Name: prathamesh sathe

59

#include<iostream>

#include<fstream>

using namespace std;

class student

{

int roll;

char div;

char name[30];

char address[50];

int size;

public:

student()

{

roll = 0;

div = 0;

size = 0;

}

void create();

void addRecord();

void deleteRecord();

void displayAll();

};

void student::create()

{

ofstream file;

file.open("db.dat", ios::out|ios::binary);

if(!file)

{

cout<<"\nFailed to create database!";

return ;

}

student obj1;

cout<<"\nEnter the number of students: ";

cin>>obj1.size;

cout<<"\nEnter the records: ";

for(int i=0 ; i<obj1.size ; i++)

{

cout<<"\nEnter the data for student " << i+1 << " :";

cout<<"\nRoll number: ";

cin>>obj1.roll;

cout<<"Division: ";

cin>>obj1.div;

cin.ignore();

cout<<"Name: ";

cin.getline(obj1.name, 30);

cout<<"Address: ";

cin.getline(obj1.address, 50);

file.write((char \*)&obj1, sizeof(obj1));

}

cout<<"\n\nDatabase successfully created!";

file.close();

}

void student::addRecord()

{

ofstream file;

file.open("db.dat", ios::app|ios::out|ios::binary);

if(!file)

{

cout<<"\nFailed to create database!";

return ;

}

student obj1;

cout<<"\nEnter the details: ";

cout<<"\nRoll number: ";

cin>>obj1.roll;

cout<<"Division: ";

cin>>obj1.div;

cin.ignore();

cout<<"Name: ";

cin.getline(obj1.name, 30);

cout<<"Address: ";

cin.getline(obj1.address, 50);

file.write((char \*)&obj1, sizeof(obj1));

size++;

}

void student::displayAll()

{

ifstream file;

file.open("db.dat", ios::binary|ios::in);

if(!file)

{

cout<<"\nFailed to open database!";

return ;

}

student obj;

cout<<"\nDATABASE: \n";

while(1)

{

file.read((char\*)&obj, sizeof(obj));

if(file.eof())

break;

cout<<"\n\nRoll number: ";

cout<<obj.roll;

cout<<"\nDivision: ";

cout<<obj.div;

cout<<"\nName: ";

cout<<obj.name;

cout<<"\nAddress: ";

cout<<obj.address;

}

file.close();

}

void student::deleteRecord()

{

ifstream file;

ofstream tempFile;

int key, flag=0;

file.open("db.dat", ios::binary|ios::in);

tempFile.open("tempdb.dat", ios::binary|ios::out);

if(!file)

{

cout<<"\nFailed to open database!";

return ;

}

student obj;

cout<<"\nEnter the roll no. to delete: ";

cin>>key;

while(1)

{

file.read((char\*)&obj, sizeof(obj));

if(file.eof())

break;

if(obj.roll==key)

{

flag=1;

cout<<"\n\nRecord found!";

cout<<"\nRoll number: ";

cout<<obj.roll;

cout<<"\nDivision: ";

cout<<obj.div;

cout<<"\nName: ";

cout<<obj.name;

cout<<"\nAddress: ";

cout<<obj.address;

continue;

}

else

tempFile.write((char\*)&obj, sizeof(obj));

}

if(flag==0)

cout<<"\nRecord not found!";

else

{

remove("db.dat");

rename("tempdb.dat", "db.dat");

}

cout<<"\nDeleted record successfully";

file.close();

tempFile.close();

}

int main()

{

MENU:

int choice;

student db1;

cout<<"\n\n\nSTUDENT DATABASE MANAGEMENT";

cout<<"\n1. Create database";

cout<<"\n2. Add record";

cout<<"\n3. Delete record";

cout<<"\n4. Display database";

cout<<"\n5. EXIT";

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

cin>> choice;

switch(choice)

{

case 1:

db1.create();

break;

case 2:

db1.addRecord();

break;

case 3:

db1.deleteRecord();

break;

case 4:

db1.displayAll();

break;

case 5:

return 0;

default:

cout<<"\nInvalid choice! Try again!";

}

goto MENU;

}

/\*

kiran î‚° kali î‚° ../Coding/SE4/DSAL î‚° g++ A11.cpp -o Asst11

kiran î‚° kali î‚° ../Coding/SE4/DSAL î‚° ./Asst11

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 1

Enter the number of students: 5

Enter the records:

Enter the data for student 1 :

Roll number: 2

Division: A

Name: Pranav

Address: Nashik

Enter the data for student 2 :

Roll number: 3

Division: A

Name: Aniket

Address: Panchavati nashik

Enter the data for student 3 :

Roll number: 1

Division: B

Name: Shubham

Address: Ambad Nashik

Enter the data for student 4 :

Roll number: 5

Division: B

Name: Divya

Address: Shahada

Enter the data for student 5 :

Roll number: 6

Division: A

Name: Rutuja

Address: Nashik

Database successfully created!

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 4

DATABASE:

Roll number: 2

Division: A

Name: Pranav

Address: Nashik

Roll number: 3

Division: A

Name: Aniket

Address: Panchavati nashik

Roll number: 1

Division: B

Name: Shubham

Address: Ambad Nashik

Roll number: 5

Division: B

Name: Divya

Address: Shahada

Roll number: 6

Division: A

Name: Rutuja

Address: Nashik

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 2

Enter the details:

Roll number: 17

Division: A

Name: Yash

Address: Yeola Nashik

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 2

Enter the details:

Roll number: 94

Division: C

Name: Vivek

Address: Sinner

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 3

Enter the roll no. to delete: 2

Record found!

Roll number: 2

Division: A

Name: Pranav

Address: Nashik

Deleted record successfully

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 4

DATABASE:

Roll number: 3

Division: A

Name: Aniket

Address: Panchavati nashik

Roll number: 1

Division: B

Name: Shubham

Address: Ambad Nashik

Roll number: 5

Division: B

Name: Divya

Address: Shahada

Roll number: 34

Division: B

Name: Rutuja

Address: Yeola Nashik

Roll number: 17

Division: A

Name: Yash

Address: Yeola Nashik

Roll number: 94

Division: C

Name: Vivek

Address: Sinner

STUDENT DATABASE MANAGEMENT

1. Create database

2. Add record

3. Delete record

4. Display database

5. EXIT

Enter your choice: 5