| **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 HƯỚNG ĐỐI TƯỢNG**  **QUẢN LÝ KHO ĐIỆN THOẠI DÒNG**  **CAO CẤP**  Thành viên nhóm:  1.Nguyễn Thanh Hà (1050080048)  2.Trần Công Đạt (1050080047)  3.Phạm Ngọc Hà Minh (1050080061)  Lớp: 10 CNTT2    Giáo Viên Phụ Trách :  Ths Phạm Trọng Huynh  **TP. Hồ Chí Minh, ngày 26/ tháng 3 năm 2023** |
| --- |

**MỤC LỤC**

[**I. Mô Tả Bài Toán 4**](#_xbhi8nca3j9i)

[**II.Xác định các lớp của bài toán 5**](#_t11nh6dcpgui)

[**1. Package quanlikho 5**](#_x328hg8w262h)

[1.1. Class Phone 5](#_2poxs5jv2hx1)

[1.2. Class IOS kế thừa từ class Phone 5](#_ftt1rwj5ubm3)

[1.3.Class Android kế thừa từ class Phone 5](#_7tiyt8k159zw)

[1.4.Class Phieu 5](#_9valmma7pui)

[1.5.Class PhieuNhap kế thừa từ class Phieu 6](#_ogimli5mcdr8)

[1.6. Class PhieuXuat kế thừa từ class Phieu 6](#_sp8gsmyqdrfh)

[1.7.Class Quanli 6](#_kg2ruw44cap8)

[**2. Package DAO 7**](#_5dp27nyhhzor)

[2.1. Class DAOInterface 7](#_o2utlgrvgqtf)

[2.2. Class DAOPhone implements class DAOInterface 7](#_dmmtw319mbg4)

[2.3 Class DAOPhieu implements class DAOInterface 7](#_qcbi0yi5byzy)

[**III. Mô tả thuật toán 8**](#_3tbzljl9mlz7)

[1. Chỉnh sửa thông tin điện thoại 8](#_9pkxx9obg8wk)

[2. Hiển thị danh sách điện thoại hiện có trong cơ sở dữ liệu kho 8](#_5njsierz1hni)

[3. Tìm kiếm 8](#_peknludmwvd8)

[4. Sắp xếp 8](#_6wuerdf2mvuq)

[**IV. Cài đặt bài toán 11**](#_6gztndxjfh8u)

[**1.Package quanlikho 11**](#_a7ta1zb3u2n7)

[1.1. Class Phone 11](#_bd4loa1j4ld0)

[1.2. Class IOS 15](#_dem7e63xohtn)

[1.3. Class Android 15](#_3ckcdsgrdbu8)

[1.4. Class Phieu 16](#_qeoidmf0o6w7)

[1.5. Class PhieuNhap 19](#_is4akh98m7s4)

[1.6.Class PhieuXuat 20](#_xz3z5ac0w9yu)

[1.7. Class Quanli 21](#_rcs638ch1z83)

[1.7. Class Main 26](#_wnwz3773879o)

[**2. Package DAO 26**](#_lwhbn8cm6b17)

[2.1. Class DAOInterface 26](#_rn7meklcdce8)

[2.2. Class DAOPhone 27](#_r8kxu6dzl35c)

[2.3. Class DAOPhieu 37](#_mog253x992qq)

[**V. Kiểm thử 48**](#_lphfeze4qvrn)

[1. Thêm điện thoại, phiếu nhập: **48**](#_mop26sn62x2g)

[2. Thêm điện thoại, phiếu xuất 49](#_szv72lz0jy0c)

[3. Hiển thị danh sách điện thoại nhập, xuất **49**](#_466slrtop49t)

[**VI. Tài liệu tham khảo 51**](#_p3uw3uaopq0u)

# I. Mô Tả Bài Toán

Hằng năm , điện thoại nói riêng và các dòng điện tử nói chung ngày càng phát triển .Vì thế sẽ có hàng ngàn mẫu điện thoại và các thiết bị điện tử ngày càng ra đời , Số lượng các sản phẩm quá lớn đòi hỏi công tác quản lý sản phẩm phải ngày càng tối ưu và hiệu quả hơn, đặc biệt là về công tác quản lý kho. trở nên khó khăn . Vì quản lý kho đóng một vai trò quan trọng trong kinh doanh. Sự thiếu hụt hay dư thừa hàng tồn kho đều có ảnh hưởng rất lớn đến chi phí kinh doanh và kế hoạch phát triển của doanh nghiệp. Để hỗ trợ công tác quản lý kho hiệu quả, hiện nay trên thị trường có rất nhiều loại phần mềm quản lý kho , tiêu biểu có thể là Bota, Gtool , và từ những nền tảng đã được học và tìm hiểu nhóm chúng em đã lên ý tưởng thiết kế và phát triển đồ án về đề tài "Quản lý kho điện thoại cao cấp". Với đề tài này, nhóm chúng em mong rằng sẽ góp phần cải thiện được tính hiệu quả, tính nhanh chóng và nâng cao hiệu suất quản lý trong công tác quản lý kho điện thoại của các doanh nghiệp

Và điều mấu chốt là dễ dàng sử dụng- mục đích đề nhắm đến ai cũng có thể sử dụng được mà không phải là chỉ mỗi doanh nghiệp

Class Phone: sẽ có các dòng điện thoại cao cấp , tên sản phẩm , mẫu mã, hệ điều hành, số lượng, đơn giá từ đó sẽ có class con là IOS và Android , Class IOS kế thừa từ class Phone, Class Android kế thừa từ class Phone.

Class Phieu: sẽ có mã phiếu, mã sản phẩm, ngày nhập/xuất, nhà cung cấp và công ty. Class PhieuNhap, PhieuXuat kế thừa từ class Phieu.

# II.Xác định các lớp của bài toán

# Package quanlikho

## 1.1. Class Phone

* Thuộc tính:
* protected String ma;
* protected String ten;
* protected String heDieuHanh;
* protected String mau;
* protected String dungLuong;
* protected double soLuong;
* protected double donGia;
* Phương thức:
* Constructor: Hàm tạo có đối số và không có đối số
* Getter, Setter: phương thức get, set cho các thuộc tính kiểu public
* Nhap: nhập thông tin các thuộc tính
* Hien: hiện ra các thông tin

## 1.2. Class IOS kế thừa từ class Phone

* Phương thức:
* Nhap: override ở class Phone
* Hien: override ở class Phone

## 1.3.Class Android kế thừa từ class Phone

* Phương thức:
* Nhap: override ở class Phone
* Hien: override ở class Phone

## 1.4.Class Phieu

* Thuộc tính:
* public String maPhieu;
* public String ngayNhap;
* public String ma;
* Phương thức:
* Constructor: hàm tạo có đối số và không có đối số
* Nhap: nhập thông tin các thuộc tính
* Hien: hiện thông tin thuộc tính
* Getter, setter: phương thức get, set cho các thuộc tính kiểu public

## 1.5.Class PhieuNhap kế thừa từ class Phieu

* Thuộc tính:
* public String tenNcc;
* Phương thức:
* Nhap: override ở class Phieu
* Hien: override ở class Phieu

## 1.6. Class PhieuXuat kế thừa từ class Phieu

* Thuộc tính:
* public String congTy;
* Phương thức:
* Nhap: override ở class Phieu
* Hien: override ở class Phieu

## 1.7.Class Quanli

* Thuộc tính:
* Phone phone
* PhieuNhap pn
* PhieuXuat px
* Phương thức:
* NhapPhieuNhap()
* NhapPhone()
* insertPN()
* insert()
* Hien()
* selectAll()
* selectAllPN()
* selectbyID()
* selectbyTen()
* delete()
* update()

# Package DAO

## 2.1. Class DAOInterface

* Phương thức:
* public int insert(T t);
* public int delete(T t);
* public void sapxep(T t);
* public ArrayList<T> selectAll();
* public ArrayList<T> selectByCondition(String condition);
* public T selectById(T t);
* public T selectByTen(T t);

## 2.2. Class DAOPhone implements class DAOInterface

* Phương thức:
* insert(T t): thêm điện thoại vào cơ sở dữ liệu
* delete(T t): xóa điện thoại
* sapxep(T t): sắp xếp theo thứ tự tăng dần / giảm dần
* selectAll(): hiển thị danh sách điện thoại
* selectById(T t): tìm kiếm điện thoại theo mã
* selectByTen(T t): tìm kiếm điện thoại theo tên

## 2.3 Class DAOPhieu implements class DAOInterface

* Phương thức:
* insert(T t): thêm phieu nhập, phiếu xuất vào cơ sở dữ liệu
* delete(T t): xóa phiếu nhập, phiếu xuất
* selectAll(): hiển thị danh sách phiếu nhập, phiếu xuất

## 

# III. Mô tả thuật toán

## Chỉnh sửa thông tin điện thoại

* Thêm: nhập một đối tượng mới vào cơ sở dữ liệu

Sử dụng câu lệnh SQL:

* INSERT INTO tên bảng

(cot1, cot2, cot3,...cotN)

VALUES (gia\_tri1, gia\_tri2, gia\_tri3,...gia\_triN);

* SELECT cot1, cot2,... cotN FROM bang1 INNER JOIN bang2 ON bang1.cot\_chung = bang2.cot\_chung;
* Xóa: xóa một đối tượng trong cơ sở dữ liệu

Sử dụng câu lệnh SQL:

* DELETE from tên bảng WHERE điều kiện;

## Hiển thị danh sách điện thoại hiện có trong cơ sở dữ liệu kho

Sử dụng câu lệnh SQL:

* SELECT \* FROM tên bảng

Hiển thị danh sách toàn bộ các đối tượng trong cơ sở dữ liệu

## Tìm kiếm

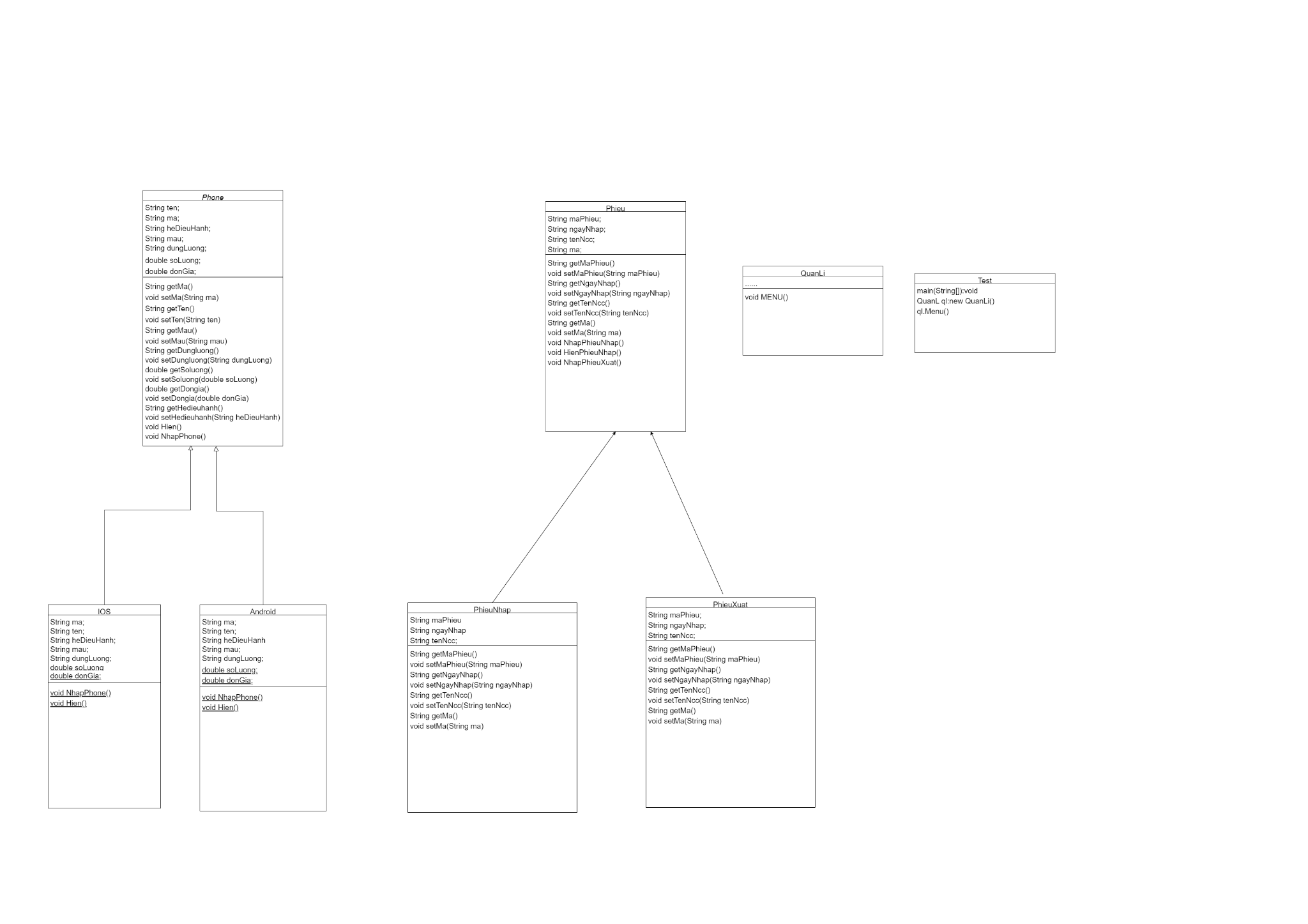
Tìm kiếm theo mã / tên

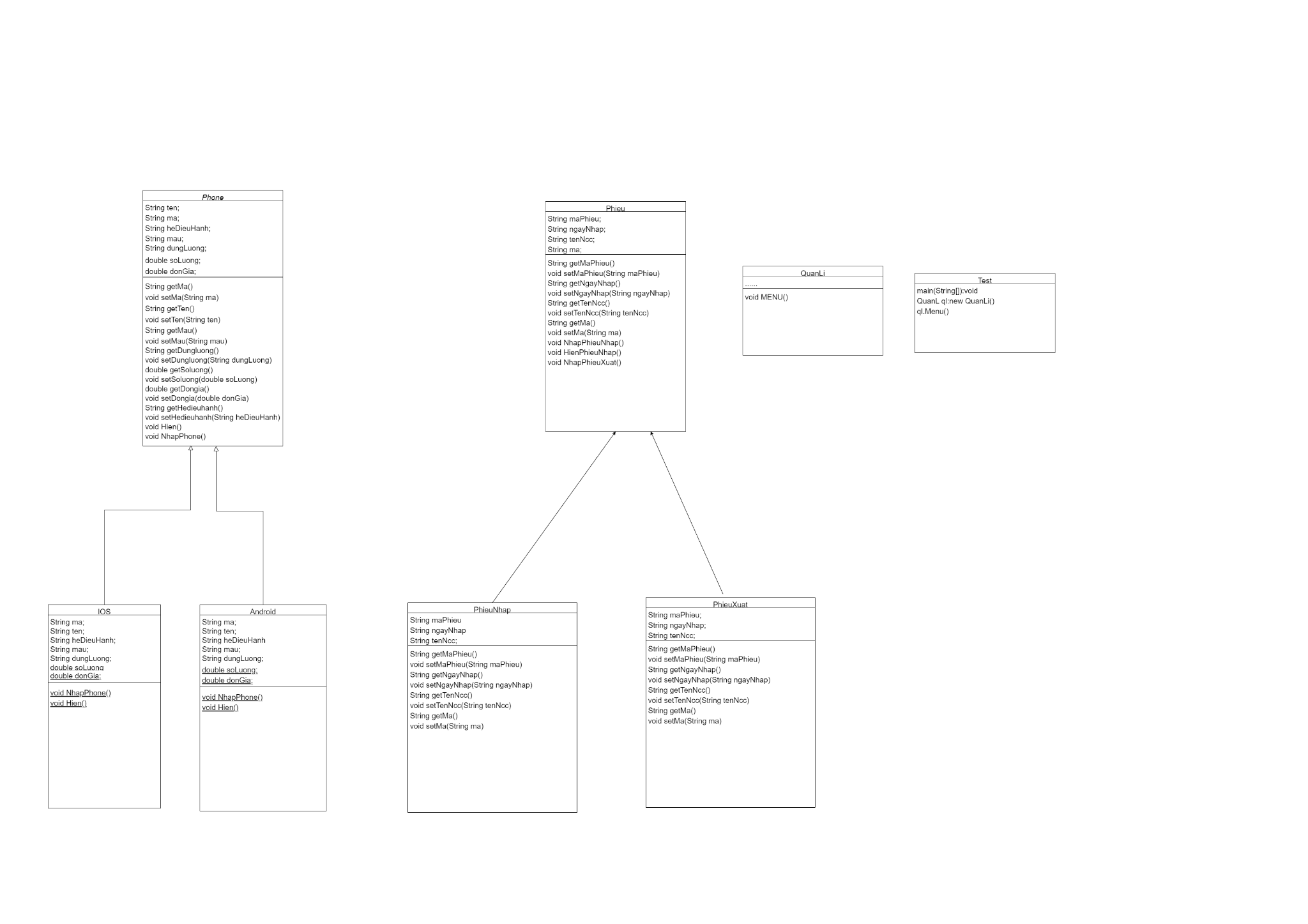
Sử dụng câu lệnh SQL:

* SELECT \* FROM tên bảng WHERE điều kiện

## Sắp xếp

Sắp xếp giá tiền / số lượng tăng dần





# 

# IV. Cài đặt bài toán

# 1.Package quanlikho

## 1.1. Class Phone

package quanlikho;

import java.util.Scanner;

public class Phone {

protected String ma;

protected String ten;

protected String heDieuHanh;

protected String mau;

protected String dungLuong;

protected double soLuong;

protected double donGia;

public Phone() {

}

public Phone(String ma, String ten, String heDieuHanh, String mau, String dungLuong, double soLuong, double donGia) {

this.ma = ma;

this.ten = ten;

this.heDieuHanh = heDieuHanh;

this.mau = mau;

this.dungLuong = dungLuong;

this.soLuong = soLuong;

this.donGia = donGia;

}

public String getMa() {

return ma;

}

public void setMa(String ma) {

this.ma = ma;

}

public String getTen() {

return ten;

}

public void setTen(String ten) {

this.ten = ten;

}

public String getMau() {

return mau;

}

public void setMau(String mau) {

this.mau = mau;

}

public String getDungluong() {

return dungLuong;

}

public void setDungluong(String dungLuong) {

this.dungLuong = dungLuong;

}

public double getSoluong() {

return soLuong;

}

public void setSoluong(double soLuong) {

this.soLuong = soLuong;

}

public double getDongia() {

return donGia;

}

public void setDongia(double donGia) {

this.donGia = donGia;

}

public String getHedieuhanh() {

return heDieuHanh;

}

public void setHedieuhanh(String heDieuHanh) {

this.heDieuHanh = heDieuHanh;

}

public void NhapPhone() {

Scanner input = new Scanner(System.in);

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

ma = input.nextLine();

System.out.println("Nhap ten san pham: ");

ten = input.nextLine();

System.out.println("Nhap he dieu hanh: ");

heDieuHanh=input.nextLine();

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

mau = input.nextLine();

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

dungLuong = input.nextLine();

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

soLuong = input.nextDouble();

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

donGia = input.nextDouble();

}

public void Hien() {

System.out.format("%-15s %-40s %-20s %-15s %-15s %-15s %-15s\n",

"MA SP",

"TEN",

"HE DIEU HANH",

"MAU",

"DUNG LUONG",

"SO LUONG",

"DON GIA"

);

System.out.format("%-15s %-40s %-20s %-15s %-15s %-15s %-15s\n",

ma,

ten,

heDieuHanh,

mau,

dungLuong,

soLuong,

donGia

);

}

}

}

## 1.2. Class IOS

package quanlikho;

public class IOS extends Phone {

public IOS() {

super();

}

public IOS(String ma, String ten, String heDieuHanh, String mau, String dungLuong, double soLuong, double donGia) {

super(ma, ten, heDieuHanh, mau, dungLuong, soLuong, donGia);

}

@Override

public void NhapPhone() {

super.NhapPhone();

}

@Override

public void Hien() {

super.Hien();

}

}

## 1.3. Class Android

package quanlikho;

public class Android extends Phone {

public Android() {

}

public Android(String ma, String ten, String heDieuHanh, String mau, String dungLuong, double soLuong, double donGia) {

super(ma, ten, heDieuHanh, mau, dungLuong, soLuong, donGia);

}

@Override

public void NhapPhone() {

super.NhapPhone();

}

@Override

public void Hien() {

super.Hien(); }

}

## 

## 1.4. Class Phieu

package quanlikho;

import java.util.Scanner;

public class Phieu

{

public String maPhieu;

public String ngayNhap;

public String tenNcc;

public String ma;

public Phieu(String maPhieu, String ngayNhap, String tenNcc) {

this.maPhieu = maPhieu;

this.ngayNhap = ngayNhap;

this.tenNcc = tenNcc;

this.ma = ma;

}

public Phieu() {

}

public String getMaPhieu() {

return maPhieu;

}

public void setMaPhieu(String maPhieu) {

this.maPhieu = maPhieu;

}

public String getNgayNhap() {

return ngayNhap;

}

public void setNgayNhap(String ngayNhap) {

this.ngayNhap = ngayNhap;

}

public String getTenNcc() {

return tenNcc;

}

public void setTenNcc(String tenNcc) {

this.tenNcc = tenNcc;

}

public String getMa() {

return ma;

}

public void setMa(String ma) {

this.ma = ma;

}

public void NhapPhieuNhap()

{

Scanner input = new Scanner(System.in);

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

maPhieu = input.nextLine();

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

ma = input.nextLine();

System.out.println("Nhap ten nha cung cap: ");

tenNcc = input.nextLine();

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

ngayNhap = input.nextLine();

}

public void HienPhieuNhap()

{

System.out.format("%-20s %-20s %-40s %-20s\n",

"MA PHIEU",

"MA",

"TENNCC",

"NGAY NHAP"

);

System.out.format("%-20s %-20s %-40s %-20s \n",

maPhieu,

ma,

tenNcc,

ngayNhap

);

}

public void NhapPhieuXuat()

{

Scanner input = new Scanner(System.in);

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

maPhieu = input.nextLine();

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

ma = input.nextLine();

System.out.println("Nhap ten cong ty: ");

tenNcc = input.nextLine();

System.out.println("Nhap ngay xuat: ");

ngayNhap = input.nextLine();

}

}

## 1.5. Class PhieuNhap

package quanlikho;

import java.util.Scanner;

public class PhieuNhap extends Phieu

{

public PhieuNhap(String maPhieu, String ngayNhap, String tenNcc) {

super(maPhieu, ngayNhap, tenNcc);

}

public PhieuNhap() {

}

public String getMaPhieu() {

return maPhieu;

}

public void setMaPhieu(String maPhieu) {

this.maPhieu = maPhieu;

}

public String getNgayNhap() {

return ngayNhap;

}

public void setNgayNhap(String ngayNhap) {

this.ngayNhap = ngayNhap;

}

public String getTenNcc() {

return tenNcc;

}

public void setTenNcc(String tenNcc) {

this.tenNcc = tenNcc;

}

public String getMa() {

return ma;

}

public void setMa(String ma) {

this.ma = ma;

}

}

## 1.6.Class PhieuXuat

package quanlikho;

public class PhieuXuat extends Phieu

{

public PhieuXuat(String maPhieu, String ngayNhap, String tenNcc) {

super(maPhieu, ngayNhap, tenNcc);

}

public PhieuXuat() {

}

public String getMaPhieu() {

return maPhieu;

}

public void setMaPhieu(String maPhieu) {

this.maPhieu = maPhieu;

}

public String getNgayNhap() {

return ngayNhap;

}

public void setNgayNhap(String ngayNhap) {

this.ngayNhap = ngayNhap;

}

public String getTenNcc() {

return tenNcc;

}

public void setTenNcc(String tenNcc) {

this.tenNcc = tenNcc;

}

public String getMa() {

return ma;

}

public void setMa(String ma) {

this.ma = ma;

}

}

## 1.7. Class Quanli

package quanlikho;

import DAO.DAOPhone;

import java.util.ArrayList;

import java.util.Scanner;

import DAO.DAOPhieu;

/\*\*

\*

\* @author thanh ha

\*/

public class QuanLi

{

public void MENU()

{

Scanner input = new Scanner(System.in);

Phone phone;

Phieu phieuNhap;

Phieu phieuXuat;

int chon;

do {

System.out.println("+---------------------MENU QUAN LY KHO DIEN THOAI CAO CAP---------------------------+");

System.out.println("1. Phieu Nhap");

System.out.println("2. Phieu Xuat");

System.out.println("3. Hien thi danh sach trong kho");

System.out.println("4. Tim kiem");

System.out.println("5. Xoa");

System.out.println("6. Sap xep");

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

System.out.print("\n Chon chuc nang: ");

chon = input.nextInt();

if (chon == 1) {

System.out.println("\n+---------PHIEU NHAP DIEN THOAI---------+");

DAOPhieu dpn = new DAOPhieu();

DAOPhone dp = new DAOPhone();

phieuNhap = new Phieu();

phone = new Phone();

phieuNhap.NhapPhieuNhap();

phone.NhapPhone();

dpn.insertPN(phieuNhap);

dp.insert(phone);

}

else if (chon==2)

{

System.out.println("\n+---------PHIEU XUAT DIEN THOAI---------+");

//Xuất hàng thì phải xóa hàng

DAOPhieu dpx = new DAOPhieu();

DAOPhone dp = new DAOPhone();

phieuXuat = new Phieu();

phone = new Phone();

phieuXuat.NhapPhieuXuat();

phone.NhapPhone();

dpx.insert(phieuXuat);

dp.insertPhoneXuat(phone);

}

else if (chon == 3) {

System.out.println("+---------HIEN THI DANH SACH TRONG KHO---------+");

System.out.println("1.Hien thi danh sach dien thoai nhap");

System.out.println("2.Hien thi danh sach dien thoai xuat");

System.out.println("3.Hien thi danh sach phieu nhap");

System.out.println("4.Hien thi thong tin phieu xuat");

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

chon = input.nextInt();

if(chon==1)

{

ArrayList<Phone> list = DAOPhone.getInstance().selectAll();

for (Phone p : list)

{

p.Hien();

}

}

else if(chon==2)

{

ArrayList<Phone> list = DAOPhone.getInstance().selectAllPhoneXuat();

for (Phone p : list)

{

p.Hien();

}

}

else if (chon==3)

{

DAOPhieu dpn = new DAOPhieu();

dpn.selectAllPN();

}

else if(chon==4)

{

DAOPhieu dpx = new DAOPhieu();

dpx.selectAll();

}

}

else if (chon == 4) {

System.out.println("+---------TIM KIEM DIEN THOAI TRONG KHO---------+");

System.out.println("1.Tim kiem theo ma");

System.out.println("2.Tim kiem theo ten");

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

chon = input.nextInt();

if (chon==1)

{

input.nextLine();

Phone find = new Phone();

System.out.println("Nhap ma can tim: ");

String ma = input.nextLine();

find.setMa(ma);

Phone ip2 = DAOPhone.getInstance().selectById(find);

ip2.Hien();

}

else if (chon==2)

{

input.nextLine();

Phone find = new Phone();

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

String ten = input.nextLine();

find.setTen(ten);

Phone ip2 = DAOPhone.getInstance().selectByTen(find);

ip2.Hien();

}

}

else if (chon==5)

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

phone = new Phone();

DAOPhone dp = new DAOPhone();

dp.delete(phone);

}

else if (chon==6)

{ }

} while (chon!= 0);

}

}

## 1.7. Class Main

package quanlikho;

public class Main {

public static void main(String[] args)

{

QuanLi ql = new QuanLi();

ql.MENU();

}

}

# 2. Package DAO

## 2.1. Class DAOInterface

package DAO;

import java.util.ArrayList;

public interface DAOInterface <T>

{

public int insert(T t);

public int update(T t);

public int delete(T t);

public ArrayList<T> selectAll();

public ArrayList<T> selectByCondition(String condition);

public T selectById(T t);

public T selectByTen(T t);

public T selectByHedieuhanh(T t);

}

## 2.2. Class DAOPhone

package DAO;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Scanner;

import quanlikho.Phone;

import Database.ConnecttoSQL;

public class DAOPhone implements DAOInterface <Phone> {

public static DAOPhone getInstance()

{

return new DAOPhone();

}

@Override

public int insert(Phone t) {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="INSERT INTO DIENTHOAINHAP (MA, TEN, HEDIEUHANH, MAU, DUNGLUONG, SOLUONG, DONGIA)"

+ "VALUES ('"+t.getMa()+

"' , '"+t.getTen()+

"' , '"+ t.getHedieuhanh()+

"' , '"+t.getMau()+

"' , '"+t.getDungluong()+

"' , '"+t.getSoluong()+

"' , '"+t.getDongia()+ "')";

rows = stm.executeUpdate(sqlInsert);

System.out.println("INSERT DIEN THOAI THANH CONG!");

while (rs.next())

{

System.out.println(rs.getString(1) +" " +rs.getString(2) +" " + rs.getString(3) +" " + rs.getString(4) +" "+ rs.getString(5) +" "+rs.getString(6)+" "+rs.getString(7) );

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

@Override

public int update(Phone t) {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP";

ResultSet rs = stm.executeQuery(selectSql);

String sqlUpdate ="UPDATE DIENTHOAINHAP "

+ "SET TEN ='" + t.getTen() + "'"

+ ", HEDIEUHANH ='" + t.getHedieuhanh()

+ ", MAU ='" + t.getMau()

+ ", DUNGLUONG ='" + t.getDungluong()

+ ", SOLUONG =" + t.getSoluong()

+ ", DONGIA =" + t.getDongia()

+ " WHERE MA ='" + t.getMa() + "\'";

rows = stm.executeUpdate(sqlUpdate);

System.out.println("UPDATE THANH CONG!");

while (rs.next())

{

System.out.println(rs.getString(1) +" " +rs.getString(2) +" " + rs.getString(3) +" " + rs.getString(4) +" "+ rs.getString(5) +" "+rs.getString(6)+" "+rs.getString(7) );

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

@Override

public int delete(Phone t) {

Scanner input = new Scanner (System.in);

System.out.println("Nhap ma dien thoai can xoa: ");

String ma = input.nextLine();

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="DELETE from DIENTHOAINHAP "+

" WHERE MA='"+ma+"'";

rows = stm.executeUpdate(sqlInsert);

System.out.println("DELETE DIEN THOAI THANH CONG!");

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

e.printStackTrace();

}

return rows;

}

@Override

public ArrayList<Phone> selectAll() {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

ArrayList <Phone> rows = new ArrayList<Phone>();

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP";

ResultSet rs = stm.executeQuery(selectSql);

while (rs.next())

{

String ma = rs.getString(1);

String ten = rs.getString(2);

String heDieuHanh = rs.getString(3);

String mau = rs.getString(4);

String dungLuong = rs.getString(5);

double soLuong = rs.getDouble(6);

double donGia = rs.getDouble(7);

Phone p = new Phone(ma, ten, heDieuHanh, mau,dungLuong,soLuong,donGia);

rows.add(p);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

@Override

public Phone selectById(Phone t)

{

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

Phone rows = null;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP WHERE MA='"+t.getMa()+"'";

ResultSet rs = stm.executeQuery(selectSql);

while (rs.next())

{

String ma = rs.getString(1);

String ten = rs.getString(2);

String heDieuHanh = rs.getString(3);

String mau = rs.getString(4);

String dungLuong = rs.getString(5);

double soLuong = rs.getDouble(6);

double donGia = rs.getDouble(7);

rows = new Phone(ma, ten, heDieuHanh, mau,dungLuong,soLuong,donGia);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");;

}

return rows;

}

@Override

public Phone selectByTen(Phone t) {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

Phone rows = null;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAINHAP WHERE TEN='"+t.getTen()+"'";

ResultSet rs = stm.executeQuery(selectSql);

while (rs.next())

{

String ma = rs.getString(1);

String ten = rs.getString(2);

String heDieuHanh = rs.getString(3);

String mau = rs.getString(4);

String dungLuong = rs.getString(5);

double soLuong = rs.getDouble(6);

double donGia = rs.getDouble(7);

rows = new Phone(ma, ten, heDieuHanh, mau,dungLuong,soLuong,donGia);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");;

}

return rows;

}

public int insertPhoneXuat(Phone t)

{

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAIXUAT";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="INSERT INTO DIENTHOAIXUAT (MA, TEN, HEDIEUHANH, MAU, DUNGLUONG, SOLUONG, DONGIA)"

+ "VALUES ('"+t.getMa()+

"' , '"+t.getTen()+

"' , '"+ t.getHedieuhanh()+

"' , '"+t.getMau()+

"' , '"+t.getDungluong()+

"' , '"+t.getSoluong()+

"' , '"+t.getDongia()+ "')";

rows = stm.executeUpdate(sqlInsert);

System.out.println("INSERT DIEN THOAI THANH CONG!");

while (rs.next())

{

System.out.println(rs.getString(1) +" " +rs.getString(2) +" " + rs.getString(3) +" " + rs.getString(4) +" "+ rs.getString(5) +" "+rs.getString(6)+" "+rs.getString(7) );

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");;

}

return rows;

}

public ArrayList<Phone> selectAllPhoneXuat() {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

ArrayList <Phone> rows = new ArrayList<Phone>();

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM DIENTHOAIXUAT";

ResultSet rs = stm.executeQuery(selectSql);

while (rs.next())

{

String ma = rs.getString(1);

String ten = rs.getString(2);

String heDieuHanh = rs.getString(3);

String mau = rs.getString(4);

String dungLuong = rs.getString(5);

double soLuong = rs.getDouble(6);

double donGia = rs.getDouble(7);

Phone p = new Phone(ma, ten, heDieuHanh, mau,dungLuong,soLuong,donGia);

rows.add(p);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

}

## 2.3. Class DAOPhieu

package DAO;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Date;

import java.util.Scanner;

import quanlikho.Phieu;

import quanlikho.Phone;

public class DAOPhieu implements DAOInterface <Phieu> {

public static DAOPhieu getInstance()

{

return new DAOPhieu();

}

public int insertPN(Phieu t) {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM PHIEUNHAP";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="INSERT INTO PHIEUNHAP (MAPHIEU, MA, TENNCC, NGAYNHAP)"

+ "VALUES ('"+t.getMaPhieu()+

"' , '"+t.getMa()+

"' , '"+ t.getTenNcc()+

"' , '"+t.getNgayNhap()+

"')";

rows = stm.executeUpdate(sqlInsert);

System.out.println("INSERT PHIEU NHAP THANH CONG!");

while (rs.next())

{

System.out.println(rs.getString(1) +

" " +rs.getString(2) +" "

+ rs.getString(3) +" "

+ rs.getString(4) );

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

@Override

public int update(Phieu t) {

throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody

}

@Override

public int delete(Phieu t) {

Scanner input = new Scanner (System.in);

System.out.println("Nhap ma phieu nhap can xoa: ");

String maPhieu = input.nextLine();

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM PHIEUNHAP";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="DELETE from PHIEUNHAP "+

" WHERE MAPHIEU='"+maPhieu+"'";

rows = stm.executeUpdate(sqlInsert);

System.out.println("DELETE PHIEU NHAP THANH CONG!");

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

e.printStackTrace();

}

return rows;

}

public ArrayList<Phieu> selectAllPN() {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

ArrayList <Phieu> rows = new ArrayList<Phieu>();

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String sql = "SELECT PHIEUNHAP.MAPHIEU, "

+ "PHIEUNHAP.TENNCC, "

+ "PHIEUNHAP.NGAYNHAP, "

+ "DIENTHOAINHAP.TEN, "

+ "DIENTHOAINHAP.HEDIEUHANH, "

+ "DIENTHOAINHAP.MAU,"

+ "DIENTHOAINHAP.DUNGLUONG,"

+ "DIENTHOAINHAP.SOLUONG, "

+ "DIENTHOAINHAP.DONGIA \n"

+"FROM PHIEUNHAP INNER JOIN DIENTHOAINHAP ON PHIEUNHAP.MA = DIENTHOAINHAP.MA";

ResultSet rs3 = stm.executeQuery(sql);

while (rs3.next()) {

String maPhieu = rs3.getString(1);

String tenNcc = rs3.getString(2);

String ngayNhap = rs3.getString(3);

String ten = rs3.getString(4);

String heDieuHanh = rs3.getString(5);

String mau = rs3.getString(6);

String dungLuong = rs3.getString(7);

double soLuong = rs3.getDouble(8);

double donGia = rs3.getDouble(9);

Phieu ph = new Phieu(maPhieu,tenNcc,ngayNhap);

rows.add(ph);

System.out.format("%-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s\n",

"MAPHIEU",

"TENNCC",

"NGAY NHAP",

"TEN",

"HEDIEUHANH",

"MAU",

"DUNGLUONG",

"SOLUONG",

"DONGIA"

);

System.out.format("%-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s\n",

maPhieu,

tenNcc,

ngayNhap,

ten,

heDieuHanh,

mau,

dungLuong,

soLuong,

donGia

);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

e.printStackTrace();

}

return rows;

}

@Override

public Phieu selectById(Phieu t) {

throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody

}

@Override

public Phieu selectByTen(Phieu t) {

throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody

}

@Override

public int insert(Phieu t) {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM PHIEUXUAT";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="INSERT INTO PHIEUXUAT (MAPHIEU, MA, CONGTY, NGAYXUAT)"

+ "VALUES ('"+t.getMaPhieu()+

"' , '"+t.getMa()+

"' , '"+ t.getTenNcc()+

"' , '"+t.getNgayNhap()+

"')";

rows = stm.executeUpdate(sqlInsert);

System.out.println("INSERT PHIEU XUAT THANH CONG!");

while (rs.next())

{

System.out.println(rs.getString(1) +

" " +rs.getString(2) +" "

+ rs.getString(3) +" "

+ rs.getString(4) );

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

System.out.println(" ");

}

return rows;

}

@Override

public ArrayList<Phieu> selectAll() {

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

ArrayList <Phieu> rows = new ArrayList<Phieu>();

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String sql = "SELECT PHIEUXUAT.MAPHIEU, "

+ "PHIEUXUAT.CONGTY, "

+ "PHIEUXUAT.NGAYXUAT, "

+ "DIENTHOAIXUAT.TEN, "

+ "DIENTHOAIXUAT.HEDIEUHANH, "

+ "DIENTHOAIXUAT.MAU,"

+ "DIENTHOAIXUAT.DUNGLUONG,"

+ "DIENTHOAIXUAT.SOLUONG, "

+ "DIENTHOAIXUAT.DONGIA \n"

+"FROM PHIEUXUAT INNER JOIN DIENTHOAIXUAT ON PHIEUXUAT.MA = DIENTHOAIXUAT.MA";

ResultSet rs3 = stm.executeQuery(sql);

while (rs3.next()) {

String maPhieu = rs3.getString(1);

String tenNcc = rs3.getString(2);

String ngayNhap = rs3.getString(3);

String ten = rs3.getString(4);

String heDieuHanh = rs3.getString(5);

String mau = rs3.getString(6);

String dungLuong = rs3.getString(7);

double soLuong = rs3.getDouble(8);

double donGia = rs3.getDouble(9);

Phieu ph = new Phieu(maPhieu,ngayNhap,tenNcc);

rows.add(ph);

System.out.format("%-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s\n",

"MAPHIEU",

"CONGTY",

"NGAY NHAP",

"TEN",

"HEDIEUHANH",

"MAU",

"DUNGLUONG",

"SOLUONG",

"DONGIA"

);

System.out.format("%-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s %-20s\n",

maPhieu,

tenNcc,

ngayNhap,

ten,

heDieuHanh,

mau,

dungLuong,

soLuong,

donGia

);

}

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

e.printStackTrace();

}

return rows;

}

public int deletePX(Phieu t) {

Scanner input = new Scanner (System.in);

System.out.println("Nhap ma phieu nhap can xoa: ");

String maPhieu = input.nextLine();

String connectionUrl =

"jdbc:sqlserver://localhost:1433;"

+ "database=KHODIENTHOAI;"

+ "user=sa;"

+ "password=123;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

int rows=0;

try (Connection connection = DriverManager.getConnection(connectionUrl);

Statement stm = connection.createStatement();) {

// Code here.

String selectSql = "SELECT\* FROM PHIEUXUAT";

ResultSet rs = stm.executeQuery(selectSql);

String sqlInsert ="DELETE from PHIEUXUAT "+

" WHERE MAPHIEU='"+maPhieu+"'";

rows = stm.executeUpdate(sqlInsert);

System.out.println("DELETE PHIEU XUAT THANH CONG!");

connection.close();

}

// Handle any errors that may have occurred.

catch (SQLException e) {

e.printStackTrace();

}

return rows;

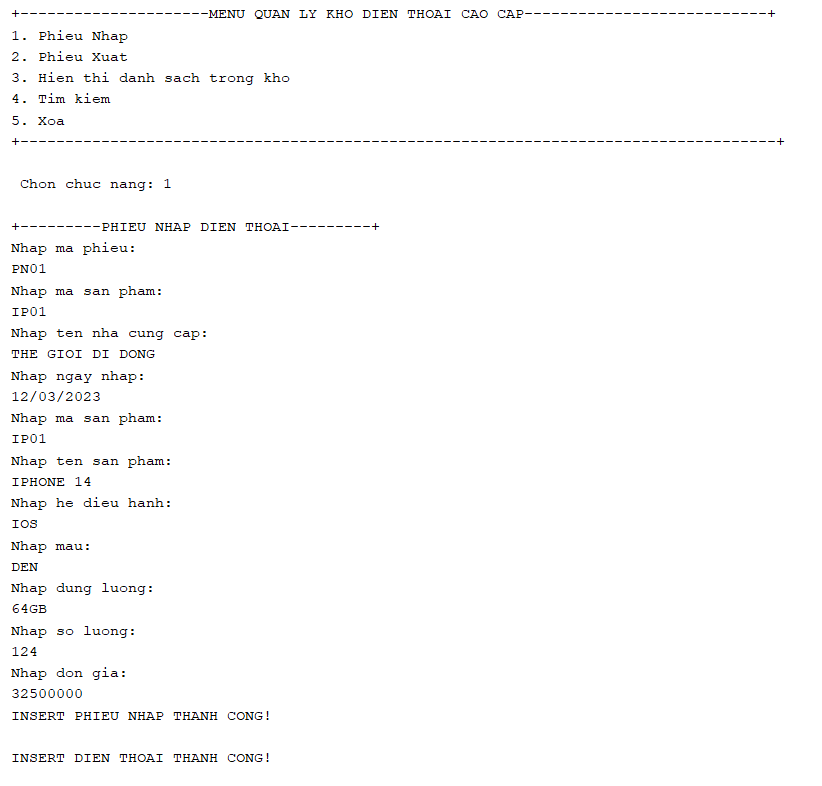
}

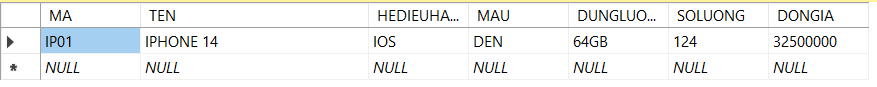
}

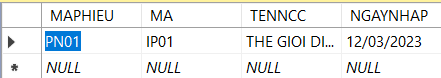
# 

# V. Kiểm thử

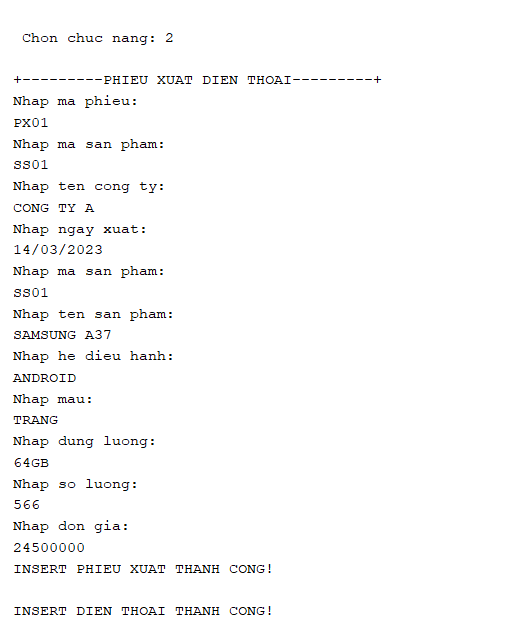
# Thêm điện thoại, phiếu nhập:

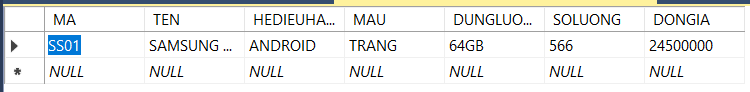






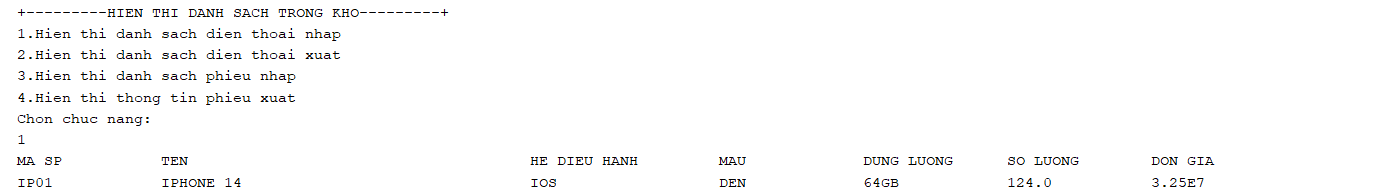
# Thêm điện thoại, phiếu xuất

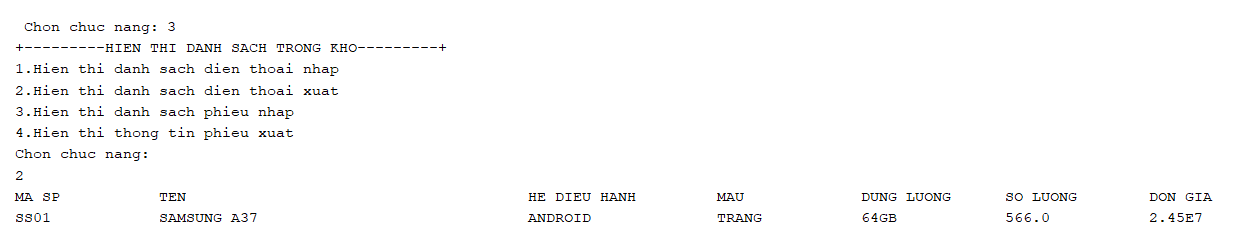


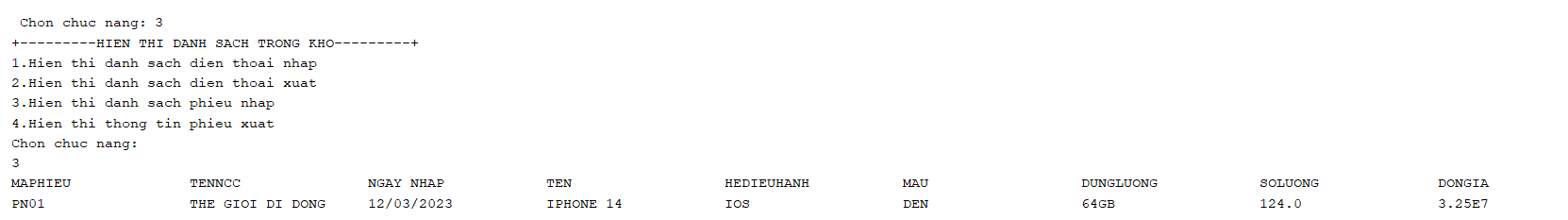


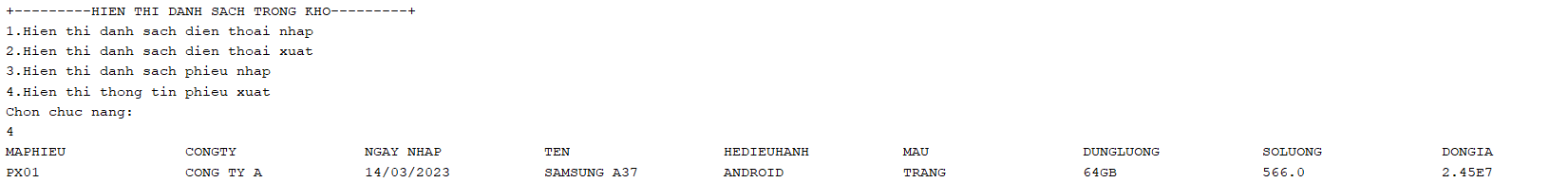
# 

# Hiển thị danh sách điện thoại nhập, xuất



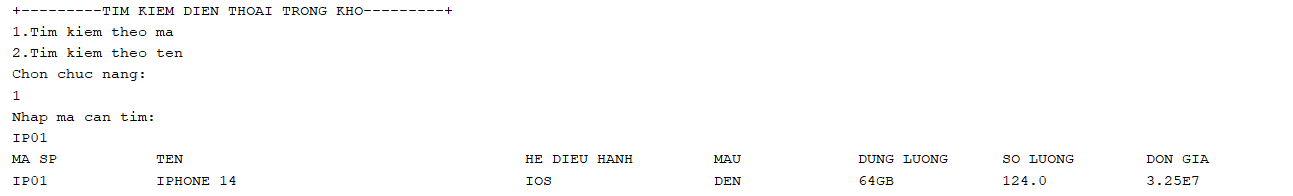




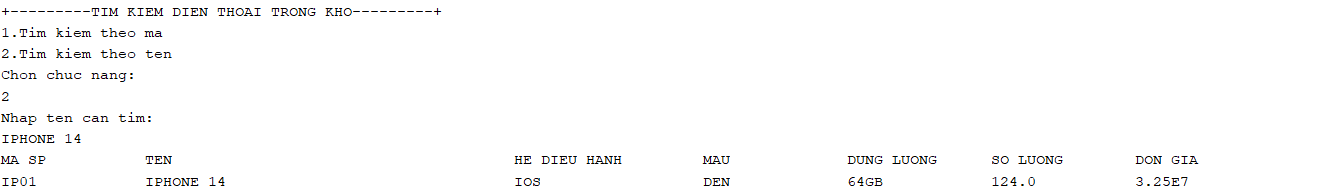


1. Tìm kiếm điện thoại:

* Theo mã điện thoại:

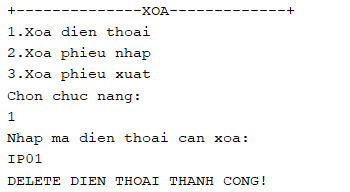


* Theo tên:



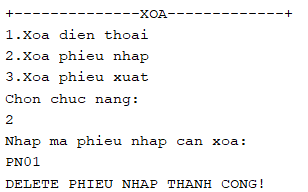
1. Xóa

* Xóa điện thoại theo mã:



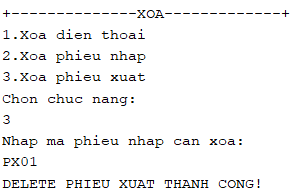


* Xóa phiếu nhập theo mã:





* Xóa phiếu xuất theo mã





# VI. Tài liệu tham khảo

1. <https://learn.microsoft.com/en-us/sql/connect/jdbc/microsoft-jdbc-driver-for-sql-server?view=sql-server-ver16>
2. <https://titv.vn/courses-page/lap-trinh-java-tuong-tac-csdl/>