**About Project**:

Our project aims in booking cinema tickets using python-mysql connectivity.

**MODULES USED IN OUR PROJECT ARE** :

mysql.connector

stdiomask

os

pretty table

* Menu

1.Management

2.Customer

3.Exit'

* **In Management**

menu

1.Display shows

2.Insert shows

3.Delete a show

4.Insert snack

5.Show customer snacks data

6.Add amount snack

7.Reduce amount snack

8.Delete snack

9.show customer details

10.snacks in theatre

11.Exit

* **In Customer**

Menu

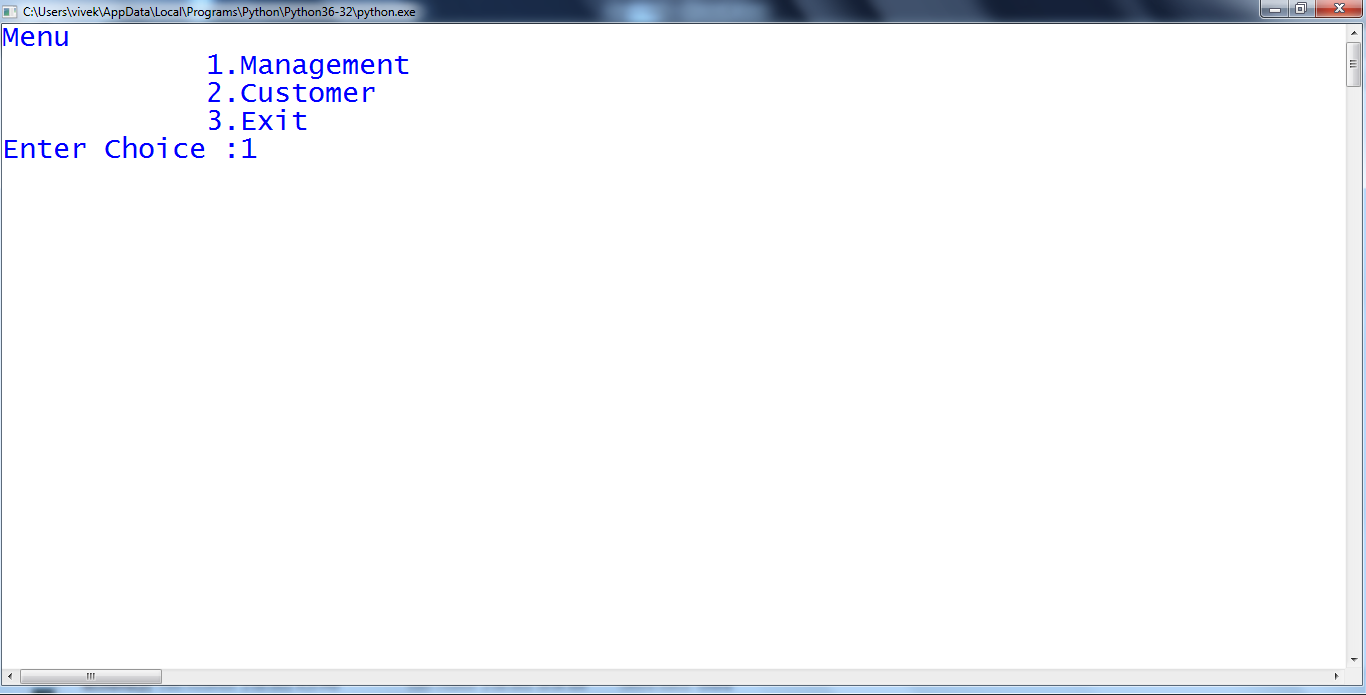
1.Book ticket

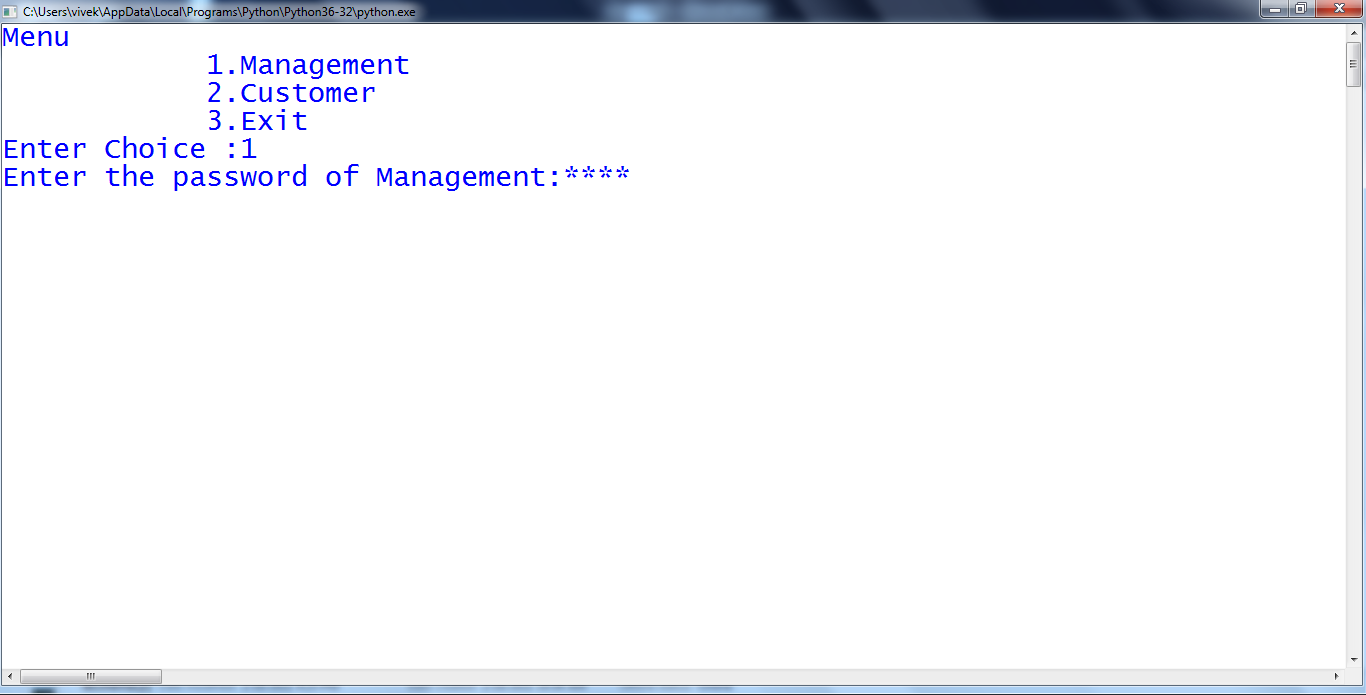
2.Book snack

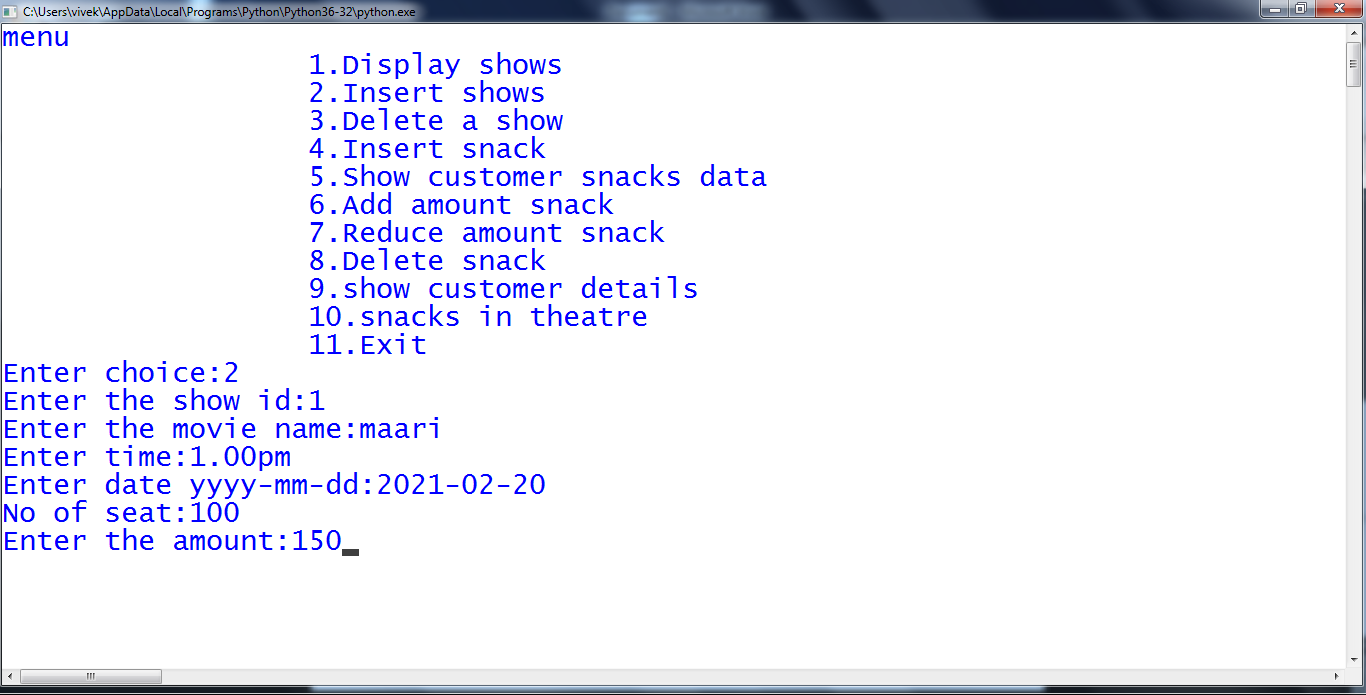
3.Exit.

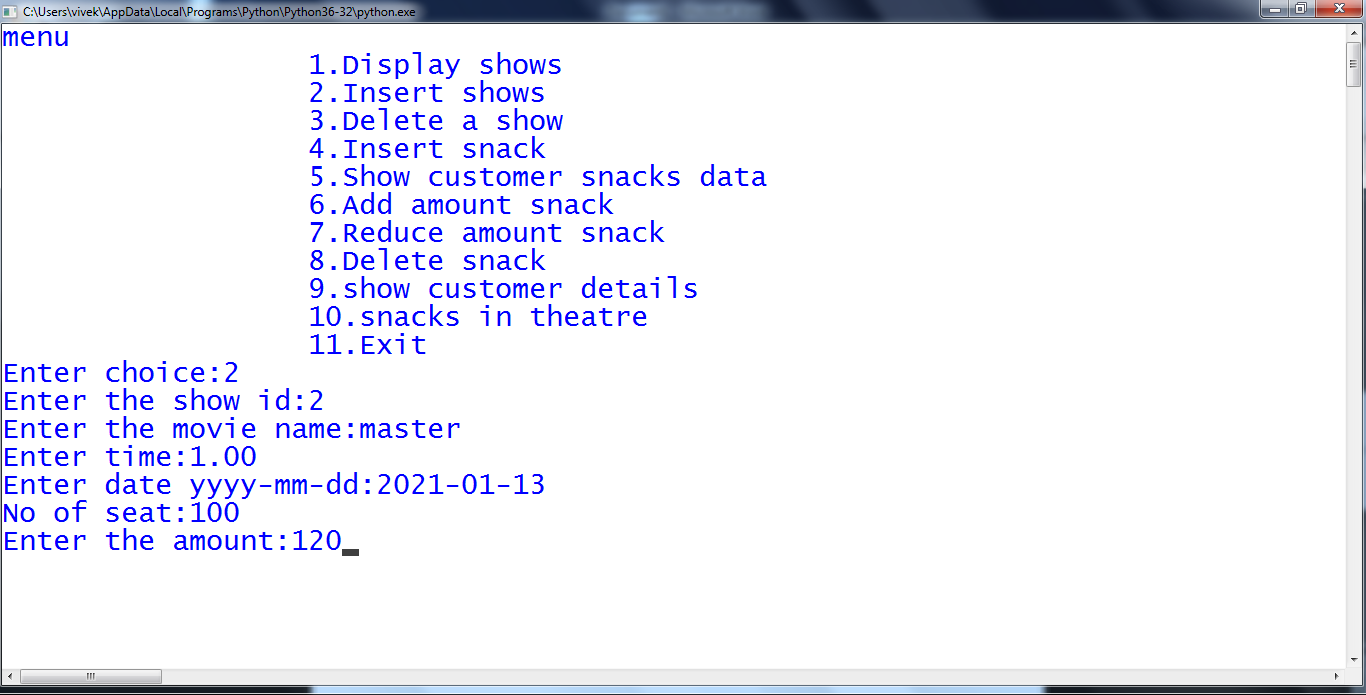
* **Using this program a person can book Movie Tickets online by a safe and secure medium.**

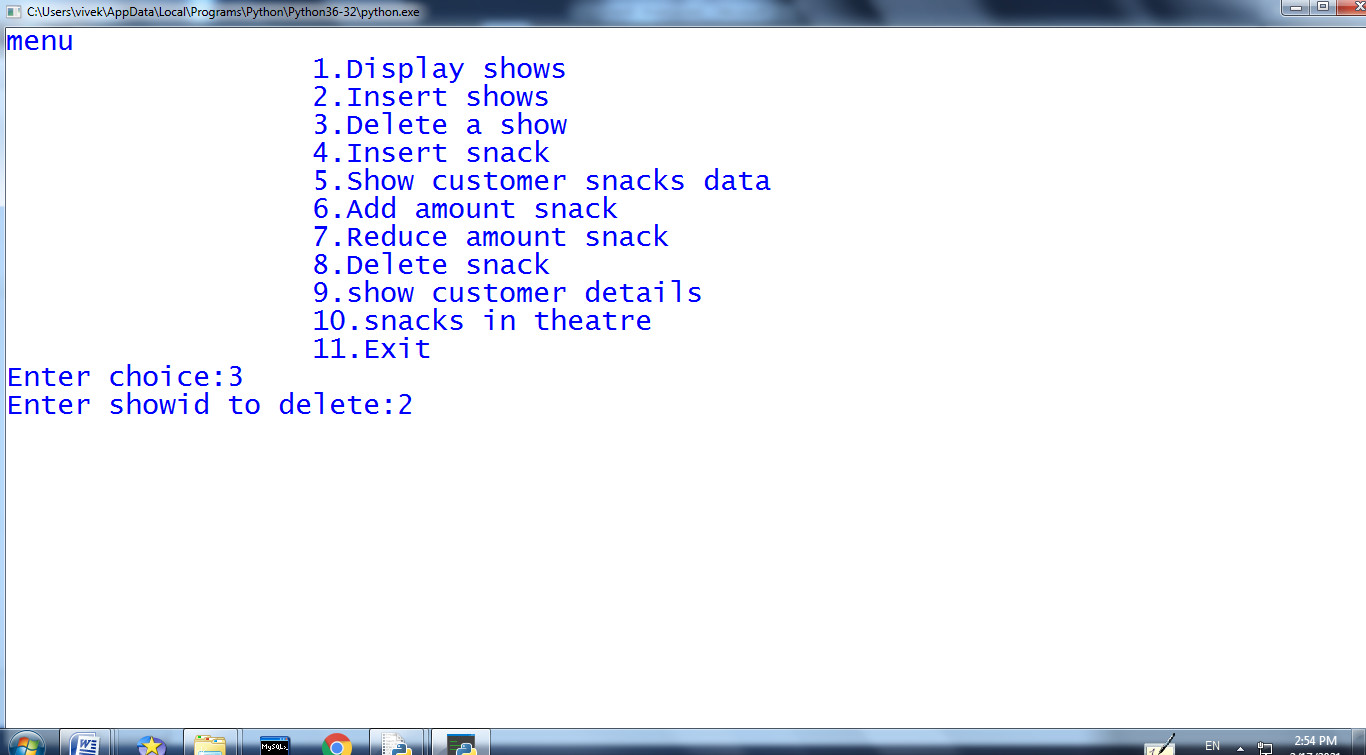
OUTPUT:

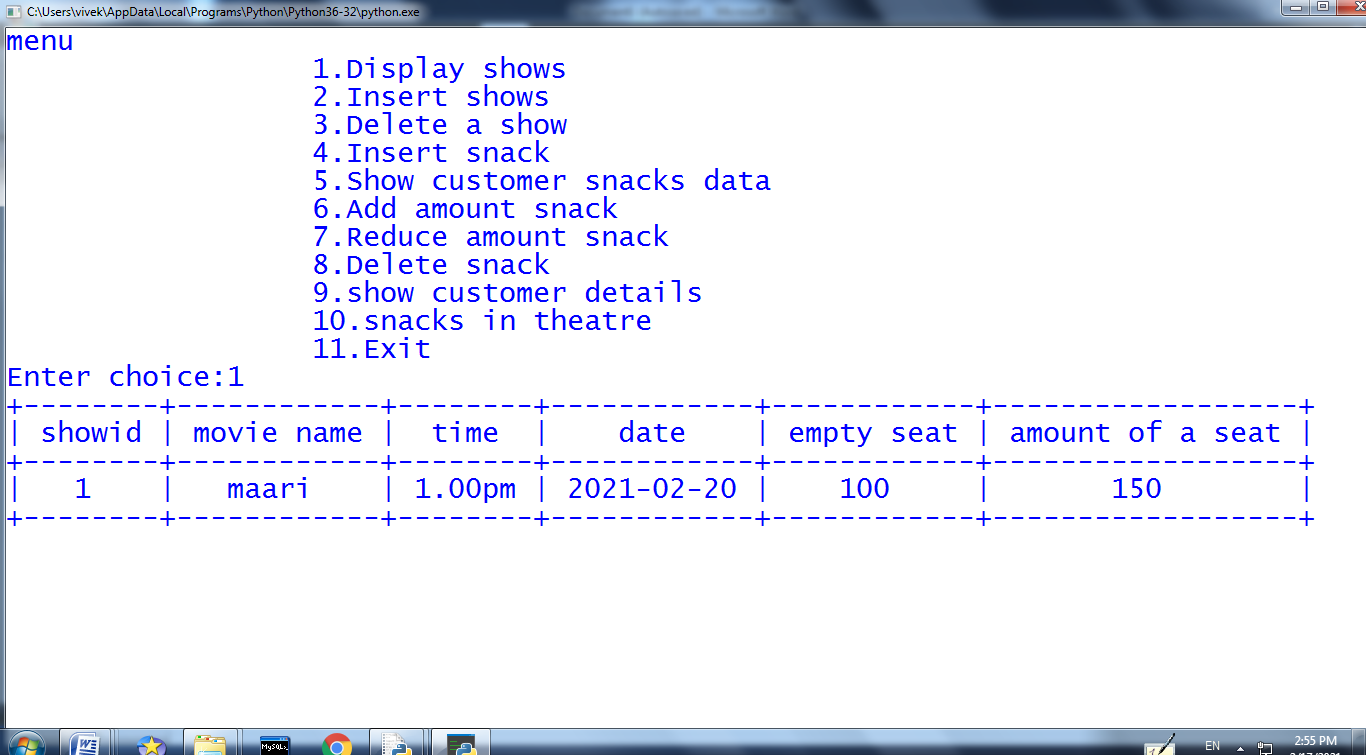


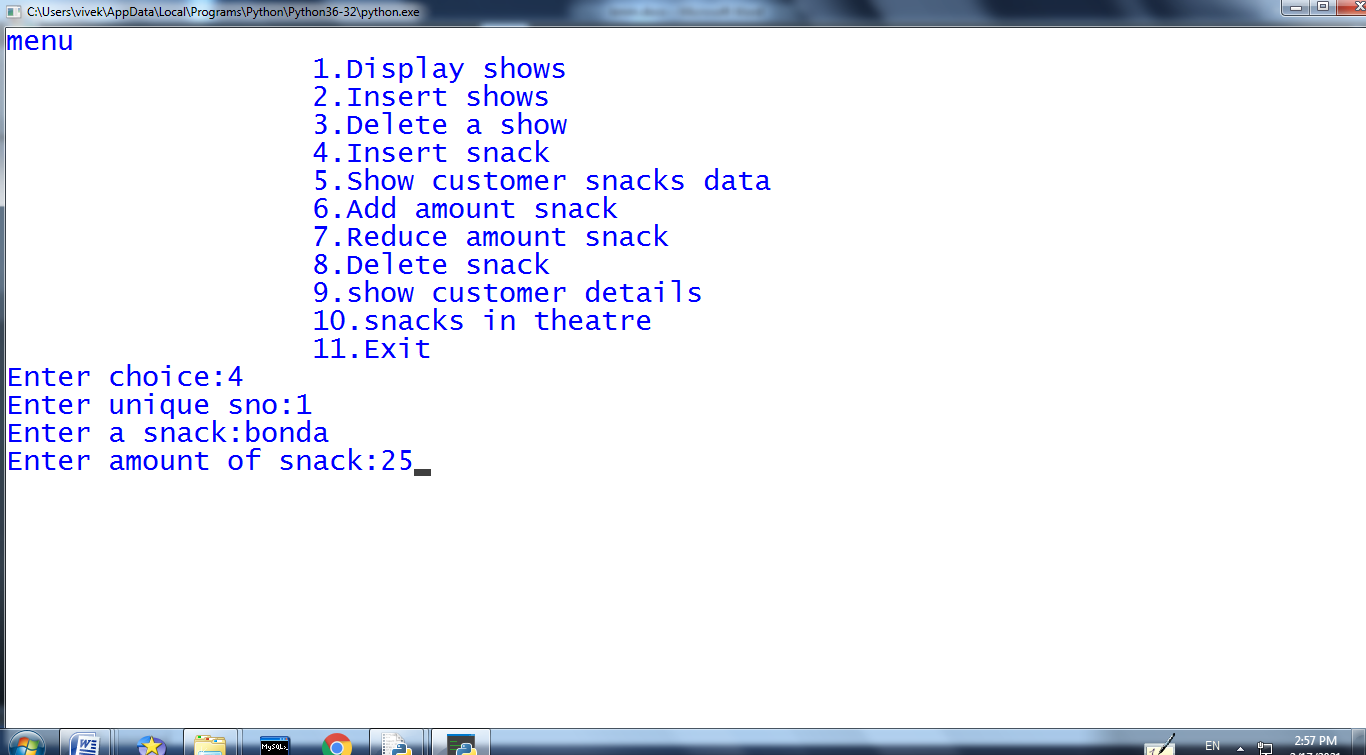


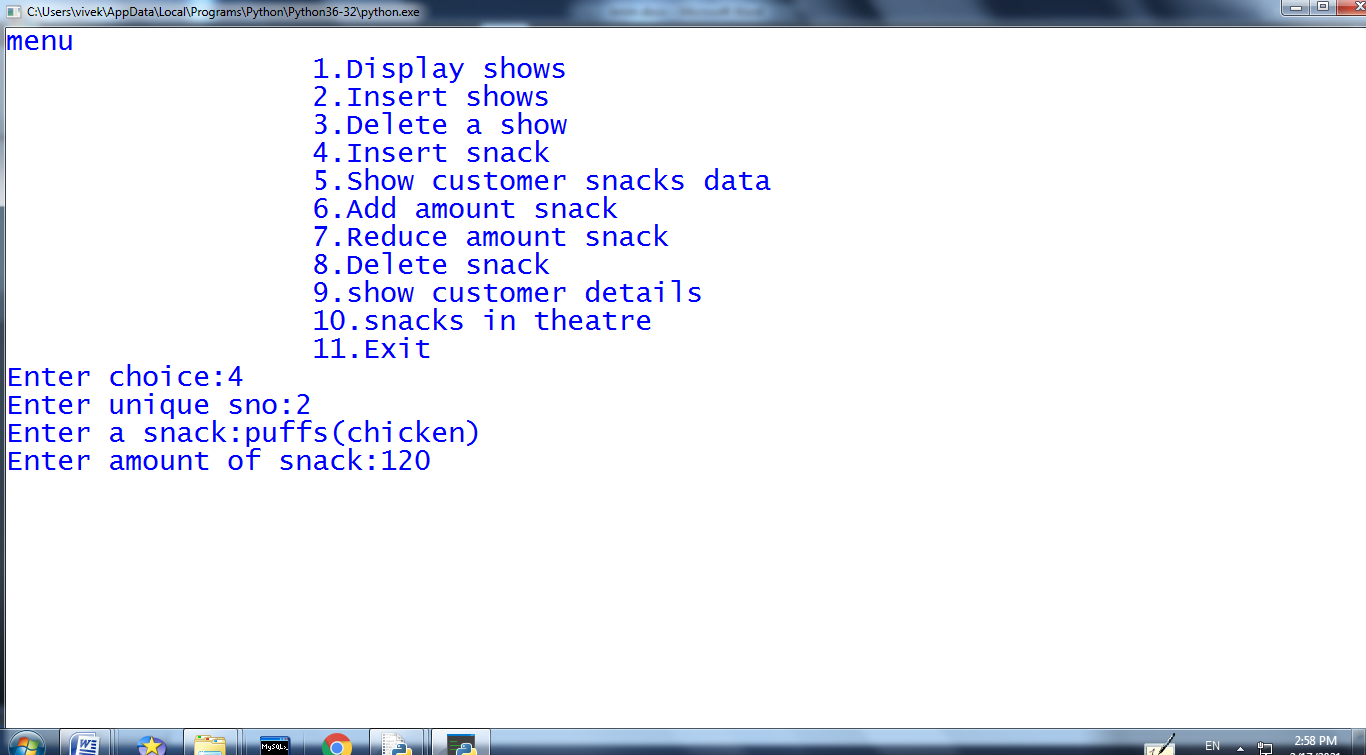


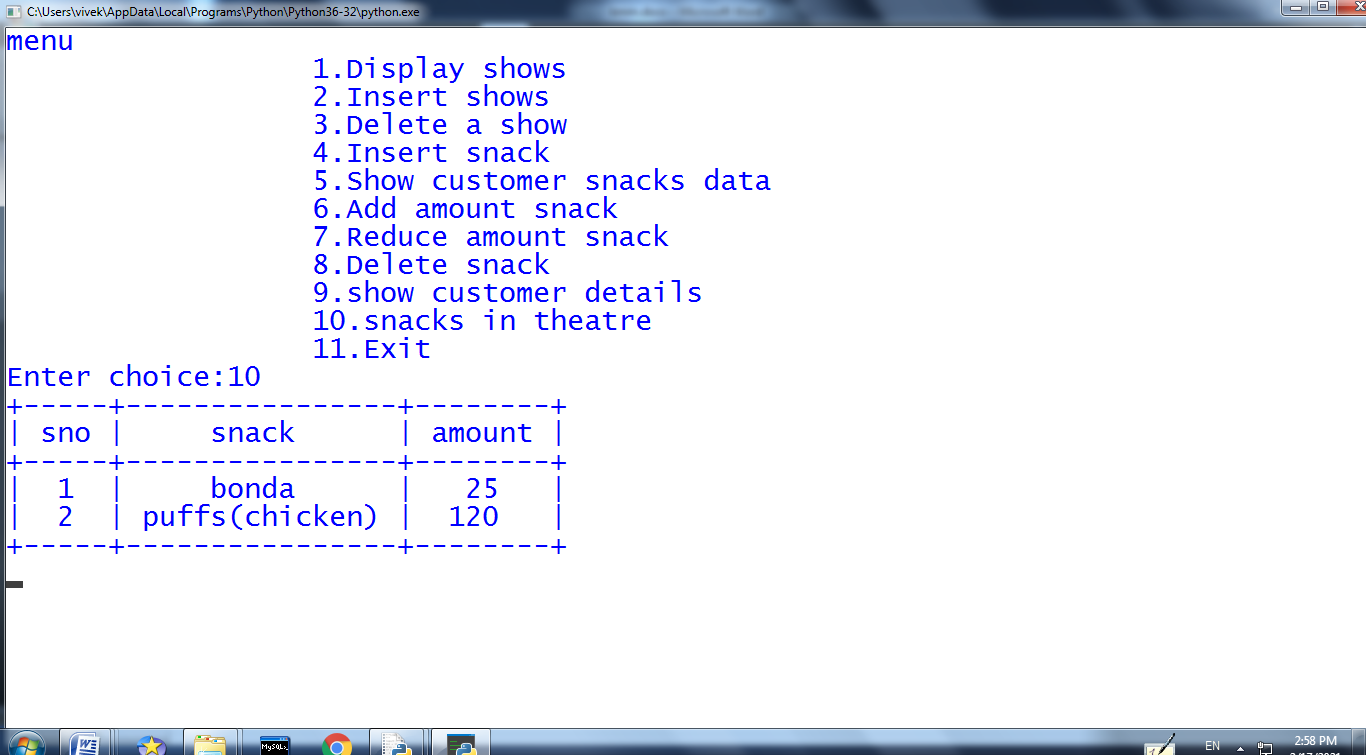


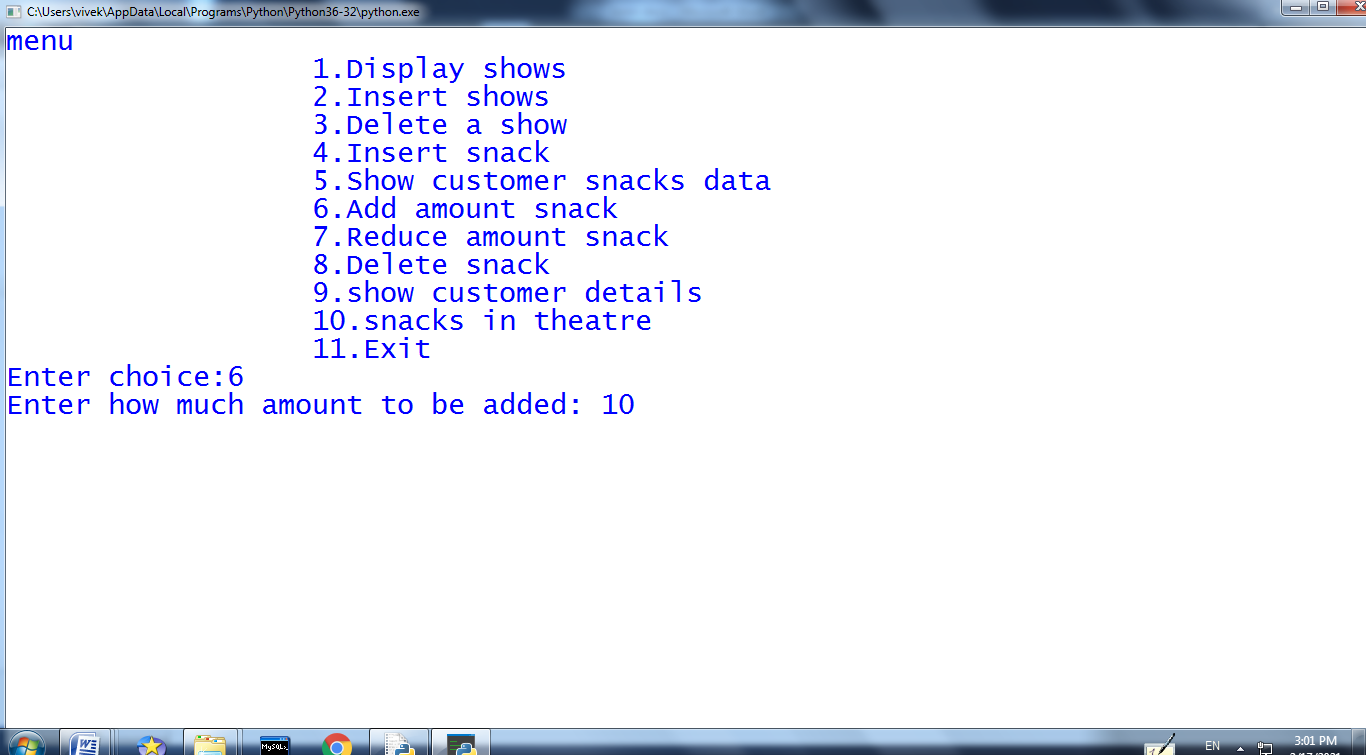


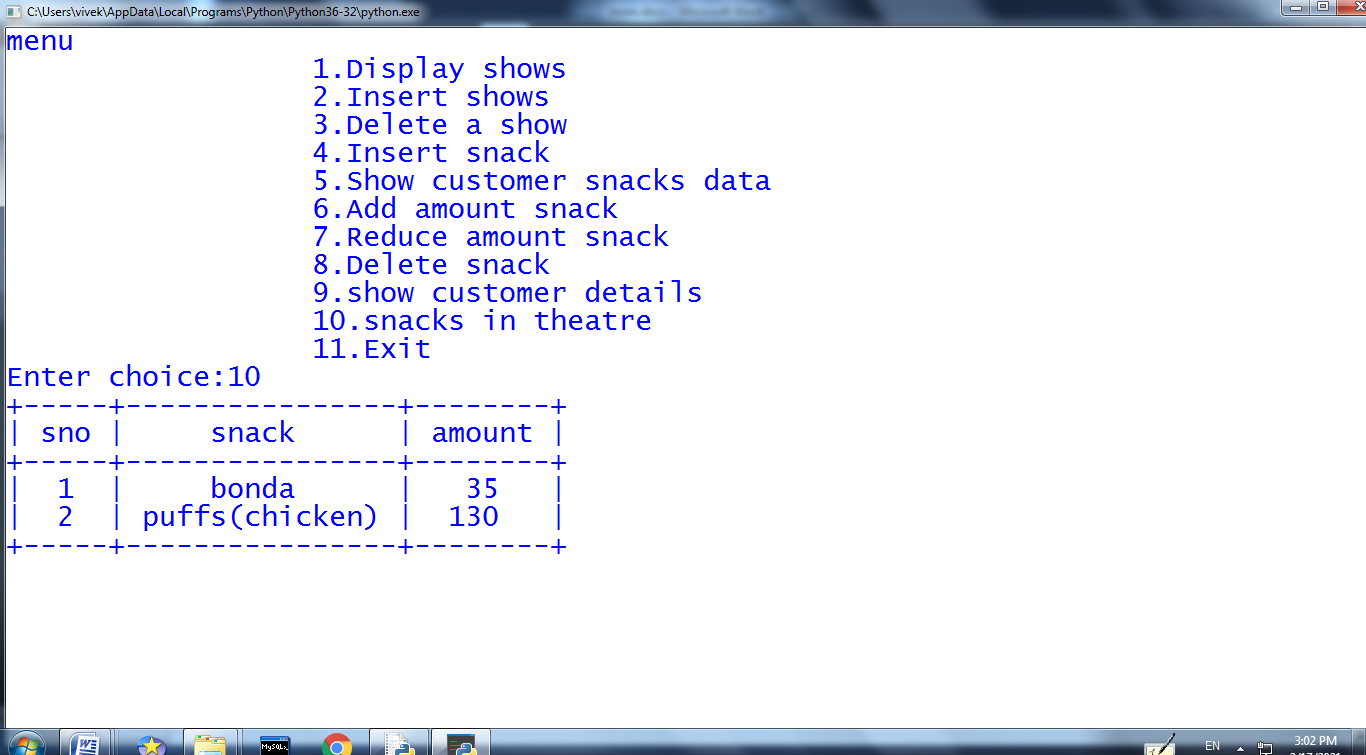


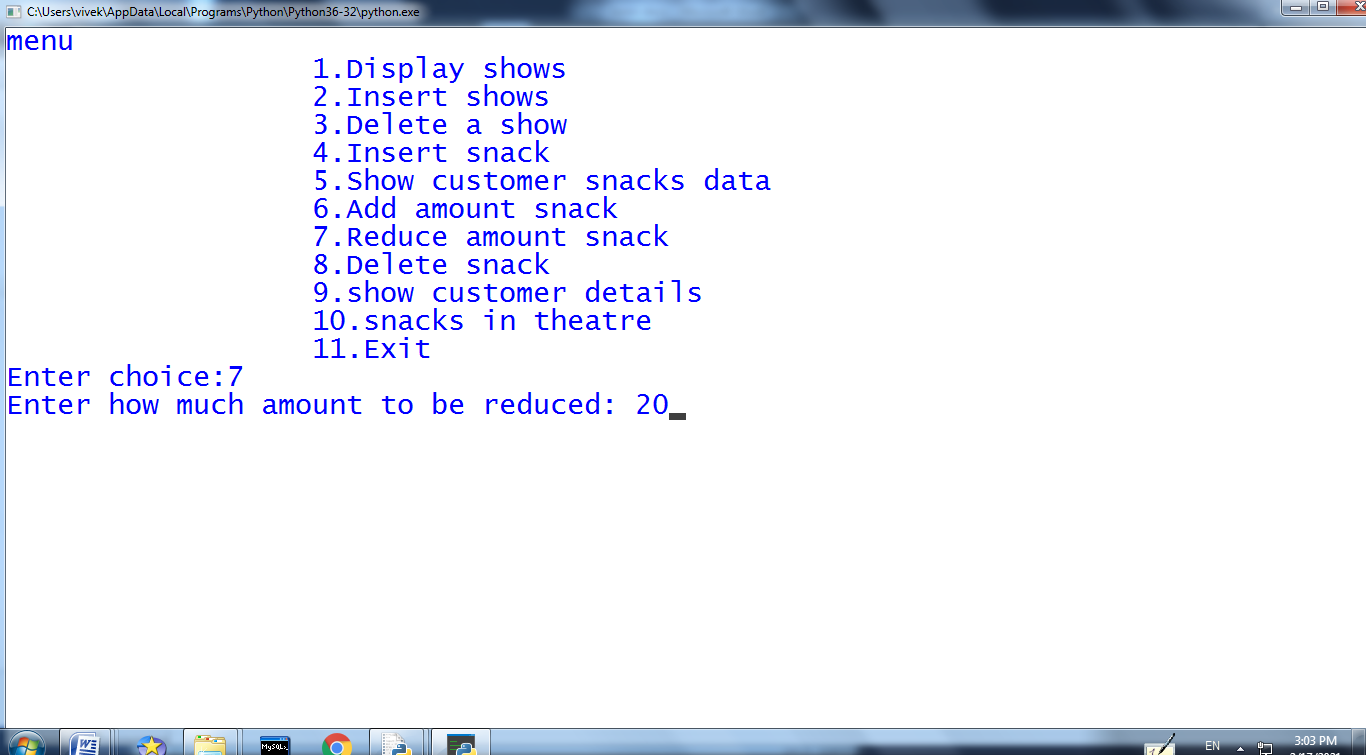


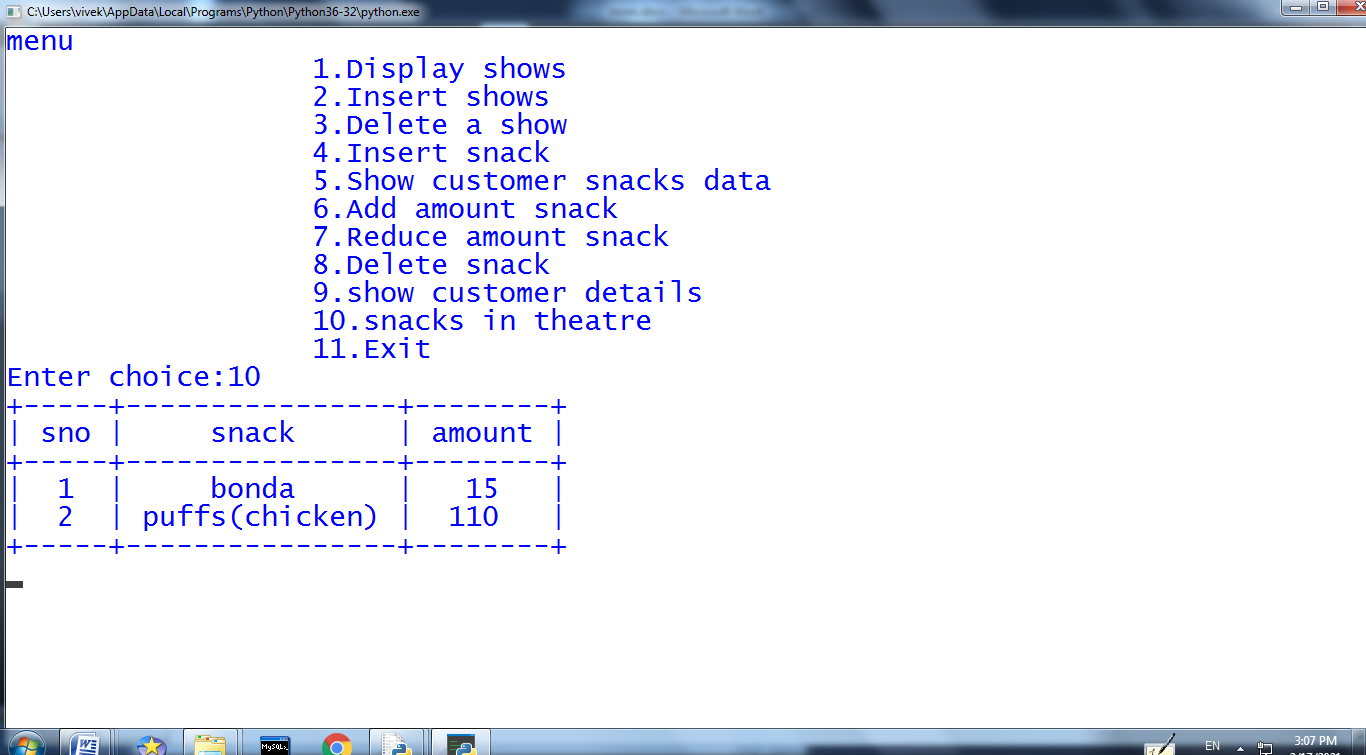


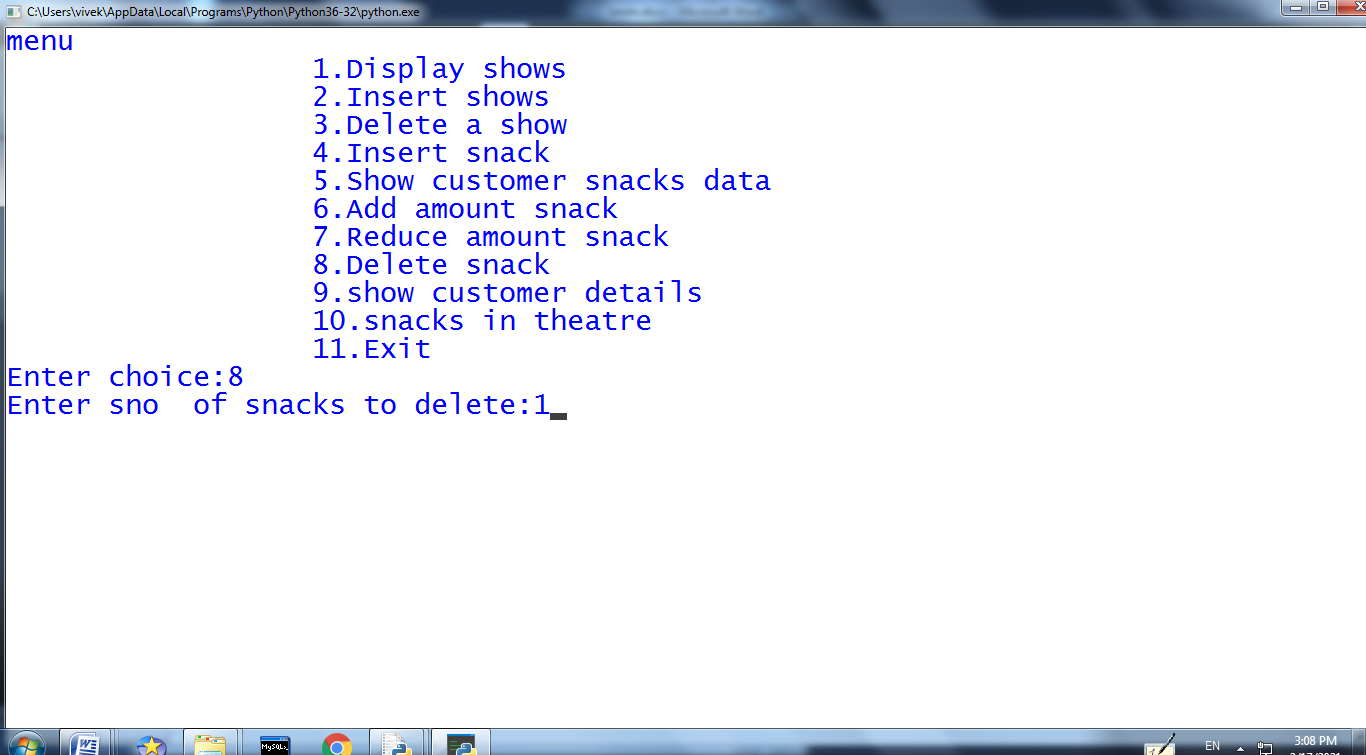


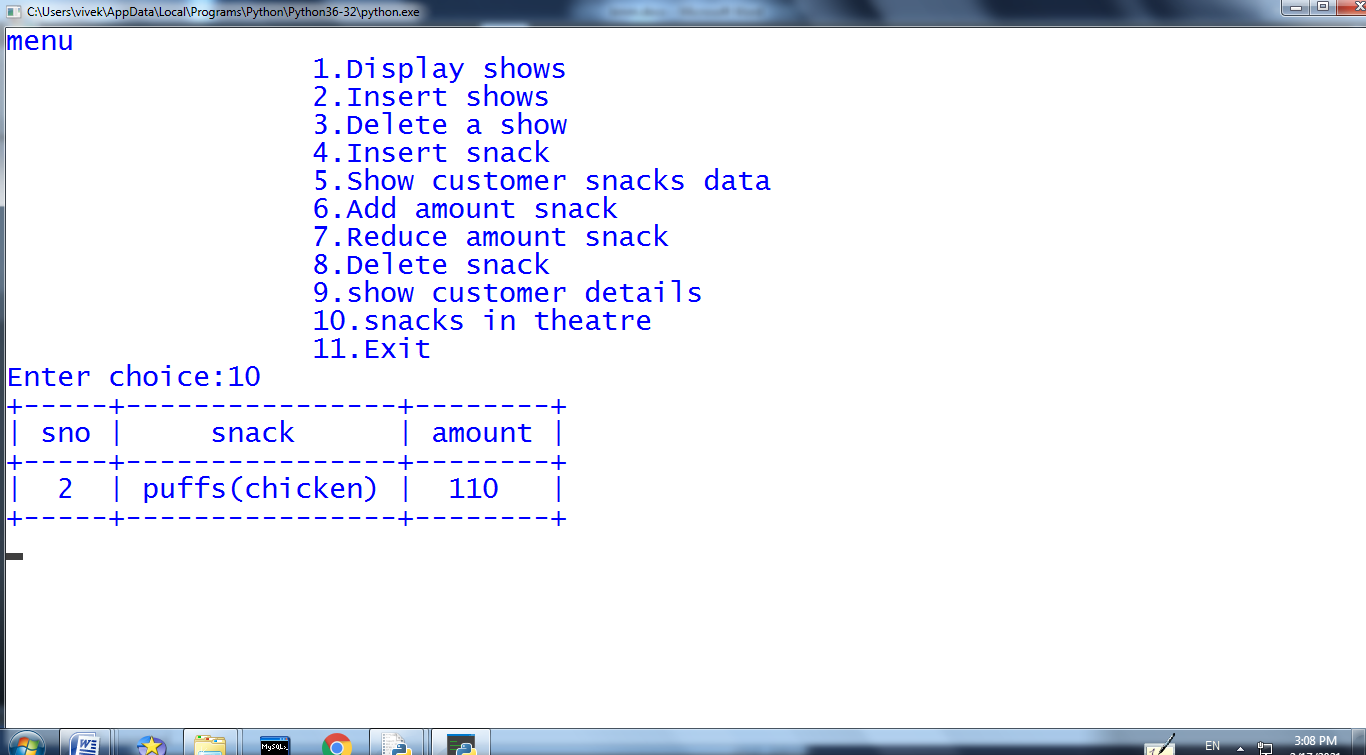


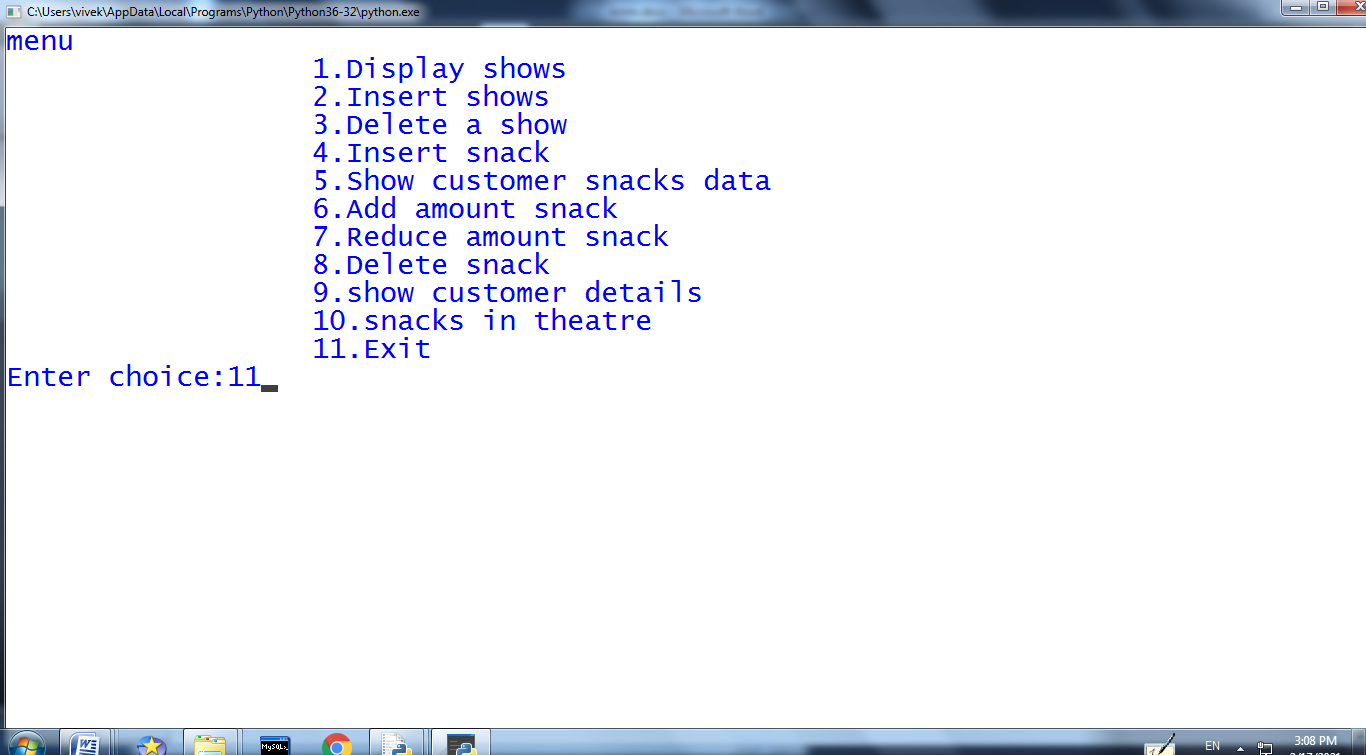


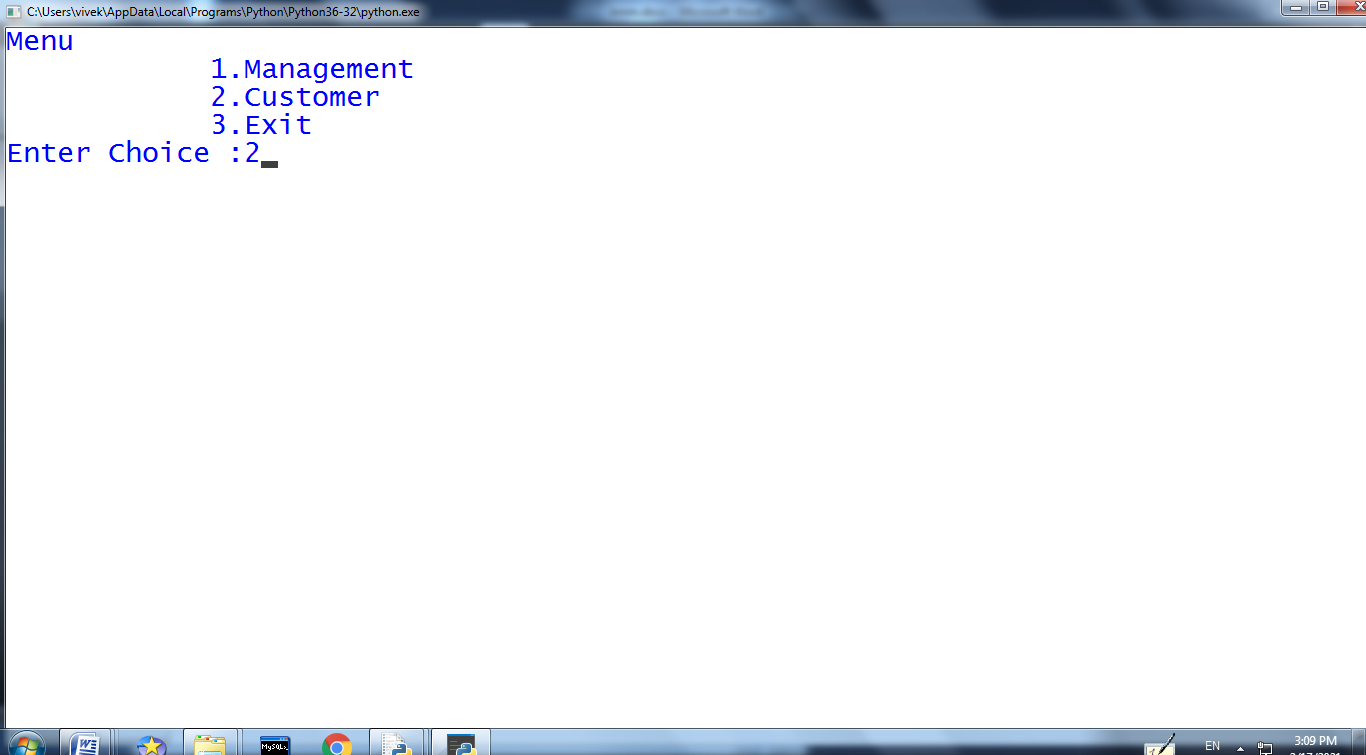


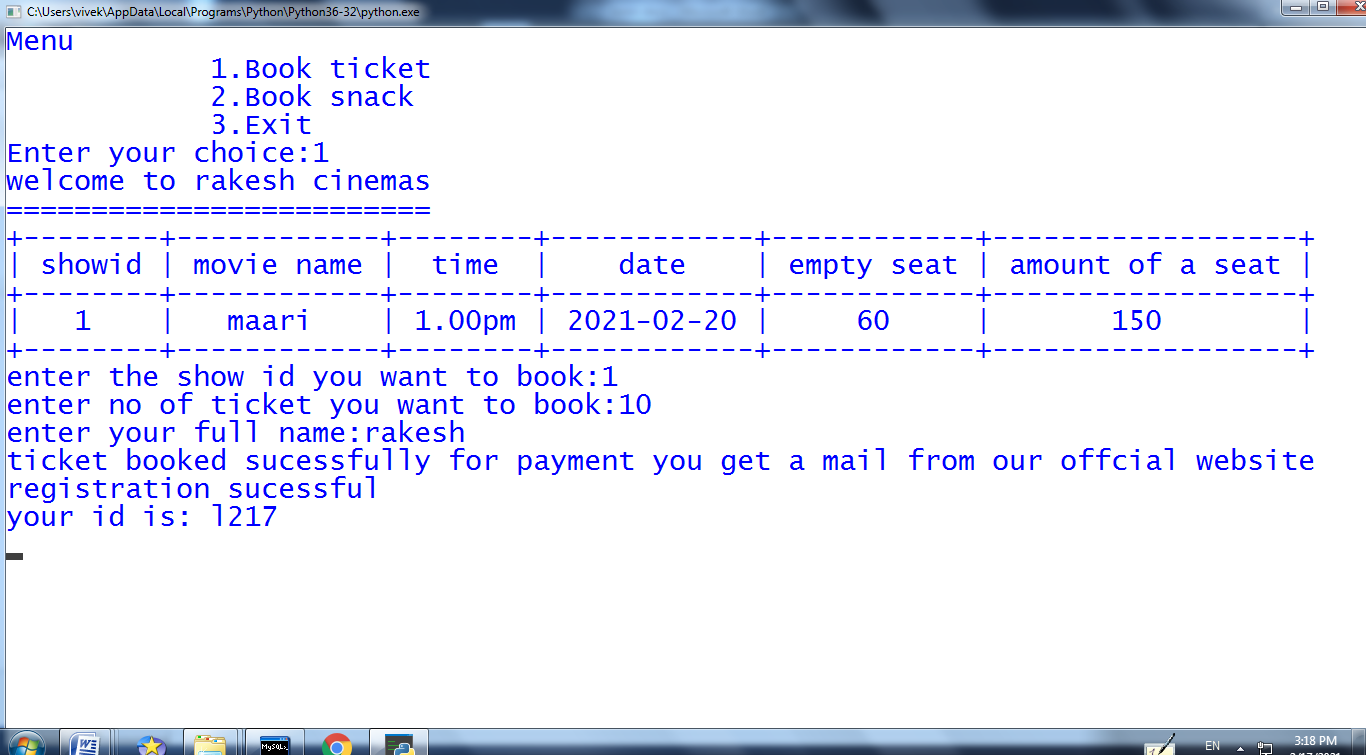


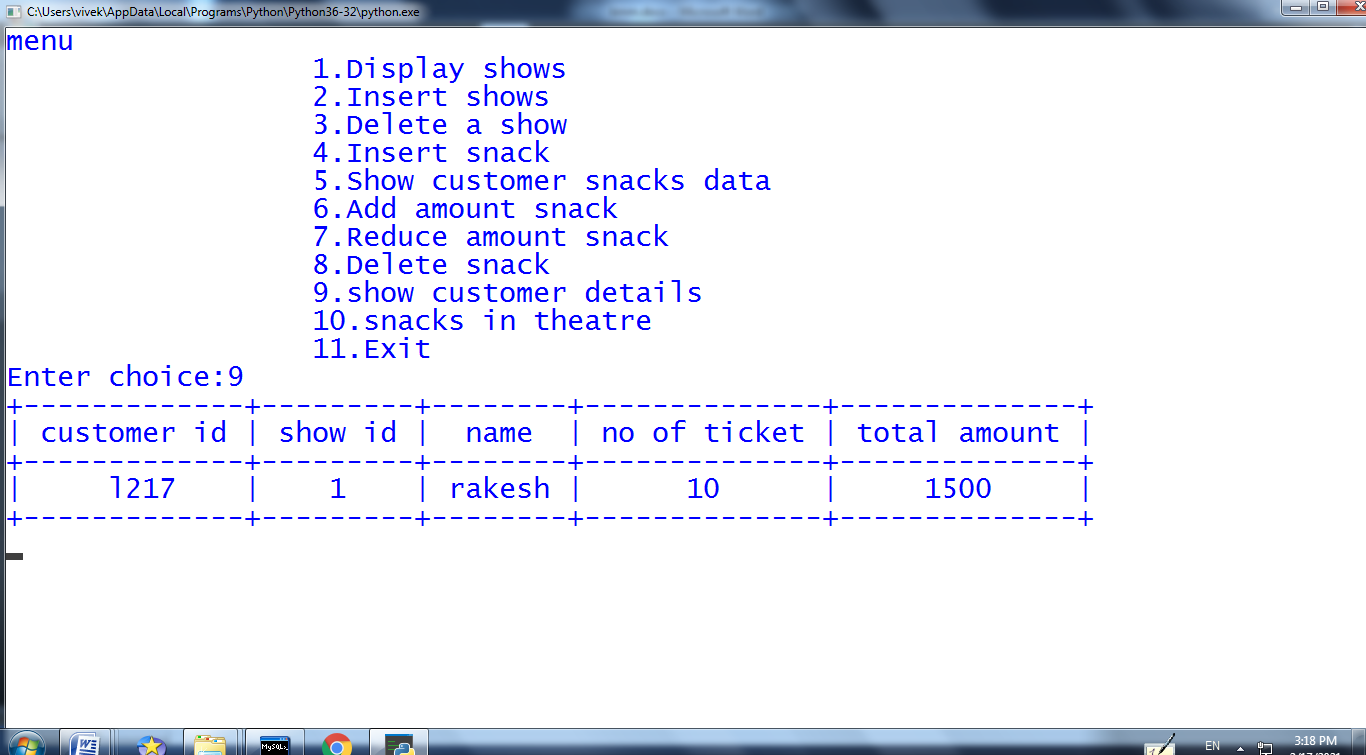


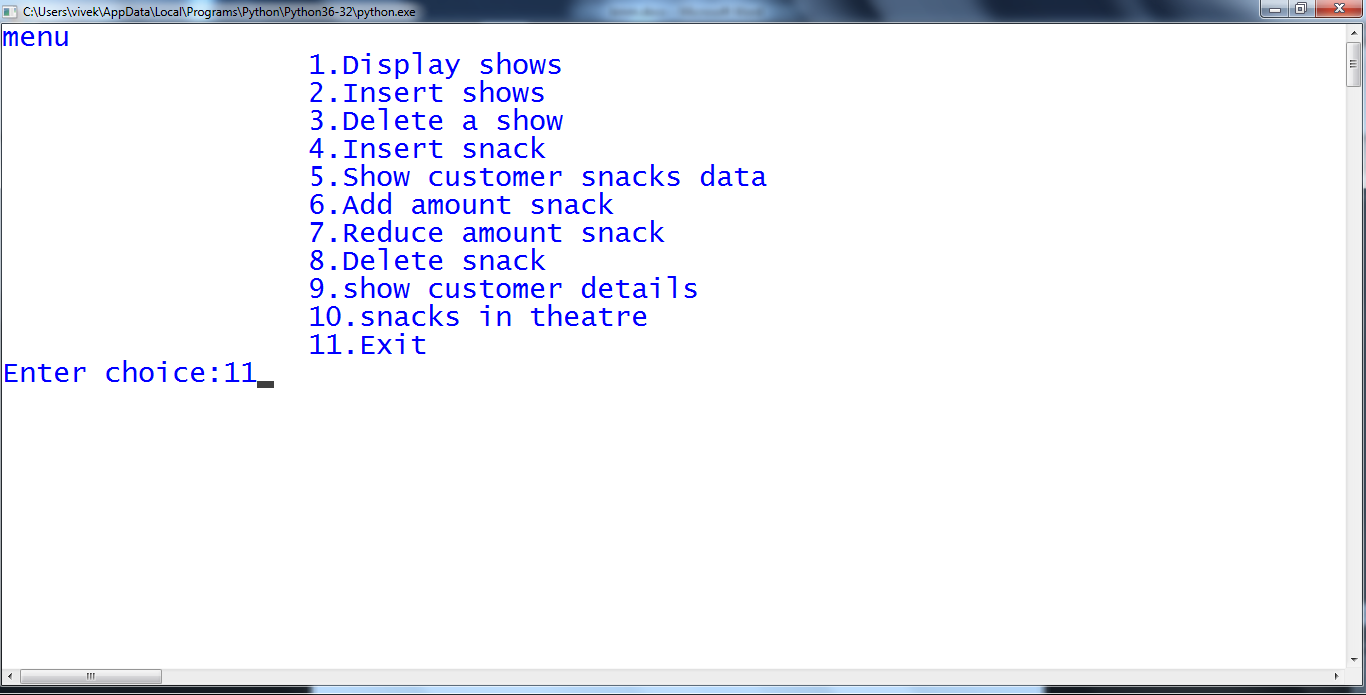


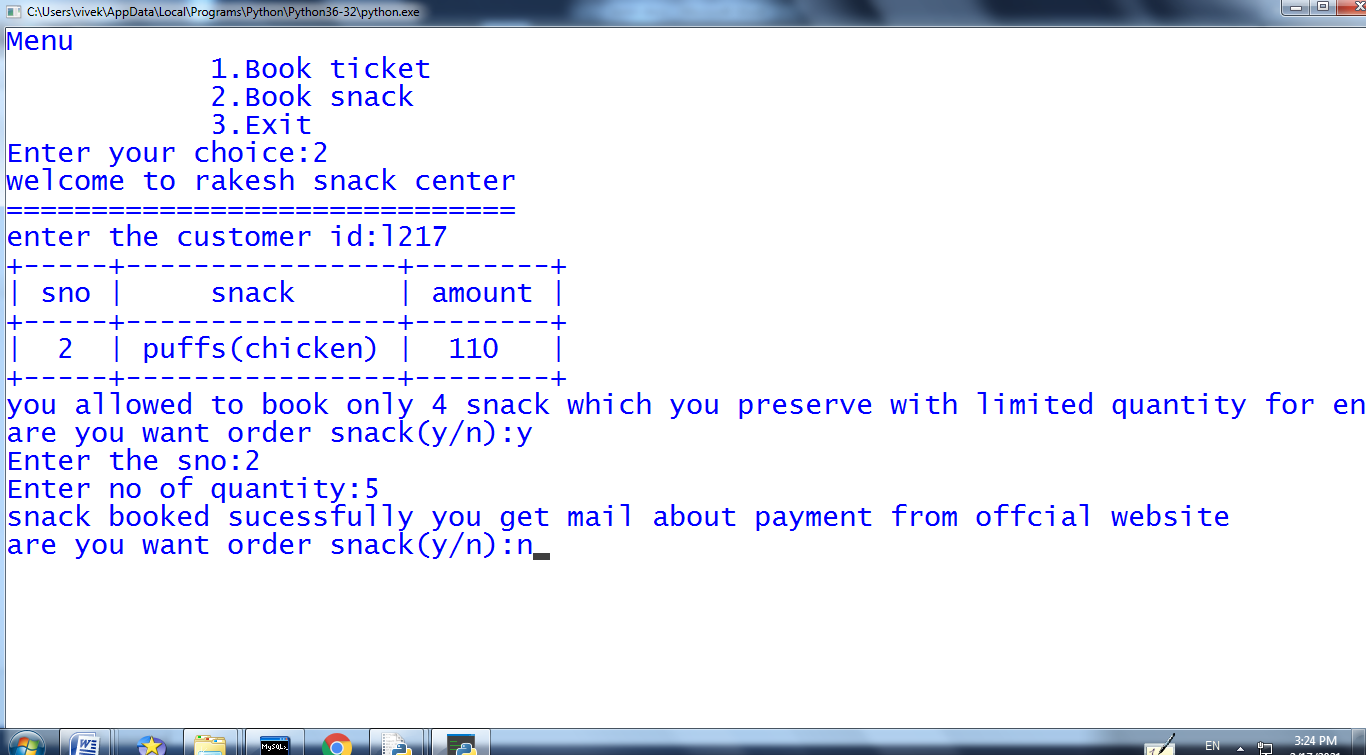


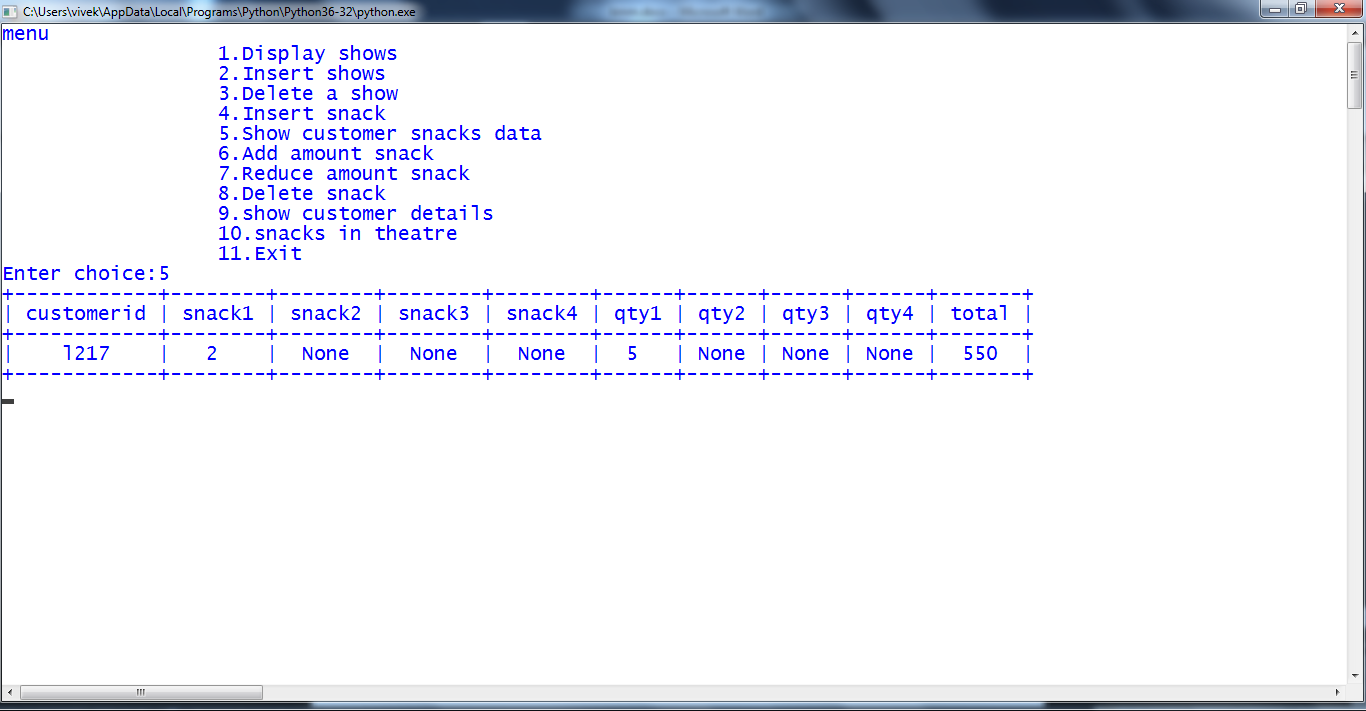


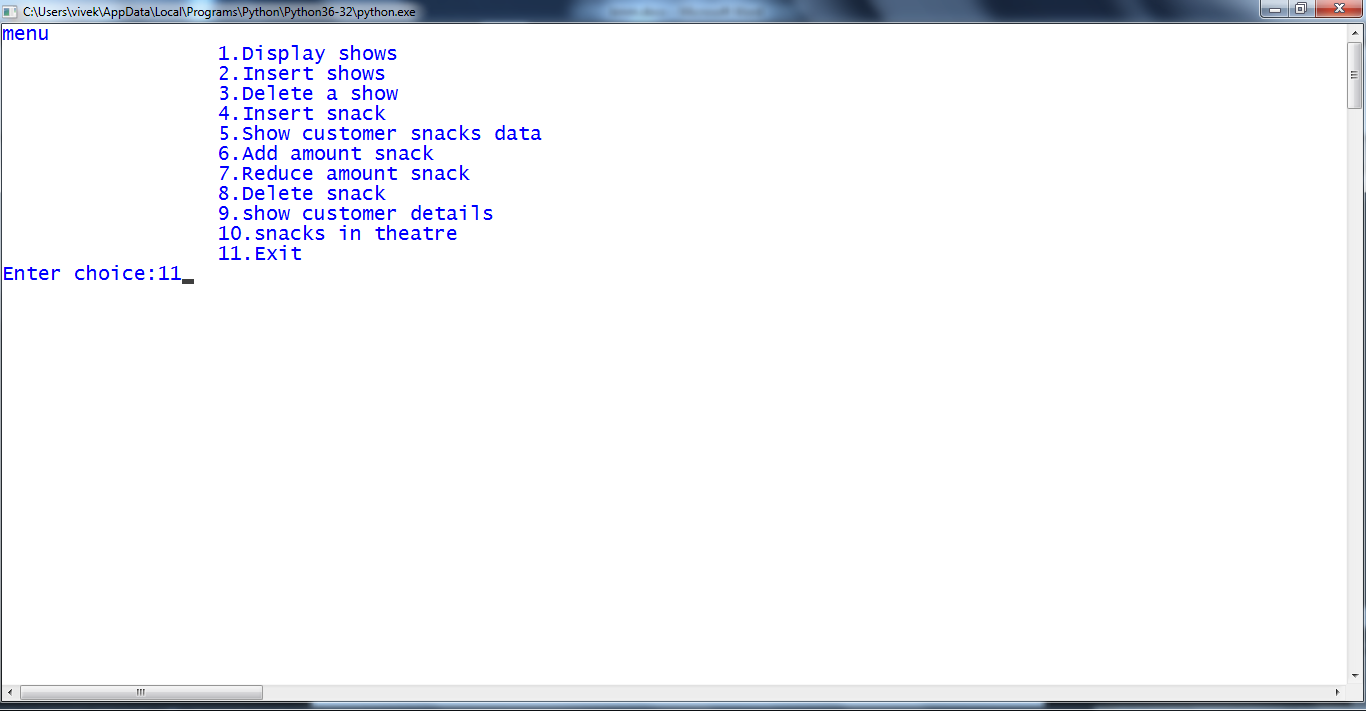


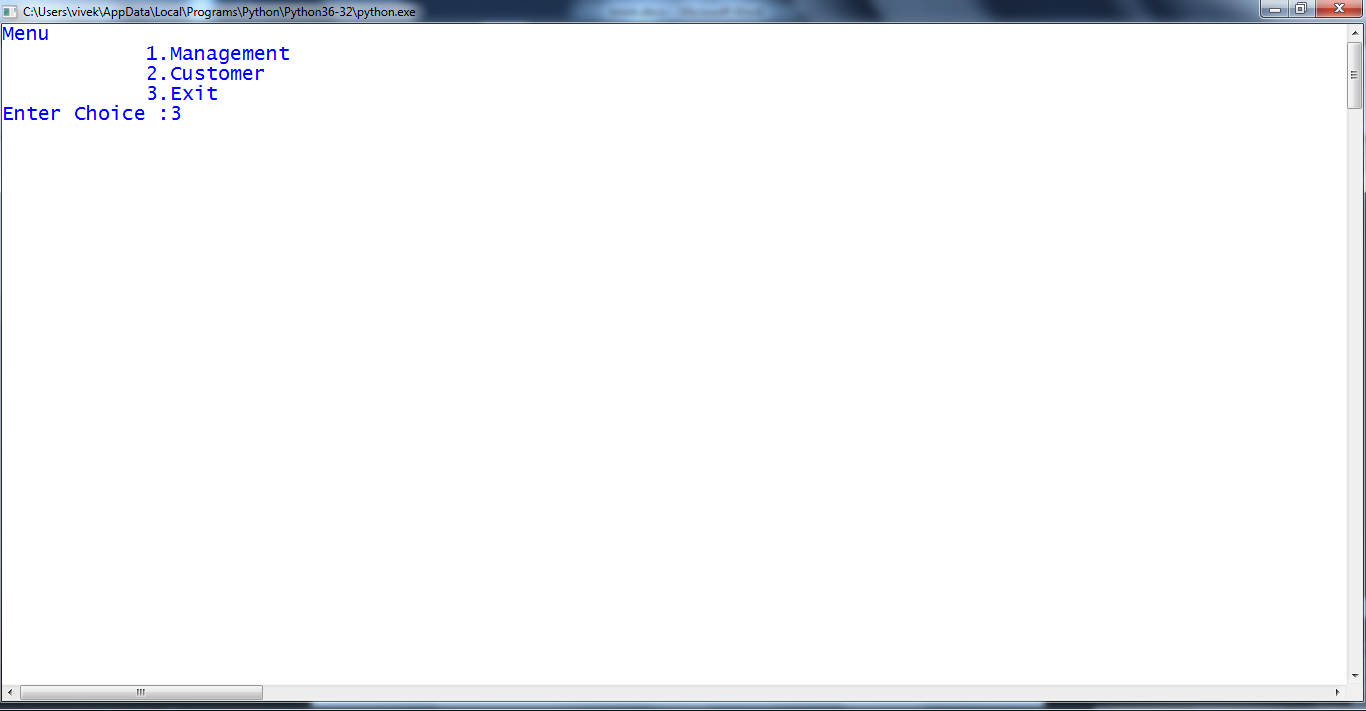












**PROGRAM**

import mysql.connector as sqltor

import random

import stdiomask

from prettytable import PrettyTable

import os

clear= lambda : os.system('cls')

db=sqltor.connect(host='localhost',user='root',passwd='vivek',database='cinema')

cursor=db.cursor()

cursor.execute('''CREATE TABLE IF NOT EXISTS MANAGEMENT

(showid int(5) primary key,

moviename varchar(20),

time varchar(20),

dateofshow date,

emptyseat int(10),

amountofseat int(5))''')

cursor.execute('''CREATE TABLE IF NOT EXISTS CUSTOMER

(customerid varchar(10) primary key,

showid int(5),

name varchar(20),

noofticket int(10),

totalamount int)''')

cursor.execute('''CREATE TABLE IF NOT EXISTS SNACKS

(customerid varchar(10) primary key,

snack1 varchar(20) default null,

snack2 varchar(20) default null,

snack3 varchar(20) default null,

snack4 varchar(20) default null,

qty1 int(5) default null,

qty2 int(5) default null,

qty3 int(5) default null,

qty4 int(5) default null,

totalamount int(20))''')

cursor.execute('''CREATE TABLE IF NOT EXISTS SNACKDETAILS

(sno int(5) primary key,

snack varchar(20),

amount int(10))''')

def management():

res=stdiomask.getpass("Enter the password of Management:")

if res=='1000':

def display():

cursor.execute('''select\*from MANAGEMENT''')

row=cursor.fetchall()

showtable=PrettyTable(['showid','movie name','time','date','empty seat','amount of a seat'])

for r in row:

showtable.add\_row([r[0],r[1],r[2],r[3],r[4],r[5]])

print(showtable)

input()

def insert():

try:

a=int(input("Enter the show id:"))

b=input("Enter the movie name:")

c=input("Enter time:")

d=input("Enter date yyyy-mm-dd:")

e=int(input("No of seat:"))

f=int(input("Enter the amount:"))

cursor.execute('''INSERT INTO MANAGEMENT VALUES(%s,%s,%s,%s,%s,%s)''',(a,b,c,d,e,f))

db.commit()

except:

print("ERROR!!!")

db.rollback()

return

db.commit()

def deleteshow():

a=int(input("Enter showid to delete:"))

cursor.execute('''delete from MANAGEMENT where showid =(%s)''',(a,))

db.commit()

def cussnack():

cursor.execute('''select\*from SNACKS''')

row=cursor.fetchall()

showtable=PrettyTable(['customerid','snack1','snack2','snack3','snack4','qty1','qty2','qty3','qty4','total'])

for r in row:

showtable.add\_row([r[0],r[1],r[2],r[3],r[4],r[5],r[6],r[7],r[8],r[9]])

print(showtable)

input()

def insertsnack():

c=int(input("Enter unique sno:"))

a=input("Enter a snack:")

b=int(input("Enter amount of snack:"))

cursor.execute('''INSERT INTO SNACKDETAILS VALUES(%s,%s,%s)''',(c,a,b))

db.commit()

def addsnackamount():

a=int(input("Enter how much amount to be added: "))

cursor.execute('''update SNACKDETAILS set amount=amount+(%s)''',(a,))

db.commit()

def subsnackamount():

a=int(input("Enter how much amount to be reduced: "))

cursor.execute('''update SNACKDETAILS set amount=amount-(%s)''',(a,))

db.commit()

def delsnack():

a=int(input("Enter sno of snacks to delete:"))

cursor.execute('''delete from SNACKDETAILS where sno = (%s)''',(a,))

db.commit()

def showcusdetails():

cursor.execute('''select\*from CUSTOMER''')

row=cursor.fetchall()

showtable=PrettyTable(['customer id','show id','name','no of ticket','total amount'])

for r in row:

showtable.add\_row([r[0],r[1],r[2],r[3],r[4]])

print(showtable)

input()

def snackintheatre():

cursor.execute('''select\*from SNACKDETAILS''')

row=cursor.fetchall()

showtable=PrettyTable(['sno','snack','amount'])

for r in row:

showtable.add\_row([r[0],r[1],r[2]])

print(showtable)

input()

while(1):

clear()

print('''menu

1.Display shows

2.Insert shows

3.Delete a show

4.Insert snack

5.Show customer snacks data

6.Add amount snack

7.Reduce amount snack

8.Delete snack

9.show customer details

10.snacks in theatre

11.Exit''')

ch=int(input("Enter choice:"))

if ch==1:

display()

elif ch==2:

insert()

elif ch==3:

deleteshow()

elif ch==4:

insertsnack()

elif ch==5:

cussnack()

elif ch==6:

addsnackamount()

elif ch==7:

subsnackamount()

elif ch==8:

delsnack()

elif ch==9:

showcusdetails()

elif ch==10:

snackintheatre()

elif ch==11:

break

else:

print("Invalid input!!")

else:

print("Wrong password!!")

u=0

v=0

def customer():

def customerdetail():

print("welcome to rakesh cinemas")

print("=========================")

cursor.execute('''select\*from MANAGEMENT''')

row=cursor.fetchall()

showtable=PrettyTable(['showid','movie name','time','date','empty seat','amount of a seat'])

for r in row:

showtable.add\_row([r[0],r[1],r[2],r[3],r[4],r[5]])

print(showtable)

global u

global v

global x

u=int(input("enter the show id you want to book:"))

v=int(input("enter no of ticket you want to book:"))

x=input("enter your full name:")

def customerid():

cursor.execute('''select\*from CUSTOMER''')

row=cursor.fetchall()

a='l'+str(random.randint(100,999))

while (a,) in row:

a='l'+str(random.randint(100,999))

print("registration sucessful\nyour id is:",a)

return a

def ticketbook():

global u

global v

global x

total=0

cursor.execute('''SELECT\*FROM MANAGEMENT WHERE showid = (%s)''',(u,))

a=cursor.fetchall()

if a[0][4]>=v:

cursor.execute('''UPDATE MANAGEMENT SET emptyseat = emptyseat-(%s)''',(v,))

print("ticket booked sucessfully for payment you get a mail from our offcial website")

total=total+(v\*a[0][5])

l=customerid()

cursor.execute('''insert into CUSTOMER VALUES(%s,%s,%s,%s,%s)''',(l,u,x,v,total))

db.commit()

elif a[0][4]==0:

print(" sorry seat is fully booked try for other shows")

else:

print("that much of seat not available try for another shows")

ticketbook()

input()

def snacks():

print("welcome to rakesh snack center")

print("==============================")

l=[]

l1=[]

l2=[]

l3=[]

amount=0

b=input("enter the customer id:")

cursor.execute('''select\*from CUSTOMER''')

row=cursor.fetchall()

for r in row:

l.append(r[0])

if b in l:

a=0

cursor.execute('''select\*from SNACKDETAILS''')

row=cursor.fetchall()

showtable=PrettyTable(['sno','snack','amount'])

for r in row:

showtable.add\_row([r[0],r[1],r[2]])

l3.append(r[2])

print(showtable)

print("you allowed to book only 4 snack which you preserve with limited quantity for environment safety ")

for i in range(4):

d=input("are you want order snack(y/n):")

if d=='y':

a=int(input("Enter the sno:"))

c=int(input("Enter no of quantity:"))

l1.append(a)

l2.append(c)

print("snack booked sucessfully you get mail about payment from offcial website")

else:

break

amount=amount

if len(l1)==1:

cursor.execute('''INSERT INTO SNACKS(customerid,snack1,qty1,totalamount)VALUES('{}','{}','{}','{}')'''.format(b,l1[0],l2[0],(l3[0]\*l2[0])))

elif len(l1)==2:

cursor.execute('''INSERT INTO SNACKS(customerid,snack1,snack2,qty1,qty2,totalamount)VALUES('{}','{}','{}','{}','{}','{}')'''.format(b,l1[0],l1[1],l2[0],l2[1],((l3[0]\*l2[0])+(l3[1]\*l2[1]))))

elif len(l1)==3:

cursor.execute('''INSERT INTO SNACKS(customerid,snack1,snack2,snack3,qty1,qty2,qty3,totalamount)VALUES('{}','{}','{}','{}','{}','{}','{}','{}')'''.format(b,l1[0],l1[1],l[2],l2[0],l2[1],l[2],((l3[0]\*l2[0])+(l3[1]\*l2[1])+(l3[2]\*l2[2]))))

elif len(l1)==4:

cursor.execute('''INSERT INTO SNACKS(customerid,snack1,snack2,snck3,snack4,qty1,qty2,qty3,qty4,totalamount)VALUES('{}','{}','{}','{}','{}','{}','{}','{}','{}','{}')'''.format(b,l1[0],l1[1],l1[2],l[3],l2[0],l2[1],l2[2],l[3],((l3[0]\*l2[0])+(l3[1]\*l2[1])+(l3[2]\*l2[2])+(l3[3]\*l2[3]))))

while 1:

clear()

print('''Menu

1.Book ticket

2.Book snack

3.Exit''')

ch=int(input("Enter your choice:"))

if ch==1:

customerdetail()

elif ch==2:

snacks()

else:

break

while(1):

clear()

print('''Menu

1.Management

2.Customer

3.Exit''')

ch=int(input("Enter Choice :"))

if ch==1:

management()

elif ch==2:

customer()

elif ch==3:

break

else:

print("Invalid Input")

db.close()