



客戶購買系統

作者 吳彥瑾

作品名稱：客戶購買系統

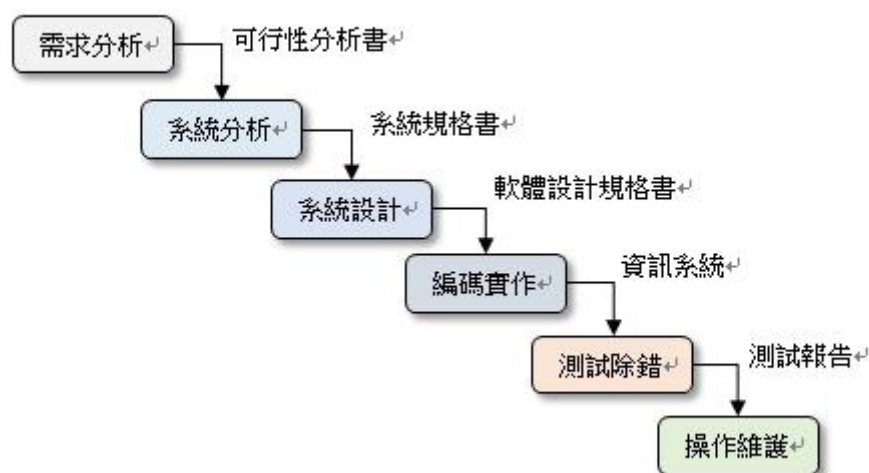
一、說明

本專題使用 Python 程式來連接 SQL Server 資料庫+SQL Server 管理工具，製作一個簡易「客戶購買系統」，以便讓讀者深入了解客戶購買紀錄的過程。讓讀者了解如何在「客戶購買系統」操作環境中，撰寫各種 SQL 指令來查詢，了解每一位客戶的購買情況。

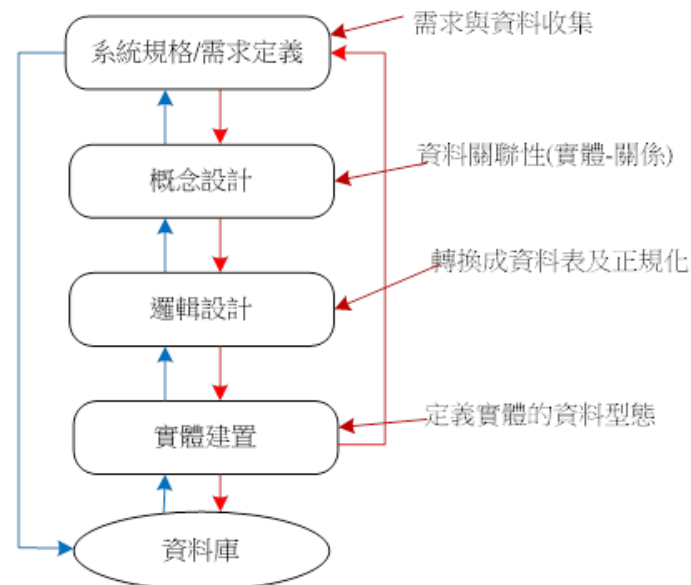
1. 下載及安裝 Anaconda 的 Python 工具進行操作。
2. 使用 sql server management studio(SSMS)工具進行操作。
3. 輸入 pip install pyodbc 指令安裝套件。
4. 利用 SQL Server 資料庫管理系統建立「客戶購買系統」資料庫，並且建立三個資料表，分別為：客戶表、產品表及購買表三種資料紀錄。

二、相關文章

開發模式採用「瀑布模式(Waterfall Model)」，又稱全功能模式(Fully Functional Approach)，是於 1970 年由溫斯頓·W·羅伊斯所提出。而瀑布模式就是一般所說的「系統發展生命週期(System Development Life Cycle, SDLC)」。由於此模式從圖形的外觀來看，各階段依序就像是梯形瀑布順勢而下，稱為瀑布模式(Waterfall Model)。



一個功能完整及有效率的資訊系統，它的幕後最大工程，就是資料庫系統之協助。因此，在設計資料庫時必須經過一連串有系統的規劃及設計。在開發資料庫系統時，首要的工作是先做資料庫的分析，在做資料庫分析工作時，需要先與使用者進行需求訪談的作業，藉著訪談的過程來了解使用者對資料庫的需求，以便讓系統設計者來設計企業所需要的資料庫。



三、實作

實作前請先至 SSMS 建立客戶表、產品表及購買表三個資料表，然後對建立好的資料表按"右鍵/選取前 1000 個資料列"皆可查詢建立結果出來。

SQLQuery1.sql - DESKTOP-HFCV7Q\SQLSERVER2019,客戶購買訂單 (DESKTOP-HFCV7Q\User (61)) - Microsoft SQL Server Management Studio

物件總管

- 客戶購買訂單
 - 資料庫圖表
 - 資料表
 - 系統資料表
 - FileTable
 - 外部資料表
 - 圖形資料表
 - dbo.客戶表
 - 資料行
 - 編號 (PK, char(5), 非 Null)
 - 姓名 (nvarchar(10), 非 Null)
 - 電話 (nvarchar(15), Null)
 - 城市 (nvarchar(10), Null)
 - 區域 (nvarchar(10), Null)
 - 索引鍵
 - 條件約束
 - 觸發程序
 - 索引
 - 統計資料
 - dbo.產品表
 - 資料行
 - 品號 (PK, char(5), 非 Null)
 - 品名 (nvarchar(10), 非 Null)
 - 定價 (int, Null)
 - 索引鍵
 - 條件約束
 - 觸發程序
 - 索引
 - 統計資料
 - dbo.購買表
 - 資料行
 - 編號 (PK, FK, char(5), 非 Null)
 - 品號 (PK, FK, char(5), 非 Null)
 - 數量 (int, 非 Null)
 - 索引鍵

SQLQuery1.sql - FCSV7Q\User (61)

```

1  /***** SSMS 中 SelectTopNRows 命令的指
2  SELECT TOP (1000) [編號]
3      ,[姓名]
4      ,[電話]
5      ,[城市]
6      ,[區域]
7  FROM [客戶購買訂單].[dbo].[客戶表]

```

165 %

編號	姓名	電話	城市	區域
1	A01	陳甲	02-15813604	新北市 板橋區
2	A02	陳乙	02-72704530	新北市 三重區
3	A03	陳丙	02-60474426	新北市 永和區
4	A04	陳丁	02-17610465	新北市 新店區
5	A05	陳戊	02-62201536	新北市 淡水區
6	A06	陳己	02-48026016	新北市 樹林區
7	A07	陳庚	02-50285385	新北市 三芝區
8	A08	陳辛	02-77510178	新北市 石門區
9	A09	陳壬	02-42368605	新北市 八里區
10	A10	陳癸	02-32748358	新北市 金山區
11	B01	王子	02-20884876	台北市 松山區
12	B02	王丑	02-37888272	台北市 大安區
13	B03	王寅	02-11231412	台北市 大同區
14	B04	王卯	02-18356618	台北市 中山區
15	B05	王辰	02-43773288	台北市 內湖區
16	B06	王巳	02-47174482	台北市 南港區
17	B07	王午	02-43083111	台北市 士林區

已成功執行查詢。

DESKTOP-HFCV7Q

就緒 第 1 行 第 1 欄

23°C 晴時多雲

搜尋

客戶購買系統之架構圖是由四個子系統組合而成。

1. 客戶管理子系統：提供新增、修改、刪除及查詢功能。
2. 產品管理子系統：提供新增、修改、刪除及查詢功能。
3. 購買作業子系統：提供加購、退貨及基本查詢作業。
4. 各項查詢作業子系統：提供各項進階查詢功能。

<p>=====客戶購買系統=====</p> <p>1. 「客戶」管理系統</p> <p>2. 「產品」管理系統</p> <p>3. 「購買」管理系統</p> <p>4. 「查詢」購買記錄</p> <p>5. 結束系統</p>			
<p>請選擇功能清單：1</p> <p>===「客戶」管理系統===</p> <p>1. 新增客戶記錄</p> <p>2. 修改客戶記錄</p> <p>3. 刪除客戶記錄</p> <p>4. 查詢客戶記錄</p> <p>5. 回主畫面</p>	<p>請選擇功能清單：2</p> <p>===「產品」管理系統===</p> <p>1. 新增產品記錄</p> <p>2. 修改產品記錄</p> <p>3. 刪除產品記錄</p> <p>4. 查詢產品記錄</p> <p>5. 回主畫面</p>	<p>請選擇功能清單：3</p> <p>===「購買」管理系統===</p> <p>1. 加購程記錄</p> <p>2. 退貨程記錄</p> <p>3. 查詢購買記錄</p> <p>4. 回主畫面</p>	<p>請選擇功能清單：4</p> <p>===「查詢」購買記錄===</p> <p>1. 查詢各位客戶購買種類數</p> <p>2. 查詢每種產品購買數</p> <p>3. 查詢每位客戶購買平均數量</p> <p>4. 查詢每種產品平均購買數</p> <p>5. 查詢客戶購買記錄資料</p> <p>6. 查詢必推銷品(全部選)</p> <p>7. 回主畫面</p>

程式碼：

```
import pyodbc
import sys
driver="{ODBC Driver 17 for SQL Server}"
server="請輸入 SSMS 的伺服器"
database="客戶購買訂單"
username="sa"
password="輸入自己的密碼"
conn=pyodbc.connect("DRIVER=" + driver
                    + ";SERVER=" + server
                    + ";DATABASE=" + database
                    + ";UID=" + username
                    + ";PWD=" + password)

# 「客戶」管理系統之主畫面
#-----
def Staff_Manager():
    print("===「客戶」管理系統===")
    print("1. 新增客戶記錄")
    print("2. 修改客戶記錄")
    print("3. 刪除客戶記錄")
    print("4. 查詢客戶記錄")
```

```

print("5. 回主畫面")
n=eval(input("請選擇「客戶」功能清單："))
if n==1:
    Insert_Staff()
elif n==2:
    Update_Staff()
elif n==3:
    Delete_Staff()
elif n==4:
    Query_Staff()
elif n==5:
    Main_Menu()
else:
    print("請選擇 1~5 項功能")

def Check_Sid(Sid): #檢查編號是否存在於客戶表中之副程式
    SQLcmd="select * from 客戶表 where 編號=' {}'.format(Sid)
    cursor=conn.execute(SQLcmd)
    return cursor.fetchone() #若無記錄則傳回 None

def Insert_Staff(): #新增客戶記錄
    Sid=input("編號：")
    if Check_Sid(Sid)!=None:
        print("編號:{}重複了".format(Sid))
        return
    Sname=input("姓名：")
    Tel=input("電話：")
    City=input("城市：")
    Area=input("區域：")
    SQLcmd="INSERT INTO 客戶表 VALUES
(' {}', ' {}', ' {}', ' {}', ' {}')".format(Sid, Sname, Tel, City, Area)
    conn.execute(SQLcmd)
    conn.commit()
    print("新增客戶記錄！")
    Staff_Manager() #返回到「客戶」管理系統之主畫面

def Update_Staff(): #修改客戶記錄
    Sid=input("編號：")

```

```

if Check_Sid(Sid)==None:
    print(" 查無此編號:{} ".format(Sid))
    return
Sname=input(" 姓名 : ")
Tel=input(" 電話 : ")
City=input(" 城市 : ")
Area=input(" 區域 : ")
SQLcmd="UPDATE 客戶表 SET 姓名=' {} ', 電話=' {} ', 城市=' {} ', 區域=' {} '
Where 編號=' {} ' ".format(Sname, Tel, City, Area, Sid)
conn.execute(SQLcmd)
conn.commit()
print(" 更新客戶記錄 ! ")
Staff_Manager() #返回到「客戶」管理系統之主畫面

def Delete_Staff(): #刪除客戶記錄
    Sid=input(" 編號 : ")
    if Check_Sid(Sid)==None:
        print(" 查無此編號:{} ".format(Sid))
        return
    SQLcmd="Delete From 客戶表 WHERE 編號=' {} ' ".format(Sid)
    conn.execute(SQLcmd)
    conn.commit()
    print(" 刪除記錄成功 ! ")
    Staff_Manager() #返回到「客戶」管理系統之主畫面

def Query_Staff(): #查詢客戶記錄
    SQLcmd="select * from 客戶表"
    Record=conn.execute(SQLcmd)
    listStaff=list(Record.fetchall())
    print(" 編號      姓名      電話      城市      區域 ")
    for row in listStaff:
        for col in row:
            print(col, end="      ")
        print()
    Record.close()
    Staff_Manager() #返回到「客戶」管理系統之主畫面

#「產品」管理系統之主畫面

```

```

#-----
def Product_Manager(): #「產品」管理系統之主畫面
    print("===「產品」管理系統===")
    print("1. 新增產品記錄")
    print("2. 修改產品記錄")
    print("3. 刪除產品記錄")
    print("4. 查詢產品記錄")
    print("5. 回主畫面")
    n=eval(input("請選擇「產品」功能清單："))
    if n==1:
        Insert_Product() #新增產品記錄
    elif n==2:
        Update_Product() #修改產品記錄
    elif n==3:
        Delete_Product() #刪除產品記錄
    elif n==4:
        Query_Product() #查詢產品記錄
    elif n==5:
        Main_Menu() #回主畫面
    else:
        print("請選擇 1~5 項功能")

def CheckProduct_NO(No):
    SQLcmd="select * from 產品表 where 品號=' {}'.format(No)
    cursor=conn.execute(SQLcmd)
    return cursor.fetchone() #若無記錄則傳回 None

def Insert_Product(): #新增產品記錄
    No=input("品號：")
    if CheckProduct_NO(No)!=None:
        print("編號:{}重複了".format(No))
        return
    Cname=input("品名：")
    Credits=input("定價：")
    SQLcmd="INSERT INTO 產品表 VALUES
(' {}', ' {}'.format(No, Cname, Credits)
    conn.execute(SQLcmd)
    conn.commit()

```

```

print("新增產品記錄!")
Product_Manager() #返回到「產品」管理系統之主畫面

def Update_Product(): #修改產品記錄
    No=input("品號:")
    if CheckProduct_NO(No)==None:
        print("查無此品號:{}".format(No))
        return
    Cname=input("品名:")
    Credits=input("定價:")
    SQLcmd="UPDATE 產品表 SET 品名=' {}',定價=' {}' Where 品號=' {}' ".format(Cname,Credits,No)
    conn.execute(SQLcmd)
    conn.commit()
    print("更新產品記錄!")
    Product_Manager() #返回到「產品」管理系統之主畫面

def Delete_Product(): #刪除產品記錄
    No=input("品號:")
    if CheckProduct_NO(No)==None:
        print("查無此品號:{}".format(No))
        return
    SQLcmd="Delete From 產品表 WHERE 品號=' {}' ".format(No)
    conn.execute(SQLcmd)
    conn.commit()
    print("刪除記錄成功!")
    Product_Manager() #返回到「產品」管理系統之主畫面

def Query_Product(): #查詢產品記錄
    SQLcmd="select * from 產品表"
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print("品號    品名    定價")
    for row in listProduct:
        for col in row:
            print(col, end="    ")
        print()
    Record.close()

```



```

Product_Manager() #返回到「產品」管理系統之主畫面

#「購買」管理系統之主畫面
#-----
def Sales_Selection(): #「購買」管理系統之主畫面
    print("===「購買」管理系統===")
    print("1. 加購程記錄")
    print("2. 退貨程記錄")
    print("3. 查詢購買記錄")
    print("4. 回主畫面")
    n=eval(input("請選擇「購買」管理清單："))
    if n==1:
        Insert_Sales() #加購程記錄
    elif n==2:
        Delete_Sales() #退貨程記錄
    elif n==3:
        Query_Sales() #查詢購買記錄
    elif n==4:
        Main_Menu() #回主畫面
    else:
        print("請選擇 1~5 項功能")

def CheckSales_NO(Sid, No):
    SQLcmd="select * from 購買表 where 編號=' {}' and 品號
=' {}' ".format(Sid, No)
    cursor=conn.execute(SQLcmd)
    return cursor.fetchone() #若無記錄則傳回 None

def Insert_Sales(): #加購程記錄
# Query_Product_2() #查詢目前的開產品
    print("=====請記錄購買資料=====")
    Sid=input("編號：")
    No=input("品號：")
    if CheckSales_NO(Sid, No)!=None:
        print("品號:{}重複選了".format(No))
        return
    SQLcmd=" INSERT INTO 購買表 VALUES ( ' {}', ' {}', ' {}' )".format(Sid, No, 0)
    conn.execute(SQLcmd)

```

```

conn.commit()
print("記錄購買成功!")
Sales_Selection() #返回到「購買」管理系統之主畫面

def Delete_Sales(): #退貨程記錄
# Query_Product_2() #查詢目前的開產品
print("====請退還====")
Sid=input("編號:")
No=input("品號:")
if CheckSales_NO(Sid,No)==None:
    print("查無此品號:{}".format(No))
    return
SQLcmd="Delete From 購買表 WHERE 編號='{}' and 品號
='{}' ".format(Sid,No)
conn.execute(SQLcmd)
conn.commit()
print("退貨成功!")
Sales_Selection() #返回到「購買」管理系統之主畫面

def Query_Sales(): #查詢購買記錄
Sid=input("編號:")
SQLcmd="select A.編號,A.姓名,C.品號,品名,定價 "
SQLcmd=SQLcmd + "from 客戶表 AS A,購買表 AS B,產品表 AS C "
SQLcmd=SQLcmd + "Where A.編號=B.編號 and B.品號=C.品號 "
SQLcmd=SQLcmd + "And A.編號='{}' ".format(Sid)
Record=conn.execute(SQLcmd)
listProduct=list(Record.fetchall())
print("編號    姓名    品號    品名    定價")
for row in listProduct:
    for col in row:
        print(col, end="    ")
    print()
Record.close()
Sales_Selection() #返回到「購買」管理系統之主畫面

# 「查詢」購買記錄系統之主畫面
#-----
def Query_Product_Record():

```

```

print("=== 「查詢」購買記錄===")
print("1. 查詢各位客戶購買種類數")
print("2. 查詢每種產品購買數")
print("3. 查詢每位客戶購買平均數量")
print("4. 查詢每種產品平均購買數")
print("5. 查詢客戶購買記錄資料")
print("6. 查詢必推銷品(全部選)")
print("7. 回主畫面")
n=eval(input("請選擇「查詢」購買清單："))
if n==1:
    Query1()      #呼叫「查詢各位客戶購買種類數」
elif n==2:
    Query2()      #呼叫「查詢每種產品購買數」
elif n==3:
    Query3()      #呼叫「查詢每位客戶購買平均數量」
elif n==4:
    Query4()      #呼叫「查詢每種產品平均購買數」
elif n==5:
    Query5()      #呼叫「查詢客戶購買記錄資料」
elif n==6:
    Query6()      #呼叫「查詢必推銷品(全部選)」
elif n==7:
    Main_Menu()   #呼叫「回主畫面」
else:
    print("請選擇 1~7 項功能")

def Query1():    #定義「查詢各位客戶購買種類數」之副程式
    SQLcmd="SELECT A. 編號, 姓名, Count(*) AS 購買種類數 "
    SQLcmd=SQLcmd + "FROM 客戶表 AS A, 購買表 AS B "
    SQLcmd=SQLcmd + "Where A. 編號=B. 編號 "
    SQLcmd=SQLcmd + "GROUP BY A. 編號, 姓名"
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print("編號    姓名    購買種類數")
    print("-----")
    for row in listProduct:
        for col in row:
            print(col, end="    ")

```

```

    print()
Record.close()
Query_Product_Record()    #返回到查詢購買記錄主畫面

def Query2(): #定義「查詢每種產品購買數」之副程式
    SQLcmd="SELECT A. 品號, 品名, Count(*) AS 購買數 "
    SQLcmd=SQLcmd + "FROM 產品表 AS A, 購買表 AS B "
    SQLcmd=SQLcmd + "Where A. 品號=B. 品號 "
    SQLcmd=SQLcmd + "GROUP BY A. 品號, 品名"
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print(" 品號    品名    購買數")
    print("-----")
    for row in listProduct:
        for col in row:
            print(col, end="    ")
        print()
    Record.close()
    Query_Product_Record()    #返回到查詢購買記錄主畫面

def Query3(): #定義「查詢每位客戶購買平均數量」之副程式
    SQLcmd="SELECT A. 編號, 姓名, Count(*) AS 選科目數, AVG(數量) AS 購買平均
數量 "
    SQLcmd=SQLcmd + "FROM 客戶表 AS A, 購買表 AS B "
    SQLcmd=SQLcmd + "Where A. 編號=B. 編號 "
    SQLcmd=SQLcmd + "GROUP BY A. 編號, 姓名"
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print(" 編號    姓名    種類    平均數量")
    print("-----")
    for row in listProduct:
        for col in row:
            print(col, end="    ")
        print()
    Record.close()
    Query_Product_Record()    #返回到查詢購買記錄主畫面

```

```

def Query4(): #定義「查詢每種產品平均購買數」之副程式
    SQLcmd="SELECT A. 品號, 品名, Count(*) AS 選修人數, AVG(數量) AS 平均購買
    數 "
    SQLcmd=SQLcmd + "FROM 產品表 AS A, 購買表 AS B "
    SQLcmd=SQLcmd + "Where A. 品號=B. 品號 "
    SQLcmd=SQLcmd + "GROUP BY A. 品號, 品名"
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print(" 品號    品名    客戶數    平均購買數")
    print("-----")
    for row in listProduct:
        for col in row:
            print(col, end="    ")
        print()
    Record.close()
    Query_Product_Record()    #返回到查詢購買記錄主畫面

```

```

def Query5(): #定義「查詢客戶購買記錄資料」之副程式
    SQLcmd="SELECT A. 編號, 姓名, 品名, 數量 "
    SQLcmd=SQLcmd + "FROM 客戶表 AS A, 購買表 AS B, 產品表 AS C "
    SQLcmd=SQLcmd + "Where A. 編號=B. 編號 And C. 品號=B. 品號 "
    Record=conn.execute(SQLcmd)
    listProduct=list(Record.fetchall())
    print(" 編號    姓名    品名    數量")
    print("-----")
    for row in listProduct:
        for col in row:
            print(col, end="    ")
        print()
    Record.close()
    Query_Product_Record()    #返回到查詢購買記錄主畫面

```

```

def Query6(): #定義「查詢必推銷品」之副程式
    SQLcmd="SELECT 品名 "
    SQLcmd=SQLcmd + "FROM 產品表 As C "
    SQLcmd=SQLcmd + "WHERE NOT EXISTS( "
    SQLcmd=SQLcmd + "SELECT * "

```

```

SQLcmd=SQLcmd + "FROM 客戶表 As A "
SQLcmd=SQLcmd + "WHERE NOT EXISTS( "
SQLcmd=SQLcmd + "SELECT * "
SQLcmd=SQLcmd + "FROM 購買表 As B "
SQLcmd=SQLcmd + "WHERE C. 品號=B. 品號 AND A. 編號=B. 編號))"
Record=conn.execute(SQLcmd)
listProduct=list(Record.fetchall())
print(" 品名")
print("-----")
for row in listProduct:
    for col in row:
        print(col, end="    ")
    print()
Record.close()
Query_Product_Record() #返回到查詢購買記錄主畫面

```

```

def Main_Menu():
    print("====客戶購買系統====")
    print("1. 「客戶」管理系統")
    print("2. 「產品」管理系統")
    print("3. 「購買」管理系統")
    print("4. 「查詢」購買記錄")
    print("5. 結束系統")
    n=eval(input("請選擇功能清單："))
    if n==1:
        Staff_Manager()
    elif n==2:
        Product_Manager()
    elif n==3:
        Sales_Selection()
    elif n==4:
        Query_Product_Record() #呼叫「查詢」購買記錄
    elif n==5:
        sys.exit()

while True:
    Main_Menu() #呼叫主選單畫面

```

conn.close()

執行結果：

1. 客戶管理系統

請選擇功能清單：1 ===「客戶」管理系統=== 1.新增客戶記錄 2.修改客戶記錄 3.刪除客戶記錄 4.查詢客戶記錄 5.回主畫面	1.新增客戶記錄	請選擇「客戶」功能清單：1 編號：C23 姓名：高甲 電話：02-26885888 城市：新北市 區域：三重區 新增客戶紀錄！	B08 王未 02-68882071 台北市 北投區 B09 王申 02-23261535 台北市 信義區 B10 王酋 02-15813604 台北市 中正區 C01 王戌 02-57270453 台北市 萬華區 C02 王亥 02-56047442 台北市 南竿鄉 C23 高甲 02-26885888 新北市 三重區
	2.修改客戶記錄	請選擇「客戶」功能清單：2 編號：C23 姓名：高甲 電話：02-26885888 城市：台北市 區域：士林區 更新客戶紀錄！	B08 王未 02-68882071 台北市 北投區 B09 王申 02-23261535 台北市 信義區 B10 王酋 02-15813604 台北市 中正區 C01 王戌 02-57270453 台北市 萬華區 C02 王亥 02-56047442 台北市 南竿鄉 C23 高甲 02-26885888 台北市 士林區
	3.刪除客戶記錄	請選擇「客戶」功能清單：3 編號：C23	B07 王午 02-43083111 台北市 士林區 B08 王未 02-68882071 台北市 北投區 B09 王申 02-23261535 台北市 信義區 B10 王酋 02-15813604 台北市 中正區 C01 王戌 02-57270453 台北市 萬華區 C02 王亥 02-56047442 台北市 南竿鄉
	4.查詢客戶記錄	請選擇「客戶」功能清單：4	B07 王午 02-43083111 台北市 士林區 B08 王未 02-68882071 台北市 北投區 B09 王申 02-23261535 台北市 信義區 B10 王酋 02-15813604 台北市 中正區 C01 王戌 02-57270453 台北市 萬華區 C02 王亥 02-56047442 台北市 南竿鄉

2. 產品管理系統

請選擇功能清單：2 ===「產品」管理系統=== 1.新增產品記錄 2.修改產品記錄 3.刪除產品記錄 4.查詢產品記錄 5.回主畫面			
請選擇「產品」功能清單：1 品號：A0011 品名：Predator 定價：1500 新增產品記錄！	請選擇「產品」功能清單：2 品號：A0011 品名：Predator 筆電 定價：56000 新增產品記錄！	請選擇「產品」功能清單：3 品號：A0011 刪除記錄成功！	請選擇「產品」功能清單：4
A0006 鍵盤 1000 A0007 滑鼠 1000 A0008 滑鼠墊 700 A0009 耳機 1200 A0010 路由器 4000 A0011 Predator 1500	A0006 鍵盤 1000 A0007 滑鼠 1000 A0008 滑鼠墊 700 A0009 耳機 1200 A0010 路由器 4000 A0011 Predator 56000	A0005 硬碟 2500 A0006 鍵盤 1000 A0007 滑鼠 1000 A0008 滑鼠墊 700 A0009 耳機 1200 A0010 路由器 4000	A0005 硬碟 2500 A0006 鍵盤 1000 A0007 滑鼠 1000 A0008 滑鼠墊 700 A0009 耳機 1200 A0010 路由器 4000

3. 購買管理系統

請選擇功能清單：3 ===「購買」管理系統=== 1. 加購程記錄 2. 退貨程記錄 3. 查詢購買記錄 4. 回主畫面	1.加購程記錄	編號：B10 品號：A0007 記錄購買成功！	編號：B10 編號 姓名 品號 品名 定價 B10 王酋 A0007 滑鼠 1000 B10 王酋 A0010 路由器 4000			
	2.退貨程記錄	編號：B10 品號：A0007 退貨成功！	編號：B10 編號 姓名 品號 品名 定價 B10 王酋 A0010 路由器 4000			
	3.查詢購買記錄	編號：B10	編號：B10 編號 姓名 品號 品名 定價 B10 王酋 A0010 路由器 4000			

4. 查詢購買紀錄

請選擇功能清單：4	請選擇「查詢」購買清單：1		
===「查詢」購買記錄===	編號	姓名	購買種類數
1.查詢各位客戶購買種類數	-----		
2.查詢每種產品購買數	A01	陳甲	1
3.查詢每位客戶購買平均數量	A02	陳乙	1
4.查詢每種產品平均購買數	A03	陳丙	1
5.查詢客戶購買記錄資料	A04	陳丁	1
6.查詢必推銷品(全部選)	A05	陳戊	1
7.回主畫面	A06	陳己	1

四、結論

已經讓我們了解整個系統開發流程，並且也學會如何利用 Python 程式來連接 SQL Server 資料庫，進而開發出一套可以模擬多個客戶購買的系統，以便讓學者深入了解。

五、參考資料

圖解資料庫系統理論-使用 SQL Server 實作