**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**ĐỒ ÁN HỌC PHẦN**

TÊN HỌC PHẦN: **Hệ quản trị cơ sở dữ liệu (DBMS)**

MÃ SỐ LỚP HP: **DBMS330284\_01**

Tên đề tài: **HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU “HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT”**

**Họ tên sinh viên: Đỗ Trường Giang [07]**

**Mã số sinh viên: 23810010**

**Lớp: 23LCSP02 [VLVH\_A]**

**Ngày nộp: 25/05/2024**

**TP.HCM, ngày …. tháng .... năm 2024**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI ĐỒ ÁN HỌC PHẦN**

TÊN HỌC PHẦN: **Hệ quản trị cơ sở dữ liệu (DBMS)**

MÃ SỐ LỚP HP: **DBMS330284\_01**

Tên đề tài: **Hệ quản trị Cơ sở dữ liệu “Hệ thống tiền lương nhân viên của công ty Fe Credit [[1]](#footnote-1)"**

**Giảng viên giảng dạy: VÕ XUÂN THỂ**

**Họ tên sinh viên: Đỗ Trường Giang [07]**

**Mã số sinh viên: 23810010**

**Lớp: 23LCSP02 [VLVH\_A]**

**Tên các files sản phẩm đề tài:**

**Tên sản phẩm đề tài: 07\_DTGiang\_FCBMS.rar**

**Tên DB SQL: a07\_DTGiang\_FCBMS.bak**  (MS. SQL 2022: Version 16.0)

a**07\_DTGiang\_FCBMS**.mdf

**a07\_DTGiang\_FCBMS\_log**.ldf   
 D:\storage\_sql

**Tên máy chủ DB SQL : .** **\MAINSERVER**

**Tên đăng nhập DB SQL: sa Mật khẩu (password) đăng nhập SQL: 123**

**NHẬN XÉT CỦA GIẢNG VIÊN**

**………………………………………………………..**

**………………………………………………………..**

**………………………………………………………..**

**Ngày……./……../202…..**

**Giảng viên:**

**Ký tên**

**TP.HCM, ngày ……. tháng …… năm 202…….**

**LỜI CẢM ƠN**

Trước tiên, em xin gửi lời cảm ơn sâu sắc đến thầy Võ Xuân Thể, giảng viên môn Cơ sở dữ liệu, Trường Đại học Sư phạm Kỹ thuật TP.HCM. Thầy không chỉ là người hướng dẫn mà còn là người truyền cảm hứng, giúp em có được những kiến thức nền tảng và chuyên sâu về cơ sở dữ liệu. Sự tận tâm, nhiệt huyết trong từng bài giảng và những chỉ dẫn quý báu của thầy trong suốt quá trình thực hiện đề tài "Hệ thống quản lý lương nhân viên của công ty Fe Credit" đã giúp em có thể vượt qua những khó khăn và hoàn thành tốt bài nghiên cứu này. Mỗi lời khuyên, sự động viên và sự đồng hành của thầy đã đóng góp vô cùng lớn trong sự tiến bộ và thành công của em.

Em cũng xin chân thành cảm ơn các thầy cô trong Khoa Công nghệ Thông tin, Trường Đại học Sư phạm Kỹ thuật TP.HCM, những người đã không ngừng truyền đạt kiến thức, kỹ năng, và tạo môi trường học tập thuận lợi cho em. Những bài học quý giá từ các thầy cô đã giúp em có được nền tảng kiến thức vững chắc, từ đó tự tin áp dụng vào thực tế và thực hiện thành công đề tài này. Em cũng xin gửi lời cảm ơn đến các thầy cô ở những môn học khác, những người đã giúp em phát triển kỹ năng tư duy, phân tích và sáng tạo, những yếu tố không thể thiếu trong quá trình nghiên cứu và triển khai đề tài.

Bên cạnh đó, em xin gửi lời cảm ơn chân thành đến các anh chị trong công ty Fe Credit, đặc biệt là những người đã tạo điều kiện cho em được tham khảo và tìm hiểu về hệ thống quản lý lương nhân viên thực tế. Sự hỗ trợ và chia sẻ kinh nghiệm thực tiễn từ các anh chị đã giúp em rất nhiều trong việc xây dựng và phát triển mô hình cơ sở dữ liệu của đề tài.

Không thể không nhắc đến các bạn cùng lớp, những người đã luôn đồng hành, chia sẻ kinh nghiệm, và giúp đỡ nhau trong suốt quá trình học tập và nghiên cứu. Tinh thần đoàn kết, tương trợ lẫn nhau đã góp phần tạo nên một môi trường học tập thân thiện và tích cực, giúp em có động lực và niềm tin để hoàn thành đề tài này.

Cuối cùng, em xin bày tỏ lòng biết ơn sâu sắc đến gia đình và bạn bè. Sự ủng hộ, động viên tinh thần và tạo điều kiện thuận lợi từ gia đình đã giúp em tập trung toàn tâm toàn ý vào việc học tập và nghiên cứu. Những lời động viên kịp thời từ bạn bè đã giúp em vượt qua những thời điểm khó khăn, mất phương hướng trong quá trình thực hiện đề tài.

Tất cả những sự giúp đỡ và hỗ trợ từ thầy cô, gia đình, bạn bè và các anh chị trong công ty Fe Credit là nguồn động lực lớn lao giúp em hoàn thành tốt bài nghiên cứu này. Một lần nữa, em xin chân thành cảm ơn tất cả mọi người.

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| CSDL hoặc DB | Cơ sở dữ liệu: DataBase |
| HQTCSDL = DBMS | Hệ quản trị Cơ sở dữ liệu |
| HTTT | Hệ thống thông tin |
| NSD = UA = Users = Account | Người Sử Dụng = User Account = là quyền làm việc trên WebApp được đảm bảo bởi tối thiểu là 2 yếu tố: tên đăng nhập (UserName) và mật khẩu (Password) |
| RBTV | Ràng buộc toàn vẹn dữ liệu |
|  |  |
| DCL | Data Control Language: Ngôn ngữ điều khiển dữ liệu |
| DDL | Data Definition Language: Ngôn ngữ định nghĩa dữ liệu |
| DML | Data Manipulation Language: Ngôn ngữ thao tác dữ liệu |
| NF | Dạng chuẩn: Normal Form |
| SP | Store Procedure: Thủ tục xử lý dữ liệu được lưu trữ trên DBMS |
| SQL | Ngôn ngữ vấn tin có cấu trúc: Structured Query Language |
| FCPMS | Fe Credit Payroll Management System |

**DANH MỤC CÁC THUẬT NGỮ**

|  |  |
| --- | --- |
| Giao diện người dùng:  User-Interface | Là hệ thống các màn hình giao tiếp cho phép người sử dụng tương tác với các thành phần phần mềm trong HTTT, thường là 1 trong các dong: Win-form, Web-form, Mobile-Form. |
| Tài khoản (Account) | Là một quyền làm việc trên HTTT được cấp phát cho một cá nhân thông qua tên tài khoản (username) và mật khẩu (password). |
| Record: Bảng tin = Mẫu tin | Là một dòng dữ liệu trong 1 bảng dữ liệu của CSDL |

**MỤC LỤC**

[1. DANH MỤC CÁC TỪ VIẾT TẮT 5](#_Toc177133051)

[Chương 1: GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN 9](#_Toc177133052)

[1.1. Tổng quan về Đồ Án Học Phần 9](#_Toc177133053)

[1.2. Nội dung chuyên môn chính của ĐỒ ÁN HỌC PHẦN 9](#_Toc177133054)

[1.3. Công cụ thực hiện ĐỒ ÁN HỌC PHẦN 9](#_Toc177133055)

[1.4. Bố cục của báo cáo 10](#_Toc177133056)

[1.5. Giới thiệu yêu cầu DB của đề tài 11](#_Toc177133057)

[1.5.1. Giới thiệu nghiệp vụ tính lương 11](#_Toc177133058)

[1.5.2. Mô tả hệ thống Cơ sở dữ liệu “HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT" 11](#_Toc177133059)

[1.5.3. ERD 12](#_Toc177133060)

[1.5.4. Danh sách LĐQH 13](#_Toc177133061)

[1.5.5. Sơ đồ vậy lý (Diagrams trên MS. SQL Server) 13](#_Toc177133062)

[Chương 2: CƠ SỞ LÝ THUYẾT CỦA HỌC PHẦN 14](#_Toc177133063)

[2.1. Tổng quan về DBMS 14](#_Toc177133064)

[2.2. Công cụ Quản trị CSDL 14](#_Toc177133065)

[2.3. Tập lệnh SQL (Thông dịch) 14](#_Toc177133066)

[2.4. Lập trình biên dịch trên MS. SQL Server 14](#_Toc177133067)

[2.5. Giới thiệu một số công cụ kết nối CSDL 14](#_Toc177133068)

[3.1. Quản trị tài khoản (Account) và an toàn bảo mật hệ thống CSDL 15](#_Toc177133069)

[3.2. Cài đặt hệ thống CSDL “Hệ thống quản lý tiền lương nhân viên của công ty Fe Credit” trên DBMS MS. SQL Server 16](#_Toc177133070)

[3.2.1. Lập CSDL mới 16](#_Toc177133071)

[3.2.2. Lập mới các Bảng dữ liệu (Tables) 20](#_Toc177133072)

[3.2.3. Lập Sơ đồ vật lý của Database (Diagram) 27](#_Toc177133073)

[3.3. Di chuyển hệ thống CSDL MS. SQL Server từ Server này đến Server khác 27](#_Toc177133074)

[4.1. Các lệnh thao tác xử lý trên CSDL trong MS. SQl Server (DML SQL) 31](#_Toc177133075)

[4.1.1. Thêm mẫu tin (bảng tin: Record) mới vào một Table trong một CSDL: Insert… 31](#_Toc177133076)

[4.1.2. Xóa mẫu tin (Record) trong một Table của CSDL: Delete.. 36](#_Toc177133077)

[4.1.3. Sửa chữa thông tin của mẫu tin (Record) trong một Table của CSDL: Update.. 36](#_Toc177133078)

[4.2. Vấn tin SQL trên MS. SQL Server trên CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT(DML) 37](#_Toc177133079)

[4.3. Cài đặt các View trên MS. SQL Server trên CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT (DML) 39](#_Toc177133080)

[4.4. Lập các Store Procedure (SP) và các Trigger cho các Table của DB 41](#_Toc177133081)

[4.5. Lập bảng sao CSDL trên hệ quản trị MS. SQL Server đối với CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT 47](#_Toc177133082)

[4.6. Kết xuất (Export) và Nạp (Import) DL trên hệ quản trị MS. SQL Server đối với CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT 48](#_Toc177133083)

[2. DANH MỤC TÀI LIỆU THAM KHẢO 51](#_Toc177133084)

[3. CÁC PHỤ LỤC 52](#_Toc177133085)

# GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN

## Tổng quan về Đồ Án Học Phần

Đồ án Học phần Hệ quản trị cơ sở dữ liệu “Hệ thống tiền lương nhân viên của công ty Fe Credit” vận dụng kiến thức và kỹ năng về DBMS trên MS. SQL Server đề tổ chức thiết lập \_ lưu trữ \_ duy trì và khai thác sử dụng Hệ thống cơ sở dữ liệu “Hệ thống tiền lương nhân viên của công ty Fe Credit”

## Nội dung chuyên môn chính của ĐỒ ÁN HỌC PHẦN

Đồ án tập tổ chức cài đặt Hệ thống CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” trên Hệ quản trị CSDL MS. SQL Server với các nội dung chuyên môn như sau:

+ Mô tả Hệ thống CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” sẽ được tổ chức trên DBMS SQL Server

+ Thiết lập và định nghĩa CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” dùng DDL.

+ Thực hiện các thiết lập an toàn vào bảo mật CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit”

+ Cài đặt các điều khiển cần thiết cho CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” dùng DCL

**+ Cài đặt các thủ tục thao tác CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” dùng DML**

+ Cài đặt các SP (Store Procedure) phục vụ khai thác sử dụng CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit”

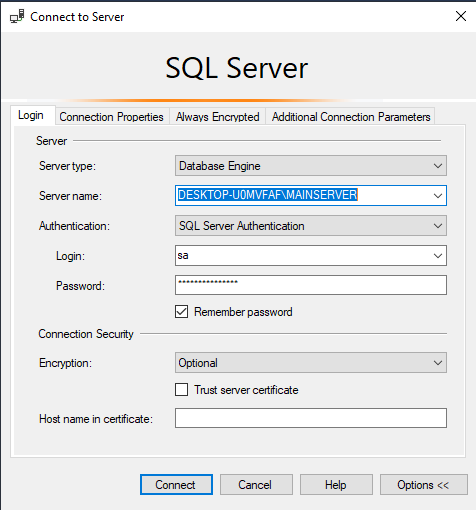
+ Cài đặt các Trigger phục vụ đảm bảo RBTV CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit”

+ Cài đặt các nền tảng kết nối CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit”

## Công cụ thực hiện ĐỒ ÁN HỌC PHẦN

SQL Server Management Studio 2022, version: 16.0

Thủ tục đăng nhập vào DBMS được thực hiện như Hình 1-1



*Hình 1-1. Màn hình đăng nhập MS. SQL Server*

## Bố cục của báo cáo

Báo cáo gồm những nội dung như sau:

Chương 1 Giới thiệu đồ án của Học phần

Chương 2 Các cơ sở lý thuyết phục việc thực hiện đề tài Học phần

Chương 3 Tổ chức hệ thống CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” trên DBMS MS. SQL Server (bao gồm bảo mật)*.*

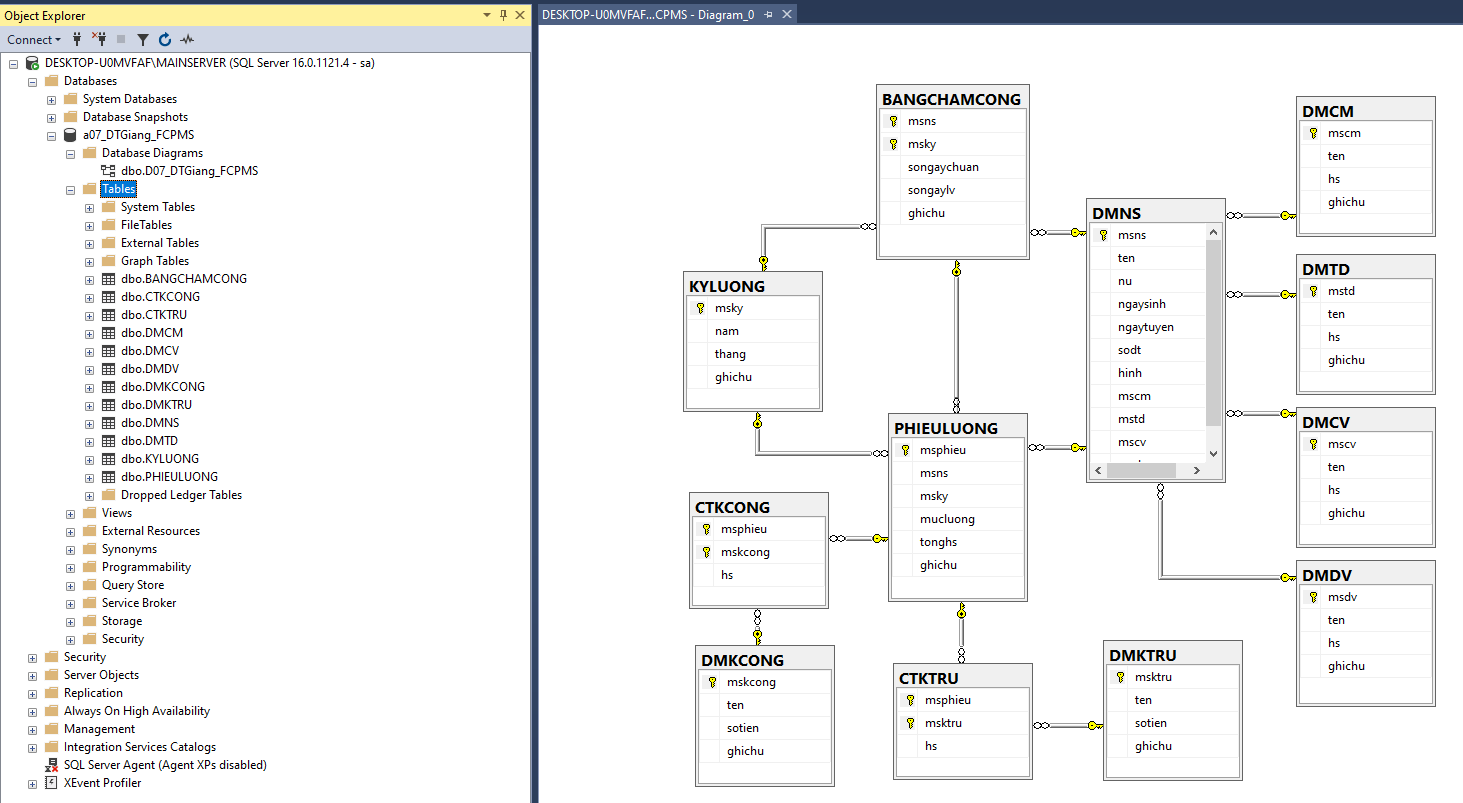
Chương 4 Cài đặt các thủ tục khai thác sử dụng CSDL “Hệ thống tiền lương nhân viên của công ty Fe Credit” trên DBMS MS. SQL Server*.*

Chương 5 Tổng kết các kết quản đạt được và còn hạn chế của đồ án, đồng thời đề xuất hướng khắc phục hạn chế và phát triển Đồ án.

## Giới thiệu yêu cầu DB của đề tài

## Giới thiệu nghiệp vụ tính lương

Kế toán NS\_lương: công thức tính lương (phổ biến): = ngày (giờ) công \* hệ số lương \* lương căn bản (min) + phụ cấp [chức vụ, độc hại, nguy hiểm, vùng xa…] – BHXH {BH YT, BH TN, BH Tai nạn,,..}



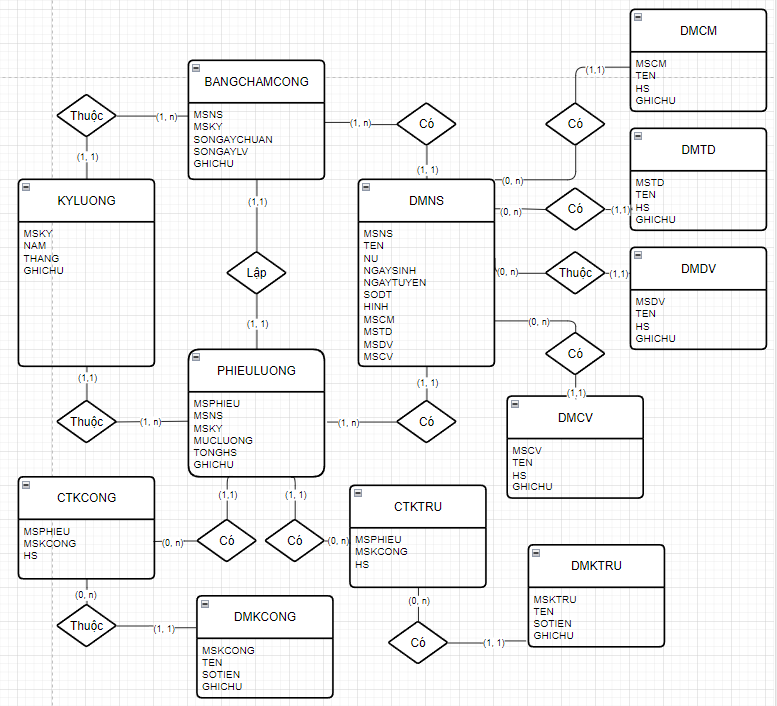
## Mô tả hệ thống Cơ sở dữ liệu “HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT"

Xét một phần (trích) CSDL **“HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT”** được mô tả như sau:

* Kỳ lương: mã kỳ lương, năm, tháng, ghi chú.
* Bảng chấm công: mã nhân sự, mã kỳ lương, số ngày chuẩn, số ngày làm việc, ghi chú.
* Phiếu lương: mã phiếu lương, mã số nhân sự, mã số kỳ lương, mức lương, tổng hệ số, ghi chú.
* Danh mục nhân sự: mã nhân sự, tên nhân sự, giới tính, ngày sinh, ngày tuyển, số điện thoại, hình, mã số chuyên môn, mã số trình độ, mã số đơn vị, mã số chức vụ.
* Danh mục chuyên môn: mã chuyên môn, tên chuyên môn, hệ số, ghi chú.
* Danh mục trình độ: mã trình độ, tên trình độ, hệ số, ghi chú.
* Danh mục đơn vị: mã đơn vị, tên đơn vị, hệ số, số điện thoại.
* Danh mục chức vụ: mã chức vụ, tên chức vụ, hệ số, ghi chú.
* Danh mục các khoảng trừ: mã khoảng trừ, tên khoản trừ, số tiền, ghi chú.
* Chi tiết các khoản trừ: mã khoản trừ, mã phiếu lương, hệ số.
* Danh mục các khoảng cộng: mã khoảng cộng, tên khoảng cộng, số tiền, ghi chú.
* Chi tiết các khoảng cộng: mã số khoảng cộng, mã phiếu lương, hệ số.

## ERD

ERD của DB quản lý hệ thống tiền lương nhân viên của công ty Fe Credit như Hình 1.5.3-1



Hình 1.5.3‑1. ERD của DB Hệ thống tiền lương nhân viên của công ty Fe Credit.

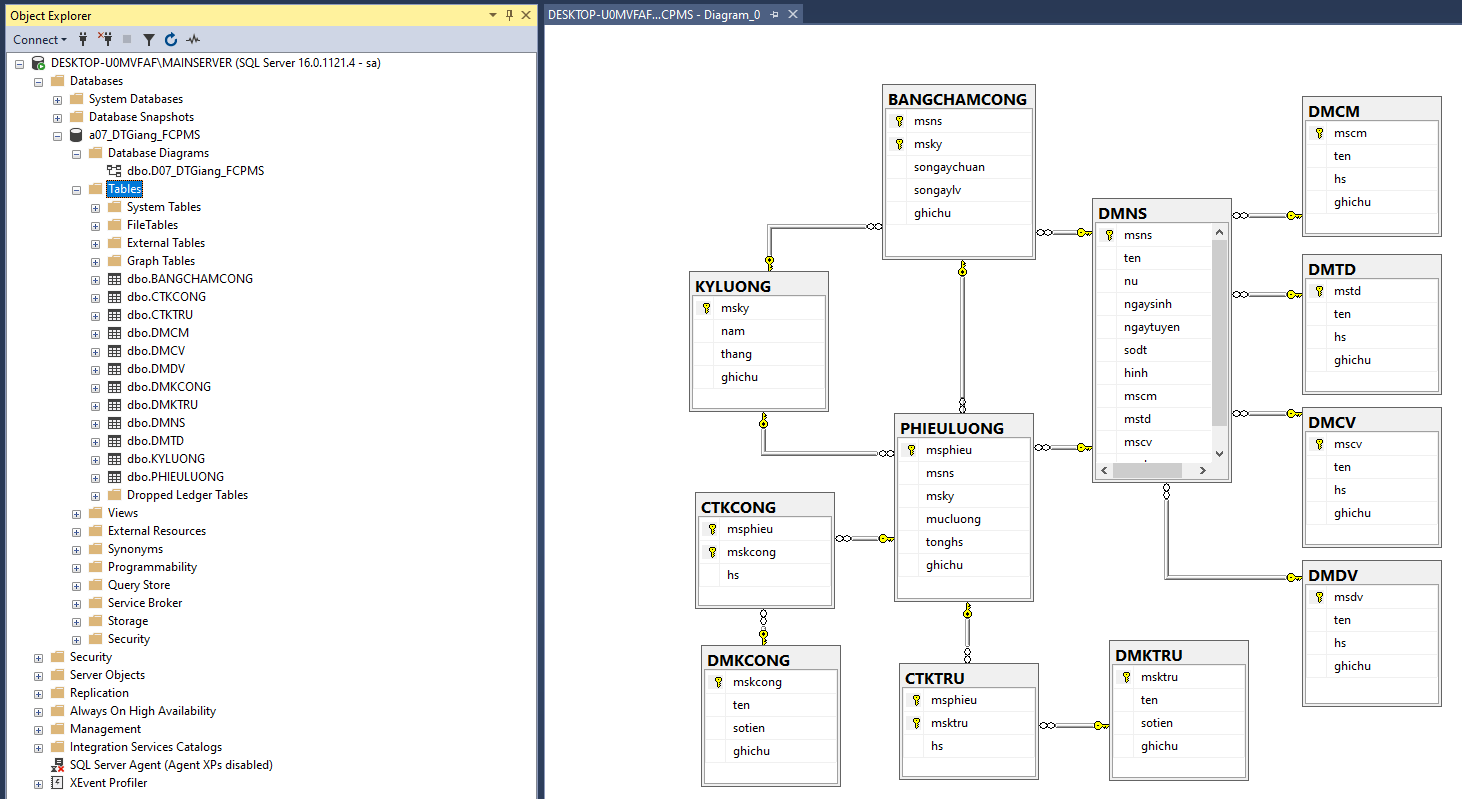
## Danh sách LĐQH

Danh sách LĐQH của DB quản lý hệ thống tiền lương nhân viên của công ty Fe Credit như sau:

* **BANGCHAMCONG**(MSNS, MSKY, SONGAYCHUAN, SONGAYLV, GHICHU)
* **KYLUONG**(MSKY, NAM, THANG, GHICHU)
* **PHIEULUONG**(MSPHIEU, MSNS, MSKY, MUCLUONG, TONGHS, GHICHU)
* **DMNS**(MSNS, TENNS, NU, NGAYSINH, NGAYTUYEN, SODT, HINH, MSCM, MSTD, MSCV)
* **DMCM**(MSCM, TENCM, HS, GHICHU)
* **DMTD**(MSTD, TENTD, HS, GHICHU)
* **DMDV**(MSDV, TENDV, DC, SODT)
* **DMCV**(MSCV, TENCV, HS, GHICHU)
* **DMKTRU**(MSKTRU, TENKTRU, SOTIEN, GHICHU)
* **CTKTRU**(MSPHIEU, MSKTRU, HS)
* **DMKCONG**(MSKCONG, TENKCONG, SOTIEN, GHICHU)
* **CTKCONG**(MSPHIEU, MSKCONG, HS)

## Sơ đồ vậy lý (Diagrams trên MS. SQL Server)

Sơ đồ vậy lý (Diagrams) của DB HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT như Hình 1.5.5‑2



Hình 1.5.5‑2. Sơ đồ vật lý của DB Hệ thống tiền lương nhân viên của công ty Fe Credit

# CƠ SỞ LÝ THUYẾT CỦA HỌC PHẦN

## Tổng quan về DBMS

+ Tổng quan về Hệ quản trị Cơ sở sữ liệu (DBMS) & Ôn tập CSDL

## Công cụ Quản trị CSDL

+ Giới thiệu Hệ quản trị CSDL MS.SQL Server

+ Các đối tượng quản lý trong SQl Server : DataBase, Table, View, Diagram, Index, …

+ An toàn và Bảo mật trong SQL Server : Authenticate, sa, … Backup\_Restore, Detach\_Attach,…

## Tập lệnh SQL (Thông dịch)

+ Tập lệnh SQL cơ bản

+ Tập lệnh định nghĩa dữ liệu [DDL] : Create, Drop,…

+ Tập lệnh điều khiển dữ liệu [DCL] : Grant, Revote, . . .

+ Tập lệnh thao tác dữ liệu [DML] : Insert, Delete, Update, . . .

+ DML: Vấn tin (Truy vấn) dữ liệu : Select…

## Lập trình biên dịch trên MS. SQL Server

+ Giới thiệu T\_SQL

+ Thủ tục lưu trữ (SP : Store Procedure)

+ Trigger và xử lý RBTV DL

## Giới thiệu một số công cụ kết nối CSDL

+ Data Link OLE DB với .udl file

+ ODBC Data Sources (64/32-bit)

+ Quản trị Các Services của MS. SQl Server

+ Giới thiệu một số vấn đề liên quan : Distribution, DWH, DMN, BigData, . . .

# TỔ CHỨC HỆ THỐNG CƠ SỞ DỮ LIỆU “HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT.” TRÊN DBMS MS. SQL SERVER*.*

## Quản trị tài khoản (Account) và an toàn bảo mật hệ thống CSDL

\* SQL Server luôn có UA cấp cao nhất : sa (System Administrator)

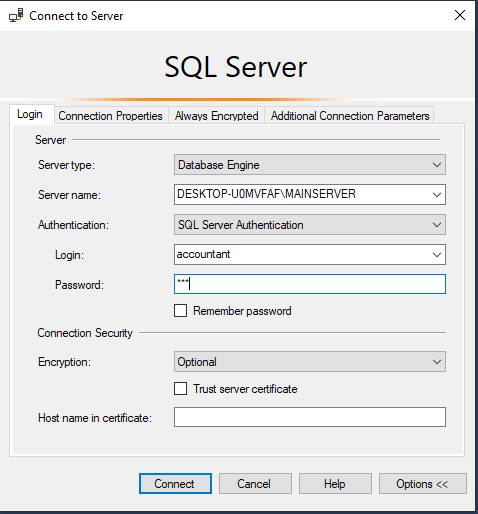
\* Ngoài ra, khi quản trị DB, người quản trị (sa) phải cấp UA theo chức năng từng đơn vị :

Trong đề tài :

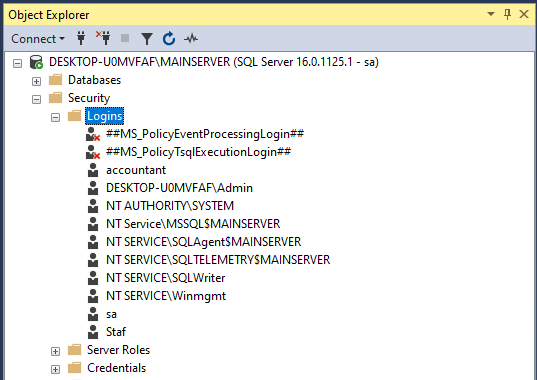
+ **Accountant**: UA dành cho bộ phận kế toán của FE Credit: được thực hiện các thao tác đối với DB BANGCHAMCONG, PHIEULUONG, … [[2]](#footnote-2).

+ **Staff** : UA dành cho bộ phận nhân viên còn lại của FE Credit: chỉ được thực hiện các thao tác liên quan đến Read (Select, View, …) [[3]](#footnote-3).

. Các DB User Account nếu trên có Password ban đầu: **123**, như Hình 2-1



*Hình 2-1. Cấp DB UA cho bộ phận kế toán của FE Credit*



*Hình 2-2. Kết quả cấp DB UA cho hệ thống FE Credit*

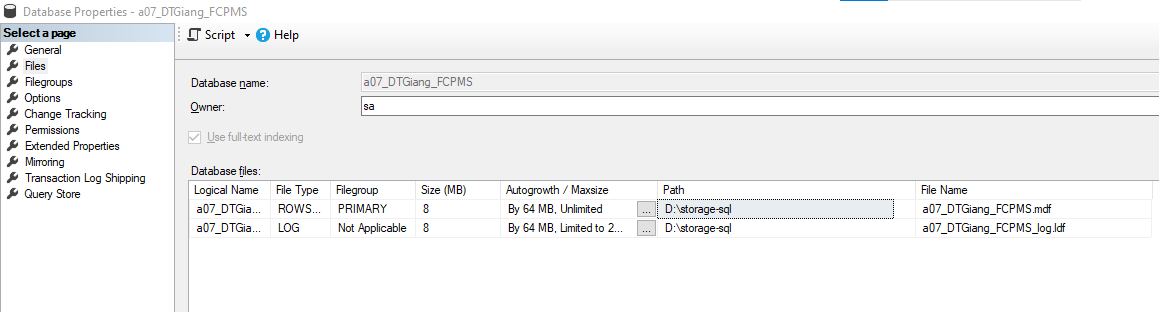
## Cài đặt hệ thống CSDL “Hệ thống quản lý tiền lương nhân viên của công ty Fe Credit” trên DBMS MS. SQL Server

### Lập CSDL mới

* Ten D: **a07\_DTGiang\_FCPMS**

1. Nơi lưu DB:

D:\storage-sql



Hình 3.2‑1 Màn hình lập CSDL mới "Hệ thống quản lý tiền lương nhân viên của công ty Fe Credit"

* **11\_07DTGiang\_FCPMS\_CreateDB.sql**

USE [master]

GO

/\*\*\*\*\*\* Object: Database [a07\_DTGiang\_FCPMS] Script Date: 8/29/2024 8:07:44 PM \*\*\*\*\*\*/

CREATE DATABASE [a07\_DTGiang\_FCPMS]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'a07\_DTGiang\_FCPMS', FILENAME = N'D:\storage-sql\a07\_DTGiang\_FCPMS.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )

LOG ON

( NAME = N'a07\_DTGiang\_FCPMS\_log', FILENAME = N'D:\storage-sql\a07\_DTGiang\_FCPMS\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )

WITH CATALOG\_COLLATION = DATABASE\_DEFAULT, LEDGER = OFF

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [a07\_DTGiang\_FCPMS].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ARITHABORT OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET AUTO\_CLOSE OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET DISABLE\_BROKER

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET RECOVERY FULL

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET MULTI\_USER

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET DB\_CHAINING OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET TARGET\_RECOVERY\_TIME = 60 SECONDS

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET ACCELERATED\_DATABASE\_RECOVERY = OFF

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET QUERY\_STORE = ON

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET QUERY\_STORE (OPERATION\_MODE = READ\_WRITE, CLEANUP\_POLICY = (STALE\_QUERY\_THRESHOLD\_DAYS = 30), DATA\_FLUSH\_INTERVAL\_SECONDS = 900, INTERVAL\_LENGTH\_MINUTES = 60, MAX\_STORAGE\_SIZE\_MB = 1000, QUERY\_CAPTURE\_MODE = AUTO, SIZE\_BASED\_CLEANUP\_MODE = AUTO, MAX\_PLANS\_PER\_QUERY = 200, WAIT\_STATS\_CAPTURE\_MODE = ON)

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET READ\_WRITE

GO

+ Tên CSDL (Database Name):

**a07\_DTGiang\_FCPMS.mdf** : File CSDL

**a07\_DTGiang\_FCPMS\_log.ldf**: Log File

+ Owner (Sở hữu DB): **sa**

* **Câu lệnh Sao lưu CSDL (Backup DataBase): 12\_07DTGiang\_FCPMS\_BackupDB.sql**

-- ===========================

-- Backup database

BACKUP DATABASE a07\_DTGiang\_FCPMS

TO DISK = N'D:\storage-sql\a07\_DTGiang\_FCPMS\_BackupDB.bak'

-- Backup log

BACKUP LOG a07\_DTGiang\_FCPMS

TO DISK = N'D:\storage-sql\a07\_DTGiang\_FCPMS\_BackupDB.bak'

WITH

NO\_TRUNCATE ,

NORECOVERY,

NOFORMAT,

NOINIT,

NAME = 'a07\_DTGiang\_FCPMS\_BackupDB',

SKIP,

NOREWIND,

NOUNLOAD,

STATS = 10

GO

* **Câu lệnh Phục hồi (đã sao lưu trước đó) CSDL (Restore DataBase): 13\_07DTGiang\_FCPMS\_RestoreDB.sql**

-- =======================================

-- Restore File and Filegroups

-- =======================================

-- Restore filegroups - one at a time

RESTORE DATABASE a07\_DTGiang\_FCPMS

FILE = N'a07\_DTGiang\_FCPMS'

FROM DISK = N'D:\storage-sql\a07\_DTGiang\_FCPMS\_BackupDB.bak'

WITH

FILE = 1,

NORECOVERY,

NOUNLOAD,

STATS = 10

GO

-- Restore log

RESTORE LOG a07\_DTGiang\_FCPMS

FROM DISK = N'D:\storage-sql\a07\_DTGiang\_FCPMS\_BackupDB.bak'

WITH

FILE = 2,

NOUNLOAD,

STATS = 10

GO

* **Câu lệnh Hủy bỏ CSDL (Xóa bỏ DataBase)**: 14\_07DTGiang\_FCPMS\_DropDB.sql

-- =========================

-- Drop Database

-- =========================

USE master

GO

IF EXISTS (

SELECT name

FROM sys.databases

WHERE name = N'a07\_DTGiang\_FCPMS'

)

DROP DATABASE [a07\_DTGiang\_FCPMS]

GO

### Lập mới các Bảng dữ liệu (Tables)

**2\_07\_DTGiang\_FCPMS\_TableDB.sql**

Thứ tự các Table được lập như sau (*PK lập trước FK & PK đơn lập trước PK tổ hợp*):

USE [master]

GO

/\* Create Tables \*/

/\*\*\*\*\*\* Object: Table [dbo].[BANGCHAMCONG] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[BANGCHAMCONG](

[msns] [nchar](20) NOT NULL,

[msky] [nchar](20) NOT NULL,

[songaychuan] [float] NULL,

[songaylv] [int] NULL,

[ghichu] [nvarchar](50) NULL,

CONSTRAINT [PK\_BANGCHAMCONG] PRIMARY KEY CLUSTERED

(

[msns] ASC,

[msky] 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].[CTKCONG] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[CTKCONG](

[msphieu] [nchar](20) NOT NULL,

[mskcong] [nchar](20) NOT NULL,

[hs] [float] NULL,

CONSTRAINT [PK\_CTKCONG] PRIMARY KEY CLUSTERED

(

[msphieu] ASC,

[mskcong] 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].[CTKTRU] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[CTKTRU](

[msphieu] [nchar](20) NOT NULL,

[msktru] [nchar](20) NOT NULL,

[hs] [float] NULL,

CONSTRAINT [PK\_CTKTRU] PRIMARY KEY CLUSTERED

(

[msphieu] ASC,

[msktru] 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].[DMCM] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMCM](

[mscm] [nchar](20) NOT NULL,

[tencm] [nvarchar](50) NULL,

[hs] [float] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMCM] PRIMARY KEY CLUSTERED

(

[mscm] 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].[DMCV] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMCV](

[mscv] [nchar](20) NOT NULL,

[tencv] [nvarchar](50) NULL,

[hs] [float] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMCV] PRIMARY KEY CLUSTERED

(

[mscv] 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].[DMDV] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMDV](

[msdv] [nchar](20) NOT NULL,

[tendv] [nvarchar](50) NULL,

[hs] [float] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMDV] PRIMARY KEY CLUSTERED

(

[msdv] 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].[DMKCONG] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMKCONG](

[mskcong] [nchar](20) NOT NULL,

[tenkcong] [nvarchar](50) NULL,

[sotien] [int] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMKCONG] PRIMARY KEY CLUSTERED

(

[mskcong] 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].[DMKTRU] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMKTRU](

[msktru] [nchar](20) NOT NULL,

[tenktru] [nvarchar](50) NULL,

[sotien] [int] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMKTRU] PRIMARY KEY CLUSTERED

(

[msktru] 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].[DMNS] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMNS](

[msns] [nchar](20) NOT NULL,

[tenns] [nvarchar](50) NULL,

[nu] [bit] NULL,

[ngaysinh] [date] NULL,

[ngaytuyen] [date] NULL,

[sodt] [nchar](20) NULL,

[hinh] [nchar](100) NULL,

[mscm] [nchar](20) NULL,

[mstd] [nchar](20) NULL,

[mscv] [nchar](20) NULL,

[msdv] [nchar](20) NULL,

CONSTRAINT [PK\_DMNS] PRIMARY KEY CLUSTERED

(

[msns] 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].[DMTD] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DMTD](

[mstd] [nchar](20) NOT NULL,

[tentd] [nvarchar](50) NULL,

[hs] [float] NULL,

[ghichu] [nvarchar](150) NULL,

CONSTRAINT [PK\_DMTD] PRIMARY KEY CLUSTERED

(

[mstd] 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].[KYLUONG] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[KYLUONG](

[msky] [nchar](20) NOT NULL,

[nam] [int] NULL,

[thang] [int] NULL,

[ghichu] [nvarchar](50) NULL,

CONSTRAINT [PK\_KYLUONG] PRIMARY KEY CLUSTERED

(

[msky] 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].[PHIEULUONG] Script Date: 9/9/2024 4:02:32 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[PHIEULUONG](

[msphieu] [nchar](20) NOT NULL,

[msns] [nchar](20) NULL,

[msky] [nchar](20) NULL,

[mucluong] [float] NULL,

[tonghs] [float] NULL,

[ghichu] [nvarchar](50) NULL,

CONSTRAINT [PK\_PHIEULUONG] PRIMARY KEY CLUSTERED

(

[msphieu] 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

/\* Lệnh tạo mối liên kêt \*/

ALTER TABLE [dbo].[BANGCHAMCONG] WITH CHECK ADD CONSTRAINT [FK\_BANGCHAMCONG\_DMNS] FOREIGN KEY([msns])

REFERENCES [dbo].[DMNS] ([msns])

GO

ALTER TABLE [dbo].[BANGCHAMCONG] CHECK CONSTRAINT [FK\_BANGCHAMCONG\_DMNS]

GO

ALTER TABLE [dbo].[BANGCHAMCONG] WITH CHECK ADD CONSTRAINT [FK\_BANGCHAMCONG\_KYLUONG] FOREIGN KEY([msky])

REFERENCES [dbo].[KYLUONG] ([msky])

GO

ALTER TABLE [dbo].[BANGCHAMCONG] CHECK CONSTRAINT [FK\_BANGCHAMCONG\_KYLUONG]

GO

ALTER TABLE [dbo].[CTKCONG] WITH CHECK ADD CONSTRAINT [FK\_CTKCONG\_DMKCONG] FOREIGN KEY([mskcong])

REFERENCES [dbo].[DMKCONG] ([mskcong])

GO

ALTER TABLE [dbo].[CTKCONG] CHECK CONSTRAINT [FK\_CTKCONG\_DMKCONG]

GO

ALTER TABLE [dbo].[CTKCONG] WITH CHECK ADD CONSTRAINT [FK\_CTKCONG\_PHIEULUONG] FOREIGN KEY([msphieu])

REFERENCES [dbo].[PHIEULUONG] ([msphieu])

GO

ALTER TABLE [dbo].[CTKCONG] CHECK CONSTRAINT [FK\_CTKCONG\_PHIEULUONG]

GO

ALTER TABLE [dbo].[CTKTRU] WITH CHECK ADD CONSTRAINT [FK\_CTKTRU\_DMKTRU] FOREIGN KEY([msktru])

REFERENCES [dbo].[DMKTRU] ([msktru])

GO

ALTER TABLE [dbo].[CTKTRU] CHECK CONSTRAINT [FK\_CTKTRU\_DMKTRU]

GO

ALTER TABLE [dbo].[CTKTRU] WITH CHECK ADD CONSTRAINT [FK\_CTKTRU\_PHIEULUONG] FOREIGN KEY([msphieu])

REFERENCES [dbo].[PHIEULUONG] ([msphieu])

GO

ALTER TABLE [dbo].[CTKTRU] CHECK CONSTRAINT [FK\_CTKTRU\_PHIEULUONG]

GO

ALTER TABLE [dbo].[DMNS] WITH CHECK ADD CONSTRAINT [FK\_DMNS\_DMCM] FOREIGN KEY([mscm])

REFERENCES [dbo].[DMCM] ([mscm])

GO

ALTER TABLE [dbo].[DMNS] CHECK CONSTRAINT [FK\_DMNS\_DMCM]

GO

ALTER TABLE [dbo].[DMNS] WITH CHECK ADD CONSTRAINT [FK\_DMNS\_DMCV] FOREIGN KEY([mscv])

REFERENCES [dbo].[DMCV] ([mscv])

GO

ALTER TABLE [dbo].[DMNS] CHECK CONSTRAINT [FK\_DMNS\_DMCV]

GO

ALTER TABLE [dbo].[DMNS] WITH CHECK ADD CONSTRAINT [FK\_DMNS\_DMDV1] FOREIGN KEY([msdv])

REFERENCES [dbo].[DMDV] ([msdv])

GO

ALTER TABLE [dbo].[DMNS] CHECK CONSTRAINT [FK\_DMNS\_DMDV1]

GO

ALTER TABLE [dbo].[DMNS] WITH CHECK ADD CONSTRAINT [FK\_DMNS\_DMTD] FOREIGN KEY([mstd])

REFERENCES [dbo].[DMTD] ([mstd])

GO

ALTER TABLE [dbo].[DMNS] CHECK CONSTRAINT [FK\_DMNS\_DMTD]

GO

ALTER TABLE [dbo].[PHIEULUONG] WITH CHECK ADD CONSTRAINT [FK\_PHIEULUONG\_BANGCHAMCONG] FOREIGN KEY([msns], [msky])

REFERENCES [dbo].[BANGCHAMCONG] ([msns], [msky])

GO

ALTER TABLE [dbo].[PHIEULUONG] CHECK CONSTRAINT [FK\_PHIEULUONG\_BANGCHAMCONG]

GO

ALTER TABLE [dbo].[PHIEULUONG] WITH CHECK ADD CONSTRAINT [FK\_PHIEULUONG\_DMNS] FOREIGN KEY([msns])

REFERENCES [dbo].[DMNS] ([msns])

GO

ALTER TABLE [dbo].[PHIEULUONG] CHECK CONSTRAINT [FK\_PHIEULUONG\_DMNS]

GO

ALTER TABLE [dbo].[PHIEULUONG] WITH CHECK ADD CONSTRAINT [FK\_PHIEULUONG\_KYLUONG] FOREIGN KEY([msky])

REFERENCES [dbo].[KYLUONG] ([msky])

GO

ALTER TABLE [dbo].[PHIEULUONG] CHECK CONSTRAINT [FK\_PHIEULUONG\_KYLUONG]

GO

USE [master]

GO

ALTER DATABASE [a07\_DTGiang\_FCPMS] SET READ\_WRITE

GO

* **CÂU LỆNH HỦY TABLE; VD Table DMDV trong DB: [a07\_DTGiang\_FCPMS]**

B1. Xác định DB

USE [a07\_DTGiang\_FCPMS]

GO

B2. Hủy Table trong DB đó

/\*\*\*\*\*\* Object: Table [dbo].[DMDV] Script Date: 9/11/2024 2:22:59 PM \*\*\*\*\*\*/

IF EXISTS (SELECT \* FROM sys.objects WHERE object\_id = OBJECT\_ID(N'[dbo].[DMDV]') AND type in (N'U'))

DROP TABLE [dbo].[DMDV]

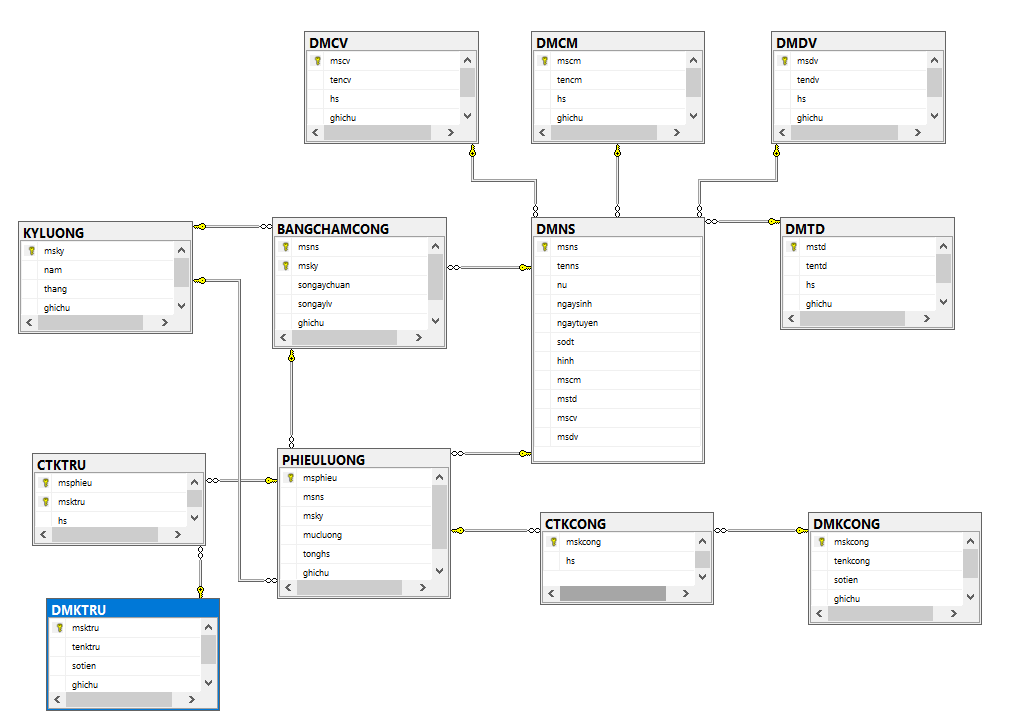
GO

\* Viết gọn:

**DROP TABLE** DMDV

GO

### Lập Sơ đồ vật lý của Database (Diagram)



*Hình 2-3. Sơ đồ vật lý của CSDL trên Hệ quản trị SQl Server*

## Di chuyển hệ thống CSDL MS. SQL Server từ Server này đến Server khác

* **Có 2 phương pháp** (cách) cơ bản:

+ Backup (Sao lưu): từ Server này & Restore (Phục hồi) đến Server mới

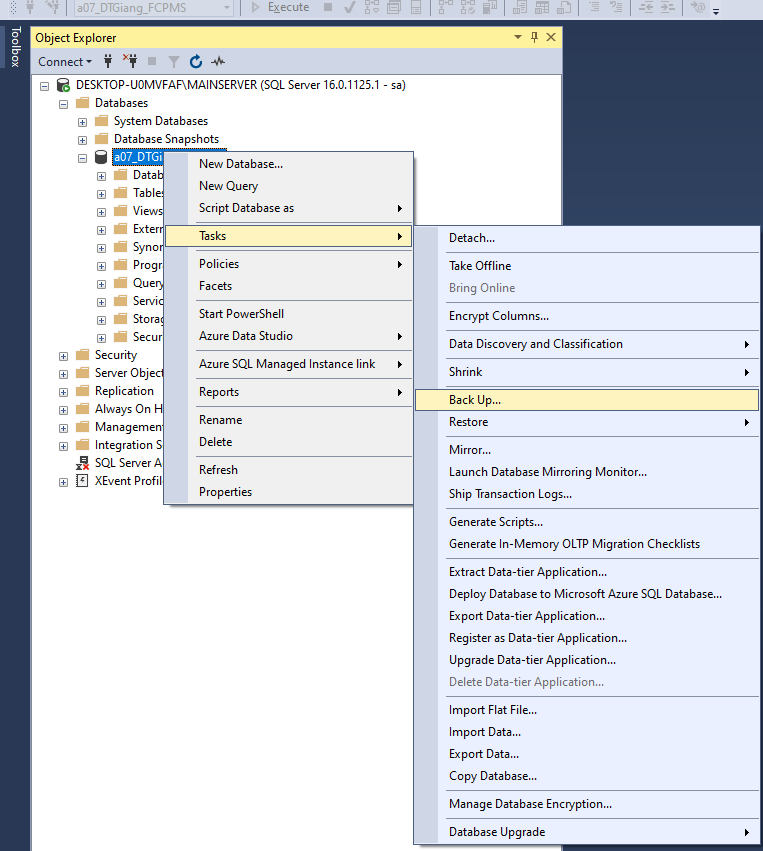
+ Detach (Tháo gỡ): từ Server này & Attach (Gán vào) Server mới

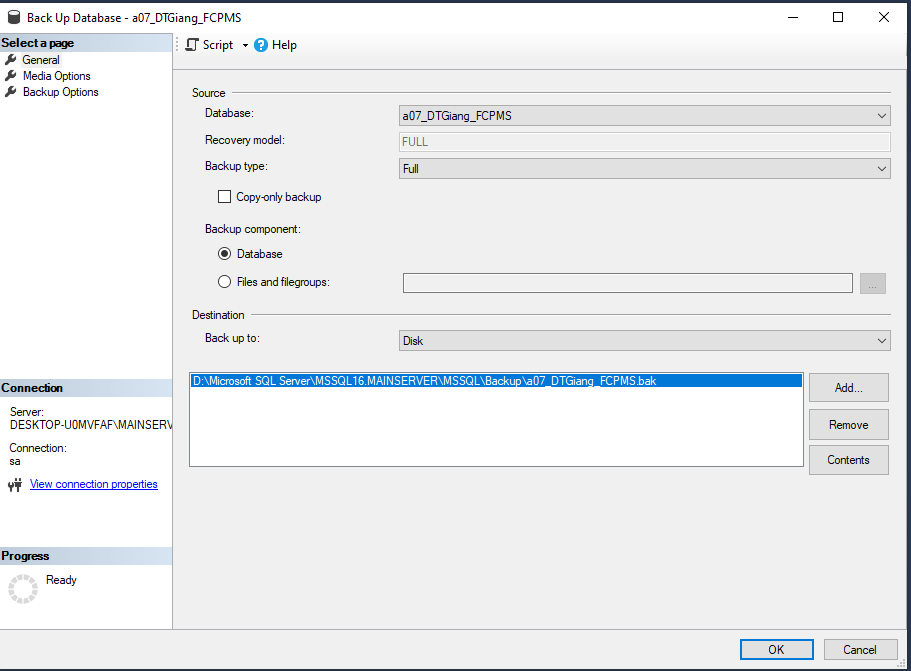
* **Backup & Restore**

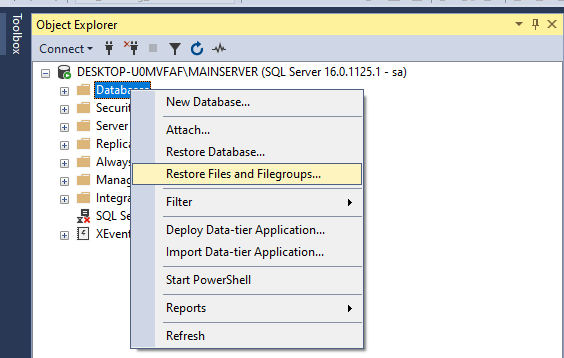
. DB trên Server source (nguồn) vẫn nguyên vẹn (Các App dùng DB vẫn bình thường)

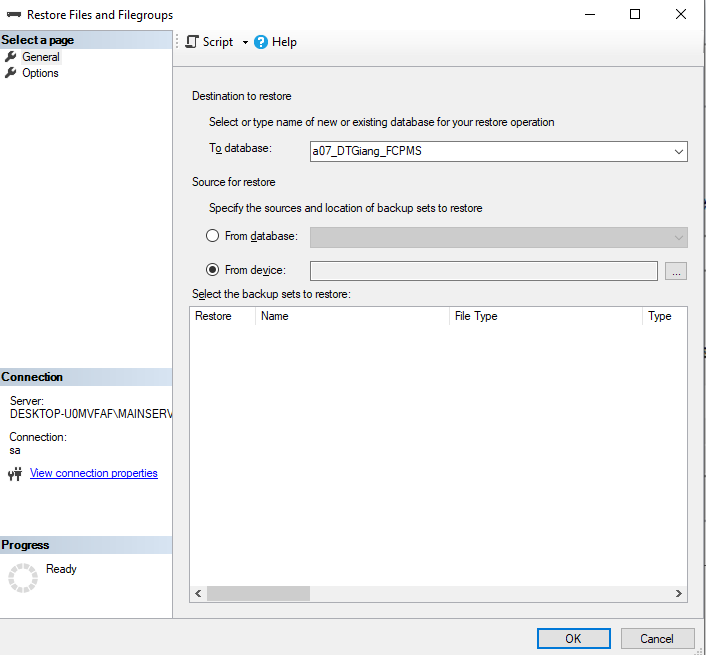
. Backup có nén DB, kích thước file copy (.bak or .tm) nhỏ = thực tế thường dùng

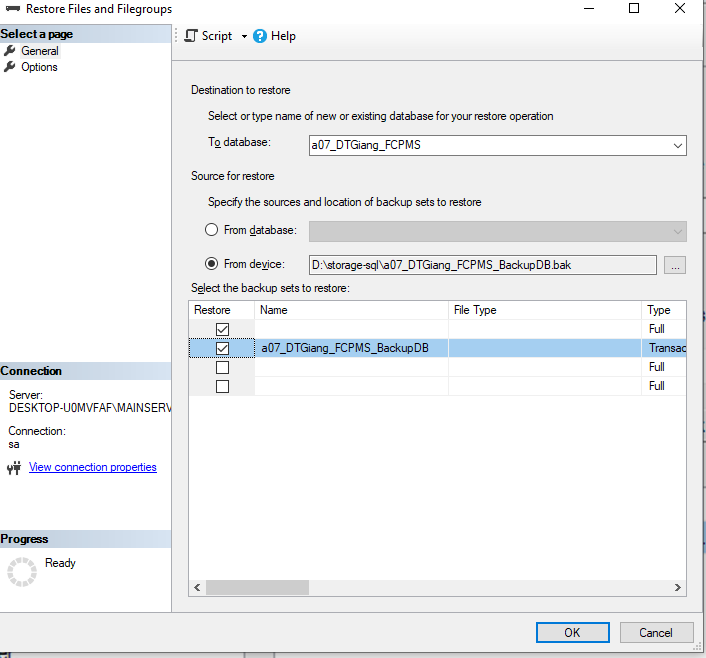
. **Nghiêm ngặt trong vấn đến Version, Authenticate, ServerName,..**



 ====







* **Detach & Attach**

. Giữ nguyên DB từ Server nguồn (.mdf & .ldf) mang sang Attach vào Server đích

. Sau khi Detach DB từ Server source (nguồn) trên server sẽ KHÔNG còn thấy DB đó nữa [.mdf & .ldf vẫn còn, khi cần thì có thể Attach lại] = Các App dùng DB KHÔNG chạy được)

. kích thước file copy (.mdf và .ldf) lớn = thường dùng với DB nhỏ

. Ít mghiêm ngặt trong vấn đến Version, Authenticate, ServerName,..

* **Chú thích**

. Nếu DB đang còn trên Server thì sẽ KHÔNG copy, move, xóa, …. Với các file .mdf và .ldf = muốn thực hiện các thao tác này thì phải Detach trước.

=> khi làm các bài thi [không lưu .mdf & .ldf trong thư mục bài làm = Không copy, nén được => NỘP .bak; OR phải Detach trước rồi nộp .mdf,..]

=> BigData không thực hiện việc này / vì nó rất lớn trên Internet.

# CÀI ĐẶT CÁC THỦ TỤC KHAI THÁC SỬ DỤNG CSDL “HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT”TRÊN DBMS MS. SQL SERVER.

## Các lệnh thao tác xử lý trên CSDL trong MS. SQl Server (DML SQL)

DML Chia làm 2 nhóm:

. Thao tác xử lý: Thêm (Insert), Xóa (Delete), Sửa (Update)

. Vấn tin (Query): Select…

### Thêm mẫu tin (bảng tin: Record) mới vào một Table trong một CSDL: Insert…

**\* Thứ tự nhập dữ liệu (Insert):**

+ DMCSBH, DMNSX, DMNV, DMKH

+ DMMH, DMHD

+ DMBILL

+ TONHANG, CT\_HD, CT\_BILL

**31\_DTGiang\_FCPMS\_InsertDB.sql**

-- Dữ liệu mẫu cho bảng DMCM (Danh mục chuyên môn)

INSERT INTO [dbo].[DMCM] ([mscm], [tencm], [hs], [ghichu])

VALUES

(N'CM001', N'Công nghệ thông tin', 1.0, N'Chuyên môn về CNTT'),

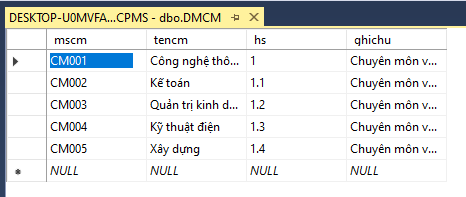
(N'CM002', N'Kế toán', 1.1, N'Chuyên môn về Kế toán'),

(N'CM003', N'Quản trị kinh doanh', 1.2, N'Chuyên môn về Quản trị'),

(N'CM004', N'Kỹ thuật điện', 1.3, N'Chuyên môn về Kỹ thuật điện'),

(N'CM005', N'Xây dựng', 1.4, N'Chuyên môn về Xây dựng')

GO



-- Dữ liệu mẫu cho bảng DMTD (Danh mục trình độ)

INSERT INTO [dbo].[DMTD] ([mstd], [tentd], [hs], [ghichu])

VALUES

(N'TD001', N'Cao đẳng', 1.0, N'Ghi chú Cao đẳng'),

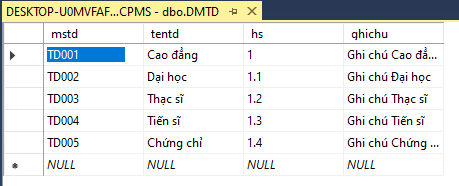
(N'TD002', N'Dại học', 1.1, N'Ghi chú Đại học'),

(N'TD003', N'Thạc sĩ', 1.2, N'Ghi chú Thạc sĩ'),

(N'TD004', N'Tiến sĩ', 1.3, N'Ghi chú Tiến sĩ'),

(N'TD005', N'Chứng chỉ', 1.4, N'Ghi chú Chứng chỉ')

GO



-- Dữ liệu mẫu cho bảng DMCV (Danh mục chức vụ)

INSERT INTO [dbo].[DMCV] ([mscv], [tencv], [hs], [ghichu])

VALUES

(N'CV001', N'Giám đốc', 1.0, N'Chức vụ Giám đốc'),

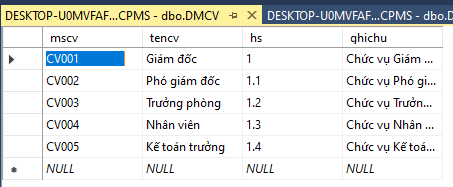
(N'CV002', N'Phó giám đốc', 1.1, N'Chức vụ Phó giám đốc'),

(N'CV003', N'Trưởng phòng', 1.2, N'Chức vụ Trưởng phòng'),

(N'CV004', N'Nhân viên', 1.3, N'Chức vụ Nhân viên'),

(N'CV005', N'Kế toán trưởng', 1.4, N'Chức vụ Kế toán trưởng')

GO



-- Dữ liệu mẫu cho bảng DMDV (Danh mục đơn vị)

INSERT INTO [dbo].[DMDV] ([msdv], [tendv], [hs], [ghichu])

VALUES

(N'DV001', N'Phòng kinh doanh', 1.0, N'Đơn vị Phòng kinh doanh'),

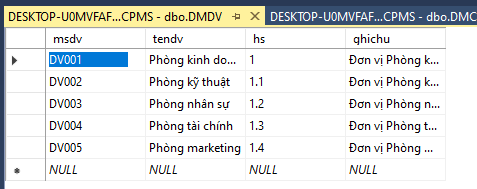
(N'DV002', N'Phòng kỹ thuật', 1.1, N'Đơn vị Phòng kỹ thuật'),

(N'DV003', N'Phòng nhân sự', 1.2, N'Đơn vị Phòng nhân sự'),

(N'DV004', N'Phòng tài chính', 1.3, N'Đơn vị Phòng tài chính'),

(N'DV005', N'Phòng marketing', 1.4, N'Đơn vị Phòng marketing')

GO



-- Dữ liệu mẫu cho bảng DMKCONG (Danh mục khoản công)

INSERT INTO [dbo].[DMKCONG] ([mskcong], [tenkcong], [sotien], [ghichu])

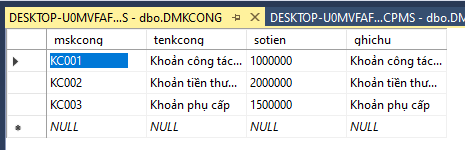
VALUES

(N'KC001', N'Khoản công tác phí', 1000000, N'Khoản công tác phí'),

(N'KC002', N'Khoản tiền thưởng', 2000000, N'Khoản tiền thưởng'),

(N'KC003', N'Khoản phụ cấp', 1500000, N'Khoản phụ cấp')

GO



-- Dữ liệu mẫu cho bảng DMKTRU (Danh mục khoản trừ)

INSERT INTO [dbo].[DMKTRU] ([msktru], [tenktru], [sotien], [ghichu])

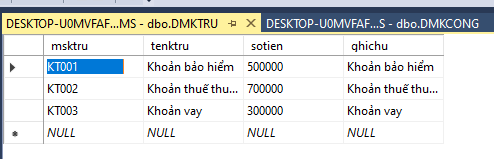
VALUES

(N'KT001', N'Khoản bảo hiểm', 500000, N'Khoản bảo hiểm'),

(N'KT002', N'Khoản thuế thu nhập', 700000, N'Khoản thuế thu nhập'),

(N'KT003', N'Khoản vay', 300000, N'Khoản vay')

GO



-- Dữ liệu mẫu cho bảng KYLUONG (Kỳ lương)

INSERT INTO [dbo].[KYLUONG] ([msky], [nam], [thang], [ghichu])

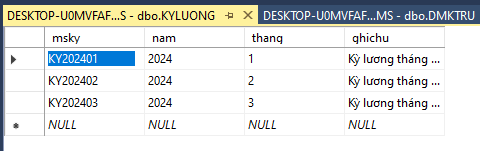
VALUES

(N'KY202401', 2024, 1, N'Kỳ lương tháng 1 năm 2024'),

(N'KY202402', 2024, 2, N'Kỳ lương tháng 2 năm 2024'),

(N'KY202403', 2024, 3, N'Kỳ lương tháng 3 năm 2024')

GO



-- Dữ liệu mẫu cho bảng DMNS (Danh mục nhân sự)

INSERT INTO [dbo].[DMNS] ([msns], [tenns], [nu], [ngaysinh], [ngaytuyen], [sodt], [hinh], [mscm], [mstd], [mscv], [msdv])

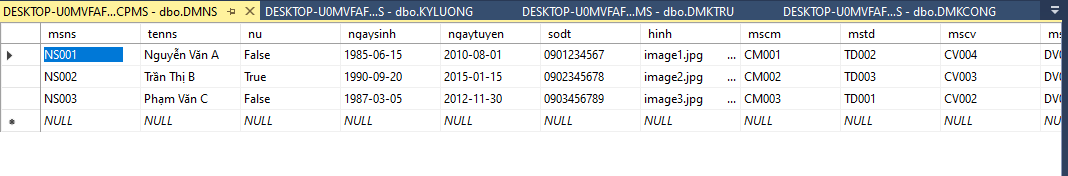
VALUES

(N'NS001', N'Nguyễn Văn A', 0, '1985-06-15', '2010-08-01', '0901234567', N'image1.jpg', N'CM001', N'TD002', N'CV004', N'DV001'),

(N'NS002', N'Trần Thị B', 1, '1990-09-20', '2015-01-15', '0902345678', N'image2.jpg', N'CM002', N'TD003', N'CV003', N'DV002'),

(N'NS003', N'Phạm Văn C', 0, '1987-03-05', '2012-11-30', '0903456789', N'image3.jpg', N'CM003', N'TD001', N'CV002', N'DV003')

GO



-- Dữ liệu mẫu cho bảng BANGCHAMCONG (Bảng chấm công)

INSERT INTO [dbo].[BANGCHAMCONG] ([msns], [msky], [songaychuan], [songaylv], [ghichu])

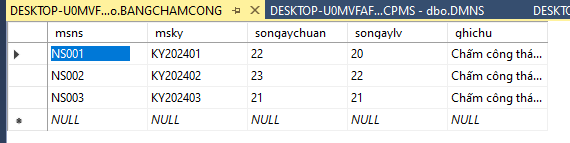
VALUES

(N'NS001', N'KY202401', 22, 20, N'Chấm công tháng 1 năm 2024'),

(N'NS002', N'KY202402', 23, 22, N'Chấm công tháng 2 năm 2024'),

(N'NS003', N'KY202403', 21, 21, N'Chấm công tháng 3 năm 2024')

GO



-- Dữ liệu mẫu cho bảng PHIEULUONG (Phiếu lương)

INSERT INTO [dbo].[PHIEULUONG] ([msphieu], [msns], [msky], [mucluong], [tonghs], [ghichu])

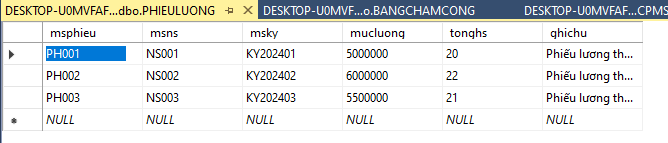
VALUES

(N'PH001', N'NS001', N'KY202401', 5000000, 20, N'Phiếu lương tháng 1 năm 2024'),

(N'PH002', N'NS002', N'KY202402', 6000000, 22, N'Phiếu lương tháng 2 năm 2024'),

(N'PH003', N'NS003', N'KY202403', 5500000, 21, N'Phiếu lương tháng 3 năm 2024')

GO



-- Dữ liệu mẫu cho bảng CTKCONG (Chi tiết khoản công)

INSERT INTO [dbo].[CTKCONG] ([msphieu], [mskcong], [hs])

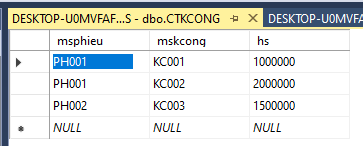
VALUES

(N'PH001', N'KC001', 1000000),

(N'PH001', N'KC002', 2000000),

(N'PH002', N'KC003', 1500000)

GO



-- Dữ liệu mẫu cho bảng CTKTRU (Chi tiết khoản trừ)

INSERT INTO [dbo].[CTKTRU] ([msphieu], [msktru], [hs])

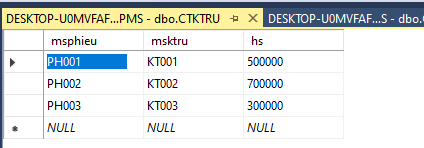
VALUES

(N'PH001', N'KT001', 500000),

(N'PH002', N'KT002', 700000),

(N'PH003', N'KT003', 300000)

GO



### Xóa mẫu tin (Record) trong một Table của CSDL: Delete..

* Xóa tất cả các nhân sự có “Mã đơn vị” = “DV001”

USE [a07\_DTGiang\_FCPMS]

GO

**DELETE FROM DMNS WHERE msdv = ‘DV001’**

GO

* Xóa nhân sự có mã số **'NS001'**

USE [a07\_DTGiang\_FCPMS]

GO

**DELETE FROM DMNS WHERE msns = ‘NS001’**

GO

### Sửa chữa thông tin của mẫu tin (Record) trong một Table của CSDL: Update..

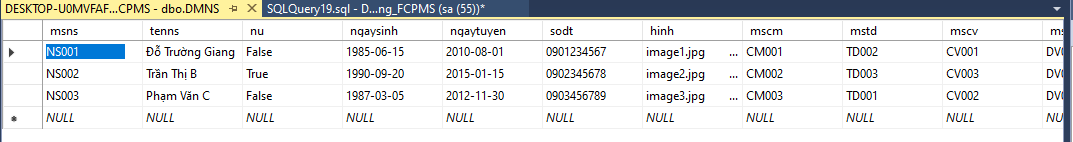
* Sửa chữa thông tin nhân sự có mã số **'NS001'** :   
  tên mới của nhân sự **'Đỗ Trường Giang'** và mscv mới là **'CV001'**

USE [a07\_DTGiang\_FCPMS]

GO

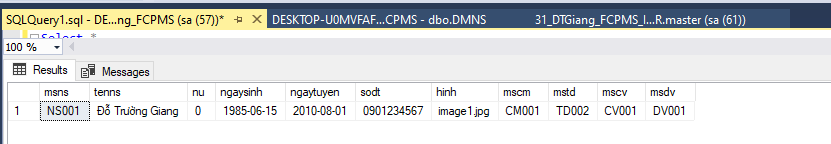
**UPDATE DMNS SET tenns = N'Đỗ Trường Giang', mscv = 'CV001' WHERE msns = 'NS001'**

GO



## Vấn tin SQL trên MS. SQL Server trên CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT(DML)

**Câu hỏi 1:** Cho biết danh sách nhân sự có : tenns = ‘Đỗ Trường Giang’, nu = false, mscm = ‘CM001’ ?



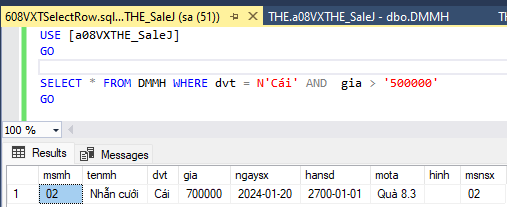
USE [a07\_DTGiang\_FCPMS]

GO

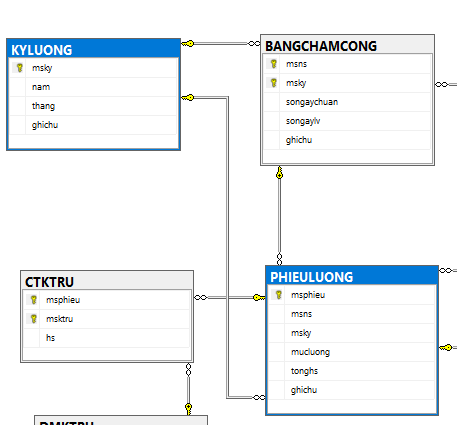
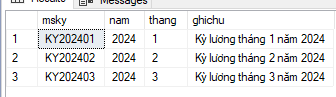
**Select \***

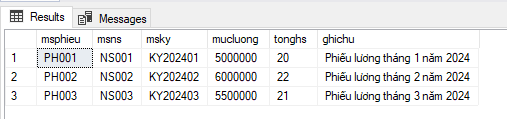
**From dbo.DMNS**

**where tenns = N'Đỗ Trường Giang' and nu = 'false' and mscm = 'CM001'**



**Câu hỏi 2:** Cho biết các PHIẾU LƯƠNG đã được lập trong KỲ LƯƠNG thứ 1 của năm 2024 (msky = KY202401) ?

**** ****

****

USE [a07\_DTGiang\_FCPMS]

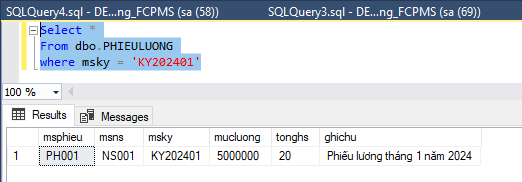
GO

**Select \***

**From dbo.PHIEULUONG**

**where msky = 'KY202401'**

GO



## Cài đặt các View trên MS. SQL Server trên CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT (DML)

**Bài 1** : Lập View dùng phép kết thông thường (không tinh toán)

**+ View : Lập danh sách các nhân sự CHƯA ĐƯỢC chấm công cho nhân viên kế toán dễ dàng kiểm soát**

SELECT dbo.DMNS.tenns, dbo.DMNS.ngaysinh, dbo.DMNS.sodt, dbo.DMCV.tencv AS [Chức vụ], dbo.DMCM.tencm AS [Chuyên môn], dbo.DMTD.tentd AS [Trình độ], dbo.DMDV.tendv AS [Đơn vị]

FROM dbo.DMNS INNER JOIN

dbo.DMCM ON dbo.DMNS.mscm = dbo.DMCM.mscm INNER JOIN

dbo.DMCV ON dbo.DMNS.mscv = dbo.DMCV.mscv INNER JOIN

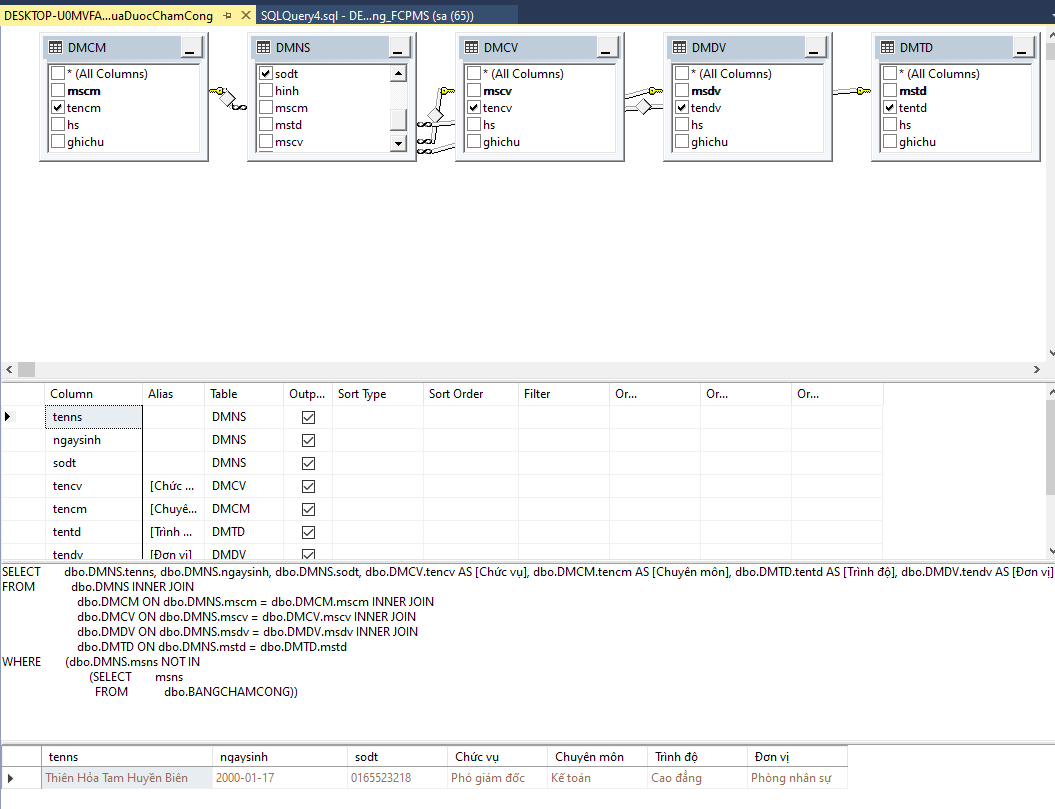
dbo.DMDV ON dbo.DMNS.msdv = dbo.DMDV.msdv INNER JOIN

dbo.DMTD ON dbo.DMNS.mstd = dbo.DMTD.mstd

WHERE (dbo.DMNS.msns NOT IN

(SELECT msns

FROM dbo.BANGCHAMCONG))

****

**Bài 2 : Lập View dùng phép kết có tính toán**

**+ View : Lập danh sách thành tiền các khoảng cộng/trừ của lương nhân viên :**

Mã số phiếu, ngày lập, tên nhân sự, giới tính (True = Nữ), ngày sinh, tên khoảng trừ, hệ số KTRU, số tiền KTRU, tổng KTRU (hệ số KTRU \* số tiền KTRU), tên khoảng cộng, hệ số KCONG, số tiền KCONG, tổng KCONG (hệ số KCONG \* số tiền KCONG)

====

SELECT dbo.PHIEULUONG.msphieu, CAST(dbo.KYLUONG.thang AS nvarchar) + '/ ' + CAST(dbo.KYLUONG.nam AS nvarchar) AS [Ngày lập], dbo.DMNS.tenns AS [Tên nhân sự], dbo.DMNS.nu AS [Giới tính], dbo.DMNS.ngaysinh,

dbo.DMKTRU.tenktru AS [Tên khoảng trừ], dbo.CTKTRU.hs AS [Hệ số KTRU], dbo.DMKTRU.sotien AS [Số tiền KTRU], dbo.CTKTRU.hs \* dbo.DMKTRU.sotien AS [Tổng KTRU], dbo.DMKCONG.tenkcong AS [Tên khoảng cộng],

dbo.CTKCONG.hs AS [Hệ số KCONG], dbo.DMKCONG.sotien AS [Số tiền KCONG], dbo.CTKCONG.hs \* dbo.DMKCONG.sotien AS [Tổng KCONG]

FROM dbo.PHIEULUONG INNER JOIN

dbo.KYLUONG ON dbo.PHIEULUONG.msky = dbo.KYLUONG.msky INNER JOIN

dbo.DMNS ON dbo.PHIEULUONG.msns = dbo.DMNS.msns INNER JOIN

dbo.CTKTRU ON dbo.PHIEULUONG.msphieu = dbo.CTKTRU.msphieu INNER JOIN

dbo.CTKCONG ON dbo.PHIEULUONG.msphieu = dbo.CTKCONG.msphieu INNER JOIN

dbo.DMKCONG ON dbo.CTKCONG.mskcong = dbo.DMKCONG.mskcong INNER JOIN

dbo.DMKTRU ON dbo.CTKTRU.msktru = dbo.DMKTRU.msktru

**Bài 3: Lập View dùng phép kết có dùng hàm SUM, COUNT, AVERAGE, MAX, MIN, . . .)**

**+ View : Lập danh sách tổng các khoảng cộng/trừ của các phiếu lương của nhân viên:**

Mã phiếu lương, Ngày lập, tên nhân sự, giới tính, ngày sinh, khoảng cộng, khoảng trừ, tổng

. Gôm lại theo : Mã phiếu lương

. Tính tổng số tiền khoảng cộng/ trừ = SUM(khoảng cộng) – SUM(khoảng trừ) theo cùng Mã phiếu lương.

SELECT msphieu, [Ngày lập], [Tên nhân sự], [Giới tính], ngaysinh, [Khoảng Cộng], [Khoảng Trừ], [Khoảng Cộng] - [Khoảng Trừ] AS Tổng

FROM (SELECT msphieu, [Ngày lập], [Tên nhân sự], [Giới tính], ngaysinh, SUM([Tổng KCONG]) AS [Khoảng Cộng], SUM([Tổng KTRU]) AS [Khoảng Trừ]

FROM (SELECT dbo.PHIEULUONG.msphieu, CAST(dbo.KYLUONG.thang AS nvarchar) + '/ ' + CAST(dbo.KYLUONG.nam AS nvarchar) AS [Ngày lập], dbo.DMNS.tenns AS [Tên nhân sự], dbo.DMNS.nu AS [Giới tính],

dbo.DMNS.ngaysinh, dbo.DMKTRU.tenktru AS [Tên khoảng trừ], dbo.CTKTRU.hs AS [Hệ số KTRU], dbo.DMKTRU.sotien AS [Số tiền KTRU], dbo.CTKTRU.hs \* dbo.DMKTRU.sotien AS [Tổng KTRU],

dbo.DMKCONG.tenkcong AS [Tên khoảng cộng], dbo.CTKCONG.hs AS [Hệ số KCONG], dbo.DMKCONG.sotien AS [Số tiền KCONG], dbo.CTKCONG.hs \* dbo.DMKCONG.sotien AS [Tổng KCONG]

FROM dbo.PHIEULUONG INNER JOIN

dbo.KYLUONG ON dbo.PHIEULUONG.msky = dbo.KYLUONG.msky INNER JOIN

dbo.DMNS ON dbo.PHIEULUONG.msns = dbo.DMNS.msns INNER JOIN

dbo.CTKTRU ON dbo.PHIEULUONG.msphieu = dbo.CTKTRU.msphieu INNER JOIN

dbo.CTKCONG ON dbo.PHIEULUONG.msphieu = dbo.CTKCONG.msphieu INNER JOIN

dbo.DMKCONG ON dbo.CTKCONG.mskcong = dbo.DMKCONG.mskcong INNER JOIN

dbo.DMKTRU ON dbo.CTKTRU.msktru = dbo.DMKTRU.msktru) AS derivedtbl\_1

GROUP BY msphieu, [Tên nhân sự], [Ngày lập], ngaysinh, [Giới tính]) AS derivedtbl\_2

## Lập các Store Procedure (SP) và các Trigger cho các Table của DB

**LẬP SP : Kiểm tra Danh sách các NHÂN VIÊN đã chấm công nhưng chưa có phiếu lương**

-- ================================================

-- Template generated from Template Explorer using:

-- Create Procedure (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- This block of comments will not be included in

-- the definition of the procedure.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Name

-- Create date:

-- Description:

-- =============================================

CREATE PROCEDURE Procedure\_NhanVien\_DaChamCong\_ChuaCoPhieuLuong

-- Add the parameters for the stored procedure here

@msns nchar(20)

AS

BEGIN

-- Insert statements for procedure here

SELECT \*

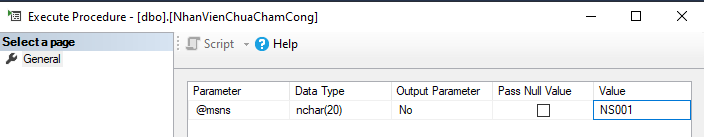
FROM dbo.View\_NhanSu\_DaChamCong\_ChuaCoPhieuLuong

Where msns LIKE @msns

END

GO

**CHẠY SP : Kiểm tra Danh sách các NHÂN VIÊN đã chấm công nhưng chưa có phiếu lương**



USE [a07\_DTGiang\_FCPMS]

GO

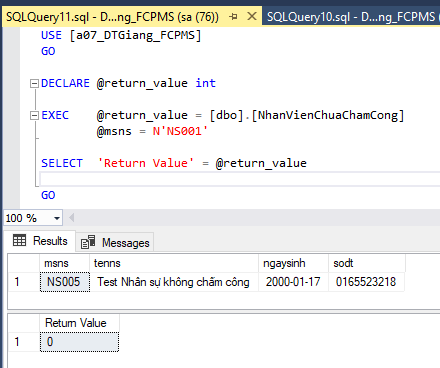
DECLARE @return\_value int

EXEC @return\_value = [dbo].[NhanVienChuaChamCong]

@msns = N'NS001'

SELECT 'Return Value' = @return\_value

GO



**LẬP Trigger :**

* **UPDATE Danh sách NS thuộc mscm/mscv/mstd/msdv vừa được xóa ở bảng DMCM/DMCV/DMTD/DMDV thành ‘init’ (được tạo mặc định ở bảng DMCM/DMCV/DMTD/DMDV)**

**\*Chú thích: Dưới đây chỉ là query tạo trigger cho một table DMCM, các table còn lại có logic tương tự, chỉ thay đổi tên**

USE [a07\_DTGiang\_FCPMS]

GO

/\*\*\*\*\*\* Object: Trigger [dbo].[Trigger\_Delete] Script Date: 9/13/2024 4:57:00 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Author,,Name>

-- Create date: <Create Date,,>

-- Description: <Description,,>

-- =============================================

ALTER TRIGGER [dbo].[Trigger\_Delete]

ON [dbo].[DMCM]

INSTEAD OF DELETE

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Lấy mscm sau khi delete

DECLARE @msinit nchar(20) = 'init'

-- Kiểm tra xem mã init có trong data không

IF NOT EXISTS (SELECT \* FROM DMCM WHERE mscm = @msinit)

BEGIN

INSERT INTO DMCM VALUES('init', N'Không được xóa mã này', 1, N'Không được xóa mã này')

END

-- Kiểm tra xem người dùng có đang xóa mã init không

IF EXISTS (SELECT \* FROM deleted WHERE mscm = @msinit)

BEGIN

PRINT 'Không được xóa mã: ' + @msinit;

RETURN;

END

-- CASE nếu người dùng xóa không phải là mã init

DECLARE @mscm nchar(20)

DECLARE @tencm nvarchar(50)

-- lấy mã người dùng định xóa

select @mscm = d.mscm, @tencm = d.tencm

from deleted d

-- Thay thế tất cả các mscm ở bảng DMNS mà người dùng định xóa thành 'init'

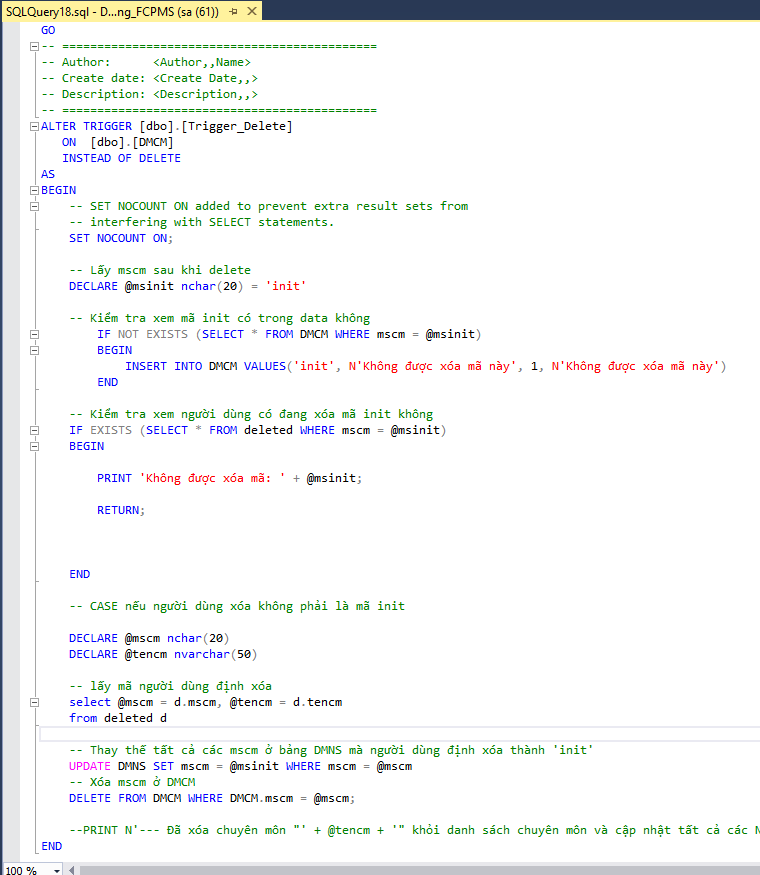
UPDATE DMNS SET mscm = @msinit WHERE mscm = @mscm

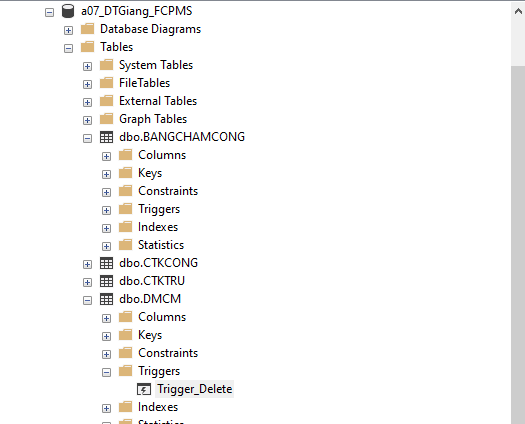
-- Xóa mscm ở DMCM

DELETE FROM DMCM WHERE DMCM.mscm = @mscm;

--PRINT N'--- Đã xóa chuyên môn "' + @tencm + '" khỏi danh sách chuyên môn và cập nhật tất cả các Nhân Sự có mã chuyên môn này thành ' + @msinit + '! ---'

END





* **Hiển thị danh sách NHÂN SỰ vừa được INSERT/UPDATE:**

-- ================================================

-- Template generated from Template Explorer using:

-- Create Trigger (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- See additional Create Trigger templates for more

-- examples of different Trigger statements.

--

-- This block of comments will not be included in

-- the definition of the function.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Author,,Name>

-- Create date: <Create Date,,>

-- Description: <Description,,>

-- =============================================

CREATE TRIGGER Trigger\_Insert\_Update

ON [dbo].[DMNS]

AFTER INSERT,UPDATE

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @msns nchar(20)

select @msns = I.msns

from inserted I

print '--- Bạn vừa thêm nhân sự có mã: ' + @msns + ' vào data ---'

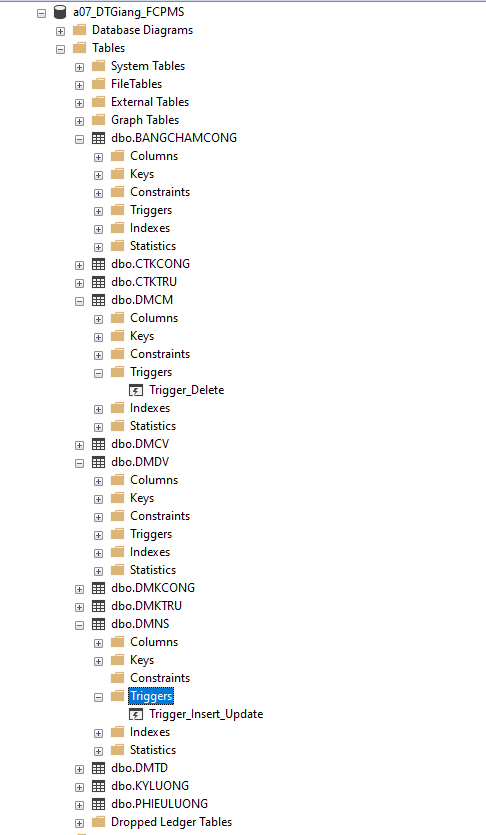
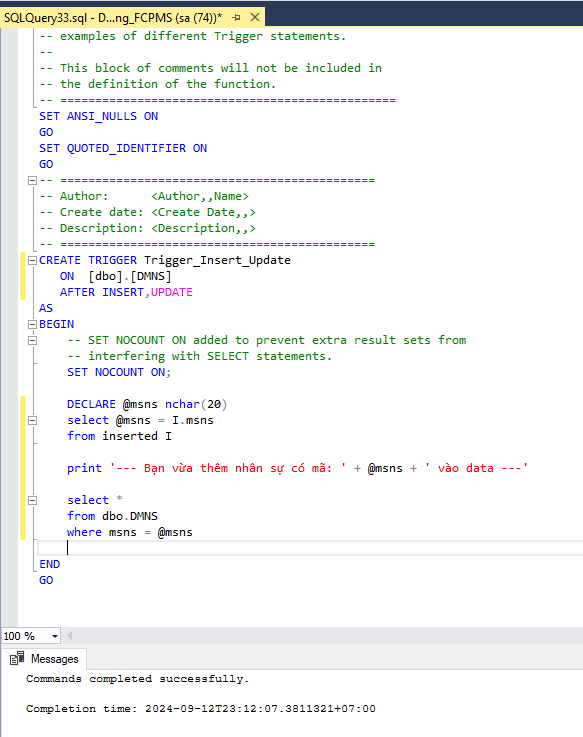
select \*

from dbo.DMNS

where msns = @msns

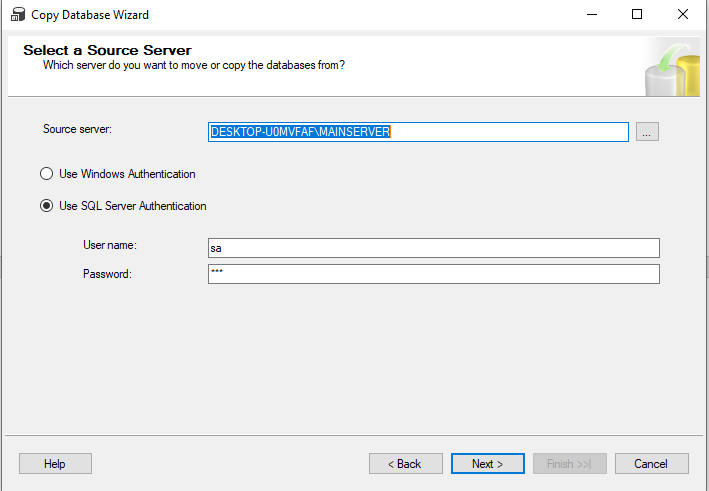
END

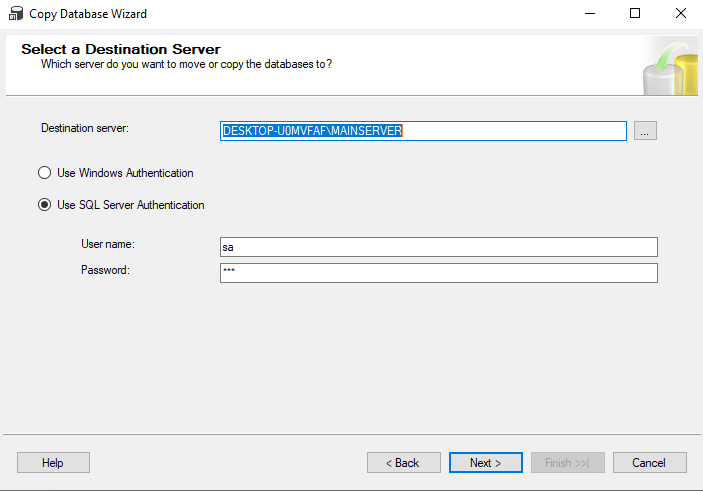
GO

****

## Lập bảng sao CSDL trên hệ quản trị MS. SQL Server đối với CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT

Click phải chuột vào [a07\_DTGiang\_FCPMT] -> Tasks -> Copy Database:

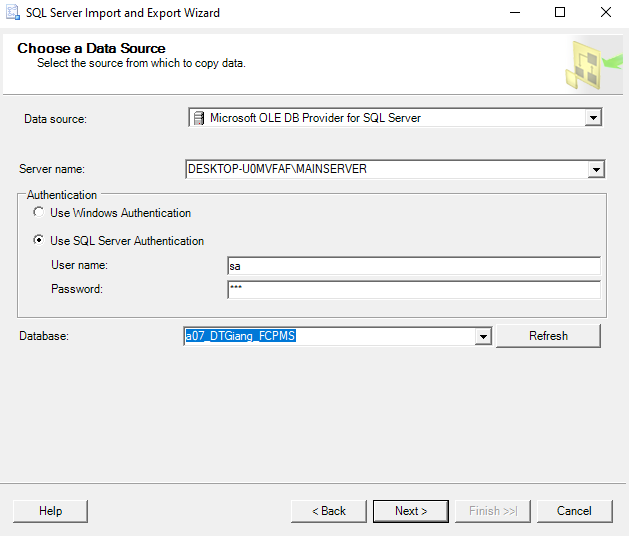


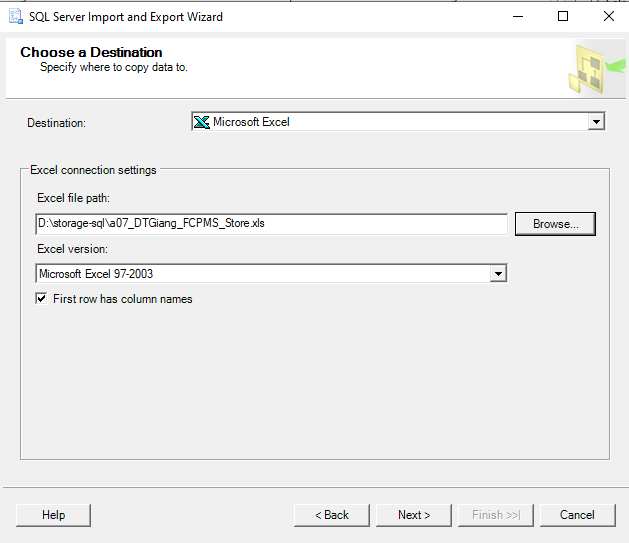


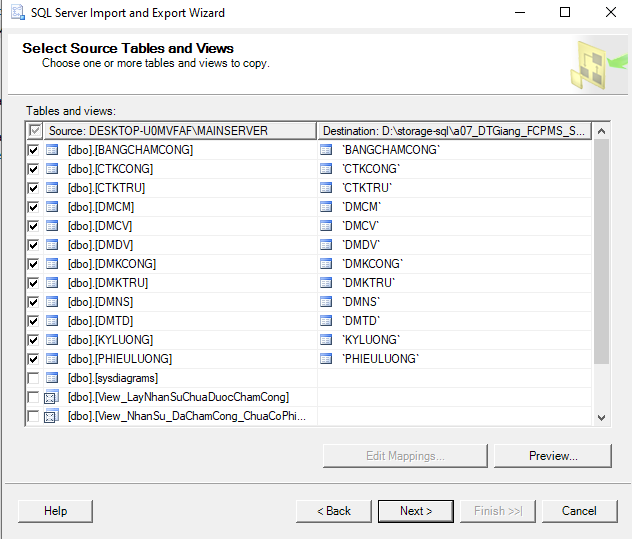
## Kết xuất (Export) và Nạp (Import) DL trên hệ quản trị MS. SQL Server đối với CSDL HỆ THỐNG TIỀN LƯƠNG NHÂN VIÊN CỦA CÔNG TY FE CREDIT

**Kết xuất (Export):**

* Click phải chuột [a07\_DTGiang\_FCPMS] -> Tasks -> Export Data.







**Nạp (Import):**

* Click phải chuột [a07\_DTGiang\_FCPMS] -> Tasks -> Import Data

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỒ ÁN HỌC PHẦN

## Kết luận

### Những kết quả đạt được

Sau quá trình nghiên cứu và thực hiện đề tài "Hệ thống quản lý lương nhân viên của công ty Fe Credit", em đã đạt được những mục tiêu đề ra và hoàn thiện hệ thống cơ sở dữ liệu phù hợp với yêu cầu quản lý lương trong doanh nghiệp. Đề tài không chỉ giúp em củng cố và mở rộng kiến thức về thiết kế cơ sở dữ liệu mà còn trang bị cho em những kỹ năng phân tích, lập trình và xử lý dữ liệu thực tế.

Hệ thống quản lý lương được xây dựng đã đáp ứng được các chức năng cơ bản như quản lý thông tin nhân viên, tính toán lương dựa trên các yếu tố như số giờ làm, các khoản thưởng và phụ cấp, cũng như quản lý các khoản khấu trừ. Việc áp dụng những nguyên tắc chuẩn hóa trong thiết kế cơ sở dữ liệu đã giúp hệ thống có khả năng mở rộng và duy trì hiệu suất hoạt động cao khi được triển khai trong môi trường doanh nghiệp thực tế.

### Hạn chế

Tuy nhiên, bên cạnh những thành tựu đạt được, hệ thống vẫn còn những hạn chế cần khắc phục và cải tiến trong tương lai. Việc tích hợp hệ thống với các phần mềm quản lý nhân sự hoặc kế toán khác là một trong những hướng phát triển tiếp theo. Đồng thời, em cũng nhận thấy việc bảo mật thông tin và tối ưu hóa quy trình xử lý dữ liệu là các vấn đề cần được quan tâm hơn nữa để đảm bảo hệ thống hoạt động ổn định và an toàn trong môi trường doanh nghiệp.

## Hướng phát triển

Mặc dù hệ thống quản lý lương nhân viên của công ty Fe Credit đã đạt được những mục tiêu ban đầu, tuy nhiên, hệ thống vẫn tồn tại một số hạn chế cần khắc phục và có nhiều tiềm năng để mở rộng. Để hệ thống hoàn thiện hơn và đáp ứng tốt hơn các yêu cầu thực tế, em xin đề xuất các hướng phát triển sau:

### Hướng khắc phục các hạn chế

* **Cải thiện tính năng bảo mật**: Hiện tại, hệ thống chỉ dừng lại ở mức bảo mật cơ bản, việc quản lý và phân quyền cho người dùng vẫn còn đơn giản. Trong tương lai, hệ thống cần được cải thiện về mặt bảo mật, đặc biệt là trong việc bảo vệ thông tin cá nhân và thông tin lương của nhân viên. Các biện pháp mã hóa dữ liệu, xác thực người dùng qua nhiều yếu tố (2FA), và quản lý quyền truy cập phức tạp hơn nên được triển khai để nâng cao độ an toàn cho hệ thống.
* **Tối ưu hóa hiệu suất hệ thống**: Khi quy mô dữ liệu tăng lên, hệ thống có thể gặp phải vấn đề về hiệu suất trong việc xử lý và truy vấn dữ liệu. Việc sử dụng các chiến lược tối ưu hóa truy vấn, đánh chỉ số cho bảng dữ liệu và nâng cấp cấu trúc cơ sở dữ liệu sẽ giúp hệ thống hoạt động mượt mà hơn khi áp dụng trong doanh nghiệp lớn.
* **Khả năng tích hợp với hệ thống khác**: Hiện tại, hệ thống chưa hỗ trợ việc tích hợp với các hệ thống quản lý nhân sự (HRM) hoặc quản lý tài chính (ERP). Đây là một hạn chế cần được khắc phục bằng cách thiết kế API mở hoặc sử dụng các chuẩn giao tiếp chung để có thể tích hợp với các hệ thống khác trong doanh nghiệp, giúp đồng bộ dữ liệu và giảm thiểu công việc thủ công.

### Hướng mở rộng ĐỒ ÁN HỌC PHẦN

 **Phát triển tính năng báo cáo chi tiết và linh hoạt**: Hệ thống hiện tại đã có các báo cáo cơ bản về lương của nhân viên, nhưng để nâng cao tính linh hoạt, cần phát triển thêm các tính năng cho phép người dùng tùy chỉnh các tiêu chí báo cáo theo yêu cầu. Ví dụ, báo cáo phân tích xu hướng lương theo thời gian, so sánh lương giữa các phòng ban, và theo dõi chi phí lương hàng tháng.

 **Áp dụng trí tuệ nhân tạo (AI) và học máy (Machine Learning)**: Một hướng mở rộng mới mẻ là tích hợp AI để dự đoán các xu hướng trong việc trả lương, hoặc phân tích hiệu quả công việc của nhân viên để từ đó đưa ra các quyết định về lương thưởng một cách chính xác hơn. Hệ thống có thể học từ dữ liệu lịch sử để dự đoán chi phí lương cho các tháng tiếp theo hoặc dự báo về các biến động chi phí.

 **Xây dựng hệ thống trên nền tảng web và di động**: Để tăng tính tiện lợi và mở rộng khả năng truy cập, hệ thống nên được phát triển thêm phiên bản web và ứng dụng di động. Điều này giúp quản lý có thể truy cập và kiểm tra dữ liệu lương nhân viên mọi lúc, mọi nơi, đồng thời nâng cao tính tiện dụng cho người dùng.

 **Phát triển hệ thống chấm công tích hợp**: Hướng mở rộng khác là phát triển thêm tính năng chấm công tự động, liên kết với hệ thống quản lý lương. Việc tích hợp này giúp hệ thống tự động tính toán lương dựa trên số giờ làm việc thực tế của nhân viên, đồng thời giảm thiểu sai sót trong quá trình nhập liệu và xử lý lương thủ công.

 **Mở rộng quản lý các khoản phúc lợi và bảo hiểm**: Ngoài việc quản lý lương cơ bản, hệ thống có thể được mở rộng để quản lý các khoản phúc lợi, bảo hiểm xã hội, bảo hiểm y tế và các quyền lợi khác mà nhân viên được hưởng. Điều này giúp hệ thống trở nên toàn diện hơn, cung cấp cho doanh nghiệp một công cụ quản lý tài chính nhân sự đầy đủ và tiện ích.

# DANH MỤC TÀI LIỆU THAM KHẢO

1. Võ Xuân Thể (2022), *Tài liệu học tập Hệ quản trị cơ sở dữ liệu*, Trường Đại học Sư phạm kỹ thuật Tp.HCM, Tài liệu lưu hành nội bộ được cung cấp theo từng buổi học
2. Võ Xuân Thể (2022), *Bài tập thực hành Hệ quản trị cơ sở dữ liệu*, Trường Đại học Sư phạm kỹ thuật Tp.HCM, Tài liệu lưu hành nội bộ được cung cấp theo từng buổi học
3. https://fecredit.com.vn/ : Trang thông tin FE Credit tham khảo thực hiện đề tài này.

# CÁC PHỤ LỤC

Giới thiệu trang chủ FE Credit



1. FeCredit: mượn tên thương hiệu, chỉ phục vụ học tập || LGB: thương hiệu giả định, phục vụ học tập [↑](#footnote-ref-1)
2. Việc này thực hiện theo “CSDL Phân tán: Distribution DB”, trong phạm vị Đề tài này chưa thực hiện. [↑](#footnote-ref-2)
3. Việc này thực hiện theo “CSDL Phân tán: Distribution DB”, trong phạm vị Đề tài này chưa thực hiện. [↑](#footnote-ref-3)