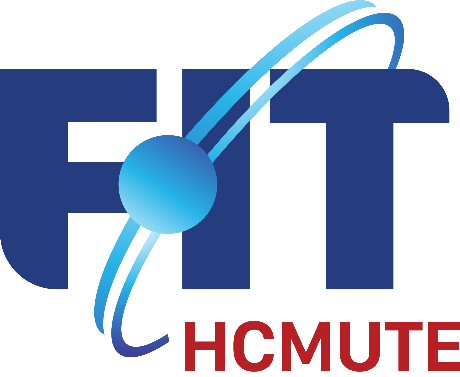
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

---------------



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

**TÊN ĐỀ TÀI:**

**QUẢN LÝ CHUỖI CỬA HÀNG XE ÔTÔ**

**GVHD:** *TS. Nguyễn Thành Sơn*

**Lớp HP: DBMS330284\_23\_1\_03**

**Nhóm thực hiện:** *Nhóm 03*

**Học kỳ: 1**

**Năm học: 2023 – 2024**

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

**DANH SÁCH SINH VIÊN NHÓM THỰC HIỆN**

**HỌC KÌ 1 NĂM HỌC 2023-2024**

**Nhóm 03**

Đề tài: Quản lý chuỗi cửa hàng xe ôtô

|  |  |
| --- | --- |
| 21110900 | Sú Minh Luân |
| 21110512 | Nguyễn Việt Khoa |
| 21110927 | Nguyễn Xuân Thể |
| 21110665 | Đặng Gia Thuận |

**NHẬN XÉT CỦA GV**

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

Giảng viên chấm điểm

Contents

[CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG 4](#_Toc30025)

[1. Đặc tả bài toán 4](#_Toc9034)

[2. Nghiệp vụ của bài toán 4](#_Toc28572)

[3. Chức năng bài toán 6](#_Toc25172)

[4. Phân quyền 7](#_Toc13082)

[5. Thuế nhập xe - nhập phụ tùng 9](#_Toc30252)

[6. Thống kê (Doanh thu, lợi nhuận) 11](#_Toc15415)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 12](#_Toc28162)

[1. Thiết kế cơ sở dữ liệu mức quan niệm 12](#_Toc30664)

[2. Thiết kế cơ sở dữ liệu mức logic 13](#_Toc15411)

[3. Các ràng buộc cần có 15](#_Toc19496)

[4. Cài đặt CSDL và các ràng buộc 17](#_Toc15702)

[5. Các ràng buộc khác 26](#_Toc7319)

[6. Các View 28](#_Toc25767)

[7. Các Function 35](#_Toc8302)

[8. Stored Procedure 42](#_Toc8417)

[8.1 Chi Nhánh 42](#_Toc29986)

[8.2 Hóa Đơn 44](#_Toc25793)

[8.3 Chi Tiết Hóa Đơn Phụ Tùng 46](#_Toc18732)

[8.4 Chi Tiết Hóa Đơn Xe 49](#_Toc22730)

[8.5 Phiếu Nhập 51](#_Toc12101)

[8.6 Chi Tiết Phiếu Nhập Phụ Tùng 53](#_Toc10242)

[8.7 Chi Tiết Phiếu Nhập Xe 55](#_Toc25771)

[8.8 Nhà Cung Cấp 57](#_Toc10178)

[8.9 Phụ Tùng 59](#_Toc27194)

[8.10 Dịch Vụ Bảo Dưỡng 61](#_Toc23241)

[8.11 Chi Tiết Phiếu Bảo Dưỡng 64](#_Toc4001)

[8.12 Khách Hàng 66](#_Toc4665)

[9. Kết nối cơ sở dữ liệu 68](#_Toc18690)

# CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG

## Đặc tả bài toán

- Cửa hàng ô tô là nơi bán các loại xe ô tô con phục vụ mục đích đi lại, vận chuyển hàng hóa của khách hàng. Sản phẩm của cửa hàng có đa dạng thương hiệu, màu sắc, đáp ứng được sở thích cũng như nhu cầu của nhiều khách hàng. Ngoài việc bán các loại xe ô tô con, cửa hàng còn cung cấp dịch vụ bảo dưỡng và bảo hành xe.

- Bên cạnh quản lý các loại xe và phụ tùng, cửa hàng còn phải quản lý thêm số lượng lớn nhân viên với nhiều chức vụ khác nhau:

* Nhân viên tư vấn - bán hàng sẽ hỗ trợ tư vấn các dịch vụ của cửa hàng và mô tả thông tin mặt hàng cho khách hàng có nhu cầu
* Nhân viên bảo dưỡng/bảo hành sẽ phụ trách các dịch vụ bảo dưỡng/bảo hành xe cho khách hàng
* Nhân viên quản lý kho hàng có nhiệm vụ nhập hàng từ nhà cung cấp và lập báo cáo về số lượng
* Nhân viên thu ngân sẽ tạo lập hóa đơn thanh toán, đồng thời lưu thông tin khách hàng và mặt hàng hoặc dịch vụ mà khách hàng đã yêu cầu
* Người quản lý: phụ trách việc quản lý toàn bộ nhân viên và các hoạt động tại chi nhánh làm việc
* Mỗi nhân viên của cửa hàng được nhận biết thông qua mã nhân viên và các thông tin như tên nhân viên, số CCCD, ngày sinh, giới tính, địa chỉ, số điện thoại, chức vụ, tình trạng làm việc, mã chi nhánh nơi làm việc, hình ảnh.

- Mỗi nhân viên sẽ được cấp một tài khoản để sử dụng ứng dụng của cửa hàng. Tài khoản bao gồm: tên đăng nhập, mật khẩu, chức vụ, mã nhân viên.

- Cửa hàng xe ô tô có nhiều chi nhánh. Mỗi chi nhánh có mã chi nhánh để nhận biết, tên chi nhánh, địa chỉ của chi nhánh. Mỗi chi nhánh quản lý nhiều nhân viên.

- Cửa hàng sẽ nhập xe và phụ tùng từ nhiều nhà cung cấp khác nhau cho mỗi chi nhánh của cửa hàng. Mỗi nhà cung cấp có: mã nhà cung cấp, tên nhà cung cấp, số điện thoại, địa chỉ.

- Sau khi nhập xe và phụ tùng về cửa hàng, chúng sẽ được vận chuyển vào kho. Mỗi cửa hang gồm có 2 kho:

+ Kho xe gồm mã chi nhánh, mã lô xe, số lượng xe còn, số lượng xe đã bán

+ Kho phụ tùng gồm mã chi nhánh, mã phụ tùng, số lượng phụ tùng còn, số lượng phụ tùng đã bán

- Cửa hàng kinh doanh nhiều loại xe, thuộc nhiều hãng khác nhau, cho khách hàng lựa chọn. Để quản lý cửa hàng một cách thuận tiện, cửa hàng sẽ nhập xe từ nhà cũng cấp theo các lô. Thông tin mỗi lô xe bao gồm: Mã lô xe, tên xe, phiên bản xe, loại xe, hãng xe, xuất xứ, số chỗ, giá xe bán, màu xe, động cơ, kích thước, trọng lượng, trọng tải, tốc độ tối đa, hệ thống phanh, công nghệ an toàn, hình ảnh.

* Động cơ bao gồm: Loại động cơ, dung tích (cc), công suất (mã lực)/vòng tua(vòng/phút), mô-men xoắn(Nm)/vòng tua(vòng/phút), loại nhiên liệu.
* Kích thước: Chiều dài, chiều rộng, chiều cao, chiều dài cơ sở, khoảng sáng gầm, bán kính quay vòng
* Hệ thống phanh bao gồm: bộ điều khiển phanh, bộ phận truyền lực phanh, phanh trước, phanh sau
* Công nghệ an toàn bao gồm: cảnh báo phương tiện, móc ghế an toàn, túi khí, cảnh báo điểm mù, cảm biến lùi, camera lùi.

**-** Với mỗi chiếc xe cụ thể trong từng lô xe sẽ được đánh mã số xe riêng biệt và mã lô xe tương ứng.

- Phụ tùng xe gồm có: Mã phụ tùng, loại phụ tùng, tên phụ tùng, thương hiệu, xuất xứ, giá, chất lượng, hình ảnh.

- Mỗi khi cửa hàng nhập xe hoặc phụ tùng, cửa hàng phải lưu trữ phiếu nhập để tiện cho việc thống kê chi tiêu. Thông tin phiếu nhập bao gồm: mã phiếu nhập, mã nhà cung cấp, mã chi nhánh, ngày nhập. Đối với mỗi mặt hàng như xe hoặc phụ tùng thì sẽ có chi tiết phiếu nhập riêng.

* Chi tiết phiếu nhập xe: Mã chi tiết phiếu nhập xe, Mã phiếu nhập, mã lô xe, giá nhập, số lượng.
* Chi tiết phiếu nhập phụ tùng: Mã chi tiết phiếu nhập phụ tùng, Mã phiếu nhập, mã phụ tùng, giá nhập, số lượng.

- Khi khách hàng đến chi nhánh để sử dụng dịch vụ, thông tin của khách hàng sẽ được lưu vào hệ thống chung của doanh nghiệp. Thông tin của khách hàng gồm: mã khách hàng, họ tên khách hàng, ngày sinh, giới tính, số điện thoại, địa chỉ, số CCCD.

- Mỗi khi khách hàng mua xe từ cửa hàng thì sẽ có hóa đơn thanh toán. Mỗi hóa đơn thanh toán bao gồm các thông tin: Mã hóa đơn, mã khách hàng, mã nhân viên, ngày lập hóa đơn, tình trạng hóa đơn, tổng tiền. Ngoài thông trên hóa đơn còn có các thông tin chi tiết như: Mã hóa đơn, mã xe, ngày nhận xe, các loại phí, số tiền đã trả.

* Các loại phí bao gồm: phí trước bạ, phí sử dụng đường bộ, bảo hiểm trách nhiệm dân sự, phí đăng kí biển số, phí đăng kiểm.

- Khi mua xe khách hàng sẽ được ký hợp đồng. Hợp đồng bảo hành gồm: Mã hợp đồng, mã xe, ngày ký bảo hành, thời hạn bảo hành, tình trạng bảo hành.

- Khi có hư hại trong phạm vi được bảo hành và xe còn trong thời hạn bảo hành, thì khách hàng sẽ được hỗ trợ bảo hành. Phiếu bảo hành gồm các thông tin như: Mã phiếu bảo hành, mã hợp đồng, mã nhân viên thực hiện, ngày nhận, ngày trả xe. Thời hạn bảo hành là từ 2 đến 4 năm tùy theo từng loại xe.

- Cửa hàng cung cấp dịch vụ bảo dưỡng cho khách hàng khi mua xe thuộc các chi nhánh của doanh nghiệp. Bảo dưỡng xe có các thông tin sau: Mã bảo dưỡng, tên bảo dưỡng, phí bảo dưỡng, loại bảo dưỡng.

- Mỗi khi khách hàng có sử dụng dịch vụ bảo dưỡng, cửa hàng sẽ xuất phiếu bảo dưỡng. Thông tin trên phiếu bảo dưỡng bao gồm: mã phiếu bảo dưỡng, mã khách hàng, mã nhân viên bảo dưỡng, tổng tiền, ngày lập phiếu. Và sẽ xuất hóa đơn bảo dưỡng cho từng dịch vụ bảo dưỡng, bao gồm: Mã phiếu, mã bảo dưỡng, thành tiền.

## Nghiệp vụ của bài toán

- Bài toán quản lý cửa hàng xe ô tô là một nghiệp vụ quản lý kinh doanh trong lĩnh vực mua bán và kỹ thuật ô tô. Nó liên quan tới việc quản lý nhân viên, quản lý hàng hóa, doanh thu và điều hành chuỗi cửa hàng ô tô. Nhân viên là cốt lõi, là người đại diện cho cửa hàng để giao tiếp và cung cấp các dịch vụ cho khách hàng. Vì vậy chúng ta cần xây dựng một đội ngũ nhân viên được đào tạo chuyên nghiệp về các quy trình tác nghiệp, văn hóa phục vụ, cách ứng xử, xử lý trong mọi tình huống…

- Quản lý nhân viên: Khi tuyển dụng nhân sự, cửa hàng cần đưa ra một số quy định (số ngày làm mỗi tuần, giờ làm mỗi ngày, không được nghỉ quá số ngày quy định, số buổi đi trễ cho phép…) và nhân viên khi ứng tuyển phải đồng ý với các quy định trên. Nhân viên khi được tuyển sẽ trải qua khóa đào tạo về nghiệp vụ, quy trình làm việc ở vị trí của nhân viên đó. Người quản lý theo dõi và giám sát các nhân sự, góp ý những điểm thiếu sót và đưa ra lời khen để làm động lực cho nhân viên.

* Quy trình khi khách đến cửa hàng

**Bước 1:** Tiếp đón khách hàng: Nhân viên bán hàng cần chào đón khách hàng một cách thân thiện và niềm nở, hỏi thăm nhu cầu của khách hàng.

**Bước 2:** Tư vấn khách hàng: Nhân viên bán hàng cần tư vấn cho khách hàng về các loại xe, phụ tùng với giá cả, các chương trình khuyến mãi,... để khách hàng có thể lựa chọn được chiếc xe, phụ tùng xe phù hợp với nhu cầu và ngân sách của mình.

**Bước 3:** Cho khách hàng lái thử xe: Nếu khách hàng mua xe, cho khách hàng lái thử xe là một cách hiệu quả để khách hàng có thể trải nghiệm trực tiếp chiếc xe và đưa ra quyết định mua hàng. Đối với trường hợp hư hỏng sẽ có biện pháp xử lý thích hợp.

**Bước 4:** Thương lượng giá cả: Nếu khách hàng đã chọn được chiếc xe hay loại phụ tùng, nhân viên bán hàng sẽ thương lượng giá cả với khách hàng. Sau khi chốt được giá với khách hàng, nhân viên sẽ hỏi khách có thẻ thành viên của cửa hàng chưa, nếu chưa thì xin thông tin của khách để tạo thẻ thành viên.

**Bước 5:** Hỗ trợ khách hàng làm thủ tục mua bán, sử dụng dịch vụ:

- Nếu khách hàng đồng ý mua mặt hàng, nhân viên bán hàng sẽ hỗ trợ khách hàng làm các thủ tục mua, bao gồm:

+ Ký hợp đồng mua bán xe

+ Lập hồ sơ vay vốn (nếu khách hàng mua xe trả góp)

+ Đăng ký xe

+ Bàn giao xe

- Nếu khách hàng xác định sử dụng dịch vụ, thì nhân viên sẽ hỗ trợ khách hàng làm thủ tục

**Bước 6:** Chăm sóc khách hàng sau bán hàng

Sau khi khách hàng mua mặt hàng, nhân viên bán hàng cần tiếp tục chăm sóc khách hàng, giải đáp thắc mắc và hỗ trợ khách hàng trong quá trình sử dụng.

- Quản lý hàng hóa, quản lý doanh thu và điều hành cửa hàng bán ô tô có thể chia thành 3 giai đoạn chính:

* Giai đoạn 1: Lập kế hoạch

Trong giai đoạn này, chủ cửa hàng cần xác định các mục tiêu kinh doanh cụ thể, như tăng doanh thu, mở rộng thị trường, nâng cao chất lượng dịch vụ,... Từ đó, xây dựng kế hoạch quản lý hàng hóa, doanh thu và điều hành cửa hàng phù hợp.

* Giai đoạn 2: Thực hiện kế hoạch

- Quản lý hàng hóa:

Cửa hàng cần triển khai các kế hoạch đã đề ra. Cụ thể:

+ Nhập hàng: Đảm bảo chất lượng, số lượng, chủng loại hàng hóa phù hợp với nhu cầu thị trường.

+ Lưu kho: Sắp xếp khoa học, hợp lý để dễ dàng tìm kiếm, kiểm kê.

+ Xuất hàng: Phân phối hàng hóa đến tay khách hàng một cách nhanh chóng, chính xác.

- Quản lý doanh thu:

+ Theo dõi doanh thu theo từng tháng, quý, năm.

+ Phân tích doanh thu để xác định các yếu tố ảnh hưởng đến doanh thu.

+ Đưa ra các giải pháp để tăng doanh thu.

- Điều hành cửa hàng:

+ Quản lý nhân sự: Tuyển dụng, đào tạo, đánh giá, khen thưởng, kỷ luật nhân viên.

+ Quản lý tài chính: Theo dõi thu chi, báo cáo tài chính.

+ Quản lý khách hàng: Xây dựng mối quan hệ tốt đẹp với khách hàng, chăm sóc khách hàng sau bán hàng.

* Giai đoạn 3: Đánh giá, điều chỉnh

Trong giai đoạn này, chủ cửa hàng cần đánh giá hiệu quả của các hoạt động quản lý hàng hóa, doanh thu và điều hành cửa hàng, tiếp thu các ý kiến phản hồi từ khách hàng về cửa hàng. Từ đó, có những điều chỉnh kịp thời để đảm bảo hoạt động kinh doanh hiệu quả.

## Chức năng bài toán

- Quản lý Phụ tùng xe ô tô

+ Thêm, sửa, xóa thông tin của phụ tùng (mã, tên, số lượng,...)

+ Tìm kiếm theo thông tin phụ tùng

+ Quản lý số lượng, hàng tồn kho, thời gian tồn kho của phụ tùng

+ Quản lý số lượng bán ra của phụ tùng

+ Quản lý giá bán của phụ tùng

- Quản lý bán xe

+ Thêm, sửa, xóa thông tin chi tiết xe hơi (mã, tên, số lượng,...)

+ Tìm kiếm theo 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 thông tin của nhân viên (mã, họ tên,...)

+ Tìm kiếm theo thông tin nhân viên

+ Phân quyền theo nhóm

+ Lọc thông tin nhân viên

- Quản lý khách hàng

+ Thêm, sửa dữ liệu thông tin khách hàng

+ Tìm kiếm theo thông tin khách hàng

+ Thông kê sử dụng dịch vụ của khách hàng

- Quản lý mua hàng

+ Xuất hóa đơn

+ Quản lý chi tiết hóa đơn

- Quản lý tài chính

+ Thống kê chi thu của từng chi nhánh theo tháng, quý, năm(có thể chọn tháng, quý, năm tương ứng), thống kê theo từng loại xe, hãng xe

## Phân quyền

- Phân quyền trong bài toán quản lý cửa hàng bán xe ô tô là cấp quyền và truy cập của nhân viên đến các tài nguyên và thông tin cửa hàng cũng như các thông tin về vật tư.

- Việc phân quyền này đảm bảo an toàn thông tin của khách hàng, an toàn dữ liệu với tài sản của cửa hàng. Tránh làm lộ thông tin, sai sót trong quá trình quản lý và sử dụng

- Thông qua chức vụ, trình độ và một số yếu tố mà quyền hạn của nhân viên trong cửa hàng xe sẽ được phân ra khác nhau. Cụ thể như sau:

1. Nhân viên tư vấn - bán hàng

Những nhân viên này sẽ có quyền truy cập thông tin cơ bản của cửa hàng, như là:

- Số lượng hàng hóa còn trong cửa hàng, giá cả của các hàng hóa này, cũng như là số lượng ở các chi nhánh và nhà cung cấp.

- Quyền yêu cầu nhập thêm các hàng hóa tương ứng.

- Quyền truy cập thông tin của khách hàng.

Từ đó dễ dàng nắm bắt những thông tin mới nhất, tư vấn khách hàng dễ dàng hơn, giúp cho lựa chọn của khách hàng tốt hơn, giải quyết các vấn đề xảy ra trong và sau khi mua hàng.

2. Nhân viên thu ngân

Nhân viên thu ngân sẽ được quyền truy cập đến những thông tin liên quan đến tài chính, như:

- Giá xe, giá phụ kiện

- Thông tin khách hàng

- Thông tin chi tiết về các giao dịch được thực hiên

- Quyền quản lý ngân sách, thu thập, chi phí, phân phối thu mua hàng hóa.

Từ đó đưa ra các thống kê cho cửa hàng, giúp dễ dàng 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, nâng cao năng suất của cửa hàng.

3. Nhân viên sửa chữa, bảo dưỡng

Nhân viên bảo dưỡng có quyền sử dụng các trang thiết bị và vật tư trong cửa hàng, nhưng phải có báo cáo lại. Đồng thời phải xem xét về những mặt hàng được dùng nhiều để nhập thêm số lượng, cũng như giảm bớt những mặt hàng không cần thiết. Ngoài ra, nhân viên bảo dưỡng 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.

4. Nhân viên kho hàng

Quyền của nhân viên kho hàng là sử dụng kho như:

- Truy cập thông tin hàng hóa

- Thống kê trang thiết bị

- Quản lý lịch trình

- Theo dõi hàng (tìm kiếm)

Từ đó thực hiện nhiệm vụ sắp xếp, đảm bảo nhu cầu cho khách hàng, đảm bảo hàng hóa sẽ luôn đầy đủ và ở trong tình trạng tốt nhất.

5. Quản lý

Quyền hạn của người quản lý sẽ nhiều hơn những nhân viên khác và sẽ là người quản lý các nhân viên của cửa hàng. Quản lý 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.

## Thuế nhập xe - nhập phụ tùng

- Thuế nhập xe:

+ Thuế nhập khẩu: áp dụng cho các xe đến từ nước ngoài, có thể dao động từ 56→74% giá trị xe

+ Thuế tiêu thụ đặc biệt: tất cả các mẫu xe dù được sản xuất trong nước hay nhập khẩu đều phải nộp thuế này. Tùy từng chủng loại, dung tích xi lanh và công nghệ sử dụng, mỗi loại xe có mức thuế tiêu thụ đặc biệt khác nhau.

+ Thuế giá trị gia tăng: 10% giá trị xe cho tất cả các dòng xe

⇒ Giá nhập xe:

Nội địa = Giá xe + Thuế tiêu thụ đặc biêt + Thuế giá trị gia tăng

Nhập khẩu = Giá xe + Thuế nhập khẩu + Thuế tiêu thụ đặc biệt + Thuế giá trị gia tăng

⇒ Giá niêm yết = Giá nhập xe \* tỉ lệ lợi nhuận (15→20%)

- Thuế mua xe (Áp dụng cho khách mua xe):

+ Phí trước bạ: 10 → 15%, tùy thành phố.

+ Phí cấp biển số: 20.000.000 VNĐ (TPHCM và Hà Nội); 1.000.000 VNĐ (các thành phố trực thuộc tỉnh, Trung Ương không phải Hà Nội và TPHCM) và 200.000 VNĐ đối với các khu vực còn lại

+ Phí đăng kiểm: 240.000 VNĐ → 560.000 VNĐ(một lần kiểm định)

+ Phí sử dụng đường bộ (1 năm): 1.430.000 VNĐ tùy theo tải trọng xe.

+ Bảo hiểm trách nhiệm dân sự (1 năm): 437.000 VNĐ

+ Phí sử dụng đường bộ + bảo hiểm xe

⇒ Giá lăn bánh = Giá niêm yết + Phí trước bạ + Phí cấp biển số + Phí đăng kiểm

- Phụ tùng

+ Thuế giá trị gia tăng: 10% giá nhập phụ tùng

+ Thuế nhập khẩu thông thường: tùy thuộc vào từng loại phụ tùng mà có giá trị thuế khác nhau, thuế nhập khẩu thông thường có thể dao động từ 4→40% giá nhập phụ tùng

+ Thuế nhập khẩu ưu đãi: có thể dao động từ 3→25% giá nhập phụ tùng

⇒ Giá nhập:

Nội địa = Giá phụ tùng + Thuế giá trị gia tăng

Nhập khẩu = Giá phụ tùng + Thuế giá trị gia tăng + Thuế nhập khẩu thông thường/Thuế nhập khẩu ưu đãi

- Bảng giá dịch vụ bảo dưỡng

+ Kiểm tra lốp xe: 80.000 VNĐ

+ Thay dầu nhớt ô tô định kỳ: 200.000 VNĐ

+ Vệ sinh hệ thống phanh thắng: 200.000 VNĐ

+ Vệ sinh khoang máy Ô tô: 900.000 VNĐ

+ Rửa xe tự động: 50.000 VNĐ

+ Thay lọc xăng: 100.000VNĐ

+ Thay lọc gió xe ô tô: 20.000VNĐ

+ Bảo dưỡng máy đề xe ô tô: 200.000VNĐ

+ Thay bugi xe ô tô: 100.000VNĐ

+ Thông súc bình xăng hoặc thùng dầu xe ô tô: 350.000VNĐ

## Thống kê (Doanh thu, lợi nhuận)

- Tổng chi bao gồm:

Tổng chi phí nhập xe + nhập phụ tùng + lương nhân viên + cơ sở vật chất

- Doanh thu: Tổng số tiền bán xe và tổng số tiền từ các dịch vụ mà doanh nghiệp đã cung cấp cho khách hàng.

Tổng doanh thu = Doanh thu bán xe + Doanh thu bảo dưỡng

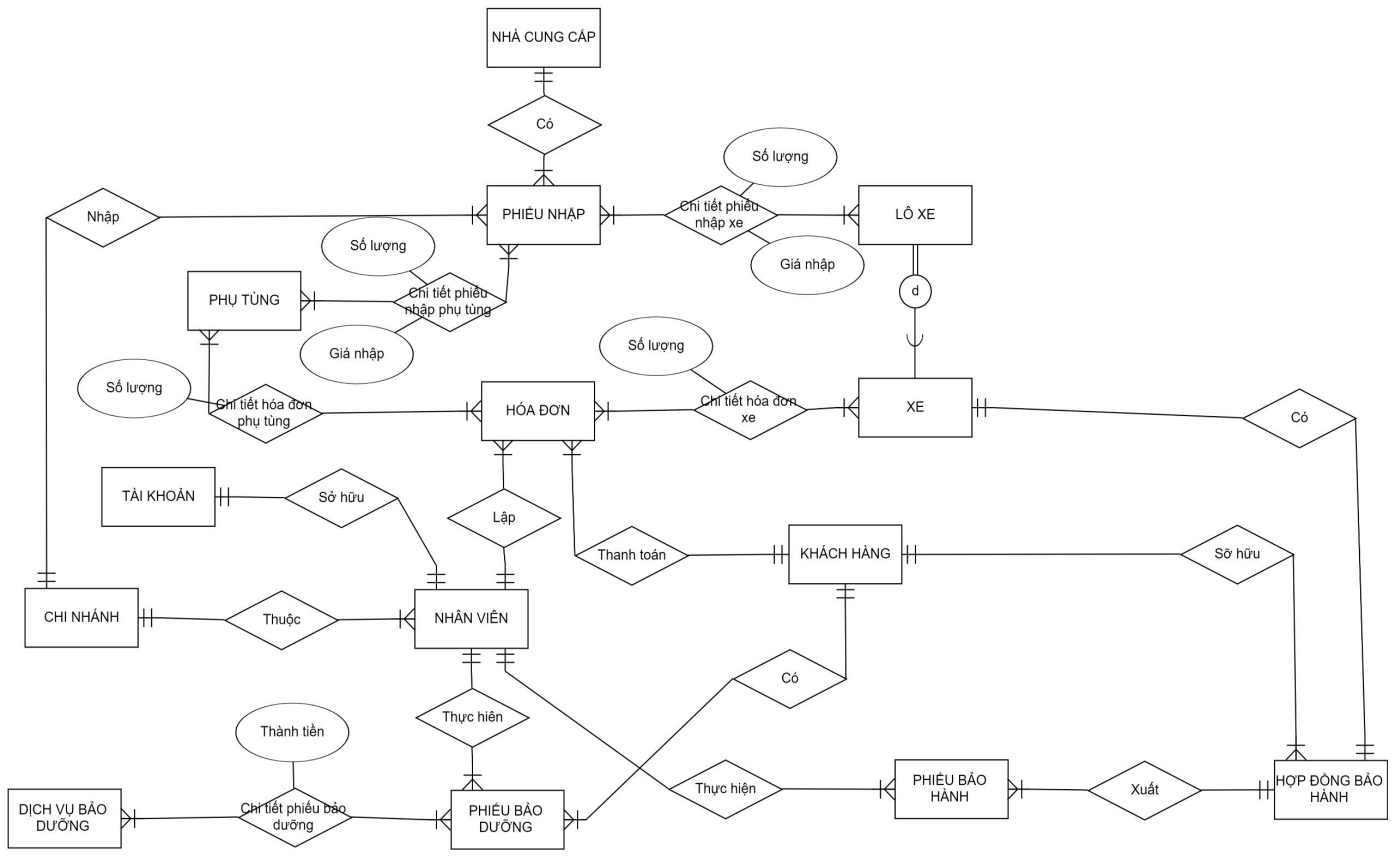
- Lợi nhuận: Số tiền thu được khi trừ tất cả chi phí

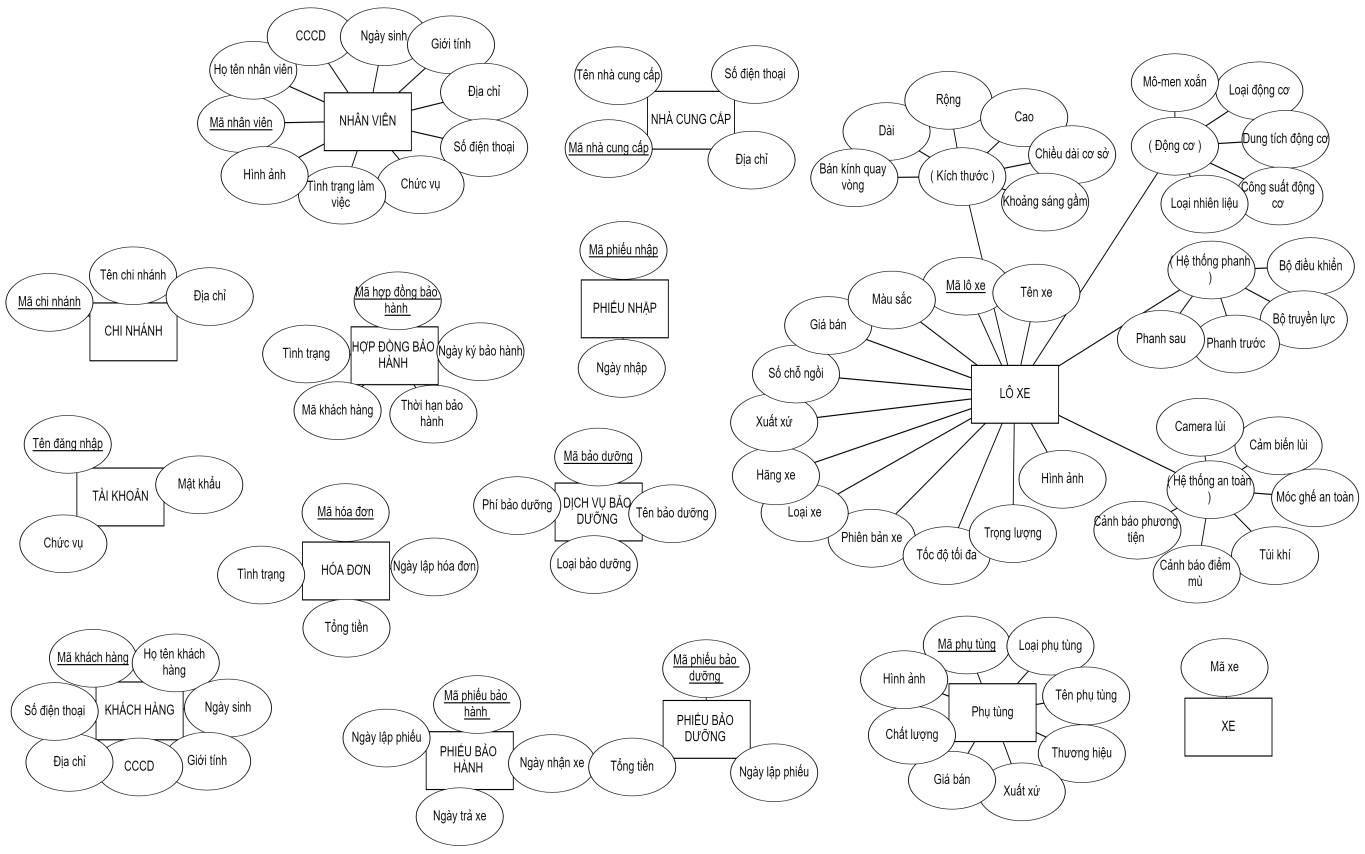
Lợi nhuận = Doanh thu - Tổng chi

# CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Thiết kế cơ sở dữ liệu mức quan niệm

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





## Thiết kế cơ sở dữ liệu mức logic

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

**CHINHANH**(maChiNhanh, tenChiNhanh, diaChi)

**NHANVIEN**(maNhanVien, hoTenNhanVien, CCCD, ngaySinh, gioiTinh, diaChi, soDienThoai, chucVu, tinhTrangLamViec, hinhAnh, maChiNhanh)

**TAIKHOAN**(tenDangNhap, matKhau, chucVu)

**NHACUNGCAP**(maNhaCungCap, tenNhaCungCap, diaChi, soDienThoai)

**LOXE**(maLoXe, tenXe, mauSac, giaBan, soChoNgoi, xuatXu, hangXe, loaiXe, phienBanXe, tocDoToiDa, trongLuong, canhBaoPhuongTien, canhBaoDiemMu, tuiKhi, mocGheAnToan, camBienLui, cameraLui, phanhSau, phanhTruoc, boTruyenLuc, boDieuKhien, mucTieuThuNhienLieu, loaiNhienLieu, congSuatDongCo, dungTichDongCo, loaiDongCo, momenXoan, khoanSangGam, chieuDaiCoSo, dai, rong, cao, banKinhQuayVong, hinhAnh).

**XE**(maXe, maLoXe)

**PHUTUNG**(maPhuTung, loaiPhuTung, tenPhuTung, thuongHieu, xuatXu, giaBan, chatLuong, hinhAnh).

**PHIEUNHAP**(maPhieuNhap, ngayNhap, maNhaCungCap, maChiNhanh)

**CHITIETPHIEUNHAPXE**(maChiTietPhieuNhap, maXe, maPhieuNhap, giaNhap, soLuong)

**CHITIETPHIEUNHAPPHUTUNG**(maChiTietPhieuNhap, maPhuTung, maPhieuNhap, giaNhap, soLuong)

**KHACHHANG**(maKhachHang, hoTenKhachHang, ngaySinh, gioiTinh, CCCD, diaChi, so DienThoai).

**HOADON**(maHoaDon, maKhachHang, maNhanVienThucHien, ngayLapHoaDon, tongTien, tinhTrang)

**CHITIETHOADONXE**(maChiTietHoaDonXe, maHoaDon, maXe, ngayNhanXe, soTienDaTra, phiDangKyBienSo, phiDangKiem, phiTruocBa, baoHiemTrachNhiemDanSu, phiSuDungDuongBo)

**CHITIETHOADONPHUTUNG(**maChiTietHoaDonPhuTung, maHoaDon, maPhuTung, soTienDaTra**)**

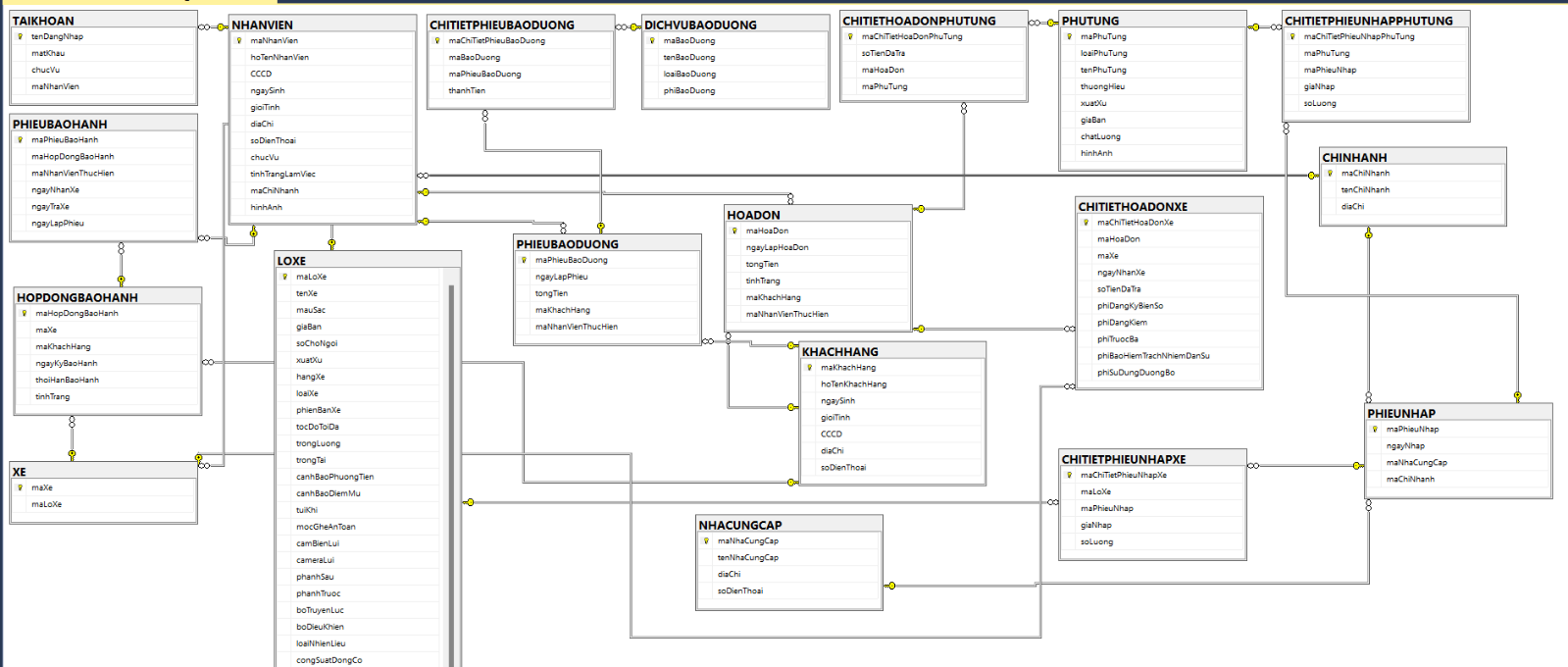
**HOPDONGBAOHANH**(maHopDongBaoHanh, maXe, maKhachHang, ngayKyBaoHanh, thoiHanBaoHanh, tinhTrang)

**PHIEUBAOHANH**(maPhieuBaoHanh, maHopDongBaoHanh, maNhanVienThucHien, ngayNhanXe, ngayTraXe, ngayLapPhieu).

**DICHVUBAODUONG**(maBaoDuong, tenBaoDuong, loaiBaoDuong, phiBaoDuong).

**PHIEUBAODUONG**(maPhieuBaoDuong, maKhachHang, maNhanVienThucHien, ngayLapPhieu, tongTien).

**HOADONBAODUONG**(maHoaDonBaoDuong, maBaoDuong, maPhieuBaoDuong, thanhTien).



1. **Các ràng buộc cần có**

|  |  |  |
| --- | --- | --- |
| **STT** | **TABLE** | **RÀNG BUỘC** |
| **1** | CHINHANH | Ràng buộc khóa chính maChiNhanh |
| **2** | NHANVIEN | Ràng buộc khóa chính maNhanVien, ràng buộc khóa ngoại maChiNhanh tham chiếu dến **CHINHANH** |
| **3** | TAIKHOAN | Ràng buộc khóa chính tenDangNhap, ràng buộc khóa ngoại maNhanVien tham chiếu đến **NHANVIEN** |
| **4** | NHACUNGCAP | Ràng buộc khóa chính maNhaCungCap |
| **5** | LOXE | Ràng buộc khóa chính maLoXe |
| **6** | XE | Ràng buộc khóa chính maXe |
| **7** | PHUTUNG | Ràng buộc khóa chính maPhuTung |
| **8** | PHIEUNHAP | Ràng buộc khóa chính maPhieuNhap, ràng buộc khóa ngoại maNhaCungCap tham chiếu đến **CUNGCAP**, ràng buộc khóa ngoại maChiNhanh tham chiếu đến **CHINHANH** |
| **9** | CHITIETPHIEUNHAPXE | Ràng buộc khóa chính maChiTietPhieuNhapXe, ràng buộc khóa ngoại maXe tham chiếu đến **XE**, ràng buộc khóa ngoại maPhieuNhap tham chiếu đến **PHIEUNHAP** |
| **10** | CHITIETNHIEUNHAPPHUTUNG | Ràng buộc khóa chính maChiTietPhieuNhapPhuTung, ràng buộc khóa ngoại maPhuTung tham chiếu đến **PHUTUNG**, ràng buộc khóa ngoại maPhieuNhap tham chiếu đến **PHIEUNHAP** |
| **11** | KHACHHANG | Ràng buộc khóa chính maKhachHang |
| **12** | HOADON | Ràng buộc khóa chính maHoaDon, ràng buộc khóa ngoại maKhachHang tham chiếu đến **KHACHHANG**, ràng buộc khóa ngoại maNhanVienThucHien tham chiếu đến **NHANVIEN** |
| **13** | CHITIETHOADONXE | Ràng buộc khóa chính maChiTietHoaDonXe, ràng buộc khóa ngoại maHoaDon tham chiếu đến **HOADON**, ràng buộc khóa ngoại maXe tham chiếu đến **XE** |
| **14** | CHITIETHOADONPHUTUNG | Ràng buộc khóa chính maChiTietHoaDonPhuTung, ràng buộc khóa ngoại maHoaDon tham chiếu đến **HOADON**, ràng buộc khóa ngoại maPhuTung tham chiếu đến **PHUTUNG** |
| **15** | HOPDONGBAOHANH | Ràng buộc khóa chính maHopDongBaoHanh, ràng buộc khóa ngoại maXe tham chiếu đến **XE**, ràng buộc khóa ngoại maKhachHang tham chiếu đến **KHACHHANG** |
| **16** | PHIEUBAOHANH | Ràng buộc khóa chính maPhieuBaoHanh, ràng buộc khóa ngoại maHopDongBaoHanh tham chiếu đến **HOPDONGBAOHANH**, ràng buộc khóa ngoại maNhanVienThucHien tham chiếu đến **NHANVIEN** |
| **17** | DICHVUBAODUONG | Ràng buộc khóa chính maBaoDuong |
| **18** | PHIEUBAODUONG | Ràng buộc khóa chính maPhieuBaoDuong, ràng buộc khóa ngoại maKhachHang tham chiếu đến **KHACHHANG**, ràng buộc khóa ngoại maNhanVienThucHien tham chiếu đến **NHANVIEN** |
| **19** | CHITIETPHIEUBAODUONG | Ràng buộc khóa chính maChiTietPhieuBaoDuong, ràng buộc khóa ngoại maBaoDuong tham chiếu đến **DICHVUBAODUONG**, ràng buộc khóa ngoại maPhieuBaoDuong tham chiếu đến **PHIEUBAODUONG** |
| **20** | KHOXE | Ràng buộc khóa chính maChiNhanh và maLoXe, ràng buộc khóa ngoại maChiNhanh tham chiếu đến **CHINHANH**, ràng buộc khóa ngoại maLoXe tham chiếu đến **LOXE** |
| **21** | KHOPHUTUNG | Ràng buộc khóa chính maChiNhanh và maPhuTung, ràng buộc khóa ngoại maChiNhanh tham chiếu đến **CHINHANH**, ràng buộc khóa ngoại maPhuTung tham chiếu đến **PHUTUNG** |

## Cài đặt CSDL và các ràng buộc

**Bảng chi nhánh**

create table CHINHANH(

maChiNhanh nvarchar(20) primary key,

tenChiNhanh nvarchar(50) not null,

diaChi nvarchar(255) not null,

)

**Bảng nhân viên**

|  |
| --- |
| create table NHANVIEN(  maNhanVien nvarchar(20) primary key,  hoTenNhanVien nvarchar(50) not null,  CCCD nvarchar(20) check (len(CCCD) = 12) unique,  ngaySinh date check (DateDiff(year, ngaySinh, GetDate()) >= 18),  gioiTinh nvarchar(5),  diaChi nvarchar(255),  soDienThoai nvarchar(20) check (len(soDienThoai) = 10 or len(soDienThoai) = 11) unique,  chucVu nvarchar(50),  tinhTrangLamViec bit default 1,  maChiNhanh nvarchar(20),  hinhAnh nvarchar(200)  foreign key (maChiNhanh) references CHINHANH(maChiNhanh)  ) |

**Bảng tài khoản**

create table TAIKHOAN(

tenDangNhap nvarchar(20) primary key,

matKhau nvarchar(50) not null,

chucVu nvarchar(50),

maNhanVien nvarchar(20),

foreign key (maNhanVien) references NHANVIEN(maNhanVien)

on delete cascade -- Nhân viên xóa thì tải khoản cũng sẽ xóa

on update cascade -- Nhân viên đổi mã thì tài khoản cũng đổi theo

)

**Bảng nhà cung cấp**

create table NHACUNGCAP(

maNhaCungCap nvarchar(20) primary key,

tenNhaCungCap nvarchar(50) not null,

diaChi nvarchar(255) not null,

soDienThoai nvarchar(20) check (len(soDienThoai) = 10 or len(soDienThoai) = 11),

)

**Bảng Lô xe**

|  |
| --- |
| create table LOXE(  maLoXe nvarchar(20) primary key,  tenXe nvarchar(50),  mauSac nvarchar(20),  giaBan float check (giaBan > 0),  soChoNgoi int check (soChoNgoi >= 2) default 4,  xuatXu nvarchar(50),  hangXe nvarchar(50),  loaiXe nvarchar(50),  phienBanXe nvarchar(50),  tocDoToiDa int check (tocDoToiDa > 0),  trongLuong int check (trongLuong > 0),  trongTai int check (trongTai > 0),  loaiNhienLieu nvarchar(50),  congSuatDongCo int check (congSuatDongCo > 0),  dungTichDongCo int check (dungTichDongCo >= 0),  loaiDongCo nvarchar(50),  khoangSangGam int check (khoangSangGam > 0),  chieuDaiCoSo int check (chieuDaiCoSo > 0),  chieuDai int check (chieuDai > 0),  chieuRong int check (chieuRong > 0),  chieuCao int check (chieuCao > 0),  banKinhQuayVong int check (banKinhQuayVong >= 0),  hinhAnh nvarchar(200)  ) |

**Bảng Xe**

CREATE TABLE XE(

maXe nvarchar(20) primary key,

maLoXe nvarchar(20)

foreign key (maLoXe) references LOXE(maLoXe)

)

**Bảng Phụ tùng**

CREATE TABLE PHUTUNG(

maPhuTung nvarchar(20) primary key,

loaiPhuTung nvarchar(50),

tenPhuTung nvarchar(50),

thuongHieu nvarchar(50),

xuatXu nvarchar(50),

giaBan money check (giaBan > 0),

chatLuong nvarchar(50),

hinhAnh nvarchar(200)

)

**Bảng Phiếu nhập**

create table PHIEUNHAP(

maPhieuNhap nvarchar(20) primary key,

ngayNhap date default GETDATE(),

maNhaCungCap nvarchar(20),

maChiNhanh nvarchar(20),

foreign key (maNhaCungCap) references NHACUNGCAP(maNhaCungCap),

foreign key (maChiNhanh) references CHINHANH(maChiNhanh)

)

**Bảng Chi tiết phiếu nhập xe**

|  |
| --- |
| create table CHITIETPHIEUNHAPXE(  maChiTietPhieuNhapXe nvarchar(20) primary key,  maLoXe nvarchar(20),  maPhieuNhap nvarchar(20),  giaNhap float check (giaNhap > 0),  soLuong int check(soLuong > 0),  foreign key (maLoXe) references LOXE(maLoXe),  foreign key (maPhieuNhap) references PHIEUNHAP(maPhieuNhap)  ) |

**Bảng Chi tiết phiếu nhập phụ tùng**

create table CHITIETPHIEUNHAPPHUTUNG(

maChiTietPhieuNhapPhuTung nvarchar(20) primary key,

maPhuTung nvarchar(20),

maPhieuNhap nvarchar(20),

giaNhap money check (giaNhap > 0),

soLuong integer check(soLuong > 0),

foreign key (maPhuTung) references PHUTUNG(maPhuTung),

foreign key (maPhieuNhap) references PHIEUNHAP(maPhieuNhap)

)

**Bảng Khách hàng**

CREATE TABLE KHACHHANG(

maKhachHang nvarchar(20) primary key,

hoTenKhachHang nvarchar(50) not null,

ngaySinh date check (DateDiff(year, ngaySinh, GetDate()) >= 18),

gioiTinh nvarchar(5),

CCCD nvarchar(20) check (len(CCCD) = 12) unique,

diaChi nvarchar(255),

soDienThoai nvarchar(20) check (len(soDienThoai) = 10 or len(soDienThoai) = 11) unique,

)

**Bảng Hóa đơn**

create table HOADON(

maHoaDon nvarchar(20) primary key,

ngayLapHoaDon nvarchar(50) default GETDATE(),

tongTien money check (tongTien > 0),

tinhTrang nvarchar(50) Check (tinhTrang = N'Chưa thanh toán' or tinhTrang = N'Đã thanh toán') default N'Chưa thanh toán',

maKhachHang nvarchar(20),

maNhanVienThucHien nvarchar(20)

foreign key (maKhachHang) references KHACHHANG(maKhachHang),

foreign key (maNhanVienThucHien) references NHANVIEN(maNhanVien)

)

**Bảng Chi tiết hóa đơn xe**

create table CHITIETHOADONXE(

maChiTietHoaDon nvarchar(20) primary key,

ngayNhanXe nvarchar(50),

soTienDaTra money check (soTienDaTra >= 0),

phiDangKyBienSo money check (phiDangKyBienSo >= 0),

phiDangKiem money check (phiDangKiem >= 0),

phiTruocBa money check (phiTruocBa >= 0),

phiBaoHiemTrachNhiemDanSu money check (phiBaoHiemTrachNhiemDanSu >= 0),

phiSuDungDuongBo money check (phiSuDungDuongBo >= 0),

maHoaDon nvarchar(20),

maXe nvarchar(20),

foreign key (maHoaDon) references HOADON(maHoaDon),

foreign key (maXe) references XE(maXe)

)

**Bảng Chi tiết hóa đơn phụ tùng**

create table CHITIETHOADONPHUTUNG(

maChiTietHoaDonPhuTung nvarchar(20) primary key,

soTienDaTra money check (soTienDaTra >= 0),

maHoaDon nvarchar(20),

maPhuTung nvarchar(20),

foreign key (maHoaDon) references HOADON(maHoaDon),

foreign key (maPhuTung) references PHUTUNG(maPhuTung)

)

**Bảng Hợp đồng bảo hành**

|  |
| --- |
| CREATE TABLE HOPDONGBAOHANH(  maHopDongBaoHanh nvarchar(20) primary key,  maXe nvarchar(20),  maKhachHang nvarchar(20),  ngayKyBaoHanh date,  thoiHanBaoHanh date,  tinhTrang nvarchar(20) check (tinhTrang = N'Còn bảo hành' or tinhTrang = N'Hết hạn'),  foreign key (maXe) references XE(maXe),  foreign key (maKhachHang) references KHACHHANG(maKhachHang)  ) |

**Bảng Phiếu bảo hành**

CREATE TABLE PHIEUBAOHANH(

maPhieuBaoHanh nvarchar(20) primary key,

maHopDongBaoHanh nvarchar(20),

maNhanVienThucHien nvarchar(20),

ngayNhanXe date,

ngayTraXe date,

ngayLapPhieu date,

foreign key (maHopDongBaoHanh) references HOPDONGBAOHANH(maHopDongBaoHanh),

foreign key (maNhanVienThucHien) references NHANVIEN(maNhanVien)

)

**Bảng Dịch vụ bảo dưỡng**

CREATE TABLE DICHVUBAODUONG(

maBaoDuong nvarchar(20) primary key,

tenBaoDuong nvarchar(50) not null,

loaiBaoDuong nvarchar(50) not null,

phiBaoDuong money check (phiBaoDuong >= 0),

)

**Bảng Phiếu bảo dưỡng**

CREATE TABLE PHIEUBAODUONG(

maPhieuBaoDuong nvarchar(20) primary key,

ngayLapPhieu date default GetDate(),

tongTien money check (tongTien >= 0),

maKhachHang nvarchar(20),

maNhanVienThucHien nvarchar(20),

foreign key (maKhachHang) references KHACHHANG(maKhachHang),

foreign key (maNhanVienThucHien) references NHANVIEN(maNhanVien)

)

**Bảng Chi tiết phiếu bảo dưỡng**

|  |
| --- |
| create table CHITIETPHIEUBAODUONG(  maChiTietPhieuBaoDuong nvarchar(20) primary key,  maBaoDuong nvarchar(20),  maPhieuBaoDuong nvarchar(20),  thanhTien float check (thanhTien >= 0),  foreign key (maBaoDuong) references DICHVUBAODUONG(maBaoDuong),  foreign key (maPhieuBaoDuong) references PHIEUBAODUONG(maPhieuBaoDuong)  ) |

**Bảng Kho xe**

|  |
| --- |
| create table KHOXE  (  maChiNhanh nvarchar(20),  maLoXe nvarchar(20),  soLuongXeCon int default 0,  soLuongXeDaBan int default 0,  primary key (maChiNhanh, maLoXe),  foreign key (maChiNhanh) references CHINHANH(maChiNhanh),  foreign key (maLoXe) references LOXE(maLoXe)  ) |

**Bảng Kho phụ tùng**

|  |
| --- |
| create table KHOPHUTUNG  (  maChiNhanh nvarchar(20),  maPhuTung nvarchar(20),  soLuongPhuTungCon int default 0,  soLuongPhuTungDaBan int default 0,  primary key (maChiNhanh, maPhuTung),  foreign key (maChiNhanh) references CHINHANH(maChiNhanh),  foreign key (maPhuTung) references PHUTUNG(maPhuTung)  ) |

## Các ràng buộc khác

Ràng buộc mã bảo dưỡng khi thêm dịch vụ bảo dưỡng

|  |
| --- |
| alter table DICHVUBAODUONG add constraint constr\_MaBaoDuong  default dbo.fn\_SinhMaBaoDuong() for maBaoDuong |

Ràng buộc mã phiếu bảo dưỡng khi thêm phiếu bảo dưỡng

|  |
| --- |
| alter table PHIEUBAODUONG add constraint contr\_MaPhieuBaoDuong  default dbo.fn\_SinhMaPhieuBaoDuong() for maPhieuBaoDuong |

Ràng buộc tạo mã chi tiết phiếu bảo dưỡng khi thêm chi tiết phiếu bảo dưỡng

|  |
| --- |
| alter table CHITIETPHIEUBAODUONG add constraint contr\_MaChiTietPBD  default dbo.fn\_SinhMaChiTietPhieuBaoDuong() for maChiTietPhieuBaoDuong |

Ràng buộc tạo mã hợp đồng bảo hành khi thêm hợp đồng bảo hành

|  |
| --- |
| alter table HOPDONGBAOHANH add constraint contr\_MaHopDongBaoHanh  default dbo.fn\_SinhMaHopDongBaoHanh() for maHopDongBaoHanh |

Ràng buộc tạo mã phiếu bảo hang khi thêm phiếu bảo hành

|  |
| --- |
| alter table PHIEUBAOHANH add constraint contr\_MaPhieuBaoHanh  default dbo.fn\_SinhMaPhieuBaoHanh() for maPhieuBaoHanh |

Ràng buộc tạo mã khách hàng khi thêm khách hàng

|  |
| --- |
| alter table KHACHHANG add constraint contr\_MaKhachHang  default dbo.fn\_SinhMaKhachHang() for maKhachHang |

Ràng buộc tạo mã phiếu nhập khi thêm phiếu nhập

|  |
| --- |
| alter table PHIEUNHAP add constraint contr\_SinhMaPhieuNhap  default dbo.fn\_SinhMaPhieuNhap() for maPhieuNhap |

Ràng buộc tạo mã chi tiết phiếu nhập xe khi thêm chi tiết phiếu nhập xe

|  |
| --- |
| alter table CHITIETPHIEUNHAPXE add constraint contr\_SinhMaChiTietPhieuNhapXe  default dbo.fn\_SinhMaChiTietPhieuNhapXe() for maChiTietPhieuNhapXe |

Ràng buộc tạo mã chi tiết phiếu nhập phụ tùng khi thêm chi tiết phiếu nhập phụ tùng

|  |
| --- |
| alter table CHITIETPHIEUNHAPPHUTUNG add constraint contr\_SinhMaChiTietPhieuNhapPhuTung  default dbo.fn\_SinhMaChiTietPhieuNhapPhuTung() for maChiTietPhieuNhapPhuTung |

Ràng buộc tạo mã hóa đơn khi thêm hóa đơn

|  |
| --- |
| alter table HOADON add constraint contr\_SinhMaHoaDon  default dbo.fn\_SinhMaHoaDon() for maHoaDon |

Ràng buộc tạo mã chi tiết hóa đơn phụ tùng khi thêm chi tiết hóa đơn phụ tùng

|  |
| --- |
| alter table CHITIETHOADONPHUTUNG add constraint contr\_SinhMaCTHDPT  default dbo.fn\_SinhMaChiTietHoaDonPhuTung() for maChiTietHoaDonPhuTung |

Ràng buộc tạo mã chi tiết hóa đơn xe khi thêm chi tiết hóa đơn xe

|  |
| --- |
| alter table CHITIETHOADONXE add constraint contr\_SinhMaCTHDXe  default dbo.fn\_SinhMaCTHDXe() for maChiTietHoaDonXe |

Ràng buộc tạo mã phụ tùng khi thêm phụ tùng

|  |
| --- |
| alter table PHUTUNG add constraint contr\_SinhMaPhuTung  default dbo.fn\_SinhMaPhuTung() for maPhuTung |

## Các View

1. **Xem danh sách nhân viên còn làm việc**

|  |
| --- |
| CREATE VIEW v\_DSNhanVienConLamViec as  SELECT hotenNhanVien, CCCD, ngaySinh, gioiTinh, soDienThoai  FROM NHANVIEN as nv  WHERE nv.tinhTrangLamViec = 1 |

1. **Xem danh sách chi nhánh**

|  |
| --- |
| CREATE VIEW v\_DSChiNhanh as  SELECT \*  FROM CHINHANH |

1. **Xem danh sách nhà cung cấp**

|  |
| --- |
| CREATE VIEW v\_DSNhaCungCap as  SELECT \*  FROM NHACUNGCAP |

1. **Xem danh sách xe**

|  |
| --- |
| CREATE VIEW v\_DSXeConTrongKho as  SELECT maXe, tenXe, mauSac, soChoNgoi, xuatXu, hangXe, loaiXe, phienBanXe, pn.soLuong  FROM XE, PHIEUNHAP as pn  WHERE pn.soLuong > 0 |

1. **Xem danh sách nhân viên có chức vụ là quản lý**

|  |
| --- |
| CREATE or alter VIEW v\_NhanVienQuanLy as  select nv.maNhanVien, nv.hoTenNhanVien, nv.soDienThoai, cn.maChiNhanh, cn.tenChiNhanh  from NHANVIEN as nv, CHINHANH as cn  where nv.maChiNhanh = cn.maChiNhanh and nv.chucVu = 'Quản lý' |

1. **Xem danh sách xe có xuất xứ từ Nhật Bản**

|  |
| --- |
| create or alter view v\_XeXuatXuNhatBan as  select tenXe, giaBan, soChoNgoi,loaiXe, loaiDongCo, loaiNhienLieu  from LOXE where xuatXu = 'Nhật Bản' |

1. **Xem số xe đã bán theo chi nhánh**

|  |
| --- |
| CREATE or ALTER VIEW v\_SoXeDaBan as  SELECT cn.maChiNhanh, cn.maXe, CASE WHEN hd.daBan IS NULL  THEN 0  ELSE hd.daBan END  AS daBan  FROM (SELECT distinct cn.maChiNhanh, x.maXe  FROM CHINHANH as cn, PHIEUNHAP as pn, CHITIETPHIEUNHAPXE as pnx, XE as x  WHERE cn.maChiNhanh = pn.maChiNhanh  and pn.maPhieuNhap = pnx.maPhieuNhap  and pnx.maXe = x.maXe) as cn  LEFT OUTER JOIN  (SELECT cn.maChiNhanh, hdx.maXe, count(\*) daBan  FROM NHANVIEN as nv,CHINHANH as cn, HOADON as hd, CHITIETHOADONXE as hdx  WHERE nv.maChiNhanh = cn.maChiNhanh and hd.maNhanVienThucHien = nv.maNhanVien and hd.maHoaDon = hdx.maHoaDon  GROUP BY cn.maChiNhanh, hdx.maXe) as hd  on hd.maChiNhanh = cn.maChiNhanh and hd.maXe = cn.maXe |

1. **Xem danh sách xe theo từng chi nhánh**

|  |
| --- |
| CREATE or AlTER VIEW v\_KhoXeTheoChiNhanh as  SELECT distinct cn.maChiNhanh, cn.maXe, (cn.soLuong - hd.daBan) as Conlai  FROM (SELECT pn.maChiNhanh, ctpn.maXe, sum(ctpn.soLuong) as soLuong  FROM PHIEUNHAP as pn, CHITIETPHIEUNHAPXE as ctpn  WHERE pn.maPhieuNhap = ctpn.maPhieuNhap  GROUP BY pn.maChiNhanh, ctpn.maXe) as cn, v\_SoXeDaBan as hd  WHERE cn.maChiNhanh = hd.maChiNhanh and cn.maXe = hd.maXe |

1. **Các Trigger**
2. **Set tên đăng nhập mặc định là mã nhân viên**

|  |
| --- |
| create or alter trigger trg\_ThemTaiKhoan  on NHANVIEN  for insert  as  begin  declare @taiKhoan nvarchar(20), @chucVu nvarchar(50)  select @taiKhoan = maNhanVien, @chucVu = chucVu from inserted  insert into TAIKHOAN values (@taiKhoan, '1', @chucVu, @taiKhoan)  end |

1. **Thay đổi trạng thái hóa đơn khi thanh toán đủ**

|  |
| --- |
| CREATE or ALTER TRIGGER tg\_ThayDoiTrangThaiHoaDon on CHITIETHOADONXE  for update, insert as  BEGIN  DECLARE @soTienDaTra money, @maHoaDon nvarchar(20), @tongTien money  SELECT @soTienDaTra = ins.soTienDaTra, @maHoaDon = ins.maHoaDon FROM inserted as ins  SElECT @tongTien = hd.tongTien FROM HOADON as hd WHERE hd.maHoaDon = @maHoaDon  IF @tongTien <= @soTienDaTra  BEGIN  UPDATE HOADON  SET tinhTrang = N'Đã Thanh Toán'  WHERE maHoaDon = @maHoaDon  END  END |

1. **Tự động tạo mã xe theo lô khi nhập lô xe**

|  |
| --- |
| create or alter trigger trg\_SinhMaXeKhiNhapXe  on CHITIETPHIEUNHAPXE  after insert  as  begin  declare @maLoXe nvarchar(20), @soLuong nvarchar(20)  -- Lấy xe mã Lô xe và số lượng từ inserted (CTPN Xe)  select @maLoXe = maLoXe, @soLuong = soLuong from inserted  -- Nếu đã nhập mã lô xe này  if (@maLoXe in (select maLoXe from XE))  begin  declare @i int = @soLuong  while @i > 0 -- Bắt đầu thực hiện cho tới khi hết số lượng  begin  -- Khởi tạo giá trị LastID = Mã xe cuối cùng trong LoXe vừa insert  declare @NewId nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maXe, 11, LEN(maXe) - 10) as int)), 0)  from XE where maLoXe = @maLoXe  if (@LastID < 999)  begin  set @NewId = @maLoXe + '\_XE' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  begin  set @NewId = @maLoXe + '\_XE' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  insert into XE values (@NewId, @maLoXe)  set @i = @i - 1  end  end  else -- Xe có mã này chưa được nhập lần nào  begin  declare @y int = @soLuong, @ID int = 1  while @y > 0 -- Bắt đầu thực hiện cho tới khi hết số lượng  begin  if (@ID < 999)  begin  set @NewId = @maLoXe + '\_XE' + RIGHT('000' + CAST(@ID as nvarchar), 3)  end  else  begin  set @NewId = @maLoXe + '\_XE' + CAST(@ID as nvarchar)  end  insert into XE values (@NewId, @maLoXe)  set @y = @y - 1  set @ID = @ID + 1  end  end  end |

1. **Không cho phép 2 người quản lý cùng 1 chi nhánh**

|  |
| --- |
| CREATE or ALTER TRIGGER tg\_MotQuanLy on NHANVIEN  AFTER UPDATE, INSERT as  BEGIN  DECLARE @maChiNhanh nvarchar(20), @chucVu nvarchar(50), @count int, @maNhanVien nvarchar(20)  SELECT @maChiNhanh = ins.maChiNhanh, @chucVu = ins.chucVu, @maNhanVien = ins.maNhanVien, @count = nv.soluong FROM inserted as ins, (SELECT nv.maChiNhanh, count(nv.maNhanVien) as soluong  FROM NHANVIEN as nv  WHERE nv.chucVu = N'Quản lý'  GROUP BY nv.maChiNhanh) as nv  WHERE ins.maChiNhanh = nv.maChiNhanh  IF @chucVu = N'Quản lý' and @count > 1  BEGIN  PRINT N'Không thể thêm quản lý'  UPDATE NHANVIEN  SET chucVu = null  WHERE maNhanVien = @maNhanVien  END  END |

1. **Trigger cập nhật kho xe khi nhập xe về chi nhánh**

|  |
| --- |
| --Trigger khi nhập hàng về chi nhánh thì sẽ cập nhật vào kho  ---------- Cập nhật kho xe  create or alter trigger trg\_CapNhatKhoXeKhiNhapHang  on CHITIETPHIEUNHAPXE  after insert  as  begin  declare @maChiNhanh nvarchar(20), @maLoXe nvarchar(20), @soLuong int  select @maChiNhanh = maChiNhanh, @maLoXe = maLoXe, @soLuong = soLuong  from PHIEUNHAP PN  inner join inserted  on PN.maPhieuNhap = inserted.maPhieuNhap  if ( (@maChiNhanh in (select maChiNhanh from KHOXE)) and (@maLoXe in (select maLoXe from KHOXE)) )  begin  Update KHOXE  set soLuongXeCon = soLuongXeCon + @soLuong  where maChiNhanh = @maChiNhanh and maLoXe = @maLoXe  end  else  begin  insert into KHOXE(maChiNhanh, maLoXe, soLuongXeCon)  values(@maChiNhanh, @maLoXe, @soLuong)  end  end |

1. **Trigger cập nhật kho phụ tùng khi nhập phụ tùng về chi nhánh**

|  |
| --- |
| ---------- Cập nhật kho phụ tùng  create or alter trigger trg\_CapNhatKhoPhuTungKhiNhapHang  on CHITIETPHIEUNHAPPHUTUNG  after insert  as  begin  declare @maChiNhanh nvarchar(20), @maPhuTung nvarchar(20), @soLuong int  select @maChiNhanh = maChiNhanh, @maPhuTung = maPhuTung, @soLuong = soLuong  from PHIEUNHAP PN  inner join inserted on PN.maPhieuNhap = inserted.maPhieuNhap  if ( (@maChiNhanh in (select maChiNhanh from KHOPHUTUNG)) and (@maPhuTung in (select maPhuTung from KHOPHUTUNG)) )  begin  update KHOPHUTUNG  set soLuongPhuTungCon = soLuongPhuTungCon + @soLuong  where maChiNhanh = @maChiNhanh and maPhuTung = @maPhuTung  end  else  begin  insert into KHOPHUTUNG(maChiNhanh, maPhuTung, soLuongPhuTungCon)  values(@maChiNhanh, @maPhuTung, @soLuong)  end  end |

**g. Trigger tự động sinh mã nhân viên**

|  |
| --- |
| -- TRIGGER tự động sinh mã nhân viên  go  CREATE or Alter trigger tg\_ThemNhanVien on NHANVIEN  instead of insert  as  BEGIN  declare @hoTenNhanVien nvarchar(50), @CCCD nvarchar(20), @ngaySinh date, @gioiTinh nvarchar(5), @diaChi nvarchar(255), @soDienThoai nvarchar(20), @chucVu nvarchar(50), @maChiNhanh nvarchar(20), @hinhAnh nvarchar(200)  select @hoTenNhanVien = nv.hoTenNhanVien, @CCCD = nv.CCCD, @ngaySinh = nv.ngaySinh, @gioiTinh = nv.gioiTinh, @diaChi = nv.diaChi, @soDienThoai = nv.soDienThoai, @chucVu = nv.chucVu, @maChiNhanh = nv.maChiNhanh, @hinhAnh = nv.hinhAnh  from inserted as nv  print (dbo.fn\_TaoMaNhanVien(@maChiNhanh))  insert into NHANVIEN(maNhanVien, hoTenNhanVien, CCCD, ngaySinh, gioiTinh, diaChi, soDienThoai, chucVu, maChiNhanh, hinhAnh) Values(dbo.fn\_TaoMaNhanVien(@maChiNhanh), @hoTenNhanVien , @CCCD , @ngaySinh, @gioiTinh, @diaChi , @soDienThoai, @chucVu, @maChiNhanh, @hinhAnh)  END |

## Các Function

1. **Function tự động sinh mã bảo dưỡng**

|  |
| --- |
| create or alter function fn\_SinhMaBaoDuong()  returns nvarchar(20)  as  begin  declare @nextID nvarchar(20), @lastID nvarchar(20)  select @lastID = ISNULL(MAX(CAST(SUBSTRING(maBaoDuong, 5, LEN(maBaoDuong) - 4) as int)) , 0)  from DICHVUBAODUONG  if (@lastID < 999)  set @nextID = 'DVBD' + RIGHT('000' + CAST(ISNULL(@lastID, 0) + 1 as nvarchar), 3)  else  set @nextID = 'DVBD' + CAST(ISNULL(@lastID, 0) + 1 as nvarchar)  return @nextID  end |

1. **Function tự động sinh mã phiếu bảo dưỡng**

|  |
| --- |
| create or alter function fn\_SinhMaPhieuBaoDuong()  returns nvarchar(20)  as  begin  declare @NextID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maPhieuBaoDuong, 4, LEN(maPhieuBaoDuong) - 3) as int)), 0)  from PHIEUBAODUONG  if (@LastID < 999)  set @NextID = 'PBD' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  else  set @NextID = 'PBD' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NextID  end |

1. **Function tự động sinh mã chi tiết phiếu bảo dưỡng**

|  |
| --- |
| create or alter function fn\_SinhMaChiTietPhieuBaoDuong()  returns nvarchar(20)  as  begin  declare @NextID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maChiTietPhieuBaoDuong, 6, LEN(maChiTietPhieuBaoDuong) - 5) as int)), 0)  from CHITIETPHIEUBAODUONG  if (@LastID < 999)  set @NextID = 'CTPBD' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  else  set @NextID = 'CTPBD' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NextID  end |

1. **Function tự động sinh mã hợp đồng bảo hành**

|  |
| --- |
| create or alter function fn\_SinhMaHopDongBaoHanh()  returns nvarchar(20)  as  begin  declare @NextID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maHopDongBaoHanh, 5, LEN(maHopDongBaoHanh) - 4) as int)), 0)  from HOPDONGBAOHANH  if (@LastID < 999)  set @NextID = 'HDBH' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  else  set @NextID = 'HDBH' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NextID  end |

1. **Function tự động sinh mã phiếu bảo hành**

|  |
| --- |
| create or alter function fn\_SinhMaPhieuBaoHanh()  returns nvarchar(20)  as  begin  declare @NextID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maPhieuBaoHanh, 4, LEN(maPhieuBaoHanh) - 3) as int)), 0)  from PHIEUBAOHANH  if (@LastID < 999)  set @NextID = 'PBH' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  else  set @NextID = 'PBH' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NextID  end |

1. **Function tự động sinh mã khách hàng**

|  |
| --- |
| create or alter function fn\_SinhMaKhachHang()  returns nvarchar(20)  as  begin  declare @NextID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maKhachHang, 3, LEN(maKhachHang) - 2) as int)), 0)  from KHACHHANG  if (@LastID < 999)  set @NextID = 'KH' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  else  set @NextID = 'KH' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NextID  end |

1. **Function tự động sinh mã phiếu nhập**

|  |
| --- |
| create or alter function fn\_SinhMaPhieuNhap()  returns nvarchar(20)  as  begin  declare @NewId nvarchar(20), @LastId nvarchar(20)  select @LastId = ISNULL(MAX(CAST(SUBSTRING(maPhieuNhap, 3, LEN(maPhieuNhap) - 2) as int)), 0)  from PHIEUNHAP  if (@LastId < 999)  begin  set @NewId = 'PN' + RIGHT('000' + CAST(ISNULL(@LastId, 0) + 1 as nvarchar), 3)  end  else  begin  set @NewId = 'PN' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  return @NewID  end |

1. **Function tự động sinh mã chi tiết phiếu nhập xe**

|  |
| --- |
| create or alter function fn\_SinhMaChiTietPhieuNhapXe()  returns nvarchar(20)  as  begin  declare @NextId nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maChiTietPhieuNhapXe, 6, LEN(maChiTietPhieuNhapXe) - 5) as int)), 0)  from CHITIETPHIEUNHAPXE    if (@LastID < 999)  begin  set @NextId = 'CTPNX' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  begin  set @NextId = 'CTPNX' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  return @NextId  end |

1. **Function tự động sinh mã chi tiết phiếu nhập phụ tùng**

|  |
| --- |
| create or alter function fn\_SinhMaChiTietPhieuNhapPhuTung()  returns nvarchar(20)  as  begin  declare @NewID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maChiTietPhieuNhapPhuTung, 7, len(maChiTietPhieuNhapPhuTung) - 6) as int)), 0)  from CHITIETPHIEUNHAPPHUTUNG  if (@LastID < 999)  begin  set @NewID = 'CTPNPT' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  set @NewID = 'CTPNPT' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NewID  end |

1. **Function tự động sinh mã hóa đơn**

|  |
| --- |
| create or alter function fn\_SinhMaHoaDon()  returns nvarchar(20)  as  begin  declare @NewID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maHoaDon, 3, len(maHoaDon) - 2) as int)), 0)  from HOADON  if (@LastID < 999)  begin  set @NewID = 'HD' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  set @NewID = 'HD' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  return @NewID  end |

1. **Function tự động sinh mã chi tiết hóa đơn xe**

|  |
| --- |
| create or alter function fn\_SinhMaCTHDXe()  returns nvarchar(20)  as  begin  declare @NewID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maChiTietHoaDonXe, 6, LEN(maChiTietHoaDonXe) - 1) as int)), 0)  from CHITIETHOADONXE  if (@LastID < 999)  begin  set @NewID = 'CTHDX' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  begin  set @NewID = 'CTHDX' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  return @NewID  end |

1. **Function tự động sinh mã chi tiết hóa đơn phụ tùng**

|  |
| --- |
| create or alter function fn\_SinhMaChiTietHoaDonPhuTung()  returns nvarchar(20)  as  begin  declare @NewID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maChiTietHoaDonPhuTung, 7, LEN(maChiTietHoaDonPhuTung) - 6) as int)), 0)  from CHITIETHOADONPHUTUNG  if (@LastID < 999)  begin  set @NewID = 'CTHDPT' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  begin  set @NewID = 'CTHDPT' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  return @NewID  end |

1. **Function tự động sinh mã phụ tùng**

|  |
| --- |
| create or alter function fn\_SinhMaPhuTung()  returns nvarchar(20)  as  begin  declare @NewID nvarchar(20), @LastID nvarchar(20)  select @LastID = ISNULL(MAX(CAST(SUBSTRING(maPhuTung, 3, LEN(maPhuTung) - 2) as int)), 0)  from PHUTUNG  if (@LastID < 999)  begin  set @NewID = 'PT' + RIGHT('000' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar), 3)  end  else  begin  set @NewID = 'PT' + CAST(ISNULL(@LastID, 0) + 1 as nvarchar)  end  return @NewID  end |

1. **Function tự động sinh mã nhân viên**

|  |
| --- |
| create or alter function fn\_TaoMaNhanVien  (@MaChiNhanh nvarchar(20))  returns nvarchar(20)  as  begin  declare @nextID nvarchar(20), @Count int, @TienTo varchar(20), @TenChiNhanh nvarchar(50), @I int  declare @DodaiMa int  SET @DodaiMa = 3  SELECT @TenChiNhanh = cn.tenChiNhanh FROM CHINHANH cn WHERE cn.maChiNhanh = @MaChiNhanh  SELECT @Count = count(\*) FROM NHANVIEN vn WHERE vn.maChiNhanh = @MaChiNhanh  SET @I = charindex(' ', @TenChiNhanh)  SET @TienTo = SUBSTRING(@TenChiNhanh, 1, 1)  WHILE @I > 0  BEGIN  SET @TienTo = CONCAT(@TienTo, SUBSTRING(@TenChiNhanh, @I + 1, 1))  SET @TenChiNhanh = SUBSTRING(@TenChiNhanh, @I + 1, len(@TenChiNhanh) -1)  SET @I = charindex(' ', @TenChiNhanh)  END  SET @nextID = CONCAT(CONCAT(CONCAT('NV', SUBSTRING(@TienTo,3, @DodaiMa)), REPLICATE('0', @DodaiMa - len(@Count))), @Count + 1)  return @nextID  end  go |

1. **Function tính tổng số tiền khách hàng phải trả khi thanh toán một chiếc xe**

|  |
| --- |
| create or alter function fn\_TongSoTienCanThanhToan  (@maKhachHang nvarchar(20), @maXe nvarchar(20))  returns money  as  begin  declare @TongTien money  declare @giaBan money, @phiDangKyBienSo money, @phiDangKiem money,  @phiTruocBa money, @phiBaoHiemTrachNhiemDanSu money, @phiSuDungDuongBo money  select @giaBan = giaBan, @phiDangKyBienSo = phiDangKyBienSo, @phiDangKiem = phiDangKiem,  @phiTruocBa = phiTruocBa, @phiBaoHiemTrachNhiemDanSu = phiBaoHiemTrachNhiemDanSu,  @phiSuDungDuongBo = phiSuDungDuongBo  from  (select KH.maKhachHang, maXe  from KHACHHANG KH  inner join HOADON HD on KH.maKhachHang = HD.maKhachHang  inner join CHITIETHOADONXE CTHDX on HD.maHoaDon = CTHDX.maHoaDon) Q  -- lấy ra mã khách hàng và mã xe  inner join  (select maXe, giaBan  from XE inner join LOXE  on XE.maLoXe = LOXE.maLoXe) P -- lấy ra mã xe và giá bán  on Q.maXe = P.maXe  inner join  (select maXe, phiDangKyBienSo, phiDangKiem, phiTruocBa,  phiBaoHiemTrachNhiemDanSu, phiSuDungDuongBo  from CHITIETHOADONXE) K -- lấy ra mã xe và các loại phí của xe  on P.maXe = K.maXe  where maKhachHang = @maKhachHang and Q.maXe = @maXe  set @TongTien = @giaBan + @phiDangKyBienSo + @phiDangKiem + @phiTruocBa  + @phiBaoHiemTrachNhiemDanSu + @phiSuDungDuongBo  return @TongTien  end |

**p. Function lấy danh sách nhân viên theo chi nhánh**

|  |
| --- |
| create OR alter function fn\_LayDanhSachNhanVienTheoChiNhanh  (@MaChiNhanh nvarchar(20))  returns table  as  return (  select \* from NHANVIEN nv  where nv.maChiNhanh = @MaChiNhanh  ) |

**q. Function lấy danh sách hóa đơn chưa thanh toán**

|  |
| --- |
| create or alter function fn\_LayDanhSachHoaDonChuaThanhToan()  returns table  as  return(  select \* from HOADON hd  where hd.tinhTrang != N'Đã thanh toán'  ) |

## Stored Procedure

### Chi Nhánh

**a. Procedure Xem thông tin chi nhánh**

create or alter proc List\_CHINHANH

as

select \* from CHINHANH

\* Code C#

public List<ChiNhanh> DanhSachChiNhanh()

{

List<ChiNhanh> listChiNhanh = new List<ChiNhanh>();

string query = "exec list\_ChiNhanh";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach(DataRow row in data.Rows)

{

ChiNhanh chiNhanh = new ChiNhanh(row);

listChiNhanh.Add(chiNhanh);

}

return listChiNhanh;

}

**b. Procedure Thêm chi nhánh**

create or alter proc Insert\_CHINHANH @maChiNhanh nvarchar(20), @tenChiNhanh nvarchar(50), @diaChi nvarchar(255)

as

insert into CHINHANH (maChiNhanh, tenChiNhanh, diaChi) values (@maChiNhanh, @tenChiNhanh, @diaChi)

\* Code C#

public bool ThemChiNhanh(string maChiNhanh, string tenChiNhanh, string diaChi)

{

string query = string.Format("exec Insert\_CHINHANH @maChiNhanh , @tenChiNhanh , @diaChi ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maChiNhanh, tenChiNhanh, diaChi });

return result > 0;

}

**c. Procedure Cập nhật chi nhánh**

create or alter proc Update\_CHINHANH @tenChiNhanh nvarchar(50), @diaChi nvarchar(255), @maChiNhanh nvarchar(20)

as

update CHINHANH set tenChiNhanh = @tenChiNhanh, diaChi = @diaChi where maChiNhanh = @maChiNhanh

\* Code C#

public bool CapNhatChiNhanh(string maChiNhanh, string tenChiNhanh, string diaChi)

{

string query = string.Format("exec Update\_CHINHANH @tenChiNhanh , @diaChi , @maChiNhanh ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { tenChiNhanh, diaChi, maChiNhanh });

return result > 0;

}

**d. Procedure Xóa chi nhánh**

create or alter proc Delete\_CHINHANH @maChiNhanh nvarchar(20)

as

delete CHINHANH where maChiNhanh = @maChiNhanh

\* Code C#

public bool XoaChiNhanh(string maChiNhanh)

{

string query = string.Format("exec Delete\_CHINHANH @maChiNhanh");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiNhanh});

return result > 0;

}

### Hóa Đơn

**a. Procedure Xem thông tin Hóa Đơn**

create or alter proc List\_HOADON

as

select \* from HOADON

\* Code C#

public List<HoaDon> DanhSachHoaDon()

{

List<HoaDon> listHoaDon = new List<HoaDon>();

string query = "exec list\_HOADON";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

HoaDon hoaDon = new HoaDon(row);

listHoaDon.Add(hoaDon);

}

return listHoaDon;

}

**b. Procedure Thêm Hóa Đơn**

create or alter proc Insert\_HOADON @ngayLapHoaDon date, @tongTien float, @tinhTrang nvarchar(50), @maKhachHang nvarchar(20), @maNhanVienThucHien nvarchar(20)

as

insert into HOADON (ngayLapHoaDon, tongTien, tinhTrang, maKhachHang, maNhanVienThucHien) values (@ngayLapHoaDon, @tongTien, @tinhTrang, @maKhachHang, @maNhanVienThucHien)

\*Code C#

public bool ThemHoaDon(string ngayLapHoaDon, float tongTien, string tinhTrang, string maKhachHang, string maNVTH)

{

string query = string.Format("exec Insert\_HOADON @ngayLapHoaDon , @tongTien , @tinhTrang , @maKhachHang , @maNhanVienThucHien ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {ngayLapHoaDon.Substring(0, 10), tongTien, tinhTrang, maKhachHang, maNVTH });

return result > 0;

}

**c. Procedure Cập nhật Hóa Đơn**

create or alter proc Update\_HOADON @maHoaDon nvarchar(20), @ngayLapHoaDon date, @tongTien float, @tinhTrang nvarchar(50), @maKhachHang nvarchar(20), @maNhanVienThucHien nvarchar(20)

as

update HOADON set ngayLapHoaDon = @ngayLapHoaDon, tongTien = @tongTien, tinhTrang = @tinhTrang, maKhachHang = @maKhachHang, maNhanVienThucHien = @maNhanVienThucHien where maHoaDon = @maHoaDon

\* Code C#

public bool CapNhatHoaDon(string maHoaDon, string ngayLapHoaDon, float tongTien, string tinhTrang, string maKhachHang, string maNVTH)

{

string query = string.Format("exec Update\_HOADON @maHoaDon , @ngayLapHoaDon , @tongTien , @tinhTrang , @maKhachHang , @maNhanVienThucHien ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maHoaDon, ngayLapHoaDon.Substring(0, 10), tongTien, tinhTrang, maKhachHang, maNVTH });

return result > 0;

}

**d. Procedure Xóa Hóa Đơn**

create or alter proc Delete\_HOADON @maHoaDon nvarchar(20)

as

delete HOADON where maHoaDon = @maHoaDon

\*Code C#

public bool XoaHoaDon(string maHoaDon)

{

string query = string.Format("exec Delete\_HOADON @maHoaDon ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maHoaDon});

return result > 0;

}

### Chi Tiết Hóa Đơn Phụ Tùng

**a. Procedure Xem thông tin Chi Tiết Hóa Đơn Phụ Tùng**

create or alter proc List\_CHITIETHOADONPHUTUNG

as

select \* from CHITIETHOADONPHUTUNG

\*Code C#

public List<ChiTietHoaDonPhuTung> DanhSachHoaDonPhuTung()

{

List<ChiTietHoaDonPhuTung> listHoaDonPhuTung = new List<ChiTietHoaDonPhuTung>();

string query = "exec List\_CHITIETHOADONPHUTUNG";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach(DataRow row in data.Rows)

{

ChiTietHoaDonPhuTung hoaDonPhuTung = new ChiTietHoaDonPhuTung(row);

listHoaDonPhuTung.Add(hoaDonPhuTung);

}

return listHoaDonPhuTung;

}

**b. Procedure Thêm Chi Tiết Hóa Đơn Phụ Tùng**

create or alter proc Insert\_CHITIETHOADONPHUTUNG @maHoaDon nvarchar(20), @maPhuTung nvarchar(20), @soTienDaTra float

as

insert into CHITIETHOADONPHUTUNG (maHoaDon, maPhuTung, soTienDaTra) values (@maHoaDon, @maPhuTung, @soTienDaTra)

\* Code C#

public bool ThemChiTietHoaDonPhuTung(string maHoaDon, string maPhuTung, float soTienDaTra)

{

string query = string.Format("exec Insert\_CHITIETHOADONPHUTUNG @maHoaDon , @maPhuTung , @soTienDaTra ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maHoaDon, maPhuTung, soTienDaTra });

return result > 0;

}

**c. Procedure Cập nhật Chi Tiết Hóa Đơn Phụ Tùng**

create or alter proc Update\_CHITIETHOADONPHUTUNG @maChiTietHoaDonPhuTung nvarchar(20), @maHoaDon nvarchar(20), @maPhuTung nvarchar(20), @soTienDaTra float

as

update CHITIETHOADONPHUTUNG set soTienDaTra = @soTienDaTra, maHoaDon = @maHoaDon, maPhuTung = @maPhuTung where maChiTietHoaDonPhuTung = @maChiTietHoaDonPhuTung

\*Code C#

public bool CapNhatChiTietHoaDonPhuTung(string maChiTietHDPT, string maHoaDon, string maPhuTung, float soTienDaTra)

{

string query = string.Format("exec Update\_CHITIETHOADONPHUTUNG @maChiTietHoaDonPhuTung , @maHoaDon , @maPhuTung , @soTienDaTra ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiTietHDPT, maHoaDon, maPhuTung, soTienDaTra});

return result > 0;

}

**d. Procedure Xóa Chi Tiết Hóa Đơn Phụ Tùng**

create or alter proc Delete\_CHITIETHOADONPHUTUNG @maChiTietHoaDonPhuTung nvarchar(20)

as

delete CHITIETHOADONPHUTUNG where maChiTietHoaDonPhuTung = @maChiTietHoaDonPhuTung

\*Code C#

public bool XoaChiTietHoaDonPhuTung(string maChiTietHDPT)

{

string query = string.Format("exec Delete\_CHITIETHOADONPHUTUNG @maChiTietHoaDonPhuTung ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maChiTietHDPT });

return result > 0;

}

### Chi Tiết Hóa Đơn Xe

**a. Procedure Xem thông tin Chi Tiết Hóa Đơn Xe**

create or alter proc List\_CHITIETHOADONXE

as

select \* from CHITIETHOADONXE

\*Code C#

public List<ChiTietHoaDonXe> DanhSachCTHDXe()

{

List<ChiTietHoaDonXe> listHoaDonXe = new List<ChiTietHoaDonXe>();

string query = "exec list\_CHITIETHOADONXE";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach(DataRow row in data.Rows)

{

ChiTietHoaDonXe hoaDonXe = new ChiTietHoaDonXe(row);

listHoaDonXe.Add(hoaDonXe);

}

return listHoaDonXe;

}

**b. Procedure Thêm Chi Tiết Hóa Đơn Xe**

create or alter proc Insert\_CHITIETHOADONXE @maHoaDon nvarchar(20), @maXe nvarchar(20), @ngayNhanXe nvarchar(50), @soTienDaTra money, @phiDangKyBienSo money ,@phiDangKiem money , @phiTruocBa money, @phiBaoHiemTrachNhiemDanSu money, @phiSuDungDuongBo money

as

insert into CHITIETHOADONXE (maHoaDon, maXe, ngayNhanXe, soTienDaTra, phiDangKyBienSo, phiDangKiem, phiTruocBa, phiBaoHiemTrachNhiemDanSu, phiSuDungDuongBo)

values (@maHoaDon, @maXe, @ngayNhanXe, @soTienDaTra, @phiDangKyBienSo, @phiDangKiem, @phiTruocBa, @phiBaoHiemTrachNhiemDanSu, @phiSuDungDuongBo)

\* Code C#

public bool ThemChiTietHoaDonXe(string maHoaDon, string maXe, string ngayNhanXe, float soTienDaTra, float phiDangKyBienSo, float phiDangKiem, float phiTruocBa, float phiBaoHiemTrachNhiemDanSu, float phiSuDungDuongBo)

{

string query = string.Format("exec Insert\_CHITIETHOADONXE @maHoaDon , @maXe , @ngayNhanXe , @soTienDaTra , @phiDangKyBienSo , @phiDangKiem , @phiTruocBa , @phiBaoHiemTrachNhiemDanSu , @phiSuDungDuongBo ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maHoaDon, maXe, ngayNhanXe.Substring(0, 10), soTienDaTra, phiDangKyBienSo, phiDangKiem, phiTruocBa, phiBaoHiemTrachNhiemDanSu, phiSuDungDuongBo });

return result > 0;

}

**c. Procedure Cập nhật Chi Tiết Hóa Đơn Xe**

create or alter proc Update\_CHITIETHOADONXE @maChiTietHoaDonXe nvarchar(20), @maHoaDon nvarchar(20), @maXe nvarchar(20), @ngayNhanXe nvarchar(50), @soTienDaTra money, @phiDangKyBienSo money, @phiDangKiem money, @phiTruocBa money, @phiBaoHiemTrachNhiemDanSu money, @phiSuDungDuongBo money

as

update CHITIETHOADONXE set maHoaDon = @maHoaDon, maXe = @maXe, ngayNhanXe = @ngayNhanXe, soTienDaTra = @soTienDaTra, phiDangKyBienSo = @phiDangKyBienSo, phiDangKiem = @phiDangKiem, phiTruocBa = @phiTruocBa, phiBaoHiemTrachNhiemDanSu = @phiBaoHiemTrachNhiemDanSu, phiSuDungDuongBo = @phiSuDungDuongBo where maChiTietHoaDonXe = @maChiTietHoaDonXe

\*Code C#

public bool CapNhatChiTietHoaDonXe(string maChiTietHoaDonXe, string maHoaDon, string maXe, string ngayNhanXe, float soTienDaTra, float phiDangKyBienSo, float phiDangKiem, float phiTruocBa, float phiBaoHiemTrachNhiemDanSu, float phiSuDungDuongBo)

{

string query = string.Format("exec Update\_CHITIETHOADONXE @maChiTietHoaDonXe , @maHoaDon , @maXe , @ngayNhanXe , @soTienDaTra , @phiDangKyBienSo , @phiDangKiem , @phiTruocBa , @phiBaoHiemTrachNhiemDanSu , @phiSuDungDuongBo ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maHoaDon, maXe, ngayNhanXe.Substring(0, 10), soTienDaTra, phiDangKyBienSo, phiDangKiem, phiTruocBa, phiBaoHiemTrachNhiemDanSu, phiSuDungDuongBo, maChiTietHoaDonXe });

return result > 0;

}

**d. Procedure Xóa Chi Tiết Hóa Đơn Xe**

create or alter proc Delete\_CHITIETHOADONXE @maChiTietHoaDonXe nvarchar(20)

as

delete CHITIETHOADONXE where maChiTietHoaDonXe = @maChiTietHoaDonXe

\*Code C#

public bool XoaChiTietHoaDonXe(string maChiTietHoaDonXe)

{

string query = string.Format("exec Delete\_CHITIETHOADONXE @maChiTietHoaDonXe ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiTietHoaDonXe });

return result > 0;

}

### Phiếu Nhập

**a. Procedure Xem thông tin Phiếu Nhập**

create or alter proc List\_PhieuNhap

as

select \* from PHIEUNHAP

\*Code C#

public List<PhieuNhap> DanhSachPhieuNhap()

{

List<PhieuNhap> listPhieuNhap = new List<PhieuNhap>();

string query = "exec List\_PhieuNhap";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

PhieuNhap phieuNhap = new PhieuNhap(row);

listPhieuNhap.Add(phieuNhap);

}

return listPhieuNhap;

}

**b. Procedure Thêm Phiếu Nhập**

create or alter proc Insert\_PhieuNhap @ngayNhap date, @maChiNhanh nvarchar(20), @maNhaCungCap nvarchar(20)

as

insert into PHIEUNHAP(ngayNhap, maChiNhanh, maNhaCungCap) values(@ngayNhap, @maChiNhanh, @maNhaCungCap)

\*Code C#

public bool ThemPhieuNhap(string ngayNhap, string maChiNhanh, string maNhaCungCap)

{

string query = string.Format("exec Insert\_PhieuNhap @ngayNhap , @maChiNhanh , @maNhaCungCap ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {ngayNhap.Substring(0, 10), maChiNhanh, maNhaCungCap });

return result > 0;

}

**c. Procedure Cập nhật Phiếu Nhập**

create or alter proc Update\_PhieuNhap @maPhieuNhap nvarchar(20), @ngayNhap date, @maChiNhanh nvarchar(20), @maNhaCungCap nvarchar(20)

as

update PHIEUNHAP set ngayNhap = @ngayNhap, maChiNhanh = @maChiNhanh, maNhaCungCap = @maNhaCungCap where maPhieuNhap = @maPhieuNhap

\*Code C#

public bool CapNhatPhieuNhap(string maPhieuNhap, string ngayNhap, string maChiNhanh, string maNhaCungCap)

{

string query = string.Format("exec Update\_PhieuNhap @maPhieuNhap , @ngayNhap , @maChiNhanh , @maNhaCungCap ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maPhieuNhap, ngayNhap.Substring(0, 10), maChiNhanh, maNhaCungCap });

return result > 0;

}

**d. Procedure Xóa Phiếu Nhập**

create or alter proc Delete\_PhieuNhap @maPhieuNhap nvarchar(20)

as

delete PHIEUNHAP where maPhieuNhap = @maPhieuNhap

\*Code C#

public bool XoaPhieuNhap(string maPhieuNhap)

{

string query = string.Format("exec Delete\_PhieuNhap @maPhieuNhap ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maPhieuNhap });

return result > 0;

}

### Chi Tiết Phiếu Nhập Phụ Tùng

**a. Procedure Xem thông tin Chi Tiết Phiếu Nhập Phụ Tùng**

create or alter proc List\_CHITIETPHIEUNHAPPHUTUNG

as

select \* from CHITIETPHIEUNHAPPHUTUNG

\* Code C#

public List<ChiTietPhieuNhapPhuTung> DanhSachChiTietPhieuNhapPhuTung()

{

List<ChiTietPhieuNhapPhuTung> listDanhSachCTPNPTung = new List<ChiTietPhieuNhapPhuTung> ();

string query = "exec list\_CHITIETPHIEUNHAPPHUTUNG";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

ChiTietPhieuNhapPhuTung ctpnPTung = new ChiTietPhieuNhapPhuTung(row);

listDanhSachCTPNPTung.Add(ctpnPTung);

}

return listDanhSachCTPNPTung;

}

**b. Procedure Thêm Chi Tiết Phiếu Nhập Phụ Tùng**

create or alter proc Insert\_CHITIETPHIEUNHAPPHUTUNG @maPhuTung nvarchar(20), @maPhieuNhap nvarchar(20), @giaNhap float, @soLuong int

as

insert into CHITIETPHIEUNHAPPHUTUNG (maPhuTung, maPhieuNhap, giaNhap, soLuong) values (@maPhuTung, @maPhieuNhap, @giaNhap, @soLuong)

\*Code C#

public bool ThemChiTietPhieuNhapPhuTung(string maPhuTung, string maPhieuNhap, float giaNhap, int soLuong)

{

string query = string.Format("exec Insert\_CHITIETPHIEUNHAPPHUTUNG @maPhuTung , @maPhieuNhap , @giaNhap , @soLuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maPhuTung, maPhieuNhap, giaNhap, soLuong });

return result > 0;

}

**c. Procedure Cập nhật Chi Tiết Phiếu Nhập Phụ Tùng**

create or alter proc Update\_CHITIETPHIEUNHAPPHUTUNG @maChiTietPhieuNhapPhuTung nvarchar(20), @maPhuTung nvarchar(20), @maPhieuNhap nvarchar(20), @giaNhap float, @soLuong int

as

update CHITIETPHIEUNHAPPHUTUNG set maPhuTung = @maPhuTung, maPhieuNhap = @maPhieuNhap, giaNhap = @giaNhap, soLuong = @soLuong where maChiTietPhieuNhapPhuTung = @maChiTietPhieuNhapPhuTung

\* Code C#

public bool CapNhatChiTietPhieuNhapPhuTung(string maChiTietPNPT, string maPhuTung, string maPhieuNhap, float giaNhap, int soLuong)

{

string query = string.Format("exec Update\_CHITIETPHIEUNHAPPHUTUNG @maChiTietPhieuNhapPhuTung , @maPhuTung , @maPhieuNhap , @giaNhap , @soLuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiTietPNPT, maPhuTung, maPhieuNhap, giaNhap, soLuong });

return result > 0;

}

**d. Procedure Xóa Chi Tiết Phiếu Nhập Phụ Tùng**

create or alter proc Delete\_CHITIETPHIEUNHAPPHUTUNG @maChiTietPhieuNhapPhuTung nvarchar(20)

as

delete CHITIETPHIEUNHAPPHUTUNG where maChiTietPhieuNhapPhuTung = @maChiTietPhieuNhapPhuTung

\*Code C#

public bool XoaChiTietPhieuNhapPhuTung(string maChiTietPNPT)

{

string query = string.Format("exec Delete\_CHITIETPHIEUNHAPPHUTUNG @maChiTietPhieuNhapPhuTung ", maChiTietPNPT);

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maChiTietPNPT });

return result > 0;

}

### Chi Tiết Phiếu Nhập Xe

**a. Procedure Xem thông tin Chi Tiết Phiếu Nhập Xe**

create or alter proc List\_CHITIETPHIEUNHAPXE

as

select \* from CHITIETPHIEUNHAPXE

\* Code C#

public List<ChiTietPhieuNhapXe> DanhSachChiTietPhieuNhapXe()

{

List<ChiTietPhieuNhapXe> listCTPNXe = new List<ChiTietPhieuNhapXe> ();

string query = "exec list\_CHITIETPHIEUNHAPXE";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

ChiTietPhieuNhapXe ctpnXe = new ChiTietPhieuNhapXe(row);

listCTPNXe.Add(ctpnXe);

}

return listCTPNXe;

}

**b. Procedure Thêm Chi Tiết Phiếu Nhập Xe**

create or alter proc Insert\_CHITIETPHIEUNHAPXE @maLoXe nvarchar(20), @maPhieuNhap nvarchar(20), @giaNhap float, @soLuong int

as

insert into CHITIETPHIEUNHAPXE (maLoXe, maPhieuNhap, giaNhap, soLuong) values (@maLoXe, @maPhieuNhap, @giaNhap, @soLuong)

\* Code C#

public bool ThemChiTietPhieuNhapXe(string maLoXe, string maPhieuNhap, float giaNhap, int soLuong)

{

string query = string.Format("exec Insert\_CHITIETPHIEUNHAPXE @maLoXe , @maPhieuNhap , @giaNhap , @soLuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maLoXe, maPhieuNhap, giaNhap, soLuong });

return result > 0;

}

**c. Procedure Cập nhật Chi Tiết Phiếu Nhập Xe**

create or alter proc Update\_CHITIETPHIEUNHAPXE @maChiTietPhieuNhapXe nvarchar(20), @maLoXe nvarchar(20), @maPhieuNhap nvarchar(20), @giaNhap float, @soLuong int

as

update CHITIETPHIEUNHAPXE set maLoXe = @maLoXe, maPhieuNhap = @maPhieuNhap, giaNhap = @giaNhap, soLuong = @soLuong where maChiTietPhieuNhapXe = @maChiTietPhieuNhapXe

\* Code C#

public bool CapNhatChiTietPhieuNhapXe(string maChiTietPhieuNhapXe, string maLoXe, string maPhieuNhap, float giaNhap, int soLuong)

{

string query = string.Format("exec Update\_CHITIETPHIEUNHAPXE @maChiTietPhieuNhapXe , @maLoXe , @maPhieuNhap , @giaNhap , @soLuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiTietPhieuNhapXe, maLoXe, maPhieuNhap, giaNhap, soLuong});

return result > 0;

}

**d. Procedure Xóa Chi Tiết Phiếu Nhập Xe**

create or alter proc Delete\_CHITIETPHIEUNHAPXE @maChiTietPhieuNhapXe nvarchar(20)

as

delete CHITIETPHIEUNHAPXE where maChiTietPhieuNhapXe = @maChiTietPhieuNhapXe

\* Code C#

public bool XoaChiTietPhieuNhapXe(string maChiTietPhieuNhapXe)

{

string query = string.Format("exec Delete\_CHITIETPHIEUNHAPXE @maChiTietPhieuNhapXe ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maChiTietPhieuNhapXe });

return result > 0;

}

### Nhà Cung Cấp

**a. Procedure Xem thông tin Nhà Cung Cấp**

create or alter proc List\_NhaCungCap

as

select \* from NHACUNGCAP

\* Code C#

public List<NhaCungCap> DanhSachNhaCungCap()

{

List<NhaCungCap> listNhaCungCap = new List<NhaCungCap>();

string query = "exec List\_NhaCungCap";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

NhaCungCap nhaCungCap = new NhaCungCap(row);

listNhaCungCap.Add(nhaCungCap);

}

return listNhaCungCap;

}

**b. Procedure Thêm Nhà Cung Cấp**

create or alter proc Insert\_NhaCungCap @maNhaCungCap nvarchar(20), @tenNhaCungCap nvarchar(50), @diaChi nvarchar(255), @soDienThoai nvarchar(20)

as

insert into NHACUNGCAP(maNhaCungCap, tenNhaCungCap, diaChi, soDienThoai) values(@maNhaCungCap, @tenNhaCungCap, @diaChi, @soDienThoai)

\* Code C#

public bool ThemNhaCungCap(string maNhaCungCap, string tenNhaCungCap, string diaChi, string soDienThoai)

{

string query = string.Format("exec Insert\_NhaCungCap @maNhaCungCap , @tenNhaCungCap , @diaChi , @soDienThoai ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maNhaCungCap, tenNhaCungCap, diaChi, soDienThoai });

return result > 0;

}

**c. Procedure Cập nhật Nhà Cung Cấp**

create or alter proc Update\_NhaCungCap @maNhaCungCap nvarchar(20), @tenNhaCungCap nvarchar(50), @diaChi nvarchar(255), @soDienThoai nvarchar(20)

as

Update NHACUNGCAP set tenNhaCungCap = @tenNhaCungCap, diaChi = @diaChi, soDienThoai = @soDienThoai where maNhaCungCap = @maNhaCungCap

\* Code C#

public bool CapNhatNhaCungCap(string maNhaCungCap, string tenNhaCungCap, string diaChi, string soDienThoai)

{

string query = string.Format("exec Update\_NhaCungCap @maNhaCungCap , @tenNhaCungCap , @diaChi , @soDienThoai ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maNhaCungCap, tenNhaCungCap, diaChi, soDienThoai });

return result > 0;

}

**d. Procedure Xóa Nhà Cung Cấp**

create or alter proc Delete\_NhaCungCap @maNhaCungCap nvarchar(20)

as

delete NHACUNGCAP where maNhaCungCap = @maNhaCungCap

\* Code C#

public bool XoaNhaCungCap(string maNhaCungCap)

{

string query = string.Format("exec Delete\_NhaCungCap @maNhaCungCap ");

return DataProvider.Instance.ExcuteNonQuery(query, new object[] { maNhaCungCap }) > 0;

}

### Phụ Tùng

**a. Procedure Xem thông tin Phụ Tùng**

create or alter proc List\_PhuTung

as

select \* from PHUTUNG

\* Code C#

public List<PhuTung> DanhSachPhuTung()

{

List<PhuTung> listPhuTung = new List<PhuTung>();

string query = "exec List\_PhuTung";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach(DataRow row in data.Rows)

{

PhuTung phuTung = new PhuTung(row);

listPhuTung.Add(phuTung);

}

return listPhuTung;

}

**b. Procedure Thêm Phụ Tùng**

create or alter proc Insert\_PhuTung @loaiPhuTung nvarchar(50), @tenPhuTung nvarchar(50), @thuongHieu nvarchar(50), @xuatXu nvarchar(50), @giaBan float, @chatLuong nvarchar(50), @hinhAnh nvarchar(200)

as

insert into PHUTUNG(loaiPhuTung, tenPhuTung, thuongHieu, xuatXu, giaBan, chatLuong, hinhAnh) values (@loaiPhuTung, @tenPhuTung, @thuongHieu, @xuatXu, @giaBan, @chatLuong, @hinhAnh)

\* Code C#

public bool ThemPhuTung(string loaiPhuTung, string tenPhuTung, string thuongHieu, string xuatXu, float giaBan, string chatLuong, string hinhAnh)

{

string query = string.Format("exec Insert\_PhuTung @loaiPhuTung , @tenPhuTung , @thuongHieu , @xuatXu , @giaBan , @chatLuong , @hinhAnh ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {loaiPhuTung, tenPhuTung, thuongHieu, xuatXu, giaBan, chatLuong, hinhAnh });

return result > 0;

}

**c. Procedure Cập nhật Phụ Tùng**

create or alter proc Update\_PhuTung @maPhuTung nvarchar(20), @loaiPhuTung nvarchar(50), @tenPhuTung nvarchar(50), @thuongHieu nvarchar(50), @xuatXu nvarchar(50), @giaBan float, @chatLuong nvarchar(50), @hinhAnh nvarchar(200)

as

Update PHUTUNG set loaiPhuTung = @loaiPhuTung, tenPhuTung = @tenPhuTung, thuongHieu = @thuongHieu, xuatXu = @xuatXu, giaBan = @giaBan, chatLuong = @chatLuong, hinhAnh = @hinhAnh where maPhuTung = @maPhuTung

\*Code C#

public bool CapNhatPhuTung(string maPhuTung, string loaiPhuTung, string tenPhuTung, string thuongHieu, string xuatXu, float giaBan, string chatLuong, string hinhAnh)

{

string query = string.Format("exec Update\_PhuTung @maPhuTung , @loaiPhuTung , @tenPhuTung , @thuongHieu , @xuatXu , @giaBan , @chatLuong , @hinhAnh ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maPhuTung, loaiPhuTung, tenPhuTung, thuongHieu, xuatXu, giaBan, chatLuong, hinhAnh });

return result > 0;

}

**d. Procedure Xóa Phụ Tùng**

create or alter proc Delete\_PhuTung @maPhuTung nvarchar(20)

as

delete PHUTUNG where maPhuTung = @maPhuTung

\* Code C#

public bool XoaPhuTung(string maPhuTung)

{

string query = string.Format("exec Delete\_PhuTung @maPhuTung ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maPhuTung });

return result > 0;

}

### Dịch Vụ Bảo Dưỡng

**a. Procedure Xem thông tin Dịch Vụ Bảo Dưỡng**

create or alter proc List\_DICHVUBAODUONG

as

select \* from DICHVUBAODUONG

\* Code C#

public List<DichVuBaoDuong> DanhSachDichVuBaoDuong()

{

List<DichVuBaoDuong> listBaoDuong = new List<DichVuBaoDuong>();

string query = "exec list\_DICHVUBAODUONG";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

DichVuBaoDuong baoDuong = new DichVuBaoDuong(row);

listBaoDuong.Add(baoDuong);

}

return listBaoDuong;

}

**b. Procedure Thêm Dịch Vụ Bảo Dưỡng**

create or alter proc Insert\_DICHVUBAODUONG @tenBaoDuong nvarchar(50), @loaiBaoDuong nvarchar(50), @phiBaoDuong float

as

insert into DICHVUBAODUONG (tenBaoDuong, loaiBaoDuong, phiBaoDuong) values (@tenBaoDuong, @loaiBaoDuong, @phiBaoDuong)

\* Code C#

public bool ThemDichVuBaoDuong(string tenBaoDuong, string loaiBaoDuong, float phiBaoDuong)

{

string query = string.Format("exec Insert\_DICHVUBAODUONG @tenBaoDuong , @loaiBaoDuong , @phiBaoDuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {tenBaoDuong, loaiBaoDuong, phiBaoDuong });

return result > 0;

}

**c. Procedure Cập nhật Dịch Vụ Bảo Dưỡng**

create or alter proc Update\_DICHVUBAODUONG @maBaoDuong nvarchar(20), @tenBaoDuong nvarchar(50), @loaiBaoDuong nvarchar(50), @phiBaoDuong float

as

update DICHVUBAODUONG set tenBaoDuong = @tenBaoDuong, loaiBaoDuong = @loaiBaoDuong, phiBaoDuong = @phiBaoDuong, maBaoDuong = @maBaoDuong

\* Code C#

public bool CapNhatDichVuBaoDuong(string maBaoDuong, string tenBaoDuong, string loaiBaoDuong, float phiBaoDuong)

{

string query = string.Format("exec Update\_DICHVUBAODUONG @maBaoDuong , @tenBaoDuong , @loaiBaoDuong , @phiBaoDuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maBaoDuong, tenBaoDuong, loaiBaoDuong, phiBaoDuong });

return result > 0;

}

**d. Procedure Xóa Dịch Vụ Bảo Dưỡng**

create or alter proc Delete\_DICHVUBAODUONG @maBaoDuong nvarchar(20)

as

delete DICHVUBAODUONG where maBaoDuong = @maBaoDuong

\* Code C#

public bool XoaDichVuBaoDuong(string maBaoDuong)

{

string query = string.Format("exec Delete\_DICHVUBAODUONG @maBaoDuong ");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maBaoDuong });

return result > 0;

}

### Chi Tiết Phiếu Bảo Dưỡng

**a. Procedure Xem thông tin Chi Tiết Phiếu Bảo Dưỡng**

create or alter proc List\_CHITIETPHIEUBAODUONG

as

select \* from CHITIETPHIEUBAODUONG

\* Code C#

public List<ChiTietPhieuBaoDuong> DanhSachChiTieuPhieuBaoDuong()

{

List<ChiTietPhieuBaoDuong> listChiTietBaoDuong = new List<ChiTietPhieuBaoDuong>();

string query = "exec list\_CHITIETPHIEUBAODUONG";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach (DataRow row in data.Rows)

{

ChiTietPhieuBaoDuong chiTietPhieuBaoDuong = new ChiTietPhieuBaoDuong(row);

listChiTietBaoDuong.Add(chiTietPhieuBaoDuong);

}

return listChiTietBaoDuong;

}

**b. Procedure Thêm Chi Tiết Phiếu Bảo Dưỡng**

create or alter proc Insert\_CHITIETPHIEUBAODUONG @maBaoDuong nvarchar(20), @maPhieuBaoDuong nvarchar(20), @thanhTien float

as

insert into CHITIETPHIEUBAODUONG (maBaoDuong, maPhieuBaoDuong, thanhTien) values (@maBaoDuong, @maPhieuBaoDuong, @thanhTien)

\* Code C#

public bool ThemChiTietPhieuBaoDuong(string maBaoDuong, string maPhieuBaoDuong, float thanhTien)

{

string query = "exec Insert\_CHITIETPHIEUBAODUONG @maBaoDuong , @maPhieuBaoDuong , @thanhTien ";

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maBaoDuong, maPhieuBaoDuong, thanhTien});

return result > 0;

}

**c. Procedure Cập nhật Chi Tiết Phiếu Bảo Dưỡng**

create or alter proc Update\_CHITIETPHIEUBAODUONG @maChiTietPhieuBaoDuong nvarchar(20), @maBaoDuong nvarchar(20), @maPhieuBaoDuong nvarchar(20), @thanhTien float

as

update CHITIETPHIEUBAODUONG set maBaoDuong = @maBaoDuong, maPhieuBaoDuong = @maPhieuBaoDuong, thanhTien = @thanhTien where maChiTietPhieuBaoDuong = @maChiTietPhieuBaoDuong

\* Code C#

public bool CapNhatChiTietPhieuBaoDuong(string maChiTietPhieuBD, string maBaoDuong, string maPhieuBaoDuong, float thanhTien)

{

string query = "exec Update\_CHITIETPHIEUBAODUONG @maChiTietPhieuBaoDuong , @maBaoDuong , @maPhieuBaoDuong , @thanhTien ";

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maChiTietPhieuBD, maBaoDuong, maPhieuBaoDuong, thanhTien });

return result > 0;

}

**d. Procedure Xóa Chi Tiết Phiếu Bảo Dưỡng**

create or alter proc Delete\_CHITIETPHIEUBAODUONG @maChiTietPhieuBaoDuong nvarchar(20)

as

delete CHITIETPHIEUBAODUONG where maChiTietPhieuBaoDuong = @maChiTietPhieuBaoDuong

\* Code C#

public bool XoaChiTietPhieuBaoDuong(string maChiTietPhieuBD)

{

string query = "exec Delete\_CHITIETPHIEUBAODUONG @maChiTietPhieuBaoDuong ";

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maChiTietPhieuBD });

return result > 0;

}

### Khách Hàng

**a. Procedure Xem thông tin Khách Hàng**

create or alter proc List\_KhachHang

as

select \* from KHACHHANG

\* Code C#

public List<KhachHang> DanhSachKhachHang()

{

List<KhachHang> listKhachHang = new List<KhachHang>();

string query = "exec List\_KhachHang";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

foreach(DataRow row in data.Rows)

{

KhachHang khachHang = new KhachHang(row);

listKhachHang.Add(khachHang);

}

return listKhachHang;

}

**b. Procedure Thêm Khách Hàng**

create or alter proc Insert\_KhachHang @hoTenKhachHang nvarchar(50), @ngaySinh date, @gioiTinh nvarchar(5), @CCCD nvarchar(20), @diaChi nvarchar(255), @soDienThoai nvarchar(20)

as

insert into KHACHHANG(hoTenKhachHang, ngaySinh, gioiTinh, CCCD, diaChi, soDienThoai) values (@hoTenKhachHang, @ngaySinh, @gioiTinh, @CCCD, @diaChi, @soDienThoai)

\* Code C#

public bool ThemKhachHang(string hoTenKH, string ngaySinh, string gioiTinh, string CCCD, string diaChi, string soDienThoai)

{

string query = string.Format("exec Insert\_KhachHang @hoTenKhachHang , @ngaySinh , @gioiTinh , @CCCD , @diaChi , @soDienThoai");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { hoTenKH, ngaySinh.ToString().Substring(0, 10), gioiTinh, CCCD, diaChi, soDienThoai });

return result > 0;

}

**c. Procedure Cập nhật Khách Hàng**

create or alter proc Update\_KhachHang @maKhachHang nvarchar(20), @hoTenKhachHang nvarchar(50), @ngaySinh date, @gioiTinh nvarchar(5), @CCCD nvarchar(20), @diaChi nvarchar(255), @soDienThoai nvarchar(20)

as

Update KHACHHANG set hoTenKhachHang = @hoTenKhachHang, ngaySinh = @ngaySinh, gioiTinh = @gioiTinh, CCCD = @CCCD, diaChi = @diaChi, soDienThoai = @soDienThoai where maKhachHang = @maKhachHang

\* Code C#

public bool CapNhatKhachHang(string maKhachHang, string hoTenKH, string ngaySinh, string gioiTinh, string CCCD, string diaChi, string soDienThoai)

{

string query = string.Format("exec Update\_KhachHang @maKhachHang , @hoTenKhachHang , @ngaySinh , @gioiTinh , @CCCD , @diaChi , @soDienThoai");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] {maKhachHang, hoTenKH, ngaySinh.ToString().Substring(0, 10), gioiTinh, CCCD, diaChi, soDienThoai});

return result > 0;

}

**d. Procedure Xóa Khách Hàng**

create or alter proc Delete\_KhachHang @maKhachHang nvarchar(20)

as

delete KHACHHANG where maKhachHang = @maKhachHang

\* Code C#

public bool XoaKhachHang(string maKhachHang)

{

string query = string.Format("exec Delete\_KhachHang @maKhachHang");

int result = DataProvider.Instance.ExcuteNonQuery(query, new object[] { maKhachHang });

return result > 0;

}

1. **Kết nối cơ sở dữ liệu**

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Data.SqlTypes;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Security.Cryptography.Pkcs;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using DBMS\_CodeDoAn.DTO;

namespace DBMS\_CodeDoAn.DAO

{

public class DataProvider

{

private static DataProvider instance;

public static DataProvider Instance

{

get { if (instance == null) instance = new DataProvider(); return instance; }

private set => instance = value;

}

private string strCon = @"Data Source=.;Initial Catalog=DBMS\_DOAN\_QUANLYCUAHANGXE;Integrated Security=True";

private DataProvider() { }

public DataTable ExcuteQuery(string query, object[] parameter = null)

{

DataTable data = new DataTable();

using (SqlConnection sqlCon = new SqlConnection(strCon))

{

sqlCon.Open();

SqlCommand cmd = new SqlCommand(query, sqlCon);

if (parameter != null)

{

// Split query

string[] listPara = query.Split(' ');

int idx = 0;

foreach (string para in listPara)

{

if (para.Contains('@'))

{

cmd.Parameters.AddWithValue(para, parameter[idx]);

idx++;

}

}

}

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

adapter.Fill(data);

sqlCon.Close();

}

return data;

}

public int ExcuteNonQuery(string query, object[] parameters = null)

{

int data = 0;

using (SqlConnection sqlCon = new SqlConnection(strCon))

{

sqlCon.Open();

SqlCommand cmd = new SqlCommand(query, sqlCon);

if (parameters != null)

{

string[] listPara = query.Split(' ');

int idx = 0;

foreach(string para in listPara)

{

if (para.Contains('@'))

{

cmd.Parameters.AddWithValue(para, parameters[idx]);

idx++;

}

}

}

data = cmd.ExecuteNonQuery();

sqlCon.Close();

}

return data;

}

public object ExcuteScalar(string query, object[] parameters = null)

{

object data = 0;

using (SqlConnection sqlCon = new SqlConnection(strCon))

{

sqlCon.Open();

SqlCommand cmd = new SqlCommand(query, sqlCon);

if (parameters != null)

{

string[] listPara = query.Split(' ');

int idx = 0;

foreach(string para in listPara)

{

if (para.Contains('@'))

{

cmd.Parameters.AddWithValue(para, parameters[idx]);

idx++;

}

}

}

data = cmd.ExecuteScalar();

}

return data;

}

public byte[] ConvertImageToByteArray(Image image)

{

try

{

if (image == null) { return null; }

using (MemoryStream stream = new MemoryStream())

{

image.Save(stream, System.Drawing.Imaging.ImageFormat.Png);

return stream.ToArray();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

return null;

}

}

public Image ConvertByteArrayToImage(byte[] byteArray)

{

if (byteArray == null)

{

return null;

}

using (MemoryStream stream = new MemoryStream(byteArray))

{

return Image.FromStream(stream);

}

}

}

}