



第3章 關聯式資料庫模型

- 3-1 資料模型的基礎
- 3-2 資料庫模型的種類
- 3-3 關聯式資料庫模型的資料結構
- 3-4 關聯式資料庫模型的完整性限制條件
- 3-5 關聯式資料庫





3-1 資料模型的基礎

- 3-1-1 資料庫設計與三層資料模型
- 3-1-2 概念資料模型
- 3-1-3 邏輯資料模型
- 3-1-4 實體資料模型





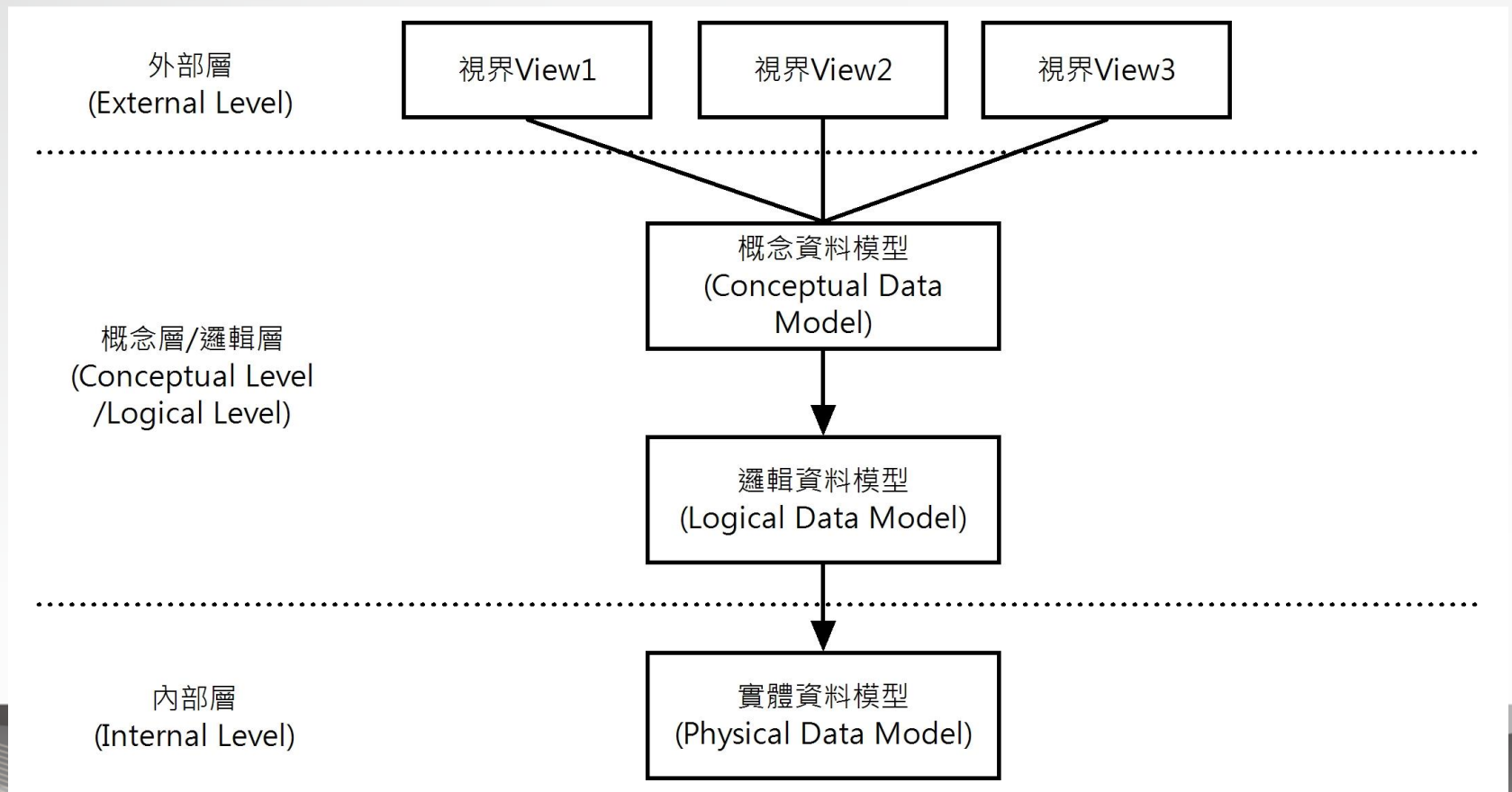
3-1 資料模型的基礎

- 「資料模型」(Data Model) 是使用一組整合觀念來描述資料與資料間的關係和資料的限制條件(是否是正確的資料)。簡單的說, 資料模型是用來描述資料庫中資料的特性。
- 一般來說, 我們使用資料庫定義語言(DDL)建立資料庫綱要, 但是, 大部分使用者並不了解DDL語法。因此, 在進行資料庫設計(Database Design)時, 為了方便與使用者溝通, 我們需要更高階的模型來描述資料庫的結構, 通常是使用圖形方式來表示, 稱為資料模型(Data Model)。



3-1-1 資料庫設計與三層資料模型 – 圖例

- 資料庫設計的資料模型也是對應ANSI/SPARC三層資料庫架構，如下圖所示：





3-1-1 資料庫設計與三層資料模型 – 圖例說明

- 概念資料模型：不涉及任何資料庫管理系統、資料庫種類和實際儲存結構建立的模型，它是使用概念方式來描述大多數使用者認知的資料。
- 邏輯資料模型：使用指定資料庫管理系統（同時也指定資料庫種類）建立的模型。例如：關聯式資料庫模型，可以用來描述ANSI/SPARC三層資料庫系統架構中，位在概念層以上的概念和外部層資料。
- 實體資料模型：針對指定資料庫管理系統建立的模型，例如：SQL Server，可以描述ANSI/SPARC三層資料庫系統架構的內部層資料。



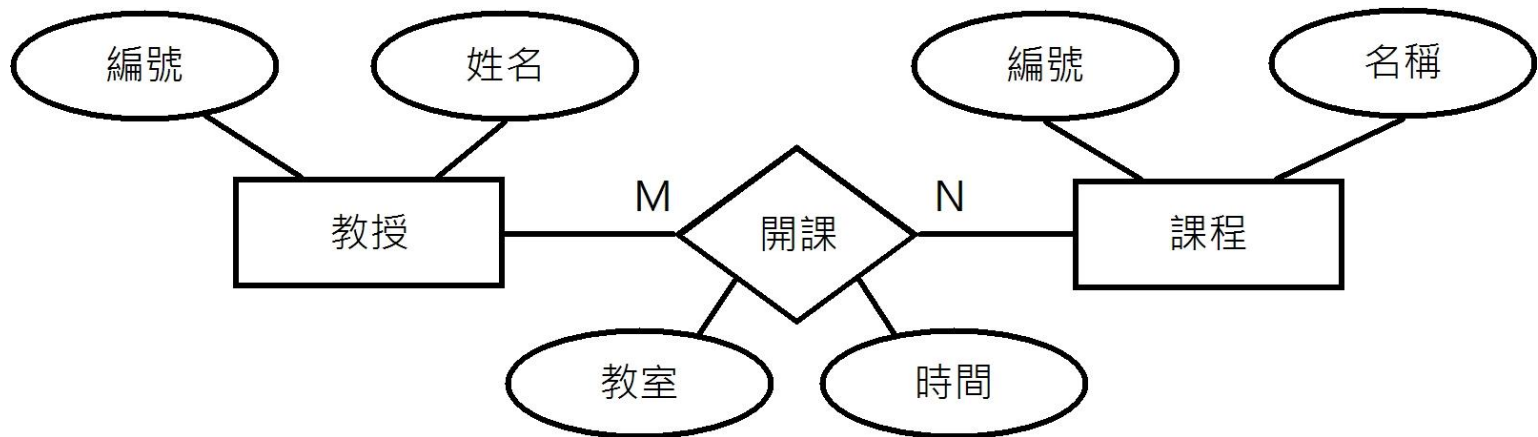
3-1-2 概念資料模型 – 說明

- 概念資料模型（Conceptual Data Model）是建立整個資料庫邏輯結構的模型，它並不涉及任何資料庫管理系統、資料庫種類、軟體和實際儲存結構。
 - 在資料庫設計需要建立概念資料模型的原因，如下所示：
 - 能夠進一步了解資料範圍。
 - 增加與最終使用者的溝通機會。
 - 開發與資料庫管理系統無關的資料模型。
 - 儘早發現資料庫設計的錯誤。



3-1-2 概念資料模型 – 圖例

- 概念資料模型最常使用「實體關聯模型」（Entity-Relationship Model；ERM）的圖形方式來建立，稱為「實體關聯圖」（Entity-Relationship Diagram；ERD）。例如：學校這學期開課資料的概念資料模型，如下圖所示：





3-1-3 邏輯資料模型 – 說明

- 邏輯資料模型（Logical Data Model）主要是針對指定資料庫管理系統建立的資料模型。第一個建立的邏輯資料模型就是關聯式模型（Relational Model），因為是針對關聯式資料庫的資料模型，為了區別，在本書稱為關聯式資料庫模型（Relational Database Model）。
- 關聯式資料庫模型是由E. F. Codd在1970年所提出，邏輯資料模型主要是由三種元素所組成，如下所示：
 - 資料結構（Data Structures）
 - 資料操作或運算（Data Manipulation或Operations）
 - 完整性限制條件（Integrity Constraints）

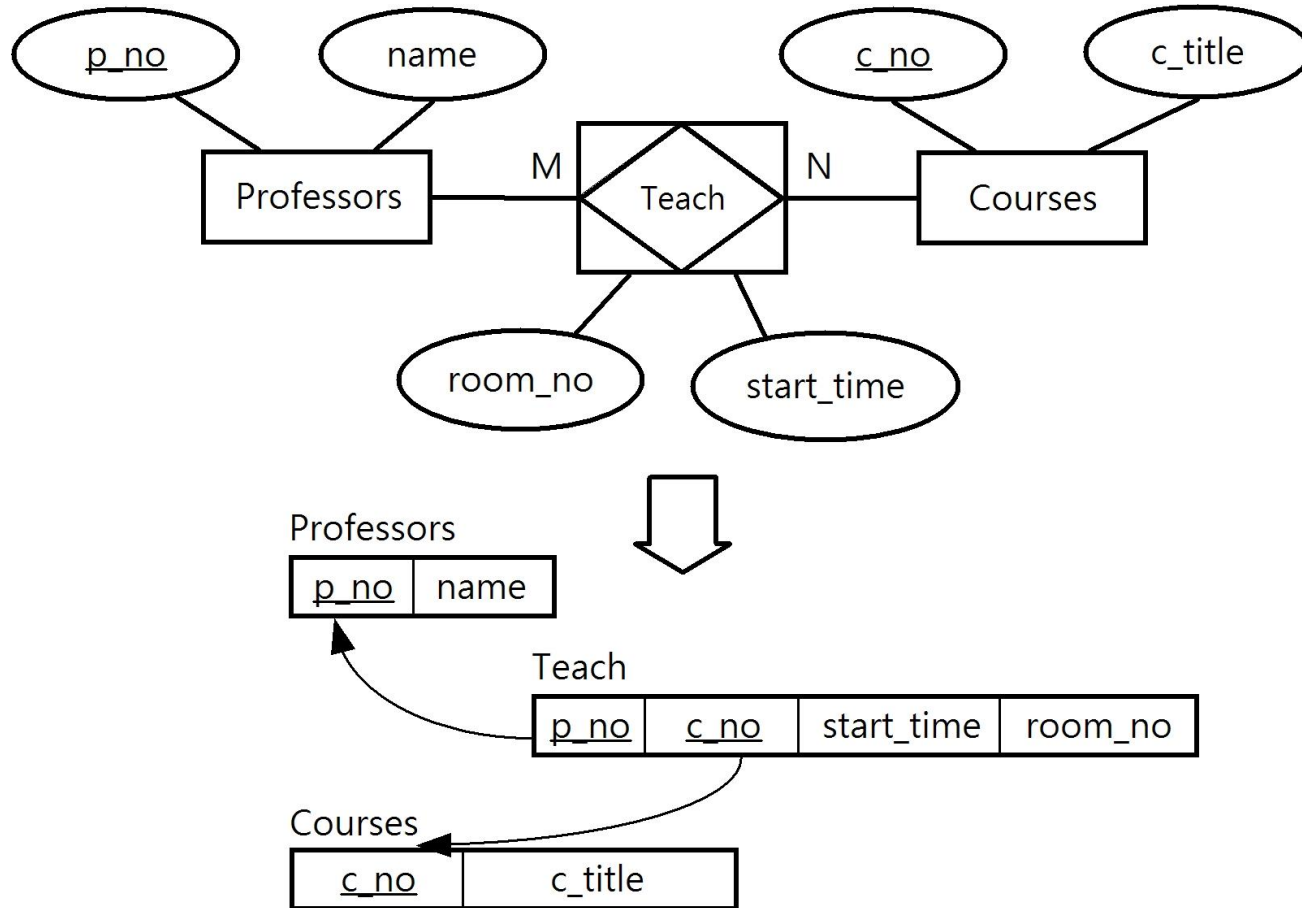


3-1-3 邏輯資料模型 – 種類

- 資料庫系統演進各年代的資料庫系統中，其使用的資料庫模型就是邏輯資料模型，主要有四種邏輯資料模型，如下所示：
 - 網路式資料庫模型（Network Database Model）。
 - 階層式資料庫模型（Hierarchical Database Model）。
 - 關聯式資料庫模型（Relational Database Model）。
 - 物件導向式資料庫模型（Object-Oriented Database Model）。



3-1-3 邏輯資料模型 - 圖例





3-1-4 實體資料模型 – 說明

- 實體資料模型（**Physical Data Model**）是針對指定資料庫管理系統建立實際資料庫結構的資料模型，例如：**SQL Server**，資料模型可以顯示資料是如何實作和儲存在資料庫中，例如：資料型別和尺寸。
- 對於關聯式資料庫模型的實體資料模型來說，就是在資料庫管理系統軟體建立關聯表（**Relation**）的表格、關聯性（**Relationship**）和索引等定義資料。



3-1-4 實體資料模型 – 圖例

- 最簡單的方式是直接使用表格建立實體資料模型，例如：將第3-1-3節邏輯資料模型轉換成實體資料模型，如下表所示：

關聯表格	欄位名稱	資料型別與尺寸	說明
Professors	<u>p_no</u>	int	主鍵
	p_name	char(10)	建立索引idx_pname
Teach	<u>c_no</u>	int	主鍵(外部鍵)
	<u>p_no</u>	int	主鍵(外部鍵)
	room_no	int	建立索引idx_roomno
	start_time	datetime	
Courses	<u>c_no</u>	int	主鍵
	c_title	varchar(50)	



3-2 資料庫模型的種類

- 3-2-1 網路式資料庫模型
- 3-2-2 階層式資料庫模型
- 3-2-3 關聯式資料庫模型





3-2 資料庫模型的種類

- 資料庫的邏輯資料模型有很多種，在本節準備簡單說明三種資料庫模型，如下所示：
 - 網路式資料庫模型
 - 階層式資料庫模型
 - 關聯式資料庫模型
- 因為關聯式資料庫模型是本書的主題，所以在第3-3節和第3-4節會詳細說明，至於物件導向式和物件關聯式資料庫模型請參閱第15章。



3-2-1 網路式資料庫模型 – 說明

- 網路式資料庫模型是依據1971年由CODASYL DBTG（Conference on Data Systems Languages Database Task Group）組織所提出的報告，或稱為DBTG模型。Database Task Group從1960年代後期就開始發展網路式資料庫系統。
- 最著名的網路式資料庫系統是IBM公司的「IDMS」（Integrated Database Management System），雖然網路式資料庫模型已經不再持續發展，不過直到現在，仍然存在一些運作中的網路式資料庫系統。



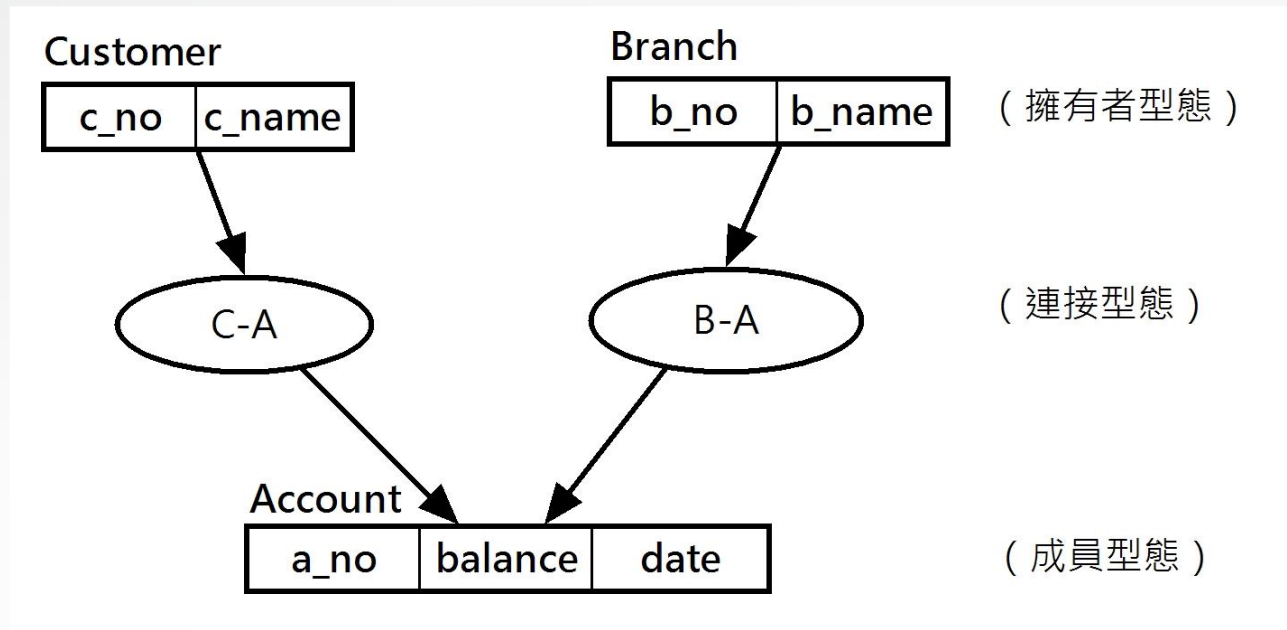
3-2-1 網路式資料庫模型 – 資料結構(說明)

- 網路式資料庫模型的資料結構是將資料連接成網路狀圖形，使用低階指標連接記錄資料，所以記錄資料存取非常快速，支援多對多關聯性（Relationship），而且在資料間的連接允許擁有迴圈。
- 網路式資料庫模型擁有兩種基本型態，其說明如下所示：
 - 記錄型態（Record Type）：記錄型態是由一組屬性所組成，每一個記錄型態的成員稱為記錄，資料是一組記錄的集合。
 - 連接型態（Link Type）：它是連接兩個記錄型態的型態，屬於一對多關聯性（Relationship），這是從稱為「擁所有者型態」（Owner Type）關聯到多個「成員型態」（Member Type）。



3-2-1 網路式資料庫模型 – 資料結構(圖例)

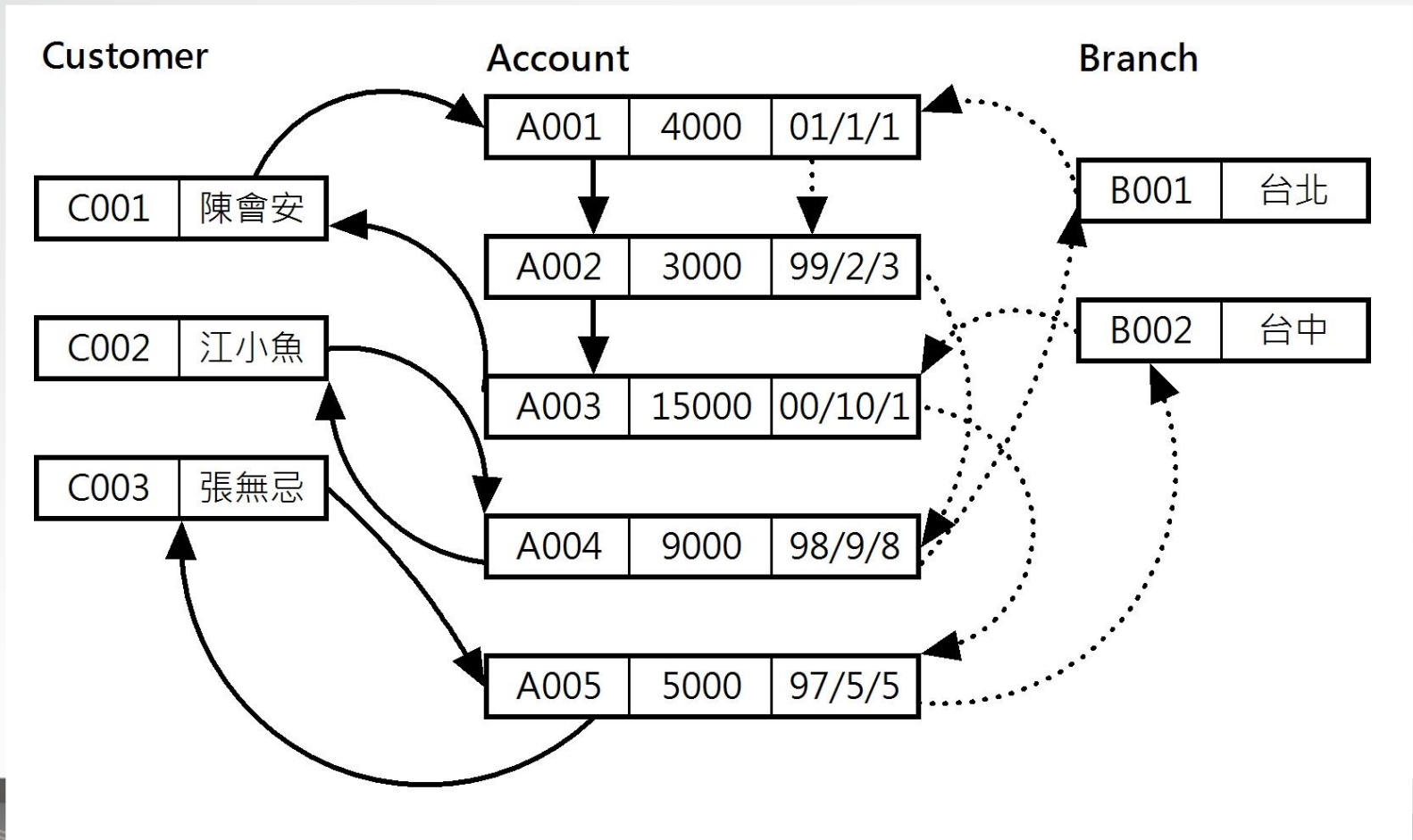
- 網路式資料庫模型是建立在兩種「集合結構」(Set Structures)，也就是一組記錄型態的記錄集合 (A Set of Records) 和一組連接型態的連接集合 (A Set of Links)，如下圖所示：





3-2-1 網路式資料庫模型 – 網路式資料庫

- 例如：完整銀行分行帳戶的網路式資料庫，如下圖所示：





3-2-2 階層式資料庫模型 – 說明

- 階層式資料庫模型（Hierarchical Database Model）類似網路式資料庫模型，也是使用低階指標來連接記錄資料，不過，它是使用樹狀結構來組織資料，記錄資料間是以父子關係來建立連接，子記錄只能擁有一個父記錄。
- 在1960年代後期到1970年代，階層式資料庫模型的資料庫管理系統非常的普遍，最著名的階層式資料庫系統是1968年IBM公司開發的「IMS」（Information Management System），這也是最早商用的資料庫管理系統。



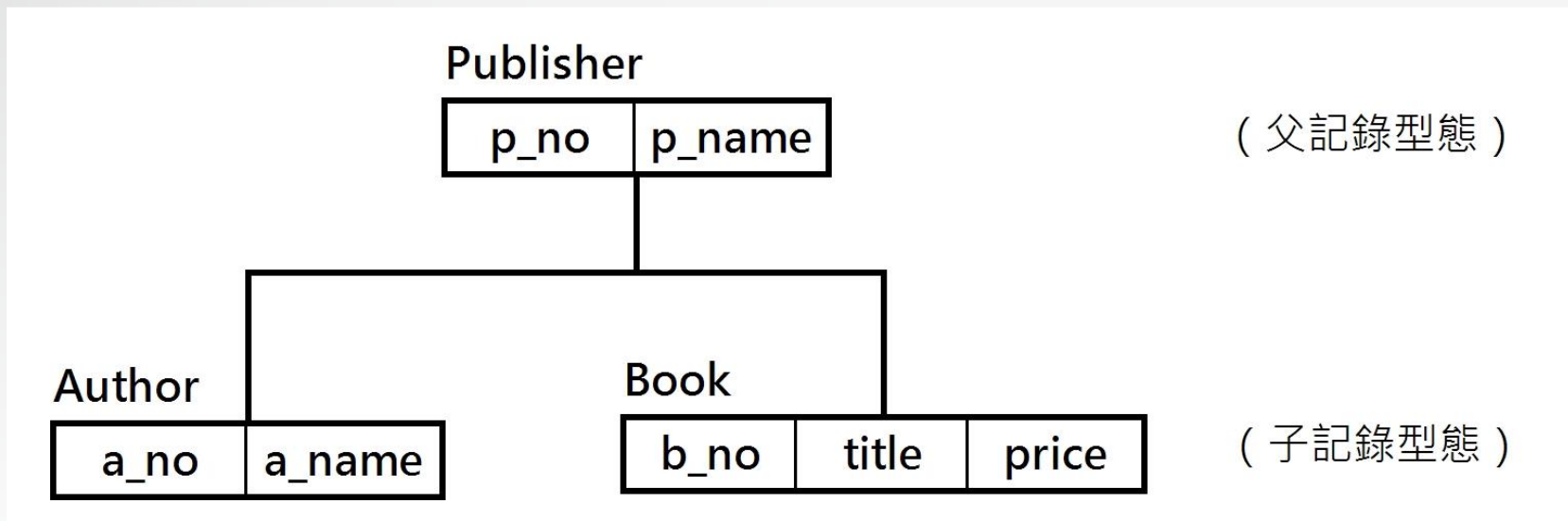
3-2-2 階層式資料庫模型 – 資料結構(說明)

- 階層式資料庫模型的資料結構一定擁有一個「樹根」(Root)，然後使用「父子關聯性」(Parent-child Relationships) 連接記錄集合，將資料建立成階層的樹狀結構。階層式資料庫模型也擁有兩種基本型態，其說明如下所示：
 - 記錄型態 (Record Type)：記錄型態是由一組欄位屬性組成。每一個記錄型態的成員稱為記錄，資料是一組記錄的集合。
 - 父子關聯型態 (Parent-child Relationship Type)：兩個記錄型態間的連接型態，屬於一對多關聯性 (Relationship)，這是從稱為「父記錄型態」(Parent Record Type) 關聯到多個「子記錄型態」(Child Record Type)。



3-2-2 階層式資料庫模型 – 資料結構(圖例)

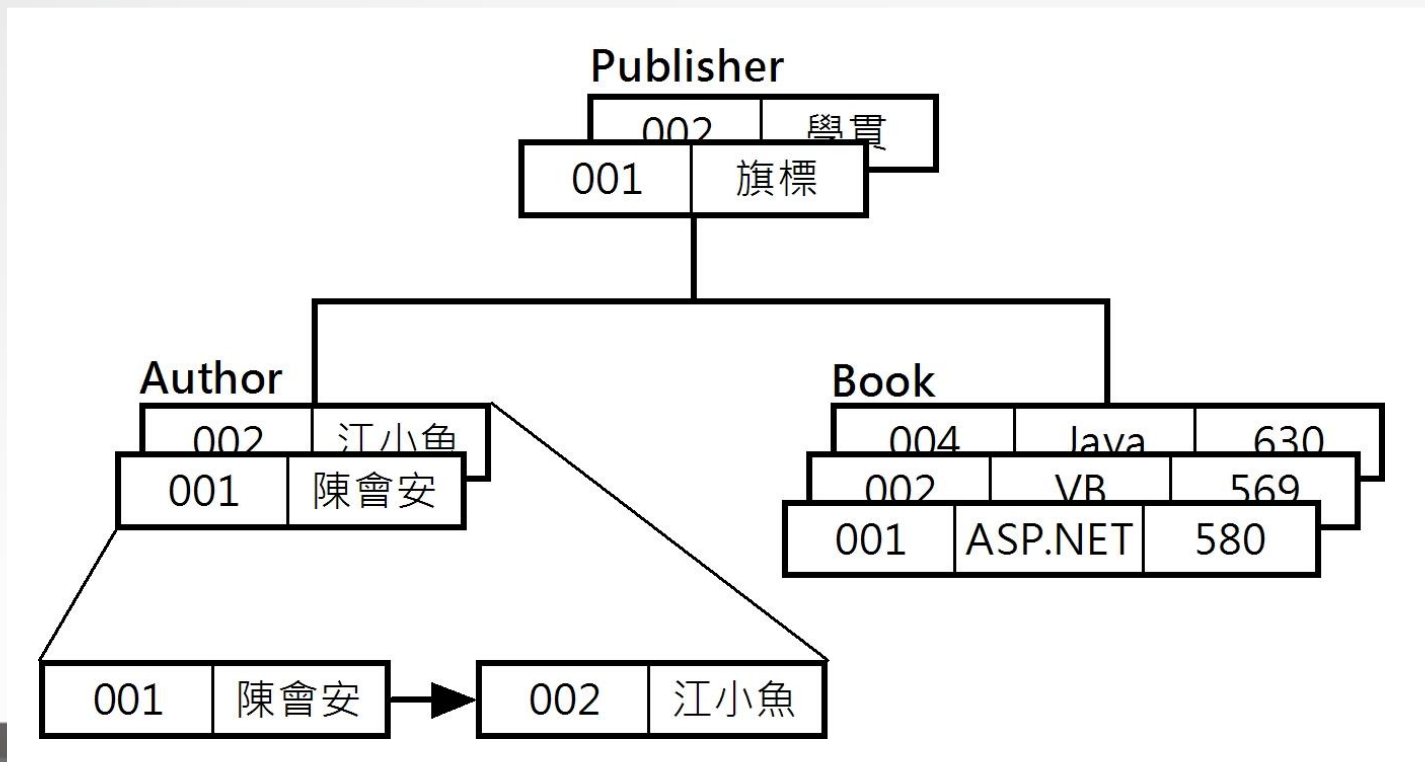
- 階層式資料庫模型是由多個記錄型態，然後使用父子關聯型態將它們連接起來，如下圖所示：





3-2-2 階層式資料庫模型 – 階層式資料庫

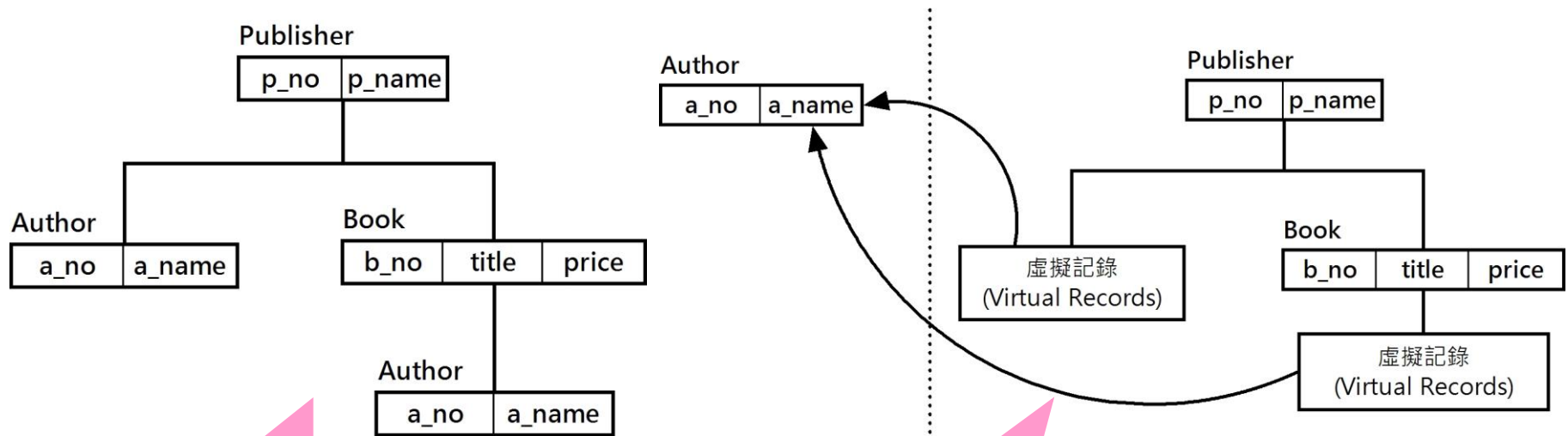
- 在階層式資料庫模型的父子關係是一個父親允許有多個兒子，可是兒子只能有一個父親。完整圖書出版的階層式資料庫，如下圖所示：





3-2-2 階層式資料庫模型 – 多對多關聯性

- 對於多對多關聯性（Relationships）來說，階層式資料庫模型可以使用重複相同記錄型態和「虛擬記錄型態」（Virtual Record Type）來實作，虛擬記錄的記錄只是一個指標，用來指向其他記錄型態的記錄，如下圖所示：



重複相同記錄型態

虛擬記錄型態



3-2-3 關聯式資料庫模型 – 說明

- 關聯式資料庫模型（Relational Database Model）是1969年E. F. Codd博士在IBM公司的研究成果，不同於其他資料庫模型的資料庫，關聯式資料庫模型是使用數學集合論為理論基礎所建立的資料庫模型，在1970~1980年之間，E. F. Codd博士持續發表多篇關聯式資料庫的相關論文，讓關聯式資料庫模型成為1980年代資料庫的主流。



3-2-3 關聯式資料庫模型 – 組成元素

- 資料結構（Data Structures）：資料的組成方式，就是欄和列組成表格的關聯表（Relations）。
- 資料操作或運算（Data Manipulation或Operations）：資料的相關操作，即第16章關聯式代數（Relational Algebra）和關聯式計算（Relational Calculus）。
- 完整性限制條件（Integrity Constraints）：維護資料完整性的條件，其目的是確保儲存的資料是合法的資料。



3-3 關聯式資料庫模型的資料結構

- 3-3-1 關聯表綱要
- 3-3-2 關聯表實例
- 3-3-3 關聯表的特性
- 3-3-4 關聯表的種類





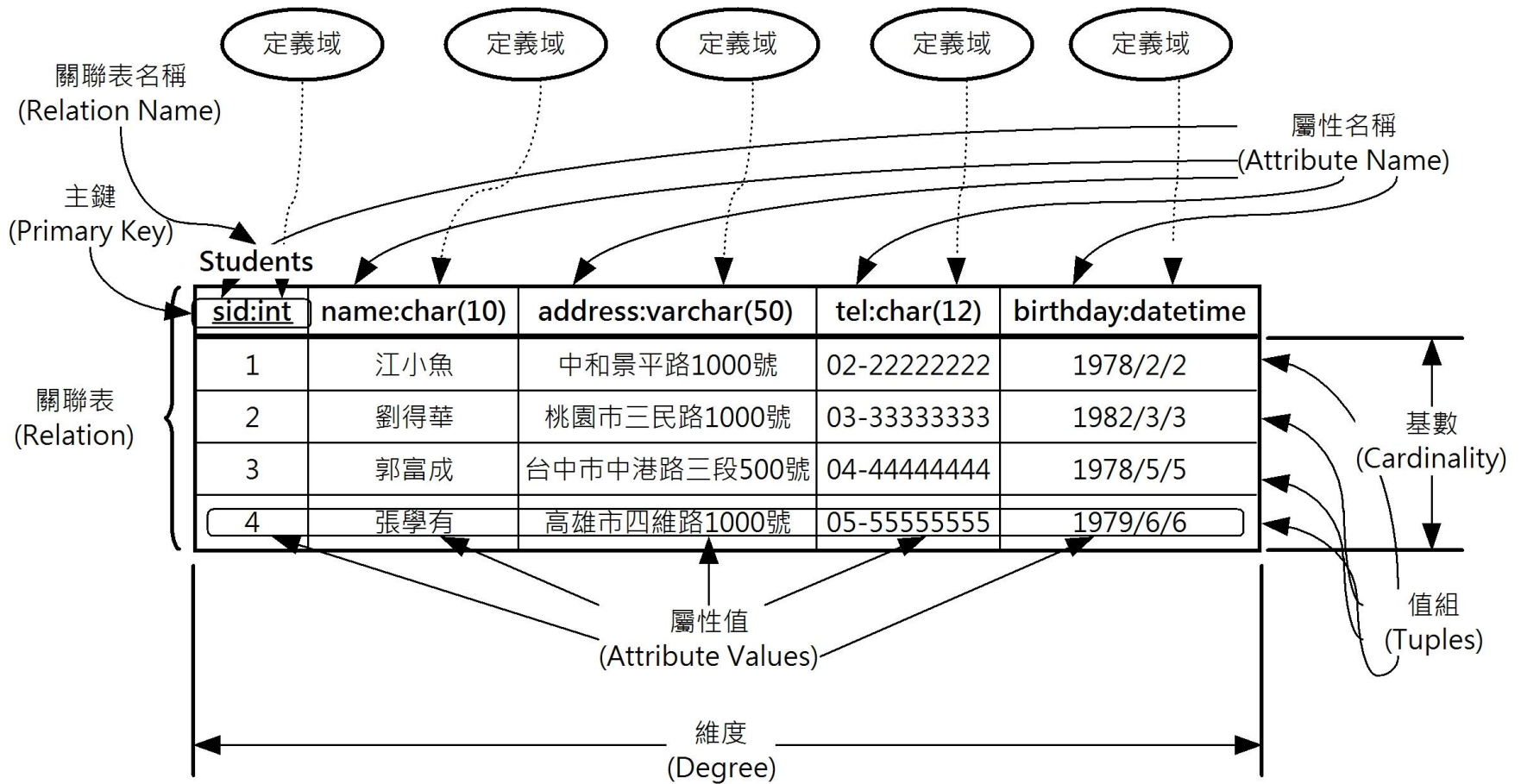
3-3 關聯式資料庫模型的資料結構

- 關聯式資料庫是一組關聯表（Relations）的集合，關聯表是關聯式資料庫模型的資料結構（Data Structures），使用二維表格來組織資料。每一個關聯表是由關聯表綱要（Relation Schema）和關聯表實例（Relation Instance）兩個部分所組成，如下圖所示：

關聯表綱要 (Relation Schema)	{	Students				
		<u>sid:integer</u>	name:string	address:string	tel:string	birthday:date
關聯表實例 (Relation Instance)	{	S001	江小魚	中和景平路1000號	02-22222222	1978/2/2
		S002	劉得華	桃園市三民路1000號	03-33333333	1982/3/3
		S003	郭富成	台中市中港路三段500號	04-44444444	1978/5/5
		S004	張學有	高雄市四維路1000號	05-55555555	1979/6/6



3-3-1 關聯表綱要 - 相關術語(圖例)





3-3-1 關聯表綱要 – 相關術語(說明1)

- 關聯表（**Relations**）：相當於是一個二維表格，不過，我們不用考慮各列和各欄資料的順序，每一個關聯表擁有一個唯一的關聯表名稱。例如：名為**Students**的關聯表。
- 屬性（**Attributes**）：在關聯表的所有屬性是一個「屬性集合」（**Attribute Set**），所以關聯表的屬性不能重複。
- 值組（**Tuples**）：關聯表的一列，也就是一筆記錄，一組目前屬性值的集合。
- 維度（**Degree**）：關聯表的維度是指關聯表的屬性數目，關聯表至少擁有**2**個屬性（主鍵加上**1**個非主鍵屬性），所以最小維度為**2**，以此例**Students**關聯表的維度是**5**。



3-3-1 關聯表綱要 – 相關術語(說明2)

- **基數 (Cardinality)**：關聯表的基數是關聯表的值組數目，如果關聯表沒有任何記錄，基數為0，以Students關聯表為例，其基數為4。
- **主鍵 (Primary Key)**：在關聯表需要選擇一個或多個屬性的屬性子集合 (**Attribute Subset**) 作為主鍵，它是用來識別值組是唯一的。以此例sid屬性是主鍵，主鍵的屬性值1可以識別出是第1筆值組，從屬性值1、2可以判斷第1筆和第2筆值組是不同的學生。
- **定義域 (Domains)**：一組可接受屬性值的集合，通常使用資料型別來代表值集合的範圍，以此例sid的屬性值範圍是int、name是char、address是varchar和birthday是datetime。



3-3-1 關聯表綱要 – 相關術語(說明3)

- 不過，關聯式資料庫模型、關聯式資料庫和檔案系統使用的術語並不一致，這些術語雖然不同，但都代表相同的意義，如下表所示：

關聯式資料庫模型	資料庫	檔案系統
關聯表 (Relation)	資料表 (Table)	檔案 (File)
值組 (Tuple)	資料列 (Row)	記錄 (Record)
屬性 (Attribute)	資料欄 (Column)	欄位 (Field)



3-3-1 關聯表綱要 – 表示法(語法)

- 本書關聯表綱要表示法的基本語法，如下所示：
relation_name (attribute1, attribute2, attribute3, ... , attributeN)
- 上述語法的說明，如下所示：
 - relation_name：關聯表名稱。
 - attribute1, attribute2, attribute3, ... , attributeN：括號中是屬性清單，通常省略屬性的定義域。
- 在屬性加上底線表示它是主鍵，外來鍵可以使用虛線底線或其他表示方法。



3-3-1 關聯表綱要 – 表示法(範例)

- 以關聯表Students為例，其關聯表綱要，如下所示：

Students (sid, name, address, tel, birthday)

- 上述關聯表Students的主鍵是sid。
- 如果在關聯表綱要需要標示定義域的int、char和datetime，可以使用括號標示在屬性後，如下所示：

Students (sid(int), name(char), address(varchar), tel(char), birthday(datetime))



3-3-2 關聯表實例 – 說明

- 在定義好關聯表綱要後，就可以將資料儲存到關聯表，稱為關聯表實例（**Relation Instance**）。這是一個有限個數的集合，集合內容是關聯表的值組（**Tuples**）。
- 關聯表實例是指某一時間點的值組集合。例如：第3-3-1節Students關聯表實例，如下表所示：

1	江小魚	中和景平路1000號	02-22222222	1978/2/2
2	劉得華	桃園市三民路1000號	03-33333333	1982/3/3
3	郭富成	台中市中港路三段500號	04-44444444	1978/5/5
4	張學有	高雄市四維路1000號	05-55555555	1979/6/6



3-3-2 關聯表實例 – 定義域(說明)

- 定義域（**Domains**）是一組可接受值的集合，這些值是不可分割的單元值（**Atomic**），也就是說，不可以是另一個集合。
- 對比程式語言，定義域相當於變數的資料型別，值組的屬性值相當於變數值，滿足資料型別的定義域範圍。



3-3-2 關聯表實例 – 定義域(種類)

- 簡單屬性（**Simple Attributes**）：簡單屬性是一種不可再分割的屬性，其定義域是相同型別的單元值集合。例如：台灣城市的集合和12個月份，如下所示：

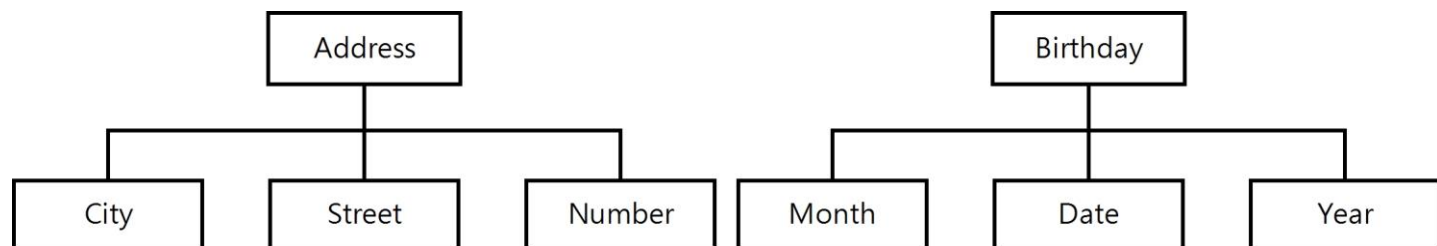
{ '台北市' , '台中市' , '高雄市' }

{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 }

- 複合屬性（**Composite Attributes**）：複合屬性是由簡單屬性所組成的屬性，可以建立成一個階層架構。例如：地址屬性和生日屬性是由數個簡單屬性所組成，如下所示：

Address = City+Street+Number

Birthday = Month+Day+Year





3-3-2 關聯表實例 – 屬性值(說明)

- 屬性值（Attribute Values）是關聯表實際儲存資料的最小單位，在關聯表屬性集合的每一個屬性都擁有一組可接受的值，即屬性的定義域。例如：學生Students關聯表的屬性值集合（Attribute Value Set）是指目前關聯表實例各屬性所包含的值範圍，如下所示：

city屬性值 = { '中和市' , '桃園市' , '台中市' , '高雄市' }

grade屬性值 = 56~89

Students

sid	name	city	grade
1	江小魚	中和市	78
2	劉得華	桃園市	56
3	郭富成	台中市	89
4	張學有	高雄市	66



3-3-2 關聯表實例 – 屬性值(特點)

- 關聯表屬性值擁有的特點，如下所示：
 - 單元值（**Atomic**）：屬性值是不可分割的單元值。
 - 需要指派定義域：屬性值一定需要指派其定義域，而且只有一個定義域，雖然屬性值屬於指派的定義域，但不表示所有定義域值都會出現，屬性值集合可能只是定義域的部分集合。
 - 可能為空值：屬性值可能是本節後說明的空值。



3-3-2 關聯表實例 – 空值(說明)

- 在關聯表的屬性值可能是一個未知或無值的空值（**Null Values**），此值是一個特殊符號，它不是0，也不是空字串，所有定義域都會包含空值。
- 空值本身並沒有意義，所以不能作為真偽的比較運算，例如：**5 = NULL**並無法判斷是**true**或**false**。
空值的意義有兩種：
 - 未知值（**Unknown**）
 - 不適性（**Not Applicable**）



3-3-2 關聯表實例 – 空值(未知值)

- 屬性值是一個未知值（**Unknown**），這個部分的空值可以分成兩種情況，如下所示：
 - 找不到（**Missing**）：屬性值存在但找不到，例如：不知道學生【陳大安】的地址，因為address屬性值一定存在只是找不到，所以它代表一個找不到的空值。
 - 完全未知（**Total Unknown**）：不知道屬性值是否存在。例如：不知道張先生是否有配偶，所以配偶欄spouse是完全未知的空值。

Students

sid	name	address	age	GPA
1	江小魚	中市景平路1000號	25	3
2	劉得華	桃園市三民路10號	26	3.5
3	郭富成	台中市中港路5號	30	4
4	陳大安	NULL	21	1

找不到

完全未知

Label

id	name	spouse	age
1	張先生	NULL	25
2	劉先生	江小姐	30



3-3-2 關聯表實例 – 空值(不適性)

- 不適性（Not Applicable）空值是指屬性沒有適合的屬性值。例如：公司員工劉先生沒有手機，所以cellphone屬性值的手機號碼是一個不適性的空值，如下圖所示：

Employees

<u>id</u>	name	cellphone	age
1	張先生	0938-000123	25
2	劉先生	NULL	30



3-3-3 關聯表的特性 – 說明

- 關聯表的名稱是唯一的，在資料庫不能有兩個關聯表擁有相同的名稱，同一個關聯表的屬性名稱也是唯一，不過，不同關聯表之間允許擁有相同名稱的屬性。
- 在關聯表除了名稱唯一性的特性外，還擁有四個特性：
 - 沒有重複的值組
 - 值組是沒有順序
 - 屬性也沒有順序
 - 所有的屬性值都是單元值



3-3-3 關聯表的特性 – 沒有重複的值組

- 關聯表是數學集合，在集合中不可有重複元素，所以關聯表沒有重複值組，其隱含意義是關聯表擁有主鍵，主鍵是值組的識別，所以沒有兩個值組是完全相同的。
 - 例如：**Access**資料庫管理系統就允許重複記錄。



3-3-3 關聯表的特性 – 值組是沒有順序

- 在關聯表的值組因為是集合，所以並沒有順序的分別，也就是說，如果重新排列關聯表的值組，也不會產生新的關聯表。



3-3-3 關聯表的特性 – 屬性也沒有順序

- 關聯表的屬性也沒有順序差別，如果重新排列關聯表的屬性，也不會產生新的關聯表。
- 事實上，大部分資料庫管理系統並不支援此特性，資料庫管理系統提供的資料庫存取函式庫，不但可以取得屬性的原始順序，而且允許使用順序來存取屬性值。



3-3-3 關聯表的特性 – 所有屬性值都是單元值

- 關聯表的屬性值都是單元值（Atomic），這是指二維表格中的每一個儲存格的值都是單一值，而不是一組值的集合，如下圖所示：

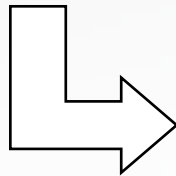
Students

sid	name	address	tel	birthday
S001	{ 江小魚, 江大魚 }	中和景平路1000號	02-22222222	{ 1978/2/2, 1975/3/3 }
S002	劉得華	桃園市三民路1000號	03-33333333	1982/3/3
S003	郭富成	台中市中港路三段500號	04-44444444	1978/5/5
S004	張學有	高雄市四維路1000號	05-55555555	1979/6/6

不是單元值

單元值

正規化



Students

sid	name	address	tel	birth day
S001-1	江小魚	中和景平路1000號	02-22222222	1978/2/2
S001-2	江大魚	中和景平路1000號	02-22222222	1975/3/3
S002	劉得華	桃園市三民路1000號	03-33333333	1982/3/3
S003	郭富成	台中市中港路三段500號	04-44444444	1978/5/5
S004	張學有	高雄市四維路1000號	05-55555555	1979/6/6



3-3-4 關聯表的種類 - 1

- 具名關聯表（Named Relations）：在資料庫管理系統使用 CREATE TABLE、CREATE VIEW和CREATE SNAPSHOT指令建立擁有名稱的關聯表，也就是使用者知道的關聯表。
- 基底關聯表（Base Relations）：一種具名關聯表，它是實際儲存資料的關聯表，也稱為「真實關聯表」（Real Relations）。
- 導出關聯表（Derived Relations）：由其他具名關聯表，經過運算而得的關聯表。
- 視界（Views）：定義在其他基底關聯表上的一種「虛擬關聯表」（Virtual Relations），屬於一種具名的導出關聯表。



3-3-4 關聯表的種類 - 2

- 快照關聯表（**Snapshots**）：也是具名的導出關聯表，在一個時間點的關聯表內容，不過它有真正儲存資料，屬於一種唯讀關聯表，需要周期更新資料，我們可以使用**CREATE SNAPSHOT**指令來建立。
- 查詢結果（**Query Results**）：一種沒有具名的導出關聯表，這不是儲存在資料庫的關聯表，而是在需要時，執行資料庫查詢所得的關聯表，當不再需要時就會刪除。
- 中間結果（**Intermediate Results**）：一種沒有具名的導出關聯表，這是**JOIN**合併查詢指令子查詢的查詢結果。
- 暫存關聯表（**Temporary Relations**）：一種具名關聯表，儲存的是資料庫管理系統暫時所需的資料，系統會視需求自動產生和刪除。



3-3-4 關聯表的種類 - 3

- 儲存型關聯表（**Stored Relations**）：實際儲存在儲存裝置的關聯表，它是真正可儲存資料的關聯表。例如：基底或快照關聯表。



3-4 關聯式資料庫模型的完整性限制條件

- 3-4-1 鍵限制條件
- 3-4-2 定義域限制條件
- 3-4-3 實體完整性
- 3-4-4 參考完整性
- 3-4-5 其他完整性限制條件





3-4 關聯式資料庫模型的完整性限制條件

- 關聯式資料庫模型的完整性限制條件（**Integrity Constraints**）是資料庫設計的一部分，其目的是建立檢查資料庫儲存資料的依據和保障資料的正確性。不但可以防止授權使用者將不合法資料存入資料庫，還能夠避免關聯表間的資料不一致。
- 關聯式資料庫模型的完整性限制條件有很多種，適用所有關聯式資料庫的完整性限制條件有四種，如下所示：
 - 鍵限制條件（**Key Constraints**）
 - 定義域限制條件（**Domain Constraints**）
 - 實體完整性（**Entity Integrity**）
 - 參考完整性（**Referential Integrity**）



3-4-1 鍵限制條件 – 說明1

- 關聯式資料庫模型的鍵是一個重要觀念，關聯表的「鍵」(Keys)是指關聯表綱要中單一屬性或一組屬性的集合。鍵限制條件 (Key Constraints) 是指關聯表一定擁有一個唯一和最小的主鍵 (Primary Key)。例如：在Students關聯表找出主鍵，其內容如下圖所示：

範例關聯表

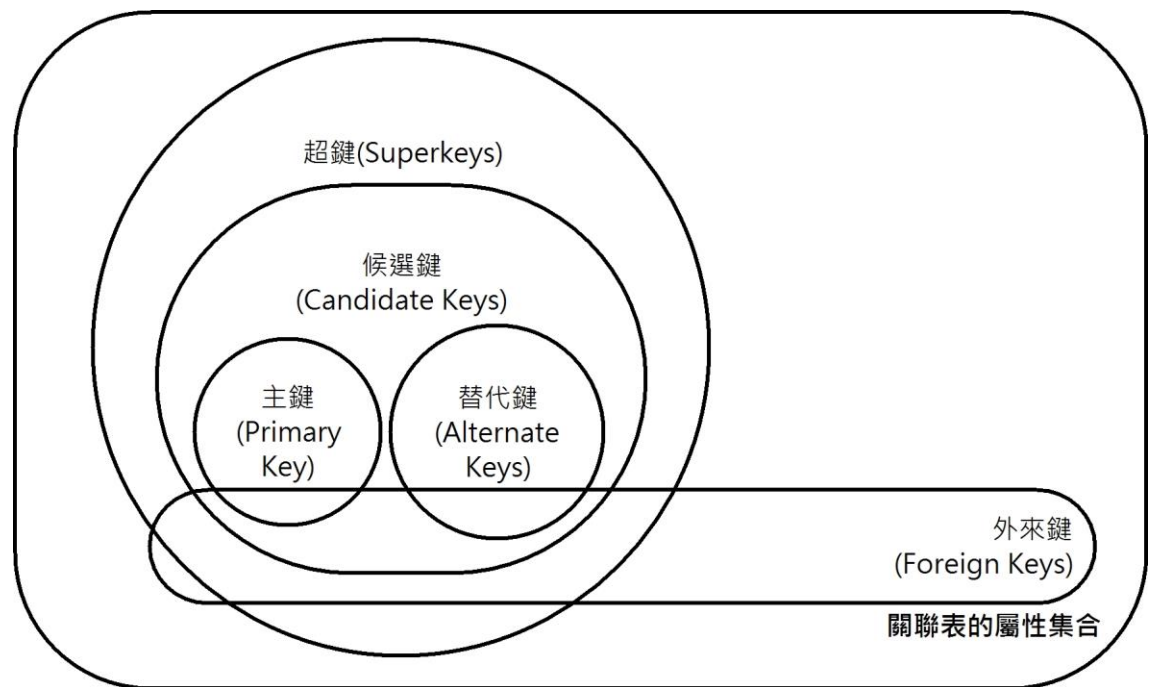
Students

sid	SSN	ename	cname	postcode	tel
1	A123456	Jane	江小魚	220	02-22222222
2	B345689	Tom	劉得華	100	03-33333333
3	H123987	John	郭富成	300	04-44444444
4	J896756	Tony	張學有	248	05-55555555



3-4-1 鍵限制條件 – 說明2

- 在Students關聯表可以找出的鍵分為：超鍵（Superkeys）、候選鍵（Candidate Keys）、主鍵（Primary Key）、替代鍵（Alternate Keys）和外來鍵（Foreign Keys），如下圖所示：





3-4-1 鍵限制條件 – 超鍵

- 超鍵（**Superkeys**）是關聯表綱要的單一屬性或屬性值集合，超鍵需要滿足唯一性，如下所示：
 - 唯一性（**Uniqueness**）：在關聯表中絕不會有兩個值組擁有相同值。
- 透過超鍵的識別，就可以在關聯表存取指定的值組。例如：學號002的學生資料，而不是學號003。
 - 以**Students**關聯表為例，符合條件的超鍵，如下所示：
(sid)、(SSN)、(sid, SSN)、(sid, ename)、(SSN, cname)、(SSN, postcode)、(sid, tel)、(sid, ename, cname)、(SSN, cname, postcode)、.....



3-4-1 鍵限制條件 – 候選鍵

- 候選鍵（Candidate Keys）是一個超鍵，在每一個關聯表至少擁有一個候選鍵，不只滿足超鍵的唯一性，還需要滿足最小性，如下所示：
 - 最小性（Minimality）：最小屬性數的超鍵，在超鍵中沒有一個屬性可以刪除，否則將違反唯一性。
- 關聯表的候選鍵需要同時滿足唯一性和最小性，也就是說，候選鍵是最小屬性數的超鍵，所以單一屬性的超鍵一定是候選鍵。
- 以Students關聯表為例，在超鍵中符合條件的候選鍵，如下所示：
(sid)、(SSN)



3-4-1 鍵限制條件 – 主鍵

- 主鍵（**Primary Key**；**PK**）是關聯表各候選鍵的其中之一，而且只有一個。例如：**Students**關聯表的(**sid**)和(**SSN**)都是候選鍵，關聯表的主鍵就是這兩個候選鍵的其中之一。
- 因為關聯表可能擁有多個候選鍵，如何在眾多候選鍵挑選主鍵的原則如下所示：
 - 不可為空值（**Not Null**）
 - 永遠不會改變（**Never Change**）
 - 非識別值（**Nonidentifying Value**）
 - 簡短且簡單的值（**Brevity and Simplicity**）



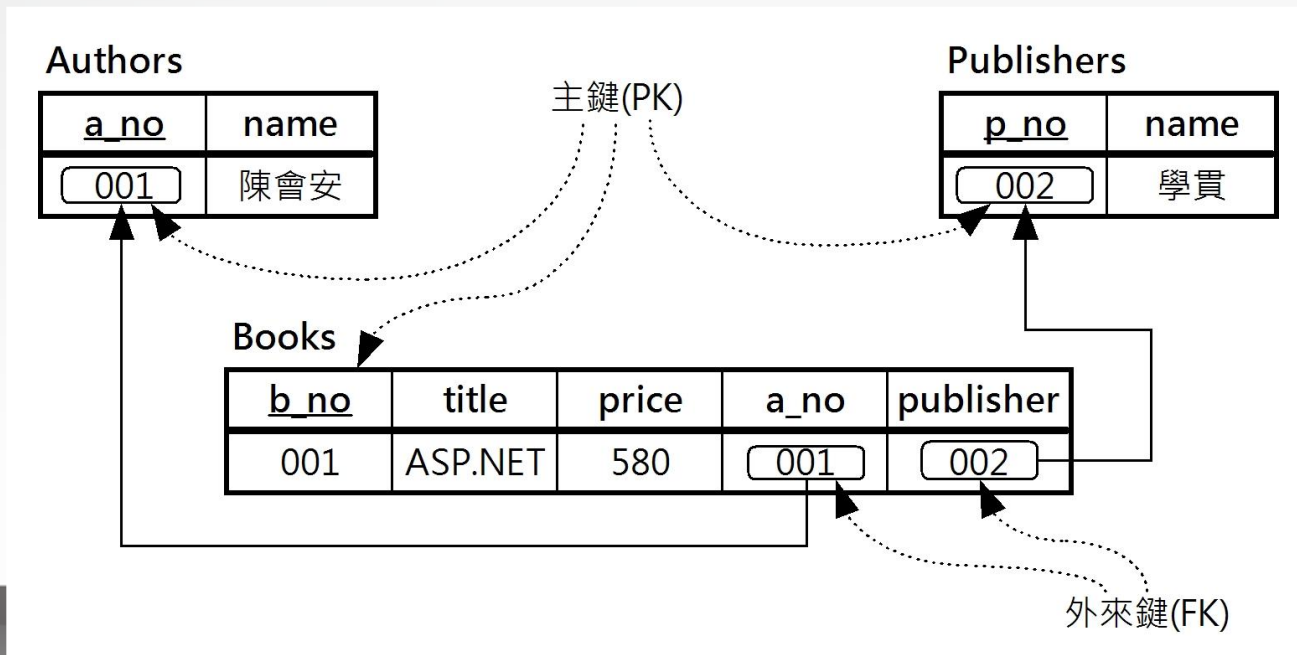
3-4-1 鍵限制條件 – 替代鍵

- 在候選鍵中不是主鍵的其他候選鍵稱為替代鍵（**Alternate Keys**），因為這些是可以用來替代主鍵的候選鍵。
 - 以Students關聯表來說，(sid)和(SSN)是兩個候選鍵，因為(sid)是主鍵；(SSN)就是替代鍵。



3-4-1 鍵限制條件 – 外來鍵(說明)

- 外來鍵（Foreign Keys；FK）是關聯表的單一或多個屬性的集合，它的屬性值是參考到其他關聯表的主鍵，當然也可能參考同一個關聯表的主鍵。外來鍵和其他關聯表的主鍵是對應的，在關聯式資料庫扮演連接關聯表的膠水功能，如下圖所示：





3-4-1 鍵限制條件 – 外來鍵(特性)

- 外來鍵一定參考其他關聯表的主鍵，可以用來建立兩個關聯表間的連接。
- 外來鍵在關聯表內不一定是主鍵。
- 外來鍵和參考的主鍵屬於相同定義域，不過屬性名稱可以不同。
- 外來鍵和參考主鍵中的主鍵如果是單一屬性；外來鍵就是單一屬性，主鍵是屬性集合；外來鍵一樣也是屬性集合。
- 外來鍵可以是空值NULL。
- 外來鍵可以參考同一個關聯表的主鍵。



3-4-2 定義域限制條件

- 定義域限制條件（**Domain Constraints**）是指在關聯表的屬性值一定是定義域的單元值（**Atomic**）
 - 例如：屬性**age**的定義域是**int**，屬性值可以為**5**，但不可以是**4.5**。對比程式語言，就是一種「強調型別」（**Strongly Typed**）程式語言。



3-4-3 實體完整性 – 說明

- 實體完整性是關聯表內部的完整性條件，主要是用來規範關聯表主鍵的使用規則。
- 實體完整性（Entity Integrity）是指在基底關聯表主鍵的任何部分都不可是空值，其規則如下所示：
 - 主鍵如果是多個屬性的集合，任何一個屬性都不可是空值，例如：**(ename, cname)**是主鍵，**ename**屬性不可為空值；**cname**屬性也不可為空值。
 - 在關聯表只有主鍵不可是空值，其他替代鍵並不適用此規則。
 - 實體完整性是針對基底關聯表，從其導出的關聯表並不用遵守。



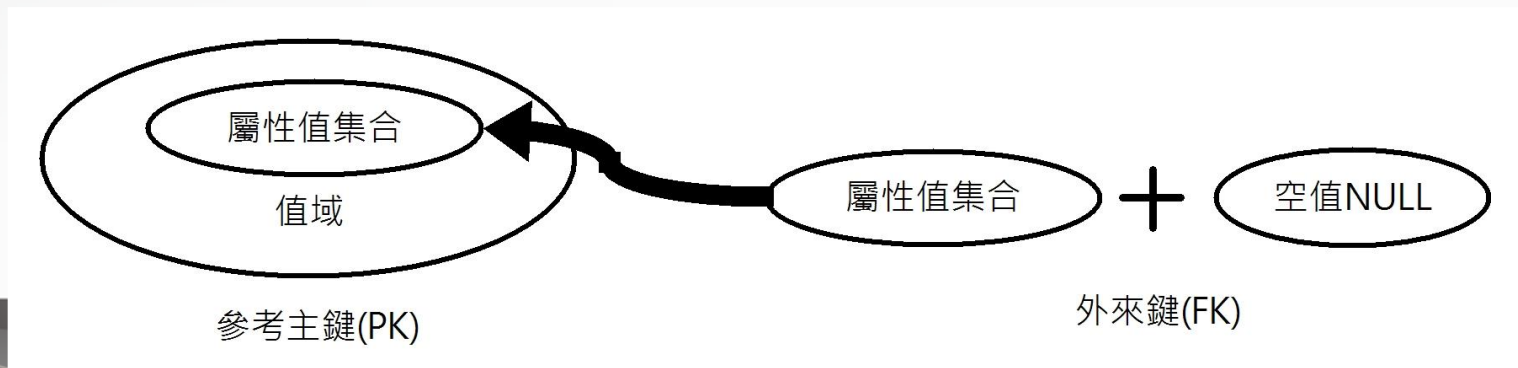
3-4-3 實體完整性 – 主鍵的使用規則

- 關聯式資料庫管理系統支援實體完整性，可以定義主鍵的更新規則，如下所示：
 - 主鍵的更新規則（**Update Rule**）：更新規則是指在基底關聯表的一個值組更新主鍵或新增值組時，如果主鍵是空值就違反實體完整性，資料庫管理系統必須拒絕這項操作。



3-4-4 參考完整性 - 說明

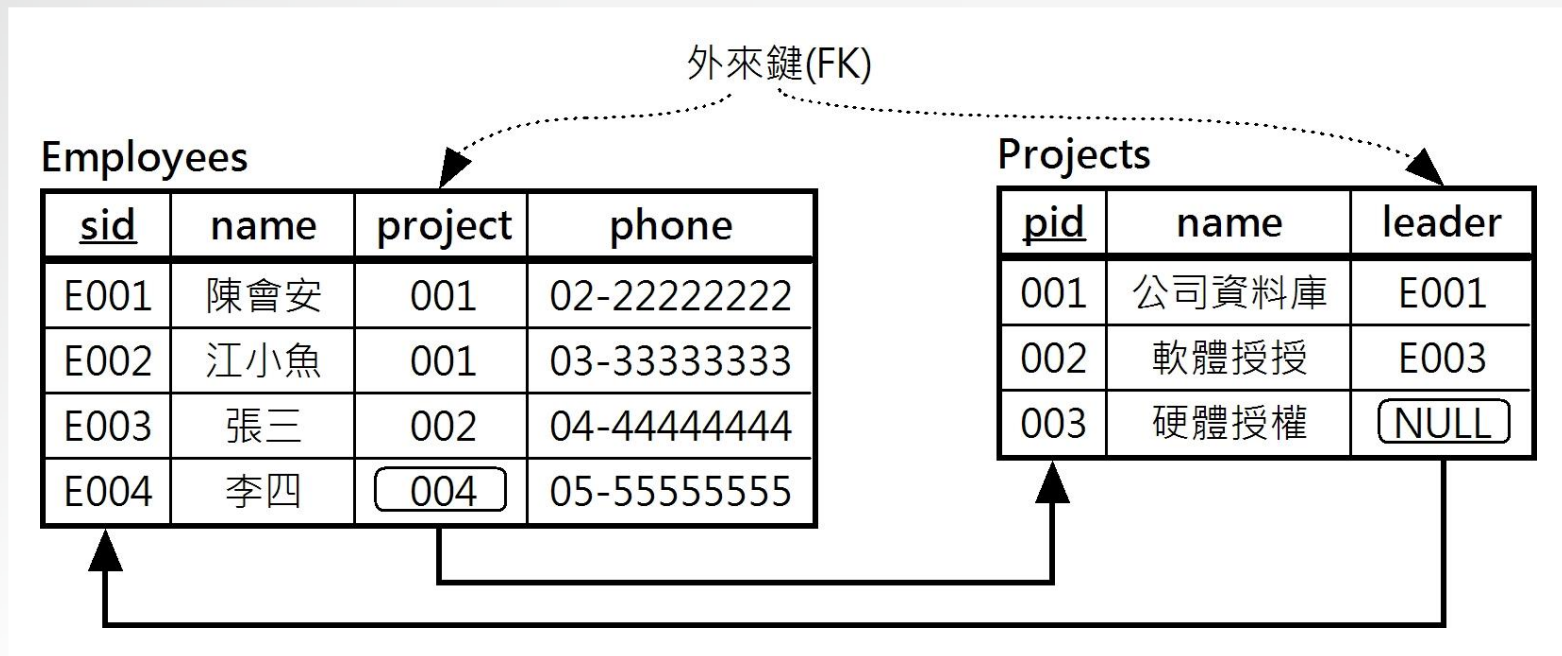
- 參考完整性是關聯表與關聯表間的完整性條件，主要是用來規範外來鍵的使用規則。
- 參考完整性（**Referential Integrity**）是當關聯表存在外來鍵時，外來鍵的值一定是來自參考關聯表的主鍵值，或為空值。也就是說，外來鍵的屬性值集合是對應參考主鍵的屬性值集合，如下圖所示：





3-4-4 參考完整性 – 範例

- 例如：公司員工關聯表Employees都會參與公司的專案關聯表Projects，如下圖所示：





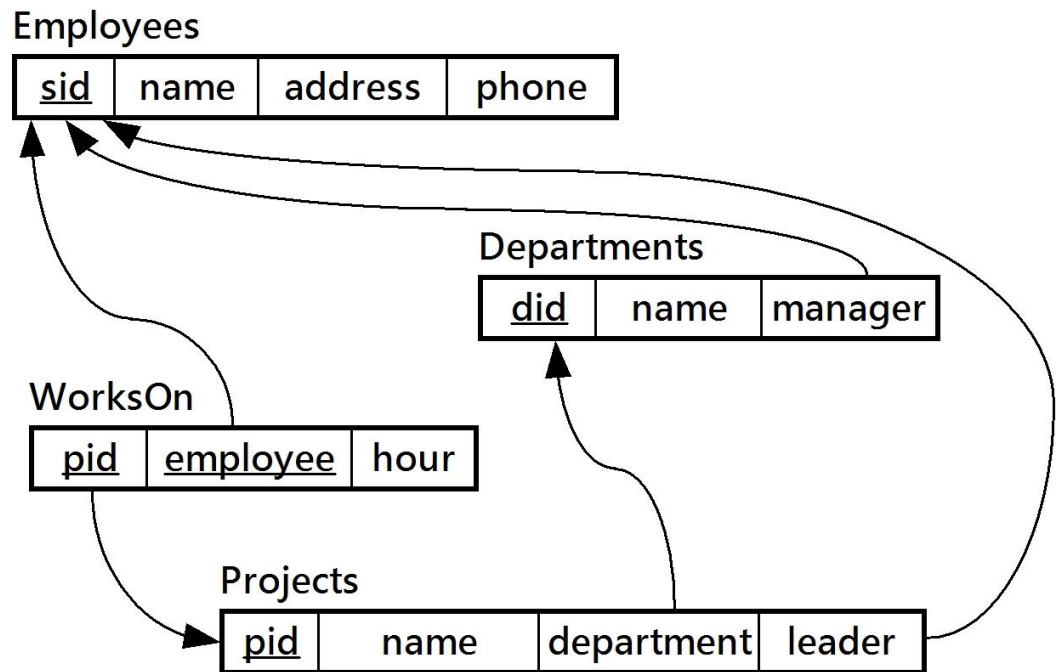
3-4-4 參考完整性 – 規則

- 參考完整性的規則如下所示：
 - 在關聯表不可包含無法參考的外來鍵。例如：員工李四的**project**外來鍵值根本不存在參考的主鍵**pid**，表示此值組違反參考完整性，因為沒有父親的**pid**，怎麼會有兒子**project**。
 - 如果外來鍵不是關聯表的主鍵，其屬性值可以為空值。例如：**Projects**關聯表硬體授權專案的**leader**外來鍵是空值，因為可能尚未指定專案領導者，所以沒有違反參考完整性。



3-4-4 參考完整性 - 外來鍵參考圖

- 在關聯式資料庫模型的外來鍵是資料庫中各關聯表間的結合劑，只需將外來鍵和參考主鍵連接起來，就可以了解關聯表間的關係，稱為外來鍵參考圖（Referential Diagram），如下圖所示：





3-4-4 參考完整性 – 外來鍵參考鏈和環

- 因為關聯表的外來鍵是參考其他關聯表的主鍵，在參考的關聯表也可能擁有其他外來鍵，再參考到其他關聯表。如果將這些外來鍵的參考關係依序繪出，可以建立「外來鍵參考鏈」（Referential Chain），如下所示：

Projects → Departments → Employees

- 如果外來鍵參考最後回到原關聯表，稱為「外來鍵參考環」（Referential Cycle），例如：關聯表R1的參考鏈最後又回到R1，其外來鍵參考環，如下所示：

R1 → R2 → R3 → ... → Rn → R1



3-4-4 參考完整性 – 外來鍵使用規則

- 外來鍵的更新規則（**Update Rule**）：如果一個值組擁有外來鍵，當合法使用者試圖在更新或新增值組時，更改外來鍵的值，資料庫管理系統會如何處理？
- 外來鍵的刪除規則（**Delete Rule**）：如果一個值組擁有外來鍵，當合法使用者試圖刪除參考的主鍵時，資料庫管理系統會怎麼處理？



3-4-4 參考完整性 – 處理違反參考完整性

- 在刪除參考主鍵或更新外來鍵時會導致違反參考完整性，資料庫管理系統可能有三種處理方式，如下所示：
 - 限制性處理方式（**Restricted**）：拒絕刪除或更新操作。
 - 連鎖性處理方式（**Cascades**）：連鎖性處理方式是當更新或刪除時，需要作用在所有影響的外來鍵，否則拒絕此操作。例如：在刪除客戶時，所有外來鍵參考的訂單資料也需一併刪除，當更改訂單項目編號，則所有訂單中擁有此項目的外來鍵也需一併更改。
 - 空值化處理方式（**Nullifies**）：將所有可能的外來鍵都設為空值，否則拒絕此操作。



3-4-5 其他完整性限制條件 – 說明

- 資料庫管理師除了前述完整性限制條件外，還可以依照實際需求在基底關聯表的屬性新增額外的完整性限制條件。通常所有導出關聯表也都會繼承在基底關聯表設定的完整性條件，這些額外條件是在關聯表新增、刪除和更新資料時，觸發的一些額外檢查條件。
- 「語意完整性」（**Semantic Integrity**）是大部分資料庫管理系統都支援的完整性條件，它是屬性內容的一些限制條件，可以檢查關聯表值組的屬性是否為合法資料。



3-4-5 其他完整性限制條件 – 語意完整性

- 空值限制條件（**Null Constraint**）：限制屬性值不可為空值，也就是說，此屬性一定要輸入資料。例如：學生 **Students** 關聯表一定需要輸入姓名 **name** 屬性。
- 預設值（**Default Value**）：如果新增時沒有輸入指定的屬性值，值組的屬性會填入預設的指定值，其主要目的是避免屬性為空值。例如：員工 **Employees** 關聯表的部門 **department** 屬性如果沒有輸入，預設填入【業務部】。
- 檢查限制條件（**Check Constraint**）：一個布林值的邏輯運算式，輸入的屬性值一定需要滿足運算式，即邏輯運算式為真（**true**）。



3-5 關聯式資料庫 – 說明

- 關聯式資料庫（Relational Database）是一種使用關聯式資料庫模型的資料庫，它是由多個已正規化的關聯表所組成。
- 在關聯表間是使用外來鍵與參考主鍵的資料值來建立連接，以便實作一對一、一對多和多對多的關聯，詳細說明請參閱〈第4章：資料庫設計與實體關聯模型〉和〈第5章：關聯表的正規化〉，當然，如果只擁有一個關聯表，也屬於合法的關聯式資料庫。



3-5 關聯式資料庫 – 關聯表綱要

- 關聯式資料庫綱要 (Relational Database Schema) 是一組關聯表綱要和各綱要附屬的完整性條件。
- 例如：學校選課與排課系統資料庫是擁有四個關聯表的資料庫，如下所示：

Students (sid, name, birthday, tel)

Instructors (eid, name, rank, department)

Courses (c_no, title, credits)

Classes (eid, sid, c_no, time, room, grade)



3-5 關聯式資料庫 – 關聯表圖例

Students

<u>sid</u>	name	birthday	tel
S001	陳會安	1967/9/3	02-22222222
S002	江小魚	1978/2/2	03-33333333
S003	張三丰	1982/3/3	04-44444444
S004	李四方	1981/4/4	05-55555555
S005	陳允傑	1966/9/3	02-22222222

Instructors

<u>eid</u>	name	rank	department
E001	陳慶新	教授	CS
E002	楊金欉	副教授	CIS
E003	李鴻章	講師	MATH
E004	陳允傑	講師	CS

Courses

<u>c_no</u>	title	credits
CS101	計算機概論	4
CS121	離散數學	4
CS203	程式語言	3
CS213	物件導向程式設計	2
CS222	資料庫管理系統	3

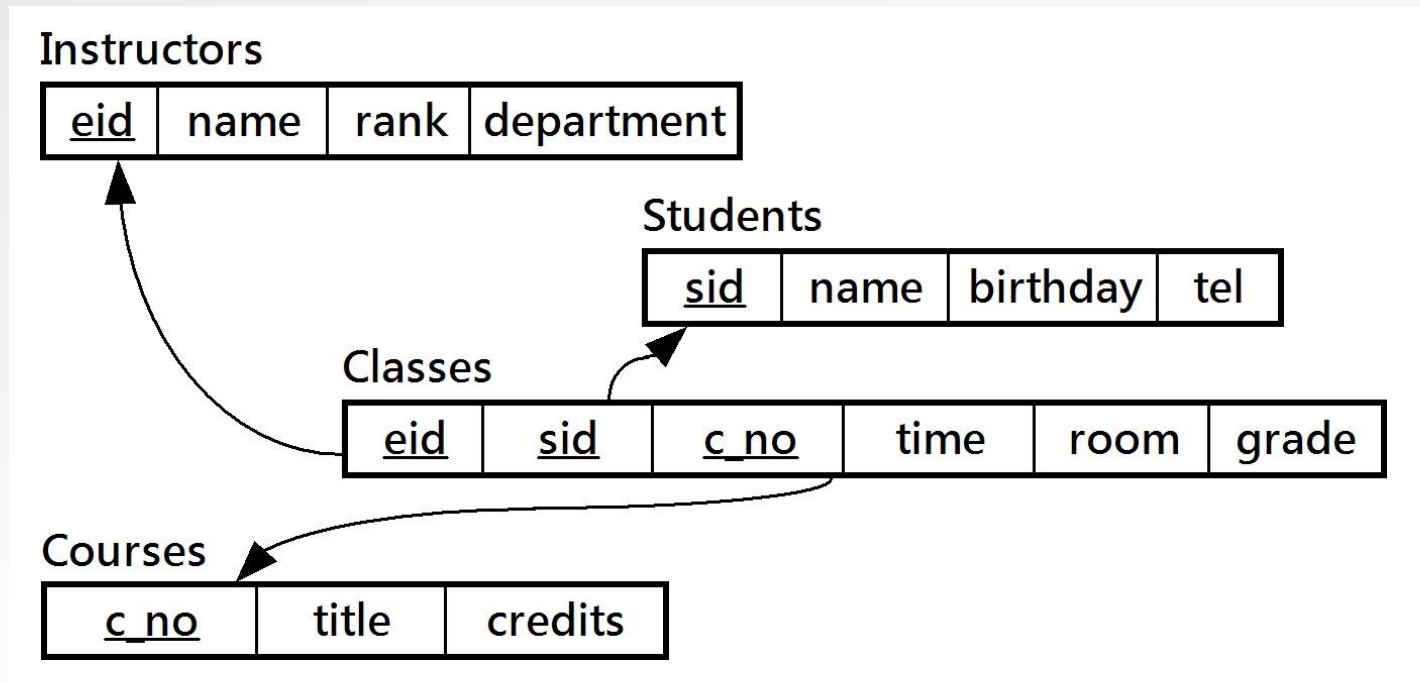
Classes

<u>eid</u>	<u>sid</u>	<u>c_no</u>	time	room	grade
E001	S001	CS101	12:00pm	180-M	85
E002	S003	CS121	8:00am	221-S	75
E003	S001	CS203	10:00am	221-S	68
E003	S002	CS203	14:00pm	327-S	85
E002	S001	CS222	13:00pm	100-M	78
E002	S002	CS222	13:00pm	100-M	58
E002	S004	CS222	13:00pm	100-M	92
E001	S003	CS213	9:00am	622-G	66
E003	S001	CS213	12:00pm	500-K	78



3-5 關聯式資料庫 - 外來鍵參考圖

- 在學校選課與排課系統資料庫的外來鍵參考圖，如下圖所示：





End
