#include <iostream>

#include <cmath>

#include <iomanip>

#include <string>

#include <fstream>

#include <cctype>

#include <cstdlib>

#include<Windows.h>

#include<stdio.h>

#include<stdlib.h>

using namespace std;

double current\_balance,sfix\_balance,jfix\_balance,loan\_repayment;

string searchusername,searchpassword,file\_name;

char searchtype[30]="";

double beneficiary\_balance,beneficiary\_sfix,beneficiary\_jfix,beneficiary\_loan\_repayment;

string beneficiaryname,beneficiarypassword,beneficiary\_type;

void test(char[], int); //To validate the input is digit

void loading(); //Text animation which display loading

void Confirmation(); //To confirm the customer transaction

void reg(); //To register a new ID

void login(); //To login

void dp\_fund(); //To move the normal saving to the fixed deposit

void fund\_transfer(); //To transfer the money from 1 user to another user

void modify\_acc(); //To modify password

void loans(); //Allow user to apply loans

void insurance(); //Allow user to buy insurance

void payment(); //To help the user pay the bill, prepaid and insurance,loan

void overwrite\_file(); //Update customer details

void balance(); //To display customer financial details

void error(); //Display respond the incorrect choice entered by the user

void Planning\_assistant(); //Calculate the interest of deposit for user

void log\_out(); //To exit the program

//Display the name of the bank and the student name,ID

void welcome()

{

system("COLOR 08");

cout << "\n\t\t\t\t WELCOME " << endl << endl;

Sleep(200);

cout << "\t\t\t\t TO " << endl << endl;

Sleep(200);

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

Sleep(200);

cout << " CCCCCC II TTTTTTTTTT YY YY BBBBB AA NN N K KKK " << endl;

Sleep(200);

cout << " CC II TT YY YY B BB A A N N N K KK " << endl;

Sleep(200);

cout << " CC II TT YYY B BBB AAAAAA N N N KK " << endl;

Sleep(200);

cout << " CC II TT YYY B BB AA AA N N N K KK " << endl;

Sleep(200);

cout << " CCCCCC II TT YYY BBBBB AA AA N NN K KKK " << endl;

Sleep(200);

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

Sleep(200);

cout << endl << endl << endl;

cout << "TIEW JOON TIAN 1501815" << endl;

cout << "PHUANG KE SHONG 1403488" << endl;

cout << "SEE YEN SIN 1403214" << endl<<endl;

system("pause");

system("COLOR 07");

loading();

}

//Text animation

void loading()

{

system("cls");

cout << "\n\nL";

Sleep(100);

cout << "O";

Sleep(100);

cout << "A";

Sleep(100);

cout << "D";

Sleep(100);

cout << "I";

Sleep(100);

cout << "N";

Sleep(100);

cout << "G";

Sleep(100);

cout << ".";

Sleep(100);

cout << ".";

Sleep(100);

cout << ".";

Sleep(100);

system("cls");

}

//Text animation

void text\_animation(char a[100])

{

double k;

for(int i=0 ; a[i] != '\0';++i)

{

for(k=0;k<9000000;++k);

cout<<a[i];

}

}

//Display the menu and let user to register or log in only

void Menu()

{

char decision[20];

char validation[]="AA";

char log\_in[]="1";

char cr8[]="2";

//Display the menu

cout<< "\n\n Welcome to City Bank \n "

" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n \n"

" \4 1. Log in \n\n"

" \4 2. Register \n\n"

" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n ";

cout<<"\nPlease select your option: ";

cin>>decision;

//To validate the decision entered by the user

while(strlen(decision)>=strlen(validation))

{

cout<<"Your option only 1 and 2. \n";

system("pause");

system("cls");

Menu();

}

//To validate the decision,if true will call reg() function

while(strcmp(decision,cr8)==0)

{

reg();

}

//To validate the decision,if true will call login() function

while(strcmp(decision,log\_in)==0)

{

login();

}

cout<<"Your option only 1 and 2. \n";

system("pause");

system("cls");

Menu();

}

//Display the menu after the user login

void showMenu()

{ int decision2;

time\_t now=time(0);

char\* dt = ctime(&now);//To display the date,day and time

system("cls");

system("COLOR 0F");

cout<<endl

<<"\n Welcome to City Bank \n "

" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n\n"

" \4 1 - Deposit funds \n"

" \4 2 - Fund transfer \n"

" \4 3 - Payment \n"

" \4 4 - Modify account \n"

" \4 5 - Planning assistant \n"

" \4 6 - Loans \n"

" \4 7 - Insurance \n"

" \4 8 - Balance enquiry \n"

" \4 9 - Log out \n\n\n"

" "<<dt

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout<<"\n Please select your option:"; // prompt for option

//Validate the option must be digit

while (! (cin>> decision2))

{

cout<<"\nMust be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

switch (decision2)

{

case 1: dp\_fund();

break;

case 2: fund\_transfer();

break;

case 3: payment();

break;

case 4: modify\_acc();

break;

case 5: Planning\_assistant();

break;

case 6: loans();

break;

case 7: insurance();

break;

case 8: balance();

break;

case 9: log\_out();

break;

default:error();

system("cls");

return showMenu();

}

}

int main()

{

welcome();

loading();

text\_animation("\n\n Welcome to City bank ");//To pass the value to the text animation function

system("cls");

Menu();//To call the main menu

system ("pause");

return 0;

}

void login()

{

system ("cls");

string username,file\_name,password;

char type[30]="";

int count\_password=0,count\_acctype=0;

fstream file;

char a[6]="";

srand(time(NULL));//to ensure the random number generated is different at a time

for(int i=0;i<6;i++)

{

a[i]=rand()%100;//to generate a random number

}

cin.ignore();

cout<<"\n Login"<<endl<<endl;

cout<<"------------------------------------------------"<<endl<<endl;

cout<<"Please Enter Username : "; //prompt the user enter the username

getline(cin,username);

//To ignore the \n

while (username.length() < 1 )

{

getline(cin,username);

}

file\_name=username + ".txt";//change the username to the name of file

cout << "\nPlease Enter Password : "; //prompt for password

getline(cin,password);

cout<< "\nPlease enter your account type(s/c) : ";//prompt for account type

cin.get(type,30);

for(int i=0;i<30;i++)

{

type[i]=toupper(type[i]);//to upper case the input entered

}

cin.ignore();

file.open(file\_name,ios::in);//open the file

//To validate the existing of the account

if(file.fail())

{

cout << "\n The username is not registered yet.\n\n ";

system("pause");

system("cls");

Menu();

}

//read all the data from the file

if(file)

{

getline(file,searchusername);

getline(file,searchpassword);

file.get(searchtype,30);

file>>current\_balance;

file>>sfix\_balance;

file>>jfix\_balance;

file>>loan\_repayment;

}

//comparing the password enter by user and the password from the account

while(password != searchpassword)

{

++count\_password;//to count the number of incorrect password entered

if(count\_password<=2)//prompt the user to reenter the password

{

cout<<"\nThe password that you entered is incorrect.\n"

<<"Please re-enter your password.\n";

getline(cin,password);

}

else//terminate the account by replacing the password generated from random number

{

system("cls");

cout<<"You enter the wrong password continuously for 3 times.\n\n";

cout<<"The account was terminated...\n\n";

searchpassword=a;

cout<<searchpassword;

overwrite\_file();

system ("pause");

exit(0);

}

}

//comparing the type of account enter by user and the type of account from the file

while(strcmp(type,searchtype)!=0)

{

++count\_acctype;

if(count\_acctype<=2)//to prompt the user to re-enter the account type

{

cout<<"\nInvalid account type.\n"

<<"Please re-enter your account type.(s/c)\n";

cin.get( type,30);

for(int i=0;i<30;i++)

{

type[i]=toupper(type[i]);

}

cin.ignore();

}

else//terminate the account by replacing the password generated from random number

{

system("cls");

cout<<"You enter the wrong account type continuously for 3 times.\n\n";

cout<<"The account was terminated...\n\n";

searchpassword=a;

overwrite\_file();

system ("pause");

exit(0);

}

}

//if the password and account type enter is true,call the showMenu() function

while((password== searchpassword)&&(strcmp(type,searchtype)==0))

{

cout<<"\nLog in successful.\n\n";

cout<<"------------------------------------------------"<<endl;

system ("pause");

system ("cls");

showMenu();

}

file.close();

}

void reg()

{

char sav\_acc[]="S";

char cur\_acc[]="C";

fstream file;

system ("cls");

cin.ignore();

REG:cout<<"\n Register \n\n"

"------------------------------------------------------------\n\n";

cout<<"Enter the username(at LEAST 6 characters): ";//prompt for username

getline(cin,searchusername);

//to validate the username entered is more than 6 character

while (searchusername.length() < 6 )

{

cout << "Username MUST at least 6 characters): ";//prompt the user to re-enter the name

getline(cin,searchusername);

}

file\_name=searchusername + ".txt";//change the name to the file name and file type that will be create

file.open(file\_name,ios::in);

if(file) //to validate the existing of the account

{

cout<<"This username has been used.\n"

<<"Please try with other username.\n";

system("pause");

system("cls");

goto REG;

}

cout<<"\nPlease key in password ( MUST be 6 characters): ";//prompt for password

getline(cin,searchpassword);

while (searchpassword.length() != 6 )//validate the password must more than 6 characters

{

cout << "Password( MUST BE 6 characters): " << endl;//prompt the user to reenter the password

getline(cin,searchpassword);

}

cout<<"\nEnter searchtype of account you want to register.\n"//prompt the user to select the account type

<<"Press S - saving account.\n"

<<"Press C - current account.\n";

cin>>searchtype;

for(int i = 0;i<30;i++)

{

searchtype[i]=toupper(searchtype[i]);

}

//to ensure the user only key in "c" ,"C","s" or "S" only

while (strlen(searchtype)>2)

{

STYPE:cout<<"Please key in S or C only:";

cin>>searchtype;

for(int i = 0;i<30;i++)

{

searchtype[i]=toupper(searchtype[i]);

}

}

//perform these statements if the user entered "s" or "S"

if(strcmp(searchtype,sav\_acc)==0)

{

cout<<"\nThe minimun saving amount is RM200 and maximum saving is RM10,000.\n";

cout<<"Pls enter your saving amount : ";

while (! (cin>> current\_balance))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(current\_balance <200 || current\_balance>10000)

{

cout<<"Invalid amount was entered.\n"

<<"Please re-enter the saving amount : ";

while (! (cin>> current\_balance))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

}

//to save the latest information to the file

overwrite\_file();

cout<<"\nCongratulation.Your account has been successfully registered.\n"

<<"------------------------------------------------------------\n";

system("pause");

system("cls");

Menu();

}

//perform these statements when user entered "c" or "C"

else if(strcmp(searchtype,cur\_acc)==0)

{

cout<<"\nThe minimun saving amount is RM500 and maximum saving is RM50,000.\n";

cout<<"Please enter your saving amount: ";

while (! (cin>> current\_balance))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(current\_balance <500 || current\_balance>50000)

{

cout<<"Invalid amount was entered.\n"

<<"Pls re-enter the saving amount : ";

while (! (cin>> current\_balance))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

}

//to update the user latest details

overwrite\_file();

cout<<"\nCongratulation.Your account has been successfully registered.\n\n";//display this output to show the account has been successfully registered

system("pause");

system("cls");

Menu();

}

else//prompt the user to re-enter the account type

{

goto STYPE;

}

}

void dp\_fund()

{

system ("cls");

int sfix\_amt,jfix\_amt, option;

char decision;

//display the choices of fixed deposit

cout<<"\n Type of deposit account\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<" 1. Senior fixed deposit\n"

<<" 2. Junior fixed deposit\n"

<<" 3. Return main menu \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Please select your option : ";//prompt for option

cin>>option;

switch(option)

{

case 1 :

SFIX:system ("cls");

//display the terms and conditions

cout<<"\n Senior Fixed Deposit\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Terms and condition\n\n"

<<"1)This account is only eligible for individuals who are holders \n"

<<" of City bank saving and current account.\n"

<<"2)The customer shall place a minimum deposit of RM 2000 with \n"

<<" tenure of one month.\n"

<<"3)The maximum deposit amount is RM5000 per transaction\n"

<<"4)The monthly interest rate will be 5%.\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

Confirmation();

system("cls");

cout<<"\n Senior fixed deposit \n";

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<"Enter amount to be transfer into S-fixed deposit : RM ";//prompt the user to enter the amount to be transfer

while (! (cin>> sfix\_amt))//to validate the input must be digit

{

cout<<"\nMust be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

if(sfix\_amt<2000 || sfix\_amt>5000)//to check the amount entered fullfill the terms and conditions

{

system("cls");

cout<<"\nInvalid amount.\n";

cout<<"Please read the terms and condition...\n";

system("pause");

goto SFIX;

}

else//to check whether the user have the sufficient money to do the transaction

{

if(current\_balance>=sfix\_amt)

{

current\_balance-=sfix\_amt;

sfix\_balance+=sfix\_amt;

cout<<"\nYour transaction is accepted.\n";

cout<<"Your current balance: RM "<<current\_balance<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

overwrite\_file();

system("pause");

balance();

system("pause");

showMenu();

}

else if(sfix\_amt>=current\_balance)

{

system("cls");

cout<<"\nYour account current\_balanced is not sufficient.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

}

break;

case 2:

JFIX:system ("cls");

//display the terms and conditions

cout<<"\n Junior Fixed Deposit\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Terms and condition\n"

<<"1)This account is only eligible for individuals who are holders \n"

<<" of City bank saving account.\n"

<<"2)The customer shall place a minimum deposit of RM200 \n"

<<"3)The maximum deposit amount is RM2000 per transaction.\n"

<<"4)The monthly interest rate will be 8%.\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

Confirmation();

system("cls");

cout<<"\n Junior fixed deposit \n";

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<"Enter amount to be transfer into J-fixed deposit : RM ";//to prompt the user to enter the amount to be transfer

while (! (cin>> jfix\_amt))//validate the input must be digit

{

cout<<"\nMust be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

if(jfix\_amt>2000 || jfix\_amt<200)//to check the amount entered full-fill the terms and conditions

{

cout<<"\nInvalid amount.\n";

cout<<"Please read the terms and conditions...";

system("pause");

goto JFIX;

}

else//to check whether the user have the sufficient money to do the transaction

{

if(current\_balance>=jfix\_amt)

{

current\_balance-=jfix\_amt;

jfix\_balance+=jfix\_amt;

cout<<"\nYour transaction is accepted.\n";

cout<<"Your current balance: RM "<<current\_balance<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

overwrite\_file();

system("pause");

balance();

system("pause");

showMenu();

}

else if(sfix\_amt>=current\_balance)

{

system("cls");

cout<<"\nYour account current\_balanced is not sufficient\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

}

break;

case 3 : showMenu();

break;

default:error();

dp\_fund();

}

}

void fund\_transfer()

{

system("cls");

fstream file;

string balance,name;

string reference;

double amt;

FT: cin.ignore();

cout<<" Fund Transfer \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Please enter the beneficiary name : ";//prompt for beneficiary name

getline(cin,name);

cout<<"\nPlease enter the amount to be transfer :RM ";

while (! (cin>> amt))//to validate the amount entered must be digit

{

cout<<"\nMust be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (amt<0)//to check whether the amount entered is relevant or not

{

cout<<"\nAmount is not relevant.\n";

cout<<"Please key in again:";

cin>>amt;

cin.clear();

}

cin.ignore();

cout<<"\nPlease enter the receipt reference :";//prompt the user to enter the receipt reference

getline(cin,reference);

system("cls");

//display the information entered by the user

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<"Name of beneficiary : "<<name<<endl;

cout<<"Amount to be transfer : RM "<<amt<<endl;

cout<<"Reference : "<<reference<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

Confirmation();//to confirm this process with user

beneficiaryname=name+".txt";//change the name of the beneficiary to the file name and type

file.open(beneficiaryname,ios::in);

if(file)//read the information from the file

{

getline(file,beneficiaryname);

getline(file,beneficiarypassword);

getline(file,beneficiary\_type);

file>>beneficiary\_balance;

file>>beneficiary\_sfix;

file>>beneficiary\_jfix;

file>>beneficiary\_loan\_repayment;

file.close();

if(current\_balance>=amt)//proceed these statements if the user have sufficient money

{

beneficiary\_balance+=amt;

system("cls");

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

beneficiaryname=name+".txt";

//to update the details of beneficiary

file.open(beneficiaryname,ios::out);

file << name << endl;

file <<beneficiarypassword << endl;

file << beneficiary\_type << endl;

file << beneficiary\_balance << endl;

file << beneficiary\_sfix<<endl;

file << beneficiary\_jfix<<endl;

file << beneficiary\_loan\_repayment<<endl;

file.close();

current\_balance-=amt;

cout<<"Your transaction is accepted.\n\n";

cout<<"Your current balance: RM "<<current\_balance<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

overwrite\_file();//update the user details

system("pause");

showMenu();

}

else

{

system("cls");

cout<<"Your account current balanced is not sufficient.\n"

<<"Please check your balance before doing any transaction.\n\n";

system("pause");

showMenu();

}

}

else

{

system("cls");

cout<<"\nThe beneficiary name doesn't exist.\n"

<<"Please make sure you entered the correct beneficiary name.\n\n";

Confirmation();

system("cls");

goto FT;

}

}

void modify\_acc()

{

string newname,newpassword,oldname,oldpassword;

int choice;

system ("cls");

cout<<"\n Modify Account \n\n"

<<"----------------------------------------------\n\n"

<<" Press 1- To modify Password. \n\n"

<<" Press 2- Return main Menu. \n\n"

<<"----------------------------------------------\n\n"

<<"Please enter your choice : ";//prompt the user choice

cin>>choice;

switch(choice)

{

case 1 :

{

cout<<"Please enter your current password.\n";//promt for password

cin.ignore();

getline(cin,oldpassword);

if (oldpassword==searchpassword)//perform these statements when password enter is correct

{

cout<<"Please enter the new password.\n";//prompt for new password

getline(cin,newpassword);

while (newpassword.length() != 6 )

{

cout << "Password( MUST BE 6 characters): " << endl;

getline(cin,newpassword);

}

searchpassword=newpassword;

overwrite\_file();//update the details

cout<<"Your information was updated.\n";

system("pause");

showMenu();

}

else

{

system("cls");

cout<<"Incorrect password was entered.\n";

system("pause");

showMenu();

}

}

break;

case 2 : showMenu();

break;

default:

error();

modify\_acc();

}

}

void loans()

{

fstream file;

char x;

char option;

int loanstype;

double amount1,amount2, amount3;

const double interest1=0.12 ;

const double interest2=0.0925 ;

const double interest3=0.04;

double total1,total2,total3;

string title="loans";

system ("cls");

//display loans type

cout<<endl

<<" TYPES OF LOAN \n\n"

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<" 1. Mortgage Loan"<<endl;

cout<<" 2. Personal Loan"<<endl;

cout<<" 3. Education Loan"<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<" Which loan you want to apply : ";

while (! (cin>> loanstype))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

switch (loanstype)

{

case 1:

system("cls");

//display terms and conditions for mortgage loan

UP:cout<<"\n MORTGAGE LOAN "<<endl<<endl

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n"

<<" \n"

<<" \4 Minimum mortgage loan apply must be RM 5000 and above.\n"

<<" \n"

<<" \4 Interest for mortgage loan is 12.00% of loan amount. \n"

<<" \n"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n\n"

<<"Amount of loan required:RM";

while (! (cin>> amount1))//validate the loan amount entered must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

if (amount1>=5000)//validate the amount full-fill the terms and conditions

{

total1=(amount1)\*interest1;

system("cls");

cout<<"\n\n--------------------------------------------------------------\n\n"

<<" \4 Amount of MORTGAGE loan applied :RM"<<setw(14)<<amount1<<endl<<endl

<<" \4 Total amount of interest per year:RM"<<setw(14)<<total1<<endl<<endl;

cout<<" \4 Total amount need to pay back :RM"<<setw(14)<<total1+amount1<<endl<<endl;

cout<<"--------------------------------------------------------------\n";

Confirmation();//confirm the process with user

current\_balance+=amount1;

loan\_repayment+=total1+amount1;

overwrite\_file();//update the user details

system("cls");

balance();

system ("pause");

}

else

{

cout<<"Invalid amount entered.\n"

<<"Our bank is only provide loan which is RM 5000 and above.\n"

<<"Please re-enter correct amount.\n";

system("pause");

system("cls");

goto UP;

}

break;

case 2 :

system("cls");

//display personal loan terms and conditions

UPP:cout<<"\n PERSONAL LOAN "<<endl<<endl

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n\n"

<<" \4 Minimum personal loan apply must be RM 5000 and above.\n"<<endl

<<" \4 Interest for personal loan is 9.25% of toal loan amount. \n"<<endl

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n\n"

<<"Amount of loan required:RM ";//prompt for amount

while (! (cin>> amount2))//validate the amount entered must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

if(amount2>=5000)//validate the amount full-fill the terms and conditions

{

total2=(amount2)\*interest2;

system("cls");

cout<<"\n\n------------------------------------------------------------\n\n"

<<" \4 Amount of PERSONAL loan applied :RM"<<setw(14)<<amount2<<endl<<endl;

cout<<" \4 Total amount of interest per year:RM"<<setw(14)<<total2<<endl<<endl;

cout<<" \4 Total amount need to pay back :RM"<<setw(14)<<total2+amount2<<endl<<endl;

cout<<"------------------------------------------------------------\n";

Confirmation();//confirm the process with user

loan\_repayment+=total2+amount2;

current\_balance+=amount2;

overwrite\_file();//update the user details

system("cls");

balance();

system ("pause");

}

else

{

cout<<"Invalid amount entered.\n"

<<"Our bank is only provide loan which is RM 5000 and above.\n"

<<"Please re-enter correct amount.\n";

system("pause");

system("cls");

goto UPP;

}

break;

case 3 :

system("cls");

UPPP:cout<<"\n EDUCATION LOAN "<<endl

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n\n"

<<" 1. Minimum loan applied must be RM 5000 and above,\n"<<endl

<<" 2. Interest for education loan is 0.04% of total amount. \n"<<endl

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3"

<<"\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\3\n\n"

<<"Amount of loan required:RM";

while (! (cin>> amount3))//validate the amount entered must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

if(amount3>=5000)//validate the amount full-fill the terms and conditions

{

total3=(amount3)\*interest3;

system("cls");

cout<<"\n\n------------------------------------------------------------\n\n"

<<" \4 Amount of EDUCATION loan applied :RM"<<setw(14)<<amount3<<endl<<endl;

cout<<" \4 Total amount of interest per year:RM"<<setw(14)<<total3<<endl<<endl;

cout<<" \4 Total amount of need to pay back :RM"<<setw(14)<<total3+amount3<<endl<<endl;

cout<<"------------------------------------------------------------\n";

Confirmation();//confirm the process with user

loan\_repayment=total3+amount3;

current\_balance+=amount3;

overwrite\_file();//update the user details

system("cls");

balance();

system ("pause");

}

else

{

cout<<"Invalid amount entered.\n"

<<"Our bank is only provide loan which is RM 5000 and above.\n"

<<"Please re-enter correct amount.\n";

system("pause");

system("cls");

goto UPPP;

}

break;

default:

error();

loans();

}

}

void insurance()

{

fstream file;

system("cls");

int familyinsurance;

int option,total;

int accidientinsurance,homeinsurance,age,tcinsurance;

double hamount,tcamount,accamount;

string title="insurance";

system("cls");

//display type of insurance

cout<<" TYPE OF INSURANCE "<<endl<<endl

<<"################################\n\n";

cout<<" 1. Home Insurance\n";

cout<<" 2. Family Insurance\n";

cout<<" 3. Travel & Car Insurance\n";

cout<<" 4. Accident Insurance\n\n"

<<"################################\n\n";

cout<<"Please key in your option:";//prompt for option

while (! (cin>> option))//validate the option must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

switch (option)

{

case 1:

A:system("cls");

//display types of home insurance

cout<<"\n\n HOME INSURANCE\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n\n";

cout<<" No Type of house Monthly payment(RM)\n";

cout<<" 1. Condo 850\n";

cout<<" 2. Home Owner 1000\n";

cout<<" 3. Specialised home 1500\n";

cout<<" 4. Mobile and manufactured 500\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n";

cout<<"Which plan you want to apply: ";//prompt for option

while (! (cin>> homeinsurance))//validate the option enter must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

Confirmation();

//debit the money from user with relative to the insurance purchase

//update the user details

switch (homeinsurance)

{

case 1:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 850 every month."<<endl;

hamount=850;

while (hamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=hamount;

overwrite\_file();

system("cls");

balance();

system ("pause");

break;

case 2:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 1000 every month."<<endl;

hamount=1000;

while (hamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=hamount;

overwrite\_file();

system("cls");

balance();

break;

case 3:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 1500 every month."<<endl;

hamount=1500;

while (hamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

current\_balance-=hamount;

system ("pause");

overwrite\_file();

system("cls");

balance();

system ("pause");

break;

case 4:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 500 every month."<<endl;

hamount=500;

while (hamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=hamount;

overwrite\_file();

system("cls");

balance();

system ("pause");

break;

default:

error();

goto A;

break;

}

break;

case 2:

SIZE1:system("cls");

//display the terms and conditions for family insurance

cout<<"\n\n FAMILY INSURANCE\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n\n";

cout<<" These family insurance is for a family with minimum 2 members.\n\n";

cout<<" Each of the member will be charge for RM 500.\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n";

cout<<"Please enter your family size : ";

while (! (cin>>familyinsurance ))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

Confirmation();

//debit the money from user with relative to the insurance purchase

while (familyinsurance>=2)

{

total=familyinsurance\*500;

cout<<"Application successful.\nYour account will be debit RM "<<total<<" every month."<<endl;

while (total>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=total;

overwrite\_file();

system("cls");

balance();

system ("pause");

}

cout<<"Invalid family size..."<<endl;

system ("pause");

goto SIZE1;

break;

case 3:

TC:system("cls");

//display the tpyes of travel and car insurance

cout<<"\n\n TRAVEL & CAR INSURANCE\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n\n";

cout<<" No Type of travel & car insurance Monthly payment(RM)\n";

cout<<" 1 E-car takaful 150\n";

cout<<" 2 E-motor takaful 70\n";

cout<<" 3 Travel(Asia) 60\n";

cout<<" 4 Travel(Europe) 120\n";

cout<<" 5 Travel(neither Asia and Europe) 80\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n";

cout<<"Which plan you want to apply: ";

while (! (cin>> tcinsurance))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

Confirmation();

//debit the money from user with relative to the insurance purchase

//update the user details

switch (tcinsurance)

{

case 1:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 150 every month.\n";

tcamount=150;

while (tcamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=tcamount;

overwrite\_file();

system("cls");

balance();

break;

case 2 :

system("cls");

cout<<"Application successful.\nYour account will be debit RM 70 every month.\n";

tcamount=70;

while (tcamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=tcamount;

overwrite\_file();

system("cls");

balance();

break;

case 3:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 60 every month.\n";

tcamount=60;

while (tcamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=tcamount;

overwrite\_file();

system("cls");

balance();

break;

case 4:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 120 every month.\n";

tcamount=120;

while (tcamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=tcamount;

overwrite\_file();

system("cls");

balance();

break;

case 5:

system("cls");

cout<<"Application successful.\nYour account will be debit RM 80 every month.\n";

tcamount=80;

while (tcamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=tcamount;

overwrite\_file();

system("cls");

balance();

break;

default:

error();

goto TC;

}

break;

case 4:

select:system("cls");

cout<<"\n\n ACCIDENT INSURANCE\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n\n";

cout<<"PLAN RANGE OF AGE Monthly payment(RM)\n";

cout<<"1 0-12 100\n";

cout<<"2 13-21 300\n";

cout<<"3 22-60 600\n";

cout<<"4 ABOVE 60 958\n\n"

<<"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4"

"\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\4\n";

cout<<"Please enter your choice:";

while (! (cin>>age))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

Confirmation();

//debit the money from user with relative to the insurance purchase

//update the user details

switch(age)

{

case 1:

system("cls");

cout<<"PLAN A>>\n";

cout<<"Application successful.\nYour account will be debit RM 100 every month."<<endl;

accamount=100;

while (accamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system("pause");

current\_balance-=accamount;

overwrite\_file();

system("cls");

balance();

break;

case 2:

system("cls");

cout<<"PLAN B\n";

cout<<"Application successful.\nYour account will be debit RM 300 every month."<<endl;

accamount=300;

while (accamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system("pause");

current\_balance-=accamount;

overwrite\_file();

system("cls");

balance();

break;

case 3:

system("cls");

cout<<"PLAN C\n";

cout<<"Application successful.\nYour account will be debit RM 600 every month."<<endl;

accamount=600;

while (accamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system("pause");

current\_balance-=accamount;

overwrite\_file();

system("cls");

balance();

break;

case 4:

system("cls");

cout<<"PLAN D\n";

cout<<"Application successful.\nYour account will be debit RM 985 every month."<<endl;

accamount=985;

while (accamount>current\_balance)

{

system("cls");

cout<<"Insufficient amount of money to buy the insurance.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

system ("pause");

current\_balance-=accamount;

overwrite\_file();

system("cls");

balance();

break;

default:

system("cls");

error();

goto select;

}

default :

go:system("cls");

error();

insurance();

}

}

//update the user details

void overwrite\_file()

{

fstream file;

file\_name=searchusername + ".txt";

file.open(file\_name,ios::out);

file << searchusername << endl;

file << searchpassword << endl;

file << searchtype << endl;

file << current\_balance << endl;

file << sfix\_balance<<endl;

file << jfix\_balance<<endl;

file << loan\_repayment<<endl;

file.close();

}

//display the user financial details

void balance()

{

char saving\_acc[]="S";

system ("cls");

cout<<fixed<<setprecision(2)

<< " \n Account balance "<<endl<<endl

<< " ----------------------------------------" <<endl<<endl

<< " Name of user : "<<setw(15)<<searchusername<<endl;

if(strcmp(searchtype,saving\_acc)==0)

cout<< " Acc . Type : "<<setw(15)<<"Saving"<<endl;

else

cout<< " Acc . Type : "<<setw(15)<<"Current"<<endl;

cout<< " Balance : RM "<<setw(15)<<current\_balance<<endl;

if(sfix\_balance !=0 )

cout<< " S-deposit Amt : RM "<<setw(15)<<sfix\_balance<<endl;

if(jfix\_balance!=0)

cout<< " J-deposit Amt : RM "<<setw(15)<<jfix\_balance<<endl;

if(loan\_repayment!=0)

cout<< " Loan Repayment : RM "<<setw(15)<<loan\_repayment<<endl;

cout<< " \n ----------------------------------------" <<endl<<endl;

system ("pause");

system("cls");

showMenu();

}

void payment()

{

const int size =100;

char phoneNo[size];

int payment\_choice, bill\_payment, prepaid, bill\_no, len;

string type;

double amount, loan=1000, insurance =1000;

bool flag = false;

char validation[]="0164651234";

system("cls");

//display the type of payment

cout<< "\n Payment \n\n"

" --------------------------\n\n"

" 1 - Bill Payment \n"

" 2 - Prepaid \n"

" 3 - Loan Payment \n"

" 4 - Return main menu \n\n"

" --------------------------\n\n";

cout<<" Please select your option: ";//prompt for option

cin>> payment\_choice;

switch(payment\_choice)

{

case 1:

system ("cls");

//display the types of bill payment

cout<<"\n Type of bill payment \n\n"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6\n\n"

" 1 - Electric bill \n"

" 2 - Water bill \n"

" 3 - Utar bill \n\n"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6\n\n";

cout<<"Please select your option: ";//prompt for option

while (! (cin>> bill\_payment))//validate the option entered must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (bill\_payment < 1 || bill\_payment > 3)//validate the option entered is in the range

{

cout<<"Wrong option.\n";

cout<<"Please select your option.\n";

cin>>bill\_payment;

cin.clear();

cin.ignore(100,'\n');

}

switch(bill\_payment)

{

case 1: type = "Electric bill";

break;

case 2: type = "Water bill";

break;

case 3: type = "Utar bill";

break;

}

cout << "\nEnter bill number reference: ";//prompt for bill number

cin >> bill\_no;

cout<<"Please key in transaction amount: ";//prompt for transaction amoount

while (! (cin>>amount))//validate the amount entered must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (amount>current\_balance)//check the sufficiency of amount to the transaction

{

cout<<"\nInsufficient amount of money to make payment.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

while (amount<=0)//ensure the amount entered is relevant

{

cout<<"\nAmount is not relevant.\n";

cout<<"Please key in again:";

cin>>amount;

cin.clear();

cin.ignore(100,'\n');

}

system("cls");

//display the details of the transaction

cout<<"\n\n Bill payment -Reference "<<endl<<endl;

cout<<"-----------------------------------"<<endl<<endl;

cout<<"Type of bill : "<<setw(15)<<type<<endl;

cout<<"Bill reference : "<<setw(15)<<bill\_no<<endl;

cout<<"Transaction amount: "<<setw(15)<<amount<<endl;

cout<<"-----------------------------------"<<endl<<endl;

Confirmation();

current\_balance-=amount;

cout<<"Your transaction is successful.\n\n";

system("pause");

overwrite\_file();

balance();

system("pause");

case 2:

system ("cls");

//display the types of prepaid

cout<<"\n Type of Prepaid \n\n"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6\n\n"

" 1- Digi \n"

" 2- Celcom \n"

" 3- Maxis \n"

" 4- U-mobile \n"

" 5- Red-one \n\n"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6"

"\6\6\6\6\6\6\6\6\6\6\6\6\6\6\n\n";

cout<<"Please select your option: ";//prompt for option

while (! (cin>> prepaid))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (prepaid < 1 || prepaid > 5)//validate the input is in the range

{

cout<<"Wrong option.\n";

cout<<"Key in again: ";

cin>>prepaid;

cin.clear();

cin.ignore(100,'\n');

}

switch(prepaid)

{

case 1: type = "Digi";

break;

case 2: type = "Celcom";

break;

case 3: type = "Maxis";

break;

case 4: type = "U-mobile";

break;

case 5: type = "Red-one";

break;

}

cout<<"\nPlease key in handphone number(10 numbers): ";//prompt for the handphone number

cin.ignore();

cin.getline(phoneNo, size);

len = strlen(phoneNo);

A: for( int i=0; i<len;i++)//validate all inputs must be digit

{

if(! isdigit (phoneNo[i]))

{

cout<<"Must be a number.\n";

cin.clear();

test(phoneNo, size);

goto A;

}

else flag =true;

}

if (flag)//to ensure the phone number enter is a proper format

{

if( (strlen(phoneNo)==strlen(validation)))

{

cout<<"This ia a valid number.\n";

cout<<"Please key in transaction amount: ";

}

else

{

cout<<"This is not proper format of a phone number in Malaysia.\n";

test(phoneNo, size);

goto A;

}

}

while (! (cin>>amount))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (amount>current\_balance)

{

cout<<"Insufficient amount of money to make payment.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

while (amount<=0)

{

cout<<"Amount is not relevant.\n";

cout<<"Please key in again:";

cin>>amount;

cin.clear();

cin.ignore(100,'\n');

}

system("cls");

//display the transaction details

cout<<"\n\n Prepaid top-up-Reference "<<endl<<endl;

cout<<"-----------------------------------"<<endl<<endl;

cout<<"Handphone number : "<<setfill('0')<<setw(10)<<phoneNo<<endl;

cout<<"Type of prepaid : "<<setfill(' ')<<setw(10)<<type<<endl;

cout<<"Transaction amount : RM "<<setw(10)<<fixed<<setprecision(2)<<amount<<endl;

cout<<"-----------------------------------"<<endl<<endl;

Confirmation();

current\_balance -= amount;

cout<<"Your transaction is successful.\n\n";

system("pause");

overwrite\_file();//update the user details

balance();

system("pause");

system("cls");

break;

case 3:

system ("cls");

//display loan payment details

cout<<"\n\n Loan payment "<<endl<<endl;

cout<<"--------------------------------------------"<<endl<<endl;

cout<<"Please key in transaction amount: ";//prompt the amount

while (! (cin>>amount))//validate the input must be digit

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while (amount>current\_balance)//check the sufficiency of balance to do transaction

{

cout<<"Insufficient amount of money to make payment.\n";

cout<<"Please check your balance before you do any transaction...\n";

system("pause");

showMenu();

}

while (amount<=0)

{

cout<<"Amount is not relevant.\n";

cout<<"Please key in again:";

cin>>amount;

cin.clear();

cin.ignore(100,'\n');

}

if(loan\_repayment<amount)

{

cout<<"Your payment was exceed the amount of loan repayment.\n"

<<"Please check your loan repayment before you do the payment.\n\n";

system("pause");

showMenu();

}

system("cls");

cout<<"\n\n Loan payment - Reference "<<endl<<endl;

cout<<"-----------------------------------------------"<<endl<<endl;

cout<<"Amount of repayment for loan: RM "<<setw(10)<<amount<<endl<<endl;

cout<<"Balance after transaction : RM "<<setw(10)<<current\_balance-amount<<endl<<endl;

cout<<"-----------------------------------------------"<<endl<<endl;

Confirmation();

current\_balance-=amount;

loan\_repayment-=amount;

cout<<"Your transaction is successful.\n\n";

system("pause");

overwrite\_file();//update user details

balance();

break;

case 4: showMenu();

break;

default:

error();

payment();

}

}

//to validate all the input entered must be digit

void test(char num[], int siz)

{

A:cout<<"Please key in handphone number(10 numbers): ";

cin.getline(num, siz);

}

void Planning\_assistant()

{

system("cls");

int choice;

cout<<endl<<endl

<<" Financial Calculator \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Press 1- Calculate interest for Senior-fixed deposit.\n"

<<"Press 2- Calculate interest for Junior-fixed deposit.\n"

<<"Press 3- Return main Menu.\n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Please enter your choice : ";

cin>>choice;

switch(choice)

{

case 1:

{

const double interest=1.05;

double result,amount;

int month;

system("cls");

cout<<" Calculator for Senior-fixed deposit. \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Please enter the amount of fixed deposit : RM ";

while (! (cin>> amount))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(amount<0)

{

cout<<"The amount enter must be greater than zero.\n";

cout<<"Please re-enter the amount : ";

cin>>amount;

}

cout<<endl

<<"Please enter the duration store in month : ";

while (! (cin>> month))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(month<0)

{

cout<<"The month enter must be greater than zero.\n";

cout<<"Please re-enter the month : ";

cin>>month;

}

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

result=amount;

for(int i=0;i<month;i++)

{

result\*=interest;

}

system("cls");

cout<<fixed<<setprecision(2)

<<"\n\n Result of Senior-fixed deposit. \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"The amount of fixed deposit : RM "<<amount

<<endl

<<"The total amount after "<<month<<" month : RM "

<<result<<endl

<<"Net profit : RM "<<result-amount<<endl

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

showMenu();

}

break;

case 2:

{

const double interest=1.08;

double result,amount;

int month;

system("cls");

cout<<"\n\n Calculator for Junior-fixed deposit. \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"Please enter the amount of fixed deposit : RM ";

while (! (cin>> amount))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(amount<0)

{

cout<<"The amount enter must be greater than zero.\n";

cout<<"Please re-enter the amount : ";

cin>>amount;

}

cout<<endl

<<"Please enter the duration store in month : ";

while (! (cin>> month))

{

cout<<"Must be a number. Key in again: ";

cin.clear();

cin.ignore(100,'\n');

}

while(month<0)

{

cout<<"The month enter must be greater than zero.\n";

cout<<"Please re-enter the month : ";

cin>>month;

}

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

result=amount;

for(int i=0;i<month;i++)

{

result\*=interest;

}

system("cls");

cout<<fixed<<setprecision(2)

<<" Result of Junior-fixed deposit. \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

<<"The amount of fixed deposit : RM "<<amount

<<endl

<<"The total amount after "<<month<<" month : RM "

<<result<<endl

<<"Net profit : RM "<<result-amount<<endl

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

showMenu();

}

break;

case 3:

showMenu();

break;

default :

system("cls");

error();

Planning\_assistant();

}

}

//display the output that user was entered incorrect choices

void error()

{

system("cls");

system("COLOR 0C");

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"

"Incorrect choice. Please enter again.\n\n"

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

cin.clear();

cin.ignore(100, '\n');

system("cls");

system("COLOR 0F");

}

//to exit the programme

void log\_out()

{

char log\_out1[] = "Y";

char log\_out2[] = "y" ;

char back\_to\_menu1[]="N";

char back\_to\_menu2[]="n";

char choice[10];

system("cls");

LOGOUT: cout<<"\n Logout \n\n"

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<"Are you sure you want to log out?(Y or N): ";

cin>>choice;

while(strlen(choice)!=strlen(log\_out1))

{

cout<<"\nYour choice must be \"y\" or \"n\" only. \n\n";

system("pause");

system("cls");

goto LOGOUT;

}

while(strcmp(choice,log\_out1)==0 || strcmp(choice,log\_out2)==0)

{

system("cls");

system("COLOR 0A");

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

<<"\nThanks for choosing our bank.\n\n"

<<"Have a nice day! \1 \2 \1 \n\n"

<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

system("pause");

exit(0);

}

while(strcmp(choice,back\_to\_menu1)==0 || strcmp(choice,back\_to\_menu2)==0)

{

showMenu();

}

cout<<"Your choice must be \"y\" or \"n\" only. \n";

system("pause");

system("cls");

goto LOGOUT;

}

//to confirm the relative process with the user

void Confirmation()

{

char a[30]="";

char proceed[] = "Y";

char back\_to\_menu[]="N";

cout<<"Continue to proceed this process?(Y/N) :";

cin>>a;

for(int i=0;i<30;i++)

{

a[i]=toupper(a[i]);

}

if(strcmp(a,back\_to\_menu)==0)

{

system("cls");

showMenu();

}

else if(strcmp(a,proceed)==0)

{

cout<<"";

}

else

{

cout<<"Your choice must be \"Y\" or \"N\" only.\n";

Confirmation();

}

}