

ASSIGNMENT

(Quản lý cửa hàng bán máy tính)

Nguyễn Hương Giang - Phạm Quốc Bảo

I. Phát biểu bài toán

- Mục tiêu của dự án này là xây dựng hệ thống cơ sở dữ liệu để quản lý hoạt động của cửa hàng bán máy tính, gồm quản lý sản phẩm, đơn hàng, khách hàng, kho hàng,.. một cách hiệu quả.

- Xác định thực thể

- + **Khách hàng** (mã khách hàng, tên khách hàng, số điện thoại, email)
- + **Sản phẩm** (mã sản phẩm, tên sản phẩm, giá sản phẩm, mã nhà cung cấp)
- + **Nhà cung cấp** (mã nhà cung cấp, tên nhà cung cấp, số điện thoại, email)
- + **Hóa đơn** (mã hóa đơn, mã khách hàng, mã nhân viên, ngày đặt hàng)
- + **Hóa đơn chi tiết** (mã hóa đơn chi tiết, mã hóa đơn, mã sản phẩm, số lượng)
- + **Nhân viên** (mã nhân viên, tên nhân viên, số điện thoại, email)

- Xác định mối quan hệ

Khách hàng – [1] ----- [N]-- Hóa đơn

Hóa đơn – [1] ————— [1] – Khách hàng

Hóa đơn – [1] ————— [1] – Nhân viên

Hóa đơn – [1] ————— [1] – Hóa đơn chi tiết

Nhân viên – [1] ————— [N] – Hóa đơn

II. Mô tả nghiệp vụ của hệ thống:

1. QUẢN LÝ KHÁCH HÀNG

- Thông tin khách hàng được lưu lại sau khi mua sản phẩm. Thông tin bao gồm: mã khách hàng, tên khách hàng, số điện thoại, email.

2. QUẢN LÝ HÓA ĐƠN

- Khi khách hàng có nhu cầu mua sản phẩm, nhân viên sẽ lập hóa đơn cho khách hàng. Thông tin hóa đơn bao gồm : mã hóa đơn, mã khách hàng, mã nhân viên, ngày đặt hàng. Thông tin hóa đơn chi tiết bao gồm : mã hóa đơn chi tiết, mã hóa đơn, mã sản phẩm, số lượng.

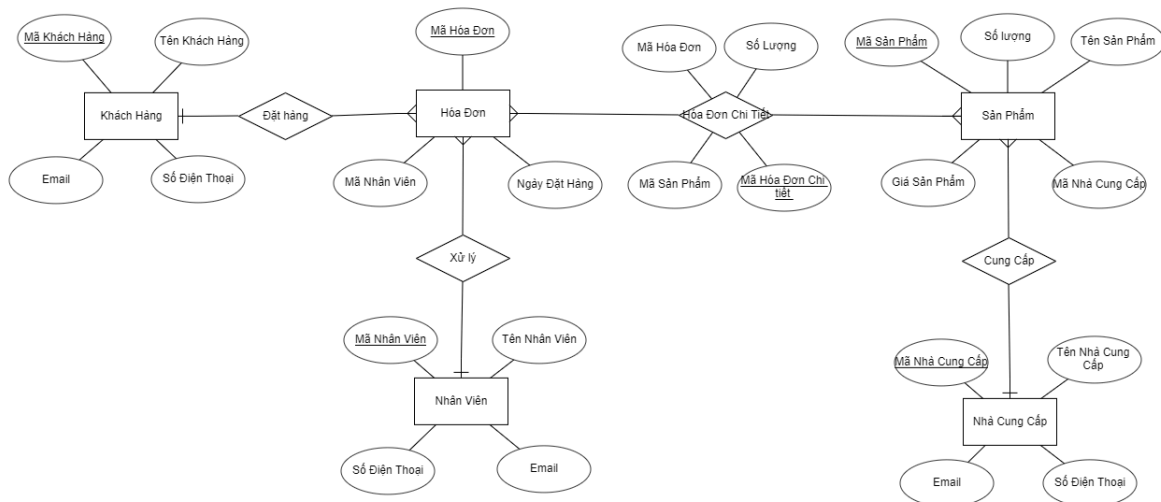
3. QUẢN LÝ SẢN PHẨM

- Sản phẩm sau khi nhập về kho sẽ được lưu lại lại thông tin vào hệ thống. Thông tin bao gồm : mã sản phẩm, tên sản phẩm, số lượng, giá sản phẩm, mã nhà cung cấp.
- Cùng với sản phẩm, thông tin về nhà cung cấp sẽ được lưu lại. Thông tin bao gồm: mã nhà cung cấp, tên nhà cung cấp, email, số điện thoại.

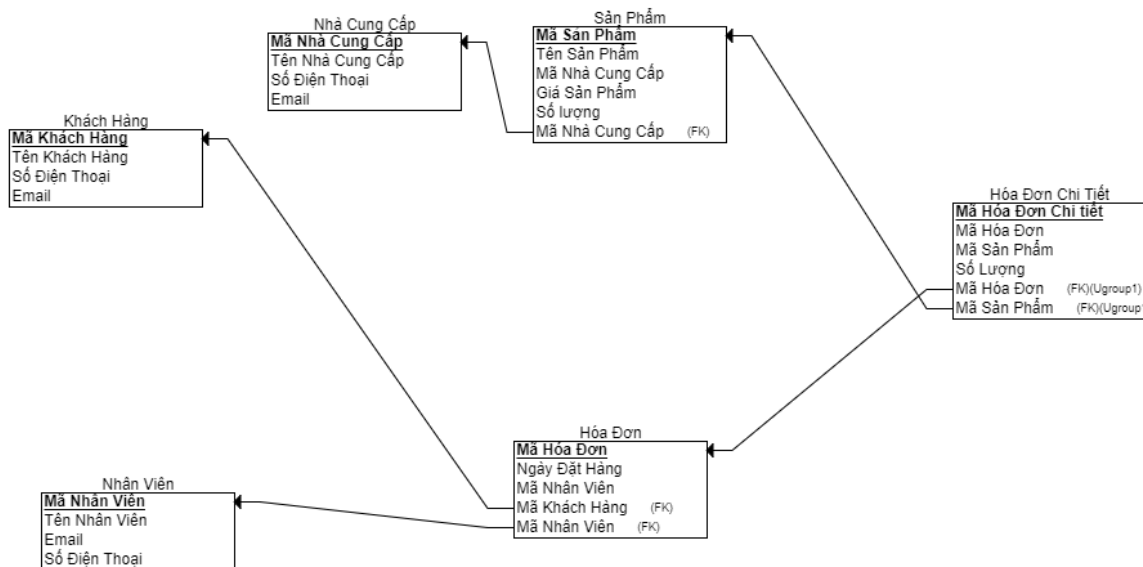
4. QUẢN LÝ NHÂN VIÊN

- Thông tin nhân viên cửa hàng bao gồm : mã nhân viên, tên nhân viên, số điện thoại, email.

III. Mô tả hình ER của hệ thống:



IV. Mô tả hình quan hệ (chuyển từ mô hình ER)



V.Đặc tả yêu cầu dữ liệu (từ điển dữ liệu) và danh sách các ràng buộc dữ liệu

1. Bảng khách hàng

Tên trường	Kiểu dl	Check	Key/Index/ Ràng buộc
maKH	Char(5)		PK
tenKH	Nvarchar(200)		Not null
SDT	Varchar(10)	Từ 0 đến 9	Not null
email	Nvarchar(200)		Not null

Ví dụ:

maKH	tenKH	SDT	email
KH001	Nguyen Van A	0973874610	A@gmail.com
KH002	Nguyen Van B	0973874611	A@gmail.com
KH003	Nguyen Van C	0973874612	B@gmail.com

Câu lệnh tạo bảng:

```
CREATE TABLE KH
(
    maKH char(5) PRIMARY KEY,
    tenKH nvarchar(200) NOT NULL,
    SDT char(10) NOT NULL CHECK(SDT LIKE
' [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] '),
    email nvarchar(200) NOT NULL
);
```

2. Bảng sản phẩm

Tên trường	Kiểu dl	Check	Key/Index/ Ràng buộc
maSP	Char(5)		PK

tenSP	Nvarchar(5)		Not null
giaSP	Int	>=0	Not null
maNCC	Char(5)		FK
soLuong	int	>=0	Not null

Ví dụ:

maSP	tenSP	giaSP	maNCC	soLuong
CP001	Lenovo ThinkPad X1 Carbon	1500000 0	B0001	30
CP002	Lenovo Yoga C930	1700000 0	B0001	30
CP003	Lenovo Legion Y740	1800000 0	B0001	30

Câu lệnh tạo bảng:

```
CREATE TABLE SP
```

```
(
```

```
    maSP char(5) PRIMARY KEY,
```

```
    tenSP nvarchar(200) NOT NULL,
```

```
    giaSP int NOT NULL CHECK(giaSP >= 0),
```

```
    soLuong int NOT NULL CHECK(soLuong >=0),
```

```
    maNCC char(5)      NOT NULL,
```

FOREIGN KEY (maNCC) REFERENCES NCC (maNCC)

);

3. Bảng nhà cung cấp

Tên trường	Kiểu dl	Check	Key/Index/Ràng buộc
maNCC	Char(5)		PK
tenNCC	Nvarchar(200)		Not null
SDT	Char(10)	Từ 0 đến 9	Unique
email	Nvarchar(200)		Unique

Ví dụ:

maNCC	tenNCC	SDT	email
B0001	Lenovo	0973874620	HaCom1@gmail.com
B0002	Dell	0973874621	HaCom2@gmail.com

G0001	Apple	097387462 3	HaCom3@gmail.com
-------	-------	----------------	------------------

Câu lệnh tạo bảng:

```
CREATE TABLE NCC
```

```
(
    maNCC char(5) PRIMARY KEY,
    tenNCC nvarchar(200) NOT NULL,
    SDT char(10) UNIQUE CHECK(SDT LIKE
' [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] [0-9] '),
    email nvarchar(200) UNIQUE
);
```

4. Bảng hóa đơn

Tên trường	Kiểu dl	Key/Index/Ràng buộc
maHoaDon	Char(7)	PK
maKH	Char(5)	FK
maNV	Char(5)	FK
ngayDatHang	Date	Not null

Ví dụ:

maHoaDon	maKH	maNV	ngayDatHang
----------	------	------	-------------

HD10001	NV001	KH001	2023-11-1
HD10002	NV001	KH002	2023-12-1
HD10003	NV001	KH003	2023-1-12

Câu lệnh tạo bảng:

```
CREATE TABLE hoaDon
(
    maHoaDon char(7) PRIMARY KEY,
    maKH char(5) NOT NULL,
    maNV char(5) NOT NULL,
    ngayDatHang DATE NOT NULL,
    FOREIGN KEY (maKH) REFERENCES KH(maKH),
    FOREIGN KEY (maNV) REFERENCES NV(maNV)
);
```

5. Bảng hóa đơn chi tiết

Tên trường	Kiểu dl	Key/Index/Ràng buộc
maHoaDonCT	Char(7)	PK
maHoaDon	Char(7)	FK
maSP	Char(5)	FK
soLuong	Int	Not null

Ví dụ:

maHoaDonCT	maHoaDon	maSP	soLuong
HD10001	HD10001	CP001	1
HD10002	HD10002	CP001	2
HD10003	HD10003	CP003	3

Câu lệnh tạo bảng:

```
CREATE TABLE hoaDonCT
(
    maHoaDonCT char(7) PRIMARY KEY,
    maHoaDon char(7) NOT NULL,
    maSP char(5) NOT NULL,
    soLuong int NOT NULL CHECK(soLuong >=0)
    FOREIGN KEY (maHoaDon) REFERENCES hoaDon(maHoaDon),
    FOREIGN KEY (maSP) REFERENCES SP(maSP)
);
```

6. Bảng nhân viên

Tên trường	Kiểu dl	Check	Key/Index/ Ràng buộc
------------	---------	-------	-------------------------

maNV	Char(5)		PK
tenNV	Nvarchar(200)		Not null
SDT	Char(10)	Từ 0 đến 9	Not null
email	Nvarchar(200)		Not null

Ví dụ:

maNV	tenNV	SDT	email
NV001	Pham Van AA	0973874610	AA@gmail.com
NV003	Pham Van BB	0773874610	BB@gmail.com
NV004	Pham Van CC	0673874610	CC@gmail.com

Câu lệnh tạo bảng:

```
CREATE TABLE NV
(
    maNV char(5) PRIMARY KEY,
    tenNV nvarchar(200) NOT NULL,
    SDT char(10) NOT NULL CHECK(SDT LIKE
'[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
    email nvarchar(200) NOT NULL
);
```

VII. Cài đặt vật lý

1. Các câu truy vấn cùng câu trả lời SQL tương ứng

- a) Lấy danh sách các hóa đơn, bao gồm mã hóa đơn, tên khách hàng, tên sản phẩm, giá sản phẩm, số lượng và tổng giá trị của mỗi hóa đơn.

```
SELECT hoaDon.maHoaDon, KH.tenKH,  
SP.tenSP, SP.giaSP, hoaDonCT.soLuong,  
SP.giaSP * hoaDonCT.soLuong AS  
TongGiaTri  
  
FROM hoaDon  
  
JOIN hoaDonCT ON hoaDon.maHoaDon =  
hoaDonCT.maHoaDon  
  
JOIN KH ON hoaDon.maKH = KH.maKH  
  
JOIN SP ON hoaDonCT.maSP = SP.maSP;
```

- b) Lấy tổng giá trị của tất cả các hóa đơn trong cửa hàng

```
SELECT SUM(SP.giaSP *  
hoaDonCT.soLuong) AS TongGiaTriHoaDon  
  
FROM hoaDonCT
```

JOIN SP ON hoaDonCT.maSP = SP.maSP;

- c) Lấy danh sách các sản phẩm mà không có trong bất kỳ hóa đơn nào và có số lượng lớn hơn 50

SELECT SP.*

FROM SP

LEFT JOIN hoaDonCT ON SP.maSP =
hoaDonCT.maSP

WHERE hoaDonCT.maSP IS NULL AND
SP.soLuong > 50;

- d) Lấy danh sách các hóa đơn và tổng giá trị của mỗi hóa đơn, sắp xếp theo giá trị giảm dần

SELECT hoaDon.maHoaDon, KH.tenKH,
SUM(SP.giaSP * hoaDonCT.soLuong) AS
TongGiaTri

FROM hoaDon

JOIN hoaDonCT ON hoaDon.maHoaDon =
hoaDonCT.maHoaDon

JOIN KH ON hoaDon.maKH = KH.maKH

```
JOIN SP ON hoaDonCT.maSP = SP.maSP  
GROUP BY hoaDon.maHoaDon, KH.tenKH  
ORDER BY TongGiaTri DESC;
```

- e) Lấy danh sách các sản phẩm mà không có trong bất kỳ hóa đơn nào, nhưng có số lượng lớn hơn 100

```
SELECT SP.*  
  
FROM SP  
  
LEFT JOIN hoaDonCT ON SP.maSP =  
hoaDonCT.maSP  
  
WHERE hoaDonCT.maSP IS NULL AND  
SP.soLuong > 100;
```

2. Các trigger để cài đặt các ràng buộc phức tạp

- a) Cập nhật số lượng sản phẩm sau khi xuất hóa đơn

```
CREATE TRIGGER CapNhatSoLuongSanPhamKhiThemHoaDon  
  
ON hoaDonCT  
  
AFTER INSERT  
  
AS  
  
BEGIN  
  
    DECLARE @maSP char(5);  
  
    DECLARE @soLuong int;
```

```

        SELECT @maSP = inserted.maSP, @soLuong =
inserted.soLuong

        FROM inserted;

UPDATE SP

SET soLuong = soLuong - @soLuong

WHERE maSP = @maSP;

END;

```

b) Kiểm tra số lượng sản phẩm còn dư trước khi thêm hóa đơn chi tiết

```

CREATE TRIGGER KiemTraSoLuongSanPham

ON hoaDonCT

INSTEAD OF INSERT

AS

BEGIN

    -- Kiểm tra số lượng sản phẩm trước khi thêm vào bảng

    IF NOT EXISTS (

        SELECT 1

        FROM inserted AS i

        JOIN SP AS p ON i.maSP = p.maSP

        WHERE i.soLuong <= p.soLuong

    )

    BEGIN

        -- Sản phẩm không đủ số lượng trong kho

```

```

        PRINT N'Sản phẩm không đủ số lượng trong kho.';

END

ELSE

BEGIN

    -- Sản phẩm có đủ số lượng trong kho, thêm vào bảng

    INSERT INTO hoaDonCT (maHoaDonCT, maHoaDon, maSP,
soLuong)

    SELECT maHoaDonCT, maHoaDon, maSP, soLuong

    FROM inserted;

END

END;

```

3. Các thủ tục và hàm để giải quyết nghiệp vụ

a) Thêm sản phẩm mới

```

CREATE PROCEDURE ThemSanPham

    @maSP char(5),

    @tenSP nvarchar(200),

    @giaSP int,

    @soLuong int,

    @maNCC char(5)

AS

BEGIN

    INSERT INTO SP (maSP, tenSP, giaSP, soLuong, maNCC)

    VALUES (@maSP, @tenSP, @giaSP, @soLuong, @maNCC);

END

```


b) Cập nhật thông tin sản phẩm

```
CREATE PROCEDURE CapNhatSanPham

    @maSP char(5),

    @tenSP nvarchar(200),

    @giaSP int,

    @soLuong int,

    @maNCC char(5)

AS

BEGIN

    UPDATE SP

    SET tenSP = @tenSP, giaSP = @giaSP, soLuong = @soLuong,
    maNCC = @maNCC

    WHERE maSP = @maSP;

END
```

c) Xóa sản phẩm

```
CREATE PROCEDURE XoaSanPham

    @maSP char(5)

AS

BEGIN

    DELETE FROM SP

    WHERE maSP = @maSP;

END
```

d) Tìm sản phẩm theo tên

```
CREATE FUNCTION TimSanPhamTheoTen  
  
(  
  
    @tenSanPham nvarchar(200)  
  
)  
  
RETURNS TABLE  
  
AS  
  
RETURN (  
  
    SELECT maSP, tenSP, giaSP, soLuong, maNCC  
  
    FROM SP  
  
    WHERE tenSP LIKE '%' + @tenSanPham + '%'  
  
);
```

e) Lập hóa đơn mới

```
CREATE PROCEDURE TaoHoaDonMoi  
  
    @maHoaDon char(7),  
  
    @maNV char(5),  
  
    @maKH char(5),  
  
    @ngayDatHang DATE  
  
AS  
  
BEGIN  
  
    INSERT INTO hoaDon (maHoaDon, maNV, maKH, ngayDatHang)  
  
    VALUES (@maHoaDon, @maNV, @maKH, @ngayDatHang);  
  
END
```

f) Thêm sản phẩm vào hóa đơn chi tiết

```
CREATE PROCEDURE ThemSanPhamVaoHoaDonCT

    @maHoaDonCT char(7),

    @maHoaDon char(7),

    @maSP char(5),

    @soLuong int

AS

BEGIN

    INSERT INTO hoaDonCT (maHoaDonCT,maHoaDon, maSP,
soLuong)

    VALUES (@maHoaDonCT,@maHoaDon, @maSP, @soLuong);

END
```

g) Tính tổng giá trị hóa đơn

```
CREATE FUNCTION TinhTongGiaTriHoaDon

(

    @maHoaDon char(7)

)

RETURNS INT

AS

BEGIN

    DECLARE @tongGiaTri INT;

    SELECT @tongGiaTri = SUM(SP.giaSP * hoaDonCT.soLuong)

    FROM SP

    JOIN hoaDonCT ON SP.maSP = hoaDonCT.maSP

    WHERE hoaDonCT.maHoaDon = @maHoaDon;
```

```
RETURN @tongGiaTri;
```

```
END
```

4. File srcipt.sql SQL

VIII. Kết luận

- Chúng ta đã xem xét quá trình thiết kế và tạo cơ sở dữ liệu SQL cho cửa hàng máy tính. Cơ sở dữ liệu này bao gồm các bảng quan trọng như Sản phẩm, Nhà cung cấp, Khách hàng, Nhân viên và Hóa đơn. Chúng ta đã đảm bảo rằng cơ sở dữ liệu này tuân theo nguyên tắc của Chia Normalization (3NF) để đảm bảo tính toàn vẹn và hiệu suất dữ liệu.
- Một số điểm quan trọng trong quá trình này bao gồm:
 - + Thiết kế các bảng với cấu trúc dữ liệu phù hợp, bao gồm các ràng buộc, kiểm tra, và các khóa ngoại để đảm bảo tính toàn vẹn dữ liệu.
 - + Sử dụng các câu truy vấn SQL để truy vấn dữ liệu từ cơ sở dữ liệu, bao gồm các truy vấn phức tạp như lấy tổng giá trị hóa đơn, tìm sản phẩm bán chạy nhất, và tìm các sản phẩm có số lượng ít.
 - + Xem xét sử dụng cơ sở dữ liệu để theo dõi danh sách các sản phẩm, hóa đơn, khách hàng và nhà cung cấp.

- + Đảm bảo tính an toàn dữ liệu bằng cách áp dụng các ràng buộc kiểm tra và sử dụng các khóa ngoại để xác minh sự nhất quán và liên kết dữ liệu.
- Cuối cùng, việc tạo cơ sở dữ liệu SQL là bước quan trọng trong quá trình xây dựng và quản lý một cửa hàng máy tính. Cơ sở dữ liệu giúp theo dõi thông tin sản phẩm, quản lý hóa đơn, tương tác với khách hàng và nhà cung cấp, và cung cấp thông tin cần thiết cho quá trình ra quyết định kinh doanh. Điều này giúp tạo ra một môi trường kinh doanh hiệu quả và cung cấp giá trị cho cả doanh nghiệp và khách hàng.