/\*

Cody Murrer

Project-1

10/13/2019

main.cpp

\*/

#include "studentDB.h"

#include "student.h"

#include "course.h"

#include<sstream>

#include<string>

#include <fstream>

#include<iostream>

using namespace std;

int main(){

string majorList;

string test;

char charAnswer,charCourse, charMajor;

StudentDB data;

cout<<"would you like to create a new student y/n: ";

cin>>charAnswer;

cin.ignore();

while(charAnswer != 'y' && charAnswer != 'n'){

cout<<"Your input of, "<<charAnswer<<" is not valid please use y or n";

cout<<"\nwould you like to create a new student y/n: ";

cin>>charAnswer;

cin.ignore();

if(charAnswer == 'y' && charAnswer == 'n')

break;

}

while(charAnswer == 'y'){

string nam,dob,maj;

cout<<"Student Name: ";

getline(cin,nam);

cin.ignore();

cout<<"\nStudent date Of Birth (month/day/year): ";

getline(cin,dob);

cin.ignore();

ifstream inFile;

inFile.open("Majors.txt");

if(!inFile){

cout<<"File didnt open or there is no such file";

exit(1);

}

while(inFile >> majorList){

cout<<majorList<<"\n";

}

inFile.close();

cout<<"Here is a list of current majors\n";

cout<<"\nDo you wish to add a major to the list y/n: ";

cin>>charMajor;

cin.ignore();

while(charMajor != 'y' && charMajor != 'n'){

cout<<"Your input of, "<<charMajor<<" is not valid please use y or n";

cout<<"\nDo you wish to add a major to the list y/n: ";

cin>>charMajor;

cin.ignore();

if(charMajor == 'y' && charMajor == 'n')

break;

}

if(charMajor == 'y'){

string newMajor;

cout<<"\nWhat is the new Major: ";

ofstream writeFile;

writeFile.open("Majors.txt", ios::app);

cin>>newMajor;

writeFile<<newMajor<<"\n";

writeFile.close();

}

cout<<"\nStudent Major: ";

getline(cin,maj);

cin.ignore();

data.create(Student(nam,dob,maj));

cout<<"\nDo you wish to create another student y/n: ";

cin>>charAnswer;

cin.ignore();

while(charAnswer != 'y' && charAnswer != 'n'){

cout<<"Your input of, "<<charAnswer<<" is not valid please use y or n";

cout<<"\nDo you wish to create another student y/n: ";

cin>>charAnswer;

cin.ignore();

if(charAnswer == 'y' && charAnswer == 'n')

break;

}

}

cout<<"Would you like to enter courses for a student y/n: ";

cin>>charCourse;

cin.ignore();

while(charCourse != 'y'&& charCourse != 'n'){

cout<<"Your input of, "<<charCourse<<" is not valid please use y or n";

cout<<"\nWould you like to enter courses for a student y/n: ";

cin>>charCourse;

cin.ignore();

if(charCourse == 'y' && charCourse == 'n')

break;

}

if(charCourse == 'y'){

cout<<"\nEnter the Students name: ";

cin>>test;

while(charCourse == 'y'){

string con,dep,sem;

char gra;

cout<<"\nEnter a course: ";

cin>>con;

cout<<"\nWhat department is the course from: ";

cin>>dep;

cout<<"\nWhat semester was it taken in: ";

cin>>sem;

cout<<"\nWhat was the final letter grade: ";

cin>>gra;

data.search(test,con,dep,sem,gra);

cout<<"\nDo you wish to enter another course y/n";

cin>>charCourse;

while(charCourse != 'y'&& charCourse != 'n'){

cout<<"Your input of, "<<charCourse<<" is not valid please use y or n";

cout<<"\nDo you wish to enter another class y/n";

cin>>charCourse;

cin.ignore();

if(charCourse == 'y' && charCourse == 'n')

break;

}

}

}

data.print();

return 0;

}

/\*

student.cpp

\*/

#include "student.h"

#include "course.h"

Student::Student(){ // default constructor

}

Student::Student(string nam, string dob, string maj){ // constructor

head = NULL;

this->setName(nam);

this->setDateOfBirth(dob);

this->setMajor(maj);

}

void Student::create(Course c){

create(c.getCourseName(),c.getDepartment(), c.getSemester(), c.getGrade());

}

void Student::create(string con, string dep, string sem, char gra){

ListNode\* newNode = new ListNode;

Course c(con,dep,sem,gra);

newNode->c = c;

newNode->next = NULL;

if(head == NULL){

newNode->next = head;

head = newNode;

return;

}

ListNode\* temp = head;

while(temp->next){

temp = temp->next;

}

newNode->next = temp->next;

temp->next = newNode;

}

void Student::setName(string nam){ // sets students name

name = nam;

}

string Student::getName(){ // gets students name

return name;

}

void Student::setDateOfBirth(string dob){ // sets students date of birth

dateOfBirth = dob;

}

string Student::getDateOfBirth(){ // gets students date of birth

return dateOfBirth;

}

void Student::setMajor(string maj){ // sets students major

major = maj;

}

string Student::getMajor(){ // gets students major

return major;

}

void Student::print(){

ListNode\* temp = head;

while(temp != NULL){

cout<<"\nCourse Name: "<<temp->c.getCourseName()<<"\n";

cout<<"Department: "<<temp->c.getDepartment()<<"\n";

cout<<"Semester: "<<temp->c.getSemester()<<"\n";

cout<<"Grade: "<<temp->c.getGrade()<<"\n";

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

temp= temp->next;

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*

student.h

\*/

#ifndef STUDENT\_H

#define STUDENT\_H

#include "course.h"

#include<stdio.h>

#include<iostream>

using namespace std;

class Student{

struct ListNode{

Course c;

ListNode\* next;

};

ListNode\* head;

private:

string name;

string dateOfBirth;

string major;

public:

Student();

Student(string nam, string dob, string maj);

void setName(string nam);

string getName();

void setDateOfBirth(string dob);

string getDateOfBirth();

void setMajor(string maj);

string getMajor();

void create(Course);

void create(string, string, string, char);

void print();

};

#endif

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*

course.cpp

\*/

#include "course.h"

using namespace std;

Course::Course(){ // default constructor

}

Course::Course(string con, string dep, string sem, char gra){ // constructor

this->setCourseName(con);

this->setDepartment(dep);

this->setSemester(sem);

this->setGrade(gra);

}

void Course::setCourseName(string con){ // sets course name

courseName = con;

}

string Course::getCourseName(){ // gets course name

return courseName;

}

void Course::setDepartment(string dep){ // sets department

department = dep;

}

string Course::getDepartment(){ // gets department

return department;

}

void Course::setSemester(string sem){ // sets semester

semester = sem;

}

string Course::getSemester(){ // gets semester

return semester;

}

void Course::setGrade(char gra){ // sets grade

grade = gra;

}

char Course::getGrade(){ // gets grade

return grade;

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*

course.h

\*/

#ifndef COURSE\_H

#define COURSE\_H

#include<stdio.h>

#include<iostream>

using namespace std;

class Course{

private:

string courseName;

string department;

string semester;

char grade;

public:

Course();

Course(string con, string dep, string sem, char gra);

void setCourseName(string con);

string getCourseName();

void setDepartment(string dep);

string getDepartment();

void setSemester(string sem);

string getSemester();

void setGrade(char gra);

char getGrade();

};

#endif

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*

studentDB.cpp

\*/

#include "studentDB.h"

#include "student.h"

#include "course.h"

#include <iostream>

StudentDB::StudentDB(){

head = NULL;

}

StudentDB::~StudentDB(){

ListNode\* temp = head;

while(temp){

temp = temp->next;

delete head;

head = temp;

}

};

void StudentDB::create(Student s){

create(s.getName(), s.getDateOfBirth(), s.getMajor());

}

void StudentDB::create(string nam, string dob, string maj){

Student s(nam,dob,maj);

ListNode\* testNode = new ListNode;

testNode->s = s;

testNode->next = NULL;

if(head == NULL){

head = testNode;

return;

}

ListNode\* temp = head;

while(temp->next){

temp = temp->next;

}

temp->next = testNode;

}

void StudentDB::search(string test, string con, string dep, string sem, char gra){

ListNode\* temp = head;

while(temp){

if(temp->s.getName().compare(test) == 0){

temp->s.create(Course(con,dep,sem,gra));

}

temp = temp->next;

}

return;

}

void StudentDB::remove(){

}

void StudentDB::update(){

}

void StudentDB::print(){

ListNode\* temp = head;

while(temp){

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

cout<<"Name: "<<temp->s.getName()<<"\n";

cout<<"DOB: "<<temp->s.getDateOfBirth()<<"\n";

cout<<"Major: "<<temp->s.getMajor()<<"\n";

temp->s.print();

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

temp= temp->next;

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*

studentDB.h

\*/

#ifndef STUDENTDB\_H

#define STUDENTDB\_H

#include "student.h"

#include "course.h"

#include<stdio.h>

#include<iostream>

using namespace std;

class StudentDB{

private:

struct ListNode{

Student s;

ListNode\* next;

};

ListNode\* head;

public:

StudentDB();

~StudentDB();

StudentDB(const Student &list);

void create(string nam, string dob, string maj);

void create(Student);

void update();

void remove();

void search(string,string,string,string,char);

void print();

};

#endif