



Báo-Cáo- CSDLNC - Cơ sở dữ liệu nâng cao cực kì đầy đủ và chi tiết

TH Phân tích dữ liệu (Trường Đại học Công nghệ Thành phố Hồ Chí Minh)

BÁO CÁO MÔN HỌC

PHÂN TÍCH VÀ THIẾT KẾ CSDL QUẢN LÝ CỬA HÀNG BÁN LINH KIỆN PC

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn :

Sinh viên thực hiện :

MSSV:

Họ và tên:

Lớp:

TP. Hồ Chí Minh, 12/2021

BÁO CÁO MÔN HỌC

PHÂN TÍCH VÀ THIẾT KẾ CSDL QUẢN LÝ CỬA HÀNG BÁN LINH KIỆN PC

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn :

Sinh viên thực hiện :

TP. Hồ Chí Minh, 07/2021

Lời Nói Đầu

Cơ Sở Dữ Liệu là một trong những tiêu chí thiết yếu để làm ra một phần mềm hoàn chỉnh, ngay cả khi chỉ là một phần mềm nhỏ, nhưng vẫn phải có một cơ sở dữ liệu hoàn chỉnh vì đó là điều thiết yếu. Việc sử dụng cơ sở dữ liệu giúp ta quản lý dữ liệu một cách khoa học, tránh trùng lặp thông tin, đảm bảo được nhất quán và toàn vẹn dữ liệu,

thông tin được quản lý ở một nơi duy nhất, đảm bảo việc bảo mật dữ liệu, tránh rò rỉ thông tin, có khả năng chia sẻ thông tin cho nhiều người sử dụng.

Môn Cơ sở dữ liệu nâng cao cung cấp lý thuyết để thiết kế một cơ sở dữ liệu quan hệ. Các vấn đề đặt ra trong các giai đoạn thiết kế, từ thiết kế cấu trúc quan niệm đến thiết kế cấu trúc vật lý được thể trình bày rất chi tiết trong môn học này.

Trên cơ sở kiến thức của môn học, nhóm em đã quyết định lựa chọn đề tài “ Xây dựng cơ sở dữ liệu Quản lý Cửa Hàng Bán Linh Kiện PC” để củng cố kiến thức và hiện thực hóa những gì đã được thu nhận thông qua môn học.

Trong quá trình thực hiện, nhóm đã cố gắng tìm hiểu và áp dụng những kiến thức đã được học để hoàn thành đề tài một cách tốt nhất. Song vẫn còn có thể tồn tại một số sai sót, nhóm em rất mong nhận được sự góp ý của thầy để bài làm được hoàn thiện hơn.

Em xin chân thành cảm ơn!

MỤC LỤC

CHƯƠNG I: THIẾT KẾ CƠ SỞ DỮ LIỆU VÀ MÔ HÌNH THỰC THỂ LIÊN KẾT.....	1
1.1 Giới thiệu sơ lược về cơ sở dữ liệu quản lý cửa hàng bán linh kiện PC.....	1
1.2 Các thực thể và các thuộc tính của chúng.....	1
1.2.1 Các khái niệm cơ bản:.....	2
1.2.2 Thuộc tính của thực thể.....	6

1.3	Xác Định RBTV.....	7
1.3.1	Định Nghĩa.....	7
1.3.2	Xác định RBTV.....	9
1.4.	Xác định phụ thuộc hàm.....	12
1.5.	Xây dựng mô hình thực thể liên kết.....	13
1.5.1	Các khái niệm.....	13
1.5.2	Xác định các quan hệ trong CSDL quản lý cửa hàng bán linh kiện PC.....	13
CHƯƠNG II: CHUYỂN TỪ MÔ HÌNH THỰC THỂ LIÊN KẾT SANG MÔ HÌNH QUAN HỆ.....		14
2.1	Cách chuyển đổi.....	14
2.2	Xác định các bảng và thuộc tính của bảng.....	15
2.3	Mô hình dữ liệu quan hệ quản lý cửa hàng bán linh kiện pc.....	18
CHƯƠNG III: XÁC ĐỊNH KHÓA.....		19
3.1:	Khái niệm khóa.....	19
3.1.1	Khóa.....	19
3.1.2	Khóa chính là gì?.....	19
3.1.3	Khóa ngoại là gì?.....	19
3.1.4	Định nghĩa về khóa.....	20
3.1.5	Thuật toán xác định khóa của lược đồ quan hệ.....	20
3.2	Xác định khóa cho các quan hệ trong CSDL quản lý nhà sách.....	20
CHƯƠNG IV: CHUẨN HÓA LƯỢC ĐỒ QUAN HỆ THÀNH DẠNG CHUẨN 3NF HOẶC BCNF.....		22
4.1	Các dạng chuẩn và thuật toán chuẩn hóa.....	22
4.1.1	Các dạng chuẩn.....	22
a.	Dạng chuẩn 1:.....	22
b.	Dạng chuẩn 2.....	22
c.	Dạng chuẩn 3.....	23
d.	Dạng chuẩn BCK (Boyee-Codd-Kent) (còn gọi là BC):.....	24
e.	Dạng chuẩn 4.....	24
4.1.2	Thuật toán chuẩn hóa.....	26
a.	Dùng phương pháp làm phẳng để chuẩn hóa 1NF.....	26
b.	Dùng phương pháp tách nhóm lặp để chuẩn hóa 1NF.....	26
4.1	Các dạng chuẩn và thuật toán chuẩn hóa.....	26
4.2	Chuẩn hóa cơ sở dữ liệu quan hệ quản lý cửa hàng bán pc.....	27
CHƯƠNG V: CÂU LỆNH SQL.....		28

CHƯƠNG I: THIẾT KẾ CƠ SỞ DỮ LIỆU VÀ MÔ HÌNH THỰC THỂ LIÊN KẾT

1.1 Giới thiệu sơ lược về cơ sở dữ liệu quản lý cửa hàng bán linh kiện

PC

Quản lý cửa hàng bán linh kiện PC gồm Quản lý nhân viên, Quản lý sản phẩm, Quản lý khách hàng, Nhà Cung Cấp, Quản lý nhập hàng, Quản lý hóa đơn, Hóa Đơn Nhập Hàng, Tồn Kho, Giỏ Hàng, Chi Tiết Hóa Đơn, Danh Mục Sản Phẩm, Chi Tiết Danh Mục Sản Phẩm, Đơn Giá.

Nhân viên (IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole). Nhân viên thì bao gồm Nhân Viên (quản lý sản phẩm, xác nhận đơn hàng, cập nhật về thông tin sản phẩm) và Quản Lý (quản lý nhân viên, quản lý sản phẩm, xác nhận đơn hàng, quản lý đơn hàng nhập). Hệ thống sẽ lưu lại tất cả hoạt động của nhân viên và quản lý.

Sản Phẩm (IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, IdNCC, HinhAnh). Sản phẩm được phân biệt với nhau bằng Id Sản Phẩm, Tên Sản Phẩm, Gia,... Sản phẩm gồm nhiều danh mục sản phẩm khác nhau, nhà cung cấp khác nhau (được phân biệt qua Id danh mục sản phẩm, Id Nhà Cung Cấp). Một Nhà Cung Cấp có thể cung cấp nhiều danh mục sản phẩm và một danh mục sản phẩm được cung cấp bởi một hoặc nhiều nhà cung cấp.

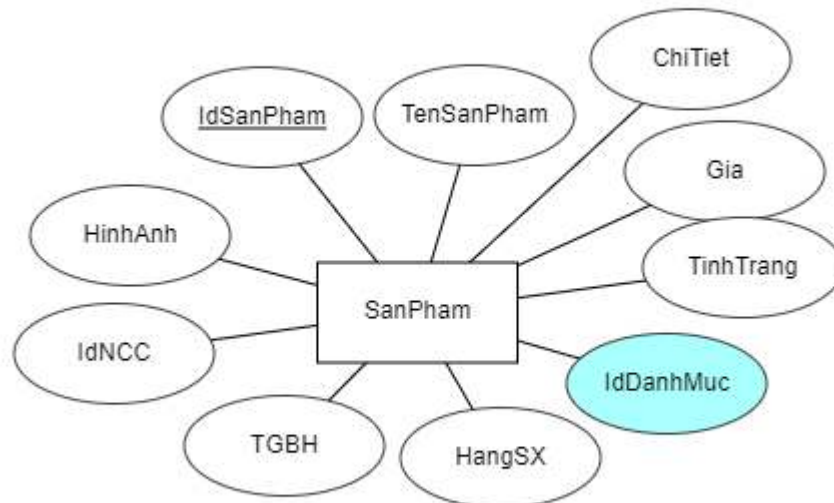
Khi số lượng sản phẩm tồn kho còn dưới 15 /1 Sản phẩm thì hệ thống sẽ gửi thông báo cho Quản lý. Quản lý sẽ lập phiếu nhập hàng (IdPhieu, IdSanPham, TenSanPham, SoLuong) để gửi xuống nhà cung cấp. Mỗi lần nhập hàng sẽ có Hóa Đơn Nhập Hàng (IdHDN, IdNCC, IdNV, NgayLapHD, NgayGiao, TongTien) do nhân viên nhập thực hiện.

Khách hàng khi đăng kí sẽ có 1 tài khoản cá nhân để lưu thông tin cũng như đặt mua hàng (IdKH, TenKH, Diachi, SDT, Email, MatKhau). Khi khách hàng thêm sản phẩm vào giỏ hàng thì số lượng hàng sẽ tạm thời trừ đi. Trong trường hợp khách hàng không chuyển sang trạng thái mua thì số lượng sẽ được khôi phục như ban đầu. Khi khách hàng tiến hành mua hàng thì sẽ lập Hóa Đơn (IdHD, IdKH, IdNV, NgayLapHD, NgayGiao, TongGia). Một khách hàng có thể có một hoặc nhiều hóa đơn.

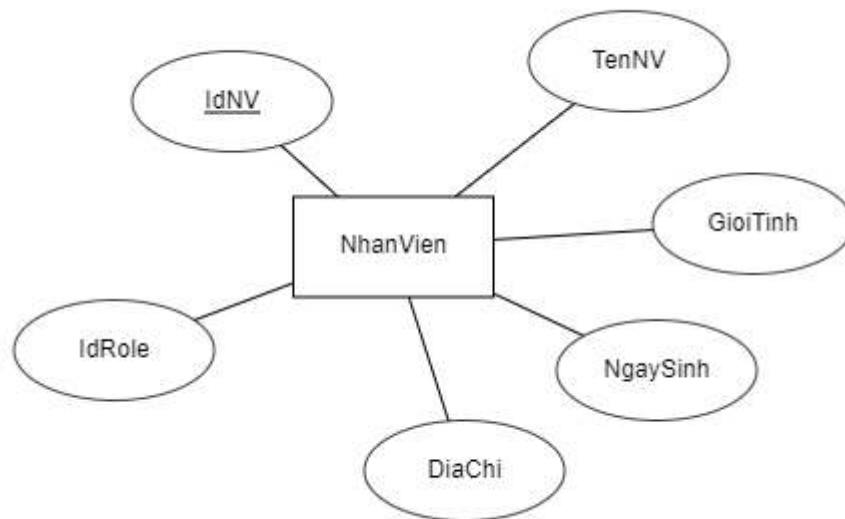
1.2 Các thực thể và các thuộc tính của chúng

1.2.1 Các khái niệm cơ bản:

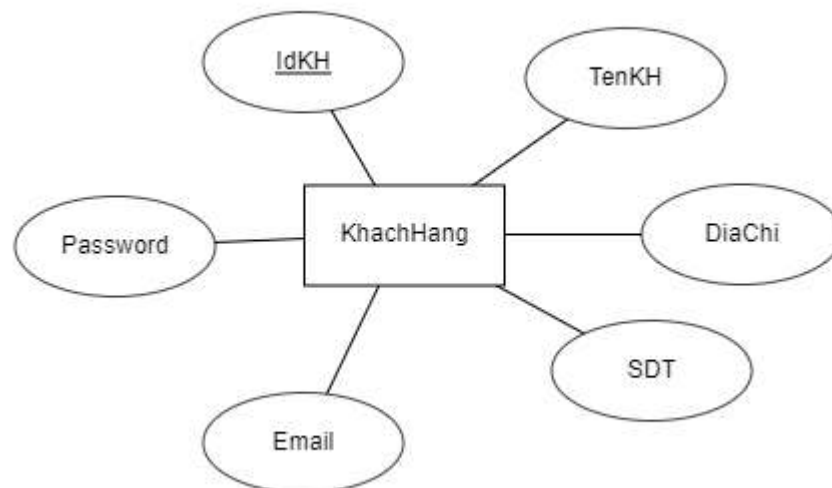
- Thực thể là khái niệm để chỉ một lớp các đối tượng có cùng đặc tính chung mà người ta muốn quản lý thông tin về nó.
- Ở đây bao gồm: Hãng Sản Xuất, Nhà Cung Cấp, Nhân Viên, Khách Hàng, Hóa Đơn, Danh Mục Sản Phẩm, Chi Tiết Hóa Đơn, Sản Phẩm, Hóa Đơn Nhập, Chi Tiết Nhập, Roles.
- Một đối tượng cụ thể trong thực thể được gọi là một cá thể (còn gọi là một thể hiện của thực thể).
- Thực thể Sản Phẩm gồm: IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, HinhAnh



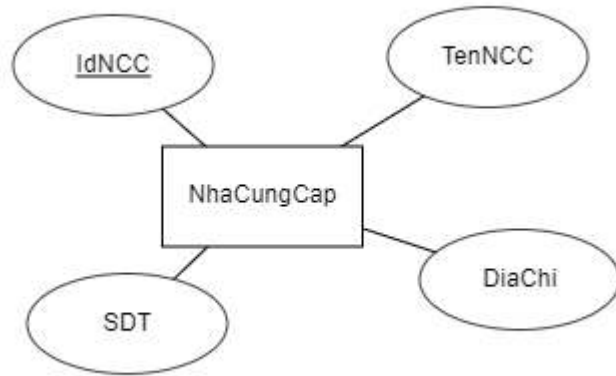
- Thực thể Nhân Viên gồm: IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole



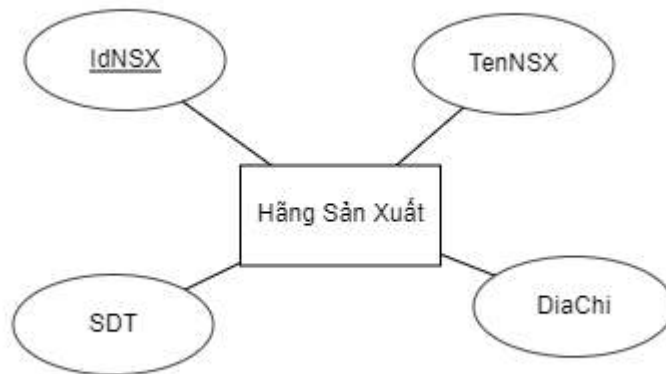
- Thực thể Khách Hàng gồm: IdKH, TenKH, DiaChi, SDT, Email, MatKhau



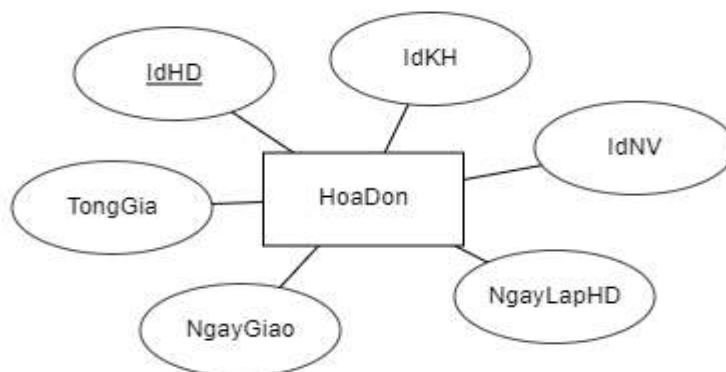
- Thực thể Nhà Cung Cấp gồm: IdNCC, TenNCC, SDT, DiaChi



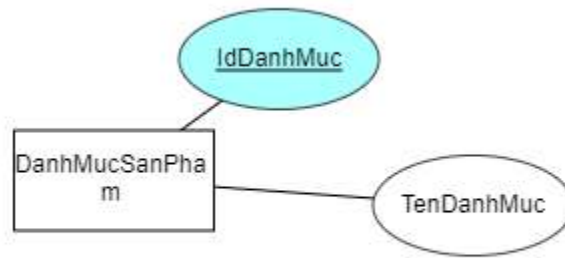
- Thực thể Hãng Sản Xuất gồm: IdNSX, TenNSX, DiaChi, SDT



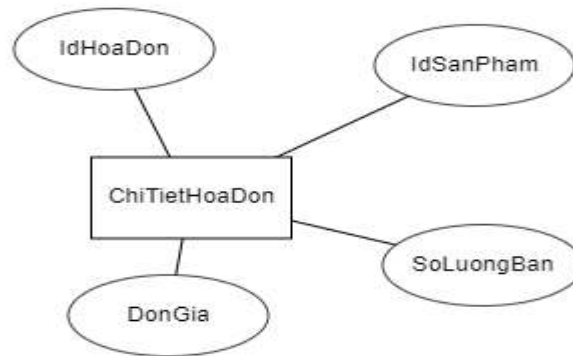
- Thực thể Hóa Đơn gồm: IdHD, IdKH, IdNV, NgayLapHD, NgayGiao, TongGia



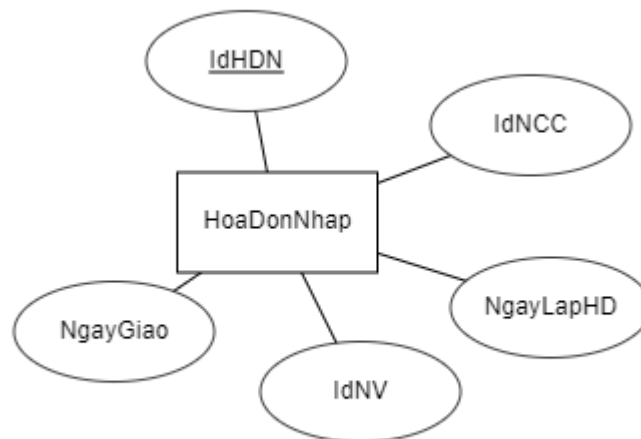
- Thực thể Danh Mục Sản Phẩm: IdDanhMuc, Ten



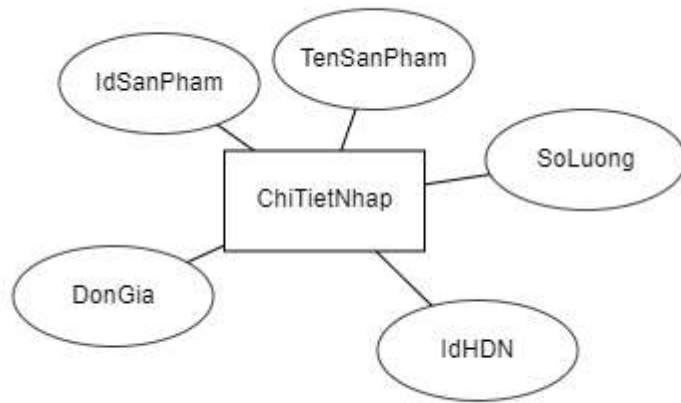
- Thực thể Chi Tiết Hóa Đơn: IdSanPham, IdHD, SoLuongBan, DonGia



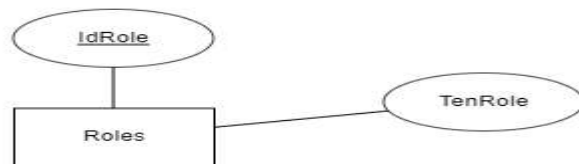
- Thực thể Hóa Đơn Nhập: IdHDN, IdNCC, IdNV, NgayLapHD, NgayGiao



- Thực thể Chi Tiết Nhập: IdSanPham, TenSanPham, SoLuong, Don Gia, IdHDN



- Thực thể Roles: IdRole, TenRole



1.2.2 Thuộc tính của thực thể

Để mô tả thông tin về một thực thể người ta thường dựa vào các đặc trưng riêng của thực thể đó. Các đặc trưng đó được gọi là thuộc tính của thực thể. Ví dụ thực thể Nhân Viên có các thuộc tính như IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole. Đối với thực thể Hóa Đơn thì giá trị của các thuộc tính IdSanPham, TenSanPham, SoLuong, DonGia,... là những thông tin mà nhà quản lý cần quan tâm để quản lý hàng hóa.

Thuộc tính của thực thể bao gồm các loại sau:

- Thuộc tính định danh (còn gọi là định danh thực thể, đôi khi còn gọi là thuộc tính khoá): Là một hoặc một số thuộc tính mà giá trị của nó cho phép phân biệt các thực thể khác nhau. Một thực thể bao giờ cũng được xác định một thuộc tính định danh làm cơ sở để phân biệt các thể hiện cụ thể của nó.

Ví dụ: IdKH, IdSanPham, IdNV,...

- Thuộc tính mô tả: Là các thuộc tính mà giá trị của chúng chỉ có tính mô tả cho thực thể hay liên kết mà thôi. Hầu hết các thuộc tính trong một kiểu thực thể đều là mô tả.

- Thuộc tính tên gọi: là thuộc tính mô tả để chỉ tên các đối tượng thuộc thực thể. Thuộc tính tên gọi để phân biệt các thực thể (tách các thực thể).
- Thuộc tính kết nối (thuộc tính khoá ngoài): là thuộc tính chỉ ra mối quan hệ giữa một thực thể đã có và một thực thể trong bảng khác. Thuộc tính kết nối giống thuộc tính mô tả thông thường trong thực thể chứa nó nhưng nó lại là thuộc tính khoá của một thực thể trong bảng khác.

1.3 Xác Định RBTV

1.3.1 Định Nghĩa

Ràng buộc toàn vẹn dữ liệu là các quy tắc trong một cơ sở dữ liệu nhằm kiểm tra tính đúng đắn và hợp lệ của dữ liệu trước khi lưu trữ. Các ràng buộc toàn vẹn dữ liệu trong SQL Server gồm:

- Constraint Default
- Constraint Not Null
- Constraint Unique
- Constraint Primary Key
- Constraint Foreign Key
- Constraint Check

a. Default constraint

Cột sẽ nhận giá trị mặc định nếu không cung cấp giá trị khi thêm dữ liệu vào bảng.

Cú pháp :

<Ten Cột> <Kieu Du Lieu> CONSTRAINT <Ten Constraint> DEFAULT <Bieu Thuc>

```
Query1.sql - DE...CV.master (sa (65))* X
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT NOT NULL PRIMARY KEY IDENTITY(1,1),
    TEN NVARCHAR (30) NOT NULL,
    NGÀYDAT DATE NOT NULL DEFAULT GETDATE()
);
```

b. Not Null constraint

Khai báo ở mức column

```

Query1.sql - DE...CV.master (sa (65))
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT CONSTRAINT NN NOT NULL PRIMARY KEY IDENTITY(1,1),
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE()
);

```

c. Unique constraint

Unique Constraint ở mức cột

```

Query1.sql - DE...CV.master (sa (65))*
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT CONSTRAINT NN NOT NULL PRIMARY KEY IDENTITY(1,1),
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE(),
    EMAIL NVARCHAR (30) UNIQUE
);

```

Unique Constraint ở mức bảng

```

Query1.sql - DE...CV.master (sa (65))*
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT CONSTRAINT NN NOT NULL PRIMARY KEY IDENTITY(1,1),
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE(),
    EMAIL NVARCHAR (30),
    CONSTRAINT UNK UNIQUE(EMAIL),
);

```

d. Primary key constraint

Khóa Chính ở mức cột

```

Query1.sql - DE...CV.master (sa (65))*
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT CONSTRAINT KB PRIMARY KEY IDENTITY(1,1),
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE(),
    EMAIL NVARCHAR (30),
    CONSTRAINT UNK UNIQUE(EMAIL),
);

```

Khóa Chính ở mức bảng

```

LQuery1.sql - DE...CV.master (sa (65))
CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT,
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE(),
    EMAIL NVARCHAR (30),
    CONSTRAINT KB PRIMARY KEY (ID),
    CONSTRAINT UNK UNIQUE(EMAIL),
);

```

Khóa Ngoại mức cột

```

CREATE TABLE [dbo].[ChiTietHDN](
    [IdSanPham] [int] NOT NULL,
    [IdHDN] [int] NOT NULL,
    [TenSanPham] [nvarchar](50) NULL,
    [SoLuong] [int] NULL,
    [Gia] [float] NULL,
    primary key (IdSanPham, IdHDN),
    foreign key (IdSanPham) references SanPham(IdSanPham),
    foreign key (IdHDN) references HoaDonNhap(IdHDN),
)

```

e. Check constraint

Khai báo những quy định mà mỗi dòng đều phải thỏa mãn.

Dùng để kiểm tra miền giá trị của dữ liệu.

Cú pháp:

1 CONSTRAINT <TEN Constraint> CHECK <Bieu Thuc>

```

CREATE DATABASE TEST
GO
CREATE TABLE DATHANG (
    ID INT,
    TEN NVARCHAR (30) NOT NULL,
    NGAYDAT DATE NOT NULL DEFAULT GETDATE(),
    EMAIL NVARCHAR (30),

    GIA INT CONSTRAINT CK CHECK( GIA >= 0 )

    CONSTRAINT KB PRIMARY KEY (ID),
    CONSTRAINT UNK UNIQUE(EMAIL),
);

```

1.3.2 Xác định RBTV

Gồm các điều kiện của RBTV và biểu diễn của chúng, đồng thời ta lập bảng tầm ảnh hưởng của mỗi RBTV.

Dấu (+): RBTV cần được kiểm tra nguy cơ dẫn tới vi phạm.

Dấu (-): RBTV không có nguy cơ bị vi phạm.

Dấu (-(*)): RBTV không bị vi phạm vì không được phép sửa đổi.

a) R1: mỗi SanPham (sản phẩm) có 1 mã sản phẩm khác nhau không trùng với mã sản phẩm nào khác

$\forall \text{ SanPham1, SanPham2 } \in \text{lop, SanPham1} \neq \text{SanPham2} \Rightarrow \text{SanPham1.IdSanPham} \neq \text{SanPham2.IdSanPham}$

Quan hệ	Thêm	Sửa	Xóa
SanPham	+(id)	-(*)	+

Bảng 1: Ràng buộc toàn vẹn bảng SanPham

b) R2: mỗi NhanVien (Nhân viên) có 1 mã số khác nhau không trùng với mã nhân viên khác.

$\forall \text{ NV1, NV2 } \in \text{lop, NV1} \neq \text{NV2} \Rightarrow \text{NV1.IdNV} \neq \text{NV2.IdNV}$

Quan hệ	Thêm	Sửa	Xóa
NhanVien	+(id)	-(*)	+

Bảng 2: Ràng buộc toàn vẹn bảng NhanVien

c) R3: mỗi KhachHang (khách hàng) có 1 mã số khác nhau nên không có khách hàng nào trùng nhau.

$\forall \text{ KhachHang1, KhachHang2 } \in \text{lop, KhachHang1} \neq \text{KhachHang2} \Rightarrow \text{KhachHang1.IdKH} \neq \text{KhachHang2.IdKH}$

Quan hệ	Thêm	Sửa	Xóa
KhachHang	+(id)	-(*)	+

Bảng 3: Ràng buộc toàn vẹn bảng KhachHang

d) R4: mỗi NhaCungCap (nhà cung cấp) có 1 mã số khác nhau không trùng với nhà cung cấp khác.

$\forall \text{ NCC1, NCC2 } \in \text{lop, NCC1} \neq \text{NCC2} \Rightarrow \text{NCC1.IdNCC} \neq \text{NCC2.IdNCC}$

Quan hệ	Thêm	Sửa	Xóa
Books	+(id)	-(*)	+

Bảng 4: Ràng buộc toàn vẹn bảng NhaCungCap

e) R5: mỗi HangSanXuat (Hãng sản xuất) có 1 mã hãng sản xuất khác nhau không trùng với hãng sản xuất nào khác.

$\forall \text{NSX1, NSX2} \in \text{lop}, \text{NSX1} \neq \text{NSX2} \Rightarrow \text{NSX1.IdNSX} \neq \text{NSX2.IdNSX}$

Quan hệ	Thêm	Sửa	Xóa
HangSanXuat	+(id)	-(*)	+

Bảng 5: Ràng buộc toàn vẹn bảng HangSanXuat

f) R6: mỗi HoaDon (Hóa đơn) có 1 mã số khác nhau không trùng với hóa đơn khác

$\forall \text{HD1, HD2} \in \text{lop}, \text{HD1} \neq \text{HD2} \Rightarrow \text{HD1.IdHD} \neq \text{HD2.IdHD}$

Quan hệ	Thêm	Sửa	Xóa
HoaDon	+(id)	-(*)	+

Bảng 6: Ràng buộc toàn vẹn bảng HoaDon

g) R7: mỗi DanhMucSanPham (Danh mục sản phẩm) có 1 mã số danh mục khác nhau không trùng với danh mục khác

$\forall \text{DM1, DM2} \in \text{lop}, \text{DM1} \neq \text{DM2} \Rightarrow \text{DM1.IdDDanhMuc} \neq \text{DM2.IdDanhMuc}$

Quan hệ	Thêm	Sửa	Xóa
DanhMucSanPham	+(id)	-(*)	+

Bảng 7: Ràng buộc toàn vẹn bảng DanhMucSanPham

h) R8: mỗi ChiTietHoaDon (Chi tiết hóa đơn) có 1 mã số khác nhau không trùng với chi tiết hóa đơn khác.

$\forall \text{CTHD1, CTHD2} \in \text{lop}, \text{CTHD1} \neq \text{CTHD2} \Rightarrow \text{CTHD1.Id} \neq \text{CTHD2.Id}$

Quan hệ	Thêm	Sửa	Xóa
ChiTietHoaDon	+(id)	-(*)	+

Bảng 8: Ràng buộc toàn vẹn bảng ChiTietHoaDon

i) R9: mỗi HoaDonNhap (Hóa đơn nhập) có 1 mã số khác nhau không trùng với hóa đơn nhập khác

$\forall \text{HDN1, HDN2} \in \text{lop}, \text{HDN1} \neq \text{HDN2} \Rightarrow \text{HDN1.IdHDN} \neq \text{HDN2.IdHDN}$

Quan hệ	Thêm	Sửa	Xóa
HoaDonNhap	+(id)	-(*)	+

Bảng 8: Ràng buộc toàn vẹn bảng HoaDonNhap

j) R10: mỗi Roles (Roles) có 1 mã số khác nhau không trùng với role khác

$\forall \text{Role1, Role2} \in \text{lop}, \text{Role1} \neq \text{Role2} \Rightarrow \text{Role1.IdRole} \neq \text{Role2.IdRole}$

Quan hệ	Thêm	Sửa	Xóa
Roles	+(id)	-(*)	+

Bảng 8: Ràng buộc toàn vẹn bảng HoaDonNhap

1.4. Xác định phụ thuộc hàm

Phụ thuộc hàm trong hệ quản trị cơ sở dữ liệu, có tên tiếng anh là Functional Dependency và viết tắt là FD, xác định mối quan hệ của một thuộc tính này với một thuộc tính khác trong hệ quản trị cơ sở dữ liệu.

Với quản lý nhà sách có những phụ thuộc hàm sau đây:

- Bảng SanPham (Sản Phẩm): Nếu ta biết được IdSanPham ta có thể lấy được các trường khác: TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, IdNCC, HinhAnh.
- Bảng NhanVien (Nhân viên): Nếu ta biết được IdNV ta có thể lấy được các trường khác: TenNV, GioiTinh, NgaySinh, DiaChi, IdRole.
- Bảng KháchHang (Khách hàng): Nếu ta biết được IdKH ta có thể lấy được các trường khác: TenKH, DiaChi, SDT, Email, MatKhau.
- Bảng NhaCungCap (Nhà cung cấp): Nếu ta biết được IdNCC ta có thể lấy được các trường khác: TenNCC, SDT, DiaChi
- Bảng HangSanXuat (Hãng sản xuất): Nếu ta biết được IdNSX ta có thể lấy được các trường khác: TenNSX, DiaChi, SDT
- Bảng HoaDon (Hóa Đơn): Nếu ta biết được IdHoaDon ta có thể lấy được các trường khác: IdKH, IdNV, NgayLapHD, NgayGiao, TongGia

- Bảng DanhMucSanPham (Danh mục sản phẩm): Nếu ta biết được IdDanhMuc ta có thể lấy được các trường khác: TenDanhMuc
- Bảng HoaDonNhap (Hóa đơn nhập): Nếu ta biết được IdHDN ta có thể lấy được các trường khác: IdNCC, NgayLapHD, NgayGiao.
- Bảng ImportDetails (chi tiết nhập): Nếu ta biết được Id ta có thể lấy được các trường khác: bookId, count, price, status.
- Bảng Roles (Roles): Nếu ta biết được IdRoles ta có thể lấy được các trường khác: TenRole.

1.5. Xây dựng mô hình thực thể liên kết

1.5.1 Các khái niệm

- Mô hình thực thể liên kết còn được gọi là Mô Hình ER.
- Mô hình ER được dùng để xây dựng mô hình hóa dữ liệu ý niệm (conceptual data modeling) nhằm biểu diễn cấu trúc và các ràng buộc của CSDL.
- Mô hình ER như một công cụ giao tiếp giữa người thiết kế CSDL và người sử dụng cuối cùng để xây dựng CSDL trong giai đoạn phân tích. Nó độc lập với DBMS và quá trình thi công CSDL.

1.5.2 Xác định các quan hệ trong CSDL quản lý cửa hàng bán linh kiện PC

- NhanVien và HoaDonNhap có mối quan hệ (1,n) và mối quan hệ giữa chúng là quản lý.
- NhanVien và Roles có mối quan hệ (1,n).
- NhanVien và HoaDon có mối quan hệ (1,n) và mối quan hệ giữa chúng là quản lý.
- KhachHang và HoaDon có mối quan hệ (1,n) và mối quan hệ giữa chúng là thanh toán.
- SanPham và ChiTietNhap có mối quan hệ (1,n) và mối quan hệ giữa chúng là nhập.
- SanPham và HoaDon có mối quan hệ (n,n) và mối quan hệ giữa chúng là ChiTietHoaDon.
- DanhMucSanPham và SanPham có mối quan hệ (n,1) và mối quan hệ giữa chúng là trong.
- HangSanXuat và SanPham có mối quan hệ (n,1) và mối quan hệ giữa chúng là trong.

[illegible]

2.1 Cách chuyển đổi

+ Chuyển đổi các tập thực thể: Mỗi thực thể được chuyển thành một quan hệ cùng tên và danh sách thuộc tính. Đối với kiểu thực thể thông thường (regular entity type): Khóa của quan hệ là khóa của kiểu thực thể. Thuộc tính của quan hệ cũng chính là thuộc tính của kiểu thực thể. Quan hệ không chứa các thuộc tính đa trị mà chỉ chứa thuộc tính thành phần của những thuộc tính phức hợp.

+ Chuyển đổi mỗi quan hệ 1 – 1: Thuộc tính khóa bên này sẽ làm khóa ngoại của bên kia hoặc ngược lại.

14

nhiều.

+ Chuyển đổi mỗi quan hệ $n - n$: Chuyển thành quan hệ mới có khoá chính gồm 2 thuộc tính khóa của 2 quan hệ trở lên, thuộc tính mỗi kết hợp (nếu có) trở thành thuộc tính của quan hệ mới.

+ Chuyển đổi mỗi quan hệ đa trị: Chuyển thành quan hệ mới có khoá chính gồm thuộc tính đa trị và thuộc tính khóa của thực thể. Sau khi chuyển thành quan hệ mới, thuộc tính đa trị sẽ tự động biến mất khỏi thực thể cũ.

- Các bước chuyển đổi:

Bước 1: Chuyển đổi từng loại thực thể thành một loại quan hệ tương ứng.

+ Chuyển đổi các liên kết 1 – 1 kết hợp hai thực thể thành một thực thể.

+ Các liên kết 1 – n lấy khóa bên có nhiều thực thể và chuyển đổi nó thành khóa ngoại.

+ Mỗi quan hệ $n - n$ tạo thành một kiểu quan hệ mới.

Bước 2: Kiểm tra lại dạng chuẩn của các quan hệ.

2.2 Xác định các bảng và thuộc tính của bảng

SanPham (IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, IdNCC, HinhAnh)

SanPham
<u>IdSanPham</u>
TenSanPham
ChiTiet
Gia
TinhTrang
IdDanhMuc
HangSx
TGBH
IdNCC
HinhAnh

SanPham (IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole)

SanPham
<u>IdNhanVien</u>
TenNV GioiTinh
NgaySinh
DiaChi
IdRole

KhachHang (IdKH, TenKH, DiaChi, SDT, Email, MatKhau)

KhachHang
<u>IdKH</u>
TenKH
DiaChi
SDT
Email
MatKhau

NhaCungCap (IdNCC, TenNCC, SDT, DiaChi)

NhaCungCap
<u>IdNCC</u>
TenNCC
SDT
DiaChi

HangSanXuat (IdNSX, TenNSX, DiaChi, SDT)

HangSanXua
<u>IdNSX</u>
TenNSX
DiaChi
SDT

HoaDon (IdHD, IdKH, IdNV, NgayLapHD, NgayGiao, TongGia)

HoaDon

<u>IdHD</u>
IdKH
IdNV
NgayLapHD
NgayGiao
TongGia

DanhMucSanPham (IdDanhMuc, Ten)

DanhMucSanPham
<u>IdDanhMuc</u>
Ten

ChiTietHoaDon (IdSanPham, IdHD, SoLuongBan, DonGia)

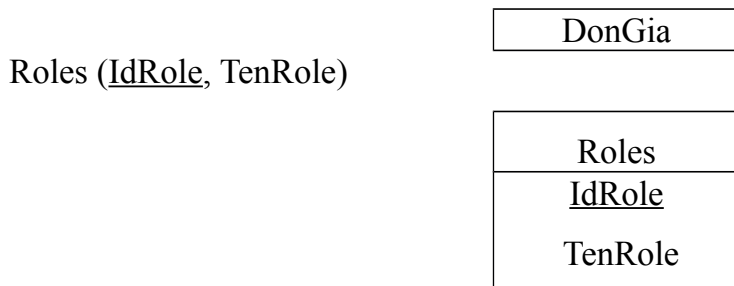
ChiTietHoaDon
<u>IdSanPham</u>
IdHD
SoLuongBan
DonGia

HoaDonNhap (IdHDN, IdNCC, NgayLapHD, NgayGiao)

HoaDonNhap
<u>IdHDN</u>
IdNCC
NgayLapHD
NgayGiao

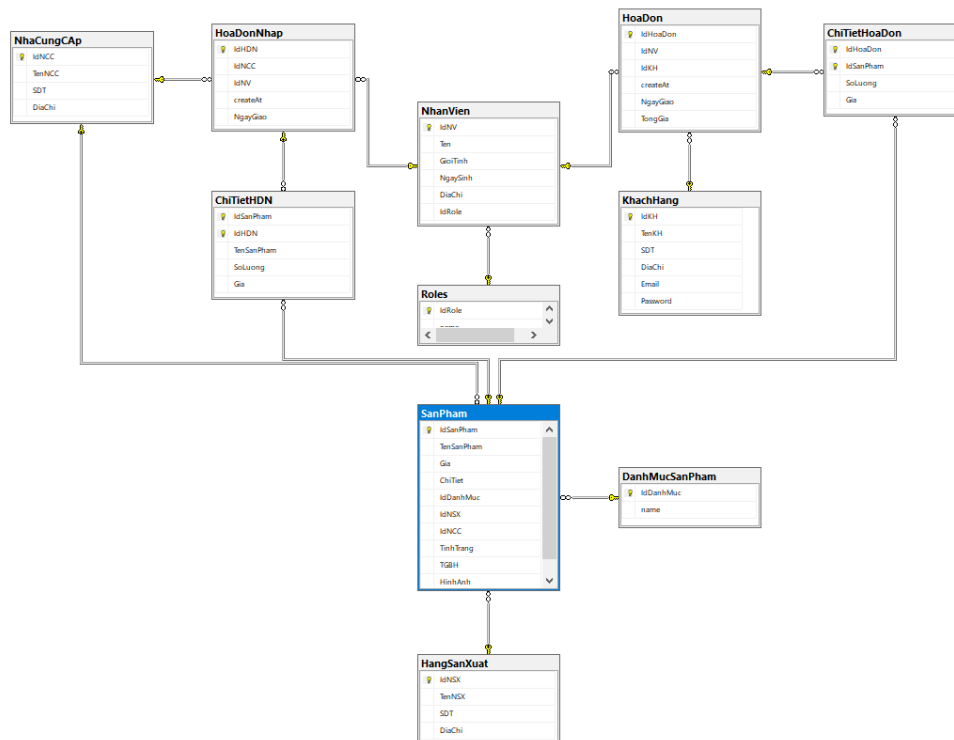
ChiTietNhap (IdSanPham, TenSanPham, SoLuong, IdNCC, Don Gia)

ChiTietNhap
<u>IdSanPham</u>
TenSanPham
SoLuong
IdNCC



2.3 Mô hình dữ liệu quan hệ quản lý cửa hàng bán linh kiện pc

- SanPham (IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, **IdDanhMuc** HangSX, TGBH, **IdNSX**, **IdNCC**, HìnhAnh)
- NhanVien (IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole)
- KháchHang (IdKH, TenKH, DiaChi, SDT, Email, MatKhu)
- NhaCungCap (IdNCC, TenNCC, SDT, DiaChi)
- HangSanXuat (IdNSX, TenNSX, DiaChi, SDT)
- HoaDon (IdHD, **IdKH**, **IdNV**, NgayLapHD, NgayGiao, TongGia)
- ChiTietNhap (**IdSanPham**, TenSanPham, SoLuong, Don Gia, **IdHDN**)
- Roles: (IdRole, TenRole)
- DanhMucSanPham(IdDanhMuc, Ten)
- ChiTietHoaDon (IdSanPham, **IdHD**, SoLuongBan, DonGia)
- HoaDonNhap (IdHDN, **IdNCC**, **IdNV**, NgayLapHD, NgayGiao)



CHƯƠNG III: XÁC ĐỊNH KHÓA

3.1: Khái niệm khóa

3.1.1 Khóa

Khóa của một bảng là một tập gồm một hay một số thuộc tính của bảng phân biệt được các cá thể.

3.1.2 Khóa chính là gì?

Khóa chính (hay ràng buộc khóa chính) được sử dụng để định danh duy nhất mỗi record trong table của cơ sở dữ liệu.

+ Ngoài ra, nó còn dùng để thiết lập quan hệ 1 - n (hay ràng buộc tham chiếu) giữa hai table trong cơ sở dữ liệu.

+ Dữ liệu (value) của field khóa chính phải có tính duy nhất. Và không chứa các giá trị Null.

+ Mỗi table nên chỉ có một khóa chính, khóa chính có thể tạo ra từ nhiều field của table.

3.1.3 Khóa ngoại là gì?

- Khóa ngoại của một table được xem như con trỏ trỏ tới khóa chính của table khác.

***Chú ý:**

+ Mỗi bảng có ít nhất một khóa. Việc xác định khóa phụ thuộc vào quan hệ logic của các dữ liệu chứ không phụ thuộc vào giá trị của các dữ liệu.

+ Nên chọn khóa chính là khóa có ít thuộc tính nhất.

- **Liên kết:** Thực chất sự liên kết giữa các bảng là dựa trên thuộc tính khóa. Chẳng hạn thuộc tính số thẻ là khóa của bảng người mượn xuất hiện lại ở bảng mượn sách đó tạo nên liên kết giữa 2 bảng này.

3.1.4 Định nghĩa về khóa

Khóa được định nghĩa lại bằng phụ thuộc hàm như sau:

$R(U)$, $U = \{A_1, A_2, \dots, A_n\}$, $F = \{f_1, f_2, \dots, f_m\}$ xác định trên R , $K \subseteq U$ là khóa của R

nếu thỏa mãn hai điều kiện sau:

(i) $K \rightarrow U$.

(ii) $\nexists K' \subset K$ mà $K' \rightarrow U$.

Biểu diễn lược đồ quan hệ $R(U)$ bằng đồ thị có hướng như sau: Mỗi nút của đồ thị là tên một thuộc tính của R .

Cung nối 2 thuộc tính A và B thể hiện phụ thuộc hàm $A \rightarrow B$.

Thuộc tính mà chỉ có các mũi tên đi ra gọi là gốc.

Thuộc tính mà chỉ có các cung đi tới gọi là nút lá.

Như vậy khóa phải là bao phủ tập hợp các nút gốc, đồng thời không chứa bất kỳ nút lá nào của đồ thị.

3.1.5 Thuật toán xác định khóa của lược đồ quan hệ

xuất phát từ tập các nút gốc (X), dựa trên tập các phụ thuộc hàm F, tìm bao đóng X^+ .
nếu $X^+ = U$ thì X là khóa.

Ngược lại thì bổ sung thuộc tính không thuộc nút lá vào X rồi tìm bao đóng. Cứ như thế cho tới khi tìm được bao đóng của X bằng U.

Cuối cùng khóa chính là X.

3.2 Xác định khóa cho các quan hệ trong CSDL quản lý nhà sách

a. Quan hệ SanPham: (IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, IdNCC, HinhAnh)

ký hiệu $Q1(U) = Q1(A,B,C,D,E,F,G,H,I,J)$

ta có tập PTH $F1 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G, A \rightarrow H, A \rightarrow I, A \rightarrow J \}$ Xét $A = ABCDEFGHIJ = U$

→ Vậy khóa của quan hệ SanPham là A(IdSanPham)

b. Quan hệ NhanVien (IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole) ký hiệu

$Q1(U) = Q1(A,B,C,D,E,F)$

ta có tập PTH $F2 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F \}$

Xét $A = ABCDEF = U$

→ Vậy khóa của quan hệ NhanVien là A(IdNV)

c. Quan hệ KhachHang (IdKH, TenKH, DiaChi, SDT, Email, MatKhau) ký hiệu $Q1(U)$

$= Q1(A,B,C,D,E,F)$

ta có tập PTH $F3 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F \}$

Xét $A = ABCDEF = U$

→ Vậy khóa của quan hệ KhachHang là A(IdKH)

d. Quan hệ NhaCungCap (IdNCC, TenNCC, SDT, DiaChi) ký hiệu $Q1(U) =$

$Q1(A,B,C,D)$

ta có tập PTH $F4 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D \}$

Xét $A = ABCDEF = U$

→ Vậy khóa của quan hệ NhaCungCap là A(IdNCC)

e. Quan hệ HangSanXuat (IdNSX, TenNSX, DiaChi, SDT) ký hiệu $Q1(U) =$

$Q1(A,B,C,D)$

ta có tập PTH $F5 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D \}$

Xét $A = ABCD = U$

→ Vậy khóa của quan hệ HangSanXuat là $A(IdNSX)$

f. Quan hệ HoaDon ($IdHD, IdKH, IdNV, NgayLapHD, NgayGiao, TongGia$) ký hiệu

$Q1(U) = Q1(A,B,C,D,E,F)$

ta có tập PTH $F6 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F \}$

Xét $A = ABCDEF = U$

→ Vậy khóa của quan hệ HoaDon là $A(IdHD)$

g. Quan hệ DanhMucSanPham ($IdDanhMuc, Ten$) ký hiệu $Q1(U) = Q1(A,B,C,D,E,F)$

ta có tập PTH $F7 = \{ A \rightarrow B \}$

Xét $A = ABC = U$

→ Vậy khóa của quan hệ DanhMucSanPham là $A(IdDanhMuc)$

h. Quan hệ ChiTietHoaDon ($IdSanPham, IdHD, SoLuongBan, DonGia$) ký hiệu $Q1(U)$

$= Q1(A,B,C,D)$

ta có tập PTH $F8 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D \}$

Xét $A = ABC = U$

→ Vậy khóa của quan hệ ChiTietHoaDon là $A(IdDanhMuc)$

i. Quan hệ HoaDonNhap ($IdHDN, IdNCC, NgayLapHD, NgayGiao$) ký hiệu $Q1(U) =$

$Q1(A,B,C,D,E)$

ta có tập PTH $F9 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D \}$

Xét $A = ABCD = U$

→ Vậy khóa của quan hệ HoaDonNhap là $A(IdHDN)$

j. Quan hệ ChiTietNhap ($IdSanPham, TenSanPham, SoLuong, IdNCC, Don Gia$) ký

hiệu $Q1(U) = Q1(A,B,C,D,E)$

ta có tập PTH $F10 = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E \}$

Xét $A = ABCDE = U$

→ Vậy khóa của quan hệ ChiTietHoaDon là $A(IdSanPham)$

k. Quan hệ Roles ($IdRole, TenRole$) ký hiệu $Q1(U) = Q1(A,B)$

ta có tập PTH $F10 = \{ A \rightarrow B \}$

Xét $A = AB = U$

→Vây khóa của quan hệ ChiTietHoaDon là A(IdRole)

CHƯƠNG IV: CHUẨN HÓA LƯỢC ĐỒ QUAN HỆ THÀNH DẠNG CHUẨN 3NF HOẶC BCNF

4.1 Các dạng chuẩn và thuật toán chuẩn hóa.

4.1.1 Các dạng chuẩn.

a. Dạng chuẩn 1:

- Khái niệm Thuộc tính đơn: Một thuộc tính được gọi là thuộc tính đơn nếu giá trị của nó không phải là sự kết hợp bởi nhiều thông tin có ý nghĩa khác nhau và hệ thống luôn truy xuất trên toàn bộ giá trị của nó ít khi truy xuất đến từng phần dữ liệu của nó. Ngược lại, là thuộc tính kép.

- Định nghĩa DC1: Một lược đồ quan hệ Q đạt dạng chuẩn 1 nếu mọi thuộc tính của Q đều là thuộc tính đơn.

b. Dạng chuẩn 2

Khái niệm phụ thuộc đầy đủ:

+ Thuộc tính A được gọi là phụ thuộc đầy đủ vào tập thuộc tính X nếu:

- $A \in X^+_F$
- $X \not\subseteq A$ là pth nguyên tố.

Khái niệm thuộc tính khóa, thuộc tính không khóa

+ Thuộc tính khóa (thuộc tính nguyên tố): là một thuộc tính của lược đồ quan hệ Q mà là thành phần của ít nhất một khóa dự tuyển trong Q.

+ Thuộc tính khóa (thuộc tính nguyên tố): là một thuộc tính của lược đồ quan hệ Q mà là thành phần của ít nhất một khóa dự tuyển trong Q

+ Thuộc tính không khóa (thuộc tính không nguyên tố): là thuộc tính không phải là thuộc tính nguyên tố (tức không tham gia vào bất kỳ một khóa nào của lược đồ quan hệ)

- Định nghĩa Dạng chuẩn 2:

Một lược đồ quan hệ Q đạt dạng chuẩn 2 nếu

- Q ở DC1
- Mọi thuộc tính không khóa đều phụ thuộc đầy đủ vào các khóa của Q.

- Thuật toán kiểm tra dạng chuẩn 2

+ Vào: lược đồ quan hệ Q, tập phụ thuộc hàm F

+ Ra: khẳng định Q đạt chuẩn 2 hay không đạt chuẩn 2.

- Bước 1: Tìm tất cả khóa của Q
- Bước 2: Với mỗi khóa K, tìm bao đóng tất cả tập con thật sự S

của K.

• Bước 3: Nếu có bao đóng S+ chứa thuộc tính không khóa thì Q không đạt chuẩn 2
Ngược lại thì Q đạt chuẩn 2

- Nhận xét:

+ Nếu mỗi khóa của quan hệ R chỉ có 1 thuộc tính thì R đạt dạng chuẩn 2.

+ Quan hệ SV ở dạng chuẩn 2 nhưng vẫn trùng lặp thông tin.

c. Dạng chuẩn 3

Khái niệm Phụ thuộc bắc cầu:

+ Cho R, F, $X \subseteq R$: $A \in R$, thuộc tính A được gọi là PTBC vào một tập thuộc tính X dựa trên F nếu tồn tại nhóm thuộc tính thỏa mãn các điều kiện sau:

- $X \subset Y$
- $Y \twoheadrightarrow X$
- $Y \subset A$
- $A \notin \{X, Y\}$

- Định nghĩa DC3:

Một lược đồ quan hệ R đạt dạng chuẩn 3 nếu:

- R ở DC2
- b. Mọi thuộc tính không khóa R đều không phụ thuộc bắc cầu vào một khóa nào của R.

- Thuật toán kiểm tra dạng chuẩn 3:

+ Vào: lược đồ quan hệ Q, tập phụ thuộc hàm F.

+ Ra: khẳng định Q đạt chuẩn 3 hay không đạt chuẩn 3.

- Bước 1: Tìm tất cả khóa của Q.
- Bước 2: Từ F tạo tập phụ thuộc hàm tương đương F1 có vẻ phải một thuộc tính.
- Bước 3: Nếu mọi phụ thuộc hàm $X \rightarrow A \in F1$ với $A \notin X$ đều có X là siêu khóa hoặc A là thuộc tính khóa thì Q đạt chuẩn 3 ngược lại Q không đạt chuẩn 3.

d. Dạng chuẩn BCK (Boyce-Codd-Kent) (còn gọi là BC):

- Định nghĩa: Một lược đồ quan hệ Q ở dạng chuẩn BCK nếu mọi phụ thuộc hàm không hiển nhiên đều có vế trái chứa khóa.

$X \subset A \in F^+ : A \notin X$ và X phải chứa khóa của Q

- Nhận xét: Nếu Q đạt dạng chuẩn BCK thì mọi vế trái của phụ thuộc hàm đều là siêu khóa.

- Thuật toán kiểm tra dạng chuẩn BC:

+ Vào: lược đồ quan hệ Q , tập phụ thuộc hàm F .

+ Ra: khẳng định Q đạt chuẩn BC hay không đạt chuẩn BC.

- Bước 1: Tìm tất cả khóa của Q .
- Bước 2: Từ F tạo tập phụ thuộc hàm tương đương F_1 có vẻ phải một thuộc tính.
- Bước 3: Nếu mọi phụ thuộc hàm $X \rightarrow A \in F_1$ với $A \notin X$ đều có X là siêu khóa thì Q đạt chuẩn BC ngược lại Q không đạt chuẩn BC.

e. Dạng chuẩn 4

- Phụ thuộc đa trị.

- Ngoài các pth đã trình bày, người ta còn xét đến một loại phụ thuộc hàm khác, đó là phụ thuộc hàm đa trị.

- Định nghĩa Phụ thuộc đa trị:

+ Cho một LDQH $Q(X,Y,Z)$ với $X \subset Q^+, Y \subset Q^+, X \cap Y = \emptyset$ và $Z = Q^+ \setminus \{X,Y\}$

+ Ký hiệu $X \twoheadrightarrow Y$ là một Phụ thuộc hàm đa trị được định nghĩa trên Q nếu mỗi giá trị x của X xác định duy nhất một tập giá trị $\{y_1, y_2, \dots\}$ của Y , và tập giá trị này không phụ thuộc vào các giá trị của Z trong các bộ có liên quan đến x, y_1, y_2, \dots

+ Nghĩa là: Với mọi bộ $(x, z_1), (x, z_2) \in Q[X,Z]$ thì $(Q: X=x \text{ và } Z=z_1)[Y] = (Q: X=x \text{ và } Z=z_2)[Y]$

- Phụ thuộc hiển nhiên: Phụ thuộc hàm đa trị $X \twoheadrightarrow Y$ là một Phụ thuộc hàm đa trị hiển nhiên trên Q nếu $X \cup Y = Q^+$ (nghĩa là $Z = \emptyset$)

- Nhận xét: Nếu $X \twoheadrightarrow Y$ là một phụ thuộc đa trị thì $Q[X,Y] \quad Q[X,Z] = Q$

Vậy với phụ thuộc đa trị $X \twoheadrightarrow Y$ thì kết nối trên không dư thừa thông tin, hay nói cách khác phân rã trên $(Q$ thành $Q[X,Y], Q[X, Z])$ không mất mát thông tin).

- Cách Kiểm tra phụ thuộc đa trị:

+ Biến đổi các phụ thuộc đa trị không hiển nhiên trong một cấu trúc này thành phụ thuộc đa trị hiển nhiên trong 1 cấu trúc khác.

- Hệ Luật dẫn trên phụ thuộc hàm đa trị: Một số hệ luật dẫn cơ bản:

Cho lược đồ quan hệ Q và $X, Y, W, Z \subset Q^+$.

+ LD1: Luật bù: $X \twoheadrightarrow Y$ thì $X \twoheadrightarrow (Q^+ - X - Y)$

- Ví dụ: Từ $M \twoheadrightarrow P$ suy ra $M \twoheadrightarrow N, G$

+ LD2: Luật thêm vào: Nếu $X \twoheadrightarrow Y$ và $Z \subseteq W$ thì $X, W \twoheadrightarrow Y, Z$

+ LD3: Luật bắc cầu: Nếu $X \twoheadrightarrow Y$ và $Y \twoheadrightarrow Z$ thì $X \twoheadrightarrow (Z-Y)$

+ LD4: Nếu $X \subset Y$ thì $X \twoheadrightarrow Y$

+ LD5: Nếu $X \twoheadrightarrow Y$ và $W \subset Z$, với $Z \subseteq Y; W \cap Y = \emptyset$ thì $X \subset Z$ Định

nghĩa Dạng chuẩn 4: Q đạt dạng chuẩn 4 nếu:

- + (i) Q ở dạng chuẩn BCK và
- + (ii) \forall Phụ thuộc đa trị không hiển nhiên $X \twoheadrightarrow Y$ được định nghĩa trên Q thì về trái X phải chứa 1 khóa của $Q^+ \setminus Y$, nghĩa là $\forall A \in Q^+ \setminus Y$ thì $X \subseteq A \in F^+$.

Nhận xét:

+ Trong các DC, DC BCK và DC4 là những dạng chuẩn nhằm giảm thiểu tối đa những thông tin trùng lặp và giải quyết tương đối hiệu quả việc kiểm tra các phụ thuộc hàm (đối với DC BCK) và phụ thuộc đa trị (đối với DC4).

+ Tuy nhiên, đôi khi vẫn còn tồn tại một số pth mà việc kiểm tra chúng không được thuận lợi vì phải thực hiện trên nhiều quan hệ. Khi đó, người thiết kế có thể lựa chọn 1 cấu trúc hợp lý, phù hợp với yêu cầu khai thác CSDL: dựa trên khối lượng dữ liệu trong mỗi quan hệ; tần suất thực hiện các thao tác thêm / xóa / sửa trên quan hệ; về yêu cầu thời gian xử lý... và sẽ đặt ra những ưu tiên:

+ Khi chỉ có phụ thuộc hàm: Chọn DC3 và chấp nhận một số bất tiện khi khai thác để đánh đổi việc kiểm tra tất cả các pth đều thuận lợi; hoặc chọn DC BCK và chấp nhận kiểm tra một số pth sẽ phức tạp hơn.

+ Khi có thêm phụ thuộc đa trị: cân nhắc giữa DC4, DC BCK, DC3 cũng dựa theo lý lẽ tương tự như trên.

4.1.2 Thuật toán chuẩn hóa

a. Dùng phương pháp làm phẳng để chuẩn hóa 1NF

- Ở đây có một tutorial rất hay về chuẩn hóa CSDL và vì đây là trang công nghệ thông tin tiếng Việt nên tutorial đó cũng đã có bản tiếng Việt cho những bạn không thích tiếng Anh. Mời các bạn thảo luận về chuẩn hóa CSDL ở topic này.

b. Dùng phương pháp tách nhóm lặp để chuẩn hóa 1NF

Chuẩn hóa về 1NF: Loại bỏ nhóm lặp và loại bỏ các thuộc tính tính toán.

-Ta có lược đồ quan hệ R, A là thuộc tính đa trị hoặc thuộc tính lặp. Để chuẩn hóa về 1NF:

- + Ta loại A ra khỏi R
- + Tạo 1 lược đồ quan hệ mới $\{k, A\}$ nếu A là thuộc tính đa trị.
- + Tạo 2 lược đồ quan hệ $\{K\}, \{K, A\}$

Chuẩn hóa về 2NF

2NF – Loại bỏ các phụ thuộc hàm không hoàn toàn vào khóa chính Ta có lược đồ quan hệ R, $X \rightarrow Y$ là phụ thuộc hàm bộ phận thuộc R Để chuẩn hóa về 2NF:

- Loại X ra khỏi R
- Đưa X về một lược đồ quan hệ mới

Chuẩn hoá R về dạng 3NF

3NF – Loại bỏ các phụ thuộc hàm bậc cầu vào khóa chính

+ Bước 1: Tìm phủ tối thiểu. Tách về phải các phụ thuộc hàm. Tối giản trái các phụ thuộc hàm. Tìm phủ không dư thừa (thuật toán Nonredundant).

+ Bước 2: Loại bỏ các thuộc tính phụ thuộc hàm bậc cầu. Tìm khoá. Tìm phụ thuộc hàm bậc cầu.

+ Bước 3: Lập bảng mới với khoá chính là khoá chính phụ thuộc hàm bậc cầu. Thuộc tính còn lại lập thành bảng với khoá chính là khoá ban đầu.

Chuẩn hoá R về dạng BCNF

+ Chọn một trong số các phụ thuộc hàm vi phạm BCNF. Giả sử $X \rightarrow Y$. Tìm X^+ .

+ Tạo một lược đồ R_1 mới gồm các thuộc tính trong X^+ . Các thuộc tính còn lại, gồm cả X hình thành một lược đồ mới R_2 .

+ Tìm các phụ thuộc hàm cho R_1 và R_2 từ tập phụ thuộc hàm ban đầu.

+ Lặp lại chu trình lên tới khi tất cả các lược đồ thuộc dạng chuẩn BCNF.

4.1 Các dạng chuẩn và thuật toán chuẩn hóa.

- Trong thực tế khi chuẩn hóa lược đồ CSDL thường được thực hiện theo các bước:

+ Bước 1: kiểm tra xem quan hệ đã đạt dạng chuẩn 1NF chưa? Nếu chưa ở 1NF có nghĩa là có các thuộc tính chưa nguyên tố / lặp. Tiến hành tách các thuộc tính đó.

+ Bước 2: kiểm tra xem chúng có ở dạng 2NF không? Nghĩa là kiểm tra xem các thuộc tính không khóa có phụ thuộc hoàn toàn vào khóa chính không? Tiến hành tách những PTH bộ phận đó thành các bảng con để giảm bớt sự trùng lặp thông tin.

+ Bước 3: kiểm tra xem chúng đã đạt dạng chuẩn 3NF chưa? Nghĩa là các thuộc tính không khóa thì phụ thuộc trực tiếp vào khóa chính. Tiến hành tách những PTH bậc cầu thành bảng con.

+ Bước 4: kiểm tra xem chúng đã đạt dạng chuẩn BCNF chưa? Nghĩa là tất cả các phụ thuộc hàm đều có vế trái là siêu khóa. Tiến hành tách PTH có vế trái chưa phải là siêu khóa.

4.2 Chuẩn hóa cơ sở dữ liệu quan hệ quản lý cửa hàng bán pc

CSDL Quản lý nhà sách có:

a. Quan hệ SanPham (IdSanPham, TenSanPham, ChiTiet, Gia, TinhTrang, IdDanhMuc, HangSX, TGBH, IdNSX, IdNCC, HinhAnh)

Có tập phụ thuộc hàm là $F_1 = \{ \text{IdSanPham} \Rightarrow \text{TenSanPham}, \text{IdSanPham} \Rightarrow \text{ChiTiet}, \text{IdSanPham} \Rightarrow \text{Gia}, \text{IdSanPham} \Rightarrow \text{TinhTrang}, \text{IdSanPham} \Rightarrow \text{IdDanhMuc}, \text{IdSanPham} \Rightarrow \text{HangSX}, \text{IdSanPham} \Rightarrow \text{TGBH}, \text{IdSanPham} \Rightarrow \text{IdNSX}, \text{IdSanPham} \Rightarrow \text{IdNCC}, \text{IdSanPham} \Rightarrow \text{HinhAnh} \}$

b. Quan hệ NhanVien (IdNV, TenNV, GioiTinh, NgaySinh, DiaChi, IdRole)

Có tập phụ thuộc hàm là $F_2 = \{ \text{IdNV} \Rightarrow \text{TenNV}, \text{IdNV} \Rightarrow \text{GioiTinh}, \text{IdNV} \Rightarrow \text{NgaySinh}, \text{IdNV} \Rightarrow \text{DiaChi}, \text{IdNV} \Rightarrow \text{IdRole} \}$

NgàySinh, IdNV => DiaChi, IdNV => IdRole}

- c. Quan hệ KháchHang (IdKH, TenKH, DiaChi, SDT, Email, MatKhau)
Có tập phụ thuộc hàm là $F3 = \{ IdKH \Rightarrow TenKH, IdKH \Rightarrow DiaChi, IdKH \Rightarrow SDT, IdKH \Rightarrow Email, IdKH \Rightarrow MatKhau \}$
- d. Quan hệ NhaCungCap (IdNCC, TenNCC, SDT, DiaChi)
Có tập phụ thuộc hàm là $F4 = \{ IdNCC \Rightarrow TenNCC, IdNCC \Rightarrow SDT, IdNCC \Rightarrow DiaChi \}$
- e. Quan hệ HangSanXuat (IdNSX, TenNSX, DiaChi, SDT)
Có tập phụ thuộc hàm là $F5 = \{ IdNSX \Rightarrow TenNSX, IdNSX \Rightarrow DiaChi, IdNSX \Rightarrow SDT \}$
- f. Quan hệ HoaDon (IdHD, IdKH, IdNV, NgayLapHD, NgayGiao, TongGia)
Có tập phụ thuộc hàm là $F6 = \{ IdHD \Rightarrow IdKH, IdHD \Rightarrow IdNV, IdHD \Rightarrow NgayLapHD, IdHD \Rightarrow NgayGiao, IdHD \Rightarrow TongGia \}$
- g. Quan hệ ChiTietNhap (IdSanPham, TenSanPham, SoLuong, DonGia, IdHDN)
Có tập phụ thuộc hàm là $F7 = \{ (IdSanPham, IdHDN) \Rightarrow TenSanPham, (IdSanPham, IdHDN) \Rightarrow SoLuong, (IdSanPham, IdHDN) \Rightarrow DonGia \}$
- h. Quan hệ Roles (IdRole, TenRole)
Có tập phụ thuộc hàm là $F8 = \{ IdRole \Rightarrow TenRole \}$
- i. Quan hệ DanhMucSanPham (IdDanhMuc, Ten)
Có tập phụ thuộc hàm là $F9 = \{ IdDanhMuc \Rightarrow Ten \}$

CHƯƠNG V: CÂU LỆNH SQL

- Tạo bảng roles

```
CREATE TABLE [dbo].[Roles](  
    [IdRole] [int] IDENTITY(1,1) NOT NULL primary key,  
    [name] [nvarchar](50) NULL  
)
```

- Tạo bảng NhanVien

```
CREATE TABLE [dbo].[NhanVien](  
    [IdNV] [int] IDENTITY (1,1) NOT NULL PRIMARY KEY,  
    [Ten] [nvarchar](200),  
    [GioiTinh] [nvarchar](200) ,  
    [NgàySinh] [date],  
    [DiaChi] [nvarchar](200),  
    [IdRole] [int]  
    FOREIGN KEY (IdRole) REFERENCES Roles(IdRole)  
)
```

- Tạo bảng KháchHang

```
CREATE TABLE [dbo].[KhachHang](  
    [IdKH] [int] NOT NULL primary key,  
    [TenKH] [nvarchar](50) NULL,
```



```

[SDT] [varchar](10) NULL,
[DiaChi] [nvarchar](50) NULL,
[Email] [nvarchar] (50) NULL,
[Password] [nvarchar] (20) NULL
)

```

- Tạo bảng NhaCungCap

```

CREATE TABLE [dbo].[NhaCungCap](
    [IdNCC] [int] IDENTITY(1,1) NOT NULL primary key,
    [TenNCC] [nvarchar](50) NULL,
    [SDT] [varchar](50) NULL,
    [DiaChi] [nvarchar](50) NULL
)

```

- Tạo bảng DanhMucSanPham

```

CREATE TABLE [dbo].[DanhMucSanPham] (
    [IdDanhMuc] [int] IDENTITY(1,1) NOT NULL primary key,
    [name] [nvarchar](50) NULL
)

```

- Tạo bảng HangSanXuat

```

CREATE TABLE [dbo].[HangSanXuat] (
    [IdNSX] [int] IDENTITY(1,1) NOT NULL primary key,
    [TenNSX] [nvarchar](50) NULL,
    [SDT] [varchar](50) NULL,
    [DiaChi] [nvarchar](50) NULL
)

```

- Tạo bảng SanPham

```

CREATE TABLE [dbo].[SanPham](
    [IdSanPham] [int] IDENTITY(1,1) NOT NULL primary key,
    [TenSanPham] [nvarchar](50) NULL,
    [Gia] [float] NULL,
    [ChiTiet] [nvarchar](100) NULL,
    [IdDanhMuc] [int] NULL,
    [IdNSX] [int] NULL,
    [IdNCC] [int] NULL,
    [TinhTrang] [nvarchar](50) NULL,
    [TGBH] [date] NULL,
    [HinhAnh] [nvarchar] (100) NULL
    foreign key(IdDanhMuc) references DanhMucSanPham(IdDanhMuc),
    foreign key(IdNCC) references NhaCungCap(IdNCC),
    foreign key(IdNSX) references HangSanXuat(IdNSX)
)

```

- Tạo bảng HoaDon

```

CREATE TABLE [dbo].[HoaDon](

```

```

[IdHoaDon] [int] IDENTITY(1,1) NOT NULL primary key,
[IdNV] [int] NULL,
[IdKH] [int] NULL,
[createAt] [date] NULL,
[NgayGiao] [date] NULL,
[TongGia] [float] NULL
foreign key (IdNV) references NhanVien(IdNV),
foreign key (IdKH) references KhachHang(IdKH)
)

```

- Tạo bảng HoaDonNhap

```

CREATE TABLE [dbo].[HoaDonNhap](
    [IdHDN] [int] IDENTITY(1,1) NOT NULL primary key,
    [IdNCC] [int] NULL,
    [IdNV] [int] NULL,
    [createAt] [date] NULL,
    [NgayGiao] [date] NULL
    foreign key (IdNV) references NhanVien(IdNV),
    foreign key (IdNCC) references NhaCungCap(IdNCC)
)

```

- Tạo bảng ChiTietHoaDon

```

CREATE TABLE [dbo].[ChiTietHoaDon](
    [IdHoaDon] [int] NOT NULL,
    [IdSanPham] [int] NOT NULL,
    [SoLuong] [int] NULL,
    [Gia] [float] NULL,
    primary key (IdHoaDon, IdSanPham),
    foreign key (IdSanPham) references SanPham(IdSanPham),
    foreign key (IdHoaDon) references HoaDon(IdHoaDon)
)

```

- Tạo bảng ChiTietHDN

```

CREATE TABLE [dbo].[ChiTietHDN](
    [IdSanPham] [int] NOT NULL,
    [IdHDN] [int] NOT NULL,
    [TenSanPham] [nvarchar](50) NULL,
    [SoLuong] [int] NULL,
    [Gia] [float] NULL,
    primary key (IdSanPham, IdHDN),
    foreign key (IdSanPham) references SanPham(IdSanPham),
    foreign key (IdHDN) references HoaDonNhap(IdHDN),
)

```

