Benim projem GÖREV TAKIBI projedir.

Şirket, görev , personel , ve main sınıfları oluşturulmuştur.

* Şirket sınıfında ( fsm) olarak tanımlanmıştır (her hangi bi örnek olur )

Onun içinde birimler dizisi static ve 5 tane( birimdekiBitmisGorevler – birimdekiBitmemisGorevler – tumBitmisGorevler – tumBitmemisGorevler – tumCalısanlar ( bulmak için metotdu tınmımlanmıştır.

import java.util.ArrayList;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author Muaz Ahmed

\*/

public class Sirket {

static ArrayList<Birim> Birimler = new ArrayList<Birim>();

Gorev birimdekiBitmisGorevler(Birim b) {

Gorev g = new Gorev();

for (Personel personel : b.getPersoneller()) {

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

if (personel.gorevler.get(i).gorevDurumu == 2) {

g = personel.gorevler.get(i);

System.out.println(personel.gorevler.get(i).gorevAdi + " görevi bitti");

}

}

}

return g;

}

/\*

- you have to call getPersoneller function to continue

- and so you will create an Array in the test class to send it to the setter

- check this out

\*/

Gorev birimdekiBitmemisGorevler(Birim b) {

Gorev g = new Gorev();

for (Personel personel : b.getPersoneller()) {

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

if (personel.gorevler.get(i).gorevDurumu == 1) {

g = personel.gorevler.get(i);

System.out.println(personel.gorevler.get(i).gorevAdi + " görevi bitmedi");

}

}

}

return g;

}

Gorev tumBitmisGorevler() {

Gorev g = new Gorev();

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

for (int j = 0; j < Sirket.Birimler.get(i).getPersoneller().length; j++) {

for (int k = 0; k < Sirket.Birimler.get(i).getPersoneller()[j].gorevler.size(); k++) {

if(Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k).gorevDurumu==2){

g=Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k);

System.out.println(Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k).gorevAdi+" görevi bitti");

}

}

}

}

return g;

}

Gorev tumBitmemisGorevler() {

Gorev g = new Gorev();

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

for (int j = 0; j < Sirket.Birimler.get(i).getPersoneller().length; j++) {

for (int k = 0; k < Sirket.Birimler.get(i).getPersoneller()[j].gorevler.size(); k++) {

if(Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k).gorevDurumu==1){

g=Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k);

System.out.println(Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k).gorevAdi+" görevi bitti");

}

}

}

}

return g;

}

//bu sırkette tum calısan personel kimdir

void tumCalısanlar(Birim b)

{

Personel p;

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

for (int j = 0; j < Sirket.Birimler.get(i).getPersoneller().length; j++) {

// b.isim=Sirket.Birimler.get(i).adi;

p=Sirket.Birimler.get(i).getPersoneller()[j];

System.out.println(Sirket.Birimler.get(i).getPersoneller()[j]+" "+Sirket.Birimler.get(i)+" de çalışıyor");

}

}

}

}

* görev sınıfında ( String gorevAdi int gorevDurumu ) ve ( gorevDurumu – gorevKimin – gorevinBirimiNedir – gorevBaslatmak – gorevBitirmek ) diye 5 tane metot tınmımlanmıştır.

public class Gorev {

String gorevAdi;

int gorevDurumu;

Gorev() {

this.gorevDurumu = 0;

}

// Bir görevv şu anda durumu ne olduğunu gösteren bi metottur

void gorevDurumu() {

if (gorevDurumu == 0) {

System.out.println(this.gorevAdi + " görevi daha Alınmamıştır");

} else if (gorevDurumu == 1) {

System.out.println(this.gorevAdi + " görevi verilmiştir");

} else if (gorevDurumu == 2) {

System.out.println(this.gorevAdi + " görevi bitmiştir");

}

}

// Bir görev kime ait peersonellerden öğrenebilmek

Personel[] gorevKimin(Gorev g) {

Personel p []=new Personel[10];

int kaacTanePersonel=0;

for (int k = 0; k < Sirket.Birimler.size(); k++) {

for (int S = 0; S < Sirket.Birimler.get(k).getPersoneller().length; S++) {

// personel.gorevler.contains(g);

for (int i = 0; i < Sirket.Birimler.get(k).getPersoneller()[S].gorevler.size(); i++) {

if (g.gorevDurumu == 1) {

if (Sirket.Birimler.get(k).getPersoneller()[S].gorevler.get(i)== g) {

p[kaacTanePersonel] = Sirket.Birimler.get(k).getPersoneller()[S];

kaacTanePersonel++;

System.out.println(g.gorevAdi + " görevi " + p + " personele aittir.");

}

}

else if(g.gorevDurumu==0)

System.out.println(g.gorevAdi+" zaten kimseye verilmemiştir ");

else

System.out.println(g.gorevAdi+" görevi zaten bitmiş");

}

}

}

return p;

}

/\*

- Sirket--> static <>birimler --> (private)[]Personelleri --> <>gorevleri --> gorevinAdi == g.gorevi

-

\*/

// Bir görev hangi birimde oldğunu gösteren metottur

// dizi donecek

Birim[] gorevinBirimiNedir(Gorev g) {

int birimSayisi=0;

Birim []b = new Birim[10];

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

for (int j = 0; j < Sirket.Birimler.get(i).getPersoneller().length; j++) {

for (int k = 0; k < Sirket.Birimler.get(i).getPersoneller()[j].gorevler.size(); k++) {

if (Sirket.Birimler.get(i).getPersoneller()[j].gorevler.get(k) == g) {

b[birimSayisi]=Sirket.Birimler.get(i);

birimSayisi++;

System.out.println(g.gorevAdi + " görevi " + b[birimSayisi].adi + " biirime aittir.");

}

}

}

}

return b;

}

/\*

bu 2 metotdun Personeli belli olmadığı durumunda aktif görevleri (yani görevdurumu ==1 ) baslatmak ve bıtırmek gösteren metotlardır

\*/

// Bir görev başlatmak için

void gorevBaslatmak(Birim b) {

if (this.gorevDurumu == 1) {

System.out.println(this.gorevAdi + " görevi zaten başladı");

} else if (this.gorevDurumu == 0) {

this.gorevDurumu = 1;

System.out.println(this.gorevAdi + " görevi başladı");

} else {

System.out.println(this.gorevAdi + " görevi bitirdi");

}

}

// Bir görev bitirmek için

void gorevBitirmek(Birim b) {

if (this.gorevDurumu == 1) {

this.gorevDurumu = 2;

System.out.println( this.gorevAdi + " bitirdi");

} else if (this.gorevDurumu == 0) {

System.out.println(this.gorevAdi + " görevi hala Başlamadı");

} else {

System.out.println(this.gorevAdi + " görevi zaten bitirdi");

}

}

}

* Personel sınıfında (ArrayList gorevler ) ve (personelinGorevleri – gorevVermek – PersonelinbirimiNedir – gorevBaslatmak – gorevBitirmek ) diye 5 tane metot tınmımlanmıştır.

import java.util.ArrayList;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author Muaz Ahmed

\*/

public class Personel {

String isim;

//private olsaydi daha iyi olurdu

// neden Personel objeyi oluşturuken Arraylıste çıkıyor ( kopyası alıyorsa )

// String isimi da alması gerekmiyor mu ??

ArrayList<Gorev> gorevler = new ArrayList<Gorev>();

Personel() {

}

/\*

- you will create an Array of Personel even if it of 1 element

- and so you will choose the second option

\*/

Personel(Birim b) {

b.setPersoneller(this);

}

// Bir personelin gorevleri nedir

void personelinGorevleri(Personel p) {

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

System.out.println(p.isim + " in görevleri :");

if (p.gorevler.get(i).gorevDurumu == 1) {

System.out.println(" " + p.gorevler.get(i).gorevAdi + " görevi bitmemiştir");

} else if (p.gorevler.get(i).gorevDurumu == 2) {

System.out.println(" " + p.gorevler.get(i).gorevAdi + " görevi bitmiştir");

}

}

}

// Bir personelin görevlendirmesi

void gorevVermek(Gorev g) {

this.gorevler.add(g);

}

/\*

bu 2 metotdun Personeli belli olduğu durumunda aktif görevleri (yani görevdurumu ==1 ) baslatmak ve bıtırmek gösteren metotlardır

\*/

// Bir personel nerede çalıştığı öğrenmek için

Birim PersonelinbirimiNedir(Personel p) {

Birim b = new Birim();

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

for (int j = 0; j < Sirket.Birimler.get(i).getPersoneller().length; j++) {

if (Sirket.Birimler.get(i).getPersoneller()[j] == p) {

b = Sirket.Birimler.get(i);

System.out.println(Sirket.Birimler.get(i).getPersoneller()[j].isim + " " + b.adi + "bölümünde çalışıyor");

}

}

}

return b;

}

/\*

bu 2 metotdun Personeli belli olduğu durumunda aktif görevleri (yani görevdurumu ==1 ) baslatmak ve bıtırmek gösteren metotlardır

\*/

void gorevBaslatmak(Gorev g) {

int i;

for (i = 0; i < this.gorevler.size(); i++) {

if(this.gorevler.get(i)==g){

if(this.gorevler.get(i).gorevDurumu==1)

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi zaten başladı");

else if(this.gorevler.get(i).gorevDurumu==0){

this.gorevler.get(i).gorevDurumu=1;

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi başladı");

}

else

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi bitirdi");

}

}

}

void gorevBitirmek(Gorev g) {

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

if(this.gorevler.get(i)==g){

if(this.gorevler.get(i).gorevDurumu==1){

this.gorevler.get(i).gorevDurumu=2;

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi bitirdi");

}

else if(this.gorevler.get(i).gorevDurumu==0){

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi hala Başlamadı");

}

else

System.out.println(this.isim + this.gorevler.get(i).gorevAdi + " görevi zaten bitirdi");

}

}

}

}

* Main sınıfında (Sirket fsm , Birim muhendislik - guzelSanatlar - mimarlik - turkDiliVeEdebbiyeti - Adelet , Gorev g1- g2 – g3 – g4 – g5 – g6 – g7 – g8 – g9 - g10 , Personel p1 – p2 – p3 – p4 – p5 – p6 – p7 – p8 – p9 – p10 – p11 – p12 – p13 – p14 - p15 – p16 – p17 – p18 – p19 – p20 ) olarak tınmımlanmıştır.

//import java.util.ArrayList;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*

\*

\* @author Muaz Ahmed

\*/

public class Birim extends Personel {

// Personel arrayın length e bi ArrayListin sizeyi gönderebiliriz

private Personel[] personeller=new Personel[10];

String adi;

int personelAdidi=0 ;

// Burada mantıksal hata vardır.

// Personel constructor metottan aldığı Personeli setPersonele gönderip

// Personel dizisine eklemesinide hata oluşuyor

public void setPersoneller(Personel personel) {

this.personeller[personelAdidi] = personel;

personelAdidi++;

}

public Personel[] getPersoneller() {

return personeller;

}

}