Logo

Description automatically generated

**TRƯỜNG ĐẠI HỌC NGOẠI NGỮ TIN HỌC THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

🙡🕮🙣

**BÁO CÁO KẾT THÚC HỌC PHẦN HỌC KỲ I**

**NĂM HỌC 2022 – 2023**

**CÁC HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**HỆ THỐNG QUẢN LÝ LƯƠNG NHÂN VIÊN**

Giảng viên hướng dẫn: Trần Thị Thanh Thảo

Sinh viên thực hiện đề tài:

19DH110742 – Nguyễn Anh Khoa

19DH111223 – Hà Bùi Mạnh Hùng

19DH111155 – Đỗ Phúc Tân

Lớp: TT1901

**Tháng 12 Năm 2022**

**MỤC LỤC**

[**Chương 1: Khảo Sát Hiện Trạng Và Xác Định Yêu Cầu** 7](#_Toc122129926)

[**I.** **Khảo Sát Hiện Trạng** 7](#_Toc122129927)

[1. Đối Tượng Quản Lý Chính: 7](#_Toc122129928)

[**II.** **Xác Định Yêu Cầu** 8](#_Toc122129929)

[**III.** **Xác Định Công Nghệ Sử Dụng Trong Đề Tài** 9](#_Toc122129930)

[**Chương 2: Cơ Sở Lý Thuyết** 9](#_Toc122129931)

[**I.** **SQL và SQL SERVER** 9](#_Toc122129932)

[1. SQL Là Gì? 9](#_Toc122129933)

[2. SQL Server Là Gì? 9](#_Toc122129934)

[**II.T-SQL** 10](#_Toc122129935)

[1. T- SQL Là Gì? 10](#_Toc122129936)

[2. Tổng Quan 10](#_Toc122129937)

[**III.** **Quản Lý Giao Tác/Giao Dịch** 11](#_Toc122129938)

[1. Khái Niệm Giao Tác 11](#_Toc122129939)

[2. Các Tính Chất Của Giáo Tác 12](#_Toc122129940)

[**IV.** **Xử Lý Đồng Thời** 13](#_Toc122129941)

[1. Tranh Chấp Đồng Thời Là Gì? 13](#_Toc122129942)

[2. Một Số Vấn Đề Xảy Ra Khi Tranh Chấp Đồng Thời 13](#_Toc122129943)

[3. Xử Lý Tranh Chấp Đồng Thời 14](#_Toc122129944)

[**Chương 3: Phân Tích – Thiết Kế Dữ Liệu** 17](#_Toc122129945)

[**I.** **Lược Đồ Usecase** 17](#_Toc122129946)

[a. Usecase Đăng Nhập 17](#_Toc122129947)

[b. Usecase Nhân Viên 18](#_Toc122129948)

[c. Usecase Trưởng Phòng 19](#_Toc122129949)

[d. Usecase Kế Toán 20](#_Toc122129950)

[**II.** **Phân Tích Dữ Liệu** 21](#_Toc122129951)

[1. Mô Hình Thực Thể Kết Hợp 21](#_Toc122129952)

[*2.* Mô tả các bảng: 25](#_Toc122129953)

[**III.** **Phát Hiện Các Ràng Buộc Toàn Vẹn** 33](#_Toc122129954)

[a. Ràng buộc toàn vẹn trên bảng BANGCONG 33](#_Toc122129955)

[b. Ràng buộc toàn vẹn trên bảng BAOHIEM 33](#_Toc122129956)

[c. Ràng buộc toàn vẹn trên bảng BIENDONGHSL 33](#_Toc122129957)

[d. Ràng buộc toàn vẹn trên bảng BOPHAN 34](#_Toc122129958)

[e. Ràng buộc toàn vẹn trên bảng CHUCVU 34](#_Toc122129959)

[f. Ràng buộc toàn vẹn trên bảng HESOLUONG 34](#_Toc122129960)

[g. Ràng buộc toàn vẹn trên bảng HOPDONG 35](#_Toc122129961)

[h. Ràng buộc toàn vẹn trên bảng KHENTHUONG\_KYLUAT 35](#_Toc122129962)

[i. Ràng buộc toàn vẹn trên bảng LOAICA 35](#_Toc122129963)

[j. Ràng buộc toàn vẹn trên bảng THANGCONG 35](#_Toc122129964)

[k. Ràng buộc toàn vẹn trên bảng THANGCONGCHITIET 36](#_Toc122129965)

[l. Ràng buộc toàn vẹn trên bảng NHANVIEN 36](#_Toc122129966)

[m. Ràng buộc toàn vẹn trên bảng NHANVIEN\_PHUCAP 37](#_Toc122129967)

[n. Ràng buộc toàn vẹn trên bảng PHONGBAN 37](#_Toc122129968)

[o. Ràng buộc toàn vẹn trên bảng PHUCAP 38](#_Toc122129969)

[p. Ràng buộc toàn vẹn trên bảng TANGCA 39](#_Toc122129970)

[q. Ràng buộc toàn vẹn trên bảng TRINHDO 39](#_Toc122129971)

[r. Ràng buộc toàn vẹn trên bảng UNGLUONG 39](#_Toc122129972)

[s. Ràng buộc toàn vẹn trên bảng THANHTOANLUONG 39](#_Toc122129973)

[**Chương 4: Thiết Kế Xử Lý Trong Client/Server** 40](#_Toc122129974)

[**I.** **Giao Tác** 40](#_Toc122129975)

[1.1. Tra Cứu Thông Tin Nhân Viên 40](#_Toc122129976)

[1.2. Tra Cứu Hợp Đồng Của Nhân Viên 41](#_Toc122129977)

[1.3. Tra Cứu Người Quản Lý Của Nhân Viên 41](#_Toc122129978)

[1.4. Thêm Chức Vụ Mới 41](#_Toc122129979)

[1.5. Xem Thông Tin Bảo Hiểm Của Nhân Viên 42](#_Toc122129980)

[1.6. Xem Thông Tin Bộ Phận Và Chức Vụ Của Nhân Viên 42](#_Toc122129981)

[1.7. Tra Cứu Nhân Viên Tăng Ca 42](#_Toc122129982)

[1.8. Thêm Phòng Ban 43](#_Toc122129983)

[1.9. Xóa Phòng Ban 43](#_Toc122129984)

[1.10. Xem Phòng Ban 43](#_Toc122129985)

[1.11. Cập Nhật Thông Tin Bảo Hiểm 44](#_Toc122129986)

[1.12. Tra Cứu Thông Tin Ứng Lương Của Nhân Viên 44](#_Toc122129987)

[1.13. Thêm Vào Bảng Ứng Lương 44](#_Toc122129988)

[1.14. Tra Cứu Thông Tin Công Nhân Viên 45](#_Toc122129989)

[1.15. Tra Cứu Số Lượng Nhân Viên Trong Phòng Ban 45](#_Toc122129990)

[1.16. Xóa HopDong và Xóa Nhân Viên 45](#_Toc122129991)

[1.17. Thêm Nhân Viên (ĐH) 46](#_Toc122129992)

[1.18. Khen Thưởng Nhân Viên (ĐH) 46](#_Toc122129993)

[1.19. Phân Công Tăng Ca (ĐH) 47](#_Toc122129994)

[**II.** **Phân Quyền** 48](#_Toc122129995)

[*a.* Các đối tượng phân quyền 48](#_Toc122129996)

[b. Phân quyền trên các bảng 48](#_Toc122129997)

[a. Phân Quyền Trên Các Giao Tác 49](#_Toc122129998)

[**III.** **Backup** 50](#_Toc122129999)

[**Chương 5: Triển Khai Thực Nghiệm** 51](#_Toc122130000)

[**I.** **Phía Server** 51](#_Toc122130001)

[1. Tạo CSDL 51](#_Toc122130002)

[**II.** **Mô Tả STORE PROCEDURE, FUNCTION, VIEW, TRIGGER** 57](#_Toc122130003)

[1. Mô Tả Function 57](#_Toc122130004)

[2. Mô Tả Store Procedure 58](#_Toc122130005)

[3. Mô tả Trigger 74](#_Toc122130006)

[*4.* Phân Quyền 88](#_Toc122130007)

[**III.** **Phía Client** 90](#_Toc122130008)

[a. Form NHANVIEN 90](#_Toc122130009)

[b. Form HOPDONG 91](#_Toc122130010)

[c. Form CHUCVU 91](#_Toc122130011)

[d. Form BOPHAN 92](#_Toc122130012)

[e. Form PHONGBAN 93](#_Toc122130013)

[f. Form TRINHDO 93](#_Toc122130014)

[**Chương 6: Kết Luận** 94](#_Toc122130015)

[**I.** **Kết Quả Đạt Được** 94](#_Toc122130016)

[1. Phía Server 94](#_Toc122130017)

[2. Phía Client 94](#_Toc122130018)

[**II.** **Kết Quả Chưa Đạt Được** 94](#_Toc122130019)

[**III.** **Tài liệu tham khảo** 95](#_Toc122130020)

[**IV.** **Phân công** 95](#_Toc122130021)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Usecase Đăng nhập 17](#_Toc122096739)

[Hình 2: Usecase Nhân viên 18](#_Toc122096740)

[Hình 3: Usecase Trưởng phòng 19](#_Toc122096741)

[Hình 4: Usecase Kế toán 20](#_Toc122096742)

[Hình 5: Module Nhân viên 1 21](#_Toc122096743)

[Hình 6: Module Nhân viên 2 22](#_Toc122096744)

[Hình 7: Module Thanh toán lương 23](#_Toc122096745)

[Hình 8: Module Hợp đồng 24](#_Toc122096746)

[Hình 9: Module Tháng công chi tiết 24](#_Toc122096747)

[Hình 10: Mô hình quan hệ 25](#_Toc122096748)

[Hình 11: Danh sách nhân viên 90](#_Toc122096749)

[Hình 12: Thêm nhân viên 90](#_Toc122096750)

[Hình 13: Danh sách hợp đồng 91](#_Toc122096751)

[Hình 14: Thêm hợp đồng 91](#_Toc122096752)

[Hình 15: Danh sách chức vụ 91](#_Toc122096753)

[Hình 16: Thêm chức vụ 92](#_Toc122096754)

[Hình 17: Danh sách bộ phận 92](#_Toc122096755)

[Hình 18: Thêm bộ phận 92](#_Toc122096756)

[Hình 19: Danh sách phòng ban 93](#_Toc122096757)

[Hình 20: Thêm phòng ban 93](#_Toc122096758)

[Hình 21: Danh sách trình độ 93](#_Toc122096759)

[Hình 22: Thêm trình độ 94](#_Toc122096760)

**DANH MỤC BẢNG**

[Bảng 1: Bảng hệ số lương 25](#_Toc122098460)

[Bảng 2: Bảng bảo hiểm 26](#_Toc122098461)

[Bảng 3: Bảng biến động hệ só lương 26](#_Toc122098462)

[Bảng 4: Bảng tháng công 27](#_Toc122098463)

[Bảng 5: Bảng chức vụ 27](#_Toc122098464)

[Bảng 6: Bảng trình độ 27](#_Toc122098465)

[Bảng 7: Bảng hợp đồng 28](#_Toc122098466)

[Bảng 8: Bảng nhân viên 28](#_Toc122098467)

[Bảng 9: Bảng phòng ban 29](#_Toc122098468)

[Bảng 10: Bảng ứng lương 29](#_Toc122098469)

[Bảng 11: Bảng nhân viên - phụ cấp 29](#_Toc122098470)

[Bảng 12: Bảng bộ phận 30](#_Toc122098471)

[Bảng 13: Bảng khen thưởng - kỷ luật 30](#_Toc122098472)

[Bảng 14: Bảng loại ca 30](#_Toc122098473)

[Bảng 15: Bảng tháng công chi tiết 31](#_Toc122098474)

[Bảng 16: Bảng phụ cấp 31](#_Toc122098475)

[Bảng 17: Bảng tăng ca 32](#_Toc122098476)

[Bảng 18: Bảng thanh toán lương 32](#_Toc122098477)

[Bảng 19: Bảng phân quyền trên các bảng 48](#_Toc122098478)

[Bảng 20: Bảng phân quyền trên các giao tác 49](#_Toc122098479)

**LỜI MỞ ĐẦU**

Hiện nay việc quản lý nhân viên và lương nhân viên đang được rất nhiều các doanh nghiệp/công ty quan tâm. Họ cần một cơ sở dữ liệu hoàn thiện và ổn định để lưu và xử lý khối lượng thông tin khổng lồ này. Hiểu được nhu cầu của các doanh nghiệp/công ty, nhóm chúng em đã chọn đề tài xây dựng hệ thống quản lý lương nhân viên để làm đồ án môn Các hệ quản trị cơ sở dữ liệu.

# **Chương 1: Khảo Sát Hiện Trạng Và Xác Định Yêu Cầu**

## **Khảo Sát Hiện Trạng**

Khi khảo sát qui trình hoạt động của quá trình quản lý nhân viên và lương nhân viên của một công ty, ta nắm được các thông tin sau:

### **Đối Tượng Quản Lý Chính**:

Thông tin nhân viên, chấm công, lương – thưởng – phạt

#### Quản lý thông tin nhân viên:

* Phòng nhân sự có trách nhiệm thực hiên chức năng này thông qua việc lập và điều chỉnh hồ sơ nhân viên.
* Khi công ty có nhu cầu và đã thực hiện tuyển dụng để đưa ra quyết định tuyển dụng một nhân viên mới từ ban giám đốc, phòng nhân sự sẽ lập danh sách và cập nhật hồ sơ cá nhân của nhân viên mới này.
* Trong quá trình hoạt động, phòng nhân sự sẽ xem thông tin của nhân viên trong công ty để có thể cập nhật các thông tin mới về nhân viên; đồng thời để phát hiện sai sót để tiến hành thay thế, sửa đổi thông tin nhân viên. Việc sửa đổi này được dựa trên cơ sở lấy thông tin cung cấp lại từ phía nhân viên. Tất cả thông tin điều chỉnh sẽ được lưu lại hồ sơ nhân viên làm cơ sở cho kiểm tra của ban giám đốc.
* Đồng thời nếu ban giám đốc yêu cầu xem hoặc điều chỉnh thông tin, phòng nhân sự cần phải lấy thông tin từ hồ sơ nhân viên để báo cáo cho ban giám đốc xem hoặc để thực hiện điều chỉnh cho phù hợp yêu cầu.
* Với nhiệm vụ theo dõi hồ sơ nhân viên nên khi giám đốc có quyết định cho một nhân viên trong công ty thay đổi mức lương, phòng nhân sự sẽ tiến hành điều chỉnh mức lương của nhân viên đó, các thông tin về điều chỉnh này được ghi vào hồ sơ nhân viên để làm cơ sở cho việc tính lương của phòng kế toán.
* Một nhân viên trong công ty thôi không làm việc, sau khi đã quyết định thôi việc từ ban giám đốc, phòng nhân sự sẽ tiến hành hủy bỏ thông tin nhân viên đó.

#### Quản lý chấm công

* Trưởng phòng có trách nhiệm thực hiện hoạt động này với chức năng cơ bản là cập nhật và điều chỉnh chấm công.
* Hằng ngày, Trưởng phòng cập nhật thông tin ngày công làm việc của nhân viên để ghi vào bảng theo dõi chấm công. Bảng chấm công này được nhân viên kiểm tra với thực tế ngày làm việc của mình.
* Khi công ty có nhu cầu tăng cường nhân lực làm thêm cho công việc nào đó, ban giám đốc sẽ ra quyết định cho Trưởng phòng điều động nhân viên làm thêm giờ. Việc điều động của Trưởng phòng phải dựa trên cơ sở cập nhật thông tin về ngày làm thực tế của nhân viên trong tháng để bố trí, sắp xếp phù hợp.
* Khi nhân viên đã làm thêm giờ, Trưởng phòng sẽ cập nhật tiếp ngày công làm thêm vào bảng chấm công, thông tin này vẫn được lấy từ phía nhân viên.
* Bảng theo dõi công sẽ được có sự đối chiếu với nhân viên.
* Cuối tháng kế toán sẽ dựa trên bảng theo dõi công để kiểm tra và để lập ra báo cáo chấm công để trình bày lên ban giám đốc kí duyệt.

#### Quản lý lương thưởng

* Trách nhiệm thực hiện chính của hoạt động này là phòng kế toán với chức năng cơ bản là tính và lập các báo cáo lương, thưởng.
* Cuối tháng, phòng kế toán sẽ nhận báo cáo chấm công đã được ban giám đốc kí duyệt của Trưởng Phòng, đồng thời cập nhật thông tin về mức thưởng và các khoản phụ cấp và các khoản BHYT, BHXH…để làm cơ sở tính lương.
* Dựa vào các thông tin trên, phòng kế toán sẽ lập ra phiếu lương cho từng nhân viên với đầy đủ mức lương, thưởng và khấu trừ các khoản phải nộp.
* Sau khi lập phiếu lương, kế toán sẽ thống kê để lên bảng lương - là mức lương thực hưởng của mỗi nhân viên để trình lên giám đốc kí duyệt.
* Sau khi giám đốc kí duyệt, nhân viên sẽ nhận lương: kế toán sẽ phải lập các Báo cáo về lương, báo cáo về thuế thu nhập, BHYT, BHXH phải nộp hàng tháng.
* Các báo cáo này sẽ được trình cho ban giám đốc xem để nắm bắt tình hình lương thưởng…của công ty; đồng thời. phải trình cho các cơ quan chức năng kiểm tra.

## **Xác Định Yêu Cầu**

+ Thiết lập hệ thống:

Một hệ thống quản lý lương nhân viền cần đầy đủ thông tin về nhân sự, khi cần xuất thông tin hoặc tra cứu thông tin của bất cứ nhân sự nào thì có thể có ngay trong thời gian ngắn nhất.

Thông tin quản lý nhân sự có thể dựa vào cách xây dựng và quản lý dữ liệu thông qua việc triển khai sau:

* Xây dựng bảng ghi chú thông tin nhân viên cụ thể (Họ tên, năm sinh, quê quán, chức danh, ... các thông tin thuộc cá nhân nhân sự)
* Bảng hợp đồng – tuyển dụng nhằm lưu giữ thông tin những nhân viên được tuyển chọn, đáp ứng được yêu cầu tuyển dụng.
* Bảng kết quả đánh giá thực hiện công việc (chấm công, khen thưởng – kỷ luật), lưu giữ quá trình hoàn thành công việc của nhân sự, từ đó có những chiến lược nhân sự phù hợp như: đào tạo, cơ cấu lại nhân sự, tăng hoặc trừ lương.
* Bảng theo dõi lương: nắm bắt được mức lương nhân viên đang được hưởng và có thể điều chuyển mức lương phù hợp cho từng vị trí
* Xây dựng các bảng ghi chú các thông tin phòng ban, ứng lương, tăng ca…

+ Xây dựng các thủ tục thêm, xóa, sửa và các trigger, function hỗ trợ tính toán lương, chấm công…

Để quản lý lương nhân viên hiệu quả, cần phải xây dựng một quy trình tính toán lương, chấm công… một các hiệu quả. Việc xây dựng này phụ thuộc vào mô hình của mỗi công ty. Các điều cần lưu ý ở đây:

* Công thức tính lương và hệ số lương
* Thuế thu nhập cá nhân
* Công nhân viên
* Các chế độ đãi ngộ, phụ cấp, lương thưởng
* Những nội quy, quy định cũng như chế độ kỹ luật tại công ty

## **Xác Định Công Nghệ Sử Dụng Trong Đề Tài**

+ Công nghệ sử dụng: Sql Server, Winform

+ Tính năng, nhu cầu sử dụng:

* SQL SERVER:
* Chức năng tạo bảng, thiết kế CSDL
* Tạo ràng buộc, tạo các function
* Tạo các thủ tục, trigger
* Tạo Transaction - xử lý đồng thời
* Tạo user, phân quyền theo role
* Backup
* Winform:
* Kết nối với CSDL
* Thiết kế giao diện
* Hiện thực hóa chức năng

# **Chương 2: Cơ Sở Lý Thuyết**

## **SQL và SQL SERVER**

### SQL Là Gì?

SQL là viết tắt của Structured Query Language là ngôn ngữ truy vấn có cấu trúc. Nó là một ngôn ngữ, là tập hợp các lệnh để tương tác với cơ sở dữ liệu. Dùng để lưu trữ, thao tác và truy xuất dữ liệu được lưu trữ trong một cơ sở dữ liệu quan hệ. Trong thực tế, SQL là ngôn ngữ chuẩn được sử dụng hầu hết cho hệ cơ sở dữ liệu quan hệ. Tất cả các hệ thống quản lý cơ sở dữ liệu quan hệ (RDMS) như MySQL, MS Access, Oracle, Postgres và SQL Server… đều sử dụng SQL làm ngôn ngữ cơ sở dữ liệu chuẩn.

### SQL Server Là Gì?

SQL server hay còn được gọi là Microsoft SQL Server, nó từ viết tắt của MS SQL Server. Đây chính là một loại phần mềm đã được phát triển bởi Microsoft và nó được sử dụng để có thể dễ dàng lưu trữ cho những dữ liệu dựa theo tiêu chuẩn RDBMS.

Từ đó, người ta sẽ lưu trữ dữ liệu dựa vào tiêu chuẩn RDBMS và nó cũng là một trong những hệ quản trị cơ sở dữ liệu dạng quan hệ đối tượng.

SQL Server có khả năng cung cấp đầy đủ các công cụ cho việc quản lý từ giao diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL. Điểm mạnh của SQL điểm mạnh của nó là có nhiều nền tảng được kết hợp cùng như: ASP.NET, C# để xây dựng Winform cũng chính nó có khả năng hoạt động độc lập.

Tuy nhiên, SQL Server thường đi kèm với việc thực hiện riêng các ngôn ngữ SQL, T-SQL, ...

## **II.T-SQL**

### T- SQL Là Gì?

T-SQL là viết tắt của Transact-SQL, đây là một phần mở rộng của ngôn ngữ truy vấn SQL được phát triển dựa theo tiêu chuẩn ISO và ANSI (American National Standards Institute).

T-SQL là việc thực thi ngôn ngữ cấu trúc chuẩn của Microsoft được dùng trong hệ thống SQL Server. Ngôn ngữ đặc biệt này có thể được dùng để thực hiện các thao tác với bảng như định nghĩa, chèn, xóa, cập nhật, truy vấn dữ liệu.

Đây là một ngôn ngữ truy vấn cơ sở dữ liệu khá mạnh đề cập đến kiểu dữ liệu, đối tượng tạm thời, thủ tục hệ thống và các thủ tục mở rộng. T-SQL còn có tính năng như sử dụng biến địa phương, xử lý các chuỗi/dữ liệu được thêm vào.

### Tổng Quan

Khối câu lệnh trong T-SQL sẽ được chia làm 3 nhóm tương đương với 3 chức năng chính như sau:

+ Nhóm định nghĩa dữ liệu: Nhóm này gồm các câu lệnh CREATE, ALTER, DROP table, dùng để tạo, sửa đổi hoặc xóa bảng dữ liệu. Các lệnh này được sử dụng để định nghĩa cấu trúc một table.

+ Nhóm điều khiển dữ liệu: Gồm các lệnh GRANT, DENY, REVOKE,…dùng để phần quyền truy cập của các đối tượng table, function, view,…

+ Nhóm thao tác dữ liệu: Là tập hợp các lệnh để thao tác trực tiếp với dữ liệu có trong table, giúp truy vấn, tìm kiếm dữ liệu nhanh chóng. Các câu lệnh nhóm này gồm INSERT, UPDATE, SELECT, DELETE.

Cấu trúc của ngôn ngữ T-SQL được xây dựng theo từng khối lệnh được bắt đầu bằng BEGIN và kết thúc là END. Một khối lệnh có thể được đặt lồng trong một khối lệnh bao trùm khác, trong mỗi khối lệnh là các câu lệnh, các câu lệnh này được ngăn cách với nhau bởi dấu chấm phẩy.

Cấu trúc một khối lệnh có dạng như sau:

A picture containing text

Description automatically generated

*Cấu trúc khối lệnh T-SQL*

## **III. Quản Lý Giao Tác/Giao Dịch**

### Khái Niệm Giao Tác

Một giao tác (transaction) là một hành động hay một chuỗi các hành động truy cập vào CSDL hoặc làm thay đổi nội dung CSDL, giao tác được đưa ra bởi một người sử dụng hay một chương trình ứng dụng

Như vậy một giao tác là một đơn vị công việc trên CSDL xét trên tính logic. Một giao tác có thể là toàn bộ một chương trình (viết trong ngôn ngữ thao tác dữ liệu bậc cao hay trong ngôn ngữ lập trình như SQL, COBOL, ...), nó cũng có thể là một phần của chương trình hay một câu lệnh đơn lẻ như câu lệnh INSERT, UPDATE của SQL.

Một số thuật ngữ liên quan đến giao tác:

+ Begin [transaction/tran]: bắt đầu một transaction

+ Commit [transaction/tran]: hoàn tất một transaction

+ Rollback [transaction/tran] : quay lui, hủy bỏ toàn bộ phần giao dịchđã thực hiện trước đó

Trong ngữ cảnh CSDL, có thể xem sự thực hiện một chương trình ứng dụng như một dãy các giao tác và xen kẽ giữa chúng là những thao tác không đòi hỏi xử lí CSDL (thao tác CPU). Để cho việc tiện theo dõi, dưới đây minh họa hai lược đồ quan hệ Products và Categories đơn giản như sau:

|  |
| --- |
| Products(id, category\_id, name, price)  Categories(id, name) |

Giải thích:

Lược đồ Products gồm các thuộc tính về mã sản phẩm (id), mã thể loại (category\_id), tên sản phẩm (name) và giá của sản phẩm (price)

Lược đồ Categories dùng để tham chiếu mã thể loại ứng với từng sản phẩm bao gồm các thuộc tính về mã thể loại (id), tên thể loại (name)

Ví dụ 1: Một giao tác đơn giản là nâng giá của một sản phẩm có mã số là xx thêm 10$%$ giá hiện tại của sản phẩm đó. Chúng ta có thể viết giao tác này như sau:

|  |
| --- |
| read(product.id = x, price)  new\_price = price \* 1.1  write(product.id = x, new\_price) |

Ở đây chúng ta quy ước dùng read(x) để chỉ thao tác đọc một mục dữ liệu có tên là xx ở CSDL vào một biến của chương trình. Để cho đơn giản việc trình bày chúng ta coi như biến chương trình này cũng có tên là xx. Tương tự như vậy write(x) dùng để chỉ thao tác ghi giá trị biến xx của chương trình vào mục dữ liệu xx của CSDL

Việc thưc hiện một lệnh read(x) sẽ bao gồm các bước sau:

+ Tìm địa chỉ của khối có chứa mục dữ liệu xx

+ Sao chép khối đó vào vùng đệm của bộ nhớ chính (nếu nó chưa có sẵn trong một vùng đệm của bộ nhớ chính)

+ Sao chép mục dữ liệu xx từ vùng đệm vào biến có tên là xx của chương trình

Việc thưc hiện một lệnh write(x) sẽ bao gồm các bước sau:

+ Tìm địa chỉ của khối có chứa mục dữ liệu xx

+ Sao chép khối đó vào vùng đệm của bộ nhớ chính (nếu nó chưa có sẵn trong một vùng đệm của bộ nhớ chính)

+ Sao chép mục dữ liệu xx từ biến chương trình có tên là xx vào chỗ chính xác của nó trong vùng đệm

+ Lưu trữ khối vừa cập nhập trong vùng đệm vào lại thiết bị nhớ để sau này có thể truy xuất (có thể lưu ngay tức thời hoặc tại thời điểm muộn hơn)

Trong ví dụ trên lệnh read(product.id = x, price) dùng để chỉ thao tác đọc dữ liệu ở thuộc tính price của bản ghi có giá trị khóa chính id=x, lệnh write(product.id = x, new\_price) dùng để chỉ giao tác ghi lại giá trị của new\_price vào cho bản ghi có khóa chính là x. Giữa hai thao tác đọc và ghi dữ liệu nới trên có một thao tác không truy cập CSDL, đó là thao tác new\_price = price \* 1.1

### Các Tính Chất Của Giáo Tác

Để đảm bảo tính nhất quán của dữ liệu, các hệ QTCSDL cần đảm bảo các tính chất sau đây cho giao tác, tính chất ACID

+ Tính nguyên tố (Atomicity): toàn bộ các thao tác trong cùng một giao tác đều được thực hiện hoặc không một thao tác nào trong chúng được thực hiện. Một giao tác là một đơn vị công việc không thể phân chia

+ Tính nhất quán (Consistency): một giao tác phải chuyển CSDL từ một trạng thái nhất quán này sang một trạng thái nhất quán khác

+ Tính cô lập (Isolation): Các giao tác phải được thực hiện một cách độc lập với nhau. Nói một cách khác những tác động của giao tác này sẽ không thể thấy được đối với những giao tác khác, khi giao tác này chưa thực hiện xong hoàn toàn

+ Tính bền vững (Durability): những thay đổi của CSDL do tác động của một giao tác thành công là bền vững, không bị mất đi

## **Xử Lý Đồng Thời**

### Tranh Chấp Đồng Thời Là Gì?

Khi nhiều giao tác cùng thực hiện truy xuất trên một đơn vị dữ liệu tại một thời điểm để thực hiện các thao tác đọc, ghi lên đơn vị dữ liệu đó 🡺 đụng độ 🡺 tranh chấp đồng thời.

### Một Số Vấn Đề Xảy Ra Khi Tranh Chấp Đồng Thời

Trong môi trường truy xuất đồng thời, có thể xảy ra một số vấn đề như sau:

+ Mất dữ liệu cập nhật (Lost update)

* Tình trạng này xảy ra khi có nhiều hơn một giao tác cùng thực hiện cập nhật trên 1 đơn vị dữ liệu. Khi đó, tác dụng của giao tác cập nhật thực hiện sau sẽ đè lên tác dụng của thao tác cập nhật trước.

+ Đọc dữ liệu chưa commit (Uncommitted data, Dirty read)

* Xảy ra khi một giao tác thực hiện đọc trên một đơn vị dữ liệu mà đơn vị dữ liệu này đang bị cập nhật bởi một giao tác khác nhưng việc cập nhật chưa được xác nhận.

+ Giao tác đọc không thể lặp lại (Unrepeatable data)

* Tình trạng này xảy ra khi một giao tác T1 vừa thực hiện xong thao tác đọc trên một đơn vị dữ liệu (nhưng chưa commit) thì giao tác khác (T2) lại thay đổi (ghi) trên đơn vị dữ liệu này. Điều này làm cho lần đọc sau đó của T1 không còn nhìn thấy dữ liệu ban đầu nữa.

+ Bóng ma (Phantom)

* Là tình trạng mà một giao tác đang thao tác trên một tập dữ liệu nhưng giao tác khác lại chèn thêm các dòng dữ liệu vào tập dữ liệu mà giao tác kia quan tâm.

### Xử Lý Tranh Chấp Đồng Thời

+ Dùng mức cô lập:

Isolation level là một thuộc tính của transaction, qui định mức độ cô lập của dữ liệu mà transaction có thể truy nhập vào khi dữ liệu đó đang được cập nhật bởi một transaction khác. SQL Server cung cấp các mức isolation level sau xếp theo thứ tự tăng dần của mức độ cô lập của dữ liệu: Read Uncommitted, Read Commited, Repeatable Read, và Serializable.

* Read Uncommitted:

Khi transaction thực hiện ở mức này, các truy vấn vẫn có thể truy nhập vào các bản ghi đang được cập nhật bởi một transaction khác và nhận được dữ liệu tại thời điểm đó mặc dù dữ liệu đó chưa được commit (uncommited data). Nếu vì lý do nào đó transaction ban đầu rollback lại những cập nhật, dữ liệu sẽ trở lại giá trị cũ. Khi đó transaction thứ hai nhận được dữ liệu sai.

* Read Commited:

Đây là mức isolation mặc định, nếu bạn không đặt gì cả thì transaction sẽ hoạt động ở mức này. Transaction sẽ không đọc được dữ liệu đang được cập nhật mà phải đợi đến khi việc cập nhật thực hiện xong. Vì thế nó tránh được dirty read như ở mức trên.

* Repeatable Read:

Mức isolation này hoạt động như mức read commit nhưng nâng thêm một nấc nữa bằng cách ngăn không cho transaction ghi vào dữ liệu đang được đọc bởi một transaction khác cho đến khi transaction khác đó hoàn tất.

* Serializable:

Mức isolation này tăng thêm một cấp nữa và khóa toàn bộ dải các bản ghi có thể bị ảnh hưởng bởi một transaction khác, dù là UPDATE/DELETE bản ghi đã có hay INSERT bản ghi mới.

+ Dùng khóa trực tiếp trong câu lệnh:

1. Kỹ Thuật Khóa

* Khóa đọc (Shared Lock - Slock):

Giao tác được phép Đọc dữ liệu, nhưng không được phép ghi, nhiều giao tác có thể đồng thời giữu Slock trên cùng 1 đơn vị dữ liệu.

* Khóa ghi ( Exclusive Lock – Xlock) :

Giao tác giữ Xlock được phép Ghi + Đọc dữ liệu,

Tại 1 thời điểm chỉ có tối đa 1 giao tác được quyền giữ Xlock trên 1 đơn vị dữ liệu

Không thể thiết lập Slock trên đơn vị dữ liệu có dạng Xlock

* Khóa dự định ghi (Updat Lock – Ulock):

Ulock sử dụng khi đọc dữ liệu với dự định ghi trở lại trên dữ liệu này.

Ulock là chế độ khóa trung gian giữa Slock và Xlock

Khi thực hiện giao tác ghi trên dữ liệu thì bắt buộc phải tự động chuyển thành Xlock

Giao tác giữ Ulock được phép ghi + đọc dữ liệu

Tại 1 thời điểm chỉ có tối đa 1 giao tác được quyền giữ Ulock trên 1 đơn vị dữ liệu

Có thể thiết lập Slock trên đơn vị dữ liệu có dạng Ulock

1. Dùng khóa trực tiếp trong câu lệnh:

+ Mức cô lập quyết định cách phát và giữ khóa S trong một transaction và có hiệu lực trên tất cả các thao tác đọc trong transactionđó.

+ Thực tế, ta cần phát và giữ khóa Slocktheo các cách khác nhau cho các thao tác đọc khác nhau trong cùng một transaction

+ Cú Pháp:

Graphical user interface, text, application

Description automatically generated

+ Các chế độ khóa trực tiếp

|  |  |  |
| --- | --- | --- |
| 1 | Readuncommited/Nolock | Không thiết lập Shared Lock khi đọc |
| 2 | Readcommited | + Đây là chế độ mặt định |
|  |  | + Chỉ đọc dữ liệu đã commit  + Thiết lập Shared Lock trên đơn vị dữ liệu cần đọc và mở Lock ra ngay sau khi đọc xong |
| 3 | Repeatableread | + Thiết lập shared lock khi select và dữ shared lock cho tới khi hết giao tác |
| 4 | Serializable/Holdlock | + Thiết lập shared lock khi đọc và dữ shared lock cho tới khi hết giao tác |
| 5 | UPDLOCK | + Sử dụng update lock thay vì shared lock |
| 6 | Xlock | + Khóa độc quyền |
| 7 | Readpast | + Chỉ có thể sử dụng trong lệnh select và chỉ áp dụng trên khóa của dòng dữ liệu. Những dòng bị khóa sẽ được bỏ qua. |
| 8 | Rowlock | + Khóa chỉ dùng những dòng cần thao tác |
| 9 | Tablock | + Khóa toàn bộ bảng trong CSDL  + Các thao tác cập nhật của những giao tác khác không thể thực hiện trên bảng này trong khi khóa vẫn đang được giữ |
| 10 | Tablockx | + Xlock + Tablock |

# **Chương 3: Phân Tích – Thiết Kế Dữ Liệu**

## **Lược Đồ Usecase**

### Usecase Đăng Nhập

Diagram

Description automatically generated

Hình 1: Usecase Đăng nhập

### Usecase Nhân Viên

**Diagram

Description automatically generated**

Hình 2: Usecase Nhân viên

### Usecase Trưởng Phòng

**Diagram

Description automatically generated**

Hình 3: Usecase Trưởng phòng

### Usecase Kế Toán

Diagram

Description automatically generated

Hình 4: Usecase Kế toán

## **Phân Tích Dữ Liệu**

### Mô Hình Thực Thể Kết Hợp

#### Module Nhân Viên

* Module Nhân Viên 1

Diagram

Description automatically generated

Hình 5: Module Nhân viên 1

Mô tả: Nhân viên có một trình độ và một chức vụ, làm việc tại một Phòng Ban, thuộc một bộ phân được quản lý bởi một trưởng phòng.

* Module Nhân Viên 2

Diagram

Description automatically generated

Hình 6: Module Nhân viên 2

Mô tả: mỗi nhân viên sẽ nhận được một mức phụ cấp khác nhau, một nhân viên có thể làm tăng ca với một loại ca, nhân viên có thể ứng lương 1 lần mỗi tháng, mỗi nhân viên có thể được khen thưởng hoặc bị kỷ luật nhiều lần và mỗi nhân viên có 1 bảo hiểm.

* Module Thanh Toán Lương

Diagram

Description automatically generated

Hình 7: Module Thanh toán lương

Mô tả: mỗi tháng nhân viên sẽ được thanh toán lương một lần, trong thanh toán lương sẽ trừ ứng lương (nếu có) và cộng tiền phụ cấp của nhân viên (nếu có).

* Module Hợp Đồng

Diagram

Description automatically generated

Hình 8: Module Hợp đồng

Mô tả: nhân viên có thể có nhiều hợp đồng, mỗi hợp đồng sẽ có một hệ số lương nhất định và mỗi hệ số lượng có thế biến động.

* Module Tháng Công Chi Tiết

Diagram

Description automatically generated

Hình 9: Module Tháng công chi tiết

Mô tả: mỗi tháng công sẽ có một bảng tháng công chi tiết và mỗi bảng tháng công chi tiết sẽ thuộc về một nhân viên.

#### Mô Hình Quan Hệ

Diagram, schematic

Description automatically generated

Hình 10: Mô hình quan hệ

### Mô tả các bảng:

#### Bảng HESOLUONG

Bảng 1: Bảng hệ số lương

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **MaHSL** | Char (4) | Mã hệ số lương  **Primary key** |
| 2 | Bac | Int | Bậc lương |
| 3 | NhomNgach | Char (20) | Nhóm ngạch |
| 4 | HeSoLuong | Float | Hệ số lương |

#### Bảng BAOHIEM

Bảng 2: Bảng bảo hiểm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDBH** | number (10) | Mã số bảo hiểm  **Primary key** |
| 2 | SOBH | Nvarchar (50) | Số bảo hiểm của nhân viên |
| 3 | NGAYCAP | Datetime | Ngày cấp bảo hiểm |
| 4 | NOICAP | Nvarchar (50) | Nơi cấp bảo hiểm |
| 5 | NOIDK | Nvarchar (50) | Nơi đăng ký |
| 6 | **MANV** | Char (5) | Mã nhân viên  **Foreign key** |

#### Bảng BIENDONGHSL

Bảng 3: Bảng biến động hệ só lương

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **MaHSL** | Char (4) | Mã hệ số lương  **Primary key,**  **Foreign key** |
| 2 | NAMAPDUNG | Date | Năm áp dụng |
| 3 | Bac | Int | Bậc lương |
| 4 | NhomNgach | Char (20) | Nhóm ngạch |
| 5 | HeSoLuong | Float | Hệ số lương |

#### Bảng THANGCONG

Bảng 4: Bảng tháng công

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **ID** | Int | Mã ID  **Primary key** |
| 2 | **MATHANGCONG** | Int | Mã Tháng Công  **Foreign key** |
| 3 | Thang | Int | Tháng theo dõi |
| 4 | Nam | Int | Nămtheo dõi |
| 5 | Khoa | Bit | Khóa |
| 6 | NGAYTINHCONG | Datetime | Ngày tính công |
| 7 | NAYCONGTRONG  THANG | Float | Ngà công trong tháng |
| 8 | TRANGTHAI | Bit | Trạng thái |

#### Bảng CHUCVU

Bảng 5: Bảng chức vụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDCV** | Int | Mã chức vụ  **Primary key** |
| 2 | TenCV | Nvarchar (50) | Tên chức vụ |

#### Bảng TRINHDO

Bảng 6: Bảng trình độ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDTD** | Int | Mã trình độ  **Primary key** |
| 2 | TENTD | Nvarchar (50) | Tên trình độ |

#### Bảng HOPDONG

Bảng 7: Bảng hợp đồng

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **SOHD** | Nvarchar (50) | Mã hợp đồng  **Primary key** |
| 2 | NGAYKY | Datetime | Ngày ký |
| 3 | NGAYBATDAU | Datetime | Ngày bắt đầu |
| 4 | NGAYKETTHUC | Datetime | Ngày kết thúc |
| 5 | NOIDUNG | Nvarchar (max) | Nội dung |
| 6 | LANKY | Int | Lần ký |
| 7 | HANHD | Int | Hạn hạp đồng |
| 8 | **MAHSL** | Varchar (4) | Mã hệ số lương  **Foreign Key** |
| 9 | **MANV** | Char (5) | Mã nhân viên  **Foreign Key** |

#### Bảng NHANVIEN

Bảng 8: Bảng nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **MaNV** | Char (5) | Mã nhân viên  **Primary Key** |
| 2 | HOTEN | Nvarchar (50) | Họ tên |
| 3 | GIOITINH | Bit | Giới tính |
| 4 | NGAYSINH | Datetime | Ngày sinh |
| 5 | DIENTHOAI | Nvarchar (50) | Điện thoại |
| 6 | CCCD | Nvarchar (50) | Căn cước công dân |
| 7 | DIACHI | Nvarchar (500) | Địa chỉ |
| 8 | HINHANH | Varbinary (Max) | Hình ảnh |
| 9 | **IDPB** | Int | Mã phòng ban  **Foreign key** |
| 10 | **IDBP** | Int | Mã bộ phận  **Foreign key** |
| 11 | **IDCV** | Int | Mã chức vụ  **Foreign key** |
| 12 | **IDTD** | Int | Mã trình độ  **Foreign key** |
| 13 | **MANQL** | Char (5) | Mã người quản lý  **Foreign key** |

#### Bảng PHONGBAN

Bảng 9: Bảng phòng ban

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDPB** | Int | Mã phòng ban  **Primary key** |
| 2 | TENPB | Nvarchar (50) | Tên phòng ban |
| 3 | TruongPhong | Char (5) | Trưởng phòng |

#### Bảng UNGLUONG

Bảng 10: Bảng ứng lương

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **ID** | Int | Mã ứng lương  **Primary Key** |
| 2 | NAM | Int | Năm |
| 3 | THANG | Int | Tháng |
| 4 | NGAY | Int | Ngày |
| 5 | SOTIEN | Float | Số tiền ứng |
| 6 | TRANGTHAI | Bit | Trạng thái |
| 7 | **MANV** | Char (5) | Mã nhân viên  **Foreign key** |

#### Bảng NHANVIEN\_PHUCAP

Bảng 11: Bảng nhân viên - phụ cấp

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **ID** | Int | Mã phụ cấp  **Primary key** |
| 2 | **MaNV** | Char (5) | Mã nhân viên  **Foreign key** |
| 3 | **IDPC** | Int | Mã phụ cấp  **Foreign key** |
| 4 | NGAY | Datetime | Ngày phụ cấp |
| 5 | NOIDUNG | Nvarchar (500) | Nội dung phụ cấp |
| 6 | SOTIEN | Float | Số tiền phụ cấp |

#### Bảng BOPHAN

Bảng 12: Bảng bộ phận

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDBP** | Int | Mã bộ phận  **Primary key** |
| 2 | TENBP | Nvarchar (50) | Tên bộ phận |

#### Bảng KHENTHUONG-KYLUAT

Bảng 13: Bảng khen thưởng - kỷ luật

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **ID** | Int | Mã khen thưởng kỷ luật  **Primary key** |
| 2 | SOKTKL | Int | Số khen thưởng kỷ luật |
| 3 | NOIDUNG | Nvarchar (500) | Nội dung |
| 4 | NGAY | Datetime | Ngày khen thưởng-kỷ luật |
| 5 | LOAI | Int | Loại |
| 6 | **MANV** | Char (5) | Mã nhân viên  **Foreign key** |

#### Bảng LOAICA

Bảng 14: Bảng loại ca

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDLC** | Int | Mã loại ca  **Primary key** |
| 2 | TENLC | Nvarchar (50) | Tên loại ca |
| 3 | HESO | Float | Hệ số |

#### Bảng THANGCONGCHITIET

Bảng 15: Bảng tháng công chi tiết

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **MATHANGCONG** | Int | Mã tháng công  **Primary key** |
| 2 | **MANV** | Char (5) | Mã nhân viên  **Foreign key** |
| 3 | N1 | Nvarchar(10) | Ngày 1 |
| … | … | … | … |
| 34 | N31 | Nvarchar(10) | Ngày 31 |
| 35 | NGAYCONG | Float | Ngày công |
| 36 | NGAYPHEP | Float | Ngày nghỉ có phép |
| 37 | NGHIKHONGPHEP | Float | Ngày nghỉ không phép |
| 38 | TONGNGAYCONG | Float | Tổng ngày công |

#### Bảng PHUCAP

Bảng 16: Bảng phụ cấp

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDPC** | Int | Mã phụ cấp  **Primary key** |
| 2 | TENPC | Nvarchar (50) | Tên phụ cấp |
| 3 | SOTIEN | Float | Số tiền |

#### Bảng TANGCA

Bảng 17: Bảng tăng ca

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **ID** | Int | Mã tang ca  **Primary key** |
| 2 | NAM | Int | Năm |
| 3 | THANG | Int | Tháng |
| 4 | NGAY | Int | Ngày |
| 5 | SOGIO | Float | Số giờ |
| 6 | **MANV** | Char (5) | Mã nhân viên  **Foreign key** |
| 7 | **IDLC** | Int | Mã loại ca  **Foreign key** |

#### Bảng THANHTOANLUONG

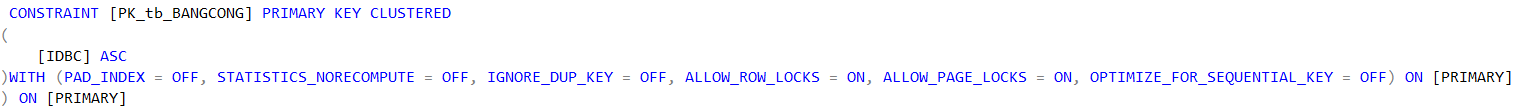
Bảng 18: Bảng thanh toán lương

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | **IDLUONG** | Int | Mã id lương  **Primary key** |
| 2 | THANGTK | Int | Tháng thống kê |
| 3 | NAMTK | Int | Năm thống kê |
| 4 | **MANV** | Char(5) | Mã nhân viên  **Foreign key** |
| 5 | **MACC** | int | Mã chấm công  **Foreign key** |
| 6 | LUONGCT | Float | Lương chưa thuế |
| 7 | THUETNCN | Float | Thuế thu nhập cá nhân |
| 8 | LUONG | Float | Lương đã tính thuế |
| 9 | **ID** | Int | Id ứng lương  **Foreign key** |
| 10 | **IDPC** | Int | Id phụ cấp  **Foreign key** |

## **Phát Hiện Các Ràng Buộc Toàn Vẹn**

### Ràng buộc toàn vẹn trên bảng BANGCONG

Ràng buộc khóa chính tên là PK\_tb\_BANGCONG trên cột IDBC



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_BANGCONG\_tb\_LOAICONG trên cột **IDLC** tham chiếu đến cột **IDLC** của bảng LOAICONG



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_BANGCONG\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV** của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng BAOHIEM

Ràng buộc khóa chính tên là Pk\_tb\_BAOHIEM trên cột IDBH

Text

Description automatically generated with low confidence

Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_BAOHIEM\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV** của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng BIENDONGHSL

Ràng buộc khóa chính tên là PK\_tb\_BIENDONGHSL trên cột MAHSL

Logo

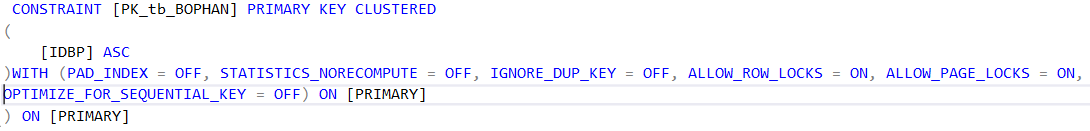
Description automatically generated with medium confidence

Khai báo 1 ràng buộc khóa ngoại tên là FK\_BDHSL\_HSLtrên cột **MAHSL** tham chiếu đến cột **MaHeSoLuong** của bảng BANGHESOLUONG



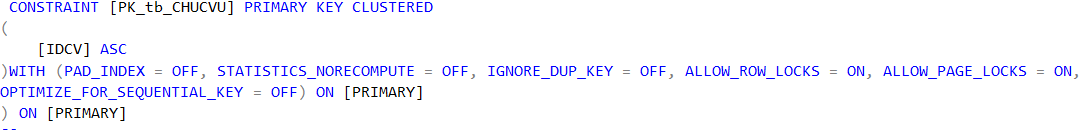
### Ràng buộc toàn vẹn trên bảng BOPHAN

Ràng buộc khóa chính tên là PK\_tb\_BOPHAN trên cột IDBP



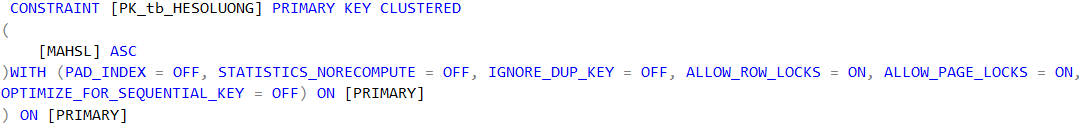
### Ràng buộc toàn vẹn trên bảng CHUCVU

Ràng buộc khóa chính tên là PK\_tb\_CHUCVUC trên cột IDCV



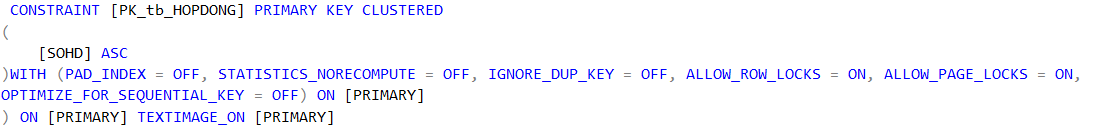
### Ràng buộc toàn vẹn trên bảng HESOLUONG

Ràng buộc khóa chính tên là PK\_tb\_HESOLUONG trên cột MAHSL



### Ràng buộc toàn vẹn trên bảng HOPDONG

Ràng buộc khóa chính tên là PK\_tb\_HOPDONG trên cột SOHD

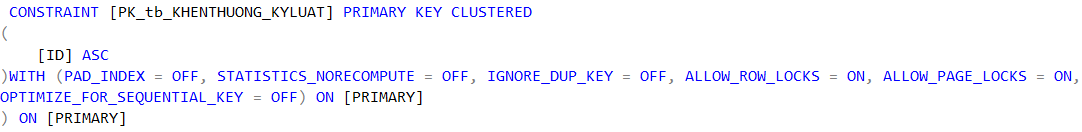


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_HOPDONG\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MaNV** của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng KHENTHUONG\_KYLUAT

Ràng buộc khóa chính tên là PK\_tb\_KHENTHUONG\_KYLUAT trên cột ID

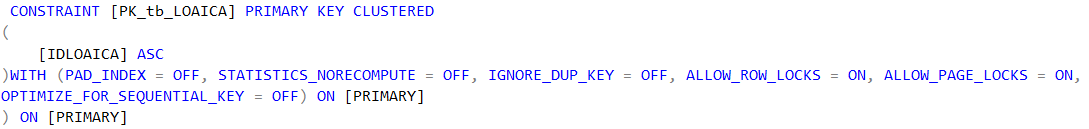


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_KHENTHUONG\_tb\_NHANVIEN trên cột **IDLUONG** tham chiếu đến cột **IDLUONG** của bảng THANHTOANLUONG



### Ràng buộc toàn vẹn trên bảng LOAICA

Ràng buộc khóa chính tên là PK\_tb\_LOAICA trên cột IDLOAICA



### Ràng buộc toàn vẹn trên bảng THANGCONG

Ràng buộc khóa chính tên là PK\_THANGCONG trên cột ID

A picture containing text

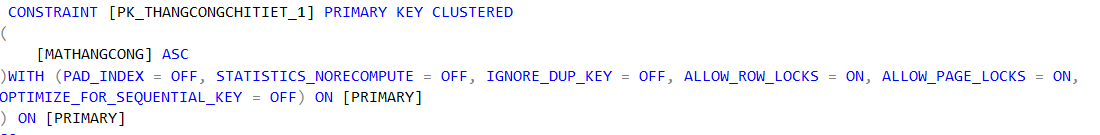
Description automatically generated

Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TC\_tb\_TCCT trên cột **MATHANGCONG** tham chiếu đến cột **MATHANGCONG** của bảng THANGCONGCHITIET



### Ràng buộc toàn vẹn trên bảng THANGCONGCHITIET

Ràng buộc khóa chính tên là PK\_THANGCONGCHITIET\_1 trên cột MATHANGCONG

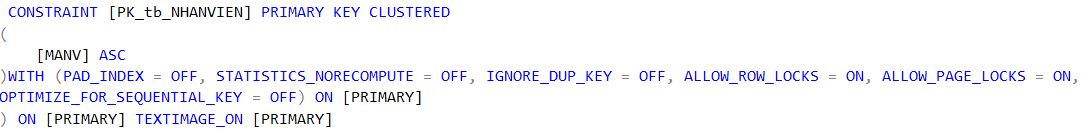


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TCCT\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV**của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng NHANVIEN

Ràng buộc khóa chính tên là PK\_tb\_NHANVIEN trên cột MANV



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_tb\_BOPHAN trên cột **IDBP** tham chiếu đến cột **IDBP** của bảng BOPHAN



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_tb\_CHUCVU trên cột **IDCV** tham chiếu đến cột **IDCV** của bảng CHUCVU



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_tb\_PHONGBAN trên cột **IDPB** tham chiếu đến cột **IDPB** của bảng PHONGBAN



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_tb\_TRINHDO trên cột **IDTD** tham chiếu đến cột **IDTD** của bảng TRINHDO

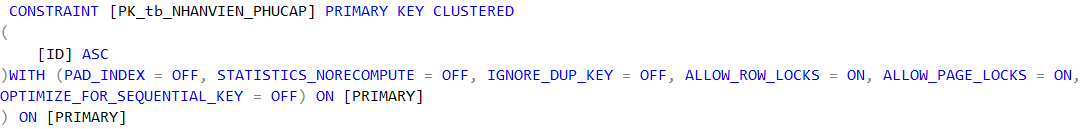


Khai báo 1 ràng buộc tên là FK\_tb\_NHANVIEN\_MANQL\_tb\_NHANVIEN\_MANV trên cột **MA\_NQL** tham chiếu đến cột **MANV** của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng NHANVIEN\_PHUCAP

Ràng buộc khóa chính tên là PK\_tb\_NHANVIEN\_PHUCAP trên cột ID



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_PHUCAP\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV** của bảng NHANVIEN

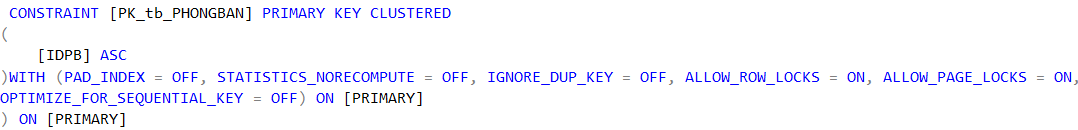


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_NHANVIEN\_PHUCAP\_tb\_PHUCAP trên cột **IDPC** tham chiếu đến cột **IDPC** của bảng PHUCAP



### Ràng buộc toàn vẹn trên bảng PHONGBAN

Ràng buộc khóa chính tên là PK\_tb\_PHONGBAN trên cột IDPB

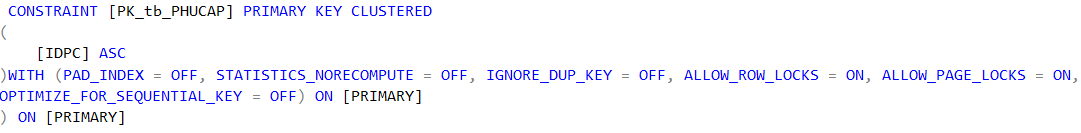


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_PHONGBAN\_TP\_tb\_NHANVIEN trên cột TRUONGPHONG tham chiếu đến cột **MANV** của bảng NHANVIEN



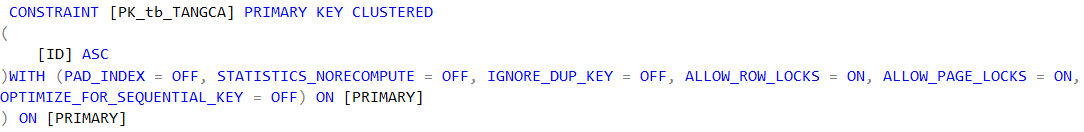
### Ràng buộc toàn vẹn trên bảng PHUCAP

Ràng buộc khóa chính tên là PK\_tb\_PHUCAP trên cột IDPC



### Ràng buộc toàn vẹn trên bảng TANGCA

Ràng buộc khóa chính tên là PK\_tb\_TANGCA trên cột ID



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TANGCA\_tb\_LOAICA trên cột **IDLOAICA** tham chiếu đến cột **IDLOAICA** của bảng LOAICA

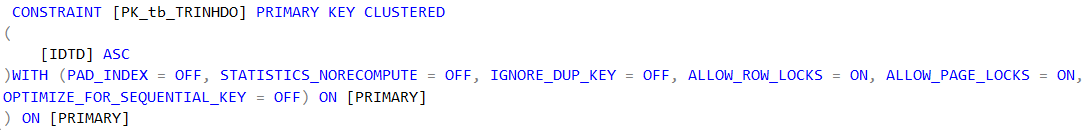


Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TANGCA\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV** của bảng MANV



### Ràng buộc toàn vẹn trên bảng TRINHDO

Ràng buộc khóa chính tên là PK\_tb\_TRINHDO trên cột IDTD



### Ràng buộc toàn vẹn trên bảng UNGLUONG

Ràng buộc khóa chính tên là PK\_tb\_UNGLUONG trên cột ID

A picture containing logo

Description automatically generated

Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_UNGLUONG\_tb\_NHANVIEN trên cột **MANV** tham chiếu đến cột **MANV** của bảng NHANVIEN



### Ràng buộc toàn vẹn trên bảng THANHTOANLUONG

Ràng buộc khóa chính tên là PK\_tb\_TT\_LUONG trên cột IDLUONG

A picture containing company name

Description automatically generated

Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TTL\_tb\_NV trên cột **MANV** tham chiếu đến cột **MANV** của bảng NHANVIEN



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TTL\_tb\_UL trên cột **ID** tham chiếu đến cột **ID** của bảng UNGLUONG



Khai báo 1 ràng buộc khóa ngoại tên là FK\_tb\_TTL\_tb\_PC trên cột **IDPC** tham chiếu đến cột **IDPC** của bảng PHUCAP



# **Chương 4: Thiết Kế Xử Lý Trong Client/Server**

## **Giao Tác**

### Tra Cứu Thông Tin Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thông tin về nhân viên tương ứng |  |
|  | Nhận MANV  Tra cứu thông tin nhân viên tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Hợp Đồng Của Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin về hợp đồng của và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin hợp đồng tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Người Quản Lý Của Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin về người quản lý của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin người quan lý tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm Chức Vụ Mới

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm chức vụ mới và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra mã chức vụ xem có trùng hay không  Thêm thông tin mới vào bảng Chứ Vụ  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xem Thông Tin Bảo Hiểm Của Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin về bảo hiểm của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin bảo hiểm tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xem Thông Tin Bộ Phận Và Chức Vụ Của Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin về bộ phận và chức vụ của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin bộ phận và chức vụ tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Nhân Viên Tăng Ca

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin tăng ca của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin tăng ca tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm Phòng Ban

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm phòng và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra id phòng ban xem có trùng hay không  Thêm thông tin mới vào bảng Phòng Ban  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xóa Phòng Ban

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xóa phòng ban và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem id phòng ban xem có tồn tại hay không  Xóa thông tưng ứng trong bảng Phòng Ban  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người  dùng |  |

### Xem Phòng Ban

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin phong ban và truyền tham số |  |
|  | Nhận IDPB  Tra cứu thông tin phòng ban tướng ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Cập Nhật Thông Tin Bảo Hiểm

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server cập nhật thông tin bảo hiểm và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem id bảo hiểm xem có tồn tại hay không  Cập nhật thông tưng ứng trong bảng Bảo Hiểm  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Thông Tin Ứng Lương Của Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin ứng lương của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin ứng lương tương ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm Vào Bảng Ứng Lương

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm vào bảng Ứng Lương và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra id ứng lương xem có trùng hay không  Kiểm tra xem mã nhân viên có tồn tại hay không  Thêm thông tin mới vào bảng Ứng Lương  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Thông Tin Công Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm thông tin công của nhân viên và truyền tham số |  |
|  | Nhận MANV  Tra cứu thông tin công tương ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tra Cứu Số Lượng Nhân Viên Trong Phòng Ban

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xem số lượng nhân viên của phòng ban và truyền tham số |  |
|  | Nhận IDPB  Tra cứu và đếm số lượng nhân viên của phòng ban tương ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xóa HopDong và Xóa Nhân Viên

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xóa hợp đồng và xóa thông tin nhân viên và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem id hợp đồng có tồn tại hay không  Kiểm tra xem mã nhân viên có tồn tại hay không  Xóa hợp đồng tương ứng ở bản hợp đồng  Xóa thông tin nhân viên tương ứng ở bản nhân viên  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm Nhân Viên (ĐH)

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm nhân viên mới và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra xem mã nhân viên có trùng hay không  Kiểm tra phòng ban của nhân viên  Thêm thông tin tương ứng vào bảng nhân viên  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Khen Thưởng Nhân Viên (ĐH)

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm thông tin mới vào bảng khen thưởng và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra xem mã nhân viên có tồn tại hay không  Kiểm tra xem ID khen thưởng có trùng hay không  Kiểm tra phòng ban của nhân viên  Thêm thông tin tương ứng vào bảng khen thưởng  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Phân Công Tăng Ca (ĐH)

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm thông tin mới vào bảng tăng ca và truyền tham số |  |
|  | Nhận tham số  Kiểm tra xem tham số có trống hay không  Kiểm tra xem mã nhân viên có tồn tại hay không  Kiểm tra xem ID tăng ca có trùng hay không  Kiểm tra phòng ban của nhân viên  Thêm thông tin tương ứng vào bảng tăng ca  Thông báo cho người dùng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

## **Phân Quyền**

### Các đối tượng phân quyền

* Quản Lý: Quản Lý Của Công Ty
* Trường Phòng: trường phòng ban trong công ty
* Kế toán: kế toán của công ty, tổng hợp, tính lương nhân viên
* Nhân viên: nhân viên của công ty

### Phân quyền trên các bảng

Ghi chú: S – Select, I – Insert, U – Update, D – Delete

Bảng 19: Bảng phân quyền trên các bảng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Quản Lý | Trường Phòng | Kế Toán | Nhân Viên |
| Tb\_HESOLUONG | S,I,U,D | S | S |  |
| Tb\_BAOHIEM | S,I,U,D | S | S | S |
| Tb\_BIENDONG  HSL | S,I,U,D |  | S |  |
| Tb\_THANGCONG | S,I,U,D |  | S,I,U |  |
| Tb\_CHUCVU | S,I,U,D | S |  |  |
| Tb\_TRINHDO | S,I,U,D | S |  |  |
| Tb\_HOPDONG | S,I,U,D | S,I,U | S |  |
| Tb\_NHANVIEN | S,I,U,D | S,I,U | S | S |
| Tb\_PHONGBAN | S,I,U,D | S |  |  |
| Tb\_UNGLUONG | S,I,U,D | S | S,I,U | S |
| Tb\_NHANVIEN\_  PHUCAP | S,I,U,D | S | S,I,U | S |
| Tb\_BOPHAN | S,I,U,D | S |  |  |
| Tb\_KHENTHUONG  KYLUAT | S,I,U,D | S | S,I,U | S |
| Tb\_LOAICA | S,I,U,D | S | S |  |
| Tb\_THANGCONG  CHITIET | S,I,U,D | S,I,U | S |  |
| Tb\_PHUCCAP | S,I,U,D | S | S |  |
| Tb\_TANGCA | S,I,U,D | S,I,U | S | S |
| Tb\_THANHTOAN  LUONG | S,I,U,D | S | S,I,U | S |

### Phân Quyền Trên Các Giao Tác

Bảng 20: Bảng phân quyền trên các giao tác

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Quản Lý | Trường Phòng | Kế Toán | Nhân Viên |
| SP\_TC\_THONGTIN  NV | X | X | X | X |
| SP\_TC\_HOPDONG  NHANVIEN | X | X | X | X |
| SP\_TC\_NHANVIEN  \_PHONGBAN\_NQL | X | X | X | X |
| SP\_THEM\_CHUC  VU | X |  |  |  |
| SP\_TC\_NHANVIEN  BAOHIEM | X | X | X | X |
| SP\_TC\_BOPHAN  \_CHUCVU\_NHAN  VIEN | X | X |  |  |
| SP\_TC\_NHANVIEN  \_TANGCA | X | X | X | X |
| SP\_THEM\_PB | X |  |  |  |
| SP\_XOA\_PB | X |  |  |  |
| SP\_UPDATE  \_BAOHIEM | X |  |  |  |
| SP\_TC\_NHANVIEN  \_UNGLUONG | X | X | X | X |
| SP\_INSERT\_UL | X |  | X |  |
| SP\_TC\_CONG  \_NHANVIEN | X | X | X | X |
| TRACUU\_DSNV  \_SLNV\_PB | X | X | X |  |
| TRACUU\_SL\_  TRINHDO\_NV\_PB | X | X |  |  |
| XOA\_HD\_NV | X |  |  |  |
| SP\_THEMNV\_DH | X | X |  |  |
| SP\_KT\_NV\_DH | X | X |  |  |
| SP\_PC\_TC\_DH | X | X |  |  |

## **Backup**

* Full database backup: Vào chủ nhật (2 tuần 1 lần) vào lúc 24 giờ.
* Differenitial backup: Mỗi thứ 3, thứ 5, thứ 7 hàng tuần vào lúc 12 giờ
* Logs backups: Mỗi bốn tiếng (từ 12 tới 24h)

# **Chương 5: Triển Khai Thực Nghiệm**

## **Phía Server**

### Tạo CSDL

USE [QLNHANSU]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_BANGCONG] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_BANGCONG](

[IDBC] [int] IDENTITY(1,1) NOT NULL,

[NAM] [int] NULL,

[THANG] [int] NULL,

[NGAY] [int] NULL,

[GIOVAO] [int] NULL,

[PHUTVAO] [int] NULL,

[GIORA] [int] NULL,

[PHUTRA] [int] NULL,

[MANV] [char](5) NULL,

[IDLC] [int] NULL,

CONSTRAINT [PK\_tb\_BANGCONG] PRIMARY KEY CLUSTERED

(

[IDBC] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_BAOHIEM] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_BAOHIEM](

[IDBH] [int] IDENTITY(1,1) NOT NULL,

[SOBH] [nvarchar](50) NULL,

[NGAYCAP] [datetime] NULL,

[NOICAP] [nvarchar](50) NULL,

[NOIDK] [nvarchar](50) NULL,

[MANV] [char](5) NULL,

CONSTRAINT [PK\_tb\_BAOHIEM] PRIMARY KEY CLUSTERED

(

[IDBH] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_BOPHAN] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_BOPHAN](

[IDBP] [int] IDENTITY(1,1) NOT NULL,

[TENBP] [nvarchar](50) NULL,

CONSTRAINT [PK\_tb\_BOPHAN] PRIMARY KEY CLUSTERED

(

[IDBP] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_CHUCVU] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_CHUCVU](

[IDCV] [int] IDENTITY(1,1) NOT NULL,

[TENCV] [nvarchar](50) NULL,

CONSTRAINT [PK\_tb\_CHUCVU] PRIMARY KEY CLUSTERED

(

[IDCV] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_HOPDONG] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_HOPDONG](

[SOHD] [int] IDENTITY(1,1) NOT NULL,

[NGAYBAYDAU] [datetime] NULL,

[NGAYKETTHUC] [datetime] NULL,

[NGAYKY] [datetime] NULL,

[NOIDUNG] [nvarchar](max) NULL,

[LANKY] [int] NULL,

[THOIHAN] [nvarchar](50) NULL,

[HESOLUONG] [float] NULL,

[MANV] [char](5) NULL,

CONSTRAINT [PK\_tb\_HOPDONG] PRIMARY KEY CLUSTERED

(

[SOHD] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_KHENTHUONG\_KYLUAT] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_KHENTHUONG\_KYLUAT](

[ID] [int] IDENTITY(1,1) NOT NULL,

[SOKTKL] [int] NULL,

[NOIDUNG] [nvarchar](500) NULL,

[NGAY] [datetime] NULL,

[MANV] [char](5) NULL,

[LOAI] [int] NULL,

CONSTRAINT [PK\_tb\_KHENTHUONG\_KYLUAT] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_LOAICA] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_LOAICA](

[IDLOAICA] [int] IDENTITY(1,1) NOT NULL,

[TENLOAICA] [nvarchar](50) NULL,

[HESO] [float] NULL,

CONSTRAINT [PK\_tb\_LOAICA] PRIMARY KEY CLUSTERED

(

[IDLOAICA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_LOAICONG] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_LOAICONG](

[IDLC] [int] IDENTITY(1,1) NOT NULL,

[TENLC] [nvarchar](50) NULL,

[HESO] [float] NULL,

CONSTRAINT [PK\_tb\_LOAICONG] PRIMARY KEY CLUSTERED

(

[IDLC] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_NHANVIEN] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_NHANVIEN](

[MANV] [char](5) CONSTRAINT IDNV DEFAULT DBO.AUTO\_IDNV(),

[HOTEN] [nvarchar](50) NULL,

[GIOITINH] [bit] NULL,

[NGAYSINH] [datetime] NULL,

[DIENTHOAI] [nvarchar](50) NULL,

[CCCD] [nvarchar](50) NULL,

[DIACHI] [nvarchar](500) NULL,

[HINHANH] [varbinary](max) NULL,

[IDPB] [int] NULL,

[IDBP] [int] NULL,

[IDCV] [int] NULL,

[IDTD] [int] NULL,

CONSTRAINT [PK\_tb\_NHANVIEN] PRIMARY KEY CLUSTERED

(

[MANV] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_NHANVIEN\_PHUCAP] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_NHANVIEN\_PHUCAP](

[ID] [int] IDENTITY(1,1) NOT NULL,

[MANV] [char](5) NULL,

[IDPC] [int] NULL,

[NGAY] [datetime] NULL,

[NOIDUNG] [nvarchar](500) NULL,

[SOTIEN] [float] NULL,

CONSTRAINT [PK\_tb\_NHANVIEN\_PHUCAP] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_PHONGBAN] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_PHONGBAN](

[IDPB] [int] IDENTITY(1,1) NOT NULL,

[TENPB] [nvarchar](50) NULL,

CONSTRAINT [PK\_tb\_PHONGBAN] PRIMARY KEY CLUSTERED

(

[IDPB] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_PHUCAP] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_PHUCAP](

[IDPC] [int] IDENTITY(1,1) NOT NULL,

[TENPC] [nvarchar](50) NULL,

[SOTIEN] [float] NULL,

CONSTRAINT [PK\_tb\_PHUCAP] PRIMARY KEY CLUSTERED

(

[IDPC] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_TANGCA] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_TANGCA](

[ID] [int] IDENTITY(1,1) NOT NULL,

[NAM] [int] NULL,

[THANG] [int] NULL,

[NGAY] [int] NULL,

[SOGIO] [float] NULL,

[MANV] [char](5) NULL,

[IDLOAICA] [int] NULL,

CONSTRAINT [PK\_tb\_TANGCA] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_TRINHDO] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_TRINHDO](

[IDTD] [int] IDENTITY(1,1) NOT NULL,

[TENTD] [nvarchar](50) NULL,

CONSTRAINT [PK\_tb\_TRINHDO] PRIMARY KEY CLUSTERED

(

[IDTD] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tb\_UNGLUONG] Script Date: 27/09/2022 8:26:28 SA \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tb\_UNGLUONG](

[ID] [int] IDENTITY(1,1) NOT NULL,

[NAM] [int] NULL,

[THANG] [int] NULL,

[NGAY] [int] NULL,

[SOTIEN] [float] NULL,

[TRANGTHAI] [bit] NULL,

[MANV] [char](5) NULL,

CONSTRAINT [PK\_tb\_UNGLUONG] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[tb\_HESOLUONG](

[MAHSL] [char](4) CONSTRAINT IDHSL DEFAULT DBO.AUTO\_IDHSL(),

[BAC] [int] NULL,

[NHOMNGACH] [char](20) NULL,

[HESOLUONG] [float] NULL

CONSTRAINT [PK\_tb\_HESOLUONG] PRIMARY KEY CLUSTERED

(

[MAHSL] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[tb\_BIENDONGHSL](

[MAHSL] [char](4) NOT NULL,

[NAMAPDUNG] DATE,

[BAC] [int] NULL,

[NHOMNGACH] [char](20) NULL,

[HESOLUONG] [float] NULL

CONSTRAINT [PK\_tb\_BIENDONGHSL] PRIMARY KEY CLUSTERED

(

[MAHSL] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[tb\_THANHTOANLUONG](

[IDLUONG] [INT] IDENTITY(1,1) NOT NULL,

[THANGTT] [INT] NULL,

[NAMTT] [INT] NULL,

[MANV] [CHAR](5) NULL,

[MACC] [INT] NULL,

[LUONGCT] [FLOAT] NULL,

[THUETNCN] [FLOAT] NULL,

[LUONG] [FLOAT] NULL,

[ID] [int] NULL

CONSTRAINT [PK\_tb\_TT\_LUONG] PRIMARY KEY CLUSTERED

(

[IDLUONG] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[tb\_THANGCONGCHITIET](

[MATHANGCONG] [int] NOT NULL,

[MANV] [char](5) NOT NULL,

[N1] [nvarchar](10) NULL,

[N2] [nvarchar](10) NULL,

[N3] [nvarchar](10) NULL,

[N4] [nvarchar](10) NULL,

[N5] [nvarchar](10) NULL,

[N6] [nvarchar](10) NULL,

[N7] [nvarchar](10) NULL,

[N8] [nvarchar](10) NULL,

[N9] [nvarchar](10) NULL,

[N10] [nvarchar](10) NULL,

[N11] [nvarchar](10) NULL,

[N12] [nvarchar](10) NULL,

[N13] [nvarchar](10) NULL,

[N14] [nvarchar](10) NULL,

[N15] [nvarchar](10) NULL,

[N16] [nvarchar](10) NULL,

[N17] [nvarchar](10) NULL,

[N18] [nvarchar](10) NULL,

[N19] [nvarchar](10) NULL,

[N20] [nvarchar](10) NULL,

[N21] [nvarchar](10) NULL,

[N22] [nvarchar](10) NULL,

[N23] [nvarchar](10) NULL,

[N24] [nvarchar](10) NULL,

[N25] [nvarchar](10) NULL,

[N26] [nvarchar](10) NULL,

[N27] [nvarchar](10) NULL,

[N28] [nvarchar](10) NULL,

[N29] [nvarchar](10) NULL,

[N30] [nvarchar](10) NULL,

[N31] [nvarchar](10) NULL,

[NGAYCONG] [float] NULL,

[NGAYPHEP] [float] NULL,

[NGHIKHONGPHEP] [float] NULL,

[CONGNGAYLE] [float] NULL,

[CONGCHUNHAT] [float] NULL,

[TONGNGAYCONG] [float] NULL,

CONSTRAINT [PK\_THANGCONGCHITIET\_1] PRIMARY KEY CLUSTERED

(

[MATHANGCONG] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

CREATE TABLE [dbo].[tb\_THANGCONG](

[ID] [int] IDENTITY(1,1) NOT NULL,

[MATHANGCONG] [int] NOT NULL,

[THANG] [int] NULL,

[NAM] [int] NULL,

[KHOA] [bit] NULL,

[NGAYTINHCONG] [datetime] NULL,

[NGAYCONGTRONGTHANG] [float] NULL,

[TRANGTHAI] [bit] NULL,

CONSTRAINT [PK\_THANGCONG] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON,

OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

## **Mô Tả STORE PROCEDURE, FUNCTION, VIEW, TRIGGER**

### Mô Tả Function

#### Function: AUTO\_IDNV

+ Công dụng: Function tạo tự động MANV theo cú pháp: NV00+Số mỗi khi có một nhân viên mới được nhập vào

|  |
| --- |
| CREATE FUNCTION AUTO\_IDNV()  RETURNS VARCHAR(5)  AS  BEGIN  DECLARE @ID VARCHAR(5)  IF (SELECT COUNT(MANV) FROM tb\_NHANVIEN) = 0  SET @ID = '0'  ELSE  SELECT @ID = MAX(RIGHT(MANV, 3)) FROM tb\_NHANVIEN  SELECT @ID = CASE  WHEN @ID >= 0 and @ID < 9 THEN 'NV00' + CONVERT(CHAR, CONVERT(INT, @ID) + 1)  WHEN @ID >= 9 THEN 'NV0' + CONVERT(CHAR, CONVERT(INT, @ID) + 1)  END  RETURN @ID  END |

#### Function: AUTO\_MA\_BH

+ Công dụng: Function tạo tự động MABH, mỗi khi nhập vào thông tin bảo hiểm của nhân viên vào bảng, một SOBH sẽ được tự động sinh ra.

|  |
| --- |
| CREATE FUNCTION AUTO\_MA\_BH()  RETURNS VARCHAR(15)  AS  BEGIN  DECLARE @MABH VARCHAR(15)  IF (SELECT COUNT(SOBH) FROM tb\_BAOHIEM) = 0  SET @MABH = '0'  ELSE  SELECT @MABH = MAX(RIGHT(SOBH, 2)) FROM tb\_BAOHIEM  SELECT @MABH = CASE  WHEN @MABH >= 0 and @MABH < 9 THEN 'DN479122510020' + CONVERT(CHAR, CONVERT(INT, @MABH) + 1)  WHEN @MABH >= 9 and @MABH <20 THEN 'DN40101161520' + CONVERT(CHAR, CONVERT(INT, @MABH) + 1)  WHEN @MABH >= 20 THEN 'DN47912299020' + CONVERT(CHAR, CONVERT(INT, @MABH) + 1)  END  RETURN @MABH  END |

#### Function: AUTO\_MA\_HD

+ Công dụng: Function tạo tự động MAHD theo cú pháp: HDLD00000+Số mỗi khi có một hợp đồng mới được thêm vào bảng HOPDONG.

|  |
| --- |
| CREATE FUNCTION AUTO\_MA\_HD()  RETURNS VARCHAR(10)  AS  BEGIN  DECLARE @ID VARCHAR(10)  IF (SELECT COUNT(SOHD) FROM tb\_HOPDONG) = 0  SET @ID = '0'  ELSE  SELECT @ID = MAX(RIGHT(SOHD, 2)) FROM tb\_HOPDONG  SELECT @ID = CASE  WHEN @ID >= 0 and @ID < 9 THEN 'HDLD00000' + CONVERT(CHAR, CONVERT(INT, @ID) + 1)  WHEN @ID >= 9 THEN 'HDLD0000' + CONVERT(CHAR, CONVERT(INT, @ID) + 1)  END  RETURN @ID  END |

### Mô Tả Store Procedure

#### Stored Procedure cho câu lệnh Select

+ **Procedure tra cứu thông tin nhân viên**: Lấy danh sách các field của bảng NHANVIEN theo MANV nhập vào, sau đó in ra thông tin nhân viên mà người dùng tra cứu, nếu MANV không tồn tại, câu lệnh báo lỗi sẽ được in ra. Dùng để tra cứu thông tin của MANV được nhập.

|  |
| --- |
| create proc sp\_TC\_ThongtinNV  @MANV char(5)  as  begin  if exists (select MANV from tb\_NHANVIEN where MANV=@MANV)  begin  select \*  from tb\_NHANVIEN  where MANV=@MANV  end  else  print 'Ma nhan vien khong ton tai'  end  execute sp\_TC\_ThongtinNV 'NV001' |

+ **Procedure tra cứu thông tin hợp đồng của nhân viên**: Lấy danh sách các field của bảng HOPDONG và NHANVIEN tai vị tí MANV nhập vào, sau đó hiển thị cho người dùng. Nếu không tìm thấy MANV tương ứng, dòng thông báo sẽ được in ra.Dùng để tra cứu thông tin hợp đồng của nhân viên với MANV được nhập.

|  |
| --- |
| create proc sp\_TC\_HopDongNhanVien  @MANV char(5)  as  begin  if exists (select MANV from tb\_HOPDONG where MANV=@MANV)  begin  select NV.HOTEN, HD.SOHD, HD.NGAYBAYDAU, HD.NGAYKETTHUC, HD.NGAYKY, HD.NOIDUNG, HD.THOIHAN, HD.HESOLUONG  from tb\_NHANVIEN NV, tb\_HOPDONG HD  where NV.MANV=HD.MANV  and HD.MANV=@MANV  end  else  print 'Ma nhan vien khong ton tai'  end  execute sp\_TC\_HopDongNhanVien 'NV003' |

+ **Procedure tra cứu người quản lí và phòng ban của nhân viên**: Lấy danh sách các field từ bảng NHANVIEN và PHONGBAN, sau đó hiển thị thông tin tương ứng cho người sử dụng. Nếu không tìm thấy MANV tương ứng, dòng thông báo sẽ được in ra. Dùng để tra cứu người quản lí của nhân viên cũng như thông tin phòng ban mà nhân viên làm việc theo MANV nhập vào.

|  |
| --- |
| create proc sp\_TC\_NHANVIEN\_PHONGBAN\_NQL  @MANV char(5)  as  begin  if exists (select MANV from tb\_NHANVIEN where MANV=@MANV)  begin  select NV1.MANV, NV1.HOTEN, NV1.MA\_NQL, NV2.HOTEN, PB.IDPB, PB.TENPB  FROM tb\_NHANVIEN NV1 LEFT JOIN tb\_NHANVIEN NV2 ON NV1.MA\_NQL = NV2.MANV, tb\_PHONGBAN PB  where NV1.IDPB=PB.IDPB  and NV1.MANV=@MANV;  end  else  print 'Ma nhan vien khong ton tai'  end  execute sp\_TC\_NHANVIEN\_PHONGBAN\_NQL 'NV023' |

+ **Procedure xem thông tin bảo hiểm của nhân viên**: Lấy danh sách các field từ bảng NHANVIEN và BAOHIEM tại vị trí MANV nhập vào, sau đó hiển thị thông tin tương ứng cho người dùng. Nếu không tìm thấy MANV tương ứng, dòng thông báo sẽ được in ra. Dùng để tra cứu thông tin về bảo hiểm của nhân viên theo MANV nhập vào

|  |
| --- |
| create proc sp\_TC\_BOPHAN\_CHUCVU\_NHANVIEN  @MANV char(5)  as  begin  if exists (select MANV from tb\_NHANVIEN where MANV=@MANV)  begin  select NV.MANV, NV.HOTEN, BP.IDBP,BP.TENBP, CV.IDCV, CV.TENCV  FROM tb\_NHANVIEN NV, tb\_BOPHAN BP, tb\_CHUCVU CV  where NV.IDBP=BP.IDBP  and NV.IDCV=CV.IDCV  and NV.MANV=@MANV;  end  else  print 'Ma nhan vien khong ton tai'  end  execute sp\_TC\_BOPHAN\_CHUCVU\_NHANVIEN 'NV003' |

+ **Procedure tra cứu nhân viên tăng ca**: Lấy danh sách các field từ bảng NHANVIEN và bảng TANGCA tại vị tí MANV nhập vào, sau đó hiển thị thông tin tương ứng cho người dùng. Nếu nhân viên không tăng ca, dòng thông báo sẽ được in ra. Dùng để tra cứu thông tin tăng ca và xem nhân viên có tăng ca hay không.

|  |
| --- |
| create proc sp\_TC\_NHANVIEN\_TANGCA  @MANV char(5)  as  begin  if exists (select MANV from tb\_TANGCA where MANV=@MANV)  begin  select NV.HOTEN, TC.ID, TC.IDLOAICA, TC.NGAY, TC.THANG, TC.NAM, TC.SOGIO  from tb\_TANGCA TC, tb\_NHANVIEN NV  where TC.MANV=NV.MANV  and TC.MANV=@MANV  end  else  print 'Nhan Vien Khong Tang Ca'  end  execute sp\_TC\_NHANVIEN\_TANGCA 'NV023' |

+ **Procedure tra cứu thông tin ứng lương của nhân viên**: Lấy danh sách các field từ bảng UNGLUONG tại vị tí MANV nhập vào, sau đó hiển thị thông tin tương ứng cho người dùng. Nếu nhân viên không ứng lương thì dòng thông báo sẽ được in ra. Dùng để tra cứu thông tin ứng lương của ngân viên.

|  |
| --- |
| create proc sp\_TC\_NHANVIEN\_UNGLUONG  @MANV char(5)  as  begin  if exists (select MANV from tb\_UNGLUONG where MANV=@MANV)  begin  select \*  from tb\_UNGLUONG  where MANV=@MANV  end  else  print 'Nhan Vien Khong Ung Luong'  end  execute sp\_TC\_NHANVIEN\_UNGLUONG 'NV003' |

+ **Procedure tra cứu công nhân viên**: Lấy danh sách các field từ bảng BANGCONG tại vị trí tương ứng với MANV, NAM, THANG nhập vào, sau đó hiển thị thông tin tương ứng. Nếu không tìm được công nhân viên tương ứng với MANV, NAM, THANG được nhập vào, thì dòng lệnh thông báo sẽ được in ra. Dùng để tra cứu thông tin công của nhân viên.

|  |
| --- |
| create proc sp\_TC\_Cong\_Nhanvien  @MANV char(5),  @Nam int,  @Thang int,  @Ngay int  as  begin  if exists (select MANV from tb\_BANGCONG where MANV=@MANV and NAM=@Nam and THANG=@Thang and NGAY=@Ngay)  begin  select \*  from tb\_BANGCONG  where MANV=@MANV  and Nam=@Nam  and THANG=@Thang and NGAY=@Ngay  end  else  print 'Khong tim thay cham cong cua nhan vien'  end  execute sp\_TC\_Cong\_Nhanvien |

+ **Procedure số lượng hoặc lấy danh sách nhân viên của phòng ban**: Lấy danh sách các field từ bảng NHANVIEN, PHONGBAN tại vị tí IDPB nhập vào, nếu StatemenType là ‘DS’ thì thông tin danh sách phòng ban tương ứng với IDPB sẽ được in ra còn nếu nếu StatemenType là ‘SL’ thì số lượng cụ thể của nhân viên trong phòng ban tương ứng với IDPB sẽ được in ra. Dùng để tra cứu danh sách các các nhân viên làm việc tại phòng ban hoặc tra cứu số lượng nhân viên của phòng ban tương ứng với IDPB nhập vào.

|  |
| --- |
| CREATE PROCEDURE TraCuu\_DSNV\_SLNV\_PB (@IDPB int,  @StatementType NVARCHAR(20) = '')  AS  BEGIN  IF @StatementType = 'DS'  BEGIN  SELECT \*  FROM tb\_NHANVIEN NV  where NV.IDPB=@IDPB  END  IF @StatementType = 'SL'  BEGIN  SELECT PB.TENPB,COUNT(NV.MANV) as 'SO LUONG NHAN VIEN'  FROM tb\_NHANVIEN NV, tb\_PHONGBAN PB  where PB.IDPB=NV.IDPB  and PB.IDPB=@IDPB  GROUP BY PB.TENPB  END  END  execute TraCuu\_DSNV\_SLNV\_PB 2002, 'DS' |

+ **Procedure xem trình độ của nhân viên trong phòng ban:** Lấy danh sách các field từ bảng NHANVIEN, TRINHDO tại vị tí MAPHONG nhập vào tại vị tí IDPB nhập vào, nếu StatemenType là ‘DH’ thì các nhân viên trình độ đại học của phòng ban với IDPB tương ứng sẽ được in ra còn nếu StatemenType là ‘CD’ thì các nhân viên trình độ cao đẳng của phòng ban với IDPB tương ứng sẽ được in ra. Dùng để tra cứu trình độ của các nhân viên làm việc tại phòng ban tương ứng với IDPB nhập vào.

|  |
| --- |
| CREATE PROCEDURE TraCuu\_SL\_TrinhDo\_NV\_PB (@IDPB int,  @StatementType NVARCHAR(20) = '')  AS  BEGIN  IF @StatementType = 'DH'  BEGIN  SELECT TD.TENTD, COUNT(NV.MANV) as 'SO LUONG NHAN VIEN TD DH'  FROM tb\_NHANVIEN NV, tb\_TRINHDO TD  where NV.IDPB=@IDPB and NV.IDTD=201  and NV.IDTD=TD.IDTD  GROUP BY TD.TENTD  END  IF @StatementType = 'CD'  BEGIN  SELECT TD.TENTD, COUNT(NV.MANV) as 'SO LUONG NHAN VIEN TD CD'  FROM tb\_NHANVIEN NV, tb\_TRINHDO TD  where NV.IDPB=@IDPB and NV.IDTD=200  and NV.IDTD=TD.IDTD  GROUP BY TD.TENTD  END  END  execute TraCuu\_SL\_TrinhDo\_NV\_PB 2002, 'DH' |

#### Stored Procedure cho câu lệnh Insert, Update, Delete

+ **Procedure thêm chức vụ mới**: Dùng để thêm chức vụ vào bảng CHUCVU với thông tin nhập từ bàn phím.

|  |
| --- |
| create proc sp\_Them\_ChucVu  @TENCV nvarchar(50)  as  declare @macv int  begin  begin try  if(@TENCV is null)  begin  print 'Thong tin nhap rong'  return  end    end try    begin catch  --báo lỗi  print 'Them khong thanh cong'  end catch    set @macv=1000  while(exists(select \* from tb\_CHUCVU where IDCV=@macv))  set @macv=@macv+1  SET IDENTITY\_INSERT tb\_CHUCVU ON  Insert into tb\_CHUCVU(IDCV,TENCV) values(@macv,@TENCV)  SET IDENTITY\_INSERT tb\_CHUCVU OFF  end  execute sp\_Them\_ChucVu 'Nhan Vien 2' |

+ **Procedure xóa, thêm, chọn phòng ban**: Dùng để xóa, chọn hoặc thêm phòng ban vào bảng PHONGBAN với thông tin nhập từ bàn phím và thực thi xóa, chọn hoặc thêm dựa trên StatementType tương ứng (Insert-Chọn, Delete-Xóa, Select-Chọn)

|  |
| --- |
| CREATE PROCEDURE Insertdelete\_PB (@ID int,  @TENPB nvarchar(50),  @TRUONGPHONG char(5),  @StatementType NVARCHAR(20) = '')  AS  BEGIN  IF @StatementType = 'Insert'  BEGIN  INSERT INTO tb\_PHONGBAN  (TENPB, TRUONGPHONG)  VALUES ( @TENPB,  @TRUONGPHONG)  END  IF @StatementType = 'Select'  BEGIN  SELECT \*  FROM tb\_PHONGBAN  END  ELSE IF @StatementType = 'Delete'  BEGIN  DELETE FROM tb\_PHONGBAN  WHERE IDPB = @id  END  END  execute Insertdelete\_PB 2009, N'Nhân Sự Hai', 'NV027', 'Delete' |

+ **Procedure cập nhật thông tin bảo hiểm**: Dùng để cập nhật thông tin bảo hiểm tại vị trí IDBH được nhập vào

|  |
| --- |
| create proc sp\_Update\_BH  @IDBH int,  @Ngaycap datetime,  @Noicap nvarchar(50),  @NoiDK nvarchar(50)  as  begin  if exists (select IDBH from tb\_BAOHIEM where IDBH=@IDBH)  begin  UPDATE tb\_BAOHIEM  SET NGAYCAP = @Ngaycap, NOICAP= @Noicap, NOIDK=@NoiDK  WHERE IDBH = @IDBH;  end  else  print 'Ma Bao Hiem Khong Ton Tai'  end  execute sp\_Update\_BH 4001, '20171204', N'Hồ Chí Minh', N'Bệnh Viện 115' |

+ **Procedure thêm vào bảng ứng lương**: Dùng để thêm nhân viên vào bảng UNGLUONG với thông tin nhập từ bàn phím

|  |
| --- |
| create proc sp\_Insert\_UL  @Nam int,  @Thang int,  @Ngay int,  @Sotien float,  @TrangThai bit,  @Manv char(5)  as  begin  Insert into tb\_UNGLUONG(NAM,THANG,NGAY,SOTIEN,TRANGTHAI, MANV)  Values(@Nam, @Thang,@Nam, @Sotien,@TrangThai,@Manv)  end  execute sp\_Insert\_UL 2022,10,12,4300000,Null,'NV005' |

+ **Procedure xóa hợp đồng nhân viên và nhân viên**: Dùng để xóa nhân viên ở bảng NHANVIEN và hợp đồng nhân viên ở bảng HOPDONG tại vị trí MANV và SOHD tương ứng được nhập từ bàn phím.

|  |
| --- |
| CREATE PROCEDURE Xoa\_HD\_NV (@SOHD nvarchar(20), @MANV char(5))  AS  BEGIN  if exists (select SOHD from tb\_HOPDONG where SOHD=@SOHD)  BEGIN  DELETE FROM tb\_HOPDONG  where SOHD=@SOHD  Delete FROM tb\_NHANVIEN  where MANV=@MANV  END  else  print 'Hop Dong Khong Ton Tai Hoa Da Bi Xoa'  END  execute Xoa\_HD\_NV 'HDLD000031', 'NV032' |

+ **Procedure thêm khen thưởng và kỷ luật** của nhân viên phòng ban điều hành: dùng để cho Trưởng Phòng Ban Điều Hành tiến hành khen thưởng hoặc kỷ luật nhân viên phòng điều hành

|  |
| --- |
| create proc sp\_KT\_NV\_DH  @MANV char(5),  @SOKTKL int,  @NOIDUNG nvarchar(50),  @LOAI int  as  begin  if exists (select MANV from tb\_NHANVIEN where MANV=@MANV and IDPB=2002)  begin  insert into tb\_KHENTHUONG\_KYLUAT(SOKTKL,NOIDUNG,NGAY,MANV,LOAI,THANG,NAM)  values(@SOKTKL,@NOIDUNG,GETDATE(),@MANV,@LOAI,MONTH(GETDATE()),YEAR(GETDATE()))  end  else  print 'Nhan Vien Khong Ton Tai Hoac Khong Thuoc Phong Ban Ban Quan Ly'  end |

+ **Procedure phân công tăng ca phòng ban điều hành**: dùng để cho Trưởng Phòng Ban Điều Hành tiến hành phân công tăng ca cho nhân viên phòng điều hành

|  |
| --- |
| create proc sp\_PC\_TC\_DH  @SOGIO float,  @MANV char(5),  @IDLOAICA int  as  begin  if exists (select MANV from tb\_NHANVIEN where MANV=@MANV and IDPB=2002)  begin  insert into tb\_TANGCA(NAM,THANG,NGAY,SOGIO,MANV,IDLOAICA)  values(YEAR(GETDATE()),MONTH(GETDATE()),DAY(GETDATE()),@SOGIO,@MANV,@IDLOAICA)  end  else  print 'Nhan Vien Khong Ton Tai Hoac Khong Thuoc Phong Ban Ban Quan Ly'  end |

+ **Procedure thêm nhân viên:** dùng để cho trưởng phòng hoặc quản lý thêm nhân viên mới

|  |
| --- |
| create proc sp\_THEM\_NV  @HOTEN nvarchar(50),  @GIOTINH bit,  @NGAYSINH datetime,  @DIENTHOAI nvarchar(50),  @CCCD nvarchar(50),  @DIACHI nvarchar(500),  @IDPB int,  @IDBP int,  @IDCV int,  @IDTD int  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_NHANVIEN with (updlock)  begin try  if exists ( select \* from tb\_NHANVIEN WHERE HOTEN = @HOTEN AND CCCD = @CCCD)  begin  raiserror ('Nhan vien da ton tai, ten va cccd khong duoc trung', 16,1)  end  else  begin  waitfor delay '00:00:20'  insert into tb\_NHANVIEN(HOTEN,GIOITINH,NGAYSINH,DIENTHOAI,CCCD,DIACHI,IDPB,IDBP,IDCV,IDTD)  values(@HOTEN,@GIOTINH,@NGAYSINH,@DIENTHOAI,@CCCD,@DIACHI,@IDPB,@IDBP,@IDCV,@IDTD)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm ứng lương**: dùng để cho trưởng phòng hoặc quản lý thêm vào bảng ứng lương khi nhân viên ứng lương.

|  |
| --- |
| create proc sp\_THEM\_UL  @SOTIEN float,  @MANV char(5)  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_UNGLUONG with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:20'  insert into tb\_UNGLUONG(NAM,THANG,NGAY,SOTIEN,TRANGTHAI,MANV)  values(Year(getdate()),MONTH(getdate()),day(getdate()),@SOTIEN,0,@MANV)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm phụ cấp cho nhân viên**: dùng để cho kế toám thêm vào nhân viên – phụ cấp khi nhân viên được nhận phụ cấp tương ứng.

|  |
| --- |
| create proc sp\_THEM\_NV\_PC  @MANV char(5),  @IDPC int  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_NHANVIEN\_PHUCAP with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:20'  insert into tb\_NHANVIEN\_PHUCAP(MANV,IDPC)  values(@MANV,@IDPC)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm khen thưởng – kỷ luật**: dùng để cho kế toám thêm vào khen thưởng – kỷ luật khi nhân viên được khen thưởng hoặc bị kỷ luật

|  |
| --- |
| create proc sp\_THEM\_KT\_KL  @SOKTKL int,  @NOIDUNG nvarchar(500),  @MANV char(5),  @LOAI int  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_KHENTHUONG\_KYLUAT with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:20'  insert into tb\_KHENTHUONG\_KYLUAT(SOKTKL,NOIDUNG,NGAY,MANV,LOAI,THANG,NAM)  values(@SOKTKL,@NOIDUNG,GETDATE(),@MANV,@LOAI,MONTH(getdate()),year(getdate()))  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thanh toán lương**: dùng để cho kế toán tiến hành thanh toán lương cho nhân viên hàng tháng

|  |
| --- |
| alter proc sp\_TT\_Luong  @MATHANGCONG int,  @ID int  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_THANHTOANLUONG with (readcommitted)  begin try  if exists (select MATHANGCONG from tb\_THANGCONGCHITIET where MATHANGCONG=@MATHANGCONG)  begin  if exists (select ID from tb\_UNGLUONG where ID=@ID and TRANGTHAI=1) or @ID is Null  begin  waitfor delay '00:00:05'  update tb\_UNGLUONG  set TRANGTHAI = 0  where ID=@ID  insert into tb\_THANGCONG(MATHANGCONG)  values(@MATHANGCONG)    insert into tb\_THANHTOANLUONG(ID,MATHANGCONG)  values(@ID,@MATHANGCONG)  end  else  print('Ma IDUL khong tin tai hoac da duoc thanh toan')  end  else  print('Ma Thang Cong Khong Ton Tai')  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm bảo hiểm**: dùng để thêm thông tin bảo hiểm của nhân viên

|  |
| --- |
| create proc sp\_THEM\_BH  @NOICAP nvarchar(50),  @NOIDK nvarchar(50),  @MANV char(5)  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_BAOHIEM with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:05'  insert into tb\_BAOHIEM(NGAYCAP,NOICAP,NOIDK,MANV)  values(GETDATE(),@NOICAP,@NOIDK,@MANV)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm tăng ca**: dùng để thêm vào bảng TangCa khi nhân viên được phân công tăng ca

|  |
| --- |
| create proc sp\_THEM\_TC\_CHUNG  @SOGIO float,  @MANV char(5),  @IDLC int  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_BAOHIEM with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:05'  insert into tb\_TANGCA(NAM,THANG,NGAY,SOGIO,MANV,IDLOAICA)  values(YEAR(GETDATE()), MONTH(GETDATE()),DAY(GETDATE()),@SOGIO,@MANV,@IDLC)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

+ **Procedure thêm tháng công chi tiết**: dùng để Trưởng Phòng thêm vào bảng ThangCongChiTiet khi trưởng phòng chấm công của nhân viên

|  |
| --- |
| create proc sp\_THEM\_TCCT  @MATC int,  @MANV char(5),  @N1 nvarchar(10),  @N2 nvarchar(10),  @N3 nvarchar(10),  @N4 nvarchar(10),  @N5 nvarchar(10),  @N6 nvarchar(10),  @N7 nvarchar(10),  @N8 nvarchar(10),  @N9 nvarchar(10),  @N10 nvarchar(10),  @N11 nvarchar(10),  @N12 nvarchar(10),  @N13 nvarchar(10),  @N14 nvarchar(10),  @N15 nvarchar(10),  @N16 nvarchar(10),  @N17 nvarchar(10),  @N18 nvarchar(10),  @N19 nvarchar(10),  @N20 nvarchar(10),  @N21 nvarchar(10),  @N22 nvarchar(10),  @N23 nvarchar(10),  @N24 nvarchar(10),  @N25 nvarchar(10),  @N26 nvarchar(10),  @N27 nvarchar(10),  @N28 nvarchar(10),  @N29 nvarchar(10),  @N30 nvarchar(10),  @N31 nvarchar(10)  as  begin  set xact\_abort on  begin tran  set transaction isolation level serializable  select \* from tb\_BAOHIEM with (updlock)  begin try  if not exists ( select MANV from tb\_NHANVIEN WHERE MANV=@MANV)  begin  raiserror ('Nhan vien khong ton tai', 16,1)  end  else  begin  waitfor delay '00:00:05'  insert into tb\_THANGCONGCHITIET(MATHANGCONG,MANV,N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,  N11,N12,N13,N14,N15,N16,N17,N18,N19,N20,N21,N22,N23,N24,N25,N26,N27,N28,N29,N30,N31)  values(@MATC,@MANV,@N1,@N2,@N3,@N4,@N5,@N6,@N7,@N8,@N9,@N10,  @N11,@N12,@N13,@N14,@N15,@N16,@N17,@N18,@N19,@N20,@N21,@N22,@N23,@N24,@N25,@N26,@N27,@N28,@N29,@N30,@N31)  end  commit  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = 'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR (@errorMessage, 16,1)  end catch    end |

### Mô tả Trigger

#### TRIGGER: trg\_KiemTra\_Tuoi\_NhanVien

Mục Tiêu: dùng để kiểm tra tuổi nhân viên, nếu nhân viên nhỏ hơn 18 tuổi thig không được vào làm

|  |
| --- |
| create trigger trg\_KiemTra\_Tuoi\_NhanVien  on tb\_NhanVien  instead of insert  as  begin  declare @HOTEN nvarchar(50), @GIOITINH Bit, @NGAYSINH datetime, @DIENTHOAI nvarchar(50), @CCCD nvarchar(50), @DIACHI NVARCHAR(500), @HINHANH varbinary(max),  @IDPB int, @IDBP int, @IDCV int, @IDTD int, @MA\_NQL char(5)  select @HOTEN=HOTEN,@GIOITINH=GIOITINH,@NGAYSINH=NGAYSINH, @DIENTHOAI=DIENTHOAI, @CCCD=CCCD, @DIACHI=DIACHI, @HINHANH=HINHANH, @IDPB=IDPB, @IDBP=IDBP,  @IDCV=IDCV, @IDTD=IDTD, @MA\_NQL=MA\_NQL  from inserted  if (year(getdate()) - year(@NGAYSINH))<18  begin  raiserror ('Nhan Vien Phai Tren 18 Tuoi', 16,1)  rollback tran  return  end  end |

#### TRIGGER: trg\_ThemHD\_NV

Mục Tiêu: tự động thêm vào bảng HOPDONG sau khi một nhân viên mới được thêm vào bảng NHANVIEN.

|  |
| --- |
| create trigger trg\_ThemHD\_NV  on tb\_NHANVIEN  after insert  as  begin  declare @MANV char(5), @NGAYKI datetime, @NGAYBATDAU datetime, @NGAYKETTHUC datetime, @HANHD int  select @MANV=MANV, @NGAYKI=GETDATE(), @NGAYBATDAU=@NGAYKI+5, @HANHD=5, @NGAYKETTHUC=DATEADD(YEAR, @HANHD, @NGAYKI) from inserted  insert into tb\_HOPDONG(NGAYBAYDAU, NGAYKETTHUC, NGAYKY, MANV, HANHD)  values(@NGAYBATDAU,@NGAYKETTHUC, @NGAYKI,@MANV,@HANHD)  end |

#### TRIGGER: trg\_KiemTra\_ThongTinSua

Mục Tiêu: dùng kiểm tra các thông tin nhập vào bảng NHANVIEN nếu các thông tin nhập vào không hợp lệ sẽ tiến hành báo lỗi

|  |
| --- |
| create trigger trg\_KiemTra\_ThongTinSua  on tb\_NHANVIEN  after update  as  begin  declare @old\_Hoten nvarchar(50), @old\_ngaysinh datetime, @old\_dienthoai nvarchar(50), @old\_DiaChi nvarchar(500)  declare @new\_manv char(5),  @new\_Hoten nvarchar(50), @new\_ngaysinh datetime, @new\_dienthoai nvarchar(50), @new\_DiaChi nvarchar(500)  select @old\_Hoten=HOTEN, @old\_ngaysinh=NGAYSINH, @old\_dienthoai=DIENTHOAI, @old\_DiaChi=DIACHI  from deleted    select @new\_Hoten=HOTEN, @new\_ngaysinh=NGAYSINH, @new\_dienthoai=DIENTHOAI,@new\_DiaChi=DIACHI  from inserted  if(UPDATE(MANV))  begin  rollback tran  raiserror('Khong duoc sua ma nhan vien',16,1)  return  end  if(UPDATE(HOTEN) and @old\_Hoten=@new\_Hoten)  begin  rollback tran  raiserror('Ho ten moi phai khac ho ten cu',16,1)  end  if(UPDATE(NGAYSINH) and @old\_ngaysinh=@new\_ngaysinh)  begin  rollback tran  raiserror('Ngay sinh moi phai khac ngay sinh cu',16,1)  end  if(UPDATE(NGAYSINH) and year(getdate())-year(@new\_ngaysinh)<18)  begin  rollback tran  raiserror('Nhan vien phai tren 18 tuoi',16,1)  end  if(UPDATE(DIENTHOAI) and @old\_dienthoai=@new\_dienthoai)  begin  rollback tran  raiserror('So dien thoai moi phai khac so dien thoai cu',16,1)  end  if(UPDATE(DIACHI) and @old\_DiaChi=@new\_DiaChi)  begin  rollback tran  raiserror('Dia chi moi phai khac so dia chi cu',16,1)  end  end |

#### TRIGGER: trg\_KiemTra\_TienUng

Mục Tiêu: kiểm tra thông tin thêm vào bảng ứng lương, một nhân viên không được ứng quá năm triệu một tháng, nếu ứng hơn năm triệu sẽ báo lỗi.

|  |
| --- |
| create trigger trg\_KiemTra\_TienUng  on tb\_UNGLUONG  instead of insert  as  begin  declare @MANV char(5), @Nam int, @Thang int, @Sotien float, @Ngay int, @Trangthai bit  select @MANV=MANV, @Nam=NAM, @Thang=THANG, @Sotien=SOTIEN, @Ngay=NGAY, @Trangthai=TRANGTHAI from inserted  if exists (select NAM, THANG from tb\_UNGLUONG where MANV=@MANV and NAM=@Nam and Thang=@Thang)  begin  raiserror ('Moi Nhan Vien Khong Duoc Ung Qua 1 Lan 1 Thang', 16,1)  rollback tran  return  end  if @Sotien>5000000  begin  raiserror ('Tien Ung Khong Duoc Qua 5000000', 16,1)  rollback tran  return  end  insert into tb\_UNGLUONG(Nam, Thang, NGAY, Sotien, Trangthai, MANV)  values(@Nam, @Thang, @Ngay, @Sotien, @Trangthai, @MANV)  end |

#### TRIGGER: trg\_KiemTra\_BaoHiem

Mục Tiêu: kiểm tra thông tin thêm vào bảng bảo hiểm, một nhân không được có hơn một bảo hiểm, nếu nhân viên có hơn một bảo hiểm sẽ báo lỗi

|  |
| --- |
| alter trigger trg\_KiemTra\_BaoHiem  on tb\_BAOHIEM  instead of insert  as  begin  declare @MANV char(5), @NoiCap nvarchar(50), @NoiDK nvarchar(50)  select @MANV=MANV, @NoiCap=NOICAP, @NoiDK=NOIDK from inserted  if exists (select MANV from tb\_BAOHIEM where MANV=@MANV)  begin  raiserror ('Moi Nhan Vien Chi Co 1 Bao Hiem', 16,1)  rollback tran  return  end  insert into tb\_BAOHIEM(NGAYCAP, NOICAP, NOIDK, MANV)  values(GETDATE(), @NoiCap, @NoiDK, @MANV)  end |

#### 

#### TRIGGER: trg\_HSL\_BIENDONGHSL

Mục Tiêu: khi cập nhật, thêm hoặc xóa trên bảng hệ số lương, thì trên bảng biến động hệ số lương cũng sẽ được cập nhật, thêm hoặc xóa tương ứng

|  |
| --- |
| alter trigger trg\_HSL\_BIENDONGHSL  on tb\_HESOLUONG  after update, insert, delete  as  begin  declare @MAHSL char(4), @BAC int, @NhomNgach char(20), @HeSoLuong Float, @action as char(1)  select @MAHSL=MAHSL, @BAC=BAC, @NhomNgach=NHOMNGACH, @HeSoLuong=HESOLUONG from inserted  SET @action = 'I'; -- Set Action to Insert by default.  IF EXISTS(SELECT \* FROM DELETED)  BEGIN  SET @action =  CASE  WHEN EXISTS(SELECT \* FROM INSERTED) THEN 'U' -- Set Action to Updated.  ELSE 'D' -- Set Action to Deleted.  END  END  ELSE  IF NOT EXISTS(SELECT \* FROM INSERTED) RETURN;  if @action='I'  begin  insert into tb\_BIENDONGHSL(MAHSL ,NAMAPDUNG, BAC, NHOMNGACH, HESOLUONG)  values (@MAHSL,GETDATE(),@BAC, @NhomNgach, @HeSoLuong)  end  if @action='U'  begin  update tb\_BIENDONGHSL  set NAMAPDUNG=Getdate(),BAC=@BAC, NHOMNGACH=@NhomNgach,HESOLUONG=@HeSoLuong  where MAHSL=@MAHSL  end  if @action='D'  begin  select @MAHSL=MAHSL from deleted  delete from tb\_BIENDONGHSL  where MAHSL=@MAHSL  end  end |

#### TRIGGER: trg\_NV\_PV

Mục Tiêu: khi cập nhật hoặc thêm trên bảng phụ cấp, thì trên bảng phụ cấp nhân viên cũng sẽ được cập nhật, hoặc thêm những thông tin tương ứng

|  |
| --- |
| alter trigger trg\_NV\_PC  on tb\_NHANVIEN\_PHUCAP  after insert, update  as  begin  declare @ID int, @IDPC int, @SOTIEN float, @TENPC nvarchar(50)  select @ID=ID, @IDPC=IDPC  from inserted  if not exists (select IDPC from tb\_PHUCAP where IDPC=@IDPC)  BEGIN  RAISERROR('Xin Kiem Tra Lai Thong Tin Nhap', 16, 1)  ROLLBACK TRAN  end  Else  begin  select @SOTIEN=SOTIEN, @TENPC=TENPC  from tb\_PHUCAP  where IDPC=@IDPC  update tb\_NHANVIEN\_PHUCAP  set SOTIEN=@SOTIEN, NGAY=GETDATE(), NOIDUNG=@TENPC  where ID=@ID  end  end |

#### TRIGGER: trg\_BANGCONGCT

Mục Tiêu: khi cập nhật hoặc thêm trên bảng bảng công chi tiết, thì các thông tin tổng ngày công, ngày công không phép, ngày công có phép sẽ được cập nhật tương ứng

|  |
| --- |
| alter trigger trg\_BangCongCT  on tb\_THANGCONGCHITIET  AFTER insert,update  as  begin  declare @MATC int, @MANV char(5), @N1 nvarchar(10),@N2 nvarchar(10),@N3 nvarchar(10),@N4 nvarchar(10),@N5 nvarchar(10),@N6 nvarchar(10),@N7 nvarchar(10),@N8 nvarchar(10),@N9 nvarchar(10),@N10 nvarchar(10),  @N11 nvarchar(10),@N12 nvarchar(10),@N13 nvarchar(10),@N14 nvarchar(10),@N15 nvarchar(10),@N16 nvarchar(10),@N17 nvarchar(10),@N18 nvarchar(10),@N19 nvarchar(10),@N20 nvarchar(10),  @N21 nvarchar(10),@N22 nvarchar(10),@N23 nvarchar(10),@N24 nvarchar(10),@N25 nvarchar(10),@N26 nvarchar(10),@N27 nvarchar(10),@N28 nvarchar(10),@N29 nvarchar(10),@N30 nvarchar(10), @N31 nvarchar(10)  declare @NGHICP int, @NGHIKP int  select @MATC=MATHANGCONG, @MANV=MANV, @N1=N1,@N2=N2,@N3=N3,@N4=N4,@N5=N5,@N6=N6,@N7=N7,@N8=N8,@N9=N9,@N10=N10, @N11=N11, @N12=N12,@N13=N13,  @N14=N14,@N15=N15,@N16=N16,@N17=N17,@N18=N18,@N19=N19,@N20=N20,  @N21=N21,@N22=N22,@N23=N23,@N24=N24,@N25=N25,@N26=N26,@N27=N27,@N28=N28, @N29=N29, @N30=N30,@N31=N31, @NGHICP=0,@NGHIKP=0 from inserted  if @N1='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N1='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N2='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N2='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N3='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N3='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N4='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N4='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N5='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N5='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N6='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N6='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N7='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N7='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N8='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N8='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N9='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N9='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N10='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N10='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N11='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N11='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N12='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N12='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N13='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N13='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N14='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N14='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N15='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N15='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N16='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N16='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N17='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N17='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N18='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N18='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N19='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N19='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N20='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N20='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N21='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N21='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N22='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N22='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N23='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N23='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N24='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N24='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N25='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N25='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N26='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N26='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N27='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N27='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N28='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N28='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N29='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N29='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N30='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N30='V'  begin  set @NGHIKP=@NGHIKP+1  end  if @N31='C'  begin  set @NGHICP=@NGHICP+1  end  else if @N31='V'  begin  set @NGHIKP=@NGHIKP+1  end  SELECT @MATC=MATHANGCONG,@MANV=MANV FROM inserted  update tb\_THANGCONGCHITIET  set NGAYPHEP=@NGHICP, NGHIKHONGPHEP=@NGHIKP  where MANV=@MANV and MATHANGCONG=@MATC    if MONTH(GETDATE())=1 or MONTH(GETDATE())=3 or MONTH(GETDATE())=5 or MONTH(GETDATE())=7 or MONTH(GETDATE())=8 or MONTH(GETDATE())=10 or MONTH(GETDATE())=12  begin  update tb\_THANGCONGCHITIET  set TONGNGAYCONG=31-NGAYPHEP  where MATHANGCONG=@MATC  end  if MONTH(GETDATE())=2  begin  update tb\_THANGCONGCHITIET  set TONGNGAYCONG=28-NGAYPHEP  where MATHANGCONG=@MATC  end  if MONTH(GETDATE())=4 or MONTH(GETDATE())=6 or MONTH(GETDATE())=9 or MONTH(GETDATE())=11  begin  update tb\_THANGCONGCHITIET  set TONGNGAYCONG=30-NGAYPHEP  where MATHANGCONG=@MATC  end    end |

#### TRIGGER: trg\_THANGCONG

Mục Tiêu: khi MATHANGCONG được nhập vào bảng tháng công, thì các thông tin còn lại sẽ được tự động cập nhật

|  |
| --- |
| alter trigger trg\_THANGCONG  on tb\_THANGCONG  after insert,update  as  begin  declare @MATC int, @NGAYCONGTRONGTHANG int  select @MATC=MATHANGCONG  from inserted  select @NGAYCONGTRONGTHANG=NGAYPHEP from tb\_THANGCONGCHITIET where MATHANGCONG=@MATC  update tb\_THANGCONG  set THANG=MONTH(GETDATE()), NAM=YEAR(GETDATE()), NGAYTINHCONG=GETDATE(), NGAYCONGTRONGTHANG=@NGAYCONGTRONGTHANG  where MATHANGCONG=@MATC    end |

#### TRIGGER: trg\_Xoa\_HD\_NV

Mục Tiêu: khi hợp đồng của nhân viên bị xóa thì thông tin của nhân viên trên bảng nhân viên cũng bị xóa.

|  |
| --- |
| alter trigger trg\_Xoa\_HD\_NV  on tb\_HOPDONG  after delete  as  begin  begin try  declare @MANV char(5)  select @MANV=MANV from deleted  delete from tb\_NHANVIEN  where MANV=@MANV  print('Xoa Thanh Cong, NV va HD da duoc xoa')  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = N'Lỗi: ' + ERROR\_MESSAGE() + ' Database da duoc rollback'  RAISERROR (@errorMessage, 16,1)  end catch  end |

#### TRIGGER: trg\_ThemNQL\_NV

Mục Tiêu: khi mã phòng được thêm vào bảng nhân viên thì người quản lý của nhân viên sẽ được tự động tương ứng với trưởng phòng đang quản lý phòng của nhân viên

|  |
| --- |
| alter trigger trg\_ThemNQL\_NV  on tb\_NHANVIEN  after insert,update  as  begin  begin try  declare @MANV char(5), @IDPB int, @MANQL char(5)  select @MANV=MANV, @IDPB=IDPB from inserted  if @IDPB is not null  begin  select @MANQL=TRUONGPHONG from tb\_PHONGBAN where IDPB=@IDPB  update tb\_NHANVIEN  set MA\_NQL=@MANQL  where MANV=@MANV  end  end try  begin catch  rollback tran  declare @errorMessage varchar(2000)  select @errorMessage = N'Lỗi: ' + ERROR\_MESSAGE() + ' Database da duoc rollback'  RAISERROR (@errorMessage, 16,1)  end catch  end |

#### TRIGGER: trg\_KT\_GIO\_TC

Mục Tiêu: khi nhân viên được phân công tăng ca hơn sáu giờ thì sẽ báo lỗi

|  |
| --- |
| create trigger trg\_KT\_GIO\_TC  on tb\_TANGCA  after insert,update  as  begin  declare @SOGIO float  select @SOGIO=SOGIO from inserted  if @SOGIO>6  begin  raiserror ('So Gio Tang Ca Khong Duoc Lon Hon 6', 16,1)  rollback tran  return  end  end |

#### TRIGGER: trg\_KT\_TP\_PB

Mục Tiêu: kiểm tra trưởng phòng được bổ nhiệm quản lý phòng ban, nếu trưởng phòng được bổ nhiệm đã quản lý một phòng ban khác thì sẽ báo lỗi

|  |
| --- |
| alter trigger trg\_KT\_TP\_PB  on tb\_PHONGBAN  instead of insert,update  as  begin  declare @TENPHONGBAN nvarchar(50), @TRUONGPHONG char(5)  select @TENPHONGBAN=TENPB,@TRUONGPHONG=TRUONGPHONG from inserted  if exists (select TRUONGPHONG from tb\_PHONGBAN where TRUONGPHONG= @TRUONGPHONG)  begin  raiserror ('Truong Phong Dang Quan Ly Phong Ban Khac', 16,1)  rollback tran  return  end  else  begin  insert into tb\_PHONGBAN(TENPB,TRUONGPHONG)  values(@TENPHONGBAN,@TRUONGPHONG)  end  end |

#### TRIGGER: tt\_Luong

Mục Tiêu: tự động tính tiền và lương nhân viên khi kế toán nhập vào các thông tin cần thiết

|  |
| --- |
| alter trigger tt\_luong  on tb\_THANHTOANLUONG  after insert,update  as  begin  declare @IDUL int, @MATC int, @MANV char(5), @IDLUONG int, @TIENPC float, @NGAYNGHI int, @LUONGBD float, @UL float, @LUONGCT float(3), @THUETNCN float(3), @Luong float(3)  select @IDUL=ID,@MATC=MATHANGCONG,@IDLUONG=IDLUONG, @MANV=MANV from inserted  select @MANV=MANV from tb\_THANGCONGCHITIET where MATHANGCONG=@MATC  update tb\_THANHTOANLUONG  set MANV=@MANV  where IDLUONG=@IDLUONG  select @NGAYNGHI=TONGNGAYCONG from tb\_THANGCONGCHITIET where MATHANGCONG=@MATC and MANV=@MANV    select @TIENPC=0,@LUONGCT=0,@THUETNCN=0,@Luong=0, @UL=0    if exists (select MANV from tb\_NHANVIEN\_PHUCAP where MANV=@MANV)  begin  select @TIENPC=SOTIEN from tb\_NHANVIEN\_PHUCAP where MANV=@MANV  end  if exists (select ID from tb\_UNGLUONG where ID=@IDUL)  begin  select @UL=SOTIEN from tb\_UNGLUONG where ID=@IDUL  end  if exists (select MANV from tb\_HOPDONG where MANV=@MANV)  begin  select @LUONGBD=LUONGBD from tb\_HOPDONG where MANV=@MANV  if MONTH(GETDATE())=1 or MONTH(GETDATE())=3 or MONTH(GETDATE())=5 or MONTH(GETDATE())=7 or MONTH(GETDATE())=8 or MONTH(GETDATE())=10 or MONTH(GETDATE())=12  begin  set @LUONGCT=(((@LUONGBD/31)\*@NGAYNGHI)+@TIENPC)  end  if MONTH(GETDATE())=2  begin  set @LUONGCT=(((@LUONGBD/28)\*@NGAYNGHI)+@TIENPC)  end  if MONTH(GETDATE())=4 or MONTH(GETDATE())=6 or MONTH(GETDATE())=9 or MONTH(GETDATE())=11  begin  set @LUONGCT=(((@LUONGBD/30)\*@NGAYNGHI)+@TIENPC)  end  set @THUETNCN = case  when @LUONGCT<=5000000 then @LUONGCT\*0.05  when @LUONGCT>5000000and @LUONGCT<=10000000 then (@LUONGCT\*0.1)-250000  when @LUONGCT>10000000 and @LUONGCT<=18000000 then (@LUONGCT\*0.15)-750000  when @LUONGCT>18000000 and @LUONGCT<=32000000 then (@LUONGCT\*0.2)-1650000  when @LUONGCT>32000000 and @LUONGCT<=52000000 then (@LUONGCT\*0.25)-3250000  when @LUONGCT>52000000 and @LUONGCT<=80000000 then (@LUONGCT\*0.3)-5850000  when @LUONGCT>80000000 then (@LUONGCT\*0.35)-9850000  end  set @Luong=@LUONGCT-@THUETNCN-@UL    update tb\_THANHTOANLUONG  set THANGTT=MONTH(GETDATE()),NAMTT=YEAR(GETDATE()),MANV=@MANV,LUONGCT=@LUONGCT,THUETNCN=@THUETNCN,LUONG=@Luong  where IDLUONG=@IDLUONG  end  else  begin  raiserror ('Nhan Vien Khong Ton Tai Hoac Khong Co Hop Dong', 16,1)  rollback tran  return  end  end |

#### TRIGGER: tt\_KT\_DL\_TC

Mục Tiêu: nếu một mã tháng công được thống kê hai lần trong bảng ThangCong thì sẽ báo lỗi

|  |
| --- |
| alter trigger trg\_KT\_SL\_TC  on tb\_THANGCONG  instead of insert, update  as  begin  declare @ID int,@MATHANGCONG int, @MATHANGCONG2 int  select @MATHANGCONG=MATHANGCONG, @ID=ID from inserted  if @MATHANGCONG in (select MATHANGCONG from tb\_THANGCONG)  begin  raiserror ('Ma Thang Cong Bi Trung', 16,1)  rollback tran  return  end  else  begin  insert into tb\_THANGCONG(MATHANGCONG)  values (@MATHANGCONG)  end  end |

### Phân Quyền

#### Tạo Role

|  |
| --- |
| create role TruongPhongDieuHanh  create role KeToan  create role NhanVien005 |

#### Phân Quyền Cho Role

|  |
| --- |
| CREATE VIEW [XEMTTNV\_DH] AS  SELECT \*  FROM tb\_NHANVIEN  WHERE IDPB=2002  --Trưởng Phòng:  select \* from XEMTTNV\_DH  grant select on XEMTTNV\_DH to TruongPhongDieuHanh  grant select on tb\_KHENTHUONG\_KYLUAT to TruongPhongDieuHanh  grant execute on sp\_KT\_NV\_DH to TruongPhongDieuHanh  grant select on tb\_TANGCA to TruongPhongDieuHanh  grant execute on sp\_PC\_TC\_DH to TruongPhongDieuHanh  grant select on XEMCNV\_DH to TruongPhongDieuHanh  grant insert, update on tb\_THANGCONGCHITIET to TruongPhongDieuHanh  grant select on tb\_HESOLUONG to TruongPhongDieuHanh  grant select on XEMBHNV\_DH to TruongPhongDieuHanh  grant select on tb\_CHUCVU TO TruongPhongDieuHanh  grant execute on sp\_THEM\_NV to TruongPhongDieuHanh  grant select on tb\_PHONGBAN to TruongPhongDieuHanh  grant select on tb\_BOPHAN to TruongPhongDieuHanh  grant select on tb\_LOAICA to TruongPhongDieuHanh  grant select on tb\_PHUCAP to TruongPhongDieuHanh  grant select on tb\_UNGLUONG to TruongPhongDieuHanh  grant select on XEMLUONGNV\_DH to TruongPhongDieuHanh  --Kế toán:Xem chấm công NV, xem khen thưởng ký luật nhân viên,Xem phụ cấp, xem lương bản đầu và mã nhân viên trong bảng hợp đồng  --Được thêm, cập nhật trong bảng thanh toán lương  grant select on tb\_HESOLUONG to KeToan  grant select on tb\_BAOHIEM to KeToan  grant select on tb\_BIENDONGHSL to KeToan  grant select, insert, update on tb\_THANGCONG to KeToan  grant select on tb\_HOPDONG to KeToan  grant execute on SP\_TC\_THONGTINNV to KeToan  grant execute on sp\_THEM\_UL to KeToan  grant execute on sp\_THEM\_NV\_PC to KeToan  grant execute on sp\_THEM\_KT\_KL to KeToan  grant select on tb\_LOAICA to KeToan  grant select on tb\_THANGCONGCHITIET to KeToan  grant select on tb\_PHUCAP to KeToan  grant select on tb\_TANGCA to KeToan  --Nhân viên xem được thông tin của mình  grant select on XEMTT\_BH\_NV005 to NhanVien005  grant select on XEMTT\_NV\_NV005 to NhanVien005  grant select on XEMTT\_UL\_NV005 to NhanVien005  grant select on XEMTT\_PC\_NV005 to NhanVien005  grant select on XEMTT\_KT\_KL\_NV005 to NhanVien005  grant select on XEMTT\_TC\_NV005 to NhanVien005  grant select on XEMTT\_TTL\_NV005 to NhanVien005 |

#### Tạo Login và Gán Role Cho User

|  |
| --- |
| --TruongPhong  create LOGIN Khoa with Password='123456' must\_change,  check\_expiration = ON  create user Khoa for Login Khoa  exec sp\_addrolemember TruongPhongDieuHanh, Khoa  --Ketoan  create LOGIN PTan with Password='123456' must\_change,  check\_expiration = ON  create user PTan for Login PTan  exec sp\_addrolemember KeToan, PTan  --Nhanvien  create LOGIN Hung with Password='123456' must\_change,  check\_expiration = ON  create user Hung for Login Hung  exec sp\_addrolemember NhanVien005, Hung |

## **Phía Client**

### Form NHANVIEN

Graphical user interface, text, application

Description automatically generated

Hình 11: Danh sách nhân viên

+ Thêm nhân viên:

Graphical user interface, text, application, email

Description automatically generated

Hình 12: Thêm nhân viên

### Form HOPDONG

Graphical user interface, application

Description automatically generated

Hình 13: Danh sách hợp đồng

+ Thêm hợp đồng:

Graphical user interface, text, application

Description automatically generated

Hình 14: Thêm hợp đồng

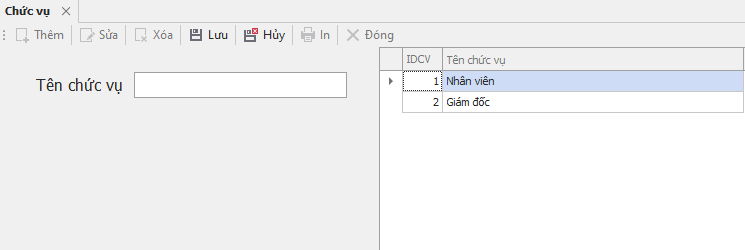
### Form CHUCVU

Graphical user interface, text, application, email

Description automatically generated

Hình 15: Danh sách chức vụ

+ Thêm chức vụ:



Hình 16: Thêm chức vụ

### Form BOPHAN

Graphical user interface, text, application, email

Description automatically generated

Hình 17: Danh sách bộ phận

+ Thêm bộ phận:

Graphical user interface, application, Word

Description automatically generated

Hình 18: Thêm bộ phận

### Form PHONGBAN

Graphical user interface, application

Description automatically generated

Hình 19: Danh sách phòng ban

+ Thêm phòng ban:

Graphical user interface, application, Word

Description automatically generated

Hình 20: Thêm phòng ban

### Form TRINHDO

Graphical user interface, text, application

Description automatically generated

Hình 21: Danh sách trình độ

+ Thêm trình độ:

Graphical user interface, application

Description automatically generated

Hình 22: Thêm trình độ

# **Chương 6: Kết Luận**

## **Kết Quả Đạt Được**

### Phía Server

* Xây dựng cấu trúc dữ liệu phù hợp với yêu cầu của người dùng.
* Xây dựng các ràng buộc toàn vẹn nhằm giảm thiểu sai sót nhập liệu của người dùng.
* Xây dựng các stored procedure, trigger và function đáp ứng yêu cầu truy vấn và giao tác của người dùng, đảm bảo việc xử lý giao tác được thực hiện nhanh và có thể xử lý đồng thời.
* Phân quyền trên database.
* Lập lịch backup tự động cho database.

### Phía Client

Xây dựng ứng dụng được các chức năng của hệ thống quản lý lương nhân viên

## **Kết Quả Chưa Đạt Được**

* Mã hóa dữ liệu để tăng tính bảo mật
* Chưa hoàn chỉnh toàn bộ chức năng quản lý của một doanh nghiệp.
* Các chức năng cần cải thiện hơn.

+ **Hướng phát triển tương lai**

* Mã hóa dữ liệu để tăng tính bảo mật.
* Thiết kế giao diện bắt mắt hơn.
* Hoàn thành đưa toàn bộ trigger, procedure, function… lên giao diện.
* Tối ưu hóa các quy trình nghiệp vụ để áp dụng cho doanh nghiệp thực tế.

## Tài liệu tham khảo

|  |  |
| --- | --- |
| [1] | "W3Schools Online Web Tutorials," [Online]. Available: https://www.w3schools.com/. |
| [2] | "Stack Over Flow - Where Developers Learn, Share, & Build Careers," [Online]. Available: https://stackoverflow.com/. |
| [3] | "Microsoft Learn: Build skills that open doors in your career," [Online]. Available: https://learn.microsoft.com/en-us/. |
| [4] | "Learn to Code - For Free - Coding Course for Busy People," [Online]. Available: https://www.freecodecamp.org/. |
| [5] | "Server-level roles - SQL Server | Microsoft Learn," 19 11 2022. [Online]. Available: https://learn.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles?view=sql-server-ver16&fbclid=IwAR1zIy7S\_SlcCdpu2-XBwHNSHZhHMBX3m4b3GdpLOIYYAfJMmrJbTwMiOYo. |

## **Phân công**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Nguyễn Anh Khoa** | **Đỗ Phúc Tân** | **Hà Bùi Mạnh Hùng** |
| **Khảo sát nghiệp vụ** | X | X | X |
| **Xác định các yêu cầu cho đề tài** | X | X | X |
| **Thiết kế lược đồ Use case** | X | X |  |
| **Phân tích hệ thống** | X | X | X |
| **Phân tích ràng buộc toàn vẹn** | X | X | X |
| **Xây dựng các ràng buộc toàn vẹn, các thủ tục và giao tác trên server** | X | X | X |
| **Xây dựng các phương thức bảo mật trên server** | X | X |  |
| **Thiết kế và xây dựng phân quyền trên server** | X | X |  |
| **Xây dựng lịch tự động** | X | X |  |
| **Thiết kế và xây dựng giao diện cho client** |  |  | X |
| **Thiết kế và xây dựng chức năng cho client** |  |  | X |
| **Test hệ thống** | X | X | X |
| **Viết báo cáo** | X | X | X |