**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO BÀI TẬP LỚN**

**THIẾT KẾ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CỬA HÀNG**

**SIÊU THỊ CO.OP MART**

Giảng viên hướng dẫn: ThS. Phạm Thị Miên

Nhóm sinh viên thực hiện:

- Bùi Văn Tân MSV: 6051071104

- Nguyễn Công Chí 6051071009

- Võ Văn Trí 6051071126

- Hồ Ngọc Thống 6051071118

- Huỳnh Xuân Lãm 6051071062

TP. Hồ Chí Minh, năm 2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO BÀI TẬP LỚN**

**THIẾT KẾ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CỬA HÀNG**

**SIÊU THỊ CO.OP MART**

Giảng viên hướng dẫn: ThS. Phạm Thị Miên

Nhóm sinh viên thực hiện:

- Bùi Văn Tân MSV: 6051071104

- Nguyễn Công Chí 6051071009

- Võ Văn Trí 6051071126

- Hồ Ngọc Thống 6051071118

- Huỳnh Xuân Lãm 6051071062

TP. Hồ Chí Minh, năm 2021

**LỜI CẢM ƠN**

Lời đầu tiên chúng em xin gửi lời cảm ơn chân thành đến quý thầy, cô giáo trong **Bộ môn Công nghệ thông tin – Phân hiệu Trường Đại học Giao thông vận tải.**

Những người đã truyền dạy, đã trang bị cho chúng em kho tàng kiến thức về bầu trời công nghệ thông tin rộng lớn.

Ở đây, chúng em không chỉ học được kiến thức về sách vở mà chúng em còn học được các bài học, kỷ năng sống trước khi tạm biệt mái trường đại học thân yêu này và tiến ra biển đời mênh mông rộng lớn.

Trong quá trình học tập và tìm hiểu nhóm chúng em đã nỗ lực rất nhiều với mong muốn hoàn thành bài tập lớn một cách tốt nhất, nhưng sẽ không tránh khỏi những thiếu sót. Nhóm chúng em mong thầy, cô có thể thông cảm và cho nhóm em những ý kiến, đóng góp để nhóm em có thể hoàn thành bài tập lớn của nhóm một cách trọn vẹn.

Sau cùng, chúng em xin kính chúc Quý Thầy Cô lời chúc sức khoẻ, luôn hạnh phúc và thành công hơn nữa trong công việc cũng như trong cuộc sống.

Nhóm chúng em xin chân thành cảm ơn!

# 

**NHẬN XÉT CỦA GIÁO VIÊN**

*Tp. Hồ Chí Minh, ngày … tháng … năm …*

Giáo viên hướng dẫn

**MỤC LỤC**

[**MỞ ĐẦU** 1](#_Toc72505628)

[**1.** **Lý do chọn đề tài.** 1](#_Toc72505629)

[**2.** **Hướng tiếp cận của đề tài.** 1](#_Toc72505630)

[**3.** **Mục tiêu nghiên cứu.** 1](#_Toc72505631)

[**4.** **Đối tượng và phạm vi nghiên cứu.** 1](#_Toc72505632)

[**5.** **Phương pháp nghiên cứu.** 1](#_Toc72505633)

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT** 2](#_Toc72505634)

[**1.1** **Hệ quản trị cơ sở dữ liệu SQL Server.** 2](#_Toc72505635)

[**1.1.1** **Giới thiệu.** 2](#_Toc72505636)

[**1.1.2** **Khái niệm.** 2](#_Toc72505637)

[**1.1.3** **Kiến trúc của hệ quản trị cơ sở dữ liệu SQL Server.** 3](#_Toc72505638)

[**1.1.4** **Ưu và nhược điểm.** 5](#_Toc72505639)

[**1.2** **Ngôn ngữ lập trình Java.** 5](#_Toc72505640)

[**1.2.1** **Giới thiệu.** 5](#_Toc72505641)

[**1.2.2** **Khái niệm.** 6](#_Toc72505642)

[**1.2.3** **Các đặc điểm của ngôn ngữ lập trình java**. 6](#_Toc72505643)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ** 8](#_Toc72505644)

[**2.1** **Mô tả chi tiết bài toán.** 8](#_Toc72505645)

[**2.2** **Lược đồ quan hệ** 8](#_Toc72505646)

[**2.2.1.** **Thực thể CHI NHÁNH** 8](#_Toc72505647)

[**2.2.2.** **Thực thể NHÂN VIÊN** 8](#_Toc72505648)

[**2.2.3.** **Thực thể KHÁCH HÀNG** 9](#_Toc72505649)

[**2.2.4.** **Thực thể TÀI KHOẢN** 9](#_Toc72505650)

[**2.2.5.** **Thực thể GIAO DỊCH** 9](#_Toc72505651)

[**2.2.6.** **Thực thể XỬ LÝ NHU CẦU** 9](#_Toc72505652)

[**2.2.7.** **Thực thể QUẢN LÝ** 10](#_Toc72505653)

[**2.2.8.** **Thực thể CHI TIẾT GIAO DỊCH** 10](#_Toc72505654)

[**2.3** **Sơ đồ thực thể các mối liên kết** 10](#_Toc72505655)

[**a.** **Xét thực thể CHINHANH và NHANVIEN** 10](#_Toc72505656)

[**b.** **Xét thực thể NHANVIEN và KHACHHANG** 10](#_Toc72505657)

[**c.** **Xét thực thể TAIKHOAN với NHANVIEN** 11](#_Toc72505658)

[**d.** **Xét thực thể TAIKHOAN và KHACHHANG** 11](#_Toc72505659)

[**e.** **Xét thực thể TAIKHOAN và GIAODICH** 11](#_Toc72505660)

[**2.4** **Sơ đồ ER** 12](#_Toc72505661)

[**2.5** **Mô hình quan hệ** 12](#_Toc72505662)

[**2.6** **Lược đồ cơ sở dữ liệu** 13](#_Toc72505663)

[**2.7 Các loại bảng** 13](#_Toc72505664)

[**2.7.1.** **Bảng CHINHANH** 13](#_Toc72505665)

[**2.7.2.** **Bảng NHANVIEN** 14](#_Toc72505666)

[**2.7.3.** **Bảng KHACHHANG** 14](#_Toc72505667)

[**2.7.4.** **Bảng XULYNHUCAU** 14](#_Toc72505668)

[**2.7.5.** **Bảng TAIKHOAN** 15](#_Toc72505669)

[**2.7.6.** **Bảng thông tin QUANLY** 15](#_Toc72505670)

[**2.7.7.** **Bảng thông tin GIAODICH** 15](#_Toc72505671)

[**2.7.8.** **Bảng thông tin CHITIETGIAODICH** 16](#_Toc72505672)

[**2.8 Danh sách các ràng buộc** 16](#_Toc72505673)

[**2.8.1** **Ràng buộc miền giá trị** 16](#_Toc72505674)

[**2.8.2** **Ràng buộc liên thuộc tính** 17](#_Toc72505675)

[**2.8.3** **Ràng buộc liên bộ** 18](#_Toc72505676)

[**2.9** **Danh sách phụ thuộc hàm** 19](#_Toc72505677)

[**2.9.1 CHINHANH**(MaCN,TenCN,DiaChi) 19](#_Toc72505678)

[**2.9.2 NHANVIEN**(MaNV,TenNV,NgayVL,MaCN) 19](#_Toc72505679)

[**2.9.3 KHACHHANG**(CMND, TenKH, NgaySinh, GioiTinh, SoDienThoai, DiaChi) 19](#_Toc72505680)

[**2.9.4 XULYNHUCAU**(CMND,MaNV,NgayLamViec,NhuCau) 19](#_Toc72505681)

[**2.9.5 TAIKHOAN**(SoTK,TenDangNhap,MatKhau,NgayDangKy,SoThe,SoDu,CMND) 19](#_Toc72505682)

[**2.9.6 QUANLY**(MaNV, SoTK,NgayKT,GhiChu) 20](#_Toc72505683)

[**2.9.7 GIAODICH**(MaGD, LoaiGD) 20](#_Toc72505684)

[**2.9.8 CHITIETGIAODICH**(MaGD, SoTK, SoTKNhan, NgayGD, SoTien,GhiChu) 20](#_Toc72505685)

[**3.1 Ngôn ngữ truy vấn** 20](#_Toc72505686)

[**a.** **Mệnh đề SELECT** 20](#_Toc72505687)

[**b. Mệnh đề WHERE** 20](#_Toc72505688)

[**c. Mệnh đề ORDER BY** 20](#_Toc72505689)

[**d.** **Mệnh đề GROUP BY** 21](#_Toc72505690)

[**e.** **Mệnh đề HAVING** 21](#_Toc72505691)

[**f.** **Truy vấn từ nhiều bảng** 22](#_Toc72505692)

[**g.** **Truy vấn lồng** 22](#_Toc72505693)

[**3.2 Các lệnh cập nhật dữ liệu** 22](#_Toc72505694)

[**a. Lệnh tạo bảng CREATE TABLE** 22](#_Toc72505695)

[**b. Thêm dòng vào bảng** 23](#_Toc72505696)

[**c. Sửa dòng trong bảng** 23](#_Toc72505697)

[**d. Xóa dòng trong bảng** 24](#_Toc72505698)

[**e. Lệnh sửa bảng** 25](#_Toc72505699)

[**f. Lệnh xóa bảng** 26](#_Toc72505700)

[**3.3 Chuẩn hóa quan hệ** 26](#_Toc72505701)

[**KẾT LUẬN VÀ KHUYẾN NGHỊ** 27](#_Toc72505702)

[**Kết quả đạt được** 27](#_Toc72505703)

[**Hiện tại** 27](#_Toc72505704)

[**Hướng phát triển** 27](#_Toc72505705)

[**Ưu điểm** 27](#_Toc72505706)

[**Nhược điểm** 27](#_Toc72505707)

[**Công việc của mỗi thành viên và mức độ hoàn thành** 28](#_Toc72505708)

[**Tài liệu tham khảo** 28](#_Toc72505709)

# **MỞ ĐẦU**

1. **Lý do chọn đề tài.**

Ngày nay việc giao dịch thông qua tiền tệ ngày càng phổ biến với mọi người, nhất là trong những nước có nền kinh tế phát triển năng động rất nhiều chi nhánh ‘mọc lên’ khắp nơi để phục vụ khách hàng. Ở những ngân hàng, một ngày có thể có hàng trăm khách hàng đến để thực hiên giao dịch. Những thông tin về khách hàng cần phải được ghi lại và lên danh sách để phục vụ cho việc kiểm soát và công tác quản lý. Do số lượng người đến giao dịch quá lớn mỗi ngày nên việc lưu thông tin dưới dạng thủ công ghi chép là không hiệu quả, do đó cần có một hệ thống đủ lớn để thực hiện việc quản lý số lượng lớn về dữ liệu như thế. Cùng với sự phát triển nhanh chóng của công nghệ thông tin nói chung, các phần mềm cũng được phát triển theo, phục vụ ngày càng nhiều cho công việc của con người ngày càng hiệu quả hơn, đỡ tốn công sức và quản lý công việc một cách chặt chẽ hơn, chống được thất thoát về tư liệu, tài chính cũng như nhân sự. Quản lí tài khoản ngân hàng cũng là một phần mềm như vậy.

1. **Hướng tiếp cận của đề tài.**

* Nghiên cứu và hiểu được cách thức hoạt động của SQL Server.
* Thiết kế cơ sở dữ liệu sao cho phù hợp với ứng dụng đang phát triển.
* Nắm vững kiến thức nền tảng ngôn ngữ lập trình Java, để phát triển ứng dụng.
* Tiến hành kiểm tra và chạy thử ứng dụng.

1. **Mục tiêu nghiên cứu.**

* Hiểu được cách thức hoạt động của cơ sở dữ liệu.
* Khiến ứng dụng trở nên có ích hơn cho xã hội.

1. **Đối tượng và phạm vi nghiên cứu.**

* Đối tượng: Sử dụng cho admin để quản lí thông tin tài khoản.
* Phạm vi ứng dụng: Sử dụng trong hệ thống ngân hàng.

1. **Phương pháp nghiên cứu.**

* Tài liệu: Tìm hiểu và khai thác các tài liệu liên quan đến cơ sở dữ liệu, các nền tảng kiến thức về Java.
* Thực hành: Tiến hành kết nối cơ sở dữ liệu với Java. Tiến hành thao tác với dữ liệu.

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

1. **Hệ quản trị cơ sở dữ liệu SQL Server.**
2. **Giới thiệu.**

SQL server chính là cụm từ viết tắt của Structure Query Language được sử dụng nhiều trong các lĩnh vực bởi chức năng quản lý dữ liệu. Các ngôn ngữ cấp cao như: [Visual C](https://www.visualstudio.com/vs/cplusplus/), [Oracle](https://www.oracle.com/database/index.html), Visual Basic,… đều có trình hỗ trợ là SQL. Những ứng dụng khi chạy phải sử dụng SQL khi người dùng truy cập tới cơ sở dữ liệu thì không cần sử dụng trực tiếp SQL.

1. **Khái niệm.**

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (Transact-SQL**)**để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server….

Một vài ấn bản SQL Server:

**Enterprise** : chứa tất cả cá đặc điểm nổi bật của SQL Server, bao gồm nhân bộ máy cơ sở dữ liệu và các dịch vụ đi kèm cùng với các công cụ cho tạo và quản lý phân cụm SQL Server. Nó có thể quản lý các CSDL lớn tới 524 [petabytes](https://en.wikipedia.org/wiki/Petabyte) và đánh địa chỉ 12 [terabytes](https://en.wikipedia.org/wiki/Terabyte) bộ nhớ và hỗ trợ tới 640 bộ vi xử lý(các core của cpu)

**Standard** : Rất thích hợp cho các công ty vừa và nhỏ vì giá thành rẻ hơn nhiều so với Enterprise Edition, nhưng lại bị giới hạn một số chức năng cao cấp (advanced features) khác, edition này có thể chạy tốt trên hệ thống lên đến 4 CPU và 2 GB RAM.

**Developer** : Có đầy đủ các tính năng của Enterprise Edition nhưng được chế tạo đặc biệt như giới hạn số lượng người kết nối vào Server cùng một lúc…. Ðây là phiên bản sử dụng cho phát triển và kiểm tra ứng dụng. Phiên bản này phù hợp cho các cá nhân, tổ chức xây dựng và kiểm tra ứng dụng

**Workgroup**: ấn bản SQL Server Workgroup bao gồm chức năng lõi cơ sở dữ liệu nhưng không có các dịch vụ đi kèm. Chú ý phiên bản này không còn tồn tại ở SQL Server 2012.

**Express**: SQL Server Express dễ sử dụng và quản trị cơ sở dữ liệu đơn giản. Được tích hợp với Microsoft Visual Studio, nên dễ dàng để phát triển các ứng dụng dữ liệu, an toàn trong lưu trữ, và nhanh chóng triển khai. SQL Server Express là phiên bản miễn phí, không giới hạn về số cơ ở dữ liệu hoặc người sử dụng, nhưng nó chỉ dùng cho 1 bộ vi xử lý với 1 GB bộ nhớ và 10 GB file cơ sở dữ liệu. SQL Server Express là lựa chọn tốt cho những người dùng chỉ cần một phiên bản SQL Server 2005 nhỏ gọn, dùng trên máy chủ có cấu hình thấp, những nhà phát triển ứng dụng không chuyên hay những người yêu thích xây dựng các ứng dụng nhỏ.

1. **Kiến trúc của hệ quản trị cơ sở dữ liệu SQL Server.**



*Hình 1. Kiến trúc của hệ quản trị cơ sở dữ liệu SQL Server*

**Database Engine:** Cái lõi của SQL Server:

Ðây là một engine có khả năng chứa data ở các quy mô khác nhau dưới dạng table và support tất cả các kiểu kết nối (data connection) thông dụng của Microsoft như

ActiveX Data Objects (ADO), OLE DB, and Open Database Connectivity (ODBC).

Ngoài ra nó còn có khả năng tự điều chỉnh (tune up) ví dụ như sử dụng thêm các tài nguyên (resource) của máy khi cần và trả lại tài nguyên cho hệ điều hành khi một user log off.

**Replication:** Cơ chế tạo bản sao (Replica):

Giả sử bạn có một database dùng để chứa dữ liệu được các ứng dụng thường xuyên cập nhật. Một ngày đẹp trời bạn muốn có một cái database giống y hệt như thế trên một server khác để chạy báo cáo (report database) (cách làm này thường dùng để tránh ảnh hưởng đến performance của server chính). Vấn đề là report server của bạn cũng cần phải được cập nhật thường xuyên để đảm bảo tính chính xác của các báo cáo. Bạn không thể dùng cơ chế back up and restore trong trường hợp này.

**Integration Services (DTS):**Là một tập hợp các công cụ đồ họa và các đối tượng lập trình cho việc di chuyển, sao chép và chuyển đổi dữ liệu.

Nếu bạn làm việc trong một công ty lớn trong đó data được chứa trong nhiều nơi khác nhau và ở các dạng khác nhau cụ thể như chứa trong Oracle, DB2 (của IBM), SQL Server, Microsoft Access…. Bạn chắc chắn sẽ có nhu cầu di chuyển data giữa các server này (migrate hay transfer) và không chỉ di chuyển bạn còn muốn định dạng (format) nó trước khi lưu vào database khác, khi đó bạn sẽ thấy DTS giúp bạn giải quyết công việc trên dễ dàng.

**Analysis Services:** Một dịch vụ phân tích dữ liệu rất hay của Microsoft

Dữ liệu (Data) chứa trong database sẽ chẳng có ý nghĩa gì nhiều nếu như bạn không thể lấy được những thông tin (Information) bổ ích từ đó. Do đó Microsoft cung cấp cho bạn một công cụ rất mạnh giúp cho việc phân tích dữ liệu trở nên dễ dàng và hiệu quả bằng cách dùng khái niệm hình khối nhiều chiều (multi-dimension cubes) và kỹ thuật “khai phá dữ liệu” (data mining).

**Notification Services:** Dịch vụ thông báo Notification Services là nền tảng cho sự phát triển và triển khai các ứng dụng tạo và gửi thông báo. Notification Services có thể gửi thông báo theo địch thời đến hàng ngàn người đăng ký sử dụng nhiều loại thiết bị khác nhau.

**Reporting Services:** Reporting Services bao gồm các thành phần server và client cho việc tạo, quản lý và triển khai các báo cáo. Reporting Services cũng là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

**Full Text Search Service:** Dịch vụ SQL Server Full Text Search là một dịch vụ đặc biệt cho đánh chỉ mục và truy vấn cho dữ liệu văn bản không cấu trúc được lưu trữ trong các CSDL SQL Server. Đánh chỉ mục với Full Text Search có thể dduwowcj tạo trên bất kỳ cột dựa trên dữ liệu văn bản. Nó sẽ rất hiệu quả cho việc tìm các sử dụng toán tử LIKE trong SQL với trường hợp tìm văn bản.

**Service Broker :** Được sử dụng bên trong mỗi Instance, là môi trường lập trình cho việc các ứng dụng nhảy qua các Instance. Service Broker giao tiếp qua giao thức TCP/IP và cho phép các component khác nhau có thể được đồng bộ cùng nhau theo hướng trao đổi các message. Service Broker chạy như một phần của bộ máy cơ sở dữ liệu, cung cấp một nền tảng truyền message tin cậy và theo hàng đợi cho các ứng dụng SQL Server.

1. **Ưu và nhược điểm.**

* **Ưu điểm:**
* Không cần code: Rất dễ dàng để quản lý các hệ thống cơ sở dữ liệu bằng việc sử dụng SQL chuẩn mà không cần phải viết bất cứ dòng code nào.
* Tiêu chuẩn được quy định rõ ràng: SQL sử dụng hai tiêu chuẩn ISO và ANSI, trong khi với các non-SQL database không có tiêu chuẩn nào được tuân thủ.
* Tính di động: SQL có thể được sử dụng trong chương trình trong PCs, servers, laptops, và thậm chí cả mobile phones.
* Ngôn ngữ tương tác: Language này có thể được sử dụng để giao tiếp với cơ sở dữ liệu và nhận câu trả lời cho các câu hỏi phức tạp trong vài giây.
* Multiple data views: Với sự trợ giúp của ngôn ngữ SQL, người dùng có thể tạo các hiển thị khác nhau về cấu trúc cơ sở dữ liệu và cơ sở dữ liệu cho những người dùng khác nhau.
* **Nhược điểm:**
* Giao diện khó dùng: SQL có giao diện phức tạp khiến một số người dùng khó truy cập.
* Không được toàn quyền kiểm soát: Các lập trình viên sử dụng SQL không có toàn quyền kiểm soát cơ sở dữ liệu do các quy tắc nghiệp vụ ẩn.
* Thực thi: Hầu hết các chương trình cơ sở dữ liệu SQL đều có phần mở rộng độc quyền riêng của nhà cung cấp bên cạnh các tiêu chuẩn SQL.
* Giá cả: Chi phí vận hành của một số phiên bản SQL khiến một số lập trình viên gặp khó khăn khi tiếp cận.

1. **Ngôn ngữ lập trình Java.**
2. **Giới thiệu.**

Ngôn ngữ [lập trình Java](http://hoclaptrinhweb.org/hoc-lap-trinh/hoc-lap-trinh-java.html) ban đầu được phát triển bởi Sun Microsystems do James Gosling khởi xướng và phát hành vào năm 1995 (Java 1.0 [J2SE]). Với ưu thế về đa nền tảng (multi platform) Java càng lúc càng được ứng dụng rộng rãi trên nhiều thiết bị từ máy tính đến mobile và nhiều thiết bị phần cứng khác...

1. **Khái niệm.**

Java là một trong những ngôn ngữ lập trình hướng đối tượng, được thiết kế độc lập với hệ điều hành, cho phép người lập trình viết chương trình một lần và có thể sử dụng tại bất kỳ đâu. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.

1. **Các đặc điểm của ngôn ngữ lập trình java**.

* **Hướng đối tượng** – Trong Java, mọi thứ đều là một Object. Java có thể dễ dàng mở rộng và bảo trì vì nó được xây dựng dựa trên mô hình Object.
* *Tính trừu tượng (Abstraction)*: là tiến trình xác định và nhóm các thuộc tính, các hành động liên quan đến một thực thể đặc thù, xét trong mối tương quan với ứng dụng đang phát triển.
* *Tính đa hình (Polymorphism):* cho phép một phương thức có các tác động khác nhau trên nhiều loại đối tượng khác nhau. Với tính đa hình, nếu cùng một phương thức ứng dụng cho các đối tượng thuộc các lớp khác nhau thì nó đưa đến những kết quả khác nhau. Bản chất của sự việc chính là phương thức này bao gồm cùng một số lượng các tham số.
* *Tính kế thừa (Inheritance)*: Điều này cho phép các đối tượng chia sẻ hay mở rộng các đặc tính sẵn có mà không phải tiến hành định nghĩa lại.
* *Tính đóng gói (Encapsulation)*:  là tiến trình che giấu việc thực thi những chi tiết của một đối tượng đối với người sử dụng đối tượng ấy.
* **Nền tảng độc lập** – Không giống nhiều ngôn ngữ lập trình khác bao gồm cả C và C ++, khi Java được biên dịch, nó không được biên dịch thành ngôn ngữ máy nền tảng cụ thể, thay vào mã byte – nền tảng độc lập. Mã byte này được thông dịch bởi máy ảo (JVM) trên nền tảng nào đó mà nó đang chạy.
* **Đơn giản** – Java được thiết kế để dễ học. Nếu bạn hiểu khái niệm cơ bản về OOP Java, sẽ rất dễ để trở thành master về java.
* **Bảo mật** – Với tính năng an toàn của Java, nó cho phép phát triển các hệ thống không có virut, giả mạo. Các kỹ thuật xác thực dựa trên mã hoá khóa công khai.
* **Kiến trúc – trung lập** – Trình biên dịch Java tạo ra định dạng tệp đối tượng kiến trúc trung lập, làm cho mã biên dịch được thực thi trên nhiều bộ vi xử lý, với sự hiện diện của hệ điều hành Java.
* **Portable** – Là kiến trúc tập trung và không có khía cạnh thực hiện phụ thuộc của đặc tả này làm cho Java khả chuyển. Trình biên dịch trong Java được viết bằng ANSI C, đó là một tập con POSIX.
* **Mạnh mẽ** – Java làm nỗ lực để loại trừ các tình huống dễ bị lỗi bằng cách kiểm tra lỗi tại thời gian biên dịch và kiểm tra lỗi tại runtime.
* **Đa luồng** – Với tính năng đa luồng của Java có thể viết các chương trình có thể thực hiện nhiều tác vụ đồng thời. Tính năng thiết kế này cho phép các nhà phát triển xây dựng các ứng dụng tương tác có thể chạy trơn tru hơn.
* **Thông dịch** – Mã byte Java được dịch trực tiếp tới các máy tính gốc và không được lưu trữ ở bất cứ đâu.
* **Hiệu năng cao** – Với việc sử dụng trình biên dịch Just-In-Time, Java cho phép thực hiện hiệu năng cao.
* **Phân tán** – Java được thiết kế cho môi trường phân tán của Internet.
* **Năng động** – Java là năng động hơn C hoặc C++ vì nó được thiết kế để thích nghi với môi trường đang phát triển. Các chương trình Java có thể mang một lượng lớn thông tin tại runtime mà có thể được sử dụng để xác minh và giải quyết các truy cập vào các đối tượng tại runtime.

# **CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ**

1. **Mô tả chi tiết bài toán.**

Để thực hiện chương trình cũng như cụ thể hóa một cách dễ hiểu nhất thì cần dựng bài toán, cụ thể như sau: Một ngân hàng sẽ có nhiều chi nhánh để quản lý. Thuộc tính của **chi nhánh** là mã chi nhánh, tên chi nhánh, địa chỉ. Một chi nhánh có nhiều nhân viên làm việc và nhân viên thì chỉ làm việc tại một chi nhánh. Thuộc tính xác định của **nhân viên** là mã nhân viên, tên nhân viên, ngày vào làm. Mỗi khách hàng đến giao dịch tại ngân hàng có thể được một hoặc nhiều nhân viên phục vụ giao dịch. Thông tin **khách hàng** là chứng minh nhân dân, tên khách hàng, ngày sinh, giới tính, số điện thoại, địa chỉ. Nhân viên trong ngày có thể xử lý nhu cầu của một hoặc nhiều khách hàng. **Xử lý nhu cầu** gồm nhu cầu và ngày làm việc. Để sử dụng dịch vụ hoặc giao dịch tại ngân hàng, khách hàng bắt buộc sẽ phải tạo một tài khoản. Một khách hàng có thể tạo nhiều tài khoản cùng lúc. Tài khoản của mỗi khách hàng là duy nhất. Thông tin **tài khoản** là số tài khoản, tên đăng nhập, mật khẩu, số thẻ, số dư và ngày đăng kí. Tài khoản sẽ được quản lý bởi nhân viên của ngân hàng. Một nhân viên có thể quản lý một hoặc nhiều tài khoản ngân hàng. Thuộc tính của **quản lý** gồm ngày kiểm tra và ghi chú. Sau khi tạo tài khoản, khách hàng có thể thực hiện một hoặc nhiều giao dịch tại ngân hàng. Thuộc tính **giao dịch** gồm mã giao dịch và tên giao dịch. Sau khi giao dịch xong, khách hàng sẻ được nhận một phiếu **chi tiết giao dịch** trong đó có số tài khoản nhận, ngày giao dịch, số tiền giao dịch ghi chú.

1. **Lược đồ quan hệ**
   * 1. **Thực thể CHI NHÁNH**

CHINHANH(MaCN, TenCN, DiaChi)

*Diễn giải*: Một ngân hàng sẽ có nhiều chi nhánh để quản lý. Mỗi chi nhánh bao gồm mã chi nhánh(MaCN) để phân biệt với chi nhánh khác. Mỗi chi nhánh còn được xác định bởi tên chi nhanh(TenCN) và địa chỉ (DiaChi).

* + 1. **Thực thể NHÂN VIÊN**

NHANVIEN(MaNV, TenNV,NgayVL)

*Diễn giải*: Mỗi nhân viên sẽ có mã nhân viên để phân biệt với nhân viên khác(MaNV), ngoài ra còn có tên nhân viên (TenNV) và ngày vào làm (NgayVL).

* + 1. **Thực thể KHÁCH HÀNG**

KHACHHANG(CMND, TenKH, NgaySinh, GioiTinh, SoDienThoai, DiaChi)

*Diễn giải*: Mỗi khách hàng sẽ có một số chứng minh nhân dân để phân biệt khách hàng với nhau(CMND). Một số thông tin của khách hàng như: tên khách hàng (TenKH), ngày sinh (NgaySinh), giới tính (GioiTinh), số điện thoại (SoDienThoai), địa chỉ (DiaChi).

* + 1. **Thực thể TÀI KHOẢN**

TAIKHOAN(SoTk,TenDangNhap,MatKhau,NgayDangKy,SoThe,SoDu,

CMND)

*Diễn giải*: Mỗi tài khoản sẽ có số tài khoản (SoTk) và CMND để biết tài khoản đó là của ai. Ngoài ra còn có các thuộc tính khác tên đăng nhập(TenDangNhap) , mật khẩu(MatKhau), ngày đăng ký(NgayDangKy), số thẻ(SoThe) và số dư(SoDu).

* + 1. **Thực thể GIAO DỊCH**

GIAODICH(MaGD,LoaiGD)

*Diễn giải*: Khách hàng có thể lựa chọn và thực hiện nhiều giao dịch tại ngân hàng, mỗi giao dịch sẽ có một mã giao dịch riêng (MAGD) và tên giao dịch (LoaiGD).

* + 1. **Thực thể XỬ LÝ NHU CẦU**

XULYNHUCAU(CMND,MaNV,NgayLamViec,NhuCau)

*Diễn giải*: Ngoài việc thực hiên các giao dịch tại ngân hàng, khách hàng còn có thể được xử lý một số vấn đề liên quan đến tài khoản của mình. Xử lý nhu cầu cần có CMND (chứng minh nhân dân) của khách và mã nhân viên(MaNV) để phân biệt với các công việc xử lý khác nhau. Ngoài ra còn có thuộc tính ngày làm việc(NgayLamViec) và nhu cầu(NhuCau).

* + 1. **Thực thể QUẢN LÝ**

QUANLY(MaNV,SoTk,NgayKT,GhiChu)

*Diễn giải*: Thực thể quản lý sẽ có mã nhân viên(MaNV) và số tài khoản(SoTK) để biết là ai xử lý và tài khoản nào cần được xử lý. Ngoài ra còn có thuộc tính ngày kiểm tra (NgayKT) và ghi chú(GhiChu).

* + 1. **Thực thể CHI TIẾT GIAO DỊCH**

CHITIETGIAODICH(MaGD,SoTK,SoTKNhan,NgayGD,SoTien,GhiChu)

*Diễn giải*: Mỗi phiếu giao dịch sẽ có mã giao dịch(MaGD), số tài khoản(SoTK) và số tài khoản nhận(SoTKNhan) để xác định việc giao dịch đó là của ai và với ai.Ngoài ra còn có các thuộc tính ngày giao dịch(NgayGD), số tiền(SoTien) và ghi chú(GhiChu).

1. **Sơ đồ thực thể các mối liên kết**
2. **Xét thực thể CHINHANH và NHANVIEN**

Ta thấy rằng một chi nhánh sẽ luôn có nhiều nhân viên làm việc và một nhân viên sẽ chỉ được làm tại một chi nhánh nhất định. Do đó mối quan hệ của NHANVIEN và CHINHANH sẽ là một – nhiều.

(1, 1) (1, n)

Thuộc

CHINHANH

NHANVIEN

HOSOBENHAN

1. **Xét thực thể NHANVIEN và KHACHHANG**

Ta thấy rằng một nhân viên có thể làm việc với một hoặc nhiều khách hàng trong một ngày và ngược lại khách hàng cũng có thể làm việc với một hoặc nhiều nhân viên trong lần giao dịch đó. Do đó mối quan hệ giữa nhân viên và khách hàng sẽ là nhiều - nhiều

(1, n) (1, n)

KHACHHANG

Xử lý nhu cầu

NHANVIEN

1. **Xét thực thể TAIKHOAN với NHANVIEN**

Ta thấy rằng một tài khoản sẽ được quản lý bởi một hoặc nhiều nhân viên và ngược lại một nhân viên cũng sẽ quản lý được nhiều tài khoản. Do đó mối quan hệ giửa NHANVIEN và TAIKHOAN sẽ là nhiều - nhiều.

(1, n) (1, n)

Quản lý

TAIKHOAN

NHANVIEN

1. **Xét thực thể TAIKHOAN và KHACHHANG**

Ta thấy rằng một khách hàng sẽ có một hoặc nhiều tài khoản trong ngân hàng và một tài khoản chỉ thuộc về một khách hàng duy nhất. Do đó mối quan hệ giữa TAIKHOAN và KHACHHANG là một - nhiều

(1, n) (1, 1)

TAIKHOAN

Có

KHACHHANG

1. **Xét thực thể TAIKHOAN và GIAODICH**

Ta xét rằng tại một thời điểm thì một tài khoản chỉ thực hiện được một giao dịch và ngược lại giao dịch đó chỉ được thực hiện bởi tài khoản này mà thôi. Do đó mối quan hệ giữa TAIKHOAN và GIAODICH là một- một.

Tuy nhiên nếu chúng ta xét trong một chu kì thời gian thì một tài khoản có thể thực hiện nhiều giao dịch và giao dịch đó cũng có thể thực hiện bởi nhiều tài khoản khác nhau.

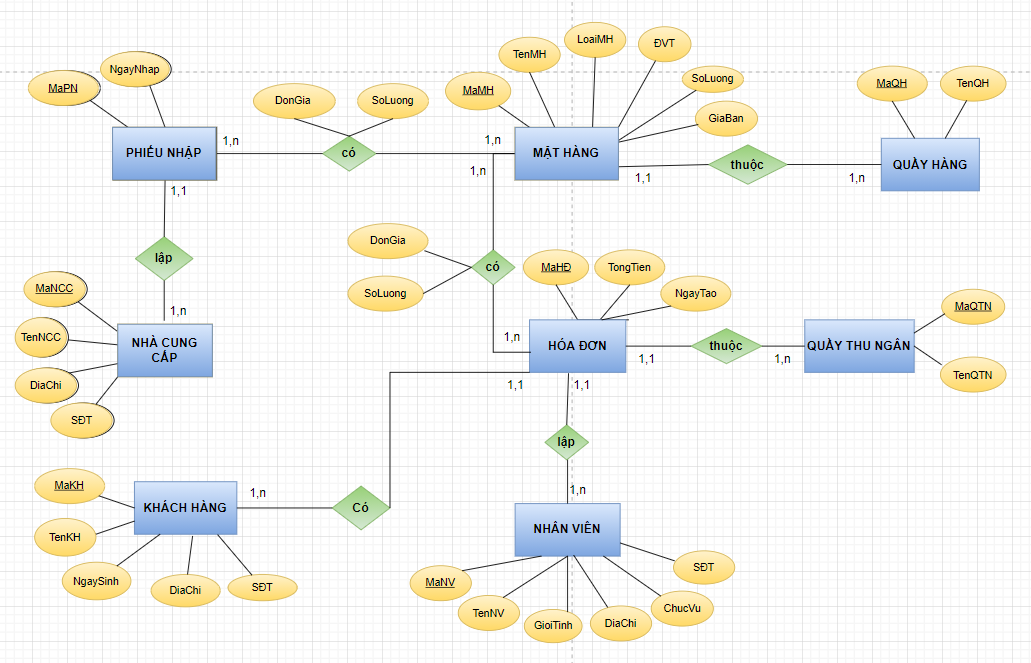
(1, 1) (1, 1)

Thực Hiện

Giao Dịch

TAIKHOAN

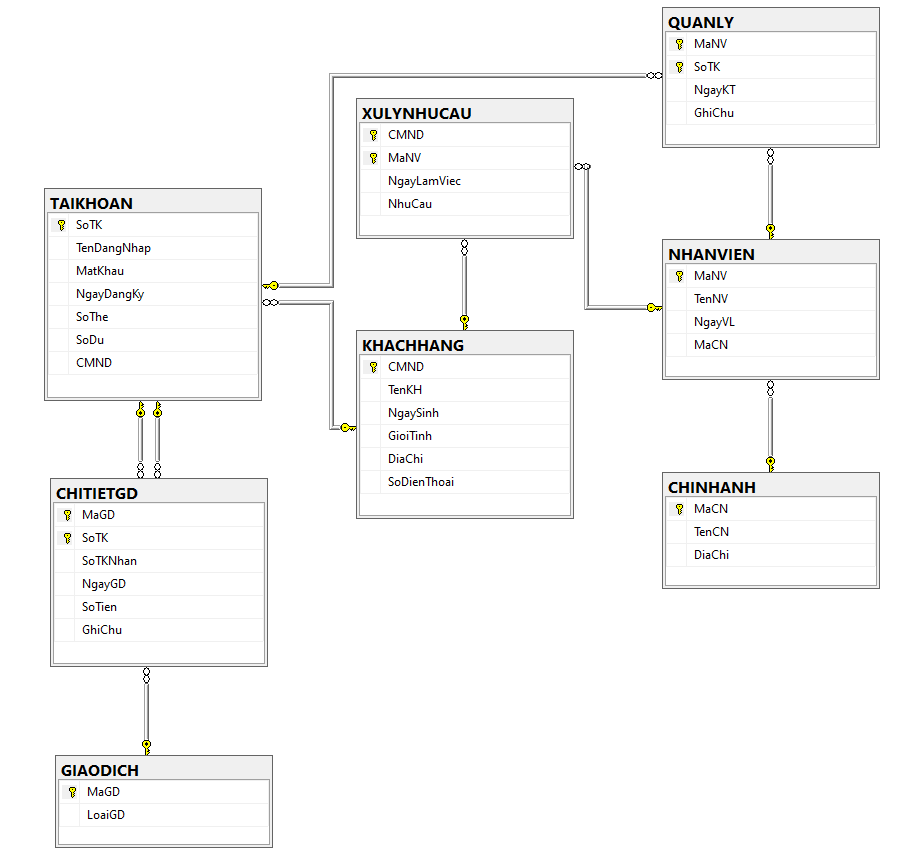
1. **Sơ đồ ER**



1. **Mô hình quan hệ**

* CHINHANH(MaCN, TenCN, DiaChi)
* NHANVIEN(MaNV, TenNV, NgayVL, **MaCN**)
* KHACHHANG(CMND, TenKH, NgaySinh, GioiTinh, SoDienThoai, DiaChi)
* TAIKHOAN(SoTk,TenDangNhap,MatKhau,NgayDK,SoThe,SoDu,**CMND**)
* XULYNHUCAU(CMND,MaNV,NgayLamViec,NhuCau)
* QUANLY(MaNV,SoTk,NgayKT,GhiChu)
* GIAODICH(MaGD,LoaiGD)
* CHITIETGIAODICH(MaGD,SoTK,NgayGD,SoTien,GhiChu, **SoTKNhan**)

1. **Lược đồ cơ sở dữ liệu**



## 2.7 Các loại bảng

* + 1. **Bảng CHINHANH**

Bảng chứa thông tin chi nhanh

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| MaCN(Khóa chính) | varchar[20] | Mã chi nhánh |
| TenCN | nvarchar[50] | Tên chi nhánh |
| DiaChi | nvarchar[50] | Địa chỉ |

* + 1. **Bảng NHANVIEN**

Bảng chứa thông tin nhân viên

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| MaNV(Khóa chính) | varchar[10] | Mã nhân viên |
| TenNV  NgayVL | nvarchar[30]  date | Tên nhân viên  Ngày vào làm |
| MaCN(Khóa ngoại) | varchcar[20] | Mã chi nhánh |

* + 1. **Bảng KHACHHANG**

Bảng chứa thông tin khách hàng

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| CMND(Khóa chính) | varchar[20] | Số chứng minh nhân dân |
| TenKH | nvarchar[30] | Họ và tên khách hàng |
| NgaySinh | date | Ngày tháng năm sinh |
| GioiTinh | bit | Giới tính khách hàng |
| DiaChi | nvarchar[50] | Địa chỉ khách hàng |
| SoDienThoai | varchar[10] | Số điện thoại khách hàng |

* + 1. **Bảng XULYNHUCAU**

Bảng chứa thông tin xử lý nhu cầu

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| CMND(Khóa chính) | varchar[20] | Số chứng minh nhân dân |
| MaNV(Khóa chính) | varchar[10] | Mã nhân viên |
| NgayLamViec | date | Ngày làm việc |
| NhuCau | nvarchar[50] | Nhu cầu của khách với nhân viên |

* + 1. **Bảng TAIKHOAN**

Bảng chứa thông tin tài khoản

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| SoTK(Khóa chính) | varchar[20] | Số tài khoản |
| TenDangNhap | nvarchar[30] | Tên đăng nhập |
| MatKhau | Varchar[20] | Mật khẩu tài khoản |
| NgayDK | date | Ngày đăng kí thẻ |
| SoThe | Varchar[20] | Số thẻ |
| SoDu | Float | Số dư trong thẻ |
| CMND(Khóa ngoại) | Varchar[20] | Số chưng minh nhân dân |

* + 1. **Bảng thông tin QUANLY**

Bảng chứa thông tin bảng quản lý

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| MaNV(Khóa chính) | varchar[10] | Mã nhân viên |
| SoTK(Khóa chính) | varchar[20] | Số tài khoản |
| NgayDK | date | Ngày đăng ký thẻ |
| GhiChu | Nvarchar[50] | Ghi chú |

* + 1. **Bảng thông tin GIAODICH**

Bảng chứa thông tin giao dịch

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| MaGD(Khóa chính) | varchar[20] | Mã giao dịch |
| LoaiGD | nvarchar[30] | Loại giao dịch |

* + 1. **Bảng thông tin CHITIETGIAODICH**

Bảng chứa thông tin chi tiết giao dịch

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Chú thích** |
| MaGD(Khóa chính) | varchar[20] | Mã giao dịch |
| SoTK(Khóa chính) | Varchar[20] | Số tài khoản |
| SoTKNhan(Khóa ngoại) | Varchar[20] | Số tài khoản giao dịch |
| NgayGD | date | Ngày giao dịch |
| SoTien | float | Số tiền giao dịch |
| GhiChu | Nvarchar[70] | Ghi chú |

## 2.8 Danh sách các ràng buộc

*Ký hiệu:*

RB : Ràng buộc

+ : Có thể gây ra vi phạm ràng buộc

– : Không thể gây ra vi phạm ràng buộc

–(\*) : Nếu không được vi phạm vì không được phép sửa đổi

* + 1. **Ràng buộc miền giá trị**
* **Ràng buộc 1: Trong quan hệ KHACHHANG thuộc tính giới tính chỉ có thể là Nam hoặc Nữ**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Khách Hàng | +(GioiTinh) | \_ | – (\*) |

* **Ràng buộc 2: Trong quan hệ TAIKHOAN thì số dư mỗi tài khoản phải lớn > 50000**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Tài Khoản | +(SoDu) | \_ | –(\*) |

* **Ràng buộc 3: Trong quan hệ CHITIETGIAODICH thì số tiền giao dịch tối thiểu phải là 50000**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Chi Tiết Giao Dịch | -(\*) | \_ | +(SoTien) |

* + 1. **Ràng buộc liên thuộc tính**
* **Ràng buộc 1: Trong quan hệ Chi Tiết Giao Dịch thì số tài khoản nhận phải khác số tài khoản**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Chi Tiết Giao Dịch | +(SoTk) | \_ | +(SoTKNhan) |

* **Ràng buộc 2: Trong quan hệ Nhân Viên với Quản Lý và Xử lý Nhu Cầu thì NgayVaoLam của nhân viên phải nhỏ hơn hoặc bằng với NgayKT và NgayLamViec**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Nhân Viên | +(NgaySinh | \_ | -(\*) |
| Quản Lý | + | - | +(NgayKT) |
| Xử Lý Nhu Cầu | + | - | +(NgayLamViec) |

* **Ràng buộc 3: Trong quan hệ Tài Khoản với Chi Tiết Giao Dịch thì số tiền giao dịch phải nhỏ hơn số dư trong tài khoản. Ngoài ra ngày đăng ký làm thẻ phải nhỏ hơn ngày giao dịch**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Chi Tiết Giao Dịch | +(SoTien)  +(NgayGD) | \_ | -(\*) |
| Tài Khoản | \_(\*) | \_ | +(SoDu)  +(NgayDangKy) |

* + 1. **Ràng buộc liên bộ**
* **Ràng buộc 1: Trong quan hệ CHINHANH mỗi chi nhánh sẽ phải có mã chi nhánh MaCN khác nhau**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Chi Nhánh | +(MaCN) | \_ | +(MaCN) |

* **Ràng buộc 2: Trong quan hệ NHANVIEN mỗi nhân viên sẽ phải có mã nhân viên MaNV khác nhau**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Nhân Viên | +(MaNV) | \_ | +(MaNV) |

* **Ràng buộc 3: Trong quan hệ KHACHHANG mỗi khách hàng sẽ phải có một số CMND khác nhau để phân biệt**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Khách Hàng | +(CMND) | \_ | +(CMND) |

* **Ràng buộc 4: Trong quan hệ TAIKHOAN mỗi tài khoản sẽ phải có số tài khoản SoTK khác nhau**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Tài Khoản | +(SoTK) | \_ | +(SoTK) |

* **Ràng buộc 5: Trong quan hệ GIAODICH mỗi giao dịch sẽ phải có một mã giao dịch MaGD khác nhau**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quan hệ** | **Thêm** | **Xóa** | **Sửa** |
| Giao Dịch | +(MaGD) | \_ | +(MaGD) |

* 1. **Danh sách phụ thuộc hàm**
     1. CHINHANH(MaCN,TenCN,DiaChi)

F1: MaCN TenCN

F2: MaCN 🡪 DiaChi

* + 1. NHANVIEN(MaNV,TenNV,NgayVL,MaCN)

F1:MaNV,MaCN TenNV

F2:MaNV,MaCN 🡪NgayVL

* + 1. KHACHHANG(CMND, TenKH, NgaySinh, GioiTinh, SoDienThoai, DiaChi)

F1: CMNDTenTK

F2: CMNDNgaySinh

F3: CMNDGioiTinh

F4: CMNDSoDienThoai

F5: CMNDDiaChi

* + 1. XULYNHUCAU(CMND,MaNV,NgayLamViec,NhuCau)

F1: CMND,MaNVNgayLamViec

F2: CMND,MaNVNhuCau

* + 1. TAIKHOAN(SoTK,TenDangNhap,MatKhau,NgayDangKy,SoThe,SoDu,CMND)

F1: SoTK,CMND TenDangNhap

F2: SoTK,CMNDMatKhau

F3: SoTKSoThe

F4: SoTKSoDu

F5: SoTKNgayDangKy

* + 1. QUANLY(MaNV, SoTK,NgayKT,GhiChu)

F1:MANV, SOTK NgayKT

F2:MANV,SOTK GhiChu

* + 1. GIAODICH(MaGD, LoaiGD)

F1:MaGD LoaiGD

* + 1. CHITIETGIAODICH(MaGD, SoTK, SoTKNhan, NgayGD, SoTien,GhiChu)

F1: MaGD, SoTK,SoTKNhan 🡪 NgayGD

F2: MaGD, SoTK,SoTKNhan 🡪 SoTien

F3: MaGD, SoTK,SoTKNhan 🡪 GhiChu

**3.1 Ngôn ngữ truy vấn**

* 1. **Mệnh đề SELECT**

SELECT \* FROM NHANVIEN



### b. Mệnh đề WHERE

SELECT \* FROM NHANVIEN NV

WHERE NV.MaNV = ‘NV001’

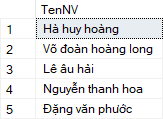
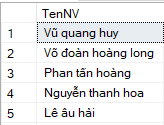


**c. Mệnh đề ORDER BY**

SELECT TenNV

FROM NHANVIEN

ORDER BY TenNV DESC

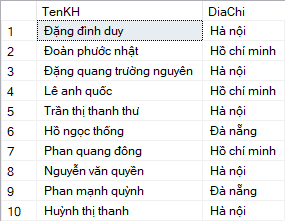
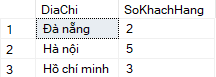
 🡪 

1. **Mệnh đề GROUP BY**

SELECT DiaChi, COUNT(\*) AS SoKhachHang

FROM KHACHHANG

GROUP BY DiaChi

 🡪 

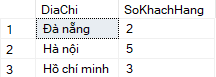
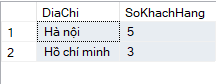
1. **Mệnh đề HAVING**

SELECT DiaChi, COUNT(\*) AS SoKhachHang

FROM KHACHHANG

GROUP BY DiaChi

HAVING COUNT(\*) > 2

 🡪 

1. **Truy vấn từ nhiều bảng**

**Liệt kê danh sách nhân viên của từng chi nhánh:**

SELECT CN.MaCN, NV.MaNV, NV.TenNV

FROM CHINHANH AS CN, NHANVIEN AS NV

WHERE CN.MaCN = NV.MaCN

1. **Truy vấn lồng**

**Tìm thông tin những khách hàng chưa chuyển tiền lần nào :**

SELECT \*

FROM KHACHHANG AS KH, TAIKHOAN AS TK

WHERE TK.CMND = KH.CMND

AND TK.SoTK NOT IN (

SELECT CTGD.SoTK

FROM TAIKHOAN AS TK, CHITIETGD AS CTGD

WHERE CTGD.SoTK = TK.SoTK )

**3.2** **Các lệnh cập nhật dữ liệu**

1. **Lệnh tạo bảng CREATE TABLE**

CREATE TABLE KHACHHANG

(

CMND VARCHAR(20),

TenKH NVARCHAR(30),

NgaySinh DATE,

GioiTinh BIT,

DiaChi NVARCHAR(50),

SoDienThoai VARCHAR(10),

CONSTRAINT PK\_KHACHHANG PRIMARY KEY(CMND)

)

1. **Thêm dòng vào bảng**

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558671', N'Đặng đình duy', '1980-1-20',1, N'Hà nội', '0982753684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558672', N'Đoàn phước nhật', '1980-12-2',1,N'Hồ chí minh', '0382753684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558673', N'Đặng quang trường nguyên','2019-2-20',1, N'Hà nội','0382853684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558674', N'Lê anh quốc', '1990-3-20',1, N'Hồ chí minh', '0982750684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558675', N'Trần thị thanh thư', '2019-11-20',0, N'Hà nội', '0982703684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558676', N'Hồ ngọc thống', '1990-11-20',1, N'Đà nẵng', '0082753684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558677', N'Phan quang đông', '1980-7-20',1, N'Hồ chí minh', '0380753684')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

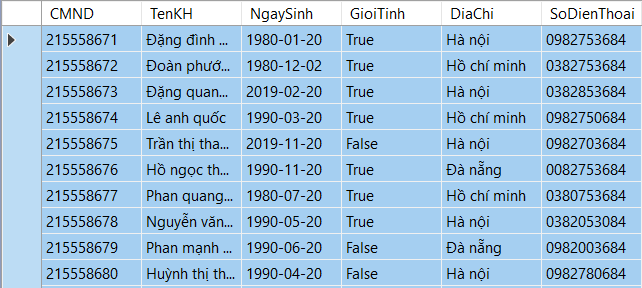
VALUES('215558678', N'Nguyễn văn quyền', '1990-5-20',1, N'Hà nội', '0382053084')

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558679', N'Phan mạnh quỳnh', '1990-6-20',0, N'Đà nẵng', '0982003684’)

INSERT INTO dbo.KHACHHANG(CMND,TenKH,NgaySinh,GioiTinh,DiaChi,SoDienThoai)

VALUES('215558680', N'Huỳnh thị thanh', '1990-4-20',0, N'Hà nội', '0982780684')



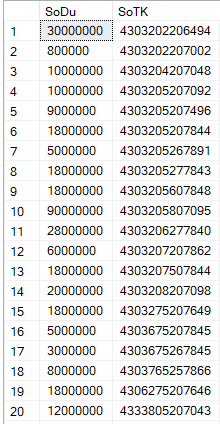
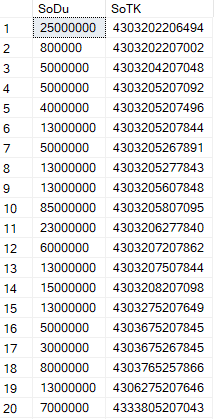
1. **Sửa dòng trong bảng**

**Giảm số tiền 5,000000 cho tất cả tài khoản có số dư lớn hơn 8,000000**

UPDATE TAIKHOAN

SET SoDu = SoDu - 5000000

WHERE SoDu > 8000000

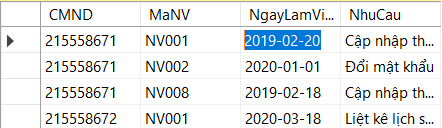
 🡪 

1. **Xóa dòng trong bảng**

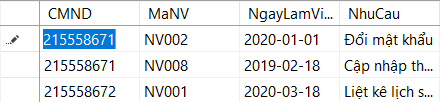
**-** **Xóa đi thông tin dịch vụ trong ngày 20-02-2019**

DELETE FROM XULYNHUCAU

WHERE NgayLamViec = ’20-02-2019’

****

**Sau khi xóa:**

****

1. **Lệnh sửa bảng**

**- Thêm cột mới:** *Thêm vào bảng NHANVIEN cột SDT*

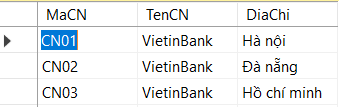


**Sau khi thêm:**

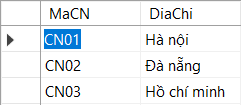
****

**- Xóa cột:** *xóa cột tên chi nhánh trong bảng CHINHANH*

ALTER TABLE CHINHANH DROP TenCN

****

**Sau khi xóa cột:**

****

**- Mở rộng cột:** *Mở rộng cột tên nhân viên trong bảng NHANVIEN*

ALTER TABLE NHANVIEN ALTER COLUMN TenNV NVARCHAR(50)

1. **Lệnh xóa bảng**

*Xóa bảng CHINHANH*

DROP TABLE CHINHANH

## 3.3 Chuẩn hóa quan hệ

* Dạng chuẩn 1 (1NF)

NHANVIEN(MaNV, TenNV,NgayVL, **MaCN**)

CHINHANH(MaCN, TenCN, DiaChi)

KHACHHANG(CMND, TenKH, NgaySinh, GioiTinh, SoDienThoai, DiaChi)

TAIKHOAN(SoTk,TenDangNhap,MatKhau,NgayDangKy,SoThe,SoDu,**CMND**)

XULYNHUCAU(CMND,MaNV,NgayLamViec,NhuCau)

QUANLY(MaNV,SoTk,NgayKT,GhiChu)

GIAODICH(MaGD,LoaiGD)

CHITIETGIAODICH(MaGD,SoTK,SoTKNhan,NgayGD,SoTien,GhiChu)

Vấn đề tồn tại trong 1NF

Trong quan hệ XULYNHUCAU(CMND,MaNV,NgayLamViec,NhuCau)

-Chứa thuộc tính NhuCau là thuộc tính đa trị vì nó có thể chứa nhiều giá trị khác nhau trong mỗi khi thể hiện một thực thể.

Trong quan hệ GIAODICH(MaGD,LoaiGD)

-Chứa thuộc tính LoaiGD là thuộc tính đa trị vì nó có thể chứa nhiều giá trị khác nhau trong mỗi khi thể hiện một thực thể.

* Dạng chuẩn 2 (2NF)

+ Vi phạm chuẩn 2NF trong quan hệ: TAIKHOAN

TAIKHOAN(SoTk,TenDangNhap,MatKhau,NgayDangKy,SoThe,SoDu,**CMND**)

-Thuộc tính NgayDangKy, SoThe ,SoDu chỉ phụ thuộc vào một mình SoTK mà không phụ thuộc hoàn toàn vào cả tập khóa (SoTK,CMND)

* + - * Dạng chuẩn 3 (3NF) 🡪 Đã đạt dạng chuẩn 3NF

# 

# **KẾT LUẬN VÀ KHUYẾN NGHỊ**

## Kết quả đạt được

Trong suốt khoảng thời gian làm bài tập lớn, chúng em đã học thêm được nhiều kiến thức nền tảng, các phương pháp tìm và đọc tài liệu cũng như kiến thức về lĩnh vực chuyên môn về đề tài mà chúng em đang theo đuổi.

## Hiện tại

Vì thời gian thực hiện đề tài cũng như kiến thức của bản thân là có giới hạn nên vẫn còn vài thiếu sót.

## Hướng phát triển

Nếu có thời gian và điều kiện để tiếp tục thực hiện em sẽ cố gắng để hoàn thành ứng dụng của em một cách thật trọn vẹn. Em sẽ phát triển thêm nhiều tính năng ứng dụng.

Mong muốn rằng một ngày nào ứng dụng của mình sẽ ngày một phát triển mở rộng. Và em tin rằng ứng dụng của mình sẽ là một ứng dụng có ích cho xã hội được nhiều người biết đến và tin tưởng khi sử dụng.

## Ưu điểm

Làm việc nhóm hiệu quả, rõ ràng, nhiều lời phản biện tốt bởi các thành viên trong nhóm. Giải quyết công việc được giao nhanh và có sáng tạo trong khi làm việc nhóm. Thái độ tốt và luôn đúng giờ.

## Nhược điểm

Còn mâu thuẫn với nhau trong ý kiến cá nhân, cãi nhau trong lúc làm việc, không chịu lắng nghe ý kiến của một số thành viên trong nhóm.

Chưa tốt trong việc tự tìm hiểu, nghiên cứu.

Còn hơi chậm hiểu vấn đề trong quá trình thực hiện đề tài.

Còn chút sai sót trong quá trình làm word và powpoint.

## Công việc của mỗi thành viên và mức độ hoàn thành

Võ Đoàn Hoàng Long: Bàn luận, lên ý tưởng, tạo bảng SQL (100%).

Nguyễn Thanh Hoà: Bàn luận, chèn dữ liệu (100%).

Bùi Văn Tân: Bàn luận, thực hiện truy vấn, ràng buộc dữ liệu (100%).

Lê Âu Hải: Bàn luận, soạn word, powpoint (100%).

## Tài liệu tham khảo

- Giáo trình môn cơ sở dữ liệu trường Đại học Giao Thông Vận Tải phân hiệu tại TPHCM.

- [https://www.howkteam.vn](https://www.howkteam.vn/course/su-dung-sql-server-31)

- <https://itviec.com>

- https://giasutinhoc.vn

- https://vietjack.com