

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

BỘ MÔN: CÔNG NGHỆ THÔNG TIN



BÁO CÁO BÀI TẬP LỚN HỆ QUẢN

TRỊ CƠ SỞ DỮ LIỆU

ĐỀ TÀI: QUẢN LÝ NHÀ THUỐC

Giảng viên hướng dẫn	: ThS. ĐỖ DUY CÔP
Họ tên sinh viên	: ĐỖ ĐỨC CHUNG
MSSV	: K215480106085
Lớp	: K57KMT.01

Thái Nguyên 2024

PHIẾU GIAO ĐỀ TÀI MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Sinh viên: Đỗ Đức Chung

MSSV: K215480106085

Lớp: K57KMT

Ngành: Kỹ thuật máy tính

Giáo viên hướng dẫn: ThS. Đỗ Duy Cốp

1. Tên đề tài: **“Quản lý nhà thuốc”**.

2. Yêu cầu của phần mềm, ứng dụng:

- Tạo cơ sở dữ liệu cho bài toán gồm các bảng và thuộc tính bằng việc phân tích

3. Các sản phẩm, kết quả:

- Bản báo cáo, và cơ sở dữ liệu.

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

ThS. Đỗ Duy Cốp

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

Thái Nguyên, ngày...tháng.....năm.....

GIÁO VIÊN HƯỚNG DẪN

(Ký ghi rõ họ tên)

Github cá nhân

- Link: https://github.com/Chung2310/TL_HQTCSDL



- QR: S

Thông tin cá nhân

- Họ và Tên: Đỗ Đức Chung
- MSSV: K215480106085
- Lớp: K57KMT
- Môn học: Hệ quản trị cơ sở dữ liệu

Tên bài toán

- Đề tài: Quản lý nhà thuốc An Khang

Các chức năng

Bài toán quản lý sản phẩm trong nhà thuốc nhằm đảm bảo quản lý hiệu quả và tối ưu hóa kho hàng

- Quản lý thông tin sản phẩm: Thêm sản phẩm, sửa thông tin sản phẩm, xóa sản phẩm, liệt kê hàng hóa.
- Quản lý thông tin đơn hàng: Thêm đơn hàng, sửa đơn hàng, xóa đơn hàng, liệt kê đơn hàng.

- Quản lý thông tin khách hàng: Thêm thông tin khách hàng, sửa thông tin khách hàng, xóa thông tin khách hàng, liệt kê danh sách các mặt hàng mà khách hàng đã mua
- Quản lý thông tin nhà cung cấp: Thêm thông tin nhà cung cấp, sửa thông tin nhà cung cấp, xóa thông tin nhà cung cấp, liệt kê danh sách các mặt hàng mà nhà cung cấp đó đã cung cấp
- Quản lý nhân viên, người quản lý: Thêm thông tin nhân viên, sửa thông tin nhân viên, xóa thông tin nhân viên, liệt kê các hóa đơn mà nhân viên đã bán,...
- Quản lý việc thông kê sản phẩm: Thêm phiếu thông kê, sửa thông tin phiếu thông kê, xóa phiếu thông kê,...

Báo cáo

- Báo cáo sản phẩm sắp hết hạn
- Báo cáo các sản phẩm tồn kho
- Tính Tổng Doanh Thu

Các bảng của hệ thống

- Bảng NhanVien(MaNhanVien, TenNhanVien, GioiTinh, NgaySinh, DiaChi, SDT, LuongCoBan, PhuCap, TaiKhoan, MatKhau, MaQuanLy);

```
CREATE TABLE NHANVIEN
(
    MANHANVIEN NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_NHANVIEN
    PRIMARY KEY (MANHANVIEN),
    TENNHANVIEN NVARCHAR(50),
    GIOITINH NVARCHAR(10),
    NGAYSINH DATETIME,
    DAICHI NVARCHAR(100),
    SDT NVARCHAR(10),
    LUONGCOBAN BIGINT,
    PHUCAP BIGINT,
    TAIKHOAN NVARCHAR(50),
    MATKHAU NVARCHAR(50),
    MAQUANLY NVARCHAR(50),
    CHECK (LUONGCOBAN >= 0),
    CHECK (PHUCAP >= 0)
)
```

- Trường MaNhanVien là Primary Key vì MaNhanVien đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaNhanVien không được phép Null. Kiểu dữ liệu là nvarchar(50)

- Trường TenNhanVien: nvarchar(50)
 - Trường GioiTinh: nvarchar(10)
 - Trường NgaySinh: datetime
 - Trường DiaChi: nvarchar(100)
 - Trường SDT: nvarchar(10)
 - Trường LuongCoBan: BigInt
 - Trường PhuCap: BigInt
 - Trường TaiKhoan: nvarchar(50)
 - Trường MatKhau: nvarchar(50)
 - Trường MaQuanLy: nvarchar(50)
- Bảng QuanLy(MaQuanLy, TenQuanLy, NgaySinh, SDT, DiaChi, TaiKhoan, MatKhau);

```
CREATE TABLE QUANLY
(
    MAQUANLY NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_QUANLY
    PRIMARY KEY (MAQUANLY),
    TENQUANLY NVARCHAR(50),
    NGAYSINH DATETIME,
    SODIENTHOAI NVARCHAR(10),
    DIACHI NVARCHAR(100),
    TAIKHOAN NVARCHAR(50),
    MATKHAU NVARCHAR(50)
)
```

- Trường MaQuanLy là Primary Key vì MaQuanLy bảo đảm tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaQuanLy không được phép Null. Kiểu dữ liệu là nvarchar(50)
 - Trường TenQuanLy: nvarchar(50)
 - Trường NgaySinh: datetime
 - Trường SDT: nvarchar(50)
 - Trường DiaChi: nvarchar(100)
 - Trường TaiKhoan: nvarchar(50)
 - Trường MatKhau: nvarchar(50)
- Bảng PhieuThongKe(SoPhieuThongKe, MaNhanVien, NgayThongKe);

```
CREATE TABLE PHIEUTHONGKE
(
    SOPHIEUTHONGKE NVARCHAR(50) NOT NULL,
    CONSTRAINT PK_PHIEUTHONGKE_NHANVIEN
    PRIMARY KEY (SOPHIEUTHONGKE),
    MANHANVIEN NVARCHAR(50),
    NGAYTHONGKE DATETIME
)
```

- Trường SoPhieuThongKe là Primary Key vì SoPhieuThongKe đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => SoPhieuThongKe không được phép Null. Kiểu dữ liệu là nvarchar(50)
 - Trường MaNhanVien là Foreign Key vì MaSanPham đảm bảo mối quan hệ giữa bảng PhieuThongKe và bảng ChiTietPhieuThongKe. Kiểu dữ liệu là nvarchar(50)
 - Trường NgayThongKe: datetime
- Bảng ChiTietPhieuThongKe(SoPhieuThongKe, MaSanPham, SoLuongDaBan, TongTien);

```
CREATE TABLE CHITIETPHIEUTHONGKE
(
    SOPHIEUTHONGKE NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_CHITIETPHIEUTHONGKE
    PRIMARY KEY (SOPHIEUTHONGKE),
    MASANPHAM NVARCHAR(50),
    SOLUONGDABAN INT,
    TONGTIEN BIGINT,
    CHECK (SOLUONGDABAN >= 0),
    CHECK (TONGTIEN >= 0)
)
```

- Trường SoPhieuThongKe là Primary Key vì SoPhieuThongKe đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => SoPhieuThongKe không được phép Null. Kiểu dữ liệu là nvarchar(50)
- Trường MaSanPham là Foreign Key vì MaSanPham đảm bảo mối quan hệ giữa bảng SanPham và bảng ChiTietPhieuThongKe. Kiểu dữ liệu là nvarchar(50)
- Trường SoLuongDaBan: int. Có CK không cho phép nhập giá trị nhỏ hơn hoặc bằng 0.
- Trường TongTien: BIGINT. Có CK không cho phép nhập giá trị nhỏ hơn hoặc bằng 0.

```

CREATE PROCEDURE CalculateTongTien
AS
BEGIN
    -- Cập nhật giá trị TONGTIEN trong bảng CHITIETDONHANG
    UPDATE CHITIETDONHANG
    SET TONGTIEN = CT.SOLUONGBAN * SP.GIABAN
    FROM CHITIETDONHANG AS CT
    JOIN SANPHAM AS SP
    ON CT.MASANPHAM = SP.MASANPHAM;
END;
GO

```

- Bảng ThanhToan(MaHinhThucThanhToan, TenHinhThucThanhToan);

```

CREATE TABLE THANHTOAN
(
    MAHINHTHUCTHANHTOAN NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_THANHTOAN
    PRIMARY KEY (MAHINHTHUCTHANHTOAN),
    TENHINHTHUCTHANHTOAN NVARCHAR(50)
)

```

- Trường MaHinhThucThanhToan là Primary Key vì MaHinhThucThanhToan đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaHinhThucThanhToan không được phép Null. Kiểu dữ liệu là nvarchar(50)
 - Trường TenHinhThucThanhToan: nvarchar(50)
- Bảng DonHang(SoHoaDon, MaKhachHang, MaNhanVien, NgayBan, MaHinhThucThanhToan);

```

CREATE TABLE DONHANG
(
    SOHOADON NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_DONHANG_KHACHHANG
    PRIMARY KEY (SOHOADON),
    MAKHACHHANG NVARCHAR(50),
    MANHANVIEN NVARCHAR(50),
    NGAYBAN DATETIME,
    MAHINHTHUCTHANHTOAN NVARCHAR(50)
)

```

- Trường SoHoaDon là Primary Key vì SoHoaDon đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => SoHoaDon không được phép Null. Kiểu dữ liệu là nvarchar(50)
 - Trường MaKhachHang là Foreign Key vì MaKhachHang đảm bảo mối quan hệ giữa bảng DonHang và KháchHang. Kiểu dữ liệu là nvarchar(50)
 - Trường MaNhanVien là Foreign Key vì MaNhanVien đảm bảo mối quan hệ giữa bảng DonHang và NhânVien. Kiểu dữ liệu là nvarchar(50)

- Trường NgayBan: datetime
 - Trường MaHinhThucThanhToan là Foreign Key vì MaHinhThucThanhToan đảm bảo mỗi quan hệ giữa bảng DonHang và ThanhToan. Kiểu dữ liệu là nvarchar(50)
- Bảng ChiTietDonHang(SoHoaDon, MaSanPham, SoLuong, GiamGia, TongTien);

```
CREATE TABLE CHITIETDONHANG
(
    SOHOADON NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_CHITIETDONHANG
    PRIMARY KEY (SOHOADON),
    MASANPHAM NVARCHAR(50),
    SOLUONGBAN INT,
    GIAMGIA BIGINT,
    TONGTIEN BIGINT,
    CHECK (SOLUONGBAN >=0),
    CHECK (GIAMGIA >=0),
    CHECK (TONGTIEN >= 0)
)
```

- Trường SoHoaDon là Primary Key vì SoHoaDon bảo đảm tính duy nhất và toàn vẹn của dữ liệu trong bảng => SoHoaDon không được phép Null. Kiểu dữ liệu là nvarchar(50)
 - Trường MaSanPham là Foreign Key vì MaSanPham đảm bảo mỗi quan hệ giữa các bảng được duy trì chính xác. Kiểu dữ liệu là nvarchar(50)
 - Trường SoLuongBan: int. Có CK không cho phép nhập giá trị nhỏ hơn hoặc bằng 0.
 - Trường GiamGia: BIGINT. Có CK không cho phép nhập giá trị nhỏ hơn hoặc bằng 0.
 - Trường TongTien: BIGINT. Có CK không cho phép nhập giá trị nhỏ hơn hoặc bằng 0.
- Bảng KhachHang(MaKhachHang, TenKhachHang, GioiTinh, DiaChi, SDT);

```
CREATE TABLE KHACHHANG
(
    MAKHACHHANG NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_KHACHHANG
    PRIMARY KEY (MAKHACHHANG),
    TENKHACHHANG NVARCHAR(50),
    GIOITINH NVARCHAR(10),
    DIACHI NVARCHAR(100),
    SODIENTHOAI NVARCHAR(10)
)
```

- Trường MaKhachHang là Primary Key vì MaKhachHang đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaKhachHang không được phép Null. Kiểu dữ liệu là nvarchar(50)
- Trường TenKhachHang: nvarchar(50)
- Trường GioiTinh: nvarchar(10)
- Trường DiaChi: nvarchar(100)
- Trường SDT: nvarchar(10)
- Bảng SanPham(MaSanPham, TenSanPham, MaNhaCC, SoLuong, GiaBan, NgaySanXuat, HanSuDung);

```
CREATE TABLE SANPHAM
(
    MASANPHAM NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_SANPHAM
    PRIMARY KEY (MASANPHAM),
    TENSANPHAM NVARCHAR(50),
    MANHACC NVARCHAR(50),
    SOLUONG INT,
    GIABAN BIGINT,
    NGAYSANXUAT DATETIME,
    HANSUDUNG DATETIME,
    CHECK (SOLUONG >= 0),
    CHECK (GIABAN >= 0)
)
```

- Trường MaSanPham là Primary Key vì MaSanPham đảm bảo tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaSanPham không được phép Null. Kiểu dữ liệu là nvarchar(50)
- Trường TenSanPham: nvarchar(50)
- Trường MaNhaCC Trường MaSanPham là Foreign Key vì MaSanPham đảm bảo mối quan hệ giữa bảng SanPham và NhaCC. Kiểu dữ liệu là nvarchar(50)
- Trường SoLuong: int. Có CK không cho phép nhập giá trị nhỏ hơn 0
- Trường GiaBan: Bigint. Có CK không cho phép nhập giá trị nhỏ hơn 0
- Trường NgaySanXuat: datetime
- Trường HanSuDung: datetime. Tạo 1 trigger để kiểm soát dữ liệu của Trường NgaySanXuat và HanSuDung để NgaySanXuat luôn là ngày trước HanSuDung

```

-- Tạo trigger kiểm tra trước khi cập nhật dữ liệu trong bảng products
CREATE TRIGGER trgBeforeUpdateSanPham
ON SANPHAM
INSTEAD OF UPDATE
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE HANSUDUNG <= NGAYSANXUAT
    )
    BEGIN
        RAISERROR('HANSUDUNG phải lớn hơn NGAYSANXUAT', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END

    UPDATE SANPHAM
    SET TENSANPHAM = inserted.TENSANPHAM,
        NGAYSANXUAT = inserted.NGAYSANXUAT,
        HANSUDUNG = inserted.HANSUDUNG
    FROM inserted
    WHERE SANPHAM.MASANPHAM = inserted.MASANPHAM;
END;

```

- Bảng NhaCungCap(MaNhaCC, TenNhaCC, DiaChi, SDT, Email, NguoiDaiDien);

```

CREATE TABLE NHACUNGCAP
(
    MANHACC NVARCHAR(50) NOT NULL ,
    CONSTRAINT PK_NHACUNGCAP
    PRIMARY KEY (MANHACC),
    TENNHACC NVARCHAR(50),
    DIACHI NVARCHAR(100),
    SODIENTHOAI NVARCHAR(50),
    EMAIL NVARCHAR(50),
    NGUOIDAIDIEN NVARCHAR(50)
)

```

- Trường MaNhaCC là Primary Key vì MaNhaCC bảo đảm tính duy nhất và toàn vẹn của dữ liệu trong bảng => MaNhaCC không được phép Null. Kiểu dữ liệu là nvarchar(50)
- Trường TenNhaCC: nvarchar(50)
- Trường DiaChi: nvarchar(100)
- Trường SDT: nvarchar(10)
- Trường Email: nvarchar(50)
- Trường NguoiDaiDien: nvarchar(50)

Các Chức Năng

- Thêm Đơn Hàng

```

CREATE PROCEDURE ThemDonHangVaChiTiet
    @SoHoaDon NVARCHAR(50),
    @MaKhachHang NVARCHAR(50),
    @MaNhanVien NVARCHAR(50),
    @NgayBan DATETIME,
    @MaHinhThucThanhToan NVARCHAR(50),
    @MaSanPham NVARCHAR(50),
    @SoLuongBan INT,
    @GiamGia BIGINT,
    @TongTien BIGINT
AS
BEGIN
    -- Bắt đầu một giao dịch
    BEGIN TRANSACTION;

    -- Thêm dữ liệu vào bảng DonHang
    BEGIN TRY
        INSERT INTO DonHang (SoHoaDon, MaKhachHang, MaNhanVien, NgayBan, MaHinhThucThanhToan)
        VALUES (@SoHoaDon, @MaKhachHang, @MaNhanVien, @NgayBan, @MaHinhThucThanhToan);
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;

    -- Thêm dữ liệu vào bảng ChiTietDonHang, để TongTien NULL hoặc một giá trị mặc định
    BEGIN TRY
        INSERT INTO ChiTietDonHang (SoHoaDon, MaSanPham, SoLuongBan, GiamGia, TongTien)
        VALUES (@SoHoaDon, @MaSanPham, @SoLuongBan, @GiamGia, @TongTien);
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;

    -- Commit giao dịch nếu không có lỗi
    COMMIT TRANSACTION;
END;

```

○ Kết Quả:

	SOHOADON	MAKHACHHANG	MANHANVIEN	NGAYBAN	MAHINHTHUCTHANHTOAN
1	a102	102	10	2024-06-18 00:00:00.000	1
2	a103	103	NV1	2024-06-18 00:00:00.000	1

	SOHOADON	MASANPHAM	SOLUONGBAN	GIAMGIA	TONGTIEN
1	a102	SP1	5	0	NULL
2	a103	SP2	10	0	NULL

• Xóa Đơn Hàng

- Tạo Trigger để khi xóa 1 trường tại bảng DonHang thì sẽ xóa 1 trường cùng MaDonHang ở bảng ChiTietDonHang

```

CREATE TRIGGER trg_DeleteChiTietDonHang
ON DonHang
FOR DELETE
AS
BEGIN
    -- Xóa các bản ghi liên quan trong bảng ChiTietDonHang
    DELETE FROM ChiTietDonHang
    WHERE SoHoaDon IN (SELECT deleted.SoHoaDon FROM deleted);
END;

```

- Kết quả:

- Chạy lệnh: "delete from DONHANG where SOHOADON='a102'"

	SOHOADON	MAKHACHHANG	MANHANVIEN	NGAYBAN	MAHINHTHUCTH
1	a103	103	NV1	2024-06-18 00:00:00.000	1

	SOHOADON	MASANPHAM	SOLUONGBAN	GIAMGIA	TONGTIEN
1	a103	SP2	10	0	NULL

- Sửa Đơn Hàng

- Thông tin cũ:

	SOHOADON	MAKHACHHANG	MANHANVIEN	NGAYBAN	MAHINHTHUCTHANHTOAN
1	a101	101	NV2	2024-06-18 00:00:00.000	1
2	a103	103	NV1	2024-06-18 00:00:00.000	1

- Thông tin mới:

	SOHOADON	MAKHACHHANG	MANHANVIEN	NGAYBAN	MAHINHTHUCTHANHTOAN
1	a101	101	NV2	2024-06-18 00:00:00.000	2
2	a103	103	NV1	2024-06-18 00:00:00.000	1

- Thêm Sản Phẩm

```

CREATE PROCEDURE ThemSanPham
    @MaSanPham NVARCHAR(50),
    @TenSanPham NVARCHAR(50),
    @MaNhaCC NVARCHAR(50),
    @SoLuong INT,
    @GiaBan BIGINT,
    @NgaySanXuat DATETIME,
    @HanSuDung DATETIME
AS
BEGIN
    INSERT INTO SanPham (MaSanPham, TenSanPham, MaNhaCC, SoLuong, GiaBan, NgaySanXuat, HanSuDung)
    VALUES (@MaSanPham, @TenSanPham, @MaNhaCC, @SoLuong, @GiaBan, @NgaySanXuat, @HanSuDung);
END;

```

- Sửa Sản Phẩm

```

CREATE PROCEDURE SuaSanPham
    @MaSanPham NVARCHAR(50),
    @TenSanPham NVARCHAR(50),
    @MaNhaCC NVARCHAR(50),
    @SoLuong INT,
    @GiaBan BIGINT,
    @NgaySanXuat DATETIME,
    @HanSuDung DATETIME
AS
BEGIN
    UPDATE SanPham
    SET TenSanPham = @TenSanPham,
        MaNhaCC = @MaNhaCC,
        SoLuong = @SoLuong,
        GiaBan = @GiaBan,
        NgaySanXuat = @NgaySanXuat,
        HanSuDung = @HanSuDung
    WHERE MaSanPham = @MaSanPham;
END;

```

- Xóa Sản Phẩm

```

CREATE PROCEDURE XoaSanPham
    @MaSanPham NVARCHAR(50)
AS
BEGIN
    DELETE FROM SanPham
    WHERE MaSanPham = @MaSanPham;
END;

```

- Thêm Thông Tin Khách Hàng

```

CREATE PROCEDURE ThemKhachHang
    @MaKhachHang NVARCHAR(50),
    @TenKhachHang NVARCHAR(50),
    @GioiTinh NVARCHAR(10),
    @DiaChi NVARCHAR(100),
    @SDT NVARCHAR(10)
AS
BEGIN
    INSERT INTO KhachHang (MaKhachHang, TenKhachHang, GioiTinh, DiaChi, SODIENTHOAI)
    VALUES (@MaKhachHang, @TenKhachHang, @GioiTinh, @DiaChi, @SDT);
END;

```

- Sửa Thông Tin Khách Hàng

```

CREATE PROCEDURE SuaKhachHang
    @MaKhachHang NVARCHAR(50),
    @TenKhachHang NVARCHAR(100),
    @GioiTinh NVARCHAR(10),
    @DiaChi NVARCHAR(200),
    @SDT NVARCHAR(20)
AS
BEGIN
    UPDATE KhachHang
    SET TenKhachHang = @TenKhachHang,
        GioiTinh = @GioiTinh,
        DiaChi = @DiaChi,
        SODIENTHOAI = @SDT
    WHERE MaKhachHang = @MaKhachHang;
END;

```

- Xóa Thông Tin Khách Hàng

```

CREATE PROCEDURE SuaKhachHang
    @MaKhachHang NVARCHAR(50),
    @TenKhachHang NVARCHAR(100),
    @GioiTinh NVARCHAR(10),
    @DiaChi NVARCHAR(200),
    @SDT NVARCHAR(20)
AS
BEGIN
    UPDATE KhachHang
    SET TenKhachHang = @TenKhachHang,
        GioiTinh = @GioiTinh,
        DiaChi = @DiaChi,
        SODIENTHOAI = @SDT
    WHERE MaKhachHang = @MaKhachHang;
END;

```

- Xóa Thông Tin Khách Hàng

```

CREATE PROCEDURE XoaKhachHang
    @MaKhachHang NVARCHAR(50)
AS
BEGIN
    DELETE FROM KhachHang
    WHERE MaKhachHang = @MaKhachHang;
END;

```

- Thêm Thông tin nhà cung cấp

```

CREATE PROCEDURE ThemNhaCungCap
    @MaNhaCC INT,
    @TenNhaCC NVARCHAR(255),
    @DiaChi NVARCHAR(255),
    @SDT NVARCHAR(20),
    @Email NVARCHAR(255),
    @NguoiDaiDien NVARCHAR(255)
AS
BEGIN
    INSERT INTO NhaCungCap (MaNhaCC, TenNhaCC, DiaChi, SODIENTHOAI, Email, NguoiDaiDien)
    VALUES (@MaNhaCC, @TenNhaCC, @DiaChi, @SDT, @Email, @NguoiDaiDien);
END;

```

- Sửa thông tin nhà cung cấp

```
CREATE PROCEDURE SuaNhaCungCap
    @MaNhaCC INT,
    @TenNhaCC NVARCHAR(255),
    @DiaChi NVARCHAR(255),
    @SDT NVARCHAR(20),
    @Email NVARCHAR(255),
    @NguoiDaiDien NVARCHAR(255)
AS
BEGIN
    UPDATE NhaCungCap
    SET TenNhaCC = @TenNhaCC,
        DiaChi = @DiaChi,
        SODIENTHOAI = @SDT,
        Email = @Email,
        NguoiDaiDien = @NguoiDaiDien
    WHERE MaNhaCC = @MaNhaCC;
END;
```

- Xóa thông tin nhà cung cấp

```
CREATE PROCEDURE XoaNhaCungCap
    @MaNhaCC NVARCHAR(50)
AS
BEGIN
    DELETE FROM NhaCungCap
    WHERE MaNhaCC = @MaNhaCC;
END;
```

- Thêm thông tin nhân viên

```
CREATE PROCEDURE ThemNhanVien
    @MaNhanVien NVARCHAR(50),
    @TenNhanVien NVARCHAR(50),
    @GioiTinh NVARCHAR(10),
    @NgaySinh DATETIME,
    @DiaChi NVARCHAR(50),
    @SDT NVARCHAR(10),
    @LuongCoBan BIGINT,
    @PhuCap BIGINT,
    @TaiKhoan NVARCHAR(50),
    @MatKhau NVARCHAR(50),
    @MaQuanLy NVARCHAR(50)
AS
BEGIN
    INSERT INTO NhanVien (MaNhanVien, TenNhanVien, GioiTinh, NgaySinh, DAICHI, SDT, LuongCoBan, PhuCap, TaiKhoan, MatKhau, MaQuanLy)
    VALUES (@MaNhanVien, @TenNhanVien, @GioiTinh, @NgaySinh, @DiaChi, @SDT, @LuongCoBan, @PhuCap, @TaiKhoan, @MatKhau, @MaQuanLy);
END;
```

- Sửa thông tin nhân viên


```

CREATE PROCEDURE SuaNhanVien
    @MaNhanVien NVARCHAR(50),
    @TenNhanVien NVARCHAR(50),
    @GioiTinh NVARCHAR(10),
    @NgaySinh DATETIME,
    @DiaChi NVARCHAR(100),
    @SDT NVARCHAR(10),
    @LuongCoBan BIGINT,
    @PhuCap BIGINT,
    @TaiKhoan NVARCHAR(50),
    @MatKhai NVARCHAR(50),
    @MaQuanLy NVARCHAR(50)
AS
BEGIN
    UPDATE NhanVien
    SET TenNhanVien = @TenNhanVien,
        GioiTinh = @GioiTinh,
        NgaySinh = @NgaySinh,
        DAICHI = @DiaChi,
        SDT = @SDT,
        LuongCoBan = @LuongCoBan,
        PhuCap = @PhuCap,
        TaiKhoan = @TaiKhoan,
        MatKhai = @MatKhai,
        MaQuanLy = @MaQuanLy
    WHERE MaNhanVien = @MaNhanVien;
END;

```

- Xóa thông tin nhân viên

```

CREATE PROCEDURE XoaNhanVien
    @MaNhanVien NVARCHAR(50)
AS
BEGIN
    DELETE FROM NhanVien
    WHERE MaNhanVien = @MaNhanVien;
END;

```

- Thêm Phiếu thông kê

```

CREATE PROCEDURE ThemPhieuThongKe
    @SoPhieuThongKe NVARCHAR(50),
    @MaNhanVien NVARCHAR(50),
    @NgayThongKe DATETIME,
    @MaSanPham NVARCHAR(50),
    @SoLuongDaBan INT,
    @TongTien BIGINT
AS
BEGIN
    -- Thêm dữ liệu vào bảng Chi Tiết Phiếu Thông kê
    BEGIN TRY
        INSERT INTO CHITIETPHIEUTHONGKE(SOPHIEUTHONGKE, MASANPHAM, SOLUONGDABAN, TONGTIEN)
        VALUES (@SoPhieuThongKe, @MaSanPham, @SoLuongDaBan, @TongTien);
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;

    -- Commit giao dịch nếu không có lỗi
    COMMIT TRANSACTION;
    -- Bắt đầu một giao dịch
    BEGIN TRANSACTION;

    -- Thêm dữ liệu vào bảng Phiếu Thống Kê
    BEGIN TRY
        INSERT INTO PHIEUTHONGKE(SOPHIEUTHONGKE, MANHANVIEN, NGAYTHONGKE)
        VALUES (@SoPhieuThongKe, @MaNhanVien, @NgayThongKe);
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;
END;

```

- Sửa Phiếu Thống Kê

```

CREATE PROCEDURE SuaPhieuThongKeVaChiTiet
    @SoPhieuThongKe NVARCHAR(50),
    @MaNhanVien NVARCHAR(50),
    @NgayThongKe DATETIME,
    @MaSanPham NVARCHAR(50),
    @SoLuongDaBan INT,
    @TongTien BIGINT
AS
BEGIN
    -- Bắt đầu một giao dịch
    BEGIN TRANSACTION;

    BEGIN TRY
        -- Sửa dữ liệu trong bảng PhieuThongKe
        UPDATE PhieuThongKe
        SET MaNhanVien = @MaNhanVien, NgayThongKe = @NgayThongKe
        WHERE SoPhieuThongKe = @SoPhieuThongKe;

        -- Sửa dữ liệu trong bảng ChiTietPhieuThongKe
        UPDATE ChiTietPhieuThongKe
        SET SoLuongDaBan = @SoLuongDaBan, TongTien = @TongTien
        WHERE SoPhieuThongKe = @SoPhieuThongKe AND MaSanPham = @MaSanPham;

        -- Commit giao dịch nếu không có lỗi
        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;
END;

```

- Xóa Phiếu Thống Kê

```

CREATE PROCEDURE XoaPhieuThongKeVaChiTiet
    @SoPhieuThongKe INT
AS
BEGIN
    -- Bắt đầu một giao dịch
    BEGIN TRANSACTION;

    BEGIN TRY
        -- Xóa dữ liệu trong bảng ChiTietPhieuThongKe
        DELETE FROM ChiTietPhieuThongKe
        WHERE SoPhieuThongKe = @SoPhieuThongKe;

        -- Xóa dữ liệu trong bảng PhieuThongKe
        DELETE FROM PhieuThongKe
        WHERE SoPhieuThongKe = @SoPhieuThongKe;

        -- Commit giao dịch nếu không có lỗi
        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        -- Nếu có lỗi, rollback giao dịch và thoát
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH;
END;

```

Các Báo Cáo

- Báo cáo sản phẩm sắp hết hạn

```

CREATE PROCEDURE SP_HET_HAN
AS
BEGIN
    DECLARE @MASANPHAM NVARCHAR(100),
            @TENSANPHAM NVARCHAR(100),
            @NGAYSANXUAT DATE,
            @HANSUDUNG DATE,
            @Today DATE,
            @ExpiringSoon DATE;

    -- Lấy ngày hiện tại
    SET @Today = GETDATE();
    -- Tính ngày hết hạn sắp tới (trong vòng 30 ngày)
    SET @ExpiringSoon = DATEADD(DAY, 30, @Today);

    -- Khai báo cursor
    DECLARE ProductCursor CURSOR FOR
    SELECT MASANPHAM, TENSANPHAM, NGAYSANXUAT, HANSUDUNG
    FROM SANPHAM
    WHERE HANSUDUNG <= @ExpiringSoon;

    -- Mở cursor
    OPEN ProductCursor;

    -- Lấy dòng đầu tiên từ cursor
    FETCH NEXT FROM ProductCursor INTO @MASANPHAM, @TENSANPHAM, @NGAYSANXUAT, @HANSUDUNG;

    -- Duyệt từng dòng trong cursor
    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Thông báo sản phẩm sắp hết hạn
        PRINT N'Sản phẩm sắp hết hạn: ';
        PRINT N'Mã Sản Phẩm: ' + CAST(@MASANPHAM AS NVARCHAR(50));
        PRINT N'Tên Sản Phẩm: ' + @TENSANPHAM;
        PRINT N'Ngày Sản Xuất: ' + CAST(@NGAYSANXUAT AS NVARCHAR(50));
        PRINT N'Hạn Sử Dụng: ' + CAST(@HANSUDUNG AS NVARCHAR(50));
        PRINT N'-----';

        -- Lấy dòng tiếp theo từ cursor
        FETCH NEXT FROM ProductCursor INTO @MASANPHAM, @TENSANPHAM, @NGAYSANXUAT, @HANSUDUNG;
    END

    -- Đóng cursor
    CLOSE ProductCursor;
    -- Hủy cursor
    DEALLOCATE ProductCursor;
END;

```

○ Kết quả:

```

Sản phẩm sắp hết hạn:
Mã Sản Phẩm: Sp101
Tên Sản Phẩm: Thuốc 1
Ngày Sản Xuất: 2021-01-29
Hạn Sử Dụng: 2024-06-29
-----

```

- Báo cáo sản phẩm tồn kho

```

CREATE PROCEDURE BaoCaoTonKho
AS
BEGIN
    -- Chọn các cột cần thiết và tính toán số lượng tồn kho
    SELECT
        SP.MaSanPham,
        SP.TenSanPham,
        SP.GiaBan,
        SP.Soluong AS SoLuongTonKho, -- Lấy số lượng tồn kho ban đầu từ bảng SanPham
        COALESCE(SUM(CT.SoluongBan), 0) AS SoLuongDaBan, -- Tính tổng số lượng đã bán từ bảng ChiTietDonHang
        SP.Soluong - COALESCE(SUM(CT.SoluongBan), 0) AS SoLuongConLai -- Tính số lượng còn lại
    FROM
        SanPham SP
    LEFT JOIN
        ChiTietDonHang CT ON SP.MaSanPham = CT.MaSanPham -- Gộp nối bảng SanPham với ChiTietDonHang để lấy thông tin số lượng đã bán
    GROUP BY
        SP.MaSanPham, SP.TenSanPham, SP.GiaBan, SP.Soluong -- Nhóm kết quả theo các cột từ bảng SanPham
    ORDER BY
        SP.MaSanPham; -- Sắp xếp kết quả theo mã sản phẩm
END;
GO

```

○ Kết quả:

	MaSanPham	TenSanPham	GiaBan	SoLuongTonKho	SoLuongDaBan	SoLuongConLai
1	SP001	S?n ph?m 1	50000	100	0	100
2	Sp101	Thuốc 1	100000	100	0	100
3	Sp102	Thuốc 2	50000	50	0	50

- Tính Tổng doanh thu

```

CREATE PROCEDURE TinhTongDoanhThu
AS
BEGIN
    DECLARE @TongDoanhThu BIGINT;

    SELECT @TongDoanhThu = SUM(TongTien)
    FROM ChiTietDonHang;

    -- In ra kết quả tổng doanh thu
    PRINT 'Tổng doanh thu từ bảng ChiTietDonHang là: ' + CAST(@TongDoanhThu AS VARCHAR);
END;
GO

```

○ Kết quả:

Tổng doanh thu từ bảng ChiTietDonHang là: 3000000