#include<iostream>

#include<fstream>

#include<conio.h>

#include<time.h>

#include<cstring>

#include<windows.h>

#include <sstream>

#include<stdio.h>

using namespace std;

int admin();

int addVoter();

int updateVoter();

int updateCandidate();

int votingResult();

int reset();

int exit();

int LoginCheck(string, string);

void displaycandidatedata();

void deletecandidatedata();

void displaycandidatedata();

int main() //main login page

{

time\_t rawtime; //System time define

struct tm \* timeinfo; //predefined structure in time.h

time ( &rawtime );

timeinfo = localtime ( &rawtime );

cout << "\n\n\t\t\t\t\t\t " << asctime (timeinfo); //display time and date

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

int vote; //voter's input to vote

char ch;

string fname; //input user id

string fend = ".txt"; //for creating text file

string voted\_user; //users who have given their vote

string username, password; //saved user id and password

fstream voter\_account; //file streaming

voter\_account.open("voter\_list.txt", ios::out | ios::app); //reading registered voter id and password

voter\_account >> username >> password;

cout << "\n\t\t\t Conducted by Election Commission of India \n";

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

cout << "\t\t\t \n";

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

cout << "\t\t\t \n";

cout << "\t\t\t ONLINE VOTING SYSTEM \n";

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

cout << "\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t\t\tEnter User ID: ";

cin>>fname; //get id from user

cout << "\n\t\t\t\t\t\tEnter PIN : ";

char pass[32];//to store password.

int i = 0;

char a;//a Temp char

for(i=0;;)//infinite loop

{

a=getch();//stores char typed in a

if((a>='a'&&a<='z')||(a>='A'&&a<='Z')||(a>='0'&&a<='9'))

//check if a is numeric or alphabet

{

pass[i]=a;//stores a in pass

++i;

cout<<"\*";

}

if(a=='\b'&&i>=1)//if user typed backspace

//i should be greater than 1.

{

cout<<"\b \b";//rub the character behind the cursor.

--i;

}

if(a=='\r')//if enter is pressed

{

pass[i]='\0';//null means end of string.

break;//break the loop

}

}

if((fname=="admin") && (strcmp(pass, "1234") == 0)) //admin id and password

{

cout << "\n\n\t\t\t\t\t\tAdmin login Successful\n\t\t\t\t\t\t";

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

Sleep(1500); //pause for 1.5 sec

system("cls"); //clearing screen

admin(); // admin page

}

else if (LoginCheck(fname, pass)!=0) //checking voter-id and password

{

cout<<"\n\n\t\t\t\t\t\tVoter login Successful\n\t\t\t\t\t\t";

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

Sleep(2000); //pause for 2 sec

system("cls");

string filename = fname + fend; //creating a text file named as username

fstream file;

file.open(filename.c\_str(), ios::in); // reading voter file with full details of the voter

cout << "\n\t\t\t\t\t\t\tFull details of Voter ID " << fname << " \n";

string info;

while(!file.eof())

{

getline(file,info);

cout << "\t\t\t\t\t" << info << "\n";//display details of voter

}

fstream voter;

voter.open("voted.txt"); //opening the file where the user has already voted

int m=0;

int k=0;

while (voter >> voted\_user) //checking if the user has voted

{

m++;

if (fname != voted\_user)

k++;

}

if (m != k)

{

cout << "\n\t\tYou have already voted" << endl; //the user has already voted

system("pause");

system("cls");

exit();

}

else

{

y: //jump statement incase of invalid choice

int p;

int a;

int b;

int c;

int d;

int total;

int i=0;

fstream candidate;

string name1, name2, party;

candidate.open("candidate\_list.txt");

cout<<"\n\n\t\t\t\t\tLIST OF CANDIDATES\n\n";

while(candidate >> name1 >> name2 >> party)

{

cout <<"\n\t\t\t\t\t" << ++i << ". " << "Name: "<<name1 << " " << name2 <<" Party: "<< party <<"\n";

}

candidate.close();

cout<<"Press 0 to log out";

cout << "\n\t\t\t\t\tEnter vote "; // casting vote

cin >> vote;

if(vote==0)

{

Sleep(1000);

system("cls");

main();

}

else if(vote>0 && vote<=i)

{

ostringstream os; //converting integer to string

os << "candidate\_" << vote << ".txt"; //concatenation strings

string s = os.str();

ifstream file;

file.open(s.c\_str(), ios::app); //opening the file of the candidate

int count;

file >> count; //reading the vote count of the candidate

ofstream newfile;

newfile.open(s.c\_str());

count++; //incrementing the vote count of the candidate by 1

newfile << count; // writing back the vote count of the candidate

file.close();

newfile.close();

fstream createdaccount;

createdaccount.open("voted.txt",ios::app); //opening the file containing list of voters who have given vote

createdaccount << fname<< endl; // adding the id of voter to the list so he cannot vote again

createdaccount.close();

system("cls");

exit();

}

else

{

cout << "invalid choice. please try again";

Sleep(1500);

system("CLS");

goto y; //jump statement to the begining

}

}

}

else

{

cout << "Invalid username/password combination" << endl;

Sleep(1500);

system("CLS");

main();

}

}

int LoginCheck (string fname, string pass ) //User Login Fuction

{

ifstream file;

string username, password;

int n=0;

file.open("voter\_list.txt"); //opening the file containing list of user id and password

if (file.is\_open())

{

while (!file.eof())

{

file >> username >> password; //reading the file

n++;

if ((fname==username) && (pass==password)) //login successful if id and password matches in the list

return n;

}

}

else

{

cout << "file not open" << endl;

}

return 0;

}

int admin() //Admin fuction

{

cout << "\n\n\n\n\t\t\t\t\t\t Only for administration ";

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

cout << "\t\t\*\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| Choose from the following options: |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 1. Add New Voter |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 2. Update Voter Information |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 3. Update candidate |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 4. Voting Result |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 5. Voting reset |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*| 6. Exit the Program |\*\*\n";

cout << "\t\t\*\*| |\*\*\n";

cout << "\t\t\*\*|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\*\*\n";

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

char i;

cout << "\t\tEnter your choice: ";

cin >> i; // taking input from the user

system("cls");

switch(i)

{

case '1':

addVoter(); //function for adding new voter

break;

case '2':

updateVoter(); //function for updating details of the voter

break;

case '3':

updateCandidate(); //function for adding and deleting candidate

break;

case '4':

votingResult(); // function for displaying voting result

break;

case '5':

reset(); //function for resetting result and candidate list

break;

case '6':

exit(); // exit function

break;

default:

{

cout << "\n\n\t\t\t\t\t\tInvalid Choice\n";

cout << "\t\t\t\t\t\tTry again...........\n\n";

Sleep(1500);

system("cls");

admin();

}

}

}

int addVoter() //Create new voter aacount

{

time\_t rawtime;

struct tm \* timeinfo;

time ( &rawtime );

timeinfo = localtime ( &rawtime );

cout << "\n\n\t\t\t\t\t\t\t" << asctime (timeinfo); //display date and time in screen

ifstream file;

ofstream newuser;

string user, pass;

string fname;

string end = ".txt";

string username, password, passwordconfirm;

file.open("voter\_list.txt", ios::app); //opening the file containing the list of voters

newuser.open("voter\_list.txt", ios::app);

int age;

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

cout<<"\n\t\t\t(Press 0 to go back)";

cout<< "\n\n\t\t\tEnter Voter's age: ";

cin>>age; //get input age from the user

if(age==0)

{

Sleep(1000);

system("cls");

admin();

}

else if(age>0 && age<18) //checking whether the user is eligible to vote

{

cout<<"\n\t\t\tYou are not eligible for voting\n\t\t\t";

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

system("pause");

system("cls");

addVoter();

}

else

{

cout<<"\n\t\t\tYou are eligible for voting";

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

Sleep(1000);

system("cls");

cout<<"\n\t\t\t(Press 0 to go back)";

cout<<"\n\n\t\t\tCreate the 6 digit User ID: ";

cin >> fname;

if(fname=="0")

{

Sleep(1000);

system("cls");

addVoter();

}

char ch1;

char ch2;

string pass1 = "";

string pass2 = "";

int m=0;

int k=0;

while (file >> user >> pass)

{

m++;

if (fname!=user)

k++;

}

if (m!=k)

{

cout << "\n\t\t\tThere is already a user with this ID." << endl;

cout<< "\n\t\t\tPress 1 to try again, 0 to go back: ";

int x;

cin>>x;

switch(x)

{

case 1:

system("cls");

addVoter();

break;

case 0:

system("cls");

admin();

break;

default:

cout<<"Invalid choice, try again.";

main();

}

}

else

{

cout << "\n\t\t\tPlease create a 4 digit PIN: ";

ch1=\_getch();

while(ch1 != 13 && ch1!=4)

{//character 13 is enter

pass1.push\_back(ch1);

cout << '\*';

ch1 = \_getch();

}

cout<<"\n\t\t\tRe-enter PIN: ";

ch2=\_getch();

while(ch2 != 13 && ch2!=4)

{//character 13 is enter

pass2.push\_back(ch2);

cout << '\*';

ch2 = \_getch();

}

}

if (m==k && pass1==pass2)

{

newuser << fname << " " << pass1 <<endl; //saving the user id and password to the voter list

string filename=fname + end;

cout << "\n\t\t\tAccount created!" << endl;

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

file.close();

newuser.close();

fstream file;

file.open(filename.c\_str(), ios::out); //creating a file for getting details about the voter

time\_t rawtime;

struct tm \* timeinfo;

time ( &rawtime );

timeinfo = localtime ( &rawtime );

cout << "\n\t\t\tUpdating Voter's information on : "<<asctime (timeinfo);file<<"\n\t\t\tUpdated Voter's information' on: "<<asctime (timeinfo)<<"\n";

struct voter\_info

{

char name[20];

char fathers\_name[20];

char address[100];

char district[20];

char state[20];

char pincode[20];

char constituency[20];

char contact[10];

char age[20];

char sex[10];

char aadhar\_id[15];

};

voter\_info ak;

cout << "\t\t\t------------------------------------------- ";file<<"\t\t\t-----------------------------------------";

cout << "\n\t\t\tEnter details in block letters";

cout << "\n ";file<<" ";gets(ak.name) ;file<<ak.name<<"\n";

cout << "\n\t\t\tName : ";file << "\t\t\tName : ";gets(ak.name) ;file << ak.name<<"\n";

cout << "\n\t\t\tFather's Name : ";file << "\t\t\tFather's Name : ";gets(ak.fathers\_name);file << ak.fathers\_name<<"\n";

cout << "\n\t\t\tVillage/Town : ";file << "\t\t\tVillage/Town : ";gets(ak.address) ;file << ak.address<<"\n";

cout << "\n\t\t\tDistrict : ";file << "\t\t\tDistrict : ";gets(ak.district) ;file << ak.district<<"\n";

cout << "\n\t\t\tState : ";file << "\t\t\tState : ";gets(ak.state) ;file << ak.state<<"\n";

cout << "\n\t\t\tConstituency : ";file << "\t\t\tConstituency : ";gets(ak.constituency);file << ak.constituency<<"\n";

cout << "\n\t\t\tPinCode : ";file << "\t\t\tPinCode : ";gets(ak.pincode) ;file << ak.pincode<<"\n";

cout << "\n\t\t\tDOB(dd/mm/yyyy): ";file << "\t\t\tDOB(dd/mm/yyyy): ";gets(ak.age) ;file << ak.age<<"\n";

cout << "\n\t\t\tGender(M/F) : ";file << "\t\t\tGender : ";gets(ak.sex) ;file << ak.sex<<"\n";

cout << "\n\t\t\tAadhar ID : ";file << "\t\t\tAadhat ID : ";gets(ak.aadhar\_id) ;file << ak.aadhar\_id<<"\n";

cout << "\n\t\t\t-------------------------------------\n";file << "\n\t\t\t-----------------------------------------\n\n";

cout << "\t\t\tInformation Saved Successfully\n";

file.close();

system("pause");

system("cls");

admin();

}

else

{

cout << "\n\t\tThe passwords given do not match." << endl;

Sleep(1500);

system("cls");

addVoter();

}

}

}

int updateVoter() //Voter details Update function

{

string fname;

ifstream file;

cout<<"\n\nEnter the Voter ID to be opened : ";

cin>>fname;

system("cls");

string filename = fname + ".txt";

file.open(filename.c\_str());

if(!file)

{

cout << "\nError while opening the file\n";

updateVoter();

}

else

{

cout << "\n\n\t\t\t\t Information about Voter ID "<<fname<<"\n\n";

string info;

while(file.good())

{

getline(file,info);

cout<<info<<"\n";

}

ofstream newfile;

newfile.open(filename.c\_str());

cout << "\n";

time\_t rawtime;

struct tm \* timeinfo;

time ( &rawtime );

timeinfo = localtime ( &rawtime );

cout<<"\n\t\t\tEnter details in block letters\n";

newfile<<"\n\t\t\tVoter's information' on: "<<asctime (timeinfo);

struct voter\_info

{

char name[20];

char fathers\_name[20];

char address[100];

char district[20];

char state[20];

char pincode[20];

char constituency[20];

char contact[10];

char age[20];

char sex[10];

char aadhar\_id[15];

};

voter\_info ak;

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

newfile<<"\n\t-------------------------------------------\n";

cout << "\n ";newfile<<" ";gets(ak.name) ;newfile<<ak.name<<"\n";

cout << "\n\t\t\tName : ";newfile << "Name : ";gets(ak.name) ;newfile << ak.name<<"\n";

cout << "\n\t\t\tFather's Name : ";newfile << "Father's Name : ";gets(ak.fathers\_name);newfile << ak.fathers\_name<<"\n";

cout << "\n\t\t\tVillage/Town : ";newfile << "Village/Town : ";gets(ak.address) ;newfile << ak.address<<"\n";

cout << "\n\t\t\tDistrict : ";newfile << "District : ";gets(ak.district) ;newfile << ak.district<<"\n";

cout << "\n\t\t\tState : ";newfile << "State : ";gets(ak.state) ;newfile << ak.state<<"\n";

cout << "\n\t\t\tConstituency : ";newfile << "Constituency : ";gets(ak.constituency);newfile << ak.constituency<<"\n";

cout << "\n\t\t\tPinCode : ";newfile << "PinCode : ";gets(ak.pincode) ;newfile << ak.pincode<<"\n";

cout << "\n\t\t\tDate of birth : ";newfile << "Date of birth : ";gets(ak.age) ;newfile << ak.age<<"\n";

cout << "\n\t\t\tGender(M/F) : ";newfile << "Gender(M/F) : ";gets(ak.sex) ;newfile << ak.sex<<"\n";

cout << "\n\t\t\tAadhar ID : ";newfile << "Aadhat ID : ";gets(ak.aadhar\_id) ;newfile << ak.aadhar\_id<<"\n";

cout << "\n\t-------------------------------------------\n";newfile << "\n\t-------------------------------------------";

cout << "\t\t\tInformation Saved Successfully\n";

file.close();

newfile.close();

cout << "\n\n";

system("pause");

system("cls");

admin();

}

}

int updateCandidate() // function for Updating candidate list

{

b:

char a;

string info;

int i = 0;

int count=0;

fstream candidate;

string name1, name2, party;

candidate.open("candidate\_list.txt");

cout<<"\n\n\t\t\t\t\tLIST OF CANDIDATES\n\n";

while(candidate >> name1 >> name2 >> party)

{

cout <<"\n\t\t\t\t\t" << ++i << ". " << "Name: "<<name1 << " " << name2 <<" Party: "<< party <<"\n";

}

candidate.close();

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

cout<<"\n\n\t\t\t\t\tpress 1 to add candidate";

cout<<"\n\n\t\t\t\t\tpress 2 to delete candidate ";

cout<<"\n\n\t\t\t\t\tpress 0 to go back: ";

cin>>a;

switch(a)

{

case '0':

system("cls");

admin();

break;

case '1':

{

string name1, name2;

string party;

fstream newcandidate;

newcandidate.open("candidate\_list.txt", ios::app);

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

cout << "\t\t\t\t\tEnter Candidate First name" <<": ";

cin>>name1;

cout<< "\n\t\t\t\t\tEnter Candidate Last name" <<": ";

cin>>name2;

cout << "\n\t\t\t\t\tEnter party name" <<": ";

cin>>party;

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

cout<<"\n\t\t\t\t\tNew candidate added to the list, please wait....";

newcandidate << name1 << " " << name2 << " " << party << endl;

ostringstream os;

os << "candidate\_" << ++i << ".txt";

string s = os.str();

fstream file;

file.open(s.c\_str(), ios::out);

int count =0;

file << count;

Sleep(2500);

system("cls");

file.close();

updateCandidate();

break;

}

case '2':

{

deletecandidatedata();

Sleep(2500);

system("cls");

updateCandidate();

break;

}

default:

{

system("cls");

cout<<"incorrect choice, try again";

Sleep(100);

updateCandidate();

}

}

}

void deletecandidatedata()

{

string name1, name2, party, tname1, tname2;

int age, x=0; // x - "counter" to check if user entered wrong name

ifstream candidate("candidate\_list.txt");

ofstream temp("temp.txt"); // temp file for input of every student except the one user wants to delete

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

cout << "\n\t\t\t\t\tEnter the candidate name you want to erase from database ";

cout << "\n\n\t\t\t\t\tFirst name: ";

cin>>tname1;

cout << " \n\t\t\t\t\tLast name: ";

cin>>tname2;

//ifstream students("students.txt");

//ofstream temp("temp.txt"); // temp file for input of every student except the one user wants to delete

while(candidate >> name1 >> name2 >> party )

{

if((tname1!=name1) && (tname2 != name2)){ // if there are students with different name, input their data into temp file

temp << name1 << ' ' << name2 << ' ' << party << ' ' << endl;

}

else if((tname1==name1) && (tname2 == name2)){ // if user entered correct name, x=1 for later output message that the user data has been deleted

x=1;

}

}

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

candidate.close();

temp.close();

remove("candidate\_list.txt");

rename("temp.txt", "candidate\_list.txt");

if(x==0){ // x was set to 0 at start, so if it didn't change, it means user entered the wrong name

cout << "\n\n\t\t\t\t\tThere is no candidate with name you entered." << endl;

}

else{ // x is not 0, it means user entered the correct name, print message that students data has been deleted

cout << "\n\n\t\t\t\t\tcandidate data deleted from the list, please wait....";

}

}

int votingResult() //displaying Voting Result function

{

int vote;

string candidate;

fstream file;

fstream vote\_file;

vote\_file.open("candidate\_list.txt");

int i=0;

while(!vote\_file.eof()) //displaying candidate list

{

getline(vote\_file,candidate);

++i;

}

vote\_file.close();

int j;

for(j=1;j<=i;j++)

{

ostringstream os;

os << "candidate\_" << j << ".txt";

string s = os.str();

fstream file;

file.open(s.c\_str()); // opening all the candidate vote count

file >> vote;

cout<<"\n\t\t\t\tCandidate\_"<<j<<":"<<vote<<"votes"<<endl; // displaying the vote count of the candidate

}

system("pause");

system("cls");

admin();

}

int reset() //Voting Result Reset function

{

int vote;

string candidate;

fstream file;

file.open("candidate\_list.txt");

int i=0;

while(!file.eof())

{

getline(file,candidate);

cout<<"\n\t\t\t\t " <<++i <<". "<< candidate <<endl; //displaying candidate list

}

int j;

for(j=1;j<=i;j++)

{

ostringstream os;

os << "candidate\_" << j << ".txt";

string s = os.str();

remove(s.c\_str()); //deleting the all the candidate's file

}

ofstream ofs;

ofs.open("candidate\_list.txt", std::ofstream::out | std::ofstream::trunc); //deleting the contents of the file

ofs.close();

ofstream fs;

fs.open("voted.txt", std::ofstream::out | std::ofstream::trunc); //deleting the contents of the file

fs.close();

Sleep(1500);

system("cls");

admin();

}

int exit() //Main Exit function

{

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

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

cout<<"\t\t\t\*\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| THANK YOU FOR USING |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| ONLINE VOTING SYSTEM |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*| |\*\*\n";

cout<<"\t\t\t\*\*|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\*\*\n";

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

Sleep(1500);

system("cls");

main();

}