

SQL 的五十道練習

從資料表選擇

數據交點 | 郭耀仁 yaojenkuo@datainpoint.com

這個章節要學起來的 SQL 保留字

- SELECT
- FROM
- LIMIT
- AS
- DISTINCT

以 `SELECT` 與 `FROM` 從資料表選擇

回憶一下「哈囉資料庫」SQL

In [5]:

```
SELECT *  
  FROM movies  
 LIMIT 5;
```

Out[5]:

id	title	release_year	rating	director	runtime
1	The Shawshank Redemption	1994	9.3	Frank Darabont	142
2	The Godfather	1972	9.2	Francis Ford Coppola	175
3	The Godfather: Part II	1974	9.0	Francis Ford Coppola	202
4	The Dark Knight	2008	9.0	Christopher Nolan	152
5	12 Angry Men	1957	9.0	Sidney Lumet	96

SELECT、FROM 與 LIMIT

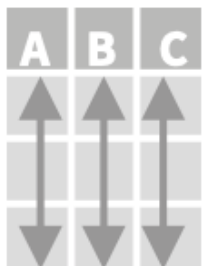
- **SELECT** 是「選擇」欄位。
- **FROM** 是指定「從」哪個資料表查詢資料。
- **LIMIT** 是讓資料查詢結果顯示指定的前幾列。
- ***** 表示「所有」。
- **;** 表示一段 SQL 的結束。

儲存在資料庫中的資料表是有兩個維度的資料結構

- 第一個維度稱為觀測值 (Observations) ，有時亦稱為列 (Rows) 。
- 第二個維度稱為變數 (Variables) ，有時亦稱為欄 (Columns) 。

如何描述一個資料表的外觀

我們習慣以 (m, n) 或者 $m \times n$ 來描述一個具有 m 列觀測值、 n 欄變數的資料表 (或者資料查詢結果) 。



&



Each **variable** is in
its own **column**

Each **observation**, or
case, is in its own **row**

Source: <https://www.rstudio.com/resources/cheatsheets>

「哈囉資料庫」SQL 資料查詢結果的外觀為 (5, 6)
或者 5 x 6

In [6]:

```
SELECT *  
  FROM movies  
 LIMIT 5;
```

Out[6]:

id	title	release_year	rating	director	runtime
1	The Shawshank Redemption	1994	9.3	Frank Darabont	142
2	The Godfather	1972	9.2	Francis Ford Coppola	175
3	The Godfather: Part II	1974	9.0	Francis Ford Coppola	202
4	The Dark Knight	2008	9.0	Christopher Nolan	152
5	12 Angry Men	1957	9.0	Sidney Lumet	96

在 **SELECT** 後加入欄位的名稱表示查詢結果只選擇指定欄位



In [7]:

```
SELECT title
  FROM movies
 LIMIT 5;
```

Out[7]:

title
The Shawshank Redemption
The Godfather
The Godfather: Part II
The Dark Knight
12 Angry Men

若想指定多個欄位，就用逗號，將多個欄位名稱隔開

In [8]:

```
SELECT title,  
       release_year,  
       rating  
FROM movies  
LIMIT 5;
```

Out[8]:

title	release_year	rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0
The Dark Knight	2008	9.0
12 Angry Men	1957	9.0

注意事項

- 保留字大小寫不會影響執行結果的對錯。
- 寫作一段 SQL 可以單行或者換行並搭配縮排。

SQL 中的保留字大小寫不會影響執行結果的對錯，但是我推薦採用「全大寫」的風格

In [9]:

```
SELECT title,  
       release_year,  
       rating  
FROM movies  
LIMIT 5;
```

Out[9]:

title	release_year	rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0
The Dark Knight	2008	9.0
12 Angry Men	1957	9.0

寫作一段 SQL 可以單行或者換行，但是我推薦採用「換行並靠右對齊縮排」的風格

In [10]:

```
SELECT title,  
       release_year,  
       rating  
FROM movies  
LIMIT 5;
```

Out[10]:

title	release_year	rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0
The Dark Knight	2008	9.0
12 Angry Men	1957	9.0

之所以推薦特定的寫作風格，是因為我遵從了 Simon Holywell 所寫的 SQL 風格指南

[SQL Style Guide by Simon Holywell](#)

其他 SQL 風格指南

- [SQL Style Guide by GitLab](#)
- [SQL Style Guide by Mozilla](#)
- ...etc.

在 SQL 中添加註解

寫作 SQL 時我們會在段落中添加註解，有兩種常用的註解形式

1. 單行註解

```
-- comments
```

1. 多行註解

```
/*  
...comments  
...comments  
...comments  
*/
```

使用兩個減號 -- 添加單行註解

In [11]:

```
-- 如何添加單行註解
SELECT title,           -- 用逗號將欄位名稱隔開
       release_year,    -- 用逗號將欄位名稱隔開
       rating           -- 最後一個欄位名稱後面不要逗號
FROM movies             -- 存於 imdb.db 中的 movies 資料表
LIMIT 3;               -- 顯示前 3 列並且用分號;表示一段 SQL 的結束
```

Out[11]:

title	release_year	rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0

使用 `/*` 與 `*/` 添加多行註解

In [12]:

```
/*  
如何添加多行註解  
用逗號將欄位名稱隔開  
最後一個欄位名稱後面不要逗號  
存於 imdb.db 中的 movies 資料表  
顯示前 3 列並且用分號;表示一段 SQL 的結束  
*/  
SELECT title,  
       release_year,  
       rating  
FROM movies  
LIMIT 3;
```

Out[12]:

title	release_year	rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0

替查詢結果的欄位取別名

使用 **AS** 為查詢結果中的變數取別名

In [13]:

```
SELECT title AS movie_title,  
       release_year AS movie_year,  
       rating AS imdb_rating  
FROM movies  
LIMIT 5;
```

Out[13]:

movie_title	movie_year	imdb_rating
The Shawshank Redemption	1994	9.3
The Godfather	1972	9.2
The Godfather: Part II	1974	9.0
The Dark Knight	2008	9.0
12 Angry Men	1957	9.0

查詢結果僅顯示獨一值

使用 **DISTINCT** 省略查詢結果中重複的資料

In [14]:

```
SELECT confName  
FROM teams; -- 存於 nba.db 中的 teams 資料表
```

Out[14]:

confName
East
East
East
West
East
West
West
West
West
West
West
West
East
East
West
East
East
East
East
East
West
West
West
West
West
East

West

West

East

East

East

In [15]: `SELECT DISTINCT confName AS distinct_conf_name
FROM teams; -- 存於 nba.db 中的 teams 資料表`

Out[15]:

distinct_conf_name
East
West

重點統整

- `SELECT` 與 `FROM` 是從資料表選擇變數的保留字。
- `LIMIT` 是讓資料查詢結果顯示指定前幾列的保留字。
- 寫作 SQL 時應該遵從指定的風格指南。

重點統整（續）

- 寫作 SQL 時會添加單行註解或者多行註解。
- 使用 `AS` 為查詢結果的變數取別名。
- 使用 `DISTINCT` 省略查詢結果中重複的資料。

```
/*  
截至目前學起來的 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  
    LIMIT m;          -- 查詢結果顯示前 m 列就好
```

