

CONTENTS:

INTRODUCTION:	<u>3</u>
SOURCE CODE	<u>3</u>
OUTPUT	<u>10</u>

“CONTACT BOOK MANAGEMENT SYSTEM”

INTRODUCTION:

This report is about the management system of CONTACT BOOK using singly linklist. In this system, we will see how the contacts are added, deleted, searching, displaying either by using contact's name or phone number. For using the linklist, we have made a structure node. A class having name ContactBook is also made that has one constructor ContactBook() and the the functions of CreateNode(), Display(), Search(), DeleteAllContact(), DeleteAllContactsBySearch(), EditContact(), BubbleSort(), OfflineSave() and Structure().

SOURCE CODE:

```
// DS THEORY PROJECT.cpp : Defines the entry point for the console application.
//

#include<iostream>
#include<fstream>
#include <sstream>
using namespace std;
struct Node
{
    string name;
    long long int phone_number;
    Node *next;
    Node *prev;
};
class ContactBook
{
    Node *head;
    string x;
    long long int y;
public:
    ContactBook()
    {
        head=NULL;
        x="";
        y=0;
    }
    void CreateNode()
    {
        if(head==NULL)
        {
            Node *newer= new Node;
            cout<<" Name of Contact: ";
            cin>>x
            newer->name=x;
            cout<<" Phone Number: ";
            cin>>y;
            newer->phone_number=y;
            newer->next=NULL;
            newer->prev==NULL;
            head=newer;
            cout<<" Contact Added"<<endl
        }
        Else
        {
            Node *newer= new Node;
            cout<<" Name of Contact: ";
            cin>>x;
            newer->name=x;
            cout<<" Phone Number: ";
            cin>>y;
            newer->phone_number=y;
            newer->next=NULL;
            Node *temp=head;
            while(temp->next!=NULL)
            {
                temp=temp->next;
            }
            temp->next=newer;
            newer->prev=temp;
            cout<<" Contact Added"<<endl;
        }
    }
}
```

```

}
void Display()
{
Node *temp=head;
int count=0;
if(temp==NULL)

cout<<" No Contacts... Please Add Some
Contacts"<<endl;
cout<<" Thanks"<<endl;
}
Else
{
BubbleSort();
cout<<" Name: "<< Number: \n"<<endl;
while(temp!=NULL)
{
count++;
cout<<" "<<temp->name;
cout<<" "<<temp->phone_number<<endl;
temp=temp->next;
}
cout<<" Total contacts: "<<count<<endl;
}
}
int Search()
{
bool check=false;
Node *temp=head;
cout<<"*****"<<endl;
cout<<" Press 1 if you want to Search By
Name."<<endl;
cout<<" Press 2 if you want to Search By
Number."<<endl;
int command;
cout<<" Enter the Command: ";
cin>>command;

if(command==1 && temp!=NULL)
{
cout<<" Enter the Name to Search: ";
cin>>x;
while(temp!=NULL)
{
if(temp->name==x)
{

```

```

cout<<"*****"<<endl;
cout<<" Name: "<<temp->name<<endl;
cout<<" Phone Number:"<<temp-
>phone_number<<endl;
cout<<"*****"<<endl;
check=true;
break
{
temp=temp->next;
}
if(check==false)
{
cout<<" Name Not Found"<<endl;
}
}
else if(command==2)
{
cout<<" Enter the Number to Search: ";
cin>>y;
while(temp!=NULL)
{
if(temp->phone_number==y)
{
cout<<"*****"<<endl;
cout<<" Name: "<<temp->name<<endl;
cout<<" Phone Number: "<<temp-
>phone_number<<endl;
cout<<"*****"<<endl;
check=true;
break;
}
temp=temp->next;
}
if(check==false)
{
cout<<" Number Not Found"<<endl;
}
}
}
void DeleteAllContacts()
{
Node *temp=head;
Node *temp2;
if(head==NULL)
{

```

```

cout<<" Already Contact Book is
Empty"<<endl;
cout<<"*****"<<endl;
}
else
{
while(temp!=NULL)
temp2=temp;
temp=temp->next;
delete temp2;
}
head=NULL;
cout<<" Successfully Deleted All
Contacts"<<endl;
cout<<"*****"<<endl;
}
}

int DeleteContactBySearch()
{
Node *temp=head;
cout<<"*****"<<endl;
cout<<" Press 1 if you want to Search By
name"<<endl;
cout<<" Press 2 if you want to Search By
Number"<<endl;
int command;
cout<<" Enter the Command: ";
cin>>command;
if(command==1)
{
bool Dcheck=false;
cout<<" Enter the Name to Delete: ";
cin>>x;
while(temp!=NULL)
{
if(temp->name==x)
{
cout<<"*****"<<endl;
cout<<" Name: "<<temp->name<<endl;
cout<<" Phone Number: "<<temp-
>phone_number<<endl;
cout<<"*****"<<endl;
Dcheck=true;
Break;

```

```

}
temp=temp->next;
}
if(Dcheck==true)
{
int command;
cout<<" Press 1 to Delete the Contact: ";
cin>>command;
if(command==1 & temp==head)
{
Node *temp1;
temp1=temp;
temp=temp->next;
delete temp1;

temp->prev=NULL;
head=temp;
cout<<" Contact Deleted Success
Fully"<<endl;
}
else if(command==1 & temp>next==NULL)
{
temp->prev->next=NULL;
delete temp;
cout<<" Contact Deleted Success
Fully"<<endl;
}
else if(command==1)
{
Node *temp1;
temp1=temp;
temp->prev->next=temp1->next;
temp->next->prev=temp1->prev;
delete temp1;
cout<<" Contact Deleted Success
Fully"<<endl;
}
else
{
cout<<" You Enter Wrong Command ... Try
Again"<<endl;
}}
else if(Dcheck==false)
{
cout<<" Contact of This Name Not
Found."<<endl;

```

```

    } }
    else if(command==2)
    {
        bool Dcheck=false;
        cout<<" Enter the Number to Delete: ";
        cin>>y;
        while(temp!=NULL)
        {
            if(temp->phone_number==y)
            {
                cout<<"*****"<<endl;
                cout<<"name: "<<temp->name<<endl;
                cout<<"Phone Number: "<<temp->phone_number<<endl;
                cout<<"*****"<<endl;
                Dcheck=true;
                break;
            }
            temp=temp->next;
        }
        if(Dcheck==true)
        {
            int command;
            cout<<" Press 1 to Delete the Contact: ";
            cin>>command;
            if(command==1 & temp==head)
            {
                Node *temp1;
                temp1=temp;
                temp=temp->next;
                delete temp1;
                temp->prev=NULL;
                head=temp;
                cout<<" Contact Deleted Success Fully"<<endl;
            }
            else if(command==1 & temp->next==NULL)
            {
                temp->prev->next=NULL;
                delete temp;
                cout<<" Contact Deleted Success Fully"<<endl;
            }
            else if(command==1)
            {
                Node *temp1;

```

```

                temp1=temp;
                temp->prev->next=temp1->next;
                temp->next->prev=temp1->prev;
                delete temp1;
                cout<<" Contact Deleted Success Fully"<<endl;
            }
            else
            {
                cout<<" You Enter Wrong Command ... Try Again"<<endl;
            } }
            else if(Dcheck==false)
            {
                cout<<" Contact of This Name Not Found."<<endl;
            } }
            else
            {
                cout<<" You Enter wrong Command"<<endl;
            }
        }
        void BubbleSort()
        {
            Node *temp=head;
            Node *i,*j;
            string n;
            long long int n2;
            if(temp=NULL)
            {
                cout<<" Empty contact Book"<<endl;
            }
            else
            {
                for(i=head;i->next!=NULL;i=i->next)
                {
                    for(j=i->next;j!=NULL;j=j->next)
                    {
                        if(i->name>j->name)
                        {
                            n=i->name;
                            i->name=j->name;
                            j->name=n;

                            n2=i->phone_number;
                            i->phone_number=j->phone_number;

```

```

j->phone_number=n2;
}
}
}}}

int EditContacts()
{
Node *temp=head;
cout<<"*****"<<endl;
cout<<" Press 1 if you want to Search By
Name"<<endl;
cout<<" Press 2 if you want to Search By
Number"<<endl;
int Ecommand;
cout<<" Enter the Command: ";
cin>>Ecommand;

if(Ecommand==1)
{
bool Echeck=false;
cout<<" Enter the Name to Edit: "
cin>>x;
while(temp!=NULL)
{
if(temp->name==x)
{
cout<<"*****"<<endl;
cout<<"Name: "<<temp->name<<endl;
cout<<"Phone Number: "<<temp-
>phone_number<<endl;
cout<<"*****"<<endl;
Echeck=true;
break;
}
temp=temp->next;
}
if(Echeck==true)
{
int command;
cout<<" Press 1 to Edit the Contact: ";
cin>>command;
if(command==1)
{
cout<<" Enter New Name: ";
cin>>x;
cout<<" Enter New Number: ";
cin>>y;

temp->name=x;
temp->phone_number=y;

cout<<" Contact Edited Success
Fully"<<endl;
}
else
{
cout<<" You Enter Wrong Command ...
Try Again"<<endl;
}
}
else if(Echeck==false)
{
cout<<" Contact Not Found"<<endl;
}
}
else if(Ecommand==2)
{
bool Echeck=false;
cout<<" Enter the Number to Edit: ";
cin>>y;
while(temp!=NULL)
{
if(temp->phone_number==y)
{
cout<<"*****"<<endl;
cout<<"Name: "<<temp->name<<endl;
cout<<"Phone Number: "<<temp-
>phone_number<<endl;
cout<<"*****"<<endl;
Echeck=true;
break;
}
temp=temp->next;
}
if(Echeck==true)
{
int command;
cout<<" Press 1 to Edit the Contact: ";
cin>>command;
if(command==1)
{
cout<<" Enter New Name: ";

```

```

cin>>x;
cout<<" Enter New Number: ";
cin>>y;

temp->name=x;
temp->phone_number=y;

cout<<" Contact Edited Success Fully"<<endl;
}
else
{
cout<<" You Enter Wrong Command"<<endl;
}
}
else if(Echeck==false)
{
cout<<" There is No Contact of this
Number."<<endl;
}
}
else
{
cout<<" You Enter Wrong Command ... Try
Again"<<endl;
}
}

void OfflineSave()
{
Node *temp=head;
ofstream myfile ("contactbook.txt");
if (myfile.is_open())
{
while(temp!=NULL)
{
myfile<<temp->name<<endl;
myfile<<temp->phone_number<<endl;
temp=temp->next;
}
myfile.close();
Structure();
}
else
{
cout<<" Thanks file is empty."<<endl;
}
}

}
}

void reopenCB()
{
bool isEmpty;
ifstream myfile ("contactbook.txt");
if (myfile.is_open() & myfile.peek() != EOF)
{
int i=0;
while(getline(myfile,x))
{
if(i % 2 == 0)
{
if(head==NULL)
{
Node *newer= new Node;
newer->name=x;

newer->next=NULL;
newer->prev==NULL;
head=newer;
}
else
{
Node *newer= new Node;

newer->name=x;
newer->next=NULL;

Node *temp=head;
while(temp->next!=NULL)
{
temp=temp->next;
}
temp->next=newer;
newer->prev=temp;
}
}
else
{
Node *temp=head;
if(temp->phone_number==0)
{
stringstream convert(x);

```

```

convert>>y;
temp->phone_number=y;
}
else
{
Node *temp=head;
while(temp->next!=NULL)
{
temp=temp->next;
}

stringstream convert(x);
convert>>y;
temp->phone_number=y;

}

}
i++;
}
myfile.close();
}
else
{
cout<<" File is Empty so Cannot
open...Sorry"<<endl;
}
}
void Structure()
{
cout<<"*****"<<endl;
cout<<" 1. Add Contact"<<endl;
cout<<" 2. Edit the Contact"<<endl;
cout<<" 3. Delete Contact"<<endl;
cout<<" 4. Search Contact"<<endl;
cout<<" 5. Display All Contacts"<<endl;
cout<<" 6. Delete All Contacts"<<endl;
cout<<"*****"<<endl;

int Scommand;
cout<<" Enter the Command: ";
cin>>Scommand;
try
{
if(Scommand>=1&&Scommand<=6)
{

```

```

if(Scommand==1)
{
CreateNode();
OfflineSave();
system("CLS");
Structure();
}
else if(Scommand==2)
{
EditContacts();
OfflineSave();
Structure();
}
else if(Scommand==3)
{
DeleteContactBySearch();
Structure();
}
else if(Scommand==4)
{
Search();
Structure();
}
else if(Scommand==5)
{
Display();
OfflineSave();
Structure();
}
else if(Scommand==6)
{
DeleteAllContacts();
OfflineSave();
Structure();
}
}
else
{
throw(Scommand);
}
}
catch(int Scommand)
{
cout<<" You Enter wrong Command... Run the
Code Again"<<endl;
Structure();
}

```



```

    }
};
int main()
{
    ContactBook cb;
    cb.reopenCB();
    string n;
    cout<<" What is Your Name: ";
    cin>>n;
    cout<<"*****"<<endl;

```

```

    cout<<" "<<n<<" WELCOME TO
CONTACTBOOK  "<<endl;
    cout<<"*****"<<endl;
    cb.Structure();
    system("pause");
    return 0;
}

```

OUTPUT:

```

C:\Users\it\Documents\laiba sohail semester 3\DS THEORY PROJECT.exe
File is Empty so Cannot open...Sorry
What is Your Name: laiba
*****
laiba WELCOME TO CONTACTBOOK
*****
*****
1. Add Contact
2. Edit the Contact
3. Delete Contact
4. Search Contact
5. Display All Contacts
6. Delete All Contacts
*****
Enter the Command:

```

TO ADD CONTACT:

```

6. Delete All Contacts
*****
Enter the Command: 1
Name of Contact: neha
Phone Number: 0300529876
Contact Added
*****
1. Add Contact
2. Edit the Contact
3. Delete Contact
4. Search Contact
5. Display All Contacts
6. Delete All Contacts
*****

```

```

C:\Users\it\Documents\laiba sohail semester 3\DS THEORY PROJECT.exe
Enter the Command: 1
Name of Contact: neha
Phone Number: 0200876493
Contact Added
*****
1. Add Contact
2. Edit the Contact
3. Delete Contact
4. Search Contact
5. Display All Contacts
6. Delete All Contacts
*****
Enter the Command: 1
Name of Contact: hisma
Phone Number: 098536241
Contact Added
*****
1. Add Contact
2. Edit the Contact
3. Delete Contact
4. Search Contact
5. Display All Contacts
6. Delete All Contacts
*****
Enter the Command: 1
Name of Contact: tanzeela
Phone Number: 0345789234
Contact Added
*****

```

TO EDIT CONTACT:

```
Enter the Command: 2
*****
Press 1 if you want to Search By Name
Press 2 if you want to Search By Number
Enter the Command: 1
Enter the Name to Edit: neha
*****
Name: neha
Phone Number: 300529876
*****
Press 1 to Edit the Contact: 1
Enter New Name: tanzeela
Enter New Number: 0300598450
Contact Edited Success Fully
*****
```

TO DELETE CONTACT:

```
*****
Enter the Command: 3
*****
Press 1 if you want to Search By name
Press 2 if you want to Search By Number
Enter the Command: 2
Enter the Number to Delete: 098536241
*****
name: bisma
Phone Number: 98536241
*****
Press 1 to Delete the Contact: 1
Contact Deleted Success Fully
*****
```

TO SEARCH CONTACT:

```
Enter the Command: 4
*****
Press 1 if you want to Search By Name.
Press 2 if you want to Search By Number.
Enter the Command: 1
Enter the Name to Search: neha
*****
Name: neha
Phone Number:3005162290
*****
```

TO DISPLAY CONTACTS:

```
Enter the Command: 5
Name:      Number:
bisma      3000456723
neha        3005162290
neha        200876493
tanzeela    300598450
tanzeela    345789234
Total contacts: 5
```

TO DELETE ALL CONTACTS:

```
*****
Enter the Command: 6
Successfully Deleted All Contacts
*****
```

NOTEPAD FILE:



