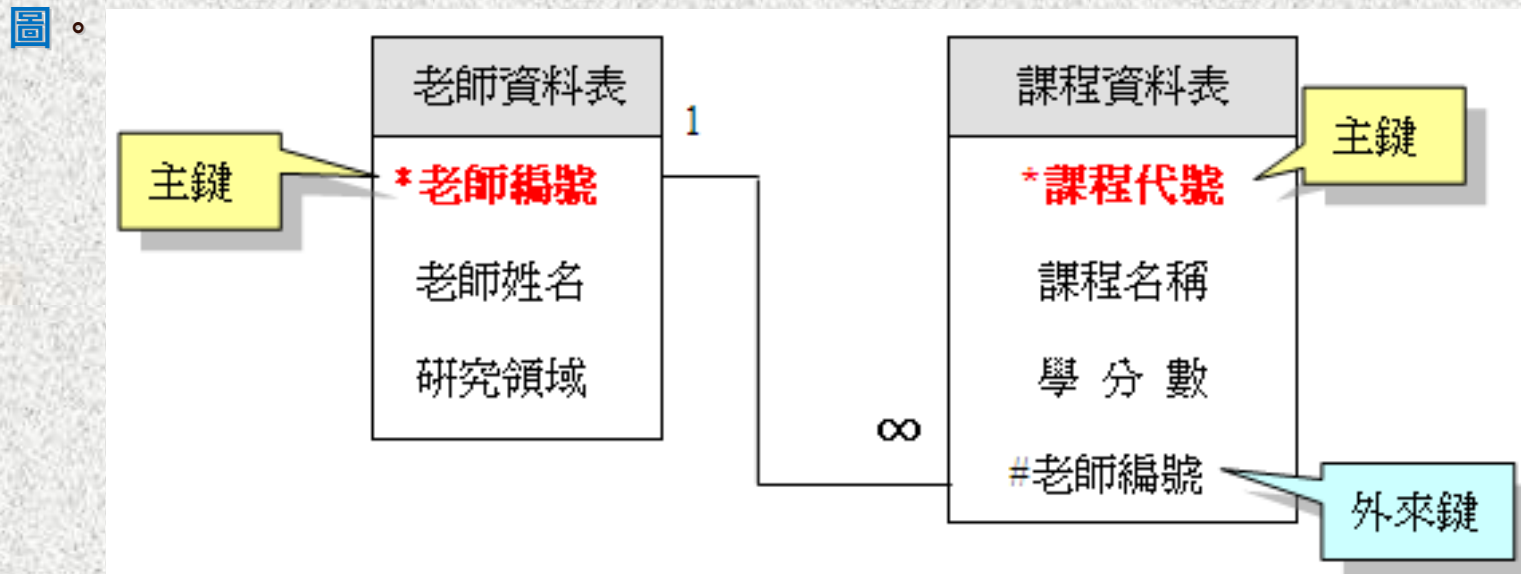
假設現在有甲與乙兩個資料表，在**一對多關聯**中，**甲資料表中的一筆記錄**，可以對應到**乙資料表中的多筆記錄**，但是**乙資料表中的一筆記錄**，卻只能對應到**甲資料表中的一筆記錄**。

【舉例】

以「數位學習系統」為例，當兩個資料表之間做**一對多的關聯**時，表示「**老師資料表**」中的**每一筆記錄**，可以對應到「**課程資料表**」中的**多筆記錄**，但「**課程資料表**」的**每一筆記錄**，只能對應到「**老師資料表**」中的一筆記錄，這就是所謂的一對多關聯，這種方式是最常被使用。

【一對多的關聯圖】

在下圖中，「老師資料表」與「課程資料表」是一對多的關係。因此，「老師資料表」的**主鍵**必須**對應**「課程資料表」的**外來鍵**，才能設定為**1:M**的關聯圖。



註：「*」代表該欄位為**主鍵**，「#」代表該欄位為**外來鍵**。

【一對多的實例】

我們建立兩個資料表，分別為「老師資料表」與「課程資料表」，此時，我們可以了解「老師資料表」中的一筆記錄(T0001)，可以對應到「課程資料表」中的多筆記錄(C001,C002,C003)，但是「課程資料表」中的一筆記錄，卻只能對應到「老師資料表」中的一筆記錄。如下圖所示。



多對多關聯(M:N)

【定義】

假設現在有甲與乙兩個資料表，在**多對多關聯**中，**甲資料表中的一筆記錄**，能夠對應到**乙資料表中的多筆記錄**，並且**乙資料表中的一筆記錄**，也能夠對應到**甲資料表中的多筆記錄**。

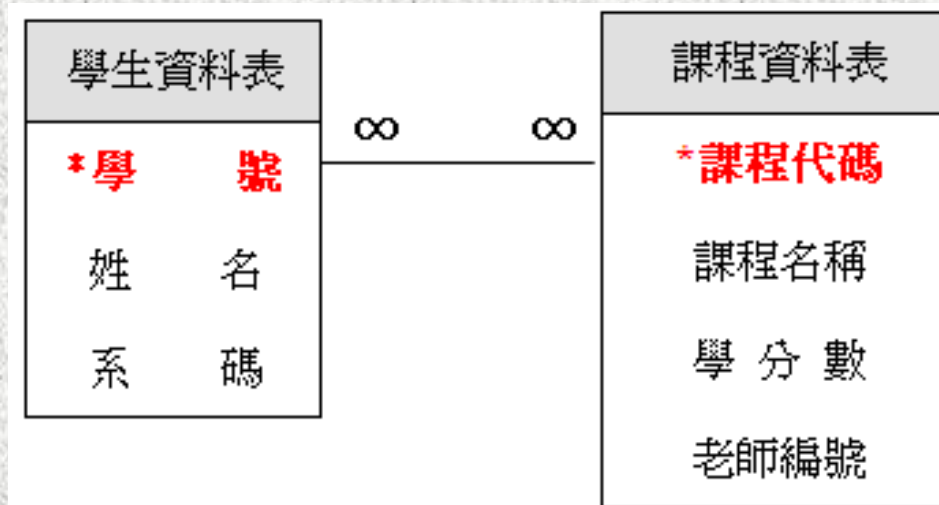
【舉例】

以「**選課系統**」為例，當兩個資料表之間做**多對多的關聯**時，表示「**學生資料表**」中的每一筆記錄，可以對應到「**課程資料表**」中的多筆記錄，並且「**課程資料表**」中的每一筆記錄，也能夠對應到「**學生資料表**」中的多筆記錄，這就是所謂的多對多關聯。

【多對多的關聯圖】

雖然，一對多關聯是最常先的一種關聯性，但是在實務上，「多對多關聯」的情況也不少，也就是說由兩個資料表(實體)呈現多對多的關聯。

例如：「學生資料表」與「課程資料表」。如下圖所示。



在上圖中表示：每一位學生可以選修多門課程，並且每一門課程也可以被多位學生來選修。

【兩個資料表多對多關聯之問題】

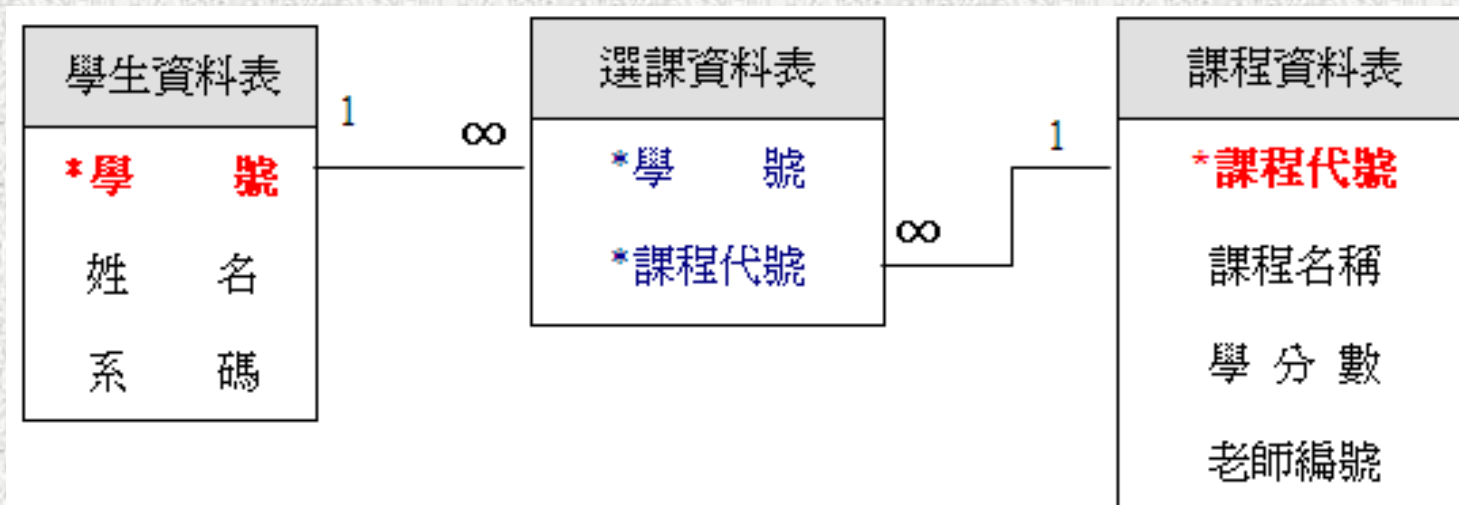
在實務上多對多關聯如果只有兩個資料表來建置，難度較高，並且容易出問題。

【解決方法】

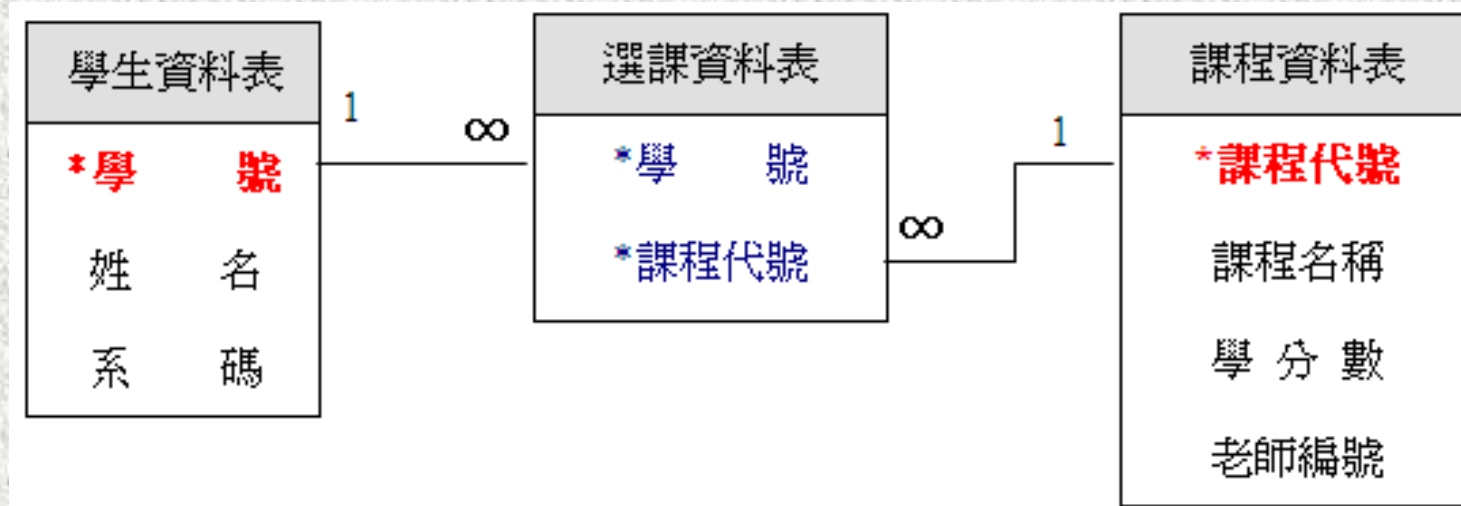
利用「三個資料表」來建置「多對多關聯」，也就是說，在**原來的兩個資料表**之間再加入一個「**聯合資料表 (Junction Table)**」，使他們可以順利處理多對多的關聯。其中，**聯合資料表 (Junction Table)** 中的**主索引鍵(複合主鍵)**是由**資料表A(學生資料表)**和**資料表B(課程資料表)**兩者的**主鍵**所組成。

例如：在「學生資料表」與「課程資料表」之間再加入第三個資料表「選課資料表」，如下圖所示。

聯合資料表



聯合資料表

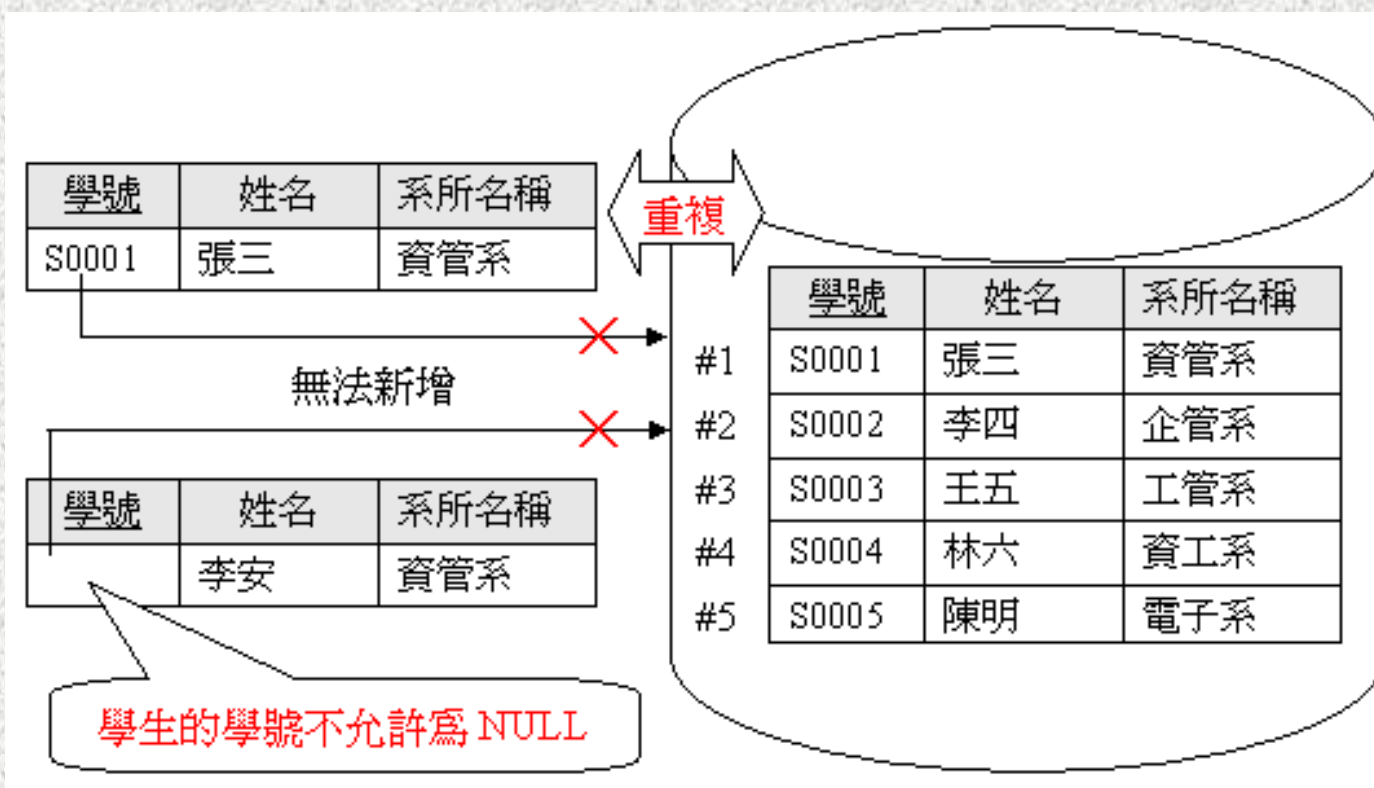


說明：

1. 在「學生資料表」與「選課資料表」的關係是以一對多。
2. 在「課程資料表」與「選課資料表」的關係是以一對多。
3. 藉由「選課資料表」的使用，使「學生資料表」與「課程資料表」關係變成多對多的關聯式，亦即每一位學生可以選修一門以上的課程並且每一門課程也可以被多位同學的選修。
4. 以資料表(Table)之方式組成關聯，將這些關聯組合起來，即形成一個關聯式資料庫。

資料庫之完整性規則

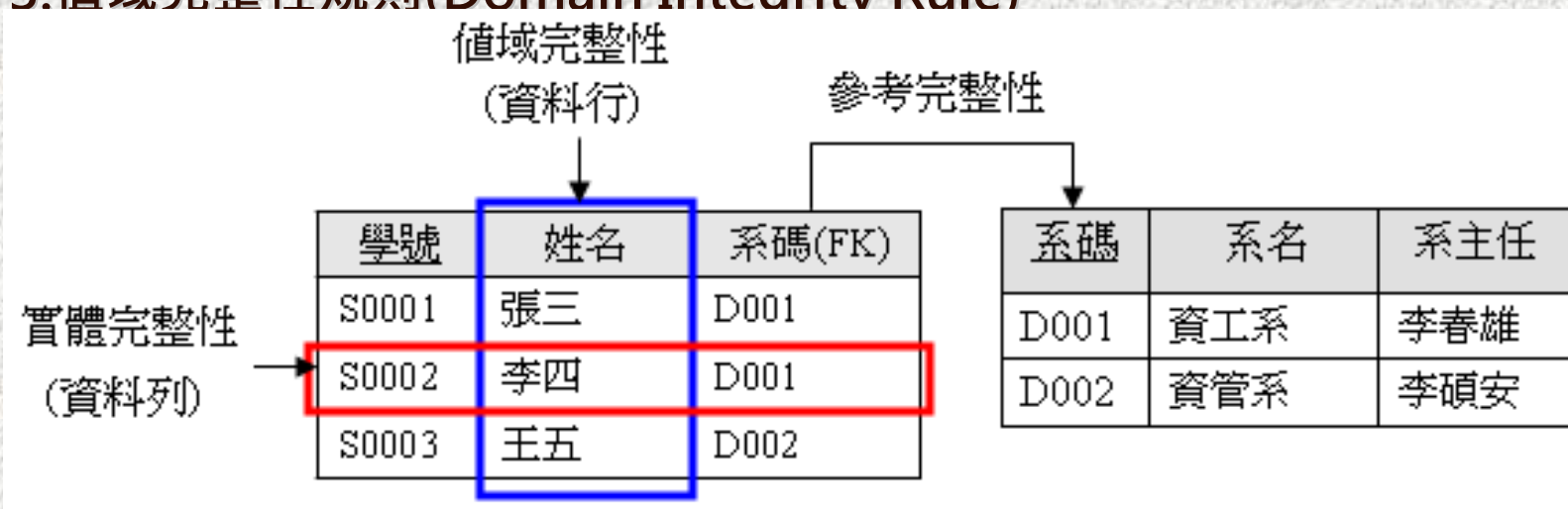
完整性規則(Integrity Rules)是用來確保資料的**一致性與完整性**，以**避免**資料在經過**新增、修改及刪除**等運算之後，而產生的**異常現象**。亦即**避免**使用者將**錯誤或不合法的資料值**存入資料庫中。如下圖所示：



【三種完整性規則】

關聯式資料模式的「完整性規則」，有下列三種：如下圖所示。

- 1.實體完整性規則(Entity Integrity Rule)
- 2.參考完整性規則(Referential Integrity Rule)
- 3.值域完整性規則(Domain Intearitv Rule)



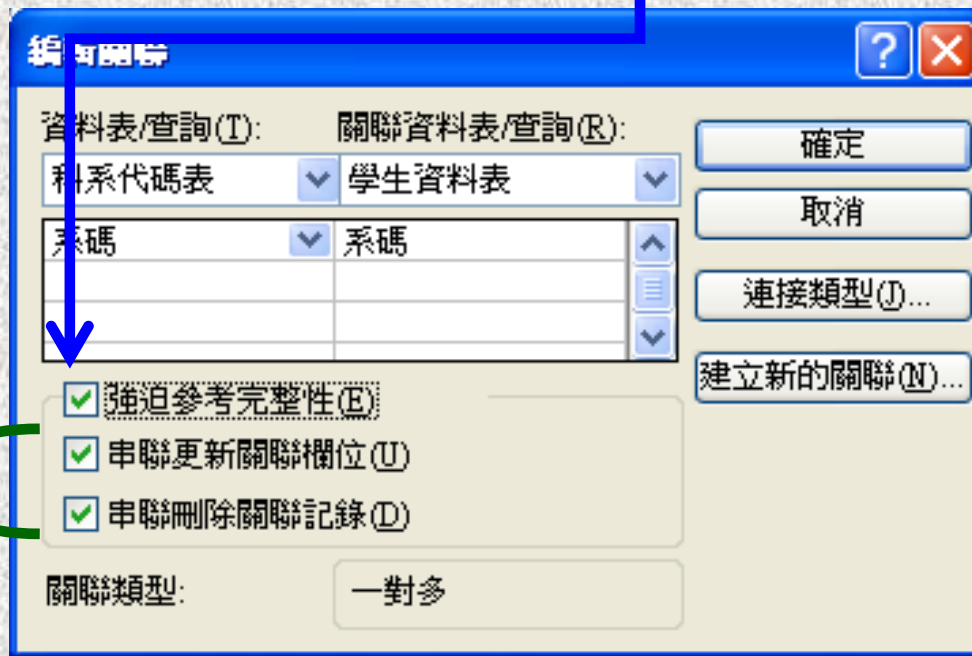
學生資料表(子關聯)

科系代碼表(父關聯)

註：在關聯式資料庫中，任兩個資料表要進行關聯(參考)時，必須透過「主鍵」對應「外來鍵」才能建立，其中「主鍵」值的所在資料表稱為「父關聯」，而「外來鍵」值的所在資料表稱為「子關聯」。

【完整性的綜合分析】

我們已經學會一對一、一對多及多對多的資料庫關聯圖了，但是，你是否注意到，當我們勾選「**強迫參考完整性**」時，Access會自動啟動「**串聯更新關聯欄位**」及「**串聯刪除關聯欄位**」的功能，來讓使用者勾選呢？說明如下所示：



【完整性的綜合分析】 <續>

編輯關聯

資料表/查詢(T): 關聯資料表/查詢(R):

科系代碼表 學生資料表

系碼 系碼

「一」為主鍵 「多」為外來鍵

強迫參考完整性(E)

串聯更新關聯欄位(U)

串聯刪除關聯記錄(D)

關聯類型: 一對多

「多」為外來鍵 「一」為主鍵

外鍵參考主鍵

學生資料表

學號	姓名	系碼	
#1	S0001	一心	D001
#2	S0002	二聖	D001
#3	S0003	三多	D002
#4	S0004	四維	D002
#5	S0005	五福	D002

科系代碼表

系碼	系名	系主任	
#1	D001	資工系	李春雄
#2	D002	資管系	李碩安

主鍵(PK)

外鍵(FK)

【註】在「編輯關聯」對話方塊中，如果有勾選「強迫參考完整性」時，

則必須要具備以下條件：

- (1) 兩個資料表之間的關聯欄位的資料類型必須要相同。
- (2) 如果關聯類型是「一對多」時，則「一」的那方必須要是「主鍵」，而「多」的那方必須要是「外來鍵」。並且主鍵不可以是「自動編

1. 強迫參考完整性

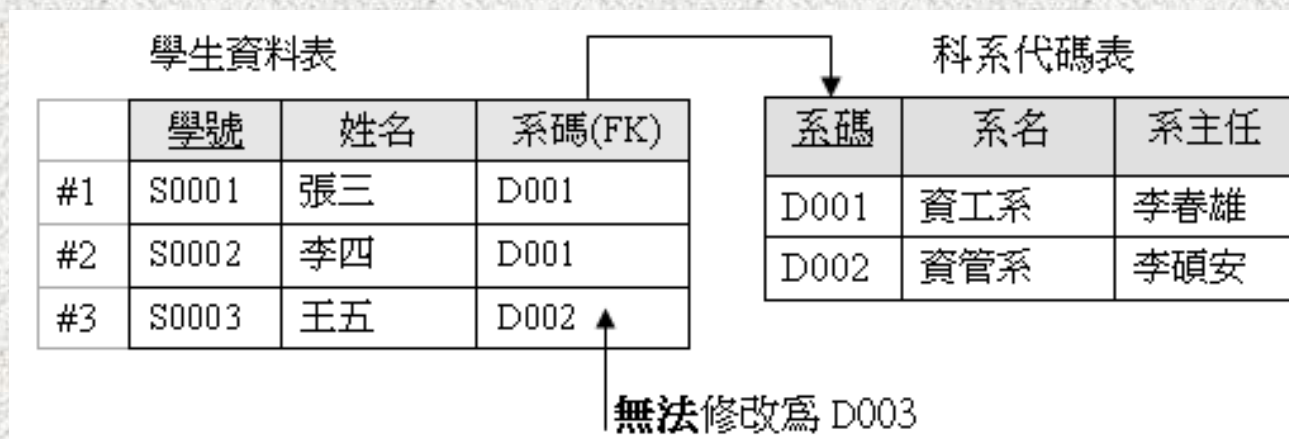
以下圖的「學生資料表」與「科系代碼表」為例：

「科系代碼表」內的「系碼」欄位為主鍵，而「學生資料表」內的「系碼」為外來鍵。

因此，「強迫參考完整性」有勾選時，則DBMS會限制使用者輸入資料是否有違反參考完整性。

何謂「參考完整性」？是指用來確保相關資料表間的資料一致性，避免因一個資料表的記錄改變時，造成另一個資料表的內容變成無效的值。

如下圖所示。



2. 串聯更新關聯欄位

在更新父關聯的主鍵時，也會同時更新子關聯的外鍵值。

例如：在下表中，欲更新「科系代碼表」中的「D001」為「A001」，也必須同時將「學生資料表」中的第二筆及第三筆記錄的「D001」修改為「A001」。

子關聯

父關聯

學生資料表				科系代碼表		
	學號	姓名	系碼(FK)	系碼	系名	系主任
#1	S0001	張三	D001	D001	資工系	李春雄
#2	S0002	李四	D001	D002	資管系	李碩安
#3	S0003	王五	D002			

實體完整性規則...針對主鍵

【定義】

每一個關聯表中的值組都必須是**可以識別的**，因此，**主鍵**必須要**具有唯一性**，並且**主鍵不可重複**或為**空值(NULL)**。否則，就無法唯一識別某一記錄（值組）。

【特性】

1. **實體**必須是**可區別的(Distinguishable)**。
2. **主鍵值未知**代表是一個**不確定的實體**，不能存放在資料關聯中。

如下圖所示：

	學號	姓名	系所名稱
#1	S0001	張三	資管系
#2	S0002	李四	企管系
#3	S0003	王五	工管系
#4	S0004	林六	資工系
#5	NULL		
↙ 無法新增			
#5	NULL	陳生	會計系

3.實體完整性規則只適用於**基本關聯(Base Relation)**，不考慮**視界(View)**。

(1)基本關聯(Base Relation)

真正存放資料的**具名關聯**，是透過SQL的**Create Table**敘述來建立。

基本關聯對應於**ANSI/SPARC**的「**概念層**」。

(2)視界(View)

是一種具名的**衍生關聯**、**虛擬關聯**，定義在某些基本關聯上，本身不含任何資料。視界相對應於**ANSI/SPARC**的「**外部層**」。

4.在建立**資料表**時可以設定某欄位為**主鍵**，以確保**實體完整性和唯一性**。

5.複合主鍵 (學號與課號) 中的任何屬性值皆不可以是空值(Null)。

如下圖所示：

	<u>學號</u>	<u>課號</u>	成績
#1	S0001	C001	76
#2	S0002	C002	56
#3	NULL	C003	86
#4	S0004	NULL	70
#5	NULL	NULL	77

無法新增

說明：主鍵是由多個欄位連結而成的組合鍵，因此，每一個欄位值都不可為空值(Null)。

參考完整性規則...針對外鍵

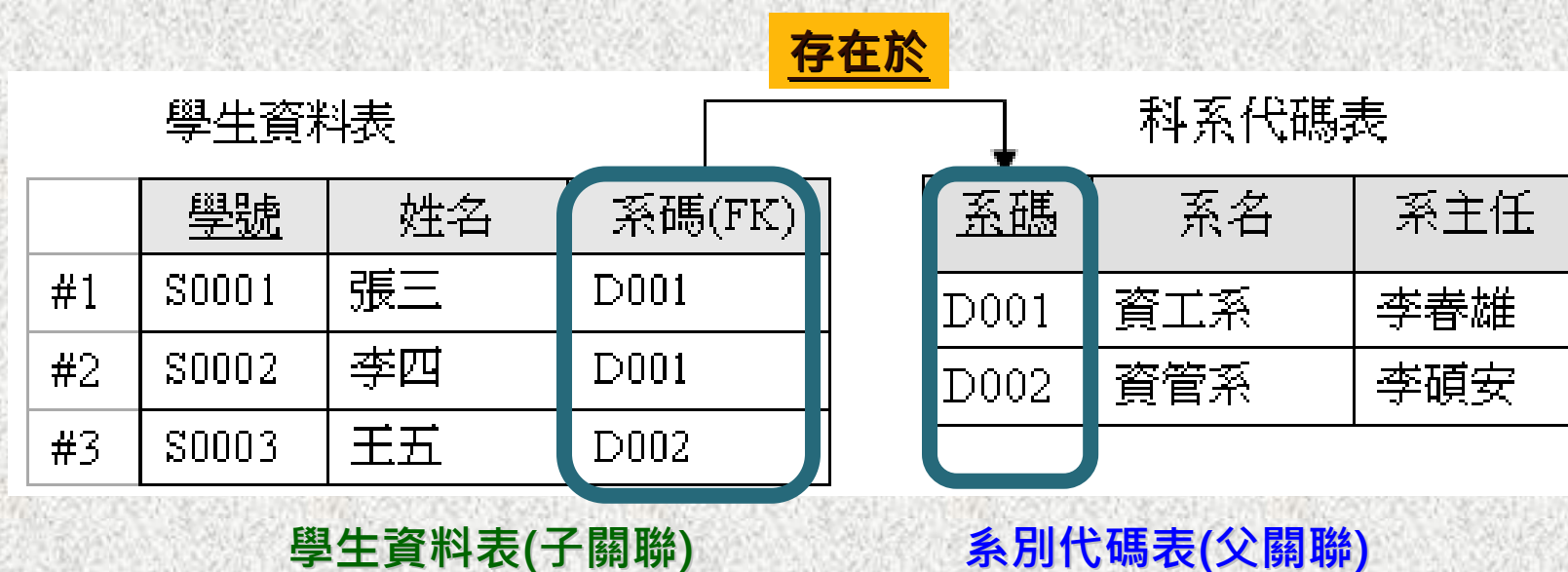
在完成建立資料庫及資料表之後，如果沒有把它們整合起來，則「學生資料表」中的外鍵(系碼)就無法與「科系代碼表」的主鍵(系碼)之間來進行關聯了，這將會導致資料庫不一致的問題。也就是違反了資料庫之「參考完整性規則」。

【定義】

是指用來確保兩個資料表之間的資料一致性，避免因一個資料表的記錄改變時，造成另一個資料表的內容變成無效的值。因此，子關聯的外來鍵(FK)的資料欄位值，一定要存在於父關聯的主鍵(PK)中的資料欄位值。

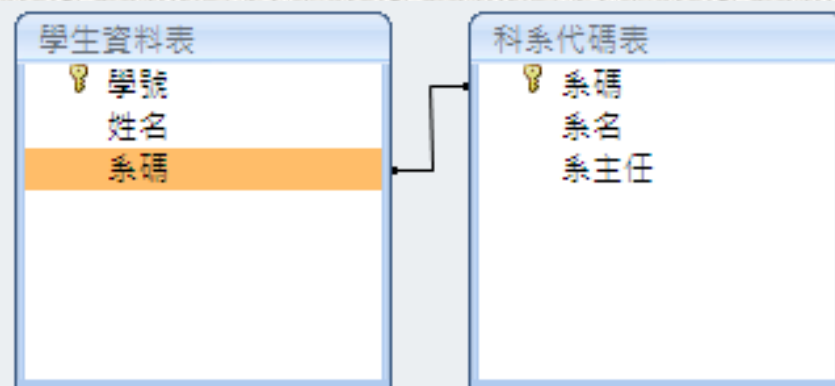
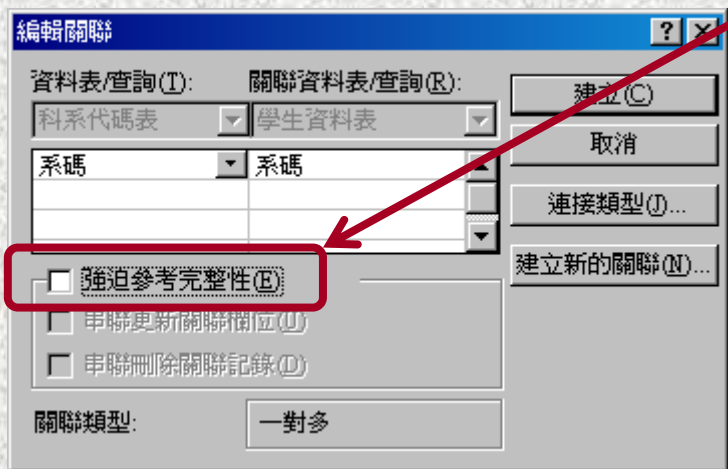
【例如】

學生資料表(子關聯)的系碼(外鍵 ; F.K.) 一定要存在於系別代碼表(父關聯)的系碼(主鍵 ; P.K.)中。如下圖所示：



【例如】尚未勾選「參考完整性規則」的情況

假設現有二個資料表，分別為「學生資料表」與「科系代碼表」，其中有一位「五福」同學欲從資工系(D001)轉為資管系(D002)，但是，在轉系的過程中，學校的校務行政系統中的「學生資料表」與「科系代碼表」尚未勾選「參考完整性規則」，如下圖所示：



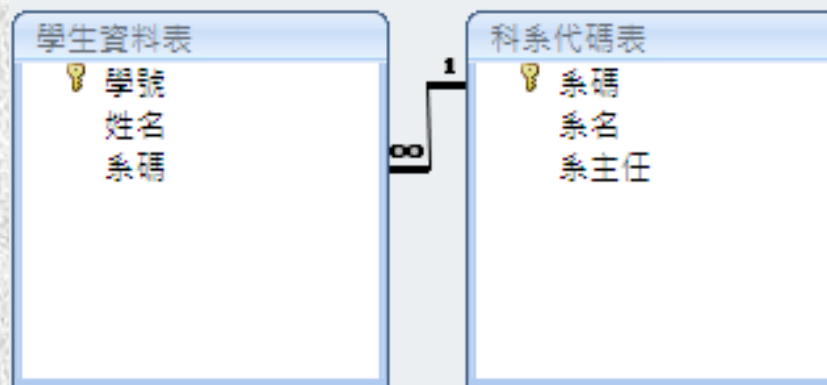
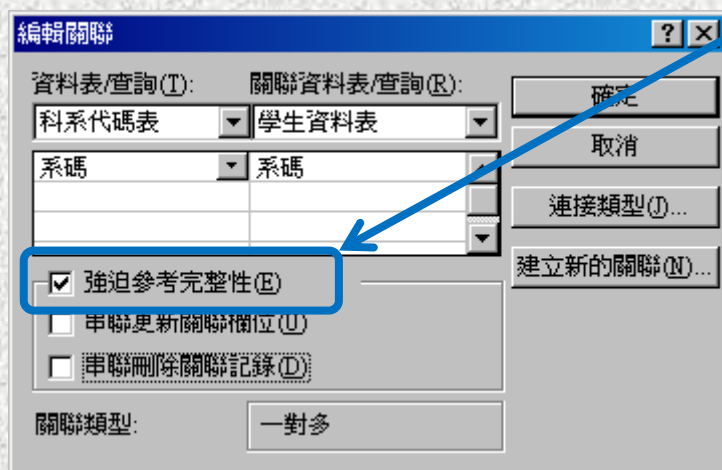
因此，行政人員假設在填入「系碼」欄位，誤填為「D003」時，DBMS就無法檢查出來，此時，將會使得系碼「D003」內容變成無效的值，以致於產生資料不一致現象。

學生資料表				科系代碼表			
	學號	姓名	系碼		系碼	系名	系主任
#1	S0001	一心	D001	#1	D001	資工系	李春雄
#2	S0002	二聖	D001	#2	D002	資管系	李碩安
#3	S0003	三多	D002				
#4	S0004	四維	D002				
#5	S0005	五福	D003				

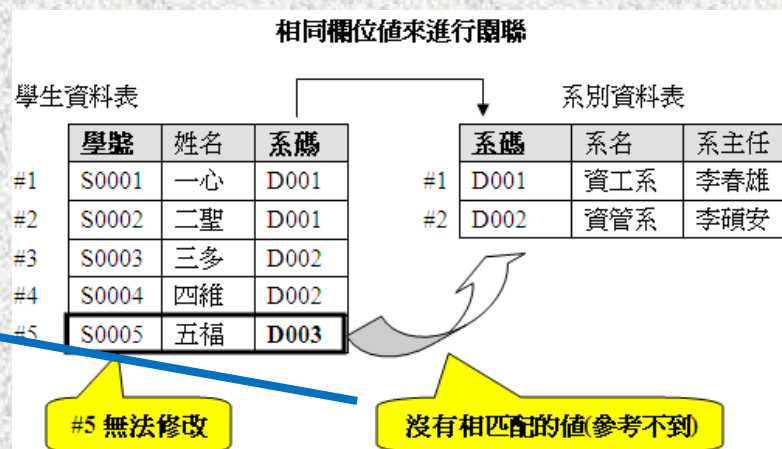
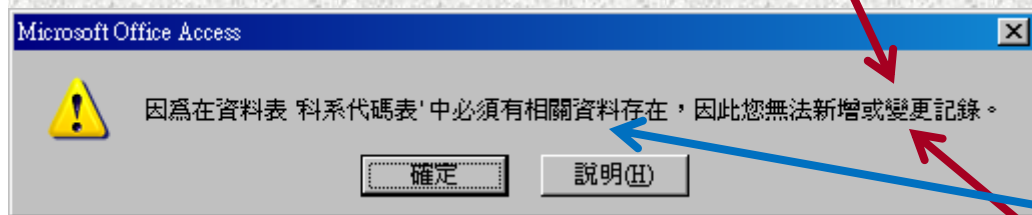
無效的值(因為無法參考到科系代碼表)

【例如】有勾選「參考完整性規則」的情況

假設現有二個資料表，分別為「學生資料表」與「科系代碼表」，其中有一位「五福」同學欲從資工系(D001)轉為資管系(D002)，並且，在轉系的過程中，學校的校務行政系統中的「學生資料表」與「系別資料表」有勾選「參考完整性規則」，如下圖所示：



因此，行政人員假設在填入「系碼」欄位，誤填為「D003」時，DBMS就可以檢查出來，此時，系碼「D003」就無法修改，因此，不會產生資料不一致現象。



【參考完整性規則的特性】

- 1.至少要有**兩個或兩個以上**的資料表才能執行「**參考完整性規則**」。
- 2.由**父關聯表**的「**主鍵**」與**子關聯表**的「**外來鍵**」的關係來建立兩資料表間資料的關聯性。
- 3.建立「參考完整性」之後，就可以**即時有效檢查**使用者的輸入值，**以避免無效的值發生**。

值域完整性規則

【定義】

是指在「**單一資料表**」中，對於**所有屬性(Attributes)**的內含值，必須來自**值域(Domain)**的合法值群中。亦即是指在「單一資料表」中，**同一資料行**中的**資料屬性**必須要**相同**。亦即**同一行的欄位**之**資料類型**要相同。

【例如】

「**性別**」屬性的內含值，必須是「**男生**」或「**女生**」，而不能超出定義域(Domain)的合法值群。

【特性】

- 1.作用在「**單一資料表**」中
- 2.「**同一資料行**」中的「**資料屬性**」必須要「**相同**」
- 3.建立資料表可以「**設定條件**」來查檢值域是否為合法值群

空值(NULL Values)

【定義】

1.空值是一種特殊記號，用以記錄目前不詳的資料值。

2.空值不是指「空白格」或「零值」。

3.空值可分為以下三種：

(1)可應用的空值(Application Null Value)：一般指目前不知道的值，

但此值確實存在。

例如：張三已婚，但其配偶欄的姓名尚未填入。

(2)不可應用的空值(Inapplicable Null Value)：目前完全沒有存在這個值。

例如：張三未婚，其配偶欄的值為空值。

(3)完全未知的空值(Totally Unknown)：完全不知道這個值是否存在。

非空值(NOT NULL)

【定義】資料行必須有正確的資料值，不可為虛值。

【例如】

在「學生資料表」中的「學號」和「姓名」兩欄位值必須確定，不可為虛值。因此，在建立資料表時就必須宣告為NOT NULL。

欄位名稱	資料類型	描述
學號	文字	
姓名	文字	
系碼	文字	

欄位名稱	欄位內容
欄位大小	8
格式	
輸入遮罩	
標題	
預設值	
驗證規則	
驗證文字	
必須有資料	是
允許零長度字串	否
索引	否
Unicode 編碼	是
輸入法模式	關
輸入法語態	無
智慧標籤	