Nome: João Paulo de Oliveira 11611BCC046

4° Aula prática

Uberlândia

2016

1.Código fonte:

* Aluno.java:

package controleacademico;

public class Aluno {

String nome;

GradeCurricular grade;

Historico historico;

Aluno (String n, GradeCurricular g) {

nome = n;

grade = g;

historico = new Historico(this);

}

boolean verificaSeFormou () {

// for (int i = 0; i < grade.nroDisciplinas(); i++) {

for (Disciplina d: grade.disciplinas){

if (! historico.constaAprovacao(d))

return false;

}

return true;

}

}

* ControleAcademico.java:

package controleacademico;

import java.util.\*;

public class ControleAcademico {

public static void main(String[] args) {

Disciplina pp = new Disciplina("PP");

Disciplina aed1 = new Disciplina("AED1");

Disciplina poo1 = new Disciplina("POO1");

Disciplina poo2 = new Disciplina("POO2");

poo2.criaPrerequisito(poo1);

GradeCurricular grade = new GradeCurricular();

grade.incluiDisciplina(pp);

//pp.incluiNaGrade(grade); Seria um bad smell: feature envy

grade.incluiDisciplina(aed1);

grade.incluiDisciplina(poo1);

grade.incluiDisciplina(poo2);

Professor profMarcelo = new Professor("Marcelo", "FACOM");

Aluno alunoSeverino = new Aluno("Severino", grade);

ArrayList<Disciplina> falta = alunoSeverino.historico.AindaFalta(grade);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

OfertaTurma ofertaPP = new OfertaTurma(2015, Semestre.SEGUNDO, pp);

alunoSeverino.historico.matriculaDisciplina(ofertaPP);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

falta = alunoSeverino.historico.AindaFalta(grade);

OfertaTurma ofertaPoo2 = new OfertaTurma(2016,Semestre.PRIMEIRO,poo2);

alunoSeverino.historico.matriculaDisciplina(ofertaPoo2);

OfertaTurma ofertaAED1 = new OfertaTurma(2016, Semestre.PRIMEIRO, aed1);

alunoSeverino.historico.matriculaDisciplina(ofertaAED1);

OfertaTurma ofertaPOO1 = new OfertaTurma(2016, Semestre.PRIMEIRO, poo1);

alunoSeverino.historico.matriculaDisciplina(ofertaPOO1);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

falta = alunoSeverino.historico.AindaFalta(grade);

alunoSeverino.historico.registraResultado(pp, 60);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

falta = alunoSeverino.historico.AindaFalta(grade);

alunoSeverino.historico.registraResultado(aed1, 60);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

falta = alunoSeverino.historico.AindaFalta(grade);

alunoSeverino.historico.registraResultado(poo1, 30);

System.out.println("O aluno " + alunoSeverino.nome + " esta' " + (alunoSeverino.verificaSeFormou() ? "formado!" : "ainda cursando"));

falta = alunoSeverino.historico.AindaFalta(grade);

}

}

* Disciplina.java:

package controleacademico;

import java.util.\*;

public class Disciplina {

String nome;

ArrayList<Disciplina> prerequisitos = new ArrayList<Disciplina>();

Disciplina (String nome) {

this.nome = nome;

}

void criaPrerequisito (Disciplina d) {

prerequisitos.add(d);

}

String getNome() {

return nome;

}

}

* GradeCurricular.java:

package controleacademico;

import java.util.Vector;

public class GradeCurricular {

Vector<Disciplina> disciplinas = new Vector<Disciplina>();

void incluiDisciplina (Disciplina d) {

disciplinas.add(d);

}

void removeDisciplina (Disciplina d) {

disciplinas.remove(d);

}

int nroDisciplinas() {

return disciplinas.size();

}

}

* Historico.java:

package controleacademico;

import java.util.\*;

public class Historico {

ArrayList<MatriculaAluno> disciplinasMatriculadas;

Aluno aluno;

Historico (Aluno a) {

aluno = a;

disciplinasMatriculadas = new ArrayList<MatriculaAluno>();

}

void matriculaDisciplina (OfertaTurma oferta) {

boolean podeCursar = true;

for(int i=0;i<oferta.disciplina.prerequisitos.size();i++){

if(!this.constaAprovacao(oferta.disciplina.prerequisitos.get(i))){

podeCursar = false;

}

}

if(podeCursar == false){

System.out.println("Não atendeu o pré-requesito na matéria "+oferta.nomeDisciplina());

oferta.cancelaMatricula(aluno);

return;

}

MatriculaAluno m = new MatriculaAluno (aluno, oferta);

this.disciplinasMatriculadas.add(m);

oferta.matriculaAluno(m);

}

boolean constaAprovacao (Disciplina d) {

for (MatriculaAluno m: disciplinasMatriculadas) {

if (m.aprovado(d))

return true;

}

return false;

}

void registraResultado (Disciplina d, int nota) {

for (MatriculaAluno m: disciplinasMatriculadas) {

if (m.oferta.nomeDisciplina().equals(d.nome) && m.status == StatusMatriculaDisciplina.CURSANDO)

m.registraNota(nota);

}

}

ArrayList<Disciplina> AindaFalta(GradeCurricular grade){

ArrayList<Disciplina> faltantes = new ArrayList<Disciplina>();

for(int i=0,j=0;i<grade.disciplinas.size();i++){

if(!this.constaAprovacao(grade.disciplinas.get(i))){

faltantes.add(grade.disciplinas.get(i));

System.out.println("falta "+faltantes.get(j++).nome);

}

}

return faltantes;

}

}

* MatriculaAluno.java:

package controleacademico;

public class MatriculaAluno {

Aluno aluno;

OfertaTurma oferta;

StatusMatriculaDisciplina status;

float nota;

MatriculaAluno (Aluno a, OfertaTurma o) {

aluno = a;

oferta = o;

status = StatusMatriculaDisciplina.CURSANDO;

}

Aluno alunoMatriculado () {

return aluno;

}

OfertaTurma turmaMatriculada() {

return oferta;

}

void registraNota (float n) {

this.nota = n;

if (this.nota >= 60)

status = StatusMatriculaDisciplina.APROVADA;

else

status = StatusMatriculaDisciplina.REPROVADA;

}

void trancaMatricula () {

this.status = StatusMatriculaDisciplina.TRANCADA;

}

boolean aprovado (Disciplina d) {

return d.getNome().equals(oferta.nomeDisciplina()) && status == StatusMatriculaDisciplina.APROVADA;

}

}

* OfertaTurma.java:

package controleacademico;

import java.util.\*;

public class OfertaTurma {

int ano;

Semestre semestre;

Disciplina disciplina;

Vector<MatriculaAluno> alunosMatriculados;

Professor professor;

OfertaTurma (int ano, Semestre s, Disciplina d) {

this.ano = ano;

this.semestre = s;

this.disciplina = d;

alunosMatriculados = new Vector<MatriculaAluno>();

}

void matriculaAluno (MatriculaAluno m) {

alunosMatriculados.add(m);

}

void cancelaMatricula (Aluno a) {

alunosMatriculados.remove(a);

}

void vinculaProfessor (Professor p) {

professor = p;

professor.ficaAlocadoTurma(this);

}

String nomeDisciplina() {

return this.disciplina.getNome();

}

}

* Professor.java:

package controleacademico;

import java.util.\*;

public class Professor {

String nome;

String departamento;

ArrayList<OfertaTurma> turmasOfertadas;

Professor (String nome, String d) {

this.nome = nome;

departamento = d;

}

void ficaAlocadoTurma (OfertaTurma o) {

turmasOfertadas.add(o);

o.vinculaProfessor(this);

}

}

* Semestre.java:

package controleacademico;

public enum Semestre {

PRIMEIRO, SEGUNDO;

}

* StatusMatriculaDisciplinas.java:

package controleacademico;

public enum StatusMatriculaDisciplina {

CURSANDO, TRANCADA, REPROVADA, APROVADA;

}

2.Print do funcionamento:

