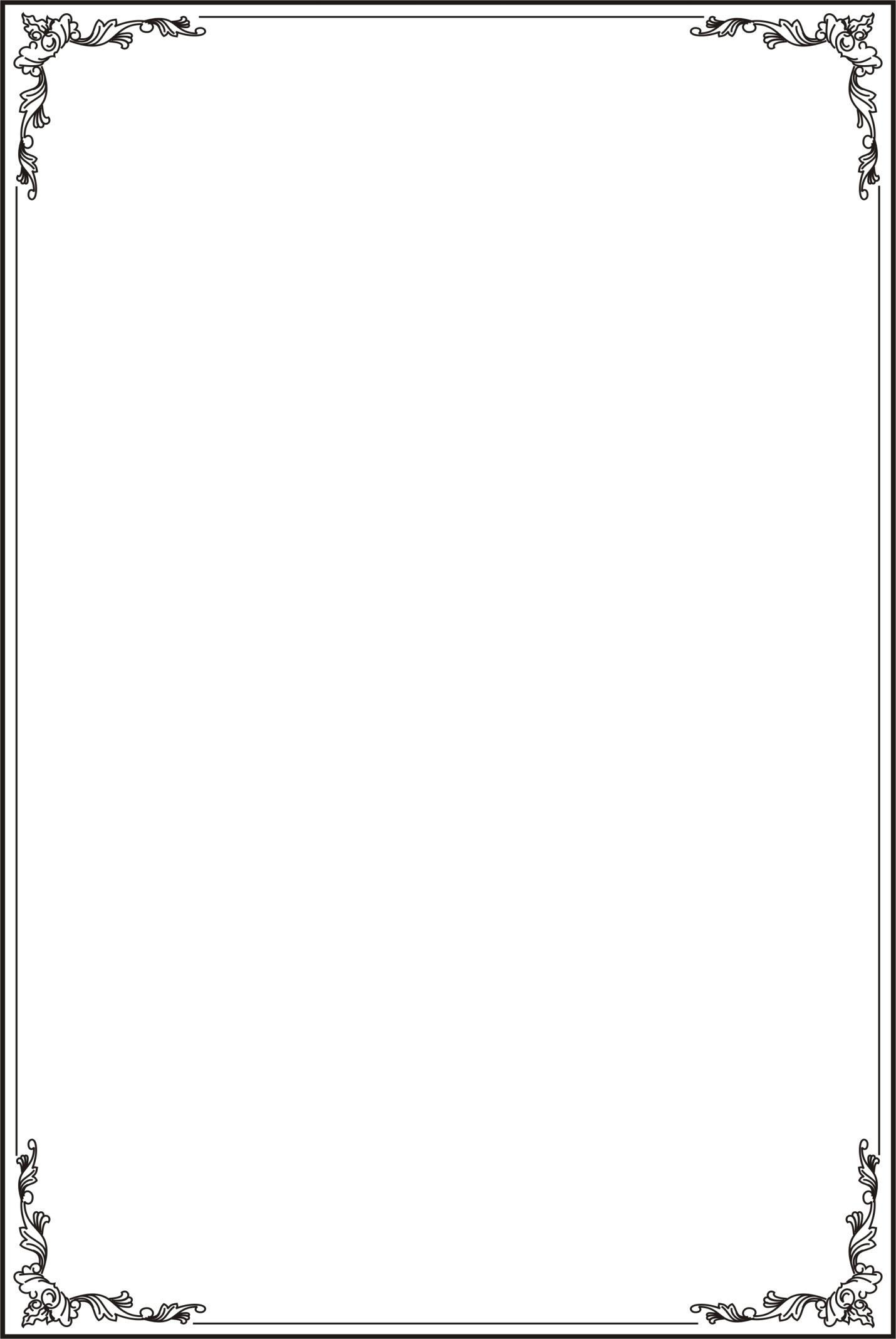
BỘ THÔNG TIN VÀ TRUYỀN THÔNG

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

Logo

Description automatically generated

BÁO CÁO BÀI TẬP LỚN

**HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**THIẾT KẾ VÀ QUẢN LÝ**

**CƠ SỞ DỮ LIỆU QUẢN LÝ QUÁN CAFE**

**Lớp:** E22CQCN01-B

**Thành viên:**

Đỗ Hải Đăng – B22DCAT082

Lê Mậu Hải Đăng – B22DCAT083

Nguyễn Duy Hải Đăng – B22DCVT138

Nguyễn Minh Đức – B22DCCN234

HÀ NỘI, THÁNG 04/2025

1. **Bối cảnh và nguyên nhân đề xuất**

Trong bối cảnh nền kinh tế thị trường ngày càng cạnh tranh gay gắt, đặc biệt là trong lĩnh vực kinh doanh dịch vụ ẩm thực, việc quản lý hiệu quả và khoa học là chìa khóa sống còn quyết định thành công hay thất bại của mỗi doanh nghiệp. Quán cafe, với đặc thù là dịch vụ có tính tương tác cao và nhu cầu quản lý dữ liệu đa dạng, phức tạp, đòi hỏi phải có một hệ thống quản lý thông tin nhanh chóng, chính xác và đáng tin cậy. Do vậy, **cơ sở dữ liệu quản lý quán cafe** ra đời không chỉ đơn thuần đóng vai trò là một công cụ lưu trữ dữ liệu thông thường, mà còn được xem là một giải pháp tối ưu giúp các nhà quản lý vận hành quán một cách bài bản và chuyên nghiệp.

Hệ cơ sở dữ liệu này góp phần hiện thực hóa khả năng quản trị toàn diện thông qua việc quản lý chặt chẽ các thông tin quan trọng như thông tin khách hàng, quản lý nguồn nguyên liệu, kiểm soát doanh thu, chi phí và hoạt động của đội ngũ nhân viên. Từ đó, các nhà quản lý có thể đưa ra những quyết định đúng đắn, kịp thời và chính xác, góp phần nâng cao hiệu quả vận hành, tối ưu hóa lợi nhuận, và gia tăng sức cạnh tranh trên thị trường.

Chính vì những lý do quan trọng ấy, việc nghiên cứu, xây dựng và triển khai thành công **hệ cơ sở dữ liệu quản lý quán cafe** không chỉ là nhiệm vụ học thuật mà còn mang ý nghĩa thực tiễn sâu sắc, góp phần tích cực vào sự phát triển bền vững và thành công lâu dài của doanh nghiệp trong tương lai.

1. **Các nhiệm vụ trọng yếu cần triển khai**

Để xây dựng thành công hệ thống cơ sở dữ liệu quản lý quán cafe một cách khoa học và có hiệu quả cao, cần tiến hành thực hiện một cách đầy đủ, nghiêm ngặt và có tính hệ thống các bước sau đây:

**1. Khảo sát, phân tích yêu cầu và xác định phạm vi quản lý**

Bước đầu tiên và quan trọng nhất khi xây dựng cơ sở dữ liệu là xác định rõ ràng các yêu cầu về quản lý dữ liệu. Cụ thể, cần:

* Nghiên cứu và làm rõ các nghiệp vụ quản lý đặc thù của quán cafe, như quản lý nhân sự, khách hàng, đồ uống, nguyên liệu, nhà cung cấp, và hóa đơn bán hàng.
* Xác định các thao tác cần được hỗ trợ bởi hệ thống, bao gồm tạo lập, cập nhật, xóa và truy xuất thông tin.
* Thu thập yêu cầu từ thực tế, các ràng buộc logic, nghiệp vụ và các quy trình quản lý nhằm đảm bảo hệ thống phản ánh đúng và đầy đủ các hoạt động quản lý.

**2. Xây dựng mô hình thực thể - liên kết (Entity-Relationship Model)**

Sau khi phân tích yêu cầu, cần xây dựng một mô hình thực thể-liên kết (E-R) nhằm trực quan hóa cấu trúc dữ liệu và mối quan hệ giữa các thành phần dữ liệu, bao gồm:

* Xác định và mô tả rõ các thực thể chính: Cafe House, Nhân viên, Khách hàng, Đồ uống, Nguyên liệu, Nhà cung cấp, Hóa đơn...
* Xác định các thuộc tính đặc trưng cho từng thực thể, lựa chọn các thuộc tính định danh (khóa chính - khóa ngoại).
* Mô tả chi tiết các mối quan hệ giữa các thực thể, bao gồm các loại liên kết như một-nhiều(1-n), nhiều-nhiều (n-n), thực thể yếu và thực thể con .

Mô tả chi tiết lược đồ:

Có tất cả 14 thực thể trong đó bao gồm 7 tập thực thể mạnh, 1 tập thực thể yếu, 3 tập thực thể con, và 3 thực thể liên kết.

● Thực thể mạnh: Cafe house, Hóa đơn, Khách hàng, Đồ uống, Nguyên liệu, Nhà cung cấp, Nhân viên.

● Thực thể yếu: Ca làm.

● Thực thể cha: Nhân viên

● Thực thể con: Bồi bàn, Quản lý, Pha chế.

● Thực thể liên kết: Phục vụ, Cung cấp, Thanh toán

**3. Thiết lập ràng buộc dữ liệu và đảm bảo toàn vẹn**

Sau khi xác định mô hình dữ liệu, cần xây dựng các ràng buộc nhằm đảm bảo tính nhất quán và chính xác của cơ sở dữ liệu, bao gồm:

* **Ràng buộc toàn vẹn thực thể**: Mỗi bảng phải có khóa chính, đảm bảo mỗi bản ghi là duy nhất và không được phép null.
* **Ràng buộc toàn vẹn tham chiếu**: Các khóa ngoại phải tham chiếu hợp lệ đến khóa chính ở bảng liên quan, đảm bảo sự nhất quán
* **Ràng buộc nghiệp vụ**: Áp dụng các quy tắc đặc thù như: mỗi nhân viên chỉ làm việc tại một quán, mỗi hóa đơn phải gắn với một khách hàng và một quán cụ thể, nguyên liệu có thể được cung cấp từ nhiều nhà cung cấp...

**4. Ánh xạ từ mô hình thực thể-liên kết sang lược đồ quan hệ**

Từ mô hình thực thể-liên kết đã xây dựng, tiến hành chuyển đổi sang mô hình dữ liệu quan hệ, bao gồm:

* Chuyển đổi các thực thể thành các bảng dữ liệu quan hệ tương ứng.
* Xác định rõ khóa chính (primary key), khóa ngoại (foreign key) và kiểu dữ liệu của các thuộc tính.

**5. Chuẩn hóa lược đồ quan hệ**

Tiến hành chuẩn hóa cơ sở dữ liệu nhằm đảm bảo cấu trúc bảng dữ liệu đạt chuẩn, tránh hiện tượng dư thừa dữ liệu và các bất thường có thể xảy ra trong quá trình thao tác dữ liệu:

* Xác định các phụ thuộc hàm giữa các thuộc tính.
* Chuẩn hóa dữ liệu tuần tự từ dạng chuẩn 1NF sang dạng 2NF, và đạt đến chuẩn 3NF
* Kiểm tra và đảm bảo mỗi bảng dữ liệu sau chuẩn hóa không còn tồn tại các bất thường như lỗi thêm, xóa hay cập nhật dữ liệu.

**6. Cài đặt cơ sở dữ liệu và nhập dữ liệu minh họa**

Sử dụng ngôn ngữ SQL để hiện thực hóa các bảng dữ liệu đã được thiết kế:

* Viết các câu lệnh SQL tạo bảng ,các khóa chính, khóa ngoại và các ràng buộc dữ liệu.
* Nhập các dữ liệu mẫu đại diện cho từng bảng để minh họa khả năng hoạt động thực tế của hệ thống.

**7. Kiểm thử hệ thống bằng các truy vấn SQL**

Kiểm thử là bước không thể thiếu để đảm bảo rằng cơ sở dữ liệu vận hành ổn định và đáp ứng chính xác các nhu cầu quản lý:

* Viết các truy vấn SQL thực tế như: tìm kiếm thông tin, truy vấn thống kê doanh số, đồ uống bán chạy, quản lý nhân viên và ca làm việc, quản lý nguồn cung nguyên liệu...
* Kiểm tra hiệu suất hoạt động của cơ sở dữ liệu qua các tình huống sử dụng thực tế.

1. **Đối tượng vận hành, các nhiệm vụ trọng yếu và dữ liệu triển khai của hệ thống**

**1. Đối tượng sử dụng hệ thống**

Hệ thống quản lý cơ sở dữ liệu quán cafe được xây dựng nhằm phục vụ và hỗ trợ các đối tượng sử dụng chính sau đây:

* **Chủ quán và nhà quản lý:**  
  Sử dụng hệ thống để theo dõi và giám sát toàn diện hoạt động kinh doanh như quản lý doanh thu, nhân viên, nguồn cung ứng nguyên liệu; từ đó đưa ra quyết định quản trị chính xác và kịp thời.
* **Nhân viên quán (bồi bàn, pha chế , quản lý :**  
  Sử dụng hệ thống để theo dõi ca làm việc, thực hiện nghiệp vụ hàng ngày, và cập nhật thông tin liên quan đến công việc của mình.

**2. Nhiệm vụ chính của hệ thống**

Hệ thống cơ sở dữ liệu quản lý quán cafe được thiết kế nhằm thực hiện những nhiệm vụ quản trị quan trọng sau đây:

* **Quản lý thông tin chung về quán:**Bao gồm thông tin mã quán, tên quán, địa chỉ, cơ sở hoạt động và thông tin liên lạc.
* **Quản lý nhân sự và ca làm việc:**Lưu trữ và theo dõi chi tiết thông tin nhân viên, bao gồm các dữ liệu cá nhân, chức vụ , mức lương và lịch làm việc hàng ngày.
* **Quản lý thông tin khách hàng:**Ghi nhận thông tin cơ bản của khách hàng (họ tên, số điện thoại) và lịch sử sử dụng dịch vụ để phục vụ công tác chăm sóc khách hàng hiệu quả.
* **Quản lý danh mục đồ uống:**Quản lý tập trung các thông tin về đồ uống (mã, tên, giá bán) giúp hỗ trợ việc quản lý tồn kho và định giá bán hợp lý.
* **Quản lý các giao dịch bán hàng và hóa đơn:**Theo dõi, ghi nhận các thông tin giao dịch bán hàng (thời gian, giá trị giao dịch, mã hóa đơn) để quản lý chính xác doanh thu và hỗ trợ bộ phận kế toán trong việc thống kê tài chính.
* **Quản lý nguyên liệu và nhà cung cấp:**Theo dõi nguồn gốc, số lượng, giá cả nguyên liệu cũng như thông tin về các nhà cung cấp để đảm bảo việc cung ứng nguyên liệu diễn ra ổn định và hiệu quả.
* **Truy vấn dữ liệu:**Cung cấp các chức năng truy vấn đa dạng và thống kê số liệu, phục vụ nhu cầu quản trị như đánh giá hiệu suất nhân viên, tình hình bán hàng, phân tích kinh doanh, qua đó nâng cao chất lượng quản lý và hiệu quả kinh doanh.

Hơn thế nữa, Cơ sở dữ liệu này cũng có thể có thể lưu trữ thông tin:

* Lưu trữ hoá đơn
* Lưu trữ nhà cung cấp
* Lưu trữ nguyên liệu được cung cấp
* Lưu trữ những loại đồ uống quán phục vụ
* Lưu trữ về thông tin  nhân viên của quán
* Lưu trữ ca làm của nhân viên

**3. Dữ liệu triển khai trong dự án**

Dữ liệu triển khai trong hệ thống cơ sở dữ liệu quản lý quán cafe được phân thành hai nhóm chính sau:

* **Dữ liệu chính:**Đây là những dữ liệu gần như không thay đổi trong quá trình vận hành, đóng vai trò nền tảng của hệ thống:
  + **Thông tin quán cafe:** mã quán, tên quán, địa chỉ, cơ sở, số điện thoại.
  + **Danh sách đồ uống:** mã đồ uống, tên đồ uống, giá bán.
  + **Danh sách nguyên liệu:** tên nguyên liệu, nguồn gốc xuất xứ, giá nhập.
  + **Danh sách nhà cung cấp:** mã nhà cung cấp, tên nhà cung cấp, địa chỉ, số điện thoại.
* **Dữ liệu phát sinh :**Là những dữ liệu được cập nhật thường xuyên, phản ánh các hoạt động nghiệp vụ hàng ngày của quán:
  + **Thông tin nhân viên:** thông tin cá nhân, chức vụ, mức lương, lịch làm việc.
  + **Thông tin khách hàng:** dữ liệu khách hàng mới và cập nhật lịch sử giao dịch.
  + **Giao dịch bán hàng:** hóa đơn bán hàng, thời gian, giá trị thanh toán.
  + **Chi tiết phục vụ khách hàng:** thông tin về khách hàng gọi đồ uống, đồ uống được gọi, thời gian và địa điểm phục vụ.
  + **Giao dịch nhập nguyên liệu:** thông tin về các lần nhập hàng từ nhà cung cấp, số lượng nhập và giá cả nguyên liệu.

Các dữ liệu này được thiết kế chặt chẽ theo các quy tắc chuẩn hóa nhằm đảm bảo tính nhất quán và toàn vẹn, cung cấp khả năng khai thác dữ liệu hiệu quả nhất cho các đối tượng sử dụng.

1. **Hệ quản trị CSDL SQL Server**

I. Giới thiệu hệ Quản trị CSDL SQL Server và lý do lựa chọn:

Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mạnh mẽ được phát triển bởi Microsoft từ năm 1989. Trải qua hơn ba thập kỷ phát triển với nhiều phiên bản cải tiến, SQL Server đã khẳng định vị thế là một trong những hệ quản trị CSDL hàng đầu thế giới, được ứng dụng rộng rãi trong các tổ chức từ quy mô nhỏ đến doanh nghiệp lớn.

SQL Server được lựa chọn dựa trên những yếu tố sau:

1. **Tích hợp toàn diện với hệ sinh thái Microsoft:** SQL Server hoạt động liền mạch với các công nghệ Microsoft khác như .NET Framework, Windows Server, Azure và Power BI.
2. **Độ tin cậy và ổn định cao:** SQL Server cung cấp các tính năng như AlwaysOn Availability Groups, Database Mirroring và Failover Clustering để đảm bảo tính khả dụng cao.
3. **Hiệu năng vượt trội:** Công nghệ In-Memory OLTP, Columnstore Indexes và Query Store giúp tối ưu hóa hiệu suất cho cả công việc phân tích và giao dịch.
4. **Bảo mật đa lớp:** SQL Server được đánh giá là một trong những hệ quản trị CSDL có ít lỗ hổng bảo mật nhất trong nhiều năm liên tiếp theo báo cáo của NIST.
5. **Khả năng mở rộng:** Hỗ trợ từ các hệ thống nhỏ đến các hệ thống dữ liệu lớn với khả năng xử lý hàng petabyte dữ liệu.
6. **Công cụ quản lý trực quan:** SQL Server Management Studio (SSMS) và các công cụ khác giúp đơn giản hóa việc quản lý và phát triển cơ sở dữ liệu.

II. Kiến trúc của hệ quản trị CSDL SQL Server:

* **Kiến trúc tổng thể:**

SQL Server được xây dựng trên một kiến trúc đa lớp phức tạp nhưng được tổ chức một cách logic, tạo nền tảng vững chắc cho việc quản lý dữ liệu hiệu quả. Kiến trúc tổng thể của SQL Server bao gồm các thành phần chính sau:

1. **Protocol Layer:** Xử lý giao tiếp mạng giữa client và server thông qua các giao thức như TDS (Tabular Data Stream), TCP/IP, Named Pipes.
2. **Relational Engine:** Thành phần cốt lõi của SQL Server, bao gồm:

-  Query Processor: Phân tích cú pháp, tối ưu hóa và thực thi các truy vấn

-  Storage Engine: Quản lý lưu trữ dữ liệu, bộ đệm và giao dịch

- SQLOS: Cung cấp các dịch vụ hệ điều hành như quản lý bộ nhớ, lập lịch và đồng bộ hóa

1. **Database Engine Services:** Bao gồm các dịch vụ như Database Mail, Service Broker, Full-Text Search và Replication.
2. **OLAP Services:** Analysis Services cho phân tích dữ liệu đa chiều.
3. **ETL Services:** Integration Services để trích xuất, chuyển đổi và tải dữ liệu.
4. **Reporting Services:** Tạo và quản lý báo cáo doanh nghiệp.
5. **Machine Learning Services:** Tích hợp R và Python cho phân tích dữ liệu nâng cao.

* **Ưu và nhược điểm:**

SQL Server mang đến nhiều ưu điểm vượt trội so với các hệ quản trị CSDL khác:

1. **Hiệu suất cao:** Công nghệ In-Memory OLTP tăng tốc độ xử lý giao dịch lên đến 30 lần.

2. **Bảo mật toàn diện:** Always Encrypted, Transparent Data Encryption, Row-Level Security.

3. **Tính sẵn sàng:** AlwaysOn Availability Groups, Failover Clustering.

4. **Khả năng mở rộng:** Hỗ trợ sharding, phân vùng bảng và cơ sở dữ liệu phân tán.

5. **Công cụ quản lý hiệu quả:** SSMS, Azure Data Studio, SQL Server Data Tools.

6. **Tích hợp AI và ML:** Phân tích dự đoán ngay trong cơ sở dữ liệu.

Tuy nhiên, SQL Server cũng có một số nhược điểm cần lưu ý. Chi phí bản quyền cao, đặc biệt đối với phiên bản Enterprise, có thể là rào cản đối với một số tổ chức, kết hợp với việc hệ thống yêu cầu tài nguyên lớn về RAM và CPU để đạt hiệu suất tối ưu. Mặc dù hiện nay đã có phiên bản cho Linux, SQL Server vẫn phụ thuộc nhiều vào nền tảng Windows. Việc triển khai đòi hỏi kiến thức chuyên sâu để cấu hình tối ưu, và đôi khi xuất hiện vấn đề tương thích giữa các phiên bản khác nhau.

III. Phương pháp lưu trữ dữ liệu:

* **Cấu trúc lưu trữ vật lý:**

SQL Server sử dụng một hệ thống lưu trữ dữ liệu có cấu trúc phân cấp và hiệu quả. Về mặt vật lý, dữ liệu được lưu trữ trong các tệp với ba loại chính: tệp dữ liệu chính (.mdf) chứa thông tin hệ thống và đối tượng người dùng, tệp dữ liệu thứ cấp (.ndf) lưu trữ dữ liệu bổ sung để phân tán tải I/O, và tệp nhật ký giao dịch (.ldf) ghi lại mọi thay đổi đối với cơ sở dữ liệu, đóng vai trò quan trọng trong khôi phục dữ liệu.

Đơn vị cơ bản nhất trong lưu trữ dữ liệu SQL Server là trang (page) có kích thước 8KB, các trang được nhóm thành các extent 64KB (8 trang liên tiếp). SQL Server sử dụng hai cấu trúc bảng chính: Heap - nơi dữ liệu được lưu trữ không theo thứ tự cụ thể khi bảng không có chỉ mục clustered, và B-Tree - cấu trúc cây cân bằng được sử dụng cho các chỉ mục, đảm bảo truy cập dữ liệu nhanh chóng và hiệu quả.

* **Đánh chỉ mục (Indexing):**

Việc đánh chỉ mục là yếu tố quan trọng quyết định hiệu suất của SQL Server. Chỉ mục Clustered xác định thứ tự vật lý của dữ liệu trong bảng, mỗi bảng chỉ có tối đa một chỉ mục clustered. Khi được triển khai phù hợp, chỉ mục này cải thiện đáng kể hiệu suất tìm kiếm và sắp xếp dữ liệu. Chỉ mục Nonclustered lưu trữ cấu trúc riêng biệt với con trỏ đến dữ liệu thực, cho phép nhiều chỉ mục trên cùng một bảng mà không làm thay đổi cấu trúc vật lý của dữ liệu.

SQL Server cung cấp nhiều loại chỉ mục chuyên biệt đáp ứng các nhu cầu khác nhau. Columnstore Index lưu trữ và nén dữ liệu theo cột thay vì theo hàng, mang lại hiệu suất vượt trội cho các truy vấn phân tích. Filtered Index chỉ đánh chỉ mục trên tập con dữ liệu trong bảng, giúp giảm không gian lưu trữ và cải thiện hiệu suất. Spatial Index tối ưu hóa cho dữ liệu không gian địa lý, trong khi XML Index và Full-Text Index lần lượt hỗ trợ tìm kiếm hiệu quả trên dữ liệu XML và văn bản. Đặc biệt, Memory-Optimized Index với các kiểu Hash và Range index được thiết kế riêng cho bảng In-Memory, tối ưu hóa hiệu suất cho các ứng dụng đòi hỏi thời gian phản hồi cực thấp.

IV. Phương pháp truy vấn và tối ưu hóa

Quá trình xử lý truy vấn trong SQL Server là một chuỗi các bước được thiết kế để đảm bảo hiệu suất và độ chính xác tối đa. Khi một truy vấn được gửi đến hệ thống, nó trải qua năm giai đoạn chính:

1. **Phân tích cú pháp:** Kiểm tra lỗi cú pháp SQL
2. **Ràng buộc:** Xác định các đối tượng được tham chiếu
3. **Tối ưu hóa:** Tạo kế hoạch thực thi hiệu quả nhất
4. **Biên dịch:** Tạo mã thực thi
5. **Thực thi:** Thực hiện truy vấn và trả về kết quả

Phương pháp tối ưu hóa truy vấn:

* **Tối ưu hóa cấu trúc cơ sở dữ liệu**

Chỉ mục (Index):

-  Tạo chỉ mục cho cột trong mệnh đề WHERE, JOIN, ORDER BY

-  Sử dụng chỉ mục không phân cụm (NONCLUSTERED INDEX) cho các cột tìm kiếm thường xuyên

-  Áp dụng chỉ mục lọc (FILTERED INDEX) cho dữ liệu có tính chọn lọc cao

-  Tránh tạo quá nhiều chỉ mục (ảnh hưởng đến hiệu suất thêm/sửa/xóa)

Cấu trúc bảng:

1. Kiểu dữ liệu:

* Sử dụng kiểu dữ liệu nhỏ nhất có thể (INT thay vì BIGINT khi không cần thiết)
* Dùng VARCHAR thay vì NVARCHAR khi không cần Unicode
* Tránh TEXT/NTEXT, thay bằng VARCHAR(MAX)/NVARCHAR(MAX)

1. Phân vùng bảng:

* Phân vùng bảng lớn theo thời gian hoặc phạm vi giá trị
* Giúp quản lý dữ liệu lớn và tăng hiệu suất truy vấn
* **Tối ưu hóa câu lệnh SQL:**
* Tránh SELECT \* - chỉ lấy các cột cần thiết, sử dụng mệnh đề WHERE hợp lý để lọc dữ liệu sớm
* Tránh hàm trên cột: Không dùng hàm trên cột trong mệnh đề WHERE, thay thế bằng điều kiện so sánh trực tiếp Tối ưu truy vấn phức tạp

V. Khôi phục dữ liệu và an toàn bảo mật:

SQL Server cung cấp một hệ thống khôi phục dữ liệu toàn diện thông qua ba mô hình phục hồi: **Simple, Full** và **Bulk-logged**.

- Simple**:** Không sao lưu nhật ký giao dịch, chỉ có thể khôi phục đến thời điểm sao lưu gần nhất

- Full**:** Sao lưu đầy đủ nhật ký giao dịch, khôi phục đến bất kỳ thời điểm nào

- Bulk-logged: Tương tự Full nhưng ghi nhật ký tối thiểu cho các thao tác hàng loạt

SQL Server cũng hỗ trợ nhiều loại sao lưu để đáp ứng nhu cầu đa dạng:

- Sao lưu đầy đủ (Full backup)

- Sao lưu khác biệt (Differential backup)

- Sao lưu nhật ký giao dịch (Transaction log backup)

- Sao lưu từng phần (Partial backup)

- Sao lưu tệp (File backup)

Ngoài ra, SQL Server còn hỗ trợ các tính năng khôi phục tiên tiến như Point-in-Time Recovery cho phép khôi phục đến thời điểm cụ thể trong quá khứ, Page Restore khôi phục chỉ các trang dữ liệu bị hỏng mà không ảnh hưởng đến toàn bộ cơ sở dữ liệu, và Accelerated Database Recovery (ADR) cải thiện đáng kể thời gian khôi phục cho các cơ sở dữ liệu lớn.

Về an toàn bảo mật, SQL Server cung cấp nhiều lớp bảo vệ bắt đầu từ xác thực (Authentication) với ba phương thức: Windows Authentication tích hợp với Active Directory, SQL Server Authentication sử dụng tài khoản và mật khẩu được lưu trữ trong SQL Server, và Azure Active Directory Authentication cho các môi trường đám mây. Sau khi xác thực, hệ thống phân quyền (Authorization) kiểm soát những gì người dùng được phép thực hiện thông qua bảo mật cấp máy chủ và cơ sở dữ liệu, vai trò và quyền, cùng bảo mật cấp đối tượng.

SQL Server áp dụng nhiều phương pháp mã hóa dữ liệu hiện đại như: Transparent Data Encryption (TDE) mã hóa toàn bộ cơ sở dữ liệu tại mức tệp mà không yêu cầu thay đổi ứng dụng; Always Encrypted bảo vệ dữ liệu nhạy cảm ngay cả khi quản trị viên cơ sở dữ liệu truy cập; và Column-level encryption cho phép mã hóa các cột cụ thể. Các tính năng kiểm soát truy cập như Row-Level Security (RLS) hạn chế quyền truy cập dữ liệu ở cấp hàng dựa trên đặc điểm người dùng, Dynamic Data Masking che giấu dữ liệu nhạy cảm, và Data Classification and Auditing giúp phân loại và theo dõi hoạt động truy cập dữ liệu quan trọng.

VI. Lưu ý cài đặt và triển khai hệ quản trị CSDL

Việc cài đặt và triển khai SQL Server đòi hỏi một kế hoạch cẩn thận và có hiểu biết nhất định về yêu cầu của hệ thống cũng như các ứng dụng thực tiễn của dự án. Về phần cứng, SQL Server yêu cầu tối thiểu CPU 2 GHz hoặc nhanh hơn, RAM tối thiểu 4GB (khuyến nghị nên có ít nhất 16GB hoặc cao hơn), và khoảng 6GB không gian đĩa trống cho các cài đặt cơ bản và lưu trữ các cơ sở dữ liệu. Các yêu cầu này có thể cao hơn tùy thuộc vào quy mô và độ phức tạp.

Bên cạnh đó, việc lựa chọn phiên bản SQL Server phù hợp ảnh hưởng rất nhiều đến chi phí và các tính năng cần thiết, vì vậy cần phải quyết định một cách chính xác. Phiên bản Developer cung cấp đầy đủ tính năng như Enterprise nhưng chỉ được phép sử dụng trong môi trường phát triển; Express là phiên bản miễn phí, cơ bản với giới hạn tài nguyên; Standard phục vụ nhu cầu của doanh nghiệp vừa và nhỏ; Enterprise cung cấp đầy đủ tính năng cho các ứng dụng quan trọng; và Azure SQL Database là công cụ quản lý giải pháp đám mây.

Trong quá trình cài đặt và cấu hình, việc phân bổ tệp dữ liệu (.mdf) và nhật ký (.ldf) trên các ổ đĩa riêng biệt để tối ưu hiệu suất I/O là cực kì quan trọng. Cấu hình tempdb với nhiều tệp dữ liệu (thường bằng số lõi CPU, tối đa 8) giúp giảm thiểu tranh chấp và cải thiện hiệu suất hệ thống. Thiết lập các tham số cấu hình phù hợp như giới hạn bộ nhớ, MAXDOP (mức độ song song tối đa), và cost threshold for parallelism sẽ đảm bảo SQL Server hoạt động tối ưu trong những môi trường cụ thể. Bên cạnh đó, việc thiết lập lịch bảo trì định kỳ cho các tác vụ như cập nhật thống kê, xây dựng lại chỉ mục và sao lưu là không thể thiếu.

Bảo mật cũng là một yếu tố cần được chú trọng. Các biện pháp bảo mật ban đầu nên bao gồm việc vô hiệu hóa tài khoản SA hoặc đổi mật khẩu mạnh, giảm thiểu quyền cho người dùng theo nguyên tắc đặc quyền tối thiểu, vô hiệu hóa các dịch vụ không cần thiết để giảm thiểu khả năng tấn công, và đảm bảo cập nhật bản vá bảo mật thường xuyên.

Giám sát hiệu suất là một phần không thể thiếu trong việc duy trì hệ thống SQL Server hoạt động ổn định . Thiết lập SQL Server Agent Jobs để tự động hóa các tác vụ quản trị, cấu hình SQL Server Profiler và Extended Events để theo dõi hoạt động hệ thống, giám sát các chỉ số hiệu suất chính (KPIs) như sử dụng CPU, bộ nhớ và I/O, cũng như thiết lập cảnh báo cho các sự kiện quan trọng giúp phát hiện và xử lý sớm các vấn đề tiềm ẩn.

Ngoài ra, các chiến lược khôi phục toàn diện sau khi bị tấn công là bắt buộc để đảm bảo tính ổn định và bền vững của doanh nghiệp. Điều này bao gồm cấu hình và chuẩn bị sẵn những chiến lược sao lưu phù hợp với yêu cầu RPO (Recovery Point Objective) và RTO (Recovery Time Objective),đồng thời thiết lập môi trường khôi phục sau khi bị tấn công, có thể là một máy chủ dự phòng hoặc giải pháp trên đám mây. Quan trọng hơn cả, chúng ta cần kiểm tra quy trình khôi phục định kỳ để đảm bảo tính hiệu quả khi có sự cố thực tế xảy ra.

1. **Phân tích thiết kế và triển khai hệ CSDL**
2. Ứng dụng của hệ cơ sở dữ liệu:

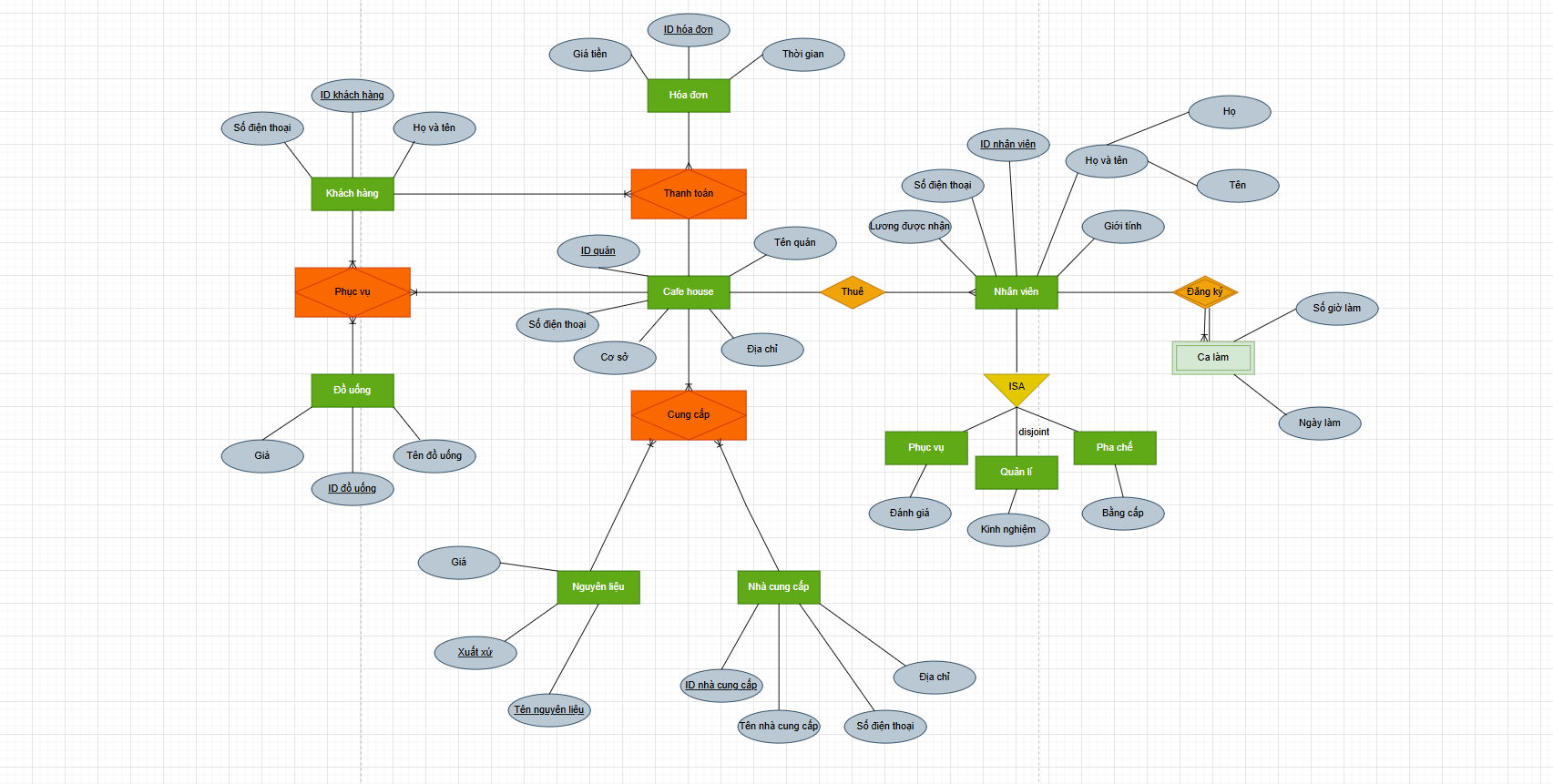
Chuỗi cửa hàng **CafeHouse** có mặt tại nhiều quận trong thành phố, mỗi quán có nhiều nhân viên, và quản lý riêng. Chủ chuỗi muốn một hệ thống giúp:

* Quản lý danh sách khách hàng.
* Quản lý danh sách nhân viên.
* Quản lý hóa đơn bán hàng.
* Quản lý nhà cung cấp và nguyên liệu.
* Quản lý đồ uống.

1. Dữ liệu cần được lưu trữ:
   * Dữ liệu café house.
   * Dữ liệu về hóa đơn bán.
   * Dữ liệu về nhà cung cấp.
   * Dữ liệu về nguyên liệu được cung cấp.
   * Dữ liệu về những loại đồ uống quán phục vụ
   * Dữ liệu về nhân viên của quán.
   * Dữ liệu về ca làm của nhân viên
2. Các thao tác trên cơ sở dữ liệu mà hệ thống có thể thực hiện được:

* Có thể truy cập để xem, xóa, sửa thông tin, thời gian hoạt động của nhân viên, khách hàng
* Có thể truy xuất đến các hóa đơn nhập và hóa đơn bán
* Quản lý doanh số của quán café

# Thiết kế khái niệm



1. Mô tả chi tiết lược đồ:

Có tất cả 14 thực thể trong đó bao gồm 7 tập thực thể mạnh, 1 tập thực thể yếu, 3 tập thực thể con, và 3 thực thể liên kết.

* Thực thể mạnh: Café house, Hóa đơn, Khách hàng, Đồ uống, Nguyên liệu, Nhà cung cấp, Nhân viên.
* Thực thể yếu: Ca làm.
* Thực thể cha: Nhân viên
* Thực thể con: Bồi bàn, Quản lý, Pha chế.
* Thực thể liên kết: Phục vụ. Cung cấp, Thanh toán,

1. Các quy tắc ràng buộc:

* Một quán cà phê có thể thuê nhiều nhân viên, một nhân viên chỉ có thể làm việc cho một quán cà phê
* Một nhân viên có thể đăng kí nhiều ca làm, một ca làm chỉ được đăng kí bởi một nhân viên
* Một số nhân viên có thể không đăng kí ca làm, tất cả các ca làm đều phải được đăng kí
* Nhân viên được chia làm 3 chức vụ, gồm: phục vụ, quản lí và pha chế
* Một quán cà phê có thể phục vụ nhiều đồ uống, một loại đồ uống có thể được phục vụ bởi nhiều quán cà phê
* Một quán cà phê có thể phục vụ nhiều khách hàng, một khách hàng có thể được phục bởi nhiều quán cà phê
* Một khách hàng có thể gọi nhiều đồ uống, một loại đồ uống có thể được gọi bởi nhiều khác hàng
* Một quán cà phê có thể nhập nhiều nguyên liệu và từ nhiều nhà cung cấp
* Một loại nguyên liệu, một nhà cung cấp có thể cung cấp cho nhiều quán cà phê
* Một quán cà phê có thể xuất ra nhiều hóa đơn, một hóa đơn chỉ thuộc về một quán cà phê
* Một khách hàng có thể Thanh toán nhiều hóa đơn
* Một quán cà phê có thể Phục Vụ cho nhiều khách hàng
* Một khách hàng có thể thanh toans nhiều hóa đơn và mỗi hóa đơn có thể sở hữu bởi một khách hàng

# Ánh xạ từ lược đồ E-R sang lược đồ quan hệ

Cafe house

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID quán | Nvarchar | Khóa chính |
| 2 | Tên quán | Nvarchar |  |
| 3 | Số điện thoại | Char |  |
| 4 | Địa chỉ | Nvarchar |  |
| 5 | Cơ sở | Nvarchar |  |

Phục vụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID khách hàng | Nvarchar | Khóa ngoại tham chiếu tới bảng khách hàng |
| 2 | ID đồ uống | Nvarchar | Khóa ngoại tham chiếu tới bảng đồ uống |
| 3 | ID quán | Nvarchar | Khóa ngoại tham chiếu tới bảng café house |

Khách hàng

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID khách hàng | Nvarchar | Khóa chính |
| 2 | Số điện thoại | Char |  |
| 3 | Họ và tên | Nvarchar |  |

Đồ uống

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID đồ uống | Nvarchar | Khóa chính |
| 2 | Tên đồ uống | Nvarchar |  |
| 3 | Giá | Int |  |

Hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID hóa đơn | Nvarchar | Khóa chính |
| 2 | Giá tiền | Int |  |
| 3 | Thời gian | Date |  |

Thanh toán

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID hóa đơn | Nvarchar | Khóa ngoại tham chiếu tới bảng hóa đơn |
| 2 | ID quán | Nvarchar | Khóa ngoại tham chiếu tới bảng café house |
| 3 | ID khách hàng | Nvarchar | Khóa ngoại tham chiếu tới bảng khách hàng |

Nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID nhân viên | Nvarchar | Khóa chính |
| 2 | ID quán | Nvarchar |  |
| 3 | Họ | Nvarchar |  |
| 4 | Tên | Nvarchar |  |
| 5 | Số điện thoại | Char |  |
| 6 | Giới tính | Nvarchar |  |
| 7 | Lương được nhận | Int |  |

Ca làm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | Giờ làm | Nvarchar |  |
| 2 | ID nhân viên | Nvarchar | Khóa ngoại tham chiếu tới bảng nhân viên |

Bồi bàn

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID nhân viên | Nvarchar | Khóa ngoại tham chiếu tới bảng nhân viên |
| 2 | Đánh giá | Nvarchar |  |

Quản lí

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID nhân viên | Nvarchar | Khóa ngoại tham chiếu tới bảng nhân viên |
| 2 | Kinh nghiệm | Nvarchar |  |

Pha chế

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID nhân viên | Nvarchar | Khóa ngoại tham chiếu tới bảng nhân viên |
| 2 | Bằng cấp | Nvarchar |  |

Cung cấp

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID quán | Nvarchar | Khóa ngoại tham chiếu tới bảng café house |
| 2 | ID nhà cung cấp | Nvarchar | Khóa ngoại tham chiếu tới bảng nhà cung cấp |
| 3 | ID nguyên liệu | Nvarchar | Khóa ngoại tham chiếu tới bảng nguyên liệu |
| 4 | Xuất xứ | Nvarchar | Khóa ngoại tham chiếu tới bảng nguyên liệu |

Nguyên liệu

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | Xuất xứ | Nvarchar | Khóa kết hợp |
| 2 | Tên nguyên liệu | Nvarchar | Khóa kết hợp |
| 3 | Số lượng | Int |  |
| 4 | Giá | Int |  |

Nhà cung cấp

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên dữ liệu | Kiểu dữ liệu | Lưu ý |
| 1 | ID nhà cung cấp | Nvarchar | Khóa chính |
| 2 | Tên nhà cung cấp | Nvarchar |  |
| 3 | Địa chỉ | Nvarchar |  |
| 4 | Số điện thoại | Char |  |

# Chuẩn hóa các lược đồ quan hệ về dạng 3NF

1. Bảng quan hệ KHACHHANG:

Q1 = (IDKhachHang, SoDienThoai, HoVaTen)

- Đặt IDKhachHang = A, SoDienThoai = B, HoVaTen = C

=> Q1 = (ABC)

Phụ thuộc hàm:

F1 = {A ‎‎→ B, A → C}

- Xác định khóa của Q1, ta có:

(A)+ = {A, B, C} = Q1 nên A là khóa của quan hệ Q1

Thuộc tính không khóa (B, C) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q1 đạt chuẩn 1NF

Thuộc tính không khóa (B, C) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q1 đạt chuẩn 2NF

Thuộc tính không khóa (B, C) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q1 đạt chuẩn 3NF

2. Bảng quan hệ DOUONG:

Q2 = (IDDoUong, TenDoUong, Gia)

- Đặt IDDoUong = A, TenDoUong = B, Gia = C

=> Q2 = (ABC)

Phụ thuộc hàm:

F2 = {A → B, A → C}

Xác định khóa của Q2, ta có:

(A)+ = {A, B} = Q2 nên A là khóa của quan hệ Q2

Thuộc tính không khóa (B) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q2 đạt chuẩn 1NF

Thuộc tính không khóa (B) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q2 đạt chuẩn 2NF

Thuộc tính không khóa (B) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q2 đạt chuẩn 3NF

3. Bảng quan hệ PHUCVU:

Q3 = (IDKhachHang, IDDoUong, IDQuan)

- Đặt IDKhachHang = A, IDDoUong = B, IDQuan = C

=> Q3 = (ABC)

Phụ thuộc hàm:

F3 = ∅

Xác định khóa của Q3, ta có

(ABC)+ = {A, B, C} = Q3 nên ABC là khóa của quan hệ Q3

Ta thấy, quan hệ Q3 không chứa thuộc tính không khóa và không có sự phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q3, đạt chuẩn 3NF

4. Bảng quan hệ CAFEHOUSE:

Q4 = (IDQuan, TenQuan, SoDienThoai, DiaChi, CoSo)

- Đặt IDQuan = A, TenQuan = B, SoDienThoai = C, DiaChi = D, CoSo = E)

=> Q4 = (ABCDE)

Phụ thuộc hàm:

F4 = {A → B, A → C, A → D, A → E}

Xác định khóa của Q4, ta có:

(A)+ = {A, B, C, D, E} = Q4 nên A là khóa của quan hệ Q4

Thuộc tính không khóa (B, C, D, E) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q4 đạt chuẩn 1NF

Thuộc tính không khóa (B, C, D, E) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q4 đạt chuẩn 2NF

Thuộc tính không khóa (B, C, D, E) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q4 đạt chuẩn 3NF

5. Bảng quan hệ NHANVIEN:

Q5 = (IDNhanVien, IDQuan, Ho, Ten, SoDienThoai, GioiTinh, LuongDuocNhan)

- Đặt IDNhanVien = A, IDQuan = B, Ho = C, Ten = D SoDienThoai = E, GioiTinh = F, LuongDuocNhan = G

=> Q5 = (ABCDEFG)

Phụ thuộc hàm:

F5 = {AB → C, AB → D, AB →E, AB →F, AB →G}

Xác định khóa của Q5, ta có:

(AB)+ = {A, B, C, D, E, F, G} = Q5 nên AB là khóa của quan hệ Q5

Thuộc tính không khóa (C, D, E, F, G) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q5 đạt chuẩn 1NF

Thuộc tính không khóa (C, D, E, F, G) phụ thuộc hàm đầy đủ vào khóa AB

=> Quan hệ Q5 đạt chuẩn 2NF

Thuộc tính không khóa (C, D, E, F, G) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q5 đạt chuẩn 3NF

6. Bảng quan hệ CALAM:

Q6 = (SoGioLam, IDNhanVien)

- Đặt SoGioLam = A, IDNhanVien = B

=> Q6 = (AB)

Phụ thuộc hàm:

F6 = {A →B}

Xác định khóa của Q6, ta có:

(A)+ = {A, B} = Q6 nên A là khóa của quan hệ Q6

Thuộc tính không khóa (B) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q6 đạt chuẩn 1NF

IDNhanVien là một khóa dự bị và thuộc tính không khóa (B) phụ thuộc hàm đầy đủ vào IDNhanVien

=> Quan hệ Q6 đạt chuẩn 2NF

Thuộc tính không khóa (B) không có phụ thuộc hàm bắc cầu vào siêu khóa

=> Quan hệ Q6 đạt chuẩn 3NF

7. Bảng quan hệ BOIBAN:

Q7 = (IDNhanVien, DanhGia)

- Đặt IDNhanVien = A, DanhGia = B

=> Q7 = (AB)

Phụ thuộc hàm:

F7 = {A →B}

Xác định khóa của Q7, ta có:

(A)+ = {A, B} = Q7 nên A là khóa của quan hệ Q7

Thuộc tính không khóa (B) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q7 đạt chuẩn 1NF

Thuộc tính không khóa (B) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q7 đạt chuẩn 2NF

Thuộc tính không khóa (B) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q7 đạt chuẩn 3NF

8. Bảng quan hệ QUANLI:

Q8 = (IDNhanVien, KinhNghiem)

- Đặt IDNhanVien = A, KinhNghiem = B

=> Q8 = (AB)

Phụ thuộc hàm:

F8 = {A →B}

Xác định khóa của Q8, ta có:

(A)+ = {A, B} = Q8 nên A là khóa của quan hệ Q8

Thuộc tính không khóa (B) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q8 đạt chuẩn 1NF

Thuộc tính không khóa (B) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q8 đạt chuẩn 2NF

Thuộc tính không khóa (B) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q8 đạt chuẩn 3NF

9. Bảng quan hệ PHACHE:

Q9 = (IDNhanVien, BangCap)

- Đặt IDNhanVien = A, BangCap = B

=> Q9 = (AB)

Phụ thuộc hàm:

F9 = {A →B}

Xác định khóa của Q9, ta có:

(A)+ = {A, B} = Q9 nên A là khóa của quan hệ Q9

Thuộc tính không khóa (B) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q8 đạt chuẩn 1NF

Thuộc tính không khóa (B) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q8 đạt chuẩn 2NF

Thuộc tính không khóa (B) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q8 đạt chuẩn 3NF

10. Bảng quan hệ HOADON:

Q10 = (IDHoaDon, GiaTien, ThoiGian)

- Đặt IDHoaDon = A, GiaTien = B, ThoiGian = C)

=> Q10 = (ABC)

Phụ thuộc hàm:

F10 = {A →B, A →C}

Xác định khóa của Q10, ta có:

(A)+ = {A, B, C} = Q10 nên A là khóa của quan hệ Q10

Thuộc tính không khóa (B, C) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q10 đạt chuẩn 1NF

Thuộc tính không khóa (B, C) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q10 đạt chuẩn 2NF

Thuộc tính không khóa (B, C) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q10 đạt chuẩn 3NF

11. Bảng quan hệ THANHTOAN:

Q11 = (IDHoaDon, IDQuan, IDKhachHang)

- Đặt IDhoaDon = A, IDQuan = B, IDKhachHang = C

=> Q11 = (ABC)

Phụ thuộc hàm:

F11 = ∅

Xác định khóa của Q11, ta có

(ABC)+ = {A, B, C} = Q11 nên ABC là khóa của quan hệ Q11

Ta thấy, quan hệ Q11 không chứa thuộc tính không khóa và không có sự phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q11, đạt chuẩn 3NF

12. Bảng quan hệ CUNGCAP:

Q12 = (IDQuan, IDNhaCungCap, TenNguyenLieu, XuatXu)

- Đặt IDQuan = A, IDNhaCungCap = B, TenNguyenLieu = C, XuatXu = D

=> Q12 = (ABCD)

Phụ thuộc hàm:

F12 = ∅

Xác định khóa của Q12, ta có

(ABC)+ = {A, B, C} = Q12 nên ABC là khóa của quan hệ Q12

Ta thấy, quan hệ Q12 không chứa thuộc tính không khóa và không có sự phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q12, đạt chuẩn 3NF

13. Bảng quan hệ NHACUNGCAP:

Q13 = (IDNhaCungCap, TenNhaCungCap, DiaChi, SoDienThoai)

- Đặt IDNhaCungCap = A, TenNhaCungCap = B, DiaChi = C, SoDienThoai = D

=> Q13 = (ABCD)

Phụ thuộc hàm:

F13 = {A →B, A →C, A →D}

Xác định khóa của Q13, ta có

(ABC)+ = {A, B, C, D} = Q13 nên A là khóa của quan hệ Q13

Thuộc tính không khóa (B, C, D) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

=> Quan hệ Q13 đạt chuẩn 1NF

Thuộc tính không khóa (B, C, D) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q13 đạt chuẩn 2NF

Thuộc tính không khóa (B, C, D) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q13 đạt chuẩn 3NF

14. Bảng quan hệ NGUYENLIEU:

Q14 = (XuatXu, TenNguyenLieu, Gia)

- Đặt XuatXu = A, TenNguyenLieu = B, Gia = C

=> Q14 = (ABC)

Phụ thuộc hàm:

F14 = {AB →C}

Xác định khóa của Q14, ta có:

(AB)+ = {A, B, C} = Q14 nên AB là khóa của quan hệ Q14

Thuộc tính không khóa (C) không có thuộc tính đa trị và không có thuộc tính dẫn xuất

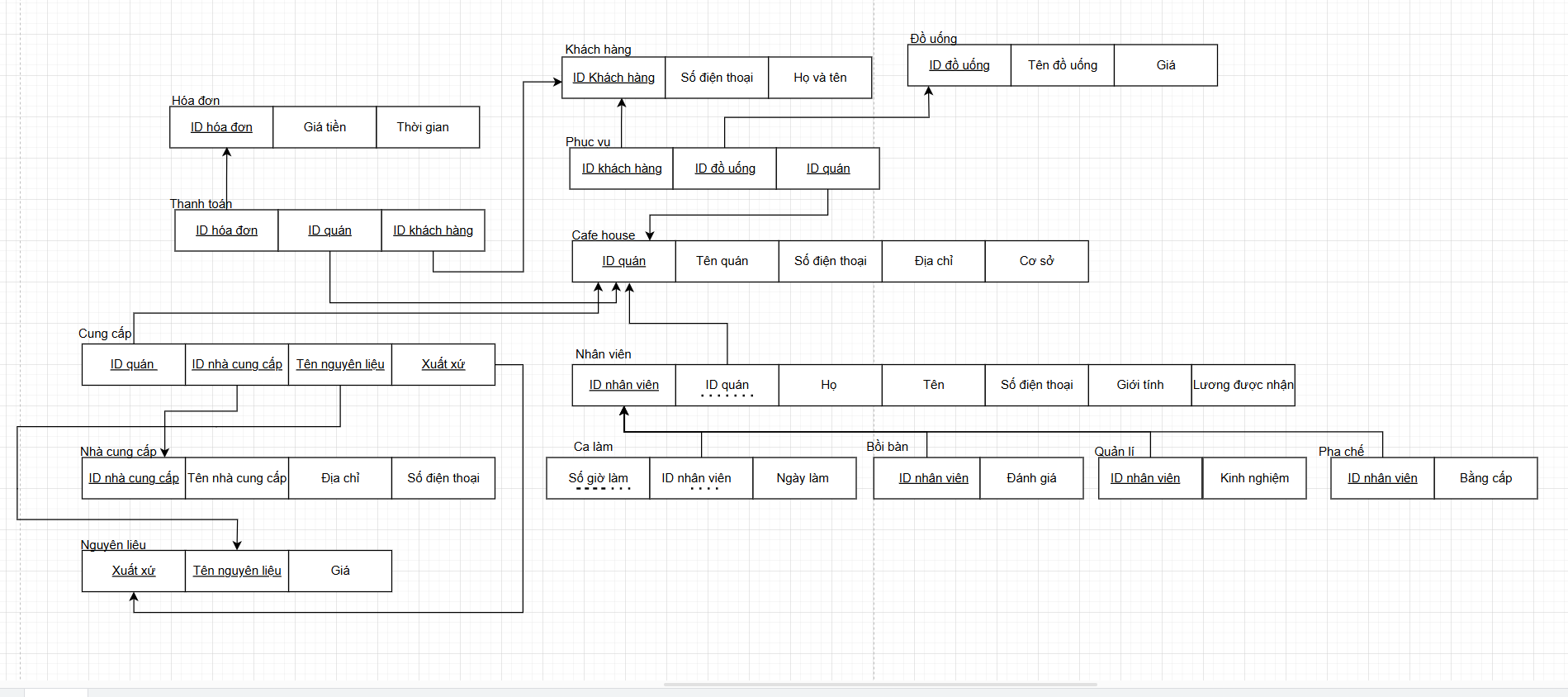
=> Quan hệ Q14 đạt chuẩn 1NF

Thuộc tính không khóa (C) phụ thuộc hàm đầy đủ vào khóa A

=> Quan hệ Q14 đạt chuẩn 2NF

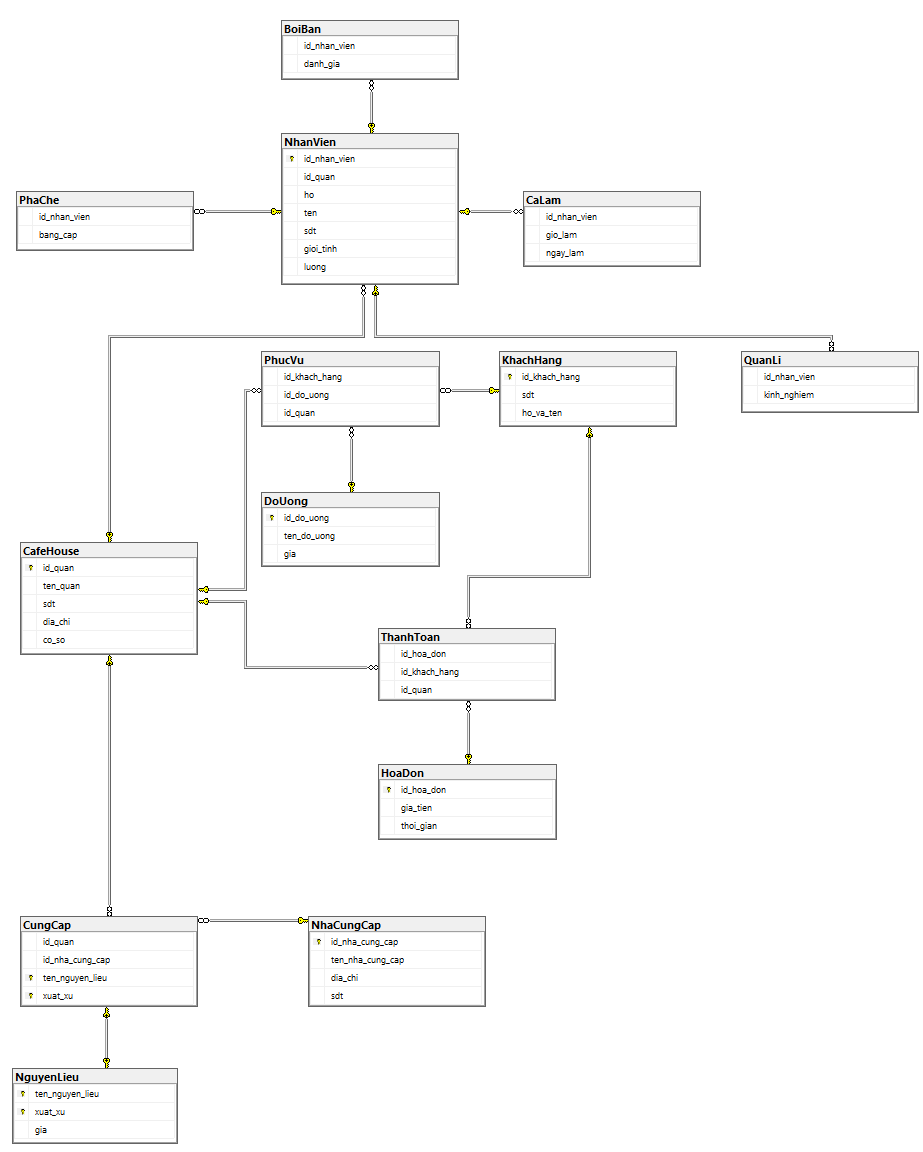
Thuộc tính không khóa (C) không có phụ thuộc hàm bắc cầu vào khóa

=> Quan hệ Q14 đạt chuẩn 3NF



# Cài đặt

* Từ 14 lược đồ ở dạng chuẩn 3NF ta xây dựng được 14 bảng dữ liệu.



* Tạo bảng CafeHouse

create table CafeHouse(

id\_quan nvarchar(10) primary key,

ten\_quan nvarchar(30),

sdt nvarchar(30),

dia\_chi nvarchar(30),

co\_so int

);

insert into CafeHouse(id\_quan, ten\_quan, sdt, dia\_chi, co\_so) values ('CF01', 'Nha go', '0988831458', 'Ho Van Quan', 1);

insert into CafeHouse(id\_quan, ten\_quan, sdt, dia\_chi, co\_so) values ('CF02', 'Nha go', '0988821341', 'CauGiay', 2);



* Tạo bảng NhanVien

create table NhanVien(

id\_nhan\_vien nvarchar(30) primary key,

id\_quan nvarchar(10),

ho nvarchar(30),

ten nvarchar(30),

sdt text,

gioi\_tinh nvarchar(30),

luong int,

foreign key (id\_quan) references CafeHouse(id\_quan)

);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV01','CF01','Nguyen','Duc Anh', '0932641235','nam',10000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV02','CF02','Nguyen',' Ba Tuan', '0195306428','nam',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV03','CF01','Ho','Anh Dung', '0956682782','nam',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV04','CF02','Luu','Trong Hieu', '0503946962','nam',2000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV05','CF01','Le','Hai Dang', '0493661974','nu',6000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV06','CF02','Bui','Kieu Diem', '0835588398','nu',8000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV07','CF01','Tran','Kim Bien', '0842438743','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV08','CF02','Ngo','Kim Quynh', '0906735660','nu',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV09','CF01','Tran','Thi Phuong', '0372569017','nu',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV10','CF02','Trinh','Nhu Hoa', '0929893151','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV11','CF01','Tran','Trong Hieu', '0842438741','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV12','CF02','Ngo','Hai Dang', '0906735760','nam',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV13','CF01','Tran','Thi Phuong Anh', '0372599017','nu',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV14','CF02','Trinh','Quang Bach', '0929853151','nam',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV15','CF02','Ngo','Chi Thanh', '0929853151','nam',4000000);



* Tạo bảng BoiBan

create table BoiBan(

id\_nhan\_vien nvarchar(30),

danh\_gia nvarchar(30),

foreign key (id\_nhan\_vien) references NhanVien(id\_nhan\_vien)

);

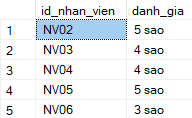
insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV02', '5 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV03', '4 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV04', '4 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV05', '5 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV06', '3 sao');



* Tạo bảng QuanLi

create table QuanLi(

id\_nhan\_vien nvarchar(30),

kinh\_nghiem nvarchar(30),

foreign key (id\_nhan\_vien) references NhanVien(id\_nhan\_vien)

);

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV01', '10 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV11', '8 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV12', '6 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV13', '4 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV14', '2 nam');



* Tạo bảng PhaChe

create table PhaChe(

id\_nhan\_vien nvarchar(30),

bang\_cap nvarchar(30),

foreign key (id\_nhan\_vien) references NhanVien(id\_nhan\_vien)

);

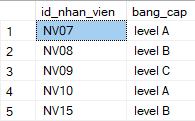
insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV07', 'level A');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV08', 'level B');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV09', 'level C');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV10', 'level A');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV15', 'level B');



* Tạo bảng CaLam

create table CaLam(

id\_nhan\_vien nvarchar(30),

gio\_lam int,

ngay\_lam date,

foreign key (id\_nhan\_vien) references NhanVien(id\_nhan\_vien)

);

select \* from calam;

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV01', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV02', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV03', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV04', 8, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV05', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV06', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV07', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV08', 4, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV09', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV10', 4, '2024-6-11');

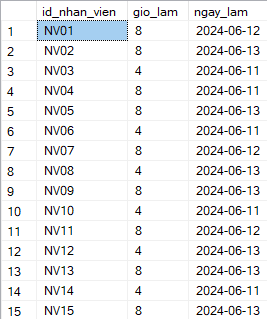
insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV11', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV12', 4, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV13', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV14', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV15', 8, '2024-6-13');



* Tạo bảng KhachHang

create table KhachHang(

id\_khach\_hang nvarchar(30) primary key,

sdt text,

ho\_va\_ten nvarchar(50)

);

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH01','0725757239', 'Nguyen Le Thu Ha');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH02','0957572397', 'Vu Quang Anh' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH03','0725574239', 'Ho Nhat Minh');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH04','0815832393', 'Ba Van Kim' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH05','0725183239', 'Truong Thu Phuong');

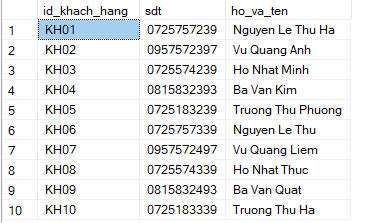
insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH06','0725757339', 'Nguyen Le Thu');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH07','0957572497', 'Vu Quang Liem' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH08','0725574339', 'Ho Nhat Thuc');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH09','0815832493', 'Ba Van Quat' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH10','0725183339', 'Truong Thu Ha');



* Tạo bảng DoUong

create table DoUong(

id\_do\_uong nvarchar(30) primary key,

ten\_do\_uong nvarchar(30),

gia int

);

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU01','cafe den', '30000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU02','cafe nau', '35000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU03','tra quyt', '40000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU04','tra vai', '45000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU05','cacao', '50000');

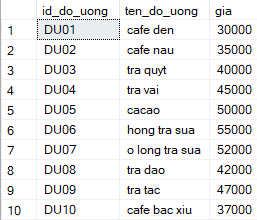
insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU06','hong tra sua', '55000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU07','o long tra sua', '52000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU08','tra dao', '42000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU09','tra tac', '47000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU10','cafe bac xiu', '37000');



* Tạo bảng PhucVu

create table PhucVu(

id\_khach\_hang nvarchar(30),

id\_do\_uong nvarchar(30),

id\_quan nvarchar(10)

foreign key (id\_khach\_hang) references KhachHang(id\_khach\_hang),

foreign key (id\_do\_uong) references DoUong(id\_do\_uong),

foreign key (id\_quan) references CafeHouse(id\_quan)

);

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH01','DU01','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU01','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU02','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU05','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH03','DU06','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH03','DU07','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH04','DU03','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH04','DU10','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU08','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU09','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU04','CF02');

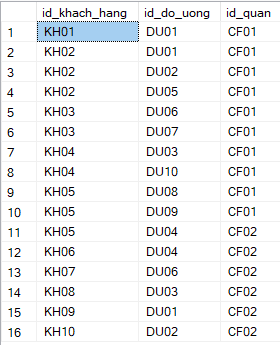
insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH06','DU04','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH07','DU06','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH08','DU03','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH09','DU01','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH10','DU02','CF02');



* Tạo bảng HoaDon:

create table HoaDon(

id\_hoa\_don nvarchar(30) primary key,

gia\_tien text,

thoi\_gian date,

);

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD01', '30000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD02', '115000', '2024-6-12');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD03', '107000', '2024-6-13');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD04', '77000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD05', '134000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD06', '45000', '2024-6-21');

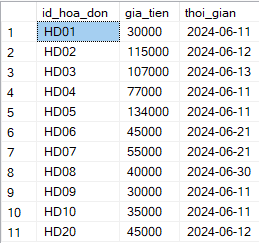
insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD07', '55000', '2024-6-21');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD08', '40000', '2024-6-30');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD09', '30000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD10', '35000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD20', '45000', '2024-6-12');



* Tạo bảng ThanhToan

create table ThanhToan(

id\_hoa\_don nvarchar(30),

id\_khach\_hang nvarchar(30),

id\_quan nvarchar(10),

foreign key (id\_quan) references CafeHouse(id\_quan),

foreign key (id\_hoa\_don) references HoaDon(id\_hoa\_don),

foreign key (id\_khach\_hang) references KhachHang(id\_khach\_hang)

);

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD01', 'KH01', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD02', 'KH02', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD03', 'KH03', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD04', 'KH04', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD05', 'KH05', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD06', 'KH06', 'CF02');

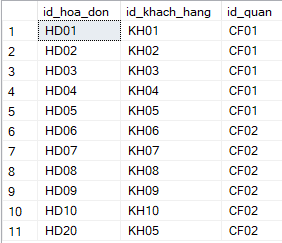
insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD07', 'KH07', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD08', 'KH08', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD09', 'KH09', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD10', 'KH10', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD20', 'KH05', 'CF02');



* Tạo bảng NhaCungCap:

create table NhaCungCap(

id\_nha\_cung\_cap nvarchar(30) primary key,

ten\_nha\_cung\_cap nvarchar(30),

dia\_chi nvarchar(30),

sdt text

);

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC01', 'Phong Ho Supplier', 'Ha Noi', '0819219785');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC02', 'Khoa Van Supplier', 'Hai Phong', '0828008645');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC03', 'Cong La Supplier', 'Nghe An', '0523575928');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC04', 'Duy Tran Company', 'Ha Tinh', '0304066831');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC05', 'Duy La Company', 'Ha Noi', '0773436692');



* Tạo bảng NguyenLieu:

create table NguyenLieu(

ten\_nguyen\_lieu nvarchar(30),

xuat\_xu nvarchar(30),

gia int,

primary key( xuat\_xu, ten\_nguyen\_lieu)

);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hong tra','Lang Son',1500000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hop vai','Bac Giang',2000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hop dao','Hai Phong',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cafe arabica','Buon Ma Thuat',4000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cafe robusta','Buon Ma Thuat',6000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cacao','Buon Ma Thuat',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('tra o long','Thai Nguyen',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('qua chanh','Ha Noi',200000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('nuoc duong','Ha Noi',300000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('ong hut','Ha Noi',150000);



* Tạo bảng CungCap:

CREATE TABLE CungCap (

id\_quan NVARCHAR(10),

id\_nha\_cung\_cap NVARCHAR(30),

ten\_nguyen\_lieu NVARCHAR(30),

xuat\_xu NVARCHAR(30),

PRIMARY KEY (xuat\_xu, ten\_nguyen\_lieu),

FOREIGN KEY (id\_quan) REFERENCES CafeHouse(id\_quan),

FOREIGN KEY (id\_nha\_cung\_cap) REFERENCES NhaCungCap(id\_nha\_cung\_cap),

FOREIGN KEY (xuat\_xu, ten\_nguyen\_lieu) REFERENCES NguyenLieu(xuat\_xu, ten\_nguyen\_lieu)

);

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC01','hong tra', 'Lang Son');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC01','hop vai', 'Bac Giang');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC02','hop dao', 'Hai Phong');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC03','bot cafe arabica', 'Buon Ma Thuat');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC03','bot cafe robusta', 'Buon Ma Thuat');

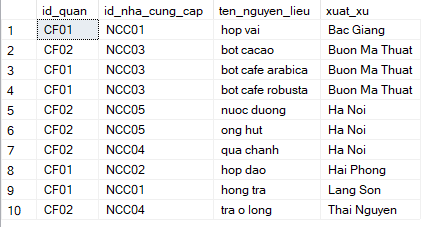
insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC03','bot cacao', 'Buon Ma Thuat');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC04','tra o long', 'Thai Nguyen');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC04','qua chanh', 'Ha Noi');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC05','nuoc duong', 'Ha Noi');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC05','ong hut', 'Ha Noi');



# Truy vấn

Câu 1: Lấy thông tin về tất cả nhân viên có giới tính nam.

Đại số quan hệ:

σ (gioi\_tinh = 'nam') (NhanVien)

SELECT \* FROM NhanVien

WHERE gioi\_tinh LIKE 'nam';



Câu 2: Lấy thông tin về tên của các nhân viên bồi bàn theo thứ tự từ đánh giá giảm dần.

Đại số quan hệ:

π (ho, ten, danh\_gia) (NhanVien ⨝ NhanVien.id\_nhan\_vien = BoiBan.id\_nhan\_vien BoiBan)

Select N.ho, N.ten, B.danh\_gia

From NhanVien N join BoiBan B on N.id\_nhan\_vien = B.id\_nhan\_vien

ORDER by danh\_gia DESC



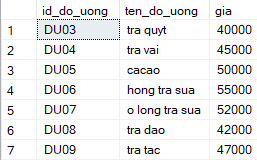
Câu 3: Liệt kê thông tin các đồ uống có giá từ 40 nghìn trờ lên.

Đại số quan hệ:

σ (gia ≥ 40000) (DoUong)

SELECT \* FROM DoUong

WHERE CAST(gia AS INTEGER) >= 40000;



Câu 4: Lấy danh sách các quán có nhiều hơn 7 nhân viên.

Đại số quan hệ:

σ (SoLuongNhanVien > 7) (π (id\_quan, ten\_quan, sdt, dia\_chi, co\_so, COUNT(id\_nhan\_vien)→SoLuongNhanVien) (CafeHouse ⨝ CafeHouse.id\_quan = NhanVien.id\_quan NhanVien))

SELECT C.\*, COUNT(N.id\_quan) AS SoLuongNhanVien

FROM CafeHouse C

JOIN NhanVien N ON C.id\_quan = N.id\_quan

GROUP BY C.id\_quan, C.ten\_quan, C.sdt, C.dia\_chi, C.co\_so

HAVING COUNT(N.id\_quan) > 7;



Câu 5: Liệt kê các quán cafe và phân loại theo số lượng nhân viên của từng quán (ít hơn 5 là 'Nhỏ', từ 5 đến 10 là 'Trung bình', trên 10 là 'Lớn').

Đại số quan hệ:

π (ten\_quan, so\_luong\_nhan\_vien, phan\_loai) ( ρ (ten\_quan, COUNT(id\_nhan\_vien)→so\_luong\_nhan\_vien, CASE khi so\_luong\_nhan\_vien < 5 thì 'Nhỏ' khi so\_luong\_nhan\_vien ≥ 5 và ≤ 10 thì 'Trung bình' ngược lại 'Lớn' END → phan\_loai ) (CafeHouse ⟕ CafeHouse.id\_quan = NhanVien.id\_quan NhanVien) )

SELECT CH.ten\_quan,

COUNT(NV.id\_nhan\_vien) AS so\_luong\_nhan\_vien,

CASE

WHEN COUNT(NV.id\_nhan\_vien) < 5 THEN 'Nhỏ'

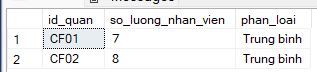
WHEN COUNT(NV.id\_nhan\_vien) BETWEEN 5 AND 10 THEN 'Trung bình'

ELSE 'Lớn'

END AS phan\_loai FROM CafeHouse CH

LEFT JOIN NhanVien NV ON CH.id\_quan = NV.id\_quan

GROUP BY CH.id\_quan;



Câu 6: Liệt kê tên các loại đồ uống sắp xếp theo số lần được gọi giảm dần.

Đại số quan hệ:

π\_{ten\_do\_uong, COUNT(id\_do\_uong) → so\_lan\_goi}

  (γ\_{ten\_do\_uong, COUNT(id\_do\_uong)}

    (σ\_{DoUong.id\_do\_uong = PhucVu.id\_do\_uong} (DoUong ⨝ PhucVu)))

SELECT DoUong.ten\_do\_uong, COUNT(PhucVu.id\_do\_uong) AS so\_lan\_goi

FROM PhucVu, DoUong

WHERE DoUong.id\_do\_uong LIKE PhucVu.id\_do\_uong

GROUP BY DoUong.ten\_do\_uong

ORDER BY COUNT(PhucVu.id\_do\_uong) DESC;



Câu 7: Xác định số lượng nhân viên có mức lương lớn hơn mức trung bình tại từng quán cafe.

Đại số quan hệ:

LET avg\_luong := γ\_{id\_quan, AVG(luong) → avg\_luong}(NhanVien)

π\_{id\_quan, COUNT(id\_nhan\_vien) → so\_nhan\_vien\_tren\_tb}

(

  γ\_{id\_quan, COUNT(id\_nhan\_vien)}

  (

    σ\_{NV.luong > AL.avg\_luong}

    (

      ((CafeHouse ⨝\_{CafeHouse.id\_quan = NhanVien.id\_quan} NhanVien)

       ⨝\_{NhanVien.id\_quan = avg\_luong.id\_quan} avg\_luong)

    )

  )

)

WITH avg\_luong AS (

SELECT id\_quan, AVG(luong) AS avg\_luong

FROM NhanVien

GROUP BY id\_quan

)

SELECT CH.id\_quan, COUNT(NV.id\_nhan\_vien) AS so\_nhan\_vien\_tren\_tb

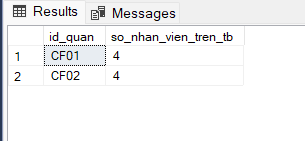
FROM CafeHouse CH

JOIN NhanVien NV ON CH.id\_quan = NV.id\_quan

JOIN avg\_luong AL ON NV.id\_quan = AL.id\_quan

WHERE NV.luong > AL.avg\_luong

GROUP BY CH.id\_quan;



Câu 8: Lấy danh sách nhân viên và tên quán nơi họ làm việc và có mức lương trên 5 triệu.

Đại số quan hệ:

π\_{ho, ten, luong, ten\_quan, co\_so}

(

  σ\_{luong > 5000000}

  (

    NhanVien ⨝\_{NhanVien.id\_quan = CafeHouse.id\_quan} CafeHouse

  )

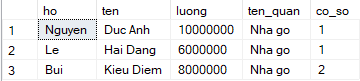
)

SELECT NhanVien.ho, NhanVien.ten, NhanVien.luong, CafeHouse.ten\_quan, CafeHouse.co\_so

FROM NhanVien

INNER JOIN CafeHouse ON NhanVien.id\_quan = CafeHouse.id\_quan

WHERE NhanVien.luong > 5000000;



Câu 9: Liệt kê số đồ uống được gọi theo tên khách hang ở café 1.

Đại số quan hệ:

π\_{id\_khach\_hang, ho\_va\_ten, COUNT(id\_do\_uong) → tong\_so\_do\_uong}

(

  γ\_{id\_khach\_hang, ho\_va\_ten, COUNT(id\_do\_uong)}

  (

    σ\_{id\_quan = 'CF01'}

    (

      (KhachHang ⨝\_{KhachHang.id\_khach\_hang = PhucVu.id\_khach\_hang} PhucVu)

      ⨝\_{PhucVu.id\_do\_uong = DoUong.id\_do\_uong} DoUong

    )

  )

)

SELECT KH.id\_khach\_hang, KH.ho\_va\_ten, COUNT(DU.id\_do\_uong) AS tong\_so\_do\_uong

FROM KhachHang KH

JOIN PhucVu PV ON KH.id\_khach\_hang = PV.id\_khach\_hang

JOIN DoUong DU ON PV.id\_do\_uong = DU.id\_do\_uong

WHERE PV.id\_quan = 'CF01'

GROUP BY KH.id\_khach\_hang, KH.ho\_va\_ten;



Câu 10: Liệt kê các nguyên liệu mà các quán nhập từ các nhà cung cấp tại Hà Nội , liệt kê cả id và tên các quán đó

Đại số quan hệ:

π\_{ten\_nguyen\_lieu, xuat\_xu, ten\_nha\_cung\_cap, dia\_chi, id\_quan, ten\_quan}

(

  σ\_{NhaCungCap.dia\_chi = 'Ha Noi'}

  (

    (

      (

        (CungCap ⨝\_{CungCap.id\_nha\_cung\_cap = NhaCungCap.id\_nha\_cung\_cap} NhaCungCap)

        ⨝\_{CungCap.id\_quan = CafeHouse.id\_quan} CafeHouse

      )

      ⨝\_{CungCap.ten\_nguyen\_lieu = NguyenLieu.ten\_nguyen\_lieu ∧ CungCap.xuat\_xu = NguyenLieu.xuat\_xu} NguyenLieu

    )

  )

SELECT CungCap.ten\_nguyen\_lieu, CungCap.xuat\_xu, NhaCungCap.ten\_nha\_cung\_cap, NhaCungCap.dia\_chi, CafeHouse.id\_quan, CafeHouse.ten\_quan

FROM CungCap

JOIN NhaCungCap ON NhaCungCap.id\_nha\_cung\_cap = CungCap.id\_nha\_cung\_cap

JOIN CafeHouse ON CungCap.id\_quan = CafeHouse.id\_quan

JOIN NguyenLieu ON (CungCap.ten\_nguyen\_lieu = NguyenLieu.ten\_nguyen\_lieu AND CungCap.xuat\_xu = NguyenLieu.xuat\_xu)

WHERE NhaCungCap.dia\_chi = 'Ha Noi';

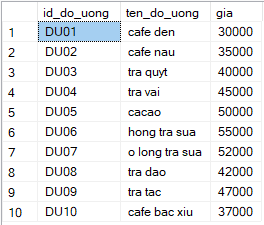


Câu 11: Liệt kê toàn bộ đồ uống trong quán

Đại số quan hệ:

π\_{\*}(DoUong)

SELECT \* FROM DoUong;



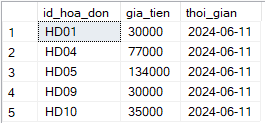
Câu 12: Xem các hóa đơn trong khoảng thời gian

Đại số quan hệ:

σ\_{thoi\_gian ≥ '2024-06-09' ∧ thoi\_gian ≤ '2024-06-11'}(HoaDon)

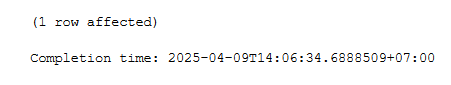
SELECT \* FROM HoaDon

WHERE thoi\_gian BETWEEN '2024-06-09' AND '2024-06-11';



Câu 13: Thêm 1 đồ uống mới

INSERT INTO DoUong(id\_do\_uong, ten\_do\_uong, gia) VALUES ('DU11', 'nuoc dua', '20000');

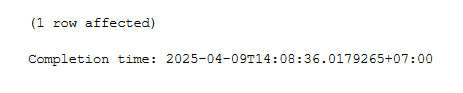


Câu 14: Cập nhật số điện thoại của nhà cung cấp

UPDATE NhaCungCap

SET sdt = '0988777666'

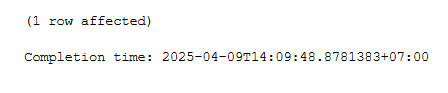
WHERE id\_nha\_cung\_cap = 'NCC05';



Câu 15: Xóa 1 đồ uống

DELETE From DoUong

WHERE id\_do\_uong = ‘DU11’;



Câu 16: Xem đánh giá của các nhân viên bồi bàn

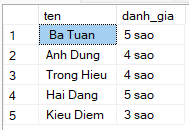
Đại số quan hệ:

πten,danh\_gia​(NhanVien⋈NhanVien.id\_nhan\_vien=BoiBan.id\_nhan\_vien​BoiBan)

SELECT nv.ten, bb.danh\_gia

FROM NhanVien nv

JOIN BoiBan bb ON nv.id\_nhan\_vien = bb.id\_nhan\_vien;

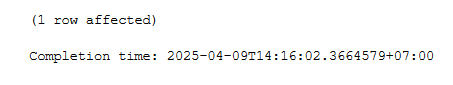


Câu 17: Cập nhật lương cho nhân viên

UPDATE NhanVien

SET luong = 7000000

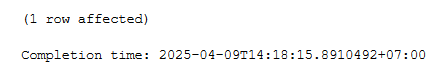
WHERE id\_nhan\_vien = 'NV03';



Câu 18: Xóa nguyên liệu

DELETE FROM NguyenLieu

WHERE ten\_nguyen\_lieu = 'coc giay';



Câu 19: Xem chi tiết thanh toán của khách hang

Đại số quan hệ:

πtt.id\_hoa\_don,tt.id\_khach\_hang,tt.id\_quan,hd.gia\_tien​(σtt.id\_khach\_hang=′KH02′​(ThanhToan⋈ThanhToan.id\_hoa\_don=HoaDon.id\_hoa\_don​HoaDon))

SELECT tt.\*, hd.gia\_tien

FROM ThanhToan tt

JOIN HoaDon hd ON tt.id\_hoa\_don = hd.id\_hoa\_don

WHERE tt.id\_khach\_hang = 'KH02';



Câu 20: Xem danh sách nguyên liệu kèm nhà cung cấp và xuất xứ

Đại số quan hệ:

πnl.ten\_nguyen\_lieu,nl.xuat\_xu,ncc.ten\_nha\_cung\_cap​((NguyenLieu⋈NguyenLieu.ten\_nguyen\_lieu=CungCap.ten\_nguyen\_lieu​CungCap)⋈CungCap.id\_nha\_cung\_cap=NhaCungCap.id\_nha\_cung\_cap​NhaCungCap)

SELECT nl.ten\_nguyen\_lieu, nl.xuat\_xu, ncc.ten\_nha\_cung\_cap

FROM NguyenLieu nl

JOIN CungCap cc ON nl.ten\_nguyen\_lieu = cc.ten\_nguyen\_lieu

JOIN NhaCungCap ncc ON cc.id\_nha\_cung\_cap = ncc.id\_nha\_cung\_cap;



# **Phân quyền quản lý người dùng**

* ***Khởi tạo 3 người dùng có 3 vai trò khác nhau***

CREATE LOGIN AdminUser WITH PASSWORD = 'Admin@123';

CREATE LOGIN QuanLyUser WITH PASSWORD = 'QuanLy@123';

CREATE LOGIN NhanVienUser WITH PASSWORD = 'NhanVien@123';

* ***Tạo user trong các database***

CREATE USER AdminUser FOR LOGIN AdminUser;

CREATE USER QuanLyUser FOR LOGIN QuanLyUser;

CREATE USER NhanVienUser FOR LOGIN NhanVienUser;

* ***Cấp quyền cho từng user sao cho phù hợp***

Admin sẽ được cấp toàn bộ quyền như xem, sửa, xóa, thêm thông tin cho database

GRANT CONTROL ON DATABASE::[Bai1] TO AdminUser;

Quản lý sẽ chỉ có toàn quyền trong việc thêm, sửa, xóa, xem thông tin Nhân Viên, Ca làm nhưng chỉ được xem các thông tin còn lại liên quan đến cửa hàng như hóa đơn, nguyên liệu…

GRANT SELECT, INSERT, UPDATE, DELETE ON NhanVien TO QuanLyUser;

GRANT SELECT, INSERT, UPDATE, DELETE ON CaLam TO QuanLyUser;

GRANT SELECT ON HoaDon TO QuanLyUser;

GRANT SELECT ON NguyenLieu TO QuanLyUser;

GRANT SELECT ON DoUong TO QuanLyUser;

GRANT SELECT ON PhucVu TO QuanLyUser;

GRANT SELECT ON CungCap TO QuanLyUser;

GRANT SELECT ON CafeHouse TO QuanLyUser;

GRANT SELECT ON ThanhToan TO QuanLyUser;

GRANT SELECT ON NhaCungCap TO QuanLyUser;

GRANT SELECT ON BoiBan TO QuanLyUser;

GRANT SELECT ON QuanLi TO QuanLyUser;

GRANT SELECT ON PhaChe TO QuanLyUser;

Nhân viên sẽ chỉ được xem hóa đơn và menu đồ uống

GRANT SELECT ON HoaDon TO NhanVienUser;

GRANT SELECT ON DoUong TO NhanVienUser;

GRANT SELECT ON CafeHouse TO NhanVienUser;

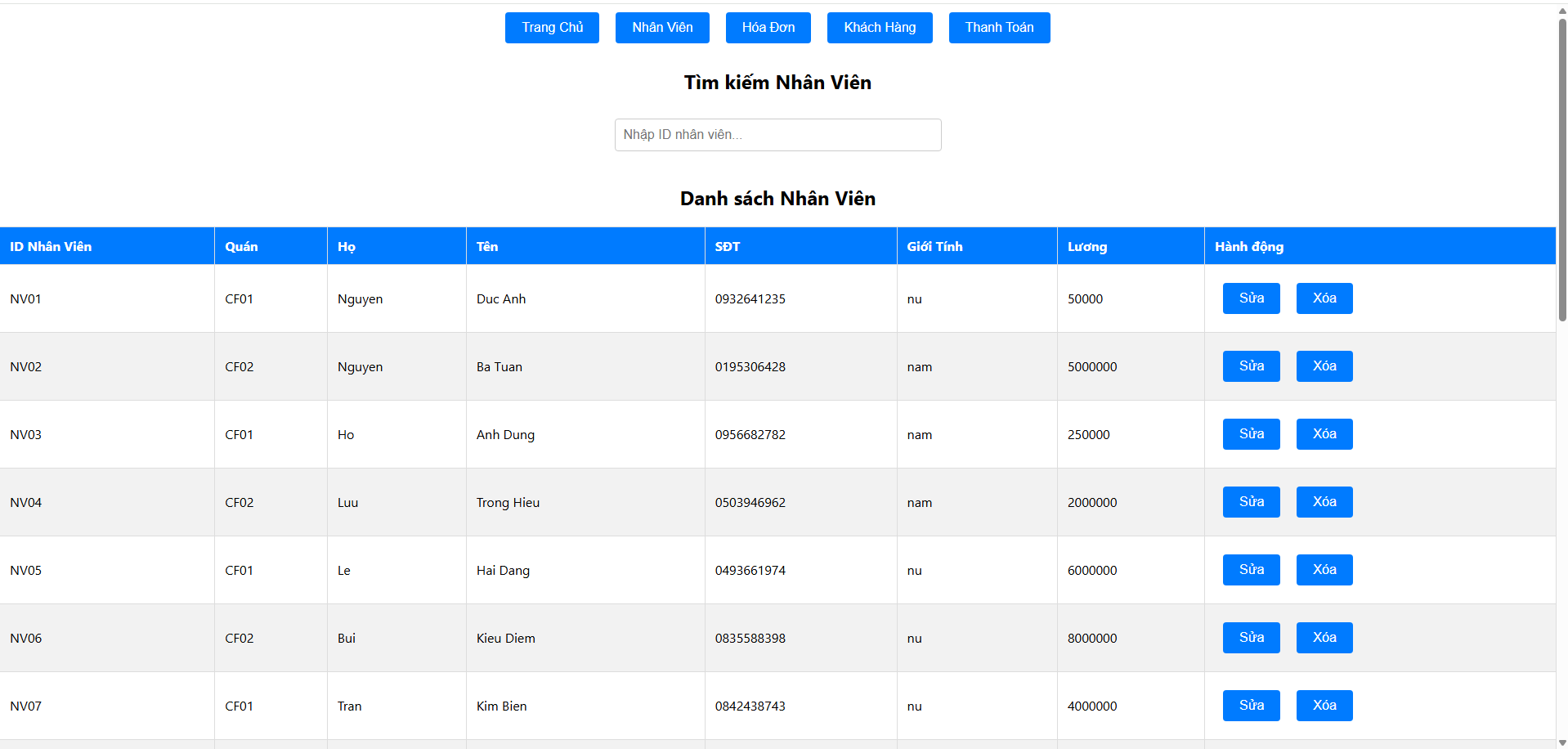
GRANT SELECT ON ThanhToan TO NhanVienUser;

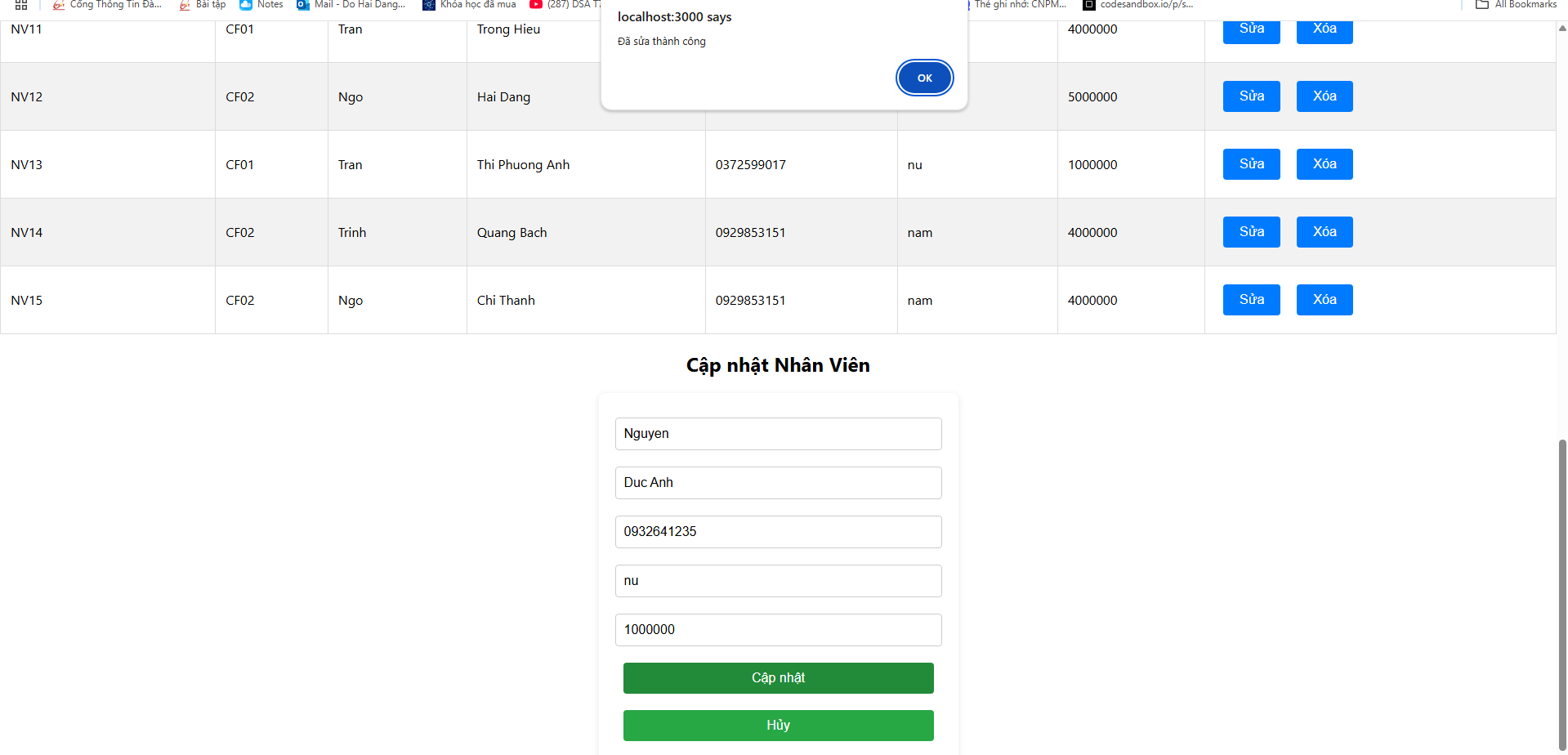
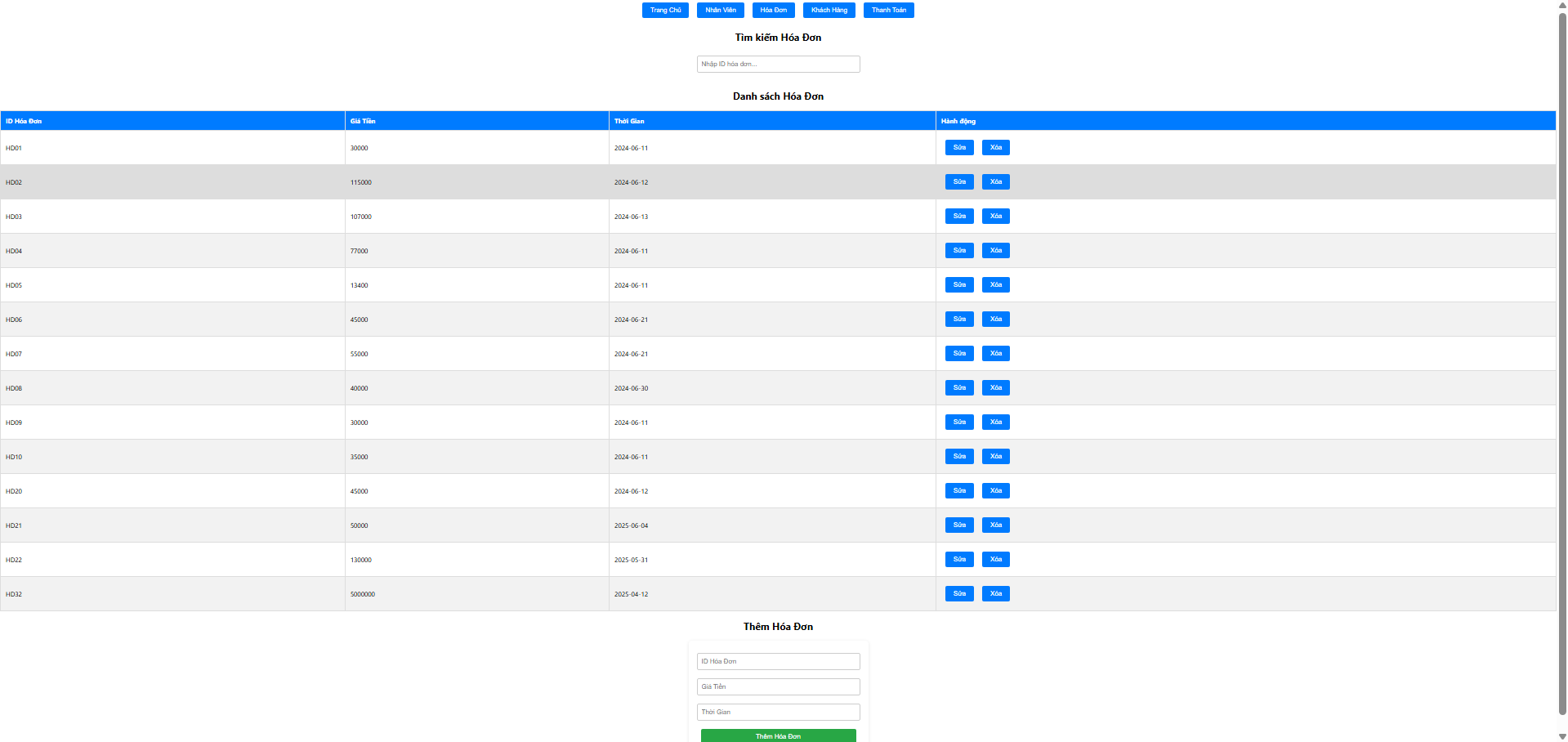
GRANT SELECT ON PhucVu TO NhanVienUser;

* ***Kiểm tra phân quyền***

Nếu ta đăng nhập vào tài khoản của Quản Lý thì ta sẽ có thể xem, sửa, xóa các thông tin của nhân viên nhưng không thể thay đổi của các bảng còn lại

1. **Hình ảnh giao diện**





Cài đặt CSDL trên một hệ quản trị CSDL khác:

* Hệ quản trị CSDL: MySQL
* Lí do lựa chọn hệ quản trị CSDL này:
  + Miễn phí và mã nguồn mở: Giúp tiết kiệm chi phí, dễ tùy chỉnh
  + Hiệu suất cao: Xử lí nhanh với khối lượng dữ liệu lớn và vừa
  + Bảo mật tốt: Hỗ trợ nhiều cơ chế bảo mật, phân quyền người dung
  + Phổ biến và được hỗ trợ rộng rãi: Dễ tìm tài liệu, cộng đồng lớn, nhiều công cụ hỗ trợ
  + Tương thích tốt: Làm việc tốt với nhiều nền tảng và ngôn ngữ lập trình: PHP, Java, Python, ...
* Cài đặt CSDL trên MySQL:

create database CAFE\_HOUSE;

use CAFE\_HOUSE;

CREATE TABLE CafeHouse (

id\_quan VARCHAR(10) PRIMARY KEY,

ten\_quan VARCHAR(30),

sdt VARCHAR(30),

dia\_chi VARCHAR(30),

co\_so INT

);

INSERT INTO CafeHouse (id\_quan, ten\_quan, sdt, dia\_chi, co\_so)

VALUES ('CF01', 'Nha go', '0988831458', 'Ho Van Quan', 1);

INSERT INTO CafeHouse (id\_quan, ten\_quan, sdt, dia\_chi, co\_so)

VALUES ('CF02', 'Nha go', '0988821341', 'CauGiay', 2);

CREATE TABLE NhanVien (

id\_nhan\_vien VARCHAR(30) PRIMARY KEY,

id\_quan VARCHAR(10),

ho VARCHAR(30),

ten VARCHAR(30),

sdt VARCHAR(30),

gioi\_tinh VARCHAR(30),

luong INT,

FOREIGN KEY (id\_quan) REFERENCES CafeHouse(id\_quan)

);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV01','CF01','Nguyen','Duc Anh', '0932641235','nam',10000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV02','CF02','Nguyen',' Ba Tuan', '0195306428','nam',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV03','CF01','Ho','Anh Dung', '0956682782','nam',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV04','CF02','Luu','Trong Hieu', '0503946962','nam',2000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV05','CF01','Le','Hai Dang', '0493661974','nu',6000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV06','CF02','Bui','Kieu Diem', '0835588398','nu',8000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV07','CF01','Tran','Kim Bien', '0842438743','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV08','CF02','Ngo','Kim Quynh', '0906735660','nu',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV09','CF01','Tran','Thi Phuong', '0372569017','nu',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV10','CF02','Trinh','Nhu Hoa', '0929893151','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV11','CF01','Tran','Trong Hieu', '0842438741','nu',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV12','CF02','Ngo','Hai Dang', '0906735760','nam',5000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV13','CF01','Tran','Thi Phuong Anh', '0372599017','nu',1000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV14','CF02','Trinh','Quang Bach', '0929853151','nam',4000000);

insert into NhanVien(id\_nhan\_vien,id\_quan,ho,ten,sdt,gioi\_tinh,luong) values ('NV15','CF02','Ngo','Chi Thanh', '0929853151','nam',4000000);

CREATE TABLE BoiBan (

id\_nhan\_vien VARCHAR(30),

danh\_gia VARCHAR(30),

FOREIGN KEY (id\_nhan\_vien) REFERENCES NhanVien(id\_nhan\_vien)

);

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV02', '5 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV03', '4 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV04', '4 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV05', '5 sao');

insert into BoiBan(id\_nhan\_vien, danh\_gia) values ('NV06', '3 sao');

CREATE TABLE QuanLi (

id\_nhan\_vien VARCHAR(30),

kinh\_nghiem VARCHAR(30),

FOREIGN KEY (id\_nhan\_vien) REFERENCES NhanVien(id\_nhan\_vien)

);

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV01', '10 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV11', '8 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV12', '6 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV13', '4 nam');

insert into QuanLi(id\_nhan\_vien, kinh\_nghiem) values ('NV14', '2 nam');

CREATE TABLE PhaChe (

id\_nhan\_vien VARCHAR(30),

bang\_cap VARCHAR(30),

FOREIGN KEY (id\_nhan\_vien) REFERENCES NhanVien(id\_nhan\_vien)

);

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV07', 'level A');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV08', 'level B');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV09', 'level C');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV10', 'level A');

insert into PhaChe(id\_nhan\_vien, bang\_cap) values ('NV15', 'level B');

CREATE TABLE CaLam (

id\_nhan\_vien VARCHAR(30),

gio\_lam INT,

ngay\_lam DATE,

FOREIGN KEY (id\_nhan\_vien) REFERENCES NhanVien(id\_nhan\_vien)

);

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV01', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV02', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV03', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV04', 8, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV05', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV06', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV07', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV08', 4, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV09', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV10', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV11', 8, '2024-6-12');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV12', 4, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV13', 8, '2024-6-13');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV14', 4, '2024-6-11');

insert into CaLam(id\_nhan\_vien, gio\_lam, ngay\_lam) values ('NV15', 8, '2024-6-13');

CREATE TABLE KhachHang (

id\_khach\_hang VARCHAR(30) PRIMARY KEY,

sdt VARCHAR(30),

ho\_va\_ten VARCHAR(50)

);

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH01','0725757239', 'Nguyen Le Thu Ha');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH02','0957572397', 'Vu Quang Anh' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH03','0725574239', 'Ho Nhat Minh');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH04','0815832393', 'Ba Van Kim' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH05','0725183239', 'Truong Thu Phuong');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH06','0725757339', 'Nguyen Le Thu');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH07','0957572497', 'Vu Quang Liem' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH08','0725574339', 'Ho Nhat Thuc');

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH09','0815832493', 'Ba Van Quat' );

insert into KhachHang(id\_khach\_hang,sdt,ho\_va\_ten) values('KH10','0725183339', 'Truong Thu Ha');

CREATE TABLE DoUong (

id\_do\_uong VARCHAR(30) PRIMARY KEY,

ten\_do\_uong VARCHAR(30),

gia INT

);

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU01','cafe den', '30000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU02','cafe nau', '35000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU03','tra quyt', '40000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU04','tra vai', '45000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU05','cacao', '50000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU06','hong tra sua', '55000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU07','o long tra sua', '52000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU08','tra dao', '42000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU09','tra tac', '47000');

insert into DoUong(id\_do\_uong,ten\_do\_uong,gia) values ('DU10','cafe bac xiu', '37000');

CREATE TABLE PhucVu (

id\_khach\_hang VARCHAR(30),

id\_do\_uong VARCHAR(30),

id\_quan VARCHAR(10),

FOREIGN KEY (id\_khach\_hang) REFERENCES KhachHang(id\_khach\_hang),

FOREIGN KEY (id\_do\_uong) REFERENCES DoUong(id\_do\_uong),

FOREIGN KEY (id\_quan) REFERENCES CafeHouse(id\_quan)

);

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH01','DU01','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU01','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU02','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH02','DU05','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH03','DU06','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH03','DU07','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH04','DU03','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH04','DU10','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU08','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU09','CF01');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH05','DU04','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH06','DU04','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH07','DU06','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH08','DU03','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH09','DU01','CF02');

insert into PhucVu(id\_khach\_hang,id\_do\_uong,id\_quan) values ('KH10','DU02','CF02');

CREATE TABLE HoaDon (

id\_hoa\_don VARCHAR(30) PRIMARY KEY,

gia\_tien VARCHAR(30),

thoi\_gian DATE

);

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD01', '30000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD02', '115000', '2024-6-12');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD03', '107000', '2024-6-13');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD04', '77000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD05', '134000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD06', '45000', '2024-6-21');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD07', '55000', '2024-6-21');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD08', '40000', '2024-6-30');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD09', '30000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD10', '35000', '2024-6-11');

insert into HoaDon(id\_hoa\_don,gia\_tien,thoi\_gian) values ('HD20', '45000', '2024-6-12');

CREATE TABLE ThanhToan (

id\_hoa\_don VARCHAR(30),

id\_khach\_hang VARCHAR(30),

id\_quan VARCHAR(10),

FOREIGN KEY (id\_quan) REFERENCES CafeHouse(id\_quan),

FOREIGN KEY (id\_hoa\_don) REFERENCES HoaDon(id\_hoa\_don),

FOREIGN KEY (id\_khach\_hang) REFERENCES KhachHang(id\_khach\_hang)

);

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD01', 'KH01', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD02', 'KH02', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD03', 'KH03', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD04', 'KH04', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD05', 'KH05', 'CF01');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD06', 'KH06', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD07', 'KH07', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD08', 'KH08', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD09', 'KH09', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD10', 'KH10', 'CF02');

insert into ThanhToan(id\_hoa\_don, id\_khach\_hang, id\_quan) values ('HD20', 'KH05', 'CF02');

CREATE TABLE NhaCungCap (

id\_nha\_cung\_cap VARCHAR(30) PRIMARY KEY,

ten\_nha\_cung\_cap VARCHAR(30),

dia\_chi VARCHAR(30),

sdt VARCHAR(30)

);

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC01', 'Phong Ho Supplier', 'Ha Noi', '0819219785');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC02', 'Khoa Van Supplier', 'Hai Phong', '0828008645');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC03', 'Cong La Supplier', 'Nghe An', '0523575928');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC04', 'Duy Tran Company', 'Ha Tinh', '0304066831');

insert into NhaCungCap(id\_nha\_cung\_cap,ten\_nha\_cung\_cap,dia\_chi,sdt) values ('NCC05', 'Duy La Company', 'Ha Noi', '0773436692');

CREATE TABLE NguyenLieu (

ten\_nguyen\_lieu VARCHAR(30),

xuat\_xu VARCHAR(30),

gia INT,

PRIMARY KEY (xuat\_xu, ten\_nguyen\_lieu)

);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hong tra','Lang Son',1500000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hop vai','Bac Giang',2000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('hop dao','Hai Phong',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cafe arabica','Buon Ma Thuat',4000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cafe robusta','Buon Ma Thuat',6000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('bot cacao','Buon Ma Thuat',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('tra o long','Thai Nguyen',3000000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('qua chanh','Ha Noi',200000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('nuoc duong','Ha Noi',300000);

insert into NguyenLieu(ten\_nguyen\_lieu,xuat\_xu,gia) values('ong hut','Ha Noi',150000);

CREATE TABLE CungCap (

id\_quan VARCHAR(10),

id\_nha\_cung\_cap VARCHAR(30),

ten\_nguyen\_lieu VARCHAR(30),

xuat\_xu VARCHAR(30),

PRIMARY KEY (xuat\_xu, ten\_nguyen\_lieu),

FOREIGN KEY (id\_quan) REFERENCES CafeHouse(id\_quan),

FOREIGN KEY (id\_nha\_cung\_cap) REFERENCES NhaCungCap(id\_nha\_cung\_cap),

FOREIGN KEY (xuat\_xu, ten\_nguyen\_lieu) REFERENCES NguyenLieu(xuat\_xu, ten\_nguyen\_lieu)

);

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC01','hong tra', 'Lang Son');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC01','hop vai', 'Bac Giang');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC02','hop dao', 'Hai Phong');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC03','bot cafe arabica', 'Buon Ma Thuat');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF01','NCC03','bot cafe robusta', 'Buon Ma Thuat');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC03','bot cacao', 'Buon Ma Thuat');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC04','tra o long', 'Thai Nguyen');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC04','qua chanh', 'Ha Noi');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC05','nuoc duong', 'Ha Noi');

insert into CungCap(id\_quan,id\_nha\_cung\_cap,ten\_nguyen\_lieu,xuat\_xu) values ('CF02','NCC05','ong hut', 'Ha Noi');

* Sơ đồ các quan hệ:

