**Code for Assignment 8:**

#include<iostream>

#include<fstream>

#include<string.h>

using namespace std;

class Student

{

char Name[10];

int Roll\_No;

public:

Student(){ Name[0]='\0';Roll\_No=-1;}

void get\_Data();

void put\_Data();

int return\_Roll(){ return Roll\_No;}

};

void Student::get\_Data()

{

cout<<"\nEnter Student Data";

cout<<"\nName";cin>>Name;

cout<<"\nRoll Number: ";cin>>Roll\_No;

}

void Student::put\_Data()

{

cout<<"\n"<<Roll\_No<<"\t"<<Name;

}

class Seq\_File

{

char File\_Name[15];

public:

Seq\_File();

Seq\_File(char F[]);

void Create();

void Display();

void Add();

void Remove(int);

void Modify(int);

};

Seq\_File::Seq\_File()

{

ofstream File("Student.txt");

strcpy(File\_Name," Student.txt");

cout<<"\nDefault Constructor";

if(File)

{

cout<<"\nFile opened Successfully";

File.close();

}

else cout<<"\nFile creation Error";

}

Seq\_File::Seq\_File(char F[15])

{

ofstream File;

strcpy(File\_Name,F);

File.open(F);

if(File)

{

cout<<"\nFile opened Successfully";

File.close();

}

else cout<<"\nFile creation Error";

}

void Seq\_File::Create()

{

ofstream File;

Student S;

File.open(File\_Name);

S.get\_Data();

File.write(reinterpret\_cast<char\*>(&S),sizeof(S));

File.close();

}

void Seq\_File::Display()

{

ifstream File;

Student S;

File.open(File\_Name);

cout<<"\nRoll No\t Student Name";

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

while(!File.eof())

{

S.put\_Data();

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

}

File.close();

}

void Seq\_File::Add()

{

ofstream File;

Student S;

File.open(File\_Name,ios::app);

S.get\_Data();

File.write(reinterpret\_cast<char\*>(&S),sizeof(S));

File.close();

}

void Seq\_File::Remove(int Roll)

{

ifstream File;

ofstream Temp;

Student S;

int Flag=0;

File.open(File\_Name);

Temp.open("Temp.Txt");

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

while(!File.eof())

{

if(Roll==S.return\_Roll())

{ S.put\_Data(); Flag=1;}

else

Temp.write(reinterpret\_cast<char\*>(&S),sizeof(S));

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

}

if(Flag==0) cout<<"Roll No. "<<Roll<<" does not present \n";

File.close();

Temp.close();

remove(File\_Name);

rename("Temp.Txt",File\_Name);

}

void Seq\_File::Modify(int Roll)

{

ifstream File;

ofstream Temp;

Student S;

int Flag=0;

File.open(File\_Name);

Temp.open("Temp.Txt");

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

while(!File.eof())

{

if(Roll==S.return\_Roll())

{

S.put\_Data();

cout<<"\n Enter data to modify";

S.get\_Data();

Flag=1;

}

Temp.write(reinterpret\_cast<char\*>(&S),sizeof(S));

File.read(reinterpret\_cast<char\*>(&S),sizeof(S));

}

if(Flag==0) cout<<"Roll No. "<<Roll<<" does not present \n";

File.close();

Temp.close();

remove(File\_Name);

rename("Temp.Txt",File\_Name);

}

int main()

{

int Choice;

char F[15];

int R;

cout<<"\nEnter File Name : ";

cin>>F;

Seq\_File sFile(F);

do

{

cout<<"\n1: Create Database\n2: Display Database\n3: Add a record\n4: Delete a record\n5: Modify a record\nEnter your choice: ";

cin>>Choice;

switch(Choice)

{

case 1:

sFile.Create();

break;

case 2:

sFile.Display();

break;

case 3:

sFile.Add();

break;

case 4:

cout<<"\nEnter Roll No to delete";

cin>>R;

sFile.Remove( R);

break;

case 5:

cout<<"\nEnter Roll No to Modify";

cin>>R;

sFile.Modify( R);

break;

}

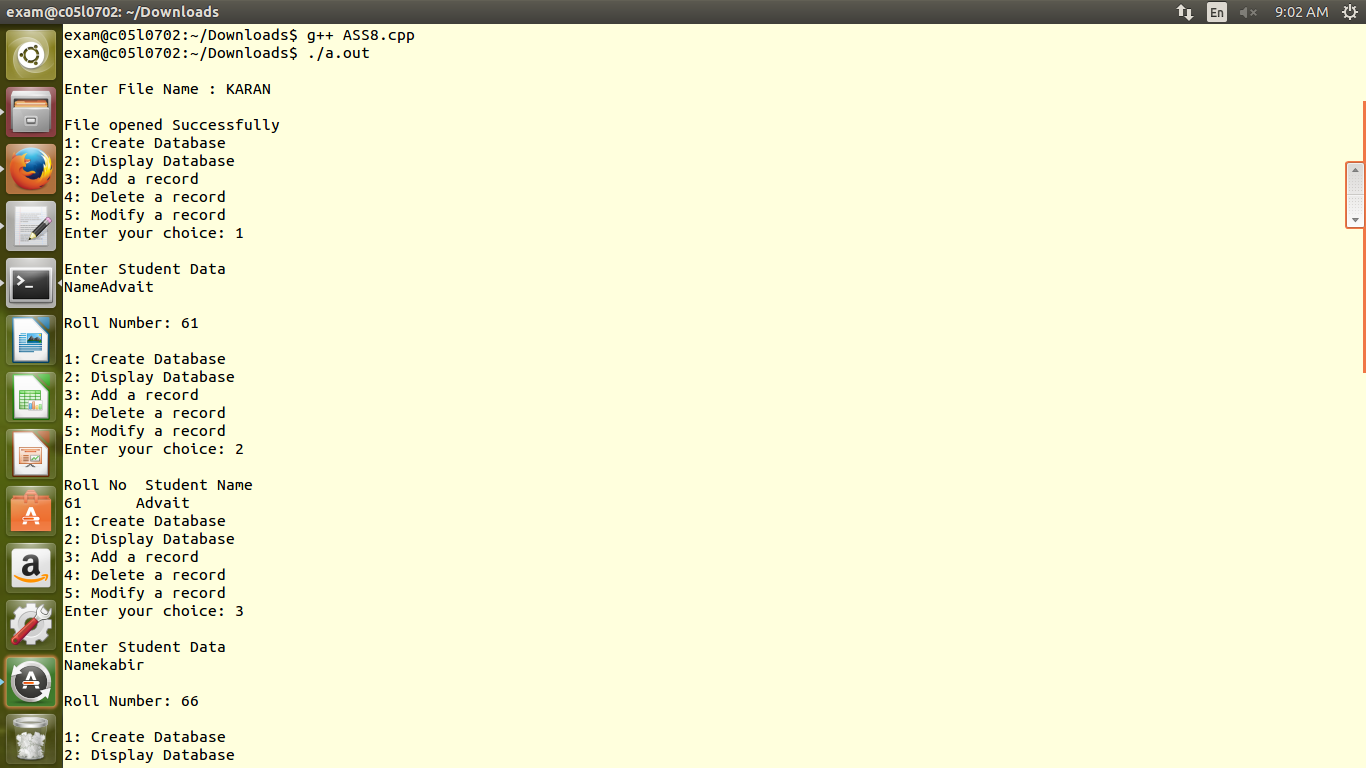
} while(Choice<6);

cout<<"\n";

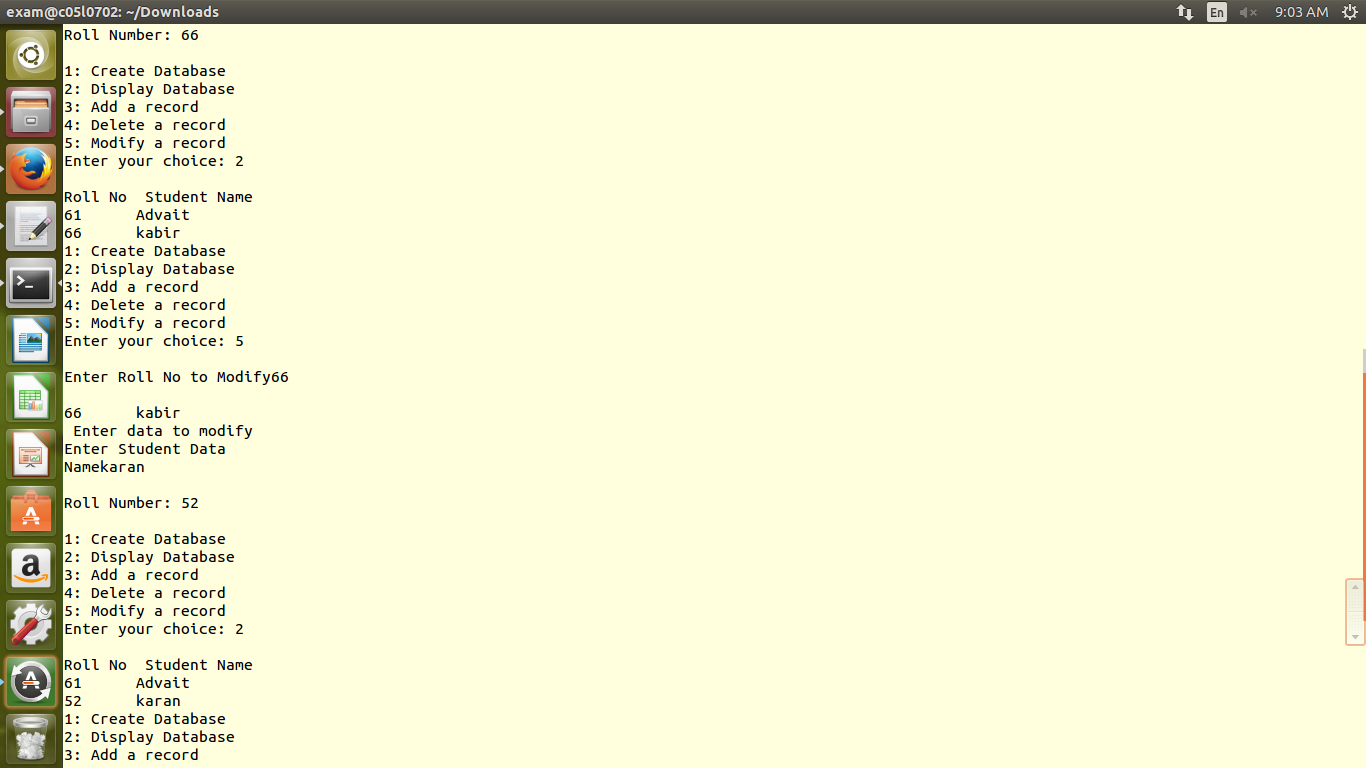
return 1;

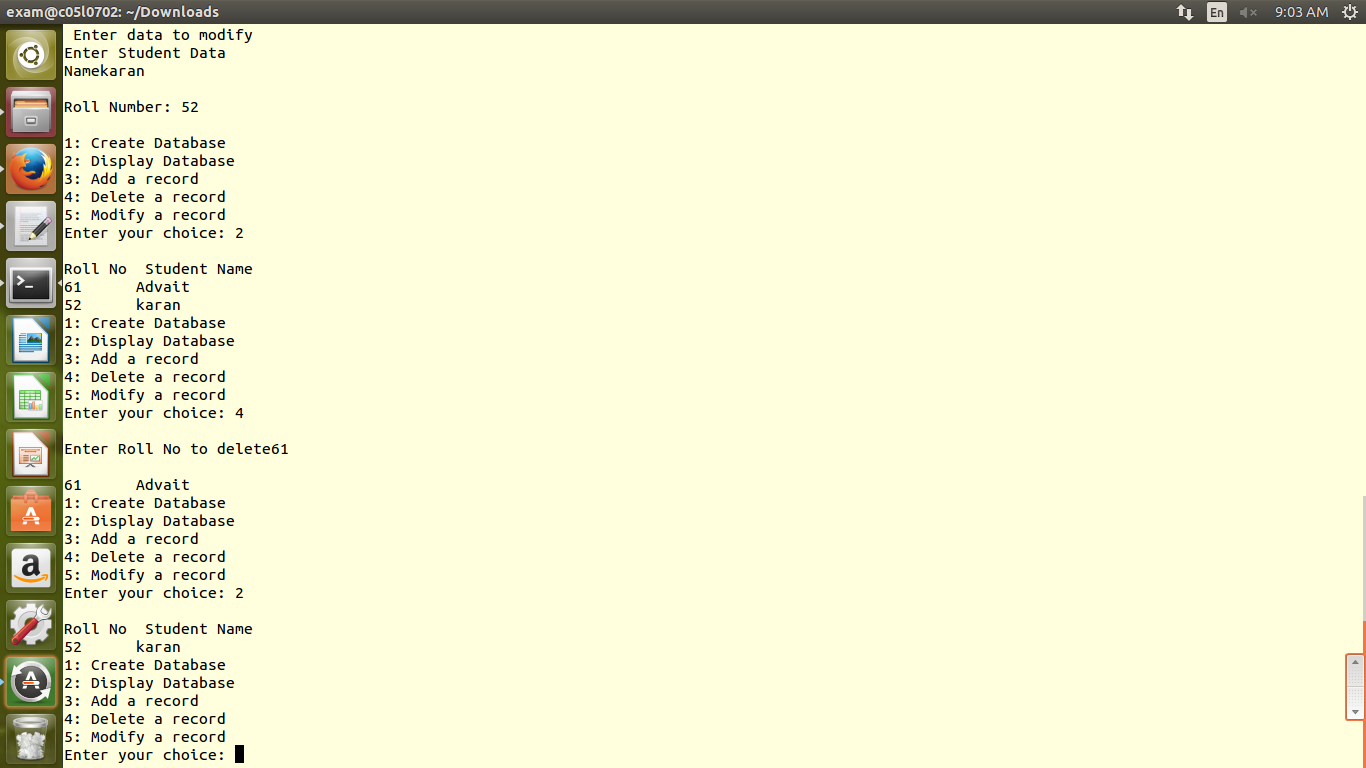
}

**Output:**

****

kedar

****

****