**COMPUTER SCIENCE**



**TITLE :**– BANK DATABASE MANAGEMENT   
 SYSTEM

**STUDENT NAME**:-AKHILESH NEKAR

**ROLLNO** :-

**CLASS** :- XII



Name : Akhilesh Nekar

Roll No : 18602831

Class : XII

School : Geetanjali Olympiad School

This is certified to be the bonafide work of the student in the **Computer Laboratory** during the academic year 2021-22.

Teacher in charge Signature :

Examiner’s Signature :

Principal’s Signature :

School Seal:

Date :

**ACKNOWLEDGEMENT**

I would like to express my deepest appreciation to all those who provided me with the possibility to complete this project. I would like to thank my Computer Science teacher Mr.KOUSHIK K S,whose valuable guidance has helped me complete this project. His suggestions and instructions have served as a major contributor towards the completion of the project.

I take this opportunity to thank our head of the institution, Ms Durba Gupta,who was always supportive and helpful in fulfilling all our academic requirements.

Last but not the least; I would like to thank all my classmates who have helped me to complete this project.

**TABLE OF CONTENTS**

* Aim
* Introduction
* Requirements
* Imported files and used functions in python
* Table created in MySQL
* Coding
* Output
* Limitation and Enhancement
* Bibliography

**AIM**

Client can do all their banking operations comfortably without any risk of losing his privacy

* New user :-

🡪 Help him create new account and store the respective data in the database

* Returning user :-

🡪 Deposit/Withdraw amount and hence update savings accordingly

🡪 Transfer amount from one customer account to another customer within same bank only after checking the credentials properly.

🡪 Show all the details of a particular customer

🡪Allow only admins to update any info related to a particular account.

**INTRODUCTION**

* The bank management system is a systematic way of storing, using and retrieving the data and all the other information related to a person's bank account.
* I will try to show how a simple bank account management system works and develop a project which will make it easier to manage data related to a person’s bank account.
* Overall this software unit facilitates users banking activity. It helps at end user and bank management as well.
* Our main objective is the customer’s satisfaction considering today’s faster in the world.

**REQUIREMENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **HARDWARE** | | **SOFTWARE** | |
| PROCESSOR | Intel P3 and higher | Operating System | Windows 10 |
| RAM | 4GB and higher | Front End Language | PYTHON IDLE  VERSION 3.7.4 |
| SYSTEM ARCHITECTURE | 64-BIT OS | Back End Database | MySQL WORKBENCH VERSION 8.0 |
| Hard Disk | 256gb |  |  |
|  |  |  |  |

**FILES IMPORTED**

* Import mysql for database connectivity

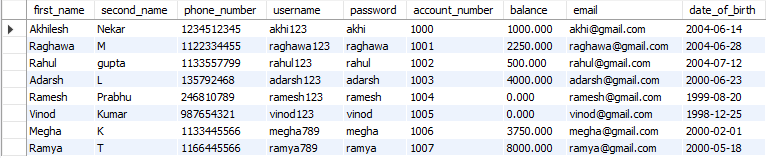
**FUNCTIONS USED IN PROJECT**

* connect() – for database and creation
* cursor() – to execute mysql queries
* fetchall() – to fetch all the rows
* fetchone() – to fetch one record according to query
* commit() – to execute current transaction

**TABLES CREATED IN MYSQL**

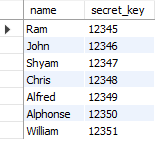
**TABLE 1:**

NEW\_CUSTOMER



**TABLE 2:**

ADMIN



**MODULES**

**LOGIN MODULE**

#Establishing connection between python and mysql

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="Akhilesh1327$l",database="bank\_data")

cursor=con.cursor()

#importing all the modules required

import deposit

import withdrawl

import transfer

import update\_info

import details

#Function to create serialised account numbers

def acc\_num():

sql="select count(\*) from new\_customer"

cursor.execute(sql)

data=cursor.fetchone()

global i

i=1000

i=data[0]+i

#Function to add new customer details into table

def new\_customer():

print("----WELCOME TO NEW CUSTOMER PAGE----")

First\_name=input("ENTER YOUR FIRST NAME:")

Second\_name=input("ENTER YOUR SECOND NAME:")

Phone\_number=int(input("ENTER YOUR PHONE NUMBER:"))

Username=input("ENTER YOUR USERNAME:")

Password=input("ENTER YOUR PASSWORD:")

Email=input("ENTER YOUR EMAIL:")

Balance=0

acc\_num()

Account\_number=i

Date\_of\_birth=input("ENTER DATE OF BIRTH IN YYYY-MM-DD FORMAT")

sql="insert into new\_customer(first\_name, second\_name, phone\_number, username, password, account\_number, balance, email, date\_of\_birth ) values ( '{}', '{}', {}, '{}', '{}', {}, {}, '{}', '{}' ) " . format (First\_name, Second\_name, Phone\_number, Username, Password, Account\_number, Balance, Email, Date\_of\_birth)

cursor.execute(sql)

con.commit()

print("SUCCESSFUL ENTRY FOR A NEW USER")

#Function to check the username and password of the customer and allowing them to login if they are correct

login=False

def existing\_customer():

print("----WELCOME----")

print("ENTER USERNAME AND PASSWORD TO PROCEED")

Username=input("ENTER USERNAME:")

Password=input("ENTER PASSWORD:")

sql="select username,password from new\_customer"

cursor.execute(sql)

data=cursor.fetchall()

for row in data:

if row[0]==Username and row[1]==Password:

print("SUCCESSFULLY LOGGED IN")

global login

login=True

break

else:

continue

else:

print("INVALID USERNAME OR PASSWORD")

#Function to allow login of admins by checking that their login credentials are correct

def adminlogin():

print("----WELCOME----")

print("ENTER YOUR NAME AND SECRET KEY TO PROCEED")

Name=input("ENTER YOUR NAME")

Secret\_key=int(input("ENTER SECRET KEY"))

sql="select \* from admin"

cursor.execute(sql)

data=cursor.fetchall()

for row in data:

if row[0].lower()==Name.lower() and row[1]==Secret\_key:

print("SUCCESSFUL ADMIN LOGIN")

print("YOU CAN NOW CHNAGE THE DETAILS OF THE RESPECTIVE   
 CUSTOMER")

update\_info.update\_info()

break

else:

continue

else:

print("INVALID ADMIN ENTRY")

#Function to check whether its a new customer,an existing customer or an admin

#This is the main function as all of the funtions from different modules are called under this function login

def login():

print("----WELCOME TO LOGIN PAGE----")

print("1. NEW CUSTOMER")

print("2. EXISTING USER")

print("3. ADMIN LOGIN")

choice=int(input("ENTER YOUR CHOICE"))

if choice==1:

print()

new\_customer()

existing\_customer()

elif choice==2:

print()

existing\_customer()

elif choice==3:

print()

adminlogin()

#Giving user choices of the activites and performing the respective activity as per their choice

if login==True:

print("CHOOSE THE ACTIVITY FROM THE FOLLOWING")

print("1. Deposit")

print("2. Withdrawl")

print("3. Transfer")

print("4. Details")

print("5. Exit")

ch=int(input("ENTER YOUR CHOICE"))

if ch==1:

deposit.deposit()

elif ch==2:

withdrawl.withdraw()

elif ch==3:

transfer.transfer()

elif ch==4:

details.details()

elif ch==5:

print("THANK YOU VISIT AGAIN")

login()

**DEPOSIT MODULE**

#connection

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="PassWord123",database="bank\_data")

cursor=con.cursor()

#Function which performs the activity of removing amount withdrawn from savings

#account of the customer after making sure that their credentials are correct

def deposit():

print("----WELCOME TO DEPOSIT PAGE----")

Account\_number=int(input("ENTER YOUR ACCOUNT NUMBER"))

Phone\_number=int(input("ENTER YOUR PHONE NUMBER"))

sql\_select\_query="select phone\_number,account\_number,balance from new\_customer"

cursor.execute(sql\_select\_query)

data=cursor.fetchall()

for row in data:

if row[0]==Phone\_number and row[1]==Account\_number:

print("PHONE NUMBER AND ACCOUNT NUMBER MATCHED")

amount=float(input("ENTER AMOUNT TO BE DEPOSITED"))

final\_amount=float(row[2])+amount

sql\_update\_query="update new\_customer set balance={} where account\_number={}".format(final\_amount,Account\_number)

cursor.execute(sql\_update\_query)

con.commit()

print("AMOUNT SUCCESSFULLY CREDITED")

break

else:

continue

else:

print("INVALID ACCOUNT NUMBER OR MOBILE NUMBER")

**WITHDRAW MODULE**

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="Akhilesh1327$l",database="bank\_data")

cursor=con.cursor()

#Function which performs the activity of adding amount to the savings

#account of the customer after making sure that their credentials are correct

def withdraw():

print("---WELCOME TO WITHDRAWL PAGE---")

Account\_number=int(input("ENTER YOUR ACCOUNT NUMBER"))

Phone\_number=int(input("ENTER YOUR PHONE NUMBER"))

amount=float(input("ENTER AMOUNT TO BE WITHDRAWED"))

sql\_select\_query="select phone\_number,account\_number,balance from new\_customer"

cursor.execute(sql\_select\_query)

data=cursor.fetchall()

for row in data:

if row[0]==Phone\_number and row[1]==Account\_number and float(row[2])>amount:

print("PHONE NUMBER AND ACCOUNT NUMBER MATCHED")

final\_amount=float(row[2])-amount

sql\_update\_query="update new\_customer set balance={} where account\_number={}".format(final\_amount,Account\_number)

cursor.execute(sql\_update\_query)

con.commit()

print("AMOUNT SUCCESSFULLY WITHDRAWED")

break

else:

continue

else:

print("INVALID ACCOUNT NUMBER OR MOBILE NUMBER OR INSUFFICIENT AMOUNT")

**TRANSFER MODULE**

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="Akhilesh1327$l",database="bank\_data")

cursor=con.cursor()

#Function which performs the activity of removing specific amount from one account and adding

#the same into the account of another customer after making sure that both the accounts

#exist and the credentials of the customer making the transfer match successfully

def transfer():

print("---WELCOME TO TRANSFER PAGE---")

Account\_number\_1=int(input("ENTER ACCOUNT NUMBER OF YOUR ACCOUNT:"))

Phone\_number=int(input("ENTER YOUR PHONE NUMBER:"))

Account\_number\_2=int(input("ENTER ACCOUNT NUMBER OF THE OTHER ACCOUNT:"))

sql\_select\_query="select phone\_number,account\_number,balance from new\_customer"

cursor.execute(sql\_select\_query)

data=cursor.fetchall()

amount=float(input("ENTER AMOUNT TO BE TRANSFERRED"))

for row in data:

existing=False

if row[1]==Account\_number\_2:

print("OTHER ACCOUNT NUMBER EXISTS")

existing=True

break

else:

continue

else:

print("THE OTHER ACCOUNT NUMBER DOESNT EXIST")

for row in data:

deduct=False

if row[0]==Phone\_number and row[1]==Account\_number\_1 and float(row[2])>=amount and existing==True:

print("YOUR ACCOUNT NUMBER AND PHONE NUMBER MATCHED")

final\_amount\_1=float(row[2])-amount

sql\_update\_query\_1="update new\_customer set balance={} where account\_number={}".format(final\_amount\_1,Account\_number\_1)

cursor.execute(sql\_update\_query\_1)

con.commit()

deduct=True

break

else:

continue

else:

print("YOUR ACCOUNT NUMBER OR PHONE NUMBER IS INVALID")

for row in data:

added=False

if row[1]==Account\_number\_2 and deduct==True:

final\_amount\_2=float(row[2])+amount

sql\_update\_query\_2="update new\_customer set balance={} where account\_number={}".format(final\_amount\_2,Account\_number\_2)

cursor.execute(sql\_update\_query\_2)

con.commit()

added=True

break

else:

continue

if added==True:

print("Rs/-",amount,"","SUCCESSFULLY TRANSFERRED FROM YOUR ACCOUNT TO ACCOUNT NUMBER",Account\_number\_2)

else:

print("TRANSFER UNSUCCESSFUL")

**UPDATE\_INFO MODULE**

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="Akhilesh1327$l",database="bank\_data")

cursor=con.cursor()

#Function which gives the admin options of which part of customer's details is to be updated

def update\_info():

print("---WELCOME TO UPDATE INFO PAGE---")

print("CHOOSE WHICH DATA TO BE MODIFIED")

print("1. First Name")

print("2. Second Name")

print("3. Phone Number")

print("4. Username")

print("5. Password")

ch=int(input("\*\*\*ENTER YOUR CHOICE\*\*\*"))

print("FOR VERIFICATION ENTER THE DETAILS BELOW")

Account\_number=int(input("ENTER CUSTOMER ACCOUNT NUMBER"))

Phone\_number=int(input("ENTER CUSTOMER PHONE NUMBER"))

sql\_select\_query="select account\_number,phone\_number from new\_customer"

cursor.execute(sql\_select\_query)

data=cursor.fetchall()

for row in data:

validity=False

if row[0]==Account\_number and row[1]==Phone\_number:

validity=True

break

else:

continue

else:

print("INVALID ACCOUNT NUMBER OR PHONE NUMBER")

if ch==1 and validity==True:

First\_name=input("ENTER NEW FIRST NAME")

sql\_update\_query\_1="update new\_customer set first\_name='{}' where account\_number={}".format(First\_name,Account\_number)

cursor.execute(sql\_update\_query\_1)

con.commit()

print("FIRST NAME SUCCESSFULLY CHANGED")

elif ch==2 and validity==True:

Second\_name=input("ENTER NEW SECOND NAME")

sql\_update\_query\_2="update new\_customer set second\_name='{}' where account\_number={}".format(Second\_name,Account\_number)

cursor.execute(sql\_update\_query\_2)

con.commit()

print("SECOND NAME SUCCESSFULLY CHANGED")

elif ch==3 and validity==True:

Phone\_number=int(input("ENTER NEW PHONE NUMBER"))

sql\_update\_query\_3="update new\_customer set phone\_number={} where account\_number={}".format(Phone\_number,Account\_number)

cursor.execute(sql\_update\_query\_3)

con.commit()

print("PHONE NUMBER SUCCESSFULLY CHANGED")

elif ch==4 and validity==True:

Username=input("ENTER NEW USERNAME")

sql\_update\_query\_4="update new\_customer set username='{}' where account\_number={}".format(Username,Account\_number)

cursor.execute(sql\_update\_query\_4)

con.commit()

print("USERNAME SUCCESSFULLY CHANGED")

elif ch==5 and validity==True:

Password=input("ENTER NEW PASSWORD")

sql\_update\_query\_5="update new\_customer set password='{}' where account\_number={}".format(Password,Account\_number)

cursor.execute(sql\_update\_query\_5)

con.commit()

print("PASSWORD SUCCESSFULLY CHANGED")

**DETAILS MODULE**

import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",passwd="Akhilesh1327$l",database="bank\_data")

cursor=con.cursor()

#Function which diplays the details of a particular customer

def details():

print("---WELCOME TO DETAILS PAGE---")

print("FOR VERIFICATION ENTER THE DETAILS BELOW")

Account\_number=int(input("ENTER YOUR ACCOUNT NUMBER"))

Phone\_number=int(input("ENTER YOUR PHONE NUMBER"))

sql\_select\_query\_1="select \* from new\_customer"

cursor.execute(sql\_select\_query\_1)

data=cursor.fetchall()

for row in data:

validity=False

if row[5]==Account\_number and row[2]==Phone\_number:

validity=True

break

else:

continue

else:

print("INVALID ACCOUNT NUMBER OR PHONE NUMBER")

if validity==True:

sql\_select\_query\_2="select \* from new\_customer where account\_number={}".format(Account\_number)

cursor.execute(sql\_select\_query\_2)

data=cursor.fetchone()

for i in range(8):

print("FIRST NAME:",data[0])

print("SECOND NAME:",data[1])

print("PHONE NUMBER:",data[2])

print("USERNAME:",data[3])

print("EMAIL:",data[7])

print("BALANCE:",data[6])

break

elif validity==False:

print("INVALID ACCOUNT NUMBER OR PHONE NUMBER")

if validity==True:

sql\_select\_query\_3="select date\_of\_birth from new\_customer where account\_number={}".format(Account\_number)

cursor.execute(sql\_select\_query\_3)

data=cursor.fetchone()

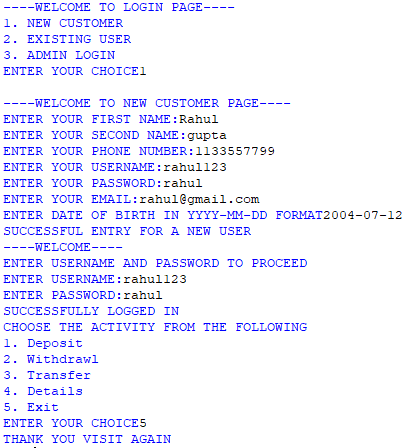
for i in data:

print("DATE OF BIRTH IS:",end='')

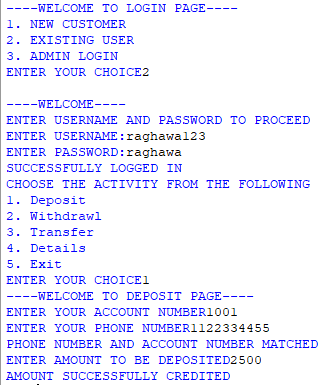
print(i,end=' ')

**OUTPUT**

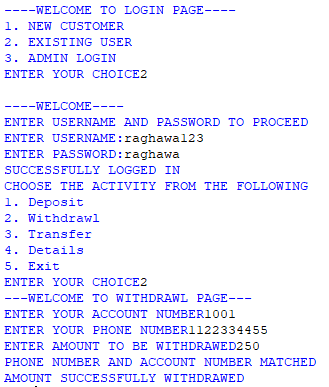
*NEW USER OUTPUT* **:**



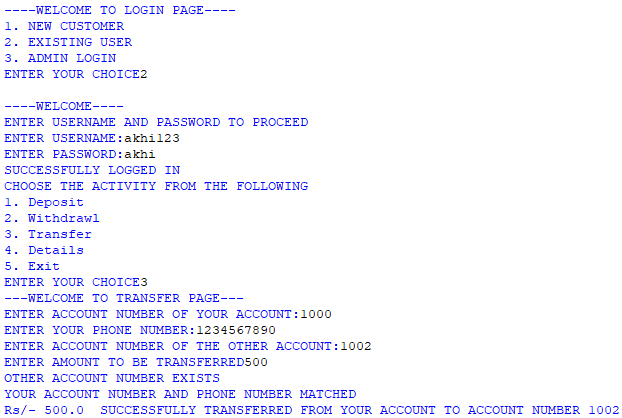
*DEPOSIT OUTPUT* :



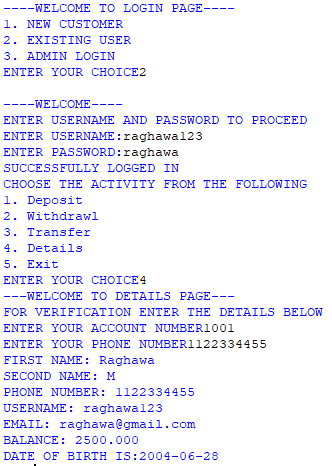
*WITHDRAW OUTPUT:*

**

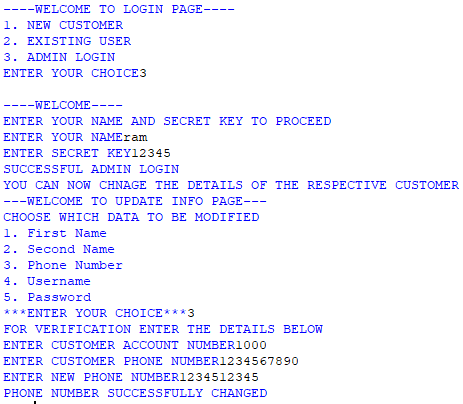
*TRANSFER OUTPUT:*



*DETAILS OUTPUT:*

**

*UPDATING INFO OUTPUT:*

**

**LIMITATIONS AND ENHANCEMENT**

LIMITATIONS *:-*

🡪All the functions a bank does could not be implemented here

🡪Security check for admins is not strong

ENHANCEMENT *:-*

🡪Briefly explained the working of a bank database management system

🡪Steps are taken only after checking that the credentials of a customer are verified with the database

🡪The authority to change the details of a particular customer are given only to the admins

**BIBLIOGRAPHY**

* <https://www.google.com/>
* <https://en.wikipedia.org/wiki/Main_Page>
* Computer Science with python class XII Sumita Arora