BANK MANAGEMENT

CERTIFICATE

This is to certify that Mr. J. Dheeraj, Ms. Aura Bhattacharyya have successfully completed the project titled Bank Management in the year 2022-2023 for Programming for Problem Solving
It is further certified that this project is a group work of the candidates.

ACKNOWLEDGEMENT

We gratefully acknowledge our sincere thanks to our Programming for Problem Solving Professor, Dr. Anbarasi, for her remarkable, valuable guidance and supervision throughout the project work.

J. Dheeraj

Aura Bhattacharyya

INDEX

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements

- X86 64-bit CPU (Intel Pentium III/AMD architecture)
- 4 GB RAM (minimum)
- 5 GB free disk space
- Optical Mouse
- 14-inch color monitor
- Keyboard-108 keys
- Processor- Intel® Core™ i5 8250U
- CPU @ 1.60 GHz 1.80 GHz
- Memory- 8.00 GB
- Hard Disk- 28.0 KB

Software Requirements

- Operating System- Windows 7 to 10/XP or
- Linux
- Ubuntu 16.04 to 17.10.
- For MySQL Installer, Microsoft.NET Framework
- 4.5.2 or later required.

OBJECTIVES OF THE PROJECT

The main objective of the banking system which is developed in Python is to provide a flexible solution for the banking industry. With this system, both the users and working personnel of the particular bank will find easy to use and be able to perform all their operations using the system. This system enables its users to open and close a bank account. Customers will be able to withdraw and deposit amount to their particular account. Working personnel of particular bank will make all the operations which are required within the banking premises.

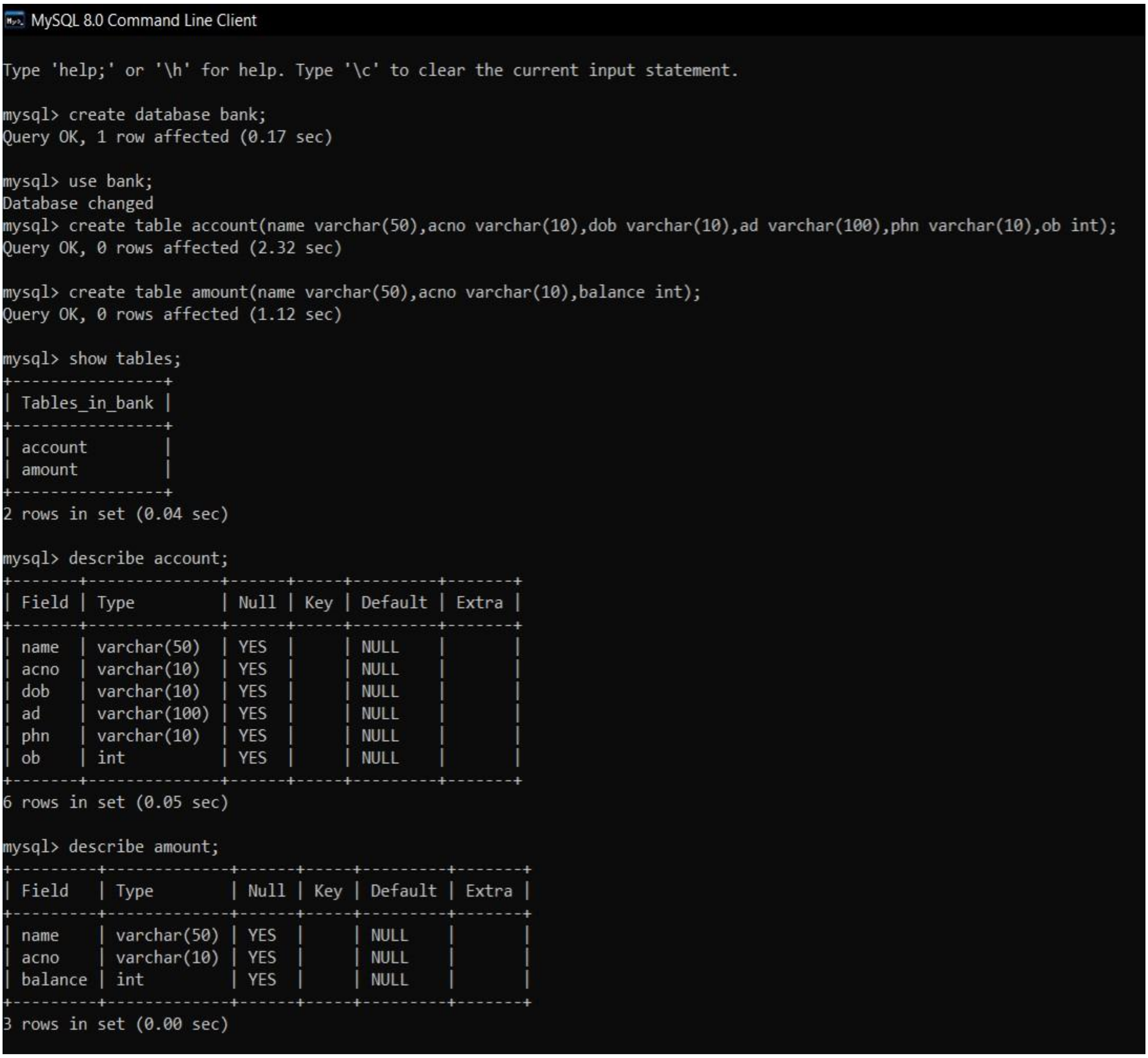
The functions which can be performed by the proposed system are:

1. Open New Account.
2. Deposit Amount.
3. Withdraw Amount.
4. Balance Enquiry.
5. Display Customer Details.
6. Close an Account.

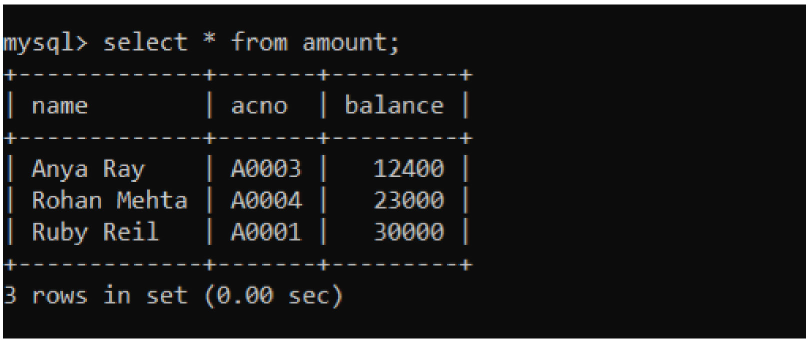
USER-DEFINED FUNCTIONS

- openAcc() : Opens a new account with the details of the customer such as name, DOB, phone number, address, and opening balance.
- depoAmo() : Deposits said amount in the specified account and updates the balance.
- witham() : Withdraws said amount from the specified account and updates the balance.
- display() : Prints/displays the details of the customer.
- balance() : Prints/displays the present balance in the account.
- closeac() : Deletes all details of the customer and the account, the account is closed.
- main() : Displays all of the other functions for the customer to choose from to manage his/her account in the bank.

CREATING THE DATABASE



```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database bank;
Query OK, 1 row affected (0.17 sec)

mysql> use bank;
Database changed
mysql> create table account(name varchar(50),acno varchar(10),dob varchar(10),ad varchar(100),phn varchar(10),ob int);
Query OK, 0 rows affected (2.32 sec)

mysql> create table amount(name varchar(50),acno varchar(10),balance int);
Query OK, 0 rows affected (1.12 sec)

mysql> show tables;
+----------------+
| Tables_in_bank |
+----------------+
| account        |
| amount         |
+----------------+
2 rows in set (0.04 sec)

mysql> describe account;
+-------+--------------+------+-----+---------+-------+
| Field | Type         | Null | Key | Default | Extra |
+-------+--------------+------+-----+---------+-------+
| name  | varchar(50)  | YES  |     | NULL    |       |
| acno  | varchar(10)  | YES  |     | NULL    |       |
| dob   | varchar(10)  | YES  |     | NULL    |       |
| ad    | varchar(100) | YES  |     | NULL    |       |
| phn   | varchar(10)  | YES  |     | NULL    |       |
| ob    | int          | YES  |     | NULL    |       |
+-------+--------------+------+-----+---------+-------+
6 rows in set (0.05 sec)

mysql> describe amount;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| name    | varchar(50) | YES  |     | NULL    |       |
| acno    | varchar(10) | YES  |     | NULL    |       |
| balance | int         | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```



```
mysql> select * from amount;
+-------------+-------+---------+
| name        | acno  | balance |
+-------------+-------+---------+
| Anya Ray    | A0003 |   12400 |
| Rohan Mehta | A0004 |   23000 |
| Ruby Reil   | A0001 |   30000 |
+-------------+-------+---------+
3 rows in set (0.00 sec)
```

SOURCE CODE

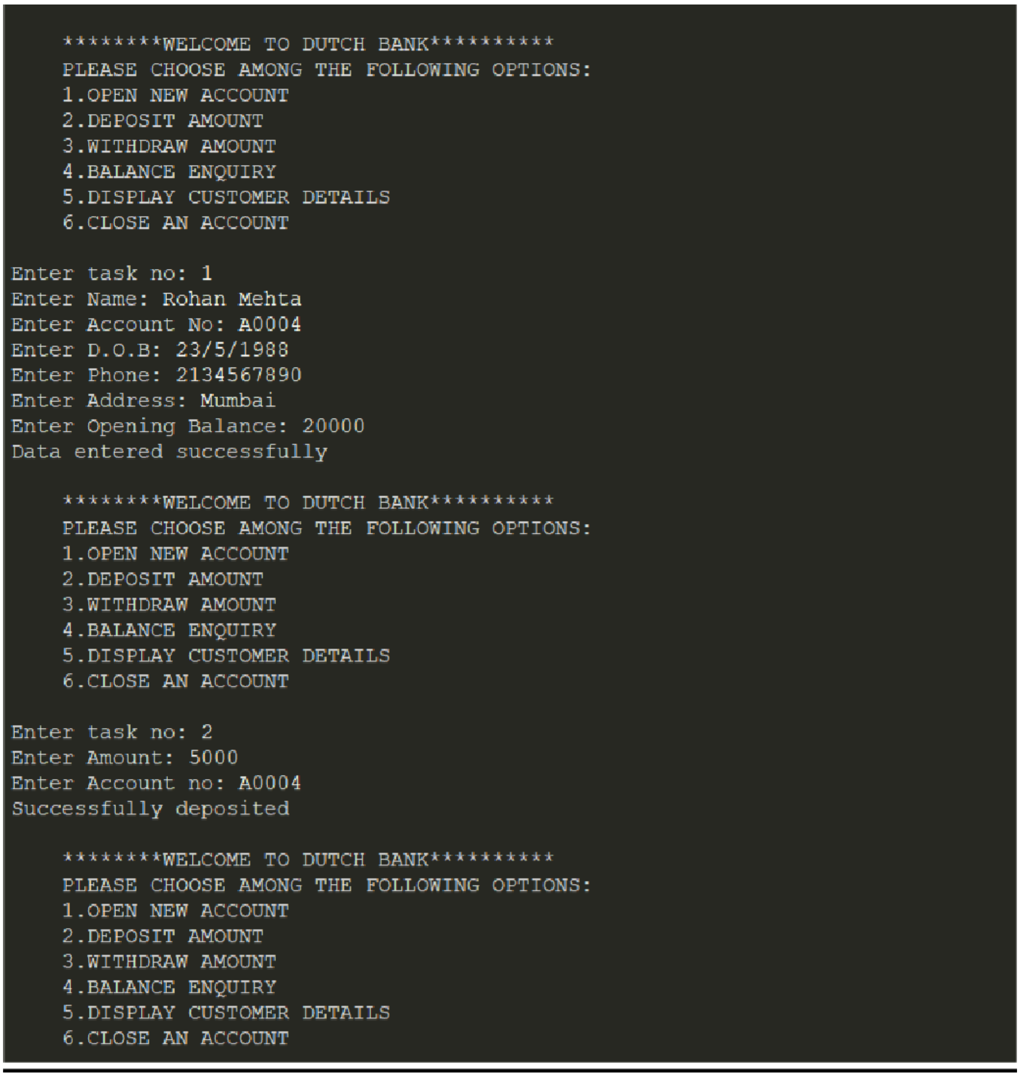
```
con=a.connect(host="localhost",user="root",passwd="Ro
wan@2004",database="bank")

def openAcc():
    n=input("Enter Name: ")
    ac=input("Enter Account No: ")
    db=input("Enter D.O.B:  ")
    p=input("Enter Phone: ")
    ad=input("Enter Address: ")
    ob=int(input("Enter Opening Balance: "))
    data1=(n,ac,db,p,ad,ob)
    data2=(n,ac,ob)
    sql1='insert into account values(%s,%s,%s,%s,%s,%s)'
    sql2='insert into amount values(%s,%s,%s)'
    c=con.cursor()
    c.execute(sql1,data1)
    c.execute(sql2,data2)
    con.commit()
    print("Data entered successfully")
    main()

def depoAmo():
   am=int(input("Enter Amount: "))
   ac=input("Enter Account no: ")
   a="select balance from amount where acno=%s"
   data=(ac,)
   c=con.cursor()
   c.execute(a,data)
```

```
  myresult=c.fetchone()
  tam=myresult[0]+am
  sql="update amount set balance=%s where acno=%s"
  d=(tam,ac)
  c.execute(sql,d)
  con.commit()
  print("Successfully deposited")
  main()
def  witham():
  am=int(input("Enter Amount: "))
  ac=input("Enter Account no: ")
  a="select balance from amount where acno=%s"
  data=(ac,)
  c=con.cursor()
  c.execute(a,data)
  myresult=c.fetchone()
  tam=myresult[0]-am
  sql="update amount set balance=%s where acno=%s"
  d=(tam,ac)
  c.execute(sql,d)
  con.commit()
  print("Amount successfully withdrawn")
  main()
def display():
   ac=input("Enter account no.:")
   a="select * from amount where acno = %s"
   data=(ac,)
   c=con.cursor()
```

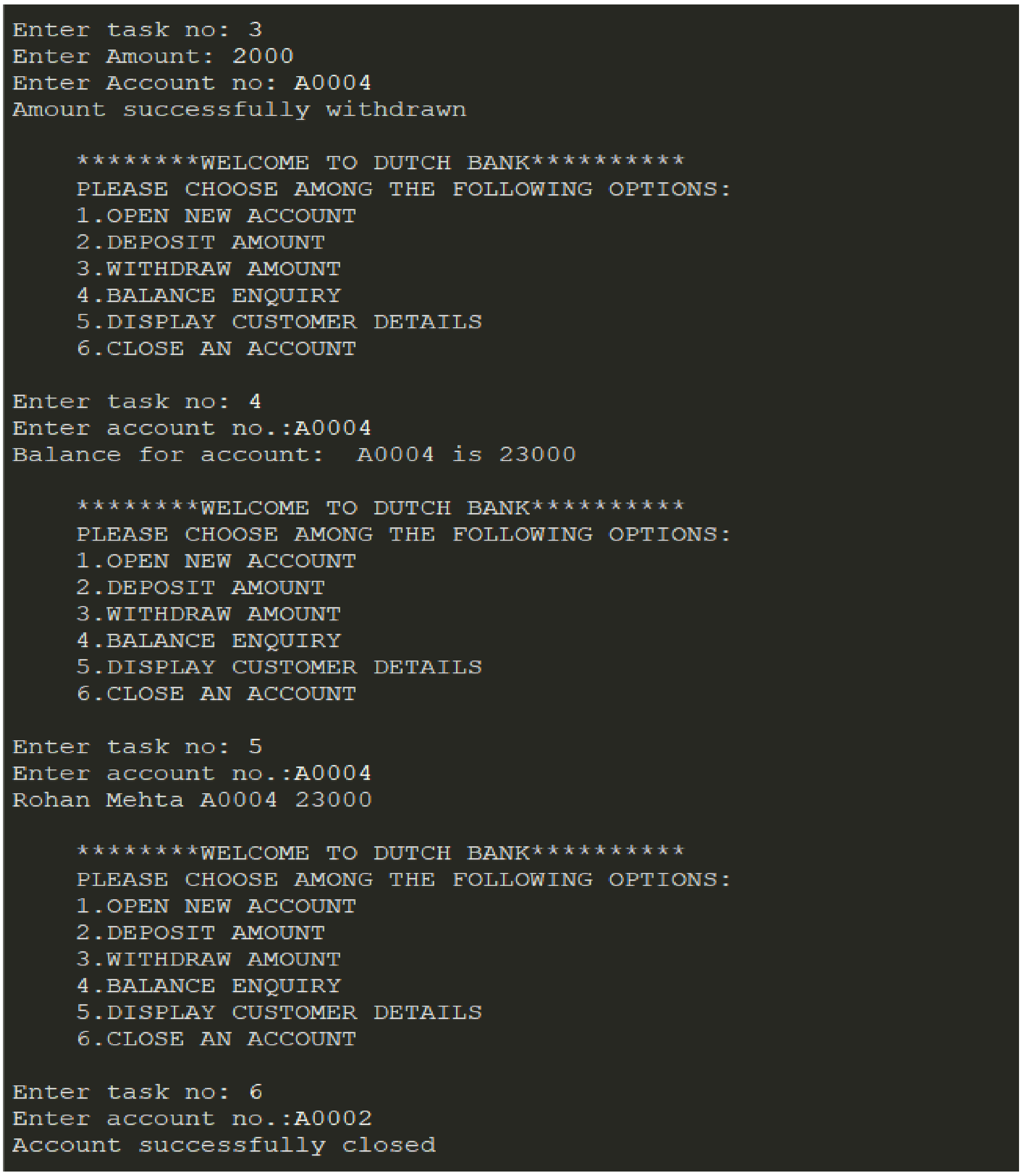
```
    c.execute(a,data)
    myresult=c.fetchone()
    for i in myresult:
        print(i,end=" ")
    print()
    main()

def balance():
    ac=input("Enter account no.:")
    a="select balance from amount where acno=%s"
    data=(ac,)
    c=con.cursor()
    c.execute(a,data)
    myresult=c.fetchone()
    print("Balance for account: ",ac,"is",myresult[0])
    main()

def closeac():
    ac=input("Enter account no.:")
    sql1="delete from account where acno=%s"
    sql2="delete from amount where acno=%s"
    data=(ac,)
    c=con.cursor()
    c.execute(sql1,data)
    c.execute(sql2,data)
    con.commit()
    print("Account successfully closed")
    main()
```

```
def main():
    print("""
    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT
    """)
    choice=input("Enter task no: ")
    if(choice=='1'):
        openAcc()
    elif(choice=='2'):
        depoAmo()
    elif(choice=='3'):
        witham()
    elif(choice=='4'):
        balance()
    elif(choice=='5'):
        display()
    elif(choice=='6'):
        closeac()
    else:
        print("Wrong choice..... ")
        main()
```

OUTPUT SCREEN



```
    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT

Enter task no: 1
Enter Name: Rohan Mehta
Enter Account No: A0004
Enter D.O.B: 23/5/1988
Enter Phone: 2134567890
Enter Address: Mumbai
Enter Opening Balance: 20000
Data entered successfully

    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT

Enter task no: 2
Enter Amount: 5000
Enter Account no: A0004
Successfully deposited

    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT
```



```
Enter task no: 3
Enter Amount: 2000
Enter Account no: A0004
Amount successfully withdrawn

    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT

Enter task no: 4
Enter account no.:A0004
Balance for account:  A0004 is 23000

    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT

Enter task no: 5
Enter account no.:A0004
Rohan Mehta A0004 23000

    ********WELCOME TO DUTCH BANK**********
    PLEASE CHOOSE AMONG THE FOLLOWING OPTIONS:
    1.OPEN NEW ACCOUNT
    2.DEPOSIT AMOUNT
    3.WITHDRAW AMOUNT
    4.BALANCE ENQUIRY
    5.DISPLAY CUSTOMER DETAILS
    6.CLOSE AN ACCOUNT

Enter task no: 6
Enter account no.:A0002
Account successfully closed
```

LIMITATIONS

- The system is directly accessible by anyone, and is therefore, not secure.
- It can be made more user-friendly.
- It only deals with the core functionality of a bank.

BIBLIOGRAPHY AND REFERENCES

1. www.geeksforgeeks.org
2. www.stackoverflow.com