

IBM® DB2® Universal Database



# SQL 入門

版本 7



IBM® DB2® Universal Database



# SQL 入門

版本 7

使用此資訊及其支援的產品之前，請先閱讀第109頁的『附錄C. 注意事項』下的一般資訊。

本文件含有 IBM 的所有權資訊。它是依據軟體使用權同意書而提供的，並受到著作權法的保護。本書中的資訊不包括任何產品保證，且其陳述也不得延伸解釋。

在美國請撥 1-800-879-2755 或在加拿大則請撥 1-800-IBM-4YOU，以向您的 IBM 業務代表或 IBM 地區分公司訂購出版品。

當您傳送資訊給 IBM 時，您即授權予 IBM 以其認為適當的方式來使用或分送資訊，而不必對您負起任何責任。

© Copyright International Business Machines Corporation 1993, 2000. All rights reserved.

## 目錄

歡迎使用 . . . . .	v	第5章 表示式與次查詢 . . . . .	33
本書相關文件 . . . . .	v	純量全選 . . . . .	33
印刷體例 . . . . .	v	轉換資料類型 . . . . .	33
<b>第1章 關聯式資料庫與 SQL . . . . .</b>	<b>1</b>	CASE 表示式 . . . . .	34
<b>第2章 組織資料 . . . . .</b>	<b>3</b>	表格表示式 . . . . .	35
表格 . . . . .	3	巢狀表格表示式 . . . . .	35
概略表 . . . . .	3	共用表格表示式 . . . . .	36
綱目 . . . . .	4	相關名稱 . . . . .	37
資料類型 . . . . .	4	相關次查詢 . . . . .	38
<b>第3章 建立表格與概略表 . . . . .</b>	<b>9</b>	相關次查詢之實作 . . . . .	40
建立表格 . . . . .	9	<b>第6章 在查詢中使用運算子和述詞 . . . . .</b>	<b>45</b>
插入資料 . . . . .	10	利用集合運算子合併查詢 . . . . .	45
變更資料 . . . . .	12	UNION 運算子 . . . . .	45
刪除資料 . . . . .	12	EXCEPT 運算子 . . . . .	46
建立概略表 . . . . .	13	INTERSECT 運算子 . . . . .	47
使用概略表處理資料 . . . . .	14	述詞 . . . . .	48
<b>第4章 使用 SQL 陳述式存取資料 . . . . .</b>	<b>17</b>	使用 IN 述詞 . . . . .	48
與資料庫連接 . . . . .	18	使用 BETWEEN 述詞 . . . . .	48
調查錯誤 . . . . .	18	使用 LIKE 述詞 . . . . .	49
選取直欄 . . . . .	19	使用 EXISTS 述詞 . . . . .	49
選取橫列 . . . . .	20	量化述詞 . . . . .	50
排序橫列 . . . . .	22	<b>第7章 進階 SQL . . . . .</b>	<b>51</b>
除去重複橫列 . . . . .	23	利用限制及觸發施行業務規則 . . . . .	51
運算次序 . . . . .	24	鍵 . . . . .	51
使用表示式來計算值 . . . . .	24	唯一限制 . . . . .	52
命名表示式 . . . . .	25	參照完整性限制 . . . . .	52
從一個以上的表格中選取資料 . . . . .	25	表格核對限制 . . . . .	53
使用次查詢 . . . . .	26	觸發函式 . . . . .	54
使用函數 . . . . .	27	結合 . . . . .	57
直欄函數 . . . . .	27	複雜查詢 . . . . .	62
純量函數 . . . . .	28	ROLLUP 和 CUBE 查詢 . . . . .	62
表格函數 . . . . .	29	遞迴查詢 . . . . .	62
分組 . . . . .	29	OLAP 函數 . . . . .	62
WHERE 子句與 GROUP BY 子句的搭配使 用 . . . . .	30	<b>第8章 自行設定及加強資料處理 . . . . .</b>	<b>65</b>
在 GROUP BY 子句後面使用 HAVING 子 句 . . . . .	30	使用者定義的類型 . . . . .	65
		使用者定義的函數 . . . . .	66
		大型物件 (LOB) . . . . .	67
		處理大型物件 (LOB) . . . . .	67

特別暫存區 . . . . .	68	Adamson 相片 . . . . .	87
目錄概略表簡介 . . . . .	68	Adamson 履歷表 . . . . .	87
從系統目錄中選取橫列 . . . . .	68	Walker 相片 . . . . .	88
		Walker 履歷表 . . . . .	89
<b>附錄A. 範例資料庫表格 . . . . .</b>	<b>71</b>		
範例資料庫 . . . . .	72	<b>附錄B. 使用 DB2 檔案庫 . . . . .</b>	<b>91</b>
建立範例資料庫 . . . . .	72	DB2 PDF 檔案與列印的書籍 . . . . .	91
若要消除範例資料庫 . . . . .	72	DB2 資訊 . . . . .	91
CL_SCHED 表格 . . . . .	72	列印 PDF 書籍 . . . . .	99
DEPARTMENT 表格 . . . . .	73	訂購印刷書籍 . . . . .	100
EMPLOYEE 表格 . . . . .	73	DB2 線上文件 . . . . .	101
EMP_ACT 表格 . . . . .	76	存取線上說明 . . . . .	101
EMP_PHOTO 表格 . . . . .	78	檢視線上資訊 . . . . .	103
EMP_RESUME 表格 . . . . .	78	使用 DB2 精靈 . . . . .	105
IN_TRAY 表格 . . . . .	79	設定文件伺服器 . . . . .	106
ORG 表格 . . . . .	79	搜尋線上資訊 . . . . .	107
PROJECT 表格 . . . . .	79		
SALES 表格 . . . . .	81	<b>附錄C. 注意事項 . . . . .</b>	<b>109</b>
STAFF 表格 . . . . .	82	商標 . . . . .	111
STAFFG 表格 . . . . .	83		
資料類型為 BLOB 和 CLOB 的範例檔案 . . . . .	84	<b>索引 . . . . .</b>	<b>113</b>
Quintana 相片 . . . . .	84		
Quintana 履歷表 . . . . .	85	<b>洽詢 IBM . . . . .</b>	<b>119</b>
Nicholls 相片 . . . . .	86	產品資訊 . . . . .	119
Nicholls 履歷表 . . . . .	86		

---

## 歡迎使用

本書的目的在於介紹使用者如何使用結構化查詢語言 (SQL) 及關聯式資料庫。書中將：

- 討論 DB2 產品中所使用的 SQL 基本概念。
- 說明如何執行資料庫處理作業。
- 利用簡單的範例示範作業。

如果您是系統管理者，在您嘗試執行本書的任何範例之前，您應：

- 針對您的作業系統依照快速入門一書中所列的內容來安裝及架構伺服器。使用「第一個步驟」選項，建立 SAMPLE 資料庫。您也可以從指令行提示建立 SAMPLE 資料庫。請參閱 *SQL Reference*，取得詳細資訊。注意事項：請勿將您的資料放入 DB2 SAMPLE 資料庫中。
- 使用 快速入門 一書中的指示，建立 DB2 管理者使用者 ID。

如果您不是系統管理者，請確定您具有有效的使用者 ID 及適當的權限與專用權，可以存取 SAMPLE 資料庫。

---

## 本書相關文件

下列出版品將對您很有幫助：

<i>快速入門</i>	內含安裝與使用資料庫管理程式所需的資訊。
<i>SQL Reference</i>	含有 SQL 參考資訊。
<i>Administration Guide</i>	含有設計、實作和維護從區域或主/從環境中存取資料庫所需的資訊。
<i>Application Development Guide</i>	討論應用程式開發處理，及如何撰寫、編譯及執行使用內含的 SQL 的應用程式來存取資料庫，或使用 SQL 程序語言 (或其它支援的程式設計語言)，來當成 DB2 儲存程序執行。

---

## 印刷體例

以下是本書使用的印刷慣例。

<b>粗體</b>	在範例中，粗體表示系統預先定義的命令和關鍵字。
<i>斜體</i>	表示下列其中一個意思： <ul style="list-style-type: none"><li>• 新的術語</li><li>• 參考其他資訊來源</li></ul>

---

大寫

表示下列其中一個意思：

- 系統預先定義的命令和關鍵字
  - 特定資料值或直欄名稱的範例
-



---

## 第1章 關聯式資料庫與 SQL

在一個關聯式資料庫中，資料是儲存於表格內。表格是由許多橫列與直欄所組成。請參閱第3頁的圖1，以取得表格的圖形式範例。直欄（垂直的）及橫列（水平的）均已在圖中標示。結構化查詢語言 (SQL) 是用來指定直欄、表格及其間的各種關係，以擷取或更新資料。

SQL 是關聯式資料庫中用來定義與處理資料的一種標準化語言。SQL 陳述式是由資料庫管理程式所執行。資料庫管理程式是用來管理資料的一個電腦程式。

分割的關聯式資料庫是一種關聯式資料庫，其中資料的管理橫跨多個分割區（亦稱為節點）。分割區的簡單說法即是將每一個分割區視為一個實體電腦。本書的重點放在單一分割區資料庫上。

您可以使用「命令行處理器」(CLP) 或「命令中心」(CC) 之類的介面，經由 交談式 SQL，存取範例資料庫並試著執行本書中的所有範例。



---

## 第2章 組織資料

本章將說明表格、概略表及綱目等重要觀念。這是一般概觀，顯示關聯式資料庫的各種建置區塊間的連接。最後一節對常用的重要資料類型將有簡短的討論。

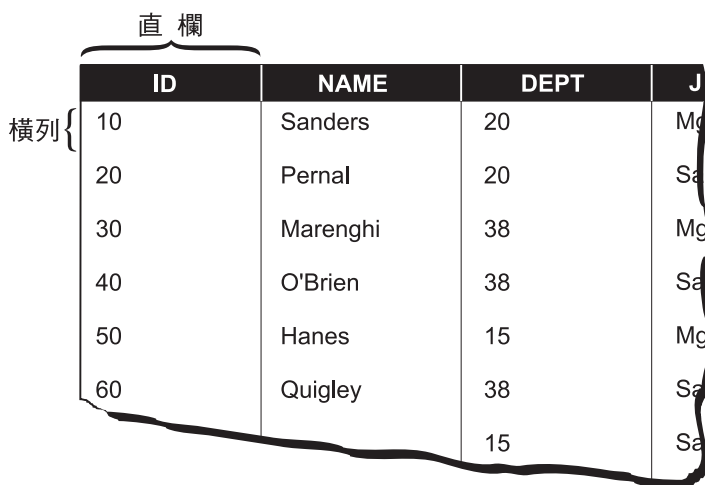
---

### 表格

表格是由已定義的直欄數及可變的橫列數所組成的邏輯結構。直欄是一組相同資料類型的值。橫列是一串值的序列，由表格中的單一記錄組成。橫列在表格中不一定會排序。若想替結果集合排序，您必須在用來從表格中選取資料的 SQL 陳述式中明確地指定次序。每個直欄與橫列的交集處即為一個特定的資料項目，稱為值。在圖1中，'Sanders' 即為表格中的一個值。

基本表格中含有使用者資料，並且是以 CREATE TABLE 陳述式建立的。結果表格是由資料庫管理程式從一或多個基本表格所選取或產生，以符合某一查詢之一組橫列。

圖1是表格中的某一段。當中標示出直欄與橫列。



The diagram shows a table with four columns and seven rows. The columns are labeled 'ID', 'NAME', 'DEPT', and 'J'. The rows contain data: (10, Sanders, 20, Mg), (20, Pernal, 20, Sa), (30, Marenghi, 38, Mg), (40, O'Brien, 38, Sa), (50, Hanes, 15, Mg), (60, Quigley, 38, Sa), and (15, Sa). A bracket above the columns is labeled '直欄' (Columns), and a bracket to the left of the rows is labeled '橫列' (Rows).

ID	NAME	DEPT	J
10	Sanders	20	Mg
20	Pernal	20	Sa
30	Marenghi	38	Mg
40	O'Brien	38	Sa
50	Hanes	15	Mg
60	Quigley	38	Sa
		15	Sa

圖 1. 圖示化的表格

---

### 概略表

概略表是檢視一個或多個表格中之資料的另一種方法。它是表格中的動態視窗。

概略表可讓多位使用者看到相同資料的不同呈現。例如，可能會有許多使用者同時存取表格中的員工資料。經理人員會查看有關他(她)的員工的資料，但不會查看其它部門的員工資料。招募主管會查看所有員工的到職日，但不會查看薪資，而財務主管則會查看薪資而不會查看到職日。這些使用者中的每一個都可以操作由實際表格所衍生出來的概略表。每一個概略表都會以表格的方式呈現，並有其自己的名稱。

使用概略表的好處之一，是您可以用它來控制敏感資料的存取。因此，不同的使用者可以存取的資料直欄與橫列也不同。

---

## 綱目

綱目是已命名物件(例如，表格及概略表)的集合。綱目會提供資料庫中物件的邏輯分類。

當您建立表格、概略表或任何其它已命名的物件時，綱目即會暗中建立。您也可以使用 `CREATE SCHEMA` 陳述式明顯地建立。

當您建立已命名的物件時，您可以將它的名稱與特定綱目的名稱連結。已命名的物件名稱分為兩部份，名稱的第一部份是物件所屬的綱目名稱。如果您沒有指定綱目名稱，物件就會指派給預設綱目。(預設綱目名稱是執行陳述式的使用者授權 ID)。

對於交談式 SQL (本書中用來執行範例的方法)來說，授權 ID 即是使用 `CONNECT` 陳述式所指定的使用者 ID。例如，如果表格名稱是 `STAFF`，且指定的使用者 ID 是 `USERXYZ`，則完整的表格名稱是 `USERXYZ.STAFF`。有關 `CONNECT` 陳述式的詳細資訊，請參閱第18頁的『與資料庫連接』。

某些綱目名稱被保留。例如，內建函數屬於 `SYSIBM` 綱目，而預先安裝的使用者定義的函數則屬於 `SYSFUN` 綱目。有關 `CREATE SCHEMA` 陳述式的詳細資訊，請參閱 *SQL Reference*。

---

## 資料類型

資料類型定義了常數、直欄、主變數、函數、表示式及特別暫存區的可接受值。本節將說明範例中提到的資料類型。如需其它資料類型的完整列示和說明，請參閱 *SQL Reference*。

### 字串

字串是一系列的位元組。字串長度即為系列中的位元組數。若長度為零，則此值稱為空字串。

### 固定長度字串

CHAR(x) 是一個固定長度字串。長度屬性 x 必須介於 1 和 254 (含) 之間。

### 可變長度字串

可變長度字串有三種類型：VARCHAR、LONG VARCHAR 和 CLOB。

VARCHAR(x) 這種類型屬於可變長度字串，長度為 9 的字串可插入 VARCHAR(15) 內，但其長度仍為 9。

有關 CLOB 的詳細資訊，請參閱第67頁的『大型物件 (LOB)』。

### 圖形字串

圖形字串是一系列的二位元組字元資料。

#### 固定長度圖形字串

GRAPHIC(x) 是一個固定長度字串。長度屬性 x 必須介於 1 和 127 (含) 之間。

#### 可變長度圖形字串

可變長度圖形字串有三種類型：VARGRAPHIC、LONG VARGRAPHIC 和 DBCLOB。有關 DBCLOB 的詳細資訊，請參閱第67頁的『大型物件 (LOB)』。

### 二進位字串

二進位字串是一系列的位元組。它是用來存放諸如圖片等非傳統資料。二進位大型物件 (BLOB) 即為一個二進位字串。如需詳細資訊，請參閱第67頁的『大型物件 (LOB)』。

### 數字

所有數字都有一個正負號及精準度。所謂精準度是指正負號之外的位元數或位數。

#### SMALLINT

SMALLINT (小整數) 是一個精準度為 5 位數的兩個位元組整數。

#### INTEGER

INTEGER (大整數) 是一個精準度為 10 位數的四個位元組整數。

#### BIGINT

BIGINT (大整數) 是一個八位元的整數，精準度為 19 位數。

REAL REAL (單準度浮點數字) 是實數的 32 位元近似值。

## DOUBLE

*DOUBLE* (倍準度浮點數字) 是實數的 64 位元近似值。DOUBLE 也稱為 FLOAT。

## DECIMAL(p,s)

*DECIMAL* 是一個小數。小數點位置是由這個數的**精準度** (*p*) 和**小數位數** (*s*) 所決定。精準度是指數字的總位數，它必須小於 32。小數位數是指小數部分的數字位數，它必小於等於精準度的值。若未指定精準度和小數位數，則小數值的預設值為精準度 5，小數位數 0。

## 日期時間值

日期時間值是日期、時間及時間戳記 (14 位數的字串，表示有效的日期及時間，格式為 *yyyymmddhhmmss*) 的表示法。儘管日期時間值既非字串也非數字，但它可用於某些算術及字串運算上，且與某些字串相容。<sup>1</sup>

**日期** *日期* 是一個三部分的值 (年，月，日)。

**時間** *時間* 是一個三部分的值 (時，分，秒)，指定 24 小時制的時間。

### 時間戳記

*時間戳記* 是一個七部分的值 (年，月，日，時，分，秒，微秒)，指定一個時間和日期。

## Null 值

*NULL* 值是一個特殊的值，不同於所有非 *NULL* 的值。它表示橫列中的該直欄不存在其它值。所有資料類型都有 *NULL* 值。

下表標示了範例中所使用的資料類型之性質。所有數字資料類型都被定義於一特定範圍內。這個表格也列出了數字資料類型的範圍。您可以使用本表來快速查閱特定資料類型的適當用法。

資料類型	類型	性質	範例或範圍
CHAR(15)	固定長度字串	254 的長度上限	'Sunny day'
VARCHAR(15)	可變長度字串	32672 的長度上限	'Sunny day'
SMALLINT	數字	長度為 2 位元組精準度為 5 位數	範圍為 -32768 到 32767
INTEGER	數字	長度為 4 位元組精準度為 10 位數	範圍為 -2147483648 到 2147483647

1. 本書採用 ISO 的日期時間值表示法。

資料類型	類型	性質	範例或範圍
BIGINT	數字	長度為 8 位元組精確度為 19 位數	範圍是 -9223372036854775808 到 9223372036854775807
REAL	數字	單一精確度浮點 32 位元近似值	範圍是 -3.402E+38 到 -1.175E-37 或 1.175E-37 到 -3.402E+38 或零
DOUBLE	數字	倍準度浮點 64 位元近似值	範圍是 -1.79769E+308 到 -2.225E-307 或 2.225E-307 到 1.79769E+308 或零
DECIMAL(5,2)	數字	精確度是 5 小數位數是 2	範圍是 -10**31+1 到 10**31-1
DATE	日期時間	三部分值	1991-10-27
TIME	日期時間	三部分值	13.30.05
TIMESTAMP	日期時間	七部分值	1991-10-27-13.30.05.000000

請參閱 *SQL Reference* 中的「資料類型相容性」表格，取得其餘資訊。





---

## 第3章 建立表格與概略表

本章將描述如何建立與處理 DB2 Universal Database 中的表格和概略表。表格和概略表的關係將以圖表和範例顯示。

本章內容涵括了：

- 建立表格及建立概略表
- 插入資料
- 變更資料
- 刪除資料
- 使用概略表處理資料

---

### 建立表格

您可使用 CREATE TABLE 陳述式來建立自己的表格，指定直欄名稱和類型以及限制 (*constraint*)。限制將在第51頁的『利用限制及觸發施行業務規則』部分討論。

以下的陳述式會建立一個名為 PERS 的表格，它和 STAFF 表格類似，但多了一個出生日期的直欄。

```
CREATE TABLE PERS
  (ID          SMALLINT NOT NULL,
   NAME       VARCHAR(9),
   DEPT       SMALLINT WITH DEFAULT 10,
   JOB        CHAR(5),
   YEARS     SMALLINT,
   SALARY     DECIMAL(7,2),
   COMM      DECIMAL(7,2),
   BIRTH_DATE DATE)
```

這個陳述式會建立一個沒有資料的空白表格。下一節將介紹如何在表格中插入資料。

如範例所示，您必須替每個直欄指定一個名稱和一種資料類型。資料類型已在第4頁的『資料類型』部分討論。NOT NULL 為選用的，可用來指定某個直欄不能有 NULL 值。預設值也是選用的。

CREATE TABLE 陳述式中還有許多選項可以指定，如唯一限制 (*unique constraint*) 或參照限制 (*referential constraint*)。如需所有選項的詳細資訊，請參閱 *SQL Reference* 中的 CREATE TABLE 陳述式。

---

## 插入資料

當您建立一個新表格時，表格內不會有任何資料。若想在表格中插入新的橫列，您必須使用 `INSERT` 陳述式。這個陳述式有兩個一般格式：

- 其中一種格式是使用 `VALUES` 子句來指定一或多個橫列的直欄值。下面三個例子即是使用此種一般格式在表格內插入資料。
- 另一種格式是不指定 `VALUES`，而指定全選 (*fullselect*)，從其它表格及 (或) 概略表所包含的橫列中識別直欄。

全選是一個選取 (`select`) 陳述式，用在 `INSERT` 或 `CREATE VIEW` 陳述式中，或用在述語 (`predicate`) 後面。以括弧括住的全選，通常被視為 次查詢。

在您所插入的每個橫列中，每一直欄您都必須提供一個值或接受預設值，視您在建立表格時所選取的預設選項而定。各種資料類型的預設值將在 *SQL Reference* 中討論。

以下的陳述式使用 `VALUES` 子句，在 `PERS` 表格中插入一列資料：

```
INSERT INTO PERS
VALUES (12, 'Harris', 20, 'Sales', 5, 18000, 1000, '1950-1-1')
```

下面的陳述式會使用 `VALUES` 子句，在 `PERS` 表格中插入三個橫列，而表格中只有 `ID`、名稱及工作為已知。如果直欄已定義為 `NOT NULL` 且沒有預設值，則您必須指定一個值給它。

`CREATE TABLE` 陳述式中的直欄定義中的 `NOT NULL` 子句，可以利用 `WITH DEFAULT` 擴充。如果一個直欄定義為 `NOT NULL WITH DEFAULT` 或常數預設值 (如 `WITH DEFAULT 10`)，且您未在直欄列示中指定該欄，則會將預設值插入要插入的橫列中的那直欄中。例如，在 `CREATE TABLE` 陳述式中，`DEPT` 直欄只有指定預設值，且定義為 10。因此，部門編號 (`DEPT`) 設定為 10，其它未明確提供值的直欄則設定為 `NULL`。

```
INSERT INTO PERS (NAME, JOB, ID)
VALUES ('Swagerman', 'Prgmr', 500),
       ('Limoges', 'Prgmr', 510),
       ('Li', 'Prgmr', 520)
```

以下陳述式會傳回插入的結果：

```
SELECT *
FROM PERS
ID      NAME      DEPT  JOB   YEARS  SALARY  COMM  BIRTH_DATE
-----
12 Harris      20 Sales    5 18000.00 1000.00 01/01/1950
```

500 Swagerman	10 Prgmr	-	-	-	-
510 Limoges	10 Prgmr	-	-	-	-
520 Li	10 Prgmr	-	-	-	-

請注意，在這個例子中，並非每個直欄均有指定值。NULL 值是以破折號 (-) 顯示。要讓這樣的設定生效，直欄名稱列示的次序與資料類型必須與 VALUE 子句中提供的值對應。若直欄名稱列示被略去 (如第一個例子所示)，則 VALUE 後面的資料值列示，其次序必須與它被插入所在的表格中的直欄相同，且值的數目也必須與表格中的直欄數相等。

每個值都必須與將插入所在的直欄之資料類型相容。若某個直欄定義為可有 NULL 值，且該欄的值未指定，則會將值 NULL 給予插入的橫列中的那直欄。

下面的例子會在 YEARS, COMM, BIRTH\_DATE 等三個直欄中插入 NULL 值，因該橫列中的這些直欄未指定值。

```
INSERT INTO PERS (ID, NAME, JOB, DEPT, SALARY)
VALUES (410, 'Perna', 'Sales', 20, 20000)
```

INSERT 陳述式的第二種格式相當方便用來將另一個表格中的橫列中的值移入某個表格中。如前所述，這種格式不指定 VALUES，而是指定全選，從其它表格及 (或) 概略表所包含的橫列中識別直欄。

下面的例子從 STAFF 表格中選取屬於部門 38 的成員之資料，將其插入 PERS 表格內：

```
INSERT INTO PERS (ID, NAME, DEPT, JOB, YEARS, SALARY)
SELECT ID, NAME, DEPT, JOB, YEARS, SALARY
FROM STAFF
WHERE DEPT = 38
```

插入資料後，下面的 SELECT 陳述式將產生與 INSERT 陳述式中的全選相同的結果。

```
SELECT ID, NAME, DEPT, JOB, YEARS, SALARY
FROM PERS
WHERE DEPT = 38
```

結果如下：

ID	NAME	DEPT	JOB	YEARS	SALARY
30	Marengi	38	Mgr	5	17506.75
40	O'Brien	38	Sales	6	18006.00
60	Quigley	38	Sales	-	16808.30
120	Naughton	38	Clerk	-	12954.75
180	Abrahams	38	Clerk	3	12009.75

---

## 變更資料

您可使用 UPDATE 陳述式來變更表格中的資料。使用此陳述式，您可以針對符合 WHERE 子句搜尋條件的每一橫列來變更一或多個直欄的值。

以下的例子將會更新 ID 為 410 的員工的相關資訊：

```
UPDATE PERS
SET JOB='Prgmr', SALARY = SALARY + 300
WHERE ID = 410
```

SET 子句指定了要更新的直欄，並提供值。

WHERE 子句是選用的，它指定了要更新的橫列。若 WHERE 子句被略去，資料庫管理程式會以您提供的值，更新表格或概略表中的每個橫列。

在這個例子中，首先是替表格命名 (PERS)，接著為要更新的橫列指定一個條件。員工編號 410 的資訊已變更：員工的職稱變更為程式設計師 (Prgmr)，員工的薪資增加了 \$300。

您可以併入適用於兩個或更多橫列 WHERE 子句，變更多個橫列中的資料。下面的例子將每個銷售員的薪資增加 15%：

```
UPDATE PERS
SET SALARY = SALARY * 1.15
WHERE JOB = 'Sales'
```

---

## 刪除資料

DELETE 陳述式可讓您依 WHERE 子句所指定的搜尋條件，刪除表格中多列的資料。下面的例子刪除了員工 ID 為 120 的橫列資料：

```
DELETE FROM PERS
WHERE ID = 120
```

WHERE 子句是選用的，它指定了要刪除的橫列。若 WHERE 子句被略去，資料庫管理程式會刪除表格或概略表中的所有橫列。

您可以使用 DELETE 陳述式來刪除一個以上的橫列。下面的例子會刪除員工 DEPT 為 20 的所有橫列：

```
DELETE FROM PERS
WHERE DEPT = 20
```

當您刪除橫列時，您會除去整個橫列，而不是從中除去特定的直欄值。

若想刪除某一個表格的定義及其內容，請使用 *SQL Reference* 中所描述的 `DROP TABLE` 陳述式。

## 建立概略表

如第3頁的『概略表』中所討論的，概略表是檢視一或多個表格中的資料的另一種方法。利用建立概略表，您可以限制不同使用者可以查閱的資訊。以下的圖表顯示了概略表和表格的關係。

在圖2中，`View_A` 限制只能存取表格 `TABLE_A` 中的直欄 `AC1` 與 `AC2`。

`View_AB` 則容許存取 `TABLE_A` 中的 `AC3` 及 `TABLE_B` 中的 `BC2`。

藉由建立 `View_A`，您可以限制使用者對 `TABLE_A` 的存取，而藉由建立 `VIEW_AB`，您可以限制對這兩個表格中某些直欄的存取。

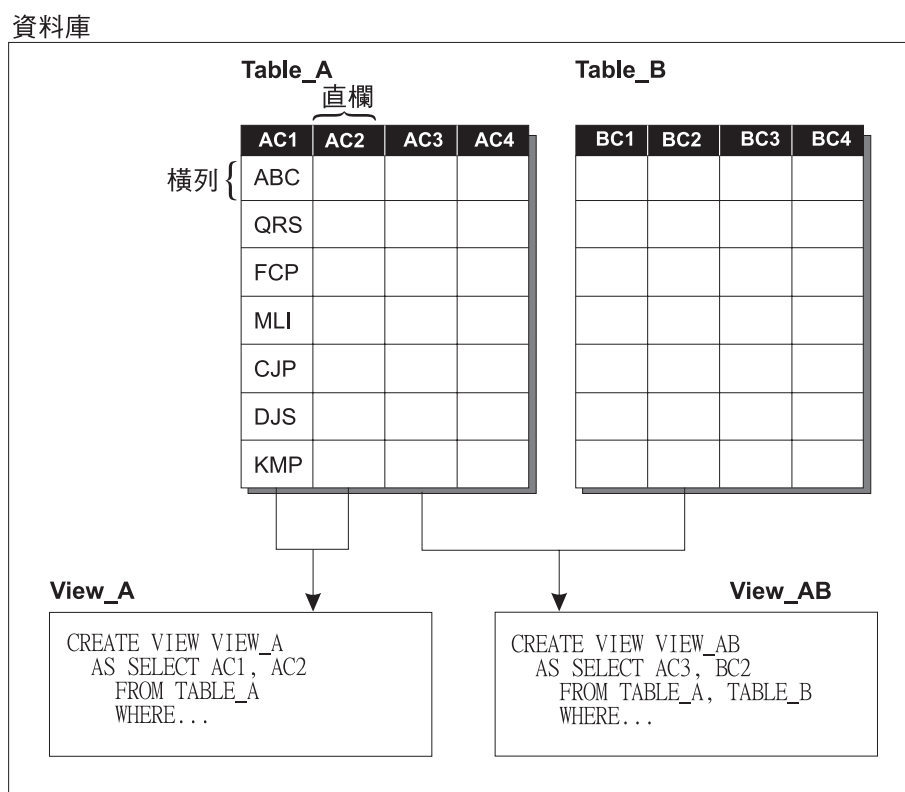


圖 2. 表格和概略表的關係

下面的陳述式會建立 STAFF 表格中部門編號為 20 的非經理人員概略表，其中基本表格中的薪資和佣金均未顯示。

```
CREATE VIEW STAFF_ONLY
AS SELECT ID, NAME, DEPT, JOB, YEARS
FROM STAFF
WHERE JOB <> 'Mgr' AND DEPT=20
```

建立此概略表之後，下面的陳述式即可顯示此概略表的內容：

```
SELECT *
FROM STAFF_ONLY
```

此陳述式會產生以下結果：

ID	NAME	DEPT	JOB	YEARS
20	Pernal	20	Sales	8
80	James	20	Clerk	-
190	Sneider	20	Clerk	8

在下面的範例中，我們可以使用 STAFF 及 ORG 表格來建立概略表，列出每一個部門的名稱及該部門的經理姓名。下列陳述式會建立此概略表：

```
CREATE VIEW DEPARTMENT_MGRS
AS SELECT NAME, DEPTNAME
FROM STAFF, ORG
WHERE MANAGER = ID
```

建立概略表時，您可使用 WITH CHECK OPTION 子句，透過概略表對表格的插入與更新作其他限制。這個子句會讓資料庫管理程式驗證概略表中的任何更新或插入皆符合概略表定義，且拒絕不符概略表定義的資料。如果您略去了這個子句，那麼插入與更新資料就不會和概略表定義比對。如需 WITH CHECK OPTION 作業方式的詳細資訊，請參閱 *SQL Reference* 中的 CREATE VIEW 陳述式。

## 使用概略表處理資料

和 SELECT 陳述式一樣，INSERT、DELETE 及 UPDATE 陳述式也適用於概略表，就像它是真實的表格一樣。這些陳述式可處理基本表格內的資料。因此您如果再次存取某個概略表，則會使用最新的基本表格來求值。如果您不使用 WITH CHECK OPTION 子句，則您使用概略表修改的資料可能就不會在您重複存取概略表時出現，因為資料可能已不再符合原始的概略表定義。

以下是適用於概略表 FIXED\_INCOME 的更新範例：

```

CREATE VIEW FIXED_INCOME (LNAME, DEPART, JOBTITLE, NEWSALARY)
AS SELECT NAME, DEPT, JOB, SALARY
FROM PERS
WHERE JOB <> 'Sales' WITH CHECK OPTION

UPDATE FIXED_INCOME
SET NEWSALARY = SALARY * 1.10
WHERE LNAME = 'Li'

```

前一個概略表的更新 (除 CHECK 選項外) 均和更新基本表格 PERS 相同：

```

UPDATE PERS
SET SALARY = SALARY * 1.10
WHERE NAME = 'Li'
AND JOB <> 'Sales'

```

請注意，由於是使用 CREATE VIEW FIXED\_INCOME 中的限制，JOB <> 'Sales' 的 WITH CHECK OPTION 建立概略表，因此當 Limoges 移到銷售員位置時，將不容許下列更新：

```

UPDATE FIXED_INCOME
SET JOBTITLE = 'Sales'
WHERE LNAME = 'Limoges'

```

由表示式 (如 SALARY + COMM 或 SALARY \* 1.25) 所定義的直欄無法被更新。如果您定義含有一或多個這類直欄的概略表，則擁有者在這些直欄上不具有 UPDATE 專用權。在含有這一類直欄的概略表中，不允許使用 INSERT 陳述式，但可使用 DELETE 陳述式。

試想一個 PERS 表格，其中沒有一個直欄被定義為 NOT NULL。您可以利用 FIXED\_INCOME 概略表，在 PER2 表格中插入橫列，即使它未包含基本表格 PERS 中的 ID、YEARS、COMM 或 BIRTHDATE 等。透過概略表中看不到的直欄會被設定為 NULL 或預設值，視情形而定。

然而，PERS 表格的確具有定義為 NOT NULL 的直欄 ID。如果您嘗試透過 FIXED\_INCOME 概略表插入一個橫列，系統會嘗試在透過概略表『看不到的』所有 PERS 直欄中插入 NULL 值。由於 ID 直欄並未含在概略表內且不允許使用 NULL 值，因此系統不容許透過概略表的插入。

有關修改概略表的規則及限制，請參照 *SQL Reference* 中的 CREATE VIEW 陳述式。





---

## 第4章 使用 SQL 陳述式存取資料

本節說明如何使用 SQL 陳述式連接資料庫及擷取資料。

在範例中，我們提出了要輸入的陳述式，且 (在大部份的情況下) 在對範例資料庫發出該陳述式後，會在陳述式後接著顯示結果。請注意，雖然此處的陳述式是以大寫顯示，但您可以隨意混合使用大小寫字元，除非是括在單引號 (') 或雙引號 (") 內。

DB2 Universal Database 隨附的 SAMPLE 資料庫是由數個表格組成，如第71頁的『附錄A. 範例資料庫表格』所列示的。您可以使用「第一個步驟」安裝發射台，來建立此資料庫。您也可以從指令行建立 SAMPLE 資料庫。如需詳細資訊，請參閱 *SQL Reference*。

請注意：其它的範例資料庫是隨附於 DB2 Universal Database，用來示範「資料倉儲中心」及 OLAP Starter Kit 功能性。本書中的範例僅使用一般的 SAMPLE 資料庫。

根據您資料庫的設定方式，您也許必須在表格名稱的，字首加上綱目名稱和句點。本書中所有例子的預設綱目均為 USERID。因此 ORG 這個表格限定的完整名稱為 USERID.ORG。請詢問您的管理員，看是否需要這麼做。

本章內容涵括了：

- 與資料庫連接
- 調查錯誤
- 選取直欄及選取橫列
- 排序橫列及除去重複橫列
- 運算次序
- 使用表示式來計算值
- 命名表示式
- 從一個以上的表格中選取資料
- 使用次查詢
- 使用函數
- 分組

---

## 與資料庫連接

在使用 SQL 陳述式來查詢或處理資料庫之前，您必須先連接到該資料庫。CONNECT 陳述式可使資料庫與某個使用者名稱產生關聯。

例如，若想連接到 SAMPLE 資料庫，請在 DB2 命令行處理器中鍵入以下命令：

```
CONNECT TO SAMPLE USER USERID USING PASSWORD
```

(請務必選擇在伺服器系統中有效的使用者 ID 和通行碼。)

在這個例子中，USER 為 USERID，而 USING 則為 PASSWORD。

以下的訊息告訴您已連接成功：

### 資料庫連接資訊

資料庫產品	= DB2/NT 7.1.0
SQL 授權 ID	= USERID
本端資料庫別名	= SAMPLE

連接成功之後，您就可以開始處理資料庫了。若需有關連接的進一步詳細資訊，請參照 *SQL Reference* 中的 CONNECT 陳述式。

---

## 調查錯誤

當您在鍵入範例時發生打字錯誤，或執行 SQL 陳述式時發生錯誤，資料庫管理程式都會傳回錯誤訊息。錯誤訊息包含一個訊息 ID、一段扼要說明，以及一個 SQLSTATE。

SQLSTATE 錯誤是 DB2 產品系列通用的錯誤碼。SQLSTATE 錯誤符合 ISO/ANSI SQL92 標準。

例如，如果 CONNET 陳述式中的使用者 ID 或通行碼有誤，資料庫管理程式會傳回訊息 ID SQL1403N 以及 SQLSTATE 08004。訊息如下：

```
SQL1403N 所提供的使用者名稱及 (或) 通行碼  
不正確。 SQLSTATE=08004
```

如需更詳細的資訊，可以鍵入一個問號 (?)，後面加上訊息 ID 或 SQLSTATE：

```
? SQL1403N  
或  
? SQL1403  
或  
? 08004
```

請注意，SQL1403N 訊息的倒數第二行提到 SQLCODE 為 -1403。SQLCODE 是一個產品特定的錯誤碼。以 N (通知) 或 C (嚴重) 結尾的訊息 ID 代表這是一個錯誤，其 SQLCODE 是一個負數。以 W (警告) 結尾的訊息 ID 代表的是警告，其 SQLCODE 是一個正數。

---

## 選取直欄

您可使用 **SELECT** 陳述式來選取表格中的特定直欄。在陳述式中指定直欄名稱的列示，中間以逗點分隔。這個列示稱為選取列示。

下面的陳述式會從 **SAMPLE** 資料庫的 **ORG** 表格中，選取部門名稱 (**DEPTNAME**) 和部門編號 (**DEPTNUMB**)：

```
SELECT DEPTNAME, DEPTNUMB
FROM ORG
```

上述陳述式會產生以下結果：

DEPTNAME	DEPTNUMB
Head Office	10
New England	15
Mid Atlantic	20
South Atlantic	38
Great Lakes	42
Plains	51
Pacific	66
Mountain	84

如果使用星號 (\*)，您就可以選取表格中所有的直欄。下面的例子列出 **ORG** 表格中所有的直欄和橫列：

```
SELECT *
FROM ORG
```

此陳述式會產生以下結果：

DEPTNUMB	DEPTNAME	MANAGER	DIVISION	LOCATION
10	Head Office	160	Corporate	New York
15	New England	50	Eastern	Boston
20	Mid Atlantic	10	Eastern	Washington
38	South Atlantic	30	Eastern	Atlanta
42	Great Lakes	100	Midwest	Chicago
51	Plains	140	Midwest	Dallas
66	Pacific	270	Western	San Francisco
84	Mountain	290	Western	Denver

---

## 選取橫列

若要從表格中選取特定橫列，請在 **SELECT** 陳述式後面加上 **WHERE** 子句，以指定要選取的橫列應符合的條件。從表格中選取橫列的準則稱為**搜尋條件**。

一個搜尋條件包含一或多個**述語**。一個述語指定了某個橫列的值為真或假 (或未知) 的條件。您可以利用下列基本述語，在 **WHERE** 子句中指定條件：

述語	函數
$x = y$	x 等於 y
$x \lt \gt y$	x 不等於 y
$x < y$	x 小於 y
$x > y$	x 大於 y
$x \leq y$	x 小於等於 y
$x \geq y$	x 大於等於 y
IS NULL/IS NOT NULL	測試是否為 NULL 值

當您建構搜尋條件時，請注意算術運算只能針對數字資料類型，且只有相容的資料類型才能作比較。例如，您必能拿字串和數字作比較。

如果您根據某個字元值來選取橫列，則該值必須以單引號括住 (如 **WHERE JOB = 'Clerk'**)，且每個字元值均需與其在資料庫中一模一樣的形式鍵入。如果該資料值在資料庫為小寫，而您鍵入大寫，則不會選取任何橫列。如果您根據某個數值來選取橫列，則該值不能以單引號括住 (如 **WHERE DEPT = 20**)。

以下的例子只會從 **STAFF** 表格中選取部門編號為 20 的橫列：

```
SELECT DEPT, NAME, JOB
FROM STAFF
WHERE DEPT = 20
```

此陳述式會產生以下結果：

```
DEPT  NAME      JOB
-----
    20 Sanders  Mgr
    20 Pernal  Sales
    20 James  Clerk
    20 Sneider Clerk
```

下一個例子使用 **AND** 來指定一個以上的條件。您可以指定的條件數沒有上限。這個例子會從 **STAFF** 表格中選取部門編號為 20 的職員：

```

SELECT DEPT, NAME, JOB
FROM STAFF
WHERE JOB = 'Clerk'
AND DEPT = 20

```

此陳述式會產生以下結果：

DEPT	NAME	JOB
20	James	Clerk
20	Sneider	Clerk

若未輸入任何值，且該直欄不支援預設值，則會造成 NULL 值的情形。當您特別將值設為 NULL，那麼同樣會造成 NULL 值。但這個情形只會發生於定義為支援 NULL 值的直欄中。定義及支援表格中的 NULL 值的方法，已在第9頁的『建立表格』中討論。

請使用述詞 IS NULL 及 IS NOT NULL，檢查是否有 NULL 值。

以下的陳述式將會列出其佣金未知的員工：

```

SELECT ID, NAME
FROM STAFF
WHERE COMM IS NULL

```

此陳述式會產生以下結果：

ID	NAME
10	Sanders
30	Marengi
50	Hanes
100	Plotz
140	Fraye
160	Molinare
210	Lu
240	Daniels
260	Jones
270	Lea
290	Quill

零這個值和 NULL 值並不同。下面的陳述式會選取佣金為零的員工：

```

SELECT ID, NAME
FROM STAFF
WHERE COMM = 0

```

由於範例表格中的 COMM 直欄中沒有一個值為零，因此傳回的結果集合為空集合。

下一個例子將選取 STAFF 表格中 YEARS 值大於 9 的所有橫列：

```
SELECT NAME, SALARY, YEARS
FROM STAFF
WHERE YEARS > 9
```

此陳述式會產生以下結果：

NAME	SALARY	YEARS
Hanes	20659.80	10
Lu	20010.00	10
Jones	21234.00	12
Quill	19818.00	10
Graham	21000.00	13

---

## 排序橫列

您可能希望傳回的資訊是以特定的次序排列。您可使用 ORDER BY 子句，依一或多個直欄中的值來排列資訊。

以下的陳述式顯示了依年資排列，部門編號為 84 的員工：

```
SELECT NAME, JOB, YEARS
FROM STAFF
WHERE DEPT = 84
ORDER BY YEARS
```

此陳述式會產生以下結果：

NAME	JOB	YEARS
Davis	Sales	5
Gafney	Clerk	5
Edwards	Sales	7
Quill	Mgr	10

請將 ORDER BY 指定為整個 SELECT 陳述式中的最後一個子句。這個子句所指定的直欄可以是表示式或是表格中的任何直欄。ORDER BY 子句中的直欄名稱不須在選取列示中指定。

您可以在 ORDER BY 子句中明確地指定 ASC 或 DESC，將橫列作升序或降序排列。如果沒有指定次序，那麼就會自動以升序排列橫列。以下的陳述式顯示了年資依降序排列，部門編號為 84 的員工：

```
SELECT NAME, JOB, YEARS
FROM STAFF
WHERE DEPT = 84
ORDER BY YEARS DESC
```

此陳述式會產生以下結果：

NAME	JOB	YEARS
Quill	Mgr	10
Edwards	Sales	7
Davis	Sales	5
Gafney	Clerk	5

您可以利用字元值與數值為橫列排序。以下的陳述式顯示了依姓名字母順序排列，部門編號為 84 的員工：

```
SELECT NAME, JOB, YEARS
FROM STAFF
WHERE DEPT = 84
ORDER BY NAME
```

此陳述式會產生以下結果：

NAME	JOB	YEARS
Davis	Sales	5
Edwards	Sales	7
Gafney	Clerk	5
Quill	Mgr	10

---

## 除去重複橫列

使用 SELECT 陳述式時，您也許不希望傳回的資料有重複的情形。例如，STAFF 表格中有個 DEPT 直欄，其中許多部門編號列出多次，而 JOB 直欄中則有許多職稱列出多次。

若要除去重複的橫列，請在 SELECT 子句中使用 DISTINCT 選項。例如，如果在陳述式中插入 DISTINCT，則部門中的每個職稱只會列出一一次：

```
SELECT DISTINCT DEPT, JOB
FROM STAFF
WHERE DEPT < 30
ORDER BY DEPT, JOB
```

此陳述式會產生以下結果：

DEPT	JOB
10	Mgr
15	Clerk
15	Mgr
15	Sales
20	Clerk
20	Mgr
20	Sales

DISTINCT 會將 SELECT 陳述式所指定的一組直欄中含有重複資料的所有橫列刪除。

---

## 運算次序

考慮運算的次序是非常重要的。一個子句的輸出是下一個子句的輸入，如下面列示中所述。第25頁的『命名表示式』是一個注意到運算次序的例子。

以 DB2 程式碼執行的作業不一定要依照下面的作業順序。這裡只是提供您簡單的說明，讓您能夠對查詢有更多的聯想。運算的順序如下：

1. FROM 子句
2. WHERE 子句
3. GROUP BY 子句
4. HAVING 子句
5. SELECT 子句
6. ORDER BY 子句

---

## 使用表示式來計算值

表示式是併入陳述式內的一個計算或函數。下面的陳述式會算出部門編號 38 的員工，若每人得到 \$500 的紅利，則每個員工的薪資為何：

```
SELECT DEPT, NAME, SALARY + 500
FROM STAFF
WHERE DEPT = 38
ORDER BY 3
```

結果如下：

DEPT	NAME	3
38	Abrahams	12509.75
38	Naughton	13454.75
38	Quigley	17308.30
38	Marengi	18006.75
38	O'Brien	18506.00

請注意，第三個直欄的直欄名稱是一個數字。這是由系統所產生的數字，因為 SALARY+500 並非指定某個直欄名稱。等一下這個數字將會用在 ORDER BY 子句中，代表第三個直欄。第25頁的『命名表示式』會討論如何替表示式取一個有意義的名稱。

您可以使用加 (+)、減 (-)、乘 (\*)、除 (/) 等基本算術運算子，撰寫算術表示式。



運算子可操作許多類型的運算元的數值，以下列出部分運算元：

- 直欄名稱 (如 RATE \* HOURS)
- 常數值 (如 RATE \* 1.07)
- 純量函數 (如 LENGTH(NAME) + 1)

---

## 命名表示式

選用的 AS 子句能讓您替表示式指定一個有意義的名稱，回頭再看表示式時較容易讀得懂。您可以使用 AS 子句，替選取列示中的任何項目指定一個名稱。

以下陳述式會顯示薪資加上佣金少於 \$13,000 的所有員工。SALARY + COMM 這個表示式的名稱為 PAY：

```
SELECT NAME, JOB, SALARY + COMM AS PAY
FROM STAFF
WHERE (SALARY + COMM) < 13000
ORDER BY PAY
```

此陳述式會產生以下結果：

NAME	JOB	PAY
Yamaguchi	Clerk	10581.50
Burke	Clerk	11043.50
Scoutten	Clerk	11592.80
Abrahams	Clerk	12246.25
Kermisch	Clerk	12368.60
Ngan	Clerk	12714.80

利用 AS 子句，您可以在 ORDER BY 子句中使用某一的直欄名稱，而非系統所產生的數字。在這個例子中，我們在 WHERE 子句中比較 (SALARY + COMM) 和 13000，而非使用 PAY 這個名稱作比較。這是一個運算次序的結果。WHERE 子句是在 (SALARY + COMM) 被命名為 PAY 之前先被演算，因為 SELECT 子句是在 WHERE 子句後執行。因此述語中無法使用 PAY。

---

## 從一個以上的表格中選取資料

您可以使用 SELECT 陳述式來產生包含兩個以上的表格中的資訊的報表。這通常被稱為結合。例如，您可以結合 STAFF 和 ORG 表格中的資料，形成一個新表格。若要結合兩個表格，請在 SELECT 子句中指定您想顯示的直欄，在 FROM 子句中指定表格名稱，在 WHERE 子句中指定搜尋條件。WHERE 子句是選用的。

下一個例子會結合每個經理的姓名與部門名稱。您必須從兩個表格中選取資訊，因為員工資訊 (STAFF 表格) 和部門資訊 (ORG 表格) 是分開存放的。下面這個查詢會分別從 STAFF 和 ORG 表格中選取 NAME 和 DEPTNAME 直欄。搜尋條件會將選擇範圍縮小到 MANAGER 直欄中的值和 ID 直欄中的值相等的橫列：

```

SELECT DEPTNAME, NAME
FROM ORG, STAFF
WHERE MANAGER = ID

```

圖3說明如何比較兩個不同表格中的直欄。框起來的值代表符合搜尋條件。

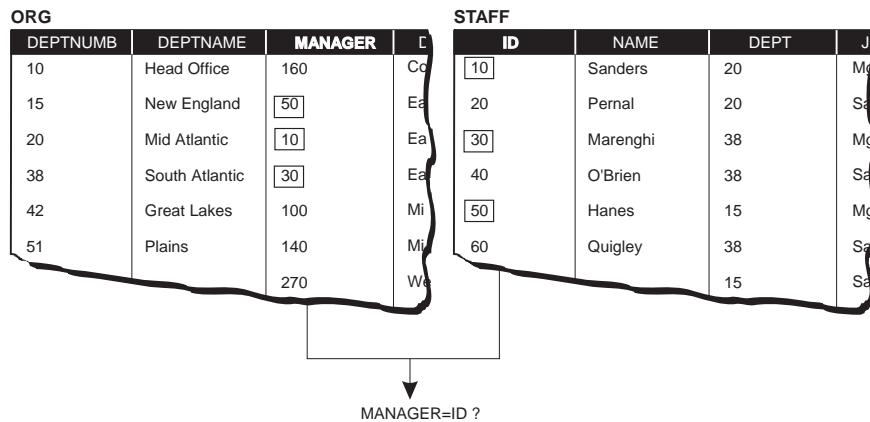


圖 3. 從 *STAFF* 和 *ORG* 表格中選取

SELECT 陳述式會產生以下結果：

```

DEPTNAME      NAME
-----
Mid Atlantic   Sanders
South Atlantic Marenghi
New England   Hanes
Great Lakes   Plotz
Plains        Fraye
Head Office   Molinare
Pacific       Lea
Mountain      Quill

```

結果會列出每一位經理的姓名及其部門。

## 使用次查詢

當您撰寫 SQL SELECT 陳述式時，可以在 WHERE 子句中放入其它的 SELECT 陳述式。每一個其它的 SELECT 都會啟動一個次查詢。

然後，次查詢還可以包含另一個個別的次查詢，其結果會被代入原始的次查詢 WHERE 子句內。此外，WHERE 子句可將次查詢併入一個以上的搜尋條件中。次查詢可以參照主查詢中所使用的表格和直欄以外的表格和直欄。

下面的陳述式會從 STAFF 表格選取員工 ID 為 280 的員工的 ORG 表格的分處及位置：

```
SELECT DIVISION, LOCATION
FROM ORG
WHERE DEPTNUMB = (SELECT DEPT
FROM STAFF
                  WHERE ID = 280)
```

處理陳述式時，DB2 會先決定次查詢的結果。此範例次查詢的結果是 66，因為 ID 280 的員工屬於部門 66。最終的結果是來自 ORG 表格中 DEPTNUMB 直欄值為 66 的橫列。最終結果如下：

```
DIVISION    LOCATION
-----
Western     San Francisco
```

當您使用次查詢時，資料庫管理會演算之，並將結果值直接代入 WHERE 子句中。

在第38頁的『相關次查詢』中會進一步地討論次查詢。

---

## 使用函數

本節將簡介本書範例中所使用的一些函數。資料庫函數是一組輸入資料值和一個結果值之間的關係。

函數可分為內建及使用者定義的兩種。DB2 Universal Database 隨附許多內建及預先安裝的使用者定義的函數。

您可以在 SYSIBM 綱目中找到內建函數；且預先安裝的使用者定義函數是在 SYSFUN 綱目。SYSIBM 和 SYSFUN 是保留的綱目。

內建和預先安裝的使用者定義函數不可能滿足所有的使用者需求。因此，應用程式開發人員可能必須特別針對其應用程式，自行建立一套適用的函數。使用者定義的函數可以滿足此項要求，它擴充了 DB2 Universal Database 的範圍，納入諸如自訂的商業或科學等函數。這些在第66頁的『使用者定義的函數』中有進一步的討論。

### 直欄函數

直欄函數可運算於一個直欄中的一組值，以衍生單一結果值。以下只是幾個直欄函數的例子。若需完整的列示，請參照 *SQL Reference*。

<b>AVG</b>	傳回一組值的加總除以該組值的數目
<b>COUNT</b>	傳回一組橫列或值中的橫列或值的數目
<b>MAX</b>	傳回一組值中最大的一個值

**MIN** 傳回一組值中最小的一個值

下面的陳述式會選取 STAFF 表格中最高的薪資：

```
SELECT MAX(SALARY)
FROM STAFF
```

這個陳述式從 STAFF 範例表格中傳回 22959.20 這個值。

下一個例子會選取收入高於平均數、服務年資小於平均數的員工的姓名和薪資。

```
SELECT NAME, SALARY
FROM STAFF
WHERE SALARY > (SELECT AVG(SALARY) FROM STAFF)
AND YEARS < (SELECT AVG(YEARS) FROM STAFF)
```

此陳述式會產生以下結果：

NAME	SALARY
Marengi	17506.75
Daniels	19260.25
Gonzales	16858.20

在上述的範例中，WHERE 子句中的直欄函數是在一次查詢中陳述，而非直接施行(例如：WHERE SALARY > AVG(SALARY))。無法在 WHERE 子句中陳述直欄函數。因為運算的次序。您可想成 WHERE 子句在 SELECT 子句之前演算。結果，當 WHERE 子句被演算求值時，直欄函數無法存取該組值。這一組值稍後會由 SELECT 子句選取。

您可以在函數被引用之前，使用 DISTINCT 元素作為直欄函數的引數的一部分，以消除重複的值。如此，COUNT (DISTINCT WORKDEPT) 即會算出不同部門的數目。

## 純量函數

純量函數會在單一值中執行作業，以傳回另一個單一值。以下只是幾個 DB2 Universal Database 提供的純量函數的例子。

**ABS** 傳回數的絕對值。

**HEX** 傳回值的十六進位表示。

**LENGTH** 傳回引數中的位元組數 (若為圖形字串，則傳回二位元組字元數。)

**YEAR** 取出日期時間值中「年」的部分

如需純量函數的詳細列示和說明，請參閱 *SQL Reference*。

以下的陳述式會從 ORG 表格傳回部門名稱和每個名稱的長度：

```
SELECT DEPTNAME, LENGTH(DEPTNAME)
FROM ORG
```

此陳述式會產生以下結果：

```
DEPTNAME      2
-----
Head Office   11
New England   11
Mid Atlantic   12
South Atlantic 14
Great Lakes   11
Plains         6
Pacific       7
Mountain      8
```

請注意：由於沒有使用 AS 子句替 LENGTH(DEPTNAME) 取一個有意義的名稱，因此第二個直欄會出現系統產生的數字。

## 表格函數

表格函數會傳回表格的直欄，類似由簡式 CREATE TABLE 陳述式所建立的表格。

表格函數僅能用在 SQL 陳述式的 FROM 子句中。

DB2 Universal Database 目前唯一支援的表格函數是 SQLCACHE\_SNAPSHOT。

### SQLCACHE\_SNAPSHOT

以表格的形式傳回 DB2 動態 SQL 陳述式快取的 snapshot 結果。

---

## 分組

DB2 Universal Database 有能力根據表格中的某些直欄來分析資料。

您可以依據 GROUP BY 子句中定義的分組結構，組織橫列。以最簡單的表格格式而言，一個群組就是一組橫列，每一個在 "GROUP BY" 直欄中都有相同的值。SELECT 子句中的直欄名稱必須是一個群組直欄或直欄函數。直欄函數會針對由 GROUP BY 子句定義的每一個群組傳回一個值。每一個群組都是由結果集中的單一橫列表示。下面的例子所產生的結果會列出每個部門編號中最高薪資：

```
SELECT DEPT, MAX(SALARY) AS MAXIMUM
FROM STAFF
GROUP BY DEPT
```

此陳述式會產生以下結果：

```
DEPT  MAXIMUM
-----
10    22959.20
15    20659.80
```

```

20 18357.50
38 18006.00
42 18352.80
51 21150.00
66 21000.00
84 19818.00

```

請注意每個部門（由 GROUP BY 子句所定義的群組）都會算出其 MAX(SALARY)，而非計算整個公司的最高薪資。

## WHERE 子句與 GROUP BY 子句的搭配使用

群組查詢可以加上標準 WHERE 子句，以便在群組形成及直欄函數計算之前，先消除資格不合的橫列。您必須在 GROUP BY 子句之前指定 WHERE 子句。例如：

```

SELECT WORKDEPT, EDLEVEL, MAX(SALARY) AS MAXIMUM
FROM EMPLOYEE
WHERE HIREDATE > '1979-01-01'
GROUP BY WORKDEPT, EDLEVEL
ORDER BY WORKDEPT, EDLEVEL

```

結果如下：

WORKDEPT	EDLEVEL	MAXIMUM
D11	17	18270.00
D21	15	27380.00
D21	16	36170.00
D21	17	28760.00
E11	12	15340.00
E21	14	26150.00

請注意，在 SELECT 陳述式中所指定的每個直欄名稱，也會在 GROUP BY 子句中提到。若沒有在兩處同時提到直欄名稱，會傳回錯誤訊息。GROUP BY 子句會為 WORKDEPT 和 EDLEVEL 的每個唯一組合傳回一個橫列。

## 在 GROUP BY 子句後面使用 HAVING 子句

您可以對群組引用定義條件，讓 DB2 只針對符合條件的群組傳回結果。若要這麼做，請在 GROUP BY 子句後面加上 HAVING 子句。HAVING 子句可以含有一個以上的述語，中間以 AND 或 OR 連接。每一個述語會將群組的性質（如 AVG(SALARY)）和下列項目比較：

- 群組的另一種性質

例如：

```
HAVING AVG(SALARY) > 2 * MIN(SALARY)
```

- 一個常數

例如：

```
HAVING AVG(SALARY) > 20000
```

例如，下面的查詢可找出門（員工人數在 4 人以上）最高和最低薪資：

```
SELECT WORKDEPT, MAX(SALARY) AS MAXIMUM, MIN(SALARY) AS MINIMUM
FROM EMPLOYEE
GROUP BY WORKDEPT
HAVING COUNT(*) > 4
ORDER BY WORKDEPT
```

此陳述式會產生以下結果：

WORKDEPT	MAXIMUM	MINIMUM
D11	32250.00	18270.00
D21	36170.00	17250.00
E11	29750.00	15340.00

一個查詢中有 **HAVING** 而無 **GROUP BY** 子句雖然不常發生，但還是有可能。在這個情況下，DB2 會將整個表格視為一個群組。由於整個表格被視為單一群組，因此您最多可以得到一個結果橫列。如果對整個表格來說，**HAVING** 條件為真，那麼選取的結果（必須完全由直欄函數組成）就會被傳回；否則沒有任何橫列會被傳回。





---

## 第5章 表示式與次查詢

DB2 在表示查詢方面提供很大的彈性。本章說明一些重要的方法，可以用來表示複雜的查詢。

本章詳細介紹以下各項：

- 純量全選
- 轉換資料類型
- CASE 表示式
- 表格表示式
- 相關名稱

---

### 純量全選

全選是一種查詢表格，可以用在 SQL 陳述式中。純量全選是一種全選，會傳回一含有一個值的一個橫列。純量全選對從資料庫中擷取資料值，以用於表示式相當有用。

- 下面範例會列出其薪資高於所有員工平均薪資的員工姓名。查詢中的純量全選是括弧中的 select 陳述式。

```
SELECT LASTNAME, FIRSTNAME
       FROM EMPLOYEE
       WHERE SALARY > (SELECT AVG(SALARY)
                       FROM EMPLOYEE)
```

- 此範例會找出 STAFF 表格中員工的平均薪資，並在 EMPLOYEE 表格中找出員工的平均薪資。

```
SELECT AVG(SALARY) AS "Average_Employee",
       (SELECT AVG(SALARY) AS "Average_Staff" FROM STAFF)
       FROM EMPLOYEE
```

---

### 轉換資料類型

有時您必須轉換某些值的資料類型，如將數值改為字串。要轉換值的資料類型，請使用 CAST 指定。

轉換指定的另一種可能的用法是將特別長的字串截斷。在 EMP\_RESUME 表格中，直欄 RESUME 是 CLOB(5K)。您可能只想顯示應徵者個人資訊的前 370 個字元。若要以 ASCII 格式顯示 EMP\_RESUME 表格中的履歷表的前 370 個字元，請發出以下查詢：

```
SELECT EMPNO, CAST(RESUME AS VARCHAR(370))
FROM EMP_RESUME
WHERE RESUME_FORMAT = 'ascii'
```

此時會出現一個訊息，警告您超過 370 個字元的值會被截斷。

您可以將 NULL 值轉換為其它資料類型，以方便查詢的作業。第36頁的『共用表格表示式』即為將轉換作為此用途的例子。

---

## CASE 表示式

您可以在 SQL 陳述式中使用 CASE 表示式，輕易地處理表格的資料表示。如此可提供強大的有條件的表示式能力，它在概念上和某些程式設計語言的 CASE 陳述式類似。

- 欲將 ORG 表格的 DEPTNAME 直欄中的部門編號變更為有意義的名稱，請輸入以下查詢：

```
SELECT DEPTNAME,
CASE DEPTNUMB
WHEN 10 THEN 'Marketing'
WHEN 15 THEN 'Research'
WHEN 20 THEN 'Development'
WHEN 38 THEN 'Accounting'
ELSE 'Sales'
END AS FUNCTION
FROM ORG
```

結果如下：

DEPTNAME	FUNCTION
Head Office	Marketing
New England	Research
Mid Atlantic	Development
South Atlantic	Accounting
Great Lakes	Sales
Plains	Sales
Pacific	Sales
Mountain	Sales

- 您可以使用 CASE 表示式來避免諸如除數為 0 等例外情況。在下面的例子中，如果員工沒有紅利或佣金收入，陳述式條件會避免使用除法運算，以避開錯誤：

```

SELECT LASTNAME, WORKDEPT FROM EMPLOYEE
WHERE(CASE
      WHEN BONUS+COMM=0 THEN NULL
      ELSE SALARY/(BONUS+COMM)
      END ) > 10

```

- 您可以使用 CASE 表示式，產生某一直欄值的子集總計與該直欄所有值的總計的比例。此比例可以內含於使用 CASE 表示式的單一陳述式；此表示式只需要在資料中有一個單一傳遞。若無 CASE 表示式，相同的計算至少需要兩個傳遞。下面的例子利用 CASE 表示式，算出部門編號 20 之薪資總和佔全公司薪資總和的比例：

```

SELECT CAST(CAST (SUM(CASE
                  WHEN DEPT = 20 THEN SALARY
                  ELSE 0
                  END) AS DECIMAL(7,2))/
           SUM(SALARY) AS DECIMAL (3,2))
FROM STAFF

```

結果是 0.11。請注意，CAST 函數可確保結果的精準度被保留。

- 您可以使用 CASE 表示式來演算簡單函數，不須呼叫函數本身，後者需要額外的間接負擔。例如：

```

CASE
WHEN X<0 THEN -1
WHEN X=0 THEN 0
WHEN X>0 THEN 1
END

```

這個表示式得到的結果，和 SYSFUN 綱目中 SIGN 這個使用者定義的函數相同。

---

## 表格表示式

如果您在單一查詢中只需要某個概略表的定義，則可使用表格表示式。

表格表示式是暫時性的，只有在 SQL 陳述式的生存期間內有效；這些表示式不像概略表一樣可以被共用，但較概略表的彈性大。

本節將說明如何在查詢中使用共用表格表示式和巢狀表格表示式。

### 巢狀表格表示式

巢狀表格表示式是一個暫時性的概略表，其中的定義在主查詢的 FROM 子句中是巢狀的 (直接定義的)。

以下查詢使用一個巢狀表格表示式，找出教育水準在 16 年以上的人員的平均薪資總計、教育水準和雇用年份：

```
SELECT EDLEVEL, HIREYEAR, DECIMAL(AVG(TOTAL_PAY), 7,2)
FROM (SELECT EDLEVEL, YEAR(HIREDATE) AS HIREYEAR,
        SALARY+BONUS+COMM AS TOTAL_PAY
FROM EMPLOYEE
WHERE EDLEVEL > 16 ) AS PAY_LEVEL
GROUP BY EDLEVEL, HIREYEAR
ORDER BY EDLEVEL, HIREYEAR
```

結果如下：

EDLEVEL	HIREYEAR	3
17	1967	28850.00
17	1973	23547.00
17	1977	24430.00
17	1979	25896.50
18	1965	57970.00
18	1968	32827.00
18	1973	45350.00
18	1976	31294.00
19	1958	51120.00
20	1975	42110.00

這個查詢使用一個巢狀表格表示式，先從 HIREDATE 直欄中取回雇用年份，再將此資料用於 GROUP BY 子句中。如果您有意使用不同的 EDLEVEL 值執行類似的查詢，您可能不想將此建立為一個概略表。

這個例子用到了純量內建函數 DECIMAL。DECIMAL 會傳回以小數表示的數字或字串。有關函數的詳細說明，請參閱 *SQL Reference*。

## 共用表格表示式

共用表格表示式是一種您為了使用整個複雜查詢而建立的表格表示式。請使用 WITH 子句，在查詢一開始替該表示式定義及命名。對一個共用表格表示式的重複參照，會使用相同的結果集。相較之下，如果您使用巢狀表格表示式或概略表，則每次都會產生結果集，而且每次的結果可能也不同。

下面的範例會列出教育水準在 16 年以上、但其薪資低於同年受雇、教育水準相當的人員之平均薪資。此查詢內容詳述於下。

**1**

```
WITH
PAYLEVEL AS
(SELECT EMPNO, EDLEVEL, YEAR(HIREDATE) AS HIREYEAR,
```

```

        SALARY+BONUS+COMM AS TOTAL_PAY
FROM EMPLOYEE
    WHERE EDLEVEL > 16),

```

**2**

```

PAYBYED (EDUC_LEVEL, YEAR_OF_HIRE, AVG_TOTAL_PAY) AS
(SELECT EDLEVEL, HIREYEAR, AVG(TOTAL_PAY)
FROM PAYLEVEL
GROUP BY EDLEVEL, HIREYEAR)

```

**3**

```

SELECT EMPNO, EDLEVEL, YEAR_OF_HIRE, TOTAL_PAY, DECIMAL(AVG_TOTAL_PAY,7,2)
FROM PAYLEVEL, PAYBYED
WHERE EDLEVEL = EDUC_LEVEL
    AND HIREYEAR = YEAR_OF_HIRE
    AND TOTAL_PAY < AVG_TOTAL_PAY

```

**1**

這是一個名為 PAYLEVEL 的共用表格表示式。這個結果表格包含員工編號、員工雇用年份、該名員工的總薪資、及其教育水準。只有教育水準在 16 年以上的員工橫列，才會列於結果表格中。

**2**

這是一個名為 PAYBYED (或 PAY BY EDucation) 的共用表格表示式。它使用前一個共用表格表示式所建立的 PAYLEVEL 表格來決定教育水準、雇用年份、同一年受雇且每一種教育水準內之員工之平均薪資。這個表格所傳回的直欄，和選取列表中所使用的直欄名稱不同 (如 EDUC\_LEVEL)。這會產生一個名為 PAYBYED 的結果集，與我們在「巢狀表格表示式」範例中產生的結果相同。

**3**

最後，我們到達產生需要的結果的實際查詢。先前的兩個表格 (PAYLEVEL, PAYBYED) 會被結合，找出其薪資總計低於同年雇用的人員的平均薪資的個人。請注意，PAYBYED 是根據 PAYLEVEL 而來的。因此事實上在完整陳述式中，PAYLEVEL 被存取兩次。兩次都使用相同的一組橫列來演算查詢。

最終結果如下：

```

EMPNO  EDLEVEL  YEAR_OF_HIRE  TOTAL_PAY      5
-----
000210      17           1979         20132.00  25896.50

```

## 相關名稱

相關名稱是用來區分一個物件的多次使用的 ID。相關名稱可在查詢的 FROM 子句，或在 UPDATE 或 DELETE 陳述式的第一個子句中定義。它可和一個表格、概略表或巢狀表格表示式結合，但只限其被定義的上下文內。

例如，子句 `FROM STAFF S, ORG O` 會將 `S` 和 `O` 分別建立為 `STAFF` 和 `ORG` 的相關名稱。

```
SELECT NAME, DEPTNAME
FROM STAFF S, ORG O
WHERE O.MANAGER = S.ID
```

一旦您定義了相關名稱後，您只能使用相關名稱來定義物件。例如，在上面的範例中，如果我們陳述 `ORG.MANAGER=STAFF.ID`，則陳述式會失效。

您也可以使用相關名稱，作為參照某個資料庫物件的簡稱。例如鍵入 `S` 要比 `STAFF` 來得快。

利用相關名稱，您可以複製物件。當您需要比較表格的登錄與其本身時，這樣做就很方便。在下面的例子中，會拿表格 `EMPLOYEE` 和本身的另一個案例作比較，以找出所有員工的經理。它會顯示職稱不是設計人員的員工姓名、其經理的姓名以及部門編號。

```
SELECT E2.FIRSTNME, E2.LASTNAME, E2.JOB, E1.FIRSTNME AS MGR_FIRSTNAME,
       E1.LASTNAME AS MGR_LASTNAME, E1.WORKDEPT
FROM EMPLOYEE E1, EMPLOYEE E2
WHERE E1.WORKDEPT = E2.WORKDEPT
      AND E1.JOB = 'MANAGER'
      AND E2.JOB <> 'MANAGER'
      AND E2.JOB <> 'DESIGNER'
```

此陳述式會產生以下結果：

FIRSTNME	LASTNAME	JOB	MGR_FIRSTNAME	MGR_LASTNAME	WORKDEPT
DOLORES	QUINTANA	ANALYST	SALLY	KWAN	C01
HEATHER	NICHOLLS	ANALYST	SALLY	KWAN	C01
JAMES	JEFFERSON	CLERK	EVA	PULASKI	D21
SALVATORE	MARINO	CLERK	EVA	PULASKI	D21
DANIEL	SMITH	CLERK	EVA	PULASKI	D21
SYBIL	JOHNSON	CLERK	EVA	PULASKI	D21
MARIA	PEREZ	CLERK	EVA	PULASKI	D21
ETHEL	SCHNEIDER	OPERATOR	EILEEN	HENDERSON	E11
JOHN	PARKER	OPERATOR	EILEEN	HENDERSON	E11
PHILIP	SMITH	OPERATOR	EILEEN	HENDERSON	E11
MAUDE	SETRIGHT	OPERATOR	EILEEN	HENDERSON	E11
RAMLAL	MEHTA	FIELDREP	THEODORE	SPENSER	E21
WING	LEE	FIELDREP	THEODORE	SPENSER	E21
JASON	GOUNOT	FIELDREP	THEODORE	SPENSER	E21

---

## 相關次查詢

被允許參照任何先前提到之表格的次查詢，稱為相關次查詢。我們也說此次查詢對主查詢中的表格有一個 相關參考。

下面範例會使用非相關次查詢，列出部門 'A00' 中薪資高於該部門平均薪資的員工編號及員工姓名：

```
SELECT EMPNO, LASTNAME
FROM EMPLOYEE
WHERE WORKDEPT = 'A00'
AND SALARY > (SELECT AVG(SALARY)
FROM EMPLOYEE
WHERE WORKDEPT = 'A00')
```

此陳述式會產生以下結果：

```
EMPNO  LASTNAME
-----
000010 HAAS
000110 LUCCHESSI
```

如果您想知道每個部門的平均薪資，則必須對每個部門執行此次查詢一次。您可以利用 SQL 的相關功能完成此作業，它可讓您撰寫一個可重複執行的次查詢，對外部層次查詢中識別的表格之每一橫列執行一次。

下面範例使用相關次查詢，列示其薪資高於其部門平均薪資的所有員工：

```
SELECT E1.EMPNO, E1.LASTNAME, E1.WORKDEPT
FROM EMPLOYEE E1
WHERE SALARY > (SELECT AVG(SALARY)
FROM EMPLOYEE E2
WHERE E2.WORKDEPT = E1.WORKDEPT)
ORDER BY E1.WORKDEPT
```

在這個查詢中，會為每一個部門演算一次次查詢。結果如下：

```
EMPNO  LASTNAME          WORKDEPT
-----
000010 HAAS              A00
000110 LUCCHESSI        A00
000030 KWAN           C01
000060 STERN            D11
000150 ADAMSON         D11
000170 YOSHIMURA      D11
000200 BROWN           D11
000220 LUTZ            D11
000070 PULASKI         D21
000240 MARINO          D21
000270 PEREZ           D21
000090 HENDERSON       E11
000280 SCHNEIDER       E11
000100 SPENSER         E21
000330 LEE             E21
000340 GOUNOT          E21
```

要撰寫一個具有相關次查詢的查詢，請使用與具有次查詢的一般外部查詢的相同基本格式。但在外部查詢的 FROM 子句中，請在表格名稱後面加上相關名稱。次查詢中還可以包含由相關名稱所限定的直欄參照。例如，如果 E1 是一個相關名稱，那麼 E1.WORKDEPT 代表外部查詢中的表格的現行橫的 WORKDEPT 值。在概念上，會為外部查詢中的表格的每一橫列重新演算次查詢。

利用相關次查詢，可以讓系統替您工作，減少應用程式中您須撰寫的程式數量。

DB2 允許使用未限定的相關參考。例如，表格 EMPLOYEE 有一個名為 LASTNAME 的直欄，但表格 SALES 有一個名為 SALES\_PERSON 的直欄，且沒有任何名為 LASTNAME 的直欄。

```
SELECT LASTNAME, FIRSTNAME, COMM
FROM EMPLOYEE
WHERE 3 > (SELECT AVG(SALES)
FROM SALES
WHERE LASTNAME = SALES_PERSON)
```

在這個例子中，系統會檢查最內層的 FROM 子句，看是否有 LASTNAME 直欄。若找不到該直欄，它接下來會檢查最內第二層的 FROM 子句 (在此處即為外部 FROM 子句)。雖然不一定都有必要，但還是建議您限定相關參照，以增加查詢的易讀性，並確保獲得需要的結果。

## 相關次查詢之實作

您什麼時候要用到相關次查詢呢？直欄函數的使用可作為一條線索。

假設您想列出教育水準高於部門平均水準的員工。

首先您必須決定選取列示項目。問題中說要『列出員工』。這暗示 EMPLOYEE 表格中的 LASTNAME 應足以能唯一識別員工。問題中也提到教育水準 (EDLEVEL) 和員工所屬部門 (WORKDEPT) 兩個條件。雖然問題中沒有明白要求要顯示直欄，但將其加入選取列示中，將有助於解釋問題的解答。現在已經可以建構部分查詢了：

```
SELECT LASTNAME, WORKDEPT, EDLEVEL
FROM EMPLOYEE
```

接下來需要的是搜尋條件 (WHERE 子句)。問題中提到『教育水準高於員工所屬部門平均水準』。這表示須算出表格中的每位員工所屬部門之平均教育水準。這個陳述式符合相關次查詢的說明。部份將會計算每一列的未知內容 (現行員工所屬部門的平均教育水準)。需要 EMPLOYEE 表格的相關名稱：

```
SELECT LASTNAME, WORKDEPT, EDLEVEL
FROM EMPLOYEE E1
```



所需的次查詢相當簡單。它會算出每個部門的平均教育水準。完整的 SQL 陳述式如下：

```
SELECT LASTNAME, WORKDEPT, EDLEVEL
FROM EMPLOYEE E1
WHERE EDLEVEL > (SELECT AVG(EDLEVEL)
FROM EMPLOYEE E2
WHERE E2.WORKDEPT = E1.WORKDEPT)
```

結果如下：

LASTNAME	WORKDEPT	EDLEVEL
HAAS	A00	18
KWAN	C01	20
PULASKI	D21	16
HENDERSON	E11	16
LUCCHESI	A00	19
PIANKA	D11	17
SCOUTTEN	D11	17
JONES	D11	17
LUTZ	D11	18
MARINO	D21	17
JOHNSON	D21	16
SCHNEIDER	E11	17
MEHTA	E21	16
GOUNOT	E21	16

假設您想列出的是員工的部門名稱而非部門編號。您所需要的資訊 (DEPTNAME) 位於另一個表格 (DEPARTMENT) 中。定義相關變數的外部層次查詢也可以是一個結合查詢 (詳細資訊請參閱第25頁的『從一個以上的表格中選取資料』)。

當您在外部層次查詢中使用結合時，請於 FROM 子句中列出要結合的表格，並將相關名稱置於適當的表格名稱旁。

若想將查詢修改成列出部門名稱而非編號，請在選取列表中以 DEPTNAME 取代 WORKDEPT。現在 FORM 子句也須包括 DEPARTMENT 表格，且 WHERE 子句必須表示適當的結合條件。

修改後的查詢如下：

```
SELECT LASTNAME, DEPTNAME, EDLEVEL
FROM EMPLOYEE E1, DEPARTMENT
WHERE E1.WORKDEPT = DEPARTMENT.DEPTNO
AND EDLEVEL > (SELECT AVG(EDLEVEL)
FROM EMPLOYEE E2
WHERE E2.WORKDEPT = E1.WORKDEPT)
```

此陳述式會產生以下結果：

LASTNAME	DEPTNAME	EDLEVEL
HAAS	SPIFFY COMPUTER SERVICE DIV.	18
LUCCHESI	SPIFFY COMPUTER SERVICE DIV.	19
KWAN	INFORMATION CENTER	20
PIANKA	MANUFACTURING SYSTEMS	17
SCOUTTEN	MANUFACTURING SYSTEMS	17
JONES	MANUFACTURING SYSTEMS	17
LUTZ	MANUFACTURING SYSTEMS	18
PULASKI	ADMINISTRATION SYSTEMS	16
MARINO	ADMINISTRATION SYSTEMS	17
JOHNSON	ADMINISTRATION SYSTEMS	16
HENDERSON	OPERATIONS	16
SCHNEIDER	OPERATIONS	17
MEHTA	SOFTWARE SUPPORT	16
GOUNOT	SOFTWARE SUPPORT	16

上面幾例顯示次查詢中所用的相關名稱，必須在含有相關次查詢的某些查詢之 FROM 子句中定義。然而，這樣的包含關係可能牽涉到許多層的巢狀。

假設某些部門只有少數幾名員工，因此他們的平均教育水準可能不具代表性。爲了讓平均教育水準成爲一個有比較意義的數字，您可能決定要比較的部門員工數必須在五人以上。因此我們現在必須列出教育水準高於其部門平均教育水準的員工，且只考慮員工數在五人以上的部門。

問題暗示了另一個次查詢，因爲針對外部層次查詢中的每個員工，我們都必須算出其部門的員工總數：

```
SELECT COUNT(*)
      FROM EMPLOYEE E3
     WHERE E3.WORKDEPT = E1.WORKDEPT
```

只有在算出的數目大於等於 5 時，才要計算其平均教育程度：

```
SELECT AVG(EDLEVEL)
      FROM EMPLOYEE E2
     WHERE E2.WORKDEPT = E1.WORKDEPT
    AND 5 <= (SELECT COUNT(*)
              FROM EMPLOYEE E3
             WHERE E3.WORKDEPT = E1.WORKDEPT)
```

最後，只有教育程度高於其部門平均教育水準的員工會被列出來：

```
SELECT LASTNAME, DEPTNAME, EDLEVEL
      FROM EMPLOYEE E1, DEPARTMENT
     WHERE E1.WORKDEPT = DEPARTMENT.DEPTNO
    AND EDLEVEL >
      (SELECT AVG(EDLEVEL)
       FROM EMPLOYEE E2
      WHERE E2.WORKDEPT = E1.WORKDEPT
      AND 5 <=
```

```
(SELECT COUNT(*)  
FROM EMPLOYEE E3  
WHERE E3.WORKDEPT = E1.WORKDEPT))
```

此陳述式會產生以下結果：

LASTNAME	DEPTNAME	EDLEVEL
PIANKA	MANUFACTURING SYSTEMS	17
SCOUTTEN	MANUFACTURING SYSTEMS	17
JONES	MANUFACTURING SYSTEMS	17
LUTZ	MANUFACTURING SYSTEMS	18
PULASKI	ADMINISTRATION SYSTEMS	16
MARINO	ADMINISTRATION SYSTEMS	17
JOHNSON	ADMINISTRATION SYSTEMS	16
HENDERSON	OPERATIONS	16
SCHNEIDER	OPERATIONS	17



---

## 第6章 在查詢中使用運算子和述詞

在 DB2 Universal Database 中，您可以將查詢和不同的集合運算子合併，利用量化述詞建構複雜的有條件的陳述式。

本章描述如何：

- 利用 UNION、EXCEPT 和 INTERSECT 等集合運算子合併不同的表格
- 利用量化述詞建構查詢的複雜條件。基本述詞在第20頁的『選取橫列』中有扼要的討論。

---

### 利用集合運算子合併查詢

UNION、EXCEPT 和 INTERSECT 這三個集合運算子可讓您將兩個或以上的外部層次查詢合併成單一查詢。由這些集合運算子所連接的每一個查詢均會分別執行，然後再將其結果合併起來。每一個運算子會產生不同的結果。

#### UNION 運算子

UNION 運算子會將兩個其它結果表格 (如 TABLE1 與 TABLE2) 合併，消除兩個表格中重複的橫列，產生一個新的結果表格。如果 ALL 與 UNION 同時使用 (即 UNION ALL)，那麼重複的橫列便不會被消除。在任一情況下，新產生的表格中的每一列不是來自 TABLE1 就是 TABLE2。

在下面這個 UNION 運算子的例子中，查詢會傳回薪資高於 \$21,000，或職稱為經理且年資低於八年的所有員工的姓名：

**1**

```
SELECT ID, NAME FROM STAFF WHERE SALARY > 21000  
UNION
```

**2**

```
SELECT ID, NAME FROM STAFF WHERE JOB='Mgr' AND YEARS < 8  
ORDER BY ID
```

個別查詢的結果如下：

**1**

```

ID      NAME
-----
    140 Fraye
    160 Molinare
    260 Jones

```

**2**

```

ID      NAME
-----
    10 Sanders
    30 Marenghi
   100 Plotz
   140 Fraye
   160 Molinare
   240 Daniels

```

資料庫管理會合併兩個查詢的結果，消除重複橫列，然後傳回遞升次序的最終結果。

```

ID      NAME
-----
    10 Sanders
    30 Marenghi
   100 Plotz
   140 Fraye
   160 Molinare
   240 Daniels
   260 Jones

```

如果在查詢中使用 `ORDER BY` 子句和其它集合運算子，則須將 `ORDER BY` 子句置於最後一個查詢後面。如此系統才會將排序功能引用於合併的結果集。

如果兩個表格中的直欄名稱不同，那麼合併的結果表格中不會出現對應直欄的名稱。相反地，會以直欄出現的次序為其編號。因此如果結果表格需要排序，請在 `ORDER BY` 子句中指定直欄編號。

## EXCEPT 運算子

`EXCEPT` 運算子會將 `TABLE1` 中有而 `TABLE2` 中沒有的所有橫列找出來，消除重複的橫列，產生一個新的結果表格。當您使用 `ALL` 和 `EXCEPT` 時 (即 `EXCEPT ALL`)，重複的橫列不會被消除。

在下面的 `EXCEPT` 運算子範例中，查詢會傳回薪資高於 \$21,000，但不是經理，且年資在八年以上的所有員工的姓名。

```

SELECT ID, NAME FROM STAFF WHERE SALARY > 21000
EXCEPT
SELECT ID, NAME FROM STAFF WHERE JOB='Mgr' AND YEARS < 8

```

個別查詢的結果列示在 UNION 的區段中。上述陳述式會產生以下結果：

```

ID      NAME
-----
260 Jones

```

## INTERSECT 運算子

INTERSECT 運算子會將 TABLE1 與 TABLE2 中同時具有的橫列找出來，消除重複的橫列，產生一個新的結果表格。當您使用 ALL 和 INTERSECT 時 (即 INTERSECT ALL)，重複的橫列不會被消除。

在下面 INTERSECT 運算子的範例中，查詢會傳回薪資高於 \$21,000，職稱為經理且年資少於八年的所有員工的姓名及 ID。

```

SELECT ID, NAME FROM STAFF WHERE SALARY > 21000
INTERSECT
SELECT ID, NAME FROM STAFF WHERE JOB='Mgr' AND YEARS < 8

```

個別查詢的結果列於 UNION 中的段落中。兩個查詢的結果經過 INTERSECT 運算的最終結果如下：

```

ID      NAME
-----
140 Fraye
160 Molinare

```

使用 UNION、EXCEPT 和 INTERSECT 運算子時，請注意以下事項：

- 在查詢的選取列表中，運算子的所有對應項目均需相容。如需詳細資訊，請參閱 *SQL Reference* 中的資料類型相容表。
- 如果查詢中使用集合運算子和 ORDER BY 子句，那麼 ORDER BY 子句必須置於最後一個查詢之後。如果直欄名稱與每一個運算子的查詢之選取列示中對應項目相同，則直欄名稱僅能用在 ORDER BY 子句中。
- 資料類型和長度相同之直欄間的運算，所產生的直欄其類型和長度也和原先相同。請參閱 *SQL Reference* 中結果資料類型的規則，以找出 UNION、EXCEPT 和 INTERSECT 三個集合運算子的運算結果。

---

## 述詞

述詞可用來建構條件，讓系統只處理符合條件的橫列。基本述詞在第20頁的『選取橫列』中有討論。本節要討論的有 IN、BETWEEN、LIKE、EXISTS 和量化述詞。

### 使用 IN 述詞

IN 述詞可用來比較一個值與其它數個值。例如：

```
SELECT NAME
FROM STAFF
WHERE DEPT IN (20, 15)
```

這個例子和下例相同：

```
SELECT NAME
FROM STAFF
WHERE DEPT = 20 OR DEPT = 15
```

IN 和 NOT IN 運算子的使用時機為次查詢傳回一組值時。例如，下列查詢會列出負責專案 MA2100 與 OP2012 之員工的姓氏：

```
SELECT LASTNAME
FROM EMPLOYEE
WHERE EMPNO IN
  (SELECT RESPEMP
   FROM PROJECT
   WHERE PROJNO = 'MA2100'
   OR PROJNO = 'OP2012')
```

次查詢會被演算一次，得到的列示會被直接代入外部層次查詢中。例如，如果上述的次查詢會選取員工編號 10 和 330，外部層次查詢的演算如同其 WHERE 為：

```
WHERE EMPNO IN (10, 330)
```

次查詢所傳回的值列示中，可以有零個、一個或多個值。

### 使用 BETWEEN 述詞

BETWEEN 述詞會比較單一值與總括範圍的值（命名於 BETWEEN 述詞中）。

下面的例子會找出薪資介於 \$10,000 和 \$20,000 之間的員工姓名：

```
SELECT LASTNAME
FROM EMPLOYEE
WHERE SALARY BETWEEN 10000 AND 20000
```

這個例子和下例相同：



```

SELECT LASTNAME
FROM EMPLOYEE
WHERE SALARY >= 10000 AND SALARY <= 20000

```

下面的例子會找出薪資低於 \$10,000 或高於 \$20,000 之所有員工的姓名：

```

SELECT LASTNAME
FROM EMPLOYEE
WHERE SALARY NOT BETWEEN 10000 AND 20000

```

## 使用 LIKE 述詞

LIKE 述詞可用來尋找具有特定型樣的字串。型樣是透過百分比符號和底線來指定。

- 底線字元 ( \_ ) 代表任何單一字元。
- 百分比符號 ( % ) 代表零或多個字元的字串。
- 其它字元則代表本身。

下面範例會選取其姓名長度為 7 個字母，且以字母 'S' 開頭的員工：

```

SELECT NAME
FROM STAFF
WHERE NAME LIKE 'S _ _ _ _ _ _ _'

```

下面的例子會選取其姓名開頭不是 'S' 的員工：

```

SELECT NAME
FROM STAFF
WHERE NAME NOT LIKE 'S%'

```

## 使用 EXISTS 述詞

您可以使用一個次查詢來測試滿足某條件的橫列是否存在。在這情況下，會利用述詞 EXISTS 或 NOT EXISTS 鏈結次查詢與外部層次查詢。

當您利用 EXISTS 述詞將次查詢鏈結到外部查詢時，次查詢不會傳回一個值。若次查詢的結果集含有一或多個橫列，則 EXISTS 述詞為真，若不含任何橫列，則為偽。

EXISTS 述詞常與相關次查詢配合使用。下列範圍會列出目前在 PROJECT 表格中沒有登錄的部門：

```

SELECT DEPTNO, DEPTNAME
FROM DEPARTMENT X
WHERE NOT EXISTS
    (SELECT *
     FROM PROJECT
     WHERE DEPTNO = X.DEPTNO)
ORDER BY DEPTNO

```

您可以在外部層次查詢的 WHERE 子句中使用 AND 及 OR，將 EXISTS 與 NOT EXISTS 述詞和其它述詞連接。

## 量化述詞

量化述詞會比較一個值與一組值。如果一個全選傳回一個以上的值，您必須修改述詞中的比較運算子，加上 ALL、ANY 或 SOME 等字尾。這些字尾決定了如何看待外部層次述詞中返回的一組值。我們以 > 比較運算子為例 (以下的備註也適用於其它運算子)：

### 表示式 > ALL (全選)

如果表示式大於全選所傳回的每個個別值，則述詞為真。如果全選未傳回任何值，則述詞為真。若至少一值的指定的關係為偽，則結果為偽。請注意 <>ALL 量化述詞和 NOT IN 述詞相等。

以下的例子使用一個次查詢和 > ALL 比較，以找出收入高於所有經理的員工姓名及職稱：

```
SELECT LASTNAME, JOB
FROM EMPLOYEE
WHERE SALARY > ALL
(SELECT SALARY
FROM EMPLOYEE
WHERE JOB='MANAGER')
```

### 表示式 > ANY (全選)

如果表示式大於全選所傳回的至少一值，則述詞為真。如果全選未傳回任何值，則述詞為偽。請注意，=ANY 量化運算子和 IN 相等。

### 表示式 > SOME (全選)

SOME 和 ANY 是同義字。

有關述詞和運算子的詳細說明，請參閱 *SQL Reference*。

---

## 第7章 進階 SQL

本章涵蓋 DB2 Universal Database 的許多特性，可讓您更有效率地設計出符合您特殊需求的查詢。此章的主題是根據您對該軟體功能有相當程度的了解而定的。

本章內容涵括了：

- 利用限制及觸發施行業務規則
- 結合
- ROLLUP 和 CUBE 查詢及遞迴查詢
- OLAP 函數

---

### 利用限制及觸發施行業務規則

在商業界，有必要確定某些規則恆被施行。例如，執行某專案的員工必須列於薪資表中。或者，我們或許會要某些事件自動發生。例如，如果銷售員售出一批貨，其佣金就會增加。

DB2 Universal Database 提供了一套好用的方法：

- **唯一限制** 可防止表格中一或多個直欄中有重複的值。
- **參照整合性限制**可確保指定表格中資料能夠一致。
- **表格核對限制**是限制直欄中所能擁有的值的規則。如果指定給直欄的值不符合該直欄的核對限制，則插入及更新會失效。
- **觸發函式**會定義一組動作，可在指定的表格中進行刪除、插入或更新作業時被執行 (或被觸發)。觸發函式可用來寫入其它表格、修改輸入值及發出警示訊息。

第一節提供鍵的一個概念性概觀。接下來我們會透過範例和圖解，介紹參照整合性、限制以及觸發函式。

### 鍵

鍵是一組直欄，您可以用它來識別或存取某一系列或某幾列。

由一個以上的直欄所組成的鍵稱為複合鍵。在具有組合鍵的表格中，組合鍵中的直欄次序不一定要對應到它們在表格中的次序。

### 唯一鍵

唯一鍵已定義為其中沒有任兩個相同值的直欄 (或一組直欄)。唯一鍵的直欄不能包含 NULL 值。此限制是由資料庫管理程式於執行 INSERT 與 UPDATE 陳述式時施行。一個表格可以有多个唯一鍵。唯一鍵是選用的，可以在 CREATE TABLE 或 ALTER TABLE 陳述式中定義。

### 主要鍵

主要鍵是作為表格定義的一部分的唯一鍵。一個表格不能有一個以上的主要鍵，且主要鍵的直欄中也不能含有 NULL 值。主要鍵是選用的，可以在 CREATE TABLE 或 ALTER TABLE 陳述式中定義。

### 外來鍵

外來鍵是在參照限制的定義中指定。一個表格可以有零個或多個外來鍵。如果值的任何元件是 NULL，則複合外來鍵的值亦為 NULL。外來鍵是選用的，可以在 CREATE TABLE 或 ALTER TABLE 陳述式中定義。

## 唯一限制

唯一限制可確保鍵的值在一個表格中是唯一的。唯一限制是選用的，您可以在 CREATE TABLE 或 ALTER TABLE 陳述句中指定 PRIMARY KEY 或 UNIQUE 子句來定義它。例如，您可以在表格的員工編號直欄中定義一個唯一限制，以確保每個員工的編號不和其他人重複。

## 參照完整性限制

藉由定義唯一限制和外來鍵，您可以定義表格間的關係，以實行某些業務規則。唯一鍵和外來鍵限制的組合，一般稱為參照完整性限制。外來鍵所參照的唯一限制，稱為母鍵。外來鍵會參照特定的母鍵或與其相關。例如，某個規則可能提到每一位員工 (EMPLOYEE 表格) 必須屬於某個現存部門 (DEPARTMENT 表格)。因此，我們將 EMPLOYEE 表格中的部門編號定義為外來鍵，對 DEPARTMENT 表格中的部門編號定義為主要鍵。以下是參照完整性限制的圖解。

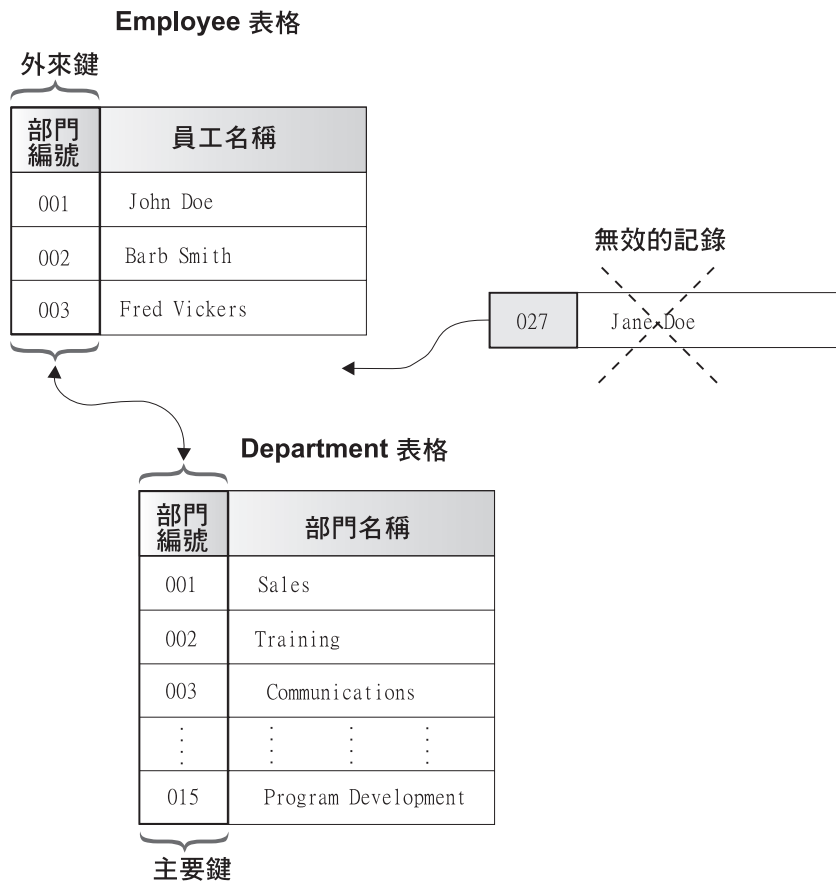


圖 4. 外來及主要限制定義了資料的關係並保護資料

## 表格核對限制

表格核對限制指定了一些條件，會為表格的每一橫列演算這些條件。您也可以在各個欄中指定核對限制。您可以利用 `CREATE` 或 `ALTER TABLE` 陳述式來新增核對限制。

下列陳述式建立了具有下列限制的表格：

- 部門編號的值必須在 10 到 100 的範圍內
- 員工的職稱只能是以下幾種：“Sales”，“Mgr” 或 “Clerk”
- 在 1986 年之前雇用的員工，其薪資必須高於 \$40,500。

```
CREATE TABLE EMP
(ID SMALLINT NOT NULL,
```

```

NAME          VARCHAR(9),
DEPT          SMALLINT CHECK (DEPT BETWEEN 10 AND 100),
JOB           CHAR(5) CHECK (JOB IN ('Sales', 'Mgr', 'Clerk')),
HIREDATE      DATE,
SALARY        DECIMAL(7,2),
COMM          DECIMAL(7,2),
PRIMARY KEY (ID),
CONSTRAINT YEARSAL CHECK
              (YEAR(HIREDATE) >= 1986 OR SALARY > 40500) )

```

如果條件的演算為偽，則表示違反一限制。例如，如果插入的列的 DEPT 為 NULL，則插入作業可以繼續進行，即使限制中定義 DEPT 的值必須在 10 和 100 之間。

下面的陳述式會在 EMPLOYEE 表格中新增名為 COMP 的限制，限制員工的總賠償金必須超過 \$15,000：

```

ALTER TABLE EMP
ADD CONSTRAINT COMP CHECK (SALARY + COMM > 15000)

```

系統會檢查表格中的現有橫列，看是否違反了新的限制。您可以利用如下的 SET CONSTRAINTS 陳述式來延緩此項核對：

```

SET CONSTRAINTS FOR EMP OFF
ALTER TABLE EMP ADD CONSTRAINT COMP CHECK (SALARY + COMM > 15000)
SET CONSTRAINTS FOR EMP IMMEDIATE CHECKED

```

首先我們使用 SET CONSTRAINTS 陳述式來延緩表格的限制核對。接下來可在表格中新增一或多個限制而不核對限制。然後 SET CONSTRAINTS 陳述式會再次被發出，重新開啓限制核對，以執行先前延遲的限制核對。

## 觸發函式

觸發函式會定義一組動作。觸發函式是由一寫修改指定基本表格中資料的作業所啓動。

部份觸發函式的使用如下：

- 執行輸入資料的驗證
- 自動產生新插入的橫列值
- 從其它的表格中讀取以供交互參照使用
- 寫入其它表格以供審核追蹤使用
- 經由電子郵件訊息，支援發出警示

使用觸發函式可以加快應用程式發展的速度、業務規則整體實施、並簡化應用程式與資料的維護。

DB2 Universal Database 支援數種類型的觸發。觸發可被定義於 DELETE、INSERT 或 UPDATE 作業之前或之後啟動。每個觸發都含有一組稱為觸發動作的 SQL 陳述式，其中可有選用的搜尋條件。

您可以進一步地定義事後觸發，對每一個橫列執行觸發動作，或對陳述式執行一次觸發。事前觸發恆會對每一個橫列執行觸發動作。

可在 INSERT、UPDATE 或 DELETE 陳述式前使用觸發，在執行觸發作業之前檢查某些條件，或在輸入值被儲存於表格之前變更這些值。

事後觸發可用來傳送必要的值或執行作為觸發作業一部分所必要的其它作業，如傳送訊息等。

下列範例說明事前及事後觸發的使用。注意記錄並追蹤股價之變更的應用。資料庫含有兩個表格 CURRENTQUOTE 和 QUOTEHISTORY，定義如下：

```
CREATE TABLE CURRENTQUOTE
(SYMBOL VARCHAR(10),
QUOTE DECIMAL(5,2),
STATUS VARCHAR(9))

CREATE TABLE QUOTEHISTORY
(SYMBOL VARCHAR(10),
QUOTE DECIMAL(5,2),
TIMESTAMP TIMESTAMP)
```

當使用以下陳述式更新 CURRENTQUOTE 的 QUOTE 直欄時：

```
UPDATE CURRENTQUOTE
SET QUOTE = 68.5
WHERE SYMBOL = 'IBM'
```

也必須更新 CURRENTQUOTE 的 STATUS 直欄，以反映股票是否：

- 漲價
- 為年度新高
- 跌價
- 為年度新低
- 持平。

這可利用以下的事前觸發達成：

**1**

```
CREATE TRIGGER STOCK_STATUS
NO CASCADE BEFORE UPDATE OF QUOTE ON CURRENTQUOTE
REFERENCING NEW AS NEWQUOTE OLD AS OLDQUOTE
FOR EACH ROW MODE DB2SQL
```

```

2
SET NEWQUOTE.STATUS =

3
CASE

4
    WHEN NEWQUOTE.QUOTE >=
        (SELECT MAX(QUOTE)
         FROM QUOTEHISTORY
         WHERE SYMBOL = NEWQUOTE.SYMBOL
         AND YEAR(TIMESTAMP) = YEAR(CURRENT DATE) )
    THEN 'High'

5
    WHEN NEWQUOTE.QUOTE <=
        (SELECT MIN(QUOTE)
         FROM QUOTEHISTORY
         WHERE SYMBOL = NEWQUOTE.SYMBOL
         AND YEAR(TIMESTAMP) = YEAR(CURRENT DATE) )
    THEN 'Low'

6
    WHEN NEWQUOTE.QUOTE > OLDQUOTE.QUOTE
    THEN 'Rising'
    WHEN NEWQUOTE.QUOTE < OLDQUOTE.QUOTE
    THEN 'Dropping'
    WHEN NEWQUOTE.QUOTE = OLDQUOTE.QUOTE
    THEN 'Steady'
END

```

- 1** 這段程式定義 `STOCK_STATUS` 作為觸發函式，並應在更新 `CURRENTQUOTE` 表格的 `QUOTE` 直欄前啟動。第二行指定觸發動作要在 `CURRENTQUOTE` 表格的實際更新造成的任何變更被引用於資料庫之前引用。 `NO CASCADE` 子句表示觸發動作將不會啟動任何其它的觸發函式。第三行指定須作為新值 (`NEWQUOTE`) 與舊值 (`OLDQUOTE`) 之直欄名稱限定元的名稱。以這些相關名稱 (`NEWQUOTE` 和 `OLDQUOTE`) 限定的直欄名稱，稱為轉移變數。第四行指出必須對每一橫列執行觸發動作。
- 2** 這標示了本觸發的觸發動作的第一個 (且是唯一的一個) `SQL` 陳述式的開頭。 `SET` 這個轉移變數陳述式是用在觸發中，以指定一值給啟動觸發的陳述式更新的表格中的橫列中的直欄。這個陳述式會指定一個值給 `CURRENTQUOTE` 表格的 `STATUS` 直欄。
- 3** 用在指定右側的表示式是一個 `CASE` 表示式。 `CASE` 表示式可延伸到 `END` 關鍵字。



- 4 第一個情況檢查新的價格 (NEWQUOTE.QUOTE) 是否高於本年度股票符號的最大值。次查詢使用後面的事後觸發所更新的 QUOTEHISTORY 表格。
- 5 第二個情況檢查新的價格 (NEWQUOTE.QUOTE) 是否小於本年度股票符號的最小值。次查詢使用後面的事後觸發所更新的 QUOTEHISTORY 表格。
- 6 最後三個情況會拿新價格 (NEWQUOTE.QUOTE) 與表格中的價格 (OLDQUOTE.QUOTE) 比較，以判斷它是大於、小於或相等。SET 轉移變數陳述式在此處結束。

除了更新 CURRENTQUOTE 表格中的登錄之外，還需建立一個審核記錄，方法是將新價格複製到 QUOTEHISTORY 表格中，並加上時間戳記。這可利用下面的事後觸發達成：

1

```
CREATE TRIGGER RECORD_HISTORY
AFTER UPDATE OF QUOTE ON CURRENTQUOTE
REFERENCING NEW AS NEWQUOTE
FOR EACH ROW MODE DB2SQL
BEGIN ATOMIC
```

2

```
INSERT INTO QUOTEHISTORY
VALUES (NEWQUOTE.SYMBOL, NEWQUOTE.QUOTE, CURRENT TIMESTAMP);
END
```

1

這段程式定義了名為 RECORD\_HISTORY 的觸發，作為 CURRENTQUOTE 表格的 QUOTE 直欄更新後要被啟動的觸發。第三行指定了須用作新值 (NEWQUOTE) 直欄名稱限定元的名稱。第四行指出必須對每一橫列執行觸發動作。

2

這個觸發的觸發動作包含了一個單一 SQL 陳述式，它會使用被更新的橫列中的資料，並加上現行時間戳記，在 QUOTEHISTORY 表格中插入橫列。CURRENT\_TIMESTAMP 是含有時間戳記的特別暫存區。第68頁的『特別暫存區』中有其列示及說明。

---

## 結合

合併兩個或以上的表格中的資料的處理，稱為結合表格。資料庫管理程式會組合將來自特定表格中的橫列。針對每個組合，它會測試結合條件。結合條件是具有一些限制的搜尋條件。如需限制的列示，請參閱 *SQL Reference*。

請注意，結合條件中牽涉到的直欄資料類型不須要完全相同，但需要相容。結合條件的演算方式和其它搜尋條件相同，且比較的相同規則也適用。

如果沒有指定結合條件，則會傳回列於 FROM 子句中的橫列的所有組合，即使這些橫列間完全沒有關係。結果被稱為兩個表格的 向量積。

本節的例子是以下面兩個表格為根據。它們是範例資料庫表格的簡化，但不存在於範例資料庫中。這兩個表格用來概述一般結合的重點。SAMP\_STAFF 列出了非約聘人員的姓名及其工作說明，而 SAMP\_PROJECT 則列出約聘且為全職員工的姓名及其參與的專案。

表格如下：

NAME	PROJ
Haas	AD3100
Thompson	PL2100
Walker	MA2112
Lutz	MA2111

圖 5. SAMP\_PROJECT 表格

NAME	JOB
Haas	PRES
Thompson	MANAGER
Lucchessi	SALESREP
Nicholls	ANALYST

圖 6. SAMP\_STAFF 表格

下面的範例會產生兩個表格的對照產品。由於沒有指定結合條件，所以每一種橫列的組合如下：

```
SELECT SAMP_PROJECT.NAME,  
       SAMP_PROJECT.PROJ, SAMP_STAFF.NAME, SAMP_STAFF.JOB  
FROM SAMP_PROJECT, SAMP_STAFF
```

此陳述式會產生以下結果：

NAME	PROJ		NAME	JOB
Haas	AD3100		Haas	PRES
Thompson	PL2100	Haas	PRES	
Walker	MA2112	Haas	PRES	
Lutz	MA2111	Haas	PRES	
Haas	AD3100	Thompson	MANAGER	
Thompson	PL2100		Thompson	MANAGER
Walker	MA2112	Thompson	MANAGER	
Lutz	MA2111	Thompson	MANAGER	
Haas	AD3100	Lucchessi	SALESREP	
Thompson	PL2100	Lucchessi	SALESREP	
Walker	MA2112	Lucchessi	SALESREP	
Lutz	MA2111	Lucchessi	SALESREP	
Haas	AD3100	Nicholls	ANALYST	
Thompson	PL2100	Nicholls	ANALYST	
Walker	MA2112	Nicholls	ANALYST	
Lutz	MA2111	Nicholls	ANALYST	

結合的兩種主要類型是內部結合和外部結合。到目前為止，我們所有的例子都使用內部結合。內部結合會只保留向量積中滿足結合條件的橫列。如果某個橫列只存在於一個表格中，不存在另一個表格中，則其資訊不會列於結果表格內。

下面的例子會產生兩個表格的內部結合。內部結合會列出參與某一專案的全職員工：

```
SELECT SAMP_PROJECT.NAME,
       SAMP_PROJECT.PROJ, SAMP_STAFF.NAME, SAMP_STAFF.JOB
FROM SAMP_PROJECT, SAMP_STAFF
WHERE SAMP_STAFF.NAME = SAMP_PROJECT.NAME
```

另外，您也可以使用下列方式指定內部結合：

```
SELECT SAMP_PROJECT.NAME,
       SAMP_PROJECT.PROJ, SAMP_STAFF.NAME, SAMP_STAFF.JOB
FROM SAMP_PROJECT INNER JOIN SAMP_STAFF
ON SAMP_STAFF.NAME = SAMP_PROJECT.NAME
```

結果如下：

NAME	PROJ	NAME	JOB
Haas	AD3100	Haas	PRES
Thompson	PL2100	Thompson	MANAGER

請注意：內部結合的結果是由在右邊及左邊的表格中，其 NAME 直欄值相符的橫列所組成 - 'Haas' 及 'Thompson' 兩者均內含於 SAMP\_STAFF 表格 (列出所有全職員工) 及 SAMP\_PROJECT 表格 (列出參與專案的全職及約聘員工) 中。

外部結合是內部結合加上位於左側表格、右側表格或兩者中，但在內部結合中遺漏的橫列。當您對兩個表格執行外部結合時，您可以任意指定一個表格作為左側表格，另一個作為右側表格。外部結合有三種類型：

1. **左側外部結合** 包括了內部結合以及位於左側表格中但不位於內部結合的橫列。
2. **右側外部結合** 包括了內部結合以及位於右側表格中但不位於內部結合的橫列。
3. **完整外部結合** 包括了內部結合以及位於左、右側表格中但不位於內部結合的橫列。

請使用 `SELECT` 陳述式指定要顯示的直欄。在 `FROM` 子句中，請列出第一個表格的名稱，然後是 `LEFT OUTER JOIN`、`RIGHT OUTER JOIN` 或 `FULL OUTER JOIN` 關鍵字。接下來您必須指定第二個表格，後面加上 `ON` 關鍵字。在 `ON` 關鍵字之後，請指定結合條件，以表示要結合的表格之間的關係。

在下面的例子中，`SAMP_STAFF` 被指定為右側表格，`SAMP_PROJECT` 為左側表格。利用 `LEFT OUTER JOIN`，我們列出了所有員工（全職及約聘）的姓名及專案（列於 `SAMP_PROJECT` 表格中），若為全職員工，還列出其職稱（列於 `SAMP_STAFF` 表格中）：

```
SELECT SAMP_PROJECT.NAME, SAMP_PROJECT.PROJ,
       SAMP_STAFF.NAME, SAMP_STAFF.JOB
FROM SAMP_PROJECT LEFT OUTER JOIN SAMP_STAFF
ON SAMP_STAFF.NAME = SAMP_PROJECT.NAME
```

此陳述式會產生以下結果：

NAME	PROJ	NAME	JOB
Haas	AD3100	Haas	PRES
Lutz	MA2111	-	-
Thompson	PL2100	Thompson	MANAGER
Walker	MA2112	-	-

具有所有直欄中的值的橫列，即為內部結合的結果。這些橫列都滿足了結合條件：'Haas' 與 'Thompson' 同時列於 `SAMP_PROJECT`（左側表格）以及 `SAMP_STAFF`（右側表格）中。至於未滿足結合條件的橫列，右側表格的直欄中會出現 `NULL` 值：'Lutz' 和 'Walker' 是列於 `SAMP_PROJECT` 表格但未列於 `SAMP_STAFF` 表格內的約聘人員。請注意，左側表格中的所有橫列都含括在結果集內。

在下面的例子中，`SAMP_STAFF` 被指定為右側表格，`SAMP_PROJECT` 為左側表格。利用 `RIGHT OUTER JOIN`，我們列出了所有全職員工的姓名及職稱（列於 `SAMP_STAFF` 表格中），若其參與某個專案，還列出其專案編號（列於 `SAMP_PROJECT` 表格中）：

```

SELECT SAMP_PROJECT.NAME,
       SAMP_PROJECT.PROJ, SAMP_STAFF.NAME, SAMP_STAFF.JOB
FROM SAMP_PROJECT RIGHT OUTER JOIN SAMP_STAFF
ON SAMP_STAFF.NAME = SAMP_PROJECT.NAME

```

結果如下：

NAME	PROJ	NAME	JOB
Haas	AD3100	Haas	PRES
-	-	Lucchessi	SALESREP
-	-	Nicholls	ANALYST
Thompson	PL2100	Thompson	MANAGER

如同左側外部結合，具有所有直欄中的值的橫列，即為內部結合的結果。這些橫列都滿足了結合條件：'Haas' 與 'Thompson' 同時列於 SAMP\_PROJECT (左側表格) 以及 SAMP\_STAFF (右側表格) 中。至於不符合結合條件的橫列，右側表格的直欄中會出現 NULL 值：'Lucchessi' 和 'Nicholls' 是沒有參與任何專案的全職員工。雖然他們列於 SAMP\_STAFF 中，但並未出現在 SAMP\_PROJECT 表格內。請注意，右側表格中的所有橫列都含括於結果集內。

下一個例子對 SAMP\_PROJECT 和 SAMP\_STAFF 兩個表格執行 FULL OUTER JOIN。它列出了所有全職員工 (包括未參與專案者) 以及約聘人員：

```

SELECT SAMP_PROJECT.NAME, SAMP_PROJECT.PROJ,
       SAMP_STAFF.NAME, SAMP_STAFF.JOB
FROM SAMP_PROJECT FULL OUTER JOIN SAMP_STAFF
ON SAMP_STAFF.NAME = SAMP_PROJECT.NAME

```

結果如下：

NAME	PROJ	NAME	JOB
Haas	AD3100	Haas	PRES
-	-	Lucchessi	SALESREP
-	-	Nicholls	ANALYST
Thompson	PL2100	Thompson	MANAGER
Lutz	MA2111	-	-
Walker	MA2112	-	-

這個結果含括了左側外部結合、右側外部結合以及內部結合。因此所有全職與約聘人員都會被列出。就像左側外部結合與右側外部結合，不符合結合條件的值，在各自的直欄中會出現 NULL 值。SAMP\_STAFF 和 SAMP\_PROJECT 表格中的每個橫列都會列於結果集內。

---

## 複雜查詢

DB2 Universal Database 可讓您利用 ROLLUP 和 CUBE，在單一結果集中分組、合併及檢視多個直欄。這個功能強大的新特性，加強並簡化了 SQL 為主的資料分析。

有許多方法可從資料庫中取回有用的資訊。您可以施行遞迴查詢，從現存資料集中產生結果表格。

### ROLLUP 和 CUBE 查詢

您可以在查詢的 GROUP BY 子句中指定 ROLLUP 和 CUBE 運算。

ROLLUP 群組會產生含有規則分組的橫列和小計橫列的結果集。CUBE 群組會產生含有來自 ROLLUP 的橫列以及交叉製表橫列的結果集。

因此，若使用 ROLLUP，您可取得按人、按月的銷售，以及每月銷售總計與整體總計。若使用 CUBE，則外加每位銷售人員的總銷售量。

如需詳細資訊，請參閱 *SQL Reference*。

### 遞迴查詢

遞迴查詢是重複使用結果資料，以判定進一步結果的查詢。您可以將它想像成行經橫貫一棵樹或一個圖。

實際範例包含預約系統、網路規劃及排程。

遞迴查詢是利用共用表格表示式寫成的，此表示式含括了本身名稱的參照。

如需遞迴查詢的範例，請參閱 *SQL Reference*。

---

## OLAP 函數

OnLine Analytical Processing (OLAP) 函數會經由資料的視窗，執行直欄函數作業。此視窗可以指定橫列的分割、分割區中的橫列次序或聚集群組。聚集群組可讓使用者指定要在計算中加入哪些橫列 (與現行橫列相關的)。這類視窗的使用可以執行累加總計及移動平均值等作業。

除了讓使用者可以指定現存直欄函數 (如 SUM 及 AVG) 的視窗外，OLAP 函數可以執行等級 (RANK 及 DENSE\_RANK) 作業，並提供列號 (ROW\_NUMBER)，指定橫列的特定分割及排列。

下面的範例查詢會依照薪資提供部門員工等級，並顯示部門 (指部門 15 及 38) 中薪資的累加總計：

```
SELECT NAME, DEPT,  
       RANK () OVER (PARTITION BY DEPT ORDER BY SALARY DESC) AS RANK,  
       SUM (SALARY) OVER (PARTITION BY DEPT  
                          ORDER BY SALARY DESC  
                          ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)  
       AS CUMULATIVE_SUM  
FROM STAFF  
WHERE DEPT IN (15,38)  
ORDER BY DEPT, RANK
```

此陳述式會產生以下結果：

NAME	DEPT	RANK	CUMULATIVE_SUM
Hanes	15	1	20659.80
Rothman	15	2	37162.63
Ngan	15	3	49670.83
Kermisch	15	4	61929.33
O'Brien	38	1	18006.00
Marenghi	38	2	35512.75
Quigley	38	3	52321.05
Naughton	38	4	65275.80
Abrahams	38	5	77285.55





---

## 第8章 自行設定及加強資料處理

本章將扼要說明 DB2 Universal Database 中的物件導向擴充。物件導向擴充有許多好優點。使用者定義的類型 (UDT) 會增加可供您應用程式使用的資料類型組，而使用者定義的函數 (UDF) 可以建立應用程式特定的函數。UDF 是作為 UDT 的方法使用，為類型提供一致的表現和囊封技術。

接下來會討論特別暫存區和系統目錄。特別暫存區是由資料庫管理程式所定義的儲存區；它們是用來儲存 SQL 陳述式可以參照的資訊。在連線時會建立特別暫存區，並專供該應用程式的處理使用。系統目錄含有資料庫物件的邏輯和實體結構的相關資訊。

本章內容涵括了：

- 使用者定義的類型
- 使用者定義的函數
- 大型物件 (LOB)
- 特別暫存區
- 目錄概略表簡介

上述主題的詳細討論資訊不在本書的範圍內，請參考 *SQL Reference* 及 *Administration Guide*。

---

### 使用者定義的類型

特殊類型是一個使用者定義的資料類型，它和現存類型 (即其『來源』類型) 共用其內部表示，但仍被視為個別類型，且和大部分運算不相容。例如，您可能想定義一個年齡類型、體重類型、以及身高類型，這幾種類型所使用的語意差異甚大，但同樣使用內建資料類型 `INTERGER` 作為內部表示。

下面的例子介紹 `PAY` 這個特殊類型的建立過程：

```
CREATE DISTINCT TYPE PAY AS DECIMAL(9,2) WITH COMPARISONS
```

雖然 `PAY` 的表示和內建資料類型 `DECIMAL(9,2)` 相同，但它仍被視為一個個別的類型，無法與 `DECIMAL(9,2)` 或其它類型比較。它只能和相同的特殊類型相比。同時，`DECIMAL` 可使用的運算子和函數在此處也不適用。例如，資料類型為 `PAY` 的值無法和 `INTERGER` 資料類型的值相乘。因此，您必須撰寫僅能用於 `PAY` 資料類型的函數。

使用特殊資料類型可以限制偶發錯誤的發生。舉例來說，如果表格 EMPLOYEE 的 SALARY 直欄被定義成 PAY 資料類型，則它無法與 COMM 相加，即使兩者的來源類型相同。

特殊資料類型支援轉換。一個來源類型可被轉換為特殊資料類型，而一個特殊資料類型也可被轉換為來源類型。例如，如果表格 EMPLOYEE 的 SALARY 直欄被定義為 PAY 資料類型，則下面的例子在遇到比較運算子時不會執行失敗。

```
SELECT * FROM EMPLOYEE
WHERE DECIMAL(SALARY) = 41250
```

DECIMAL (SALARY) 會傳回小數資料類型。反過來說，數字資料類型也可以轉換為 PAY 類型。例如，您可以使用 PAY (41250) 轉換 41250 這個數字。

---

## 使用者定義的函數

如第27頁的『使用函數』所述，DB2 Universal Database 提供了內建和使用者定義的函數 (UDF)。然而這組函數不可能滿足所有需求。您經常需要建立自定函數來執行某一作業。使用者定義的函數可讓您建立自定函數。

有四種類型的使用者定義函數：來源 (或模板)、外部純量、外部表格及 OLE DB 外部表格。

本節說明來源及外部純量類型。有關外部表格及 OLE DB 表格類型的其餘資訊，請參閱 *SQL Reference*。

來源使用者定義的函數可讓使用者定義的類型選擇性地參照資料庫已知的另一個內建或使用者定義的函數。您可以使用純量與直欄函數。

在下面的範例中，使用者定義的函數 (稱為 MAX) 已依照內建的 MAX 直欄函數建立了，它會將 DECIMAL 資料類型當作輸入。MAX 這個 UDF 的輸入是一個 PAY 類型，並傳回一個 PAY 類型作為輸出。

```
CREATE FUNCTION MAX(PAY) RETURNS PAY
SOURCE MAX(DECIMAL)
```

外部使用者定義的函數是由使用者利用程式設計語言所寫成的。它可分為 外部純量函數以及外部表格函數，兩者都將在 *SQL Reference* 中討論。

至於另一個範例，假設您已撰寫了一個可計算字串中的字數的函數，您可以利用 CREATE FUNCTION 陳述式，在資料庫中將此函數登記為 WORDCOUNT 這個名稱。這個函數接下來就可以用於 SQL 陳述式中。

下面的陳述式將以 ASCII 格式傳回員工履歷表中的員工編號及字數。WORDCOUNT 是使用者已在資料庫中登記的一個外部純量函數，現在即可用於陳述式中。

```
SELECT EMPNO, WORDCOUNT(RESUME)
FROM EMP_RESUME
WHERE RESUME_FORMAT = 'ascii'
```

如需撰寫使用者定義的函數的詳細資訊，請參閱 *Application Development Guide*。

---

## 大型物件 (LOB)

大型物件這個名詞及其頭字語 *LOB*，代表三種資料類型：BLOB、CLOB 及 DBCLOB。這三種類型可以包含大量資料，適用於音效、圖片和文件等物件。

二進位大型物件 (*BLOB*) 是以位元組為單位的可變長度字串，最長可以高達 2 GB。BLOB 主要是用來存放諸如圖片、聲音和混合媒體等非傳統資料。

字元大型物件 (*CLOB*) 是以位元組為單位的可變長度字串，最長可以高達 2 GB。CLOB 可用來儲存諸如文件等大型單位元組字集資料 (SBCS)。CLOB 被視為一個字串。

二位元組大型物件 (*DBCLOB*) 是一個二位元組字元的可變長度字串，最長可達 2GB (1 073 741 823 二位元組字元)。DBCLOB 可用來儲存諸如文件等大型二位元組字集資料。DBCLOB 被視為一個圖形字串。

## 處理大型物件 (LOB)

由於 LOB 值可能相當大，將這些值從資料庫伺服器轉送到從屬站應用程式可能相當費時。然而，一般會以一次一大塊的方式處理 LOB 值，而非整個作為一個單位處理。在某些情況下，如果應用程式不需要 (或不希望) 整個 LOB 值儲存到應用程式記憶體內，則可利用大型物件定位器變數來參照這個值。

後續的陳述式可使用定位器對資料執行運算，不必擷取整個大型物件。定位器變數可以減少應用程式的儲存體需求，並藉由降低從屬站和伺服器之間的資料流量來提昇系統效能。

另一種機制是檔案參照變數。它們是用來直接將一個大型物件擷取到檔案中，或直接從檔案中更新表格內的大型物件。檔案參照變數可用來減少應用程式的儲存體需求，因為它不需要儲存大型物件資料。如需詳細資訊，請參閱 *Application Development Guide* 及 *SQL Reference*。

---

## 特別暫存區

特別暫存區是資料庫管理程式為連接所定義的儲存區，用來儲存可在 SQL 陳述式中參照的資訊。下面就是一些常用的特別暫存區的例子。如需所有特別暫存區的列示及詳細說明，請參閱 *SQL Reference*。

- **CURRENT DATE**：存放 SQL 陳述式執行時的日期時間時脈的日期。
- **CURRENT FUNCTION PATH**：存放指定函數路徑的值，該路徑乃用來轉換函數與資料類型參照。
- **CURRENT SERVER**：指定現行應用程式伺服器。
- **CURRENT TIME**：存放 SQL 陳述式執行時的日期時間時脈的時間。
- **CURRENT TIMESTAMP**：存放 SQL 陳述式執行時的日期時間時鐘的時間戳記。
- **CURRENT TIMEZONE**：指定世界標準時間與應用程式伺服器本地時間之間的時差。
- **USER**：指定執行時間授權 ID。

您可以利用 **VALUES** 陳述式來顯示特別暫存區的內容。例如：

```
VALUES (CURRENT TIMESTAMP)
```

您也可以使用：

```
SELECT CURRENT TIMESTAMP FROM ORG
```

它會為表格中每個橫列登錄 **TIMESTAMP**。

---

## 目錄概略表簡介

DB2 會為每個資料庫建立並維護一組廣泛的系統目錄表格。這些表格含有關於資料庫物件的邏輯和實體結構 (如表格、概略表、資料包、參照完整性關係、函數、顯著類型和觸發等) 的資訊。它是在建立資料庫時建立的，並在日常作業期間被更新。您無法明確地建立或捨棄這些系統目錄，但是可以查詢或檢視其內容。

如需詳細資訊，請參閱 *SQL Reference*。

### 從系統目錄中選取橫列

目錄概略表和其它的資料庫概略表類似。您可以使用 SQL 陳述式來查閱資料，方式和查閱系統中的其它概略表完全相同。

您可以在 **SYSCAT.TABLES** 目錄中找到很有用的表格資訊。要找出您所建立的現存表格的名稱，請發出類似下列的陳述式：

```

SELECT TABNAME, TYPE, CREATE_TIME
FROM SYSCAT.TABLES
WHERE DEFINER = USER

```

此陳述式會產生以下結果：

TABNAME	TYPE	CREATE_TIME
ORG	T	1999-07-21-13.42.55.128005
STAFF	T	1999-07-21-13.42.55.609001
DEPARTMENT	T	1999-07-21-13.42.56.069001
EMPLOYEE	T	1999-07-21-13.42.56.310001
EMP_ACT	T	1999-07-21-13.42.56.710001
PROJECT	T	1999-07-21-13.42.57.051001
EMP_PHOTO	T	1999-07-21-13.42.57.361001
EMP_RESUME	T	1999-07-21-13.42.59.154001
SALES	T	1999-07-21-13.42.59.855001
CL_SCHED	T	1999-07-21-13.43.00.025002
IN_TRAY	T	1999-07-21-13.43.00.055001

以下列示中含有與本書討論的主旨相關的目錄概略表。有許多其它的型錄概略表，詳細列於 *SQL Reference* 與 *Administration Guide* 手冊中。

說明	目錄概略表
核對限制	SYSCAT.CHECKS
直欄	SYSCAT.COLUMNS
核對限制所參照的直欄	SYSCAT.COLCHECKS
鍵中使用的直欄	SYSCAT.KEYCOLUSE
資料類型	SYSCAT.DATATYPES
函數參數或函數結果	SYSCAT.FUNCPARMS
參照限制	SYSCAT.REFERENCES
綱目	SYSCAT.SCHEMATA
表格限制	SYSCAT.TABCONST
表格	SYSCAT.TABLES
觸發	SYSCAT.TRIGGERS
使用者定義的函數	SYSCAT.FUNCTIONS
概略表	SYSCAT.VIEWS



---

## 附錄A. 範例資料庫表格

本附錄會顯示在範例資料庫 SAMPLE 的範例表格中所包含的資訊，及如何建立與除去它們。

其它的範例資料庫隨附於 DB2 Universal Database，示範 business intelligence 功能，並用於 business intelligence 教學指導。然而，本附錄中僅說明範例資料庫 SAMPLE 的內容。請參照資料倉儲中心管理手冊，取得 business intelligence 範例資料庫的其餘相關資訊。

這些範例表格將出現在這本手冊及書庫中的其它手冊中的範例內。此外，本附錄也列出範例檔案中資料類型為 BLOB 及 CLOB 的資料。

本附錄含有以下幾節：

- 第72頁的『範例資料庫』
- 第72頁的『建立範例資料庫』
- 第72頁的『若要消除範例資料庫』
- 第72頁的『CL\_SCHED 表格』
- 第73頁的『DEPARTMENT 表格』
- 第73頁的『EMPLOYEE 表格』
- 第76頁的『EMP\_ACT 表格』
- 第78頁的『EMP\_PHOTO 表格』
- 第78頁的『EMP\_RESUME 表格』
- 第79頁的『IN\_TRAY 表格』
- 第79頁的『ORG 表格』
- 第79頁的『PROJECT 表格』
- 第81頁的『SALES 表格』
- 第82頁的『STAFF 表格』
- 第83頁的『STAFFG 表格』
- 第84頁的『資料類型為 BLOB 和 CLOB 的範例檔案』
- 第84頁的『Quintana 相片』
- 第85頁的『Quintana 履歷表』
- 第86頁的『Nicholls 相片』
- 第86頁的『Nicholls 履歷表』
- 第87頁的『Adamson 相片』
- 第87頁的『Adamson 履歷表』
- 第88頁的『Walker 相片』
- 第89頁的『Walker 履歷表』。

## 範例資料庫表格

範例表格中的橫線 (-) 代表 NULL 值。

---

## 範例資料庫

本書中的例子均出自一個範例資料庫。欲使用這些範例，您必須建立 SAMPLE 資料庫。要使用 SAMPLE 資料庫，您還得先安裝資料庫管理程式。

### 建立範例資料庫

可執行檔會建立範例資料庫。<sup>2</sup> 欲建立資料庫，您必須具有 SYSADM 權限。

- 使用 **UNIX** 為主的系統時

如果使用作業系統命令提示，請鍵入：

```
sqllib/bin/db2samp1 <path>
```

其中路徑是選用的參數，用來指定範例資料庫的建立路徑。按 Enter 鍵。<sup>3</sup> DB2SAMPL 綱目為 CURRENT SCHEMA 特定登錄值。

- 使用 **OS/2** 或 **Windows** 平台時

如果使用作業系統命令提示，請鍵入：

```
db2samp1 e
```

其中 *e* 是一個選用的參數，用來指定資料庫建立所在的磁碟機。按 Enter 鍵。

<sup>4</sup>

如果您尚未透過使用者設定檔管理登入您的工作站，會出現提示您登入的訊息。

### 若要消除範例資料庫

如果不需存取範例資料庫，您可以使用 DROP DATABASE 命令將其消除：

```
db2 drop database sample
```

### CL\_SCHED 表格

名稱：	CLASS_CODE	DAY	STARTING	ENDING
類型：	char(7)	smallint	time	time

---

2. 與此命令相關的資訊，請參閱 *Command Reference* 中的 DB2SAMPL 命令。

3. 如果未指定路徑參數，會在資料庫管理程式架構檔的 DFTDBPATH 參數所指定的預設路徑中建立範例資料庫。

4. 如果未指定磁碟機參數，會在與 DB2 相同的磁碟機中建立範例資料庫。



## 範例資料庫表格

<b>名稱：</b>	<b>CLASS_CODE</b>	<b>DAY</b>	<b>STARTING</b>	<b>ENDING</b>
說明：	課程碼 (room:teacher)	Day # of 4 day schedule	課程開始時間	課程結束時間

## DEPARTMENT 表格

<b>名稱：</b>	<b>DEPTNO</b>	<b>DEPTNAME</b>	<b>MGRNO</b>	<b>ADMRDEPT</b>	<b>LOCATION</b>
類型：	char(3) not null	varchar(29) not null	char(6)	char(3) not null	char(16)
說明：	部門編號	說明部門一般活動的名稱	部門經理之員工編號 (EMPNO)	此部門之上級部門 (DEPTNO)	遠端位置名稱
值：	A00	SPIFFY 電腦服務部	000010	A00	-
	B01	規劃	000020	A00	-
	C01	資訊中心	000030	A00	-
	D01	開發中心	-	A00	-
	D11	製造系統	000060	D01	-
	D21	管理系統	000070	D01	-
	E01	支援服務	000050	A00	-
	E11	作業	000090	E01	-
	E21	軟體支援	000100	E01	-

## EMPLOYEE 表格

<b>名稱：</b>	<b>EMPNO</b>	<b>FIRSTNAME</b>	<b>MIDINIT</b>	<b>LASTNAME</b>	<b>WORKDEPT</b>	<b>PHONENO</b>	<b>HIREDATE</b>
類型：	char(6) not null	varchar(12) not null	char(1) not null	varchar(15) not null	char(3)	char(4)	date
說明：	員工編號	名	中間名縮寫	姓	員工所屬部門 (DEPTNO)	電話號碼	雇用日期
<b>JOB</b>	<b>EDLEVEL</b>	<b>性別</b>	<b>出生年月日</b>	<b>SALARY</b>	<b>紅利</b>	<b>COMM</b>	
char(8)	smallint not null	char(1)	date	dec(9,2)	dec(9,2)	dec(9,2)	
工作	接受正式教育年數	性別 (M 男性, F 女性)	出生年月日	年薪	年度紅利	年度佣金	

EMPLOYEE 表格中的值請參閱下一頁。

範例資料庫表格

EMPNO	FIRSTNAME	MID INIT	LASTNAME	WORK DEPT	PHONE NO	HIREDATE	JOB	ED LEVEL	SEX	BIRTHDATE	SALARY	BONUS	COMM
char(6) not null	varchar(12) not null	char(1) not null	varchar(15) not null	char(3) not null	char(4) not null	date	char(8)	smallint not null	char(1)	date	dec(9,2)	dec(9,2)	dec(9,2)
000010	CHRISTINE	I	HAAS	A00	3978	1965-01-01	PRES	18	F	1933-08-24	52750	1000	4220
000020	MICHAEL	L	THOMPSON	B01	3476	1973-10-10	MANAGER	18	M	1948-02-02	41250	800	3300
000030	SALLY	A	KWAN	C01	4738	1975-04-05	MANAGER	20	F	1941-05-11	38250	800	3060
000050	JOHN	B	GEYER	E01	6789	1949-08-17	MANAGER	16	M	1925-09-15	40175	800	3214
000060	IRVING	F	STERN	D11	6423	1973-09-14	MANAGER	16	M	1945-07-07	32250	500	2580
000070	EVA	D	PULASKI	D21	7831	1980-09-30	MANAGER	16	F	1953-05-26	36170	700	2893
000090	EILEEN	W	HENDERSON	E11	5498	1970-08-15	MANAGER	16	F	1941-05-15	29750	600	2380
000100	THEODORE	Q	SPENSER	E21	0972	1980-06-19	MANAGER	14	M	1956-12-18	26150	500	2092
000110	VINCENZO	G	LUCCHESI	A00	3490	1958-05-16	SALESREP	19	M	1929-11-05	46500	900	3720
000120	SEAN	M	O'CONNELL	A00	2167	1963-12-05	CLERK	14	M	1942-10-18	29250	600	2340
000130	DOLORES	M	QUINTANA	C01	4578	1971-07-28	ANALYST	16	F	1925-09-15	23800	500	1904
000140	HEATHER	A	NICHOLLS	C01	1793	1976-12-15	ANALYST	18	F	1946-01-19	28420	600	2274
000150	BRUCE		ADAMSON	D11	4510	1972-02-12	DESIGNER	16	M	1947-05-17	25280	500	2022
000160	ELIZABETH	R	PIANKA	D11	3782	1977-10-11	DESIGNER	17	F	1955-04-12	22250	400	1780
000170	MASATOSHI	J	YOSHIMURA	D11	2890	1978-09-15	DESIGNER	16	M	1951-01-05	24680	500	1974
000180	MARILYN	S	SCOUTTEN	D11	1682	1973-07-07	DESIGNER	17	F	1949-02-21	21340	500	1707
000190	JAMES	H	WALKER	D11	2986	1974-07-26	DESIGNER	16	M	1952-06-25	20450	400	1636
000200	DAVID		BROWN	D11	4501	1966-03-03	DESIGNER	16	M	1941-05-29	27740	600	2217
000210	WILLIAM	T	JONES	D11	0942	1979-04-11	DESIGNER	17	M	1953-02-23	18270	400	1462
000220	JENNIFER	K	LUTZ	D11	0672	1968-08-29	DESIGNER	18	F	1948-03-19	29840	600	2387
000230	JAMES	J	JEFFERSON	D21	2094	1966-11-21	CLERK	14	M	1935-05-30	22180	400	1774
000240	SALVATORE	M	MARINO	D21	3780	1979-12-05	CLERK	17	M	1954-03-31	28760	600	2301
000250	DANIEL	S	SMITH	D21	0961	1969-10-30	CLERK	15	M	1939-11-12	19180	400	1534
000260	SYBIL	P	JOHNSON	D21	8953	1975-09-11	CLERK	16	F	1936-10-05	17250	300	1380
000270	MARIA	L	PEREZ	D21	9001	1980-09-30	CLERK	15	F	1953-05-26	27380	500	2190
000280	ETHEL	R	SCHNEIDER	E11	8997	1967-03-24	OPERATOR	17	F	1936-03-28	26250	500	2100
000290	JOHN	R	PARKER	E11	4502	1980-05-30	OPERATOR	12	M	1946-07-09	15340	300	1227
000300	PHILIP	X	SMITH	E11	2095	1972-06-19	OPERATOR	14	M	1936-10-27	17750	400	1420
000310	MAUDE	F	SETRIGHT	E11	3332	1964-09-12	OPERATOR	12	F	1931-04-21	15900	300	1272

範例資料庫表格

EMPNO	FIRSTNME	MID INIT	LASTNAME	WORK DEPT	PHONE NO	HIREDATE	JOB	ED LEVEL	SEX	BIRTHDATE	SALARY	BONUS	COMM
000320	RAMLAL	V	MEHTA	E21	9990	1965-07-07	FIELDREP	16	M	1932-08-11	19950	400	1596
000330	WING		LEE	E21	2103	1976-02-23	FIELDREP	14	M	1941-07-18	25370	500	2030
000340	JASON	R	GOUNOT	E21	5698	1947-05-05	FIELDREP	16	M	1926-05-17	23840	500	1907

範例資料庫表格

**EMP\_ACT** 表格

名稱：	EMPNO	PROJNO	ACTNO	EMPTIME	EMSTDATE	EMENDATE
類型：	char(6) not null	char(6) not null	smallint not null	dec(5,2)	date	date
說明：	員工編號	專案編號	活動編號	員工花在專案上的時間比例	活動開始日期	活動結束日期
值：	000010	AD3100	10	.50	1982-01-01	1982-07-01
	000070	AD3110	10	1.00	1982-01-01	1983-02-01
	000230	AD3111	60	1.00	1982-01-01	1982-03-15
	000230	AD3111	60	.50	1982-03-15	1982-04-15
	000230	AD3111	70	.50	1982-03-15	1982-10-15
	000230	AD3111	80	.50	1982-04-15	1982-10-15
	000230	AD3111	180	1.00	1982-10-15	1983-01-01
	000240	AD3111	70	1.00	1982-02-15	1982-09-15
	000240	AD3111	80	1.00	1982-09-15	1983-01-01
	000250	AD3112	60	1.00	1982-01-01	1982-02-01
	000250	AD3112	60	.50	1982-02-01	1982-03-15
	000250	AD3112	60	.50	1982-12-01	1983-01-01
	000250	AD3112	60	1.00	1983-01-01	1983-02-01
	000250	AD3112	70	.50	1982-02-01	1982-03-15
	000250	AD3112	70	1.00	1982-03-15	1982-08-15
	000250	AD3112	70	.25	1982-08-15	1982-10-15
	000250	AD3112	80	.25	1982-08-15	1982-10-15
	000250	AD3112	80	.50	1982-10-15	1982-12-01
	000250	AD3112	180	.50	1982-08-15	1983-01-01
	000260	AD3113	70	.50	1982-06-15	1982-07-01
	000260	AD3113	70	1.00	1982-07-01	1983-02-01
	000260	AD3113	80	1.00	1982-01-01	1982-03-01
	000260	AD3113	80	.50	1982-03-01	1982-04-15
	000260	AD3113	180	.50	1982-03-01	1982-04-15
	000260	AD3113	180	1.00	1982-04-15	1982-06-01
	000260	AD3113	180	.50	1982-06-01	1982-07-01
	000270	AD3113	60	.50	1982-03-01	1982-04-01
	000270	AD3113	60	1.00	1982-04-01	1982-09-01
	000270	AD3113	60	.25	1982-09-01	1982-10-15
	000270	AD3113	70	.75	1982-09-01	1982-10-15
	000270	AD3113	70	1.00	1982-10-15	1983-02-01
	000270	AD3113	80	1.00	1982-01-01	1982-03-01

範例資料庫表格

名稱：	EMPNO	PROJNO	ACTNO	EMPTIME	EMSTDATE	EMENDATE
	000270	AD3113	80	.50	1982-03-01	1982-04-01
	000030	IF1000	10	.50	1982-06-01	1983-01-01
	000130	IF1000	90	1.00	1982-01-01	1982-10-01
	000130	IF1000	100	.50	1982-10-01	1983-01-01
	000140	IF1000	90	.50	1982-10-01	1983-01-01
	000030	IF2000	10	.50	1982-01-01	1983-01-01
	000140	IF2000	100	1.00	1982-01-01	1982-03-01
	000140	IF2000	100	.50	1982-03-01	1982-07-01
	000140	IF2000	110	.50	1982-03-01	1982-07-01
	000140	IF2000	110	.50	1982-10-01	1983-01-01
	000010	MA2100	10	.50	1982-01-01	1982-11-01
	000110	MA2100	20	1.00	1982-01-01	1982-03-01
	000010	MA2110	10	1.00	1982-01-01	1983-02-01
	000200	MA2111	50	1.00	1982-01-01	1982-06-15
	000200	MA2111	60	1.00	1982-06-15	1983-02-01
	000220	MA2111	40	1.00	1982-01-01	1983-02-01
	000150	MA2112	60	1.00	1982-01-01	1982-07-15
	000150	MA2112	180	1.00	1982-07-15	1983-02-01
	000170	MA2112	60	1.00	1982-01-01	1983-06-01
	000170	MA2112	70	1.00	1982-06-01	1983-02-01
	000190	MA2112	70	1.00	1982-02-01	1982-10-01
	000190	MA2112	80	1.00	1982-10-01	1983-10-01
	000160	MA2113	60	1.00	1982-07-15	1983-02-01
	000170	MA2113	80	1.00	1982-01-01	1983-02-01
	000180	MA2113	70	1.00	1982-04-01	1982-06-15
	000210	MA2113	80	.50	1982-10-01	1983-02-01
	000210	MA2113	180	.50	1982-10-01	1983-02-01
	000050	OP1000	10	.25	1982-01-01	1983-02-01
	000090	OP1010	10	1.00	1982-01-01	1983-02-01
	000280	OP1010	130	1.00	1982-01-01	1983-02-01
	000290	OP1010	130	1.00	1982-01-01	1983-02-01
	000300	OP1010	130	1.00	1982-01-01	1983-02-01
	000310	OP1010	130	1.00	1982-01-01	1983-02-01
	000050	OP2010	10	.75	1982-01-01	1983-02-01
	000100	OP2010	10	1.00	1982-01-01	1983-02-01
	000320	OP2011	140	.75	1982-01-01	1983-02-01
	000320	OP2011	150	.25	1982-01-01	1983-02-01

## 範例資料庫表格

名稱：	EMPNO	PROJNO	ACTNO	EMPTIME	EMSTDATE	EMENDATE
	000330	OP2012	140	.25	1982-01-01	1983-02-01
	000330	OP2012	160	.75	1982-01-01	1983-02-01
	000340	OP2013	140	.50	1982-01-01	1983-02-01
	000340	OP2013	170	.50	1982-01-01	1983-02-01
	000020	PL2100	30	1.00	1982-01-01	1982-09-15

## EMP\_PHOTO 表格

名稱：	EMPNO	PHOTO_FORMAT	PICTURE
類型：	char(6) not null	varchar(10) not null	blob(100k)
說明：	員工編號	相片格式	員工相片
值：	000130	點陣圖	db200130.bmp
	000130	gif	db200130.gif
	000130	xwd	db200130.xwd
	000140	點陣圖	db200140.bmp
	000140	gif	db200140.gif
	000140	xwd	db200140.xwd
	000150	點陣圖	db200150.bmp
	000150	gif	db200150.gif
	000150	xwd	db200150.xwd
	000190	點陣圖	db200190.bmp
	000190	gif	db200190.gif
	000190	xwd	db200190.xwd

- 第84頁的『Quintana 相片』顯示了員工 Delores Quintana 的相片。
- 第86頁的『Nicholls 相片』顯示了員工 Heather Nicholls 的相片。
- 第87頁的『Adamson 相片』顯示了員工 Bruce Adamson 的相片。
- 第88頁的『Walker 相片』顯示了員工 James Walker 的相片。

## EMP\_RESUME 表格

名稱：	EMPNO	RESUME_FORMAT	RESUME
類型：	char(6) not null	varchar(10) not null	clob(5k)
說明：	員工編號	履歷表格式	員工履歷表
值：	000130	ascii	db200130.asc
	000130	script	db200130.scr
	000140	ascii	db200140.asc

## 範例資料庫表格

名稱：	EMPNO	RESUME_FORMAT	RESUME
	000140	script	db200140.scr
	000150	ascii	db200150.asc
	000150	script	db200150.scr
	000190	ascii	db200190.asc
	000190	script	db200190.scr

- 第85頁的『Quintana 履歷表』顯示了員工 Delores Quintana 的履歷表。
- 第86頁的『Nicholls 履歷表』顯示了員工 Heather Nicholls 的履歷表。
- 第87頁的『Adamson 履歷表』顯示了員工 Bruce Adamson 的履歷表。
- 第89頁的『Walker 履歷表』顯示了員工 James Walker 的履歷表。

## IN\_TRAY 表格

名稱：	RECEIVED	SOURCE	SUBJECT	NOTE_TEXT
類型：	timestamp	char(8)	char(64)	varchar(3000)
說明：	接收日期和時間	傳送便箋人員的使用者 ID	扼要說明	便箋

## ORG 表格

名稱：	DEPTNUMB	DEPTNAME	MANAGER	DIVISION	LOCATION
類型：	smallint not null	varchar(14)	smallint	varchar(10)	varchar(13)
說明：	部門編號	部門名稱	經理編號	公司部門	城市
值：	10	Head Office	160	Corporate	New York
	15	New England	50	Eastern	Boston
	20	Mid Atlantic	10	Eastern	Washington
	38	South Atlantic	30	Eastern	Atlanta
	42	Great Lakes	100	Midwest	Chicago
	51	Plains	140	Midwest	Dallas
	66	Pacific	270	Western	San Francisco
	84	Mountain	290	Western	Denver

## PROJECT 表格

名稱：	PROJNO	PROJNAME	DEPTNO	RESPEMP	PRSTAFF	PRSTDATE	PRENDATE	MAJPROJ
類型：	char(6) not null	varchar(24) not null	char(3) not null	char(6) not null	dec(5,2)	date	date	char(6)

## 範例資料庫表格

名稱：	PROJNO	PROJNAME	DEPTNO	RESPEMP	PRSTAFF	PRSTDATE	PRENDATE	MAJPROJ
說明：	專案編號	專案名稱	負責部門	負責員工	預估平均所需 員工數	預計開始日期	預計完成日期	主專案 (次專 案適用)
值：	AD3100	ADMIN SERVICES	D01	000010	6.5	1982-01-01	1983-02-01	-
	AD3110	GENERAL ADMIN SYSTEMS	D21	000070	6	1982-01-01	1983-02-01	AD3100
	AD3111	PAYROLL PROGRAMMING	D21	000230	2	1982-01-01	1983-02-01	AD3110
	AD3112	PERSONNEL PROGRAMMING	D21	000250	1	1982-01-01	1983-02-01	AD3110
	AD3113	ACCOUNT PROGRAMMING	D21	000270	2	1982-01-01	1983-02-01	AD3110
	IF1000	QUERY SERVICES	C01	000030	2	1982-01-01	1983-02-01	-
	IF2000	USER EDUCATION	C01	000030	1	1982-01-01	1983-02-01	-
	MA2100	WELD LINE AUTOMATION	D01	000010	12	1982-01-01	1983-02-01	-
	MA2110	W L PROGRAMMING	D11	000060	9	1982-01-01	1983-02-01	MA2100
	MA2111	W L PROGRAM DESIGN	D11	000220	2	1982-01-01	1982-12-01	MA2110
	MA2112	W L ROBOT DESIGN	D11	000150	3	1982-01-01	1982-12-01	MA2110
	MA2113	W L PROD CONT PROGS	D11	000160	3	1982-02-15	1982-12-01	MA2110
	OP1000	OPERATION SUPPORT	E01	000050	6	1982-01-01	1983-02-01	-
	OP1010	OPERATION	E11	000090	5	1982-01-01	1983-02-01	OP1000
	OP2000	GEN SYSTEMS SERVICES	E01	000050	5	1982-01-01	1983-02-01	-
	OP2010	SYSTEMS SUPPORT	E21	000100	4	1982-01-01	1983-02-01	OP2000
	OP2011	SCP SYSTEMS SUPPORT	E21	000320	1	1982-01-01	1983-02-01	OP2010
	OP2012	APPLICATIONS SUPPORT	E21	000330	1	1982-01-01	1983-02-01	OP2010
	OP2013	DB/DC SUPPORT	E21	000340	1	1982-01-01	1983-02-01	OP2010
	PL2100	WELD LINE PLANNING	B01	000020	1	1982-01-01	1982-09-15	MA2100



## SALES 表格

名稱：	SALES_DATE	SALES_PERSON	REGION	SALES
類型：	date	varchar(15)	varchar(15)	int
說明：	銷售日期	員工姓氏	銷售區域	銷售數目
值：	12/31/1995	LUCCHESSI	Ontario-South	1
	12/31/1995	LEE	Ontario-South	3
	12/31/1995	LEE	Quebec	1
	12/31/1995	LEE	Manitoba	2
	12/31/1995	GOUNOT	Quebec	1
	03/29/1996	LUCCHESSI	Ontario-South	3
	03/29/1996	LUCCHESSI	Quebec	1
	03/29/1996	LEE	Ontario-South	2
	03/29/1996	LEE	Ontario-North	2
	03/29/1996	LEE	Quebec	3
	03/29/1996	LEE	Manitoba	5
	03/29/1996	GOUNOT	Ontario-South	3
	03/29/1996	GOUNOT	Quebec	1
	03/29/1996	GOUNOT	Manitoba	7
	03/30/1996	LUCCHESSI	Ontario-South	1
	03/30/1996	LUCCHESSI	Quebec	2
	03/30/1996	LUCCHESSI	Manitoba	1
	03/30/1996	LEE	Ontario-South	7
	03/30/1996	LEE	Ontario-North	3
	03/30/1996	LEE	Quebec	7
	03/30/1996	LEE	Manitoba	4
	03/30/1996	GOUNOT	Ontario-South	2
	03/30/1996	GOUNOT	Quebec	18
	03/30/1996	GOUNOT	Manitoba	1
	03/31/1996	LUCCHESSI	Manitoba	1
	03/31/1996	LEE	Ontario-South	14
	03/31/1996	LEE	Ontario-North	3
	03/31/1996	LEE	Quebec	7
	03/31/1996	LEE	Manitoba	3
	03/31/1996	GOUNOT	Ontario-South	2
	03/31/1996	GOUNOT	Quebec	1
	04/01/1996	LUCCHESSI	Ontario-South	3
	04/01/1996	LUCCHESSI	Manitoba	1
	04/01/1996	LEE	Ontario-South	8
	04/01/1996	LEE	Ontario-North	-
	04/01/1996	LEE	Quebec	8
	04/01/1996	LEE	Manitoba	9
	04/01/1996	GOUNOT	Ontario-South	3

## 範例資料庫表格

名稱：	SALES_DATE	SALES_PERSON	REGION	SALES
	04/01/1996	GOUNOT	Ontario-North	1
	04/01/1996	GOUNOT	Quebec	3
	04/01/1996	GOUNOT	Manitoba	7

## STAFF 表格

名稱：	ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
類型：	smallint not null	varchar(9)	smallint	char(5)	smallint	dec(7,2)	dec(7,2)
說明：	員工編號	員工姓名	部門編號	工作類型	服務年資	目前薪資	佣金
值：	10	Sanders	20	Mgr	7	18357.50	-
	20	Pernal	20	Sales	8	18171.25	612.45
	30	Marenghi	38	Mgr	5	17506.75	-
	40	O'Brien	38	Sales	6	18006.00	846.55
	50	Hanes	15	Mgr	10	20659.80	-
	60	Quigley	38	Sales	-	16808.30	650.25
	70	Rothman	15	Sales	7	16502.83	1152.00
	80	James	20	CLERK	-	13504.60	128.20
	90	Koonitz	42	Sales	6	18001.75	1386.70
	100	Plotz	42	Mgr	7	18352.80	-
	110	Ngan	15	CLERK	5	12508.20	206.60
	120	Naughton	38	CLERK	-	12954.75	180.00
	130	Yamaguchi	42	CLERK	6	10505.90	75.60
	140	Fraye	51	Mgr	6	21150.00	-
	150	Williams	51	Sales	6	19456.50	637.65
	160	Molinare	10	Mgr	7	22959.20	-
	170	Kermisch	15	CLERK	4	12258.50	110.10
	180	Abrahams	38	CLERK	3	12009.75	236.50
	190	Sneider	20	CLERK	8	14252.75	126.50
	200	Scoutten	42	CLERK	-	11508.60	84.20
	210	Lu	10	Mgr	10	20010.00	-
	220	Smith	51	Sales	7	17654.50	992.80
	230	Lundquist	51	CLERK	3	13369.80	189.65
	240	Daniels	10	Mgr	5	19260.25	-
	250	Wheeler	51	CLERK	6	14460.00	513.30
	260	Jones	10	Mgr	12	21234.00	-
	270	Lea	66	Mgr	9	18555.50	-
	280	Wilson	66	Sales	9	18674.50	811.50

範例資料庫表格

名稱：	ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
	290	Quill	84	Mgr	10	19818.00	-
	300	Davis	84	Sales	5	15454.50	806.10
	310	Graham	66	Sales	13	21000.00	200.30
	320	Gonzales	66	Sales	4	16858.20	844.00
	330	Burke	66	CLERK	1	10988.00	55.50
	340	Edwards	84	Sales	7	17844.00	1285.00
	350	Gafney	84	CLERK	5	13030.50	188.00

STAFFG 表格

註：STAFFG 僅限雙位元組字碼頁使用。

名稱：	ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
類型：	smallint not null	varchar(9)	smallint	graphic(5)	smallint	dec(9,0)	dec(9,0)
說明：	員工編號	員工姓名	部門編號	工作類型	服務年資	目前薪資	佣金
值：	10	Sanders	20	Mgr	7	18357.50	-
	20	Pernal	20	Sales	8	18171.25	612.45
	30	Marengi	38	Mgr	5	17506.75	-
	40	O'Brien	38	Sales	6	18006.00	846.55
	50	Hanes	15	Mgr	10	20659.80	-
	60	Quigley	38	Sales	-	16808.30	650.25
	70	Rothman	15	Sales	7	16502.83	1152.00
	80	James	20	CLERK	-	13504.60	128.20
	90	Koonitz	42	Sales	6	18001.75	1386.70
	100	Plotz	42	Mgr	7	18352.80	-
	110	Ngan	15	CLERK	5	12508.20	206.60
	120	Naughton	38	CLERK	-	12954.75	180.00
	130	Yamaguchi	42	CLERK	6	10505.90	75.60
	140	Fraye	51	Mgr	6	21150.00	-
	150	Williams	51	Sales	6	19456.50	637.65
	160	Molinare	10	Mgr	7	22959.20	-
	170	Kermisch	15	CLERK	4	12258.50	110.10
	180	Abrahams	38	CLERK	3	12009.75	236.50
	190	Sneider	20	CLERK	8	14252.75	126.50
	200	Scoutten	42	CLERK	-	11508.60	84.20
	210	Lu	10	Mgr	10	20010.00	-

## 範例資料庫表格

名稱：	ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
	220	Smith	51	Sales	7	17654.50	992.80
	230	Lundquist	51	CLERK	3	13369.80	189.65
	240	Daniels	10	Mgr	5	19260.25	-
	250	Wheeler	51	CLERK	6	14460.00	513.30
	260	Jones	10	Mgr	12	21234.00	-
	270	Lea	66	Mgr	9	18555.50	-
	280	Wilson	66	Sales	9	18674.50	811.50
	290	Quill	84	Mgr	10	19818.00	-
	300	Davis	84	Sales	5	15454.50	806.10
	310	Graham	66	Sales	13	21000.00	200.30
	320	Gonzales	66	Sales	4	16858.20	844.00
	330	Burke	66	CLERK	1	10988.00	55.50
	340	Edwards	84	Sales	7	17844.00	1285.00
	350	Gafney	84	CLERK	5	13030.50	188.00

## 資料類型為 BLOB 和 CLOB 的範例檔案

這一節列出可於 EMP\_PHOTO 檔 (員工相片) 和 EMP\_RESUME 檔 (員工履歷表) 中找到的資料。

### Quintana 相片



圖 7. *Delores M. Quintana*

**Quintana 履歷表**

以下的本文出現在 db200130.asc 和 db200130.scr 檔內。

**履歷表： Delores M. Quintana****個人資料**

地址： 1150 Eglinton Ave Mellonville, Idaho 83725  
 電話： (208) 555-9933  
 出生年月日： 1925 年 9 月 15 日  
 性別： 女  
 婚姻狀態： 已婚  
 身高： 5 呎 2 吋  
 體重： 120 磅

**部門資料**

員工編號： 000130  
 部門編號： C01  
 經理： Sally Kwan  
 職位： 分析師  
 電話： (208) 555-4578  
 雇用日期： 1971-07-28

**教育程度**

1965 艾達菲大學數學和英文學士  
 1960 佛羅里達技術學院牙醫技術士

**工作經歷**

10/91 - 迄今 替工程部門設計文件工具的系統分析顧問。  
 12/85 - 9/91 技術文件撰寫員撰寫、本文編排及規劃。  
 1/79 - 11/85 替柴油公司撰寫薪資程式的 COBOL 薪資程式設計師。

**興趣**

- 烹飪
- 閱讀
- 裁縫
- 室內裝璜

## 範例資料庫表格

### Nicholls 相片



圖 8. Heather A. Nicholls

### Nicholls 履歷表

以下的本文出現在 db200140.asc 和 db200140.scr 檔內。

#### 履歷表： Heather A. Nicholls

##### 個人資料

地址：	844 Don Mills Ave Mellonville, Idaho 83734
電話：	(208) 555-2310
出生年月日：	1946 年 1 月 19 日
性別：	女
婚姻狀態：	單身
身高：	5 呎 8 吋
體重：	130 磅

##### 部門資料

員工編號：	000140
部門編號：	C01
經理：	Sally Kwan
職位：	分析師
電話：	(208) 555-1793
雇用日期：	1976-12-15

##### 教育程度

## 範例資料庫表格

<b>1972</b>	華盛頓大學電腦工程博士
<b>1969</b>	華沙學院音樂與物理學碩士
<b>工作經歷</b>	
<b>2/83 - 迄今</b>	設計師，OCR 發展，設計 OCR 產品之結構。
<b>12/76 - 1/83</b>	本文設計師，利用 PL/I 語言作光學字元辨識 (OCR) 程式設計。
<b>9/72 - 11/76</b>	打孔卡品質分析員，檢查打孔卡是否符合品質規格。
<b>興趣</b>	
	<ul style="list-style-type: none"><li>• 鐵道模型</li><li>• 室內設計</li><li>• 刺繡</li><li>• 編織</li></ul>

## Adamson 相片



圖 9. Bruce Adamson

## Adamson 履歷表

以下的本文出現在 db200150.asc 和 db200150.scr 檔內。

履歷表：**Bruce Adamson**

個人資料

## 範例資料庫表格

地址：	3600 Steeles Ave Mellonville, Idaho 83757
電話：	(208) 555-4489
出生年月日：	1947 年 5 月 17 日
性別：	男
婚姻狀態：	已婚
身高：	6 呎 0 吋
體重：	175 磅
<b>部門資料</b>	
員工編號：	000150
部門編號：	D11
經理：	Irving Stern
職位：	設計人員
電話：	(208) 555-4510
雇用日期：	1972-02-12
<b>教育程度</b>	
<b>1971</b>	約翰霍普金斯大學環境工程碩士
<b>1968</b>	西北大學美國歷史學士
<b>工作經歷</b>	
<b>8/79 - 迄今</b>	神經網路設計，為機器智慧產品發展神經網路。
<b>2/72 - 7/79</b>	機器人視覺發展，發展以規則為本的系統以模擬人類視力。
<b>9/71 - 1/72</b>	數字整合專員，協助銀行系統之間的通信。
<b>興趣</b>	
<ul style="list-style-type: none"><li>• 機車賽車</li><li>• 製作音響擴大機</li><li>• 組裝個人電腦</li><li>• 素描</li></ul>	

## Walker 相片





圖 10. James H. Walker

### Walker 履歷表

以下的本文出現在 db200190.asc 和 db200190.scr 檔內。

履歷表：**James H. Walker**

#### 個人資料

地址： 3500 Steeles Ave Mellonville, Idaho 83757  
 電話： (208) 555-7325  
 出生年月日： 1952 年 6 月 25 日  
 性別： 男  
 婚姻狀態： 單身  
 身高： 5 呎 11 吋  
 體重： 166 磅

#### 部門資料

員工編號： 000190  
 部門編號： D11  
 經理： Irving Stern  
 職位： 設計人員  
 電話： (208) 555-2986  
 雇用日期： 1974-07-26

#### 教育程度

1974 麻州大學計算機研究學士  
 1972 多倫多大學語言人類學學士

## 範例資料庫表格

### 工作經歷

**6/87 - 迄今**

微碼設計，使數學函數的演算法最佳化。

**4/77 - 5/87**

印表機技術支援，安裝及支援雷射印表機。

**9/74 - 3/77**

維護程式設計，替大型電腦修補組譯語言編譯器。

### 興趣

- 品酒
- 滑雪
- 游泳
- 舞蹈

---

## 附錄B. 使用 DB2 檔案庫

DB2 Universal Database 檔案庫是由線上說明、手冊 (PDF 及 HTML)及 HTML 格式的範例程式所組成。本節將描述此檔案庫所提供的資訊，以及存取此檔案庫的方法。

若要取得線上產品資訊，您可以使用「資訊中心」。相關資訊，請參閱第104頁的『用資訊中心來存取資訊』。您可以在 Web 上檢視作業資訊、疑難排解資訊、範例程式及 DB2 資訊。

---

### DB2 PDF 檔案與列印的書籍

#### DB2 資訊

下列表格將 DB2 書籍分成四類：

##### DB2 手冊與參考資訊

這些書籍包含所有平台的一般 DB2 資訊。

##### DB2 安裝與架構資訊

這些書籍適用於特定平台上的 DB2。例如，針對各個不同的作業平台快速入門如 OS/2、Windows、UNIX 等的書籍。

##### HTML 格式的跨平台範例程式

這些範例為 HTML 版的範例程式，會隨 Application Development Client 一起安裝。這些範例為參考用資訊，並不會取代實際的程式。

##### 版本注意事項

這些檔案包含 DB2 書籍中未包含的最新資訊。

您可以從產品 CD-ROM 中，直接檢視 HTML 格式的安裝手冊、版次注意事項及教學指導。大部份的書籍以 HTML 格式存在產品 CD-ROM 中，以供檢視，而以 Adobe Acrobat (PDF) 格式存在 DB2 出版品 CD-ROM 中，供檢視與列印。您也可以從 IBM 訂購印刷的書籍；請參閱 第100頁的『訂購印刷書籍』。下表會列出可以訂購的書籍。

在 OS/2 及 Windows 平台上，您可以在 sqllib\doc\html 目錄中安裝 HTML 檔案。DB2 資訊會轉換為不同的語言；然而，不是所有資訊都可以轉換成每一種語言。該資訊無特定語言版本時，則提供英文資訊

在 UNIX 平台中，您可以在 doc/%L/html 目錄中安裝多種語言版本的 HTML 檔案，其中 %L 代表語言環境。若需其餘相關資訊，請參照適當的快速入門書籍。

您可以使用不同方式，取得 DB2 書籍及存取資訊：

- 第103頁的『檢視線上資訊』
- 第107頁的『搜尋線上資訊』
- 第100頁的『訂購印刷書籍』
- 第99頁的『列印 PDF 書籍』

表 1. DB2 資訊

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<b>DB2 手冊與參考資訊</b>			
<i>Administration Guide</i>	<p><i>Administration Guide: 規劃</i> 提供資料庫概念的綜覽、設計事項的相關資訊 (如邏輯及實體資料庫設計) 及高可用性的討論。</p> <p><i>Administration Guide: 實作</i> 提供施行事項的相關資訊，如施行您的設計、存取資料庫、審核、備份及回復。</p> <p><i>Administration Guide: 效能</i> 提供資料庫環境及應用程式效能評估及調整的相關資訊。</p> <p>您可洽北美服務中心，訂購這三本英文版的 <i>Administration Guide</i>，書號為 SBOF-8934。</p>	<p>SC09-2946 db2d1x70</p> <p>SC09-2944 db2d2x70</p> <p>SC09-2945 db2d3x70</p>	db2d0
<i>Administrative API Reference</i>	說明您可以用來管理資料庫的 DB2 應用程式設計介面 (API) 及資料結構。本書也解釋如何從應用程式呼叫 API。	SC09-2947 db2b0x70	db2b0
<i>應用程式開發手冊</i>	提供環境安裝資訊以及逐步的指示，教您如何在 Windows、OS/2 及 UNIX 平台上，編譯、鏈結及執行 DB2 應用程式。	SC40-0493 db2axx70	db2ax
<i>APPC、CPI-C 與 SNA Sense Codes</i>	提供有關您使用 DB2 Universal Database 產品時，可能會遇到之 APPC、CPI-C 及 SNA 感應碼的一般資訊。	沒有書號 db2apx70	db2ap
	只提供 HTML 格式。		

表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<i>Application Development Guide</i>	解釋如何使用內含的 SQL 或 Java (JDBC 及 SQLJ) 開發存取 DB2 資料庫的應用程式。討論主題包含在分段的环境中，或使用聯合系統撰寫儲存程序、撰寫使用者定義功能、建立使用者定義類型、使用觸發函式及開發應用程式。	SC09-2949 db2a0x70	db2a0
<i>CLI Guide and Reference</i>	說明如何使用 DB2 CLI 這個可呼叫的 SQL 介面 (與 Microsoft ODBC 規格相容) 來發展可存取 DB2 資料庫的應用程式。	SC09-2950 db2l0x70	db2l0
<i>Command Reference</i>	解釋如何使用「命令行處理器」，並說明您可以用來管理資料庫的 DB2 指令。	SC09-2951 db2n0x70	db2n0
連接環境補充資料	提供有關如何使用 DB2 for AS/400、DB2 for OS/390、DB2 for MVS 或 DB2 for VM 作為使用 DB2 Universal Database 伺服器的 DRDA 應用程式要求程式的設定及參考資料。本書亦詳細說明如何使用 DRDA 應用程式伺服器與 DB2 Connect 應用程式要求程式。  僅提供 HTML 及 PDF 格式。	沒有書號 db2h1x70	db2h1
<i>Data Movement Utilities Guide and Reference</i>	解釋如何使用 DB2 公用程式，如匯入、匯出、載入、AutoLoader 及 DPROP，以便利資料的移動。	SC09-2955 db2dmx70	db2dm
資料倉儲中心 管理手冊	提供使用「資料倉儲中心」，如何開發及維護資料倉儲的相關資訊。	SC40-4096 db2ddx70	db2dd
資料倉儲中心 <i>Application Integration Guide</i>	提供相關資訊，協助程式設計師整合應用程式與「資料倉儲中心」及「資訊型錄管理程式」。	SC26-9994 db2adx70	db2ad
<i>DB2 Connect User's Guide</i>	提供有關 DB2 Connect 產品的概念、程式設計及一般使用資訊。	SC09-2954 db2c0x70	db2c0
<i>DB2 Query Patroller Administration Guide</i>	提供 DB2 Query Patroller 系統的作業概觀、特定的作業及管理資訊與作業資訊，供管理圖形式使用者介面公用程式使用。	SC09-2958 db2dwx70	db2dw

表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<i>DB2 Query Patroller User's Guide</i>	說明如何使用 DB2 Query Patroller 的工具及功能。	SC09-2960	db2ww
		db2wwx70	
<i>Glossary</i>	提供在 DB2 及其元件中所使用的術語定義。	沒有書號	db2t0
		db2t0x70	
	提供 HTML 格式及在 <i>SQL Reference</i> 中讀取。		
<i>Image, Audio, and Video Extenders 管理與程式設計手冊</i>	提供有關 DB2 擴充元的一般資訊，並提供有關管理及架構映像檔、音效及影像 (IAV) 擴充元，及有關利用 IAV 擴充元進行程式設計的資訊。它包含了參考資料、診斷資訊 (附有訊息) 及範例。	SC40-0525	dmbu7
		dmbu7x70	
<i>資訊型錄管理程式 管理手冊</i>	提供有關管理資訊型錄的指引。	SC40-0497	db2di
		db2dix70	
<i>資訊型錄管理程式 Programming Guide and Reference</i>	提供「資訊型錄管理程式」的架構介面定義。	SC26-9997	db2bi
		db2bix70	
<i>資訊型錄管理程式 使用手冊</i>	提供使用「資訊型錄管理程式」使用者介面的相關資訊。	SC40-0498	db2ai
		db2aix70	
<i>安裝與架構補充</i>	指引您規劃、安裝及設定特定平台的 DB2 從屬站。此補充資訊也包含了連結、設定從屬站及伺服器通信、DB2 GUI 工具、DRDA AS、分散式安裝、架構分散式要求及存取不同資料來源等的相關資訊。	GC40-0480	db2iy
		db2iyx70	
<i>Message Reference</i>	列示由 DB2、資訊型錄管理程式 及資料倉儲中心 所發出的訊息與訊息碼，並說明您應採取的動作。	第一冊 GC40-0491	db2m0
		db2m1x70	
		第二冊 GC40-0492	
		SBOF-8932。	
		db2m2x70	
<i>OLAP Integration Server Administration Guide</i>	解釋如何使用 OLAP Integration Server 的「管理管理程式」(Administration Manager) 元件。	SC27-0787	無
		db2dpx70	

表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<i>OLAP Integration Server Metaoutline User's Guide</i>	解釋如何使用標準 OLAP meta 框架介面 (不使用「meta 框架輔助程式») 建立及移入 OLAP meta 框架。	SC27-0784 db2upx70	無
<i>OLAP Integration Server Model User's Guide</i>	解釋如何利用標準「OLAP 模型介面」(而不使用「模型輔助程式») 來建立 OLAP 模型。	SC27-0783 db2lpx70	無
<i>OLAP Setup and User's Guide</i>	提供「OLAP 起始者套件」(OLAP Starter Kit) 的架構及設定資訊。	SC40-0520 db2ipx70	db2ip
<i>OLAP Spreadsheet Add-in for Excel 使用手冊</i>	說明如何使用 Excel 試算表程式來分析 OLAP 資料。	SC27-0786 db2epx70	db2ep
<i>OLAP Spreadsheet Add-in for Lotus 1-2-3 使用手冊</i>	說明如何使用 Lotus 1-2-3 試算表程式來分析 OLAP 資料。	SC27-0785 db2tpx70	db2tp
<i>Replication 指南與參考手冊</i>	提供 DB2 所附之「IBM 抄寫工具」的規劃、架構、管理及使用資訊。	SC40-0499 db2e0x70	db2e0
<i>Spatial Extender User's Guide and Reference</i>	提供有關安裝、架構、管理、程式設計及疑難排解 Spatial Extender 的資訊。亦提供空間資料概念的重要說明，並附有 Spatial Extender 的特定參考資料 (訊息及 SQL)。	SC40-0527 db2sbx70	db2sb
<i>SQL 入門</i>	介紹 SQL 概念，並提供許多建構及作業的範例。	SC40-0494 db2y0x70	db2y0
<i>SQL Reference, 第一冊及第二冊</i>	敘述 SQL 語法、語意與語言的規則。本書也包含版本間不相容處、產品限制及目錄畫面等相關資訊。  您可洽北美服務中心，訂購這兩本英文版的 <i>SQL 參考手冊</i> ，書號為 SBOF-8933。	第一冊 SC09-2974 db2s1x70 第二冊 SC09-2975 db2s2x70	db2s0
<i>System Monitor Guide and Reference</i>	敘述如何收集關於資料庫與資料庫管理程式的各種資訊。本書解釋如何使用該資訊來了解資料庫活動、增進效能並判斷問題產生的原因。	SC09-2956 db2f0x70	db2f0

表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<i>Text Extender</i> 管理與程式設計手冊	提供有關 DB2 擴充元的一般資訊，並附有關於管理及架構 <i>Text Extender</i> ，及有關使用 <i>Text Extender</i> 進行程式設計的資訊。它包含了參考資料、診斷資訊 (附有訊息) 及範例。	SC40-0526	desu9
		desu9x70	
<i>Troubleshooting Guide</i>	協助您判斷錯誤的來源、從問題中回復，以及透過「DB2 客戶服務」的諮詢來使用診斷工具。	GC09-2850	db2p0
		db2p0x70	
新特性介紹	說明 DB2 Universal Database 版本 7 中的新特性、功能及加強功能。	SC40-0495	db2q0
		db2q0x70	
DB2 安裝與架構資訊			
<i>DB2 Connect Enterprise Edition for OS/2 與 Windows</i> 快速入門	提供在 OS/2 及 Windows 32 位元作業系統上，DB2 Connect Enterprise Edition 的規劃、移轉、安裝與架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0479	db2c6
		db2c6x70	
<i>DB2 Connect Enterprise Edition for UNIX</i> 快速入門	提供在 UNIX 系列平台上，DB2 Connect Enterprise Edition 的規劃、移轉、安裝、架構及作業資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0478	db2cy
		db2cyx70	
<i>DB2 Connect Personal Edition</i> 快速入門	提供在 OS/2 及 Windows 32 位元作業系統上，DB2 Connect Personal Edition 的規劃、移轉、安裝、架構及作業資訊。本書亦包含所有支援從屬站的安裝及設定資訊。	GC40-0486	db2c1
		db2c1x70	
<b>DB2 Connect Personal Edition Quick Beginnings for Linux</b>	提供在所有支援 Linux 分送式系統上，DB2 Connect Personal Edition 的規劃、安裝、移轉及架構資訊。	GC09-2962	db2c4
		db2c4x70	
<i>DB2 Data Links Manager</i> 快速入門	提供 DB2 Data Links Manager 在 AIX 及 Windows 32 位元作業系統上的規劃、安裝、架構及作業資訊。	GC40-0485	db2z6
		db2z6x70	
<i>DB2 Enterprise - Extended Edition for UNIX</i> 快速入門	提供 DB2 Enterprise - Extended Edition 在 UNIX 系列平台上的規劃、安裝及架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0483	db2v3
		db2v3x70	



表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
<i>DB2 Enterprise - Extended Edition for Windows</i> 快速入門	提供 DB2 Enterprise - Extended Edition 在 Windows 32 位元作業系統上的規劃、安裝及架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0482	db2v6
		db2v6x70	
<i>DB2 for OS/2</i> 快速入門	提供 OS/2 作業系統上之 DB2 Universal Database Personal Edition 的規劃、安裝、移轉及架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0487	db2i2
		db2i2x70	
<i>DB2 for UNIX</i> 快速入門	提供 UNIX 平台上 DB2 Universal Database Personal Edition 的規劃、安裝、移轉及架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0489	db2ix
		db2ixx70	
<i>DB2 for Windows</i> 快速入門	提供 DB2 Universal Database 在 Windows 32 位元作業系統上的規劃、安裝、移轉及架構資訊。本書亦包含了許多支援從屬站的安裝及設定資訊。	GC40-0490	db2i6
		db2i6x70	
<i>DB2 Personal Edition</i> 快速入門	提供 DB2 Universal Database Personal Edition 在 OS/2 及 Windows 32 位元作業系統上的規劃、安裝、移轉及架構資訊。	GC40-0488	db2i1
		db2i1x70	
<b>DB2 Personal Edition Quick Beginnings for Linux</b>	提供 DB2 Universal Database Personal Edition 在所有支援 Linux 分散式系統上的規劃、安裝、移轉及架構資訊。	GC09-2972	db2i4
		db2i4x70	
<i>DB2 Query Patroller</i> 安裝手冊	提供有關 DB2 Query Patroller 的安裝資訊。	GC40-0481	db2iw
		db2iwx70	
<b>DB2 Warehouse Manager</b> 安裝手冊	提供有關倉儲代理程式、倉儲轉換程式及「資訊型錄管理程式」的安裝資訊。	GC40-0521	db2id
		db2idx70	
<b>HTML 格式的跨平台範例程式</b>			

表 1. DB2 資訊 (繼續)

名稱	說明	書號	HTML 目錄
		PDF 檔名	
HTML 格式的範例程式	以 HTML 格式提供在 DB2 支援的所有平台上，程式設計語言的範例程式。範例程式僅供參考。並非所有程式設計語言皆有範例可用。只有在安裝了 DB2 Application Development Client 時，才能使用 HTML 範例。  若需程式的其餘相關資訊，請參照應用程式開發手冊。	沒有書號	db2hs
版本注意事項			
DB2 Connect 版本注意事項	提供 DB2 Connect 書籍中未包含的最新資訊。	請參閱備註 #2。	db2cr
DB2 安裝注意事項	提供 DB2 書籍中未包含的最新安裝特定資訊。	僅附於產品 CD-ROM 中。	
DB2 版本注意事項	提供 DB2 書籍中未包含的所有 DB2 產品及特性的最新資訊。	請參閱備註 #2。	db2ir

**註:**

1. 檔名中第六個位置上的字元 *x* 表示書籍的語言版本。例如，檔名 db2d0e70 會識別英文版的 *Administration Guide*，而檔名 db2d0f70 則識別同一本書的法文版。下列字母會用在檔名的第六個位置上，以表示語言版本：

語言	識別字
巴西葡萄牙文	b
保加利亞文	u
捷克文	x
丹麥文	d
荷蘭文	q
英文	e
芬蘭文	y
法文	f
德文	g
希臘文	a
匈牙利文	h
義大利文	i
日文	j
韓文	k
挪威文	n

波蘭文	p
葡萄牙文	v
俄文	r
簡體中文	c
斯洛維尼亞文	l
西班牙文	z
瑞典文	s
繁體中文	t
土耳其文	m

2. 「版本注意事項」中可取得 DB2 書籍中未包含的最新資訊 (有兩種檔案格式，HTML 及 ASCII)。而 HTML 版本則可以從「資訊中心」及產品 CD-ROM 中取得。欲檢視 ASCII 檔：

- 在 UNIX 平台上，請參閱 Release.Notes 檔案。此檔案是位在 DB2DIR/Readme/%L 目錄中，其中 %L 代表語言環境名稱，而 DB2DIR 代表：
  - /usr/lpp/db2\_07\_01 (在 AIX 上)
  - /opt/IBMd2/V7.1 (在 HP-UX、PTX、Solaris、及 Silicon Graphics IRIX 上)
  - /usr/IBMd2/V7.1 (在 Linux 上)。
- 在其它平台上，請參閱 RELEASE.TXT 檔案。這個檔案位在產品安裝的目錄中。在 OS/2 平台上，您可以按兩下 **IBM DB2** 資料夾，然後按兩下 **版本注意事項** 圖示。

## 列印 PDF 書籍

如果您想擁有印妥的書籍副本，您可以列印 DB2 出版品 CD-ROM 上的 PDF 檔。利用 Adobe Acrobat Reader，您可以列印整本書或只列印特定範圍的頁數。若需檔案庫中各書籍的檔名，請參閱 第92頁的表1。

您可以從 Adobe 網站 (<http://www.adobe.com>) 取得最新版本的 Adobe Acrobat Reader。

DB2 出版品 CD-ROM 中已包含 PDF 檔案，其副檔名為 PDF。欲存取 PDF 檔：

1. 插入 DB2 出版品 CD-ROM。在 UNIX 系列平台上，裝載 DB2 出版品 CD-ROM。請參照您的 *快速入門* 一書，取得裝載程序。
2. 啓動 Acrobat Reader。
3. 從下列其中一個位置開啓想要的 PDF 檔：
  - 在 OS/2 及 Windows 平台上：
    - `x:\doc\language` 目錄，其中 *x* 代表 CD-ROM 光碟機，且 *language* 表示兩個字元的國碼，代表您所使用的語言 (例如，EN 代表英文)。

- 在 UNIX 平台上：  
CD-ROM 中的 `/cdrom/doc/%L` 目錄，其中 `/cdrom` 代表 CD-ROM 的裝載點，且 `%L` 代表想要的語言環境名稱。

您也可以從 CD-ROM 中將 PDF 檔複製到本端或本端磁碟機中，並從該處讀取檔案。

## 訂購印刷書籍

您可以利用書號銷售單 (SBOF) 各別或整組 (僅限北美洲) 訂購印刷 DB2 書籍。欲訂購書籍，請聯絡您的 IBM 授權經銷商或業務代表，如您在美國，請撥 1-800-879-2755，如在加拿大，請撥 1-800-IBM-4YOU。您也可以從出版品網頁 (<http://www.elink.ibm.com/pbl/pbl>) 訂購書籍。

有兩組書籍可供訂購。SBOF-8935 提供 DB2 Warehouse Manager 的參照及使用資訊。SBOF-8931 提供所有其它 DB2 Universal Database 產品及特性的參照及使用資訊。每一張 SBOF 的內容均列示在下列表格中：

表 2. 訂購印刷書籍

SBOF 編號	訂購書籍
SBOF-8931	<ul style="list-style-type: none"> <li>• Administration Guide: Planning</li> <li>• Administration Guide: Implementation</li> <li>• Administration Guide: Performance</li> <li>• Administrative API Reference</li> <li>• 應用程式開發指南</li> <li>• Application Development Guide</li> <li>• CLI Guide and Reference</li> <li>• Command Reference</li> <li>• Data Movement Utilities Guide and Reference</li> <li>• 資料倉儲中心管理手冊</li> <li>• Data Warehouse Center Application Integration Guide</li> <li>• DB2 Connect User's Guide</li> <li>• 安裝與架構補充資料</li> <li>• Image, Audio, and Video Extenders 管理與規畫手冊</li> <li>• 訊息參考手冊, 第一冊與第二冊</li> <li>• OLAP Integration Server Administration Guide</li> <li>• OLAP Integration Server Metaoutline User's Guide</li> <li>• OLAP Integration Server Model User's Guide</li> <li>• OLAP Integration Server User's Guide</li> <li>• OLAP Setup and User's Guide</li> <li>• OLAP Spreadsheet Add-in User's Guide for Excel</li> <li>• OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3</li> <li>• Replication 指南與參考手冊</li> <li>• Spatial Extender Administration and Programming Guide</li> <li>• SQL 入門</li> <li>• SQL Reference, Volumes 1 and 2</li> <li>• System Monitor Guide and Reference</li> <li>• Text Extender 管理與程式設計</li> <li>• Troubleshooting Guide</li> <li>• 新特性介紹</li> </ul>
SBOF-8935	<ul style="list-style-type: none"> <li>• Information Catalog Manager 管理手冊</li> <li>• Information Catalog Manager 使用手冊</li> <li>• Information Catalog Manager Programming Guide and Reference</li> <li>• Query Patroller Administration Guide</li> <li>• Query Patroller User's Guide</li> </ul>

## DB2 線上文件

### 存取線上說明

所有 DB2 元件都有線上說明。下列表格將描述各種類型的說明。

說明類型	內容	如何存取...
指令說明	解釋命令行處理器中指令的語法。	<p>從交談模式的命令行處理器中，輸入：</p> <p style="text-align: center;"><code>? command</code></p> <p>其中 <i>command</i> 代表某一關鍵字或整個指令。</p> <p>例如，<code>? catalog</code> 將顯示所有 CATALOG 指令的說明，至於 <code>? catalog database</code> 則會顯示 CATALOG DATABASE 指令的說明</p>
從屬站架構輔助程式說明	說明您可以在視窗或筆記本中執行的作業。說明包含您必須知道的概觀及先決條件資訊，並說明如何使用視窗或筆記本控制項。	在視窗或筆記本中按一下 <b>說明</b> 按鈕，或按 <b>F1</b> 鍵。
命令中心說明		
控制中心說明		
資料倉儲中心說明		
事件分析程式說明		
資訊型錄管理程式說明		
衛星管理中心說明		
<b>Script 中心說明</b>		
訊息說明	說明訊息的原因，及所有您應採取的動作。	<p>從交談模式的命令行處理器中，輸入：</p> <p style="text-align: center;"><code>? XXXnnnnn</code></p> <p>其中 <i>XXXnnnnn</i> 代表有效的訊息 ID。</p> <p>例如，<code>? SQL30081</code> 將顯示關於 SQL30081 訊息的說明</p> <p>欲一次一個螢幕，檢視訊息說明，請輸入：</p> <p style="text-align: center;"><code>? XXXnnnnn</code>   <code>more</code></p> <p>欲將訊息說明儲存在檔案中，請輸入：</p> <p style="text-align: center;"><code>? XXXnnnnn &gt; filename.ext</code></p> <p>其中 <i>filename.ext</i> 代表您要儲存訊息說明的檔案。</p>

說明類型	內容	如何存取...
SQL 說明	解釋 SQL 陳述式的語法。	<p>從交談模式的命令行處理器中，輸入：</p> <pre>help statement</pre> <p>其中 <i>statement</i> 代表 SQL 陳述式。</p> <p>例如，<code>help SELECT</code> 會顯示有關 <code>SELECT</code> 陳述式的說明。</p> <p>註：UNIX 型的平台上沒有 SQL 說明。</p>
SQL 陳述式說明	解釋 SQL 陳述式及類別碼。	<p>從交談模式的命令行處理器中，輸入：</p> <pre>? sqlstate 或 ? class code</pre> <p>其中 <i>sqlstate</i> 代表有效的五位數 SQL 狀態，且 <i>class code</i> 代表 SQL 狀態的前兩位數。</p> <p>例如，<code>? 08003</code> 將顯示 08003 SQL 陳述式的說明，至於 <code>? 08</code> 則將顯示 08 類別碼的說明</p>

## 檢視線上資訊

隨本產品所附的書籍軟本均為超文字標記語言 (HTML) 格式。軟本格式可讓您搜尋或瀏覽資訊，並提供相關資訊的超文字鏈結。它同時也使得您的整個環境中，更易於共用此檔案庫。

您可以用符合 HTML 3.2 版規格的任何瀏覽器，來檢視線上書籍或範例程式。

欲檢視線上手冊或範例程式：

- 如果您正在執行 DB2 管理工具，請使用「資訊中心」。
- 在瀏覽器上按一下 **檔案** → **開啓網頁**。您所開啓的網頁含有 DB2 資訊的說明及鏈結：

- 在 UNIX 平台上，開啓下列網頁：

```
INSTHOME/sql1lib/doc/%L/html/index.htm
```

其中 *%L* 代表語言環境名稱。

- 在其它平台上，開啓下列網頁：

```
sql1lib\doc\html\index.htm
```

路徑位在 DB2 安裝所在的磁碟機上。

如果您尚未安裝「資訊中心」，您可以按兩下**DB2 資訊**圖示來開啓網頁。視您將使用的系統而定，圖示將位於主要產品資料夾或「Windows 啓動」功能表中。

### 安裝 Netscape 瀏覽器

如果您尚未安裝 Web 瀏覽器，您可以從產品包裝盒中的 Netscape CD-ROM 中安裝 Netscape。若須有關如何安裝的詳細指示，請執行下列：

1. 插入 Netscape CD-ROM。
2. 僅限於 UNIX 系列平台上，裝載 CD-ROM。請參照您的 *快速入門* 一書，取得裝載程序。
3. 若須安裝指示，請參照 CDNAV $nn$ .txt 檔，其中  $nn$  代表您的兩個字元的語言識別字。檔案是位在 CD-ROM 的根目錄中。

### 用資訊中心來存取資訊

「資訊中心」可讓您迅速地存取 DB2 產品資訊。具有 DB2 管理工具的所有平台皆有「資訊中心」。

您可以按兩下「資訊中心」圖示，開啓「資訊中心」。視您使用的系統而定，圖示會位在主產品資料夾的「資訊」資料夾或 Windows **開始**功能表中。

您也可以 DB2 Windows 平台上使用工具列及**說明**功能表，存取「資訊中心」。

「資訊中心」提供六種類型的資訊。按一下適當的標籤，可以查閱所提供的該類型的主题。

<b>作業</b>	列出您可以 DB2 執行的作業。
<b>參照</b>	DB2 參考資料、如關鍵字、指令及 API。
<b>書籍</b>	DB2 書籍。
<b>疑難排解</b>	錯誤訊息的種類及其回復動作。
<b>範例程式</b>	隨 DB2 Application Development Client 所附的範例程式。如果您未安裝 DB2 Application Development Client, 則不會顯示此標籤。
<b>Web</b>	全球資訊網 (WWW) 上的 DB2 資訊。欲存取這個資訊，您必須從您的系統中與 Web 連接。

當您在其中一個列示中選取某個項目時，「資訊中心」即會啓動一個檢視器，來顯示資訊。檢視器可以是系統說明檢視器、編輯器或 Web 瀏覽器，視您選取的資訊種類而定。

「資訊中心」提供尋找特性，所以您可以搜尋特定主题而無需瀏覽列示。



若需全文搜尋，則請遵循「資訊中心」中的超文字鏈結，進入**搜尋 DB2 線上資訊** 搜尋套表。

通常 HTML 搜尋伺服器會自動啓動。如果 HTML 資訊的搜尋無法運作，您可能必須使用下列方法之一，啓動搜尋伺服器：

#### 在 Windows 中

按一下**開始**，並選取**程式集** → **IBM DB2** → **資訊** → **啓動 HTML 搜尋伺服器**。

#### 在 OS/2 中

按兩下 **DB2 for OS/2** 資料夾，然後按兩下 **啓動 HTML 搜尋伺服器** 圖示。

如果您在搜尋 HTML 資訊時遭遇任何其它問題，請參考版本注意事項。

**註：**在 Linux、PTX 及 Silicon Graphics IRIX 環境中，無法使用「搜尋」功能。

## 使用 DB2 精靈

精靈在每一項作業中，可以逐步協助您完成特定的管理作業。您可以經由 控制中心 及從屬站架構輔助程式來使用精靈。下列會列出精靈並說明其目的。

**註：**「建立資料庫」、「建立索引」、「架構多位置更新」及「效能架構」精靈均可在 已分割的資料庫 環境中使用。

精靈	協助您...	如何存取...
新增資料庫	在從屬工作站上將資料庫編目。	從「從屬站架構輔助程式」中，按一下 <b>新增</b> 。
備份資料庫	備份計畫的決定、建立及排程。	在「控制中心」中，對著您要備份的資料庫按一下滑鼠右鍵，然後選取 <b>使用精靈，備份</b> → <b>資料庫</b> 。
架構多位置更新	架構多位置更新、分散式異動或兩階段確定。	在「控制中心」中，在 <b>資料庫</b> 資料夾上按一下滑鼠右鍵，然後選取 <b>多位置更新</b> 。
建立資料庫	建立一個資料庫，並執行某些基本架構作業。	在「控制中心」中，在 <b>資料庫</b> 資料夾上按一下滑鼠右鍵，然後選取 <b>使用精靈，建立</b> → <b>資料庫</b> 。
建立表格	選取基本資料類型，及建立表格的主要鍵。	在「控制中心」中，在 <b>表格</b> 圖示上按一下滑鼠右鍵，然後選取 <b>使用精靈，建立</b> → <b>表格</b> 。

精靈	協助您...	如何存取...
建立表格空間	建立新的表格空間。	在「控制中心」中，在 <b>表格空間</b> 圖示上按一下滑鼠右鍵，然後選取 <b>使用精靈</b> ， <b>建立</b> → <b>表格空間</b> 。
建立索引	針對您所有的查詢來建議您要建立及捨棄哪些索引。	在「控制中心」中，在 <b>索引</b> 圖示上按一下滑鼠右鍵，然後選取 <b>使用精靈</b> ， <b>建立</b> → <b>索引</b> 。
效能架構	藉由更新架構參數調整資料庫效能，以符合您業務上的需求。	在「控制中心」中，對著您要調整的資料庫按一下滑鼠右鍵，然後選取 <b>使用精靈</b> ， <b>架構效能</b> 。  至於在分段的資料庫環境中，則在「資料庫分割區」畫面中，對著您要調整的第一個資料庫分割區按一下滑鼠右鍵，然後選取 <b>使用精靈</b> ， <b>架構效能</b> 。
復置資料庫	錯誤發生後回復資料庫。它會幫助您了解使用哪一個備份及要回轉哪些日誌。	在「控制中心」中，對著您要復置的資料庫按一下滑鼠右鍵，然後選取 <b>使用精靈</b> ， <b>復置</b> → <b>資料庫</b> 。

## 設定文件伺服器

根據預設值，DB2 資訊會安裝到您的本端系統上。這表示每一位必須存取 DB2 資訊的人，必須安裝相同的檔案。欲將 DB2 資訊儲存在單一位置中，請執行下列步驟：

1. 在本端系統中，從 `\sqllib\doc\html` 複製所有的檔案及次目錄到 Web 伺服器。每一本書都有它自己的次目錄，其中包含了所有構成該書的必要 HTML 及 GIF 檔。請確定目錄結構沒有改變。
2. 架構 Web 伺服器，在新的位置搜尋檔案。若需相關資訊，請參照 **安裝與架構補充** 中的「NetQuestion 附錄」。
3. 如果使用 Java 版本的「資訊中心」，您可以對所有的 HTML 檔指定一個基礎 URL。您應該使用該 URL 取得書籍列示。
4. 當您可以檢視書籍檔案時，您可以在經常查閱的主題上加上書籤。您也許會想要將下列網頁加上書籤：
  - 書籍列示
  - 經常使用之書籍的目錄
  - 經常參考的文章，如「變更表」主題
  - 「搜尋」表格

若需如何從中央電腦上使用 DB2 Universal Database 線上文件檔的相關資訊，請參照安裝與架構補充 中的「NetQuestion 附錄」。

## 搜尋線上資訊

欲在 HTML 檔中尋找資訊，請使用下列方法之一：

- 按一下頂端訊框中的**搜尋**。使用搜尋表格頁面，來尋找特定主題。在 Linux、PTX 或 Silicon Graphics IRIX 環境中無法使用此功能。
- 按一下頂端訊框中的**索引**。使用索引，來找出書籍中的特定主題。
- 顯示目錄或說明或 HTML 書籍的索引，然後使用 Web 瀏覽器的尋找功能，找尋書中的特定主題。
- 使用 Web 瀏覽器的書籤功能，來迅速地回到特定主題。
- 使用「資訊中心」的搜尋功能，來找出特定主題。詳細資訊，請參閱第104頁的『用資訊中心來存取資訊』。



---

## 附錄C. 注意事項

而在其它國家中，IBM 不見得有提供本書中所提的各項產品、服務或功能。要知道在您所在之區是否可用到這些產品與服務時，請向當地的 IBM 服務代表查詢。本書在提及 IBM 產品、程式或服務時，不表示或暗示只能使用 IBM 的產品、程式或服務。只要未侵犯 IBM 的智慧財產權，任何功能相當的產品、程式或服務都可以取代 IBM 的產品、程式或服務。不過，其它非 IBM 產品、程式、或服務在運作上的評價與驗證，其責任屬於使用者。

在這本書或文件中可能包含著 IBM 所擁有之專利或專利申請案。本書使用者並不享有前述專利之任何授權。 您可以用書面方式來查詢授權，來函請寄到：

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

若要查詢有關二位元組 (DBCS) 資訊的特許權限事宜，請聯絡您國家的 IBM 智慧財產部門，或者用書面方式寄到：

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

下列段落若與該國之法律條款抵觸，即視為不適用：IBM 僅以現狀提供本書，而不提供任何明示或默示之保證 (包括但不限於可售性或符合特定效用的保證)。若有些地區在某些交易上並不允許排除上述保證，則該排除無效。

本書中可能會有技術上或排版印刷上的訛誤。因此，IBM 會定期修訂；並將修訂後的內容納入新版中。同時，IBM 得隨時改進並 (或) 變動本書中所提及的產品及 (或) 程式。

本書對於非 IBM 網站的援引只是為了方便而提供，並不對這些網站作任何認可。該些網站上的內容並非本 IBM 產品內容的一部份，用戶使用該網站時應自行承擔風險。

當您提供資訊給 IBM 時，您即授權予 IBM 以其認為適當的方式來使用或分送資訊，而不必對您負起任何責任。

本程式之獲授權者若希望取得相關資料，以便使用下列資訊者可洽詢 IBM。其下列資訊指的是：(1) 獨立建立的程式與其它程式 (包括此程式) 之間更換資訊的方式 (2) 相互使用已交換之資訊方法。若有任何問題請聯絡：

IBM Canada Limited  
Office of the Lab Director  
1150 Eglinton Ave. East  
North York, Ontario  
M3C 1H7  
CANADA

上述資料的取得有其特殊要件，在某些情況下必須付費方得使用。

本書所描述的任何程式及其所有可用的授權著作是由 IBM 所提供，並受到「IBM 客戶合約」、「國際程式授權合約」或雙方之間任何同等合約條款之規範。

此間所含之任何效能資料，皆是得自控制的環境之下；因此不同作業環境之下所得的結果，可能會有很大差異。部份測量可能是在開發中的系統上執行，因此不保證可以從一般的系統獲致相同的結果。甚至有部份的測量，是利用插補法而得的估計值，其實際結果可能會有所不同。本書的使用者應根據其特有的環境，驗證出適用的資料。

本書所提及之非 IBM 產品資訊，係一由產品的供應商，或其出版的聲明或其它公開管道取得。IBM 並未測試過這些產品，也無法確認這些非 IBM 產品的執行效能、相容性、或任何對產品的其它主張是否完全無誤。如果您對非 IBM 產品的性能有任何的疑問，請逕向該產品的供應商查詢。

有關 IBM 未來動向的任何陳述，僅代表 IBM 的目標而已，並可能於未事先聲明的情況下有所變動或撤回。

本書中含有日常商業活動所用的資料及報告範例。為了提供完整的說明，這些範例包括個人、公司、廠牌和產品的名稱。這些名稱全屬虛構，若與任何公司的名稱和住址雷同，純屬巧合。

著作權授權：

本書包含原始語言的範例應用程式，用以說明各種作業平台上的程式設計技術。您可以基於研發、使用、銷售或散佈符合作業平台 (用於執行所撰寫的範例程式) 之應用程式設計介面的應用程式等目的，以任何形式複製、修改及散佈這些範例程式，而無需付費給 IBM。但這些範例皆未經過完整的測試。因此，IBM 不會保證或暗示這些程式的穩定性、服務能力或功能。

這些範例程式或是任何衍生著作的每一份拷貝或任何部份，都必須具有下列的著作權聲明：

© (您的公司名稱) (年份)。Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 請在此輸入年份或年數。 All rights reserved.

---

## 商標

下列術語 (以星號 (\*) 標示) 是 IBM 公司在美國、其它國家或兩者的商標。

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

下列術語是其它公司的商標或註冊商標：

Microsoft、Windows、Windows NT 是微軟公司的商標或註冊商標。

Java 以及所有與 Java 有關的商標與標章，以及 Solaris 是 Sun Microsystems, Inc. 在美國、其它國家或兩者的商標。

Tivoli 與 NetView 是 Tivoli Systems Inc. 在美國、其它國家或兩者的商標。

UNIX 是 X/Open Company Limited 在美國、其它國家或兩者的註冊商標，須經該公司授權始可使用。

其它公司、產品或服務名稱 (以兩顆星號 (\*) 標示) 可能是其它公司的商標或服務標誌。



## 索引

索引順序以中文字，英文字，及特殊符號之次序排列。

### 〔二劃〕

二進位整數，說明 4

### 〔三劃〕

大型物件位置，定義 67

小計橫列 62

小數，說明 4

### 〔四劃〕

內部結合 57

分組直欄，定義 29

分割關聯式資料庫，定義 1

日期時間值，說明 4

### 〔五劃〕

主要鍵 52

外來鍵 52

外部表格函數 66

外部純量函數 66

外部結合

說明 57

FULL OUTER 結合 57

LEFT OUTER 結合 57

RIGHT OUTER 結合 57

外部層次查詢，相關 40

外部層次述詞 50

正負號，作為數字屬性 4

母鍵，定義 52

### 〔六劃〕

交叉產品 57

交叉製表橫列 62

交談式 SQL，定義 1

全選 33

次查詢 10, 50

使用 INSERT 陳述式 10

ALL 關鍵字 50

ANY 關鍵字 50

全選，定義 10

共用表格表示式

說明 36

列印 PDF 書籍 99

合併查詢結果 45

合併，查詢 45

多重節點關聯式資料庫，定義 1

字串

可變長度 4

作為資料類型 4

固定長度 4

LOB 67

安裝

Netscape 瀏覽器 104

次查詢

定義 26

次查詢中所用的比較運算子 50

### 〔七劃〕

快速入門 v

系統目錄 68

### 〔八劃〕

使用者定義的函數 66

外部表格函數 66

外部純量函數 66

來源函數 66

定義 66

OLE DB 外部表格函數 66

來源函數 66

函數

內建 27

使用者定義 27

直欄 27

表格 29

函數 (繼續)

純量 27

說明 27

線上分析程序 (OLAP) 62

命令行處理器 1

定位器 67

版次注意事項 99

直欄

定義 3

ASC，升序排序 22

DESC，降序排序 22

直欄函數 27

AVG 27

COUNT 27

MAX 27

MIN 27

表示式 24

表示式，命名 25

表格

主要鍵 52

外來鍵 52

使直欄名稱符合資格 37

函數 29

定義 3

唯一限制 52

唯一鍵 52

基本表格 3

組合資料 (結合) 25

結果表格 3

範例資料庫 71

表格函數

SQLCACHE\_SNAPSHOT 29

表格和概略表的關係 13

表格表示式

說明 35

表格核對限制

定義 51

延緩限制核對 53

說明 53

## 〔九劃〕

- 保留綱目 4
- 建立表格空間精靈 105
- 建立表格精靈 105
- 建立資料庫精靈 105
- 建立範例資料庫 72
- 架構多位置更新精靈 105
- 查詢，連接 47
- 相關
  - 名稱 40
  - 次查詢 38
  - 使用結合的次查詢 41
  - 說明 37
- 相關文件 v
- 相關名稱
  - 直欄名稱的完整參照 37
  - 規則 37
- 相關次查詢
  - 使用時機 40
  - 說明 38
- 相關參考，說明 38
- 述語
  - IS NOT NULL 20
  - IS NULL 20
- 限制
  - 參照限制 9
  - 唯一限制 9
  - 集合運算子 47
- 限定物件 4, 17

## 〔十劃〕

- 值
  - 定義 3
- 效能架構精靈 106
- 書籍 91, 100
- 消除範例資料庫 72
- 特別暫存區 68
  - CURRENT DATE 68
  - CURRENT DEGREE 68
  - CURRENT FUNCTION PATH 68
  - CURRENT PATH 68
  - CURRENT SERVER 68
  - CURRENT TIME 68
  - CURRENT TIMESTAMP 68
  - CURRENT TIMEZONE 68

- 特別暫存區 68 (繼續)
  - USER 68
- 特殊資料類型 65
- 索引精靈 106
- 純量全選
  - 說明 33
- 純量函數 27
  - ABS 28
  - DECIMAL 35
  - HEX 28
  - LENGTH 28
  - SIGN 28
  - YEAR 28
- 除去重複橫列 23

## 〔十一劃〕

- 參照完整性限制
  - 外來鍵 52
  - 母鍵 52
  - 定義 51
  - 說明 52
- 唯一限制 52
  - 定義 51
- 唯一鍵 52
  - 唯一限制 52
- 基本表格 3
- 巢狀表格表示式，說明 35
- 巢狀嵌套相關次查詢 42
- 強制轉型表示式
  - 說明 34
  - SIGN 函數 34
- 授權 ID 4
- 排序橫列 22
- 設定文件伺服器 106
- 連接查詢 47
- 透過概略表修改表格 14
  - WITH CHECK OPTION 14

## 〔十二劃〕

- 備份資料庫精靈 105
- 最新資訊 99
- 復置精靈 106
- 測試是否存在 49
- 測試，存在 49

- 結合
  - 交叉產品 57
  - 定義 25
  - 相關次查詢 41
  - 無結合條件 57
  - 結合條件 57
  - 資料轉換 59
- 結合條件 57
- 結果表格 3
- 結構化查詢語言 (SQL)，定義 1

## 〔十三劃〕

- 搜尋
  - 線上資訊 105, 107
- 搜尋條件 20
- 新增資料庫精靈 105, 106
- 概略表
  - 使直欄名稱符合資格 37
  - 說明 3
  - 優點 3
- 資料庫管理程式 1
- 資料結構
  - 直欄 3
  - 值 3
  - 橫列 3
- 資料轉換
  - 結合條件 59
  - 集合運算子 47
- 資料類型
  - 特殊 65
  - BIGINT 4
  - CHAR 4
  - DATE 4
  - DATETIME 4
  - DECIMAL 4
  - DOUBLE 4
  - FLOAT 4
  - INTEGER 4
  - REAL 4
  - SMALLINT 4
  - TIME 4
  - TIMESTAMP 4
  - VARCHAR 4
- 資料類型強制轉型
  - 說明 33
- 資訊中心 104

運算次序 24, 27

## 〔十四劃〕

圖形字串

可變長度 4

固定長度 4

管理手冊 v

算術運算子 24

精準度，作為數字屬性 4

精靈

完成作業 105

建立表格 105

建立表格空間 105

建立資料庫 105

架構多位置更新 105

效能架構 106

索引 106

備份資料庫 105

復置資料庫 106

新增資料庫 105, 106

綱目

定義 4

語言識別字

書籍 98

遞迴查詢，說明 62

## 〔十五劃〕

數字，說明 4

範例表格 71, 91

範例程式

跨平台 97

HTML 97

範例資料庫 71

建立 72

消除 72

線上分析程序 62

線上資訊

搜尋 107

檢視 103

線上說明 101

複合鍵 51

## 〔十六劃〕

橫列

定義 3

橫列 (繼續)

選取 20

選取列示 19

錯誤訊息

訊息識別字 18

SQLSCODE 18

SQLSTATE 18

## 〔十七劃〕

應用程式開發手冊 v

檢視

線上資訊 103

## 〔十八劃〕

擷取資料 19

## 〔十九劃〕

關聯式資料庫，定義 1

關鍵字

主要 52

外來 52

定義 51

唯一 52

組合 51

## 〔二十劃〕

觸發

定義 51

轉移變數 56

觸發函式

事前觸發 54

事後觸發 54

說明 54

CREATE TRIGGER 54

## A

ADD CONSTRAINT 陳述式 53

ALL，用於查詢中 50

ALTER TABLE 陳述式 53

ANY 關鍵字 50

AS 子句 25

## B

BETWEEN 述詞 48

BIGINT，資料類型 4

BLOB 字串 67

BLOB 資料類型 67

## C

CHAR，資料類型 4

CLOB 字串 67

CLOB 資料類型 67

CL\_SCHED 範例表格 72

CONNECT 陳述式 18

明示 18

隱含 18

CREATE DISTINCT TYPE 65

CREATE FUNCTION 66

CREATE TABLE 陳述式 9

直欄的 NOT NULL/NOT NULL

WITH DEFAULT 值 9

CREATE TRIGGER 54

CREATE VIEW 陳述式 13

WITH CHECK OPTION 13

CUBE 62

小計橫列 62

交叉製表橫列 62

CURRENT DATE 特別暫存區 68

CURRENT FUNCTION PATH 特別暫

存區 68

CURRENT SERVER 特別暫存區 68

CURRENT TIME 特別暫存區 68

CURRENT TIMESTAMP 特別暫存區

68

CURRENT TIMEZONE 特別暫存區

68

## D

DATE，資料類型 4

DATETIME，資料類型 4

DB2 檔案庫

列印 PDF 書籍 99

訂購印刷書籍 100

書籍 91

書籍的語言識別字 98

設定文件伺服器 106

## DB2 檔案庫 (繼續)

- 最新資訊 99
- 結構 91
- 搜尋線上資訊 107
- 資訊中心 104
- 精靈 105
- 線上說明 101
- 檢視線上資訊 103
- DBLOB 字串 67
- DBLOB 資料類型 67
- DECIMAL, 資料類型 4
- DELETE 陳述式 12
- DEPARTMENT 範例表格 73
- DISTINCT 關鍵字 23, 28
- DOUBLE, 資料類型 4

## E

- EMPLOYEE 範例表格 73
- EMP\_ACT 範例表格 76
- EMP\_PHOTO 範例表格 78
- EMP\_RESUME 範例表格 78
- EXCEPT ALL 46
- EXCEPT 運算子 46
  - 相關的使用限制 47
  - 資料類型 47
  - 結果排序 47
- EXISTS 述詞 49

## F

- FLOAT, 資料類型 4
- FROM 子句 19
- FULL OUTER 結合 57

## G

- GROUP BY 24
- GROUP BY 子句
  - 配合 HAVING 子句 30
  - 群組直欄 29

## H

- HAVING 24
- HAVING 子句
  - 說明 30

## HTML

- 範例程式 97

## I

- IN 述詞 48
- INSERT 陳述式 10
  - 直欄的 NOT NULL/NOT NULL WITH DEFAULT 值 10
- INTEGER, 資料類型 4
- INTERSECT ALL 47
- INTERSECT 運算子 47
  - 相關的使用限制 47
  - 資料類型 47
  - 結果排序 47
- IN\_TRAY 範例表格 79

## L

- LEFT OUTER 結合 57
- LIKE 述詞 49
- LOB
  - 字串, 定義 67
  - 定位器, 定義 67

## N

- Netscape 瀏覽器
  - 安裝 104
- NOT BETWEEN 述詞 48
- NOT EXISTS 述詞 49
- NOT IN 述詞 48
- NOT LIKE 述詞 49
- NULL 值 45
  - 刪除直欄值 12
- null 值, 說明 4

## O

- OLAP 函數 62
  - 分割橫列 62
  - 列的次序 62
  - 聚集群組 62
- OLE DB 外部表格函數 66
- ORDER BY 子句 22
  - 集合運算子 47

## ORG 範例表格 79

## P

- PDF 99
- PROJECT 範例表格 79

## R

- REAL, 資料類型 4
- RIGHT OUTER 結合 57
- ROLLUP 62
- ROLL-UP
  - 小計橫列 62

## S

- SALES 範例表格 81
- SELECT 陳述式 19
- SET CONSTRAINTS 陳述式 53
- SET 子句
  - 使用 UPDATE 陳述式 12
- SMALLINT, 資料類型 4
- SmartGuides
  - 精靈 105
- SOME 關鍵字 50
- SQL 中的值 4
- SQL 參考手冊 v
- SQL 程序語言 v
- STAFF 範例表格 82
- STAFFG 範例表格 83

## T

- TIME, 資料類型 4
- TIMESTAMP, 資料類型 4

## U

- UNION ALL 45
- UNION 運算子 45, 46
  - 相關的使用限制 47
  - 資料類型 47
  - 結果排序 46
  - 說明 45
- UPDATE 陳述式 12

USER 特別暫存區 68

## V

VALUES 子句

    使用 INSERT 陳述式 10

VARCHAR，資料類型 4

## W

WHERE 子句 20

    組合 SELECT 陳述式中的資料  
    (結合) 25

    群組的考慮 30

WITH CHECK OPTION 14

WITH 子句 36



---

## 洽詢 IBM

當您有技術上的問題時，請在洽詢「DB2 客戶支援中心」之前，仔細閱讀並執行疑難排解指南所建議的動作。該指南會告訴您必須預先準備的資訊，協助「DB2 客戶支援中心」提供更完善的服務。

若要取得 DB2 Universal Database 產品的相關資訊，或是訂購該系列產品，請洽詢當地 IBM 分公司的業務代表，或是 IBM 授權的軟體經銷商。

如果您住在美國當地，請撥下列一組電話號碼：

- 1-800-237-5511，客戶支援中心
- 1-888-426-4343，取得可用服務選項的資訊

---

## 產品資訊

如果您住在美國當地，請撥下列一組電話號碼：

- 1-800-IBM-CALL (1-800-426-2255) 或 1-800-3IBM-OS2 (1-800-342-6672)，訂購產品或取得一般資訊。
- 1-800-879-2755，訂購出版品。

**<http://www.ibm.com/software/data/>**

DB2 World Wide Web 頁面將提供關於新聞、產品說明、教育課程以及其他種種的現行 DB2 資訊。

**<http://www.ibm.com/software/data/db2/library/>**

DB2 Product and Service Technical Library 可讓您存取常見的問題、修正程式、書籍，以及最新的 DB2 技術資訊。

**註：**這項資訊可能只會以英文表示。

**<http://www.elink.ibm.com/pbl/pbl/>**

International Publications 訂購網站會提供書籍的訂購資訊。

**<http://www.ibm.com/education/certify/>**

IBM 網站中的 Professional Certification Program 會提供包括 DB2 在內之各種 IBM 產品的認證測試資訊。

**<ftp://software.ibm.com>**

以匿名方式登入。您可以在目錄 /ps/products/db2 中找到 DB2 及其它產品的相關示範程式、修訂程式、資訊及工具。

**comp.databases.ibm-db2, bit.listserv.db2-l**

使用者可以利用這些 Internet 新聞群組討論 DB2 產品的使用經驗。

**在 CompuServe 上：GO IBMDB2**

輸入此項指令，即可存 IBM DB2 Family 論壇。所有 DB2 產品均可透過這些論壇取得支援。

關於如何聯絡美國以外的 IBM 的資訊，請參閱 **IBM Software Support Handbook** 的附錄 A。若要存取本文件，請造訪下列網頁：<http://www.ibm.com/support/>，然後選取接近網頁底端的 IBM Software Support Handbook 鏈結。

**註：**在某些國家中，IBM-authorized 授權的代理商應該洽詢它們的產品支援體系，而不是洽詢「IBM 支援中心」。



IBM DB2 Universal Database  
SQL 入門  
版本 7

SC40-0494-00

折疊線

台北市敦化南路一段二號十二樓

臺灣國際商業機器股份有限公司  
中文支援中心 啟



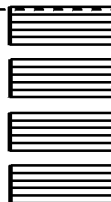
廣告回信
台灣北區郵政管理局 登記
北台字第 0587 號

(免貼郵票)

寄件人 姓名：  
地址：

寄

折疊線



## 讀者意見表

為使本書盡善盡美，本公司極需您寶貴的意見；懇請您使用過後，撥冗填寫下表，惠予指教。

請於下表適當空格內，填入記號（√）；我們會在下一版中，作適當修訂，謝謝您的合作！

評估項目	評估意見	備註
正確性	內容說明與實際程序是否符合	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	參考書目是否正確	<input type="checkbox"/> 是 <input type="checkbox"/> 否
一致性	文句用語及風格，前後是否一致	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	實際畫面訊息與本書所提之畫面訊息是否一致	<input type="checkbox"/> 是 <input type="checkbox"/> 否
完整性	是否遺漏您想知道的項目	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	字句、章節是否有遺漏	<input type="checkbox"/> 是 <input type="checkbox"/> 否
術語使用	術語之使用是否恰當	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	術語之使用，前後是否一致	<input type="checkbox"/> 是 <input type="checkbox"/> 否
可讀性	文句用語是否通順	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	有否不知所云之處	<input type="checkbox"/> 是 <input type="checkbox"/> 否
內容說明	內容說明是否詳盡	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	例題說明是否詳盡	<input type="checkbox"/> 是 <input type="checkbox"/> 否
排版方式	本書的形狀大小，版面安排是否方便使用	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	字體大小，顏色編排，是否有助於閱讀	<input type="checkbox"/> 是 <input type="checkbox"/> 否
目錄索引	目錄內容之編排，是否便於查考	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	索引語錄之排定，是否便於查考	<input type="checkbox"/> 是 <input type="checkbox"/> 否
	※評估意見為“否”者，請於備註欄說明。	

其他：（篇幅不夠時，請另紙說明。）

---

---

---

---

---

---

---

---

---

---

上述改正意見，一經採用，本公司有合法之使用及發佈權利，特此聲明。





Part Number: CT7YHTC

Printed in Singapore

SC40-0494-00



CT7YHTC

