#include<iostream.h>

#include<ctype.h>

#include<conio.h>

#include<string.h>

#include<fstream.h>

#include<time.h>

#include<math.h>

#include<stdio.h>

#include<process.h>

#include<dos.h>

const char a=176;

const char b=219;

const char ll=179;

void tp\_lgo()

{

clrscr();

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

cout<<" "<<a<<a<<" "<<a<<" "<<a<<" "<<a<<" ";

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

cout<<" "<<a<<" "<<" "<<a<<" "<<a<<" "<<a<<" "<<a<<a<<a;

cout<<" "<<" Smart ";

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

cout<<" "<<a<<" "<<a<<a<<" "<<a<<" "<<a<<" ";

}

void std\_err1()

{

cout<<"\n Error : Filed Cannot Be Left Blank -> ";

}

void std\_err2()

{

cout<<"\n Error : Invalid Parameter -> ";

}

void std\_err3()

{

cout<<"\n Press ~ [SHIFT+`] to retain the old one \n";

}

int digit\_sqncr(int key\_var\_digit)

{

int sum=0;

int digit=0;

while(key\_var\_digit>0)

{

digit=key\_var\_digit%10;

sum+=digit;

key\_var\_digit=key\_var\_digit/10;

}

return(sum);

}

int length\_calc(char stringlen[100])

{

for(int len=0;stringlen[len]!='\0';++len);

return(len);

}

int oddmeter(int key\_var\_digit2)

{

if(key\_var\_digit2%2==0)

{

return(0);

}

else if(key\_var\_digit2%2==1)

{

return(1);

}

}

int licence\_key(char key[])

{

int validation=0;

if(key[9]!='\0')

{

return(1);

}

else if(key[9]=='\0')

{

int x;

x=key[1];

int temp\_key1=digit\_sqncr(x);

x=key[2];

int temp\_key2=digit\_sqncr(x);

if(temp\_key1==temp\_key2)

{

validation++;

}

temp\_key1=key[0];

temp\_key2=key[3];

x=temp\_key1+temp\_key2;

int check=oddmeter(x);

if(check==1)

{

validation++;

}

x=key[5];

temp\_key1=digit\_sqncr(x);

x=key[8];

temp\_key2=digit\_sqncr(x);

if(temp\_key1==temp\_key2)

{

validation++;

}

temp\_key1=key[6];

temp\_key2=key[7];

x=temp\_key1+temp\_key2;

check=oddmeter(x);

if(check==1)

{

validation++;

}

if(validation==4)

{

return(0);

}

else

{

return(1);

}

}

}

class login\_profile

{

private:

char profile\_user\_fname[20];

char profile\_user\_lname[20];

char profile\_user\_appname[20];

char login\_password[36];

public:

void getfname()

{

gets(profile\_user\_fname);

}

void getlname()

{

gets(profile\_user\_lname);

}

void getappname()

{

gets(profile\_user\_appname);

}

void getapppass()

{

while(1)

{

cin>>login\_password;

char login\_password\_confirm[36];

cout<<" Confirm Your Password : ";

cin>>login\_password\_confirm;

while(1)

{

if(strcmp(login\_password,login\_password\_confirm)==0)

{

break;

}

else

{

cout<<"\n ERROR : Password Does not Match \n";

cout<<" Confirm Your Password : ";

cin>>login\_password\_confirm;

cout<<"\n\n ! Remember your password well in case you lose it all the files associated with it will be Destroyed \n";

}

}

break;

}

}

void getdata()

{

cout<<"\n Enter Your First Name : ";

gets(profile\_user\_fname);

cout<<" Enter Your last Name : ";

gets(profile\_user\_lname);

cout<<" Enter Your Profile Name : ";

gets(profile\_user\_appname);

cout<<" Enter You Password : ";

getapppass();

}

void putdata()

{

cout<<"\n Your First Name : ";

cout<<profile\_user\_fname;

cout<<"\n Your last Name : ";

cout<<profile\_user\_lname;

cout<<"\n Your Profile Name : ";

cout<<profile\_user\_appname;

cout<<"\n You Password : ";

cout<<login\_password;

}

char\* return\_profile\_user\_fname()

{

return(profile\_user\_fname);

}

char\* return\_profile\_user\_lname()

{

return(profile\_user\_lname);

}

char\* return\_profile\_user\_appname()

{

return(profile\_user\_appname);

}

char\* return\_login\_password()

{

return(login\_password);

}

void modify()

{

char mprofile\_user\_fname[20];

char mprofile\_user\_lname[20];

char mprofile\_user\_appname[20];

char mlogin\_password[36];

std\_err3();

cout<<"\n Enter Your First Name : ";

gets(mprofile\_user\_fname);

std\_err3();

cout<<" Enter Your last Name : ";

gets(mprofile\_user\_lname);

std\_err3();

cout<<" Enter Your Profile Name : ";

gets(mprofile\_user\_appname);

std\_err3();

cout<<" Enter You Password : ";

cin>>mlogin\_password;

if(strcmp(mprofile\_user\_fname,"~")!=0)

{

strcpy(profile\_user\_fname,mprofile\_user\_fname);

}

if(strcmp(mprofile\_user\_lname,"~")!=0)

{

strcpy(profile\_user\_lname,mprofile\_user\_lname);

}

if(strcmp(mprofile\_user\_appname,"~")!=0)

{

strcpy(profile\_user\_appname,mprofile\_user\_appname);

}

if(strcmp(mlogin\_password,"~")!=0)

{

char mlogin\_password\_confirm[36];

cout<<" Confirm Your Password : ";

cin>>mlogin\_password\_confirm;

while(1)

{

if(strcmp(mlogin\_password,mlogin\_password\_confirm)==0)

{

break;

}

else

{

cout<<"\n ERROR : Password Does not Match \n";

cout<<" Confirm Your Password : ";

cin>>mlogin\_password\_confirm;

cout<<"\n\n ! Remember your password well in case you lose it all the files associated with it will be Destroyed \n";

}

}

strcpy(login\_password,mlogin\_password);

}

}

};

class server\_security

{

private:

char password[40];

char security\_question[50];

char security\_ans[20];

public:

void getdata()

{

cout<<"\n Enter the Password : ";

cin>>password;

cout<<" Enter The Security Question : ";

gets(security\_question);

cout<<" Enter the answer of Security Question : ";

gets(security\_ans);

}

void putdata()

{ cout<<"\n The Password : ";

cout<<password;

cout<<"\n The Security Question : ";

cout<<security\_question;

cout<<"\n Enter the answer of Security Question : ";

cout<<security\_ans;

}

void modify()

{

char mpassword[40]=" ";

char msecurity\_question[50]=" ";

char msecurity\_ans[20]=" ";

std\_err3();

cout<<" Enter the Password : ";

cin>>mpassword;

std\_err3();

cout<<" Enter The Security Question : ";

gets(msecurity\_question);

std\_err3();

cout<<" Enter the answer of Security Question : ";

gets(msecurity\_ans);

if(strcmp(mpassword,"~")!=0)

{

strcpy(password,mpassword);

}

if(strcmp(msecurity\_question,"~")!=0)

{

strcpy(security\_question,msecurity\_question);

}

if(strcmp(msecurity\_ans,"~")!=0)

{

strcpy(security\_ans,msecurity\_ans);

}

}

};

class server\_login : public server\_security

{

private:

char protocol[10];

char site\_name[15];

char id[36];

char domain[10];

public:

void getdata()

{

cout<<"\n Enter the Transfer Protocol : ";

cin>>protocol;

cout<<" Enter the Site Name : ";

gets(site\_name);

cout<<" Enter the Domain : ";

cin>>domain;

cout<<" Enter the Login ID ( complete ): ";

gets(id);

server\_security :: getdata();

}

void putdata()

{

cout<<"\n The Transfer Protocol : ";

cout<<protocol;

cout<<"\n The Site Name : ";

cout<<site\_name;

cout<<"\n The Domain : ";

cout<<domain;

cout<<"\n Login ID : ";

cout<<id;

server\_security :: putdata();

}

char\* return\_id()

{

return(id);

}

char\* return\_sitename()

{

return(site\_name);

}

void modify()

{

char mprotocol[10]=" ";

char msite\_name[15]=" ";

char mid[36]=" ";

char mdomain[10]=" ";

std\_err3();

cout<<" Enter the new Transfer Protocol : ";

cin>>mprotocol;

std\_err3();

cout<<" Enter the new Site Name : ";

gets(msite\_name);

std\_err3();

cout<<" Enter the new Domain : ";

cin>>mdomain;

std\_err3();

cout<<" Enter the new Login ID ( complete ): ";

gets(mid);

if(strcmp(mprotocol,"~")!=0)

{

strcpy(protocol,mprotocol);

}

if(strcmp(msite\_name,"~")!=0)

{

strcpy(site\_name,msite\_name);

}

if(strcmp(mdomain,"~")!=0)

{

strcpy(domain,mdomain);

}

if(strcmp(mid,"~")!=0)

{

strcpy(id,mid);

}

server\_security :: modify();

}

};

class user : public server\_login

{

private:

char f\_name[20];

char l\_name[20];

public:

void getdata()

{

cout<<"\n Enter The First Name of User : ";

gets(f\_name);

cout<<" Enter The Last Name of the User : ";

gets(l\_name);

server\_login :: getdata();

}

void putdata()

{

cout<<"\n The First Name of the User is : ";

cout<<f\_name;

cout<<"\n The Last Name of the User is : ";

cout<<l\_name;

server\_login :: putdata();

}

void modify()

{

char mf\_name[20]=" ";

char ml\_name[20]=" ";

std\_err3();

cout<<" Enter the New First Name of User : ";

gets(mf\_name);

std\_err3();

cout<<" Enter the New Last Name of the User : ";

gets(ml\_name);

if(strcmp(mf\_name,"~")!=0)

{

strcpy(f\_name,mf\_name);

}

if(strcmp(ml\_name,"~")!=0)

{

strcpy(l\_name,ml\_name);

}

server\_login :: modify();

}

};

int logix(int logix\_low,unsigned char logix\_in,int logix\_high)

{

unsigned int logix\_in\_conver=logix\_in;

if((logix\_in\_conver>=logix\_low)&&(logix\_in\_conver<=logix\_high))

{

return(0);

}

else

{

return(1);

}

}

void encom\_lgo()

{

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

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

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

cout<<b<<b<<b<<b<<" "<<b<<b<<b<<b<<b<<" "<<b<<b<<" "<<b<<b;

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

cout<<b<<" "<<" "<<b<<" "<<b<<" "<<b<<" "<<b<<" "<<b;

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

cout<<b<<b<<b<<b<<" "<<b<<b<<b<<b<<b<<" "<<b<<" "<<b;

}

void ln\_drw(int a)

{

if(a==1)

{

for(int i=0;i<80;++i)

{

cout<<"=";

}

}

else

{ for(int i=0;i<80;++i)

{

cout<<"\_";

}

}

}

void mrgn\_sqncr(int spc,int activate\_line)

{

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

{

cout<<"\t";

}

if(activate\_line==0)

{

cout<<ll;

}

}

void hedr\_tle1()

{

cout<<"\t\t\t Your Smart Password Manager \n";

}

void hedr\_tle2()

{

cout<<"\n\t\t\t Terminal Window\n";

}

void devlprs()

{

cout<<"\n";

cout<<(char)1;

cout<<" Encom Vers 0.01 \n Devloped by Harshit Yadav\n\n";

}

void fwrd\_msg()

{

cout<<"\n PRESS ANY KEY TO CONTINUE :\>";

}

int dsp\_gui1()

{

unsigned char mnu\_1;

int mnu\_flag=0;

cout<<"\n";

mrgn\_sqncr(6,0);

cout<<" Select An Choice to continue :";

mrgn\_sqncr(6,0);

cout<<" 1. User Profile\n ";

mrgn\_sqncr(6,0);

cout<<" 2. Enter Data\n";

mrgn\_sqncr(6,0);

cout<<" 3. Saved Data\n";

mrgn\_sqncr(6,0);

cout<<" 4. Time Pass\n ";

mrgn\_sqncr(6,0);

cout<<" 5. Control Panel\n ";

mrgn\_sqncr(6,0);

cout<<" 6. About us\n ";

mrgn\_sqncr(6,0);

cout<<" 0. Exit\n\n ";

mrgn\_sqncr(6,1);

cout<<" --> ";

cin>>mnu\_1;

mnu\_flag=logix(48,mnu\_1,54);

while(mnu\_flag)

{

mrgn\_sqncr(5,1);

std\_err2();

cout<<" --> ";

cin>>mnu\_1;

mnu\_flag=logix(48,mnu\_1,54);

}

int rt\_disp\_guide=mnu\_1;

return(rt\_disp\_guide);

}

void login\_time()

{

time\_t rawtime;

struct tm \* timeinfo;

time (&rawtime);

timeinfo = localtime (&rawtime);

cout<<asctime(timeinfo);

}

void profile\_creator()

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(2,2);

cout<<"SIGN UP TERMINAL \n";

ln\_drw(1);

char case\_49\_loop;

int case\_49\_loop\_var1=0;

login\_profile gamma;

cout<<"\n Enter Your First Name : ";

gamma.getfname();

cout<<" Enter Your last Name : ";

gamma.getlname();

cout<<" Enter Your Profile Name : ";

gamma.getappname();

cout<<" Enter You Password : ";

gamma.getapppass();

cout<<"\n \n Select From Below : ";

cout<<"\n 7. Submit ";

cout<<"\n 8. Retry \n--> ";

cin>>case\_49\_loop;

case\_49\_loop\_var1=logix(55,case\_49\_loop,56);

while(case\_49\_loop\_var1)

{

std\_err2();

cin>>case\_49\_loop;

case\_49\_loop\_var1=logix(55,case\_49\_loop,56);

}

switch(case\_49\_loop)

{

case 55:

{

ofstream pif;

pif.open("EncomLog.exe",ios::out);

pif.write((char\*)&gamma,sizeof(login\_profile));

pif.close();

cout<<"\n Message : File Sucessfully Written ";

break;

}

case 56:

{

profile\_creator();

}

}

}

void tron();

void password(char lock[])

{

char password[40];

int value;

gotoxy(26,12);

cout<<"Enter your password : ";

cin>>password;

value=strcmp(lock,password);

clrscr();

if(value!=0)

{

gotoxy(33,12);

cout<<"INVALID PASSWORD ";

delay(700);

gotoxy(26,14);

cout<<" :> Restarting Application <: ";

delay(1000);

tron();

}

}

void tron()

{

int tron\_1=0;

int tron\_2=0;

int tron\_3=0;

clrscr();

cout<<"\n MESSAGE : Activating Tron (Runtime Application Gaurdian)";

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

delay(1000);

cout<<"\nMESSAGE : TRON STATUS REPORT::\n\n";

cout<<"\n Diagnostics Complete at : \n\n\n\t\t\t";

login\_time();

cout<<"\n";

cout<<"\n Request : System File [";

ofstream file\_check1;

file\_check1.open("EncomQ.exe",ios::nocreate);

if(!file\_check1)

{

cout<<" >\_\_< ";

tron\_1=1;

}

else

{

cout<<" VALID ";

tron\_1=2;

}

cout<<"]\n";

file\_check1.close();

delay(500);

cout<<" Request : User File [";

ifstream file\_check2;

file\_check2.open("EncomLog.exe",ios::binary|ios::in);

if(!file\_check2)

{

cout<<" NOT FOUND ";

tron\_2=2;

}

else

{

cout<<" FOUND ";

tron\_2=3;

}

file\_check2.close();

cout<<"]\n";

delay(500);

cout<<" Request : Vault File [";

ofstream file\_check3;

file\_check3.open("vault.exe",ios::noreplace);

if(!file\_check3)

{

cout<<" FOUND ";

}

else

{

cout<<" NOT FOUND ";

tron\_3=3;

}

cout<<"]\n";

file\_check3.close();

fwrd\_msg();

getch();

if(tron\_1==1)

{

clrscr();

gotoxy(17,12);

cout<<"ENCOM : System File Invaild Appliction Will Terminate ";

gotoxy(28,17);

fwrd\_msg();

getch();

exit(1);

}

if(tron\_2==3)

{

ifstream file\_check2;

file\_check2.open("EncomLog.exe",ios::binary|ios::in);

login\_profile logalpha;

file\_check2.read((char\*)&logalpha,sizeof(login\_profile));

clrscr();

gotoxy(26,10);

cout<<"User Profile Name : "<<logalpha.return\_profile\_user\_appname();

password(logalpha.return\_login\_password());

file\_check2.close();

}

if(tron\_2==2)

{

clrscr();

gotoxy(1,12);

cout<<"USER FILE NOT FOUND :CREATE NEW PROFILE ( Previous Record File will be deleted )";

gotoxy(22,15);

cout<<"Press 1 To Confirm Else Press Any Key To Exit ";

int rom;

gotoxy(40,16);

cin>>rom;

if(rom==1)

{

gotoxy(28,18);

fwrd\_msg();

getch();

remove("vault.exe");

profile\_creator();

}

else

{

exit(0);

}

}

if(tron\_3==3)

{

clrscr();

gotoxy(13,12);

cout<<" APPLICATION FILE MISSING: Previous Record File Not Found ";

gotoxy(28,17);

fwrd\_msg();

getch();

}

}

int intelligence(char intellarr[])

{

int intelsize=length\_calc(intellarr);

long int intlsum=0;

if(stricmp(intellarr,"exit")==0)

{

return(0);

}

else

{

for(int intllloop=0;intllloop<intelsize;++intllloop)

{

intlsum+=intellarr[intllloop];

}

int intl\_para;

intl\_para=oddmeter(intlsum);

switch(intl\_para)

{

case 0:

{

cout<<"\n Encom : Yes ";

break;

}

case 1:

{

cout<<"\n Encom : No ";

break;

}

}

return(1);

}

}

void main()

{

clrscr();

encom\_lgo();

cout<<"\n\n";

ln\_drw(2);

hedr\_tle1();

devlprs();

fwrd\_msg();

getch();

clrscr();

tron();

char start='n';

do

{

clrscr();

tp\_lgo();

cout<<"\n";

ln\_drw(2);

int mnu\_switch\_guide=dsp\_gui1();

switch(mnu\_switch\_guide)

{

case 49:

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"USER PROFILE \n";

ln\_drw(1);

fstream file\_check2;

file\_check2.open("EncomLog.exe",ios::nocreate|ios::binary|ios::in|ios::out);

login\_profile logalpha;

long upos=0;

file\_check2.read((char\*)&logalpha,sizeof(login\_profile));

logalpha.putdata();

cout<<"\n \n Select From Below : ";

cout<<"\n 7. Edit ";

cout<<"\n 8. Main Menu \n--> ";

char ul;

cin>>ul;

int ulc=logix(55,ul,56);

while(ulc)

{

std\_err2();

cin>>ul;

ulc=logix(55,ul,56);

}

switch(ul)

{

case 55:

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"EDIT OPTION \n";

ln\_drw(1);

logalpha.modify();

cout<<"\n File Modified ";

file\_check2.seekg(upos);

file\_check2.write((char\*)&logalpha,sizeof(login\_profile));

cout<<"\n File Written ";

break;

}

case 56:

{

break;

}

}

file\_check2.close();

break;

}

case 50:

{

clrscr();

user alpha;

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"ENTER YOUR DETAILS \n";

ln\_drw(1);

cout<<"\n Object Created of size : ";

cout<<sizeof(alpha);

cout<<" Bytes ";

alpha.getdata();

getch();

cout<<" \n\n ";

cout<<"\n Writing into the memory ";

ofstream fout;

fout.open("vault.exe",ios::out|ios::app|ios::binary);

fout.write((char\*)&alpha,sizeof(alpha));

fout.close();

cout<<"\n Record Sucessfully Writen into File ";

getch();

break;

}

case 51:

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"RETRIVING DATA \n";

ln\_drw(1);

cout<<"\n\nRecords : ";

user retrive[6];

int ocr=0;

ifstream retro;

retro.open("vault.exe",ios::binary|ios::in|ios::nocreate);

if(!retro)

{

cout<<"\n No data in File to Display ! ";

getch();

break;

}

for(int r=1;!retro.eof();++r)

{

retro.read((char\*)&retrive[r],sizeof(user));

if(!retro.eof()==0)

{

break;

}

ocr=1;

cout<<"\n"<<" "<<r<<". "<<retrive[r].return\_id()<<" ( "<<retrive[r].return\_sitename()<<" )";

if(r==5)

{

cout<<"\n Want to Retrive more Data (y/n) ";

char sdk;

cin>>sdk;

if(sdk=='y')

{

r=1;

clrscr();

}

if(sdk=='n')

{

break;

}

}

}

if(ocr==1)

{

retro.close();

cout<<"\n\n Choose The Site Number whose data you want retrive : ";

r-=1;

int sdk\_no;

cin>>sdk\_no;

while(1)

{

if(sdk\_no<=r)

{

break;

}

else

{

std\_err2();

cin>>sdk\_no;

}

}

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<retrive[sdk\_no].return\_sitename()<<"\n";

ln\_drw(1);

retrive[sdk\_no].putdata();

cout<<"\n \n Select From Below : ";

cout<<"\n 7. Edit ";

cout<<"\n 8. Delete ";

cout<<"\n 9. Main Menu \n--> ";

char retroo;

cin>>retroo;

int retrool=logix(55,retroo,57);

while(retrool)

{

std\_err2();

cin>>retroo;

retrool=logix(55,retroo,57);

}

switch(retroo)

{

case 55:

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"EDIT OPTION \n";

ln\_drw(1);

user retrive\_e;

fstream retr("vault.exe",ios::in|ios::out|ios::binary);

long rpos=0;

while(!retr.eof())

{

rpos=retr.tellg();

retr.read((char\*)&retrive\_e,sizeof(user));

{

if((strcmp(retrive\_e.return\_id(),retrive[sdk\_no].return\_id())==0)&&(strcmp(retrive\_e.return\_sitename(),retrive[sdk\_no].return\_sitename())==0));

{

retrive\_e.modify();

cout<<"\n File Modified ";

retr.seekg(rpos);

retr.write((char\*)&retrive\_e,sizeof(user));

cout<<"\n File Written ";

break;

}

}

}

retr.close();

cout<<"\n Message : File Sucessfully Written ";

break;

}

case 56:

{

user retrive\_e;

clrscr();

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"DELETE OPTION \n";

ln\_drw(1);

char del='n';

fstream finc6("vault.exe",ios::in|ios::out|ios::binary);

fstream finc7("temp.exe",ios::in|ios::out|ios::binary);

while(!finc6.eof())

{

finc6.read((char\*)&retrive\_e,sizeof(user));

if(!finc6.eof()==0)

{

break;

}

if(strcmp(retrive\_e.return\_id(),retrive[sdk\_no].return\_id())==0)

{

retrive\_e.putdata();

cout<<"\n Are You Sure You Want to Delete Data (y/n) : ";

cin>>del;

if(del=='n')

{

finc7.write((char\*)&retrive\_e,sizeof(user));

}

if(del=='y')

{

cout<<"\n Record Sucessfully Deleted : ";

}

}

else

{

finc7.write((char\*)&retrive\_e,sizeof(user));

}

}

finc6.close();

finc7.close();

remove("vault.exe");

rename("temp.exe","vault.exe");

break;

}

case 57:

{

break;

}

}

cout<<"\n Press any key to continue ";

getch();

}

else

{

cout<<"\n\n File Empty Nothing To Display ! ";

}

break;

}

case 52:

{

clrscr();

int intl\_loopask=1;

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"TIME PASS \n";

ln\_drw(1);

cout<<"\n MESSAGE : All the Reply By The System will be Yes Or No :";

cout<<"\n {Type exit/EXIT to exit this Menu}";

while(intl\_loopask==1)

{

cout<<"\n Encom : Enter the Question you want ask ";

cout<<"\n User : ";

char intlques[100];

gets(intlques);

intl\_loopask=intelligence(intlques);

}

getch();

break;

}

case 53:

{

clrscr();

tp\_lgo();

cout<<"\n";

ln\_drw(1);

mrgn\_sqncr(4,2);

cout<<"CONTROL PANEL \n";

ln\_drw(1);

cout<<"\n \n Select From Below : ";

cout<<"\n 6. Corrupt APPLICATION ";

cout<<"\n 7. Delete User Profile ( Saved Data associated with it will also be deleted ) ";

cout<<"\n 8. Delete File Containing Saved File ";

cout<<"\n 9. Exit \n -->";

int sos=0;

cin>>sos;

if(sos==9)

{

break;

}

cout<<"\n Enter APPLICATION KEY TO CARRY OUT ANY OF THE FUNCTION ";

char appkey[9];

cout<<"\n (ie XXXX-XXXX )\n KEY == ";

cin>>appkey;

int h=licence\_key(appkey);

if(h==0)

{

getch();

clrscr();

switch(sos)

{

case 6:

{

remove("EncomQ.exe");

gotoxy(26,14);

cout<<" :> Event Sucessfully Completed <: ";

gotoxy(26,16);

cout<<" :> Shutting Down Application <: ";

delay(1000);

exit(1);

}

case 7:

{

remove("EncomLog.exe");

gotoxy(26,14);

cout<<" :> Event Sucessfully Completed <: ";

gotoxy(26,16);

cout<<" :> Shutting Down Application <: ";

delay(1000);

exit(1);

}

case 8:

{

remove("vault.exe");

gotoxy(26,14);

cout<<" :> Event Sucessfully Completed <: ";

gotoxy(26,16);

cout<<" :> Shutting Down Application <: ";

delay(1000);

exit(1);

}

default:

{

gotoxy(26,14);

cout<<" :> Event Failed (Invalid Parameter) <: ";

gotoxy(26,16);

cout<<" :> Shutting Down Application <: ";

delay(1000);

exit(1);

}

}

}

else

{

cout<<"Invalid Key ";

}

break;

}

case 54:

{

clrscr();

ln\_drw(1);

mrgn\_sqncr(5,2);

cout<<"ABOUT \n";

ln\_drw(1);

encom\_lgo();

cout<<"\n\n";

hedr\_tle1();

devlprs();

cout<<" Email ID : harshityadav95@gmail.com ";

cout<<"\n Visit : http:\\actroidnotex.blogspot.in \n\n";

fwrd\_msg();

getch();

break;

}

case 48:

{

exit(1);

}

}

cout<<"\n\n\nExit To Main Menu (y/n) : ";

cin>>start;

}while(start=='y');

getch();

}