**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**

**QUẢN LÝ KHO ĐIỆN THOẠI CAO CẤP**

Thành viên của nhóm:

1. Huỳnh Nhật Huy (1050080136)
2. Trần Duy Kha (1050080139)
3. Trần Quốc Bảo (1050080129)

Lớp: 10ĐH\_CNTT4

***TP. Hồ Chí Minh, 27 tháng 3 năm 2022***

**MỤC LỤC**

[**I.** **Giới thiệu đồ án** 4](#_Toc130759519)

[**1.** **Định nghĩa vấn đề** 4](#_Toc130759520)

[**2.** **Phạm vi của đồ án** 4](#_Toc130759521)

[**3.** **Mục tiêu, sự cần thiết của đề tài, tại sao lại chọn đề tài này?** 4](#_Toc130759522)

[**II.** **Phân tích đề tài** 5](#_Toc130759523)

[**1.** **Phân tích yêu cầu hệ thống** 5](#_Toc130759524)

[**2.** **Yêu cầu chức năng** 5](#_Toc130759525)

[**3.** **Yêu cầu phi chức năng** 6](#_Toc130759526)

[**4.** **Các công việc cần giải quyết** 7](#_Toc130759527)

[**III.** **Mô tả thuật toán thao tác** 9](#_Toc130759528)

[**1.** **Tìm kiếm điện thoại theo tên** 9](#_Toc130759529)

[**2.** **Thêm mẫu điện thoại mới vào danh sách** 9](#_Toc130759530)

[**3.** **Xóa một mẫu điện thoại trong mảng qua id** 10](#_Toc130759531)

[**4.** **Kiểm tra một mẫu điện thoại có tồn tại trong kho hay không** 10](#_Toc130759532)

[**5.** **Sếp giảm điện thoại theo giá bán** 10](#_Toc130759533)

[**6.** **Nhập kho** 10](#_Toc130759534)

[**7.** **Xuất kho** 10](#_Toc130759535)

[**IV.** **Cài đặt bài toán** 11](#_Toc130759536)

[**1.** **Class dienThoai** 11](#_Toc130759537)

[**2.** **Class nhapKho** 14](#_Toc130759538)

[3. **Class xuatKho** 15](#_Toc130759539)

[4. **Class danhSachDienThoai** 17](#_Toc130759540)

[**5.** **Class danhSachNhapKho** 19](#_Toc130759541)

[**6.** **Class danhSachXuatKho** 20](#_Toc130759542)

[**7.** **Class main** 21](#_Toc130759543)

[**V.** **Kiểm thử lập bộ test** 26](#_Toc130759544)

[**1.** **Danh sách điện thoại:** 26](#_Toc130759545)

[**2.** **Search điện thoại:** 26](#_Toc130759546)

[**3.** **Nhập kho:** 26](#_Toc130759547)

[**4.** **Xuất kho :** 27](#_Toc130759548)

[**5.** **Thêm mẫu điện thoại mới:** 27](#_Toc130759549)

[**6.** **Xóa điện thoại:** 27](#_Toc130759550)

[**7.** **Kiểm tra điện thoại có tồn tại trong kho hay không:** 28](#_Toc130759551)

[**8.** **Sắp xếp điện thoại theo giá:** 28](#_Toc130759552)

*BẢNG PHÂN CÔNG CÔNG VIỆC*

* Ghi chú : Ngoài công việc được phân công, mọi thành viên phải tham gia đọc

và góp ý nội dung của các thành viên khác.

* Tất cả công việc đều theo quyết định của nhóm trưởng Huỳnh Nhật Huy,

nếu có sai sót trong bài mong thầy thông cảm và em xin chịu trách nhiệm.

|  |  |  |  |
| --- | --- | --- | --- |
| **SST** | **Họ tên** | **MSSV** | **Công việc** |
| 1 | Huỳnh Nhật Huy | 1050080136 | Thiết kế form |
| 2 | Trần Duy Kha | 1050080139 | Code |
| 3 | Trần Quốc Bảo | 1050080129 | Code |

**LỜI MỞ ĐẦU**

* Ngày nay Công nghệ thông tin đã và đang phát triển với tốc độ chóng mặt. Công nghệ thông tin đang được ứng dụng nhiều trong nhiều lĩnh vực khác nhau như nghiên cứu khoa học, phát triển kinh tế.
* Do vậy, công tác quản lý kho là một công tác không thể thiếu của tất cả các cá nhân tổ chức kinh doanh. Với các lý do ấy thì phát triển Công nghệ thông tin đã trở thành một ngành quan trọng và không thể thiếu đặc biệt là Công nghệ phần mềm.
* Sự ra đời của các sản phẩm phần mềm đặc biệt là các phần mềm ứng dụng như quản lý kho trong vài năm gần đây mang lại nhiều thuận lợi trong công tác quản lý hàng hóa tránh sự sai sót, nhầm lẫn dẫn đến thất thu, mất mát.
* Tuy nhiên bên cạnh những tiện lợi của phần mềm mang lại, vẫn còn nhiều khó khăn, nhược điểm cần được khắc phục như chúng em còn hạn chế về trình độ cũng như kinh nghiêm.
* Tuy vậy nhưng chúng em đã chọn đề tài “ Quản lý kho điện thoại cao cấp” một đề tài mà các cá nhân cũng như các tổ chức kinh doanh cần. Để có thể tích lũy kinh nghiệm cũng như kiến thức về lập trình hứng đối tượng cho bản thân chúng em thông qua việc làm đồ án.

Chúng em xin chân thành cảm ơn.

1. **Giới thiệu đồ án**
2. **Định nghĩa vấn đề**

Trong những năm gần đây, sự phát triển mạnh mẽ của công nghệ thông tin đã làm cho khả năng thu thập và lưu trữ thông tin cảu hệ thống thông tin tăng một cách nhanh chóng. Bên cạnh đó, việc chuyển đổi số một cách chóng mặt của các hoạt động sản xuất, kinh doanh cũng như nhiều lĩnh vực hoạt động khác đã tạo ra một lượng dữ liệu cần lưu trữ và xử lý khổng lồ.

Trong đó lập trình là công đoạn quan trọng chủ chốt và không thể thiếu để tạo ra sản phẩm phần mềm. phần mềm càng trở nên đa dạng và ngành công nghiệp phầm mềm càng phát triển thì người ta càng thấy rõ tầm quan trọng của phương pháp lập trình. Phương pháp lập trình tốt không chỉ đảm bảo tạo ra phần mềm tốt mà còn hỗ trợ thiết kế phần mềm có tính mở và hỗ trợ khả năng sự dụng lại các mô đun. Nhờ đó chúng ta có thể dễ dàng bảo trì, nâng cấp phần mềm cũng như giảm chi phí phát triển phần mềm.

1. **Phạm vi của đồ án**

Ta sẽ phát triển ứng dụng với phạm vi nghiên cứu cụ thể ở các chức năng quản lý điện thoại cao cấp, quản lý nhập xuất kho.

1. **Mục tiêu, sự cần thiết của đề tài, tại sao lại chọn đề tài này?**
   1. *Mục tiêu*

Xây dựng nên các chức năng cơ bản trong việc quản lý kho thực phẩm có áp dụng được lập trình hướng đối tượng. các chức năng của ứng dụng có thể hỗ trợ cho người quản lý thuận lợi trong việc xem xét nắm rõ tình hình một cách dễ dàng hơn, nhanh chóng và ít sai sót hơn về các điện thoại cao cấp trong kho trong quá trình làm việc.

* 1. *Sự cần thiết*

Ngành công nghiệp đang tăng mạnh và cũng cần một phần mềm tốt để có thể quản lý các kho hàng phục vụ cho mục đích kinh doanh một cách nhanh chóng và tiện lợi, tránh gây ra các sai sót không đáng có.

* 1. *Tại sao lại chọn đề tài này?*

Việc nghiên cứu để ra các phương pháp, công cụ mới hỗ trợ con người khám phá, phân tích, tổng hợp thông tin nhằm để tìm và rút ra các tri thức các quy luật là một nhu cầu cần thiết. Từ đó giúp cho nhà quản lý có cái nhìn rõ ràng, tổng quan hơn về dữ liệu, có thể đưa ra những nhận định, quyết định cũng như dự đoán một cách chính xác mang tính chiến lực. Do điện thoại cao cấp là một sản phẩm không thể thiếu trong thời đại ngày nay nên chúng em thấy đây là một hướng đi có tiềm năng và còn có thể phát triển thêm nữa trong tương lai, nên chúng em đã chọn đề tài: “Quản lý điện thoại cao cấp” trong đợt làm đồ án môn học này.

1. **Phân tích đề tài**
2. **Phân tích yêu cầu hệ thống**
   1. *Công việc chính của quản lý điện thoại cao cấp*

* Quản lý nhập hàng:

Sau mỗi lần nhập hàng, quản lý kho sẽ ghi lại chi tiết về sản phẩm và ngày hàng đó được nhập vào kho để tiện theo dõi. Sau khi về kho bộ phận kho chịu trách nhiệm kiểm tra sản phẩm về số lượng của sản phẩm. Sau khi kiểm kê hoàn thành, quản lý kho sẽ tiến hành lập phiếu nhập sản phẩm và đưa vào kho. Việc theo dõi sẽ được bộ phận kho kiểm kê liên tục, theo dõi có thể diễn ra hàng ngày, hàng tuần, hàng tháng. Quản lý kho chịu trách nhiệm viết báo cáo hằng ngày.

* Quản lý xuất hàng:

Bộ phận kho sẽ kiểm kế số lượng sản phẩm trước khi xuất hàng. Bộ phận quản lý khó sẽ tiến hành viết phiếu xuất kho. Khi xuất hàng thông tin về hàng xuất cũng được ghi lại rõ ràng để tiện cho việc đối chiếu kiểm tra.

1. **Yêu cầu chức năng**
   1. *Chức năng thêm điện thoại cao cấp*

Trong chức năng này nhân viên sẽ nhập thông tin các sản phẩm mới lên hệ thống.

* ID điện thoại cao cấp
* Tên điện thoại cao cấp
* Ngày nhập
* Số lượng
  1. *Chức năng sửa điện thoại cao cấp*

Sau khi nhập xong nếu có sai sót gì thì nhân viên sẽ chỉnh sửa lên hệ thống.

* ID điện thoại cao cấp
* Tên điện thoại cao cấp
* Ngày nhập
* Số lượng
  1. *Chức năng xóa điện thoại cao cấp*

Khi một mặt hàng trong kho đã hết, người quản lý sẽ tiến hành xóa loại hàng đó ra khỏi danh sách hàng hóa đó còn trong kho.

* ID điện thoại cao cấp
* Tên điện thoại cao cấp
* Ngày nhập
* Số lượng

1. **Yêu cầu phi chức năng**
   1. *Độ tin cây*

Phần mềm được phát triển bởi các sinh viên chua có kinh nghiệm thực tế cũng như còn hạn chết về mặt kiến thức nên độ tin cậy vẫn chưa được cao và cũng như các chức năng về mặt giao diện có thể uy tín như các công ty có chuyên môn khác.

* 1. *Khả năng sử dụng*

Phần mềm có khả năng sử dụng khá đơn giản, nhân viên có thể sử dụng các khả năng của hệ thống một cách ổn định, dữ liệu có thể cập nhập nhanh chóng dễ dàng. Ngoài ra phần mềm còn được thiết kế với giao diện thân thiện, dễ nhìn đối với người sử dụng, có thể hiển thị thông tin một cách rõ ràng.

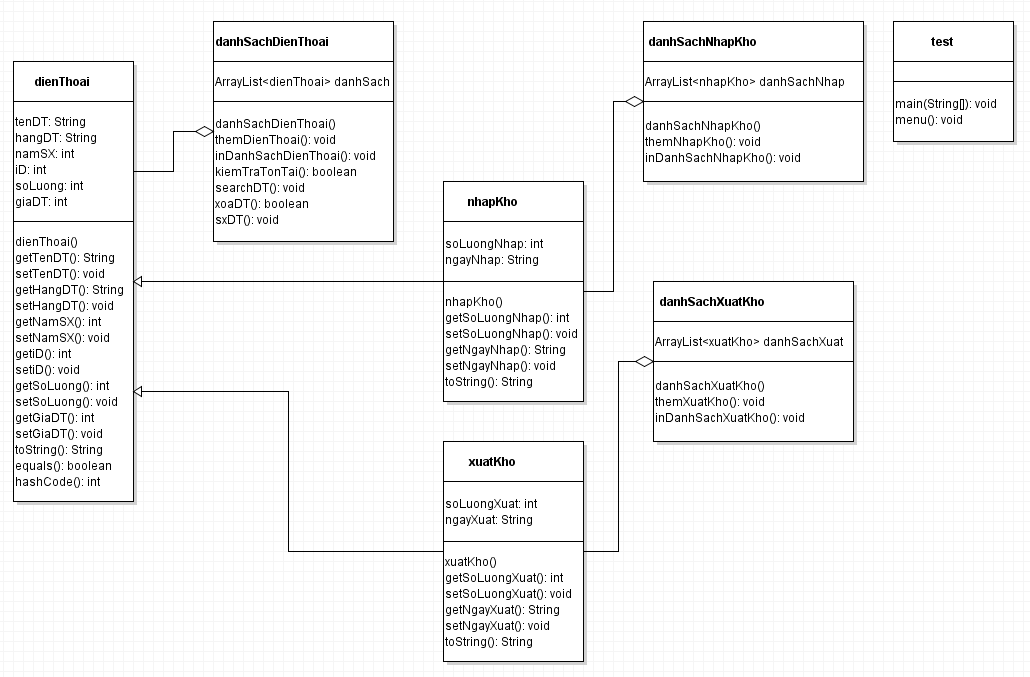
* 1. *Hiệu suất*

Hiệu năng của hệ thống khá ổn định, các chức năng được triển khai khá hoàn thiện, tốc độ phản phồi của phần mềm nhanh.

1. **Các công việc cần giải quyết**
   1. *Phát biểu bài toán*

Các đối tượng cần quản lý:

* Điện thoai: Tên điện thoại, hãng điện thoại, năm sản xuất, ID, số lượng, giá điện thoại.
* Nhập kho: Số lượng nhập, ngày nhập.
* Xuất kho: Số lượng xuất, ngày xuất.
* Nhập kho và xuất kho kế thừa lớp điện thoại
  1. *Xác định các lớp của bài toán*
     1. Class dienThoai
* Thuộc tính: String tenDT, String hangDT, int namSX, int iD, int soLuong, int giaDT.
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private.
* String toString(): ghi ra thông tin.
  + 1. Class danhSachDienThoai
* Thuộc tính: ArrayList<dienThoai> danhSach
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Void themDienThoai().
* Void inDanhSachDienThoai().
* Boolean kiemTraTonTai().
* Void searchDT().
* Boolean xoaDT().
* Void sxDT().
  + 1. Class danhSachNhapKho
* Thuộc tính: ArrayList<nhapKho> danhSachNhap.
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Void themNhapKho().
* Void inDanhSachNhapKho().
  + 1. Class danhSachXuatKho
* Thuộc tính: ArrayList<xuatKho> danhSachXuat.
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Void ThemXuatKho().
* Void inDanhSachXuatKho().
  + 1. Class nhapKho kết thừa từ class dienThoai
* Thuộc tính: int soLuongNhap, String ngayNhap.
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private.
* String toString(): ghi ra thông tin.
  + 1. Class xuatKho kết thừa từ class dienThoai
* Thuộc tính: int soLuongXuat, String ngayXuat.
* Phương thức:
* Constructor: Hàm tạo có đối số và hàm tạo không đối số.
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private.
* String toString(): ghi ra thông tin.



1. **Mô tả thuật toán thao tác**
2. **Tìm kiếm điện thoại theo tên**

Sử dùng vòng lập for để dò từng phần tử trong danh sách, sau đó sử dụng phương thức indexOf để so sánh chuỗi kí tự (Tên điện thoại) vừa nhập vào có giống với phần tử nào trong danh sách hay không. Nếu có thì in thông tin của điện thoại đó ra màn hình.

1. **Thêm mẫu điện thoại mới vào danh sách**

Nhập vào thông tin của điện thoại mới. Sau đó sử dụng phương thức add để thêm phần tử mới vào danh sách.

1. **Xóa một mẫu điện thoại trong mảng qua id**

Nhập id của mẫu điện thoại cần xóa. Nếu trong danh sách không tồn tại id thì sẽ trả về false, còn ngược lại sẽ trả về true (boolean), sau đó in thông tin điện thoại cần xóa ra màn hình và sử dụng lệnh remove để xóa phần tử đó ra khỏi danh sách.

1. **Kiểm tra một mẫu điện thoại có tồn tại trong kho hay không**

Nhập id của mẫu điện thoại cần kiểm tra. Sau đó sử dụng phương thức contains để kiểm tra trong danh sách có tồn tại phần tử đó hay không, nếu có trả lại true, nếu không trả lại false (boolean).

1. **Sếp giảm điện thoại theo giá bán**

* Sử dụng Collections Sort trong ArrayList<dienThoai>
* Tạo phương thức int compare(dienThoai dt1, dienThoai dt2)

Nguyenvong a1, Nguyenvong a2)

* Sử dụng lệnh if để so sánh từng trường hợp:
* Nếu dt1.getGiaDT() < dt2.getGiaDT() thì return 1
* Nếu dt1.getGiaDT() > dt2.getGiaDT() thì return -1
* Cuối cùng in danh sách đã được sắp xếp ra màn hình.

1. **Nhập kho**

Tạo ra một danh sách nhập kho. Nhập kho bao gồm 2 phương thức là nhập dữ liệu và in danh sách.

1. **Xuất kho**

Tạo ra một danh sách xuất kho. Xuất kho bao gồm 2 phương thức nhập dữ liệu và in danh sách ra màn hình.

1. **Cài đặt bài toán**
2. **Class dienThoai**

package QLKho;

import java.util.Objects;

public class dienThoai {

private String tenDT;

private String hangDT;

private int namSX;

private int iD;

private int soLuong;

private int giaDT;

public dienThoai(int iD) {

this.iD = iD;

}

dienThoai(){}

dienThoai(String tenDT, String hangDT, int namSX, int iD, int soLuong, int giaDT){

super();

this.hangDT=hangDT;

this.namSX=namSX;

this.tenDT=tenDT;

this.iD=iD;

this.soLuong=soLuong;

this.giaDT=giaDT;

}

public String getTenDT() {

return tenDT;

}

public void setTenDT(String tenDT) {

this.tenDT = tenDT;

}

public String getHangDT() {

return hangDT;

}

public void setHangDT(String hangDT) {

this.hangDT = hangDT;

}

public int getNamSX() {

return namSX;

}

public void setNamSX(int namSX) {

this.namSX = namSX;

}

public int getiD() {

return iD;

}

public void setiD(int iD) {

this.iD = iD;

}

public int getSoLuong() {

return soLuong;

}

public void setSoLuong(int soLuong) {

this.soLuong = soLuong;

}

public int getGiaDT() {

return giaDT;

}

public void setGiaDT(int giaDT) {

this.giaDT = giaDT;

}

@Override

public String toString() {

return "dienThoai{" + "tenDT=" + tenDT + ", hangDT=" + hangDT + ", namSX=" + namSX + ", iD=" + iD + ", soLuong=" + soLuong + ", giaDT=" + giaDT + '}';

}

@Override

public boolean equals(Object obj) {

if (this == obj) {

return true;

}

if (obj == null) {

return false;

}

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

return false;

}

dienThoai other = (dienThoai) obj;

return Objects.equals(this.iD, other.iD);

}

@Override

public int hashCode() {

int hash = 3;

hash = 59 \* hash + this.iD;

return hash;

}

}

1. **Class nhapKho**

package QLKho;

public class nhapKho extends dienThoai{

private int soLuongNhap;

private String ngayNhap;

public nhapKho(int soLuongNhap, String ngayNhap) {

this.soLuongNhap = soLuongNhap;

this.ngayNhap=ngayNhap;

}

nhapKho(){}

nhapKho(String tenDT, String hangDT, int namSX, int iD, int soLuong, int giaDT, int soLuongNhap, String ngayNhap){

super(tenDT, hangDT, namSX, iD, soLuong, giaDT);

this.soLuongNhap=soLuongNhap;

this.ngayNhap=ngayNhap;

}

public int getSoLuongNhap() {

return soLuongNhap;

}

public void setSoLuongNhap(int soLuongNhap) {

this.soLuongNhap = soLuongNhap;

}

public String getNgayNhap() {

return ngayNhap;

}

public void setNgayNhap(String ngayNhap) {

this.ngayNhap = ngayNhap;

}

@Override

public String toString() {

return(this.getClass().getName()+" ["+super.toString().substring((this.getClass().getSuperclass().getName().length()-3

), (super.toString().length())-1)+ ", soLuongNhap="+soLuongNhap + ", NgayNhap="+ ngayNhap+"]");

}

}

1. **Class xuatKho**

package QLKho;

public class xuatKho extends dienThoai{

private int soLuongXuat;

private String ngayXuat;

public xuatKho(int soLuongXuat, String ngayXuat) {

this.soLuongXuat = soLuongXuat;

this.ngayXuat=ngayXuat;

}

xuatKho(){}

xuatKho(String tenDT, String hangDT, int namSX, int iD, int soLuong, int giaDT ,int soLuongXuat ,String ngayXuat){

super(tenDT, hangDT, namSX, iD, soLuong, giaDT);

this.soLuongXuat=soLuongXuat;

this.ngayXuat=ngayXuat;

}

public int getSoLuongXuat() {

return soLuongXuat;

}

public void setSoLuongXuat(int soLuongXuat) {

this.soLuongXuat = soLuongXuat;

}

public String getNgayXuat() {

return ngayXuat;

}

public void setNgayXuat(String ngayXuat) {

this.ngayXuat = ngayXuat;

}

@Override

public String toString() {

return(this.getClass().getName()+" ["+super.toString().substring((this.getClass().getSuperclass().getName().length()-3

), (super.toString().length())-1)+ ", soLuongXuat="+soLuongXuat +", NgayXuat="+ ngayXuat+"]");

}

}

1. **Class danhSachDienThoai**

package QLKho;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Comparator;

public class danhSachDienThoai{

private ArrayList<dienThoai> danhSach;

public danhSachDienThoai(ArrayList<dienThoai> danhSach) {

this.danhSach = danhSach;

}

public danhSachDienThoai() {

this.danhSach = new ArrayList<dienThoai>();

}

public void themDienThoai(dienThoai dt){

this.danhSach.add(dt);

}

public void inDanhSachDienThoai(){

for( dienThoai DienThoai : danhSach){

System.out.println(DienThoai);

}

}

public boolean kiemTraTonTai(dienThoai dt){

return this.danhSach.contains(dt);

}

public void searchDT(String tenDT){

for ( dienThoai DienThoai : danhSach){

if(DienThoai.getTenDT().indexOf(tenDT)>=0){

System.out.println(DienThoai);

}

}

}

public boolean xoaDT(dienThoai dt){

return this.danhSach.remove(dt);

}

public void sxDT(){

Collections.sort(this.danhSach, new Comparator<dienThoai>(){

@Override

public int compare(dienThoai dt1, dienThoai dt2){

if(dt1.getGiaDT()<dt2.getGiaDT()){

return 1;

}

else if(dt1.getGiaDT() > dt2.getGiaDT()){

return -1;

}

else{

return 0;

}

}

});

}}

1. **Class danhSachNhapKho**

package QLKho;

import java.util.ArrayList;

public class danhSachNhapKho {

private ArrayList<nhapKho> danhSachNhap;

public danhSachNhapKho(ArrayList<nhapKho> danhSachNhap) {

this.danhSachNhap = danhSachNhap;

}

public danhSachNhapKho() {

this.danhSachNhap = new ArrayList<nhapKho>();

}

public void themNhapKho(nhapKho nk){

this.danhSachNhap.add(nk);

}

public void inDanhSachNhapKho(){

for( nhapKho NhapKho : danhSachNhap){

System.out.println(NhapKho);

}

}

}

1. **Class danhSachXuatKho**

package QLKho;

import java.util.ArrayList;

public class danhSachXuatKho {

private ArrayList<xuatKho> danhSachXuat;

public danhSachXuatKho(ArrayList<xuatKho> danhSachXuat) {

this.danhSachXuat = danhSachXuat;

}

public danhSachXuatKho() {

this.danhSachXuat = new ArrayList<xuatKho>();

}

public void themXuatKho(xuatKho xk){

this.danhSachXuat.add(xk);

}

public void inDanhSachXuatKho(){

for( xuatKho XuatKho : danhSachXuat){

System.out.println(XuatKho);

}

}

}

1. **Class main**

package QLKho;

import java.util.Scanner;

public class main {

public static void menu(){

danhSachDienThoai dsdt = new danhSachDienThoai();

danhSachNhapKho dsnk = new danhSachNhapKho();

danhSachXuatKho dsxk = new danhSachXuatKho();

Scanner sc = new Scanner(System.in);

boolean luachon=true;

do{

System.out.println("#1.Danh sach dien thoai");

System.out.println("#2.Search dien thoai");

System.out.println("#3.Nhap kho");

System.out.println("#4.Xuat kho");

System.out.println("#5.Add mau dien thoai moi");

System.out.println("#6.Xoa dien thoai");

System.out.println("#7.Kiem tra dien thoai co ton tai trong kho hay khong");

System.out.println("#8.Sap xep dien thoai theo gia");

System.out.println("Chon chuc nang");

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

int chon = sc.nextInt();

sc.nextLine();

switch(chon){

case 1 -> {

dsdt.inDanhSachDienThoai();

}

case 2 -> {

System.out.println("Nhap ten dien thoai: ");

String tenDT = sc.nextLine();

System.out.println("Ket qua tim kiem: ");

dsdt.searchDT(tenDT);

}

case 3 -> {

System.out.println("Nhap ten dien thoai: ");

String tenDT = sc.nextLine();

System.out.println("Nhap hang dien thoai: ");

String hangDT = sc.nextLine();

System.out.println("Nhap nam san xuat: ");

int namSX = sc.nextInt();

System.out.println("Nhap ID: ");

int iD = sc.nextInt();

System.out.println("Nhap so luong: ");

int soLuong = sc.nextInt();

System.out.println("Nhap don gia dien thoai: ");

int giaDT = sc.nextInt();

System.out.println("So luong nhap: ");

int soLuongNhap = sc.nextInt();

System.out.println("nhap ngay nhap hang: ");

String ngayNhap = sc.nextLine();

nhapKho nk = new nhapKho(tenDT, hangDT, namSX, iD, soLuong, giaDT, soLuongNhap, ngayNhap);

dsnk.themNhapKho(nk);

dsnk.inDanhSachNhapKho();

}

case 4 -> {

System.out.println("Nhap ten dien thoai: ");

String tenDT = sc.nextLine();

System.out.println("Nhap hang dien thoai: ");

String hangDT = sc.nextLine();

System.out.println("Nhap nam san xuat: ");

int namSX = sc.nextInt();

System.out.println("Nhap ID: ");

int iD = sc.nextInt();

System.out.println("Nhap so luong: ");

int soLuong = sc.nextInt();

System.out.println("So luong: ");

int soLuongNhap = sc.nextInt();

System.out.println("Nhap gia dien thoai: ");

int giaDT = sc.nextInt();

System.out.println("So luong xuat hang: ");

int soLuongXuat = sc.nextInt();

System.out.println("nhap ngay xuat hang: ");

String ngayXuat = sc.nextLine();

xuatKho nx = new xuatKho(tenDT, hangDT, namSX, iD, soLuong, giaDT, soLuongXuat, ngayXuat);

dsxk.themXuatKho(nx);

dsxk.inDanhSachXuatKho();

}

case 5 -> {

System.out.println("Nhap ten dien thoai: ");

String tenDT = sc.nextLine();

System.out.println("Nhap hang dien thoai: ");

String hangDT = sc.nextLine();

System.out.println("Nhap nam san xuat: ");

int namSX = sc.nextInt();

System.out.println("Nhap ID: ");

int iD = sc.nextInt();

System.out.println("Nhap so luong: ");

int soLuong = sc.nextInt();

System.out.println("Nhap gia dien thoai: ");

int giaDT = sc.nextInt();

dienThoai dt = new dienThoai(tenDT, hangDT, namSX, iD, soLuong, giaDT);

dsdt.themDienThoai(dt);

}

case 6 -> {

System.out.println("Nhap id can xoa: ");

int iD = sc.nextInt();

dienThoai dt = new dienThoai(iD);

System.out.println("Xoa dien thoai trong danh sach: "+dsdt.xoaDT(dt));

}

case 7 -> {

System.out.println("Nhap ID dien thoai: ");

int iD = sc.nextInt();

dienThoai dt = new dienThoai(iD);

System.out.println("Kiem tra dien thoai co trong kho: " + dsdt.kiemTraTonTai(dt));

}

case 8 -> {

dsdt.sxDT();

dsdt.inDanhSachDienThoai();

}

default -> System.exit(0);

}

}while(luachon);

}

public static void main(String[] args) {

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

menu();

}

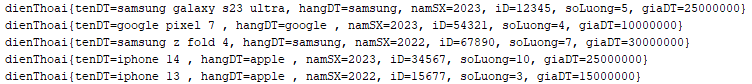
}

1. **Kiểm thử lập bộ test**

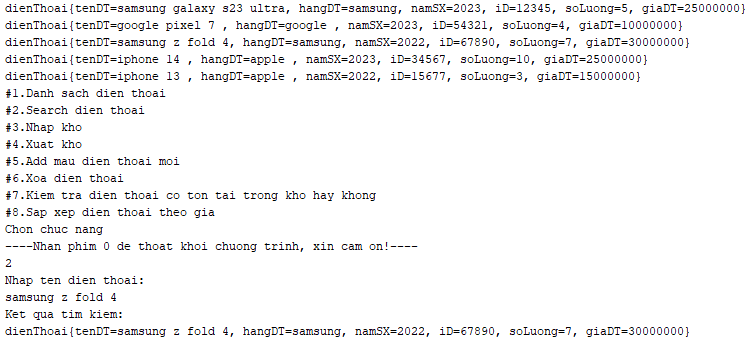
Thông tin điện thoại

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tên điện thoại | Hãng điện thoại | Năm sản xuất | ID | Số lượng | Giá điện thoại |
| Samsung galaxy s23 ultra | Samsung | 2023 | 12345 | 5 | 25.000.000 |
| Google pixel 7 | Google | 2023 | 54321 | 4 | 10.000.000 |
| Samsung z fold 4 | Samsung | 2022 | 67890 | 7 | 30.000.000 |
| Iphone 14 | Apple | 2023 | 34567 | 10 | 25.000.000 |
| Iphone 13 | Apple | 2022 | 15677 | 3 | 15.000.000 |

1. **Danh sách điện thoại:**



1. **Search điện thoại:**



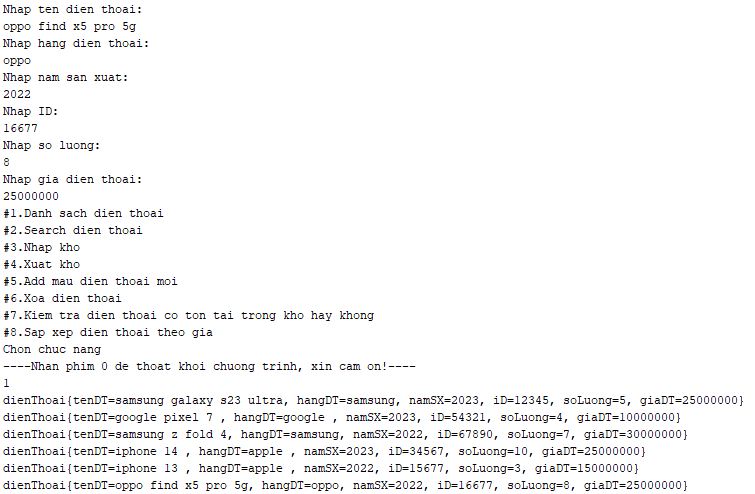
1. **Nhập kho:**



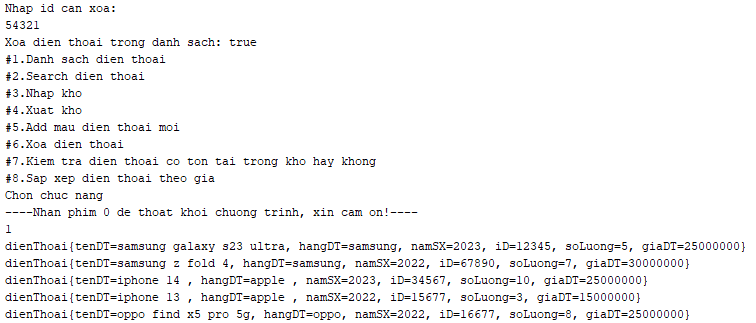
1. **Xuất kho :**



1. **Thêm mẫu điện thoại mới:**



1. **Xóa điện thoại:**



1. **Kiểm tra điện thoại có tồn tại trong kho hay không:**





1. **Sắp xếp điện thoại theo giá:**

