專屬指令

朱克剛

Pivot – 樞紐轉換

	new_field	1	2	3	4
1	SumFee	1750	4200	*NULL*	200

- 計算每季帳單金額總和

```
select *
from (
    select 'SumFee' as 'new_field', month(dd) / 3 + 1 as quarter, fee
    from Bill
    where year(dd) = 2019
) as temp
pivot (
    sum(fee)
    for quarter in ([1], [2], [3], [4])
) as PivotTable
```

y軸資料

x軸資料

運算資料

這裡是欄位內容

沒有 Pivot 時的替代方案

- 使用 CASE WHEN

```
select
    sum(case when (month(dd) / 3 + 1) = 1 then fee else 0 end) as '第一季',
    sum(case when (month(dd) / 3 + 1) = 2 then fee else 0 end) as '第二季',
    sum(case when (month(dd) / 3 + 1) = 3 then fee else 0 end) as '第三季',
    sum(case when (month(dd) / 3 + 1) = 4 then fee else 0 end) as '第四季'
from Bill
where year(dd) = 2019
```

使用 union all 的另一種作法

```
select quarter, sum(value) as sum_fee from (
    select month(dd) /3 + 1 as quarter, sum(fee) as value
    from Bill
    where year(dd) = 2019
    group by month(dd) / 3 + 1
    union all select 1, 0
    union all select 2, 0
    union all select 3, 0
    union all select 4, 0
) as temp
group by quarter
```

	quarter	sum_fee
1	1	1750
2	2	4200
3	3	0
4	4	200

練習

- 把上一章投影片 union all 結果轉成如下表

	year	1	2	3	4
1	2023	1750	4200	0	200

常見應用

- 建立學生作業繳交紀錄表

*sid	學號
*title	作業名稱
upload_date	繳交時間

- 產生一張哪位學生繳交了哪些作業的二維表單

	sid	hw1	hw2	hw3
1	s01	1	1	0
2	s02	0	1	0
3	s03	1	1	1
4	s04	0	0	1
5	s05	0	1	1
6	s06	0	1	0

Unpivot

- 與 pivot 相反。將下列指令建立成 View（不建也可以）

	year	鉛筆	原子筆	橡皮擦
1	2022	180	130	260
2	2023	200	180	213

```
create view vw_sold_analysis as
  select 2022 as [year], 180 as [鉛筆], 130 as [原子筆], 260 as [橡皮擦]
  union all
  select 2023 as [year], 200 as [鉛筆], 180 as [原子筆], 213 as [橡皮擦]
```

- 使用 unpivot

	year	productName	soldNumber
1	2022	鉛筆	180
2	2022	原子筆	130
3	2022	橡皮擦	260
4	2023	鉛筆	200
5	2023	原子筆	180
6	2023	橡皮擦	213

```
select [year], productName, soldNumber from (
    select * from vw_sold_analysis
) as temp
unpivot (
    soldNumber for productName in ([鉛筆], [原子筆], [橡皮擦])
) as unp
```

Apply

- cross apply = inner join
- outer apply = left outer join

```
select * from UserInfo
outer apply (
    select * from Live where UserInfo.uid = Live.uid) as x
outer apply (
    select * from House where x.hid = House.hid
) as y
order by UserInfo.uid
```

遵守第一正規化

- 若資料已違反第一正規化

uid	courses
A01	數學,英文,化學
A02	數學,物理

uid	courses
A01	數學
A01	英文
A01	化學
A02	數學
A02	物理

- 查詢 A01 修了哪些課

```
select uid, a.value
from CourseSelection
    cross apply string_split(courses, ',') a
where uid = 'A01'
```

此函數傳回表

Binary Data

- 將圖片存入使用者的 headphoto 欄位（型態為：varbinary）
- 字串前的 N 表示該字串為 nvarchar 而不是 varchar

```
update UserInfo set headphoto = (
    select * from OPENROWSET(BULK N'/tmp/sonia.jpg', SINGLE_BLOB) as tmp
)
where uid = 'A01'
```

從檔案讀取

官方建議只使用這個，因其支援所有 Windows 編碼轉換

輸出 JSON

- 將 UserInfo 的查詢結果以 JSON 格式輸出，很適合用在 Web API 設計

```
select * from UserInfo
for json path, include_null_values
```

- 注意 null 處理，Web API 中不應該出現 null

OPENJSON – 字典

- 當 JSON 字串僅是字典型態時

```
select * from openjson(N'
    {
        "name": "Tom",
        "age": 30
    }
')
```

	key	value	type
1	name	Tom	1
2	age	30	2

- 指定欄位

指定欄位

- 使用 with 來指定欄位

```
select * from openjson(N'
    {
        "name": "Tom",
        "age": 30
    }
') with (
    name nvarchar(50) '$.name',
    age int '$.age'
)
```

	name	age
1	Tom	30

OPENJSON – 陣列

- JSON 字串內容為陣列時

```
select * from openjson(N'[
    {
        "name": "David",
        "age": 18
    },
    {
        "name": "Betty",
        "age": 17
    }
]')
with (
    name nvarchar(50)   '$.name',
    age int             '$.age'
)
```

	name	age
1	David	18
2	Betty	17

實際試試 - AQI

	sitename	county	aqi	status	publishtime
1	基隆	基隆市	41	良好	2024-01-14 19:00:00.000
2	汐止	新北市	37	良好	2024-01-14 19:00:00.000
3	萬里	新北市	38	良好	2024-01-14 19:00:00.000
4	新店	新北市	42	良好	2024-01-14 19:00:00.000
5	土城	新北市	46	良好	2024-01-14 19:00:00.000

- 將產生的資料存到實際資料表去

```
declare @json nvarchar(max)
set @json = N'{ ... }'

select *
from openjson(@json, '$.records')
with (
    sitename nvarchar(50) '$.sitename',
    county nvarchar(50) '$.county',
    aqi int '$.aqi',
    status nvarchar(50) '$.status',
    publishtime datetime '$.publishtime'
)
```