



TRƯỜNG ĐẠI HỌC NGOẠI NGỮ - TIN HỌC THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



BÀI BÁO CÁO KẾT THÚC HỌC KỲ II

NĂM HỌC 2022-2023

MÔN HỌC: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

QUẢN LÝ CỬA HÀNG BÁN GIÀY (OFFLINE)

GV hướng dẫn: Ths. Trần Thị Thanh Thảo

Sinh viên thực hiện:

- | | | |
|---------------------|---|------------|
| 1. Trần Lê Duy | – | 20DH110518 |
| 2. Huỳnh Tấn Hiệp | – | 20DH111987 |
| 3. Nguyễn Hoàng Kha | – | 20DH112017 |

Thành phố Hồ Chí Minh, ngày ... tháng ... năm ...



QUẢN LÝ CỬA HÀNG BÁN GIÀY

LỜI CẢM ƠN

Kính gửi quý thầy cô.

Chúng em xin được gửi lời cảm ơn chân thành đến thầy cô hướng dẫn đã hỗ trợ chúng em trong quá trình thực hiện đồ án báo cáo tổng kết này.

Sự hỗ trợ, sự chỉ dẫn cùng với sự động viên và khuyến khích của các thầy cô đã giúp chúng em hoàn thành tốt hơn báo cáo này. Chúng em đã được học hỏi nhiều kinh nghiệm thiết kế và tổ chức cơ sở dữ liệu, cũng như kỹ năng nghiên cứu sản phẩm và xây dựng báo cáo.

Đồ án báo cáo tổng kết là một bước quan trọng trong quá trình học tập và phát triển cá nhân của chúng em. Chúng em luôn trân trọng và biết ơn những kiến thức và trải nghiệm mà các thầy cô đã truyền đạt. Qua đó, chúng em tự tin hơn khi tiếp cận với thực tế kinh tế và công nghiệp trong tương lai.

Một lần nữa, chúng em xin chân thành cảm ơn sự hướng dẫn và giúp đỡ của các thầy cô trong quá trình thực hiện đồ án này. Chúng em hy vọng sẽ tiếp tục nhận được sự hỗ trợ và giúp đỡ từ các thầy cô, phát triển được bản thân và đạt được những mục tiêu sau này.

Chúng em xin chân thành cảm ơn quý thầy cô.

MỤC LỤC

GIỚI THIỆU ĐỀ TÀI	1
CHƯƠNG 1: KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU.....	2
1.1. Khảo sát hiện trạng.....	2
1.2. Xác định yêu cầu	3
1.2.1 Sơ đồ tổ chức	3
1.2.2 Ý nghĩa các bộ phận	3
1.2.3 Yêu cầu người dùng.....	4
1.2.4 Quy định	5
1.3. Công nghệ sử dụng.....	5
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	6
2.1. Thủ Tục.....	6
2.2. Con Trỏ	6
2.3. Function	6
2.4. Package	6
2.5. Trigger.....	7
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ DỮ LIỆU.....	8
3.1. Phân tích dữ liệu	8
3.2. Chi tiết thực thể.....	8
3.3. Thiết kế dữ liệu.....	12
4.1. Danh sách khung nhìn (View).....	13
4.2. Danh sách thủ tục (Store Procedured).....	13
4.3. Danh sách hàm (Function)	16
4.4. Danh sách Trigger.....	16
CHƯƠNG 5: TRIỂN KHAI THỰC NGHIỆM.....	17
5.1. Lệnh tạo bảng.....	17
5.2. Lệnh tạo ràng buộc	22
5.3. Lệnh tạo khung nhìn (View)	24
5.4. Lệnh tạo thủ tục (Store Procedured)	26
5.5. Lệnh tạo hàm (Function).....	49
5.6. Lệnh tạo Trigger	52
5.7. Backup dữ liệu.....	55
5.8. Phân quyền sử dụng CSDL.....	55

5.8.1. Tạo tài khoản.....	55
5.8.2. Tạo role	55
5.8.3. Tạo người dùng	55
5.8.4. Gán quyền.....	56
PHỤ LỤC	60
TÀI LIỆU THAM KHẢO.....	61

GIỚI THIỆU ĐỀ TÀI

Ngày nay, ứng dụng công nghệ thông tin và việc tin học hóa được xem là một trong những yếu tố mang tính quyết định trong hoạt động của các chính phủ, tổ chức, cũng như của các công ty, nó đóng vai trò hết sức quan trọng, có thể tạo ra những bước đột phá mạnh mẽ.

Nhờ có Internet mà chúng ta có thể thực hiện được nhiều công việc với tốc độ nhanh hơn và chi phí thấp hơn so với các cách thức truyền thống. Chính điều này đã thúc đẩy sự khai sinh và phát triển của thương mại điện tử và chính phủ điện tử trên khắp thế giới, làm biến đổi đáng kể bộ mặt văn hóa, nâng cao chất lượng cuộc sống con người.

Trong hoạt động sản xuất kinh doanh, giờ đây nền thương mại điện tử khẳng định được vai trò xúc tiến và thúc đẩy sự phát triển của doanh nghiệp. Đối với những công ty thường xuyên sử dụng việc quảng bá thương hiệu và giới thiệu đến khách hàng các chương trình hấp dẫn do công ty tổ chức trong từng thời điểm một cách hiệu quả và kịp thời là yếu tố mang tính quyết định cho sự tồn tại và phát triển của công ty. Cho nên sẽ rất thiếu sót nếu công ty của bạn chưa xây dựng được cho mình một website để giới thiệu rộng rãi các sản phẩm của mình đang bán. Bên cạnh đó, vấn đề được đặt ra là làm sao để có thể quản lý và điều hành website một cách dễ dàng và hiệu quả.

Với mục tiêu xây dựng một hệ thống bán hàng qua mạng uy tín, đơn giản, thân thiện, cũng như đem lại cho khách hàng những lựa chọn tối ưu để ngay cả những người không rành về tin học hay công nghệ cũng có thể tham gia mua hàng trên mạng chỉ với một số thao tác đơn giản và có định hướng buôn bán gãy gọn thời trang.

Mặc dù rất cố gắng để hoàn thành công việc, song thời gian có hạn là kiến thức chưa nhiều nên việc phân tích thiết kế còn có nhiều thiếu sót cần được bổ sung. Vì vậy, nhóm thực hiện đề tài mong nhận được ý kiến đóng góp của thầy cô và bạn bè để đề tài ngày càng hoàn thiện hơn. Cuối cùng thay cho lời kết chúng em xin chân thành cảm ơn giảng viên hướng dẫn đã tận tình giúp đỡ, hướng dẫn, sửa chữa trong suốt quá trình khảo sát thiết kế đề tài này.

CHƯƠNG 1: KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

1.1. Khảo sát hiện trạng

Trong thời đại hiện nay, việc sử dụng công nghệ thông tin vào quản lý cơ sở dữ liệu là hết sức cần thiết. Việc quản lý thông qua các ứng dụng sẽ giúp con người làm việc hiệu quả, nhanh chóng và tăng độ chính xác hơn so với việc làm thông qua giấy bút cũng như làm thủ công.

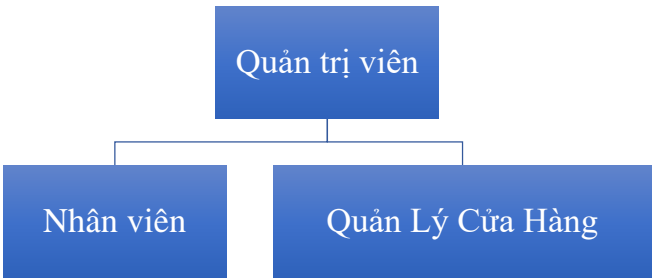
Với việc có thể kiểm soát lượng hàng hóa, cũng như các đơn hàng, quản lý nhân viên với số lượng vô cùng lớn hiện nay tại các cửa hàng, doanh nghiệp sẽ vô cùng khó khăn và dễ dàng gặp những lỗi về sự chính xác dữ liệu thông tin cũng như tìm kiếm dữ liệu. Cho nên đây cũng là lý do hệ thống quản lý được ra đời nhằm tối ưu trong việc quản lý danh sách cũng như theo dõi doanh thu. Ngoài ra, khi có hệ thống quản lý thì số lượng nhân lực sẽ được giảm đi đáng kể, không cần phải bỏ ra nhiều người và hàng giờ liền chỉ để thống kê số liệu,... Thay vào đó chỉ cần với cú nhân chuột thì đã có được những thông tin chúng ta cần như “Danh sách sản phẩm, Danh sách nhân viên, Các đơn hàng đã và đang xử lý, Tuy nhiên, vẫn có một số khó khăn nhất định về vấn đề lưu trữ như dễ bị thất lạc thông tin dẫn đến doanh thu trên số liệu bị hụt,...

Mặc dù vậy, nhưng khi có sự góp sức của công nghệ thông tin, mọi thứ trở nên đơn giản và tối ưu hơn bao giờ hết. Ví dụ: Thay vì phải ghi giấy đơn hàng của khách hàng, điều này dẫn đến tốn rất nhiều thời gian, giấy và mực và cũng như thời gian, nay nhân viên chỉ cần nhập vào form đã được tạo sẵn và vài nút bấm đã có thể hoàn thành việc tạo đơn hàng. Hệ thống sẽ tự động tính cả giá tiền, lưu toàn bộ chi tiết của đơn hàng vào hệ thống.

Việc ứng dụng công nghệ thông tin không những tối ưu thời gian mà còn tối ưu hiệu suất hoạt động của cửa hàng, giúp quản lý doanh thu chính xác hơn. Qua đó cửa hàng có thể phát triển nhanh chóng và tồn tại lâu dài trong môi trường cạnh tranh này.

1.2. Xác định yêu cầu

1.2.1 Sơ đồ tổ chức



1.2.2 Ý nghĩa các bộ phận

STT	Tên bộ phận	Mô tả
1	Quản trị viên	Quản lý toàn bộ hệ thống
2	Quản lý cửa hàng	<ul style="list-style-type: none">• Quản lý Sản phẩm• Xem số lượng sản phẩm• Quản lý Nhân viên• Quản lý Khách hàng• Xem và thêm Đơn hàng• Thay đổi trạng thái Đơn hàng• Xem chi tiết đơn hàng• Cập nhật lại số lượng sản phẩm của đơn hàng
3	Nhân viên	<ul style="list-style-type: none">• Thêm, Xóa, Cập nhật Sản phẩm• Xem số lượng sản phẩm• Xem danh sách Khách hàng• Thêm mới Khách hàng• Xem, Thêm Đơn hàng• Thay đổi Trạng thái Đơn hàng

1.2.3 Yêu cầu người dùng

Danh sách yêu cầu			
Bộ phận: Quản trị viên			
STT	Nghịệp vụ	Mô tả tóm tắt	Ghi chú
1	Quản lý nhân viên	Xem, thêm, xóa, sửa, cấp quyền	
2	Quản lý tài khoản	Cấp tài khoản cho nhân viên	
3	Quản lý khách hàng	Xem, thêm, xóa, sửa thông tin khách	
4	Quản lý sản phẩm	Xem, thêm, xóa, sửa các sản phẩm và phân loại sản phẩm	
5	Quản lý nhà cung cấp	Xem, thêm, xóa, sửa các nhà cung cấp mà cửa hàng có liên kết	
6	Quản lý phòng ban	Xem, thêm, xóa, sửa Phòng ban	
7	Quản lý chức vụ	Xem, thêm, xóa, sửa Chức vụ nhân viên	
8	Quản lý đơn hàng	Xem, thêm, xóa, sửa các đơn hàng	
9	Quản lý Biến động giá	Xem, thêm, xóa, sửa giá sản phẩm	
10	Xem doanh thu	Thống kê doanh thu đã bán	
11	Quản lý Phiếu Nhập	Xem, thêm, xóa, sửa Các Phiếu nhập	

Danh sách yêu cầu			
Bộ phận: Quản lý cửa hàng			
STT	Nghịệp vụ	Mô tả tóm tắt	Ghi chú
1	Quản lý Sản phẩm	Xem, thêm, xóa, sửa Sản phẩm	
2	Quản lý Nhân viên	Xem, thêm, xóa, sửa Nhân viên	
3	Quản lý Nhà cung cấp	Xem, thêm, cập nhật Nhà cung cấp	
4	Quản lý khách hàng	Xem, thêm, xóa, sửa Khách hàng	
5	Quản lý Biến động giá	Xem, thêm, xóa, sửa Giá bán sản phẩm	
6	Quản lý Đơn hàng	Xem, thêm Đơn hàng, Cập nhật trạng thái Đơn hàng	
	Quản lý Chi tiết ĐH	Thay đổi số lượng sản phẩm của Đơn hàng	
7	Quản lý Phiếu nhập	Xem, thêm, xóa, sửa Phiếu nhập	
	Chi tiết Phiếu	Cập nhật Số lượng Nhập, Đơn giá nhập cũng như tình trạng của Phiếu	

Danh sách yêu cầu			
Bộ phận: Nhân viên			
STT	Nhiệm vụ	Mô tả tóm tắt	Ghi chú
1	Quản lý Sản phẩm	Xem, thêm, sửa Sản phẩm	
		Biết được số lượng Sản phẩm	
2	Thêm Khách hàng	Nhập thông tin Khách hàng vào hệ thống	
	Xem danh sách Khách	Xem được Danh sách Khách hàng đã từng mua ở Cửa hàng	
3	Tạo đơn hàng	Tạo đơn hàng cho Khách hàng	
	Thanh toán	Thay đổi trạng thái đơn hàng sau khi Khách đã thanh toán	
	Xem doanh thu	Thấy được danh sách các Đơn hàng đã thực hiện	

1.2.4 Quy định

Tên quy định	Nội dung
QĐ01	Doanh thu = Tổng tiền các Đơn hàng đã thanh toán
QĐ02	Đơn hàng chưa thanh toán mới có thể Hủy, nhưng sau đó số lượng sản phẩm sẽ được trả lại trạng thái ban đầu
QĐ03	Khi số lượng của sản phẩm ở tất cả kích cỡ hết sẽ cập nhật lại trạng thái của Sản phẩm là Hết hàng

1.3. Công nghệ sử dụng

Ứng dụng được viết dưới dạng Webform, sử dụng công nghệ .NET Framework, chi tiết hơn là ASP.NET MVC.

Ứng dụng sử dụng cơ sở dữ liệu SQL Server, triển khai dưới dạng cơ sở dữ liệu tập trung.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Thủ Tục

Store procedure (thủ tục thường trú): là một tập hợp chứa các dòng lệnh, các biến và các cấu trúc điều khiển trong ngôn ngữ Transaction-SQL dùng để thực hiện một hành động nào đó.

Lợi ích của thủ tục

- Tối ưu hóa việc phân tích, biên dịch và thực thi câu lệnh.
- Thực hiện một yêu cầu bằng một câu lệnh đơn giản hơn thay vì phải sử dụng nhiều dòng lệnh SQL khi thực thi → làm giảm thiểu sự lưu thông trên mạng
- Tăng khả năng bảo mật khi cấp phát quyền thông qua thủ tục
- Tốc độ xử lý của các thủ tục nội tại rất nhanh.

2.2. Con Trỏ

Cursor là một cấu trúc dữ liệu ánh xạ đến một tập các dòng dữ liệu kết quả của một câu truy vấn và được duyệt với từng dòng dữ liệu đó.

Vị trí hiện hành của cursor có thể được dùng như điều kiện where ở lệnh Update hoặc Delete.

2.3. Function

Cũng giống như Stored Procedure Hàm là một đối tượng trong cơ sở dữ liệu bao gồm một tập nhiều câu lệnh SQL được nhóm lại với nhau thành một nhóm.

Điểm khác biệt giữa hàm và thủ tục là hàm trả về một giá trị thông qua tên hàm. Điều này cho phép ta sử dụng hàm như là một thành phần của một biểu thức chẳng hạn như trong các câu lệnh truy vấn hay các câu lệnh thực hiện cập nhật dữ liệu.

2.4. Package

Package là một tập hợp các đối tượng được nhóm lại với nhau thành một tên. Package bao gồm các thủ tục, hàm, cursor, các khai báo, các kiểu và biến.

Package được sử dụng trong các ứng dụng mà các thủ tục và hàm thường được sử dụng lặp đi lặp lại.

Package gồm 2 phần

- Phần mô tả.
- Phần thân.

Phần mô tả: chứa thông tin về nội dung của package nhưng không phải mã lệnh của các thủ tục hoặc hàm.

Tất cả những đối tượng đặt trong phần mô tả của package được gọi là các đối tượng dùng chung.

Phần thân: chứa mã lệnh có thể thực thi thật sự của các đối tượng được chỉ định trong phần mô tả của package

Một số quy tắc trong phần thân:

- Phải có sự so khớp tuyệt đối giữa cursor, các tiêu đề của module và những định nghĩa trong phần mô tả.
- Không lặp lại phần khai báo trong phần thân đối với các biến, ngoại lệ hoặc các hằng đã khai báo trong phần mô tả.
- Bất kỳ phần nào đã khai báo trong phần mô tả có thể được tham chiếu trong phần thân.

2.5. Trigger

Mục đích của trigger

- Đáp ứng các quy tắc nghiệp vụ phức tạp mà không thể sử dụng các ràng buộc toàn vẹn để thay thế.
- Quản lý các quy tắc an toàn phức tạp.
- Tự động tạo ra các giá trị cho các cột nhận giá trị tính toán từ các cột khác.
- Ngăn chặn các thao tác không hợp lệ.
- Cung cấp cách kiểm tra các giá trị

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ DỮ LIỆU**3.1. Phân tích dữ liệu**

Hệ thống gồm các thực thể sau:

STT	Tên thực thể	Mô tả
1	CHUCVU	Chức vụ
2	PHONGBAN	Phòng ban
3	NHANVIEN	Nhân viên
4	KHACHHANG	Khách Hàng
5	NHACUNGCAP	Nhà cung cấp
6	PHIEUNHAPHANG	Phiếu mua hàng
7	CHITIET_PHIEUNHAP	Chi tiết phiếu mua hàng
8	LOAISANPHAM	Loại sản phẩm
9	SANPHAM	Sản phẩm
10	KICHCO_SANPHAM	Kích cỡ sản phẩm
11	DONHANG	Đơn hàng
12	CHITIET_DONHANG	Chi tiết đơn hàng

3.2. Chi tiết thực thể**3.2.1. Chi tiết thực thể “PHONGBAN”**

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaPB	Mã Phòng ban	Char(10)	PB00000001
2	TenPB	Tên Phòng ban	Nvarchar(100)	

3.2.2. Chi tiết thực thể “CHUCVU”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaCV	Mã Chức vụ	Char(10)	CVQTV
2	TenCV	Tên Chức vụ	Nvarchar(100)	

3.2.3. Chi tiết thực thể “NHANVIEN”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaNV	Mã Nhân viên	Char(10)	NV00000001
2	HoTen	Họ và tên Nhân viên	Nvarchar(100)	
3	SDT	Số điện thoại	Char(11)	
4	Email	Email	Char(100)	
5	NgaySinh	Ngày sinh	Date	
6	GioiTinh	Giới tính	Char(1)	
7	DiaChi	Địa chỉ	Nvarchar(200)	

8	Avatar	Hình đại diện	Char(30)	
9	Username	Tài khoản	Char(50)	
10	UserPass	Mật khẩu	Char(30)	
11	TinhTrang	Tình trạng Nhân viên	Int	
12	MaCV	Mã Chức vụ	Char(10)	CV00000001
13	MaPB	Mã Phòng ban	Char(10)	PB00000001

3.2.4. Chi tiết thực thể “KHACHHANG”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaKH	Mã Khách hàng	Char(10)	KH00000001
2	HoTen	Họ và tên Khách Hàng	Nvarchar(100)	
3	SDT	Số điện thoại	Char(11)	
4	Email	Email	Char(100)	
5	NgaySinh	Ngày sinh	Date	
6	GioiTinh	Giới tính	Char(1)	
7	DiaChi	Địa chỉ	Nvarchar(200)	

3.2.5. Chi tiết thực thể “NHACUNGCAP”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaNCC	Mã Nhà cung cấp	Char(10)	NCC00000001
2	TenNCC	Tên Nhà cung cấp	Nvarchar(100)	

3.2.6. Chi tiết thực thể “PHIEUNHAPHANG”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaPNH	Mã Phiếu mua hàng	Char(10)	PN000000001
2	NgayNhap	Ngày nhập hàng	Date	
3	TongTien	Tổng tiền	Decimal(18,2)	
4	MaNV	Mã Nhân Viên nhập	Char(10)	
5	MaNCC	Mã Nhà cung cấp	Char(10)	NCC00000001

3.2.7. Chi tiết thực thể “CHITIET_PHIEUNHAP”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaPNH	Mã Phiếu nhập hàng	Char(10)	PM000000001
2	MaSP	Mã sản phẩm	Char(10)	SP000000001
3	MaKC	Số lượng	Int	
4	SoLuongNhap	Đơn giá	Numeric(18,0)	
5	DonGiaNhap	Kích cỡ	Char(10)	
6	TinhTrang	Tình trạng đơn nhập	Nvarchar(255)	

3.2.8. Chi tiết thực thể “LOAISANPHAM”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaLoaiSP	Mã Loại sản phẩm	Char(10)	LS00000001
2	TenLoaiSP	Tên Loại sản phẩm	Nvarchar(100)	

3.2.9. Chi tiết thực thể “SANPHAM”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaSP	Mã Sản phẩm	Char(10)	SP00000001
2	TenSP	Tên sản phẩm	Nvarchar(100)	
3	HinhSP	Hình sản phẩm	Char(30)	
4	MoTa	Mô tả	Char(255)	
5	GiaSP	Giá sản phẩm	Numeric(18,0)	
6	TinhTrangSP	Tình trạng sản phẩm	Int	
7	MaLoaiSP	Mã loại sản phẩm	Char(10)	LSP0000001
8	MaNCC	Mã Nhà cung cấp	Char(10)	NC00000001

3.2.10. Chi tiết thực thể “KICHCO_SANPHAM”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaKC	Mã Kích Cỡ	Char(10)	KC00000001
2	MaSP	Mã Sản Phẩm	Char(10)	SP00000001
3	SoLuong	Số lượng	Int	

3.2.11. Chi tiết thực thể “DONHANG”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaDH	Mã Đơn hàng	Char(10)	DH00000001
2	NgayDatHang	Ngày Đặt hàng	Date	
3	PTThanhToan	Phương thức thanh toán	Nvarchar(50)	
4	TrangThaiDH	Trạng thái Đơn hàng	Int	
5	TongTien	Tổng tiền	Decimal(18,2)	
6	MaKH	Mã Khách hàng	Char(10)	KH00000001
7	MaNV	Mã Nhân Viên	Char(10)	NV00000001

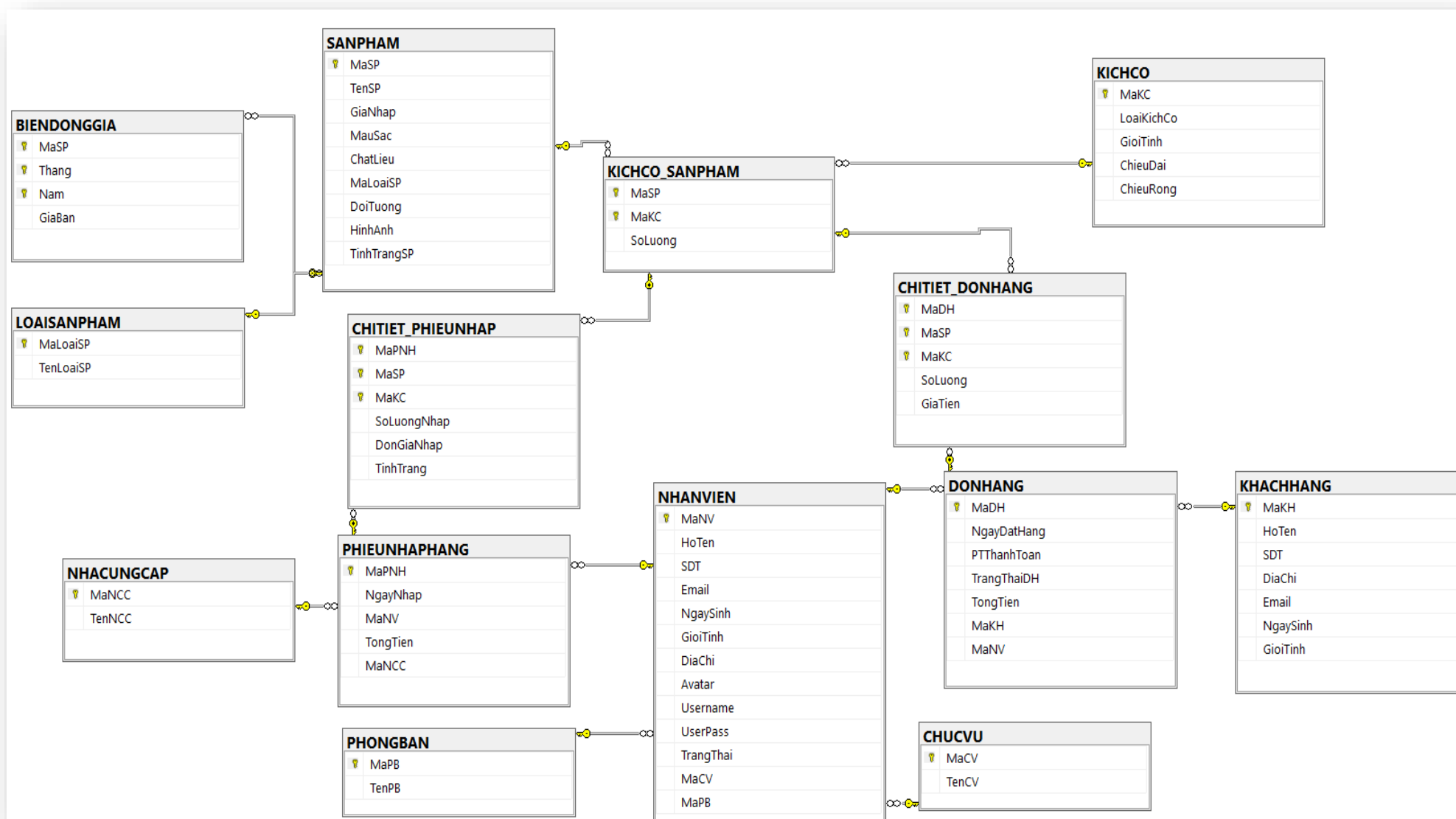
3.2.12. Chi tiết thực thể “CHITIET_DONHANG”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaDH	Mã Đơn Hàng	Char(10)	CD000000001
2	MaSP	Mã Sản Phẩm	Char(10)	SP000000001
3	MaKC	Mã Kích Cỡ	Char(10)	KC000000001
4	SoLuong	Số lượng	Int	
5	GiaTien	Giá 1 sản phẩm	Decimal(18,2)	

3.2.13. Chi tiết thực thể “KICHCO”

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Miền giá trị
1	MaKC	Mã Kích Cỡ	Char(10)	KC000000001
2	LoaiKichCo	Loại Kích Cỡ (số)	Float	
3	GioiTinh	Giá 1 sản phẩm	Char(1)	
4	ChieuDai	Chiều dài	Float	
5	ChieuRong	Chiều rộng	Float	

3.3. Thiết kế dữ liệu



CHƯƠNG 4: THIẾT KẾ XỬ LÝ**4.1. Danh sách khung nhìn (View)**

STT	Tên khung nhìn	Mô tả
1	V_NHANVIEN_LAMVIEC	Danh sách nhân viên đang làm việc
2	V_NHANVIEN_NGHIVIEC	Danh sách nhân viên nghỉ việc
3	V_SANPHAM_DANHSACH	Danh sách tất cả sản phẩm
4	V_DONHANG_THANHTOAN	Danh sách các đơn hàng đã thanh toán
5	V_NHAPHANG_NCC	Danh sách tổng số lần nhập hàng và tổng tiền
6	V_NHACUNGCAP_DS	Danh sách nhà cung cấp
7	V_NHANVIEN_DS	Danh sách nhân viên
8	V_KHACHHANG_DS	Danh sách khách hàng
9	V_DONHANG_CHITIET_DS	Danh sách chi tiết đơn hàng
10	V_NHAPHANG_CHITIET_DS	Danh sách chi tiết phiếu nhập hàng

4.2. Danh sách thủ tục (Store Procedured)

STT	Tên thủ tục	Mô tả
1	sp_ThemPhongBan	Thêm phòng ban
2	sp_XoaPhongBan	Xóa phòng ban
3	sp_CapNhatPhongBan	Cập nhật phòng ban
4	sp_ThemNhanVien	Thêm nhân viên
5	sp_XoaNhanVien	Xóa nhân viên
6	sp_CapNhatNhanVien	Cập nhật nhân viên
7	sp_Tong_NhanVien_NghiViec	Xem tổng nhân viên nghỉ việc
8	sp_Tong_NhanVien_DangLamViec	Xem tổng nhân viên đang làm việc
9	sp_TimKiem_NhanVien_Thuoc_PhongBan	Xem các nhân viên thuộc phòng ban
10	sp_TimKiem_NhanVien_TheoTen	Tìm kiếm nhân viên

11	sp_Tinh_TongTien_DonHang	Tính tổng tiền đơn hàng
12	sp_ThemNhaCungCap	Thêm nhà cung cấp
13	sp_XoaNhaCungCap	Xóa nhà cung cấp
14	sp_CapNhatNhaCungCap	Cập nhật nhà cung cấp
15	sp_ThemSanPham	Thêm sản phẩm
16	sp_XoASanPham	Xóa sản phẩm
17	sp_CapNhatSanPham	Cập nhật sản phẩm
18	sp_Tong_SanPham_DangKinhDoanh	Xem tổng số sản phẩm đang bán
19	sp_Tong_SanPham_NgungKinhDoanh	Xem tổng số sản phẩm ngừng bán
20	sp_TimKiem_LoiSanPham	Tìm kiếm loại sản phẩm
21	sp_TimKiem_SanPham	Tìm kiếm sản phẩm
22	sp_TimKiem_NhaCungCap	Tìm kiếm nhà cung cấp
23	sp_ThemChucVu	Thêm chức vụ
24	sp_XoaChucVu	Xóa chức vụ
25	sp_CapNhatChucVu	Cập nhật chức vụ
26	sp_ThemPhieuNhapHang	Thêm phiếu nhập hàng
27	sp_XoaPhieuNhapHang	Xóa phiếu nhập hàng
28	sp_CapNhatPhieuNhapHang	Cập nhật phiếu nhập hàng
29	sp_Tong_SoLuong_SanPham_DaMua	Xem tổng số sản phẩm đã nhập

30	sp_Tong_SoLan_MuaHang	Xem tổng số lần nhập hàng
31	sp_TimKiem_ChucVu	Tìm kiếm chức vụ
32	sp_TimKiem_PhieuMuaHang	Tìm kiếm phiếu mua hàng
33	sp_ThemKhachHang	Thêm khách hàng
34	sp_UPDATEKhachHang	Cập nhật khách hàng
35	sp_XoaKhachHang	Xóa khách hàng
36	sp_ThemLoaiSP	Thêm loại sản phẩm
37	sp_UPDATELoaiSP	Cập nhật loại sản phẩm
38	sp_TaoDonHang	Tạo đơn hàng
39	sp_MuaHang	Các mặt hàng được mua sẽ được thêm vào bảng chi tiết đơn hàng
40	sp_UPDATEDonHang	Cập nhật đơn hàng
41	sp_xoaCTDONHANG	Xóa sản phẩm đã đặt trong chi tiết đơn hàng
42	sp_UPDATECTDONHANG_SoLuong	Cập nhật số lượng mua của sản phẩm trong đơn hàng
43	sp_TimKiem_KhachHang	Tìm kiếm khách hàng
44	sp_TimKiem_DONHANG	Tìm kiếm đơn hàng
45	sp_NhapHang	Sản phẩm được nhập hàng sẽ được thêm vào bảng chi tiết nhập hàng với các số liệu tương ứng
46	sp_UPDATECTPhieuNhap_SoLuong	Cập nhật số lượng sản phẩm được nhập

4.3. Danh sách hàm (Function)

STT	Tên khung nhìn	Mô tả
1	fn_TimKiem_TenPhongBan	Tìm kiếm tên phòng ban
2	fn_TimKiem_NhanVien_TheoTen	Tìm kiếm nhân viên theo tên
3	fn_Tinh_ThanhTien_SanPham	Tính tổng tiền của sản phẩm
4	fn_TimKiem_LoaiSanPham	Tìm kiếm loại sản phẩm
5	fn_TimKiem_SanPham	Tìm kiếm sản phẩm
6	fn_TimKiem_NhaCungCap	Tìm kiếm nhà cung cấp
7	fn_TimKiem_ChucVu	Tìm kiếm chức vụ
8	fn_TimKiem_PhieuMuaHang	Tìm kiếm phiếu mua hàng
9	fn_TinhTongDoanhThuTheoThang	Tính tổng doanh thu theo tháng
10	fn_TimKiem_KHACHHANG	Tìm kiếm khách hàng
11	fn_TimKiem_DONHANG	Tìm kiếm đơn hàng

4.4. Danh sách Trigger

STT	Tên khung nhìn	Mô tả
1	trg_GiaTri_MacDinh_TinhTrang_SanPham	Tình trạng mặc định của sản phẩm khi được thêm vào luôn = 0
2	trg_KichCo_SoLuong_Zero	Nếu số lượng của tất cả kích cỡ của 1 sản phẩm = 0 thì cập nhật tình trạng sản phẩm = 0
3	trg_HuyDon	Khi hủy đơn sẽ trả các giá trị số lượng, tổng tiền về như ban đầu
4	trg_ThongBaoNhapHang	Khi số lượng sản phẩm của 1 sản phẩm nhỏ hơn 10 thì sẽ thông báo Nhập hàng

CHƯƠNG 5: TRIỂN KHAI THỰC NGHIỆM

5.1. Lệnh tạo bảng

Tên bảng	Lệnh SQL
BIENDONGGIA	<pre>CREATE TABLE [dbo].[BIENDONGGIA]([MaSP] [char](10) NOT NULL, [Thang] [int] NOT NULL, [Nam] [int] NOT NULL, [GiaBan] [decimal](18, 0) NOT NULL, CONSTRAINT [PK_BIENDONGGIA] PRIMARY KEY CLUSTERED ([MaSP] ASC, [Thang] ASC, [Nam] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] GO</pre>
CHITIET_DONHANG	<pre>CREATE TABLE [dbo].[CHITIET_DONHANG]([MaDH] [char](10) NOT NULL, [MaSP] [char](10) NOT NULL, [MaKC] [char](10) NOT NULL, [SoLuong] [int] NOT NULL, [GiaTien] [decimal](18, 0) NOT NULL, CONSTRAINT [PK_CHITIET_DONHANG] PRIMARY KEY CLUSTERED ([MaDH] ASC, [MaSP] ASC, [MaKC] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]</pre>
CHITIET_PHIEUNHAP	<pre>CREATE TABLE [dbo].[CHITIET_PHIEUNHAP]([MaPNH] [char](10) NOT NULL, [MaSP] [char](10) NOT NULL, [MaKC] [char](10) NOT NULL,</pre>

	<pre> [SoLuongNhap] [int] NOT NULL, [DonGiaNhap] [decimal](18, 0) NOT NULL, [TinhTrang] [int] NOT NULL, CONSTRAINT [PK_CHITIET_PHIEUNHAP] PRIMARY KEY CLUSTERED ([MaPNH] ASC, [MaSP] ASC, [MaKC] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
CHUCVU	<pre> CREATE TABLE [dbo].[CHUCVU]([MaCV] [char](10) NOT NULL, [TenCV] [nvarchar](50) NULL, CONSTRAINT [PK_CHUCVU] PRIMARY KEY CLUSTERED ([MaCV] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
DONHANG	<pre> CREATE TABLE [dbo].[DONHANG]([MaDH] [char](10) NOT NULL, [NgayDatHang] [date] NOT NULL, [PTThanhToan] [nvarchar](50) NOT NULL, [TrangThaiDH] [int] NOT NULL, [TongTien] [decimal](18, 0) NOT NULL, [MaKH] [char](10) NOT NULL, [MaNV] [char](10) NOT NULL, CONSTRAINT [PK_DONHANG] PRIMARY KEY CLUSTERED ([MaDH] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, </pre>

	<pre> OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
KHACHANG	<pre> CREATE TABLE [dbo].[KHACHHANG]([MaKH] [char](10) NOT NULL, [HoTen] [nvarchar](100) NOT NULL, [SDT] [char](11) NOT NULL, [DiaChi] [nvarchar](max) NOT NULL, [Email] [char](50) NULL, [NgàySinh] [date] NOT NULL, [GioiTinh] [char](1) NOT NULL, CONSTRAINT [PK_KHACHHANG] PRIMARY KEY CLUSTERED ([MaKH] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY] </pre>
KICHCO	<pre> CREATE TABLE [dbo].[KICHCO]([MaKC] [char](10) NOT NULL, [LoaiKichCo] [float] NOT NULL, [GioiTinh] [char](1) NULL, [ChieuDai] [float] NOT NULL, [ChieuRong] [float] NOT NULL, CONSTRAINT [PK_KICHCO] PRIMARY KEY CLUSTERED ([MaKC] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
KICHCO_SANPHAM	<pre> CREATE TABLE [dbo].[KICHCO_SANPHAM]([MaSP] [char](10) NOT NULL, [MaKC] [char](10) NOT NULL, [SoLuong] [int] NOT NULL, CONSTRAINT [PK_KICHCO_SANPHAM] PRIMARY KEY CLUSTERED (</pre>

	<pre> [MaSP] ASC, [MaKC] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
LOAISANPHAM	<pre> CREATE TABLE [dbo].[LOAISANPHAM]([MaLoaiSP] [char](10) NOT NULL, [TenLoaiSP] [nvarchar](50) NULL, CONSTRAINT [PK_LOAISANPHAM] PRIMARY KEY CLUSTERED ([MaLoaiSP] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
NHACUNGCAP	<pre> CREATE TABLE [dbo].[NHACUNGCAP]([MaNCC] [char](10) NOT NULL, [TenNCC] [nvarchar](100) NULL, CONSTRAINT [PK_NHACUNGCAP] PRIMARY KEY CLUSTERED ([MaNCC] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
NHANVIEN	<pre> CREATE TABLE [dbo].[NHANVIEN]([MaNV] [char](10) NOT NULL, [HoTen] [nvarchar](100) NOT NULL, [SDT] [char](11) NOT NULL, [Email] [char](50) NOT NULL, [NgàySinh] [date] NOT NULL, [GioiTinh] [char](1) NOT NULL, [DiaChi] [nvarchar](max) NOT NULL, [Avatar] [char](20) NULL, </pre>

	<pre> [Username] [char](50) NOT NULL, [UserPass] [char](30) NOT NULL, [TrangThai] [int] NOT NULL, [MaCV] [char](10) NOT NULL, [MaPB] [char](10) NOT NULL, CONSTRAINT [PK_NHANVIEN] PRIMARY KEY CLUSTERED ([MaNV] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY] </pre>
PHIEUNHAPHANG	<pre> CREATE TABLE [dbo].[PHIEUNHAPHANG]([MaPNH] [char](10) NOT NULL, [NgayNhap] [datetime] NOT NULL, [MaNV] [char](10) NOT NULL, [TongTien] [decimal](18, 0) NOT NULL, [MaNCC] [char](10) NOT NULL, CONSTRAINT [PK_PHIEUNHAPHANG] PRIMARY KEY CLUSTERED ([MaPNH] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>
PHONGBAN	<pre> CREATE TABLE [dbo].[PHONGBAN]([MaPB] [char](10) NOT NULL, [TenPB] [nvarchar](50) NULL, CONSTRAINT [PK_PHONGBAN] PRIMARY KEY CLUSTERED ([MaPB] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY] </pre>

) ON [PRIMARY]
SANPHAM	<pre> CREATE TABLE [dbo].[SANPHAM]([MaSP] [char](10) NOT NULL, [TenSP] [nvarchar](100) NOT NULL, [GiaNhap] [decimal](18, 0) NULL, [MauSac] [nvarchar](20) NOT NULL, [ChatLieu] [nvarchar](50) NOT NULL, [MaLoaiSP] [char](10) NOT NULL, [DoiTuong] [nvarchar](20) NOT NULL, [HinhAnh] [char](20) NULL, [TinhTrangSP] [int] NOT NULL, CONSTRAINT [PK_SANPHAM] PRIMARY KEY CLUSTERED ([MaSP] ASC)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] </pre>

5.2. Lệnh tạo ràng buộc

Tên bảng	Lệnh SQL
BIENDONGGIA	<pre> ALTER TABLE [dbo].[BIENDONGGIA] WITH CHECK ADD CONSTRAINT [FK_BIENDONGGIA_SANPHAM] FOREIGN KEY([MaSP]) REFERENCES [dbo].[SANPHAM] ([MaSP]) </pre>
CHITIET_DONHANG	<pre> ALTER TABLE [dbo].[CHITIET_DONHANG] WITH CHECK ADD CONSTRAINT [FK_CHITIET_DONHANG_DONHANG] FOREIGN KEY([MaDH]) REFERENCES [dbo].[DONHANG] ([MaDH]) ALTER TABLE [dbo].[CHITIET_DONHANG] WITH CHECK ADD CONSTRAINT [FK_CHITIET_DONHANG_KICHCO_SANPHAM] FOREIGN KEY([MaSP], [MaKC]) REFERENCES [dbo].[KICHCO_SANPHAM] ([MaSP], [MaKC]) </pre>
CHITIET_PHIEUNHAP	<pre> ALTER TABLE [dbo].[CHITIET_PHIEUNHAP] WITH CHECK ADD CONSTRAINT [FK_CHITIET_PHIEUNHAP_KICHCO_SANPHAM] FOREIGN KEY([MaSP], [MaKC]) </pre>

	<p>REFERENCES [dbo].[KICHCO_SANPHAM] ([MaSP], [MaKC])</p> <p>ALTER TABLE [dbo].[CHITIET_PHIEUNHAP] WITH CHECK ADD CONSTRAINT [FK_CHITIET_PHIEUNHAP_PHIEUNHAPHANG] FOREIGN KEY([MaPNH]) REFERENCES [dbo].[PHIEUNHAPHANG] ([MaPNH])</p>
DONHANG	<p>ALTER TABLE [dbo].[DONHANG] WITH CHECK ADD CONSTRAINT [FK_DONHANG_KHACHHANG] FOREIGN KEY([MaKH]) REFERENCES [dbo].[KHACHHANG] ([MaKH])</p> <p>ALTER TABLE [dbo].[DONHANG] WITH CHECK ADD CONSTRAINT [FK_DONHANG_NHANVIEN] FOREIGN KEY([MaNV]) REFERENCES [dbo].[NHANVIEN] ([MaNV])</p>
KICHCO_SANPHAM	<p>ALTER TABLE [dbo].[KICHCO_SANPHAM] WITH CHECK ADD CONSTRAINT [FK_KICHCO_SANPHAM_KICHCO] FOREIGN KEY([MaKC]) REFERENCES [dbo].[KICHCO] ([MaKC])</p> <p>ALTER TABLE [dbo].[KICHCO_SANPHAM] WITH CHECK ADD CONSTRAINT [FK_KICHCO_SANPHAM_SANPHAM] FOREIGN KEY([MaSP]) REFERENCES [dbo].[SANPHAM] ([MaSP])</p>
SANPHAM	<p>ALTER TABLE [dbo].[SANPHAM] WITH CHECK ADD CONSTRAINT [FK_SANPHAM_LOAISANPHAM] FOREIGN KEY([MaLoaiSP]) REFERENCES [dbo].[LOAISANPHAM] ([MaLoaiSP])</p>
NHANVIEN	<p>ALTER TABLE [dbo].[NHANVIEN] WITH CHECK ADD CONSTRAINT [FK_NHANVIEN_CHUCVU] FOREIGN KEY([MaCV]) REFERENCES [dbo].[CHUCVU] ([MaCV])</p> <p>ALTER TABLE [dbo].[NHANVIEN] WITH CHECK ADD CONSTRAINT [FK_NHANVIEN_PHONGBAN] FOREIGN KEY([MaPB]) REFERENCES [dbo].[PHONGBAN] ([MaPB])</p>
PHIEUNHAPHANG	<p>ALTER TABLE [dbo].[PHIEUNHAPHANG] WITH CHECK ADD CONSTRAINT</p>

	<pre>[FK_PHIEUNHAPHANG_NHACUNGCAP] FOREIGN KEY([MaNCC]) REFERENCES [dbo].[NHACUNGCAP] ([MaNCC]) ALTER TABLE [dbo].[PHIEUNHAPHANG] WITH CHECK ADD CONSTRAINT [FK_PHIEUNHAPHANG_NHANVIEN] FOREIGN KEY([MaNV]) REFERENCES [dbo].[NHANVIEN] ([MaNV])</pre>
PHONGBAN	<pre>ALTER TABLE [dbo].[PHONGBAN] ADD CONSTRAINT [UC_TenPB] UNIQUE ([TenPB])</pre>
CHUCVU	<pre>ALTER TABLE [dbo].[CHUCVU] ADD CONSTRAINT [UC_TenCV] UNIQUE ([TenCV])</pre>

5.3. Lệnh tạo khung nhìn (View)

Tên khung nhìn	Lệnh SQL
V_NHANVIEN_LAMVIEC	<pre>CREATE OR ALTER VIEW V_NHANVIEN_LAMVIEC AS SELECT [MaNV], [HoTen], [SDT], [Email], [TenCV], [TenPB] FROM NHANVIEN, CHUCVU, PHONGBAN WHERE NHANVIEN.TrangThai = 1 AND NHANVIEN.MaCV = CHUCVU.MaCV AND NHANVIEN.MaPB = PHONGBAN.MaPB</pre>
V_NHANVIEN_NGHI VIEC	<pre>CREATE OR ALTER VIEW V_NHANVIEN_NGHI VIEC AS SELECT [MaNV], [HoTen], [SDT], [Email], [TenCV], [TenPB] FROM NHANVIEN, CHUCVU, PHONGBAN WHERE NHANVIEN.TrangThai = 0 AND NHANVIEN.MaCV = CHUCVU.MaCV AND NHANVIEN.MaPB = PHONGBAN.MaPB</pre>
V_SANPHAM_DANH SACH	<pre>CREATE OR ALTER VIEW V_SANPHAM_DANH SACH AS SELECT SANPHAM.[MaSP], TenSP, LoaiKichCo, GioiTinh, TenLoaiSP, GiaBan, SoLuong FROM SANPHAM, KICHCO_SANPHAM, KICHCO, LOAISANPHAM, BIENDONGGIA WHERE SANPHAM.MaSP = KICHCO_SANPHAM.MaSP AND KICHCO.MaKC = KICHCO_SANPHAM.MaKC</pre>

	<p>AND SANPHAM.MaSP = BIENDONGGIA.MaSP AND SANPHAM.MaLoaiSP = LOAISANPHAM.MaLoaiSP</p>
V_DONHANG_THANH TOAN	<p>CREATE OR ALTER VIEW V_DONHANG_THANH TOAN AS SELECT DONHANG.[MaDH], NgayDatHang, TongTien, KHACHHANG.HoTen AS HOTEN_KHACH, NHANVIEN.HoTen AS HOTEN_NHANVIEN FROM DONHANG, KHACHHANG, NHANVIEN WHERE DONHANG.MaKH = KHACHHANG.MaKH AND DONHANG.MaNV = NHANVIEN.MaNV AND TrangThaiDH = 1</p>
V_NHAPHANG_NCC	<p>CREATE OR ALTER VIEW V_NHAPHANG_NCC AS SELECT [TenNCC], COUNT(MaPNH) AS [TongLanNhap], SUM(TongTien) AS [TongTienNhap] FROM NHACUNGCAP JOIN PHIEUNHAPHANG ON NHACUNGCAP.MaNCC = PHIEUNHAPHANG.MaNCC GROUP BY PHIEUNHAPHANG.MaNCC, TenNCC</p>
V_NHACUNGCAP_DS	<p>CREATE OR ALTER VIEW V_NHACUNGCAP_DS AS SELECT * FROM NHACUNGCAP</p>
V_NHANVIEN_DS	<p>CREATE OR ALTER VIEW V_NHANVIEN_DS AS SELECT [MaNV], [HoTen], [SDT], [Email], [TenCV], [TenPB] FROM NHANVIEN, PHONGBAN, CHUCVU WHERE NHANVIEN.MaCV = CHUCVU.MaCV AND NHANVIEN.MaPB = PHONGBAN.MaPB</p>
V_KHACHHANG_DS	<p>CREATE OR ALTER VIEW V_KHACHHANG_DS AS SELECT [MaKH], [HoTen], [SDT], [Email], [DiaChi] FROM KHACHHANG</p>
V_DONHANG_CHITI ET_DS	<p>CREATE OR ALTER VIEW V_DONHANG_CHITIET_DS</p>

	<pre> AS SELECT DH.[MaDH], [TenSP], [LoaiKichCo], KC.[GioiTinh], CTD.[SoLuong], [GiaTien], KH.HoTen AS HOTEN_KHACH, NV.HoTen AS HOTEN_NHANVIEN FROM DONHANG DH, CHITIET_DONHANG CTD, KHACHHANG KH, NHANVIEN NV, SANPHAM SP, KICHCO_SANPHAM KCSP, KICHCO KC WHERE DH.MaDH = CTD.MaDH AND CTD.MaSP = KCSP.MaSP AND CTD.MaKC = KCSP.MaKC AND KCSP.MaSP = SP.MaSP AND KCSP.MaKC = KC.MaKC AND DH.MaKH = KH.MaKH AND DH.MaNV = NV.MaNV </pre>
V_NHAPHANG_CHITIET_DS	<pre> CREATE OR ALTER VIEW V_NHAPHANG_CHITIET_DS AS SELECT NH.[MaPNH], [TENSP], [LoaiKichCo], KC.[GioiTinh], [SoLuongNhap], [DonGiaNhap], [TenNCC], [HoTen] FROM NHACUNGCAP NCC, PHIEUNHAPHANG NH, CHITIET_PHIEUNHAP CTNH, SANPHAM SP, KICHCO KC, KICHCO_SANPHAM KCSP, NHANVIEN NV WHERE NCC.MaNCC = NH.MaNCC AND NH.MaPNH = CTNH.MaPNH AND CTNH.MaSP = KCSP.MaSP AND CTNH.MaKC = KCSP.MaKC AND NH.MaNV = NV.MaNV AND KCSP.MaKC = KC.MaKC AND KCSP.MaSP = SP.MaSP </pre>

5.4. Lệnh tạo thủ tục (Store Procedured)

Tên thủ tục	Lệnh SQL
sp_ThemPhongBan	<pre> CREATE OR ALTER PROC sp_ThemPhongBan @mapb CHAR(10), @tenpb NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF NOT EXISTS (SELECT * FROM PHONGBAN WHERE MaPB = @mapb) BEGIN </pre>

	<pre> INSERT INTO PHONGBAN (MaPB, TenPB) VALUES (@mapb, @tenpb); PRINT N'Thêm phòng ban thành công'; END; ELSE RAISERROR(N'Đã tồn tại mã phòng ban: %s', 16, 1, @mapb); END; </pre>
sp_XoaPhongBan	<pre> CREATE OR ALTER PROC sp_XoaPhongBan @mapb CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM PHONGBAN WHERE MaPB = @mapb) BEGIN DELETE FROM PHONGBAN WHERE MaPB = @mapb; PRINT N'Xóa phòng ban thành công'; END; ELSE RAISERROR(N'Không tìm thấy phòng ban có mã: %s.', 16, 1, @mapb); END; </pre>
sp_CapNhatPhongBan	<pre> CREATE OR ALTER PROC sp_CapNhatPhongBan @mapb CHAR(10), @tenpb NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM PHONGBAN WHERE MaPB = @mapb) BEGIN UPDATE PHONGBAN SET TenPB = @tenpb WHERE MaPB = @mapb; PRINT N'Cập nhật phòng ban thành công'; END; ELSE </pre>

	<pre> RAISERROR(N'Không tìm thấy phòng ban có mã: %s.', 16, 1, @mapb); END; </pre>
sp_ThemNhanVien	<pre> CREATE OR ALTER PROC sp_ThemNhanVien @manv CHAR(10), @hoten NVARCHAR(100), @sdt CHAR(11), @email CHAR(100), @ngaysinh DATE, @gioitinh CHAR(1), @diachi NVARCHAR(200), @username CHAR(50), @userpASs CHAR(30), @tinhtrang INT, @macv CHAR(10), @mapb CHAR(10) AS BEGIN SET NOCOUNT ON; IF NOT EXISTS (SELECT * FROM NHANVIEN WHERE MaNV = @manv) BEGIN INSERT INTO NHANVIEN (MaNV, HoTen, SDT, Email, NgaySinh, GioiTinh, DiaChi, Username, UserPASs, TrangThai, MaCV, MaPB) VALUES (@manv, @hoten, @sdt, @email, @ngaysinh, @gioitinh, @diachi, @username, @userpASs, @tinhtrang, @macv, @mapb); PRINT N'Thêm nhân viên thành công'; END; ELSE RAISERROR(N'Đã tồn tại mã nhân viên: %s', 16, 1, @manv); END; </pre>
sp_XoaNhanVien	<pre> CREATE OR ALTER PROC sp_XoaNhanVien @manv CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM NHANVIEN WHERE MaNV = @manv) BEGIN DELETE FROM NHANVIEN WHERE MaNV = @manv; PRINT N'Xóa nhân viên thành công'; END; </pre>

	<pre> ELSE RAISERROR(N'Không tìm thấy nhân viên có mã: %s.', 16, 1, @manv); END; </pre>
sp_CapNhatNhanVien	<pre> CREATE OR ALTER PROC sp_CapNhatNhanVien @manv CHAR(10), @hoten NVARCHAR(100), @sdt CHAR(11), @email CHAR(100), @ngaysinh DATE, @gioitinh CHAR(1), @diachi NVARCHAR(200), @username CHAR(50), @userpASs CHAR(30), @tinhtrang INT, @macv CHAR(10), @mapb CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM NHANVIEN WHERE MaNV = @manv) BEGIN UPDATE NHANVIEN SET HoTen = @hoten, SDT = @sdt, Email = @email, NgaySinh = @ngaysinh, GioiTinh = @gioitinh, DiaChi = @diachi, Username = @username, UserPASs = @userpASs, TrangThai = @tinhtrang, MaCV = @macv, MaPB = @mapb WHERE MaNV = @manv; PRINT N'Cập nhật nhân viên thành công'; END; ELSE RAISERROR(N'Không tìm thấy nhân viên có mã: %s.', 16, 1, @manv); END; </pre>
sp_Tong_NhanVien_NghiViec	<pre> CREATE OR ALTER PROC sp_Tong_NhanVien_NghiViec AS BEGIN SELECT COUNT(*) AS TongSoNhanVienNghiViec FROM NHANVIEN WHERE TrangThai = 0; -- 0 là tình trạng nghỉ việc END; </pre>

sp_Tong_NhanVien_DangLamViec	<pre> CREATE OR ALTER PROC sp_Tong_NhanVien_DangLamViec AS BEGIN SELECT COUNT(*) AS TongSoNhanViENDangLamViec FROM NHANVIEN WHERE TrangThai != 0; END; </pre>
sp_TimKiem_NhanVien_Thuoc_PhongBan	<pre> CREATE OR ALTER PROC sp_TimKiem_NhanVien_Thuoc_PhongBan (@tenpb NVARCHAR(100)) AS BEGIN SET NOCOUNT ON; SELECT * FROM NHANVIEN WHERE MaPB IN (SELECT MaPB FROM dbo.fn_TimKiem_TenPhongBan(@tenpb)); END; </pre>
sp_TimKiem_NhanVien_TheoTen	<pre> CREATE OR ALTER PROC sp_TimKiem_NhanVien_TheoTen @tennv NVARCHAR(100) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_NhanVien_TheoTen(@tennv); END; </pre>
sp_Tinh_TongTien_DonHang	<pre> CREATE OR ALTER PROC sp_Tinh_TongTien_DonHang @madh CHAR(10) AS BEGIN </pre>

	<pre> SELECT MaDH,SUM(dbo.fn_Tinh_ThanhTien_SanPham(SoLuong ,GiaTien)) AS TongTien FROM CHITIET_DONHANG WHERE MaDH = @madh GROUP BY MADH END;</pre>
sp_ThemNhaCungCap	<pre> CREATE OR ALTER PROC sp_ThemNhaCungCap @mancc CHAR(10), @tenncc NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF NOT EXISTS (SELECT * FROM NHACUNGCAP WHERE MaNCC = @mancc) BEGIN INSERT INTO NHACUNGCAP (MaNCC, TenNCC) VALUES (@mancc, @tenncc); PRINT N'Thêm nhà cung cấp thành công'; END; ELSE RAISERROR(N'Đã tồn tại mã nhà cung cấp: %s', 16, 1, @mancc); END;</pre>
sp_XoaNhaCungCap	<pre> CREATE OR ALTER PROC sp_XoaNhaCungCap @mancc CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM NHACUNGCAP WHERE MaNCC = @mancc) BEGIN DELETE FROM NHACUNGCAP WHERE MaNCC = @mancc; PRINT N'Xóa nhà cung cấp thành công'; END; ELSE RAISERROR(N'Không tìm thấy nhà cung cấp có mã: %s.', 16, 1, @mancc);</pre>

	END;
sp_CapNhatNhaCungCap	<pre> CREATE OR ALTER PROC sp_CapNhatNhaCungCap @mancc CHAR(10), @tenncc NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM NHACUNGCAP WHERE MaNCC = @mancc) BEGIN UPDATE NHACUNGCAP SET TenNCC = @tenncc WHERE MaNCC = @mancc; PRINT N'Cập nhật nhà cung cấp thành công'; END; ELSE RAISERROR(N'Không tìm thấy nhà cung cấp có mã: %s.', 16, 1, @mancc); END; </pre>
sp_ThemSanPham	<pre> CREATE OR ALTER PROC sp_ThemSanPham @tensp NVARCHAR(100), @giASp NUMERIC(18,0), @malois CHAR(10), @mausac NVARCHAR(20), @chatlieu NVARCHAR(50), @doituong NVARCHAR(50), @tinhtrang INT AS BEGIN SET NOCOUNT ON; DECLARE @MASP CHAR(10) SELECT TOP 1 @MASP = MASP FROM SANPHAM WHERE MASP LIKE 'SP%' AND TenSP IS NULL ORDER BY MASP ASC IF @MASP IS NULL BEGIN DECLARE @lAst_MASP CHAR(10) SELECT TOP 1 @lAst_MASP = MASP FROM SANPHAM WHERE MASP LIKE 'SP%' ORDER BY MASP DESC DECLARE @next_MASP INT SET @next_MASP = CAST(RIGHT(@lAst_MASP, 8) AS INT) + 1 </pre>

	<pre> SET @MASP = 'SP' + RIGHT('00000000' + CAST(@next_MASP AS VARCHAR(8)), 8) END IF NOT EXISTS (SELECT * FROM SANPHAM WHERE MASP = @mASp) BEGIN INSERT INTO SANPHAM (MASP, TenSP, GiaNhap, MaLoaiSP, MauSac, ChatLieu, DoiTuong, TinhTrangSP) VALUES (@mASp, @tensp, @giASp, @maloaish, @mausac, @chatlieu, @doituong, @tinhtrang) PRINT N'Thêm sản phẩm thành công'; END; ELSE RAISERROR(N'Đã tồn tại mã sản phẩm: %s', 16, 1, @mASp); END; </pre>
sp_XoASanPham	<pre> CREATE OR ALTER PROC sp_XoASanPham @mASp CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM SANPHAM WHERE MASP = @mASp) BEGIN DELETE FROM SANPHAM WHERE MASP = @mASp; PRINT N'Xóa sản phẩm thành công'; END; ELSE RAISERROR(N'Không tìm thấy sản phẩm có mã: %s.', 16, 1, @mASp); END; </pre>
sp_CapNhatSanPham	<pre> CREATE OR ALTER PROC sp_CapNhatSanPham @mASp CHAR(10), @tensp NVARCHAR(100), @giASp NUMERIC(18,0), @maloaish CHAR(10), @mausac NVARCHAR(20), @chatlieu NVARCHAR(50), @doituong NVARCHAR(50) AS BEGIN SET NOCOUNT ON; </pre>

	<pre> IF EXISTS (SELECT * FROM SANPHAM WHERE MASP = @mASp) BEGIN UPDATE SANPHAM SET TenSP = @tensp, GiaNhap = @giASp, MaLoaiSP = @maloaisp, MauSac = @mausac, ChatLieu = @chatlieu, DoiTuong = @doituong WHERE MASP = @mASp; PRINT N'Cập nhật sản phẩm thành công'; END; ELSE RAISERROR(N'Không tìm thấy sản phẩm có mã: %s.', 16, 1, @mASp); END; </pre>
sp_Tong_SanPham_Dang KinhDoanh	<pre> CREATE OR ALTER PROC sp_Tong_SanPham_DangKinhDoanh AS BEGIN SELECT COUNT(*) AS TongSoLuongSanPhamDangKinhDoanh FROM SANPHAM WHERE TinhTrangSP != 0; END; </pre>
sp_Tong_SanPham_Ngun gKinhDoanh	<pre> CREATE OR ALTER PROC sp_Tong_SanPham_NgungKinhDoanh AS BEGIN SELECT COUNT(*) AS TongSoLuongSanPhamNgungKinhDoanh FROM SANPHAM WHERE TinhTrangSP = 0; -- 0 là tình trạng ngừng kinh doanh END; </pre>
sp_TimKiem_LoiSanPha m	<pre> CREATE OR ALTER PROC sp_TimKiem_LoiSanPham @tenloaisp NVARCHAR(100) AS BEGIN SET NOCOUNT ON; SELECT * </pre>

	<pre> FROM dbo.fn_TimKiem_LoaiSanPham(@tenloaisp); END;</pre>
sp_TimKiem_SanPham	<pre> CREATE OR ALTER PROC sp_TimKiem_SanPham @tensp NVARCHAR(100) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_SanPham(@tensp); END;</pre>
sp_TimKiem_NhaCungCap	<pre> CREATE OR ALTER PROC sp_TimKiem_NhaCungCap @tenncc NVARCHAR(100) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_NhaCungCap(@tenncc); END;</pre>
sp_ThemChucVu	<pre> CREATE OR ALTER PROC sp_ThemChucVu @macv CHAR(10), @tencv NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF NOT EXISTS (SELECT * FROM CHUCVU WHERE MaCV = @macv) BEGIN INSERT INTO CHUCVU(MaCV, TenCV) VALUES (@macv, @tencv); PRINT N'Thêm nhà chức vụ thành công'; END; ELSE RAISERROR(N'Đã tồn tại mã chức vụ: %s', 16, 1, @macv); END;</pre>
sp_XoaChucVu	<pre> CREATE OR ALTER PROC sp_XoaChucVu @macv CHAR(10)</pre>

	<pre> AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM CHUCVU WHERE MaCV = @macv) BEGIN DELETE FROM CHUCVU WHERE MaCV = @macv; PRINT N'Xóa chức vụ thành công'; END; ELSE RAISERROR(N'Không tìm thấy chức vụ có mã: %s.', 16, 1, @macv); END; </pre>
sp_CapNhatChucVu	<pre> CREATE OR ALTER PROC sp_CapNhatChucVu @macv CHAR(10), @tencv NVARCHAR(100) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM CHUCVU WHERE MaCV = @macv) BEGIN UPDATE CHUCVU SET TenCV = @tencv WHERE MaCV = @macv; PRINT N'Cập nhật chức vụ thành công'; END; ELSE RAISERROR(N'Không tìm thấy chức vụ có mã: %s.', 16, 1, @macv); END; </pre>
sp_ThemPhieuNhapHang	<pre> CREATE OR ALTER PROC sp_ThemPhieuNhapHang @ngaynhap DATE, @manv CHAR(10), @mancc CHAR(10) AS BEGIN SET NOCOUNT ON; DECLARE @tongtien DECIMAL(18,2) SET @tongtien = 0 </pre>

	<pre> DECLARE @MaPNH CHAR(10) Begin transaction SELECT TOP 1 @MAPNH = MaPNH FROM PHIEUNHAPHANG with (XLOCK) WHERE MAPNH LIKE 'PN%' AND NgayNhap IS NULL ORDER BY MaPNH ASC IF @MaPNH IS NULL BEGIN DECLARE @lAst_MaPNH CHAR(10) SELECT TOP 1 @lAst_MaPNH = MaPNH FROM PHIEUNHAPHANG WHERE MaPNH LIKE 'PN%' ORDER BY MaPNH DESC DECLARE @next_MaPNH INT SET @next_MaPNH = CAST(RIGHT(@lAst_MaPNH, 8) AS INT) + 1 SET @MaPNH = 'PN' + RIGHT('00000000' + CAST(@next_MaPNH AS VARCHAR(8)), 8) END INSERT INTO PHIEUNHAPHANG(MaPNH, NgayNhap, TongTien, MaNV, MaNCC) VALUES (@MaPNH, @ngaynhap, @tongtien, @manv, @mancc); PRINT N'Thêm phiếu nhập hàng thành công'; commit END; </pre>
sp_XoaPhieuNhapHang	<pre> CREATE OR ALTER PROC sp_XoaPhieuNhapHang @mapmh CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM PHIEUNHAPHANG WHERE MaPNH = @mapmh) BEGIN DELETE FROM PHIEUNHAPHANG WHERE MaPNH = @mapmh; PRINT N'Xóa phiếu mua hàng thành công'; END; ELSE </pre>

	<pre> RAISERROR(N'Không tìm thấy phiếu mua hàng có mã: %s.', 16, 1, @mapmh); END; </pre>
sp_CapNhatPhieuNhapHang	<pre> CREATE OR ALTER PROC sp_CapNhatPhieuNhapHang @mapmh CHAR(10), @ngaynhap DATE, @tongtien DECIMAL(18,2), @manv CHAR(10), @mancc CHAR(10) AS BEGIN SET NOCOUNT ON; IF EXISTS (SELECT * FROM PHIEUNHAPHANG WHERE MaPNH = @mapmh) BEGIN UPDATE PHIEUNHAPHANG SET NgayNhap = @ngaynhap, TongTien = @tongtien, MaNV=@manv, MaNCC = @mancc WHERE MaPNH = @mapmh; PRINT N'Cập nhật phiếu mua hàng thành công'; END; ELSE RAISERROR(N'Không tìm thấy phiếu mua hàng có mã: %s.', 16, 1, @mapmh); END; </pre>
sp_Tong_SoLuong_SanPham_DaMua	<pre> CREATE OR ALTER PROC sp_Tong_SoLuong_SanPham_DaMua @mancc CHAR(10) AS BEGIN SELECT SUM(ct.SoLuongNhap) AS TongSoSanPham FROM PHIEUNHAPHANG pm INNER JOIN CHITIET_PHIEUNHAP ct ON ct.MaPNH = pm.MaPNH WHERE pm.MaNCC = @mancc; END; </pre>
sp_Tong_SoLan_MuaHang	<pre> CREATE OR ALTER PROC sp_Tong_SoLan_MuaHang @mancc CHAR(10) AS BEGIN </pre>

	<pre> SELECT COUNT(*) AS TongSoLanMuaHang FROM PHIEUNHAPHANG WHERE MaNCC = @mancc; END; </pre>
sp_TimKiem_ChucVu	<pre> CREATE OR ALTER PROC sp_TimKiem_ChucVu @tencv NVARCHAR(100) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_ChucVu(@tencv); END; </pre>
sp_TimKiem_PhieuMuaHang	<pre> CREATE OR ALTER PROC sp_TimKiem_PhieuMuaHang @mapmh CHAR(10) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_PhieuMuaHang(@mapmh); END; </pre>
sp_ThemKhachHang	<pre> CREATE OR ALTER PROC sp_ThemKhachHang @hoten NVARCHAR(100), @sdt CHAR(11), @diachi NVARCHAR(200), @email CHAR(100), @ngaysinh CHAR(10), @gioitinh CHAR(1) AS BEGIN DECLARE @MAKH CHAR(10) begin transaction SELECT TOP 1 @MAKH = MAKH FROM KHACHHANG with (XLOCK) WHERE MAKH LIKE 'KH%' AND hoten IS NULL ORDER BY MAKH ASC IF @MAKH IS NULL BEGIN DECLARE @lASt_makh CHAR(10) </pre>

	<pre> SELECT TOP 1 @lAst_makh = MAKH FROM KHACHHANG WHERE MAKH LIKE 'KH%' ORDER BY MAKH DESC DECLARE @next_makh INT SET @next_makh = CAST(RIGHT(@lAst_makh, 8) AS INT) + 1 SET @MAKH = 'KH' + RIGHT('00000000' + CAST(@next_makh AS VARCHAR(8)), 8) END INSERT INTO KHACHHANG VALUES (@makh, @hoten, @sdt, @diachi, @email, @ngaysinh, @gioitinh) commit END; </pre>
sp_UPDATEKhachHang	<pre> CREATE OR ALTER PROC sp_UPDATEKhachHang @MaKH CHAR(10), @HoTen NVARCHAR(100), @SDT CHAR(11), @email CHAR(100), @ngaysinh CHAR(10), @gioitinh CHAR(1), @DiaChi NVARCHAR(200) AS BEGIN UPDATE KHACHHANG SET HoTen = @HoTen, SDT = @SDT, Email = @Email, NgaySinh = @NgaySinh, GioiTinh = @GioiTinh, DiaChi = @DiaChi WHERE MaKH = @MaKH; END </pre>
sp_XoaKhachHang	<pre> CREATE OR ALTER PROC sp_XoaKhachHang @MaKH CHAR(10) AS BEGIN SET NOCOUNT ON; DELETE FROM KHACHHANG WHERE MaKH = @MaKH; </pre>

	<pre>END;</pre>
sp_ThemLoaiSP	<pre>CREATE OR ALTER PROC sp_ThemLoaiSP @tenloaisp NVARCHAR(100) AS BEGIN DECLARE @MaLoaiSP CHAR(10) SELECT TOP 1 @MaLoaiSP = MaLoaiSP FROM LOAISANPHAM WHERE MaLoaiSP LIKE 'LSP%' AND TenLoaiSP IS NULL ORDER BY MaLoaiSP ASC IF @MaLoaiSP IS NULL BEGIN DECLARE @lASt_MaLoaiSP CHAR(10) SELECT TOP 1 @lASt_MaLoaiSP = MaLoaiSP FROM LOAISANPHAM WHERE MaLoaiSP LIKE 'LSP%' ORDER BY MaLoaiSP DESC DECLARE @next_MaLoaiSP INT SET @next_MaLoaiSP = CAST(RIGHT(@lASt_MaLoaiSP, 7) AS INT) + 1 SET @MaLoaiSP = 'LSP' + RIGHT('0000000' + CAST(@next_MaLoaiSP AS VARCHAR(7)), 7) END INSERT INTO LOAISANPHAM VALUES (@maloaisp, @tenloaisp) END</pre>
sp_UPDATELoaiSP	<pre>CREATE OR ALTER PROC sp_UPDATELoaiSP @MaLoaiSP CHAR(10), @TenLoaiSP NVARCHAR(100) AS BEGIN UPDATE LOAISANPHAM SET TenLoaiSP = @TenLoaiSP WHERE MaLoaiSP = @MaLoaiSP; END; GO CREATE OR ALTER PROC sp_TinhTongDoanhThuTheoThang</pre>

	<pre> @thang INT, @nam INT AS BEGIN DECLARE @tongtien DECIMAL SELECT @tongtien = dbo.fn_TinhTongDoanhThuTheoThang(@thang, @nam) SELECT @tongtien AS 'TongDoanhThu' END </pre>
sp_TaoDonHang	<pre> CREATE OR ALTER PROC sp_TaoDonHang @makh CHAR(10), @manv CHAR(10) AS BEGIN DECLARE @ngaydat DATE, @pptt NVARCHAR(50), @ttdh INT, @tongtien DECIMAL SET @pptt = N'Tiền Mặt' SET @ttdh = 0 SET @tongtien = 0 SET @ngaydat = GETDATE() DECLARE @MaDH CHAR(10) Begin transaction SELECT TOP 1 @MADH = MaDH FROM DONHANG with (XLOCK) WHERE MADH LIKE 'DH%' AND MaKH IS NULL ORDER BY MADH ASC IF @MADH IS NULL BEGIN DECLARE @lAst_MADH CHAR(10) SELECT TOP 1 @lAst_MADH = MADH FROM DONHANG WHERE MADH LIKE 'DH%' ORDER BY MADH DESC DECLARE @next_MADH INT SET @next_MADH = CAST(RIGHT(@lAst_MADH, 8) AS INT) + 1 SET @MADH = 'DH' + RIGHT('00000000' + CAST(@next_MADH AS VARCHAR(8)), 8) END INSERT INTO DONHANG VALUES (@MaDH, @ngaydat, @pptt, @ttdh, @tongtien, @makh, @manv) Commit </pre>

	END
sp_MuaHang	<pre> CREATE OR ALTER PROC sp_MuaHang @madh CHAR(10), @mASp CHAR(10), @soluong INT, @size FLOAT, @gioitinh CHAR(1) AS BEGIN BEGIN TRANSACTION; BEGIN TRY DECLARE @soluongconlai INT SELECT @soluongconlai = SoLuong FROM KICHCO_SANPHAM WHERE MASP = @mASp and MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh) IF @soluongconlai >= @soluong BEGIN DECLARE @dongia NUMERIC(18,0), @tensp NVARCHAR(100), @giatien NUMERIC(18,0) SELECT @giatien= GiaBan FROM BIENDONGGIA WHERE MASP = @mASp SET @dongia = (SELECT GiaBan FROM BIENDONGGIA WHERE MASP = @mASp) * @soluong UPDATE DONHANG SET TongTien = TongTien + @dongia WHERE MaDH = @madh UPDATE KICHCO_SANPHAM SET SoLuong = Soluong - @soluong WHERE MASP = @mASp and MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh) DECLARE @mASize CHAR(10) </pre>

	<pre> SELECT @mASize = MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh INSERT INTO CHITIET_DONHANG VALUES (@madh, @mASp, @mASize, @soluong, @giatien) COMMIT; END; ELSE BEGIN RAISERROR(N'Không đủ hàng', 16, 1) ROLLBACK; END; END TRY BEGIN CATCH IF @@TRANCOUNT > 0 ROLLBACK; THROW; END CATCH END; </pre>
sp_UPDATEDonHang	<pre> CREATE OR ALTER PROC sp_UPDATEDonHang @madh CHAR(10), @trangthai INT AS BEGIN UPDATE DONHANG SET TrangThaiDH = @trangthai WHERE @madh = MaDH END </pre>
sp_xoaCTDONHANG	<pre> CREATE OR ALTER PROC sp_xoaCTDONHANG @mact CHAR(10), @mASp CHAR(10), @makc CHAR(10) AS BEGIN DELETE FROM CHITIET_DONHANG WHERE @mact = MaDH and MASP = @mASp and @makc = MaKC END </pre>

<p>sp_UPDATECTDONHANG_SoLuong</p>	<pre> CREATE OR ALTER PROCEDURE sp_UPDATECTDONHANG_SoLuong @madh CHAR(10), @mASp CHAR(10), @soluong INT, @size FLOAT, @gioitinh CHAR(1) AS BEGIN DECLARE @tongtienhientai DECIMAL(18,0) SELECT @tongtienhientai = GiaTien * SoLuong FROM CHITIET_DONHANG DECLARE @soluonghientai INT SELECT @soluonghientai = SoLuong FROM CHITIET_DONHANG WHERE MASP = @mASp AND MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh) DECLARE @soluongconlai INT SELECT @soluongconlai = SoLuong FROM KICHCO_SANPHAM WHERE MASP = @mASp AND MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh) IF @soluongconlai >= @soluong BEGIN DECLARE @dongia NUMERIC(18,0), @giatien NUMERIC(18,0) SELECT @giatien = GiaBan FROM BIENDONGGIA WHERE MASP = @mASp SET @dongia = (SELECT GiaBan FROM BIENDONGGIA WHERE MASP = @mASp) * @soluong UPDATE DONHANG SET TongTien = TongTien - @tongtienhientai WHERE MaDH = @madh UPDATE DONHANG </pre>
-----------------------------------	---

	<pre> SET TongTien = TongTien + @dongia WHERE MaDH = @madh UPDATE KICHCO_SANPHAM SET SoLuong = Soluong + @soluonghientai WHERE MASP = @mASp AND MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh) UPDATE KICHCO_SANPHAM SET SoLuong = Soluong - @soluong WHERE MASP = @mASp AND MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh) DECLARE @mASize CHAR(10) SELECT @mASize = MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh UPDATE CHITIET_DONHANG SET SoLuong = @soluong, GiaTien = @giatien WHERE MaDH = @madh AND MASP = @mASp AND MaKC = @mASize END; ELSE BEGIN RAISERROR(N'Không đủ hàng', 16, 1); END; END; </pre>
sp_TimKiem_KhachHang	<pre> CREATE OR ALTER PROC sp_TimKiem_KhachHang (@tenkh NVARCHAR(100)) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_KHACHHANG(@tenkh) END; </pre>

<p>sp_TimKiem_DONHANG</p>	<pre>CREATE OR ALTER PROC sp_TimKiem_DONHANG (@madh CHAR(10)) AS BEGIN SET NOCOUNT ON; SELECT * FROM dbo.fn_TimKiem_DONHANG(@madh) END;</pre>
<p>sp_NhapHang</p>	<pre>CREATE OR ALTER PROCEDURE sp_NhapHang @mapnh CHAR(10), @mASp CHAR(10), @soluong INT, @size FLOAT, @gioitinh CHAR(1), @tinhtrang INT AS BEGIN DECLARE @dongia NUMERIC(18,0), @giatien NUMERIC(18,0) SELECT @giatien = GiaBan FROM BIENDONGGIA WHERE MASP = @mASp SET @dongia = (SELECT GiaBan FROM BIENDONGGIA WHERE MASP = @mASp) * @soluong UPDATE PHIEUNHAPHANG SET TongTien = TongTien + @dongia WHERE MaPNH = @mapnh UPDATE KICHCO_SANPHAM SET SoLuong = Soluong + @soluong WHERE MASP = @mASp AND MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh) DECLARE @mASize CHAR(10) SELECT @mASize = MaKC FROM KICHCO WHERE LoaiKichCo = @size AND GioiTinh = @gioitinh</pre>

	<pre> INSERT INTO CHITIET_PHIEUNHAP VALUES (@mapnh, @mASp, @mASize, @soluong, @giatien, @tinhtrang) END;</pre>
sp_UPDATECTPhieuNhap_SoLuong	<pre> CREATE OR ALTER PROC sp_UPDATECTPhieuNhap_SoLuong @mapnh CHAR(10), @mASp CHAR(10), @soluong INT, @size FLOAT, @gioitinh CHAR(1), @tinhtrang INT AS BEGIN DECLARE @tongtienhientai DECIMAL(18,0) SELECT @tongtienhientai = DonGiaNhap *SoLuongNhap FROM CHITIET_PHIEUNHAP DECLARE @soluonghientai INT SELECT @soluonghientai = SoLuongNhap FROM CHITIET_PHIEUNHAP WHERE MASP = @mASp and MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh) DECLARE @dongia NUMERIC(18,0) , @giatien NUMERIC(18,0) SELECT @giatien = GiaBan FROM BIENDONGGIA WHERE MASP = @mASp SET @dongia = (SELECT GiaBan FROM BIENDONGGIA WHERE MASP = @mASp) * @soluong UPDATE PHIEUNHAPHANG SET TongTien = TongTien - @tongtienhientai WHERE MaPNH = @mapnh UPDATE PHIEUNHAPHANG SET TongTien = TongTien + @dongia WHERE MaPNH = @mapnh UPDATE KICHCO_SANPHAM</pre>

	<pre> SET SoLuong = Soluong - @soluonghientai WHERE MASP = @mASp and MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh) UPDATE KICHCO_SANPHAM SET SoLuong = Soluong + @soluong WHERE MASP = @mASp and MAKC = (SELECT MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh) DECLARE @mASize CHAR(10) SELECT @mASize = MaKC FROM KICHCO WHERE LoaiKichCo = @size and GioiTinh = @gioitinh UPDATE CHITIET_PHIEUNHAP SET SoLuongNhap = @soluong, DonGiaNhap = @giatien WHERE @mapnh = MaPNH and MASP = @mASp and MaKC = @mASize END </pre>
--	--

5.5. Lệnh tạo hàm (Function)

Tên hàm	Lệnh SQL
fn_TimKiem_TenPhongBan	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_TenPhongBan (@tenpb nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM PHONGBAN WHERE TenPB LIKE '%' + @tenpb + '%'); </pre>
fn_TimKiem_NhanVien_TheoTen	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_NhanVien_TheoTen (@tennv nvarchar(100)) </pre>

	<pre> RETURNS TABLE AS RETURN (SELECT * FROM NHANVIEN WHERE HoTen LIKE '%' + @tennv + '%'); </pre>
fn_Tinh_ThanhTien_SanPham	<pre> CREATE OR ALTER FUNCTION fn_Tinh_ThanhTien_SanPham (@soluong int, @dongia numeric(18,0)) RETURNS decimal(18,2) AS BEGIN DECLARE @thanhtien decimal(18,2) SET @thanhtien = @soluong * @dongia; RETURN @thanhtien; END; </pre>
fn_TimKiem_LoaiSanPham	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_LoaiSanPham (@tenloaisp nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM LOAISANPHAM WHERE TenLoaiSP = @tenloaisp); </pre>
fn_TimKiem_SanPham	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_SanPham (@tensp nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM SANPHAM WHERE TenSP LIKE '%' + @tensp + '%' </pre>

);
fn_TimKiem_NhaCungCap	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_NhaCungCap (@tenncc nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM NHACUNGCAP WHERE TenNCC LIKE '%' + @tenncc + '%'); </pre>
fn_TimKiem_ChucVu	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_ChucVu (@tencv nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM CHUCVU WHERE TenCV LIKE '%' + @tencv + '%'); </pre>
fn_TimKiem_PhieuMuaHang	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_PhieuMuaHang (@mapmh char(10)) RETURNS TABLE AS RETURN (SELECT pm.MaPNH, pm.NgayNhap, pm.MaNV, pm.MaNCC, ct.MaSP, ct.SoLuongNhap , ct.DonGiaNhap, ct.MaKC, ct.TinhTrang FROM PHIEUNHAPHANG pm INNER JOIN CHITIET_PHIEUNHAP ct ON ct.MaPNH = pm.MaPNH WHERE pm.MaPNH LIKE @mapmh); </pre>
fn_TinhTongDoanhThuTheoThang	<pre> CREATE OR ALTER FUNCTION fn_TinhTongDoanhThuTheoThang (</pre>

	<pre> @thang INT, @nam INT) RETURNS decimal AS BEGIN DECLARE @tongtien decimal SELECT @tongtien = SUM(TongTien) FROM DONHANG WHERE MONTH(NgayDatHang) = @thang AND YEAR(NgayDatHang) = @nam RETURN @tongtien END; </pre>
fn_TimKiem_KHACHHANG	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_KHACHHANG (@tenkh nvarchar(100)) RETURNS TABLE AS RETURN (SELECT * FROM KHACHHANG WHERE HoTen LIKE '%' + @tenkh + '%'); </pre>
fn_TimKiem_DONHANG	<pre> CREATE OR ALTER FUNCTION fn_TimKiem_DONHANG (@madh char(10)) RETURNS TABLE AS RETURN (SELECT * FROM DONHANG WHERE MaDH LIKE '%' + @madh + '%'); </pre>

5.6. Lệnh tạo Trigger

Tên Trigger	Lệnh SQL
trg_GiaTri_MacDinh_TinhTrang_SanPham	<pre> CREATE TRIGGER trg_GiaTri_MacDinh_TinhTrang_SanPham ON SANPHAM AFTER INSERT AS BEGIN SET NOCOUNT ON; UPDATE SANPHAM SET TinhTrangSP = 1 WHERE MaSP IN (SELECT MaSP FROM inserted); END; </pre>
trg_KichCo_SoLuong_Zero	<pre> CREATE TRIGGER trg_KichCo_SoLuong_Zero ON KICHCO_SANPHAM AFTER UPDATE AS BEGIN IF (UPDATE(SoLuong)) BEGIN UPDATE SANPHAM SET TinhTrangSP = 0 WHERE MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000001' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000002' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000003' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000004' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000005' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000006' AND SoLuong = 0) AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000007' AND SoLuong = 0) END END </pre>

	<pre> AND MaSP IN (SELECT MaSP FROM KICHCO_SANPHAM WHERE MaKC = 'KC00000008' AND SoLuong = 0) ; END END;</pre>
trg_HuyDon	<pre> CREATE TRIGGER trg_HuyDon ON CHITIET_DONHANG AFTER DELETE AS BEGIN DECLARE @masp VARCHAR(50), @soluong INT, @size char(10), @dongia numeric(18,0), @madh char(10); SELECT @masp = MASP, @soluong = SoLuong, @size = MaKC , @dongia=GiaTien*SoLuong, @madh = MaDH FROM deleted; UPDATE KICHCO_SANPHAM SET Soluong = SoLuong + @soluong WHERE MASP = @masp and MaKC = @size; UPDATE DONHANG SET TongTien = TongTien - @dongia WHERE MaDH = @madh END;</pre>
trg_ThongBaoNhapHang	<pre> CREATE OR ALTER TRIGGER trg_ThongBaoNhapHang ON KICHCO_SANPHAM AFTER UPDATE AS BEGIN IF (UPDATE(SoLuong)) BEGIN IF EXISTS (SELECT 1 FROM inserted WHERE MaSP = inserted.MaSP AND MaKC = inserted.MaKC AND SoLuong < 10) BEGIN Print N'Cần nhập hàng'; END END END;</pre>

5.7. Backup dữ liệu

Công nghệ sử dụng SQL Server Agent:

- Backup toàn bộ dữ liệu mỗi tuần một lần, file nhận được là file “.bak”
- Backup các giao thức được thực hiện trong ngày, mỗi ngày một lần vào 12:00am mỗi ngày.

5.8. Phân quyền sử dụng CSDL**5.8.1. Tạo tài khoản**

```
CREATE LOGIN leduy WITH PASSWORD = '123';  
GO
```

```
CREATE LOGIN tanhiep WITH PASSWORD = '123';  
GO
```

```
CREATE LOGIN hoangkha WITH PASSWORD = '123';  
GO
```

```
CREATE LOGIN thaيدuy WITH PASSWORD = '123';  
GO
```

```
CREATE LOGIN thanhdat WITH PASSWORD = '123';  
GO
```

```
CREATE LOGIN nhathuy WITH PASSWORD = '123';  
GO
```

5.8.2. Tạo role**5.8.3. Tạo người dùng**

```
CREATE USER u_Admin1 FOR LOGIN leduy;  
GO
```

```
CREATE USER u_Admin2 FOR LOGIN hoangkha;  
GO
```

--Quản lý

```
CREATE USER u_Manager1 FOR LOGIN tanhiep;  
GO
```

```
CREATE USER u_Manager2 FOR LOGIN thaiduy;  
GO
```

--Nhân viên

```
CREATE USER u_Employee2 FOR LOGIN thanhdat;  
GO
```

```
CREATE USER u_Employee3 FOR LOGIN nhathuy;  
GO
```

5.8.4. Gán quyền

-----Quản trị viên-----

--Gán toàn quyền cho role QUANTRIVIEN

--Dùng role db_owner có sẵn trong SQL SERVER

```
ALTER ROLE db_owner ADD MEMBER r_QUANTRIVIEN;  
GO
```

-----Quản lý cửa hàng-----

--Bảng sản phẩm

```
GRANT SELECT, INSERT, UPDATE, DELETE ON SANPHAM TO  
r_QUANLY_CUAHANG;  
GO
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON KICHCO TO  
r_QUANLY_CUAHANG;  
GO
```

```
GRANT SELECT ON KICHCO_SANPHAM TO r_QUANLY_CUAHANG;  
GO
```

--Bảng nhà cung cấp

```
GRANT SELECT, INSERT, UPDATE ON NHACUNGCAP TO  
r_QUANLY_CUAHANG;  
GO
```

--Bảng nhân viên

```
GRANT SELECT, INSERT, UPDATE, DELETE ON NHANVIEN TO  
r_QUANLY_CUAHANG;  
GO
```

--Bảng khách hàng

```
GRANT SELECT, INSERT, UPDATE, DELETE ON KHACHHANG TO  
r_QUANLY_CUAHANG;  
GO
```

--Bảng Biến động giá

```
GRANT SELECT, INSERT, UPDATE, DELETE ON BIENDONGGIA TO  
r_QUANLY_CUAHANG;  
GO
```

--Bảng đơn hàng

```
GRANT SELECT, INSERT ON DONHANG TO r_QUANLY_CUAHANG;  
  
GO
```

```
GRANT UPDATE(TrangThaiDH) ON DONHANG TO r_QUANLY_CUAHANG  
GO
```

--Bảng chi tiết đơn hàng

```
GRANT SELECT ON CHITIET_DONHANG TO r_QUANLY_CUAHANG;  
GO
```

```
GRANT UPDATE(SoLuong) ON CHITIET_DONHANG TO  
r_QUANLY_CUAHANG;
```

GO

--Bảng phiếu nhập hàng

```
GRANT SELECT, INSERT, UPDATE, DELETE ON PHIEUNHAP TO  
r_QUANLY_CUAHANG;
```

```
GO
```

```
GRANT SELECT, INSERT, UPDATE(SoLuongNhap, DonGiaNhap, TinhTrang) ON  
CHITIET_PHIEUNHAP TO r_QUANLY_CUAHANG;
```

```
GO
```

```
-----Nhân viên bán hàng-----
```

```
--Bảng sản phẩm
```

```
GRANT SELECT, INSERT, UPDATE ON SANPHAM TO  
r_NHANVIEN_BANHANG;
```

```
GO
```

```
GRANT SELECT, INSERT, UPDATE ON KICHCO TO  
r_NHANVIEN_BANHANG;
```

```
GO
```

```
GRANT SELECT ON KICHCO_SANPHAM TO r_NHANVIEN_BANHANG;
```

```
GO
```

```
--Bảng khách hàng
```

```
GRANT SELECT, INSERT ON KHACHHANG TO r_NHANVIEN_BANHANG;
```

```
GO
```

```
--Bảng đơn hàng
```

```
GRANT SELECT, INSERT, UPDATE(TrangThaiDH) ON DONHANG TO  
r_NHANVIEN_BANHANG;
```

```
GO
```

```
-----CẤP QUYỀN CHO NGƯỜI DÙNG-----
```

```
--Quyền quản trị
```

```
ALTER ROLE r_QUANTRIVIEN ADD MEMBER u_Admin1;
```

```
GO
```

```
ALTER ROLE r_QUANTRIVIEN ADD MEMBER u_Admin2;
```

```
GO
```

--Quyền quản lý

```
ALTER ROLE r_QUANLY_CUAHANG ADD MEMBER u_Manager1;  
GO
```

```
ALTER ROLE r_QUANLY_CUAHANG ADD MEMBER u_Manager2;  
GO
```

--Quyền nhân viên

```
ALTER ROLE r_NHANVIEN_BANHANG ADD MEMBER u_Employee2;  
GO
```

```
ALTER ROLE r_NHANVIEN_BANHANG ADD MEMBER u_Employee3;  
GO
```


PHỤ LỤC

MSSV	Họ và tên	Nội dung thực hiện	Trưởng nhóm (Đánh dấu X)
20DH110518	Trần Lê Duy	Thiết kế Database. Table, ràng buộc. Stored Procedure, Function, Trigger, View. Backup Viết Báo cáo	X
20DH111987	Huỳnh Tấn Hiệp	Thiết kế Database. Dữ liệu. Stored Procedure, Function, Trigger, View. Giao tác, xử lý đồng thời.	
20DH112017	Nguyễn Hoàng Kha	Thiết kế Database. Table, ràng buộc. Stored Procedure, Function, Trigger, View. Phân Quyền.	

TÀI LIỆU THAM KHẢO

- Slide bài giảng