# OBJECTIVE

This project on Bank management aims at implementation of various operations for the functioning of a computer based customer care interface used for entertainment of various customer related services such as creation of new account , Withdrawal of money , Deposition of money, etc. Through the use of proper data structure we’ve tried to achieve these objectives to the maximum extent .

The code aims at ease of access of the user , though which user can access the account creation, withdrawal , deposition, balance enquiry , etc.

# INTRODUCTION

We All know that Banks are one of those important enterprises those have to store large amount of data. In terms of Quantity the data is infinitely huge and hence without proper channelization and arrangement of data storage and data access it will become really cumbersome to hold millions of accounts in a populated country like India .

Regarding the storage , banks use huge data storage devices to store data but to access data it is really necessary to have to efficient user interface to enable user to easily access the services offered .

In this project too, we have tried to implement these operations , using a user friendly approach . We start from the Humble Main menu to give users the options to firstly create the account then performing various operations like money deposition , money withdrawal , etc. each in a new window, by using switch case . Furthermore the program starts with a password protected window after which we are guided to the main menu and various operations are displayed thereafter.

# ALGORITHM

**STEP 1:** Password ( ) check

If passwords match proceed through steps 2

**STEP 2**: press 1 for account creation

Press 2 for account details

Press 3 for Deletion

Press 4 for Deposition

Press 5 for Account Enquiry

Press 6 to Exit

**STEP 3:** input choice

**STEP 4**: while condition is true perform step 5 onwards

**STEP 5**: switch(choice)

***STEP 5.1***: if (choice==1)

***STEP 5.1.1***: create new account

***STEP 5.1.2:*** input account holder’s name

***STEP 5.1.3***: input the Address

**STEP 5.1.4:** Input date of birth

***STEP 5.1.5***: input account creation date

***STEP 5.1.6:*** input money to be deposited

***STEP 5.2.***: if(choice ==2)

***STEP 5.2.1.:*** Show customer details

***STEP 5.2.2***: print account holders name

***STEP 5.2.3***: print date of birth of customer

**STEP 5.2.4:**  print the account creation date

***STEP 5.2.5***: print the money deposited

***STEP 5.2.6***: else print account detail does not exist

***STEP 5.3***: if (choice==3) Delete account

***STEP 5.5.1***: input the customer’s account number

***STEP 5.5.2***: Delete Account

***STEP 5.6***: if(choice==4) Deposit Money

***STEP 5.6.1***: Input account number

**STEP 5.6.2**: Input The Amount to be Deposited

***STEP 5.6.3***: End of money Deposition

**STEP 6:** If (choice==5) do balance enquiry

***STEP 6.1.1***: input the account Number

***STEP 6.1.2***: input the name of the account holder

***STEP 6.2***: Completed Balance Enquiry

***STEP 7***: If (choice==6) do money withdrawal

***STEP 7.1***: input name of the customer

***STEP 7.2***: input the name of the Customer

***STEP 7.3***: Input the amount to be withdrawn

***STEP 7.4***: completed Money Withdrawal

***STEP 8***: If (choice==7)

***STEP 8.1***: Exit the Program Execution

***STEP 9***: otherwise ‘Beep’ Displaying Error message

***STEP 10***: END OF main( )