|  |  |  |  |
| --- | --- | --- | --- |
| Yêu cầu | | Tình trạng | Giải pháp |
| View | | Đã có | k phải “select \* from tableName” thì auto dùng view |
| Procedure | K tham số | Đã có | proc\_CapNhatGiaTriDon\_DonHang  Nút thanh toán (fTrangChu) |
| Có tham số | Đã có | Thêm, Xóa, Sửa table |
| Có tham số output | Đã có | Load thông tin khách hàng (fDonHang) |
| Function  (Khi dùng scalar phải có dbo còn 5 cái kia k cần cx được) | scalar | Đã có | Tổng tiền (TrangChu) |
| inline table-valued | Đã có | Tìm kiếm theo tên loại sản phẩm (fQLLoaiSanPham) |
| multi-statement table valued | Đã có | Tìm kiếm theo tên sản phẩm (fTrangChu) |

**View**

1.

phải thêm 1 cột trạng thái vào nữa

CREATE VIEW v\_DanhSachSanPhamDaChon AS

SELECT sp.MaSP, sp.TenSP, sp.Gia, cthd.SoLuong, cthd.TongTien

FROM ChiTietHoaDon cthd

INNER JOIN SanPham sp ON cthd.MaSP = sp.MaSP

INNER JOIN DonHang dh ON cthd.MaDH = dh.MaDH

WHERE TrangThai = N'Chưa thanh toán'

2.

CREATE VIEW v\_DanhSachSanPham AS

SELECT sp.MaSP, sp.TenSP, sp.Gia, lsp.TenLoaiSP

FROM SanPham sp

INNER JOIN LoaiSanPham lsp ON sp.MaLoaiSP = lsp.MaLoaiSP

3.

CREATE VIEW v\_NguyenLieuDungDePhaCheSanPham AS

SELECT nl.MaNL, nl.TenNL, pc.SoLuong, nl.SoLuongTonKho, nl.DonVi

FROM NguyenLieu nl

INNER JOIN PhaChe pc ON nl.MaNL = pc.MaNL

INNER JOIN SanPham sp ON pc.MaSP = sp.MaSP

4. Danh sách hóa đơn đã thanh toán

CREATE VIEW v\_DanhSachHoaDon AS

SELECT dh.MaDH, dh.NgayMua, kh.SoDienThoai, kh.TenKH, dh.GiaTriDon

FROM DonHang dh

INNER JOIN ChiTietHoaDon cthd ON dh.MaDH = cthd.MaDH

INNER JOIN NhanVien nv ON dh.MaNV = nv.MaNV

INNER JOIN KhachHang kh ON dh.MaKH = kh.MaKH

WHERE dh.TrangThai = N'Đã thanh toán'

5.

CREATE VIEW v\_DanhSachNhanVien AS

SELECT nv.\*, nvttg.LuongCoDinh, nvbtg.SoGio, LuongTheoGio

FROM NhanVien nv

FULL OUTER JOIN NhanVienToanThoiGian nvttg ON nv.MaNV = nvttg.MaNV

FULL OUTER JOIN NhanVienBanThoiGian nvbtg ON nv.MaNV = nvbtg.MaNV

6.

7.

8.

9.

**Function, Procedure**

1. Loại Sản Phẩm

1.1.

CREATE PROCEDURE proc\_Them\_LoaiSanPham

@MaLoaiSP NVARCHAR(50),

@TenLoaiSP NVARCHAR(50)

AS

BEGIN

INSERT INTO LoaiSanPham (MaLoaiSP, TenLoaiSP)

VALUES (@MaLoaiSP, @TenLoaiSP)

END

Exec ThemLoaiSanPham 'newID', 'nhap'

1.2

CREATE PROCEDURE proc\_Xoa\_LoaiSanPham

@MaLoaiSP NVARCHAR(50)

AS

BEGIN

DELETE FROM LoaiSanPham

WHERE MaLoaiSP = @MaLoaiSP

END

1.3.

CREATE PROCEDURE proc\_Sua\_LoaiSanPham

@MaLoaiSP NVARCHAR(50),

@TenLoaiSP NVARCHAR(50)

AS

BEGIN

UPDATE LoaiSanPham

SET TenLoaiSP = @TenLoaiSP

WHERE MaLoaiSP = @MaLoaiSP

END

1.4.

CREATE FUNCTION func\_TimKiemTheoTen\_LoaiSanPham(@TenLoaiSP NVARCHAR(50))

RETURNS TABLE

AS

RETURN

(

SELECT \*

FROM LoaiSanPham

WHERE TenLoaiSP LIKE N'%' + @TenLoaiSP + '%'

)

2. Đăng ký (Chưa làm - Dùng func hoặc proc hoặc trigger)

Thêm nhân viên vào bảng khi đăng kí

CREATE PROCEDURE proc\_ThemNhanVien

@HoTenNV nvarchar(50),

@SoDienThoai nvarchar(50),

@NamSinh int,

@GioiTinh nvarchar(50),

@DiaChi nvarchar(200),

@TenDangNhap nvarchar(50),

@MatKhau nvarchar(50),

@LoaiNhanVien nvarchar(20)

AS

BEGIN

INSERT INTO NhanVien (HoTenNV, SoDienThoai, NamSinh, GioiTinh, DiaChi, TenDangNhap, MatKhau)

VALUES (@HoTenNV, @SoDienThoai, @NamSinh, @GioiTinh, @DiaChi, @TenDangNhap, @MatKhau);

DECLARE @mNV nvarchar(50);

SELECT @mNV = MaNV

FROM NhanVien

WHERE SoDienThoai = @SoDienThoai;

IF @LoaiNhanVien = N'Toàn thời gian'

BEGIN

INSERT INTO NhanVienToanThoiGian (MaNV, LuongCoDinh, VaiTro)

VALUES (@mNV, 1, '');

END

ELSE IF @LoaiNhanVien = N'Bán thời gian'

BEGIN

INSERT INTO NhanVienBanThoiGian (MaNV, LuongTheoGio, SoGio)

VALUES (@mNV, 1, 0);

END

END;

3. Đăng nhập (Chưa làm - Dùng func hoặc proc hoặc trigger)

4. Hóa đơn

CREATE FUNCTION func\_TimKiemTheoSoDienThoai\_DonHang(@SoDienThoai NVARCHAR(50))

RETURNS TABLE

AS

RETURN

(

SELECT MaDH, NgayMua, SoDienThoai, TenKH, GiaTriDon

FROM v\_DanhSachHoaDon

WHERE SoDienThoai LIKE N'%' + @SoDienThoai + '%'

)

5. Kho hàng(Chưa làm)

6. Phân ca (Hùng làm)

7. Quản lý khách hàng (Chưa làm)

8. Quản lý nhân viên (Chưa làm)

8.1. Nút thêm nhân viên.

CREATE PROCEDURE proc\_ThemNhanVien

@HoTenNV nvarchar(50),

@SoDienThoai nvarchar(50),

@NamSinh int,

@GioiTinh nvarchar(50),

@DiaChi nvarchar(200),

@TenDangNhap nvarchar(50),

@MatKhau nvarchar(50),

@LoaiNhanVien nvarchar(20)

AS

BEGIN

INSERT INTO NhanVien (HoTenNV, SoDienThoai, NamSinh, GioiTinh, DiaChi, TenDangNhap, MatKhau)

VALUES (@HoTenNV, @SoDienThoai, @NamSinh, @GioiTinh, @DiaChi, @TenDangNhap, @MatKhau);

DECLARE @mNV nvarchar(50);

SELECT @mNV = MaNV

FROM NhanVien

WHERE SoDienThoai = @SoDienThoai;

IF @LoaiNhanVien = N'Toàn thời gian'

BEGIN

INSERT INTO NhanVienToanThoiGian (MaNV, LuongCoDinh, VaiTro)

VALUES (@mNV, 1, '');

END

ELSE IF @LoaiNhanVien = N'Bán thời gian'

BEGIN

INSERT INTO NhanVienBanThoiGian (MaNV, LuongTheoGio, SoGio)

VALUES (@mNV, 1, 0);

END

END;

8.2. Nút xóa

8.2.1. Xóa nhân viên toàn thời gian

CREATE PROCEDURE proc\_XoaNhanVienToanThoiGian

@MaNV nvarchar(50)

AS

BEGIN

DELETE FROM NhanVienToanThoiGian

WHERE MaNV = @MaNV

END

8.2.2. Xóa nhân viên bán thời gian

CREATE PROCEDURE proc\_XoaNhanVienBanThoiGian

@MaNV nvarchar(50)

AS

BEGIN

DELETE FROM NhanVienBanThoiGian

WHERE MaNV = @MaNV

END

8.2.3. Xóa nhân viên.

CREATE PROCEDURE proc\_XoaNhanVien

@MaNV nvarchar(50)

AS

BEGIN

DELETE FROM NhanVien

WHERE MaNV = @MaNV

END

8.3. Nút sửa.

8.3.1. Sửa nhân viên.

CREATE PROCEDURE proc\_SuaNhanVien

@MaNV NVARCHAR(50),

@HoTenNV nvarchar(50),

@SoDienThoai nvarchar(50),

@NamSinh int,

@GioiTinh nvarchar(50),

@DiaChi nvarchar(200),

@TenDangNhap nvarchar(50),

@MatKhau nvarchar(50)

AS

BEGIN

UPDATE NhanVien

SET HoTenNV = @HoTenNV,

SoDienThoai = @SoDienThoai,

NamSinh = @NamSinh,

GioiTinh = @GioiTinh,

DiaChi = @DiaChi,

TenDangNhap = @TenDangNhap,

MatKhau = @MatKhau

WHERE MaNV = @MaNV

END

8.3.2. Sửa nhân viên toàn thời gian.

CREATE PROCEDURE proc\_SuaNhanVienToanThoiGian

@MaNV NVARCHAR(50),

@LuongCoDinh FLOAT

AS

BEGIN

UPDATE NhanVienToanThoiGian

SET LuongCoDinh = @LuongCoDinh

WHERE MaNV = @MaNV

END

8.3.3. Sửa nhân viên toàn thời gian.

CREATE PROCEDURE proc\_SuaNhanVienBanThoiGian

@MaNV NVARCHAR(50),

@SoGio INT,

@LuongTheoGio FLOAT

AS

BEGIN

UPDATE NhanVienBanThoiGian

SET SoGio = @SoGio,

LuongTheoGio = @LuongTheoGio

WHERE MaNV = @MaNV

END

8.4. Tìm kiếm tên nhân viên

CREATE FUNCTION func\_timKiemTrenViewNhanVien (@HoTenNV NVARCHAR(50))

RETURNS @KetQua TABLE (

MaNV NVARCHAR(50),

HoTenNV NVARCHAR(50),

SoDienThoai NVARCHAR(50),

NamSinh INT,

GioiTinh NVARCHAR(50),

DiaChi NVARCHAR(200),

TenDangNhap NVARCHAR(50),

MatKhau NVARCHAR(50),

LuongCoDinh FLOAT,

SoGio INT,

LuongTheoGio FLOAT

)

AS

BEGIN

INSERT INTO @KetQua

SELECT MaNV, HoTenNV, SoDienThoai, NamSinh, GioiTinh, DiaChi, TenDangNhap, MatKhau, LuongCoDinh, SoGio, LuongTheoGio

FROM v\_DanhSachNhanVien

WHERE HoTenNV LIKE '%' + @HoTenNV + '%';

RETURN;

END;

9. Quản lý Sản phẩm

9.1

CREATE PROCEDURE proc\_Them\_SanPham

@MaSP NVARCHAR(50),

@TenSP NVARCHAR(50),

@Gia FLOAT,

@AnhSP NVARCHAR(50),

@MaLoaiSP NVARCHAR(50)

AS

BEGIN

INSERT INTO SanPham(MaSP, TenSP, Gia, AnhSP, MaLoaiSP)

VALUES (@MaSP, @TenSP, @Gia, @AnhSP, @MaLoaiSP)

END

9.2

CREATE PROCEDURE proc\_Xoa\_SanPham

@MaSP NVARCHAR(50)

AS

BEGIN

DELETE FROM SanPham

WHERE MaSP = @MaSP

END

9.3

CREATE PROCEDURE proc\_Sua\_SanPham

@MaSP NVARCHAR(50),

@TenSP NVARCHAR(50),

@Gia FLOAT,

@AnhSP NVARCHAR(50),

@MaLoaiSP NVARCHAR(50)

AS

BEGIN

UPDATE SanPham

SET TenSP = @TenSP,

Gia = @Gia,

AnhSP = @AnhSP,

MaLoaiSP = @MaLoaiSP

WHERE MaSP = @MaSP

END

9.4

CREATE FUNCTION func\_TimKiemTheoTen\_SanPham(@TenSP NVARCHAR(50))

RETURNS @KetQua TABLE

(

MaSP NVARCHAR(50),

TenSP NVARCHAR(50),

Gia FLOAT,

TenLoaiSP NVARCHAR(50),

AnhSP NVARCHAR(200)

)

AS

BEGIN

INSERT INTO @KetQua

SELECT MaSP, TenSP, Gia, TenLoaiSP, AnhSP

FROM v\_DanhSachSanPham

WHERE TenSP LIKE N'%' + @TenSP + '%'

RETURN

END

9.5

CREATE PROCEDURE proc\_Them\_PhaChe

@MaSP NVARCHAR(50),

@MaNL NVARCHAR(50),

@SoLuong INT

AS

BEGIN

INSERT INTO PhaChe(MaSP, MaNL, SoLuong)

VALUES (@MaSP, @MaNL, @SoLuong)

END

9.6

CREATE PROCEDURE proc\_Xoa\_PhaChe

@MaSP NVARCHAR(50),

@MaNL NVARCHAR(50)

AS

BEGIN

DELETE FROM PhaChe

WHERE MaSP = @MaSP AND MaNL = @MaNL

END

9.7

CREATE FUNCTION func\_LoadDanhSachNguyenLieuTheoMaSP(@MaSP NVARCHAR(50))

RETURNS TABLE

AS

RETURN

(

SELECT nl.MaNL, nl.TenNL, pc.SoLuong, nl.SoLuongTonKho, nl.DonVi

FROM NguyenLieu nl

INNER JOIN PhaChe pc ON nl.MaNL = pc.MaNL

INNER JOIN SanPham sp ON pc.MaSP = sp.MaSP

WHERE sp.MaSP = @MaSP

)

10. Quên mật khẩu (Chưa làm - Dùng func hoặc proc hoặc trigger)

11. Trang Chủ

11.1.

CREATE FUNCTION func\_TimKiemTheoTen\_SanPham(@TenSP NVARCHAR(50))

RETURNS @KetQua TABLE

(

MaSP NVARCHAR(50),

TenSP NVARCHAR(50),

Gia FLOAT,

AnhSP NVARCHAR(50)

)

AS

BEGIN

INSERT INTO @KetQua

SELECT MaSP, TenSP, Gia, AnhSP

FROM SanPham

WHERE TenSP LIKE N'%' + @TenSP + '%'

RETURN

END

11.2. Nút new order

11.2.1. Tạo mới khách hàng (tạm thời)

CREATE PROCEDURE proc\_Them\_KhachHang

@SoDienThoai NVARCHAR(50)

AS

BEGIN

DECLARE @maxMaKH NVARCHAR(50);

DECLARE @newMaKH NVARCHAR(50); -- @newMaKH = @maxMaKH + 1

DECLARE @numPart INT;

-- Tìm giá trị MaKH lớn nhất hiện có

SELECT @maxMaKH = MAX(MaKH)

FROM KhachHang

WHERE MaKH LIKE 'KH%';

-- Lấy phần số từ MaKH (bỏ phần 'KH' phía trước) và convert sang kiểu INT

IF @maxMaKH IS NOT NULL

BEGIN

SET @numPart = CAST(SUBSTRING(@maxMaKH, 3, LEN(@maxMaKH) - 2) AS INT) + 1;

END

ELSE

BEGIN

-- Nếu chưa có MaKH nào, bắt đầu từ 1

SET @numPart = 1;

END

-- Tạo giá trị mới cho MaKH, với định dạng KHxx (2 số)

SET @newMaKH = 'KH' + RIGHT('00' + CAST(@numPart AS NVARCHAR), 2);

-- Thực hiện chèn bản ghi với maDH mới

INSERT INTO KhachHang(MaKH, TenKH, SoDienThoai, SoDiemTichLuy)

VALUES(@newMaKH, N'Chưa nhập tên', @SoDienThoai, 0)

END

11.2.2. Tạo mới đơn hàng (tạm thời)

CREATE PROCEDURE proc\_Them\_DonHang

@SoDienThoai NVARCHAR(50),

@MaNV NVARCHAR(50)

AS

BEGIN

DECLARE @MaKH NVARCHAR(50)

SELECT @MaKH = MaKH

FROM KhachHang

WHERE SoDienThoai = @SoDienThoai

INSERT INTO DonHang(MaKH, MaNV, NgayMua, GiaTriDon, TrangThai)

VALUES (@MaKH, @MaNV, GETDATE(), 1, N'Chưa thanh toán')

END

11.3. Nút xóa tất cả (được thực thi “mặc định” khi fTrangChu được load để Clear dữ liệu)

CREATE PROCEDURE proc\_Xoa\_KhachHangChuaNhapTen\_DonHangChuaThanhToan

AS

BEGIN

DECLARE @MaDH NVARCHAR(50)

SELECT @MaDH = MaDH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

DELETE FROM ChiTietHoaDon

WHERE MaDH = @MaDH

DELETE FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

DELETE FROM KhachHang

WHERE TenKH = N'Chưa nhập tên'

END

11.4. Nút đặt

CREATE PROCEDURE proc\_Them\_ChiTietDonHang

@MaSP NVARCHAR(50)

AS

BEGIN

DECLARE @MaDH NVARCHAR(50)

SELECT @MaDH = MaDH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

INSERT INTO ChiTietHoaDon(MaDH, MaSP, SoLuong, TongTien)

VALUES (@MaDH, @MaSP, 1, (SELECT Gia\*1 FROM SanPham WHERE MaSP = @maSP))

END

11.5. Nút thanh toán

CREATE PROCEDURE proc\_CapNhatGiaTriDon\_DonHang

AS

BEGIN

DECLARE @MaDH NVARCHAR(50)

SELECT @MaDH = MaDH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

UPDATE DonHang

SET GiaTriDon = (SELECT SUM(TongTien) FROM ChiTietHoaDon WHERE MaDH = @MaDH)

WHERE TrangThai = N'Chưa thanh toán'

END

11.6



CREATE FUNCTION func\_TongTien\_ChiTietHoaDon

(

)

RETURNS FLOAT

AS

BEGIN

DECLARE @MaDH NVARCHAR(50)

DECLARE @Tong FLOAT = 0

SELECT @MaDH = MaDH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

-- Tính tổng tiền

SELECT @Tong = SUM(TongTien)

FROM ChiTietHoaDon

WHERE MaDH = @MaDH

RETURN @Tong

END;

12. Đơn hàng

12.1. button xác nhận thanh toán

CREATE PROCEDURE proc\_CapNhatSoDiemTichLuy\_CapTrangThaiDonHang

AS

BEGIN

DECLARE @GiaTriDon FLOAT

DECLARE @MaKH NVARCHAR(50)

SELECT @GiaTriDon = GiaTriDon, @MaKH = MaKH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

UPDATE KhachHang

SET SoDiemTichLuy = SoDienThoai + @GiaTriDon \* 0.01

WHERE MaKH = @MaKH

UPDATE DonHang

SET TrangThai = N'Đã thanh toán'

WHERE TrangThai = N'Chưa thanh toán'

END

12.2. Load thông tin khách hàng

CREATE PROCEDURE proc\_LoadThongTin\_KhachHang

@SoDienThoai NVARCHAR(50),

@Ten NVARCHAR(50) OUTPUT,

@SoDiemTichLuy FLOAT OUTPUT

AS

BEGIN

SELECT @Ten = kh.TenKH, @SoDiemTichLuy = kh.SoDiemTichLuy

FROM KhachHang kh

WHERE SoDienThoai = @SoDienThoai

END

12.3. Sửa khách hàng

CREATE PROCEDURE proc\_Sua\_KhachHang

@TenKH NVARCHAR(50),

@SoDienThoaiCu NVARCHAR(50),

@SoDienThoaiMoi NVARCHAR(50)

AS

BEGIN

UPDATE KhachHang

SET TenKH = @TenKH, SoDienThoai = @SoDienThoaiMoi

WHERE SoDienThoai = @SoDienThoaiCu;

END

12.4 tính thành tiền trong fDonHang

CREATE PROCEDURE proc\_TinhThanhTienTrongFormDonHang\_DonHang

@SoDiemTichLuy FLOAT

AS

BEGIN

UPDATE DonHang

SET GiaTriDon = GiaTriDon - @SoDiemTichLuy

WHERE TrangThai = N'Chưa thanh toán';

END

12.5

CREATE FUNCTION func\_TimMaDHChuaThanhToan\_DonHang

(

)

RETURNS NVARCHAR(50)

AS

BEGIN

DECLARE @MaDH NVARCHAR(50)

SELECT @MaDH = MaDH

FROM DonHang

WHERE TrangThai = N'Chưa thanh toán'

RETURN @MaDH

END

12.6

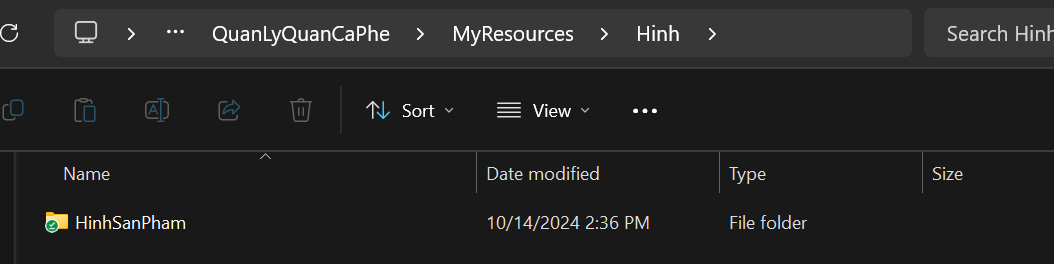
12.7

13. Thống kê (Chưa design, cũng chưa code j hết -> Làm sau cùng)

Yêu cầu

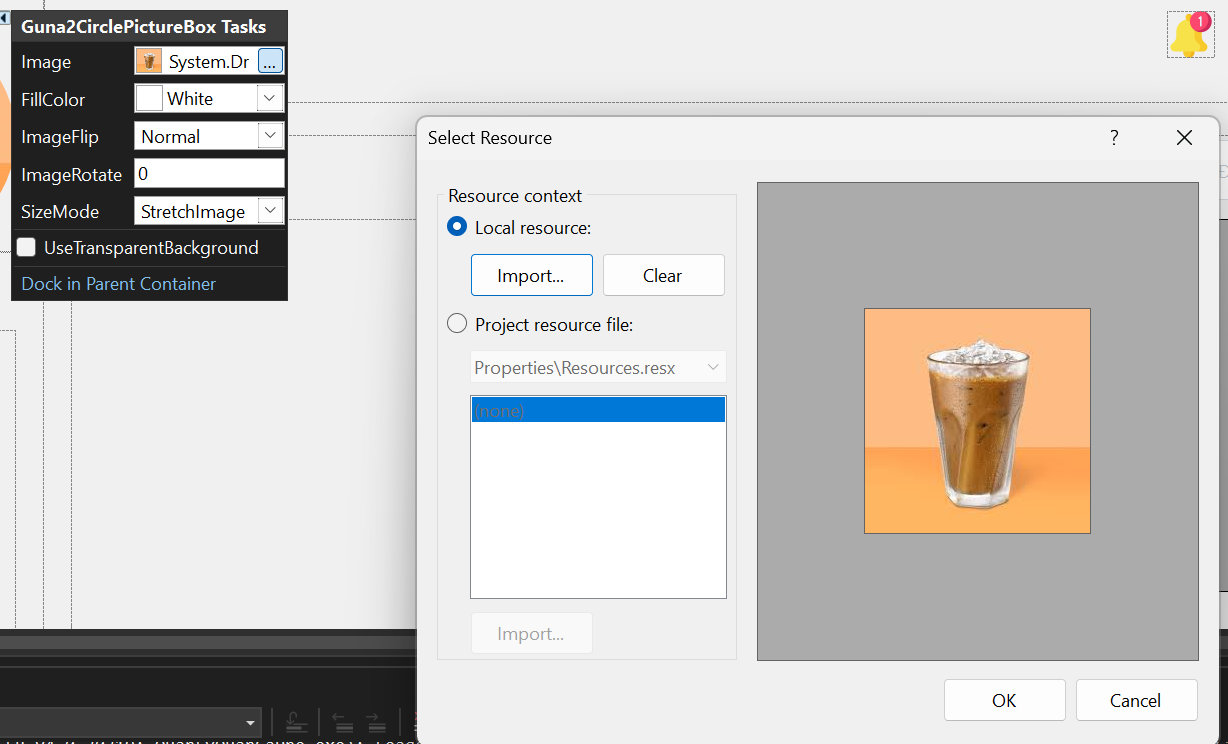
1. Lưu ý khi lưu ảnh sản phẩm bằng PictureBox

- Lưu ảnh trong MyResources/Hinh/HinhSanPham (Sao này thêm xóa sửa dễ dàng, k bị lỗi)



1.1. Giao diện (Dùng tạm lúc design thôi)

- Load ảnh: khi dùng giao diện thì nhớ chọn Local resource



1.2. Code C# (dùng nhiều)

- Load ảnh: bỏ tên file ảnh vào là xong, (lưu tên file dưới database rồi gọi lên)



2. Class DBConnection và DAO t dùng static hết, khỏi cần tạo đối tượng

3. trong Class DBConnection t mới nghĩ ra cái gọi proc có tham số và k có thamm số, function inline vs multi-statement,

Proc có tham số OUTPUT và function scalar t chưa code (ai làm xong nhớ share liền lên nhóm để mn test)

4. Xem class QLLoaiSanPhamDAO (t có làm mẫu cách dùng rồi á)

**Phân công**

|  |  |
| --- | --- |
| Hạnh | Bổ sung database cho hoàn chỉnh, nhớ kiếm ảnh cho từng sản phẩm rồi lưu tên file dô database, gửi ảnh 1 folder riêng  Code cái đăng nhập, quên mật khẩu, kho hàng |
| Hùng | Phân ca, QLKhachHang |
| Khoa | Đăng ký, QL Nhân viên |
| Trí | QLLoaiSP, HoaDon, QLSanPham, TrangChu, DonHang |

**TRIGGER**

1. fTrangChu

- Tự động kích hoạt trigger dưới đây để lấy maDH

CREATE TRIGGER trg\_TuDongTaoMaDH\_DonHang

ON DonHang

INSTEAD OF INSERT

AS

BEGIN

DECLARE @maxMaDH NVARCHAR(50);

DECLARE @newMaDH NVARCHAR(50);

DECLARE @numPart INT;

-- Tìm giá trị maDH lớn nhất hiện có

SELECT @maxMaDH = MAX(maDH)

FROM DonHang

WHERE maDH LIKE 'DH%';

-- Lấy phần số từ maDH (bỏ phần 'DH' phía trước) và convert sang kiểu INT

IF @maxMaDH IS NOT NULL

BEGIN

SET @numPart = CAST(SUBSTRING(@maxMaDH, 3, LEN(@maxMaDH) - 2) AS INT) + 1;

END

ELSE

BEGIN

-- Nếu chưa có maDH nào, bắt đầu từ 1

SET @numPart = 1;

END

-- Tạo giá trị mới cho maDH, với định dạng DHxx (2 số)

SET @newMaDH = 'DH' + RIGHT('00' + CAST(@numPart AS NVARCHAR), 2);

-- Thực hiện chèn bản ghi với maDH mới

INSERT INTO DonHang (MaDH, MaKH, MaNV, NgayMua, GiaTriDon, TrangThai)

SELECT @newMaDH, MaKH, MaNV, NgayMua, GiaTriDon, TrangThai

FROM inserted;

END;

- Khi click ‘Đặt’ món

insert into ChiTietHoaDon(MaDH,MaSP,SoLuong)

values ('DH06','SP01',1)

thì có 2 TH:

+ Đã có trong chi tiết hóa đơn -> Tăng SoLuong lên 1

+ Chưa có thì tạo mới

CREATE TRIGGER trg\_Them\_ChiTietHoaDon

ON ChiTietHoaDon

INSTEAD OF INSERT

AS

BEGIN

DECLARE @maDH NVARCHAR(50);

DECLARE @maSP NVARCHAR(50);

DECLARE @soLuong INT;

-- Lấy giá trị từ bảng inserted

SELECT @maDH = MaDH, @maSP = MaSP

FROM inserted;

-- Kiểm tra xem bản ghi đã tồn tại chưa

IF EXISTS (SELECT 1 FROM ChiTietHoaDon WHERE MaDH = @maDH AND MaSP = @maSP)

BEGIN

-- Nếu đã tồn tại, tăng số lượng lên 1

UPDATE ChiTietHoaDon

SET SoLuong = SoLuong + 1

WHERE MaDH = @maDH AND MaSP = @maSP;

-- Cập nhật tổng tiền

SELECT @soLuong = SoLuong

FROM ChiTietHoaDon

WHERE MaDH = @maDH AND MaSP = @maSP;

UPDATE ChiTietHoaDon

SET TongTien = (SELECT Gia\*@soLuong FROM SanPham WHERE MaSP = @maSP)

WHERE MaDH = @maDH AND MaSP = @maSP;

END

ELSE

BEGIN

-- Nếu chưa tồn tại, chèn bản ghi mới

INSERT INTO ChiTietHoaDon (MaDH, MaSP, SoLuong, TongTien)

VALUES (@maDH, @maSP, 1, (SELECT Gia\*1 FROM SanPham WHERE MaSP = @maSP));

END

END;

2. fDonHang

- Tự động kích hoạt trigger dưới đây để lấy maLoaiSP

CREATE TRIGGER trg\_TuDongTaoMaLoaiSanPham\_LoaiSanPham

ON LoaiSanPham

INSTEAD OF INSERT

AS

BEGIN

DECLARE @maxMaLoaiSP NVARCHAR(50);

DECLARE @newMaLoaiSP NVARCHAR(50);

DECLARE @numPart INT;

-- Tìm giá trị maLoaiSP lớn nhất hiện có

SELECT @maxMaLoaiSP = MAX(MaLoaiSP)

FROM LoaiSanPham

WHERE MaLoaiSP LIKE 'LSP%';

-- Lấy phần số từ maLoaiSP (bỏ phần 'LSP' phía trước) và convert sang kiểu INT

IF @maxMaLoaiSP IS NOT NULL

BEGIN

SET @numPart = CAST(SUBSTRING(@maxMaLoaiSP, 4, LEN(@maxMaLoaiSP) - 2) AS INT)+1;

END

ELSE

BEGIN

-- Nếu chưa có maSP nào, bắt đầu từ 1

SET @numPart = 1;

END

-- Tạo giá trị mới cho maLoaiSP, với định dạng LSPxx (2 số)

SET @newMaLoaiSP = 'LSP' + RIGHT('00' + CAST(@numPart AS NVARCHAR), 2);

-- Thực hiện chèn bản ghi với maLoaiSP mới

INSERT INTO LoaiSanPham(MaLoaiSP, TenLoaiSP)

SELECT @newMaLoaiSP, TenLoaiSP

FROM inserted;

END;

3. fQLSanPham

- Tự động kích hoạt trigger dưới đây để lấy maSP

CREATE TRIGGER trg\_TuDongTaoMaSP\_SanPham

ON SanPham

INSTEAD OF INSERT

AS

BEGIN

DECLARE @maxMaSP NVARCHAR(50);

DECLARE @newMaSP NVARCHAR(50);

DECLARE @numPart INT;

-- Tìm giá trị maSP lớn nhất hiện có

SELECT @maxMaSP = MAX(MaSP)

FROM SanPham

WHERE MaSP LIKE 'SP%';

-- Lấy phần số từ maSP (bỏ phần 'SP' phía trước) và convert sang kiểu INT

IF @maxMaSP IS NOT NULL

BEGIN

SET @numPart = CAST(SUBSTRING(@maxMaSP, 3, LEN(@maxMaSP) - 2) AS INT)+1;

END

ELSE

BEGIN

-- Nếu chưa có maSP nào, bắt đầu từ 1

SET @numPart = 1;

END

-- Tạo giá trị mới cho maSP, với định dạng SPxx (2 số)

SET @newMaSP = 'SP' + RIGHT('00' + CAST(@numPart AS NVARCHAR), 2);

-- Thực hiện chèn bản ghi với maSP mới

INSERT INTO SanPham(MaSP, TenSP, Gia, AnhSP, MaLoaiSP)

SELECT @newMaSP, TenSP, Gia, AnhSP, MaLoaiSP

FROM inserted;

END;

4. Đăng kí.

CREATE TRIGGER trg\_TuDongTaoMaNV\_NhanVien

ON NhanVien

INSTEAD OF INSERT

AS

BEGIN

DECLARE @maxMaNV NVARCHAR(50);

DECLARE @newMaNV NVARCHAR(50);

DECLARE @numPart INT;

SELECT @maxMaNV = MAX(MaNV)

FROM NhanVien

WHERE MaNV LIKE 'NV%';

IF @maxMaNV IS NOT NULL

BEGIN

SET @numPart = CAST(SUBSTRING(@maxMaNV, 3, LEN(@maxMaNV) - 2) AS INT) + 1;

END

ELSE

BEGIN

SET @numPart = 1;

END

SET @newMaNV = 'NV' + RIGHT('00' + CAST(@numPart AS NVARCHAR), 2);

INSERT INTO NhanVien (MaNV, HoTenNV, SoDienThoai, NamSinh, GioiTinh, DiaChi,TenDangNhap,MatKhau)

SELECT @newMaNV, HoTenNV, SoDienThoai, NamSinh, GioiTinh, DiaChi, TenDangNhap, MatKhau

FROM inserted;

END;

Thay đổi

- Database thêm DonVi (NguyenLieu), thêm TrangThai (DonHang)

- thêm cái lblIdSP

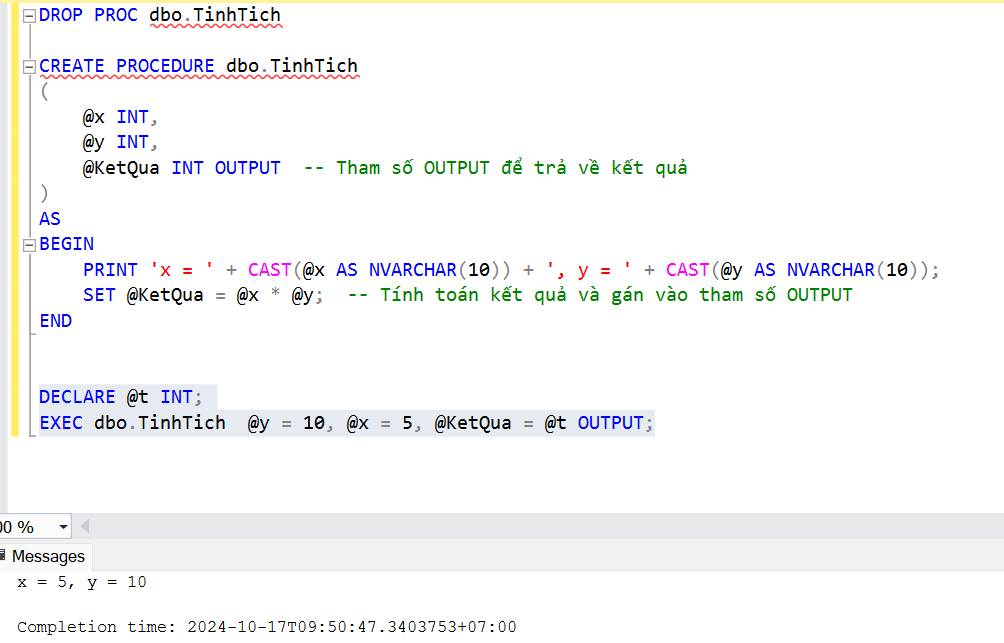
- Gán cố định

MaDH = “DH06”

MaKH = “KH01”

MaNV = “NV01”

MaSP = “SP01”, “SP02”

- 

Thêm cái for này sao SqlCommand (2 cái hàm gọi func của DBConnection) - sửa ở trên value thành key

