LABORATOR 4:

import java.io.\*;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Main

{

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

int optiune;

List<echipament> echipamente = new ArrayList<echipament>();

citire\_fisier(echipamente);

do

{

meniu();

optiune = scanner.nextInt();

switch ( optiune ) {

case 0:

System.out.println("La revedere!");

break;

case 1:

afisare(echipamente);

break;

case 2:

afisare\_Imprimanta(echipamente);

break;

case 3:

afisare\_Copiatoar(echipamente);

break;

case 4:

afisare\_Sistem\_calcul(echipamente);

break;

case 5:

modificare\_situatie(echipamente);

break;

case 6:

modificare\_mod\_scriere(echipamente);

break;

case 7:

modificare\_formatCopiere(echipamente);

break;

case 8:

modificare\_sistem\_operare(echipamente);

break;

case 9:

afisare\_echipamente\_vandute(echipamente);

break;

case 10:

serializare\_deserializare( echipamente);

break;

default:

System.out.println("Opțiune invalida!");

break;

}

}while( optiune != 0);

}

public static void meniu()

{

System.out.println("MENIU");

System.out.println("0.Iesire");

System.out.println("1.Afisarea tuturor echipamentelor");

System.out.println("2.Afişarea imprimantelor");

System.out.println("3.Afişarea copiatoarelor");

System.out.println("4.Afisarea sistemelor de calcul");

System.out.println("5.Modificarea situatiei unui echipament");

System.out.println("6.Modificarea modului de scriere al unei imprimante");

System.out.println("7.Modificarea unui format de copiere pentru un copiator");

System.out.println("8.Modificarea unui sistem de operare pentru un anumit sistem de calcul");

System.out.println("9.Afisarea echipamentelor vandute");

System.out.println("10.Serializare/ deserializare");

System.out.println("Dati optiunea: ");

}

public static void citire\_fisier(List<echipament> echipamente)

{

String denumire;

int numar\_inventar;

float pret;

String zona\_magazin;

situatie\_produs situatie;

int situatieInt;

int tip;

try

{

Scanner scanner = new Scanner(new File("src/date.in"));

while(scanner.hasNext())

{

tip = scanner.nextInt();

denumire = scanner.next();

numar\_inventar = scanner.nextInt();

pret = scanner.nextFloat();

zona\_magazin = scanner.next();

situatieInt = scanner.nextInt();

situatie = situatie\_produs.values()[situatieInt];

if(tip == 0)

{

int ppm;

int dpi;

int p\_car;

boolean color;

ppm = scanner.nextInt();

dpi = scanner.nextInt();

p\_car = scanner.nextInt();

color = scanner.nextBoolean();

echipament e = new imprimanta(denumire,numar\_inventar,pret,zona\_magazin,situatie,ppm,dpi,p\_car,color);

echipamente.add(e);

}

else if(tip == 1)

{

int p\_ton;

int formatInt;

formate format;

p\_ton = scanner.nextInt();

formatInt = scanner.nextInt();

format = formate.values()[formatInt];

echipament e = new copiator(denumire,numar\_inventar,pret,zona\_magazin,situatie,p\_ton,format);

echipamente.add(e);

}

else if(tip == 2)

{

String tip\_mon;

double vit\_proc;

int c\_hdd;

int sistemOperareInt;

sistem\_op sistem\_operare;

tip\_mon = scanner.next();

vit\_proc = scanner.nextDouble();

c\_hdd = scanner.nextInt();

sistemOperareInt = scanner.nextInt();

sistem\_operare = sistem\_op.values()[sistemOperareInt];

echipament e = new sisteme\_calcul(denumire,numar\_inventar,pret,zona\_magazin,situatie,tip\_mon,vit\_proc,c\_hdd,sistem\_operare);

echipamente.add(e);

}

}

} catch (FileNotFoundException e)

{

throw new RuntimeException(e);

}

}

public static void afisare(List<echipament> echipamente)

{

System.out.println("Echipamentele: ");

for(echipament ech: echipamente)

{

System.out.println(ech);

}

}

public static void afisare\_Imprimanta(List<echipament> echipamente)

{

System.out.println("Echipamentele: ");

for(echipament ech: echipamente)

{

if(ech instanceof imprimanta)

System.out.println(ech);

}

}

public static void afisare\_Copiatoar(List<echipament> echipamente)

{

System.out.println("Echipamentele: ");

for(echipament ech: echipamente)

{

if(ech instanceof copiator)

System.out.println(ech);

}

}

public static void afisare\_Sistem\_calcul(List<echipament> echipamente)

{

System.out.println("Echipamentele: ");

for(echipament ech: echipamente)

{

if(ech instanceof sisteme\_calcul)

System.out.println(ech);

}

}

public static void modificare\_situatie(List<echipament> echipamente)

{

Scanner scanner = new Scanner(System.in);

String den;

System.out.println("Dati denumirea echipamentului la care doriti sa modificati situatia: ");

den = scanner.next();

int noua\_SituatieInt;

situatie\_produs noua\_situatie;

for(echipament ech: echipamente)

{

if(den.equals(ech.getDenumire()))

{

System.out.println("Dati noua situatie (0->achizitionat/ 1->expus/ 2->vandut): ");

noua\_SituatieInt = scanner.nextInt();

noua\_situatie = situatie\_produs.values()[noua\_SituatieInt];

ech.setSituatie(noua\_situatie);

}

}

}

public static void modificare\_mod\_scriere(List<echipament> echipamente)

{

Scanner scanner = new Scanner(System.in);

String den;

System.out.println("Dati denumirea imprimantei la care doriti sa modificati formatul: ");

den = scanner.next();

boolean noul\_modColor\_Necolor;

for(echipament ech: echipamente)

{

if((ech instanceof imprimanta) && den.equals(ech.getDenumire()))

{

System.out.println("Schimbati modul de tiparire: true(color)/ false(alb-negru): ");

noul\_modColor\_Necolor = scanner.nextBoolean();

imprimanta imp = (imprimanta) ech;

imp.setColor(noul\_modColor\_Necolor);

break;

}

}

}

public static void modificare\_formatCopiere(List<echipament> echipamente)

{

Scanner scanner = new Scanner(System.in);

String den;

System.out.println("Dati denumirea imprimantei la care doriti sa modificati formatul: ");

den = scanner.next();

int newFormatInt;

formate newFormat;

for(echipament ech: echipamente)

{

if((ech instanceof copiator) && den.equals(ech.getDenumire()))

{

copiator cop = (copiator) ech;

System.out.println("Dati noul format de copiere (0->A3/ 1->A4): ");

newFormatInt = scanner.nextInt();

newFormat = formate.values()[newFormatInt];

cop.setFormat(newFormat);

}

}

}

public static void modificare\_sistem\_operare(List<echipament> echipamente)

{

Scanner scanner = new Scanner(System.in);

String den;

System.out.println("Dati denumirea sistemului de calcul la care doriti sa modificati sistemul de operare: ");

den = scanner.next();

int new\_sist\_op;

sistem\_op new\_sistem;

for(echipament ech: echipamente)

{

if((ech instanceof sisteme\_calcul) && den.equals(ech.getDenumire()))

{

sisteme\_calcul sist = (sisteme\_calcul) ech;

System.out.println("Dati noul sistem de operare (0->Windows/ 1->Linux): ");

new\_sist\_op = scanner.nextInt();

new\_sistem = sistem\_op.values()[new\_sist\_op];

sist.setSistemOperare(new\_sistem);

}

}

}

public static void afisare\_echipamente\_vandute(List<echipament> echipamente)

{

for(echipament ech: echipamente)

if(ech.getSituatie() == situatie\_produs.values()[2])

{

System.out.println(ech);

}

}

public static void serializare\_deserializare(List<echipament> echipamente)

{

try

{

FileOutputStream f = new FileOutputStream("src/echip.bin");

ObjectOutputStream oos = new ObjectOutputStream(f);

for (echipament ech : echipamente) {

oos.writeObject(ech);

}

oos.close();

f.close();

}

catch (IOException e) {

e.printStackTrace();

}

}

}

COPIATOR:

import java.io.Serializable;

public class copiator extends echipament implements Serializable

{

private int p\_ton;

private formate format; // A3 sau A4

public copiator(String denumire, int numar\_inventar, float pret, String zona\_magazin, situatie\_produs situatie, int p\_ton, formate format)

{

super(denumire, numar\_inventar, pret, zona\_magazin, situatie);

this.p\_ton = p\_ton;

this.format = format;

}

@Override

public String toString() {

String m = mesaj();

return m+"\nPagini per toner: "+ this.p\_ton+"\nFormat: "+this.format;

}

public void setFormat(formate f)

{

this.format = f;

}

}

SISTEME CALCUL

import java.io.Serializable;

public class sisteme\_calcul extends echipament implements Serializable

{

private String tip\_mon;

private double vit\_proc;

private int c\_hdd;

private sistem\_op sistemOperare;

public sisteme\_calcul(String denumire, int numar\_inventar, float pret, String zona\_magazin, situatie\_produs situatie, String tip\_mon, double vit\_proc, int c\_hdd, sistem\_op sistemOperare)

{

super(denumire, numar\_inventar, pret, zona\_magazin, situatie);

this.tip\_mon = tip\_mon;

this.vit\_proc = vit\_proc;

this.c\_hdd = c\_hdd;

this.sistemOperare = sistemOperare;

}

@Override

public String toString()

{

String m = mesaj();

return m+"\nMonitor: "+ tip\_mon+"\nViteza procesare: "+this.vit\_proc +"\nCapacitate HDD: "+c\_hdd+"\nSistem de operare: "+this.sistemOperare;

}

public void setSistemOperare(sistem\_op SO)

{

this.sistemOperare = SO;

}

}

DATE

0 HP\_Smart\_Tank 1 725 Flanco 1 3 100 25 true

1 Pantum 2 600 Media\_galaxy 2 15 0

2 Lenovo 3 3000 Emag 2 Samsung 125000 1024 0

0 HP\_OfficeJet\_Pro 4 550 Internet 0 4 75 15 false

1 Epson\_EcoTank 5 2000 OLX 1 27 1

ECHIPAMENTE

import java.io.Serializable;

public class echipament implements Serializable

{

protected String denumire;

protected int numar\_inventar;

protected float pret;

protected String zona\_magazin;

protected situatie\_produs situatie;

public echipament(String denumire, int numar\_inventar, float pret, String zona\_magazin, situatie\_produs situatie)

{

this.denumire = denumire;

this.numar\_inventar = numar\_inventar;

this.pret = pret;

this.zona\_magazin = zona\_magazin;

this.situatie = situatie;

}

public String getDenumire()

{

return this.denumire;

}

public int getNumar\_inventar()

{

return this.numar\_inventar;

}

public float getPret()

{

return this.pret;

}

public situatie\_produs getSituatie()

{

return this.situatie;

}

public void setSituatie(situatie\_produs s)

{

this.situatie= s;

}

public String mesaj()

{

return "\nDenumire: "+this.denumire+"\nNumar inventar: "+this.numar\_inventar+"\nPret:"+this.pret+"\nZona magazin: "+this.zona\_magazin+"\nSituatie produs: "+this.situatie;

}

}

IMPRIMANTA

import java.io.Serializable;

public class imprimanta extends echipament implements Serializable

{

private int ppm;

private int dpi;

private int p\_car;

private boolean color;

public imprimanta(String denumire, int numar\_inventar, float pret, String zona\_magazin, situatie\_produs situatie, int ppm,int dpi,int p\_car,boolean color)

{

super(denumire, numar\_inventar, pret, zona\_magazin, situatie);

this.ppm = ppm;

this.dpi = dpi;

this.p\_car = p\_car;

this.color = color;

}

public int getPPM() {

return ppm;

}

public int getDPI() {

return dpi;

}

public int getPCar() {

return p\_car;

}

public boolean isColor() {

return color;

}

public void setColor(boolean C)

{

this.color = C;

}

@Override

public String toString() {

String m = mesaj();

return m+"\nPagini/ minut: "+ this.ppm + "\nDPI: " + this.dpi+"\nPagini/ cartus: "+ this.p\_car+"\nColor: "+this.color;

}

}

SITUATIE PRODUS

enum situatie\_produs{achizitionat, expus, vandut}

enum formate{A3, A4}

enum sistem\_op{Windows, Linux}