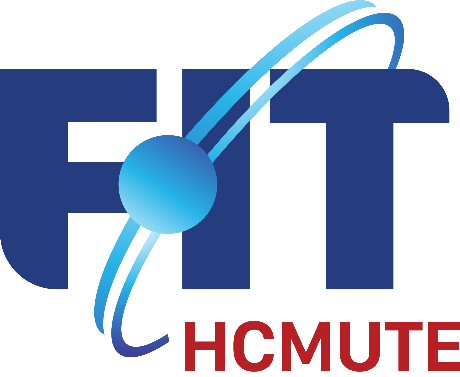
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN HỆ THỐNG THÔNG TIN**

-----🙞🙞🙞🙞🙞-----



**ĐỀ TÀI CUỐI KỲ MÔN HỆ QUẢN TRỊ CSDL**

**TÊN ĐỀ TÀI:**

**QUẢN LÝ CHUỖI CỬA HÀNG XE MÁY**

**GVHD:** *TS. Nguyễn Thành Sơn*

**Lớp HP:** *DBMS330284\_22\_2\_04CLC*

**Nhóm thực hiện:** *Nhóm 05*

**Học kỳ:** 2

**Năm học: 2022 – 2023**

*Tp. Hồ Chí Minh ngày …, tháng…, năm 2023*

**DANH SÁCH SINH VIÊN NHÓM THỰC HIỆN**

**HỌC KÌ 2 NĂM HỌC 2022-2023**

**Nhóm 05**

Đề tài: Quản lý chuỗi cửa hàng xe máy

|  |  |
| --- | --- |
| 21110290 | Nguyễn Duy Sơn |
| 21110857 | Huỳnh Ngọc Triệu |
| 21110282 | Nguyễn Khánh Quy |
| 21110166 | Nguyễn Hồng Thông Điệp |

**NHẬN XÉT CỦA GV**

Tp. Hồ Chí Minh, ngày …, tháng…, năm 2023

Giảng viên chấm điểm

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG 1](#_Toc134414819)

[1. Đặc tả đề tài 1](#_Toc134414820)

[1.1. Mô tả bài toán 1](#_Toc134414821)

[1.2. Quản lý cửa hàng xe máy 2](#_Toc134414822)

[1.3. Phân quyền 3](#_Toc134414823)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 5](#_Toc134414824)

[1. Thiết kế cơ sở dữ liệu mức quan niệm. 5](#_Toc134414825)

[2. Thiết kế cơ sở dữ liệu mức logic. 6](#_Toc134414826)

[3. Các ràng buộc cần có 7](#_Toc134414827)

[4. Cài đặt CSDL và các ràng buộc 8](#_Toc134414828)

[5. Các ràng buộc khác 12](#_Toc134414829)

[6. Trigger 13](#_Toc134414830)

[7. Các view 16](#_Toc134414831)

[CHƯƠNG 3. THIẾT KẾ CÁC CHỨC NĂNG 19](#_Toc134414832)

[1. Kết nối cơ sở dữ liệu 19](#_Toc134414833)

[2. Quản lý nhân viên 19](#_Toc134414834)

[3. Quản lý khách hàng 22](#_Toc134414835)

[4. Quản lý xe máy 24](#_Toc134414836)

[5. Quản lý phụ tùng xe máy 28](#_Toc134414837)

[6. Quản lý Bảo dưỡng xe máy 29](#_Toc134414838)

[7. Quản lý phiếu nhập 31](#_Toc134414839)

[8. Quản lý hóa đơn 32](#_Toc134414840)

[9. Quản lý nhà cung cấp 34](#_Toc134414841)

[10. Quản lý Chi tiết hóa đơn 34](#_Toc134414842)

[11. Quản lý phiếu nhập 36](#_Toc134414843)

[CHƯƠNG 4: TẠO USER VÀ PHÂN QUYỀN 38](#_Toc134414844)

[CHƯƠNG 5. THIẾT KẾ GIAO DIỆN HỆ THỐNG 46](#_Toc134414845)

[1. Form Menu Quản lý Thông tin MoToStore 46](#_Toc134414846)

[2. Form đăng nhập 47](#_Toc134414847)

[3. Các Form danh mục 47](#_Toc134414848)

[4. Quản lý nhân viên 48](#_Toc134414849)

[5. Quản lý khách hàng 53](#_Toc134414850)

[6. Quản lý xe 55](#_Toc134414851)

[7. Quản lý Phụ tùng 60](#_Toc134414852)

[8. Quản lý Bảo dưỡng 65](#_Toc134414853)

[9. Quản lý Hóa đơn 68](#_Toc134414854)

[10. Quản lý Phiếu nhập 75](#_Toc134414855)

**LỜI CẢM ƠN**

Lời đầu tiên, chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của thầy ***Nguyễn Thành Sơn*** đã giúp chúng em hoàn thành bài báo cáo cuối kì bộ môn Hệ quản trị cơ sở dữ liệu này. Nhờ những kiến thức thầy truyền đạt mà chúng em đã nắm chắc hơn được những kiến thức căn bản, nền tảng cho việc khi xây dựng một hệ thống quản trị cơ sở dữ liệu. Thông qua đồ án này, nhóm em xin trình bày lại quá trình phát triển của một hệ thống quản trị cơ sở dữ liệu, cũng như lập trình một đồ án liên quan. Trong quá trình thực hiện đồ án môn học, khó tránh khỏi những sai sót. Do đó nhóm chúng em rất mong sự góp ý đến từ thầy để bài làm của nhóm chúng em được hoàn thiện hơn, phát triển và đầy đủ hơn. Kính chúc thầy sức khỏe, hạnh phúc thành công trên con đường sự nghiệp giảng dạy. Cuối cùng chúng em cũng chân thành cảm ơn các thầy cô và các bạn cùng tham gia học bộ môn Hệ quản trị cơ sở dữ liệu đã nhiệt tình giúp đỡ động viên chúng em trong trình thực hiện đồ án cuối kỳ.

**LỜI NÓI ĐẦU**

Trong thời đại công nghệ số hiện nay, việc áp dụng Công nghệ thông tin vào quản lý và vận hành các hệ thống kinh doanh đã trở thành một xu hướng không thể phủ nhận. Trong lĩnh vực bán lẻ, việc quản lý chuỗi cửa hàng đòi hỏi sự chính xác, linh hoạt và hiệu quả để đáp ứng nhu cầu ngày càng cao của khách hàng. Đồng thời, ngành công nghiệp xe máy cũng đang phát triển mạnh mẽ và cạnh tranh khốc liệt, khiến cho việc tối ưu hóa quy trình quản lý và nâng cao hiệu suất hoạt động trở nên cấp thiết.

Chính vì những lý do đó, ứng dụng của Công nghệ thông tin đối với công tác quản lý dữ liệu trong ngân đóng một vai trò hết sức quan trọng. Và để làm rõ hơn về ứng dụng

của Công nghệ thông tin đối với công tác quản lý trong thực tế nhóm chúng em lựa chọn đề tài ***“Quản lý chuỗi cửa hàng xe máy”*** để nghiên cứu và phát triển ứng dụng của việc quản lý hệ thống dữ liệu trong môn học Hệ quản trị cơ sở dữ liệu.

Trong phần còn lại của bài, chúng em sẽ trình bày chi tiết về các yêu cầu và chức năng của hệ thống, cũng như giải pháp và công nghệ đã được sử dụng để xây dựng hệ thống. Chúng tôi cũng sẽ trình bày về quy trình triển khai và các bước kiểm thử để đảm bảo tính ổn định và hiệu suất của hệ thống.

# **CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG**

## **1. Đặc tả đề tài**

### **1.1. Mô tả bài toán**

Cửa hàng xe máy là một nơi bán các loại xe máy với mục đích phục vụ nhu cầu giao thông của người dùng. Trong cửa hàng, có nhiều loại xe máy với các thương hiệu, kiểu dáng, màu sắc khác nhau, cho phép khách hàng lựa chọn theo nhu cầu, sở thích của mình.

Ngoài việc bán các loại xe máy, cửa hàng còn cung cấp các dịch vụ hỗ trợ cho khách hàng như bảo dưỡng xe máy, cung cấp phụ tùng và linh kiện chính hãng. Và còn phải quản lý một lượng lớn nhân viên với các chức vụ khác nhau như: nhân viên tư vấn - bán hàng, nhân viên bảo dưỡng, nhân viên quản lý kho hàng, nhân viên thu ngân. Với rất nhiều khó khăn để người quản lý có thể kiểm soát hết lượng thông tin trên.

Một chuỗi cửa hàng xe máy có nhiều chi nhánh. Mỗi chi nhánh có Tên chi nhánh, Mã chi nhánh, Số Điện Thoại, Địa chỉ. Mỗi chi nhánh có một quản lý. Một chi nhánh quản lý nhiều nhân viên và quản lý một số lượng xe và phụ tùng nhất định trong kho.

Một chi nhánh có nhiều khách hàng khác nhau và điều cần thiết với mỗi hóa đơn là thông tin khách hàng sử dụng dịch vụ. Thông tin khách hàng gồm: Mã khách hàng, Họ tên khách hàng, Số điện thoại, Địa Chỉ, Số CCCD

Một khách hàng có thể có một hoặc nhiều hoá đơn. Thông tin của mỗi hoá đơn: Mã hoá đơn, Ngày lập hoá đơn, Thành tiền.

Nhân viên có các thông tin: Mã nhân viên, Tên nhân viên, Ngày sinh, Số CCCD, Giới tính, Địa chỉ, Số điện thoại, Chức vụ, Tình trạng làm.

Bảo dưỡng có chức năng bảo dưỡng xe. Bảo dưỡng cần có các thông tin sau: Mã bảo dưỡng, Tên bảo dưỡng, Phí bảo dưỡng, Thông tin bảo dưỡng.

Mỗi nhân viên theo từng chức vụ sẽ có nhiệm vụ khác nhau: Chăm sóc khách hàng và khi khách hàng mua sản phẩm sẽ lập hoá đơn để lưu thông tin khách hàng, Nhân viên quản kho có nhiệm vụ đi nhập hàng từ nhà cung cấp và lập báo cáo về sản lượng,

Mặt hàng có thể là xe hay phụ tùng có các thông tin như: tên, mã mặt hàng, đơn giá, hãng.

Xe thì có thêm 1 số thuộc tính: Loại xe, Màu sắc, Khối Lượng bản thân, Dài x Rộng x Cao, Khoảng cách trục bánh xe, Độ cao Yên, Khoảng sáng gầm xe, Dung tích bình xăng, Kích cỡ lớp trước, Kích cỡ lớp sau, Phuộc trước, Phuộc sau, Loại động cơ, Công suất tối đa, Dung tích nhớt máy, Loại truyền động, Hệ thống khởi động, Moment cực đại, Dung tích xy-lanh, Đường kính x Hành trình pít tông, Tỷ số nén, Động cơ - Công nghệ

Phụ tùng gồm có: Chất lượng, Loại phụ kiện.

Nhà cung cấp thì có mã nhà cung cấp, tên nhà cung cấp, địa chỉ.

Phiếu nhập gồm có ngày nhập gồm có tổng tiền, ngày nhập, mã phiếu nhập.

### **1.2. Quản lý cửa hàng xe máy**

* **Quản lý phụ tùng**
  + Thêm, sửa dữ liệu thông tin của phụ tùng (mã, tên, số lượng,...)
  + Tìm kiếm thông tin phụ tùng
  + Quản lý số lượng, thời gian tồn kho của phụ tùng
  + Quản lý số lượng bán ra của phụ tùng
  + Quản lý giá bán của phụ tungSSS
* **Quản lý bán hàng xe**
  + Thêm, sửa dữ liệu thông tin của xe (mã, tên, số lượng,...)
  + Tìm kiếm thông tin xe
  + Quản lý số lượng, thời gian tồn kho của xe
  + Quản lý số lượng bán ra của xe
  + Quản lý giá bán của xe
* **Quản lý nhân viên**
  + Thêm, sửa, xóa dữ liệu thông tin của nhân viên (mã, họ tên,...)
  + Lọc thông tin nhân viên
  + Phân quyền theo nhóm
* **Quản lý khách hàng**
  + Thêm, sửa dữ liệu thông tin của khách hàng (mã, họ tên,..)
  + Tìm kiếm thông tin khách hàng
* **Quản lý mua hàng**
  + Xuất hoá đơn
  + Quản lý chi tiết hoá đơn

### **1.3. Phân quyền**

Phân quyền trong hệ thống quản lý cửa hàng xe máy là việc quản lý quyền và khả năng truy cập của nhân viên đối với tài nguyên và thông tin của cửa hàng.

Việc này giúp kiểm soát truy cập và giúp bảo mật, góp phần tăng cường an ninh, ngăn chặn các hoạt động xâm nhập, lộ thông tin hay thậm chí là mất cắp.

Thông qua vai trò, chức năng, trình độ và một số yếu tố mà quyền hạn của mọi người trong cửa hàng xe sẽ được phân khác nhau. Cụ thể sẽ là:

* **Nhân viên tư vấn - bán hàng:**

Đây sẽ là các nhân viên có quyền được truy cập những thông tin cơ bản về cửa hàng, như là số lượng hàng hóa hiện trong cửa hàng, thông tin của hàng hóa, số lượng ở các chi nhánh và nhà cung cấp, cũng như quyền yêu cầu xin nhập thêm các mặc hàng, và theo dõi đơn, quyền truy cập thông tin khách hàng. Từ đó nắm bắt những thông tin mới nhất, kịp thời cung cấp và phục vụ cho khách hàng, đưa ra lựa chọn tốt nhất, thúc đẩy việc bán hàng diễn ra và giải quyết những thắc mắc, vấn đề xảy ra sau khi mua hàng.

* **Nhân viên thu ngân**

Nhân viên thu ngân được cung cấp quyền truy cập những thông tin liên quan đến tài chính, như giá xe, phụ kiện, một số thông tin về khách hàng và thông tin chi tiết về giao dịch được hiện, quyền quản lý ngân sách, thu nhập, chi phí, phân phối thu mua hàng hóa. Dựa vào đó để lập nên thống kê cho cửa hàng, nhằm quản lý chặt chẽ những vấn đề liên quan đến tài chính, tránh hiện tượng thiếu hụt, mất mát, đưa ra những số liệu chính xác nhất, nhằm nâng cao năng suất của cửa hàng.

* **Nhân viên sửa chữa, bảo dưỡng**

Nhân viên bảo dưỡng sẽ có quyền sử dụng các trang thiết bị và vật tư trong cửa hàng, nhưng sẽ phải báo cáo lại, và sẽ góp ý về những mặt hàng được dùng nhiều để nhập thêm, cũng như giảm bớt những mặt hàng không cần thiết, ngoài ra còn có quyền sử dụng xe để kiểm tra và bảo dưỡng định kỳ, qua đó có thể đảm bảo thiết bị của cửa hàng sẽ luôn ở trạng thái tốt nhất, làm hài lòng khách hàng với những dịch vụ kèm theo khi mua sản phẩm.

* **Nhân viên kho hàng**

Quyền của nhân viên kho hàng là sử dụng kho, truy cập thông tin và sử dụng hàng hóa, từ đó thực hiện nhiệm vụ nhiệm vụ sắp xếp, thống kê trang thiết bị, quản lý lịch trình, theo dõi hàng, để đảm bảo nhu cầu cho khách hàng, đảm bảo hàng hóa sẽ luôn luôn đầy đủ và ở trong tình trạng mới như ban đầu.

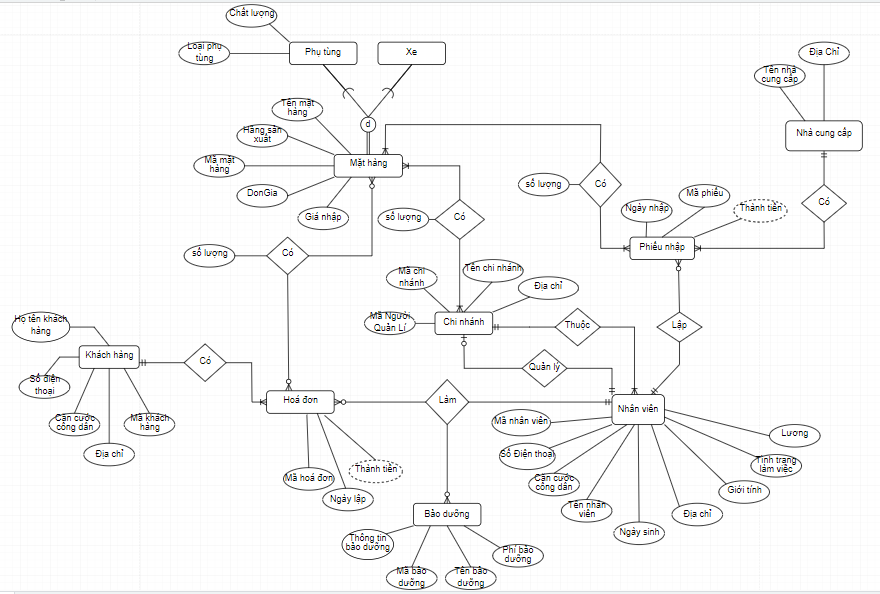
* **Quản lý**

Quản lý sẽ có quyền hạn truy cập và quản lý nhiều hơn những nhân viên khác, và sẽ là người quản lý các nhân viên của cửa hàng, sẽ là người đứng ra để nhận yêu cầu từ giám đốc và giao nhiệm vụ cho các nhân viên, là người thúc đẩy và quản lý quá trình làm việc của nhân viên, để đảm bảo mọi khâu đều liền mạch, ổn định, nâng cao hiệu suất làm việc của mọi người, cũng là người quản lý thu chi, là người nhận đề xuất của nhân viên và báo cáo.

# **CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## **1. Thiết kế cơ sở dữ liệu mức quan niệm.**

Từ mô tả về dữ liệu cần có ở phần mô tả bài toán, hình thành được sơ đồ thực thể kết hợp (ERD).



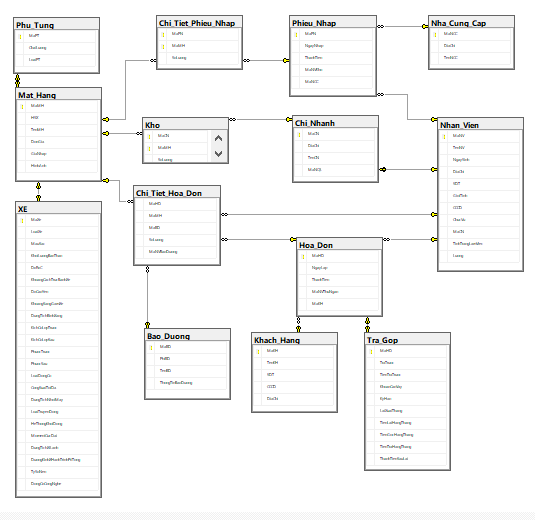
**Các thuộc tính của thực thể**

|  |  |  |
| --- | --- | --- |
| **STT** | **Thực thể** | **Thuộc tính** |
| 1 | XE | MaXe, LoaiXe, MauSac, KhoiLuongBanThan, DxRxC, KhoangCachTrucBanhXe, DoCaoYen, KhoangSangGamXe, DungTichBinhXang, KichCoLopTruoc, KichCoLopSau, PhuocTruoc, PhuocSau, LoaiDongCo, CongSuatToiDa, DungTichNhotMay, LoaiTruyenDong, HeThongKhoiDong, MomentCucDai, DungTichXy-Lanh, DuongKinhxHanhTrinhPitTong, TySoNen, DongCoCongNghe |

## **2. Thiết kế cơ sở dữ liệu mức logic.**

Từ sơ đồ thực thể kết hợp (ERD), ta có các lược đồ quan hệ:

1. **Mat\_Hang**(MaMH, HSX, TenMH, DonGia, GiaNhap, HinhAnh)
2. **Phu\_Tung**(MaPT, ChatLuong, LoaiPT, MaMH)
3. **XE**( MaXe, LoaiXe, MauSac, KhoiLuongBanThan, DxRxC, KhoangCachTrucBanhXe, DoCaoYen, KhoangSangGamXe, DungTichBinhXang, KichCoLopTruoc, KichCoLopSau, PhuocTruoc, PhuocSau, LoaiDongCo, CongSuatToiDa, DungTichNhotMay, LoaiTruyenDong, HeThongKhoiDong, MomentCucDai, DungTichXy-Lanh, DuongKinhxHanhTrinhPitTong, TySoNen, DongCoCongNghe
4. **Nha\_Cung\_Cap**(MaNCC, DiaChi, TenNCC)
5. **Chi\_Nhanh**(MaCN, DiaChi, TenCN, MaNQL)
6. **Khach\_Hang**(MaKH, TenKH, SDT, CCCD, DiaChi)
7. **Nhan\_Vien**(MaNV, TenNV, NgaySinh, DiaChi, SDT, GioiTinh, CCCD, ChucVu, MaCN, TinhTrangLamViec, Luong)
8. **Bao\_Duong**(MaBD, PhiBD, TenBD, ThongTinBaoDuong)
9. **Hoa\_Don**(MaHD, NgayLap, ThanhTien, MaNVThuNgan, MaKH)
10. **Chi\_Tiet\_Hoa\_Don**(MaHD, MaMH, MaBD, SoLuong, MaNVBaoDuong)
11. **Phieu\_Nhap**(MaPN, NgayNhap, ThanhTien, MaNVKho, MaNCC)
12. **Chi\_Tiet\_Phieu\_Nhap**(MaPN, MaMH, SoLuong)
13. **Kho**(MaCN, MaMH, SoLuong)



## **3. Các ràng buộc cần có**

|  |  |  |
| --- | --- | --- |
| **STT** | **TABLE** | **RÀNG BUỘC** |
| **1** | Mat\_Hang | Ràng buộc khoá chính MaMH |
| **2** | Phu\_Tung | Ràng buộc khoá chính MaPT, ràng buộc khoá ngoại MaMH tham chiếu đến **Mat\_Hang** |
| **3** | XE | Ràng buộc khoá chính MaXe, ràng buộc khoá ngoại MaMH tham chiếu đến **Mat\_Hang** |
| **4** | Nha\_Cung\_Cap | Ràng buộc khoá chính MaNCC |
| **5** | Chi\_Nhanh | Ràng buộc khoá chính MaCN , ràng buộc khoá ngoại MaNQL tham chiếu đến **Nhan\_vien** |
| **6** | Khach\_Hang | Ràng buộc khoá chính MaKH |
| **7** | Nhan\_Vien | Ràng buộc khoá chính MaNV, ràng buộc khoá ngoại MaCN tham chiếu đến **Chi\_Nhanh** |
| **8** | Bao\_Duong | Ràng buộc khoá chính MaBD |
| **9** | Hoa\_Don | Ràng buộc khoá chính MaHD, ràng buộc khoá ngoại MaKH , MaNVThuNgan tham chiếu đến **Khach\_Hang, Nhan\_Vien** |
| **10** | Chi\_Tiet\_Hoa\_Don | Ràng buộc khoá ngoại MaHD, MaKH, MaBD, MaNVBaoDuong tham chiếu đến **Hoa\_Don**, **Khach\_Hang, Bao\_Duong, Nhan\_Vien** |
| **11** | Phieu\_Nhap | Ràng buộc khoá chính MaPN, ràng buộc khoá ngoại MaNCC, MaNVKho tham chiếu đến **Nha\_Cung\_Cap, Nhan\_Vien** |
| **12** | Chi\_Tiet\_Phieu\_Nhap | Ràng buộc khoá chính MaPN , MaMH, ràng buộc khoá ngoại MaPN, MaMH tham chiếu đến **Phieu\_Nhap, Mat\_Hang** |
| **13** | Kho | Ràng buộc khoá chính MaCN , MaMH, ràng buộc khoá ngoại MaCN, MaMH tham chiếu đến **Chi\_Nhanh, Mat\_Hang** |

## **4. Cài đặt CSDL và các ràng buộc**

4.1. Tạo bảng và các ràng buộc đơn giản

***Bảng Mặt hàng***

|  |
| --- |
| CREATE TABLE Mat\_Hang  (  MaMH nvarchar(11) PRIMARY KEY,  HSX nvarchar(30),  TenMH nvarchar(100),  DonGia money check (DonGia>0),  GiaNhap money check (GiaNhap>0),  HinhAnh Image  ) |

***Bảng Phụ tùng***

|  |
| --- |
| CREATE TABLE Phu\_Tung  (  MaPT nvarchar(11) PRIMARY KEY,  ChatLuong nvarchar(30),  LoaiPT nvarchar(100),  FOREIGN KEY (MaPT) REFERENCES Mat\_Hang(MaMH) ON DELETE CASCADE,  ) |

***Bảng Xe***

|  |
| --- |
| CREATE TABLE XE (  MaXe nvarchar(11) PRIMARY KEY,  LoaiXe nvarchar(MAX),  MauSac nvarchar(MAX),  KhoiLuongBanThan nvarchar(max),  DxRxC nvarchar(max),  KhoangCachTrucBanhXe nvarchar(max),  DoCaoYen nvarchar(max),  KhoangSangGamXe nvarchar(max),  DungTichBinhXang nvarchar(max),  KichCoLopTruoc nvarchar(max),  KichCoLopSau nvarchar(max),  PhuocTruoc nvarchar(max),  PhuocSau nvarchar(max),  LoaiDongCo nvarchar(max),  CongSuatToiDa nvarchar(max),  DungTichNhotMay nvarchar(max),  LoaiTruyenDong nvarchar(max),  HeThongKhoiDong nvarchar(max),  MomentCucDai nvarchar(max),  DungTichXiLanh nvarchar(max),  DuongKinhXHanhTrinhPitTong nvarchar(max),  TySoNen nvarchar(max),  DongCoCongNghe nvarchar(max),  FOREIGN KEY (MaXe) REFERENCES Mat\_Hang(MaMH) ON DELETE CASCADE,  ); |

***Bảng Nhà cung cấp***

|  |
| --- |
| CREATE TABLE Nha\_Cung\_Cap  (  MaNCC nvarchar(11) PRIMARY KEY,  DiaChi nvarchar(75) not null,  TenNCC nvarchar(40) not null,  SDT nvarchar(11) null,  ) |

***Bảng Chi nhánh***

|  |
| --- |
| CREATE TABLE Chi\_Nhanh  (  MaCN nvarchar(11) PRIMARY KEY,  DiaChi nvarchar(75) not null,  TenCN nvarchar(40) not null,  MaNQL nvarchar(11),  );  go  alter table Chi\_Nhanh  add FOREIGN KEY (MaNQL) REFERENCES Nhan\_Vien(MaNV) on delete set null |

***Bảng Khách hàng***

|  |
| --- |
| CREATE TABLE Khach\_Hang  (  MaKH nvarchar(11) PRIMARY KEY,  TenKH nvarchar(40) not null,  SDT nvarchar(11) unique check (len(SDT)=10),  GioiTinh nvarchar(5) not null,  CCCD nvarchar(13) unique check (len(CCCD)=12),  DiaChi nvarchar(75) not null,  ); |

***Bảng Nhân viên***

|  |
| --- |
| CREATE TABLE Nhan\_Vien  (  MaNV nvarchar(11) PRIMARY KEY,  TenNV nvarchar(40) ,  NgaySinh date ,  DiaChi nvarchar(75) ,  SDT nvarchar(11) unique check (len(SDT)=10),  GioiTinh nvarchar(4) CHECK (GioiTinh = N'Nam' OR GioiTinh = N'Nữ'),  CCCD nvarchar(13) unique check (len(CCCD)=12),  ChucVu nvarchar(50),  MaCN nvarchar(11),  TinhTrangLamViec bit default 1,  Luong money,  FOREIGN KEY (MaCN) REFERENCES Chi\_Nhanh(MaCN),  ); |

***Bảng Bảo dưỡng***

|  |
| --- |
| CREATE TABLE Bao\_Duong  (  MaBD nvarchar(11) PRIMARY KEY,  PhiBD money check (PhiBD>0),  TenBD nvarchar(30),  ThongTinBaoDuong nvarchar(50)  ) |

***Bảng Hoá đơn***

|  |
| --- |
| CREATE TABLE Hoa\_Don  (  MaHD nvarchar(11) PRIMARY KEY,  NgayLap date not null check (DATEDIFF(day, NgayLap, GETDATE())>=0),  ThanhTien money default 0,  MaNVThuNgan nvarchar(11) ,  MaKH nvarchar(11),  FOREIGN KEY (MaNVThuNgan) REFERENCES Nhan\_Vien(MaNV),  FOREIGN KEY (MaKH) REFERENCES Khach\_Hang(MaKH)  ) |

***Bảng Chi tiết hoá đơn***

|  |
| --- |
| CREATE TABLE Chi\_Tiet\_Hoa\_Don  (  MaHD nvarchar(11) not null,  MaMH nvarchar(11) null,  MaBD nvarchar(11) null,  SoLuong int default 1 check(SoLuong>=0),  MaNVBaoDuong nvarchar(11) null,  unique (MaHD,MaMH,MaBD) ,  FOREIGN KEY (MaHD) REFERENCES Hoa\_Don(MaHD) ON DELETE CASCADE,  FOREIGN KEY (MaMH) REFERENCES Mat\_Hang(MaMH),  FOREIGN KEY (MaBD) REFERENCES Bao\_Duong(MaBD),  FOREIGN KEY (MaNVBaoDuong) REFERENCES Nhan\_Vien(MaNV)  ) |

***Bảng Phiếu nhập***

|  |
| --- |
| CREATE TABLE Phieu\_Nhap  (  MaPN nvarchar(11) PRIMARY KEY,  NgayNhap date check (DATEDIFF(day, NgayNhap, GETDATE())>=0),  ThanhTien money default 0,  MaNVKho nvarchar(11),  MaNCC nvarchar(11) ,  FOREIGN KEY (MaNVKho) REFERENCES Nhan\_Vien(MaNV),  FOREIGN KEY (MaNCC) REFERENCES Nha\_Cung\_Cap(MaNCC)  ) |

***Bảng Chi tiết phiếu nhập***

|  |
| --- |
| CREATE TABLE Chi\_Tiet\_Phieu\_Nhap  (  MaPN nvarchar(11),  MaMH nvarchar(11) ,  SoLuong int check (SoLuong>=0) default 0,  PRIMARY KEY (MaPN, MaMH),  FOREIGN KEY (MaPN) REFERENCES Phieu\_Nhap(MaPN) ON DELETE CASCADE,  FOREIGN KEY (MaMH) REFERENCES Mat\_Hang(MaMH)  ) |

***Bảng Kho***

|  |
| --- |
| CREATE TABLE Kho  (  MaCN nvarchar(11),  MaMH nvarchar(11),  SoLuong int check (SoLuong>=0) default 0,  PRIMARY KEY (MaCN, MaMH),  FOREIGN KEY (MaCN) REFERENCES Chi\_Nhanh(MaCN),  FOREIGN KEY (MaMH) REFERENCES Mat\_Hang(MaMH) ON DELETE CASCADE  ) |

## **5. Các ràng buộc khác**

***Ràng buộc người lập phiếu là nhân viên quản lý kho***

|  |
| --- |
| --ràng buộc người lập phiếu nhập phải là nhân viên quản lý kho  ALTER TABLE Phieu\_Nhap  ADD CONSTRAINT constr\_chk\_NhanVienKhoKhang  CHECK (dbo.fn\_CheckChucVu(N'Nhân viên quản lý kho',MaNVKho) = 1)  GO |

***Ràng buộc người lập hóa đơn là nhân viên thu ngân***

|  |
| --- |
| --ràng buộc người lập hoá đơn phải là thu ngân  ALTER TABLE Hoa\_Don  ADD CONSTRAINT constr\_chk\_NhanVienThuNgan  CHECK (dbo.fn\_CheckChucVu(N'Nhân viên thu ngân',MaNVThuNgan)=1)  GO |

***Ràng buộc mỗi chi nhánh chỉ có một nhân viên quản lý***

|  |
| --- |
| -- Ràng buộc mỗi chi nhánh chỉ có một quản lý  ALTER TABLE Nhan\_Vien  add CONSTRAINT constr\_chk\_QuanLyChiNhanh  CHECK (dbo.fn\_CheckSoLuongQuanLyChiNhanh()=1) -- check xem coi thỏa hay kh  GO |

***Ràng buộc người thực hiện bảo dưỡng là nhân viên bảo dưỡng***

|  |
| --- |
| -- Ràng buộc nhân viên thực hiện bảo dưỡng phải là nhân viên bảo dưỡng hoặc là null nếu chi tiết này chứa mặt hàng  ALTER TABLE Chi\_Tiet\_Hoa\_Don  ADD CONSTRAINT constr\_chk\_NhanVienBaoDuong  CHECK (MaNVBaoDuong is null or dbo.fn\_CheckChucVu(N'Nhân viên bảo dưỡng', MaNVBaoDuong)=1)  GO |

***Ràng buộc mã khách hàng khi thêm khách hàng***

|  |
| --- |
| -- mỗi khi thêm khách hàng thì mã sẽ được tạo tự động  -- Đặt ràng buộc mã khách hàng được tạo tự động.  ALTER TABLE Khach\_Hang  ADD CONSTRAINT constr\_MaKH  Default dbo.fn\_MaKhachHangTuDong() for MaKH  GO |

***Ràng buộc mã hóa đơn khi thêm hóa đơn***

|  |
| --- |
| --ràng buộc mặc định cho cột "MaHD" trong bảng "Hoa\_Don".  --Mỗi khi thêm hóa đơn mã hóa đơn sẽ được tạo tự động  ALTER TABLE Hoa\_Don  ADD CONSTRAINT constr\_MaHD  DEFAULT [dbo].[fn\_MaHoaDonTuDong]() FOR MaHD;  GO |

***Ràng buộc mã nhân viên khi thêm nhân viên***

|  |
| --- |
| --Mỗi khi thêm nhân viên mã nhân viên sẽ được tạo tự động  ALTER TABLE Nhan\_Vien  ADD CONSTRAINT DF\_Nhan\_Vien\_MaNV DEFAULT ([dbo].[fn\_MaNhanVienTuDong]()) FOR MaNV;  GO |

***Ràng buộc mã phiếu nhập khi thêm phiếu nhập***

|  |
| --- |
| -- Mỗi khi thêm phiếu nhập mã phiếu nhập sẽ được tạo tự động ALTER TABLE Phieu\_Nhap  ADD CONSTRAINT DF\_Phieu\_Nhap\_MaPN DEFAULT ([dbo].[fn\_MaPhieuNhapTuDong]()) FOR MaPN;  GO |

## ***6. Trigger***

***Trigger tính tiền hóa đơn, kiểm tra và điều chỉnh kho nếu mặt hàng đã được bán***

|  |
| --- |
| CREATE TRIGGER trg\_CapNhatThanhTienHoaDon ON Chi\_Tiet\_Hoa\_Don  AFTER INSERT, DELETE, UPDATE AS  BEGIN  --tạo biến để lấy giá trị của hoá đơn vừa được sửa, và 2 biến để biết số tiền tăng hay giảm  DECLARE @MaHDUpdate NVARCHAR(11), @MaCN NVARCHAR(11),@TienTang MONEY, @TienGiam MONEY , @MaMHBiXoa NVARCHAR(11), @MaMHDuocThem NVARCHAR(11), @SoluongBiXoa INT, @SoluongDuocThem INT;  --nếu bảng deleted tồn tại thì tính tiền giảm  IF EXISTS(SELECT 1 FROM deleted)  BEGIN  IF ((SELECT MaMH FROM DELETED)IS NOT NULL) -- nếu trong chi tiết hoá đơn tồn tại mã mặt hàng  BEGIN -- tính tổng tiền mặt hàng trong bảng deleted và tại đây lấy luôn mã hoá đơn có sự chỉnh sửa  SELECT @TienGiam = deleted.SoLuong \* Mat\_Hang.DonGia, @MaHDUpdate = deleted.MaHD, @MaMHBiXoa=Mat\_Hang.MaMH, @SoluongBiXoa=SoLuong  FROM deleted JOIN Mat\_Hang on deleted.MaMH = Mat\_Hang.MaMH  END  ELSE -- nếu trong chi tiết hoá đơn tồn tại mã bảo dưỡng  BEGIN -- tính tổng tiền bảo dưỡng trong bảng deleted  SELECT @TienGiam = deleted.SoLuong \* Bao\_Duong.PhiBD, @MaHDUpdate = deleted.MaHD -- tính tổng tiền của bảo dưỡng đó  FROM deleted JOIN Bao\_Duong on deleted.MaBD = Bao\_Duong.MaBD    END  END  ELSE -- nếu không tồn tại bảng deleted  BEGIN  SET @TienGiam = 0 -- set cho tiền giảm bằng 0  SET @SoluongBiXoa=0--xet so luong bị xoá là 0  SELECT @MaHDUpdate = inserted.MaHD FROM inserted -- do bảng deleted không tồn tại nên chắc ăn đây là trường hợp insert  END  --nếu bảng inserted tồn tại thì tính tiền tăng  IF EXISTS(SELECT 1 FROM inserted)  BEGIN  IF ((SELECT MaMH FROM inserted)IS NOT NULL) -- nếu trong chi tiết hoá đơn tồn tại mã mặt hàng  BEGIN -- tính tổng tiền mặt hàng trong bảng inserted  SELECT @TienTang = inserted.SoLuong \* Mat\_Hang.DonGia, @MaMHDuocThem=Mat\_Hang.MaMH,@SoluongDuocThem=SoLuong -- tính tổng tiền mặt hàng trong đó  FROM inserted JOIN Mat\_Hang on inserted.MaMH = Mat\_Hang.MaMH  END  ELSE -- nếu trong chi tiết hoá đơn tồn tại mã bảo dưỡng  BEGIN -- tính tổng tiền bảo dưỡng trong bảng inserted  SELECT @TienTang = inserted.SoLuong \* Bao\_Duong.PhiBD -- tính tổng tiền của bảo dưỡng đó  FROM inserted JOIN Bao\_Duong on inserted.MaBD = Bao\_Duong.MaBD  END  END  ELSE -- nếu không tồn tại bảng inserted  BEGIN  SET @TienTang = 0 -- set cho tiền giảm bằng 0  SET @SoluongDuocThem=0-- set so luong tăng là 0  END  SELECT @MaCN=MaCN -- lấy mã chi nhánh của hoá đơn bị tác động để cập nhật kho  FROM Hoa\_Don JOIN Nhan\_Vien on MaNVThuNgan=MaNV  WHERE MaHD=@MaHDUpdate  BEGIN TRY-- dùng try catch bắt lại vì có thể có trường hợp sau khi thêm hay sửa chi tiết hóa đơn làm cho kho bị âm    UPDATE Kho  SET SoLuong = SoLuong + @SoluongBiXoa  WHERE MaMH = @MaMHBiXoa and @MaMHBiXoa is not null AND Kho.MaCN=@MaCN      UPDATE Kho  SET SoLuong = SoLuong - @SoluongDuocThem  WHERE MaMH = @MaMHDuocThem and @MaMHDuocThem is not null AND Kho.MaCN=@MaCN    UPDATE Hoa\_Don-- cập nhật lại tiền trong hóa đơn  SET ThanhTien = ThanhTien + @TienTang - @TienGiam WHERE MaHD=@MaHDUpdate      END TRY  BEGIN CATCH  RAISERROR(N'Kho Không ?? S? L??ng', 16, 1)  END CATCH;  END;  GO |

***Trigger thêm xoá sửa 1 chi tiết phiếu nhập thì thành tiền trong phiếu nhập và số lượng trong kho được cập nhật***

|  |
| --- |
| -- tạo trigger để mỗi khi thêm xoá sửa 1 chi tiết phiếu nhập thì thành tiền trong phiếu nhập và số lượng trong kho đc cập nhật  CREATE TRIGGER trg\_CapNhatThanhTienPhieuNhap ON Chi\_Tiet\_Phieu\_Nhap  AFTER INSERT AS  BEGIN  --tạo biến để lấy giá trị của phiếu nhập vừa được sửa, và 2 biến để biết số tiền tăng hay giảm , 2 biến chứa số lượng tăng giảm và mã chi nhánh để chỉnh sửa trong kho  DECLARE @MaPNUpdate NVARCHAR(11), @TienTang MONEY, @MaCN NVARCHAR(11), @MaMHDuocThem NVARCHAR(11), @Soluong INT;    SELECT @MaPNUpdate=MaPN-- lấy mã phiếu nhập được thêm  FROM inserted  SELECT @TienTang = inserted.SoLuong \* Mat\_Hang.GiaNhap, @MaMHDuocThem=Mat\_Hang.MaMH,@Soluong=SoLuong --lấy số tiền tăng lên và mặt hàng được thêm vào số lượng lun  FROM inserted JOIN Mat\_Hang on inserted.MaMH = Mat\_Hang.MaMH    UPDATE Phieu\_Nhap -- tính lại tiền của phiếu nhập đó  SET ThanhTien = ThanhTien + @TienTang WHERE MaPN=@MaPNUpdate    SELECT @MaCN=MaCN -- lấy mã chi nhánh của phiếu nhập bị tác động để cập nhật kho  FROM Phieu\_Nhap JOIN Nhan\_Vien on MaNVKho=MaNV  WHERE MaPN=@MaPNUpdate    UPDATE Kho-- cập nhật số lượng trong kho  SET SoLuong+=@Soluong  Where MaCN=@MaCN and @MaMHDuocThem is not null and MaMH=@MaMHDuocThem  END;  GO |

***Trigger thêm nhân viên***

|  |
| --- |
| CREATE TRIGGER trg\_ThemNhanVien ON Nhan\_Vien  AFTER INSERT AS  BEGIN  DECLARE @MaNV NVARCHAR(11) SELECT @MaNV = INSERTED.MaNV FROM INSERTED  DECLARE @Mk NVARCHAR(11)  SET @Mk = LEFT(CONVERT(VARCHAR(255), NEWID()), 6)  DECLARE @ChucVu NVARCHAR(50) SELECT @ChucVu = INSERTED.ChucVu FROM INSERTED  --Mỗi khi thêm hay sửa nhân viên kiểm tra xem phải quản lý hay không  IF (@ChucVu) = N'Nhân viên quản lý'  BEGIN  --Nếu nhân viên được thêm là quản lý thì qua bảng chi nhánh cập nhật  DECLARE @MaCN NVARCHAR(11)  SELECT @MaCN=MaCN FROM INSERTED  UPDATE Chi\_Nhanh SET MaNQL=@MaNV WHERE MaCN=@MaCN  END  INSERT INTO Tai\_Khoan VALUES (@MaNV,@MaNV,@Mk,@ChucVu)  END  GO |

***Trigger xóa nhân viên***

|  |
| --- |
| CREATE TRIGGER trg\_XoaNhanVien ON Nhan\_Vien  INSTEAD OF DELETE AS -- sài instead of để kh cho xóa bảng ghi nhân viên đó mà chỉ set thuộc tính đi làm bằng 0  BEGIN  UPDATE Nhan\_Vien  SET TinhTrangLamViec=0  WHERE MaNV = (SELECT MaNV  FROM deleted)  IF (SELECT DELETED.ChucVu FROM deleted) = N'Nhân viên quản lý' -- nếu quản lý bị xóa thì sẽ cập nhật quản lý ở cửa hàng này là null  BEGIN  DECLARE @MaCN NVARCHAR(11)  SELECT @MaCN=MaCN FROM deleted  UPDATE Chi\_Nhanh SET MaNQL = null WHERE MaCN=@MaCN  END  END;  go |

***Trigger sửa nhân viên***

|  |
| --- |
| CREATE TRIGGER trg\_SuaNhanVien ON Nhan\_Vien--Mỗi khi thêm hay sửa nhân viên kiểm tra xem phải quản lý hay không  AFTER UPDATE AS  BEGIN  DECLARE @MaNV NVARCHAR(11) SELECT @MaNV=INSERTED.MaNV FROM INSERTED  DECLARE @ChucVu NVARCHAR(50) SELECT @ChucVu=INSERTED.ChucVu FROM INSERTED  UPDATE Tai\_Khoan SET ChucVu = (@ChucVu) WHERE MaNV = (@MaNV)  IF (SELECT DELETED.ChucVu FROM DELETED) = N'Nhân viên quản lý' -- nếu quản lý bị xóa thì sẽ cập nhật quản lý ở cửa hàng này là null  BEGIN  DECLARE @MaCN NVARCHAR(11)  SELECT @MaCN=MaCN FROM DELETED  UPDATE Chi\_Nhanh SET MaNQL = null WHERE MaCN=@MaCN  END  IF (SELECT INSERTED.ChucVu FROM INSERTED) = N'Nhân viên quản lý'-- nếu nhân viên được thêm là quản lý thì qua bảng chi nhánh cập nhật  BEGIN  SELECT @MaCN=MaCN FROM INSERTED  UPDATE Chi\_Nhanh SET MaNQL=@MaNV WHERE MaCN=@MaCN  END  END  GO |

## ***7. Các view***

***View xem Danh sách bảo dưỡng***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSBaoDuong]  AS  SELECT \*  FROM Bao\_Duong  GO |

***View xem Danh sách chi nhánh***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSChiNhanh]  AS  SELECT Chi\_Nhanh.MaCN,Chi\_Nhanh.TenCN,Chi\_Nhanh.DiaChi FROM Chi\_Nhanh  GO |

***View xem Danh sách hóa đơn***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSHoaDon]  AS  SELECT Hoa\_Don.MaHD,Khach\_Hang.MaKH,Khach\_Hang.TenKH, Khach\_Hang.SDT,Khach\_Hang.CCCD,Khach\_Hang.DiaChi, Nhan\_Vien.TenNV, Nhan\_Vien.MaNV, Hoa\_Don.NgayLap, Hoa\_Don.ThanhTien,Chi\_Nhanh.TenCN  FROM Hoa\_Don  INNER JOIN Khach\_Hang ON Hoa\_Don.MaKH = Khach\_Hang.MaKH  INNER JOIN Nhan\_Vien ON Hoa\_Don.MaNVThuNgan = Nhan\_Vien.MaNV  INNER JOIN Chi\_Nhanh ON Nhan\_Vien.MaCN=Chi\_Nhanh.MaCN  GO |

***View xem Danh sách khách hàng***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSKhachHang]  AS  SELECT \*  FROM Khach\_Hang  GO |

***View xem Danh sách nhà cung cấp***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSNhaCungCap]  AS  SELECT Nha\_Cung\_Cap.MaNCC  ,Nha\_Cung\_Cap.TenNCC, Nha\_Cung\_Cap.DiaChi, Nha\_Cung\_Cap.SDT FROM Nha\_Cung\_Cap  GO |

***View xem Danh sách nhân viên***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSNhanVien]  AS  SELECT MaNV, TenNV, NgaySinh, Nhan\_Vien.DiaChi, SDT, GioiTinh, CCCD, ChucVu,Chi\_Nhanh.TenCN, Luong  FROM Nhan\_Vien, Chi\_Nhanh  WHERE Chi\_Nhanh.MaCN=Nhan\_Vien.MaCN AND Nhan\_Vien.TinhTrangLamViec=1;  GO |

***View xem Danh sách phiếu nhập***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSPhieuNhap]  AS  SELECT Phieu\_Nhap.MaPN, Nha\_Cung\_Cap.MaNCC, Nha\_Cung\_Cap.TenNCC, Nha\_Cung\_Cap.SDT,Nha\_Cung\_Cap.DiaChi, Nhan\_Vien.TenNV, Nhan\_Vien.MaNV,Phieu\_Nhap.NgayNhap, Phieu\_Nhap.ThanhTien , Chi\_Nhanh.TenCN    FROM Phieu\_Nhap  INNER JOIN Nha\_Cung\_Cap ON Phieu\_Nhap.MaNCC = Nha\_Cung\_Cap.MaNCC  INNER JOIN Nhan\_Vien ON Phieu\_Nhap.MaNVKho = Nhan\_Vien.MaNV  INNER JOIN Chi\_Nhanh ON Nhan\_Vien.MaCN=Chi\_Nhanh.MaCN  GO |

***View xem Danh sách Phụ Tùng***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSPhuTung]  AS  SELECT MaMH,Mat\_Hang.HinhAnh,Mat\_Hang.TenMH, Mat\_Hang.HSX, Mat\_Hang.DonGia,Mat\_Hang.GiaNhap,Phu\_Tung.LoaiPT,Phu\_Tung.ChatLuong FROM Phu\_Tung JOIN Mat\_Hang ON MaPT = MaMH  GO |

***View xem Danh sách Xe***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSNhaCungCap]  AS  SELECT Nha\_Cung\_Cap.MaNCC  ,Nha\_Cung\_Cap.TenNCC, Nha\_Cung\_Cap.DiaChi, Nha\_Cung\_Cap.SDT FROM Nha\_Cung\_Cap  GO |

***View xem Danh sách nhà cung cấp***

|  |
| --- |
| CREATE VIEW [dbo].[vw\_DSXe]  AS  SELECT Mat\_Hang.MaMH,Mat\_Hang.HinhAnh,Mat\_Hang.TenMH, Mat\_Hang.HSX, Mat\_Hang.DonGia, Mat\_Hang.GiaNhap, XE.LoaiXe, XE.MauSac, XE.KhoiLuongBanThan, XE.DxRxC, XE.KhoangCachTrucBanhXe, XE.DoCaoYen, XE.KhoangSangGamXe, XE.DungTichBinhXang, XE.KichCoLopTruoc, XE.KichCoLopSau, XE.PhuocTruoc, XE.PhuocSau, XE.LoaiDongCo, XE.CongSuatToiDa, XE.DungTichNhotMay, XE.LoaiTruyenDong, XE.HeThongKhoiDong, XE.MomentCucDai, XE.DungTichXiLanh, XE.DuongKinhXHanhTrinhPitTong, XE.TySoNen, XE.DongCoCongNghe  FROM XE JOIN Mat\_Hang ON MaXe = Mat\_Hang.MaMH  GO |

# **CHƯƠNG 3. THIẾT KẾ CÁC CHỨC NĂNG**

1. **Kết nối cơ sở dữ liệu**

|  |
| --- |
| internal class MyDB  {  SqlConnection Connectionconnection = DangNhapForm.conn;  public SqlConnection Connection { get => Connectionconnection; }  public void Close()  {  try  {  if (Connectionconnection != null && Connectionconnection.State == ConnectionState.Open)  {  Connectionconnection.Close();  }  }  catch (Exception){}  } |

1. **Quản lý nhân viên**

***Hàm sinh mã nhân viên tự động***

|  |
| --- |
| CREATE FUNCTION fn\_MaNhanVienTuDong()-- hàm sinh mã nhân viên tự động  RETURNS NVARCHAR(11)  AS  BEGIN  DECLARE @NextID NVARCHAR(11), @LastID INT  SELECT @LastID = ISNULL(MAX(CAST(SUBSTRING(MaNV, 3,LEN(MaNV)-2) AS INT)),0 )FROM Nhan\_Vien -- lấy số nhân viên có mã số cao nhất  IF @LastID <= 99999  SET @NextID = 'NV' + RIGHT('0000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 5)  ELSE  SET @NextID = 'NV' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  RETURN @NextID  END  go |

***Mỗi khi thêm hóa đơn, mã sẽ được tạo tự động***

|  |
| --- |
| ALTER TABLE Nhan\_Vien -- mỗi khi thêm hoá đơn mã sẽ được tạo tự động  ADD CONSTRAINT constr\_MaNV  Default dbo.fn\_MaNhanVienTuDong() for MaNV  go |

***Thêm nhân viên vào bảng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemNhanVien](  @TenNV nvarchar(40),  @NgaySinh date,  @DiaChi nvarchar(75),  @SDT nvarchar(11),  @GioiTinh nvarchar(4),  @CCCD nvarchar(13),  @ChucVu nvarchar(50),  @TenCN nvarchar(50),  @Luong MONEY)  AS  BEGIN  BEGIN TRY  -- Lấy mã chi nhánh tương ứng với tên chi nhánh được truyền vào  DECLARE @MaCN NVARCHAR(11)  SELECT @MaCN = MaCN FROM Chi\_Nhanh WHERE TenCN = @TenCN;  -- Thêm nhân viên mới vào bảng Nhan\_Vien  INSERT INTO Nhan\_Vien (TenNV, NgaySinh, DiaChi, SDT, GioiTinh, CCCD, ChucVu, MaCN, Luong)  VALUES (@TenNV, @NgaySinh, @DiaChi, @SDT, @GioiTinh, @CCCD, @ChucVu, @MaCN, @Luong);  END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_SDT\_\_2A164134%'  RAISERROR(N'SĐT phải có 10 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_CCCD\_\_75A278F5%'  RAISERROR(N'CCCD phải có 12 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_Nhan\_Vie\_\_A955A0AAB43DACE3%'  RAISERROR(N'CCCD bị trùng với nhân viên khác', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_Nhan\_Vie\_\_CA1930A59E569782%'  RAISERROR(N'SĐT bị trùng với nhân viên khác', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_SDT\_\_2EDAF651%'  RAISERROR(N'SĐT chỉ được phép chứa các chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_CCCD\_\_2FCF1A8A%'  RAISERROR(N'CCCD chỉ được phép chứa các chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%constr\_chk\_QuanLyChiNhanh%'  RAISERROR(N'Một chi nhánh không thể có tận 2 quản lý', 16, 1)  END CATCH  END;  GO |

***Sửa thông tin nhân viên***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaNhanVien](  @MaNV nvarchar(11),  @TenNV nvarchar(40),  @NgaySinh date,  @DiaChi nvarchar(75),  @SDT nvarchar(11),  @GioiTinh nvarchar(4),  @CCCD nvarchar(13),  @ChucVu nvarchar(50),  @TenCN nvarchar(50),  @Luong MONEY  )  AS  BEGIN  BEGIN TRY  DECLARE @MaCN nvarchar(11);  SELECT @MaCN = MaCN FROM Chi\_Nhanh WHERE TenCN = @TenCN;  UPDATE Nhan\_Vien  SET TenNV = @TenNV, NgaySinh = @NgaySinh,DiaChi = @DiaChi,SDT = @SDT, GioiTinh = @GioiTinh,CCCD = @CCCD, ChucVu = @ChucVu,MaCN=@MaCN, Luong = @Luong  WHERE MaNV = @MaNV  END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_SDT\_\_2A164134%'  RAISERROR(N'SĐT phải có 10 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_CCCD\_\_75A278F5%'  RAISERROR(N'CCCD phải có 12 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_Nhan\_Vie\_\_A955A0AAB43DACE3%'  RAISERROR(N'CCCD bị trùng với nhân viên khác', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_Nhan\_Vie\_\_CA1930A59E569782%'  RAISERROR(N'SĐT bị trùng với nhân viên khác', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_SDT\_\_2EDAF651%'  RAISERROR(N'SĐT chỉ được phép chứa các chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Nhan\_Vien\_\_CCCD\_\_2FCF1A8A%'  RAISERROR(N'CCCD chỉ được phép chứa các chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%constr\_chk\_QuanLyChiNhanh%'  RAISERROR(N'Một chi nhánh không thể có tận 2 quản lý', 16, 1)  END CATCH  END  GO |

***Kiểm tra xem nhân viên đó còn làm việc không***

|  |
| --- |
| CREATE FUNCTION dbo.fn\_CheckChucVu(  @ChucVu NVARCHAR(50),  @MaNV NVARCHAR(11)  )  RETURNS INT  AS  BEGIN  DECLARE @Result INT  SELECT @Result = CASE WHEN EXISTS (  SELECT 1  FROM Nhan\_Vien  WHERE ChucVu = @ChucVu AND MaNV = @MaNV AND TinhTrangLamViec = 1  ) THEN 1 ELSE 0 END  RETURN @Result  END  GO |

***Kiểm tra một chi nhánh chỉ có một nhân viên quản lý***

|  |
| --- |
| CREATE FUNCTION fn\_CheckSoLuongQuanLyChiNhanh()  RETURNS INT  AS  BEGIN  IF EXISTS (  SELECT 1  FROM Nhan\_Vien  JOIN Chi\_Nhanh ON Nhan\_Vien.MaCN = Chi\_Nhanh.MaCN  WHERE Nhan\_Vien.ChucVu = N'Nhân viên quản lý' AND Nhan\_Vien.TinhTrangLamViec = 1  GROUP BY Chi\_Nhanh.MaCN  HAVING COUNT(\*) > 1  )  RETURN 0  RETURN 1  END;  GO |

***Lọc và truy xuất những thông tin trong Nhân viên***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_DSNhanVien]  (  @TenCN NVARCHAR(40),  @ChucVu NVARCHAR(50),  @TenNV NVARCHAR(40),  @MaNV NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT \*  FROM vw\_DSNhanVien  WHERE TenCN = CASE WHEN @TenCN = N'' THEN TenCN ELSE @TenCN END  AND ChucVu = CASE WHEN @ChucVu = N'' THEN ChucVu ELSE @ChucVu END  AND TenNV = CASE WHEN @TenNV = N'' THEN TenNV ELSE @TenNV END  AND MaNV = CASE WHEN @MaNV = N'' THEN MaNV ELSE @MaNV END  )  GO |

***Lấy thông tin nhân viên***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_LayThongTinNhanVien]  (  @MaNV NVARCHAR(11)  )  -- WITH ENCRYPTION, SCHEMABINDING, EXECUTE AS CALLER|SELF|OWNER|USER  RETURNS TABLE  AS RETURN  (  SELECT MaNV, TenNV, NgaySinh, Nhan\_Vien.DiaChi, SDT, GioiTinh, CCCD, ChucVu,Chi\_Nhanh.TenCN, Luong  FROM Nhan\_Vien, Chi\_Nhanh  WHERE Chi\_Nhanh.MaCN=Nhan\_Vien.MaCN AND Nhan\_Vien.TinhTrangLamViec=1 AND Nhan\_Vien.MaNV = @MaNV  )  GO |

1. **Quản lý khách hàng**

***Hàm sinh mã khách hàng tự động***

|  |
| --- |
| CREATE FUNCTION fn\_MaKhachHangTuDong()-- hàm sinh mã khách hàng tự động  RETURNS NVARCHAR(11)  AS  BEGIN  DECLARE @NextID NVARCHAR(11), @LastID INT  SELECT @LastID = ISNULL(MAX(CAST(SUBSTRING(MaKH, 3,LEN(MaKH)-2) AS INT)),0) FROM Khach\_Hang -- lấy số của khách hàng mới nhất  IF @LastID <= 999999  SET @NextID = 'KH' + RIGHT('00000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 6)  ELSE  SET @NextID = 'KH' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  RETURN @NextID  END  go |

***Procedure thêm thông tin khách hàng***

|  |
| --- |
| create PROCEDURE [dbo].[sp\_ThemKhachHang](  @TenKH nvarchar(40),  @SDT nvarchar(11),  @GioiTinh NVARCHAR(5),  @CCCD nvarchar(13),  @DiaChi nvarchar(75)  )  AS  BEGIN  BEGIN TRY  INSERT INTO Khach\_Hang (TenKH, SDT,GioiTinh, CCCD, DiaChi)  VALUES (@TenKH, @SDT,@GioiTinh, @CCCD, @DiaChi)  END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%UQ\_\_tmp\_ms\_x\_\_A955A0AA8907BFB4%'  RAISERROR(N'Trùng CCCD với 1 khách hàng khác', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_tmp\_ms\_x\_\_CA1930A590397736%'  RAISERROR(N'Trùng số điện thoại với 1 khách hàng khác', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_SDT\_\_0C50D423%'  RAISERROR(N'SĐT phải có 10 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_CCCD\_\_0D44F85C%'  RAISERROR(N'CCCD phải có 12 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_CCCD\_\_0F2D40CE%'  RAISERROR(N'CCCD chỉ được phép có số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_SDT\_\_0E391C95%'  RAISERROR(N'SĐT chỉ được phép có số', 16, 1)  END CATCH  END  GO |

***Produce sửa thông tin khách hàng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaKhachHang]  (  @MaKH nvarchar(11),  @TenKH nvarchar(40),  @SDT nvarchar(11),  @GioiTinh NVARCHAR(5),  @CCCD nvarchar(13),  @DiaChi nvarchar(75)  )  AS  BEGIN  BEGIN TRY  UPDATE Khach\_Hang SET TenKH = @TenKH,SDT = @SDT,GioiTinh=@GioiTinh,CCCD = @CCCD, DiaChi = @DiaChi WHERE MaKH = @MaKH  END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%UQ\_\_tmp\_ms\_x\_\_A955A0AA8907BFB4%'  RAISERROR(N'Trùng CCCD với 1 khách hàng khác', 16, 1)  ELSE IF @errorMessage LIKE '%UQ\_\_tmp\_ms\_x\_\_CA1930A590397736%'  RAISERROR(N'Trùng số điện thoại với 1 khách hàng khác', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_SDT\_\_0C50D423%'  RAISERROR(N'SĐT phải có 10 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_CCCD\_\_0D44F85C%'  RAISERROR(N'CCCD phải có 12 chữ số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_CCCD\_\_0F2D40CE%'  RAISERROR(N'CCCD chỉ được phép có số', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Khach\_Hang\_\_SDT\_\_0E391C95%'  RAISERROR(N'SĐT chỉ được phép có số', 16, 1)  END CATCH  END  GO |

***Tìm và truy xuất thông tin khách hàng theo CCCD***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_TimKhachHangTheoCCCD]  (  @CCCD nvarchar(13)  )  RETURNS TABLE  AS  RETURN  (  SELECT \* FROM vw\_DSKhachHang  WHERE CCCD = @CCCD  )  GO |

***Tìm và truy xuất thông tin khách hàng theo SDT***

|  |
| --- |
| create FUNCTION [dbo].[fn\_TimKhachHangTheoSĐT]  (  @SDT nvarchar(11)  )  RETURNS TABLE  AS  RETURN  (  SELECT \* FROM vw\_DSKhachHang  WHERE SDT = @SDT  )  GO |

1. **Quản lý xe máy**

***Thêm thông tin xe***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemXe](  @HSX nvarchar(30),  @TenMH nvarchar(40),  @DonGia money,  @GiaNhap money,  @HinhAnh IMAGE,  @LoaiXe nvarchar(MAX),  @MauSac nvarchar(MAX),  @KhoiLuongBanThan nvarchar(max),  @DxRxC nvarchar(max),  @KhoangCachTrucBanhXe nvarchar(max),  @DoCaoYen nvarchar(max),  @KhoangSangGamXe nvarchar(max),  @DungTichBinhXang nvarchar(max),  @KichCoLopTruoc nvarchar(max),  @KichCoLopSau nvarchar(max),  @PhuocTruoc nvarchar(max),  @PhuocSau nvarchar(max),  @LoaiDongCo nvarchar(max),  @CongSuatToiDa nvarchar(max),  @DungTichNhotMay nvarchar(max),  @LoaiTruyenDong nvarchar(max),  @HeThongKhoiDong nvarchar(max),  @MomentCucDai nvarchar(max),  @DungTichXiLanh nvarchar(max),  @DuongKinhXHanhTrinhPitTong nvarchar(max),  @TySoNen nvarchar(max),  @DongCoCongNghe nvarchar(max)  )  AS  BEGIN -- tạo biến để lưu mã mặt hàng  DECLARE @MaMH NVARCHAR(11), @LastID INT ;  IF EXISTS (SELECT 1 FROM XE JOIN Mat\_Hang ON MaMH = MaXe WHERE HSX = @HSX AND TenMH = @TenMH AND LoaiXe = @LoaiXe AND MauSac = @MauSac AND KhoiLuongBanThan = @KhoiLuongBanThan AND DxRxC = @DxRxC  AND KhoangCachTrucBanhXe = @KhoangCachTrucBanhXe AND DoCaoYen = @DoCaoYen AND KhoangSangGamXe = @KhoangSangGamXe AND DungTichBinhXang = @DungTichBinhXang AND KichCoLopTruoc = @KichCoLopTruoc  AND KichCoLopSau = @KichCoLopSau AND PhuocTruoc = @PhuocTruoc AND PhuocSau = @PhuocSau AND LoaiDongCo = @LoaiDongCo AND CongSuatToiDa = @CongSuatToiDa AND DungTichNhotMay = @DungTichNhotMay  AND LoaiTruyenDong = @LoaiTruyenDong AND HeThongKhoiDong = @HeThongKhoiDong AND MomentCucDai = @MomentCucDai AND DungTichXiLanh = @DungTichXiLanh AND DuongKinhXHanhTrinhPitTong = @DuongKinhXHanhTrinhPitTong  AND TySoNen = @TySoNen AND DongCoCongNghe = @DongCoCongNghe  )--check nếu xe chưa tồn tại thì cho thêm  BEGIN  RAISERROR (N'Thông tin xe đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  ELSE  BEGIN  SET @LastID = (SELECT ISNULL(MAX(CAST(SUBSTRING(MaXe, 3,LEN(MaXe)-1) AS INT)),0) FROM Xe)  IF @LastID <= 99999  SET @MaMH = 'XE' + RIGHT('0000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 5)  ELSE  SET @MaMH = 'XE' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  INSERT INTO Mat\_Hang VALUES (@MaMH, @HSX, @TenMH, @DonGia, @GiaNhap, @HinhAnh)  INSERT INTO Xe VALUES (@MaMH,@LoaiXe, @MauSac, @KhoiLuongBanThan, @DxRxC, @KhoangCachTrucBanhXe, @DoCaoYen, @KhoangSangGamXe, @DungTichBinhXang, @KichCoLopTruoc, @KichCoLopSau, @PhuocTruoc, @PhuocSau, @LoaiDongCo, @CongSuatToiDa, @DungTichNhotMay, @LoaiTruyenDong, @HeThongKhoiDong, @MomentCucDai, @DungTichXiLanh, @DuongKinhXHanhTrinhPitTong, @TySoNen, @DongCoCongNghe)    INSERT INTO Kho (MaCN, MaMH, SoLuong)  SELECT Chi\_Nhanh.MaCN, @MaMH, 0  FROM Chi\_Nhanh ;  END  END;  GO |

***Sửa thông tin xe***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaXe](  @MaMH nvarchar(30),  @HSX nvarchar(30),  @TenMH nvarchar(40),  @DonGia money,  @GiaNhap money,  @HinhAnh IMAGE,  @LoaiXe nvarchar(MAX),  @MauSac nvarchar(MAX),  @KhoiLuongBanThan nvarchar(max),  @DxRxC nvarchar(max),  @KhoangCachTrucBanhXe nvarchar(max),  @DoCaoYen nvarchar(max),  @KhoangSangGamXe nvarchar(max),  @DungTichBinhXang nvarchar(max),  @KichCoLopTruoc nvarchar(max),  @KichCoLopSau nvarchar(max),  @PhuocTruoc nvarchar(max),  @PhuocSau nvarchar(max),  @LoaiDongCo nvarchar(max),  @CongSuatToiDa nvarchar(max),  @DungTichNhotMay nvarchar(max),  @LoaiTruyenDong nvarchar(max),  @HeThongKhoiDong nvarchar(max),  @MomentCucDai nvarchar(max),  @DungTichXiLanh nvarchar(max),  @DuongKinhXHanhTrinhPitTong nvarchar(max),  @TySoNen nvarchar(max),  @DongCoCongNghe nvarchar(max)  )  AS  BEGIN  IF EXISTS (SELECT 1 FROM XE JOIN Mat\_Hang ON MaMH = MaXe WHERE HSX = @HSX AND TenMH = @TenMH AND LoaiXe = @LoaiXe AND MauSac = @MauSac AND KhoiLuongBanThan = @KhoiLuongBanThan AND DxRxC = @DxRxC  AND KhoangCachTrucBanhXe = @KhoangCachTrucBanhXe AND DoCaoYen = @DoCaoYen AND KhoangSangGamXe = @KhoangSangGamXe AND DungTichBinhXang = @DungTichBinhXang AND KichCoLopTruoc = @KichCoLopTruoc  AND KichCoLopSau = @KichCoLopSau AND PhuocTruoc = @PhuocTruoc AND PhuocSau = @PhuocSau AND LoaiDongCo = @LoaiDongCo AND CongSuatToiDa = @CongSuatToiDa AND DungTichNhotMay = @DungTichNhotMay  AND LoaiTruyenDong = @LoaiTruyenDong AND HeThongKhoiDong = @HeThongKhoiDong AND MomentCucDai = @MomentCucDai AND DungTichXiLanh = @DungTichXiLanh AND DuongKinhXHanhTrinhPitTong = @DuongKinhXHanhTrinhPitTong  AND TySoNen = @TySoNen AND DongCoCongNghe = @DongCoCongNghe  )--check nếu xe nào giống thông số xe được sửa không  BEGIN  RAISERROR (N'Thông tin xe đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  UPDATE Mat\_Hang SET HSX = @HSX,TenMH = @TenMH,DonGia = @DonGia,GiaNhap = @GiaNhap,HinhAnh = @HinhAnh  WHERE MaMH = @MaMH  UPDATE XE  SET LoaiXe = @LoaiXe,MauSac = @MauSac,KhoiLuongBanThan = @KhoiLuongBanThan,DxRxC = @DxRxC,KhoangCachTrucBanhXe = @KhoangCachTrucBanhXe,DoCaoYen = @DoCaoYen,  KhoangSangGamXe = @KhoangSangGamXe,DungTichBinhXang = @DungTichBinhXang,KichCoLopTruoc = @KichCoLopTruoc,KichCoLopSau = @KichCoLopSau,PhuocTruoc = @PhuocTruoc,  PhuocSau = @PhuocSau,LoaiDongCo = @LoaiDongCo,CongSuatToiDa = @CongSuatToiDa,DungTichNhotMay = @DungTichNhotMay,LoaiTruyenDong = @LoaiTruyenDong,HeThongKhoiDong = @HeThongKhoiDong,  MomentCucDai = @MomentCucDai,DungTichXiLanh = @DungTichXiLanh,DuongKinhXHanhTrinhPitTong = @DuongKinhXHanhTrinhPitTong,TySoNen = @TySoNen,DongCoCongNghe = @DongCoCongNghe  WHERE MaXE = @MaMH  END;  GO |

***Xem thông tin xe máy***

|  |
| --- |
| CREATE FUNCTION fn\_XemThongTinXe(  @MaXE NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT HinhAnh,HSX, TenMH, LoaiXe,MauSac,KhoiLuongBanThan,DxRxC,KhoangCachTrucBanhXe,DoCaoYen,KhoangSangGamXe,DungTichBinhXang,KichCoLopTruoc,KichCoLopSau,PhuocTruoc,PhuocSau,LoaiDongCo,CongSuatToiDa,DungTichNhotMay,  LoaiTruyenDong,HeThongKhoiDong,MomentCucDai,DungTichXiLanh,DuongKinhXHanhTrinhPitTong,TySoNen,DongCoCongNghe,Dongia  FROM XE JOIN Mat\_Hang ON MaXe = Mat\_Hang.MaMH  WHERE Mat\_Hang.MaMH=@MaXE  )  GO |

***Lọc và truy xuất thông tin của Xe máy***

|  |
| --- |
| SET ANSI\_NULLS ON  GO  SET QUOTED\_IDENTIFIER ON  GO  CREATE FUNCTION [dbo].[fn\_DSXe]  (  @LoaiXe NVARCHAR(MAX)  )  RETURNS TABLE  RETURN  (  SELECT\*  FROM vw\_DSXe  WHERE LoaiXe=@LoaiXe  )  GO |

1. **Quản lý phụ tùng xe máy**

***Thêm thông tin phụ tùng xe máy***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemPhuTung](  @HSX nvarchar(30),  @TenMH nvarchar(40),  @DonGia money,  @GiaNhap money,  @HinhAnh Image,  @ChatLuong nvarchar(30),  @LoaiPT nvarchar(100)  )  AS  BEGIN -- tạo biến để lưu mã mặt hàng  DECLARE @MaMH NVARCHAR(11), @LastID INT ;  IF EXISTS( SELECT 1 FROM Mat\_Hang JOIN Phu\_Tung on MaMH=MaPT WHERE HSX=@HSX and TenMH=@TenMH and ChatLuong=@ChatLuong and LoaiPT=@LoaiPT)-- check xem có phụ tùng nào giống mọi thông số không  BEGIN  RAISERROR (N'Thông tin phụ tùng đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  ELSE  BEGIN  SET @LastID = (SELECT ISNULL(MAX(CAST(SUBSTRING(MaPT, 3,LEN(MaPT)-2) AS INT)),0) FROM Phu\_Tung)-- tạo mã phụ tùng tự động  IF @LastID <= 99999  SET @MaMH = 'PT' + RIGHT('0000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 5)  ELSE  SET @MaMH = 'PT' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  INSERT INTO Mat\_Hang VALUES (@MaMH, @HSX, @TenMH, @DonGia, @GiaNhap, @HinhAnh)  INSERT INTO Phu\_Tung VALUES (@MaMH, @ChatLuong, @LoaiPT)  INSERT INTO Kho (MaCN, MaMH, SoLuong)  SELECT Chi\_Nhanh.MaCN, @MaMH, 0  FROM Chi\_Nhanh ;  END  END;  GO |

***Sửa thông tin phụ tùng xe máy***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaPhuTung]  @MaMH nvarchar(11),  @HSX nvarchar(30),  @TenMH nvarchar(40),  @DonGia money,  @GiaNhap money,  @HinhAnh Image,  @ChatLuong nvarchar(30),  @LoaiPT nvarchar(100)  AS  BEGIN  IF EXISTS(  SELECT 1  FROM Mat\_Hang  JOIN Phu\_Tung ON MaMH=MaPT  WHERE HSX=@HSX AND TenMH=@TenMH AND ChatLuong=@ChatLuong AND LoaiPT=@LoaiPT  AND DonGia=@DonGia AND GiaNhap=@GiaNhap HashBytes('SHA1', CONVERT(VARBINARY(MAX), HinhAnh)) = HashBytes('SHA1', CONVERT(VARBINARY(MAX), @HinhAnh))  ) BEGIN  RAISERROR (N'Thông tin phụ tùng đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  UPDATE Mat\_Hang SET HSX = @HSX, TenMH = @TenMH, DonGia = @DonGia, GiaNhap = @GiaNhap, HinhAnh = @HinhAnh WHERE MaMH = @MaMH  UPDATE Phu\_Tung SET ChatLuong = @ChatLuong, LoaiPT = @LoaiPT WHERE MaPT = @MaMH  END  GO |

***Lọc và truy xuất thông tin của phụ tùng xe máy***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_DSPhuTung]  (  @TenPT nvarchar(40),  @HSX nvarchar(30),  @ChatLuong nvarchar(30),  @LoaiPT nvarchar(100)  )  RETURNS TABLE  AS RETURN  (    SELECT \* FROM vw\_DSPhuTung  WHERE (@TenPT = N'' OR TenMH = @TenPT)  AND (@HSX = N'' OR HSX = @HSX)  AND (@ChatLuong = N'' OR ChatLuong = @ChatLuong)  AND (@LoaiPT = N'' OR LoaiPT = @LoaiPT)  )  GO |

1. **Quản lý Bảo dưỡng xe máy**

***Hàm sinh mã phiếu nhập tự động***

|  |
| --- |
|  |

***Lọc và truy xuất Tên Bảo dưỡng trong Danh sách bảo dưỡng***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_DSBaoDuong]  (  @TenBD nvarchar(30)  )  RETURNS TABLE  AS RETURN  (  SELECT \*  FROM vw\_DSBaoDuong  WHERE TenBD LIKE '%' + @TenBD + '%'  )  GO |

***Sửa bảo dưỡng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaBaoDuong](  @MaBD NVARCHAR(11),  @TenBD nvarchar(30),  @PhiBD money,  @ThongTinBaoDuong nvarchar(max)  )  AS  BEGIN  IF EXISTS (SELECT 1 FROM Bao\_Duong WHERE TenBD = @TenBD AND PhiBD = @PhiBD AND ThongTinBaoDuong = @ThongTinBaoDuong)  BEGIN  RAISERROR (N'Thông tin bảo dưỡng đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  UPDATE Bao\_Duong SET TenBD=@TenBD,PhiBD=@PhiBD,ThongTinBaoDuong=@ThongTinBaoDuong WHERE Bao\_Duong.MaBD=@MaBD  END  GO |

***Thêm bảo dưỡng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemBaoDuong](  @TenBD nvarchar(30),  @PhiBD money,  @ThongTinBaoDuong nvarchar(max)  )  AS  BEGIN  IF EXISTS (SELECT 1 FROM Bao\_Duong WHERE TenBD = @TenBD AND PhiBD = @PhiBD AND ThongTinBaoDuong = @ThongTinBaoDuong)  BEGIN  RAISERROR (N'Thông tin bảo dưỡng đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  ELSE  BEGIN  DECLARE @NextID NVARCHAR(11),@LastID INT  SELECT @LastID = ISNULL(MAX(CAST(SUBSTRING(MaBD, 3, LEN(MaBD) - 2) AS INT)), 0)  FROM Bao\_Duong -- lấy số bảo dưỡng có mã số cao nhất  IF @LastID <= 99999  SET @NextID = 'BD' + RIGHT('0000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 5)  ELSE  SET @NextID = 'BD' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  INSERT INTO Bao\_Duong VALUES (@NextID, @TenBD,@PhiBD, @ThongTinBaoDuong)  END  END  GO |

1. **Quản lý phiếu nhập**

***Hàm sinh mã phiếu nhập tự động***

|  |
| --- |
| CREATE FUNCTION fn\_MaPhieuNhapTuDong()-- hàm sinh mã phiếu nhập tự động  RETURNS NVARCHAR(11)  AS  BEGIN  DECLARE @NextID NVARCHAR(11), @LastID INT  SELECT @LastID = ISNULL(MAX(CAST(SUBSTRING(MaPN, 3,LEN(MaPN)-2) AS INT)),0) FROM Phieu\_Nhap -- lấy số của phiếu nhập gần đây nhất  IF @LastID <= 999999  SET @NextID = 'PN' + RIGHT('00000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 6)  ELSE  SET @NextID = 'PN' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  RETURN @NextID  END  go |

***Mỗi khi thêm mã phiếu nhập sẽ được tạo tự động***

|  |
| --- |
| ALTER TABLE Phieu\_Nhap -- mỗi khi thêm mã phiếu nhập sẽ được tạo tự động  ADD CONSTRAINT constr\_MaPN  Default dbo.fn\_MaPhieuNhapTuDong() for MaPN  GO |

***Lọc và truy xuất thông tin của Phiếu Nhập***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_DSPhieuNhap]  (  @MaNV NVARCHAR(11),  @TenNV NVARCHAR(40),  @TenNCC NVARCHAR(40),  @SDT NVARCHAR(11),  @TuNgay DATE,  @DenNgay DATE  )  RETURNS TABLE  AS RETURN  (  SELECT \* FROM vw\_DSPhieuNhap  WHERE (MaNV = CASE WHEN @MaNV = N'' THEN MaNV ELSE @MaNV END)  AND (TenNV = CASE WHEN @TenNV = N'' THEN TenNV ELSE @TenNV END)  AND (TenNCC = CASE WHEN @TenNCC= N'' THEN TenNCC ELSE @TenNCC END)  AND (SDT = CASE WHEN @SDT = N'' THEN SDT ELSE @SDT END)  AND (NgayNhap >= CASE WHEN @TuNgay IS NULL THEN NgayNhap ELSE @TuNgay END)  AND (NgayNhap <= CASE WHEN @DenNgay IS NULL THEN NgayNhap ELSE @DenNgay END)  )  GO |

***Lọc và truy xuất MaPN trong Thông tin phiếu nhập***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_ThongTinPhieuNhap]  (  @MaPN NVARCHAR(11)    )  RETURNS TABLE  AS RETURN  (  SELECT vw\_DSPhieuNhap.MaPN,vw\_DSPhieuNhap.TenCN, vw\_DSPhieuNhap.TenNCC,vw\_DSPhieuNhap.ThanhTien FROM vw\_DSPhieuNhap  WHERE vw\_DSPhieuNhap.MaPN=@MaPN  )  GO |

***Thêm phiếu nhập***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemPhieuNhap](  @NgayNhap date ,  @MaNVKho nvarchar(11) ,  @MaNCC nvarchar(11)  )  AS BEGIN  BEGIN try  INSERT INTO Phieu\_Nhap (NgayNhap,MaNVKho, MaNCC)  values (@NgayNhap,@MaNVKho,@MaNCC)    END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%constr\_chk\_NhanVienKhoHang%'  RAISERROR(N'Người lập hóa đơn phải là nhân viên quản lý kho còn đang làm việc', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Phieu\_Nha\_\_NgayN\_\_0E6E26BF%'  RAISERROR(N'Ngày nhập không được là 1 ngày trong tương lai', 16, 1)  END CATCH  END  GO |

1. **Quản lý hóa đơn**

***Xem hóa đơn***

|  |
| --- |
| CREATE FUNCTION fn\_XemHoaDon(  @MaKH NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT \* FROM Hoa\_Don WHERE Hoa\_Don.MaKH=@MaKH  )  go |

***Mã tạo hóa đơn tự động***

|  |
| --- |
| CREATE FUNCTION fn\_MaHoaDonTuDong()-- hàm sinh mã hoá đơn tự động  RETURNS NVARCHAR(11)  AS  BEGIN  DECLARE @NextID NVARCHAR(11), @LastID INT  SELECT @LastID = ISNULL(MAX(CAST(SUBSTRING(MaHD, 3,LEN(MaHD)-2) AS INT)),0) FROM Hoa\_Don -- lấy số của hoá đơn mới nhất  IF @LastID <= 999999  SET @NextID = 'HD' + RIGHT('00000' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR), 6)  ELSE  SET @NextID = 'HD' + CAST(ISNULL(@LastID, 0) + 1 AS NVARCHAR)  RETURN @NextID  END  GO |

***Lọc và truy xuất thông tin trong Hóa đơn***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_DSHoaDon]  (  @MaNV NVARCHAR(11),  @TenNV NVARCHAR(40),  @TenKH NVARCHAR(40),  @SDT NVARCHAR(11),  @TuNgay DATE,  @DenNgay DATE  )  RETURNS TABLE  AS RETURN  (  SELECT \* FROM vw\_DSHoaDon  WHERE (MaNV = CASE WHEN @MaNV = N'' THEN MaNV ELSE @MaNV END)  AND (TenNV = CASE WHEN @TenNV = N'' THEN TenNV ELSE @TenNV END)  AND (TenKH = CASE WHEN @TenKH = N'' THEN TenKH ELSE @TenKH END)  AND (SDT = CASE WHEN @SDT = N'' THEN SDT ELSE @SDT END)  AND (NgayLap >= CASE WHEN @TuNgay IS NULL THEN NgayLap ELSE @TuNgay END)  AND (NgayLap <= CASE WHEN @DenNgay IS NULL THEN NgayLap ELSE @DenNgay END)  )  GO |

***Lọc và truy xuất Mã HD trong Thông tin Hóa Đơn***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_ThongTinHoaDon]  (  @MaHD NVARCHAR(11)    )  RETURNS TABLE  AS RETURN  (  SELECT vw\_DSHoaDon.MaHD, vw\_DSHoaDon.TenCN, vw\_DSHoaDon.TenKH, vw\_DSHoaDon.ThanhTien FROM vw\_DSHoaDon  WHERE vw\_DSHoaDon.MaHD=@MaHD  )  GO |

***Thêm hóa đơn***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemHoaDon](  @NgayLap date ,  @MaNVThuNgan nvarchar(11) ,  @MaKH nvarchar(11)  )  AS BEGIN  BEGIN try  INSERT INTO Hoa\_Don (NgayLap,MaKH,MaNVThuNgan)  values (@NgayLap,@MaKH,@MaNVThuNgan)  END TRY  BEGIN CATCH  DECLARE @errorMessage NVARCHAR(4000)  SET @errorMessage = ERROR\_MESSAGE()  IF @errorMessage LIKE '%constr\_chk\_NhanVienThuNgan%'  RAISERROR(N'Người lập hóa đơn phải là nhân viên thu ngân còn đang làm việc', 16, 1)  ELSE IF @errorMessage LIKE '%CK\_\_Hoa\_Don\_\_NgayLap\_\_7C4F7684%'  RAISERROR(N'Ngày lập không được là 1 ngày trong tương lai', 16, 1)  END CATCH  END  GO |

1. **Quản lý nhà cung cấp**

***Lọc và truy xuất thông tin nhà cung cấp***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_ThongTinNCC](  @MaNCC NVARCHAR(11)  )  RETURNS TABLE  AS  RETURN (SELECT \* FROM vw\_DSNhaCungCap WHERE vw\_DSNhaCungCap.MaNCC=@MaNCC)  GO |

1. **Quản lý Chi tiết hóa đơn**

***Xem chi tiết hóa đơn Xe***

|  |
| --- |
| CREATE FUNCTION fn\_XemChiTietHoaDonXe(  @MaHD NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT mh.TenMH,XE.MauSac,mh.DonGia,cthd.SoLuong  FROM Chi\_Tiet\_Hoa\_Don cthd  JOIN Mat\_Hang mh ON cthd.MaMH = mh.MaMH  JOIN XE ON mh.MaMH = XE.MaXe  WHERE cthd.MaHD=@MaHD  )  go |

***Xem chi tiết hóa đơn phụ tùng***

|  |
| --- |
| CREATE FUNCTION fn\_XemChiTietHoaDonPhuTung(  @MaHD NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT mh.TenMH,Phu\_Tung.LoaiPT,Phu\_Tung.ChatLuong,mh.DonGia,cthd.SoLuong  FROM Chi\_Tiet\_Hoa\_Don cthd  JOIN Mat\_Hang mh ON cthd.MaMH = mh.MaMH  JOIN Phu\_Tung ON mh.MaMH = Phu\_Tung.MaPT  WHERE cthd.MaHD=@MaHD  )  go |

***Xem chi tiết hóa đơn bảo dưỡng***

|  |
| --- |
| CREATE FUNCTION fn\_XemChiTietHoaDonBaoDuong(  @MaHD NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT Bao\_Duong.TenBD,Bao\_Duong.PhiBD,cthd.SoLuong,cthd.MaNVBaoDuong  FROM Chi\_Tiet\_Hoa\_Don cthd  JOIN Bao\_Duong ON cthd.MaBD = Bao\_Duong.MaBD  WHERE cthd.MaHD=@MaHD  )  go |

***Sửa chi tiết hóa đơn Mặt Hàng ( Xe, Phụ tùng)***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaCTHDMatHang](  @MaHD NVARCHAR(11),  @MaMH NVARCHAR(11),  @SoLuong int  )  AS BEGIN  update Chi\_Tiet\_Hoa\_Don  set SoLuong=@SoLuong  where MaHD=@MaHD and MaMH=@MaMH  END  GO |

***Sửa chi tiết hóa đơn Bảo dưỡng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaCTHDBaoDuong](  @MaHD NVARCHAR(11),  @MaBD NVARCHAR(11),  @MaNVBD NVARCHAR(11),  @SoLuong int  )  AS BEGIN  update Chi\_Tiet\_Hoa\_Don  set SoLuong=@SoLuong  where MaHD=@MaHD and MaBD=@MaBD AND Chi\_Tiet\_Hoa\_Don.MaNVBaoDuong=@MaNVBD  END  GO |

***Thêm Chi tiết hóa đơn bảo dưỡng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemCTHDBaoDuong](  @MaHD NVARCHAR(11),  @MaBD NVARCHAR(11),  @MaNVBD NVARCHAR(11)  )  AS BEGIN  INSERT INTO Chi\_Tiet\_Hoa\_Don (MaHD, MaBD,SoLuong,MaNVBaoDuong)  VALUES (@MaHD, @MaBD, 1,@MaNVBD);  END  GO |

***Thêm chi tiết hóa đơn mặt hàng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemCTHDMatHang](  @MaHD NVARCHAR(11),  @MaMH NVARCHAR(11)  )  AS BEGIN  INSERT INTO Chi\_Tiet\_Hoa\_Don (MaHD, MaMH,SoLuong)  VALUES (@MaHD, @MaMH,1);  END  GO |

1. **Quản lý phiếu nhập**

***Xem chi tiết phiếu nhập Xe***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_XemChiTietPhieuNhapXe](  @MaPN NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT ctpn.MaMH,mh.TenMH,XE.MauSac,mh.GiaNhap,ctpn.SoLuong  FROM Chi\_Tiet\_Phieu\_Nhap ctpn  JOIN Mat\_Hang mh ON ctpn.MaMH = mh.MaMH  JOIN XE ON mh.MaMH = XE.MaXe  WHERE ctpn.MaPN=@MaPN  )  GO |

***Xem chi tiết phiếu nhập Phụ tùng xe máy***

|  |
| --- |
| CREATE FUNCTION [dbo].[fn\_XemChiTietPhieuNhapPhuTung](  @MaPN NVARCHAR(11)  )  RETURNS TABLE  AS RETURN  (  SELECT ctpn.MaMH,mh.TenMH,Phu\_Tung.LoaiPT,Phu\_Tung.ChatLuong,mh.GiaNhap,ctpn.SoLuong  FROM Chi\_Tiet\_Phieu\_Nhap ctpn  JOIN Mat\_Hang mh ON ctpn.MaMH = mh.MaMH  JOIN Phu\_Tung ON mh.MaMH = Phu\_Tung.MaPT  WHERE ctpn.MapN=@MaPN  )  GO |

***Xem chi tiết phiếu nhập Bảo dưỡng***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_SuaBaoDuong](  @MaBD NVARCHAR(11),  @TenBD nvarchar(30),  @PhiBD money,  @ThongTinBaoDuong nvarchar(max)  )  AS  BEGIN  IF EXISTS (SELECT 1 FROM Bao\_Duong WHERE TenBD = @TenBD AND PhiBD = @PhiBD AND ThongTinBaoDuong = @ThongTinBaoDuong)  BEGIN  RAISERROR (N'Thông tin bảo dưỡng đã tồn tại trong hệ thống!', 16, 1)  RETURN;  END  UPDATE Bao\_Duong SET TenBD=@TenBD,PhiBD=@PhiBD,ThongTinBaoDuong=@ThongTinBaoDuong WHERE Bao\_Duong.MaBD=@MaBD  END  GO |

***Thêm chi tiết phiếu nhập***

|  |
| --- |
| CREATE PROCEDURE [dbo].[sp\_ThemCTPN](  @MaPN NVARCHAR(11),  @MaMH NVARCHAR(11),  @SoLuong MONEY  )  AS BEGIN  INSERT INTO Chi\_Tiet\_Phieu\_Nhap (MaPN, MaMH,SoLuong)  VALUES (@MaPN, @MaMH,@SoLuong);  END  GO |

**CHƯƠNG 4: TẠO USER VÀ PHÂN QUYỀN**

**Trigger tạo tài khoản login và user khi tài khoản được tạo trên bảng Tai\_Khoan và thêm vào Role tương ứng:**

***Trigger thêm tài khoản***

|  |
| --- |
| CREATE TRIGGER trg\_ThemTaiKhoan ON Tai\_Khoan  AFTER INSERT AS  BEGIN  DECLARE @userName NVARCHAR(11), @passWord NVARCHAR(11), @role NVARCHAR(50)  SELECT @userName=INSERTED.TenDangNhap, @passWord=INSERTED.MatKhau, @role=INSERTED.ChucVu FROM INSERTED    DECLARE @sqlString NVARCHAR(200)  -- Tạo tài khoản login cho nhân viên, tên người dùng và mật khẩu là tài khoản được tạo trên bảng Tai\_Khoan  SET @sqlString = 'CREATE LOGIN ' + @userName + ' WITH PASSWORD=''' + @passWord + ''', DEFAULT\_DATABASE=[QLMoToStore], CHECK\_EXPIRATION=OFF, CHECK\_POLICY=OFF'  EXEC (@sqlString)  -- Tạo tài khoản người dùng đối với nhân viên đó trên database (tên người dùng trùng với tên login)  SET @sqlString= 'CREATE USER ' + @userName +' FOR LOGIN '+ @userName  EXEC (@sqlString)  IF (@role) = N'Nhân viên quản lý'  BEGIN  SET @sqlString = 'ALTER SERVER ROLE sysadmin ADD MEMBER ' + @userName;  EXEC (@sqlString)  END    IF (@role) = N'Nhân viên kho hàng' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_KhoHang ADD MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên bảo dưỡng'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_BaoDuong ADD MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên thu ngân'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_ThuNgan ADD MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên tư vấn' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_TuVan ADD MEMBER ' + @userName;  EXEC (@sqlString)  END  END  GO  CREATE TRIGGER trg\_XoaNhanVien ON Nhan\_Vien  INSTEAD OF DELETE AS  -- sài instead of để không cho xóa bảng ghi của nhân viên đó, mà chỉ set thuộc tính đi làm bằng 0  BEGIN  DECLARE @MaNV NVARCHAR(11) SELECT @MaNV = DELETED.MaNV FROM DELETED  UPDATE Nhan\_Vien  SET Nhan\_Vien.TinhTrangLamViec = 0  WHERE Nhan\_Vien.MaNV = (@MaNV)  IF (SELECT DELETED.ChucVu FROM DELETED) = N'Nhân viên quản lý' -- nếu quản lý bị xóa thì sẽ cập nhật quản lý ở cửa hàng này là null  BEGIN  DECLARE @MaCN NVARCHAR(11)  SELECT @MaCN=MaCN FROM DELETED  UPDATE Chi\_Nhanh SET MaNQL = null WHERE MaCN=@MaCN  END  DELETE FROM Tai\_Khoan WHERE Tai\_Khoan.MaNV = (@MaNV)  END  GO  CREATE TRIGGER trg\_XoaTaiKhoan ON Tai\_Khoan  AFTER DELETE AS  BEGIN  DECLARE @userName NVARCHAR(11), @role NVARCHAR(50)  SELECT @userName=DELETED.TenDangNhap, @role=DELETED.ChucVu FROM DELETED  DECLARE @sqlString NVARCHAR(200)  IF (@role) = N'Nhân viên quản lý'  BEGIN  SET @sqlString = 'ALTER SERVER ROLE sysadmin DROP MEMBER ' + @userName;  EXEC (@sqlString)  END    IF (@role) = N'Nhân viên kho hàng' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_KhoHang DROP MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên bảo dưỡng'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_BaoDuong DROP MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên thu ngân'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_ThuNgan DROP MEMBER ' + @userName;  EXEC (@sqlString)  END  IF (@role) = N'Nhân viên tư vấn' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_TuVan DROP MEMBER ' + @userName;  EXEC (@sqlString)  END  SET @sqlString= 'DROP USER ' + @userName  EXEC (@sqlString)  SET @sqlString= 'DROP LOGIN '+ @userName  EXEC (@sqlString)  END  GO |

***Trigger sửa tài khoản***

|  |
| --- |
| CREATE TRIGGER trg\_SuaTaiKhoan ON Tai\_Khoan  AFTER UPDATE AS  BEGIN  DECLARE @userNameD NVARCHAR(11), @roleD NVARCHAR(50)  SELECT @userNameD=DELETED.TenDangNhap, @roleD=DELETED.ChucVu FROM DELETED  DECLARE @sqlString NVARCHAR(200)  IF (@roleD) = N'Nhân viên quản lý'  BEGIN  SET @sqlString = 'ALTER SERVER ROLE sysadmin DROP MEMBER ' + @userNameD;  EXEC (@sqlString)  END  IF (@roleD) = N'Nhân viên kho hàng' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_KhoHang DROP MEMBER ' + @userNameD;  EXEC (@sqlString)  END  IF (@roleD) = N'Nhân viên bảo dưỡng'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_BaoDuong DROP MEMBER ' + @userNameD;  EXEC (@sqlString)  END  IF (@roleD) = N'Nhân viên thu ngân'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_ThuNgan DROP MEMBER ' + @userNameD;  EXEC (@sqlString)  END  IF (@roleD) = N'Nhân viên tư vấn' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_TuVan DROP MEMBER ' + @userNameD;  EXEC (@sqlString)  END  DECLARE @userNameI NVARCHAR(11), @roleI NVARCHAR(50)  SELECT @userNameI=INSERTED.TenDangNhap, @roleI=INSERTED.ChucVu FROM INSERTED  IF (@roleI) = N'Nhân viên quản lý'  BEGIN  SET @sqlString = 'ALTER SERVER ROLE sysadmin ADD MEMBER ' + @userNameI;  EXEC (@sqlString)  END  IF (@roleI) = N'Nhân viên kho hàng' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_KhoHang ADD MEMBER ' + @userNameI;  EXEC (@sqlString)  END  IF (@roleI) = N'Nhân viên bảo dưỡng'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_BaoDuong ADD MEMBER ' + @userNameI;  EXEC (@sqlString)  END  IF (@roleI) = N'Nhân viên thu ngân'  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_ThuNgan ADD MEMBER ' + @userNameI;  EXEC (@sqlString)  END  IF (@roleI) = N'Nhân viên tư vấn' --Sửa trong form  BEGIN  SET @sqlString = 'ALTER ROLE role\_NhanVien\_TuVan ADD MEMBER ' + @userNameI;  EXEC (@sqlString)  END  END  GO |

**Ứng với hai nhóm người dùng (Nhân viên và quản lý) là các role riêng**

+ Đối với Role NhanVienQuanLy, yêu cần cần có quyền gán quyền cho nhân viên, sử dụng server Role sysadmin cho Quản lý.

+ Đối với NhanVienKhoHang

|  |
| --- |
| -- Role nhân viên kho  CREATE ROLE role\_NhanVien\_KhoHang  GO  GRANT EXECUTE ON [dbo].[sp\_ThemCTPN] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[Nhan\_Vien] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_DSPhuTung] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_DSPhieuNhap] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_DSXe] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[vw\_DSXe] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[Phieu\_Nhap] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_ThongTinPhieuNhap] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_LayThongTinNhanVien] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[vw\_DSPhieuNhap] TO role\_NhanVien\_KhoHang  GO  ---  GRANT SELECT ON [dbo].[fn\_ThongTinNCC] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[vw\_DSPhuTung] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[vw\_DSNhaCungCap] TO role\_NhanVien\_KhoHang  GO  GRANT EXECUTE ON [dbo].[sp\_ThemPhuTung] TO role\_NhanVien\_KhoHang  GO  GRANT EXECUTE ON [dbo].[sp\_SuaPhuTung] TO role\_NhanVien\_KhoHang  GO  GRANT EXECUTE ON [dbo].[sp\_ThemPhieuNhap] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_XemChiTietPhieuNhapPhuTung] TO role\_NhanVien\_KhoHang  GO  GRANT SELECT ON [dbo].[fn\_XemChiTietPhieuNhapXe] TO role\_NhanVien\_KhoHang  GO |

+ Đối với nhân viên thu ngân

|  |
| --- |
| -- Role nhân viên bảo dưỡng  CREATE ROLE role\_NhanVien\_BaoDuong  GO  GRANT SELECT ON [dbo].[fn\_DSBaoDuong] TO role\_NhanVien\_BaoDuong  GO  ---  GRANT SELECT ON [dbo].[Nhan\_Vien] TO role\_NhanVien\_BaoDuong  GO  GRANT SELECT, DELETE ON [dbo].[Bao\_Duong] TO role\_NhanVien\_BaoDuong  GO  GRANT SELECT ON [dbo].[vw\_DSBaoDuong] TO role\_NhanVien\_BaoDuong  GO  GRANT EXECUTE ON [dbo].[sp\_ThemBaoDuong] TO role\_NhanVien\_BaoDuong  GO  GRANT EXECUTE ON [dbo].[sp\_SuaBaoDuong] TO role\_NhanVien\_BaoDuong  GO |

+ Đối với nhân viên thu ngân

|  |
| --- |
| -- Role nhân viên thu ngân  CREATE ROLE role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[Chi\_Tiet\_Hoa\_Don] TO role\_NhanVien\_ThuNgan  GO  GRANT DELETE ON [dbo].[Chi\_Tiet\_Hoa\_Don] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_SuaCTHDBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemCTHDBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_SuaCTHDMatHang] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemCTHDMatHang] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_DSNhanVien] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[vw\_DSPhuTung] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[vw\_DSXe] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_DSXe] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_XemChiTietHoaDonBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_XemChiTietHoaDonPhuTung] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_XemChiTietHoaDonXe] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_ThongTinHoaDon] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[Hoa\_Don] TO role\_NhanVien\_ThuNgan  GO  ---  GRANT SELECT ON [dbo].[Nhan\_Vien] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_DSHoaDon] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_LayThongTinNhanVien] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[vw\_DSKhachHang] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_TimKhachHangTheoSĐT] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_TimKhachHangTheoCCCD] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[vw\_DSBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[fn\_DSBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT ON [dbo].[vw\_DSHoaDon] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemHoaDon] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_SuaBaoDuong] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemKhachHang] TO role\_NhanVien\_ThuNgan  GO  GRANT EXECUTE ON [dbo].[sp\_SuaKhachHang] TO role\_NhanVien\_ThuNgan  GO  GRANT SELECT, DELETE ON [dbo].[Bao\_Duong] TO role\_NhanVien\_ThuNgan  GO |

+ Đối với nhân viên tư vấn bán hàng

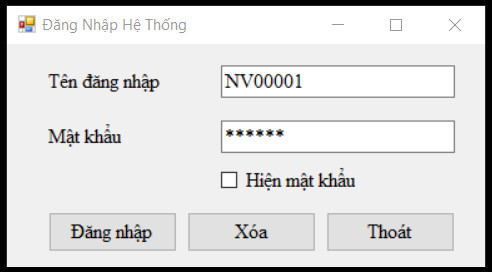
|  |
| --- |
| -- Role nhân viên tư vấn  CREATE ROLE role\_NhanVien\_TuVan  GO  GRANT SELECT ON [dbo].[fn\_DSPhuTung] TO role\_NhanVien\_TuVan  GO  ---  GRANT SELECT ON [dbo].[Nhan\_Vien] TO role\_NhanVien\_TuVan  GO  GRANT SELECT ON [dbo].[vw\_DSKhachHang] TO role\_NhanVien\_TuVan  GO  GRANT SELECT ON [dbo].[vw\_DSPhuTung] TO role\_NhanVien\_TuVan  GO  GRANT EXECUTE ON [dbo].[sp\_ThemKhachHang] TO role\_NhanVien\_TuVan  GO  GRANT EXECUTE ON [dbo].[sp\_SuaKhachHang] TO role\_NhanVien\_TuVan  GO |

**CHƯƠNG 5. THIẾT KẾ GIAO DIỆN HỆ THỐNG**

**Môi trường cài đặt và công nghệ sử dụng**

- Hệ cơ sở dữ liệu Microsoft SQL Server Management Studio 18.

- Chương trình Winform được viết bằng Microsoft Visual Studio 2022 - trên nền tảng ASP.NET Web Application (.NET Framework) C#. Giao diện và chức năng của chương trình:

1. **Form Menu Quản lý Thông tin MoToStore**
2. 
3. **Form đăng nhập**
4. **Các Form danh mục**

- Các hàm, trigger, view, thủ tục liên quan được gọi từ bên sql qua c#

1. **Danh mục Danh sách Chi Nhánh**

|  |
| --- |
| public DataTable LayDSChiNhanh()  {  using (SqlDataAdapter adapter = new SqlDataAdapter($"Select \* From vw\_DSChiNhanh", Connection))  {  DataTable data = new DataTable();  adapter.Fill(data);  return data;  }  } |

1. **Danh mục Danh sách Nhà Cung Cấp**

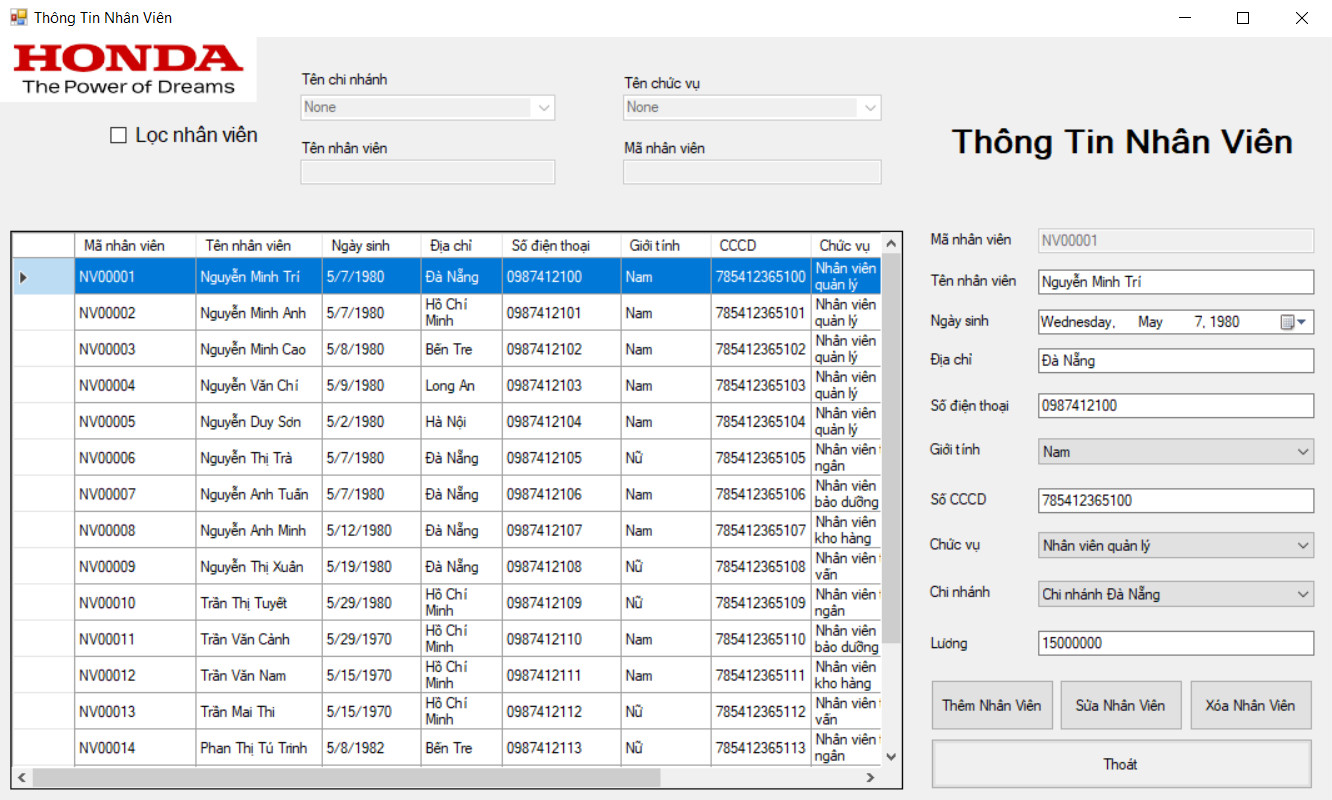
|  |
| --- |
| public DataTable LayDSNhaCungCap()  {  using (SqlDataAdapter adapter = new SqlDataAdapter($"Select \* From vw\_DSNhaCungCap", Connection))  {  DataTable data = new DataTable();  adapter.Fill(data);  return data;  }  } |

1. **Danh mục Thông tin Nhà Cung Cấp**

|  |
| --- |
| public DataTable ThongTinNhaCungCap(string MaNCC)  {  using (SqlDataAdapter adapter = new SqlDataAdapter($"Select \* From fn\_ThongTinNCC(@MaNCC)", Connection))  {  adapter.SelectCommand.Parameters.AddWithValue("@MaNCC", MaNCC);  DataTable data = new DataTable();  adapter.Fill(data);  return data;  }  } |

1. **Quản lý nhân viên**

* ***From quản lý danh sách nhân viên***



1. ***Hiện thị danh sách nhân viên***

|  |
| --- |
| public void LocNhanVien()  {  if (chkLoc.Checked)  {  string TenCN = cbbLocTenCN.Text == "None"?"" : cbbLocTenCN.Text;  string ChucVu = cbbLocChucVu.Text == "None" ? "" : cbbLocChucVu.Text;  string TenNV = txtLocTenNV.Text;  string MaNV = txtLocMaNV.Text;  dgvNhanVien.DataSource = employee.LayDS(TenCN, ChucVu, TenNV, MaNV);  }  else  dgvNhanVien.DataSource = employee.LayDS();  } |

1. ***Hiển thị các thông tin đã được thêm***

|  |
| --- |
| private void dgvNhanVien\_Click(object sender, EventArgs e)  {  if (dgvNhanVien.CurrentRow != null)  {  DataGridViewRow row = dgvNhanVien.CurrentRow;  if (row.Cells["MaNV"].Value.ToString() != "")  {  txtMaNV.Text = row.Cells["MaNV"].Value.ToString();  txtTenNV.Text = row.Cells["TenNV"].Value.ToString();  dtpNgaySinh.Value = (DateTime)row.Cells["NgaySinh"].Value;  txtDiaChi.Text = row.Cells["DiaChi"].Value.ToString();  txtSDT.Text = row.Cells["SDT"].Value.ToString();  cbbGioiTinh.Text = row.Cells["GioiTinh"].Value.ToString();  txtCCCD.Text = row.Cells["CCCD"].Value.ToString();  cbbChucVu.Text = row.Cells["ChucVu"].Value.ToString();  cbbTenCN.Text = row.Cells["TenCN"].Value.ToString();  txtLuong.Text = Convert.ToDecimal(row.Cells["Luong"].Value).ToString("#");  }  else  ClearPanel();  }  }  public void ClearPanel()  {  this.txtMaNV.Clear();  this.txtTenNV.Clear();  this.dtpNgaySinh.ResetText();  this.txtDiaChi.Clear();  this.txtSDT.Clear();  this.cbbGioiTinh.SelectedIndex = -1;  this.txtCCCD.Clear();  this.cbbChucVu.SelectedIndex = -1;  this.cbbTenCN.SelectedIndex = -1;  this.txtLuong.Clear();  } |

1. ***Lọc Nhân Viên***

|  |
| --- |
| private void cbbLocTenCN\_SelectedIndexChanged(object sender, EventArgs e)  {  LocNhanVien();  }  private void cbbLocChucVu\_SelectedIndexChanged(object sender, EventArgs e)  {  LocNhanVien();  }  private void txtLocTenNV\_TextChanged(object sender, EventArgs e)  {  LocNhanVien();  }  private void txtLocMaNV\_TextChanged(object sender, EventArgs e)  {  LocNhanVien();  }  private void chkLoc\_CheckedChanged(object sender, EventArgs e)  {  if (this.chkLoc.Checked == true)  {  this.cbbLocTenCN.Enabled = true;  this.cbbLocChucVu.Enabled = true;  this.txtLocTenNV.Enabled = true;  this.txtLocMaNV.Enabled = true;  }  else  {  this.cbbLocTenCN.SelectedIndex = 5;  this.cbbLocTenCN.Enabled = false;  this.cbbLocChucVu.SelectedIndex = 5;  this.cbbLocChucVu.Enabled = false;  this.txtLocTenNV.Clear();  this.txtLocTenNV.Enabled = false;  this.txtLocMaNV.Clear();  this.txtLocMaNV.Enabled = false;  }  LocNhanVien();  } |

1. ***Thêm Nhân Viên***

|  |
| --- |
| private void btnThemNhanVien\_Click(object sender, EventArgs e)  {  this.txtMaNV.Clear();  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (!Regex.IsMatch(txtLuong.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Lương phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (this.cbbChucVu.SelectedItem.ToString() == "Nhân viên quản lý" && MenuForm.chucVu != "Quản trị viên")  {  MessageBox.Show("Nhân viên quản lý không được thêm nhân viên quản lý khác", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string TenNV = txtTenNV.Text;  DateTime NgaySinh = dtpNgaySinh.Value;  string DiaChi = txtDiaChi.Text;  string SDT = txtSDT.Text;  string GioiTinh = cbbGioiTinh.Text;  string CCCD = txtCCCD.Text;  string ChucVu = cbbChucVu.Text;  string TenCN = cbbTenCN.Text;  decimal Luong = Convert.ToDecimal(txtLuong.Text);  this.chkLoc.Checked = false;  employee.Insert(TenNV, NgaySinh, DiaChi, SDT, GioiTinh, CCCD, ChucVu, TenCN, Luong);  LocNhanVien();  ClearPanel();  }  } |

1. ***Sửa Nhân Viên***

|  |
| --- |
| private void btnSuaNhanVien\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn chỉnh sửa không?", "Xác nhận chỉnh sửa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaNV.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn nhân viên", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!Regex.IsMatch(txtLuong.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Lương phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (this.cbbChucVu.SelectedItem.ToString() == "Nhân viên quản lý" && MenuForm.chucVu != "Quản trị viên")  {  MessageBox.Show("Nhân viên quản lý không được sửa thông tin của mình hoặc nhân viên quản lý khác", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string MaNV = txtMaNV.Text;  string TenNV = txtTenNV.Text;  DateTime NgaySinh = dtpNgaySinh.Value;  string DiaChi = txtDiaChi.Text;  string SDT = txtSDT.Text;  string GioiTinh = cbbGioiTinh.Text;  string CCCD = txtCCCD.Text;  string ChucVu = cbbChucVu.Text;  string TenCN = cbbTenCN.Text;  decimal Luong = Convert.ToDecimal(txtLuong.Text);  this.chkLoc.Checked = false;  employee.Update(MaNV, TenNV, NgaySinh, DiaChi, SDT, GioiTinh, CCCD, ChucVu, TenCN, Luong);  ClearPanel();  LocNhanVien();  }  } |

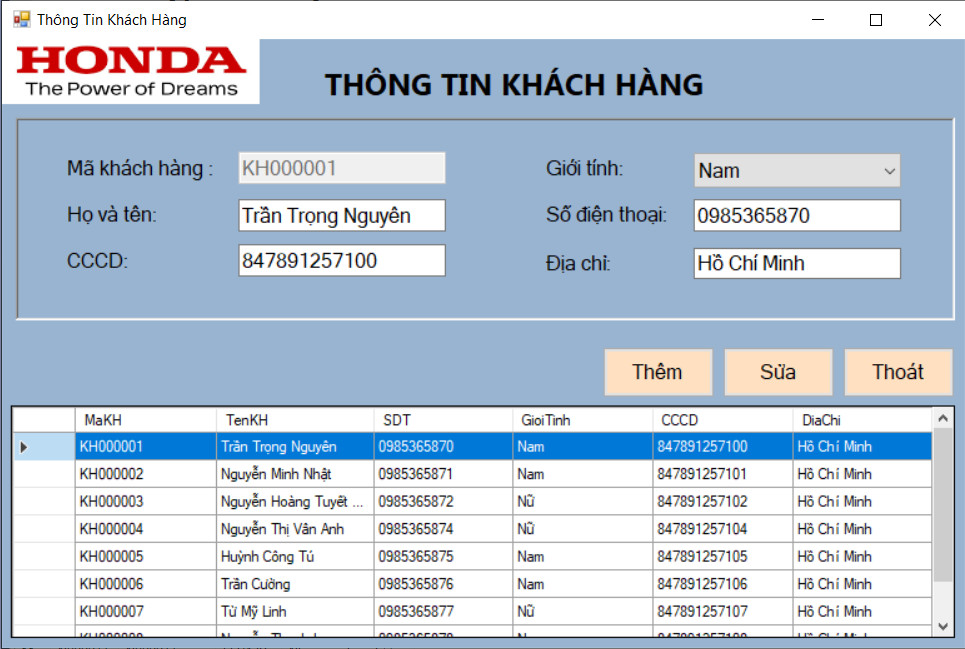
1. ***Xóa Nhân Viên***

|  |
| --- |
| private void btnXoaNhanVien\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn xóa không?", "Xác nhận xóa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  {  return;  }  if (this.txtMaNV.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn nhân viên", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (this.cbbChucVu.SelectedItem.ToString() == "Nhân viên quản lý" && MenuForm.chucVu != "Quản trị viên")  {  MessageBox.Show("Nhân viên quản lý không được xoá thông tin của mình hoặc nhân viên quản lý khác", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  this.chkLoc.Checked = false;  employee.Delete(txtMaNV.Text);  LocNhanVien();  } |

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. ***Thoát khỏi From Danh sách nhân viên***
2. **Quản lý khách hàng**

* ***From quản lý khách hàng***

****

1. ***Thêm khách hàng***

|  |
| --- |
| private void btnThem\_Click(object sender, EventArgs e)  {  this.txtMaKH.Clear();  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (!checkEmpty())  {  string TenKH = txtTenKH.Text;  string CCCD = txtCCCD.Text;  string SDT = txtSDT.Text;  string GioiTinh = cbbGioiTinh.Text;  string DiaChi = txtDiaChi.Text;  customer.Insert(TenKH, SDT, GioiTinh, CCCD, DiaChi);  KhachHangForm\_Load(sender, e);  }  } |

1. ***Sửa khách hàng***

|  |
| --- |
| private void btnSua\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn chỉnh sửa không?", "Xác nhận chỉnh sửa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaKH.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn khách hàng", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string MaKH = txtMaKH.Text;  string TenKH = txtTenKH.Text;  string CCCD = txtCCCD.Text;  string SDT = txtSDT.Text;  string GioiTinh = cbbGioiTinh.Text;  string DiaChi = txtDiaChi.Text;  customer.Update(MaKH, TenKH, SDT, GioiTinh, CCCD, DiaChi);  ClearPanel();  KhachHangForm\_Load(sender, e);  }  } |

1. ***Hiện thị thông tin khách hàng***

|  |
| --- |
| private void dgvKhachHang\_Click(object sender, EventArgs e)  {  if (dgvKhachHang.CurrentRow != null)  {  DataGridViewRow row = dgvKhachHang.CurrentRow;  if (row.Cells["MaKH"].Value.ToString() != "")  {  txtMaKH.Text = row.Cells["MaKH"].Value.ToString();  txtTenKH.Text = row.Cells["TenKH"].Value.ToString();  txtDiaChi.Text = row.Cells["DiaChi"].Value.ToString();  txtSDT.Text = row.Cells["SDT"].Value.ToString();  cbbGioiTinh.Text = row.Cells["GioiTinh"].Value.ToString();  txtCCCD.Text = row.Cells["CCCD"].Value.ToString();  }  else  ClearPanel();  }  }  public void ClearPanel()  {  this.txtMaKH.Clear();  this.txtTenKH.Clear();  this.txtCCCD.Clear();  this.cbbGioiTinh.SelectedIndex = -1;  this.txtSDT.Clear();  this.txtDiaChi.Clear();  } |

1. ***Thoát khỏi From thông tin khách hàng***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. **Quản lý xe**

* ***Form quản lý thông tin xe máy***

****

1. ***Thêm xe máy***

|  |
| --- |
| private void btnThemXe\_Click(object sender, EventArgs e)  {  this.txtMaXe.Clear();  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (!Regex.IsMatch(txtGiaNhap.Text, @"^[0-9,]+$") || !Regex.IsMatch(txtDonGia.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Đơn giá và Giá nhập phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string HSX = txtHSX.Text;  string TenXe = txtTenXe.Text;  decimal GiaNhap = Convert.ToDecimal(txtGiaNhap.Text);  decimal DonGia = Convert.ToDecimal(txtDonGia.Text);  string LoaiXe = txtLoaiXe.Text;  string MauSac = txtMauSac.Text;  string KhoiLuong = txtKhoiLuongBanThan.Text;  string KhoangCachTrucBanhXe = txtKhoangCachTrucBanhXe.Text;  string KhoangSangGamXe = txtKhoangSangGamXe.Text;  string DungTichBinhXang = txtDungTichBinhXang.Text;  string KichCoLopTruoc = txtKichCoLopTruoc.Text;  string KichCoLopSau = txtKichCoLopSau.Text;  string HeThongKhoiDong = txtHeThongKhoiDong.Text;  string MomentCucDai = txtMomentCucDai.Text;  string DungTichXiLanh = txtDungTichXiLanh.Text;  string TiSoNen = txtTiSoNen.Text;  string DxRxC = txtDxRxC.Text;  string DoCaoYenXe = txtDoCaoYenXe.Text;  string PhuocTruoc = txtPhuocTruoc.Text;  string PhuocSau = txtPhuocSau.Text;  string LoaiDongCo = txtLoaiDongCo.Text;  string CongSuatToiDa = txtCongSuatToiDa.Text;  string DungTichNhot = txtDungTichNhotMay.Text;  string LoaiTruyenDong = txtLoaiTruyenDong.Text;  string CongNgheDongCo = txtCongNgheDongCo.Text;  string DuongKinhxHanhTrinhPitTong = txtDuongKinhXHanhTrinhPitTong.Text;  Image image = picXe.Image;  using (MemoryStream ms = new MemoryStream())  {  image.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);  byte[] HinhAnh = ms.ToArray();  moto.Insert(HSX, TenXe, DonGia, GiaNhap, HinhAnh, LoaiXe, MauSac, KhoiLuong, DxRxC, KhoangCachTrucBanhXe, DoCaoYenXe, KhoangSangGamXe, DungTichBinhXang,  KichCoLopTruoc, KichCoLopSau, PhuocTruoc, PhuocSau, LoaiDongCo, CongSuatToiDa, DungTichNhot, LoaiTruyenDong, HeThongKhoiDong, MomentCucDai, DungTichXiLanh,  DuongKinhxHanhTrinhPitTong, TiSoNen, CongNgheDongCo);  }  LocXe();  }  }  private bool checkEmpty()  {  if (txtHSX.Text.Trim() == "" || txtTenXe.Text.Trim() == "" || txtGiaNhap.Text.Trim() == "" || txtDonGia.Text.Trim() == "" || txtLoaiXe.Text.Trim() == "" || txtMauSac.Text.Trim() == "" || txtKhoiLuongBanThan.Text.Trim() == "" || txtKhoangCachTrucBanhXe.Text.Trim() == "" || txtKhoangSangGamXe.Text.Trim() == "" || txtDungTichBinhXang.Text.Trim() == "" || txtKichCoLopTruoc.Text.Trim() == "" || txtKichCoLopSau.Text.Trim() == "" || txtHeThongKhoiDong.Text.Trim() == "" || txtMomentCucDai.Text.Trim() == "" || txtDungTichXiLanh.Text.Trim() == "" || txtTiSoNen.Text.Trim() == "" || txtDxRxC.Text.Trim() == "" || txtDoCaoYenXe.Text.Trim() == "" || txtPhuocTruoc.Text.Trim() == "" || txtPhuocSau.Text.Trim() == "" || txtLoaiDongCo.Text.Trim() == "" || txtCongSuatToiDa.Text.Trim() == "" || txtDungTichNhotMay.Text.Trim() == "" || txtLoaiTruyenDong.Text.Trim() == "" || txtCongNgheDongCo.Text.Trim() == "" || txtDuongKinhXHanhTrinhPitTong.Text.Trim() == "" || picXe.Image == null)  {  MessageBox.Show("Vui lòng nhập đầy đủ thông tin Xe.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return true;  }  else  {  return false;  }  } |

1. ***Sửa xe máy***

|  |
| --- |
| private void btnSuaXe\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn chỉnh sửa không?", "Xác nhận chỉnh sửa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaXe.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn xe", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!Regex.IsMatch(txtGiaNhap.Text, @"^[0-9,]+$") || !Regex.IsMatch(txtDonGia.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Đơn giá và Giá nhập phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string MaXe = txtMaXe.Text;  string HSX = txtHSX.Text;  string TenXe = txtTenXe.Text;  decimal GiaNhap = Convert.ToDecimal(txtGiaNhap.Text);  decimal DonGia = Convert.ToDecimal(txtDonGia.Text);  string LoaiXe = txtLoaiXe.Text;  string MauSac = txtMauSac.Text;  string KhoiLuong = txtKhoiLuongBanThan.Text;  string KhoangCachTrucBanhXe = txtKhoangCachTrucBanhXe.Text;  string KhoangSangGamXe = txtKhoangSangGamXe.Text;  string DungTichBinhXang = txtDungTichBinhXang.Text;  string KichCoLopTruoc = txtKichCoLopTruoc.Text;  string KichCoLopSau = txtKichCoLopSau.Text;  string HeThongKhoiDong = txtHeThongKhoiDong.Text;  string MomentCucDai = txtMomentCucDai.Text;  string DungTichXiLanh = txtDungTichXiLanh.Text;  string TiSoNen = txtTiSoNen.Text;  string DxRxC = txtDxRxC.Text;  string DoCaoYenXe = txtDoCaoYenXe.Text;  string PhuocTruoc = txtPhuocTruoc.Text;  string PhuocSau = txtPhuocSau.Text;  string LoaiDongCo = txtLoaiDongCo.Text;  string CongSuatToiDa = txtCongSuatToiDa.Text;  string DungTichNhot = txtDungTichNhotMay.Text;  string LoaiTruyenDong = txtLoaiTruyenDong.Text;  string CongNgheDongCo = txtCongNgheDongCo.Text;  string DuongKinhxHanhTrinhPitTong = txtDuongKinhXHanhTrinhPitTong.Text;  Image image = picXe.Image;  using (MemoryStream ms = new MemoryStream())  {  image.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);  byte[] HinhAnh = ms.ToArray();  moto.Update(MaXe, HSX, TenXe, DonGia, GiaNhap, HinhAnh, LoaiXe, MauSac, KhoiLuong, DxRxC, KhoangCachTrucBanhXe, DoCaoYenXe, KhoangSangGamXe, DungTichBinhXang,  KichCoLopTruoc, KichCoLopSau, PhuocTruoc, PhuocSau, LoaiDongCo, CongSuatToiDa, DungTichNhot, LoaiTruyenDong, HeThongKhoiDong, MomentCucDai, DungTichXiLanh,  DuongKinhxHanhTrinhPitTong, TiSoNen, CongNgheDongCo);  }  LocXe();  }  } |

1. ***Hiện thị thông tin xe máy***

|  |
| --- |
| private void dgvXe\_Click(object sender, EventArgs e)  {  Image image = picXe.Image;  if (dgvXe.CurrentRow != null)  {  DataGridViewRow row = dgvXe.CurrentRow;  if (row.Cells["MaMH"].Value.ToString() != "")  {  txtMaXe.Text = row.Cells["MaMH"].Value.ToString();  txtHSX.Text = row.Cells["HSX"].Value.ToString();  txtTenXe.Text = row.Cells["TenMH"].Value.ToString();  txtGiaNhap.Text = Convert.ToDecimal(row.Cells["GiaNhap"].Value).ToString("#");  txtDonGia.Text = Convert.ToDecimal(row.Cells["DonGia"].Value).ToString("#");  txtLoaiXe.Text = row.Cells["LoaiXe"].Value.ToString();  txtMauSac.Text = row.Cells["MauSac"].Value.ToString();  txtKhoiLuongBanThan.Text = row.Cells["KhoiLuongBanThan"].Value.ToString();  txtKhoangCachTrucBanhXe.Text = row.Cells["KhoangCachTrucBanhXe"].Value.ToString();  txtKhoangSangGamXe.Text = row.Cells["KhoangSangGamXe"].Value.ToString();  txtDungTichBinhXang.Text = row.Cells["DungTichBinhXang"].Value.ToString();  txtKichCoLopTruoc.Text = row.Cells["KichCoLopTruoc"].Value.ToString();  txtKichCoLopSau.Text = row.Cells["KichCoLopSau"].Value.ToString();  txtHeThongKhoiDong.Text = row.Cells["HeThongKhoiDong"].Value.ToString();  txtMomentCucDai.Text = row.Cells["MomentCucDai"].Value.ToString();  txtDungTichXiLanh.Text = row.Cells["DungTichXiLanh"].Value.ToString();  txtTiSoNen.Text = row.Cells["TySoNen"].Value.ToString();  txtDxRxC.Text = row.Cells["DxRxC"].Value.ToString();  txtDoCaoYenXe.Text = row.Cells["DoCaoYen"].Value.ToString();  txtPhuocTruoc.Text = row.Cells["PhuocTruoc"].Value.ToString();  txtPhuocSau.Text = row.Cells["PhuocSau"].Value.ToString();  txtLoaiDongCo.Text = row.Cells["LoaiDongCo"].Value.ToString();  txtCongSuatToiDa.Text = row.Cells["CongSuatToiDa"].Value.ToString();  txtDungTichNhotMay.Text = row.Cells["DungTichNhotMay"].Value.ToString();  txtLoaiTruyenDong.Text = row.Cells["LoaiTruyenDong"].Value.ToString();  txtCongNgheDongCo.Text = row.Cells["DongCoCongNghe"].Value.ToString();  txtDuongKinhXHanhTrinhPitTong.Text = row.Cells["DuongKinhxHanhTrinhPitTong"].Value.ToString();  }  }  } |

1. ***Thêm ảnh bằng link***

|  |
| --- |
| private void btnThemAnh\_Click(object sender, EventArgs e)  {  try  {  byte[] HinhAnh = new WebClient().DownloadData(txtLinkAnh.Text);  MemoryStream stream = new MemoryStream(HinhAnh);  picXe.Image = Image.FromStream(stream);  }  catch  {  MessageBox.Show("Link ảnh không đúng", "Thông Báo", MessageBoxButtons.OK, MessageBoxIcon.Error);  }  } |

1. ***Chọn ảnh có sẵn***

|  |
| --- |
| private void btnChonAnh\_Click(object sender, EventArgs e)  {  OpenFileDialog openFileDialog1 = new OpenFileDialog();  openFileDialog1.Filter = "Image files (\*.jpg, \*.jpeg, \*.jpe, \*.jfif, \*.png, \*.bmp) | \*.jpg; \*.jpeg; \*.jpe; \*.jfif; \*.png; \*.bmp";  if (openFileDialog1.ShowDialog() == DialogResult.OK)  {  string imagePath = openFileDialog1.FileName;  picXe.Image = Image.FromFile(imagePath);  }  } |

1. ***Xóa ảnh***

|  |
| --- |
| private void btnXoaAnh\_Click(object sender, EventArgs e)  {  picXe.Image = null;  } |

1. ***Thoát khỏi From Thông Tin Xe***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. **Quản lý Phụ tùng**

* ***Form quản lý Thông tin phụ tùng***

|  |
| --- |
|  |

1. ***Thêm phụ tùng***

|  |
| --- |
| private void btnThemPhuTung\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (!Regex.IsMatch(txtGiaNhap.Text, @"^[0-9,]+$") || !Regex.IsMatch(txtDonGia.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Đơn giá và Giá nhập phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string HSX = txtHSX.Text;  string TenPT = txtTenPT.Text;  decimal GiaNhap = Convert.ToDecimal(txtGiaNhap.Text);  decimal DonGia = Convert.ToDecimal(txtDonGia.Text);  string LoaiPT = txtLoaiPT.Text;  string ChatLuong = txtChatLuong.Text;  Image image = picPhuTung.Image;  using (MemoryStream ms = new MemoryStream())  {  image.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);  byte[] HinhAnh = ms.ToArray();  sparepart.Insert(HSX, TenPT, DonGia, GiaNhap, HinhAnh, ChatLuong, LoaiPT);  LocPhuTung();  }  }  private bool checkEmpty()  {  if (txtHSX.Text.Trim() == "" || txtTenPT.Text.Trim() == "" || txtGiaNhap.Text.Trim() == "" || txtDonGia.Text.Trim() == "" || txtLoaiPT.Text.Trim() == "" || txtChatLuong.Text.Trim() == "" || picPhuTung.Image==null)  {  MessageBox.Show("Vui lòng nhập đầy đủ thông tin Phụ tùng.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return true;  }  else  {  return false;  }  } |

1. ***Sửa phụ tùng***

|  |
| --- |
| private void btnSuaPhuTung\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn chỉnh sửa không?", "Xác nhận chỉnh sửa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaPT.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn phụ tùng", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!Regex.IsMatch(txtGiaNhap.Text, @"^[0-9,]+$") || !Regex.IsMatch(txtDonGia.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Đơn giá và Giá nhập phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string MaPT=txtMaPT.Text;  string HSX = txtHSX.Text;  string TenPT = txtTenPT.Text;  decimal GiaNhap = Convert.ToDecimal(txtGiaNhap.Text);  decimal DonGia = Convert.ToDecimal(txtDonGia.Text);  string LoaiPT = txtLoaiPT.Text;  string ChatLuong = txtChatLuong.Text;  Image image = picPhuTung.Image;  using (MemoryStream ms = new MemoryStream())  {  image.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);  byte[] HinhAnh = ms.ToArray();  sparepart.Update(MaPT,HSX, TenPT, DonGia, GiaNhap, HinhAnh, ChatLuong, LoaiPT);  LocPhuTung();  }  }  } |

1. ***Hiển thị danh sách phụ tùng***

|  |
| --- |
| public void LocPhuTung()  {  if (chkLoc.Checked)  {  string HSX = txtLocHSX.Text;  string TenPT = txtLocTenPT.Text;  string LoaiPT = txtLocLoaiPT.Text;  string ChatLuong = txtLocChatLuong.Text;  dgvPhuTung.DataSource = sparepart.LayDS(TenPT, HSX, ChatLuong, LoaiPT);  }  else  dgvPhuTung.DataSource = sparepart.LayDS();  } |

1. ***Thêm ảnh bằng link***

|  |
| --- |
| private void btnThemAnh\_Click(object sender, EventArgs e)  {  try  {  byte[] HinhAnh = new WebClient().DownloadData(txtLinkAnh.Text);  MemoryStream stream = new MemoryStream(HinhAnh);  picPhuTung.Image = Image.FromStream(stream);  }  catch  {  MessageBox.Show("Link ảnh không đúng", "Thông Báo", MessageBoxButtons.OK, MessageBoxIcon.Error);  }  } |

1. ***Chọn ảnh có sẵn***

|  |
| --- |
| private void btnChonAnh\_Click(object sender, EventArgs e)  {  OpenFileDialog openFileDialog1 = new OpenFileDialog();  openFileDialog1.Filter = "Image files (\*.jpg, \*.jpeg, \*.jpe, \*.jfif, \*.png, \*.bmp) | \*.jpg; \*.jpeg; \*.jpe; \*.jfif; \*.png; \*.bmp";  if (openFileDialog1.ShowDialog() == DialogResult.OK)  {  string imagePath = openFileDialog1.FileName;  picPhuTung.Image = Image.FromFile(imagePath);  }  } |

1. ***Xóa ảnh***

|  |
| --- |
| private void btnXoaAnh\_Click(object sender, EventArgs e)  {  picPhuTung.Image = null;  }  private void SetupAutoComplete(TextBox textBox, string columnName,DataTable dataTable)  {  string[] columnValues = dataTable.AsEnumerable()  .Select(row => row.Field<string>(columnName))  .Distinct()  .ToArray();  AutoCompleteStringCollection autoComplete = new AutoCompleteStringCollection();  autoComplete.AddRange(columnValues);  textBox.AutoCompleteSource = AutoCompleteSource.CustomSource;  textBox.AutoCompleteCustomSource = autoComplete;  } |

1. ***Lọc phụ tùng***

|  |
| --- |
| private void chkLoc\_CheckedChanged(object sender, EventArgs e)  {  if (this.chkLoc.Checked == true)  {  this.txtLocTenPT.Enabled = true;  this.txtLocHSX.Enabled = true;  this.txtLocChatLuong.Enabled = true;  this.txtLocLoaiPT.Enabled = true;  }  else  {  this.txtLocTenPT.Clear();  this.txtLocTenPT.Enabled = false;  this.txtLocHSX.Clear();  this.txtLocHSX.Enabled = false;  this.txtLocChatLuong.Clear();  this.txtLocChatLuong.Enabled = false;  this.txtLocLoaiPT.Clear();  this.txtLocLoaiPT.Enabled = false;  }  LocPhuTung();  }  private void txtLocTenPT\_TextChanged(object sender, EventArgs e)  {  LocPhuTung();  }  private void txtLocHSX\_TextChanged(object sender, EventArgs e)  {  LocPhuTung();  }  private void txtLocChatLuong\_TextChanged(object sender, EventArgs e)  {  LocPhuTung();  }  private void txtLocLoaiPT\_TextChanged(object sender, EventArgs e)  {  LocPhuTung();  } |

1. ***Xem thông tin Phụ tùng***

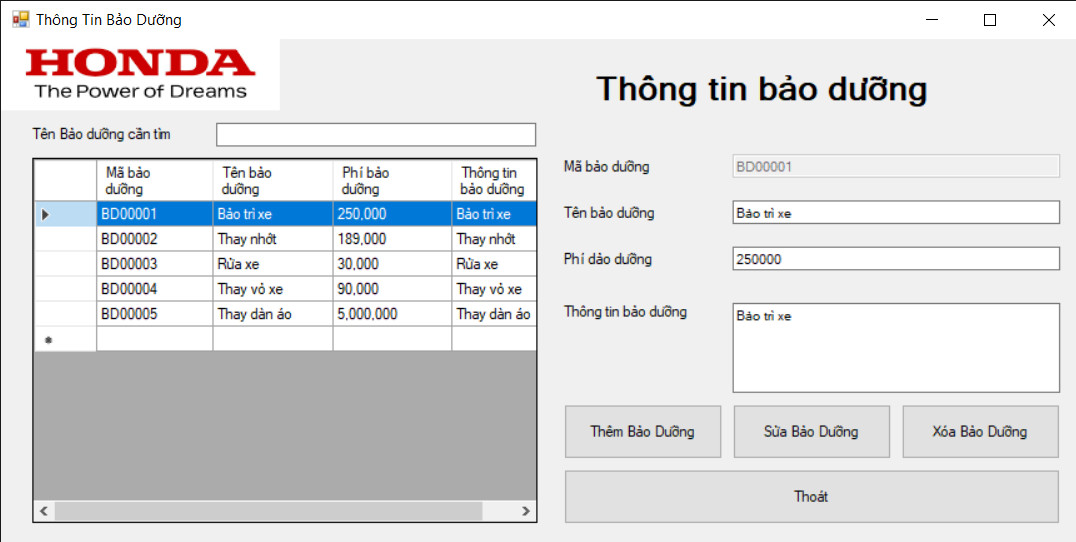
|  |
| --- |
| private void dgvPhuTung\_Click(object sender, EventArgs e)  {  Image image = picPhuTung.Image;  if (dgvPhuTung.CurrentRow != null)  {  DataGridViewRow row = dgvPhuTung.CurrentRow;  if (row.Cells["MaMH"].Value.ToString() != "")  {  txtMaPT.Text = row.Cells["MaMH"].Value.ToString();  txtHSX.Text = row.Cells["HSX"].Value.ToString();  txtTenPT.Text = row.Cells["TenMH"].Value.ToString();  txtLoaiPT.Text = row.Cells["LoaiPT"].Value.ToString();  txtChatLuong.Text = row.Cells["ChatLuong"].Value.ToString();  txtDonGia.Text = Convert.ToDecimal(row.Cells["DonGia"].Value).ToString("#");  txtGiaNhap.Text = Convert.ToDecimal(row.Cells["GiaNhap"].Value).ToString("#");  }  else  ClearPanel();  }  }  public void ClearPanel()  {  this.txtMaPT.Clear();  this.txtHSX.Clear();  this.txtTenPT.Clear();  this.txtGiaNhap.Clear();  this.txtDonGia.Clear();  this.txtChatLuong.Clear();  this.txtLoaiPT.Clear();  this.txtLinkAnh.Clear();  } |

1. ***Thoát khỏi form Thông tin phụ tùng***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. **Quản lý Bảo dưỡng**

* ***Form quản lý thông tin bảo dưỡng***

****

1. ***Thêm bảo dưỡng***

|  |
| --- |
| private void btnThemBaoDuong\_Click(object sender, EventArgs e)  {  this.txtMaBD.Clear();  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (!Regex.IsMatch(txtPhiBD.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Phí bảo dưỡng phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string TenBD = txtTenBD.Text;  decimal PhiBD = Convert.ToDecimal(txtPhiBD.Text);  string ThongTinBaoDuong = txtThongTinBaoDuong.Text;  maintenance.Insert(TenBD, PhiBD, ThongTinBaoDuong);  dgvBaoDuong.DataSource = maintenance.LayDS();  }  ClearPanel();  }  private bool checkEmpty()  {  if (txtTenBD.Text.Trim() == "" || txtPhiBD.Text.Trim() == "" || txtThongTinBaoDuong.Text.Trim() == "")  {  MessageBox.Show("Vui lòng nhập đầy đủ thông tin bảo dưỡng.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return true;  }  else  {  return false;  }  } |

1. ***Sửa bảo dưỡng***

|  |
| --- |
| private void btnSuaBaoDuong\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn chỉnh sửa không?", "Xác nhận chỉnh sửa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaBD.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn bảo dưỡng", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!Regex.IsMatch(txtPhiBD.Text, @"^[0-9,]+$"))  {  MessageBox.Show("Phí bảo dưỡng phải là 1 số", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  if (!checkEmpty())  {  string MaBD=txtMaBD.Text;  string TenBD = txtTenBD.Text;  decimal PhiBD = Convert.ToDecimal(txtPhiBD.Text);  string ThongTinBaoDuong = txtThongTinBaoDuong.Text;  maintenance.Update(MaBD,TenBD, PhiBD, ThongTinBaoDuong);  dgvBaoDuong.DataSource = maintenance.LayDS();  }  ClearPanel();  } |

1. ***Xóa bảo dưỡng***

|  |
| --- |
| private void btnXoaBaoDuong\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn xóa không?", "Xác nhận xóa", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  {  return;  }  if (txtMaBD.Text.Trim() == "")  {  MessageBox.Show("Vui lòng chọn bảo dưỡng", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);  return;  }  maintenance.Delete(txtMaBD.Text);  dgvBaoDuong.DataSource = maintenance.LayDS();  ClearPanel();  } |

1. ***Xem thông tin danh sách bảo dưỡng***

|  |
| --- |
| private void dgvBaoDuong\_Click(object sender, EventArgs e)  {  if (dgvBaoDuong.CurrentRow != null)  {  DataGridViewRow row = dgvBaoDuong.CurrentRow;  if (row.Cells["MaBD"].Value.ToString() != "")  {  txtMaBD.Text = row.Cells["MaBD"].Value.ToString();  txtTenBD.Text = row.Cells["TenBD"].Value.ToString();  txtPhiBD.Text = Convert.ToDecimal(row.Cells["PhiBD"].Value).ToString("#");  txtThongTinBaoDuong.Text = row.Cells["ThongTinBaoDuong"].Value.ToString();  }  else  ClearPanel();  }  }  public void ClearPanel()  {  this.txtMaBD.Clear();  this.txtTenBD.Clear();  this.txtPhiBD.Clear();  this.txtThongTinBaoDuong.Text = null;  }  private void txtLocBaoDuong\_TextChanged(object sender, EventArgs e)  {  dgvBaoDuong.DataSource=maintenance.LayDS(txtLocBaoDuong.Text);  } |

1. ***Thoát khỏi Form Bảo dưỡng***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. **Quản lý Hóa đơn**

* ***Form Quản lý thông tin hóa đơn***

******

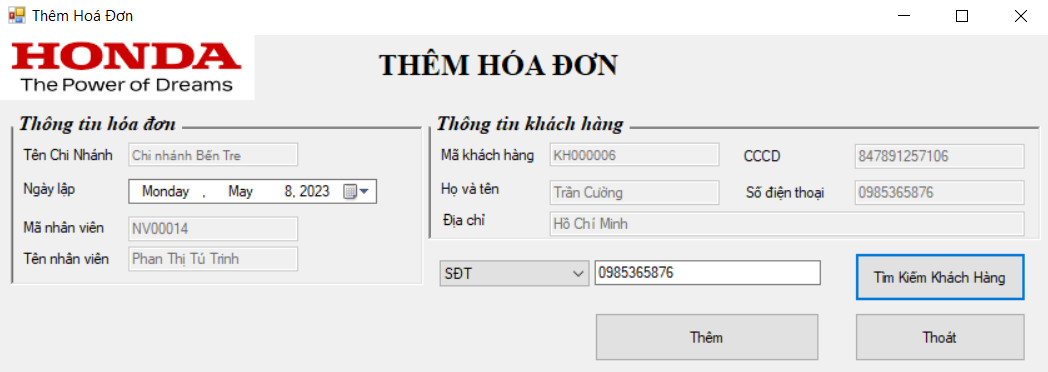
1. ***Tìm kiếm***

|  |
| --- |
| private void btnTimKiem\_Click(object sender, EventArgs e)  {    string MaNV = txtMaNV.Text;  string TenNV = txtTenNV.Text;  string TenKH = txtTenKH.Text;  string SDTKH=txtSDTKH.Text;  DateTime? BatDau=null;  DateTime? KetThuc=null;  if (chkTimKiemTheoNgay.Checked)  {  BatDau = dtpBatDau.Value.Date;  KetThuc = dtpKetThuc.Value.Date;  }  dgvHoaDon.DataSource = invoice.LayDS(MaNV, TenNV, TenKH, SDTKH, BatDau, KetThuc);  }  private void chkTimKiemTheoNgay\_CheckedChanged(object sender, EventArgs e)  {  if (this.chkTimKiemTheoNgay.Checked == true)  {  this.dtpBatDau.Enabled = true;  this.dtpKetThuc.Enabled = true;  }  else  {  this.dtpBatDau.ResetText();  this.dtpBatDau.Enabled = false;  this.dtpKetThuc.ResetText();  this.dtpKetThuc.Enabled = false;  }  } |

1. ***Thêm hóa đơn***

|  |
| --- |
| private void btnThemHoaDon\_Click(object sender, EventArgs e)  {  Form themHoaDonForm = new ThemHoaDonForm();  this.Hide();  themHoaDonForm.ShowDialog();  HoaDonForm\_Load(sender, e);  this.Show();  } |

***Form thêm hóa đơn***

******

1. ***Tìm kiếm hóa đơn***

|  |
| --- |
| private void btnTimKiem\_Click(object sender, EventArgs e)  {  DataTable table= customer.LayDS(cbbTimKiem.Text,txtTimKiem.Text);  if (table.Rows.Count > 0)  {  txtCCCD.Text = table.Rows[0]["CCCD"].ToString();  txtSDT.Text = table.Rows[0]["SDT"].ToString();  txtTenKH.Text = table.Rows[0]["TenKH"].ToString();  txtMaKH.Text = table.Rows[0]["MaKH"].ToString();  txtDiaChi.Text= table.Rows[0]["DiaChi"].ToString();  }  else  {  if (MessageBox.Show("Không tìm thấy khách hàng!\n\nCó muốn thêm khách hàng không?", "Cảnh báo", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) == DialogResult.Yes)  {  KhachHangForm khachHangForm = new KhachHangForm();  khachHangForm.ShowDialog();  }  }  } |

1. ***Thêm hóa đơn***

|  |
| --- |
| private void btnThem\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaKH.Text.Trim() == "" || txtMaNV.Text.Trim() == "")  {  MessageBox.Show("Vui Lòng Nhập đấy đủ Mã Khách Hàng và Mã Nhân Viên", "Thông Báo", MessageBoxButtons.OK, MessageBoxIcon.Information);  return;  }  string MaKH=txtMaKH.Text;  string MaNV=txtMaNV.Text;  DateTime NgayLap=dtpNgayLapHD.Value;  if (invoice.Insert(MaKH, MaNV, NgayLap))  {  Form themChiTietHoaDonForm = new ThemChiTietHoaDonForm(invoice.MaHoaDonCuoiCung());  themChiTietHoaDonForm.ShowDialog();  }  }  private void cbbTimKiem\_SelectedIndexChanged(object sender, EventArgs e)  {  this.txtTimKiem.Clear();  if (this.cbbTimKiem.SelectedIndex == 0)  this.txtTimKiem.MaxLength = 10;  else  this.txtTimKiem.MaxLength = 12;  } |

1. ***Thoát khỏi form thêm hóa đơn***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

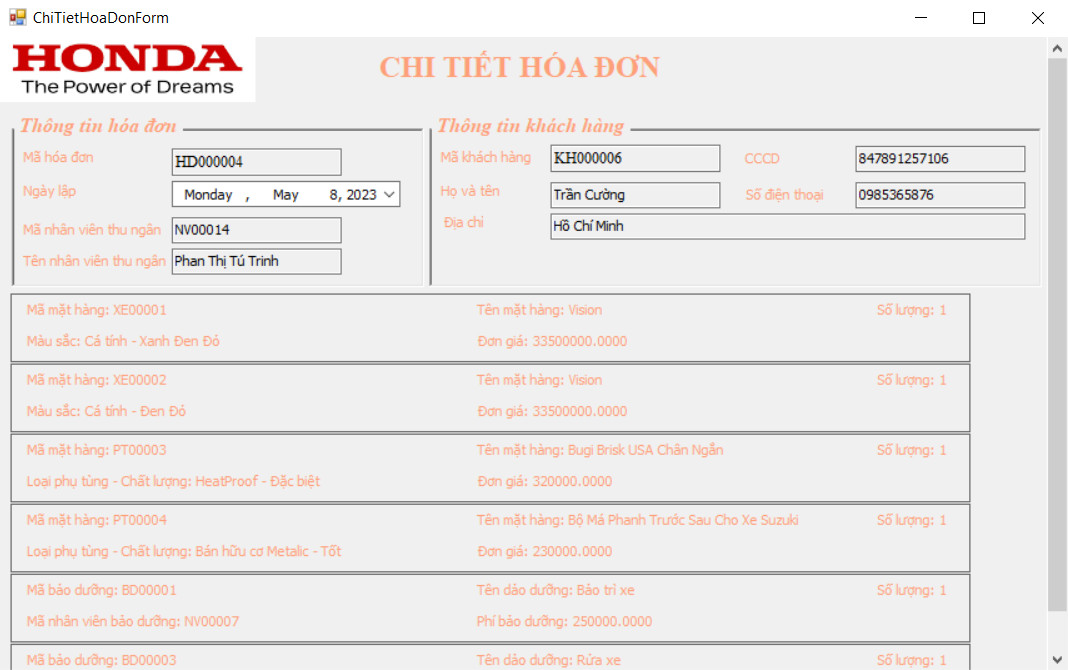
1. ***Thoát khỏi form thông tin hóa đơn***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. ***Chuột phải để xem chi tiết hóa đơn***

|  |
| --- |
| private void xemChiTiếtHóaĐơnToolStripMenuItem\_Click(object sender, EventArgs e)  {  ChiTietHoaDonForm chiTietHoaDonForm = new ChiTietHoaDonForm(dgvHoaDon.CurrentRow.Cells);  chiTietHoaDonForm.Show();  } |

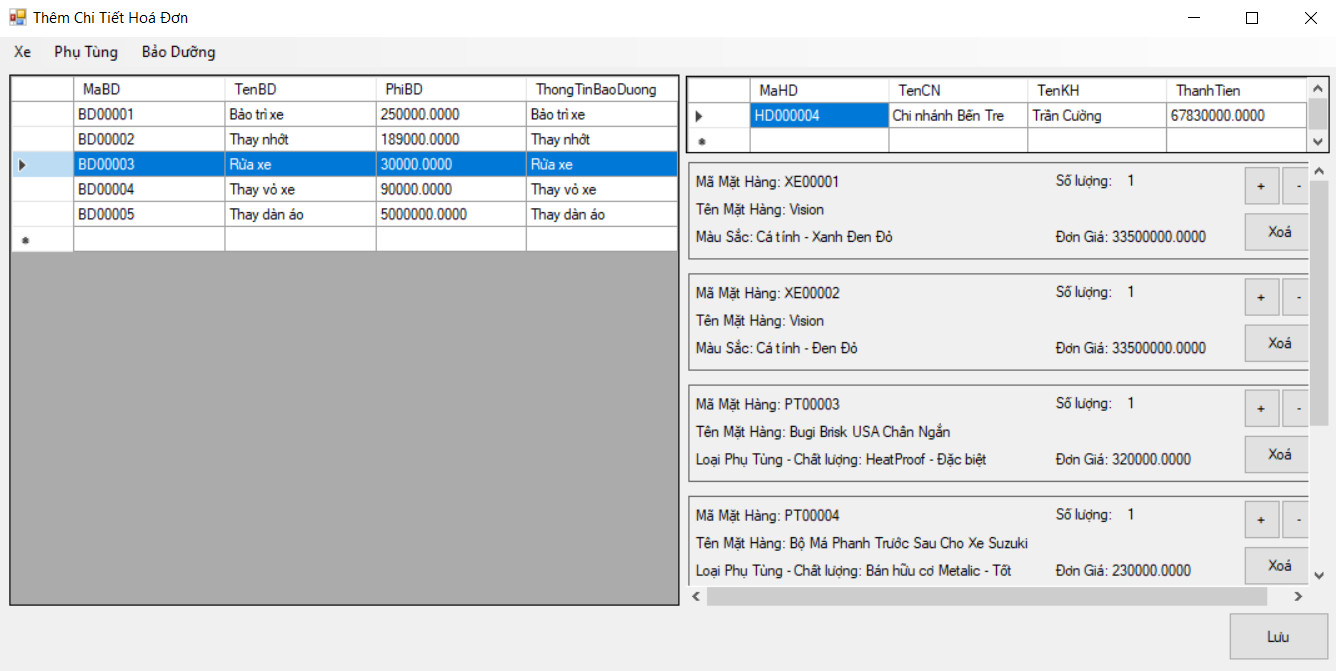
* ***Form xem chi tiết hóa đơn***

******

1. ***Chuột phải để sửa chi tiết hóa đơn***

|  |
| --- |
| private void sửaHóaĐơnToolStripMenuItem\_Click(object sender, EventArgs e)  {  ThemChiTietHoaDonForm suaChiTietHoaDonForm = new ThemChiTietHoaDonForm(dgvHoaDon.CurrentRow.Cells[0].Value.ToString());  suaChiTietHoaDonForm.ShowDialog();  } |

***- Form sửa chi tiết hóa đơn***

******

1. ***Thêm chi tiết hóa đơn***

|  |
| --- |
| private void ThemCTHD(string maMH, string tenMH, int soluong, string DonGia, string text)  {  CTHDPanel cthd;  if (maMH.Substring(0, 2) == "XE")  {  cthd = new CTHDPanel("Mã Mặt Hàng: " + maMH, "Tên Mặt Hàng: " + tenMH, soluong.ToString(), "Đơn Giá: " + DonGia, "Màu Sắc: " + text);  cthd.pnMaMH.Name = maMH;  }  else if (maMH.Substring(0, 2) == "PT")  {  cthd = new CTHDPanel("Mã Mặt Hàng: " + maMH, "Tên Mặt Hàng: " + tenMH, soluong.ToString(), "Đơn Giá: " + DonGia, "Loại Phụ Tùng - Chất lượng: " + text);  cthd.pnMaMH.Name = maMH;  }  else  {  cthd = new CTHDPanel("Mã Bảo Dưỡng: " + maMH, "Tên Bảo Dưỡng: " + tenMH, soluong.ToString(), "Phí Bảo Dưỡng: " + DonGia, "Mã Nhân Viên Bảo Dưỡng: " + text);  cthd.pnMaMH.Name = maMH;  }  cthd.Name = MaHD;  cthd.btnTang.Click += Cthd\_Click;  cthd.btnGiam.Click += Cthd\_Click;  cthd.btnXoa.Click += Cthd\_Click;  if (Ktrcthd(cthd))  {  bool a;    if(maMH.Substring(0, 2) == "BD")  a= invoice.ThemCTHDBaoDuong(MaHD, maMH,text);  else  a= invoice.ThemCTHDMatHang(MaHD, maMH);  if(a)  pnCTHD.Controls.Add(cthd);  dgvHoaDon.DataSource = invoice.LayThongTinHoaDon(MaHD);  }  } |

1. ***Kiểm tra chi tiết hóa đơn***

|  |
| --- |
| private bool Ktrcthd(CTHDPanel cthd)  {  foreach (CTHDPanel control in pnCTHD.Controls)  {  if (control == cthd)  {  bool a;  if(control.pnMaMH.Name.Substring(0,2)=="BD")  a= invoice.SuaCTHDBaoDuong(MaHD, control.pnMaMH.Name, control.lbl.Text.Substring(control.lbl.Text.LastIndexOf(' ') + 1), int.Parse(control.lblSoLuong.Text)+1);  else  a=invoice.SuaCTHDMatHang(MaHD, control.pnMaMH.Name, int.Parse(control.lblSoLuong.Text)+1);  if(a)  control.lblSoLuong.Text = (int.Parse(control.lblSoLuong.Text) + 1).ToString();  dgvHoaDon.DataSource = invoice.LayThongTinHoaDon(MaHD);  return false;  }  }  return true;  } |

1. ***Sửa chi tiết hóa đơn***

|  |
| --- |
| private void SuaChiTietHoaDonForm\_Load(object sender, EventArgs e)  {  dgvHoaDon.DataSource = invoice.LayThongTinHoaDon(MaHD);  DataSet dataSet = invoice.LayChiTietHoaDon(MaHD);  foreach (DataRow row in dataSet.Tables["Xe"].Rows)  {  CTHDPanel cthd = new CTHDPanel("Mã Mặt Hàng: " + row["MaMH"].ToString(), "Tên Mặt Hàng: " + row["TenMH"].ToString(), row["SoLuong"].ToString(), "Đơn Giá: " + row["DonGia"].ToString(), "Màu Sắc: " + row["MauSac"].ToString());  cthd.pnMaMH.Name = row["MaMH"].ToString();  cthd.Name = MaHD;  cthd.btnTang.Click += Cthd\_Click;  cthd.btnGiam.Click += Cthd\_Click;  cthd.btnXoa.Click += Cthd\_Click;  pnCTHD.Controls.Add(cthd);  }  foreach (DataRow row in dataSet.Tables["PhuTung"].Rows)  {  CTHDPanel cthd = new CTHDPanel("Mã Mặt Hàng: " + row["MaMH"].ToString(), "Tên Mặt Hàng: " + row["TenMH"].ToString(), row["SoLuong"].ToString(), "Đơn Giá: " + row["DonGia"].ToString(), "Loại Phụ Tùng: " + row["LoaiPT"].ToString() + " - Chất lượng: " + row["ChatLuong"].ToString());  cthd.Name = MaHD;  cthd.pnMaMH.Name = row["MaMH"].ToString();  pnCTHD.Controls.Add(cthd);  cthd.btnTang.Click += Cthd\_Click;  cthd.btnGiam.Click += Cthd\_Click;  cthd.btnXoa.Click += Cthd\_Click;  }  foreach (DataRow row in dataSet.Tables["BaoDuong"].Rows)  {  CTHDPanel cthd = new CTHDPanel("Mã Bảo Dưỡng: " + row["MaBD"].ToString(), "Tên Bảo Dưỡng: " + row["TenBD"].ToString(), row["SoLuong"].ToString(), "Phí Bảo Dưỡng: " + row["PhiBD"].ToString(), "Mã Nhân Viên Bảo Dưỡng: " + row["MaNVBaoDuong"].ToString());  cthd.Name = MaHD;  cthd.pnMaMH.Name = row["MaBD"].ToString();  pnCTHD.Controls.Add(cthd);  cthd.btnTang.Click += Cthd\_Click;  cthd.btnGiam.Click += Cthd\_Click;  cthd.btnXoa.Click += Cthd\_Click;  }} |

1. ***Bảng xem chi tiết hóa đơn***

|  |
| --- |
| private void Cthd\_Click(object sender, EventArgs e)  {  dgvHoaDon.DataSource = invoice.LayThongTinHoaDon(MaHD);  } |

1. ***Bảng xem chi tiết mặt hàng***

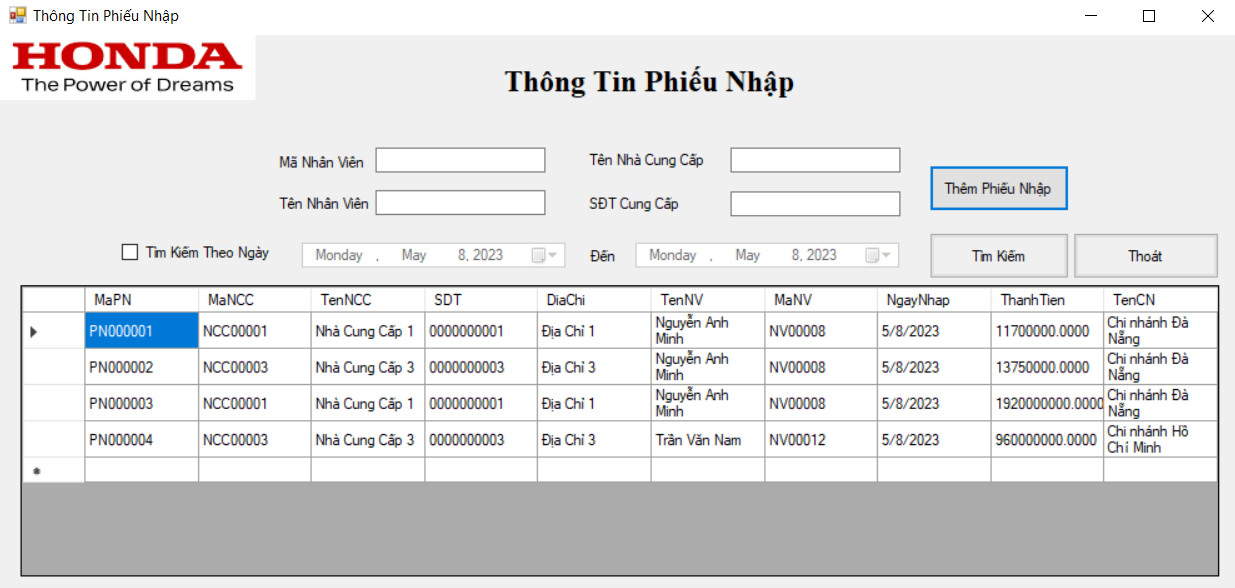
|  |
| --- |
| private void dgvMH\_Click(object sender, EventArgs e)  {  try  {  DataGridViewRow selectedRow = dgvMH.CurrentRow;  string MaMH = selectedRow.Cells[0].Value.ToString();  string TenMH = selectedRow.Cells[1].Value.ToString();  string text = "";  string DonGia = selectedRow.Cells[2].Value.ToString();  if (MaMH.Substring(0, 2) == "XE")  text = selectedRow.Cells["MauSac"].Value.ToString();  else  if (MaMH.Substring(0, 2) == "PT")  text = selectedRow.Cells["LoaiPT"].Value.ToString() + " - " + selectedRow.Cells["ChatLuong"].Value.ToString();  else  {  Form form = new Form();  DataGridView dataGridView = new DataGridView();  dataGridView.DataSource = employee.LayDS("","Nhân viên bảo dưỡng","","");  dataGridView.Dock = DockStyle.Fill;  form.Controls.Add(dataGridView);  dataGridView.Click += (sender1, e1) =>  {  text = dataGridView.CurrentRow.Cells["MaNV"].Value.ToString();  form.Close();  };  form.ShowDialog();  }  if(text.Trim()!="")  ThemCTHD(MaMH, TenMH, 1, DonGia, text);  }  catch  {  MessageBox.Show("Không có dữ liệu", "Thông báo");  }  }  private void xeSốToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = moto.LayDSTrenHoaDon("Xe Số");  }  private void xeTayGaToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = moto.LayDSTrenHoaDon("Xe Tay Ga");  }  private void xeCônTayToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = moto.LayDSTrenHoaDon("Xe Côn Tay");  }  private void tấtCảXeToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = moto.LayDSTrenHoaDon();  }  private void tấtCảPhụTùngToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = sparepart.LayDSTrenHoaDon();  }  private void tấtCảBảoDưỡngToolStripMenuItem\_Click(object sender, EventArgs e)  {  dgvMH.DataSource = maintenance.LayDS();  } |

1. ***Lưu thông tin và thoát khỏi form chi tiết hóa đơn***

|  |
| --- |
| private void btnLuu\_Click(object sender, EventArgs e)  {  this.Close();  } |

**10. Quản lý Phiếu nhập**

***- Form quản lý thông tin Phiếu nhập***

******

1. ***Tìm kiếm thông tin phiếu nhập***

|  |
| --- |
| rivate void btnTimKiem\_Click(object sender, EventArgs e)  {  string MaNV = txtMaNV.Text;  string TenNV = txtTenNV.Text;  string TenNCC = txtTenNCC.Text;  string SDT = txtSDT.Text;  DateTime? BatDau = null;  DateTime? KetThuc = null;  if (chkTimKiemTheoNgay.Checked)  {  BatDau = dtpBatDau.Value.Date;  KetThuc = dtpKetThuc.Value.Date;  }  dgvPhieuNhap.DataSource = receipt.LayDS(MaNV, TenNV, TenNCC,SDT,BatDau,KetThuc);  } |

1. ***Xem thông tin phiếu nhập***

|  |
| --- |
| private void dgvPhieuNhap\_Click(object sender, EventArgs e)  {  Form chiTietPhieuNhapForm = new ChiTietPhieuNhapForm(dgvPhieuNhap.CurrentRow.Cells);  chiTietPhieuNhapForm.ShowDialog();  PhieuNhapForm\_Load(sender, e);  }  private void btnThemPhieuNhap\_Click(object sender, EventArgs e)  {  Form themPhieuNhapForm = new ThemPhieuNhapForm();  this.Hide();  themPhieuNhapForm.ShowDialog();  PhieuNhapForm\_Load(sender, e);  this.Show();  } |

1. ***Tìm thông tin phiếu nhập***

|  |
| --- |
| private void chkTimKiemTheoNgay\_CheckedChanged(object sender, EventArgs e)  {  if (this.chkTimKiemTheoNgay.Checked == true)  {  this.dtpBatDau.Enabled = true;  this.dtpKetThuc.Enabled = true;  }  else  {  this.dtpBatDau.ResetText();  this.dtpBatDau.Enabled = false;  this.dtpKetThuc.ResetText();  this.dtpKetThuc.Enabled = false;  }  } |

1. ***Thoát khỏi form phiếu nhập***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. ***Thêm phiếu nhập***

|  |
| --- |
| private void btnThemPhieuNhap\_Click(object sender, EventArgs e)  {  Form themPhieuNhapForm = new ThemPhieuNhapForm();  this.Hide();  themPhieuNhapForm.ShowDialog();  PhieuNhapForm\_Load(sender, e);  this.Show();  } |

******

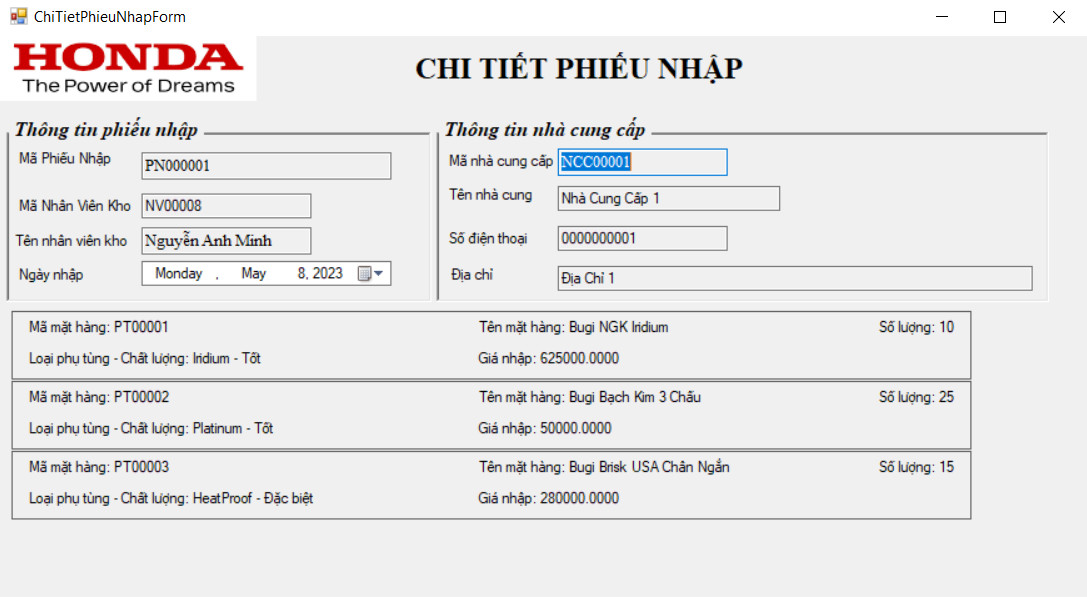
1. ***Thêm phiếu nhập***

|  |
| --- |
| private void btnThemPhieuNhap\_Click(object sender, EventArgs e)  {  if (MessageBox.Show("Bạn có chắc chắn muốn thêm mới không?", "Xác nhận thêm mới", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)  return;  if (txtMaNCC.Text.Trim() == "" || txtMaNV.Text.Trim() == "")  {  MessageBox.Show("Vui Lòng Nhập đấy đủ Mã Nhà Cung và Mã Nhân Viên", "Thông Báo", MessageBoxButtons.OK, MessageBoxIcon.Information);  return;  }  string MaNCC = txtMaNCC.Text;  string MaNV = txtMaNV.Text;  DateTime NgayNhap = dtpNgayNhap.Value;  if (receipt.Insert(MaNCC, MaNV, NgayNhap))  {  Form themChiTietPhieuNhapForm = new ThemChiTietPhieuNhapForm(receipt.MaPhieuNhapCuoiCung());  themChiTietPhieuNhapForm.ShowDialog();  }  } |

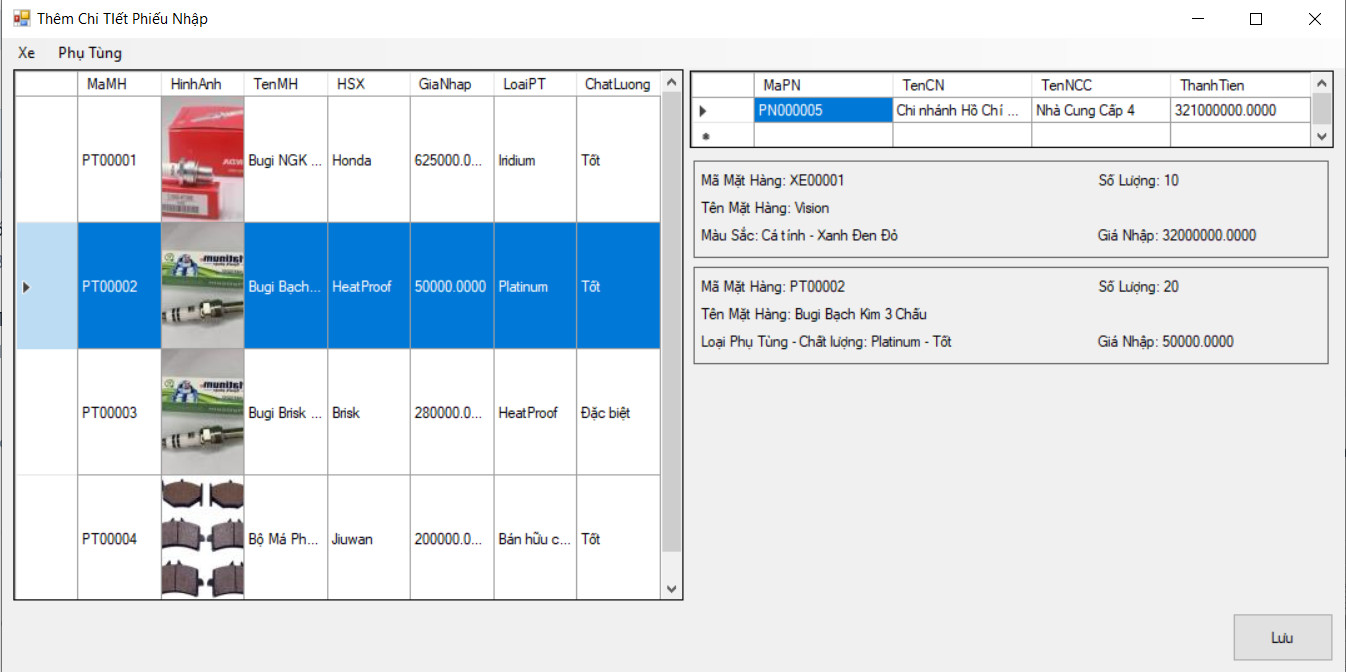
1. ***Thoát khỏi form thêm phiếu nhập***

|  |
| --- |
| private void btnThoat\_Click(object sender, EventArgs e)  {  this.Close();  } |

1. ***Form xem chi tiết phiếu nhập***

******

1. ***Form sửa chi tiết phiếu nhập***

******

1. ***Thêm chi tiết phiếu nhập***

|  |
| --- |
| private void ThemCTPN(string maMH, string tenMH, int soluong, string giaNhap, string text)  {  CTPNPanel ctpn;  if(maMH.Substring(0,2)=="XE")  ctpn = new CTPNPanel("Mã Mặt Hàng: " + maMH, "Tên Mặt Hàng: " + tenMH, "Số Lượng: "+soluong , "Giá Nhập: " + giaNhap, "Màu Sắc: " + text);  else  ctpn = new CTPNPanel("Mã Mặt Hàng: " + maMH, "Tên Mặt Hàng: " + tenMH, "Số Lượng: " + soluong, "Giá Nhập: " + giaNhap, "Loại Phụ Tùng - Chất lượng: " + text);  ctpn.Location = new Point(0, y);  if (Ktrctpn(ctpn))  {  pnCTPN.Controls.Add(ctpn);  receipt.ThemChiTietPhieuNhap(MaPN, maMH, soluong);  dgvPhieuNhap.DataSource = receipt.LayThongTinPhieuNhap(MaPN);  y += ctpn.Height;  }  } |

1. ***Kiểm tra phiếu nhập***

|  |
| --- |
| private bool Ktrctpn(CTPNPanel ctpn)  {  foreach (CTPNPanel control in pnCTPN.Controls)  {  if (control == ctpn)  {  return false;  }  }  return true;  } |

1. ***Bảng xem thông tin mặt hàng***

|  |
| --- |
| private void dgvMH\_Click(object sender, EventArgs e)  {  try  {  int soluong = 0;  Form form = new Form();  form.Size = new Size(210, 110);  Label label = new Label();  label.Text = "Số lượng:";  label.Location = new Point(10, 10);  label.AutoSize = true;  TextBox textBox = new TextBox();  textBox.Location = new Point(80, 10);  Button okButton = new Button();  okButton.Text = "OK";  okButton.Location = new Point(80, 40);  okButton.Click += (sender1, e1) =>  {  if (textBox.Text.All(c => char.IsDigit(c)))  {  soluong = Convert.ToInt32(textBox.Text);  if (soluong <= 0)  {  MessageBox.Show("Số lượng phải hơn 0");  return;  }  form.Close();  form.DialogResult = DialogResult.OK;  }  else  {  MessageBox.Show("Số lượng phải là số");  }  };  form.Controls.Add(label);  form.Controls.Add(textBox);  form.Controls.Add(okButton);  form.ShowDialog();  DataGridViewRow selectedRow = dgvMH.CurrentRow;  string MaMH = selectedRow.Cells["MaMH"].Value.ToString();  string TenMH = selectedRow.Cells["TenMH"].Value.ToString();  string text = "";  string GiaNhap = selectedRow.Cells["GiaNhap"].Value.ToString();  if (MaMH.Substring(0, 2) == "XE")  text = selectedRow.Cells["MauSac"].Value.ToString();  else  text = selectedRow.Cells["LoaiPT"].Value.ToString() + " - " + selectedRow.Cells["ChatLuong"].Value.ToString();  ThemCTPN(MaMH, TenMH, soluong, GiaNhap, text);  }  catch  {  MessageBox.Show("Không có dữ liệu", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);  }  } |

1. ***Lưu và thoát khỏi form***

|  |
| --- |
| private void btnLuu\_Click(object sender, EventArgs e)  {  this.Close();  } |