**107學年度 資料庫管理系統 作業二**

**107.11.04**

1. 作業要求
   1. 請沿用作業一所建立的資料庫結構，並將作業一所建立的資料清空。
   2. 更改資料表內容

**會員資料表(Member)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **M\_ID** | **account** | **passwd** | **M\_name** | **birthday** | **M\_tel** | **M\_address** |
| 1 | yun0001 | pwd001 | Cairns | 1994/01/08 | 02-2222-2222 | No.1, Sec. 3, University Road, Douliu City, Yunlin County |
| 2 | yun0002 | pwd002 | Tim | 1955/10/15 | 03-3333-3333 | No.2, Sec. 3, University Road, Douliu City, Yunlin County |
| 3 | yun0003 | pwd003 | Genie | 1993/06/09 | 04-4444-4444 | No.3, Sec. 3, University Road, Douliu City, Yunlin County |
| 4 | yun0004 | pwd004 | Kevin | 1992/04/15 | 05-5555-5555 | No.4, Sec. 3, University Road, Douliu City, Yunlin County |
| 5 | yun0005 | pwd005 | Angela | 1965/04/20 | 06-6666-6666 | No.5, Sec. 3, University Road, Douliu City, Yunlin County |
| 6 | yun0006 | pwd006 | Edwin | 1990/10/25 | 07-7777-7777 | No.6, Sec. 3, University Road, Douliu City, Yunlin County |
| 7 | yun0007 | pwd007 | Jam | 1978/05/08 | 08-8888-8888 | No.7, Sec. 3, University Road, Douliu City, Yunlin County |
| 8 | yun0008 | pwd008 | Jacob | 1990/08/08 | 02-2222-2233 | No.8, Sec. 3, University Road, Douliu City, Yunlin County |
| 9 | yun0009 | yun0009 | Jay | 1991/11/11 | 03-3333-3344 | No.9, Sec. 3, University Road, Douliu City, Yunlin County |
| 10 | yun0010 | yun0010 | Jane | 1988/10/10 | 05-5533-5555 | No.10, Sec. 3, University Road, Douliu City, Yunlin County |

**書籍資料表(Supplier)**

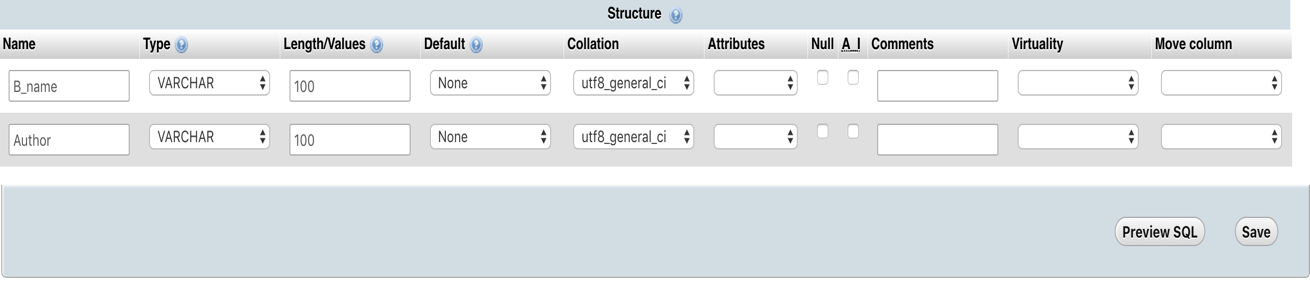
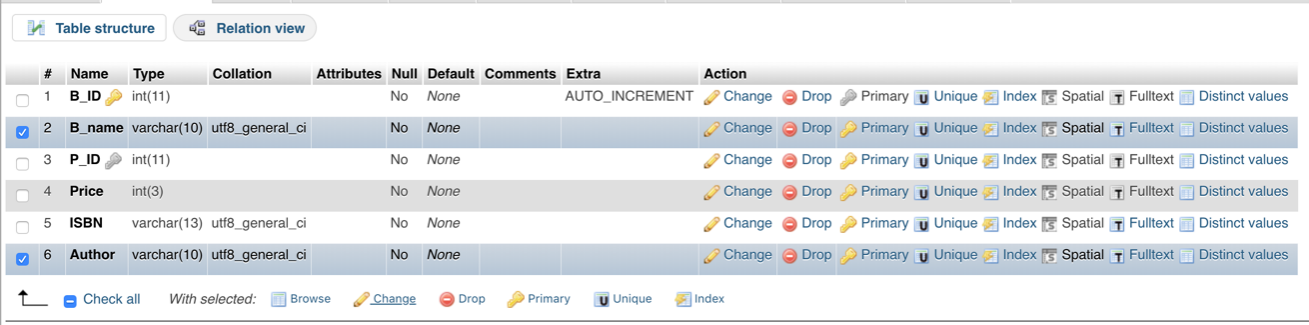
|  |  |  |  |
| --- | --- | --- | --- |
| **P\_Id** | **P\_name** | **P\_tel** | **P\_address** |
| 1 | OREILLY | 05-5555-5555 | No.515, Sec. 2, Yunlin Rd., Douliu City, Yunlin County |
| 2 | Packt Publishing | 03-3333-3333 | No.17, Fuqian Rd., Hualien City, Hualien County |
| 3 | Apress | 04-4444-4444 | No. 99, Sec. 3, Taiwan Boulevard, Xitun Dist., Taichung City |
| 4 | Cisco Systems | 02-2222-2222 | No.1, City Hall Rd., Xinyi District, Taipei City |

**書籍資料(Book)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **B\_ID** | **B\_name** | **P\_ID** | **Price** | **ISBN** | **Author** |
| 1 | Designing with Data | 1 | 458 | 9789864769247 | Rochelle King |
| 2 | Programming iOS 12 | 1 | 2,533 | 9781492044635 | Matt Neuburg |
| 3 | Azure Analytics | 1 | 537 | 9789864769209 | Zoiner Tejada |
| 4 | Mastering Python for Networking and Security | 2 | 1,366 | 9781788992510 | Jose Manuel Ortega |
| 5 | Python Deep Learning Projects | 2 | 1,507 | 9781788997096 | Rahul Kumar, Matthew Lamons |
| 6 | Mastering PostgreSQL 11 | 2 | 1,366 | 9781789537819 | Hans-Jurgen Schonig |
| 7 | Applied Natural Language Processing with Python | 3 | 1,201 | 9781484237328 | Taweh Beysolow II |
| 8 | Python Descriptors | 3 | 1,482 | 9781484237267 | Jacob Zimmerman |
| 9 | Building Scalable Cisco Networks | 4 | 199 | 9781578702282 | Catherine Paquet, Diane Teare |
| 10 | Managing Cisco Network Security | 4 | 1,050 | 9781578701032 | Mike Wenstrom |

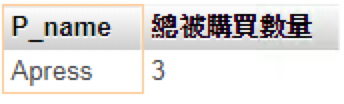
\*請各組使用Update、Delete、Insert或是清空資料表的SQL指令完成操作。

\*Supplier.P\_name、Book.B\_name、Book.Author的長度改為varchar(100)，Supplier.P\_tel、Member.Mtel的長度改為varchar(15)，方法如圖步驟。

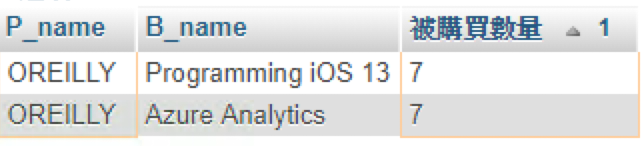


將原先已建立的資料**全部刪除**並**再新增資料**，更改後的資料表內容**須符合以下條件：**

1. 至少10筆購買記錄，且每個會員至少有一筆購買記錄、每項產品至少出現在一筆購買明細中。
2. 至少四位會員單筆購買記錄有買三項以上的產品。
3. 針對此5張資料表，在上述3種指令中(Update、Delete、Insert)其中2種需使用1次以上，應用方式請各組自行發揮。
   1. 請依照下列指定的情況，利用SQL指令完成要求：
      1. 針對某一個出版社，列出該出版社的書籍總被購買數量



* + 1. 針對某一個出版社，列出該出版社名稱所有書籍，並只顯示被購買數量大於2的書籍。最後以被購買數量由小到大排序。。



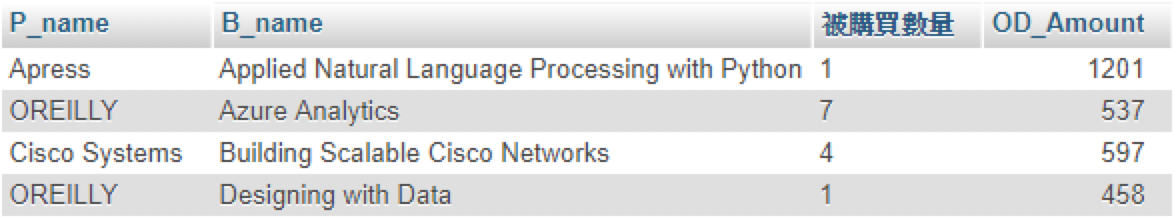
* + 1. 針對某一個出版社，列出該出版社名稱，以及其所有產品之被購買數量的總平均。



* + 1. 請列出所有會員資訊以及他們的總購買次數，並且依照次數由高到低排序。



* 1. 請依照下列指定的情況，利用SQL指令完成要求：
     1. 請建立一視觀表(View)命名為「View\_Order\_Count」，顯示各出版社、其所有書籍名稱和各書籍被購買數量以及書籍的銷售總額。



* + 1. 將書籍” Programming iOS 12”之品名改為” Programming iOS 13”，並針對視觀表View\_Order\_Count查詢OREILLY出版社之所有書籍。



* + 1. 針對視觀表View\_Order\_Count，查詢哪些出版社的銷售金額加總後，低於$5000，並顯示出版社名稱及總被購買次數及總金額



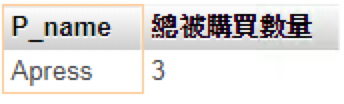
1. 作業規定
   1. 作業型態：小組作業，請延續作業一的分組
   2. 繳交方式：由組長代表上傳至e-Learning網路學園
   3. 繳交期限：107/12/03(一)
   4. 對於檔案匯入、SQL語法或作業規定有問題者，可向吳俊逸助教諮詢：[m10623017@gemail.yuntech.edu.tw](mailto:m10623017@gemail.yuntech.edu.tw)
   5. 嚴禁抄襲。發現抄襲者，無論抄襲或被抄襲，視情節輕重扣分或以零分計算。
   6. 遲交一天扣10分。
2. 繳交內容
   1. 資料庫SQL檔
3. 將已建立的資料庫以「.sql」的檔案格式匯出
4. 命名規則：組長學號Hw2.sql (ex: B10323001Hw2.sql)
   1. 書面說明PDF檔
      1. 內含每一條SQL指令及執行畫面。依照題號標示清楚，並以條列式簡述該SQL功能，最後附上該題的執行畫面。

Ex： (1) 針對某一個出版社，列出該出版社的書籍總被購買次數

SELECT \*

FROM …

WHERE ….

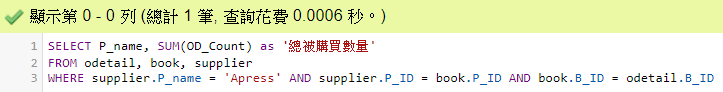


* + 1. 命名規則：組長學號Hw2.pdf (ex: B10323001Hw2.pdf)

1. 評分標準
2. 資料內容10%
3. 第3到第4題每小題10%
4. 書面內容20%

1.請依照下列指定的情況，利用SQL指令完成要求：

(1)針對某一個出版社，列出該出版社的書籍總被購買數量





1.取得「P\_name (supplier)」跟「總購買數量OD\_Count(odetail)」資料

2.從odetail, book, supplier選取資料表

3.從「P\_name」中找”Apress資料&串聯supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

/

Code:

SELECT P\_name, SUM(OD\_Count) as '總被購買數量'

FROM odetail, book, supplier

WHERE supplier.P\_name = 'Apress' AND supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

(2)針對某一個出版社，列出該出版社名稱所有書籍，

並只顯示被購買數量大於2的書籍。最後以被購買數量由小到大排序。



1.取得「P\_name (supplier)」跟「被購買數量OD\_Count(odetail)」資料

2.從odetail, book, supplier選取資料表

3.從「supplier.P\_name」中找” 'OREILLY'資料&串聯supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

4.設定條件為「被購買數量>2」

5.聯集第二個table…

(5~最後)重複上列動作

/

Code

SELECT P\_name, B\_name, SUM(OD\_Count) as '被購買數量'

FROM supplier, odetail, book

WHERE supplier.P\_name='OREILLY' AND book.B\_name='Designing with Data' AND supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

HAVING 被購買數量>2

UNION

SELECT P\_name, B\_name, SUM(OD\_Count) as '被購買數量'

FROM supplier, odetail, book

WHERE supplier.P\_name='OREILLY' AND book.B\_name='Programming iOS 13' AND supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

HAVING 被購買數量>2

UNION

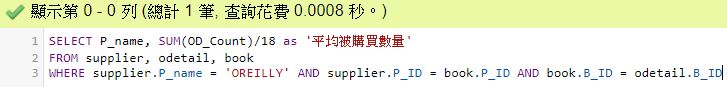
SELECT P\_name, B\_name, SUM(OD\_Count) as '被購買數量'

FROM supplier, odetail, book

WHERE supplier.P\_name='OREILLY' AND book.B\_name='Azure Analytics' AND supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

HAVING 被購買數量>2

(3)針對某一個出版社，列出該出版社名稱，以及其所有產品之被購買數量的總平均。





1.取得「P\_name (supplier)」跟「平均被購買數量OD\_Count/18筆訂單」資料

2.從odetail, book, supplier選取資料表

3.從「P\_name」中找” OREILLY”資料&串聯supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

/

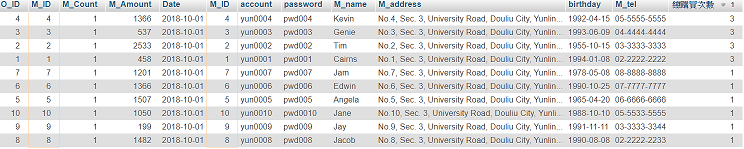
Code

SELECT P\_name, SUM(OD\_Count)/18 as '平均被購買數量'

FROM supplier, odetail, book

WHERE supplier.P\_name = 'OREILLY' AND supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

(4)請列出所有會員資訊以及他們的總購買次數，並且依照次數由高到低排序。



1.取得所有會員及「總購買次數」資料

2.從orderhistory,member選取資料表

3.從「orderhistory.M\_ID」中找” member.M\_ID”資料&設定各會員ID

4.聯集其他會員

(4~最後)重複上列動作

5.ORDER BY 總購買次數由高到低排序

/

Code:

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 1

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 2

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 3

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 4

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 5

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 6

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 7

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 8

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 9

UNION

SELECT \*, COUNT(O\_ID) AS '總購買次數'

FROM orderhistory,member

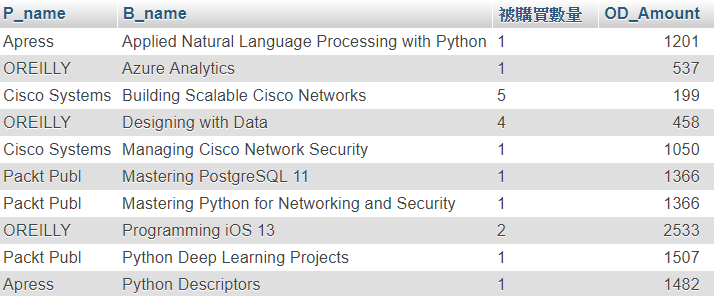
WHERE orderhistory.M\_ID = member.M\_ID AND member.M\_ID = 10

ORDER BY 總購買次數 DESC

4.請依照下列指定的情況，利用SQL指令完成要求：

(1)請建立一視觀表(View)命名為「View\_Order\_Count」，顯示各出版社、其所有書籍名稱和各書籍被購買數量以及書籍的銷售總額。





1.建立一個檢視表(VIEW)名稱為view\_order\_count

2.取得「P\_name」「B\_name」跟「被購買數量」跟「OD\_Amount」資料

3.從supplier, odetail, book選取資料

4.設定條件為supplier.P\_ID = book.P\_ID 跟 book.B\_ID = odetail.B\_ID

5.將B\_name設為群組

/

Code:

CREATE view view\_order\_count AS

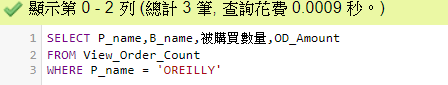
SELECT P\_name, B\_name, SUM(OD\_Count) as '被購買數量', OD\_Amount

FROM supplier, odetail, book

WHERE supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

GROUP BY B\_name

(2)將書籍” Programming iOS 12”之品名改為” Programming iOS 13”，並針對視觀表View\_Order\_Count查詢OREILLY出版社之所有書籍。





1.更新資料'Programming iOS 12'改為’Programming iOS 13'

2.取得「P\_name」「B\_name」跟「被購買數量」跟「OD\_Amount」資料

3.從View\_Order\_Count選取資料

4. 設定條件為P\_name = 'OREILLY'

/

Code

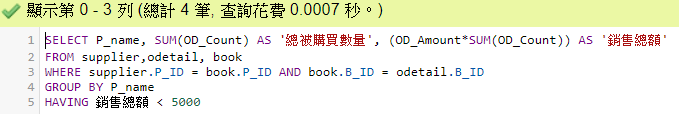
UPDATE book SET B\_name='Programming iOS 13' WHERE B\_name='Programming iOS 12';

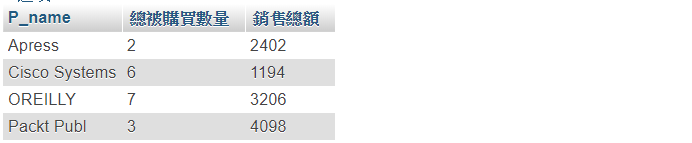
SELECT P\_name,B\_name,被購買數量,OD\_Amount

FROM View\_Order\_Count

WHERE P\_name = 'OREILLY'

(3)針對視觀表View\_Order\_Count，查詢哪些出版社的銷售金額加總後，低於$5000，並顯示出版社名稱及總被購買次數及總金額





1.取得「P\_name」跟「總被購買數量」跟「銷售總額」資料

2.從supplier,odetail, book選取資料

3.設定條件為supplier.P\_ID = book.P\_ID 跟 book.B\_ID = odetail.B\_ID

4.把P\_name設為群組

5.將銷售總額 < 5000設為條件

/

SELECT P\_name, SUM(OD\_Count) AS '總被購買數量', (OD\_Amount\*SUM(OD\_Count)) AS '銷售總額'

FROM supplier,odetail, book

WHERE supplier.P\_ID = book.P\_ID AND book.B\_ID = odetail.B\_ID

GROUP BY P\_name

HAVING 銷售總額 < 5000