

SQL 的五十道練習

條件邏輯

數據交點 | 郭耀仁 yaojenkuo@datainpoint.com

這個章節要學起來的 SQL 保留字

- CASE
- WHEN
- THEN
- ELSE
- END

以 CASE 衍生計算欄位

條件除了能夠運用在 **WHERE** 篩選資料，另外一個常見的應用場景是使用 **CASE** 衍生計算欄位

```
SELECT CASE WHEN condition_1 THEN result_1
          WHEN condition_2 THEN result_2
          ...
          ELSE result_else
END AS alias;
```

使用布林 (Boolean) 表示電影是否在千禧年之後上映

In [5]:

```
SELECT title,  
       release_year,  
       CASE WHEN release_year >= 2000 THEN 1  
            ELSE 0 END AS after_millennium  
FROM movies  
LIMIT 5;
```

Out[5]:

title	release_year	after_millennium
The Shawshank Redemption	1994	0
The Godfather	1972	0
The Godfather: Part II	1974	0
The Dark Knight	2008	1
12 Angry Men	1957	0

5 rows in set (0.00 sec)

在 **nba** 資料庫中球員鋒衛位置的原始分類比較多元

In [6]:

```
SELECT DISTINCT pos
FROM players;
```

Out[6]:

pos
F
C-F
G-F
G
F-G
C
F-C

7 rows in set (0.00 sec)

使用文字將球員的鋒衛種類分為三個位置

In [7]:

```
SELECT DISTINCT CASE WHEN pos IN ('C', 'C-F') THEN 'Center'
                    WHEN pos IN ('G', 'G-F') THEN 'Guard'
                    ELSE 'Forward' END AS pos_new,
pos
FROM players
ORDER BY pos;
```

Out[7]:

pos_new	pos
Center	C
Center	C-F
Forward	F
Forward	F-C
Forward	F-G
Guard	G
Guard	G-F

7 rows in set (0.00 sec)

如果在某個觀測值發生 `condition_1` 與 `condition_2` 都判斷為真的情況下，會以 `result_1` 表示

```
SELECT CASE WHEN condition_1 THEN result_1
           WHEN condition_2 THEN result_2
           ...
           ELSE result_else
END AS alias;
```


當條件沒有互斥的時候，撰寫 **CASE** 要特別注意順序

`heightMeters > 1.90` 與 `heightMeters > 2.10` 這兩個條件交集。

In [8]:

```
SELECT DISTINCT CASE WHEN heightMeters > 2.10 THEN 'Tall' -- 先判斷 Tall 結果和預期相符
                    WHEN heightMeters > 1.90 THEN 'Medium'
                    ELSE 'Short' END AS height_category
FROM players;
```

Out[8]:

<u>height_category</u>
Medium
Short
Tall

3 rows in set (0.00 sec)

In [9]: `SELECT DISTINCT CASE WHEN heightMeters > 1.90 THEN 'Medium' -- 先判斷 Medium 結果不如預期
WHEN heightMeters > 2.10 THEN 'Tall'
ELSE 'Short' END AS height_category
FROM players;`

Out[9]:

<u>height_category</u>
Medium
Short

2 rows in set (0.00 sec)

如果不想特別注意順序，可以將條件設計為互斥

```
In [10]: SELECT DISTINCT CASE WHEN heightMeters > 1.90 AND  
                                heightMeters <= 2.10 THEN 'Medium'  
                                WHEN heightMeters > 2.10 THEN 'Tall'  
                                ELSE 'Short' END AS height_category  
FROM players;
```

```
Out[10]:
```

<u>height_category</u>
Medium
Short
Tall

3 rows in set (0.00 sec)

以 CASE 衍生計算欄位排序

CASE 除了搭配 SELECT 使用，亦能夠搭配 ORDER BY 使用

```
SELECT CASE WHEN condition_1 THEN result_1
          WHEN condition_2 THEN result_2
          ...
          ELSE result_else END AS alias
FROM table_name
ORDER BY alias;
```

In [11]:

```
SELECT firstName,
       lastName,
       CASE WHEN pos IN ('C', 'C-F') THEN 'Center'
            WHEN pos IN ('G', 'G-F') THEN 'Guard'
            ELSE 'Forward' END AS pos_new
FROM players
ORDER BY pos_new
LIMIT 5;
```

Out[11]:

firstName	lastName	pos_new
Dwight	Howard	Center
LaMarcus	Aldridge	Center
Al	Horford	Center
Marc	Gasol	Center
Brook	Lopez	Center

5 rows in set (0.00 sec)

也能夠直接在 **ORDER BY** 加入 **CASE**

```
SELECT column_names
  FROM table_name
 ORDER BY CASE WHEN condition_1 THEN result_1
              WHEN condition_2 THEN result_2
              ...
              ELSE result_else END;
```

In [12]:

```
SELECT firstName,  
       lastName  
FROM players  
ORDER BY CASE WHEN pos IN ('C', 'C-F') THEN 'Center'  
              WHEN pos IN ('G', 'G-F') THEN 'Guard'  
              ELSE 'Forward' END  
LIMIT 5;
```

Out[12]:

firstName	lastName
Dwight	Howard
LaMarcus	Aldridge
Al	Horford
Marc	Gasol
Brook	Lopez

5 rows in set (0.00 sec)

以 CASE 衍生計算欄位篩選

CASE 除了搭配 SELECT 、 ORDER BY 使用，亦能夠
搭配 WHERE 使用

In [13]:

```
SELECT DISTINCT CASE WHEN pos IN ('C', 'C-F') THEN 'Center'
                    WHEN pos IN ('G', 'G-F') THEN 'Guard'
                    ELSE 'Forward' END AS pos_new,
pos
FROM players
WHERE pos_new = 'Center';
```

Out[13]:

pos_new	pos
Center	C-F
Center	C

2 rows in set (0.00 sec)

重點統整

- 條件除了能夠運用在 `WHERE` 篩選資料，另一個常見的應用場景是使用 `CASE` 衍生計算欄位。
- `CASE` 可以搭配 `SELECT`、`ORDER BY` 與 `WHERE` 使用，其中在搭配 `ORDER BY` 使用時可以直接加入。

```
/*
```

```
截到目前學起來的 SQL 有哪些？
```

```
SQL 寫作順序必須遵從標準 SQL 的規定。
```

```
*/
```

```
SELECT column_names      -- 選擇哪些欄位  
  FROM table_name        -- 從哪個資料庫的資料表  
  WHERE conditions       -- 篩選哪些觀測值  
  ORDER BY column_names  -- 指定依照哪個變數排序  
  LIMIT m;              -- 查詢結果顯示前 m 列就好
```

