****

TRƯỜNG ĐẠI HỌC NGOẠI NGỮ - TIN HỌC THÀNH PHỒ HỒ CHÍ MINH

**KHOA CÔNG NGHỆ THÔNG TIN**

🙡🕮🙣

**BÁO CÁO KẾT THÚC HỌC PHẦN**

**CƠ SỞ DỮ LIỆU NÂNG CAO**

**PHÂN TÍCH, THIẾT KẾ VÀ CÀI ĐẶT CSDL QUẢN LÝ ĐIỂM DANH SINH VIÊN**

Giảng viên hướng dẫn: **ThS. Huỳnh Thành Lộc**

Sinh viên thực hiện:

1. Nguyễn Trần Thanh Nhã 22DH112486
2. Nguyễn Thùy Ngân 22DH112324
3. Trần Bảo Ngọc 22DH112413

***Thành phố Hồ Chí Minh, tháng 7 năm 2024***

**This page intentionally left blank**

*(Mặt sau của trang BÌA CHÍNH, khi in để trống trang này)*

*(BÌA CHÍNH in trên GIẤY CỨNG MÀU XANH DƯƠNG)*

**A yellow oval with red text

Description automatically generated**

TRƯỜNG ĐẠI HỌC NGOẠI NGỮ - TIN HỌC THÀNH PHỒ HỒ CHÍ MINH

**KHOA CÔNG NGHỆ THÔNG TIN**

🙡🕮🙣

**BÁO CÁO KẾT THÚC HỌC PHẦN**

**CƠ SỞ DỮ LIỆU NÂNG CAO**

**PHÂN TÍCH, THIẾT KẾ VÀ CÀI ĐẶT CSDL QUẢN LÝ ĐIỂM DANH SINH VIÊN**

Mã lớp học phần: 233122202404

Năm học: **2023 – 2024**

Học kỳ: **3**

Sinh viên thực hiện:

1. Nguyễn Trần Thanh Nhã 22DH112486
2. Nguyễn Thùy Ngân 22DH112324
3. Trần Bảo Ngọc 22DH112413

***Thành phố Hồ Chí Minh, tháng 7 năm 2024***

**This page intentionally left blank**

*(Mặt sau của trang BÌA PHỤ, khi in để trống trang này)*

*(BÌA PHỤ in trên GIẤY TRẮNG như các trang nội dung)*

**PHIẾU CHẤM ĐIỂM ĐỒ ÁN CUỐI KỲ  
MÔN CƠ SỞ DỮ LIỆU NÂNG CAO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MSSV:** | | | MSSV1 | MSSV2 | MSSV3 |
| **Họ tên SV:** | | | Họ tên SV1 | Họ tên SV2 | Họ tên SV3 |
| **CLO** | **Nội dung đánh giá** | **Thang điểm** | **Điểm** | **Điểm** | **Điểm** |
| CLO1 | Giải thích và vận dụng được các giai đoạn thiết kế dữ liệu trong hệ thống thông tin. | 1,0 |  |  |  |
| CLO2 | Hiểu rõ phụ thuộc hàm và các thuật toán liên quan đến phụ thuộc hàm để chuẩn hóa được CSDL quan hệ ở dạng chuẩn 3NF hoặc Boyce Codd. | 1,0 |  |  |  |
| CLO3 | Áp dụng ngôn ngữ lập trình CSDL, viết thành thạo các xử lý tại server bằng Store procedure và các loại ràng buộc bằng trigger. | 1,0 |  |  |  |
| CLO4 | Có khả năng nghiên cứu tìm hiểu và cập nhật các công nghệ trong tổ chức dữ liệu bán cấu trúc và phi cấu trúc. | 1,0 |  |  |  |
| CLO5 | Có năng lực giao tiếp và trình bày các giải pháp trong thiết kế, tổ chức và khai thác các loại dữ liệu. | 1,0 |  |  |  |
| CLO6 | Tích cực tham gia các hoạt động nhóm:  *- Khối lượng công việc thực hiện*  *- Tương tác với các thành viên khác*  *- Mức độ hoàn thành nội dung được phân công* | 2,0 |  |  |  |
| CLO7 | Chủ động, chuyên cần và nghiêm túc trong học tập và nghiên cứu:  *- Trình bày quyển báo cáo theo yêu cầu*  *- Hoàn chỉnh nội dung các phần* | 3,0 |  |  |  |
| **Tổng điểm:** | | **10,0** |  |  |  |
| **Bằng chữ:** | |  |  |  |  |

|  |  |
| --- | --- |
| **Họ tên và chữ ký của Giám khảo 1** | **Họ tên và chữ ký của Giám khảo 2** |
| **Huỳnh Thành Lộc** | **Phạm Minh Dũng** |

**This page intentionally left blank**

*(Mặt sau của phiếu điểm, khi in để trống trang này*

MỤC LỤC

[DANH MỤC HÌNH i](#_Toc172372124)

[DANH MỤC BẢNG ii](#_Toc172372125)

[CHƯƠNG 1. MÔ TẢ ĐỀ TÀI 1](#_Toc172372126)

[1.1. Giới thiệu tổng quan đề tài 1](#_Toc172372127)

[1.1.1. Thông tin chung 1](#_Toc172372128)

[1.1.2. Cơ cấu tổ chức 3](#_Toc172372129)

[1.2. Yêu cầu lưu trữ 3](#_Toc172372130)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 5](#_Toc172372131)

[2.1. Giới thiệu về các Cơ sở lý thuyết 5](#_Toc172372132)

[2.1.1. Mô hình ERD 5](#_Toc172372133)

[2.1.2. Phụ thuộc hàm 5](#_Toc172372134)

[2.1.3. Các dạng chuẩn 5](#_Toc172372135)

[2.1.4. Bảo toàn thông tin 5](#_Toc172372136)

[2.1.5. Lập trình CSDL 5](#_Toc172372137)

[2.1.6. Ràng buộc toàn vẹn 5](#_Toc172372138)

[2.1.7. NoSQL và NewSQL 5](#_Toc172372139)

[CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU 6](#_Toc172372140)

[3.1. Mô hình thực thể kết hợp 6](#_Toc172372141)

[3.2. Mô tả các thực thể 6](#_Toc172372142)

[3.2.1. Thực thể sinh viên 6](#_Toc172372143)

[3.2.2. Thực thể giảng viên 7](#_Toc172372144)

[3.2.3. Thực thể khoa 7](#_Toc172372145)

[3.2.4. Thực thể môn học 7](#_Toc172372146)

[3.2.5. Thực thể lớp học phần 7](#_Toc172372147)

[3.2.6. Thực thể lớp điểm danh 8](#_Toc172372148)

[3.2.7. Thực thể buổi học 8](#_Toc172372149)

[3.3. Lược đồ cơ sở dữ liệu quan hệ 8](#_Toc172372150)

[3.3.1. Biểu diễn dạng text 8](#_Toc172372151)

[3.3.2. Biểu diễn dạng hình (dùng công cụ ERDPlus) 9](#_Toc172372152)

[3.3.3. Phân tích dạng chuẩn của lược đồ quan hệ 9](#_Toc172372153)

[3.3.4. Lược đồ cơ sở dữ liệu quan hệ đạt Dạng chuẩn 3 (hoặc BCNF) 10](#_Toc172372154)

[3.4. Ràng buộc toàn vẹn 10](#_Toc172372155)

[3.4.1. R1: “…Nội dung RBTV…” 10](#_Toc172372156)

[3.4.2. R2: “…Nội dung RBTV…” 10](#_Toc172372157)

[CHƯƠNG 4. TRIỂN KHAI CƠ SỞ DỮ LIỆU TRÊN Ms SQL SERVER 11](#_Toc172372158)

[4.1. Tạo cơ sở dữ liệu 11](#_Toc172372159)

[4.2. Nhập liệu mẫu 11](#_Toc172372160)

[4.3. Cài đặt các ràng buộc toàn vẹn 11](#_Toc172372161)

[4.3.1. R1: “…Nội dung RBTV…” 11](#_Toc172372162)

[4.3.2. R2: “…Nội dung RBTV…” 11](#_Toc172372163)

[4.4. Cài đặt các store procedure 11](#_Toc172372164)

[4.4.1. SP\_<tên SP> : “…chức năng của SP…” 11](#_Toc172372165)

[4.4.2. SP\_<tên SP> : “…chức năng của SP…” 11](#_Toc172372166)

[4.4.3. 11](#_Toc172372167)

[CHƯƠNG 5. TRIỂN KHAI CƠ SỞ DỮ LIỆU dạng nosql 12](#_Toc172372168)

[5.1. Chuyển đổi cơ sở dữ liệu quan hệ sang NoSQL 12](#_Toc172372169)

[5.2. Các câu lệnh tạo dữ liệu 12](#_Toc172372170)

[5.3. Truy vấn dữ liệu NoSQL 12](#_Toc172372171)

[CHƯƠNG 6. KẾT LUẬN 13](#_Toc172372172)

[6.1. Kết quả đạt được và chưa đạt được 13](#_Toc172372173)

[6.2. Hướng phát triển mở rộng ứng dụng trong tương lai 13](#_Toc172372174)

[TÀI LIỆU THAM KHẢO 14](#_Toc172372175)

[PHỤ LỤC 15](#_Toc172372176)

# DANH MỤC HÌNH

[Hình 1.1. Thiết kế CSDL 1](#_Toc172273134)

# DANH MỤC BẢNG

[Bảng 1.1. Thông tin lưu trữ của nhân viên 1](#_Toc172273124)

[Bảng 3.1. Mô tả thực thể… 3](#_Toc172273125)

[Bảng 3.2. Mô tả thực thể… 3](#_Toc172273126)

# MÔ TẢ ĐỀ TÀI

## Giới thiệu tổng quan đề tài

Đề tài này tập trung vào xây dựng hệ thống điểm danh cho học sinh sinh viên. Việc điểm danh theo cách thủ công rất mất thời gian ở các trường học, trung tâm đào tạo, giáo viên có lượng học sinh sinh viên đông. Vậy nên thay vào việc điểm danh theo cách cũ, sẽ dễ dàng hơn nếu nhà trường, trung tâm giáo dục, giáo viên có sử dụng phần mềm để hỗ trợ.

### Thông tin chung

Ngày nay, thế giới đang bước vào thời đại công nghệ 4.0 với sự bùng nổ mạnh mẽ của cuộc cách mạng công nghệ. Vì thế trường Đại học Ngoại ngữ - Tin học Huflit tạo ra một ứng dụng giúp quản lý việc điểm danh của sinh viên một cách dễ dàng hơn, trung thực và minh bạch về kết quả điểm danh chính xác nhất. Cơ cấu tổ chức sẽ gồm có:

* Ban Giám Hiệu:
* Hiệu trưởng: Chịu trách nhiệm quản lý và điều hành toàn bộ hoạt động của trường.
* Phó Hiệu trưởng: Hỗ trợ Hiệu trưởng trong các công việc quản lý và điều hành.
* Các Phòng Ban:
* Phòng Đào tạo Đại học: Quản lý chương trình đào tạo và các hoạt động liên quan đến giảng dạy và học tập của sinh viên.
* Phòng Công tác Sinh viên: Quản lý các hoạt động hỗ trợ sinh viên, giải quyết các vấn đề liên quan đến đời sống sinh viên.
* Các Phòng Khoa:
* Khoa Ngoại ngữ: Cung cấp các chương trình đào tạo về ngoại ngữ.
* Khoa CNTT (Công nghệ Thông tin): Đào tạo các chuyên ngành liên quan đến công nghệ thông tin.
* Khoa Quan hệ Quốc tế: Đào tạo các chuyên ngành liên quan đến quan hệ quốc tế.
* Khoa Luật: Cung cấp các chương trình đào tạo về luật.
* Khoa Quản trị Kinh doanh: Đào tạo các chuyên ngành liên quan đến quản trị kinh doanh.
* Khoa Đông phương học: Đào tạo các chuyên ngành liên quan đến nghiên cứu Đông phương.
* Khoa Du lịch - Khách sạn: Cung cấp các chương trình đào tạo về du lịch và khách sạn.
* Các Đơn Vị Phục Vụ:
* Giảng viên: Trực tiếp giảng dạy và hướng dẫn sinh viên.
* Bảo vệ: Đảm bảo an ninh, trật tự trong khuôn viên trường.
* Sinh viên: Đối tượng học tập và tham gia các hoạt động tại trường.

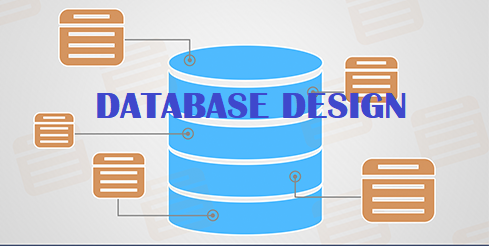
### Cơ cấu tổ chức

## Yêu cầu lưu trữ

Trình bày cụ thể các thông tin cần lưu trữ của đề tài, ví dụ như trong Bảng 1.

Bảng 1.1. Thông tin lưu trữ của nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Dữ liệu** | **Kiểu** | **Hình thức nhập** | **Ràng buộc** |
| 1 | Mã nhân viên | Text | Nhập từ bàn phím | Không trùng nhau |
| 2 | Họ tên nhân viên | Text | Nhập từ bàn phím |  |
| 3 | Phòng ban | Text | Chọn từ danh sách | Danh sách phòng ban của công ty |
| 4 | … |  |  |  |



Hình 1.1. Thiết kế CSDL

# CƠ SỞ LÝ THUYẾT

## Giới thiệu về các Cơ sở lý thuyết

### **Mô hình ERD**

* Mô hình quan hệ thực thể ERD (được viết tắt là Entity Relationship model) được chen giới thiệu vào năm 197
* Là một mô hình được sử udnjg rổng rãi trong các bản thiết kế cơ sở dữ liệu ở mức khái niệm, được xây dựng dựa trên việc nhận thức thể giới thực thông qua tập các đối tượng được gọi là các thực thể và các mối quan hệ giữa các đối tượng này.
* Mô hình ERD bào gồm các thực thể, thể hiện, thuộc tính, mối kết hợp, vai trò và bản số.
* A diagram of a flowchart

  Description automatically generated

*Hình 23: Mô hình ERD (hình minh họa)*

#### **2.1.1.1. Thực thể**

- Là các đối tượng của thế giới thực bao gồm

- Tập hợp các thực thể có tính chất giống nhau tạo thành 1 tập thực thể

- Ký hiệu: Hình chữ nhật, bên trong có đặt tên của tập thực thể.

- Tên của tập thực thể là danh từ hoặc cụm danh từ

### **Phụ thuộc hàm**

#### **2.1.2.1. Các vấn đề thường gặp khi tổ chức dữ liệu:**

- Sự dư thừa

- Thiếu thống nhất.

- Dị thường khi thêm bộ.

- Dị thường khi xóa bộ.

#### 2.1.2.2. Định nghĩa phụ thuộc hàm

- Phụ thuộc hàm là công cụ dùng để biểu diễn một cách hình thức mối quan hệ dữ liệu của các thuộc tính bên trong cơ sở dữ liệu.

- Thông qua cách biểu diễn phụ thuộc hàm, có thể dễ dàng xác định khóa của quan hệ.

- Phương pháp biểu diễn này có vai trò quan trọng trong các phương pháp thiết kế một lược đồ quan niệm của CSDL:

* + Nhằm tạo ra những quan hệ độc lập nhau, giảm thiểu sự trùng lắp, dư thừa dữ liệu
  + Do đó, giảm bớt các sai sót khi cập nhật dữ liệu của người sử dụng.
  + Ngoài ra, còn dùng để đánh giá chất lượng thiết kế một CSDL.
* Cho r(U), với r là quan hệ (hoặc bảng) và U là tập thuộc tính. Cho A, B ⊆ U (với A, B là tập các thuộc tính), phụ thuộc hàm A → B (A xác định B) được định nghĩa là: ∀t, t’ ∈ r nếu t’. A = t.A thì t’. B=t.B
* A → B được gọi là phụ thuộc hàm hiển nhiên nếu B ⊆ A
* A → B được gọi là phụ thuộc hàm đầy đủ vào A nếu ∀ A’ ⊂ A thì đều không có phụ thuộc hàm A’ → B
* A → B phải là nguyên tố tức không có thuộc tính nào phụ thuộc vào mỗi phần của (nghĩa là: A’ ∈ A mà B ∈ A').

**Ví dụ:**

Cho quan hệ CHITIET\_HD (SoHD, MaMH, Soluong, Dongia, Trigia) có phụ thuộc hàm sau:

f1: SoHD, MaMH → Soluong

f2: SoHD, MaMH → Dongia

f3: SoHD, MaMH → Trigia

f4: Soluong, Dongia → Trigia

(Thuộc tính Dongia không phụ thuộc đầy đủ vào khóa

#### **2.1.2.3. Hệ luật dẫn Armstrong**

* **Hê tiền đề Armstrong:**
* Luật phản xạ: Y ⊆ X ==> X → Y
* Luật thêm vào: Nếu X → Y và Z ⊆ W thì X, W → Y, Z
* Luật bắc cầu: Nếu X → Y và Y → Z thì X → Z
* **Một số luật dẫn suy từ hệ tiên đề Amstrong:**
* Luật phân rã: Nếu X→ Y, Z thì X→Y và X→ Z
* Luật hội: Nếu X→ Y và X → Z thì X → Y, Z
* Luật giả bắc cầu: Nếu X → Y và WY → Z thì WX → Z

#### **2.1.2.4. Bao đóng**

* **Bao đóng thuộc tập phụ thuộc hàm F:** Bao đóng(closure) của tập phụ thuộc hàm F (ký hiệu F+) là tập tất cả các phụ thuộc hàm có thể suy ra từ F dựa vào hệ tiên đề Armstrong
* **Ví dụ:** Cho lược đồ quan hệ Q(ABCDEGH) và tập phụ thuộc hàm F như sau F = {AB→C; B→D; CD→E; CE→GH; G→A}
* Áp dụng hệ tiên đề Amstrong, tìm một chuỗi suy diễn AB →E

1. AB→C (cho trước f1)
2. AB→AB (Luật phản xạ)
3. AB→ B (luật phân rã)
4. B→D (cho trước f2)
5. AB→D (bắc cầu 3 & 4)
6. AB→CD. (hợp 1 & 5)
7. CD→E (cho trước f3)
8. AB→E (bác cầu 6 & 7). Kết thúc

- **Bao đóng của tập thuộc tính X:**

* Bao đóng của tập thuộc tính X nhờ vào tập phụ thuộc F, ký hiệu X + F, được định nghĩa:

A black text on a white background

Description automatically generated

Hình 24: Minh họa bao đóng

* Khái niệm bao đóng của một tập thuộc tính được sử dụng để kiểm tra xem một phụ thuộc hàm f có được suy dẫn từ F hay không
* Bao đóng của tập thuộc tính X thực chất là tập tất cả các thuộc tính mà có thể suy ra nó từ tập thuộc tính X ban đầu.

#### **2.1.2.5. Phụ thuộc hàm tương đương**

* Hai tập phụ thuộc hàm F và G được gọi là tương đương với nhau nếu bao đóng của chúng giống nhau F + = G+.
* Nghĩa là: ∀f ∈ F thì f ∈ G+ và ∀g∈ G thì g ∈ F +
* Ký hiệu: F ≡ G
* Ví dụ: Xét tính tương đương của hai tập phụ thuộc hàm sau, định nghĩa trên Q(ABCDE) F = {A→BC; A →D; CD →E} G = {A →BCE; A →ABD; CD →E}

Bước 1: Tìm bao đóng của F

1.1. Tập nguồn {A}

1.2. Do A → BC nên ta có {ABC}

1.3. Do A → D nên ta có {ABCD}

1.4. Do CD → E nên ta có {ABCDE}

F+={ABCDE}

Bước 2: Tìm bao đóng của G

2.1. Tập nguồn {A}

2.2. Do A → BCE nên ta có {ABCE}

2.3. Do A → ABD nên ta có {ABCDE}

G+={ABCDE}

Kết luận: F tương đương G

#### 2.1.2.6. Phủ và phủ tối thiểu

* Định nghĩa về phủ:
* Một tập phụ thuộc hàm F’ được gọi là phủ của F nếu F’≡ F
* Ví dụ ở phụ thuộc hàm tương đương, G là một phủ của F

- Phủ tối thiểu: K là phủ tối thiểu của F nếu K là Phủ của F, đồng thời thỏa 3 điều kiện:

• Vế phải của các phụ thuộc hàm trên K chỉ chứa một thuộc tính.

• K chỉ gồm những phụ thuộc hàm đầy đủ.

• Không chứa pth thừa: ⎤∃ (X→A) ∈ K sao cho K ≡ (K – {X→A})

- Thuật toán tìm Phủ tối thiểu của F:

Bước 1: Gán tập G = F.

G= {A→BC; A →D; CD →E, A →A}

Bước 2: Tách vế phải phụ thuộc hàm trong G sao cho vế phải của mỗi phụ thuộc hàm chỉ chứa 1 thuộc tính.

G= {A→B; A→ C; A →D; CD →E, A →A}

Bước 3: Tìm phụ thuộc đầy đủ bằng cách loại bỏ các thuộc tính dư thừa ở vế trái của từng phụ thuộc hàm.

G= {A→B; A→ C; A →D; CD →E, A →A}

Lưu ý: Chỉ xét các phụ thuộc hàm có vế trái của có từ 2 thuộc tính trở lên.

• Xét CD →E: thấy rằng đây là phụ thuộc hàm đầy đủ nên giữ nguyên.

Bước 4: Loại bỏ thuộc tính dư thừa.

G= {A→B; A→ C; A →D; CD →E, A →A} Loại bỏ A →A.

Đáp án: Phủ tối thiểu của F là G= {A→B; A→ C; A →D; CD →E}

### **Các dạng chuẩn**

* Chuẩn hóa là quá trình tích bảng (Phân rã) thành các bảng nhỏ hơn dựa vào các phụ thuộc hàm. Các dạng chuẩn là các chỉ dẫn để thiết kế các bảng trong cơ sở dữ liệu.
* Mục đích của chuẩn hóa là loại bỏ các dư thừa dữ liệu và lỗi khi thao tác dư thừa và các lỗi thao tác dữ liệu (Insert, Delete, Update). Nhưng chuẩn hóa làm tăng thời gian truy vấn.
* Có 4 dạng chuẩn: dạng chuẩn (1NF), dạng chuẩn 2(2NF), dạng chuẩn 3(3NF) và dạng chuẩn BOYCE CODD KENT (BCK).

#### **2.1.3.1. Dạng chuẩn của lược đồ quan hệ**

Chuấn lượng thiết kế của một lược đồ cơ sở dữ liệu có thể đánh giá dựa trên nhiều tiêu chuẩn trong đó:

* Sự trùng lắp thông tin.
* Chi phí kiểm tra các rang buộc toàn vẹn.
* Bảo toàn qui tắc quản lý.
* Bảo toàn thông tin.
* Ví dụ: Xét một thể hiện của quan hệ quản lý học tập của sinh viên QLHT (MsSV, Ten, NS, Phai, ĐC, MsLop, MsMH, TenMH, Diem)

F = {f1: MsSV → Ten, NS, Phai, ĐC, MsLop.

* f2: MsLop → TenLop.
* f3: MsMH → TenMH.
* f4: TenMH → MsMH.
* f4: MsSV, MsMH → Diem}

#### **2.1.3.2. Dạng chuẩn 1 (1NF)**

- Định nghĩa: Một quan hệ đạt dạng chuẩn 1(DC1) là một quan hệ mà các giá trị trên từng thuộc tính phải là giá trị nguyên tố và còn được gọi là cấu trúc phẳng.

A screenshot of a computer

Description automatically generated

#### **2.1.3.3. Dạng chuẩn 2 (2NF)**

Định nghĩa: Một quan hệ đạt dạng chuẩn 2(DC2) nếu và chỉ nếu nó đạt DC1 và tất cả các thuộc tính không khóa phụ thuộc đầy đủ vào khóa.

Lưu ý: Dạng chuẩn 2 có thể vi phạm khi quan hệ khóa gồm nhiều hơn một thuộc tính.

A screenshot of a computer

Description automatically generated

#### **2.1.3.4. Dạng chuẩn 3 (3NF)**

Định nghĩa: Một quan hệ đạt dạng chuẩn 3(DC3) nếu và chỉ nếu nó đạt DC2 và tất cả các thuộc tính không khóa không phụ thuộc bắc cầu vào khóa.

A close-up of a computer screen

Description automatically generated

#### **2.1.3.5. Dạng Chuẩn BOYCE CODD KENT (BCK)**

Định nghĩa: Một quan hệ đạt dạng chuẩn BOYCE CODD KENT(BCK) nếu với mọi phụ thuộc hàm X→A không hiển nhiên định nghĩa trên quan hệ, A∉X thì X (mọi vế trái của tất cả các phụ thuộc hàm) là một siêu khóa của quan hệ, nghĩa là X là một khóa hoặc chứa một khóa.

A close-up of a table

Description automatically generated

#### **2.1.3.6. Dạng chuẩn 4**

Phụ thuộc đa trị: Một quan hệ R có phụ thuộc đa trị X ->-> Y nếu với mỗi giá trị của X, có một tập hợp các giá trị của Y liên kết với nó, độc lập với các thuộc tính khác trong R. Nói cách khác, nếu ta biết giá trị của X, ta có thể xác định được tất cả các giá trị có thể có của Y mà không cần quan tâm đến giá trị của các thuộc tính khác.

**Phụ thuộc đa trị hiển nhiên:** Một phụ thuộc đa trị X ->-> Y được gọi là hiển nhiên nếu một trong hai điều kiện sau được thỏa mãn:

Y là tập con của X (Y ⊆ X)

Y là phần bù của X trong R (Y = R - X)

Mục đích của dạng chuẩn 4: là không cho phép xuất hiện phụ thuộc hàm đa trị không hiển nhiên trên một quan hệ.

• Nếu có, cần tách nhỏ các quan hệ nhằm biến các phụ thuộc hàm đa trị không hiển nhiên thành hiển nhiên trong các quan hệ mới để không cần kiểm tra nữa.

• Cách tách phụ thuộc đa trị không hiển nhiên:

✓ Biến đổi các phụ thuộc đa trị không hiển nhiên trong một cấu trúc này thành phụ thuộc đa trị hiển nhiên trong 1 cấu trúc khác

✓ Ví dụ: Trên Q(X, Y, Z) có phụ thuộc đa trị không hiển nhiên X --> Y

• Ta tạo ra cấu trúc: C = {Q1(X,Y); Q2(X,Z) }

### Bảo toàn thông tin

* Ngăn chặn các truy cập không được phép.
* Hạn chế tối đa các sai sót của người dùng.
* Đảm bảo thông tin không bị mất và thay đổi ngoài ý muốn.
* Không tiết lộ nội dung dữ liệu cũng như chương trình xử lý.
* Các giải pháp chủ yếu cho bảo mật hệ thống gồm chính sách ý thức, phân quyền truy cập và nhận dạng người dùng, mã hóa thông tin và nén dữ liệu.

### Lập trình CSDL

**a. Biến cục bộ:**

Biến cục bộ là đối tượng dùng về lưu trữ một dữ liệu đơn thuộc một kiểu dữ liệu cụ thể (int, char, date, ...).

Các biến này thường được dùng để:

* Lưu trữ số lần lập của một câu lệnh lặp.
* Lưu trữ giả trị dùng để kiểm tra trong câu lệnh điều kiện.
* Lưu từ giá trị trả về từ các hàm con

Khai báo biển cục bộ cần thông tin sau:

* Từ khóa khai báo biển: DECLARE
* Tên biển: bắt đầu bằng ký tự @
* Khai báo kiểu dữ liệu và chiều dài kiểu dữ liệu (nếu là kiểu chuỗi)
* Cú pháp: DECLARE @tên\_biến <\_ kiểu\_dữ\_liệu\_ >

**Ví dụ:** DECLARE @MAKH CHAR (10), HD11 CHAR (10), @NGAY DATE, @kq int

Biến cục bộ chỉ nhận giá trị đơn. Nếu kết quả của câu lệnh SELECT trả về là một cột gồm nhiều giá trị thì biến sẽ được gán giá trị trả về cuối cùng. Nếu câu truy vấn không trả về kết quả thì giá trị của biến là NULL.

**b. Biến hệ thống**

* Không giống các ngôn ngữ lập trình khác, T\_SQL không có khái niệm
* biến toàn cục.
* Thay vào đó Microsoft SQL Server cung cấp cho người lập trình danh
* sách các biến hệ thống.
* Biến hệ thống luôn bắt đầu bằng @@
* Người lập trình không thể gán giá trị vào biến hệ thống.
* Giá trị mà chúng đang lưu trữ là do hệ thống Microsoft SQL Server cung cấp.

• Ví dụ: @@SERVERNAME, @@LANGUAGE, @@VERSION, …

**c. Cấu trúc điều khiển**

* Cấu trúc điều khiển cho phép chương trình kiểm tra một điều kiện là đúng thì sẽ thực hiện một hoạt động cụ thể.
* Cú pháp:

IF <điều kiện>

{Đoạn chương trình sẽ thực hiện khi điều kiện đúng}

[ ELSE

{Đoạn chương trình sẽ thực hiện khi điều kiện Sai}]

* Trong trường hợp, nếu phải thực hiện nhiều hơn 1 lệnh thì bổ sung thêm cặp lệnh BEGIN-END
* Cấu trúc điều khiển thường dùng kết hợp với biển cục bộ để kiểm tra tính hợp lệ giá trị của dữ liệu

DECLARE @lk char (10), @KQ INT EXEC @KQ sp\_VIP @MAKH, @LK OUT IF @KQ = 1 UPDATE KHACHHANG

SET LOAIKHACH = @lk

WHERE MAKH = @MAKH return @KQ

**d. Cấu trúc lặp**

* Cấu trúc lặp cho phép thực hiện cùng một thao tác với nhiều đối
* tượng khác nhau.
* Cú Pháp: **WHILE** <điều kiện>
* {Đoạn chương trình sẽ thực hiện khi điều kiện đúng | **BREAK** | **CONTINUE**}
* Trong trường hợp, nếu phải thực hiện nhiều hơn 1 lệnh thì bổ sung
* thêm cặp lệnh BEGIN-END
* **Ví dụ:** In ra màn hình các số từ 1 đến 10.

DECLARE @var INT;

SET @var=1;

-- Khai báo cấu trúc lặp

WHILE @var <= 10

BEGIN

PRINT @var;

SET @var=@var+1;

END

**e. Store procedure (thủ tục thường trú)**

* **Khái niệm:** là tập hợp chứa các dòng lệnh, các biến và các cấu trúc điều kiển trong ngôn ngữ Transaction-SQL dùng để thực hiện một hành động nào đó.
* **Lợi ích của thủ tục:**
* Tối ưu hóa việc phân tích, biên dịch và thực thi câu lệnh.
* Thực hiện một yêu cầu bằng một câu lệnh đơn giản hơn thay vì phải sử dụng nhiều dòng lệnh SQL khi thực thi.
* Tăng khả năng bảo mật khi cấp phát quyền thông qua thủ tục.
* Tốc độ xử lý của các thủ tục nội tại rất nhanh.
* **Phân loại thủ tục:**
* Thủ tục hệ thống: Bắt đầu bằng chữ sp\_ và hầu hết tất cả các thủ tục hệ thống được lưu trữ bên trong CSDL Master.
* Thủ tục do người dùng xây dựng
* **Tạo thủ tục**
* Cú pháp khai báo thủ tục:

CREATE PROC[EDURE] Tên\_thủ\_tục

AS

BEGIN

các\_lệnh\_SQL

END

* Cú pháp thực thi thủ tục:

EXEC[UTE]Tên\_thủ\_tục

* Cú pháp xóa thủ tục:

DROP PROC[EDURE] Tên\_thủ\_tục

* Thay đổi nội dung thủ tục

ALTER PROC<\_Tên\_thủ\_tục><Ds\_tham\_số>

AS

Khai báo biến cục bộ

Thân chương trình

- Thủ tục với tham số đầu vào :

CREATE PROC<Tên\_thủ\_tục>

@Tên\_tham\_số kiểu dữ liệu [=giá trị]

- Thủ tục với tham số đầu ra(luôn đặt các tên tham chiếu muốn xuất ở cuối cùng):

CREATE PROC <Tên thủ tục>

@Tên\_tham\_chiếu Kiểu\_dữ\_liệu OUTPUT out

AS

Khai báo biến cục bộ

Thân chương trình

create proc sp\_TSTIEN @MAKH CHAR(10), @TTONG INT out

as

If exists (SELECT \* FROM HOADON WHERE MAKH= @MAKH

begin

SELECT @TTONG = SUM(TONGTIEN)

FROM HOADON

WHERE MAKH = @MAKH

return 1

end

else return 2

Thực thì thủ tục:

DECLARE @TENKH NVARCHAR(20), @MAKH CHAR(10), @KQ INT

SET @TENKH =N’Nguyễn An’

EXEC @KQ = SP\_MaKHang @TENKH, @MAKH OUT

IF @KQ = 1

PRINT N’Khách hàng có tên là: ‘ + @TENKH + ‘Có mã khách hàng là: ‘+ @MAKH

ELSE

PRINT N’Không tồn tại’ + @TENKH

### Ràng buộc toàn vẹn

### NoSQL và NewSQL

Thuật ngữ NoSQL đang phân loại cơ sở dữ liệu mang tính mô tả ở SQL". NoSQL là một danh mục cơ sở dữ liệu toàn diện được phát triển để khắc phục các sự cố do cơ sở dữ liệu SQL tạo ra. Chúng được gọi là tài liệu lược đồ ít lưu trữ dữ trong tài liệu, biểu đồ, khóa-giá trị và kiểu không có thứ tự.

#### **2.1.7.1. Ưu điểm của NoSQL:**

* Chúng mở rộng quy mô tốt hơn các hệ thống truyền thống khi có nhu cầu về hành vi năng động.
* Các hệ thống này được tối ưu hóa tốt hơn cho dữ liệu không quan hệ.
* Cho phép thực hiện các thao tác ghi trên lược đồ.

#### **2.1.7.2. Nhược điểm của NoSQL:**

* Hệ thống được xây dựng bằng NoSQL về cơ bản là không mang tỉnh giao dịch.
* Khối lượng dữ liệu được tạo ra rất lớn và không cung cấp bất kỳ khả năng cơ sở dữ liệu truyền thống nào.
* NoSQL không tuân theo sự nhất quán khi nhiều giao dịch được thực hiện đồng thời.

#### **2.1.7.3. NewSQL:**

Thuật ngữ NewSQL phân loại cơ sở dữ liệu là sự kết hợp của mô hình quan hệ với sự tiến bộ về khả năng mở rộng, tính linh hoạt với các loại dữ liệu. Các cơ sở dữ liệu này tập trung vào các tính năng không có trong NoSQL, mang lại sự đảm bảo nhất quản mạnh mẽ. Điều này bao gồm hai lớp dữ liệu, một lớp quan hệ và một lớp lưu trữ khóa-giá trị.

#### **2.1.7.4. Ưu điểm của NewSQL:**

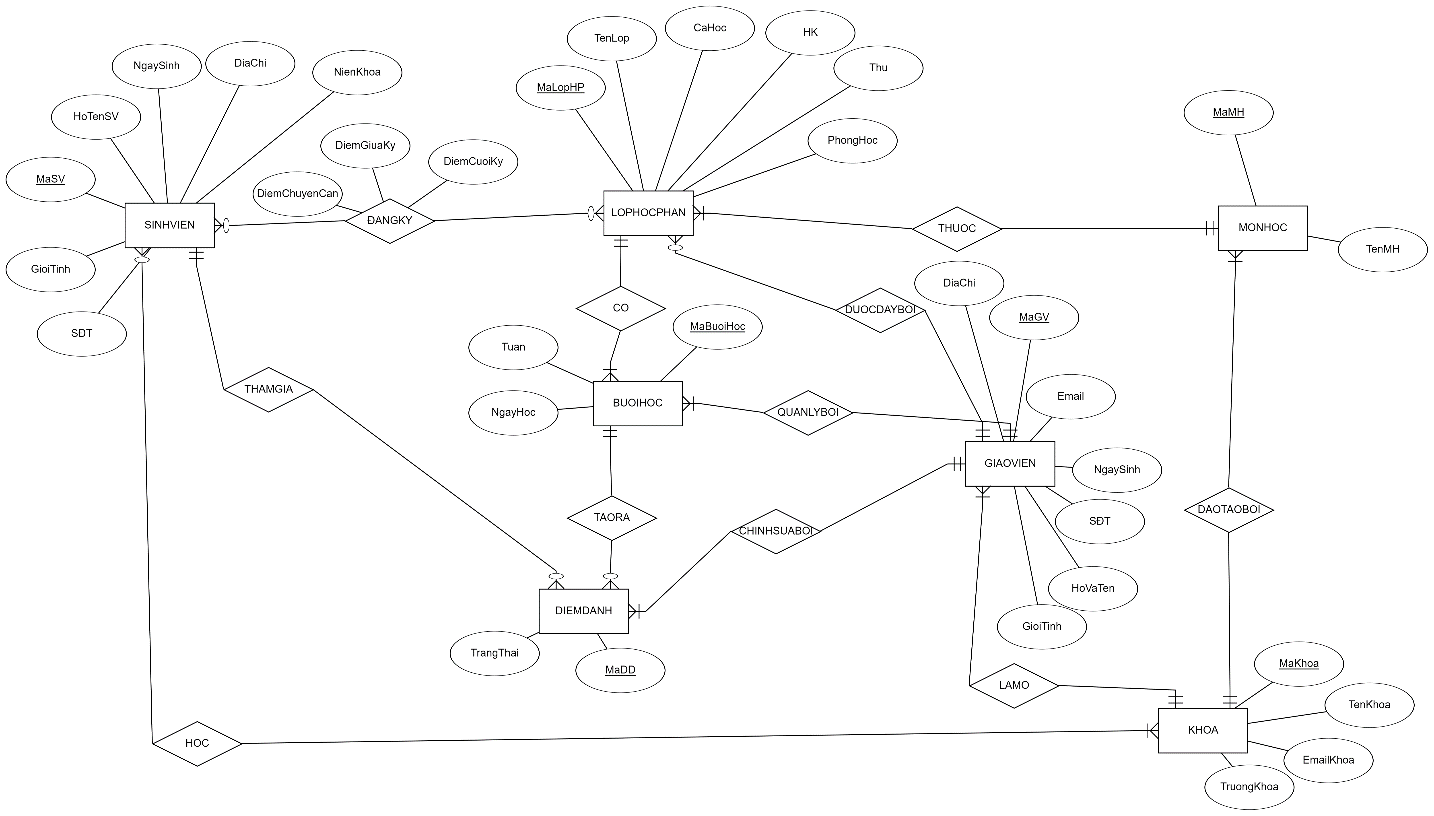
* NewSQL giới thiệu cách triển khai mới cho cơ sở dữ liệu quan hệ truyền thống.
* NewSQL tập hợp những ưu điểm của SQL và NoSQL.
* Có thể dễ dàng di chuyển giữa loại và nhu cầu của người dùng

#### **2.1.7.5. Nhược điểm của NewSQL:**

* Họ cung cấp quyền truy cập một phần vào các hệ thống truyền thống phong phú.
* Nó có thể gây ra sự có trong kiến trúc bộ nhớ khi vượt quá khối lượng dữ liệu.
* Nền tảng cốt lõi của các cơ sở dữ liệu như vậy là các hệ thống quan hệ

# PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

## Mô hình thực thể kết hợp



## Mô tả các thực thể

### Thực thể sinh viên

Bảng 3.1. Mô tả thực thể sinh viên

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaSV** | VARCHAR | 10 | Khác rỗng |  |
| **2** | HoTenSV | NVARCHAR | 50 | Khác rỗng | Có dấu tiếng việt |
| **3** | NgaySinh | DATE |  | Khác rỗng |  |
| **4** | DiaChi | NVARCHAR | 70 | Khác rỗng | Có dấu tiếng việt |
| **5** | GioiTinh | BIT |  | Khác rỗng | 0:Nam,1:Nữ |
| **6** | SĐT | VARCHAR | 10 | Khác rỗng |  |
| **7** | NienKhoa | DATE |  | Khác rỗng |  |

### Thực thể giảng viên

Bảng 3.2. Mô tả thực thể giảng viên

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaGV** | VARCHAR | 10 | Khác rỗng |  |
| **2** | HoTenGV | NVARCHAR | 50 | Khác rỗng | Có dấu tiếng việt |
| **3** | GioiTinh | BIT |  | Khác rỗng | 0:Nam,1:Nữ |
| **4** | DiaChi | NVARCHAR | 70 | Khác rỗng | Có dấu tiếng việt |
| **5** | NgaySinh | DATE |  | Khác rỗng |  |
| **6** | SĐT | VARCHAR | 10 | Khác rỗng |  |
| **7** | Email | VARCHAR | 30 | Khác rỗng |  |

### Thực thể khoa

Bảng 3.2. Mô tả thực thể khoa

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaKhoa** | VARCHAR | 10 | Khác rỗng |  |
| **2** | TenKhoa | NVARCHAR | 30 | Khác rỗng | Có dấu tiếng việt |
| **3** | DiaChi | NVARCHAR | 70 | Khác rỗng | Có dấu tiếng việt |
| **4** | EmailKhoa | VARCHAR | 30 | Khác rỗng |  |

### Thực thể môn học

Bảng 3.2. Mô tả thực thể môn học

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaMH** | VARCHAR | 10 | Khác rỗng |  |
| **2** | TenMH | NVARCHAR | 30 | Khác rỗng | Có dấu tiếng việt |

### Thực thể lớp học phần

Bảng 3.2. Mô tả thực thể lớp học phần

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaLopHP** | VARCHAR | 10 | Khác rỗng |  |
| **2** | TenLop | NVARCHAR | 30 | Khác rỗng | Có dấu tiếng việt |
| **3** | DiaDiem | NVARCHAR | 70 | Khác rỗng | Có dấu tiếng việt |
| **4** | ThoiGianHoc | DATE |  | Khác rỗng |  |
| **5** | Thu | INT |  | Khác rỗng |  |
| **6** | PhongHoc | VARCHAR | 6 | Khác rỗng | bắt đầu bằng A hoặc B |

### Thực thể lớp điểm danh

Bảng 3.2. Mô tả thực thể lớp điểm danh

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaDiemDanh** | VARCHAR | 10 | Khác rỗng |  |
| **2** | TrangThai | INT |  | Khác rỗng | 0:Có,1:Trễ,2: Vắng |

### Thực thể buổi học

Bảng 3.2. Mô tả thực thể buổi học

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc** | **Ghi chú** |
| **1** | **MaBuoiHoc** | VARCHAR | 10 | Khác rỗng |  |
| **2** | NgayHoc | DATE |  | Khác rỗng |  |
| **3** | Tuan | INT |  | Khác rỗng |  |

## Lược đồ cơ sở dữ liệu quan hệ

### Biểu diễn dạng text

**SINHVIEN**(MaSV,HoTenSV,NgaySinh,DiaChi,GioiTinh,SĐT,NienKhoa,

#MaKhoa)

**GIANGVIEN**(MaGV,HoTenGV,GioiTinh,DiaChi,NgaySinh,SĐT,Email, #MaKhoa)

**LOPHOCPHAN**(MaLopHP,TenLop,CaHoc,HK,Thu, #MaMH,#MaGV)

**MONHOC**(MaMH,TenMH, #MaKhoa)

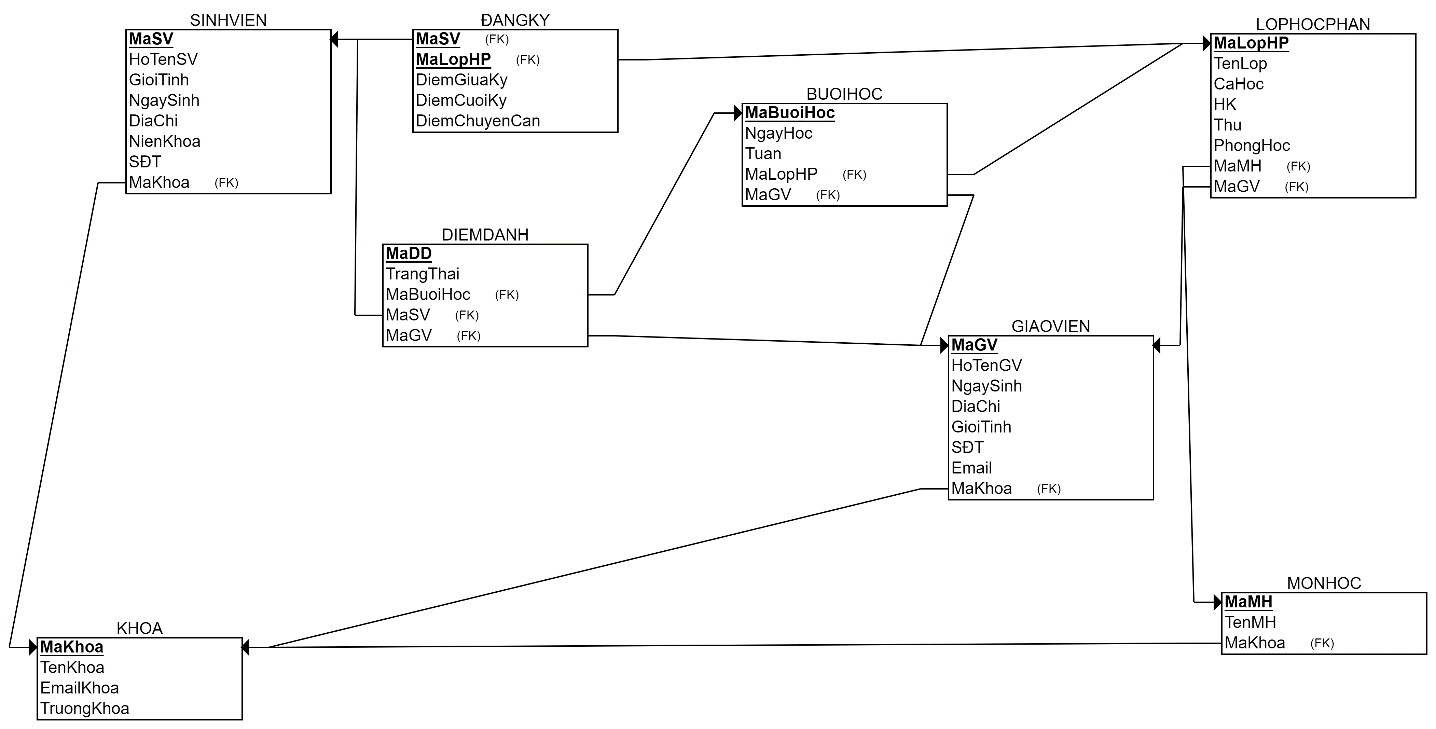
**KHOA**(MaKhoa,TenKhoa,TruongKhoa,EmailKhoa)

**BUOIHOC**(MaBuoiHoc,Tuan,NgayHoc,Thu, #MaLopHP,#MaGV)

**DIEMDANH**(MaDD,TrangThai, #MaBuoiHoc,#MaSV,#MaGV)

**DANGKY**(#MaSV,#MaLopHP,DiemChuyenCan,DiemGiuaKy,DiemCuoiKy)

### Biểu diễn dạng hình (dùng công cụ ERDPlus)



### Phân tích dạng chuẩn của lược đồ quan hệ

**Quan hệ: SINHVIEN**

* Khóa:MaSV
* Phụ thuộc hàm:MaSV -> HoTenSV, MaSV -> GioiTinh, MaSV -> NgaySinh, MaSV -> DiaChi, MaSV-> NiemKhoa, MaSV -> SĐT, MaSV

-> MaKhoa

* Quan hệ SINHVIEN đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ SINHVIEN đã đạt chuẩn 2 vì quan hệ SINHVIEN đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaSV
* Quan hệ SINHVIEN đã đạt được chuẩn 3 vì quan hệ SINHVIEN đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ SINHVIEN đã đạt chuẩn BCNF vì quan hệ SINHVIEN đã đạt chuẩn 3 và khóa chính MaSV xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaSV là siêu khóa của quan hệ SINHVIEN. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: KHOA**

* Khóa:MaKhoa
* Phụ thuộc hàm: MaKhoa -> TenKhoa, MaKhoa -> EmailKhoa, MaKhoa -> TruongKhoa
* Quan hệ KHOA đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ KHOA đã đạt chuẩn 2 vì quan hệ KHOA đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaKhoa
* Quan hệ KHOA đã đạt được chuẩn 3 vì quan hệ KHOA đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ KHOA đã đạt chuẩn BCNF vì quan hệ KHOA đã đạt chuẩn 3 và khóa chính MaKhoa xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaKhoa là siêu khóa của quan hệ KHOA. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: GIANGVIEN**

* Khóa:MaGV
* Phụ thuộc hàm: MaGV -> HoTenGV, MaGV -> GioiTinh, MaGV -> NgaySinh, MaGV -> DiaChi, MaSV-> Email, MaSV -> SĐT, MaSV

-> MaKhoa

* Quan hệ GIANGVIEN đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ GIANGVIEN đã đạt chuẩn 2 vì quan hệ GIANGVIEN đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaGV
* Quan hệ GIANGVIEN đã đạt được chuẩn 3 vì quan hệ GIANGVIEN đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ GIANGVIEN đã đạt chuẩn BCNF vì quan hệ GIANGVIEN đã đạt chuẩn 3 và khóa chính MaGV xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaGV là siêu khóa của quan hệ GIANGVIEN. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: MONHOC**

* Khóa:MaMH
* Phụ thuộc hàm: MaMH -> TenMH, MaMH -> MaKhoa
* Quan hệ MONHOC đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ MONHOC đã đạt chuẩn 2 vì quan hệ MONHOC đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaMH
* Quan hệ MONHOC đã đạt được chuẩn 3 vì quan hệ MONHOC đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ MONHOC đã đạt chuẩn BCNF vì quan hệ MONHOC đã đạt chuẩn 3 và khóa chính MaMH xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaMH là siêu khóa của quan hệ MONHOC. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: LOPHOCPHAN**

* Khóa:MaLopHP
* Phụ thuộc hàm: MaLopHP -> TenLop, MaLopHP -> CaHoc, MaLopHP -> HK, MaLopHP -> Thu, MaLopHP -> PhongHoc, MaLopHP -> MaMH, MaLopHP -> MaGV
* Quan hệ LOPHOCPHAN đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ LOPHOCPHAN đã đạt chuẩn 2 vì quan hệ LOPHOCPHAN đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaLopHP
* Quan hệ LOPHOCPHAN đã đạt được chuẩn 3 vì quan hệ LOPHOCPHAN đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ LOPHOCPHAN đã đạt chuẩn BCNF vì quan hệ LOPHOCPHAN đã đạt chuẩn 3 và khóa chính MaLopHP xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaLopHP là siêu khóa của quan hệ LOPHOCPHAN. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: DANGKY**

* Khóa:MaSV, MaLopHP
* Phụ thuộc hàm: MaSV,MaLopHP -> DiemChuyenCan, MaSV,MaLopHP -> DiemGiuaKy, MaSV,MaLopHP -> DiemCuoiKy
* Quan hệ DANGKY đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ DANGKY đã đạt chuẩn 2 vì quan hệ DANGKY đã đạt chuẩn 1 và tất cả các thuộc tính không khóa đều phụ thuộc đầy đủ vào khóa chính MaSV và MaLopHP
* Quan hệ DANGKY đã đạt được chuẩn 3 vì quan hệ DANGKY đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ DANGKY đã đạt chuẩn BCNF vì quan hệ DANGKY đã đạt chuẩn 3 và khóa chính MaSV,MaLopHP xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaSV, MaLopHP là siêu khóa của quan hệ DANGKY. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

**Quan hệ: BUOIHOC**

* Khóa:MaBuoiHoc
* Phụ thuộc hàm: MaBuoiHoc -> NgayHoc, MaBuoiHoc -> Tuan, MaLopHP -> MaGV, MaLopHP, Tuan -> MaBuoiHoc
* Quan hệ BUOIHOC đã đạt chuẩn 1 vì tất cả các thuộc tính đều là nguyên tố
* Quan hệ BUOIHOC chưa đạt chuẩn 2 vì tất cả các thuộc tính không khóa chưa phụ thuộc đầy đủ vào khóa chính MaBuoiHoc
* Chuẩn hóa BUOIHOC để đạt dạng chuẩn 2:
* Bước 1: Tìm khóa chính của BUOIHOC (MaBuoiHoc, NgayHoc, Tuan, MaLopHP, MaGV)
* 1.1 K = (MaBuoiHoc, NgayHoc, Tuan, MaLopHP, MaGV)
* 1.2 Lần lượt loại bỏ các thuộc tính của K
* 1.2.1 Bỏ MaBuoiHoc: Ta có vì phụ thuộc hàm MaLHP, Tuan -> MaBuoiHoc nên K = (NgayHoc, Tuan, MaLopHP, MaGV)
* 1.2.2 Bỏ NgayHoc: Ta có vì phụ thuộc hàm MaLopHP, Tuan -> MaBuoiHoc và MaBuoiHoc -> NgayHoc nên MaLopHP, Tuan -> NgayHoc. Do đó K = (Tuan, MaLopHP, MaGV)
* 1.2.3 Bỏ Tuan: Ta có nên K = (Tuan, MaLopHP, MaGV)
* 1.2.4 Bỏ MaLopHP: Ta có nên K = (Tuan, MaLopHP, MaGV)
* 1.2.5 Bỏ MaGV: Ta có vì MaLopHP, Tuan -> MaBuoiHoc và MaBuoiHoc -> NgayHoc nên MaLopHP, Tuan -> NgayHoc và MaLopHP -> MaGV nên K = (Tuan, MaLopHP)
* 1.3 Vậy (MaLopHP, Tuan) là khóa chính của BUOIHOC
* Bước 2: Tách BUOIHOC thành 2 quan hệ vì MaGV không phụ thuộc hoàn toàn vào khóa chính (MaLopHP, Tuan) nên BUOIHOC được tách thành Quan hệ BUOIHOC(#MaLopHP, Tuan, NgayHoc) và Quan hệ BUOIHOC\_2(#MaLopHP, #MaGV)
* Bước 3: Loại bỏ quan hệ BUOIHOC\_2 vì nó là tập con của quan hệ LOPHOCPHAN
* Vậy BUOIHOC(#MaLopHP, Tuan, NgayHoc) đạt dạng chuẩn 2
* Quan hệ BUOIHOC đã đạt được chuẩn 3 vì quan hệ BUOIHOC đã đạt chuẩn 2 và không có thuộc tính không khóa nào phụ thuộc vào thuộc tính không khóa khác.
* Quan hệ LOPHOCPHAN đã đạt chuẩn BCNF vì quan hệ LOPHOCPHAN đã đạt chuẩn 3 và khóa chính MaLopHP xuất hiện ở vế trái của tất cả các phụ thuộc hàm, và MaLopHP là siêu khóa của quan hệ LOPHOCPHAN. Do đó, mọi phụ thuộc hàm đều có vế trái là siêu khóa.

### Lược đồ cơ sở dữ liệu quan hệ đạt Dạng chuẩn 3 (hoặc BCNF)

*Hình lược đồ quan hệ*

## Ràng buộc toàn vẹn

### R1: *“…Nội dung RBTV…”*

Bối cảnh:

Nội dung:

Bảng tầm ảnh hưởng:

### R2: *“…Nội dung RBTV…”*

Bối cảnh:

Nội dung:

Bảng tầm ảnh hưởng:

# TRIỂN KHAI CƠ SỞ DỮ LIỆU TRÊN Ms SQL SERVER

## Tạo cơ sở dữ liệu

## Nhập liệu mẫu

## Cài đặt các ràng buộc toàn vẹn

*(Tương ứng với mô tả RBTV ở Chương 3 – xóa dòng này)*

### R1: *“…Nội dung RBTV…”*

…

### R2: *“…Nội dung RBTV…”*

…

## Cài đặt các store procedure

*(Cài đặt các store procedure dùng để xử lý các thao tác chính của chương trình- xóa dòng này)*

### SP\_<tên SP> : *“…chức năng của SP…”*

### SP\_<tên SP> : *“…chức năng của SP…”*

### 

# TRIỂN KHAI CƠ SỞ DỮ LIỆU dạng nosql

## Chuyển đổi cơ sở dữ liệu quan hệ sang NoSQL

## Các câu lệnh tạo dữ liệu

*(Có thể thêm một vài dòng dữ liệu minh họa – xóa dòng này)*

## Truy vấn dữ liệu NoSQL

# KẾT LUẬN

## Kết quả đạt được và chưa đạt được

## Hướng phát triển mở rộng ứng dụng trong tương lai

# TÀI LIỆU THAM KHẢO

Sử dụng chức năng tạo danh mục tài liệu tham khảo của MS Word và dùng Style tham chiếu là IEEE.

# PHỤ LỤC

**Bảng phân công công việc của các thành viên trong nhóm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nội dung công việc** | MSSV1  Họ và tên SV1 | MSSV2  Họ và tên SV2 | MSSV3  Họ và tên SV3 |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| … |  |  |  |  |

*(Sinh viên thực hiện nội dung nào thì đánh dấu X tương ứng)*