



Thiết kế CSDL quan hệ

Vũ Tuyết Trinh

trinhvt@soict.hust.edu.vn

Bộ môn Hệ thống thông tin
Viện CNTT&TT, ĐHBKHN



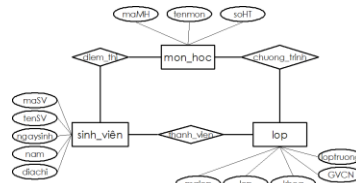
Các cách tiếp cận

- Trên xuống (*Top-down*)
 - 1. Mô hình hóa dữ liệu (ER)
 - 2. Chuyển đổi sơ đồ dữ liệu lưu trữ (quan hệ)
 - 3. Rà soát, hiệu chỉnh
- Dưới lên (*bottom-up*)
 - 1. Biểu diễn dữ liệu người dùng (biểu mẫu, báo cáo) dưới dạng các quan hệ
 - 2. Chuẩn hoá các quan hệ này
 - 3. Ghép các quan hệ có cùng khoá chính

Cách tiếp cận top-down



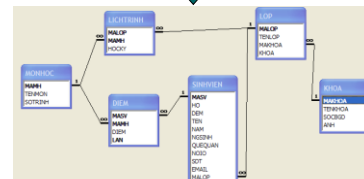
1: PHÂN TÍCH



Mô tả ứng dụng

Mô hình hoá DL (vd: Sơ đồ thực thể-liên kết)

2: THIẾT KẾ



3: CÀI ĐẶT



Cài đặt với 1 hệ quản trị CSDL
(vd: ORACLE)

Mô tả DL logic với 1 mô hình DL cụ thể
(vd: Sơ đồ quan hệ)

Mô hình hoá dữ liệu với mô hình thực thể - liên kết

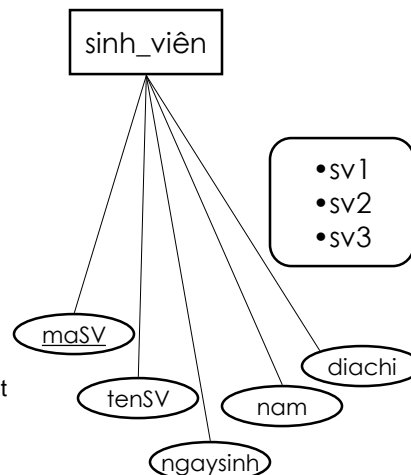
Đặc điểm

- Thích hợp để mô hình hoá dữ liệu cho CSDL
 - Gần gũi với nhận thức của con người → dễ sử dụng
 - dễ chuyển đổi sang mô hình quan hệ
 - Dựa trên các khái niệm chính
 - **Thực thể**: một đối tượng trong thế giới thực
 - **Tập thực thể**: các thực thể có cùng các tính chất
 - **Thuộc tính**: một đặc tính của một tập thực thể
 - **Khoá**: xác định sự duy nhất của 1 thực thể
 - **Liên kết**: mối liên hệ có nghĩa giữa nhiều thực thể
 - **Tập liên kết**: tập hợp các liên kết cùng kiểu
- Được biểu diễn bởi **sơ đồ thực thể - liên kết**

5

Thực thể và thuộc tính

- **Thực thể**: một đối tượng trong thế giới thực
- **Tập thực thể**: gồm các thực thể có tính chất giống nhau
- **Thuộc tính**: một đặc tính của một tập thực thể
 - **Miền giá trị** ~ tập các giá trị có thể
 - **Khoá** ~ xác định sự duy nhất của 1 thực thể



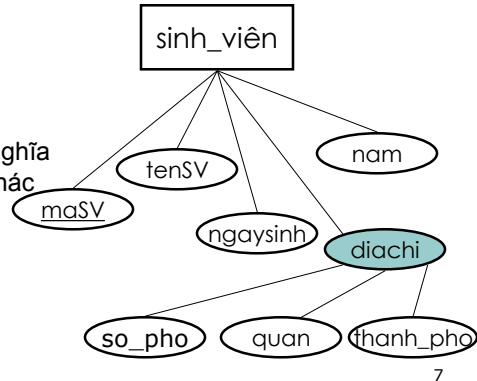
6

Kiểu thuộc tính

- Thuộc tính đơn giản (thuộc tính nguyên tố)
 - có kiểu dữ liệu nguyên tố

tenSV = "Trần T. Bình"
tenSV = "Ng. Đ. Trung"

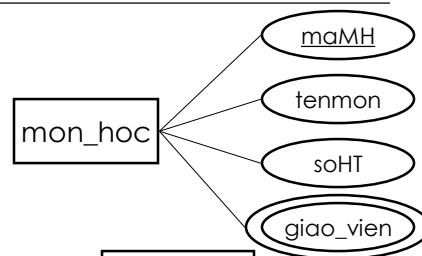
- Thuộc tính phức
 - có kiểu phức, định nghĩa bởi các thuộc tính khác



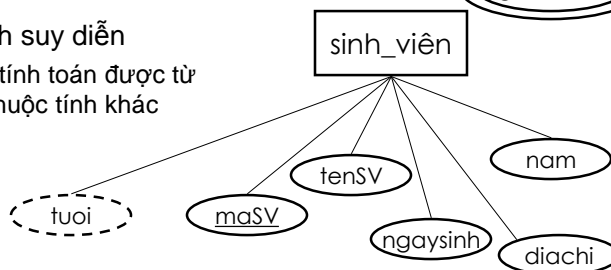
7

Kiểu thuộc tính (2)

- Thuộc tính đa giá trị
 - tương ứng với mỗi thực thể, có thể nhận nhiều giá trị



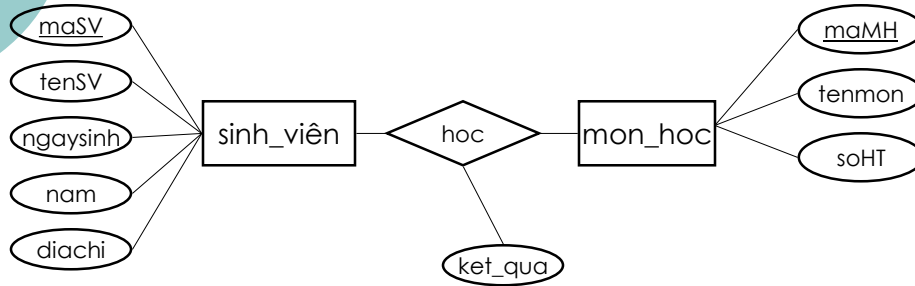
- Thuộc tính suy diễn
 - có thể tính toán được từ (các) thuộc tính khác



8

Liên kết

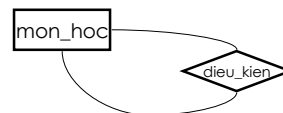
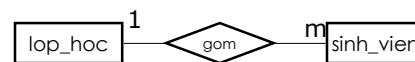
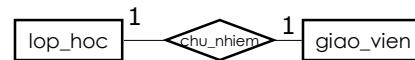
- Đ/n: là sự kết hợp giữa một số thực thể
- Thuộc tính



9

Ràng buộc của kết nối

- **1-1**: Liên kết 1 thực thể của một tập thực thể với nhiều nhất 1 thực thể của tập thực thể khác
- **1-n**: Liên kết 1 thực thể của một tập thực thể với nhiều thực thể của tập thực thể khác
- **n-m**: Liên kết 1 thực thể của một tập thực thể với nhiều thực thể của tập thực thể khác và ngược lại
- **đệ quy**: Liên kết giữa các thực thể cùng kiểu



10



Lập sơ đồ thực thể - liên kết

- B1: Xác định các thực thể
- B2: Xác định các liên kết giữa các thực thể
 - Bậc của liên kết
 - Ràng buộc (1-1, 1-n, n-m, đệ quy)

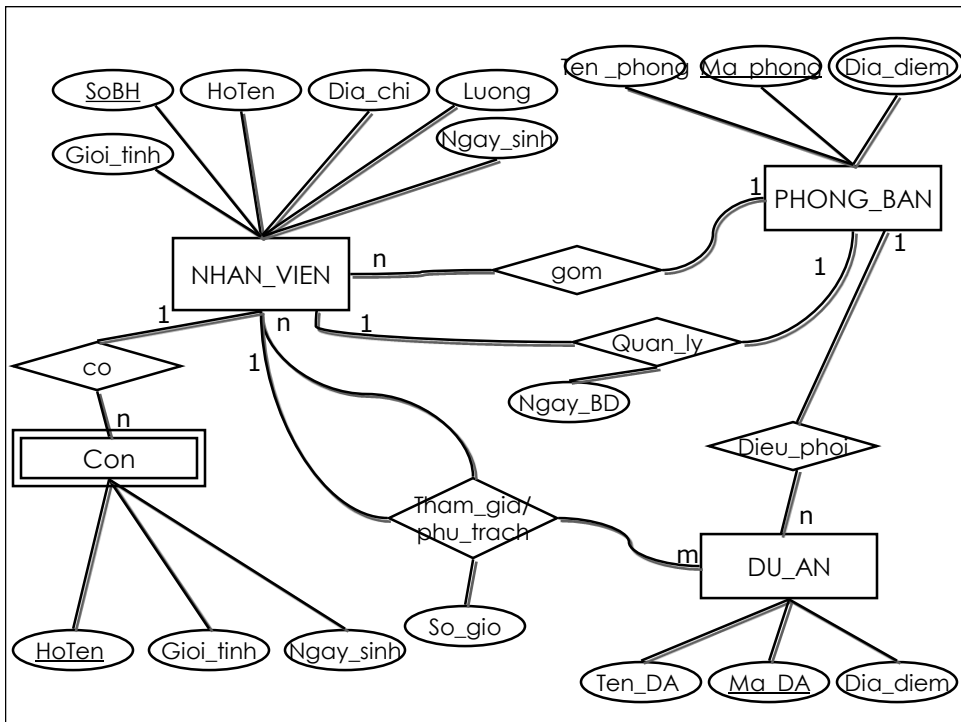
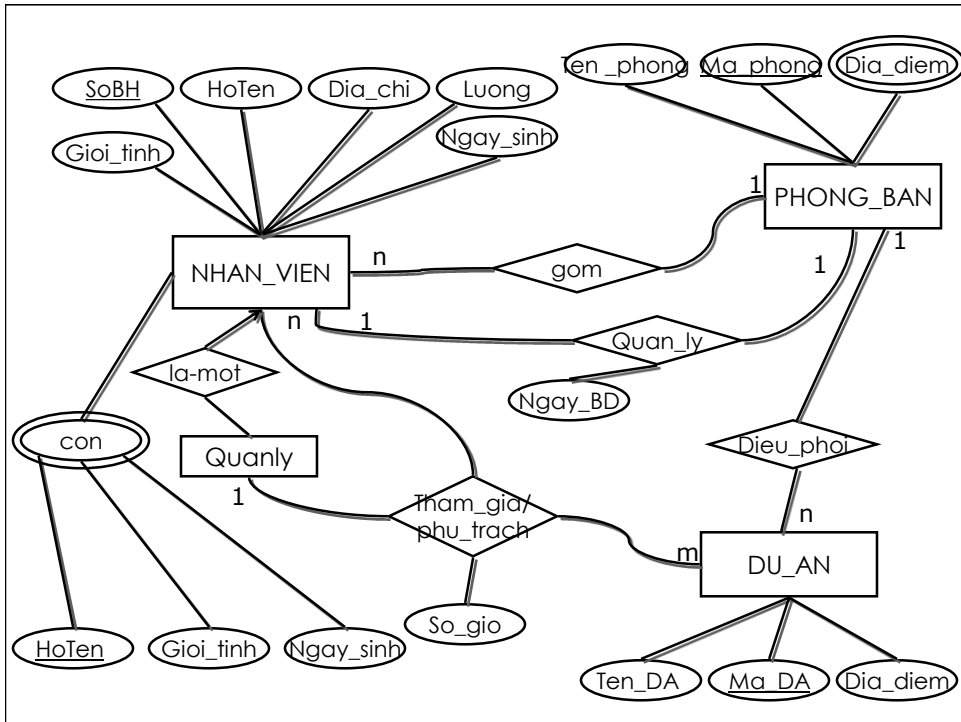
11



Bài tập

- Bài toán: phân tích và thiết kế 1 CSDL gồm các thông tin trong 1 công ty (nhân viên, phòng ban, dự án)
 - Công ty được tổ chức bởi các phòng ban. Mỗi phòng ban có 1 tên duy nhất, 1 số duy nhất và 1 người quản lý (thời điểm bắt đầu công tác quản lý của người này cũng được lưu lại trong CSDL). Mỗi phòng ban có thể có nhiều trụ sở làm việc khác nhau
 - Mỗi phòng điều phối một số dự án. Mỗi dự án có 1 tên và 1 mã số duy nhất, thực hiện tại một địa điểm duy nhất
 - Các thông tin về nhân viên cần được quan tâm gồm: tên, số bảo hiểm, địa chỉ, lương, giới tính, ngày sinh. Mỗi nhân viên làm việc tại một phòng ban nhưng có thể tham gia nhiều dự án điều phối bởi các phòng ban khác nhau. Thông tin về số giờ làm việc trong từng dự án (theo tuần) cũng như người quản lý trực tiếp của các nhân viên cũng được lưu trữ
 - Thông tin về con cái của từng nhân viên: tên, giới tính, ngày sinh

12



Biến đổi: Sơ đồ thực thể - liên kết → Sơ đồ quan hệ

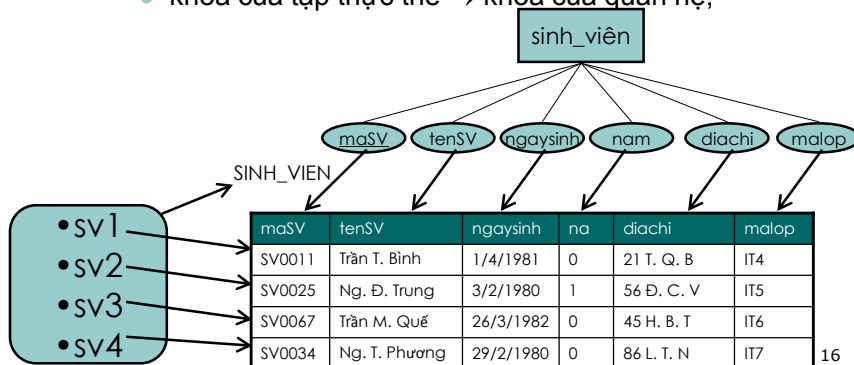
- Biến đổi tập các thực thể
- Biến đổi các liên kết
- Các khoá của các sơ đồ quan hệ
- Các sơ đồ quan hệ với khoá chung

15

Biến đổi các tập thực thể

B1: 1 tập thực thể

- 1 quan hệ,
 - thuộc tính → thuộc tính (trường),
 - 1 thực thể → 1 bộ
 - khoá của tập thực thể → khoá của quan hệ,



16

Biến đổi các tập thực thể (2)

B2: 1 tập thực thể xác định từ tập thực thể khác (E) qua 1 liên kết

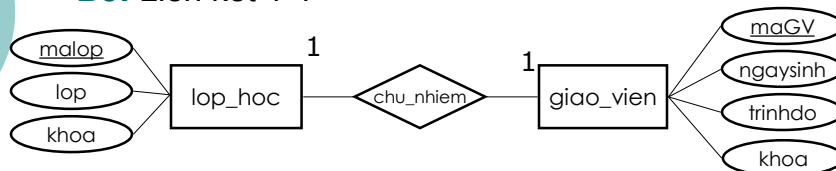


- 1 quan hệ chứa khoá của E
LOPTRUONG(maSV)

17

Biến đổi các liên kết

B3: Liên kết 1-1

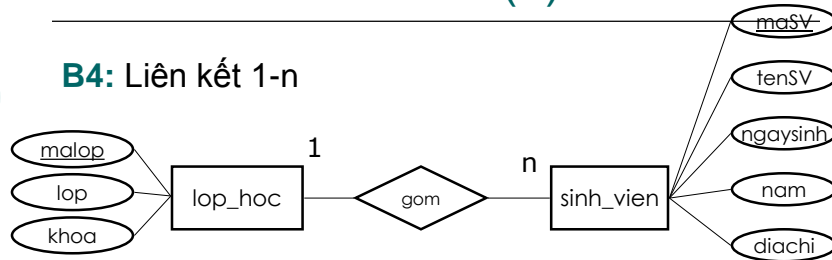


- Dùng khoá ngoài
LOP_HOC(malop, lop, khoa, maGV)

18

Biến đổi các liên kết (2)

B4: Liên kết 1-n

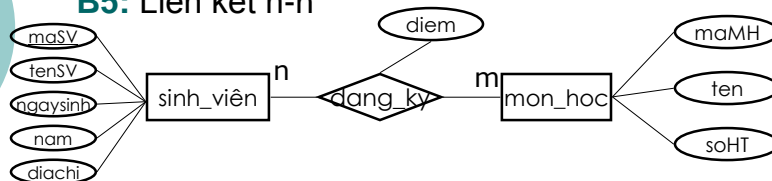


- Dùng khoá ngoài: thêm khoá chính của quan hệ bên 1 vào quan hệ bên n làm khoá ngoài
`SINH_VIEN(maSV, tenSV, ngaysinh, nam, diachi, malop)`

19

Biến đổi các liên kết (3)

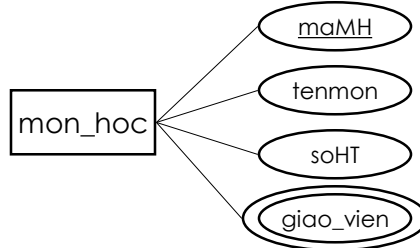
B5: Liên kết n-n



- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khóa của các thực thể có liên quan và các thuộc tính của liên kết
`DANG_KY(maSV, maMH, diem)`

20

Thuộc tính đa trị



B6: Với mỗi thuộc tính đa trị

- Thêm 1 quan hệ mới xác định bởi thuộc tính đa trị và khoá của tập thực thể tương ứng

MH_GV(maMH,giao_vien)

21

Bài tập

- Biến đổi sơ đồ thực thể/liên kết → sơ đồ quan hệ

22



Một số điểm lưu ý

- Điểm khác nhau của các mô hình dữ liệu
 - khả năng biểu diễn dữ liệu về mặt ngữ nghĩa
 - khả năng biểu diễn truy vấn dữ liệu
 - hiệu quả của cài đặt trong máy tính
- Mô hình thực thể-liên kết cho phép biểu diễn dữ liệu gần với nhận thức của con người
- Mô hình quan hệ
 - Cho phép biểu diễn logic dữ liệu
 - dễ ánh xạ sang cấu trúc lưu trữ vật lý
 - Dựa trên nền tảng toán học cho phép tối ưu hoá các truy xuất dữ liệu
- Các bước xây dựng một CSDL
- Biến đổi từ sơ đồ thực thể - liên kết sang lược đồ quan hệ



Cách tiếp cận bottom-up

- Xuất phát điểm
- Mục đích của chuẩn hoá là gì?
- Thế nào là chuẩn? Có bao nhiêu chuẩn?



Ví dụ

- 1 CSDL về các hãng cung ứng.
Suppliers(sid, sname, city, NOE, product, quantity)

Sid	Sname	City	NOE	Product	quantity
S1	Smith	London	100	Screw	50
S1	Smith	London	100	Nut	100
S2	J&J	Paris	100	Screw	78
S3	Blake	Tokyo	75	Bolt	100

- Các vấn đề đặt ra
- Đề xuất các giải pháp

25



Mục đích của chuẩn hoá

- Xác định được 1 tập các lược đồ quan hệ cho phép tìm kiếm thông tin một cách dễ dàng, **đồng thời** tránh được dư thừa dữ liệu
- Hướng tiếp cận:
Tách các lược đồ quan hệ “có vấn đề” thành những lược đồ quan hệ “chuẩn hơn”

26



Nội dung

- Phụ thuộc hàm
- Phép tách các sơ đồ quan hệ
- Các dạng chuẩn
- Phụ thuộc đa trị
- Kết luận

27



Phụ thuộc hàm (*Functional dependencies - FD*)

- Đ/N Phụ thuộc hàm trong 1 quan hệ
Cho
 - $R(U)$ là 1 sơ đồ quan hệ, U là tập các thuộc tính.
 - $X, Y \subseteq U$ X xác định hàm Y hay Y phụ thuộc hàm vào X nếu
 - với \forall quan hệ r xác định trên $R(U)$ và với 2 bộ t_1 và t_2 bất kỳ mà $t_1[X] = t_2[X]$ thì $t_1[Y] = t_2[Y]$.
- Ký hiệu: $X \rightarrow Y$

28



Ví dụ

Suppliers(sid, sname, city, NOE, product, quantity)

Supp(sid, sname, city, NOE)

- $\text{sid} \rightarrow \text{sname}$
- $\text{sid} \rightarrow \text{city}$
- $\text{sid} \rightarrow \text{NOE}$

Supply(sid, product, quantity)

- $\text{sid, product} \rightarrow \text{quantity}$

29



Hệ tiên đề Amstrong

Cho

- $R(U)$ là 1 sơ đồ quan hệ, U là tập các thuộc tính.
- $X, Y, Z, W \subseteq U$
(Ký hiệu: $XY = X \cup Y$)

- **Phản xạ** (*reflexivity*)

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

- **Tăng trưởng** (*augmentation*)

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

- **Bắc cầu** (*transitivity*)

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

30



Hệ quả

- Luật hợp (*union*)
Nếu $X \rightarrow Y$, $X \rightarrow Z$ thì $X \rightarrow YZ$.
- Luật tựa bắc cầu (*pseudotransitivity*)
Nếu $X \rightarrow Y$, $WY \rightarrow Z$ thì $XW \rightarrow Z$.
- Luật tách (*decomposition*)
Nếu $X \rightarrow Y$, $Z \subseteq Y$ thì $X \rightarrow Z$.

31



Bao đóng của 1 tập phụ thuộc hàm

- Đ/N : Bao đóng của tập phụ thuộc hàm F là tập lớn nhất các phụ thuộc hàm có thể được suy diễn logic từ F
 - Ký hiệu là F^+
- Suy diễn logic
 $X \rightarrow Y$ được suy diễn logic từ F nếu với mỗi quan hệ r xác định trên $R(U)$ thoả các phụ thuộc hàm trong F thì cũng thoả $X \rightarrow Y$
- F là họ đầy đủ (*full family*) nếu
 $F = F^+$

32



Khoá

- Đ/N: Cho lược đồ quan hệ $R(U)$, tập các phụ thuộc hàm F . $K \subseteq U$, K được gọi là **khóa tối thiểu** của R nếu như
 - $K \rightarrow U \in F^+$
 - với $\forall K' \subset K$ thì $K' \rightarrow U \notin F^+$
- Nhận xét: Nếu K là một khóa tối thiểu thì
 - $K^+ = U$
 - K là tập thuộc tính nhỏ nhất có tính chất như vậy.

33



Bao đóng của 1 tập các thuộc tính

- Đ/N **Bao đóng của tập thuộc tính** X là tập tất cả các thuộc tính được xác định hàm bởi X thông qua tập F
 - ký hiệu là X^+

$$X^+ = \{A \in U \mid X \rightarrow A \in F^+\}$$

34



Nhận xét

- Hệ tiên đề Amstrong là đúng đắn và đầy đủ
- $X \rightarrow Y$ được suy diễn từ hệ tiên đề Amstrong
 $\Leftrightarrow Y \subseteq X^+$
- Thiết kế CSDL ? Các khái niệm
 - Phụ thuộc hàm
 - Bao đóng của tập phụ thuộc hàm
 - Khoá
 - Bao đóng của 1 tập các thuộc tính

35



Tính bao đóng của 1 tập thuộc tính

- **Vào:** Tập hữu hạn các thuộc tính U
tập các phụ thuộc hàm F trên U
 $X \subseteq U$
- **Ra:** X^+
- **Thuật toán**
 - B⁰** $X^0 = X$.
 - Bⁱ** Tính X^i từ X^{i-1}
 - Nếu $\exists Y \rightarrow Z \in F \wedge Y \subseteq X^{i-1} \wedge A \in Z \wedge A \notin X^{i-1}$
thì $X^i = X^{i-1} \cup A$
 - ngược lại, $X^i = X^{i-1}$.
 - Nếu $X^i \neq X^{i-1}$
thì thực hiện Bⁱ
 - ngược lại, thực hiện Bⁿ
 - Bⁿ** $X^+ = X^i$

36



Tìm khoá tối thiểu

- **Vào:** $U = \{A_1, A_2, \dots, A_n\}$, F
- **Ra:** khóa tối thiểu K xác định được trên U và F
- **Thuật toán**
 - B^0 $K^0 = U$, $n = |U|$
 - B^i Nếu $(K^{i-1} \setminus \{A_i\}) \rightarrow U$
thì $K^i = K^{i-1} \setminus \{A_i\}$
ngược lại, $K^i = K^{i-1}$
Nếu $K^i \neq K^{i-1}$ v $i < n$
thì thực hiện B^i
ngược lại, thực hiện B^n
 - B^n $K = K^i$

37



Ví dụ

- Cho $R(U)$ trong đó $U = \{A, B, C, D, E, F, G\}$. $F = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G\}$
 - 1. Tìm một khóa tối thiểu của R
 - $K^0 = ABCDEFG$
 - $K^1 = K^0$ do nếu loại A thì $BCDEFG \rightarrow U$ không thuộc $F+$
 - $K^2 = K^1 \setminus \{B\} = ACDEFG$ do $ACDEFG \rightarrow U$ thuộc $F+$
 - $K^3 = K^2$ do nếu loại C thì $ADEFG \rightarrow U$ không thuộc $F+$
 - $K^4 = K^3$ do nếu loại D thì $ACEFG \rightarrow U$ không thuộc $F+$
 - $K^5 = K^4 \setminus \{E\} = ACDFG$ do $ACDFG \rightarrow U$ thuộc $F+$
 - $K^6 = K^5$ do nếu loại F thì $ACDG \rightarrow U$ không thuộc $F+$
 - $K^7 = K^6 \setminus \{G\} = ACDF$ do $ACDF \rightarrow U$ thuộc $F+$
- Vậy khóa tối thiểu cần tìm là $ACDF$

38



Nhận xét về phụ thuộc hàm

- từ một tập các phụ thuộc hàm có thể suy diễn ra các phụ thuộc hàm khác
 - trong một tập phụ thuộc hàm cho sẵn có thể có các phụ thuộc hàm bị coi là dư thừa.
- Làm thế nào để có được một tập phụ thuộc hàm tốt?

39



Tập phụ thuộc hàm tương đương

- Đ/N: Tập phụ thuộc hàm F là **phủ** của tập phụ thuộc hàm G hay G là phủ của F hay F và G **tương đương** nếu $F^+ = G^+$.
 - Ký hiệu là $F \approx G$
- Kiểm tra tính tương đương của 2 tập phụ thuộc hàm
 - B.1. Với mỗi $Y \rightarrow Z \in F$, $Z \subseteq Y^+$ (trên G) thì $Y \rightarrow Z \in G^+$
Nếu với $\forall f \in F, f \in G^+$ thì $F^+ \subseteq G^+$
 - B.2. Tương tự, nếu $\forall f \in G, f \in F^+$ thì $G^+ \subseteq F^+$
 - B.3. Nếu $F^+ \subseteq G^+$ và $G^+ \subseteq F^+$ thì $F \approx G$

40

Tập phụ thuộc hàm không dư thừa

- Đ/N: Tập phụ thuộc hàm F là **không dư thừa** nếu $\exists X \rightarrow Y \in F$ sao cho $F \setminus \{X \rightarrow Y\} \approx F$.
- Tìm phủ không dư thừa của 1 tập phụ thuộc hàm
 - Vào: Tập thuộc tính U , $F = \{L_i \rightarrow R_i : i = 1..n\}$
 - Ra : Phủ không dư thừa F' của F
 - Thuật toán
 - B⁰** $F^0 = F$, $n = |F|$
 - Bⁱ** Nếu $F^{i-1} \setminus \{L_i \rightarrow R_i\} \approx F^{i-1}$
thì $F^i = F^{i-1} \setminus \{L_i \rightarrow R_i\}$
ngược lại, $F^i = F^{i-1}$
Nếu $F^i \neq F^{i-1}$ v $i < n$
thì thực hiện Bⁱ
ngược lại, thực hiện Bⁿ
 - Bⁿ** $F' = F^i$

41

Phủ tối thiểu của 1 tập phụ thuộc hàm

- Đ/N: F_c được gọi là **phủ tối thiểu của 1 tập phụ thuộc hàm** F nếu thỏa mãn 3 điều kiện sau:
 - Đk1:** Với $\forall f \in F_c$, f có dạng $X \rightarrow A$,
trong đó A là 1 thuộc tính
 - Đk2:** Với $\forall f = X \rightarrow Y \in F_c$, $\exists A \in X$ (A là 1 thuộc tính):
 $(F_c \setminus f) \cup \{(X \setminus A) \rightarrow Y\} \approx F_c$
 - Đk3:** $\exists X \rightarrow A \in F_c : F_c \setminus \{X \rightarrow A\} \approx F_c$

42



Tính phủ tối thiểu

- **Vào:** Tập thuộc tính U , $F = \{L_i \rightarrow R_i; i = 1..n\}$
- **Ra:** phủ tối thiểu F_c của tập phụ thuộc hàm F
- **Thuật toán**
 - B.1.** Biến đổi F về dạng $F_1 = \{L_i \rightarrow A_j\}$
trong đó A_j là 1 thuộc tính bất kỳ thuộc U (thỏa mãn đk1)
 - B.2.** Loại bỏ thuộc tính thừa trong vế trái của các phụ thuộc hàm
Lần lượt giản ước từng thuộc tính trong vế trái của từng phụ thuộc hàm trong F_1 thu được F_1' . Nếu $F_1' \approx F_1$ thì loại bỏ thuộc tính đang xét
Khi không có sự giản ước nào xảy ra nữa ta thu được F_2 thỏa mãn đk2
 - B.3.** Loại bỏ phụ thuộc hàm dư thừa
Lần lượt loại kiểm tra từng phụ thuộc hàm f . Nếu $F_2 \setminus f \approx F_2$ thì loại bỏ f
Khi không có phụ thuộc hàm nào có thể loại bỏ thì thu được F_3 thỏa mãn đk3
 - B.4.** $F_c = F_3$

43



Mục đích của thiết kế CSDL – nhắc lại

- Xác định được 1 tập các lược đồ quan hệ cho phép **tìm kiếm thông tin một cách dễ dàng, đồng thời tránh được dư thừa dữ liệu** (cf. slide 7)
- Phát biểu lại mục đích này sử dụng các khái niệm vừa học ?

44



Phép tách các lược đồ quan hệ

- Mục đích
 - Thay thế một sơ đồ quan hệ $R(A_1, A_2, \dots, A_n)$ bằng một tập các sơ đồ con $\{R_1, R_2, \dots, R_k\}$ trong đó $R_i \subseteq R$ và $R = R_1 \cup R_2 \cup \dots \cup R_k$
- Yêu cầu của phép tách
 - Bảo toàn thuộc tính, ràng buộc
 - Bảo toàn dữ liệu

45



Phép tách không mất mát thông tin (*Lossless join*)

- Đ/N: Cho lược đồ quan hệ $R(U)$ phép tách R thành các sơ đồ con $\{R_1, R_2, \dots, R_k\}$ được gọi là **phép tách không mất mát thông tin** đ/v một tập phụ thuộc hàm F nếu với mọi quan hệ r xác định trên R thỏa mãn F thì:

$$r = \Pi_{R_1}(r) \bowtie \Pi_{R_2}(r) \bowtie \dots \bowtie \Pi_{R_k}(r)$$

- Ví dụ:
Supplier(sid, sname, city, NOE,
pname, colour, quantity)
 $\Rightarrow S1(\text{sid, sname, city, NOE})$
SP1(sid, pname, colour, quantity)

46



Kiểm tra tính không mất mát thông tin

- **Vào:** $R(A_1, A_2, \dots, A_n)$, F , phép tách $\{R_1, R_2, \dots, R_k\}$
- **Ra:** phép tách là mất mát thông tin hay không
- **Thuật toán**
 - B.1.** Thiết lập một bảng k hàng, n cột
Nếu A_j là thuộc tính của R_i thì điền a_j vào ô (i,j) .
Nếu không thì điền b_{ij} .
 - B.i.** Xét $f = X \rightarrow Y \in F$.
Nếu \exists 2 hàng t_1, t_2 thuộc bảng : $t_1[X] = t_2[X]$
thì $t_1[Y] = t_2[Y]$, ưu tiên đồng nhất về giá trị a
Lặp cho tới khi không thể thay đổi được giá trị nào trong bảng
 - B.n.** Nếu bảng có 1 hàng gồm các kí hiệu a_1, a_2, \dots, a_n
thì phép tách là không mất mát thông tin.
ngược lại, phép tách không bảo toàn thông tin.

47



Phép tách bảo toàn tập phụ thuộc hàm

- **Hình chiếu của tập phụ thuộc hàm**
Cho sơ đồ quan hệ R , tập phụ thuộc hàm F , phép tách $\{R_1, R_2, \dots, R_k\}$ của R trên F .
Hình chiếu F_i của F trên R_i là tập tất cả $X \rightarrow Y \in F+$:
 $XY \subseteq R_i$.
- Phép tách sơ đồ quan hệ R thành $\{R_1, R_2, \dots, R_k\}$ là một **phép tách bảo toàn tập phụ thuộc hàm** F nếu
$$(F_1 \cup F_2 \dots \cup F_k)+ = F+$$

hay hợp của tất cả các phụ thuộc hàm trong các hình chiếu của F lên các sơ đồ con sẽ suy diễn ra các phụ thuộc hàm trong F .

48



Bài tập

- Kiểm tra xem 1 phép tách có bảo toàn tập phụ thuộc hàm không
- Kiểm tra xem 1 phép tách có mất mát thông tin không

49



Các dạng chuẩn


- Vấn đề đặt ra
 - Có cần phải tinh chỉnh thiết kế nữa hay không?
 - Thiết kế đã là tốt hay chưa?
 - Định nghĩa về các dạng chuẩn.
- Mục đích:
Mỗi dạng chuẩn đảm bảo ngăn ngừa (giảm thiểu) một số các dạng dư thừa hay dị thường dữ liệu
- Các dạng chuẩn hay sử dụng
 - Dạng chuẩn 1 (1NF)
 - Dạng chuẩn 2 (2NF)
 - Dạng chuẩn 3 (3NF)
 - Dạng chuẩn Boye-Code (BCNF)
 - Dạng chuẩn 4 (4NF)

50

Dạng chuẩn 1 (1NF)

- Đ/N: Một sơ đồ quan hệ R được gọi là ở dạng chuẩn 1 nếu *tất cả các miền giá trị của các thuộc tính trong R đều chỉ chứa giá trị nguyên tố*.
 - Giá trị nguyên tố là giá trị mà không thể chia nhỏ ra được nữa
- Ví dụ: Quan hệ không ở 1NF và quan hệ sau khi chuẩn hóa về 1NF

sname	city	product	
		name	price
Blake	London	Nut	100
		Bolt	120
Smith	Paris	Screw	75



sname	city	item	price
Blake	London	Nut	100
Blake	London	Bolt	120
Smith	Paris	Screw	75

51

Dạng chuẩn 2 (2NF)

- Đ/N: Một sơ đồ quan hệ R được coi là ở dạng chuẩn 2 nếu
 - Sơ đồ quan hệ này ở 1NF
 - Tất cả các thuộc tính không khóa đều phụ thuộc hàm đầy đủ vào (khóa chính) mọi khoá tối thiểu**
 - Với 1 khoá tối thiểu bất kỳ, các thuộc tính không thuộc khoá này thì phải phụ thuộc đầy đủ vào khoá này

52



Phụ thuộc hàm đầy đủ

- Đ/N: Cho lược đồ quan hệ $R(U)$, F là tập phụ thuộc hàm trên R . $X, Y \subseteq U$. Y được gọi là **phụ thuộc đầy đủ** vào X nếu:
 - $X \rightarrow Y$ thuộc F^+
 - $\nexists X' \subset X : X' \rightarrow Y \in F^+$
- Các phụ thuộc hàm không đầy đủ còn gọi là **phụ thuộc bộ phận**

53



Ví dụ

- Sales(sid, sname, city, item, price)
- $F = \{ \text{sid} \rightarrow (\text{sname}, \text{city}), (\text{sid}, \text{item}) \rightarrow \text{price} \}$
- Khóa chính (sid,item)
 - sname, city không phụ thuộc hàm đầy đủ vào khóa chính
 - ⇒ Sales không thuộc 2NF
 - ⇒ Chuẩn hoá
 - S(sid, sname, city)
 - Sales (sid, item, price)

54



Dạng chuẩn 3 (3NF)

- Đ/N: Một sơ đồ quan hệ R được coi là ở dạng chuẩn 3 nếu
 - Sơ đồ quan hệ này ở 2NF
 - Mọi thuộc tính không khóa đều không phụ thuộc bắc cầu vào (khóa chính) mọi khóa tối thiểu

55



Ví dụ

- S (sid, sname, city)
Sales(sid, item, price)
F = {sid → sname, city}
- S, Sales thuộc dạng chuẩn 3
ItemInfo(item, price, discount).
F = {item → price, price → discount}
 - thuộc tính không khóa discount phụ thuộc bắc cầu vào khóa chính item.
 - Vậy quan hệ này không ở 3NF.
 - Chuẩn hoá
ItemInfo(item, price)
Discount(price, discount)


56



Dạng chuẩn Boye-Codd

- Đ/N: Một sơ đồ quan hệ $R(U)$ với một tập phụ thuộc hàm F được gọi là ở dạng chuẩn Boye-Codd (BCNF) nếu với $\forall X \rightarrow A \in F^+$ thì
 - A là thuộc tính xuất hiện trong X hoặc
 - X chứa một khóa của quan hệ R .
- Ví dụ
 - $R = \{A, B, C\}$; $F = \{AB \rightarrow C, C \rightarrow B\}$.
 - R không phải ở BCNF vì $\exists C \rightarrow B$, C không phải là khóa
- Chú ý:
 - Một quan hệ thuộc 3NF thì chưa chắc đã thuộc BCNF. Nhưng một quan hệ thuộc BCNF thì thuộc 3NF

57



Tách bảo toàn tập phụ thuộc hàm về 3NF

- **Vào:** $R(U)$, F (giả thiết F là phủ tối thiểu)
- **Ra:** Phép tách bảo toàn tập phụ thuộc hàm về 3NF
- **Thuật toán**
 - B1.** Với các $A_i \in U$, $A_i \notin F$ thì loại A_i khỏi R và lập 1 quan hệ mới cho các A_i
 - B2.** Nếu $\exists f \in F$, f chứa tất cả các thuộc tính của R thì kết quả là R
 - B3.** Ngược lại, với mỗi $X \rightarrow A \in F$, xác định một quan hệ $R_i(XA)$.
Nếu $\exists X \rightarrow A_i, X \rightarrow A_j$ thì tạo một quan hệ chung $R'(XA_iA_j)$

58



Ví dụ

Cho $R = \{A, B, C, D, E, F, G\}$

$F = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G\}$

- Xác định phép tách bảo toàn tập phụ thuộc hàm về 3NF

B1. không lập được quan hệ nào mới.

B2. $\exists f \in F$: f chứa tất cả các thuộc tính của R

B3. $A \rightarrow B \quad \Rightarrow R1(AB)$

$ACD \rightarrow E \quad \Rightarrow R2(ACDE)$

$EF \rightarrow G \quad \Rightarrow R3(EFG)$

59



Tách không mất mát thông tin và bảo toàn tập phụ thuộc hàm về 3NF

- Yêu cầu:
 - Bảo toàn tập phụ thuộc hàm (như thuật toán trên)
 - Đảm bảo là có một lược đồ con chứa khóa của lược đồ được tách
- Các bước tiến hành
 - B1. Tìm một khóa tối thiểu của lược đồ quan hệ R đã cho
 - B2. Tách lược đồ quan hệ R theo phép tách bảo toàn tập phụ thuộc
 - B3. Nếu 1 trong các sơ đồ con có chứa khóa tối thiểu thì kết quả của B2 là kết quả cuối cùng.
Ngược lại, thêm vào kết quả đó một sơ đồ quan hệ được tạo bởi khóa tối thiểu tìm được ở 1.

60



Ví dụ

Cho $R(A,B,C,D,E,F,G)$.

$$F = \{A \rightarrow B, ACD \rightarrow E, EF \rightarrow G\}$$

- B1.** Khóa tối thiểu cần tìm là ACDF (xem slide 19)
- B2.** Phép tách bảo toàn tập phụ thuộc hàm R cho 3 sơ đồ con $R_1(AB), R_2(ACDE), R_3(EFG)$ (xem slide 40)
- B3.** Do khóa ACDF không nằm trong bất kỳ một sơ đồ con nào trong 3 sơ đồ con trên, ta lập một sơ đồ con mới

$$R_4(ACDF)$$

Kết quả cuối cùng ta có phép tách R thành 4 sơ đồ con $\{R_1, R_2, R_3, R_4\}$ là một phép tách không mất mát thông tin và bảo toàn tập phụ thuộc hàm

61



Tách không mất mát thông tin về BCNF

- **Vào:** Sơ đồ quan hệ R, tập phụ thuộc hàm F.
- **Ra:** phép tách không mất mát thông tin bao gồm một tập các sơ đồ con ở BCNF với các phụ thuộc hàm là hình chiếu của F lên sơ đồ đó.
- Cách tiến hành
 - B1.** $KQ = \{R\}$,
 - B2.** Với mỗi $S \in KQ$, S không ở BCNF, xét $X \rightarrow A \in S$, với điều kiện X không chứa khóa của S và $A \notin X$. Thay thế S bởi S1, S2 với $S1 = A \cup \{X\}$, $S2 = \{S\} \setminus A$.
 - B3.** Lặp (B2) cho đến khi $\forall S \in KQ$ đều ở BCNF
KQ gồm các sơ đồ con của phép tách yêu cầu


62



Phụ thuộc đa trị

- Đ/N: Cho $R(U)$, $X, Y \in U$. X xác định đa trị Y hay Y phụ thuộc đa trị vào X nếu với $\forall r$ xác định trên R và với hai bộ t_1 và t_2 bất kỳ mà $t_1[X] = t_2[X]$ thì \exists bộ t_3 :
 $t_3[X] = t_1[X]$, $t_3[Y] = t_1[Y]$ và $t_3[Z] = t_2[Z]$ với $Z = U \setminus XY$.
 - Ký hiệu $X \twoheadrightarrow Y$

63




Hệ tiên đề đối với các phụ thuộc hàm và phụ thuộc đa trị

Cho $R(U)$, $X, Y, Z, W \subseteq U$ ($XY = X \cup Y$)

- A1: Phản xạ đối với FD (*reflexivity*):
Nếu $Y \subseteq X$ thì $X \rightarrow Y$.
- A2: Tăng trưởng đối với FD (*augmentation*):
Nếu $X \rightarrow Y$ thì $XZ \rightarrow YZ$.
- A3: Bắt cầu đối với FD (*transitivity*):
Nếu $X \rightarrow Y$, $Y \rightarrow Z$ thì $X \rightarrow Z$.
- A4: Luật bù đối với MVD (*complementation*):
Nếu $X \twoheadrightarrow Y$ thì $X \twoheadrightarrow U \setminus XY$.

64



Hệ tiên đề đối với các phụ thuộc hàm và phụ thuộc đa trị (2)

Cho $R(U)$, $X, Y, Z, W \subseteq U$ ($XY = X \cup Y$)

- A5: **Tăng trưởng đối với MVD** (*augmentation*):
Nếu $X \twoheadrightarrow Y$ và $V \subseteq W$ thì $WX \twoheadrightarrow VY$.
- A6: **Bắc cầu đối với MVD** (*transitivity*):
Nếu $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$ thì $X \twoheadrightarrow Z \setminus Y$.
- A7:
Nếu $X \rightarrow Y$ thì $X \twoheadrightarrow Y$.
- A8:
Nếu $X \twoheadrightarrow Y$, $W \rightarrow Z$ với $Z \subseteq Y$ và $W \cap Y = \emptyset$ thì $X \rightarrow Z$.

65



Các luật suy diễn bổ sung đối với các phụ thuộc đa trị

- **Luật hợp** (*union*):
Nếu $X \twoheadrightarrow Y$, $X \twoheadrightarrow Z$ thì $X \twoheadrightarrow YZ$.
- **Luật tựa bắc cầu** (*pseudotransitivity*):
Nếu $X \twoheadrightarrow Y$, $WY \twoheadrightarrow Z$ thì $WX \twoheadrightarrow Z \setminus WY$.
- **Luật tựa bắc cầu hỗn hợp** (*mixed pseudotransitivity*):
Nếu $X \twoheadrightarrow Y$, $XY \rightarrow Z$ thì $X \rightarrow Z \setminus Y$.
- **Luật tách** (*decomposition*):
Nếu $X \twoheadrightarrow Y$, $X \twoheadrightarrow Z$ thì
 $X \twoheadrightarrow Y \cap Z$, $X \twoheadrightarrow Y \setminus Z$, $X \twoheadrightarrow Z \setminus Y$.

66



Bao đóng của tập phụ thuộc hàm và phụ thuộc đa trị

- Đ/N: bao đóng của tập các phụ thuộc hàm và phụ thuộc đa trị D là tập tất cả các phụ thuộc hàm và các phụ thuộc đa trị được suy diễn logic từ D
 - Ký hiệu: D^+

67



Tính cơ sở phụ thuộc

- **Vào:** Tập các phụ thuộc đa trị M trên tập thuộc tính U và tập thuộc tính $X \subseteq U$.
- **Ra:** Cơ sở phụ thuộc của X đối với M .
- Cách tiến hành:
 - B1.** Đặt T là tập các tập con Z của U : với $W \rightarrow \rightarrow Y \in M$ mà $W \subseteq X$ thì Z là $Y \setminus X$ hoặc $U \setminus XY$.
 - B2.** T được thiết lập cho tới khi là một tập các tập rời nhau, nếu có một cặp $Z1, Z2$ không tách rời nhau thì thay chúng bởi $Z1 \setminus Z2, Z2 \setminus Z1, Z1 \cap Z2$ với điều kiện không ghi nhận tập rỗng. Gọi S là tập thu được sau bước này.
 - B3.** Tìm các phụ thuộc có dạng $V \rightarrow \rightarrow W$ trong M và một tập Y trong S : $Y \cap W \neq \emptyset, Y \cap V = \emptyset$
Thay Y bằng $Y \cap W$ và $Y \setminus W$ cho đến khi không thay đổi S được nữa.
 - B4.** Tập S thu được sau bước này là cơ sở phụ thuộc của X .

68

Phép tách không mất thông tin

- **Vào:** $R(A_1, A_2, \dots, A_n)$, F , M , phép tách $\{R_1, R_2, \dots, R_k\}$
- **Ra:** phép tách là mất mát thông tin hay không
- **Thuật toán** (tổng quát hoá thuật toán trình bày ở slide 28)
 - B.1.** Thiết lập một bảng k hàng, n cột (xem B1. slide 28)
 - B.i.** Xét $f = X \rightarrow Y \in F$:
 - thực hiện đồng nhất bảng (xem B2. slide 28)
 - Xét $X \twoheadrightarrow Y$:
 - nếu $\exists 2$ hàng $t1, t2$ thuộc bảng : $t1[X] = t2[X]$
thì thêm vào bảng đó một hàng mới u
 $u[X] = t1[X], u[Y] = t1[Y],$
 $u[R \setminus XY] = t2[R \setminus XY]$*
 - Lặp cho tới khi không thể thay đổi được giá trị nào trong bảng
 - B.n.** Nếu bảng có 1 hàng gồm các kí hiệu a_1, a_2, \dots, a_n thì phép tách là không mất mát thông tin.
ngược lại, phép tách không bảo toàn thông tin.

Dạng chuẩn 4 (4NF)

- Đ/N: Một quan hệ R ở dạng chuẩn bốn nếu có một phụ thuộc đa trị $X \twoheadrightarrow Y$ với $Y \neq \emptyset$, $Y \not\subset X$ và $XY \subset R$ thì X chứa một khóa của R
- Chú ý: nếu R chỉ có các phụ thuộc hàm thì dạng chuẩn bốn chính là dạng chuẩn Boye-Codd và $X \twoheadrightarrow Y$ phải có nghĩa là $X \rightarrow Y$.



Kết luận

- Tầm quan trọng của thiết kế CSDL
 - ảnh hưởng đến chất lượng dữ liệu lưu trữ
 - Hiệu quả của việc khai thác dữ liệu
- Mục đích của thiết kế CSDL: tránh
 - Dư thừa dữ liệu
 - Dị thường dữ liệu khi thêm/xoá/sửa đổi
 - Hiệu quả trong tìm kiếm
- Đưa về các dạng chuẩn
 - 2NF: giảm bớt sự dư thừa để tránh các dị thường khi cập nhật
 - 3NF: tránh các dị thường khi thêm/xoá

71



72