

**JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY
SECTOR 128, NOIDA,
UTTAR PRADESH**

“BANK MANAGEMENT SYSTEM”

Mini Project of SDF-II Lab

Submitted By :

**Rahi Agarwal (9921103145)
Aditi Bhargava (9921103148)
Aryan Mishra (9921103150)
Shivansh Pandey (9921103152)**

Under the Supervision of :

Dr. Arti Jain

Mr. Rupesh Kumar Koshariya



JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY

Department of CSE/IT

Jaypee Institute of Information Technology University,

ABSTRACT

The Bank Account Management System is an application for maintaining a person's account in a bank. In this project we tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System. The purpose of project is in partial fulfilment of the requirements of customer using the banking system. The Design and development of this Bank Management system provides a more secured approach in managing bank customer's information which strengthens the relationships between banks and their customers by providing the right solutions to improve customer satisfaction. The programming language used to develop project C++.

The Domain 'Banking System keeps the day by day tally record as a complete banking. It can keep the information of Account type, account opening form, Deposit, Withdrawal, and Searching the transaction, Transaction report, Individual account opening Form. The Bank Account Management System undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manual systems, which are overcome by this software. The system is designed as an interactive and thus increase the efficiency of the work

ACKNOWLEDGEMENT

We would like to acknowledge all those without whom this project would not have been successful. Firstly, we would wish to thank our SDF Lab teacher Dr. Arti Jain and Mr. Rupesh Kumar Koshariya who guided us throughout the project and gave his immense support. She made us understand how to successfully complete this project and without him, the project would not have been complete.

This project has been a source to learn and bring our theoretical knowledge to the real-life world. So, we would really acknowledge his help and guidance for this project.

we would also like to our Batch-mates who have always been there whenever needed.

Once again, thanks to everyone for making this project successful.

Any valuable suggestions or amendments further for our project will be entertained and looked upon.

Table of content



INTRODUCTION.....	Page 1
● What is Bank Management	
LANGUAGE DESCRIPTION.....	Page 2
● Concept	
PROBLEM DEFINATION.....	Page 3
UML DIAGRAM.....	Page 4
● Diagram Information	
PROJECT CODE.....	Page 7
MODULES IN PROGRAM SCREENSHOT.....	Page 19
● CREATE AN ACCOUNT	
● DEPOSIT MONEY	
● WITHDRAW MONEY	
● BALANCE ENQUIRY	
● ACCOUNT HOLDER LIST	
● CLOSE ACCOUNT	
● UPDATE AN ACCOUNT	
● EXIT	
NEW THINGS LEARNED.....	Page 23
● Clock Making	
FUTURE SCOPE.....	Page 24
CONCLUSION.....	Page 25
REFERENCE.....	Page 26

INTRODUCTION

What is Bank Management..??

There are many definitions of bank management. In general, bank management refers to the process of managing the bank's statutory activity.

Every successful banker has to perform managerial responsibilities along with technical banking activities. Although a bank is a financial institution like other businesses, its main objective is to maximize its wealth by earning profit. Bank business is different from other types of businesses. it and we hope you all like i

But here we just use our knowledge to form the outer layer of the bank and uses our codes to demonstrate We hope you like it

We hope you all like it.



LANGUAGE DESCRIPTION

The project is based on the concepts of C++ Programming and uses many wide ranges of things used in C programming. Here the concepts used in the program are as follows :-

- Classes
- Polymorphism
- Data Abstraction
- Method Overriding
- Scope Resolution
- Switch-Case
- User defined function
- Inbuilt Functions
- Arrays
- Pointers
- Conditional statement
- File Handling
- Calculation

INTERESTING THINGS USED IN PROGRAM

- Clock
- Beep Sound
- Color variants
- Patterns
- Account type(Saving, Current, Fixed)

Problem Definition:

Although the basic type of services offered by a bank depends upon the type of bank and the country, services provided usually include: Taking deposits from their customers and issuing current or checking accounts and savings accounts to individuals and business. Extending loans to individuals and business, Cashing cheque. Facilitating money transactions such as wire transfer and cashiers cheque, Consumer & commercial financial advisory services, .

Motivation behind this project:

To develop a system that will overlook the activities going transaction the particular bank without manual processing. All transaction will be updated automatically by using the information stored in record. The main motive behind this project is to develop a system which will be able to handle the overall tasks going inside the institutions without much effort.

UML DIAGRAM

<<Generic Class>>
Bank Account

- Money_Depoist: int
- type: char
- account_number: int
- name : char[60]

+ report(): void
+ depositMoney: int
+ create_Bank_Account: void
+ deposit(int): int
+ accNo: int
+ Display_Account: void
+ Updation: void
+ withdrawl(int): int

UML DIAGRAM DESCRIPTION

In the class Bank Account, there are four variables, ie, Money_Deposit of int type, Type of char type, account number of int type, name of char type with maximum limit of 60 characters. The class has following functions:

1. Report() : It is of void return type and reports name, account number, type of account and balance.
2. depositMoney(): It is of int return type and it displays current balance.
3. Create_Bank_Account: It is of void return type and it creates bank account.
4. deposit(int): It is of int return type and it performs deposition of money in account.
5. accNo: It is of int return type and it displays account number
6. Display_Account: It is of void return type and it displays the details of account such as name, account type, current balance etc
7. Updation: It is of void return type and it updates the account balance when deposition/withdrawal of money occurs.
8. withdrawal(int): It is of int return type and performs withdrawal of money from current balance

“ Certainly not every good programmer is object-oriented, and not every object -oriented programmer is good “

:- Bjarne Stroustrup

PROJECT CODE

/*****

Starting of code

*******/**

```
#include<iostream>
#include<fstream>
#include<cctype>
#include<iomanip>
#include <ctime>
#include <cstdlib>
using namespace std;
int seconds, minutes, hours;
string str;
class Bank_Account
{

int Money_Deposit;
char type;
int account_number;
char name[70];
public:

void report() const;
int depositMoney() const;
void create_Bank_Account();
void deposite(int);
int accNo() const;
void Display_Account() const;
void Updation();
char rettype() const;
void withdrawl(int);
};
void print_basic();
```

```

void clockie()
{
    int sec_prev=0;

    // storing total seconds
    time_t total_seconds=time(0);
    struct tm* ct=localtime(&total_seconds);

    seconds=ct->tm_sec;
    minutes=ct->tm_min;
    hours=ct->tm_hour;

    //converting it into 12 hour format
    if(hours>=12)
        str="PM";
    else
        str="AM";
    hours=hours>12?hours-12:hours;
}

void Bank_Account::Updation()
{
    system("color 7");
    cout<<"\n\tBank_Account No. : "<<account_number;
    cout<<"\n\tBank Account Holder Name : ";
    cin.ignore();
    cin.getline(name,50);
    cout<<"\n\tType of Bank Account : ";
    cin>>type;
    type=toupper(type);
    cout<<"\n\tBalance Total-Money : ";
    cin>>Money_Deposit;
}

void Bank_Account::create_Bank_Account()

```

```

{
    system("CLS");
    system("color 4");
    cout<<"\n\tPlease Enter the Bank Account No. : ";
    cin>>account_number;
    cout<<"\n\n\tPlease Enter the Name of the Bank Account holder : ";
    cin.ignore();
    cin.getline(name,50);
    cout<<"\n\tPlease Enter Type of the Bank Account (C/S) : ";      //C :-CURRENT
    S:- SAVING
    cin>>type;
    type=toupper(type);
    cout<<"\n\tPlease Enter The Starting Total-Money : ";
    cin>>Money_Deposit;
    cout<<"\n\n\t.....Bank Account Created.....";
}
void Bank_Account::Display_Account() const
{
    cout<<"\n\tBank Account No. : "<<account_number;
    cout<<"\n\tBank Account Holder Name : ";
    cout<<name;
    cout<<"\n\tType of Bank Account : "<<type;
    cout<<"\n\tBalance Total-Money : "<<Money_Deposit;
}
int Bank_Account::accNo() const
{
    return account_number;
}

char Bank_Account::rettype() const
{
    return type;
}
void Bank_Account::report() const
{

```

```

cout<<account_number<<setw(10)<<""<<name<<setw(10)<<""<<type<<setw(6)<<
Money_Deposit<<endl;
}
void Bank_Account::depsit(int x)
{
    Money_Deposit+=x;
}
void Bank_Account::withdrawl(int x)
{
    Money_Deposit-=x;
}
int Bank_Account::depositMoney() const
{
    return Money_Deposit;
}
void write_Bank_Account();
void display_sp(int);
void display_all();

void delete_Bank_Account(int);
void Money_Deposit_withdraw(int, int);
void Updation_Bank_Account(int);
int main()
{
    char ch;
    int num;
    do
    {
        clockie();
        cout << (hours<10?"0":"") << hours <<":" << (minutes<10?"0":"") << minutes <<
":" << (seconds<10?"0":"") << seconds << " " << str <<endl;
        print_basic();
        cin>>ch;
        switch(ch)

```

```

{
case '1':
    write_Bank_Account();
    break;
case '2':
    system("CLS");
    cout<<"\n\n\tPlease Enter The Bank Account No. : "; cin>>num;
    Money_Deposit_withdraw(num, 1);
    break;
case '3':
    system("CLS");
    cout<<"\n\n\tPlease Enter The Bank Account No. : "; cin>>num;
    Money_Deposit_withdraw(num, 2);
    break;
case '4':
    system("CLS");
    cout<<"\n\n\tPlease Enter The Bank Account No. : "; cin>>num;
    display_sp(num);
    break;
case '5':
    display_all();
    break;
case '6':
    system("CLS");
    cout<<"\n\n\tPlease Enter The Bank Account No. : "; cin>>num;
    delete_Bank_Account(num);
    break;
case '7':
    system("CLS");
    cout<<"\n\n\tPlease Enter The Bank Account No. : "; cin>>num;
    Updation_Bank_Account(num);
    break;
case '8':
    system("CLS");

```

```

system("color 2");
{
cout<<"\n\n\n\n *** This C++ Project is developed by F8!  *****\n\n";
cout<<"      _____ Thank you for Your Time _____\n";
cout<<"                                     ";
cout<<"                                     ";
cout<<"

```

Shivansh Pandey, [18-05-2022 13:20]

```

\n";
}
system("color 6");
cout<<" TEAM MEMBERS :-RAHI AGARWAL\n";
cout<<" ENROLL NO :-9921103145\n\n";
system("pause>nul");
system("color 2");
cout<<" TEAM MEMBERS :-ARYAN MISHRA\n";
cout<<" ENROLL NO :-9921103150\n\n";
system("pause>nul");
system("color 4");
cout<<" TEAM MEMBERS :-ADITI BHARGAVA\n";
cout<<" ENROLL NO :-9921103148\n\n";
system("pause>nul");
system("color 3");
cout<<" TEAM MEMBERS :-SHIVANSH PANDEY\n";
cout<<" ENROLL NO :-9921103152\n\n";
system("pause>nul");
break;
default :cout<<"\a";
}
cin.ignore();
cin.get();
}while(ch!='8');

```

```

return 0;
}
void write_Bank_Account()
{
    Bank_Account ac;
    ofstream outputFile;
    outputFile.open("Bank_Account.dat",ios::binary|ios::app);
    ac.create_Bank_Account();
    outputFile.write(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
    outputFile.close();
}
void delete_Bank_Account(int n)
{
    Bank_Account ac;
    ifstream inFile;
    ofstream outFile;
    inFile.open("Bank_Account.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        return;
    }
    outFile.open("Temp.dat",ios::binary);
    inFile.seekg(0,ios::beg);
    while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(Bank_Account)))
    {
        if(ac.accNo() != n)
        {
            outFile.write(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
        }
    }
    inFile.close();
    outFile.close();
    remove("Bank_Account.dat");
}

```



```

rename("Temp.dat","Bank_Account.dat");
cout<<"\n\nRECORD Deleted ..";
}
void display_sp(int n)
{
    Bank_Account ac;
    bool flag=false;
    ifstream inFile;
    inFile.open("Bank_Account.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        return;
    }
    cout<<"\n\tBALANCE DETAILS\n";
    while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(Bank_Account)))
    {
        if(ac.accNo()==n)
        {
            ac.Display_Account();
            flag=true;
        }
        inFile.close();
    }
    if(flag==false)
        cout<<"\n\n\tBank Account number does not exist";
}
void display_all()
{
    system("CLS");
    Bank_Account ac;
    ifstream inFile;
    inFile.open("Bank_Account.dat",ios::binary);
    if(!inFile)
    {

```

```

cout<<"File could not be open !! Press any Key...";
return;
}
cout<<"\n\n\t\tBANK ACCOUNT HOLDER LIST\n\n";
cout<<"-----\n";
cout<<"A/c no.    NAME        Type  Balance\n";
cout<<"-----\n";
while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(Bank_Account)))
{
    ac.report();
}
inFile.close();
}
void Updation_Bank_Account(int n)
{
    bool found=false;
    Bank_Account ac;
    fstream File;
    File.open("Bank_Account.dat",ios::binary|ios::in|ios::out);
    if(!File)
    {
        cout<<"File could not be open !! Press any Key...";
        return;
    }
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
        if(ac.accNo()==n)
        {
            ac.Display_Account();
            cout<<"\n\n\tPlease Enter The New Details of Bank Account"<<endl;
            ac.Updation();
            int pos=(-1)*static_cast<int>(sizeof(Bank_Account));
            File.seekp(pos,ios::cur);

```

```

    File.write(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
    cout<<"\n\n\tRECORD UPDATED";
    found=true;
    }}
File.close();
if(found==false)
    cout<<"\n\n\tRECORD Not Found ";
}
void Money_Deposit_withdraw(int n, int option)
{
    int amt;
    bool found=false;
    Bank_Account ac;
    fstream File;
    File.open("Bank_Account.dat", ios::binary|ios::in|ios::out);
    if(!File)
    {
        cout<<"File could not be open !! Press any Key...";
        return;
    }
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
        if(ac.accNo()==n)
        {
            ac.Display_Account();
            if(option==1)
            {
                cout<<"\n\n\tTO DEPOSIT MONEY ";
                cout<<"\n\n\tPlease Enter The Total-Money to be Money Deposited: ";
                cin>>amt;
                ac.depsit(amt);
            }
            if(option==2)

```

```

{
cout<<"\n\n\tTO WITHDRAW MONEY";
    cout<<"\n\n\tPlease Enter The Total-Money to be withdraw: ";
    cin>>amt;
    int bal=ac.depositMoney()-amt;
    if(bal<0)
        cout<<"Insufficiency balance";
    else
        ac.withdrawl(amt);
    }
int pos=(-1)*static_cast<int>(sizeof(ac));
File.seekp(pos,ios::cur);//fn1353
File.write(reinterpret_cast<char *> (&ac), sizeof(Bank_Account));
cout<<"\n\n\tRECORD UPDATED";
found=true;
    }
    }
    File.close();
if(found==false)
    cout<<"\n\n\tRECORD Not Found ";
}
void print_basic()
{
system("CLS");
;
system("color 9");
    cout << "\t\t\t\t\t";
    string str ="\n\n\t\t\t*****
*****";
    string str2 ="\n\n\t\t\t*****
*****";
    for (int i = 0; i < 82; i++)
    {
        // Sleep(50);
        cout << str[i];

```

BANK MANAGEMENT SYSTEM

JAYPEE BANK OF NOIDA

```

}
cout << "\n\t\t\t ";
for (int i = 0; i < 82; i++)
{
    // Sleep(50);
    cout << str2[i];
}
cout<<"\n\n\n\t\t1. CREATE AN ACCOUNT";
cout<<"\n\t\t2. DEPOSIT MONEY";
cout<<"\n\t\t3. WITHDRAW MONEY";
cout<<"\n\t\t4. BALANCE ENQUIRY";
cout<<"\n\t\t5. ACCOUNT HOLDER LIST";
cout<<"\n\t\t6. CLOSE ACCOUNT";
cout<<"\n\t\t7. UPDATE AN ACCOUNT";
cout<<"\n\t\t8. EXIT";
cout<<"\n\n\n\t\tSelect Your Option (1-8): ";
}

```

/*****

Ending of code

*****/

SNAPSHOT

Main screen

```
***** BANK MANAGEMENT SYSTEM *****
***** JAYPEE BANK OF NOIDA *****

1. CREATE AN ACCOUNT
2. DEPOSIT MONEY
3. WITHDRAW MONEY
4. BALANCE ENQUIRY
5. ACCOUNT HOLDER LIST
6. CLOSE ACCOUNT
7. UPDATE AN ACCOUNT
8. EXIT

Select Your Option (1-8): _
```

1) Create an Account

```
Please Enter the Bank Account No. : 123456789

Please Enter the Name of the Bank Account holder : Rahi

Please Enter Type of the Bank Account (C/S) : C

Please Enter The Starting Total-Money : 5000

.....Bank Account Created.....
```

2) Deposit Money

Please Enter The Bank Account No. : 123456789

Bank Account No. : 123456789

Bank Account Holder Name : Rahi

Type of Bank Account : C

Balance Total-Money : 5000

TO DEPOSIT MONEY

Please Enter The Total-Money to be Money Deposited: 5000

RECORD UPDATED_

3) Withdraw Money

Please Enter The Bank Account No. : 123456789

BALANCE DETAILS

Bank Account No. : 123456789

Bank Account Holder Name : Rahi

Type of Bank Account : C

Balance Total-Money : 5000

4) Balance Details

Please Enter The Bank Account No. : 123456789

Bank Account No. : 123456789

Bank Account Holder Name : Rahi

Type of Bank Account : C

Balance Total-Money : 5000

TO WITHDRAW MONEY

Please Enter The Total-Money to be withdraw: 2000

RECORD UPDATED

5) Account Holder list

BANK ACCOUNT HOLDER LIST

A/c no.	NAME	Type	Balance
123456789	Rahi	C	3000

6) Account deleted

Please Enter The Bank Account No. : 123456789

RECORD Deleted ..

7) Exit

```
*** This C++ Project is developed by F8! ****
```

```
____ Thank you for Your Time ____
```

```
TEAM MEMBERS :-RAHI AGARWAL  
ENROLL NO :-9921103145
```

```
TEAM MEMBERS :-ARYAN MISHRA  
ENROLL NO :-9921103150
```

```
TEAM MEMBERS :-ADITI BHARGAVA  
ENROLL NO :-9921103148
```

```
TEAM MEMBERS :-SHIVANSH PANDEY  
ENROLL NO :-9921103152
```

NEW THING LEARNED

CLOCK MAKING

```
int sec_prev=0;

// storing total seconds
time_t total_seconds=time(0);

//getting values of seconds, minutes and hours
struct tm* ct=localtime(&total_seconds);

seconds=ct->tm_sec;
minutes=ct->tm_min;
hours=ct->tm_hour;

//converting it into 12 hour format
if(hours>=12)
    str="PM";
else
    str="AM";
hours=hours>12?hours-12:hours
```

Please Enter the Bank Account No. : 23

Please Enter the Name of the Bank Account holder : Shivansh

Please Enter Type of the Bank Account (C/S) : C

Please Enter The Starting Total-Money : 9000

.....Bank Account Created.....

12:57:52 PM

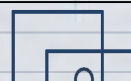
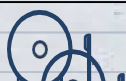
REPORT

The main purpose of developing bank management system is to design an application, which could store bank data and provide an interface for retrieving customer related details with 100% accuracy.

Salient features are as follows :-

- This bank management system also allow user to add new customer account, delete account and user can also modify existing user account information.
- Using this system user can also search any individual account in few seconds.
- Using our bank management system user can also check any transaction in any account.
- Our system also provide security check to reduce fraud.
- The main object of this system is to provide a secure system. Our system is password protected and it only allows authorized user to access various functions available in the system.
- Using this system user can manage following account type
 - Savings Account
 - Current Account
 - Fixed Account

The proposed system is more efficient, fast, reliable, user friendly. Over and above the proposed system does not have any possibility of data loss during processing



FUTURE SCOPE

The scope of the Bank Management System extends to all the users who wish for easy banking facilities. This software product will be used for storing user's account information and the transactions made by them.

- 1. This project will help the bankers in fast reporting.**
- 2. This project enable banker to maintain a great database of all Customer' s details from the software.**
- 3. Project will enable to see report regarding query.**
- 4. It is easy to maintain in future prospect**



CONCLUSION

Bank management system is a virtualization of transactions in banking system. The banking system are used manual working but when we used online banking system it is totally virtualization process which avoid manual process and converts it in automatic process . Bank management system is saving the time with accuracy than bank manual system.

This was our project of System Design about “Bank management system”.Development of this System takes a lot of efforts. we think that this system gave alot of satisfaction.

Though every task is never said to be perfect in this development field even more improvement may be possible in this system. We learnt so many things andgained a lot of knowledge about programming field. I hope this will prove fruitful



REFERENCES

Book

P.M. Morse and H. Feshback, Methods of Theoretical Physics. New York: McGraw Hill, 1953.

Journal Article

GitHub - alxschwartz/codex_py2cpp: Convert python code into c

https://github.com/alxschwartz/codex_py2cpp

Online:

Online C++ Compiler - Online C++ Editor - Run C++ Online

<https://www.jdoodle.com/online-compiler-c++>

WEBSITES REFERRED

1. www.google.com
2. www.slideshare.net
3. www.stackoverflow.com
4. <https://www.geeksforgeeks.org/c-plus-plus/>
5. www.youtube.com
6. <https://www.javatpoint.com/cpp-program>

***** (: THANK YOU :) *****