TRƯỜNG ĐẠI HỌC THỦY LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN**

MÔN HỌC: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

**ĐỀ TÀI: QUẢN LÝ SIÊU THỊ ĐIỆN MÁY**

Giáo viên hướng dẫn: Nguyễn Ngọc Quỳnh Châu

Nhóm sinh viên thực hiện: Nhóm 6

1. Nguyễn Minh Tuấn, lớp 62HT, MSV:2051063441

2. Mai Xuân Hiếu, lớp 62HT, MSV:2051063674

3. Bùi Quang Huy, lớp 62HT, MSV:2051062383

**Hà Nội, năm 2022**

MỤC LỤC

[PHẦN 1: TỔNG QUAN 3](#_Toc124200016)

[1.1. Phân công công việc 3](#_Toc124200017)

[1.2. Mô tả bài toán 3](#_Toc124200018)

[1.3. Quy tắc nghiệp vụ 4](#_Toc124200019)

[1.4. Xác định thực thể 4](#_Toc124200020)

[1.4.1. Xác định thực thể và các thuộc tính 4](#_Toc124200021)

[1.4.2. Biểu diễn mô hình ER 5](#_Toc124200022)

[1.5. Chuyển từ mô hình ER sang các bảng 5](#_Toc124200023)

[PHẦN 2: LẬP TRÌNH 7](#_Toc124200024)

[2.1. Tạo CSDL: 7](#_Toc124200025)

[2.1.1. Tạo CSDL: 7](#_Toc124200026)

[2.1.2. Tạo bảng 8](#_Toc124200027)

[2.2. Các câu lệnh 14](#_Toc124200028)

[2.2.1. Cursor 14](#_Toc124200029)

[2.2.2. Stored Procedure 15](#_Toc124200030)

[2.2.3. Fuction 18](#_Toc124200031)

[2.2.4. View 19](#_Toc124200032)

[2.2.5. Trigger 20](#_Toc124200033)

[2.2.6. Transaction 23](#_Toc124200034)

[2.3. Giao diện 25](#_Toc124200035)

[2.3.1. Đăng nhập 25](#_Toc124200036)

[2.3.2. Giao diện chính 25](#_Toc124200037)

[PHẦN 3: TỔNG KẾT 31](#_Toc124200038)

# PHẦN 1: TỔNG QUAN

## 1.1. Phân công công việc

|  |  |  |
| --- | --- | --- |
| **Tên thành viên** | **Tên công việc** | **Ghi chú** |
| Nguyễn Minh Tuấn | - Lập trình giao diện  - Xây dựng CSDL  - Lập trình SQL Server |  |
| Mai Xuân Hiếu | - Thêm dữ liệu  - Xây dựng CSDL  - Lập trình SQL Server |  |
| Bùi Quang Huy | - Làm báo cáo  - Xây dựng CSDL  - Lập trình SQL Server |  |

## 1.2. Mô tả bài toán

Siêu thị điện máy bán một số mặt hàng thiết bị điện tử, các thông tin về các mặt hàng bao gồm: Mã mặt hàng, Tên mặt hàng, Đơn vị tính, Số lượng, Giá tiền. Các mặt hàng sẽ thuộc một loại hàng nào đó, siêu thị sẽ lưu lại trên hệ thống với thông tin về Mã loại hàng và Tên loại hàng.

Nhân viên sẽ được quản lí qua các thông tin: Mã nhân viên, Tên nhân viên, Giới tính, Địa chỉ, Điện thoại, Ngày sinh, Ngày vào làm, Lương cơ bản, Hệ số lương, Phụ cấp, Tình trạng (còn đi làm hay đã nghỉ việc). Các nhân viên sẽ làm 2 nhiệm vụ chính: Quản lí nhập hàng và Bán hàng:

+ Nhập hàng: Khi nhập hàng, nhân viên sẽ lập phiếu nhập bao gồm các thông tin: Mã phiếu nhập, Ngày nhập. Trên một phiếu nhập hàng có thể có nhiều mặt hàng, thông tin của từng mặt hàng sẽ được lưu vào hệ thống bao gồm: Giá nhập, Số lượng. Bên cạnh đó, thông tin về nhà cung cấp cũng được lưu lại với các thông tin: Mã nhà cung cấp, Tên nhà cung cấp, Địa chỉ, Điện thoại, Tình trạng (còn cung cấp không hay đã dừng).

+ Bán hàng: Nhân viên sẽ tiến hành lập hóa đơn cho khách hàng mua hàng. Khách hàng sẽ được lưu trữ các thông tin cơ bản bao gồm: Mã khách hàng, Tên khách hàng, Giới tính, Địa chỉ, Điện thoại, Số lần mua. Thông tin cơ bản trên hóa đơn gồm: Mã hóa đơn, Ngày bán. Trên hóa đơn của một khách hàng có thể mua nhiều mặt hàng, thông tin của từng mặt hàng được lưu vào hệ thống bao gồm: Giá Bán, Số lượng, Mức giảm giá.

Sau khi tiến hành giao dịch mua bán thì cuối tháng siêu thị sẽ tổng kết lại tất cả các phiếu nhập và hóa đơn, xem tổng tiền đã nhập hàng và tổng tiền thu được trong quá trình bán hàng trong tháng, từ đó tổng kết doanh thu trong tháng, sau đó là doanh thu trong năm.

## 1.3. Quy tắc nghiệp vụ

- Một loại hàng sẽ có thể chứa nhiều mặt hàng.

- Một nhân viên có thể lập nhiều phiếu nhập.

- Một phiếu nhập có thể chứa nhiều mặt hàng.

- Một mặt hàng có thể xuất hiện trong nhiều phiếu nhập.

- Một nhà cung cấp có thể cung cấp nhiều mặt hàng.

- Một nhân viên có thể lập nhiều hóa đơn.

- Một khách hàng có thể có nhiều hóa đơn.

- Một hóa đơn có thể chứa nhiều mặt hàng.

- Một mặt hàng cũng có thể xuất hiện trong nhiều hóa đơn.

## 1.4. Xác định thực thể

### 1.4.1. Xác định thực thể và các thuộc tính

- MatHang (***MaMH***, TenMH, DonViTinh, SoLuong)

- LoaiHang (***MaLH***, TenLH)

- NhanVien ***(MaNV***, TenNV, GioiTinh, DiaChi, DienThoai, NgaySinh, NgayVaoLam, LuongCoBan, HeSoLuong, PhuCap, TinhTrang)

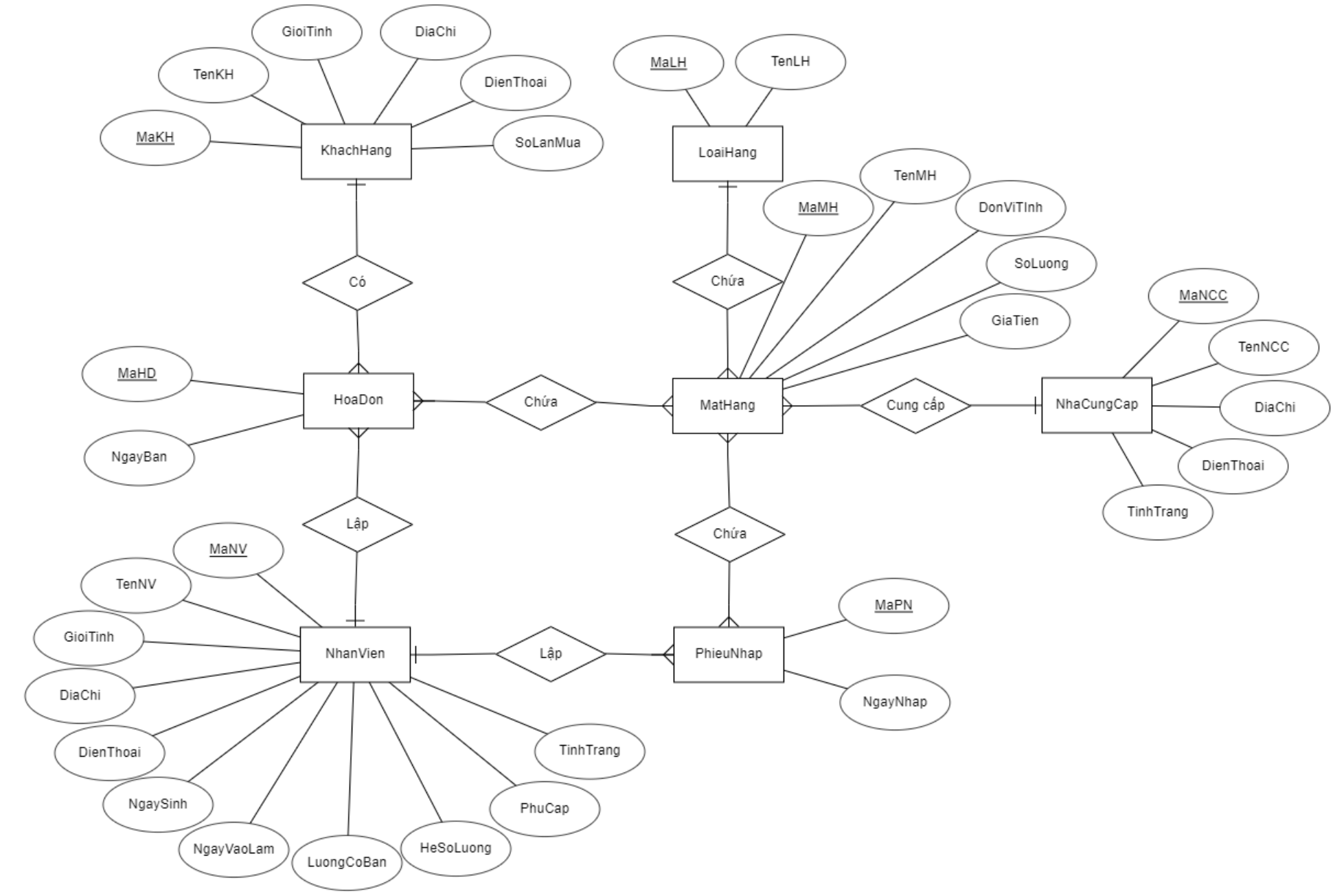
- NhaCungCap (***MaNCC***, TenNCC, DiaChi, DienThoai, TinhTrang)

- KhachHang (***MaKH***, TenKH, GioiTinh, DiaChi, DienThoai, SoLanMua)

- HoaDon (***MaHD***, NgayBan)

- PhieuNhap (***MaPN***, NgayNhap)

### 1.4.2. Biểu diễn mô hình ER



## 1.5. Chuyển từ mô hình ER sang các bảng

- Mỗi thực thể trở thành 1 bảng: Ta có 7 bảng:

+ MatHang (***MaMH***, TenMH, DonViTinh, SoLuong, GiaTien)

+ LoaiHang (***MaLH***, TenLH)

+ NhanVien ***(MaNV***, TenNV, GioiTinh, DiaChi, DienThoai, NgaySinh, NgayVaoLam, LuongCoBan, HeSoLuong, PhuCap, TinhTrang)

+ NhaCungCap (***MaNCC***, TenNCC, DiaChi, DienThoai, TinhTrang)

+ KhachHang (***MaKH***, TenKH, GioiTinh, DiaChi, DienThoai, SoLanMua)

+ HoaDon (***MaHD***, NgayBan)

+ PhieuNhap (***MaPN***, NgayNhap)

- Liên kết 1 – N: Khóa của bảng bên 1 trở thành khóa ngoại của bảng bên nhiều

+ KhachHang – HoaDon: 1 – N

=> HoaDon (***MaHD***, *MaKH*, NgayBan)

+ NhanVien – HoaDon: 1 – N

=> HoaDon (***MaHD***, *MaKH*, *MaNV*, NgayBan)

+ NhanVien – PhieuNhap: 1 – N

=> PhieuNhap (***MaPN***, *MaNV*, NgayNhap)

+ NhaCungCap – MatHang: 1 – N

=> MatHang (***MaMH***, TenMH, *MaNCC*, DonViTinh, SoLuong, GiaTien)

+ LoaiHang – MatHang: 1 – N

=> MatHang (***MaMH***, TenMH, *MaNCC*, *MaLH*, DonViTinh, SoLuong, GiaTien)

- Liên kết M – N: Tạo thành 1 bảng mới, chuyển khóa chính của 2 bên M và N thành khóa ngoại của bảng mới

+ HoaDon – MatHang: M – N

=> ChiTietHoaDon (*MaHD*, *MaMH*, GiaBan, SoLuong, MucGiamGia)

+ PhieuNhap – MatHang: M – N

=> ChiTietPhieuNhap (*MaPN*, *MaMH*, GiaNhap, SoLuong)

🡪 Ta được 9 bảng hoàn chỉnh như sau:

+ MatHang (***MaMH***, TenMH, *MaNCC*, *MaLH*, DonViTinh, SoLuong, GiaTien)

+ LoaiHang (***MaLH***, TenLH)

+ NhanVien ***(MaNV***, TenNV, GioiTinh, DiaChi, DienThoai, NgaySinh, NgayVaoLam, LuongCoBan, HeSoLuong, PhuCap, TinhTrang)

+ NhaCungCap (***MaNCC***, TenNCC, DiaChi, DienThoai, TinhTrang)

+ KhachHang (***MaKH***, TenKH, GioiTinh, DiaChi, DienThoai, SoLanMua)

+ HoaDon (***MaHD***, *MaKH*, *MaNV*, NgayBan)

+ ChiTietHoaDon (*MaHD*, *MaMH*, GiaBan, SoLuong, MucGiamGia)

+ PhieuNhap (***MaPN***, *MaNV*, NgayNhap)

+ ChiTietPhieuNhap (*MaPN*, *MaMH*, GiaNhap, SoLuong)

# PHẦN 2: LẬP TRÌNH

## 2.1. Tạo CSDL:

### 2.1.1. Tạo CSDL:

- Cơ sở dữ liệu có tên là QuanLySieuThiDienMay gồm 2 file:

+ File chính:

* Tên: HTQLBH\_data
* Đường dẫn: E:\BE\_STUDYING\He\_Quan\_Tri\_CSDL\Project\DataApp\QuanLySieuThiDienMay\_data.mdf
* Kích thước: 10MB
* Kích thước tăng trưởng: 2MB

+ File nhật ký:

* Tên: HTQLBH\_log
* Đường dẫn: E:\BE\_STUDYING\He\_Quan\_Tri\_CSDL\Project\DataApp\QuanLySieuThiDienMay\_Log.ldf
* Kích thước: 5MB
* Kích thước tăng trưởng: 1MB

+ Chi tiết:

create database QuanLySieuThiDienMay

On(

Name = HTQLBH\_data,

filename = 'E:\BE\_STUDYING\He\_Quan\_Tri\_CSDL\Project\DataApp\QuanLySieuThiDienMay\_data.mdf',

size = 10MB,

filegrowth = 2MB)

Log on

(

Name = HTQLBH\_Log,

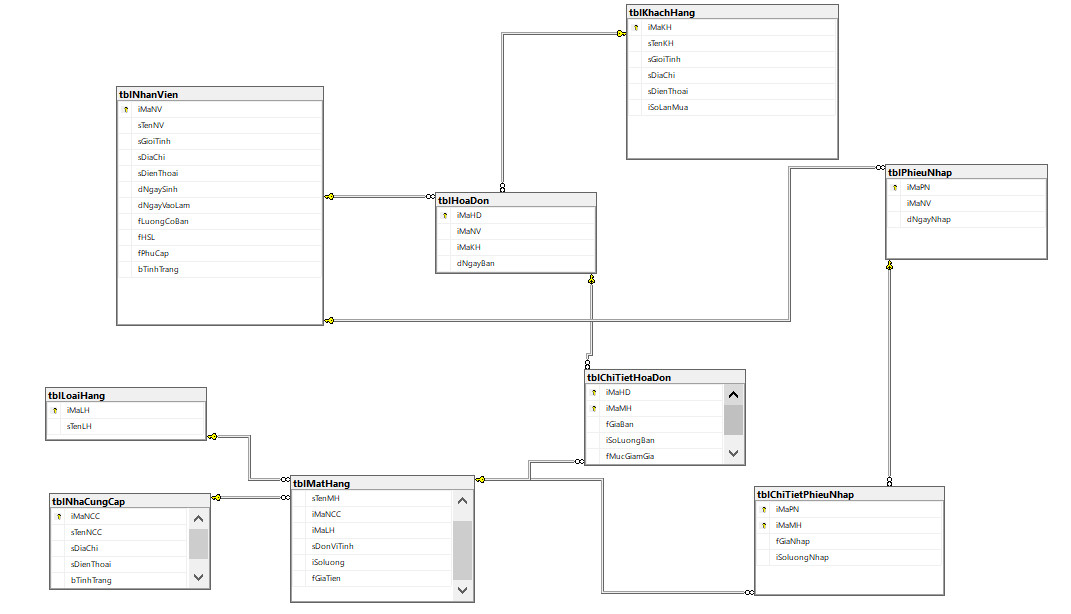
Filename = 'E:\BE\_STUDYING\He\_Quan\_Tri\_CSDL\Project\DataApp\QuanLySieuThiDienMay\_Log.ldf',

size =5MB,

filegrowth = 1MB);

### 2.1.2. Tạo bảng

- Sơ đồ liên kết giữa các bảng:



- Tạo các bảng:

+ Nhân viên:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblNhanVien*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaNV** | Mã nhân viên | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| sTenNV | Tên nhân viên | Nvarchar | 30 | - Not NULL |
| sGioiTinh | Giới tính | Nvarchar | 3 | - Not NULL  - Check (sGioiTinh=’Nam’ or sGioiTinh=N’Nữ’) |
| sDiaChi | Địa chỉ | Nvarchar | 50 |  |
| sDienThoai | Điện thoại | Varchar | 12 | - Not NULL  - Unique |
| dNgaySinh | Ngày sinh | Date |  |  |
| dNgayVaoLam | Ngày vào làm | Date |  |  |
| fLuongCoBan | Lương cơ bản | Float |  |  |
| fHeSoLuong | Hệ số lương | Float |  |  |
| fPhuCap | Phụ cấp | Float |  |  |
| bTinhTrang | Tình trạng | Bit |  | - Default ‘1’ |

create table tblNhanVien

(

iMaNV int identity(1,1) not null primary key,

sTenNV Nvarchar(30) not null,

sGioiTinh Nvarchar (3) not null check (sGioiTinh='Nam' or sGioiTinh=N'Nữ' ),

sDiaChi Nvarchar(50),

sDienThoai varchar(12) not null ,

dNgaySinh date,

dNgayVaoLam date,

fLuongCoBan float,

fHSL float,

fPhuCap float,

bTinhTrang BIT not null default(1)

Unique(sDienThoai)

);

+ Khách hàng:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblKhachHang*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaKH** | Mã khách hàng | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| sTenKH | Tên khách hàng | Nvarchar | 30 | - Not NULL |
| sGioiTinh | Giới tính | Nvarchar | 3 | - Not NULL |
| sDiaChi | Địa chỉ | Nvarchar | 50 | - Not NULL |
| sDienThoai | Điện thoại | Varchar | 12 | - Not NULL  - Unique |
| iSoLanMua | Số lần mua | Int |  | - Default ‘0’ |

create table tblKhachHang

(

iMaKH int identity(1,1) not null primary key,

sTenKH Nvarchar(30) not null,

sGioiTinh nvarchar(3) check (sGioiTinh='Nam' or sGioiTinh=N'Nữ' ),

sDiaChi nvarchar(50) not null,

sDienThoai varchar(12) not null Unique,

iSoLanMua int not null default 0 check(iSolanmua >=0)

);

+ Nhà cung cấp:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblNhaCungCap*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaNCC** | Mã nhà cung cấp | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| sTenNCC | Tên nhà cung cấp | Nvarchar | 20 | - Not NULL |
| sDiaChi | Địa chỉ | Nvarchar | 50 | - Not NULL |
| sDienThoai | Điện thoại | Varchar | 12 | - Not NULL  - Unique |
| bTinhTrang | Tình trạng | Bit |  | - Default ‘1’ |

create table tblNhaCungCap

(

iMaNCC int identity(1,1) not null primary key,

sTenNCC nvarchar(20) not null,

sDiaChi Nvarchar(50) not null,

sDienThoai varchar(12) not null Unique,

bTinhTrang BIT not null default 1

);

+ Loại hàng:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblLoaiHang*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaLH** | Mã loại hàng | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| sTenLH | Tên loại hàng | Nvarchar | 30 | - Not NULL |

create table tblLoaiHang

(

iMaLH int identity(1,1) not null primary key,

sTenLH nvarchar(30) not null

);

+ Mặt hàng:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblMatHang*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaMH** | Mã mặt hàng | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| sTenMH | Tên mặt hàng | Nvarchar | 20 | - Not NULL |
| *iMaNCC* | Mã nhà cung cấp | Int |  | - Not NULL  - Foreign key references tblNhaCungCap(iMaNCC) |
| *iMaLH* | Mã loại hàng | Int |  | - Not NULL  - Foreign key references tblLoaiHang(iMaLH) |
| sDonViTinh | Đơn vị tính | Nvarchar | 10 | - Not NULL |
| iSoLuong | Số lượng | Int |  | - Default ‘0’ |
| fGiaTien | Giá tiền | Float |  |  |

create table tblMatHang

(

iMaMH int identity(1,1) not null primary key,

sTenMH nvarchar(20) not null,

iMaNCC int not null,

iMaLH int not null,

sDonViTinh Nvarchar(10) not null,

iSoluong int not null default 0 check(isoluong >=0),

fGiaTien float,

foreign key (iMaNCC) references tblnhacungcap(imancc),

foreign key (iMaLH) references tblloaihang(imalh)

);

+ Phiếu nhập:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblPhieuNhap*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaPN** | Mã phiếu nhập | Int |  | - Int identity (1,1)  - Not NULL Primary key |
| *iMaNV* | Mã nhân viên | Int |  | - Not NULL  - Foreign key references tblNhanVien(iMaNV) |
| dNgayNhap | Ngày nhập | Date |  |  |

create table tblPhieuNhap

(

iMaPN int identity(1,1) not null primary key,

iMaNV int not null,

dNgayNhap date,

foreign key (iMaNV) references tblNhanVien(iMaNV)

);

+ Chi tiết phiếu nhập:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblChiTietPhieuNhap*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| ***iMaPN*** | Mã phiếu nhập | Int |  | - Not NULL  - Primary key  - Foreign key references tblPhieuNhap(iMaPN) |
| ***iMaMH*** | Mã mặt hàng | Int |  | - Not NULL  - Primary key  - Foreign key references tblMatHang(iMaMH) |
| fGiaNhap | Mã nhà cung cấp | Float |  |  |
| iSoluongNhap | Số lượng nhập | Int |  |  |

create table tblChiTietPhieuNhap

(

iMaPN int not null,

iMaMH int not null,

fGiaNhap float,

iSoluongNhap int not null default 1 check(isoluongnhap >=1),

primary key (imapn,imamh),

foreign key (iMaPN) references tblPhieuNhap(imapn),

foreign key (iMaMH) references tblMatHang(imaMH)

);

+ Hóa đơn:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblHoaDon*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| **iMaHD** | Mã hóa đơn | Int |  | - Int identity (1,1)  - Not NULL  - Primary key |
| *iMaNV* | Mã nhân viên | Int |  | - Not NULL  - Foreign key references tblNhanVien(iMaNV) |
| *iMaKH* | Mã khách hàng | Int |  | - Not NULL  - Foreign key references tblKhachHang(iMaKH) |
| dNgayBan | Ngày bán | Date |  |  |

create table tblHoaDon

(

iMaHD int identity(1,1) not null primary key,

iMaNV int not null,

iMaKH int not null,

dNgayBan date,

foreign key (iManv) references tblnhanvien(imanv),

foreign key (iMaKH) references tblkhachhang(iMaKH)

);

+ Chi tiết hóa đơn:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***tblChiTietHoaDon*** | | | | |
| Tên trường | Chú thích | Kiểu dữ liệu | Độ rộng | Ràng buộc |
| ***iMaHD*** | Mã hóa đơn | Int |  | - Not NULL  - Primary key  - Foreign key references tblHoaDon(iMaHD) |
| ***iMaMH*** | Mã mặt hàng | Int |  | - Not NULL  - Primary key  - Foreign key references tblMatHang(iMaMH) |
| fGiaBan | Giá bán | Float |  | - Not Null |
| iSoLuongBan | Số lượng bán | Int |  |  |
| fMucGiamGIa | Mức giảm giá | Float |  |  |

create table tblChiTietHoaDon

(

iMaHD int not null,

iMaMH int not null,

fGiaBan float not null,

iSoLuongBan int not null default 1 check(isoluongban >=1),

fMucGiamGia float,

primary key (imahd, imamh),

foreign key (iMaHD) references tblHoadon(iMaHD),

foreign key (iMaMH) references tblMatHang(imaMH)

);

## 2.2. Các câu lệnh

### 2.2.1. Cursor

- Sử dụng Cursor để in ra những mặt hàng bán chạy nhất trong năm:

*(SV thực hiện: Huy)*

declare cur\_1 cursor dynamic scroll for

select cthd.imamh, sum(cthd.iSoLuongBan)

from tblChiTietHoaDon cthd inner join tblHoaDon hd on cthd.iMaHD = hd.iMaHD

where year(hd.dngayban) = year(getdate())

group by cthd.iMaMH,cthd.iMaHD

having sum(cthd.iSoLuongBan) >= all (select sum(cthd.iSoLuongBan)

from tblChiTietHoaDon cthd inner join tblHoaDon hd on cthd.iMaHD = hd.iMaHD

where year(hd.dngayban) = year(getdate()))

open cur\_1;

declare @MaMH int, @soluong int;

fetch first from cur\_1 into @MaMH, @soluong;

while (@@FETCH\_STATUS = 0)

begin

print N'Mặt hàng bán chạy nhất trong năm ' + cast(year(getdate()) as char(4)) + N' là :' + cast(@MaMH as char(3)) ;

print N'với số lượng bán ra: '+ cast(@soluong as nvarchar(10));

fetch next from cur\_1 into @MaMH,@soluong;

end

close cur\_1; deallocate cur\_1;

- Sử dụng Cursor để in ra những khách hàng chi tiêu nhiều nhất tại cửa hàng trong năm:

*(SV thực hiện: Tuấn)*

declare cur\_top1customer cursor dynamic scroll

for

select top(1) imakh as [Mã khách] , sum((fGiaBan \* iSoLuongBan) \*(1-fMucGiamGia)) as [Tổng tiền]

from tblHoaDon, tblChiTietHoaDon

where tblHoaDon.iMaHD = tblChiTietHoaDon.iMaHD and year(tblHoaDon.dNgayBan) = year(getdate())

group by tblHoaDon.imakh

order by [Tổng tiền] desc;

open cur\_top1customer;

declare @makh int,@chitieu float;

fetch first from cur\_top1customer into @makh,@chitieu;

print N'Khách hàng chi tiêu nhiều nhất trong năm '+cast(year(getdate()) as nvarchar(10)) + N'là'+ cast(@makh as nvarchar(10)) + N'với tổng tiền đã chi: '+ cast(@chitieu as nvarchar(10));

close cur\_top1customer;

deallocate cur\_top1customer;

- Sử dụng Cursor để in ra những hóa đơn có tổng tiền cao nhất trong năm:

*(SV thực hiện: Hiếu)*

create view view\_Bill\_with\_totalMoney

as

select tblChiTietHoaDon.iMaHD as [Mã hóa đơn],dNgayBan as [Ngày tạo hóa đơn] , count(iMaMH) as [Tổng số mặt hàng], sum((fGiaBan \* iSoLuongBan) \*(1-fMucGiamGia)) as [Tổng tiền]

from tblHoaDon, tblChiTietHoaDon

where tblHoaDon.iMaHD = tblChiTietHoaDon.iMaHD

group by tblChiTietHoaDon.iMaHD, dNgayBan;

declare cur\_3 cursor dynamic scroll

for select [Mã hóa đơn]

from view\_Bill\_with\_totalMoney

where year([Ngày tạo hóa đơn]) = year(getdate()) and [Tổng tiền] >= all(

select [Tổng tiền] from view\_Bill\_with\_totalMoney where year([Ngày tạo hóa đơn]) = year(getdate()))

open cur\_3

declare @x int;

fetch first from cur\_3 into @x;

while(@@FETCH\_STATUS=0)

begin

print N'Mã đơn có tổng tiền cao nhất trong năm: '+cast(@x as nvarchar(10));

fetch next from cur\_3 into @x;

end

close cur\_3; deallocate cur\_3;

### 2.2.2. Stored Procedure

- Thủ tục hiển thị toàn bộ nhân viên:

*(SV thực hiện: Huy)*

create proc view\_all\_Employee

as

select iMaNV as [Mã Nhân viên], sTenNV as [Họ tên], sGioiTinh as [Giới Tính] ,

sDiaChi as [Địa Chỉ] , sDienThoai as [SĐT], dNgaySinh as [Ngày Sinh], dNgayVaoLam as [Ngày vào làm],

fLuongCoBan as [Lương cơ bản], fHSL as [Hệ số lương], fPhuCap as [Phụ cấp],bTinhTrang as [Tình trạng] from tblNhanVien order by iMaNv;

- Thủ tục tìm kiếm nhân viên:

*(SV thực hiện: Tuấn)*

create procedure find\_Employee

@findX nvarchar(50)

as

select iMaNV as N'Mã Nhân viên', sTenNV as N'Họ tên', sGioiTinh as N'Giới Tính' ,

sDiaChi as N'Địa Chỉ' , sDienThoai as N'SĐT', dNgaySinh as N'Ngày Sinh', dNgayVaoLam as N'Ngày vào làm',

fLuongCoBan as N'Lương cơ bản', fHSL as N'Hệ số lương', fPhuCap as N'Phụ cấp' ,bTinhTrang as N'Tình trạng' from tblNhanVien where

imanv like '%' + @findX + '%' or sTenNv like N'%' +@findX + '%' or sGioiTinh like N'%' + @findX + '%' or sDiaChi like N'%' + @findX + '%' or sDienThoai like '%' + @findX + '%'

or dNgaySinh like '%' + @findX + '%' or dNgayVaoLam like '%' + @findX + '%' or fLuongcoban like '%' + @findX + '%' or fHSL like '%' + @findX + '%'

or fphucap like '%' + @findX + '%' or bTinhTrang like '%' + @findX + '%' order by iMaNv;

- Thủ tục hiển thị toàn bộ khách hàng:

*(SV thực hiện: Hiếu)*

create proc view\_all\_customer

as

select iMaKh as N'Mã khách hàng' , sTenKH as N'Họ tên' , sGioiTinh as N'Giới tính', sDiaChi as N'Địa chỉ' , sDienThoai as N'SĐT' ,iSolanMua as N'Số lần mua hàng'

from tblKhachHang order by iMaKH;

-Thủ tục tìm kiếm khách hàng:

*(SV thực hiện: Huy)*

create proc find\_Customer

@findX nvarchar(50)

as

select iMaKh as N'Mã khách hàng' , sTenKH as N'Họ tên' , sGioiTinh as N'Giới tính', sDiaChi as N'Địa chỉ' , sDienThoai as N'SĐT' ,iSolanMua as N'Số lần mua hàng'

from tblKhachHang

where iMaKH like '%'+@findX+'%' or sTenKH like '%'+@findX+'%' or sGioiTinh like '%'+@findX+'%'

or sDiaChi like '%'+@findX+'%' or sDienThoai like '%'+@findX+'%' or iSolanMua like '%'+@findX+'%'

order by iMaKH;

-Thủ tục hiển thị toàn bộ nhà cung cấp:

*(SV thực hiện: Tuấn)*

create proc view\_all\_Supplier

as

select iMaNCC as N'Mã nhà cung cấp', sTenNCC as N'Tên nhà cung cấp' , sDiaChi as N'Địa chỉ', sDienThoai as N'Số điện thoại',bTinhTrang as N'Tình trạng'

from tblNhaCungCap

order by iMaNCC;

- Thủ tục tìm kiếm nhà cung cấp:

*(SV thực hiện: Hiếu)*

create proc find\_Supplier

@findX nvarchar(50)

as

select

iMaNCC as N'Mã nhà cung cấp', sTenNCC as N'Tên nhà cung cấp' , sDiaChi as N'Địa chỉ', sDienThoai as N'Số điện thoại' ,bTinhTrang as N'Tình trạng'

from tblNhaCungCap

where iMaNCC like '%'+@findX+'%' or sTenNCC like '%'+@findX+'%' or sDiaChi like '%'+@findX+'%' or sDiaChi like '%'+@findX+'%'

order by iMaNCC;

- Thủ tục hiển thị toàn bộ loại hàng:

*(SV thực hiện: Huy)*

create proc view\_all\_Category

as

select

iMaLH as N'Mã loại hàng', sTenLH as N'Tên loại hàng'

from tblLoaiHang;

- Thủ tục tìm kiếm loại hàng:

*(SV thực hiện: Tuấn)*

create proc find\_Category

@findX nvarchar(50)

as

select

iMaLH as N'Mã loại hàng', sTenLH as N'Tên loại hàng'

from tblLoaiHang

where iMaLH like '%'+@findX+'%' or sTenLH like N'%'+@findX+'%'

order by iMaLH;

- Thủ tục hiển thị toàn bộ mặt hàng:

*(SV thực hiện: Hiếu)*

create proc [dbo].[view\_all\_Product]

as

select iMaMH as N'Mã mặt hàng' , sTenMH as N'Tên mặt hàng' , iMaNCC as N'Mã nhà cung cấp' , iMaLH as N'Mã loại hàng', sDonViTinh as N'Đơn vị tính',

iSoluong as N'Số lượng', fGiaTien as N'Giá tiền'

from tblMatHang

order by iMaMH;

- Thủ tục tìm kiếm mặt hàng:

*(SV thực hiện: Huy)*

create proc find\_product

@findX nvarchar(50)

as

select imamh as[Mã mặt hàng], stenmh as [Tên mặt hàng] ,iMaNCC as [Mã nhà cung cấp] , iMaLH as [Mã loại hàng], sDonViTinh as [Đơn vị tính],

iSoluong as [Số lượng], fGiaTien as [Giá tiền]

from tblMatHang

where imamh like '%'+@findX+'%' or stenmh like '%'+@findX+'%' or iMaNCC like '%'+@findX+'%' or imalh like '%'+@findX+'%' or sDonViTinh like '%'+@findX+'%' or iSoluong like '%'+@findX+'%' or fGiaTien like '%'+@findX+'%'

order by iMaMH;

- Thủ tục hiển thị toàn bộ chi tiết hóa đơn:

*(SV thực hiện: Tuấn)*

create proc view\_Bill\_detail\_X

@mahd int

as

select [Tên mặt hàng],[Số lượng mua],[Giá bán],[Mức giảm giá], [Thành tiền] from view\_all\_Bill where [Mã hóa đơn]=@mahd;

### 2.2.3. Fuction

- Function trả về bảng với các thuộc tính Mã mặt hàng, Tên mặt hàng, Giá nhập, Số lượng nhập, Thành tiền với những mặt hàng đã được nhập:

*(SV thực hiện: Tuấn)*

Create function func\_view\_detail\_Import (@MaPN int)

Returns table as

Return(

Select ctpn.iMaMH as [Mã mặt hàng], sTenMH as [Tên mặt hàng], fGiaNhap as [Giá nhập], iSoluongNhap as [Số lượng nhập], fGiaNhap \* iSoluongNhap as [Thành tiền]

from tblChiTietPhieuNhap ctpn, tblMatHang mh where ctpn.iMaMH = mh.iMaMH and ctpn.iMaPN = @MaPN);

- Function trả về bảng tổng kết doanh thu tháng:

*(SV thực hiện: Huy)*

create view view\_Bill\_with\_totalMoney

as

select tblChiTietHoaDon.iMaHD as [Mã hóa đơn],dNgayBan as [Ngày tạo hóa đơn] , count(iMaMH) as [Tổng số mặt hàng], sum((fGiaBan \* iSoLuongBan) \*(1-fMucGiamGia)) as [Tổng tiền]

from tblHoaDon, tblChiTietHoaDon

where tblHoaDon.iMaHD = tblChiTietHoaDon.iMaHD

group by tblChiTietHoaDon.iMaHD, dNgayBan;

create function f\_doanhThuThang(@x nvarchar(20))

returns float

as

begin

declare @return float

select @return= SUM([Tổng tiền])

from view\_Bill\_with\_totalMoney

where month([Ngày tạo hóa đơn]) = month(@x) and year([Ngày tạo hóa đơn]) = year(@x);

return @return;

end

- Function trả về bảng tổng kết doanh thu năm:

*(SV thực hiện: Hiếu)*

create view view\_Bill\_with\_totalMoney

as

select tblChiTietHoaDon.iMaHD as [Mã hóa đơn],dNgayBan as [Ngày tạo hóa đơn] , count(iMaMH) as [Tổng số mặt hàng], sum((fGiaBan \* iSoLuongBan) \*(1-fMucGiamGia)) as [Tổng tiền]

from tblHoaDon, tblChiTietHoaDon

where tblHoaDon.iMaHD = tblChiTietHoaDon.iMaHD

group by tblChiTietHoaDon.iMaHD, dNgayBan;

create function f\_doanhThuNam(@x nvarchar(20))

returns float

as

begin

declare @return float

select @return= SUM([Tổng tiền])

from view\_Bill\_with\_totalMoney

where year([Ngày tạo hóa đơn]) = year(@x);

return @return;

end

### 2.2.4. View

- View hiển thị toàn bộ chi tiết hóa đơn:

*(SV thực hiện: Huy)*

create view view\_all\_Bill

as

select hd.iMaHD as [Mã hóa đơn], kh.sTenKH as [Tên khách hàng], nv.sTenNV as [Tên nhân viên], mh.sTenMH as [Tên mặt hàng], cthd.iSoLuongBan as [Số lượng mua],

cthd.fGiaBan as [Giá bán], cthd.fMucGiamGia as [Mức giảm giá], cthd.fGiaBan\*cthd.iSoLuongBan\*(1-cthd.fMucGiamGia) as [Thành tiền]

from tblChiTietHoaDon cthd inner join tblHoaDon hd on cthd.iMaHD=hd.iMaHD

inner join tblKhachHang kh on kh.iMaKH=hd.iMaKH

inner join tblNhanVien nv on nv.iMaNV=hd.iMaNV

inner join tblMatHang mh on mh.iMaMH =cthd.iMaMH;

- View hiển thị toàn bộ đơn với Mã hóa đơn, Ngày tạo, Tổng số mặt hàng, Tổng tiền:

*(SV thực hiện: Tuấn)*

create view view\_Bill\_with\_totalMoney

as

select tblChiTietHoaDon.iMaHD as [Mã hóa đơn],dNgayBan as [Ngày tạo hóa đơn] , count(iMaMH) as [Tổng số mặt hàng], sum((fGiaBan \* iSoLuongBan) \*(1-fMucGiamGia)) as [Tổng tiền]

from tblHoaDon, tblChiTietHoaDon

where tblHoaDon.iMaHD = tblChiTietHoaDon.iMaHD

group by tblChiTietHoaDon.iMaHD, dNgayBan;

- View hiển thị toàn bộ chi tiết phiếu nhập:

*(SV thực hiện: Hiếu)*

create view view\_all\_Import

as

select pn.imapn as [Mã phiếu nhập] , pn.imanv as [Mã nhân viên] ,

pn.dngaynhap as [Ngày tạo phiếu] , mh.stenmh as [Tên mặt hàng] ,ctpn.fGiaNhap as [Giá nhập], ctpn.isoluongnhap as [Số lượng nhập]

from tblChiTietPhieuNhap ctpn inner join tblPhieuNhap pn on pn.imapn= ctpn.imapn

inner join tblMatHang mh on mh.imamh=ctpn.imamh;

- View hiển thị Mã phiếu nhập, Mã nhân viên, Ngày tạo phiếu, Tổng số hàng, Tổng tiền hàng:

*(SV thực hiện: Huy)*

create view view\_all\_Import\_with\_totalMoney

as

select [Mã phiếu nhập],[Mã nhân viên],[Ngày tạo phiếu],sum([Số lượng nhập]) as [Tổng số hàng],

sum([Giá nhập]\*[Số lượng nhập]) as [Tổng tiền hàng]

from view\_all\_Import group by [Mã phiếu nhập],[Mã nhân viên],[Ngày tạo phiếu];

### 2.2.5. Trigger

- Trigger chỉ được phép thêm nhân viên trên 18 tuổi vào bảng nhân viên:

*(SV thực hiện: Tuấn)*

create trigger trg\_Kiemtratuoi

on tblNhanVien

for insert

as

begin

Declare @tuoi int

select @tuoi=DATEDIFF(day,dNgaySinh,dNgayvaolam)/365 from inserted

if(@tuoi<18)

begin

Raiserror(N'Chưa đủ 18 tuổi',16,1)

rollback tran

end

else

print N'Thêm nhân viên thành công'

end;

- Trigger kiểm soát Ngày vào làm của Nhân viên <= thời gian hiện tại:

*(SV thực hiện: Hiếu)*

create trigger trg\_Kiem\_Soat\_dNgayVaoLam

on tblNhanVien

for insert

as

begin

declare @NgayVaoLam datetime

select @NgayVaoLam = dNgayvaolam from inserted

if(DATEDIFF(day,@NgayVaoLam,getdate())<0)

begin

raiserror(N'Ngày bạn nhập không hợp lệ',16,10)

rollback tran

end

else

print N'insert thành công'

end;

- Trigger tự động tăng số lần mua của Khách hàng sau mỗi lần tạo hóa đơn thành công:

*(SV thực hiện: Huy)*

create trigger trg\_TangSoLanMua

on tblHoaDon

after insert

as

begin

declare @iMaKH int

declare @iTenKH int

declare @iMaHD int

declare @iSoLanMua int

select @iMaKH = iMaKH, @iMaHD = iMaHD from inserted

select @iSoLanMua = count(iMaKH) from tblHoaDon where @iMaHD = iMaHD

update tblKhachHang

set tblKhachHang.iSoLanMua += @iSoLanMua where iMaKH = @iMaKH

end;

- Trigger tự động giảm số lượng mặt hàng trong kho sau mỗi lần tạo hóa đơn thành công:

*(SV thực hiện: Tuấn)*

create trigger trg\_GiamSoLuongMH

on tblChiTietHoaDon

instead of insert

as

begin tran

declare @fgiaban float, @fmucgiamgia float;

declare @iMaHD int

declare @iMaMH int

declare @iSoLuongBan int

select @iMaHD = iMaHD, @iMaMH = iMaMH from inserted

select @iSoLuongBan = iSoLuongBan from inserted

select @fgiaban=fgiaban from inserted;

select @fmucgiamgia=fmucgiamgia from inserted;

if((select isoluong from tblMatHang where imamh =@imaMh)< @isoluongban)

begin

rollback tran;

print N'số lượng không đủ để bán'

end

else begin

update tblMatHang

set iSoLuong -= @iSoLuongBan where iMaMH = @iMaMH;

insert into tblchiTietHoaDon values (@imahd,@imaMh,@fgiaban,@isoluongban,@fmucgiamgia);

end

commit tran;

- Trigger tự động xóa toàn bộ thông tin trong bảng chi tiết hóa đơn khi xóa hóa đơn đó trong bảng hóa đơn, đồng thời giảm số lần mua của khách hàng trong bảng khách hàng:

*(SV thực hiện: Hiếu)*

create trigger trig\_delete\_Bill

on tblHoaDon

instead of delete

as begin tran

declare @mahd int;

select @mahd = imahd from deleted;

delete from tblChiTietHoaDon where imahd = @mahd;

delete from tblHoaDon where imahd=@mahd;

--Giảm số lần mua của khách

update tblKhachHang

set iSoLanMua-=1 where iMaKH = (select iMaKH from deleted);

- Trigger tự động tăng số lượng hàng trong kho khi xóa từ bảng chi tiết hóa đơn:

*(SV thực hiện: Huy)*

create trigger trig\_delete\_bill\_detail

on tblChiTietHoaDon

after delete

as begin tran

update tblMatHang

set iSoluong += (select iSoluong from deleted) where iMaMH = (select iMaMH from deleted);

commit tran;

- Trigger tự động xóa toàn bộ thông tin trong bảng chi tiết phiếu nhập khi xóa một phiếu nhập trong bảng phiếu nhập:

*(SV thực hiện: Tuấn)*

create trigger trig\_delete\_Import

on tblPhieuNhap

instead of delete

as begin tran

declare @mapn int;

select @mapn = imapn from deleted;

delete from tblChiTietPhieuNhap where imapn = @mapn;

delete from tblPhieuNhap where imapn = @mapn;

commit tran;

- Trigger tự động giảm số lượng hàng trong bảng mặt hàng khi xóa một phiếu nhập từ bảng chi tiết phiếu nhập:

*(SV thực hiện: Hiếu)*

create trigger trig\_delete\_import\_detail

on tblChiTietPhieuNhap

after delete

as begin tran

declare @iSoLuongNhap int

select @iSoLuongNhap = iSoLuongNhap from deleted

update tblMatHang

set iSoluong -= @iSoLuongNhap where iMaMH = (select iMaMH from deleted);

commit tran;

- Trigger tự động tăng số lượng mặt hàng trong bảng mặt hàng sau khi tạo một phiếu nhập trong bảng chi tiết phiếu nhập:

*(SV thực hiện: Huy)*

create trigger trg\_TangSoLuongMatHang

on tblChiTietPhieuNhap

after insert

as

begin

declare @iMaPN int

declare @iMaMH int

declare @iSoLuongNhap int

select @iMaPN = iMaPN, @iMaMH = iMaMH from inserted

select @iSoLuongNhap = iSoLuongNhap from tblChiTietPhieuNhap

update tblMatHang

set iSoLuong += @iSoLuongNhap where @iMaMH = iMaMH

end;

### 2.2.6. Transaction

- Trigger dưới dạng Transaction tự động thay đổi trạng thái làm việc thay vì xóa một nhân viên trong bảng nhân viên:

*(SV thực hiện: Tuấn)*

create trigger XoaNhanVien

on tblNhanVien

instead of delete as begin tran

If (exists (select iMaNV from tblNhanVien where iMaNV in (select iMaNV from deleted))).

update tblNhanVien

set bTinhTrang = '0' from tblNhanVien where iMaNV in (select iMaNV from deleted)

commit tran;

- Trigger dưới dạng Transaction tự động thay đổi trạng thái hợp tác thay vì xóa một nhà cung cấp trong bảng cung cấp

*(SV thực hiện: Hiếu)*

create trigger XoaNCC

on tblNhaCungCap

instead of delete

as begin tran

If (exists (select iMaNCC from tblNhaCungCap where iMaNCC in (select iMaNCC from deleted)))

update tblNhaCungCap

set bTinhTrang = '0' from tblNhaCungCap where iMaNCC in (select iMaNCC from deleted)

commit tran;

- Trigger dưới dạng Transaction chỉ cho tạo phiếu nhập với nhân viên còn đang làm việc:

*(SV thực hiện: Huy)*

create trigger TaoPhieuNhap

on tblPhieuNhap

for insert,update as

begin tran

if ((select iMaNV from inserted) not in (select iMaNV from tblNhanVien where bTinhTrang = 1))

begin

print N'Nhân viên đã nghỉ việc';

rollback tran;

end

else

print N'Thêm thành công';

commit tran;

- Trigger dưới dạng Transaction chỉ cho tạo hóa đơn với nhân viên còn đang làm việc:

*(SV thực hiện: Tuấn)*

create trigger TaoHoaDon

on tblHoaDon

for insert,update as

begin tran

if ((select iMaNV from inserted) not in (select iMaNV from tblNhanVien where bTinhTrang = 1))

begin

print N'Nhân viên đã nghỉ việc';

rollback tran;

end

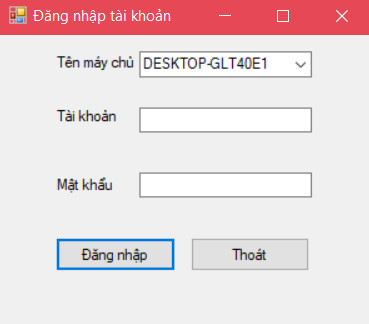
else

print N'Thêm thành công';

commit tran;

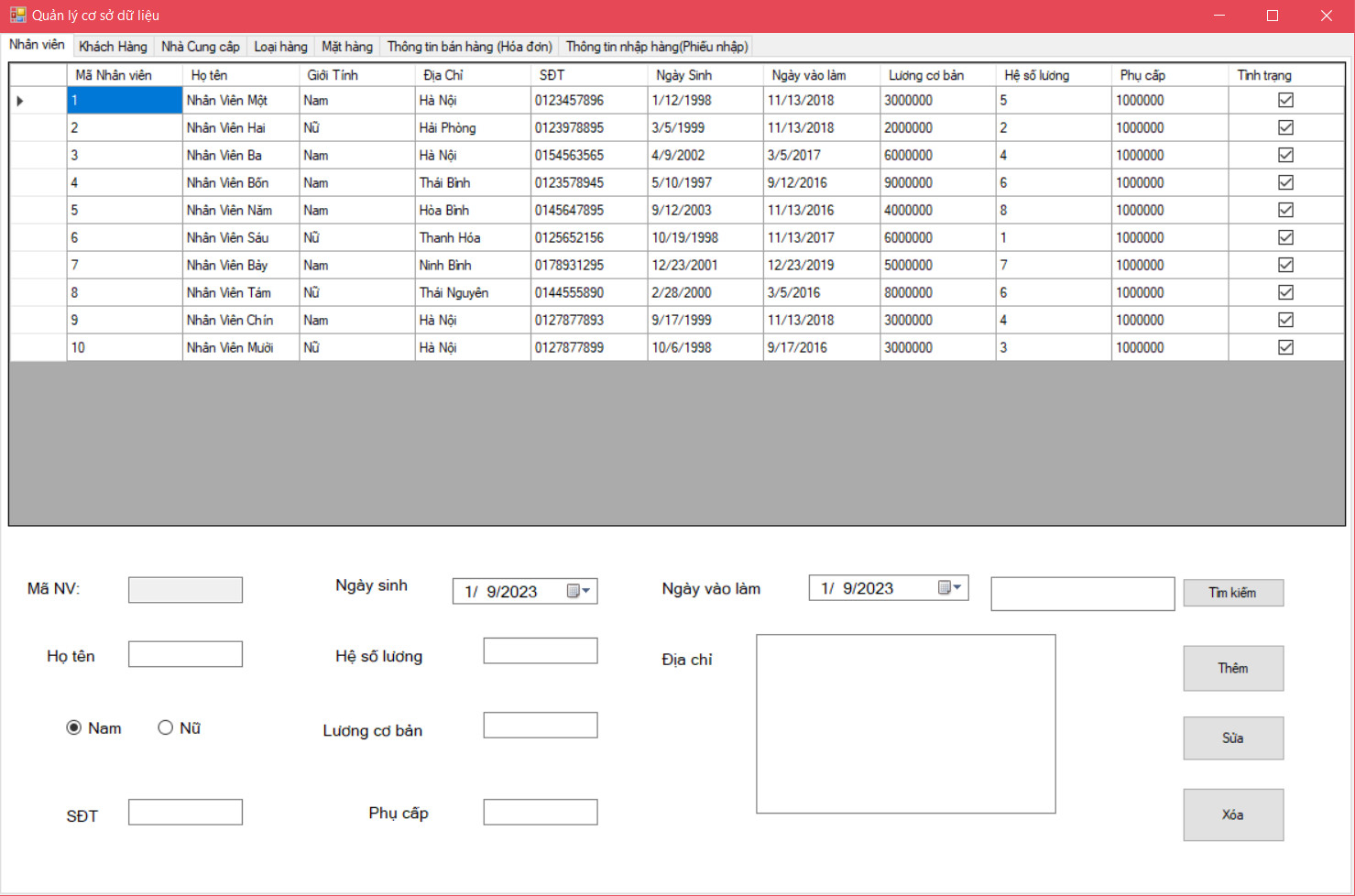
## 2.3. Giao diện

### 2.3.1. Đăng nhập

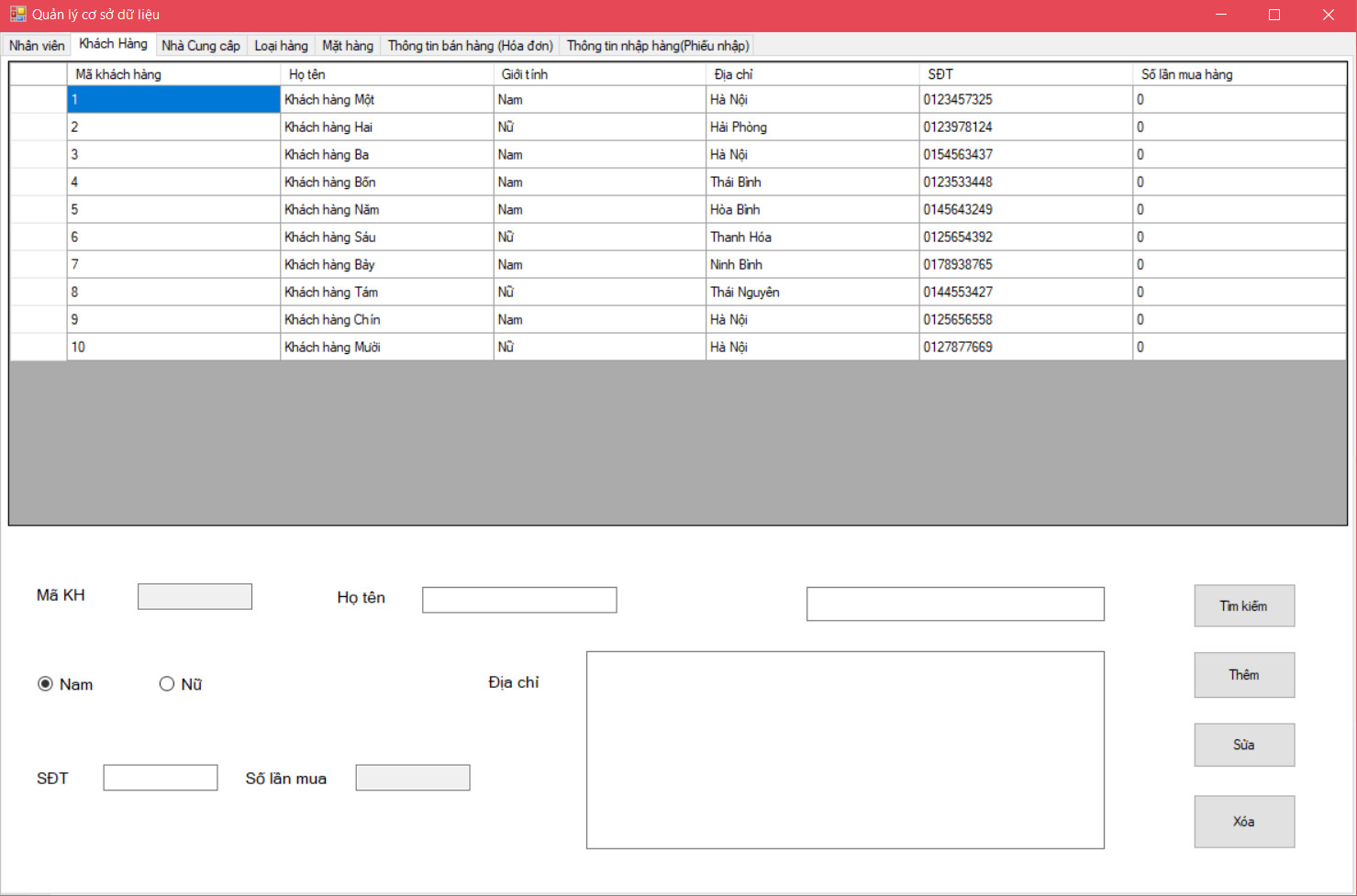


### 2.3.2. Giao diện chính

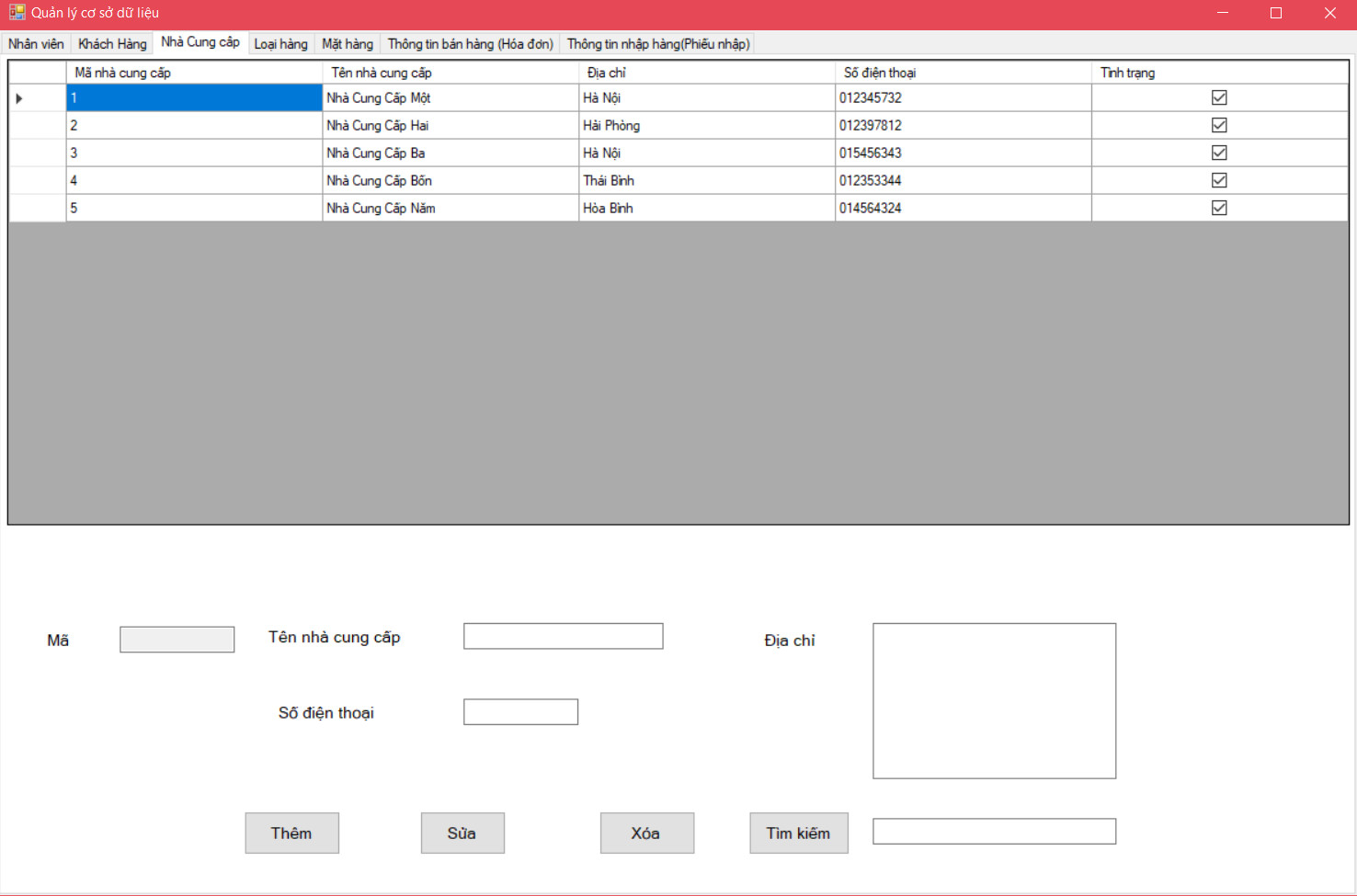
- Tab Nhân viên:



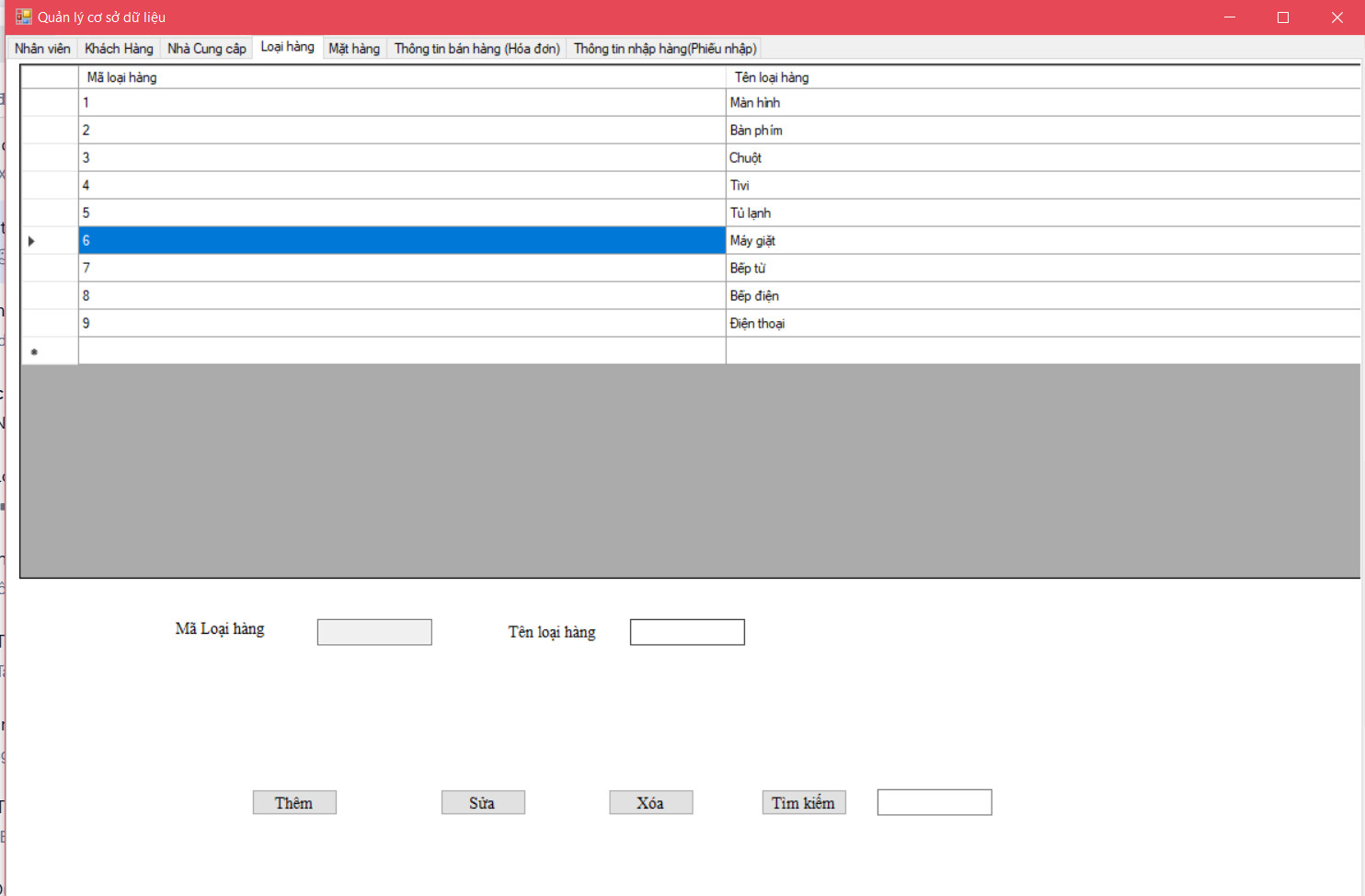
- Tab Khách hàng:



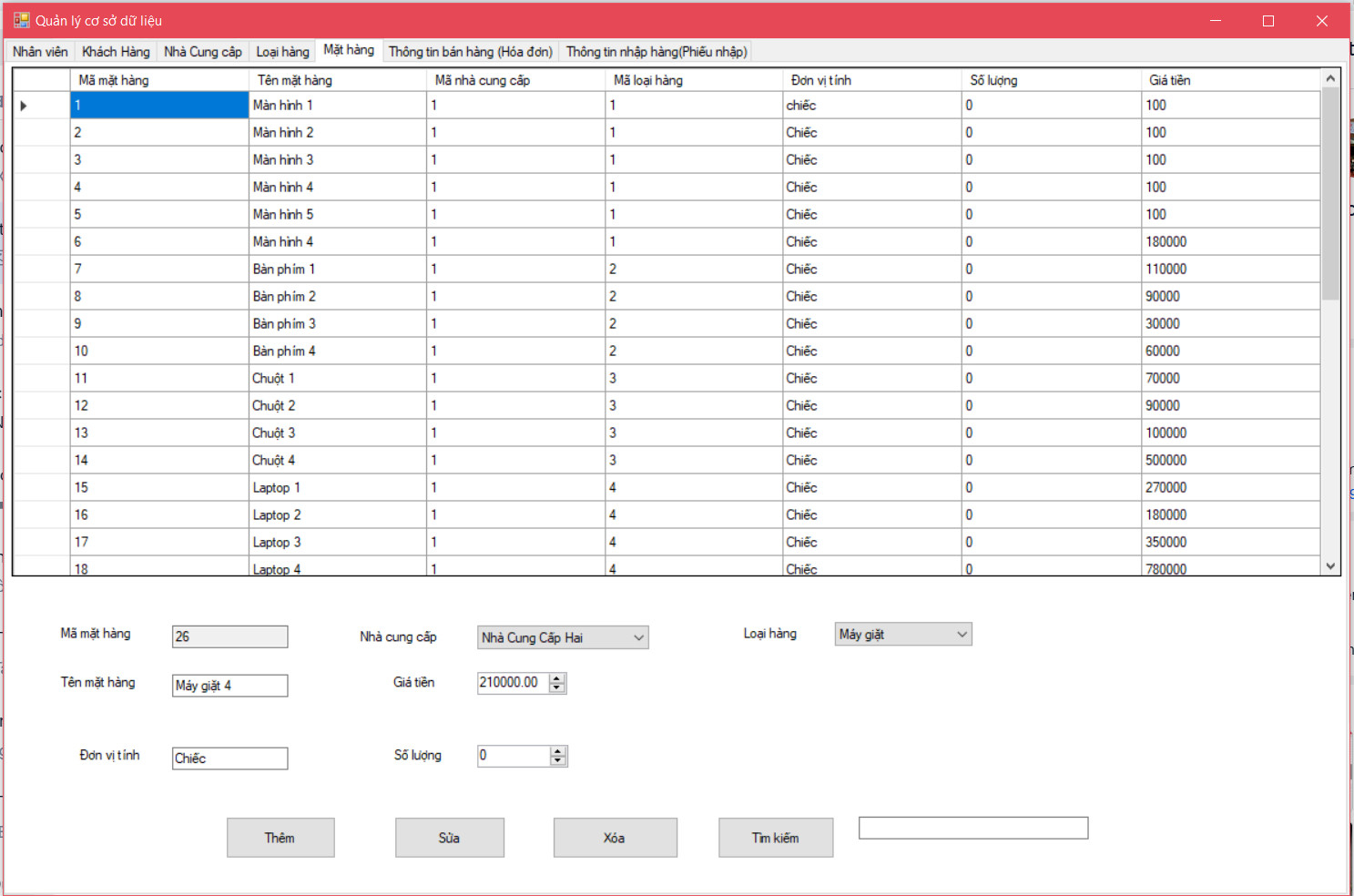
- Tab Nhà cung cấp:



- Tab Loại hàng:

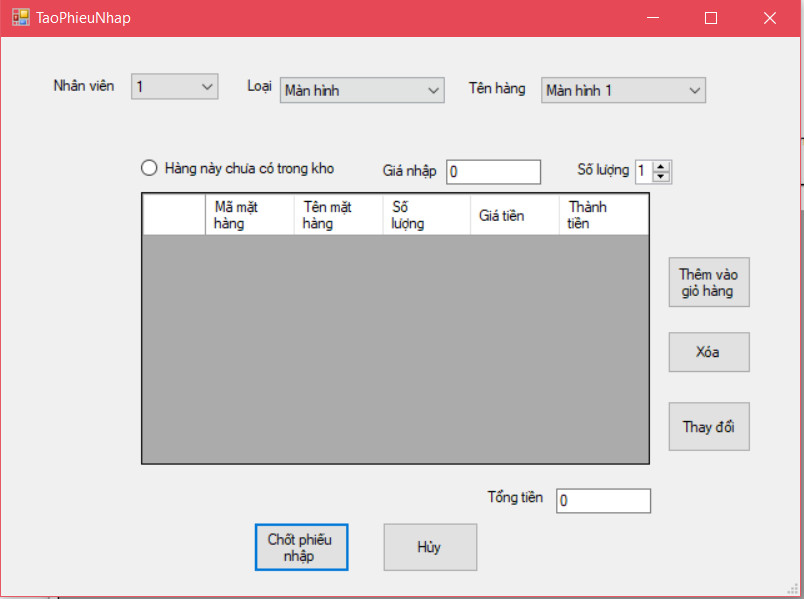


- Tab Mặt hàng:

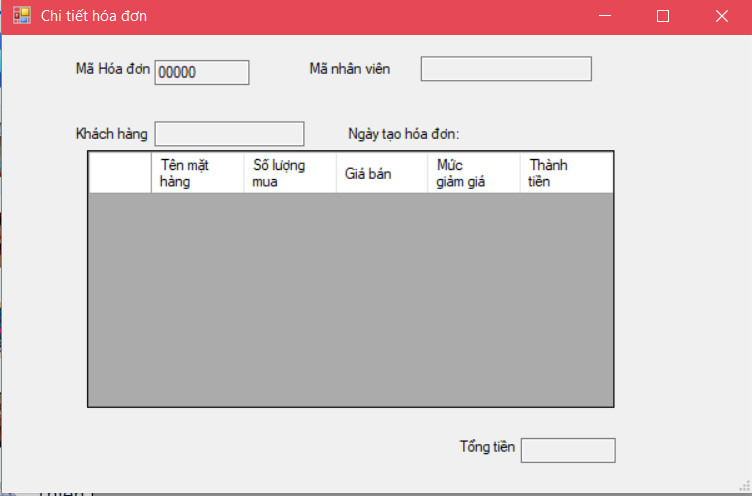


- Tab Thông tin bán hàng (Hóa đơn):

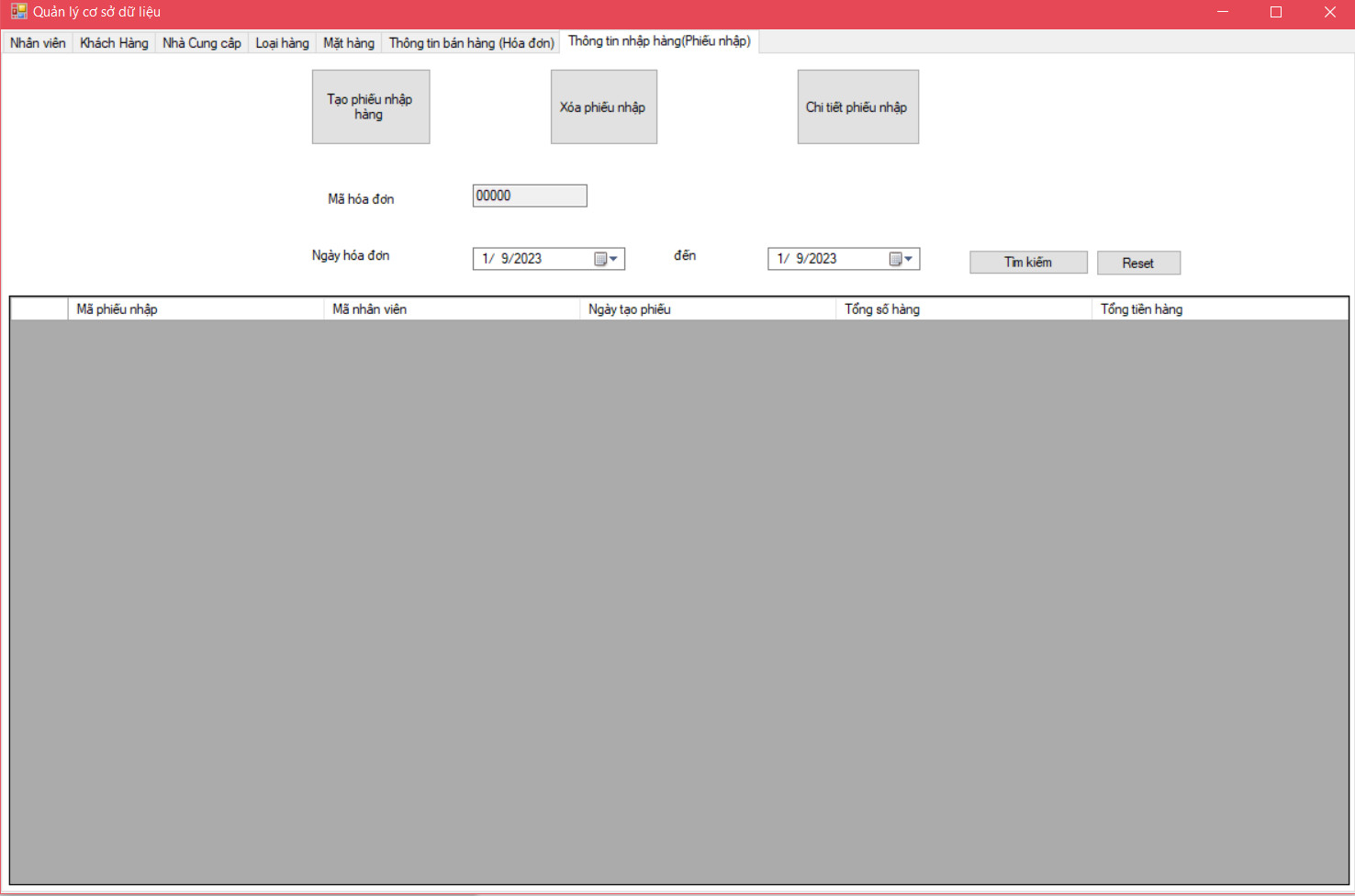
+ Button Tạo hóa đơn mua hàng:



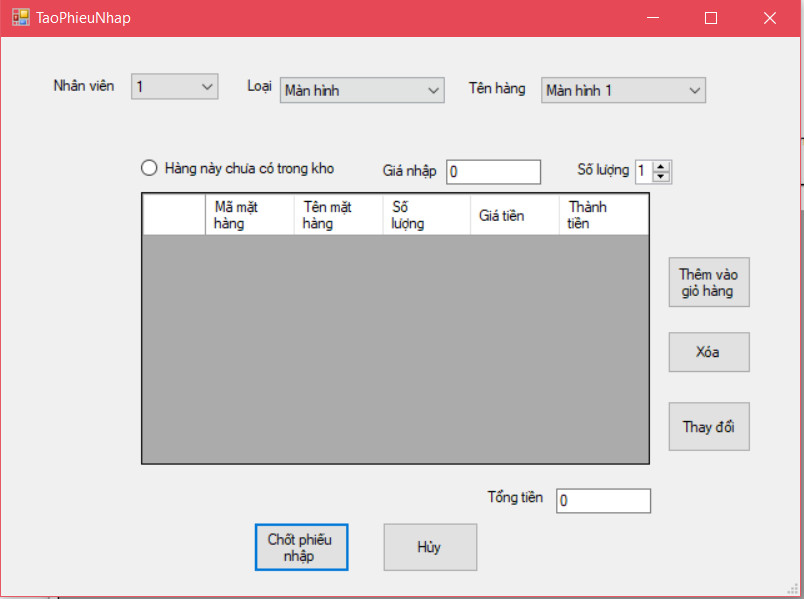
+ Button Chi tiết hóa đơn



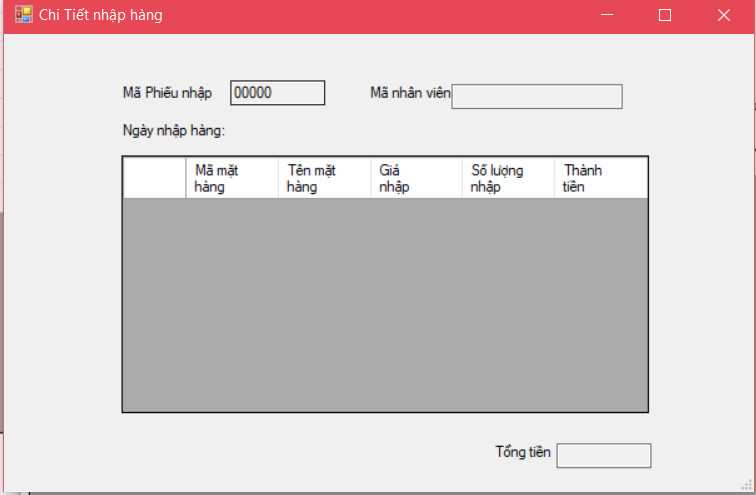
- Tab Thông tin nhập hàng (Phiếu nhập):



+ Button Tạo phiếu nhập hàng:



+ Button Chi tiết phiếu nhập:



# PHẦN 3: TỔNG KẾT

- Bài tập lớn đã giúp bài toán “Quản lý siêu thị điện máy” được phát triển và hoàn thiện phần lớn các chức năng cơ bản trong một bài toán quản lý, với CSDL được lưu trữ trên SQL Server và giao diện được lập trình trên ứng dụng Winform, sử dụng ngôn ngữ C#.

- Qua đó giúp chúng em hiểu và nắm rõ hơn về cách làm việc với CSDL, cụ thể là làm việc với SQL Sever.