OVERVIEW

* C++ is an object oriented language based on the C programming language.
* It can be viewed as a super set of C, almost all of the features and constructs available in C are available in C++. However C++ is more than just an extension of C. Its additional features support the programming style known as Object Oriented Programming (OOP).
* Several features that are already available in C, such as Input and output may be implemented differently in C++.
* In C++ you may use the conventional C Input and Output routines or you may use Object Oriented Input and Output by using I/O stream library.
* Its additional features support the programming style known as Object Oriented Programming (OOP).
* Object oriented programming is a programming style that is associated with the concept of Class, Objects and various other concepts revolving around these two, like Inheritance, Polymorphism, Data Abstraction, Encapsulation etc.

Features of OOP:

* Encapsulation is the process of binding up of data and function into a single entity
* Abstraction is the process of representing only the essential features hiding the background details.
* Inheritance is the capability of one class to inherit the properties from another class.
* Polymorphism is the characteristic of being able to assign a different meaning or usage to something in different contexts.

ABSTRACT

* The project provides a user friendly way of sending and receiving messages.
* The need for the project is to communicate with other users.
* With the growing stress in human beings all around the world, people very rarely find time for themselves to talk.
* Thus, this program provides a source of fun communication.
* This program is an excellent way to chat with people to communicate real-time with everyone and keep record of the conversations.

**REQUIREMENTS**

**HARDWARE REQUIREMENT:**

* Computer
* 1.50Ghz processor
* Hard disk
* 30+MB ROM Compact Disk(optional)
* RAM-512MB
* Memory Requirement:1MB
* Processor Speed:1.2GHz
* Resolution:1336x768

**SOFTWARE REQUIREMENT:**

* Turbo C++ Version 3.0/Dos box version 0.77
* Notepad
* Windows 7 or 8

**INSTALLATION OF APPLICATION:**

* Insert the ROM CD which contains the application into the CPU.
* Scan the CD for malicious objects.
* Copy the contents in the CD to your system.
* Now, you can run the application

HEADER FILES

* #include<fstream.h> -This is used for Input Output Operations.
* #include<conio.h> -This is used to provide console Input and Output.
* #include<string.h> -This is used to manipulate string or character array.
* #include<stdio.h> -This is used to provide Standard Input Output functions.

CLASSES AND OBJECTS:

1. **class account:**

Data Members: id[20], fn[10], ln[10], pass[20], gen[10], coun[10], loc[10], pno[10], sq[50], sa[20].

Member Functions: password(), input(), gender(), output(), ret\_fn(), ret\_id(), ret\_pass(), ret\_sq(), ret\_sa().

**2. class message:public mails**

Data Members: char from[10], to[10], sub[10], msg[500].

Member Functions: input1(), output1(), ret\_from(), ret\_to().

FUNCTIONS

* void signup() - To create accounts and enter details.
* void changepass() - To reset password.
* int countmsg() - To count the new messages and display in the home page of the account.
* void compose() - To compose mails and deliver them to the given to address.
* void inbox() - To check the mails the user had received and keep track of conversations.
* void outbox() - To check the mails the user had sent and keep track of conversations.
* void searchinbox() - To search inbox and display messages from a particular account.
* void searchoutbox() - To search outbox and display messages to a particular account
* void delacc() - To delete the user’s account.
* void home() - Home page of the account with a menu filled with various options.
* void signin() - To create login page for the user.

SOURCE CODE:

#include<fstream.h>

#include<conio.h>

#include<string.h>

#include<stdio.h>

#include<graphics.h>

char eid[20],ufn[20],p[20];

class account

{

char id[20],fn[10],ln[10],pass[20],gen[10],coun[10],loc[10],pno[10],sq[50],sa[20];

int dd,mm,yy,ch;

public:

void password();

void input()

{

clrscr();

cout<<"\n First Name:";

cin>>fn;

cout<<"\n Last Name:";

cin>>ln;

cout<<"\n Date of Birth:";

cin>>dd>>mm>>yy;

gender();

cout<<"\n Location:";

cin>>loc;

cout<<"\n Country:";

cin>>coun;

cout<<"\n Phone Number:";

cin>>pno;

clrscr();

cout<<"\n Email ID:";

cin>>id;

password();

cout<<"\n Security Question:";

gets(sq);

cout<<"\n Security Answer (1 word):";

cin>>sa;

}

void gender()

{

cout<<"\n Gender:"<<"\t 1.Male 2.Female 3.Others";

cout<<"\n Enter:";

cin>>ch;

if(ch==1)

strcpy(gen,"Male");

else if(ch==2)

strcpy(gen,"Female");

else if(ch==3)

strcpy(gen,"Others");

}

void output()

{

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

cout<<"\n Last Name:"<<ln;

cout<<"\n Date of Birth:"<<dd<<"."<<mm<<"."<<yy;

cout<<"\n Password:"<<pass;

cout<<"\n Gender:"<<gen;

cout<<"\n Nationality:"<<coun;

cout<<"\n Email ID:"<<id;

}

char\* ret\_fn()

{

return fn;

}

char\* ret\_id()

{

return id;

}

char\* ret\_pass()

{

return pass;

}

char\* ret\_sq()

{

return sq;

}

char\* ret\_sa()

{

return sa;

}

};

class message:public account

{

char from[10],to[10],sub[10],msg[500];

public:

void input1()

{

strcpy(from,eid);

cout<<"\n From:"<<from;

cout<<"\n To:";

cin>>to;

cout<<"\n Subject:";

gets(sub);

cout<<"\n Message:";

gets(msg);

}

void output1()

{

cout<<"\n From:"<<from;

cout<<"\n To:"<<to;

cout<<"\n Subject:"<<sub;

cout<<"\n Message:"<<msg;

cout<<"\n\n";

}

char\* ret\_from()

{

return from;

}

char\* ret\_to()

{

return to;

}

}s;

void account::password()

{

a:

int k=0;

char p[20],p1[20],p2[20],c=' ';

cout<<"\n Enter Password:";

while(c!=char(13))

{

c=getch();

if(c!=char(13))

{

if(c!='\b')

{

p1[k]=c;

cout<<"\*";

k++;

}

else

k--;

}

}

p1[k]='\0';

strcpy(pass,p1);

k=0;

c=' ';

cout<<"\n Confirm Password:";

while(c!=char(13))

{

c=getch();

if(c!=char(13))

{

if(c!='\b')

{

p2[k]=c;

cout<<"\*";

k++;

}

else

k--;

}

}

p2[k]='\0';

strcpy(p,p2);

if(strcmp(pass,p)==0)

cout<<"\n Password Confirmed!";

else

{

cout<<"\n Passwords don't match! Try again";

getch();

clrscr();

goto a;

}

}

void signup()

{

cout<<"\n NEW ACCOUNT";

s.input();

ofstream fo("email.dat",ios::binary|ios::app);

fo.write((char\*)&s,sizeof(s));

fo.close();

}

void changepass()

{

a:

int k=0;

char c,p[20];

fstream f("email.dat",ios::binary|ios::in|ios::out|ios::app);

cout<<"\n Enter your current password:";

while(c!=char(13))

{

c=getch();

if(c!=char(13))

{

if(c!='\b')

{

p[k]=c;

cout<<"\*";

k++;

}

else

k--;

}

}

p[k]='\0';

if(strcmp(::p,p)==0)

{

while(f.read((char\*)&s,sizeof(s)))

{

if(strcpy(eid,s.ret\_id())==0)

{

s.password();

f.seekg(f.tellg()-sizeof(s));

f.write((char\*)&s,sizeof(s));

cout<<"\n Password changed! \n";

break;

}

}

}

cout<<"\n Password Incorrect!";

cout<<"\n Press ENTER to retry.";

cout<<"\n Press BACKSPACE to exit.";

c=getch();

if(c=='\b')

{

f.close();

clrscr();

return;

}

else if(c==char(13))

{

clrscr();

f.close();

goto a;

}

f.close();

}

int countmsg()

{

ifstream fi("server.dat",ios::binary);

int c=0;

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_to())==0)

c++;

}

fi.close();

return c;

}

void compose()

{

clrscr();

ofstream fo("server.dat",ios::binary|ios::app);

s.input1();

fo.write((char\*)&s,sizeof(s));

fo.close();

}

void inbox()

{

clrscr();

ifstream fi("server.dat",ios::binary);

char c;

while(!fi)

{

cout<<"\n No mails received";

return;

}

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_to())==0)

s.output1();

}

cout<<"\n Press ENTER to retry";

cout<<"\n Press BACKSPACE to exit.";

c=getch();

if(c==char(13))

compose();

fi.close();

}

void outbox()

{

ifstream fi("server.dat",ios::binary);

while(!fi)

{

cout<<"\n No mails found!";

return;

}

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_from())==0)

s.output1();

}

fi.close();

}

void searchinbox()

{

ifstream fi("server.dat",ios::binary);

char t[20],c;

cout<<"\n SENDER ID:";

cin>>t;

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_to())==0 && strcmp(t,s.ret\_from())==0)

s.output1();

}

cout<<"\n Press ENTER to reply.";

cout<<"\n Press BACKSPACE to exit.";

c=getch();

if(c==char(13))

compose();

fi.close();

}

void searchoutbox()

{

ifstream fi("server.dat",ios::binary);

char t[20];

cout<<"\n Receiver ID:";

cin>>t;

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_from())==0 && strcmp(t,s.ret\_to())==0)

s.output1();

}

}

void delacc()

{

ifstream fi("email.dat",ios::binary);

ofstream fo("temp.dat",ios::binary);

a:

clrscr();

int k=0;

char p[20],c=' ',r;

cout<<"\n Enter Password:";

while(c!=char(13))

{

c=getch();

if(c!=char(13))

{

if(c!='\b')

{

p[k]=c;

cout<<"\*";

k++;

}

else

k--;

}

}

p[k]='\0';

if(strcmp(::p,p)==0)

{

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(eid,s.ret\_id())==0)

cout<<"\n Your Account has been deleted!";

else

fo.write((char\*)&s,sizeof(s));

}

remove("email.dat");

rename("temp.dat","email.dat");

fo.close();

return;

}

cout<<"\n Password Incorrect!";

cout<<"\n Press ENTER to retry.";

cout<<"\n Press BACKSPACE to exit.";

r=getch();

if(r==char(13))

goto a;

else if(r=='\b')

return;

remove("email.dat");

rename("temp.dat","email.dat");

fo.close();

}

void home()

{

clrscr();

int flag=0;

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

cout<<"\n You have "<<countmsg()<<" new messages!";

int ch;

do

{

cout<<"\n1.Inbox \n2.Compose \n3.Sent \n4.Search Inbox \n5.Search Outbox \n6.Change Password \n7.Delete Account \n8.Logout";

cout<<"\n Enter your choice:";

cin>>ch;

switch(ch)

{

case 1:inbox();

break;

case 2:compose();

break;

case 3:outbox();

break;

case 4:searchinbox();

break;

case 5:searchoutbox();

break;

case 6:changepass();

break;

case 7:delacc();

flag=1;

break;

case 8:clrscr();

strcpy(eid,NULL);

strcpy(::p,NULL);

flag=1;

getch();

clrscr();

break;

default:cout<<"\n Enter the right choice!";

}

}while(flag==1);

}

void signin()

{

a:

int k=0;

char c=' ',ch,id[20],p[20];

ifstream fi("email.dat",ios::binary);

clrscr();

gotoxy(32,32);

cout<<"\n SIGN IN";

gotoxy(32,33);

cout<<"\n Email ID:";

cin>>id;

gotoxy(32,34);

cout<<"\n Password:";

while(c!=char(13))

{

c=getch();

if(c!=char(13))

{

if(c!='\b')

{

p[k]=c;

cout<<"\*";

k++;

}

else

k--;

}

}

p[k]='\0';

while(fi.read((char\*)&s,sizeof(s)))

{

if(strcmp(id,s.ret\_id())==0 && strcmp(p,s.ret\_pass())==0)

{

strcpy(ufn,s.ret\_fn());

strcpy(eid,id);

strcpy(::p,p);

home();

return;

}

}

cout<<"\n Email ID or password incorrect!";

cout<<"\n Press ENTER to retry.";

cout<<"\n Press BACKSPACE to exit.";

c=getch();

if(c=='\b')

{

fi.close();

clrscr();

return;

}

else if(c==char(13))

{

clrscr();

fi.close();

goto a;

}

}

void main()

{

int gd=DETECT,gm;

initgraph(&gd,&gm,"c:\\tc\\bgi");

setcolor(RED);

settextstyle(0,HORIZ\_DIR,7);

outtextxy(200,210,"E");

setcolor(WHITE);

settextstyle(0,HORIZ\_DIR,5);

delay(2000);

outtextxy(260,225,"MAIL");

setfillstyle(SOLID\_FILL,DARKGRAY);

bar(200,280,440,290);

char s[2];

for(int i=200;i<=440;i+=1)

{

setfillstyle(SOLID\_FILL,LIGHTGRAY);

bar(200,280,i,290);

setcolor(WHITE);

settextstyle(0,HORIZ\_DIR,1.5);

outtextxy(200,270,"Loading...");

delay(20);

}

setcolor(WHITE);

closegraph();

int ch;

remove("email.dat");

remove("server.dat");

do

{

cout<<"\n\n1.SIGN UP \n2.SIGN IN \n3.EXIT";

cout<<"\n Enter your choice:";

cin>>ch;

switch(ch)

{

case 1:signup();

break;

case 2:signin();

break;

case 3:cout<<"\n Thank You!";

break;

}

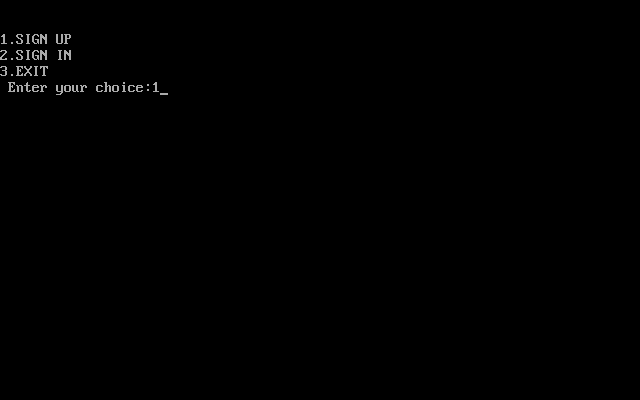
}while(ch!=3);

getch();

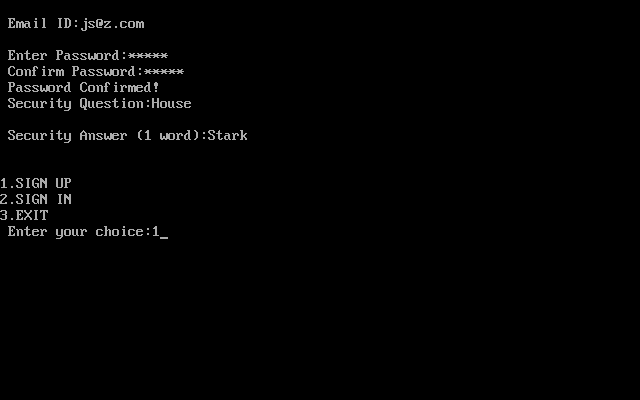
}

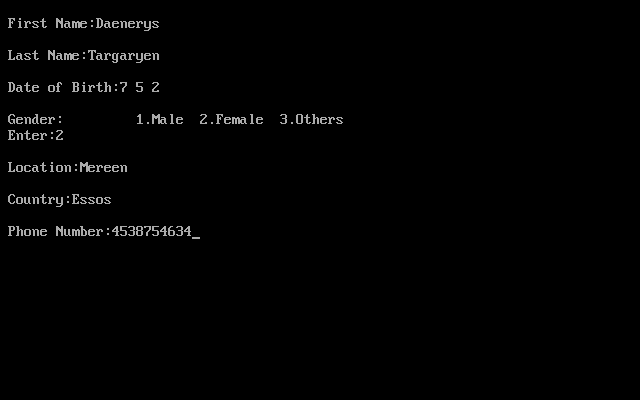
**OUTPUT:**

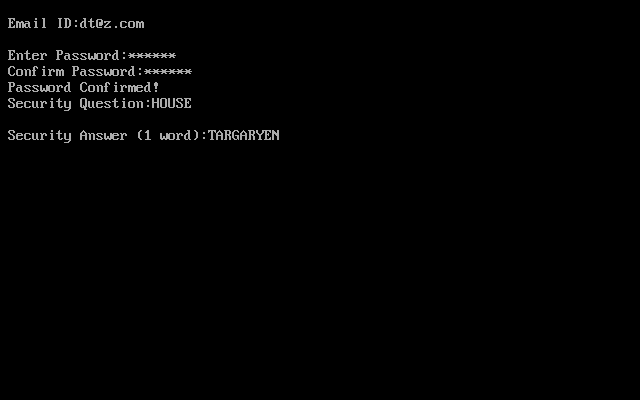


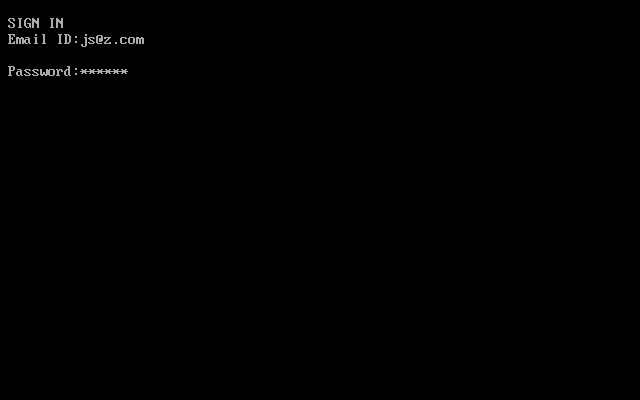


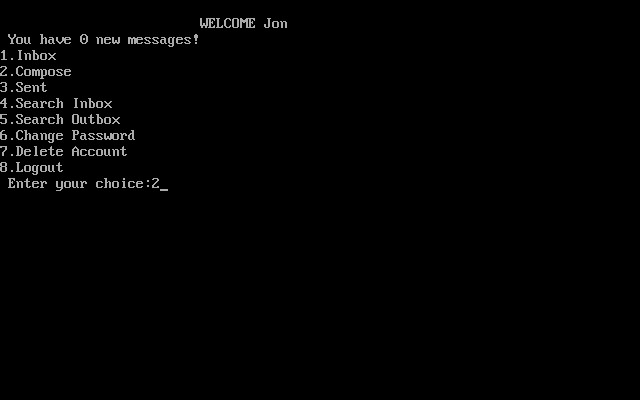


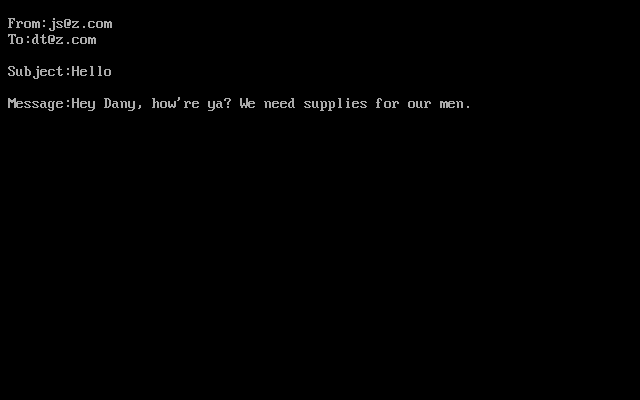


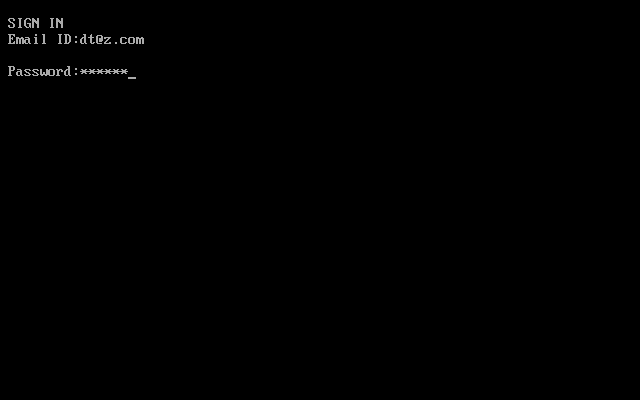




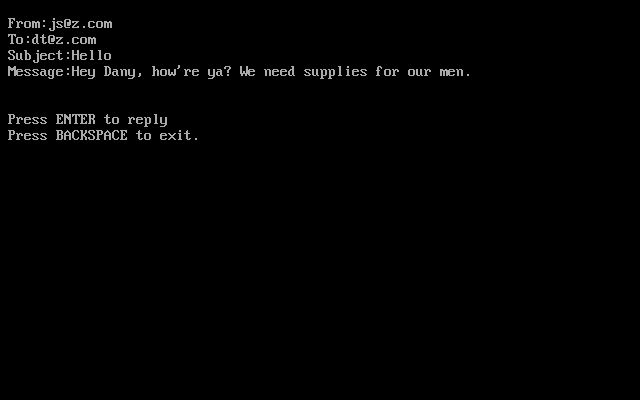


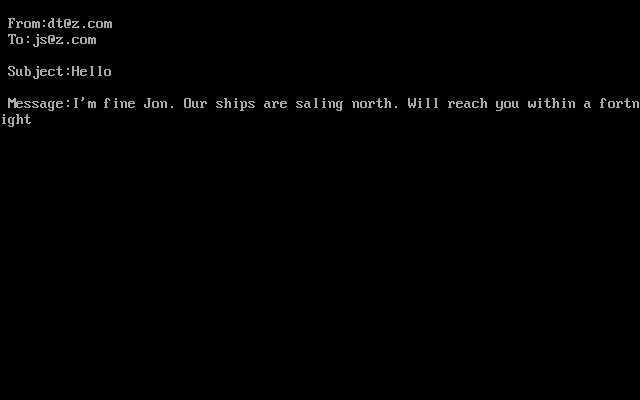




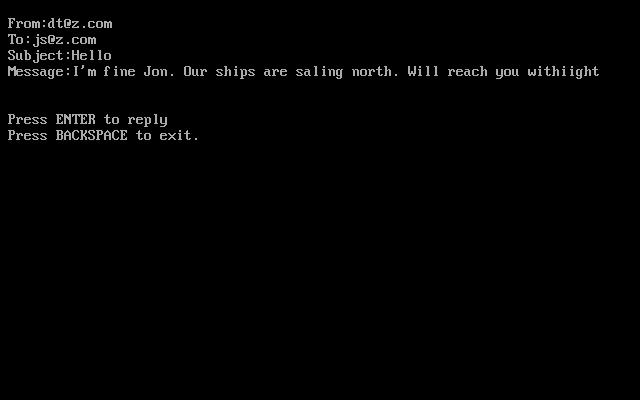


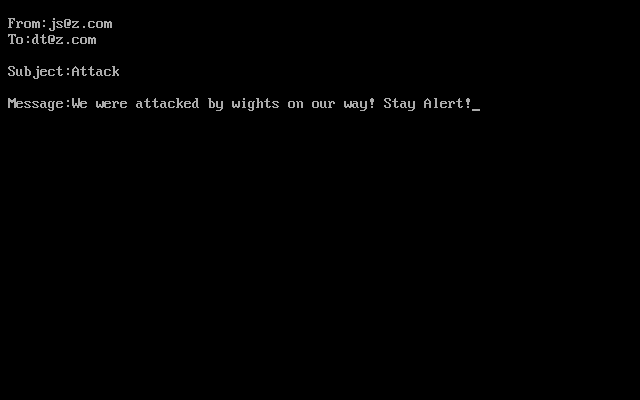


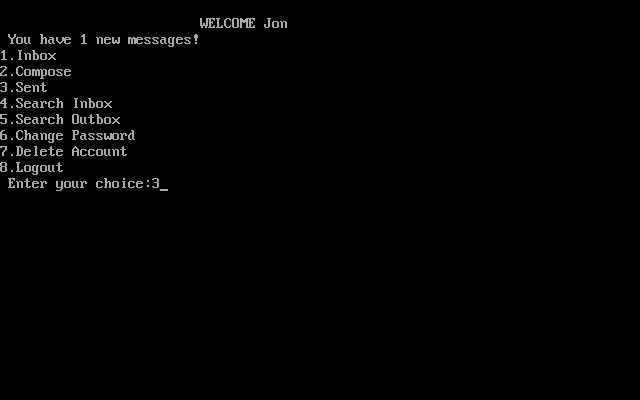


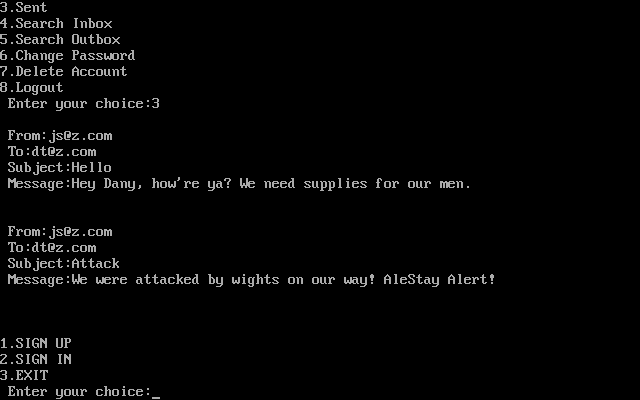


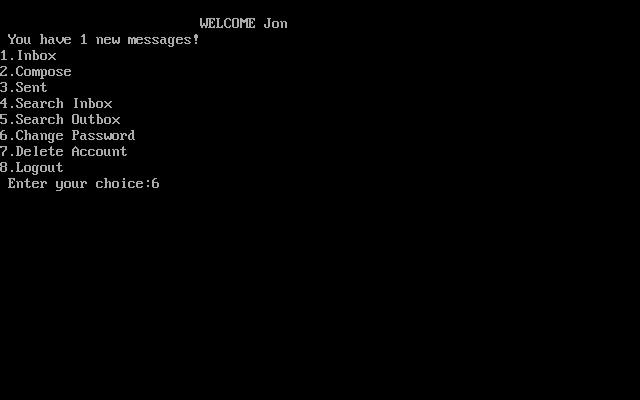




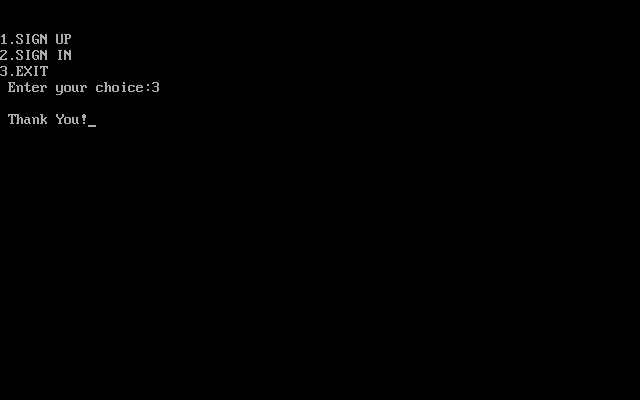








****



**CONCLUSION:**

* Thus the program has been executed successfully.
* This program models the management of data and endures the features of OOP language.
* Now the email service is very useful to people as it acts as a communication device and people can enjoy it by chatting with other people.



FUTURE ENHANCEMENTS:

* The program “email service” is just a prototype of the actual full version of the program.
* The further updates include more accessibility and including various other options.
* The future enhancements will also include adding more attractive background using graphics and to make the service more attractive.



BIBILIOGRAPHY:

* Computer Science with C++ by Sumita Aroara
* The C++ Programming Language by Bjarne Stroustrup
* Object Oriented Programming by Robert Lafore

WEBLIOGRAPHY:

* <https://stackoverflow.com/>
* <https://www.geeksforgeeks.org/c-programming-language/>
* <https://www.hackerrank.com/>

