**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG THÀNH PHỐ HỒ CHÍ MINH**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**

Ảnh có chứa biểu đồ

Mô tả được tạo tự động

**BÁO CÁO ĐỒ ÁN MÔN LẬP TRÌNH HĐT**

**Quản Lý Kho Máy Ảnh**

Lớp: 10DH-CNTT5  
Nhóm : 8

Thành viên của nhóm:

1.Nguyễn Thanh Bình......... (1050080171)

2.Trần Thái Thiên................ (1050080202)

3.Nguyễn Quang Minh........ (1050080184)

***TP. Hồ Chí Minh, tháng 04 năm 2023***

1. **Phát biểu bài toán**

Chương trình quản lý kho máy ảnh là một phần mềm được thiết kế để giúp quản lý và tổ chức các máy ảnh, thiết bị và phụ kiện liên quan trong một kho lưu trữ hoặc cửa hàng. Chương trình này có thể giúp các cửa hàng, đại lý máy ảnh hay các nhà sản xuất máy ảnh quản lý các sản phẩm của mình một cách chuyên nghiệp và hiệu quả hơn.Một hệ thống quản lý kho máy ảnh tốt sẽ giúp cho cửa hàng có thể dễ dàng quản lý, theo dõi và tối ưu hóa quá trình nhập xuất sản phẩm giảm thiểu hàng tồn kho và đảm bảo cung ứng đầy đủ các sản phẩm cho khách hàng

Các đối tượng cần quản lý gồm:

-Máy ảnh: gồm các thuộc tính như tên, mã, thương hiệu, màu sắc, pin kích thước…

-Compacct: máy ảnh point and shot kế thừa các thuộc thuộc tính của máy ảnh

là tên, mã, thương hiệu, pin,…

-Mirroless máy ảnh không gương lật kế thừa các thuộc tính của máy ảnh là tên , mã, thương hiệu….

-DSRL: máy ảnh gương lật kế thừa các thuộc tính của máy ảnh là tên, mã, thương hiệu,….

-Bridgee: máy ảnh zoom kế thừa các thuộc tính của máy là tên, mã,…

-Quản lý kho: quản lý thông tin nhâp, sửa, xóa, tìm kiếm, in thông tin của từng máy ảnh

**2.Xác định các lớp bài toán**



**- Class MaysAnh**

**+** Thuộc tính: String name, code, thuongHieu, ,color, material, namRaMat, int size, pin, price, memory

+ Phương thức :

* get,set cho các thuộc tính kiểu private
* toString:

**-Class Mirroless kế thừa từ class máy ảnh**

+Thuộc tính: boolean video, super( String name, code, thuonghieu, ,color, material, namRaMat, int size, pin, price, memory)

+Phương thức:

* get, set
* toString:

-**Class Compact kế thừa từ class máy ảnh**

+Thuộc tính: super(: String name, code, thuonghieu, ,color, material, namRaMat, int size, pin, price, memory)

+ Phương thức:

* get, set cho các thuộc tính kiểu private
* toString: phương thức để trả về thông tin đầy đủ của sản phẩm máy ảnh compact dưới dạng chuỗi

**-Class Bridgee kế thừa từ class máy ảnh**

* +Thuộc tính: boolean screen xác định xem sản phẩm có sử dụng màn hình không

Super ( String name, code, thuonghieu, ,color, material, namRaMat, int size, pin, price, memory)

* Thuộc tính: get, set
* toString: phương thức để trả về thông tin đầy đủ của sản phẩm máy ảnh bridgee dưới dạng chuỗi

**-Class DSRLL kế thừa Bridgee**

+Thuộc tính: super(double size, String pin, price, color, material, nam, thuonghieu, namRaMat, code, int memory, boolean screen)

+Phương thức :

* toString

**-Class QuanLyKho**

**+**Thuộc tính:Arraylist

+Phương thức:

QuanLyKho(Arraylist<MaysAnh>msa) : khởi tạo đối tượng QuanLyKho với một danh sách các đối tượng MaysAnh

QuanLyKho():khởi tạo đối tượng QuanLyKho với một danh sách trống

> them\_may\_anh(MaysAnh mt) thêm một đối tượng máy ảnh vào danh sách

>inThongTin() :in thông tin các đối tượng máy ảnh trong danh sách

>kiem\_tra\_so\_luong(): kiểm tra số lượng đối tượng trong danh sách

>tim\_may\_anh\_theo\_ten (name:String): tìm kiếm và in ra thông tin đối tượng máy ảnh theo tên

>tim\_may\_anh\_theo\_code (name:String): tìm kiếm và in ra thông tin đối tượng máy ảnh theo mã

>sap\_xep\_theo\_gia\_giam(): sắp xếp các đối tượng máy ảnh theo giá giầm dần

>sap\_xep\_theo\_gia\_tang(): sắp xếp các đối tượng máy ảnh theo giá tăng dần

>xoa\_ mot\_may\_anh (MaysAnh mt): xóa một đối tượng máy ảnh khỏi danh sách

>xoa\_tat\_ca(): xóa tât cả các đối tượng máy ảnh khỏi danh sách

Ảnh có chứa bàn

Mô tả được tạo tự động

**Ảnh có chứa văn bản

Mô tả được tạo tự độngẢnh có chứa văn bản

Mô tả được tạo tự động Ảnh có chứa văn bản

Mô tả được tạo tự động Ảnh có chứa bàn

Mô tả được tạo tự độngẢnh có chứa bàn

Mô tả được tạo tự độngẢnh có chứa văn bản

Mô tả được tạo tự động 3.Mô tả thuật toán**

1.Nhập thông tin máy ảnh :

-Dùng Arraylist thêm thông tin của máy ảnh vào trong kho

2.In thông tin máy ảnh:

-Dựa vào Arraylist đã nhập sẽ in thông tin tất cả máy ảnh đã nhập vào kho ra màn hình

3.Kiểm tra số lương:

-Dựa vào Arraylist đã nhập , phươg thức kiểm tra số lượng sẽ trả ra số nguyên cụ thể để ta biết chính xác có bao nhiêu máy trong kho

4.Tìm máy ảnh theo tên:

-Dựa vào Arraylsit đã nhập ta nhập tên của máy ảnh cần tìm

-Dùng câu lệnh if nếu máy ảnh đó có tồn tại trong kho sẽ in ra toàn bộ thông tin của máy ảnh đó

5.Tìm máy ảnh theo code:

-Dựa vào Arraylist đã nhập ta nhập mã của máy ảnh cần tìm

-Dùng câu lệnh if nếu máy ảnh có tồn tại trong kho sẽ in ra toàn bộ thông tin của máy ảnh đó

6. Sắp xếp máy ảnh theo giá giảm:

-Dùng câu lệnh if..else để ràng buộc điều kiện, so sánh giá từng máy ảnh và sắp xếp theo giá giàm dần

7.Sắp xếp máy ảnh theo giá tăng:

-Dùng câu lệnh if..else để ràng buộc điều kiện, so sánh giá từng máy ảnh và sắp xếp theo giá giàm dần

8.Xóa một máy ảnh:

-Dựa vào Arraylist đã nhập, tìm máy ảnh có trong kho và xóa máy ảnh đó khỏi kho

9.Xóa tất cả:

-Dùng lệnh removeAll() xóa toàn bộ thông tin máy ảnh đã nhập

**4.Cài đặt bài toán**

* **Class Máy ảnh**

package QuanLy;

import java.util.Objects;

public class MaysAnh implements Comparable<MaysAnh>{

private String noiSanXuat;

private String thuongHieu, color, material, code, name ;

private int namRaMat, size, pin, price, memory;

public MaysAnh() {

}

public MaysAnh(String code) {

this.code = code;

}

public MaysAnh(String name, String thuongHieu, String color, String material, String code, int namRaMat, int size, int pin, int price, int memory) {

this.name= name;

this.thuongHieu= thuongHieu;

this.color = color;

this.material = material;

this.code = code;

this.namRaMat = namRaMat;

this.size = size;

this.pin = pin;

this.price = price;

this.memory = memory;

}

public String getNoiSanXuat() {

return noiSanXuat;

}

public void setNoiSanXuat(String noiSanXuat) {

this.noiSanXuat = noiSanXuat;

}

public String getThuongHieu() {

return thuongHieu;

}

public void setThuongHieu(String thuongHieu) {

this.thuongHieu = thuongHieu;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

public String getMaterial() {

return material;

}

public void setMaterial(String material) {

this.material = material;

}

public String getCode() {

return code;

}

public void setCode(String code) {

this.code = code;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getNamRaMat() {

return namRaMat;

}

public void setNamRaMat(int namRaMat) {

this.namRaMat = namRaMat;

}

public int getSize() {

return size;

}

public void setSize(int size) {

this.size = size;

}

public int getPin() {

return pin;

}

public void setPin(int pin) {

this.pin = pin;

}

public int getPrice() {

return price;

}

public void setPrice(int price) {

this.price = price;

}

public int getMemory() {

return memory;

}

public void setMemory(int memory) {

this.memory = memory;

}

@Override

public String toString() {

return "MaysAnh{" + "Noi San Xuat= " + noiSanXuat + ", Thuong Hieu= " + thuongHieu + ", Mau Sac= " + color + ", Chat Lieu=" + material + ", Ma San Pham= " + code + ", Ten San Pham= " + name + ", Nam Ra Mat=" + namRaMat + ", Kich Thuoc May Anh= " + size + ", Dung Luong Pin= " + pin + " mAh"+", Gia Tien= " + price + "VNĐ" + ", memory=" + memory +" GB"+ '}';

}

@Override

public int compareTo(MaysAnh o) {

return this.code.compareTo(o.code);

}

@Override

public boolean equals(Object obj) {

if(obj instanceof MaysAnh){

MaysAnh another = (MaysAnh) obj;

if(this.code.equals(another.code)){

return true;

}

}

return false;

}

@Override

public int hashCode() {

return code.compareTo(code);

}

}

* **Class Compact**

package QuanLy;

import java.util.Objects;

public class Compactt extends MaysAnh {

private boolean screen;

public Compactt() {

}

public Compactt(String code) {

super(code);

}

public Compactt(boolean screen, String name, String thuongHieu, String color, String material, String code, int namRaMat, int size, int pin, int price, int memory) {

super(name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

this.screen = screen;

}

public boolean isScreen() {

return screen;

}

public void setScreen(boolean screen) {

this.screen = screen;

}

@Override

public String toString() {

return "Compactt{" + "Ten San Pham=" + getName() + " , Ma San Pham= " +getCode() + ", Thuong Hieu= " +getThuongHieu() + ", Sac Mau= " + getColor() + ", Chat Lieu=" + getMaterial() + ", Nam Ra Mat=" + getNamRaMat() + ", Kich Thuoc May Anh=" + getSize()+"(inch)" + ", Dung Luong Pin=" + getPin() +"(mAh)"+ ", Gia Tien=" + getPrice() +" (VND)" + ", Dung Luong Bo Nho=" + getMemory()+"(Gb)"+", Su Dung Man Hinh=" + screen + '}';

}

}

* **Class Bridgee**

package QuanLy;

import java.util.Objects;

public class Bridgee extends MaysAnh {

private boolean screen;

public Bridgee(String code){

super(code);

}

public Bridgee() {

}

public Bridgee(boolean screen, String name, String thuongHieu, String color, String material, String code, int namRaMat, int size, int pin, int price, int memory) {

super(name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

this.screen = screen;

}

public boolean isScreen() {

return screen;

}

public void setScreen(boolean screen) {

this.screen = screen;

}

@Override

public String toString() {

return "Bridgee{" + "Ten San Pham=" + getName() + " , Ma San Pham= " +getCode() + ", Thuong Hieu= " +getThuongHieu() + ", Sac Mau= " + getColor() + ", Chat Lieu=" + getMaterial() + ", Nam Ra Mat=" + getNamRaMat() + ", Kich Thuoc May Anh=" + getSize()+"(inch)" + ", Dung Luong Pin=" + getPin() +"(mAh)"+ ", Gia Tien=" + getPrice() +"(VND)" + ", Dung Luong Bo Nho=" + getMemory()+"(Gb)" + ", Su Dung Man Hinh=" + screen +'}';

}

}

* **Class DSLRR**

package QuanLy;

import java.util.Objects;

public class DSLRR extends Bridgee {

public DSLRR() {

}

public DSLRR(String code) {

super(code);

}

public DSLRR(boolean screen, String name, String thuongHieu, String color, String material, String code, int namRaMat, int size, int pin, int price, int memory) {

super(screen, name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

}

@Override

public String toString() {

return "DSLRR{" + "Ten San Pham=" + getName() + " , Ma San Pham= " +getCode() + ", Thuong Hieu= " +getThuongHieu() + ", Sac Mau= " + getColor() + ", Chat Lieu=" + getMaterial() + ", Nam Ra Mat=" + getNamRaMat() + ", Kich Thuoc May Anh=" + getSize()+"(inch)" + ", Dung Luong Pin=" + getPin() +"(mAh)"+ ", Gia Tien=" + getPrice() +"(VND)" + ", Dung Luong Bo Nho=" + getMemory()+ "(Gb)" + ", Su Dung Man Hinh=" + isScreen()+ '}';

}

}

* **Class Mirroless**

package QuanLy;

import java.util.Objects;

public class Mirroless extends MaysAnh implements Comparable< Mirroless>{

public Mirroless() {

}

public Mirroless(String code) {

super(code);

}

public Mirroless(String name, String thuongHieu, String color, String material, String code, int namRaMat, int size, int pin, int price, int memory) {

super(name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

}

@Override

public String toString() {

return "Mirroless{" + "Ten San Pham=" + getName() + " , Ma San Pham= " +getCode() + ", Thuong Hieu= " +getThuongHieu() + ", Sac Mau= " + getColor() + ", Chat Lieu=" + getMaterial() + ", Nam Ra Mat=" + getNamRaMat() + ", Kich Thuoc May Anh=" + getSize()+"(inch) " + ", Dung Luong Pin=" + getPin() +"(mAh)"+ ", Gia Tien=" + getPrice() +" (VNĐ)" + ", Dung Luong Bo Nho=" + getMemory()+ "(Gb)" + '}';

}

public int compareTo(Mirroless o) {

return this.getCode().compareTo(o.getCode());

}

@Override

public int hashCode() {

return Objects.hash(getCode());

}

@Override

public boolean equals (Object obj) {

if (this == obj) {

return true;

}

if (obj == null) {

return false;

}

if (getClass() != obj.getClass()) {

return false;

}

Mirroless other = (Mirroless) obj;

return Objects.equals(getCode(), other.getCode());

}

}

* **CLASS Quản lý kho**

package QuanLy;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Comparator;

public class QuanLyKho {

private ArrayList<MaysAnh> msa;

public QuanLyKho(ArrayList<MaysAnh> msa) {

this.msa = msa;

}

public QuanLyKho() {

msa = new ArrayList<>();

}

public void them\_may\_anh(MaysAnh mt) {

this.msa.add(mt);

System.out.println("Da them thanh cong");

}

public void inThongTin() {

for (MaysAnh mayanh : msa) {

System.out.println(mayanh.toString());

}

}

public int kiem\_tra\_so\_luong() {

return this.msa.size();

}

public void tim\_may\_anh\_theo\_ten(String name) {

for (MaysAnh mayAnh : msa) {

if (mayAnh.getName().indexOf(name) >= 0) {

System.out.println(mayAnh);

}

}

}

public void tim\_may\_anh\_theo\_code(String code) {

for (MaysAnh mayAnh : msa) {

if (mayAnh.getCode().indexOf(code) >= 0) {

System.out.println(mayAnh);

}

}

}

public void xap\_xep\_theo\_gia\_giam() {

Collections.sort(msa, new Comparator<MaysAnh>() {

@Override

public int compare(MaysAnh mt1, MaysAnh mt2) {

if (mt1.getPrice() > mt2.getPrice()) {

return -1;

} else if (mt1.getPrice() < mt2.getPrice()) {

return 1;

} else {

return 0;

}

}

});

}

public void xap\_xep\_theo\_gia\_tang() {

Collections.sort(msa, new Comparator<MaysAnh>() {

@Override

public int compare(MaysAnh mt1, MaysAnh mt2) {

if (mt1.getPrice() < mt2.getPrice()) {

return -1;

} else if (mt1.getPrice() > mt2.getPrice()) {

return 1;

} else {

return 0;

}

}

});

}

public boolean xoa\_mot\_may\_anh(MaysAnh mt) {

return this.msa.remove(mt);

}

public void xoa\_tat\_ca() {

this.msa.removeAll(msa);

}

}

* **Class main**

package QuanLy.MM;

import java.util.Objects;

import java.util.Scanner;

import QuanLy.Compactt;

import QuanLy.Mirroless;

import QuanLy.Bridgee;

import QuanLy.DSLRR;

import QuanLy.QuanLyKho;

import QuanLy.MaysAnh;

public class NewMain {

public static void main(String[] args) {

QuanLyKho qlk = new QuanLyKho();

Scanner sc = new Scanner(System.in);

int menu;

do {

System.out.println("--------------------------------------");

System.out.println("Menu lua chon --------");

System.out.println("1. Them thong tin may anh");

System.out.println("2. In thong tin may anh");

System.out.println("3. Kiem tra so luong may anh");

System.out.println("4. Tim may anh theo ten");

System.out.println("5. Tim may anh theo code");

System.out.println("6. Xap xep may anh theo gia giam");

System.out.println("7. Xap xep may anh theo gia tang");

System.out.println("8. xoa mot may anh theo ma code");

System.out.println("9. Xoa toan bo may anh trong kho");

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

System.out.print("Nhap vao lua chon cua ban: ");

menu = sc.nextInt();

switch (menu) {

case 1:

int menu\_cp;

do {

System.out.println("Chao ban! CAm on ban da su dung chuong trinh");

System.out.println("--------------------------------------------");

System.out.println("1. Compact (May Anh Point & Shoot)");

System.out.println("2. DSLR (May Anh Guong Lat) ");

System.out.println("3. Mirroless( May Anh Khong Guong Lat) ");

System.out.println("4. Bridge (May Anh Zoom)");

System.out.println("0. Quay lai chuong trinh");

System.out.print("Nhap vao lua chon cua ban: ");

menu\_cp = sc.nextInt();

switch (menu\_cp) {

case 1:{

sc.nextLine();

System.out.print("Nhap Ten San Pham: ");

String name = sc.nextLine();

System.out.print("Nhap Ma San Pham: ");

String code = sc.nextLine();

System.out.print("Nhap Thuong Hieu: ");

String thuongHieu = sc.nextLine();

System.out.print("Nhap vao Mau Sac: ");

String color = sc.nextLine();

System.out.print("Nhap vao Chat Lieu: ");

String material = sc.nextLine();

System.out.print("Nhap vao Nam Ra Mat: ");

int namRaMat = sc.nextInt();

System.out.print("Nhap vao Kich Thuoc: ");

int size = sc.nextInt();

System.out.print("Nhap vao dung luong pin: ");

int pin = sc.nextInt();

System.out.print("Nhap vao dung luong bo nho: ");

int memory = sc.nextInt();

System.out.print("Nhap gia san pham: ");

int price = sc.nextInt();

System.out.print("Nhap True hoac False cho man hinh: ");

boolean screen = sc.nextBoolean();

MaysAnh cpn = new Compactt (screen, name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

qlk.them\_may\_anh(cpn);

}

break;

case 2: {

sc.nextLine();

System.out.print("Nhap Ten San Pham: ");

String name = sc.nextLine();

System.out.print("Nhap Ma San Pham: ");

String code = sc.nextLine();

System.out.print("Nhap Thuong Hieu: ");

String thuongHieu = sc.nextLine();

System.out.print("Nhap vao Mau Sac: ");

String color = sc.nextLine();

System.out.print("Nhap vao Chat Lieu: ");

String material = sc.nextLine();

System.out.print("Nhap vao Nam Ra Mat: ");

int namRaMat = sc.nextInt();

System.out.print("Nhap vao Kich Thuoc: ");

int size = sc.nextInt();

System.out.print("Nhap vao dung luong pin: ");

int pin = sc.nextInt();

System.out.print("Nhap vao dung luong bo nho: ");

int memory = sc.nextInt();

System.out.print("Nhap gia san pham: ");

int price = sc.nextInt();

System.out.print("Nhap True hoac False cho man hinh: ");

boolean screen = sc.nextBoolean();

MaysAnh cpn = new DSLRR(screen, name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

qlk.them\_may\_anh(cpn);

}

break;

case 3: {

sc.nextLine();

System.out.print("Nhap Ten San Pham: ");

String name = sc.nextLine();

System.out.print("Nhap Ma San Pham: ");

String code = sc.nextLine();

System.out.print("Nhap Thuong Hieu: ");

String thuongHieu = sc.nextLine();

System.out.print("Nhap vao Mau Sac: ");

String color = sc.nextLine();

System.out.print("Nhap vao Chat Lieu: ");

String material = sc.nextLine();

System.out.print("Nhap vao Nam Ra Mat: ");

int namRaMat = sc.nextInt();

System.out.print("Nhap vao Kich Thuoc: ");

int size = sc.nextInt();

System.out.print("Nhap vao dung luong pin: ");

int pin = sc.nextInt();

System.out.print("Nhap vao dung luong bo nho: ");

int memory = sc.nextInt();

System.out.print("Nhap gia san pham: ");

int price = sc.nextInt();

System.out.print("Nhap vao True hoac Fasle cho che do quay phim: ");

boolean video = sc.nextBoolean();

System.out.print("Nhap True hoac False cho man hinh: ");

boolean screen = sc.nextBoolean();

MaysAnh cpn = new Mirroless(screen, video, name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

qlk.them\_may\_anh(cpn);

}

break;

case 4: {

sc.nextLine();

System.out.print("Nhap Ten San Pham: ");

String name = sc.nextLine();

System.out.print("Nhap Ma San Pham: ");

String code = sc.nextLine();

System.out.print("Nhap Thuong Hieu: ");

String thuongHieu = sc.nextLine();

System.out.print("Nhap vao Mau Sac: ");

String color = sc.nextLine();

System.out.print("Nhap vao Chat Lieu: ");

String material = sc.nextLine();

System.out.print("Nhap vao Nam Ra Mat: ");

int namRaMat = sc.nextInt();

System.out.print("Nhap vao Kich Thuoc: ");

int size = sc.nextInt();

System.out.print("Nhap vao dung luong pin: ");

int pin = sc.nextInt();

System.out.print("Nhap vao dung luong bo nho: ");

int memory = sc.nextInt();

System.out.print("Nhap gia san pham: ");

int price = sc.nextInt();

System.out.print("Nhap True hoac False cho man hinh: ");

boolean screen = sc.nextBoolean();

MaysAnh cpn = new Bridgee(screen, name, thuongHieu, color, material, code, namRaMat, size, pin, price, memory);

qlk.them\_may\_anh(cpn);

}

break;

default:

if (menu\_cp == 0) {

System.out.println("Quay lai chuong trinh thanh cong");

} else {

System.out.println("Vui long nhap lai lua chon");

}

}

} while (menu\_cp != 0);

break;

case 2:

if (qlk.kiem\_tra\_so\_luong() == 0) {

System.out.println("Hien khong co may trong kho");

} else {

System.out.println("So may hien tai la: " + qlk.kiem\_tra\_so\_luong());

}

qlk.inThongTin();

break;

case 3:

System.out.println("So may hien tai la: " + qlk.kiem\_tra\_so\_luong());

break;

case 4:

sc.nextLine();

System.out.print("Nhap ten can tim: ");

String name = sc.nextLine();

qlk.tim\_may\_anh\_theo\_ten(name);

System.out.println("\n");

break;

case 5:

sc.nextLine();

System.out.print("Nhap code can tim: ");

String code = sc.nextLine();

qlk.tim\_may\_anh\_theo\_code(code);

System.out.println("\n");

break;

case 6:

qlk.xap\_xep\_theo\_gia\_giam();

System.out.println("Thong tin sau khi giam la: ");

qlk.inThongTin();

break;

case 7:

qlk.xap\_xep\_theo\_gia\_tang();

System.out.println("Thong tin sau khi tang la: ");

qlk.inThongTin();

break;

case 8:

sc.nextLine();

System.out.print("Nhap vao ma code: ");

code = sc.nextLine();

MaysAnh mtm = new MaysAnh(code);

System.out.println("Da xoa may " + qlk.xoa\_mot\_may\_anh(mtm));

break;

case 9:

qlk.xoa\_tat\_ca();

System.out.println("Da xoa tat ca may trong kho");

break;

default:

if (menu == 0) {

System.out.println("Thoat chuong trinh");

} else {

System.out.println("Vui long nhap lai lua chon");

}

}

} while (menu != 0);

}

}

**5.Kiểm tra hộp thử**

1. Thêm thông tin máy ảnh

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

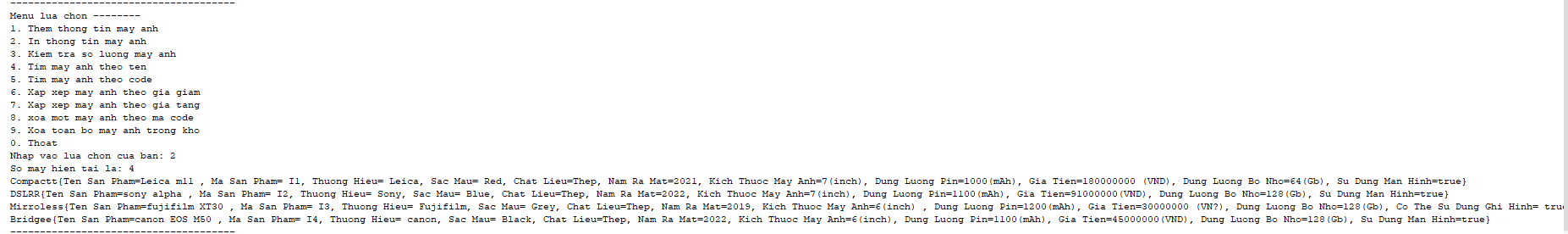
Mô tả được tạo tự độngẢnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

2.In thông tin máy ảnh

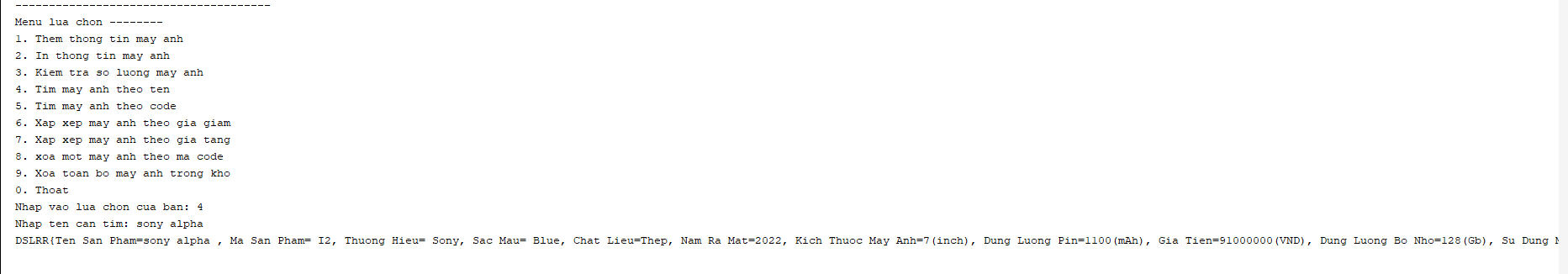


3.Kiểm tra số lượng máy

Ảnh có chứa văn bản

Mô tả được tạo tự động

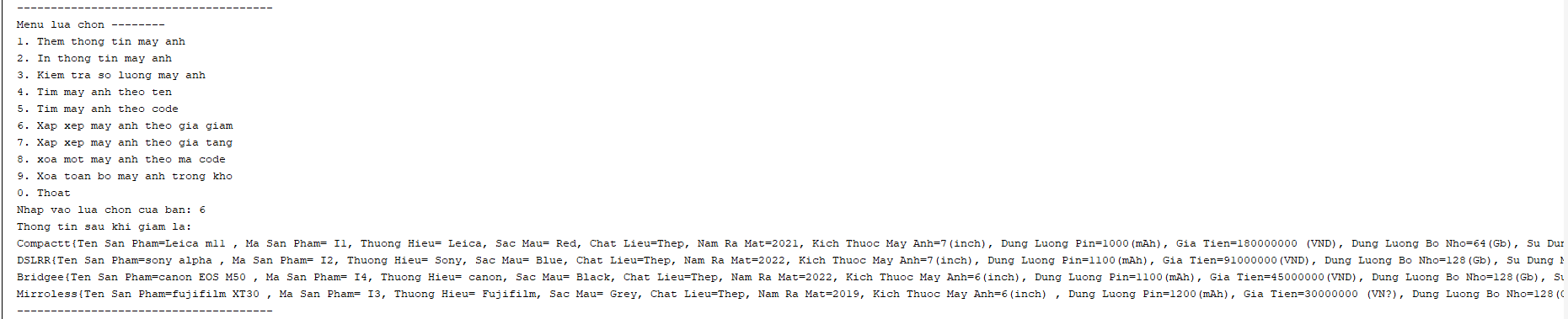
4.Tìm máy ảnh theo tên



Ảnh có chứa văn bản

Mô tả được tạo tự động5.Tìm máy ảnh theo code

6.Sắp xếp máy ảnh giảm theo giá



Ảnh có chứa văn bản

Mô tả được tạo tự động7.Sắp xếp máy ảnh theo giá tăng

8.Xóa máy ảnh theo code

9.Xóa toàn bộ máy ảnh

Ảnh có chứa bàn

Mô tả được tạo tự động

**6.Kết luận**

Ưu điểm

Dễ hiểu: Chương trình có cấu trúc rõ ràng, dễ hiểu, giúp cho người dùng dễ dàng sử dụng và tùy chỉnh theo ý muốn.

Tính linh hoạt: Chương trình có thể xử lý được nhiều loại tập tin âm thanh khác nhau, từ đó đáp ứng được nhiều nhu cầu của người dùng.

Độ chính xác cao: Chương trình sử dụng thuật toán đơn giản nhưng hiệu quả, đem lại kết quả chính xác và tốt.

Nhược điểm

Tốc độ xử lý chậm: Thuật toán của chương trình đòi hỏi một số tính toán phức tạp, điều này có thể làm tốc độ xử lý của chương trình bị giảm xuống.

Giới hạn về độ chính xác: Chương trình chỉ sử dụng một phương pháp nhận dạng giọng nói, do đó, độ chính xác của chương trình có thể bị giới hạn và không đủ cao để đáp ứng các nhu cầu đặc biệt của người dùng.