**Name:** Md Tahsin

**Id:** 2020-2-60-112

package Lab\_5;

import java.util.ArrayList;

public class **Course** {

private String courseId;

private String courseTitle;

private double credit;

private ArrayList<Student> studentList =new ArrayList<Student>();

private int numberOfStudents;

private Faculty faculty;

public Course(){

}

public Course(String courseId,String courseTitle,double credit){

this.courseId = courseId;

this.courseTitle = courseTitle;

this.credit =credit ;

}

public void setcourseId(String courseId){

this.courseId=courseId;

}

public String getcourseId(){

return courseId;

}

public void setcourseTitle(String courseTitle){

this.courseTitle=courseTitle;

}

public String getcourseTitle(){

return courseTitle;

}

public void setCredit(double credit){

this.credit = credit;

}

public double getCredit(){

return credit;

}

public Faculty getFaculty(){

return faculty;

}

public int getnumberOfStudents(){

return numberOfStudents;

}

//arraylist return.

public String toString(){

return "Course Id:"+getcourseId()+"\nTitle:"+getcourseTitle()+"\nCredit:"+getCredit();

}

public void addStudent(Student std){ //adding std

studentList.add(std);

}

public void dropStudent(double studentId){

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

if( studentId == studentList.get(i).getStudent() ){

studentList.remove(i);

}

}

}

public void addFaculty(Faculty faculty){

this.faculty=faculty;

}

public void dropFaculty(){

this.faculty=null;

}

public void printStudentList(){

numberOfStudents=studentList.size();

System.out.print(numberOfStudents+"\n"+studentList);

}

int getnumberOfStudent() {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

int studentID(int j) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

Object getcourseID() {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

void setcourseID(String nextLine) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

}

package Lab\_5;

public class **Faculty** {

int facultyId;

String facultyName;

String facultyPosition;

public Faculty(){

}

public Faculty(int facultyId,String facultyName,String facultyPosition){

}

public void setfacultyId(int facultyId){

this.facultyId=facultyId;

}

public int getfacultyId(){

return facultyId;

}

public void setfacultyName(String facultyName){

this.facultyName=facultyName;

}

public String getfacultyName(){

return facultyName;

}

public void setfacultyPosition(String facultyPosition){

this.facultyPosition=facultyPosition;

}

public String getfacultyPosition(){

return facultyPosition;

}

public String toString(){

return "Faculty Id:"+getfacultyId()+"\nName:"+getfacultyName()+"\nPosition:"+getfacultyPosition();

}

}

package Lab\_5;

public class **Student** {

int studentId;

String studentName;

double studentCGPA;

public Student(){

}

public Student(int studentId,String studentName,double studentCGPA){

this.studentId=studentId;

this.studentName=studentName;

this.studentCGPA=studentCGPA;

}

public void setstudentId(int studentId){

this.studentId=studentId;

}

public int getStudent(){

return studentId;

}

public void setstudentName(String studentName){

this.studentName=studentName;

}

public String getstudentName(){

return studentName;

}

public void setstudentCGPA(double studentCGPA){

this.studentCGPA=studentCGPA;

}

public double getstudentCGPA(){

return studentCGPA;

}

public String toString(){

return getStudent()+getstudentName()+getstudentCGPA();

}

int getstudentId() {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

}

package Lab\_5;

import java.util.\*;

public class **main** {

public static void prnt(String x) {

System.out.printf("%s", x);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

ArrayList<Student> studentList = new ArrayList<>();

ArrayList<Course> courseList = new ArrayList<>();

ArrayList<Faculty> facultyList = new ArrayList<>();

x:

while (true) {

// main menu

System.out.println("a.Add\nb.Delete\nc.Update\nd.Search\ne.Exit");

String option = sc.nextLine();

switch (option.charAt(0)) {

case 'a': {

while (true) {

// Add

System.out.println("a. Add a Student");

System.out.println("b. Add a Course");

System.out.println("c. Add a Faculty");

System.out.println("d. Add a student to a existed Course");

System.out.println("e. Add a Course to a existed Faculty");

System.out.println("f. GOTO MAIN MENU");

String option1 = sc.nextLine();

switch (option1.charAt(0)) {

case 'a': {

//add student

Student student = new Student();

prnt("Student ID :");

student.setstudentId(sc.nextInt());

sc.nextLine();

prnt("Student Name:");

student.setstudentName(sc.nextLine());

prnt("Student CGPA:");

student.setstudentCGPA(sc.nextDouble());

sc.nextLine();

studentList.add(student);

prnt("Student ID added successfully\n");

}

break;

case 'b': {

//add course

Course course = new Course();

prnt("Course ID :");

course.setcourseID(sc.nextLine());

prnt("Course Name:");

course.setcourseTitle(sc.nextLine());

prnt("Credit:");

course.setCredit(sc.nextDouble());

sc.nextLine();

courseList.add(course);

prnt("Course ID added successfully\n");

}

break;

case 'c': {

Faculty faculty = new Faculty();

prnt("Faculty ID :");

faculty.setfacultyId(sc.nextInt());

sc.nextLine();

prnt("Faculty Name:");

faculty.setfacultyName(sc.nextLine());

prnt("Faculty Position:");

faculty.setfacultyPosition(sc.nextLine());

facultyList.add(faculty);

prnt("Faculty ID added successfully....\n");

}

break;

case 'd': {

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

prnt("Course ID:");

String cID = sc.nextLine();

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

if (sID == studentList.get(i).getstudentId()) {

for (int j = 0; j < courseList.size(); j++) {

if (cID.equals(courseList.get(j).getcourseID())) {

courseList.get(j).addStudent(studentList.get(i));

prnt("Student data added successfully\n");

}

else{

if( j == courseList.size() - 1)

System.out.println("No Course found for" + cID);

}

}

}

else{

if( i == studentList.size() - 1)

System.out.println("No Student data found for" + sID);

}

}

}

break;

case 'e': {

prnt("Course ID:");

String cID = sc.nextLine();

prnt("Faculty ID");

int fID = sc.nextInt();

sc.nextLine();

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

if (fID == facultyList.get(i).getfacultyId()) {

for (int j = 0; j < courseList.size(); j++) {

if (cID.equals(courseList.get(j).getcourseID())) {

courseList.get(j).addFaculty(facultyList.get(i));

prnt("Course data added successfully\n");

}

else{

if( j == courseList.size() - 1)

System.out.println("No Course data found for" + cID);

}

}

}

else{

if( i == facultyList.size() - 1)

System.out.println("No Faculty data found for" + fID);

}

}

}

break;

case 'f': {

continue x;

}

default: {

System.out.println("No option in character " + option1.charAt(0));

}

}

}

}

case 'b': {

while (true) {

System.out.println("a. Delete a Student");

System.out.println("b. Delete a Course");

System.out.println("c. Delete a Faculty");

System.out.println("d. Delete a Student from a course");

System.out.println("e. Delete a Course from a Faculty");

System.out.println("f. GOTO MAIN MENU");

String option1 = sc.nextLine();

switch (option1.charAt(0)) {

case 'a': {

// delete Student

prnt("Student Id:");

int sID = sc.nextInt();

sc.nextLine();

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

if (sID == studentList.get(i).getstudentId()) {

studentList.remove(i);

prnt("Student remove from data successfully\n");

}

else{

if( i == studentList.size()-1 )

System.out.println("No Course data found for " + sID);

}

}

}

break;

case 'b': {

//Delete course

prnt("Course Id:");

String cID = sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

courseList.remove(i);

prnt("Course remove from data successfully\n");

}

else{

if( i == facultyList.size() )

System.out.println("No Course data found for " + cID);

}

}

}

break;

case 'c': {

//Delete faculty

prnt("Faculty Id:");

int fID = sc.nextInt();

sc.nextLine();

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

if ( fID == facultyList.get(i).getfacultyId() ) {

facultyList.remove(i);

prnt("Faculty remove from data successfully\n");

}

else{

if( i == facultyList.size() )

System.out.println("No Faculty data found for " + fID);

}

}

}

break;

case 'd': {

//Delete a Student from a course

prnt("Course ID:");

String cID = sc.nextLine();

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

courseList.get(i).dropStudent(sID);

prnt("Student remove from the course successfully....\n");

}

else{

if( i == courseList.size()-1 )

System.out.println("No Student data found for" + sID);

}

}

}

break;

case 'e': {

//Delete a Course from a faculty

prnt("Course ID");

String cID = sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

courseList.get(i).dropFaculty();

prnt("Course remove from the faculty successfully....\n");

}

else{

if( i == courseList.size()-1)

System.out.println("No Course data found for " + cID);

}

}

}

break;

case 'f': {

continue x;

}

default: {

System.out.println("No option in character " + option1.charAt(0));

}

}

}

}

case 'c': {

while (true) {

System.out.println("a. Update a Student");

System.out.println("b. Update a Course");

System.out.println("c. Update a Faculty");

System.out.println("d. GOTO MAIN MENU");

String option1 = sc.nextLine();

switch (option1.charAt(0)) {

case 'a': {

//Update Std

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

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

if (sID == studentList.get(i).getstudentId()) {

prnt("Student Name:");

studentList.get(i).setstudentName(sc.nextLine());

prnt("Student ID:");

studentList.get(i).setstudentId(sc.nextInt());

sc.nextLine();

prnt("Student CGPA:");

studentList.get(i).setstudentCGPA(sc.nextDouble());

sc.nextLine();

prnt("Student data updated successfully....\n");

}

else{

if( i == studentList.size()-1 )

System.out.println("No Student data found for " + sID);

}

}

}

break;

case 'b': {

//Update crs

prnt("Course ID:");

String cID = sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

prnt("Student Name:");

studentList.get(i).setstudentName(sc.nextLine());

prnt("Student ID:");

studentList.get(i).setstudentId(sc.nextInt());

prnt("Student CGPA:");

studentList.get(i).setstudentCGPA(sc.nextDouble());

}

else{

if( i == courseList.size() -1)

System.out.println("No Course data found for " + cID);

}

}

}

break;

case 'c': {

//Update faculty

prnt("Faculty ID:");

int fID = sc.nextInt();

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

if (fID == facultyList.get(i).getfacultyId()) {

prnt("Faculty ID :");

facultyList.get(i).setfacultyId(sc.nextInt());

sc.nextLine();

prnt("Faculty Name:");

facultyList.get(i).setfacultyName(sc.nextLine());

prnt("Faculty Position");

facultyList.get(i).setfacultyPosition(sc.nextLine());

}

else{

if( i == facultyList.size()-1 ){

System.out.println("No Faculty found for " + fID);

}

}

}

}

break;

case 'd': {

continue x;

}

default: {

System.out.println("No option in character " + option1.charAt(0));

}

}

}

}

case 'd': {

while (true) {

System.out.println("a. Search a Student");

System.out.println("b. Search a Course");

System.out.println("c. Search a Faculty");

System.out.println("d. Search whether a student takes a course");

System.out.println("e. Search whether a faculty teaches a course");

System.out.println("f. Search courses taken by a student");

System.out.println("g. Search courses taught by a faculty");

System.out.println("h. GOTO MAIN MENU");

String option1 = sc.nextLine();

switch (option1.charAt(0)) {

case 'a': {

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

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

if (sID == studentList.get(i).getstudentId()) {

System.out.println(studentList.get(i).toString());

}

else{

if( i == studentList.size() - 1)

System.out.println("No Student found data for " + sID);

}

}

}

break;

case 'b': {

prnt("Course ID:");

String cID = sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

System.out.println(courseList.get(i).toString());

}

else{

if( i == courseList.size() - 1)

System.out.println("No course data found for " + cID);

}

}

}

break;

case 'c': {

prnt("Faculty ID:");

int fID = sc.nextInt();

sc.nextLine();

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

if (fID == facultyList.get(i).getfacultyId()) {

System.out.println(facultyList.get(i).toString());

}

else{

// If no data found for a faculty

if( i == facultyList.size() - 1)

System.out.println("No faculty data found for " + fID);

}

}

}

break;

case 'd': {

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

prnt("Course ID:");

String cID = sc.nextLine();

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

if (courseList.get(i).getcourseID().equals(cID)) {

for (int j = 0; j < courseList.get(i).getnumberOfStudent(); j++) {

if (sID == courseList.get(i).studentID(j)) {

System.out.println(sID + " takes the " + cID + " course");

}

else {

if( j == courseList.get(i).getnumberOfStudent()-1 )

System.out.println(sID + " doesn't take the " + cID + " course");

}

}

}

else{

if( i == courseList.size()-1)

System.out.println("Course data not found for "+cID);

}

}

}

break;

case 'e': {

prnt("Faculty ID:");

int fID = sc.nextInt();

sc.nextLine();

prnt("Course ID:");

String cID = sc.nextLine();

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

if (cID.equals(courseList.get(i).getcourseID())) {

if (fID == courseList.get(i).getFaculty().getfacultyId()) {

System.out.println(fID + " faculty teaches " + cID + " course");

}

else{

System.out.println(fID + " faculty doesn't teach " + cID + " course");

}

}

else{

if( i == courseList.size()-1)

System.out.println("Course data not found for "+cID);

}

}

}

break;

case 'f': {

prnt("Student ID:");

int sID = sc.nextInt();

sc.nextLine();

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

for (int j = 0; j < courseList.get(i).getnumberOfStudent(); j++) {

if (sID == courseList.get(i).studentID(j)) {

System.out.println("Student " + sID + " takes " + courseList.get(i).getcourseTitle());

}

else{

if( j == courseList.size() - 1 ){

System.out.println("Student data not found for "+sID);

}

}

}

}

}

break;

case 'g': {

prnt("Faculty ID:");

int fID = sc.nextInt();

sc.nextLine();

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

if (fID == courseList.get(i).getFaculty().getfacultyId()) {

System.out.println("Faculty " + fID + " have " + courseList.get(i).getcourseTitle());

}

else{

if( i == courseList.size() - 1)

System.out.println("Faculty data not found for "+fID);

}

}

}

break;

case 'h': {

continue x;

}

default: {

System.out.println("No option in character " + option.charAt(0));

}

}

}

}

case 'e': {

break x;

}

default: {

System.out.println("No option in character " + option.charAt(0));

}

}

}

}

}

