DAV CENTENARY PUBLIC SCHOOL, NARELA

PROJECT IN

COMPUTER SCIENCE

ON

"BANK MANAGEMENT SYSTEM"

Session 2020-21

PREPARED BY:-

ARPAN KUMAR
XII D
Roll No. __________

CERTIFICATE

This is to certify **ARPAN KUMAR** student of class XII-D, **D.A.V Centenary Public School, Narela** has successfully completed the Python project on the topic "**BANK MANAGMENT SYSTEM**" under the guidance of subject teacher, **Mrs.Anubha Khatri**.

Signature of Internal Examiner

Signature of Internal Examiner

ACKNOWLEDGEMENT

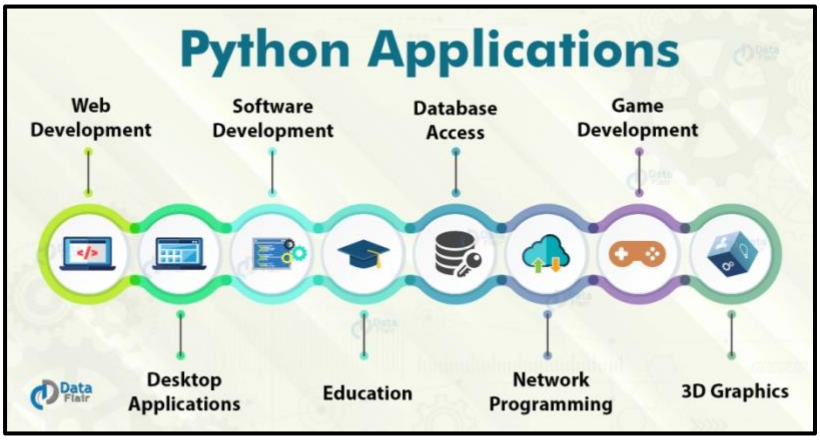
I would like to express my deepest sense of gratitude to my honoured teacher and mentor Ms.Anubha Khatri, Head of Department,IT Department, DAV Centenary Public School, Narela and, for her untiring guidance, constant supervision, enthusiastic encouragement, brilliant advice and an effective surveillance throughout the entire period of my project & thesis work. I greatly say thank you.

I also want to express my appreciation to my project partner Aditya Sharma who helped me in one way or another during the course of developing this project.

ARPAN KUMAR

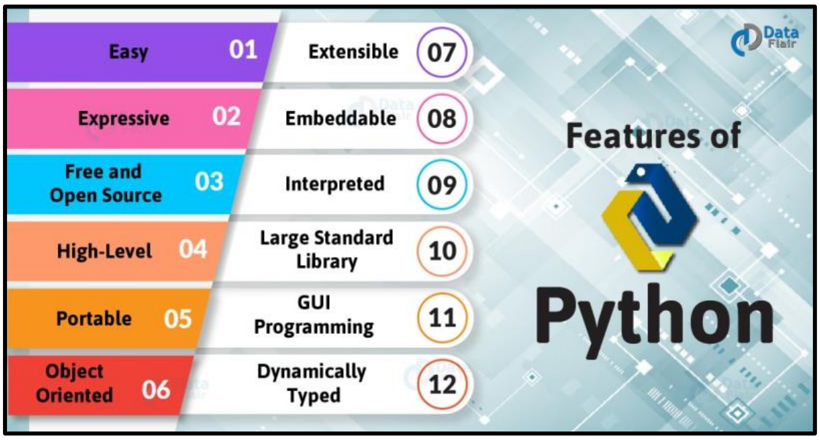
CONTENTS

ABOUT PYTHON



Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.



Python was created in the late 1980s, and first released in 1991, by Guido van Rossum as a successor to the ABC programming language. Python 2.0, released in 2000, introduced new features, such as list comprehensions, and a garbage collection system with reference counting, and was discontinued with version 2.7 in 2020.

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary-precision decimals manipulating regular expressions, and unit testing.

Python uses dynamic typing and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

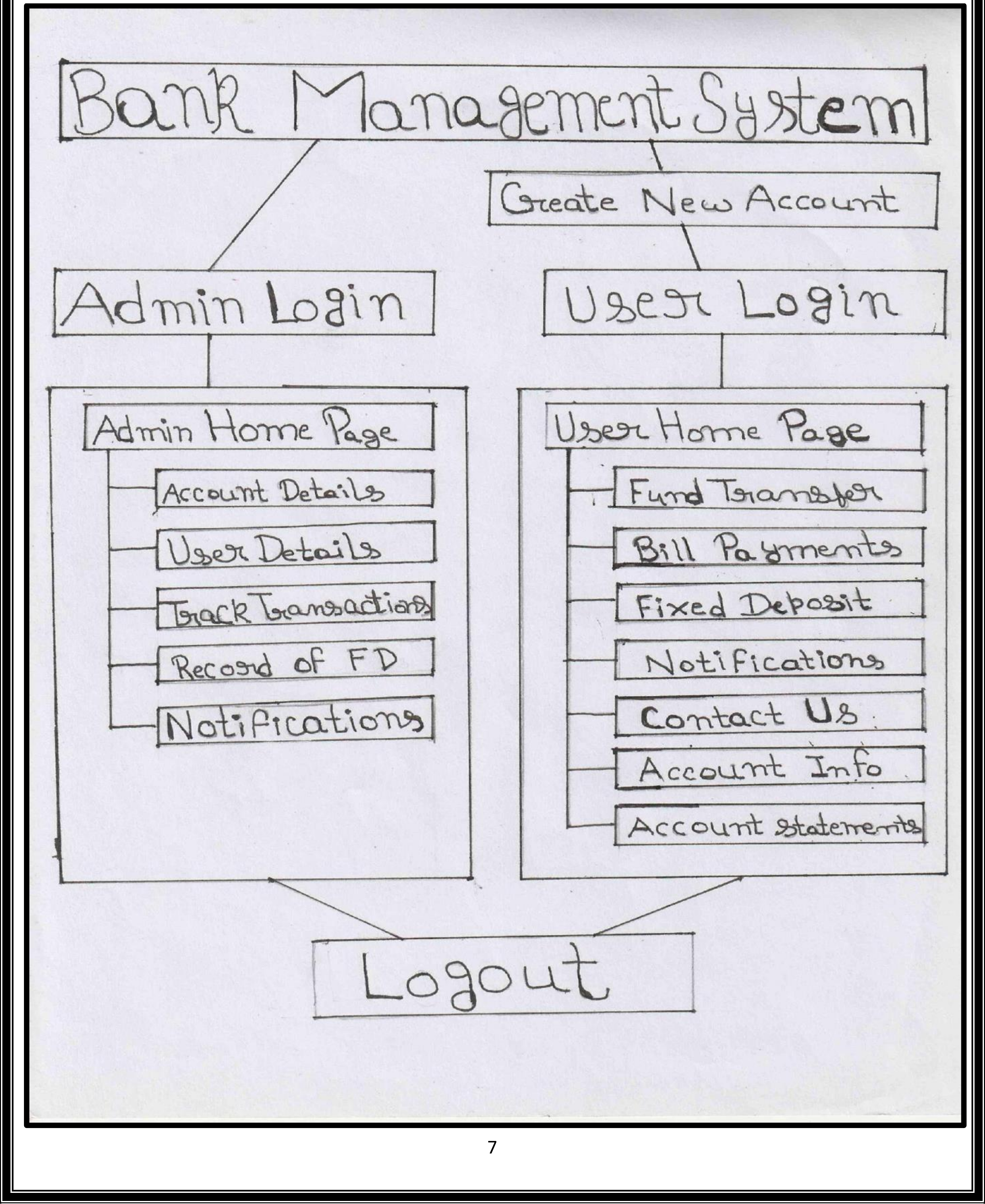
ABOUT THE PROJECT

This project “Bank Management System” enables the customers to perform the basic banking transactions by sitting at their office or at homes through PC or laptop. The system provides the access to the customer to create an account, the fund transfers from his account, also to view reports of all transactions. The customers can access their Account details and perform the transactions on account as per their requirements.

Now a days, managing a bank is an uninteresting job up to certain limit. So software that reduces the work is essential. Also today’s world is a genuine computer world and is getting faster and faster day-by-day. Thus, considering above necessities, the software for bank management has became necessary which would be useful in managing the bank more efficiently. All transactions are carried out online by transferring from accounts in the same Bank or international bank. This software is meant to overcome the drawbacks of the manual system. This software has been developed using the most powerful and secure backend MySQL database.

Bank Management System keeps the day by day tally record as a complete banking system. It can keep the information of Account type, account opening form,Fund transfers,Bill payments, Searching the particular account , Transaction reports, Fixed deposits.

PROJECT DESIGN



HARDWARE & SOFTWARE SPECIFICATIONS

Hardware Requirements :

Processor : Intel Pentium

Main Memory(RAM) : 256 MB

Cache Memory : 150 KB

Monitor : 15.6 inch Color Monitor

Keyboard : 108 Keys

Mouse : Optical Mouse

Hard Disk : 1TB

Software Requirements :

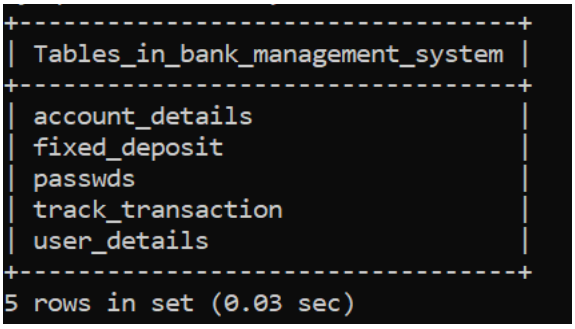
Front End/Language : Python 3.7.3

Back End/Database : MYSQL 8.0

Operating System : Windows 8

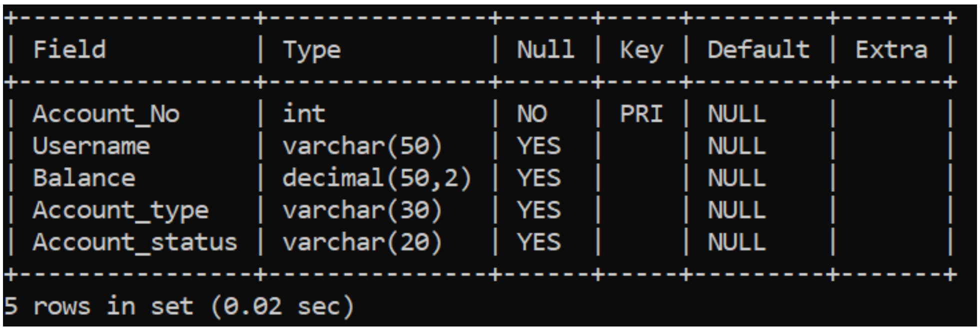
DATABASE DESIGN

Tables:-



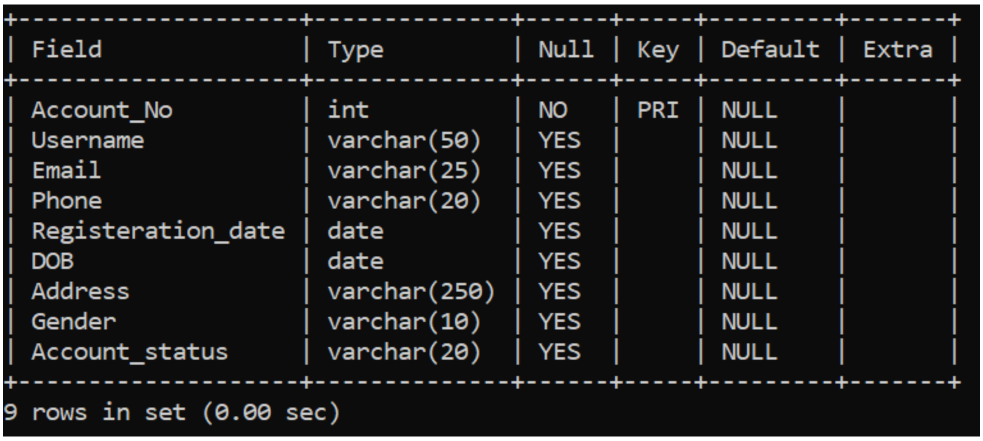
```
+--------------------------------+
| Tables_in_bank_management_system |
+--------------------------------+
| account_details                |
| fixed_deposit                  |
| passwds                        |
| track_transaction              |
| user_details                   |
+--------------------------------+
5 rows in set (0.03 sec)
```

1. Account Details



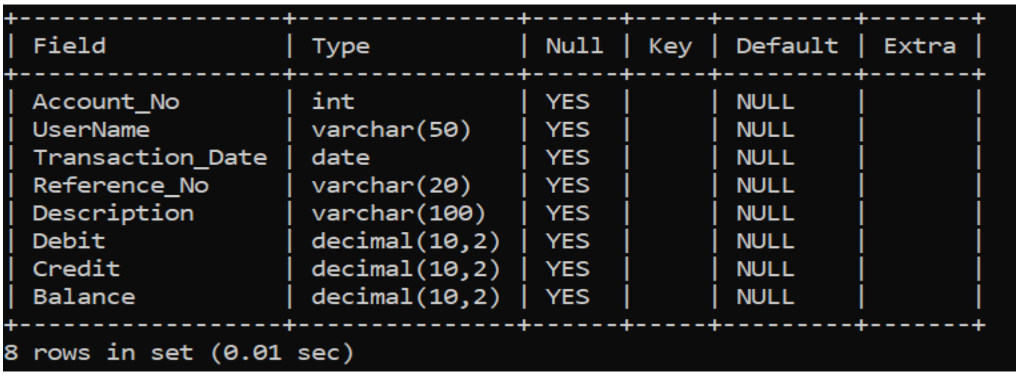
```
+----------------+---------------+------+-----+---------+-------+
| Field          | Type          | Null | Key | Default | Extra |
+----------------+---------------+------+-----+---------+-------+
| Account_No     | int           | NO   | PRI | NULL    |       |
| Username       | varchar(50)   | YES  |     | NULL    |       |
| Balance        | decimal(50,2) | YES  |     | NULL    |       |
| Account_type   | varchar(30)   | YES  |     | NULL    |       |
| Account_status | varchar(20)   | YES  |     | NULL    |       |
+----------------+---------------+------+-----+---------+-------+
5 rows in set (0.02 sec)
```

2. User Details



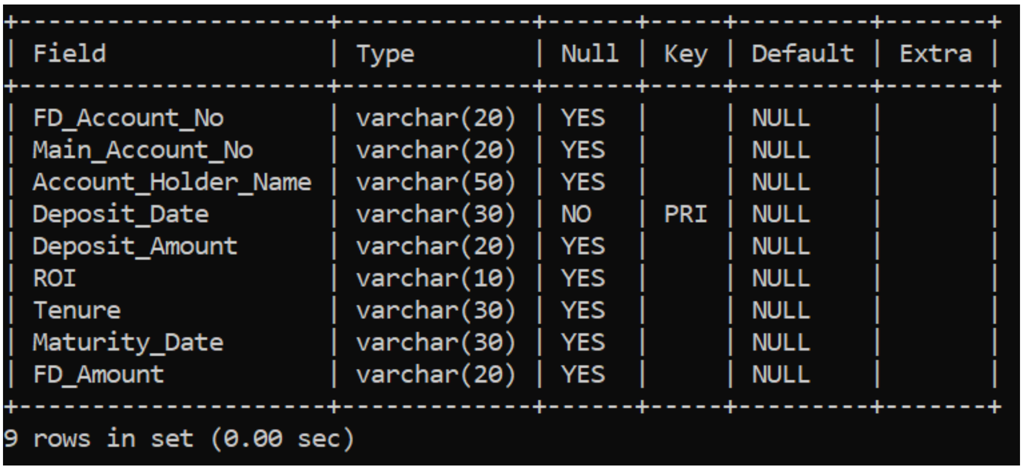
```
+--------------------+--------------+------+-----+---------+-------+
| Field              | Type         | Null | Key | Default | Extra |
+--------------------+--------------+------+-----+---------+-------+
| Account_No         | int          | NO   | PRI | NULL    |       |
| Username           | varchar(50)  | YES  |     | NULL    |       |
| Email              | varchar(25)  | YES  |     | NULL    |       |
| Phone              | varchar(20)  | YES  |     | NULL    |       |
| Registeration_date | date         | YES  |     | NULL    |       |
| DOB                | date         | YES  |     | NULL    |       |
| Address            | varchar(250) | YES  |     | NULL    |       |
| Gender             | varchar(10)  | YES  |     | NULL    |       |
| Account_status     | varchar(20)  | YES  |     | NULL    |       |
+--------------------+--------------+------+-----+---------+-------+
9 rows in set (0.00 sec)
```

3. Track Transaction



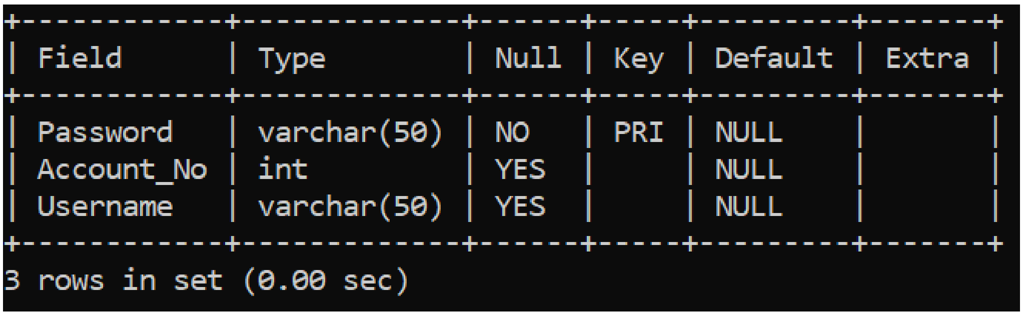
```
+------------------+---------------+------+-----+---------+-------+
| Field            | Type          | Null | Key | Default | Extra |
+------------------+---------------+------+-----+---------+-------+
| Account_No       | int           | YES  |     | NULL    |       |
| UserName         | varchar(50)   | YES  |     | NULL    |       |
| Transaction_Date | date          | YES  |     | NULL    |       |
| Reference_No     | varchar(20)   | YES  |     | NULL    |       |
| Description      | varchar(100)  | YES  |     | NULL    |       |
| Debit            | decimal(10,2) | YES  |     | NULL    |       |
| Credit           | decimal(10,2) | YES  |     | NULL    |       |
| Balance          | decimal(10,2) | YES  |     | NULL    |       |
+------------------+---------------+------+-----+---------+-------+
8 rows in set (0.01 sec)
```

4. Fixed Deposit



```
+---------------------+-------------+------+-----+---------+-------+
| Field               | Type        | Null | Key | Default | Extra |
+---------------------+-------------+------+-----+---------+-------+
| FD_Account_No       | varchar(20) | YES  |     | NULL    |       |
| Main_Account_No     | varchar(20) | YES  |     | NULL    |       |
| Account_Holder_Name | varchar(50) | YES  |     | NULL    |       |
| Deposit_Date        | varchar(30) | NO   | PRI | NULL    |       |
| Deposit_Amount      | varchar(20) | YES  |     | NULL    |       |
| ROI                 | varchar(10) | YES  |     | NULL    |       |
| Tenure              | varchar(30) | YES  |     | NULL    |       |
| Maturity_Date       | varchar(30) | YES  |     | NULL    |       |
| FD_Amount           | varchar(20) | YES  |     | NULL    |       |
+---------------------+-------------+------+-----+---------+-------+
9 rows in set (0.00 sec)
```

5. Passwords



```
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| Password   | varchar(50) | NO   | PRI | NULL    |       |
| Account_No | int         | YES  |     | NULL    |       |
| Username   | varchar(50) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

SOURCE CODE

```
from tkinter import *
from tkinter import ttk,messagebox
from PIL import Image,ImageTk
import time
import datetime
import captcha
from captcha.image import ImageCaptcha
import random
import string
import mysql.connector as sql
print("......INITIALIZING PROGRAM.......")
def inputs():
   global sqluser
   global sqlpasswd
   sqluser=input("Enter username of your MYSQL : ")
   sqlpasswd=input("Enter password of your MYSQL : ")
   if sqluser=="" or sqlpasswd=="":
      print("\nAll fields are required\n")
      inputs()
def initalize():
   start=input("MySQL is need to run this program. Is MYSQL there in your device(Yes/No)\n>>>")
   if start.lower()=="yes":
      inputs()
   elif start=="":
      initalize()
   else:
      print("You can't run this program without MYSQL. You need to install MYSQL in your device")
      exit()
initalize()
#SQL connectivity
try:
   mydb=sql.connect(host="localhost",user=sqluser,passwd=sqlpasswd)
   mysql=mydb.cursor()
except:
   print("Incorect username or password")
   quit()

root=Tk()
scr_width=str(root.winfo_screenwidth()-20)
scr_height=str(root.winfo_screenheight()-70)
root.minsize(700,650)
root.geometry(scr_width+"x"+scr_height+"+2+0")
#root.maxsize(1000,650)
root.title("BANK MANGAEMENT SYSTEM")
root.configure(bg="blue")
#heading
heading=Label(root,text="Welcome to All Citizens Bank",fg='WHITE',font="comicsans 50 bold",bg="blue")
heading.pack(pady=20)
#frame
f1=Frame(root,bg='blue')
f1.place(x=50,y=500,width=520,height=200)
```

```
f2=Frame(root,bg='blue')
f2.place(x=720,y=500,width=520,height=200)
img=Image.open("Untitled.png")
img=img.resize((300,150))
logo=ImageTk.PhotoImage(img)
Label(root,image=logo).pack()
#variables
vfname=StringVar()
upswd=StringVar()
usnm=StringVar()
anm=StringVar()
apswd=StringVar()
a_usernm=StringVar()
a_accno=StringVar()
a_blc=StringVar()
a_typ=StringVar()
a_stus=StringVar()
a_uemail=StringVar()
a_uno=StringVar()
a_udate=StringVar()
a_ugender=StringVar()
a_udob=StringVar()
a_uaddr=StringVar()
vctcno=StringVar()
vgender=StringVar()
vemail=StringVar()
vdob=StringVar()
vaddr=StringVar()
vopen=IntVar()
vpswd=StringVar()
vcnfpswd=StringVar()
vcheck=IntVar()
searchby=StringVar()
vtxt_search=StringVar()
vdate=StringVar()
vrfno=StringVar()
vdesc=StringVar()
vdb=StringVar()
vcr=StringVar()
vstus=StringVar()
vtyp=StringVar()
ans=StringVar()
amount=StringVar()
sen=IntVar()
rad=StringVar()
rad.set(" ")
tenure=StringVar()
maturity=StringVar()
maturity.set(" ")
tendays=StringVar()
tendays.set(" ")
```

```
tenuredays=IntVar()
tenuremonths=StringVar()
tenuremonths.set(" ")
tenureyears=StringVar()
tenureyears.set(" ")
tdays=IntVar()
tmonths=IntVar()
tyears=IntVar()
tnc=IntVar()
fd_pre_accno=StringVar()
Remarks_fd=StringVar()
fundamount=StringVar()
purpose=StringVar()
beneficary=StringVar()
fcheck=IntVar()
bene=StringVar()
bills=StringVar()
bilpay=StringVar()
operator=StringVar()
number=StringVar()

#DATE AND TIME
time_string = time.strftime('%I:%M:%S:%p')
def tick():

        time_string = time.strftime('%I:%M:%S:%p')
        clock.config(text=time_string)
        clock.after(200,tick)
date_string=datetime.date.today()
date_n_time= datetime.datetime.today()
def get_trans():
    mysql.execute("select * from track_transaction")
    y=mysql.fetchall()
    list2=[]
    for i in y:
      x=i[3]
      list2.append(x)
    return list2
def utrno():
   global utr
   n=random.randint(1111111111,9999999999)

   ch=get_trans()

   if n not in ch:
     utr=n
     return n
   else:
      utrno()
try:
   mysql.execute("CREATE DATABASE BANK_MANAGEMENT_SYSTEM")
```

```
    mysql.execute("USE BANK_MANAGEMENT_SYSTEM")
except:
    mysql.execute("USE BANK_MANAGEMENT_SYSTEM")
#user account database module
try:
    mysql.execute("CREATE TABLE account_details(Account_No int(20) primary key,\
Username varchar(50),Balance decimal(50,2),Account_type varchar(30),Account_status varchar(20))")

except:
    pass

try:
    mysql.execute("CREATE TABLE Fixed_Deposit(FD_Account_No varchar(20),\
Main_Account_No varchar(20),Account_Holder_Name varchar(50),Deposit_Date varchar(30) Primary
key,Deposit_Amount Varchar(20),ROI varchar(10),Tenure varchar(50),Maturity_Date varchar(30),FD_Amount
decimal(50,2))")
except:
    pass

try:
    mdt=datetime.date.today()
    mature_date=str(mdt.year)+"-"+str(mdt.month)+"-"+str(mdt.day)
    mysql.execute("select * from Fixed_Deposit where Maturity_Date<='"+str(mature_date)+"'")
    fdb=mysql.fetchall()
    for jil in fdb:
        MaturiAMT=jil[8]
        Main_accno=jil[1]
        mdate=jil[7]
        mysql.execute('select * from account_details')
        rows=mysql.fetchall()
        for row in rows:
            if str(row[0])==str(Main_accno):
                utrno()
                fdBB=int(row[2])+int(MaturiAMT)
                mysql.execute("update account_details set balance="+str(fdBB)+" where Account_No="+str(row[0]))
                mysql.execute("insert into track_transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(row[0],row[1],str(mdate),utr,"Redeemed Maturity
Amount",None,str(MaturiAMT),str(fdBB)))
                mysql.execute("insert into table"+str(row[0])+" values(%s,%s,%s,%s,%s,%s)",(str(mdate),utr,"Redeemed
Maturity Amount",None,str(MaturiAMT),str(fdBB)))
                mysql.execute("delete from Fixed_Deposit where Maturity_Date='"+str(mdate)+"'")

        mydb.commit()
except:
     pass
def acc_insert():
    cmnd="Insert into account_details values(%s,%s,%s,%s,%s)"
    mysql.execute(cmnd,(a_acno,vfname.get(),vopen.get(),vtyp.get(),"Active"))
    mydb.commit()
def get_acc(event):
    focus=t1.focus()
    item=t1.item(focus)
    content=item['values']
    if content != '':
        a_accno.set(content[0])
def pre_get_acc(event):
    focus=fd4_pre.focus()
    item=fd4_pre.item(focus)
    content=item['values']
    if content!="":
        fd_pre_accno.set(content[0])
def acc_search():
    if searchby.get()=="Account_Status":
        mysql.execute("Select * from account_details where Account_status = '"+str(vtxt_search.get())+"'")
        rows=mysql.fetchall()
        t1.delete(*t1.get_children())
        for row in rows:
            t1.insert('',END,values=row)
        mydb.commit()
    elif searchby.get()=="Account_No":
        mysql.execute("Select * from account_details where Account_No = '"+str(vtxt_search.get())+"'")
        rows=mysql.fetchall()
        t1.delete(*t1.get_children())
        for row in rows:
            t1.insert('',END,values=row)
        mydb.commit()
    else:
        mysql.execute("Select * from account_details where "+str(searchby.get())+" LIKE
'%"+str(vtxt_search.get())+"%'")
        rows=mysql.fetchall()

        t1.delete(*t1.get_children())
        for row in rows:
            t1.insert('',END,values=row)
        mydb.commit()
def display_acc():
    mysql.execute("Select * from account_details")
    row=mysql.fetchall()
    t1.delete(*t1.get_children())
    for j in row:
            t1.insert("",END,values=j)
def fd_display_acc():
    mysql.execute("Select * from Fixed_Deposit")
    fd_row=mysql.fetchall()
    if len(fd_row)!=0:
        fd4.delete(*fd4.get_children())
        for j in fd_row:
            fd4.insert("",END,values=j)
def pre_display_acc():
    mysql.execute("Select
FD_Account_No,Account_Holder_Name,Deposit_Date,Deposit_Amount,ROI,Tenure,Maturity_Date,FD_Amount
from Fixed_Deposit where Main_Account_No = '"+str(acountno)+"'")
```

```
    fd_pre_row=mysql.fetchall()
    if len(fd_pre_row)!=0:
        fd4_pre.delete(*fd4_pre.get_children())
        for j in fd_pre_row:
            fd4_pre.insert("",END,values=j)
def send_notice():
    global tx
    global ntic
    if a_accno.get()=="":
        messagebox.showerror("ERROR","Please choose an account to send the notice",parent=ac)
    else:
        ntic=Toplevel(ac)
        ntic.title("NOTICE")
        ntic.geometry("700x400+290+160")
        ntic.config(bg="limegreen")
        Label(ntic,text="Send Notice to "+a_accno.get(),bg="limegreen",fg="#fff",font="comicsans 15
bold").place(x=10,y=10)
        tx=Text(ntic,font="comicsans 14")
        tx.place(x=10,y=50,width=680,height=300)
        Label(ntic,text="*Please don't terminate your message using Enter key*",font="comicsans
10",bg="limegreen").place(x=10,y=360)
        Button(ntic,text="Send",command=nsend,bg="blue",font="comicsans
13",fg="#fff",activebackground="blue",activeforeground="#fff").place(x=590,y=360,width=100)
def nsend():
    f=open(a_accno.get(),'a+')
    lst=[str(date_n_time)+'\n',tx.get('1.0',END)]
    for i in lst:
        f.write(i)
    f.close()
    ntic.destroy()
#TERMS AND CONDITIONS
def terms_n_conditions(event):
    terms=Toplevel(root)
    terms.minsize(1200,670)
    terms.geometry(scr_width+"x"+scr_height+"+2+0")

    terms.title("All Citizens Bank")
    terms.configure(bg="#fff")

    can=Canvas(terms)
    can.place(width=scr_width,height=700,y=30)

    #Scroll bar
    yscroll=Scrollbar(terms,orient=VERTICAL,command=can.yview)
    yscroll.pack(side=RIGHT,fill=Y)

    can.config(yscrollcommand=yscroll.set)
    can.bind('<Configure>',lambda e: can.configure(scrollregion=can.bbox("all")))

    fra2=Frame(can,bg="#fff")
    fra2.place(y=100,width=scr_width,height=scr_height)
    can.create_window((0,0),window=fra2,width=scr_width,height=3000)

    Label(terms,text="Term & Conditions",font="arial 20",bg="#ADD8E6",fg="#0000CD").pack(fill=X)

    Label(fra2,text="Savings Bank Account",font="comicsans 14",bg="white",fg="black").place(x=30,y=20)
    Label(fra2,text='''i. Savings Bank Accounts (SB A/cs) are designed to help customers inculcate the habit of savings.
It helps the customers keep their surplus funds with the bank and earn interest
        while providing the flexibility for withdrawals. ''',font="comicsans
12",bg="white",fg="black",justify=LEFT).place(x=20,y=50)
    Label(fra2,text='''ii. SB A/cs can be opened by an eligible individual in single name or jointly with others and by
certain organisations/agencies approved by RBI.''',font="comicsans
12",bg="white",fg="black",justify=LEFT).place(x=20,y=100)
    Label(fra2,text='''iii. The prospective customer will need to comply with the Know Your Customer (KYC) guidelines
which are mandatory. The objective of KYC guidelines is to prevent misuse of
        the banking system intentionally or unintentionally for criminal purposes/ money laundering and other
fraudulent activities. The KYC guidelines also help banks to understand
        their customers better.''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=130)
    Label(fra2,text='''iv. The customer identification will be on the basis of documents provided by the customer as (a)
Proof of identity and (b) Proof of address. The customer has to submit the
        prescribed application form along with Photographs all cases.''',font="comicsans
12",bg="white",fg="black",justify=LEFT).place(x=20,y=190)
    Label(fra2,text='''(a) Proof of identity (any of the following with authenticated photographs thereon):
    (i) Passport.
    (ii) Voter ID card
    (iii) PAN Card
    (iv) Govt./Defence ID card
    (v) ID cards of reputed employers
    (vi) Driving Licence ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=30,y=240)
    Label(fra2,text='''(b) Proof of current address (any of the following)
    (i) Credit Card Statement
    (ii) Salary slip
    (iii) Income/Wealth Tax Assessment Order
    (iv) Electricity Bill
    (v) Telephone Bill
    (vi) Bank account statement
    (vii) Letter from reputed employer
    (viii) Letter from any recognized public authority
    (ix) Ration Card ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=30,y=380)
    Label(fra2,text='''v. In case of joint accounts, applicants who are not closely related to each other would be
required to establish their identity and address independently. ''',font="comicsans
12",bg="white",fg="black",justify=LEFT).place(x=20,y=570)
    Label(fra2,text='''vi. No frills Account:- Branches may open accounts for those customers who are in no position to
submit the above mentioned documents provided they intend to maintain
       balances not exceeding rupees fifty thousand (Rs. 50,000/-) in all their accounts taken together and the total
credit summation in all the accounts taken together is not expected
       to exceed rupees one lakh (Rs. 1,00,000/-) in a year, subject to:

      a) introduction from another account holder who has been subjected to full KYC procedure. The introducers
account with the bank should be at least six months old and should
        show satisfactory transactions.
      OR
      b) any other evidence as to the identity and address of the customer to the satisfaction of the
bank.''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=600)
    Label(fra2,text='''vii. The applicant(s) will need to come to the branch, in person, for opening the account and will
sign at the relevant places in the presence of a Bank Official. The introducer may
        be required to come to the Bank in person if it is so warranted. ''',font="comicsans
12",bg="white",fg="black",justify=LEFT).place(x=20,y=760)
    Label(fra2,text='''viii. The Bank is required to obtain Permanent Account Number (PAN) of the customer or
declaration in Form No. 60 or 61 as per the I.T. Act (vide Section 39A) from the person
        opening the account. ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=810)
    Label(fra2,text='''ix. Interest @ 4% p.a. with half yearly interests is paid on SB A/cs on the minimum balance
maintained in the account between the 10th and the last day of the month. Interest is
        credited on June 30 and December 31 every year. The interest rate and the method of application are subject to
changes from time to time. ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=860)
    Label(fra2,text='''x. A passbook is issued in all Savings Bank Accounts. Passbooks are immediately updated across
the counter on request. Cheque books are issued on request. 25 cheque
        leaves are issued free in a year. ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=910)
    Label(fra2,text='''xi. Local cheques for collection will be credited to the account within a maximum period of 3/4
days depending on the clearing norms at the centre.

    xii. Immediate credit will be provided for outstation collections up to Rs 20000/- for accounts that are satisfactorily
conducted. Bank will pay interest for delays in collection of
        outstation cheques presented for credit to account beyond 7/10/14 days, depending on the centre of
collection.

    xiii. Service charges are applicable for cheques returned unpaid.

    xiv. Amount in withdrawal slip should be in whole rupees and minimum should be Rs.10/-. Payment by withdrawal
slip to third parties is not allowed.

    xv. Standing Instructions (S.I.) are accepted in SB A/cs. Service charges apply.

    xvi. No minimum balance required to be maintained in a Savings Bank account :

    xvii. For details of service charges applicable to savings bank accounts, please ask your branch. Service charges are
also available on the Banks web site.

    xviii. The service charges/minimum balance requirements are subject to change. For the latest rates please feel
free to contact the branch where you maintain the account or call
        our helpline numbers. ''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=950)



    Label(fra2,text='''Current Account''',font="comicsans 14",bg="white",fg="black",justify=LEFT).place(x=20,y=1300)
    Label(fra2,text='''Current Accounts (C/As) can be opened by individuals, partnership firms, private and public
limited companies, HUFs/ specified associations, societies, trusts etc.

    Formalities/procedures relating to introduction and opening of Current Accounts for individuals are same as those
mentioned for Savings Bank Accounts. For partnership firms,
        limited companies HUFs, trusts etc the documentation formalities will be provided to you on request.

    No interest is payable on credit balances in Current Accounts.
```

The customers may receive the statements of account according to the frequency desired by them.

Cheque books are issued to all Current Account holders and all withdrawals should be made only through issue of cheques. A cheque should not be issued for an amount of less
than Rs. 50/-.

A cheque which is presented more than 6 months after the date of issue will be treated as "stale" and shall not be paid. Such cheques shall be paid only after revalidation by the
drawer.

Cheques should not be drawn without adequate balance or against uncleared effects, in order not to attract the penal provisions of section 138 of the Negotiable Instruments Act.

The cheque book should be kept safely to prevent any misuse and consequential loss to the depositor(s). The loss of any cheque or the cheque book should be promptly reported
to the Bank.

Payment of a cheque can be stopped by the drawer, by giving notice in writing to the Bank, mentioning full details of the cheque, before the cheque is presented for payment. The
Bank will not pay this cheque after recording 'stop payment' in its books.

Standing Instructions (S.I.) are accepted in Current Accounts.

As per RBI directive, the applicant for Current Account should declare in the account opening form or separately that he/they is/are not enjoying any credit facility with any Bank and
if he/they does/do enjoy any credit facility, he/they should declare full particulars thereof indicating the name of the Bank/branch concerned.

In Personal Banking Branches, minimum balance of all deposit accounts is Rs.50,000/- in Metro/Urban branches and Rs.25,000/- in Semi-urban centres.

For pension drawing accounts, a 50% concession is allowed both in minimum balance requirements as well as in service charges.

Facilities like local clearing, immediate credit of outstation cheques etc are as applicable to savings accounts.

For details of service charges applicable to current accounts, please refer Annexure 2.

The service charges/minimum balance requirements are subject to change. For the latest rates please feel free to contact the branch where you maintain the account or call our
helpline numbers mentioned in page''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=1340)

Label(fra2,text='''Term Deposit Accounts ''',font="comicsans 14",bg="white",fg="black",justify=LEFT).place(x=20,y=2050)
Label(fra2,text='''Term Deposit Accounts can be opened by individuals, partnership firms, private and public limited companies, HUFs/specified associations, societies, trusts etc.

Formalities/procedures relating to identification and introduction for opening of Term Deposit Accounts in the name of individuals are same as those mentioned for Savings Bank

Account. For term deposit accounts of limited companies, partnership firms, societies, trusts etc. the documentation formalities will be made available on request.

The Bank is required to obtain Permanent Account Number (PAN) of the customer or declaration in Form No. 60 or 61 as per the I.T. Act (vide Section 39A) from the person opening the account.

Term Deposit Accounts can be opened for a minimum period of 7 days up to maximum period of 10 years. The minimum/maximum periods are subject to change.

The rates of interest vary depending on the period of deposit. The rates applicable as on date can be obtained from the branch and is also available at our web site. Interest is payable at quarterly intervals or at the time of maturity. Interest is also payable monthly at discounted rates.

The Bank issues receipt (Term Deposit Receipt - TDR) for amounts kept in each fixed deposit account. The TDR can be kept in safe custody of the bank free of charge and a safe custody receipt will be issued.

Premature closure of Term Deposit is normally allowed. The rate of interest payable will be the applicable rate (at the time of opening the fixed deposit account) for the period for which the deposit has run less penalty of 1%. The penal provisions for premature closure are subject to change from time to time and may also vary with deposit schemes.

Loan facility is available up to 90% of the principal amount of Term Deposit.

In the absence of specific instructions from the customer, a Term Deposit on maturity is automatically renewed for the same period at the rate of interest prevailing on the date of maturity.

Term Deposit Account can be transferred from one branch to another free of cost. The depositor has the option to submit his application and the TDR at the transferor or transferee branch.

The maturity proceeds will be credited to the current/savings bank account of the depositor if exceeding Rs.20,000/-.

Interest on Term Deposits is subject to income tax. Exemptions are allowed under certain conditions as specified under the Income tax Act 1961. The full particulars of tax provisions applicable will be provided on request. Depositors may furnish Form 15H in duplicate to receive interest without tax deduction at source (TDS).

Form 15H is not acceptable when total interest payable by the Branch to a customer exceeds Rs.50,000/-.

The Bank will issue TDS certificates for the tax deducted.

The Bank may introduce branded term deposit schemes with options /features more attractive to customers. Some of the above provisions in such deposit schemes may be subject to restrictions. Please contact the branch/web site to know the details of such deposit schemes.

```
''',font="comicsans 12",bg="white",fg="black",justify=LEFT).place(x=20,y=2100)
```

```
   Button(fra2,text="Close",command=terms.destroy,font="comicsans 14",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=30,pady=2).place(x=570,y=2870)

#NOTIFICATION FOR BANK(WRITING MODULE)

def write():
   f=open("Bank.txt",'a+')
   lst=[str(acountno)+'\n',str(date_n_time)+'\n',txt.get('1.0',END)]
   for i in lst:
      f.write(i)
   f.close()
   cntc.destroy()

#NOTIFICATION FOR BANK(READING MODULE)

def bank():
   bnk=Toplevel(r)
   bnk.title("NOTIFICATIONS")
   bnk.geometry(scr_width+"x"+scr_height+"+2+0")
   #heading
   Label(bnk,text="Notifications",font="arial 30 bold",bg="#fff").pack(fill=X)
   hei=30000
   my_canvas=Canvas(bnk,bg="#fff")
   my_canvas.place(y=50,width=scr_width,height=hei)
   yscroll=Scrollbar(bnk,orient=VERTICAL,command=my_canvas.yview)
   yscroll.pack(side=RIGHT,fill=Y)

   my_canvas.config(yscrollcommand=yscroll.set)
   my_canvas.bind('<Configure>',lambda e: my_canvas.configure(scrollregion=my_canvas.bbox("all")))

   #frames
   nficfr=Frame(my_canvas,bg="#FFE4C4")
   my_canvas.create_window((0,0),window=nficfr,width=scr_width,height=30000)
   try:
      f=open("Bank.txt",'r')
      y=50
      while True:
            x=f.readline()
            if len(x)==0:
               break
            else:
               Label(nficfr,text="From : "+x,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y)
            x=f.readline()
            Label(nficfr,text=x,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+40)

            x=f.readline()
            if len(x)<=120:
               Label(nficfr,text=x,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+80)
               y+=160
               if y>=500:
```

```
                        hei-=160
                        my_canvas.place(width=scr_width,height=hei)

                elif len(x)>120 and len(x)<240:
                    a=x[:121]
                    Label(nficfr,text=a,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+80)
                    b=x[121:]
                    Label(nficfr,text=b,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+110)

                    y+=190
                    if y>=500:
                        hei-=130
                        my_canvas.place(width=scr_width,height=hei)

    except:
                Label(nficfr,text="*No messages to show*",font="comicsans 30",bg="#FFE4C4").place(x=400,y=200)
#user database module
try:
    mysql.execute("CREATE TABLE user_details(Account_No int(20) primary key,\
Username varchar(50),Email varchar(25),Phone Varchar(20),Registeration_date date,DOB date,\
Address varchar(250),Gender varchar(10),Account_status varchar(20))")
except:
    pass
def userback():
    ud.destroy()
    au_clear()
def acback():
    ac.destroy()
    a_clear()
def insert():
    Active="Active"
    cmd="INSERT INTO user_details VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s)"

mysql.execute(cmd,(a_acno,vfname.get(),vemail.get(),vctcno.get(),date_string,vdob.get(),vaddr.get(),vgender.get(),"
Active"))
    mydb.commit()
def user_search():
    if searchby.get()== "Account_status":
        mysql.execute("Select * from user_details where Account_status LIKE '"+str(vtxt_search.get())+"'")
        rows=mysql.fetchall()
        t2.delete(*t2.get_children())
        for row in rows:
            t2.insert('',END,values=row)
        mydb.commit()
    elif searchby.get()== "Account_No":
        mysql.execute("Select * from user_details where Account_No LIKE '"+str(vtxt_search.get())+"'")
        rows=mysql.fetchall()
        t2.delete(*t2.get_children())
        for row in rows:
            t2.insert('',END,values=row)
        mydb.commit()
```

```
   else :
      mysql.execute("Select * from user_details where "+searchby.get()+" LIKE '"+str(vtxt_search.get())+"'")
      rows=mysql.fetchall()
      t2.delete(*t2.get_children())
      for row in rows:
         t2.insert('',END,values=row)
      mydb.commit()
def display_user():
   mysql.execute("Select * from user_details")
   rows=mysql.fetchall()
   t2.delete(*t2.get_children())
   for row in rows:
         t2.insert("",END,values=row)
def getuser(event) :
   focus=t2.focus()
   item=t2.item(focus)
   content=item["values"]
   if content!="":
      a_accno.set(content[0])
      a_usernm.set(content[1])
      a_uemail.set(content[2])
      a_uno.set(content[3])
      a_udate.set(content[4])
      a_udob.set(content[5])
      a_uaddr.set(content[6])
      a_ugender.set(content[7])
      a_stus.set(content[8])
def delete_user():
   if a_accno.get()=="":
      messagebox.showerror("ERROR","Please choose an account to delete",parent=ac)
   else:
      mysql.execute("Delete from user_details where Account_No="+a_accno.get())
      mysql.execute("Delete from account_details where Account_No="+a_accno.get())
      mysql.execute("Delete from passwds where Account_No="+a_accno.get())
      mysql.execute("Delete from track_transaction where Account_No="+a_accno.get())
      mysql.execute("Delete from fixed_deposit where Main_Account_No="+a_accno.get())
      mysql.execute("Drop table table"+a_accno.get())
      mydb.commit()
      display_acc()
      a_clear()
      d.clear()
      dictionary()
def update_user():
   user_detail="UPDATE user_details set
Username=%s,Email=%s,Phone=%s,DOB=%s,Address=%s,Gender=%s,Account_status=%s where Account_No=%s"

mysql.execute(user_detail,(a_usernm.get(),a_uemail.get(),a_uno.get(),a_udob.get(),a_uaddr.get(),a_ugender.get(),a
_stus.get(),a_accno.get()))
   acc_detail="UPDATE account_details set Username=%s,Account_status=%s where Account_No=%s"
   mysql.execute(acc_detail,(a_usernm.get(),a_stus.get(),a_accno.get()))
   mysql.execute("UPDATE passwds set Username='"+a_usernm.get()+"' where Account_No="+a_accno.get())
```

```
    mysql.execute("UPDATE track_transaction set Username='"+a_usernm.get()+"' where 
Account_No="+a_accno.get())
    mysql.execute("UPDATE fixed_deposit set Account_Holder_Name='"+a_usernm.get()+"' where 
Main_Account_No="+a_accno.get())
    mydb.commit()
    display_user()
    au_clear()
    d.clear()
    dictionary()
def notice() :
    global hei
    nfication=Toplevel(us)
    nfication.title("NOTIFICATIONS")
    nfication.geometry(scr_width+"x"+scr_height+"+2+0")
    hei=30000
    my_canvas=Canvas(nfication,bg="#fff")
    my_canvas.place(y=50,width=scr_width,height=hei)
    yscroll=Scrollbar(nfication,orient=VERTICAL,command=my_canvas.yview)
    yscroll.pack(side=RIGHT,fill=Y)

    my_canvas.config(yscrollcommand=yscroll.set)
    my_canvas.bind('<Configure>',lambda e: my_canvas.configure(scrollregion=my_canvas.bbox("all")))

    #frames
    nficfr=Frame(my_canvas,bg="#FFE4C4")
    my_canvas.create_window((0,0),window=nficfr,width=scr_width,height=30000)
    Label(nfication,text="Noifications from Bank",font="comicsans 30 bold",bg="#ADD8E6",fg="#0000CD").pack(fill=X)
    try:
        file=str(acountno)
        f=open(file,'r')
        y=50
        while True:
                x=f.readline()
                if len(x)==0:
                    break
                else:
                    Label(nficfr,text=x,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y)

                x=f.readline()
                if len(x)==0:
                    break
                elif len(x)<=120:
                    Label(nficfr,text=x,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+40)
                    y+=110
                    if y>=500:
                        hei-=160
                        my_canvas.place(width=scr_width,height=hei)

                elif len(x)>120 and len(x)<240:
                    a=x[:121]
                    Label(nficfr,text=a,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+40)
                b=x[121:]
                Label(nficfr,text=b,font="comicsans 15",bg="#FFE4C4").place(x=10,y=y+70)

                y+=140
                if y>=500:
                    hei-=160
                    my_canvas.place(width=scr_width,height=hei)
    except:
            Label(nficfr,text="*No messages to show*",font="comicsans 30",bg="#FFE4C4").place(x=200,y=200)
#modules for generating account no.
def show():
    mysql.execute("select * from user_details")
    y=mysql.fetchall()
    list1=[]
    for i in y:
      x=i[0]
      list1.append(x)
    return list1
def account_num():
    global a_acno
    n=random.randint(10000000,99999999)
    ch=show()
    if n not in ch:
      a_acno=n
      return n
    else:
        account_num()
def accinfo():
    global dis4
    info=Toplevel(us)
    info.geometry(scr_width+"x"+scr_height+"+2+0")
    info.title("ACCOUNT INFO")
    info.config(bg="#fff")
    #heading
    Label(info,text="Account info",bg="#ADD8E6",fg="#0000CD",font="Arial 50 bold").pack(fill=X)
    #frame
    infr=Frame(info,bg="#fff")
    infr.place(y=70,width=scr_width,height=scr_height)
    #content
    Label(infr,text="Account No.",font="comicsans 23 bold").place(x=10,y=40)
    Label(infr,text="Customer Name",font="comicsans 23 bold").place(x=10,y=140)
    Label(infr,text="Email",font="comicsans 23 bold").place(x=620,y=40)
    Label(infr,text="Contact No.",font="comicsans 23 bold").place(x=620,y=140)
    Label(infr,text="Date Of Birth",font="comicsans 23 bold").place(x=10,y=240)
    Label(infr,text="Gender",font="comicsans 23 bold").place(x=620,y=240)
    Label(infr,text="Registration Date",font="comicsans 23 bold").place(x=10,y=340)
    Label(infr,text="Account Type",font="comicsans 23 bold").place(x=620,y=340)
    Label(infr,text="Account Status",font="comicsans 23 bold").place(x=10,y=440)
    Label(infr,text="Balance",font="comicsans 23 bold").place(x=620,y=440)
    Label(infr,text="Address",font="comicsans 23 bold").place(x=10,y=540)
    Label(infr,text=acountno,font="comicsans 23",bg="#fff").place(x=200,y=40)
    Label(infr,text=user,font="comicsans 23",bg="#fff").place(x=300,y=140)
    Label(infr,text=mail,font="comicsans 23",bg="#fff").place(x=790,y=40)
    Label(infr,text=phno,font="comicsans 23",bg="#fff").place(x=820,y=140)
    Label(infr,text=dob,font="comicsans 23",bg="#fff").place(x=230,y=240)
    Label(infr,text=gender,font="comicsans 23",bg="#fff").place(x=800,y=240)
    Label(infr,text=reg_date,font="comicsans 23",bg="#fff").place(x=300,y=340)
    Label(infr,text=acctyp,font="comicsans 23",bg="#fff").place(x=850,y=340)
    Label(infr,text=status,font="comicsans 23",bg="#fff").place(x=300,y=440)
    dis4=Label(infr,text=blnc,font="comicsans 23",bg="#fff")
    dis4.place(x=820,y=440)
    Label(infr,text=address,font="comicsans 23",bg="#fff").place(x=200,y=540)
def per_user_details():
    global acountno
    global user
    global mail
    global phno
    global reg_date
    global dob
    global address
    global gender
    global status
    global blnc
    global acctyp
    global accstus
    mysql.execute("select * from account_details")
    uy=mysql.fetchall()

    mysql.execute("select * from user_details")
    dy=mysql.fetchall()

    mysql.execute("select * from passwds")
    py=mysql.fetchall()
    for j in py:
        if upswd.get()==j[0]:
            for i in uy:
                if j[1]==i[0]:
                    blnc=i[2]
                    acctyp=i[3]
                    accstus=i[4]
            for k in dy:
                if j[1]==k[0]:
                    acountno=k[0]
                    user=k[1]
                    mail=k[2]
                    phno=k[3]
                    reg_date=k[4]
                    dob=k[5]
                    address=k[6]
                    gender=k[7]
                    status=k[8]
```

```
#USER ACCOUNT TRANSACTIONS TABLE AND UPDATES

try:
        tab="Create table Track_Transaction(Account_No int(10),UserName varchar(50),Transaction_Date
date,Reference_No varchar(20),Description varchar(100),Debit decimal(10,2),Credit decimal(10,2),Balance
decimal(10,2))"
        mysql.execute(tab)

except:
        pass
def transaction():
   try:
      tablename='table'+str(a_acno)
      t="Create table "+tablename+"(Transaction_Date date,Reference_No varchar(20),Description
varchar(100),Debit decimal(10,2),Credit decimal(10,2),Balance decimal(10,2))"
      mysql.execute(t)
   except:
     pass
def fund_beneficary():
   global balanc
   mysql.execute("Select * from account_details")
   r=mysql.fetchall()
   for i in r:
     if int(beneficary.get())==i[0]:
         balanc=i[2]
         balanc=int(balanc)+int(fundamount.get())
         desc="From Account no. "+str(acountno)
         mysql.execute("Insert into "+"table"+beneficary.get()+"
values(%s,%s,%s,%s,%s,%s)",(date_string,utr,desc,None,fundamount.get(),balanc))
         mysql.execute("Insert into Track_Transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(beneficary.get(),i[1],date_string,utr,desc,None,fundamount.get(),balanc))
         mysql.execute("Update account_details set Balance=%s where Account_No=%s",(balanc,beneficary.get()))
   mydb.commit()
def fund_update():
   global balanc
   global blnc
   blnc=int(blnc)-int(fundamount.get())
   bal=str(blnc)
   dis.config(text=bal)
   tname="table"+str(acountno)
   desc=purpose.get()+"    Account no."+beneficary.get()
   mysql.execute("Update account_details set Balance=%s where Account_No=%s",(bal,acountno))
   mysql.execute("Insert into "+tname+"
values(%s,%s,%s,%s,%s,%s)",(date_string,utr,desc,fundamount.get(),None,bal))
   mysql.execute("Insert into Track_Transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(acountno,user,date_string,utr,desc,fundamount.get(),None,bal))
   mydb.commit()
   fund_beneficary()
def bill_update():
   global blnc
   x=int(blnc)
   y=int(bilpay.get())
   blnc=x-y
   bal=str(blnc)
   dis.config(text=bal)
   mysql.execute("Update account_details set Balance=%s where Account_No=%s",(bal,acountno))
   if bills.get()=="Electricity Bill":
      desc=bills.get()+"    Account no. 899287635"
   elif bills.get()=="Water Bill":
      desc=bills.get()+"    Account no. 978065476"
   elif bills.get()=="Gas Bill":
      desc=bills.get()+"    Account no. 790865354"
   statement="Insert into "+tablename+" values(%s,%s,%s,%s,%s,%s)"
   mysql.execute(statement,(date_string,utr,desc,bilpay.get(),None,bal))
   mysql.execute("Insert into Track_Transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(acountno,user,date_string,utr,desc,bilpay.get(),None,bal))
   mydb.commit()
def fetch_statement():
   sta="Select * from "+tablename
   mysql.execute(sta)
   rows=mysql.fetchall()
   t4.delete(*t4.get_children())
   for row in rows:
      t4.insert("",END,values=row)
def track_user():
   mysql.execute("Select * from Track_Transaction")
   rows=mysql.fetchall()
   t3.delete(*t3.get_children())
   for row in rows:
      t3.insert("",END,values=row)
#user dictionary
try:
   mysql.execute("CREATE TABLE passwds(Password varchar(50) Primary key,Account_No int(20),Username
Varchar(50))")
except:
   pass
def paswd():
   store="Insert into passwds values(%s,%s,%s)"
   mysql.execute(store,(vpswd.get(),a_acno,vfname.get()))
   mydb.commit()
d={}
def dictionary():
   d.clear()
   mysql.execute("Select * from passwds")
   row=mysql.fetchall()
   if len(row)!= 0:
      for i in row:
         psd=i[0]
         usn=i[2]
         acno=i[1]
```

```
         d[psd]=usn
dictionary()
#Captcha modules
def generate_captcha():

     global image_captcha
     data_set=list(string.ascii_lowercase+string.digits)
     s=''
     for i in range(6):
       a=random.choice(data_set)

       s+=a
       data_set.remove(a)
     image_captcha=s
     return s
def generate_first_image():
  img=ImageCaptcha()
  s=generate_captcha()
  value=img.generate(s)
  img.write(s,"1.png")
def regenerate_image_captcha():
  global c
  global cap
  img=ImageCaptcha()
  s=generate_captcha()
  value=img.generate(s)
  img.write(s,"1.png")
  #os.startfile('2.png')
  cap=PhotoImage(file="1.png")
  c.config(image=cap)
def check_image_captcha():
   if ans.get()==image_captcha:
     account_num()
     ok=messagebox.showinfo("Correct","Account is successfully created.\n Your Account no. is 
"+str(a_acno),parent=capt)
     if ok=="ok":
        insert()
        acc_insert()
        paswd()
        dictionary()
        capt.destroy()
        vfname.set("")
        vctcno.set("")
        vdob.set("")
        vaddr.set("")
        vopen.set(0)
        vpswd.set("")
        vcnfpswd.set("")
        vgender.set("--Select--")
        vtyp.set("--Select--")
        vemail.set("")
```

```
          vcheck.set(0)
          transaction()
          e.destroy()
    else:
         messagebox.showerror("Error","Wrong!!",parent=capt)
         regenerate_image_captcha()
def captcha():
    global a
    global c
    global capt
    capt=Toplevel(root)
    capt.title("HUMAN VERIFICATION")
    capt.config(bg="white")
    xaxis=int(capt.winfo_screenwidth()/2-175)
    yaxis=int(capt.winfo_screenheight()/2-125)
    capt.geometry("350x250+"+str(xaxis)+"+"+str(yaxis))
    generate_first_image()
    a=PhotoImage(file="1.png")
    c=Label(capt,image=a)
    c.place(x=50,y=20)
    Entry(capt,textvariable=ans,font="comicsans 15",bd=2,relief=GROOVE).place(x=50,y=100)
    ans.set("")
    btn=Button(capt,text='check',command=check_image_captcha,font="comicsans 15")
    btn.place(x=50,y=150)
#clear commands for Admin section
def a_clear():
    a_accno.set("")
    searchby.set("--Select--")
    vtxt_search.set("")
def au_clear():
    a_usernm.set("")
    a_accno.set("")
    a_blc.set("")
    a_typ.set("")
    a_stus.set("--Select--")
    a_uemail.set("")
    a_uno.set("")
    a_udate.set("")
    a_ugender.set("--Select--")
    a_udob.set("")
    a_uaddr.set("")
    searchby.set("--Select--")
    vtxt_search.set("")

#Admin Acoount Details
def acc_details():
    global t1
    global ac
    #r.destroy()
    ac=Toplevel(r)
    ac.config(bg="blue")
    ac.title('ACCOUNT DETAILS')
    ac.geometry(scr_width+"x"+scr_height+"+2+0")
    Label(ac,text="Account Details List",font="comicsans 40 bold",bg="blue",fg="#fff").pack(fill=X)
    #main frames
    mframe1=Frame(ac,bg='blue')
    mframe1.place(x=10,y=90,width=1250,height=620)
    #mframe1 content
    Label(mframe1,text="Account No.",bg="blue",fg="white",font=("Times new
roman",20,"bold")).grid(row=0,column=0,padx=10)
    Entry(mframe1,font=("Times new
roman",20,"bold"),bd=3,relief=GROOVE,textvariable=a_accno,state="readonly").grid(row=0,column=1,pady=10,stick
y='w')
    a_accno.set("")
    #button
    Button(mframe1,text="Send Notice",width=10,pady=3,font=("Times new
roman",20,"bold"),command=send_notice,bg="limegreen",fg="#fff",activebackground="lawngreen",activeforegroun
d="#fff").grid(row=2,column=0,pady=20,padx=10)
    Button(mframe1,text="Clear",width=10,pady=3,font=("Times new
roman",20,"bold"),command=a_clear).grid(row=3,column=0,pady=20,padx=10)
    Button(mframe1,text="Delete",width=10,pady=3,font=("Times new
roman",20,"bold"),command=delete_user,bg="red",fg="#fff",activebackground="red",activeforeground="#fff").grid(
row=4,column=0,pady=20,padx=10)
    #mframe2 contents
    search=Label(mframe1,text="Search By",bg="blue",fg="white",font=("Times new roman",20,"bold"))
    search.grid(row=0,column=3,padx=10,pady=10)

    ch_search=ttk.Combobox(mframe1,width=12,textvariable=searchby,font=("Times new
roman",15,"bold"),justify=CENTER,state="readonly")
    ch_search['values']=("--Select--",'Account_No','Username','Account_type','Account_Status')
    ch_search.grid(row=0,column=4,pady=10,padx=10,sticky='w')
    ch_search.current(0)

    ac.option_add("*TCombobox*Listbox.font","comicsans 13")

    txt_search=Entry(mframe1,textvariable=vtxt_search,font=("Times new roman",15,"bold"),bd=2,relief=GROOVE)
    txt_search.grid(row=0,column=5,pady=10,padx=5,sticky='w')
    vtxt_search.set("")


search_btn=Button(mframe1,text="Search",width=10,pady=5,command=acc_search).grid(row=0,column=6,pady=10
,padx=5)
    Showall=Button(mframe1,text="Show
All",width=10,pady=5,command=display_acc).grid(row=0,column=7,pady=10,padx=5)
    #subframe
    tablefr=Frame(mframe1)
    #scrollbar
    xscroll=Scrollbar(tablefr,orient=HORIZONTAL)
    yscroll=Scrollbar(tablefr,orient=VERTICAL)
    #table
```

```
t1=ttk.Treeview(tablefr,show="headings",columns=('accno','uname','blc','typ','stus'),xscrollcommand=xscroll.set,yscr
ollcommand=yscroll.set)
   xscroll.pack(side=BOTTOM,fill=X)
   yscroll.pack(side=RIGHT,fill=Y)
   xscroll.config(command=t1.xview)
   yscroll.config(command=t1.yview)
   tablefr.place(x=200,y=70,width=1050,height=500)
   #column
   t1.column('uname',anchor=CENTER)
   t1.column('accno',anchor=CENTER)
   t1.column('blc',anchor=CENTER)
   t1.column('typ',anchor=CENTER)
   t1.column('stus',anchor=CENTER)
   #headings
   t1.heading('accno',text="Account No.")
   t1.heading('uname',text="UserName")
   t1.heading('blc',text="Balance")
   t1.heading('typ',text="Account Type")
   t1.heading('stus',text="Account Status")
   t1.pack(fill=BOTH,expand='yes')
   display_acc()
   t1.bind("<ButtonRelease-1>",get_acc)
   bck=Button(mframe1,text="<Back",font="comicsans 12 bold",command=acback).place(x=1150,y=575)
def user_details():
   global t2
   global ud
   #r.destroy()
   ud=Toplevel(r)
   ud.config(bg="blue")
   ud.title('USER DETAILS')
   ud.geometry(scr_width+"x"+scr_height+"+2+0")
   Label(ud,text="User Details List",font="comicsans 40 bold",bg="blue",fg="#fff").pack(fill=X)
   #main frames
   mframe3=Frame(ud,bg='blue')
   mframe3.place(x=0,y=90,width=1270,height=610)
   #title
   title=Label(mframe3,text="Manage Users",bg="blue",fg="white",font=("Times new roman",25,"bold"))
   title.grid(row=0,columnspan=2,pady=20)

   #Label & Entries
   Label(mframe3,text="Account No.",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=3,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_accno,state='readonly').grid(row=3,column=1,pady=5,sticky=
'w')
   a_accno.set("")
   Label(mframe3,text="UserName",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=4,column=0,padx=10)
```

```
   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_usernm).grid(row=4,column=1,pady=5,sticky='w')
   a_usernm.set("")
   Label(mframe3,text="Email",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=5,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_uemail).grid(row=5,column=1,pady=5,sticky='w')
   a_uemail.set("")
   Label(mframe3,text="Phone No.",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=6,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_uno).grid(row=6,column=1,pady=5,sticky='w')
   a_uno.set("")
   Label(mframe3,text="Register Date",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=7,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_udate,state="readonly").grid(row=7,column=1,pady=5,sticky
='w')
   a_udate.set("")
   Label(mframe3,text="D.O.B.",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=8,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_udob).grid(row=8,column=1,pady=5,sticky='w')
   a_udob.set("")
   Label(mframe3,text="Address",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=9,column=0,padx=10)

   Entry(mframe3,font=("Times new
roman",19,"bold"),bd=2,relief=GROOVE,textvariable=a_uaddr).grid(row=9,column=1,pady=5,sticky='w')
   a_uaddr.set("")
   gender=Label(mframe3,text="Gender",bg="blue",fg="white",font=("Times new roman",19,"bold"))
   gender.grid(row=10,column=0,padx=10)

   txt_gender=ttk.Combobox(mframe3,textvariable=a_ugender,font=("Times new
roman",19,"bold"),state="readonly")
   txt_gender['values']=("--Select--","Male","Female","Other")
   txt_gender.grid(row=10,column=1,pady=5,sticky='w')
   txt_gender.current(0)

   Label(mframe3,text="Account Status",bg="blue",fg="white",font=("Times new
roman",19,"bold")).grid(row=11,column=0,padx=10)

   txt_stus=ttk.Combobox(mframe3,font=("Times new roman",19,"bold"),textvariable=a_stus,state="readonly")
   txt_stus['values']=('--Select--','Active','Inactive')
   txt_stus.grid(row=11,column=1,pady=5,sticky='w')
   txt_stus.current(0)
   ud.option_add("*TCombobox*Listbox.font","comicsans 13")
```

```
    #button
    bfrm=Frame(mframe3,bg="blue")
    update=Button(bfrm,text="Update",width=10,pady=3,font=("Times new
roman",15,"bold"),command=update_user).grid(row=1,column=1,pady=0,padx=25)
    clear=Button(bfrm,text="Clear",width=10,pady=3,font=("Times new
roman",15,"bold"),command=au_clear).grid(row=1,column=3,pady=0,padx=15)
    bfrm.place(x=5,y=550,height=45)

    bck=Button(mframe3,text="<Back",font="comicsans 12 bold",command=userback).place(x=1200,y=575)

    #search
    search=Label(mframe3,text="Search By",bg="blue",fg="white",font=("Times new roman",20,"bold"))
    search.grid(row=0,column=4,padx=40,pady=10)

    ch_search=ttk.Combobox(mframe3,width=12,textvariable=searchby,justify=CENTER,font=("Times new
roman",15,"bold"),state="readonly")
    ch_search['values']=('--Select--','Account_No','Username','Account_Status','Gender')
    ch_search.grid(row=0,column=5,pady=10,padx=10,sticky='w')
    ch_search.current(0)
    txt_search=Entry(mframe3,textvariable=vtxt_search,font=("Times new roman",15,"bold"),bd=2,relief=GROOVE)
    txt_search.grid(row=0,column=6,pady=10,padx=10,sticky='w')
    vtxt_search.set("")
    search_btn=Button(mframe3,text="Search",width=8,command=user_search,pady=5,font=("Times new
roman",12,"bold")).grid(row=0,column=7,pady=10,padx=5)
    Showall=Button(mframe3,text="Show All",width=8,command=display_user,pady=5,font=("Times new
roman",12,"bold")).grid(row=0,column=8,pady=10,padx=5)

    #subframe
    tablefr=Frame(mframe3)
    #scrollbar
    xscroll=Scrollbar(tablefr,orient=HORIZONTAL)
    yscroll=Scrollbar(tablefr,orient=VERTICAL)
    #table
    t2=ttk.Treeview(tablefr,show="headings",xscrollcommand=xscroll.set,yscrollcommand=yscroll.set)
    t2['columns']=('accno','uname','email','phno','audate','dob','address','gender','stus')
    xscroll.pack(side=BOTTOM,fill=X)
    yscroll.pack(side=RIGHT,fill=Y)
    xscroll.config(command=t2.xview)
    yscroll.config(command=t2.yview)
    tablefr.place(x=530,y=70,width=730,height=500)

    #column
    t2.column('uname',anchor=CENTER)
    t2.column('accno',anchor=CENTER,width=100)
    t2.column('email',anchor=CENTER)
    t2.column('phno',anchor=CENTER,width=100)
    t2.column('audate',anchor=CENTER,width=130)
    t2.column('stus',anchor=CENTER,width=100)
    t2.column('gender',anchor=CENTER,width=100)
    t2.column('dob',anchor=CENTER,width=100)
    t2.column('address',anchor=CENTER,width=400)
    #headings
    t2.heading('accno',text="Account No.")
    t2.heading('uname',text="UserName")
    t2.heading('email',text="Email Address")
    t2.heading('phno',text="Phone No.")
    t2.heading('audate',text="Registeration Date")
    t2.heading('stus',text="Account Status")
    t2.heading('gender',text="Gender")
    t2.heading('dob',text="Date Of Birth")
    t2.heading('address',text="Address")
    t2.pack(fill=BOTH,expand='yes')
    t2.bind("<ButtonRelease-1>",getuser)
    display_user()

#Account Statements

def acc_stats():
    global t3
    acs=Toplevel(r)
    acs.config(bg="blue")
    acs.title('ACCOUNT STATEMENTS')
    acs.geometry(scr_width+"x"+scr_height+"+2+0")
    Label(acs,text="Account Statements",font="comicsans 40 bold",bg="blue",fg="#fff").pack(fill=X)
    #main frames
    mframe5=Frame(acs,bg='blue')
    mframe5.place(y=90,width=1300,height=620)
    #mframe5 content



    #subframe
    tablefr=Frame(mframe5)
    #scrollbar
    xscroll=Scrollbar(tablefr,orient=HORIZONTAL)
    yscroll=Scrollbar(tablefr,orient=VERTICAL)
    #table

t3=ttk.Treeview(tablefr,show="headings",columns=('acno','user','date','utr','desc','debit','credit','balnc'),xscrollcomm
and=xscroll.set,yscrollcommand=yscroll.set)
    xscroll.pack(side=BOTTOM,fill=X)
    yscroll.pack(side=RIGHT,fill=Y)
    xscroll.config(command=t3.xview)
    yscroll.config(command=t3.yview)
    tablefr.place(x=10,y=70,width=1260,height=500)

    #column
    t3.column('acno',anchor=CENTER,width=150)
    t3.column('user',anchor=CENTER)
    t3.column('date',anchor=CENTER,width=150)
    t3.column('utr',anchor=CENTER,width=150)
    t3.column('desc',anchor=CENTER,width=300)
    t3.column('debit',anchor=CENTER,width=150)
    t3.column('credit',anchor=CENTER,width=150)
    t3.column('balnc',anchor=CENTER,width=150)
    #headings
    t3.heading('acno',text="Account No.")
    t3.heading('user',text="Username")
    t3.heading('date',text="Transaction Date")
    t3.heading('utr',text="Reference No.")
    t3.heading('desc',text="Description")
    t3.heading('debit',text="Debit")
    t3.heading('credit',text="Credit")
    t3.heading('balnc',text='Balance')
    t3.pack(fill=BOTH,expand='yes')
    track_user()

    bck=Button(mframe5,text="<Back",font="comicsans 12 bold",command=acs.destroy).place(x=1150,y=575)

#Record of Fixed Deposits
def records_FD():
    global fd4
    fdad=Toplevel(r)
    scr_width=str(fdad.winfo_screenwidth()-20)
    scr_height=str(fdad.winfo_screenheight()-70)
    fdad.minsize(scr_width,670)
    fdad.geometry(scr_width+"x"+scr_height+"+2+0")

    fdad.title("Record of Fixed Deposits")
    fdad.geometry(scr_width+"x"+scr_height+"+2+0")
    fdad.config(bg="blue")
    #heading
    Label(fdad,text="Record of Fixed Deposits",font="comicsans 35 bold",bg="blue",fg="#fff").pack(fill=X)
    #frame
    tablefd=Frame(fdad)
    tablefd.place(x=0,y=65,width=1260,height=550)
    #scrollbar
    xscroll=Scrollbar(tablefd,orient=HORIZONTAL)
    yscroll=Scrollbar(tablefd,orient=VERTICAL)
    #table

fd4=ttk.Treeview(tablefd,show="headings",columns=('FD_AccNo','AccNo','AHN','DepositDate','Deposit_Amount','RO
I','Tenure','MaturityDate','Amountt'),xscrollcommand=xscroll.set,yscrollcommand=yscroll.set)
    xscroll.pack(side=BOTTOM,fill=X)
    yscroll.pack(side=RIGHT,fill=Y)
    xscroll.config(command=fd4.xview)
    yscroll.config(command=fd4.yview)

    #column
    fd4.column('FD_AccNo',anchor=CENTER,width=50)
    fd4.column('AccNo',anchor=CENTER,width=50)
    fd4.column('AHN',anchor=CENTER,width=130)
    fd4.column('DepositDate',anchor=CENTER,width=110)
```

```
   fd4.column('Deposit_Amount',anchor=CENTER,width=60)
   fd4.column('ROI',anchor=CENTER,width=15)
   fd4.column('Tenure',anchor=CENTER,width=115)
   fd4.column('MaturityDate',anchor=CENTER,width=60)
   fd4.column('Amountt',anchor=CENTER,width=60)

   #headings
   fd4.heading('FD_AccNo',text="FD Account No.")
   fd4.heading('AccNo',text="Main Account No.")
   fd4.heading('AHN',text="Account Holder Name")
   fd4.heading('DepositDate',text="Deposit Date")
   fd4.heading('Deposit_Amount',text="Deposit Amount")
   fd4.heading('ROI',text="ROI")
   fd4.heading('Tenure',text="Tenure")
   fd4.heading('MaturityDate',text='Maturity Date')
   fd4.heading('Amountt',text='Maturity Amount')
   fd4.pack(fill=BOTH,expand='yes')

   fd_display_acc()

   Button(fdad,text="<Back",font="comicsans 12 bold",command=fdad.destroy,width=10).place(x=1130,y=625)

def ent(event):
   tc.config(font=("Times New Roman",12,"underline"))
def lea(event):
   tc.config(font=("Times New Roman",12))

#Create new account

def create_acc():
   global tc
   global e
   e=Toplevel(root)
   e.title("CREATE NEW ACCOUNT")
   e.geometry("1260x850")
   e.configure(bg="blue")
   def register():
    global unm
    a='-'
    if(vfname.get()=="" or vtyp.get()=="--Select--"  or vgender.get()=="--Select--" or vemail.get()=="" or 
vdob.get()==""
     or vaddr.get()=="" or vopen.get()==0 or vpswd.get()=="" or vcnfpswd.get()==""):
       messagebox.showerror("Error","All Fields Are Required",parent=rfrm)
    elif len(vctcno.get())!= 10 or vctcno.get().isdigit()== False:
       messagebox.showerror("Error","Incorrect Contact Number",parent=rfrm)
    elif vemail.get()[-4:]!=".com":
        messagebox.showerror("Error","Incorrect Email",parent=rfrm)
    elif a in vdob.get()[:4]:
        messagebox.showinfo("Error","Incorrect Date Format.\nDate must be in format yyyy-mm-dd",parent=rfrm)
    elif vopen.get()<1000:
        messagebox.showinfo("Notification","Opening amount should not be less than Rs1000",parent=rfrm)
    elif vopen.get()>100000:
        messagebox.showinfo("Notification","Opening amount should not be more than Rs100000",parent=rfrm)
    elif len(vpswd.get())<8:
      messagebox.showinfo("Notification","Your Password must contain atleast 8 characters",parent=rfrm)
    elif vpswd.get() in d.keys():
        messagebox.showerror("Error","This password has already taken\n Set a different password.",parent=rfrm)
    elif vpswd.get()!= vcnfpswd.get():
      messagebox.showerror("Error","Passwords don't match",parent=rfrm)
    elif vcheck.get()==0:
       messagebox.showerror("Error","Please Agree to our Terms and Conditions",parent=rfrm)

    else:
        captcha()
  #frame
  rfrm=Frame(e,bg='white')
  #heading
  Label(e,text='Registeration form',font="comicsans 30 bold",bg='white').pack(fill=X)
  #Labels
  Label(rfrm,text='User Name',font="comicsans 20 bold").place(x=50,y=30)
  Label(rfrm,text='Account Type',font="comicsans 20 bold").place(x=590,y=30)
  Label(rfrm,text='Contact No.',font="comicsans 20 bold").place(x=50,y=130)
  Label(rfrm,text='Gender',font="comicsans 20 bold").place(x=590,y=130)
  Label(rfrm,text='Email ID',font="comicsans 20 bold").place(x=50,y=230)
  Label(rfrm,text='Date of Birth(yyyy-mm-dd)',font="comicsans 20 bold").place(x=590,y=230)
  Label(rfrm,text='Address',font="comicsans 20 bold").place(x=50,y=330)
  Label(rfrm,text="Opening Amount",font="comicsans 20 bold").place(x=590,y=330)
  Label(rfrm,text='Set a Password ',font="comicsans 20 bold").place(x=50,y=430)
  Label(rfrm,text='Confirm Password',font="comicsans 20 bold").place(x=590,y=430)
  #Entries

  Entry(rfrm,font="comicsans 20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vfname).place(x=50,y=80)
  ty=ttk.Combobox(rfrm,font="comicsans 20",justify=CENTER,textvariable=vtyp,values=('--Select--',"Saving
Account","Current Account"),state='readonly')
  ty.place(x=590,y=80)
  ty.current(0)
  Entry(rfrm,font="comicsans 20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vctcno).place(x=50,y=180)
  box=ttk.Combobox(rfrm,font="comicsans 20",values=("--Select--
","Male","Female","Other"),textvariable=vgender,state='readonly',justify=CENTER)
  box.place(x=590,y=180)
  box.current(0)
  e.option_add("*TCombobox*Listbox.font","comicsans 15")
  Entry(rfrm,font="comicsans 20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vemail).place(x=50,y=280)
  Entry(rfrm,font="comicsans
20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vdob).place(x=590,y=280,width=280)
  Entry(rfrm,font="comicsans
20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vaddr).place(x=50,y=380,width=500)
  Entry(rfrm,font="comicsans 20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vopen).place(x=590,y=380)
  Entry(rfrm,font="comicsans 20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vpswd).place(x=50,y=480)
  Entry(rfrm,font="comicsans
20",bd=3,relief=RIDGE,justify=CENTER,textvariable=vcnfpswd).place(x=590,y=480,width=300)
```

```
   Checkbutton(rfrm,text="I Agree The ",variable=vcheck,onvalue=1,offvalue=0,bg='white',font=("Times New
Roman",12)).place(x=50,y=520)
   tc=Label(rfrm,text="Terms and Conditions",bg="#fff",fg="#0000CD",font=("Times New Roman",12))
   tc.place(x=150,y=523)
   tc.bind("<Enter>",ent)
   tc.bind("<Leave>",lea)
   tc.bind("<ButtonPress-1>",terms_n_conditions)

   Button(rfrm,text="Register
Now",bg="lime",fg='white',activebackground="#32CD32",activeforeground="#fff",font="arial 13
bold",width=20,height=2,command=register).place(x=50,y=560)

   rfrm.place(x=120,y=80,width=1000,height=620)

#admin login
alog=Label(root,text="Admin Login",font="comicsans 30 bold",bg="white",fg="blue",width=10).place(x=170,y=300)

anm.set("")
apswd.set("")

nm=Label(f1,text="Username : ",bg='blue',fg='#fff',font="comicsans 20").grid(row=1,column=0,pady=20)
txt_nm=Entry(f1,bd=2,relief=GROOVE,font="comicsans 20",textvariable=anm).grid(row=1,column=1,padx=10)

psd=Label(f1,text="Password : ",bg='blue',fg='white',font="comicsans 20").grid(row=2,column=0)
txt_psd=Entry(f1,bd=2,relief=GROOVE,font="comicsans
20",textvariable=apswd,show="*").grid(row=2,column=1,padx=10)

nfoc=Label(f1,font="arial 15 bold",bg='blue')
nfoc.grid(row=3,column=1)

#admin login function
def login_config():
      if anm.get()=="" or apswd.get()=="":
         nfoc.config(fg="red",text="All fields are required*")
         nfoc.update()
         time.sleep(1)
         nfoc.config(text="")
      elif anm.get()=="Admin" and apswd.get()=="admin123":
         anm.set("")
         apswd.set("")
         global a_img
         global a_ig
         global img
         global r
         r=Toplevel(root)
         r.config(bg="lightyellow")
         r.title("ADMIN HOME PAGE")
         r.geometry(str(int(r.winfo_screenwidth()-20))+'x'+str(r.winfo_screenheight()-70)+"+2+0")

         adminimg=Image.open('icon.jpg')
         adminimg=adminimg.resize((550,450))
```

```
        a_img=ImageTk.PhotoImage(adminimg)
        Label(r,image=a_img).place(y=70,x=0)

        adminig=Image.open('admin.png')
        adminig=adminig.resize((450,130))
        a_ig=ImageTk.PhotoImage(adminig)
        Label(r,image=a_ig).place(y=520,x=50)

        frame=Frame(r,bg='navyblue')
        frame.pack(fill=X)

        im=Image.open('Untitled.png')
        im=im.resize((100,50))
        img=ImageTk.PhotoImage(im)
        Label(frame,image=img).pack(side=LEFT,pady=5,padx=10,ipadx=10)

        Label(r,text="HOME PAGE",font="comicsans 35 bold",bg='navy',fg='white').place(x=480)
        #buttons
        Button(r,text="Account Details",command=acc_details,font="comicsans 20
bold",bg="dodgerblue",fg='white',activebackground="deepskyblue",activeforeground="#fff").place(x=610,y=90,widt
h=380,height=75)
        Button(r,text="User Details",font="comicsans 20
bold",command=user_details,bg="dodgerblue",fg='white',activebackground="red",activeforeground="#fff").place(x=
850,y=200,width=380,height=75)
        Button(r,text="Track Account Transactions",font="comicsans 20
bold",command=acc_stats,bg="dodgerblue",fg='white',activebackground="lime",activeforeground="#fff").place(x=6
10,y=310,width=380,height=75)
        Button(r,text="Record of Fixed Deposits",command=records_FD,font="comicsans 20
bold",bg="dodgerblue",fg='white',activebackground="cyan",activeforeground="#fff").place(x=850,y=420,width=380,
height=75)
        Button(r,text="Notifications",command=bank,font="comicsans 20
bold",bg="dodgerblue",fg='white',activebackground="cyan",activeforeground="#fff").place(x=610,y=530,width=380,
height=75)

        Button(frame,text="Logout",fg='#fff',bg='red',height=1,command=r.destroy,font="comicsans 12 bold
",activeforeground="#000",activebackground="#fff").pack(side=RIGHT,pady=5,padx=10,ipadx=10)
      else:
        nfoc.config(fg='red',text="Incorrect Username or Password*")
        nfoc.update()
        time.sleep(1)
        nfoc.config(text="")
Button(f1,text="Login",font="comicsans 15",command=login_config).grid(row=3,column=0,pady=20)
#user login

ulog=Label(root,text="User Login",font="comicsans 30 bold",fg="blue",bg='white',width=10).place(x=820,y=300)

anm.set("")
apswd.set("")

Button(root,text="Create a new account",font="Calibri 25
bold",command=create_acc,width=30).place(x=750,y=380)
```

```
Label(root,text="OR",font="comicsans 15 bold",bg='blue',fg='white').place(x=950,y=450)
Label(root,text="Login to existing account",font="comicsans 15 bold",bg="blue",fg='white').place(x=820,y=480)

unm=Label(f2,text="Username : ",bg='blue',fg='#fff',font="comicsans 20").grid(row=1,column=0,pady=20)
txt_unm=Entry(f2,bd=2,relief=GROOVE,font="comicsans 20",textvariable=usnm)
txt_unm.grid(row=1,column=1,padx=10)

upsd=Label(f2,text="Password : ",bg='blue',fg='white',font="comicsans 20").grid(row=2,column=0)
txt_upsd=Entry(f2,bd=2,relief=GROOVE,font="comicsans 20",textvariable=upswd,show="*")
txt_upsd.grid(row=2,column=1,padx=10)

nfic=Label(f2,font="arial 15 bold",bg='blue')
nfic.grid(row=3,column=1)
#login function
def ulogin_config():
    if usnm.get()=="" or upswd.get()=="":
      nfic.config(fg="red",text="All fields are required")
      nfic.update()
      time.sleep(1)
      nfic.config(text="")
    elif True:
     try:
      if d[upswd.get()]==usnm.get():
        per_user_details()
        usnm.set("")
        upswd.set("")
        if accstus=="Inactive":
               inmsg=messagebox.showerror("ACCOUNT DISABLED","Your account has been disabled.\nFor more
information Contact us")
               if inmsg=="ok":
                       contact()
        else:
             global us
             global tablename
             tablename="table"+str(acountno)
             us=Toplevel(root)
             us.title("WELCOME TO ALL CITIZENS BANK")
             us.config(bg='#F5F5F5')
             wid=str(us.winfo_screenwidth()-25)
             hei=str(us.winfo_screenheight()-70)
             us.geometry(wid+'x'+hei+"+5+0")
             #frames
             fr1=Frame(us,bg='#FFF')
             fr1.place(width=500,height=200,x=30,y=100)

             fr2=Frame(us,bg='light pink')
             fr2.place(width=500,height=320,x=30,y=330)

             fr3=Frame(us,bg="#3FC1C9")
             fr3.place(width=320,height=260,x=570,y=100)
             fr4=Frame(us,bg="#3FC1C9")
             fr4.place(width=320,height=260,x=910,y=100)

             fr5=Frame(us,bg="#364F6B")
             fr5.place(width=320,height=260,x=570,y=390)

             fr6=Frame(us,bg="#364F6B")
             fr6.place(width=320,height=260,x=910,y=390)

             global clock
             clock=Label(fr2,font=("DejaVu Sans",17,"bold"))
             clock.place(x=180,y=10)
             tick()
             #menu frame
             global img
             global mymenu
             global dis
             frame=Frame(us,bg='blue')
             frame.pack(fill=X)

             im=Image.open('Untitled.png')
             im=im.resize((100,50))
             img=ImageTk.PhotoImage(im)
             Label(frame,image=img).pack(side=LEFT,pady=5,padx=10,ipadx=10)

             Button(frame,text="Home",bg='#000',fg='#fff',height=1,font="comicsans 12
",width=10,activebackground="#000",activeforeground="#fff").pack(side=LEFT,pady=5,padx=10,ipadx=10)

             mymenu=ttk.Combobox(frame,font="comicsans 14
",state='readonly',justify=CENTER,width=13,values=("e-STDR","Close A/C Prematurely"))
             mymenu.set("Fixed Deposits")
             us.option_add('*TCombobox*Listbox.font',"comicsans 13")
             mymenu.bind("<<ComboboxSelected>>",select)
             mymenu.pack(side=LEFT,pady=15,padx=10,ipadx=5)
             Button(frame,text="Notifications",fg='#000',bg='#fff',command=notice,height=1,font="comicsans 12
",activeforeground="#fff",activebackground="#000").pack(side=LEFT,pady=5,padx=10,ipadx=10)
             Button(frame,text="For Enquiries Contact
Us",fg='#000',bg='#fff',command=contact,height=1,font="comicsans 12
",activeforeground="#fff",activebackground="#000").pack(side=LEFT,pady=5,padx=10,ipadx=10)
             Button(frame,text="Logout",fg='#000',bg='#fff',height=1,command=us.destroy,font="comicsans 12
",activeforeground="#fff",activebackground="red").pack(side=RIGHT,pady=5,padx=10,ipadx=10)

             #Buttons

             Button(fr3,text="Fund Transfer ",font=("DejaVu
Sans",20),command=fund).pack(side=BOTTOM,fill=X,padx=15,pady=10)

             Button(fr4,text="Bill Payments",font=("DejaVu
Sans",20),command=bill).pack(side=BOTTOM,fill=X,padx=15,pady=10)
```

```
        Button(fr5,text="Account Info",font=("DejaVu
Sans",20),command=accinfo).pack(side=BOTTOM,fill=X,padx=15,pady=10)

        Button(fr6,text="Account Statement",font=("DejaVu
Sans",20),command=state).pack(side=BOTTOM,fill=X,padx=15,pady=10)
        #labels

        Label(fr1,text="Welcome to All Citizens Bank",bg="white",font=("DejaVu Sans",25)).pack()

        Label(fr1,text="The most secure bank you have probably used",bg="white",font=("DejaVu
Sans",12)).pack()

        Label(fr1,text="Ac No. :",bg="white",font=("DejaVu Sans",15)).place(x=10,y=100)
        Label(fr1,text=acountno,bg="#fff",font=("DejaVu Sans",15)).place(x=150,y=100)
        Label(fr1,text="Username :",bg="white",font=("DejaVu Sans",15)).place(x=10,y=150)
        Label(fr1,text=user,bg="#fff",font=("DejaVu Sans",15)).place(x=150,y=150)

        Label(fr2,text=datetime.date.today(),font=("DejaVu Sans",17,"bold")).place(x=10,y=10,width=150)

        Label(fr2,text="Transaction Account Balance :",bg="light pink",font=("DejaVu
Sans",20,"bold")).place(x=10,y=100)

        #Entries

        Entry(fr2,font="comicsans 25",bd=5,state='readonly').place(x=50,y=170)

        dis=Label(fr2,font="comicsans 24",text=blnc)
        dis.place(x=150,y=175)
        #images
        global i2
        global i4
        global i6
        global i8
        i1=Image.open("transfer.png")
        i1=i1.resize((300,200))
        i2=ImageTk.PhotoImage(i1)
        Label(fr3,image=i2).pack(pady=10)

        i3=Image.open("images.png")
        i3=i3.resize((300,200))
        i4=ImageTk.PhotoImage(i3)
        Label(fr4,image=i4).pack(pady=10)


        i5=Image.open("info.jpg")
        i5=i5.resize((300,200))
        i6=ImageTk.PhotoImage(i5)
        Label(fr5,image=i6).pack(pady=10)

        i7=Image.open("statement.jpg")
        i7=i7.resize((300,200))
```

```
             i8=ImageTk.PhotoImage(i7)
             Label(fr6,image=i8).pack(pady=10)
      else:
       nfic.config(fg="red",text="Incorrect Username or Password*")
       nfic.update()
       time.sleep(1)
       nfic.config(text="")
     except:
       nfic.config(fg="red",text="Incorrect Username or Password*")
       nfic.update()
       time.sleep(1)
       nfic.config(text="")

#USER CONTACT OPTION
def contact():
          global txt
          global caicon
          global loicon
          global cntc
          cntc=Toplevel(root)
          cntc.title("CONTACT US")
          cntc.geometry("750x300+"+str(int(cntc.winfo_screenwidth()/2-
350))+'+'+str(int(cntc.winfo_screenheight()/2-300)))
          cntc.maxsize(750,300)
          cntc.minsize(750,300)
          cntc.config(bg="#fff")
          #contact frames
          cfra1=Frame(cntc,bg="#000")
          cfra1.place(height=300,width=300,x=0,y=0)
          cfra2=Frame(cntc,bg="limegreen")
          cfra2.place(height=300,width=450,x=300,y=0)
          #cfra1 content
          ##images
          cicon=Image.open("call.jpg")
          cicon=cicon.resize((50,50))
          caicon=ImageTk.PhotoImage(cicon)
          Label(cfra1,image=caicon).place(x=5,y=30)
          licon=Image.open("locate.jpg")
          licon=licon.resize((50,50))
          loicon=ImageTk.PhotoImage(licon)
          Label(cfra1,image=loicon).place(x=5,y=180)
          ##content
          Label(cfra1,text="Give us a ring",font=("Times New Roman",15
,"bold"),fg="#fff",bg="#000").place(x=65,y=10)
          Label(cfra1,text='''All Citizens Bank
  +91 9012938492
  Mon-Fri,8:00-22:00''',bg="#000",fg="#fff",font="comicsans 12").place(y=40,x=65)
          Label(cfra1,text="Find us at the office",font=("Times New Roman",15
,"bold"),fg="#fff",bg="#000").place(x=65,y=160)
          Label(cfra1,text='''Block B 3A,Janakpuri,
  New Delhi
```

```
   Delhi-110058''',fg="#fff",bg="#000",font="comicsans 12").place(x=65,y=190)

               #cfra2 content
               Label(cfra2,text="Send a message",font="comicsans 15 bold",fg="#fff",bg="limegreen").place(x=5,y=10)
               txt=Text(cfra2,font="calibri 15")
               txt.place(height=200,width=440,x=5,y=50)
               Label(cfra2,text="*Please write the message in single line.",bg="limegreen").place(x=5,y=255)
               Label(cfra2,text="Don't terminate it using Enter key.",bg="limegreen").place(x=10,y=275)

               Button(cfra2,text="Send",command=write,font="comicsans
12",bg="mediumblue",fg="#fff",activebackground="blue",activeforeground="#fff").place(x=360,y=260,width=75)
#INTEREST RATE TABLE
def domestic(event):
   rot=Toplevel(tdr)
   rot.geometry('1250x600+'+str(int(rot.winfo_screenwidth()/2-1250/2))+'+'+str(int(rot.winfo_screenheight()/2-
700/2)))
   rot.minsize(1110,600)
   #rot.maxsize(1110,600)
   rot.config(bg="#fff")
   rot.title('DOMESTIC INTEREST')

   fr=Frame(rot,bg="#fff")
   fr.place(x=5,y=5,width=1250,height=580)
   Label(fr,text="Revised in Interest Rates On Domestic Term Deposis w.e.f. 20.06.2020",font="comicsans
20",bg="#fff").grid(row=0,columnspan=7)
   #headings
   Label(fr,text="Tenors",bg="deepskyblue",font="comicsans
15",fg="#fff",width=20).grid(row=1,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text='''Existing Public
        w.e.f. 29.04.2020''',bg="deepskyblue",font="comicsans 15",fg="#fff").grid(row=1,column=2,padx=1,ipadx=5)
   Label(fr,text='''Revised Public
        w.e.f. 20.06.2020''',bg="deepskyblue",font="comicsans 15",fg="#fff").grid(row=1,column=3,padx=1,ipadx=5)
   Label(fr,text='''Existing for Senior Citizens
   w.e.f. 29.04.2020''',bg="deepskyblue",font="comicsans 15",fg="#fff").grid(row=1,column=4,padx=1,ipadx=5)
   Label(fr,text='''Revised for Senior Citizens
   w.e.f. 20.06.2020''',bg="deepskyblue",font="comicsans 15",fg="#fff").grid(row=1,column=5,padx=1,ipadx=5)
   #Column 1
   Label(fr,text="8 days to 45 days",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=2,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="46 days to 179 days",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=3,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="180 days to 210 days",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=4,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="211 days to less than 1 year",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=5,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="1 year",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=6,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="Above 1 year to 455 days",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=7,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="456 days to less than 2 years",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=8,column=1,padx=1,ipady=11,ipadx=5)
```

```
   Label(fr,text="2 years to less than 3 years",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=9,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="3 years to less than 5 years",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=10,column=1,padx=1,ipady=11,ipadx=5)
   Label(fr,text="5 years and up to 10 years",bd=2,relief=GROOVE,bg="lightblue",font="comicsans
13",fg="#000",width=25).grid(row=11,column=1,padx=1,ipady=11,ipadx=5)
   #Column 2
   Label(fr,text="5.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=2,column=2,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=3,column=2,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=4,column=2,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=5,column=2,padx=1,ipadx=7)
   Label(fr,text="6.90",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=6,column=2,padx=1,ipadx=7)
   Label(fr,text="6.90",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=7,column=2,padx=1,ipadx=7)
   Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=8,column=2,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=9,column=2,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=10,column=2,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=11,column=2,padx=1,ipadx=7)
   #Column 3
   Label(fr,text="5.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=2,column=3,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=3,column=3,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=4,column=3,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=5,column=3,padx=1,ipadx=7)
   Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=6,column=3,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=7,column=3,padx=1,ipadx=7)
   Label(fr,text="6.50",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=8,column=3,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=9,column=3,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=10,column=3,padx=1,ipadx=7)
   Label(fr,text="6.25",bg="lightblue",font="calibri
13",width=25,height=2,bd=2,relief=GROOVE,).grid(row=11,column=3,padx=1,ipadx=7)
   #Column 4
   Label(fr,text="6.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=2,column=4,padx=1,ipadx=5)
```

```
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=3,column=4,padx=1,ipadx=5)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=4,column=4,padx=1,ipadx=5)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=5,column=4,padx=1,ipadx=5)
    Label(fr,text="7.25",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=6,column=4,padx=1,ipadx=5)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=7,column=4,padx=1,ipadx=5)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=8,column=4,padx=1,ipadx=5)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=9,column=4,padx=1,ipadx=5)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=10,column=4,padx=1,ipadx=5)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=11,column=4,padx=1,ipadx=5)
    #Column 5
    Label(fr,text="6.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=2,column=5,padx=1,ipadx=6)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=3,column=5,padx=1,ipadx=6)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=4,column=5,padx=1,ipadx=6)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=5,column=5,padx=1,ipadx=6)
    Label(fr,text="7.25",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=6,column=5,padx=1,ipadx=6)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=7,column=5,padx=1,ipadx=6)
    Label(fr,text="7.00",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=8,column=5,padx=1,ipadx=6)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=9,column=5,padx=1,ipadx=6)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=10,column=5,padx=1,ipadx=6)
    Label(fr,text="6.75",bg="lightblue",font="calibri
13",width=26,height=2,bd=2,relief=GROOVE,).grid(row=11,column=5,padx=1,ipadx=6)

#Fixed Deposits
def select(event):
    global tdr
    if mymenu.get()=="e-STDR":
        mymenu.set("Fixed Deposits")
        tdr=Toplevel(us)
        width=tdr.winfo_screenwidth()
        height=tdr.winfo_screenheight()

        tdr.geometry(str(width-10)+'x700+2+0')
        tdr.title("e-STDR(FIXED DEPOSIT)")
```

```
def submit():

  if amount.get()=="":
      messagebox.showinfo("Error","Please enter the amount for Fixed Deposit",parent=tdr)
  elif int(amount.get())>int(blnc):
    messagebox.showerror("ERROR","Insufficient Balance",parent=tdr)

  elif int(amount.get()) < 1000:
    messagebox.showinfo("Error","Please enter amount above Rs 999",parent=tdr)
    amount.set("")
  elif int(amount.get()) > 40000:
    messagebox.showinfo("Error","Maximum Limit is Rs 40000, Please enter amount less than this",parent=tdr)
    amount.set("")
  elif rad.get()==" ":
      messagebox.showinfo("Error","Please choose the Type of Deposit",parent=tdr)
  elif tenure.get()==" ":
      messagebox.showinfo("Error","Please choose the Type of Tenure of Deposit",parent=tdr)
  elif (tenure.get()=="Day(s)" and tenuredays.get()==0) or (tenure.get()=="Year(s) / Month(s) / Day(s)" and tyears.get()==0 and tmonths.get()==0 and tdays.get()==0):
      messagebox.showinfo("Error","Please choose Tenure of Deposit",parent=tdr)
  elif (tenure.get()=="Year(s) / Month(s) / Day(s)" and tyears.get()==0 and tmonths.get()==0 and tdays.get()!=0):
      messagebox.showinfo("Error","If your tenure period is less than 1 month then please choose Days option in tenure of deposit",parent=tdr)
  elif (tenure.get()=="Year(s) / Month(s) / Day(s)" and tmonths.get()==12):
      messagebox.showinfo("Error","Please choose year instead of 12 months",parent=tdr)
  elif tnc.get()==0:
      messagebox.showinfo("Terms and Conditions","Please Agree to our Terms & Conditions",parent=tdr)
  else:

    if sen.get()==0:
        n="No"
    else:
        n="Yes"

    verify=Frame(tdr,bg="#fff")
    verify.place(y=40,x=0,width=1250,height=690)

    l2=Label(verify,text="Verify the details and click confirm",font="comicsans 17",fg="navyblue",bg="white")
    l2.place(x=60,y=35)

    Label(verify,text="Debit Account No.",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=85)
    Label(verify,text=acountno,font="comicsans 15",fg="#000",bg="white").place(x=465,y=85)

    Label(verify,text="Account Type",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=120)
    Label(verify,text=acctyp,font="comicsans 15",fg="black",bg="white").place(x=465,y=120)

    Label(verify,text="Amount",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=155)
    Label(verify,text=amount.get(),font="comicsans 15",fg="black",bg="white").place(x=465,y=155)

    Label(verify,text="Type of Deposit",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=190)
```

```
        Label(verify,text=rad.get(),font="comicsans 15",fg="black",bg="white").place(x=465,y=190)
        o=datetime.datetime.now()
        if o.month in [1,3,5,7,8,10,12]:
          eb=31
        elif o.month in [4,6,9,11]:
          eb=30
        elif o.month in [2]:
          if o.year/4==0:
            eb=29
          else:
            eb=28

        Label(verify,text="Tenure",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=225)
        Label(verify,text="Maturity Date",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=400)
        if tenure.get()=="Day(s)":
          Label(verify,text=(tenuredays.get(),tenure.get()),font="comicsans
15",fg="#000",bg="white").place(x=465,y=225)
          if tenuredays.get()+o.day<=eb:
            ad=tenuredays.get()+o.day
            ye=o.month
            ey=o.year

          elif tenuredays.get()+o.day>eb:
            ad=tenuredays.get()+o.day-eb
            ye=o.month+1
            if ye==13:
              ye=1
              ey=o.year+1
            else:
              ey=o.year
          Label(verify,text=(ey,ye,ad),font="comicsans 15",fg="black",bg="white").place(x=465,y=400)

        else:
          Label(verify,text=(str(tyears.get())+" Year(s) "+ str(tmonths.get())+" Month(s) "+ str(tdays.get())+" Day(s)
"),font="comicsans 15",fg="#000",bg="white").place(x=465,y=225)
          if tmonths.get()==0 and tdays.get()==0:
            ye=o.month
            ad=o.day
            ey=o.year+tyears.get()
          elif tmonths.get()==0 and tdays.get()!=0:
              if (o.day+tdays.get())>eb:
                  ye=1
                  ey=o.year+tyears.get()+1
                  ad=(o.day+tdays.get())-eb
              elif (o.day+tdays.get())<=eb:
                ad=o.day+tdays.get()
                ye=o.month
                ey=o.year+tyears.get()
          elif tmonths.get()!=0 and tdays.get()==0:
            if tyears.get()==0:
              if tmonths.get()+o.month>12:
```

```
            ey=o.year+1
            ye=tmonths.get()+o.month-12
            ad=o.day
        else:
            ey=o.year
            ye=o.month+tmonths.get()
            ad=o.day
    else:
        if tmonths.get()+o.month>12:
            ey=o.year+tyears.get()+1
            ye=tmonths.get()+o.month-12
            ad=o.day
        else:
            ey=o.year_tyears.get()
            ye=o.month+tmonths.get()
            ad=o.day
elif tmonths.get()!=0 and tdays.get()!=0:
    if tyears.get()==0:
        if tmonths.get()+o.month>12:
            if tdays.get()+o.day<=eb:
                ad=tdays.get()+o.day
                ye=tmonths.get()+o.month-12
                ey=o.year+1
            elif tdays.get()+o.day>eb:
                ad=tdays.get()+o.day-eb
                ye=tmonths.get()+o.month-12+1
                if ye==13:
                    ye=1
                    ey=o.year+2
                else:
                    ey=o.year+1
        else:
            if tdays.get()+o.day<=eb:
                ad=tdays.get()+o.day
                ye=tmonths.get()+o.month
                ey=o.year
            elif tdays.get()+o.day>eb:
                ad=tdays.get()+o.day-eb
                ye=tmonths.get()+o.month+1
                if ye==13:
                    ye=1
                    ey=o.year+1
                else:
                    ey=o.year
    elif tyears.get()!=0:
        if tmonths.get()+o.month>12:
            if tdays.get()+o.day<=eb:
                ad=tdays.get()+o.day
                ye=tmonths.get()+o.month-12
                ey=o.year+tyears.get()+1
            elif tdays.get()+o.day>eb:
                    ad=tdays.get()+o.day-eb
                    ye=tmonths.get()+o.month-12+1
                    if ye==13:
                        ye=1
                        ey=o.year+tyears.get()+2
                    else:
                        ey=o.year+tyears.get()+1
            else:
                if tdays.get()+o.day<=eb:
                    ad=tdays.get()+o.day
                    ye=tmonths.get()+o.month
                    ey=o.year+tyears.get()
                elif tdays.get()+o.day>eb:
                    ad=tdays.get()+o.day-eb
                    ye=tmonths.get()+o.month+1
                    if ye==13:
                        ye=1
                        ey=o.year+tyears.get()+1
                    else:
                        ey=o.year+tyears.get()
    Label(verify,text=(ey,ye,ad),font="comicsans 15",fg="black",bg="white").place(x=465,y=400)

Label(verify,text="Mode of Operation",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=260)
Label(verify,text="Single",font="comicsans 15",fg="black",bg="white").place(x=465,y=260)

Label(verify,text="Interest Rate Type",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=295)
Label(verify,text="Fixed Rate",font="comicsans 15",fg="black",bg="white").place(x=465,y=295)

if sen.get()==0 :
    if (tenuredays.get()>0) and (tenuredays.get()<=7):
        ROI="0%"
        fd_amount=amount.get()
    elif (tenuredays.get()>0) and (tenuredays.get()>7):
        ROI="5.50%"
        fd_amount=int(amount.get())+(int(amount.get())*0.055)
    elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
        ROI="6.50%"
        fd_amount=int(amount.get())+(int(amount.get())*0.065)
    elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
        ROI="6.75%"
        fd_amount=int(amount.get())+(int(amount.get())*0.0675)
    elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
        ROI="6.50%"
        fd_amount=int(amount.get())+(int(amount.get())*0.065)
    elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
        ROI="6.25%"
        fd_amount=int(amount.get())+(int(amount.get())*0.0625)
    Label(verify,text=ROI,font="comicsans 15",fg="black",bg="white").place(x=465,y=330)
    Label(verify,text=int(fd_amount),font="comicsans 15",fg="black",bg="white").place(x=465,y=365)
elif sen.get() !=0 and int(amount.get())>9999:
    if (tenuredays.get()>0) and (tenuredays.get()<=7):
```

```
                ROI="0%"
                fd_amount=int(amount.get())
            elif (tenuredays.get()>0) and (tenuredays.get()>7):
                ROI="6.00%"
                fd_amount=int(amount.get())+(int(amount.get())*0.06)
            elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
                ROI="7.00%"
                fd_amount=int(amount.get())+(int(amount.get())*0.07)
            elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
                ROI="7.25%"
                fd_amount=int(amount.get())+(int(amount.get())*0.0725)
            elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
                ROI="7.00%"
                fd_amount=int(amount.get())+(int(amount.get())*0.07)
            elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
                ROI="6.75%"
                fd_amount=int(amount.get())+(int(amount.get())*0.0675)
            Label(verify,text=ROI,font="comicsans 15",fg="black",bg="white").place(x=465,y=330)
            Label(verify,text=int(fd_amount),font="comicsans 15",fg="black",bg="white").place(x=465,y=365)
        else:
            if (tenuredays.get()>0) and (tenuredays.get()<=7):
                ROI="0%"
                fd_amount=int(amount.get())
            elif (tenuredays.get()>0) and (tenuredays.get()>7):
                ROI="5.50%"
                fd_amount=int(amount.get())+(int(amount.get())*0.055)
            elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
                ROI="6.50%"
                fd_amount=int(amount.get())+(int(amount.get())*0.065)
            elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
                ROI="6.75%"
                fd_amount=int(amount.get())+(int(amount.get())*0.0675)
            elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
                ROI="6.50%"
                fd_amount=int(amount.get())+(int(amount.get())*0.065)
            elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
                ROI="6.25%"
                fd_amount=int(amount.get())+(int(amount.get())*0.0625)
            Label(verify,text=ROI,font="comicsans 15 ",fg="black",bg="white").place(x=465,y=330)
            Label(verify,text=int(fd_amount),font="comicsans 15",fg="black",bg="white").place(x=465,y=365)
        Label(verify,text="Rate Of Interest",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=330)
        Label(verify,text="Maturity Amount",font="comicsans 15 bold",fg="black",bg="white").place(x=65,y=365)

        Label(verify,text="Senior Citizen Option Chosen",font="comicsans 15
bold",fg="black",bg="white").place(x=65,y=435)
        Label(verify,text=n,font="comicsans 15",fg="black",bg="white").place(x=465,y=435)

        Label(verify,text="Selected Maturity Instruction Details",font="comicsans 17
bold",fg="navyblue",bg="white").place(x=60,y=480)
        Label(verify,text="Selected Maturity Instruction Details",font="comicsans 16
bold",fg="#000",bg="white").place(x=60,y=510)
```

```
         Label(verify,text="Repay Principal and Interest",font="calibri 15 ",bg="#fff").place(x=465,y=510)

         Button(verify,text="Confirm",command=confirm,font="comicsans
15",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=5,pady=2).place(x=500,y=
580)
         Button(verify,text="Back",command=verify.destroy,font="comicsans
15",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=580)

      def fdshow():
         mysql.execute("select * from Fixed_Deposit")
         fdy=mysql.fetchall()
         fdlist1=[]
         for i in fdy:
           fdx=i[0]
           fdlist1.append(fdx)
         return fdlist1
      def fd_account_num():
         global fd_a_acno
         fdn=random.randint(1000000000,9999999999)
         fd_ch=show()
         if fdn not in fd_ch:
           fd_a_acno=fdn
           return fdn
         else:
            fd_account_num()
      def confirm():
         utrno()
         if sen.get()==0:
            n="No"
         else:
            n="Yes"

         fd_account_num()

         fra3=Frame(tdr,bg="lightgreen")
         fra3.place(y=40,x=0,width=1250,height=690)

         Label(fra3,text="Your request for a new e-STDR (Cumulative) is created on ",font="comicsans
16",fg="black",bg="lightgreen").place(x=65,y=30)
         Label(fra3,text=datetime.datetime.now(),font="comicsans 16",fg="black",bg="lightgreen").place(x=645,y=30)

         Label(fra3,text="Please note this transaction number for future reference : ",font="comicsans
16",fg="black",bg="lightgreen").place(x=65,y=70)
         Label(fra3,text=utr,font="comicsans 16",fg="black",bg="lightgreen").place(x=665,y=70)

         Label(fra3,text="Details of Deposit Account",font="comicsans 17
bold",fg="navyblue",bg="lightgreen").place(x=130,y=120)

         Label(fra3,text="e-STDR A/C No.",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=165)
         Label(fra3,text=fd_a_acno,font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=165)
```

```
        Label(fra3,text="Debit Account No.",font="comicsans 15
bold",fg="black",bg="lightgreen").place(x=130,y=200)
        Label(fra3,text=acountno,font="comicsans 15",fg="#000",bg="lightgreen").place(x=465,y=200)

        Label(fra3,text="Name",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=235)
        Label(fra3,text=user,font="comicsans 15 ",bg="lightgreen").place(x=465,y=235)

        Label(fra3,text="Amount",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=270)
        Label(fra3,text=amount.get(),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=270)

        Label(fra3,text="Type of Deposit",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=305)
        Label(fra3,text=rad.get(),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=305)

        o=datetime.datetime.now()
        if o.month in [1,3,5,7,8,10,12]:
           eb=31
        elif o.month in [4,6,9,11]:
           eb=30
        elif o.month in [2]:
           if o.year/4==0:
              eb=29
           else:
              eb=28

        Label(fra3,text="Tenure",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=340)
        if tenure.get()=="Day(s)":
           ymd=(tenuredays.get(),tenure.get())
           Label(fra3,text=(tenuredays.get(),tenure.get()),font="comicsans
15",fg="#000",bg="lightgreen").place(x=465,y=340)
           if tenuredays.get()+o.day<=eb:
              ad=tenuredays.get()+o.day
              ye=o.month
              ey=o.year

           elif tenuredays.get()+o.day>eb:
              ad=tenuredays.get()+o.day-eb
              ye=o.month+1
              if ye==13:
                 ye=1
                 ey=o.year+1
              else:
                 ey=o.year
           Label(fra3,text=(ey,ye,ad),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=515)

        else:
           ymd=(str(tyears.get())+" Year(s) "+ str(tmonths.get())+" Month(s) "+ str(tdays.get())+" Day(s) ")
           Label(fra3,text=(str(tyears.get())+" Year(s) "+ str(tmonths.get())+" Month(s) "+ str(tdays.get())+" Day(s)
"),font="comicsans 15",fg="#000",bg="lightgreen").place(x=465,y=340)
           if tmonths.get()==0 and tdays.get()==0:
              ye=o.month
```

```
   ad=o.day
   ey=o.year+tyears.get()
elif tmonths.get()==0 and tdays.get()!=0:
   #if tdays.get()<o.day:
      if (o.day+tdays.get())>eb:
        # if ye+1>12:
            ye=1
            ey=o.year+tyears.get()+1
            ad=(o.day+tdays.get())-eb
         #else:
            #ad=(o.day+tdays.get())-eb
            #ye=+1
      elif (o.day+tdays.get())<=eb:
         ad=o.day+tdays.get()
         ye=o.month
         ey=o.year+tyears.get()
      '''elif tdays.get()>=o.day:
      if ye+1>12:
            ye=(ye+1)-12
            ey=o.year+1
            ad=(o.day+tdays.get())-eb
      else:
         ad=(o.day+tdays.get())-eb
         ye=+1'''
elif tmonths.get()!=0 and tdays.get()==0:
   if tyears.get()==0:
      if tmonths.get()+o.month>12:
         ey=o.year+1
         ye=tmonths.get()+o.month-12
         ad=o.day
      else:
         ey=o.year
         ye=o.month+tmonths.get()
         ad=o.day
   else:
      if tmonths.get()+o.month>12:
         ey=o.year+tyears.get()+1
         ye=tmonths.get()+o.month-12
         ad=o.day
      else:
         ey=o.year_tyears.get()
         ye=o.month+tmonths.get()
         ad=o.day
elif tmonths.get()!=0 and tdays.get()!=0:
   if tyears.get()==0:
      if tmonths.get()+o.month>12:
         if tdays.get()+o.day<=eb:
            ad=tdays.get()+o.day
            ye=tmonths.get()+o.month-12
            ey=o.year+1
         elif tdays.get()+o.day>eb:
            ad=tdays.get()+o.day-eb
            ye=tmonths.get()+o.month-12+1
            if ye==13:
                ye=1
                ey=o.year+2
            else:
                ey=o.year+1
      else:
         if tdays.get()+o.day<=eb:
            ad=tdays.get()+o.day
            ye=tmonths.get()+o.month
            ey=o.year
         elif tdays.get()+o.day>eb:
            ad=tdays.get()+o.day-eb
            ye=tmonths.get()+o.month+1
            if ye==13:
                ye=1
                ey=o.year+1
            else:
                ey=o.year
   elif tyears.get()!=0:
      if tmonths.get()+o.month>12:
         if tdays.get()+o.day<=eb:
            ad=tdays.get()+o.day
            ye=tmonths.get()+o.month-12
            ey=o.year+tyears.get()+1
         elif tdays.get()+o.day>eb:
            ad=tdays.get()+o.day-eb
            ye=tmonths.get()+o.month-12+1
            if ye==13:
                ye=1
                ey=o.year+tyears.get()+2
            else:
                ey=o.year+tyears.get()+1
      else:
         if tdays.get()+o.day<=eb:
            ad=tdays.get()+o.day
            ye=tmonths.get()+o.month
            ey=o.year+tyears.get()
         elif tdays.get()+o.day>eb:
            ad=tdays.get()+o.day-eb
            ye=tmonths.get()+o.month+1
            if ye==13:
                ye=1
                ey=o.year+tyears.get()+1
            else:
                ey=o.year+tyears.get()

Label(fra3,text=(ey,ye,ad),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=515)
```

```
        Label(fra3,text="Mode of Operation",font="comicsans 15
bold",fg="black",bg="lightgreen").place(x=130,y=375)
        Label(fra3,text="Single",font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=375)

        Label(fra3,text="Interest Rate Type",font="comicsans 15
bold",fg="black",bg="lightgreen").place(x=130,y=410)
        Label(fra3,text="Fixed Rate",font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=410)

        Label(fra3,text="Rate Of Interest",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=445)
        if sen.get()==0 :
          if (tenuredays.get()>0) and (tenuredays.get()<=7):
              ROI="0%"
              fd_amount=amount.get()
          elif (tenuredays.get()>0) and (tenuredays.get()>7):
              ROI="5.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.055)
          elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
              ROI="6.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.065)
          elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
              ROI="6.75%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0675)
          elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="6.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.065)
          elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="6.25%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0625)
          Label(fra3,text=ROI,font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=445)
          Label(fra3,text=int(fd_amount),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=480)
        elif sen.get() !=0 and int(amount.get())>9999:
          if (tenuredays.get()>0) and (tenuredays.get()<=7):
              ROI="0%"
              fd_amount=int(amount.get())
          elif (tenuredays.get()>0) and (tenuredays.get()>7):
              ROI="6.00%"
              fd_amount=int(amount.get())+(int(amount.get())*0.06)
          elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
              ROI="7.00%"
              fd_amount=int(amount.get())+(int(amount.get())*0.07)
          elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
              ROI="7.25%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0725)
          elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="7.00%"
              fd_amount=int(amount.get())+(int(amount.get())*0.07)
          elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="6.75%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0675)
          Label(fra3,text=ROI,font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=445)
          Label(fra3,text=int(fd_amount),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=480)
        else:
          if (tenuredays.get()>0) and (tenuredays.get()<=7):
              ROI="0%"
              fd_amount=int(amount.get())
          elif (tenuredays.get()>0) and (tenuredays.get()>7):
              ROI="5.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.055)
          elif ((tyears.get()==0) and (tmonths.get()>0) and (tmonths.get()<12) and (tdays.get()>=0)):
              ROI="6.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.065)
          elif ((tyears.get()==1) and (tmonths.get()==0) and (tdays.get()==0)):
              ROI="6.75%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0675)
          elif ((tyears.get()==1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="6.50%"
              fd_amount=int(amount.get())+(int(amount.get())*0.065)
          elif ((tyears.get()>1) and (tmonths.get()>=0) and (tmonths.get()<=12) and (tdays.get()>=0)):
              ROI="6.25%"
              fd_amount=int(amount.get())+(int(amount.get())*0.0625)
          Label(fra3,text=ROI,font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=445)
          Label(fra3,text=int(fd_amount),font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=480)

        Label(fra3,text="Maturity Amount",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=480)

        Label(fra3,text="Maturity Date",font="comicsans 15 bold",fg="black",bg="lightgreen").place(x=130,y=515)

        Label(fra3,text="Senior Citizen Option Chosen",font="comicsans 15
bold",fg="black",bg="lightgreen").place(x=130,y=550)
        Label(fra3,text=n,font="comicsans 15",fg="black",bg="lightgreen").place(x=465,y=550)

        Button(fra3,text="Back to Home Page",command=tdr.destroy,font="comicsans 13
bold",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=900,
y=580)
        global blnc
        blnc=int(blnc)-int(amount.get())
        dis.config(text=blnc)
        mysql.execute("update account_details set balance="+str(blnc)+" where Account_No="+str(acountno))
        mysql.execute("Insert into "+tablename+" values(%s,%s,%s,%s,%s,%s)",(date_string,utr,"Fixed
Deopsit",amount.get(),None,blnc))
        mysql.execute("Insert into track_transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(acountno,user,date_string,utr,"Fixed Deposit",amount.get(),None,blnc))
        mydb.commit()

        ey=str(ey)
        ye=str(ye)
        ad=str(ad)
        fd_amount=str(fd_amount)
        fd_date=datetime.datetime.now()
        def fd_acc_insert():
          if tenure.get()=="Day(s)":
            fd_cmnd="Insert into Fixed_Deposit values(%s,%s,%s,%s,%s,%s,%s,%s,%s)"
```

```
mysql.execute(fd_cmnd,(str(fd_a_acno),acountno,user,str(fd_date),str(amount.get()),ROI,str(tenuredays.get())+"
Day(s) ",ey+"-"+ye+"-"+ad,fd_amount))
          mydb.commit()
        else:
          fd_cmnd="Insert into Fixed_Deposit values(%s,%s,%s,%s,%s,%s,%s,%s,%s)"

mysql.execute(fd_cmnd,(str(fd_a_acno),acountno,user,str(fd_date),str(amount.get()),ROI,str(tyears.get())+" Year(s)
"+ str(tmonths.get())+" Month(s) "+ str(tdays.get())+" Day(s) ",ey+"-"+ye+"-"+ad,fd_amount))
          mydb.commit()
      fd_acc_insert()
    def reset():
      amount.set("")
      sen.set(0)
      tnc.set(0)
      rad.set(" ")
      tdays.set(0)
      maturity.set(" ")
      tenure.set(" ")
      tenureyears.set(" ")
      tenuremonths.set(" ")
      tenuredays.set(0)
      fra7.place(width=width,y=350,height=300)
      can.place(width=width,height=1500,y=40)
    def rate_int(event):
      rate.config(font="calibri 14 underline")
    def rate_int2(event):
      rate.config(font="calibri 14")
    def condit1(event):
      condit.config(font="calibri 14 underline")
    def condit2(event):
      condit.config(font="calibri 14")

    def opt():
      fra3.place(width=width,y=350,height=600)
      fra7.place(width=width,height=500,y=430)
      can.place(width=width,height=1400,y=40)
    def ten():
      if tenure.get()=="Day(s)":
        fra4.place(width=width,height=80,y=70)
        fra5.place(width=width,height=500,y=870)
        fra7.place(width=width,height=500,y=520)
        can.place(width=width,height=1300,y=40)

        days=[]
        for i in range(0,32):
          days.append(i)
        #FRAME 4 CONTENT
        fra4.place(width=width,height=80,y=70)
        da=Label(fra4,text="Select the no. of days",bg="#fff",font="comicsans 14")
        da.place(y=20,x=20)
        day=ttk.Combobox(fra4,font="calibri 14",textvariable=tenuredays,values=days,state='readonly')
        day.current(days[0])
        day.place(y=20,x=500)

    elif tenure.get()=="Year(s) / Month(s) / Day(s)":
        fra4.place(width=width,height=80,y=870)
        fra7.place(width=width,height=500,y=600)
        can.place(width=width,height=1220,y=40)

        days=[]
        for i in range(0,32):
            days.append(i)

        Months=[]
        for q in range(0,13):
            Months.append(q)

        years0=[]
        for v in range(0,11):
            years0.append(v)

        #FRAME 5 CONTENT
        fra5.place(width=width,height=500,y=70)

        Label(fra5,text="Year(s)",bg="#fff",font="comicsans 14").place(y=20,x=20)
        day=ttk.Combobox(fra5,font="calibri 14",textvariable=tyears,values=years0,state='readonly')
        day.place(y=20,x=500)
        day.current(days[0])

        Label(fra5,text="Month(s)",bg="#fff",font="comicsans 14").place(y=80,x=20)
        day=ttk.Combobox(fra5,font="calibri 14",textvariable=tmonths,values=Months,state='readonly')
        day.place(y=80,x=500)
        day.current(days[0])

        Label(fra5,text="Day(s)",bg="#fff",font="comicsans 14").place(y=140,x=20)
        day=ttk.Combobox(fra5,font="calibri 14",textvariable=tdays,values=days,state='readonly')
        day.place(y=140,x=500)
        day.current(days[0])
#FD->e-STDR

#heading
Label(tdr,text="e-STDR(Fixed Deposit)",font="arial 20",bg="#ADD8E6",fg="#0000CD").pack(fill=X)
#frames
can=Canvas(tdr)
can.place(width=width,height=1500,y=40)

#Scroll bar
yscroll=Scrollbar(tdr,orient=VERTICAL,command=can.yview)
yscroll.pack(side=RIGHT,fill=Y)

can.config(yscrollcommand=yscroll.set)
```

```
    can.bind('<Configure>',lambda e: can.configure(scrollregion=can.bbox("all")))

    fra2=Frame(can,bg="#fff")
    can.create_window((0,0),window=fra2,width=width,height=1500)

    fra3=Frame(fra2,bg="#fff")
    fra4=Frame(fra3,bg="#fff")
    fra5=Frame(fra3,bg="#fff")
    fra7=Frame(fra2,bg="#fff")
    fra7.place(width=width,y=350,height=300)
    #heading
    Label(tdr,text="e-STDR(Fixed Deposit)",font="arial 20",bg="#ADD8E6",fg="#0000CD").place(width=width)
    l1=Label(fra2,text="Open new A/C",font="comicsans 20")
    l1.place(x=50,y=1)
    #FRAME 2 CONTENT
    Label(fra2,text="Account No.",font="comicsans 16 bold",bg="#fff").place(x=20,y=50)
    Label(fra2,text="Username",font="comicsans 16 bold",bg="#fff").place(x=420,y=50)
    Label(fra2,text="Account Type",font="comicsans 16 bold",bg="#fff").place(x=820,y=50)
    Label(fra2,text=acountno,font="comicsans 14 ",bg="#fff").place(x=20,y=100)
    Label(fra2,text=user,font="comicsans 14 ",bg="#fff").place(x=420,y=100)
    Label(fra2,text=acctyp,font="comicsans 14 ",bg="#fff").place(x=820,y=100)
    Label(fra2,text="Amount",font="comicsans 15",bg="#fff").place(x=20,y=150)
    Entry(fra2,textvariable=amount,font="comicsans 14",bd=3,relief=GROOVE).place(x=200,y=150)
    amount.set("")
    Checkbutton(fra2,text="",font="comicsans
14",bg="#fff",activebackground="#fff",variable=sen).place(x=20,y=200)
    sen.set(0)
    Label(fra2,text="Senior Citizen",font="comicsans 14",bg="#fff").place(x=40,y=202)
    Label(fra2,text="(Age 60 years or above)",font="comicsans 14",bg="#fff",fg="#D2691E").place(x=165,y=201)
    Label(fra2,text="Senior Citizens will get higher rate of interest on deposits of Rs10,000 and above",font="calibri
14",bg="#fff").place(x=20,y=227)
    Label(fra2,text="Select the term deposit options",font="comicsans
15",bg="#fff",fg="#00008B").place(x=20,y=270)
    Radiobutton(fra2,text="",font="comicsans 14",bg="#fff",command=opt,variable=rad,value="STDR(Cumulative -
Interest paid at maturity)").place(x=20,y=270)
    rad.set(" ")
    Label(fra2,text="STDR(Cumulative - Interest paid at maturity)",font="calibri 14",bg="#fff").place(x=45,y=270)
    #FRAME 3 CONTENT
    Label(fra3,text="Tenure of Deposit",font="comicsans 15",bg="#fff",fg="#00008B").place(x=20)
    Radiobutton(fra3,text="",font="comicsans
14",bg="#fff",variable=tenure,value="Day(s)",command=ten).place(x=20,y=40)
    tenure.set(" ")
    Label(fra3,text="Days",font="calibri 14",bg="#fff").place(x=40,y=35)
    Radiobutton(fra3,bg="#fff",variable=tenure,value="Year(s) / Month(s) /
Day(s)",command=ten).place(x=225,y=40)
    tenure.set(" ")
    Label(fra3,text="Year(s) / Month(s) / Day(s)",font="calibri 14",bg="#fff").place(x=245,y=35)
    #FRAME 7 CONTENT
    Checkbutton(fra7,text="",bg="#fff",activebackground="#fff",variable=tnc).place(x=20,y=10)
    tnc.set(0)
    Label(fra7,text="I accept the",font="calibri 14",bg="#fff").place(x=40,y=5)
```

```
        condit=Label(fra7,text="Terms and conditions",font="calibri 14",bg="#fff",fg="#0000CD")
        condit.place(x=135,y=5)
        condit.bind("<Enter>",condit1)
        condit.bind("<Leave>",condit2)
        condit.bind("<ButtonPress-1>",terms_n_conditions)

        Label(fra7,text="View interest rate ",font="calibri 14",bg="#fff").place(x=500,y=5)
        rate=Label(fra7,text="Domestic term deposit",font="calibri 14",bg="#fff",fg="#0000CD")
        rate.place(x=650,y=5)
        rate.bind("<Enter>",rate_int)
        rate.bind("<Leave>",rate_int2)
        rate.bind("<ButtonPress-1>",domestic)



        Label(fra7,text='''*  As per section 206AA introduced by Finance (No.2) Act 2009 effective April 01,2010, every
person who receives income on which TDS is deductible
shall furnish his PAN , failing which TDS shall be deducted at the rate of 20% in case of Domestic deposits and 30.90%
in case of NRO deposits.
            ''',font="calibri 14",bg="#fff",fg="red").place(x=40,y=50)
        Label(fra7,text='''*  Additionally, in the absence of PAN, Form 15G/H and other exemption certificates will be
invalid even if submitted & penal TDS will be applicable.''',font="calibri 14",bg="#fff",fg="red").place(x=40,y=100)
        Label(fra7,text='''*  NRO depositors are required to submit self declaration / tax residency certificates for
availing concessional TDS facility under DTAA.''',font="calibri 14",bg="#fff",fg="red").place(x=40,y=125)
        Label(fra7,text='''*  No Loan can be sanctioned against the term deposits opened under Tax Savings
Scheme.''',font="calibri 14",bg="#fff",fg="red").place(x=40,y=150)
        Button(fra7,text="Submit",command=submit,font="comicsans
15",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff").place(x=500,y=250)
        Button(fra7,text="Reset",command=reset,font="comicsans
15",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff").place(x=600,y=250)

    elif mymenu.get()=="Close A/C Prematurely":
        mymenu.set("Fixed Deposits")
        premature()
#CLOSE ACCOUNT PREMATURELY
def premature():
    global pre
    global fd4_pre
    global fd_pre_accno



    #Buttons

    pre=Toplevel(us)
    pre.minsize(1000,670)
    pre.geometry(scr_width+"x"+scr_height+"+2+0")

    pre.title("e-STDR(Fixed Deposit)")
    pre.configure(bg="#fff")

    Label(pre,text="Close A/C Prematurely",font="arial 20",bg="#ADD8E6",fg="#0000CD").pack(fill=X)
    #frames
    prfr=Frame(pre,bg="#fff")
    prfr.place(y=50,width=scr_width,height=scr_height)

    #labels
    Label(prfr,text="Modify Maturity Instruction",font="arial 14",bg="#fff",fg="navyblue").place(x=50,y=10)


Label(prfr,text="____________________________________________________________________________________
__________________________________________________________________________________________________
_____________________________________________________________________",bg="#fff",fg="grey").place(x=30,y
=35)

    L3=Label(prfr,text="Mandatory fields are marked with an asterisk(*)",font="arial 10 italic ",
bg="#fff",fg="red").place(x=30,y=55)

    L4=Label(prfr,text="Select Fixed Deposit Account For Closing",font="arial 14",
bg="#fff",fg="navyblue").place(x=30,y=90)

    tablepre=Frame(pre)
    tablepre.place(x=2,y=190,width=1260,height=370)
    #scrollbar
    xscroll=Scrollbar(tablepre,orient=HORIZONTAL)
    yscroll=Scrollbar(tablepre,orient=VERTICAL)
    #table

fd4_pre=ttk.Treeview(tablepre,show="headings",columns=('FD_AccNo','AHN','DepositDate','Deposit_Amount','ROI','
Tenure','MaturityDate','Amountt'),xscrollcommand=xscroll.set,yscrollcommand=yscroll.set)
    xscroll.pack(side=BOTTOM,fill=X)
    yscroll.pack(side=RIGHT,fill=Y)
    xscroll.config(command=fd4_pre.xview)
    yscroll.config(command=fd4_pre.yview)

    #column
    fd4_pre.column('FD_AccNo',anchor=CENTER,width=50)
    fd4_pre.column('AHN',anchor=CENTER,width=130)
    fd4_pre.column('DepositDate',anchor=CENTER,width=110)
    fd4_pre.column('Deposit_Amount',anchor=CENTER,width=60)
    fd4_pre.column('ROI',anchor=CENTER,width=15)
    fd4_pre.column('Tenure',anchor=CENTER,width=115)
    fd4_pre.column('MaturityDate',anchor=CENTER,width=60)
    fd4_pre.column('Amountt',anchor=CENTER,width=60)

    #headings
    fd4_pre.heading('FD_AccNo',text="FD Account No.")
    fd4_pre.heading('AHN',text="Account Holder Name")
    fd4_pre.heading('DepositDate',text="Deposit Date")
    fd4_pre.heading('Deposit_Amount',text="Deposit Amount")
    fd4_pre.heading('ROI',text="ROI")
    fd4_pre.heading('Tenure',text="Tenure")
    fd4_pre.heading('MaturityDate',text='Maturity Date')
    fd4_pre.heading('Amountt',text='Maturity Amount')
    fd4_pre.pack(fill=BOTH,expand='yes')
    pre_display_acc()
    fd4_pre.bind("<ButtonRelease-1>",pre_get_acc)

    L14=Label(prfr,text="Selected Account Number",font="arial 14", bg="#fff",fg="#000").place(x=30,y=520)

    Entry(prfr,font="comicsans
14",bd=3,relief=GROOVE,justify=CENTER,state="readonly",textvariable=fd_pre_accno).place(x=270,y=520)
    fd_pre_accno.set("")
    def pre_coff():
        if fd_pre_accno.get()==(""):
            messagebox.showinfo("Error","Please select an account for closing it.",parent=pre)
        else:
            proceed()

    Button(prfr,text="Proceed",command=pre_coff,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=5,pady=2).place(x=500,y=
570)
    Button(prfr,text="Back",command=pre.destroy,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=570)
def per_FD():
    global e_STR_AC_No
    global credit_AC
    global Deposit_At
    global Deposit_Dt
    global Maturity_Dt
    global Amount_pay
    mysql.execute("select * from Fixed_Deposit where FD_Account_No='"+fd_pre_accno.get()+"'")
    fd=mysql.fetchall()
    for j in fd:
        e_STR_AC_No=j[0]
        credit_AC=j[1]
        Deposit_Dt=j[3]
        Deposit_At=j[4]
        Amount_pay=j[4]
        Maturity_Dt=j[7]
def proceed():
    #frames
    fr1=Frame(pre,bg="#fff")
    fr1.place(y=50,width=scr_width,height=scr_height)
    per_FD()
    #labels
    Label(fr1,text="Modify Maturity Instruction",font="arial 14", bg="#fff",fg="navyblue").place(x=50,y=10)


Label(fr1,text="____________________________________________________________________________________
_______________________________________________________________________________________
__________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y
=35)

    Label(fr1,text="Mandatory fields are marked with an asterisk(*)",font="arial 10 italic ",
bg="#fff",fg="red").place(x=30,y=55)

    Label(fr1,text="Please verify the e-STDR details and confirm closure request",font="arial 14",
bg="#fff",fg="navyblue").place(x=30,y=90)

    Label(fr1,text="e-STDR a/c no.",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=130)
    Label(fr1,text=e_STR_AC_No,font="comicsans 13",fg="black",bg="white").place(x=465,y=130)

    Label(fr1,text="Credit Account Number",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=170)
    Label(fr1,text=credit_AC,font="comicsans 13",fg="black",bg="white").place(x=465,y=170)

    Ll7=Label(fr1,text="Deposit Amount",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=210)
    Label(fr1,text=Deposit_At,font="comicsans 13",fg="black",bg="white").place(x=465,y=210)

    Ll8=Label(fr1,text="Deposit Date",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=250)
    Label(fr1,text=Deposit_Dt,font="comicsans 13",fg="black",bg="white").place(x=465,y=250)

    Ll9=Label(fr1,text="Maturity Date",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=290)
    Label(fr1,text=Maturity_Dt,font="comicsans 13",fg="black",bg="white").place(x=465,y=290)

    Ll11=Label(fr1,text="Penalty Amount",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=330)
    Label(fr1,text="0.00 INR",font="comicsans 13",fg="black",bg="white").place(x=465,y=330)

    Ll12=Label(fr1,text='''Rate of Interest Applicable
    (% p.a.) ''',font="comicsans 13 bold",fg="black",bg="white")
    Ll12.place(x=65,y=370)
    Label(fr1,text="0.00%",font="comicsans 13 ",fg="black",bg="white").place(x=465,y=370)

    Ll13=Label(fr1,text="Amount Payable",font="comicsans 13 bold",fg="black",bg="white").place(x=65,y=420)
    Label(fr1,text=Amount_pay,font="comicsans 13",fg="black",bg="white").place(x=465,y=420)

    #buttons
    Button(fr1,text="Confirm",command=confirm_fd,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=5,pady=2).place(x=500,y=
520)
    Button(fr1,text="Cancel",command=fr1.destroy,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=520)
def enter(event):
    backhome.config(font="comicsans 13 underline")
def leave(event):
    backhome.config(font="comicsans 13")
def enterleave(event):
    pre.destroy()
def confirm_fd():
    utrno()
    global backhome
```

```
    global blnc
    fr1=Frame(pre,bg="#fff")
    fr1.place(y=50,width=scr_width,height=scr_height)
    con_fd=Frame(pre,bg="lightgreen")
    con_fd.place(x=290,y=260,width=700,height=200)
    Label(con_fd,text="Your e-STD a/c XXXXXXX"+e_STR_AC_No[7:]+" has been successfully closed, Please
",font="comicsans 15",fg="black",bg="lightgreen").place(x=20,y=30)
    Label(con_fd,text="check the debit a/c XXXXX"+credit_AC[5:]+" for credit proceeds.",font="comicsans
15",fg="black",bg="lightgreen").place(x=20,y=75)
    blnc=int(blnc)+int(Amount_pay)
    dis.config(text=blnc)
    mysql.execute("update account_details set balance="+str(blnc)+" where Account_No="+str(acountno))
    mysql.execute("delete from Fixed_Deposit where FD_Account_No='"+e_STR_AC_No+"'")
    mydb.commit()
    backhome=Label(con_fd,text="Back to Home Page",font="comicsans
13",bg="lightgreen",fg="#0000CD",padx=10,pady=2)
    backhome.place(x=20,y=150)
    backhome.bind("<Enter>",enter)
    backhome.bind("<Leave>",leave)
    backhome.bind("<ButtonPress-1>",enterleave)
    mysql.execute("Insert into "+tablename+" values(%s,%s,%s,%s,%s,%s)",(date_string,utr,"Prematurely Closed
FD",None,Amount_pay,blnc))
    mysql.execute("Insert into track_transaction
values(%s,%s,%s,%s,%s,%s,%s,%s)",(acountno,user,date_string,utr,"Prematurely Closed
FD",None,Amount_pay,blnc))
    mydb.commit()
def beneficaries():
    global diction
    mysql.execute("Select * from account_details")
    rows=mysql.fetchall()
    diction={"-Account No.-":""}
    for row in rows:
        if row[0]!=acountno and row[4]!="Inactive":
            index=str(row[0])
            value=row[1]
            diction[index]=value
def change(event):
    ben.config(text=diction[beneficary.get()])
def TnC1(event):
    TnC.config(font="calibri 14 underline")
def TnC2(event):
    TnC.config(font="calibri 14")
def fund():
    global TnC
    beneficaries()
    global ben
    global fun
    global dis2
    fun=Toplevel(us)
    fun.minsize(scr_width,scr_height)
    fun.geometry(scr_width+"x"+scr_height+"+2+0")
    fun.title("FUND TRANSFER")
    fun.configure(bg="#fff")
    #frames

    fra2=Frame(fun,bg="#fff")
    fra2.place(y=40,width=scr_width,height=scr_height)

    Label(fun,text="Fund Transfer",font="arial 20",bg="#ADD8E6",fg="#0000CD").pack(fill=X)
    #labels
    Label(fra2,text="Mandatory fields are marked with an asterisk(*)",font="arial 10 italic ",
bg="#fff",fg="red").place(x=30,y=10)

    Label(fra2,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=35,y=50)

    Label(fra2,text="Account Holder Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=280,y=50)

    Label(fra2,text="Account Type",font="arial 14 bold", bg="#fff",fg="#000").place(x=800,y=50)

    Label(fra2,text="Balance",font="arial 14 bold", bg="#fff",fg="#000").place(x=1100,y=50)


Label(fra2,text="__________________________________________________________________________
_________________________________________________________________________________
____________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=7
5)

    Label(fra2,text=acountno,font="arial 14",bg="#fff").place(x=35,y=100)
    Label(fra2,text=user,font="arial 14",bg="#fff").place(x=280,y=100)
    Label(fra2,text=acctyp,font="arial 14",bg="#fff").place(x=800,y=100)
    dis2=Label(fra2,text=blnc,font="arial 14",bg="#fff")
    dis2.place(x=1100,y=100)

    Label(fra2,text="Amount",font="comicsans 14 bold",fg="black",bg="white").place(x=65,y=150)

    r8=Label(fra2,text="*",font="comicsans 14 bold",fg="red",bg="white")
    r8.place(x=140,y=150)

    r9=Label(fra2,text="INR",font="comicsans 14 bold",fg="black",bg="white")
    r9.place(x=500,y=1500)

    r10=Label(fra2,text="Purpose",font="comicsans 14 bold",fg="black",bg="white")
    r10.place(x=65,y=200)

    r11=Label(fra2,text="Select the Beneficiary Account",font="comicsans 14 bold",fg="navyblue",bg="white")
    r11.place(x=35,y=260)

    Label(fra2,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=35,y=300)

    Label(fra2,text="Beneficiary Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=320,y=300)
Label(fra2,text="___________________________________________________________________________
________________________________________________________________________________________
____________________________________________________________",bg="#fff",fg="grey").place(x=30,y=3
25)

    r16=Label(fra2,text="Selected Account Number",font="comicsans 14 bold",fg="black",bg="white")
    r16.place(x=65,y=450)

    r17=Label(fra2,text="Please note, as per RBI instructions credit will be effected based solely on the beneficiary
account number information and",font="comicsans 13 bold",fg="black",bg="white")
    r17.place(x=65,y=520)

    r18=Label(fra2,text="the beneficiary name particulars will not be used therefore.",font="comicsans 13
bold",fg="black",bg="white")
    r18.place(x=65,y=550)

    TnC=Label(fra2,text="Terms and conditions",font="calibri 14",bg="#fff",fg="#0000CD")
    TnC.place(x=185,y=583)
    TnC.bind("<Enter>",TnC1)
    TnC.bind("<Leave>",TnC2)
    TnC.bind("<ButtonPress-1>",terms_n_conditions)
    #combobox
    ty=ttk.Combobox(fra2,font="comicsans 14 ",justify=CENTER,width=30,textvariable=purpose,values=('--Select
Purpose--',"Payment towards loan repayment","Deposit / Investment","Gift to relatives /
Friends","Donation","Payment of Education fee"),state='readonly')
    ty.place(x=270,y=200)
    ty.current(0)
    #entries
    l1=[]
    for c in diction.keys():
        l1.append(c)
    Entry(fra2,font="comicsans
14",bd=3,relief=GROOVE,textvariable=fundamount,justify=CENTER).place(x=270,y=150)
    acombo=ttk.Combobox(fra2,font="comicsans 14",textvariable=beneficary,state='readonly',justify=CENTER)
    acombo.place(x=35,y=350,width=150)
    acombo['values']=l1
    acombo.current(0)
    acombo.bind("<<ComboboxSelected>>",change)
    fun.option_add("*TCombobox*Listbox.font","comicsans 13")

    ben=Label(fra2,text=diction[beneficary.get()],font="arial 14", bg="#fff",fg="#000")
    ben.place(x=320,y=350)

    Entry(fra2,font="comicsans
14",bd=3,relief=GROOVE,textvariable=beneficary,state='readonly',justify=CENTER).place(x=350,y=450)

    #checkbutton
    Checkbutton(fra2,text="I accept the
",variable=fcheck,onvalue=1,offvalue=0,bg='white',font=("calibri",14)).place(x=65,y=580)
    #buttons
    Button(fra2,text="Submit",command=fun_submit,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=500,y
=610)
    Button(fra2,text="Cancel",command=fun_destroy,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=610)
def fun_destroy():
    fundamount.set("")
    fcheck.set(0)
    beneficary.set("")
    fun.destroy()
def fun_submit():
    if fundamount.get()=="":
        messagebox.showinfo("ERROR","Please Enter the Amount to be transfer",parent=fun)
    elif int(fundamount.get())>int(blnc):
        messagebox.showerror("ERROR","Insufficient Balance",parent=fun)
    elif int(fundamount.get())>25000:
        messagebox.showinfo("ERROR","You can't transfer more than 25,000",parent=fun)
    elif purpose.get()=="--Select Purpose--":
        messagebox.showinfo("ERROR","Please select the purpose",parent=fun)
    elif beneficary.get()=="-Account No.-":
        messagebox.showinfo("ERROR","Please select the beneficary account for transfer",parent=fun)
    elif fcheck.get()==0:
        messagebox.showinfo("ERROR","Please Agree to our Terms and Conditions",parent=fun)
    else:
        #frames
        nfr1=Frame(fun,bg="#fff")
        nfr1.place(y=50,width=scr_width,height=scr_height)
        #labels
        Label(nfr1,text="Verify details and confirm this transaction",font="arial 17 bold",
bg="#fff",fg="black").place(x=50,y=50)
        Label(nfr1,text="Debit Account Details",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=50,y=100)

        Label(nfr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=150)
        Label(nfr1,text=acountno,font="arial 14", bg="#fff").place(x=350,y=150)

        Label(nfr1,text="Account Type",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=200)
        Label(nfr1,text=acctyp,font="arial 14", bg="#fff").place(x=350,y=200)

        Label(nfr1,text="Credit to Beneficiary",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=250)
        Label(nfr1,text=fundamount.get(),font="arial 14", bg="#fff").place(x=350,y=250)

        Label(nfr1,text="Commission Amount",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=300)
        Label(nfr1,text="0.0% Goods and Service Tax",font="arial 14", bg="#fff",fg="#000").place(x=350,y=300)

        Label(nfr1,text="Total Debit Amount",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=350)
        Label(nfr1,text=fundamount.get(),font="arial 14", bg="#fff").place(x=350,y=350)

        Label(nfr1,text="Purpose",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=400)
        Label(nfr1,text=purpose.get(),font="arial 14", bg="#fff",fg="#000").place(x=350,y=400)
        Label(nfr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=60,y=500)
        Label(nfr1,text=beneficary.get(),font="arial 14", bg="#fff",fg="#000").place(x=60,y=550)

        Label(nfr1,text="Beneficiary Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=370,y=500)
        Label(nfr1,text=diction[beneficary.get()],font="arial 14", bg="#fff",fg="#000").place(x=370,y=550)

        Label(nfr1,text="Credit Account Details",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=50,y=450)


Label(nfr1,text="___________________________________________________________________________________________________
__________________________________________________________________________________________________________________________
__________________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=5
25)

        Label(nfr1,text="Inter Bank Fund Transfer",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=900,y=550)

        #buttons
        Button(nfr1,text="Confirm",command=fun_confirm,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=500,y
=610)
        Button(nfr1,text="Cancel",command=nfr1.destroy,font="comicsans
12",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=610)
def fun_confirm():
    utrno()
    fund_update()
    #frames
    nr1=Frame(fun,bg="lightgreen")
    nr1.place(y=35,width=scr_width,height=scr_height)
    #labels
    Label(nr1,text="Please note this transaction number for future reference : ",font="comicsans 15
",fg="black",bg="lightgreen").place(x=50,y=20)
    Label(nr1,text=utr,font="comicsans 15",bg="lightgreen").place(x=580,y=25)
    Label(nr1,text="(UTR No.)",font="comicsans 15",bg="lightgreen").place(x=700,y=25)

    Label(nr1,text="Debit Transaction Status : Completed Successfully",font="comicsans 15
",fg="black",bg="lightgreen").place(x=50,y=60)

    Label(nr1,text="Debit Account Details",font="arial 14 bold", bg="lightgreen",fg="navyblue").place(x=50,y=120)

    Label(nr1,text="Account No.",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=70,y=160)
    Label(nr1,text=acountno,font="arial 14", bg="lightgreen",fg="#000").place(x=70,y=230)

    Label(nr1,text="Account Type",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=300,y=160)
    Label(nr1,text=acctyp,font="arial 14", bg="lightgreen",fg="#000").place(x=300,y=230)

    Label(nr1,text="Amount",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=530,y=160)
    Label(nr1,text=fundamount.get(),font="comicsans 14",fg="black",bg="lightgreen").place(x=530,y=230)

    Label(nr1,text='''Commission
    Amount''',font="arial 14 bold", bg="lightgreen",fg="#000").place(x=680,y=160)
    Label(nr1,text='0.0%',font="comicsans 14",fg="black",bg="lightgreen").place(x=680,y=230)

    Label(nr1,text='''Transaction
    Type''',font="arial 14 bold", bg="lightgreen",fg="#000").place(x=860,y=160)

    Label(nr1,text="UTR Number",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=1050,y=160)
    Label(nr1,text=utr,font="comicsans 14",fg="black",bg="lightgreen").place(x=1050,y=230)


Label(nr1,text="______________________________________________________________________________________
__________________________________________________________________________________________________
_________________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=210)

    Label(nr1,text='''Inter Bank
    Fund Transfer''',font="comicsans 14",fg="black",bg="lightgreen").place(x=860,y=230)

    Label(nr1,text="Account No.",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=70,y=350)
    Label(nr1,text=beneficary.get(),font="arial 14", bg="lightgreen",fg="#000").place(x=70,y=400)

    Label(nr1,text="Beneficiary Name",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=300,y=350)
    Label(nr1,text=diction[beneficary.get()],font="arial 14", bg="lightgreen",fg="#000").place(x=300,y=400)

    Label(nr1,text="Credit Account Details",font="arial 14 bold", bg="lightgreen",fg="navyblue").place(x=50,y=310)


Label(nr1,text="______________________________________________________________________________________
__________________________________________________________________________________________________
_________________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=375)

    Label(nr1,text='''Inter Bank
    Fund Transfer''',font="comicsans 14",fg="black",bg="lightgreen").place(x=860,y=400)

    Label(nr1,text="Transfer Type",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=860,y=350)

    Label(nr1,text="Purpose",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=500,y=350)
    Label(nr1,text=purpose.get(),font="arial 14", bg="lightgreen",fg="#000").place(x=500,y=400)

    Label(nr1,text="Amount",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=1050,y=350)
    Label(nr1,text=fundamount.get(),font="comicsans 14",fg="black",bg="lightgreen").place(x=1050,y=400)


Label(nr1,text="______________________________________________________________________________________
__________________________________________________________________________________________________
_________________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=450)
    #button
    Button(nr1,text="Back to Home Page",command=fun_destroy,font="comicsans 13
bold",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=850,
y=560)
```

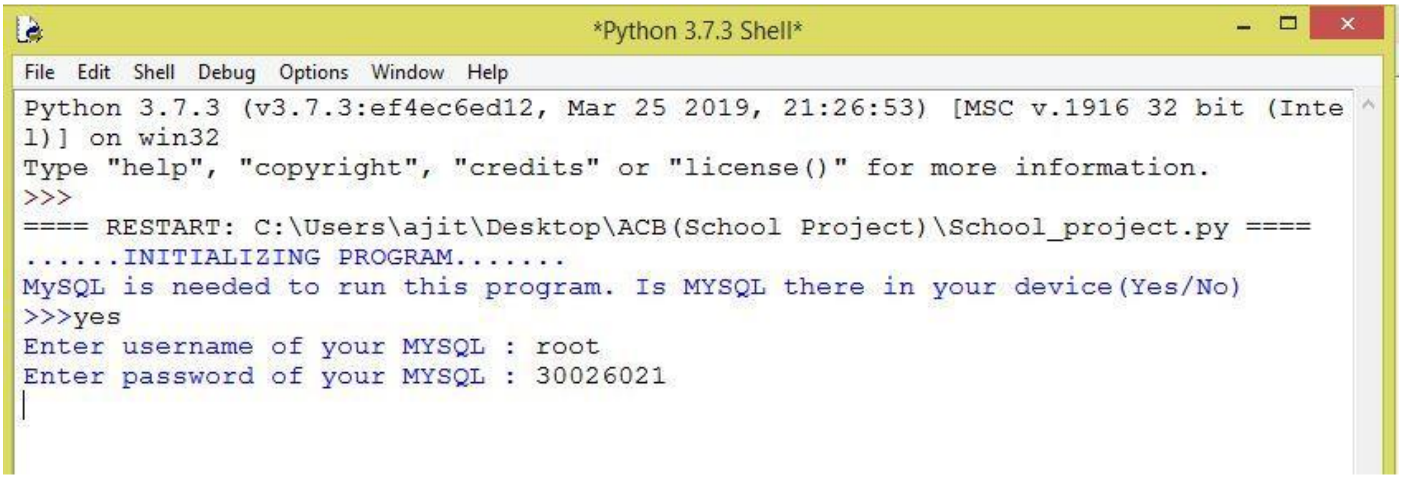
```
def state():
   global acst
   global t4
   acst=Toplevel(us)
   acst.title("ACCOUNT STATEMENT")
   acst.geometry(scr_width+"x"+scr_height+"+2+0")
   acst.config(bg="blue")
   #heading
   Label(acst,text="Account Statements",font="comicsans 40 bold",bg="#fff").pack(fill=X)
   #frame
   tablefr=Frame(acst)
   tablefr.place(x=0,y=150,width=1260,height=500)
   #scrollbar
   xscroll=Scrollbar(tablefr,orient=HORIZONTAL)
   yscroll=Scrollbar(tablefr,orient=VERTICAL)
   #table

t4=ttk.Treeview(tablefr,show="headings",columns=('date','refrno','desc','debit','credit','balnc'),xscrollcommand=xscr
oll.set,yscrollcommand=yscroll.set)
   xscroll.pack(side=BOTTOM,fill=X)
   yscroll.pack(side=RIGHT,fill=Y)
   xscroll.config(command=t4.xview)
   yscroll.config(command=t4.yview)
   #column
   t4.column('date',anchor=CENTER,width=150)
   t4.column('refrno',anchor=CENTER,width=125)
   t4.column('desc',anchor=CENTER,width=320)
   t4.column('debit',anchor=CENTER,width=100)
   t4.column('credit',anchor=CENTER,width=100)
   t4.column('balnc',anchor=CENTER,width=150)
   #headings
   t4.heading('date',text="Transaction Date")
   t4.heading('refrno',text="Reference No.")
   t4.heading('desc',text="Description")
   t4.heading('debit',text="Debit")
   t4.heading('credit',text="Credit")
   t4.heading('balnc',text='Balance')
   t4.pack(fill=BOTH,expand='yes')
   fetch_statement()
   Button(acst,text="<Back",font="comicsans 12 bold",command=acst.destroy).place(x=1150,y=655)
def bill():
   def biller(event):
      if bills.get()=="--Select--":
         baccno.config(text="")
         bname.config(text="")
         bcat.config(text="")
      elif bills.get()=="Electricity Bill":
         baccno.config(text="899287635")
         bname.config(text="Tata Power - DDL")
         bcat.config(text="Electricity Bill")
      elif bills.get()=="Water Bill":
        baccno.config(text="978065476")
        bname.config(text="Delhi Jal Board")
        bcat.config(text="Water Bill")
      elif bills.get()=="Gas Bill":
        baccno.config(text="790865354")
        bname.config(text="Indraprastha Gas Limited")
        bcat.config(text="Gas Bill")
    global bil
    global baccno
    global bcat
    global bname
    global dis3
    bil=Toplevel(us)
    bil.minsize(1000,670)
    bil.geometry(scr_width+"x"+scr_height+"+2+0")
    bil.title("BILL PAYMMENTS")
    bil.configure(bg="#fff")
    Label(bil,text="Bill Payments",font="arial 30",bg="#ADD8E6",fg="#0000CD").pack(fill=X)
    #frames
    fr1=Frame(bil,bg="#fff")
    fr1.place(y=50,width=scr_width,height=scr_height)
    Label(fr1,text="Amount to Pay ",bg="#fff",font="arial 16 bold").place(x=100,y=400)
    am=Entry(fr1,textvariable=bilpay,font="arial 14",bd=3,relief=GROOVE)
    am.place(x=400,y=400)
    bilpay.set("")
    #labels
    Label(fr1,text="Select Bill Category",font="arial 17 bold", bg="#fff",fg="navyblue").place(x=70,y=170)

    Label(fr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=35,y=50)
    Label(fr1,text=acountno,font="arial 14", bg="#fff",fg="#000").place(x=35,y=100)

    Label(fr1,text="Account Holder Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=280,y=50)
    Label(fr1,text=user,font="arial 14", bg="#fff",fg="#000").place(x=280,y=100)

    Label(fr1,text="Account Type",font="arial 14 bold", bg="#fff",fg="#000").place(x=800,y=50)
    Label(fr1,text=acctyp,font="arial 14", bg="#fff",fg="#000").place(x=800,y=100)

    Label(fr1,text="Balance",font="arial 14 bold", bg="#fff",fg="#000").place(x=1100,y=50)
    dis3=Label(fr1,text=blnc,font="arial 14", bg="#fff",fg="#000")
    dis3.place(x=1100,y=100)


Label(fr1,text="_________________________________________________________________________________________
_____________________________________________________________________________________________________
____________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=75
)


Label(fr1,text="_________________________________________________________________________________________
_____________________________________________________________________________________________________
___________________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=20
0)

    Label(fr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=500,y=220)
    baccno=Label(fr1,text="",font="arial 14", bg="#fff",fg="#000")
    baccno.place(x=500,y=270)

    Label(fr1,text="Category",font="arial 14 bold", bg="#fff",fg="#000").place(x=70,y=220)
    bcat=Label(fr1,text="",font="arial 14", bg="#fff",fg="#000")
    bcat.place(x=70,y=270)

    Label(fr1,text="Beneficiary Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=900,y=220)
    bname=Label(fr1,text="",font="arial 14", bg="#fff",fg="#000")
    bname.place(x=900,y=270)


Label(fr1,text="_____________________________________________________________________________________________________________
___________________________________________________________________________________________________________________________
___________________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=24
5)

    Entry(fr1,)

    cat=ttk.Combobox(fr1,font="comicsans 14",textvariable=bills,justify=CENTER,state='readonly',values=("--Select--
","Electricity Bill","Water Bill","Gas Bill"))
    cat.place(x=500,y=170)
    cat.bind("<<ComboboxSelected>>",biller)
    cat.current(0)

Label(fr1,text="_____________________________________________________________________________________________________________
___________________________________________________________________________________________________________________________
___________________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=32
0)

    #buttons
    Button(fr1,text="Submit",command=bill_submit,font="comicsans
13",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=500,y
=600)
    Button(fr1,text="Cancel",command=bil.destroy,font="comicsans
13",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=600)

def bill_submit():
    if bills.get()=="--Select--":
      messagebox.showinfo("ERROR","Please choose the bill category",parent=bil)
    elif bilpay.get()=="":
      messagebox.showinfo("ERROR","Please Enter the Amount to pay",parent=bil)
    elif int(bilpay.get())>int(blnc):
      messagebox.showerror("ERROR","Insufficient Balance",parent=bil)
    else:
      #frame
```

```
    fr1=Frame(bil,bg="#fff")
    fr1.place(y=50,width=scr_width,height=scr_height)

    #labels
    Label(fr1,text="Verify details and confirm this transaction",font="arial 17 bold",
bg="#fff",fg="black").place(x=50,y=40)

    Label(fr1,text="Debit Account Details",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=50,y=100)

    Label(fr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=150)
    Label(fr1,text=acountno,font="arial 14", bg="#fff",fg="#000").place(x=500,y=150)

    Label(fr1,text="Account Type",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=200)
    Label(fr1,text=acctyp,font="arial 14", bg="#fff",fg="#000").place(x=500,y=200)

    Label(fr1,text="Account Holder",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=250)
    Label(fr1,text=user,font="arial 14", bg="#fff",fg="#000").place(x=500,y=250)

    Label(fr1,text="Credit to Beneficiary",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=300)
    Label(fr1,text=bilpay.get(),font="arial 14", bg="#fff",fg="#000").place(x=500,y=300)

    Label(fr1,text="Purpose",font="arial 14 bold", bg="#fff",fg="#000").place(x=50,y=350)
    Label(fr1,text=bills.get(),font="arial 14", bg="#fff",fg="#000").place(x=500,y=350)

    Label(fr1,text="Account No.",font="arial 14 bold", bg="#fff",fg="#000").place(x=60,y=500)
    Label(fr1,text=baccno["text"],font="arial 14", bg="#fff",fg="#000").place(x=60,y=550)

    Label(fr1,text="Beneficiary Name",font="arial 14 bold", bg="#fff",fg="#000").place(x=370,y=500)
    Label(fr1,text=bname["text"],font="arial 14", bg="#fff",fg="#000").place(x=370,y=550)

    Label(fr1,text="Credit Account Details",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=50,y=450)


Label(fr1,text="______________________________________________________________________________________________
_____________________________________________________________________________________________________________
_________________________________________________________________________",bg="#fff",fg="grey").place(x=30,y=52
5)

    Label(fr1,text="Inter Bank Fund Transfer",font="arial 14 bold", bg="#fff",fg="navyblue").place(x=900,y=550)

    #buttons
    Button(fr1,text="Confirm",command=bil_confirm,font="comicsans
13",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=500,y
=600)
    Button(fr1,text="Cancel",command=fr1.destroy,font="comicsans
13",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=600,y
=600)
def bil_confirm():
  utrno()
  bill_update()
  #frames
```

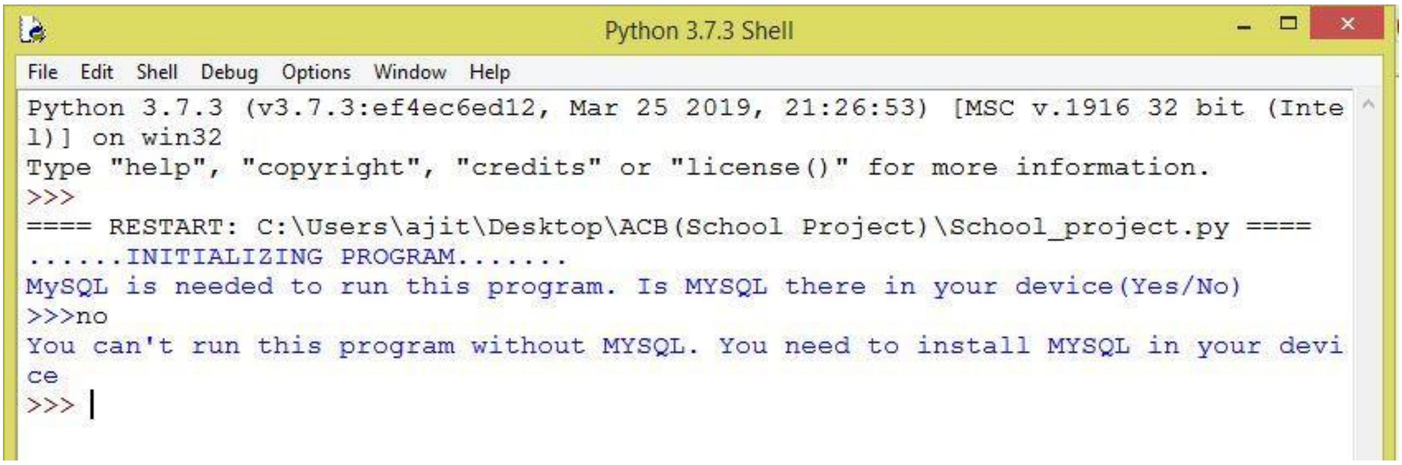
```
    nr1=Frame(bil,bg="lightgreen")
    nr1.place(y=47,width=scr_width,height=scr_height)

    #labels
    Label(nr1,text="Please note this transaction number for future reference : ",font="comicsans 15
",fg="black",bg="lightgreen").place(x=50,y=20)
    Label(nr1,text=utr,font="comicsans 15",bg="lightgreen").place(x=580,y=25)
    Label(nr1,text="(UTR No.)",font="comicsans 15",bg="lightgreen").place(x=710,y=25)

    Label(nr1,text="Debit Transaction Status : Completed Successfully",font="comicsans 15
",fg="black",bg="lightgreen").place(x=50,y=60)

    Label(nr1,text="Debit Account Details",font="arial 14 bold", bg="lightgreen",fg="navyblue").place(x=50,y=160)

    Label(nr1,text="Account No.",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=70,y=200)
    Label(nr1,text=acountno,font="arial 14", bg="lightgreen",fg="#000").place(x=70,y=300)

    Label(nr1,text="Account Holder",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=300,y=200)
    Label(nr1,text=user,font="arial 14", bg="lightgreen",fg="#000").place(x=300,y=300)

    Label(nr1,text="Account Type",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=530,y=200)
    Label(nr1,text=acctyp,font="arial 14", bg="lightgreen",fg="#000").place(x=530,y=300)

    Label(nr1,text="Amount",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=700,y=200)
    Label(nr1,text=bilpay.get(),font="arial 14", bg="lightgreen",fg="#000").place(x=700,y=300)

    Label(nr1,text='''Transaction
    Type''',font="arial 14 bold", bg="lightgreen",fg="#000").place(x=860,y=200)

    Label(nr1,text="UTR Number",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=1050,y=200)
    Label(nr1,text=utr,font="comicsans 14",fg="black",bg="lightgreen").place(x=1050,y=300)


Label(nr1,text="____________________________________________________________________________________
____________________________________________________________________________________________
_____________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=250)

    Label(nr1,text='''Inter Bank
    Fund Transfer''',font="comicsans 14",fg="black",bg="lightgreen").place(x=860,y=270)

    Label(nr1,text="Account No.",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=70,y=390)
    Label(nr1,text=baccno["text"],font="arial 14", bg="lightgreen",fg="#000").place(x=70,y=440)


    Label(nr1,text="Beneficiary Name",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=300,y=390)
    Label(nr1,text=bname["text"],font="arial 14", bg="lightgreen",fg="#000").place(x=300,y=440)

    Label(nr1,text="Credit Account Details",font="arial 14 bold", bg="lightgreen",fg="navyblue").place(x=50,y=350)
Label(nr1,text="____________________________________________________________________________
_______________________________________________________________________________________
___________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=415)

    Label(nr1,text='''Inter Bank
    Fund Transfer''',font="comicsans 14",fg="black",bg="lightgreen").place(x=860,y=440)

    Label(nr1,text="Transfer Type",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=860,y=390)

    Label(nr1,text="Purpose",font="arial 14 bold", bg="lightgreen",fg="#000").place(x=600,y=390)
    Label(nr1,text=bills.get(),font="arial 14", bg="lightgreen",fg="#000").place(x=600,y=440)

    Label(nr1,text="Amount",font="comicsans 14 bold",fg="black",bg="lightgreen").place(x=1050,y=390)
    Label(nr1,text=bilpay.get(),font="comicsans 14",fg="black",bg="lightgreen").place(x=1050,y=440)


Label(nr1,text="____________________________________________________________________________
_______________________________________________________________________________________
___________________________________________________________",bg="lightgreen",fg="grey").place(x=3
0,y=490)
    #entries
    #button
    Button(nr1,text="Back to Home Page",command=bil.destroy,font="comicsans 13
bold",bg="darkorange",fg="#fff",activebackground="orange",activeforeground="#fff",padx=10,pady=2).place(x=850,
y=600)

Button(f2,text="Login",font="comicsans 15",command=ulogin_config).grid(row=3,column=0,pady=20)
root.mainloop()
```

OUTPUT SNIPPETS

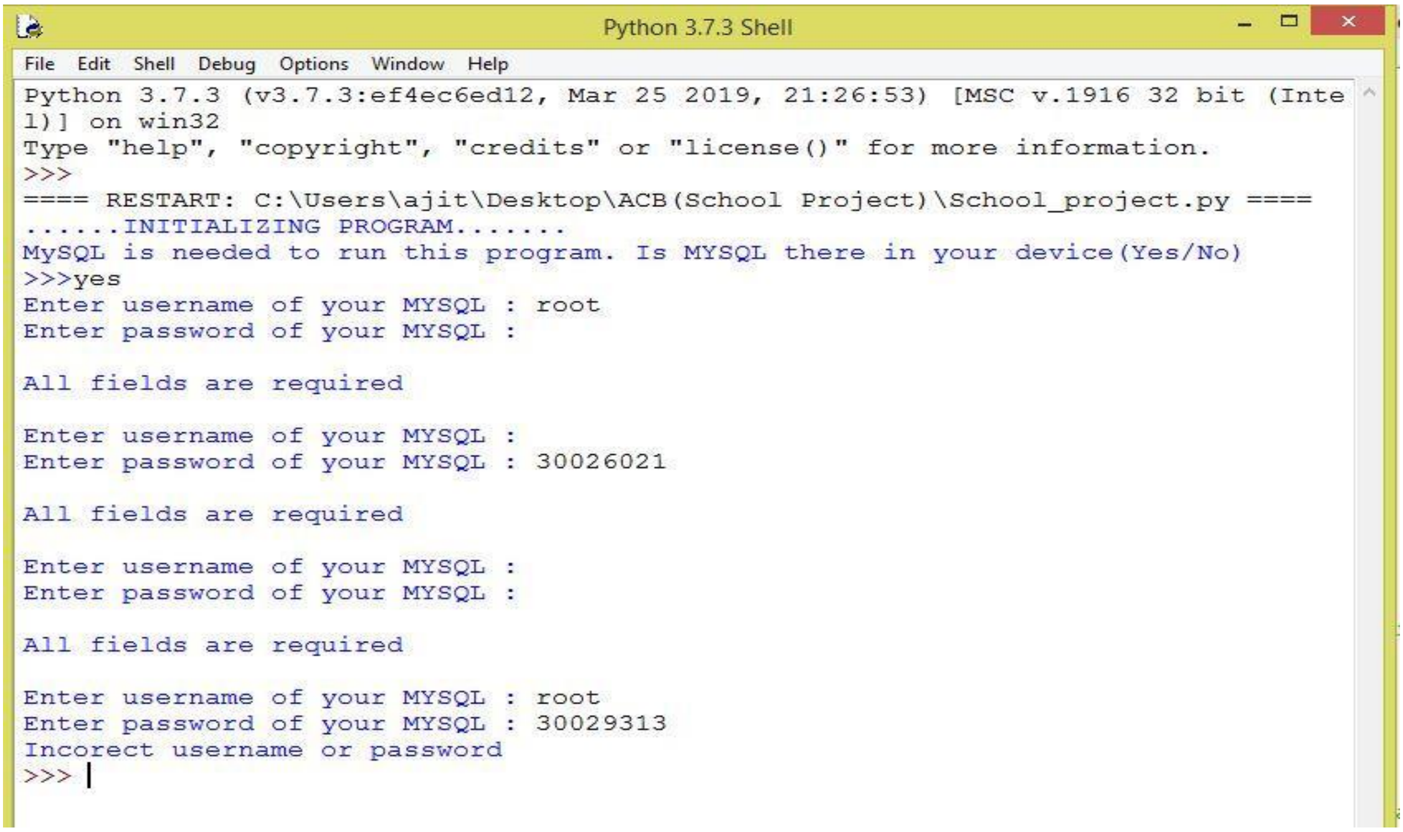
Initializing Program



```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte
l)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\ajit\Desktop\ACB(School Project)\School_project.py ====
......INITIALIZING PROGRAM.......
MySQL is needed to run this program. Is MYSQL there in your device(Yes/No)
>>>yes
Enter username of your MYSQL : root
Enter password of your MYSQL : 30026021
```

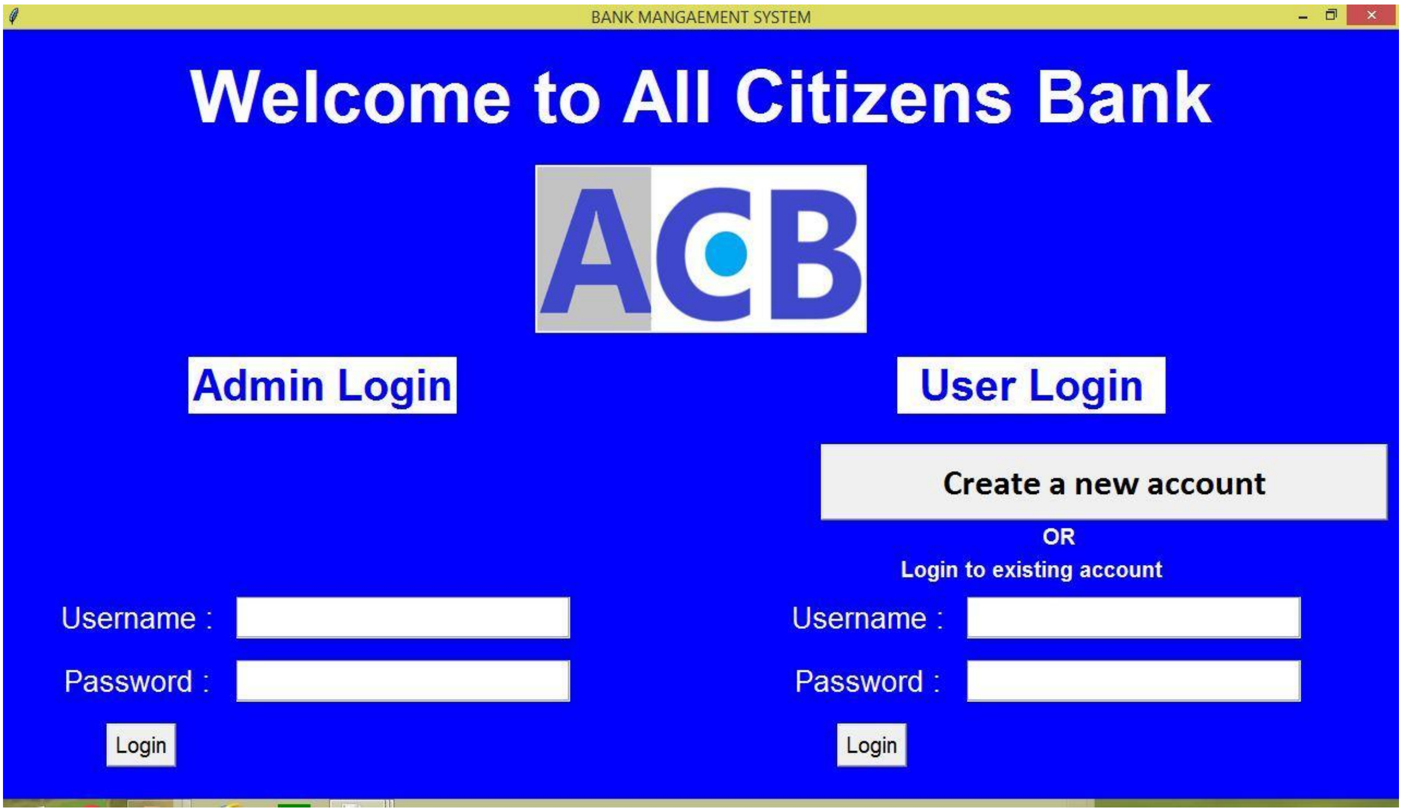


```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte
l)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\ajit\Desktop\ACB(School Project)\School_project.py ====
......INITIALIZING PROGRAM.......
MySQL is needed to run this program. Is MYSQL there in your device(Yes/No)
>>>no
You can't run this program without MYSQL. You need to install MYSQL in your devi
ce
>>>
```



```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte
l)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\ajit\Desktop\ACB(School Project)\School_project.py ====
......INITIALIZING PROGRAM.......
MySQL is needed to run this program. Is MYSQL there in your device(Yes/No)
>>>yes
Enter username of your MYSQL : root
Enter password of your MYSQL :

All fields are required

Enter username of your MYSQL :
Enter password of your MYSQL : 30026021

All fields are required

Enter username of your MYSQL :
Enter password of your MYSQL :

All fields are required

Enter username of your MYSQL : root
Enter password of your MYSQL : 30029313
Incorect username or password
>>>
```

Home Page



BANK MANGAEMENT SYSTEM

Welcome to All Citizens Bank

Admin Login

User Login

Create a new account

OR

Login to existing account

Username :

Password :

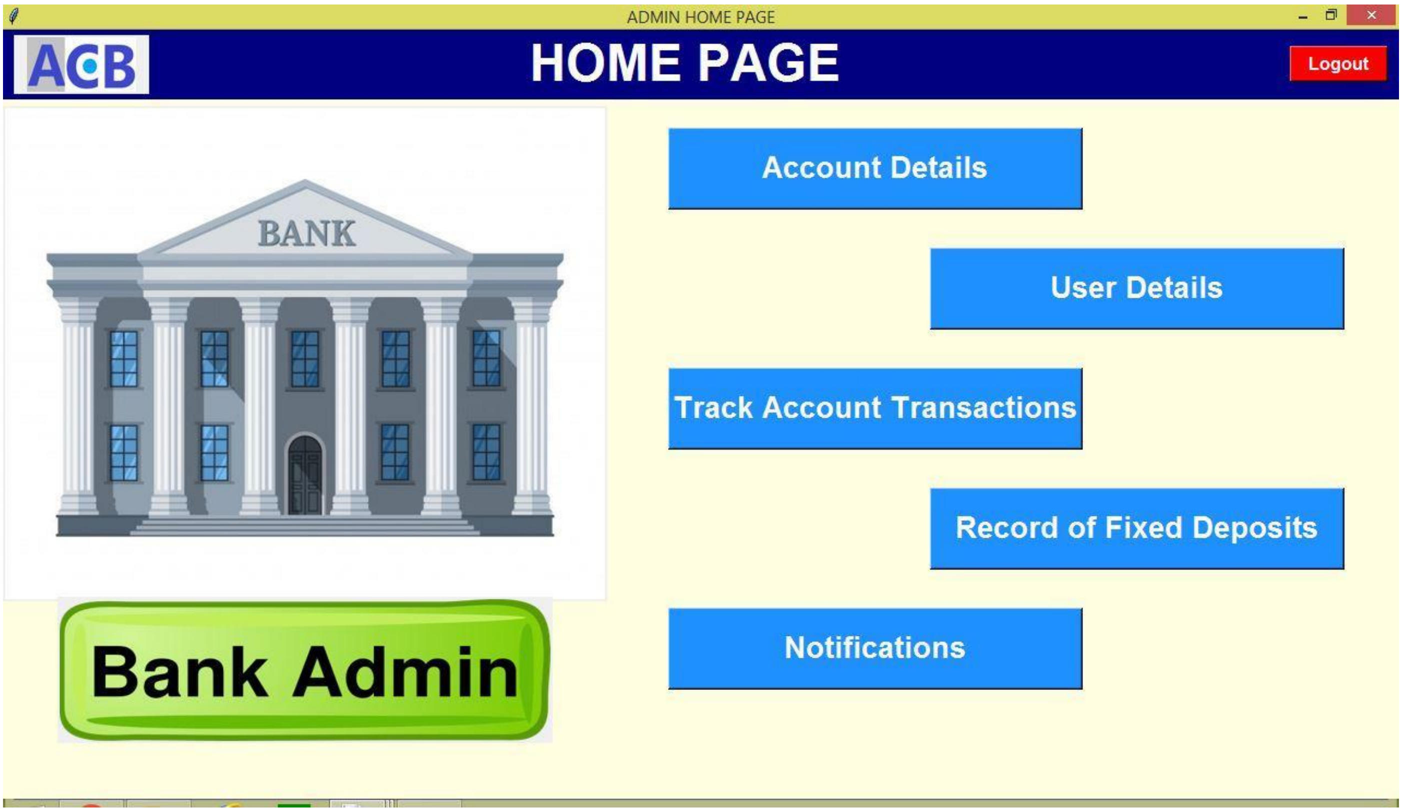
Login

Username :

Password :

Login

Admin Section



ADMIN HOME PAGE

HOME PAGE

Account Details

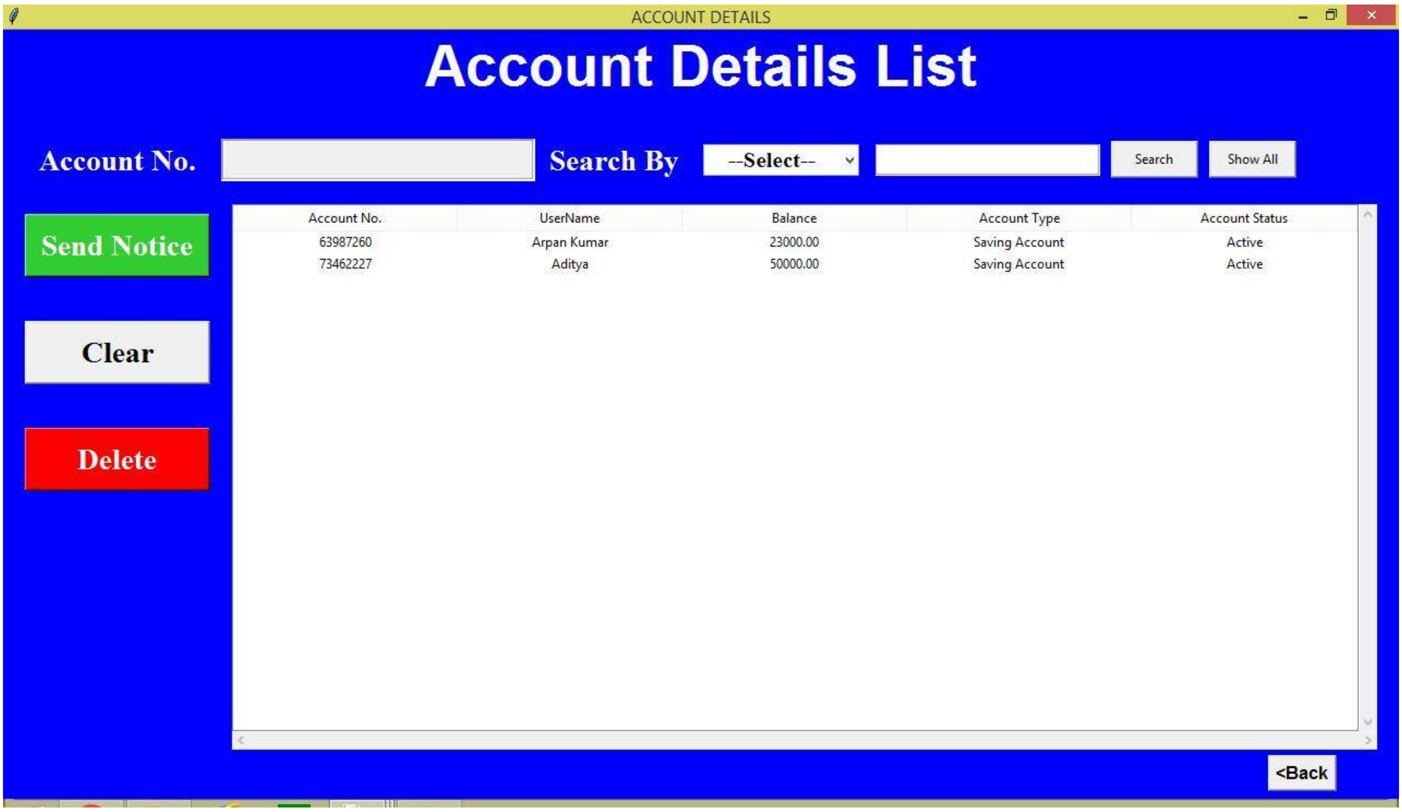
User Details

Track Account Transactions

Record of Fixed Deposits

Notifications

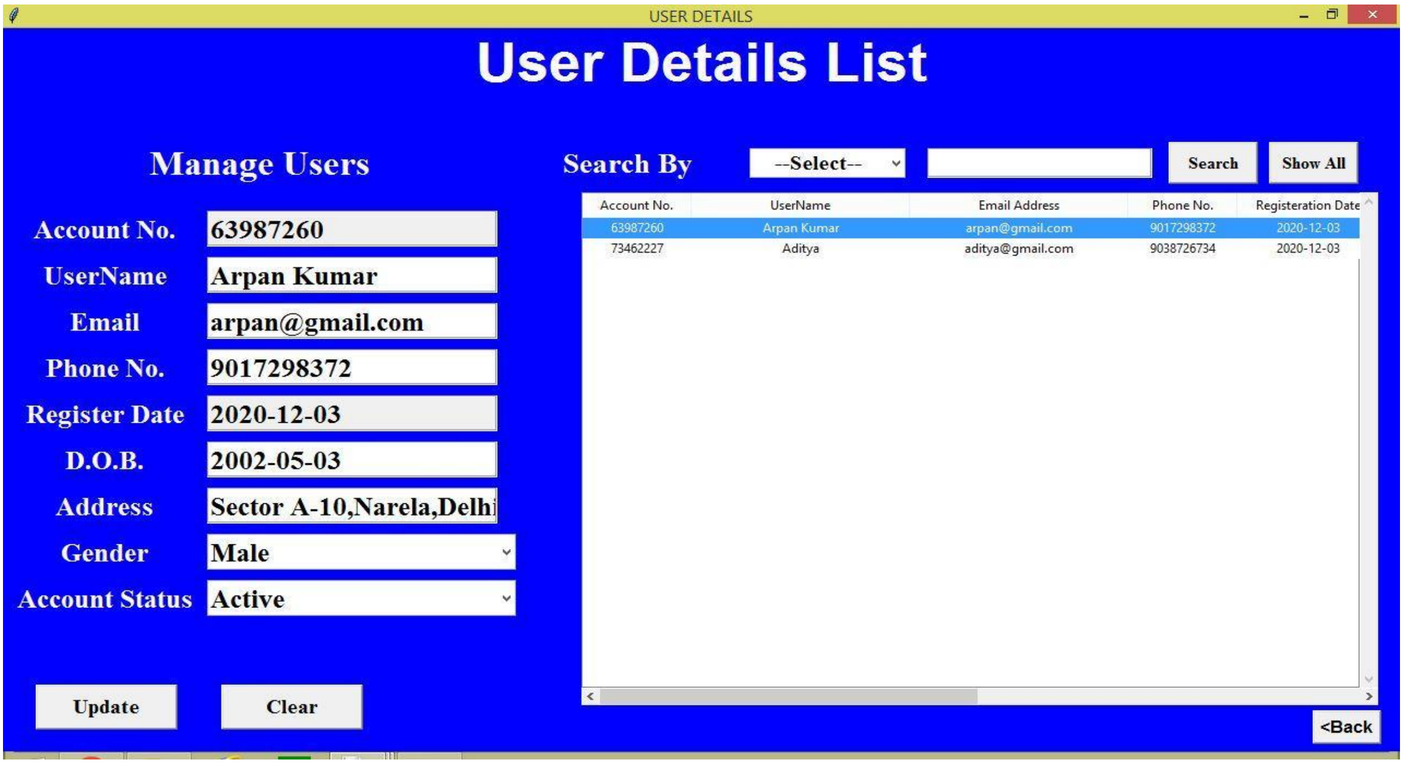
Account Details



Account Details List

Account No.	UserName	Balance	Account Type	Account Status
63987260	Arpan Kumar	23000.00	Saving Account	Active
73462227	Aditya	50000.00	Saving Account	Active

User Details



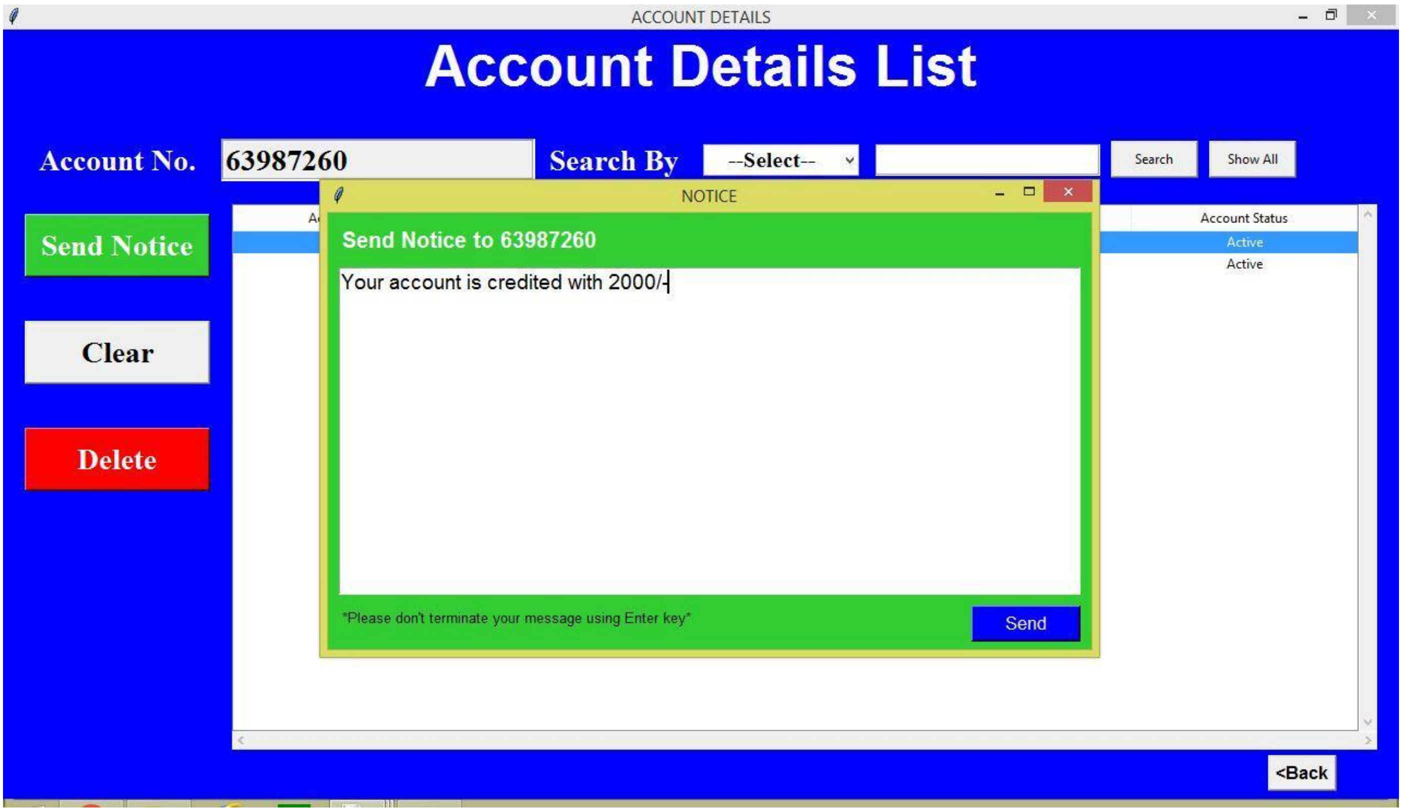
User Details List

Manage Users

Account No.	63987260
UserName	Arpan Kumar
Email	arpan@gmail.com
Phone No.	9017298372
Register Date	2020-12-03
D.O.B.	2002-05-03
Address	Sector A-10,Narela,Delhi
Gender	Male
Account Status	Active

Account No.	UserName	Email Address	Phone No.	Registeration Date
63987260	Arpan Kumar	arpan@gmail.com	9017298372	2020-12-03
73462227	Aditya	aditya@gmail.com	9038726734	2020-12-03

Notice



Account Details List

Account No. 63987260

Search By --Select--

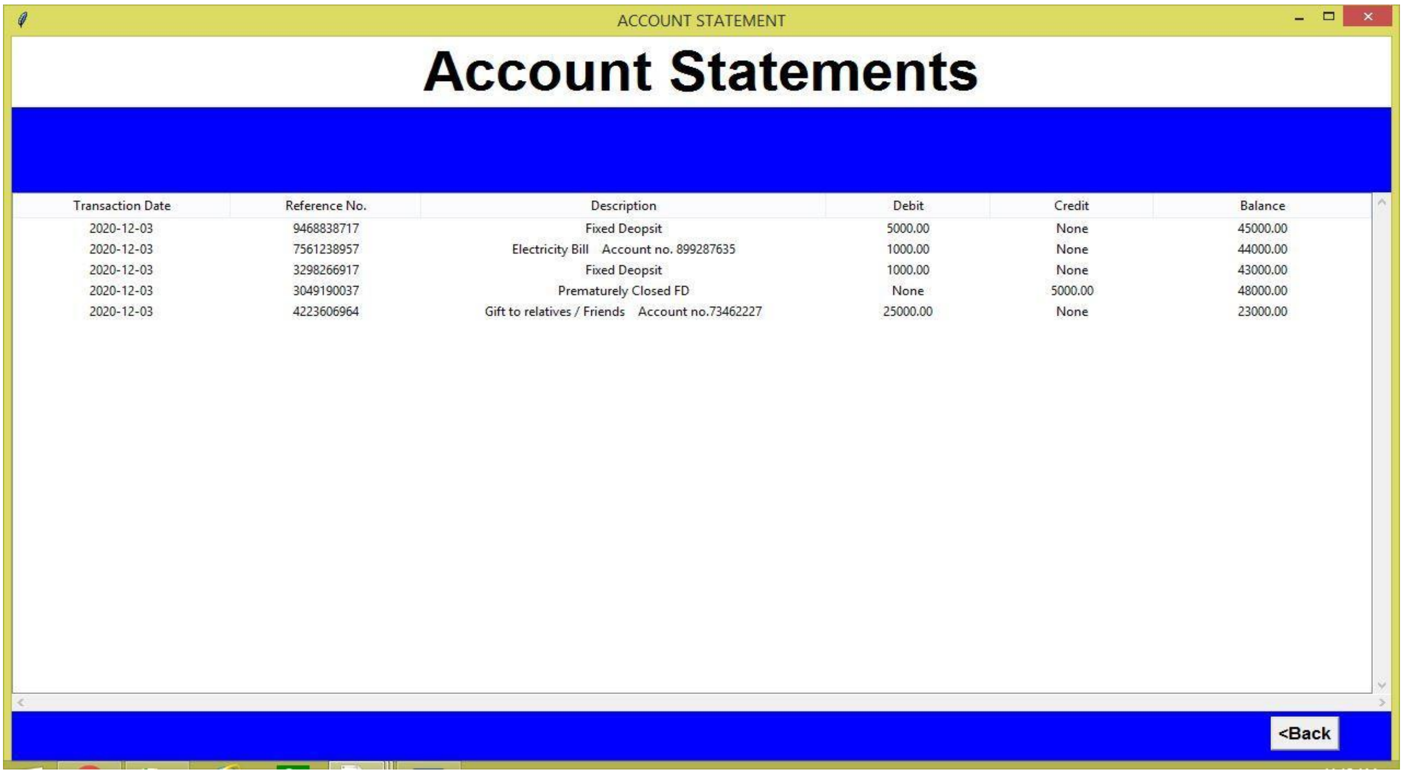
Account Status
Active
Active

Send Notice to 63987260

Your account is credited with 2000/-

Please don't terminate your message using Enter key

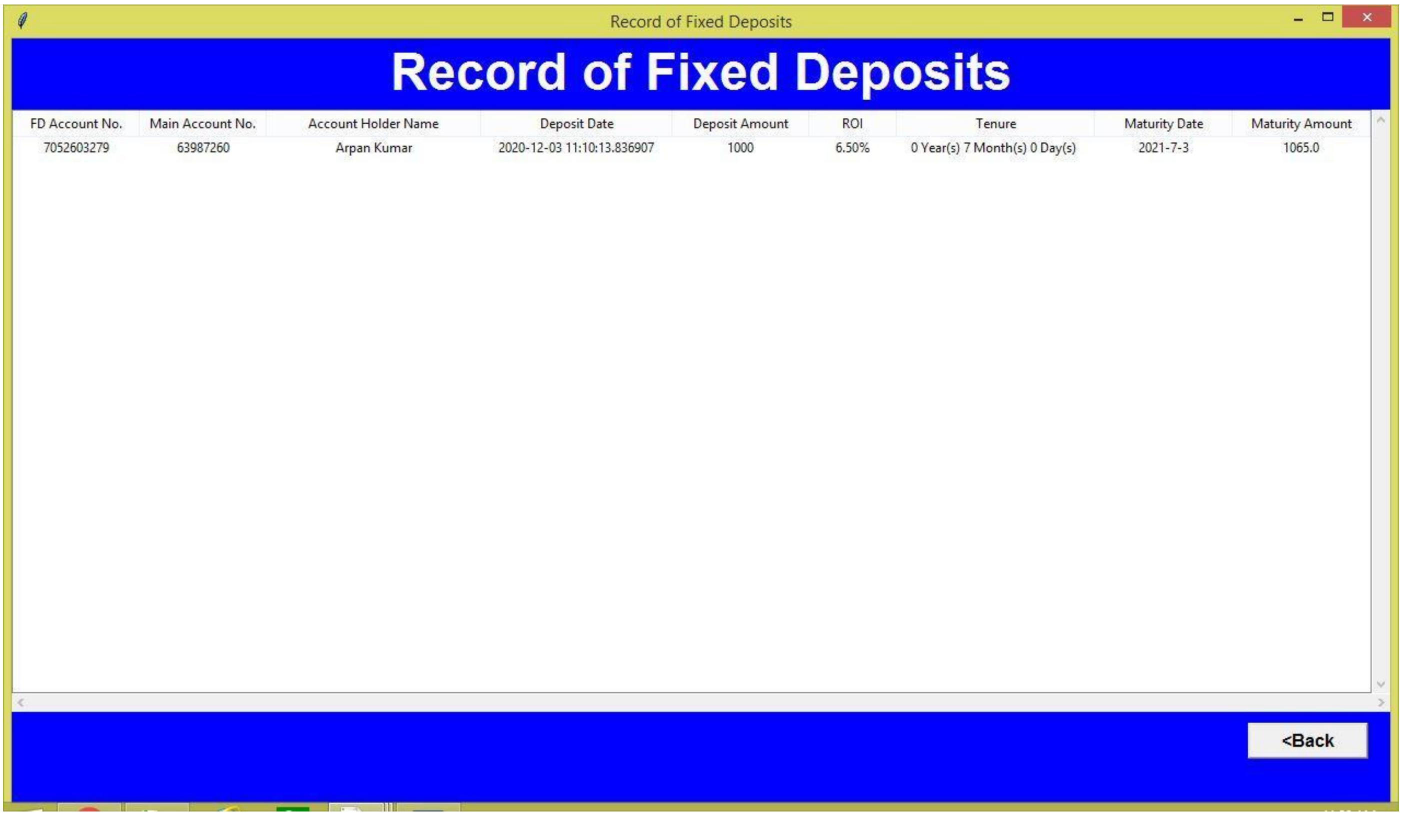
Account Statements



Account Statements

Transaction Date	Reference No.	Description	Debit	Credit	Balance
2020-12-03	9468838717	Fixed Deopsit	5000.00	None	45000.00
2020-12-03	7561238957	Electricity Bill Account no. 899287635	1000.00	None	44000.00
2020-12-03	3298266917	Fixed Deopsit	1000.00	None	43000.00
2020-12-03	3049190037	Prematurely Closed FD	None	5000.00	48000.00
2020-12-03	4223606964	Gift to relatives / Friends Account no.73462227	25000.00	None	23000.00

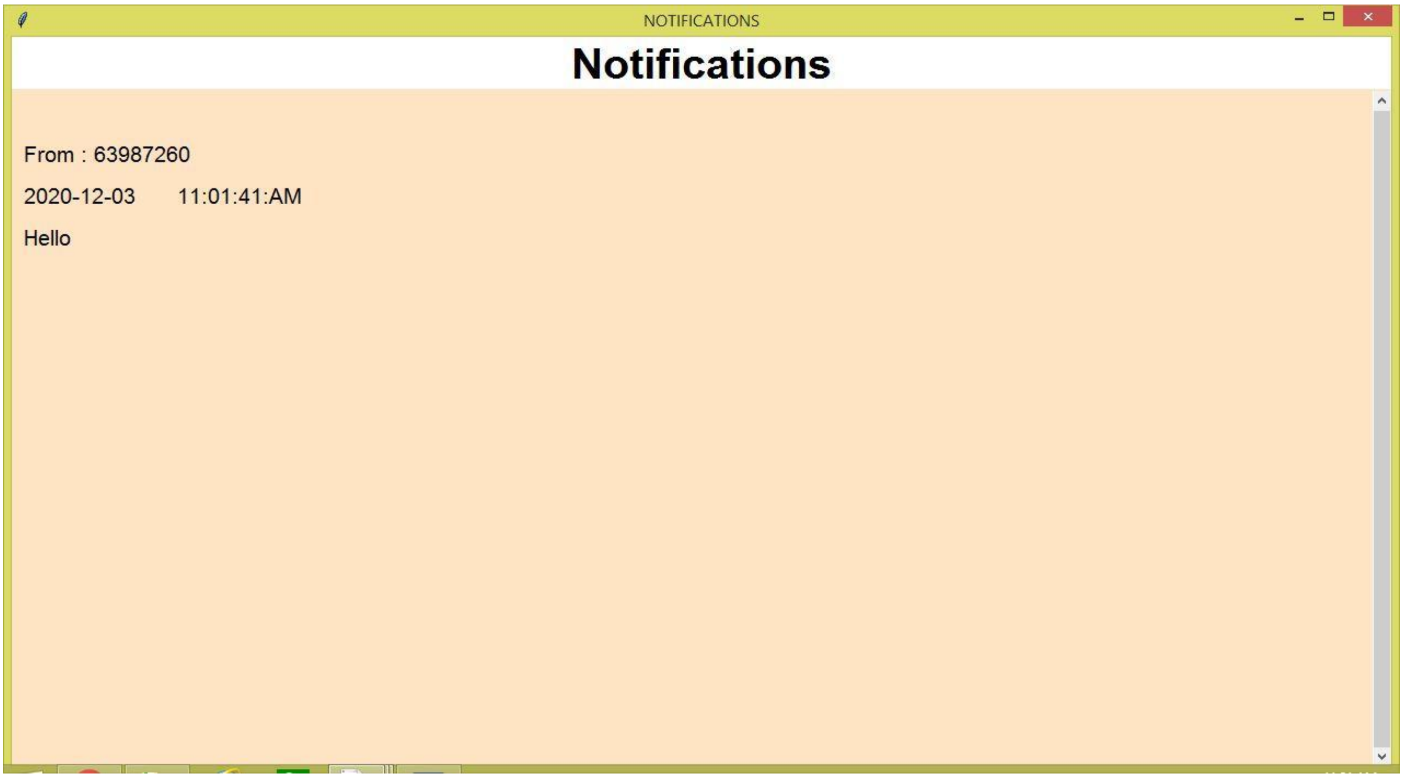
Fixed Deposits



Record of Fixed Deposits

FD Account No.	Main Account No.	Account Holder Name	Deposit Date	Deposit Amount	ROI	Tenure	Maturity Date	Maturity Amount
7052603279	63987260	Arpan Kumar	2020-12-03 11:10:13.836907	1000	6.50%	0 Year(s) 7 Month(s) 0 Day(s)	2021-7-3	1065.0

Notifications



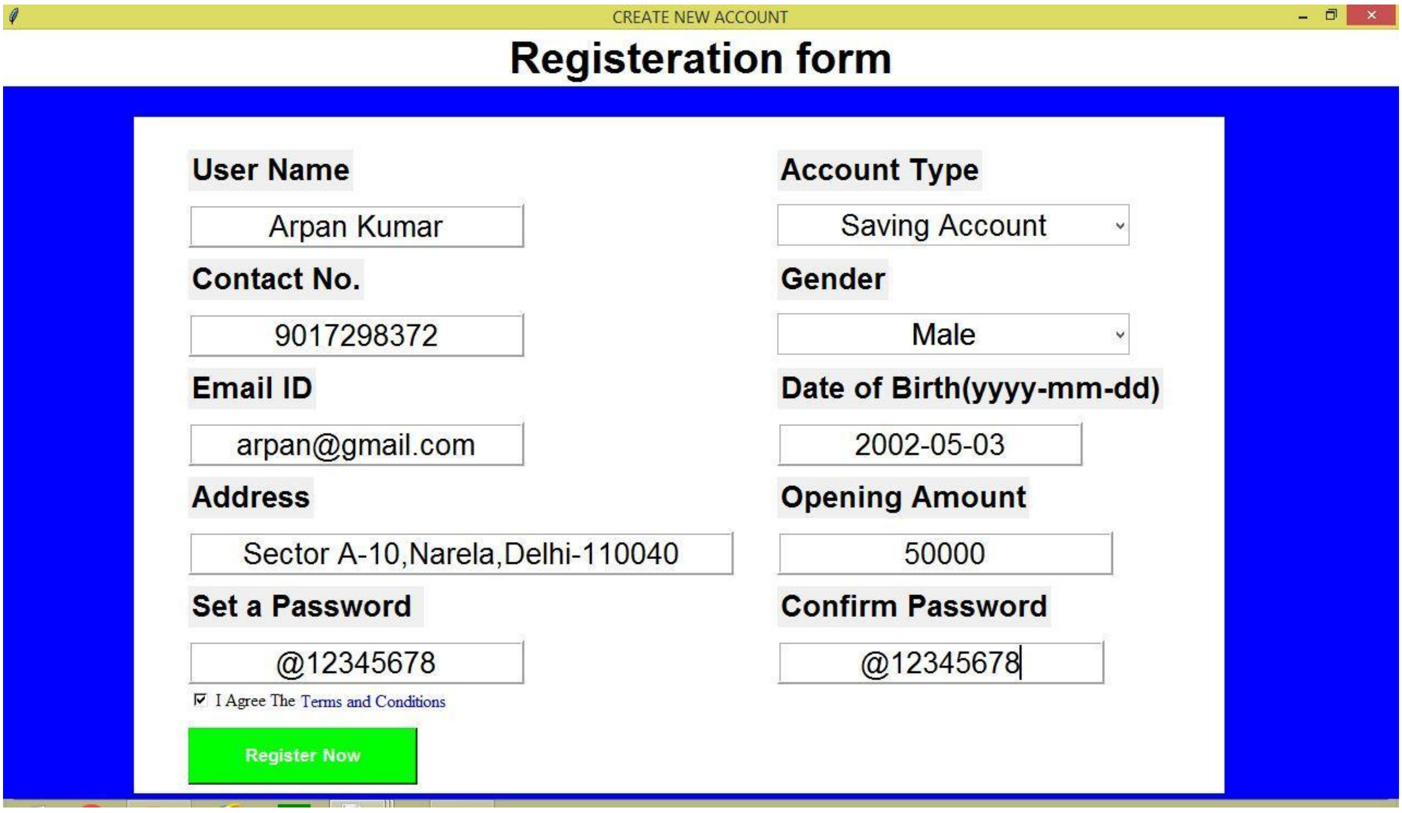
Notifications

From : 63987260

2020-12-03 11:01:41:AM

Hello

Registeration Form

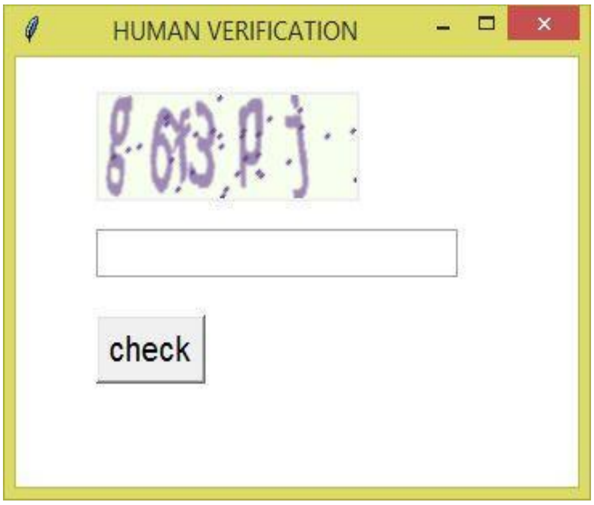


CREATE NEW ACCOUNT

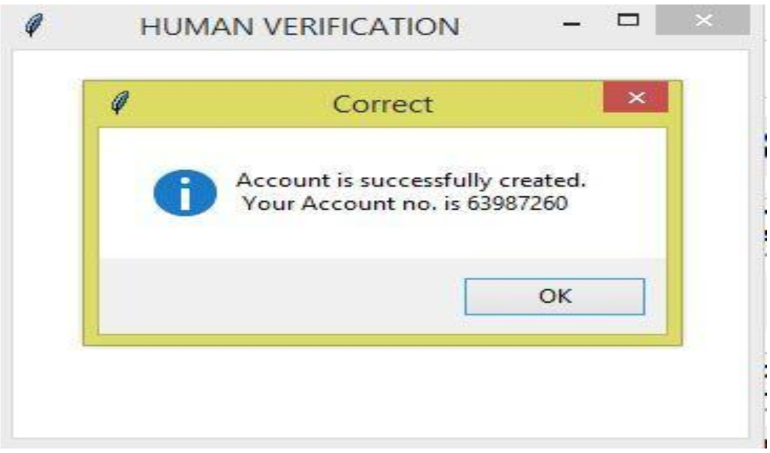
Registeration form

User Name	Account Type
Arpan Kumar	Saving Account
Contact No.	Gender
9017298372	Male
Email ID	Date of Birth(yyyy-mm-dd)
arpan@gmail.com	2002-05-03
Address	Opening Amount
Sector A-10,Narela,Delhi-110040	50000
Set a Password	Confirm Password
@12345678	@12345678

☑ I Agree The Terms and Conditions



HUMAN VERIFICATION

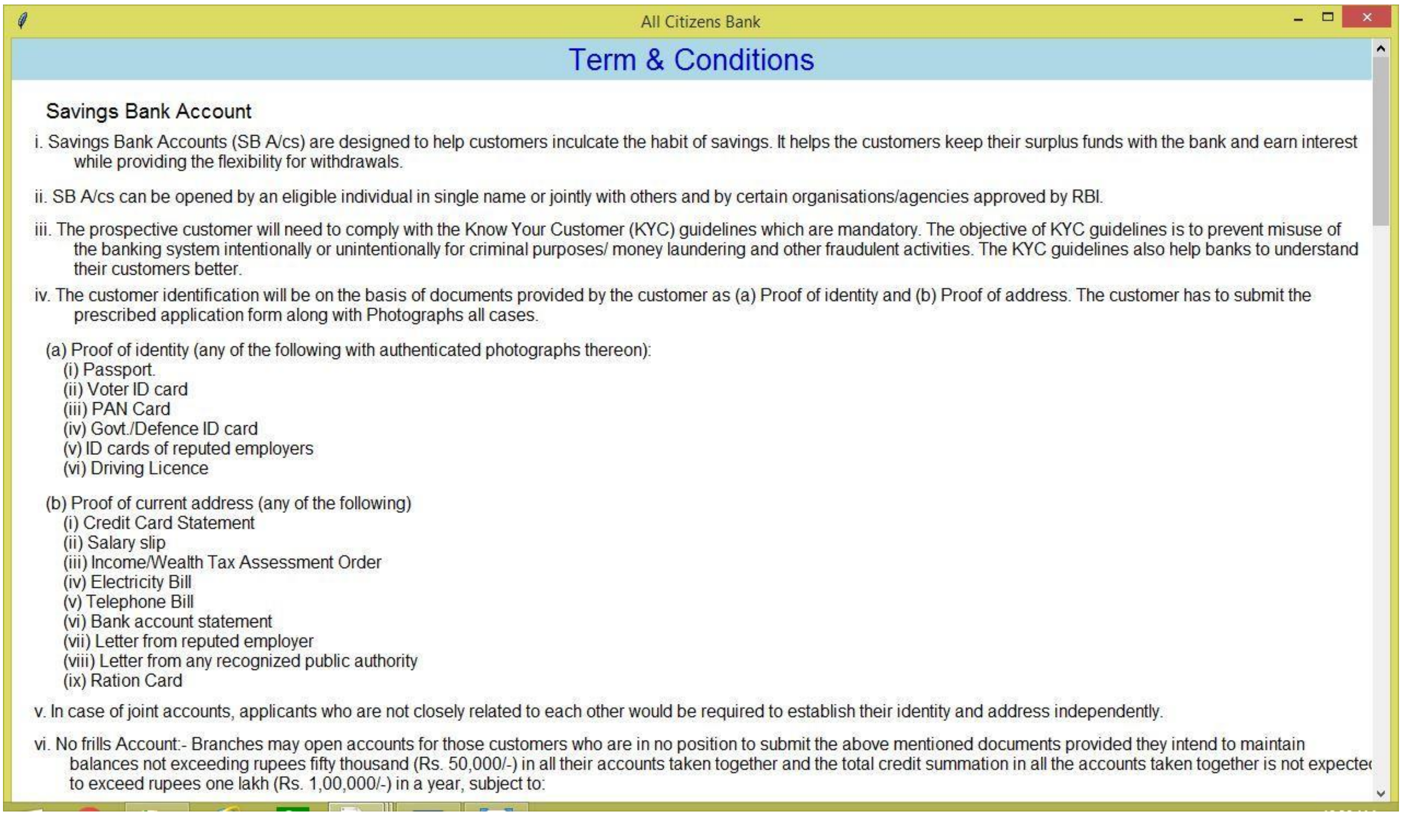


HUMAN VERIFICATION

Correct

Account is successfully created.
Your Account no. is 63987260

Terms and Conditions



All Citizens Bank

Term & Conditions

Savings Bank Account

i. Savings Bank Accounts (SB A/cs) are designed to help customers inculcate the habit of savings. It helps the customers keep their surplus funds with the bank and earn interest while providing the flexibility for withdrawals.

ii. SB A/cs can be opened by an eligible individual in single name or jointly with others and by certain organisations/agencies approved by RBI.

iii. The prospective customer will need to comply with the Know Your Customer (KYC) guidelines which are mandatory. The objective of KYC guidelines is to prevent misuse of the banking system intentionally or unintentionally for criminal purposes/ money laundering and other fraudulent activities. The KYC guidelines also help banks to understand their customers better.

iv. The customer identification will be on the basis of documents provided by the customer as (a) Proof of identity and (b) Proof of address. The customer has to submit the prescribed application form along with Photographs all cases.

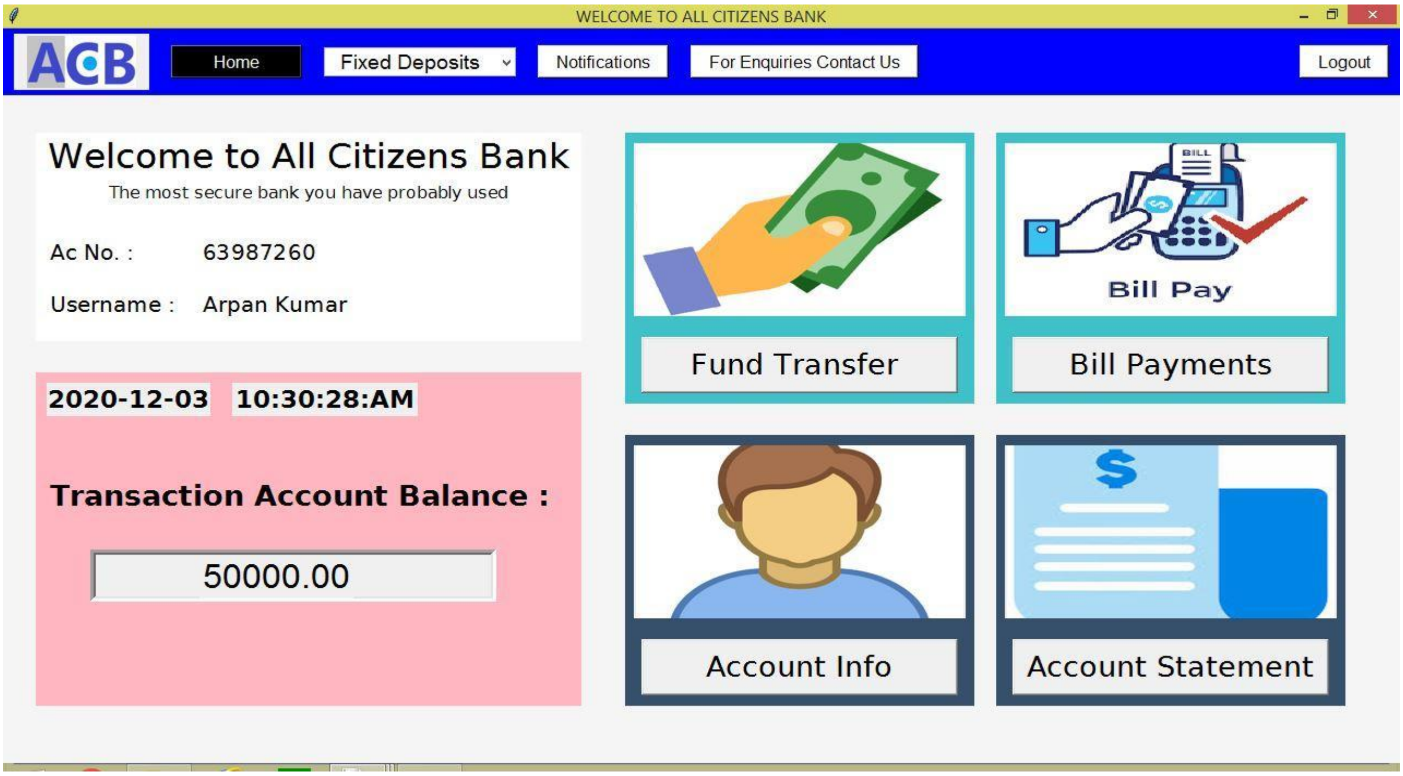
(a) Proof of identity (any of the following with authenticated photographs thereon):
(i) Passport.
(ii) Voter ID card
(iii) PAN Card
(iv) Govt./Defence ID card
(v) ID cards of reputed employers
(vi) Driving Licence

(b) Proof of current address (any of the following)
(i) Credit Card Statement
(ii) Salary slip
(iii) Income/Wealth Tax Assessment Order
(iv) Electricity Bill
(v) Telephone Bill
(vi) Bank account statement
(vii) Letter from reputed employer
(viii) Letter from any recognized public authority
(ix) Ration Card

v. In case of joint accounts, applicants who are not closely related to each other would be required to establish their identity and address independently.

vi. No frills Account:- Branches may open accounts for those customers who are in no position to submit the above mentioned documents provided they intend to maintain balances not exceeding rupees fifty thousand (Rs. 50,000/-) in all their accounts taken together and the total credit summation in all the accounts taken together is not expecte to exceed rupees one lakh (Rs. 1,00,000/-) in a year, subject to:

User Section



WELCOME TO ALL CITIZENS BANK

Welcome to All Citizens Bank

The most secure bank you have probably used

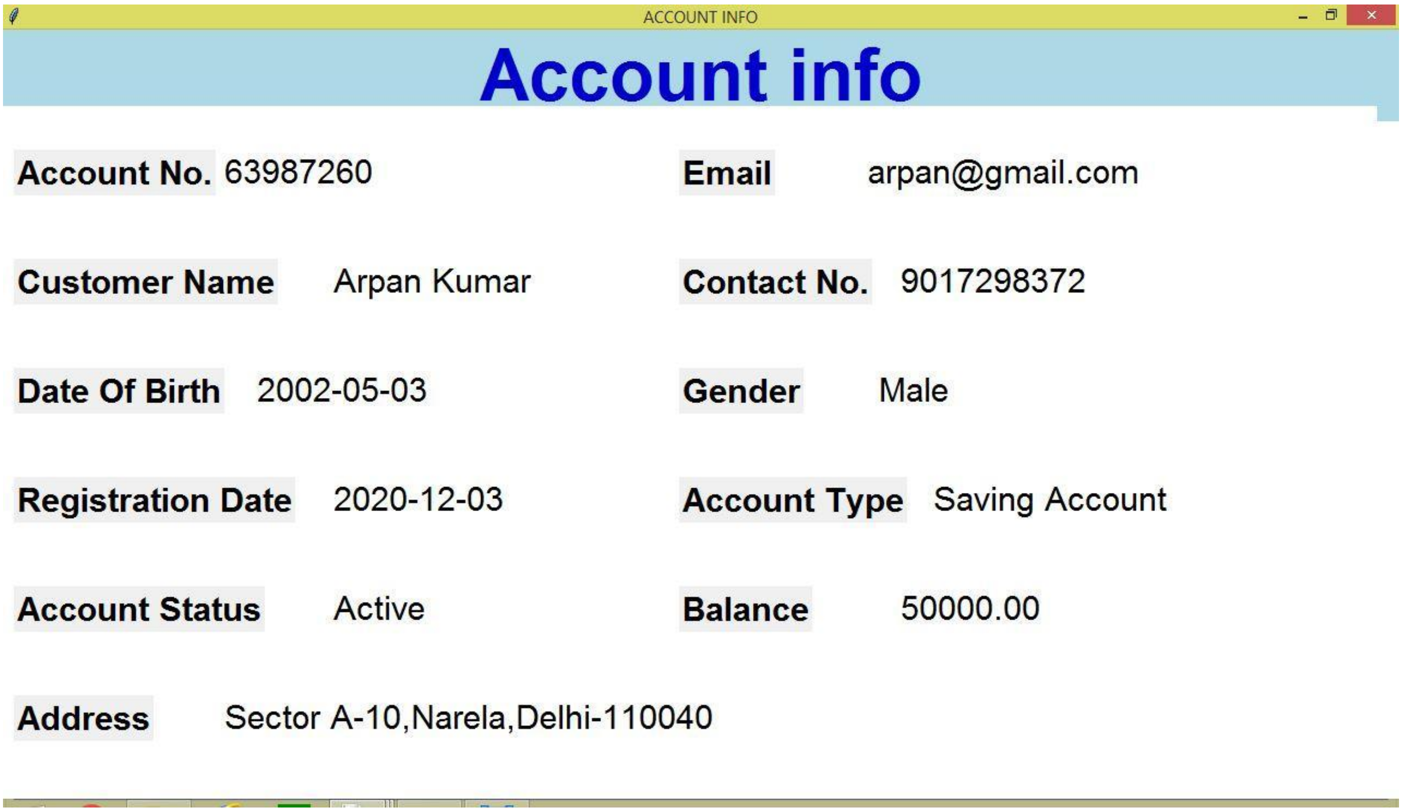
Ac No. : 63987260

Username : Arpan Kumar

2020-12-03 10:30:28:AM

Transaction Account Balance : 50000.00

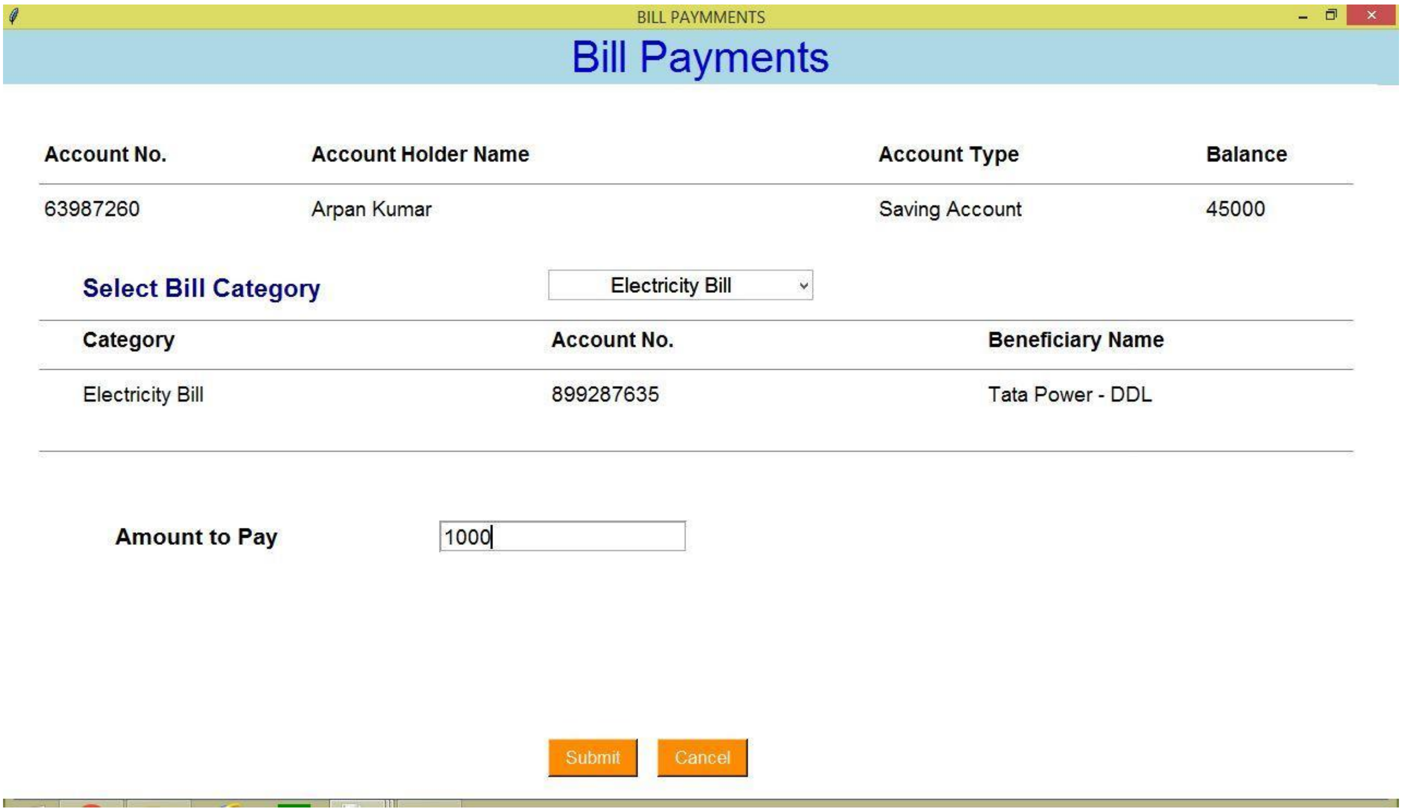
Account info



Account info

Account No.	63987260	**Email**	arpan@gmail.com
Customer Name	Arpan Kumar	**Contact No.**	9017298372
Date Of Birth	2002-05-03	**Gender**	Male
Registration Date	2020-12-03	**Account Type**	Saving Account
Account Status	Active	**Balance**	50000.00
Address	Sector A-10,Narela,Delhi-110040		

Bill Payments



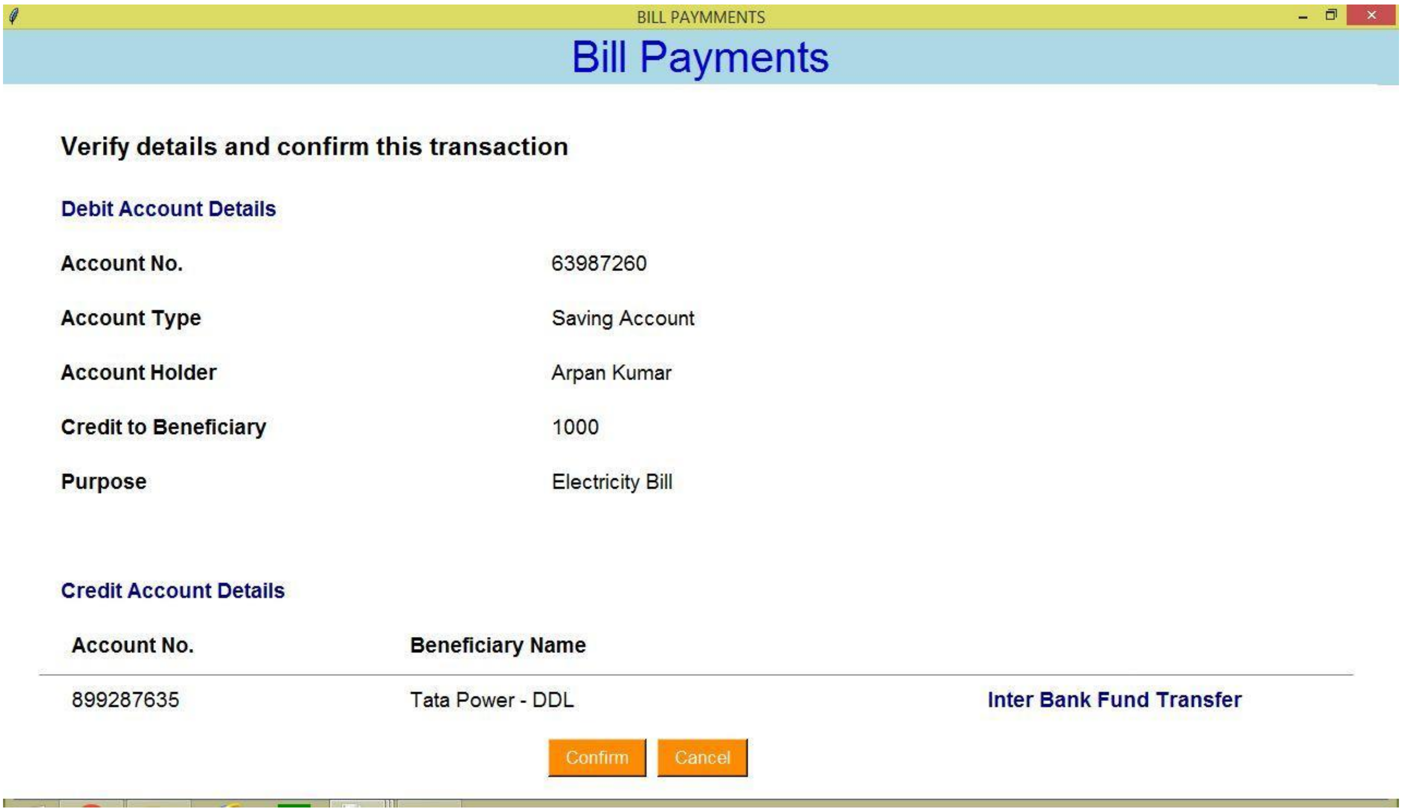
Bill Payments

Account No.	Account Holder Name	Account Type	Balance
63987260	Arpan Kumar	Saving Account	45000

Select Bill Category Electricity Bill

Category	Account No.	Beneficiary Name
Electricity Bill	899287635	Tata Power - DDL

Amount to Pay 1000



Bill Payments

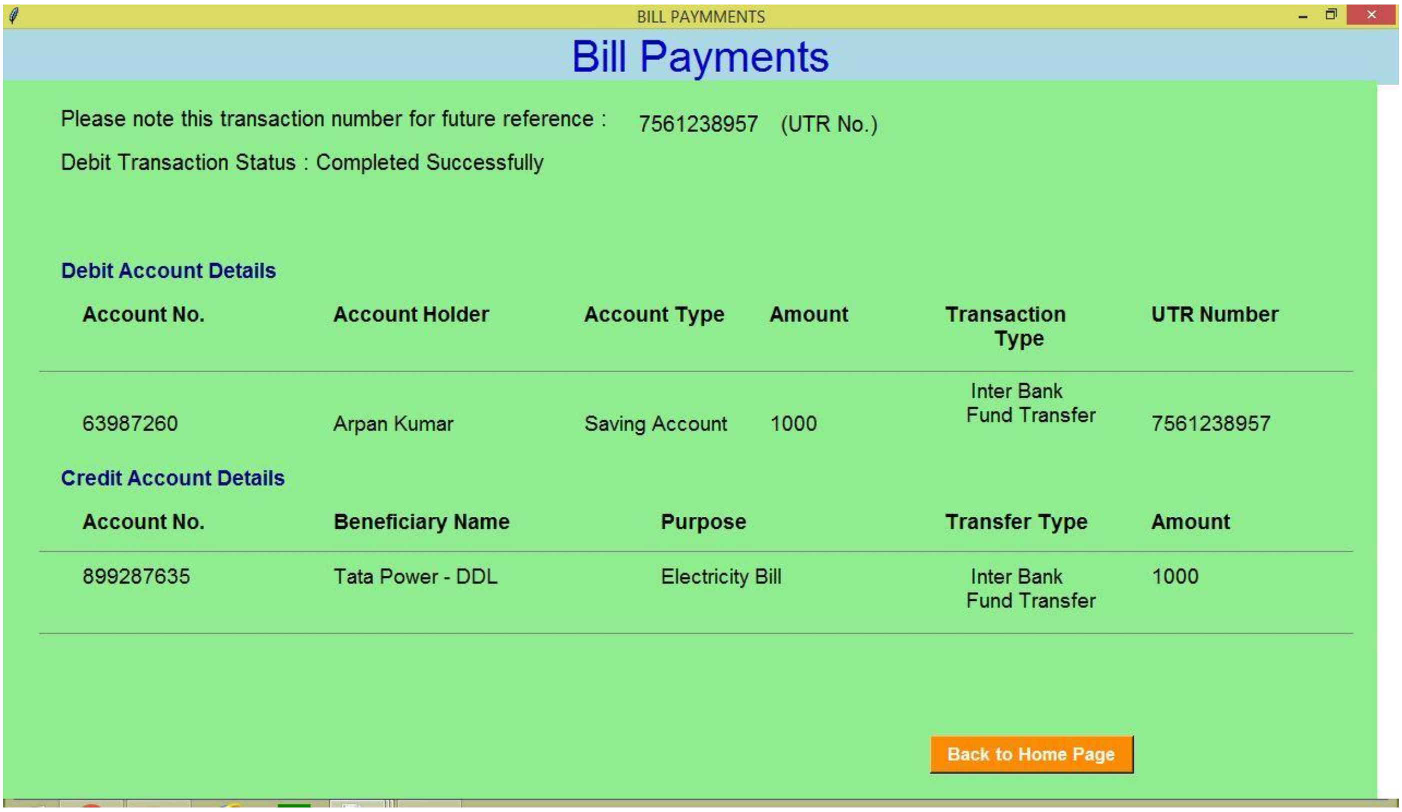
Verify details and confirm this transaction

Debit Account Details

Account No.	63987260
Account Type	Saving Account
Account Holder	Arpan Kumar
Credit to Beneficiary	1000
Purpose	Electricity Bill

Credit Account Details

Account No.	Beneficiary Name	
899287635	Tata Power - DDL	**Inter Bank Fund Transfer**



Bill Payments

Please note this transaction number for future reference : 7561238957 (UTR No.)

Debit Transaction Status : Completed Successfully

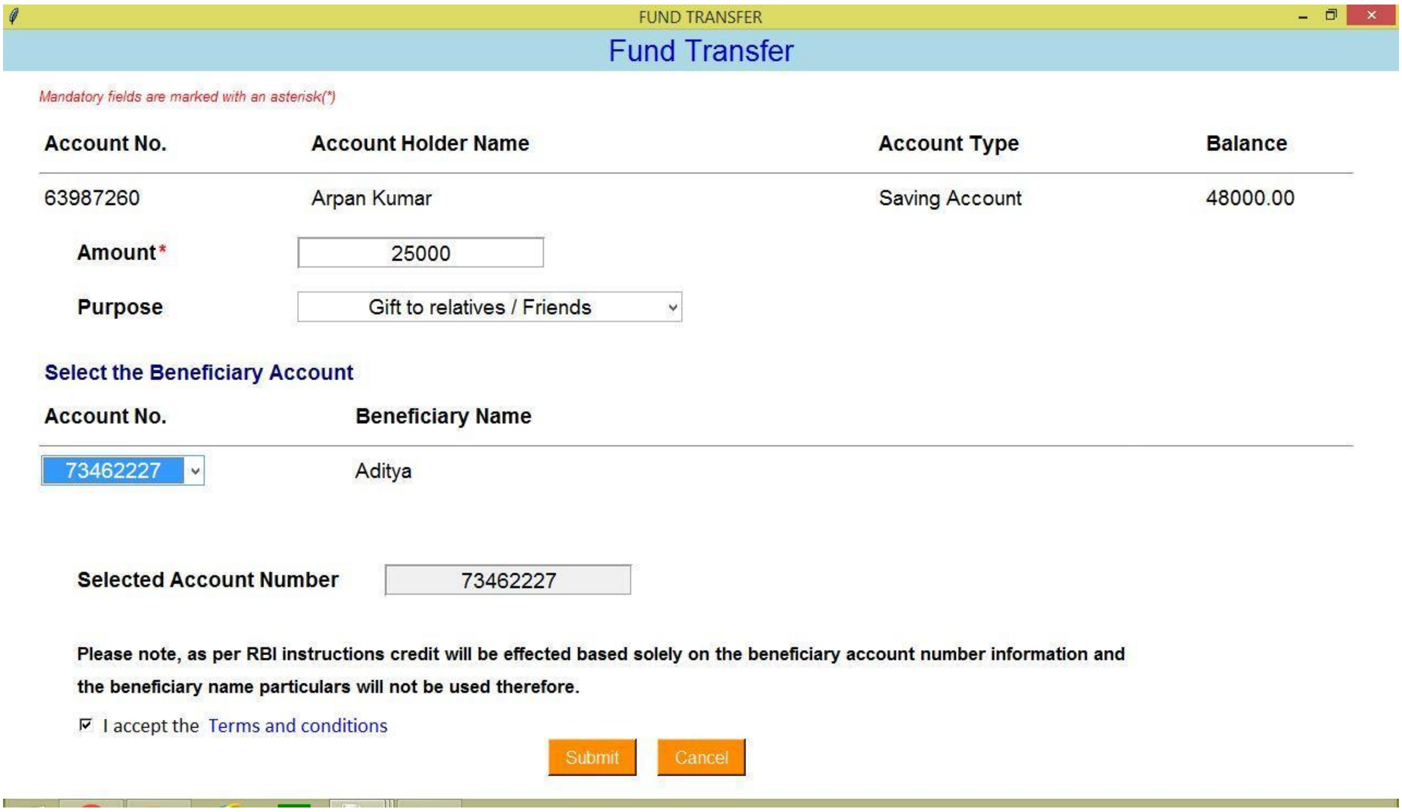
Debit Account Details

Account No.	Account Holder	Account Type	Amount	Transaction Type	UTR Number
63987260	Arpan Kumar	Saving Account	1000	Inter Bank Fund Transfer	7561238957

Credit Account Details

Account No.	Beneficiary Name	Purpose	Transfer Type	Amount
899287635	Tata Power - DDL	Electricity Bill	Inter Bank Fund Transfer	1000

Fund Transfer



Fund Transfer

Mandatory fields are marked with an asterisk(*)

Account No.	Account Holder Name	Account Type	Balance
63987260	Arpan Kumar	Saving Account	48000.00

Amount* 25000

Purpose Gift to relatives / Friends

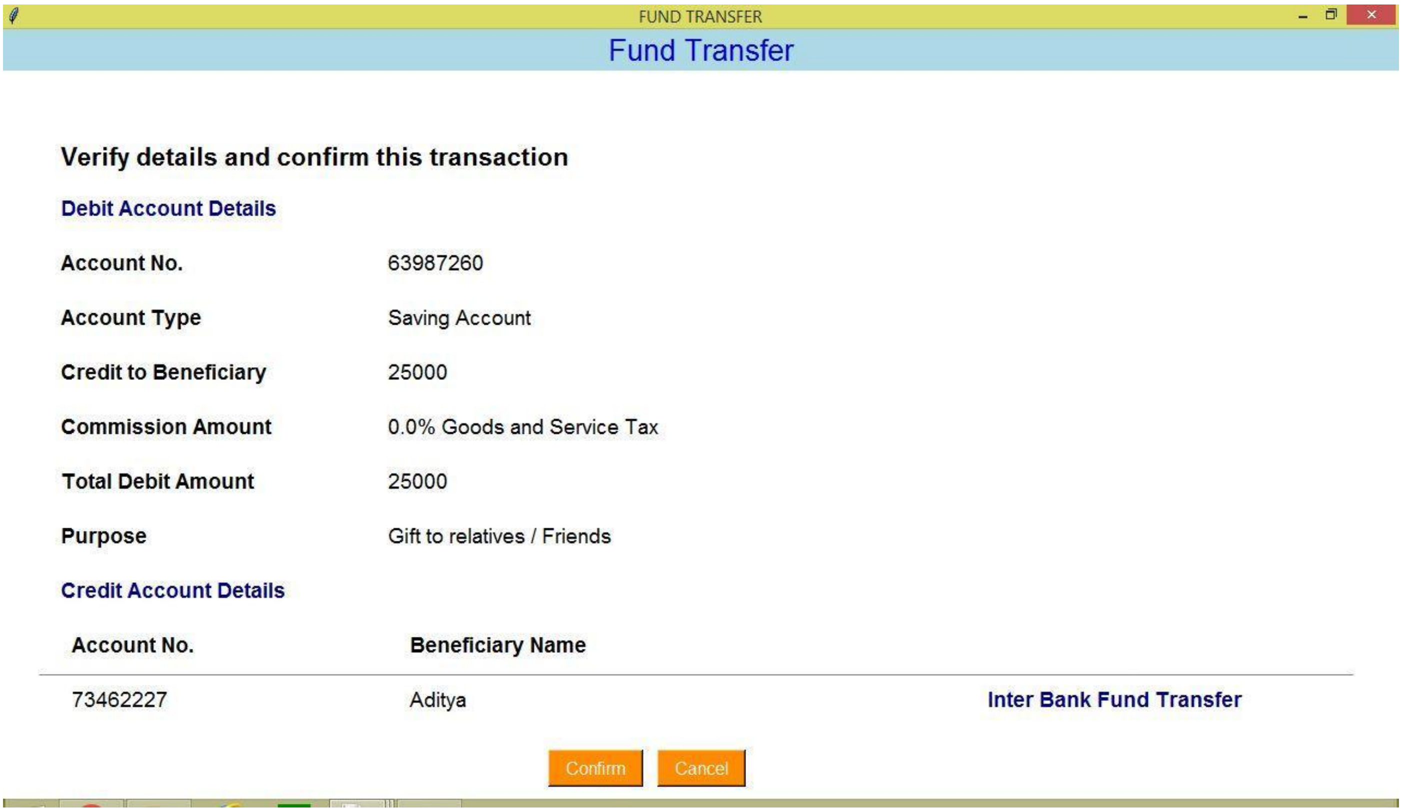
Select the Beneficiary Account

Account No.	Beneficiary Name
73462227	Aditya

Selected Account Number 73462227

Please note, as per RBI instructions credit will be effected based solely on the beneficiary account number information and the beneficiary name particulars will not be used therefore.

☑ I accept the Terms and conditions



Fund Transfer

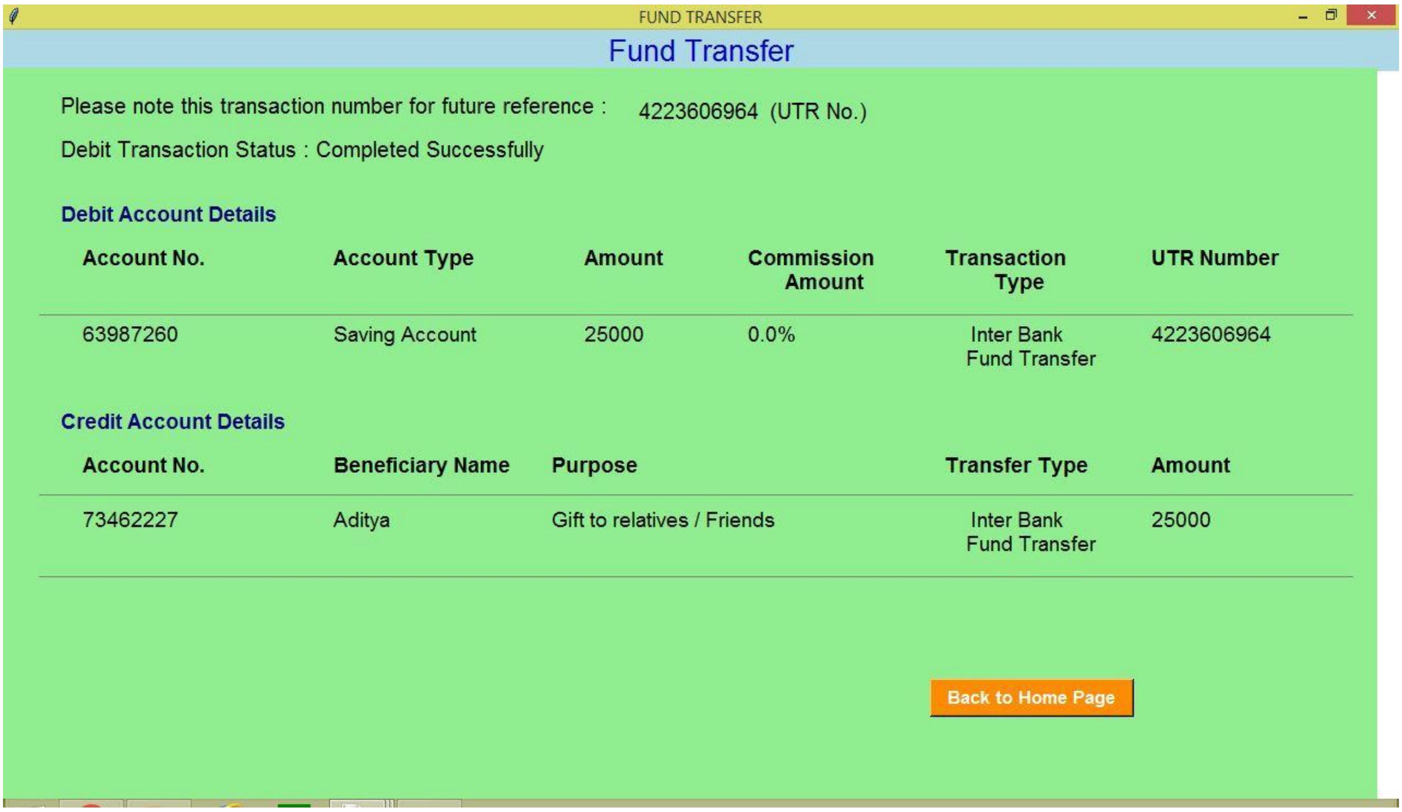
Verify details and confirm this transaction

Debit Account Details

Account No.	63987260
Account Type	Saving Account
Credit to Beneficiary	25000
Commission Amount	0.0% Goods and Service Tax
Total Debit Amount	25000
Purpose	Gift to relatives / Friends

Credit Account Details

Account No.	Beneficiary Name	
73462227	Aditya	**Inter Bank Fund Transfer**



Fund Transfer

Please note this transaction number for future reference : 4223606964 (UTR No.)

Debit Transaction Status : Completed Successfully

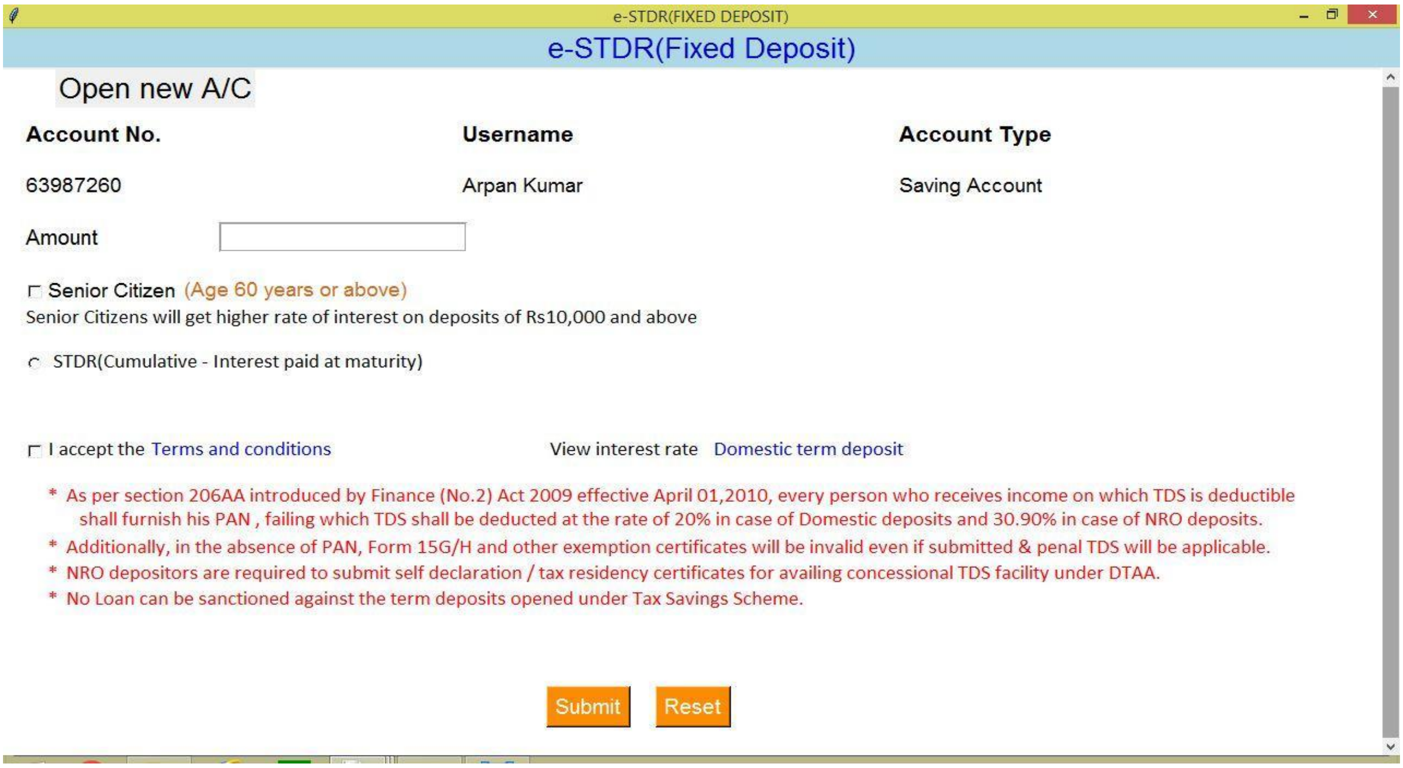
Debit Account Details

Account No.	Account Type	Amount	Commission Amount	Transaction Type	UTR Number
63987260	Saving Account	25000	0.0%	Inter Bank Fund Transfer	4223606964

Credit Account Details

Account No.	Beneficiary Name	Purpose	Transfer Type	Amount
73462227	Aditya	Gift to relatives / Friends	Inter Bank Fund Transfer	25000

Fixed Deposits



e-STDR(Fixed Deposit)

Open new A/C

Account No.	Username	Account Type
63987260	Arpan Kumar	Saving Account

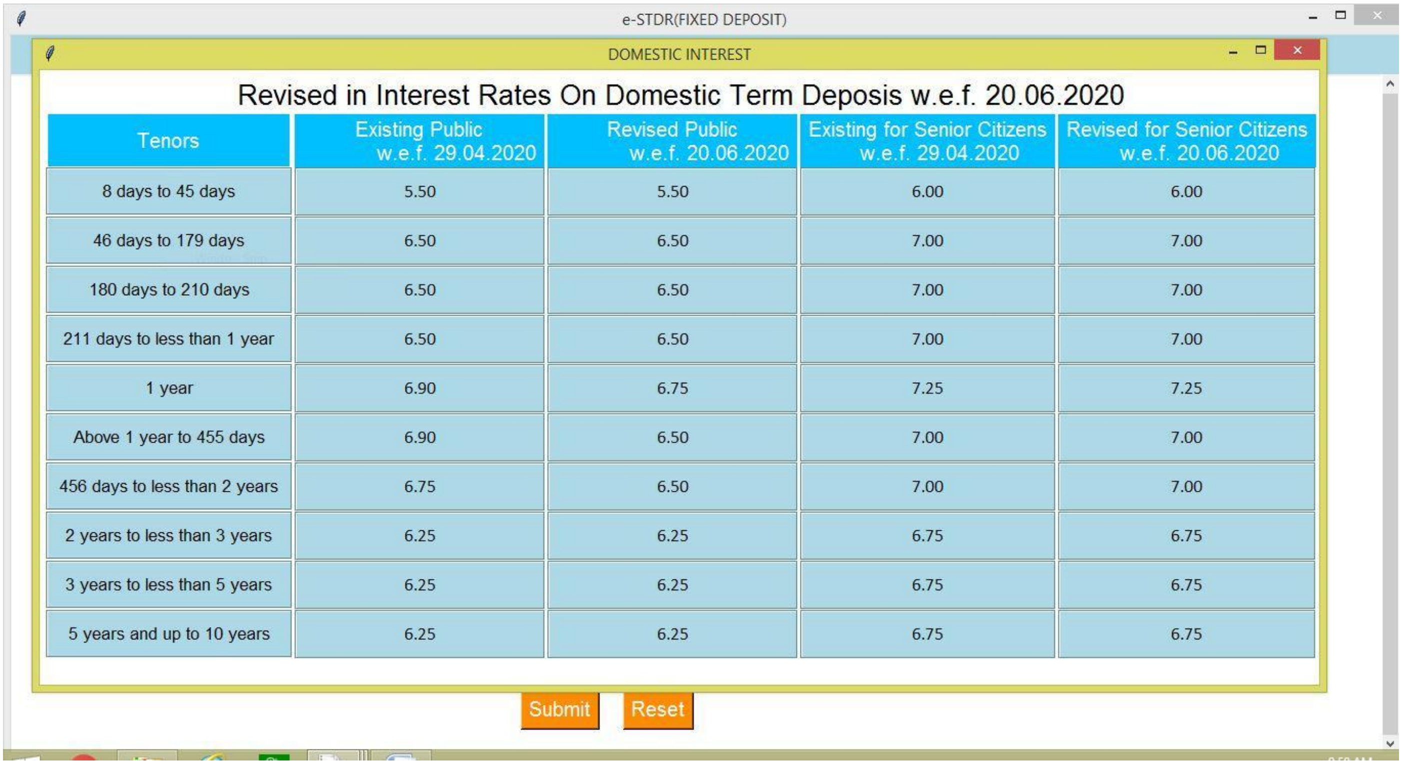
Amount

□ Senior Citizen (Age 60 years or above)
Senior Citizens will get higher rate of interest on deposits of Rs10,000 and above

○ STDR(Cumulative - Interest paid at maturity)

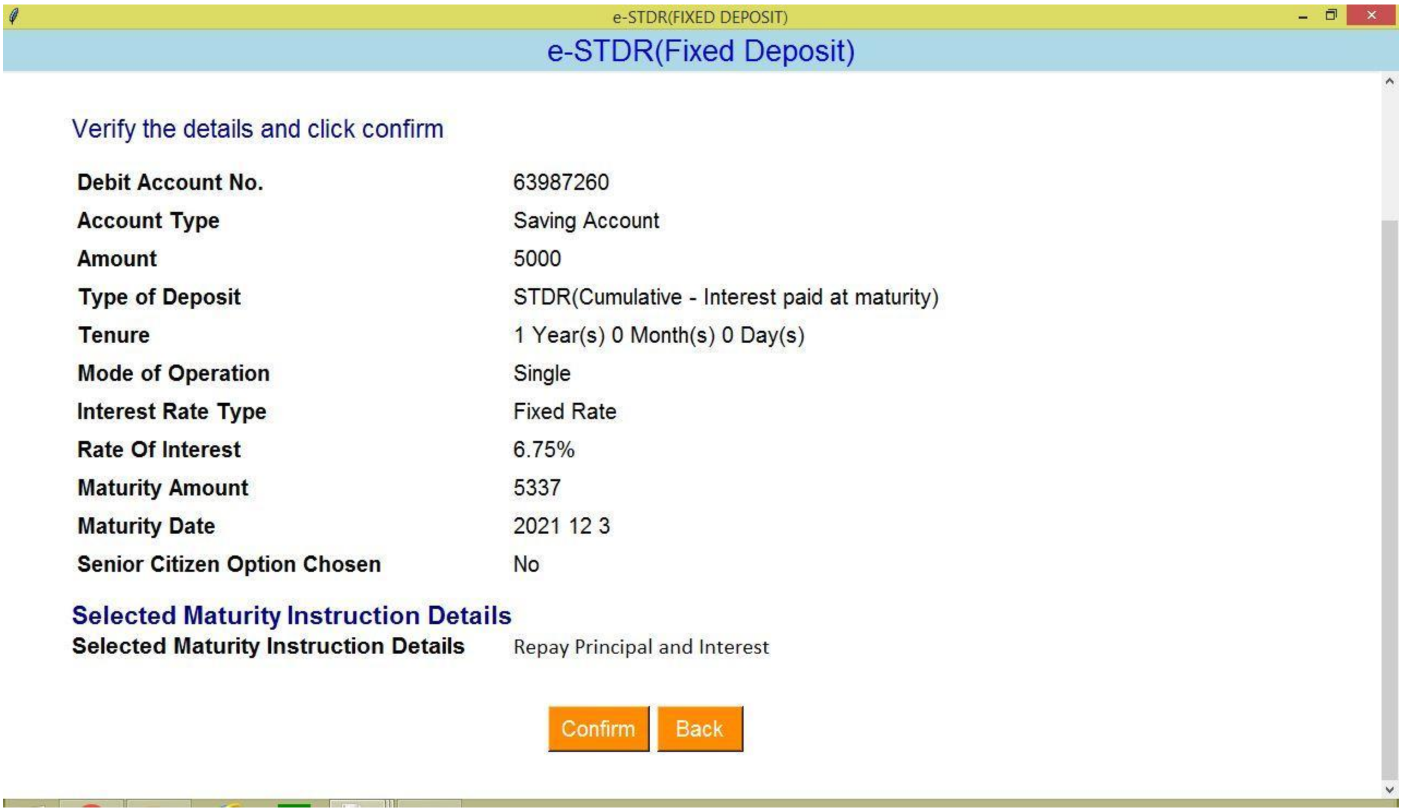
□ I accept the Terms and conditions View interest rate Domestic term deposit

- * As per section 206AA introduced by Finance (No.2) Act 2009 effective April 01,2010, every person who receives income on which TDS is deductible shall furnish his PAN , failing which TDS shall be deducted at the rate of 20% in case of Domestic deposits and 30.90% in case of NRO deposits.
- * Additionally, in the absence of PAN, Form 15G/H and other exemption certificates will be invalid even if submitted & penal TDS will be applicable.
- * NRO depositors are required to submit self declaration / tax residency certificates for availing concessional TDS facility under DTAA.
- * No Loan can be sanctioned against the term deposits opened under Tax Savings Scheme.



Revised in Interest Rates On Domestic Term Deposis w.e.f. 20.06.2020

Tenors	Existing Public w.e.f. 29.04.2020	Revised Public w.e.f. 20.06.2020	Existing for Senior Citizens w.e.f. 29.04.2020	Revised for Senior Citizens w.e.f. 20.06.2020
8 days to 45 days	5.50	5.50	6.00	6.00
46 days to 179 days	6.50	6.50	7.00	7.00
180 days to 210 days	6.50	6.50	7.00	7.00
211 days to less than 1 year	6.50	6.50	7.00	7.00
1 year	6.90	6.75	7.25	7.25
Above 1 year to 455 days	6.90	6.50	7.00	7.00
456 days to less than 2 years	6.75	6.50	7.00	7.00
2 years to less than 3 years	6.25	6.25	6.75	6.75
3 years to less than 5 years	6.25	6.25	6.75	6.75
5 years and up to 10 years	6.25	6.25	6.75	6.75



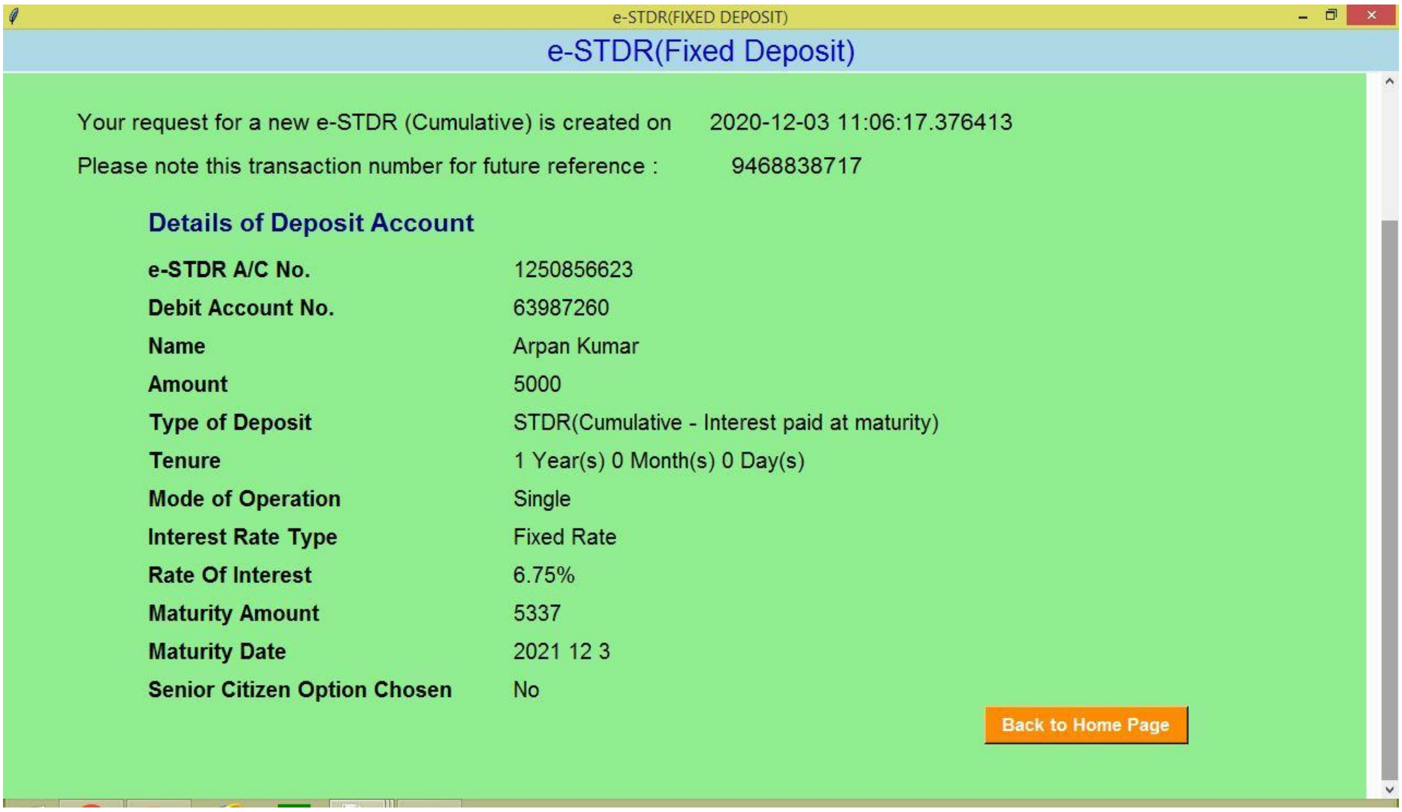
e-STDR(Fixed Deposit)

Verify the details and click confirm

Debit Account No.	63987260
Account Type	Saving Account
Amount	5000
Type of Deposit	STDR(Cumulative - Interest paid at maturity)
Tenure	1 Year(s) 0 Month(s) 0 Day(s)
Mode of Operation	Single
Interest Rate Type	Fixed Rate
Rate Of Interest	6.75%
Maturity Amount	5337
Maturity Date	2021 12 3
Senior Citizen Option Chosen	No

Selected Maturity Instruction Details

Selected Maturity Instruction Details	Repay Principal and Interest



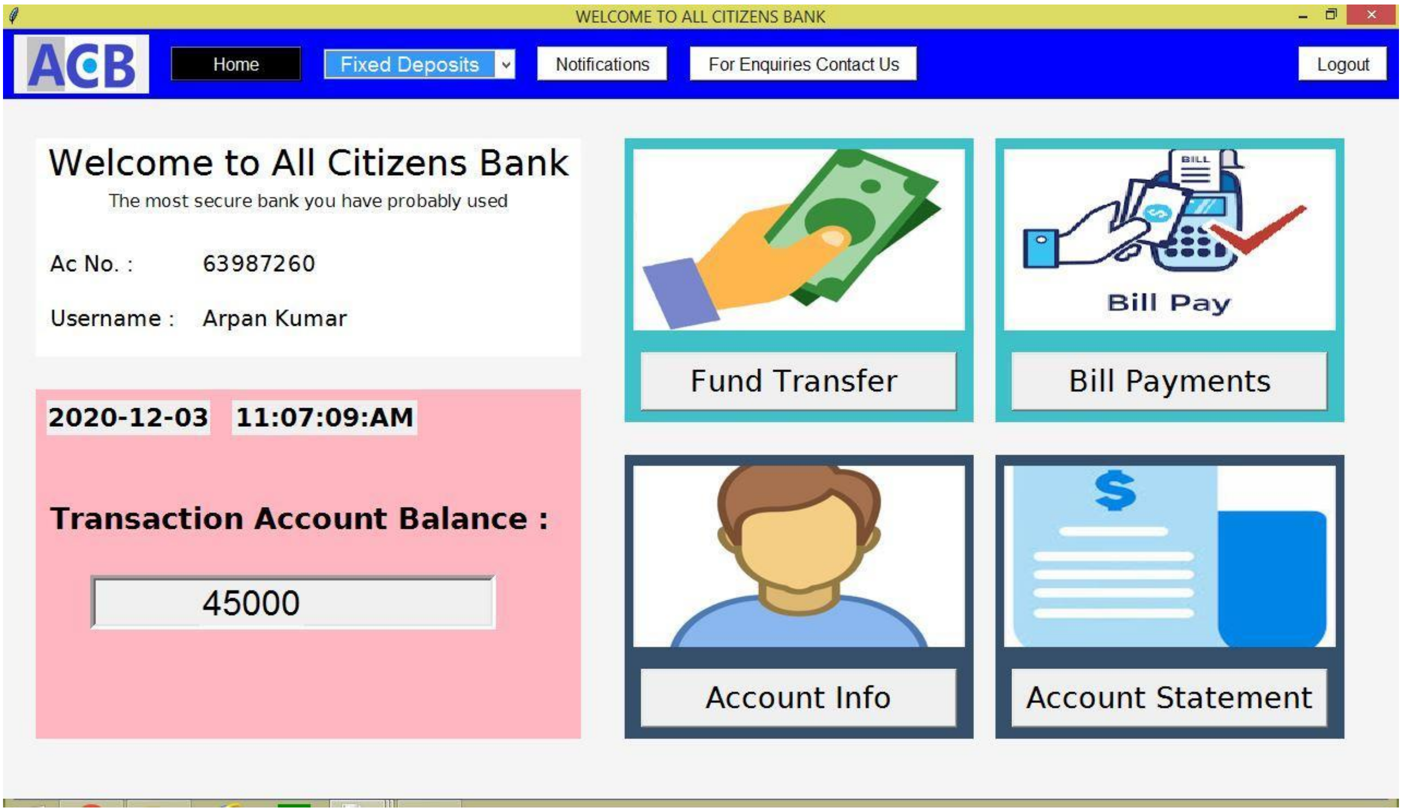
e-STDR(Fixed Deposit)

Your request for a new e-STDR (Cumulative) is created on 2020-12-03 11:06:17.376413

Please note this transaction number for future reference : 9468838717

Details of Deposit Account

e-STDR A/C No.	1250856623
Debit Account No.	63987260
Name	Arpan Kumar
Amount	5000
Type of Deposit	STDR(Cumulative - Interest paid at maturity)
Tenure	1 Year(s) 0 Month(s) 0 Day(s)
Mode of Operation	Single
Interest Rate Type	Fixed Rate
Rate Of Interest	6.75%
Maturity Amount	5337
Maturity Date	2021 12 3
Senior Citizen Option Chosen	No



Welcome to All Citizens Bank

The most secure bank you have probably used

Ac No. : 63987260

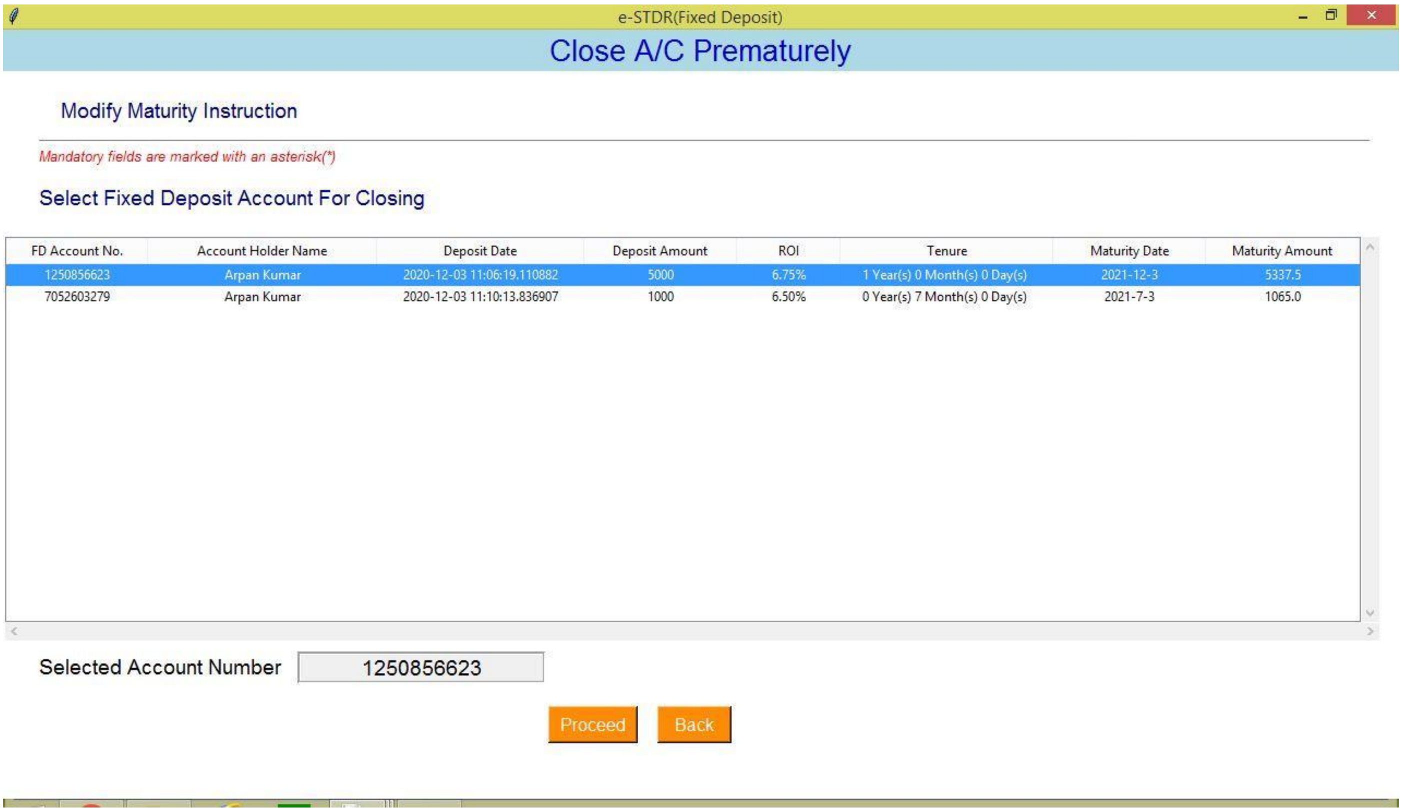
Username : Arpan Kumar

2020-12-03 11:07:09:AM

Transaction Account Balance :

45000

Close A/C Prematurely



Close A/C Prematurely

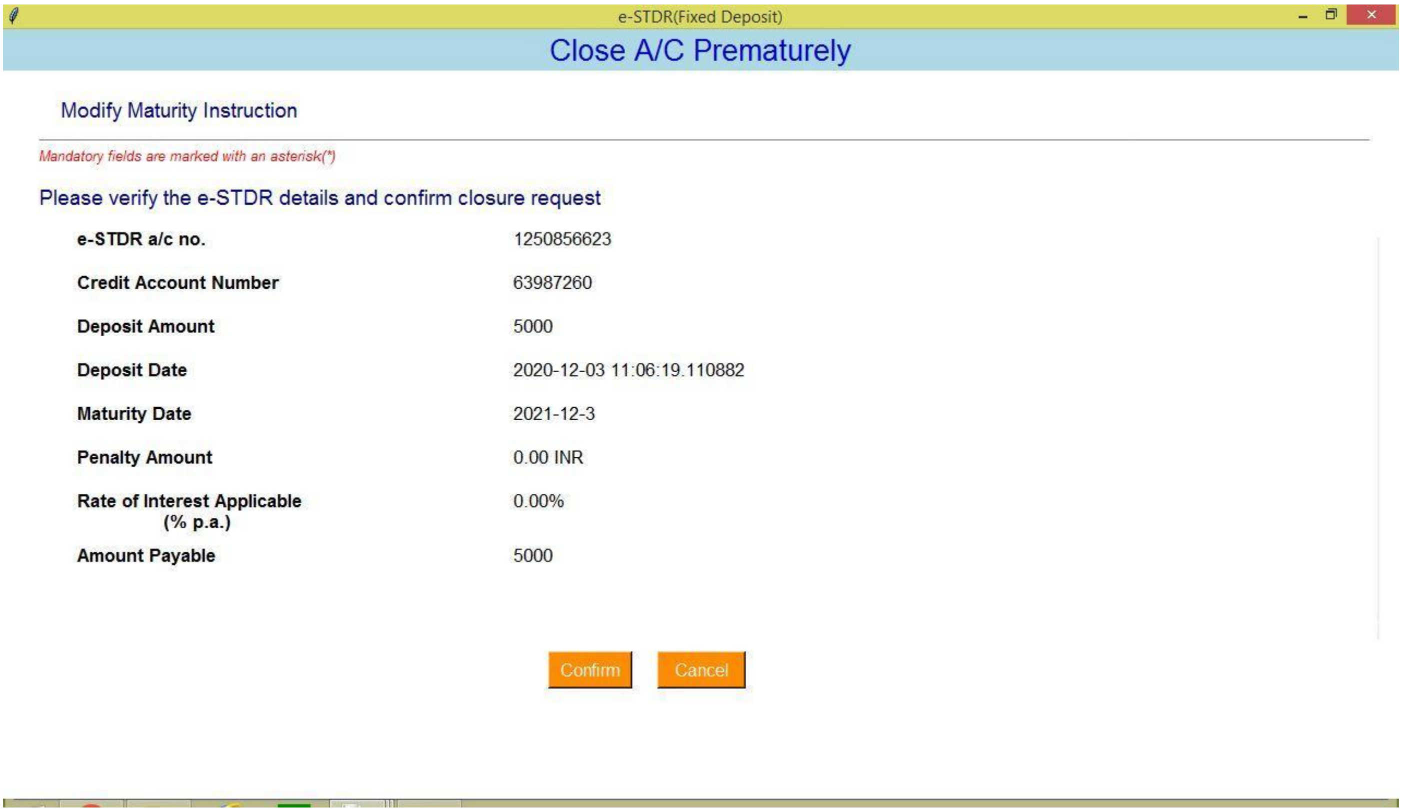
Modify Maturity Instruction

Mandatory fields are marked with an asterisk(*)

Select Fixed Deposit Account For Closing

FD Account No.	Account Holder Name	Deposit Date	Deposit Amount	ROI	Tenure	Maturity Date	Maturity Amount
1250856623	Arpan Kumar	2020-12-03 11:06:19.110882	5000	6.75%	1 Year(s) 0 Month(s) 0 Day(s)	2021-12-3	5337.5
7052603279	Arpan Kumar	2020-12-03 11:10:13.836907	1000	6.50%	0 Year(s) 7 Month(s) 0 Day(s)	2021-7-3	1065.0

Selected Account Number: 1250856623



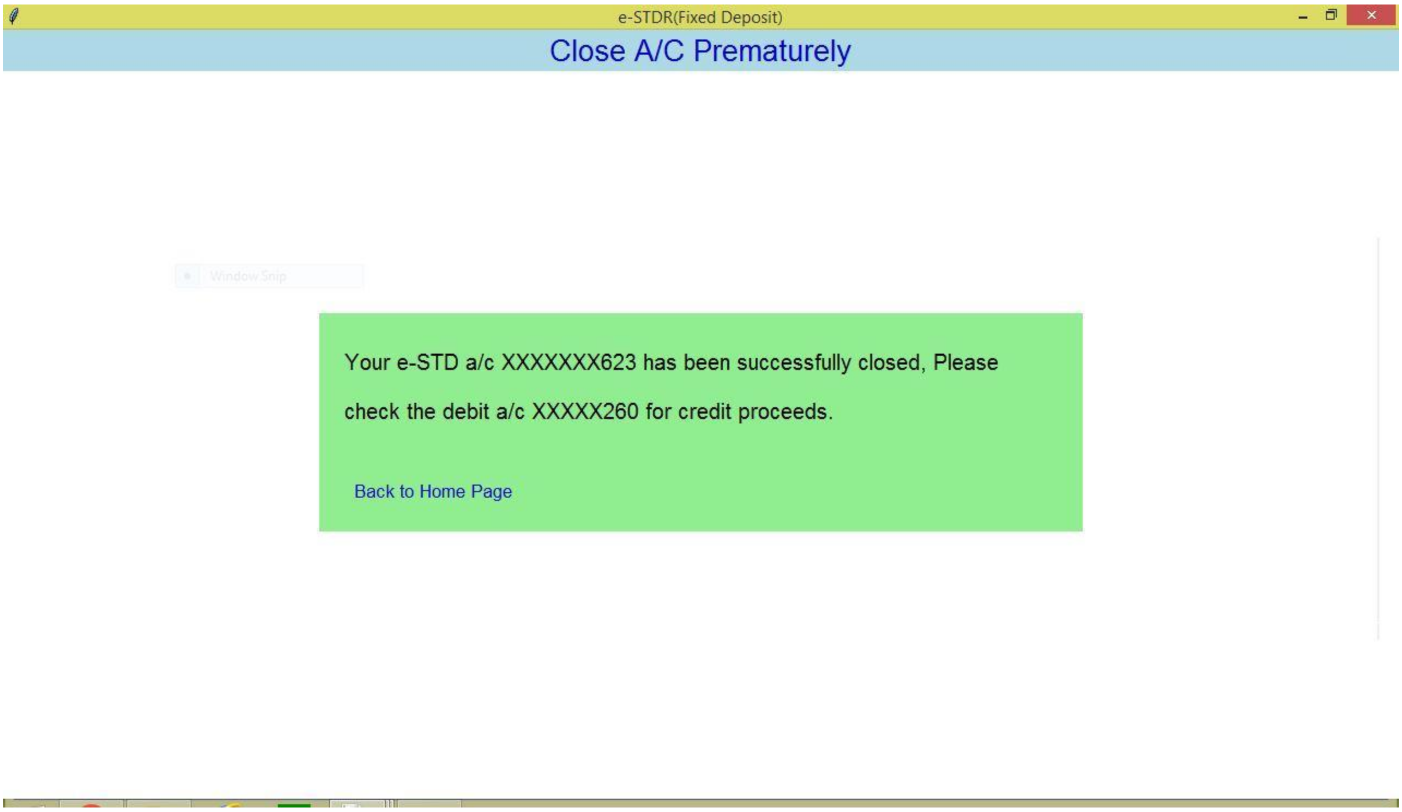
Close A/C Prematurely

Modify Maturity Instruction

Mandatory fields are marked with an asterisk(*)

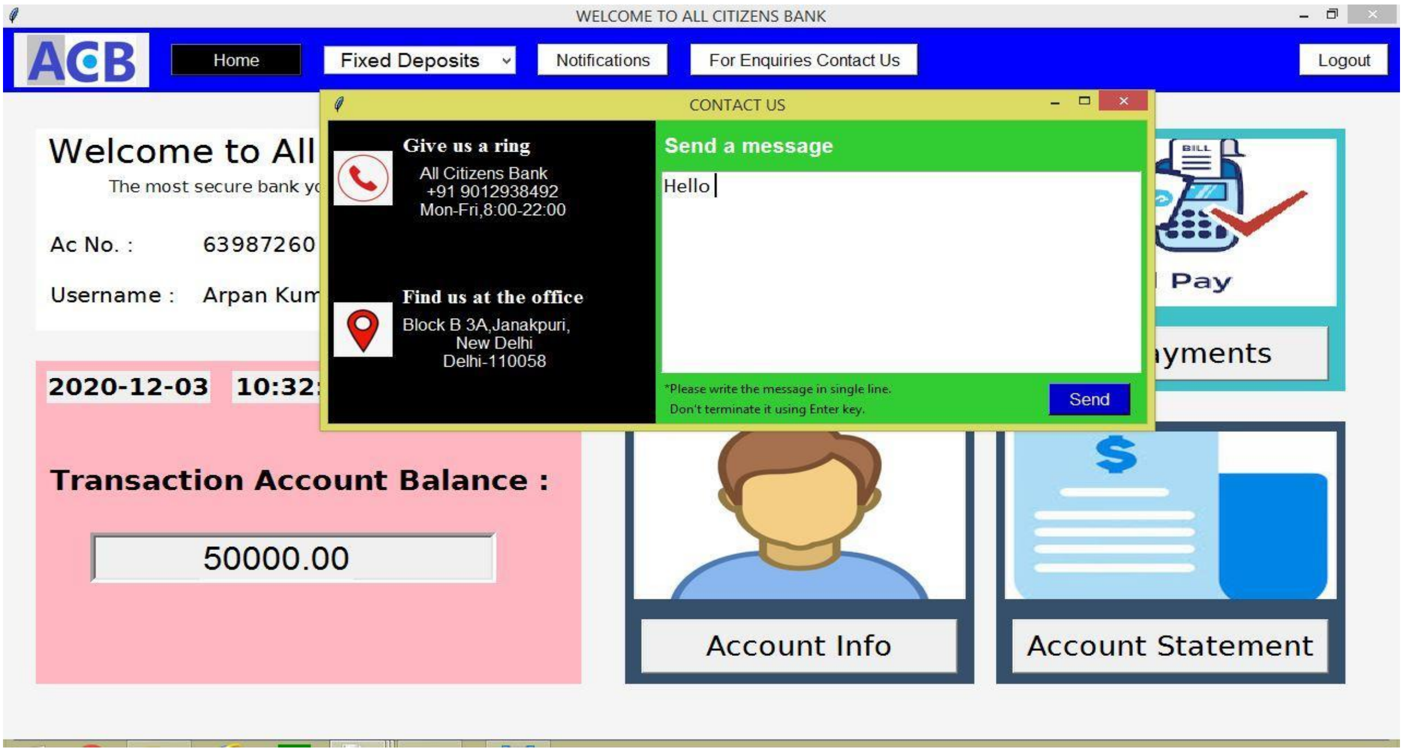
Please verify the e-STDR details and confirm closure request

e-STDR a/c no.	1250856623
Credit Account Number	63987260
Deposit Amount	5000
Deposit Date	2020-12-03 11:06:19.110882
Maturity Date	2021-12-3
Penalty Amount	0.00 INR
Rate of Interest Applicable (% p.a.)	0.00%
Amount Payable	5000



Your e-STD a/c XXXXXXX623 has been successfully closed, Please check the debit a/c XXXXX260 for credit proceeds.

Customer Care



Welcome to All
The most secure bank yo

Ac No. : 63987260

Username : Arpan Kum

2020-12-03 10:32:

Transaction Account Balance :

50000.00

Give us a ring
All Citizens Bank
+91 9012938492
Mon-Fri,8:00-22:00

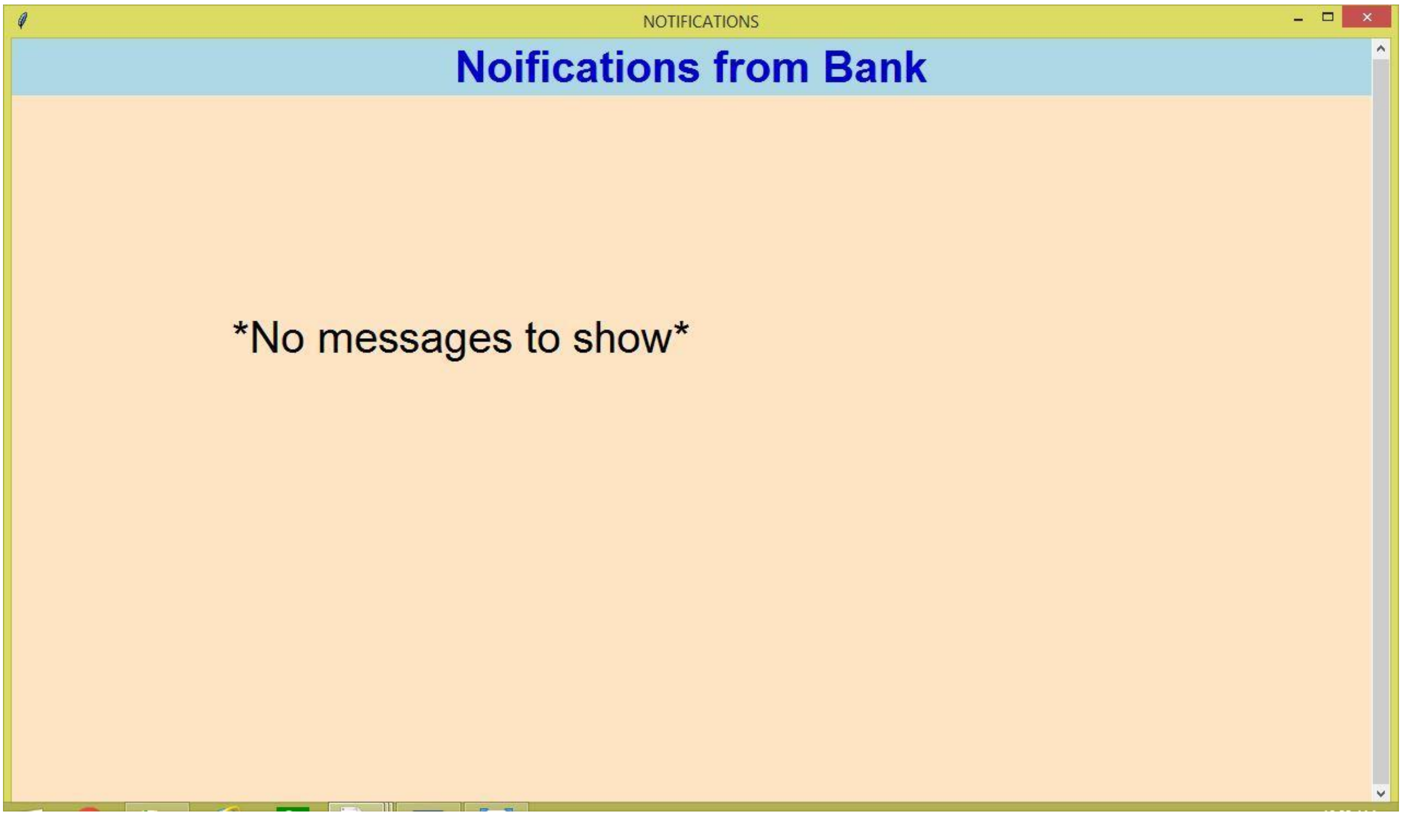
Find us at the office
Block B 3A,Janakpuri,
New Delhi
Delhi-110058

Send a message

Hello

*Please write the message in single line.
Don't terminate it using Enter key.

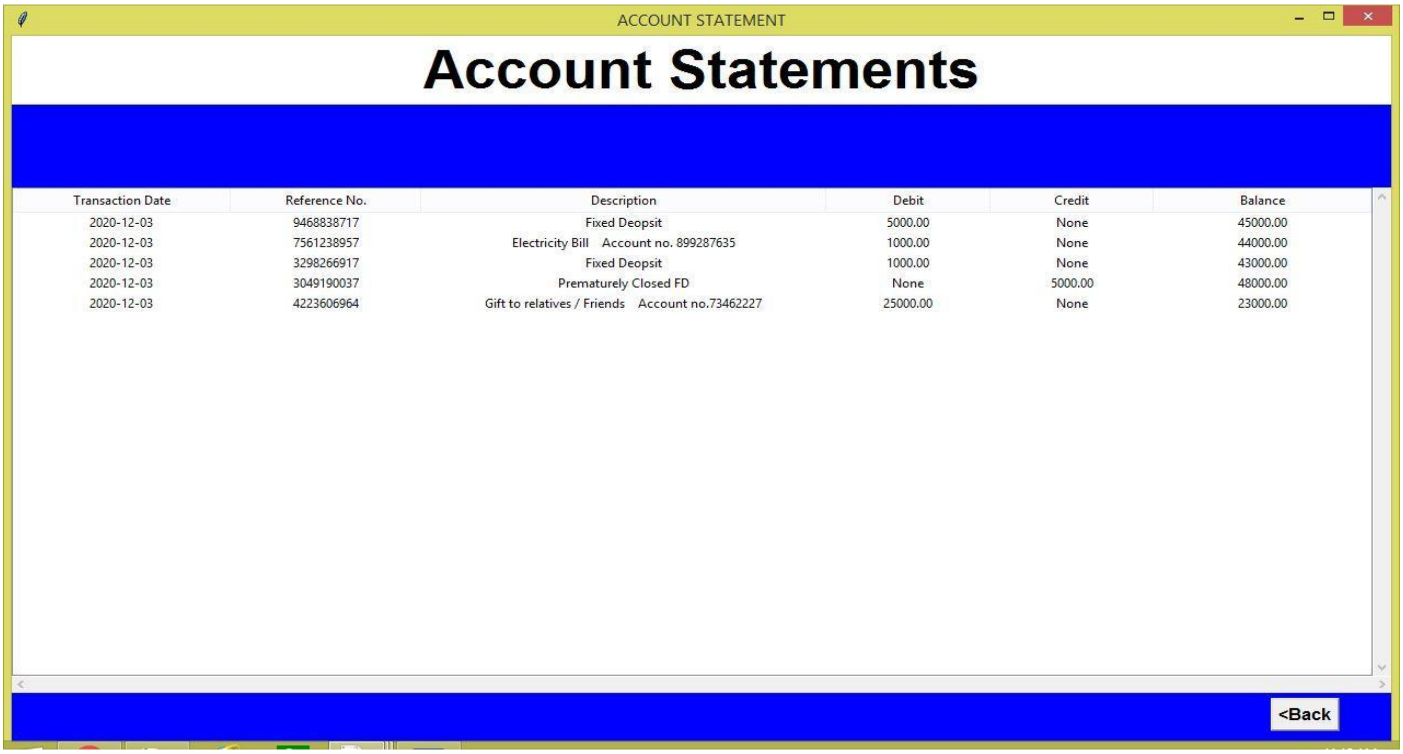
Bank Notifications



Noifications from Bank

No messages to show

Account Statements



Account Statements

Transaction Date	Reference No.	Description	Debit	Credit	Balance
2020-12-03	9468838717	Fixed Deopsit	5000.00	None	45000.00
2020-12-03	7561238957	Electricity Bill Account no. 899287635	1000.00	None	44000.00
2020-12-03	3298266917	Fixed Deopsit	1000.00	None	43000.00
2020-12-03	3049190037	Prematurely Closed FD	None	5000.00	48000.00
2020-12-03	4223606964	Gift to relatives / Friends Account no.73462227	25000.00	None	23000.00

LIMITATIONS

- Apply Loans
- Merchant services
- Insurance
- Debit & Credit cards
- Tax Payment

FUTURE PROSPECTS

This project can be handled in future by doing various modifications like:-

Three or four step security verification

- Facillity to change password
- Adding loan schemes
- Option to change account info
- Linking aadhar card to your account

BIBLIOGRAPHY

- Computer Science with Python Class XII, Preeti Arora
- https://www.youtube.com/playlist?list=PLu0W_9lII9ajLcqRcj4PoEihkukF_OTzA
- https://youtu.be/YTqDYmfccQU
- https://youtu.be/GCM-WpuRlCY
- https://youtu.be/e-mvAagwdGM
- https://youtu.be/lM0KTHgsEFg