前言:

webcs 的cs\_employee 基本上會指向cs\_company , cs\_department

但指向的cs\_department如果要知道最上面是甚麼廠 則必須cs\_department 要再靠parentid 自己lookup自己上去

假設我問到一個人

公司代碼 1000 部門代碼 GF2712 鼓山廠嘉義分廠業務股

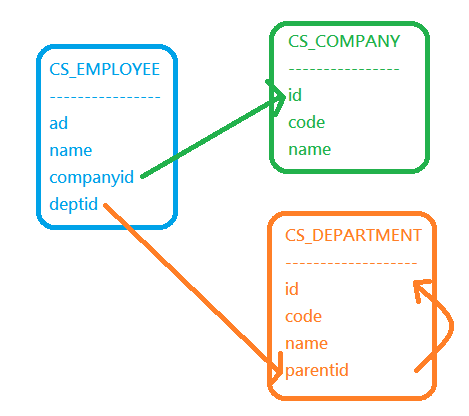
公司代碼 1000 部門代碼 GF2710 鼓山廠嘉義分廠(往上問一層)

公司代碼 1000 部門代碼 GF2700 鼓山廠(最上層)

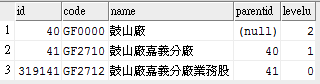
* 遞迴基本觀念

1. 可以向上或向下
2. 可以使用最上層 或 目標當成base

目的: 如何用sql 就可以找出最上層,當然自己用程式處理也可以

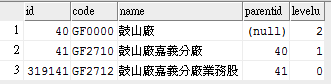


Cs\_department關係sample如下圖



以下分別就ms\_sql / oracle 如何知道最上層廠別

* MS SQL SERVER (使用ms sql CTE 語法)
* 解法1 列出某個部門代碼以上的清單:



--sql start--

with unitLevel as

(

select

id

,code

,name

,parentid

,0 as levelu

from

cs\_department

where

--id=319141

--code=’GF2700’

code='GF2712'

union all

select

x.id

,x.code

,x.name

,x.parentid

,levelu +1 as levelu

from

cs\_department x join unitLevel y on

--x.parentid = y.id --down use

y.parentid = x.id --up use

)

select \* from unitLevel

order by levelu desc

--sql end—

* 解法2列出某個部門代碼最上面的資料:



--sql start--

with unitLevel as

(

select

id

,code

,name

,parentid

,0 as levelu

from

cs\_department

where id=319141

--code=’GF2712’

union all

select

x.id

,x.code

,x.name

,x.parentid

,levelu +1 as levelu

from

cs\_department x join unitLevel y on

x.id = y.parentId

)

select top 1 \* from unitLevel

order by levelu desc

--sql end—

* 解法3列出某個部門以下的資料

--sql start--

with unitLevel as

(

select

id

,code

,name

,parentid

,0 as levelu

from

cs\_department

where

--id=319141

--code=’GF2712’

code='GF2712'

union all

select

x.id

,x.code

,x.name

,x.parentid

,levelu +1 as levelu

from

cs\_department x join unitLevel y on

x.parentid = y.id --down use

--y.parentid = x.id --up use

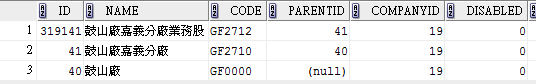
)

select \* from unitLevel

order by levelu desc

--sql end--

* Oracle
* 解法1列出某部門上層所有:



--sql start—

select \* from cs\_department

START WITH code = 'GF2712'

CONNECT BY PRIOR parentid = id

--sql end--

* 解法2列出最上層code



--sql start--

select \* from cs\_department

where parentid is null

START WITH code = 'GF2712'

CONNECT BY PRIOR parentid = id

--sql end—

* 解法3 列出想要的某一層

--sql start—

select \* from

(

select z.\*,rownum as pos from

(

select id,name,code,parentid from cs\_department

START WITH code = 'GF2712'

CONNECT BY PRIOR parentid = id

) z

) x

where x.pos = 3

--sql end--

* 解法4 某個部門向下loop

--sql start—

select id,name,code,parentid from cs\_department

START WITH code = 'GF0000'

CONNECT BY parentid = PRIOR id

--sql end--

* 延伸研究 一個sql 就搞定

Company\_code,dep\_code,factory\_code

* Oracle版本(subquery)



--sql start—

select

x.code as c\_code

,x.name as c\_name

,y.code as d\_code

,y.name as d\_name

,

(

select code from cs\_department

where parentid is null

START WITH code = y.code

CONNECT BY PRIOR parentid = id

) as top\_level\_code

,

(

select name from cs\_department

where parentid is null

START WITH code = y.code

CONNECT BY PRIOR parentid = id

) as top\_level\_name

,z.code

,z.name

from

cs\_company x

,cs\_department y

,cs\_employee z

where

z.companyid = x.id

and z.deptid = y.id

--and z.name = '高康漢'

and z.code = '1870200'

--sql end--

* Ms sql 版本
* 補充說明

1. ms sql cte 語法 可以定位出到底是第幾層

如果想知道 該部門是第幾層單位

Oracle這邊還沒研究出來

1. 遞迴語法 基本上有點暴力 (個人認為)

因此沒有考慮 直接寫

Cs\_employee,cs\_company,cs\_department

一次達陣的寫法

(不過目前我還沒寫出一次達陣的語法就是了)

比較建議先問出employee對應的department\_code

再去問最上層的department\_code是甚麼

1. Webcs目前設計如果拿最上層department

可能子廠會忽略 變成母廠

因此如果care子廠

應該要遞迴一層 就問一下mapping table

是否有存在的廠別

有就停止 才能停在子廠的對應