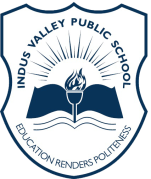
INDUS VALLEY PUBLIC SCHOOL, NOIDA



PROJECT FILE

**SUBJECT:** COMPUTER SCIENCE

**TOPIC:** BANK MANAGEMENT SYSTEM

**SESSION:** 2020-21

**Submitted to:** Ms. Deepali Agarwal

**Name:** Rohan Kumar Singh

**Class:** XII-Science B

Acknowledgement

I would like to express my special thanks of gratitude to my teacher Ms. Deepali Agarwal as well as our principal Ms. Shikha Sharma who gave me the golden opportunity to do this wonderful project on the topic Bank Management System, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them. Secondly, I would also like to thank my parents and friends who helped me a lot in finishing this project within the limited time. I am making this project not only for marks but to also increase my knowledge. Thanks again to all who helped me.

Certificate

This is to certify that Rohan Kumar Singh of class XII Science B of Indus Valley Public School, Noida has completed his practical file under my supervision. He has taken proper care and has shown utmost sincerity in completion of these practicals.

I certify that these practicals are up to my expectations and as per the guidelines issued by C.B.S.E.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Teacher’s Signature Principal’s Signature

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner’s Signature

Code

import time

import tabulate

import mysql.connector

import datetime

import random

def loading(line):

print('',end="")

print()

print(line,end="")

for j in range(4):

time.sleep(1)

print('.', end='')

print()

print()

time.sleep(0.5)

def again(input\_field,count):

if count<2:

input\_field=input(" You have entered incorrect value, please re-enter >> ")

else:

input\_field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

return input\_field

def check\_empty(field,typee):

if typee==4:

counting=1

cur.execute("select cust\_userid from cust\_mst")

storage1=cur.fetchall()

while (field,) not in storage1:

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee==2:

counting=1

while field.strip()=="":

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee==3:

counting=1

while field.strip()=="" or len(field.split())!=1 or not field.isdigit():

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee==1:

counting=1

while field.strip()=="" or len(field.split())!=1:

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=='amt':

counting=1

while field.strip()=="" or not ''.join(field.split('.')).isdigit() or float(field)>balance or float(field)<0:

if counting<2:

field=input(" You have entered incorrect value or more than current balance, please re-enter >> ")

else:

field=input(" You have entered incorrect value or more than current balance, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=='dep':

counting=1

while field.strip()=="" or not ''.join(field.split('.')).isdigit() or float(field)<=0:

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee==5:

counting=1

cur.execute("select cust\_userid from cust\_mst")

storage1=cur.fetchall()

while field.strip()=="" or len(field.split())!=1 or (field,) in storage1:

if counting<2:

field=input(" You have entered incorrect value or already exists, please re-enter >> ")

else:

field=input(" You have entered incorrect value or already exists, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee==6:

counting=1

cur.execute("select acc\_code from cust\_acc where close\_dt is NULL")

storage1=cur.fetchall()

while field.strip()=="" or len(field.split())!=1 or not (field,) in storage1 or field==acc\_code:

if counting<2:

field=input(" You have entered incorrect value or doesn't exist, please re-enter >> ")

else:

field=input(" You have entered incorrect value or doesn't exist, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=='acc':

counting=1

cur.execute("select acc\_code from cust\_acc where cust\_userid='"+newID+"' and close\_dt is NULL")

asd=cur.fetchall()

while not (field,) in asd:

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee[:-1]=='opt':

counting=1

while field.strip()=="" or not field.isdigit() or not int(field) in tuple(i for i in range(1,int(typee[-1])+1)):

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

###############incorrect value pe input statement se pehle aur sahii hone par ek line ka gap dena(jab sab hojaye)

elif typee=='mob':

counting=0

while not field.isdigit() or len(str(int(field)))!=10:

if counting<1:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=='mail':

counting=0

while '@' not in field or '.' not in field.split('@')[1] or len(field.split('@'))!=2 or len(field.split('.'))!=2 or not ''.join(''.join(field.split('.')).split('@')).isalnum():

if counting<1:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=="integer":

counting=0

cur.execute("select cust\_pin from cust\_mst where cust\_userid='"+newID+"'")

storage1=cur.fetchall()

while (field,)!=storage1[0]:

if counting<1:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=="name":

counting=1

names=field.split()

while field.strip()=="" or not ''.join(field.split()).isalpha():

if counting<2:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=="Q":

return field,counting

counting+=1

return field,counting

elif typee=="date":

counting=0

while True:

try:

field=datetime.datetime.strptime(field,"%Y-%m-%d")

if field<datetime.datetime.now():

break

else:

if counting<1:

field=input(" You have entered incorrect value, please re enter >> ")

elif counting>0:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if str(field).upper()=='Q':

return field,counting

counting+=1

except ValueError:

if counting<1:

field=input(" You have entered incorrect value, please re enter >> ")

elif counting>0:

field=input (" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if str(field).upper()=='Q':

return field,counting

counting+=1

return field,counting

"""

elif typee=="date":

counting=0

while field.strip()=="" or '-' not in field or len(field.split('-'))!=3 or not field.replace('-','1').isdigit() or len(field.split('-')[0])!=4 or len(field.split('-')[1])!=2 or len(field.split('-')[2])!=2 or field.split('-')[0]>=datetime.datetime.now().year or field.split('-')[1]>=datetime.datetime.now().month or field.split('-')[2]>=datetime.datetime.now().day:

if counting<1:

field=input(" You have entered incorrect value, please re-enter >> ")

else:

field=input(" You have entered incorrect value, please re-enter or type 'q' to end >> ")

if field.upper()=='Q':

return field,counting

break

counting+=1

return field,counting

"""

#A savings /fixed / recurring bank deposit account can be opened by a minor of any age through his/her natural or legally appointed guardian.

#Minors above the age of 10 years may be allowed to open and operate savings bank accounts independently, if they so desire

def acc\_check():

global ch

global newID

ch=1

count1=0

acc=input(" Do you already have an account with us.(Y/N) >> ")

print()

while acc.upper() not in ('YES','Y','NO','N'):

count1+=1

acc=again(acc,count1)

if count1>=2 and acc.upper()=='Q':

ch=2

print('',end="")

break

while ch!=2:#matlab jab sahi value input ki acc mei

if acc.upper() in ('YES','Y'):

print()

newID=input(" Enter your UserID >> ")

newID,count2=check\_empty(newID,4)

if count2>=2 and newID.upper()=='Q':

ch=2

print('',end="")

break

PIN=input(" Enter your PIN >> ")

PIN,count3=check\_empty(PIN,"integer")

if count3>=1 and PIN.upper()=="Q":

ch=2

print('',end="")

break

loading(" Loading")

print("Successfully Logged In".center(154))

break

else:

print()

print("----- .\*.\*.\*.\*. According to our Bank's policy, Account code cannot be changed once created .\*.\*.\*.\*. -----".center(153))

print()

print("For registering with us: ".center(154))

print()

newID=input(" Enter your New UserID >> ")

newID,count4=check\_empty(newID,5)

if count4>=2 and newID.upper()=='Q':

ch=2

print('',end="")

break

name=input(" Enter your Full Name >> ")

name,count5=check\_empty(name,"name")

if count5>=2 and name.upper()=="Q":

ch=2

print('',end="")

break

sex=input(" Enter your sex(M/F) >> ")

count6=0

while sex.upper() not in ('MALE','FEMALE','M','F'):

count6+=1

sex=again(sex,count6)

if count6>=2 and sex.upper()=='Q':

ch=2

print('',end="")

break

if ch==2:

break

if sex.upper in ('MALE','M'):

sex="M"

else:

sex="F"

add=input(" Enter your Address(In short) >> ")

add,count7=check\_empty(add,2)

if count7>=2 and add.upper()=='Q':

ch=2

print('',end="")

break

city=input(" Enter your City >> ")

city,count8=check\_empty(city,1)

if count8>=2 and city.upper()=='Q':

ch=2

print('',end="")

break

dob=input(" Enter your Date of birth(YYYY-MM-DD) >> ")

dob,count9=check\_empty(dob,"date")

if count9>=1 and str(dob).upper()=="Q":

ch=2

print('',end="")

break

mob=input(" Enter your mobile number >> ")

mob,count10=check\_empty(mob,"mob")

if count10>=1 and mob.upper()=="Q":

ch=2

print('',end="")

break

email=input(" Enter your E-mail address >> ")

email,count11=check\_empty(email,"mail")

if count11>=1 and email.upper()=="Q":

ch=2

print('',end="")

break

while True:

newPIN1=input(" Enter your New PIN >> ")

newPIN1,count12=check\_empty(newPIN1,3)

if count12>=2 and newPIN1.upper()=='Q':

ch=2

print('',end="")

break

newPIN2=input(" Enter your PIN again >> ")

if newPIN1==newPIN2:

#database mei id, pin aur sab detail save karo

loading(" Loading")

print("Your account has been successfully registered and Logged In".center(154))

break

print()

print("Both Pins Didn't match, please re-enter".center(154))

print()

if ch==2:

break

cur.execute("select count(\*) from cust\_acc where acc\_code like 'A%'")

(rows,)=cur.fetchall()[0]

rows=int(rows)+1

code="A0"+str(rows)

cur.execute("insert into cust\_mst values('"+newID+"','"+newPIN2+"','"+name+"','"+sex+"','"+add+"','"+city+"','"+str(dob)+"',"+mob+",'"+email+"')")

mydb.commit()

cur.execute("insert into cust\_acc values('"+newID+"','SNG0123456','"+code+"','"+str(datetime.date.today())+"',NULL,4.00,0.00)")

mydb.commit()

print("Congratulations, Your New Account has been created.".center(154))

print(" Account code for new account is "+code+". All your other credentials are same for this Account.")

print()

break

def menu():

global ch

global newID

global acc\_code

global balance

acc\_if=1

while ch!=2:

print()

print()

print('',end='')

print()

print()

cur.execute("select cust\_name from cust\_mst where cust\_userid='"+newID+"'")

storage1=cur.fetchall()

print(" Hello",list(storage1[0])[0]," ---- &&&& ---- Today is",datetime.date.today())

cur.execute("select count(\*) from cust\_acc where cust\_userid='"+newID+"' and close\_dt is NULL")

if cur.fetchall()[0]==(0,):

print()

print("Your all Accounts are closed with this User ID".center(154))

print()

print()

print(" We offer Following Services:")

print()

print(" >> Open an Account (1)")

print(" >> Exit (2)")

print()

option=input(" To Choose any above option, type the number which is in front of them >> ")

option,count19=check\_empty(option,"opt2")

if count19>=2 and option.upper()=="Q":

ch=2

print('',end="")

break

print()

if option=='1':

cur.execute("select count(\*) from cust\_acc where acc\_code like 'A%'")

(rows,)=cur.fetchall()[0]

rows=int(rows)+1

code="A0"+str(rows)

cur.execute("insert into cust\_acc values('"+newID+"','SNG0123456','"+code+"','"+str(datetime.date.today())+"',NULL,4.00,0.00)")

mydb.commit()

loading(" Loading")

print()

print("Congratulations, Your New Account has been created.".center(154))

print(" Account code for new account is "+code+". All your other credentials are same for this Account.")

elif option=='2':

print()

print("Successfully Logged out".center(153))

ch=2

print('',end="")

break

cur.execute("select count(\*) from cust\_acc where cust\_userid='"+newID+"' and close\_dt is NULL")

if cur.fetchall()[0]!=(0,):

if acc\_if==1:

cur.execute("select acc\_code,cur\_bal from cust\_acc where cust\_userid='"+newID+"' and close\_dt is NULL")

storageacc=cur.fetchall()

(acc\_code,balance)=storageacc[0]

cur.execute("select acc\_code,cur\_bal from cust\_acc where cust\_userid='"+newID+"' and close\_dt is NULL")

storageacc=cur.fetchall()

print()

print(" --- You are currently using account",acc\_code,"---")

print()

print("----".center(153))

#username ki jagah jis username se login hua hai vo aayega

print()

print()

print(" We offer Following Services:")

print()

print(" >> Pay (1)")

print(" >> Passbook (2)")

print(" >> Update Details (3)")

print(" >> Deposit Money (4)")

print(" >> View your Credentials (5)")

print(" >> Open another Account (6)")

print(" >> Close an Account (7)")

print(" >> Exit (8)")

listopt=[str(i) for i in range(1,9)]

if len(storageacc)>1:

listopt.append('9')

print(" >> Use another Account (9)")

print()

option=input(" To Choose any above option, type the number which is in front of them >> ")

count27=0

while option.upper() not in listopt:

count27+=1

option=again(option,count27)

if count27>=2 and option.upper()=='Q':

ch=2

print('',end="")

break

if ch==2:

break

print()

print()

if option=='1':

amt=input(" Enter amount to be paid(in INR) >> ")

amt,count13=check\_empty(amt,"amt")

if count13>=2 and amt.upper()=="Q":

ch=2

print('',end="")

break

tran\_to\_id=input(" Enter the Account Code to which payment will be done >> ")

tran\_to\_id,count14=check\_empty(tran\_to\_id,6)

if count14>=2 and tran\_to\_id.upper()=='Q':

ch=2

print('',end="")

break

cur.execute("select emp\_code,emp\_name from emp\_mst")

(empcode,empname,)=random.choice(cur.fetchall())

cur.execute("select cust\_userid,cur\_bal,bank\_IFSC\_code from cust\_acc where acc\_code='"+tran\_to\_id+"'")

(ucode2,cur\_bal2,bcode2)=cur.fetchall()[0]

cur.execute("insert into bank\_tran values('SNG0123456','"+newID+"','"+acc\_code+"','"+str(datetime.datetime.now().strftime('%Y-%m-%d %X'))+"','debit',"+amt+",'"+empname+"','"+empcode+"',"+str(float(balance))+","+str(float(balance))+"-"+amt+")")

mydb.commit()

cur.execute("insert into bank\_tran values('"+bcode2+"','"+ucode2+"','"+tran\_to\_id+"','"+str(datetime.datetime.now().strftime('%Y-%m-%d %X'))+"','credit',"+amt+",'"+empname+"','"+empcode+"',"+str(float(cur\_bal2))+","+str(float(cur\_bal2))+"+"+amt+")")

mydb.commit()

cur.execute("update cust\_acc set cur\_bal=cur\_bal-"+amt+" where acc\_code='"+acc\_code+"'")

mydb.commit()

cur.execute("update cust\_acc set cur\_bal=cur\_bal+"+amt+" where acc\_code='"+tran\_to\_id+"'")

mydb.commit()

loading(" Loading")

print("Transaction Successful".center(154))

elif option=='2':

cur.execute("select \* from bank\_tran where acc\_code='"+acc\_code+"' order by date\_time\_tran")

print()

lasd=["Bank IFSC Code","User ID","Acc Code","Date and Time","Type","Amount","Emp Name","Emp Code","Opening Balance","Closing Balance"]

print(tabulate.tabulate(cur.fetchall(),headers=lasd,tablefmt="grid"))

elif option=='3':

print()

print()

print("\*\*\*\*As mentioned at the time of creation of account, Account code cannot be changed\*\*\*\*".center(153))

print()

print(" Following credentials can be changed/updated:")

print()

print(" >> User ID (1)")

print(" >> PIN (2)")

print(" >> Full Name (3)")

print(" >> Address and City (4)")

print(" >> Mobile Number (5)")

print(" >> Email Address (6)")

print()

up=input(" To Choose any above option, type the number which is in front of them >> ")

up,count19=check\_empty(up,"opt6")

if count19>=2 and up.upper()=="Q":

ch=2

print('',end="")

break

print()

if up=='1':

newID1=input(" Enter your New UserID >> ")

newID1,count20=check\_empty(newID1,5)

if count20>=2 and newID1.upper()=='Q':

ch=2

print('',end="")

break

cur.execute("update cust\_mst set cust\_userid='"+newID1+"' where cust\_userid='"+newID+"'")

mydb.commit()

cur.execute("update cust\_acc set cust\_userid='"+newID1+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated User ID".center(154))

newID=newID1

elif up=='2':

while True:

newPIN1=input(" Enter your New PIN >> ")

newPIN1,count21=check\_empty(newPIN1,3)

if count21>=2 and newPIN1.upper()=='Q':

ch=2

print('',end="")

break

newPIN2=input(" Enter your PIN again >> ")

if newPIN1==newPIN2:

cur.execute("update cust\_mst set cust\_PIN='"+newPIN2+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated PIN".center(154))

break

print()

print("Both Pins Didn't match, please re-enter".center(154))

print()

elif up=='3':

name=input(" Enter your Full Name >> ")

name,count22=check\_empty(name,"name")

if count22>=2 and name.upper()=="Q":

ch=2

print('',end="")

break

cur.execute("update cust\_mst set cust\_name='"+name+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated Name".center(154))

elif up=='4':

add=input(" Enter your Address(In short) >> ")

add,count23=check\_empty(add,2)

if count23>=2 and add.upper()=='Q':

ch=2

print('',end="")

break

city=input(" Enter your City >> ")

city,count24=check\_empty(city,1)

if count24>=2 and city.upper()=='Q':

ch=2

print('',end="")

break

cur.execute("update cust\_mst set cust\_add='"+name+"', cust\_city='"+city+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated Address and City".center(154))

elif up=='5':

mob=input(" Enter your mobile number >> ")

mob,count25=check\_empty(mob,"mob")

if count25>=1 and mob.upper()=="Q":

ch=2

print('',end="")

break

cur.execute("update cust\_mst set cust\_mobile='"+mob+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated Mobile Number".center(154))

elif up=='6':

email=input(" Enter your E-mail address >> ")

email,count26=check\_empty(email,"mail")

if count26>=1 and email.upper()=="Q":

ch=2

print('',end="")

break

cur.execute("update cust\_mst set cust\_email='"+email+"' where cust\_userid='"+newID+"'")

mydb.commit()

print()

loading(" Loading")

print()

print("Successfully Updated Mobile Number".center(154))

print()

print("Your Updated Credentials are:-".center(154))

print()

cur.execute("select \* from cust\_mst where cust\_userid='"+newID+"'")

print()

lasd1=["User ID","PIN","Name","Sex","Address","City","DOB","Mobile Number","Email Address"]

print(tabulate.tabulate(cur.fetchall(),headers=lasd1,tablefmt="grid"))

lasd1=["Number of Accounts Active","Account Code(s)"]

cur.execute("select acc\_code from cust\_acc where cust\_userid='"+newID+"'and close\_dt is NULL" )

acodelist=cur.fetchall()

print(tabulate.tabulate(((str(len(acodelist)),)+(' '.join(list(i for (i,) in acodelist)),),),headers=lasd1,tablefmt="grid"))

elif option=='4':

amt=input(" Enter amount to be deposited(in INR) >> ")

amt,count17=check\_empty(amt,"dep")

if count17>=2 and amt.upper()=="Q":

ch=2

print('',end="")

break

cur.execute("select emp\_code,emp\_name from emp\_mst")

(empcode,empname,)=random.choice(cur.fetchall())

cur.execute("insert into bank\_tran values('SNG0123456','"+newID+"','"+acc\_code+"','"+str(datetime.datetime.now().strftime('%Y-%m-%d %X'))+"','Deposit',"+amt+",'"+empname+"','"+empcode+"',"+str(float(balance))+","+str(float(balance))+"+"+amt+")")

mydb.commit()

cur.execute("update cust\_acc set cur\_bal=cur\_bal+"+amt+" where acc\_code='"+acc\_code+"'")

mydb.commit()

loading(" Loading")

print("Transaction Successful".center(154))

elif option=='5':

print("Your Credentials are:-".center(154))

print()

cur.execute("select \* from cust\_mst where cust\_userid='"+newID+"'")

print()

lasd1=["User ID","PIN","Name","Sex","Address","City","DOB","Mobile Number","Email Address"]

print(tabulate.tabulate(cur.fetchall(),headers=lasd1,tablefmt="grid"))

lasd1=["Account Code","Current Balance","Opening Date","Closing Date"]

cur.execute("select acc\_code,cur\_bal,open\_dt,close\_dt from cust\_acc where cust\_userid='"+newID+"'")

print(tabulate.tabulate(cur.fetchall(),headers=lasd1,tablefmt="grid"))

elif option=='6':

print()

accs=input(" Are you sure that you want to create another account? (Y/N) >> ")

count18=0

while accs.upper() not in ('YES','NO','Y','N'):

count18+=1

accs=again(accs,count18)

if count18>=2 and accs.upper()=='Q':

ch=2

print('',end="")

break

if accs.upper() in ('YES','Y'):

cur.execute("select count(\*) from cust\_acc where acc\_code like 'A%'")

(rows,)=cur.fetchall()[0]

rows=int(rows)+1

code="A0"+str(rows)

cur.execute("insert into cust\_acc values('"+newID+"','SNG0123456','"+code+"','"+str(datetime.date.today())+"',NULL,4.00,0.00)")

mydb.commit()

print()

loading(" Loading")

print()

print("Congratulations, Your New Account has been created.".center(154))

print(" Account code for new account is "+code+". All your other credentials are same for this Account.")

elif option=='7':

print()

print(" You have following Accounts registered on",newID,"User ID:")

print()

lasd1=[" Active Account Code(s) "," Current Balance(INR) "]

print(tabulate.tabulate(storageacc,headers=lasd1,tablefmt="grid"))

print()

acc\_rem=input(" Type the Account Code you want to close >> ")

acc\_rem,count28=check\_empty(acc\_rem,"acc")

if count28>=2 and acc\_rem.upper()=="Q":

ch=2

print('',end="")

break

print()

if acc\_rem==acc\_code:

acc\_if=1

cur.execute("update cust\_acc set close\_dt='"+str(datetime.date.today())+"' where acc\_code='"+acc\_rem+"'")

mydb.commit()

print("Successfully closed".center(154))

elif option=='8':

print("Successfully Logged out".center(153))

ch=2

print('',end="")

break

elif option=='9':

print()

print(" You have following Accounts registered on",newID,"User ID:")

print()

lasd1=[" Active Account Code(s) "," Current Balance(INR) "]

print(tabulate.tabulate(storageacc,headers=lasd1,tablefmt="grid"))

print()

acc\_choice=input(" Type the Account Code you want to swap to >> ")

acc\_choice,count27=check\_empty(acc\_choice,"acc")

if count27>=2 and acc\_choice.upper()=="Q":

ch=2

print('',end="")

break

print()

print("Successfully changed".center(154))

acc\_if=2

acc\_code=acc\_choice

cur.execute("select cur\_bal from cust\_acc where acc\_code='"+acc\_code+"'")

(balance,)=cur.fetchall()[0]

print()

print()

ask=input(" Do you again want to go to menu?(Y/N) >> ")

count16=0

while ask.upper() not in ('YES','Y','NO','N'):

count16+=1

ask=again(ask,count16)

if (count16>=2 and ask.upper()=='Q') or ask.upper() in ('NO','N'):

ch=2

print('',end="")

break

mydb=mysql.connector.connect(host="localhost",user="root",passwd="iamrohan",database="my\_banking\_system")

cur=mydb.cursor()

loading(" Getting Banking Systems Online")

print(" Secure Connection Established",end="")

print()

print()

print()

print()

time.sleep(1.3)

print("---- Welcome to SNG Bank Portal ----".center(154))

print()

print()

time.sleep(0.4)

print("\*\*\*\*\*\*\*\*\*\*SNG Bank never asks for PIN and OTP from customers\*\*\*\*\*\*\*\*\*\*".center(154))

print()

print()

print()

print()

time.sleep(0.4)

acc\_check()

while ch!=2:

menu()

ch=2

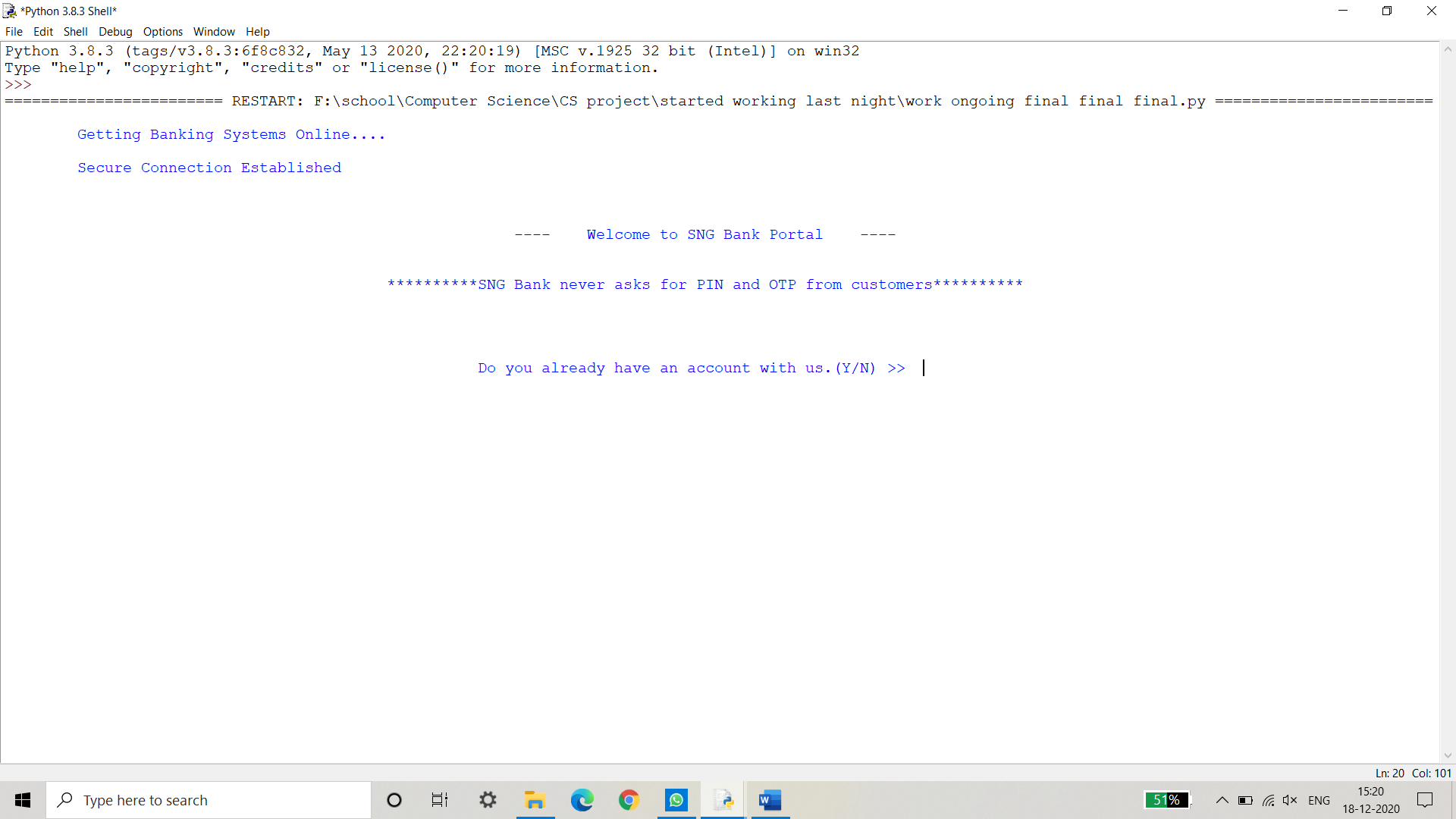
else:

print()

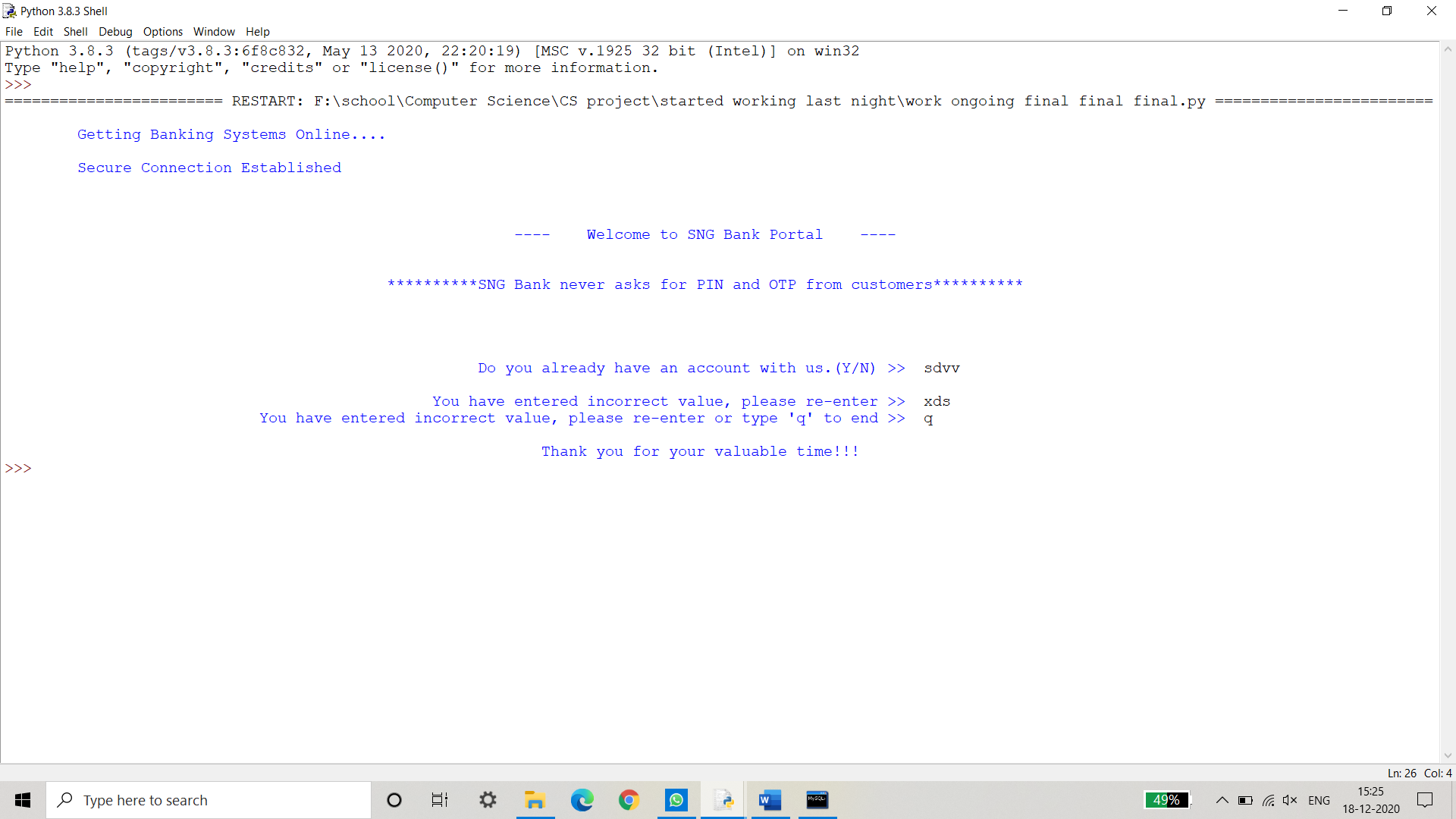
print("Thank you for your valuable time!!!".center(154))

Code Output

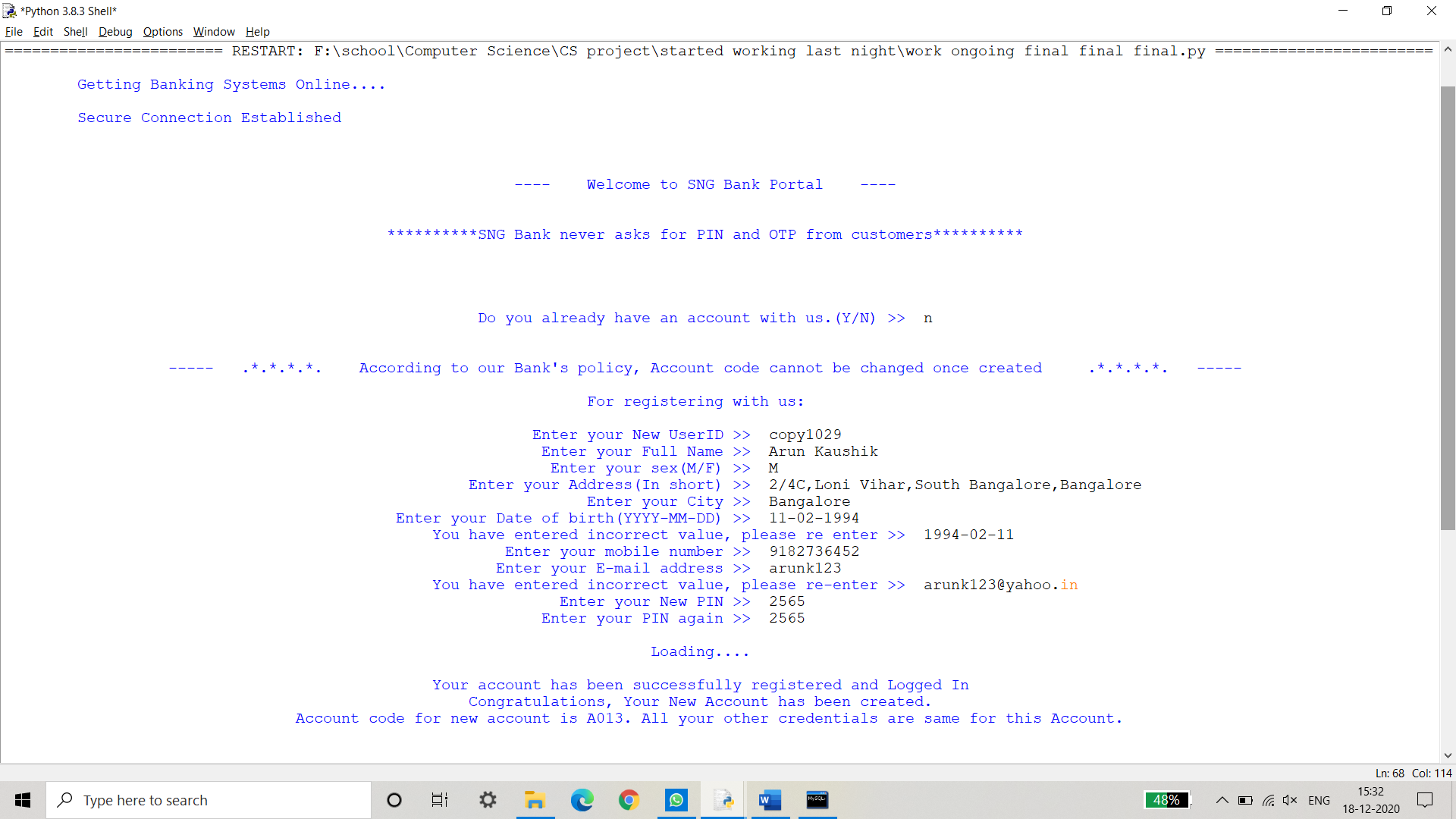
1.First Interface



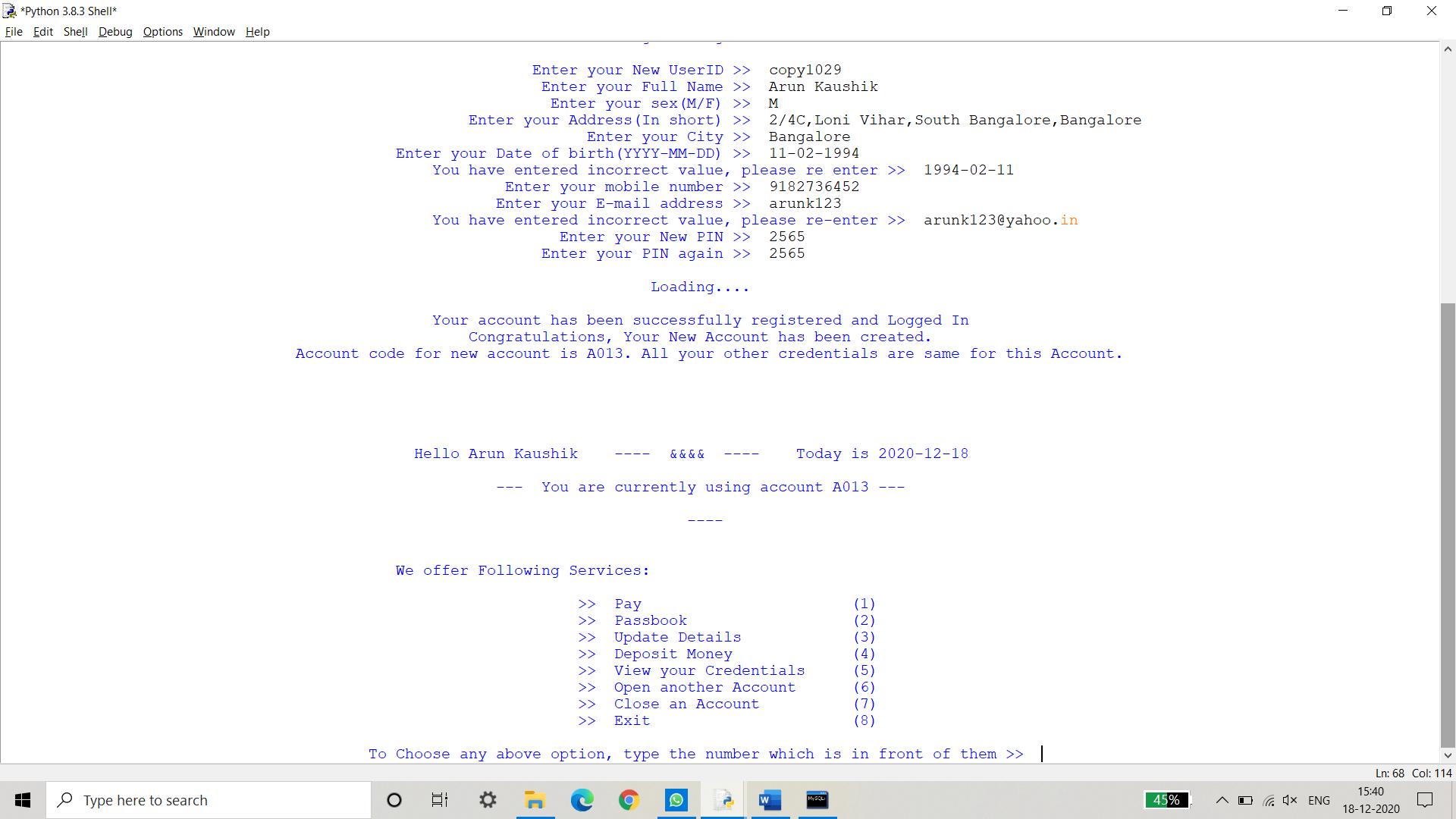
2.Wrong input



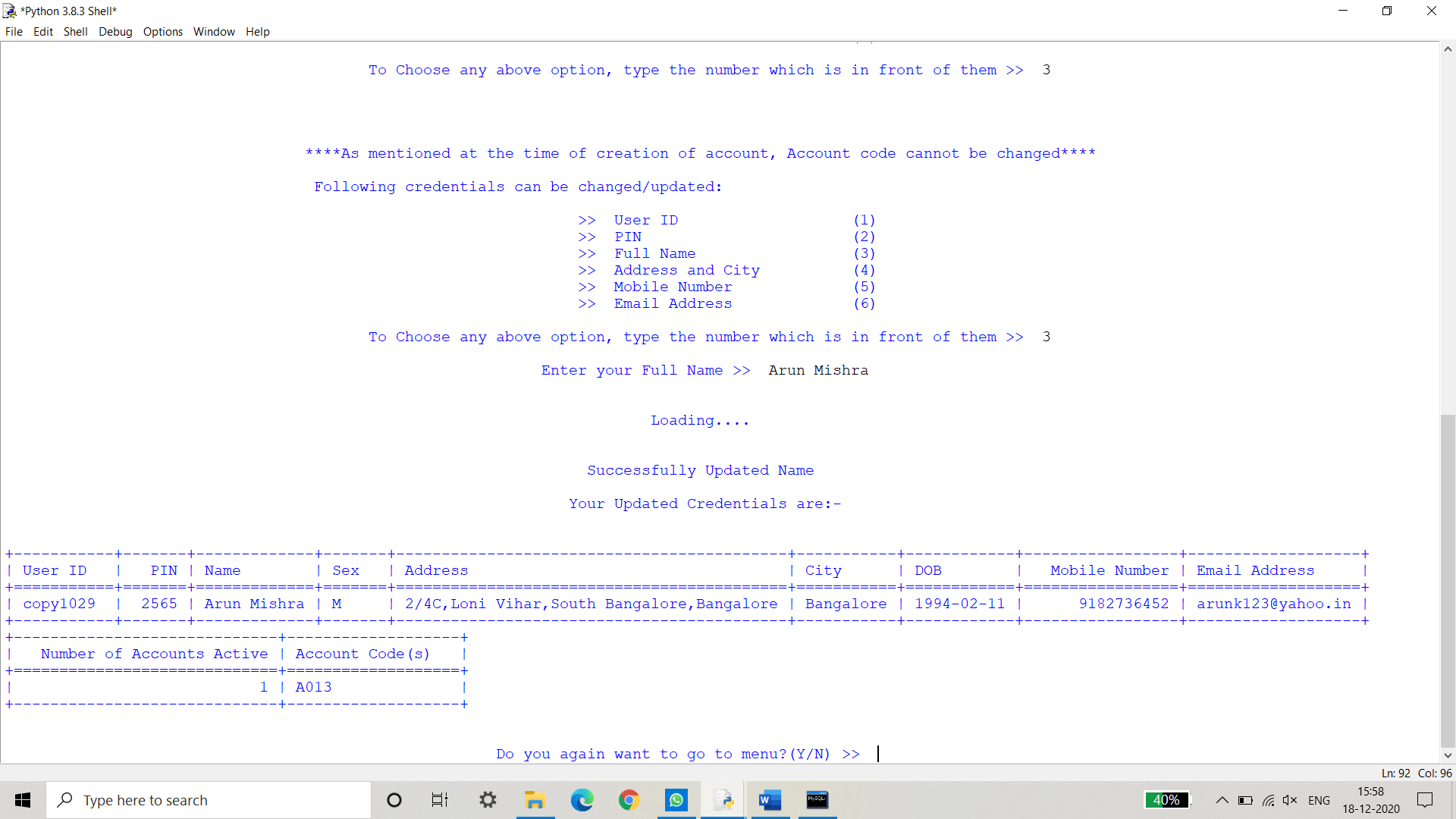
3.New customer



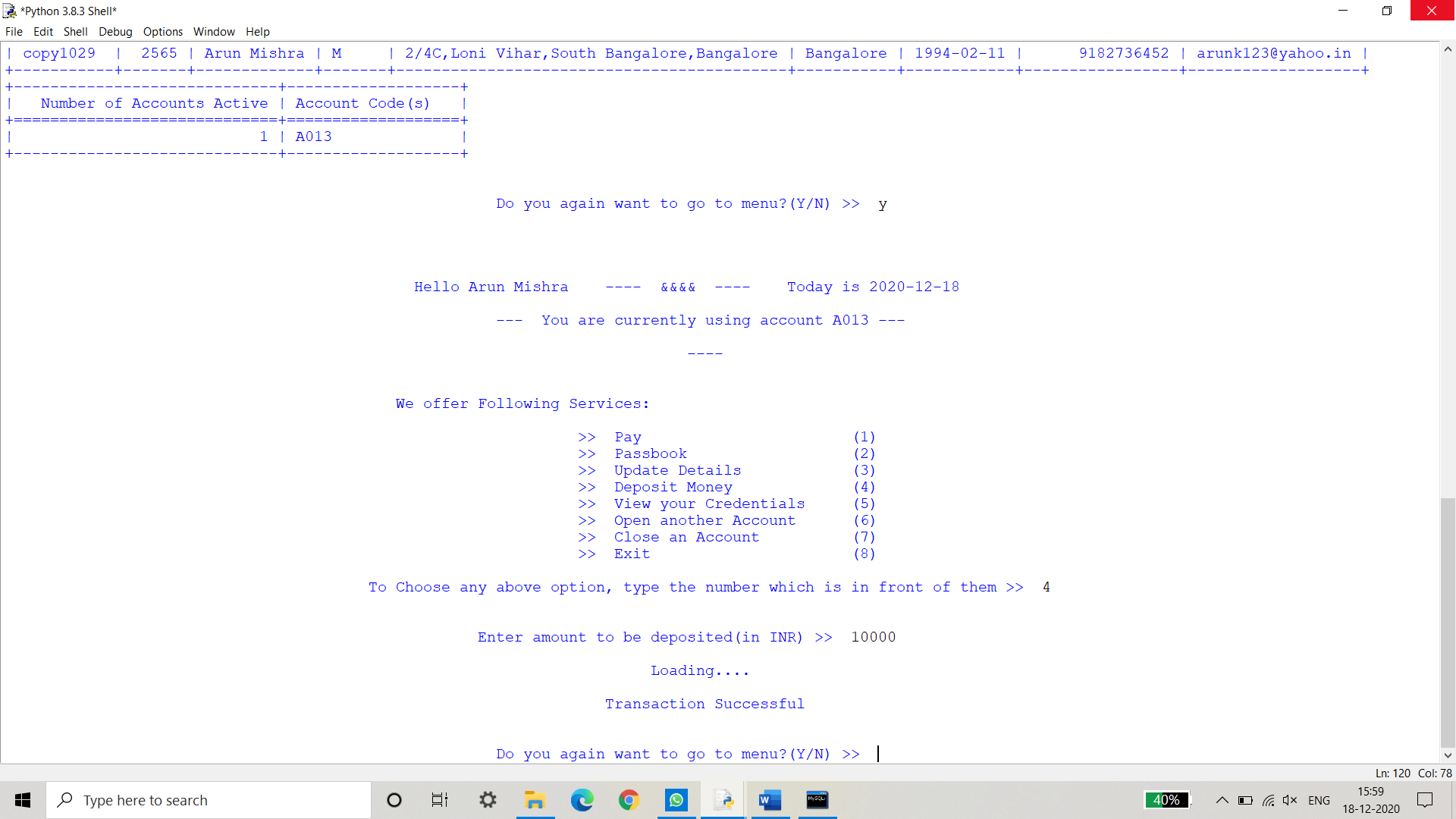
4.After UserID created



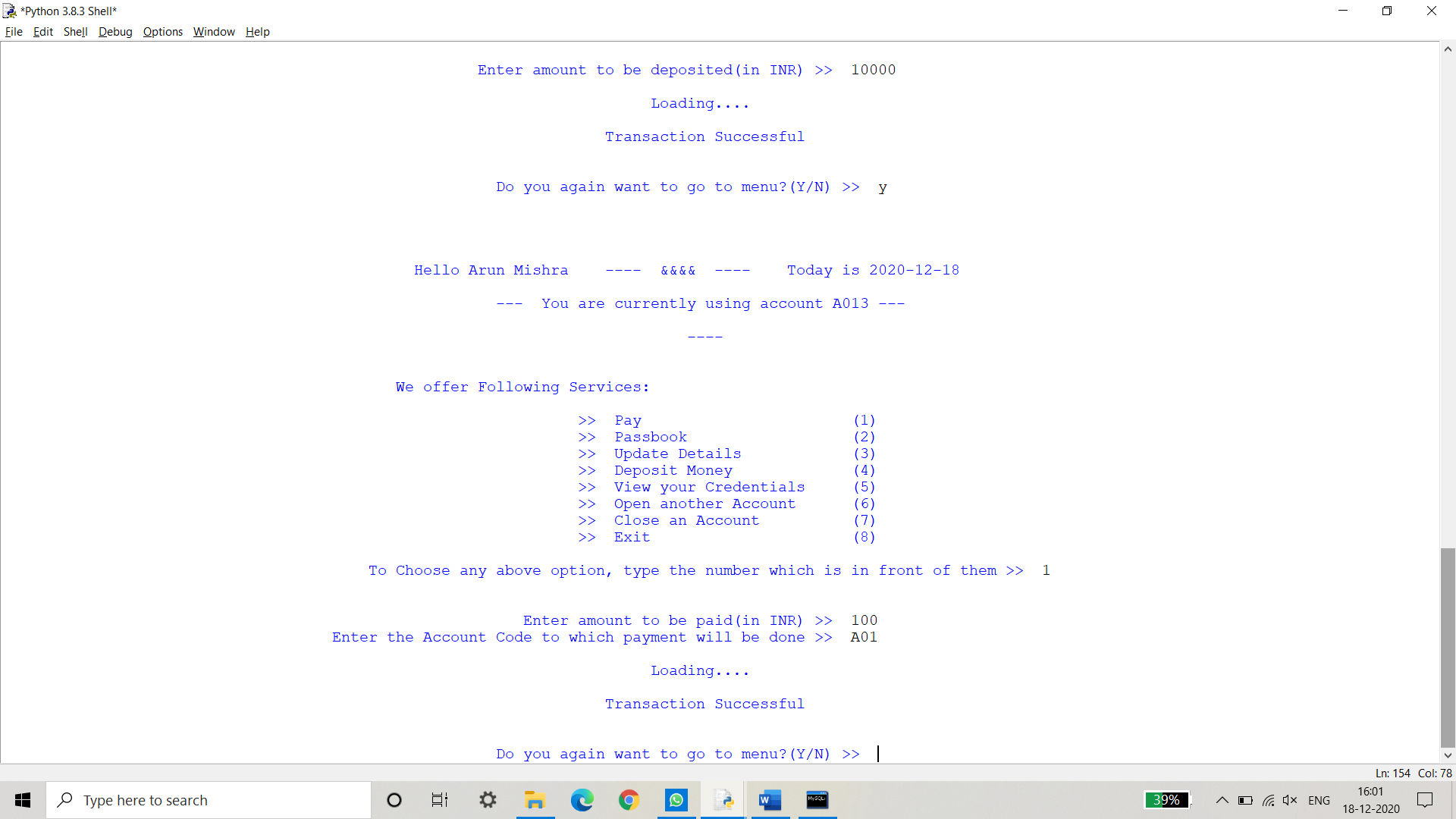
5.Updating details



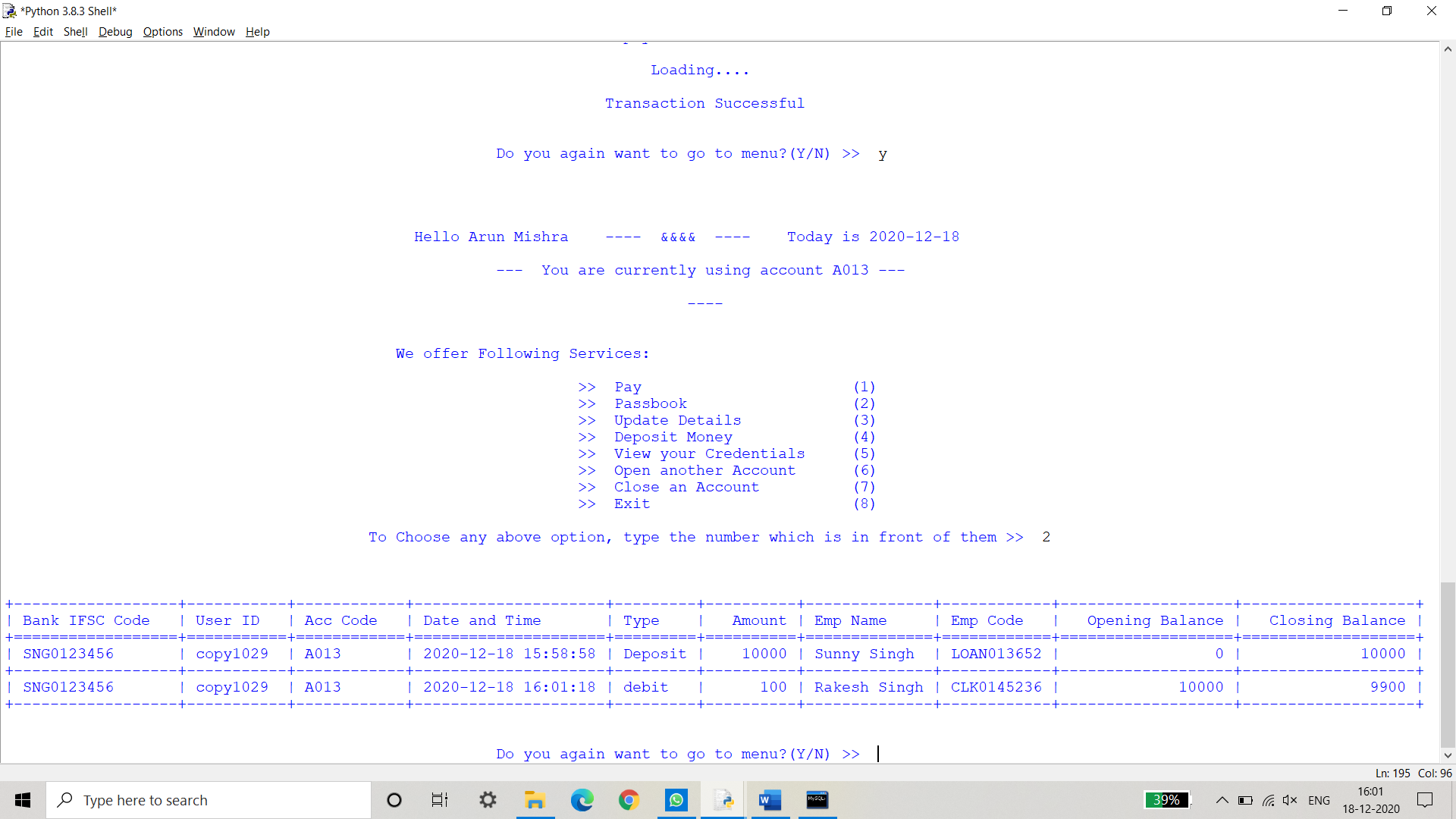
6.Deposit money



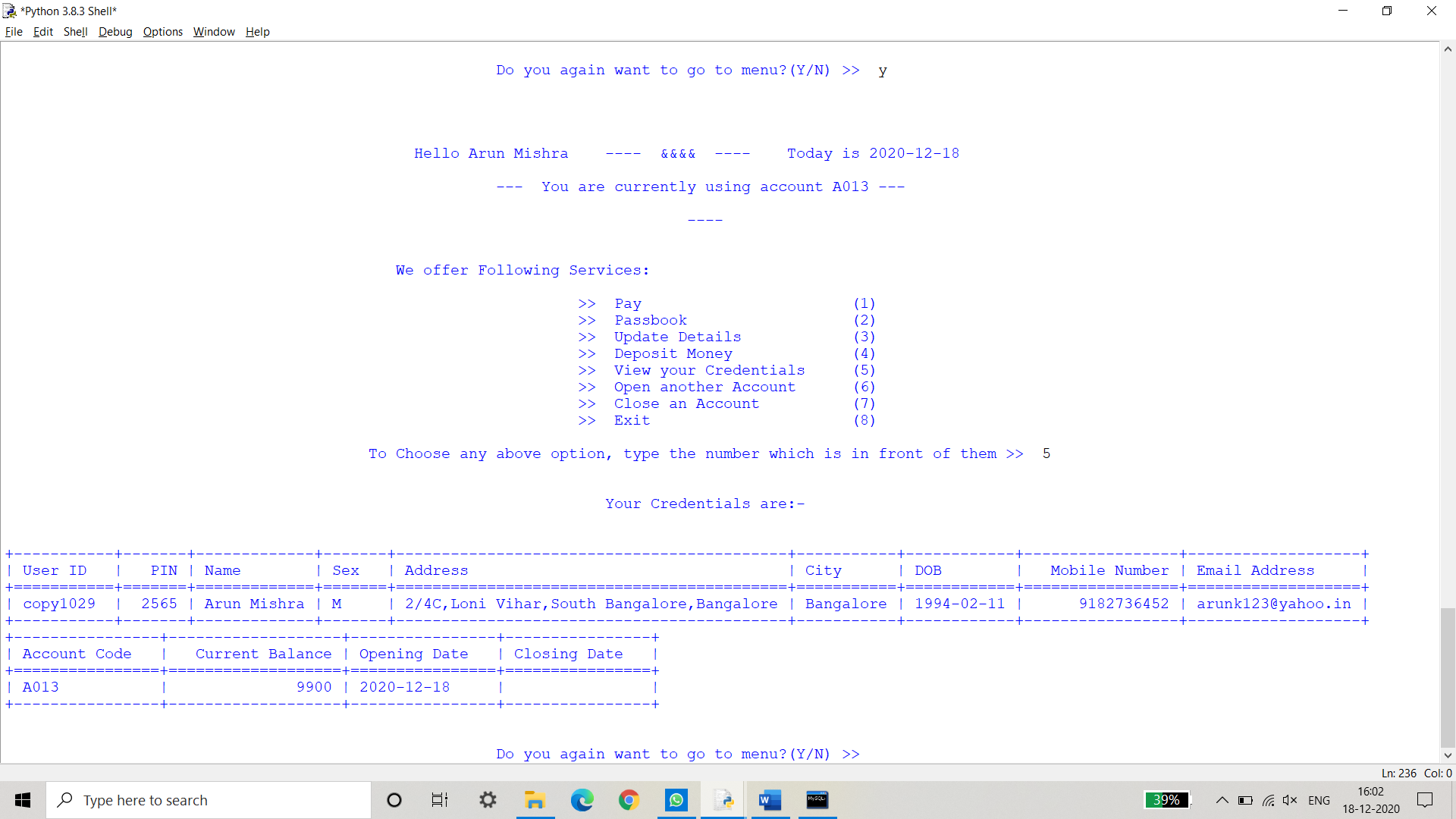
7.Pay money



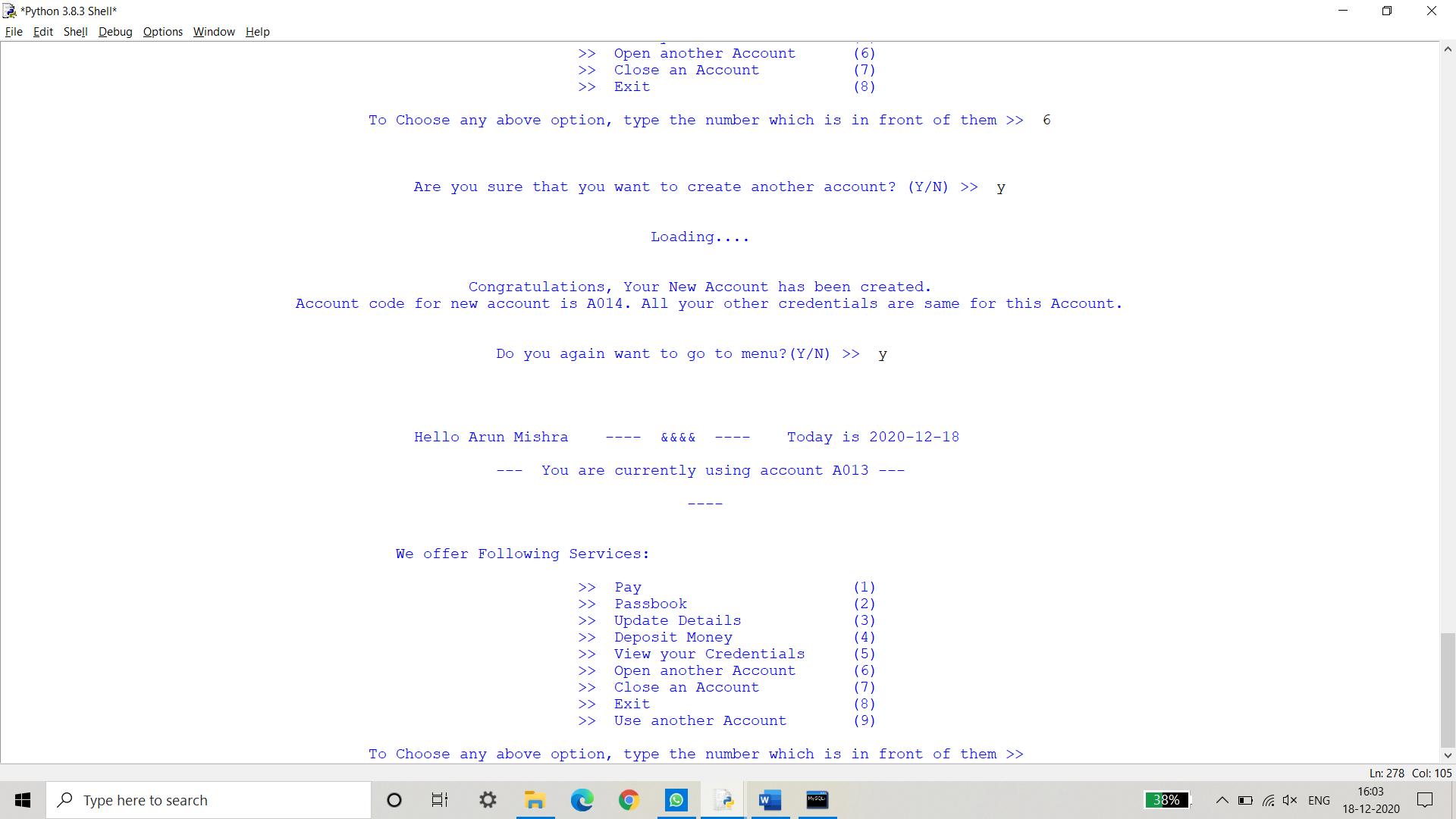
8.Passbook



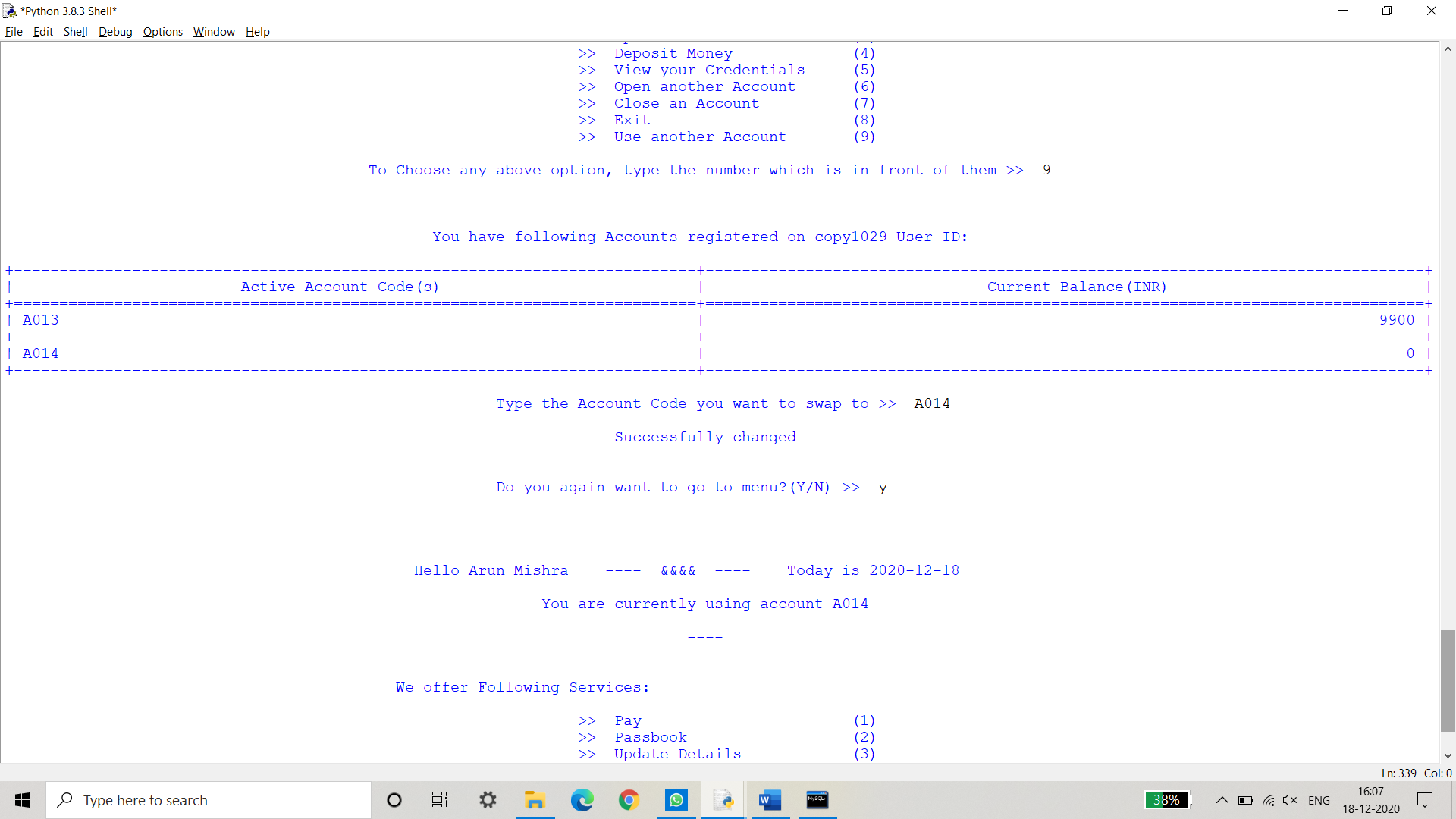
9.View credentials



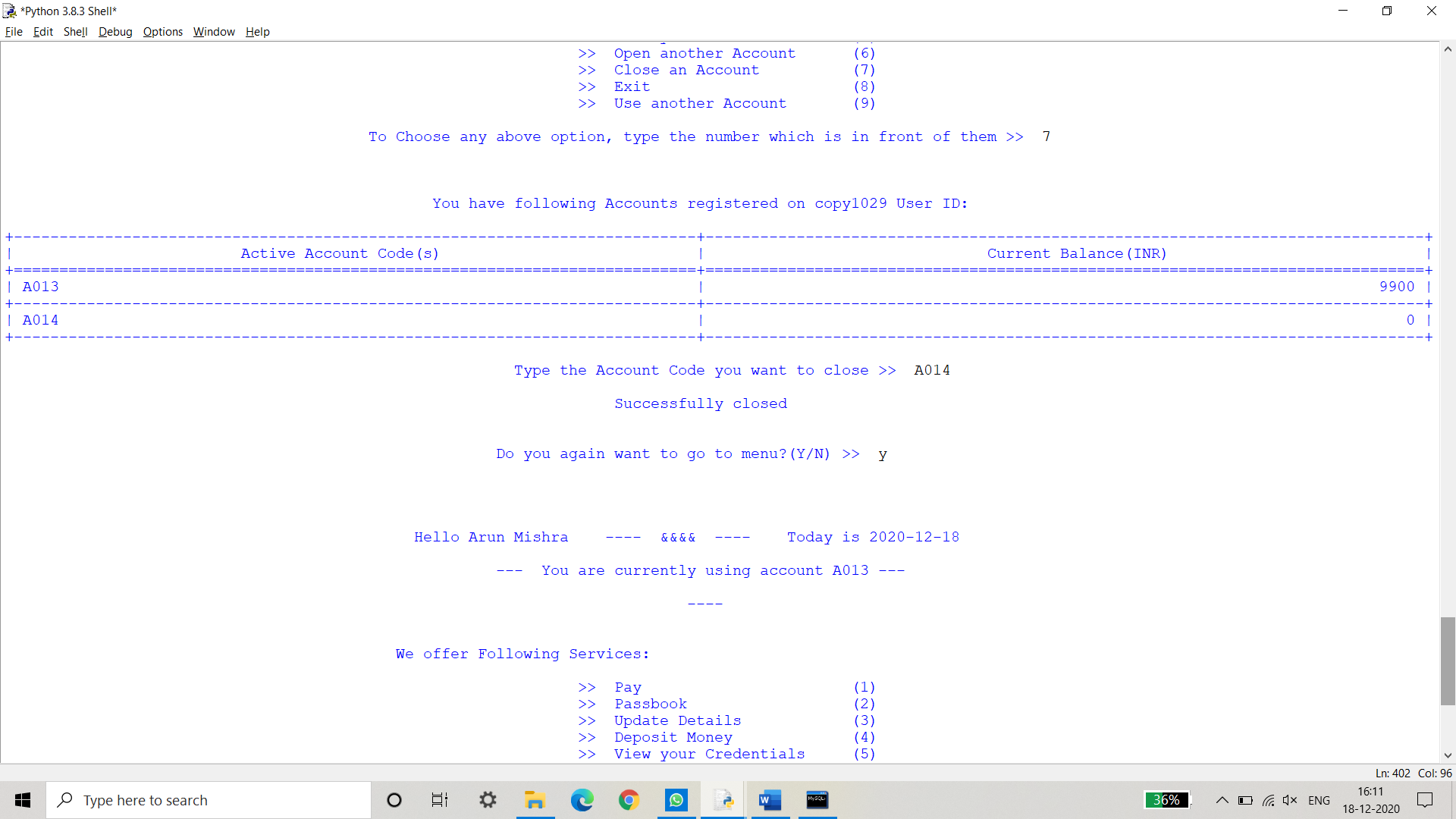
10.Opening new account

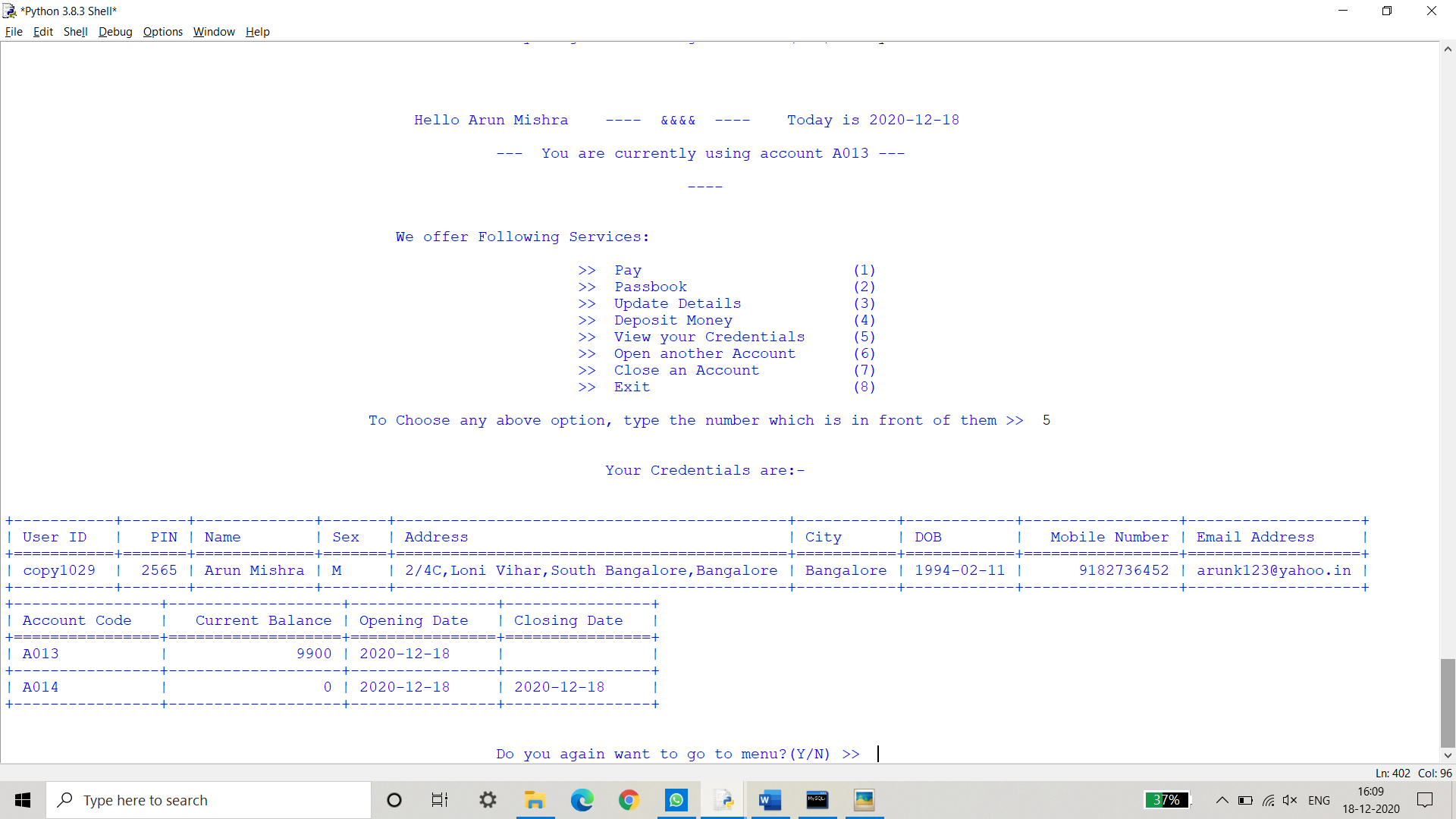


11.Swaping to another account

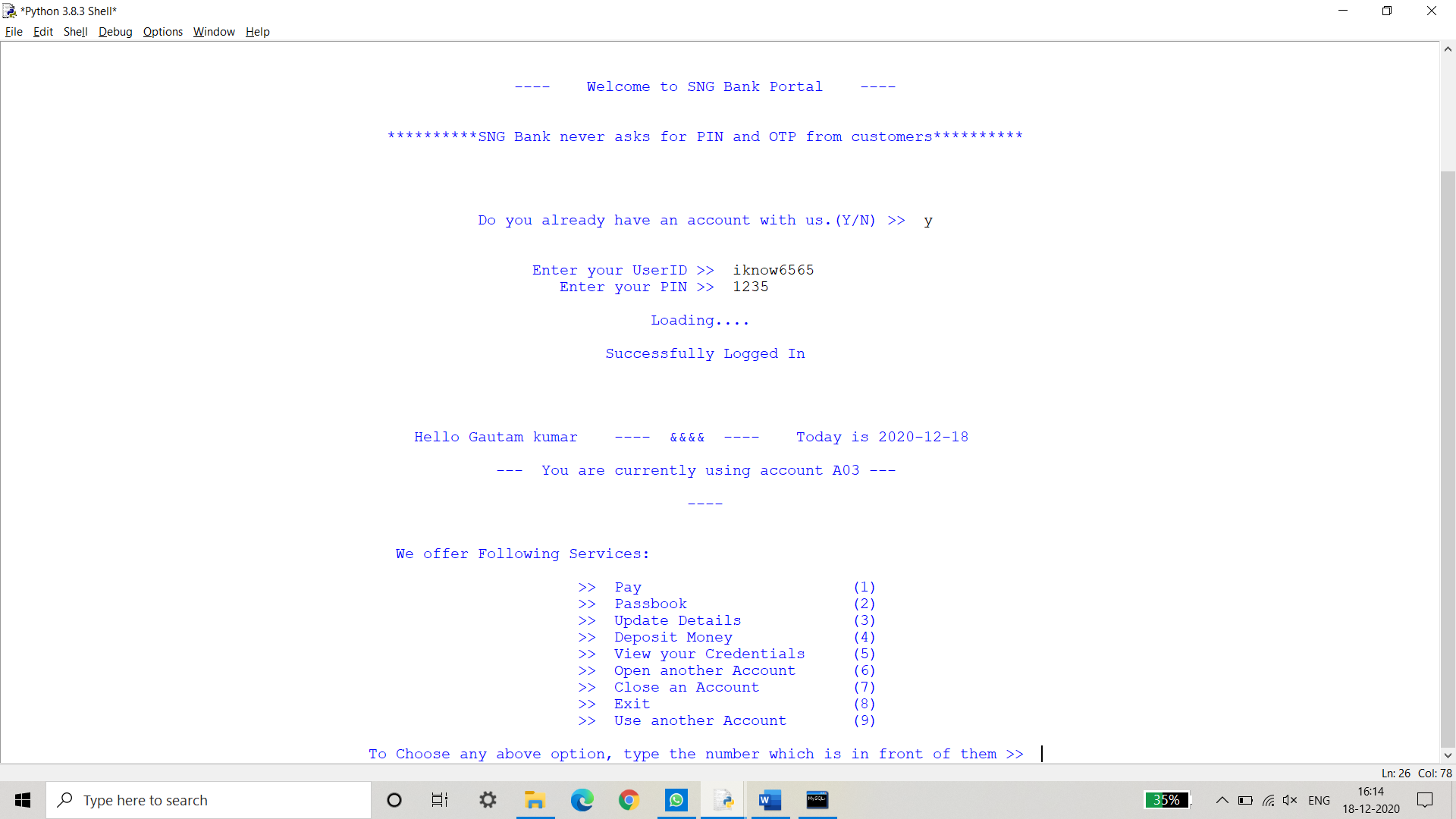


12.Closing an account

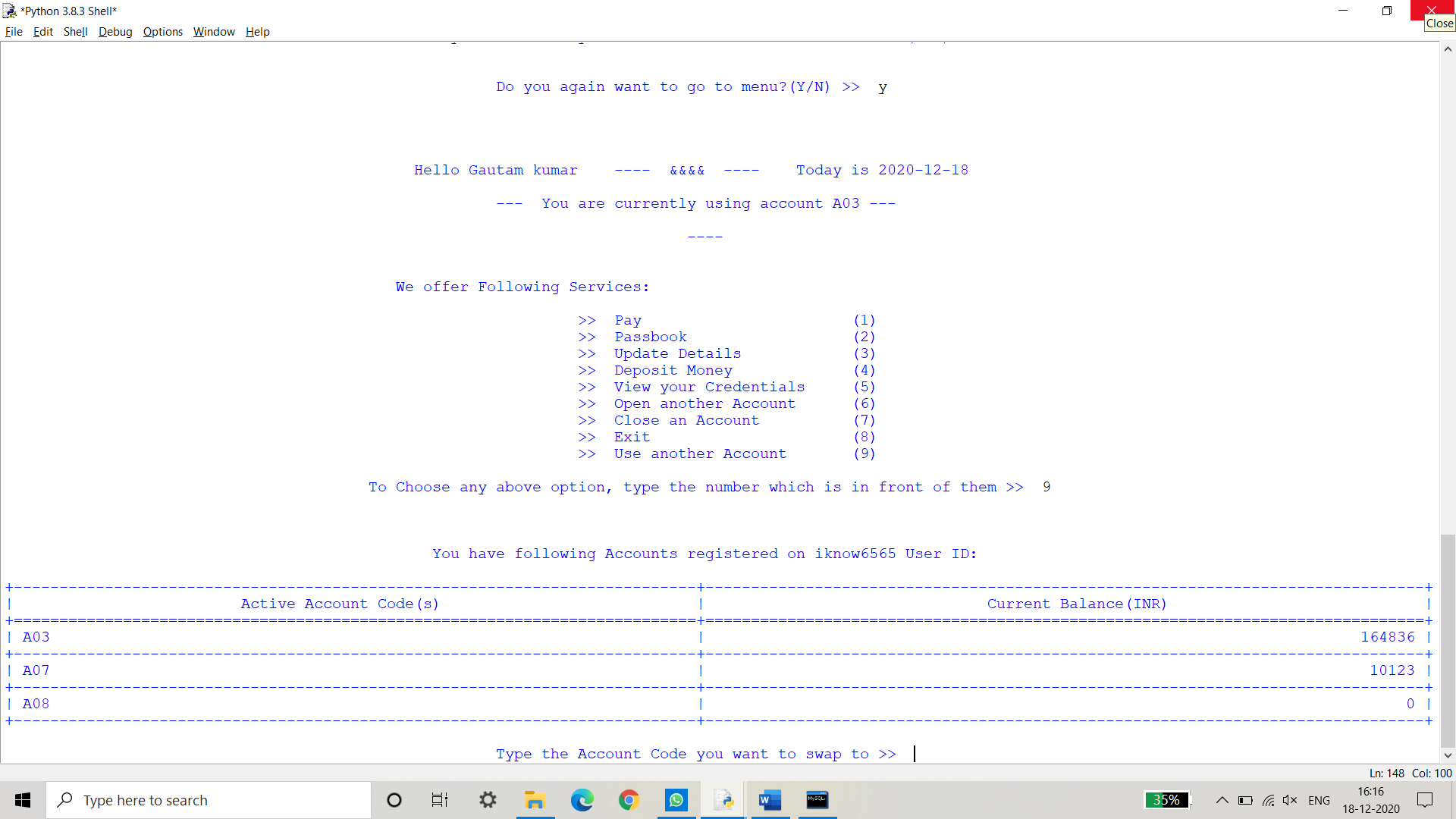




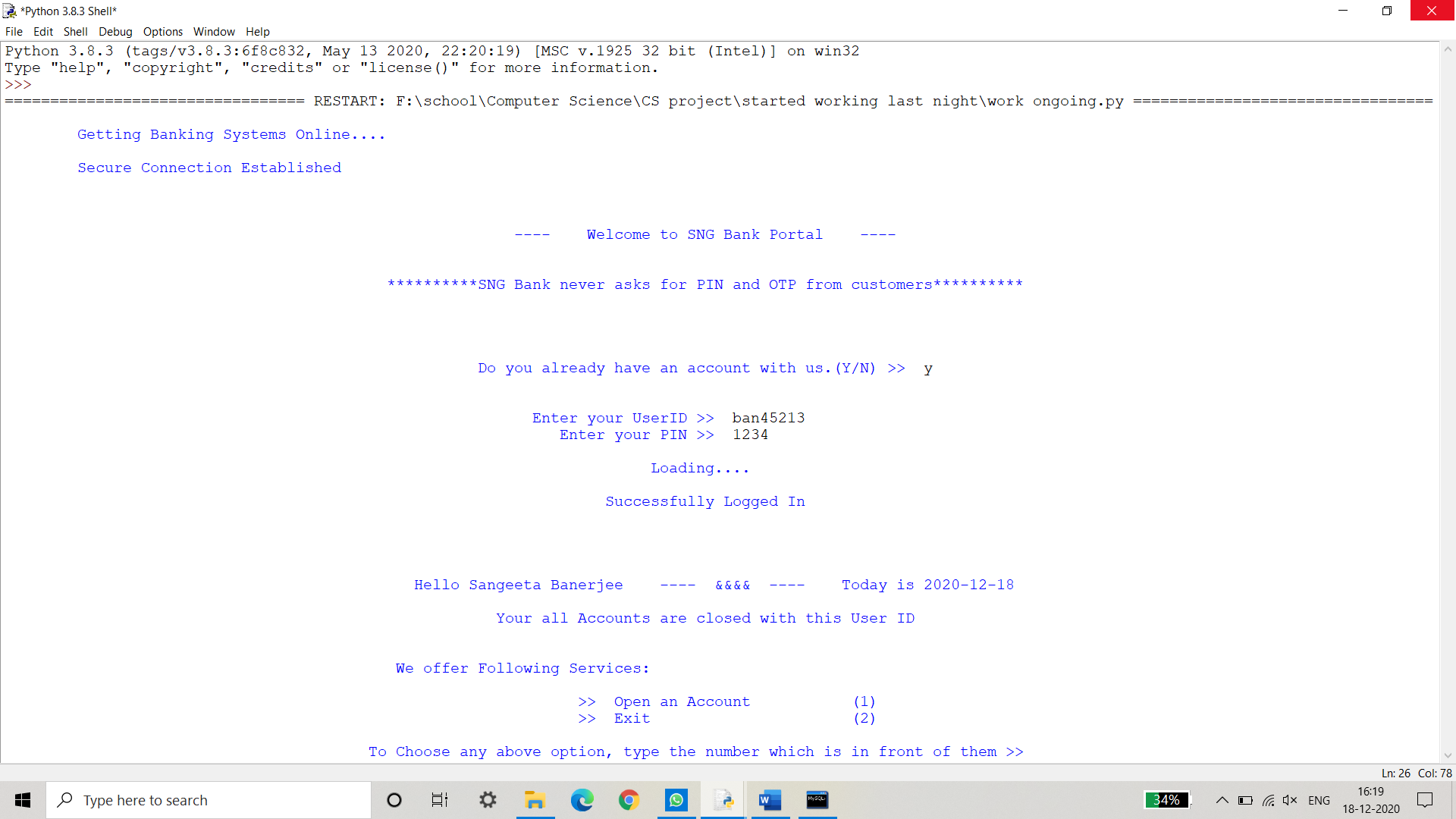
13.Old customer



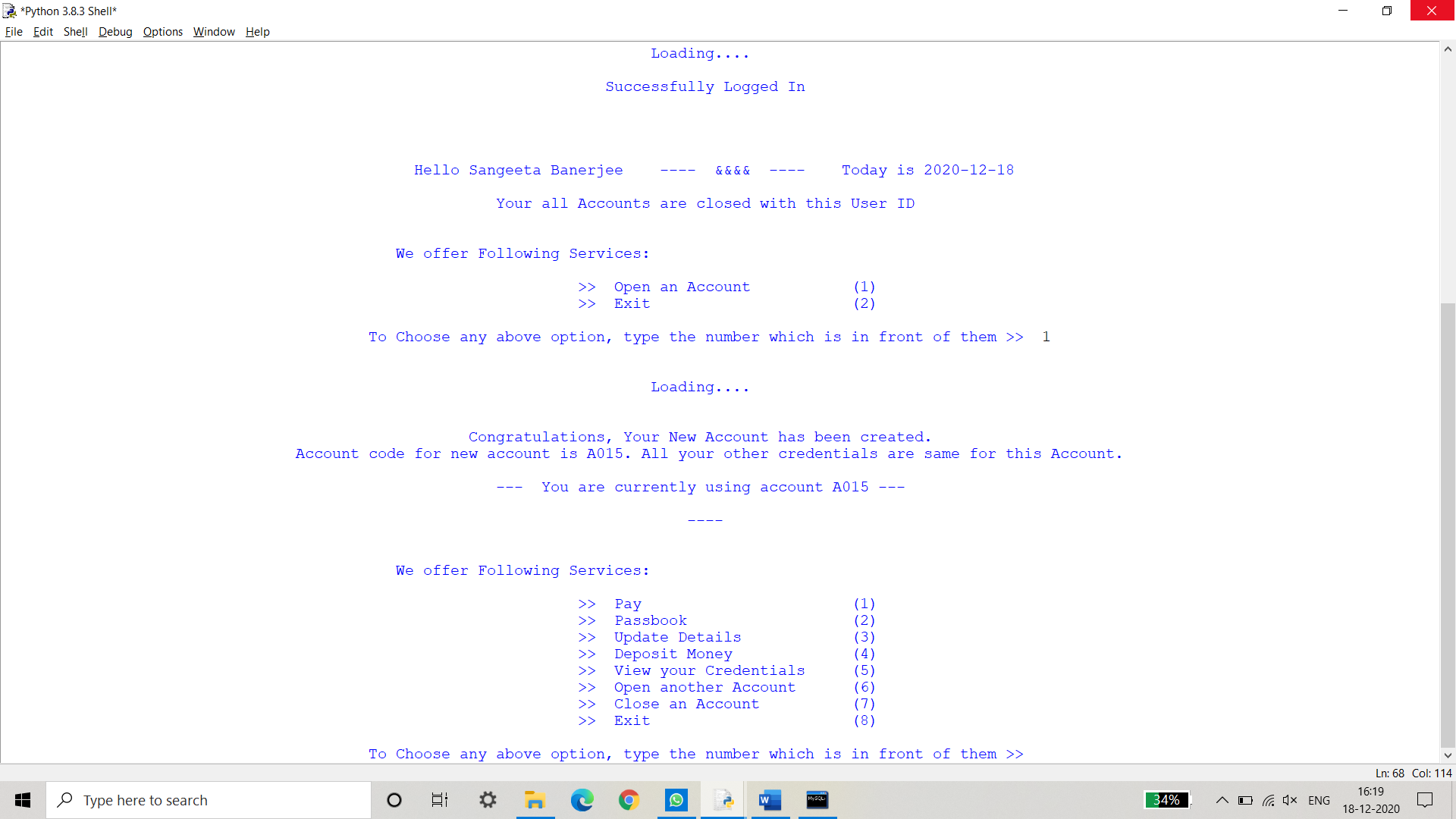
14.Closed accounts not shown while swaping accounts



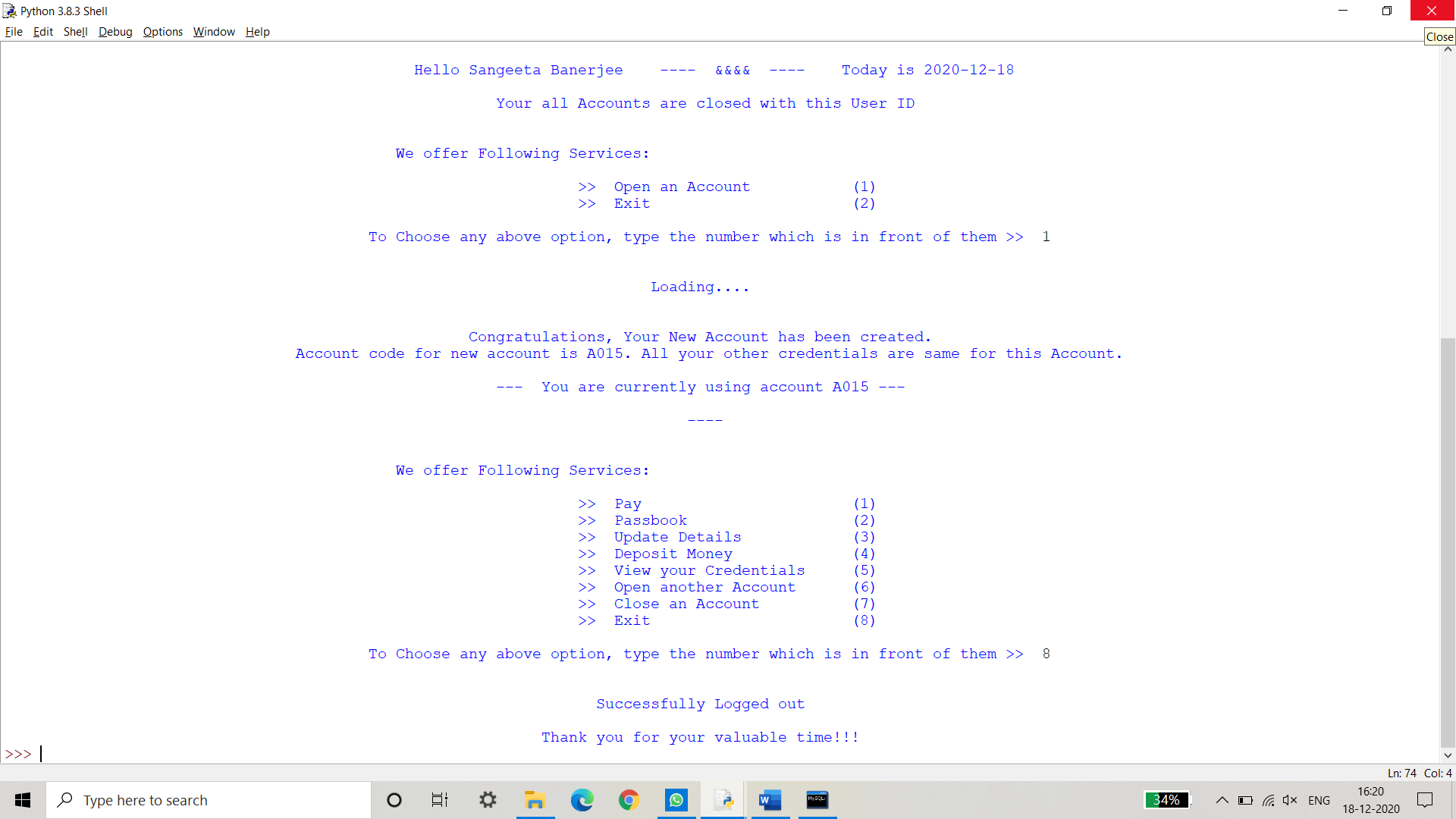
15.When all accounts are closed



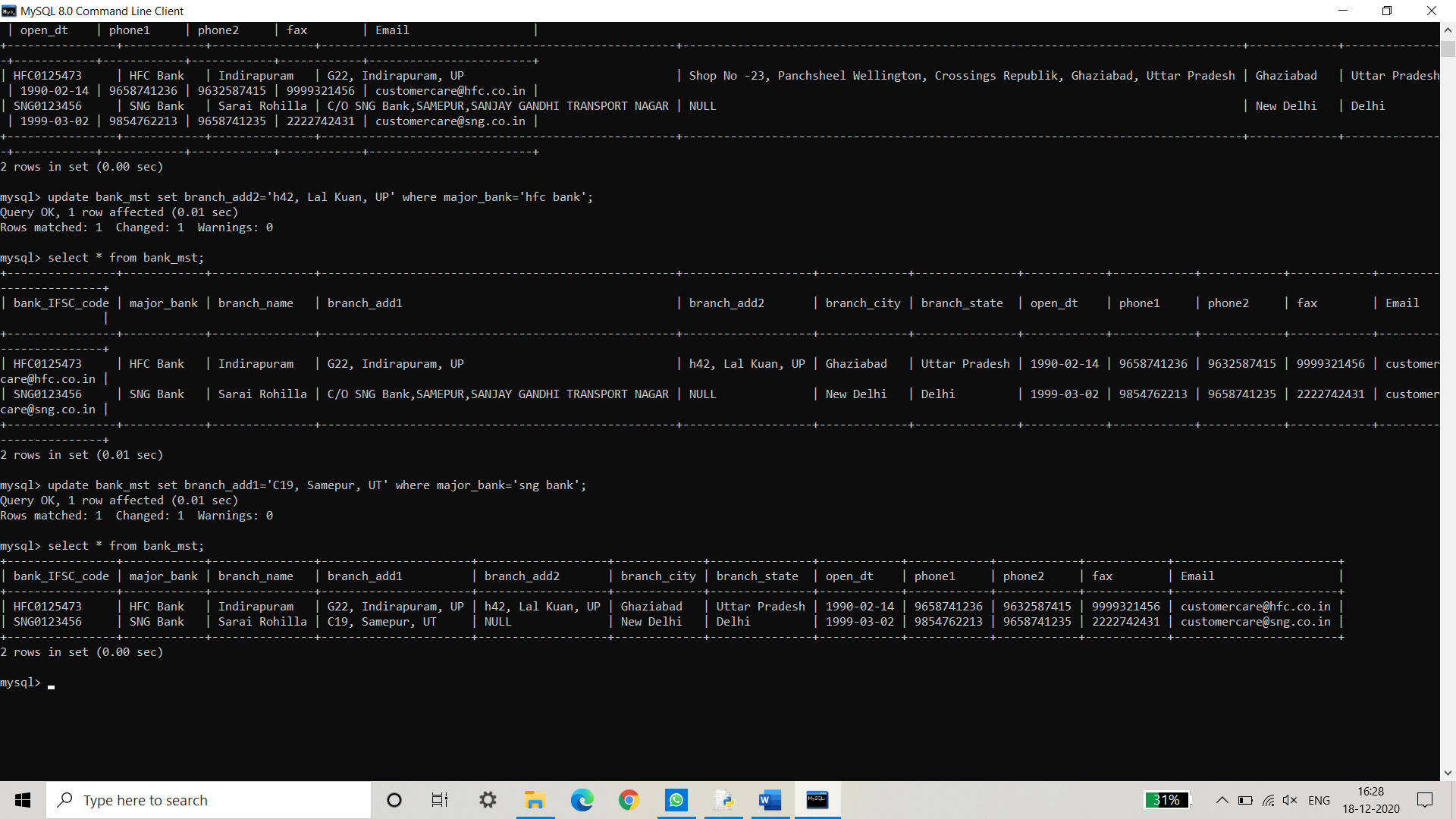
16.After creating an account

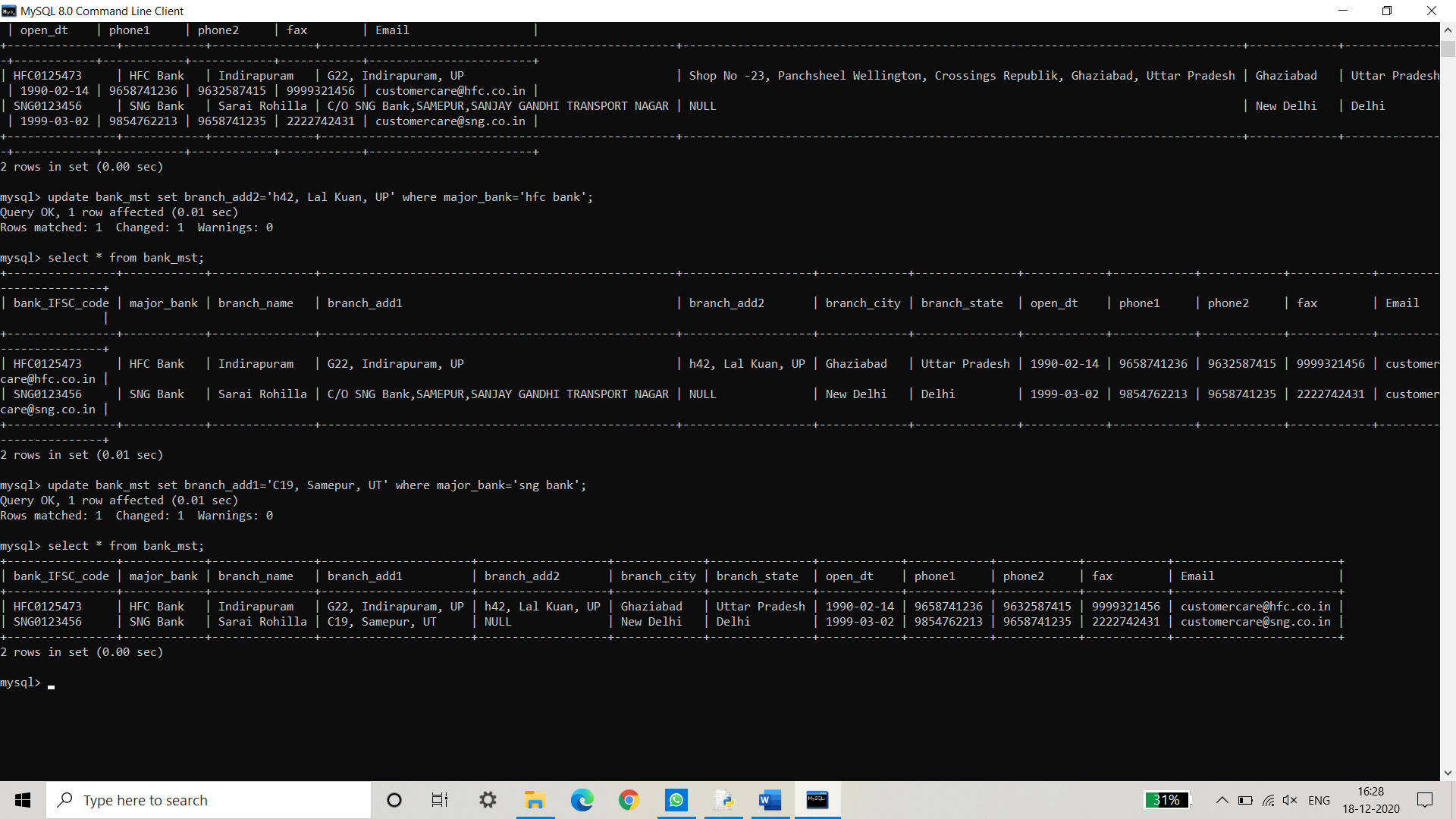


17.Exit

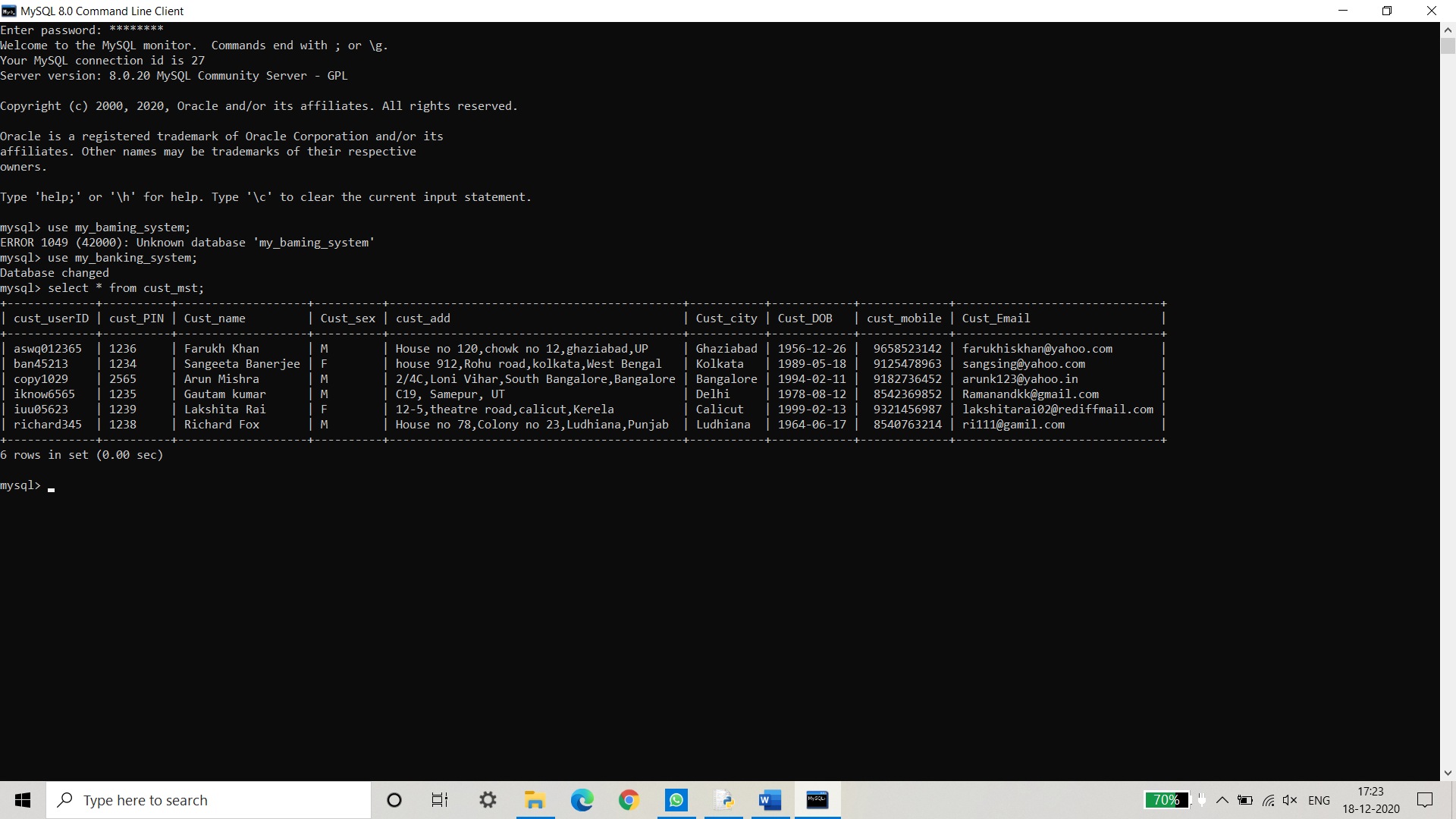


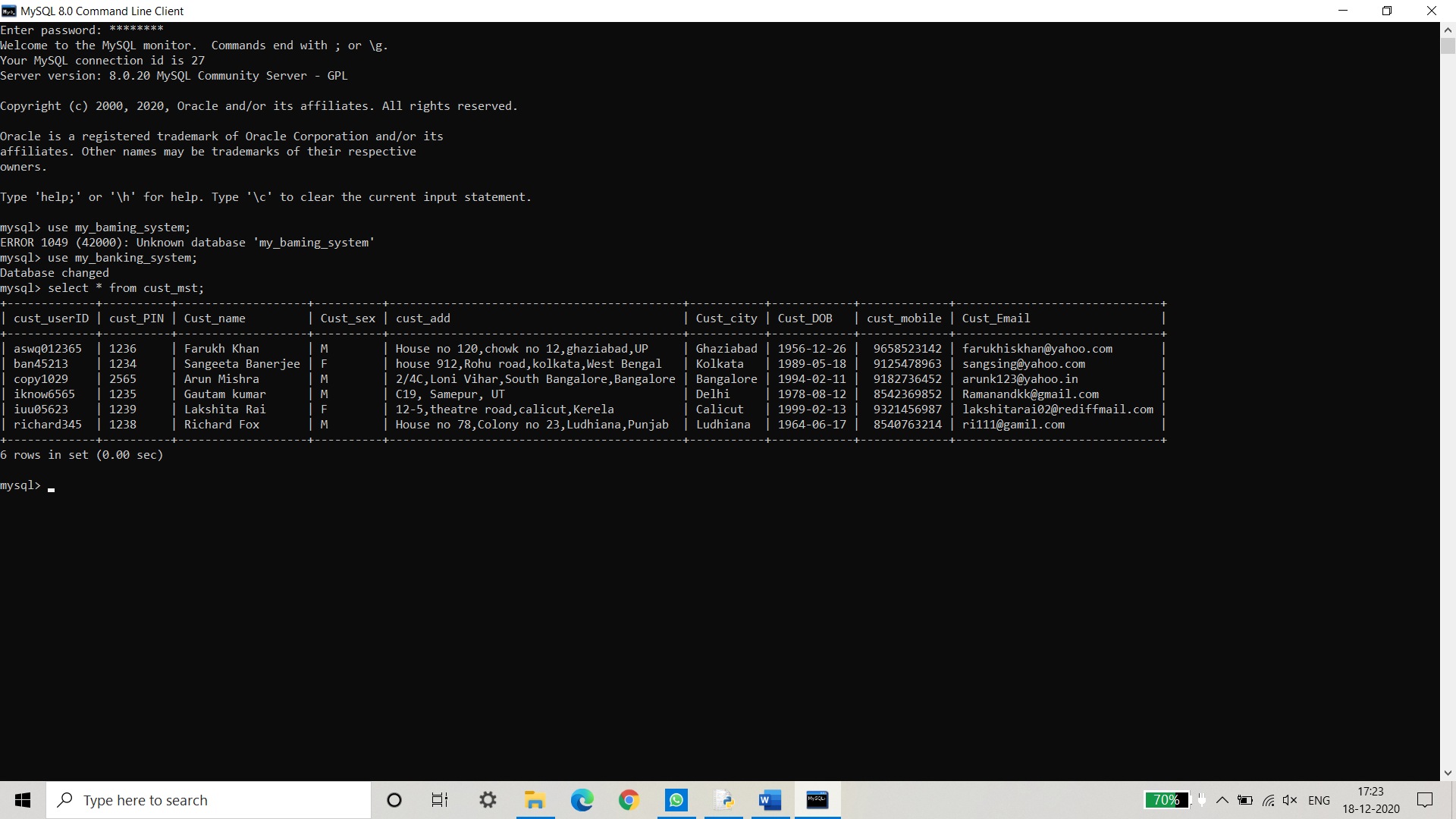
18.Bank Master table



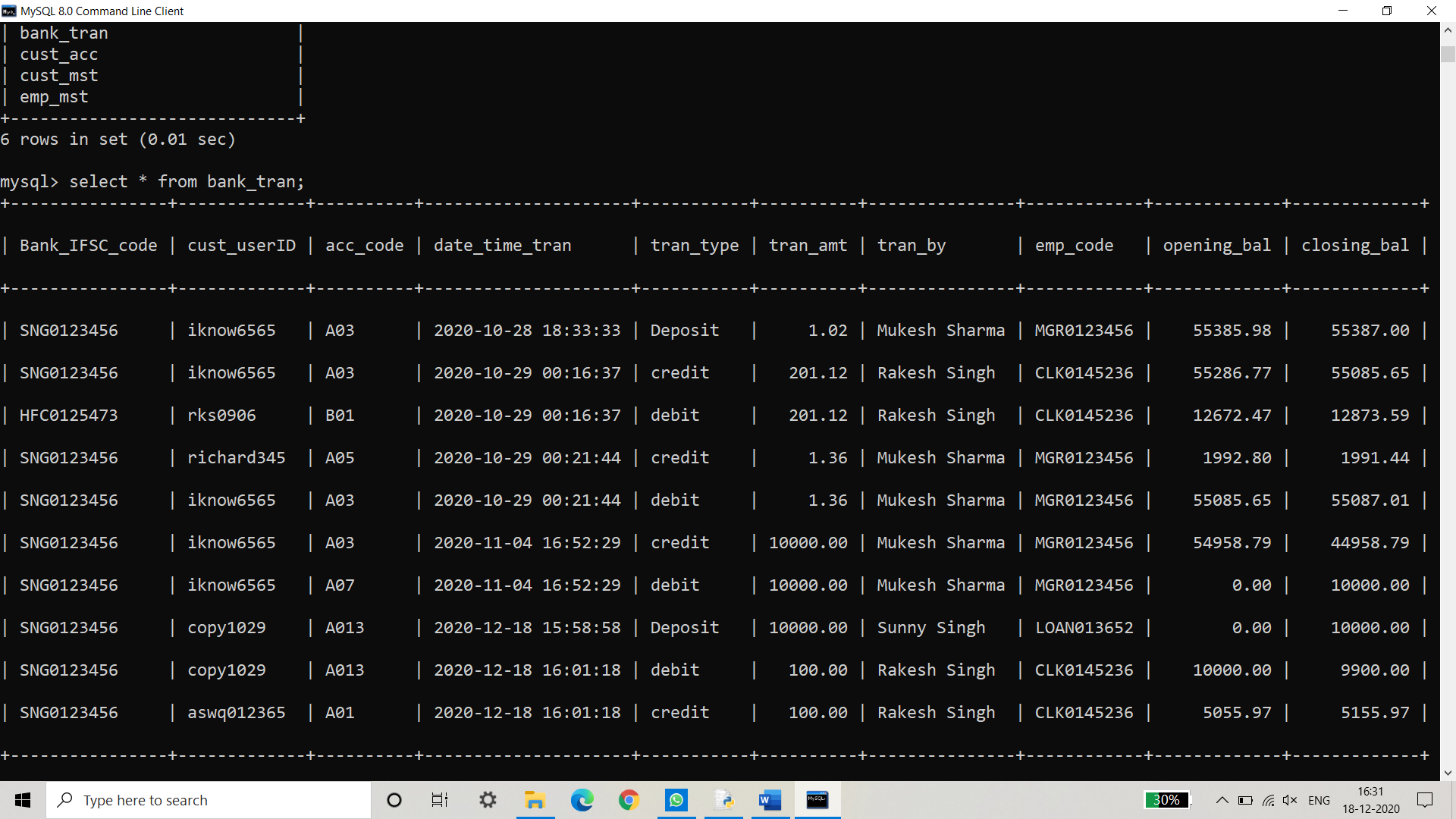


19.Customer master table

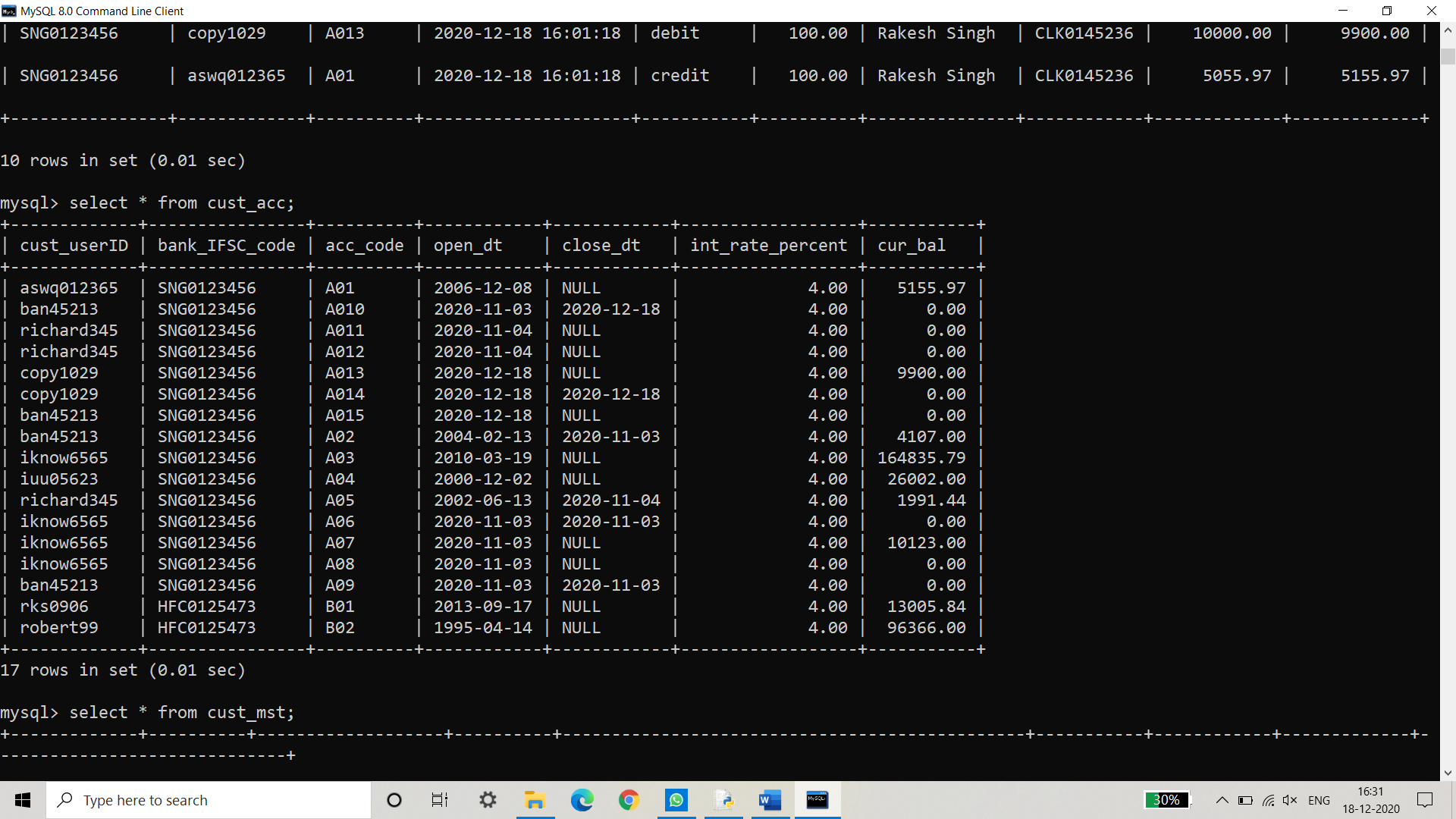




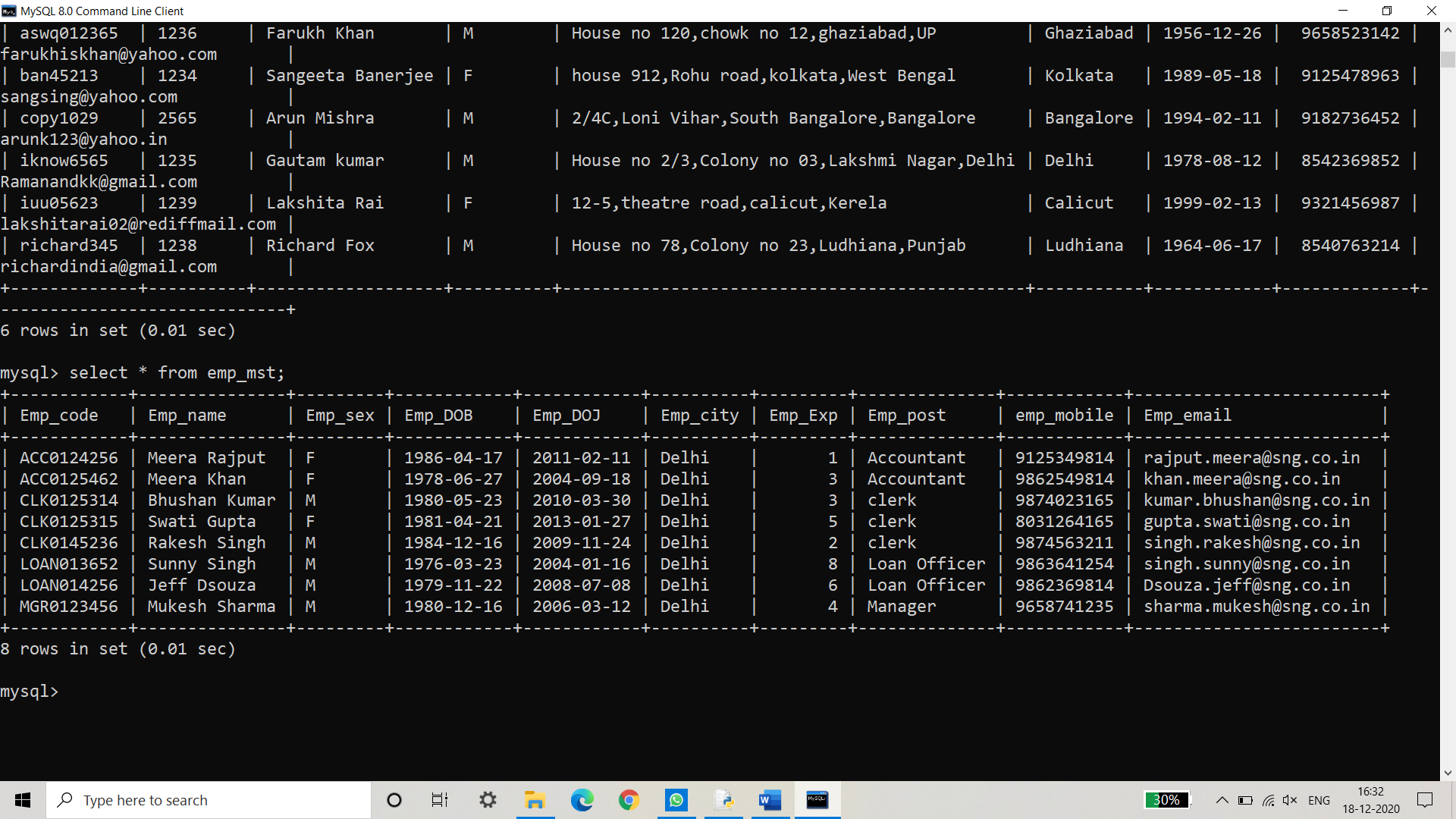
20.Bank Transaction table

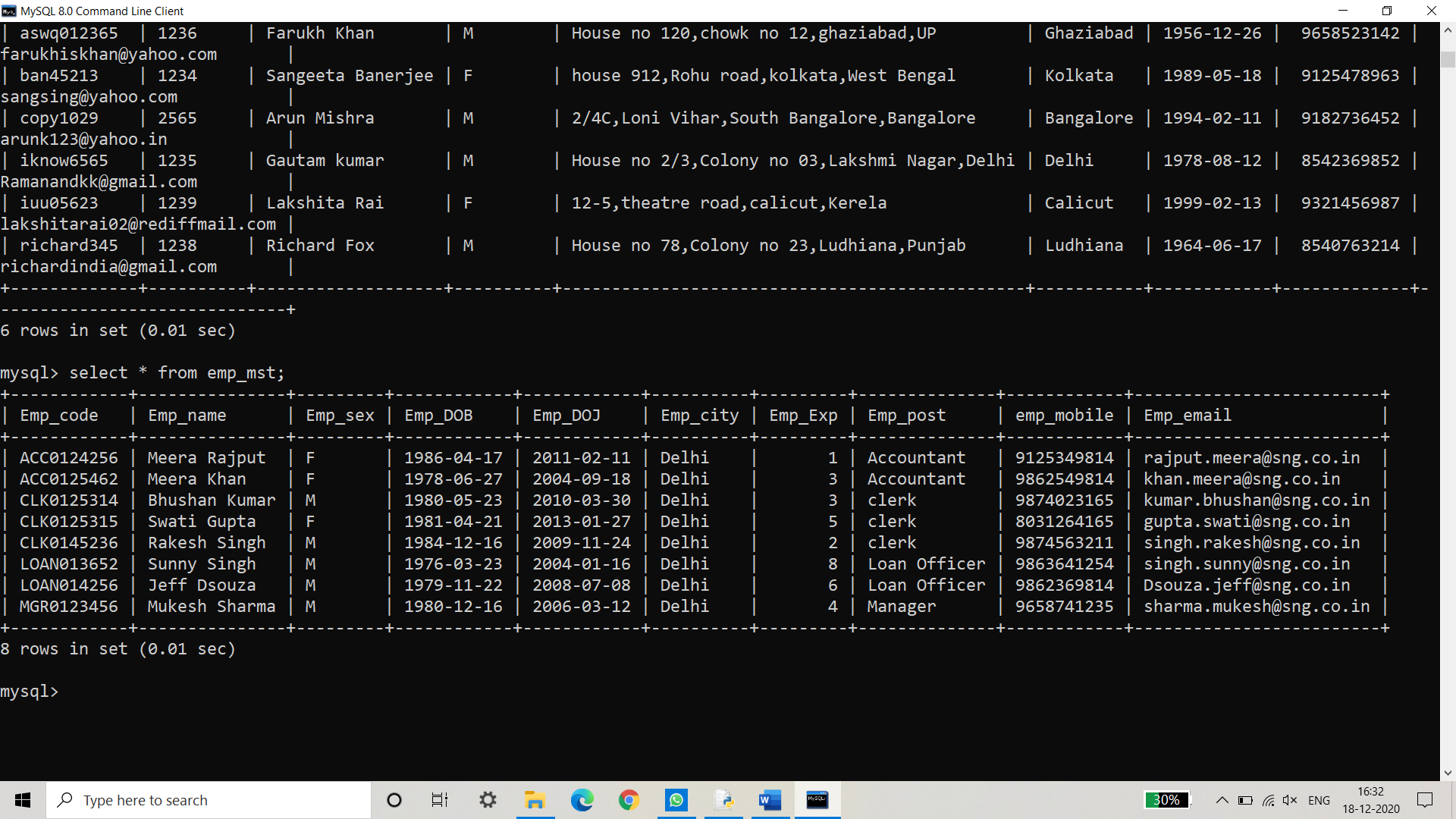


21.Customer account table



22.Employee table





Bibliography

Source of information: -

* http://geeksforgeeks.org/
* Computer Science Sumita Arora
* Computer Science Preeti Arora

Thank You