# Bài 1:

Cho lược đồ quan hệ r(student, name, birthday, age, advisor, department, semester, course, grade) và tập các phụ thuộc hàm

F = { student→name, birthday, age, advisor, department; birthday → age; advisor → department }

* Tìm một khóa của r dựa vào F
* Tìm một phân rã của r bảo toàn thông tin đối với F

# Bài 2:

Cho lược đồ quan hệ r=ABCDEF và tập các phụ thuộc hàm F = {AB→C, C→B, ABD→E, F→A}. Xét phép tách r thành các lược đồ con sau: r1=BC, r2=AC, r3=ABDE, r4=ABDF. Phép tách này có bảo toàn thông tin không?

# Bài 3:

Cho CSDL gồm các quan hệ:

Kháchhàng(**MãKH**, Họtên, Sốđiệnthoại, Cơquan)

Nhàchothuê(**MãN**, Địachỉ, Giáthuê, Tênchủnhà)

Hợpđồng(***MãN, MãKH***, Ngàybắtđầu, Ngàykếtthúc)

1. Dùng các câu lệnh SQL tạo các bảng trên.
2. Biểu diễn các yêu cầu sau bằng SQL và đại số quan hệ (nếu có thể)

* Đưa ra danh sách (Địachỉ, Tênchủnhà) của những ngôi nhà có giá thuê ít hơn 10 triệu.
* Đưa ra danh sách (MãKH, Họtên, Cơquan) của những người đã từng thuê nhà của chủ nhà có tên là "Nông Văn Dền"
* Đưa ra danh sách các ngôi nhà chưa từng được ai thuê
* Đưa ra giá thuê cao nhất trong số các giá thuê của các ngôi nhà đã từng ít nhất một lần được thuê.

**Bài chữa:**

# Bài 1:

Mã hóa thông tin như sau:

student(A), name(B), birthday(C), age(D), advisor(E), department(F), semester(G), course(H), grade(J)

U=ABCDEFGHJ

Tập phụ thuộc hàm F trở thành: F = {A→BCDEF, C→D, E→F}

## a) Tìm một khóa tối thiểu của r dựa vào F

+) Đặt K0=ABCDEFGHJ

+) Thử loại bỏ thuộc tính A

(BCDEFGHJ)+=BCDEFGHJ ≠ U ⇒ không bỏ được A. Vậy K1=K0=ABCDEFGHJ

+) Thử loại bỏ thuộc tính B

(ACDEFGHJ)+=ABCDEFGHJ = U ⇒ bỏ được B. Vậy K2=K1/{B}=ACDEFGHJ

+) Thử loại bỏ thuộc tính C

(ADEFGHJ)+=ABCDEFGHJ = U ⇒ bỏ được C. Vậy K3=K2/{C}=ADEFGHJ

+) Thử loại bỏ thuộc tính D

(AEFGHJ)+=ABCDEFGHJ = U ⇒ bỏ được D. Vậy K4=K3/{D}=AEFGHJ

+) Thử loại bỏ thuộc tính E

(AFGHJ)+=ABCDEFGHJ = U ⇒ bỏ được E. Vậy K5=K4/{E}=AFGHJ

+) Thử loại bỏ thuộc tính F

(AGHJ)+=ABCDEFGHJ = U ⇒ bỏ được F. Vậy K6=K5/{F}=AGHJ

+) Thử loại bỏ thuộc tính G

(AHJ)+=ABCDEFHJ ≠ U ⇒ không bỏ được G. Vậy K7=K6=AGHJ

+) Thử loại bỏ thuộc tính H

(AGJ)+=ABCDEFGJ ≠ U ⇒ không bỏ được G. Vậy K8=K7=AGHJ

+) Thử loại bỏ thuộc tính J

(AGH)+=ABCDEFGH ≠ U ⇒ không bỏ được J. Vậy K9=K8=AGHJ

Kết luận: một khóa tối thiểu là: **K = AGHJ,** tức là **K={student, semester, course, grade}**

## b) Tìm một phân rã của r bảo toàn thông tin đối với F

F = {A→BCDEF, C→D, E→F}

**Bước 1:** F1={A→B, A→C, A→D, A→E, A→F, C→D, E→F}

**Bước 2:** Không có phụ thuộc hàm nào có vế trái nhiều hơn một thuộc tính nên

F2= F1={A→B, A→C, A→D, A→E, A→F, C→D, E→F}

**Bước 3:** F30= F2= {A→B, A→C, A→D, A→E, A→F, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm A→B khỏi F30

(A)+=ACDEF không chứa B nên không loại được phụ thuộc hàm này.

F31= F30= {A→B, A→C, A→D, A→E, A→F, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm A→C

(A)+=ABDEF không chứa C nên không loại được phụ thuộc hàm này.

F32= F31= {A→B, A→C, A→D, A→E, A→F, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm A→D

(A)+=ABCDEF chứa D nên loại được phụ thuộc hàm này.

F33= F32 / { A→D } = {A→B, A→C, A→E, A→F, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm A→E

(A)+=ABCDF không chứa E nên không loại được phụ thuộc hàm này.

F34= F33 = {A→B, A→C, A→E, A→F, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm A→F

(A)+=ABCDEF chứa F nên loại được phụ thuộc hàm này.

F35= F34 / { A→F} = {A→B, A→C, A→E, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm C→D

(C)+= C không chứa D nên không loại được phụ thuộc hàm này.

F36= F35 = {A→B, A→C, A→E, C→D, E→F}

+) Thử loại bỏ phụ thuộc hàm E→F

(E)+= E không chứa F nên không loại được phụ thuộc hàm này.

F37= F36 = {A→B, A→C, A→E, C→D, E→F}

Vậy F3= F37 = {A→B, A→C, A→E, C→D, E→F}

Phủ tối thiểu FC = F3 = {A→B, A→C, A→E, C→D, E→F}

**Phân rã:**

Nhận thấy G, H, J không xuất hiện trong FC nên tạo 3 quan hệ: r1(G), r2(H), r3(J)

Từ FC, tạo các quan hệ: r4(ABCE), r5(CD), r6(EF)

***Trả lại tên cho em:*** r1(semester), r2(course), r3(grade), r4(student, name, birthday, advisor), r5(birthday, age), r6(advisor, department)

# Bài 2:

Cho lược đồ quan hệ r=ABCDEF và tập các phụ thuộc hàm F = {AB→C, C→B, ABD→E, F→A}. Xét phép tách r thành các lược đồ con sau: r1=BC, r2=AC, r3=ABDE, r4=ABDF. Phép tách này có bảo toàn thông tin không?

Lập bảng như sau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| r1 | b11 | a2 | a3 | b14 | b15 | b16 |
| r2 | a1 | b22 | a3 | b24 | b25 | b26 |
| r3 | a1 | a2 | b33 | a4 | a5 | b36 |
| r4 | a1 | a2 | b43 | a4 | b45 | a6 |

Xét phụ thuộc hàm C→B, ta có:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| r1 | b11 | a2 | a3 | b14 | b15 | b16 |
| r2 | a1 | a2 | a3 | b24 | b25 | b26 |
| r3 | a1 | a2 | b33 | a4 | a5 | b36 |
| r4 | a1 | a2 | b43 | a4 | b45 | a6 |

Xét phụ thuộc hàm ABD→E, ta có:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| r1 | b11 | a2 | a3 | b14 | b15 | b16 |
| r2 | a1 | a2 | a3 | b24 | b25 | b26 |
| r3 | a1 | a2 | b33 | a4 | a5 | b36 |
| r4 | a1 | a2 | b43 | a4 | a5 | a6 |

Xét phụ thuộc hàm AB→C, ta có:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| r1 | b11 | a2 | a3 | b14 | b15 | b16 |
| r2 | a1 | a2 | a3 | b24 | b25 | b26 |
| r3 | a1 | a2 | a3 | a4 | a5 | b36 |
| r4 | a1 | a2 | a3 | a4 | a5 | a6 |

Nhận thấy dòng r4 có toàn các giá trị ai nên phép tách không mất mát thông tin.

# Bài 3:

## a) Dùng các câu lệnh SQL tạo bảng:

CREATE TABLE Kháchhàng(

MãKH char(8) primary key NOT NULL,

Họtên char(30) NOT NULL,

Sốđiệnthoại char(11),

Cơquan char(30)

);

CREATE TABLE Nhàchothuê(

MãN char(6) primary key NOT NULL,

Địachỉ varchar(70),

Giáthuê int,

Tênchủnhà char(30)

);

CREATE TABLE Hợpđồng(

MãN char(6) ,

MãKH char(8),

Ngàybắtđầu date,

Ngàykếtthúc date,

primary key(MãN, MãKH),

foreign key (MãN) references Nhàchothuê(MãN),

foreign key (MãKH) references Kháchhàng(MãKH)

);

## b) Biểu diễn SQL và đại số quan hệ (nếu có thể)

### b.1) Đưa ra danh sách (Địachỉ, Tênchủnhà) của những ngôi nhà có giá thuê ít hơn 10 triệu

SELECT Địachỉ, Tênchủnhà

FROM Nhàchothuê

WHERE Giáthuê < 10;

∏ Địachỉ, Tênchủnhà σ Giáthuê < 10(Nhàchothuê)

### b.2) Đưa ra danh sách (MãKH, Họtên, Cơquan) của những người đã từng thuê nhà của chủ nhà có tên là "Nông Văn Dền"

SELECT MãKH, Họtên, Cơquan

FROM Kháchhàng, Nhàchothuê, Hợpđồng

WHERE Kháchhàng.MãKH = Hợpđồng.MãKH AND Hợpđồng.MãN = Nhàchothuê.MãN AND Tênchủnhà = 'Nông Văn Dền';

∏MãKH, Họtên, Cơquan σTênchủnhà = 'Nông Văn Dền' (Kháchhàng\*Hợpđồng\*Nhàchothuê)

### b.3) Đưa ra danh sách các ngôi nhà chưa từng được ai thuê

SELECT \*

FROM Nhàchothuê

WHERE MãN NOT IN(SELECT MãN FROM Hợpđồng);

Nhàchothuê \*(∏MãN(Nhàchothuê)-∏MãN(Hợpđồng))

### b.4) Đưa ra giá thuê cao nhất trong số các giá thuê của các ngôi nhà đã từng ít nhất một lần được thuê.

SELECT MAX(Giáthuê)

FROM Nhàchothuê

WHERE MãN IN (SELECT MãN FROM Hợpđồng);