LUCKNOW PUBLIC SCHOOLS & COLLEGES

ज्ञानात् लभते सर्वम्

COMPUTER SCIENCE

PROJECT FILE

(Session : 2020-21)

INVENTORY MANAGEMENT

Submitted to- Mr. Indresh Kumar Singh

Submitted by- Utkarsh Verma [XII –A]

Acknowledgement

I *would like to extend my sincere and heartfelt gratitude to my teacher Mr Indresh Kumar Singh who has helped me in this endeavor and has always been very cooperative and without his/her cooperation,guidance and encouragement,the project couldn't have been what it evolved to be.*

I extend my heartfelt thanks to my faculty for their guidance and constant supervision , as well as,for providing me the necessary information regarding the project.

I am also thankful to my parents for their cooperation and encouragement.

At last but not least,gratitude to all my friends who helped me to complete this project within a limited time frame.

--- Utkarsh Verma

XII A (Roll No- 43)

CERTIFICATE

This is to certify that Utkarsh Verma ,student of class XII- A has successfully completed the research on the project "Inventory Management " under the guidance of Mr Indresh Kumar Singh(Subject teacher) during academic year 2020-2021 in the partial fulfillment of the requirements in COMPUTER SCIENCE prescribed by CBSE BOARD in the school LUCKNOW PUBLIC SCHOOL,Sitapur.

Teacher's Signature: ***Principal Signature:***

Internal Examiner: ***External Examiner:***

Contents

IB

PREFACE

Inventory management is the practice of ordering, storing, tracking, and controlling the inventory .

After a lot of thinking and planning , the idea of designing a GUI based application for inventory management came in my mind as inventory management applies to every item a business uses to produce , its products or services —from raw materials to finished goods. In other words, inventory management covers every aspect of a business's inventory ,it came as wonderful opportunity to me to follow upon the very thought and program Inventory Manager as CS Project.

The project contains following modules:-

1. Product Management: This module is used to add, update and delete the products.

2. Purchase Management: This module is used to manage the purchase system.

3. Sales Managernent: This module is used to manage the sale of the products.

4. User Management: This module is used to add/delete the user/staff.

5. Database setup: This module is used to setup the database in the system for the first time.

NOTE: The password for mysql should be "1234" or it can be changed by changing password variable's value

In Upack.py file and the database will exist with name "stock" in mysql backend.

All these modules are invoked through logging in via "LoginScreen" and then "MenuScreen" module through GUI interaction while the program in turn runs when " A1)Executor" file is executed. The Functions associated along with these modules are all in "Upack " file . So in a nutshell the program contains 4 python files:

- ***A1)Executor.py***
- ***LoginScreen.py***
- ***MenuScreen.py***
- ***Upack.py***

PREREQUISITES

- ***Windows 7 or above OS***
- ***Python 3.8 or above***
- ***MySql 5.7***
- ***Images which can be downloaded at-***
- ***https://drive.google.com/drive/folders/1e0wyeMlhjRAAneo88pP_Wn1aUo2O0Bsf?usp=sharing***
- ***Mysql password = 1234***

The above password can be changed by changing variable password's value in Upack.py file.

SOURCE CODE

A1)Executor.py

```
import LoginScreen as lsn
lsn.mainscreen()
```

LoginScreen.py

```
#------------------IMPORT SECTION-----------------------
import MenuScreen as msn
import Upack as u
from sys import *
from tkinter import*
from PIL import ImageTk, Image
import mysql.connector
#------------------FUNCTION SECTION-----------------------
def mainscreen():
    def dbmanagement():
        Window.destroy()
        u.create_database()
    def login():
        if e1.get()=="root" and e2.get()=="1234":
            Window.destroy()
            msn.mainscrn()
            exit()
        mydb=mysql.connector.connect(host="localhost",user="root",\
                            password=u.password, database="stock")
        mycursor= mydb.cursor()
    sql = "SELECT uid,upwd from user"
    mycursor.execute(sql)
    d=mycursor.fetchall()
    for a in d:
        if a[0]==e1.get() and a[1]==e2.get():
            Window.destroy()
            msn.mainscrn()
            exit()
    else:
        Window.destroy()
        u.message("Access Denied","INVALID DETAILS")
    exit()
#--------------------Main Window-------------------

Window = Tk()
Window.title('Welcome to Inventory Manager')
Window.geometry('1025x684+0+0')
Window.config()

#--------------------Importing Images-----------------

b = ImageTk.PhotoImage(Image.open("background0.jpg"))
Button1 = ImageTk.PhotoImage(Image.open("Button0.jpg"))
```

```
Button2 = ImageTk.PhotoImage(Image.open("Button05.jpg"))
label1 = ImageTk.PhotoImage(Image.open("label77.jpg"))
label2 = ImageTk.PhotoImage(Image.open("label99.jpg"))

#----------------------Window Labels---------------------------
background = Label(Window, image=b)
background.place(x=0, y=0)
l1=Label(Window,image=label1).place(x=200,y=250)
l2=Label(Window,image=label2).place(x=200,y=350)
#------------------------Entries-------------------------------
e1=ttk.Entry(Window, width=20,font=('calibre', 15, 'bold'))
e1.place(x=520,y=265)
e2=ttk.Entry(Window, width=20,font=('calibre', 15, 'bold'))
e2.place(x=520,y=365)
#----------------------Buttons----------------------------------
B1= Button(Window,bd=0,image=Button1,command=login)
B1.place(x=350,y=420)
B5= Button(Window,bd=0,image=Button2,command=\
                                                    dbmanagement)
B5.place(x=350,y=580)
Window.mainloop()
```

MenuScreen.py

```
#------------------IMPORT SECTION-------------------------
from sys import *
from tkinter import*
from PIL import ImageTk, Image
import Upack as u
def mainscrn():
    def productmanagement():
        Window.destroy()
        u.product_mgmt()
    def purchasemanagement():
        Window.destroy()
        u.purchase_mgmt()
    def salesmanagement():
        Window.destroy()
        u.sales_mgmt()
    def usermanagement():
        Window.destroy()
        u.user_mgmt()
    def dbmanagement():
```

```
    Window.destroy()
    u.db_mgmt()
def destroy():
    Window.destroy()
    exit()

#--------------------Main Window------------------

Window = Tk()
Window.title('Welcome to Inventory Manager')
Window.geometry('1025x684+0+0')
Window.config()

#--------------------Importing Images----------------

b = ImageTk.PhotoImage(Image.open("background.jpg"))
Button1 = ImageTk.PhotoImage(Image.open("Button1.png"))
Button2 = ImageTk.PhotoImage(Image.open("Button2.jpg"))
Button3 = ImageTk.PhotoImage(Image.open("Button3.jpg"))
Button4 = ImageTk.PhotoImage(Image.open("Button4.jpg"))
Button5 = ImageTk.PhotoImage(Image.open("Button5.jpg"))
Button6 = ImageTk.PhotoImage(Image.open("Button6.png"))
#---------------------Window Labels--------------------------
```

```
background = Label(Window, image=b)
background.place(x=0, y=0)

#---------------------Buttons----------------------------------
B1= Button(Window,bd=0,image=Button1,command=\
                                        productmanagement)
B1.place(x=225,y=280)
B2= Button(Window,bd=0,image=Button2,command=\
                                       purchasemanagement)
B2.place(x=555,y=280)
B3= Button(Window,bd=0,image=Button3,command=\
                                        salesmanagement)
B3.place(x=225,y=380)
B4= Button(Window,bd=0,image=Button4,command=
                                        usermanagement)
B4.place(x=555,y=380)
B5= Button(Window,bd=0,image=Button5,command=\
                                         dbmanagement)
B5.place(x=225,y=480)
B6 = Button(Window,image=Button6,bd=0,command=destroy)
B6.place(x=555, y=480)
Window.mainloop()
```

Upack.py

```
                         # INVENTORY MANAGEMENT
                         # ******UPACK*********
                            # ------Utkarsh-------
#**********************************************************
from sys import *
from tkinter import*
from tkinter import messagebox     # IMPORT SECTION
from tkinter import ttk
from PIL import ImageTk, Image
import os
import mysql.connector
import datetime
now = datetime.datetime.now()
password="1234”
#**********************************************************
                                   # SECTION I
def message1(titl,messag):
   a=Tk()
   a.title("Success")
   messagebox.showinfo(title=titl, message=messag,parent=a)
```

```
  # Functions For Messages
   a.destroy()
   return
def message(titl,messag):
   a=Tk()
   a.title("Error")
   messagebox.showerror(title=titl, message=messag,parent=a)
   a.destroy()
   return
def Text(Background,title,lc,tx):
 #Text Function-GUI text and Output
  bck=Tk()
  bck.geometry("1025x684")
  bck.title(title)
  bck.config()
  #------------------------------------------------------
  BG = ImageTk.PhotoImage(Image.open(Background))
  #-------------------------------------------------------
  Label(bck,image=BG).place(x=0,y=0)
  for a in range(len(lc)):
     l=Label(bck,text=tx[a], bd=5, bg='white',\
                          font=('ariel', 18)).place(x=(lc[a])[0],y=(lc[a])[1])
  bck.mainloop()
```

SECTION II

```
def scrn(title,background,cord,B1=0,C1=0,B2=0,C2=0,B3=0,\
                                    C3=0,B4=0,C4=0,B5=0,C5=0,B6=0):
  def back():
    window.destroy()
    import MenuScreen as msn     #Screen Function-Creates GUIs
    msn.mainscrn()
  def exe(a):
    window.destroy()
    a()
  #--------------------Window Creator------------------
  window = Tk()
  window.title(title)
  window.geometry('1025x684+0+0')
  window.config()

  #--------------------Importing Images-----------------

  b = ImageTk.PhotoImage(Image.open(background))
  if B1:
    Button1 = ImageTk.PhotoImage(Image.open(B1))
  if B2:
    Button2 = ImageTk.PhotoImage(Image.open(B2))
```

```
if B3:
    Button3 = ImageTk.PhotoImage(Image.open(B3))
if B4:
    Button4 = ImageTk.PhotoImage(Image.open(B4))
if B5:
    Button5 = ImageTk.PhotoImage(Image.open(B5))
if B6:
    Button6 = ImageTk.PhotoImage(Image.open(B6))
#--------------------window Labels------------------
background = Label(window,image=b)
background.place(x=0, y=0)
#--------------------Buttons---------------------------------
if B1:
    b1= Button(window,bd=0,image=Button1,command=\
                                                            lambda:exe(C1))
    b1.place(x=cord[0],y=cord[1])
if B2:
    b2= Button(window,bd=0,image=Button2,command=\
                                                           lambda:exe(C2))
    b2.place(x=cord[2],y=cord[3])
if B3:
    b3= Button(window,bd=0,image=Button3,command=\
                                                            lambda:exe(C3))
```

```
        b3.place(x=cord[4],y=cord[5])
    if B4:
        b4= Button(window,bd=0,image=Button4,command=\
                                                    lambda:exe(C4))
        b4.place(x=cord[6],y=cord[7])
    if B5:
        b5= Button(window,bd=0,image=Button5,command=\
                                                   lambda:exe(C5))
        b5.place(x=cord[8],y=cord[9])
    if B6:
        b6 = Button(window,image=Button6,bd=0,command=back)
        b6.place(x=cord[10], y=cord[11])
    window.mainloop()
#*****************************************************
                        #SECTION III
def product_mgmt():
    titl="Welcome to Product Management"
    bckrnd="background1.jpg"
    i1="Button11.jpg"
    i2="Button21.jpg"
    i3="Button31.jpg"
    i4="Button41.jpg"
    i5="Button61.jpg"
    cordinates=[225,280,555,280,225,380,555,380,0,0,400,480]
                                         #  Functions Using Screen
    c1= add_product                      #  Create Second GUI layer
    c2= list_product
    c3= update_product
    c4= delete_product
    scrn(titl,bckrnd,cordinates,i1,c1,i2,c2,i3,c3,i4,c4,B6=i5)
def purchase_mgmt() :
    titl="Welcome to Purchase Management"
    bckrnd="background2.jpg"
    i1="Button12.jpg"
    i2="Button22.jpg"
    i3="Button61.jpg"
    cordinates=[170,200,170,380,0,0,0,0,0,0,290,550]
    c1= add_order
    c2= list_order
    scrn(titl,bckrnd,cordinates,i1,c1,i2,c2,B6=i3)
def sales_mgmt():
    titl="Welcome to Sales Management"
    bckrnd="background3.jpg"
    i1="Button13.jpg"
    i2="Button23.jpg"
    i3="Button61.jpg"
    cordinates=[270,200,270,380,0,0,0,0,0,0,390,550]
    c1= sale_product
    c2= list_sale
    scrn(titl,bckrnd,cordinates,i1,c1,i2,c2,B6=i3)
def user_mgmt():
    titl="Welcome to User Management"
    bckrnd="background4.jpg"
    i1="Button14.jpg"
    i2="Button24.jpg"
    i3="Button61.jpg"
    cordinates=[270,200,270,380,0,0,0,0,0,0,390,550]
    c1= add_user
    c2= list_user
    scrn(titl,bckrnd,cordinates,i1,c1,i2,c2,B6=i3)
def db_mgmt( ):
    titl="Welcome to Database Management"
    bckrnd="background5.jpg"
    i1="Button15.jpg"
    i2="Button25.jpg"
    i3="Button61.jpg"
    cordinates=[270,200,270,380,0,0,0,0,0,0,390,550]
    c1= create_database
    c2= list_database
    scrn(titl,bckrnd,cordinates,i1,c1,i2,c2,B6=i3)
#***************************************************
                              #SECTION IV
entries=[]
def form(title,B,C,Entrycord,bi,bc,cmd):
    def exe(a):
        window.destroy()
        a()                                        # Form Function
    def retriever():                         # Creates Data Accepting GUIs
        entries.clear()
        for a in entry:
            entries.append(a.get())
        window.destroy()
        return
    window = Tk()
    window.title(title)
    window.geometry('1025x684+0+0')
    window.config()
    #--------------------Importing Images-----------------
    img=[]
    bimg=[]
    for a in range(len(B)):
        img.append(ImageTk.PhotoImage(Image.open(B[a])))
    for a in range(len(bi)):
        bimg.append(ImageTk.PhotoImage(Image.open(bi[a])))
    #---------------------Labels and Buttons------------------
    for b in range(len(img)):
        Label(window,image=img[b]).place(x=(C[b])[0],y=(C[b])[1])
    for b in range(1,len(bimg)):
        Button(window,image=bimg[b],bd=0,command\
                    =lambda:exe(cmd[b-1])).place(x=(bc[b])[0],y=(bc[b])[1])
    Button(window,image=bimg[0],bd=0,command=retriever)\
                                                  .place(x=(bc[0])[0],y=(bc[0])[1])
    #---------------------Entries-----------------------------
    entry=[]
    for a in range(len(Entrycord)):
        if Entrycord:
            entry.append(ttk.Entry(window,width=20,font=\
                                                        ('calibre', 15, 'bold')))
        entry[a].place(x=(Entrycord[a])[0],y=(Entrycord[a])[1])
    window.mainloop()
    return
#------------------------------------------------------------------------------------
                                             # SECTION V

def create_database():
```

```
try:
    mydb=mysql.connector.connect(host="localhost",user="root",\
                                                   password=password)
    mycursor = mydb.cursor()
    mycursor.execute("Create database stock")
    mycursor.execute("Use stock")
    message1("System","PRODUCT table Created")
    sql = "CREATE TABLE if not exists product\
                  (pcode int(4) PRIMARY KEY,\
                  pname char(30) NOT NULL, \
                  pprice float(8,2),\
                  pqty int(9),pcat char(30));"
    mycursor.execute(sql)                  #Functions using Form()
                                          #Creating thrid layer GUIs
   message1("System","ORDER table Created")
   sql = "CREATE TABLE if not exists orders\
                 (orderid int(4)PRIMARY KEY,\
                  orderdate DATE,\
                  pcode char(30) NOT NULL,\
                 pprice float(8,2),\
                 pqty int(4),\
                supplier char(50),\
               pcat char(30));"
    mycursor.execute(sql)
    message1("System","SALES table Created")
    sql = "CREATE TABLE if not exists sales\
                    (salesid int(4) PRIMARY KEY,\
                    salesdate DATE,\
                   pcode char(30) references product(pcode),\
                  pprice float(8,2),\
                 pqty int(4),\
                Total double(8,2));"
    mycursor.execute(sql)
    message1("System","USER table Created")
    sql = "CREATE TABLE if not exists user\
          (uid char(60) PRIMARY KEY,\
          uname char(30) NOT NULL,\
          upwd char(30));"
    mycursor.execute(sql)
    message1("Success", "Database Is Ready")
    message1("NOTE","Default login id='root' and password\
                                                    =”1234”)
  except:
    message("NOTE","SETUP IS PREINSTALLED")
 exit()
#---------------------------------------------------------------------------
```

```
def list_database():
  mydb = mysql.connector.connect(host="localhost", user="root",\
                        password=password,database="stock")
  mycursor = mydb.cursor()
  sql = "show tables;"
  mycursor.execute(sql)
  d=mycursor.fetchall()
  c=0
  ka="Database has following tables"
  b="""S.No\t   Table    \t
____________________________
"""
  for a in (d):
     for g in a:
      c=c+1
      b+=str(c)+"\t"+(g)+"""
____________________________
 """
  Text("background5.jpg","Database",[[350,200],[350,300]],[ka,b])
  exit()
#__________________________________________________________
def add_order():
  img=["background244.jpg","label11.jpg","label22.jpg",\
                              "label31.jpg","label41.jpg","label51.jpg"]
imc=[[0,0],[150,100],[150,200],[150,300],[150,400],[150,500]]
ent=[[500,110],[500,210],[500,310],[500,410],[500,510]]
bim=["Enter.jpg","Button61.jpg"]
btc=[[159,590],[509,590]]
bcm=[purchase_mgmt]
form("Add Order",img,imc,ent,bim,btc,bcm)
mydb = mysql.connector.connect(host="localhost", user="root",\
        password=password, database="stock")
mycursor = mydb.cursor()

sql = "INSERT INTO orders (orderid, orderdate, pcode,\
          pprice, pqty, supplier, pcat) values\
          (%s,%s,%s,%s,%s,%s,%s)"
now = datetime.datetime.now()
oid = now.year+now.month+now.day+now.hour+\
                                        now.minute+now.second
if entries[0]:
   pcode = entries[0]
else:
   message("Error","Enter all details")
   add_order()
search = "SELECT count(*) FROM orders WHERE orderid=%s;"
```

```
val = (oid,)
mycursor.execute(search,val)
for x in mycursor:
    cnt = x[0]
if cnt == 0:
    if entries[1]:
        supplier = entries[1]
    else:
        message("Error","Enter all details")
        add_order()
    if entries[2]:
        pprice =  entries[2]
    else:
        message("Error","Enter all details")
        add_order()
    if entries[3]:
        pqty = entries[3]
    else:
        message("Error","Enter all details")
        add_order()
    if entries[4]:
        pcat = entries[4]
    else:
        message("Error","Enter all details")
        add_order()
      val = (oid, now, pcode, pprice, pqty, supplier, pcat)
     mycursor.execute(sql, val)
     mydb.commit()
     message1("Success","Order created")
     exit()
  else:
     message("Error","Order Already exists")
     add_order()
#-------------------------------------------------------------------------------------
def list_order():
  mydb = mysql.connector.connect(host="localhost", user="root",\
            password=password,database="stock")
  mycursor = mydb.cursor()
  sql = "SELECT * from orders;"
  mycursor.execute(sql)
  d=mycursor.fetchall()
  ka="THE ORDER DETAILS ARE-"
  b=("orderid | date | code | price | quantity |\
                                             supplier | category\n")
  b+=("-" *92+"\n")
  for k in d:
      for i in range(len(k)):
        b+=str(k[i])+" | "
     b+=("\n"+"-" *92+"\n")
  Text("background2.jpg","ORDERS",[[350,170],[150,220]],[ka,b])
  exit()
#--------------------------------------------------------------------------------
def add_product():
  img=["background1.jpg","label11.jpg","label21.jpg",\
                                  "label31.jpg","label41.jpg","label51.jpg"]
  imc=[[0,0],[250,100],[250,200],[250,300],[250,400],[250,500]]
  ent=[[550,110],[550,210],[550,310],[550,410],[550,510]]
  bim=["Enter.jpg","Button61.jpg"]
  btc=[[250,590],[600,590]]
  bcm=[product_mgmt]
  form("Add Product",img,imc,ent,bim,btc,bcm)
  mydb = mysql.connector.connect(host="localhost", user="root",\
              password=password,database="stock")
  mycursor = mydb.cursor()
  sql = "INSERT INTO product\
            (pcode,pname,pqty,pprice,pcat) values\
           (%s,%s,%s,%s,%s)"
  if entries[0]:
     code = entries[0]
```

```
else:
   message("Error","Enter all details")
   add_product()
search = "SELECT count(*) FROM product WHERE pcode=%s;"
val = (code,)
mycursor.execute(search,val)
for x in mycursor:
   cnt = x[0]
if cnt == 0:
   if entries[1]:
      name = entries[1]
   else:
      message("Error","Enter all details")
      add_product()
   if entries[2]:
      qty =  entries[2]
   else:
      message("Error","Enter all details")
      add_product()
   if entries[3]:
      price = entries[3]
   else:
      message("Error","Enter all details")
```

```
          add_product()
        if entries[4]:
          cat = entries[4]
        else:
          message("Error","Enter all details")
          add_product()
        val = (code,name,price,qty,cat)
        mycursor.execute(sql,val)
        mydb.commit()
        message1("Success","Product created")
        exit()

      else:
        message("Error","Product Already exists")
        add_product()
#-------------------------------------------------------------------------------------
def update_product():
  img=["background1.jpg","label11.jpg","label41.jpg"]
  imc=[[0,0],[250,200],[250,300]]
  ent=[[600,210],[600,310]]
  bim=["Enter.jpg","Button61.jpg"]
  btc=[[400,400],[400,500]]
  bcm=[product_mgmt]
```

```
form("Update Product",img,imc,ent,bim,btc,bcm)
mydb = mysql.connector.connect(host="localhost", user="root",\
                          password=password,database="stock")
mycursor = mydb.cursor()
sql = "UPDATE product SET pqty=pqty+%s WHERE pcode=%s;"
if entries[0]:
   code = entries[0]
else:
   message("Error","Enter all details")
   update_product()

val = (code,)
search = "SELECT count(*) FROM product WHERE pcode=%s;"
mycursor.execute(search,val)
for x in mycursor:
   cnt = x[0]
if cnt != 0:
   if entries[1]:
      qty = entries[1]
   else:
      message("Error","Enter all details")
      update_product()
   val = (qty,code)
```

```
        mycursor.execute(sql,val)
        mydb.commit()
        message1("Success","Product Quantity updated")
        exit()
    else:
        message("Error","Product doesn't exist")
        update_product()
#----------------------------------------------------------------------------------
def delete_product():
    img=["background1.jpg","label11.jpg"]
    imc=[[0,0],[250,250]]
    ent=[[600,260]]
    bim=["Enter.jpg","Button61.jpg"]
    btc=[[400,400],[400,500]]
    bcm=[product_mgmt]
    form("Delete Product",img,imc,ent,bim,btc,bcm)
    mydb = mysql.connector.connect(host="localhost", user="root",\
               password=password,database="stock")
    mycursor = mydb.cursor()
    sql = "DELETE FROM product WHERE pcode = %s;"
    if entries[0]:
        code = entries[0]
    else:
        message("Error","Enter Product Code")
        delete_product()

    val = (code,)
    search = "SELECT count(*) FROM product WHERE pcode=%s;"
    mycursor.execute(search,val)
    for x in mycursor:
        cnt = x[0]
    if cnt != 0:
        val = (code,)
        mycursor.execute(sql,val)
        mydb.commit()
        message1("Success","Product Deleted")
        exit()
    else:
        message("Error","Product doesn't exist")
        delete_product()
#____________________________________________________________
def list_product():
    mydb = mysql.connector.connect(host="localhost", user="root",\
                  password=password,database="stock")
    mycursor = mydb.cursor()
    sql = "SELECT * from product"
```

```
   mycursor.execute(sql)
   d=mycursor.fetchall()
   ka="PRODUCT DETAILS ARE-"
   b=(" code | name | price | quantity | category\n")
   b+=("-" *92+"\n")
   for k in d:
      for i in range(len(k)):
         b+=str(k[i])+" | "
      b+=("\n"+"-" *92+"\n")
   Text("background2.jpg","Products",[[370,20],[150,80]],[ka,b])
   exit()
#_____________________________________________________________
def sale_product():
  img=["background3.jpg","label11.jpg","label41.jpg"]
  imc=[[0,0],[250,200],[250,300]]
  ent=[[550,210],[550,310]]
  bim=["Enter.jpg","Button61.jpg"]
  btc=[[250,450],[600,450]]
  bcm=[sales_mgmt,]
  form("Sale Product",img,imc,ent,bim,btc,bcm)
  mydb = mysql.connector.connect(host="localhost", user="root",\
                 password=password, database="stock")
  mycursor = mydb.cursor()
```

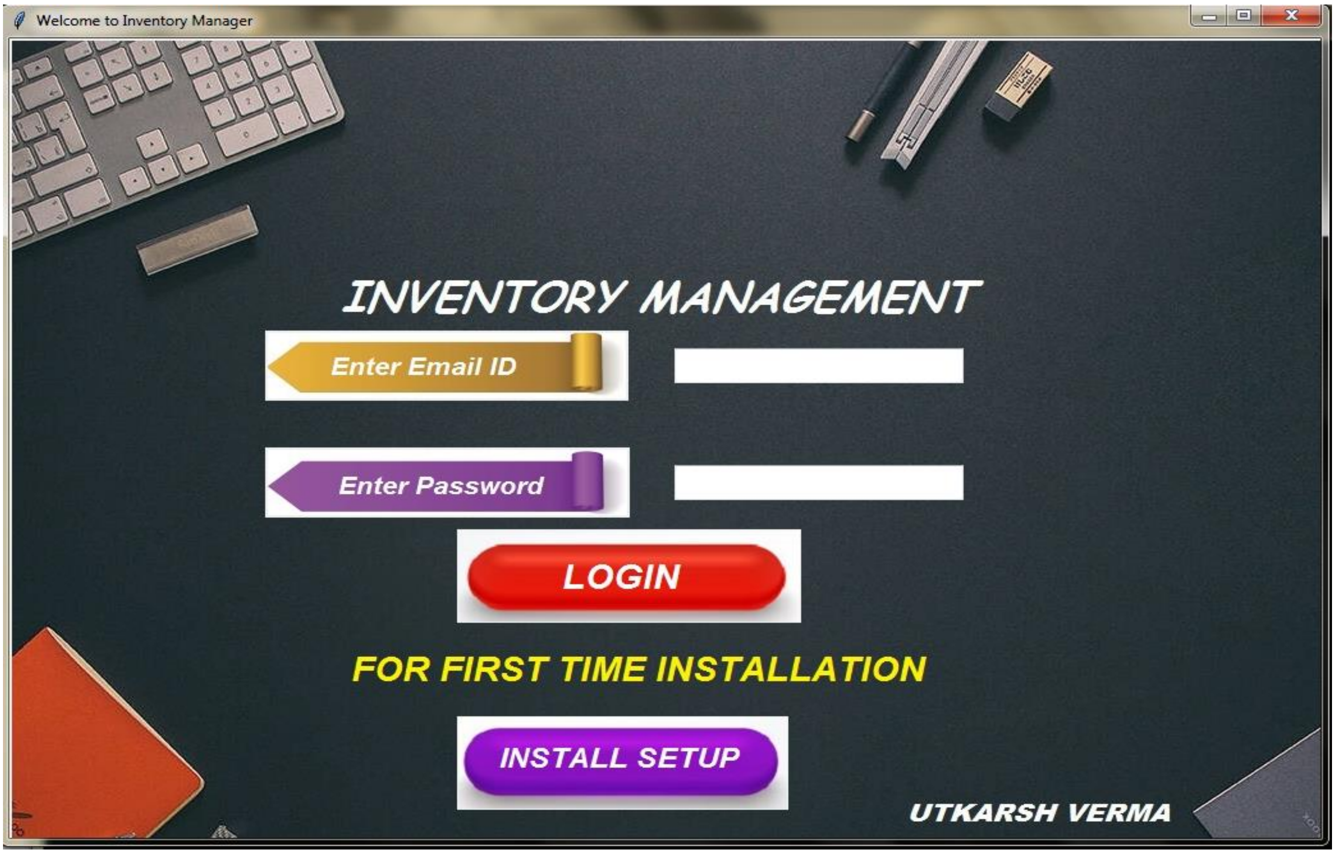
```
pcode = entries[0]
sql = "SELECT count(*) from product WHERE pcode=%s;"
val=(pcode,)
mycursor.execute(sql,val)
for x in mycursor:
  cnt = x[0]
if cnt != 0:
  sql = "SELECT * from product WHERE pcode=%s;"
  val = (pcode,)
  mycursor.execute(sql, val)
  for x in mycursor:
    price = int(x[2])
    pqty = int(x[3])
    qty=int(entries[1])
    if qty <= pqty :
      total= qty * price
      ss="Collect Rs. "+ str(total)+"\n Product data="+str(x)
      message1("Success",ss)
      mycursor.execute("Select count(*) from sales")
      for x in mycursor:
          count=x[0]+1
      sql = "INSERT into sales values(%s,%s,%s,%s,%s,%s)"
      val = (count,datetime.datetime.now(),pcode,price,qty,total)
```

```
        mycursor.execute(sql,val)
        sql = "UPDATE product SET pqty=pqty-%s WHERE pcode=%s"
        val = (qty, pcode)
        mycursor.execute(sql, val)
        mydb.commit()
      else:
        message("Error","Quantity not available")
  else:
      message("Error","Product is not available")
  exit()
#_______________________________________________________
def list_sale():
      mydb=mysql.connector.connect(host="localhost", user="root",\
                                                 password=password,database="stock")
      mycursor = mydb.cursor()
      sql = "SELECT * FROM sales"
      mycursor.execute(sql)
      d=mycursor.fetchall()
      ka="SALES DETAILS ARE-"
      b="Sales ID | Date | Code | Price | Quantity | Total\n"
      b+=("-" *92+"\n")
      for k in d:
       for i in range(len(k)):
        b+=str(k[i])+" | "
      b+=("\n"+"-" *92+"\n")
     Text("background2.jpg","SALES",[[370,20],[150,80]],[ka,b])
     exit()
#-------------------------------------------------------------------------------------
def add_user():
    try:
        img=["background4.jpg","label77.jpg","label88.jpg",\
                                                      "label99.jpg"]
        imc=[[0,0],[250,200],[250,300],[250,400]]
        ent=[[550,210],[550,310],[550,410]]
        bim=["Enter.jpg","Button61.jpg"]
        btc=[[250,590],[600,590]]
        bcm=[user_mgmt,]
        form("ADD USER",img,imc,ent,bim,btc,bcm)
        mydb = mysql.connector.connect(host="localhost",\
                user="root",password=password, database="stock")
        mycursor= mydb.cursor()
        uid = entries[0]
        name = entries[1]
        passwor = entries[2]
        sql = "INSERT INTO user values (%s,%s,%s);"
        val = (uid, name, passwor)
      mycursor.execute(sql, val)
      mydb.commit()
      message1("Success", "User created")
   except:
      message("Error","Login id is already taken")
   exit()
#------------------------------------------------------------------------------------
def list_user():
    mydb=mysql.connector.connect(host="localhost", user="root",\
                        password=password, database="stock")
    mycursor= mydb.cursor()
    sql = "SELECT uid, uname from user"
    mycursor.execute(sql)
    d=mycursor.fetchall()
    ka="USER DETAILS ARE-"
    b="USER ID | NAME\n"
    b+=("-" *92+"\n")
    for k in d:
     for i in range(len(k)):
        b+=str(k[i])+" | "
     b+=("\n"+"-" *92+"\n")
    Text("background2.jpg","USERS",[[370,20],[150,80]],[ka,b])
    exit()
```

So here ends the source code of all 4 python files.Now we will go through Sample outputs generated by the program and control flow.

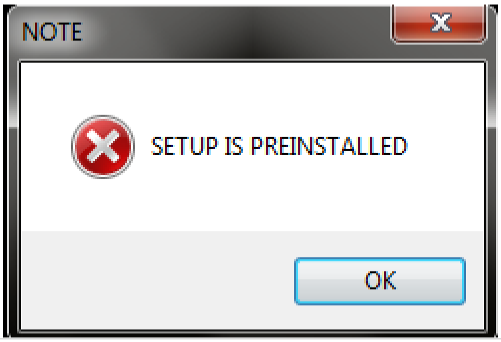
SAMPLE WORKING AND OUTPUTS

The program opens with this window-



Here for first time installation we need to click on INSTALL SETUP Button. After the installation the program provides default login and password using which we can login for the first time and then we can create new user id and password as we will se.

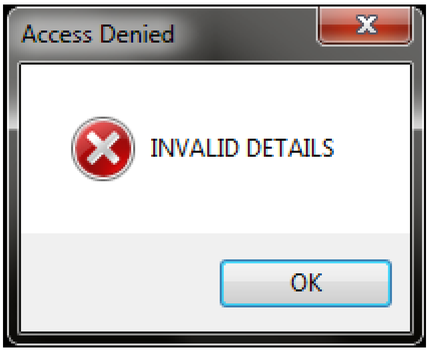
NOTE: Once program is installed, re-clicking the install setup button will generate an error as-



NOTE

SETUP IS PREINSTALLED

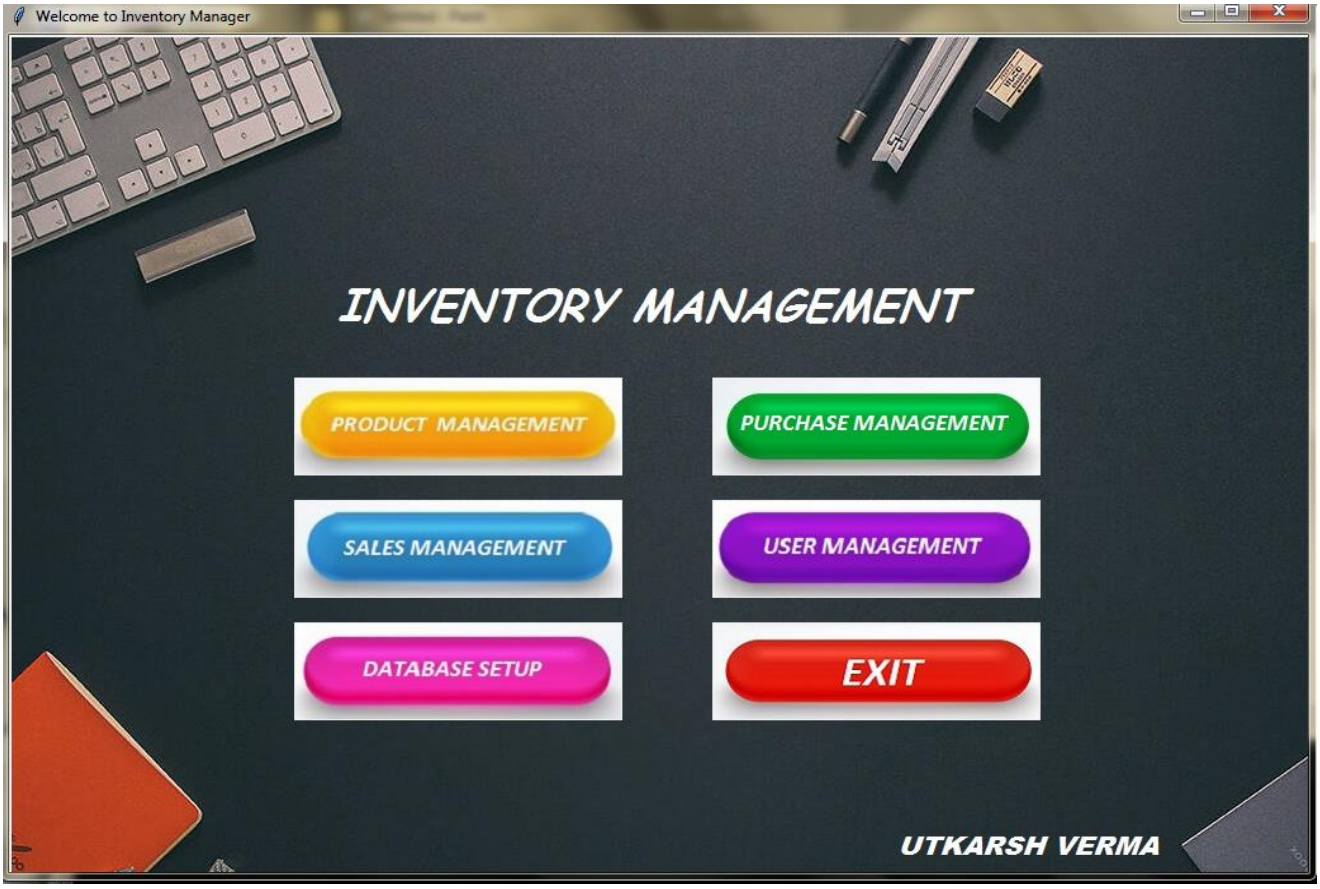
We can login with the default id and password or with self created User ids and passwords once setup is installed. Trying to logging in without valid ids and passwords will deny the access.



Access Denied

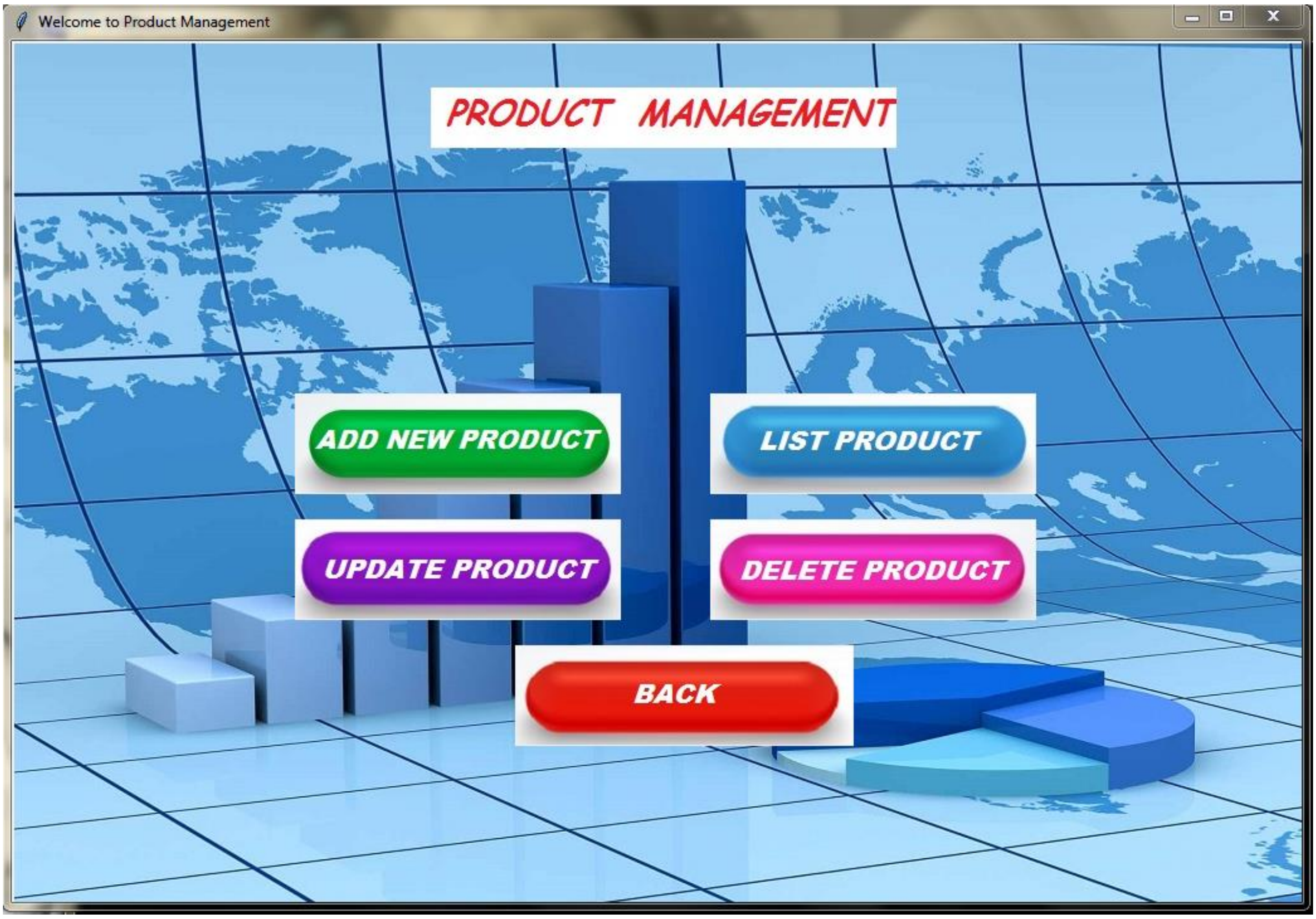
INVALID DETAILS

After successfully logging in the main menu opens like-



This window then takes users to the windows associated with given 5 buttons while the EXIT button closes the program.

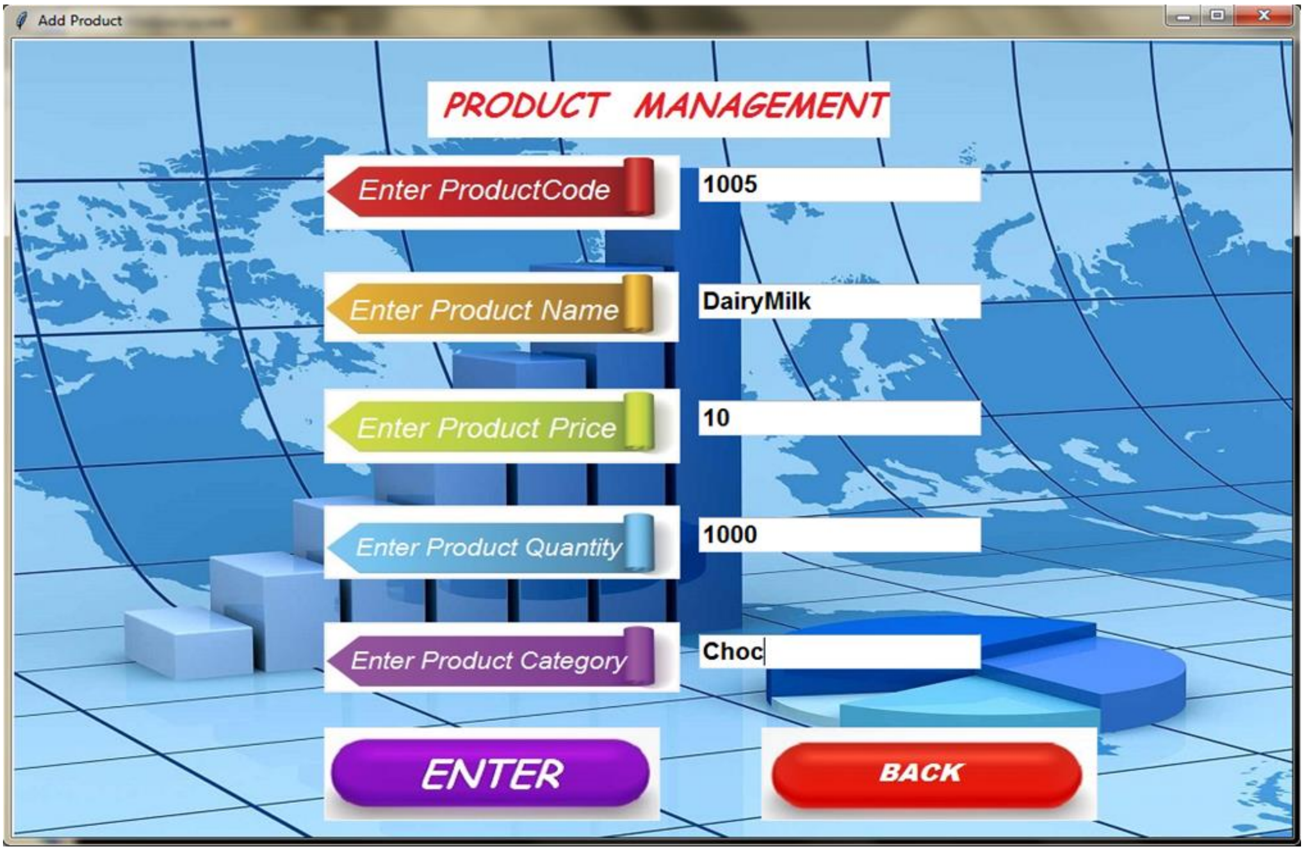
The Product Management Section opens like-



The Buttons provided open their associated options while the back button takes user again to Home Screen.Using update we increase the stock quantity of a particular product.

Now lets have a look of each sub menu of this window :

ADD PRODUCT SCREEN



Enter ProductCode: 1005

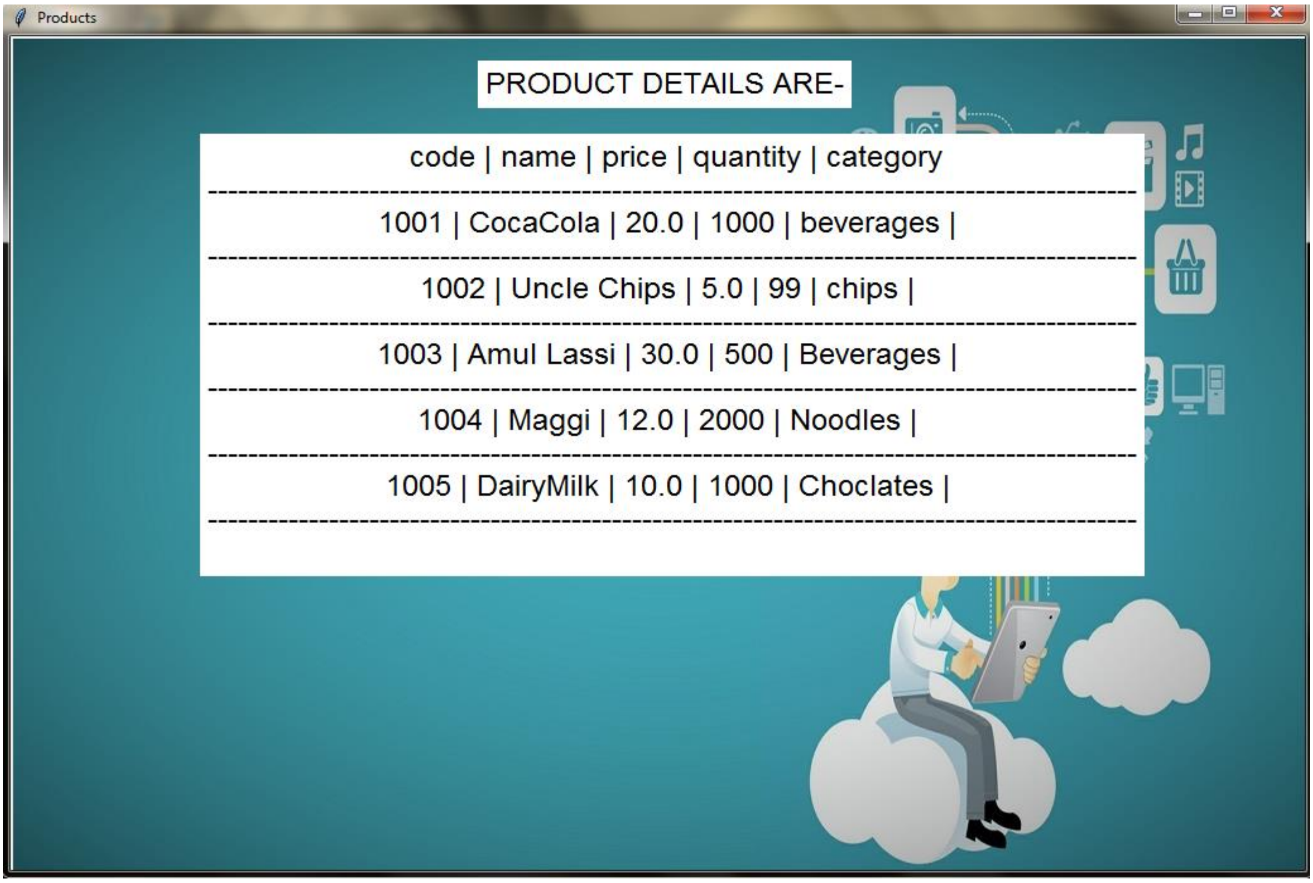
Enter Product Name: DairyMilk

Enter Product Price: 10

Enter Product Quantity: 1000

Enter Product Category: Choc

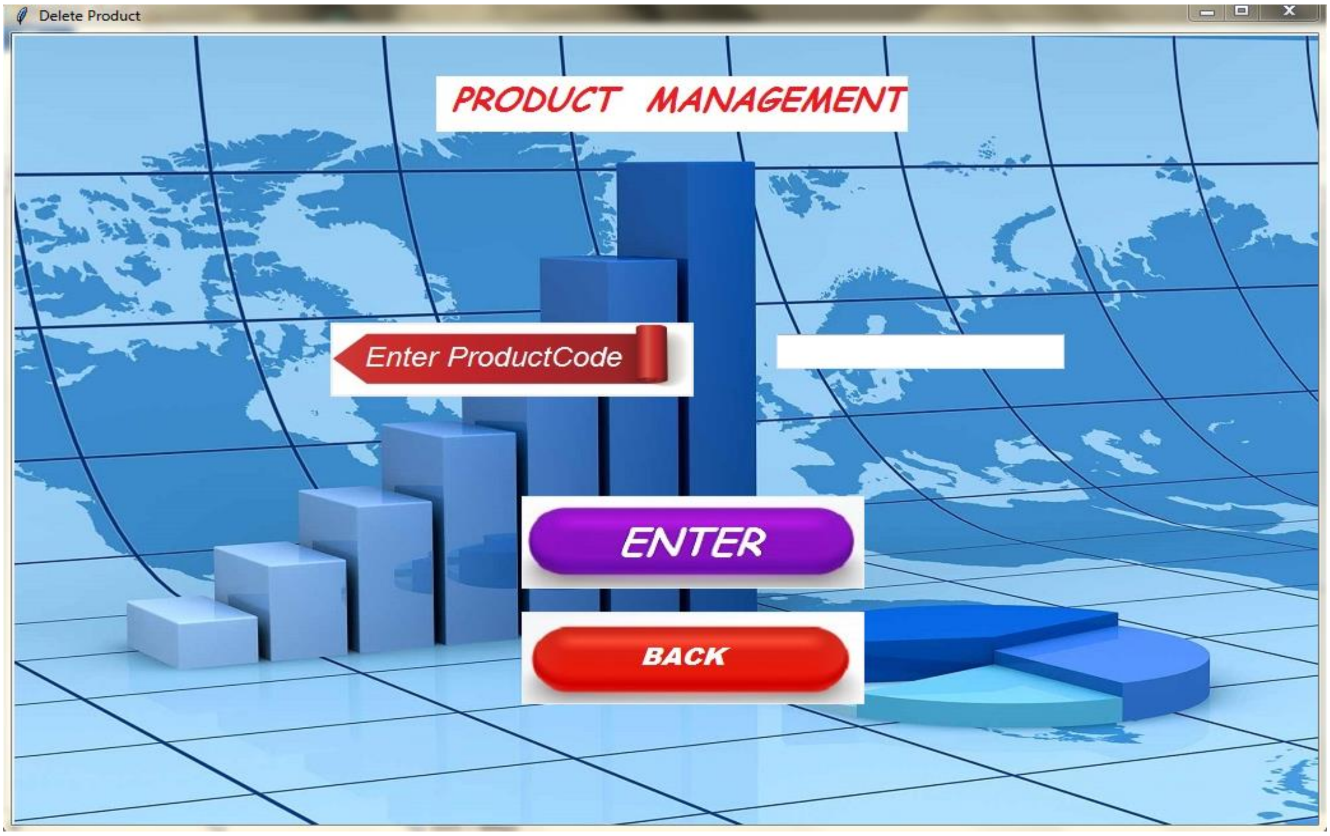
LIST PRODUCT SCREEN



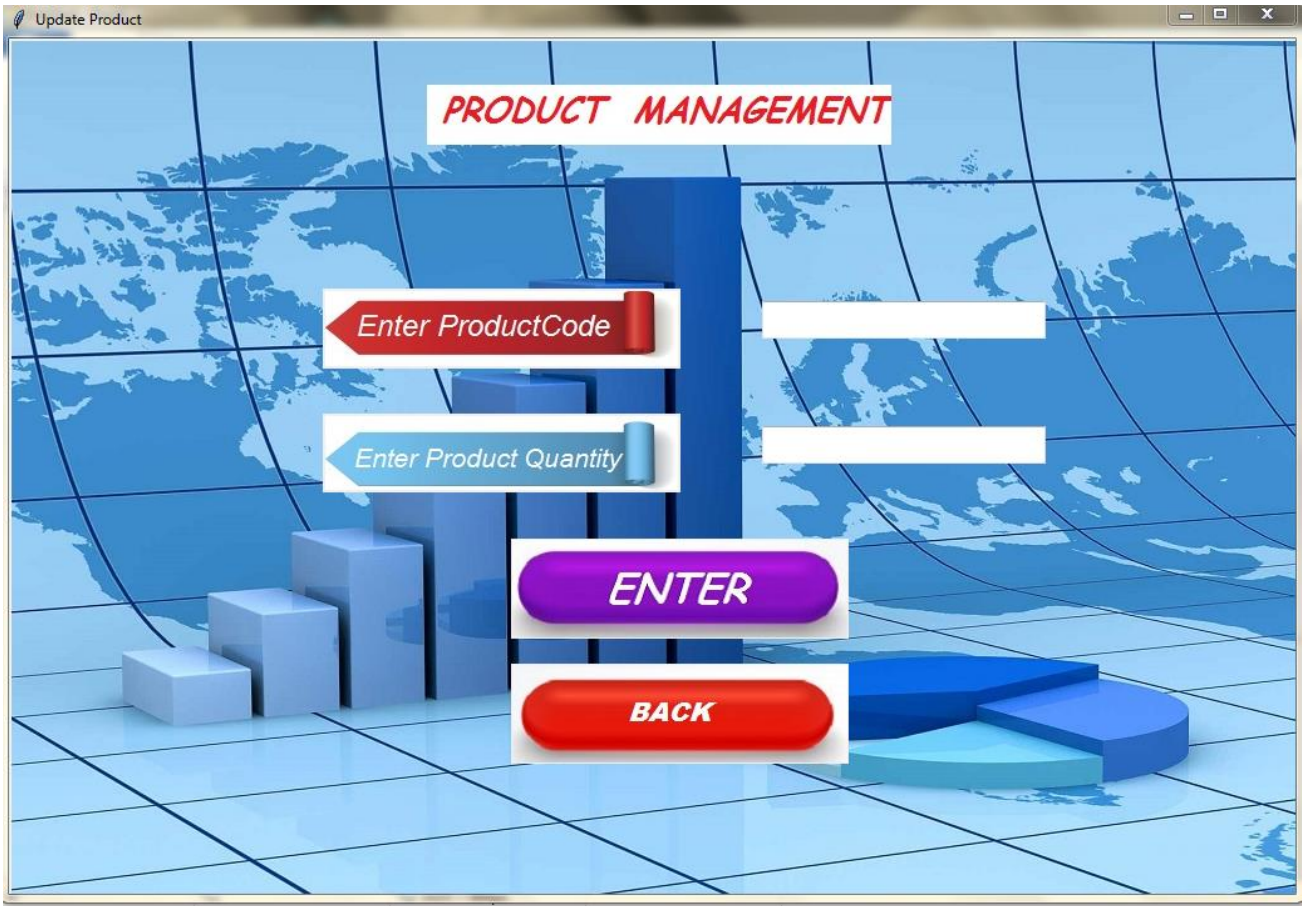
```
PRODUCT DETAILS ARE-

code | name | price | quantity | category
--------------------------------------------------------------------------------
1001 | CocaCola | 20.0 | 1000 | beverages |
--------------------------------------------------------------------------------
1002 | Uncle Chips | 5.0 | 99 | chips |
--------------------------------------------------------------------------------
1003 | Amul Lassi | 30.0 | 500 | Beverages |
--------------------------------------------------------------------------------
1004 | Maggi | 12.0 | 2000 | Noodles |
--------------------------------------------------------------------------------
1005 | DairyMilk | 10.0 | 1000 | Choclates |
--------------------------------------------------------------------------------
```

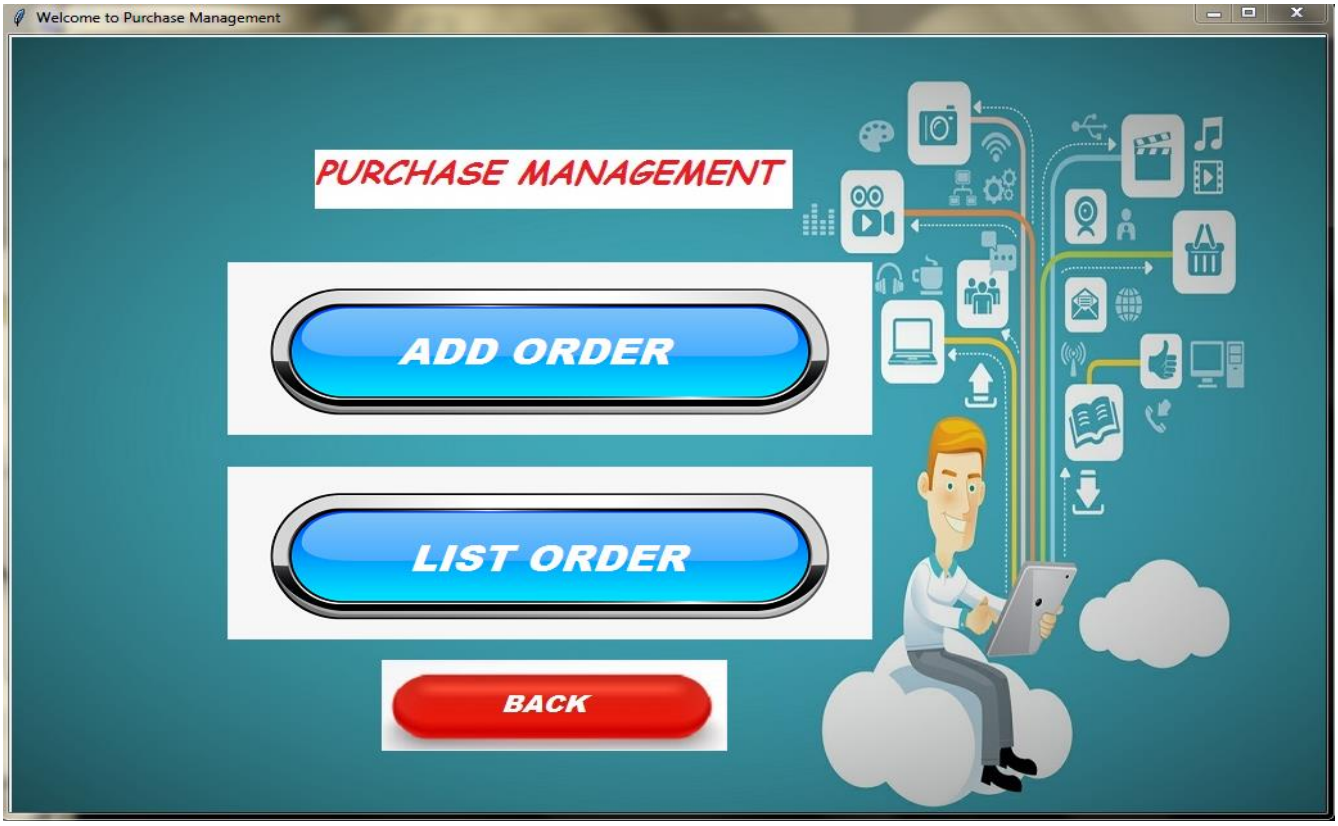
DELETE PRODUCT SCREEN



UPDATE PRODUCT SCREEN

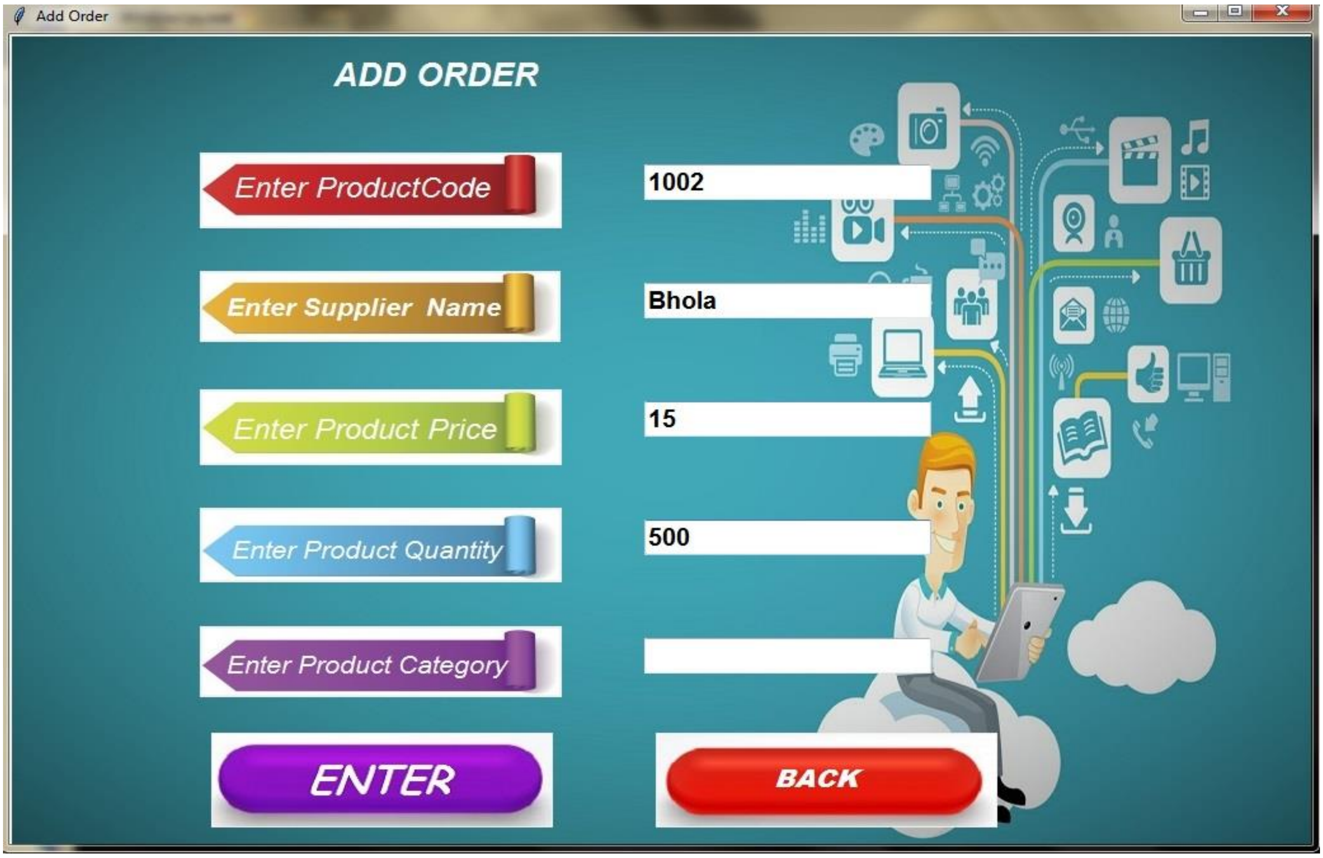


The Purchase Management Section looks like-



This Window provides options to add and list all orders

ADD ORDER SCREEN.



Enter ProductCode: 1002

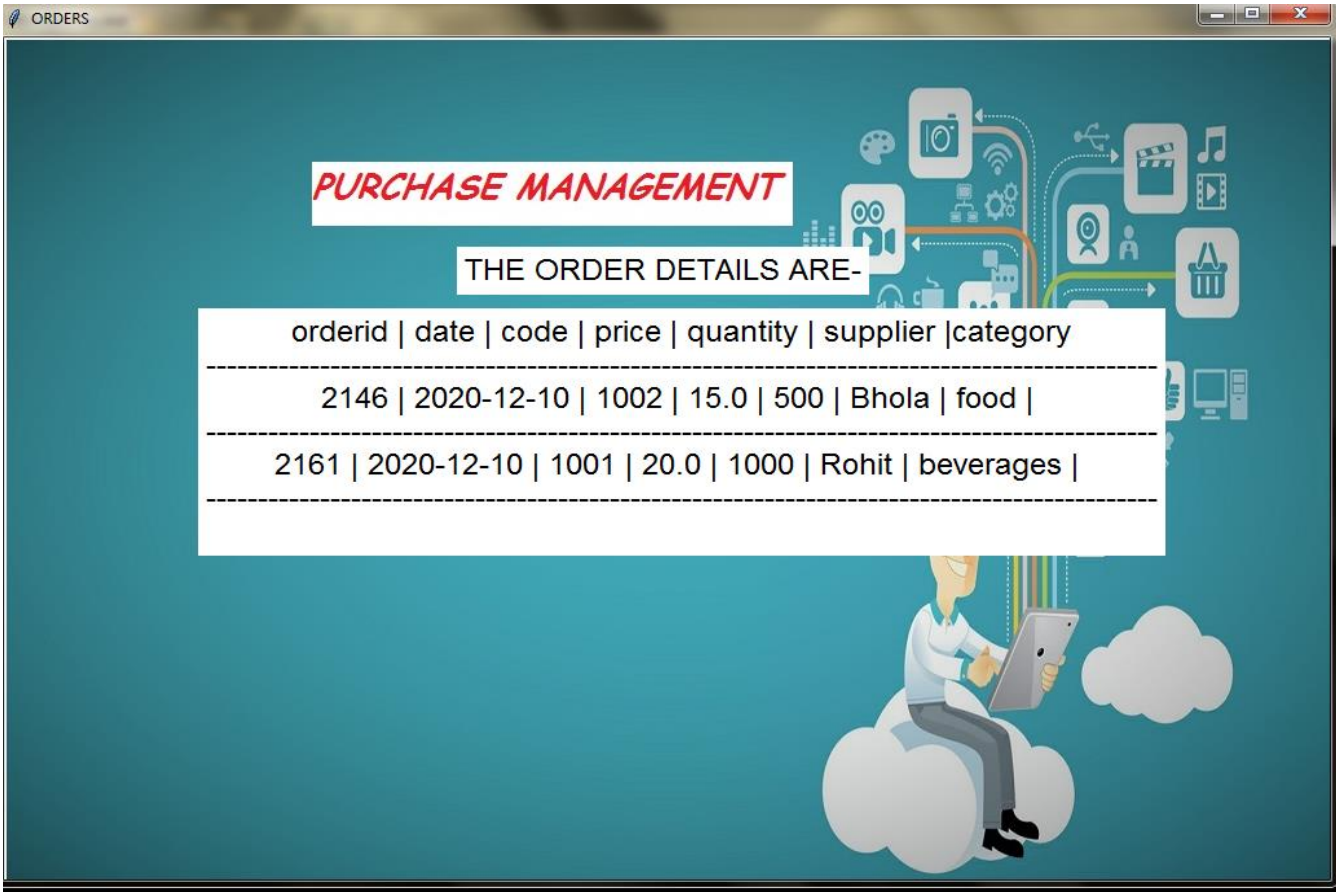
Enter Supplier Name: Bhola

Enter Product Price: 15

Enter Product Quantity: 500

Enter Product Category:

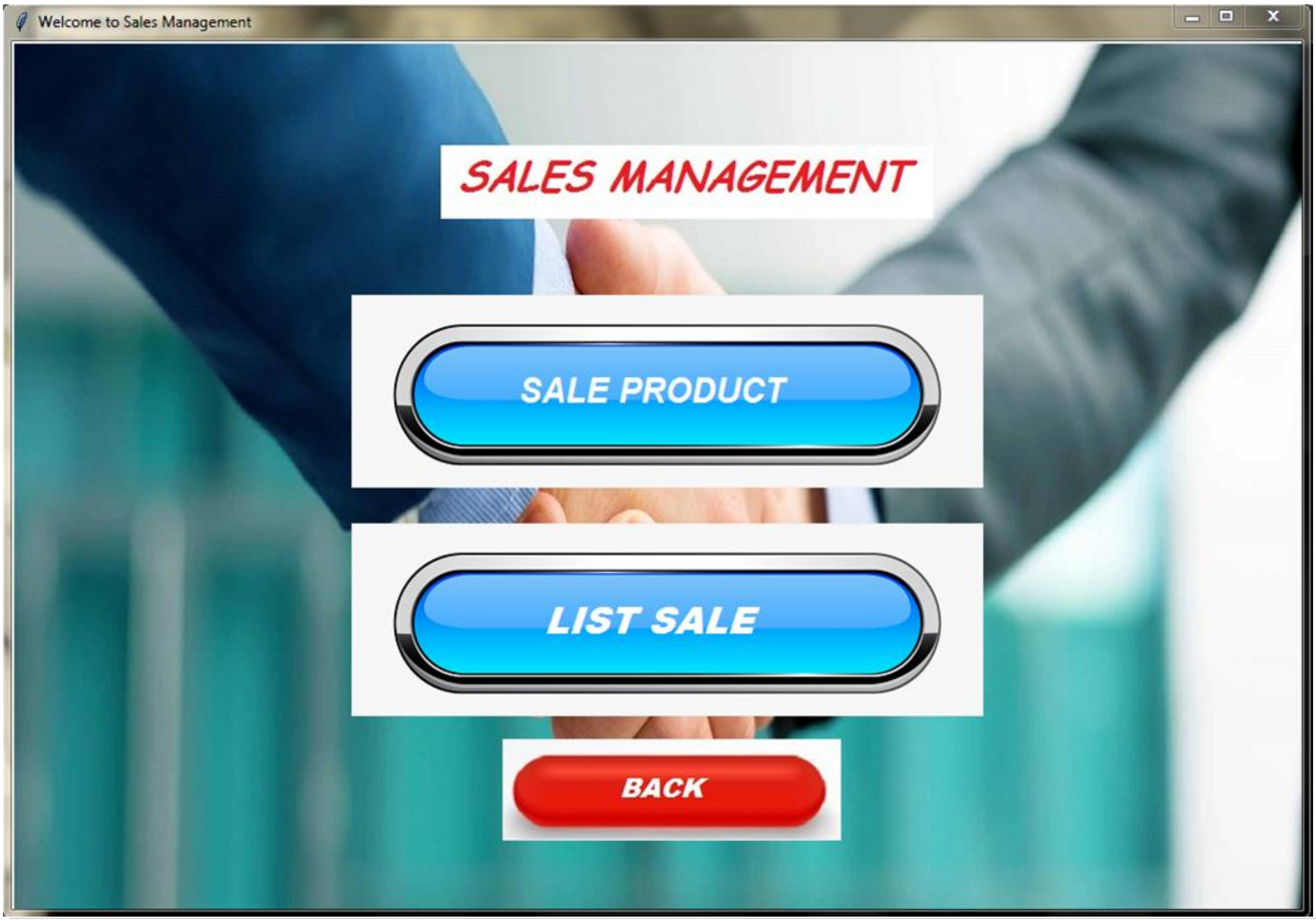
LIST ORDER SCREEN



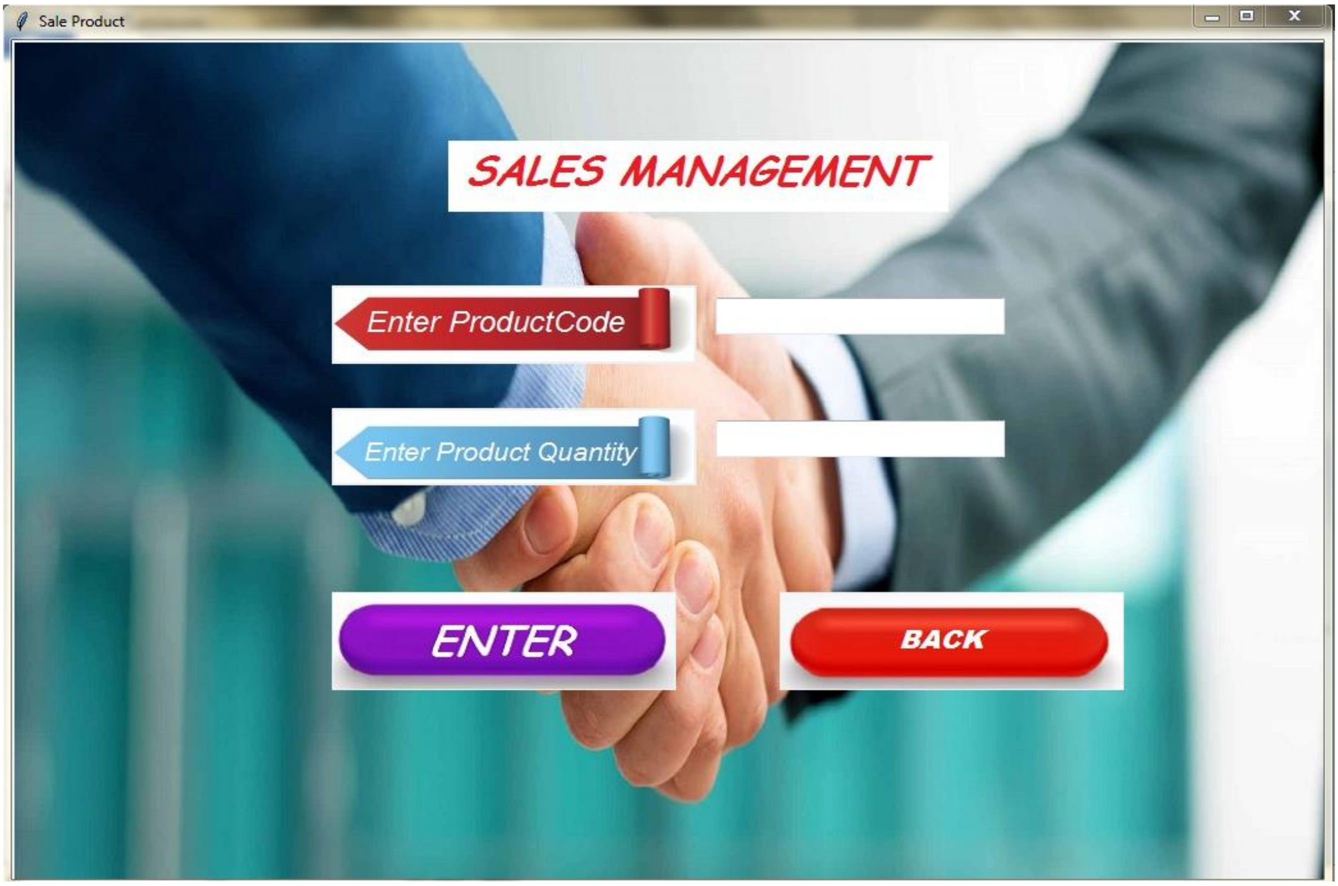
THE ORDER DETAILS ARE-

```
orderid | date | code | price | quantity | supplier |category
---------------------------------------------------------------------------------------
2146 | 2020-12-10 | 1002 | 15.0 | 500 | Bhola | food |
---------------------------------------------------------------------------------------
2161 | 2020-12-10 | 1001 | 20.0 | 1000 | Rohit | beverages |
---------------------------------------------------------------------------------------
```

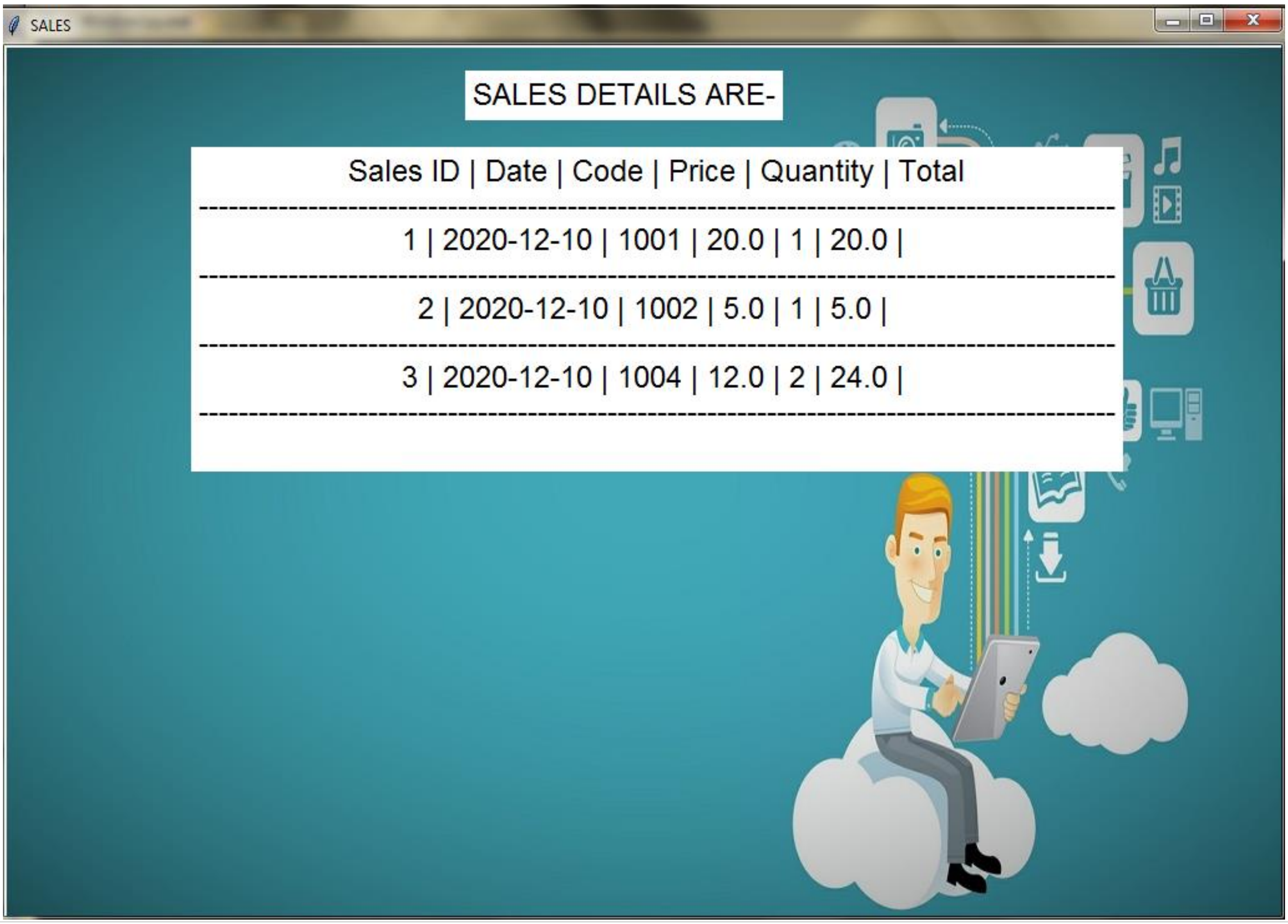
The Sales Management Section looks like-



SALE PRODUCT SCREEN



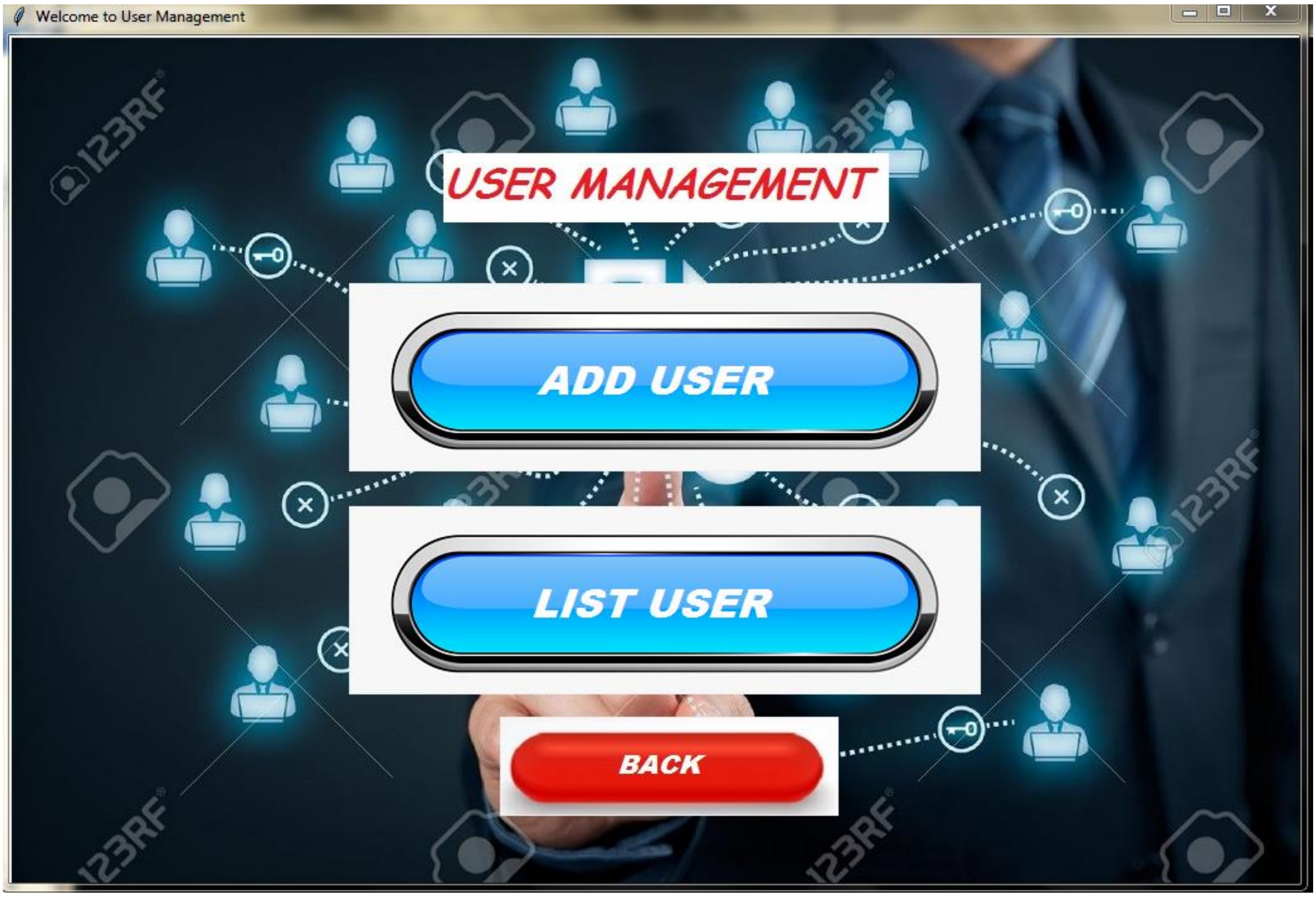
LIST SALE SCREEN



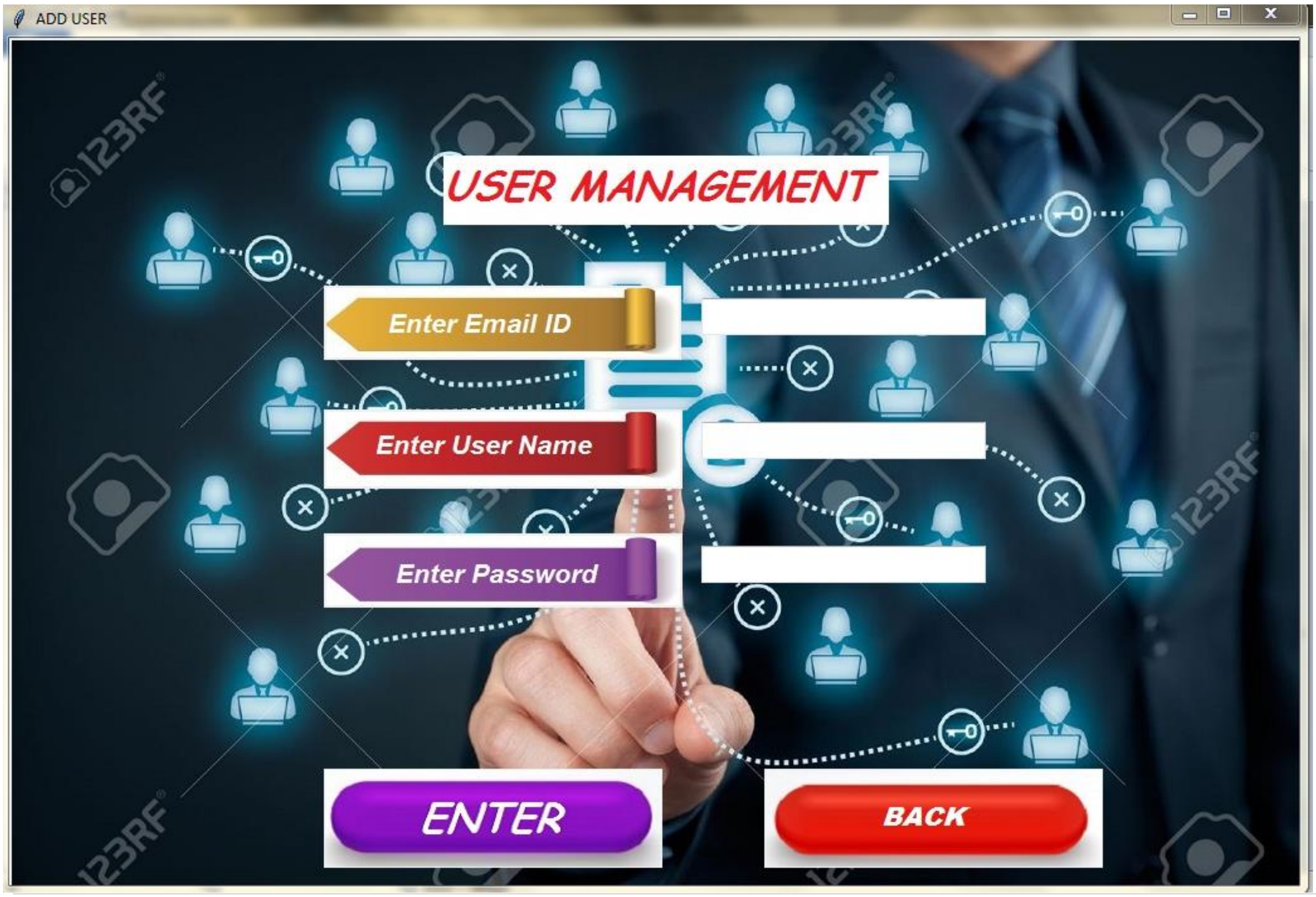
```
SALES DETAILS ARE-

Sales ID | Date | Code | Price | Quantity | Total
--------------------------------------------------------------------------------
1 | 2020-12-10 | 1001 | 20.0 | 1 | 20.0 |
--------------------------------------------------------------------------------
2 | 2020-12-10 | 1002 | 5.0 | 1 | 5.0 |
--------------------------------------------------------------------------------
3 | 2020-12-10 | 1004 | 12.0 | 2 | 24.0 |
--------------------------------------------------------------------------------
```

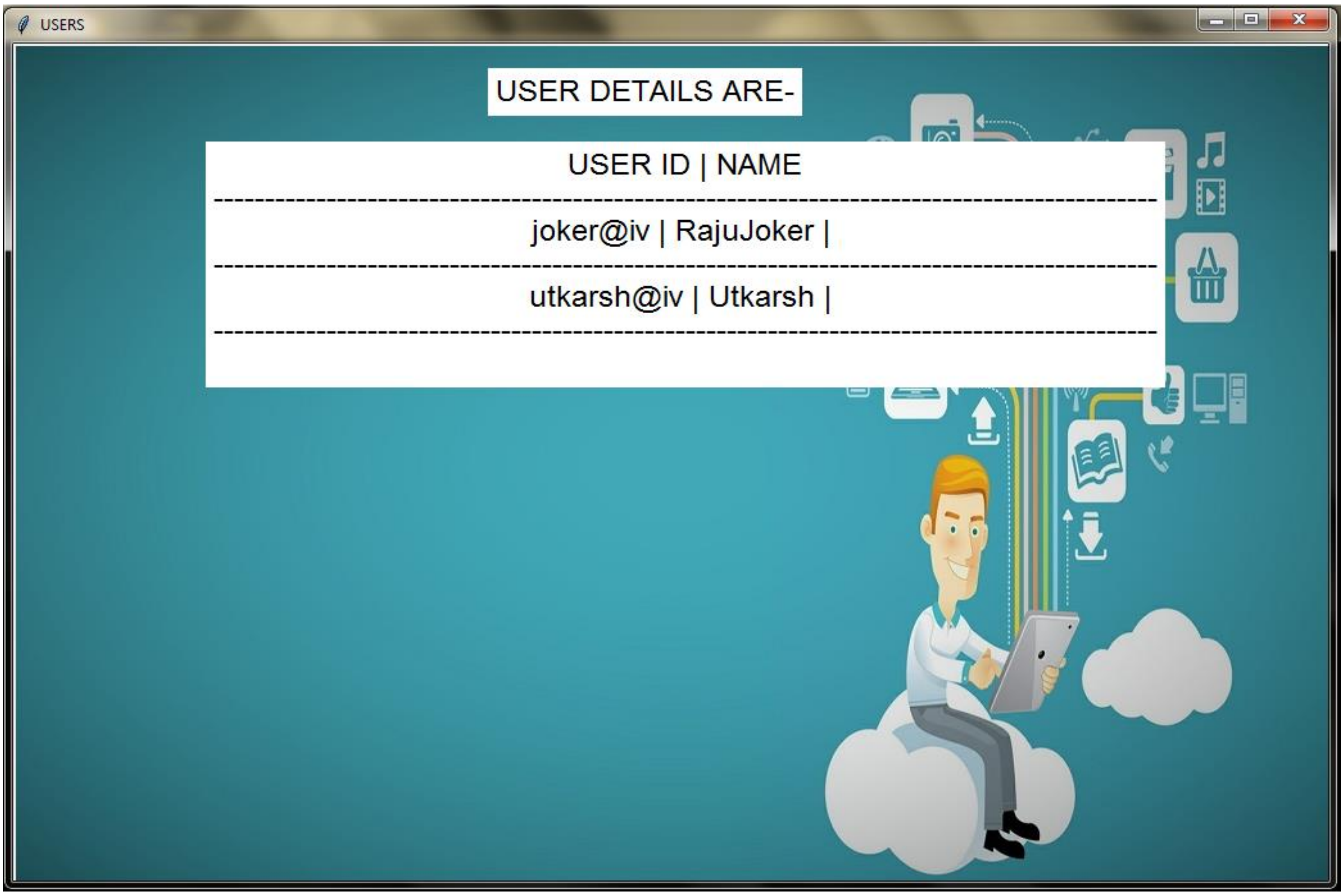
The User Management Section looks like-



ADD USER SCREEN



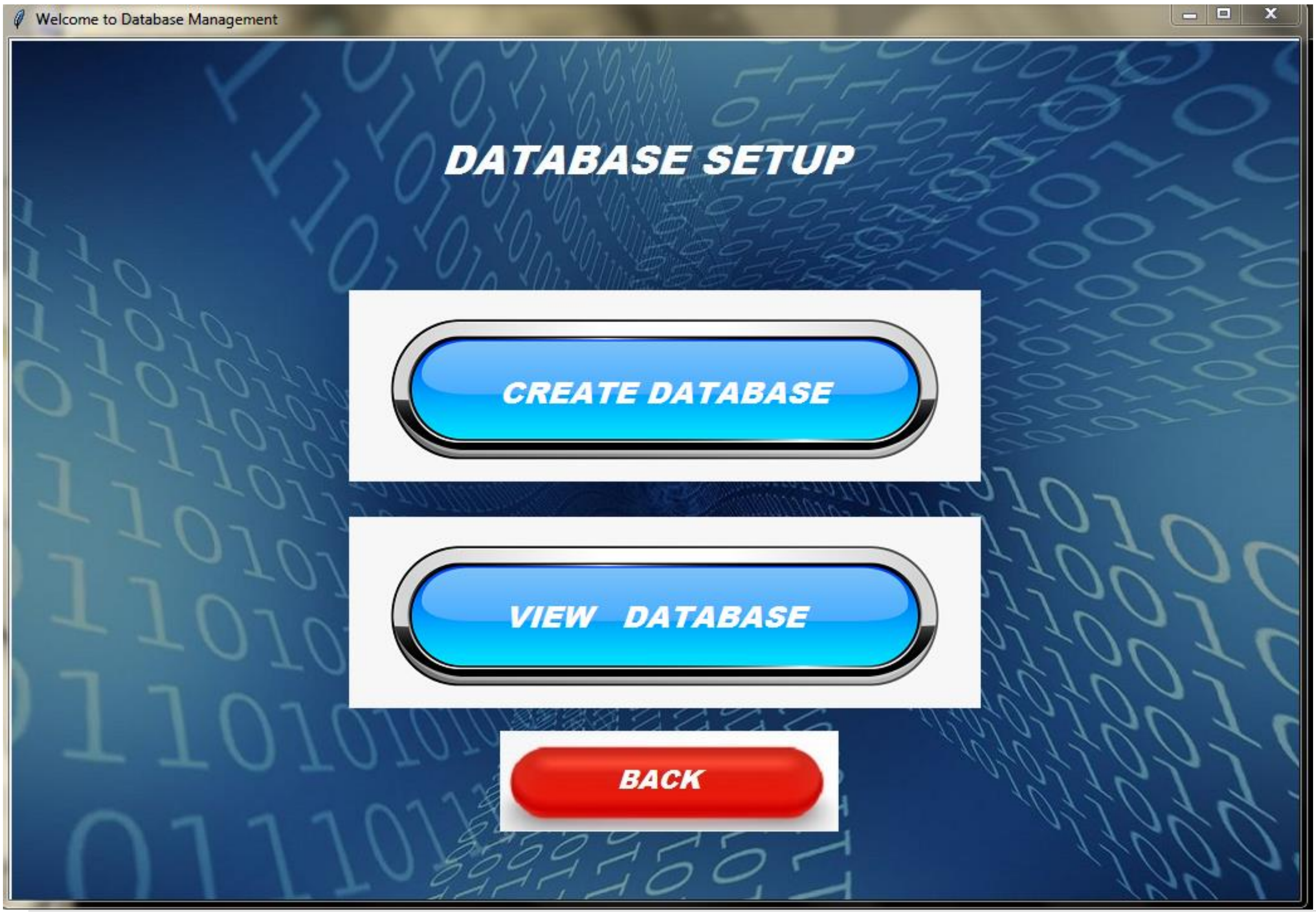
LIST USER SCREEN



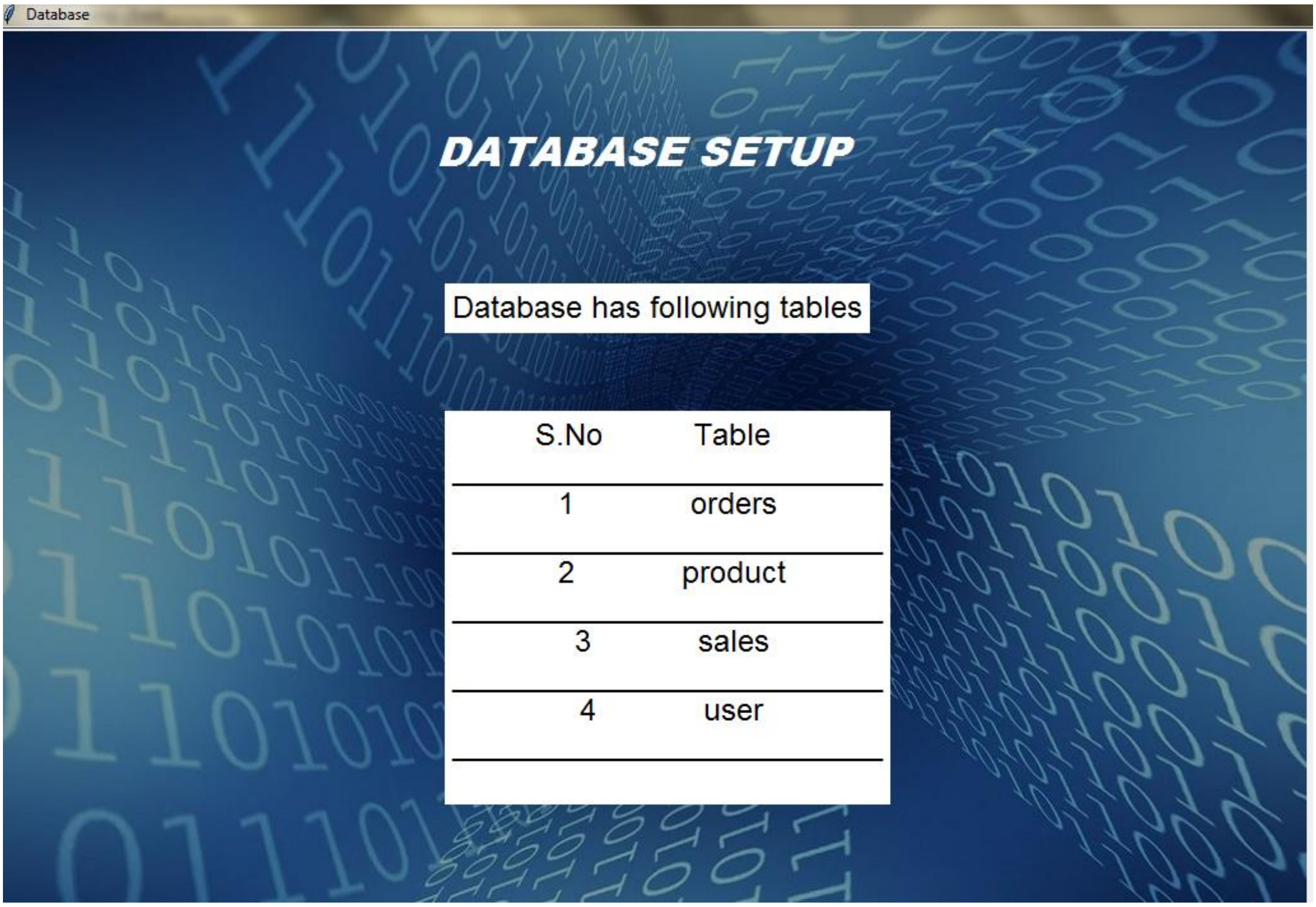
USER DETAILS ARE-

```
USER ID | NAME
--------------------------------------------------------------------------------------------------
joker@iv | RajuJoker |
--------------------------------------------------------------------------------------------------
utkarsh@iv | Utkarsh |
--------------------------------------------------------------------------------------------------
```

The Database Management Section looks like-



List Database screen



DATABASE SETUP

Database has following tables

S.No	Table
1	orders
2	product
3	sales
4	user

Thank You

BIBLIOGRAPHY

- ***www.google.com***
- ***www.stackoverflow.com***
- ***Google Images***
- ***www.quora.com***
- ***YouTube***
- ***Programbuddy.com***
- ***Computer Science with Python by Sumita Arora***
- ***Kips Learning Python***
- ***Python Documentation***