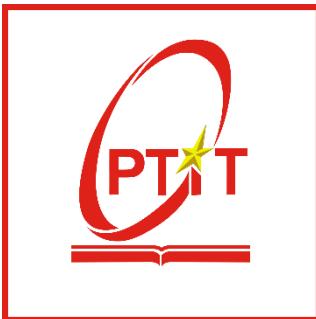


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MÍNH



## BÁO CÁO ĐỒ ÁN

**ĐỀ TÀI: XÂY DỰNG CƠ SỞ DỮ LIỆU  
QUẢN LÝ CỦA HÀNG BÁN GIÀY DÉP**

Môn học: Thực tập cơ sở

**Giảng viên: Nguyễn Thị Bích Nguyên**

**Thành viên:**

**Nguyễn Thị Huyền My**

**N21DCCN147**

**Lê Ngọc Luân**

**N21DCCN144**

**Nguyễn Đình Luật**

**N21DCCN145**

**Đặng Xuân Nam**

**N21DCCN148**

**Hồ Chí Minh, 2024**

# MỤC LỤC

<b>A. Khảo sát về cửa hàng.....</b>	<b>4</b>
I. Về nhân sự (quản lý và nhân viên của TEAM-9).....	4
II. Về tài nguyên cửa hàng.....	4
III. Về những sự kiện xoay quanh sản phẩm .....	4
IV. Về trải nghiệm của khách hàng khi đến với TEAM-9.....	5
V. Về giao hàng, thanh toán và hậu phát sinh .....	5
<b>B. Đối tượng sử dụng và tính năng .....</b>	<b>6</b>
I. Quản lý .....	6
II. Nhân viên .....	7
III. Khách hàng .....	7
<b>C. Xoay quanh về cấu trúc, nguyên lý hoạt động của cơ sở dữ liệu.....</b>	<b>8</b>
I. XÁC ĐỊNH THỰC THẾ CƠ SỞ DỮ LIỆU .....	8
II. MÔ HÌNH QUAN HỆ ĐẠT DẠNG CHUẨN 3 .....	9
III. DIAGRAM .....	10
1. Về nhân sự và tài nguyên của cửa hàng.....	10
2. Về khách hàng .....	10
3. Về hoạt động của cửa hàng và các vấn đề phát sinh.....	11
IV. TỪ ĐIỂN DỮ LIỆU QUAN HỆ .....	12
1. Bảng Nhân viên.....	12
2. Bảng Quá trình Quản lý .....	12
3. Bảng Tài khoản Nhân viên.....	12
4. Bảng Hàng.....	13
5. Bảng Loại .....	13
6. Bảng Sản phẩm .....	13
7. Bảng Thay đổi giá .....	14
8. Bảng Khuyến mãi.....	14
9. Bảng Chi tiết Khuyến mãi .....	14
10. Bảng Khách hàng .....	15
11. Bảng Tài khoản Khách hàng.....	15
12. Bảng Giỏ hàng .....	15
13. Bảng Đơn đặt hàng .....	16
14. Bảng Chi tiết Đơn đặt hàng.....	16
15. Bảng Giao hàng .....	16
16. Bảng Hóa đơn .....	17
17. Bảng Đánh giá .....	17
18. Bảng Đổi trả.....	17
19. Bảng Phiếu bảo hành .....	18
20. Bảng Nhận bảo hành.....	18
21. Bảng Trả bảo hành .....	18

<b>V. MỘT SỐ RÀNG BUỘC TOÀN VẸN.....</b>	<b>19</b>
1. Ràng buộc toàn vẹn về miền giá trị .....	19
2. Ràng buộc toàn vẹn liên thuộc tính .....	20
3. Ràng buộc toàn vẹn liên thuộc tính liên quan hệ.....	20
<b>VI. TRIGGER VÀ STORED PROCEDURE.....</b>	<b>21</b>
1. Trigger .....	21
2. Stored Procedure.....	26
3. View.....	49
<b>VII. PHÂN QUYỀN.....</b>	<b>50</b>
1. Quản lý:.....	50
2. Nhân viên: .....	50
3. Khách hàng: .....	51
<b>D. Backup .....</b>	<b>53</b>
I. Backup.....	53
II. Restore .....	56
III. Lập lịch Backup.....	57

# CỦA HÀNG GIÀY DÉP TEAM-9

**Mục đích của cơ sở dữ liệu quản lý cửa hàng bán giày dép:** quản lý các hoạt động mua bán của cửa hàng. Sau đó thống kê doanh thu và các đơn hàng được bán ra của cửa hàng để đưa ra những chiến lược kinh doanh hợp lý.

## A. Khảo sát về cửa hàng

### I. Về nhân sự (quản lý và nhân viên của TEAM-9)

- Cửa hàng sẽ có một đội ngũ nhân viên, bao gồm cả quản lý và các nhân viên khác, để đảm bảo hoạt động kinh doanh diễn ra một cách suôn sẻ và hiệu quả nhất. Trong cửa hàng giày dép, luôn chỉ có một quản lý duy nhất trong một thời điểm, người này cũng chính là một nhân viên của cửa hàng. Quản lý có thể được thay đổi theo thời gian tùy theo nhu cầu của cửa hàng trong suốt quá trình hoạt động.
- Thông tin của nhân viên sẽ bao gồm họ tên, giới tính, ngày sinh, số căn cước công dân (CCCD), số điện thoại và địa chỉ email. Quản lý sẽ có thông tin về thời gian bắt đầu làm việc và nhiệm kỳ quản lý (tính theo tháng). Trong trường hợp quản lý phải nghỉ việc đột ngột hoặc vì bất kỳ lý do gì không thể tiếp tục vị trí quản lý, một nhân viên khác sẽ được chọn để thay thế. Quản lý cũng có trách nhiệm tạo tài khoản cho mỗi nhân viên mới tham gia làm việc tại cửa hàng. Mỗi nhân viên chỉ được phép có một tài khoản duy nhất, bao gồm tên đăng nhập (mã nhân viên sẽ được dùng làm tên đăng nhập) và mật khẩu. Thông tin cá nhân của mỗi nhân viên chỉ có thể được thay đổi bởi quản lý khi cần thiết.

### II. Về tài nguyên cửa hàng

- Trong cửa hàng, có một loạt sản phẩm đa dạng. Mỗi sản phẩm sẽ được phân loại theo hãng sản xuất như Adidas, Converse,... và các loại như Sneaker, Sandal,... Thông tin về sản phẩm bao gồm tên, màu sắc, kích thước, chất liệu (như da, vải polyester,...), xuất xứ, giá bán, thời gian bảo hành và số lượng tồn kho. Các thông tin sản phẩm luôn được mô tả cụ thể để khách hàng yên tâm về chất lượng sản phẩm của TEAM-9.

### III. Về những sự kiện xoay quanh sản phẩm

- Giá của các sản phẩm có thể biến động tùy thuộc vào nhu cầu của thị trường hoặc chiến lược kinh doanh của cửa hàng. Có một nhân viên được ủy quyền chịu trách nhiệm cập nhật giá cho các sản phẩm và ghi chính xác thời gian bắt đầu áp dụng giá mới, bao gồm cả ngày, tháng và năm, kèm theo giờ và phút.
- Trong quá trình hoạt động, cửa hàng sẽ tổ chức nhiều đợt khuyến mãi khác nhau, mỗi đợt có thời gian diễn ra riêng. Trong mỗi đợt khuyến mãi, nhiều sản phẩm sẽ được giảm giá. Việc tổ chức mỗi đợt khuyến mãi sẽ do một nhân

viên chịu trách nhiệm, bao gồm việc nhập phần trăm giảm giá cho từng sản phẩm. Mỗi đợt khuyến mãi sẽ có thời gian bắt đầu và kết thúc cụ thể, cùng với lí do tổ chức (như kỷ niệm, lễ Tết, và các dịp đặc biệt khác).

#### IV. Về trải nghiệm của khách hàng khi đến với TEAM-9

- Khách hàng muốn mua hàng trên nền tảng thương mại điện tử phải bắt đầu bằng việc tạo tài khoản, trừ khi họ đã có sẵn. Quá trình đăng ký tài khoản mới bao gồm việc cung cấp các thông tin như họ tên, ngày sinh, số điện thoại, và email (nếu có). Thông tin đăng nhập sẽ bao gồm tên đăng nhập và mật khẩu. Mỗi khách hàng có thể tạo nhiều tài khoản khác nhau, nhưng một tài khoản chỉ có thể thuộc về duy nhất một khách hàng. Khách hàng cũng có thể thay đổi thông tin cá nhân như họ tên, email, và mật khẩu đăng nhập khi cần thiết.
- Khách hàng đến cửa hàng mua trực tiếp sẽ được nhân viên hỗ trợ trong việc tạo tài khoản, tương tự như khi mua hàng trực tuyến. Nhân viên sẽ thu thập thông tin như họ tên và số điện thoại trực tiếp từ khách hàng để tạo tài khoản.
- Sau khi đăng nhập vào tài khoản, khách hàng có thể xem các sản phẩm hiện đang được kinh doanh tại cửa hàng. Mỗi tài khoản được gắn với một giỏ hàng riêng, trong đó khách hàng có thể thêm các sản phẩm yêu thích và xem thông tin ngắn gọn về mỗi sản phẩm cùng với số lượng đã chọn. Từ đó, khách hàng có thể lựa chọn sản phẩm và tiến hành đặt hàng.
- Mỗi đơn đặt hàng có thể chứa một hoặc nhiều sản phẩm khác nhau. Trong mỗi đơn đặt hàng, sẽ được liệt kê tên các sản phẩm được đặt cùng với số lượng, giá tiền, và thông tin chi tiết về người đặt hàng. Trạng thái của đơn hàng sẽ được ghi rõ, có thể là "**đang duyệt**", "**đang chuẩn bị hàng**", "**đang giao hàng**", "**đã hoàn thành**", hoặc "**đã hủy**". Hình thức thanh toán cũng được xác định bởi người mua hàng, có thể là chuyển khoản hoặc thanh toán bằng tiền mặt. Hình thức mua hàng có 3 dạng: online, offline (giao hàng) và offline (không giao hàng).
- Mỗi đơn hàng chỉ thuộc về một tài khoản khách hàng duy nhất. Một tài khoản khách hàng có thể đặt nhiều đơn hàng khác nhau.

**Lưu ý:** Khách hàng chỉ có thể hủy đơn hàng khi đơn hàng đó chưa bắt đầu trong quá trình vận chuyển.

#### V. Về giao hàng, thanh toán và hậu phát sinh

- Sau khi một đơn hàng được đặt, nó sẽ được chuyển đến một nhân viên để tiến hành duyệt. Nhân viên này sẽ xem xét các chi tiết của đơn hàng, bao gồm sản phẩm được đặt, số lượng, và thông tin của người mua. Sau khi đơn hàng được duyệt, nhân viên sẽ bắt đầu chuẩn bị các sản phẩm trong đơn hàng để tiến hành đóng gói và lập hóa đơn. Trong hóa đơn sẽ bao gồm thông tin chi tiết về người mua và tổng số tiền của các sản phẩm đã mua. Nếu có khuyến mãi, số tiền đã được trừ sẽ được hiển thị rõ ràng trong hóa đơn. Thời gian xuất hóa đơn cũng sẽ được ghi lại để tiện theo dõi. Mỗi sản phẩm sẽ

được tính thời gian bảo hành bắt đầu từ thời điểm hóa đơn mua hàng được lập.

- Sau khi xuất hóa đơn thì sẽ có 1 nhân viên khác phụ trách việc giao hàng cho khách. Thông tin giao hàng sẽ bao gồm tên nhân viên giao, thông tin đơn hàng và thông tin giao hàng đã được khách hàng cung cấp từ trước. Trong trường hợp khách hàng đặt hàng cho người khác, họ sẽ cung cấp thông tin về người nhận hàng như tên, số điện thoại và địa chỉ giao hàng của người đó. Nếu không, thông tin giao hàng sẽ mặc định là thông tin của tài khoản khách hàng hiện tại. Thời gian bắt đầu và thời gian hoàn thành việc giao hàng sẽ được ghi nhận lại. (để phòng có sự cố xảy ra hay cần truy xuất lại thông tin,...)
- Chỉ tài khoản nào đã mua sản phẩm đó thì mới được đánh giá sản phẩm. Mỗi sản phẩm chỉ được một tài khoản đánh giá một lần duy nhất. Phần đánh giá bao gồm việc đánh số sao từ 1 đến 5 và bổ sung một nội dung chi tiết. Điều này giúp người dùng diễn đạt ý kiến của họ về sản phẩm một cách cụ thể và đầy đủ.
- Khách hàng chỉ được quyền đổi trả sản phẩm trong trường hợp sản phẩm đã được giao không vừa size hoặc không đúng mẫu mã đã đặt. Điều kiện tiên quyết để đổi trả là khách hàng cần quay video khi mở gói hàng và sản phẩm được giao không quá 7 ngày kể từ lúc nhận hàng. Khách hàng có thể chọn sử dụng dịch vụ giao hàng để đổi trả hoặc đưa sản phẩm trực tiếp đến cửa hàng. Một nhân viên sẽ được ủy quyền lập phiếu đổi trả, bao gồm thông tin chi tiết về khách hàng, lý do đổi trả, và thời gian thực hiện việc đổi trả. Điều này giúp quy trình đổi trả diễn ra một cách trơn tru và hiệu quả.
- Việc bảo hành sản phẩm chỉ áp dụng trong trường hợp sản phẩm bị lỗi do nhà sản xuất, như việc đế giày bung ra hoặc dây giày bị đứt,... Không chấp nhận bảo hành đối với các tình huống hư hỏng phát sinh do việc sử dụng không đúng cách hoặc do các yếu tố tự nhiên trong quá trình sử dụng.
- Khi nhận sản phẩm để bảo hành, một nhân viên sẽ đảm nhận trách nhiệm lập phiếu nhận, ghi lại các thông tin về khách hàng, lý do bảo hành, và thời gian nhận sản phẩm. Khi trả sản phẩm sau khi bảo hành, cũng tương tự, sẽ có một nhân viên lập phiếu trả, dựa trên thông tin từ phiếu nhận, mô tả tình trạng sản phẩm sau khi đã được bảo hành, và thời gian trả sản phẩm. Điều này giúp quy trình bảo hành diễn ra một cách dễ dàng và minh bạch.

## B. Đối tượng sử dụng và tính năng

### I. Quản lý

- Xem thông tin của các sản phẩm trong cửa hàng.
- Có quyền thêm tài khoản khi có nhân viên mới, xóa tài khoản nhân viên khi người đó nghỉ việc.
- Chính sửa thông tin cá nhân của nhân viên.

- Xem các đơn hàng đã hoặc đang được xử lý.
- Xem doanh thu (lợi nhuận) theo từng giai đoạn. (ngày, tháng, quý,...)

## **II. Nhân viên**

- Đăng nhập vào phần mềm.
- Thay đổi mật khẩu của tài khoản.
- Thêm, xóa sản phẩm trong cửa hàng. Chính sửa thông tin của sản phẩm.
- Cập nhật các đợt khuyến mãi của cửa hàng cho từng sản phẩm cụ thể.
- Lập phiếu đặt hàng cho các đơn hàng online. Duyệt các đơn hàng và xuất hóa đơn.
- Lập đơn giao hàng.
- Xem lại lịch sử mua hàng.
- Lập phiếu đổi trả hàng.
- Lập phiếu nhận, trả bảo hành.
- Thống kê lợi nhuận theo từng mốc thời gian cụ thể. (ngày, tháng, năm, quý,...)

## **III. Khách hàng**

- Đăng ký tài khoản.
- Quản lý tài khoản của mình như chỉnh sửa thông tin cá nhân, đổi mật khẩu,...
- Xem các sản phẩm và thông tin sản phẩm trong cửa hàng.
- Đặt mua hàng.
- Quản lý giỏ hàng như thêm, xóa, sửa sản phẩm.
- Theo dõi đơn hàng (chờ duyệt, đang chuẩn bị hàng, đang giao hàng, đã hoàn tất đơn hàng hay đã hủy)
- Đổi trả hàng và bảo hành (có một vài điều kiện nhất định khi đổi trả hàng và bảo hành).

## C. Xoay quanh về cấu trúc, nguyên lý hoạt động của cơ sở dữ liệu

### I. XÁC ĐỊNH THỰC THẾ CƠ SỞ DỮ LIỆU

1. **QTQuanLy**(MaQL, Ngay\_BD, Ngay\_KT)
2. **NhanVien**(MaNV, Ho, Ten, GioiTinh, NgaySinh, CCCD, SDT, Email, HinhAnh)
3. **Loai**(MaLoai, TenLoai)
4. **Hang**(MaHang, TenHang)
5. **SanPham**(MaSP, TenSP, MauSac, Size, ChatLieu, XuatXu, Gia, HinhAnh, TG\_BaoHanh, SLTon)
6. **PhieuBH**(MaBH, Ngay\_BD, Ngay\_KT)
7. **KhachHang**(MaKH, Ho, Ten, SDT, Email)
8. **DatHang**(MaDDH, TrangThai, TG\_Dat)
9. **GiaoHang**(MaGH, ThoiGian\_BD, ThoiGian\_KT, DiaChiGH)
10. **KhuyenMai**(MaDotKM, LyDo, Ngay\_BD, Ngay\_KT)
11. **DoiTra**(MaDT, LyDo)
12. **HoaDon**(MaHD, TongTien)

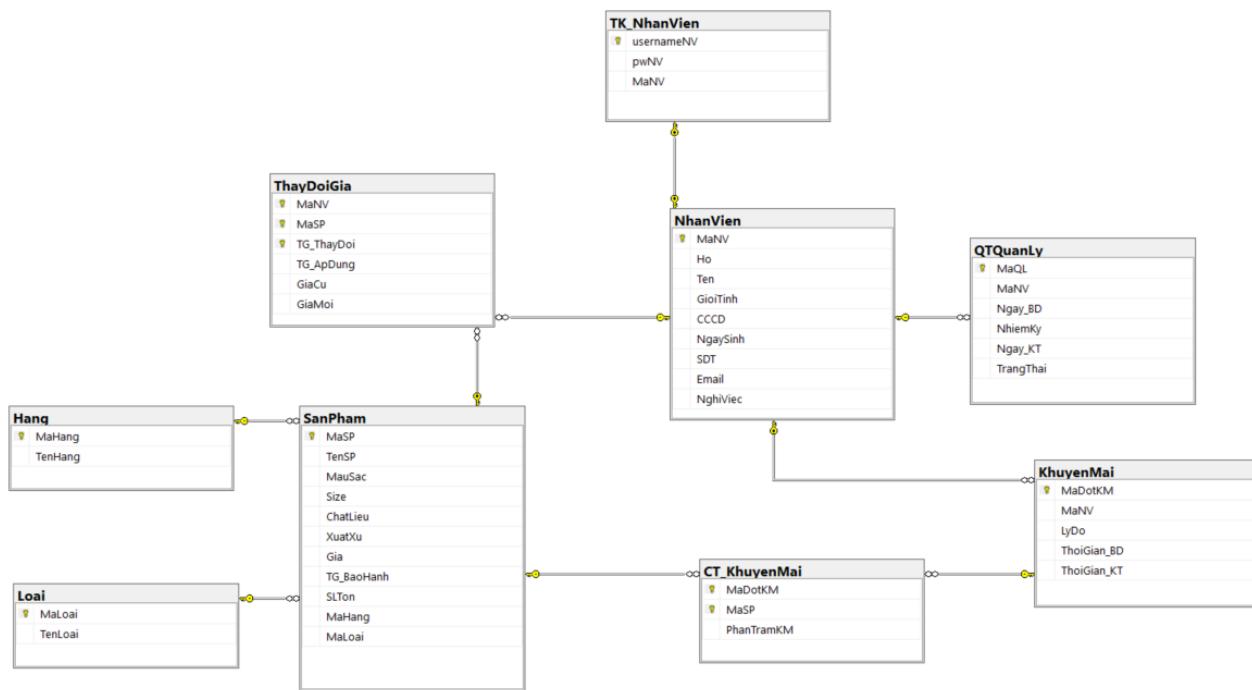
## II. MÔ HÌNH QUAN HỆ ĐẠT DẶNG CHUẨN 3

(*Chú thích : ABC* : Tên bảng ; *Abc* : Khóa chính; *Abc* : Khóa ngoại )

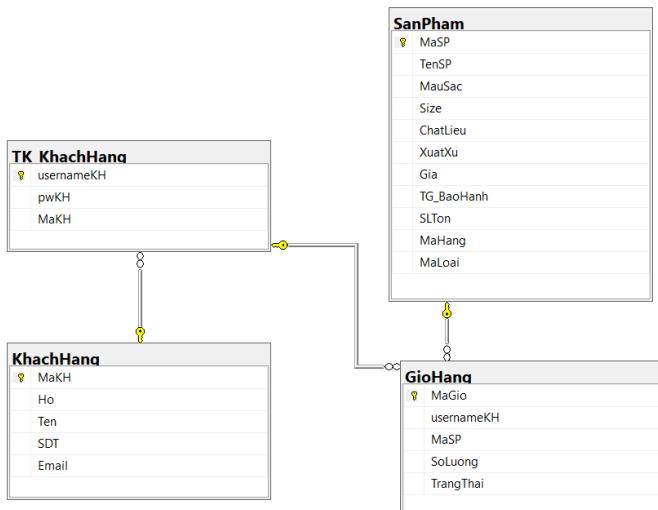
- **NhanVien**(MaNV, Ho, Ten, GioiTinh, CCCD, NgaySinh, SDT, Email, NghiViec)
- **QTQuanLy**(MaQL, **MaNV**, Ngay\_BD, NghiemKy, Ngay\_KT, TrangThai)
- **TK\_NhanVien**(usernameNV, pwNV, **MaNV**)
- **Hang**(MaHang, TenHang)
- **Loai**(MaLoai, TenLoai)
- **SanPham**(MaSP, TenSP, MauSac, Size, ChatLieu, XuatXu, Gia, TG\_BaoHanh, SLTon, **MaHang**, **MaLoai**)
- **ThayDoiGia**(**MaNV**, **MaSP**, **TG\_ThayDoi**, TG\_ApDung, GiaCu, GiaMoi)
- **KhuyenMai**(MaDotKM, **MaNV**, LyDo, ThoiGian\_BD, ThoiGian\_KT)
- **CT\_KhuyenMai**(MaDotKM, **MaSP**, PhanTramKM)
- **KhachHang**(MaKH, Ho, Ten, *SDT*, *Email*)
- **TK\_KhachHang**(usernameKH, pwKH, **MaKH**)
- **GioHang**(MaGio, usernameKH, **MaSP**, SoLuong, TrangThai)
- **DonDH**(MaDDH, TG\_Dat, TrangThai, HinThuc, ThanhToan, usernameKH , **MaNV**)
- **CT\_DonDH**(MaCTDDH, **MaDDH**, **MaGio**, SoLuong, **MaHD**)
- **GiaoHang**(MaGH, **MaNV**, **MaDDH**, ThoiGian\_BD, ThoiGian\_KT, TenNguoiNhan, SDT, DiaChiGH)
- **HoaDon**(MaHD, TongTien, ThoiGian, **MaNV**)
- **DanhGia**(MaCTDDH, Sao, NoiDung)
- **DoiTra**(MaDT, **MaNV**, MaCTDDH, LyDo, TG\_DoiTra)
- **PhieuBH**(MaBH, MaCTDDH, **MaNV**, Ngay\_BD, Ngay\_KT)
- **NhanBH**(MaNhanBH, **MaBH**, **MaNV**, LyDoBH, TG\_BaoHanh)
- **TraBH**(MaTraBH, **MaNhanBH**, **MaNV**, TinhTrangSauBH, TG\_Tra)

### III. DIAGRAM

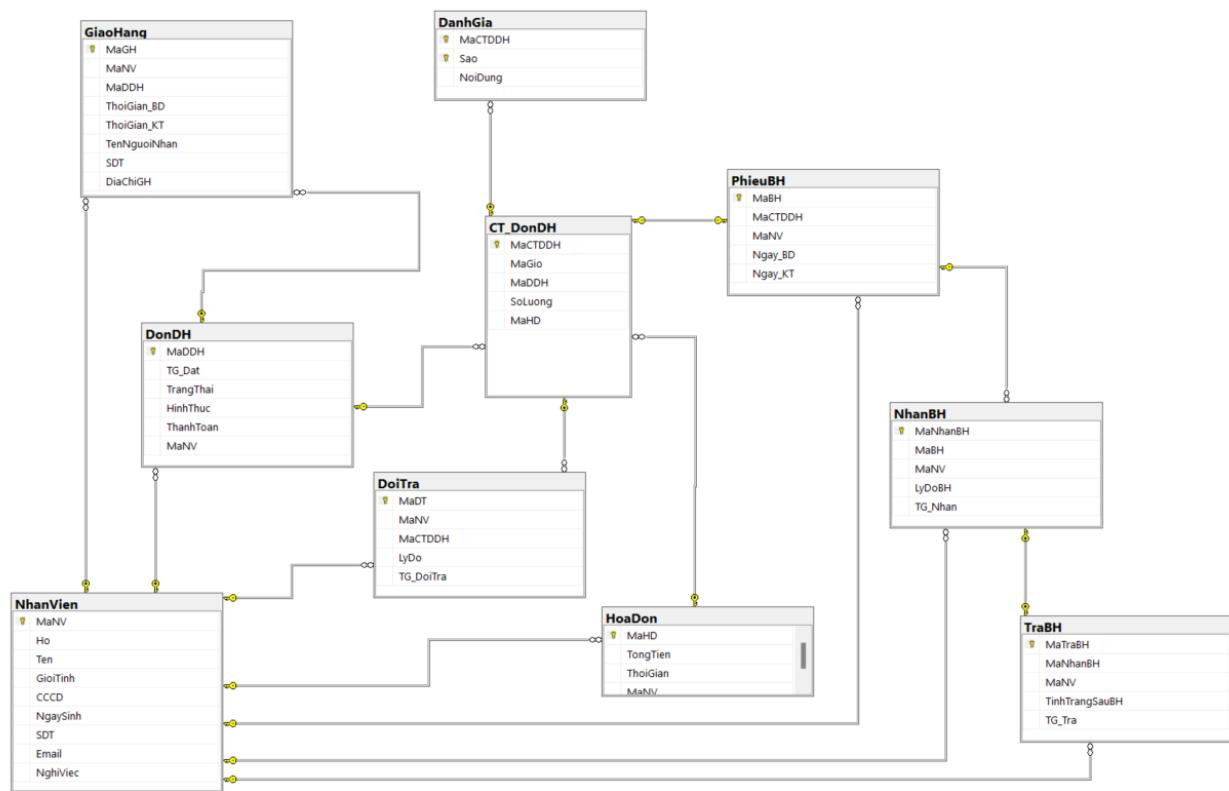
#### 1. Về nhân sự và tài nguyên của cửa hàng



#### 2. Về khách hàng



### 3. Về hoạt động của cửa hàng và các vấn đề phát sinh



## IV. TÙ ĐIỂN DỮ LIỆU QUAN HỆ

### 1. Bảng Nhân viên

Field Name	Type	Length	Constraint
MaNV	int		PK
Ho	nvarchar	100	Not Null
Ten	nvarchar	50	Not Null
GioiTinh	nvarchar	3	Not Null
CCCD	varchar	12	UK, Not Null
NgaySinh	date		Not Null
SDT	varchar	10	UK, Not Null
Email	varchar	50	UK, Not Null
NghiViec	bit		Not Null

Khóa duy nhất: CCCD, SDT và Email

### 2. Bảng Quá trình Quản lý

Field Name	Type	Length	Constraint
MaQL	int		PK
MaNV	varchar	10	FK. Not Null
Ngay_BD	date		Not Null
NhiemKy	tinyint		Not Null
Ngay_KT	date		
TrangThai	bit		Not Null

Khóa duy nhất: MaNV + Ngay\_BD

### 3. Bảng Tài khoản Nhân viên

Field Name	Type	Length	Constraint

usernameNV	varchar	10	PK
pwNV	varchar	50	Not Null
MaNV	varchar	10	FK, Not Null

Khóa duy nhất: MaNV

#### 4. Bảng Hàng

Field Name	Type	Length	Constraint
MaHang	varchar	10	PK
TenHang	nvarchar	50	UK, Not Null

Khóa duy nhất: TenHang

#### 5. Bảng Loại

Field Name	Type	Length	Constraint
MaLoai	varchar	10	PK
TenLoai	nvarchar	50	UK, Not Null

Khóa duy nhất: TenLoai

#### 6. Bảng Sản phẩm

Field Name	Type	Length	Constraint
MaSP	Int		PK
TenSP	nvarchar	50	Not Null
MauSac	nvarchar	10	Not Null
Size	tinyint		Not Null
ChatLieu	nvarchar	30	Not Null
XuatXu	nvarchar	30	Not Null
Gia	money		Not Null
TG_BaoHanh	tinyint		Not Null

SLTon	smallint		Not Null
MaHang	varchar	10	FK, Not Null
MaLoai	varchar	10	FK, Not Null

Khóa duy nhất: TenSP + MauSac + Size

## 7. Bảng Thay đổi giá

Field Name	Type	Length	Constraint
MaNV	int		PK, FK
MaSP	int		PK, FK
TG_ThayDoi	datetime		PK
TG_ApDung	datetime		Not Null
GiaCu	money		Not Null
GiaMoi	money		Not Null

## 8. Bảng Khuyến mãi

Field Name	Type	Length	Constraint
MaDotKM	int		PK
MaNV	int		FK, Not Null
LyDo	nvarchar	100	Not Null
ThoiGian_BD	datetime		Not Null
ThoiGian_KT	datetime		Not Null

## 9. Bảng Chi tiết Khuyến mãi

Field Name	Type	Length	Constraint
MaDotKM	int		PK, FK

MaSP	int		PK, FK
PhanTramKM	float		Not Null

## 10. Bảng Khách hàng

Field Name	Type	Length	Constraint
MaKH	int		PK
Ho	nvarchar	100	Not Null
Ten	nvarchar	50	Not Null
SDT	varchar	10	UK, Not Null
Email	varchar	50	UK, Not Null

Khóa duy nhất: SDT, Email

## 11. Bảng Tài khoản Khách hàng

Field Name	Type	Length	Constraint
usernameKH	varchar	50	PK
pwKH	varchar	100	Not Null
MaKH	int		FK, Not Null

## 12. Bảng Giỏ hàng

Field Name	Type	Length	Constraint
MaGio	int		PK
usernameKH	varchar	50	FK
MaSP	Int		FK, Not Null
SoLuong	tinyint		Not Null
TrangThai	bit		Not Null

### 13. Bảng Đơn đặt hàng

Field Name	Type	Length	Constraint
MaDDH	int		PK
TG_Dat	datetime		Not Null
TrangThai	nvarchar	20	Not Null
HinhThuc	nvarchar	20	Not Null
ThanhToan	nvarchar	20	Not Null
MaNV	int		FK

### 14. Bảng Chi tiết Đơn đặt hàng

Field Name	Type	Length	Constraint
MaCTDDH	int		PK
MaGio	int		FK, Not Null
MaDDH	int		FK. Not Null
SoLuong	tinyint		Not Null
MaHD	varchar	10	FK

### 15. Bảng Giao hàng

Field Name	Type	Length	Constraint
MaGH	int		PK
MaNV	Int		FK
MaDDH	int		FK
ThoiGian_BD	datetime		Not Null
ThoiGian_KT	datetime		

TenNguoiNhan	nvarchar	50	
SDT	varchar	10	
DiaChiGH	nvarchar	100	Not Null

## 16. Bảng Hóa đơn

Field Name	Type	Length	Constraint
MaHD	varchar	10	PK
TongTien	money		
ThoiGian	datetime		
MaNV	int		FK

## 17. Bảng Đánh giá

Field Name	Type	Length	Constraint
MaCTDDH	int		PK, FK
Sao	tinyint		PK
NoiDung	nvarchar	100	Not Null

## 18. Bảng Đổi trả

Field Name	Type	Length	Constraint
MaDT	int		PK
MaNV	int		FK, Not Null
MaCTDDH	int		FK, Not Null
LyDo	nvarchar	100	Not Null
TG_DoiTra	datetime		Not Null

## 19. Bảng Phiếu bảo hành

Field Name	Type	Length	Constraint
MaBH	Int		PK
MaCTDDH	int		FK, Not Null
MaNV	int		FK, Not Null
Ngay_BD	date		Not Null
Ngay_KT	date		Null

Khóa duy nhất: MaCTDDH

## 20. Bảng Nhận bảo hành

Field Name	Type	Length	Constraint
MaNhanBH	int		PK
MaBH	int		FK, Not Null
MaNV	int		FK, Not Null
LyDoBH	nvarchar	200	Not Null
TG_Nhan	datetime		Not Null

## 21. Bảng Trả bảo hành

Field Name	Type	Length	Constraint
MaTraBH	int		PK
MaNhanBH	int		FK, Not Null
MaNV	int		FK, Not Null
TinhTrangSauBH	nvarchar	100	Not Null
TG_Tra	datetime		Not Null

Khóa duy nhất: MaNhanBH

## V. MỘT SỐ RÀNG BUỘC TOÀN VẸN

### 1. Ràng buộc toàn vẹn về miền giá trị

- Nhân viên chỉ có 2 giới tính Nam và Nữ.  
 $\forall nv \in NhanVien: nv.GioiTinh = N'Nam' OR nv.GioiTinh = N'Nữ'$
- Ngày sinh của nhân viên phải nhỏ hơn thời gian hiện tại.  
 $\forall nv \in NhanVien: nv.NgaySinh < CONVERT(DATE, GETDATE())$
- Size sản phẩm nằm trong khoảng từ 15 đến 45.  
 $\forall sp \in SanPham: sp.size BETWEEN 15 AND 45$
- Giá sản phẩm phải lớn hơn 0.  
 $\forall sp \in SanPham: sp.Gia > 0$
- Thời gian bảo hành của mỗi sản phẩm phải lớn hơn hoặc bằng 0.  
 $\forall sp \in SanPham: sp.TG_BaoHanh >= 0$
- Số lượng tồn của sản phẩm phải lớn hơn hoặc bằng 0.  
 $\forall sp \in SanPham: sp.SLTon >= 0$
- Thời gian bắt đầu của đợt khuyến mãi sẽ lớn hơn hoặc bằng thời gian hiện tại.  
 $\forall km \in KhuyenMai: km.ThoiGian_BD >= GETDATE()$
- Phần trăm khuyến mãi phải lớn hơn 0 và nhỏ hơn 100.  
 $\forall km \in KhuyenMai: km.PhanTramKM > 0 AND km.PhanTramKM < 100$
- Số lượng sản phẩm trong giỏ hàng phải lớn hơn hoặc bằng 0.  
 $\forall gh \in GioHang: gh.SoLuong >= 0$
- Thời gian đặt hàng phải lớn hơn hoặc bằng thời gian hiện tại.  
 $\forall ddh \in DonDH: ddh.TG_Dat >= GETDATE()$
- Đơn hàng có 5 trạng thái: đang duyệt, đang chuẩn bị, đang giao hàng, đã giao hàng, đã duyệt.  
 $\forall ddh \in DonDH: ddh.TrangThai = N'Dang duyệt' OR ddh.TrangThai = N'Dang chuẩn bị' OR ddh.TrangThai = N'Dang giao hàng' OR ddh.TrangThai = N'Dã hoàn thành' OR ddh.TrangThai = N'Dã hủy'$
- Đơn hàng có 2 hình thức thanh toán: chuyển khoản và tiền mặt.  
 $\forall ddh \in DonDH: ddh.ThanhToan = N'Chuyển khoản' OR N'Tiền mặt'$
- Số lượng sản phẩm trong Chi tiết đơn đặt hàng phải lớn hơn 0.  
 $\forall ct \in CT_DonDH: ct.SoLuong > 0$
- Thời gian bắt đầu giao hàng phải lớn hơn hoặc bằng thời gian hiện tại.  
 $\forall gh \in GiaoHang: gh.ThoiGian_BD >= GETDATE()$
- Thời gian xuất hóa đơn phải lớn hơn hoặc bằng thời gian hiện tại.  
 $\forall hd \in HoaDon: hd.ThoiGian >= GETDATE()$
- Số sao đánh giá nằm trong khoảng từ 1 đến 5 sao.  
 $\forall dg \in DanhGia: dg.Sao BETWEEN 1 AND 5$
- Thời gian đổi trả phải lớn hơn hoặc bằng thời gian hiện tại.  
 $\forall dt \in DoiTra: dt.TG_DoiTra >= GETDATE()$
- Ngày bắt đầu của phiếu bảo hành phải lớn hơn hoặc bằng ngày hiện tại.  
 $\forall pbh \in PhieuBH: pbh.Ngay_BD >= CONVERT(DATE, GETDATE())$
- Thời gian nhận bảo hành phải lớn hơn hoặc bằng thời gian hiện tại.

$\forall \text{bh} \in \text{NhanBH}: \text{bh.TG_BaoHanh} \geq \text{GETDATE()}$

## 2. Ràng buộc toàn vẹn liên thuộc tính

- Ngày kết thúc quá trình quản lý phải lớn hơn ngày bắt đầu.  
 $\forall \text{ql} \in \text{QTQuanLy}: \text{ql.Ngay_KT} > \text{ql.Ngay_BD}$
- Khi thay đổi giá thì giá mới phải khác giá cũ.  
 $\forall \text{td} \in \text{ThayDoiGia}: \text{td.GiaCu} \neq \text{td.GiaMoi}$
- Thời gian kết thúc khuyến mãi phải lớn hơn thời gian bắt đầu.  
 $\forall \text{km} \in \text{KhuyenMai}: \text{km.ThoiGian_KT} > \text{km.ThoiGian_BD}$
- Thời gian kết thúc quá trình giao hàng phải lớn hơn thời gian bắt đầu.  
 $\forall \text{gh} \in \text{GiaoHang}: \text{gh.ThoiGian_KT} > \text{gh.ThoiGian_BD}$
- Ngày kết thúc bảo hành phải lớn hơn ngày bắt đầu.  
 $\forall \text{pbh} \in \text{PhieuBH}: \text{pbh.Ngay_KT} > \text{pbh.Ngay_BD}$

## 3. Ràng buộc toàn vẹn liên thuộc tính liên quan hệ

- Thời gian trả bảo hành phải lớn hơn thời gian nhận bảo hành.

## VI. TRIGGER VÀ STORED PROCEDURE

### 1. Trigger

a. Khi thay đổi giá thì giá cũ phải giống giá hiện tại và khác với giá mới.

```
CREATE TRIGGER TG_TDG_ThayDoiGia ON ThayDoiGia
FOR INSERT, UPDATE
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted i
        INNER JOIN SanPham sp ON i.MaSP = sp.MaSP
        WHERE i.GiaCu <> sp.Gia
    )
    BEGIN
        RAISERROR ('Giá cũ phải giống với giá hiện tại của Sản phẩm.', 16, 1);
    END;

    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE GiaCu = GiaMoi
    )
    BEGIN
        RAISERROR ('Giá cũ phải khác với Giá mới.', 16, 1);
    END;
    UPDATE SanPham
    SET Gia = (SELECT i.GiaMoi FROM inserted i)
    WHERE MaSP = (SELECT i.MASP FROM inserted i)
END;
```

b. Nếu nhân viên đã nghỉ việc thì không được tạo đợt khuyến mãi.

```
CREATE TRIGGER TG_KM_CheckNhanVienNghiViec
ON KhuyenMai
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND NghiViec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;
```

c. Nếu nhân viên đã nghỉ việc thì không được thay đổi giá sản phẩm.

```
CREATE TRIGGER TG_TDG_CheckNhanVienNghiViec
ON ThayDoiGia
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;
```

```

IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND Nghiviec = 0)
BEGIN
    RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
    ROLLBACK TRANSACTION;
    RETURN;
END;
END;

```

d. Nếu nhân viên đã nghỉ việc thì không được duyệt đơn hàng.

```

CREATE TRIGGER TG_DDH_CheckNhanVienNghiViec
ON DonDH
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND Nghiviec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;

```

e. Nếu nhân viên đã nghỉ việc thì không được giao hàng.

```

CREATE TRIGGER TG_GH_CheckNhanVienNghiViec
ON GiaoHang
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND Nghiviec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;

```

f. Nếu nhân viên đã nghỉ việc thì không được lập phiếu đổi trả.

```

CREATE TRIGGER TG_DT_CheckNhanVienNghiViec
ON DoiTra
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND Nghiviec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;

```

```
END;
```

g. Nếu nhân viên đã nghỉ việc thì không được lập phiếu bảo hành.

```
CREATE TRIGGER TG_PBH_CheckNhanVienNghiViec
ON PhieuBH
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND NghiViec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;
```

h. Nếu nhân viên đã nghỉ việc thì không được lập phiếu nhận bảo hành.

```
CREATE TRIGGER TG_NBH_CheckNhanVienNghiViec
ON NhanBH
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND NghiViec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;
```

i. Nếu nhân viên đã nghỉ việc thì không được lập phiếu trả bảo hành.

```
CREATE TRIGGER TG_TBH_CheckNhanVienNghiViec
ON TraBH
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @MaNV VARCHAR(10);

    SELECT @MaNV = inserted.MaNV
    FROM inserted;

    IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @MaNV AND NghiViec = 0)
    BEGIN
        RAISERROR('Nhân viên này hiện đang nghỉ việc.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END;
END;
```

j. Trong một thời điểm chỉ tồn tại một quản lý.

```
CREATE TRIGGER TG_QTQL_TrangThaiQuanLy
ON QTQuanLy
AFTER INSERT, UPDATE
```

```

AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM inserted WHERE TrangThai = 1)
    BEGIN
        IF EXISTS (SELECT 1 FROM QTQuanLy WHERE TrangThai = 1 AND MaQL NOT IN (SELECT MaQL FROM
inserted WHERE TrangThai = 1))
        BEGIN
            RAISERROR ('Đang có nhân viên hiện là Quản lý.', 16, 1)
            ROLLBACK TRANSACTION;
            RETURN;
        END
    END
END;

```

k. Số lượng của sản phẩm ở giỏ hàng không được lớn hơn số lượng tồn của sản phẩm đó.

```

CREATE TRIGGER TG_GioH_CheckSLTon
ON GioHang
FOR INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (
        SELECT 1
        FROM inserted i
        INNER JOIN SanPham s ON i.MaSP = s.MaSP
        WHERE i.SoLuong > s.SLTon
    )
    BEGIN
        RAISERROR ('Số lượng sản phẩm trong giỏ hàng không được lớn hơn số lượng tồn.', 16, 1)
        ROLLBACK TRANSACTION;
        RETURN;
    END
END;

```

l. Cập nhật số lượng tồn của sản phẩm và giỏ hàng khi được đặt hàng.

```

CREATE TRIGGER [dbo].[TG_CTDDH_CapNhatSLTon]

ON [dbo].[CT_DonDH]
FOR INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;
    Update GioHang
        set soluong = soluong - (select inserted.Soluong from inserted,GioHang where
inserted.MaGio = GioHang.MaGio)
        if exists (select 1 from GioHang where soluong<0)
        begin
            RAISERROR ('Số lượng không thể âm.', 16, 1);
            ROLLBACK TRANSACTION;
            end
    UPDATE SanPham
    SET SLTon = SLTon - (SELECT inserted.Soluong
                            FROM SanPham, inserted, GioHang
                            WHERE inserted.MaGio = GioHang.MaGio
                            AND GioHang.MaSP = SanPham.MaSP)

    IF EXISTS (SELECT 1 FROM SanPham WHERE SLTon < 0)
    BEGIN
        RAISERROR ('Số lượng tồn không thể âm.', 16, 1);
    END

```

```

ROLLBACK TRANSACTION;
END
END;

```

*m. Tự động thêm mã hóa đơn khi có đơn đặt hàng mới.*

```

ALTER TRIGGER [dbo].[TG_DDH_ThemHoaDon]
ON [dbo].[DonDH]
AFTER INSERT
AS
BEGIN

    INSERT INTO HoaDon (TongTien)
    VALUES (0)
END;

```

*n. Tự động thêm ngày kết thúc trong bảng Phiếu bảo hành.*

```

CREATE TRIGGER [dbo].[TG_PBH_NgayKT]
ON [dbo].[PhieuBH]
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @thang int
    SET @thang = (
        SELECT TG_BaoHanh
        FROM SanPham SP
        INNER JOIN GioHang GH
        ON SP.MaSP = GH.MaSP
        INNER JOIN CT_DonDH
        ON GH.MaGio = CT_DonDH.MaGio
        INNER JOIN inserted i
        ON CT_DonDH.MaCTDDH = i.MaCTDDH)

    UPDATE PhieuBH
    SET Ngay_KT = DATEADD(MONTH, @thang, Ngay_BD)
END;

```

*o. Tự động update trạng thái khi SoLuong bằng 0 trong bảng Giỏ hàng.*

```

CREATE TRIGGER TG_GioH_UpdateTrangThaiGioHang
ON GioHang
AFTER INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE GioHang
    SET TrangThai = CASE
        WHEN SoLuong = 0 THEN 'False'
        ELSE 'True'
    END
    WHERE MaGio IN (SELECT MaGio FROM inserted);

END;

```

*p. Tự động cập nhật trạng thái của Đơn hàng khi đã giao hàng xong.*

```

CREATE TRIGGER TG_GH_CapNhatTrangThaiSauKhiGiaoHang
ON GiaoHang
AFTER INSERT, UPDATE
AS
BEGIN
    IF UPDATE(ThoiGian_KT)
    BEGIN
        UPDATE DonDH
        SET TrangThai = N'Dã hoàn thành'
    END

```

```

    FROM DonDH d
    JOIN inserted i ON d.MaDDH = i.MaDDH
    WHERE i.ThoiGian_KT IS NOT NULL;
END

```

## 2. Stored Procedure

### a. Tạo tài khoản nhân viên.

```

CREATE PROCEDURE SP_TaoTKNhanVien
    @Ho NVARCHAR(50),
    @Ten NVARCHAR(50),
    @GioiTinh NVARCHAR(3),
    @NgaySinh date,
    @SDT NVARCHAR(20),
    @cccd VARCHAR(12),
    @Email NVARCHAR(100),
    @Username NVARCHAR(50),
    @Password NVARCHAR(50)

AS
BEGIN TRANSACTION
begin try
    SET NOCOUNT ON;

    INSERT INTO NhanVien (Ho, Ten, SDT, Email, NghiViec, GioiTinh, CCCD, NgaySinh)
    VALUES (@Ho, @Ten, @SDT, @Email, 0, @GioiTinh, @cccd, @NgaySinh);
    declare @MANV varchar(10)
    SELECT TOP 1 @MANV = MANV FROM NHANVIEN ORDER BY MANV DESC

    INSERT INTO TK_NhanVien (usernameNV, pwNV, MaNV)
    VALUES (@Username, @Password, @MANV);

commit
end try
BEGIN CATCH
    ROLLBACK
    DECLARE @ErrorMessage VARCHAR(2000)
    SELECT @ErrorMessage = 'Lỗi: ' + ERROR_MESSAGE()
    RAISERROR(@ErrorMessage, 16, 1)
END CATCH

```

Trước khi chạy:

Mie.BGD - dbo.NhanVien		Mie.BGD - dbo.TK_NhanVien									
MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec	usernameNV	pwNV	MaNV
1	Nguyễn Thị Mỹ	Nữ	232323232	2003-10-18	0363032802	nrmhh33@g...	False		huyennhi	yennhi123	3
2	Trần Anh Hùng	Nam	390483248	2002-12-12	0787989799	tahung@g...	False		ll	123456	4
3	Lâm Thị Yến Như	Nữ	398420938	2005-04-01	0243288579	ltyennhu@g...	False		nthmy	123465	1
4	Lê Ngọc Luân	Nam	990694300	2001-01-14	0365325865	llluan123@...	False		tranhung	159357	2
6	Nguyễn Minh Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True		*	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

The screenshot shows two tables side-by-side. The left table, titled 'Mie.BGD - dbo.NhanVien', has columns: MaNV, Ho, Ten, GioiTinh, CCCD, NgaySinh, SDT, Email, and NghiViec. It contains 8 rows of data. The right table, titled 'Mie.BGD - dbo.TK\_NhanVien', has columns: usernameNV, pwNV, and MaNV. It contains 6 rows of data.

	MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec
1		Nguyễn Thị	My	Nữ	232323232	2003-10-18	0363032802	nmhh33@g...	False
2		Trần Anh	Hùng	Nam	390483248	2002-12-12	0787989799	tahung@g...	False
3		Lâm Thị	Yến	Nữ	398420938	2005-04-01	0243288579	lyennhu@g...	False
4		Lê Ngọc	Luân	Nam	990694300	2001-01-14	0365325865	lluan123@g...	False
6		Nguyễn Minh	Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True
8		Trần Minh	Hoàng	Nam	3546490097	2004-02-02	056989789	minhhoang...	False
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

	usernameNV	pwNV	MaNV
▶	huynhhi	yennhi123	3
▶	ll	123456	4
▶	mhoang233	tranminhho...	8
▶	nthmy	123456	1
▶	tranhung	159357	2
*	NULL	NULL	NULL

### b. Nhân viên đăng nhập vào phần mềm.

```

CREATE PROC SP_NhanVienDangNhap
    @username NVARCHAR(50),
    @pw NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM TK_NhanVien WHERE usernameNV = @username)
    BEGIN
        IF EXISTS (SELECT 1 FROM TK_NhanVien WHERE usernameNV = @username AND pwNV = @pw)
        BEGIN
            SELECT NV.Ho, NV.Ten, NV.GioiTinh, NV.CCCD, NV.NgaySinh, NV.SDT, NV.Email
            FROM NhanVien NV
            INNER JOIN TK_NhanVien TK ON NV.MaNV = TK.MaNV
            WHERE TK.usernameNV = @username;
        END
        ELSE
        BEGIN
            RAISERROR('Sai mật khẩu.', 16, 1);
        END
    END
    ELSE
    BEGIN
        RAISERROR('Tài khoản này không tồn tại.', 16, 1);
    END
END;

```

Sau khi chạy:

The screenshot shows the execution of the stored procedure SP\_NhanVienDangNhap. The code is as follows:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC  @return_value = [dbo].[SP_NhanVienDangNhap]
    @username = N'nthmy',
    @pw = N'123456'
SELECT 'Return Value' = @return_value
GO

```

The results pane shows the output of the query 'SELECT 'Return Value' = @return\_value', which returns the value 1. Below the results, a table displays the details of the user 'nthmy' from the NhanVien table.

	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email
1	Nguyễn Thị Huyền	My	Nữ	232323232	2003-10-18	0363032802	nmhh33@gmail.com

c. Chỉnh sửa thông tin nhân viên.

```

CREATE PROC SP_ChinhSuaThongTinNhanVien
@MANV INT,
@HO NVARCHAR(100), @TEN NVARCHAR(50), @GIOITINH NVARCHAR(3), @CCCD VARCHAR(12),
@NGAYSINH DATE, @SDT VARCHAR(10), @EMAIL VARCHAR(50), @NGHIVIEC BIT
AS
BEGIN
    UPDATE NhanVien
    SET
        Ho=@HO, Ten=@TEN, GioiTinh=@GIOITINH, CCCD=@CCCD, NgaySinh=@NGAYSINH, SDT=@SDT, Email=@EMAIL, NghiViec=@NGHIVIEC
    WHERE MANV = @MANV
END

```

Trước khi chạy:

	MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec
▶	1	Nguyễn Thị ...	My	Nữ	232323232	2003-10-18	0363032802	nmhh33@g...	False
	2	Trần Anh	Hùng	Nam	390483248	2002-12-12	0787989799	tahung@g...	False
	3	Lâm Thị Yến	Như	Nữ	398420938	2005-04-01	0243288579	ltynnhu@g...	False
	4	Lê Ngọc	Luân	Nam	990694300	2001-01-14	0365325865	lluan123@...	False
	6	Nguyễn Minh	Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True
	8	Trần Minh	Hoàng	Nam	3546490097	2004-02-02	056989789	minhhoang...	False
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC   @return_value = [dbo].[SP_ChinhSuaThongTinNhanVien]
        @MANV = 2,
        @HO = N'Trần Anh',
        @TEN = N'Dũng',
        @GIOITINH = N'Nam',
        @CCCD = N'415465212',
        @NGAYSINH = '12/12/2002',
        @SDT = N'056528776',
        @EMAIL = N'tahung@gmail.com',
        @NGHIVIEC = 0

SELECT 'Return Value' = @return_value
GO

```

	MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec
▶	1	Nguyễn Thị ...	My	Nữ	232323232	2003-10-18	0363032802	nmhh33@g...	False
	2	Trần Anh	Dũng	Nam	415465212	2002-12-12	056528776	tahung@g...	False
	3	Lâm Thị Yến	Như	Nữ	398420938	2005-04-01	0243288579	ltynnhu@g...	False
	4	Lê Ngọc	Luân	Nam	990694300	2001-01-14	0365325865	lluan123@...	False
	6	Nguyễn Minh	Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

d. Đổi mật khẩu của tài khoản nhân viên.

```

CREATE PROC SP_DoiMatKhauTKNhanVien
    @username NVARCHAR(50),
    @oldPW NVARCHAR(50),
    @newPW NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @existingPW NVARCHAR(50);

    SELECT @existingPW = pwNV
    FROM TK_NhanVien

    WHERE usernameNV = @username;

```

```

IF @existingPW <> @oldPW
BEGIN
    RAISERROR ('Mật khẩu cũ không chính xác.', 16, 1);
    RETURN;
END;

UPDATE TK_NhanVien
SET pwNV = @newPW
WHERE usernameNV = @username;

END;

```

Trước khi chạy:

	usernameNV	pwNV	MaNV
1	huyynhi	yennhi123	3
2	ll	123456	4
3	mhoang233	tranminhho...	8
4	nthmy	123456	1
5	tranhung	159357	2
*	NULL	NULL	NULL

Sau khi chạy:

The screenshot shows the SQL Server Management Studio interface. On the left, the script pane displays the following T-SQL code:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC   @return_value = [dbo].[SP_DoiMatKhauTKNhanVien]
        @username = N'tranhung',
        @oldPW = N'159357',
        @newPW = N'thung159'

SELECT 'Return Value' = @return_value
GO

```

On the right, the results pane shows a table with the same data as the initial screenshot, but the row for 'tranhung' has been updated. The 'pwNV' column now contains 'thung159'.

	usernameNV	pwNV	MaNV
1	huyynhi	yennhi123	3
2	ll	123456	4
3	mhoang233	tranminhho...	8
4	nthmy	123456	1
5	tranhung	thung159	2
*	NULL	NULL	NULL

The results pane also shows a single row with a value of 0 under the 'Return Value' column.

e. Xóa tài khoản của nhân viên nghỉ việc.

```

CREATE PROC SP_XoaTaiKhoanNhanVien
@manv int
AS
BEGIN
    BEGIN TRANSACTION;

    BEGIN TRY
        IF NOT EXISTS (SELECT 1 FROM NhanVien WHERE MaNV = @manv)
        BEGIN
            RAISERROR('Mã nhân viên không tồn tại.', 16, 1);
            ROLLBACK TRANSACTION;
            RETURN;
        END
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION;
        RETURN;
    END CATCH
END

```

```

        UPDATE NhanVien
        SET NghiViec = 1
        WHERE MaNV = @manv;

        DELETE FROM TK_NhanVien
        WHERE MaNV = @manv;

        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
END

```

Trước khi chạy:

Mie.BGD - dbo.NhanVien								
MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec
1	Nguyễn Thị	Mỹ	Nữ	232323232	2003-10-18	0363032802	nmhh33@g...	False
2	Trần Anh	Dũng	Nam	415465212	2002-12-12	056528776	tahung@g...	False
3	Lâm Thị	Yến	Nữ	398420938	2005-04-01	0243288579	lyennhu@g...	False
4	Lê Ngọc	Luân	Nam	990694300	2001-01-14	0365325865	lluan123@...	False
6	Nguyễn Minh	Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True
8	Trần Minh	Hoàng	Nam	3546490097	2004-02-02	056989789	minhhoang...	False
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Mie.BGD - dbo.TK_NhanVien		
usernameNV	pwNV	MaNV
huynhni	yennhi123	3
ll	123456	4
mhoang233	tranminhho...	8
nthmy	123456	1
tranhung	thung159	2
*	NULL	NULL

Sau khi chạy:

Mie.BGD - dbo.NhanVien								
MaNV	Ho	Ten	GioiTinh	CCCD	NgaySinh	SDT	Email	NghiViec
1	Nguyễn Thị	Mỹ	Nữ	232323232	2003-10-18	0363032802	nmhh33@g...	False
2	Trần Anh	Dũng	Nam	415465212	2002-12-12	056528776	tahung@g...	False
3	Lâm Thị	Yến	Nữ	398420938	2005-04-01	0243288579	lyennhu@g...	False
4	Lê Ngọc	Luân	Nam	990694300	2001-01-14	0365325865	lluan123@...	False
6	Nguyễn Minh	Nhật	Nam	48723500675	2002-02-02	0458792239	soisieuquay...	True
8	Trần Minh	Hoàng	Nam	3546490097	2004-02-02	056989789	minhhoang...	True
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Mie.BGD - dbo.TK_NhanVien		
usernameNV	pwNV	MaNV
huynhni	yennhi123	3
ll	123456	4
nthmy	123456	1
tranhung	thung159	2
*	NULL	NULL

f. Thêm một sản phẩm mới.

```

CREATE PROCEDURE SP_ThemSanPhamMoi
    @tenSP NVARCHAR(50),
    @mausac NVARCHAR(10),
    @size TINYINT,
    @chatlieu NVARCHAR(30),
    @xuatxu NVARCHAR(30),
    @gia money,
    @tg_bh tinyint,
    @slt smallint,
    @mahang varchar(10),
    @maloai varchar(10)
AS
BEGIN
    INSERT INTO SanPham(TenSP, MauSac, Size, ChatLieu,
XuatXu,Gia,TG_BaoHanh,SLTon,MaHang,MaLoai)
    VALUES (@tenSP, @mausac, @size, @chatlieu, @xuatxu,@gia,@tg_bh,@slt,@mahang,@maloai);
END;

```

Trước khi chạy:

	MaSP	TenSP	MauSac	Size	ChatLieu	XuatXu	Gia	TG_BaoHanh	SLTon	MaHang	MaLoai
▶	1	Giày Nike Air Force 1	Đỏ	36	Da	Viet Nam	1500000.00...	6	3	NIKE	SNE
	2	Giày Nike Air Force 1	Đỏ	37	Da	Viet Nam	2000000.00...	6	12	NIKE	SNE
	3	Giày Nike Jordan 1	Xanh trắng	37	Da	Viet Nam	2300000.00...	12	18	NIKE	SNE
	4	Giày Converse 1970s	Xanh	37	Da	Viet Nam	1820000.00...	12	12	CONV	SNE
	5	Giày Converse 1970s	Vàng	37	Da	Viet Nam	2000000.00...	6	15	CONV	SNE
	6	Giày Converse 1970s	Vàng	36	Da	Viet Nam	2100000.00...	6	7	CONV	SNE
	7	Giày Adidas Ultra Boost 22	Trắng	36	Da	Viet Nam	3100000.00...	5	9	ADID	SNE
	8	Giày Adidas Terrex Speed Ultra	Đen	36	Da	Viet Nam	1100000.00...	6	25	ADID	SNE
	9	Giày Adidas Ultra Boost 22	Đen Trắng	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
*	10	Giày Adidas Ultra Boost 22	Đen Xanh	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int

EXEC    @return_value = [dbo].[SP_ThemSanPhamMoi]
        @tensp = N'Giày Nike Ultra Boots',
        @mausac = N'Trắng',
        @size = 38,
        @chatlieu = N'Da',
        @xuatxu = N'Việt Nam',
        @gia = 5000000,
        @tg_bh = 10,
        @slt = 10,
        @mahang = N'NIKE',
        @maloai = N'SNE'

SELECT 'Return Value' = @return_value

```

	MaSP	TenSP	MauSac	Size	ChatLieu	XuatXu	Gia	TG_BaoHanh
▶	1	Giày Nike Air Force 1	Đỏ	36	Da	Viet Nam	1500000.00...	6
	2	Giày Nike Air Force 1	Đỏ	37	Da	Viet Nam	2000000.00...	6
	3	Giày Nike Jordan 1	Xanh trắng	37	Da	Viet Nam	2300000.00...	12
	4	Giày Converse 1970s True Navy	Xanh	37	Da	Viet Nam	1820000.00...	12
	5	Giày Converse 1970s True Navy	Vàng	37	Da	Viet Nam	2000000.00...	6
	6	Giày Converse 1970s True Navy	Vàng	36	Da	Viet Nam	2100000.00...	6
	7	Giày Adidas Ultra Boost 22	Trắng	36	Da	Viet Nam	3100000.00...	5
	8	Giày Adidas Duramo 10	Đen	36	Da	Viet Nam	1100000.00...	6
	9	Giày Adidas Terrex Speed Ultra	Đen Trắng	37	Da	Viet Nam	3300000.00...	12
	10	Giày Adidas Terrex Speed Ultra	Đen Xanh	37	Da	Viet Nam	3300000.00...	12
*	11	Giày Nike Ultra Boots	Trắng	38	Da	Viet Nam	5000000.00...	10
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

g. Thay đổi thông tin sản phẩm.

CREATE PROCEDURE SP\_ThayDoiThongTinSanPham

```

@masp INT,
@tensp NVARCHAR(50),
@mausac NVARCHAR(10),
@size TINYINT,
@chatlieu NVARCHAR(30),
@xuatxu NVARCHAR(30),
@gia money,
@tg_bh tinyint,
@slt smallint,
@mahang varchar(10),
@maloai varchar(10)

```

AS

BEGIN

```
SET NOCOUNT ON;
```

```
UPDATE SanPham
```

```

SET TenSP = @tensp,
MauSac = @mausac,
Size = @size,
ChatLieu=@chatlieu,
XuatXu=@xuatxu,
Gia=@gia,
TG_BaoHanh=@tg_bh,
SLTon=@slt,
MaHang=@mahang,

```

MaLoai=@maloai

```
WHERE MaSP = @masp;
END;
```

Trước khi chạy:

	MaSP	TenSP	MauSac	Size	ChatLieu	XuatXu	Gia	TG_BaoHanh	SLTon	MaHang	MaLoai
▶	1	Giày Nike Air Force 1	Đỏ	36	Da	Viet Nam	1500000.00...	6	3	NIKE	SNE
	2	Giày Nike Air Force	Đỏ	37	Da	Viet Nam	2000000.00...	6	12	NIKE	SNE
	3	Giày Nike Jordan 1	Xanh trắng	37	Da	Viet Nam	2300000.00...	12	18	NIKE	SNE
	4	Giày Converse 1970s True Navy	Xanh	37	Da	Viet Nam	1820000.00...	12	12	CONV	SNE
	5	Giày Converse 1970s True Navy	Vàng	37	Da	Viet Nam	2000000.00...	6	15	CONV	SNE
	6	Giày Converse 1970s True Navy	Vàng	36	Da	Viet Nam	2100000.00...	6	7	CONV	SNE
	7	Giày Adidas Ultra Boost 22	Trắng	36	Da	Viet Nam	3100000.00...	5	9	ADID	SNE
	8	Giày Adidas Duramo 10	Đen	36	Da	Viet Nam	1100000.00...	6	25	ADID	SNE
	9	Giày Adidas Terrex Speed Ultra	Đen Trắng	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
	10	Giày Adidas Terrex Speed Ultra	Đen Xanh	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
*	12	Giày Nike Ultra Boots	Trắng	38	Da	Viet Nam	5000000.00...	10	10	NIKE	SNE
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

The screenshot shows two panes in SQL Server Management Studio. The left pane displays the T-SQL code for a stored procedure named [SP\_ThayDoiThongTinSanPham]. The right pane shows the results of the stored procedure execution, which is a table titled 'Mie.BGD - dbo.SanPham'. The table contains 12 rows of product information, identical to the one shown in the previous screenshot.

	MaSP	TenSP	MauSac	Size	ChatLieu	XuatXu	Gia	TG_BaoHanh	SLTon	MaHang	MaLoai
▶	1	Giày Nike Air Force 1	Đỏ	36	Da	Viet Nam	1500000.00...	6	3	NIKE	SNE
	2	Giày Nike Air Force	Đỏ	37	Da	Viet Nam	2000000.00...	6	12	NIKE	SNE
	3	Giày Nike Jordan 1	Xanh trắng	37	Da	Viet Nam	2300000.00...	12	18	NIKE	SNE
	4	Giày Converse 1970s True Navy	Xanh	37	Da	Viet Nam	1820000.00...	12	12	CONV	SNE
	5	Giày Converse 1970s True Navy	Vàng	37	Da	Viet Nam	2000000.00...	6	15	CONV	SNE
	6	Giày Converse 1970s True Navy	Vàng	36	Da	Viet Nam	2100000.00...	6	7	CONV	SNE
	7	Giày Adidas Ultra Boost 22	Trắng	36	Da	Viet Nam	3100000.00...	5	9	ADID	SNE
	8	Giày Adidas Duramo 10	Đen	37	Da	Viet Nam	2300000.00...	10	25	ADID	SNE
	9	Giày Adidas Terrex Speed Ultra	Đen Trắng	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
	10	Giày Adidas Terrex Speed Ultra	Đen Xanh	37	Da	Viet Nam	3300000.00...	12	12	ADID	SNE
*	12	Giày Nike Ultra Boots	Trắng	38	Da	Viet Nam	5000000.00...	10	10	NIKE	SNE
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

h. Xóa sản phẩm trong cửa hàng nếu nó chưa được thêm vào gio.

```
CREATE PROC SP_XoaSanPham
@masp INT
AS
BEGIN
    BEGIN TRANSACTION;

    BEGIN TRY
        IF EXISTS (SELECT 1 FROM GioHang WHERE MaSP = @masp)
        BEGIN
            RAISERROR('Sản phẩm này không thể xóa.', 16, 1);
            ROLLBACK TRANSACTION;
            RETURN;
        END

        DELETE FROM SanPham
        WHERE MaSP = @masp;

        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
END
```

Trước khi chạy:

MaSP	TenSP	MauSac	Size	ChatLieu	XuatXu	Gia	TG_BaoHanh	SLTon
1	Giày Nike Air Force 1	Đỏ	36	Da	Viet Nam	1500000.00...	6	3
2	Giày Nike Air Force	Đỏ	37	Da	Viet Nam	2000000.00...	6	12
3	Giày Nike Jordan 1	Xanh trắng	37	Da	Viet Nam	2300000.00...	12	18
4	Giày Converse 1970s True Navy	Xanh	37	Da	Viet Nam	1820000.00...	12	12
5	Giày Converse 1970s True Navy	Vàng	37	Da	Viet Nam	2000000.00...	6	15
6	Giày Converse 1970s True Navy	Vàng	36	Da	Viet Nam	2100000.00...	6	7
7	Giày Adidas Ultra Boost 22	Trắng	36	Da	Viet Nam	3100000.00...	5	9
8	Giày Adidas Duramo 10	Đen	37	Da	Viet Nam	2300000.00...	10	25
9	Giày Adidas Terrex Speed Ultra	Đen Trắng	37	Da	Viet Nam	3300000.00...	12	12
10	Giày Adidas Terrex Speed Ultra	Đen Xanh	37	Da	Viet Nam	3300000.00...	12	12
12	Giày Nike Ultra Boots	Trắng	38	Da	Viet Nam	5000000.00...	10	10
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

MaGio	usernameKH	MaSP	Soluong	TrangThai
1	mie239	2	0	False
2	mie239	5	1	True
3	luat03	7	0	False
5	luat03	8	1	True
6	luat03	2	0	False
7	xnam2003	3	1	True
8	tmkhai22	1	3	True
10	luat03	6	0	False
11	luat03	10	0	False
*	NULL	NULL	NULL	NULL

Sau khi chạy:

MaGio	usernameKH	MaSP	Soluong	TrangThai
1	mie239	2	0	False
2	mie239	5	1	True
3	luat03	7	0	False
5	luat03	8	1	True
6	luat03	2	0	False
7	xnam2003	3	1	True
8	tmkhai22	1	3	True
10	luat03	6	0	False
11	luat03	10	0	False
*	NULL	NULL	NULL	NULL

### i. Tạo đợt khuyến mãi.

```

CREATE PROC SP_TaoDotKhuyenMai
@manv INT,
@lyDo NVARCHAR(100),
@tgianBD DATETIME,
@tgianKT DATETIME
AS
BEGIN
    SET NOCOUNT ON;
    IF EXISTS (SELECT 1 FROM KhuyenMai WHERE ThoiGian_BD = @tgianBD AND ThoiGian_KT =
@tgianKT)
        BEGIN
            RAISERROR('Đã tồn tại mã đợt khuyến mãi khác trùng với khoảng thời gian này.', 16,
1);
        END
    ELSE
        BEGIN
            INSERT INTO KhuyenMai(MaNV, LyDo, ThoiGian_BD, ThoiGian_KT)
            VALUES (@manv, @lyDo, @tgianBD, @tgianKT)
        END
END

```

Trước khi chạy:

Sau khi chạy:

### j. Thêm sản phẩm vào một đợt khuyến mãi.

```

CREATE PROC SP_ThemSanPhamVaoDotKM
    @madotkm int,
    @masp int,
    @phantram tinyint
AS
BEGIN
    IF NOT EXISTS (SELECT * FROM KhuyenMai WHERE MaDotKM = @madotkm)
    BEGIN
        RAISERROR('Mã đợt khuyến mãi này không tồn tại.', 16, 1);
        RETURN
    END

    IF NOT EXISTS (SELECT * FROM SanPham WHERE MaSP = @masp)
    BEGIN
        RAISERROR('Mã sản phẩm này không tồn tại.', 16, 1);
        RETURN
    END

    INSERT INTO CT_KhuyenMai (MaDotKM, MaSP, PhanTramKM)
    VALUES (@madotkm, @masp, @phantram)
END

```

Trước khi chạy:

Sau khi chạy:

The screenshot shows two panes in SQL Server Management Studio. The left pane displays the T-SQL code for a stored procedure:

```
USE [BGD]
GO

DECLARE @return_value int
EXEC    @return_value = [dbo].[SP_ThemSanPhamVaoDotKM]
        @madotkm = 10,
        @masp = 8,
        @phantram = 10
SELECT  'Return Value' = @return_value
GO
```

The right pane shows the results of the execution. It contains a table named 'CT\_KhuyenMai' with three columns: MaDotKM, MaSP, and PhanTramK\_. The data is as follows:

	MaDotKM	MaSP	PhanTramK_
1	6	2	10
2	10	8	10
*	NULL	NULL	NULL

Below the table, the 'Results' tab shows a single row with the value '0' under the 'Return Value' column.

k. Cập nhật thông tin của đợt khuyến mãi.

```
CREATE PROC SP_CapNhatDotKhuyenMai
@maDotKM int,
@manv int,
@lyDo NVARCHAR(100),
@tgianBD DATETIME,
@tgianKT DATETIME
AS
BEGIN
    SET NOCOUNT ON;
    UPDATE KhuyenMai
    SET MaNV = @manv,
    LyDo = @lyDo,
    ThoiGian_BD = @tgianBD,
    ThoiGian_KT = @tgianKT
    WHERE MaDotKM = @maDotKM
END
```

Trước khi chạy:

The screenshot shows the 'KhuyenMai' table in the 'Mie.BGD - dbo.KhuyenMai' window. The table has six columns: MaDotKM, MaNV, LyDo, ThoiGian\_BD, and ThoiGian\_KT. There are three rows of data:

	MaDotKM	MaNV	LyDo	ThoiGian_BD	ThoiGian_KT
1	6	1	Tri ân 1 Ngày nhà giáo Việt Nam	2024-05-30 07:3...	2024-06-05 0...
2	10	2	Mừng ngày Trái Đất	2024-06-06 00:0...	2024-06-10 0...
*	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC   @return_value = [dbo].[SP_CapNhatDotKhuyenMai]
@maDotKM = 6,
@manv = 3,
@lyDo = N'Tri ân Ngày Nhà giáo Việt Nam',
@tgianBD = N'6/1/2024',
@tgianKT = N'6/5/2024'
SELECT  'Return Value' = @return_value
GO

```

	MaDotKM	MaNV	LyDo	ThoiGian_BD	ThoiGian_KT
1	6	3	Tri ân Ngày Nhà giáo Việt Nam	2024-06-01 00:00:00.000	2024-06-05 00:00:00.000
2	10	2	Mừng ngày Trái Đất	2024-06-06 00:00:00.000	2024-06-10 00:00:00.000
3	NULL	NULL	NULL	NULL	NULL

### *l. Xem đơn đặt hàng theo trạng thái.*

```

CREATE PROC SP_XemDonDHTheoTrangThai
    @trangthai NVARCHAR(30)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT *
    FROM DonDH
    WHERE TrangThai = @trangthai;
END;

```

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC   @return_value = [dbo].[SP_XemDonDHTheoTrangThai]
@trangthai = N'Dã hoàn thành'

SELECT  'Return Value' = @return_value
GO

```

MaDDH	TG_Dot	TrangThai	HinhThuc	ThanhToan	MaNV
1	2024-05-30 14:50:00.000	Đã hoàn thành	Online	Chuyển khoản	2
2	2024-06-01 00:00:00.000	Đã hoàn thành	Offline (không giao hàng)	Tiền mặt	1
3	2024-07-02 00:00:00.000	Đã hoàn thành	Offline (không giao hàng)	Chuyển khoản	3
4	2024-07-12 00:00:00.000	Đã hoàn thành	Online	Chuyển khoản	1
5	2024-09-02 00:00:00.000	Đã hoàn thành	Online	Tiền mặt	3
6	2024-05-30 02:10:00.000	Đã hoàn thành	Online	Chuyển khoản	2

### *m. Duyệt đơn hàng đã đặt.*

```

CREATE PROC SP_DuyetDonHang
    @maddh int,
    @manv int
AS
BEGIN
    IF EXISTS (SELECT * FROM DonDH WHERE MaDDH = @maddh AND TrangThai = N'Đang duyệt')
    BEGIN
        UPDATE DonDH
        SET TrangThai = N'Đang chuẩn bị',
            MaNV = @manv
        WHERE MaDDH = @maddh
    END

```

```

END

ELSE
BEGIN
    RAISERROR('Đơn hàng này không tồn tại hoặc không thể duyệt.', 16, 1)
END

END

```

Trước khi chạy:

	MaDDH	TG_Dat	TrangThai	HinhThuc	ThanhToan	MaNV
▶	2	2024-05-30 ...	Đã hoàn thành	Online	Chuyển kho...	2
	3	2024-06-01 ...	Đã hoàn thành	Offline (khô...	Tiền mặt	1
	5	2024-07-02 ...	Đã hoàn thành	Offline (khô...	Chuyển kho...	3
	6	2024-07-12 ...	Đã hoàn thành	Online	Chuyển kho...	1
	7	2024-09-02 ...	Đã hoàn thành	Online	Tiền mặt	3
	10	2024-05-30 ...	Đã hoàn thành	Online	Chuyển kho...	2
*	17	2024-05-31 ...	Đang duyệt	Online	Chuyển kho...	NULL
	* NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

n. Lập đơn giao hàng.

```

CREATE PROCEDURE SP_GiaoHangTheoDon
    @manv INT,
    @maddh INT,
    @ten NVARCHAR(100),
    @sdt NVARCHAR(15),
    @diachi NVARCHAR(200)
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRANSACTION;

    BEGIN TRY
        IF EXISTS (SELECT 1 FROM DonDH WHERE MaDDH = @maddh AND TrangThai = N'Dang chuẩn bị')
        BEGIN
            INSERT INTO GiaoHang (MaNV, MaDDH, TenNguoiNhan, SDT, DiaChiGH)
            VALUES (@manv, @maddh, @ten, @sdt, @diachi);

            UPDATE DonDH
            SET TrangThai = N'Dang giao hàng'
            WHERE MaDDH = @maddh;
        END
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION;
    END CATCH
END

```

```

ELSE
BEGIN
    RAISERROR('Đơn hàng không tồn tại hoặc trạng thái không hợp lệ.', 16, 1);
    ROLLBACK TRANSACTION;
    RETURN;
END

COMMIT TRANSACTION;
END TRY

BEGIN CATCH
DECLARE @ErrorMessage NVARCHAR(4000)

SELECT
    @ErrorMessage = ERROR_MESSAGE()

RAISERROR (@ErrorMessage, 16, 1);

ROLLBACK TRANSACTION;
END CATCH
END

```

Trước khi chạy:

	MaDDH	TG_Dat	TrangThai	HinThuc	ThanhToan	MaNV
▶	2	2024-05-30 ...	Đã hoàn thành	Online	Chuyển kho...	2
	3	2024-06-01 ...	Đã hoàn thành	Offline (khô...	Tiền mặt	1
	5	2024-07-02 ...	Đã hoàn thành	Offline (khô...	Chuyển kho...	3
	6	2024-07-12 ...	Đã hoàn thành	Online	Chuyển kho...	1
	7	2024-09-02 ...	Đã hoàn thành	Online	Tiền mặt	3
	10	2024-05-30 ...	Đã hoàn thành	Online	Chuyển kho...	2
*	17	2024-05-31 ...	Đang chuẩn bị	Online	Chuyển kho...	3
	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int

EXEC  @return_value = [dbo].[SP_GiaoHangTheoDon]
@manv = 1,
@madvn = 17,
@ten = N'Cường',
@sdt = N'054678999',
@diachi = N'Quận Bình Thạnh'

SELECT 'Return Value' = @return_value
GO

```

	MaGH	MaNV	MaDDH	ThoiGian_BD	ThoiGian_KT	TenNguoiN...	SDT	DiaChiGH
▶	2	1	2	2024-05-30 ...	2024-05-31 ...	Mie	0363032802	Phường Tân...
	3	3	6	2024-07-13 ...	2024-07-20 ...	Thành Nam	0329699232	Phường 2, Q...
	7	1	10	2024-05-31 ...	2024-06-02 ...	Mie	023458	Quận 9
*	8	1	17	2024-05-31 ...	NULL	Cường	054678999	Quận Bình T...
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

o. Tạo tài khoản khách hàng.

```
CREATE PROCEDURE SP_TaoTKKhachHang
```

```

    @Ho NVARCHAR(50),
    @Ten NVARCHAR(50),
    @SDT NVARCHAR(20),
    @Email NVARCHAR(100),
    @Username NVARCHAR(50),
    @Password NVARCHAR(50)

```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```

INSERT INTO KhachHang (Ho, Ten, SDT, Email)
VALUES (@Ho, @Ten, @SDT, @Email);
declare @MaKH NVARCHAR(10)
select top 1 @MaKH = MaKH from KhachHang order by MaKH desc
INSERT INTO TK_KhachHang (usernameKH, pwKH, MaKH)
VALUES (@Username, @Password, @MaKH);

```

END;

Sau khi chạy:

SQLQuery129.sql - MIE.BGD (sa (66))					Mie.BGD - dbo.KhachHang			Mie.BGD - dbo.TK_KhachHang		
	MaKH	Ho	Ten	SDT	Email		usernameKH	pwKH	MaKH	
▶	1	Nguyễn Thị Mỹ		0363032802	nmhhuong3...		luan003	ahihi	2	
	2	Lê Ngọc Luân		0545452158	Ingola@gm...		luat03	456456	3	
	3	Nguyễn Đinh Luật		0916487224	luatnwe@g...		mie239	12345	1	
	4	Đặng Xuân Nam		0787512942	dangnam2...		nam	123	6	
	5	Trần Minh Khải		0456487898	serwr@gma...		thaiha22	22022005	11	
	6	dang	nam	23232	n@gmail.co...		tmkhai22	123456	5	
	9	dang	nam	010603	nam111@g...		vqha	123	10	
	10	v	a	09013	q12@gmail...		xnam2003	hihihihi	4	
*	11	Nguyễn Thái Hà		034568725	thaiha222@...		NULL	NULL	NULL	
	NULL	NULL	NULL	NULL	NULL		NULL	NULL	NULL	

p. Khách hàng đăng nhập vào phần mềm.

```

CREATE PROC SP_KhachHangDangNhap
    @username NVARCHAR(50),
    @pw NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM TK_KhachHang WHERE usernameKH = @username)
    BEGIN
        IF EXISTS (SELECT 1 FROM TK_KhachHang WHERE usernameKH = @username AND pwKH = @pw)
        BEGIN
            SELECT KH.MaKH, KH.Ho, KH.Ten, KH.SDT, KH.Email
            FROM KhachHang KH
            INNER JOIN TK_KhachHang TK ON KH.MaKH = TK.MaKH
            WHERE TK.usernameKH = @username;
        END
        ELSE
        BEGIN
            RAISERROR('Sai mật khẩu.', 16, 1);
        END
    END
    ELSE
    BEGIN
        RAISERROR('Tài khoản không tồn tại.', 16, 1);
    END
END;

```

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC   @return_value = [dbo].[SP_KhachHangDangNhap]
        @username = N'thaiha.22',
        @pw = N'123456'
SELECT 'Return Value' = @return_value
GO

```

Results

```

Msg 50000, Level 16, State 1, Procedure dbo.SP_KhachHangDangNhap, Line 19 [Batch Start Line 2]
Sai mật khẩu.

(1 row affected)

Completion time: 2024-05-31T05:41:15.3603855+07:00

```

q. Đổi mật khẩu tài khoản của Khách hàng.

```

CREATE PROC SP_DoiMatKhuuTKKhachHang
    @username NVARCHAR(50),
    @oldPW NVARCHAR(50),
    @newPW NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @existingPW NVARCHAR(50);

    SELECT @existingPW = pwKH
    FROM TK_KhachHang
    WHERE usernameKH = @username;

    IF @existingPW <> @oldPW
    BEGIN
        RAISERROR ('Mật khẩu cũ không chính xác.', 16, 1);
        RETURN;
    END;

    UPDATE TK_KhachHang
    SET pwKH = @newPW
    WHERE usernameKH = @username;

END;

```

Trước khi chạy:

	Mie.BGD - dbo.KhachHang	Mie.BGD - dbo.TK_KhachHang	
	usernameKH	pwKH	MaKH
▶	luan003	ahihii	2
	luat03	456456	3
	mie239	12345	1
	nam	123	6
	thaiha.22	22022005	11
	tmkhai22	123456	5
	vqha	123	10
	xnam2003	hihihihi	4
*	NULL	NULL	NULL

Sau khi chạy:

The screenshot shows two panes in SQL Server Management Studio. The left pane displays the T-SQL code for changing a password:

```
USE [BGD]
GO

DECLARE @return_value int
EXEC  @return_value = [dbo].[SP_DoiMatKhauTKKhachHang]
@username = N'mie239',
@oldPW = N'12345',
@newPW = N'000159'

SELECT 'Return Value' = @return_value
GO
```

The right pane shows the results of the query, which is a single row with a value of 0.

usernameKH	pwKH	MaKH
luan003	ahihii	2
luat03	456456	3
mie239	000159	1
nam	123	6
thaisha22	22022005	11
tmkhai22	123456	5
vqha	123	10
xnam2003	hihihihi	4
*	NULL	NULL

r. Thay đổi thông tin của Khách hàng.

```
CREATE PROCEDURE SP_ChinhSuaThongTinKhachHang
@usernameKH VARCHAR(50),
@Ho NVARCHAR(100),
@Ten NVARCHAR(50),
@Sdt NVARCHAR(20),
@email NVARCHAR(100)

AS
BEGIN
SET NOCOUNT ON;

DECLARE @MaKh VARCHAR(10);

SELECT @MaKh = MaKh
FROM TK_KhachHang
WHERE usernameKH = @usernameKH;

IF @MaKh IS NULL
BEGIN
PRINT 'không tìm thấy KH nào';
RETURN;
END

UPDATE KhachHang
SET
Ho = COALESCE(@Ho, Ho),
Ten = COALESCE(@Ten, Ten),
Email = COALESCE(@Email, Email),
SDT = COALESCE(@Sdt, Sdt)
WHERE maKh = @MaKh
END;
```

Trước khi chạy:

Mie.BGD - dbo.KhachHang

MaKH	Ho	Ten	SDT	Email
1	Nguyễn Thị ...	My	0363032802	nmhuong3...
2	Lê Ngọc	Luân	0545452158	Ingola@gm...
3	Nguyễn Đinh	Luật	0916487224	luatnwe@g...
4	Đặng Xuân	Nam	0787512942	dangxnam2...
5	Trần Minh	Khai	0456487898	serwr@gma...
6	dang	nam	23232	n@gmail.co...
9	dang	nam	010603	nam111@g...
10	v	a	09013	q12@gmail...
11	Nguyễn Thái	Hà	034568725	thaisha222@...

Mie.BGD - dbo.TK\_KhachHang

usernameKH	pwKH	MaKH
luan003	ahihi	2
luat03	456456	3
mie239	000159	1
nam	123	6
thaiha22	22022005	11
tmkhai22	123456	5
vqua	123	10
xnam2003	hihihihi	4
*	NULL	NULL

Sau khi chạy:

SQLQuery132.sql - MIE.BGD (sa (82))

```

USE [BGD]
GO

DECLARE @return_value int

EXEC    @return_value = [dbo].[SP_ChinhSuaThongTinKhachHang]
        @usernameKH = N'thaiha.22',
        @Ho = N'Nguyễn Thái',
        @Ten = N'Hà',
        @Sdt = N'0123123215',
        @Email = N'thaiha_ha@gmail.com'

SELECT  'Return Value' = @return_value
GO

```

Results

Return Value
1 0

s. Thông tin của Giỏ hàng theo tài khoản Khách hàng.

```

CREATE PROC SP_ThongTinGioHangTheoTKKhachHang
    @username NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM GioHang WHERE usernameKH = @username AND TrangThai = 1)
    BEGIN
        SELECT SP.TenSP, SP.MauSac, SP.Size, GH.SoLuong, SP.Gia
        FROM GioHang GH
        INNER JOIN SanPham SP ON GH.MaSP = SP.MaSP
        WHERE GH.usernameKH = @username AND GH.TrangThai = 1;
    END
END;

```

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC    @return_value = [dbo].[SP_ThongTinGioHangTheoTKKhachHang]
        @username = N'mie239'

SELECT 'Return Value' = @return_value
GO

```

Results

TenSP	MauSac	Size	SoLuong	Gia
Giày Converse 1970s True Navy	Vàng	37	1	2000000.00

### t. Đặt một đơn hàng.

```

CREATE PROC SP_TaoDonHang
@HINHTHUC NVARCHAR(20), @THANHTOAN NVARCHAR(20)
AS
BEGIN
    INSERT INTO DonDH(TG_Dat, TrangThai, Hinht Thuc, ThanhToan)
    VALUES(GETDATE(), N'Dang duyệt', @HINHTHUC, @THANHTOAN);
END

```

Trước khi chạy:

Mie.BGD - dbo.DonDH

	MaDDH	TG_Dat	TrangThai	Hinht Thuc	ThanhToan	MaNV
2	2024-05-30 ...	Đã hoàn thà...	Online	Chuyển kho...	2	
3	2024-06-01 ...	Đã hoàn thà...	Offline (khô...	Tiền mặt	1	
5	2024-07-02 ...	Đã hoàn thà...	Offline (khô...	Chuyển kho...	3	
6	2024-07-12 ...	Đã hoàn thà...	Online	Chuyển khoản	1	
7	2024-09-02 ...	Đã hoàn thà...	Online	Tiền mặt	3	
10	2024-05-30 ...	Đã hoàn thà...	Online	Chuyển kho...	2	
*	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

Mie.BGD - dbo.DonDH

	MaDDH	TG_Dat	TrangThai	Hinht Thuc	ThanhToan	MaNV
2	2024-05-30 ...	Đã hoàn thà...	Online	Chuyển kho...	2	
3	2024-06-01 ...	Đã hoàn thà...	Offline (khô...	Tiền mặt	1	
5	2024-07-02 ...	Đã hoàn thà...	Offline (khô...	Chuyển kho...	3	
6	2024-07-12 ...	Đã hoàn thà...	Online	Chuyển kho...	1	
7	2024-09-02 ...	Đã hoàn thà...	Online	Tiền mặt	3	
10	2024-05-30 ...	Đã hoàn thà...	Online	Chuyển kho...	2	
17	2024-05-31 ...	Dang duyệt	Online	Chuyển kho...	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL

Results

Return Value	
1	0

### u. Thêm chi tiết đơn hàng.

```

CREATE PROCEDURE SP_ThemChiTietDonHang
    @magio INT,
    @maddh INT,
    @sl INT,
    @mahd INT
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRANSACTION;

    IF NOT EXISTS (SELECT 1 FROM GioHang WHERE MaGio = @magio AND TrangThai <> 0)
    BEGIN
        RAISERROR('Mã Giỏ không tồn tại.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END

    IF NOT EXISTS (SELECT 1 FROM DonDH WHERE MaDDH = @maddh)
    BEGIN
        RAISERROR('Mã Đơn hàng không tồn tại.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END

    IF EXISTS (SELECT 1
                FROM CT_DonDH CT
                JOIN HoaDon HD ON CT.MaHD = HD.MaHD
                WHERE CT.MaDDH = @maddh AND HD.TongTien <> 0)
    BEGIN
        RAISERROR('Đơn hàng này đã được thanh toán.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END

    INSERT INTO CT_DonDH (MaGio, MaDDH, SoLuong, MaHD)
    VALUES (@magio, @maddh, @sl, @mahd);

    COMMIT TRANSACTION;
END

```

Sau khi chạy:

MaCTDDH	MaGio	MaDDH	SoLuong	MaHD
1	2	1	3	
2	3	3	1	4
3	6	3	2	4
4	1	5	1	5
5	10	6	1	6
6	11	6	1	6
7	8	7	10	7
9	10	10	1	9
10	2	17	1	14
*	NULL	NULL	NULL	NULL

v. *Tính tiền Hóa đơn theo mã Đơn đặt hàng.*

```

CREATE PROC SP_TinhTienHoaDonTheoMa
    @MADDH INT,
    @MANV INT

```

```

AS
BEGIN
    DECLARE @TONGTIEN FLOAT, @MAHD VARCHAR(20)
    SELECT DISTINCT @MAHD = MAHD FROM CT_DonDH WHERE MADDH = @MADDH

    SELECT SP.MASP, SP.GIA, CTDH.SOLUONG, PhanTramKM = (
CASE
    WHEN (SP.MASP IN (SELECT MASP FROM CT_KhuyenMai) AND GETDATE() BETWEEN KM.ThoiGian_BD AND KM.ThoiGian_KT) THEN ISNULL(PhanTramKM, 0)
    ELSE 0
END), THANHTIEN = (
CASE
    WHEN (SP.MASP IN (SELECT MASP FROM CT_KhuyenMai) AND GETDATE() BETWEEN KM.ThoiGian_BD AND KM.ThoiGian_KT) THEN CTDH.SOLUONG*SP.GIA*((100-ISNULL(PhanTramKM, 0))/100.00)
    ELSE CTDH.SOLUONG*SP.GIA
END)
INTO #TEMP
FROM CT_DonDH CTDH INNER JOIN GioHang GH ON CTDH.MaGio = GH.MaGio INNER JOIN SanPham SP ON GH.MaSP = SP.MASP
LEFT JOIN CT_KhuyenMai CTKM ON SP.MaSP = CTKM.MASP LEFT JOIN KhuyenMai KM ON CTKM.MaDotKM = KM.MaDotKM
WHERE CTDH.MADDH = @MADDH
SELECT @TONGTIEN = SUM(THANHTIEN) FROM #TEMP
UPDATE HoaDon SET TongTien=@TONGTIEN, ThoiGian=GETDATE(), MaNV = @MANV WHERE MAHD = @MAHD
END

```

Trước khi chạy:

	TongTien	ThoiGian	MaNV
3	1800000.00	2024-05-30 ..	2
4	6700000.00	2024-06-01 ..	1
5	2000000.00	2024-07-02 ..	3
6	5400000.00	2024-07-12 ..	1
7	100000.0000	2024-05-30 ..	1
9	2100000.00	2024-05-30 ..	1
14	0.0000	NULL	NULL
*	NULL	NULL	NULL

	MaCTDDH	MaGio	MaDDH	SoLuong	MaHD
1	1	2	1	3	
2	3	3	1	4	
3	6	3	2	4	
4	1	5	1	5	
5	10	6	1	6	
6	11	6	1	6	
7	8	7	10	7	
9	10	10	1	9	
10	2	17	1	14	
*	NULL	NULL	NULL	NULL	NULL

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int

EXEC    @return_value = [dbo].[SP_TinhTienHoaDonTheoMa]
        @MADDH = 17,
        @MANV = 4

SELECT  'Return Value' = @return_value
GO

```

	TongTien	ThoiGian	MaNV
3	1800000.00	2024-05-30 ..	2
4	6700000.00	2024-06-01 ..	1
5	2000000.00	2024-07-02 ..	3
6	5400000.00	2024-07-12 ..	1
7	100000.0000	2024-05-30 ..	1
9	2100000.00	2024-05-30 ..	1
14	200000.00	2024-05-31 ..	4
*	NULL	NULL	NULL

w. Xem các đơn hàng của tài khoản Khách hàng.

```
CREATE PROC SP_XemDonHangKhachHang
```

```

@username NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;
    IF EXISTS (SELECT 1 FROM TK_KhachHang WHERE usernameKH = @username)
    BEGIN
        SELECT DISTINCT DonDH.MaDDH, @username, Ho, Ten, TG_Dat, TrangThai, HinThuc,
        ThanhToan
        FROM (SELECT MaKH, Ho, Ten FROM KhachHang) KH
        JOIN (SELECT usernameKH, MaKH FROM TK_KhachHang) TKKH
        ON KH.MaKH = TKKH.MaKH
        JOIN (SELECT MaGio, usernameKH, MaSP FROM GioHang) GH
        ON TKKH.usernameKH = GH.usernameKH
        JOIN (SELECT MaSP, TenSP, Gia FROM SanPham) SP
        ON GH.MaSP = SP.MaSP
        JOIN (SELECT MaDDH, MaCTDDH, MaGio FROM CT_DonDH) CTDDH
        ON GH.MaGio = CTDDH.MaGio
        JOIN DonDH
        ON CTDDH.MaDDH = DonDH.MaDDH
        WHERE TKKH.usernameKH = @username
    END
    ELSE
    BEGIN
        RAISERROR('Không tồn tại tài khoản này.', 16, 1);
    END
END

```

Sau khi chạy:

	MaDDH	(No column name)	Ho	Ten	TG_Dat	TrangThai	HinThuc	ThanhToan
1	2	mie239	Nguyễn Thị Huyền	My	2024-05-30 14:50:00.000	Đã hoàn thành	Online	Chuyển khoản
2	5	mie239	Nguyễn Thị Huyền	My	2024-07-02 00:00:00.000	Đã hoàn thành	Offline (không giao hàng)	Chuyển khoản
3	17	mie239	Nguyễn Thị Huyền	My	2024-05-31 05:57:47.670	Đang giao hàng	Online	Chuyển khoản

#### x. Tính tổng doanh thu theo năm.

```

CREATE PROC SP_TongDoanhThuTheoNam
@nam INT
AS
BEGIN
    SELECT DOANHTHU = SUM(TongTien)
    FROM HoaDon
    WHERE YEAR([ThoiGian]) = @nam
END;

```

Sau khi chạy:

```

SQLQuery142.sql - MIE.BGD (sa (57))  ✎ ×
USE [BGD]
GO

DECLARE @return_value int
EXEC    @return_value = [dbo].[SP_TongDoanhThuTheoNam]
        @nam = 2024
SELECT  'Return Value' = @return_value
GO

```

The screenshot shows a SQL query window titled "SQLQuery142.sql - MIE.BGD (sa (57))". The code runs a stored procedure `SP\_TongDoanhThuTheoNam` with parameter `@nam` set to 2024. The result is a single row with column "DOANHThu" containing the value 20100000.00.

	DOANHThu
1	20100000.00

y. Tính tổng doanh thu theo một khoảng thời gian.

```

CREATE PROC SP_TongDoanhThuTheoKhoangTG
    @fromDate DATETIME,
    @toDate DATETIME
AS
BEGIN
    SET NOCOUNT ON;

    SELECT SUM(TongTien) AS TongDoanhThu
    FROM HoaDon
    WHERE ThoiGian BETWEEN @fromDate AND @toDate;
END;

```

Sau khi chạy:

```

SQLQuery143.sql - MIE.BGD (sa (87))  ✎ ×
USE [BGD]
GO

DECLARE @return_value int
EXEC    @return_value = [dbo].[SP_TongDoanhThuTheoKhoangTG]
        @fromDate = N'6/1/2024',
        @toDate = N'6/30/2024'
SELECT  'Return Value' = @return_value
GO

```

The screenshot shows a SQL query window titled "SQLQuery143.sql - MIE.BGD (sa (87))". The code runs a stored procedure `SP\_TongDoanhThuTheoKhoangTG` with parameters `@fromDate` and `@toDate` both set to 'N'6/1/2024'. The result is a single row with column "TongDoanhThu" containing the value 6700000.00.

	TongDoanhThu
1	6700000.00

z. Xem doanh thu top các sản phẩm bán chạy nhất.

```
CREATE PROC SP_XemDoanhThuTopCacSanPhamBanChayNhat
@top TINYINT
AS
BEGIN
    WITH ProductSales AS (
        SELECT
            SP.MaSP,
            SP.TenSP,
            CTDDH.SoLuong,
            SP.Gia AS DonGia,
            CASE
                WHEN SP.MaSP IN (SELECT MaSP FROM CT_KhuyenMai) AND HD.ThoiGian BETWEEN
KM.ThoiGian_BD AND KM.ThoiGian_KT
                    THEN ISNULL(CTKM.PhanTramKM, 0) * CTDDH.SoLuong * SP.Gia / 100
                ELSE 0
            END AS TienGiamGia,
            CTDDH.SoLuong * SP.Gia AS TongTienTruocGiamGia
        FROM SanPham SP
        INNER JOIN GioHang GH ON SP.MaSP = GH.MaSP
        INNER JOIN CT_DonDH CTDDH ON GH.MaGio = CTDDH.MaGio
        LEFT JOIN CT_KhuyenMai CTKM ON SP.MaSP = CTKM.MaSP
        LEFT JOIN KhuyenMai KM ON CTKM.MaDotKM = KM.MaDotKM
        INNER JOIN HoaDon HD ON CTDDH.MaHD = HD.MaHD
        WHERE CTDDH.MaHD IS NOT NULL
    )
    SELECT TOP (@top)
        MaSP,
        TenSP,
        SUM(SoLuong) AS SoLuong,
        DonGia,
        SUM(TongTienTruocGiamGia - TienGiamGia) AS TongTien,
        SUM(TongTienTruocGiamGia) AS TongTienTruocGiamGia,
        SUM(TienGiamGia) AS TienGiamGia
    FROM ProductSales
    GROUP BY MaSP, TenSP, DonGia
    ORDER BY TongTien DESC
END
```

Sau khi chạy:

```

USE [BGD]
GO

DECLARE @return_value int
EXEC    @return_value = [dbo].[SP_XemDoanhThuTopCacSanPhamBanChayNhat]
        @top = 3
SELECT  'Return Value' = @return_value
GO

```

The screenshot shows a SQL query window with the following content:

```

130 % ▾
Results Messages

```

	MaSP	TenSP	SoLuong	DonGia	TongTien	TongTienTruocGiamGia	TienGiamGia
1	1	Giày Nike Air Force 1	10	1500000.00	15000000.00	15000000.00	0.00
2	2	Giày Nike Air Force	4	2000000.00	7600000.00	8000000.00	400000.00
3	6	Giày Converse 1970s True Navy	2	2100000.00	4200000.00	4200000.00	0.00

### 3. View

a. Xem các sản phẩm hiện có

```

CREATE VIEW V_XemCacSanPhamHienCo AS
SELECT * FROM SanPham

```

b. Xem doanh thu theo ngày hiện tại

```

CREATE VIEW V_XemDoanhThuTheoNgayHienTai AS
SELECT SUM(TongTien) AS TongDoanhThu
FROM HoaDon
WHERE CONVERT(date, ThoiGian) = CONVERT(date, GETDATE());

```

## VII. PHÂN QUYỀN

### 1. Quản lý:

*Vai trò và Quyền hạn:*

- Nhân viên: Quản lý các thông tin liên quan đến nhân viên trong hệ thống.
- Tạo tài khoản nhân viên: Quản lý có thể tạo tài khoản cho nhân viên mới.
- Tạo tài khoản Quản lý: Quản lý có thể tạo tài khoản khi có Quản lý mới lên thay.
- Chỉ xem tất cả các thông tin về cửa hàng (trừ thông tin của Khách hàng).
- Xem doanh thu và báo cáo.

*Quyền trong SQL Server:*

- db\_datareader: Đọc dữ liệu từ tất cả các bảng trong cơ sở dữ liệu.
- db\_securityadmin: Quản lý các quyền bảo mật, có thể thay đổi quyền của các tài khoản khác.
- db\_accessadmin: Quản lý quyền truy cập các tài khoản vào cơ sở dữ liệu.

Database role name: QuanLy		
Securables:		
Schema	Name	Type
dbo	GioHang	Table
dbo	KhachHang	Table
dbo	SP_ChinhSuaThongTinNhanVien	Stored procedure
dbo	SP_NhanVienDangNhap	Stored procedure
dbo	SP_TaoTKNhanVien	Stored procedure
dbo	SP_TongDoanhThuTheoKhoangTG	Stored procedure
dbo	SP_TongDoanhThuTheoNam	Stored procedure
dbo	SP_XemDoanhThuTopCacSanPhamBan...	Stored procedure
dbo	SP_XemDonDHTheoTrangThai	Stored procedure
dbo	SP_XemTopSanPhamCoSLBanNhiuNhat	Stored procedure
dbo	SP_XoaTaiKhoanNhanVien	Stored procedure
dbo	TK_KhachHang	Table
dbo	V_XemCacSanPhamHienCo	View
dbo	V_XemDoanhThuTheoNgayHienTai	View

### 2. Nhân viên:

*Vai trò và Quyền hạn:*

- Xem và sửa dữ liệu: Có thể xem và chỉnh sửa dữ liệu các bảng Sản phẩm, Loại, Hàng, Thay đổi giá, Khuyến mãi, Chi tiết khuyến mãi, Giao hàng, Nhận bảo hành, Trả bảo hành.
- Tạo tài khoản Khách hàng: Nhân viên có thể tạo tài khoản mới cho khách hàng.

*Quyền trong SQL Server:*

- db\_datareader: Đọc dữ liệu từ một số bảng trong cơ sở dữ liệu.
- db\_datawriter: Ghi dữ liệu một số bảng trong cơ sở dữ liệu.
- db\_accessadmin: Quản lý quyền truy cập các tài khoản vào cơ sở dữ liệu.

Database role name: NhanVien		
Securables:		
Schema	Name	Type
dbo	DanhGia	Table
dbo	GioHang	Table
dbo	KhachHang	Table
dbo	NhanVien	Table
dbo	QTQuenLy	Table
dbo	SP_CapNhatDotKhuyenMai	Stored procedure
dbo	SP_DoiMatKhauTKNhanVien	Stored procedure
dbo	SP_DuyetDonHang	Stored procedure
dbo	SP_GiaoHangTheoDon	Stored procedure
dbo	SP_NhanVienDangNhap	Stored procedure
dbo	SP_TaoDonHang	Stored procedure
dbo	SP_TaoDotKhuyenMai	Stored procedure
dbo	SP_TaoTKKhachHang	Stored procedure
dbo	SP_ThayDoiThongTinSanPham	Stored procedure
dbo	SP_ThemSanPhamMoi	Stored procedure
dbo	SP_ThemSanPhamVaoDotKM	Stored procedure
dbo	SP_TinhTienHoaDonTheoMa	Stored procedure
dbo	SP_XemDonDHTheoTrangThai	Stored procedure
dbo	SP_XoaSanPham	Stored procedure
dbo	TK_KhachHang	Table
dbo	TK_NhanVien	Table
dbo	V_XemCacSanPhamHienCo	View
dbo	V_XemDoanhThuTheoNgayHienTai	View

### 3. Khách hàng:

*Vai trò và quyền hạn:*

- Tạo tài khoản: Khách hàng có thể tự tạo tài khoản mới.
- Thao tác:
  - o Có thể xem danh sách sản phẩm.
  - o Có thể thêm sản phẩm vào giỏ hàng.
  - o Có thể tạo đơn đặt hàng mới.
- Sửa dữ liệu: Khách hàng có thể chỉnh sửa giỏ hàng và thông tin cá nhân của mình.

*Quyền trong SQL Server:*

- db\_datareader: Đọc dữ liệu từ một số bảng trong cơ sở dữ liệu.
- db\_datawriter: Ghi dữ liệu vào một số bảng trong cơ sở dữ liệu.

Database role name: KhachHang

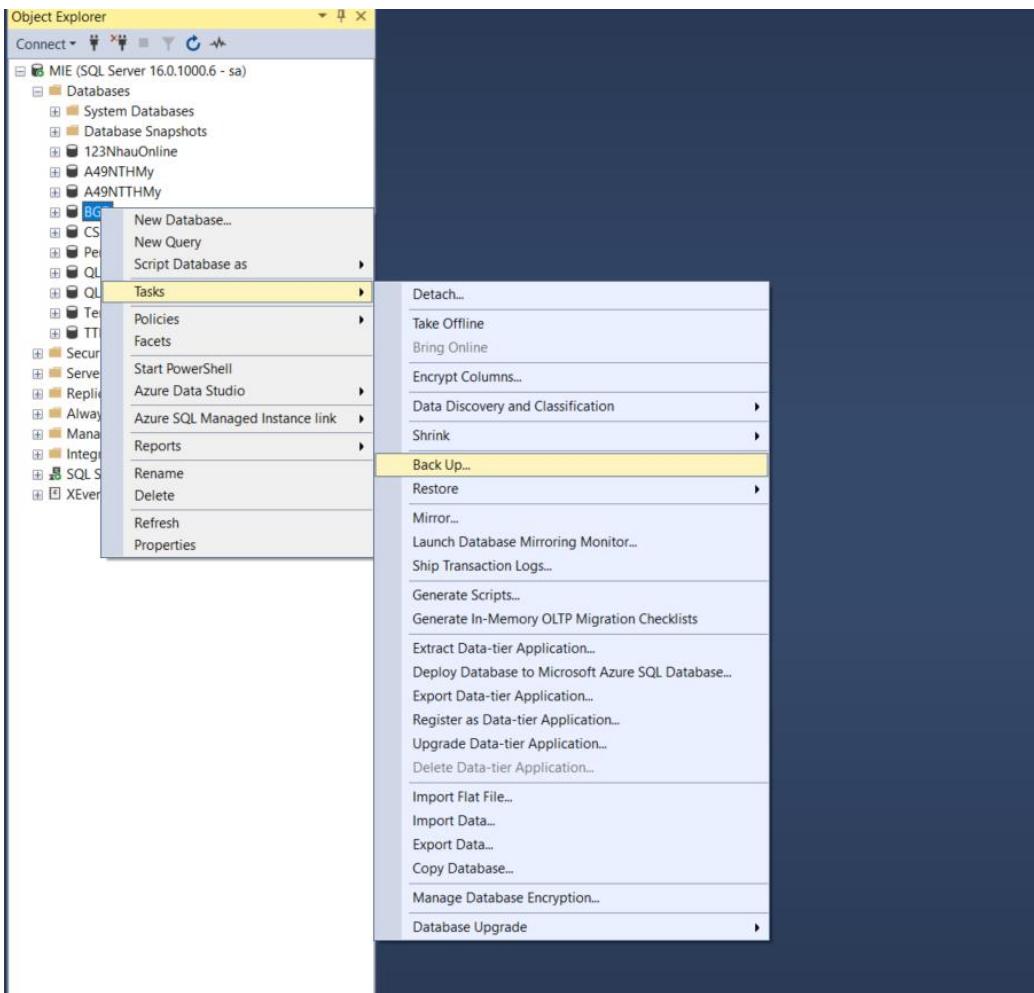
Securables:

Schema	Name	Type
dbo	DanhGia	Table
dbo	GioHang	Table
dbo	KhachHang	Table
dbo	SP_ChinhSuaSoLuongSanPhamTrongGi...	Stored procedure
dbo	SP_ChinhSuaThongTinKhachHang	Stored procedure
dbo	SP_DoiMatKhauTKKhachHang	Stored procedure
dbo	SP_KhachHangDangNhap	Stored procedure
dbo	SP_TaoDonHang	Stored procedure
dbo	SP_TaoTKKhachHang	Stored procedure
dbo	SP_ThemChiTietDonHang	Stored procedure
dbo	SP_ThongTinGioHangTheoTKKhachHang	Stored procedure
dbo	SP_XemDonHangKhachHang	Stored procedure
dbo	TK_KhachHang	Table
dbo	V_XemCacSanPhamHienCo	View

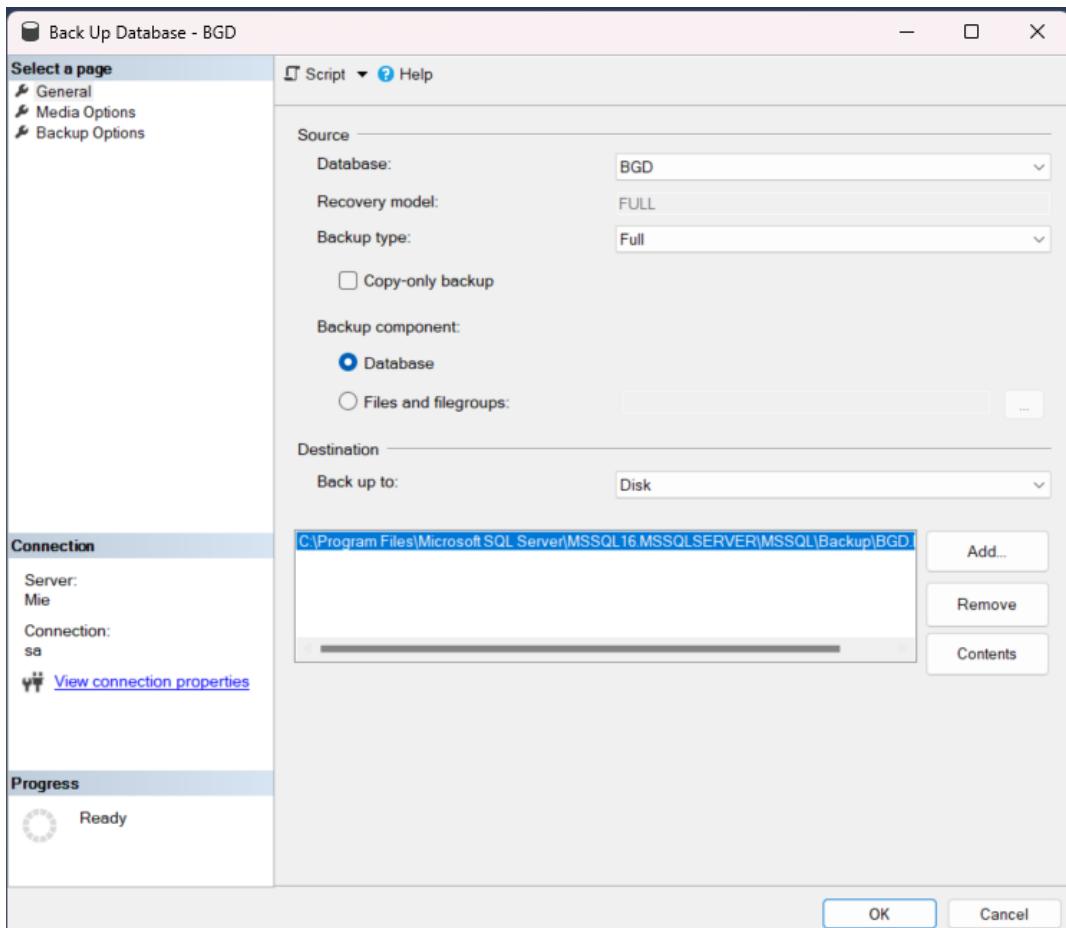
## D. Backup

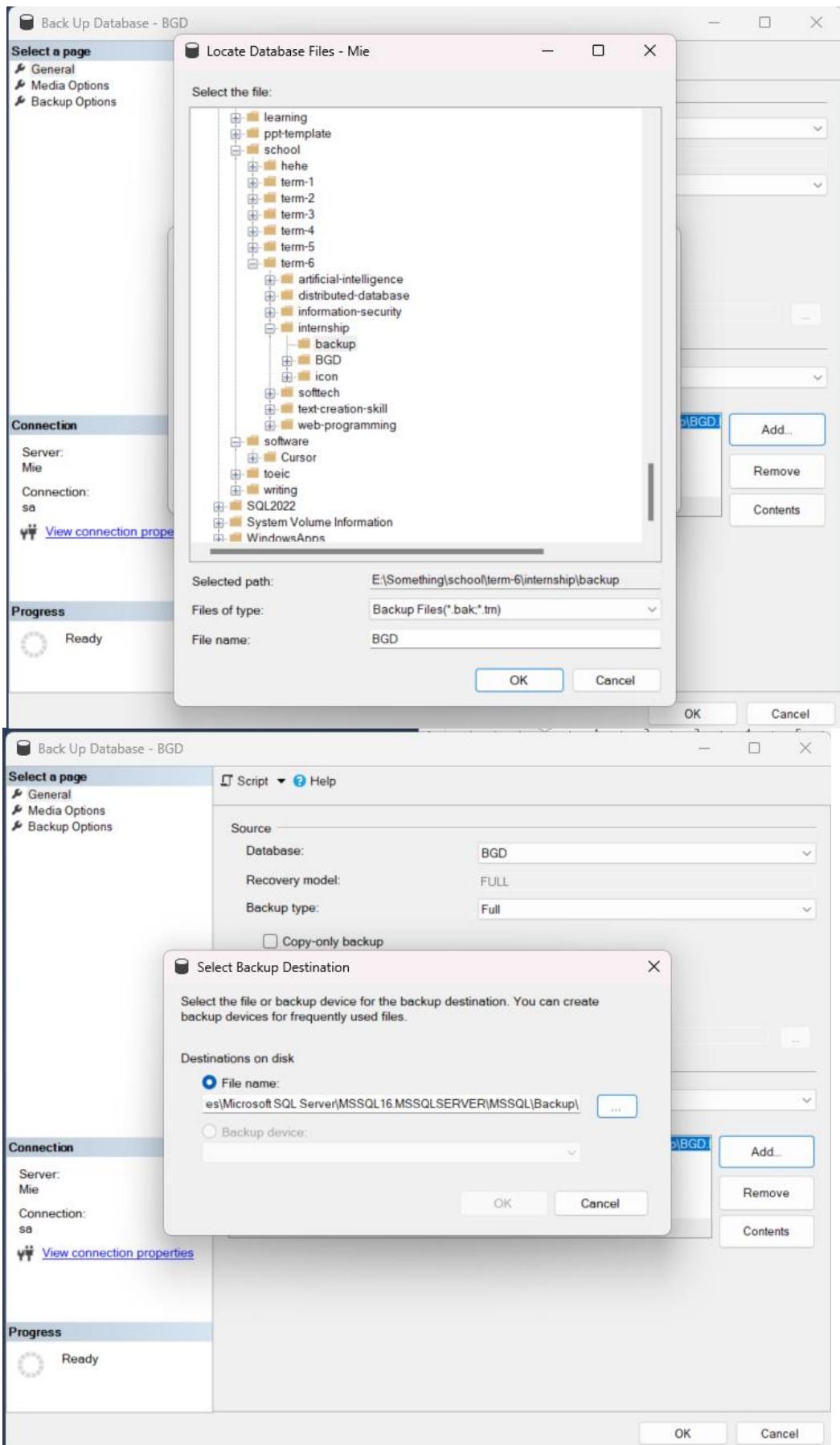
### I. Backup

1. Chọn Cơ sở dữ liệu cần Backup rồi ánhTasks → Backup.



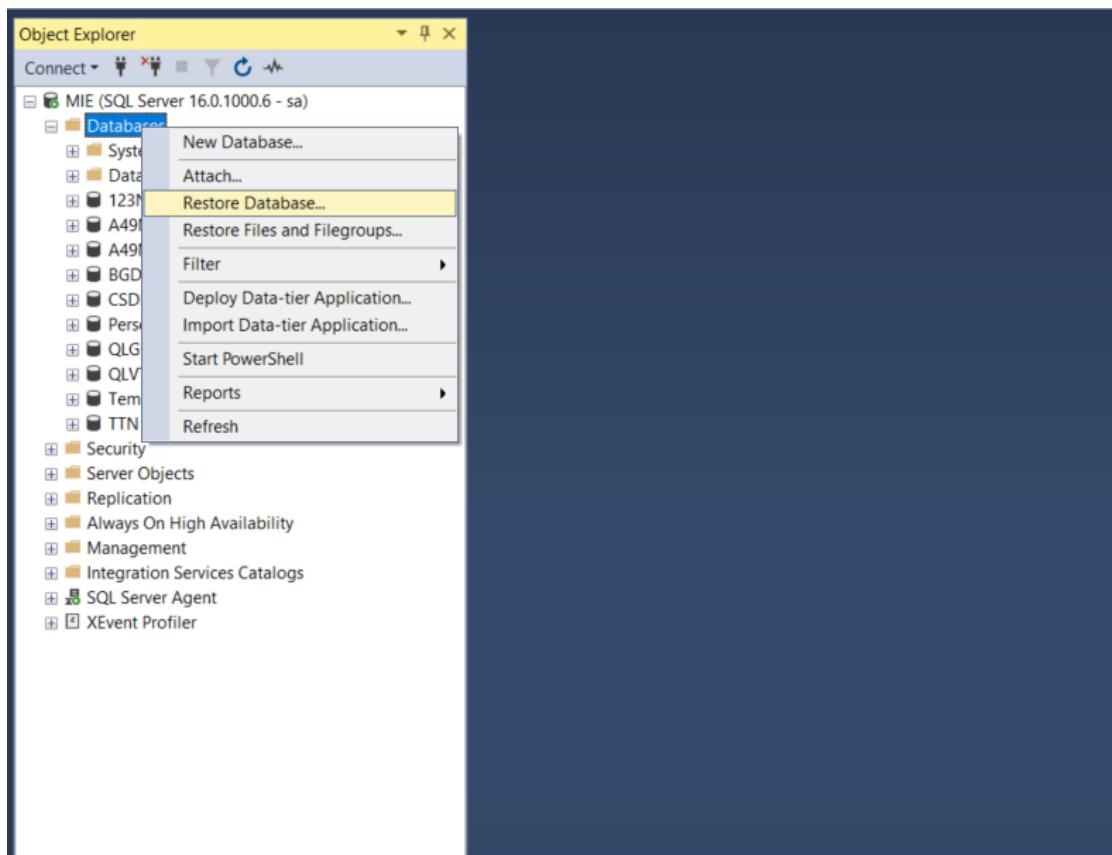
2. Chọn Add để thêm Đường dẫn tới thư mục lưu bản Backup.



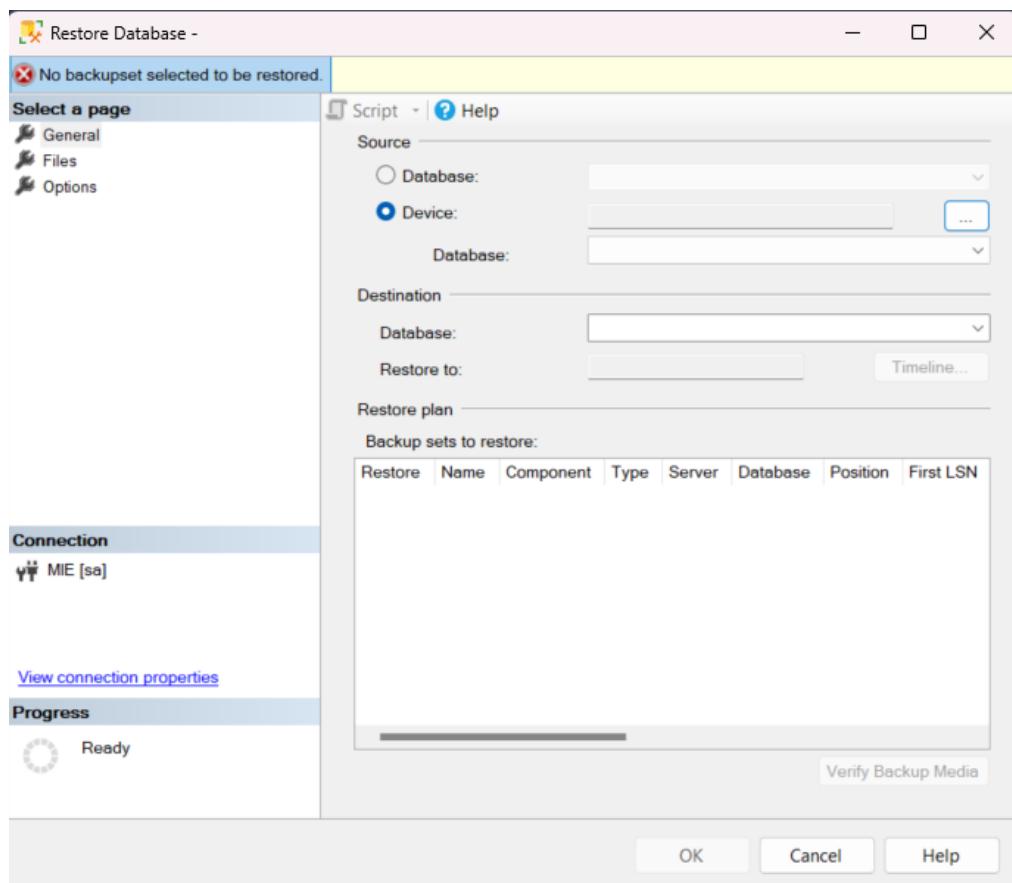


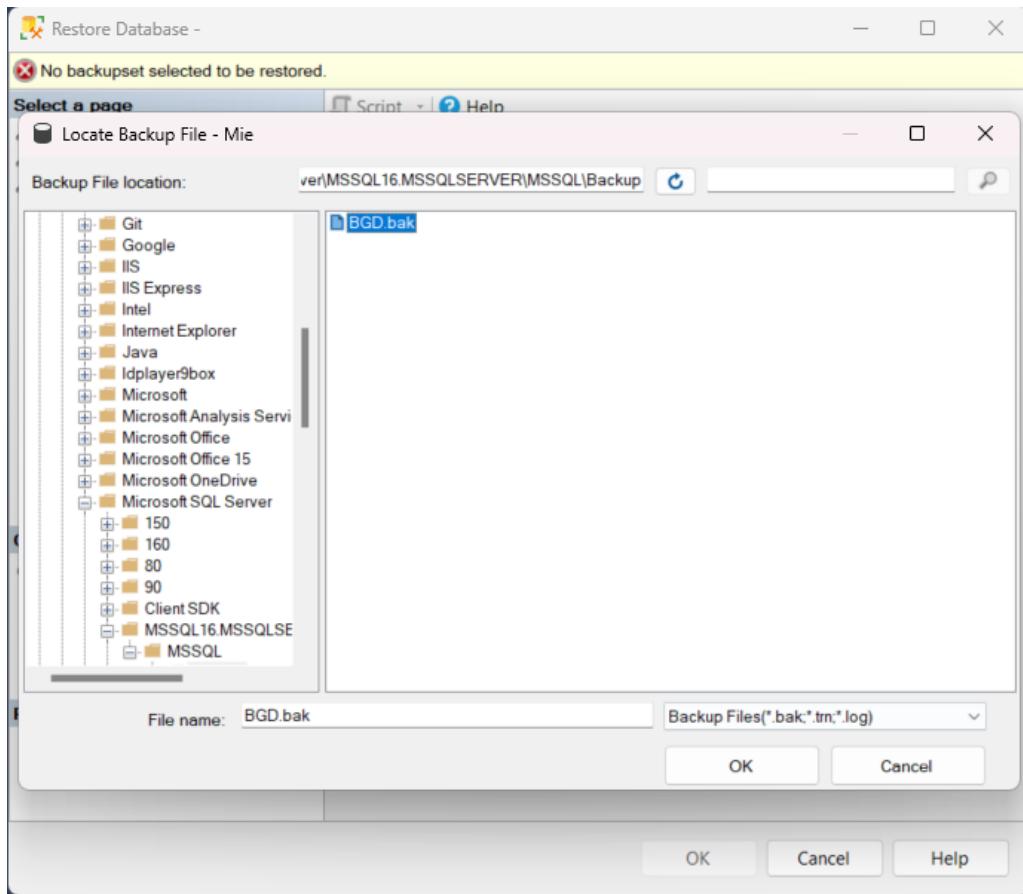
## II. Restore

- Trong SQL Server, nhấp chuột phải vào Databases chọn Restore Database.



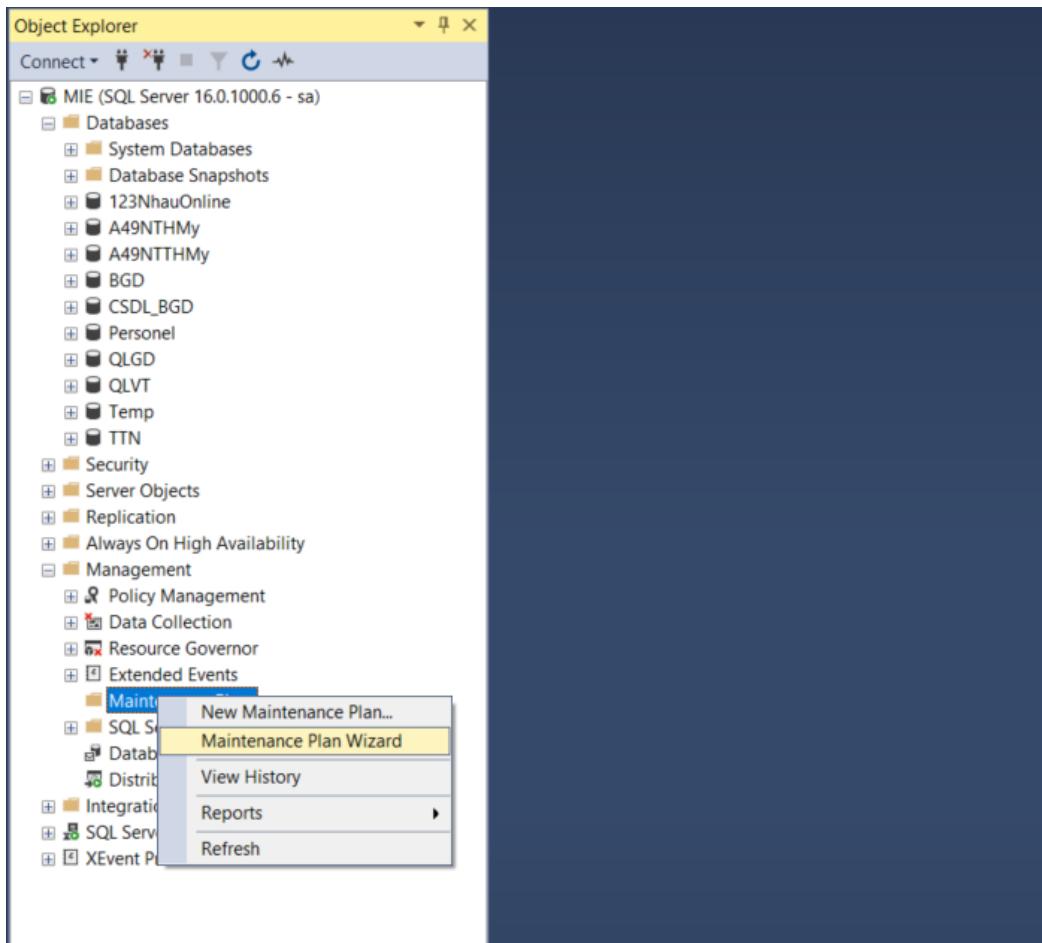
- Chọn Device và ấn Add để chọn file Backup rồi nhấn OK.



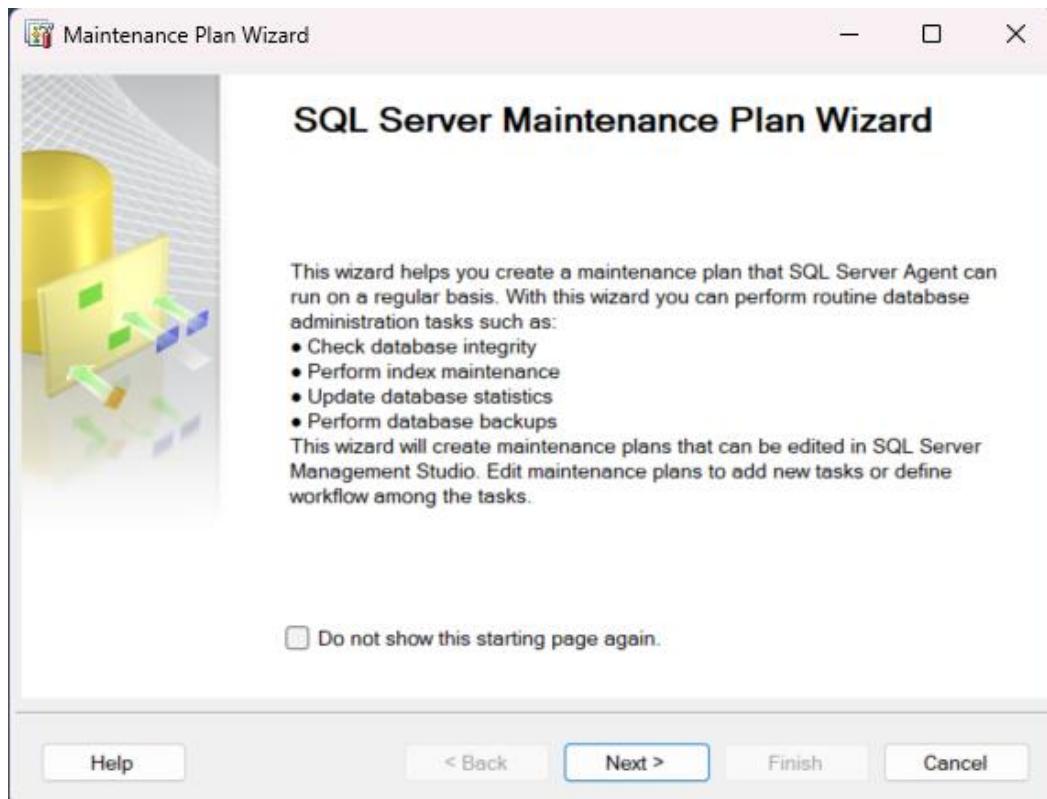


### III. Lập lịch Backup

1. Vào management → Maintenance Plans → Maintenance Plan Wizard



## 2. Bấm Next



## 3. Đặt tên cho Job rồi chọn lịch Backup.

New Job Schedule

Name: Backup Plan

Schedule type: Recurring  Enabled

One-time occurrence

Date: 30/05/2024 Time: 2:54:50 am

Frequency

Occurs: Daily

Recur every: 1 day(s)

Daily frequency

Occurs once at: 7:00:00 am  
 Occurs every: 1 hour(s) Starting at: 12:00:00 am Ending at: 11:59:59 pm

Duration

Start date: 30/05/2024  End date: 30/05/2024  No end date:

Summary

Description: Occurs every day at 7:00:00 am. Schedule will be used starting on 30/05/2024.

OK Cancel Help

#### 4. Chọn Back Up Database (Full)

Maintenance Plan Wizard

Select Maintenance Tasks

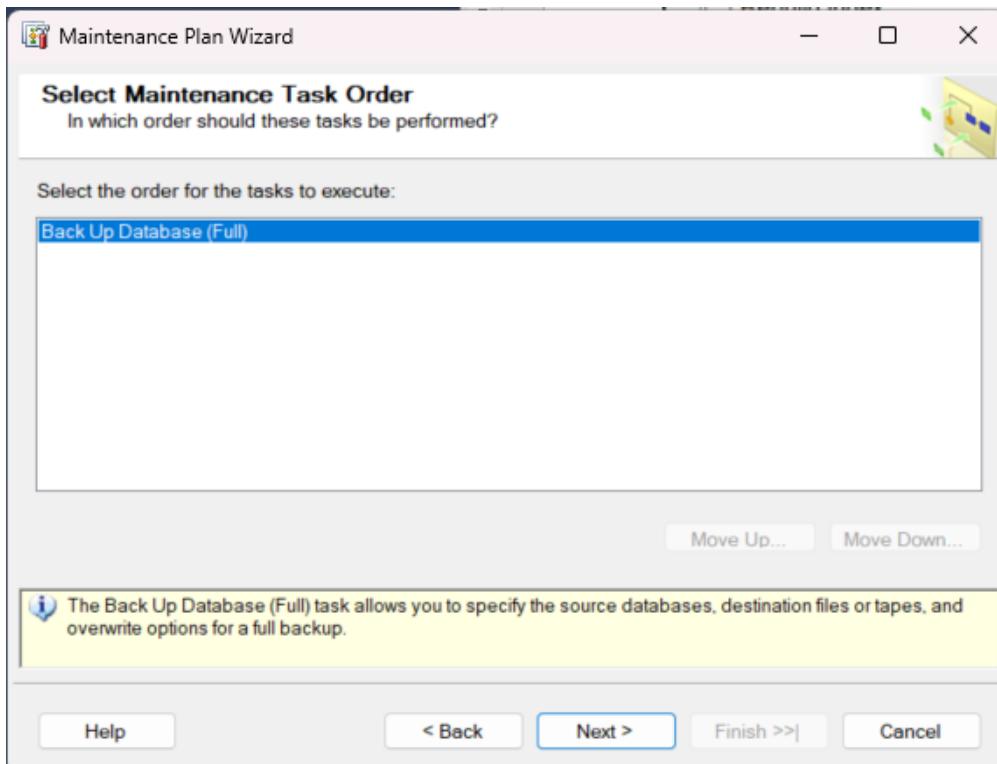
Which tasks should this plan perform?

Select one or more maintenance tasks:

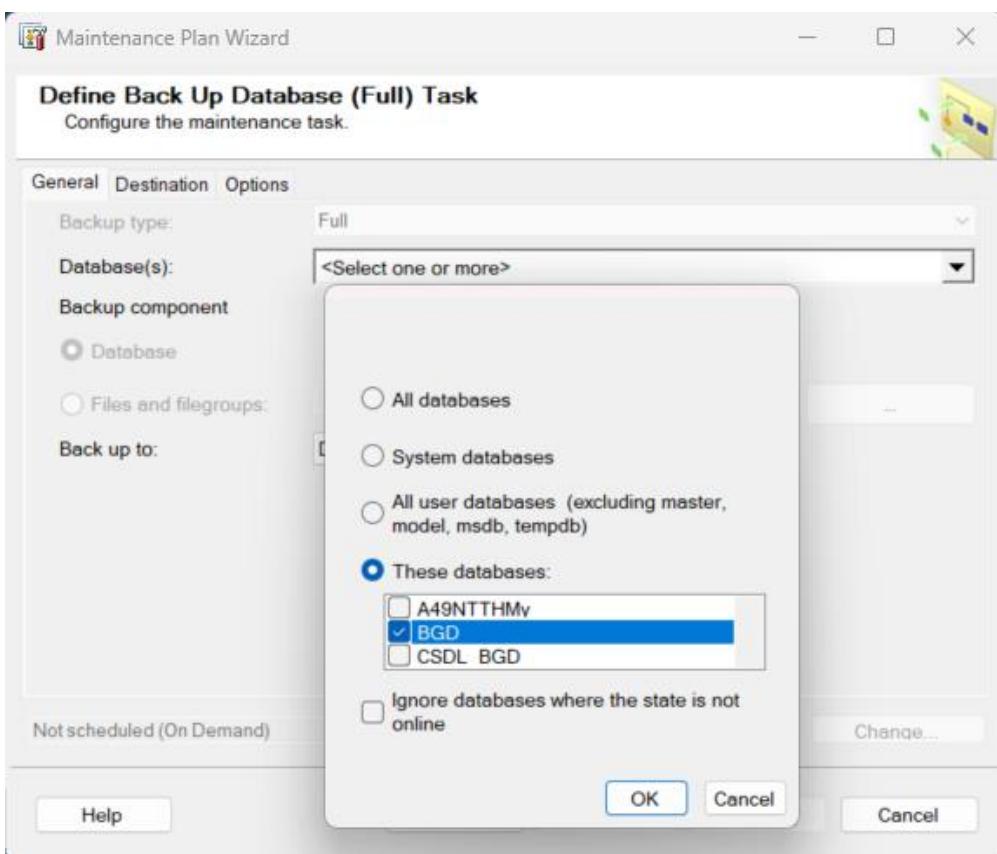
Check Database Integrity  
 Shrink Database  
 Reorganize Index  
 Rebuild Index  
 Update Statistics  
 Clean Up History  
 Execute SQL Server Agent Job  
 Back Up Database (Full)  
 Back Up Database (Differential)  
 Back Up Database (Transaction Log)  
 Maintenance Cleanup Task

**The Back Up Database (Full) task allows you to specify the source databases, destination files or tapes, and overwrite options for a full backup.**

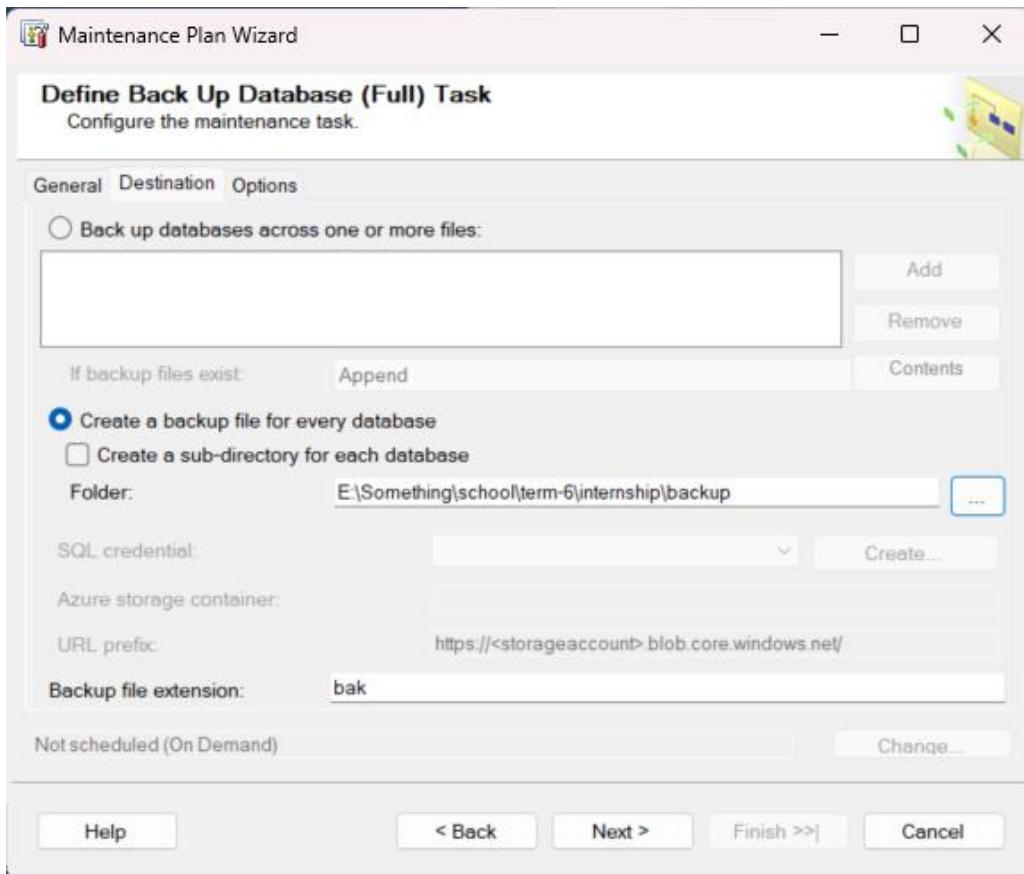
Help < Back Next > Finish Cancel



5. Ở tab General, chọn những Database cần backup.



6. Qua tab Destination để chọn vị trí backup.



7. Chọn vị trí lưu file Report rồi bấm Finish để kết thúc.

