

BANK MANAGEMENT

{USING MYSQL}



February 22, 2023

[VELAMMAL ENGINEERING COLLAGE]

[SURAPET]

|  |  |
| --- | --- |
| TEAM MEMBER | ROLL NO |
| RAGESHWARAN HR | 21 |
| BHARATH VAAISHNAV TB | 04 |
| PRANESH.S | 18 |

|  |
| --- |
| CSE A |

ABSTRACT:

This bank management system is built using the C programming language and MySQL database. The system provides a menu-driven interface for customers to perform various banking operations such as creating new accounts, checking balance, transferring funds, depositing and withdrawing money, checking account details, deleting customer records, applying for loans, checking loan status, and paying back loans.

The system starts with a welcome message and displays a menu with different options for the customers to choose from. The system uses functions to handle each option selected by the customer, such as creating a new account, checking balance, transferring funds, depositing and withdrawing money, and so on.

The system also allows customers to apply for loans and check their loan status. Additionally, the system has the functionality to delete customer records in case a customer wants to close their account.

Overall, this bank management system provides an efficient and user-friendly way for customers to perform banking operations, making it easier for the bank to manage its operations and customer records.

SOURCE CODE:

#include <mysql.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define DB\_HOST "localhost"

#define DB\_USER "root"

#define DB\_PASS "ragesh2004"

#define DB\_NAME "bankmanagements"

void auth(MYSQL\*);

void menu(MYSQL\*);

void new\_acc(MYSQL\*);

void check\_balance(MYSQL\*);

void transfer(MYSQL\*);

void deposit(MYSQL\*);

void withdraw(MYSQL\*);

void check\_details(MYSQL\*);

void delete\_customer(MYSQL\*);

void apply\_loan(MYSQL\*);

void check\_loan\_status(MYSQL\*);

void pay\_back\_loan(MYSQL\*);

void menu(MYSQL \*con)

{

int a;

printf("\n\n\t\t==========Welcome to RB bank of india ==========\n\n");

printf("\n\t\t----Please choose one of the options below----\n");

printf("\n\t\t<1> create new Accounts \n");

printf("\n\t\t<2> Check Balance\n");

printf("\n\t\t<3> bank transfer\n");

printf("\n\t\t<4> Deposit\n");

printf("\n\t\t<5> Withdraw\n");

printf("\n\t\t<6> check details \n");

printf("\n\t\t<7> delete customer\n");

printf("\n\t\t<8> apply loan\n");

printf("\n\t\t<9> check loan status\n");

printf("\n\t\t<10> pay\_back\_loan\n");

printf("\n\t\t<11> Exit\n\n");

printf("\n\n\t\t===============================================\n\n");

printf("enter the choice: ");

scanf("%d", &a);

switch (a)

{

case 1:

new\_acc(con);

break;

case 2:

check\_balance(con);

break;

case 3:

transfer(con);

break;

case 4:

deposit(con);

break;

case 5:

withdraw(con);

break;

case 6:

check\_details(con);

break;

case 7:

delete\_customer(con);

break;

case 8:

apply\_loan(con);

break;

case 9:

check\_loan\_status(con);

break;

case 10:

pay\_back\_loan(con);

case 11:

exit(0);

default:

printf("Invalid choice.\n");

break;

}

}

void auth(MYSQL \*con)

{

char name[50];

char emp\_pin[10];

char query[200];

printf("Enter your name: ");

scanf("%s", name);

printf("Enter your employee pin: ");

scanf("%s", emp\_pin);

sprintf(query, "SELECT \*FROM employee WHERE name='%s' AND

emp\_pin='%s'", name, emp\_pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

int num\_fields = mysql\_num\_fields(result);

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

printf("Authentication successful.\n");

}

else

{

printf("Authentication failed.\n");

mysql\_close(con);

exit(1);

}

mysql\_free\_result(result);

}

void new\_acc(MYSQL \*con)

{

char name[50];

char email[50];

char phone\_number[20];

char address[1000];

char adhar\_number[20];

char query[500];

char pin[6];

char amount[20];

int num\_fields;

char accountno[11];

printf("Enter customer name: ");

scanf("%s", name);

printf("Enter customer email: ");

scanf("%s", email);

printf("Enter customer phone number: ");

scanf("%s", phone\_number);

printf("Enter customer address: ");

scanf(" %[^\n]", address);

printf("Enter customer adhar number: ");

scanf("%s", adhar\_number);

printf("Enter initial amount: ");

scanf("%s", &amount);

printf("Enter account number: ");

scanf("%s", accountno);

printf("Enter your pin: ");

scanf("%s", &pin);

sprintf(query, "INSERT INTO customers (name, email, phone\_number, address, adharnumber, amount, pin, accountnumber) VALUES ('%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s')", name, email, phone\_number, address, adhar\_number, amount, pin, accountno);

int status = mysql\_query(con, query);

if (status != 0)

{

printf("Failed to create account.\n");

}

else

{

if (mysql\_affected\_rows(con) > 0)

{

printf("Account created successfully.\n");

}

else

{

printf("Failed to create account.\n");

}

}

menu(con);

}

void check\_balance(MYSQL \*con)

{

char from\_acc[12], pin[4];

char query[200];

printf("Enter the account number to check balance: ");

scanf("%s", from\_acc);

printf("Enter your pin: ");

scanf("%s", pin);

sprintf(query, "SELECT \*FROM customers WHERE accountnumber='%s' AND pin='%s' ", from\_acc, pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

int num\_fields = mysql\_num\_fields(result);

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

printf("Your balance is: %s\n", row[6]);

}

else

{

printf("No customer found with adhar number %s.\n", from\_acc);

}

mysql\_free\_result(result);

menu(con);

}

void withdraw(MYSQL \*con)

{

char from\_acc[12], pin[4];

char query[200], new\_balances[20];

int amount;

printf("Enter the account number to withdraw: ");

scanf("%s", from\_acc);

printf("Enter your pin: ");

scanf("%s", pin);

sprintf(query, "SELECT \*FROM customers WHERE accountnumber='%s' AND pin='%s'", from\_acc, pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

int num\_fields = mysql\_num\_fields(result);

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

printf("Enter amount to withdraw: ");

scanf("%d", &amount);

int balance = atoi(row[6]);

int new\_balance = balance - amount;

if (new\_balance < 0)

{

printf("Withdrawal amount exceeds balance. Please try again.\n");

}

else

{

sprintf(new\_balances, "%d", new\_balance);

sprintf(query, "UPDATE customers SET amount=%s WHERE

accountnumber='%s'", new\_balances, from\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

else

{

printf("Withdrawal successful. New balance is: %d\n", new\_balance);

}

}

}

else

{

printf("No customer found with accountnumber %s.\n", from\_acc);

}

mysql\_free\_result(result);

menu(con);

}

void deposit(MYSQL \*con)

{

char from\_acc[12], pin[4];

char amount\_str[20];

int amount;

char query[200];

char new\_balances[20];

printf("Enter the account number to deposit : ");

scanf("%s", from\_acc);

printf("Enter your pin: ");

scanf("%s", pin);

sprintf(query, "SELECT \*FROM customers WHERE accountnumber='%s' AND

pin='%s'", from\_acc, pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

int num\_fields = mysql\_num\_fields(result);

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

printf("Enter amount to deposit: ");

scanf("%s", amount\_str);

amount = atoi(amount\_str);

int current\_balance = atoi(row[6]);

int new\_balance = current\_balance + amount;

sprintf(new\_balances, "%d", new\_balance);

sprintf(query, "UPDATE customers SET amount=%s WHERE accountnumber='%s'", new\_balances, from\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

printf("deposit successful. New balance is: %d\n", new\_balance);

}

else

{

printf("No customer found with accountnumber %s.\n", from\_acc);

}

mysql\_free\_result(result);

menu(con);

}

void transfer(MYSQL \*con)

{

char from\_acc[20], pin[4];

char to\_acc[20];

int amount;

char query[500];

char from\_acc\_amount[20];

char to\_acc\_amount[20];

int flag = 0;

printf("Enter the account number to transfer from: ");

scanf("%s", from\_acc);

printf("Enter your pin: ");

scanf("%s", pin);

printf("Enter the account number to transfer to: ");

scanf("%s", to\_acc);

printf("Enter the amount to transfer: ");

scanf("%d", &amount);

sprintf(query, "SELECT \*FROM customers WHERE accountnumber='%s' AND pin='%s'", from\_acc, pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

int from\_amount = atoi(row[0]);

if (from\_amount >= amount)

{

from\_amount -= amount;

sprintf(from\_acc\_amount, "%d", from\_amount);

sprintf(query, "UPDATE customers SET amount='%s' WHERE accountnumber='%s'", from\_acc\_amount, from\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

flag = 1;

}

else

{

printf("Insufficient balance.\n");

}

}

else

{

printf("No customer found with account number %s.\n", from\_acc);

}

if (flag == 1)

{

sprintf(query, "SELECT amount FROM customers WHERE accountnumber='%s'",

to\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

if ((row = mysql\_fetch\_row(result)))

{

int to\_amount = atoi(row[0]);

to\_amount += amount;

sprintf(to\_acc\_amount, "%d", to\_amount);

sprintf(query, "UPDATE customers SET amount='%s' WHERE accountnumber='%s'", to\_acc\_amount, to\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

printf("Amount transferred successfully.\n");

}

else

{

printf("No customer found with account number %s.\n", to\_acc);

}

}

mysql\_free\_result(result);

menu(con);

}

void check\_details(MYSQL \*con)

{

char from\_acc[12], pin[4];

char query[200];

printf("Enter the account number to check balance: ");

scanf("%s", from\_acc);

sprintf(query, "SELECT \*FROM customers WHERE accountnumber='%s'", from\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

int num\_fields = mysql\_num\_fields(result);

MYSQL\_ROW row;

if ((row = mysql\_fetch\_row(result)))

{

printf("Name: %s\n", row[1]);

printf("Email: %s\n", row[2]);

printf("Phone number: %s\n", row[3]);

printf("Address: %s\n", row[4]);

printf("Adhar number: %s\n", row[5]);

printf("Account balance: %s\n", row[6]);

printf("Account number: %s\n", row[8]);

}

else

{

printf("No customer found with account number %s.\n", from\_acc);

}

mysql\_free\_result(result);

menu(con);

}

void delete\_customer(MYSQL \*con)

{

char account\_number[11];

char query[500];

printf("Enter the account number of the customer you want to delete: ");

scanf("%s", account\_number);

sprintf(query, "DELETE FROM customers WHERE accountnumber = '%s'", account\_number);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

int num\_rows\_affected = mysql\_affected\_rows(con);

if (num\_rows\_affected > 0)

{

printf("Customer deleted successfully.\n");

}

else

{

printf("Customer not found.\n");

}

menu(con);

}

void apply\_loan(MYSQL \*con)

{

char accountno[11];

char loan\_amount[20];

char interest\_rate[10];

char duration[10];

char query[500];

char Query[500];

int loan\_id;

int balance;

printf("Enter account number: ");

scanf("%s", accountno);

printf("Enter loan amount: ");

scanf("%s", loan\_amount);

printf("Enter interest rate (in percentage): ");

scanf("%s", interest\_rate);

printf("Enter loan duration (in months): ");

scanf("%s", duration);

sprintf(query, "INSERT INTO loans (accountnumber, amount, interest\_rate, duration) VALUES ('%s', '%s', '%s', '%s')", accountno, loan\_amount, interest\_rate, duration);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

loan\_id = mysql\_insert\_id(con);

printf("Loan application submitted successfully. Your loan id is: %d\n", loan\_id);

menu(con);

}

void check\_loan\_status(MYSQL \*con)

{

char loanid[20];

int loan\_id;

char query[200];

char is\_paid[15];

printf("Enter loan ID: ");

scanf("%s", loanid);

loan\_id = atoi(loanid);

sprintf(query, "SELECT \*FROM loans WHERE loan\_id=%d", loan\_id);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(0);

}

int num\_rows = mysql\_num\_rows(result);

if (num\_rows == 0)

{

printf("Loan not found.\n");

}

else

{

MYSQL\_ROW row = mysql\_fetch\_row(result);

printf("Loan ID: %s\n", row[0]);

printf("accountn number: %srs\n", row[1]);

printf("amount: %s%\n", row[2]);

printf("Interest rate: %s%%\n", row[3]);

printf("duration: %s\n", row[4]);

strcpy(is\_paid, row[5]);

if (strcmp(is\_paid, "paid") == 0)

{

printf("Loan status: paid\n");

}

else

{

printf("Loan status: unpaid\n");

}

}

mysql\_free\_result(result);

menu(con);

}

void pay\_back\_loan(MYSQL \*con)

{

char loan\_id\_str[20];

int loan\_id, old, amounts, months, rate;

char query[200], confirm;

char from\_acc[12], pin[4];

char new\_balance\_str[20];

char Query[200];

printf("Enter loan ID: ");

scanf("%s", loan\_id\_str);

printf("Enter the months taken to pay: ");

scanf("%d", &months);

printf("Enter the interest rate in percentage: ");

scanf("%d", &rate);

loan\_id = atoi(loan\_id\_str);

sprintf(query, "SELECT \*FROM loans WHERE loan\_id=%d", loan\_id);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

MYSQL\_RES \*result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

int num\_rows = mysql\_num\_rows(result);

if (num\_rows == 0)

{

printf("Loan not found.\n");

}

else

{

MYSQL\_ROW row = mysql\_fetch\_row(result);

int is\_paid = atoi(row[5]);

if (is\_paid == 1)

{

printf("This loan has already been paid.\n");

}

else

{

printf("Loan ID: %s\n", row[0]);

printf("Amount given: $%s\n", row[2]);

old = atoi(row[2]);

amounts = old + (old \*months \*rate) / 100;

printf("Amount to be paid: $%d\n", amounts);

printf("Are you sure you want to pay back this loan? (y/n): ");

scanf(" %c", &confirm);

if (confirm == 'y' || confirm == 'Y')

{

printf("Enter the account number to withdraw: ");

scanf("%s", from\_acc);

printf("Enter your PIN: ");

scanf("%s", pin);

sprintf(query, "SELECT \*FROM customers WHERE

accountnumber='%s' AND pin='%s'", from\_acc, pin);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

result = mysql\_store\_result(con);

if (result == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

num\_rows = mysql\_num\_rows(result);

if (num\_rows == 0)

{

printf("Invalid account number or PIN.\n");

}

else

{

MYSQL\_ROW row = mysql\_fetch\_row(result);

int balance = atoi(row[6]);

int new\_balance = balance - amounts;

if (new\_balance < 0)

{

printf("Withdrawal amount exceeds the balance.

Please try again.\n");

}

else

{

sprintf(new\_balance\_str, "%d", new\_balance);

sprintf(query, "UPDATE customers SET amount='%s'

WHERE accountnumber='%s'", new\_balance\_str,

from\_acc);

if (mysql\_query(con, query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

sprintf(Query, "UPDATE loans SET status='paid' WHERE

loan\_id=%d", loan\_id);

if (mysql\_query(con, Query))

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

printf("Loan successfully paid.\n");

}

}

}

}

mysql\_free\_result(result);

menu(con);

}

}

int main()

{

MYSQL \*con = mysql\_init(NULL);

if (con == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

exit(1);

}

if (mysql\_real\_connect(con, DB\_HOST, DB\_USER, DB\_PASS, DB\_NAME, 0, NULL, 0) == NULL)

{

fprintf(stderr, "%s\n", mysql\_error(con));

mysql\_close(con);

exit(1);

}

auth(con);

menu(con);

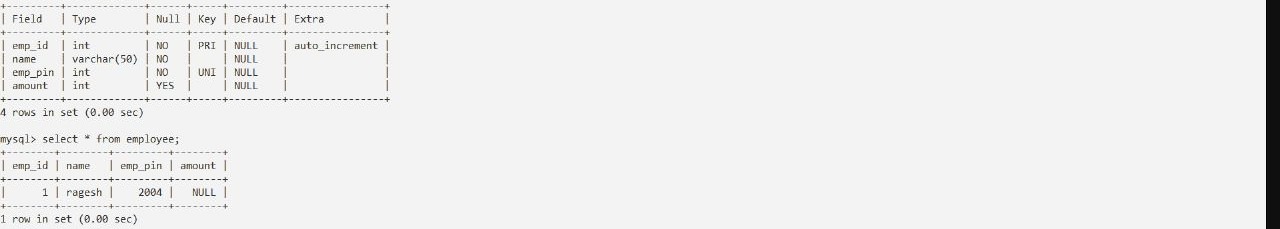
mysql\_close(con);

return 0;

}

TABLE:

Customer database in MYSQL:



Employee database in MYSQL:

Table

Description automatically generated

Table

Description automatically generated

Table

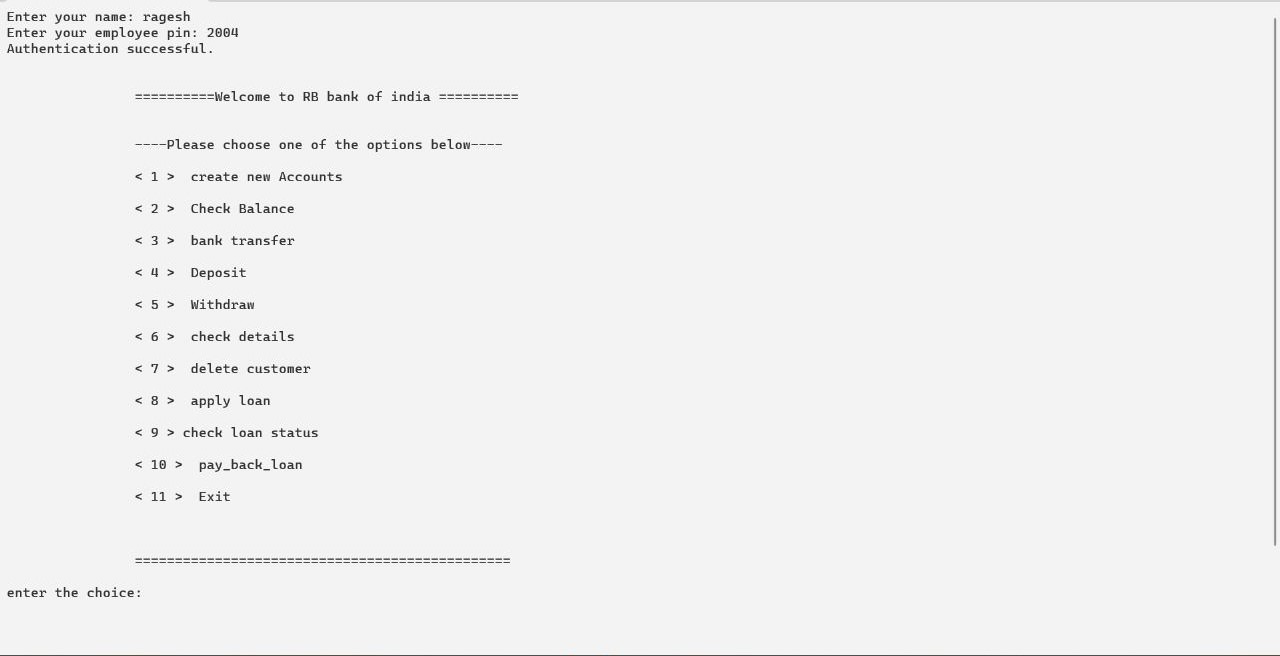
Description automatically generated

LOAN TABEL:

Table

Description automatically generated

OUTPUT:

MENU:

CREATE NEW ACCOUNT: -

Text, letter

Description automatically generated

CHECK BALANCE: -

Graphical user interface, text, application, email

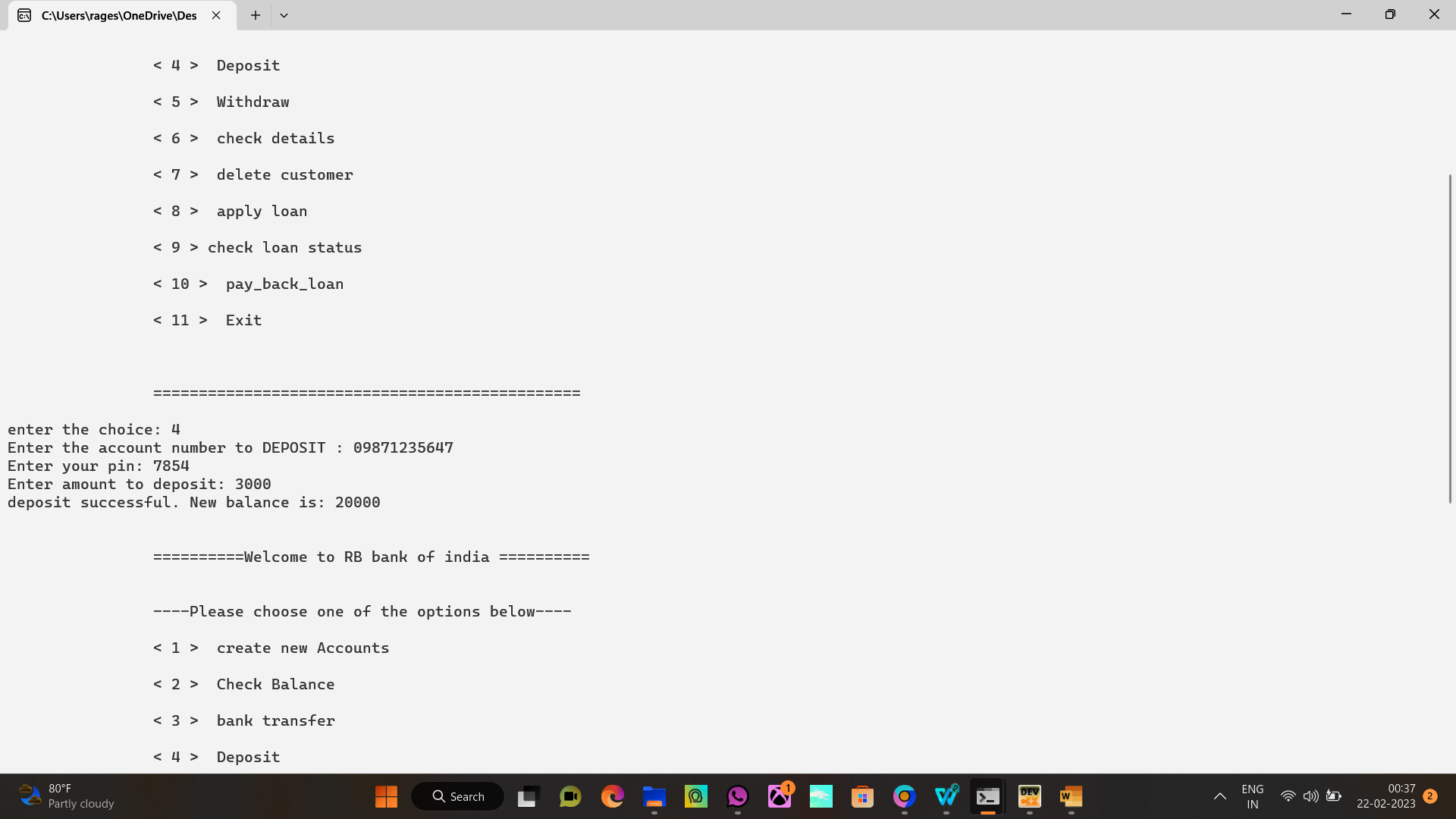
Description automatically generated

BANK TRANSFER:-

Graphical user interface, text, application

Description automatically generated

DEPOSIT:-



WITHDRAW: -

Graphical user interface, text, application, email

Description automatically generated

CHECK BALANCE: -

Graphical user interface, text, letter

Description automatically generated

REMOVE CUSTOMER: -

Text, letter

Description automatically generated

LOAN

APPLY LOAN: -



STATUS OF LOAN: -



PAY BACK LOAN: -

Text, letter

Description automatically generated

CHECKING STATUS AFTER PAYING LOAN: -



EXIT: -

