INDEX

KENDRIYA VIDYALAYA

SOUTHERN COMMAND, PUNE

411001

COMPUTER SCIENCE (083)

SESSION 2020-21

I.S.R.M

Item and Stock Record Manager

Made by

Abir Abhyankar

Class XII-A

Roll No.: -

PROJECT

ITEM and STOCK RECORD MANAGER

USING

PYTHON & MySQL CONNECTIVITY

DATA FILE HANDLING

WITH THE HELP OF

MRS. RICHA CHAUDHARY

CERTIFICATE

THIS IS TO CERTIFY THAT ABIR ABHYANKAR OF CLASS XII-A HAS SUCCESSFULLY COMPLETED HIS COMPUTER SCIENCE PROJECT, ENTITLED "ITEM and RECORD MANAGER", UNDER MY GUIDANCE.

THIS PROJECT REPORT IS SUBMITTED FOR EVALUATION AS A PART OF CBSE CURRICULUM FOR AISSE 2020-21.

_____	_____	_____
COMPUTER SCIENCE TEACHER [MRS. RICHA CHAUDARY	***EXAMINER'S SIGNATURE***	***PRICIPAL'S SIGNATURE [MRS. SNEHAL MARATHE]***

ACKNOWLEDGEMENT

I WOULD LIKE TO EXPRESS MY SINCERE GRATITUDE TO THE SCHOOL AND PRINCIPAL FOR PRESENTING ME WITH THE OPPORTUNITY TO DEVELOP THIS PROJECT SUCCESSFULLY.

I WOULD ALSO LIKE TO THANK MY GUIDE TEACHER, MRS. RICHA CHAUDHARY FOR HELPING ME COMPLETE THIS PROJECT AND FOR HER VALUABLE SUGGESTIONS.

LASTLY, I WOULD LIKE TO THANK MY PARENTS, AS THEY HAD GIVEN ME RESOURCES, AND MOST IMPORTANT OF ALL, NEW IDEAS THAT HAVE BEEN INCORPORATED INTO THE FINAL DESIGN OF I.S.R.M.

REQUIREMENTS

THE REQUIREMENTS A COMPUTER SHOULD HAVE TO OPERATE THE SOFTWARE, "ITEM and STOCK RECORD MANAGER", ARE: -

1.HARDWARE REQUIREMENTS:

****INTEL 13 PROCESSOR***

****MINIMUM 2 GB RAM***

****AT LEAST 500 GB HDD***

2.SOFTWARE REQUIREMENTS:

****WINDOWS 8/10 OS***

****PYTHON 3.5 OR ABOVE (64 BIT)***

****MySQL***

AIM OF THE PROJECT

THE MAIN OBJECTIVE OF THIS PROJECT IS TO DESIGN AND DEVELOP AN ITEM AND STOCK RECORD MANAGER.

I.S.R.M IS MAINLY INTENDED FOR SHOP-OWNERS WHO NEED AN EASY-TO-USE STOCK AND ITEM RECORDER.

WHILE USING THIS APPLICATION, I HAD TO ENSURE TO INCLUDE A NUMBER OF FIELDS SO THAT THE USER FEELS COMFORTABLE AND IS NOT CONFUSED.

THE PROJECT "ITEM and STOCK RECORD MANAGER" CARRIES OUT THE FOLLOWING FUNCTIONS:

- ***KEEPS RECORDS AND MANAGES ITEMS.***
- ***CAN DO BILLING.***
- ***CAN UPDATE STOCK AND ENTER NEW ITEMS.***

IMPORTED MODULES

1.IMPORT MySQL.CONNECTOR

2.IMPORT TKINTER

3.IMPORT TKINTER.TTK

4.IMPORT DATETIME

5.IMPORT TABULATE

DATABASES AND TABLES

1)DATABASE: -

STORE STOCK RECORD

2)TABLES: -

i)EMPFB

ii)BILLDET

iii)STOCK ENTRY

PROJECT CODE

```
import mysql.connector as sql

mytab=sql.connect(host='localhost',user='root',passwd='26012004',database='store stockrecord')

o1=mytab.cursor()

c1='CREATE TABLE EMPFB(ECODE char(6),EmpName varchar(30),Passwd char(6));'

o1.execute(c1)

mytab.commit()

s=[('AB1507','Aman','BA7015'),('WD1908','Owais','DW8019'),('FN2204','Hari','NF4022')]

for i in range(len(s)):

    c2="INSERT INTO empfb(ECODE,EmpName,Passwd) VALUES('{}','{}','{}')".format(s[i][0],s[i][1],s[i][2])

    o1.execute(c2)

mytab.commit()

import mysql.connector as sql

mydb=sql.connect(host='localhost',user='root',passwd='26012004')

o1=mydb.cursor()#Object 1

c1='CREATE DATABASE StoreStockRecord'

o1.execute(c1)

c2='USE StoreStockRecord'

o1.execute(c2)

c3='CREATE TABLE StockEntry(ItemName varchar(30) PRIMARY KEY,Price FLOAT,Quantity int)'

o1.execute(c3)

mydb.commit()

import mysql.connector as sql
```

```
mytab=sql.connect(host='localhost',user='root',passwd='26012004',database='store
stockrecord')
curse=mytab.cursor()
curse.execute('CREATE TABLE BillDet(BillNo int,Dt Date,SalesmanName
varchar(20),CustomerName varchar(20),PhoneNumber varchar(10))')
mytab.commit()
import tkinter as tk
import tkinter.ttk as tk1
from datetime import datetime
import json
from tabulate import tabulate
import mysql.connector as sql
mytab=sql.connect(host='localhost',user='root',passwd='26012004',database='store
stockrecord')
global o1
o1=mytab.cursor()
mytab.commit()
ee=0
o1.execute('Select ItemName from stockentry')
l=[]
global a
for j in o1:
    l.append(str(j[0]))
    a=tuple(l)
db1={}#Item quantity for billing
db2={}#Item saleprice for billing
o1.execute('SELECT ItemName,SalePrice,Quantity FROM stockentry')
for j1 in o1:
    db1[j1[0]]=j1[2]
    db2[j1[0]]=j1[1]
```

```
def MSI():
    ch0a=d0a.get()
    ch0b=d0b.get()
    for j2 in l0:
        global ee
        ee+=1
        if j2[0]==ch0a and j2[1]==ch0b:
            ee=417
            import tkinter as tk
            import tkinter.ttk as tk1
            d1iq={}# items quantity in stock dict
            d2ip={}#items Wholesale cost dict
            d3ip={}#items sell price dict
            w1=tk.Tk()#window 1 main window at start
            w1.title('Main Store Interface')
            w1.geometry('1265x700')
            t1=tk.Label(w1,text='Welcome To The Main Store Interface',font=('Arial
Bold',28),fg='Blue')#text1
            t1.grid(column=16,row=1)
            import tkinter as tk
            import mysql.connector as sql

mytab=sql.connect(host='localhost',user='root',passwd='26012004',database='store
stockrecord')
            o1=mytab.cursor()
            c0='SELECT ItemName,Quantity,WholeSaleCost,SalePrice FROM
stockentry'
            o1.execute(c0)
            for i2 in o1:
                d1iq[i2[0]]=i2[1]
                d2ip[i2[0]]=i2[2]
```

```
            d3ip[i2[0]]=i2[3]
#------------------STOCKENTRY PROOGRAM PART 1 START----------------------
--------------------------------------------------------------------------------
        def Stock_Entry():#StockEntry Command1
            n1=0
            global w2
            w2=tk.Tk()#window 2 for stock entry
            w2.geometry('600x600')
            w2.title('Stock Entry')
            def INSERT():
                z1,z2,z3,z4=i1.get(),p1.get(),q1.get(),p2.get()
                d1iq[z1]=z3
                db1[z1]=z3
                db2[z1]=z4
                c1="INSERT INTO
stockentry(ItemName,WholeSaleCost,SalePrice,Quantity)
VALUES('{}',{},{},{})".format(z1,z2,z4,z3)
                o1.execute(c1)
                c2='SELECT ItemName,WholeSaleCost,Quantity,SalePrice FROM
stockentry'
                o1.execute(c2)
                for i in o1:
                    d2ip[i[0]]=i[1]
                    d1iq[i[0]]=i[2]
                    d3ip[i[0]]=i[3]
                mytab.commit()
                quit(w3)
                quit(w2)
            def quit(self):
                self.destroy()
            def Enter():
```

```
        global w3
        w3=tk.Tk()#window3
        w3.title('Confirmation')
        w3.geometry('600x150')
        w3.title('Confirmation')
        t6=tk.Label(w3,text='Are you Sure you want to Continue',font=('Arial
Bold',18))#text6
        t6.grid(column=4,row=0)

b3=tk.Button(w3,text='Yes',fg='blue',width=7,height=3,command=INSERT)#button
3
        b3.grid(column=3,row=1)

b4=tk.Button(w3,text='No',fg='blue',width=7,height=3,command=lambda:[quit(w3)]
)#button4
        b4.grid(column=5,row=1)
        w3.mainloop()
        w2.mainloop()
    w2.title('Stock Entry')
    t2=tk.Label(w2,text='Enter Item Name:',font=('Arial Bold',18))#text2
    t2.grid(column=0,row=2)
    i1=tk.Entry(w2,width=20,bg='yellow')#item entry1 dabba1
    i1.grid(column=1,row=2)
    t3=tk.Label(w2,text='Enter Whole Sale Cost:',font=('Arial Bold',18))#text3
    t3.grid(column=0,row=4)
    p1=tk.Entry(w2,width=10,bg='yellow')#Wholesale cost of item entry
dabba2
    p1.grid(column=1,row=4)
    t7=tk.Label(w2,text='Enter Sale Price:',font=('Arial Bold',18))#text7
    t7.grid(column=0,row=6)
    p2=tk.Entry(w2,width=10,bg='yellow')#sale price of item
```

```
        p2.grid(column=1,row=6)
        t4=tk.Label(w2,text='Enter Quantity:',font=('Arial Bold',18))#text4
        t4.grid(column=0,row=8)
        q1=tk.Entry(w2,width=10,bg='yellow')#quantity entry dabba3
        q1.grid(column=1,row=8)
        t5=tk.Label(w2,text='Press Enter to Continue',font=('Arial
Bond',18))#text5
        t5.grid(column=1,row=9)

b2=tk.Button(w2,text='Enter',fg='blue',width=10,height=3,command=Enter)
        b2.grid(column=1,row=10)
#------------------STOCKENTRY PROGRAM PART 1 ENDED!!!!!!!!----------------
--------------------------------------------------------------------------------
#------------------STOCKDETAILS PROGRAM PART 2 START!!!!!!----------------
--------------------------------------------------------------------------------
    def Stock_Details():#Stock_Details Command2
        def Search():

mytab2=sql.connect(host='localhost',user='root',passwd='26012004',database='stor
estockrecord')
            o2=mytab2.cursor()
            i1=0
            o3=str(d1.get())
            o3a=o3.capitalize()
            c5a='SELECT * FROM stockentry WHERE ItemName LIKE \''
            c5b=str(o3a)+'%\' or ItemName LIKE \'%'
            c5c=str(o3a)+'%\' or ItemName LIKE \'%'
            c5d=str(o3a)+'\';'
            global c5
            c5=str(c5a+c5b+c5c+c5d)#command5
            o2.execute(c5)
```

```
            for j1 in o2 :
                if len(j1)!=0:
                    print()
                    print('Item : ',j1[0],'\nQuantiy in Stock : ',j1[3],'\nWhole Sale Cost :
',j1[1],'\nSale Price : ',j1[2])
                else:
                    print('Invaid Name Please Try Again')
        w4=tk.Tk()#window 4
        w4.geometry('700x700')
        w4.title('Stock Details')
        list1=[]
        try:

mytab2=sql.connect(host='localhost',user='root',passwd='26012004',database='stor
estockrecord')
            o2=mytab2.cursor()
            c3='Select ItemName,Quantity,WholeSaleCost,SalePrice FROM
stockentry'#command3
            o2.execute(c3)
        except mysql.connector.errors.InternalError:
            w4.destroy()
        i=1
        a00=[]
        for j in o2:
            s3='   '+str(i)+'.   '
            rv=list(j)
            rv.insert(0,s3)
            list1.append(rv)
            a00.append(j[0])
            i+=1
        a000=tuple(a00)
```

```
        from tabulate import tabulate
        s1=tabulate(list1,headers=[' Sno. ',' ItemName ',' Quantity ',' WholeSaleCost ',' SalePrice ']) 
        l1=tk.Label(w4,text=s1)
        l1.grid(column=0,row=0)
        t8=tk.Label(w4,text='Enter Item to be Searched:',font=('Arial Bond',18))#text8
        t8.grid(column=0,row=1)
        d6=tk.Entry(w4,width=20)#dabba6
        d6.grid(column=1,row=1)
        d1=tk1.Combobox(w4)
        d1['values']=a000
        d1.grid(column=1,row=1)

b6=tk.Button(w4,text='Enter',fg='blue',width=7,height=3,command=Search)#button6
        b6.grid(column=4,row=1)
        b7=tk.Button(w4,text='Exit Stock Details',fg='blue',width=15,height=3,command=w4.destroy)#button7
        b7.grid(column=0,row=3)
#----------------------STOCKDETAILS PROGRAM PART 2 ENDED!!!!!!----------------------------------------------------------------------------------------
#----------------------BILLING PROGRAM PART 3 STARTS!!!!!----------------------------------------------------------------------------------------
    def Billing():
        global bill1list
        bill1list=[]
        def Make_Bill():
            def FinalBill():
                global sss1,sss2
                sss1=dbdb1.get()
                sss2=dbdb2.get()
```

```
            if int(sss2.strip())<=int(db1[sss1]):
                def FinalBill2():

mytab3=sql.connect(host='localhost',user='root',passwd='26012004',database='stor
estockrecord')
                    o3=mytab3.cursor()
                    global bill2list
                    b222=len(bill1list)+1
                    bill2list=[]
                    o3.execute("SELECT BillNo from billdet")
                    for i11 in o3:
                        billno=int(i11[0])+1
                    bill2list.append(b222)
                    bill2list.append(sss1)
                    bill2list.append(int(db2[sss1])*int(sss2))
                    bill2list.append(sss2)
                    bill1list.append(bill2list)
                    o3.execute("UPDATE stockentry SET Quantity=(Quantity - {})
WHERE ItemName='{}'".format(int(sss2),sss1))
                    mytab3.commit()
                global wb3
                wb3=tk.Tk()
                wb3.geometry('500x500')
                wb3.title('Confirmation')
                dbbbb1=tk.Label(wb3,text='Are You Sure You Want To
Continue??',font=('Arial Bond',18))
                dbbbb1.grid(column=1,row=0)

dbbbd1=tk.Button(wb3,text='Yes',command=lambda:[FinalBill2(),wb3.destroy()])
                dbbbd1.grid(column=0,row=1)
                dbbbd2=tk.Button(wb3,text='No',command=wb3.destroy)
```

```
                dbbbd2.grid(column=2,row=1)
            else:
                from tkinter import messagebox
                ccc1='Sorry Only '+str(db1[sss1])+' '+str(sss1)+' items are in Stock'
                messagebox.showinfo('Out Of Stock',ccc1)
        def FinalBill3():
            from datetime import datetime,date
            now=datetime.now()
            dt=now.strftime('%d/%m/%Y %H:%M:%S')#date and time of billing
            dt2=date.today()#date for database

mytab3=sql.connect(host='localhost',user='root',passwd='26012004',database='stor
estockrecord')
            o3=mytab3.cursor()
            o3.execute("SELECT BillNo from billdet")
            for i11 in o3:
                billno=int(i11[0])+1
            billname='ABCDSuperBill'+str(billno)
            o3.execute("INSERT INTO billdet
VALUES({},'{}','{}','{}','{}')".format(billno,str(dt2),str(ss1),str(ss2),str(ss3)))
            mytab3.commit()
            f=open(billname,'w')
            bill1='Salesman Name :' + str(ss1)
            f.write(bill1)
            f.write('\n                    ABCD SUPERMARKET\n')
            f.write('------------------------------------------------------------\n')
            bill2='Bill
Number:'+str(billno)+'\nDT:'+str(dt)+'\nDesc:'+str(ss2)+'\nContact
Number'+str(ss3)+'\n------------------------------------------------------------'
            f.write(bill2)
            bill3='\nSNo    Item_Name     Price     Qty\n'
```

```
            f.write(bill3)
            totamt=0
            for jd in bill1list:
                bill5=str(jd[0])+'  '+str(jd[1])+'  '+str(jd[2])+'  '+str(jd[3])+'\n'
                f.write(bill5)
                totamt+=int(jd[2])
            f.write('------------------------------------------------------------\n')
            bill4='          Amount: Rs.' + str(totamt)+'\n'
            f.write(bill4)
            f.write('------------------------------------------------------------\n')
            f.write('Thank You Visit Again')
            f.close()
            f2=open(billname,'r')
            s121=f2.read()
            print(s121)
            f2.close()
        global wb2,ss1,ss2,ss3
        ss1=dbd1.get()
        ss2=dbd2.get()
        ss3=dbd3.get()
        wb2=tk.Tk()
        wb2.geometry('600x600')
        wb2.title('Billing')
        dbbb1=tk.Label(wb2,text='Enter Item Name : ',font=('Arial Bond',18))
        dbbb1.grid(column=0,row=0)
        dbdb1=tk1.Combobox(wb2)
        dbdb1['values']=a
        dbdb1.grid(column=1,row=0)
        dbbb2=tk.Label(wb2,text='Enter Quantity to be Purchased :
',font=('Arial Bond',18))
```

```
            dbbb2.grid(column=0,row=1)
            dbdb2=tk.Entry(wb2,width=15)
            dbdb2.grid(column=1,row=1)
            dbbb9=tk.Button(wb2,text='Enter',fg='blue',command=FinalBill)
            dbbb9.grid(column=1,row=2)
            dbbb10=tk.Button(wb2,text='Print Bill',fg='blue',command=FinalBill3)
            dbbb10.grid(column=1,row=3)
        global wb1
        wb1=tk.Tk()
        wb1.geometry('600x600')
        wb1.title('Billing')
        dbb1=tk.Label(wb1,text='Enter Salesman Name : ',font=('Arial Bond',18))
        dbb1.grid(column=0,row=0)
        dbd1=tk.Entry(wb1,width=20)
        dbd1.grid(column=1,row=0)
        dbb2=tk.Label(wb1,text='Enter Buyer Name : ',font=('Arial Bond',18))
        dbb2.grid(column=0,row=1)
        dbd2=tk.Entry(wb1,width=20)
        dbd2.grid(column=1,row=1)
        dbb3=tk.Label(wb1,text='Enter Contact Number : ',font=('Arial Bond',18))
        dbb3.grid(column=0,row=2)
        dbd3=tk.Entry(wb1,width=10)
        dbd3.grid(column=1,row=2)
        dbb9=tk.Button(wb1,text='Enter',fg='blue',command=Make_Bill)#button 9
        dbb9.grid(column=1,row=3)
#----------------------BILLING PPROGRAM PART 3 ENDS!!!!!!!!!!!!!!----------------
--------------------------------------------------------------------------------
#----------------------STOCK UPDATE PART 4 STARTS!!!!!!!!!!!!!!!!-----------------
--------------------------------------------------------------------------------
    def Stock_Update():
```

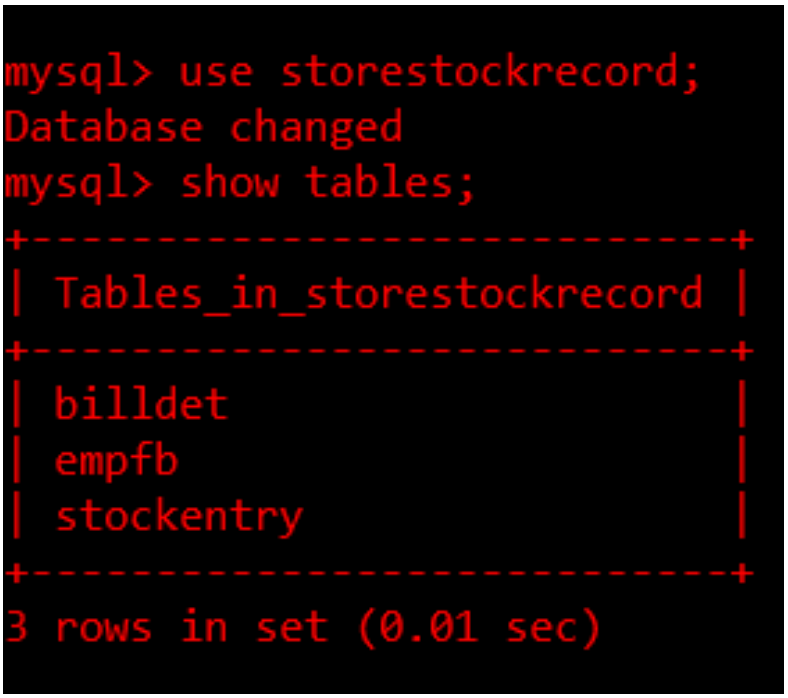
```
        def Updater():
            global su1,su2
            su1=csu1.get()
            su2=int(csu2.get())
            def Updater2():
                import mysql.connector as sql

mytab3=sql.connect(host='localhost',user='root',passwd='26012004',database='storestockrecord')
                o3=mytab3.cursor()
                cm1="UPDATE stockentry SET Quantity={} WHERE ItemName='{}';".format(su2,su1)
                o3.execute(cm1)
                mytab3.commit()
            wsu2=tk.Tk()
            wsu2.geometry('700x700')
            wsu2.title('Confirmation')
            lsu3=tk.Label(wsu2,text='Are You Sure You Want To Continue',font=('Arial Bold',18))
            lsu3.grid(column=2,row=1)
            bsu2=tk.Button(wsu2,text='Yes',font=('Arial Bold',18),command=lambda:[Updater2(),wsu2.destroy()])
            bsu2.grid(column=1,row=2)
            bsu3=tk.Button(wsu2,text='No',font=('Arial Bold',18),command=wsu2.destroy)
            bsu3.grid(column=23,row=2)
        wsu1=tk.Tk()
        wsu1.geometry('600x600')
        wsu1.title('Stock Update')
        lsu1=tk.Label(wsu1,text='Select Item : ',font=('Arial Bold',18))
        lsu1.grid(column=0,row=0)
```

```
        csu1=tk1.Combobox(wsu1)
        csu1.grid(column=1,row=0)
        csu1['values']=a
        lsu2=tk.Label(wsu1,text='Enter New Quantity of the Items : ',font=('Arial
Bold',18))
        lsu2.grid(column=0,row=1)
        csu2=tk.Entry(wsu1,width=10)
        csu2.grid(column=1,row=1)
        bsu1=tk.Button(wsu1,text='Enter',fg='blue',font=('Arial
Bold',18),command=Updater)
        bsu1.grid(column=1,row=3)
        bsu4=tk.Button(wsu1,text='EXIT',fg='blue',font=('Arial
Bold',18),command=wsu1.destroy)
        bsu4.grid(column=1,row=4)
#------------------------STOCK UPDATE PART 4 ENDS!!!!----------------------------
--------------------------------------------------------------------------------
    b1=tk.Button(w1,text='Stock Entry',fg='red',font=('Arial
Bold',22),width=16,height=5,command=Stock_Entry)#button1
    b1.grid(column=15,row=3)
    b5=tk.Button(w1,text='Stock Details',fg='red',font=('Arial
Bold',22),width=16,height=5,command=Stock_Details)#button5
    b5.grid(column=17,row=3)
    b8=tk.Button(w1,text='Billing',fg='red',font=('Arial
Bold',22),width=16,height=5,command=Billing)#button8
    b8.grid(column=15,row=4)
    b111=tk.Button(w1,text='Stock Update',fg='red',font=('Arial
Bold',22),width=16,height=5,command=Stock_Update)
    b111.grid(column=17,row=4)
    b112=tk.Button(w1,text='EXIT',fg='red',font=('Arial
Bold',22),width=16,height=5,command=w1.destroy)
    b112.grid(column=16,row=5)
    print(a,type(a))
```

```
    if ee==len(l0):
        from tkinter import messagebox
        messagebox.showinfo('Entry Error','Username or Password is incorrect')
c0='SELECT ECODE,Passwd FROM empfb'
o1.execute(c0)
global l0
l0=[]
for i0 in o1:
    l0.append(i0)
w0=tk.Tk()
w0.geometry('400x400')
w0.title('USER ID and PASSWORD')
t0a=tk.Label(w0,text='UserCode : ')#usercode text
t0a.grid(column=0,row=0)
d0a=tk.Entry(w0,width=10)#dabba0 username
d0a.grid(column=1,row=0)
t0b=tk.Label(w0,text='Password : ')#password text
t0b.grid(column=0,row=1)
d0b=tk.Entry(w0,width=10)#dabba0 password
d0b.grid(column=1,row=1)
b0=tk.Button(w0,text='Enter',fg='blue',command=MSI)
b0.grid(column=1,row=3)
```

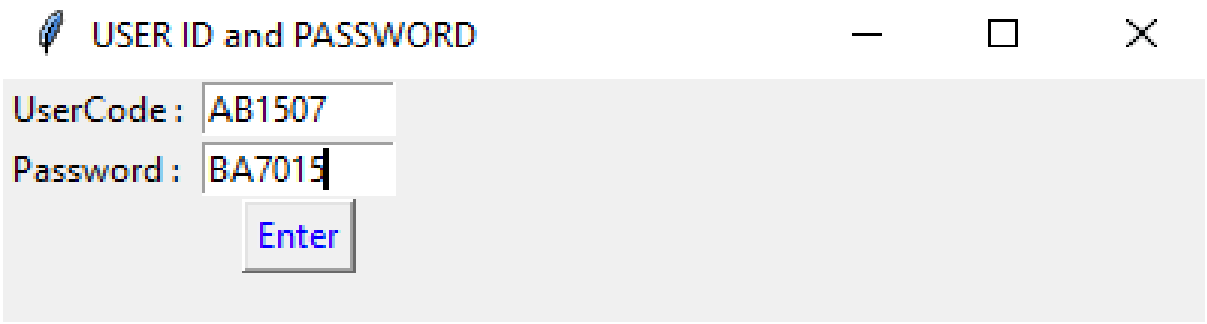
OUTPUT

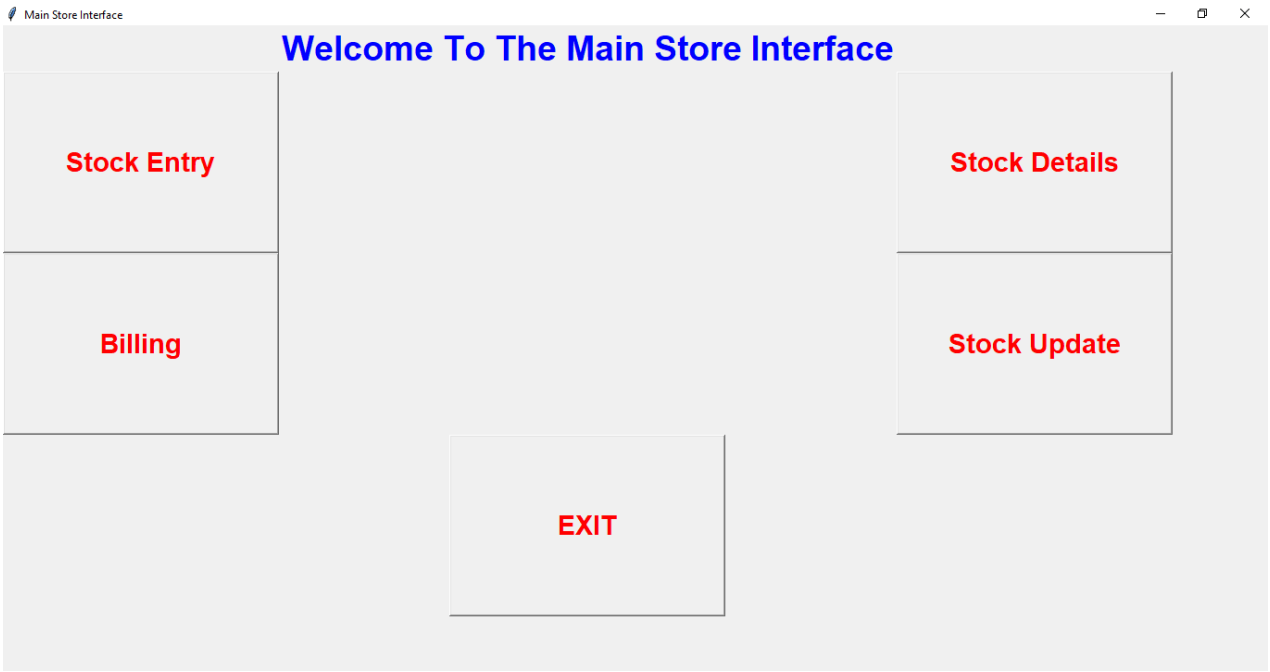

mysql> use storestockrecord;
Database changed
mysql> show tables;
+-----------------------------+
| Tables_in_storestockrecord |
+-----------------------------+
| billdet |
| empfb |
| stockentry |
+-----------------------------+
3 rows in set (0.01 sec)

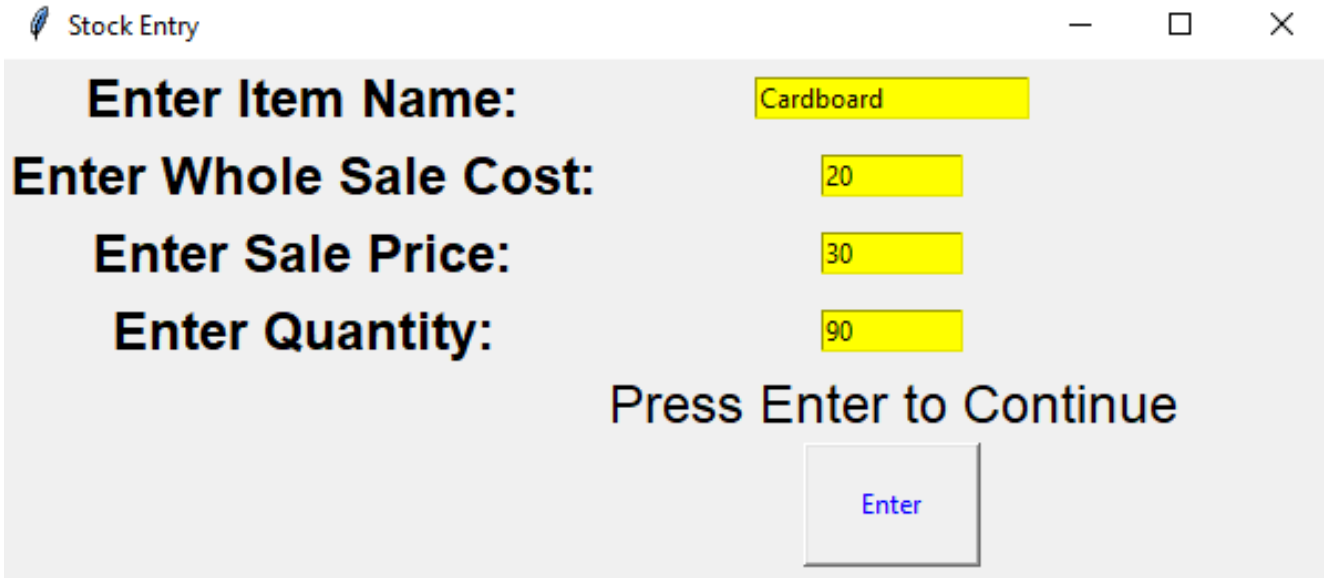
```
mysql> select * from stockentry;
+-----------+---------------+-----------+----------+
| ItemName  | WholeSaleCost | SalePrice | Quantity |
+-----------+---------------+-----------+----------+
| Biscuit   |            14 |        25 |       70 |
| Broom     |            37 |        50 |       77 |
| Cardboard |            20 |        30 |       85 |
| Detergent |            47 |        96 |       50 |
| Maggi     |            10 |        20 |       50 |
| Milk      |            42 |        56 |       58 |
| Nachos    |            23 |        40 |       77 |
| Oil       |            78 |       120 |       30 |
| Paint     |            88 |       128 |       30 |
| Parle G   |             5 |        10 |      450 |
| Soap      |            10 |        20 |       43 |
| USB Drive |           400 |       600 |        8 |
+-----------+---------------+-----------+----------+
12 rows in set (0.00 sec)
```

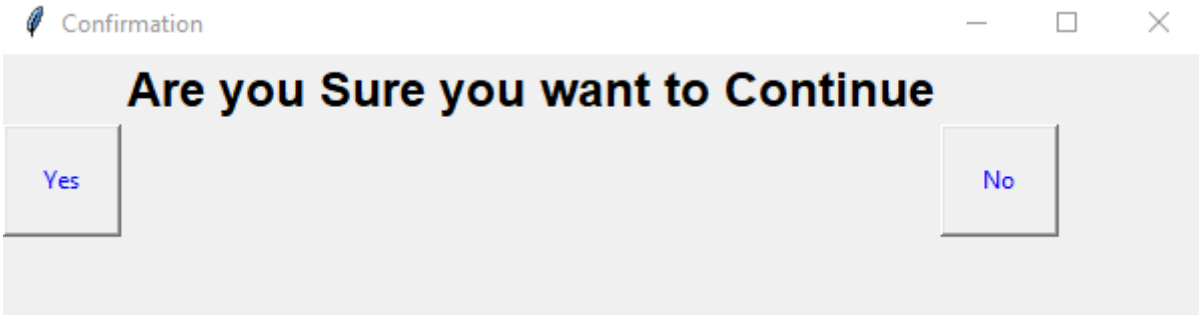
```
mysql> select * from empfb;
+--------+----------+--------+
| ECODE  | EmpName  | Passwd |
+--------+----------+--------+
| AB1507 | Aman     | BA7015 |
| WD1908 | Owais    | DW8019 |
| FN2204 | Hari     | NF4022 |
+--------+----------+--------+
3 rows in set (0.00 sec)
```

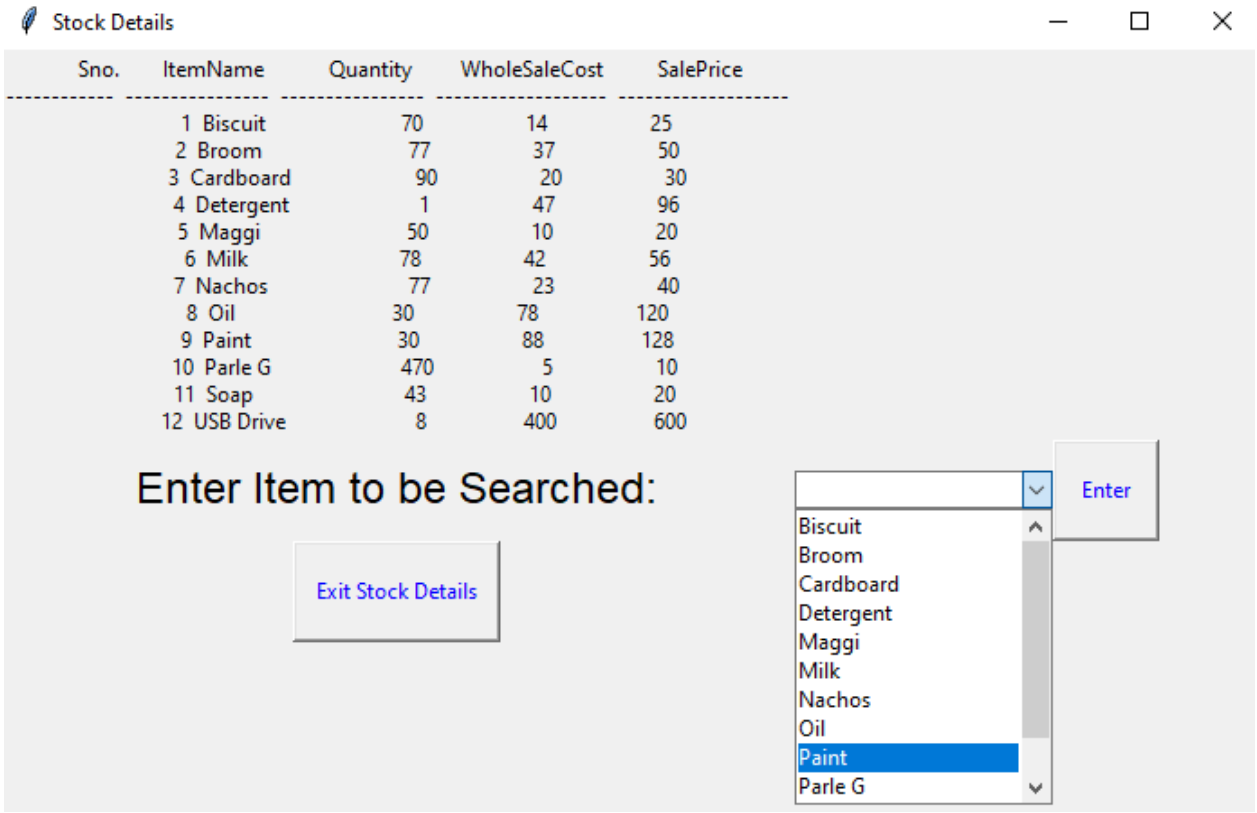
```
mysql> select * from billdet;
+--------+------------+--------------+--------------+-------------+
| BillNo | Dt         | SalesmanName | CustomerName | PhoneNumber |
+--------+------------+--------------+--------------+-------------+
|      0 | 2020-10-10 | Trial        | Successful   | Works       |
|      1 | 2020-11-19 | Aman         | Abir         | 9866774121  |
+--------+------------+--------------+--------------+-------------+
2 rows in set (0.00 sec)
```



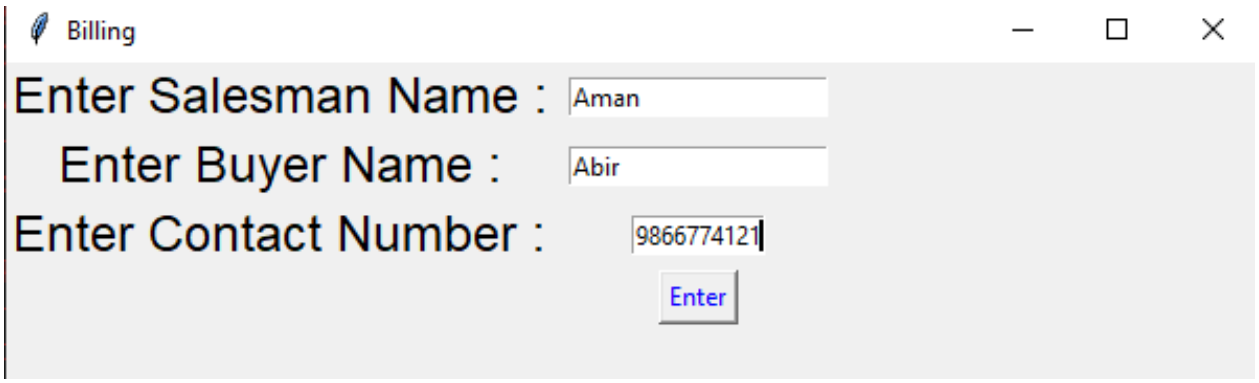


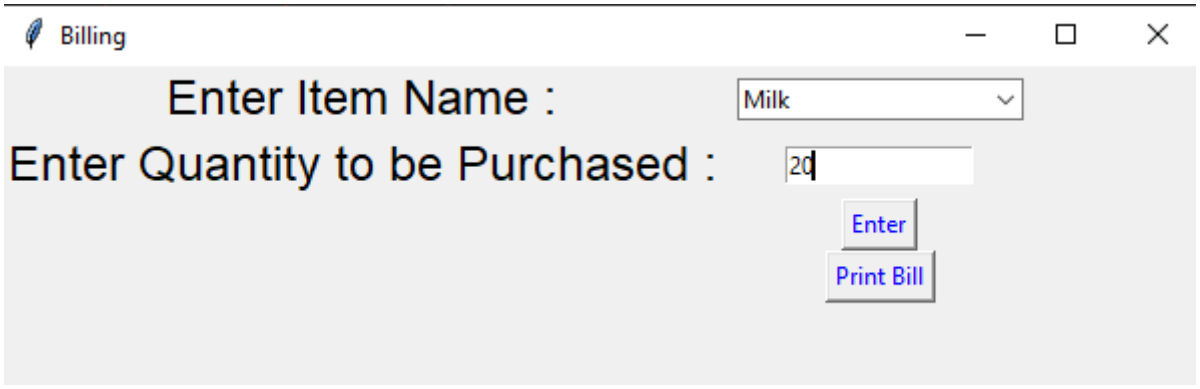

Stock Entry
Enter Item Name:
Cardboard
Enter Whole Sale Cost:
20
Enter Sale Price:
30
Enter Quantity:
90
Press Enter to Continue
Enter

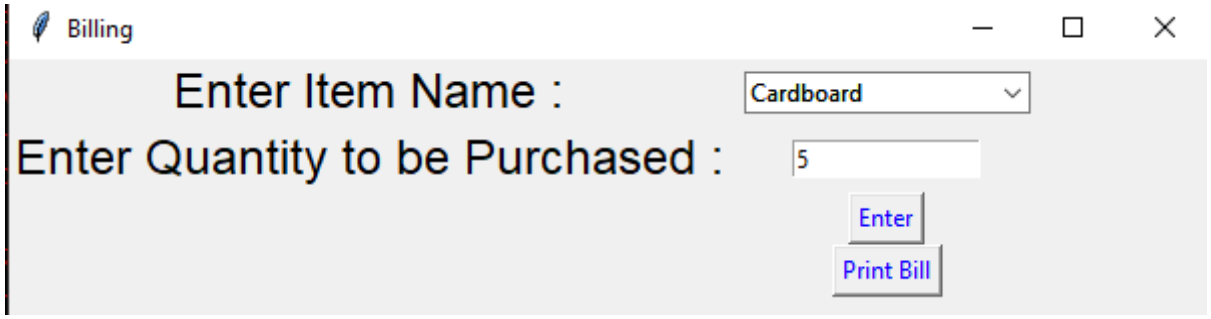

Confirmation
Are you Sure you want to Continue
Yes
No

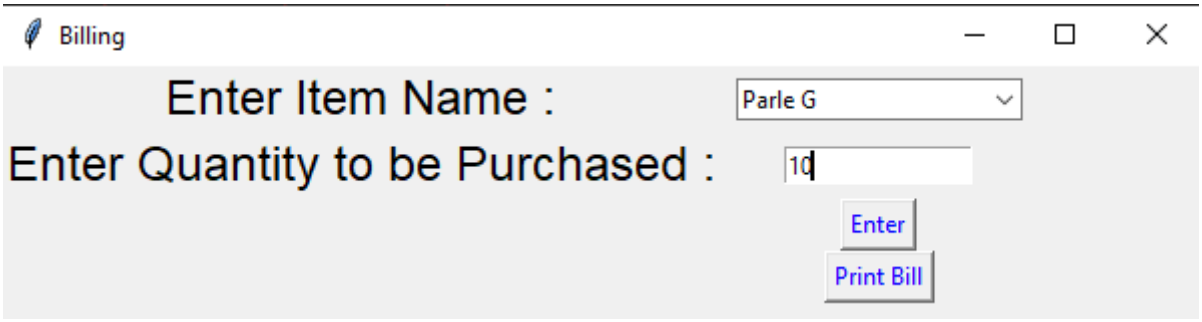

Stock Details
Sno. ItemName Quantity WholeSaleCost SalePrice
1 Biscuit 70 14 25
2 Broom 77 37 50
3 Cardboard 90 20 30
4 Detergent 1 47 96
5 Maggi 50 10 20
6 Milk 78 42 56
7 Nachos 77 23 40
8 Oil 30 78 120
9 Paint 30 88 128
10 Parle G 470 5 10
11 Soap 43 10 20
12 USB Drive 8 400 600
Enter Item to be Searched:
Enter
Biscuit
Broom
Cardboard
Detergent
Maggi
Milk
Nachos
Oil
Paint
Parle G
Exit Stock Details

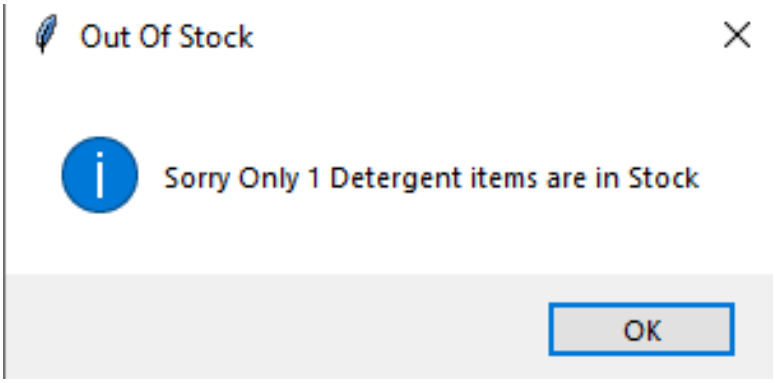
```
Item :  Paint
Quantiy in Stock :  30
Whole Sale Cost :  88.0
Sale Price :  128.0
```

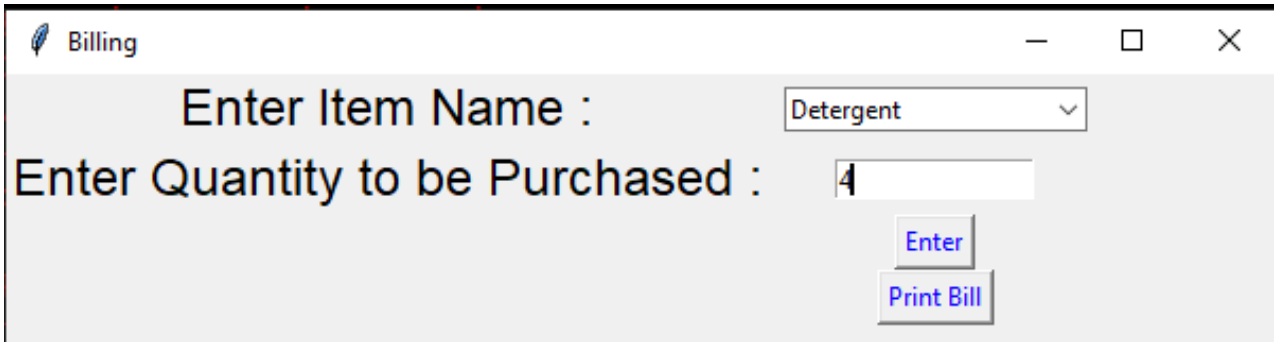




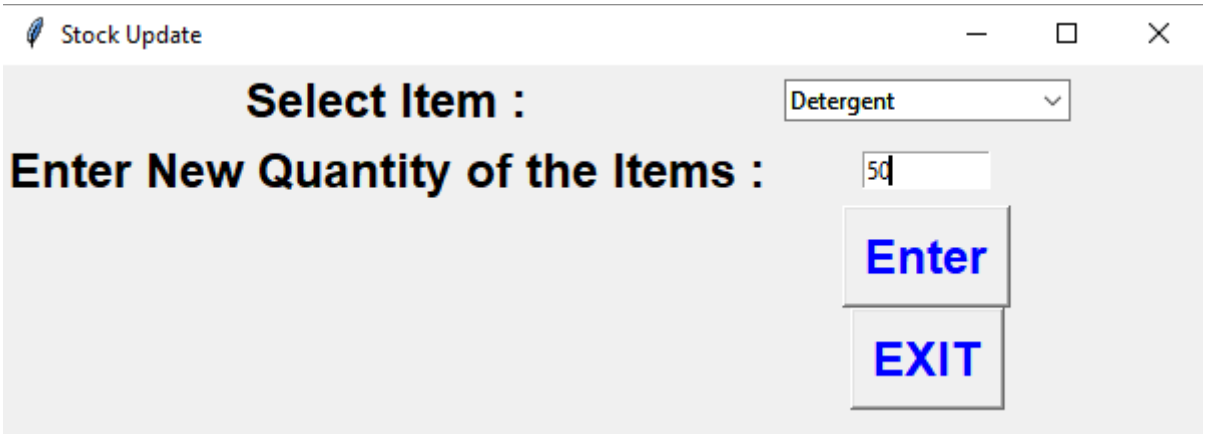









```
Salesman Name :Aman
                        ABCD SUPERMARKET
-----------------------------------------------------------------
Bill Number:1
DT:19/11/2020 15:35:40
Desc:Abir
Contact Number9866774121
-----------------------------------------------------------------
SNo    Item_Name     Price     Qty
1  Milk  1120  20
2  Cardboard  150  5
3  Parle G  100  10
4  Parle G  100  10
-----------------------------------------------------------------
          Amount: Rs.1470
-----------------------------------------------------------------
Thank You Visit Again
```



CONCLUSION

THIS PROJECT, AS YOU KNOW BY NOW, IS TO SIMPLIFY THE PROCESS OF BILLING AND KEEPING RECORDS. THE PRESENTED PROJECT CAN BE USED ANYWHERE, FROM THE ROADISIDE STALL, TO A HUGE SUPERMARKET CHAIN.

THIS PROJECT HAS GUI BASED PROGRAM, WHICH WILL HELP IN STORING, RETRIEVING AND UPDATING

INFORMATION THROUGH USER-FRIENDLY WINDOWS.

THE PROJECT, "ITEMS and STOCK RECORD MANAGER", IS AIMED TO HELP STORE OWNERS WITH A SECURE ITEM MANAGER. WITH EACH EMPLOYEE HAVING THEIR OWN UNIQUE LOG IN CREDENTIALS THE RECORDS KEPT CANNOT BE ALTERED BY OUTSIDERS. ALL THE DATA STORED IS CHANGED AND WRITTEN BY MANUALLY FILLING THE REQUIRED FIELDS. THIS SOFTWARE WILL HELP THE EMPLOYEES IN BILLING OF ITEMS, UPDATING STOCKS AND OTHER PROCESSES. IN THE MANUAL PROCESS THERE IS A HUGE SCOPE OF ERRORS. SO TO REMOVE THESE HUMAN ERRORS THIS PROGRAM IS OF EXTREME SIGNIFICANCE.

BIBLIOGRAPHY

BOOKS AND WEBSITES ALONG WITH MY TEACHERS ASSISTANCE HAVE MADE THIS PROJECT A REALITY.

THE FOLLOWING BOOKS AD WEBSITES HAVE BEEN REFERRED TO WHILE DEVELOPING THIS PROJECT.

BOOK :- COMPUTER SCIENCE WITH PYTHONTEXTBOOK OF CLASS 12 BY SUMITA ARORA

WEBSITES :-

(i) YOUTUBE CHANNEL freeCodeCamp.org
https://www.youtube.com/channel/UC8butISFwT-Wl7EVohUKoBQ

(ii) https://www.w3schools.com/