

ĐẠI HỌC QUỐC GIA HÀ NỘI TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

**Nhóm: N3**

* *Lường Minh Trí 23020440*
* *Kiều Quốc Công 23020336*
* *Hoàng Minh Vũ*

# HỆ THỐNG QUẢN LÝ BÁN HÀNG

**BÀI TẬP LỚN MÔN CƠ SỞ DỮ LIỆU**

**Ngành: Trí tuệ nhân tạo**

# HÀ NỘI - 2024

ĐẠI HỌC QUỐC GIA HÀ NỘI **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ BÁO CÁO BÀI TẬP LỚN**

**Học phần: Cơ sở dữ liệu**

# ĐỀ TÀI: CƠ SỞ DỮ LIỆU CHO QUẢN LÝ BÁN HÀNG

**Nhóm thực hiện:**

|  |  |
| --- | --- |
| Lường Minh Trí | 23020440 |
| Kiều Công Quốc | 23020336 |
| Hoàng Minh Vũ |  |

**Github Repo:** [mtreeno1/sqlProject](https://github.com/mtreeno1/sqlProject)

## Phát Biểu Bài Toán

* 1. Ý Nghĩa

Hệ thống quản lý bán hàng: Hệ thống này được thiết kế để quản lý hoạt động kinh doanh tại các cửa hàng, bao gồm quản lý thông tin mặt hàng, nhóm mặt hàng, khách hàng, nhà cung cấp, nhập hàng, và bán hàng. Mục tiêu chính là cung cấp một nền tảng quản lý hiệu quả, giúp tối ưu hóa hoạt động kinh doanh.

* 1. Các Thực Thể (Entities) Và Chức Nǎng

Dựa trên lược đồ cơ sở dữ liệu cung cấp, các thực thể trong hệ thống bao gồm:

1. *Nhóm Mặt Hàng (nhom\_mat\_hang):*
   * **Chức năng**: Quản lý danh mục nhóm hàng hoá, hỗ trợ phân loại các sản phẩm.

### Thuộc tính:

* + - ma\_nhom\_mh: Mã nhóm mặt hàng.
    - ten\_nhom\_mh: Tên nhóm mặt hàng.

1. *Mặt Hàng (mat\_hang):*
   * **Chức năng**: Quản lý danh sách các mặt hàng đang kinh doanh, bao gồm số lượng, giá bán, và nhóm.

### Thuộc tính:

* + - ma\_mh: Mã mặt hàng.
    - ten\_mh: Tên mặt hàng.
    - don\_vi\_tinh: Đơn vị tính.
    - don\_gia: Đơn giá bán.
    - ngay\_nhap: Ngày nhập kho.
    - so\_luong\_ton\_kho: Số lượng tồn kho.
    - nhom\_hang: Liên kết với ma\_nhom\_mh trong nhom\_mat\_hang.

1. *Khách Hàng (khach\_hang):*
   * **Chức năng**: Quản lý thông tin khách hàng, bao gồm thông tin cá nhân và điểm thưởng.

### Thuộc tính:

* + - ma\_kh: Mã khách hàng.
    - ten\_kh: Tên khách hàng.
    - dia\_chi: Địa chỉ.
    - so\_dt: Số điện thoại.
    - diem\_thuong: Điểm thưởng.

1. *Nhà Cung Cấp (nha\_cung\_cap):*
   * **Chức năng**: Quản lý thông tin nhà cung cấp, bao gồm địa chỉ và liên hệ.

### Thuộc tính:

* + - ma\_ncc: Mã nhà cung cấp.
    - ten\_ncc: Tên nhà cung cấp.
    - dia\_chi: Địa chỉ nhà cung cấp.
    - so\_dt: Số điện thoại.

1. *Nhập Hàng (nhap\_hang):*
   * **Chức năng**: Quản lý hoạt động nhập hàng từ nhà cung cấp.

### Thuộc tính:

* + - ma\_nhap: Mã phiếu nhập hàng.
    - ma\_ncc: Liên kết với nha\_cung\_cap.
    - ngay\_nhap: Ngày nhập hàng.
    - tong\_tien: Tổng tiền nhập hàng.

1. *Chi Tiết Nhập Hàng (chi\_tiet\_nhap\_hang):*
   * **Chức năng**: Lưu trữ chi tiết mặt hàng trong phiếu nhập.

### Thuộc tính:

* + - ma\_nhap: Liên kết phiếu nhập.
    - ma\_mh: Liên kết mặt hàng.
    - so\_luong: Số lượng nhập.
    - don\_gia\_nhap: Đơn giá nhập.
    - thanh\_tien: Thành tiền nhập.

1. *Hoá Đơn (hoa\_don):*
   * **Chức năng**: Quản lý thông tin hoá đơn bán hàng.

### Thuộc tính:

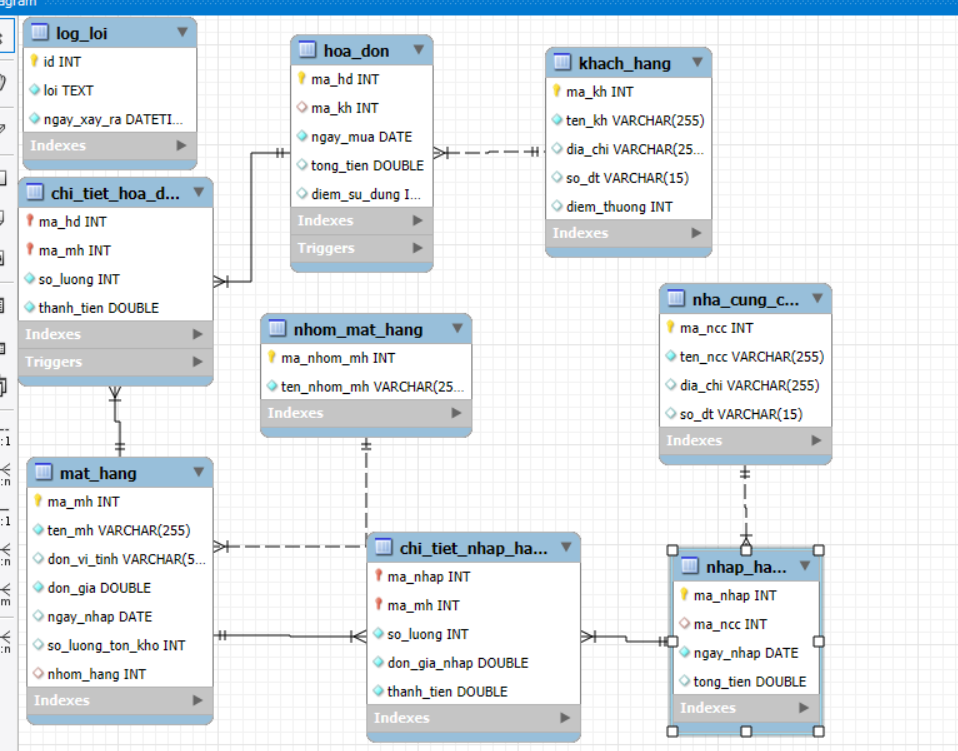
* + - ma\_hd: Mã hoá đơn.
    - ma\_kh: Liên kết khách hàng.
    - ngay\_mua: Ngày mua hàng.
    - diem\_su\_dung: Điểm sử dụng
    - tong\_tien: Tổng tiền hoá đơn.

1. *Chi Tiết Hoá Đơn (chi\_tiet\_hoa\_don):*
   * **Chức năng**: Lưu trữ chi tiết mặt hàng trong hoá đơn bán hàng.

### Thuộc tính:

* + - ma\_hd: Liên kết hoá đơn.
    - ma\_mh: Liên kết mặt hàng.
    - so\_luong: Số lượng bán.
    - thanh\_tien: Thành tiền bán.

## Mô Hình ER



* 1. Các Mối Quan Hệ:
     1. *Mặt Hàng - Nhóm Mặt Hàng:*
        + **Quan hệ**: Mỗi mặt hàng thuộc về một nhóm mặt hàng.
        + **Ràng buộc**: (n:1)
     2. *Nhập Hàng - Nhà Cung Cấp:*
        + **Quan hệ**: Mỗi phiếu nhập hàng do một nhà cung cấp cung cấp.
        + **Ràng buộc**: (n:1)
     3. *Chi Tiết Nhập Hàng - Mặt Hàng:*
        + **Quan hệ**: Mỗi phiếu nhập bao gồm nhiều mặt hàng.
        + **Ràng buộc**: (n:m)
     4. *Hoá Đơn - Khách Hàng:*
        + **Quan hệ**: Mỗi hoá đơn thuộc về một khách hàng.
        + **Ràng buộc**: (n:1)
     5. *Chi Tiết Hoá Đơn - Mặt Hàng:*
        + **Quan hệ**: Mỗi hoá đơn bán gồm nhiều mặt hàng.
        + **Ràng buộc**: (n:m)

## Kết Luận

Hệ thống quản lý bán hàng được thiết kế để hỗ trợ tối ưu hoá quá trình nhập kho, bán hàng, và quản lý kho. Các mối quan hệ được thiết kế chxặt chẽ, đảm bảo tính nhất quán và tính toàn vẹn dữ liệu.

## Mô Hình Quan Hệ

Chuyển Đổi Mô Hình ER Sang Mô Hình Quan Hệ

* 1. *Nhóm Mặt Hàng (nhom\_mat\_hang):*
* **Phụ thuộc hàm**: ma\_nhom\_mh → ten\_nhom\_mh
* **Khoá chính**: ma\_nhom\_mh
* **Giải thích**: Khi biết ma\_nhom\_mh, ta có thể biết ten\_nhom\_mh.
  1. *Mặt Hàng (mat\_hang):*
* **Phụ thuộc hàm**: ma\_mh → (ten\_mh, don\_vi\_tinh, don\_gia, ngay\_nhap, so\_luong\_ton\_kho, nhom\_hang)
* **Khoá chính**: ma\_mh
* **Giải thích**: Khi biết ma\_mh, ta có thể biết tất cả các thông tin của sản phẩm.

ma\_mh là unique

* 1. *Khách Hàng (khach\_hang):*
* **Phụ thuộc hàm**: ma\_kh → (ten\_kh, dia\_chi, so\_dt, diem\_thuong)
* **Khoá chính**: ma\_kh
* **Giải thích**: Khi biết ma\_kh, ta có thể biết thông tin liên quan đến khách hàng.
  1. *Nhà Cung Cấp (nha\_cung\_cap):*
* **Phụ thuộc hàm**: ma\_ncc → (ten\_ncc, dia\_chi, so\_dt)
* **Khoá chính**: ma\_ncc
* **Giải thích**: Khi biết ma\_ncc, ta có thể biết thông tin liên quan đến nhà cung cấp.
  1. *Nhập Hàng (nhap\_hang):*
* **Phụ thuộc hàm**: ma\_nhap → (ma\_ncc, ngay\_nhap, tong\_tien)
* **Khoá chính**: ma\_nhap
* **Giải thích**: Khi biết ma\_nhap, ta có thể biết thông tin của phiếu nhập hàng.
  1. *Chi Tiết Nhập Hàng (chi\_tiet\_nhap\_hang):*
* **Phụ thuộc hàm**: (ma\_nhap, ma\_mh) → (so\_luong, don\_gia\_nhap, thanh\_tien)
* **Khoá chính**: (ma\_nhap, ma\_mh)
* **Giải thích**: Khi biết ma\_nhap và ma\_mh, ta có thể biết thông tin chi tiết nhập hàng.
  1. *Hoá Đơn (hoa\_don):*
* **Phụ thuộc hàm**: ma\_hd → (ma\_kh, ngay\_mua, tong\_tien)
* **Khoá chính**: ma\_hd
* **Giải thích**: Khi biết ma\_hd, ta có thể biết thông tin liên quan đến hoá đơn.
  1. *Chi Tiết Hoá Đơn (chi\_tiet\_hoa\_don):*
* **Phụ thuộc hàm**: (ma\_hd, ma\_mh) → (so\_luong, thanh\_tien)
* **Khoá chính**: (ma\_hd, ma\_mh)
* **Giải thích**: Khi biết ma\_hd và ma\_mh, ta có thể biết thông tin chi tiết hoá đơn.

Chuẩn Hóa

*1NF (Dạng Chuẩn 1):*

* Tất cả bảng trong hệ thống đều đảm bảo thuộc tính ngôi ngắn (atomic attributes).
* Không có giá trị đa trị hay lồng ghép trong cột.

*2NF (Dạng Chuẩn 2):*

* Đảm bảo 1NF.
* Không có thuộc tính không khoá nào phụ thuộc một phần vào khoá chính.
  + Ví dụ: Trong bảng chi\_tiet\_nhap\_hang, tất cả thuộc tính phụ thuộc đầy đủ vào khoá chính (ma\_nhap, ma\_mh).

*3NF (Dạng Chuẩn 3):*

* Đảm bảo 2NF.
* Không có phụ thuộc bắc cầu giữa các thuộc tính không phải khoá.
  + Ví dụ: Trong bảng hoa\_don, tong\_tien phụ thuộc trực tiếp vào ma\_hd mà không qua bất kỳ thuộc tính trung gian nào.

Kết Luận

Hệ thống cơ sở dữ liệu quản lý bán hàng đã được chuẩn hoá đến dạng chuẩn 3NF, đảm bảo tính toàn vẹn dữ liệu, loại bỏ dư thừa, và tăng hiệu quả trong quản lý. Hệ thống này sẵn sàng để triển khai thực tế và mở rộngtheo nhu cầu.

# BẢNG ĐẶC TẢ YÊU CẦU DỮ LIỆU

1. **Tạo bảng**

CREATE DATABASE IF NOT EXISTS quan\_ly\_cua\_hang\_tap\_hoa;

USE quan\_ly\_cua\_hang\_tap\_hoa;

CREATE TABLE nhom\_mat\_hang (

ma\_nhom\_mh INT PRIMARY KEY AUTO\_INCREMENT,

ten\_nhom\_mh VARCHAR(255) NOT NULL

);

CREATE TABLE mat\_hang (

ma\_mh INT PRIMARY KEY AUTO\_INCREMENT,

ten\_mh VARCHAR(255) NOT NULL,

don\_vi\_tinh VARCHAR(50),

don\_gia DOUBLE NOT NULL,

ngay\_nhap DATE,

so\_luong\_ton\_kho INT DEFAULT 0,

nhom\_hang INT,

FOREIGN KEY (nhom\_hang) REFERENCES nhom\_mat\_hang(ma\_nhom\_mh)

);

CREATE TABLE khach\_hang (

ma\_kh INT PRIMARY KEY AUTO\_INCREMENT,

ten\_kh VARCHAR(255) NOT NULL,

dia\_chi VARCHAR(255),

so\_dt VARCHAR(15),

diem\_thuong INT DEFAULT 0

);

CREATE TABLE nha\_cung\_cap (

ma\_ncc INT PRIMARY KEY AUTO\_INCREMENT,

ten\_ncc VARCHAR(255) NOT NULL,

dia\_chi VARCHAR(255),

so\_dt VARCHAR(15)

);

CREATE TABLE hoa\_don (

ma\_hd INT PRIMARY KEY AUTO\_INCREMENT,

ma\_kh INT,

ngay\_mua DATE NOT NULL,

tong\_tien DOUBLE DEFAULT 0,

FOREIGN KEY (ma\_kh) REFERENCES khach\_hang(ma\_kh)

);

CREATE TABLE chi\_tiet\_hoa\_don (

ma\_hd INT,

ma\_mh INT,

so\_luong INT NOT NULL,

thanh\_tien DOUBLE NOT NULL,

PRIMARY KEY (ma\_hd, ma\_mh),

FOREIGN KEY (ma\_hd) REFERENCES hoa\_don(ma\_hd),

FOREIGN KEY (ma\_mh) REFERENCES mat\_hang(ma\_mh)

);

CREATE TABLE nhap\_hang (

ma\_nhap INT PRIMARY KEY AUTO\_INCREMENT,

ma\_ncc INT,

ngay\_nhap DATE NOT NULL,

tong\_tien DOUBLE DEFAULT 0,

FOREIGN KEY (ma\_ncc) REFERENCES nha\_cung\_cap(ma\_ncc)

);

CREATE TABLE chi\_tiet\_nhap\_hang (

ma\_nhap INT,

ma\_mh INT,

so\_luong INT NOT NULL,

don\_gia\_nhap DOUBLE NOT NULL,

thanh\_tien DOUBLE NOT NULL,

PRIMARY KEY (ma\_nhap, ma\_mh),

FOREIGN KEY (ma\_nhap) REFERENCES nhap\_hang(ma\_nhap),

FOREIGN KEY (ma\_mh) REFERENCES mat\_hang(ma\_mh)

);

ALTER TABLE hoa\_don

ADD COLUMN diem\_su\_dung INT DEFAULT 0;

ALTER TABLE mat\_hang

ADD CONSTRAINT unique\_ten\_mh\_nhom\_hang UNIQUE (ten\_mh, nhom\_hang);

-- Tạo bảng tạm để lưu thông tin tổng hợp

CREATE TEMPORARY TABLE temp\_mat\_hang AS

SELECT ten\_mh, nhom\_hang, MIN(ma\_mh) AS min\_ma\_mh, SUM(so\_luong\_ton\_kho) AS total\_so\_luong

FROM mat\_hang

GROUP BY ten\_mh, nhom\_hang;

-- Cập nhật số lượng tồn kho trong bảng chính dựa trên bảng tạm

UPDATE mat\_hang m1

JOIN temp\_mat\_hang t

ON m1.ma\_mh = t.min\_ma\_mh

SET m1.so\_luong\_ton\_kho = t.total\_so\_luong;

SET SQL\_SAFE\_UPDATES = 0;

-- Xóa các hàng trùng lặp (không phải hàng có mã nhỏ nhất)

DELETE m1

FROM mat\_hang m1

LEFT JOIN temp\_mat\_hang t

ON m1.ma\_mh = t.min\_ma\_mh

WHERE t.min\_ma\_mh IS NULL;

SELECT \* FROM mat\_hang;

1. **Chèn dữ liệu**

INSERT INTO nhom\_mat\_hang (ten\_nhom\_mh)

VALUES

('Đồ uống'),

('Bánh kẹo'),

('Gia vị'),

('Hóa mỹ phẩm'),

('Đồ hộp'),

('Đồ gia dụng'),

('Đồ đông lạnh'),

('Sữa và chế phẩm'),

('Đồ khô'),

('Rau củ quả'),

('Thịt cá tươi sống'),

('Thức ăn nhanh'),

('Gia cầm trứng'),

('Đồ dùng học tập'),

('Thức ăn cho thú cưng');

SELECT \* from mat\_hang where nhom\_hang = 1;

INSERT INTO mat\_hang (ten\_mh, don\_vi\_tinh, don\_gia, ngay\_nhap, so\_luong\_ton\_kho, nhom\_hang)

VALUES

('Coca-Cola', 'Lon', 10000, '2024-06-01', 100, 1),

('Pepsi', 'Lon', 9500, '2024-06-02', 120, 1),

('Bánh Oreo', 'Hộp', 25000, '2024-06-02', 150, 2),

('Muối i-ốt', 'Gói', 5000, '2024-06-02', 200, 3),

('Dầu ăn Simply', 'Chai', 45000, '2024-06-03', 80, 3),

('Xà phòng Lifebuoy', 'Bánh', 12000, '2024-06-04', 90, 4),

('Mì tôm Hảo Hảo', 'Thùng', 96000, '2024-06-05', 60, 5),

('Sữa Vinamilk', 'Hộp', 12000, '2024-06-05', 150, 8),

('Bột giặt Omo', 'Gói', 35000, '2024-06-06', 75, 4),

('Cá basa tươi', 'Kg', 75000, '2024-06-06', 50, 11),

('Thịt gà ta', 'Kg', 120000, '2024-06-07', 60, 11),

('Rau cải xanh', 'Kg', 15000, '2024-06-07', 100, 10),

('Bánh Chocopie', 'Hộp', 35000, '2024-06-07', 85, 2),

('Sữa đặc Ông Thọ', 'Lon', 20000, '2024-06-08', 90, 8),

('Cà phê G7', 'Hộp', 55000, '2024-06-08', 70, 1);

INSERT INTO mat\_hang (ten\_mh, don\_vi\_tinh, don\_gia, ngay\_nhap, so\_luong\_ton\_kho, nhom\_hang)

VALUES ('Pepsi', 'Lon', 9500, '2024-06-02', 1200, 1);

SELECT \* FROM mat\_hang;

INSERT INTO khach\_hang (ten\_kh, dia\_chi, so\_dt, diem\_thuong)

VALUES

('Nguyen Van A', '123 Đường ABC, TP.HCM', '0123456789', 10),

('Tran Thi B', '456 Đường XYZ, Hà Nội', '0987654321', 20),

('Le Van C', '789 Đường LMN, Đà Nẵng', '0345678901', 15),

('Pham Van D', '321 Đường UVW, Hải Phòng', '0765432198', 25),

('Hoang Thi E', '654 Đường DEF, Cần Thơ', '0981122334', 30),

('Bui Van F', '987 Đường GHI, Bình Dương', '0356789123', 40),

('Nguyen Thi G', '741 Đường JKL, Nha Trang', '0777899102', 10),

('Vu Thi H', '852 Đường MNO, Vũng Tàu', '0888999222', 35),

('Tran Van I', '963 Đường PQR, Đà Lạt', '0912233445', 20),

('Phan Thi K', '159 Đường STU, Huế', '0345566778', 50),

('Nguyen Van L', '753 Đường VWX, Buôn Ma Thuột', '0707070707', 45),

('Dang Thi M', '369 Đường YZ, Quảng Ninh', '0901234567', 15),

('Ho Van N', '951 Đường ABC, Biên Hòa', '0987001122', 25),

('Tran Thi O', '147 Đường DEF, Long An', '0978003344', 35),

('Ngo Van P', '258 Đường GHI, Tây Ninh', '0944005566', 10);

INSERT INTO nha\_cung\_cap (ten\_ncc, dia\_chi, so\_dt)

VALUES

('Công ty TNHH ABC', 'KCN Bình Dương', '0274123456'),

('Công ty CP XYZ', 'KCN Long An', '0274234567'),

('Công ty TNHH Thực Phẩm', 'KCN Đồng Nai', '0251357890'),

('Nhà phân phối Minh Phát', 'TP.HCM', '0281234567'),

('Công ty Gia Vị Việt', 'Hà Nội', '0249876543'),

('Công ty TNHH Sữa Việt Nam', 'TP.HCM', '0285678901'),

('Công ty Bánh Kẹo Bình An', 'Đà Nẵng', '0236781234'),

('Nhà cung cấp Fresh Food', 'Cần Thơ', '0291234987'),

('Công ty Xuất Nhập Khẩu Nam Định', 'Nam Định', '0228123456'),

('Nhà phân phối Green Market', 'Hải Phòng', '0225781234'),

('Công ty Hóa Mỹ Phẩm An Khang', 'Bình Dương', '0271234789'),

('Công ty Nông Sản Việt', 'Đồng Tháp', '0675432198'),

('Công ty Bánh Kẹo SweetHome', 'Nha Trang', '0258123456'),

('Công ty TNHH Thức Ăn Chăn Nuôi', 'Long An', '0278123456'),

('Công ty Thủy Hải Sản Thanh Long', 'Quảng Ninh', '0203123987');

INSERT INTO nhap\_hang (ma\_ncc, ngay\_nhap, tong\_tien)

VALUES

(1, '2024-06-01', 1000000),

(2, '2024-06-02', 2000000),

(3, '2024-06-03', 1500000),

(4, '2024-06-04', 800000),

(5, '2024-06-05', 1200000),

(6, '2024-06-06', 950000),

(7, '2024-06-07', 1100000),

(8, '2024-06-08', 1300000),

(9, '2024-06-09', 1700000),

(10, '2024-06-10', 900000),

(11, '2024-06-11', 750000),

(12, '2024-06-12', 680000),

(13, '2024-06-13', 990000),

(14, '2024-06-14', 1200000),

(15, '2024-06-15', 2100000);

INSERT INTO hoa\_don (ma\_kh, ngay\_mua, tong\_tien)

VALUES

(1, '2024-06-10', 300000),

(2, '2024-06-11', 450000),

(3, '2024-06-12', 120000),

(4, '2024-06-13', 600000),

(5, '2024-06-14', 750000),

(6, '2024-06-15', 200000),

(7, '2024-06-16', 900000),

(8, '2024-06-17', 150000),

(9, '2024-06-18', 320000),

(10, '2024-06-19', 700000),

(11, '2024-06-20', 800000),

(12, '2024-06-21', 1000000),

(13, '2024-06-22', 110000),

(14, '2024-06-23', 500000),

(15, '2024-06-24', 650000);

INSERT INTO chi\_tiet\_hoa\_don (ma\_hd, ma\_mh, so\_luong, thanh\_tien)

VALUES

(1, 1, 10, 100000),

(1, 2, 5, 47500),

(2, 3, 4, 100000),

(2, 4, 10, 50000),

(3, 5, 2, 90000),

(4, 6, 8, 96000),

(4, 7, 2, 192000),

(5, 8, 5, 60000),

(6, 9, 3, 105000),

(7, 10, 10, 750000),

(8, 11, 1, 120000),

(9, 12, 3, 45000),

(10, 13, 4, 140000),

(11, 14, 5, 100000),

(12, 15, 2, 110000);

INSERT INTO chi\_tiet\_nhap\_hang (ma\_nhap, ma\_mh, so\_luong, don\_gia\_nhap, thanh\_tien)

VALUES

(1, 1, 100, 9000, 900000),

(1, 2, 50, 8700, 435000),

(2, 3, 150, 22000, 3300000),

(2, 4, 200, 4500, 900000),

(3, 5, 80, 40000, 3200000),

(3, 6, 90, 10000, 900000),

(4, 7, 60, 90000, 5400000),

(4, 8, 150, 10000, 1500000),

(5, 9, 75, 30000, 2250000),

(6, 10, 50, 70000, 3500000),

(7, 11, 60, 110000, 6600000),

(8, 12, 100, 14000, 1400000),

(9, 13, 85, 33000, 2805000),

(10, 14, 90, 18000, 1620000),

(11, 15, 70, 50000, 3500000);

SELECT \* FROM mat\_hang;

1. **Thực hiện truy vấn.**

USE quan\_ly\_cua\_hang\_tap\_hoa;

# Câu 1: Danh sách khách hàng mua mặt hàng thuộc nhóm "Đồ uống" trong khoảng 01/06/2024 - 15/06/2024

SELECT DISTINCT KH.ten\_kh, KH.dia\_chi

FROM khach\_hang KH

JOIN hoa\_don HD ON KH.ma\_kh = HD.ma\_kh

JOIN chi\_tiet\_hoa\_don CTHD ON HD.ma\_hd = CTHD.ma\_hd

JOIN mat\_hang MH ON CTHD.ma\_mh = MH.ma\_mh

JOIN nhom\_mat\_hang NH ON MH.nhom\_hang = NH.ma\_nhom\_mh

WHERE NH.ten\_nhom\_mh = 'Đồ uống'

AND HD.ngay\_mua BETWEEN '2024-06-01' AND '2024-06-15';

# Câu 2: Tìm mặt hàng bán chạy nhất (doanh thu lớn nhất)

SELECT MH.ten\_mh, SUM(CTHD.thanh\_tien) AS doanh\_thu

FROM chi\_tiet\_hoa\_don CTHD

JOIN mat\_hang MH ON CTHD.ma\_mh = MH.ma\_mh

GROUP BY MH.ten\_mh

ORDER BY doanh\_thu DESC

LIMIT 1;

# Câu 3: Đếm số lượng mặt hàng thuộc nhóm "Gia vị"

SELECT COUNT(\*) AS so\_luong\_mat\_hang

FROM mat\_hang MH

JOIN nhom\_mat\_hang NH ON MH.nhom\_hang = NH.ma\_nhom\_mh

WHERE NH.ten\_nhom\_mh = 'Gia vị';

# Câu 4: Khách hàng có tổng giá trị mua hàng lớn nhất từ 01/06/2024 đến 15/06/2024

SELECT KH.ten\_kh, KH.dia\_chi, KH.so\_dt, SUM(HD.tong\_tien) AS tong\_gia\_tri

FROM khach\_hang KH

JOIN hoa\_don HD ON KH.ma\_kh = HD.ma\_kh

WHERE HD.ngay\_mua BETWEEN '2024-06-01' AND '2024-06-15'

GROUP BY KH.ten\_kh, KH.dia\_chi, KH.so\_dt

ORDER BY tong\_gia\_tri DESC

LIMIT 1;

# Câu 5: Xem danh sách nhập hàng từ nhà cung cấp cụ thể

SELECT NH.ma\_nhap, NH.ngay\_nhap, NH.tong\_tien, NCC.ten\_ncc

FROM nhap\_hang NH

JOIN nha\_cung\_cap NCC ON NH.ma\_ncc = NCC.ma\_ncc

WHERE NH.ma\_ncc = 1

ORDER BY NH.ngay\_nhap DESC;

# Câu 6: Danh sách các nhóm mặt hàng hiện có

SELECT \*

FROM nhom\_mat\_hang;

# Câu 7: Danh sách tất cả các phiếu nhập hàng

SELECT NH.ma\_nhap, NH.ngay\_nhap, NH.tong\_tien, NCC.ten\_ncc

FROM nhap\_hang NH

JOIN nha\_cung\_cap NCC ON NH.ma\_ncc = NCC.ma\_ncc

ORDER BY NH.ngay\_nhap DESC;

# Câu 8: Danh sách chi tiết của các phiếu nhập hàng

SELECT CTHN.ma\_nhap, MH.ten\_mh, CTHN.so\_luong, CTHN.don\_gia\_nhap, CTHN.thanh\_tien

FROM chi\_tiet\_nhap\_hang CTHN

JOIN mat\_hang MH ON CTHN.ma\_mh = MH.ma\_mh

ORDER BY CTHN.ma\_nhap;

# Câu 9: Tổng doanh thu theo từng nhóm mặt hàng

SELECT NH.ten\_nhom\_mh, SUM(CTHD.thanh\_tien) AS tong\_doanh\_thu

FROM chi\_tiet\_hoa\_don CTHD

JOIN mat\_hang MH ON CTHD.ma\_mh = MH.ma\_mh

JOIN nhom\_mat\_hang NH ON MH.nhom\_hang = NH.ma\_nhom\_mh

GROUP BY NH.ten\_nhom\_mh

ORDER BY tong\_doanh\_thu DESC;

# Câu 10: Tổng số lượng hàng nhập theo từng nhà cung cấp

SELECT NCC.ten\_ncc, SUM(CTNH.so\_luong) AS tong\_so\_luong

FROM chi\_tiet\_nhap\_hang CTNH

JOIN nhap\_hang NH ON CTNH.ma\_nhap = NH.ma\_nhap

JOIN nha\_cung\_cap NCC ON NH.ma\_ncc = NCC.ma\_ncc

GROUP BY NCC.ten\_ncc

ORDER BY tong\_so\_luong DESC;

# Câu 11: Khách hàng mua nhiều nhất theo số lượng hàng

SELECT KH.ten\_kh, SUM(CTHD.so\_luong) AS tong\_so\_luong

FROM khach\_hang KH

JOIN hoa\_don HD ON KH.ma\_kh = HD.ma\_kh

JOIN chi\_tiet\_hoa\_don CTHD ON HD.ma\_hd = CTHD.ma\_hd

GROUP BY KH.ten\_kh

ORDER BY tong\_so\_luong DESC

LIMIT 1;

# Câu 12: Mặt hàng có số lượng tồn kho ít nhất

SELECT MH.ten\_mh, MH.so\_luong\_ton\_kho

FROM mat\_hang MH

ORDER BY MH.so\_luong\_ton\_kho ASC

LIMIT 1;

# Câu 13: Tổng tiền nhập hàng trong một ngày cụ thể

SELECT NH.ngay\_nhap, SUM(NH.tong\_tien) AS tong\_tien\_nhap

FROM nhap\_hang NH

WHERE NH.ngay\_nhap = '2024-06-02'

GROUP BY NH.ngay\_nhap;

# Câu 14: Doanh thu từ một mặt hàng cụ thể

SELECT MH.ten\_mh, SUM(CTHD.thanh\_tien) AS doanh\_thu

FROM chi\_tiet\_hoa\_don CTHD

JOIN mat\_hang MH ON CTHD.ma\_mh = MH.ma\_mh

WHERE MH.ten\_mh = 'Pepsi'

GROUP BY MH.ten\_mh;

VI) Trigger:

USE quan\_ly\_cua\_hang\_tap\_hoa;

# Trigger 1: Cập nhật điểm thưởng khi khách hàng có giao dịch

DELIMITER //

CREATE TRIGGER cap\_nhat\_diem\_thuong

AFTER INSERT ON hoa\_don

FOR EACH ROW

BEGIN

-- Tính số điểm thưởng dựa trên tổng tiền hoá đơn

DECLARE diem\_thuong\_moi INT;

SET diem\_thuong\_moi = FLOOR(NEW.tong\_tien / 10000);

-- Cập nhật điểm thưởng cho khách hàng

UPDATE khach\_hang

SET diem\_thuong = diem\_thuong + diem\_thuong\_moi

WHERE ma\_kh = NEW.ma\_kh;

END;

//

DELIMITER ;

# Trigger 2: Sử dụng điểm thưởng để giảm giá

DELIMITER //

CREATE TRIGGER su\_dung\_diem\_thuong

BEFORE INSERT ON hoa\_don

FOR EACH ROW

BEGIN

DECLARE diem\_hien\_tai INT;

DECLARE tien\_giam INT;

-- Lấy số điểm thưởng hiện tại của khách hàng

SELECT diem\_thuong INTO diem\_hien\_tai

FROM khach\_hang

WHERE ma\_kh = NEW.ma\_kh;

-- Đảm bảo khách hàng không sử dụng quá số điểm họ có

IF NEW.diem\_su\_dung > diem\_hien\_tai THEN

SET NEW.diem\_su\_dung = diem\_hien\_tai;

END IF;

-- Tính số tiền giảm giá dựa trên điểm thưởng (1 điểm = 100 VNĐ)

SET tien\_giam = NEW.diem\_su\_dung \* 100;

-- Nếu số tiền giảm lớn hơn tổng tiền, điều chỉnh điểm sử dụng

IF tien\_giam > NEW.tong\_tien THEN

SET NEW.diem\_su\_dung = FLOOR(NEW.tong\_tien / 100);

SET tien\_giam = NEW.diem\_su\_dung \* 100;

END IF;

-- Cập nhật tổng tiền hóa đơn sau khi giảm giá

SET NEW.tong\_tien = NEW.tong\_tien - tien\_giam;

-- Cập nhật điểm thưởng của khách hàng

UPDATE khach\_hang

SET diem\_thuong = diem\_thuong - NEW.diem\_su\_dung

WHERE ma\_kh = NEW.ma\_kh;

END;

//

DELIMITER ;

# Xem điểm thưởng cập nhật sau khi thêm 1 giao dịch

SELECT ma\_kh, ten\_kh, diem\_thuong

FROM khach\_hang;

INSERT INTO hoa\_don (ma\_kh, ngay\_mua, tong\_tien)

VALUES (1, '2024-12-26', 250000);

# Xem điểm sử dụng và tiền thanh toán sau khi giảm

INSERT INTO hoa\_don (ma\_kh, ngay\_mua, tong\_tien, diem\_su\_dung)

VALUES (1, '2024-12-26', 200000, 30);

SELECT \*

FROM hoa\_don

WHERE ma\_hd = (SELECT MAX(ma\_hd) FROM hoa\_don);

# Xem hoá đơn khách hàng cụ thể

SELECT \*

FROM hoa\_don

WHERE ma\_kh = 1

ORDER BY ngay\_mua DESC;

# Triger 3: Tự động xử lý việc kiểm tra, thêm hoặc cập nhật

DELIMITER //

CREATE PROCEDURE them\_hoac\_cap\_nhat\_mat\_hang(

IN p\_ten\_mh VARCHAR(255),

IN p\_don\_vi\_tinh VARCHAR(50),

IN p\_don\_gia DOUBLE,

IN p\_ngay\_nhap DATE,

IN p\_so\_luong INT,

IN p\_nhom\_hang INT

)

BEGIN

IF EXISTS (

SELECT 1 FROM mat\_hang

WHERE ten\_mh = p\_ten\_mh AND nhom\_hang = p\_nhom\_hang

) THEN

-- Nếu mặt hàng đã tồn tại, cập nhật số lượng tồn kho

UPDATE mat\_hang

SET so\_luong\_ton\_kho = so\_luong\_ton\_kho + p\_so\_luong

WHERE ten\_mh = p\_ten\_mh AND nhom\_hang = p\_nhom\_hang;

ELSE

-- Nếu mặt hàng chưa tồn tại, thêm mới

INSERT INTO mat\_hang (ten\_mh, don\_vi\_tinh, don\_gia, ngay\_nhap, so\_luong\_ton\_kho, nhom\_hang)

VALUES (p\_ten\_mh, p\_don\_vi\_tinh, p\_don\_gia, p\_ngay\_nhap, p\_so\_luong, p\_nhom\_hang);

END IF;

END;

//

DELIMITER ;

CALL them\_hoac\_cap\_nhat\_mat\_hang('Pepsi', 'Lon', 9500, '2024-06-02', 1200, 1);

SELECT \* FROM mat\_hang;

# Trigger 4

DELIMITER //

CREATE TRIGGER cap\_nhat\_ton\_kho\_sau\_mua\_hang

BEFORE INSERT ON chi\_tiet\_hoa\_don

FOR EACH ROW

BEGIN

DECLARE so\_luong\_kho INT;

-- Lấy số lượng tồn kho hiện tại

SELECT so\_luong\_ton\_kho INTO so\_luong\_kho

FROM mat\_hang

WHERE ma\_mh = NEW.ma\_mh;

-- Kiểm tra nếu tồn kho không đủ

IF NEW.so\_luong > so\_luong\_kho THEN

-- Lưu thông tin lỗi vào bảng log

INSERT INTO log\_kiem\_tra (ma\_mh, so\_luong\_dat, so\_luong\_ton, trang\_thai, ngay\_ghi)

VALUES (NEW.ma\_mh, NEW.so\_luong, so\_luong\_kho, 'Thất bại: Không đủ hàng', NOW());

-- Dừng thêm chi tiết hóa đơn

SET NEW.so\_luong = 0; -- Không cho phép thêm

ELSE

-- Trừ số lượng tồn kho nếu đủ hàng

UPDATE mat\_hang

SET so\_luong\_ton\_kho = so\_luong\_ton\_kho - NEW.so\_luong

WHERE ma\_mh = NEW.ma\_mh;

-- Lưu thông tin thành công vào bảng log

INSERT INTO log\_kiem\_tra (ma\_mh, so\_luong\_dat, so\_luong\_ton, trang\_thai, ngay\_ghi)

VALUES (NEW.ma\_mh, NEW.so\_luong, so\_luong\_kho, 'Thành công', NOW());

END IF;

END;

//

DELIMITER ;

CREATE TABLE log\_kiem\_tra (

id INT AUTO\_INCREMENT PRIMARY KEY,

ma\_mh INT,

so\_luong\_dat INT,

so\_luong\_ton INT,

trang\_thai VARCHAR(50),

ngay\_ghi DATETIME

);

SELECT \* FROM log\_kiem\_tra;

VII) Procedure

START TRANSACTION;

-- 1. Kiểm tra tồn kho

SELECT so\_luong\_ton\_kho INTO @so\_luong\_kho

FROM mat\_hang

WHERE ma\_mh = 2;

-- Kiểm tra nếu số lượng tồn kho không đủ

IF @so\_luong\_kho < 10 THEN

-- Rollback và báo lỗi

ROLLBACK;

SELECT CONCAT('Không đủ hàng trong kho cho mặt hàng mã: ', 2) AS Loi;

ELSE

-- 2. Tạo hóa đơn

INSERT INTO hoa\_don (ma\_kh, ngay\_mua, tong\_tien, diem\_su\_dung)

VALUES (1, '2024-12-26', 0, 10);

SET @ma\_hd = LAST\_INSERT\_ID();

-- 3. Thêm chi tiết hóa đơn

INSERT INTO chi\_tiet\_hoa\_don (ma\_hd, ma\_mh, so\_luong, thanh\_tien)

VALUES (@ma\_hd, 2, 10, 95000);

-- 4. Trừ tồn kho

UPDATE mat\_hang

SET so\_luong\_ton\_kho = so\_luong\_ton\_kho - 10

WHERE ma\_mh = 2;

-- 5. Tính tổng tiền hóa đơn

UPDATE hoa\_don

SET tong\_tien = (SELECT SUM(thanh\_tien) FROM chi\_tiet\_hoa\_don WHERE ma\_hd = @ma\_hd)

WHERE ma\_hd = @ma\_hd;

-- Commit giao dịch

COMMIT;

END IF;