

PROGRAM CODE

```
from Tkinter import *
import pickle
import os
import random
import time
class customer:
    L=[]
    def __init__(self):
        self.gender="n/a"
        self.password="***#"
        self.accno="xyzw"
        self.acctype="n/a"
        self.name="n/a"
        self.pobox=0
        self.income=0
        self.id="n/a"
        self.dob="00/00/0000"
        self.address="n/a"
        self.d="n/a"
        self.workplace="n/a"
        self.phno="n/a"
        self.balance=0
        self.amt=0
    def user_pass(self):
        c=1
        while c==1:
            try:
                ch1=Exception("Password you have
entered should contain only digits")
                ch2=Exception("Password should have a
maximum lenght of 6 characters and a minimum of 4")
                ch3=Exception("Username
should contain only alphabets")
                u=raw_input("Enter the USER NAME: ")
                print
                "-----"
                print "password contains only digits"
```

```

        print "-----"
        pas=raw_input("Enter the password: ")
        if len(pas)>6 or len(pas)<4:
            raise ch2
        for i in pas:
            if i.isalpha():
                raise ch1
        for i in u:
            if i.isdigit():
                raise ch3
    except Exception,ch1:
        print "wait... "
        time.sleep(1)
        print ch1.message
    except Exception,ch2:
        print "wait... "
        time.sleep(1)
        print ch2.message
    except Exception,ch3:
        print "wait... "
        time.sleep(1)
        print ch3.message
    else:
        self.user=u
        ch4=Exception("Passwrod does not
match")
    try:
        cpd=raw_input("Enter password
for confirmation: ")
        time.sleep(1)
        if cpd==pas:
            print
            "~~~~~"
            print "| Successfully logged
in | "
            print
            "~~~~~"
            self.password=pas

```

```

        c=0
        time.sleep(1)
    else:
        raise ch4
except Exception, ch4:
    print "ERROR"

    print ch4.message

def getdata(self):
    print "THE FOLLOWING INFO IS MANDATORY"
    self.name=raw_input("Name of the new a/c
holder: ")
    self.gender=raw_input("Gender: ")
    self.address=raw_input("Address: ")
    self.accno=random.randint(1111,9999)

    try:
        self.phno=input("Phone number: ")
        self.pobox=input("P.O.Box number: ")
    except:
        self.phno=input("Phone number: ")
        self.pobox=input("P.O.Box number: ")
    self.user_pass()
    time.sleep(1)
    print " Please note your a/c no.
here",self.accno
    print " Dear'",self.name ,"'we require more
details for the creation of your a/c. Do you wish to add
your details now or later?"
    dbl=raw_input("Yes/No:
") if dbl=="yes":
        self.d=raw_input("Designation: ")
        self.workplace=raw_input("Place of work:
")
        self.id=raw_input("Email ID: ")
        self.dob=raw_input("DOB dd/mm/yyyy/ -")
    else:
        pass
print "~~~~~"
print "| LOADING BANK DETAILS... |"
print "~~~~~"

```

```

        time.sleep(2)
        print
"~~~~~"
"~~~~~"
        print " At MAZE BANK we provide 3 different
types of debit cards based on the income of the account
holder:-"
        time.sleep(1)
        print " A Silver card provides 15% on
mobile bills"
        time.sleep(1)
        print " A Gold card buys you movie
tickets at HALF PRICE \n Also 20% on mobile bills"

        time.sleep(1)
        print "           A Platinum card provides 25% on
Air tickets. \n           Also 30% on mobile bills!"

        print
"~~~~~"
"~~~~~"
        print
        time.sleep(1)
        print "~~~~~"
        print " |Card      | Minimum-Income |"
        print "~~~~~"
        print " |silver    |      10000   |"
        print " |gold     |      25000   |"
        print " |platinum |      40000   |"
        print
"~~~~~" s=1
        print "Dear customer, your balance is
3/4th your income"
        while s==1:
            print
            self.income=input("Enter your income:
") if self.income<=25000:
                self.acctype="Silver"
                self.balance=self.income*0.75
            elif self.income<=40000:
                self.acctype="Gold"
                self.balance=self.income*0.75

```

```

        elif self.income>40000:
            self.acctype="Platinum"
            self.balance=self.income*0.75
            print "          your a/c is",self.acctype
            print "          your balance
is",self.balance
            s=input("Do you wish to change ur income?
You can change it later \nEnter 1 for yes \nEnter 0 for
no: ")

```



```

def update(self):
    print
"^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^"
"^^^^^^^^^^^^^^^^^^^^^^^^"
    print "Please note that Account type,
Account number and balance cannot be updated."
    print "Only personal details can be updated"
    print "-----"
-----"

```



```

self.acctype,self.balance,self.accno=c.acctype,c.balance,
c.accno
    ch=raw_input("Do you want to change the name?")
    if ch.lower()=='y' or ch.lower()=='yes':
        self.name=raw_input("Name:")
    else:
        self.name=c.name
    ch=raw_input("Do you want to change the
income?:")
    if ch.lower()=='y' or ch.lower()=='yes':
        self.income=raw_input("Income:")
    else:
        self.income=c.income
    ch=raw_input("Do you want to change the email
id?")
    if ch.lower()=='y' or ch.lower()=='yes':
        self.id=raw_input("Email:")
    else:
        self.id=c.id

```

```

        ch=raw_input("Do you want to change the date of
birth?")
        if ch.lower() == "y" or ch.lower() == "yes":
            self.dob=raw_input("Date of
birth(dd/mm/yyyy:")
        else:
            self.dob=c.dob
        ch=raw_input("Do you want to change residence
address?")
        if ch.lower() == "y" or ch.lower() == "yes":
            self.address=raw_input("Residence
address:")
        else:
            self.address=c.address
        ch=raw_input("Do you want to change the
designation?")
        if ch.lower() == "y" or ch.lower() == "yes":
            self.d=raw_input("Designation:")
        else:
            self.d=c.d
        ch=raw_input("Do you want to change
the workplace name?")
        if ch.lower() == "y" or ch.lower() == "yes":
            self.workplace=raw_input("workplace
Name:")
        else:
            self.workplace=c.workplace
        ch=raw_input("Do you want to change the mobile
number?")
        if ch.lower() == "y" or ch.lower() == "yes":
            self.phno=raw_input("Mobile number:")
        else:
            self.phno=c.phno
        self.password=c.password
        self.gender=c.gende
    r def display(self):
        root=Tk()
        root.geometry('500x500')
        Label(root,fg="red", text="Name of the A/C
holder: ").grid(row=0,column=0)

```

```

        Label(root,fg="blue",
text=self.name).grid(row=0,column=2)
        Label(root,fg="red", text="Account number:
").grid(row=1,column=0)
        Label(root,fg="blue",
text=self.accno).grid(row=1,column=2)
        Label(root,fg="red", text="Balance in the A/C:
").grid(row=2,column=0)
        Label(root,fg="blue",
text=self.balance).grid(row=2,column=2)
        Label(root,fg="red", text="Phone
number: ").grid(row=8,column=0)
        Label(root,fg="blue",
text=self.phno).grid(row=8,column=2)
        Label(root,fg="red",
text="Gender ").grid(row=9,column=0)
        Label(root,fg="blue",
text=self.gender).grid(row=9,column=2)
        #####
        Label(root,fg="red", text="Email id:
").grid(row=3,column=0)
        Label(root,fg="blue",
text=self.id).grid(row=3,column=2)
        Label(root,fg="red", text="Account
type: ").grid(row=4,column=0)
        Label(root,fg="blue",
text=self.acctype).grid(row=4,column=2)
        Label(root,fg="red", text="Date of birth:
").grid(row=5,column=0)
        Label(root,fg="blue",
text=self.dob).grid(row=5,column=2)
)
        Label(root,fg="red", text="Income of the
A/C holder: ").grid(row=6,column=0)
        Label(root,fg="blue",
text=self.income).grid(row=6,column=2)
        Label(root,fg="red",
text="Address: ").grid(row=7,column=0)
        Label(root,fg="blue",
text=self.address).grid(row=7,column=2)
        Label(root,fg="red", text="Designation
: ").grid(row=10,column=0)

```

```

        Label(root,fg="blue",
text=self.d).grid(row=10,column=2)

root.mainloop()

class ccp(customer):
    def MobileBill(self):
        b=input('Enter Your Mobile Bill Amount =>
') if self.income < 25000:
            print"Silver card benefit: 15 percent off
"
            discount = (15/100.0)*
            b newb=b-discount
        elif self.income >= 25000 and self.income <=
40000:
            print"Gold card benefit: 20 percent off "
            discount = (20/100.0)*b
            newb=b-discount
        elif self.income > 40000:
            print"Platinum card benefit: 30 percent
off "
            discount = (30/100.0)*b
            newb=b-discount
        print newb
        time.sleep(2)
    def MovieTicket(self):
        amt=0
        k=0
        global y
        while k==0:
            try:
                ch6=Exception("maximum number of
seats is 4")
                n=input("Enter no of seats => ")
                if n>4:

```

```

                    raise ch6
            else:
                k=1
        except Exception:
            print ch6.message
    print "please note the movie ticket
prices" print "~~~~~"
    print "    vip      AED 500"
    print "Standard    AED 350"
    print "~~~~~"
seat=raw_input("Enter Type as standard or vip
=> ")
for i in range (n):
    if seat == "standard":
        amt=amt+350
    elif seat == "vip":
        amt=amt+500
    print "Total amount",amt

if y=="gold":
    print"Gold card benefit: 40 percent off "
    discount=(40/100.0)*amt newamt=amt-
    discount
elif y=="platinum":
    print"Platinum card benefit: 50 percent
off "
    discount=(50/100.0)*amt
    newamt=amt-discount
    print "thankyou for using MAZE bank for making
tansactions!"
    print "amount you have to pay after
discount is:",newamt
    time.sleep(2)
def AirTicket(self):
    try:
        ch7=Exception("maximum number of seats is
5")
        n=input("Enter no of seats => ")
        if n>5:

```

```

        raise ch7
    except Exception:
        print ch7.message
    amt=0
    print "please note the fares here"
    print "~~~~~"
    print " Economy      AED 2400"
    print " Business     AED 5600"
    print " Firstclass   AED 10000"
    print "~~~~~"

    Class=raw_input("Enter Class as Economy or
Business or Firstclass [case sensitive]=> ")
    for i in range (n):

        if Class == "Economy":
            amt=amt+2400
        elif Class == "Business":
            amt=amt+5600
        elif Class == "Firstclass":
            amt = amt+10000

        print "Platinum card Benefit: 15 percent off "
        discount=0.15*amt
        print "discount provided to you",discount
        finalamt=amt-discount
        print "thankyou for using MAZE bank for making
tansactions!"

        print "amount you have to pay
is:",finalamt time.sleep(2)

class loans:

    def carloan(self):
        self.loanamt=input("enter loan amount: ")
        n=input("no. of years: ")
        self.intrest=0.08*self.loanamt*n
        t=self.loanamt+self.intrest
        print "total intrest",self.intrest

```

```

        print "total amount to be repaid",t
        o=raw_input("Will you pay every month or every
3 months (enter 1 or 3): ")
        if o=="1":
            print"you'll have to pay:",t/(12*n),"every
month"
        elif o=="3":
            print"you'll have to
pay:",3*t/(12*n),"every three months"
        else:
            print"INVALID OPTION"

    def GoldLoan(self):
        print "GOLD LOAN"
        print "|Minimum deposit amount is 10 grams|"
        self.amt=input("Enter the amount of Gold in
grams => ")
        self.carat=input("Enter 22 Carat or 24 carat
=>")
        print
        print "|Maximum Duration of Repayment is 5
years|"
        print
        n=input("no of years => ")
        if self.carat==22:
            self.loanamt=self.amt*30000
            self.intrest=0.02*self.loanamt*n
            t=self.loanamt+self.intrest
            print "total intrest",self.intrest print
            "total amount to be repaid",t
            o=raw_input("Will you pay every month or
every 3 months (enter 1 or 3): ")
            if o=="1":
                print"you'll have to
pay:",t/(12*n),"every month"
            elif o=="3":
                print"you'll have to
pay:",3*t/(12*n),"every three months"
            else:
                print"INVALID
OPTION" elif self.carat==24:

```

```

        self.loanamt=self.amt*32000
        self.intrest=0.03*self.loanamt*n
        t=self.loanamt+self.intrest
        print "total intrest",self.intrest print
        "total amount to be repaid",t
        o=raw_input("Will you pay every month or
every 3 months(enter 1 or 3):")
        if o=="1":
            print"you'll have to
pay:",t/(12*n),"every month"
        elif o=="3":
            print"you'll have to
pay:",3*t/(12*n),"every three months"
        else:
            print"INVALID
OPTION" def HomeLoan(self):
        print "HOME LOAN"
        print "|Collateral Documents to be submitted at
Counter |"
        self.amt=input("Enter the Loan Amount =>
") print
        print "|Maximum Duration of Repayment is 10
years|"
        print
        n=input("no of years => ")
        ch=raw_input("Have Documents been Approved
=> Yes/No => ")
        if ch=='yes'or ch=='Yes':
            self.loanamt=self.amt
            self.intrest=1.5*self.loanamt*n
            t=self.loanamt+self.intrest
            print "total intrest",self.intrest print
            "total amount to be repaid",t
            o=raw_input("Will you pay every month or
every 3 months(enter 1 or 3):")
            if o=="1":
                print"you'll have
to pay:",t/(12*n),"every month"
            elif o=="3":
                print"you'll have to
pay:",3*t/(12*n),"every three months"

```

```

        else:
            print "INVALID OPTION"
    else:
        print "Since You Document Are not Approved,
Loan Cannot be Sanctioned"

nx=0
w=0
def newaccount():
    c=customer()
    c.getdata()

    f=open("Account.dat","ab")
    pickle.dump(c,f)
    f.close
    global
    nx nx=1

def existing():
    m=input("Enter Account number:
") f=open("Account.dat","rb")
try:
    while True:
        c=pickle.load(f)
        if m==c.accno:
            w=input("enter the password")
            if c.password==str(w):
                print "LOGGED IN",c.name.upper()
                print "|Hello|Welcom|"
                time.sleep(1)
                global
                nx nx=1

except EOFError:
#####
    if nx==0:
        print "WRONG ACCOUNT NUMBER"

```

```

f.close()

def admin():
    admin="man"
    passw=raw_input("Enter ADMIN password: ")
    f=open("Account.dat","rb")
    if passw==admin:

        print"~~~~~"
        print"~~~~~"
        print"|      Account No.          NAME"
        print"Password      Balance |"
        print"~~~~~"
        print"~~~~~"
        try:
            while True:
                c=pickle.load(f)
                print"%16d%16s%16s%16d"%(c.accno,c.name,c.password,c.bala
                nce)

            print"~~~~~"
            print"~~~~~"

        except EOFError:
            pass

    f.close()
    print "~~~~~"
    print "| return to menu [existing | new customer]?"
    print "~~~~~"
    opn1=raw_input("yes/no: ")
    if opn1=="yes":
        print "~~~~~"
        print "| 1.existing      |"
        print "| 2.new customer |"
        print "~~~~~"

```

```

        opn2=raw_input("enter the option number:
") if opn2=="1":
    existing()
else:
    newaccount()
else:
    print "Bye Admin"

print "~~~~~"
print "| WELCOME TO MAZE BANK! |"
print "~~~~~"
print "| 1:Existing customer |"
print "| 2:New customer |"
print "| 3:Admin |"
print "~~~~~"

time.sleep(1)
opn=input("Enter option number:
") print "~~~~~"
print "|LOADING... |"
print "~~~~~"
time.sleep(2)
if opn==1:
    existing()
elif opn==2:
    newaccount()
elif opn==3:
    admin()

while nx==1:
    print
"~~~~~"
    print "| Choose which operation to perform:
| "

```

```
print
```

```
"~~~~~" "
```

```

        print " |           1:BanK services
| "
        print " |           2:Credit card perks
| "
        print " |           3:Quit
| "
        print

"~~~~~"

d=input("Enter option number: ")

if d==2:
    print "~~~~~"
    print " | LOADING... | "
    print "~~~~~"
    time.sleep(2)
    y=raw_input("Enter the A/C type,
ie silver/gold/platinum: ")
    if y=="silver":
        c=ccp()
        c.MobileBill()
    elif y=="gold":
        c1=ccp()
        c1.MovieTicket()
    elif y=="platinum":
        c2=ccp()
        c2.AirTicket()
elif d==3:
    quit()

elif d==1:
    print "~~~~~"
    print " | LOADING... | "
    print "~~~~~"
    print
    time.sleep(2)
    print "Your Transaction Options Are:"
    print "~~~~~"
    print "1.1) Deposit Money"

```

```

        print "1.2) Withdraw Money"
        print "1.3) Car loan"
        print "1.4) Gold Loan"
        print "1.5) Home Loan"
        print "1.6) Update personal info"
        print "1.7) Display personal info"
        print "~~~~~"
        ch="yes"
        while ch=="yes":
            o=input('Enter choice number: ')
            print "~~~~~"
            print "|LOADING... |"
            print "~~~~~"
            time.sleep(2)
            if o==1.1:
                x=input("Enter Account number:")
                f=open("Account.dat","rb")
                g=open("temp.dat","wb")
                try:
                    while True:
                        c=pickle.load(f)
                        if x==c.accno:
                            a=input("Enter amt to
deposit")
                            c.balance=c.balance+a
                print"balance:",c.balance
                            pickle.dump(c,g)
                except EOFError:
                    pass
                    f.close()
                    g.close()
                    os.remove("Account.dat")
                    os.rename("temp.dat","Account.dat")

            elif o==1.2:
                x=input("Enter Account number")
                f=open("Account.dat","rb")
                g=open("temp.dat","wb")
                flag=0

```

```

try:
    while True:
        c=pickle.load(f)
        if c.accno==x:
            b=input("Enter amt to
withdraw")
            if b>c.balance:
                print
                "Transaction not possible as only",c.balance,"AED left in
account"
            else:
                c.balance=c.balance-b
                print"balance:",c.balance
                ######
                pickle.dump(c,g)
                flag=1
            else:
                pickle.dump(c,g)
        except EOFError:
            pass
        if flag==0:
            print "Not a valid account
number"
        f.close()
        g.close()
        os.remove("Account.dat")
        os.rename("temp.dat","Account.dat")
    elif o==1.3:
        x=input("enter account number:
") f=open("Account.dat","rb")
        g=open("temp.dat","wb")
        try:
            while True:
                c=pickle.load(f)
                if c.accno==x:
                    c=loans()
                    c.carloan()
                    pickle.dump(c,g)
                else:

```

```

                print "not found"
        except EOFError:
                pass
        f.close()
        g.close()

elif o==1.4:
        x=input("enter account number:")
        f=open("Account.dat","rb")
        g=open("temp.dat","wb")
        c=1
        try:
                while True:
                        c=pickle.load(f)
                        if c.accno==x:
                                c.loans()
                                c.GoldLoan()
                                pickle.dump(c,g)
                                c=0
                        else:
                                pass
        except EOFError:
                pass
        if c==0:
                pass
        else:
                print "Account not
found" f.close()
        g.close()

elif o==1.5:
        x=input("enter account number:")
        f=open("Account.dat","rb")
        g=open("temp.dat","wb")
        c=1

        try:
                while True:
                        c=pickle.load(f)
                        if c.accno==x:

```

```

c=loans()
c.HomeLoan()
pickle.dump(c,g)
c=0
else:
    pass
except EOFError:
    pass
if c==0:
    pass
else:
    print "Account not
found" f.close()
g.close()
elif o==1.6:
    x=input("Enter Account number:
") f=open("Account.dat","rb")
g=open("temp.dat","wb")
c=1
try:
    while True:
        c=pickle.load(f)
        if c.accno==x:
            d=customer()
            d.update()
            pickle.dump(d,g)
            c=0
        else:
            pickle.dump(c,g)
except EOFError:
    pass
if c==0:
    print "Information updated
successfully!"
else:
    print "Account not
found" f.close()
g.close()
os.remove("Account.dat")
os.rename("temp.dat","Account.dat")

```

```

        elif o==1.7:
            f=open("Account.dat","rb")
            j=input("Enter account
no:") try:
                while True:
                    x=pickle.load(f)
                    x.display()
            except EOFError:
                pass

        else:
            print "Invalid option"
            print "If you wish to move to credit card
perks, pls enter (menu)"
            ch=raw_input("do u want to continue?
[yes/no/menu] ")

            if ch=="no":
                print "Thankyou for using MAZE BANK"
                print"
~~~~~"
                print" | \     /|    /\     -----.
-----"
                print" | \     / |    / \     /
| "
                print" |     \/   | /----\     /
| ----"
                print" |         | /       \     /
| "
                print" |         | /           \     -----
`-----"

print"~~~~~"
        elif ch=="ccp":
            break
        pass

```

```

  ⚡ IDLE File Edit Shell Debug Options Window Help *Python 2.7.12 Shell*
  Python 2.7.12 (v2.7.12:d33e0cf91556, Jun 26 2016, 12:03:00)
  [GCC 4.2.1 (Apple Inc. build 5577)] on darwin
  Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/anjumehta/Downloads/MYBANKBESTEST-3.py =====
| WELCOME TO MAZE BANK! |
|-----|
| 1:Existing customer |
| 2>New customer |
| 3:Admin |
Enter option number: 2
|LOADING... |
THE FOLLOWING INFO IS MANDATORY
Name of the new a/c holder: Andy Gray
Gender: male
Address: 24 jump street
Phone number: 456789
P.O.Box number: 234
Enter the USER NAME: andy
password contains only digits
Enter the password: 12345
Enter password for confirmation: 12345
|Successfully logged in |
Please note your a/c no. here 2483
Dear' Andy Gray 'we require more details for the creation of your a/c. Do you wish to add your d
etails now or later?
Ln: 29 Col: 0
  ⚡ Python 2.7.12 Shell*
Please note your a/c no. here 2483
Dear' Andy Gray 'we require more details for the creation of your a/c. Do you wish to add your d
etails now or later?
Yes/No: yes
Designation: Footballer
Place of work: Manchester, England
Email ID: andygray@mufc.net
DOB dd/mm/yyyy/ -14/01/1989
|LOADING BANK DETAILS... |

At MAZE BANK we provide 3 different types of debit cards based on the income of the account hold
er:-
A Silver card provides 15% on mobile bills
A Gold card buys you movie tickets at HALF PRICE
Also 20% on mobile bills
A Platinum card provides 25% on Air tickets.
Also 30% on mobile bills!
|Card | Minimum-Income |
|silver | 10000 |
|gold | 25000 |
|platinum | 40000 |
Dear customer, your balance is 3/4th your income
Enter your income:
Ln: 60 Col: 19

```

```

*Python 2.7.12 Shell*
Enter your income: 26750
    your a/c is Gold
    your balance is 20062.5
Do you wish to change ur income? You can change it later
Enter 1 for yes
Enter 0 for no: 0
Choose which operation to perform:
1:BanK services
2:Credit card perks
3:Quit
Enter option number: 1
|LOADING... |

Your Transaction Options Are:
1.1) Deposit Money
1.2) Withdraw Money
1.3) Car loan
1.4) Gold Loan
1.5) Home Loan
1.6) Update personal info
1.7) Display personal info
Enter choice number: 1.1
|LOADING... |
Enter Account number: 2483
Ln: 92 Col: 26
*Python 2.7.12 Shell*
|LOADING... ||
Enter Account number: 2483
Enter amt to deposit14575
balance: 34637.5
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.2
|LOADING... |
Enter Account number2483
Enter amt to withdraw12500
balance: 22137.5
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.3
|LOADING... |
enter account number: 2483
enter loan amount: 12000
no. of years: 2
total intrest 1920.0
total amount to be repaid 13920.0
Will you pay every month or every 3 months(enter 1 or 3): 1
you'll have to pay: 580.0 every month
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.4
|LOADING... |
enter account number: 2483
Ln: 90 Col: 15

```

```
enter account number: 2483
*Python 2.7.12 Shell*
GOLD LOAN
|Minimum deposit amount is 10 grams|
Enter the amount of Gold in grams => 250
Enter 22 Carat or 24 carat =>22

|Maximum Duration of Repayment is 5 years|

no of years => 4
total intrest 600000.0
total amount to be repaid 8100000.0
Will you pay every month or every 3 months(enter 1 or 3):1
you'll have to pay: 168750.0 every month
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.5
-----
|LOADING... |

enter account number: 2483
HOME LOAN
|Collateral Documents to be submitted at Counter |
Enter the Loan Amount => 23000

|Maximum Duration of Repayment is 10 years|

no of years => 4
Have Documents been Approved => Yes/No => yes
total intrest 138000.0
total amount to be repaid 161000.0
Will you pay every month or every 3 months(enter 1 or 3):1
you'll have to pay: 3354.16666667 every month
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
```

```
*Python 2.7.12 Shell*
enter account number: 2483
GOLD LOAN
|Minimum deposit amount is 10 grams|
Enter the amount of Gold in grams => 250
Enter 22 Carat or 24 carat =>22

|Maximum Duration of Repayment is 5 years|

no of years => 4
total intrest 600000.0
total amount to be repaid 8100000.0
Will you pay every month or every 3 months(enter 1 or 3):1
you'll have to pay: 168750.0 every month
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.5
-----
|LOADING... |

enter account number: 2483
HOME LOAN
|Collateral Documents to be submitted at Counter |
Enter the Loan Amount => 23000

|Maximum Duration of Repayment is 10 years|

no of years => 4
Have Documents been Approved => Yes/No => yes
total intrest 138000.0
total amount to be repaid 161000.0
Will you pay every month or every 3 months(enter 1 or 3):1
you'll have to pay: 3354.16666667 every month
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
```

```
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.6
-----
|LOADING... |
-----
Enter Account number: 2483
=====
Please note that Account type, Account number and balance cannot be updated.
Only personal details can be updated
-----
Do you want to change the name?:no
Do you want to change the income?:no
Do you want to change the email id?:no
Do you want to change the date of birth?:yes
Date of birth(dd/mm/yyyy):04/10/1988
Do you want to change residence address?:no
Do you want to change the designation?:no
Do you want to change the workplace name?:no
Do you want to change the mobile number?:no
Information updated successfully!
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu] yes
Enter choice number: 1.7
-----
|LOADING... |
-----
Enter account no:2483
If you wish to move to credit card perks, pls enter (menu)
do u want to continue? [yes/no/menu]
```

Ln: 156 Col: 2

```
*Python 2.7.12 Shell*
-----
| Choose which operation to perform: |
-----
| 1:BanK services
| 2:Credit card perks
| 3:Quit
-----
Enter option number: 2
-----
|LOADING... |
-----
Enter the A/C type, ie silver/gold/platinum: gold
Enter no of seats => 2
please note the movie ticket prices
-----
    vip      AED 500
Standard   AED 350
-----
Enter Type as standard or vip ==> vip
Total amount 1000
Gold card benefit: 40 percent off
thankyou for using MAZE bank for making tansactions!
amount you have to pay after discount is: 600.0
-----
| Choose which operation to perform: |
-----
| 1:BanK services
| 2:Credit card perks
| 3:Quit
-----
Enter option number: 2
-----
|LOADING... |
-----
```

Ln: 184 Col: 25

```

*Python 2.7.12 Shell*
Enter option number: 2
|LOADING... |
Enter the A/C type, ie silver/gold/platinum: platinum
Enter no of seats => 4
please note the fares here
Economy    AED 2400
Business   AED 5600
Firstclass AED 10000
Enter Class as Economy or Business or Firstclass [case sensitive]> Firstclass
Platinum card Benefit: 15 percent off
discount provided to you 6000.0
thankyou for using MAZE bank for making tansactions!
amount you have to pay is: 34000.0
| Choose which operation to perform: |
1:BanK services
2:Credit card perks
3:Quit
Enter option number: 2
|LOADING... |
Enter the A/C type, ie silver/gold/platinum: silver
Enter Your Mobile Bill Amount => 2500
Silver card benefit: 15 percent off
2125.0
| Choose which operation to perform: |

```

Ln: 220 Col: 20

```

tk

Name of the A/C holder: Andy Gray
Account number: 2483
Balance in the A/C: 22137.5
Email id: andygray@mufc.net
Account type: Gold
Date of birth: 04/10/1988
Income of the A/C holder: 26750
Address: 24 jump street
Phone number: 456789
Gender: male
Designation : Footballer

```

```

*Python 2.7.12 Shell*
Economy AED 2400
Business AED 5600
Firstclass AED 10000
Enter Class as Economy or Business
Platinum card Benefit: 15 percent off
discount provided to you 6000.0
thankyou for using MAZE bank for making tansactions!
amount you have to pay is: 34000.0
Choose which operation to perform:
1:BanK services
2:Credit card perks
3:Quit
Enter option number: 2
LOADING...
Enter the A/C type, ie silver/gold/platinum: silver
Enter Your Mobile Bill Amount => 2500
Silver card benefit: 15 percent off
2125.0
Choose which operation to perform:
1:BanK services
2:Credit card perks
3:Quit
Enter option number: 3

```

Program closes at this point.

Admin: Can check all the accounts created. (here, only 1 account)

```

Python 2.7.12 (v2.7.12:d33e0cf91556, Jun 26 2016, 12:03:00)
[GCC 4.2.1 (Apple Inc. build 5577)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/anjumehta/Downloads/MYBANKBESTEST-3.py =====
WELCOME TO MAZE BANK!
1:Existing customer
2>New customer
3:Admin
Enter option number: 3
LOADING...
Enter ADMIN password: man
Account No.      NAME          Password        Balance |
2483            Andy Gray     12345           22137
| return to menu [existing | new customer]?
yes/no:
Bye Admin
>>> |

```

Signing in as existing customer.

```
*Python 2.7.12 Shell*
>>>
===== RESTART: /Users/anjumehta/Downloads/MYBANKBESTEST-3.py =====
-----| WELCOME TO MAZE BANK! |-----
| 1:Existing customer |
| 2>New customer |
| 3:Admin |
-----|
Enter option number: 3
-----| LOADING... |-
Enter ADMIN password: man
-----| Account No. NAME Password Balance |
| 2483 Andy Gray 12345 22137 |
-----|
| return to menu [existing | new customer]? |
-----|
yes/no: yes
-----| 1.existing |
| 2.new customer |
-----|
enter the option number: 1
Enter Account number: 2483
enter the password12345
LOGGED IN ANDY GRAY
Hello|Welcome|
-----| Choose which operation to perform: |
```