

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN THỰC HÀNH 02

Nhóm

Trần Huy Vũ 18127257

Quách Phú Thành 18127215

Môn học: Cơ sở dữ liệu nâng cao

Thành phố Hồ Chí Minh – 2020

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN THỰC HÀNH 02

| Giáo viên hướng dẫn |

ThS. Hồ Thị Hoàng Vy

Môn học: Cơ sở dữ liệu nâng cao

Thành phố Hồ Chí Minh – 2020

LỜI CẢM ƠN

Để hoàn thành được báo cáo này nhóm em xin gửi lời cảm ơn đến quý thầy cô khoa Công Nghệ Thông Tin trường đại học Khoa Học Tự Nhiên.

Đặc biệt, nhóm xin gửi đến cô Hồ Thị Hoàng Vy, người đã tận tình giúp đỡ chúng em hoàn thành đồ án này, lời cảm ơn chân thành và sâu sắc nhất. Nếu không có những lời hướng dẫn của thầy cô, có lẽ chúng em không thể hoàn thành đồ án tốt như vậy.

Trong quá trình học tập và làm việc, không thể tránh những sai sót, rất mong các thầy cô bỏ qua. Đồng thời, vì lượng kiến thức và kinh nghiệm thực tiễn còn hạn chế nên báo cáo có nhiều thiếu sót, nhóm em rất mong nhận được những góp ý của thầy cô để hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

LỜI CẢM ƠN.....	3
MỤC LỤC	4
MÔ TẢ DỮ LIỆU.....	6
MÔ HÌNH ER.....	7
ĐẶC TẢ CÁC QUAN HỆ	8
1. Bảng NHAN_VIEN.....	8
2. Bảng TAI_KHOAN.....	8
3. Bảng KHACH_HANG	8
4. Bảng VOUCHER	9
5. Bảng HOA_DON	9
6. Bảng GIO_HANG	9
7. Bảng SU_DUNG_VOUCHER	9
8. Bảng CHI_TIET_GIO_HANG	9
9. Bảng MAT_HANG	10
10. Bảng LOAI_HANG	10
11. Bảng NHA_CUNG_UNG	10
12. Bảng THONG_TIN_CUNG_UNG	10
XÉT DẠNG CHUẨN	11
ĐỒ THỊ CON ĐƯỜNG TRUY XUẤT THÔ	13
CÁC CÂU TRUY VẤN.....	14
PHÂN TÍCH GIAO TÁC.....	15
1. Truy vấn 1	16
2. Truy vấn 2.....	17
3. Truy vấn 3.....	20
4. Truy vấn 4.....	22
CÀI ĐẶT CHỨC NĂNG.....	24
1. Thiết kế vật lý.....	24
2. Thiết kế ràng buộc logic	28
LỰA CHỌN CHỈ MỤC	31

MÔ TẢ DỮ LIỆU

Một công ty thương mại điện tử muốn có một cơ sở dữ liệu quản lý toàn bộ quá trình thương mại của họ.

Khách hàng sẽ đăng ký tài khoản bao gồm các thông tin cơ bản như tên tài khoản, mật khẩu, Họ tên, ngày sinh, SDT, địa chỉ, email, giới tính. Sau khi có tài khoản khách hàng có thể bắt đầu mua hàng (Chỉ sau khi đã có tài khoản) bằng cách bỏ lần lượt các mặt hàng vào giỏ hàng.

Giỏ hàng bao gồm nhiều mặt hàng khác nhau với đơn giá, thành tiền từng mặt hàng, phương thức thanh toán, địa chỉ nhận hàng và tổng tiền giỏ hàng. Với từng mặt hàng cần có các tùy chọn như số lượng từng mặt hàng, màu, nhà cung ứng để khách hàng có thể tùy biến. Cần có số lượng voucher áp dụng cho giỏ hàng, tiền ship. Mỗi giỏ hàng chỉ được dùng một code voucher cho mỗi loại voucher.

Cần lưu thông tin các loại hàng để truy xuất cho thao tác tìm kiếm, cần mã loại hàng, tên. Voucher có mã voucher, giá trị giảm giá theo %, loại voucher (ship hay cả đơn). Mỗi voucher có điều kiện áp dụng khác nhau.

Mỗi mặt hàng bao gồm một mã mặt hàng, tên, trọng lượng, xuất xứ, thương hiệu, nhiều màu khác nhau, mô tả. Mặt hàng thuộc một loại hàng và có thể do nhiều nhà cung ứng cung cấp (đồng giá).

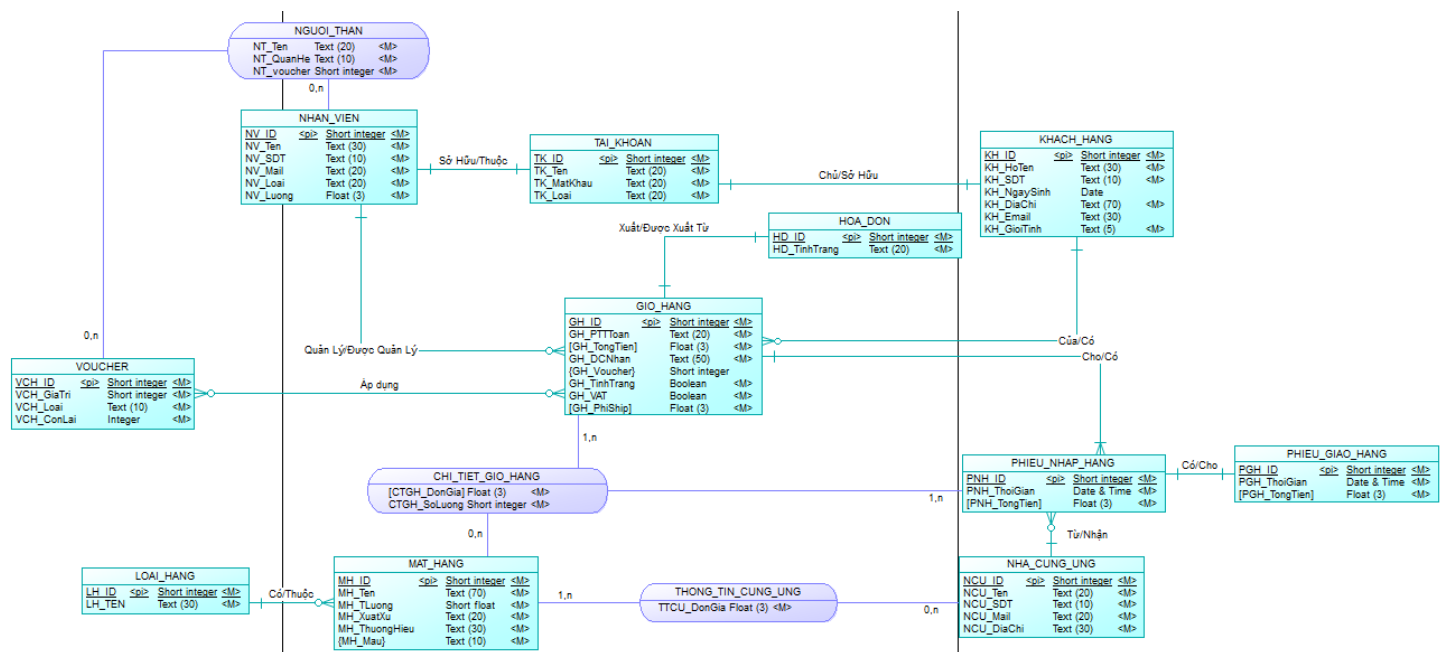
Nhân viên có các thông tin về tên, địa chỉ, SDT, mail, loại nhân viên, lương theo tháng. Nhân viên được cấp một tài khoản riêng cho từng người. Một nhân viên có thể có một người thân (để nhận khuyến mãi) bao gồm các thông tin cơ bản và một mã voucher dành cho người thân được cấp định kỳ.

Sau khi khách hàng xác nhận giỏ hàng, nó sẽ được chuyển cho nhân viên duyệt giỏ hàng và tiến hành xuất hóa đơn, nếu không duyệt sẽ gửi thông báo để khách hàng đổi giỏ hàng. Hóa đơn bao gồm các thông tin của giỏ hàng, tình trạng hóa đơn, có thông tin khách hàng và thông tin nhân viên kiểm duyệt giỏ hàng đó. Trong đó tình trạng hóa đơn sẽ chuyển từ chưa duyệt → duyệt → đang lấy hàng → đã lấy hàng → đang vận chuyển → đã nhận hàng (do nhân viên kiểm duyệt thay đổi).

Nhà cung ứng bao gồm tên nhà cung ứng, địa chỉ, mail, SDT liên lạc. Một nhà cung ứng có thể cung cấp nhiều mặt hàng khác nhau với các mức giá khác nhau.

Mỗi khi có đơn hàng được duyệt nhân viên sẽ xuất phiếu nhập hàng cho bên nhà cung ứng để lấy hàng. Một phiếu lấy hàng chỉ lấy từ một nhà cung ứng và thuộc về một đơn hàng. Khi đã xác nhận thì bên nhà cung ứng bắt đầu giao hàng khi đã tập hợp đầy đủ các mặt hàng trong phiếu nhập. Đơn hàng chỉ được duyệt để vận chuyển khi đã hoàn thành tất cả phiếu nhập thuộc đơn hàng đó.

MÔ HÌNH ER



ĐẶC TẢ CÁC QUAN HỆ

1. Bảng NHAN_VIEN

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	NV_ID	PK	
2	NV_TEN		
3	NV_SDT	unique	
4	NV_MAIL		
5	NV_LOAI		Chỉ có 2 loại nhân viên là nhân viên kiểm duyệt và quản lý
6	NV_GTINH		Chỉ có 2 giới tính là nam hoặc nữ
7	NV_LUONG		
8	NV_IDTK	FK	

2. Bảng TAI_KHOAN

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	TK_ID	PK	Tự động cập nhật thêm mỗi lần có thêm tài khoản mới
2	TK_TEN		
3	TK_MATKHAU		
4	TK_LOAI		Chỉ có 2 loại là Khách hàng và Nhân viên

3. Bảng KHACH_HANG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	KH_ID	PK	Tự động cập nhật thêm mỗi lần có thêm khách hàng mới
2	KH_HOTEN		
3	KH_SDT	unique	
4	KH_NGAYSINH		
5	KH_DIACHI		
6	KH_MAIL		
7	KH_GIOITINH		
8	KH_IDTK	FK	

4. Bảng VOUCHER

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	VCH_ID	PK	
2	VCH_GIATRI		Tính theo %
3	VCH_LOAI		Có 2 loại là trừ vào tiền ship và trừ vào tiền hàng
4	VCH_CONLAI		

5. Bảng HOA_DON

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	HD_ID	PK	
2	HD_TINHTRANG		Tình trạng hiện tại của đơn hàng này
3	HD_IDGH	FK	
4	HD_NGAY		

6. Bảng GIO_HANG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	GH_ID	PK	
2	GH_PTTOAN		
3	GH_TONGTIEN		
4	GH_DCNhan		
5	GH_TINHTRANG		
6	GH_VAT		
7	GH_PHISHIP		
8	GH_IDNV	FK	
9	GH_IDKH	FK	

7. Bảng SU_DUNG_VOUCHER

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	SDVC_IDVC	PK	
2	SDVC_IDGH	PK	
3	SDVC_LOAIVC		Có 2 loại là trừ vào tiền ship và trừ vào tiền hàng
4	SDVC_GIATRI		Tính theo %

8. Bảng CHI_TIET_GIO_HANG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	CTGH_IDGH	PK	
2	CTGH_IDMH	PK	
3	CTGH_DONGIA		
4	CTGH_SOLUONG		

9. Bảng MAT_HANG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	MH_ID	PK	
2	MH_TEN	Not null	
3	MH_XUATXU	Not null	Trọng lượng
4	MH_TLUONG	Not null	
5	MH_THUONGHIEU	Not null	
6	MH_IDLOAI	FK	

10. Bảng LOAI_HANG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	LH_ID	PK	
2	LH_TEN	unique	

11. Bảng NHA_CUNG_UNG

STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	NCU_ID	PK	
2	NCU_TEN		
3	NCU_DIACHI		
4	NCU_SDT	unique	
5	NCU_MAIL		

12. Bảng THÔNG_TIN_CUNG_UNG

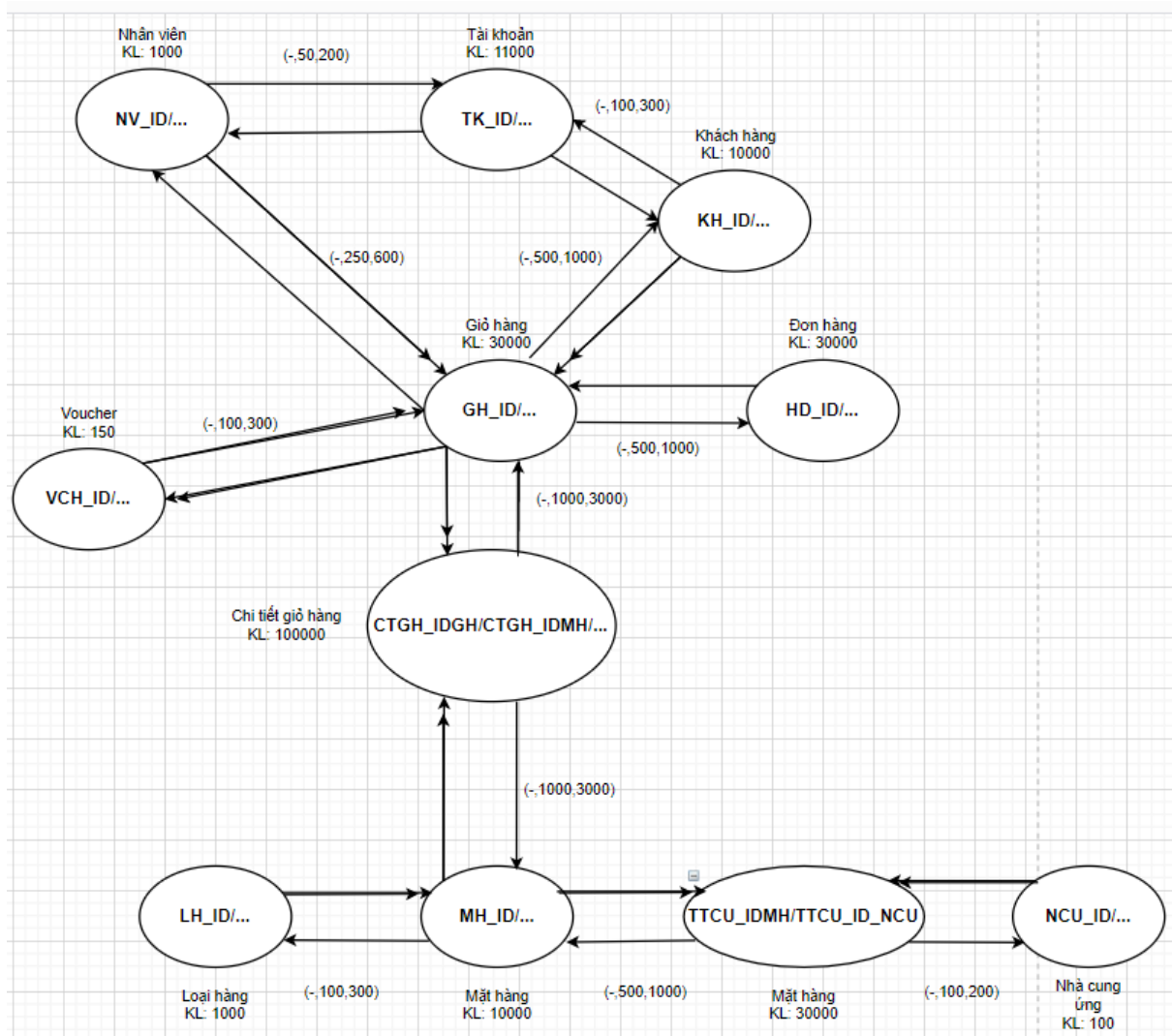
STT	Tên thuộc tính	Ràng buộc	Ý nghĩa/Ghi chú
1	TTCU_ID_NCU	PK	Nhà cung ứng cung cấp hàng
2	TTCU_IDMH	PK	Mặt hàng
3	TTCU_DONGIA	Not nul	

XÉT DẠNG CHUẨN

Quan hệ	Các thuộc tính	Khóa ngoại	Dạng chuẩn
NGUOI_THAN	<u>NV_ID</u> , <u>NT_TEN</u> , <u>NT_QuanHe</u> , <u>NT_voucher</u>	NT_voucher → VOUCHER(VCH_ID)	BCK
NHAN_VIEN	<u>NV_ID</u> , NV_TEN, NV_SDT, NV_Mail, NV_Loai, NV_GTINH, NV_LUONG, <u>NV_IDTK</u>	NV_IDTK → TAI_KHOAN(TK_ID)	DC3 (NV_ID suy ra được TKNV_ID và ngược lại)
TAI_KHOAN	TK_ID, TK_Ten, TK_MatKhau, TK_Loai		BCK
KHACH_HANG	<u>KH_ID</u> , KH_HoTen, KH_SDT, KH_NgaySinh, KH_DiaChi, KH_mail, KH_GioiTinh, <u>KH_IDTK</u>	KH_IDTK → TAI_KHOAN(TK_ID)	DC3 (KH_ID suy ra được TKKH_ID và ngược lại)
VOUCHER	<u>VCH_ID</u> , VCH_GiaTri, VCH_Loai, VCH_ConLai		BCK
HOA_DON	<u>HD_ID</u> , HD_TinhTrang, <u>HD_IDGH</u>	HD_IDGH → GIOHANG(GH_ID)	DC3 (HD_ID suy ra được GH_ID và ngược lại)
GIO_HANG	<u>GH_ID</u> , GH_PTTToan, [GH_TongTien], GH_DCNhan, GH_TinhTrang, GH_VAT, [GH_PhiShip], <u>GH_IDNV</u> , <u>GH_IDKH</u>	GH_IDNV → NHAN_VIEN(NV_ID) GH_IDKH → KHACH_HANG(KH_ID)	BCK
SU_DUNG_VOUC HER	<u>SDVC_IDVC</u> , <u>SDVC_IDGH</u> , SDVC_LoiVC, SDVC_GiaTri	SDVC_IDVC → VOUCHER (VCH_ID) SDVC_IDGH → VOUCHER (GH_ID)	DC1 (SDVC_IDVC là thuộc tính khóa suy ra được SDVC_LoiVC, SDVC_GiaTri)
CHI_TIET_GIO_H ANG	<u>CTGH_IDGH</u> , <u>CTGH_IDMH</u> , CTGH_DONGIA, CTGH_SOLUONG	CTGH_IDGH → GIO_HANG(GH_ID) CTGH_IDMH → MAT_HANG(MH_ID)	DC1 (CTGH_IDMH là thuộc tính khóa → thuộc tính không khóa CTGH_DONGIA)
MAT_HANG	<u>MH_ID</u> , MH_TEN, MH_XUATXU, MH_TLUONG, MH_THUONGHIE U, <u>MH_IDLOAI</u>	MH_IDLOAI → LOAI_HANG(LH_ID)	BCK
LOAI_HANG	<u>LH_ID</u> , LH_TEN		BCK
NHA_CUNG_UNG	<u>NCU_ID</u> , NCU_TEN, NCU_DCHI, NCU_SDT, NCU_MAIL		BCK
THONG_TIN_CUN G_UNG	<u>TTCU_ID_NCU</u> , <u>TTCU_IDMH</u> , TTCU_DONGIA	TTCU_ID_NCU → NHA_CUNG_UNG(NCU_I D) TTCU_IDMH → MAT_HANG(MH_ID)	DC1 (Thuộc tính khóa TTCU_IDMH → thuộc tính không khóa TTCU_DONGIA)
PHIEU_NHAP_HA NG	<u>PNH_ID</u> , PNH_THOIGIAN, PNH_TONGTIEN, <u>PNH_IDGH</u> , <u>PNH_IDNCU</u>	PNH_IDGH → GIO_HANG(GH_ID) PNH_IDNCU → NHA_CUNG_UNG(NCU_ID)	BCK

PHIEU_GIAO_HANG	<u>PGH_ID</u> ,PGH_THOIGIAN,PGH_TONGTIEN, <u>PGH_ID_PNH</u>	PGH_ID_PNH →PHIEU_NHAP_HANG(PNH_ID)	BCK
-----------------	--	--	-----

ĐỒ THỊ CON ĐƯỜNG TRUY XUẤT THÔ



CÁC CÂU TRUY VẤN

1. Truy vấn 1

Khách hàng yêu cầu tính tổng tiền cho giỏ hàng có mã là “4” sau khi nhập các chi tiết giỏ hàng

2. Truy vấn 2

Nhân viên hoặc khách hàng xem thông tin chi tiết các mặt hàng của giỏ hàng có mã là “4”

3. Truy vấn 3

Khách hàng tìm những mặt hàng có loại hàng là “Thú cưng”

4. Truy vấn 4

Khách hàng muốn tìm mua những mặt hàng nào đó có giá bé hơn 1 triệu đồng

PHÂN TÍCH GIAO TÁC

QH \ TV	TV1				TV2				TV3				TV4			
	I	R	U	D	I	R	U	D	I	R	U	D	I	R	U	D
NHAN_VIEN						X										
KHACH_HANG																
NHA_CUNG_UNG																
GIO_HANG		X	X			X										
HOA_DON																
CHI_TIET_GIO_HANG		X				X										
THONG TIN_CUNG_U NG														X		
MAT_HANG										X				X		
VOUCHER																
LOAI_HANG										X						
TAI_KHOAN																
SU_DUNG_VOUCHER		X														

1. Truy vấn 1

Truy vấn 1: Khách hàng yêu cầu tính tổng tiền cho giỏ hàng có mã là “4” sau khi nhập các chi tiết giỏ hàng

Tần suất TV: Trung bình 400 lần/h

Cao điểm 1000 lần/h (dịp lễ tết hay có event tặng voucher)

```
create proc Tinh_Tong_Tien
(
    @IDGH int
)
as
begin tran
    begin try
        if exists (select g.GH_ID from GIO_HANG g where g.GH_ID =
@IDGH)
            begin
                declare @Tong money = (select SUM(ct.CTGH_SOLUONG *
ct.CTGH_DONGIA) from GIO_HANG gh join CHI_TIET_GIO_HANG ct on (gh.GH_ID =
ct.CTGH_IDGH) where gh.GH_ID = @IDGH)
                declare @Voucher_ship tinyint = (select S.SDVC_GIATRI
from SU_DUNG_VOUCHER s where s.SDVC_IDGH = @IDGH and s.SDVC_LOAIVC =
'Ship')
                declare @Voucher_Product tinyint = (select S.SDVC_GIATRI
from SU_DUNG_VOUCHER s where s.SDVC_IDGH = @IDGH and s.SDVC_LOAIVC =
'Product')
                if @Voucher_Product is not null
                begin
                    set @Tong = @Tong - (@Tong * @Voucher_Product) /
100
                end
                if @Voucher_ship is not null
                begin
                    set @Tong = @Tong + (select gh.GH_PHISHIP -
((gh.GH_PHISHIP * @Voucher_ship) / 100) from GIO_HANG gh where gh.GH_ID =
@IDGH)
                end
                else
                begin
                    set @Tong = @Tong + (select gh.GH_PHISHIP from
GIO_HANG gh where gh.GH_ID = @IDGH)
                end
                set @Tong = @Tong + (@Tong * (select gh.GH_VAT from
GIO_HANG gh where gh.GH_ID = @IDGH) / 100)
                update GIO_HANG set GH_TONGTIEN = @Tong where GH_ID =
@IDGH
            end
            else
            begin
                print N'Không tìm thấy giỏ hàng để tính tổng tiền'
                rollback tran
                return 1
            end
        end try
    end catch
```

Điều kiện:

g.GH_ID = @IDGH
gh.GH_ID = @IDGH

s.SDVC_IDGH = @IDGH
and s.SDVC_LOAIVC =
'Ship'

s.SDVC_IDGH = @IDGH
and s.SDVC_LOAIVC =
'Product'

Thuộc tính kết:

gh.GH_ID =
ct.CTGH_IDGH

Thuộc tính sx: Không

Thuộc tính gom nhóm:

Không

Các hàm xây dựng: SUM

Thuộc tính cập nhật:

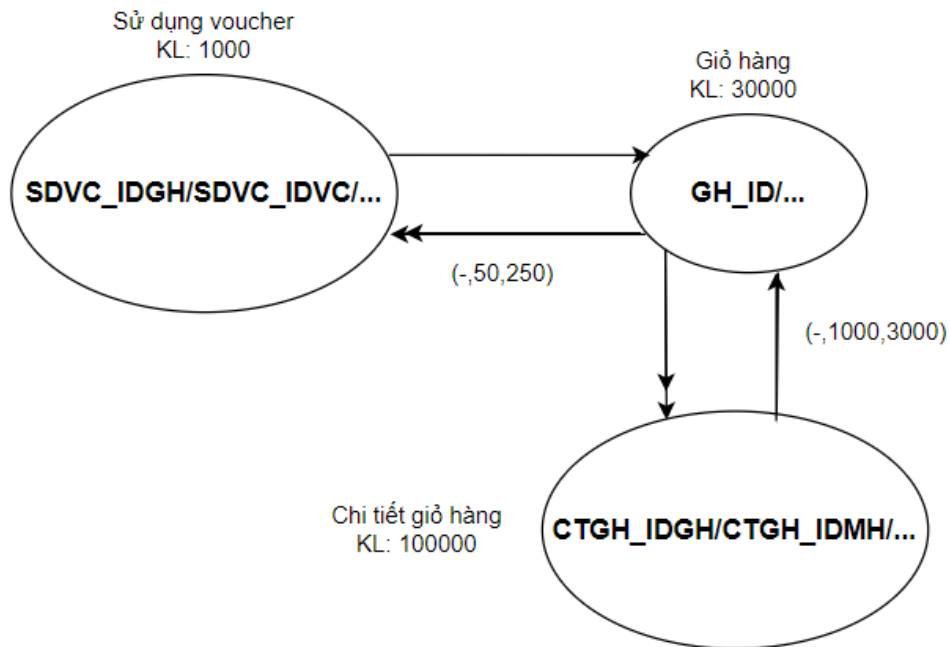
@Tong
@Voucher_ship
@Voucher_Product


```

print N'Lỗi hệ thống'
rollback tran
return 1
end catch
commit tran
return 0

```

Đồ thị con đường truy xuất dữ liệu:



Truy xuất	Quan hệ	Loại truy xuất	Thông số về truy xuất		
			Trên truy vấn	Trung bình (/h)	Cao điểm (/h)
1	GIO_HANG	R	1200 - 3600	480000 - 1440000	1200000 - 3600000
2	GIO_HANG	U	1	400	1000
3	SU_DUNG_VOUCHER	R	40 - 150	16000 - 60000	40000 - 150000
4	CHI_TIET_GIO_HANG	R	4000 - 12000	1600000 - 4800000	4000000 - 12000000
Tổng cộng truy xuất			5241 - 15751	2096400 - 6300400	5241000 - 15751000

Chuỗi kết: SU_DUNG_VOUCHER kết GIO_HANG kết CHI_TIET_GIO_HANG

Chọn ngõ vào là GIO_HANG vì giỏ hàng có thể truy xuất đến

CHI_TIET_GIO_HANG cũng như SU_DUNG_VOUCHER (nếu có). Từ đó giảm phép kết

2. Truy vấn 2

Truy vấn 2: Nhân viên xem thông tin chi tiết các mặt hàng của giỏ hàng có mã là “4”

Tần suất TV: Trung bình 100 lần/h
Cao điểm 500 lần/h (dịp lễ tết hay có event tặng voucher)

```
create proc NV_Xem_CT_GioHang
(
    @IDNV int,
    @IDGH int
)
as
begin tran
    begin try
        if exists (select n.NV_ID from NHAN_VIEN n where n.NV_ID =
@IDNV)
            begin
                if exists (select g.GH_ID from GIO_HANG g where g.GH_ID
= @IDGH and g.GH_IDNV = @IDNV)
                    begin
                        select * from CHI_TIET_GIO_HANG ct where
ct.CTGH_IDGH = @IDGH
                    end
                else
                    begin
                        print N'Không tồn tại giỏ hàng này hoặc nhân viên
không có quyền truy cập giỏ hàng này'
                        rollback tran
                        return 1
                    end
            end
        else
            begin
                print N'Không tồn tại nhân viên này'
                rollback tran
                return 1
            end
        end try
    begin catch
        print N'Lỗi hệ thống'
        rollback tran
        return 1
    end catch
commit tran
return 0
```

Điều kiện:

n.NV_ID = @IDNV

g.GH_ID = @IDGH and
g.GH_IDNV = @IDNV

ct.CTGH_IDGH = @IDGH

Thuộc tính kết: Không

Thuộc tính sx: Không

Thuộc tính gom nhóm:

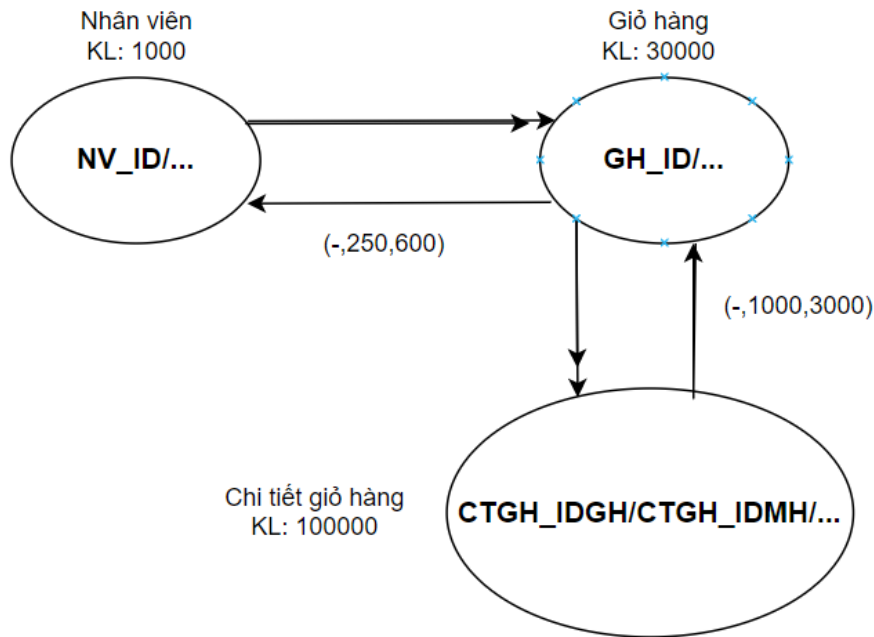
Không

Các hàm xây dựng:

Không

Thuộc tính cập nhật:

Không

Đồ thị con đường truy xuất dữ liệu:

Truy xuất	Quan hệ	Loại truy xuất	Thông số về truy xuất		
			Trên truy vấn	Trung bình (/h)	Cao điểm (/h)
1	GIO_HANG	R	1200 - 3600	120000 – 360000	600000 – 1800000
2	NHAN_VIEN	R	40 - 120	4000 – 12000	20000 – 60000
3	CHI_TIET_GIO_HANG	R	4000 - 12000	400000 - 1200000	2000000 - 6000000
Tổng cộng truy xuất			5240 – 15720	524000 – 1572000	2620000 - 7860000

Chuỗi kết: NHAN_VIEN kết GIO_HANG kết CHI_TIET_GIO_HANG

Chọn ngõ vào là GIO_HANG vì giỏ hàng có thể truy xuất đến

CHI_TIET_GIO_HANG cũng như NHAN_VIEN. Từ đó giảm phép kết.

3. Truy vấn 3

Truy vấn 3: Khách hàng tìm những mặt hàng có loại hàng là thú cưng

Tần suất TV: Trung bình 50 lần/h

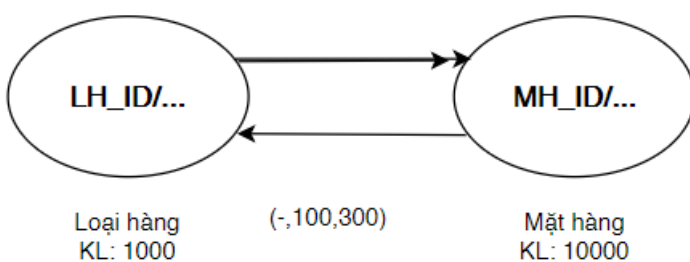
Cao điểm 300 lần/h (dịp sale ở các mặt hàng thú cưng)

```
create proc KH_TimMH_TheoLoaiHang
(
    @TEN_LOAIHANG nvarchar(25)
)
as
begin tran
    begin try
        if not exists (select * from LOAI_HANG where LH_TEN = @TEN_LOAIHANG)
        begin
            print N'Không có mặt hàng nào thuộc loại hàng này'
            rollback tran
            return 1
        end
        select * from MAT_HANG mh join LOAI_HANG lh on (mh.MH_IDLOAI =
lh.LH_ID) where lh.LH_TEN = @TEN_LOAIHANG
    end try
    begin catch
        print N'LỖI HỆ THỐNG'
        rollback tran
        return 1
    end catch
commit tran
return 0
go
```

Điều kiện:
mh.MH_IDLOAI = lh.LH_ID
lh.LH_TEN =
@TEN_LOAIHANG

Thuộc tính kết: Không
Thuộc tính sx: Không
Thuộc tính gom nhóm:
Không
Các hàm xây dựng:
Không
Thuộc tính cập nhật:
Không

Đồ thị con đường truy xuất dữ liệu:



Truy xuất	Quan hệ	Loại truy xuất	Thông số về truy xuất		
			Trên truy vấn	Trung bình (/h)	Cao điểm (/h)
1	MAT_HANG	R	500 - 1500	5000 - 15000	15000 - 45000
2	LOAI_HANG	R	50 - 150	500 - 1500	1500 - 4500
Tổng cộng truy xuất			550 - 1650	5500 - 16500	16500 - 49500

Chuỗi kết: MAT_HANG kết với LOAI_HANG, chọn ngõ vào là LOAI_HANG vì số lượng truy vấn ít hơn, từ đó giảm được phép kết

4. Truy vấn 4

Truy vấn 4: Khách hàng muốn tìm những mặt hàng nào đó có giá bé hơn 1 triệu

Tần suất TV: Trung bình 300 lần/h

Cao điểm 600 lần/h (dịp sale ở các mặt hàng, sắm đồ cho các dịp lễ tết)

```
create proc KH_TimMH_TheoGia
(
    @GIA_NHO_NHAT money,
    @GIA_LON_NHAT money
)
as
begin tran
    begin try
        if exists (select *
            from MAT_HANG mh join THONG_TIN_CUNG_UNG ttcu on (mh.MH_ID =
ttcu.TTCU_IDMH)
            where ttcu.TTCU_DONGIA >= @GIA_NHO_NHAT and ttcu.TTCU_DONGIA
<=@GIA_LON_NHAT)
            begin
                select *
                from MAT_HANG mh join THONG_TIN_CUNG_UNG ttcu on (mh.MH_ID =
ttcu.TTCU_IDMH)
                where ttcu.TTCU_DONGIA >= @GIA_NHO_NHAT and ttcu.TTCU_DONGIA
<=@GIA_LON_NHAT
                order by ttcu.TTCU_DONGIA
            end
        else
            begin
                print N'Không tìm thấy'
                rollback tran
                return 1
            end
        end try
    begin catch
        print N'LỖI HỆ THỐNG'
        rollback tran
        return 1
    end catch
commit tran
return 0
go
```

Điều kiện:

mh.MH_ID =
ttcu.TTCU_IDMH
ttcu.TTCU_DONGIA >=
@GIA_NHO_NHAT
ttcu.TTCU_DONGIA
<=@GIA_LON_NHAT

Thuộc tính kết: Không

Thuộc tính sx:
ttcu.TTCU_DONGIA

Thuộc tính gom nhóm:

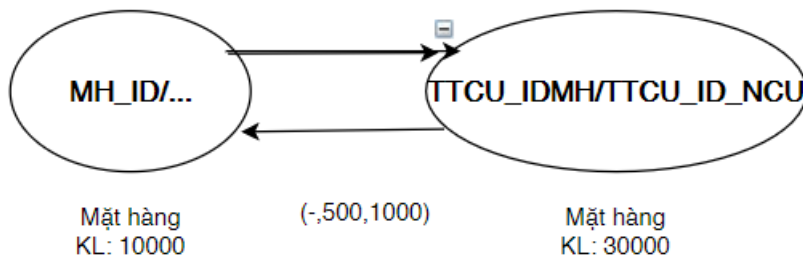
Không

Các hàm xây dựng:

Không

Thuộc tính cập nhật:

Không

Đồ thị con đường truy xuất dữ liệu:

Truy xuất	Quan hệ	Loại truy xuất	Thông số về truy xuất		
			Trên truy vấn	Trung bình (/h)	Cao điểm (/h)
1	MAT_HANG	R	500 - 1500	250000 - 500000	750000 - 1500000
2	THONG_TIN_CUNG_UNG	R	1500 - 4500	750000 - 1500000	2250000 - 6750000
Tổng cộng truy xuất			2000 - 6000	1000000 - 2000000	3000000 - 8250000

Chuỗi kết: MAT_HANG kết với THONG_TIN_CUNG_UNG, chọn ngõ vào là MAT_HANG vì số lượng truy vấn ít hơn, từ đó giảm được phép kết

CÀI ĐẶT CHỨC NĂNG

- Tính tổng tiền cho giỏ hàng sau khi khách hàng đã thêm đầy đủ chi tiết giỏ hàng
- Nhân viên xem thông tin chi tiết các mặt hàng của giỏ hàng có mã xác định nào đó
- Khách hàng tìm kiếm mặt hàng theo loại hàng nhập vào
- Khách hàng tìm kiếm mặt hàng nằm trong 1 khoảng giá nhập vào

1. Thiết kế vật lý

a) NHAN_VIEN

Thiết kế vật lý cho bảng nhân viên
<pre>CREATE TABLE NHAN_VIEN (NV_ID int, NV_TEN nvarchar(50), NV_SDT char(10) unique, NV_MAIL text, NV_LOAI nvarchar(30), NV_GTINH nvarchar(8), NV_LUONG float(3), NV_IDTK int, PRIMARY KEY (NV_ID)) ALTER TABLE NHAN_VIEN ADD CONSTRAINT FK_NHANVIEN_TAIKHOAN_IDTK FOREIGN KEY (NV_IDTK) REFERENCES TAI_KHOAN(TK_ID) ON DELETE SET NULL</pre>

b) TAI_KHOAN

Thiết kế vật lý cho bảng tài khoản
<pre>CREATE TABLE TAI_KHOAN (TK_ID int, TK_TEN text, TK_MATKHAU text, TK_LOAI nvarchar(20), PRIMARY KEY (TK_ID))</pre>

c) KHACH_HANG

Thiết kế vật lý cho bảng khách hàng
<pre>CREATE TABLE KHACH_HANG (KH_ID int, KH_HOTEN nvarchar(50), KH_SDT char(10) not null unique, KH_NGAYSINH date, KH_DIACHI nvarchar(100), KH_MAIL text not null,</pre>


```

        KH_GIOITINH nvarchar(8),
        KH_IDTK int,
        PRIMARY KEY(KH_ID)
    )
ALTER TABLE KHACH_HANG
    ADD CONSTRAINT FK_KHACHHANG_TAIKHOAN_IDTK
    FOREIGN KEY (KH_IDTK) REFERENCES TAI_KHOAN(TK_ID) ON DELETE SET
NULL

```

d) VOUCHER

Thiết kế vật lý cho bảng voucher

```

CREATE TABLE VOUCHER
(
    VCH_ID int,
    VCH_GIATRI tinyint,
    VCH_LOAI nvarchar(20),
    VCH_CONLAI int,
    PRIMARY KEY (VCH_ID)
)

```

e) HOA_DON

Thiết kế vật lý cho bảng hóa đơn

```

CREATE TABLE HOA_DON
(
    HD_ID int,
    HD_TINHTRANG nvarchar(50) default N'Dang lấy hàng',
    HD_IDGH int,
    HD_NGAY date default GETDATE(),
    PRIMARY KEY (HD_ID)
)
ALTER TABLE HOA_DON
    ADD CONSTRAINT FK_HOADON_GIOHANG_IDGH
    FOREIGN KEY (HD_IDGH) REFERENCES GIO_HANG(GH_ID) ON DELETE CASCADE

```

f) GIO_HANG

Thiết kế vật lý cho bảng giỏ hàng

```

CREATE TABLE GIO_HANG
(
    GH_ID int,
    GH_PTTTOAN nvarchar(20),
    GH_TONGTIEN money,
    GH_DCNhan nvarchar(100),
    GH_TINHTRANG nvarchar(50) not null default N'Chưa duyệt',
    GH_VAT tinyint default 10,
    GH_PHISHIP float(3) default 15000,
    GH_IDNV int,
    GH_IDKH int,
    PRIMARY KEY (GH_ID)
)

```

```

ALTER TABLE GIO_HANG
  ADD CONSTRAINT FK_GIOHANG_NHANVIEN_IDNV
  FOREIGN KEY (GH_IDNV) REFERENCES NHAN_VIEN(NV_ID) ON DELETE SET
NULL

ALTER TABLE GIO_HANG
  ADD CONSTRAINT FK_GIOHANG_KHACHHANG_IDKH
  FOREIGN KEY (GH_IDKH) REFERENCES KHACH_HANG(KH_ID) ON DELETE SET
NULL

```

g) SU_DUNG_VOUCHER

Thiết kế vật lý cho bảng sử dụng voucher

```

CREATE TABLE SU_DUNG_VOUCHER
(
  SDVC_IDVC int,
  SDVC_IDGH int,
  SDVC_LOAIVC nvarchar(20),
  SDVC_GIATRI tinyint default 0,
  PRIMARY KEY (SDVC_IDVC, SDVC_IDGH)
)

ALTER TABLE SU_DUNG_VOUCHER
  ADD CONSTRAINT FK_SUDUNGVOUCHER_VOUCHER_IDVC
  FOREIGN KEY (SDVC_IDVC) REFERENCES VOUCHER(VCH_ID)

ALTER TABLE SU_DUNG_VOUCHER
  ADD CONSTRAINT FK_SUDUNGVOUCHER_GIOHANG_IDGH
  FOREIGN KEY (SDVC_IDGH) REFERENCES GIO_HANG(GH_ID)

```

h) CHI_TIET_GIO_HANG

Thiết kế vật lý cho bảng chi tiết giỏ hàng

```

CREATE TABLE CHI_TIET_GIO_HANG
(
  CTGH_IDGH int,
  CTGH_IDMH int,
  CTGH_DONGIA money,
  CTGH_SOLUONG int DEFAULT 1,
  PRIMARY KEY (CTGH_IDGH, CTGH_IDMH)
)

ALTER TABLE CHI_TIET_GIO_HANG
  ADD CONSTRAINT FK_CHITIETGIOHANG_GIOHANG_IDGH
  FOREIGN KEY (CTGH_IDGH) REFERENCES GIO_HANG(GH_ID) ON DELETE
CASCADE

ALTER TABLE CHI_TIET_GIO_HANG
  ADD CONSTRAINT FK_CHITIETGIOHANG_MATHANG_IDMH
  FOREIGN KEY (CTGH_IDMH) REFERENCES MAT_HANG(MH_ID) ON DELETE
CASCADE

```

i) MAT_HANG

Thiết kế vật lý cho bảng mặt hàng

```
CREATE TABLE MAT_HANG
(
    MH_ID int,
    MH_TEN nvarchar(50) not null,
    MH_XUATXU nvarchar(30) not null,
    MH_TLUONG float(3) not null,
    MH_THUONGHIEU nvarchar(35) not null,
    MH_IDLOAI int,
    PRIMARY KEY (MH_ID)
)
ALTER TABLE MAT_HANG
ADD CONSTRAINT FK_MATHANG_LOAIHANG_IDLOAI
FOREIGN KEY (MH_IDLOAI) REFERENCES LOAI_HANG(LH_ID) ON DELETE SET
NULL
```

j) LOAI_HANG

Thiết kế vật lý cho bảng loại hàng

```
CREATE TABLE LOAI_HANG
(
    LH_ID int,
    LH_TEN nvarchar(25) not null unique,
    PRIMARY KEY (LH_ID)
)
```

k) NHA_CUNG_UNG

Thiết kế vật lý cho bảng nhà cung ứng

```
CREATE TABLE NHA_CUNG_UNG
(
    NCU_ID int,
    NCU_TEN nvarchar(50),
    NCU_DCHI nvarchar(100),
    NCU_SDT char(10) unique,
    NCU_MAIL text,
    PRIMARY KEY (NCU_ID)
)
```

l) THONG_TIN_CUNG_UNG

Thiết kế vật lý cho bảng thông tin cung ứng

```
CREATE TABLE THONG_TIN_CUNG_UNG
(
    TTCU_ID_NCU int,
    TTCU_IDMH int,
    TTCU_DONGIA money not null,
    PRIMARY KEY (TTCU_ID_NCU, TTCU_IDMH)
)
ALTER TABLE THONG_TIN_CUNG_UNG
ADD CONSTRAINT FK_THONGTINCUNGUNG_NHACUNGUNG_IDNCU
FOREIGN KEY (TTCU_ID_NCU) REFERENCES NHA_CUNG_UNG(NCU_ID) ON DELETE
CASCADE
ALTER TABLE THONG_TIN_CUNG_UNG
```

```
ADD CONSTRAINT FK_THONGTINCUNGUNG_MATHANG_IDMH  
FOREIGN KEY (TTCU_IDMH) REFERENCES MAT_HANG(MH_ID) ON DELETE  
CASCADE
```

2. Thiết kế ràng buộc logic

- a) Ràng buộc 1: Giới tính của nhân viên và khách hàng chỉ có thể là nam hoặc nữ

```
ALTER TABLE NHAN_VIEN  
  
ADD CONSTRAINT CK_NHANVIEN_GIOITINH  
  
CHECK (NV_GTINH = N'Nam' or NV_GTINH = N'Nữ')
```

```
ALTER TABLE KHACH_HANG  
  
ADD CONSTRAINT CK_KHACHHANG_GIOITINH  
  
CHECK (KH_GIOITINH = N'Nam' or KH_GIOITINH = N'Nữ')
```

- b) Ràng buộc 2: Ngày sinh của khách hàng phải trước ngày hiện nay của hệ thống

```
ALTER TABLE KHACH_HANG  
  
ADD CONSTRAINT CK_KHACHHANG_NGAYSINH  
  
CHECK (KH_NGAYSINH < GETDATE())
```

- c) Ràng buộc 3: Voucher luôn có giá trừ từ 0 đến 100 (mang ý nghĩa %)

```
ALTER TABLE VOUCHER  
  
ADD CONSTRAINT CK_VOUCHER_GIATRI  
  
CHECK (VCH_GIATRI > 0 and VCH_GIATRI <= 100)
```

- d) Ràng buộc 4: Số lượng voucher còn lại luôn phải lớn hơn 0

```
ALTER TABLE VOUCHER  
  
ADD CONSTRAINT CK_VOUCHER_CONLAI  
  
CHECK (VCH_CONLAI >= 0)
```

- e) Ràng buộc 5: Giỏ hàng chỉ có 2 tình trạng là chưa duyệt và đã duyệt, người có quyền xét tình trạng duyệt hay chưa là nhân viên

```
ALTER TABLE GIO_HANG  
  
ADD CONSTRAINT CK_GIOHANG_TINHTRANG  
  
CHECK (GH_TINHTRANG = N'Đã duyệt' or GH_TINHTRANG = N'Chưa duyệt')
```

- f) Ràng buộc 6: Tổng tiền của tất cả sản phẩm trong giỏ hàng phải luôn lớn hơn 0

```
ALTER TABLE GIO_HANG
ADD CONSTRAINT CK_GIOHANG_TONGTIEN
CHECK (GH_TONGTIEN > 0)
```

- g) Ràng buộc 7: Voucher chỉ có 2 dạng là ship (giảm vào tiền ship) hoặc product (giảm vào tiền sản phẩm)

```
ALTER TABLE SU_DUNG_VOUCHER
ADD CONSTRAINT CK_SUDUNGVOUCHER_LOAIVC
CHECK (SDVC_LOAIVC = 'Ship' or SDVC_LOAIVC = 'Product')
```

- h) Ràng buộc 8: Trong bảng CHI_TIET_GIO_HANG thì đơn giá và số lượng phải luôn lớn hơn 0

```
ALTER TABLE CHI_TIET_GIO_HANG
ADD CONSTRAINT CK_CTGIOHANG_DONGIA
CHECK (CTGH_DONGIA > 0)
```

```
ALTER TABLE CHI_TIET_GIO_HANG
ADD CONSTRAINT CK_CTGIOHANG_SOLUONG
CHECK (CTGH_SOLUONG > 0)
```

- i) Ràng buộc 9: Mặt hàng phải có trọng lượng lớn hơn 0

```
ALTER TABLE MAT_HANG
ADD CONSTRAINT CK_MATHANG_TLUONG
CHECK (MH_TLUONG > 0)
```

- j) Ràng buộc 10: Đơn giá trong bảng THONG_TIN_CUNG_UNG phải lớn hơn 0

```
ALTER TABLE THONG_TIN_CUNG_UNG
ADD CONSTRAINT CK_TTCUNGUNG_DONGIA
CHECK (TTCU_DONGIA > 0)
```

- k) Ràng buộc 10: Ngày xuất hóa đơn phải trước ngày hiện tại theo hệ thống

```
ALTER TABLE HOA_DON
ADD CONSTRAINT CK_HOADON_NGAY
CHECK (HD_NGAY <= GETDATE())
```

- l) Ràng buộc 11: Kiểm tra trong bảng SU_DUNG_VOUCHER với bên bảng VOUCHER có đồng bộ với nhau hay không

```
create function CK_GiaTri_Voucher ()
```

```
returns int
as
begin
    if not exists (select s.SDVC_IDVC from SU_DUNG_VOUCHER s where exists
(select v.VCH_ID from VOUCHER v where v.VCH_ID = s.SDVC_IDVC
and (s.SDVC_LOAIVC != v.VCH_LOAI or s.SDVC_GIATRI != v.VCH_GIATRI)))
    begin
        return 1
    end
    return 0
end;
GO
ALTER TABLE SU_DUNG_VOUCHER
    ADD CONSTRAINT CK_SUDUNGVOUCHER_GIATRI_FK
    CHECK (dbo.CK_GiaTri_Voucher() = 1)
GO
create function CK_SoLuong_Voucher ()
returns int
as
begin
    if ( 2 >= all (select COUNT(SDVC_IDGH) from SU_DUNG_VOUCHER group by
SDVC_IDGH))
        and ( 1 >= all (select COUNT(s.SDVC_IDGH) from SU_DUNG_VOUCHER s where
s.SDVC_LOAIVC = 'Ship' group by s.SDVC_IDGH))
        and ( 1 >= all (select COUNT(s.SDVC_IDGH) from SU_DUNG_VOUCHER s where
s.SDVC_LOAIVC = 'Product' group by s.SDVC_IDGH))
    begin
        return 1
    end
    return 0
end;
GO
ALTER TABLE SU_DUNG_VOUCHER
    ADD CONSTRAINT CK_SUDUNGVOUCHER_SOLUONG
    CHECK (dbo.CK_SoLuong_Voucher () = 1)
```

LỰA CHỌN CHỈ MỤC

1. Lựa chọn cài đặt chỉ mục (index)

STT	Quan hệ	Thuộc tính	Giao tác	Lý do
1	SU_DUNG_VOUCHER	SDVC_GIATRI, SDVC_LOAIVC	(1)	Có tìm kiếm trên 2 thuộc tính
2	LOAI_HANG	LH_TEN	(3)	Sử dụng nhiều
3	THONG_TIN_CUNG_UNG	TTCU_DONGIA	(4)	Sử dụng nhiều, tìm kiếm với nhiều điều kiện, ngoài ra còn có sự sắp xếp từ bé đến lớn

2. Xem xét hiệu năng của việc lựa chọn chỉ mục

a. Trên quan hệ SU_DUNG_VOUCHER

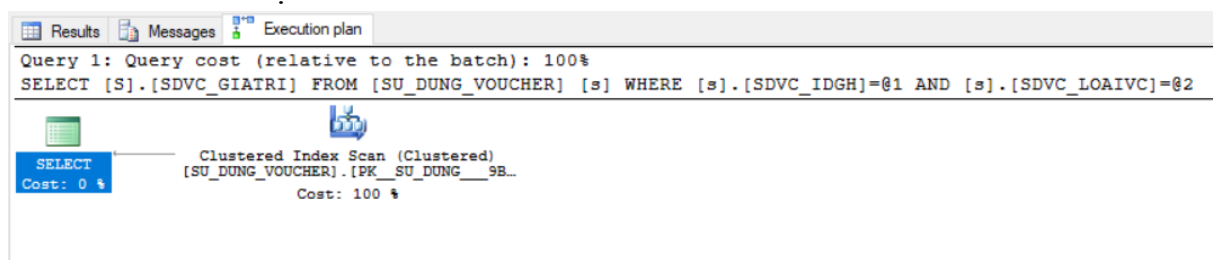
Phương án cài đặt: Cài đặt non-clustered index (composite) trên thuộc tính SDVC_GIATRI và SDVC_LOAIVC vì nó được sử dụng bởi nhiều khách hàng, còn có thể lặp lại bởi cung một khách hàng nên tần suất truy xuất lớn và phát sinh nhiều dòng dữ liệu.

Ngoài ra, còn có chức năng tính tổng tiền phải liên tục truy xuất 3 – 4 lần trên giá trị voucher mỗi 1 giao tác và giao tác này thực thi liên tục khi thêm xóa chi tiết giỏ hàng.

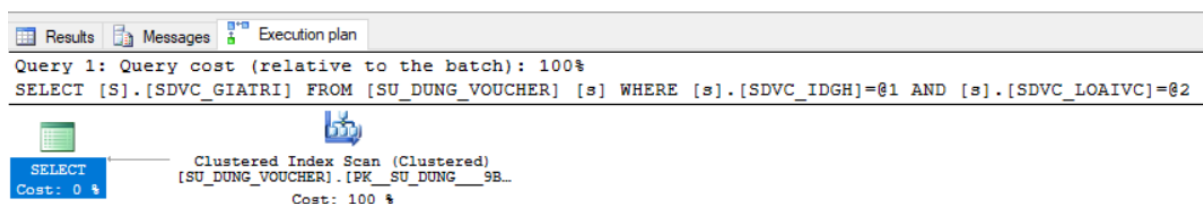
Test hiệu năng trước khi có chỉ mục thông qua truy vấn:

```
select S.SDVC_GIATRI from SU_DUNG_VOUCHER s where s.SDVC_IDGH = 4 and s.SDVC_LOAIVC = 'Product'
```

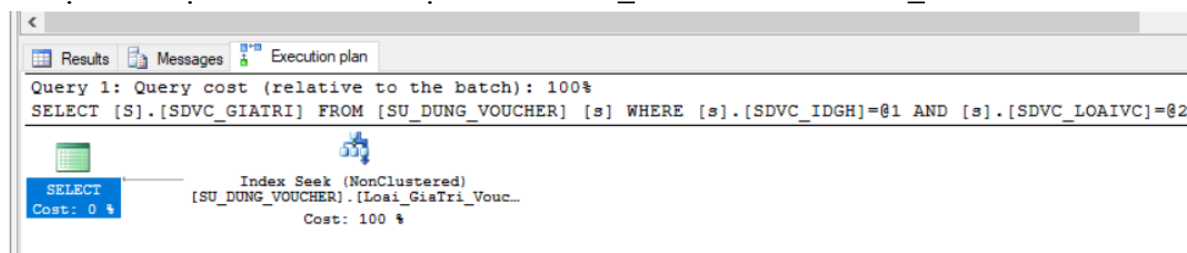
Khi chưa có chỉ mục:



Khi tạo chỉ mục trên mỗi thuộc tính SDVC_LOAIVC



Kết luận: Vậy khi chưa có chỉ mục và tạo chỉ mục trên mỗi thuộc tính SDVC_LOAIVC là không có ý nghĩa, HQT vẫn Scan trên primary key. Khi tạo chỉ mục trên cả hai thuộc tính SDVC_LOAIVC và SDVC_GIATRI



Kết luận: Tạo chỉ mục composite đã có hiệu quả (HQT đã dùng Index Seek) → Giảm thời gian tìm kiếm.

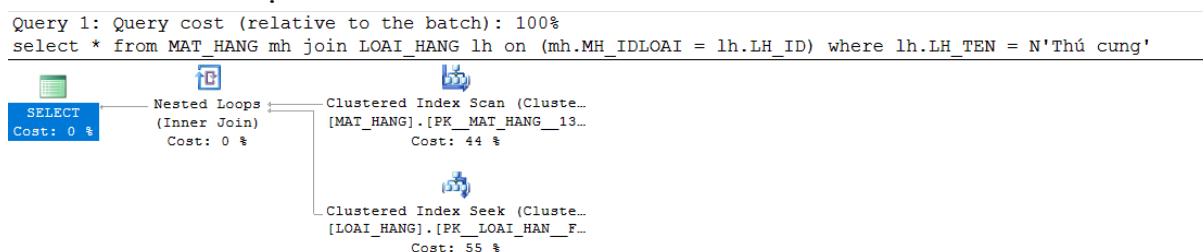
b. Trên quan hệ LOAI_HANG

Phương án cài đặt: Cài đặt non-clustered index (composite) trên thuộc tính LH_TEN vì nó được sử dụng bởi nhiều khách hàng, còn có thể lặp lại bởi cùng một khách hàng nên tần suất truy xuất lớn và phát sinh nhiều dòng dữ liệu.

Test hiệu năng trước khi tạo index thông qua truy vấn sau (truy vấn mang ý nghĩa in ra những mặt hàng có tên loại hàng là 'Thú cưng')

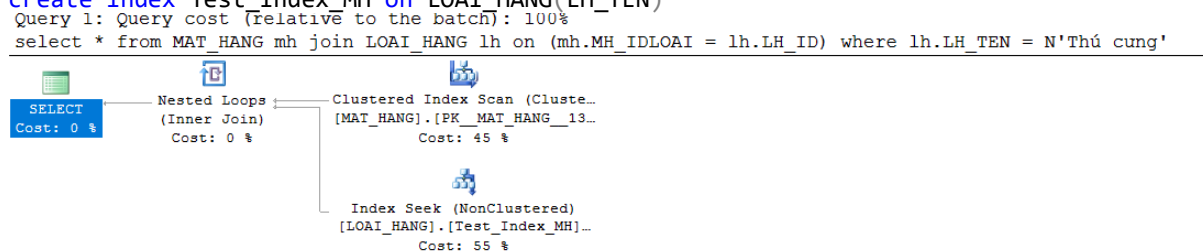
```
select * from MAT_HANG mh join LOAI_HANG lh on (mh.MH_IDLOAI = lh.LH_ID) where lh.LH_TEN = N'Thú cưng'
```

Khi chưa có chỉ mục:



Sau khi tạo chỉ mục với cú pháp sau:

```
create index Test_Index_MH on LOAI_HANG(LH_TEN)
```



Tạo chỉ mục có hiệu quả (HQT đã dùng index seek) → giảm thời gian tìm kiếm

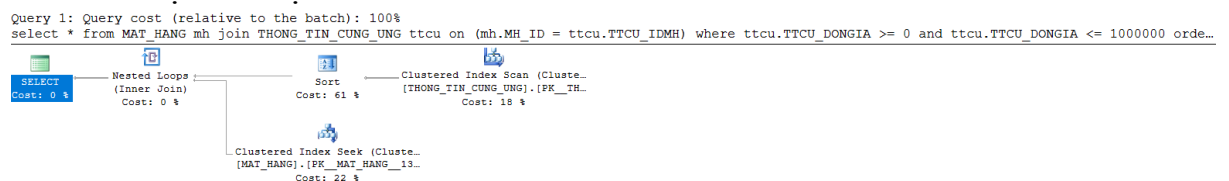
c. Trên quan hệ THÔNG TIN_CUNG_UNG

Phương án cài đặt: Cài đặt non-clustered index (composite) trên thuộc tính TTCU_DONGIA vì nó được khách hàng quan tâm nhiều (kiểm tra mức giá trong khoảng nào đó), còn có thể lặp lại bởi cùng một khách hàng nên tần suất truy xuất lớn và phát sinh nhiều dòng dữ liệu, ngoài ra còn có sự sắp xếp theo đơn giá từ bé đến lớn

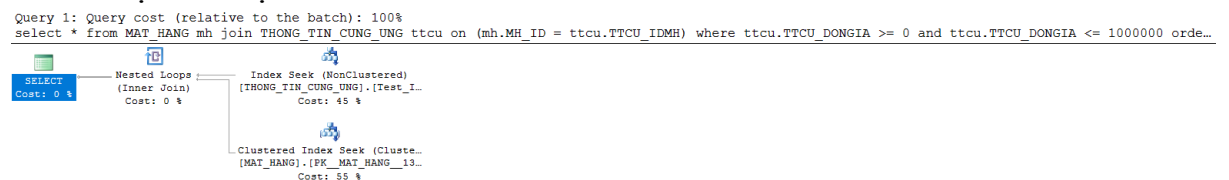
Test hiệu năng trước khi tạo index thông qua truy vấn sau (truy vấn mang ý nghĩa in ra những mặt hàng có giá bé hơn 1 triệu đồng)

```
select *
      from MAT_HANG mh join THONG_TIN_CUNG_UNG ttcu on (mh.MH_ID =
ttcu.TTCU_IDMH)
      where ttcu.TTCU_DONGIA >= 0 and ttcu.TTCU_DONGIA <= 1000000
      order by ttcu.TTCU_DONGIA
```

Trước khi tạo chỉ mục:



Sau khi tạo chỉ mục:



Dễ dàng nhận thấy sự khác biệt lớn giữa việc có tạo chỉ mục hay không, tạo chỉ mục ở trường hợp này là có hiệu quả (index seek, nested loops) → giúp tăng tốc độ tìm kiếm sản phẩm