**BỘ TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**TRƯỜNG ĐH TÀI NGUYÊN VÀ MÔI TRƯỜNG TP. HCM**

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



**BÁO CÁO ĐỒ ÁN MÔN LẬP TRÌNH HĐT**

**ĐỒ ÁN: QUẢN LÝ KHO QUẦN ÁO**

Thành viên của nhóm:

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | Hoàng Thị Minh Trang | (10500080125) | |
| 2. | Võ Minh Hân | | (1050080096) |
| 3. | Huỳnh Thị Thu Thảo | | (1050080117) |

Lớp: 10\_ĐH\_CNTT3

***TP. Hồ Chí Minh, tháng 4 năm 2023***

**Mục Lục**

[**I.** **Phát biểu bài toán** 3](#_Toc133270095)

[**II.** **Xác định các lớp của bài toán** 3](#_Toc133270096)

[**1.** **Class Quần Áo** 3](#_Toc133270097)

[**2.** **Class Quần Áo Nữ** 4](#_Toc133270098)

[**3.** **Class Quần Áo Nam** 4](#_Toc133270099)

[**4.** **Class Kho** 4](#_Toc133270100)

[**5.** **Main** 4](#_Toc133270101)

[**III.** **Mô tả thuật toán thao tác** 5](#_Toc133270102)

[**1.** **Thêm thông tin quần hoặc áo** 5](#_Toc133270103)

[**2.** **Xuất thông tin** 5](#_Toc133270104)

[**3.** **Xóa sản phẩm** 6](#_Toc133270105)

[**4.** **Sửa một thông tin** 6](#_Toc133270106)

[**5.** **Kiểm tra số lượng** 6](#_Toc133270107)

[**6.** **Tìm kiếm sản phẩm** 7](#_Toc133270108)

[**7.** **Sao lưu thông tin** 7](#_Toc133270109)

[**8.** **Khôi phục thông tin** 7](#_Toc133270110)

[**IV.** **Cài đặt bài toán** 8](#_Toc133270111)

[**1.** **Class Quần Áo** 8](#_Toc133270112)

[**2.** **Class Quần Áo Nữ** 13](#_Toc133270113)

[**3.** **Class Quần Áo Nam** 13](#_Toc133270114)

[**4.** **Class Kho** 14](#_Toc133270115)

[**5.** **Main** 21](#_Toc133270116)

[**V.** **Kiểm thử** 24](#_Toc133270117)

[**1.** **Nhập thông tin Quần Áo** 25](#_Toc133270118)

[**2.** **Xuất thông tin Quần Áo** 26](#_Toc133270119)

[**3.** **Số lượng sản phẩm Quần Áo** 27](#_Toc133270120)

[**4.** **Tìm kiếm sản phẩm theo mã** 27](#_Toc133270121)

[**5.** **Chỉnh sửa thông tin** 28](#_Toc133270122)

[**6.** **Xóa một thông tin** 29](#_Toc133270123)

[**7.** **Sao lưu thông tin** 30](#_Toc133270124)

[**8.** **Khôi phục thông tin** 30](#_Toc133270125)

[**VI.** **Tài liệu tham khảo** 31](#_Toc133270126)

# **Phát biểu bài toán**

* Quản lý kho quần áo là quá trình quản lý các hoạt động liên quan đến nhập, xuất và lưu trữ các sản phẩm quần áo trong kho hàng.
* Quản lý kho quần áo là một phần quan trọng của hoạt động bán lẻ, đặc biệt là trong ngành thời trang. Các hoạt động quản lý kho quần áo bao gồm việc nhập hàng từ các nhà cung cấp, lưu trữ sản phẩm trong kho, đóng gói sản phẩm để vận chuyển đến khách hàng và theo dõi số lượng sản phẩm còn lại trong kho.
* Mục đích của việc quản lý kho quần áo là đảm bảo rằng kho luôn được cung cấp đủ sản phẩm để đáp ứng nhu cầu của khách hàng và tránh tình trạng thiếu hụt hoặc lãng phí hàng hóa.
* Để quản lý kho quần áo hiệu quả, các doanh nghiệp cần sử dụng các phần mềm quản lý kho để giám sát số lượng sản phẩm trong kho và quản lý các hoạt động nhập, xuất hàng.
* Họ cũng cần sử dụng các chiến lược quản lý kho hiệu quả như tối ưu hóa quy trình lưu trữ sản phẩm, tối ưu hóa khoảng cách giữa các sản phẩm và định kỳ kiểm tra lượng hàng tồn kho để đảm bảo rằng kho luôn được quản lý một cách tốt nhất
* Hoạt động các đối tượng cần quản lý kho quần áo như sau:

**+ Quần, áo:** Style, Size, Màu sắc, Chất liệu, Mã, Ngày nhập.

**+ Xuất – nhập kho:** Tiếp nhận, kiểm tra, giấy tờ yêu cầu nhập, xuất , ghi nhận và lưu trữ

**+ Quần ,áo :** cùng có các thuộc tính như Style, Size, Màu sắc, Chất liệu, Mã, Ngày nhập nên được kế thừa từ lớp Quần Áo

* Tính thừa kế

Arraylist<QuanAo> quản lý cả Quần và áo , khi nhập ta có thể nhập tùy  theo Đối tượng Quần và áo vào ArrayList này

* Tính đa hình

# **Xác định các lớp của bài toán**

## **Class Quần Áo**

* Thuộc tính: String Style, Size, Màu sắc, Chất liệu, Mã, Date Ngaynhap.
* Phương thức :

+ Contructor: Hàm tạo có đối số và hàm tạo không có đối số

+ Getter, Setter: Phương thức get, set cho các thuộc tính kiểu public

## **Class Quần Áo Nữ**

* Thuộc tính : kế thừa toàn bộ ở lớp Quần Áo
* Phương thức :

+ Nhập: Nhập thông tin các thuộc tính

## **Class Quần Áo Nam**

* Thuộc tính : kế thừa toàn bộ lớp Quần Áo
* Phương thức :

+ Nhập thông tin các thuộc tính

## **Class Kho**

* Thuộc tính :

+ Arraylist<Quanao> kho;

* Phương thức :

+themQuan(): Thêm quần vào trong kho

+themAo(): Thêm áo vào trong kho

+xuatSanPham(): Xuất thông tin Áo nam, Áo nữ, Quần nữ, Quần nam

+xoaSanPham(): Xóa 1 sản phẩm dựa trên mã

+suaSanPham(): Chỉnh sửa lại thông của sản phẩm

+kiemtrasoluong(): Xem số lượng sản phẩm ở trong kho

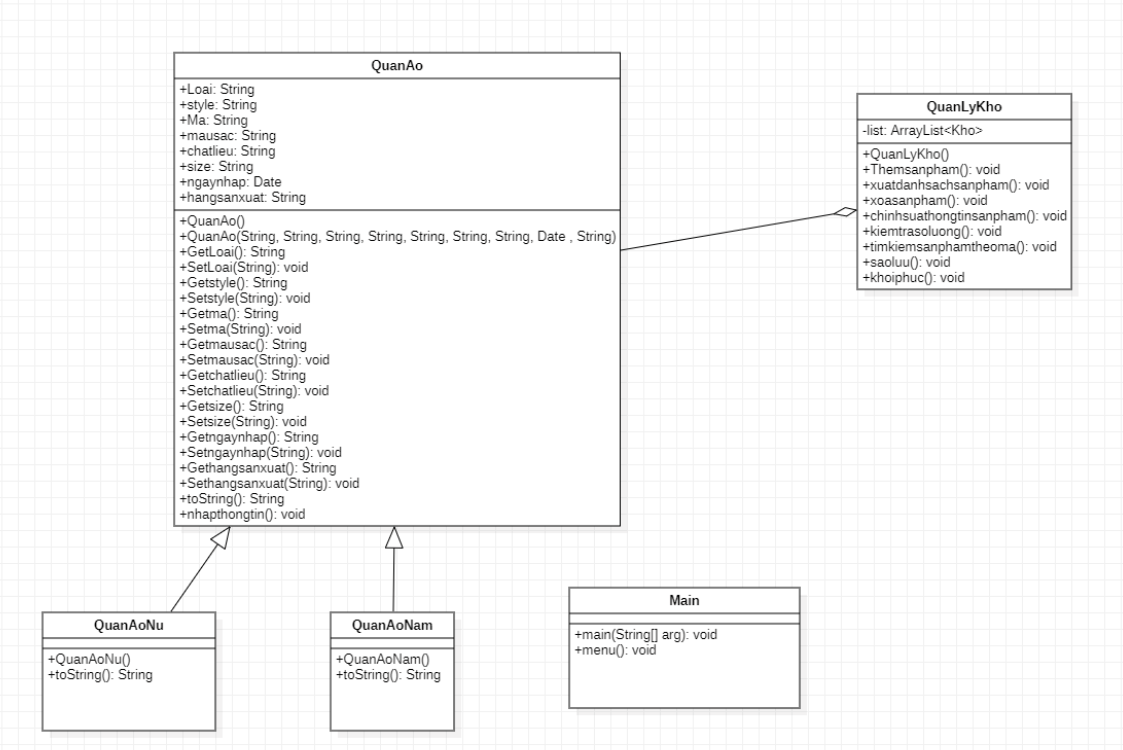
+timkiemSanPham(): Tìm kiếm sản phẩm dựa trên mã

+luuFile(): sao lưu sản phẩm

+docFile(): khôi phục lại sản phẩm

## **Main**

* Menu: Menu hiện thị các chức năng thêm thông tin, xuất thông tin, xem số lượng, xóa một thông tin, tìm kiếm thông tin, chỉnh sửa thông tin , lưu trữ thông tin, khôi phục thông tin



# **Mô tả thuật toán thao tác**

## **Thêm thông tin quần hoặc áo**

* Thuật toán :

+ Nhập thêm thông tin quần áo vào trong kho: bao gồm các thông tin như Style, size, chất liệu, màu xắc , mã

+ Dùng câu lệnh so sánh và vòng lặp sao cho không nhập trùng một mã

+ Phương thức này được sử dụng để thêm một đối tượng mới vào kho. Trước khi thêm, phương thức sẽ kiểm tra xem đối tượng có trùng mã với bất kỳ đối tượng nào trong kho hay không. Nếu có, phương thức sẽ in ra thông báo và trả về false, nếu không, phương thức sẽ thêm đối tượng vào kho và trả về true

## **Xuất thông tin**

* Thuật toán:

+ Phương thức sử dụng một vòng lặp for để duyệt qua toàn bộ danh sách sản phẩm được lưu trữ trong một đối tượng "kho" (presumably là một ArrayList hoặc một Collection khác), và in ra từng phần tử của danh sách bằng cách sử dụng phương thức get(i) của đối tượng "kho".

+ Ví dụ Quần là biến duyệt vòng lặp, mỗi lần duyệt qua một sản phẩm thì biến này sẽ tham chiếu đến sản phẩm đó để in ra thông tin.

## **Xóa sản phẩm**

* Thuật toán :

Sử dụng một vòng lặp for để duyệt qua danh sách sản phẩm được lưu trữ trong đối tượng "kho" (presumably là một ArrayList hoặc một Collection khác).

Trong vòng lặp, phương thức kiểm tra xem sản phẩm có mã bằng với mã được nhập từ bàn phím hay không bằng cách sử dụng phương thức getMa() của đối tượng sản phẩm tại vị trí i. Nếu sản phẩm được tìm thấy, phương thức sẽ gọi phương thức remove(i) của đối tượng "kho" để xóa sản phẩm tại vị trí i và in ra thông báo "San pham co ma [Ma] da duoc xoa". Sau đó, phương thức kết thúc bằng cách sử dụng từ khóa return để kết thúc phương thức.

Nếu không có sản phẩm nào có mã bằng với mã được nhập từ bàn phím, phương thức sẽ kết thúc vòng lặp và in ra thông báo "Khong tim thay san pham co ma [Ma]".

## **Sửa một thông tin**

* Thuật toán :

Nó sử dụng một vòng lặp for để duyệt qua danh sách sản phẩm được lưu trữ trong đối tượng "kho" (presumably là một ArrayList hoặc một Collection khác). Trong vòng lặp, phương thức kiểm tra xem sản phẩm có mã bằng với mã được nhập từ bàn phím hay không bằng cách sử dụng phương thức getMa() của đối tượng sản phẩm tại vị trí i.

Nếu sản phẩm được tìm thấy, phương thức sẽ yêu cầu người dùng nhập thông tin mới cho sản phẩm bằng cách gọi phương thức nhapthongtin() của đối tượng sản phẩm tại vị trí i.

Sau đó, phương thức in ra thông báo "San pham co ma [ma] da duoc sua" và kết thúc phương thức bằng cách sử dụng từ khóa return.

Nếu không có sản phẩm nào có mã bằng với mã được nhập từ bàn phím, phương thức sẽ kết thúc vòng lặp và in ra thông báo "Khong tim thay san pham co ma [ma]".

## **Kiểm tra số lượng**

* Thuật toán :

Phương thức này bắt đầu bằng cách kiểm tra xem danh sách sản phẩm trong đối tượng "kho" có rỗng hay không. Nếu danh sách rỗng, phương thức sẽ in ra thông báo "Kho khong co san pham" và kết thúc phương thức bằng cách sử dụng từ khóa return.

Nếu danh sách không rỗng, phương thức sẽ in ra số lượng sản phẩm trong kho bằng cách sử dụng phương thức size() của đối tượng "kho". Cuối cùng, phương thức sẽ kết thúc bằng cách in ra thông báo "Tong so san pham trong kho la: [so\_luong\_san\_pham]".

## **Tìm kiếm sản phẩm**

* Thuật toán:

Phương thức sử dụng một vòng lặp for để duyệt qua danh sách sản phẩm được lưu trữ trong đối tượng "kho" (presumably là một ArrayList hoặc một Collection khác). Trong vòng lặp, phương thức kiểm tra xem sản phẩm có mã bằng với mã được nhập từ bàn phím hay không bằng cách sử dụng phương thức getMa() của đối tượng sản phẩm tại vị trí i.

Nếu sản phẩm được tìm thấy, phương thức sẽ in ra thông tin của sản phẩm đó bằng cách sử dụng phương thức toString() của đối tượng sản phẩm tại vị trí i và kết thúc phương thức bằng cách sử dụng từ khóa return.

Nếu không có sản phẩm nào có mã bằng với mã được nhập từ bàn phím, phương thức sẽ kết thúc vòng lặp và in ra thông báo "Khong tim thay san pham co ma [Ma]".

## **Sao lưu thông tin**

* Thuật toán :

Phương thức này sử dụng đối tượng FileWriter để ghi các thông tin của sản phẩm vào tệp.

Phương thức bắt đầu bằng cách nhận một đối tượng File làm tham số đầu vào, đại diện cho tệp mà phương thức sẽ ghi thông tin sản phẩm vào.

Trong phương thức, một vòng lặp for được sử dụng để duyệt qua danh sách các sản phẩm trong kho. Với mỗi sản phẩm, phương thức kiểm tra xem sản phẩm đó có phải là loại "Quần" hay "Áo" bằng cách sử dụng toán tử instanceof.

Nếu sản phẩm đang được xét là một "Quần", phương thức sẽ chuyển đổi đối tượng "Quần" sang đối tượng "String" để có thể ghi thông tin của sản phẩm đó vào tệp.

Sau đó, phương thức sử dụng phương thức write() của đối tượng FileWriter để ghi thông tin của sản phẩm "Quần" vào tệp. Tương tự, nếu sản phẩm đang được xét là một "Áo", phương thức sẽ chuyển đổi đối tượng "Áo" sang đối tượng "String" và ghi thông tin của sản phẩm "Áo" vào tệp.

Cuối cùng, phương thức đóng đối tượng FileWriter và in ra thông báo "da ghi file vao [duong\_dan\_tep]" để cho biết rằng quá trình ghi tệp đã hoàn tất. Nếu có lỗi xảy ra trong quá trình ghi tệp, phương thức sẽ in ra stack trace của ngoại lệ.

## **Khôi phục thông tin**

* Thuật toán :

Nó sử dụng một đối tượng Scanner để đọc file, lặp qua từng dòng và chia dòng đó thành các trường bằng cách sử dụng phương thức split(). Nếu số lượng trường bằng 5 và trường đầu tiên là "quan" hoặc "ao", phương thức tạo và thêm đối tượng Quan hoặc Áo vào danh sách. Nếu xảy ra lỗi khi đọc file, một thông báo lỗi cụ thể sẽ được in ra trên màn hình. Cuối cùng, phương thức in ra một thông báo để cho biết file đã được đọc xong và đối tượng Scanner được đóng để giải phóng tài nguyên.

# **Cài đặt bài toán**

## **Class Quần Áo**

**package** qlykhoquanao;

**import** java.sql.Date;

**import** java.time.LocalDate;

**import** java.time.format.DateTimeFormatter;

**import** java.util.Objects;

**import** java.util.Scanner;

**public** **class** QuanAo{

**public** String Loai,Style,Size, Chatlieu, Ma, Mausac, Hang;

**public** Date Ngaynhap;

**public** QuanAo(String Loai,String Hang, String Style,String size, String chatlieu, String mausac, String ma, Date ngaynhap) {

Loai = Loai;

Hang = Hang;

Style = Style;

Size = size;

Chatlieu = chatlieu;

Mausac = mausac;

Ma = ma;

Ngaynhap = ngaynhap;

}

**public** QuanAo() {

}

**public** String getLoai() {

**return** Loai;

}

**public** **void** setLoai(String Loai) {

**this**.Loai=Loai;

}

**public** String getHang() {

**return** Hang;

}

**public** **void** setHang(String Hang) {

**this**.Hang=Hang;

}

**public** String getStyle() {

**return** Style;

}

**public** **void** setStyle(String Style) {

**this**.Style = Style;

}

**public** String getMa() {

**return** Ma;

}

**public** **void** setMa(String ma) {

Ma = ma;

}

**public** String getSize() {

**return** Size;

}

**public** **void** setSize(String size) {

Size = size;

}

**public** String getChatlieu() {

**return** Chatlieu;

}

**public** **void** setChatlieu(String chatlieu) {

Chatlieu = chatlieu;

}

**public** String getMausac() {

**return** Mausac;

}

**public** **void** setMausac(String mausac) {

Mausac = mausac;

}

**public** Date getNgaynhap() {

**return** Ngaynhap;

}

**public** **void** setNgaynhap(Date ngaynhap) {

Ngaynhap = ngaynhap;

}

**public** String getSoluong() {

**return** getSoluong();

}

@Override

**public** **boolean** equals(Object obj) {

**if** (**this** == obj)

**return** **true**;

**if** (obj == **null**)

**return** **false**;

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

**return** **false**;

QuanAo other = (QuanAo) obj;

**return** Objects.*equals*(Ma, other.Ma);

}

**public** **void** nhapthongtin() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Nhap LoaiSP");

**this**.setLoai(sc.nextLine());

System.***out***.println(" Nhap Hang sx");

**this**.setHang(sc.nextLine());

System.***out***.println("Nhap Style");

**this**.setStyle(sc.nextLine());

System.***out***.print("Nhap size: ");

**this**.setSize(sc.nextLine());

System.***out***.print("Nhap chat lieu: ");

**this**.setChatlieu(sc.nextLine());

System.***out***.print("Nhap mau sac: ");

**this**.setMausac(sc.nextLine());

System.***out***.println("Nhap ma: ");

**this**.setMa(sc.nextLine());

System.***out***.println("Ngay hang duoc nhap (theo dinh dang dd/MM/yyyy): ");

String input = sc.nextLine();

LocalDate date;

**try** {

DateTimeFormatter formatter = DateTimeFormatter.*ofPattern*("dd/MM/yyyy");

date = LocalDate.*parse*(input, formatter);

**this**.setNgaynhap(Date.*valueOf*(date));

} **catch** (Exception e) {

System.***out***.println("Khong hop le, thu lai");

}

}

@Override

**public** String toString() {

**return** "QuanAo [Loai="+Loai+",Hang="+Hang+",Style="+Style+",Size=" + Size + ", Chatlieu=" + Chatlieu + ", Mausac=" + Mausac + ", Ngaynhap=" + Ngaynhap + "]";

}

}

## **Class Quần Áo Nữ**

package bt.qlykhoquanao;

**import** java.io.Serializable;

**import** java.sql.Date;

**public** **class** QuanAoNu **extends** QuanAo **implements** Serializable {

**public** QuanAoNu(String Loai,String Hang,String Style,String size, String chatlieu, String mausac, String ma, Date Ngaynhap) {

**super**(Loai,Hang,Style,size, chatlieu, mausac, ma, Ngaynhap);

}

**public** QuanAoNu() {

}

**public** QuanAoNu(String Loai,String Hang,String style, String size, String chatlieu, String mausac, String ma, java.util.Date parse) {

}

@Override

**public** String toString() {

**return** "QuanAoNu [Loai="+Loai+",Hang="+Hang+",Style="+Style+",Size=" + Size + ", Chatlieu=" + Chatlieu + ", Mausac=" + Mausac + ", Ngaynhap=" + Ngaynhap + "]";

}

}

## **Class Quần Áo Nam**

package bt.qlykhoquanao;

**import** java.io.Serializable;

**import** java.sql.Date;

**public** **class** QuanAoNam **extends** QuanAo **implements** Serializable {

**public** QuanAoNam(String Loai,String Hang,String Style,String size, String chatlieu, String mausac,String Ma, Date Ngaynhap) {

**super**(Loai,Hang,Style,size, chatlieu, mausac,Ma, Ngaynhap);

}

**public** QuanAoNam() {

}

**public** QuanAoNam(String Loai,String Hang,String style, String size, String chatlieu, String mausac, String ma, java.util.Date parse) {

}

@Override

**public** String toString(){

**return** "QuanAoNam[Loai="+Loai+",Hang="+Hang+",Style="+Style+",size="+Size+", mausac="+Mausac+", Chatlieu="+Chatlieu+", Ma="+Ma+", Ngaynhap=" + Ngaynhap + "]";

}

}

## **Class Kho**

**package** qlykhoquanao;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileOutputStream;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.io.ObjectInputStream;

**import** java.io.ObjectOutputStream;

**import** java.io.OutputStream;

**import** java.sql.Date;

**import** java.text.ParseException;

**import** java.text.SimpleDateFormat;

**import** java.util.ArrayList;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Kho {

**private** ArrayList<QuanAo> list = **new** ArrayList<>();

**private** ArrayList<QuanAo> kho;

**public** Kho() {

kho = **new** ArrayList<>();

list = **new** ArrayList<>();

}

**public** **boolean** themDoNam(QuanAo nam) {

**for** (QuanAo n : kho) {

**if** (n.getMa().equals(n.getMa())) {

System.***out***.println("Ma đa ton tai trong kho");

**return** **false**;

}

}

kho.add(nam);

System.***out***.println("Them quan thanh cong");

**return** **true**;

}

**public** **boolean** themDoNu(QuanAo nu){

**for** (QuanAo w : kho) {

**if** (w.getMa().equals(w.getMa())) {

System.***out***.println("Ma da ton tai trong kho");

**return** **false**;

}

}

kho.add(nu);

System.***out***.println("Them ao thanh cong");

**return** **true**;

}

**public** **void** xuatSanPham() {

System.***out***.println("Danh sach san pham:");

**for** (**int** i = 0; i < kho.size(); i++) {

System.***out***.println(kho.get(i));

}

}

**public** **void** xoaSanPham() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Nhap ma san pham can xoa: ");

String Ma=sc.nextLine();

**for** (**int** i = 0; i < kho.size(); i++) {

**if** (kho.get(i).getMa().equals(Ma)) {

kho.remove(i);

System.***out***.println("San pham co ma " + Ma + " da duoc xoa.");

**return**;

}

}

System.***out***.println("Khong tim thay san pham co ma " + Ma);

}

**public** **void** suaSanPham() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Nhap ma san pham can sua: ");

String ma = sc.nextLine();

**for** (**int** i = 0; i < kho.size(); i++) {

**if** (kho.get(i).getMa().equals(ma)) {

System.***out***.println("Nhap thong tin moi cho san pham:");

kho.get(i).nhapthongtin();

System.***out***.println("San pham co ma " + ma + " da duoc sua.");

**return**;

}

}

System.***out***.println("Khong tim thay san pham co ma " + ma);

}

**public** **void** kiemTraSoLuong() {

**if** (kho.isEmpty()) {

System.***out***.println("Kho khong co san pham.");

**return**;

}

System.***out***.println("Tong so san pham trong kho la: " + kho.size());

}

**public** **void** timKiemSanPham() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Nhap ma san pham can tim kiem: ");

String Ma=sc.nextLine();

**for** (**int** i = 0; i < kho.size(); i++) {

**if** (kho.get(i).getMa().equals(Ma)) {

System.***out***.println(kho.get(i));

**return**;

}

}

System.***out***.println("Khong tim thay san pham co ma " + Ma);

}

**public** **void** luuFile(File f) {

**try** {

FileWriter fw = **new** FileWriter(f);

**for** (QuanAo qa : kho) {

**if** (qa **instanceof** QuanAoNam) {

QuanAoNam nam = (QuanAoNam) qa;

fw.write("DoNam,"+nam.getLoai()+","+nam.getHang()+"," +nam.getStyle() +"," + nam.getMa() + "," + nam.getMausac() + "," + nam.getSize() + "," + nam.getChatlieu() +","+nam.getNgaynhap() +"\n");

} **else** **if** (qa **instanceof** QuanAoNu) {

QuanAoNu nu = (QuanAoNu) qa;

fw.write("DoNu,"+nu.getLoai()+","+nu.getHang()+"," +nu.getStyle() +"," + nu.getMa() + "," + nu.getMausac() + "," + nu.getSize() + "," + nu.getChatlieu() +","+nu.getNgaynhap()+"\n");

}

}

System.***out***.println("da ghi file vao " + f.getAbsolutePath());

fw.close();

} **catch** (IOException e) {

e.printStackTrace();

}

}

**public** **void** docFile(File f) {

**try** {

Scanner sc = **new** Scanner(f);

SimpleDateFormat dateFormat = **new** SimpleDateFormat("yyyy-MM-dd");

**while** (sc.hasNextLine()) {

String line = sc.nextLine();

String[] fields = line.split(",");

**if** (fields.length == 9) {

**if** (fields[0].equals("nam")) {

QuanAoNam nam = **new** QuanAoNam(fields[1], fields[2], fields[3], fields[4], fields[5],fields[6] ,fields[7], dateFormat.parse(fields[8]));

themDoNam(nam);

} **else** **if** (fields[0].equals("")) {

QuanAoNu nu = **new** QuanAoNu(fields[1], fields[2], fields[3], fields[4], fields[5],fields[6] ,fields[7], dateFormat.parse(fields[8]));

themDoNu(nu);

}

} **else** {

System.***out***.println("Dong khong hop le: " + line);

}

}

System.***out***.println("Da doc file " + f.getAbsolutePath());

sc.close();

} **catch** (IOException | ParseException e) {

e.printStackTrace();

}

}

**public** **void** xuatThongTin(File f) {

**try** {

Scanner sc = **new** Scanner(f);

SimpleDateFormat dateFormat = **new** SimpleDateFormat("yyyy-MM-dd");

**while** (sc.hasNextLine()) {

String[] fields = sc.nextLine().split(",");

**if** (fields.length == 9) {

String SP = fields[0];

String Loai= fields[1];

String Hang = fields[2];

String style = fields[3];

String ma = fields[4];

String mausac = fields[5];

String size = fields[6];

String chatlieu = fields[7];

Date ngaynhap = **new** java.sql.Date(dateFormat.parse(fields[8]).getTime());

System.***out***.printf("SP: %s,Loai: %s, Hang: %s, Style: %s, Ma: %s, Mau sac: %s, Size: %s, Chat lieu: %s, Ngay nhap: %s\n", SP,Loai,Hang, style, ma, mausac, size, chatlieu, dateFormat.format(ngaynhap));

}

}

sc.close();

} **catch** (IOException | ParseException e) {

e.printStackTrace();

}

}

}

## **Main**

**package** qlykhoquanao;

**import** java.io.File;

**import** java.text.ParseException;

**import** java.util.Scanner;

**public** **class** MAIN{

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

Kho kho = **new** Kho();

Scanner sc = **new** Scanner(System.***in***);

**int** choice = -1;

**while** (choice != 0) {

System.***out***.println("MENU");

System.***out***.println("1. Them san pham");

System.***out***.println("2. Xuat danh sach san pham");

System.***out***.println("3. Xoa san pham");

System.***out***.println("4. Chinh sua thong tin san pham");

System.***out***.println("5. Kiem tra so luong");

System.***out***.println("6. Tim kiem san pham theo ma");

System.***out***.println("7. Sao luu ");

System.***out***.println("8. Khoi phuc");

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

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

choice = sc.nextInt();

sc.nextLine();

**switch** (choice) {

**case** 0:

System.***out***.println("Ket thuc chuong trinh");

**case** 1:

System.***out***.println("Nhap loai san pham muon them (1. Quan Ao nam, 2. Quan Ao nu): ");

**int** type = sc.nextInt();

sc.nextLine();

**if** (type == 1) {

System.***out***.println("Nhap loai quan muon them (1.Quan nam, 2. Ao nam): ");

**int** loai = sc.nextInt();

sc.nextLine();

QuanAoNam nam = **new** QuanAoNam();

nam.nhapthongtin();

kho.themDoNam(nam);

} **else** **if** (type == 2) {

System.***out***.println("Nhap loai ao muon them (1. Quan nu, 2.Ao nu): ");

**int** loai = sc.nextInt();

sc.nextLine();

QuanAoNu nu = **new** QuanAoNu();

nu.nhapthongtin();

kho.themDoNu(nu);

} **else** {

System.***out***.println("Nhap sai loai san pham");

}

**break**;

**case** 2:

kho.xuatSanPham();

**break**;

**case** 3:

kho.xoaSanPham();

**break**;

**case** 4:

kho.suaSanPham();

**break**;

**case** 5:

kho.kiemTraSoLuong();

**break**;

**case** 6:

kho.timKiemSanPham();

**break**;

**case** 7:

System.***out***.println(" nhap ten file");

String f = sc.nextLine();

File newfile=**new** File(f);

kho.luuFile(newfile);

**break**;

**case** 8:

System.***out***.println(" nhap ten file");

String F= sc.nextLine();

File newFile=**new** File(F);

kho.docFile(newFile);

kho.xuatThongTin(newFile);

**break**;

**default**:

System.***out***.println("Nhap sai lua chon");

**break**;

}

}

}

}

# **Kiểm thử**

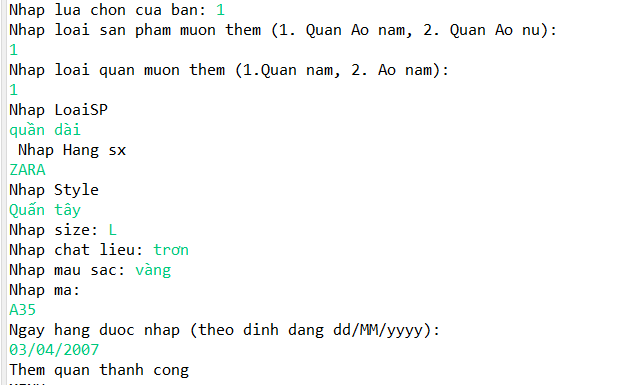
**Dữ liệu Quần Áo Nữ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Mã | Hãng sản xuất | Loại | Size | Chất liệu | Màu sắc | Style | Ngày nhập |
| M01 | GUCCI | Áo tay lỡ | X | Trơn | Trắng | Áo thun | 03/02/2003 |
| M02 | CHANNEL | Quần jeans | XL | Len | Đen | Quần ống loe | 04/05/2010 |
| M03 | ZARA | Áo cổ tròn | L | Nhung | Đỏ | Áo crotop | 03/06/2005 |

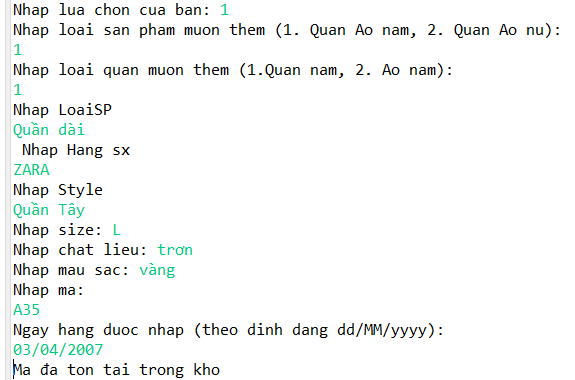
**Dữ liệu Quần Áo Nam**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Mã | Hãng sản xuất | Loại | Size | Chất liệu | Màu sắc | Style | Ngày nhập |
| A33 | GUCCI | Áo cọc | M | Cotton | Trắng | Áo thun | 03/04/2006 |
| A34 | CHANNEL | Áo tay dài | XL | Nhung | Đen | Áo sơ mi | 03/02/2005 |
| A35 | ZARA | Quần dài | L | Trơn | Vàng | Quần Tây | 03/04/2007 |

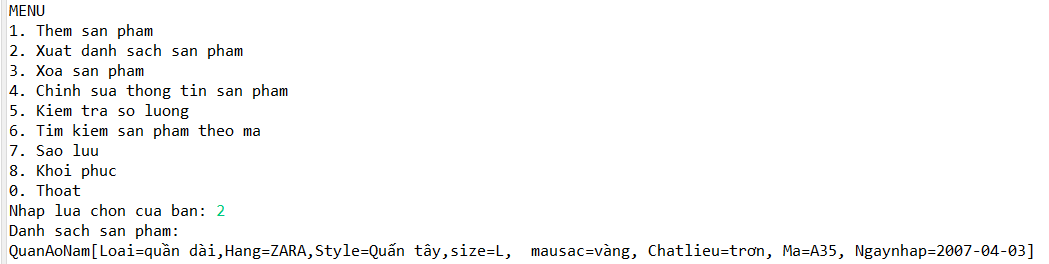
## **Nhập thông tin Quần Áo**



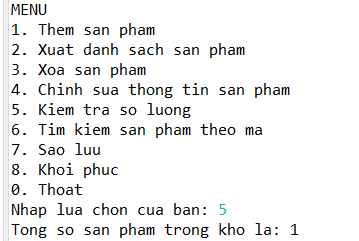
* Xảy ra lỗi nếu nhập trùng mã



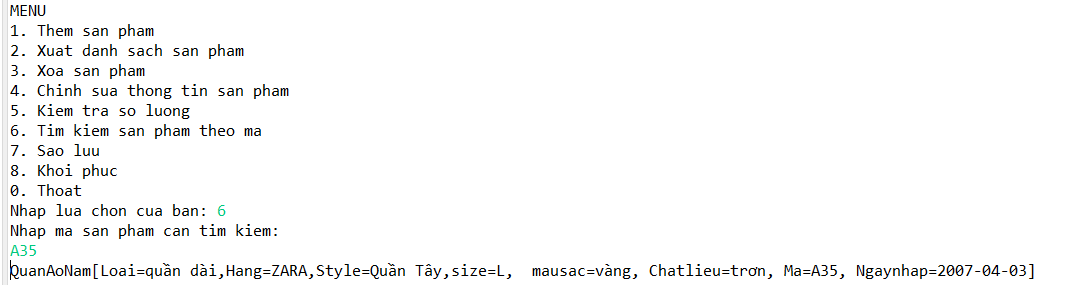
## **Xuất thông tin Quần Áo**



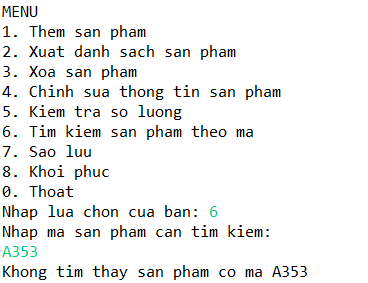
## **Số lượng sản phẩm Quần Áo**



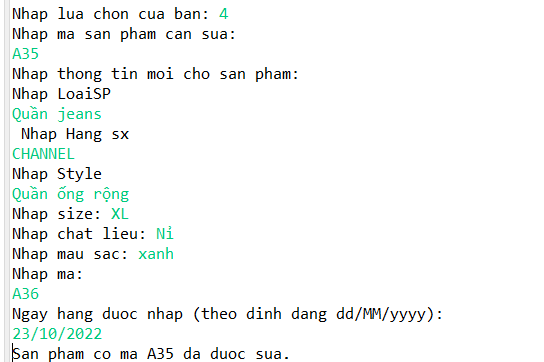
## **Tìm kiếm sản phẩm theo mã**



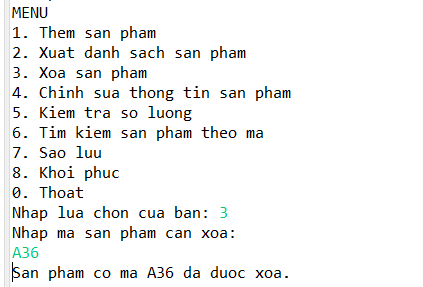
* Nếu tìm không thấy mã thì không cho ra kết quả



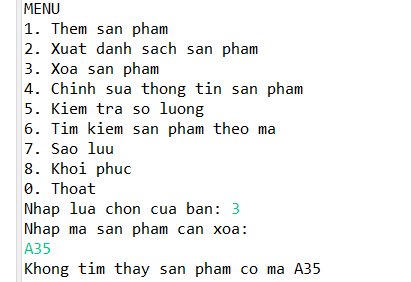
## **Chỉnh sửa thông tin**



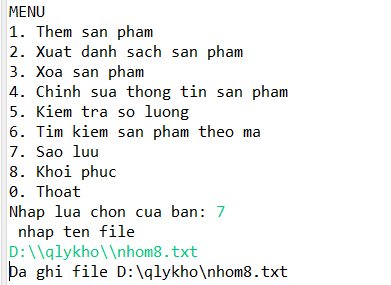
## **Xóa một thông tin**



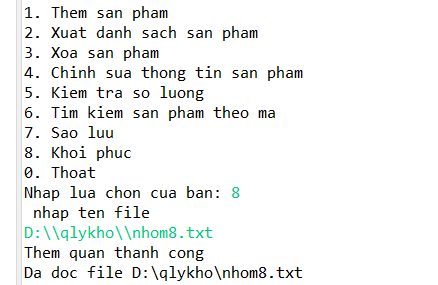
* Nếu không nhập đúng mã thì sẽ không xóa được

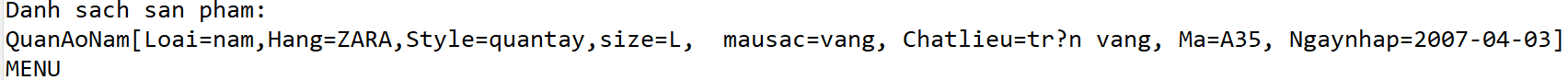


## **Sao lưu thông tin**



## **Khôi phục thông tin**





# **Tài liệu tham khảo**

[1] Tài liệu trên lớp

[2] Các trang web tham khảo: <https://www.w3schools.com/>

<https://vnexpress.net/chu-de/xu-huong-thoi-trang-1242>

<https://www.ecount.com/>

[3] Kênh youtube: <https://www.youtube.com/@TITVvn>