/// Andrew Souza

/// Comp 200 -- Fall 2023

/// Pointers & Structs Assignment -- Part 2

#include <iostream>

#include <vector>

using namespace std;

// Define a structure Student with a name and a vector<Course\*> of courses.

struct Course; // Student throws an error at compilation if Course is not pre-declared

struct Student {

string firstName;

string lastName;

int studentID = 0;

vector<Course\*> courses;

};

// Define a structure Course with a name and a vector<Student\*> of enrolled students.

struct Course {

string courseName;

vector<Student\*> students;

};

// prints the name of a student and the names of all courses that the student takes.

void print\_student(Student\* s) {

cout << "------" << endl;

cout << "Name: " << s->lastName << ", " << s->firstName << endl;

cout << "ID#: " << s->studentID << endl;

cout << "Enrolled courses: " << endl;

for (int i = 0; i < s->courses.size(); i++) {

cout << s->courses.at(i)->courseName << ", ";

}

cout << endl;

cout << "------" << endl;

}

// prints the name of a course and the names of all students in that course.

void print\_course(Course\* c) {

cout << "------" << endl;

cout << "Course: " << c->courseName << endl;

cout << "Enrolled students: " << endl;

for (int i = 0; i < c->students.size(); i++) {

cout << c->students.at(i)->lastName << " ";

cout << c->students.at(i)->firstName.substr(0,1) << ", ";

}

cout << endl;

cout << "------" << endl;

}

// enrolls the given student in the given course, updating both vectors.

void enroll(Student\* s, Course\* c) {

c->students.push\_back(s);

s->courses.push\_back(c);

}

int main() {

// In your main function, define several students and courses,

// and enroll students in the courses.

// Then call print\_student for all students and print\_course for all courses.

Course\* courses = new Course[6];

courses[0].courseName = "Computer Science";

courses[1].courseName = "Chemistry";

courses[2].courseName = "History";

courses[3].courseName = "Math";

courses[4].courseName = "Philosophy";

courses[5].courseName = "Physical Education";

int numStudents;

cout << "Enter a number of students: ";

cin >> numStudents;

cout << endl;

Student\* students = new Student[numStudents];

cout << "Input student info:" << endl;

for (int i = 0; i < numStudents; i++) {

cout << "Name: ";

cin >> students[i].firstName;

cin >> students[i].lastName;

students[i].studentID = i + 1;

}

cout << endl;

cout << "Students: " << endl;

for (int i = 0; i < numStudents; i++) {

cout << "Name:" << students[i].lastName << ", " << students[i].firstName;

cout << "\t|\tStudent ID#: " << students[i].studentID << endl;

}

cout << endl;

cout << "Courses: " << endl;

for (int i = 0; i < 6; i++) {

cout << "Name:" << courses[i].courseName;

cout << "\t|\tCourse ID#: " << i + 1 << endl;

}

cout << endl;

char userSelect;

cout << "Enroll a student in a course? y/n" << endl;

cin >> userSelect;

while (userSelect == 'y') {

int studentSelect;

int courseSelect;

cout << "Enroll a student in a course.";

cout << " Enter Student ID# followed by Course ID#" << endl;

cin >> studentSelect;

cin >> courseSelect;

Student\* studentPtr = &students[studentSelect - 1];

Course\* coursePtr = &courses[courseSelect - 1];

enroll(studentPtr, coursePtr);

print\_student(studentPtr);

print\_course(coursePtr);

cout << "Enroll again? y/n" << endl;

cin >> userSelect;

}

// Prints out each student at the end

for (int i = 0; i < numStudents; i++) {

Student\* studentPtr = &students[i];

print\_student(studentPtr);

}

// Prints out each course at the end

for (int i = 0; i < 6; i++) {

Course\* coursePtr = &courses[i];

print\_course(coursePtr);

}

delete[] courses;

delete[] students;

return 0;

}





