

04

衍生計算欄位



讀者如果是資料科學的初學者，可以略過下述的程式碼；讀者如果不是資料科學的初學者，欲使用 JupyterLab 執行本章節內容，必須先執行下述程式碼載入所需模組與連接資料庫。

```
%LOAD sqlite3 db=../databases/imdb.db timeout=2 shared_cache=true
ATTACH "../databases/covid19.db" AS covid19;
ATTACH "../databases/nba.db" AS nba;
```

4.1 複習一下

使用單獨存在的 `SELECT` 保留字指定希望在查詢結果中顯示的常數，常用的常數類別有四種，分別是整數、浮點數、文字與空值，我們可以使用 `typeof()` 函數顯示常數或者資料表欄位的類別。

```
SELECT 5566 AS an_integer,
       2.718 AS a_real,
       'Hello, World!' AS a_text,
       NULL AS a_null;
```



```
+-----+-----+-----+-----+
| an_integer | a_real | a_text      | a_null |
+-----+-----+-----+-----+
| 5566       | 2.718  | Hello, World! | NULL   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
SELECT TYPEOF(5566) AS typeof_an_integer,
       TYPEOF(2.718) AS typeof_a_real,
       TYPEOF('Hello, World!') AS typeof_a_text,
       TYPEOF(NULL) AS typeof_a_null;
```

```
+-----+-----+-----+-----+
| typeof_an_integer | typeof_a_real | typeof_a_text | typeof_a_null |
+-----+-----+-----+-----+
| integer          | real          | text          | null          |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

在第三章「從資料表選擇」我們將 SQL 敘述歸納為以下幾個部分的組成：

- ◎ 保留字：具有特定功能的指令。
- ◎ 符號：具有特定功能的符號。
- ◎ 常數：由使用者給予的資料。
- ◎ 函數：具有特定邏輯的輸入與輸出對應。

4.2 運算符

其中在「符號」這個部分，我們可以使用屬於符號分類下的運算符（Operators），來對不同資料類別進行運算，包含整數、浮點數、文字與空值，不僅能運算常數，亦能運算資料表的欄。運算符區分為：

1. 數值運算符：應用在資料類別為整數或浮點數的常數或欄位，運算結果為數值。
2. 文字運算符：應用在資料類別為文字의常數或欄位，運算結果為文字。
3. 關係運算符：應用會得到條件判斷結果 0（布林值 False）或 1（布林值 True）兩者其中之一。
4. 邏輯運算符：應用在資料類別屬於 0（布林值 False）或 1（布林值 True）的敘述、常數或欄位，運算結果為布林值。

4.3 數值運算符

針對整數（integer）與帶小數點的浮點數（real）的常數或欄位可以使用數值運算符衍生計算欄位，運算結果為數值，基礎的數值運算符有：

數值運算符	作用描述
+	相加
-	相減
*	相乘
/	相除
%	回傳餘數
()	優先運算

```
SELECT 55 + 66 AS add_two_integers,  
       55 - 66 AS subtract_two_integers,  
       55 * 66 AS multiply_two_integers,  
       7 / 2 AS divide_two_integers,  
       7 % 2 AS modulo;
```



add_two_integers	subtract_two_integers	multiply_two_integers	divide_two_integers	modulo
121	-11	3630	3	1

1 row in set (0.00 sec)

值得注意的是使用 / 相除兩個整數的時候要特別注意所衍生的欄位依然會以整數 (integer) 存在，如果希望調整運算結果為浮點數 (real)，就要將分子或分母至少一者改變為浮點數，最簡單的做法就是乘以 1.0。

```
SELECT 7 / 2 AS divide_two_integers,  
       7 * 1.0 / 2 AS divide_real,  
       7 / 2 * 1.0 AS divide_integer; -- be aware of the priority  
of operation
```

divide_two_integers	divide_real	divide_integer
3	3.5	3.0

1 row in set (0.00 sec)

前述例子在分子乘以 1.0 能順利獲得預期運算結果 3.5，但是在分母乘以 1.0 卻仍然是 3.0，原因在於運算的優先順序，乘除的運算優先順序相同（先乘除後加減），在沒有以 () 指定優先運算時，會先算完 $7 / 2$ 才算 $3 * 1.0$ ，所以若是將分母改為浮點數，必須使用 () 指定優先運算。

```
SELECT 7 / 2 AS divide_two_integers,  
       7 * 1.0 / 2 AS divide_real,  
       7 / (2 * 1.0) AS divide_by_real; -- be aware of the priority  
of operation
```

```

+-----+-----+-----+
| divide_two_integers | divide_real | divide_by_real |
+-----+-----+-----+
| 3                   | 3.5         | 3.5             |
+-----+-----+-----+
1 row in set (0.00 sec)

```

除了常數，我們也能運用數值運算符於資料類別為整數或浮點數的欄位，舉例來說，在 imdb 資料庫的 movies 資料表中 runtime 欄是以「分鐘」記錄電影長度，我們可以透過相除運算符 / 以及回傳餘數運算符 % 衍生運算電影長度為 x 小時 y 分鐘。

```

SELECT runtime,
       runtime / 60 AS hours,
       runtime % 60 AS minutes
FROM movies
LIMIT 5;

```

```

+-----+-----+-----+
| runtime | hours | minutes |
+-----+-----+-----+
| 142     | 2     | 22      |
+-----+-----+-----+
| 175     | 2     | 55      |
+-----+-----+-----+
| 152     | 2     | 32      |
+-----+-----+-----+
| 202     | 3     | 22      |
+-----+-----+-----+
| 96      | 1     | 36      |
+-----+-----+-----+
5 rows in set (0.00 sec)

```



4.4 文字運算符

針對文字 (text) 的常數或欄位可以使用文字運算符衍生計算欄位，運算結果為文字，相較於數值運算符，文字運算符的數量少了許多，僅有 || 兩個垂直線 (可透過 Shift + \ 按出來) 能夠連接文字。

```
SELECT 'Tony' AS first_name,  
       'Stark' AS last_name,  
       'Tony' || ' ' || 'Stark' AS ironman;
```

```
+-----+-----+-----+  
| first_name | last_name | ironman   |  
+-----+-----+-----+  
| Tony      | Stark    | Tony Stark |  
+-----+-----+-----+  
1 row in set (0.00 sec)
```

除了常數，我們也能運用文字運算符於資料類別為文字的字欄位，舉例來說，前一個小節我們透過相除運算符 / 以及回傳餘數運算符 % 衍生運算電影長度為 x 小時 y 分鐘，可以進一步用 || 將兩個衍生欄位再連接為一欄。

```
SELECT runtime,  
       (runtime / 60) || ' hours ' || (runtime % 60) || ' minutes'  
AS hours_minutes  
FROM movies  
LIMIT 5;
```

```
+-----+-----+  
| runtime | hours_minutes |  
+-----+-----+  
| 142     | 2 hours 22 minutes |  
+-----+-----+  
| 175     | 2 hours 55 minutes |  
+-----+-----+  
| 152     | 2 hours 32 minutes |  
+-----+-----+
```

```
| 202      | 3 hours 22 minutes |
+-----+-----+
| 96       | 1 hours 36 minutes |
+-----+-----+
5 rows in set (0.00 sec)
```

由於 `runtime / 60` 與 `runtime % 60` 是整數的資料類別，使用文字運算符的同時產生了隱性類別轉換，如果想明確宣告類別轉換，就使用保留字 `CAST`。

```
SELECT CAST(constants/columns AS datatype)
FROM table;
```

```
SELECT runtime / 60 AS hours,
       TYPEOF(runtime / 60) AS typeof_hours,
       CAST(runtime / 60 AS text) AS hours_text,           --
convert datatype explicitly
       TYPEOF(CAST(runtime / 60 AS text)) AS typeof_hours_text --
check datatype conversion
  FROM movies
 LIMIT 5;
```

```
+-----+-----+-----+-----+
| hours | typeof_hours | hours_text | typeof_hours_text |
+-----+-----+-----+-----+
| 2     | integer      | 2         | text               |
+-----+-----+-----+-----+
| 2     | integer      | 2         | text               |
+-----+-----+-----+-----+
| 2     | integer      | 2         | text               |
+-----+-----+-----+-----+
| 3     | integer      | 3         | text               |
+-----+-----+-----+-----+
| 1     | integer      | 1         | text               |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```



4.5 關係運算符

針對常數或欄位可以使用關係運算符衍生計算欄位，應用後會得到 0（布林值 False）或 1（布林值 True）兩者其中之一，基礎的關係運算符有：

關係運算符	作用描述
=	相等
!=	不相等
>	大於
>=	大於等於
<	小於
<=	小於等於
LIKE	相似
IN	存在於
BETWEEN lower_bound AND upper_bound	大於等於 lower_bound 且小於等於 upper_bound
IS NULL	是否為空值

```
SELECT 55 = 66 AS False,  
       55 != 66 AS True,  
       55 > 55 AS False,  
       55 >= 55 AS True,  
       66 < 66 AS False,  
       66 <= 66 AS True,  
       'Apple' LIKE 'A%' AS True,  
       'Banana' LIKE 'A%' AS False,  
       'A' IN ('A', 'B', 'C') AS True,  
       59 BETWEEN 55 AND 66 AS True,  
       NULL IS NULL AS True;
```



```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| False | True | False | True | False | True | True | False | True | True | True |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 0      | 1    | 0      | 1    | 0      | 1    | 1     | 0     | 1     | 1     | 1     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

```

其中值得注意的是 `LIKE` 關係運算符，作用是文字特徵的判斷，會搭配萬用字元（Wildcards）使用，這裡我們使用了 `'%'` 萬用字元代表「任意文字、長短不拘」這樣的特徵；值得注意的還有 `IS NULL` 關係運算符，作用是判斷空值 `NULL` 是否存在。關係運算符在後續的「篩選觀測值」以及「條件邏輯」的章節中佔有舉足輕重的地位，我們屆時會再複習以及更詳細地解說。

除了常數，我們也能運用關係運算符於資料表的欄位，這時所形成的 0（布林值 `False`）或 1（布林值 `True`）就會隨著列數而產生基於列（Row-wise）的運算。

```

SELECT rating >= 9 AS rating_is_high,
       director == 'Steven Spielberg' AS
is_directed_by_steven_spielberg,
       title IN ('The Shawshank Redemption', 'The Dark Knight') AS
is_specific_movie
FROM movies
LIMIT 10;

```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| rating_is_high | is_directed_by_steven_spielberg | is_specific_movie |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1              | 0                                | 1                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1              | 0                                | 0                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1              | 0                                | 1                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1              | 0                                | 0                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```



```
+-----+-----+-----+
| 1           | 0           | 0           |
+-----+-----+-----+
| 1           | 1           | 0           |
+-----+-----+-----+
| 1           | 0           | 0           |
+-----+-----+-----+
| 0           | 0           | 0           |
+-----+-----+-----+
| 0           | 0           | 0           |
+-----+-----+-----+
| 0           | 0           | 0           |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

4.6 邏輯運算符

針對常數或欄位可以使用邏輯運算符衍生計算欄位，應用在資料類別屬於 0（布林值 False）或 1（布林值 True）的敘述、常數或欄位，運算結果為布林值，基礎的邏輯運算符有：

邏輯運算符	作用描述
AND	和，交集
OR	或，聯集
NOT	反轉布林值，將 0（布林值 False）與 1（布林值 True）互換

```
SELECT 0 AND 0 AS False,
       0 AND 1 AS False,
       1 AND 1 AS True,
       0 OR 0 AS False,
       0 OR 1 AS True,
       1 OR 1 AS True,
       'Apple' NOT LIKE 'A%' AS False,
       'Banana' NOT LIKE 'A%' AS True,
       'A' NOT IN ('A', 'B', 'C') AS False,
```

```
59 NOT BETWEEN 55 AND 66 AS False,
NULL IS NOT NULL AS False;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| False | False | True  | False | True  | True  | False | True  | False | False | False |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 0     | 0     | 1     | 0     | 1     | 1     | 0     | 1     | 0     | 0     | 0     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

除了常數，我們也能運用邏輯運算符於資料表的欄位，這時所形成的 0（布林值 False）或 1（布林值 True）就會隨著列數而產生基於列（Row-wise）的運算。

```
SELECT rating >= 9 AND release_year < 2000 AS
rating_is_high_and_released_before_millennium,
       director == 'Steven Spielberg' OR
       director == 'Christopher Nolan' AS
is_directed_by_steven_spielberg_or_christopher_nolan,
       title == 'The Shawshank Redemption' OR
       title == 'The Dark Knight' AS is_specific_movie
FROM movies
LIMIT 10;
```

關係運算符與邏輯運算符在後續的「從資料表篩選」以及「條件邏輯」的章節中佔有舉足輕重的地位，我們屆時會再複習並更詳細地解說。



重點統整

- ◎ 運算符區分為：
 - 數值運算符：應用在資料類別為整數或浮點數的常數或欄位，運算結果為數值。
 - 文字運算符：應用在資料類別為文字의常數或欄位，運算結果為文字。
 - 關係運算符：應用會得到條件判斷結果 0（布林值 False）或 1（布林值 True）兩者其中之一。
 - 邏輯運算符：應用在資料類別屬於 0（布林值 False）或 1（布林值 True）的敘述、常數或欄位，運算結果為布林值。

- ◎ 這個章節學起來的 SQL 保留字：
 - CAST
 - LIKE
 - IN
 - BETWEEN lower_bound AND upper_bound
 - IS NULL
 - AND
 - OR
 - NOT

- ◎ 將截至目前所學的 SQL 保留字集中在一個敘述中，寫作順序必須遵從標準 SQL 的規定。

```
SELECT DISTINCT columns AS alias
FROM table
LIMIT m;
```

練習題

練習題會涵蓋四個學習資料庫，記得要依據題目的需求，調整編輯器選單的學習資料庫，在自己電腦的 SQLiteStudio 寫出跟預期輸出相同的 SQL 敘述，寫作過程如果卡關了，可以參考附錄 A「練習題參考解答」。

06

從 covid19 資料庫的 `daily_report` 資料表根據 `Confirmed`、`Deaths` 欄位以及下列公式衍生計算欄位 `Fatality_Ratio`，參考下列的預期查詢結果。

$$\text{Fatality_Ratio} = \frac{\text{Deaths}}{\text{Confirmed}}$$

預期輸出 (4011, 3) 的查詢結果。

```
-- 礙於紙本篇幅僅顯示出前五列示意
+-----+-----+-----+
| Confirmed | Deaths | Fatality_Ratio |
+-----+-----+-----+
| 180347    | 7705   | 0.0427231947301591 |
+-----+-----+-----+
| 276101    | 3497   | 0.0126656549595981 |
+-----+-----+-----+
| 265884    | 6875   | 0.0258571407079779 |
+-----+-----+-----+
| 42894     | 153    | 0.00356693243810323 |
+-----+-----+-----+
| 99761     | 1900   | 0.0190455187899079 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```



07

從 `nba` 資料庫的 `players` 資料表依據 `heightMeters`、`weightKilograms` 欄位以及下列公式衍生計算欄位 `bmi`，參考右列的預期查詢結果。

$$\text{BMI} = \frac{\text{weight}_{\text{kg}}}{\text{height}_{\text{m}}^2}$$

預期輸出 (506, 3) 的查詢結果。

-- 礙於紙本篇幅僅顯示出前五列示意

```
+-----+-----+-----+
| heightMeters | weightKilograms | bmi          |
+-----+-----+-----+
| 2.06         | 113.4           | 26.7225940239419 |
+-----+-----+-----+
| 2.01         | 108             | 26.7320115838717 |
+-----+-----+-----+
| 2.03         | 106.6           | 25.8681356014463 |
+-----+-----+-----+
| 2.08         | 120.2           | 27.7829142011834 |
+-----+-----+-----+
| 1.98         | 97.5            | 24.8699112335476 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

08

從 `nba` 資料庫的 `teams` 資料表連接 `confName`、`divName` 欄位後使用 `DISTINCT` 去除重複值，參考下列的預期查詢結果。

預期輸出 (6, 1) 的查詢結果。

```
+-----+
| conf_div          |
+-----+
| East, Southeast  |
+-----+
| East, Atlantic   |
+-----+
| East, Central    |
+-----+
| West, Southwest  |
+-----+
| West, Northwest  |
+-----+
| West, Pacific    |
+-----+
6 rows in set (0.00 sec)
```