TEMA GSDC

Curiban Simona

**Clasa abstracta echipa:**

**import** java.lang.reflect.Member;

**import** java.util.ArrayList;

**import** javax.swing.JOptionPane;

**public** **abstract** **class** Team{

**private** String nume;

**private** String lider;

**int** ani\_experienta;

**public** Team(String num, String lid, **int** YhrExp) {

**this**.nume=num;

**this**.lider=lid;

}

**public** String getNume(){

**return** nume;

}

**public** **abstract** **int** getNr\_Max();

**public** **void** setYrsExp(**int** ani\_experienta) {

**this**.ani\_experienta=ani\_experienta;

}

**public** String getLider() {

**return** lider;

}

**public** String toString1() {

**return** nume+ " "+ lider+ " "+getNr\_Max();

}

**public** **boolean** isInTeam(Strategy strategy) {

**for**(Member m :membru) {

**if**(strategy.isCondition(m)) {

**return** **true**;

}

}

**return** **false**;

}

**public** **boolean** setLider() {

**for**(Member m:membru) {

**if**(ani\_experienta(m)>=5) {

**return** **true**;

}

}

**return** **false**;

}

**protected** **abstract** **int** ani\_experienta(Member m);

}

**Clasa membrilor:**

**import** java.util.ArrayList;

**public** **class** Member {

**private** String nume;

**private** String prenume;

**private** **int** varsta;

**private** **int** salariu;

**private** **int** ani\_experienta;

**private** **int** lider;

**private** ArrayList<Member>membru=**new** ArrayList<Member>();

//add members

**public** **void** addMember(Member m) {

**if**(!membru.contains(m)) {

membru.add(m);

}

**else** {

System.***out***.println("Member"+ m+ "is already in the team.");

}

}

**public** String toString() {

String S="";

**for**(**int** i=1;i<membru.size();i++) {

S=S+membru + "\n";

}

**return** S;

}

**public** **boolean** isInTeam(Strategy strategy) {

**for**(Member m :membru) {

**if**(strategy.isCondition(m)) {

**return** **true**;

}

}

**return** **false**;

}

**public** **static** **int**[] removeElement(**int**[] arr, **int** index) {

**if** (arr == **null** || index < 0

|| index >= arr.length) {

**return** arr;

}

**int**[] anotherArray = **new** **int**[arr.length - 1];

**for** (**int** i = 0, k = 0; i < arr.length; i++) {

**if** (i == index) {

**continue**;

}

anotherArray[k++] = arr[i];

}

**return** anotherArray;

}

**public** String toString2() {

**return** lider+ " "+ membru;

}

**public** Object getLider() {

// **TODO** Auto-generated method stub

**return** **null**;

}

}

**Echipa Dev:**

**import** java.lang.reflect.Member;

**public** **class** Dev **extends** Team {

**private** **int** ani\_experienta;

**public** Dev(String num, String lid) {

**super**(num, lid);

// **TODO** Auto-generated constructor stub

**public** setYrsExp(**int** ani\_experienta) {

**this**.ani\_experienta=ani\_experienta;

}

**public** **double** getCost() {

**if**(lid) {

**return** 2500+(2500+ani\_experienta\*250);

}

**else** { **if**(ani\_experinta<2){

**return** 1500;

}

**if**(ani\_experienta>2 && ani\_experienta<5) {

**return** 1500+374;

}

**if**(ani\_experienta>5) {

**return** 1500+750;

}

}

}

}

@Override

**public** **int** getNr\_Max() {

// **TODO** Auto-generated method stub

**return** 0;

}

@Override

**protected** **int** ani\_experienta(Member m) {

// **TODO** Auto-generated method stub

**return** 0;

}

**Echipa HR:**

**import** java.lang.reflect.Member;

**public** **class** HR **extends** Team {

**private** **int** ani\_experienta=0;

**public** HR(String num, String lid) {

**super**(num, lid,9);

}

**public** **void** setYrsExp(**int** ore\_experienta) {

**this**.ani\_experienta=ani\_experienta;

}

**public** **double** getCost() {

**if**(lid) {

**return** 1350+(ani\_experienta\*300);

}

**else** { **if**(ani\_experienta<2){

**return** 1000;

}

**if**(ani\_experienta>2 && ani\_experienta<5) {

**return** 1000+250;

}

**if**(ani\_experienta>5) {

**return** 1000+500;

}

}

}

@Override

**public** **int** getNr\_Max() {

// **TODO** Auto-generated method stub

**return** 0;

}

@Override

**protected** **int** ani\_experienta(Member m) {

// **TODO** Auto-generated method stub

**return** 0;

}

}

**public** **interface** Strategy {

**boolean** isCondition(java.lang.reflect.Member m);

}

**public** **class** Lider **implements** Strategy {

**private** String lider;

**public** Lider(String lider) {

**this**.lider=lider;

}

**public** **boolean** isCondition(Member member) {

**return** lider.equals(member.getLider());

}

**public** String getLider() {

**return** lider;

}

}

**Main:**

**import** java.lang.reflect.Member;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Dev a1=**new** Dev("Dan", "Mihai");

Dev a3=**new** Dev("Robert", "Adrian");

Dev a5=**new** Dev("Sergiu","Matei");

HR a2=**new** HR("Radu", "Alin");

HR a4=**new** HR("Razvan", "Raul");

Member p1;

//adaugare angajati

a1.addMember(p1);

Member p2;

a2.addMember(p2);

Member p3;

a3.addMember(p3);

Member p4;

a4.addMember(p4);

Member p5;

a5.addMember(p5);

Lider l1=**new** Lider("Alex");

**if**(a1.isInTeam(l1)) {

System.***out***.println("\n"+l1.getLider()+ "is part of the team.");

}

**else** {

System.***out***.println("\n"+ l1.getLider()+ "is not part of the team.");

}

}

Team t1=**new** Team();

}