TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ MÔN CƠ SỞ DỮ LIỆU**

**BÁO CÁO CUỐI KỲ**

*Người hướng dẫn*: **THẦY TRẦN THỊ BÍCH VÂN**

*Người thực hiện*: **NGUYỄN THỊ THU HỒNG – 52100962**

Lớp **: 21050401**

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 CUỐI KỲ MÔN CƠ SỞ DỮ LIỆU**

**BÁO CÁO CUỐI KỲ**

*Người hướng dẫn*: **THẦY TRẦN THỊ BÍCH VÂN**

*Người thực hiện*: **NGUYỄN THỊ THU HỒNG – 52100962**

Lớp **: 21050401**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn đến quý thầy/cô trường **Đại học Tôn Đức Thắng** và các thầy/cô là giảng viên thỉnh giảng của các trường Đại học khác đã giảng dạy bằng tâm huyết để truyền đạt vốn kiến thức quý báu cho chúng em trong suốt học kỳ vừa rồi. Đặc biệt là cô Trần Thị Bích Vân, giảng viên môn Hệ Cơ Sở Dữ Liệu cả lý thuyết nhóm 1 và thực hành của nhóm 1 tổ 1Em rất biết ơn vì cô và thầy đã dành rất nhiều tâm huyết để dẫn dắt chúng em trong quá trình học tập và tìm hiểu về bộ môn Giải tích ứng dụng cho công nghệ thông tin. Em đã nhận được rất nhiều sự quan tâm, giúp đỡ, hướng dẫn tận tình và tâm huyết của thầy, cô. Thầy cô đã giúp em tích lũy thêm nhiều kiến thức bổ ích để có cái nhìn tổng quan và rõ ràng hơn về môn học. Từ những kiến thức mà thầy cô truyền tải, em đã dần trả lời được rất nhiều dạng bài tập và câu hỏi mới. Thông qua bài báo cáo cuối kỳ này, em xin trình bày lại những kiến thức mà các thầy cô truyền tải và cũng như những kiến thức từ sự tìm hiểu của bản thân em qua những bài tập giải tích để gửi đến thầy cô.

Có lẽ kiến thức là vô hạn mà sự tiếp nhận kiến thức của bản thân mỗi người luôn tồn tại những hạn chế nhất định. Do đó, trong quá trình hoàn thành bài báo cáo, chắc chắn sẽ không thể tránh khỏi những thiếu sót. Bản thân em rất mong nhận những lời góp ý đến từ thầy cô để bài báo cáo của em có thể trở nên hoàn thiên hơn.

Kính chúc thầy cô sức khỏe, hạnh phúc và thành công trên con đường truyền tải tri thức cao quý. Và một lần nữa em xin kính chúc thầy cô có một năm mới vui vẻ, an lành và ấm áp bên người thân, gia đình.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên, bài báo cáo này của các quý thầy/cô đã tạo thêm điều kiện để em có thể bổ sung, nâng cao thêm kiến thức, kinh nghiệm của mình.

Em xin chân thành cảm ơn. !

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm báo cáo của riêng tôi và được sự hướng dẫn của thầy Doãn 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 bài tiểu luận 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 23 tháng 11 năm 2022*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Hồng*

*Nguyễn Thị Thu Hồng*

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)

MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc106094162)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 3](#_Toc106094163)

[MỤC LỤC 4](#_Toc106094164)

[MỤC LỤC HÌNH ẢNH 5](#_Toc106094165)

[PHẦN 1 6](#_Toc106094166)

[1.1 Đề bài: 6](#_Toc106094167)

[1.2 Cơ sở dữ liệu quản lý các dịch vụ tại ngân hàng: 7](#_Toc106094168)

[1.2.1 Mô tả: 7](#_Toc106094169)

[1.2.2 Yêu cầu: 8](#_Toc106094172)

[1.2.2.1 Các mối quan hệ (2 điểm) 9](#_Toc106094173)

[1.2.2.2 Vẽ mô hình ERD cho đặc tả (1 điểm) 10](#_Toc106094174)  
1.2.2.3 Chuyển mô hình ERD sang mô hình quan hệ (1 điểm)…11

1.2.2.3 Sử dụng câu lệnh SQL để thực hiện các thao tác sau:…...12

[PHẦN 2 25](#_Toc106094184)

[2.1 Đề bài 25](#_Toc106094185)

[2.2 Bài làm 26](#_Toc106094186)

[2.2.1 Câu 1: Sử dụng Python 26](#_Toc106094169)

[2.2.2 Câu 2: Sử dụng Java: 27](#_Toc106094172)MỤC LỤC HÌNH ẢNH

[Hình 1. Mô hình ERD 10](#_Toc106095232)

[Hình 2. Mô hình quan hệ 11](#_Toc106095233)

[Hình 3. Bảng ATM 17](#_Toc106095234)

[Hình 4. Bảng ChiNhanh 17](#_Toc106095235)

[Hình 5. Bảng KhachHang 17](#_Toc106095236)

[Hình 6. Bảng NganHang 18](#_Toc106095237)

[Hình 7. Bảng TaiKhoan 18](#_Toc106095238)

[Hình 8. Bảng tkkhongtrailai 18](#_Toc106095239)

[Hình 9. Bảng tkchovay 18](#_Toc106095240)

[Hình 10. Bảng tktietkiem 19](#_Toc106095241)

[Hình 11. Bảng giadinh 19](#_Toc106095242)

PHẦN 1

* 1. 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.

Yêu cầu:

1. Các mối quan hệ: 2 điểm

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

2. Vẽ mô hình ERD cho đặc tả (1 điểm)

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

4. 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. Cơ sở dữ liệu về quản lý các dịch vụ tại ngân hàng
     1. Mô tả:

**Đề tài:** Quản lý các dịch vụ tại ngân hàng.

Thiết kế một cơ sở dữ liệu phục quản lý các dịch vụ tại ngân hàng. Mỗi ngân hàng có nhiều chi nhánh nhưng một chi nhánh chỉ thuộc về một ngân hàng. Mỗi chi nhánh ngân hàng sẽ có một mã số để phân biệt và tên của chi nhánh. Ngân hàng có những thông tin sau: mã số ngân hàng và tên ngân hàng. Mỗi trụ ATM sẽ có một mã để quản lý. Một chi nhánh có thể quản lý nhiều máy ATM. Ngoài ra, mỗi trụ ATM còn có thêm các thông tin như: tình trạng hoạt động, nơi đặt. Một tài khoản của một ngân hàng sẽ có một mã số tài khoản (tất cả các số tài khoản là duy nhất), tên tài khoản. Một ngân hàng có nhiều tài khoản. Có ba loại tài khoản ngân hàng: Tài khoản tiết kiệm: phải lưu trữ số dư của tài khoản, lãi suất, ngày mở tài khoản Tài khoản không trả lãi: theo dõi số dư và ngày mở tài khoản Tài khoản cho vay: lưu trữ ngày, khoản vay được vay, tỷ lệ lãi suất Một tài khoản thuộc về duy nhất một khách hàng và một khách hàng có thể sử dụng nhiều tài khoản. Thông tin về khách hàng bao gồm: căn cước công dân, tên khách hàng, địa chỉ, số điện thoại và thông tin về gia đình. Thông tin về gia đình bao gồm: số lượng thành viên, tình trạng hôn nhân. Khi khách đến trụ máy ATM để thực hiện giao dịch rút tiền, sau mỗi giao dịch, hệ thống sẽ ghi nhận lại mã số giao dịch cho lần giao dịch đó. Mã giao dịch sẽ cho biết mã tài khoản mà người giao dịch đã dùng, ngày giờ giao dịch, số tiền giao dịch

* + 1. Yêu cầu:

1. Các mối quan hệ: 2 điểm

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

2. Vẽ mô hình ERD cho đặc tả (1 điểm)

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

4. 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)

**1.2.2.1**  **Các mối quan hệ (2 điểm)**

a. Quan hệ 1-n: 0.5 điểm

+ Ngân hàng – Chi nhánh:

Quan hệ 1-n: Mỗi ngân hàng có nhiều chi nhánh nhưng một chi nhánh chỉ thuộc về một ngân hàng.

+ Chi nhánh – ATM:

Quan hệ 1-n: Một chi nhánh có thể quản lý nhiều máy ATM.

+ Ngân hàng – Tài khoản:

Quan hệ 1-n: Một tài khoản của một ngân hàng sẽ có một mã số tài khoản (tất cả các số tài khoản là duy nhất), tên tài khoản. Một ngân hàng có nhiều tài khoản.

+ Khách hàng – Tài khoản:

Quan hệ 1-n: Một tài khoản thuộc về duy nhất một khách hàng và một khách hàng có thể sử dụng nhiều tài khoản.

b. Quan hệ n-n: 0.5 điểm

+ Khách hàng – ATM:

Quan hệ n-n: Một khách hàng có thể rút tiền tại nhiều ATM và một cây ATM có thể được sử dụng bởi nhiều khách hàng và nhiều lần.

c. Quan hệ cha con: 0.5 điểm

+ Tài khoản – Tài khoản không trả lãi:

+ Tài khoản – Tài khoản tiết kiệm:

+ Tài khoản – Tài khoản cho vay:

* Các tài khoản không trả lãi, tài khoản cho vay và tài khoản tiết kiệm sẽ kế thừa các thuộc tính của bảng tài khoản và có thêm một vài thuộc tính của riêng mình.

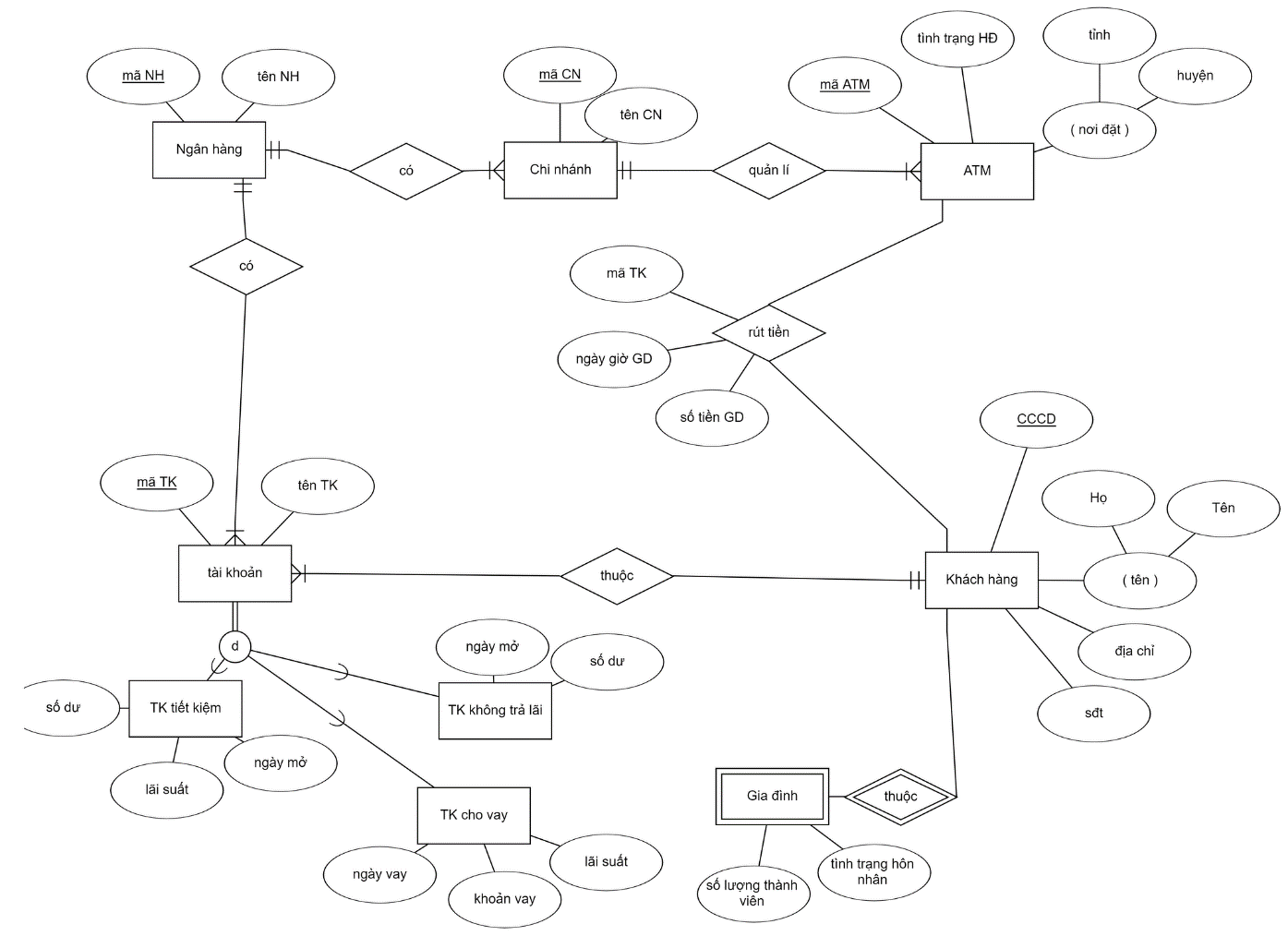
d. Quan hệ 1-1: 0.25 điểm

+

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

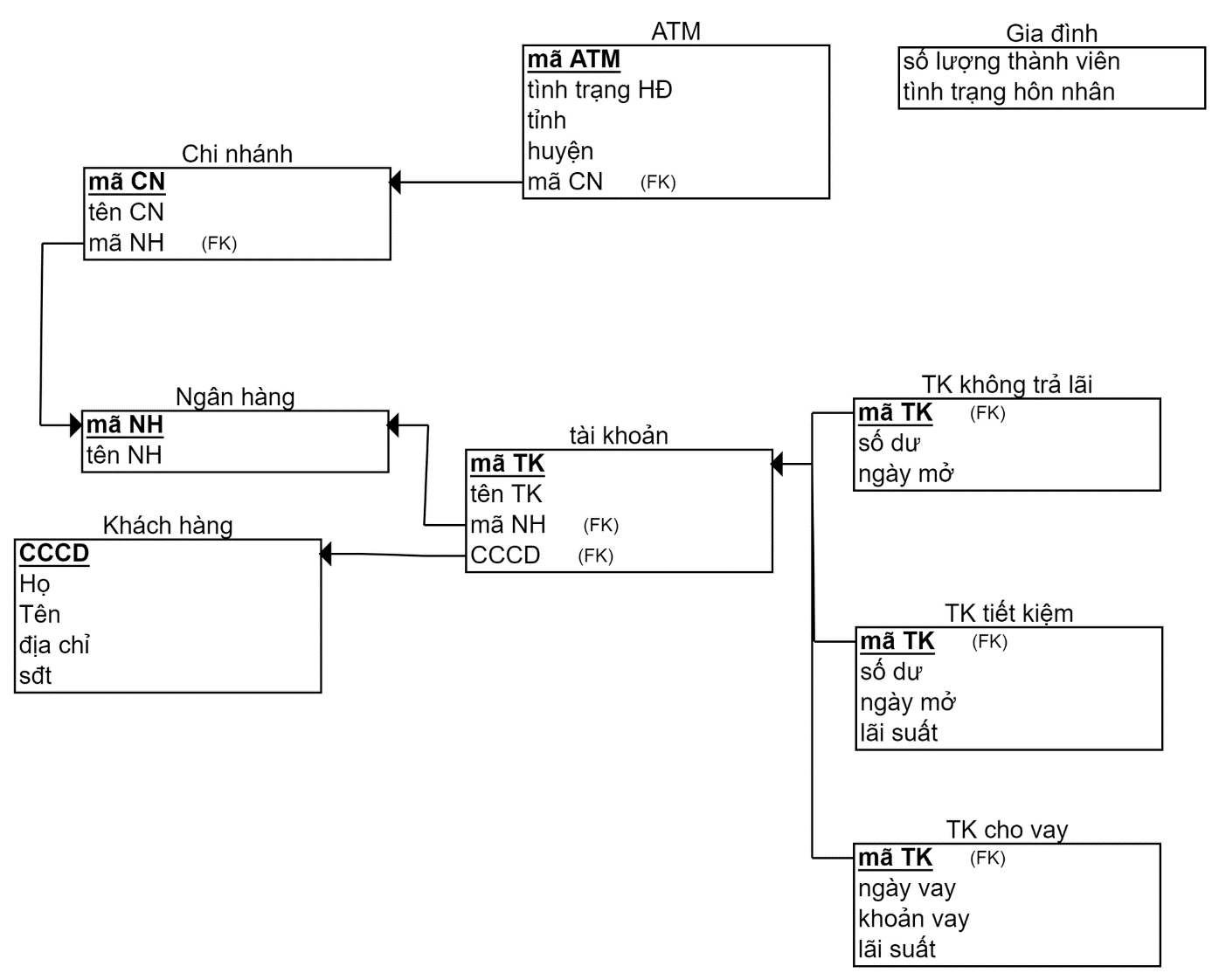
+ Khách hàng – Gia đình:

**1.2.2.2 Vẽ mô hình ERD cho đặc tả (1 điểm)**

****

Hình 1: Mô hình ERD

**1.2.2.3 Chuyển mô hình ERD sang mô hình quan hệ (1 điểm)**

****

Hình 2: Mô hình quan hệ

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

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

use master

if exists(select\* from sysdatabases where name='QuanLyDichVuNganHang')

drop database QuanLyDichVuNganHang

go

-- Tạo cơ sở dữ liệu

create database QuanLyDichVuNganHang

go

use QuanLyDichVuNganHang

go

-- Tạo các bảng

create table NganHang (

manganhang char(10) not null,

tennganhang nvarchar(50) not null,

primary key(manganhang)

)

create table ChiNhanh (

machinhanh char(10) not null,

tenchinhanh nvarchar(50) not null,

manganhang char(10) not null,

primary key(machinhanh),

constraint ChiNhanh\_fk\_manganhang foreign key(manganhang) references NganHang(manganhang)

)

create table ATM (

maATM char(10) not null,

tinhtranghd nvarchar(20) not null,

tinh nvarchar(20) not null,

huyen nvarchar(20) not null,

machinhanh char(10) not null,

primary key(maATM),

constraint ATM\_fk\_machinhanh foreign key(machinhanh) references ChiNhanh(machinhanh)

)

create table KhachHang (

cccd char(12) not null,

ho nvarchar(20) not null,

ten nvarchar(20) not null,

diachi nvarchar(50) not null,

sdt char(12) not null,

primary key(cccd)

)

create table TaiKhoan (

mataikhoan char(10) not null,

tentaikhoan nvarchar(50) not null,

manganhang char(10) not null,

cccd char(12) not null,

primary key(mataikhoan),

constraint TaiKhoan\_fk\_manganhang foreign key(manganhang) references NganHang(manganhang),

constraint TaiKhoan\_fk\_cccd foreign key(cccd) references KhachHang(cccd)

)

create table tkkhongtralai (

mataikhoan char(10) not null,

sodu int not null check(sodu >= 0),

ngaymo date,

primary key(mataikhoan),

constraint tkkhongtralai\_fk\_mataikhoan foreign key(mataikhoan) references TaiKhoan(mataikhoan)

)

create table tktietkiem (

mataikhoan char(10) not null,

sodu int not null check(sodu >= 0),

ngaymo date,

laisuat float not null check (laisuat > 0),

primary key(mataikhoan),

constraint tktietkiem\_fk\_mataikhoan foreign key(mataikhoan) references TaiKhoan(mataikhoan)

)

create table tkchovay (

mataikhoan char(10) not null,

ngayvay date,

khoanvay int not null check (khoanvay >= 0),

laisuat int,

primary key(mataikhoan),

constraint tkchovay\_fk\_mataikhoan foreign key(mataikhoan) references TaiKhoan(mataikhoan)

)

create table giadinh (

cccd char(12) not null,

soluongtv int not null check (soluongtv >= 1),

tinhtranghonnhan nvarchar(20) not null check (tinhtranghonnhan in (N'Đã kết hôn', N'Độc thân')),

primary key(cccd),

constraint giadinh\_fk\_ccd foreign key(cccd) references KhachHang(cccd)

)

1. 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.

* Code:

-- Thêm dữ liệu

--- Bảng Ngân Hàng

insert into NganHang values('NH001', N'VietComBank')

insert into NganHang values('NH002', N'Aribank')

insert into NganHang values('NH003', N'VietTinBank')

insert into NganHang values('NH004', N'TP')

insert into NganHang values('NH005', N'ACB')

--- Bảng Chi Nhánh

insert into ChiNhanh values('CN001', N'Nam Sài Gòn', 'NH001')

insert into ChiNhanh values('CN002', N'Bến Tre', 'NH002')

insert into ChiNhanh values('CN003', N'Bình Dương', 'NH003')

insert into ChiNhanh values('CN004', N'Thủ Đức', 'NH004')

insert into ChiNhanh values('CN005', N'Long An', 'NH005')

insert into ChiNhanh values('CN006', N'Thủ Đức', 'NH001')

insert into ChiNhanh values('CN007', N'Thủ Dầu Một', 'NH002')

insert into ChiNhanh values('CN008', N'Bình Phước', 'NH003')

insert into ChiNhanh values('CN009', N'Cần Thơ', 'NH004')

insert into ChiNhanh values('CN0010', N'Cà Mau', 'NH005')

--- Bảng ATM

insert into ATM values('ATM001', N'Bình thường', N'Bình Dương', N'Bến Cát', 'CN003')

insert into ATM values('ATM002', N'Bình thường', N'Bình Dương', N'Dĩ An', 'CN003')

insert into ATM values('ATM003', N'Bình thường', N'Bình Dương', N'Thuận An', 'CN003')

insert into ATM values('ATM004', N'Bình thường', N'Bình Phước', N'Hớn Quản', 'CN008')

insert into ATM values('ATM005', N'Bình thường', N'Bình Phước', N'Lộc Ninh', 'CN008')

insert into ATM values('ATM006', N'Bình thường', N'Bến Tre', N'Châu Thành', 'CN002')

insert into ATM values('ATM007', N'Bình thường', N'Long An', N'Bến Lức', 'CN005')

insert into ATM values('ATM008', N'Bình thường', N'Cần Thơ', N'Cờ Đỏ', 'CN003')

insert into ATM values('ATM009', N'Bình thường', N'Cà Mau', N'Phú Tân', 'CN003')

--- Bảng Khách Hàng

insert into KhachHang values('070303000412', N'Lê Thị Phi', N'Du', N'Bến Tre 001', '0918477203')

insert into KhachHang values('070303000782', N'Lê Thị Bích', N'Ngọc', N'Long An 001', '0987213775')

insert into KhachHang values('070303000991', N'Nguyễn Cẩm', N'Tú', N'Bình Phước 001', '0872146973')

insert into KhachHang values('070303000223', N'Châu Thị Tuyết', N'Mai', N'Bến Tre 001', '0333098142')

insert into KhachHang values('070303000410', N'Huỳnh Đức', N'Hòa', N'Bình Phước 002', '0743114873')

insert into KhachHang values('070303000001', N'Bùi Đại Phú', N'Quí', N'Cà Mau 001', '0983112203')

--- Bảng Tài Khoản

insert into TaiKhoan values('TK001', N'lê Thị Phi Du', 'NH001', '070303000412')

insert into TaiKhoan values('TK002', N'lê Thị Bích Ngọc', 'NH003', '070303000782')

insert into TaiKhoan values('TK003', N'Nguyễn Cẩm Tú', 'NH001', '070303000991')

insert into TaiKhoan values('TK004', N'Châu Thị Tuyết Mai', 'NH002', '070303000223')

insert into TaiKhoan values('TK005', N'Huỳnh Đức Hòa', 'NH005', '070303000410')

insert into TaiKhoan values('TK006', N'Bùi Đại Phú Quí', 'NH004', '070303000001')

--- Bảng Tài Khoản Không Trả Lãi

insert into tkkhongtralai values('TK001', 50000000, '1/1/2018')

insert into tkkhongtralai values('TK002', 1000000, '5/3/2018')

--- Bảng Tài Khoản Tiết Kiệm

insert into tktietkiem values('TK003', 25000000, '9/5/2014', 0.01)

insert into tktietkiem values('TK004', 10000000, '12/5/2016', 0.01)

--- Bảng Tải Khoản Cho Vay

insert into tkchovay values('TK005', '11/4/2014', 5000000, 0.025)

insert into tkchovay values('TK006', '9/7/2012', 70000000, 0.025)

--- Bảng Gia Đình

insert into giadinh values('070303000412', 3, N'Độc thân')

insert into giadinh values('070303000782', 4, N'Độc thân')

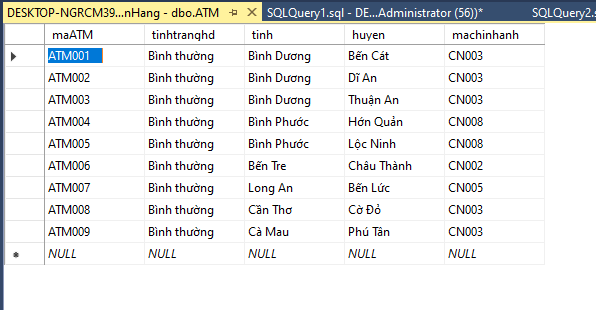
insert into giadinh values('070303000991', 5, N'Độc thân')

insert into giadinh values('070303000223', 3, N'Đã kết hôn')

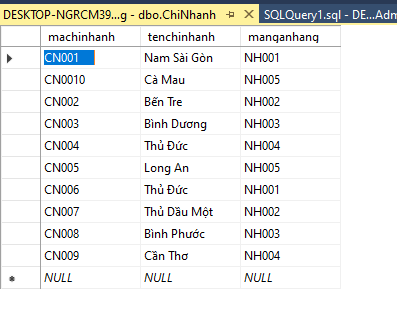
insert into giadinh values('070303000410', 5, N'Độc thân')

insert into giadinh values('070303000001', 3, N'Đã kết hôn')

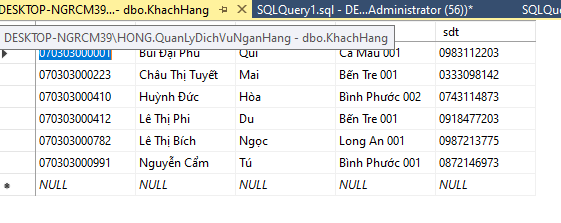
* Kết quả chạy thực thi:



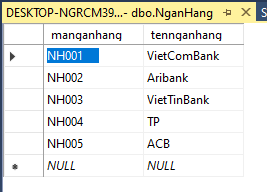
Hình 3: Bảng ATM



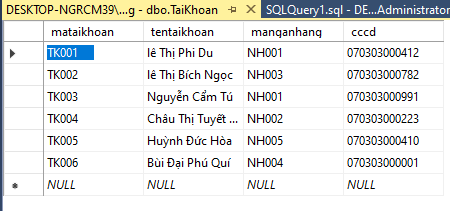
Hình 4: Bảng ChiNhanh



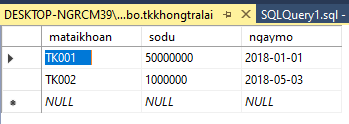
Hình 5: Bảng KhachHang



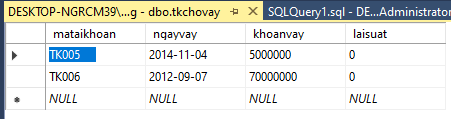
Hình 6: Bảng NganHang



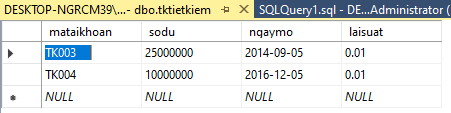
Hình 7: Bảng TaiKhoan



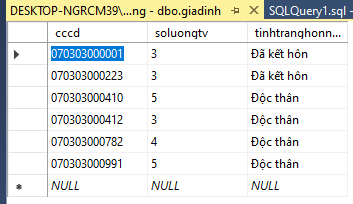
Hình 8: Bảng tkkhongtralai



Hình 9: Bảng tkchovay



Hình 10: Bảng tktietkiem



Hình 11: giadinh

* Viết mã tự động cho các bảng:

-- Tạo mã tự động

--- Mã tự động cho bảng TaiKhoan

CREATE FUNCTION AUTO\_MaTK\_TaiKhoan()

RETURNS CHAR(10)

AS

BEGIN

DECLARE @ID CHAR(10)

DECLARE @a CHAR(10)

IF (SELECT COUNT(mataikhoan) FROM TaiKhoan) = 0

SET @ID = '0'

ELSE

SELECT top 1 @ID = mataikhoan FROM TaiKhoan order by mataikhoan desc

set @a = SUBSTRING(@ID, 4, len(@ID))

if

@a >= 0 and @a < 9 SET @ID = 'TK00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

else

set @ID = 'TK00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

RETURN @ID

END

go

create procedure add\_TK @mataikhoan char(10),

@tentaikhoan nvarchar(50),

@manganhang char(10),

@cccd char(12)

as

begin

insert into TaiKhoan

values (dbo.AUTO\_MaTK\_TaiKhoan(), @mataikhoan, @tentaikhoan, @manganhang, @cccd)

end

--- Mã tự động cho bảng NganHang

CREATE FUNCTION AUTO\_MaNH\_NganHang()

RETURNS CHAR(10)

AS

BEGIN

DECLARE @ID CHAR(10)

DECLARE @a CHAR(10)

IF (SELECT COUNT(manganhang) FROM NganHang) = 0

SET @ID = '0'

ELSE

SELECT top 1 @ID = manganhang FROM NganHang order by manganhang desc

set @a = SUBSTRING(@ID, 4, len(@ID))

if

@a >= 0 and @a < 9 SET @ID = 'NH00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

else

set @ID = 'NH00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

RETURN @ID

END

go

create procedure add\_NH @manganhang char(10),

@tennganhang nvarchar(50)

as

begin

insert into TaiKhoan

values (dbo.AUTO\_MaTK\_TaiKhoan(), @manganhang, @tennganhang)

end

--- Mã tự động cho bảng ChiNhanh

CREATE FUNCTION AUTO\_MaCN\_ChiNhanh()

RETURNS CHAR(10)

AS

BEGIN

DECLARE @ID CHAR(10)

DECLARE @a CHAR(10)

IF (SELECT COUNT(machinhanh) FROM ChiNhanh) = 0

SET @ID = '0'

ELSE

SELECT top 1 @ID = machinhanh FROM ChiNhanh order by machinhanh desc

set @a = SUBSTRING(@ID, 4, len(@ID))

if

@a >= 0 and @a < 9 SET @ID = 'CN00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

else

set @ID = 'CN00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

RETURN @ID

END

go

create procedure add\_CN @machinhanh char(10),

@tenchinhanh nvarchar(50),

@manganhang char(10)

as

begin

insert into TaiKhoan

values (dbo.AUTO\_MaCN\_ChiNhanh(), @machinhanh, @tenchinhanh, @manganhang)

end

--- Mã tự động cho bảng ATM

CREATE FUNCTION AUTO\_MaATM\_ATM()

RETURNS CHAR(10)

AS

BEGIN

DECLARE @ID CHAR(10)

DECLARE @a CHAR(10)

IF (SELECT COUNT(maATM) FROM ATM) = 0

SET @ID = '0'

ELSE

SELECT top 1 @ID = maATM FROM ATM order by maATM desc

set @a = SUBSTRING(@ID, 5, len(@ID))

if

@a >= 0 and @a < 9 SET @ID = 'ATM00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

else

set @ID = 'ATM00' + CONVERT(CHAR, CONVERT(INT, @a) + 1)

RETURN @ID

END

go

create procedure add\_ATM @maATM char(10),

@tinhtranhd nvarchar(50),

@tinh nvarchar(20),

@huyen nvarchar(20),

@machinhanh char(10)

as

begin

insert into TaiKhoan

values (dbo.AUTO\_MaATM\_ATM(), @maATM, @tinhtranhd, @tinh, @huyen, @machinhanh)

end

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

-- Tạo trigger

--- Trigger cho tkkhongtralai

CREATE TRIGGER khoangoai\_tkktl ON tkkhongtralai instead of INSERT

AS

IF EXISTS(SELECT inserted.mataikhoan FROM TaiKhoan INNER JOIN inserted ON TaiKhoan.mataikhoan = inserted.mataikhoan WHERE inserted.sodu < 0)

begin

print (N'Số dư ra không hợp lệ')

rollback Tran

end

else if exists(select inserted.mataikhoan from inserted where inserted.mataikhoan not in (select mataikhoan from TaiKhoan))

begin

print (N'Mã tài khoản không hợp lệ')

rollback Tran

end

else if exists(select inserted.ngaymo from inserted where inserted.ngaymo > getdate())

begin

print (N'Ngày mở không hợp lệ')

rollback Tran

end

--- Trigger cho tktietkiem

CREATE TRIGGER khoangoai\_tktk ON tktietkiem instead of INSERT

AS

IF EXISTS(SELECT inserted.mataikhoan FROM TaiKhoan INNER JOIN inserted ON TaiKhoan.mataikhoan = inserted.mataikhoan WHERE inserted.sodu < 0)

begin

print (N'Số dư ra không hợp lệ')

rollback Tran

end

else if exists(select inserted.mataikhoan from inserted where inserted.mataikhoan not in (select mataikhoan from TaiKhoan))

begin

print (N'Mã tài khoản không hợp lệ')

rollback Tran

end

else if exists(select inserted.ngaymo from inserted where inserted.ngaymo > getdate())

begin

print (N'Ngày mở không hợp lệ')

rollback Tran

end

else if exists(select inserted.laisuat from inserted where inserted.laisuat < 0)

begin

print (N'Lãi suất ra không hợp lệ')

rollback Tran

end

--- Trigger cho tkchovay

CREATE TRIGGER khoangoai\_tkchovay ON tkchovay instead of INSERT

AS

IF EXISTS(SELECT inserted.mataikhoan FROM TaiKhoan INNER JOIN inserted ON TaiKhoan.mataikhoan = inserted.mataikhoan WHERE inserted.khoanvay < 0)

begin

print (N'Khoảng vay ra không hợp lệ')

rollback Tran

end

else if exists(select inserted.mataikhoan from inserted where inserted.mataikhoan not in (select mataikhoan from TaiKhoan))

begin

print (N'Mã tài khoản không hợp lệ')

rollback Tran

end

else if exists(select inserted.ngayvay from inserted where inserted.ngayvay > getdate())

begin

print (N'Ngày vay không hợp lệ')

rollback Tran

end

else if exists(select inserted.laisuat from inserted where inserted.laisuat < 0)

begin

print (N'Lãi suất ra không hợp lệ')

rollback Tran

end

PHẦN 2

2.1 Đề bài

Sinh viên sử dụng ngôn ngữ Java hoặc Python để cài đặt chức năng sau:

1. 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)

2. 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)

**2.2 Bài làm**

**2.2.1 Câu 1: Sử dụng Python**

input = open("input1.txt", "r")

output = open("output1.txt", "a")

relationships, spl, nameTable, name, nameAttribute = [], [], [], [], []

x = input.read().split("\n" )

relationships.append(x[len(x) - 2].split(", "))

relationships.append(x[len(x) - 1].split(", "))

for i in range(0, len(x) - 2):

spl.append(x[i].split(" ("))

for i in range(len(spl)):

nameTable.append(spl[i][0])

for i in range(len(spl)):

name.append(spl[i][1].split(")"))

for i in range(len(name)):

nameAttribute.append(name[i][0])

for i in relationships:

if (i[1] == "1-n"):

for j in range(len(nameTable)):

if (nameTable[j] == i[0]):

output.write(nameTable[j] + "(" + nameAttribute[j] + ")\n")

nameAttribute\_1 = nameAttribute[j].split(", ")

n = nameAttribute\_1[0]

table\_1 = i[0]

if (nameTable[j] == i[2]):

output.write(nameTable[j] + '(' + nameAttribute[j] + ', ' + n.replace('pk', 'fk') +')\n')

table\_2 = i[2]

**2.2.2 Câu 2: Sử dụng Java**

**import** **java.io.FileInputStream**;

**import** **java.io.FileNotFoundException**;

**import** **java.io.FileWriter**;

**import** **java.util.ArrayList**;

**import** **java.util.Scanner**;

**public** **class** **Cau2** {

**public** **static** **void** **main**(String[] args) **throws** FileNotFoundException {

ArrayList<String> data = **new** ArrayList<>();

Scanner sc= **new** Scanner(System.in);

data = readFile("input/file2.txt");

System.out.println(data);

ArrayList<String> phuthuoc= **new** ArrayList<>();

phuthuoc=PhuThuocHam(data);

String attr=attribute(data.get(**0**).trim());

ArrayList<String> Khoa=timKhoa(attr, phuthuoc);

show(Khoa);

}

**private** **static** **void** **show**(ArrayList<String> khoa) {

String s="";

**for**(String **k:**khoa){

String temp="";

**for**(**int** i=**0**;i<k.length();i++){

**if**(k.charAt(i)!=','){

temp=temp+k.charAt(i);

}

}

s=s+temp+",";

}

System.out.println("{ "+s+" }");

}

**private** **static** ArrayList<String> **timKhoa**(String attr,ArrayList<String> pt) {

ArrayList<String> N=**new** ArrayList<>();

ArrayList<String> D=**new** ArrayList<>();

ArrayList<String> L= **new** ArrayList<>();

ArrayList<String> K= **new** ArrayList<>();

String Tvp="",Tvt="";

**for**(String **s:**pt){

String s1[]=s.split(":");

String vp=s1[**1**].trim();

String vt=s1[**0**].trim();

Tvp=Tvp+vp;

Tvt=Tvt+vt;

}

**for** (**int** i = **0**; i < attr.length(); i++) {

**if**(!xoaTrung(Tvp).contains(attr.charAt(i)+"") ||(!xoaTrung(Tvp).contains(attr.charAt(i)+"") && !xoaTrung(Tvt).contains(attr.charAt(i)+""))){

N.add(attr.charAt(i)+"");

}

**if**(!xoaTrung(Tvt).contains(attr.charAt(i)+""))

{

D.add(attr.charAt(i)+"");

}

**if** (xoaTrung(Tvp).contains(attr.charAt(i)+"") && xoaTrung(Tvt).contains(attr.charAt(i)+"")){

L.add(attr.charAt(i)+"");

}

}

ArrayList<String> temp= **new** ArrayList<>();

String s="";

**for**(String **k:**N){

s=s+","+k;

}

**for**(String **i:**L){

String x=s;

x=x+","+i;

temp.add(x.substring(**1**));

}

**for**(String **i:** temp){

**if**(baodong(pt,i).length()==attr.length()){

K.add(i);

}

}

**return** K;

}

**private** **static** String **attribute**(String s) {

String re="";

String temp=s.substring(**2**,s.length()-**1**);

**for**(**int** i=**0**;i<temp.length();i++){

**if**(!(temp.charAt(i)==','))

{

re=re+temp.charAt(i);

}

}

**return** re;

}

**private** **static** String **baodong**(ArrayList<String> pt,String c) {

String re="";

//System.out.println(pt);

**if**(c.length()>**1**){

String tt[]=c.split(",");

**for** (**int** i=**0**;i<tt.length;i++){

re=re+tt[i].toUpperCase();

}

**for**(**int** i=**0**;i<tt.length;i++){

**for**(String **s:**pt){

String s1[]=s.split(":");

String vt=s1[**0**];

String vp=s1[**1**];

**if**(vt.equals(tt[i].toUpperCase())){

re=re+vp;

}

}

}

**for**(**int** i=**0**;i<pt.size();i++){

**for**(String **s:**pt){

// System.out.println(s);

String s1[]=s.split(":");

String vt=s1[**0**].trim(); //gh

// System.out.println(vt);

String vp=s1[**1**].trim(); //C

**boolean** flag=**true**;

**for** (**int** j=**0**;j<vt.length();j++){ // =

//System.out.println(vt.charAt(j)+"/"+re.indexOf(vt.charAt(j)+""));

**if**(re.indexOf(vt.charAt(j)+"")==-**1**){

flag=**false**;

}

}

**if**(flag){

re=re+vp;

}

}

}

}

**else** {

re=c.toUpperCase();

**for**(String **s:**pt){

String s1[]=s.split(":");

String vt=s1[**0**];

String vp=s1[**1**];

**if**(vt.equals(c.toUpperCase())){

re=re+vp;

}

}

}

**return** **xoaTrung**(re);

}

**public** **static** String **xoaTrung**(String str) {

String re="";

**for**(**int** i=**0**;i<str.length();i++){

**if**(!re.contains(str.charAt(i)+"")){

re=re+str.charAt(i);

}

}

**return** re;

}

**public** **static** ArrayList<String> PhuThuocHam(ArrayList<String> data) {

String arr[]=data.get(1).trim().split(",");

ArrayList<String> phuthuoc= new ArrayList<>();

String tmp=arr[0].trim();

String temp="";

**for** (int i=0;i<tmp.length();i++){

**if** (tmp.charAt(i)=='{'){

temp=tmp.substring(i+1, tmp.length());

}

}

phuthuoc.add(temp);

**for** (int i=1;i<arr.length-1;i++){

phuthuoc.add(arr[i]);

}

String tmp1=arr[arr.length-1];

phuthuoc.add(tmp1.substring(0,tmp1.length()-1));

**return** phuthuoc;

}

**private** **static** ArrayList<String> **readFile**(String path) **throws** FileNotFoundException {

FileInputStream file = **new** FileInputStream(path);

Scanner sc = **new** Scanner(file);

ArrayList<String> result = **new** ArrayList<>();

**try** {

**while** (sc.hasNextLine()) {

String temp = sc.nextLine();

result.add(temp);

}

sc.close();

} **catch** (Exception e) {

System.out.println(e);

}

**return** result;

}

}

**--- HẾT ---**