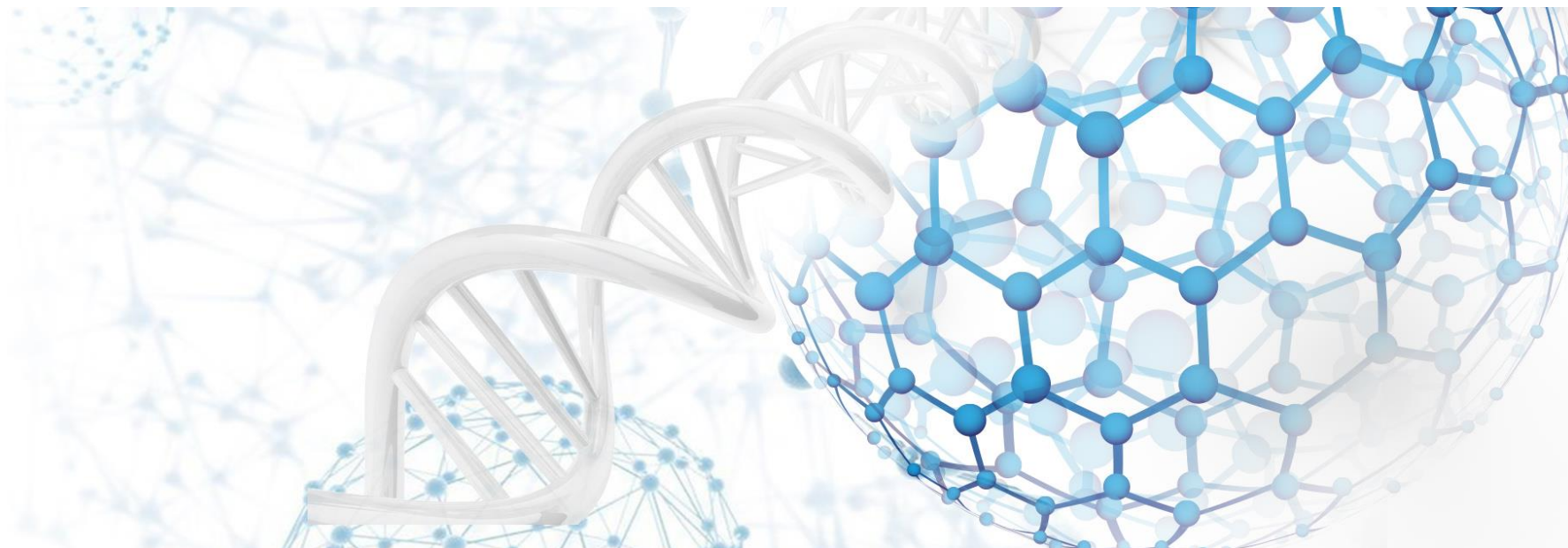


SQL資料分析



劉豐泰
francisft@scu.edu.tw

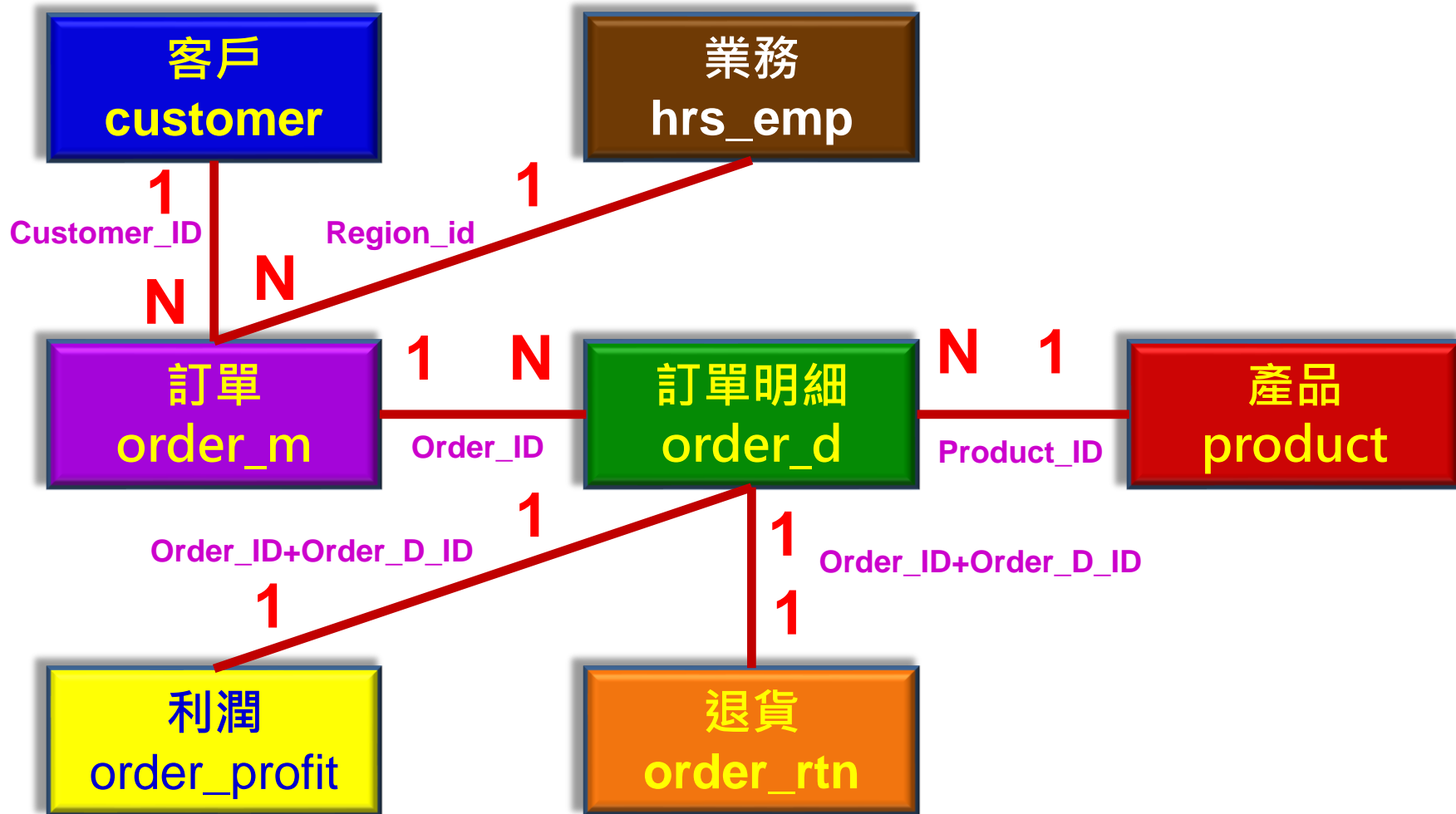


東吳大學
Soochow University

SQL指令

- create table 、 alter table 、 drop table
- create index
- insert into 、 update 、 delete from
- use 、 select 、 from
- order by 、 asc 、 desc
- where 、 and 、 or
- between " and " 、 like
- case when 、 getdate()
- top ?? 、 substring()
- group by 、 max 、 min 、 avg 、 sum 、 count
- inner join 、 left join 、 right join 、 full join 、 union all

資料定義語言(DDL)



資料定義語言(DDL)

customer

Column	Description
Customer_ID	客戶編號
Customer_Name	客戶名稱
Segment	分類

hrs_emp

Column	Description
emp_no	員工編號
emp_name	員工名稱
Region_id	區域ID

product

Column	Description
Product_ID	產品編號
Category_id	大分類ID
Sub_Category_id	中分類ID
Product_Name	產品名稱
Price	單價

order_m

Column	Description
Order_ID	訂單編號
Order_Date	訂單日期
Ship_Date	送貨日期
Ship_Mode	送貨模式
Customer_ID	客戶編號
Region_id	區域ID
State_id	國家ID
Country_id	州/省ID
city_id	城市ID

order_d

Column	Description
Order_ID	訂單編號
Order_D_ID	訂單明細編號
Product_ID	產品編號
Sales	銷售總額
Quantity	數量
Discount	折扣
price	單價

order_rtn

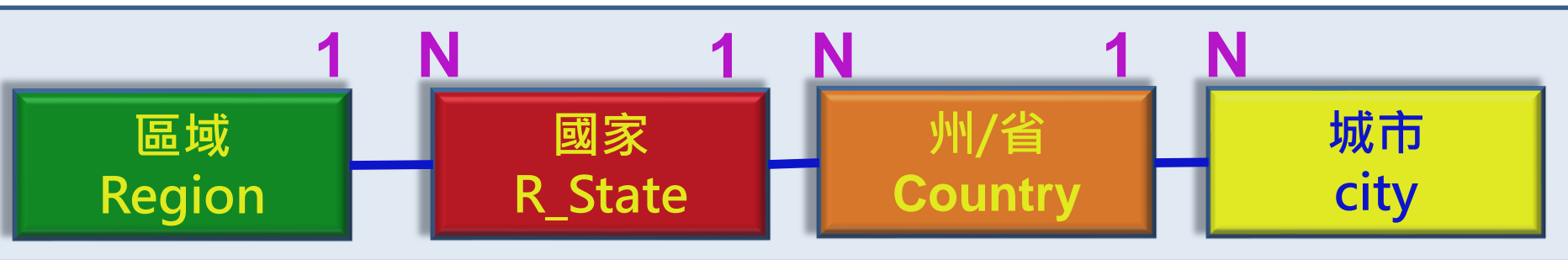
Column	Description
Order_ID	訂單編號
Order_D_ID	訂單明細編號
Customer_ID	客戶編號
rtn_Date	退貨日期
Product_ID	產品編號
rtn_amount	退貨金額
Quantity	數量

order_profit

Column	Description
Order_ID	訂單編號
Order_D_ID	訂單明細編號
Product_ID	產品編號
Quantity	數量
profit	利潤

資料定義語言(DDL)

銷售區域(複合屬性)



Region	
Column	Description
Region_id	區域代碼
Region_name	區域名稱

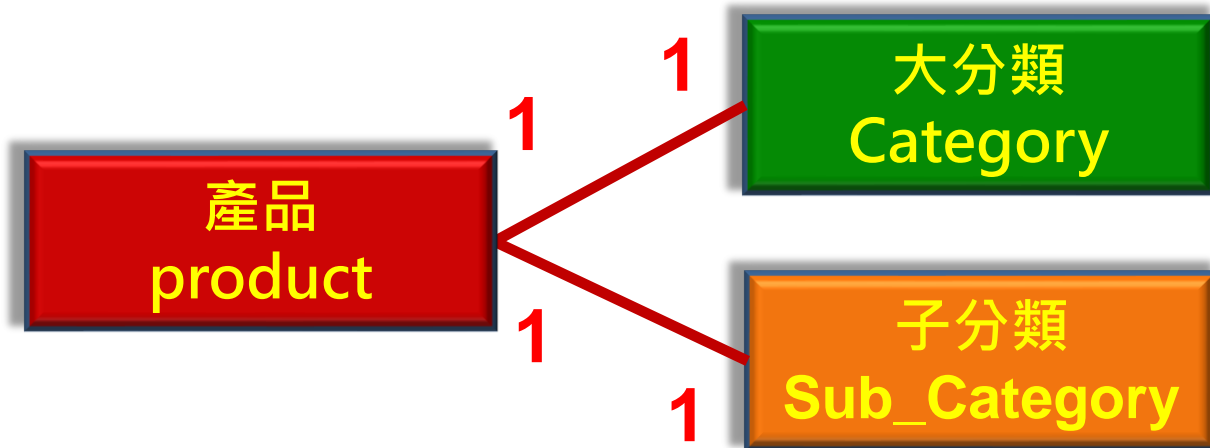
R_State	
Column	Description
Region_id	區域代碼
State_id	國別代碼
State_name	國別名稱

Country	
Column	Description
Region_id	區域代碼
State_id	國別代碼
Country_id	州/省代碼
Country_name	州/省名稱

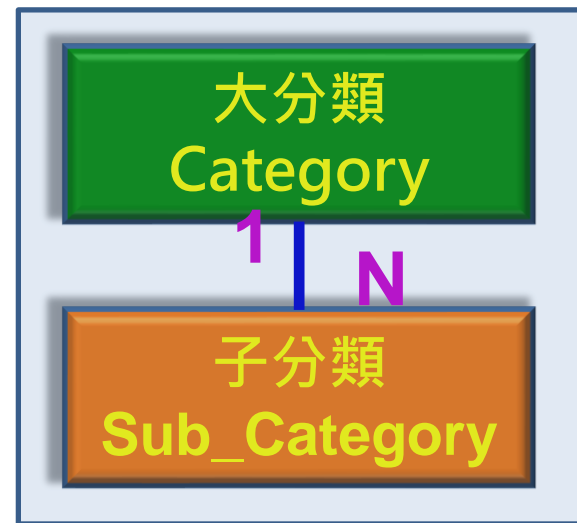
city	
Column	Description
Region_id	區域代碼
State_id	國別代碼
Country_id	州/省代碼
city_id	城市代碼
city_name	城市名稱

資料定義語言(DDL)

產品分類(複合屬性)



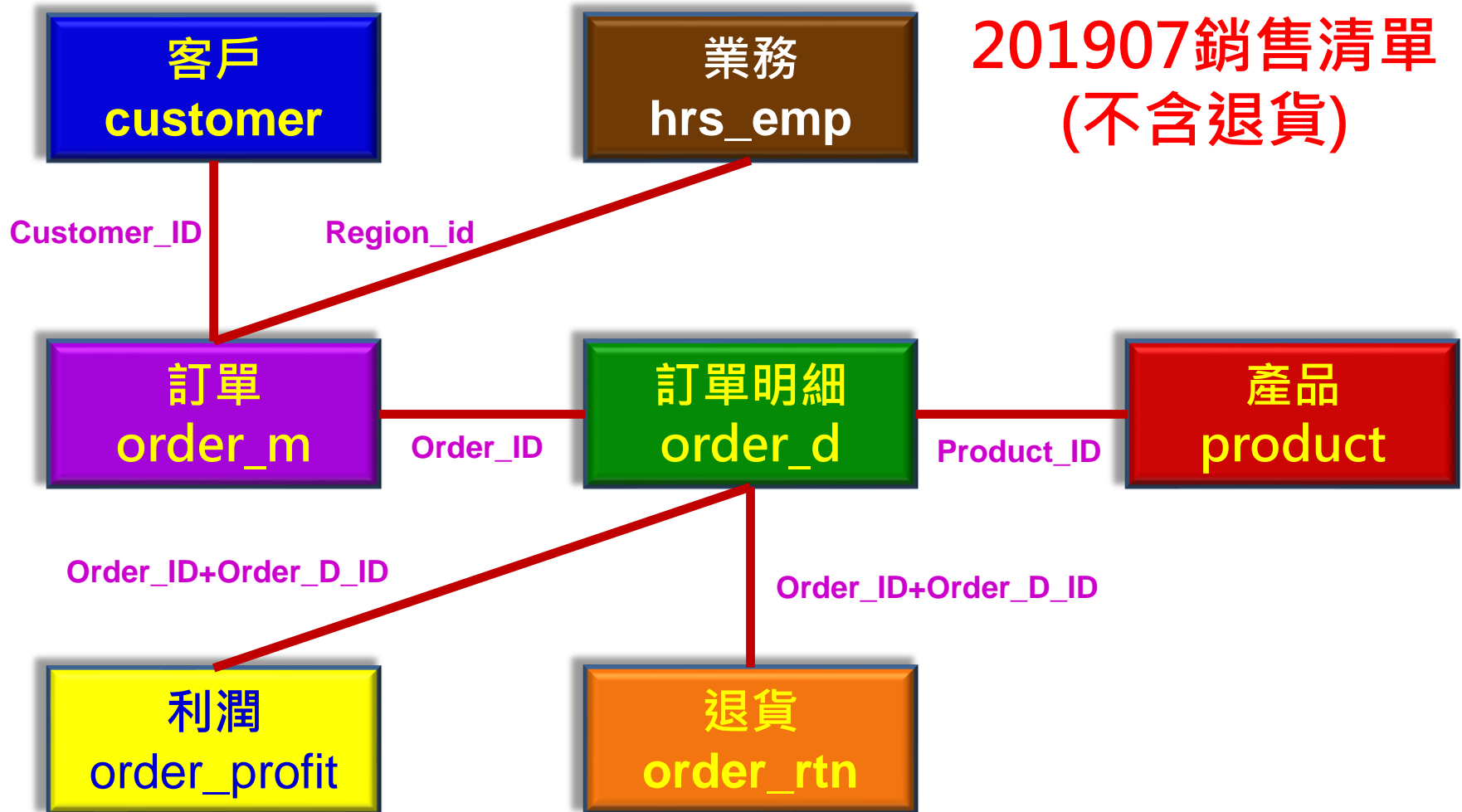
Category	
Column	Description
Category_id	大分類代碼
Category	大分類名稱



Sub_Category	
Column	Description
Category_id	大分類代碼
Sub_Category_id	中分類代碼
Sub_Category	中分類名稱

SQL指令與語法

訂單日期	客戶編號	客戶名稱	產品編號	產品名稱	數量	金額小計
------	------	------	------	------	----	------



SQL指令與語法

訂單日期	客戶編號	客戶名稱	產品編號	產品名稱	數量	金額小計
------	------	------	------	------	----	------

```
select a.order_date,a.Customer_ID,b.Customer_Name
      ,d.Product_ID,d.Product_Name
      ,c.Quantity,c.Sales
from order_m a
inner join customer b on a.customer_id=b.Customer_ID
inner join order_d c on a.order_id=c.Order_ID
inner join product d on c.Product_ID=d.Product_ID
where a.Order_Date between '2019/07/01' and '2019/07/31'
order by a.customer_id,a.order_date
```

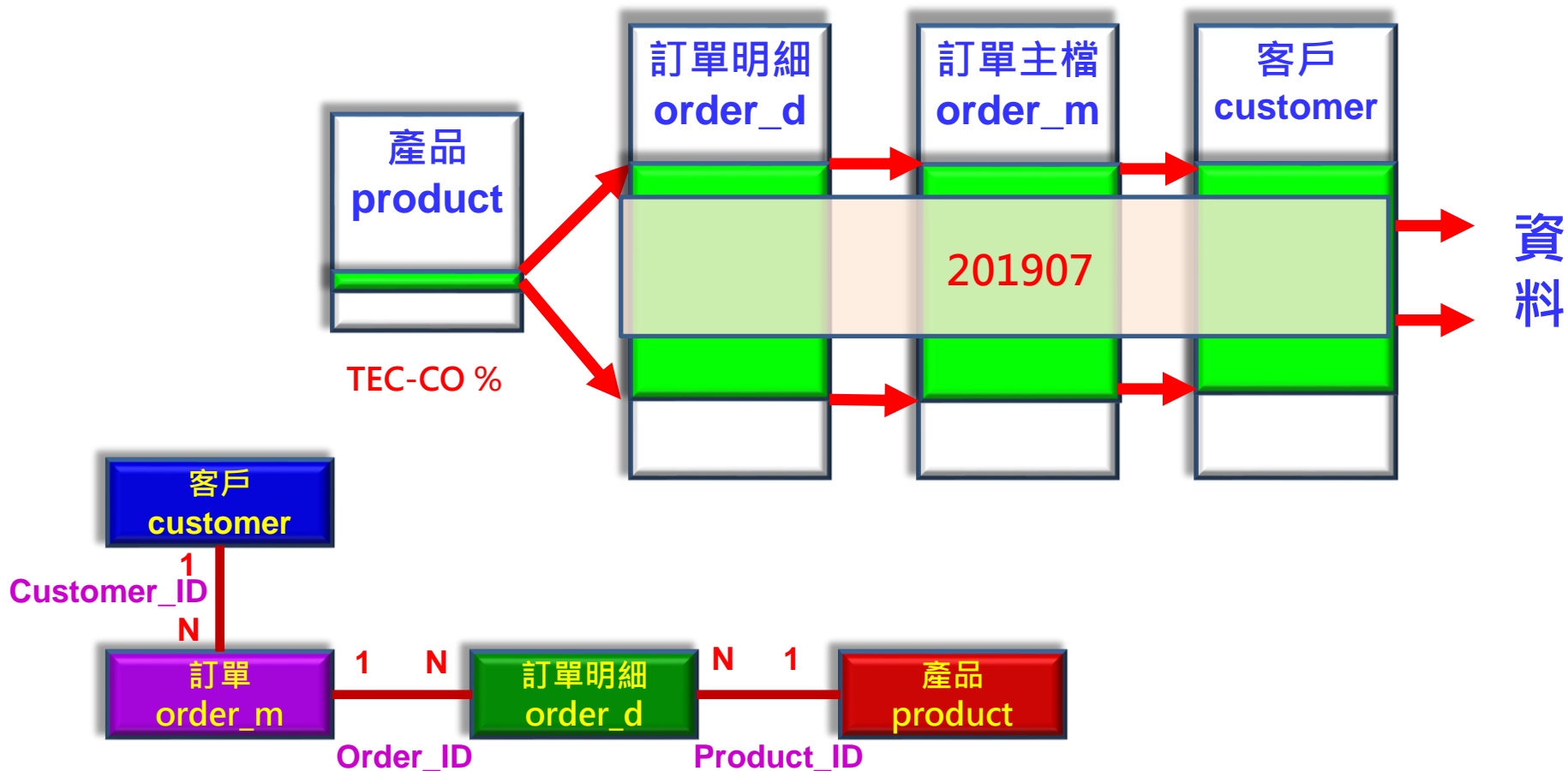
理解流程與關聯，資料取得就會比較容易

	ABC order_date	ABC Customer_ID	ABC Customer_Name	ABC Product_ID	ABC Product_Name	123 Quantity	123 Sales
1	2019/07/10	AA-10375	Allen Arnold	FUR-FU-10004064	Rubbermaid Door Stop, Ergonomic	2	84.84
2	2019/07/10	AA-10375	Allen Arnold	OFF-AP-10002963	Hoover Toaster, Red	8	669.12
3	2019/07/10	AA-10375	Allen Arnold	OFF-PA-10002618	Eaton Memo Slips, Multicolor	1	15.15
4	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10004841	Eldon Box, Blue	6	61.56
5	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10003907	Rogers Shelving, Wire Frame	4	244.8
6	2019/07/12	AG-10330	Alex Grayson	OFF-AP-10001956	Breville Refrigerator, Black	4	1,722.2832
7	2019/07/08	AG-10495	Andrew Gjertsen	OFF-AP-10001197	KitchenAid Stove, Red	1	285.075
8	2019/07/17	AG-10495	Andrew Gjertsen	OFF-BI-10000698	Wilson Jones Hole Reinforcements, Recycled	5	20.7825
9	2019/07/27	AR-10345	Alex Russell	OFF-AR-10000720	Binney & Smith Markers, Fluorescent	2	50.7

練習一

訂單日期	客戶編號	客戶名稱	產品編號	產品名稱	數量	金額
------	------	------	------	------	----	----

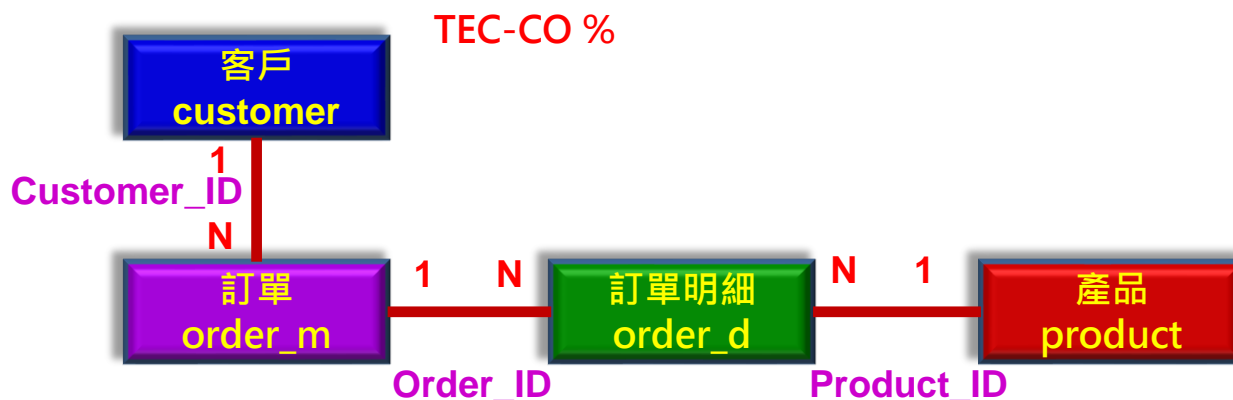
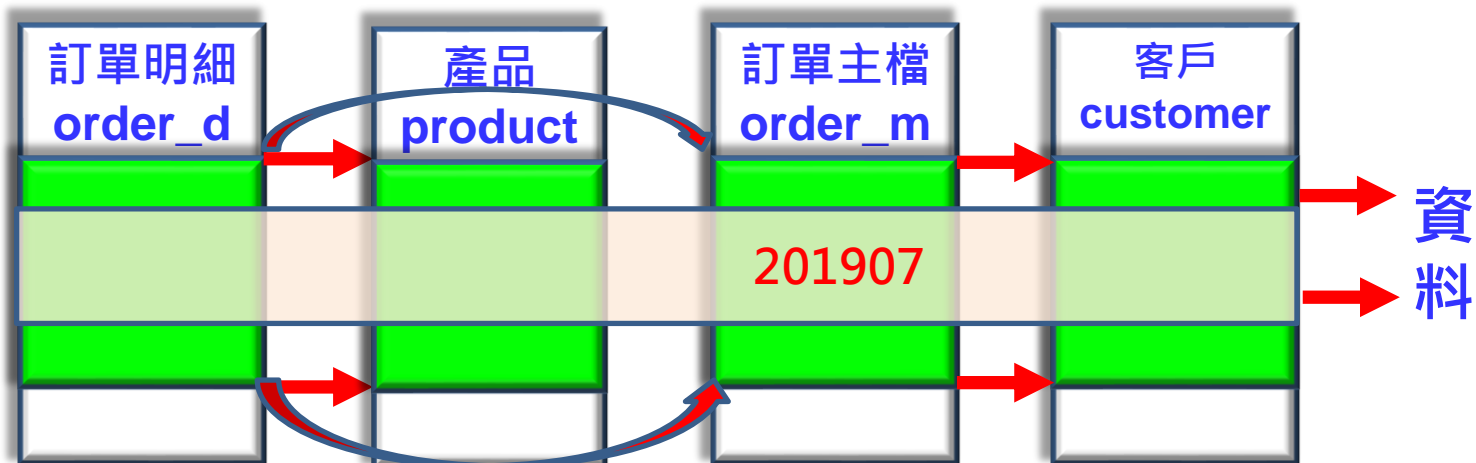
1.請找出產品代碼為TEC-CO的訂單有哪些(201907銷售清單)



練習一

訂單日期	客戶編號	客戶名稱	產品編號	產品名稱	數量	金額
------	------	------	------	------	----	----

1.請找出產品代碼為TEC-CO的訂單有哪些(201907銷售清單)



練習一答案

```
select c.order_date,c.Customer_ID,d.Customer_Name
      ,a.Product_ID,a.Product_Name
      ,b.Quantity,b.Sales
from product a
inner join order_d b on a.Product_ID=b.Product_ID
inner join order_m c on b.order_id=c.Order_ID
inner join customer d on c.Customer_ID=d.Customer_ID
where (c.Order_Date between '2019/07/01' and '2019/07/31')
      and a.Product_ID like 'TEC-CO%'
order by c.customer_id,c.order_date
```

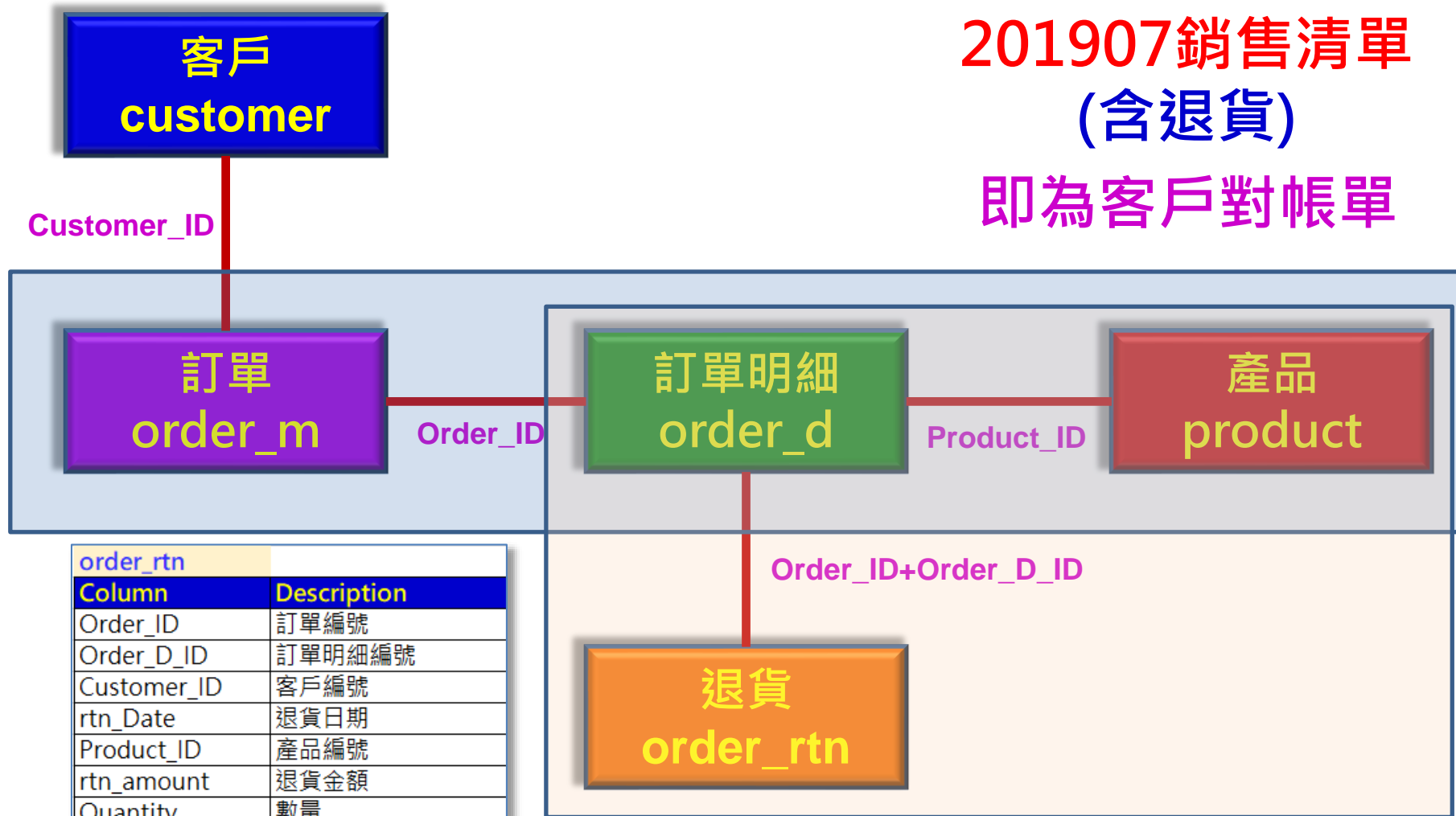
	ABC order_date	ABC Customer_ID	ABC Customer_Name	ABC Product_ID	ABC Product_Name	123 Quantity	123 Sales
1	2019/07/29	BT-11485	Brad Thomas	TEC-CO-10001166	HP Personal Copier, Digital	2	247.86
2	2019/07/22	ED-13885	Emily Ducich	TEC-CO-10000129	Brother Fax and Copier, Color	3	578.79
3	2019/07/09	EM-14140	Eugene Moren	TEC-CO-10003982	HP Ink, High-Speed	4	507.6
4	2019/07/08	JE-15610	Jim Epp	TEC-CO-10000452	HP Copy Machine, Laser	2	243.33
5	2019/07/11	JH-16180	Justin Hirsh	TEC-CO-10000660	Brother Fax and Copier, Laser	5	890.4285
6	2019/07/23	JP-15520	Jeremy Pistek	TEC-CO-10004981	Canon Copy Machine, High-Speed	2	468.612
7	2019/07/01	KW-16570	Kelly Williams	TEC-CO-10004125	Sharp Ink, Laser	2	248.82
8	2019/07/29	MD-17860	Michael Dominguez	TEC-CO-10002119	Sharp Fax Machine, Digital	7	1,922.3379
9	2019/07/09	PB-19210	Phillip Breyer	TEC-CO-10004897	Sharp Fax Machine, High-Speed	2	370.251
10	2019/07/02	RB-19330	Randy Bradley	TEC-CO-10003217	Canon Wireless Fax, Laser	2	757.32

☒ Save
 ☐ Cancel

練習二

訂單日期	客戶編號	客戶名稱	產品編號	產品名稱	數量	金額小計
------	------	------	------	------	----	------

201907銷售清單
(含退貨)
即為客戶對帳單



練習二答案

取出銷售資料

```
select a.order_date,a.Customer_ID,b.Customer_Name
      ,d.Product_ID,d.Product_Name
      ,c.Quantity,c.Sales
from order_m a
inner join customer b on a.customer_id=b.Customer_ID
inner join order_d c on a.order_id=c.Order_ID
inner join product d on c.Product_ID=d.Product_ID
where a.Order_Date between '2019/07/01' and '2019/07/31'
```

取出退貨資料

```
select a.rtn_Date,c.Customer_ID,c.Customer_Name
      ,b.Product_ID,b.Product_Name
      ,-1 * a.Quantity,-1* a.rtn_amount
from order_rtn a
inner join product b on a.Product_ID=b.Product_ID
inner join customer c on a.customer_id=c.Customer_ID
where a.rtn_Date between '2019/07/01' and '2019/07/31'
```


練習二答案

```
select a.order_date,a.Customer_ID,b.Customer_Name
      ,d.Product_ID,d.Product_Name
      ,c.Quantity,c.Sales
from order_m a
inner join customer b on a.customer_id=b.Customer_ID
inner join order_d c on a.order_id=c.Order_ID
inner join product d on c.Product_ID=d.Product_ID
where a.Order_Date between '2019/07/01' and '2019/07/31'
union ALL
select a.rtn_Date,c.Customer_ID,c.Customer_Name
      ,b.Product_ID,b.Product_Name
      ,-1 * a.Quantity,-1* a.rtn_amount
from order_rtn a
inner join product b on a.Product_ID=b.Product_ID
inner join customer c on a.customer_id=c.Customer_ID
where a.rtn_Date between '2019/07/01' and '2019/07/31'
order by a.customer_id,a.order_date
```

資料合併

排序

練習二答案

	ABC order_date	ABC Customer_ID	ABC Customer_Name	ABC Product_ID	ABC Product_Name	123 Quantity	123 Sales
1	2019/07/10	AA-10375	Allen Arnold	FUR-FU-10004064	Rubbermaid Door Stop, Ergonomic	2	84.84
2	2019/07/10	AA-10375	Allen Arnold	OFF-AP-10002963	Hoover Toaster, Red	8	669.12
3	2019/07/10	AA-10375	Allen Arnold	OFF-PA-10002618	Eaton Memo Slips, Multicolor	1	15.15
4	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10003907	Rogers Shelving, Wire Frame	4	244.8
5	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10004841	Eldon Box, Blue	6	61.56
6	2019/07/12	AG-10330	Alex Grayson	OFF-AP-10001956	Breville Refrigerator, Black	4	1,722.2832
7	2019/07/08	AG-10495	Andrew Gjertsen	OFF-AP-10001197	KitchenAid Stove, Red	1	285.075
8	2019/07/17	AG-10495	Andrew Gjertsen	OFF-BI-10000698	Wilson Jones Hole Reinforcements, I	5	20.7825
9	2019/07/02	AH-10690	Anna Häberlin	OFF-BI-10001659	Cardinal Index Tab, Clear	-1	-7.26
10	2019/07/27	AR-10345	Alex Russell	OFF-AR-10000720	Binney & Smith Markers, Fluorescer	2	50.7
11	2019/07/08	BD-11725	Bruce Degenhardt	FUR-FU-10002659	Eldon Clock, Black	2	113.04
12	2019/07/08	BD-11725	Bruce Degenhardt	OFF-SU-10002362	Acme Ruler, Serrated	5	67.05
13	2019/07/15	BE-11410	Bobby Elias	OFF-PA-10002974	Enermax Note Cards, 8.5 x 11	3	70.7409
14	2019/07/16	BM-11140	Becky Martin	FUR-CH-10002589	SAFCO Steel Folding Chair, Red	2	171.42
15	2019/07/16	BM-11140	Becky Martin	FUR-FU-10003609	Rubbermaid Frame, Duo Pack	2	219.9
16	2019/07/16	BM-11140	Becky Martin	OFF-BI-10004562	Acco Binder, Clear	2	29.22
17	2019/07/16	BM-11140	Becky Martin	OFF-LA-10002086	Hon Shipping Labels, 5000 Label Se	2	23.52
18	2019/07/16	BM-11140	Becky Martin	OFF-SU-10001488	Stiletto Scissors, Steel	1	23.01
19	2019/07/26	BM-11575	Brendan Murry	FUR-FU-10000628	Rubbermaid Light Bulb, Black	3	42.39
20	2019/07/26	BM-11575	Brendan Murry	OFF-LA-10000506	Harbour Creations Legal Exhibit Lab	2	12.045

練習二答案

另一種做法

```
select a.order_date,a.Customer_ID,c.Customer_Name
      ,a.Product_ID,b.Product_Name
      ,a.Quantity,a.Sales
from
(
    select a.order_date,a.Customer_ID,c.Product_ID
          ,c.Quantity,c.Sales
    from order_m a
    inner join order_d c on a.order_id=c.Order_ID
    where a.Order_Date between '2019/07/01' and '2019/07/31'
    union ALL
    select a.rtn_Date,a.Customer_ID,a.Product_ID
          ,-1 * a.Quantity,-1* a.rtn_amount
    from order_rtn a
    where a.rtn_Date between '2019/07/01' and '2019/07/31'
) a
inner join product b on a.Product_ID=b.Product_ID
inner join customer c on a.customer_id=c.Customer_ID
order by a.customer_id,a.order_date
```

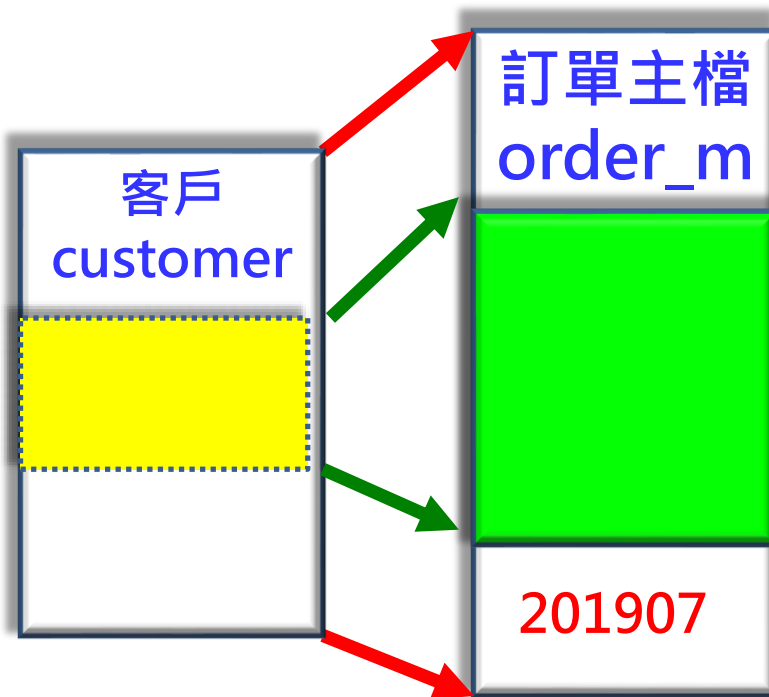
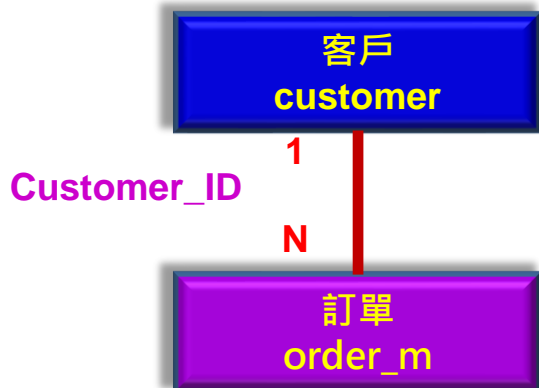
名稱主要是顯示使用，所以可以先取得
資料後再進行後續處理

練習三

客戶編號 客戶名稱

針對未購買客戶可採取哪些行動

3.請找出哪些客戶沒有購買產品，只要第二個table裡的隨便一個欄位，如 order_m別名為b，則b.Customer_ID is null (201907銷售清單)



練習三答案

錯誤版(因為先合併資料後再處理201907資料)

```
select a.Customer_ID,a.Customer_Name
from customer a
left join order_m b on a.Customer_ID=b.Customer_ID
where b.Order_Date between '2019/07/01' and '2019/07/31'
      and b.Customer_ID is null
order by a.customer_id
```

	ABC Customer_ID	ABC Customer_Name
1	AA-10315	Alex Avila
2	AA-10480	Andrew Allen
3	AA-10645	Anna Andreadi
4	AB-10015	Aaron Bergman
5	AB-10060	Adam Bellavance

正確版

```
select a.Customer_ID,a.Customer_Name
from customer a
left join order_m b on a.Customer_ID=b.Customer_ID and b.Order_Date between '2019/07/01' and '2019/07/31'
where b.Customer_ID is null
order by a.customer_id
```

or

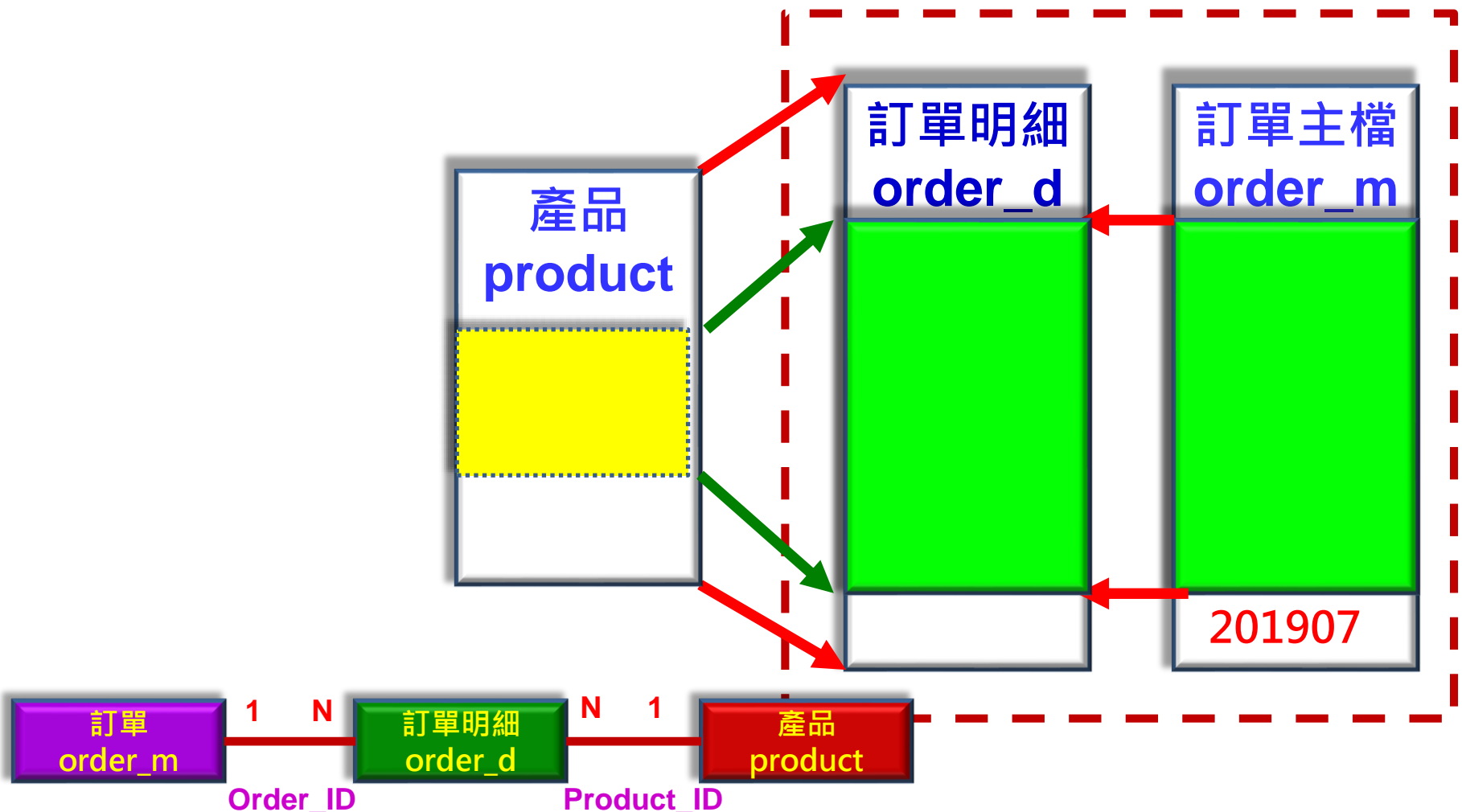
```
select a.Customer_ID,a.Customer_Name
from customer a
left join (select * from order_m where Order_Date between '2019/07/01' and '2019/07/31') b on a.Customer_ID=b.Customer_ID
where b.Customer_ID is null
order by a.customer_id
```


練習四

產品編號 產品名稱

針對未銷售的產品可採取哪些行動

4.請找出哪些產品沒有被購買 (201907銷售清單)



練習四答案

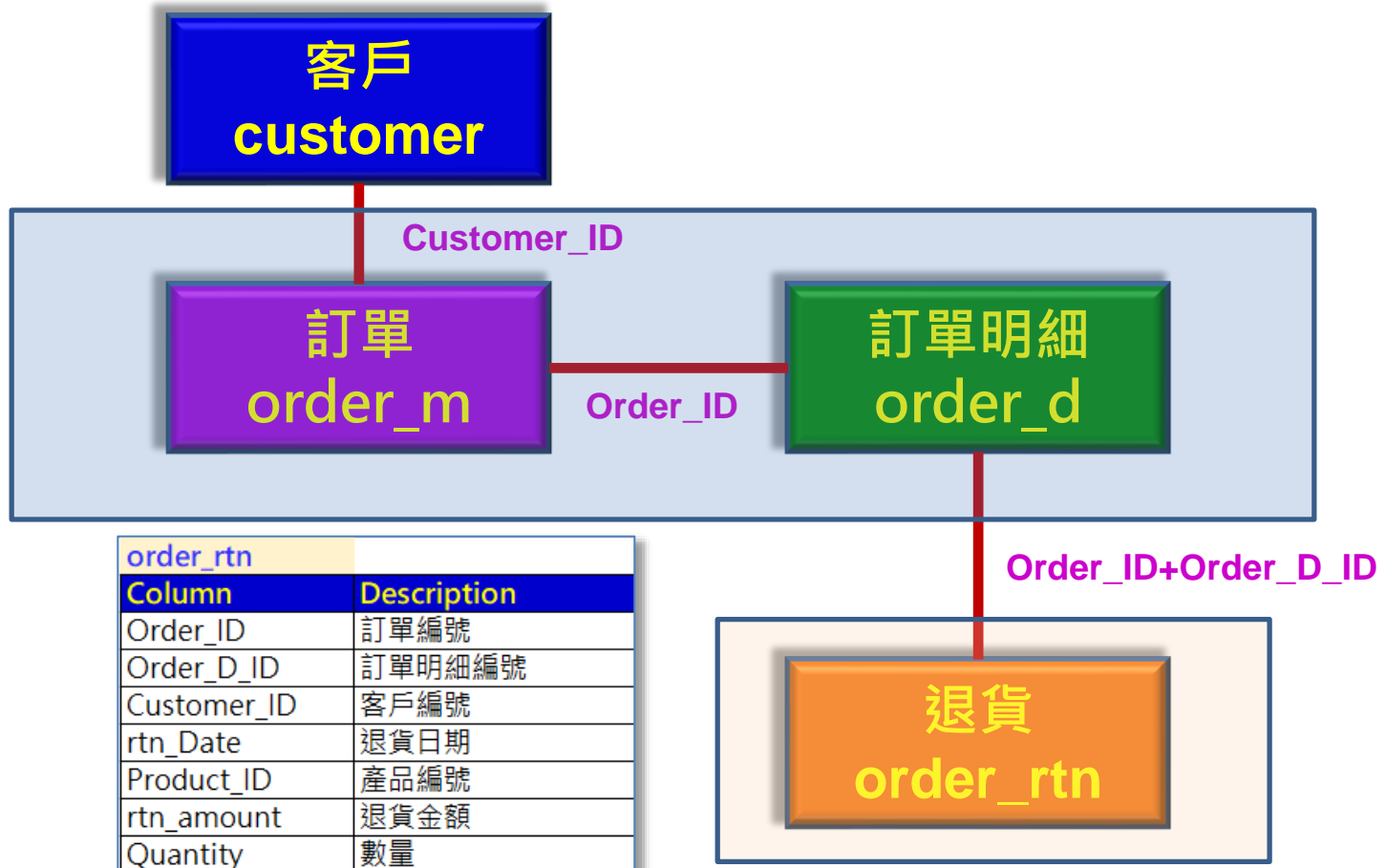
```
select a.Product_ID,a.Product_Name
from product a
left join
(
    select y.Product_ID
    from order_m x
    inner join order_d y on x.order_id=y.order_id
    where substring(x.Order_Date,1,7)='2019/07'
    group by y.Product_ID
) b on a.Product_ID=b.Product_ID
where b.Product_ID is null
order by a.Product_ID
```

	ABC Product_ID	ABC Product_Name
1	FUR-BO-10000021	Dania Corner Shelving, Metal
2	FUR-BO-10000035	Dania Classic Bookcase, Pine
3	FUR-BO-10000071	Safco 3-Shelf Cabinet, Metal
4	FUR-BO-10000087	Dania Classic Bookcase, Mobile
5	FUR-BO-10000175	Ikea Stackable Bookrack, Metal
6	FUR-BO-10000195	Dania Stackable Bookrack, Pine
7	FUR-BO-10000203	Sauder Library with Doors, Metal
8	FUR-BO-10000208	Bush Stackable Bookrack, Mobile
9	FUR-BO-10000210	Ikea Library with Doors, Pine
10	FUR-BO-10000243	Safco Floating Shelf Set, Pine

練習五

年月	客戶編號	客戶名稱	銷售金額	退貨金額	合計金額
----	------	------	------	------	------

5.客戶銷售總表(201907銷售,含退貨)



練習五答案

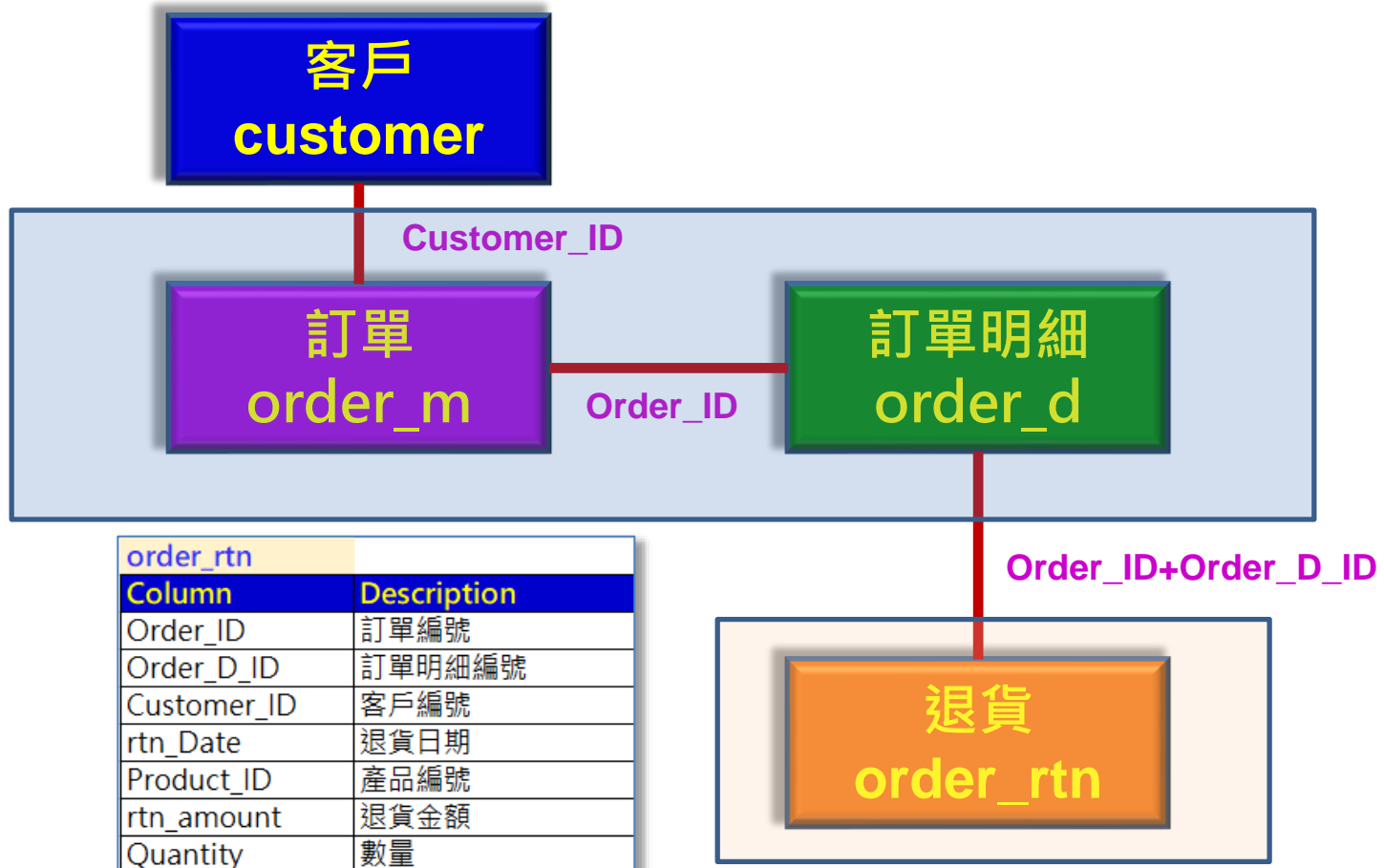
```
select a.yyyymm,a.Customer_ID,c.Customer_Name
      ,sum(a.Sales) sale_amt,sum(a.rtn_amt) rtn_amt
      ,sum(a.Sales-a.rtn_amt) total_amt
from
(
    select substring(a.order_date,1,7) yyyymm,a.Customer_ID,c.Product_ID
          ,c.Sales, 0 rtn_amt
    from order_m a
    inner join order_d c on a.order_id=c.Order_ID
    where substring(a.order_date,1,7)='2019/07'
    union ALL
    select substring(a.rtn_Date,1,7),a.Customer_ID,a.Product_ID
          ,0, a.rtn_amount
    from order_rtn a
    where substring(a.rtn_Date,1,7)='2019/07'
) a
inner join customer c on a.customer_id=c.Customer_ID
group by a.yyyymm,a.Customer_ID,c.Customer_Name
order by a.yyyymm,a.Customer_ID
```

	ABC yyyymm	ABC Customer_ID	ABC Customer_Name	123 sale_amt	123 rtn_amt	123 total_amt
16	2019/07	CD-12280	Christina DeMoss	2,634.48	0	2,634.48
17	2019/07	CK-12595	Clytie Keltz	291.78	291.78	0
18	2019/07	CM-12385	Christopher Martinez	993.114	0	993.114
19	2019/07	CM-12715	Craig Molinari	198.33	0	198.33
20	2019/07	CP-12340	Christine Phan	4,752.81	0	4,752.81
21	2019/07	CR-12580	Clay Rozendal	26.712	0	26.712
22	2019/07	CY-12745	Craig Yedwab	456.27	0	456.27
23	2019/07	DB-13060	Dave Brooks	47.1357	0	47.1357
24	2019/07	DB-13660	Duane Benoit	2,583.9	2,583.9	0
25	2019/07	DH-13675	Duane Huffman	448.8	0	448.8

練習六

年月	客戶編號	客戶名稱	合計金額	銷售總額	比率
----	------	------	------	------	----

6.客戶銷售佔銷售總額比例,比率由大到小排序(201907銷售,含退貨)



練習六答案

```
-- drop table #temp1
```

如果已在視窗內進行分析,若想再一次分析時,必須卸除暫存檔

```
select a.yyyymm,a.Customer_ID,c.Customer_Name
      ,sum(a.Sales) sale_amt,sum(a.rtn_amt) rtn_amt
      ,sum(a.Sales-a.rtn_amt) total_amt
```

```
into #temp1
```

```
from
(
    select substring(a.order_date,1,7) yyyymm,a.Customer_ID,c.Product_ID
          ,c.Sales, 0 rtn_amt
    from order_m a
    inner join order_d c on a.order_id=c.Order_ID
    where substring(a.order_date,1,7)='2019/07'
    union ALL
    select substring(a.rtn_Date,1,7),a.Customer_ID,a.Product_ID
          ,0, a.rtn_amount
    from order_rtn a
    where substring(a.rtn_Date,1,7)='2019/07'
) a
inner join customer c on a.customer_id=c.Customer_ID
group by a.yyyymm,a.Customer_ID,c.Customer_Name
order by a.yyyymm,a.Customer_ID;
```

```
select *
from
(
    select a.yyyymm,a.Customer_ID,a.Customer_Name,a.total_amt
          ,b.sum_total_amt
          ,convert(varchar(8), convert(decimal(18,2),a.total_amt/b.sum_total_amt*100) )+ '%' ct_percent
    from #temp1 a
    left join (select sum(total_amt) sum_total_amt from #temp1) b on 1=1
) a
order by ct_percent desc
```

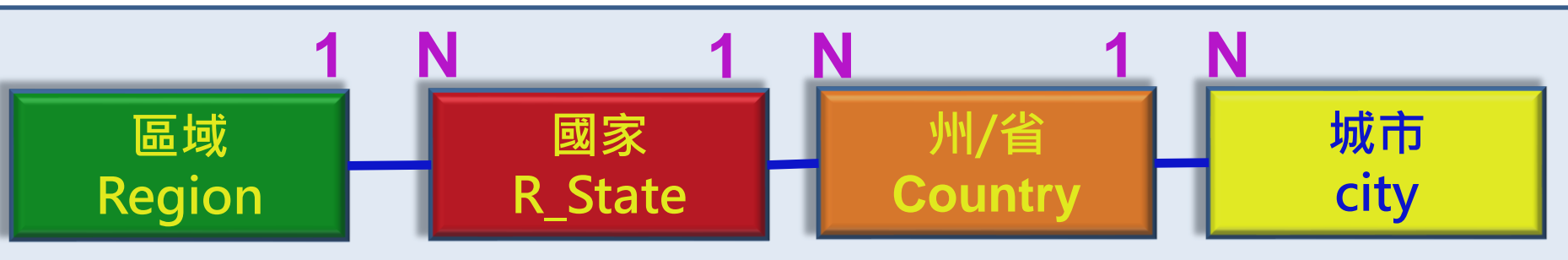
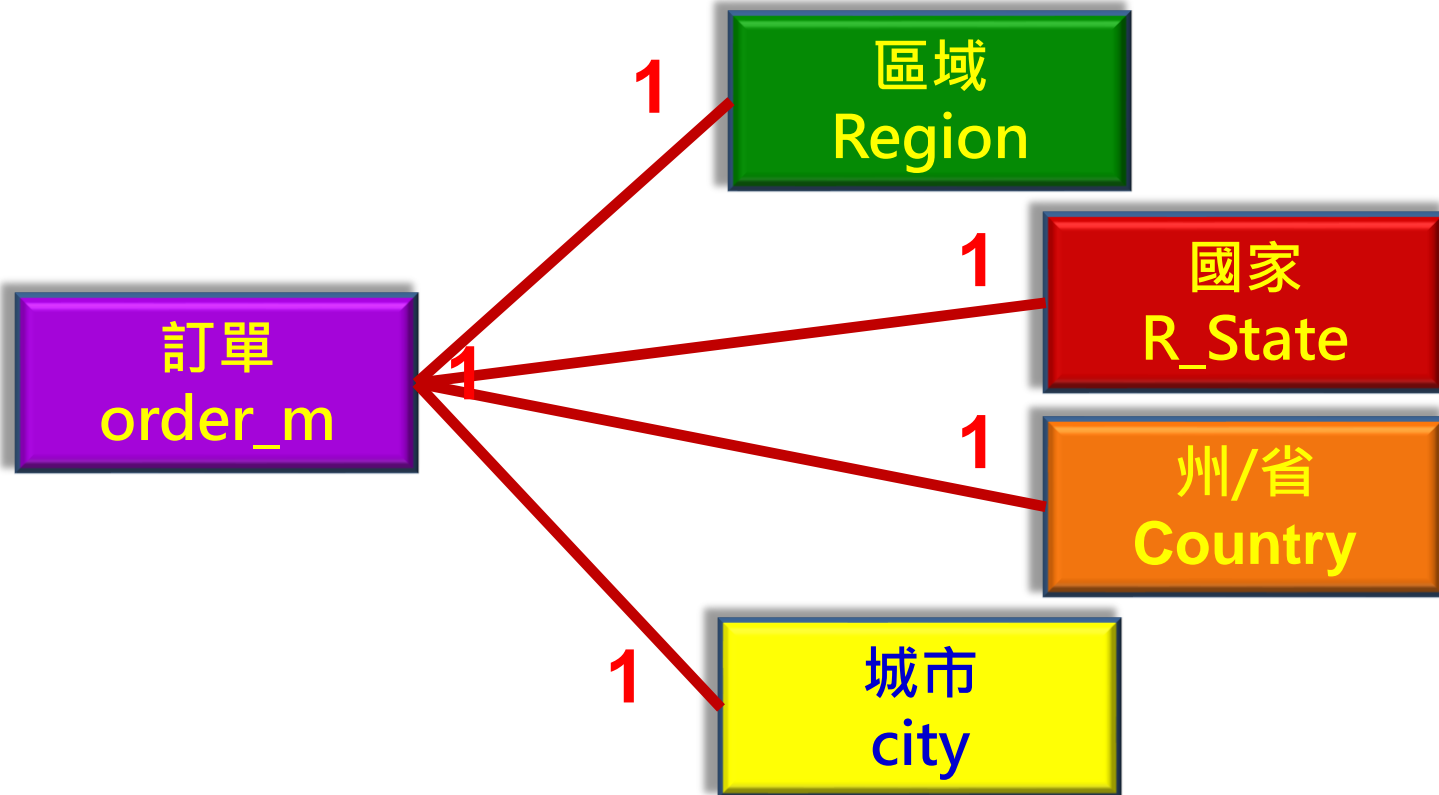
如果想顯示%,必須轉換為文字

	avg yyyymm	avg Customer_ID	avg Customer_Name	123 total_amt	123 sum_total_amt	avg ct_percent
1	2019/07	CP-12340	Christine Phan	4,752.81	62,517.4743	7.60%
2	2019/07	BN-11470	Brad Norvell	4,727.76	62,517.4743	7.56%
3	2019/07	JK-16120	Julie Kriz	3,559.086	62,517.4743	5.69%

SQL指令

- declare(宣告變數) 、 set (設定變數值)
- dateadd 、 datediff 、 datepart
- if
- create view 、 alter view 、 drop view
- create function 、 alter function 、 drop function
- create procedure 、 alter procedure 、 drop procedure

資料關聯圖



進階SQL資料分析

區域

國家號

州/省

城市

Region

Column	Description
<u>Region_id</u>	
Region_name	

City

Column	Description
<u>Region_id</u>	
<u>State_id</u>	
<u>Country_id</u>	
<u>city_id</u>	
city_name	

R_state

Column	Description
<u>Region_id</u>	
<u>State_id</u>	
State_name	

Country

Column	Description
<u>Region_id</u>	
<u>State_id</u>	
<u>Country_id</u>	
Country_name	

	ABC Region_name T	ABC State_name T	ABC Country_name T	ABC city_name T
1	中亞	Afghanistan	Kabul	Kabul
2	中亞	Afghanistan	Kandahar	Kandahar
3	中亞	Afghanistan	Nangarhar	Jalalabad
4	中亞	Bangladesh	Chittagong	Chittagong
5	中亞	Bangladesh	Chittagong	Cox's Bāzār
6	中亞	Bangladesh	Dhaka	Dhaka

```
select a.Region_name ,b.State_name ,c.Country_name ,d.city_name
from Region a
inner join R_State b on a.Region_id=b.Region_id
inner join Country c on b.Region_id=c.Region_id and b.State_id=c.State_id
inner join city d on c.Region_id=d.Region_id and c.State_id=d.State_id and c.Country_id=d.Country_id
order by a.Region_name ,b.State_name ,c.Country_name ,d.city_name
```

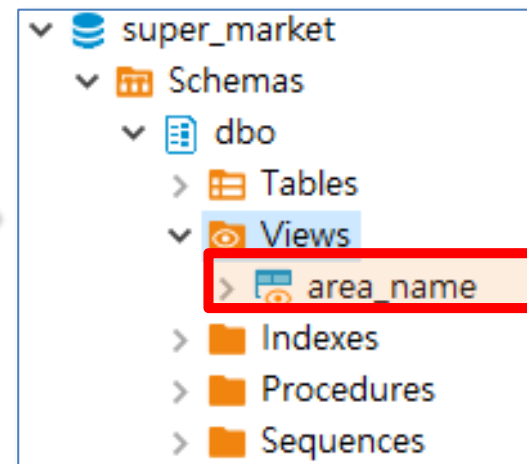
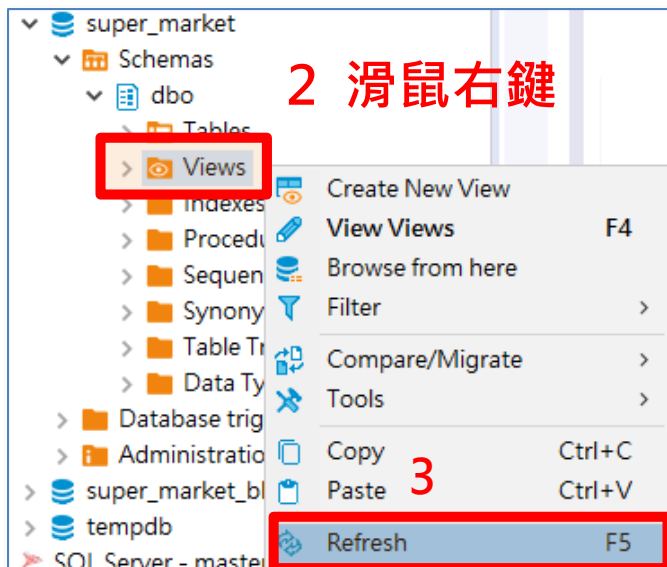
進階SQL資料分析-View

- CREATE [OR ALTER] VIEW** [schema_name .] view_name
[(column_name [,...n])] **AS** <select_statement> [;]

```
create view area_name
as

select a.Region_id ,a.Region_name ,b.State_id ,b.State_name
      ,c.Country_id ,c.Country_name ,d.city_id ,d.city_name
from Region a
inner join R_State b on a.Region_id=b.Region_id
inner join Country c on b.Region_id=c.Region_id and b.State_id=c.State_id
inner join city d on c.Region_id=d.Region_id and c.State_id=d.State_id and c.Country_id=d.Country_id
```

1



進階SQL資料分析-View

需先建立view，執行後，會產生一個view，再select這個view，view等同一個資料集(從多個資料集整合成一個資料集)

```
select *  
from area_name  
where region_id='R004' and State_id='S0021'  
order by Region_id , State_id ,Country_id ,city_id
```

area_name

select * from area_name where region_id = 'R004' and state_id = 'S0021'

	ABC Region_id	ABC Region_name	ABC State_id	ABC State_name	ABC Country_id	ABC Country_name	ABC city_id	ABC city_name
1	R004	北亞	S0021	Taiwan	C0152	Taichung City	CY00442	Taichung
2	R004	北亞	S0021	Taiwan	C0153	Tainan City	CY00443	Tainan
3	R004	北亞	S0021	Taiwan	C0154	Taipei City	CY00444	Taipei

View通常用在給特定的人部分資料

進階SQL資料分析-Function

declare(宣告變數)

set (設定變數值)

```
declare
@date_s    nvarchar(10)='2018/01/01',
@date_e    nvarchar(10)='2018/01/01',
@search_str nvarchar(10)=N'二%'

-- 設定變數值
set @date_s='2018/01/01'

--也可使用此方法設定變數值
--select @date_s='2018/01/01'

select *
from order_m
where (Order_Date between @date_s and @date_e) and Ship_Mode like @search_str
```

當連續要宣告變數時，需要有【,】
最後一個宣告不可有【,】



	ABC Order_ID	ABC Order_Date	ABC Ship_Date	ABC Ship_Mode	ABC Customer_ID	ABC Region_id	ABC State_id	ABC Country_id	ABC city_id
1	ID-2016-20212	2018/01/01	2018/01/03	二級	JK-15325	R003	S0022	C0011	CY00038
2	IN-2016-10720	2018/01/01	2018/01/05	二級	KT-16480	R002	S0007	C0173	CY00042
3	IN-2016-80447	2018/01/01	2018/01/03	二級	KN-16390	R001	S0002	C0106	CY00085

進階SQL資料分析-Function

日期函數 (在某個日期加上或減去指定時間間隔)

DATEADD (datepart , number , date)

datepart	縮寫
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww
weekday	dw
hour	hh
minute	mi, n
second	ss, s

```
select 'year' as title, dateadd(year,-1,getdate())  
union all  
select 'month' as title, dateadd(month,-1,getdate())  
union all  
select 'dd' as title, dateadd(dd,1,getdate())  
union all  
select 'week' as title, dateadd(week,1,getdate())  
union all  
select 'hh' as title, dateadd(hh,1,getdate())
```



	ABC title	
1	year	2019-02-14 09:54:27
2	month	2020-01-14 09:54:27
3	dd	2020-02-15 09:54:27
4	week	2020-02-21 09:54:27
5	hh	2020-02-14 10:54:27

進階SQL資料分析-Function

日期函數 (兩個指定日期之間的時間間隔數)

DATEDIFF (datepart , startdate , enddate)

datepart	縮寫
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww
weekday	dw
hour	hh
minute	mi, n
second	ss, s

```
select 'year' as title, datediff(year,'20200101',getdate())  
union all  
select 'month' as title, datediff(month,'20200101',getdate())  
union all  
select 'dd' as title, datediff(dd,'20200101',getdate())  
union all  
select 'week' as title, datediff(week,'20200101',getdate())  
union all  
select 'hh' as title, datediff(hh,'20200101',getdate())
```



	ABC title ↑↓	123 ↑↓
1	year	0
2	month	1
3	dd	44
4	week	6
5	hh	1,065

進階SQL資料分析-Function

日期函數 (返回日期/時間的單獨資料，如年、月、日、小時、分鐘等)

DATEPART (datepart , date)

datepart	縮寫
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww
weekday	dw
hour	hh
minute	mi, n
second	ss, s

```
select 'year' as title, datepart(year,getdate())  
union all  
select 'month' as title, datepart(month,getdate())  
union all  
select 'dd' as title, datepart(dd,getdate())  
union all  
select 'week' as title, datepart(week,getdate())  
union all  
select 'hh' as title, datepart(hh,getdate())
```



	ABC title	123
1	year	2,020
2	month	2
3	dd	14
4	week	7
5	hh	9

進階SQL資料分析-Function

IF.... ELSE 函數

```
declare @test int=2;

if @test=1
    Begin
        select N'變數為1';
    end
else
    begin
        if @test=2
            begin
                select N'變數為2';
            end
        else
            begin
                select N'變數不是1或2';
            end
        end
    end
end
```



	ABC ↑↓	
1	變數為2	

進階SQL資料分析-Function

```
CREATE [ OR ALTER ] FUNCTION [ schema_name. ] function_name
( [ { @parameter_name [ AS ] [ type_schema_name. ]
parameter_data_type
    [ = default ] [ READONLY ] }
    [ ,...n ]
]
)
RETURNS return_data_type
    [ WITH <function_option> [ ,...n ] ]
    [ AS ]
BEGIN
    function_body
    RETURN scalar_expression
END
[ ; ]
```

進階SQL資料分析-Function

純量值函式範例(取得地址所在的相關區域)

```
create function dbo.get_addr_area (@Region_id nvarchar(10),@State_id nvarchar(10)
                                ,@Country_id nvarchar(10),@city_id nvarchar(10))
returns nvarchar(200)
as
begin
    declare
        @data_str nvarchar(200)

    select @data_str=a.Region_name+'_'+b.State_name+'_'+c.Country_name+'_'+d.city_name
    from Region a
    inner join R_State b on a.Region_id=b.Region_id
    inner join Country c on b.Region_id=c.Region_id and b.State_id=c.State_id
    inner join city d on c.Region_id=d.Region_id and c.State_id=d.State_id and c.Country_id=d.Country_id
    where @Region_id=d.Region_id and @State_id=d.State_id
        and @Country_id=d.Country_id and @city_id=d.city_id

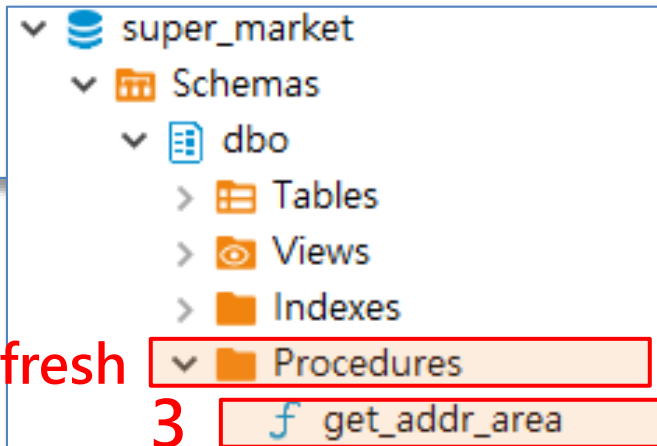
    return @data_str
end;
```

1

常用在演算法或公式計算，
並將結果回傳

2 Refresh

3



進階SQL資料分析-Function

```
select top 10
    dbo.get_addr_area(Region_id,State_id,Country_id,city_id) addr_area 1
,*
from order_m
```

	ABC addr_area	ABC Order_ID	ABC Order_Date	ABC Ship_Date	ABC Ship_Mode
1	東南亞_Philippines_Metro Manila_Manila	ID-2016-10706	2016/04/19	2016/04/23	標準級
2	東南亞_Thailand_Bangkok_Bangkok	ID-2016-11126	2016/11/26	2016/11/30	標準級
3	北亞_South Korea_Seoul_Seoul	ID-2016-11385	2016/08/06	2016/08/12	標準級
4	中亞_India_Maharashtra_Mumbai	ID-2016-11392	2016/09/06	2016/09/08	二級
5	東南亞_Thailand_Nakhon Ratchasima_Nakhon Ratchasima	ID-2016-12596	2016/01/03	2016/01/08	標準級
6	東南亞_Thailand_Bangkok_Bangkok	ID-2016-13114	2016/12/15	2016/12/20	標準級
7	中亞_Pakistan_Punjab_Lahore	ID-2016-13170	2016/12/16	2016/12/21	標準級
8	東南亞_Philippines_Rizal_Antipolo	ID-2016-13240	2016/11/26	2016/12/01	標準級
9	中亞_Pakistan_Punjab_Rawalpindi	ID-2016-13436	2016/05/19	2016/05/24	標準級
10	東南亞_Thailand_Bangkok_Bangkok	ID-2016-13660	2016/01/10	2016/01/15	二級

這SQL與上述取得的資料結果是相同的

```
select top 10
    (d.Region_name+'_'+e.State_name+'_'+f.Country_name+'_'+g.city_name) addr_area
,*
from order_m a
inner join Region d on a.Region_id=d.Region_id
inner join R_State e on a.Region_id=e.Region_id and a.State_id=e.State_id
inner join Country f on a.Region_id=f.Region_id and a.State_id=f.State_id and a.Country_id=f.Country_id
inner join city g on a.Region_id=g.Region_id and a.State_id=g.State_id and a.Country_id=g.Country_id and a.city_id=g.city_id
```

進階SQL資料分析-Function

資料表值函式範例(取得地址所在的相關區域)

```
create function dbo.get_addr_area_DS ()
```

```
returns TABLE
```

```
as
```

```
return
```

```
(
```

```
select d.Region_id ,d.State_id ,d.Country_id ,d.city_id
      ,(a.Region_name+'_'+b.State_name+'_'+c.Country_name+'_'+d.city_name) addr_area
from Region a
inner join R_State b on a.Region_id=b.Region_id
inner join Country c on b.Region_id=c.Region_id and b.State_id=c.State_id
inner join city d on c.Region_id=d.Region_id and c.State_id=d.State_id and c.Country_id=d.Country_id
```

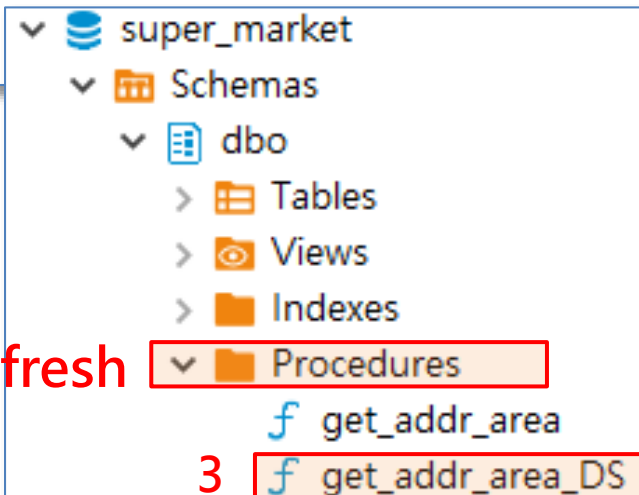
```
)
```

可以撰寫傳入參數,回傳結果是一個資料集

1

也常利用演算法或公式計算每筆資料，
再關聯(join)此資料表函式

2 Refresh



3

進階SQL資料分析-Function

```
select top 10
    b.addr_area
    ,a.*
from order_m a
inner join dbo.get_addr_area_DS() b on
    a.Region_id=b.Region_id and a.State_id=b.State_id and a.Country_id=b.Country_id and a.city_id=b.city_id
```

1

	ABC addr_area	ABC Order_ID	ABC Order_Date	ABC Ship_Date	ABC Ship_Mode
1	東南亞_Philippines_Metro Manila_Manila	ID-2016-10706	2016/04/19	2016/04/23	標準級
2	東南亞_Thailand_Bangkok_Bangkok	ID-2016-11126	2016/11/26	2016/11/30	標準級
3	北亞_South Korea_Seoul_Seoul	ID-2016-11385	2016/08/06	2016/08/12	標準級
4	中亞_India_Maharashtra_Mumbai	ID-2016-11392	2016/09/06	2016/09/08	二級
5	東南亞_Thailand_Nakhon Ratchasima_Nakhon Ratchasima	ID-2016-12596	2016/01/03	2016/01/08	標準級
6	東南亞_Thailand_Bangkok_Bangkok	ID-2016-13114	2016/12/15	2016/12/20	標準級
7	中亞_Pakistan_Punjab_Lahore	ID-2016-13170	2016/12/16	2016/12/21	標準級
8	東南亞_Philippines_Rizal_Antipolo	ID-2016-13240	2016/11/26	2016/12/01	標準級
9	中亞_Pakistan_Punjab_Rawalpindi	ID-2016-13436	2016/05/19	2016/05/24	標準級
10	東南亞_Thailand_Bangkok_Bangkok	ID-2016-13660	2016/01/10	2016/01/15	二級

2

這SQL與上述取得的資料結果是相同的

```
select top 10
    (d.Region_name+'_'+e.State_name+'_'+f.Country_name+'_'+g.city_name) addr_area
    ,a.*
from order_m a
inner join Region d on a.Region_id=d.Region_id
inner join R_State e on a.Region_id=e.Region_id and a.State_id=e.State_id
inner join Country f on a.Region_id=f.Region_id and a.State_id=f.State_id and a.Country_id=f.Country_id
inner join city g on a.Region_id=g.Region_id and a.State_id=g.State_id and a.Country_id=g.Country_id and a.city_id=g.city_id
```

進階SQL資料分析-Stored Procedure

-- Transact-SQL Syntax for Stored Procedures in SQL Server and Azure

```
CREATE [ OR ALTER ] { PROC | PROCEDURE }  
    [ schema_name. ] procedure_name [ ; number ]  
    [ { @parameter [ type_schema_name. ] data_type }  
        [ VARYING ] [ = default ] [ OUT | OUTPUT | [ READONLY ]  
    ] [ , ...n ]  
    [ WITH <procedure_option> [ , ...n ] ]  
    [ FOR REPLICATION ]  
AS { [ BEGIN ] sql_statement [ ; ] [ ...n ] [ END ] }  
[ ; ]
```

```
<procedure_option> ::=  
    [ ENCRYPTION ]  
    [ RECOMPILE ]  
    [ EXECUTE AS Clause ]
```

進階SQL資料分析-Stored Procedure

Stored Procedure範例(客戶對帳單)

```
create procedure cust_sale_sp(@yyyymm nvarchar(7),@Customer_ID nvarchar(50))  
AS
```

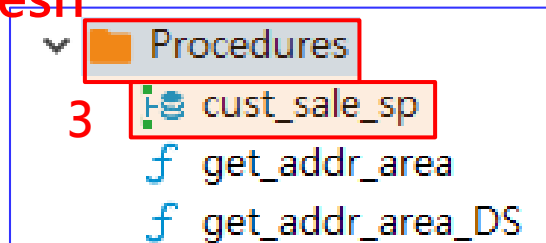
1

```
select a.order_date,a.Customer_ID,b.Customer_Name  
      ,d.Product_ID,d.Product_Name  
      ,c.Quantity,c.Sales  
from order_m a  
inner join customer b on a.customer_id=b.Customer_ID  
inner join order_d c on a.order_id=c.Order_ID  
inner join product d on c.Product_ID=d.Product_ID  
where substring(a.Order_Date,1,7)=@yyyymm  
      and a.customer_id=case when @Customer_ID='' then a.customer_id else @Customer_ID end  
union ALL  
select a.rtn_Date,c.Customer_ID,c.Customer_Name  
      ,b.Product_ID,b.Product_Name  
      ,-1 * a.Quantity,-1* a.rtn_amount  
from order_rtn a  
inner join product b on a.Product_ID=b.Product_ID  
inner join customer c on a.customer_id=c.Customer_ID  
where substring(a.rtn_Date,1,7)=@yyyymm  
      and a.customer_id=case when @Customer_ID='' then a.customer_id else @Customer_ID end  
order by a.customer_id,a.order_date
```

傳入參數

回傳結果可以是1..N個資料集

2 Refresh



如果@Customer_ID="",a.customer_id=a.customer_id
如果@Customer_ID<> "",a.customer_id=@Customer_ID

進階SQL資料分析-Stored Procedure

```
execute cust_sale_sp '2019/07','BN-11470';
```

	order_date	Customer_ID	Customer_Name	Product_ID	Product_Name	Quantity	Sales
1	2019/07/04	BN-11470	Brad Norvell	FUR-FU-10001477-	Deflect-O Frame, Ergonomic	2	217.32
2	2019/07/04	BN-11470	Brad Norvell	OFF-AP-10002793	KitchenAid Stove, Black	6	3,417.48
3	2019/07/04	BN-11470	Brad Norvell	OFF-FA-10004705	OIC Thumb Tacks, 12 Pack	2	27
4	2019/07/04	BN-11470	Brad Norvell	TEC-MA-10000334	Panasonic Printer, Wireless	4	1,065.96

```
execute cust_sale_sp '2019/07','';
```

	order_date	Customer_ID	Customer_Name	Product_ID	Product_Name	Quantity	Sales
1	2019/07/10	AA-10375	Allen Arnold	FUR-FU-10004064	Rubbermaid Door Stop, Ergonomic	2	84.84
2	2019/07/10	AA-10375	Allen Arnold	OFF-AP-10002963	Hoover Toaster, Red	8	669.12
3	2019/07/10	AA-10375	Allen Arnold	OFF-PA-10002618	Eaton Memo Slips, Multicolor	1	15.15
4	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10003907	Rogers Shelving, Wire Frame	4	244.8
5	2019/07/10	AA-10375	Allen Arnold	OFF-ST-10004841	Eldon Box, Blue	6	61.56
6	2019/07/12	AG-10330	Alex Grayson	OFF-AP-10001956	Breville Refrigerator, Black	4	1,722.2832
7	2019/07/08	AG-10495	Andrew Gjertsen	OFF-AP-10001197	KitchenAid Stove, Red	1	285.075
8	2019/07/17	AG-10495	Andrew Gjertsen	OFF-BI-10000698	Wilson Jones Hole Reinforcements, I	5	20.7825
9	2019/07/02	AH-10690	Anna Häberlin	OFF-BI-10001659	Cardinal Index Tab, Clear	-1	-7.26

練習一

客戶2019年1-12月銷售統計

```
select a.customer_id,a.Customer_Name
, sum(case when substring(b.order_date,6,2)='01' then b.Sales else 0 end) sale_01
, sum(case when substring(b.order_date,6,2)='02' then b.Sales else 0 end) sale_02
, sum(case when substring(b.order_date,6,2)='03' then b.Sales else 0 end) sale_03
, sum(case when substring(b.order_date,6,2)='04' then b.Sales else 0 end) sale_04
, sum(case when substring(b.order_date,6,2)='05' then b.Sales else 0 end) sale_05
, sum(case when substring(b.order_date,6,2)='06' then b.Sales else 0 end) sale_06
, sum(case when substring(b.order_date,6,2)='07' then b.Sales else 0 end) sale_07
, sum(case when substring(b.order_date,6,2)='08' then b.Sales else 0 end) sale_08
, sum(case when substring(b.order_date,6,2)='09' then b.Sales else 0 end) sale_09
, sum(case when substring(b.order_date,6,2)='10' then b.Sales else 0 end) sale_10
, sum(case when substring(b.order_date,6,2)='11' then b.Sales else 0 end) sale_11
, sum(case when substring(b.order_date,6,2)='12' then b.Sales else 0 end) sale_12
from customer a
left join
(
    select a.Customer_ID,a.order_date,b.Sales
    from order_m a
    inner join order_d b on b.order_id=a.Order_ID
    where substring(a.Order_Date,1,4)='2019'
    union ALL
    select a.Customer_ID,a.rtn_Date,-1* a.rtn_amount
    from order_rtn a
    where substring(a.rtn_Date,1,4)='2019'
) b on b.customer_id=a.Customer_ID
group by a.customer_id,a.Customer_Name
order by a.customer_id
```

先取出資料,再進行解析

	ABC customer_id	ABC Customer_Name	123 sale_01	123 sale_02	123 sale_03	123 sale_04	123 sale_05	123 sale_06	123 sale_07
1	AA-10315	Alex Avila	639.3675	0	0	0	0	0	
2	AA-10375	Allen Arnold	0	0	0	0	0	0	1,07
3	AA-10480	Andrew Allen	0	0	2,645.376	0	0	0	
4	AA-10645	Anna Andreadi	0	0	0	0	0	95.58	
5	AB-10015	Aaron Bergman	0	911.6322	0	0	0	600.318	

練習二

同期比(201901-03與201801-03)

```
select '2019/01-03' YM,a.customer_id,a.Customer_Name
, isnull(b.Sales,0) Sales,isnull(b.pre_Sales,0) pre_Sales
, case when isnull(b.Sales,0)=0 and isnull(b.pre_Sales,0)=0 then '0%'
      when isnull(b.pre_Sales,0)=0 then '100%'
      else
        convert(varchar(10),
          convert(decimal(10,2),(isnull(b.Sales,0)-b.pre_Sales)*100.00/(b.pre_Sales))
        ) + '%'
      end g_rate
from customer a
left join
(
  select a.Customer_ID,sum(a.Sales) Sales,sum(a.pre_Sales) pre_Sales
  from
  (
    select a.Customer_ID,b.Sales,0 pre_Sales
    from order_m a
    inner join order_d b on b.order_id=a.Order_ID
    where substring(a.Order_Date,1,7) between '2019/01' and '2019/03'
    union ALL
    select a.Customer_ID,-1*a.rtn_amount,0 pre_Sales
    from order_rtn a
    where substring(a.rtn_Date,1,7) between '2019/01' and '2019/03'
    union ALL
    select a.Customer_ID,0 Sales,b.Sales pre_Sales
    from order_m a
    inner join order_d b on b.order_id=a.Order_ID
    where substring(a.Order_Date,1,7) between '2018/01' and '2018/03'
    union ALL
    select a.Customer_ID,0 Sales,-1*a.rtn_amount pre_Sales
    from order_rtn a
    where substring(a.rtn_Date,1,7) between '2018/01' and '2018/03'
  ) a
  group by a.Customer_ID
) b on b.customer_id=a.Customer_ID
order by a.customer_id
```

	ABC YM	ABC customer	ABC Customer_Name	123 Sales	123 pre_Sales	ABC g_rate
1	2019/01-03	AA-10315	Alex Avila	639.3675	0	100%
2	2019/01-03	AA-10375	Allen Arnold	0	0	0%
3	2019/01-03	AA-10480	Andrew Allen	2,645.376	0	100%
4	2019/01-03	AA-10645	Anna Andreadi	0	0	0%
5	2019/01-03	AB-10015	Aaron Bergman	911.6322	0	100%
6	2019/01-03	AB-10060	Adam Bellavance	0	0	0%
7	2019/01-03	AB-10105	Adrian Barton	0	0	0%
8	2019/01-03	AB-10150	Aimee Bixby	0	207.6	-100.00%
9	2019/01-03	AB-10165	Alan Barnes	0	0	0%
10	2019/01-03	AB-10255	Alejandro Ballentine	306.492	0	100%
11	2019/01-03	AB-10600	Ann Blume	2,062.704	2,242.29	-8.01%

練習三

文字資料拆解 CHARINDEX (expressionToFind , expressionToSearch [, start_location])

```
create procedure city_sp(@city_str nvarchar(max))
```

```
AS
```

```
declare
```

```
@mark_rec      nvarchar(255),      --字串
@mark_symbol    nvarchar(1),        --分隔符號
@mark_value     nvarchar(255),      --取得值
@mark_flag      int;                --位置
```

```
select convert(nvarchar(8),'') city into #temp1
delete from #temp1;
```

```
set @mark_rec=@city_str;
set @mark_symbol=N',';
set @mark_flag=1;
set @mark_flag=CHARINDEX(@mark_symbol, @mark_rec);
```

```
WHILE (@mark_flag <>0)
```

```
BEGIN
```

```
    set @mark_value=SUBSTRING ( @mark_rec , 1 , @mark_flag-1 );
```

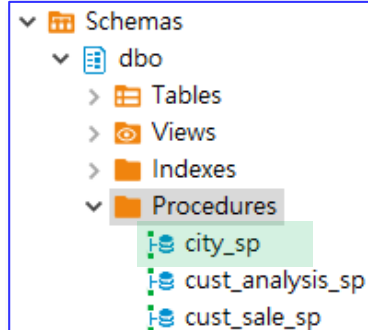
```
    insert into #temp1 (city) select @mark_value;
```

```
    set @mark_rec=SUBSTRING ( @mark_rec , @mark_flag+1,LEN(@mark_rec)) ;
```

```
    set @mark_flag=CHARINDEX(@mark_symbol, @mark_rec);
```

```
END
```

```
execute city_sp N'台北市,新北市,台中市,台南市,東京,胡志明市,';
```



	ABC city	
1	台北市	
2	新北市	
3	台中市	
4	台南市	
5	東京	
6	胡志明市	

SQL指令

```
declare temp1_cursor CURSOR FOR
```

```
select 客戶,購買日期,產品
```

```
from 客戶
```

```
inner join 購買明細
```

```
open temp1_cursor
```

```
fetch next from temp1_cursor
```

```
into @客戶,@購買日期,@產品
```

```
while @@fetch_status = 0
```

```
Begin
```

```
--取得客戶每一筆與上一筆的差距日數
```

```
fetch next from temp1_cursor
```

```
into @客戶,@購買日期,@產品
```

```
end
```

```
close temp1_cursor
```

```
deallocate temp1_cursor
```

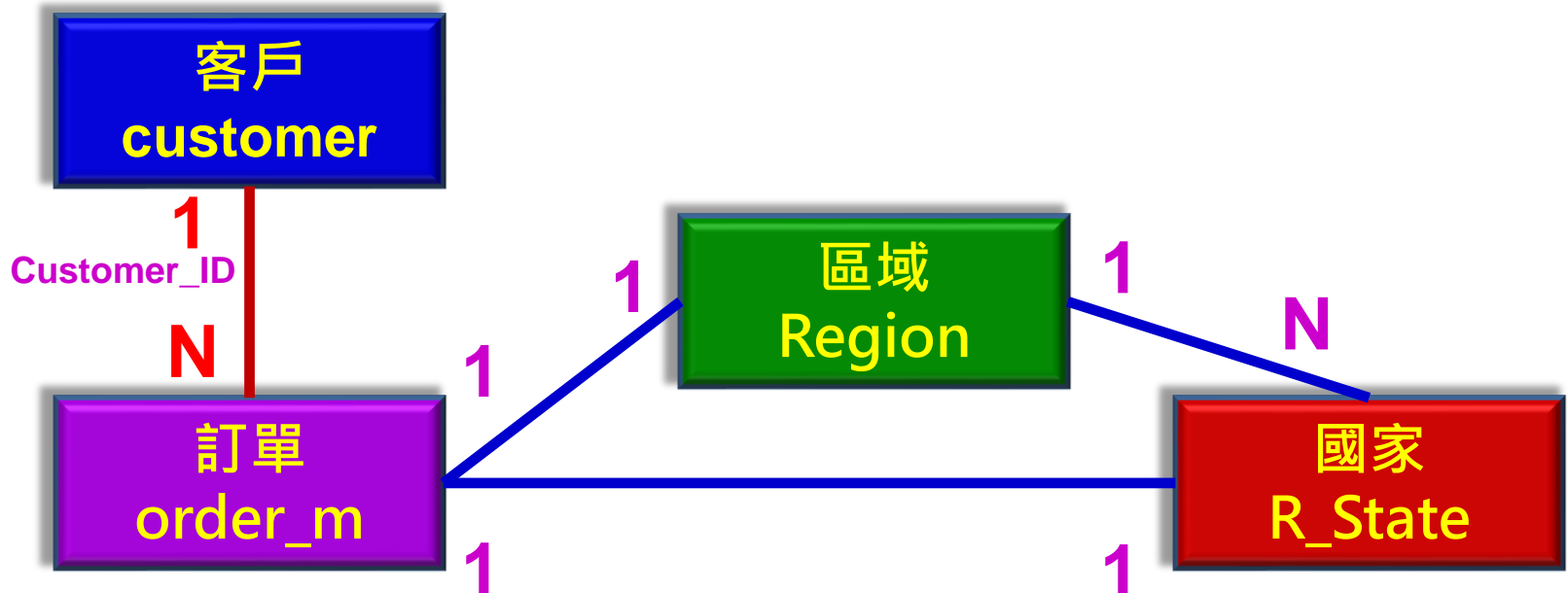
作業二-1/3

客戶編號	客戶名稱
------	------

1.請分析下述條件哪些客戶沒有購買產品(不考慮退貨)

條件

- a.客戶訂單之送貨區域曾經在**北亞**,國家是**Japan**
- b.這些客戶在**201901~201912**沒有購買任何產品
- c.客戶編號不能重複出現(需使用group by)
- d.依客戶編號排序



作業二-2/3

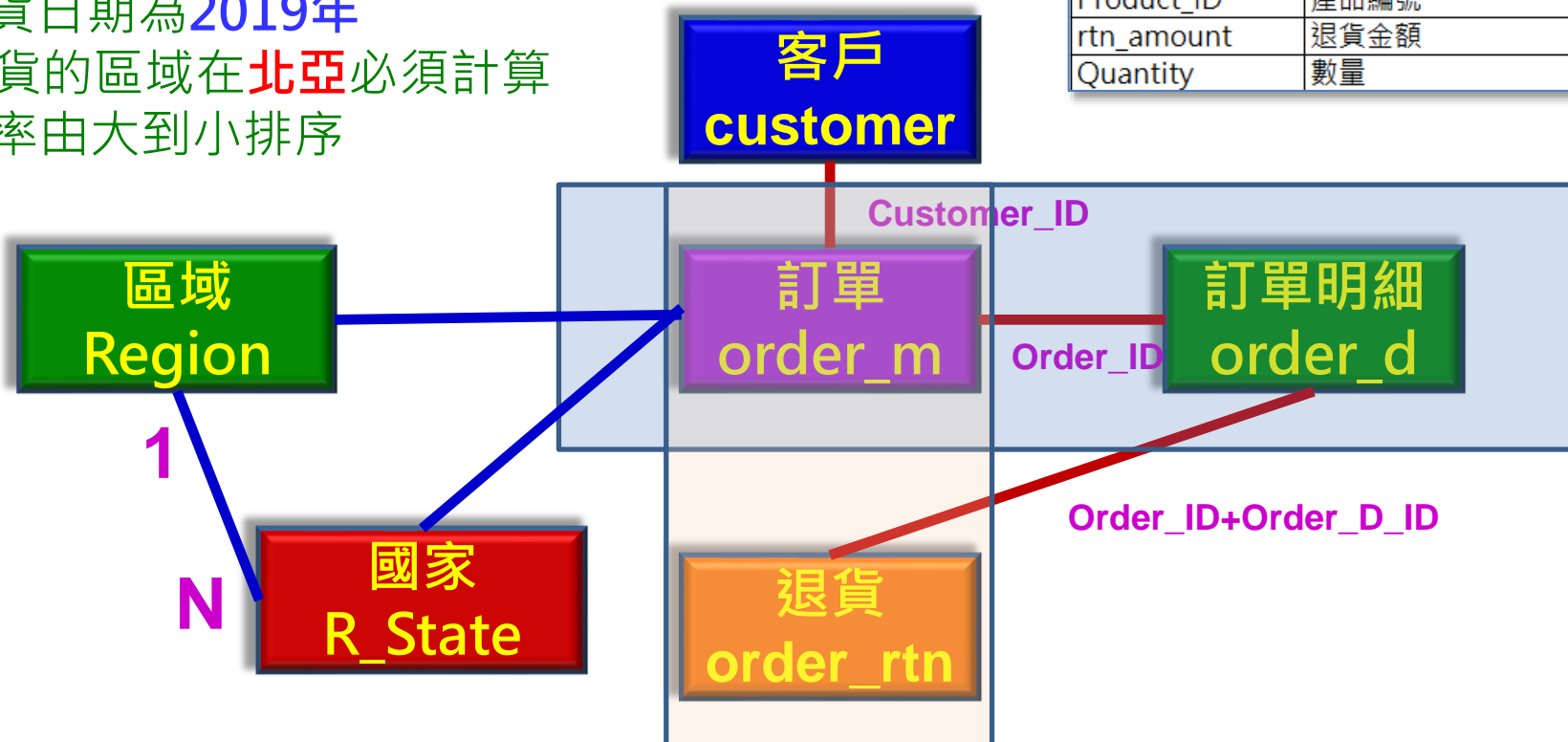
年	區域名稱	國家名稱	合計金額	銷售總額	比率
---	------	------	------	------	----

1.請分析下述條件之銷售佔銷售總額比率

條件

- a.需列出**北亞**所有國家與**北亞**每個國家銷售金額
- b.訂單日期為**2019年**
- c.退貨日期為**2019年**
- d.退貨的區域在**北亞**必須計算
- e.比率由大到小排序

order_rtn	
Column	Description
Order_ID	訂單編號
Order_D_ID	訂單明細編號
Customer_ID	客戶編號
rtn_Date	退貨日期
Product_ID	產品編號
rtn_amount	退貨金額
Quantity	數量



[illegible]



Q&A



東吳大學
Soochow University