

SỞ LĐT&XH THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG TRUNG CẤP SÀI GÒN



GIÁO TRÌNH

MÔ ĐUN: CƠ SỞ DỮ LIỆU

NGÀNH: QUẢN TRỊ MẠNG

TRÌNH ĐỘ: TRUNG CẤP

*(Ban hành kèm theo Quyết định số: /QĐ-CDN ngày tháng năm 2024
của Hiệu trưởng Trường Trung cấp Sài Gòn)*

Thành phố Hồ Chí Minh, năm 2024

(Lưu hành nội bộ)

TUYÊN BỐ BẢN QUYỀN

Tài liệu này thuộc loại sách giáo trình nên các nguồn thông tin có thể được phép dùng nguyên bản hoặc trích dùng cho các mục đích về đào tạo và tham khảo.

Mọi mục đích khác mang tính lệch lạc hoặc sử dụng với mục đích kinh doanh thiếu lành mạnh sẽ bị nghiêm cấm.

LỜI GIỚI THIỆU

Ngày nay, cơ sở dữ liệu đã có nhiều ứng dụng trong mọi hoạt động của xã hội. Muốn thiết kế và sử dụng cơ sở dữ liệu chúng ta phải nắm được các kỹ thuật cơ bản của cơ sở dữ liệu. Tài liệu học tập này nhằm trình bày các kỹ thuật cơ sở của cơ sở dữ liệu truyền thống, đó là mô hình liên kết thực thể, mô hình cơ sở dữ liệu quan hệ. Tài liệu cũng trình bày cách thiết kế một cơ sở dữ liệu quan hệ, cách sử dụng các phép toán đại số quan hệ để tạo, cập nhật và truy vấn cơ sở dữ liệu và khái niệm phụ thuộc hàm ứng dụng trong lý thuyết thiết kế và chuẩn hóa cơ sở dữ liệu quan hệ.

Trong quá trình biên soạn, tôi đã tham khảo và trích dẫn từ nhiều tài liệu được liệt kê tại mục Danh mục tài liệu tham khảo. Tôi chân thành cảm ơn các tác giả của các tài liệu mà tôi đã tham khảo.

Bên cạnh đó, giáo trình cũng không thể tránh khỏi những sai sót nhất định. rất mong nhận được những ý kiến đóng góp, phản hồi từ quý đồng nghiệp, các bạn người học và bạn đọc.

Trân trọng cảm ơn./.

Thành phố Hồ Chí Minh, ngày ... tháng..... năm 2022

Biên soạn

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU.....	14
1. Một số khái niệm cơ bản.....	15
2. Các mô hình dữ liệu.....	23
3. Mô hình thực thể kết hợp.....	23
CHƯƠNG 2: MÔ HÌNH DỮ LIỆU QUAN HỆ.....	28
MÔ HÌNH DỮ LIỆU QUAN HỆ.....	29
1. Các khái niệm cơ bản.....	29
1.1. Thuộc tính (Attribute).....	29
1.2. Lược đồ quan hệ (Relation schema).....	31
1.3. Quan hệ (Relation).....	31
1.4. Bộ giá trị (Tuple).....	32
1.5. Thể hiện của quan hệ (Occurrence of a Relation).....	33
1.6. Khóa - Siêu khóa - Khóa dự tuyển - Khóa chính - Khóa ngoại.....	33
1.7. Phụ thuộc hàm (Functional Dependency).....	35
1.8. Ràng buộc toàn vẹn (Integrity Constraint, Rule).....	36
2. Chuyển mô hình thực thể kết hợp sang mô hình dữ liệu quan hệ.....	36
3. Ngôn ngữ đại số quan hệ.....	37
3.1. Phép hợp 2 quan hệ (Union).....	37
3.2. Phép giao 2 quan hệ (Intersection).....	38
3.3. Phép trừ 2 quan hệ (Minus).....	39
3.4. Tích Decac của 2 quan hệ (Cartesian).....	39
3.5. Phép chia 2 quan hệ (Division).....	39
3.6. Phép chiếu (Projection).....	40
3.7. Phép chọn (Selection).....	42
3.8. Phép θ - kết.....	42
3.9. Các phép toán kết nối khác.....	47
CHƯƠNG 3: NGÔN NGỮ TRUY VẤN DỮ LIỆU SQL.....	52
1. Mở đầu.....	54
2. Tìm thông tin từ các cột của bản – Mệnh đề Select.....	55
Mệnh đề SELECT.....	55
3. Chọn các dòng của bản – Mệnh đề Where.....	56
5. Sắp xếp các dòng của bảng – Mệnh đề Order by.....	57

6.	Truy vấn thông tin từ nhiều bảng.....	58
6.1	Kết nối tự nhiên.....	58
6.2	Kết nối ngoại (Outer join).....	59
6.3	Truy vấn lồng nhau (Query with SubQuery).....	59
7.	Gom nhóm dữ liệu – mệnh đề Group by.....	62
8.1	Các hàm tính toán trên nhóm các bảng ghi.....	63
8.2	Các hàm tính toán trên bản ghi.....	64
	CHƯƠNG 4: RÀNG BUỘC TOÀN VỆN.....	68
1.	Ràng buộc toàn vẹn.....	70
1.1.	Khái niệm ràng buộc toàn vẹn.....	70
1.2.	Các yếu tố của ràng buộc toàn vẹn.....	71
1.2.1	Điều kiện của ràng buộc toàn vẹn.....	71
1.2.2.	Bối cảnh của Ràng buộc toàn vẹn.....	72
1.2.3.	Tầm ảnh hưởng của ràng buộc toàn vẹn.....	72
1.2.4.	Hành động khi RBTV bị vi phạm.....	73
2.	Phân loại ràng buộc toàn vẹn.....	73
2.1.	Ràng buộc toàn vẹn có bối cảnh là một quan hệ.....	74
2.1.1	Ràng buộc toàn vẹn về miền giá trị của thuộc tính.....	74
2.1.2.	Ràng buộc toàn vẹn liên thuộc tính.....	74
2.1.3.	Ràng buộc toàn vẹn liên bộ, liên thuộc tính.....	74
2.2.	Ràng buộc toàn vẹn có bối cảnh là nhiều quan hệ.....	75
2.2.1	Ràng buộc toàn vẹn về phụ thuộc tồn tại.....	75
2.2.2	Ràng buộc toàn vẹn tổng hợp (liên bộ - liên quan hệ).....	77
	CHƯƠNG 5: LÝ THUYẾT THIẾT KẾ CƠ SỞ DỮ LIỆU.....	79
1.	Các vấn đề gặp phải khi tổ chức dữ liệu.....	80
2.	Các phụ thuộc hàm.....	81
2.1.	Định nghĩa phụ thuộc hàm.....	82
2.2.	Cách xác định phụ thuộc hàm cho lược đồ quan hệ.....	82
2.3.	Một số tính chất của phụ thuộc hàm-Hệ luật dẫn Armstrong.....	82
3.	Bao đóng của tập phụ thuộc hàm và bao đóng của tập thuộc tính.....	84
3.1.	Bao đóng của tập phụ thuộc hàm F.....	84
3.2.	Bao đóng của tập phụ thuộc tính X.....	84
3.3.	Bài toán thành viên.....	85
3.4.	Thuật toán tìm bao đóng của một tập thuộc tính.....	85

4. Khóa của lược đồ quan hệ - Một số thuật toán tìm khóa.....	87
4.1. Định nghĩa khóa của quan hệ.....	87
4.2. Thuật toán tìm một khóa của một lược đồ quan hệ Q.....	88
4.3. Thuật toán tìm tất cả khóa của một lược đồ quan hệ.....	88
5. Phủ tối thiểu.....	91
5.1. Tập phụ thuộc hàm tương đương.....	91
5.2. Phủ tối thiểu.....	91
5.3. Thuật toán tìm phủ tối thiểu.....	92
6. Dạng chuẩn của lược đồ quan hệ.....	92
6.1. Một số khái niệm liên quan đến các dạng chuẩn.....	92
6.1.1. Phân rã.....	92
6.1.4 Dạng chuẩn (Normal Form-NF).....	97
6.2. Dạng chuẩn 1 (First Normal Form).....	97
6.3. Dạng chuẩn 2 (Second Normal Form).....	97
6.4. Dạng chuẩn 3 (Third Normal Form).....	98
6.5. Dạng chuẩn BC (Boyce codd Normal Form).....	99

GIÁO TRÌNH MÔN HỌC

1. Tên môn học: CƠ SỞ DỮ LIỆU

2. Mã môn học: MH13

3. Vị trí, tính chất, ý nghĩa và vai trò của môn học/mô đun:

+Vị trí:

Giáo trình dành cho người học trình độ Trung cấp tại Trường Trung cấp nghề Tây Ninh.

+ Tính chất:

Là môn học cơ sở.

+ Ý nghĩa và vai trò của môn học/mô đun:

Môn học cơ sở dữ liệu là môn học dành cho người học thuộc ngành quản trị hệ thống cơ sở dữ liệu trình độ cao đẳng nghề trong hệ thống giáo dục nghề nghiệp tại trường Trung cấp Á Châu . Môn học cung cấp các kiến thức, kỹ năng về cơ sở dữ liệu cho người học có thể thiết kế một cơ sở dữ liệu quan hệ, cách sử dụng các phép toán đại số quan hệ để tạo, cập nhật và truy vấn cơ sở dữ liệu và khái niệm phụ thuộc hàm ứng dụng trong lý thuyết thiết kế và chuẩn hóa cơ sở dữ liệu quan hệ.

4. Mục tiêu của môn học/mô đun:

+Về kiến thức:

- Trình bày được các khái niệm, các thuật ngữ trong cơ sở dữ liệu.
- Trình bày được các dạng chuẩn và chuẩn hóa được bài toán cơ sở dữ liệu trước khi cài đặt.

+Về kỹ năng:

- Vận dụng được các mô hình dữ liệu của cơ sở dữ liệu quan hệ vào việc thiết kế cơ sở dữ liệu cho bài toán cụ thể.
- Sử dụng ngôn ngữ truy vấn dữ liệu quan hệ thành thạo.

+Về năng lực tự chủ và trách nhiệm:

- Bố trí làm việc khoa học đảm bảo an toàn cho người và phương tiện học tập.
- Có ý thức trách nhiệm, tự giác và chủ động học tập.

5. Nội dung của môn học

5.1 Chương trình khung

Mã MH/ MĐ	Tên môn học/mô đun	Số tín chỉ	Thời gian học tập (giờ)			
			Tổng số	Trong đó		
				Lý thuyết	Thực hành/ thực tập	Thi/kiể m tra
1	Các môn học chung	12	255	94	148	13
MH1	Giáo dục Chính trị	2	30	15	13	2
MH2	Giáo dục Pháp luật	1	15	9	5	1
MH3	Giáo dục thể chất	1	30	4	24	2
MH4	Giáo dục quốc phòng và an ninh	2	45	21	21	3
MH5	Tin học	2	45	15	29	1
MH6	Ngoại ngữ	4	90	30	56	4
2	Các môn học, mô đun chuyên môn	62	1485	444	1011	60
MH07	An toàn vệ sinh công nghiệp	2	30	24	4	2
MĐ08	Tin học VP	4	90	30	56	4
MH09	Anh văn chuyên ngành	2	45	30	12	3
MH10	Kiến trúc máy tính	2	45	15	28	2
MH11	Nguyên lý hệ điều hành	2	45	15	28	2
MH12	Lập trình căn bản	3	60	30	26	4
MH13	Cơ sở dữ liệu	3	75	45	27	3
MH14	Mạng máy tính	4	90	30	56	4
MĐ15	Lắp ráp và cài đặt máy tính	5	120	30	85	5
MĐ16	Autocad	3	60	15	42	3
MĐ17	Vẽ đồ họa	2	45	15	28	2
MĐ18	Quản trị mạng	5	120	30	85	5
MĐ19	Quản trị CSDL	3	60	15	42	3
MĐ20	Hệ điều hành Linux	3	60	15	42	3

MĐ21	Lập trình trực quan	3	60	15	42	3
MĐ22	Thiết kế web	4	90	30	56	4
MĐ23	Thiết kế và xây dựng mạng Lan	4	90	30	56	4
MĐ24	Quản trị hệ thống WebServer và MailServer	4	90	30	56	4
MĐ25	Thực tập tốt nghiệp	5	240		240	
Tổng cộng:		74	1740			

5.2 Chương trình chi tiết môn học

Số TT	Tên chương mục	Thời gian			
		Tổng số	Lý thuyết	Thực hành	Kiểm tra* (LT hoặc TH)
				Bài tập	
I	Tổng quan về cơ sở dữ liệu	4	3	1	
	- Dữ liệu – thông tin.				
	- Các hệ thống xử lý truyền thống.				
	- Phương pháp cơ sở dữ liệu.				
	- Phân loại người dùng CSDL.				
II	Mô hình dữ liệu quan hệ	15	6	9	
	- Các khái niệm cơ bản				
	- Chuyển mô hình thực thể kết hợp sang mô hình dữ liệu quan hệ				
	- Ngôn ngữ đại số quan hệ				
III	Ngôn ngữ truy vấn dữ liệu SQL	15	7	8	*

	Tìm thông tin từ các cột của bản – Mệnh đề Select				
	Chọn các dòng của bản – Mệnh đề Where				
	Sắp xếp các dòng của bảng – Mệnh đề Order by				
	Câu lệnh truy vấn lồng nhau				
	Gom nhóm dữ liệu – mệnh đề Group by				
IV	Ràng buộc toàn vẹn	7	4	3	
	Ràng buộc toàn vẹn				
	Phân loại ràng buộc toàn vẹn				
VI	Lý thuyết thiết kế cơ sở dữ liệu	19	10	9	*
	Các vấn đề gặp phải khi tổ chức dữ liệu				
	Các phụ thuộc hàm				
	Bao đóng của tập phụ thuộc hàm và bao đóng của tập thuộc tính				
	Khóa của lược đồ quan hệ - Một số thuật toán tìm khóa				
	Phủ tối thiểu				
	Dạng chuẩn của lược đồ quan hệ				
	Cộng	60	30	30	

6. Điều kiện thực hiện mô đun:

6.1. Phòng học Lý thuyết/Thực hành: Đáp ứng phòng học chuẩn.

6.2. Trang thiết bị dạy học: Projector, máy vi tính, bảng, phấn.

6.3. Học liệu, dụng cụ, mô hình, phương tiện:

- Viết bảng, bảng trắng
- Các hình vẽ, ví dụ minh họa.

- Bút viết bảng, phấn, giấy vẽ, bút lông.
- Bài giảng soạn trên Slide, tài liệu phát cho học sinh (nội dung bài học và các bài tập).
- Bài giảng soạn trên phần mềm dạy học
- Phần mềm kiểm tra trắc nghiệm
- Tài liệu từ Internet, tài liệu tham khảo, các tài liệu thiết kế cơ sở dữ liệu.

6.4. Các điều kiện khác:

- Phòng học lý thuyết và phòng thực hành đủ điều kiện để thực hiện môn học.
- Phòng thực hành có đầy đủ máy tính cho học sinh thực hành.

7. Nội dung và phương pháp đánh giá:

7.1. Nội dung:

- Kiến thức:

- + Trình bày được các khái niệm về các thực thể, bộ, quan hệ, khóa, phụ thuộc hàm,..
- + Trình bày được các mô hình cơ sở dữ liệu quan hệ.
- + Trình bày được cú pháp của ngôn ngữ SQL
- + Trình bày được các dạng chuẩn của lược đồ quan hệ.

- Kỹ năng:

- + Phân tích dữ liệu và vẽ được các mô hình dữ liệu thực thể - kết hợp (mô hình E-R); chuyển đổi E-R sang lược đồ quan hệ. Xác định được các khóa, chuẩn hóa lược đồ ở mức tốt nhất có thể.
- + Sử dụng thành thạo ngôn ngữ truy vấn dữ liệu SQL chuẩn cho việc truy vấn dữ liệu đã cài đặt.

- Năng lực tự chủ và trách nhiệm:

- + Có thái độ nghiêm túc trong thực hành
- + Cẩn thận, thao tác nhanh chuẩn xác, tự giác trong học tập.

7.2. Phương pháp:

Người học được đánh giá tích lũy mô đun như sau:

7.2.1. Cách đánh giá

- Áp dụng quy chế đào tạo Cao đẳng hệ chính quy ban hành kèm theo Thông tư số 09/2017/TT-LĐTBXH, ngày 13/3/2017 của Bộ trưởng Bộ Lao động – Thương binh và Xã hội.
- Hướng dẫn thực hiện quy chế đào tạo áp dụng tại Trường Cao đẳng nghề Tây Ninh như sau:

Điểm đánh giá	Trọng số
+ Điểm kiểm tra thường xuyên (Hệ số 1)	40%
+ Điểm kiểm tra định kỳ (Hệ số 2)	

+ Điểm thi kết thúc mô đun

60%

7.2.2. Phương pháp đánh giá

Phương pháp đánh giá	Phương pháp tổ chức	Hình thức kiểm tra	Chuẩn đầu ra đánh giá	Số cột	Thời điểm kiểm tra
Thường xuyên	Viết/ Thuyết trình	Tự luận/ Trắc nghiệm/ Báo cáo	A1, A2 B1, B2 C1, C2	1	Sau 14 giờ.
Định kỳ	Viết/ Thuyết trình/ Thực hành	Tự luận/ Trắc nghiệm/ Báo cáo/ Thực hành	A3,B3, A4, B4 A5, B5 A6,B6	1 1	Sau 28 giờ Sau 43 giờ
Kết thúc mô đun	Viết	Trắc nghiệm/ Thực hành	A1, A2, A3, A4, A5, A6, A7, A8 B1, B2, B3, B4, B5, B6, B7, B8 C1, C2, C3.	1	Sau 74 giờ

7.2.3. Cách tính điểm

- Điểm đánh giá thành phần và điểm thi kết thúc mô đun được chấm theo thang điểm 10 (từ 0 đến 10), làm tròn đến một chữ số thập phân.

- Điểm mô đun là tổng điểm của tất cả điểm đánh giá thành phần của mô đun nhân với trọng số tương ứng. Điểm mô đun theo thang điểm 10 làm tròn đến một chữ số thập phân, sau đó được quy đổi sang điểm chữ và điểm số theo thang điểm 4 theo quy định của Bộ Lao động Thương binh và Xã hội về đào tạo theo tín chỉ.

8. Hướng dẫn thực hiện mô đun

8.1. Phạm vi, đối tượng áp dụng: Chương trình môn học được sử dụng để giảng dạy cho trình độ Trung cấp ngành Quản trị mạng máy tính.

8.2. Phương pháp giảng dạy, học tập mô đun

8.2.1. Đối với người dạy

- Giáo viên cần giới hạn phạm vi giảng dạy theo khung chương trình để học sinh định hướng được vấn đề cốt lõi khi thiết kế cơ sở dữ liệu quan hệ.

8.2.2. Đối với người học: Người học phải thực hiện các nhiệm vụ như sau:

- Học sinh trao đổi với nhau, thực hiện các bài thực hành và trình bày theo nhóm.

- Tham dự tối thiểu 70% các buổi giảng lý thuyết. Nếu người học vắng >30% số tiết lý thuyết phải học lại mô đun mới được tham dự kì thi lần sau.

- Tự học và thảo luận nhóm: là một phương pháp học tập kết hợp giữa làm việc theo nhóm và làm việc cá nhân. Một nhóm gồm 8-10 người học sẽ được cung cấp chủ đề thảo luận trước khi học lý thuyết, thực hành. Mỗi người học sẽ chịu trách nhiệm về 1 hoặc một số nội dung trong chủ đề mà nhóm đã phân công để phát triển và hoàn thiện tốt nhất toàn bộ chủ đề thảo luận của nhóm.

- Tham dự đủ các bài kiểm tra thường xuyên, định kỳ.

- Tham dự thi kết thúc môn học.

- Chủ động tổ chức thực hiện giờ tự học.

9. Tài liệu tham khảo

[1]. Lê Tiến Vương, nhập môn cơ sở dữ liệu quan hệ, 2000.

[2]. Vũ Đức Thi, Cơ sở dữ liệu kiến thức và thực hành, Nhà xuất bản thống kê 1997.

[3]. Nguyễn An Tế, Giáo trình nhập môn cơ sở dữ liệu, ĐHKHTN- ĐHQGTPHCM 1996.

[4]. Đỗ Trung Tuấn, Cơ sở dữ liệu, Nhà xuất bản giáo dục 1998.

CHƯƠNG 1: TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU

❖ GIỚI THIỆU CHƯƠNG 1

Chương này sẽ trình bày các khái niệm cơ bản của một hệ Cơ sở dữ liệu (CSDL) nhằm cung cấp cho các học viên những kiến thức cơ sở, một cái nhìn ban đầu về một cơ sở dữ liệu và một hệ quản trị CSDL. Đây là mốc xuất phát, nêu ra các vấn đề sẽ được trình bày cách giải quyết cụ thể trong các chương sau.

MỤC TIÊU CHƯƠNG 1

Sau khi học xong bài này, người học có khả năng:

- ***Về kiến thức:***
 - Trình bày sơ lược các khái niệm về cơ sở dữ liệu, các mô hình dữ liệu;
 - Trình bày chi tiết mô hình thực thể kết hợp (ERD), có thể
- ***Về kỹ năng:***
 - Phân tích dữ liệu và thiết kế được mô hình thực thể kết hợp.
 - Thực hiện các thao tác an toàn với máy tính
- ***Về năng lực tự chủ và trách nhiệm:***
 - Rèn luyện tính chính xác, sáng tạo, linh hoạt.
 - Tuân thủ nội quy, quy định nơi làm việc.

❖ PHƯƠNG PHÁP GIẢNG DẠY VÀ HỌC TẬP CHƯƠNG 1

- Đối với người dạy: sử dụng phương pháp giảng dạy tích cực (diễn giảng, vấn đáp, dạy học theo vấn đề); yêu cầu người học thực hiện câu hỏi thảo luận và bài tập bài 1 (cá nhân hoặc nhóm).
- Đối với người học: chủ động đọc trước giáo trình (Chương 1) trước buổi học; hoàn thành đầy đủ câu hỏi thảo luận và bài tập tình huống chương 1 theo cá nhân hoặc nhóm và nộp lại cho người dạy đúng thời gian quy định.

❖ ĐIỀU KIỆN THỰC HIỆN CHƯƠNG 1

- ***Phòng học chuyên môn hóa/nhà xưởng:*** Phòng máy tính
- ***Trang thiết bị máy móc:*** Máy chiếu và các thiết bị dạy học khác
- ***Học liệu, dụng cụ, nguyên vật liệu:*** Bài giảng soạn trên slide môn học, giáo trình, tài liệu tham khảo, giáo án và các tài liệu liên quan.
- ***Các điều kiện khác:*** Không có

❖ KIỂM TRA VÀ ĐÁNH GIÁ CHƯƠNG 1

- **Nội dung:**

- ✓ *Kiến thức: Kiểm tra và đánh giá tất cả nội dung đã nêu trong mục tiêu kiến thức*
- ✓ *Kỹ năng: Đánh giá tất cả nội dung đã nêu trong mục tiêu kỹ năng.*
- ✓ *Năng lực tự chủ và trách nhiệm: Trong quá trình học tập, người học cần:*
 - + *Nghiên cứu bài trước khi đến lớp*
 - + *Chuẩn bị đầy đủ tài liệu học tập.*
 - + *Tham gia đầy đủ thời lượng môn học.*
 - + *Nghiêm túc trong quá trình học tập.*
- **Phương pháp:**
- ✓ **Điểm kiểm tra thường xuyên:** 1 điểm kiểm tra (hình thức: vấn đáp)
- ✓ **Kiểm tra định kỳ lý thuyết:** không có

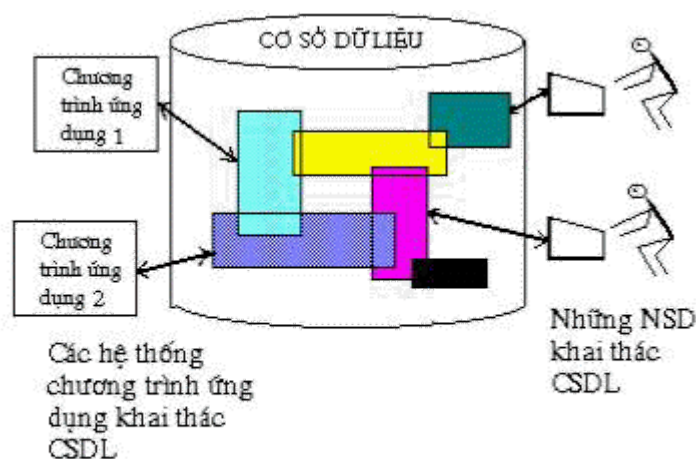
NỘI DUNG CHƯƠNG 1

TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU

1. Một số khái niệm cơ bản

1.1. Định nghĩa cơ sở dữ liệu

Cơ sở dữ liệu là một hệ thống các thông tin có cấu trúc được lưu trữ trên các thiết bị lưu trữ thông tin thứ cấp (như băng từ, đĩa từ ...) để có thể thỏa mãn yêu cầu khai thác thông tin đồng thời của nhiều người sử dụng hay nhiều chương trình ứng dụng với nhiều mục đích khác nhau.



Hình 1.2.1 Sơ đồ tổng quát về một cơ sở dữ liệu

Trong định nghĩa này cần nhấn mạnh những khía cạnh của định nghĩa. Trước hết, CSDL phải là một tập hợp các thông tin mang tính hệ thống chứ không phải là các thông tin rời rạc, không có mối quan hệ với nhau. Các thông tin này phải có cấu trúc và tập hợp các

thông tin này phải có khả năng đáp ứng các nhu cầu khai thác của nhiều người sử dụng một cách đồng thời. Đó cũng chính là các đặc trưng của CSDL.

1.2. Ưu điểm của cơ sở dữ liệu

Giảm sự trùng lặp thông tin xuống mức thấp nhất và do đó bảo đảm được tính nhất quán và toàn vẹn dữ liệu.

Đảm bảo dữ liệu có thể được truy xuất theo nhiều cách khác nhau.

Khả năng chia sẻ thông tin cho nhiều người sử dụng và nhiều ứng dụng khác nhau.

1.3. Những vấn đề mà CSDL cần phải giải quyết

Tuy nhiên, để đạt được các ưu điểm trên, CSDL đặt ra những vấn đề cần phải giải quyết. Đó là:

1- Tính chủ quyền của dữ liệu. Do tính chia sẻ của CSDL nên tính chủ quyền của dữ liệu có thể bị lu mờ và làm mờ nhạt tinh thần trách nhiệm, được thể hiện trên vấn đề an toàn dữ liệu, khả năng biểu diễn các mối liên hệ ngữ nghĩa của dữ liệu, và tính chính xác của dữ liệu. Điều này có nghĩa là người khai thác CSDL phải có nghĩa vụ cập nhật các thông tin mới nhất của CSDL.

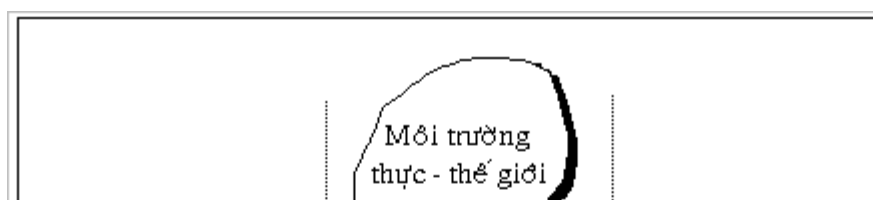
2- Tính bảo mật và quyền khai thác thông tin của người sử dụng. Do có nhiều người được phép khai thác CSDL một cách đồng thời nên cần phải có một cơ chế bảo mật và phân quyền hạn khai thác CSDL. Các hệ điều hành nhiều người sử dụng hay hệ điều hành mạng cục bộ (Novell Netware, Windows For WorkGroup, WinNT, ...) đều có cung cấp cơ chế này.

3- Tranh chấp dữ liệu. Nhiều người được phép truy nhập vào cùng một tài nguyên dữ liệu (Data Source) của CSDL với những mục đích khác nhau: Xem, thêm, xóa hoặc sửa dữ liệu. Cần phải có một cơ chế ưu tiên truy nhập dữ liệu cũng như cơ chế giải quyết tình trạng khóa chết (DeadLock) trong quá trình khai thác cạnh tranh. Cơ chế ưu tiên có thể được thực hiện bằng việc cấp quyền (hay mức độ) ưu tiên cho từng người khai thác - người nào được cấp quyền hạn ưu tiên cao hơn thì được ưu tiên truy nhập dữ liệu trước; theo biến có hoặc loại truy nhập - quyền đọc được ưu tiên trước quyền ghi dữ liệu; dựa trên thời điểm truy nhập - ai có yêu cầu truy xuất trước thì có quyền truy nhập dữ liệu trước; hoặc theo cơ chế lập lịch truy xuất hay các cơ chế khóa...

4- Đảm bảo dữ liệu khi có sự cố. Việc quản lý dữ liệu tập trung có thể làm tăng khả năng mất mát hoặc sai lệch thông tin khi có sự cố như mất điện đột xuất, một phần đĩa lưu trữ CSDL bị hư v.v... Một số hệ điều hành mạng có cung cấp dịch vụ sao lưu ảnh đĩa cứng (cơ chế sử dụng đĩa cứng dự phòng - RAID), tự động kiểm tra và khắc phục lỗi khi có sự cố, tuy nhiên, bên cạnh dịch vụ của hệ điều hành, để đảm bảo CSDL luôn luôn ổn định, một CSDL nhất thiết phải có một cơ chế khôi phục dữ liệu khi các sự cố bất ngờ xảy ra.

Các mức biểu diễn một CSDL

Theo kiến trúc ANSI-PARC, một CSDL có 3 mức biểu diễn: Mức trong (còn gọi là mức vật lý - Physical), mức quan niệm (Conception hay Logical) và mức ngoài.



a) Mức trong

Đây là mức lưu trữ CSDL. Tại mức này, vấn đề cần giải quyết là, dữ liệu gì và được lưu trữ như thế nào? ở đâu (đĩa từ, băng từ, track, sector ... nào)? Cần các chỉ mục gì? Việc truy xuất là tuần tự (Sequential Access) hay ngẫu nhiên (Random Access) đối với từng loại dữ liệu.

Những người hiểu và làm việc với CSDL tại mức này là người quản trị CSDL (Administrator), những người sử dụng (NSD) chuyên môn.

b) Mức quan niệm:

Tại mức này sẽ giải quyết cho câu hỏi CSDL cần phải lưu giữ bao nhiêu loại dữ liệu? đó là những dữ liệu gì? Mỗi quan hệ giữa các loại dữ liệu này như thế nào?

Từ thế giới thực (Real Universe) các chuyên viên tin học qua quá trình khảo sát và phân tích, cùng với những người sẽ đảm nhận vai trò quản trị CSDL, sẽ xác định được những loại thông tin gì được cho là cần thiết phải đưa vào CSDL, đồng thời mô tả rõ mối liên hệ giữa các thông tin này. Có thể nói cách khác, CSDL mức quan niệm là một sự biểu diễn trừu tượng CSDL mức vật lý; hoặc ngược lại, CSDL vật lý là sự cài đặt cụ thể của CSDL mức quan niệm.

Ví dụ 1.2:

Người ta muốn xây dựng một hệ quản trị CSDL để quản lý các nhân viên của một công ty. Môi trường (thế giới thực) của công ty ở đây gồm có các phòng ban (Department) - mỗi phòng ban có một tên gọi khác nhau, một địa chỉ trụ sở chính (Location), các số điện thoại (Telephone) để liên lạc, có một người làm trưởng phòng ban, hàng năm được cấp một khoản kinh phí để hoạt động (Expense Budget), và phải đạt một doanh thu (Revenue Budget). Để tránh viết tên phòng ban dài dễ dẫn đến viết sai, người ta thường đặt cho mỗi phòng ban một giá trị số (gọi là số hiệu phòng ban - Department Number) và sử dụng số hiệu này để xác định tên và các thông tin khác của nó.

Công ty có một số công việc có thể sắp xếp cho các nhân viên trong công ty. Để thuận lợi cho việc theo dõi công việc cũng như trong công tác tuyển chọn nhân viên mới, người ta lập thành một bảng các công việc (JOBS) gồm các thông tin: tên tắt công việc (Job), tên công việc (Job Name), mức lương tối thiểu (Min Salary) và tối đa (Max Salary) của công việc này và cho biết công việc này cần có người lãnh đạo không. Một công việc có thể có nhiều người cùng làm.

Mỗi phòng ban có thể có từ 1 đến nhiều nhân viên (Employee). Mỗi nhân viên có một tên gọi, một công việc làm (Job), một khoản tiền lương hàng tháng (Salary), số hiệu phòng ban mà anh ta đang công tác. Nếu muốn, người ta có thể theo dõi thêm các thông tin khác như ngày sinh (Birth Day), giới tính (Sex) v.v... Để tránh viết tên nhân viên dài dễ dẫn đến sai sót, mỗi nhân viên có thể được gán cho một con số duy nhất, gọi là mã số nhân viên (EmpNo).

Nếu yêu cầu quản lý của công ty chỉ dừng ở việc theo dõi danh sách nhân viên trong từng phòng ban cùng các công việc của công ty thì cần 3 loại thông tin: Phòng ban (DEPARTMENT), Công việc (JOBS) và Nhân viên (EMPLOYEE) với các thông tin như trên là đủ. Có thể công ty có thêm yêu cầu quản lý cả quá trình tuyển dụng và nâng lương

thì cần có thêm một (hoặc một số) loại thông tin về quá trình: Mã số nhân viên, lần thay đổi, thời gian bắt đầu và kết thúc sự thay đổi, mức lương,

Từ môi trường thế giới thực, xuất phát từ nhu cầu quản lý, việc xác định các loại thông tin cần lưu trữ và các mối quan hệ giữa các thông tin đó như thế nào ... đó chính là công việc ở mức quan niệm.

c) Mức ngoài

Đó là mức của người sử dụng và các chương trình ứng dụng. Làm việc tại mức này có các nhà chuyên môn, các kỹ sư tin học và những người sử dụng không chuyên.

Mỗi người sử dụng hay mỗi chương trình ứng dụng có thể được "nhìn" (View) CSDL theo một góc độ khác nhau. Có thể "nhìn" thấy toàn bộ hay chỉ một phần hoặc chỉ là các thông tin tổng hợp từ CSDL hiện có. Người sử dụng hay chương trình ứng dụng có thể hoàn toàn không được biết về cấu trúc tổ chức lưu trữ thông tin trong CSDL, thậm chí ngay cả tên gọi của các loại dữ liệu hay tên gọi của các thuộc tính. Họ chỉ có thể làm việc trên một phần CSDL theo cách "nhìn" do người quản trị hay chương trình ứng dụng quy định, gọi là khung nhìn (View).

Ví dụ

Cũng ví dụ trên, Phòng Tổ chức nhân sự giờ đây còn quản lý thêm cả các thông tin chi tiết trong lý lịch của nhân viên trong công ty: quá trình đào tạo chuyên môn kỹ thuật - kinh tế - chính trị - quản lý Nhà nước, quá trình được khen thưởng, các lần bị kỷ luật, quá trình hoạt động Cách mạng bị địch bắt - bị tù đầy, quá trình công tác, quá trình nâng lương, sơ lược tiểu sử cha mẹ - anh chị em ruột - vợ chồng - con v.v... Rõ ràng rằng, Phòng Kế toán có thể chỉ được nhìn thấy CSDL là danh sách nhân viên đang làm các công việc cụ thể trong từng Phòng ban với các mức lương thỏa thuận, mà không được thấy lý lịch của các nhân viên. Lãnh đạo công ty có thể chỉ cần "nhìn" thấy số lượng nhân viên, tổng số lương phải trả và ai là người lãnh đạo của từng Phòng ban. Trong khi đó ngay cả những người trong Phòng Tổ chức nhân sự cũng có thể có người được xem lý lịch của tất cả cán bộ, công nhân viên của công ty, nhưng cũng có thể có người chỉ được xem lý lịch của những cán bộ, công nhân viên với mức lương từ xx đồng trở xuống...

Như vậy, cấu trúc CSDL vật lý (mức trong) và mức quan niệm thì chỉ có một; nhưng tại mức ngoài, mức của các chương trình ứng dụng và người sử dụng trực tiếp CSDL, thì có thể có rất nhiều cấu trúc ngoài tương ứng.

1.4. các đối tượng sử dụng CSDL

Những người sử dụng CSDL không chuyên về lĩnh vực tin học và CSDL, do đó CSDL cần có các công cụ để cho những người sử dụng không chuyên có thể sử dụng để khai thác CSDL khi cần thiết.

Các chuyên viên tin học biết khai thác CSDL. Những người này có thể xây dựng các ứng dụng khác nhau phục vụ cho các mục đích khác nhau trên CSDL.

Những người quản trị CSDL, đó là những người hiểu biết về tin học, về các hệ quản trị CSDL và hệ thống máy tính. Họ là người tổ chức CSDL (khai báo cấu trúc CSDL, ghi nhận các yêu cầu bảo mật cho các dữ liệu cần bảo vệ ...) do đó họ phải nắm rõ các vấn đề kỹ thuật về CSDL để có thể phục hồi dữ liệu khi có sự cố. Họ là những người cấp quyền hạn khai thác CSDL, do vậy họ có thể giải quyết được các vấn đề tranh chấp dữ liệu, nếu có.

1.5. Hệ quản trị CSDL (Data Base Management System)

Để giải quyết tốt tất cả các vấn đề đặt ra cho một CSDL như đã nêu trên: tính chủ quyền, cơ chế bảo mật hay phân quyền hạn khai thác CSDL, giải quyết tranh chấp trong quá trình truy nhập dữ liệu, và phục hồi dữ liệu khi có sự cố ... thì cần phải có một hệ thống các phần mềm chuyên dụng. Hệ thống các phần mềm đó được gọi là hệ quản trị CSDL (tiếng Anh là DataBase Management System - DBMS). Đó là các công cụ hỗ trợ tích cực cho các nhà phân tích & thiết kế CSDL và những người khai thác CSDL. Cho đến nay có khá nhiều hệ quản trị CSDL mạnh được đưa ra thị trường như: Visual FoxPro, MicroSoft Access, SQL-Server, DB2, Sybase, Paradox, Informix, Oracle... với các chất lượng khác nhau.

Mỗi hệ quản trị CSDL đều được cài đặt dựa trên một mô hình dữ liệu cụ thể. Hầu hết các hệ quản trị CSDL hiện nay đều dựa trên mô hình quan hệ (Xem chương II). Dù dựa trên mô hình dữ liệu nào, một hệ quản trị CSDL cũng phải có:

1) Ngôn ngữ giao tiếp giữa người sử dụng (NSD) và CSDL, bao gồm:

Ngôn ngữ mô tả dữ liệu (Data Definition Language - DDL) để cho phép khai báo cấu trúc của CSDL, khai báo các mối liên hệ của dữ liệu (Data RelationShip) và các quy tắc (Rules, Constraint) quản lý áp đặt lên các dữ liệu đó.

Ngôn ngữ thao tác dữ liệu (Data Manipulation Language - DML) cho phép người sử dụng có thể thêm (Insert), xóa (Delete), sửa (Update) dữ liệu trong CSDL.

Ngôn ngữ truy vấn dữ liệu, hay ngôn ngữ hỏi đáp có cấu trúc (Structured Query Language - SQL) cho phép những người khai thác CSDL (chuyên nghiệp hoặc không chuyên) sử dụng để truy vấn các thông tin cần thiết trong CSDL.

Ngôn ngữ quản lý dữ liệu (Data Control Language - DCL) cho phép những người quản trị hệ thống thay đổi cấu trúc của các bảng dữ liệu, khai báo bảo mật thông tin và cấp quyền hạn khai thác CSDL cho người sử dụng.

2) Từ điển dữ liệu (Data Dictionary) dùng để mô tả các ánh xạ liên kết, ghi nhận các thành phần cấu trúc của CSDL, các chương trình ứng dụng, mật mã, quyền hạn sử dụng v.v....

3) Có biện pháp bảo mật tốt khi có yêu cầu bảo mật.

4) Cơ chế giải quyết vấn đề tranh chấp dữ liệu. Mỗi hệ quản trị CSDL cũng có thể cài đặt một cơ chế riêng để giải quyết các vấn đề này. Một số biện pháp sau đây được sử dụng:

5) Cấp quyền ưu tiên cho từng người sử dụng (người quản trị CSDL thực hiện).

Đánh dấu yêu cầu truy xuất dữ liệu, phân chia thời gian, người nào có yêu cầu trước thì có quyền truy xuất dữ liệu trước.

6) Hệ quản trị CSDL cũng phải có cơ chế sao lưu (Backup) và phục hồi (Restore) dữ liệu khi có sự cố xảy ra. Điều này có thể được thực hiện bằng cách:

Định kỳ kiểm tra CSDL, sau một thời gian nhất định hệ quản trị CSDL sẽ tự động tạo ra một bản sao CSDL. Cách này hơi tốn kém, nhất là đối với các CSDL lớn.

Tạo nhật ký (LOG) thao tác CSDL. Mỗi thao tác trên CSDL đều được hệ thống ghi lại, khi có sự cố xảy ra thì tự động lần ngược lại (RollBack) để phục hồi CSDL.

7) Hệ quản trị CSDL phải cung cấp một giao diện (Interface) tốt, dễ sử dụng, dễ hiểu cho những người sử dụng không chuyên.

8) Ngoài ra, một hệ quản trị CSDL phải đáp ứng được một yêu cầu rất quan trọng, đó là bảo đảm tính độc lập giữa dữ liệu và chương trình: Khi có sự thay đổi dữ liệu (như sửa đổi cấu trúc lưu trữ các bảng dữ liệu, thêm các chỉ mục (Index) ...) thì các chương trình ứng dụng (Application) đang chạy trên CSDL đó vẫn không cần phải được viết lại, hay cũng không làm ảnh hưởng đến những NSD khác.

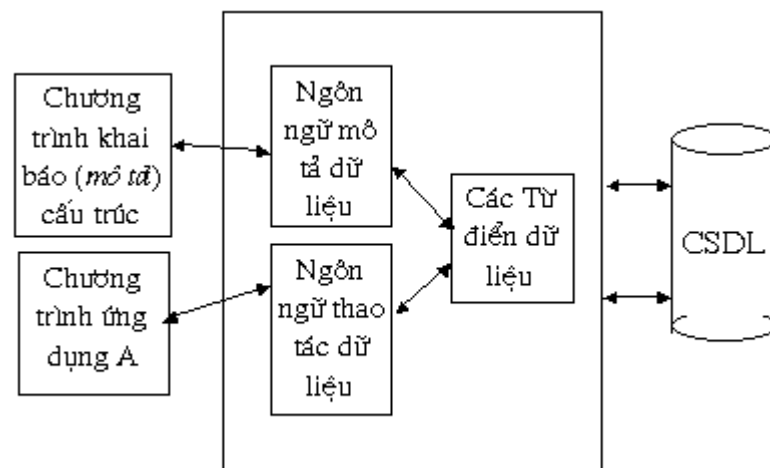
Vài nét về quá trình phát triển các hệ quản trị CSDL:

Trải qua gần 40 năm nghiên cứu và cài đặt ứng dụng, các hệ quản trị CSDL không ngừng được phát triển. Các hệ quản trị CSDL đầu tiên ra đời vào đầu những năm 60 của thế kỷ 20 dựa trên mô hình dữ liệu phân cấp và mạng, trong số đó có hệ quản trị CSDL có tên là IMS của hãng IBM dựa trên mô hình dữ liệu phân cấp.

Năm 1976, hệ quản trị CSDL đầu tiên dựa trên mô hình dữ liệu quan hệ của hãng IBM mang tên System-R ra đời. Từ năm 1980 hãng IBM cho ra đời hệ quản trị CSDL trên các máy Main Frame mang tên DB2, tiếp theo là các hệ quản trị CSDL Dbase, Sybase, Oracle, Informix, SQL-Server ...

Từ những năm 1990 người ta bắt đầu cố gắng xây dựng các hệ quản trị CSDL hướng đối tượng (Oriented Object DataBase Management System) như Orion, Illustra, Itasca, ... Tuy nhiên hầu hết các hệ này đều vẫn là quan hệ - hướng đối tượng, nghĩa là, xét về bản chất, chúng vẫn dựa trên nền tảng của mô hình quan hệ. Hệ quản trị CSDL hướng đối tượng thuần nhất có thể là hệ ODMG ra đời vào năm 1996.

Sơ đồ tổng quát của một hệ quản trị CSDL



Hình 1.6.1 Sơ đồ tổng quát của một hệ quản trị CSDL

Hình 1.6.1 minh họa sơ đồ tổng quát của một hệ quản trị CSDL. Chúng ta thấy có 3 mức: mức chương trình khai báo cấu trúc và chương trình ứng dụng; mức mô tả CSDL, thao tác CSDL và các từ điển dữ liệu; và mức CSDL.

Mỗi hệ quản trị CSDL có một ngôn ngữ khai báo (hay mô tả: Data Definition Language - DDL) cấu trúc CSDL riêng. Những người thiết kế và quản trị CSDL thực hiện các công việc khai báo cấu trúc CSDL.

Các chương trình khai báo cấu trúc CSDL được viết bằng ngôn ngữ mà hệ quản trị CSDL cho phép. Hai công việc khai báo là khai báo cấu trúc logic (đó là việc khai báo các loại dữ liệu và các mối liên hệ giữa các loại dữ liệu đó, cùng các ràng buộc toàn vẹn dữ liệu - RBTV) và khai báo vật lý (dữ liệu được lưu trữ theo dạng nào?, có bao nhiêu chỉ mục?).

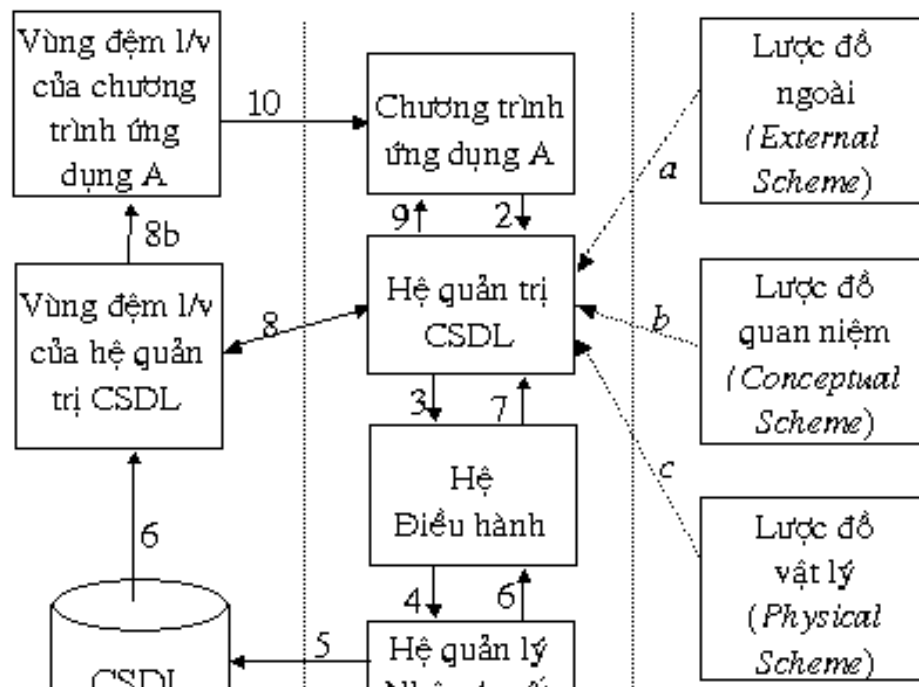
Các chương trình ứng dụng được viết bằng ngôn ngữ thao tác CSDL (Data Manipulation Language - DML) với mục đích:

- Truy xuất dữ liệu
- Cập nhật dữ liệu (thêm, xóa, sửa dữ liệu)
- Khai thác dữ liệu

Ngôn ngữ thao tác CSDL còn được sử dụng cho những NSD thao tác trực tiếp với CSDL.

Từ điển dữ liệu (Data Dictionary - DD) là một CSDL của hệ quản trị CSDL sử dụng để lưu trữ cấu trúc CSDL, các thông tin bảo mật, bảo đảm an toàn dữ liệu và các cấu trúc ngoài. Những người đã làm quen với hệ quản trị CSDL của MicroSoft Access có thể thấy các từ điển dữ liệu này thông qua các bảng (Table) có tên bắt đầu bằng chữ MSys như MSysACEs, MSysColumn, MSysIMEXColumn, MSysIMEXSpecs, MSysIndexes, MSysMacros, MSysObjects, MSysQueries, MSysRelationships ... Từ điển dữ liệu còn được gọi là Siêu CSDL (Meta-DataBase).

(*) Quá trình hoạt động của một chương trình ứng dụng thông qua các tầng của CSDL:



Hình 1.6.2. Quá trình hoạt động của một chương trình ứng dụng thông qua các tầng của CSDL.

Hình 1.6.2 cho chúng ta một cách nhìn về quá trình hoạt động của một chương trình ứng dụng thông qua các tầng của CSDL:

Các yêu cầu của chương trình ứng dụng được chuyển tới hệ quản trị CSDL (theo con đường số 2). Tại đây hệ quản trị CSDL sẽ tham khảo các từ điển dữ liệu (Meta DataBase) để tìm kiếm các ánh xạ cấu trúc ngoài với cấu trúc quan niệm và cấu trúc vật lý (các ngõ a, b và c). Tại đây hệ quản trị CSDL có thể sẽ tham khảo tới vùng đệm của nó để xác định xem câu trả lời đã có sẵn ở đó chưa, nếu có thì trả lại cho chương trình ứng dụng thông qua con đường số 9; ngược lại sẽ yêu cầu hệ điều hành truy xuất thông tin theo con đường số 3. Tới đây hệ điều hành sẽ gửi yêu cầu truy xuất thông tin trong CSDL thông qua hệ thống xuất nhập của HĐH (các con đường số 4 và 5). Nếu việc truy xuất không thành công nó sẽ trả lại yêu cầu về cho hệ quản trị CSDL (có thể thông qua các mã lỗi) qua con đường số 6; nếu thành công thì dữ liệu sẽ được chuyển vào vùng đệm của hệ quản trị CSDL. Qua xử lý, hệ quản trị CSDL sẽ chuyển dữ liệu vào vùng đệm của chương trình ứng dụng để nó xử lý (qua con đường 8a) và cho ra kết quả trả lời của chương trình ứng dụng qua con đường số 10.

Theo sơ đồ trên có thể nhận thấy các trục trặc có thể xảy ra tại các con đường (2a), (3), (4), (5), (6) và (8). Lỗi tại 2 con đường số (6) và (8) có thể là do tràn vùng làm việc.

1.6. Các ứng dụng của CSDL

Hiện nay, hầu như CSDL gắn liền với mọi ứng dụng của tin học; chẳng hạn như việc quản lý hệ thống thông tin trong các cơ quan nhà nước, việc lưu trữ và xử lý thông tin trong các doanh nghiệp, trong các lĩnh vực nghiên cứu khoa học, trong công tác giảng dạy, cũng như trong việc tổ chức thông tin đa phương tiện,...

2. Các mô hình dữ liệu

Mô hình dữ liệu là sự trừu tượng hoá môi trường thực. Mỗi loại mô hình dữ liệu đặc trưng cho một cách tiếp cận dữ liệu khác nhau của những nhà phân tích thiết kế CSDL. Mỗi loại mô hình dữ liệu đều có những ưu điểm và những mặt hạn chế của nó, nhưng vẫn có những mô hình dữ liệu nổi trội và được nhiều người quan tâm nghiên cứu.

Sau đây chúng ta sẽ điểm qua lịch sử phát triển của các mô hình dữ liệu.

Vào những năm sáu mươi, thế hệ đầu tiên của CSDL ra đời dưới dạng mô hình thực thể kết hợp, mô hình mạng và mô hình phân cấp.

Vào những năm bảy mươi, thế hệ thứ hai của CSDL ra đời. Đó là mô hình dữ liệu quan hệ do EF. Codd phát minh. Mô hình này có cấu trúc logic chặt chẽ. Đây là mô hình đã và đang được sử dụng rộng khắp trong công tác quản lý trên phạm vi toàn cầu. Việc nghiên cứu mô hình dữ liệu quan hệ nhằm vào lý thuyết chuẩn hoá các quan hệ và là một công cụ quan trọng trong việc phân tích thiết kế các hệ CSDL hiện nay. Mục đích của nghiên cứu này nhằm bỏ đi các phần tử không bình thường của quan hệ khi thực hiện các phép cập nhật, loại bỏ các phần tử dư thừa.

Sang thập kỷ tám mươi, mô hình CSDL thứ ba ra đời, đó là mô hình cơ sở dữ liệu hướng đối tượng, mô hình cơ sở dữ liệu phân tán, mô hình cơ sở dữ liệu suy diễn, ...

Trong phần tiếp theo sau đây, chúng tôi sẽ trình bày về mô hình dữ liệu tiêu biểu nhất để thiết kế (bước đầu) một ứng dụng tin học đó là mô hình thực thể kết hợp. Trong các chương còn lại của giáo trình này chúng tôi sẽ trình bày về mô hình dữ liệu quan hệ.

3. Mô hình thực thể kết hợp

Hiện nay mô hình dữ liệu quan hệ thường được dùng trong các hệ quản trị CSDL, đây là mô hình dữ liệu ở mức vật lý. Để thành lập được mô hình này, thường là phải dùng mô hình dữ liệu ở mức quan niệm để đặc tả, một trong những mô hình ở dạng đó là mô hình thực thể kết hợp (sau đó mới dùng một số quy tắc để chuyển hệ thống từ mô hình này về mô hình dữ liệu quan hệ – các quy tắc này sẽ được nói đến trong phần sau).

Mô hình dữ liệu thực thể - kết hợp (Entity - Relationship Model) do P.P.Chen đề xuất vào năm 1976. Sau đây là các khái niệm của mô hình thực thể kết hợp.

3.1. Thực thể

Thực thể (Entity): Là một thể hiện hoặc một đối tượng của một loại thực thể. Khái niệm này tương tự như khái niệm mẫu tin trong mô hình dữ liệu mạng và mô hình dữ liệu phân cấp. Thực thể là một sự vật tồn tại và phân biệt được, chẳng hạn sinh viên Nguyễn Văn Thành, lớp Cao Đẳng Tin Học 2A, môn học Cơ Sở Dữ Liệu, xe máy có biển số đăng ký 52-0549, ... là các ví dụ về thực thể.

3.2. Loại thực thể

Loại thực thể (Entity Type): Là một loại đối tượng cần quản lý trong CSDL, chẳng hạn, KHOA, LỚP-HỌC, MÔN-HỌC, GIẢNG-VIÊN, HỌC-VIÊN, tức là, cũng tương tự như khái niệm về loại mẫu tin trong mô hình mạng và mô hình phân cấp.

3.3. Thuộc tính

Thuộc tính của loại thực thể (Entity Attribute): Là các đặc tính riêng biệt cơ bản của loại thực thể, tương tự khái niệm thuộc tính (Attribute) trong mô hình dữ liệu quan hệ sẽ trình bày trong Chương II. Ví dụ, loại thực thể KHOA có các thuộc tính Mã-Khoa, Tên-Khoa. Loại thực thể LỚP-HỌC có một số thuộc tính Mã-Lớp, Tên-Lớp, Niên-Khóa, Số-Học-Viên. Loại thực thể MÔN-HỌC có một số thuộc tính Mã-Môn, Tên-Môn, Số-Đv-Học-Trình. Loại thực thể HỌC-VIÊN có một số thuộc tính Mã-Học-Viên, Tên-Học-Viên, Ngày-Sinh, Quê-Quán. Loại thực thể GIẢNG-VIÊN có một số thuộc tính Mã-Giảng-Viên, Tên-Giảng-Viên, Cấp-Học-Vị, Chuyên-Ngành. v.v...

3.4. Khóa

Khóa của loại thực thể (Entity Key): Đó là các thuộc tính nhận diện của loại thực thể. Căn cứ vào các giá trị của các thuộc tính nhận diện này người ta có thể xác định một thực thể duy nhất của một loại thực thể. Ví dụ, khóa của loại thực thể LỚP-HỌC có thể là Mã-Lớp; khóa của loại thực thể HỌC-VIÊN có thể là Mã-Học-Viên; khóa của loại thực thể MÔN-HỌC có thể là Mã-Môn-Học ... Khái niệm này cũng tương tự như khái niệm khóa (Key) trong mô hình dữ liệu quan hệ sẽ trình bày trong Chương II.

3.5. Mối kết hợp

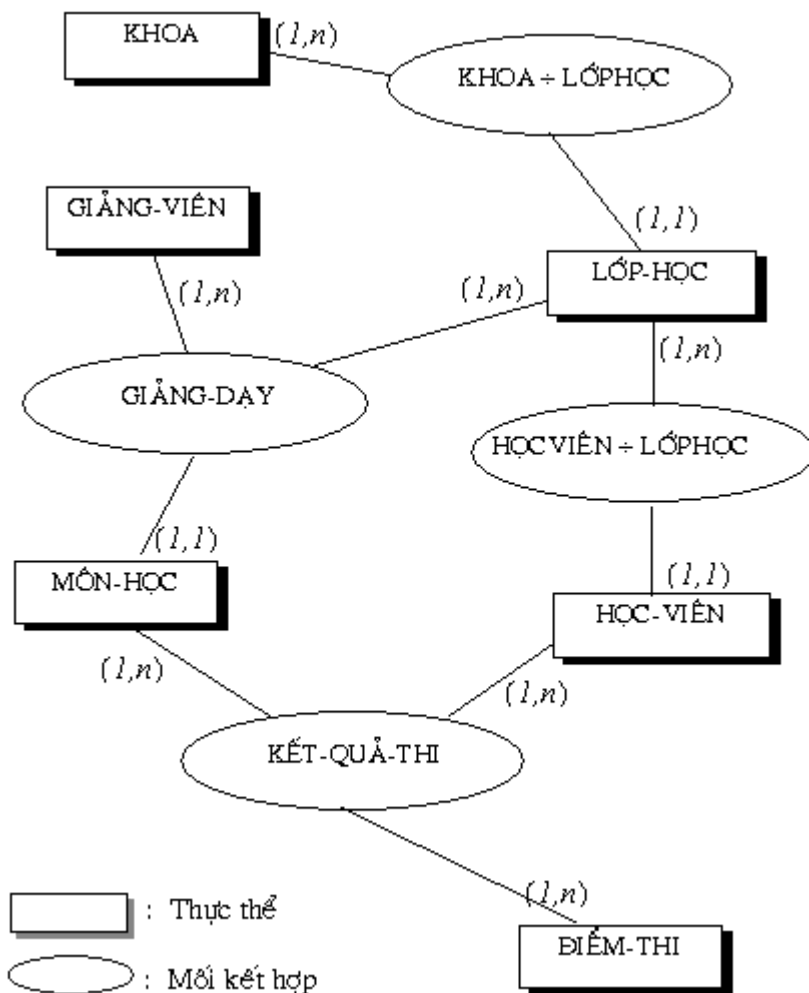
Loại mối kết hợp (Entity Relationship): Tương tự như loại mối liên hệ trong mô hình dữ liệu mạng. Trong đồ thị biểu diễn của mô hình này người ta cũng sử dụng hình elíp để thể hiện một mối kết hợp giữa các thực thể. Giữa 2 loại thực thể có thể tồn tại nhiều hơn một mối kết hợp.

Số ngôi của mối kết hợp (Relationship Degree): Là tổng số loại thực thể tham gia vào mối kết hợp. Ví dụ, giữa loại thực thể SINH-VIÊN và KHOA tồn tại mối kết hợp "trực thuộc" - đó là mối kết hợp 2 ngôi. KẾT-QUẢ-THI (hoặc KIỂM-TRA) của sinh viên là mối kết hợp giữa 3 thực thể SINH-VIÊN, MÔN-THI và ĐIỂM-THI - đó là mối kết hợp 3 ngôi.

Thuộc tính của mối kết hợp (Relationship Attribute): Mối kết hợp có thể có các thuộc tính của riêng nó. Thông thường mối kết hợp có các thuộc tính là khóa của các loại thực thể tham gia vào mối kết hợp, ngoài ra còn có thêm những thuộc tính bổ sung khác. Ví dụ, trong mối kết hợp 3 ngôi kể trên, thuộc tính của mối kết hợp này có thể bao gồm Mã-Học-Viên, Mã-Môn-Học, Điểm-Thi; và có thể có thêm các thuộc tính bổ sung khác như Lần-Thi-Thứ, Ngày-Thi, Ghi-Chú v.v...

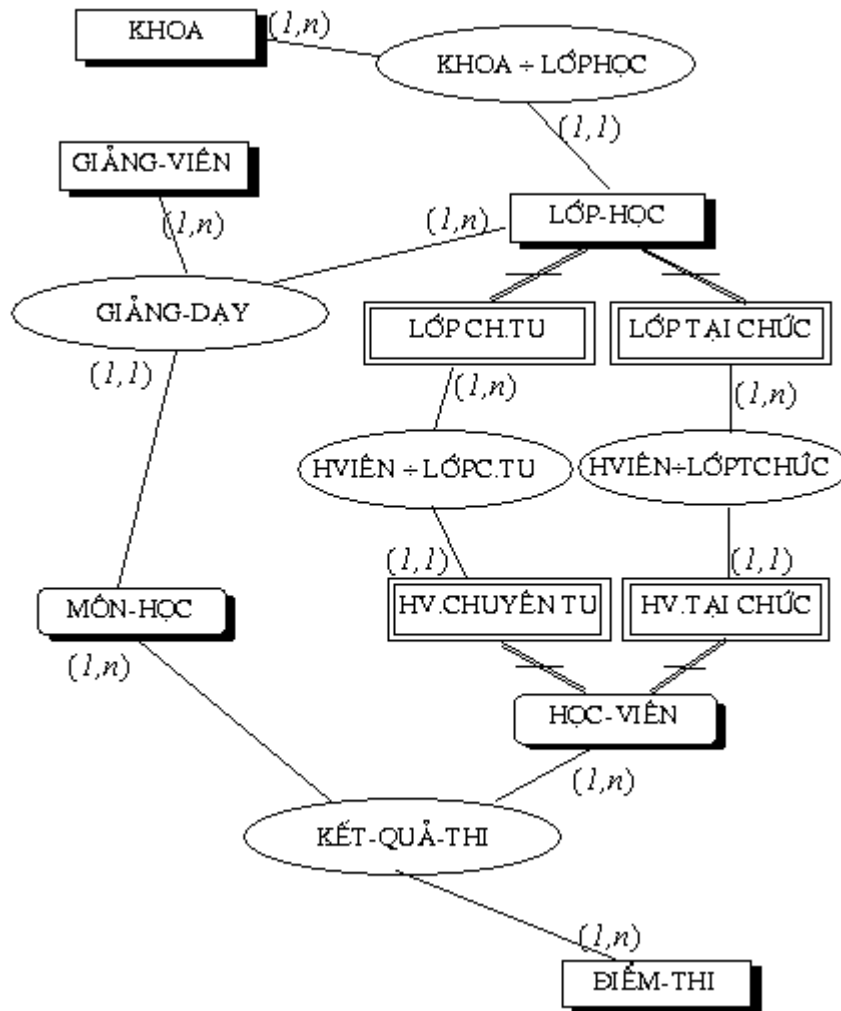
Bản số của mỗi nhánh của mối kết hợp (Relationship Cardinal): Mỗi nhánh (hay mỗi chiều) của mối kết hợp là mối kết hợp nối một loại thực thể với một mối kết hợp. Trong nhánh này cần xác định số lượng tối thiểu và số lượng tối đa các thực thể của nhánh đó sẽ tham gia vào một thực thể của mối kết hợp. Hai đại lượng này - đặt trong cặp dấu ngoặc tròn - được gọi là bản số của mối kết hợp. Ví dụ, trong mối kết hợp 3 ngôi nêu trên, tại nhánh nối loại thực thể HỌC-VIÊN với mối kết hợp KẾT-QUẢ-THI là (1,n), bởi vì sẽ có ít nhất một học viên tham gia kỳ thi và nhiều nhất là tất cả số học viên học môn đó cùng dự thi.

Vào khoảng năm 1980, mô hình dữ liệu thực thể-kết hợp đã được mở rộng thêm một số khái niệm mới như "loại thực thể chuyên biệt hóa" (Specialized Entity) và "mối kết hợp đệ quy" (Recursive Relationship). Mô hình này cùng với mô hình dữ liệu quan hệ và mô hình hướng đối tượng được sử dụng khá phổ biến trong việc thiết kế các CSDL hiện nay, bởi sự kết hợp này làm cho mô hình dữ liệu thể hiện được nhiều ngữ nghĩa của những loại dữ liệu trong CSDL hơn.



Hình 2.4.1 Mô hình dữ liệu thực thể - kết hợp của CSDL quản lý học viên.

Hình 2.4.1 trình bày một mô hình thực thể - kết hợp cho CSDL quản lý học viên gồm các loại thực thể KHOA, LỚP-HỌC, MÔN-HỌC, GIẢNG-VIÊN, HỌC-VIÊN. Mối kết hợp HỌCVIÊN, LỚP-HỌC giữa 2 loại thực thể HỌC-VIÊN và LỚP-HỌC. Tại nhánh HỌC-VIÊN bản số của nó là (1,1), điều này nhà phân tích và thiết kế đã khẳng định tình trạng thực tế là một học viên phải theo học (hay có tên trong danh sách) ít nhất là một lớp và cũng chỉ thuộc tối đa một lớp. Bản số của nhánh LỚP-HỌC là (1,n) nghĩa là một lớp (nếu đã có tên trong danh sách lớp học) thì có ít nhất 1 học viên và tối đa có thể là nhiều (n) học viên theo học.



Hình 2.4.2. Mô hình thực thể kết hợp với các thực thể chuyên biệt hóa.

Nếu yêu cầu quản lý đòi hỏi phải phân tích rạch ròi thành 2 loại lớp học chính quy và tại chức, và các lớp học tại chức có thể sẽ không phải học một số môn học hoặc số đơn vị học trình ít hơn so với các lớp chính quy, khi đó mô hình dữ liệu thực thể - kết hợp được thể hiện như trong sơ đồ 2.4.2. Thực thể chuyên biệt hóa được thể hiện trong mô hình bằng hình chữ nhật có khung (Frame) đôi; một đường kết hợp bằng nét đôi có dấu gạch ngang thể hiện mối quan hệ là chuyên biệt hoá thành các loại thực thể con chi tiết hơn. Các tên trong các hình hộp chữ nhật thể hiện các thuộc tính của loại thực thể đó.

❖ TÓM TẮT CHƯƠNG 1

Trong chương này, một số nội dung chính được giới thiệu:

- Các khái niệm về cơ sở dữ liệu, các mô hình dữ liệu;
- Mô hình thực thể kết hợp (ERD), phân tích dữ liệu và thiết kế mô hình thực thể kết hợp.

❖ CÂU HỎI VÀ BÀI TẬP CHƯƠNG 1

Bài tập 1: Dựa vào những khái niệm đã học hãy biểu diễn CSDL có các loại mẫu tin Phòng, Nhân viên, Công việc, lý lịch đã trình bày trong mô hình mạng theo cách tiếp cận phân cấp.

Loại liên hệ là phân cấp:
Phòng có nhiều nhân-viên; mỗi nhân-viên chỉ thuộc 1 phòng duy nhất.
Công-việc có nhiều nhân-viên cùng làm, mỗi nhân-viên chỉ làm một công-việc duy nhất.

Mỗi nhân-viên có một lý-lịch; mỗi lý-lịch chỉ thuộc 1 nhân-viên duy nhất.

Bài tập 2: Dựa vào những khái niệm đã học, Anh/Chị hãy biểu diễn CSDL về Tổng điều tra số dân toàn quốc 0 giờ ngày 01 tháng 01 năm 1989 có các loại mẫu tin tỉnh-thành-phố, Quận-huyện, Phường-xã, Địa-bàn, hộ-địa và nhân-khẩu đã trình bày trong mô hình phân cấp theo cách tiếp cận mạng.

Loại liên hệ phân mạng là loại "thuộc về"

Nhân khẩu (thì) thuộc một hộ-điều tra.

Hộ-điều tra (thì) thuộc một địa-bàn.

Địa-bàn điều tra (thì) thuộc một phường-xã.

Phường-xã thuộc một quận-huyện

Quận-huyện thuộc một tỉnh/thành phố.

Bài tập 3: Hệ thống thông tin quản lý kho lưu trữ các văn bản pháp quy tại một cơ quan quản lý Nhà nước có CSDL được phân tích và thiết kế theo cách tiếp cận "Thực thể-kết hợp" gồm các loại thực thể và các mối kết hợp sau:

CÔNG-VĂN-ĐẾN (ngày phát hành; Số công văn; nội dung; ngày nhận; số trang; ghi chú).

CÔNG-VĂN-ĐI (ngày phát hành; Số công văn; nội dung; người ký; số trang; Số tờ trình ký; Ngày trình ký; ghi chú).

CÔNG-VĂN-ĐẾN và CÔNG-VĂN-ĐI đều là CÔNG-VĂN, là hai loại thực thể chuyên biệt hóa của loại thực thể CÔNG-VĂN

CHUYÊN-VIÊN (mã CViên; Tên CViên; Phòng ban; Ghi chú).

GIẢI QUYẾT(mã CViên, số-CViệc, ngày nhận, thời hạn trả lời, KQ giải quyết).

Mỗi công văn đến (từ một đơn vị hay một tác nhân nào đó) có yêu cầu giải quyết thì công văn đó sẽ được chuyển cho một chuyên viên nghiên cứu và đề xuất hướng giải quyết trong một thời hạn nhất định.

Hãy biểu diễn CSDL trên theo cách tiếp cận "thực thể kết hợp".

CHƯƠNG 2: MÔ HÌNH DỮ LIỆU QUAN HỆ

❖ GIỚI THIỆU CHƯƠNG 2

Chương này sẽ giới thiệu các khái niệm cơ bản của cách tiếp cận CSDL theo mô hình quan hệ của E.F.Codd và một số thao tác cơ bản trên các quan hệ. Phần mở đầu của chương II sẽ trình bày kỹ hơn về các khái niệm đã được nhắc tới trong chương I về cách tiếp cận mô hình dữ liệu kiểu quan hệ và sẽ coi đó như là những cơ sở nền tảng để tiếp tục nghiên cứu các phần tiếp theo.

MỤC TIÊU CHƯƠNG 2

Sau khi học xong bài này, người học có khả năng:

□ ***Về kiến thức:***

- Trình bày cụ thể về mô hình dữ liệu quan hệ, các khái niệm, cơ sở lý thuyết của mô hình dữ liệu quan hệ (đại số quan hệ).

□ ***Về kỹ năng:***

- Chuyển đổi được hình ERD sang mô hình dữ liệu quan hệ (ở dạng lược đồ).

- Thực hiện các thao tác an toàn với máy tính

□ ***Về năng lực tự chủ và trách nhiệm:***

- Rèn luyện tính chính xác, sáng tạo, linh hoạt.

- Tuân thủ nội quy, quy định nơi làm việc.

❖ **PHƯƠNG PHÁP GIẢNG DẠY VÀ HỌC TẬP CHƯƠNG 2**

- Đối với người dạy: sử dụng phương pháp giảng dạy tích cực (diễn giảng, vấn đáp, dạy học theo vấn đề); yêu cầu người học thực hiện câu hỏi thảo luận và bài tập chương 2 (cá nhân hoặc nhóm).

- Đối với người học: chủ động đọc trước giáo trình (Chương 2) trước buổi học; hoàn thành đầy đủ câu hỏi thảo luận và bài tập tình huống chương 2 theo cá nhân hoặc nhóm và nộp lại cho người dạy đúng thời gian quy định.

❖ **ĐIỀU KIỆN THỰC HIỆN CHƯƠNG 2**

- ***Phòng học chuyên môn hóa/nhà xưởng:*** Phòng máy tính

- ***Trang thiết bị máy móc:*** Máy chiếu và các thiết bị dạy học khác

- ***Học liệu, dụng cụ, nguyên vật liệu:*** Bài giảng soạn trên slide môn học, giáo trình, tài liệu tham khảo, giáo án và các tài liệu liên quan.

- ***Các điều kiện khác:*** Không có

❖ **KIỂM TRA VÀ ĐÁNH GIÁ CHƯƠNG 2**

- **Nội dung:**

✓ ***Kiến thức:*** Kiểm tra và đánh giá tất cả nội dung đã nêu trong mục tiêu kiến thức

✓ ***Kỹ năng:*** Đánh giá tất cả nội dung đã nêu trong mục tiêu kỹ năng.

✓ ***Năng lực tự chủ và trách nhiệm:*** Trong quá trình học tập, người học cần:

+ *Nghiên cứu bài trước khi đến lớp*

+ *Chuẩn bị đầy đủ tài liệu học tập.*

+ *Tham gia đầy đủ thời lượng môn học.*

+ *Nghiêm túc trong quá trình học tập.*

- **Phương pháp:**
- ✓ **Điểm kiểm tra thường xuyên:** không có
- ✓ **Kiểm tra định kỳ lý thuyết:** 1 điểm kiểm tra (hình thức: viết)

NỘI DUNG CHƯƠNG 2

MÔ HÌNH DỮ LIỆU QUAN HỆ

1. Các khái niệm cơ bản

1.1. Thuộc tính (Attribute)

Thuộc tính là một tính chất riêng biệt của một đối tượng cần được lưu trữ trong CSDL để phục vụ cho việc khai thác dữ liệu về đối tượng.

Ví dụ 2.1.1:

Đối tượng KHOA (tương ứng với loại thực thể KHOA trong mô hình thực thể kết hợp) có các thuộc tính Mã-khoa, Tên-khoa.

Loại thực thể LỚP-HỌC có một số thuộc tính Mã-lớp, Tên-lớp, Niên-khóa, Số-học-viên.

Loại thực thể MÔN-HỌC có một số thuộc tính Mã-môn, Tên-môn, Số-Đv-Học-Trình.

Loại thực thể HỌC-VIÊN có một số thuộc tính Mã-khoa, Mã-học-viên, Tên-học-viên, Ngày-sinh, Quê-quán.

Loại thực thể GIẢNG-VIÊN có một số thuộc tính Mã-giảng-viên, Tên-giảng-viên, Cấp-học-vị, Chuyên-ngành. v.v...

Các thuộc tính được đặc trưng bởi một tên gọi, kiểu giá trị và miền giá trị của chúng. Trong giáo trình này, để trình bày một cách tổng quát và nếu không cần lưu ý đến ngữ nghĩa thì tên của các thuộc tính thường được ký hiệu bằng các chữ cái in hoa đầu tiên trong bảng chữ cái la tinh: A, B, C, D, ... Những chữ cái X, Y, W, Z, ... dùng thay cho một nhóm (hay tập hợp) gồm nhiều thuộc tính. Đôi khi còn dùng các ký hiệu chữ cái với các chỉ số A1, A2, ..., An để chỉ các thuộc tính trong trường hợp tổng quát hay muốn đề cập đến số ngôi (hay số lượng các thuộc tính) của một quan hệ.

Trong các ứng dụng thực tế, người phân tích - thiết kế thường đặt tên thuộc tính một cách gọi nhớ; nhưng để làm rõ hơn ý nghĩa của những tên gọi, người ta có thể đặt tên khá dài cho các thuộc tính với các chữ in hoa đầu từ hoặc viết cách nhau bởi dấu gạch chân (Underscore: _).

Trong các ví dụ của tài liệu này, các tên thuộc tính được viết bằng tiếng Việt gồm nhiều từ Việt nối với nhau bởi dấu trừ (-) có chữ cái đầu tiên được viết in hoa nhằm mục đích chuyển tải cả ngữ nghĩa của tên thuộc tính. Điều này không có gì sai, bởi vì hiện nay có một số hệ quản trị CSDL cho phép làm như vậy (MicroSoft Access, SQL-Server cho phép đặt tên dài tới 255 ký tự và có thể có chứa các khoảng trắng, các ký tự tiếng Việt có dấu và các ký tự đặc biệt khác). Những tên thuộc tính hoặc tên quan hệ như vậy, khi sử dụng trong Micro Soft Access hoặc SQL-Server phải viết chúng trong cặp dấu ngoặc vuông ([]), khi sử dụng trong ORACLE phải viết trong cả dấu nháy kép (" – Quotes). Trong tài liệu này chúng ta sử dụng ký pháp của SQL-Server.

Trong cài đặt cụ thể với một hệ quản trị CSDL cần lưu ý đến khía cạnh đặt tên cho các bảng cũng như tên của thuộc tính. Trong hầu hết các ngôn ngữ lập trình nói chung và một số ngôn ngữ quản trị CSDL nói riêng, tên đối tượng (tên biến, tên quan hệ hay tên thuộc tính v.v...) đều chỉ được phép viết bằng các chữ cái la tinh, chữ số và/hoặc dấu gạch chân (underscore '_'), bắt đầu bằng chữ cái hoặc dấu gạch chân, với độ dài tên theo quy định. Theo lý thuyết, người ta vẫn khuyên rằng không nên đặt tên thuộc tính quá dài (bởi vì nó làm cho việc viết các câu lệnh truy vấn trở nên vất vả hơn) và cũng không nên đặt tên thuộc tính quá ngắn (vì nó không cho thấy ngữ nghĩa của thuộc tính của quan hệ), đặc biệt là không đặt trùng tên hai thuộc tính mang ngữ nghĩa khác nhau thuộc hai đối tượng khác nhau. Chẳng hạn, nếu có hai đối tượng HỌC-VIÊN và GIẢNG-VIÊN đều có thuộc tính TÊN thì nên đặt tên thuộc tính rõ ràng là Tên_học-viên của loại đối tượng HỌC-VIÊN và Tên-giáo-viên cho đối tượng GIẢNG-VIÊN, bởi vì 2 thuộc tính TÊN đó mang ngữ nghĩa khác nhau trong 2 quan hệ khác nhau.

Mỗi thuộc tính đều phải thuộc một kiểu dữ liệu (Data Type) nhất định. Kiểu dữ liệu có thể là vô hướng (đó là các kiểu dữ liệu cơ bản như chuỗi - String hoặc Text hoặc Character, số - Number, Luận lý - Logical, ...) hoặc các kiểu dữ liệu có cấu trúc được định nghĩa dựa trên các kiểu dữ liệu đã có sẵn. Một số kiểu dữ liệu vô hướng sau đây thường được sử dụng trong các hệ quản trị CSDL :

Text (hoặc Character, String, hoặc Char) – kiểu văn bản.

Number (hoặc Numeric, hoặc float) – kiểu số

Logical (hoặc Boolean) – kiểu luận lý

Date/Time – kiểu thời gian : ngày tháng năm + giờ phút

Memo (hoặc VarChar) – kiểu văn bản có độ dài thay đổi.

Mỗi hệ quản trị CSDL có thể gọi tên các kiểu dữ liệu nói trên bằng các tên gọi khác nhau, ngoài ra còn bổ sung thêm một số kiểu dữ liệu riêng của mình. Chẳng hạn, MicroSoft Access có kiểu dữ liệu OLE để chứa các đối tượng nhúng như hình ảnh, âm thanh, audio, video ... ORACLE có kiểu dữ liệu LONG cho phép chứa dữ liệu có kích thước lớn tới 2 tỷ bytes.

Mỗi thuộc tính có thể chỉ chọn lấy những giá trị trong một tập hợp con của kiểu dữ liệu. Tập hợp các giá trị mà một thuộc tính A có thể nhận được gọi là miền giá trị (domain) của thuộc tính A và được ký hiệu là MGT(A) hoặc Dom(A).

Ví dụ 2.1.2:

Học viên đang theo học tại trường ĐHKHTN thì tuổi của họ nhiều nhất là 60 và tuổi ít nhất là 18, vừa mới tốt nghiệp PTTH. Mặc dù nói rằng Năm-sinh của học viên là một số nguyên, nhưng không phải số nguyên nào cũng có thể được chọn để gán vào thuộc tính Năm-sinh. Giá trị năm sinh của học viên chỉ cần lưu 2 chữ số sau của năm sinh tức là chỉ cần một byte để ghi nhận những năm sinh của họ trong thế kỷ 20: từ năm 40 đến năm 82. Với miền giá trị chỉ chứa từ 40 đến 82, như vậy chỉ cần dùng 1 byte để lưu là đủ.

Nếu kiểu dữ liệu của thuộc tính A là có cấu trúc thì miền giá trị của A là tích Đề-các (hoặc tập con của tích Đề-các - Cartesian) của các miền giá trị thành phần.

Ví dụ 2.1.3:

Ta có kiểu dữ liệu ngày tháng năm theo dương lịch với mô tả bằng ngôn ngữ Pascal như sau:

Type DATE = Record

Day : 1 .. 31;

Month : 1 .. 12;

Year : 0 .. 2500;

End;

Tích Đề-các của 3 miền giá trị của các thành phần là $[1 \dots 31] \times [1 \dots 12] \times [0 \dots 2500]$. Nếu thuộc tính A thuộc kiểu DATE thì $MGT(A)$ là $[1..31] \times [1..12] \times [0..2500]$, bởi vì $\{30,02,1999\}$ không phải là một ngày tháng năm hợp lệ nên tổ hợp đó không thuộc $MGT(A)$. Ngày nay, hầu hết các ngôn ngữ quản trị dữ liệu đều có định nghĩa kiểu này như một kiểu cơ bản. Tổ hợp 3 giá trị thành phần luôn luôn được kiểm tra tính đúng đắn trước khi được coi là một giá trị kiểu ngày tháng. Hai phép toán số học có thể tác động trên kiểu DATE là phép cộng (+) hoặc trừ (-) một DATE với một số nguyên để cho kết quả là một giá trị kiểu DATE; hiệu 2 giá trị kiểu DATE là số ngày trôi qua giữa 2 ngày tháng năm đó.

Trong nhiều hệ quản trị CSDL, người ta thường đưa thêm vào miền giá trị của các thuộc tính một giá trị đặc biệt gọi là giá trị rỗng (NULL). Tùy theo ngữ cảnh mà giá trị này có thể đặc trưng cho một giá trị không thể xác định được hoặc một giá trị chưa được xác định ở vào thời điểm nhập tin nhưng có thể được xác định vào một thời điểm khác.

Nếu thuộc tính có kiểu dữ liệu là vô hướng thì nó được gọi là thuộc tính đơn hoặc thuộc tính nguyên tố; nếu thuộc tính có kiểu dữ liệu có cấu trúc thì ta nói rằng nó là thuộc tính kép (hay không phải là nguyên tố).

1.2. Lược đồ quan hệ (Relation schema)

Lược đồ quan hệ là sự trừu tượng hóa của quan hệ, một sự trừu tượng hóa ở mức độ cấu trúc của một bảng hai chiều. Khi nói tới lược đồ quan hệ tức là đề cập tới cấu trúc tổng quát của một quan hệ; khi đề cập tới quan hệ thì điều đó được hiểu rằng đó là một bảng có cấu trúc cụ thể hoặc một định nghĩa cụ thể trên một lược đồ quan hệ với các bộ giá trị của nó.

Lược đồ cơ sở dữ liệu C là tập hợp các lược đồ quan hệ con $\{R_i\}$.

Đôi khi người ta có thể dùng lược đồ quan hệ và quan hệ thay thế cho nhau trong một số trường hợp.

1.3. Quan hệ (Relation)

Một quan hệ R có n ngôi được định nghĩa trên tập các thuộc tính $U = \{A_1, A_2, \dots, A_n\}$ (thứ tự của các thuộc tính là không quan trọng) và kèm theo nó là một tân từ, tức là một quy tắc để xác định mối quan hệ giữa các thuộc tính A_i và được ký hiệu là R ($A_1, A_2, \dots, A_n$).

Tập thuộc tính của quan hệ R đôi khi còn được ký hiệu là R^+ .

Với A_i là một thuộc tính có miền giá trị là $MGT(A_i)$, như vậy $R(A_1, A_2, \dots, A_n)$ là tập con của tích Đề-các: $MGT(A_1) \times MGT(A_2) \times \dots \times MGT(A_n)$.

Quan hệ còn được gọi bằng thuật ngữ khác là bảng (Table).

Ví dụ 2.1.4:

KHOA (Mã-khoa, Tên-khoa), là một quan hệ 2 ngôi.

Tân từ: "Mỗi khoa có một tên gọi và một mã số duy nhất để phân biệt với tất cả các khoa khác của trường".

Ví dụ 2.1.5:

LỚP-HỌC (Mã-lớp, Tên-lớp, Niên-khoá, Số-học-viên, Mã-khoa) là quan hệ 5 ngôi với tân từ: "Mỗi lớp học trong trường có một mã số quy ước duy nhất để phân biệt với tất cả các lớp học khác trong trường; có một tên gọi của lớp học, một số lượng học viên theo học và thuộc một khoa của trường".

Ví dụ 2.1.6:

MÔN-HỌC (Mã-môn, Tên-môn, Số-đv-học-trình) là quan hệ 3 ngôi.

Tân từ: "Mỗi môn học có một tên gọi cụ thể, được học trong một số đơn vị học trình nhất định và ứng với môn học là một mã số duy nhất để phân biệt với mọi môn học khác".

Ví dụ 2.1.7:

HỌC-VIÊN (Mã-học-viên, Tên-học-viên, Ngày-sinh, Quê-quán, Mã-lớp) là quan hệ 5 ngôi.

Tân từ: "Mỗi học viên có một họ và tên, ngày sinh, quê quán, ... và được cấp một mã số duy nhất để phân biệt với mọi học viên khác trong trường; học viên được ghi danh vào một lớp học duy nhất trong trường".

Ví dụ 2.1.8:

GIẢNG-VIÊN (Mã-giảng-viên, Tên-giảng-viên, Cấp-học-vị, Chuyên-ngành).

Đây là quan hệ 4 ngôi.

Tân từ: "Mọi giảng viên đều có họ tên, cấp học vị thuộc một chuyên ngành nhất định và được gán cho một mã số duy nhất, gọi là Mã giảng viên, để phân biệt với mọi giảng viên khác trong trường".

1.4. Bộ giá trị (Tuple)

Một bộ giá trị là các thông tin của một đối tượng thuộc quan hệ. Bộ giá trị cũng thường được gọi là mẫu tin hay bản ghi (record) hoặc dòng của bảng (Row). Về mặt hình thức, một bộ q là một vectơ gồm n thành phần thuộc tập hợp con của tích Đề-các miền giá trị của các thuộc tính và thỏa mãn tân từ đã cho của quan hệ:

$$q = (a_1, a_2, \dots, a_n) \in (MGT(A_1) \times MGT(A_2) \times \dots \times MGT(A_n))$$

Ví dụ 2.1.9:

Đây là 4 bộ giá trị dựa trên các thuộc tính của quan hệ HỌC-VIÊN:

$q_1 = (SV001, \text{Nguyễn Văn Nam}, 27/03/1970, \text{Cần Thơ}, QTKD1)$

$q_2 = (SV005, \text{Vũ Thị Tuyết Mai}, 26/02/1968, \text{Đồng Nai}, KTKC1)$

$q_3 = (SV014, \text{Hồng Đăng}, 30/04/1975, \text{Đồng Nai}, CNTK3)$

$q_4 = (SV015, \text{Lê Hoài Nhớ}, 23/03/1965, \text{Long An}, CNTK4)$

Để lấy thành phần A_i (tức là giá trị thuộc tính A_i) của bộ giá trị q, ta viết $q.A_i$. Phép trích rút này được gọi là phép chiếu một bộ lên thuộc tính A_i . $Q_1.[\text{Tên-học-viên}] = \text{'Nguyễn Văn Nam'}$.

Lưu ý: Trong MicroSoft Access, hằng văn bản được viết trong cặp dấu nháy kép (""), còn trong SQL-Server và trong ORACLE, hằng văn bản được viết trong cặp dấu nháy đơn (' ').

1.5 Thể hiện của quan hệ (Occurrence of a Relation)

Thể hiện (hoặc còn gọi là tình trạng) của quan hệ R, ký hiệu bởi T_R , là tập hợp các bộ giá trị của quan hệ R vào một thời điểm. Tại những thời điểm khác nhau thì quan hệ sẽ có những thể hiện khác nhau. Thể hiện (hay tình trạng) của các lược đồ quan hệ con T_{R_i} gọi là tình trạng của lược đồ cơ sở dữ liệu C.

Ví dụ 2.1.10:

Các thể hiện của quan hệ LỚP-HỌC và MÔN-HỌC:

Quan hệ LỚP-HỌC:

Mã-lớp	Tên-lớp	Niên-khóa	Số-Hvien	Mã-khoa
QTKD1	Quản trị kinh doanh QT01	96-99	145	QTKD
KTCK1	Tài chính - Kế toán KT4	96-99	230	TCKT
KTCK2	Tài chính - Kế toán KT5	97-2000	120	TCKT
CNTK3	Cử nhân Cao Đẳng Tin học K3	98-2000	172	CNTT
CNTK4	Cử nhân Cao Đẳng Tin học K4	99-2001	241	CNTT

Quan hệ MÔN-HỌC:

Mã-môn	Tên-môn	Số-đv-học-trình
TCKT	Tài chính – kế toán	4
KTCT	Kinh tế chính trị	4
MSACB	Hệ quản trị CSDL (Cơ bản)	5
TOANC	Toán Cơ sở	4
LTCBC	Lập trình căn bản C	5
CSDL1	Nhập môn Cơ sở dữ liệu	5
LTHDT	Lập trình hướng đối tượng	5

1.6. Khóa - Siêu khóa - Khóa dự tuyển - Khóa chính - Khóa ngoại

Có nhiều cách khác nhau để định nghĩa khóa:

Định nghĩa 1.6.1

Khóa (Key) của lược đồ quan hệ R định nghĩa trên tập các thuộc tính $U=\{A_1, A_2, \dots, A_n\}$ là một tập con $K \subseteq U$ thỏa mãn các tính chất sau: với mọi bộ giá trị q_1, q_2 của R đều tồn tại một thuộc tính $A \in K$ sao cho $q_1.A \neq q_2.A$.

Điều này có nghĩa là không tồn tại hai bộ nào có giá trị bằng nhau trên mọi thuộc tính của K. Mở rộng phép chiếu của bộ lên tập thuộc tính K ta có thể viết $q_1.K \neq q_2.K$. Như vậy mỗi giá trị của khóa K phải là xác định duy nhất trên quan hệ R.

Theo định nghĩa trên, nếu $K' \subseteq K \subseteq U$ là khóa của lược đồ quan hệ R thì K cũng là khóa của R, bởi vì $q_1.K' \neq q_2.K'$ thì cũng có $q_1.K \neq q_2.K$. Như vậy trong lược đồ quan hệ

có thể có rất nhiều khóa. Việc xác định tất cả các khóa của một lược đồ quan hệ là rất khó khăn.

Định nghĩa này là chưa chặt chẽ. Chúng ta có thể định nghĩa khóa tốt hơn một cách hình thức như sau:

Định nghĩa 1.6.2

Quan hệ R định nghĩa trên tập các thuộc tính $U = \{A_1, A_2, A_n\}$

$K \subseteq U$ là khóa của quan hệ R nếu thỏa 2 điều kiện sau đây:

- (i) K xác định được giá trị của A_j với mọi $j = 1, 2, \dots, n$
- (ii) Không tồn tại $K' \subseteq K$ mà K' có thể xác định được giá trị của A_j với mọi $j = 1, 2, \dots, n$

Nghĩa là K là tập con nhỏ nhất mà giá trị của nó có thể xác định duy nhất một bộ giá trị của quan hệ.

Khoá dự tuyển (Candidate): Khóa của quan hệ theo định nghĩa 1.6.2 được gọi là khóa dự tuyển và là khóa nội của quan hệ. Trong các phần tiếp theo, nếu không có chú thích gì thêm, thì các khóa dự tuyển đều được gọi chung là các khóa.

Siêu khoá (Super key): K là siêu khóa của quan hệ R nếu $K' \subseteq K$ là một khóa của quan hệ. Một lược đồ quan hệ Q của quan hệ R luôn luôn có ít nhất một siêu khóa và có thể có nhiều siêu khóa.

Ví dụ 2.1.11:

Quan hệ LỚP-HỌC (Mã-lớp, Tên-lớp, Niên-khóa, Số-học-viên, Mã-khoa)

Lược đồ LỚP-HỌC có khóa là Mã-lớp và một số siêu khóa sau:

$K1 = \{ \text{Mã-lớp, Tên-lớp} \}$

$K2 = \{ \text{Mã-lớp, Tên lớp, Số-học-viên} \}$

$K3 = \{ \text{Mã-lớp, Số-học-viên} \}$

$K4 = \{ \text{Mã-lớp, Niên-khóa} \}$

Ý nghĩa thực tế của khóa là dùng để nhận diện một bộ trong một quan hệ, nghĩa là, khi cần truy tìm một bộ q nào đó ta chỉ cần biết giá trị của thành phần khóa của q là đủ để dò tìm và hoàn toàn xác định được nó trong quan hệ.

Trong thực tế, đối với các loại thực thể tồn tại khách quan (ví dụ: sinh viên, giảng viên, nhân viên, hàng hóa, ...) người thiết kế cơ sở dữ liệu thường gán thêm cho chúng một thuộc tính giả gọi là mã số để làm khóa chỉ định (ví dụ: mã số sinh viên, mã số giảng viên, mã số nhân viên, mã số hàng hóa, ...). Trong khi đó, các lược đồ quan hệ biểu diễn cho sự trừu tượng hóa thường có khóa chỉ định là một tổ hợp của hai hay nhiều thuộc tính của nó.

Khoá chính (Primary key): Trong trường hợp lược đồ quan hệ Q có nhiều khóa dự tuyển, khi cài đặt trên một hệ quản trị CSDL người sử dụng có thể chọn một trong số các khóa dự tuyển để tạo chỉ mục (Index) chi phối việc truy cập đến các bộ. Khi đó khóa dự tuyển này được gọi là khóa chính. Các khóa còn lại gọi là các khóa tương đương. Khóa chính chỉ thật sự có ý nghĩa trong quá trình khai thác cơ sở dữ liệu và xét trên phương diện lý thuyết, khóa chính hoàn toàn không có vai trò gì khác so với các khóa dự tuyển còn lại.

Một số hệ quản trị CSDL như MicroSoft Access, Paradox, Oracle, Informix, DB2 ... có cài đặt cơ chế tự động kiểm tra tính duy nhất trên khóa chính. Tức là, nếu thêm một bộ mới

q2 có giá trị khóa chính trùng với giá trị khóa chính của một bộ q1 nào đó đã có trong quan hệ thì hệ thống sẽ báo lỗi và yêu cầu nhập lại một giá trị khác.

Người ta cũng qui ước rằng:

Trong một bộ của một quan hệ các thuộc tính khóa không chứa giá trị rỗng.

Không được phép sửa đổi giá trị của thuộc tính khóa. Nếu muốn sửa đổi giá trị thuộc tính khóa của một bộ q, người sử dụng phải hủy bỏ bộ q và sau đó thêm mới một bộ q' với giá trị khóa đã được sửa đổi.

Các thuộc tính có tham gia vào một khóa được gọi là thuộc tính khóa. Trong lược đồ quan hệ, các thuộc tính khóa sẽ được gạch dưới. Ngược lại, các thuộc tính không tham gia vào một khóa nào gọi là thuộc tính không khóa.

Ví dụ 2.1.12:

KHOA (Mã-khoa, Tên-khoa)

LỚP-HỌC (Mã-lớp, Tên-lớp, Niên-khoá, Số-học-viên, Mã-khoa)

MÔN-HỌC (Mã-môn, Tên-môn, Số-đv-học-trình).

HỌC-VIÊN (Mã-học-viên, Tên-học-viên, Ngày-sinh, Quê-quán, Mã-lớp).

GIẢNG-VIÊN (Mã-giảng-viên, Tên-giảng-viên, Cấp-học-vị, Chuyên-ngành).

KQUẢ_THI (Mã-học-viên, Mã-môn, Lần-thi, Ngày-thi, Điểm-thi, Ghi-chú).

Khoá ngoại (Foreign key): K là khoá ngoại của một quan hệ nếu nó là khoá nội của một quan hệ khác.

Ví dụ 2.1.13:

Mã-khoa trong quan hệ LỚP-HỌC là khoá ngoại vì nó là khoá nội của quan hệ KHOA.

Mã-lớp trong quan hệ HỌC-VIÊN là khoá ngoại của quan hệ HỌC-VIÊN vì nó là khoá nội của quan hệ LỚP-HỌC.

1.7. Phụ thuộc hàm (Functional Dependency)

Quan hệ R được định nghĩa trên tập thuộc tính $U = \{ A_1, A_2, \dots, A_n \}$. $X, Y \subset U$ là 2 tập con của tập thuộc tính U. Nếu tồn tại một ánh xạ $f: X \rightarrow Y$ thì ta nói rằng X xác định hàm Y, hay Y phụ thuộc hàm vào X và ký hiệu là $X \rightarrow Y$. Chúng ta sẽ tìm hiểu kỹ hơn về phụ thuộc hàm trong Chương IV,

1.8. Ràng buộc toàn vẹn (Integrity Constraint, Rule)

Ràng buộc toàn vẹn (viết tắt là RBTV) là một quy tắc định nghĩa trên một (hay nhiều) quan hệ do môi trường ứng dụng quy định. Đó chính là quy tắc để đảm bảo tính nhất quán của dữ liệu trong CSDL.

Mỗi RBTV được định nghĩa bằng một thuật toán trong CSDL.

Ví dụ 2.1.16:

Quan hệ CCVC (Mã-CBVC, Họ-tên-CBVC, Hệ-số-lương)

Quy tắc: Hệ số lương của cán bộ viên chức (CBVC) phải lớn hơn hay bằng 1.00 và nhỏ hơn hay bằng 10.00.

Thuật toán: "cc $\in$ CCVC thì cc.Hệ-số-lương ≥ 1 & cc.Hệ-số-lương ≤ 10 ."

Các khái niệm cũng như các vấn đề chủ yếu của RBTV sẽ được trình bày chi tiết trong Chương IV.

2. Chuyển mô hình thực thể kết hợp sang mô hình dữ liệu quan hệ

Sau đây là một số quy tắc được sử dụng trong việc chuyển đổi mô hình thực thể kết hợp sang mô hình dữ liệu quan hệ.

Quy tắc 1:

Chuyển đổi mỗi loại thực thể thành một lược đồ quan hệ, các thuộc tính của loại thực thể thành các thuộc tính của lược đồ quan hệ, thuộc tính khoá của loại thực thể là thuộc tính khoá của lược đồ quan hệ.

Chẳng hạn loại thực thể Sinhvien ở ví dụ 2.1.12 khi áp dụng quy tắc 1 thì sẽ được chuyển thành lược đồ quan hệ HỌC-VIÊN như sau:

HỌC-VIÊN (Mã-học-viên, Tên-học-viên, Ngày-sinh, Quê-quán, Mã-lớp).

Quy tắc 2:

Nếu mỗi kết hợp mà cả hai nhánh của nó đều có bản số max là n thì mỗi kết hợp này sẽ được chuyển thành một lược đồ quan hệ K' gồm các thuộc tính của mỗi kết hợp K, cộng thêm các thuộc tính khoá của hai lược đồ quan hệ A, B tương ứng với hai thực thể tham gia vào mỗi kết hợp. Khoá của lược đồ quan hệ K' gồm cả hai khoá của hai lược đồ quan hệ A và B.

Chẳng hạn mỗi kết hợp Phancong giữa ba loại thực thể Giangvien, Monhoc và Lop được chuyển thành lược đồ quan hệ Phancong và có tập khoá là {MAGV,MAMH,MALOP} như sau: Phancong(MAGV,MAMH,MALOP)

Quy tắc 3:

Mỗi kết hợp mà một nhánh có bản số là n (nhánh B) và nhánh còn lại có bản số max là 1 (nhánh A) thì loại bỏ mỗi kết hợp này khỏi mô hình thực thể kết hợp và thêm các thuộc tính khoá của lược đồ tương ứng với loại thực thể ở nhánh B vào lược đồ tương ứng với loại thực thể ở nhánh A (khoá của B sẽ thành khoá ngoại của A). Nếu mỗi kết hợp có các thuộc tính thì những thuộc tính này cũng được thêm vào lược đồ quan hệ tương ứng với loại thực thể ở nhánh A.

Chẳng hạn mỗi kết hợp thuộc giữa hai loại thực thể Sinhvien và Lop nên lược đồ quan hệ Sinhvien được sửa thành như sau: Sinhvien(MASV,HOTENSV,NU,NGAYSINH,TINH,MALOP)

Quy tắc 4:

Nếu mỗi kết hợp mà cả hai nhánh đều có bản số max là 1 thì áp dụng quy tắc 3 cho một trong hai nhánh tùy chọn.

Ví dụ 2.1:

Sau đây là mô hình dữ liệu quan hệ được chuyển từ mô hình thực thể kết hợp ở ví dụ 2.1.12

KHOA (Mã-khoa, Tên-khoa)

LỚP-HỌC (Mã-lớp, Tên-lớp, Niên-khoá, Số-học-viên, Mã-khoa)

MÔN-HỌC (Mã-môn, Tên-môn, Số-đv-học-trình).

HỌC-VIÊN (Mã-học-viên, Tên-học-viên, Ngày-sinh, Quê-quán, Mã-lớp).

GIẢNG-VIÊN (Mã-giảng-viên, Tên-giảng-viên, Cấp-học-vị, Chuyên-ngành).

KQUẢ_THI (Mã-học-viên, Mã-môn, Lần-thi, Ngày-thi, Điểm-thi, Ghi-chú).

3. Ngôn ngữ đại số quan hệ

Ngôn ngữ đại số quan hệ là ngôn ngữ biểu diễn câu hỏi về các quan hệ. Do các cách biểu diễn khác nhau nên trong tài liệu này, ngôn ngữ đại số quan hệ được chia làm 3 nhóm: Các phép toán tập hợp - các phép toán này được biểu diễn như các phép toán trên tập hợp, các phép toán quan hệ và các phép toán khác liên quan chủ yếu đến phép kết nối.

Các phép toán cơ bản trên tập hợp được áp dụng trên tập các bộ giá trị của các quan hệ, đó là: Hợp (*Union*), Hiệu (Trừ - *Minus*), Giao (*Intersection*), Tích Đề-các (*Cartesian*) và phép chia (*Division*).

Giả thiết: $U = \{ A_1, A_2, A_3, \dots, A_n \}$ là tập các thuộc tính.

R và S là 2 quan hệ được định nghĩa trên U có cùng thứ tự của các thuộc tính. Và ở đây chúng ta luôn luôn giả thiết là R và S có số lượng hữu hạn các bộ giá trị. Bài dưới đây sẽ trình bày các phép toán tập hợp trên hai quan hệ.

3.1. Phép hợp 2 quan hệ (Union)

Hợp của hai quan hệ R và S - được ký hiệu là $R \cup S$ - là một quan hệ Q xác định trên tập thuộc tính U, có cùng thứ tự thuộc tính như trong quan hệ R và S, được định nghĩa như sau:

$$Q = R \cup S = \{ t / t \in R \text{ hoặc } t \in S \}$$

Nói một cách đơn giản, hợp của 2 quan hệ R và S là một quan hệ có cùng ngôi với quan hệ R và S với các bộ giá trị bằng gộp các bộ giá trị của cả R và S; những bộ giá trị trùng nhau chỉ được giữ lại 1 bộ.

Ví dụ 2.2.1: Quan hệ Đơn vị A có các bộ giá trị sau:

<u>MãSố</u>	<u>Họ-tên</u>	<u>Phái</u>	<u>Chức-danh</u>	<u>Lương</u>	<u>MãĐV</u>
100	Nguyễn Văn Nam	Nam	Giám đốc	2.500.000	10
101	Hoàng Thị Xuân	Nữ	Kế toán trưởng	1.700.000	10
103	Đặng Ngọc Chiến	Nữ	Thư ký	1.000.000	10
105	Phan Kỳ Nhân	Nam	Lái xe	700.000	10

Quan hệ Đơn vị B có các bộ giá trị sau:

<u>MãSố</u>	<u>Họ-tên</u>	<u>Phái</u>	<u>Chức-danh</u>	<u>Lương</u>	<u>MãĐV</u>
210	Nguyễn Thị Cao	Nữ	Trưởng phòng	1.200.000	30
101	Hoàng Thị Xuân	Nữ	Kế toán trưởng	1.700.000	10

221	Đỗ Hữu Ngọc	Nam	Phó phòng	1.000.000	30
233	Hoàng Thao	Nam	Chuyên viên	1.000.000	30

Hợp của hai quan hệ trên cho kết quả là quan hệ NV-CTy có 7 bộ giá trị sau:

<u>MãSố</u>	<u>Họ-tên</u>	<u>Phái</u>	<u>Chức-danh</u>	<u>Lương</u>	<u>MãĐV</u>
100	Nguyễn Văn Nam	Nam	Giám đốc	2.500.000	10
101	Hoàng Thị Xuân	Nữ	Kế toán trưởng	1.700.000	10
103	Đặng Ngọc Chiến	Nữ	Thư ký	1.000.000	10
105	Phan Kỳ Nhân	Nam	Lái xe	700.000	10
210	Nguyễn Thị Cao	Nữ	Trưởng phòng	1.200.000	30
221	Đỗ Hữu Ngọc	Nam	Phó phòng	1.000.000	30
233	Hoàng Thao	Nam	Chuyên viên	1.000.000	30

Bộ giá trị có mã số nhân viên là 101 xuất hiện 2 lần trong 2 quan hệ Đơn vị A và Đơn vị B, nhưng trong quan hệ NV-Cty bộ này chỉ được giữ lại 1 lần, do đó chỉ còn 7 bộ giá trị.

3.2. Phép giao 2 quan hệ (Intersection)

Giao của hai quan hệ R và S, được ký hiệu là $R \cap S$, là một quan hệ Q xác định trên tập thuộc tính U, có cùng thứ tự thuộc tính như trong quan hệ R và S, được định nghĩa như sau:

$$Q = R \cap S = \{ t \mid t \in R \text{ và } t \in S \}$$

Nói một cách đơn giản, giao của 2 quan hệ R và S là một quan hệ có cùng ngôi với quan hệ R và S với các bộ giá trị là các bộ giống nhau của cả hai quan hệ R và S.

Ví dụ 2.2.2:

Với hai quan hệ như trên, hiệu của Đơn vị A và Đơn vị B là quan hệ NV-CtyA với các bộ sau:

<u>MãSố</u>	<u>Họ-tên</u>	<u>Phái</u>	<u>Chức-danh</u>	<u>Lương</u>	<u>MãĐV</u>
101	Hoàng Thị Xuân	Nữ	Kế toán trưởng	1700.000	10

3.3. Phép trừ 2 quan hệ (Minus)

Hiệu của hai quan hệ R và S, được ký hiệu là $R - S$, là một quan hệ Q xác định trên tập thuộc tính U, có cùng thứ tự thuộc tính như trong quan hệ R và S, được định nghĩa như sau:

$$Q = R - S = \{ t \mid t \in R \text{ và } t \notin S \}$$

Nói một cách đơn giản, hiệu của 2 quan hệ R và S là một quan hệ có cùng ngôi với quan hệ R và S với các bộ giá trị là các bộ giá trị của R sau khi đã loại bỏ đi các bộ có mặt trong quan hệ S.

Ví dụ 2.2.3:

Với hai quan hệ như trên, hiệu của Đơn vị A và Đơn vị B là quan hệ NV-CtyA với các bộ sau:

<i>MãSố</i>	<i>Họ-tên</i>	<i>Phái</i>	<i>Chức-danh</i>	<i>Lương</i>	<i>MãĐV</i>
100	Nguyễn Văn Nam	Nam	Giám đốc	2.500.000	10
103	Đặng Ngọc Chiến	Nữ	Thư ký	1.000.000	10
105	Phan Kỳ Nhân	Nam	Lái xe	700.000	10

3.4. Tích Decac của 2 quan hệ (Cartesian)

$R (A_1, A_2, \dots, A_n)$ và $S (B_1, B_2, \dots, B_m)$ là hai quan hệ có số bộ giá trị hữu hạn. Tích Đề-các của hai quan hệ R và S , được ký hiệu là $R \times S$, là một quan hệ Q xác định trên tập thuộc tính của R và S (với $n + m$ thuộc tính) và được định nghĩa như sau:

$Q = R \times S = \{ t / t \text{ có dạng } (a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m) \text{ trong đó } (a_1, a_2, \dots, a_n) \in R \text{ và } (b_1, b_2, \dots, b_m) \in S \}$

Nói một cách đơn giản, tích Đề-các của 2 quan hệ R và S là một quan hệ Q có số ngôi bằng tổng số ngôi của R và S , với các bộ giá trị gồm 2 phần: phần bên trái là một bộ giá trị của R và phần bên phải là một bộ giá trị của S . Như vậy, nếu R có n_1 bộ giá trị và S có n_2 bộ giá trị, thì Q sẽ có $n_1 \times n_2$ bộ giá trị.

Ví dụ 2.2.4:

$R (A, B, C)$	$S(D, E, F)$	$R \times S = Q (A, B, C, D, E, F)$
a_1, b_1, c_1	d_1, e_1, f_1	$a_1, b_1, c_1, d_1, e_1, f_1$
a_2, b_2, c_2	d_2, e_2, f_2	$a_1, b_1, c_1, d_2, e_2, f_2$
a_3, b_3, c_3		$a_2, b_2, c_2, d_1, e_1, f_1$
		$a_2, b_2, c_2, d_2, e_2, f_2$
		$a_3, b_3, c_3, d_1, e_1, f_1$
		$a_3, b_3, c_3, d_2, e_2, f_2$

3.5. Phép chia 2 quan hệ (Division)

R là quan hệ n ngôi và S là quan hệ m ngôi ($n > m$ và $S \uparrow I$), có m thuộc tính chung (giống nhau về mặt ngữ nghĩa, hoặc các thuộc tính có thể so sánh được) giữa R và S . Phép chia 2 quan hệ R và S , ký hiệu là $R \div S$, là một quan hệ Q có $n - m$ ngôi được định nghĩa như sau:

$Q = R \div S = \{ t \mid t \uparrow u \in S, (t, u) \in R \}$

Sử dụng định nghĩa phép tích Đề-các, có thể định nghĩa phép chia hình thức hơn như sau:

$R \div S = Q$ sao cho $Q \times S \uparrow R$ (với giả thiết thêm là thứ tự thuộc tính của R, S, Q là không quan trọng).

Ví dụ 2.2.5:

$R (A, B, C, D)$	$S (C, D)$	$R \div S = (A, B)$
a, b, c, d	c, d	a, b
a, b, e, f	e, f	c, d
b, c, e, f		
c, d, c, d		
c, d, e, f		
a, b, d, e		

Ví dụ 2.2.6:

Cho quan hệ về khả năng lái các loại máy bay của các phi công:

KHẢ-NĂNG (Số-hiệu-phi-công, Số-hiệu-máy-bay)

<u>Số-hiệu-phi-công</u>	<u>Số-hiệu-máy-bay</u>
32	102
30	101
30	103
32	103
33	100
30	102
31	102
30	100
31	100

Câu hỏi: Cho biết các phi công có khả năng lái được cả 3 loại máy bay 100, 101, và 103 ?

Trả lời: Đó là thương của phép chia quan hệ KHẢ-NĂNG cho quan hệ MÁY-BAY (Số-hiệu-máy-bay):

100
101
103

Và kết quả là quan hệ PHI-CÔNG (Số-hiệu-phi-công) có 1 bộ giá trị (30).

3.6. Phép chiếu (Projection)

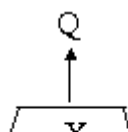
Giả sử R là một quan hệ xác định trên tập thuộc tính $U = \{ A_1, A_2, \dots, A_n \}$. $X \subseteq U$. Phép chiếu quan hệ R trên tập con các thuộc tính X là một quan hệ Q xác định trên tập thuộc tính X , ký hiệu là $R[X]$, được định nghĩa như sau:

$$Q = R[X] = \{ q \mid \exists t \in R: q = t.X \}$$

Đôi khi người ta còn ký hiệu phép chiếu quan hệ R trên tập thuộc tính X bằng $P_X(R)$. Nếu $X = \{ A_{i1}, A_{i2}, \dots, A_{im} \}$ thì có thể viết cụ thể là $P_{A_{i1}A_{i2}\dots A_{im}}(R)$.

Ngữ nghĩa: Trích từ R một số thuộc tính nào đó để tạo thành một quan hệ mới. Số ngôi của quan hệ mới này bằng số thuộc tính của tập con X . Các bộ giá trị của các cột được trích nếu giống nhau sẽ được loại bỏ để chỉ giữ lại một bộ duy nhất (trong thể hiện của quan hệ mới không có 2 bộ nào giống nhau).

Phép chiếu được biểu diễn bằng sơ đồ như sau:



4. Hình 2.3.1. Sơ đồ biểu diễn phép chiếu

Ví dụ 2.3.1:

Cho quan hệ về các nhân viên đang làm việc tại một Công ty (EMPLOYEE) như sau:

<i>Empno</i>	<i>Name</i>	<i>Job</i>	<i>Salary</i>	<i>Comm</i>	<i>Deptno</i>	<i>Sex</i>
100	Wilson	Clrk	1.700		10	M
101	Smith	Slsn	2.500	1.300	40	F
103	Reed	Anlt	3.500		30	M
105	Watson	Mngr	4.500	0	30	M
109	Allen	Mngr	3.800	8.000	40	F
110	Turner	Clrk	1.800		50	M
200	Chen	Mngr	2.900		10	F
210	Ramirez	Mngr	3.600		50	M
213	McDonnell	Clrk	1.625		60	M
214	Simpson	Drvr	825		60	M
215	Di Salvo	Spvr	2.700		60	M
220	Schwartz	Slsn	4.200	5.300	40	F

Câu hỏi: Hãy cho biết các loại công việc mà các nhân viên của Cty đang làm?

Câu trả lời: Chiếu quan hệ EMPLOYEE lên thuộc tính Job, chỉ giữ lại các giá trị khác nhau. Kết quả là một quan hệ $JOB_{EMPLOYEE} = EMPLOYEE [JOB]$ gồm 1 cột 6 dòng với các giá trị { Clrk, Slsn, Anlt, Mngr, Drvr, và Spvr }.

3.7 Phép chọn (Selection)

Phép chọn cho phép chọn lựa chỉ những bản ghi thỏa mãn một điều kiện Đ nào đó để đưa vào quan hệ kết quả. Điều kiện Đ chính là một biểu thức logic cho kết quả hoặc là đúng (*True*) hoặc là sai (*False*) khi đánh giá trên các bộ giá trị của quan hệ nguồn; nó là tổ hợp của các biểu thức logic cơ sở. Mỗi biểu thức cơ sở chứa một phép so sánh: nhỏ hơn (<), nhỏ hơn hay bằng (<=), lớn hơn (>), lớn hơn hay bằng (>=), bằng nhau (=) và khác (≠ hoặc <>) có dạng:

Thuộc tính so sánh với thuộc tính.

Thuộc tính so sánh với hằng (*literal*)

Các biểu thức logic cơ sở được tổ hợp với nhau bởi các phép toán logic: phép "và" logic - hay còn gọi là phép nối liền (- *conjunction*), phép "hoặc" logic - hay còn gọi là phép nối rời (- *disjunction*) và phép phủ định (- *not*).

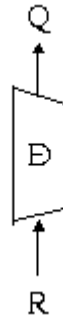
Giả sử R ($A_1, A_2, \dots, A_n$) là một quan hệ, và Đ là một điều kiện (hoặc còn gọi là một *phát biểu*) dựa trên tập con thuộc tính R^+ . Đánh giá điều kiện Đ trên bộ giá trị t Ỗ R được ký hiệu là E (t_D) hoặc để đơn giản, ta có thể viết Đ(t). Phép chọn các bản ghi của R thỏa mãn

điều kiện Đ là một quan hệ Q có cùng ngôi với R, ký hiệu là R:Đ, được định nghĩa hình thức như sau:

$$Q = \{ t \in R \mid \text{Đ}(t) = \text{đúng} \}$$

Phép chọn cũng có thể được ký hiệu theo C.J.Date [3] bởi dấu sigma (σ): $\sigma_F(R)$.

Phép chọn được biểu diễn bằng sơ đồ như sau:



5. Hình 2.3.2. Sơ đồ biểu diễn phép chọn

6.

Ví dụ 2.3.2: Cũng với quan hệ EMPLOYEE như trong ví dụ 5.3.1, điều kiện chọn là: "những người làm lãnh đạo các phòng ban và có mức lương tháng từ 3.500 \$ trở lên". Bằng ngôn ngữ đại số qua hệ với phép toán chọn, câu hỏi được diễn đạt như sau:

EMPLOYEE : (Job = Mngr) (Salary $\geq$ 3500).

3.8 Phép θ - kết

Giả sử có 2 quan hệ R ($A_1, A_2, \dots, A_n$) và S ($B_1, B_2, \dots, B_m$).

$t = (a_1, a_2, \dots, a_n)$ là một bộ giá trị của R và $u = (b_1, b_2, \dots, b_m)$ là một bộ giá trị của S.

Gọi v là bộ ghép nối u vào t (hay bộ giá trị t và u được "xếp cạnh nhau" để tạo thành bộ giá trị mới v) được định nghĩa như sau:

$$v = (t, u) = (a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m).$$

$A \in R^+$ và $B \in S^+$ là hai thuộc tính có thể so sánh được.

Gọi q là một trong các phép toán so sánh $\{ <, \leq, >, \geq, =, ^1 \}$.

Phép kết nối hai quan hệ (có thể nói tắt là *phép kết*) R với S trên các thuộc tính A và B với phép so sánh q , với giả thiết là giá trị cột R[A] có thể so sánh được (qua phép so sánh q) với mỗi giá trị của cột R[B], được định nghĩa qua:

$$R \bowtie_{A \theta B} S = \{ v = (t, u) \mid t \in R, u \in S \text{ và } t.A \theta u.B \}$$

$$\text{Hoặc: } R \bowtie_{A \theta B} S = (R \times S) : (A \theta B).$$

Phép kết nối 2 quan hệ R và S có thể xem như được thực hiện qua 2 bước:

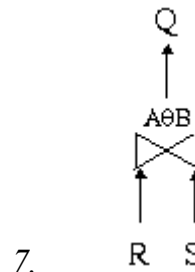
Bước 1: Thực hiện tích Đề-các hai quan hệ R và S.

Bước 2: Chọn các bộ giá trị thỏa mãn điều kiện $A \theta B$.

Ngữ nghĩa: Định nghĩa trên cho ta kết quả của phép kết nối hai quan hệ R và S với phép so sánh q trên 2 thuộc tính A và B là một quan hệ mới, Đó là kết quả cuối cùng của phép toán quan hệ (phép *Chọn*) trên quan hệ kết quả của phép toán tập hợp (tích *Đề-các*).

Nếu q là phép toán so sánh bằng nhau ($=$) thì ta gọi đó là phép *kết nối bằng* (*Equi Join*). Nếu các thuộc tính so sánh là giống tên nhau thì trong kết quả của phép kết nối sẽ loại bỏ đi một trong 2 thuộc tính đó, khi đó phép kết nối được gọi là *phép kết nối tự nhiên* (*Natural Join*) và sử dụng ký hiệu cho phép toán là " $*$ " hoặc chỉ ký hiệu (không có $A \ q \ B$) ở phía trên của phép toán. Trong các trường hợp còn lại, phép toán được gọi chung là *phép kết nối theta* (q -Join).

Phép kết nối được biểu diễn bằng sơ đồ như sau:



7.
8. Hình 2.3.3. Sơ đồ biểu diễn phép kết nối
9.

Ví dụ 2.3.3:

Cho 2 quan hệ R và S với các bộ giá trị cho trong bảng ở cột lớn thứ nhất và thứ hai bên trái. Kết quả *phép kết nối theta* (q -Join) của 2 quan hệ trên được cho trong bảng ở cột lớn thứ ba dưới đây:

Kết quả là một quan hệ gồm 3 dòng sau:

<i>Empno</i>	<i>Name</i>	<i>Job</i>	<i>Salary</i>	<i>Comm</i>	<i>Deptno</i>	<i>Sex</i>
105	Watson	Mngr	4.500		30	M
109	Allen	Mngr	3.800	8.000	40	F
210	Ramirez	Mngr	3.600		50	M

được định nghĩa qua:

$$R \bowtie_{A \theta B} S = \{ v = (t, u) \mid t \in R, u \in S \text{ và } t.A \theta u.B \}$$

$$\text{Hoặc: } R \bowtie_{A \theta B} S = (R \times S) : (A \theta B).$$

Phép kết nối 2 quan hệ R và S có thể xem như được thực hiện qua 2 bước:

Bước 1: Thực hiện tích Đề-các hai quan hệ R và S.

Bước 2: Chọn các bộ giá trị thỏa mãn điều kiện $A \theta B$.

Ngữ nghĩa: Định nghĩa trên cho ta kết quả của phép kết nối hai quan hệ R và S với phép so sánh q trên 2 thuộc tính A và B là một quan hệ mới, Đó là kết quả cuối cùng của phép toán quan hệ (phép *Chọn*) trên quan hệ kết quả của phép toán tập hợp (tích *Đề-các*).

Nếu q là phép toán so sánh bằng nhau ($=$) thì ta gọi đó là phép *kết nối bằng* (*Equi Join*). Nếu các thuộc tính so sánh là giống tên nhau thì trong kết quả của phép kết nối sẽ loại bỏ đi một trong 2 thuộc tính đó, khi đó phép kết nối được gọi là *phép kết nối tự nhiên* (*Natural Join*) và sử dụng ký hiệu cho phép toán là " $*$ " hoặc chỉ ký hiệu (không có $A \ q$

B) ở phía trên của phép toán. Trong các trường hợp còn lại, phép toán được gọi chung là *phép kết nối theta* (θ -Join).

Phép kết nối được biểu diễn bằng sơ đồ như sau:



10. R S

11. Hình 2.3.3. Sơ đồ biểu diễn phép kết nối

12.

Ví dụ 2.3.3:

Cho 2 quan hệ R và S với các bộ giá trị cho trong bảng ở cột lớn thứ nhất và thứ hai bên trái. Kết quả *phép kết nối theta* (θ -Join) của 2 quan hệ trên được cho trong bảng ở cột lớn thứ ba dưới đây:

R	A	B	C	S	C	D	E	$R.B \geq S.C$	=Q	A	B	C	C	D	E
	a_1	1	1		1	d_1	e_1			a_1	1	1	1	D_1	e_1
	a_2	2	1		2	d_2	e_2			a_2	2	1	1	D_1	e_1
	a_3	2	2		3	d_3	e_3			a_2	2	1	2	D_2	e_2
										a_3	2	2	1	D_1	e_1
										a_3	2	2	2	D_2	e_2

Kết quả *phép kết nối tự nhiên* của 2 quan hệ R và S là quan hệ Q' với các bộ giá trị như sau:

$R * S = Q'$	A	B	C	D	E
	a_1	1	1	d_1	e_1
	a_2	2	1	d_1	e_1
	a_3	2	2	d_2	e_2

Ví dụ 2.3.4:

Cho CSDL về cán bộ - viên chức Nhà nước (CBVC) với các quan hệ sau đây:

1. Quan hệ ĐƠN-VỊ:

Mã-ĐV	Tên-đơn-vị	Loại
10002	Trường Kỹ thuật nghiệp vụ máy tính	3
10003	Phòng quản lý hành chính	2

10022	Văn phòng đại diện Thanh niên	2
10070	Ban quản lý vốn sinh viên	5
10071	Lực lượng quản lý thị trường	5

2. Quan hệ loại hình tổ chức của đơn vị LOAI-ĐV:

<i>Loại</i>	<i>Tên-loại-hình</i>
2	Hành chính
3	Sự nghiệp hoàn toàn
5	Hạch toán độc lập

3. Quan hệ Ngạch-CBVC:

<i>Ngạch</i>	<i>Tên ngạch</i>
01002	Chuyên viên chính
01003	Chuyên viên
01004	Cán sự
01005	Kỹ thuật viên đánh máy
01008	Nhân viên văn thư
01010	Lái xe cơ quan

4. Quan hệ Ngạch-Bậc-lương:

<i>Ngạch</i>	<i>Bậc</i>	<i>Hệ-số-lương</i>
01002	01	3.35
01002	02	3.63
01002	03	3.91
01003	05	2.82
01003	07	3.31
01003	08	3.56
01004	07	2.18
01004	08	2.30
01004	10	2.55
01004	11	2.68
01005	04	2.06
01005	05	2.18

01005	06	2.30
01008	11	2.12
01010	08	2.47
01010	11	2.80
01010	13	3.02

5. Quan hệ danh sách CBVC với các bộ giá trị sau:

<i>Mã-ĐV</i>	<i>Mã-CC</i>	<i>Họ-lót</i>	<i>Tên</i>	<i>Giới</i>	<i>Ngày-sinh</i>	<i>Ngạch</i>	<i>Bậc</i>	<i>Ngày-xếp</i>
10002	1000028	Trần Tứ	Hải	Nam	05/09/40	01003	08	01/12/96
10002	1000040	Trần Ngọc	Son	Nam	04/08/57	01003	05	01/12/97
10002	1000042	Nguyễn Văn	Sang	Nam	20/04/61	01004	10	01/01/97
10002	1000043	Nguyễn Văn	Thành	Nam	04/10/44	01004	10	01/01/97
10002	1000065	Huỳnh Thị	Hoa	Nữ	06/04/61	01004	07	01/01/97
10003	1000156	Huỳnh Ngọc	Thúy	Nữ	28/10/54	01005	06	01/09/97
10003	1000134	Nguyễn Văn	Bạc	Nam	08/09/42	01010	13	01/12/97
10003	1000159	Lê Văn	Sang	Nam	15/06/50	01008	11	01/03/96
10003	1000160	Trịnh Ngọc	Tâm	Nam	18/11/66	01010	08	01/02/96
10022	1000218	Nguyễn Cửu	Châu	Nam	19/11/47	01010	11	01/10/96
10022	1000219	Nguyễn Văn	Hùng	Nam	15/05/55	01003	05	01/12/95
10022	1000220	Nguyễn Kim	Lưu	Nữ	22/07/55	01004	12	01/12/95

Phép kết nối tự nhiên 2 quan hệ ĐƠN-VI và LOẠI-ĐV là một quan hệ gồm 4 thuộc tính: Mã-ĐV, Tên-đơn-vị, Loại và Tên-loại với các bộ giá trị sau:

<i>Mã-ĐV</i>	<i>Tên-đơn-vị</i>	<i>Loại</i>	<i>Tên-loại-hình</i>
10002	Trường Kỹ thuật nghiệp vụ máy tính	3	Sự nghiệp hoàn toàn
10003	Phòng quản lý hành chính	2	Hành chính
10022	Văn phòng đại diện Thanh niên	2	Hành chính
10070	Ban quản lý vốn sinh viên	5	Hạch toán độc lập

10071	Lực lượng quản lý thị trường	5	Hạch toán độc lập
-------	------------------------------	---	-------------------

3.9 Các phép toán kết nối khác

Mục này trình bày 3 phép toán kết nối mở rộng khác đặc biệt quan trọng, mà bản chất của chúng vẫn là kết nối. Chúng đã được cài đặt trong một số hệ quản trị CSDL như MicroSoft Access, SQL-Server, Oracle. Các phép kết nối đó là: Kết nối nội (*Inner Join*), Kết nối trái (*Left Join*) và Kết nối phải (*Right Join*).

3.9.1. Phép kết nối nội (Inner Join)

Thực chất là phép *kết nối bằng* đã trình bày trên. Tuy nhiên, ngay cả trong trường hợp hai thuộc tính so sánh có cùng tên thì kết quả phép kết nối vẫn giữ lại 2 tên thuộc tính đó.

Ví dụ 3.4.1.1:

Cho 2 quan hệ R (A, B, C) và S (A, D, E, F) với các bộ giá trị như dưới đây. Kết quả của phép *kết nối nội* được cho trong bảng phía bên phải.

R(A B C)	S(A D E F)	R	S=Q (A B C A D E F)
a ₁ b ₁ c ₁	a ₁ d ₁ e ₁ f ₁		a ₁ b ₁ c ₁ a ₁ d ₁ e ₁ f ₁
a ₂ b ₂ c ₂	a ₂ d ₂ e ₂ f ₂		a ₂ b ₂ c ₂ a ₂ d ₂ e ₂ f ₂
a ₃ b ₃ c ₃	a ₄ d ₄ e ₄ f ₄		a ₇ b ₇ c ₇ a ₇ d ₇ e ₇ f ₇
a ₅ b ₅ c ₅	a ₆ d ₆ e ₆ f ₆		
a ₇ b ₇ c ₇	a ₇ d ₇ e ₇ f ₇		

Ví dụ 3.4.1.2:

Phép kết nối nội 2 quan hệ ĐƠN-VỊ và LOẠI-ĐV cho kết quả là một bảng sau:

Mã-ĐV	Tên-đơn-vị	Loại	Loại	Tên-loại-hình
10002	Trường K.thuật nghiệp vụ máy tính	3	3	SN hoàn toàn
10003	Phòng quản lý hành chính	2	2	Hành chính
10022	Văn phòng đại diện Thanh niên	2	2	Hành chính
10070	Ban quản lý vốn sinh viên	5	5	Hạch toán đ.lập
10071	Lực lượng quản lý thị trường	5	5	Hạch toán đ.lập

3.9.2. Phép kết nối trái (Left Join)

Giả sử có 2 quan hệ R (A₁, A₂, ..., A_n) và S (B₁, B₂, ..., B_m).

t = (a₁, a₂, ..., a_n) và u = (b₁, b₂, ..., b_m) là hai bộ giá trị của R và S. Gọi v là bộ ghép nối u vào t (hay bộ giá trị t và u được "xếp cạnh nhau") và ký hiệu là:

$v = (t, u) = (a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m).$

Bộ $t_{\text{NULL}} = (\text{NULL}, \text{NULL}, \dots, \text{NULL})$ là một bộ đặc biệt của R gồm n giá trị của các thuộc tính $A_1, A_2, \dots, A_n$ đều là không xác định và $u_{\text{NULL}} = (\text{NULL}, \text{NULL}, \dots, \text{NULL})$ là một bộ đặc biệt của S gồm m giá trị của các thuộc tính $B_1, B_2, \dots, B_m$ đều là không xác định.

$A \text{ Y } R^+$ và $B \text{ Y } S^+$ là hai thuộc tính có thể so sánh được.

Phép *kết nối trái* hai quan hệ R với S trên các thuộc tính A và B với phép so sánh bằng ($=$), với giả thiết là giá trị cột R[A] có thể so sánh tương đương được với mỗi giá trị của cột S[B], được định nghĩa là:

$$R \overset{A=B}{\bowtie} S = \{ v = (t, u) \mid (t \text{ Y } R, u \text{ Y } S \text{ và } t.A = u.B) \text{ hoặc } (t \text{ Y } R, u = u_{\text{NULL}} \text{ với } t.A \text{ S[B]}) \}$$

nghĩa là, tất cả các bộ v có được nhờ cách đặt bộ giá trị của R và S xếp cạnh nhau, nếu có giá trị giống nhau trên 2 thuộc tính kết nối; và các bộ v có được nhờ cách đặt bộ của R với các bộ NULL của S, nếu không tìm được giá trị tương ứng của thuộc tính kết nối trên quan hệ S.

Ví dụ 3.4.2.1:

Với hai quan hệ R và S cùng các bộ giá trị của chúng đã được cho trong ví dụ 3.4.1, kết quả của phép kết nối trái của R và S là:

$$R \overset{R.A = S.A}{\bowtie} S = Q(A, B, C, A, D, E, F)$$

a_1	b_1	c_1	a_1	d_1	e_1	f_1
a_2	b_2	c_2	a_2	d_2	e_2	f_2
a_3	b_3	c_3	-	-	-	-
a_5	b_5	c_5	-	-	-	-
a_7	b_7	c_7	a_7	d_7	e_7	f_7

Ký hiệu dấu trừ (-) trong các thuộc tính của S được hiểu là giá trị không xác định (*giá trị Null*).

Các dòng có giá trị thuộc tính A của R là a_3 và a_5 không tìm được giá trị của thuộc tính A tương ứng trong quan hệ S, nên phần còn lại của nó được để là không xác định. Qua bảng kết quả trình bày trên, chúng ta thấy ý nghĩa của phép toán này là nhằm xác định các bộ giá trị của quan hệ bên trái nhưng không có bộ giá trị tương ứng trong quan hệ phía bên phải.

Ví dụ 3.4.2.2:

Giả sử chúng ta thêm một bộ mới cho bảng ĐƠN-VỊ gồm có Mã-ĐV là 10090, Tên-đơn-vị là Hội khuyến nông Quận X và loại hình đơn vị là 7. Khi đó phép kết nối trái (*Left Join*) hai quan hệ ĐƠN-VỊ và LOẠI-ĐV cho kết quả là quan hệ có thể hiện như sau:

Mã-ĐV	Tên-đơn-vị	Loại	Loại	Tên-loại-hình
10002	Trường K.thuật nghiệp vụ máy tính	3	3	SN hoàn toàn
10003	Phòng quản lý hành chính	2	2	Hành chính
10022	Văn phòng đại diện Thanh niên	2	2	Hành chính
10070	Ban quản lý vốn sinh viên	5	5	Hạch toán đ.lập
10071	Lực lượng quản lý thị trường	5	5	Hạch toán đ.lập
10090	Hội khuyến nông Huyện X	7	Null	Null

Bởi vì trong quan hệ LOẠI-ĐV không có bộ nào có giá trị ở cột *Loại* là 7, do đó ở dòng cuối cùng của quan hệ trên, một bộ gồm toàn giá trị NULL ở cả hai cột *Loại* và *Tên-loại-hình* được đặt cạnh bộ giá trị mới được bổ sung trong thể hiện của quan hệ ĐƠN-VỊ.

Ứng dụng của phép kết nối này có thể thấy rõ trong bài toán quản lý CBVC nêu trên: Trước hết cần xác định những đơn vị có loại hình tổ chức không có trong danh mục Loại-ĐV. Câu trả lời rất đơn giản: chỉ việc chọn ra các dòng của bảng kết nối có giá trị NULL ở cột *Loại* trong phần đuôi của bộ giá trị là NULL.

Cũng trong bài toán quản lý CBVC nêu trên: Theo yêu cầu quản lý, mỗi CBVC có tên trong danh sách của đơn vị phải có một bản khai lý lịch, tức là một bộ giá trị về lý lịch chi tiết. Mỗi CBVC có một mã số CBVC để phân biệt và nhận dạng đồng thời khóa của quan hệ lý lịch cũng là mã số này. Có thể ứng dụng phép kết nối trái để xác định những CBVC nào có tên trong danh sách mà chưa có lý lịch trong CSDL.

3.9.3. Phép kết nối phải (Right Join)

Vẫn với các quan hệ R, S; các thuộc tính A, B; và các bộ giá trị $v, t, u, t_{\text{NULL}}, u_{\text{NULL}}$ được xác định như trên.

Phép kết nối phải hai quan hệ R với S trên các thuộc tính A và B với phép so sánh =, với giả thiết là giá trị cột R[A] có thể so sánh tương đương được với mỗi giá trị của cột S[B], được định nghĩa là:

$$R \bowtie S = \{ v = (t, u) \mid (t \in R, u \in S \text{ và } t.A = u.B) \text{ hoặc } (t = t_{\text{NULL}}, u \in S, \text{ với } t.A = S[B]) \}$$

nghĩa là, tất cả các bộ v có được nhờ cách đặt bộ giá trị của R và S xếp cạnh nhau nếu chúng có giá trị giống nhau trên 2 thuộc tính kết nối, và các bộ NULL của R với các bộ của S, nếu không tìm được giá trị tương ứng của thuộc tính kết nối trên quan hệ R.

Ví dụ 3.4.3.1:

Giả sử với các quan hệ R và S cùng các bộ giá trị của chúng được xác định như trong ví dụ 5.4.2 nêu trên. Kết quả của phép kết nối phải R với S là quan hệ với các bộ giá trị sau:

$$R \bowtie S = Q(A, B, C, A, D, E, F)$$

a_1	b_1	c_1	a_1	d_1	e_1	f_1
a_2	b_2	c_2	a_2	d_2	e_2	f_2
-	-	-	a_4	d_4	e_4	f_4
-	-	-	a_6	d_6	e_6	f_6
a_7	b_7	c_7	a_7	d_7	e_7	f_7

Ký hiệu dấu trừ (-) trong các thuộc tính của R được hiểu là giá trị không xác định (*giá trị Null*).

Các dòng có giá trị tại thuộc tính A của S là a_4 và a_6 không tìm được giá trị của thuộc tính A tương ứng trong quan hệ R, do đó phần đầu của nó được để là không xác định. Qua bảng kết quả trình bày trên, chúng ta thấy ý nghĩa của phép toán này là nhằm xác định các bộ giá trị của quan hệ bên phải không có bộ giá trị tương ứng trong quan hệ phía bên trái.

Ví dụ 3.4.3.2:

Giả sử chúng ta thêm một bộ mới cho quan hệ LOẠI-ĐV gồm có mã *Loại* là 6, *Tên-loại-hình* là Cơ quan Đảng / Đoàn. Khi đó phép kết nối phải (*Right Join*) hai quan hệ ĐƠN-VỊ và LOẠI-ĐV cho kết quả là quan hệ có thể hiện như sau:

<i>Mã-ĐV</i>	<i>Tên-đơn-vị</i>	<i>Loại</i>	<i>Loại</i>	<i>Tên-loại-hình</i>
10002	Trường K.thuật nghiệp vụ máy tính	3	3	SN hoàn toàn
10003	Phòng quản lý hành chính	2	2	Hành chính
10022	Văn phòng đại diện Thanh niên	2	2	Hành chính
10070	Ban quản lý vốn sinh viên	5	5	Hạch toán đ.lập
10071	Lực lượng quản lý thị trường	5	5	Hạch toán đ.lập
10090	Hội khuyến nông Huyện X	7	Null	Null
Null	Null	Null	6	Cơ quan Đảng

Vì không có đơn vị nào có loại hình tổ chức đơn vị là 6, nên ở dòng cuối cùng trong bảng trên, cả 3 cột thuộc phần của quan hệ ĐƠN-VỊ đã bị để trống bằng các giá trị Null không xác định.

Một ứng dụng của phép kết nối này là: xác định những CBVC nào có lý lịch trong CSDL (do nhập sai mã số CBVC) nhưng không có tên trong danh sách (*tức là các lý lịch vô chủ*). Hoặc: xác định các loại hình đơn vị (*Loại*) mà không có đơn vị nào thuộc vào.

Trong một số ngôn ngữ truy vấn CSDL (MicroSoft SQL-Server, Oracle ...) người ta gọi 2 phép toán kết nối trái (*Left Join*) và kết nối phải (*Right Join*) bằng chung một từ là "phép kết nối ngoài" (*Outer Join*). Mục đích của phép kết nối này là cho phép giữ lại tất cả các bản ghi (hoặc bộ giá trị) của 2 quan hệ có cùng giá trị của các thuộc tính kết nối và những bộ giá trị của cả 2 quan hệ không tìm được bộ giá trị giống nhau trên các thuộc tính kết nối thuộc quan hệ đối ứng.

❖ TÓM TẮT CHƯƠNG 2

Trong chương này, một số nội dung chính được giới thiệu:

- Mô hình dữ liệu quan hệ, các khái niệm, cơ sở lý thuyết của mô hình dữ liệu quan hệ (đại số quan hệ).
- Chuyển đổi được hình ERD sang mô hình dữ liệu quan hệ (ở dạng lược đồ).

❖ CÂU HỎI VÀ BÀI TẬP CHƯƠNG 2

Cho CSDL EMPLOYMENT về quản lý nhân viên của một công ty bao gồm quan hệ EMPLOYEE và các quan hệ khác DEPARTMENT, JOBS và EMPLHIST. Quan hệ DEPARTMENT (*phòng, ban*) định nghĩa trên các thuộc tính: DeptNo (*Mã phòng*), DeptName (*Tên phòng*), Loc (*Địa điểm*), Mgr (*Mã NV phụ trách phòng*), Exp_budg (*Ngân sách | kinh phí hoạt động*), Rev_budg (*Doanh thu*) với các bộ giá trị sau:

<i>DeptNo</i>	<i>DeptName</i>	<i>Loc</i>	<i>Mgr</i>	<i>Exp_budg</i>	<i>Rev_budg</i>
10	Accounting	Dallas	200	10.000	
30	Research	San Fransisco	105	125.000	

40	Sales	Boston	109	280.000	800.000
50	Manufacturing	Houston	210	130.000	
60	Shipping	Houston	215	90.000	

Và quan hệ JOBS (các chức danh công việc) định nghĩa trên các thuộc tính: Job (Mã chức danh), JobName (Tên gọi chức danh), MinSalary (mức lương tối thiểu), MaxSalary (mức lương tối đa) MgrFlag (Có thể tiến cử làm lãnh đạo ?) với các bộ giá trị sau:

Job	JobName	MinSalary	MaxSalary	MgrFlag
Mngr	Manager	2.500	5.500	Y
Clrk	Clerk	950	1.800	N
Slsa	Sales Assist	950	2.000	N
Amgr	Assist Manager	1.500	3.000	Y
Drvr	Driver	1.050	1.700	N
Slsn	Salesman	750	5.000	N
Spvr	Supervisor	1.500	2.000	Y
Anlt	Analyst	1.300	3.500	N
Secy	Secretary	800	2.200	N
Newh	New Hire	500	800	N

Và cuối cùng là quan hệ EMPLHIST về quá trình thay đổi lương, phòng làm việc, thăng quan tiến chức của các nhân viên định nghĩa trên các thuộc tính EmpNo (Mã số nhân viên), Seq (Số thứ tự lần thay đổi), Date_beg, Date_end, Salary, Frjob, Tojob, Frdept, Todept, Promo với các bộ giá trị được cho như sau:

EmpNo	Seq	Date_Beg	Date_End	Salary	FrJob	ToJob	Promo	FrDept	ToDept
101	1	01/03/81	30/06/81	800	Newh	Slsa	N		40
101	2	01/07/81	30/06/82	1125	Slsa	Slsa	N	40	40
101	3	01/07/82	30/06/84	1900	Slsa	Slsn	Y	40	40
101	4	01/07/84		2500	Slsn	Slsn	N	40	40
109	1	01/04/80	30/09/80	725	Newh	Slsa	N		40
109	2	01/10/80	30/06/82	1500	Slsa	Slsn	Y	40	40
109	3	01/07/82	31/03/83	2600	Slsn	Amgr	Y	40	40

109	4	01/04/83	30/06/85	3200	Amgr	Mngr	Y	40	40
109	5	01/07/85		3800	Mngr	Mngr	N	40	40
220	1	01/05/82	31/08/82	1900	Newh	Mngr	N		60
220	2	01/09/82	30/06/83	1600	Mngr	Slsm	N	60	40
220	3	01/07/83	31/12/83	2200	Slsm	Slsm	N	40	40
220	4	01/01/84	30/06/84	3400	Slsm	Slsm	N	40	40
220	5	01/07/84		4250	Slsm	Slsm	N	40	40

Hãy biểu diễn các câu hỏi sau bằng ngôn ngữ đại số quan hệ:

Câu 1: Cho biết Mã phòng (*DeptNo*), Tên phòng (*DeptName*) và ngân sách hoạt động (*Exp_budg*) của các phòng ban.

Câu 2: Cho Tên phòng ban (*Deptname*) và địa điểm trụ sở (*Loc*) của các phòng ban.

Câu 3: Cho tên chức danh công việc (*Jobname*) và mức lương tối thiểu (*MinSalary*) và tối đa (*MaxSalary*) của từng chức danh công việc.

Câu 4: Cho danh sách các phòng ban ở thành phố Houston.

Câu 5: Cho danh sách các chức danh công việc có mức lương tối thiểu từ 1.500\$ trở lên.

Câu 6: Cho Tên phòng ban (*DeptName*), Mã số người lãnh đạo (*Mgr*), Ngân sách (*Exp_budg*) và Doanh thu (*Rev_Budg*) của các phòng có Doanh thu.

CHƯƠNG 3: NGÔN NGỮ TRUY VẤN DỮ LIỆU

❖ GIỚI THIỆU CHƯƠNG 3

Chương này sẽ trình bày khái quát về ngôn ngữ dữ liệu SQL nhằm cung cấp cho các học viên những kiến thức cấu trúc cú pháp của ngôn ngữ dữ liệu SQL được sử dụng trong một hệ quản trị CSDL.

MỤC TIÊU CHƯƠNG 3

Sau khi học xong bài này, người học có khả năng:

☐ Về kiến thức:

- Cú pháp truy vấn của dữ liệu quan hệ, điển hình là ngôn ngữ truy vấn SQL chuẩn.

☐ Về kỹ năng:

- Thực hiện truy vấn được trên các dữ liệu đã cài đặt.
- Thực hiện các thao tác an toàn với máy tính

☐ Về năng lực tự chủ và trách nhiệm:

- Rèn luyện tính chính xác, sáng tạo, linh hoạt.
- Tuân thủ nội quy, quy định nơi làm việc.

❖ **PHƯƠNG PHÁP GIẢNG DẠY VÀ HỌC TẬP CHƯƠNG 3**

- Đối với người dạy: sử dụng phương pháp giảng dạy tích cực (diễn giảng, vấn đáp, dạy học theo vấn đề); yêu cầu người học thực hiện câu hỏi thảo luận và bài tập chương 3 (cá nhân hoặc nhóm).
- Đối với người học: chủ động đọc trước giáo trình (Chương 3) trước buổi học; hoàn thành đầy đủ câu hỏi thảo luận và bài tập tình huống chương 3 theo cá nhân hoặc nhóm và nộp lại cho người dạy đúng thời gian quy định.

❖ **ĐIỀU KIỆN THỰC HIỆN CHƯƠNG 3**

- **Phòng học chuyên môn hóa/nhà xưởng:** Phòng máy tính
- **Trang thiết bị máy móc:** Máy chiếu và các thiết bị dạy học khác
- **Học liệu, dụng cụ, nguyên vật liệu:** Bài giảng soạn trên slide môn học, giáo trình, tài liệu tham khảo, giáo án và các tài liệu liên quan.
- **Các điều kiện khác:** Không có

❖ **KIỂM TRA VÀ ĐÁNH GIÁ CHƯƠNG 3**

- **Nội dung:**
 - ✓ **Kiến thức:** Kiểm tra và đánh giá tất cả nội dung đã nêu trong mục tiêu kiến thức
 - ✓ **Kỹ năng:** Đánh giá tất cả nội dung đã nêu trong mục tiêu kỹ năng.
 - ✓ **Năng lực tự chủ và trách nhiệm:** Trong quá trình học tập, người học cần:
 - + Nghiên cứu bài trước khi đến lớp
 - + Chuẩn bị đầy đủ tài liệu học tập.
 - + Tham gia đầy đủ thời lượng môn học.
 - + Nghiêm túc trong quá trình học tập.
- **Phương pháp:**
 - ✓ **Điểm kiểm tra thường xuyên:** không có
 - ✓ **Kiểm tra định kỳ lý thuyết:** 1 điểm kiểm tra (hình thức: viết)

NỘI DUNG CHƯƠNG 3

NGÔN NGỮ TRUY VẤN DỮ LIỆU

1. Mở đầu

Ta thấy một hệ quản trị CSDL (DBMS) phải có ngôn ngữ giao tiếp giữa người sử dụng với CSDL. Ngôn ngữ giao tiếp CSDL gồm các thành phần:

- Ngôn ngữ mô tả dữ liệu (*Data Definition Language - DDL*): cho phép khai báo cấu trúc các bảng của CSDL, khai báo các mối quan hệ của dữ liệu và các quy tắc áp đặt lên các dữ liệu đó.
- Ngôn ngữ thao tác dữ liệu (*Data manipulation Language - DML*): cho phép người sử dụng có thể thêm, xoá, sửa dữ liệu trong CSDL.
- Ngôn ngữ truy vấn dữ liệu hay ngôn ngữ hỏi đáp có cấu trúc (*Structured Query Language - SQL*): cho phép những người khai thác CSDL (chuyên nghiệp hoặc không chuyên) sử dụng để truy vấn các thông tin cần thiết trong CSDL.
- Ngôn ngữ quản lý dữ liệu (*Data Control Language - DCL*): cho phép những người quản trị hệ thống thay đổi cấu trúc của các bảng dữ liệu, khai báo bảo mật thông tin và cấp quyền hạn khai thác CSDL cho người sử dụng.

Những năm 1975-1976, IBM lần đầu tiên đưa ra hệ quản trị CSDL kiểu quan hệ mang tên SYSTEM-R với ngôn ngữ giao tiếp CSDL SEQUEL (Structured English Query language), đó một ngôn ngữ con để thao tác với CSDL.

Năm 1976 ngôn ngữ SEQUEL được cải tiến thành SEQUEL2. Khoảng năm 1978-1979 SEQUEL2 được cải tiến và đổi tên thành Ngôn ngữ truy vấn có cấu trúc và cuối năm 1979 hệ quản trị CSDL được cải tiến thành SYSTEM-R.

Năm 1986 Viện Tiêu chuẩn quốc gia Mỹ đã công nhận và chuẩn hoá ngôn ngữ SQL và sau đó Tổ chức Tiêu chuẩn Thế giới cũng đã công nhận ngôn ngữ này. Đó là chuẩn SQL-86.

Tới nay SQL đã qua 3 lần chuẩn hoá lại (1989, 1992, 1996) để mở rộng các phép toán và tăng cường khả năng bảo mật và tính toàn vẹn dữ liệu.

Để trình bày cú pháp các câu lệnh SQL được gọn gàng và dễ hiểu, tài liệu này có đưa ra một số quy ước ký pháp như sau:

- Các biến cú pháp (Syntax Variables), người sử dụng phải điền cụ thể vào khi viết lệnh sẽ được viết bằng chữ thường (lowercase), trong cặp dấu(< >)
- Các thành phần tùy chọn được viết trong cặp dấu ngoặc vuông ([]).
- Việc lựa chọn một trong các khả năng được thể hiện bằng dấu (!).
- Thành phần bắt buộc phải chọn trong danh sách được viết trong cặp dấu móc đậm nét ({ }).

Lệnh SQL có thể được viết trên nhiều dòng và kết thúc lệnh bởi dấu chấm phẩy (;), tuy nhiên từ khoá, tên, hàm, tên thuộc tính, tên bảng, tên đối tượng thì không được phép viết tách xuống hàng. SQL không phân biệt chữ hoa và chữ thường.

Sau đây chúng ta sử dụng CSDL quản lý bán hàng để minh hoạ cho các câu lệnh.

Khachhang(Makh, tenkh, diachi, dienthoai)

Loaihang(Maloai, tenloai)

Hang(mahang, tenhang, dongia, soluongton, maloai)

HoaDon(SoHD, ngayHD, Makh)

ChitietHD(SoHD, mahang, soluong)

2. Tìm thông tin từ các cột của bản – Mệnh đề Select

Câu lệnh SELECT – SQL tìm kiếm là một trong số các câu lệnh SQL cài đặt đầy đủ các phép toán quan hệ dựa trên các từ khoá cơ bản SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY. Đây là câu lệnh được sử dụng phổ biến nhất với mục đích tìm kiếm thông tin trong CSDL quan hệ cú pháp tổng quát của câu lệnh như sau:

SELECT [DISTINCT] <biểu thức 1>, <biểu thức 2>,...

FROM <tên bảng 1>, <tên bảng 2>,...

[WHERE] <điều kiện chọn>

[GROUP BY <tên cột 1>,<tên cột 2>,...]

[HAVING<điều kiện in kết quả>]

[ORDER BY <tên cột 1> | <biểu thức số 1> [ASC| DESC],...];

Chúng ta sẽ lần lượt làm rõ từng phần của cú pháp ngôn ngữ.

Mệnh đề SELECT

SELECT [DISTINCT] {*| <biểu thức 1> [AS<tên mới 1>], <biểu thức 2> [AS <tên mới 2>],...}

FROM <tên bảng>;

- Cho biết tên các khách hàng của cửa hàng

Select TENKH

From KHACHHANG;

Kết quả là hiển thị ra một cột tên các khách hàng

- Nếu không muốn lấy tên các khách hàng trùng nhau thì dùng từ khoá DISTINCT.

Select Distinct TENKH

From KHACHHANG;

- Muốn hiển thị hết tất cả các cột của bảng dùng ký tự đại diện “*”

*Select **

From KHACHHANG;

Kết quả là dữ liệu của tất cả các cột trong bảng KHACHHANG

- Có thể dùng các phép toán số học +, -, *, /, ^, %, các hàm tính toán đối với các cột kiểu số.

```
Select MAHANG, SOLUONGTON*10
```

```
From HANG;
```

Kết quả là số lượng hàng còn trong kho của mỗi mặt hàng được nhân với 10

- Có thể thay đổi tên của các cột trong bảng kết quả ta dùng từ khoá AS

```
Select MAKH AS ma_so_khach_hang
```

```
From KHACHHANG;
```

Kết quả là hiển thị ra bảng gồm có một cột có tên là **ma_so_khach_hang**

Nhận xét: Sau từ khoá SELECT ta còn có thể có từ khoá TOP *n*. Điều này cho phép chúng ta chỉ hiển thị *n* hàng trong bảng kết quả. Thông thường khi dùng TOP thì thường kết hợp với mệnh đề sắp xếp ORDER BY.

Ví dụ: Đưa ra 3 MAHANG đầu tiên trong danh sách.

```
Select TOP 3 MAHANG
```

```
From HANG;
```

3. Chọn các dòng của bản – Mệnh đề Where

WHERE <điều kiện chọn>

Các bản ghi thoả mãn <điều kiện chọn > mới được thể hiện trong bảng kết quả.

Điều kiện chọn có thể chứa các phép toán And, Or, Between, Not Between, like, =, <, <=, >, >=.

Ta có thể sử dụng các ký tự thay thế: % thay thế cho một chuỗi ký tự

_ thay thế cho một ký tự bất buộc

‘HaNoi%’ thay thế cho tất cả các chuỗi bắt đầu bằng từ ‘HaNoi’

‘_ _ _’ thay thế cho tất cả các chuỗi có ba ký tự

Chú ý: Trong SQL hằng ký tự được bao bởi cặp dấu nháy đơn. Trong Access dấu * thay thế cho một nhóm ký tự, dấu ? thay thế cho một ký tự, hằng ký tự là cặp dấu nháy kép “”, hằng ngày tháng là cặp dấu # #.

- Hiển thị các mặt hàng có số lượng tồn lớn hơn 100

```
Select *
```

```
From HANG
```

```
Where SOLUONGTON>100;
```

- Cho hiển thị các khách hàng ở địa chỉ bắt đầu bằng chữ H

*Select **

From KHACHHANG

Where diachi like "H";*

4. Mệnh đề From

FROM <bảng1, bảng2,...>

Dùng xác định các bảng cần có trong câu lệnh

- Cho biết các thông tin về khách hàng của các hoá đơn ngày 20/11/2005

*Select KHACHHANG.**

From KHACHHANG, HOADON

Where NGAYHD=#20/11/2005# and HOADON.MAKH=KHACHHANG.MAKH;

- Nếu có 2 cột giống nhau trên nhiều bảng, ta cần dùng tên bảng và dấu chấm (.) để phân biệt.

- Ta có thể gán bí danh cho các bảng để cho câu lệnh đơn giản hơn. Sau khi gán ta sử dụng bí danh thay cho tên của bảng

*Select KH.**

From KHACHHANG KH, HOADON HD

Where NGAYHD=#20/11/2005# and HD.MAKH=KH.MAKH;

5. Sắp xếp các dòng của bảng – Mệnh đề Order by

ORDER BY <tên cột> | <biểu thức> [ASC | DESC], <tên cột> | <biểu thức> [ASC | DESC],...

Biểu thức phải có giá trị số: nó thể hiện số thứ tự của cột trong bảng kết quả được chỉ định sắp xếp thứ tự thay vì phải chỉ rõ tên cột, hơn nữa nếu cột kết quả là cột tính toán thì nó chưa có tên nên các sử dụng biểu thức là một biện pháp thay thế hữu dụng. Có thể sắp xếp theo thứ tự tăng dần (với từ khoá ASC - viết tắt của ASCending - mặc định là ASC) hoặc giảm dần (Descending) theo giá trị cột. Trước hết các bản ghi được xếp theo thứ tự của cột thứ nhất; các bản ghi có cùng giá trị ở cả 2 cột 1 và 2 sẽ được xếp theo cột thứ 3.

- Cho biết các mặt hàng sắp xếp tăng theo số lượng tồn kho

Select MAHANG, TENHANG, SOLUONGTON

From HANG

Order By SOLUONGTON;

- Cho biết các khách hàng được sắp xếp theo địa chỉ, nếu cùng địa chỉ thì giảm theo tên

*Select **

From KHACHHANG

Order By DIACHI Asc, TENKH Desc;

6. Truy vấn thông tin từ nhiều bảng

Việc thực hiện các câu truy vấn trên nhiều bảng, về bản chất là giống như trên một bảng, tức là chỉ cần chỉ ra thông tin gì cần tìm và lấy từ các nguồn dữ liệu nào. Các bảng dữ liệu nguồn này cần chỉ ra trong mệnh đề FROM trong câu lệnh SELECT.

Nếu các bảng dữ liệu nguồn có các tên thuộc tính giống nhau thì tên thuộc tính này phải được viết tường minh trong biểu thức tìm kiếm với tên bảng đi kèm phía trước. Nói chung trong một CSDL quan hệ, các bảng thường có các mối quan hệ với nhau. Các bảng được liên hệ với nhau qua phép kết nối của mệnh đề FROM hoặc thông qua điều kiện của mệnh đề WHERE của câu lệnh SELECT. Nếu không thể hiện mối quan hệ này, kết quả sẽ là bảng tích Đề các của bảng 2.

6.1 Kết nối tự nhiên

Select ...

From ...

Where <điều kiện kết nối>...

- Cho biết tên các khách hàng mua hàng trong ngày 20/12/2005

Select KH.TENKHACH

From KHACHHANG KH, HOADON HD

Where KH.MAKH=HD.MAKH and NGAYHD=#20/12/2005#;

- Ta có thể sử dụng phép kết nối nội **Inner join** để viết lại câu lệnh trên

Select KHACHHANG.TENKHACH

From KHACHHANG Inner Join HOADON on

KHACHHANG.MAKH=HOADON.MAKH

Where NGAYHD=#20/12/2005#;

6.2 Kết nối ngoại (Outer join)

Kết nối ngoại gồm 2 loại, kết nối trái (Left Outer Join) và kết nối phải (Right Outer Join)

Cho biết các thông tin về khách hàng và các đơn mua hàng của họ nếu có

Select KHACHHANG., HOADON.**

From KHACHHANG Left Outer Join HOADON On

KHACHHANG.MAKH=HOADON.MAKH

6.3 Truy vấn lồng nhau (Query with SubQuery)

Một truy vấn lồng vào một truy vấn khác gọi là Subquery, Subquery cũng bao gồm các mệnh đề cơ bản như Query và có thể lồng nhau nhiều mức. Subquery được bao bởi hai dấu ngoặc và lồng vào truy vấn tại mệnh đề Where hoặc Having.

Có hai loại truy vấn lồng nhau:

- Truy vấn lồng nhau phân cấp: Mức cao hơn chỉ nhận kết quả của mức thấp. Khi thực hiện, các truy vấn cấp thấp hơn sẽ định trị trước một lần rồi cung cấp kết quả cho truy vấn cấp cao hơn.

- Truy vấn lồng nhau tương quan: Mỗi một tính toán của truy vấn mức cao hơn có tham chiếu đến các truy vấn mức thấp hơn, mỗi lần tham chiếu như vậy các truy vấn mức thấp hơn phải định trị lại.

Cho biết đầy đủ thông tin về những mặt hàng có tồn kho lớn nhất

*Select **

From HANG

Where SOLUONGTON=(Select Max(SOLUONGTON)

From HANG);

☞ Truy vấn con thực hiện trước và tìm ra số lượng hàng tồn lớn nhất, sau đó làm điều kiện cho truy vấn ngoài để liệt kê những mặt hàng có số lượng tồn bằng với số lượng tồn lớn nhất.

Cho biết n mặt hàng có tồn kho lớn nhất

*Select **

From HANG H

Where (Select count()*

From HANG

Where SOLUONGTON>H.SOLUONGTON)<n;

☞ Với mỗi mặt hàng của truy vấn ngoài, truy vấn con bên trong sẽ đếm các mặt hàng có số lượng tồn lớn hơn mặt hàng đó, nếu có ít hơn n mặt hàng có số lượng tồn lớn hơn chúng thì có nghĩa là nó nằm trong n mặt hàng lớn nhất.

Tương tự cho biết n mặt hàng có tồn kho bé nhất

*Select **

From HANG H

Where (Select count()*

From HANG

Where SOLUONGTON<H.SOLUONGTON)<n;

Các phép toán có thể dùng đối với truy vấn lồng nhau

Phép toán tập hợp In, Not in

Để xem một bản ghi có thuộc một bảng hay không ta dùng Subquery với toán tử **In** hoặc **Not In**.

Cho biết các khách hàng ở Hatay mua hàng trong tháng 12/2004

*Select **

From KHACHHANG

Where DIACHI like “Hatay” and

MAKH in (Select MAKH

From HOADON

Where month(NGAYHD)=12 and year(NGAYHD)=2004);

Cho biết các mặt hàng chưa từng được bán

*Select **

From HANG

Where MAHANG Not in (Select MAHANG

From ChitietHD);

Phép so sánh tập hợp

<some, <=some, >some, >=some, =some, <>some Tương đương với:

<any, <=any, >any, >=any, =any, <>any

<all, <=all, >all, >=all, =all, <>all

Chú ý: =some tương đương với **In** nhưng <>some không tương đương với **Not In**, <>all tương đương với **Not In**

- Liệt kê các mặt hàng không phải là mặt hàng có tồn kho lớn nhất

*Select **

From HANG

Where SOLUONGTON<some (Select SOLUONGTON

From HANG);

- Cho biết số lượng trung bình một lần đặt hàng của một mặt hàng

Select MAHANG, Avg(SOLUONGBAN)

From ChitietHD

Group By MAHANG;

- Bây giờ ta muốn biết mặt hàng có số lượng đặt hàng trung bình lớn nhất. Dễ dàng có thể nghĩ đến cách dùng Max(Avg(SOLUONGBAN)) nhưng trong SQL không cho phép các hàm thống kê lồng nhau. Cách giải quyết là:

Select MAHANG, Avg(SOLUONGBAN)

From ChitietHD

Group By MAHANG

Having Avg(SOLUONGBAN) >= All (Select Avg(SOLUONGBAN)

From ChitietHD

Group By MAHANG);

Phép toán kiểm tra bảng rỗng

Exists(Q)= True nếu có ít nhất một bản ghi trong Q

= false nếu ngược lại

Not Exists(Q)= True Q không có bộ nào

= false nếu ngược lại

- Cho biết thông tin về các mặt hàng được bán trong tháng 2/2005

*Select H.**

From HANG H

*Where Exists (Select **

From HOADON D, ChitietHD C

Where year(NGAYHD)=2005 And month(NGAYHD)=2 and D.SOHD=C.SOHD and C.MAHANG=H.MAHANG);

Kiểm tra các bản ghi trùng nhau

Unique(Q) = True nếu Q không có các bộ trùng nhau

= False nếu ngược lại

Not Unique(Q) = True nếu Q có các bộ trùng nhau

= False nếu ngược lại

- Tìm các khách hàng chỉ mua hàng một lần

*Select **

From KHACHHANG K

Where Unique (Select MAKHACH

From HOADON H

Where K.MAKHACH=H.MAKHACH);

- Tìm các khách hàng có ít nhất hai lần mua hàng

*Select **

From KHACHHANG K

Where Not Unique (Select MAKHACH

From HOADON H

Where K.MAKHACH=H.MAKHACH);

7. Gom nhóm dữ liệu – mệnh đề Group by

GROUP BY <tên cột 1>,<tên cột 2>,...

[HAVING <điều kiện>]

- Nhóm các bản ghi có giá trị giống nhau trên các cột chỉ định.

- HAVING theo sau GROUP BY dùng để kiểm tra điều kiện nhóm. Nhóm nào thoả mãn điều kiện sau HAVING thì mới được hiển thị. Lưu ý , dữ liệu được nhóm trước rồi mới kiểm tra điều kiện sau HAVING

Cho biết các khách hàng có nhiều hơn 15 lần mua hàng

Select MAKH, count(MAKH)

From HOADON

Group By MAKH

Having count(MAKH)>15;

- Chú ý: nếu trong một câu lệnh vừa có điều kiện Where và Having thì điều kiện sau Where được xử lý trước. Chỉ có những bộ nào thoả mãn điều kiện Where mới được nhóm và sau khi nhóm xong, mỗi nhóm lại kiểm tra điều kiện sau Having.

Cho biết các khách hàng có nhiều hơn 10 lần mua hàng trong tháng 2 năm 2005

Select MAKH, count(MAKH)

From HOADON

Where NGAYDH Between #01/02/2005# And #29/02/2005#

Group By MAKH

Having count(MAKH)>10;

Chú ý: Chỉ có các cột phân nhóm mới được thể hiện trên mệnh đề Select.

Tính tổng số lượng hàng của mỗi hoá đơn bán

Select SOHD, sum(SOLUONGBAN) as So_luong_hang_ban

From chitietHD

Group By SOHD;

- Khi phân nhóm dữ liệu ta có thể sử dụng các hàm phân nhóm để tính toán trên mỗi nhóm như count, sum, avg, max, min,...

8. Các hàm thao tác dữ liệu

8.1 Các hàm tính toán trên nhóm các bản ghi

Qua ví dụ trên, chúng ta đã nhận thấy sự cần thiết của những tính toán trong câu lệnh SELECT. SQL cung cấp một số hàm xây dựng sẵn (Built- in) làm việc trên nhóm theo kỹ thuật tính toán nhanh tiên tiến RushMore.

COUNT (*| <tên cột> - đếm số bản ghi có giá trị xác định tại cột được cho bởi <tên cột>

SUM (<biểu thức>) – tìm tổng giá trị các biểu thức

MIN (<biểu thức>) – tìm giá trị nhỏ nhất

MAX (<biểu thức>) – tìm giá trị lớn nhất

AVG (<biểu thức>) – tính giá trị trung bình của biểu thức dựa trên các bản ghi của các nhóm.

Các hàm này thường phải được đi kèm với mệnh đề GROUP BY để thực hiện phân nhóm các bản ghi theo giá trị các cột nào đó trước khi tính toán. Nếu không có mệnh đề GROUP BY thì câu lệnh sẽ coi toàn bộ các bản ghi là một nhóm.

- Cho biết số lượng tồn lớn nhất và nhỏ nhất của mỗi mặt hàng

Select MAHANG, TENHANG, Max(SOLUONGTON), Min(SOLUONGTON)

From HANG;

- Đưa ra số lượng lớn nhất của một mặt hàng trong mỗi đơn hàng.

Select SOHD, Max(SOLUONGBAN) as [So luong Max]

From ChitietHD

Group By SOHD;

Chú ý: các hàm SUM, MAX, MIN, AVG chỉ áp dụng với biểu thức kiểu số

8.2 Các hàm tính toán trên bản ghi.

Các hàm toán học.

ASB (x) Trị tuyệt đối của x.

SQRT (x) Căn bậc hai của x (Access và SQL – Server là: SQR (x))

LOG (x) Logarit tự nhiên của x.

EXP (x) Hàm mũ cơ số e của x:

SING (x) Lấy dấu của số x (-1 : x < 0, 0: x = 0, +1: x > 0)

ROUND (x,n) Làm tròn tới n số lẻ (Access và SQL – Server, RND (x))

.....và các hàm lượng giác: SIN, COS, TAN, ASIN, ACOS, ATAN...

Các hàm xử lý chuỗi ký tự.

LEN (str) Cho chiều dài dãy ký tự

LEFT (str, n) Lấy n ký tự về phía trái của dãy str

RIGHT (str, n) Lấy n ký tự về phía phải của dãy str

MID (str, p, n) Lấy n ký tự của dãy str kể từ vị trí p trong dãy

Các hàm xử lý ngày tháng và thời gian.

DATE () Cho ngày tháng năm hiện tại (oracle: SYSDATE)

DAY (dd) Cho số thứ tự ngày trong tháng của biểu thức ngày dd

MONTH (dd) Cho số thứ tự tháng trong năm của biểu thức ngày dd

YEAR (dd) Cho năm của biểu thức ngày dd

HOURL (tt) Cho giờ trong ngày (0- 23)

MINUTE (tt) Cho số phút của thời gian tt

SECONDS (tt) Cho số giây của biểu thức giờ tt.

Các hàm chuyển đổi kiểu giá trị.

FORMAT (biểu thức, mẫu): Đổi biểu thức có kiểu bất kỳ thành chuỗi theo mẫu đã cho trong tham số thứ 2. Có thể sử dụng hàm STR để thay thế.

Họ các hàm chuyển đổi biểu thức có kiểu bất kỳ thành một giá trị thuộc kiểu xác định: CSTR, CINT, CLNG, CSIN, CDBL,...

Cú pháp và ngữ nghĩa cụ thể của các hãng cung cấp phần mềm. Tài liệu này không có tham vọng trình bày chi tiết các hàm của ngôn ngữ hệ quản trị CSDL cụ thể.

Trong chương này, một số nội dung chính được giới thiệu:

- Truy vấn của dữ liệu quan hệ, điển hình là ngôn ngữ truy vấn SQL chuẩn.
- Thực hiện truy vấn được trên các dữ liệu đã cài đặt.

❖ CÂU HỎI VÀ BÀI TẬP CHƯƠNG 3

Bài tập 1: Cho cơ sở dữ liệu dùng để quản lý các chuyến đi của một công ty du lịch

1. DIADIEM(MADD, TENDD)

Mỗi một địa điểm có một mã số(MADD) dùng để phân biệt với các địa điểm khác và có một tên (TENDD)

2. XE(BIENSO, KHTD)

Mỗi một xe có một biển số duy nhất(BIENSO) để phân biệt với các xe khác và có số lượng khách tối đa mà xe đó có thể chở(KHTD)

3. HUONGDV(MAHDV, HTHDV, DCHDV)

Mỗi một hướng dẫn viên của công ty có một mã số duy nhất để phân biệt(MAHDV), có họ tên(HTHDV) và địa chỉ của hướng dẫn viên(DCHDV)

4. CHUYENDI(MACD, TENCD, NGKH, NGKT, KHDK)

Mỗi một chuyến đi có một mã số để phân biệt(MACD), thông tin về chuyến đi bao gồm: tên chuyến đi(TENCD), ngày khởi hành(NGKH), ngày kết thúc(NGKT) và số khách dự kiến(KHDK).

5. CTIETCD(MACD, MADD, SNLUU)

Chi tiết của chuyến đi (MACD) là các địa điểm mà chuyến đi đó đi qua (MADD), (SNLUU) là số ngày lưu lại tại điểm du lịch đó.

6.HUONGDAN(MACD, MAHDV)

Ghi nhận các hướng dẫn viên(MAHDV) tham gia hướng dẫn cho chuyến đi (MACD)

7. KHACH(MACD, HTKH, TUOI, DCKH, DTKH)

Ghi nhận thông tin về khách hàng đăng ký vào chuyến đi(MACD), bao gồm: họ tên(HTKH), tuổi (TUOIKH), địa chỉ(DCKH) và điện thoại liên lạc của khách(DTKH)

8. XEPV(MACD, BIENSO)

Ghi nhận các xe (BIENSO) phục vụ cho chuyến đi (MACD)

Dùng câu lệnh SQL để thực hiện các yêu cầu sau:

1. Tạo tất cả các bảng trên.
2. Cho biết danh sách các hướng dẫn viên của công ty.
3. Liệt kê đầy đủ thông tin về các điểm du lịch liên kết với công ty.

4. Cho biết đầy đủ thông tin về các địa điểm mà chuyến đi mã số CD2000 đi qua.
5. Liệt kê các lữ khách của chuyến đi CD2000.
6. Cho biết số lượng khách của chuyến đi CD1999.
7. Chuyến đi nào có số lượng khách lớn hơn số lượng dự kiến.
8. Cho biết tổng số lượng khách của tất cả các chuyến đi có ngày khởi hành trong tháng 12/2001.
9. Cho biết số ngày lưu lại trung bình, số ngày lưu lại lớn nhất, nhỏ nhất qua các điểm du lịch của chuyến đi CD2000.
10. Cho biết số lượng xe phụ vụ cho chuyến đi CD2000.
11. Điểm du lịch nào(Mã số, tên) có số ngày lưu lại lớn nhất của chuyến đi CD2000.
12. Điểm du lịch nào(Mã số, tên) có số ngày lưu lại lớn hơn số ngày lưu lại trung bình qua các điểm của chuyến đi CD2000.
13. Điểm du lịch SaPa(mã số SP) có bao nhiêu chuyến đi ghé qua và khai thác được bao nhiêu ngày(tổng số ngày phục vụ).
14. Liệt kê 3 điểm du lịch đầu tiên của chuyến đi CD2000 có số ngày lưu lại lớn nhất.
15. Liệt kê 3 điểm du lịch đầu tiên của chuyến đi CD2000 có số ngày lưu lại ít nhất.
16. Liệt kê các điểm du lịch của chuyến đi CD2000 ngoại trừ điểm có số ngày lưu ít nhất
17. Cho biết số lượng các điểm du lịch, tổng số ngày lưu lại tại các địa điểm, số lượng các hướng dẫn viên, số lượng xe phụ vụ cho từng chuyến đi có ngày khởi hành trong tháng 12/2000.
18. Chuyến đi nào (đầy đủ thông tin) có số lượng khách nhiều nhất.
19. liệt kê các chuyến đi, ngoại trừ chuyến đi điều động xe ít nhất.
20. liệt kê các chuyến đi (đầy đủ thông tin) cùng với số lượng các địa điểm mà nó ghé qua ngoại trừ chuyến đi ghé ít điểm du lịch nhất.
21. Hướng dẫn viên nào chưa từng tham gia hướng dẫn.
22. Cho biết các hướng dẫn viên hướng dẫn cùng lúc từ hai chuyến trở lên.
23. Cho biết chuyến đi nào mà một hướng dẫn viên hướng dẫn trên 20 khách.
24. Cho biết tất cả các địa điểm cùng với các chuyến đi ghé qua nó nếu có.
25. Điểm du lịch nào mà tất cả các chuyến đi có ngày khởi hành trong năm 2000 đều ghé qua.
26. Hướng dẫn viên nào có tham gia vào tất cả các chuyến đi từ 20 khách trở lên.
27. Chuyến đi nào đã đi ghé qua tất cả các điểm du lịch.

28. Trong năm 2000 điểm du lịch nào đón nhiều khách nhất.

Bài tập 2:

Xét CSDL quản lý công chức viên chức CCVC, gồm các bảng ĐƠN-VỊ, LOẠI-ĐVỊ, NGÁCH-CBVC, NGÁCH-BẬC-LƯƠNG và CBVC ở chương 2. Hãy viết các câu lệnh truy vấn thông tin cho các câu hỏi sau đây:

1. Cho danh sách CBVC theo thứ tự Alphabet của tên của các CBVC.
2. Cho danh sách CBVC có hệ số lương từ 3.0 trở lên.
3. Cho biết tổng hệ số lương của từng đơn vị.
4. Cho danh sách CBVC thuộc các đơn vị mà tên có chữ "phòng".
5. Cho danh sách CBVC thuộc các đơn vị có tên loại hình tổ chức là "hành chính"
6. Cho danh sách CBVC thuộc ngạch "cán sự" có bậc 7 trở lên, hoặc những người có hệ số lương lớn hơn 3.06
7. Cho danh sách CBVC (mà) có thời hạn xếp lương tính đến cuối năm 1998 là 3 năm trở lên đối với các ngạch chuyên viên và chuyên viên chính; hoặc 2 năm trở lên đối với các ngạch còn lại. (*Đây là danh sách CBVC đến hạn nâng lương trong năm 1998*).
8. Cho danh sách các CBVC có hệ số lương cao hơn hệ số lương của những người thuộc ngạch "cán sự".

Bài tập 3:

Cho lược đồ CSDL sau dùng để quản lý việc học tập của sinh viên

1. KHOA(MAKH, VPKH)

Mỗi khoa có 1 mã số phân biệt (MAKH), ta biết được vị trí của văn phòng khoa.

2. LOP(MALOP, MAKH)

Mỗi lớp có 1 mã số để phân biệt (MALOP) thuộc duy nhất một khoa nào đó (MAKH).

3. SINHVIEN(MASV, HSSV, TENSX, NSSV, DCSV,LOPTR, MALOP)

Mỗi sinh viên có một mã số để phân biệt với các sinh viên khác (MASV), thông tin của từng sinh viên là họ và đệm (HOSV), tên (TENSX), năm sinh(NSSV), địa chỉ (DCSX), có phải là lớp trưởng không (LOPTR) và thuộc một lớp duy nhất nào đó (MALOP)

4. MONHOC(MAMH, TENMH, LT, TH)

Mỗi môn học có một mã số duy nhất (MAMH), có một tên (TENMH), số tiết lý thuyết (LT), số tiết thực hành (TH)

5. CTHOC(MALOP, HK, MAMH)

Mỗi lớp học (MALOP) trong từng học kỳ (HK) sẽ có một số môn học (MAMH) được giảng dạy cho lớp đó.

6. DIEMSV(MASV, MAMH, LAN, DIEM)

Ghi nhận điểm của các môn học (MAMH) ở lần thi nào (LAN), của sinh viên(MASV).

Yêu cầu: Viết câu lệnh SQL để thực hiện yêu cầu sau:

1. Cho biết danh sách lớp
2. Cho biết danh sách sinh viên lớp TH1.
3. Cho biết danh sách SV khoa CNTT
4. Cho biết chương trình học của lớp TH1
5. Điểm lần 1 môn CSDL của SV lớp TH1.

6. Điểm trung bình lần 1 môn CTDL của lớp TH1.
7. Số lượng SV của lớp TH2.
8. Lớp TH1 phải học bao nhiêu môn trong HK1 và HK2.
9. Cho biết 3 SV đầu tiên có điểm thi lần 1 cao nhất môn CSDL.
10. Cho biết sĩ số từng lớp.
11. Khoa nào đông SV nhất.
12. Lớp nào đông nhất khoa CNTT.
13. Môn học nào mà ở lần thi 1 có số SV không đạt nhiều nhất.
14. Tìm điểm thi lớn nhất của mỗi SV cho mỗi môn học (vì SV được thi nhiều lần).
15. Điểm trung bình của từng lớp khoa CNTT ở lần thi thứ nhất môn CSDL.
16. Sinh viên nào của lớp TH1 đã thi đạt tất cả các môn học ở lần 1 của HK2.
17. Danh sách SV nhận học bổng học kỳ 2 của lớp TH2, nghĩa là đạt tất cả các môn học của học kỳ này ở lần thi thứ nhất.
18. Biết rằng lớp TH1 đã học đủ 6 học kỳ, cho biết SV nào đủ điều kiện thi tốt nghiệp, nghĩa là đã đạt đủ tất cả các môn.

CHƯƠNG 4: RÀNG BUỘC TOÀN VỆN

❖ GIỚI THIỆU CHƯƠNG 4

Ràng buộc toàn vẹn (Integrity Constraint / Rule viết tắt là: RBTV) và kiểm tra sự vi phạm ràng buộc toàn vẹn là hai trong những vấn đề rất quan trọng trong quá trình phân tích, thiết kế và khai thác CSDL. Trong quá trình phân tích - thiết kế cơ sở dữ liệu, nếu không quan tâm đúng mức đến những vấn đề trên, thì có thể dẫn đến những hậu quả rất nghiêm trọng về tính an toàn và toàn vẹn dữ liệu, đặc biệt trong những CSDL tương đối lớn.

Chương này trình bày các khái niệm cơ bản của ràng buộc toàn vẹn và giới thiệu các cách phân loại ràng buộc toàn vẹn.

MỤC TIÊU CHƯƠNG 4

Sau khi học xong bài này, người học có khả năng:

- ☐ **Về kiến thức:**
 - Trình bày được các thuật ngữ ràng buộc, ràng buộc toàn vẹn, Hiểu rõ các khái niệm về ràng buộc toàn vẹn.
- ☐ **Về kỹ năng:**
 - Phân loại được các loại ràng buộc toàn vẹn.
 - Thực hiện các thao tác an toàn với máy tính
- ☐ **Về năng lực tự chủ và trách nhiệm:**
 - Rèn luyện tính chính xác, sáng tạo, linh hoạt.

- Tuân thủ nội quy, quy định nơi làm việc.

❖ **PHƯƠNG PHÁP GIẢNG DẠY VÀ HỌC TẬP CHƯƠNG 4**

- Đối với người dạy: sử dụng phương pháp giảng dạy tích cực (diễn giảng, vấn đáp, dạy học theo vấn đề); yêu cầu người học thực hiện câu hỏi thảo luận và bài tập chương 4 (cá nhân hoặc nhóm).

- Đối với người học: chủ động đọc trước giáo trình (Chương 4) trước buổi học; hoàn thành đầy đủ câu hỏi thảo luận và bài tập tình huống chương 4 theo cá nhân hoặc nhóm và nộp lại cho người dạy đúng thời gian quy định.

❖ **ĐIỀU KIỆN THỰC HIỆN CHƯƠNG 4**

- **Phòng học chuyên môn hóa/nhà xưởng:** Phòng máy tính
- **Trang thiết bị máy móc:** Máy chiếu và các thiết bị dạy học khác
- **Học liệu, dụng cụ, nguyên vật liệu:** Bài giảng soạn trên slide môn học, giáo trình, tài liệu tham khảo, giáo án và các tài liệu liên quan.
- **Các điều kiện khác:** Không có

❖ **KIỂM TRA VÀ ĐÁNH GIÁ CHƯƠNG 4**

- **Nội dung:**
 - ✓ **Kiến thức:** Kiểm tra và đánh giá tất cả nội dung đã nêu trong mục tiêu kiến thức
 - ✓ **Kỹ năng:** Đánh giá tất cả nội dung đã nêu trong mục tiêu kỹ năng.
 - ✓ **Năng lực tự chủ và trách nhiệm:** Trong quá trình học tập, người học cần:
 - + Nghiên cứu bài trước khi đến lớp
 - + Chuẩn bị đầy đủ tài liệu học tập.
 - + Tham gia đầy đủ thời lượng môn học.
 - + Nghiêm túc trong quá trình học tập.
- **Phương pháp:**
 - ✓ **Điểm kiểm tra thường xuyên:** 1 điểm kiểm tra (hình thức: vấn đáp)
 - ✓ **Kiểm tra định kỳ lý thuyết:** không có

NỘI DUNG CHƯƠNG 4

RÀNG BUỘC TOÀN VỆ

1. Ràng buộc toàn vẹn

1.1. Khái niệm ràng buộc toàn vẹn

Ràng buộc toàn vẹn là một điều kiện bất biến không được vi phạm trong một CSDL.

Trong một CSDL, luôn luôn tồn tại rất nhiều mối liên hệ ảnh hưởng qua lại lẫn nhau giữa các thuộc tính của một quan hệ, giữa các bộ giá trị trong một quan hệ và giữa các thuộc

tính của các bộ giá trị trong các quan hệ với nhau. Các mối quan hệ phụ thuộc lẫn nhau này chính là những điều kiện bất biến mà tất cả các bộ của những quan hệ có liên quan trong cơ sở dữ liệu đều phải thỏa mãn ở bất kỳ thời điểm nào. Ràng buộc toàn vẹn còn được gọi là các quy tắc quản lý (Rules) được áp đặt lên trên các đối tượng của thế giới thực.

Ví dụ :

Trong CSDL về quản lý học viên của một trường học đã cho trong các ví dụ của chương trước, chúng ta có một số ràng buộc toàn vẹn như sau:

R1 : Mỗi lớp học phải có một mã số duy nhất để phân biệt với mọi lớp học khác trong trường.

R2 : Mỗi lớp học phải thuộc một KHOA của trường.

R3 : Mỗi học viên có một mã số riêng biệt, không trùng với bất cứ học viên nào khác.

R4 : Mỗi học viên phải đăng ký vào một lớp của trường.

R5 : Mỗi học viên được thi tối đa 3 lần cho mỗi môn học.

R6 : Tổng số học viên của một lớp phải lớn hơn hoặc bằng số lượng đếm được của lớp tại một thời điểm.

Khóa nội, Khóa ngoại, giá trị NOT NULL ... là những RBTV về miền giá trị của các thuộc tính. Những RBTV vừa nêu trên cũng mới chỉ là những RBTV đơn giản trong CSDL nhỏ về quản lý học viên. Trong thực tế, tất cả các RBTV của một cơ sở dữ liệu phải được người phân tích thiết kế phát hiện đầy đủ và mô tả một cách chính xác, rõ ràng trong hồ sơ phân tích, thiết kế.

Trong một CSDL, ràng buộc toàn vẹn được xem như một công cụ để diễn đạt ngữ nghĩa của cơ sở dữ liệu đó. Trong suốt quá trình khai thác cơ sở dữ liệu, các RBTV đều phải được thỏa mãn ở bất kỳ thời điểm nào nhằm đảm bảo cho CSDL luôn luôn ở trạng thái an toàn và nhất quán về dữ liệu.

Các hệ quản trị CSDL thường có các cơ chế tự động kiểm tra các RBTV về miền giá trị của Khóa nội, Khóa ngoại, giá trị NOT NULL qua khai báo cấu trúc các bảng (mô hình dữ liệu của quan hệ) hoặc thông qua những thủ tục kiểm tra và xử lý vi phạm RBTV do những người phân tích - thiết kế cài đặt. Việc kiểm tra RBTV có thể được tiến hành vào một trong các thời điểm sau:

Kiểm tra ngay khi thực hiện một thao tác cập nhật CSDL (thêm, sửa, xóa). Thao tác cập nhật chỉ được xem là hợp lệ nếu như nó không vi phạm bất cứ một RBTV nào, nghĩa là nó không làm mất tính toàn vẹn dữ liệu của CSDL. Nếu vi phạm RBTV, thao tác cập nhật bị coi là không hợp lệ và sẽ bị hệ thống hủy bỏ (hoặc có một xử lý thích hợp nào đó).

Kiểm tra định kỳ hay đột xuất, nghĩa là việc kiểm tra RBTV được tiến hành một cách độc lập đối với thao tác cập nhật dữ liệu. Đối với những trường hợp vi phạm RBTV, hệ thống sẽ có những xử lý ngầm định hoặc yêu cầu người sử dụng xử lý những sai sót một cách tường minh.

Khi xác định một RBTV cần chỉ rõ:

Điều kiện (tức là nội dung) của RBTV, từ đó xác định cách biểu diễn.

Bối cảnh xảy ra RBTV : trên một hay nhiều quan hệ, cụ thể trên các quan hệ nào.

Tầm ảnh hưởng của RBTV. Khả năng tính toàn vẹn dữ liệu bị vi phạm, và

Hành động cần phải có khi phát hiện có RBTV bị vi phạm.

1.2. Các yếu tố của ràng buộc toàn vẹn

1.2.1 Điều kiện của ràng buộc toàn vẹn

Điều kiện của RBTV là sự mô tả, và biểu diễn hình thức nội dung của nó, có thể được biểu diễn bằng ngôn ngữ tự nhiên, thuật giải (bằng mã giả - Pseudo Code, ngôn ngữ tựa Pascal), ngôn ngữ đại số tập hợp, đại số quan hệ v.v hoặc bằng các phụ thuộc hàm (sẽ được trình bày chi tiết trong mục dưới đây).

Ví dụ :

Giả sử có một CSDL quản lý hóa đơn bán hàng gồm các bảng sau:

HÓAĐƠN (Số-hóa-đơn, Số-chủng-loại-mặt-hàng, Tổng-trị-giá).

DM_HÀNG (Mã-hàng, Tên-hàng, Đơn-vị-tính).

CHITIẾT_HĐ (Số-hóa-đơn, Mã-hàng, Số-lượng-đặt, Đơn-giá, Trị-giá).

Điều kiện của ràng buộc toàn vẹn có thể biểu diễn như sau:

R1 : “Mỗi hóa đơn có một Số hóa đơn riêng biệt, không trùng với hóa đơn khác”:

$\forall hd1, hd2 \in HÓAĐƠN, hd1 \neq hd2 \Rightarrow hd1.Số-hóa-đơn \neq hd2.Số-hóa-đơn$.

R2 : “Số-chủng-loại-mặt-hàng = số bộ của CHITIẾT_HĐ có cùng Số-hóa-đơn”:

$\forall hd \in HÓAĐƠN$ thì:

$hd.Số-chủng-loại-mặt-hàng = COUNT (cthđ \in CHITIẾT_HĐ, cthđ.Số-hóa-đơn = hd.Số-hóa-đơn)$

R3 : “Tổng các trị giá của các mặt hàng trong CHITIẾT_HĐ có cùng Số-hóa-đơn phải bằng Tổng-trị-giá ghi trong HÓAĐƠN”:

$\forall hd \in HÓAĐƠN$ thì:

$hd.Tổng-trị-giá = SUM (cthđ.Tri-giá)$ đối với các $cthđ \in CHITIẾT_HĐ$ sao cho : $cthđ.Số-hóa-đơn = hd.Số-hóa-đơn$.

R4 : “Mỗi bộ của CHITIẾT_HĐ phải có mã hàng thuộc về danh mục hàng”:

$CHITIẾT_HĐ [Mã-hàng] \subseteq DM_HÀNG [Mã-hàng]$

hoặc biểu diễn bằng cách khác:

$\forall cthđ \in CHITIẾT_HĐ, \exists hh \in DM_HÀNG$

sao cho: $cthđ.Mã-hàng = hh.Mã-hàng$.

Chú ý: Chúng ta quy ước: chữ COUNT (...) nghĩa là đếm số bộ giá trị của một quan hệ thỏa mãn điều kiện đã cho trong dấu ngoặc tròn. Chữ SUM (...) nghĩa là lấy tổng các giá trị của các bộ trên một quan hệ thỏa mãn điều kiện đã cho trong dấu ngoặc tròn. Các ngôn ngữ quản trị CSDL đều có các thủ tục hàm này để hỗ trợ tính toán.

1.2.2. Bối cảnh của Ràng buộc toàn vẹn

Bối cảnh có thể định nghĩa trên một quan hệ cơ sở hay nhiều quan hệ cơ sở. Đó là những quan hệ mà RBTV được áp dụng trên đó.

Như trong ví dụ trên mục 1.2, bối cảnh của ràng buộc toàn vẹn R1 chỉ là một quan hệ HÓAĐƠN; bối cảnh của ràng buộc toàn vẹn R2 và R3 là hai quan hệ HÓAĐƠN và CHITIẾT_HĐ; bối cảnh của ràng buộc toàn vẹn R4 là hai quan hệ CHITIẾT_HĐ và DM_HÀNG.

1.2.3. Tầm ảnh hưởng của ràng buộc toàn vẹn

Một RBTV có thể liên quan đến một số quan hệ, và chỉ khi có thao tác cập nhật (Thêm, Sửa, Xóa) mới có nguy cơ dẫn đến vi phạm RBTV, do đó cần xác định rõ thao tác nào dẫn đến việc cần phải kiểm tra RBTV.

Trong quá trình phân tích, thiết kế một CSDL, người phân tích cần lập bảng xác định tầm ảnh hưởng cho mỗi ràng buộc toàn vẹn nhằm xác định khi nào thì phải tiến hành kiểm tra các ràng buộc toàn vẹn đó. Bảng này gồm 4 cột: cột 1 là cột chủ từ chứa tên các quan hệ liên quan tới RBTV; 3 cột tiếp theo là thao tác Thêm / Sửa / Xóa bộ giá trị của quan hệ. Nếu RBTV cần được kiểm tra nguy cơ dẫn tới vi phạm thì tại ô (giao điểm dòng và cột) đó người ta đánh dấu bằng dấu gạch chéo (x) hoặc dấu cộng (+), và có thể chỉ rõ thêm các thuộc tính nào nếu được cập nhật mới dẫn đến vi phạm RBTV bằng cách liệt kê chúng dưới dấu (x) hoặc dấu (+). Nếu RBTV không có nguy cơ bị vi phạm khi cập nhật CSDL thì đánh dấu trừ (-) vào ô tương ứng. Nếu không bị vi phạm vì không được phép sửa đổi thì ký hiệu là trừ với dấu sao (-(*)).

Ví dụ:

Bảng tầm ảnh hưởng của ràng buộc toàn vẹn R₁

Quan hệ	Thêm	Sửa	Xóa
HÓAĐƠN	+ (Số-hóa-đơn)	- (*)	+

Bảng tầm ảnh hưởng của ràng buộc toàn vẹn R₂

Quan hệ	Thêm	Sửa	Xóa
HÓAĐƠN	-	+ (Số-chủng-loại-mặt-hàng)	+
CHITIẾT_H Đ	+	-	+

Bảng tầm ảnh hưởng của ràng buộc toàn vẹn R₃

Quan hệ	Thêm	Sửa	Xóa
HÓAĐƠN	-	+ (Tổng-trị-giá)	+
CHITIẾT_HĐ	+	+ (Trị-giá)	+

Bảng tầm ảnh hưởng của ràng buộc toàn vẹn R₄

Quan hệ	Thêm	Sửa	Xóa
CHITIẾT_HĐ	+ (Mã-hàng)	- (*)	+
DM_HÀNG	-	- (*)	+

Trong thực tế, ràng buộc toàn vẹn R₁ là không cần thiết, bởi thuộc tính Số-hóa-đơn là khóa của quan hệ HÓAĐƠN, do vậy nó luôn luôn phải là duy nhất và không được phép chứa giá trị rỗng; đồng thời không được phép sửa đổi.

Sau khi xây dựng các bảng tầm ảnh hưởng của từng RBTV trên các quan hệ liên quan, cần phải tổng hợp lại bằng cách xây dựng một bảng tầm ảnh hưởng tổng hợp các RBTV nhằm xác định tất cả các RBTV cần phải kiểm tra trên từng quan hệ. Bảng này gồm cột chủ từ là các RBTV, các cột còn lại là các thao tác Thêm (T), Sửa (S) và Xóa (X) của từng quan hệ nằm trong bối cảnh của các RBTV trong CSDL.

Ví dụ:

Lập bảng tầm ảnh hưởng tổng hợp của các RBTV trong CSDL quản lý hóa đơn bán hàng nêu trên:

Q.Hệ	HÓA ĐƠN			CHITIẾT_HĐ			DM_HÀNG		
RBT V	T	S			S				
R ₁	+	- (*)							
	(Số-hd)								
R ₂	-	+(Số-loại-MH)			- (*)				
R ₃	-	+(Tổng-TG)			+(Tri-giá)				
R ₄					-			(*)	

Nhìn vào bảng tổng hợp trên chúng ta có thể thấy quan hệ HÓA ĐƠN khi thêm và xóa một bộ giá trị, phải kiểm tra ràng buộc toàn vẹn R1, R2 và R3; khi sửa giá trị thuộc tính Số-chủng-loại-mặt-hàng thì phải kiểm tra ràng buộc toàn vẹn R2 và khi sửa giá trị thuộc tính Tổng-trị-giá thì phải kiểm tra ràng buộc toàn vẹn R3. Quan hệ CHITIẾT_HĐ khi được cập nhật cần kiểm tra 2 RBTV: R2 và R3; Quan hệ DM_HÀNG cần kiểm tra ràng buộc toàn vẹn R4 khi xóa một bộ giá trị.

12.4. Hành động khi RBTV bị vi phạm

Khi một RBTV bị vi phạm cần có những hành động thích hợp. Thông thường có 2 giải pháp:

(1) Đưa ra thông báo và yêu cầu sửa chữa dữ liệu của các thuộc tính cho phù hợp với quy tắc đảm bảo tính nhất quán dữ liệu. Thông báo phải đầy đủ và tạo được sự thân thiện với người sử dụng. Giải pháp này là phù hợp cho việc xử lý thời gian thực.

(2) Từ chối thao tác cập nhật. Giải pháp này là phù hợp đối với việc xử lý theo lô (Batch processing). Việc từ chối cũng phải được lưu lại bằng những thông báo đầy đủ, rõ ràng vì sao thao tác bị từ chối và cần phải sửa lại những dữ liệu nào.

2. Phân loại ràng buộc toàn vẹn

2.1. Ràng buộc toàn vẹn có bối cảnh là một quan hệ

2.1.1 Ràng buộc toàn vẹn về miền giá trị của thuộc tính.

Trong hầu hết các CSDL quan hệ, loại RBTV này là rất phổ biến. Như chúng ta đã biết, thuộc tính được đặc trưng không chỉ bởi kiểu giá trị mà nó còn bị giới hạn bởi miền giá trị trong kiểu dữ liệu đó. Do đó, khi thực hiện các thao tác cập nhật Thêm, Sửa bộ giá trị mới cho quan hệ đều phải kiểm tra RBTV này.

Ví dụ :

Trong quan hệ KQUẢ-THI mô tả trong chương trước, do quy định mỗi học viên chỉ được thi một môn học tối đa là 3 lần, hiển nhiên là điểm thi của mỗi môn học trong mọi lần thi không bị âm và không vượt quá 10. Có 2 ràng buộc toàn vẹn về miền giá trị trong quan hệ này:

R1: $\forall kq \in KQU\grave{A}\text{-}THI \text{ thì } 0 \leq kq.L\grave{a}n\text{-}thi \leq 3$

R2: $\forall kq \in KQU\grave{A}\text{-THI}$ thì $0 \leq kq.\text{Điểm-thi} \leq 10$

Giả sử các giảng viên có “châm chước” thêm rằng điểm thi lần sau không nhỏ hơn điểm thi lần trước đó. Chúng ta có thêm ràng buộc toàn vẹn về miền giá trị:

R3: $\forall kq \in KQU\grave{A}\text{-THI} \mid kq.\text{Điểm-thi (lần trước)} \leq kq.\text{Điểm-thi} \leq 10.0$

2.1.2. Ràng buộc toàn vẹn liên thuộc tính.

Đó là loại RBTV có liên quan tới nhiều thuộc tính của một quan hệ. Thông thường đó là các phụ thuộc tính toán, hoặc một suy diễn từ giá trị của một hay nhiều thuộc tính trong cùng một bộ giá trị.

Ví dụ 1:

Quan hệ CHITIẾT_HĐ trong CSDL quản lý hóa đơn bán hàng cho trong ví dụ tại tiết 1, có RBTV liên thuộc tính là:

$\forall cthđ \in CHITIẾT_HĐ \mid cthđ.\text{Trị-giá} = cthđ.\text{Số-lượng-đặt} * cthđ.\text{Đơn-giá}.$

Ví dụ 2:

Quan hệ danh sách cán bộ - công chức Nhà nước CBCC với tập các thuộc tính:

{ Mã-đơn-vị, Mã-CBCC, Họ-tên, Giới-tính, Ngày-sinh, Ngày-tuyển-dụng, Ngạch-CBCC, Bậc, Hệ-số-lương, Ngày-xếp-lương }.

Với quy định nam từ 18 đến 60 và nữ từ 18 đến 55 tuổi và phải từ 18 tuổi trở lên mới được tuyển vào làm công chức Nhà nước. Chúng ta có các RBTV về miền giá trị liên thuộc tính như sau:

R1: $\forall cc \in CBCC \mid$ nếu $cc.\text{Giới-tính} = \text{Nam}$ thì $(\text{Now}() - cc.\text{Ngày_sinh}) / 365$ trong khoảng 18 và 60. Nếu $cc.\text{Giới-tính} = \text{Nữ}$ thì $(\text{Now}() - cc.\text{Ngày-sinh}) / 365$ trong khoảng 18 và 55.

R2: $\forall cc \in CBCC \mid (cc.\text{Ngày-tuyen-dung} - cc.\text{Ngày-sinh}) / 365 \geq 18$ và $cc.\text{Ngày-tuyen-dung} \leq \text{Now}()$.

Chú ý:

Now() là lấy ngày tháng năm hiện tại và một năm trung bình có 365 ngày;

Hiệu 2 giá trị ngày tháng là số ngày trôi qua giữa 2 ngày đó.

2.1.3. Ràng buộc toàn vẹn liên bộ, liên thuộc tính.

Đây là loại RBTV có liên quan tới nhiều bộ và có thể tới nhiều thuộc tính của (các) bộ giá trị trong một quan hệ.

Ví dụ 1:

Trong ví dụ trên vừa nêu, chúng ta thấy điểm thi không chỉ liên quan đến thuộc tính Lần-thi mà còn liên quan tới điểm thi của lần thi trước đó nếu đã thi 1 hay 2 lần rồi. RBTV đầy đủ phải được diễn đạt bằng thuật toán như sau:

R3: $\forall kq \in KQU\grave{A}\text{-THI} \mid$ Nếu $kq.\text{Lần-thi} = 1$ thì $0 \leq kq.\text{Điểm-thi} \leq 10.0$

hoặc:

Nếu $kq.\text{Lần thi} > 1$ thì $\exists kq' \in KQU\grave{A}\text{-THI}$

sao cho $kq'.\text{Lần-thi} = kq.\text{Lần-thi} - 1$ và $kq.\text{Điểm-thi} \geq kq'.\text{Điểm-thi}.$

Ví dụ 2:

Giả thiết thêm rằng, để xác định hệ số lương của một cán bộ - công chức Nhà nước căn cứ vào Ngạch công chức và Bậc được xếp, chúng ta có thêm bảng quy định Ngạch, bậc và hệ số lương cán bộ công chức Nhà nước theo Nghị định 25CP của Thủ tướng Chính phủ:

NGẠCH-BẬC-LƯƠNG (Mã-ngạch, Bậc, Hệ-số-lương).

Ứng với một Ngạch công chức và một Bậc cụ thể thì có một hệ số lương tương ứng (từ 1.0 đến 10.0).

Việc biểu diễn ràng buộc toàn vẹn Hệ-số-lương phụ thuộc vào Ngạch và Bậc của quan hệ CBCC nêu trong ví dụ trên bằng thuật giải có thể trở nên rắc rối. Người ta đã đưa thêm một cách biểu diễn mới để làm cho RBTV trở nên rõ ràng hơn, đó là cách biểu diễn RBTV bằng phụ thuộc hàm mà chúng ta sẽ trình bày rõ hơn trong tiết 3 của chương này.

2.2. Ràng buộc toàn vẹn có bối cảnh là nhiều quan hệ

2.2.1 Ràng buộc toàn vẹn về phụ thuộc tồn tại.

Ràng buộc toàn vẹn về phụ thuộc tồn tại (Existential Dependency hay Referential Dependency) còn được gọi là phụ thuộc về khóa ngoại. Đây là loại RBTV khá phổ biến trong các CSDL bởi các quan hệ trong một CSDL luôn luôn có mối quan hệ mật thiết với nhau. Bộ giá trị của quan hệ này được thêm vào một cách hợp lệ nếu tồn tại một bản ghi tương ứng của một quan hệ khác.

Phụ thuộc tồn tại xảy ra nếu có một trong hai trường hợp sau:

- (i) Có sự hiện diện của khóa ngoại.
- (ii) Có sự lồng khóa giữa các quan hệ.

Ví dụ 1:

Trong thể hiện của quan hệ CHITIẾT_HĐ, sự tồn tại của mỗi bộ giá trị cthđ đều phụ thuộc vào sự tồn tại của một bộ giá trị hđ trong thể hiện của quan hệ HÓAĐƠN sao cho hđ.Số-hóa-đơn = cthđ.Số-hóa-đơn, và phụ thuộc cả vào sự tồn tại của một bộ giá trị mh trong thể hiện của quan hệ DM_HÀNG sao cho mh.Mã-hàng = cthđ.Mã-hàng.

Biểu diễn các RBTV này như sau:

RBTV1 : “Mỗi bộ của CHITIẾT_HĐ phải có một hóa đơn với Số-hóa-đơn tương ứng”:

$$\forall \text{cthđ} \in \text{CHITIẾT_HĐ}, \exists \text{hđ} \in \text{HÓAĐƠN}$$

sao cho cthđ.Số-hóa-đơn = hđ.Số-hóa-đơn.

hoặc biểu diễn bằng cách khác:

$$\text{CHITIẾT_HĐ} [\text{Số-hóa-đơn}] \subseteq \text{HÓAĐƠN} [\text{Số-hóa-đơn}]$$

RBTV2 : “Mỗi bộ của CHITIẾT_HĐ phải có mã hàng thuộc về danh mục hàng”:

$$\forall \text{cthđ} \in \text{CHITIẾT_HĐ}, \exists \text{hh} \in \text{DM_HÀNG} \text{ sao cho } \text{cthđ.Mã-hàng} = \text{hh.Mã-hàng}$$

hoặc biểu diễn bằng cách khác:

$$\text{CHITIẾT_HĐ} [\text{Mã-hàng}] \subseteq \text{DM_HÀNG} [\text{Mã-hàng}]$$

Ví dụ 2 :

Trong CSDL về quản lý CBCC nêu trong các ví dụ trên, RBTV về phụ thuộc tồn tại giữa 2 quan hệ CBCC và NGÁCH-BẬC-LƯƠNG được xác định:

$$\forall \text{cbcc} \in \text{CBCC}, \exists \text{ng} \in \text{NGÁCH-BẬC-LƯƠNG}$$

sao cho (cbcc.Mã-ngạch = ng.Mã-ngạch) ^ (cbcc.Bậc = ng.Bậc)

hoặc biểu diễn bằng cách khác:

CBCC [Mã-ngạch, Bậc] $\subseteq$ NGẠCH-BẬC-LƯƠNG [Mã-ngạch, Bậc]

Ví dụ 3:

Trong CSDL về quản lý học viên đã nêu trong chương trước, các RBTV về phụ thuộc tồn tại gồm:

RBTV1 : “Mỗi LỚP-HỌC đều phải thuộc một KHOA nhất định”:

$\forall lh \in \text{LỚP-HỌC}, \exists kh \in \text{KHOA}$ sao cho $lh.\text{Mã-khoa} = kh.\text{Mã-khoa}$.

hoặc biểu diễn qua phép chiếu quan hệ:

$\text{LỚP-HỌC [Mã-khoa]} \subseteq \text{KHOA [Mã-khoa]}$.

RBTV2 : “Mỗi HỌC-VIÊN đều phải thuộc một LỚP-HỌC nhất định”:

$\forall hv \in \text{HỌC-VIÊN}, \exists lh \in \text{LỚP-HỌC}$ sao cho $hv.\text{Mã-lớp} = lh.\text{Mã-lớp}$.

hoặc biểu diễn qua phép chiếu quan hệ:

$\text{HỌC-VIÊN [Mã-lớp]} \subseteq \text{LỚP-HỌC [Mã-lớp]}$.

RBTV3 : “Mỗi KQUẢ-THI đều phải là của một HỌC-VIÊN nhất định”:

$\forall kq \in \text{KQUẢ-THI}, \exists hv \in \text{HỌC-VIÊN}$

sao cho $kq.\text{Mã-học-viên} = hv.\text{Mã-học-viên}$.

hoặc biểu diễn qua phép chiếu quan hệ:

$\text{KQUẢ-THI [Mã-học-viên]} \subseteq \text{HỌC-VIÊN [Mã-học-viên]}$.

RBTV4 : “Mỗi môn thi trong KQUẢ-THI đều phải có tên trong danh sách các môn học” :

$\forall kq \in \text{KQUẢ-THI}, \exists mh \in \text{MÔN-HỌC}$ sao cho $kq.\text{Mã-môn} = mh.\text{Mã-môn}$

hoặc biểu diễn qua phép chiếu quan hệ:

$\text{KQUẢ-THI [Mã-môn]} \subseteq \text{MÔN-HỌC [Mã-môn]}$

Chúng ta có thể thấy về phải của phép toán tập con ($\subseteq$) là phép chiếu trên thuộc tính khóa nội của một quan hệ, còn về trái là phép chiếu trên tập các thuộc tính khóa ngoại của một quan hệ khác. Chính vì lẽ đó mà người ta còn gọi RBTV loại này là RBTV về khóa ngoại. Phát biểu tổng quát về loại RBTV này là như sau:

R và S là hai quan hệ định nghĩa trên các tập thuộc tính R^+ và S^+ . $K_R \subseteq R^+$ là tập các thuộc tính khóa nội của quan hệ R; $K_S \subseteq S^+$ là tập các thuộc tính khóa nội của quan hệ S; và $W \subseteq S^+$ là tập các thuộc tính khóa ngoại của S đối với R. Khi đó ta có phụ thuộc tồn tại của S vào R và được biểu diễn thông qua phép chiếu:

$S[W] \subseteq R[W]$.

Nếu $W \subseteq K_S$, thì ta nói rằng có sự lồng khóa giữa hai quan hệ R và S.

Trong bảng tầm ảnh hưởng của loại RBTV này, các thao tác Thêm và Sửa một bộ giá trị của quan hệ R (về phải của phụ thuộc tồn tại) không gây ra sự vi phạm RBTV (trừ khi có sự lồng khóa giữa R với một quan hệ khác), chỉ có thao tác Xóa bỏ một bộ giá trị của R mới cần có sự kiểm tra RBTV. Ngược lại, thao tác Xóa một bộ giá trị của S không gây ra sự vi phạm RBTV (trừ khi có sự lồng khóa của một quan hệ khác vào S), thao tác Thêm một bộ giá trị mới vào S luôn luôn phải được kiểm tra RBTV này; nếu W là các thuộc tính khóa ngoại của S thì việc Sửa bộ giá trị của S trên các thuộc tính khóa ngoại W vẫn phải kiểm tra

RBTV; nếu có sự lồng khóa thì việc sửa không đòi hỏi kiểm tra RBTV vì theo quy ước là không được sửa giá trị của thuộc tính khóa.

Bảng tầm ảnh hưởng có 2 dạng ứng với 2 trường hợp trên như sau:

a. Ứng với trường hợp khóa ngoại:

Quan hệ	Thêm	Sửa	Xóa
R	-	- (*)	+
S	+	+ _(w)	-

b. Ứng với trường hợp lồng khóa:

Quan hệ	Thêm	Sửa	Xóa
R	-	- (*)	+
S	+	- (*)	-

2.2.2 Ràng buộc toàn vẹn tổng hợp (liên bộ - liên quan hệ)

Khi có sự hiện diện của 1 thuộc tính mang tính chất tổng hợp (tức là giá trị của thuộc tính có thể được tính toán từ giá trị của các thuộc tính khác trên một hay nhiều bộ giá trị của các quan hệ trong CSDL), hay phụ thuộc tồn tại lồng khóa thì có RBTV liên quan hệ - liên bộ.

Ví dụ 1:

Xét CSDL về quản lý học viên nêu trong ví dụ trên, RBTV liên quan hệ - liên bộ có thể được xác định: “Với mọi bộ giá trị của LỚP-HỌC, nếu Số lượng học viên lớn hơn 0 thì số lượng này phải lớn hơn hay bằng tổng số bộ giá trị đếm được của các học viên có cùng Mã lớp”. Đây chính là RBTV R6 đã nêu trong ví dụ tại mục 1.1.

Biểu diễn hình thức của RBTV này như sau:

$\forall lh \in \text{LỚP-HỌC}$ thì:

nếu $lh.\text{Số-học-viên} > 0$ thì:

$h.\text{Số-học-viên} = \text{COUNT}(hv \in \text{HỌC-VIÊN}, hv.\text{Mã-lớp} = lh.\text{Mã-lớp})$.

Ví dụ 2 :

Xét CSDL quản lý hóa đơn bán hàng đã cho trong ví dụ 1.2 với 3 quan hệ:

1) HÓAĐƠN (Số-hóa-đơn, Số_chủng_loại_mặt_hàng, Tổng-trị-giá).

2) CHITIẾT_HĐ (Số-hóa-đơn, Mã-hàng, Số-lượng-đặt, Đơn-giá, Trị-giá).

3) DM_HÀNG (Mã-hàng, Tên-hàng, Đơn-vị-tính).

RBTV1 : “Số_chủng_loại_mặt_hàng = số bộ của CHITIẾT_HĐ có cùng Số-hóa-đơn”

:

$\forall hđ \in \text{HÓAĐƠN}$ thì:

$hđ.\text{Số-chủng-loại-mặt-hàng} = \text{COUNT}(cthđ \in \text{CHITIẾT_HĐ}, cthđ.\text{Số-hóa-đơn} = hđ.\text{Số-hóa-đơn})$

RBTV2 : “Tổng tất cả các Trị-giá của các mặt hàng trong CHITIẾT_HĐ có cùng Số-hóa-đơn phải bằng Tổng-trị-giá của hóa đơn đó trong HÓAĐƠN”:

$\forall \text{hđ} \in \text{HÓAĐƠN}$ thì $\text{hđ.Tổng-trị-giá} = \text{SUM}(\text{cthd.Tri-gia})$

đối với các $\text{cthd} \in \text{CHITIẾT_HĐ}$

sao cho : $\text{cthd.Số-hóa-đơn} = \text{hđ.Số-hóa-đơn}$.

Chúng ta có thể nhận thấy trong CSDL này có sự dư thừa thông tin một cách cố ý. Đó là thuộc tính tính toán Trị-giá của các mặt hàng trong chi tiết hoá đơn bán hàng. Một trong những phương pháp kiểm định tính đúng đắn của dữ liệu được nhập vào là tổ chức nhập “thừa” dữ liệu tính toán được (Computable Value) rồi so sánh với công thức tính toán. Nếu có sự sai sót nào trong các thành phần có liên quan trong công thức thì logic biểu thức sẽ không còn phù hợp nữa.

Bây giờ để đạt được dạng chuẩn (Normal Form) tốt hơn cho quan hệ CHITIẾT-HĐ, chúng ta có thể loại bỏ thuộc tính Trị-giá. Khi đó RBTV2 được viết lại là:

$\forall \text{hđ} \in \text{HÓAĐƠN}$ thì

$\text{hđ.Tổng-trị-giá} = \text{SUM}(\text{cthd.Số-lượng-đặt} * \text{cthd.Đơn-giá})$

đối với các $\text{cthd} \in \text{CHITIẾT_HĐ}$

sao cho : $\text{cthd.Số-hóa-đơn} = \text{hđ.Số-hóa-đơn}$.

❖ TÓM TẮT CHƯƠNG 4

Trong chương này, một số nội dung chính được giới thiệu:

- Các thuật ngữ ràng buộc, ràng buộc toàn vẹn, các khái niệm về ràng buộc toàn vẹn.
- Phân loại ràng buộc toàn vẹn.

❖ CÂU HỎI VÀ BÀI TẬP CHƯƠNG 4

Cho lược đồ CSDL về hệ thống kế toán của một doanh nghiệp với các quan hệ sau:

1. DM-TK (Mã-TK, Tên-TK)

Tên từ: Danh mục các tài khoản hạch toán kế toán theo chế độ kế toán hiện hành của Nước CHXHCN Việt nam bao gồm các tài khoản. Mỗi tài khoản có một tên gọi cụ thể và một mã số duy nhất để phân biệt với mọi tài khoản khác.

2. TK-ĐỐI-ỨNG (Mã-TK, TK-Đối-ứng);

Tên từ: Mỗi tài khoản, theo chế độ hạch toán hình chữ T, khi được phát sinh bên NỢ (hạch toán tăng) thì phải có một mã tài khoản đối ứng bên CÓ (hạch toán giảm) để đảm bảo cân đối tài khoản. Một tài khoản được ghi NỢ có thể có nhiều tài khoản khác nhau được ghi CÓ. Mã tài khoản NỢ và mã tài khoản đối ứng đều phải thuộc danh mục các tài khoản

3. SỔ-CT (Loại-CT, Số-CT, NGÀY-CT, Diễn-Giải, Số-Tiền, TK-NỢ, TK-CÓ).

Tên từ: Trong phương pháp kế toán ghi sổ, các chứng từ ban đầu được ghi vào sổ theo dõi, gọi là sổ chứng từ. Mỗi chứng từ đều thuộc một loại chứng từ cụ thể (LOẠI-CT); có một số chứng từ (SỔ-CT) phân biệt với mọi chứng từ khác. Chứng từ được ghi rõ ngày tháng phát sinh (NGÀY-CT), diễn giải nội dung phát sinh (DIỄN-GIẢI), số tiền phát sinh (SỐ-TIỀN), mã tài khoản ghi NỢ (TK-NỢ) và mã tài khoản đối ứng ghi CÓ (TK-CÓ);

4. SỔ-CÁI (Mã-TK, NỢ-ĐK, CÓ-ĐK, PS-NỢ, PS-CÓ, NỢ-CK, CÓ-CK).

Tân từ: Từ sổ chứng từ (SỔ-CT), các chứng từ ghi sổ được tổng hợp theo từng loại tài khoản (Mã-TK) và lập thành sổ cái. Mỗi mã tài khoản (Mã-TK) trong SỔ-CÁI được phản ánh duy nhất 1 lần các số dư NỢ, dư CÓ đầu kỳ (NỢ-ĐK, CÓ-ĐK); số phát sinh NỢ, CÓ trong tháng (PS-NỢ, PS-CÓ), và số dư NỢ, dư CÓ cuối kỳ (NỢ-CK, CÓ-CK). Mã tài khoản phải có trong danh mục tài khoản (DM-TK) nêu trên.

Câu 1: Xác định khóa của các quan hệ trong CSDL nêu trên.

Câu 2: Xác định các RBTV của CSDL. Nêu rõ nội dung RBTV, bối cảnh và lập (các) bảng tầm ảnh hưởng của các RBTV của lược đồ CSDL

CHƯƠNG 5: LÝ THUYẾT THIẾT KẾ CƠ SỞ DỮ LIỆU

❖ GIỚI THIỆU CHƯƠNG 5

Chương này sẽ trình bày xây dựng một tập các lược đồ quan hệ cho phép ta lưu trữ thông tin sao cho tránh được những dư thừa không cần thiết và khai thác được dễ dàng. Một cách tiếp cận với thiết kế CSDL quan hệ là dạng chuẩn (Normal Form). Để xác định lược đồ quan hệ ở dạng chuẩn nào ta phải mô tả các quy tắc của thông tin thế giới thực dưới mô hình CSDL quan hệ.

MỤC TIÊU CHƯƠNG 5

Sau khi học xong bài này, người học có khả năng:

□ **Về kiến thức:**

- Mô tả, trình bày được khái niệm cơ sở của lý thuyết cơ sở dữ liệu như khóa, phụ thuộc hàm, bao đóng, các dạng chuẩn,..
- Trình bày được dữ liệu ở mức tốt nhất (có thể ứng dụng được) bằng các phép tách, giải thuật chuẩn hóa lược đồ.

□ **Về kỹ năng:**

- Thiết kế được dữ liệu ở mức tốt nhất (có thể ứng dụng được) bằng các phép tách, giải thuật chuẩn hóa lược đồ.
- Thực hiện các thao tác an toàn với máy tính

□ **Về năng lực tự chủ và trách nhiệm:**

- Rèn luyện tính chính xác, sáng tạo, linh hoạt.
- Tuân thủ nội quy, quy định nơi làm việc.

❖ PHƯƠNG PHÁP GIẢNG DẠY VÀ HỌC TẬP CHƯƠNG 5

- Đối với người dạy: sử dụng phương pháp giảng dạy tích cực (diễn giảng, vấn đáp, dạy học theo vấn đề); yêu cầu người học thực hiện câu hỏi thảo luận và bài tập chương 5 (cá nhân hoặc nhóm).

- Đối với người học: chủ động đọc trước giáo trình (Chương 5) trước buổi học; hoàn thành đầy đủ câu hỏi thảo luận và bài tập tình huống chương 5 theo cá nhân hoặc nhóm và nộp lại cho người dạy đúng thời gian quy định.

❖ **ĐIỀU KIỆN THỰC HIỆN CHƯƠNG 5**

- **Phòng học chuyên môn hóa/nhà xưởng:** Phòng máy tính
- **Trang thiết bị máy móc:** Máy chiếu và các thiết bị dạy học khác
- **Học liệu, dụng cụ, nguyên vật liệu:** Bài giảng soạn trên slide môn học, giáo trình, tài liệu tham khảo, giáo án và các tài liệu liên quan.
- **Các điều kiện khác:** Không có

❖ **KIỂM TRA VÀ ĐÁNH GIÁ CHƯƠNG 5**

- **Nội dung:**
 - ✓ **Kiến thức:** Kiểm tra và đánh giá tất cả nội dung đã nêu trong mục tiêu kiến thức
 - ✓ **Kỹ năng:** Đánh giá tất cả nội dung đã nêu trong mục tiêu kỹ năng.
 - ✓ **Năng lực tự chủ và trách nhiệm:** Trong quá trình học tập, người học cần:
 - + Nghiên cứu bài trước khi đến lớp
 - + Chuẩn bị đầy đủ tài liệu học tập.
 - + Tham gia đầy đủ thời lượng môn học.
 - + Nghiêm túc trong quá trình học tập.
- **Phương pháp:**
 - ✓ **Điểm kiểm tra thường xuyên:** không có
 - ✓ **Kiểm tra định kỳ lý thuyết:** 1 điểm kiểm tra (hình thức: viết)

NỘI DUNG CHƯƠNG 5

LÝ THUYẾT THIẾT KẾ CƠ SỞ DỮ LIỆU

1. Các vấn đề gặp phải khi tổ chức dữ liệu

Một trong hai nguyên nhân sau đây do thiết kế kém sẽ gây nguy hiểm cho CSDL

Trùng lặp thông tin. Không có khả năng trình bày thông tin một cách chắc chắn

VD: Cho một lược đồ quan hệ dùng để ghi nhận giáo viên và lớp giảng dạy của giáo viên

GIANGDAY(MÔNHOc, SOTIET,LOP,GV,HV,DC)

Các phụ thuộc hàm: MONHOC → SOTIET. MONHOC, LOP → GV .
GV→HOCVI,DC

Xét một tình trạng dữ liệu như sau:

MONHOC	SOTIE T	LOP	GV	HOCVI	DC
CSDL	75	TH1	N.V.A	C.N	Q1
CSDL	75	TH2	N.V.A	C.N	Q1
CSDL	45	TH1	N.V.B	T.S	Q3
CSDL	45	TH2	N.V.B	T.S	Q3

CN: Cử nhân, TS: Tiến sĩ

Do có phụ thuộc hàm $MONHOC \rightarrow SOTIET$ nên số tiết của dòng thứ 2 và dòng thứ 4 gây nên trùng lặp thông tin. Tương tự phụ thuộc hàm $GV \rightarrow HOCVI, DC$ nên học vị và địa chỉ của dòng thứ 2 và dòng thứ 4 gây nên trùng lặp thông tin. Các dữ liệu gây trùng lặp thông tin là các dữ liệu có thể suy đoán được một cách chắc chắn và duy nhất từ phụ thuộc hàm.

Ở đây để lưu học vị và địa chỉ của một giáo viên thì giáo viên đó phải tham gia giảng dạy một lớp nào đó. Để giải quyết vấn đề lưu thông tin các giáo viên không tham gia giảng dạy người ta dùng giá trị NULL cho các thuộc tính MONHOC, SOTIET, LOP. Như vậy, lược đồ quan hệ này lưu trữ hai thông tin của hai đối tượng khác nhau một là giảng dạy của các giáo viên tham gia giảng dạy, hai là thông tin của các giáo viên không tham gia giảng dạy. Vấn đề nảy sinh ở đây là khi ta chỉ cần cập nhật việc giảng dạy ta phải đảm bảo không gây ảnh hưởng tới các giáo viên không tham gia giảng dạy và ngược lại. Như vậy thông tin lưu trữ ở lược đồ quan hệ này không chắc chắn.

2. Các phụ thuộc hàm

Phụ thuộc hàm (functional dependancy) là một công cụ dùng để biểu diễn một cách hình thức các ràng buộc toàn vẹn. Phương pháp biểu diễn này có rất nhiều ưu điểm, và đây là một công cụ kỳ quan trọng, gắn chặt với lý thuyết thiết kế cơ sở dữ liệu.

Trong chương này chúng ta sẽ tìm hiểu về lý thuyết thiết kế cơ sở dữ liệu quan hệ, mà bắt đầu là phụ thuộc hàm và một số ứng dụng trong việc giải quyết các bài toán như: tìm khoá, tìm phủ tối thiểu, xác định dạng chuẩn. Sau đó, chúng ta sẽ tiếp tục tìm hiểu về cách thức chuẩn hoá một cơ sở dữ liệu.

2.1. Định nghĩa phụ thuộc hàm

Phụ thuộc hàm có tầm quan trọng rất lớn trong việc phân tích và thiết kế mô hình dữ liệu. Mục này sẽ trình bày kỹ hơn về vấn đề này.

Khái niệm: Quan hệ R được định nghĩa trên tập thuộc tính $U = \{ A_1, A_2, \dots, A_n \}$. $A, B \subset U$ là 2 tập con của tập thuộc tính U. Nếu tồn tại một ánh xạ $f: A \rightarrow B$ thì ta nói rằng A xác định hàm B, hay B phụ thuộc hàm vào A, và ký hiệu là $A \rightarrow B$.

2.2. Cách xác định phụ thuộc hàm cho lược đồ quan hệ

Định nghĩa hình thức của phụ thuộc hàm như sau:

Quan hệ Q (A, B, C) có phụ thuộc hàm A xác định B (ký hiệu là $A \rightarrow B$) nếu:

$\forall q, q' \in Q$, sao cho $q.A = q'.A$ thì $q.B = q'.B$

(Nghĩa là: ứng với 1 giá trị của A thì có một giá trị duy nhất của B)

A là vế trái của phụ thuộc hàm, B là vế phải của phụ thuộc hàm.

$A \rightarrow B$ được gọi là *phụ thuộc hàm hiển nhiên* nếu $B \subseteq A$. Nghĩa là, một tập A lớn hơn và bao tập con B thì A xác định được giá trị của các thuộc tính trong tập B là điều hiển nhiên.

$A \rightarrow B$ được gọi là *phụ thuộc hàm nguyên to*, hoặc nói cách khác, B được gọi là *phụ thuộc hàm đầy đủ* (fully functional dependence) vào A nếu $\forall A' \not\subset A$ đều không có $A' \rightarrow B$.

Ví dụ 1 :

Trong lược đồ CSDL quản lý hóa đơn bán hàng đã cho trong ví dụ trên, quan hệ HÓAĐƠN (Số-hóa-đơn, Số_chủng_loại_mặt_hàng, Tổng-trị-giá) có các phụ thuộc hàm sau:

f1: Số-hóa-đơn $\rightarrow$ Số_chủng_loại_mặt_hàng;

f2: Số-hóa-đơn $\rightarrow$ Tổng-trị-giá;

bởi vì Số-hóa-đơn là khóa của lược đồ quan hệ HÓAĐƠN. Nếu biết số hóa đơn thì ta có thể xác định được tất cả các thông tin còn lại liên quan đến hóa đơn đó, trong đó có thông tin về Số_chủng_loại_mặt_hàng và Tổng-trị-giá tất cả các mặt hàng của hóa đơn. Các phụ thuộc hàm trên đều là nguyên tố.

Quan hệ CHITIẾT_HĐ (Số-hóa-đơn, Mã-hàng, Số-lượng-đặt, Đơn-giá, Trị-giá) có các phụ thuộc hàm sau:

f1: Số-hóa-đơn, Mã-hàng $\rightarrow$ Số-lượng-đặt.

f2: Số-hóa-đơn, Mã-hàng $\rightarrow$ Đơn-giá.

f3: Số-hóa-đơn, Mã-hàng $\rightarrow$ Trị-giá.

f4: Số-lượng-đặt, Đơn-giá $\rightarrow$ Trị-giá.

Thuộc tính Đơn-giá phụ thuộc hàm không đầy đủ vào khóa (Số-hóa-đơn, Mã-hàng), bởi vì nó chỉ phụ thuộc vào mặt hàng (thông qua Mã-hàng).

Qua ví dụ vừa nêu chúng ta thấy, trên một lược đồ quan hệ có thể tồn tại nhiều phụ thuộc hàm. Tập các phụ thuộc hàm thường được ký hiệu bằng chữ F.

2.3. Một số tính chất của phụ thuộc hàm-Hệ luật dẫn Armstrong

Năm 1974, Armstrong đã đưa ra hệ tiên đề (còn gọi là hệ luật dẫn Armstrong, hay các tính chất của phụ thuộc hàm) (D.Maier - 1983 [2]) như sau:

$X, Y, Z, W \subseteq U$. Phụ thuộc hàm có các tính chất sau đây:

(i) Tính phản xạ:

Nếu $Y \subseteq X$ thì $X \rightarrow Y$.

(ii) Tính gia tăng:

Nếu $X \rightarrow Y$ thì $XZ \rightarrow YZ$.

(iii) Tính bắc cầu:

Nếu $X \rightarrow Y$ và $Y \rightarrow Z$ thì $X \rightarrow Z$.

Và người ta đã chứng minh rằng hệ tiên đề Armstrong là đúng đắn và đầy đủ thông qua 3 bổ đề (ở đây không chứng minh):

Bổ đề 1: Hệ tiên đề Armstrong là đúng, nghĩa là, với F là tập phụ thuộc hàm đúng trên quan hệ R , nếu $X \rightarrow Y$ là một phụ thuộc hàm

Bổ đề 2: Từ hệ tiên đề Armstrong suy ra một số luật bổ sung sau đây:

(iv) Tính phân rã (hoặc luật tách):

Nếu $X \rightarrow YZ$ thì $X \rightarrow Y$ và $X \rightarrow Z$.

(v) Tính hợp (hoặc luật hợp):

Nếu $X \rightarrow Y$ và $X \rightarrow Z$ thì $X \rightarrow YZ$.

(vi) Tính tựa bắc cầu, hoặc bắc cầu giả:

Nếu $X \rightarrow Y$ và $YZ \rightarrow W$ thì $XZ \rightarrow W$.

Ví dụ 1:

Cho lược đồ quan hệ $R(A, B, C, D, E, G, H)$ và tập các phụ thuộc hàm $F = \{AB \rightarrow C, B \rightarrow D, CD \rightarrow E, CE \rightarrow GH, G \rightarrow A\}$. Áp dụng hệ tiên đề Armstrong, tìm một chuỗi suy diễn $AB \rightarrow E$.

Giải:

1. $AB \rightarrow C$ (cho trước - phụ thuộc hàm f_1)
2. $AB \rightarrow AB$ (tính chất phản xạ)
3. $AB \rightarrow B$ (luật tách)
4. $B \rightarrow D$ (cho trước - phụ thuộc hàm f_2)
5. $AB \rightarrow D$ (bắc cầu 3 & 4)
6. $AB \rightarrow CD$ (hợp 1 & 5)
7. $CD \rightarrow E$ (cho trước - phụ thuộc hàm f_3)
8. $AB \rightarrow E$ (bắc cầu 6 & 7). Kết thúc.

Ví dụ 2:

Cho lược đồ quan hệ $R(A, B, C, D, E, G, H, I, J)$ và tập các phụ thuộc hàm $F = \{AB \rightarrow E, AG \rightarrow J, BE \rightarrow I, E \rightarrow G, GI \rightarrow H\}$. Tìm chuỗi suy diễn $AB \rightarrow GH$

Giải:

1. $AB \rightarrow E$ (cho trước - phụ thuộc hàm f_1)
2. $AB \rightarrow AB$ (phản xạ)
3. $AB \rightarrow B$ (luật tách)
4. $AB \rightarrow BE$ (hợp của 1 & 3)
5. $BE \rightarrow I$ (cho trước - phụ thuộc hàm f_3)
6. $AB \rightarrow I$ (bắc cầu 4 & 5)
7. $E \rightarrow G$ (cho trước - phụ thuộc hàm f_4)
8. $AB \rightarrow G$ (bắc cầu 1 & 7)
9. $AB \rightarrow GI$ (hợp 6 & 8)
10. $GI \rightarrow H$ (cho trước - phụ thuộc hàm f_5)
11. $AB \rightarrow H$ (bắc cầu 9 & 10)
12. $AB \rightarrow GH$ (hợp 8 & 11)

Bổ đề 3: $X \rightarrow Y$ được suy diễn logic từ F nhờ hệ tiên đề Armstrong khi và chỉ khi $Y \subseteq X_f^+$.

X_f^+ .

(X_f^+ là bao đóng của tập thuộc tính X đối với tập phụ thuộc hàm F chúng ta sẽ nghiên cứu trong mục 3.3 sau đây.)

3. Bao đóng của tập phụ thuộc hàm và bao đóng của tập thuộc tính

3.1. Bao đóng của tập phụ thuộc hàm F

Bao đóng (closure) của tập phụ thuộc hàm F (ký hiệu là F^+) là tập hợp tất cả các phụ thuộc hàm có thể suy ra từ F dựa vào các tiên đề Armstrong. Rõ ràng $F \subseteq F^+$

Ví dụ

Cho lược đồ quan hệ $Q(ABCDEFGH)$ và F được cho như sau:

$$F = \{B \twoheadrightarrow A; DA \twoheadrightarrow CE; D \twoheadrightarrow H; GH \twoheadrightarrow C; AC \twoheadrightarrow D\}$$

$$\text{Khi đó } F^+ = \{B \twoheadrightarrow A; DA \twoheadrightarrow CE; D \twoheadrightarrow H; GH \twoheadrightarrow C; AC \twoheadrightarrow D; \\ BC \twoheadrightarrow AC; BC \twoheadrightarrow D; DA \twoheadrightarrow AH; DG \twoheadrightarrow C; BC \twoheadrightarrow AD; \dots\}$$

(Lưu ý rằng, nếu mỗi thuộc tính được biểu diễn bằng một ký tự thì danh sách các thuộc tính có hoặc không có dấu phẩy đều được, còn giữa các phụ thuộc hàm phải có dấu chấm phẩy)

Các tính chất của tập F^+

1. Tính phản xạ:

Với mọi tập phụ thuộc hàm F^+ ta luôn có $F \subseteq F^+$

2. Tính đơn điệu:

Nếu $F \subseteq G$ thì $F^+ \subseteq G^+$

3. Tính lũy đẳng:

Với mọi tập phụ thuộc hàm F ta luôn luôn có $F^{++} = F^+$.

3.2. Bao đóng của tập phụ thuộc tính X

Định nghĩa: Bao đóng (Closure) của tập các thuộc tính X đối với tập các phụ thuộc hàm F (ký hiệu là X_F^+ hoặc X^+) là tập tất cả các thuộc tính A có thể suy dẫn từ X nhờ tập bao đóng của các phụ thuộc hàm F^+ :

$$X_F^+ = \{A \mid X \rightarrow A \in F^+\}$$

Thuật toán tìm bao đóng của X dựa trên tập phụ thuộc hàm F đối với quan hệ R được mô tả bằng ngôn ngữ tựa Pascal như sau

Procedure Closure (X, F)

Begin

ketqua:= X

 While (có sự thay đổi trên tập **ketqua**) Do

 For (mỗi pth $W \rightarrow Z$ trong F) Do

 If $W \subseteq \text{ketqua}$ Then

ketqua := **ketqua** $\cup$ Z

 Return **ketqua**;

End;

Tập **ketqua** là bao đóng của tập phụ thuộc hàm X

Ví dụ: cho tập phụ thuộc hàm $F = \{A \rightarrow BC, I \rightarrow K, GB \rightarrow H, CG \rightarrow I, B \rightarrow H\}$ của quan hệ $R(ABCDEFGHIK)$. Hãy tính bao đóng của tập thuộc tính AG , $(AG)^+$

Áp dụng thuật toán trên ta tính như sau:

Ban đầu **ketqua**= AG

Ta lần lượt xét tất cả các phụ thuộc hàm trong F:

$A \rightarrow BC$ có $A \subseteq \text{ketqua}$ nên $\text{ketqua} = \text{ketqua} \cup BC = AGBC$

$I \rightarrow K$ có $I \not\subseteq \text{ketqua}$ nên ketqua vẫn giữ nguyên

$GB \rightarrow H$ có $GB \subseteq \text{ketqua}$ nên $\text{ketqua} = \text{ketqua} \cup H = AGBCH$

$CG \rightarrow I$ có $CG \subseteq \text{ketqua}$ nên $\text{ketqua} = \text{ketqua} \cup I = AGBCHI$

$B \rightarrow H$ có $B \subseteq \text{ketqua}$ nhưng đã có H trong ketqua nên ketqua giữ nguyên

Quay lại từ đầu tập F lần 2:

$A \rightarrow BC$ có $A \subseteq \text{ketqua}$ nhưng đã có BC trong ketqua nên ketqua giữ nguyên

$I \rightarrow K$ có $I \subseteq \text{ketqua}$ nên $\text{ketqua} = \text{ketqua} \cup K = AGBCHIK$

Tiếp tục các phụ thuộc hàm sau không làm thay đổi kết quả.

Lần này tập ketqua có thay đổi nên lại quay lại từ đầu tập F lần 3:

Lần này ketqua không thay đổi nên dừng.

Cuối cùng ta được $(AG)^+ = \text{ketqua} = AGBCHIK$.

Để xác định một phụ thuộc hàm có thuộc F^+

Để xác định phụ thuộc hàm $X \rightarrow Y$ có thuộc F^+ ta tính X^+

Nếu $Y \subseteq X^+$ thì $X \rightarrow Y$ thuộc F^+ , trái lại thì không thuộc.

Ví dụ: $F = \{CD \rightarrow A, E \rightarrow B, DB \rightarrow C, C \rightarrow D\}$

phụ thuộc hàm nào sau đây thuộc F^+ : $DE \rightarrow BC, AC \rightarrow BE$

Vì $(DE)^+ = DEBCA$ chứa BC nên BC nên $DE \rightarrow DC$ thuộc F^+

Vì $(AC)^+ = ACD$ không chứa BE nên $AC \rightarrow BE$ không thuộc F^+

3.3. Bài toán thành viên

Qua phần trên ta nhận thấy X^+ được định nghĩa thông qua F^+ . Vấn đề nảy sinh khi nghiên cứu lý thuyết CSDL là: Cho trước tập các phụ thuộc hàm F và một phụ thuộc hàm f, bài toán kiểm tra có hay không $f \in F^+$ gọi là bài toán thành viên.

Để giải quyết bài toán bài toán thành viên thật sự không đơn giản; vì mặc dù F là rất nhỏ nhưng F^+ thì có thể rất lớn. Tuy nhiên ta có thể giải bằng cách tính X^+ và so sánh X^+ với tập Y. Dựa vào tính chất $X \twoheadrightarrow Y \in F^+ \iff Y \subseteq X^+$, ta có ngay câu trả lời $X \twoheadrightarrow Y \in F^+$ hay không? Như vậy thay vì giải bài toán thành viên ta đưa về giải bài toán tìm bao đóng của tập thuộc tính.

3.4. Thuật toán tìm bao đóng của một tập thuộc tính

Thuật toán 3.4.1

Thuật toán tìm bao đóng với độ phức tạp $O(N^2)$, với N là số lượng thuộc tính của lược đồ quan hệ Q.

Dữ Liệu Vào Q, F, X $\in$ Q+

Dữ Liệu Ra X^+

Bước 1: Đặt $X^+ = X$

Bước 2: Temp = X^+

$\text{do } f \in U \twoheadrightarrow V \in F$

if ($U \subseteq X^+$)

$X^+ = X^+ \cup V$. $F = F - f$;

Bước 3: if ($X^+ = \text{Temp}$)

“ X^+ chính là kết quả cần tìm “ Dừng thuật toán

Else

Trở lại Bước 2.

Ví dụ 3.4

Cho lược đồ quan hệ $Q(ABCDEFGH)$ và tập phụ thuộc hàm F

$F = \{ f_1: B \twoheadrightarrow A;$

$f_2: DA \twoheadrightarrow CE;$

$f_3: D \twoheadrightarrow H; f_4: GH \twoheadrightarrow C; f_5: AC \twoheadrightarrow D \}$

Tìm bao đóng của các tập $X = \{AC\}$ dựa trên F .

Giải:

$X^+ = AC$

Do f_1, f_2, f_3, f_4 không thỏa. f_5 thỏa : $X^+ = ACD$

Lập lại bước 2. f_1 không thỏa, f_2 thỏa: $X^+ = ACDE$, f_3 thỏa : $X^+ = ACDEH$

Đến đây rõ ràng không có phụ thuộc hàm nào làm thay đổi X^+ nữa, thuật toán dừng lại và kết quả $X^+ = ACDEH$

Chú ý rằng bạn đọc hãy nắm thật kỹ thuật toán này – nó mở đầu cho một loạt ứng dụng quan trọng về sau. Tiếp theo, chúng tôi nêu lên một thuật toán tìm bao đóng với độ phức tạp tuyến tính để các bạn tham khảo.

Thuật toán 3.4.2

Thuật toán tìm bao đóng với độ phức tạp tuyến tính[3]

Bước 1:

Xây dựng mảng một chiều COUNT

Với $COUNT(i)$ là số thuộc tính về trái của phụ thuộc hàm thứ i

Bước 2:

Xây dựng mảng LIST với $LIST(A) = \{X \twoheadrightarrow Y \mid F, A \twoheadrightarrow X\}$ (lưu chỉ số phụ thuộc hàm)

Bước 3:

$X^+ = X$

Bước 4:

Mọi thuộc tính $A \twoheadrightarrow X^+$

Giảm $COUNT(X \twoheadrightarrow Y)$ đi một nếu $A \twoheadrightarrow X$

Nếu $\text{COUNT}\{X \bowtie Y\} = 0$ thì $X^+ = X^+ \bowtie Y$

Quay lại duyệt thuộc tính kế tiếp trong X^+ cho đến khi nào duyệt hết mọi phần tử của X^+ thì dừng lại. Kết quả X^+ là bao đóng cần tìm.

4. Khóa của lược đồ quan hệ - Một số thuật toán tìm khóa

4.1. Định nghĩa khóa của quan hệ

Cho quan hệ $Q(A_1, A_2, \dots, A_n)$ được xác định bởi tập thuộc tính Q^+ và tập phụ thuộc hàm F định nghĩa trên Q , cho $K \subseteq Q^+$.

K là một khóa của Q nếu thỏa đồng thời cả hai điều kiện sau:

1. $K \bowtie Q^+ \rightarrow F^+$ (hay $K \twoheadrightarrow F^+$)

F

(K chỉ thỏa điều kiện 1 thì được gọi là siêu khoá)

2. Không tồn tại $K' \subsetneq K$ sao cho $K'^+ \neq Q^+$

Một lược đồ quan hệ có thể có nhiều siêu khoá, nhiều khoá.

4.2. Thuật toán tìm một khóa của một lược đồ quan hệ Q

Thuật toán 3.4.3

$K = Q^+$;

While A thuộc K do

if $(K - A)^+ = Q^+$ then $K = K - A$

K còn lại chính là một khóa cần tìm.

Nếu muốn tìm các khóa khác (nếu có) của lược đồ quan hệ, ta có thể thay đổi thứ tự loại bỏ các phần tử của K.

Ví dụ

Cho lược đồ quan hệ Q(ABC) và tập phụ thuộc hàm

$F = \{ A \twoheadrightarrow B; A \twoheadrightarrow C;$

$B \twoheadrightarrow A \}$

Hãy tìm một khóa của Q.

Giải:

$K = \{A, B, C\}$

Loại thuộc tính A, do $(K - A)^+ = Q^+$ nên $K = \{B, C\}$

thuộc tính B không loại được do $(K - B)^+ \neq Q^+$ nên $K = \{B, C\}$

Loại thuộc tính C, do $(K - C)^+ = Q^+$ nên $K = \{B\}$. Vậy một khóa của Q là B.

4.3. Thuật toán tìm tất cả khóa của một lược đồ quan hệ

Định nghĩa về khóa trong chương trước được định nghĩa lại bằng phụ thuộc hàm như sau:

R là lược đồ quan hệ định nghĩa trên tập các thuộc tính $U = \{ A_1, A_2, \dots, A_n \}$, với tập các phụ thuộc hàm $F = \{ f_1, f_2, \dots, f_m \}$ xác định trên R. $K \subseteq U$ là khóa của R nếu thỏa mãn hai điều kiện sau đây:

(i) $K \rightarrow U$.

(ii) $\nexists K' \subsetneq K$ mà $K' \rightarrow U$.

Bây giờ chúng ta hãy biểu diễn lược đồ quan hệ R (U) bằng đồ thị có hướng như sau:

- Mỗi nút của đồ thị là tên một thuộc tính của R.

- Cung nối 2 thuộc tính A và B thể hiện phụ thuộc hàm $A \rightarrow B$

Thuộc tính mà chỉ có các mũi tên đi ra (tức là chỉ nằm trong vế trái của các phụ thuộc hàm) được gọi là nút gốc.

Thuộc tính mà tới nó chỉ có các cung đi tới (tức là chỉ nằm trong vế phải của các phụ thuộc hàm) được gọi là nút lá.

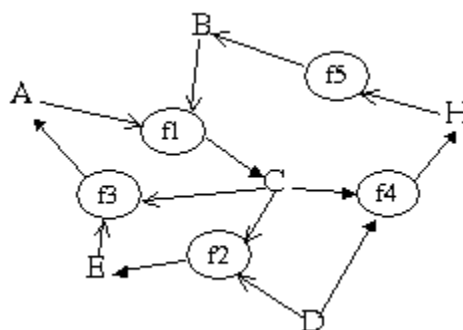
Như vậy khóa của lược đồ quan hệ phải bao phủ tập các nút gốc, đồng thời không chứa bất kỳ nút lá nào của đồ thị.

Thuật toán

Xuất phát từ tập các nút gốc (X), dựa trên tập các phụ thuộc hàm F, chúng ta tìm bao đóng X^+ . Nếu $X^+ = U$ thì X là khóa, ngược lại thì bổ sung một thuộc tính không thuộc nút lá vào X rồi tìm bao đóng. Cứ như thế cho đến khi tìm được bao đóng của X bằng U.

Ví dụ 1:

Cho R (A, B, C, D, E, H) với $F = \{ AB \rightarrow C, CD \rightarrow E, EC \rightarrow A, CD \rightarrow H, H \rightarrow B \}$.
Hãy tìm khóa của R.

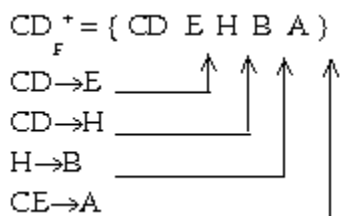


5.

Trong đồ thị trên chúng ta thấy chỉ có nút D là nút gốc, các nút còn lại đều không phải nút lá. Khóa của R phải chứa thuộc tính D.

$D^+ = \emptyset$, bởi vì không tìm thấy phụ thuộc hàm nào có vế trái chỉ có một mình D.

Vì CD có mặt trong vế trái của 2 phụ thuộc hàm do đó ghép thêm C vào tập các nút gốc để xét khóa.



Như vậy $CD \rightarrow \{C, D, E, H, B, A\}$ do đó CD là khóa của R.

Ví dụ 2: Cho quan hệ GIẢNG-DẠY (MS_CBGD, MS_MH, T_CBGD, HH_CBGD, ML, TSSV) với tập phụ thuộc hàm:

$F = \{ \text{MS_CBGD} \rightarrow \text{T_CBGD};$
 $\text{MS_MH} \rightarrow \text{HH_CBGD, ML};$
 $\text{HH_CBGD} \rightarrow \text{ML};$

$MS_CBGD \rightarrow HH_CBGD;$

$MS_CBGD, MS_MH \rightarrow TSSV \}$

Ở đây MS_CBGD là mã số cán bộ giảng dạy; MS_MH là mã số môn học; T_CBGD là tên cán bộ giảng dạy; HH_CBGD là học hàm của cán bộ giảng dạy; ML là mã số lớp học; và $TSSV$ là tổng số sinh viên theo học môn MS_MH do giảng viên MS_CBGD phụ trách.

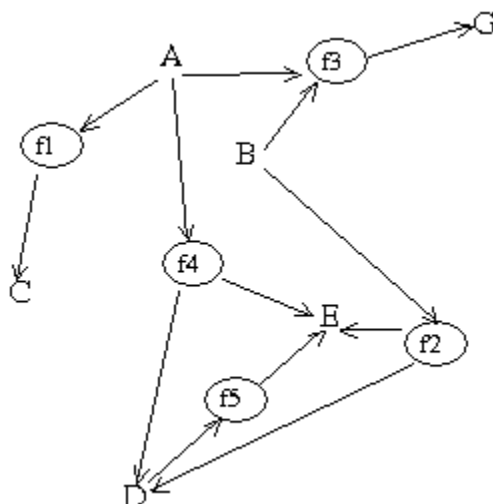
Bài toán: Xác định khóa của quan hệ GIẢNG-DẠY.

Lời giải: Để cho đơn giản, chúng ta hãy ký hiệu tên các thuộc tính của quan hệ trên lần lượt là A, B, C, D, E, G tương ứng. Khi đó quan hệ GIẢNG-DẠY và tập phụ thuộc hàm F được viết ngắn gọn lại là:

$R(A, B, C, D, E, G)$

$F = \{ A \rightarrow C; B \rightarrow D, E; D \rightarrow E; A \rightarrow ED; AB \rightarrow G \}$

Đồ thị biểu diễn các phụ thuộc hàm như sau:



Chúng ta nhận thấy trên đồ thị, hai thuộc tính A và B là các nút gốc. E, C và G là các nút lá. Khóa của quan hệ phải chứa các thuộc tính ở các nút gốc.

Xét $X = \{ A \}$

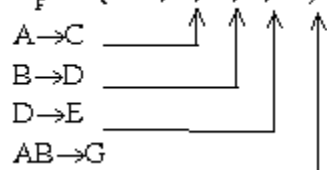
$X^+ = \{ A, C, D, E \}$ // Còn thiếu thuộc tính B và G

Xét $X = \{ B \}$

$X^+ = \{ D, E \}$ // Còn thiếu thuộc tính A, B, C và G

Lấy $X = \{ A, B \}$

$X_F^+ = \{ AB, C, D, E, G \} = U$



Vậy AB là khóa của R. Tức là, khóa của quan hệ giảng-dạy là {MS_CBGD, MS_MH}

5. Phủ tối thiểu

5.1. Tập phụ thuộc hàm tương đương

Hai tập phụ thuộc hàm F và G dựa trên Q được gọi tương đương. Ký hiệu là $F \equiv G$ nếu $F^+ = G^+$

$F \equiv G$ thì F được gọi là 1 phủ của G, hay G là một phủ của F.

Để chứng minh $F \equiv G$ ta đi chứng minh:

$$(i) \quad \forall X \rightarrow Y \in F \Rightarrow X \rightarrow Y \in G^+$$

Để chứng minh điều này ta tính X_G^+ , nếu $Y \subseteq X_G^+$ thì $X \rightarrow Y \in G^+$

$$(ii) \quad \forall W \rightarrow Z \in G \Rightarrow W \rightarrow Z \in F^+$$

Tương tự ta tính W_F^+ , nếu $Z \subseteq W_F^+$ thì $W \rightarrow Z \in F^+$

5.2. Phủ tối thiểu

- Cho F là tập các phụ thuộc hàm dựa trên Q và một tập các RBTV dạng phụ thuộc hàm.

- Điều quan trọng ở đây là khi ta cập nhật CSDL thì hệ thống phải bảo đảm tất cả các phụ thuộc hàm trên không bị vi phạm.

- Ta biết rằng từ F ban đầu ta có tìm ra nhiều tập F_i tương đương với F bằng cách suy từ các phụ thuộc hàm của F. Quan hệ thoả các F_i thì cũng thoả F và ngược lại.

Ví dụ: $F = \{A \rightarrow B, B \rightarrow C\}$

Thì ta cũng có : $F_1 = \{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$, $F_2 = \{A \rightarrow B, B \rightarrow C, A \rightarrow C, AB \rightarrow BC\}$, ...

và $F_1 \equiv F$, $F_2 \equiv F$

- Vấn đề được đặt ngược lại là nếu cho F thì ta có thể tìm ra được tập phụ thuộc hàm đơn giản hơn F và tương đương với F.

- Tập phụ thuộc hàm có ít phụ thuộc hàm hơn F và các phụ thuộc hàm đó cũng đơn giản hơn. Và như vậy hệ thống dễ dàng kiểm tra hơn.

Định nghĩa thuộc tính dư thừa (Extraneous):

Một thuộc tính được gọi là dư thừa trong tập phụ thuộc hàm F nếu như ta bỏ nó ra khỏi các phụ thuộc hàm mà bao đóng của F vẫn không đổi.

Cho $X \rightarrow Y$ là một phụ thuộc hàm của F

- A là thuộc tính dư thừa trong X nếu $A \in X$ và $F \equiv (F - \{X \rightarrow Y\}) \cup \{(X-A) \rightarrow Y\}$

- B là thuộc tính dư thừa trong Y nếu $B \in Y$ và $F \equiv (F - \{X \rightarrow Y\}) \cup \{X \rightarrow (Y-B)\}$

Phủ tối thiểu của F ký hiệu là F_c là tập phụ thuộc hàm tương đương với F và thoả các tính chất sau:

- Không có phụ thuộc hàm nào trong F_c chứa các thuộc tính dư thừa.
- Không có 2 phụ thuộc nào cùng về trái.

5.3. Thuật toán tìm phủ tối thiểu

Thuật toán tìm phủ tối thiểu từ F.

Repeat

Sử dụng công thức hợp để thay thế các phụ thuộc hàm có cùng về trái:

$$X_1 \rightarrow Y_1 \text{ và } X_1 \rightarrow Y_2 \Rightarrow X_1 \rightarrow Y_1 Y_2$$

Nếu có một phụ thuộc hàm $X \rightarrow Y$ có các thuộc tính dư thừa bên về trái hay về phải thì xoá nó khỏi $X \rightarrow Y$

Until F không thay đổi

Ví dụ: Cho F là tập phụ thuộc hàm trên lược đồ quan hệ (ABC) như sau:

$$F = \{A \rightarrow BC, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$$

Tìm phủ tối thiểu của F

- Có hai phụ thuộc hàm cùng về trái: $A \rightarrow BC, A \rightarrow B$

Ta thay thế hai phụ thuộc hàm trên bằng phụ thuộc hàm $A \rightarrow BC$

$$\text{Tập phụ thuộc hàm mới: } F = \{A \rightarrow BC, B \rightarrow C, AB \rightarrow C\}$$

- A là thuộc tính dư thừa trong $AB \rightarrow C$ vì $F \equiv (F - \{AB \rightarrow C\}) \cup \{B \rightarrow C\}$

Do đó $AB \rightarrow C$ được thay bằng $B \rightarrow C$

$$\text{Tập phụ thuộc hàm mới là: } F = \{A \rightarrow BC, B \rightarrow C\}$$

- C là thuộc tính dư thừa trong $A \rightarrow BC$ vì $F \equiv (F - \{A \rightarrow BC\}) \cup \{A \rightarrow B\}$

Do đó $A \rightarrow BC$ được thay thế bằng $A \rightarrow B$

$$\text{Tập phụ thuộc hàm mới là: } F = \{A \rightarrow B, B \rightarrow C\}$$

Vậy phủ tối thiểu của F là: $F_c = \{A \rightarrow B, B \rightarrow C\}$

6. Dạng chuẩn của lược đồ quan hệ

6.1. Một số khái niệm liên quan đến các dạng chuẩn

6.1.1. Phân rã

Từ một lược đồ quan hệ kém chất lượng ban đầu cùng với tập phụ thuộc hàm của nó ta tuân theo một nguyên tắc nào đó phân rã thành những lược đồ quan hệ chất lượng hơn.

Ví dụ: Phân rã lược đồ quan hệ GIANGDAY thành hai lược đồ TKB và GV

TKB(MONHOC, SOTIET, LOP)

GV(LOP, GV, HOCVI, DC)

Tình trạng dữ liệu của hai lược đồ trên như sau:

TKB = $\Pi_{\text{MONHOC, SOTIET, LOP}}(\text{GIANGDAY})$			GV = $\Pi_{\text{LOP, GV, HOCVI, DC}}(\text{GIANGDAY})$			
MONHOC	SOTIET	LOP	LOP	GV	HOCVI	DC
CSDL	75	TH1	TH1	N.V.A	C.N	Q1
CSDL	75	TH2	TH2	N.V.A	C.N	Q1
CSDL	45	TH1	TH1	N.V.B	T.S	Q3
CSDL	45	TH2	TH2	N.V.B	T.S	Q3

Sau đây là 2 rắc rối xảy ra

Để trả lời câu hỏi “Cho biết thông tin của giáo viên dạy CSDL của TH1” ta phải kết tự nhiên hai quan hệ TKB và GV. Kết quả như sau:

MONHOC	SOTIET	LOP	GV	HOCVI	DC
CSDL	75	TH1	N.V.A	C.N	Q1
CSDL	75	TH1	N.V.B	T.S	Q3
CSDL	75	TH2	N.V.A	C.N	Q1
CSDL	75	TH2	N.V.B	T.S	Q3
CTDL	45	TH1	N.V.B	T.S	Q3
CTDL	45	TH1	N.V.A	C.N	Q1
CTDL	45	TH2	N.V.B	T.S	Q3
CTDL	45	TH2	N.V.A	C.N	Q1

Ta thấy rằng có tới hai giáo viên dạy môn CSDL của lớp TH1 trong khi thông tin ban đầu chỉ có N.V.A

→ **Vấn đề này gọi là phân rã không bảo toàn thông tin.**

Xét phụ thuộc hàm trên lược đồ phân rã:

TKB(MONHOC, SOTIET, LOP) MONHOC → SOTIET

GV(LOP, GV, HOCVI, DC) GV → HOCVI, DẠNG CHUẨN

Từ hai phụ thuộc hàm trên ta không thể suy ra được phụ thuộc hàm MONHOC, LOP → GV. Như vậy, hai phụ thuộc hàm trên không đảm bảo kiểm tra các ràng buộc toàn vẹn do 3 phụ thuộc hàm ban đầu gây ra nên.

→ **Vấn đề này gọi là phân rã không bảo toàn phụ thuộc hàm.**

Sau đây, ta sẽ xét các quy tắc phân rã sao cho không vi phạm hai vấn đề trên.

6.1.2. Phân rã bảo toàn thông tin.

Cho lược đồ quan hệ Q. Ta có định nghĩa sau:

Tập $\{Q_1, Q_2, \dots, Q_n\}$ là một phân rã của Q nếu:

$$Q = Q_1 \cup Q_2 \cup \dots \cup Q_n$$

Một cách tổng quát TQ là một quan hệ của Q thì:

$$TQ \subseteq \Pi_{R_1}(TQ) \bowtie \Pi_{R_2}(TQ) \bowtie \dots \bowtie \Pi_{R_n}(TQ)$$

Phân rã thông tin trên bảo toàn thông tin nếu:

$$TQ = \Pi_{R_1}(TQ) \bowtie \Pi_{R_2}(TQ) \bowtie \dots \bowtie \Pi_{R_n}(TQ)$$

Điều kiện để phân rã bảo toàn thông tin

Cho Q và F là tập phụ thuộc hàm định nghĩa trên Q

Q_1 và Q_2 là một phân rã bảo toàn thông tin trên Q nếu thỏa một trong hai phụ thuộc hàm sau:

$$Q_1 \cap Q_2 \rightarrow Q_1 \setminus Q_2 \text{ hoặc } Q_1 \cap Q_2 \rightarrow Q_2 \setminus Q_1$$

Vì vậy nếu $X \rightarrow Y \in F^+$ thì phân rã sau sẽ bảo toàn thông tin

$$Q_1(XY)$$

$$Q_2(R-Y)$$

Thật vậy, vì Q_1 có $X \rightarrow Y$ và $Q_1 \cap Q_2 = X$, $Q_2 \setminus Q_1 = Y$ do đó $Q_1 \cap Q_2 \rightarrow Q_1 \setminus Q_2$

Ví dụ:

Lược đồ GIANGDAY nếu phân rã thành hai lược đồ sau thì bảo tồn thông tin

$$Q_1(\text{MONHOC}, \text{SOTIET}, \text{LOP}, \text{GV})$$

$$Q_2(\text{GV}, \text{HOCVI}, \text{DC})$$

vì $Q_1 \cap Q_2 = \text{GV}$ mà $\text{GV} \rightarrow \text{HOCVI}, \text{DẠNG CHUẨN}$

Phương tiện để kiểm tra phân rã bảo toàn thông tin:

- Phương tiện để kiểm tra phân rã bảo toàn thông tin là kỹ thuật Tableau: là một bảng T như sau:

	1	2	...	m
Q1				
Q2				
...				
Qn				

- m cột cho m thuộc tính của Q
- n dòng cho n quan hệ phân rã
- $(i,j) = a_j$ nếu Q_i có chứa thuộc tính thứ j của Q
 $= b_k$ nếu ngược lại, k bắt đầu bằng 1 và tăng dần

Áp dụng luật phụ thuộc hàm để biến đổi bảng T thành T^* theo thuật toán sau:

While ($X \rightarrow A \in F$) do

Begin

Chọn dòng W1 và W2 sao cho $W1.X = W2.X$

If $W1.A \neq W2.A$ then

Begin

Th1: $W1.A = a_j$ và $W2.A = b_k$ thay $W2.A$ bằng $W1.A$

Th2: $W1.A = b_k$ và $W2.A = a_j$ thay $W1.A$ bằng $W2.A$

Th3: $W1.A = a_j$ và $W2.A = b_k$, thay $W2.A$ bằng $W1.A$

End

End

Cuối cùng xem bảng kết quả nếu trong bảng xuất hiện hàng gồm các kí hiệu $a_1, a_2, a_3, \dots, a_m$ thì phân rã bảo toàn thông tin.

Ví dụ: Kiểm tra $Q1(\text{MONHOC}, \text{SOTIET}, \text{LOP}, \text{GV}), Q2(\text{GV}, \text{HOCVI}, \text{DC})$

$F = \{\text{MONHOC} \rightarrow \text{SOTIET}; \text{MONHOC}, \text{LOP} \rightarrow \text{GV}; \text{GV} \rightarrow \text{HOCVI}, \text{DẠNG CHUẨN}\}$

T

	MONHOC	SOTIET	LOP	GV	HOCVI	DC
Q1	a_1	a_2	a_3	a_4	b_1	b_2
Q2	b_3	b_4	b_5	a_4	a_5	a_6

Từ $\text{GV} \rightarrow \text{HOCVI}$, DẠNG CHUẨN ta thay thế b_1 thành a_5 và b_2 thành a_6

T^*

	MONHO C	SOTIET	LOP	GV	HOCVI	DC
Q1	a_1	a_2	a_3	a_4	a_5	a_6

Q2	b_3	b_4	b_5	a_4	a_5	a_6
----	-------	-------	-------	-------	-------	-------

Như vậy ta được dòng thứ nhất toàn a_j suy ra phân rã trên bảo toàn thông tin

6.1.3 Phân rã bảo toàn phụ thuộc hàm

Cho lược đồ quan hệ Q và tập phụ thuộc hàm F xác định trên Q

Phân rã Q thành $\{Q_1, Q_2, \dots, Q_n\}$ thì mỗi Q_i sẽ xác định một tập phụ thuộc hàm F_i :

$$F_i = \{X \rightarrow Y : XY \subseteq Q_i \text{ và } X \rightarrow Y \in F^+\}$$

F_i được gọi là tham chiếu của F^+ lên Q_i

Phân rã trên bảo toàn phụ thuộc hàm nếu:

$$\text{Đặt } F' = F_1 \cup F_2 \cup \dots \cup F_n$$

$$\text{thì } F' = F \text{ (nghĩa là } F'^+ = F^+)$$

Để kiểm tra phân rã bảo toàn phụ thuộc hàm ta đi kiểm tra $F_1 \cup F_2 \cup \dots \cup F_n \equiv$

F

Lưu ý: Khi tính các F_i thường hay thiếu sót các phụ thuộc hàm vì F_i là chiếu của F^+ lên Q_i chứ không phải F lên Q_i . Như vậy để tính đầy đủ F_i của Q_i ta tính bao đóng của tất cả các tập con thực sự của Q_i .

$$X \subset Q_i. \text{ Nếu } X^+ \cap Q_i \neq X \text{ thì } (X \rightarrow (X^+ \cap (Q_i - X))) \in F_i.$$

Ví dụ:

$$\text{Cho } Q(ABCD), F = \{A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow A\}$$

Phân rã Q thành $\{Q_1(AB), Q_2(BC), Q_3(CD)\}$ sẽ dễ dàng nhầm lẫn:

$$Q_1(AB), F_1 = \{A \rightarrow B\}$$

$$Q_2(BC), F_2 = \{B \rightarrow C\}$$

$$Q_3(CD), F_3 = \{C \rightarrow D\}$$

$$\text{Lúc này } F' = F_1 \cup F_2 \cup F_3 = \{A \rightarrow B, B \rightarrow C, C \rightarrow D\}$$

Rõ ràng là F' không tương đương với F vì $F'^+ \neq F^+$ do $D \rightarrow A \notin F'^+$

Như vậy nếu vội vã kết luận phân rã trên không bảo toàn phụ thuộc hàm là sai

Thực ra:

$$Q_1(AB) A_f^+ = ABCD \Rightarrow A \rightarrow B \in F_1 \quad B_f^+ = BCDA \Rightarrow B \rightarrow A \in F_1$$

$$\text{Vậy } F_1 = \{A \rightarrow B, B \rightarrow A\}$$

$$Q_2(BC) B_f^+ = BCDA \Rightarrow B \rightarrow C \in F_2 \quad C_f^+ = CDAB \Rightarrow C \rightarrow B \in F_2$$

$$\text{Vậy } F_2 = \{B \rightarrow C, C \rightarrow B\}$$

$$Q_3(CD) C_f^+ = CDAB \Rightarrow C \rightarrow D \in F_3 \quad D_f^+ = DABC \Rightarrow D \rightarrow C \in F_3$$

$$\text{Vậy } F_3 = \{C \rightarrow D, D \rightarrow C\}$$

Vậy $F' = F1 \cup F2 \cup F3 = \{ A \rightarrow B, B \rightarrow A, B \rightarrow C, C \rightarrow B, C \rightarrow D, D \rightarrow C \}$

Ta tính được $F'^+ = F^+ \Rightarrow F' \equiv F$

Kết luận: Phân rã trên bảo toàn phụ thuộc hàm

6.1.4 Dạng chuẩn (Normal Form-NF)

Lược đồ quan hệ đạt được dạng chuẩn tốt nếu như việc kiểm tra các RBTV dưới dạng phụ thuộc hàm chỉ là việc kiểm tra khóa

Nếu xây dựng tập giá trị của khoá không trùng nhau (DBMS kiểm tra điều này) thì không có sự trùng lặp thông tin, nghĩa là không có bất cứ dữ liệu nào mà ta có thể suy ra được dựa vào các dữ liệu khác của quan hệ.

Xét dạng chuẩn dựa trên phụ thuộc hàm

- Thuộc tính khoá: Thuộc tính tham gia vào bất kỳ khoá nào đó của quan hệ chứa nó.

- Ngược lại gọi là thuộc tính không khoá.

Ví dụ: $Q(\underline{ABCDEF})$ A, B, D, E là các thuộc tính khoá. C, F là các thuộc tính không khoá.

- Thuộc tính đơn: Miền giá trị của nó không phải là tích hợp của các miền giá trị khác

- $X \rightarrow A$ là phụ thuộc hàm nguyên tố của tập F nếu: Không $\exists Y$ là tập con thực sự của X, $Y \rightarrow A \in F^+$

Ví dụ: Cho $F = \{AB \rightarrow C, B \rightarrow C\}$ thì:

$AB \rightarrow C$: không là phụ thuộc hàm nguyên tố vì có $B \rightarrow C$

$B \rightarrow C$: là phụ thuộc hàm nguyên tố

- A là thuộc tính phụ thuộc đầy đủ vào X nếu $X \rightarrow A$ là phụ thuộc hàm nguyên tố

Ví dụ trên thì C là thuộc tính phụ thuộc đầy đủ vào B chứ không phụ thuộc đầy đủ vào AB

6.2. Dạng chuẩn 1 (First Normal Form)

Lược đồ quan hệ Q ở dạng 1NF nếu tất cả các thuộc tính của Q đều là thuộc tính đơn.

Lược đồ CSDL C ở 1NF nếu tất cả các Q_i của C đều ở 1NF

Đây là dạng chuẩn đơn giản nhất, nó không chú ý đến các phụ thuộc hàm do đó có rất nhiều trùng lặp thông tin do các phụ thuộc hàm trên gây ra.

6.3. Dạng chuẩn 2 (Second Normal Form)

Lược đồ quan hệ Q ở dạng 2NF nếu ở 1NF và tất cả thuộc tính không khoá đều phụ thuộc đầy đủ vào khoá.

Lược đồ CSDL ở C dạng 2NF nếu tất cả Q_i của C đều ở 2NF

VD1: Cho $Q(\underline{AB}CD)$, $F = \{A \rightarrow C, B \rightarrow D\}$, không đạt 2NF vì:

Khoá chính là AB

C, D là hai thuộc tính không khoá

$AB \rightarrow C$ không là phụ thuộc hàm nguyên tố vì có $A \rightarrow C$

$AB \rightarrow D$ không là phụ thuộc hàm nguyên tố vì có $B \rightarrow D$

C và D không phụ thuộc đầy đủ vào khoá.

Xét một tình trạng Q có sự trùng lặp thông tin (các giá trị trong ngoặc là trùng lặp)

Q	<u>A</u>	<u>B</u>	C	D
	(a1)	b1	(c1)	d1
	(a1)	b2	(?)	d2
	a2	(b3)	c2	(d3)
	a3	(b3)	c3	(?)

VD2: Cho $Q(\underline{AB}CD)$, $F = \{AB \rightarrow C, C \rightarrow D\}$ ở 2NF vì:

Khoá chính là AB.

C, D là hai thuộc tính không khoá. $AB \rightarrow C$ và $AB \rightarrow D$ đều là các phụ thuộc hàm nguyên tố.

$\Rightarrow C$ và D đều là phụ thuộc đầy đủ vào khoá.

Xét một tình trạng Q có sự trùng lặp thông tin:

Q	<u>A</u>	<u>B</u>	C	D
	a1	b1	(c1)	(d1)
	a2	b2	(c1)	(?)

Ta thấy rằng ở VD2, $C \rightarrow D$ gây ra trùng lặp thông tin vì thuộc tính không khoá D phụ thuộc bắc cầu vào khoá (nghĩa là phụ thuộc hàm **khoá** $\rightarrow D$ suy diễn nhờ qui tắc bắc cầu Armstrong)

6.4. Dạng chuẩn 3 (Third Normal Form)

Lược đồ quan hệ Q ở dạng 3NF nếu ở 2NF và tất cả các thuộc tính không khoá không phụ thuộc bắc cầu vào khoá.

Định nghĩa này có thể định nghĩa lại như sau:

Lược đồ quan hệ Q ở dạng 3NF nếu ở 2NF và tất cả phụ thuộc hàm không hiển nhiên $X \rightarrow Y$ của F^+ thoả một trong hai điều kiện sau:

- (i) X là một siêu khoá (X chứa một khoá nào đó)
- (ii) Mỗi thuộc tính trong tập (Y - X) nằm trong một khoá nào đó

Lược đồ CSDL C ở dạng 3NF nếu tất cả các Q_i của C đều ở dạng 3NF

VD1: Cho $Q(\underline{AB}CD)$, $F = \{AB \rightarrow CD\}$ ở dạng 3NF

Vì $AB \rightarrow CD$ có vế trái là một siêu khoá.

VD2: Cho $Q(\underline{ABCDE})$, $F = \{AB \rightarrow CDE, B \rightarrow D, DE \rightarrow ABC\}$ ở dạng 3NF vì:

$AB \rightarrow CD$ có vế trái là một siêu khoá.

$B \rightarrow D$ có $(VP) - (VT) = D$ chứa trong khoá DE .

$DE \rightarrow ABC$ có vế trái là một siêu khoá.

Xét một tình trạng Q có sự trùng lặp thông tin:

Q	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>E</u>
	a1	(b1)	c1	(d1)	e1
	a2	(b1)	c2	(?)	e1

Ở dạng 3NF nếu có nhiều khoá sẽ có khả năng trùng lặp thông tin do các phụ thuộc hàm loại (ii) trong định nghĩa dạng chuẩn 3.

6.5. Dạng chuẩn BC (Boyce codd Normal Form)

Lược đồ quan hệ Q ở BCNF nếu ở dạng 1NF và tất cả phụ thuộc hàm không hiển nhiên $X \rightarrow Y$ của F^+ thì X là một siêu khoá (X chứa một khoá nào đó).

Lược đồ CSDL C ở dạng BCNF nếu tất cả các Q_i của C đều ở dạng BCNF.

VD: Cho $Q(\underline{ABCD})$, $F\{AB \rightarrow CD, D \rightarrow AB\}$ ở dạng BCNF vì:

$AB \rightarrow CD$ có vế trái là một siêu khoá

$D \rightarrow AB$ có vế trái là một siêu khoá

$D \rightarrow C \in F^+$ vế trái là một siêu khoá

Ở dạng chuẩn BCNF không có sự trùng lặp thông tin do phụ thuộc hàm gây ra. Tuy nhiên dạng BCNF có điều kiện quá khắt khe, đôi khi người ta chỉ cần chấp nhận ở dạng 3NF.

❖ TÓM TẮT CHƯƠNG 5

Trong chương này, một số nội dung chính được giới thiệu:

- ❖ Khái niệm cơ sở của lý thuyết cơ sở dữ liệu như khoá, phụ thuộc hàm, bao đóng, các dạng chuẩn, ...
- ❖ Thiết kế được dữ liệu ở mức tốt nhất (có thể ứng dụng được) bằng các phép tách, giải thuật chuẩn hóa lược đồ.

❖ CÂU HỎI VÀ BÀI TẬP CHƯƠNG 5

Bài tập 1:

Xác định khoá và xét các phân rã sau đây theo hai tiêu chuẩn bảo toàn thông tin và

bảo toàn phụ thuộc hàm.

a. $Q(ABCD), F = \{A \rightarrow BC, C \rightarrow D\}$

$Q_1(AB), F_1 = \{A \rightarrow B\}$

$Q_2(CD), F_2 = \{C \rightarrow D\}$

b. $Q(ABCD), F = \{A \rightarrow B, AC \rightarrow D\}$

$Q_1(AB), F_1 = \{A \rightarrow B\}$

$Q_2(ADC), F_2 = \{AC \rightarrow D\}$

c. $Q(ABDCE), F = \{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$

$Q_1(AB), F_1 = \{A \rightarrow D\}$

$Q_2(CD), F_2 = \emptyset$

$Q_3(AB), F_3 = \emptyset$

$Q_4(CD), F_4 = \{C \rightarrow D, DE \rightarrow C, CE \rightarrow D\}$

$Q_5(AB), F_5 = \emptyset$

d. $Q(ABCD), F = \{A \rightarrow B, C \rightarrow D\}$

$Q_1(AB), F_1 = \{A \rightarrow B\}$

$Q_2(CD), F_2 = \{C \rightarrow D\}$

Bài tập 2:

Xét dạng chuẩn của các lược đồ quan hệ sau:

a. $Q(ABCD), F = \{A \rightarrow B, C \rightarrow D\}$

b. $Q(ABCD), F = \{AB \rightarrow C, C \rightarrow D\}$

c. $Q(ABCD), F = \{AB \rightarrow CD, CD \rightarrow AB, C \rightarrow B\}$

d. $Q(ABCD), F = \{AB \rightarrow CD, D \rightarrow E, DE \rightarrow ABC\}$

e. $Q(ABCDEF), F = \{AB \rightarrow E, AC \rightarrow F, AD \rightarrow B, B \rightarrow C, C \rightarrow D\}$

Bài tập 3:

Cho lược đồ quan hệ $Q(ABCDEF), F = \{C \rightarrow F, E \rightarrow A, CE \rightarrow D, A \rightarrow B\}$

a. Xác định khoá của Q

b. Phân rã thành dạng chuẩn Boyce-Codd bảo toàn thông tin

c. Phân rã thành dạng chuẩn 3 bảo toàn thông tin và bảo toàn phụ thuộc hàm.

Bài tập 4:

Giả sử bài 3 được phân rã thành $Q_1(CDEF)$ và $Q_2(ABE)$, hãy xác định F_1, F_2 và đánh giá phân rã trên.

Bài tập 5:

Nếu bài 3 được phân rã thành $Q_1(CF), Q_2(AE), Q_3(CDE), Q_4(AB)$. Hãy xác định F_1, F_2, F_3, F_4 và đánh giá chúng.

Bài tập 6:

Cho lược đồ quan hệ

VẬNCHUYỀN(TÀU, LOẠITÀU, CHUYẾN, HÀNG, CẢNG, NGÀY)

Mỗi tàu (TÀU) thuộc duy nhất một loại tàu nào đó (LOẠITÀU), mỗi chuyến có một mã số riêng biệt (CHUYẾN) dùng để xác định một chuyến tàu (TÀU) chở một khối lượng hàng hoá nào đó (HÀNG), mỗi chiếc tàu trong một ngày(NGÀY) chỉ cập vào một cảng duy nhất (CẢNG) của một chuyến vận chuyển nào đó (CHUYẾN)

- a. Xác định tập các phụ thuộc hàm trên.
- b. Xác định dạng chuẩn của VẬNCHUYỀN
- c. Nếu VẬNCHUYỀN chưa tốt hãy tìm một phân rã tốt cho nó.

Bài tập 3: Xác định khóa của các lược đồ quan hệ sau:

Q1 (A,B,C,D,E,H)

với F = { AB-> C; CD-> E; AH-> B; B-> D; A-> D }

Q2 (A,B,C,D,M,N,P,Q)

với F = { AM-> NB; BN-> CM; A-> P; D-> M; PC-> A; DQ-> A }

Q3 (M, N, P, Q, R, S, T, U, W)

với F = { M-> W; MR-> T; T-> R; QR-> T; M-> U; MT-> P; NP-> Q; UW-> R }

TÀI LIỆU THAM KHẢO:

1. PGS Phạm Văn Ất, Hướng dẫn sử dụng Microsoft Access 97, NXB KHK, 1999
2. Vũ Đức Thi, Cơ sở dữ liệu kiến trúc và thực hành. NXB thống kê, Hà nội 1997.
3. Đỗ Trung Tuấn, *Cơ sở dữ liệu (DataBase)*, NXB Giáo dục Hà nội, 1998.
4. TS. Trần Văn Tư, Microsoft SQL Server 7.0, NXB Thống kê, 2000
5. Trần Thành Trai, *Nhập môn Cơ sở dữ liệu*, NXB Giáo dục, TP.Hồ Chí Minh 1996.
6. Lê Tiến Vương. *Nhập môn cơ sở dữ liệu*. NXB Thống kê Hà nội, 2000. Tái bản lần 5.
7. *Introduction to Oracle SQL and PL/SQL Using Procedure Builder*. Vol 1,2,3,4. ORACLE 7.3. 1996
8. *Introduction to Oracle: SQL, SQL*PLUS and PL/SQL*
9. Education Services. Vol 1,2,3,4,5,6,7. ORACLE 1992