TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO BÀI TẬP LỚN MÔN HỆ CƠ SỞ DỮ LIỆU**

*Người hướng dẫn*: **GV. DZOÃN XUÂN THANH**

*Người thực hiện*: **TRẦN THỊ VẸN - 53200674**

Lớp **: 21050301**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO BÀI TẬP LỚN MÔN HỆ CƠ SỞ DỮ LIỆU**

*Người hướng dẫn*: **GV. DZOÃN XUÂN THANH**

*Người thực hiện*: **TRẦN THỊ VẸN - 53200674**

Lớp **: 21050301**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

LỜI CẢM ƠN

Lời đầu tiên, em xin chân thành gửi lời cảm ơn và sự tri ân sâu sắc đến với các thầy cô, giảng viên của khoa Công nghệ thông tin nói chung và các thầy cô môn Cơ sở dữ liệu nói riêng. Trong suốt quá trình học tập và rèn luyện, em đã nhận được rất nhiều sự giúp đỡ tận tình, sự quan tâm, chăm sóc của các thầy cô.

Ngoài ra, em còn được thầy Dzoãn Xuân Thanh, thầy đã dạy bảo những kiến thức, phương pháp mới về SQL hay ho và thú vị, thầy còn giúp sinh viên có được nhiều niềm vui trong việc học và cảm thấy thoải mái, … Em xin cảm ơn các thầy cô rất nhiều trong suốt quá trình học tập này ạ!!!

Bởi lượng kiến thức của em còn hạn hẹp và gặp nhiều vấn đề trong quá trình học nên báo cáo giữa kỳ này sẽ còn nhiều thiếu sót và cần được học hỏi thêm. Em rất mong em sẽ nhận được sự góp ý của quý thầy cô về bài báo cáo này của em để em rút kinh nghiệm trong những môn học sắp tới. Cuối cùng, em xin chân thành cảm ơn quý thầy cô.

TP Hồ Chí Minh, ngày 15 tháng 11 năm 2022

Sinh viên: Trần Thị Vẹn

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi và được sự hướng dẫn của GV. Dzoãn Xuân Thanh. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 15 tháng 11 năm 2022*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Trần Thị Vẹn*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Trong bài báo cáo này tôi sẽ áp dụng các kiến thức mà giảng viên đã cung cấp trong quá trình giảng dạy để giải quyết các câu hỏi cũng như vấn đề đã được đặt ra trong đề báo cáo . Thêm vào đó, sử dụng phần mềm Microsoft SQL server management studio để hỗ trợ bài báo cáo này hoàn thiện và chính xác hơn.

Bài báo cáo gồm 2 phần về những kiến thức SQL và cả kiến thức về Python hay Java nên đòi hỏi sinh viên cần phải vững nắm vững kiến thức để có thể suy luận và kết hợp lại với nhau nên đôi khi sẽ khó tránh khỏi sai sót nhưng tôi sẽ cố gắng hoàn thành thật tốt đề tài này thật tốt.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc120370562)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc120370563)

[TÓM TẮT iv](#_Toc120370564)

[MỤC LỤC 1](#_Toc120370565)

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 2](#_Toc120370566)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 3](#_Toc120370567)

[1. PHẦN I 4](#_Toc120370568)

[1.1. Viết đặc tả theo yêu cầu 4](#_Toc120370569)

[1.1.1. Yêu cầu đề bài 4](#_Toc120370570)

[1.1.2. Thực hiện yêu cầu 4](#_Toc120370571)

[1.2. Vẽ mô hình ERD 5](#_Toc120370572)

[1.2.1. Yêu cầu đề bài 6](#_Toc120370573)

[1.2.2. Thực hiện yêu cầu 6](#_Toc120370574)

[1.3. Chuyển đổi sang mô hình quan hệ 6](#_Toc120370575)

[1.3.1. Yêu cầu đề bài 6](#_Toc120370576)

[1.3.2. Thực hiện yêu cầu 6](#_Toc120370577)

[1.4. Câu lệnh SQL 7](#_Toc120370578)

[1.4.1. Yêu cầu đề bài 7](#_Toc120370579)

[1.4.2. Thực hiện yêu cầu 8](#_Toc120370580)

[2. PHẦN II 18](#_Toc120370581)

[1.1. Câu 1 18](#_Toc120370582)

[1.1.1. Yêu cầu đề bài 19](#_Toc120370583)

[1.1.2. Thực hiện yêu cầu 19](#_Toc120370584)

[1.2. Câu 2 20](#_Toc120370585)

[1.2.1. Yêu cầu đề bài 20](#_Toc120370586)

[1.2.2. Thực hiện yêu cầu 20](#_Toc120370587)

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

**CÁC KÝ HIỆU**

**CÁC CHỮ VIẾT TẮT**

NV: nhân viên

PB: phòng ban

DM: danh mục

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

[Hình 1: Mô hình ERD 6](#_Toc120370104)

[Hình 2: Mô hình quan hệ 7](#_Toc120370105)

[Hình 3: Function MaPB trước khi execute 13](#_Toc120370106)

[Hình 4:Function MaPB trước và sau khi execute 14](#_Toc120370107)

[Hình 5: Function MaNV trước khi execute 15](#_Toc120370108)

[Hình 6: Trước và sau khi execute MANV 16](#_Toc120370109)

[Hình 7: Trigger check miền giá trị khi insert đúng điều kiện 18](#_Toc120370110)

[Hình 8: Trigger check miền giá trị khi insert sai điều kiện 18](#_Toc120370111)

[Hình 9: Nội dung trong file input.txt của câu 1 20](#_Toc120370112)

[Hình 10: Nội dung file output.txt của câu 1 20](#_Toc120370113)

[Hình 11: Nội dung file input.txt của câu 2 23](#_Toc120370114)

[Hình 12: Nội dung trong file output.txt của câu 2 23](#_Toc120370115)

**DANH MỤC BẢNG**

1. PHẦN I
   1. Viết đặc tả theo yêu cầu
      1. Yêu cầu đề bài

Sinh viên tự viết đặc tả cho một nghiệp vụ nào đó. Đặc tả phải có mối quan hệ 1-n, quan hệ n-n, quan hệ cha – con. Có thể có mối quan hệ 1-1, quan hệ giữa thực thhể mạnh và thực thể yếu.

Chứa các mối quan hệ:

a. Quan hệ 1-n: 0.5 điểm

b. Quan hệ n-n: 0.5 điểm

c. Quan hệ cha con: 0.5 điểm

d. Quan hệ 1-1: 0.25 điểm

e. Quan hệ giữa thực thể mạnh và yếu: 0.25 điểm

* + 1. Thực hiện yêu cầu

Trong thời đại công nghệ phát triển mạnh mẽ như hiện nay bắt buộc các doanh nghiệp phải có các chiến lược thay đổi và áp dụng công nghệ vào trong hoạt động quản lý doanh nghiệp để nâng cao khả năng cạnh tranh trên thị trường. Đặc biệt rất nhiều doanh nghiệp đã xây dựng nên các [hệ thống quản lý nhân sự](https://bizfly.vn/techblog/he-thong-quan-ly-nhan-su.html) toàn diện từ đó nâng cao chất lượng nhân viên và tạo ra lợi nhuận khổng lồ. Và một doanh nghiệp cũng cần xây dựng một hệ thống quản lý bằng cấp và lương thưởng của nhân viên để tiện cho việc thu thập thông tin nhân sự, lưu trữ và sử dụng thông tin một cách chính xác, tùy theo từng năng lực làm việc và kinh nghiệm mà một hệ thống quản lý nhân sự sẽ phân chia công việc và mức lương thưởng một cách hợp lý. Hệ thống này có những hoạt động sau đây:

Doanh nghiệp này có rất nhiều nhân viên với nhiều trình độ và nhiều ngành nghề khác nhau nhưng mỗi nhân viên chỉ làm việc ở một phòng ban duy nhất, trong một phòng ban thì có nhiều nhân viên khác nhau. Mỗi nhân viên được định danh bởi một mã nhân viên các mã nhân viên này khác nhau và là duy nhất, ngoài ra mỗi nhân viên còn có họ, tên đệm, tên, số điện thoại, địa chỉ, ngày sinh. Mỗi phòng ban cũng có đặc trưng riêng như mã phòng ban, tên phòng ban, địa chỉ phòng ban (theo cơ cấu tầng 1 đến tầng 5).

Ngoài ra doanh nghiệp này còn chia nhân viên ra thành 3 nhóm nhân viên: nhân viên toàn thời gian (fulltime), nhân viên bán thời gian (part time) và nhân viên tư vấn. Cụ thể thì nhân viên toàn thời gian lương được tính theo lương hàng năm, định kì mỗi năm lương thay đổi một lần, còn nhân viên bán thời gian lương được tính theo thời gian làm và trả công theo giờ, nhân viên tư vấn để trả lương theo cấp bậc tư vấn nếu tư vấn và thuyết phục được nhiều khách hàng thì được nhận lương cao hơn. Mỗi nhân viên khi vào công ty sẽ có một mức lương và tiền thưởng, mức độ thưởng theo cấp bậc từ thấp đến cao khác nhau (từ 1 đến 5). Để dễ quản lý về thông tin cá nhân cũng như lý lịch, gia đình thì doanh nghiệp cũng có một mục lưu trữ thông tin quan hệ gia đình của nhân viên, một nhân viên có thể có nhiều người thân nhưng một người thân chỉ có duy nhất một nhân viên trong doanh nghiệp này. Thông tin của một một người thân bao gồm số thứ tự để biết xem nhân viên đó có bao nhiêu người thân và tên người thân, số điện thoại để liên lạc, ngày sinh của thân nhân đó.

Để vào doanh nghiệp thì phải có bằng cấp và kĩ năng cũng như trình độ của nhân viên phải đạt yêu cầu của doanh nghiệp. Mỗi nhân viên có thể có nhiều bằng cấp về nhiều ngành nghề và trình độ học vấn khác nhau, trong các ngành nghề sẽ có mã ngành (là duy nhất không được trùng lặp), tên ngành, ngoài ra về trình độ của nhân viên cũng có mã trình độ là duy nhất, tên trình độ. Về bằng cấp của nhân viên ngoài lưu thông tin về trình độ, mã ngành thì còn có thông tin như ngày cấp bằng và loại bằng. Bằng cấp là vậy nhưng doanh nghiệp còn muốn tìm hiểu thêm về kĩ năng ngoại ngữ của nhân viên nên doanh nghiệp cũng lưu trữ thông tin về kĩ năng như: mã kĩ năng, trên kĩ năng, mô tả. Khi lưu các thông tin kĩ năng này lại doanh nghiệp còn bổ sung mức độ thành thạo của nhân viên (từ 1 đến 5) nữa để biết rõ hơn phân công nghiệp vụ.

* 1. Vẽ mô hình ERD
     1. Yêu cầu đề bài

Vẽ mô hình ERD cho đặc tả (1 điểm)

* + 1. Thực hiện yêu cầu

Diagram

Description automatically generated

Hình 1: Mô hình ERD

* 1. Chuyển đổi sang mô hình quan hệ
     1. Yêu cầu đề bài

Chuyển mô hình ERD sang mô hình quan hệ (1 điểm).

* + 1. Thực hiện yêu cầu

Diagram

Description automatically generated

Hình 2: Mô hình quan hệ

* 1. Câu lệnh SQL
     1. Yêu cầu đề bài

Sử dụng câu lệnh SQL để thực hiện các thao tác sau:

a. Tạo các bảng cho mô hình quan hệ trên (0.5 điểm)

b. Tạo ít nhất 2 function để thêm dữ liệu vào các bảng trên. (1 điểm)

i. Trong đó, khoá chính của các bảng phải được phát sinh tự động theo một quy tắc nào đó. Ví dụ Mô hình quan hệ có bảng Sinh viên, khoá chính là Mã số sinh viên gồm 2 ký tự đầu là “SV”, 4 ký tự sau là số (thể hiện số thứ tự). Ví dụ Bảng sinh viên đang có Mã số sinh viên lớn nhất là “SV1010” thì khi thêm sinh viên mới sẽ có mã số là “SV1011”.

ii. Sinh viên tự đặt ra quy tắc phát sinh

iii. KHÔNG được sử dụng Identity trong SQL Server để phát sinh tự động.

c. Tạo ít nhất một trigger để kiểm tra ràng buộc khoá ngoại, ràng buộc miền giá trị.(0.5 điểm)

* + 1. Thực hiện yêu cầu

1. **Tạo các bảng cho mô hình quan hệ trên**

USE MASTER

GO

DROP DATABASE IF EXISTS TRANTHIVEN\_52100674\_ASSIGNMENT

GO

CREATE DATABASE TRANTHIVEN\_52100674\_ASSIGNMENT

GO

USE TRANTHIVEN\_52100674\_ASSIGNMENT

GO

IF OBJECT\_ID('DM\_NHANVIEN') IS NOT NULL

  DROP TABLE DM\_NHANVIEN

IF OBJECT\_ID('DM\_PHONGBAN') IS NOT NULL

  DROP TABLE DM\_PHONGBAN

IF OBJECT\_ID('DM\_BANGCAP') IS NOT NULL

  DROP TABLE DM\_BANGCAP

IF OBJECT\_ID('DM\_TRINHDO') IS NOT NULL

  DROP TABLE DM\_TRINHDO

IF OBJECT\_ID('DM\_NGANH') IS NOT NULL

  DROP TABLE DM\_NGANH

IF OBJECT\_ID('DM\_THUONG') IS NOT NULL

  DROP TABLE DM\_THUONG

IF OBJECT\_ID('DM\_QHGIADINH') IS NOT NULL

  DROP TABLE DM\_QHGIADINH

IF OBJECT\_ID('DM\_KINANG') IS NOT NULL

  DROP TABLE DM\_KINANG

IF OBJECT\_ID('DM\_KINANG\_NV') IS NOT NULL

  DROP TABLE DM\_KINANG\_NV

IF OBJECT\_ID('NV\_FULLTIME') IS NOT NULL

  DROP TABLE NV\_FULLTIME

IF OBJECT\_ID('NV\_PARTTIME') IS NOT NULL

  DROP TABLE NV\_PARTTIME

IF OBJECT\_ID('NV\_TUVAN') IS NOT NULL

  DROP TABLE NV\_TUVAN

GO

--CREATE TABLE

CREATE TABLE DM\_NHANVIEN(

    MANV CHAR(6),

    HO VARCHAR (50),

    TENDEM VARCHAR (50),

    TEN VARCHAR (50),

    SODIENTHOAI VARCHAR(20),

    DIACHI VARCHAR(50),

    NGAYSINH DATE,

    MAPB CHAR(6),

    CONSTRAINT PK\_NHANVIEN PRIMARY KEY (MANV)

)

GO

CREATE TABLE DM\_PHONGBAN(

    MAPB CHAR(6),

    TENPB VARCHAR(50),

    DIACHIPB INT NOT NULL,

    CONSTRAINT PK\_PHONGBAN PRIMARY KEY (MAPB)

)

GO

CREATE TABLE DM\_BANGCAP(

    LOAIBANG VARCHAR(50),

    NGAYCAPBANG DATE,

    MANV CHAR(6),

    MATRINHDO VARCHAR(50),

    MANGANH CHAR(6)

)

GO

CREATE TABLE DM\_TRINHDO(

    MATRINHDO VARCHAR(50),

    TENTRINHDO VARCHAR(50),

    CONSTRAINT PK\_TRINHDO PRIMARY KEY (MATRINHDO)

)

GO

CREATE TABLE DM\_NGANH(

    MANGANH CHAR(6),

    TENNGANH VARCHAR(50),

    CONSTRAINT PK\_NGANH PRIMARY KEY (MANGANH)

)

GO

CREATE TABLE DM\_THUONG(

    TIENTHUONG INT NOT NULL,

    MUCDOTHUONG INT NOT NULL,

    MANV CHAR(6),

)

GO

CREATE TABLE DM\_QHGIADINH(

    STT INT NOT NULL,

    TENNGUOIQH VARCHAR(50),

    DIACHI VARCHAR(50),

    NGAYSINH DATE,

    MANV CHAR(6),

)

GO

CREATE TABLE DM\_KINANG(

    MAKINANG CHAR(6),

    TENKINANG VARCHAR (50),

    MOTA VARCHAR (50),

    CONSTRAINT PK\_KINANG PRIMARY KEY (MAKINANG)

)

GO

CREATE TABLE DM\_KINANG\_NV(

    MUCDO INT NOT NULL,

    MANV CHAR(6),

    MAKINANG CHAR(6),

)

GO

CREATE TABLE NV\_FULLTIME(

    MANV CHAR(6),

    LUONGNAM INT,

)

GO

CREATE TABLE NV\_PARTTIME(

    MANV CHAR(6),

    LUONGGIO INT,

)

GO

CREATE TABLE NV\_TUVAN(

    MANV CHAR(6),

    LUONGTHEOCAPBAC INT,

)

GO

--ADD FOREIGN KEY

--DM\_NHANVIEN

ALTER TABLE DM\_NHANVIEN ADD CONSTRAINT FK\_PHONGBAN FOREIGN KEY (MAPB) REFERENCES DM\_PHONGBAN(MAPB)

GO

--DM\_BANGCAP

ALTER TABLE DM\_BANGCAP ADD CONSTRAINT FR\_NHANVIEN FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

ALTER TABLE DM\_BANGCAP ADD CONSTRAINT FR\_TRINHDO FOREIGN KEY (MATRINHDO) REFERENCES DM\_TRINHDO(MATRINHDO)

ALTER TABLE DM\_BANGCAP ADD CONSTRAINT FR\_NGANH FOREIGN KEY (MANGANH) REFERENCES DM\_NGANH(MANGANH)

GO

--DM\_THUONG

ALTER TABLE DM\_THUONG ADD CONSTRAINT FR\_NHANVIEN\_THUONG FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

--DM\_QHGIADINH

ALTER TABLE DM\_QHGIADINH ADD CONSTRAINT FR\_NHANVIEN\_GD FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

--NV FULLTIME, PARTTIME, NV\_TUVAN

ALTER TABLE NV\_FULLTIME ADD CONSTRAINT FR\_NHANVIEN\_FULL FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

ALTER TABLE NV\_PARTTIME ADD CONSTRAINT FR\_NHANVIEN\_PART FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

ALTER TABLE NV\_TUVAN ADD CONSTRAINT FR\_NHANVIEN\_TV FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

--DM\_KINANG\_NV

ALTER TABLE DM\_KINANG\_NV ADD CONSTRAINT FR\_KINANG FOREIGN KEY (MAKINANG) REFERENCES DM\_KINANG(MAKINANG)

ALTER TABLE DM\_KINANG\_NV ADD CONSTRAINT FR\_NHANVIEN\_KN FOREIGN KEY (MANV) REFERENCES DM\_NHANVIEN(MANV)

--ADD CHECK\_ DIA CHI PHONG BAN VA LUONGTHEOCAPDO

ALTER TABLE DM\_PHONGBAN

ADD CONSTRAINT CHK\_DIACHIPB CHECK (DIACHIPB >= 1 AND DIACHIPB <=5);

ALTER TABLE DM\_THUONG

ADD CONSTRAINT CHK\_MUCDOTHUONG CHECK (MUCDOTHUONG >= 1 AND MUCDOTHUONG <=5);

ALTER TABLE DM\_KINANG\_NV

ADD CONSTRAINT CHK\_MUCDO CHECK (MUCDO >= 1 AND MUCDO <=5);

**b. Tạo ít nhất 2 function để thêm dữ liệu vào các bảng trên**

**Function 1: Hàm tự động insert mã phòng ban:**

--FUNCTION 1

DROP FUNCTION IF EXISTS MA\_PB

GO

CREATE FUNCTION MA\_PB()

RETURNS CHAR(6)

AS

BEGIN

    DECLARE @ID\_PB CHAR(2)

    SET @ID\_PB = 'PB'

    DECLARE @SLPB INT

    SET @SLPB  = 1+ (SELECT COUNT(MAPB) FROM DM\_PHONGBAN)

    IF (@SLPB < 10)

        RETURN @ID\_PB + '000' + CAST(@SLPB AS CHAR)

    IF (@SLPB  < 100)

        RETURN @ID\_PB + '00' + CAST(@SLPB AS CHAR)

    IF (@SLPB < 1000)

        RETURN @ID\_PB + '0' + CAST(@SLPB AS CHAR)

    RETURN @ID\_PB + CAST(@SLPB AS CHAR)

END

GO

--PRINT DBO.MA\_PB()

--GO

DROP PROCEDURE IF EXISTS THEM\_MAPB

GO

CREATE PROCEDURE THEM\_MAPB(

    @TENPB VARCHAR(50),

    @DIACHIPB INT

)

AS

BEGIN

    INSERT INTO DM\_PHONGBAN

    VALUES

            (DBO.MA\_PB(),@TENPB,@DIACHIPB)

END

GO

--print truoc khi exec

SELECT \* FROM DM\_PHONGBAN

GO

**Kết quả TRƯỚC khi EXEC:**

Graphical user interface, application

Description automatically generated

Hình 3: Function MaPB trước khi execute

**Kết quả SAU khi EXEC:**

EXEC THEM\_MAPB 'KETOAN','1'

EXEC THEM\_MAPB 'NHANSU','2'

EXEC THEM\_MAPB 'KINHDOANH','3'

EXEC THEM\_MAPB 'MARKETING','4'

EXEC THEM\_MAPB 'EDIT','5'

--print sau khi exec

SELECT \* FROM DM\_PHONGBAN

Graphical user interface, application, table

Description automatically generated

Hình 4:Function MaPB trước và sau khi execute

**Funtion 2: Hàm tự động insert mã nhân viên:**

--FUNCTION 2

SET DATEFORMAT DMY

DROP FUNCTION IF EXISTS MA\_NV

GO

CREATE FUNCTION MA\_NV()

RETURNS CHAR(6)

AS

BEGIN

    DECLARE @ID\_NV CHAR(2)

    SET @ID\_NV = 'NV'

    DECLARE @SOLUONG INT

    SET @SOLUONG = 1+ (SELECT COUNT(MANV) FROM DM\_NHANVIEN)

    IF (@SOLUONG < 10)

        RETURN @ID\_NV + '000' + CAST(@SOLUONG AS CHAR)

    IF (@SOLUONG < 100)

        RETURN @ID\_NV + '00' + CAST(@SOLUONG AS CHAR)

    IF (@SOLUONG < 1000)

        RETURN @ID\_NV + '0' + CAST(@SOLUONG AS CHAR)

    RETURN @ID\_NV + CAST(@SOLUONG AS CHAR)

END

GO

--PRINT DBO.MA\_NV()

--GO

DROP PROCEDURE IF EXISTS THEM\_NV

GO

CREATE PROCEDURE THEM\_NV(

    @HO VARCHAR (50),

    @TENDEM VARCHAR (50),

    @TEN VARCHAR (50),

    @SODIENTHOAI VARCHAR(20),

    @DIACHI VARCHAR(50),

    @NGAYSINH DATE,

    @MAPB CHAR(6)

)

AS

BEGIN

    INSERT INTO DM\_NHANVIEN

    VALUES

            (DBO.MA\_NV(),@HO,@TENDEM,@TEN,@SODIENTHOAI, @DIACHI,@NGAYSINH,@MAPB)

END

GO

--print truoc khi exec

SELECT \* FROM DM\_NHANVIEN

GO

**Kết quả TRƯỚC khi EXEC:**

Graphical user interface, application, Word

Description automatically generated

Hình 5: Function MaNV trước khi execute

**Kết quả SAU khi EXEC:**

EXEC THEM\_NV 'PHAM','QUANG','HUY','0447364834', 'VUNG TAU','15/02/1995','PB0001'

EXEC THEM\_NV 'TRAN','THUY','LINH','0346364234', 'BINH DINH','20/07/2002','PB0002'

EXEC THEM\_NV 'DOAN','MINH','ANH','0973743644', 'NINH BINH','02/07/2000','PB0003'

EXEC THEM\_NV 'TRINH','HUNG','CUONG','0943472473', 'AN GIANG','20/12/2002','PB0004'

EXEC THEM\_NV 'VO','HOANG','OANH','0843463244', 'HA GIANG','17/10/2001','PB0005'

EXEC THEM\_NV 'HOANG','MINH','ANH','0843463244', 'DA NANG','17/12/1995','PB0002'

--print sau khi exec

SELECT \* FROM DM\_NHANVIEN

Table

Description automatically generated

Hình 6: Trước và sau khi execute MANV

**c.** **Tạo ít nhất một trigger để kiểm tra ràng buộc khoá ngoại, ràng buộc miền giá trị**

**TRIGGER 1: CHECK KHÓA NGOẠI**

--TRIGGER 1 CHECK KHOA NGOAI

DROP TRIGGER IF EXISTS CHK\_KHOANGOAI\_NV

GO

CREATE TRIGGER CHK\_KHOANGOAI\_NV

ON DM\_NHANVIEN

INSTEAD OF INSERT

AS

BEGIN

    DECLARE @MA\_PBB CHAR(6)

    SET @MA\_PBB = (SELECT MAPB FROM INSERTED);

    IF @MA\_PBB IN (SELECT MAPB FROM DM\_PHONGBAN)

        INSERT INTO DM\_NHANVIEN SELECT \* FROM INSERTED;

    ELSE

        PRINT 'ERROR'

END

GO

**INSERT ‘PB0006’ là mã phòng ban chưa được insert ở dữ liệu câu b**

EXEC THEM\_NV 'PHAM','NHAT','ANH',0843463244, 'DA LAT','17/10/1995','PB0006'

**Kết quả sau khi insert dữ liệu lỗi:**

Graphical user interface, text

Description automatically generated with medium confidence

Báo lỗi do trong code sql trigger print ‘ERROR’ vì chưa có dữ liệu import của mã Phòng ban nên mã nhân viên thuộc phòng ban này cũng không tồn tại.

**TRIGGER 2: CHECK MIỀN GIÁ TRỊ**

--TRIGGER 2 CHECK MIEN GIA TRI

DROP TRIGGER IF EXISTS CHK\_LUONG\_PARTTIME

GO

CREATE TRIGGER CHK\_LUONG\_PARTTIME

ON NV\_PARTTIME

INSTEAD OF INSERT

AS

BEGIN

    DECLARE @LUONG\_GIO INT

    SET @LUONG\_GIO = (SELECT LUONGGIO FROM INSERTED);

    IF @LUONG\_GIO > 0

        INSERT INTO NV\_PARTTIME SELECT \* FROM INSERTED;

    ELSE

        PRINT 'ERROR'

END

GO

INSERT giá trị của lương lần lượt là 20 và -5

INSERT INTO NV\_PARTTIME

VALUES

    ('NV0001',20)

GO

SELECT \* FROM NV\_PARTTIME

INSERT INTO NV\_PARTTIME

VALUES

    ('NV0002',-5)

GO

**Kết quả sau khi insert 20 dữ liệu được insert đúng.**

Graphical user interface, text, application, email

Description automatically generated

Hình 7: Trigger check miền giá trị khi insert đúng điều kiện

**Kết quả sau khi insert -5 dữ liệu lỗi (lương có giá trị âm):**

Text

Description automatically generated

Hình 8: Trigger check miền giá trị khi insert sai điều kiện

Báo lỗi do trong code sql trigger print ‘ERROR’ Báo lỗi do trong code sql trigger print ‘ERROR’ vì mức lương không được âm.

1. PHẦN II
   1. Câu 1
      1. Yêu cầu đề bài

Input: File .txt chứa các thực thể và mối quan hệ giữa các thực thể của mô hình ERD. Output: File .txt chứa các bảng dữ liệu và mối quan hệ giữa các bảng. Định dạng của file Input.txt do sinh viên tự thiết lập. (2 điểm)

* + 1. Thực hiện yêu cầu

#read and write files

rfile = open('input.txt','r')

wfile = open('output.txt','w')

#read files input

for line in rfile.readlines():

    #split string and get relationship from input file

    line = line.strip().split(':');

    relationship\_input = line[1].strip();

    #split string and get table in database schema

    data = line[0].strip().split('--');

    #merge relationship and table into a list

    data.append(str(relationship\_input));

    relation = data[-1];

    tables = data[:-1];

    if relation == '1-1' or relation == '1-N':

        txt = "{} -- {}\n".format(\*tables);

        wfile.write(txt);

    if relation == 'N-N':

        main = "\_".join(tables);

        for i in data[:-1]:

            txt = "{} -- {}\n".format(main, i);

            wfile.write(txt);

Trong file input.txt, file này được chia thành từng dòng với mỗi dòng là hai thực thể trong mô hình ERD đã vẽ ở Phần 1, hai thực thể này được ngăn cách bởi dấu “--“. Sau 2 thực thể này thì ngăn cách tiếp theo bởi dấu “:” là mối quan hệ giữa các mối quan hệ với nhau gồm: “1-N, N-N, 1-1”.

Graphical user interface, text

Description automatically generated

Hình 9: Nội dung trong file input.txt của câu 1

Trong file output.txt, file output này chứa dữ liệu đã được xử lý và được chia thành từng dòng, mỗi dòng trong file này bao gồm các bảng dữ liệu mới và có những bảng mới xuất hiện do các mối quan hệ tạo ra.

Graphical user interface, text, application

Description automatically generated

Hình 10: Nội dung file output.txt của câu 1

* 1. Câu 2
     1. Yêu cầu đề bài

Input: File .txt chứa lược đồ CSDL bao gồm các bảng, các thuộc tính và các phụ thuộc hàm. Output: File .txt chứa bao đóng của tập thuộc tính X (Tập thuộc tính X do người dùng truyền vào); các khoá của lược đồ quan hệ. (2 điểm)

* + 1. Thực hiện yêu cầu

from itertools import combinations

# input and output

rfile = open("input.txt", "r")

wfile = open("output.txt", "w")

tables = {}

# read input.txt and add data to tables dictionary

for line in rfile.readlines():

    fd = line[line.index(')') + 2:].strip().split(',');

    fds = []

    for j in fd:

        src, des = j.split('->')

        if '+' not in src:

            if '+' not in des:

                fds.extend([[src, des]])

            else:

                fds.extend([[src, des.split('+')]])

        else:

            if '+' not in des:

                fds.extend([[src.split('+'), des]])

            else:

                fds.extend([[src.split('+'), des.split('+')]])

    tables[line[0:line.index('(')]] = {

        'attrs': line.strip('\n')[line.index('(')+1:line.index(')')].split(','),

        'fds': fds

    }

# function to find closure from list of attributes

def closureOfSetAttributes(name, set\_att):

    while(True):

        is\_change = False;

        for Ul, Ur in tables[name]['fds']:

            if type(Ul) is not list and Ul in set\_att:

                if type(Ur) is not list and Ur not in set\_att:

                    set\_att.append(Ur);

                    is\_change = True;

                elif type(Ur) is list and any(item  not in set\_att for item in Ur):

                    set\_att.extend(Ur);

                    is\_change = True;

            if type(Ul) is list and all(item in set\_att for item in Ul):

                if type(Ur) is not list and Ur not in set\_att:

                    set\_att.append(Ur);

                    is\_change = True;

                elif type(Ur) is list and any(item not in set\_att for item in Ur):

                    set\_att.extend(Ur);

                    is\_change = True;

        if is\_change == False:

            break;

    return list(set(set\_att));

X = ['B', 'G'];

wfile.write("X: {}\n".format(X));

for table\_name in tables.keys():

    wfile.write("CLOSURE OF {}: {}\n".format(table\_name, closureOfSetAttributes(table\_name, X)));

# function to find keys of tables

def key():

    keys = {};

    for table\_name in tables.keys():

        res = [];

        R = list(set([item for Ul, Ur in tables[table\_name]['fds'] for item in Ur]));

        L = list(set([item for Ul, Ur in tables[table\_name]['fds'] for item in Ul]));

        # TG = R - L

        TG = list(set(L).intersection(set(R)));

        U = tables[table\_name]['attrs'];

        # TD = U - R

        TN = [item for item in U if item not in R]

        Xi = sum([list(map(list, combinations(TG, i))) for i in range(len(TG) + 1)], [])

        for item in Xi:

            temp = TN + item;

            temp1 = temp.copy();

            closure\_temp = closureOfSetAttributes(table\_name, temp);

            if set(closure\_temp) == set(tables.get(table\_name)['attrs']):

                res.append(list(set(temp1)));

        keys[table\_name] = [x for x in res if len(x) == len(min(res, key=len))];

    return keys;

wfile.write("KEYS: \n");

for table\_name in key().keys():

    wfile.write("{} : {}\n".format(table\_name, key().get(table\_name)));

Trong file input.txt, file này chứa một lược đồ cơ sở dữ liệu và các phụ thuộc hàm, được chia thành từng dòng và mỗi dòng một lược đồ quan hệ đi kèm với các phụ thuộc hàm tương ứng của lược đồ quan hệ đó.

Text

Description automatically generated

Hình 11: Nội dung file input.txt của câu 2

Trong file output.txt, file output này sẽ có tập thuộc tính X do người dùng truyền vào ở đây cụ thể là X: [‘B’,’G’], bao đóng (CLOSURE) của từng lược đồ quan hệ và cuối cùng là tập hợp danh sách các khoá (KEYS) của lược đồ được viết trên từng dòng.

A picture containing text

Description automatically generated

Hình 12: Nội dung trong file output.txt của câu 2

**TÀI LIỆU THAM KHẢO**

**Tiếng Anh:**

1. Learning SQL, Alan Beaulieu, 2nd edition, Penguin Books Ltd
2. SQL Cookbook (O’Reilly), Anthony Molinaro, 1st edition, O′Reilly