A Project Report On

STOCKS MANAGEMENT

Under the Guidance of

Mr. Satpal Singh

Name	• Darshil Kumar
Class	• XIIA
Roll No	• 14601083

Vanasthali Public School

A-461-462, Mayur Vihar Phase III, Delhi-110096

Contents

Working Description

- To start the main program the User is asked about the Password which is the **Password of SQL** i.e. 123456(in this case)
- After this the program creates the database and the table for our program.
- Then the **menu** appears in which we can choose options for carrying out the mentioned operation.
- The User can view the products which are in the database(Supermarket) using the **View Stocks Option**(1).
- The user can also enter the information (Pno, PName, Brand, MRP, SellingP,Qty) of the product(Only If there is no entry with same Pno) in database using the **Add product option**(2).
- The user can also remove a record using Pno, PName or Brand of the product in database using the **Remove product option**(3).
- The user can also update the Pno, Name, Brand, Mrp, Sellingp, and Qty of the any record using the **Update Products option**(4).
- The user can also search records which are in the Database by Modes(Pno,PName,Brand,Qty).If multiple records found it will show multiple using the **Search Record option**(5).
- The user can also create billing for the sale using **Generate Billing option**(6).In this,first the user can see the stocks available and then later he needs to give the pno and amount of the product to be sold.(However the amount should be in the domain of the Qty of the stocks) and then a billing is generated and then user is asked whether he wants to update the Qty according the sold amount.

Module Imported

- **mysql.connector**
  - To establish a connection between python(front end) and mysql(back end) and carry out the queries needed as per the program.
- **datetime**
  - To capture system date and time and use it to display current date and time in the program.
- **time**
  - To add some delay between parts of program

Coding

```
import mysql.connector as con

import datetime

import time


while True:

    try:

        pswd=input("Enter SQL Password:")


dbobj=con.connect(host="localhost",user="root",password=p
swd,charset='utf8')

        print("Connecting....")

        time.sleep(2)

        break

    except :

        print("***WRONG PASSWORD***")

crsr=dbobj.cursor()



def dbtable():

    crsr.execute("create database if not exists
supermarket")

    crsr.execute("use supermarket")

    crsr.execute("create table if not exists product(PNO
int,PNAME char(20),BRAND char(20),MRP int,SELLINGP
int,QTY int,primary key(PNO))")
```

```
def intro():
    now = datetime.datetime.now()
    print()
    print("_"*85)
    print("""
                                    STOCK MANAGEMENT
{}

    0)EXIT
    1)VIEW STOCKS
    2)ADD PRODUCT
    3)REMOVE PRODUCT
    4)UPDATE PRODUCTS
    5)SEARCH RECORD
    6)GENERATE BILLING
    """.format(now.strftime('%d-%m-%Y %H:%M')))



def intinp(stmnt):
    y=0
    while y==0:
        try:
            x=input(stmnt)
            if bool(int(x))==True:
                y=1
                return int(x)
```

```
            if x=="0":
                y=1
                return int(x)
        except:
            print("****Integer Value Required****")



def viewproduct():
    print()
    print("-"*105)
    crsr.execute("desc product;")
    recs=crsr.fetchall()

print(recs[0][0].ljust(7),recs[1][0].ljust(25),recs[2][0]
.ljust(25),recs[3][0].ljust(10),recs[4][0].ljust(10),recs
[5][0].ljust(15),sep="")
    crsr.execute("select * from product;")
    recs=crsr.fetchall()
    for rec in recs:

print(str(rec[0]).ljust(7),rec[1].ljust(25),rec[2].ljust(
25),str(rec[3]).ljust(10),str(rec[4]).ljust(10),str(rec[5
]).ljust(15),sep="")
```

```
def addproduct():
    print("ADD PRODUCT".center(85,'='))
    try:
        n=intinp("Enter no. of records to be added:")
        for w in range (n):
            try:
                pno=intinp("Enter PNO:")
                pname=input("Enter PNAME:")
                brand=input("Enter BRAND:")
                mrp=intinp("Enter MRP:")
                sellingp=intinp("Enter SELLINGP:")
                stock=intinp("Enter STOCK:")

record=(str(pno),pname,brand,str(mrp),str(sellingp),str(s
tock))
                print(record)
                print()
                confirm=input("Confirm(y/n):")
                print()
                try:
                    if confirm=="y" or confirm=="Y":
                        crsr.execute("insert into product
values {}".format(record))
                        crsr.execute("commit")
                        print()
                except con.errors.IntegrityError:
```

```
                    print("****DUPLICATE KEY NOT
ALLOWED****")
                except con.errors.DataError:
                    print("DATA TOO LONG")
            except ValueError:
                print("_____Wrong values_____")
    except ValueError:
        print("_____Wrong values_____")



def remove(by,val):
    try:
        crsr.execute("select * from product where
{}='{}'".format(by,val))
        rec=crsr.fetchone()
        if rec==None:
            print("EMPTY RECORD")
        else:
            crsr.execute("desc product;")
            recs=crsr.fetchall()

print(recs[0][0].ljust(7),recs[1][0].ljust(25),recs[2][0]
.ljust(25),recs[3][0].ljust(10),recs[4][0].ljust(10),recs
[5][0].ljust(15),sep="")


print(str(rec[0]).ljust(7),rec[1].ljust(25),rec[2].ljust(
25),str(rec[3]).ljust(10),str(rec[4]).ljust(10),str(rec[5
]).ljust(15),sep="")
            confirm=input("Confirm(y/n):")
```

```
            if confirm=="y" or confirm=="Y":
                if by=="pno":
                    crsr.execute("delete from product
where pno ={}".format(val))
                elif by=="pname":
                    crsr.execute("delete from product
where pname = '{}'".format(val))
                elif by=="brand":
                    crsr.execute("delete from product
where brand ='{}'".format(val))
                crsr.execute("commit")
                print("RECORD DELETED")
                return
            else:
                pass
    except:
        print("_____Wrong values_____")



def removeproduct():
    print("REMOVE PRODUCT".center(85,'='))
    print("""
    MODES:
    0)EXIT
    1)PNO
    2)PNAME
```

```
    3)BRAND""")
    print("_"*85)
    y=0
    data=""
    while y==0:
        try:
            print()
            mod=input("Enter Mode:")
            if mod=="0":
                y=1
                break
            elif mod=="1" or mod=="pno" or mod=="PNO":
                z=1
                while z==1:
                    try:
                        val=intinp("Enter PNO")
                        if val==0:
                            z=0
                        else:
                            remove("pno",val)
                            z=0
                    except:
                        print("_____Wrong values_____")
            elif mod=="2" or mod=="PNAME" or
mod=="pname":
                z=1
```

```
                while z==1:
                    try:
                        data=input("Enter PNAME:")
                        if data=="0":
                            z=1
                        else:
                            z=0
                            remove("pname",data)
                    except:
                        print("_____Wrong values_____")
            elif mod=="3" or mod=="BRAND" or
mod=="brand":
                z=1
                while z==1:
                    try:
                        data=input("Enter BRAND:")
                        if data=="0":
                            z=1
                        else:
                            z=0
                            remove("brand",data)
                    except:
                        print("_____Wrong values_____")
            else:
                print("WRONG MODE SELECTED...")
        except ValueError:
```

```
            print("_____Wrong values_____")




def update(mod,pn):
    try:
        crsr.execute("select * from product where
pno={}".format(pn))
        rec=crsr.fetchone()
        if rec==None:
            print("EMPTY RECORD")
        else:
            crsr.execute("desc product;")
            recs=crsr.fetchall()


print(recs[0][0].ljust(7),recs[1][0].ljust(25),recs[2][0]
.ljust(25),recs[3][0].ljust(10),recs[4][0].ljust(10),recs
[5][0].ljust(15),sep="")


print(str(rec[0]).ljust(7),rec[1].ljust(25),rec[2].ljust(
25),str(rec[3]).ljust(10),str(rec[4]).ljust(10),str(rec[5
]).ljust(15),sep="")
            confirm=input("Confirm(y/n):")
            if confirm=="y" or confirm=="Y":
                if mod=="pno":
                    newrec=intinp("Enter NEW PNO:")
                    crsr.execute("update product set
pno={} where pno={}".format(newrec,pn))
                elif mod=="pname":
```

```
                    newrec=input("Enter NEW PNAME:")
                    crsr.execute("update product set
pname='{}' where pno={}".format(newrec,pn))
                elif mod=="brand":
                    newrec=input("Enter NEW BRAND:")
                    crsr.execute("update product set
brand='{}' where pno={}".format(newrec,pn))
                elif mod=="mrp":
                    newrec=intinp("Enter NEW MRP:")
                    crsr.execute("update product set
mrp={} where pno={}".format(newrec,pn))
                elif mod=="sellingp":
                    newrec=intinp("Enter NEW SELLINGP:")
                    crsr.execute("update product set
sellingp={} where pno={}".format(newrec,pn))
                elif mod=="qty":
                    newrec=intinp("Enter NEW QTY:")
                    crsr.execute("update product set
qty={} where pno={}".format(newrec,pn))
                crsr.execute("commit")
                print("RECORD UPDATED")
                return
            else:
                pass
    except:
        print("_____Wrong values_____")
```

```
def updateproduct():
    print("UPDATE PRODUCT".center(85,'='))
    print("""
    0)EXIT
    1)PNO
    2)PNAME
    3)BRAND
    4)MRP
    5)SELLINGP
    6)QTY
    """)
    y=0
    data=""
    while y==0:
        try:
            print()
            mod=input("Enter Mode:")
            if mod=="0":
                y=1
                break
            elif mod=="1" or mod=="pno" or mod=="PNO":
                z=1
                while z==1:
                    try:
                        no=intinp("Enter CURRENT PNO: ")
                        if no==0:
```

```
                        z=0
                    else:
                        update("pno",no)
                        z=0
                except:
                    print("_____Wrong values_____")
            elif mod=="2" or mod=="PNAME" or
mod=="pname":
                z=1
                while z==1:
                    try:
                        no=intinp("Enter CURRENT PNO: ")
                        if no==0:
                            z=1
                        else:
                            z=0
                            update("pname",no)
                    except:
                        print("_____Wrong values_____")
            elif mod=="3" or mod=="BRAND" or
mod=="brand":
                z=1
                while z==1:
                    try:
                        no=intinp("Enter CURRENT PNO: ")
                        if no==0:
```

```
                        z=1
                    else:
                        z=0
                        update("brand",no)
                except:
                    print("_____Wrong values_____")
            elif mod=="4" or mod=="MRP" or mod=="mrp":
                z=1
                while z==1:
                    try:
                        no=intinp("Enter CURRENT PNO: ")
                        if no==0:
                            z=1
                        else:
                            z=0
                            update("mrp",no)
                    except:
                        print("_____Wrong values_____")
            elif mod=="5" or mod=="SELLINGP" or
mod=="sellingp":
                z=1
                while z==1:
                    try:
                        no=intinp("Enter CURRENT PNO: ")
                        if no==0:
                            z=1
```

```
                    else:
                        z=0
                        update("sellingp",no)
                except:
                    print("_____Wrong values_____")
        elif mod=="6" or mod=="qty" or mod=="QTY":
            z=1
            while z==1:
                try:
                    no=intinp("Enter CURRENT PNO: ")
                    if no==0:
                        z=1
                    else:
                        z=0
                        update("qty",no)
                except:
                    print("_____Wrong values_____")
        else:
            print("WRONG MODE SELECTED...")
    except ValueError:
        print("_____Wrong values_____")
```

```
def searchby():
    print("SEARCH PRODUCT".center(85,'='))
    print("""
    MODES:
    0)EXIT
    1)PNO
    2)PNAME
    3)BRAND
    4)QTY""")
    print("_"*85)
    y=0
    data=""
    while y==0:
        try:
            print()
            mod=input("Enter Mode:")
            if mod=="0":
                y=1
                break
            elif mod=="1" or mod=="pno" or mod=="PNO":
                z=1
                while z==1:
                    try:
                        val=intinp("Enter PNO:")
                        if val==0:
                            z=0
```

```
                        else:
                            searchrecord("pno",val,"int")
                            z=0
                    except Exception as e:
                        print("_____Wrong values____",e)
            elif mod=="2" or mod=="PNAME" or
mod=="pname":
                z=1
                while z==1:
                    try:
                        data=input("Enter PNAME:")
                        if data=="0":
                            z=1
                        else:
                            z=0

searchrecord("pname",data,"chr")
                    except:
                        print("_____Wrong values_____")
            elif mod=="3" or mod=="BRAND" or
mod=="brand":
                z=1
                while z==1:
                    try:
                        data=input("Enter BRAND:")
                        if data=="0":
                            z=1
```

```
                        else:
                            z=0

searchrecord("brand",data,"chr")
                    except:
                        print("_____Wrong values_____")
            elif mod=="4" or mod=="qty" or mod=="QTY":
                z=1
                while z==1:
                    try:
                        val=intinp("Enter QTY:")
                        if val==0:
                            z=0
                        else:
                            searchrecord("qty",val,"int")
                            z=0
                    except:
                        print("_____Wrong values_____")
            else:
                print("WRONG MODE SELECTED...")
        except ValueError:
            print("_____Wrong values_____")
```

```
def instock(pno,demand):
    z=1
    while z==1:
        try:
            crsr.execute("select * from product where
pno={}".format(str(pno)))
            data=crsr.fetchone()
            cqty=data[5]
            if cqty==0:
                print(data[1]," IS OUT OF STOCK")
                return 0
            elif demand>cqty:
                print("Sorry,we only have",cqty)
                ch=intinp("Want to buy all?(0/1)")
                if ch==1:
                    return cqty
                else:
                    return 0
            else:
                return demand
        except Exception as err:
            print("_____Wrong values_____",err)
```

```
def searchrecord(by,val,typ):
    try:
        if typ=="int":
            crsr.execute("select * from product where
{}={}".format(by,str(val)))
            data=crsr.fetchall()
            if data==[]:
                print("EMPTY RECORD")
            else:
                crsr.execute("desc product;")
                recs=crsr.fetchall()

print(recs[0][0].ljust(7),recs[1][0].ljust(25),recs[2][0]
.ljust(25),recs[3][0].ljust(10),recs[4][0].ljust(10),recs
[5][0].ljust(10),sep="")
                for rec in data:

print(str(rec[0]).ljust(7),rec[1].ljust(25),rec[2].ljust(
25),str(rec[3]).ljust(10),str(rec[4]).ljust(10),str(rec[5
]).ljust(10),sep="")
        elif typ=="chr":
            crsr.execute("select * from product where
{}='{}'".format(by,val))
            data=crsr.fetchall()
            if data==[]:
                print("EMPTY RECORD")
            else:
                crsr.execute("desc product;")
                recs=crsr.fetchall()
```

```
print(recs[0][0].ljust(7),recs[1][0].ljust(25),recs[2][0]
.ljust(25),recs[3][0].ljust(10),recs[4][0].ljust(10),recs
[5][0].ljust(10),sep="")
                for rec in data:

print(str(rec[0]).ljust(7),rec[1].ljust(25),rec[2].ljust(
25),str(rec[3]).ljust(10),str(rec[4]).ljust(10),str(rec[5
]).ljust(10),sep="")



    except:
        print("_____Wrong values_____")



def billing():
    print("BILLING".center(90,'='))
    viewproduct()
    pnolist=[]
    qtylist=[]
    y=0
    sm=0
    print("ENTER PNO AS 0 TO EXIT")
    print()
    while y==0:
        try:
            no=intinp("Enter PNO:")
            if no==0:
```

```
                sm=0
                disc=0
                tdisc=0
                smprice=0
                print("-"*85)
                crsr.execute("desc product;")
                recs=crsr.fetchall()

print(recs[0][0].ljust(3),recs[1][0].ljust(20),recs[2][0]
.ljust(20),recs[3][0].ljust(7),recs[4][0].ljust(8),"AMOUN
T".ljust(7),"PRICE".ljust(7),"DISC".ljust(4),"COST".ljust
(7))
                for i in range(len(pnolist)):
                    crsr.execute("select * from product
where pno={}".format(pnolist[i]))
                    rec=crsr.fetchone()
                    cost=int(rec[4])*qtylist[i]
                    mrp=int(rec[3])
                    tmrp=mrp*qtylist[i]
                    sm=sm+cost
                    disc=tmrp-cost
                    tdisc=tdisc+disc
                    smprice=smprice+tmrp

print(str(rec[0]).ljust(3),rec[1].ljust(20),rec[2].ljust(
20),str(rec[3]).ljust(7),str(rec[4]).ljust(8),str(qtylist
[i]).ljust(7),str(rec[3]*qtylist[i]).ljust(7),str(disc).l
just(4),str(cost).ljust(7))
```

```
                print("%58s %6s"%("TOTAL
PRICE:",smprice))
                print("%58s %6s"%("TOTAL
DISCOUNT:",tdisc))
                print("%58s %6s"%("AMOUNT TO BE
PAID:",sm))
                print('''
    0) No
    1) Yes''')
                ch=intinp("Want To Update QTY?(0/1):")
                if ch==1:
                    for i in range(len(pnolist)):
                        crsr.execute("update product set
qty=qty-{} where pno={}".format(qtylist[i],pnolist[i]))
                    print("QTY UPDATED")
                    crsr.execute("commit")
                else:
                    print("QTY NOT UPDATED")
                break
                y=1
            else:
                z=0
                while z==0:
                    amt=intinp("Enter amount:")
                    amount=instock(no,amt)
                    if amount!=0:
                        crsr.execute("select * from
product where pno={}".format(no))
```

```
                        rec=crsr.fetchone()
                        if rec==None:
                            print("_____Wrong
values_____")
                            z=1
                        else:
                            cost=int(rec[4])*amount
                            sm=sm+cost
                            crsr.execute("desc product;")
                            recs=crsr.fetchall()

print(recs[0][0].ljust(3),recs[1][0].ljust(20),recs[2][0]
.ljust(20),recs[3][0].ljust(7),recs[4][0].ljust(8),"AMOUN
T".ljust(7),"COST".ljust(7))

print(str(rec[0]).ljust(3),rec[1].ljust(20),rec[2].ljust(
20),str(rec[3]).ljust(7),str(rec[4]).ljust(8),str(amount)
.ljust(7),str(cost).ljust(7))
                            print()
                            pnolist.append(no)
                            qtylist.append(amount)
                    z=1
                    break
        except ValueError:
            print("_____Wrong values_____")
```

```
def main():
    dbtable()
    while True:
        time.sleep(1)
        intro()
        option=input("Enter Option:")
        if option=="1":
            viewproduct()
        elif option=="2":
            addproduct()
        elif option=="3":
            removeproduct()
        elif option=="4":
            updateproduct()
        elif option=="5":
            searchby()
        elif option=="6":
            billing()
        elif option=="0":
            print("Thanks for using...")
            break
        else:
            print("Try Again")
main()
```

Output

```
Enter SQL Password:123
***WRONG PASSWORD***
Enter SQL Password:123456
Connecting....
```

```
                              STOCK MANAGEMENT                    01-02-2022 19:10

    0)EXIT
    1)VIEW STOCKS
    2)ADD PRODUCT
    3)REMOVE PRODUCT
    4)UPDATE PRODUCTS
    5)SEARCH RECORD
    6)GENERATE BILLING

Enter Option:
```

```
Enter Option:1

-------------------------------------------------------------------------------------
PNO     PNAME                   BRAND                   MRP       SELLINGP  QTY
1       Pencil                  Apsara                  50        45        70
2       Eraser                  Natraj                  20        15        40
3       Pen                     Doms                    4         3         300
4       Pen                     Linc Pentonic           20        19        100
5       Scale                   Natraj                  6         5         40
6       Highlighter             Cello                   15        13        40
```

```
Enter Option:2
====================================ADD PRODUCT====================================
Enter no. of records to be added:2
Enter PNO:6
Enter PNAME:Scissor
Enter BRAND:Munix
Enter MRP:75
Enter SELLINGP:70
Enter STOCK:30
('6', 'Scissor', 'Munix', '75', '70', '30')

Confirm(y/n):y

****DUPLICATE KEY NOT ALLOWED****
```

```
Enter no. of records to be added:2
Enter PNO:7
Enter PNAME:Scissor
Enter BRAND:Munix
Enter MRP:75
Enter SELLINGP:70
Enter STOCK:30
('7', 'Scissor', 'Munix', '75', '70', '30')

Confirm(y/n):y
```

```
                              DARSHIL SUPEMARKET                    29-09-2021 21:39

    0)EXIT
    1)VIEW PRODUCTS
    2)ADD PRODUCT
    3)REMOVE PRODUCT
    4)UPDATE PRODUCTS
    5)GENERATE BILLING

Enter Option:1

-------------------------------------------------------------------------------------
PNO PNAME           BRAND             MRP  SELLINGP
  1 Pencil          Apsara              5      4
  2 Eraser          Natraj              5      4
  3 Pen             Doms                4      3
  4 Scale           Natraj              6      5
  5 Highlighter     Cello              15     13
  6 Sharpner        Doms                5      4
```

```
                              STOCK MANAGEMENT                    01-02-2022 19:10

    0)EXIT
    1)VIEW STOCKS
    2)ADD PRODUCT
    3)REMOVE PRODUCT
    4)UPDATE PRODUCTS
    5)SEARCH RECORD
    6)GENERATE BILLING

Enter Option:1

-----------------------------------------------------------------------------------------
PNO     PNAME                     BRAND                     MRP       SELLINGP  QTY
1       Pencil                    Apsara                    50        45        70
2       Eraser                    Natraj                    20        15        40
3       Pen                       Doms                      4         3         300
4       Pen                       Linc Pentonic             20        19        100
5       Scale                     Natraj                    6         5         40
6       Highlighter               Cello                     15        13        40
7       Scissor                   Munix                     75        70        30
```

```
Enter Option:3
====================================REMOVE PRODUCT====================================

    MODES:
    0)EXIT
    1)PNO
    2)PNAME
    3)BRAND
______________________________________________________________________________________

Enter Mode:1
Enter PNO8
EMPTY RECORD

Enter Mode:1
Enter PNO7
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
7       Scissor                 Munix                   75          70          40
Confirm(y/n):y
RECORD DELETED

Enter Mode:2
Enter PNAME:Highlighter
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
6       Highlighter             Cello                   15          13          40
Confirm(y/n):y
RECORD DELETED

Enter Mode:0
```

```
Enter Option:1

--------------------------------------------------------------------------------------
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
1       Pencil                  Apsara                  50          45          70
2       Eraser                  Natraj                  20          15          40
3       Pen                     Doms                    4           3           300
4       Pen                     Linc Pentonic           20          19          100
5       Scale                   Natraj                  6           5           40
```

```
Enter Option:4
====================================UPDATE PRODUCT====================================

    0)EXIT
    1)PNO
    2)PNAME
    3)BRAND
    4)MRP
    5)SELLINGP
    6)QTY


Enter Mode:2
Enter CURRENT PNO: 2
PNO     PNAME                   BRAND                   MRP         SELLINGP  QTY
2       Eraser                  Natraj                  20          15        40
Confirm(y/n):y
Enter NEW PNAME:Sharpner
RECORD UPDATED

Enter Mode:4
Enter CURRENT PNO: 2
PNO     PNAME                   BRAND                   MRP         SELLINGP  QTY
2       Sharpner                Natraj                  20          15        40
Confirm(y/n):y
Enter NEW MRP:5
RECORD UPDATED

Enter Mode:5
Enter CURRENT PNO: 2
PNO     PNAME                   BRAND                   MRP         SELLINGP  QTY
2       Sharpner                Natraj                  5           15        40
Confirm(y/n):y
Enter NEW SELLINGP:4
RECORD UPDATED
```

```
Enter Mode:6
Enter CURRENT PNO: 2
PNO     PNAME                   BRAND                   MRP         SELLINGP  QTY
2       Sharpner                Natraj                  5           4         40
Confirm(y/n):y
Enter NEW QTY:80
RECORD UPDATED

Enter Mode:0
```

```
Enter Option:5
====================================SEARCH PRODUCT====================================

    MODES:
    0)EXIT
    1)PNO
    2)PNAME
    3)BRAND
    4)QTY
______________________________________________________________________________________

Enter Mode:1
Enter PNO:2
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
2       Sharpner                Natraj                  5           4           80

Enter Mode:2
Enter PNAME:pen
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
3       Pen                     Doms                    4           3           300
4       Pen                     Linc Pentonic           20          19          100

Enter Mode:4
Enter QTY:300
PNO     PNAME                   BRAND                   MRP         SELLINGP    QTY
3       Pen                     Doms                    4           3           300

Enter Mode:0
```

```
Enter Option:6
=========================================BILLING=========================================

-------------------------------------------------------------------------------------
PNO     PNAME                   BRAND                   MRP       SELLINGP  QTY
1       Pencil                  Apsara                  50        45        70
2       Sharpner                Natraj                  5         4         80
3       Pen                     Doms                    4         3         300
4       Pen                     Linc Pentonic           20        19        0
5       Scale                   Natraj                  6         5         5
ENTER PNO AS 0 TO EXIT

Enter PNO:1
Enter amount:5
PNO PNAME               BRAND               MRP     SELLINGP AMOUNT  COST
1   Pencil              Apsara              50      45       5       225

Enter PNO:4
Enter amount:3
Pen  IS OUT OF STOCK
Enter PNO:3
Enter amount:6
PNO PNAME               BRAND               MRP     SELLINGP AMOUNT  COST
3   Pen                 Doms                4       3        6       18

Enter PNO:5
Enter amount:6
Sorry,we only have 5
Want to buy all?(0/1)1
PNO PNAME               BRAND               MRP     SELLINGP AMOUNT  COST
5   Scale               Natraj              6       5        5       25
```

```
Enter PNO:0
-------------------------------------------------------------------------------
PNO PNAME               BRAND               MRP     SELLINGP AMOUNT  PRICE   DISC COST
1   Pencil              Apsara              50      45       5       250     25   225
3   Pen                 Doms                4       3        6       24      6    18
5   Scale               Natraj              6       5        5       30      5    25
                                          TOTAL PRICE:    304
                                      TOTAL DISCOUNT:     36
                                   AMOUNT TO BE PAID:    268

    0) No
    1) Yes
Want To Update QTY?(0/1):1
QTY UPDATED
```

Bibliography

http://www.google.com/

https://www.w3schools.com/

https://www.geeksforgeeks.org/

Computer Science with Python by Sumita Arora