

# CHƯƠNG 5

## CHUẨN HÓA CÁC BẢNG DỮ LIỆU

# Mục tiêu

- Giải thích quá trình chuẩn hóa và vai trò của nó trong quá trình thiết kế cơ sở dữ liệu.
- Xác định và mô tả từng dạng chuẩn: **1NF, 2NF, 3NF, BCNF** và **4NF**.
- Giải thích cách chuyển đổi dạng chuẩn từ thấp đến cao
- Áp dụng quy tắc chuẩn hóa để đánh giá và sửa đổi cấu trúc bảng.
- Xác định các tình huống yêu cầu phi chuẩn để sinh thông tin một cách hiệu quả.



# Đặt vấn đề

- Thiết kế cơ sở dữ liệu tốt phải có cấu trúc bảng tốt
  - Một bảng tốt:
    - Tránh những dư thừa dữ liệu.
    - Tránh những lỗi khi thao tác dữ liệu: chèn, sửa, xóa
- Quá trình tạo ra bảng tốt gọi là quá trình *chuẩn hóa*
- Thế nào là một bảng chuẩn?
- Có những dạng chuẩn nào?
- Liệu có thể sửa các bảng chưa chuẩn thành bảng tốt không? Có thể dựa trên các tiêu chuẩn về bảng về bảng để có định hướng thiết kế tốt hơn.

# Các bất cập khi cập nhật dữ liệu

- Cho một quan hệ *Thông tin cung cấp* (Sname, Add, Pro, Price) gồm tên nhà cung cấp, địa chỉ, mặt hàng và giá cả:
  - Dư thừa dữ liệu: Mỗi khi xuất hiện tên nhà cung cấp thì địa chỉ của ông ta lại lặp lại trong quan hệ.
  - Không nhất quán: Sửa đổi địa chỉ của nhà cung cấp ở một bộ nào đó còn bộ khác vẫn giữ nguyên.
  - Dị thường khi thêm bộ: Một nhà cung cấp chưa cung cấp một mặt hàng nào, khi đó không thể đưa địa chỉ.
  - Dị thường khi xóa bộ: Không thể xóa tất cả các mặt hàng được cung ứng bởi một nhà cung cấp vì một mặt hàng đó có thể được nhiều người cùng cung ứng.

# Sự cần thiết chuẩn hóa dữ liệu

- Mục đích của chuẩn hóa CSDL là nhóm các thuộc tính vào quan hệ nhằm giảm thiểu dư thừa dữ liệu, loại bỏ các bất thường khi cập nhật CSDL.
- > Cần có các bước chuẩn hóa dữ liệu từ một CSDL chưa chuẩn hóa sang chuẩn hóa.
- Dạng chưa chuẩn hóa (unnormalized form – UNF): quan hệ chưa chuẩn hóa là quan hệ chứa các bộ dữ liệu lặp lại giá trị.

# Các bất cập khi cập nhật dữ liệu

- Tách thành 2 quan hệ:
  - S(Sname, Add)
  - SP(Sname, Pro, Price)

# Chuẩn hóa CSDL

- Chuẩn hóa là quá trình phân rã lược đồ quan hệ dựa trên một tập phụ thuộc hàm và khóa chính nhằm đảm bảo các lược đồ quan hệ thỏa mãn 2 tính chất:
  - Trùng lặp dữ liệu ít nhất
  - Khả năng gây ra bất thường khi cập nhật được giảm thiểu

# Quá trình chuẩn hóa

- Mỗi bảng đại diện cho một chủ đề duy nhất.
  - Bảng SINHVIEN, bảng MONHOC, bảng GIANGVIEN,...
- Giao nhau giữa dòng và cột chỉ cần 1 giá trị, không gồm nhóm giá trị.
- Không có mục dữ liệu nào được lưu trữ một cách không cần thiết trong nhiều bảng
- Tất cả các thuộc tính *không khóa* phụ thuộc vào khóa chính.
- Mỗi bảng tránh những bất thường: *chèn, sửa, xóa*



# Quá trình chuẩn hóa

- Mục tiêu của chuẩn hóa là đảm bảo rằng tất cả *các bảng ít nhất đạt 3NF.*
- Các chuẩn cao hơn ít cần thiết Trong các bài toán
- Chuẩn hóa làm việc với *từng bảng.*
- Chia bảng thành các bảng mới dựa trên *phụ thuộc hàm* đã có.

# Các dạng chuẩn lược đồ quan hệ

- Dạng chuẩn 1
- Dạng chuẩn 2
- Dạng chuẩn 3
- Dạng chuẩn Boyce Codd
- Dạng chuẩn 4

# Nhắc lại các khái niệm về khóa

- Siêu khóa
- Khóa dự tuyển
- Khóa chính
- Khóa ngoại
- Thuộc tính khóa
- Thuộc tính không khóa
- Phụ thuộc hàm

# Một số dạng phụ thuộc hàm

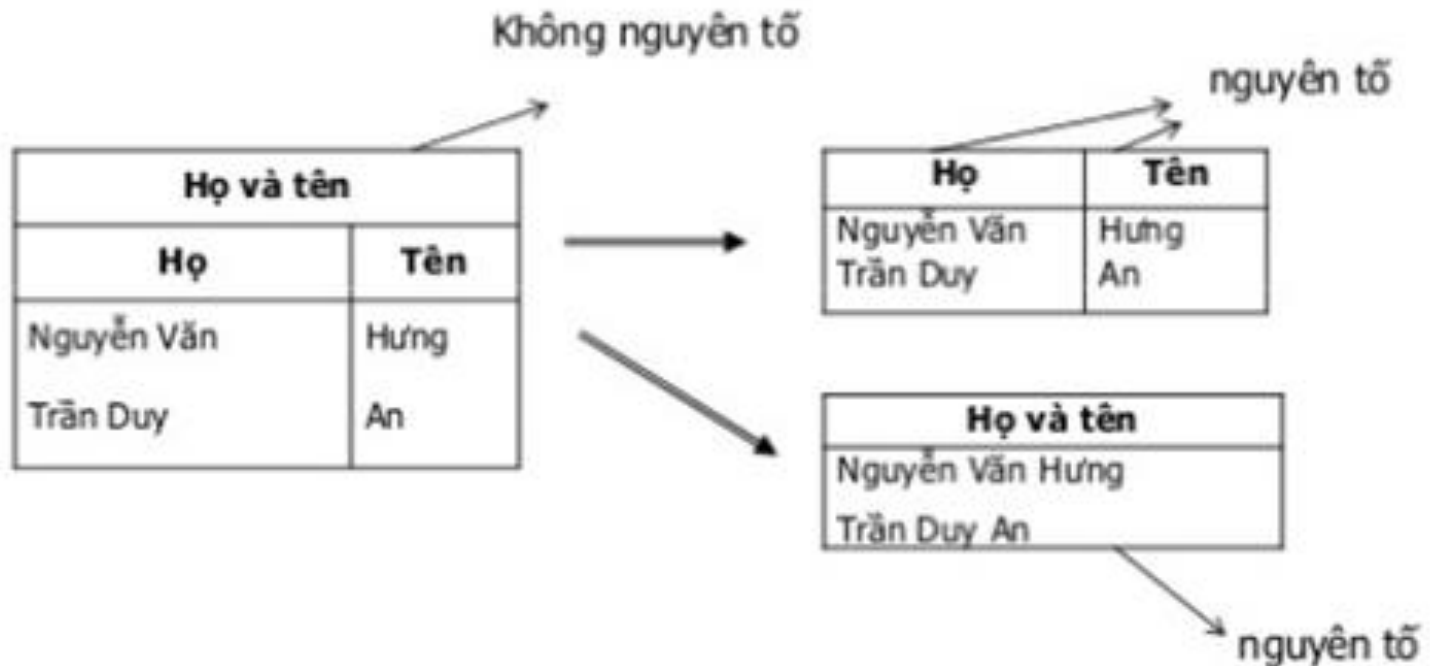
Khái niệm	Định nghĩa
Phụ thuộc hàm	Kí hiệu: $A \rightarrow B$ . Thuộc tính B phụ thuộc hàm vào thuộc tính A nếu với mỗi giá trị của thuộc tính A xác định duy nhất 1 giá trị của thuộc tính B.
Phụ thuộc hàm (định nghĩa khái quát)	Thuộc tính A xác định thuộc tính B (hay B phụ thuộc hàm vào A) nếu tất cả các hàng trong bảng giống nhau giá trị thuộc tính A thì cũng giống nhau với giá trị thuộc tính B.
Phụ thuộc hàm đầy đủ (khóa tổng hợp)	Nếu thuộc tính B phụ thuộc hàm vào khóa tổng hợp A, nhưng không phụ thuộc vào bất cứ tập con nào của khóa tổng hợp thì B được gọi là phụ thuộc hàm đầy đủ vào A.

# Dạng chuẩn 1 – 1NF

- Khái niệm: Lược đồ quan hệ R được gọi là 1NF nếu và chỉ nếu tất cả các thuộc tính của R thỏa mãn cả 3 điều kiện sau:
- Là nguyên tử (thuộc tính đơn – giá trị ko chia nhỏ được nữa)
  - Giá trị của các thuộc tính trên các bộ là đơn trị, không chứa nhóm lặp.
  - Không có một số thuộc tính nào có giá trị mà có thể tính toán được từ một số thuộc tính khác.

# Dạng chuẩn 1 – 1NF

- Ví dụ: Quan hệ sau được chuẩn hóa về dạng chuẩn 1



# Đưa về dạng chuẩn 1

- Nguyên tắc chung: Loại bỏ thuộc tính lặp hoặc đa trị
- Các bước thực hiện:
  - Tách nhóm thuộc tính lặp/ đa trị sang một bảng mới.
  - Khóa của bảng mới là Khóa của bảng ban đầu và Khóa nhóm lặp.
  - Bảng còn lại là bảng gồm có khóa và các thuộc tính còn lại
- Ví dụ: Cho quan hệ  $R(\underline{A1}, A2, A3, \{A4, A5, A6\})$ 
  - Với A1 là Khóa chính của quan hệ,  $\{A4, A5, A6\}$  là thuộc tính lặp.
  - Quan hệ R sẽ được tách thành 2 quan hệ R1, R2.



# Đưa về dạng chuẩn 1

➤ Ví dụ1: Cho bảng ĐƠN VỊ

MasoDV	TenDV	MasoNQL	DiaDiem
5	Nghiên cứu	NV002	Nam Định, Hà Nội, Bắc Ninh
4	Hành chính	NV014	Hà Nội
1	Lãnh đạo	NV061	Hà Nội

- Ta có quan hệ sau DONVIE(MasoDV, TenDV, MasoNQL, {DiaDiem})
- Thuộc tính đa trị là {DiaDiem} quan hệ này được tách thành
  - R1(MasoDV, DiaDiem)
  - R2(MasoDV, TenDV, MasoNQL)



# Đưa về dạng chuẩn 1

➤ Ví dụ 2: Cho bảng NHANVIEN\_DUAN

MasoDA	TenDA	TenNV	Sogio
1	DA01	Vân	15
		Nam	20
2	DA02	Nam	10
		Thanh	12
		Bằng	28
3	DA03	Thanh	20
		Vân	12

- Quan hệ này là  $R(\text{MasoDA}, \text{TenDA}, \{\text{TenNV}, \text{Sogio}\})$
- Thuộc tính lặp là  $\{\text{TenNV}, \text{Sogio}\}$  quan hệ được tách thành:
  - $R1(\underline{\text{MasoDA}}, \underline{\text{TenNV}}, \text{Sogio})$
  - $R2(\underline{\text{MasoDA}}, \text{TenDA})$

# Ví dụ chuyển đổi thành 1NF

FIGURE 6.2 A TABLE IN FIRST NORMAL FORM

Table name: DATA\_ORG\_1NF

Database name: Ch06\_ConstructCo

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
15	Evergreen	101	John G. News	Database Designer	105.00	19.4
15	Evergreen	105	Alice K. Johnson *	Database Designer	105.00	35.7
15	Evergreen	106	William Smithfield	Programmer	35.75	12.6
15	Evergreen	102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
18	Amber Wave	118	James J. Frommer	General Support	18.36	45.3
18	Amber Wave	104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
18	Amber Wave	112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
22	Rolling Tide	104	Anne K. Ramoras	Systems Analyst	96.75	48.4
22	Rolling Tide	113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
22	Rolling Tide	111	Geoff B. Wabash	Clerical Support	26.87	22.0
22	Rolling Tide	106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
25	Starflight	115	Travis B. Bawangi	Systems Analyst	96.75	45.8
25	Starflight	101	John G. News *	Database Designer	105.00	56.3
25	Starflight	114	Annelise Jones	Applications Designer	48.10	33.1
25	Starflight	108	Ralph B. Washington	Systems Analyst	96.75	23.6
25	Starflight	118	James J. Frommer	General Support	18.36	30.5
25	Starflight	112	Darlene M. Smithson	DSS Analyst	45.95	41.4

# Chuyển đổi thành 1NF

- Chuẩn hóa là quy trình 3 bước:
  - Bước 1: Xác định nhóm lặp:
    - Loại giá trị NULL: mỗi thuộc tính nhóm lặp chứa 1 giá trị dữ liệu thích hợp.
  - Bước 2: Xác định tất cả các yếu tố phụ thuộc  
Ví dụ:
    - PROJ\_NUM, EMP\_NUM → PROJ\_NAME, EMP\_NAME, JOB\_CLASS, CHG\_HOUR, HOURS
    - Nghĩa là, các giá trị PROJ\_NAME, EMP\_NAME, JOB\_CLASS, CHG\_HOUR và HOURS đều phụ thuộc vào PROJ\_NUM, EMP\_NUM
  - Bước 3: Xác định khóa chính
    - Phải xác định duy nhất giá trị của thuộc tính
    - Khóa mới phải được tạo. VD: với bảng trên là PROJ\_NUM và EMP\_NUM

# Chuyển đổi thành 1NF

## ➤ Sơ đồ phụ thuộc:

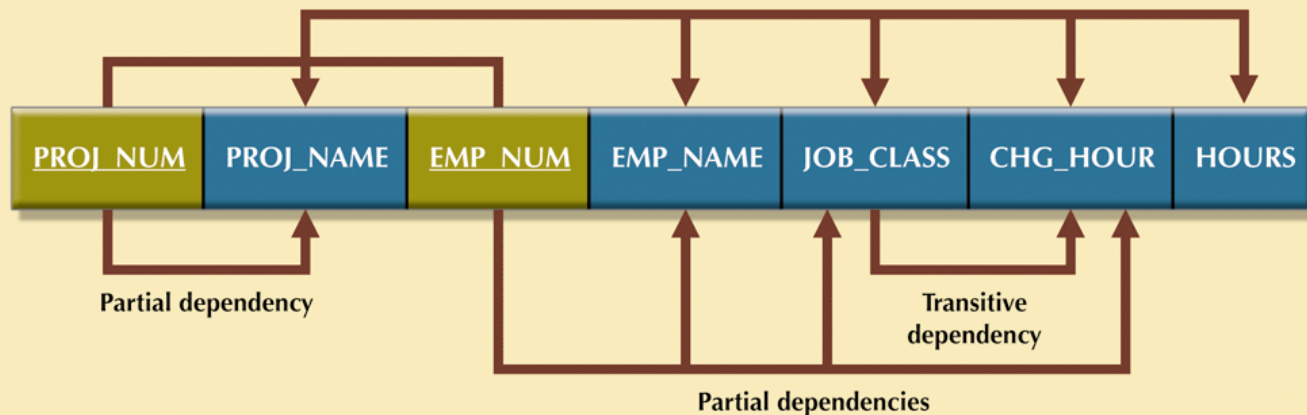
- Là mô tả tất cả các phụ thuộc được tìm thấy trong cấu trúc bảng nhất định.
- Hữu ích trong việc có được cái nhìn tổng thể về tất cả các mối quan hệ giữa các thuộc tính của bảng
- Làm cho nó có khả năng sẽ bỏ qua một phụ thuộc quan trọng.

# Ví dụ chuyển đổi thành 1NF

## ➤ Sơ đồ phụ thuộc:

FIGURE  
6.3

First normal form (1NF) dependency diagram



1NF (PROJ\_NUM, EMP\_NUM, PROJ\_NAME, EMP\_NAME, JOB\_CLASS, CHG\_HOURS, HOURS)

PARTIAL DEPENDENCIES:

(PROJ\_NUM  $\twoheadrightarrow$  PROJ\_NAME)

(EMP\_NUM  $\twoheadrightarrow$  EMP\_NAME, JOB\_CLASS, CHG\_HOUR)

TRANSITIVE DEPENDENCY:

(JOB\_CLASS  $\twoheadrightarrow$  CHG\_HOUR)

# Ví dụ chuyển đổi thành 1NF

- Sau khi chuyển sang 1NF được các bảng sau:
  - **ASSIGNMENT** (PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)
  - **PROJECT** (PROJ\_NUM, PROJ\_NAME, EMP\_NUM, EMP\_NAME, JOB\_CLASS, CHG\_HOUR)

# Ví dụ chuyển đổi thành 1NF

- Mô tả định dạng bảng đạt 1NF:
  - Tất cả các thuộc tính khóa được xác định
  - Không có nhóm lặp lại trong bảng
  - Tất cả các thuộc tính đều phụ thuộc vào khóa chính
- Nhận xét: Tất cả các bảng đạt 1NF:
  - Một số bảng chứa phụ thuộc hàm bộ phận
  - Sự phụ thuộc dựa trên 1 phần của khóa chính
  - Nên thận trọng khi sử dụng bảng đạt 1NF
- Kiểm tra các bất thường khi: chèn, sửa, xóa???

# Dạng chuẩn 2 – 2NF

- **Phụ thuộc hàm đầy đủ:** Y phụ thuộc hàm đầy đủ vào X nếu với bất kì A thuộc X thì  $X - \{A\} \nrightarrow Y$ . Ngược lại Y phụ thuộc hàm bộ phận vào X.
- **Ví dụ:** Lược đồ quan hệ
  - *Cộng tác viên-Chương trình (Mã số CTV, Mã CT, Họ tên CTV, Số giờ, Tên CT, Trường quay)* với các phụ thuộc hàm:
    - *Mã số CTV, Mã CT  $\rightarrow$  Số giờ* (1)
    - *Mã số CTV  $\rightarrow$  Họ tên CTV* (2)
    - *Mã CT  $\rightarrow$  Tên CT, Trường quay* (3)
  - (1) Phụ thuộc đầy đủ vào khóa; (2) và (3) Phụ thuộc bộ phận vào khóa;



# Dạng chuẩn 2 – 2NF

- Dạng chuẩn 2 dựa trên khái niệm phụ thuộc hàm đầy đủ.
- Một lược đồ quan hệ ở dạng chuẩn 2 nếu nó thỏa mãn dạng chuẩn 1 và mỗi thuộc tính không khóa phụ thuộc hàm đầy đủ vào khóa chính.

# Ví dụ 1

- Chẳng hạn xét lược đồ quan hệ
  - $Q(A, B, C, D)$  và
  - $F = \{ AB \rightarrow CD; B \rightarrow D; C \rightarrow A \}$
- Khoá là  $\{AB\}$  và  $\{BC\}$ .
- Do đó  $D$  là thuộc tính không khoá không phụ thuộc hàm đầy đủ vào khóa vì có  $B \rightarrow D$ .  $Q$  ko đạt 2NF.
- Vậy  $Q$  đạt chuẩn 1.

# Ví dụ 2

➤ Xác định dạng chuẩn của lược đồ quan hệ sau.

- $Q(\text{GMVNHP})$
- $F = \{G \rightarrow N; G \rightarrow H; G \rightarrow P; G \rightarrow V, G \rightarrow M, M \rightarrow V; NHP \rightarrow M\}$

## Ví dụ 2

- Xác định dạng chuẩn của lược đồ quan hệ sau.
  - $Q(GMVNHP)$
  - $F = \{G \rightarrow N; G \rightarrow H; G \rightarrow P; G \rightarrow V, G \rightarrow M, M \rightarrow V; NHP \rightarrow M\}$
  - Dễ thấy khoá của  $Q$  là  $G$ .
  - Thuộc tính không khoá là  $M, V, N, H, P$ .
- Do các phụ thuộc hàm  $G \rightarrow M; G \rightarrow V; G \rightarrow N; G \rightarrow H; G \rightarrow P$  là các phụ thuộc hàm có thuộc tính không khoá phụ thuộc đầy đủ vào khoá, nên lược đồ quan hệ  $Q$  đạt dạng chuẩn 2

# Hệ quả

- R đạt 2NF nếu R là 1NF và tập thuộc tính không khoá của R bằng rỗng.
- Nếu khoá của quan hệ có một thuộc tính thì quan hệ đó ít nhất đạt chuẩn 2.

# Ví dụ

➤  $R(ACDEH)$

–  $F = \{A \rightarrow E; C \rightarrow D; E \rightarrow DH\}$

# Ví dụ

## ➤ R(ACDEH)

- $F = \{A \rightarrow E; C \rightarrow D; E \rightarrow DH\}$
- Dễ thấy khoá của R là  $K = \{AC\}$
- D là thuộc tính không khoá và  $C \rightarrow D$ , vì C là tập con thực sự của khoá nên R không đạt dạng chuẩn 2

# Đưa về dạng chuẩn 2NF

## ➤ Quy tắc chuẩn hóa **2NF**:

- **Bước 1:** Loại bỏ các thuộc tính không khóa phụ thuộc vào một bộ phận khóa chính và tách thành ra một bảng riêng, khóa chính của bảng là bộ phận khóa mà chúng phụ thuộc vào.
- **Bước 2:** Các thuộc tính còn lại lập thành một quan hệ, khóa chính của nó là khóa chính ban đầu.



# Đưa về dạng chuẩn 2NF



- Khóa chính là bộ  $\{\underline{A_1}, \underline{A_2}\}$  (tức là từ  $\{\underline{A_1}, \underline{A_2}\} \rightarrow \{A_3, A_4, A_5, A_6\}$ ).
- Có phụ thuộc hàm phụ thuộc vào một phần của khóa  $\{\underline{A_2}\} \rightarrow \{A_5, A_6\}$ .
- Quan hệ được tách thành hai quan hệ sau:
  - $R_1(\underline{A_2}, A_5, A_6)$  : Tách các thuộc tính phụ thuộc vào 1 phần của khóa
  - $R_2(\underline{A_1}, \underline{A_2}, A_3, A_4)$  : Bảng ban đầu còn lại

# Đưa về dạng chuẩn 2NF

- Ví dụ: Cho quan hệ NHANVIEN\_DUAN(**MasoNV**, **MasoDA**, Sogio, HotenNV, TenDA, DiadiemDA) và các phụ thuộc hàm:
- {MasoNV, MasoDA} → {Sogio}
  - {MasoNV} → {HotenNV}
  - {MasoDA} → {TenDA, DiadiemDA}

# Đưa về dạng chuẩn 2NF

- Ví dụ: Cho quan hệ NHANVIEN\_DUAN(MasoNV, MasoDA, Sogio, HotenNV, TenDA, DiadiemDA) và các phụ thuộc hàm:

MasoNV	MasoDA	Sogio	HotenNV	TenDA	DiadiemDA
NV01	DA01	10	Nguyễn Lan	Dự án 1	Hà Nội
NV02	DA02	20	Lê Trung	Dự án 2	Hà Nam
NV01	DA02	15	Nguyễn Lan	Dự án 2	Hà Nam
NV02	DA01	30	Lê Trung	Dự án 1	Hà Nội
NV03	DA01	12	Đinh Trung	Dự án 1	Hà Nội
NV03	DA03	15	Đinh Trung	Dự án 3	Hải Phòng
NV02	DA03	26	Lê Trung	Dự án 3	Hải Phòng
NV04	DA02	16	Hoài Nam	Dự án 2	Hà Nam

# Đưa về dạng chuẩn 2NF

- Quan hệ này được tách thành các quan hệ sau:
  - R1(MasoNV, HotenNV)
  - R2(MasoNV, MasoDA, TenDA, DiadiemDA, Sogio)
  - Trong quan hệ R2 chứa phụ thuộc hàm {MasoDA → {TenDA, DiadiemDA} tách thành:
    - R21(MasoDA, TenDA, DiadiemDA)
    - R22(MasoNV, MasoDA, Sogio)

# Ví dụ chuyển đổi thành 2NF

FIGURE 6.2 A TABLE IN FIRST NORMAL FORM

Table name: DATA\_ORG\_1NF

Database name: Ch06\_ConstructCo

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
15	Evergreen	101	John G. News	Database Designer	105.00	19.4
15	Evergreen	105	Alice K. Johnson *	Database Designer	105.00	35.7
15	Evergreen	106	William Smithfield	Programmer	35.75	12.6
15	Evergreen	102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
18	Amber Wave	118	James J. Frommer	General Support	18.36	45.3
18	Amber Wave	104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
18	Amber Wave	112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
22	Rolling Tide	104	Anne K. Ramoras	Systems Analyst	96.75	48.4
22	Rolling Tide	113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
22	Rolling Tide	111	Geoff B. Wabash	Clerical Support	26.87	22.0
22	Rolling Tide	106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
25	Starflight	115	Travis B. Bawangi	Systems Analyst	96.75	45.8
25	Starflight	101	John G. News *	Database Designer	105.00	56.3
25	Starflight	114	Annelise Jones	Applications Designer	48.10	33.1
25	Starflight	108	Ralph B. Washington	Systems Analyst	96.75	23.6
25	Starflight	118	James J. Frommer	General Support	18.36	30.5
25	Starflight	112	Darlene M. Smithson	DSS Analyst	45.95	41.4

# Ví dụ chuyển đổi thành 2NF

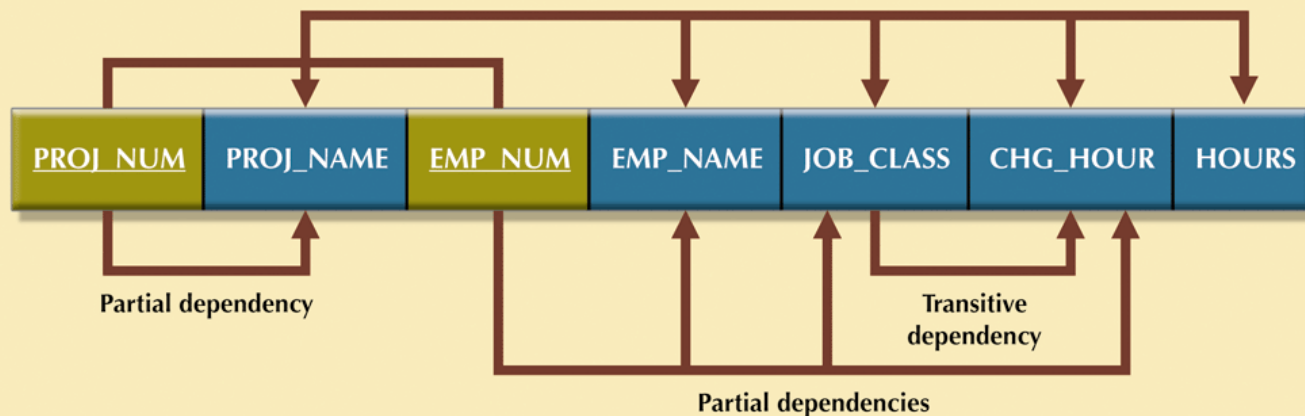
- Bước 1: Xác định tất cả các yếu tố phụ thuộc
- Bước 1: Xác định khóa
  - Viết mỗi khóa thành phần trên một dòng, khóa tổng hợp dòng cuối.
  - Mỗi thành phần này sẽ trở thành bảng mới.

# Ví dụ chuyển đổi thành 2NF

## ➤ Sơ đồ phụ thuộc:

FIGURE 6.3

First normal form (1NF) dependency diagram



1NF (PROJ\_NUM, EMP\_NUM, PROJ\_NAME, EMP\_NAME, JOB\_CLASS, CHG\_HOURS, HOURS)

PARTIAL DEPENDENCIES:

(PROJ\_NUM → PROJ\_NAME)

(EMP\_NUM → EMP\_NAME, JOB\_CLASS, CHG\_HOUR)

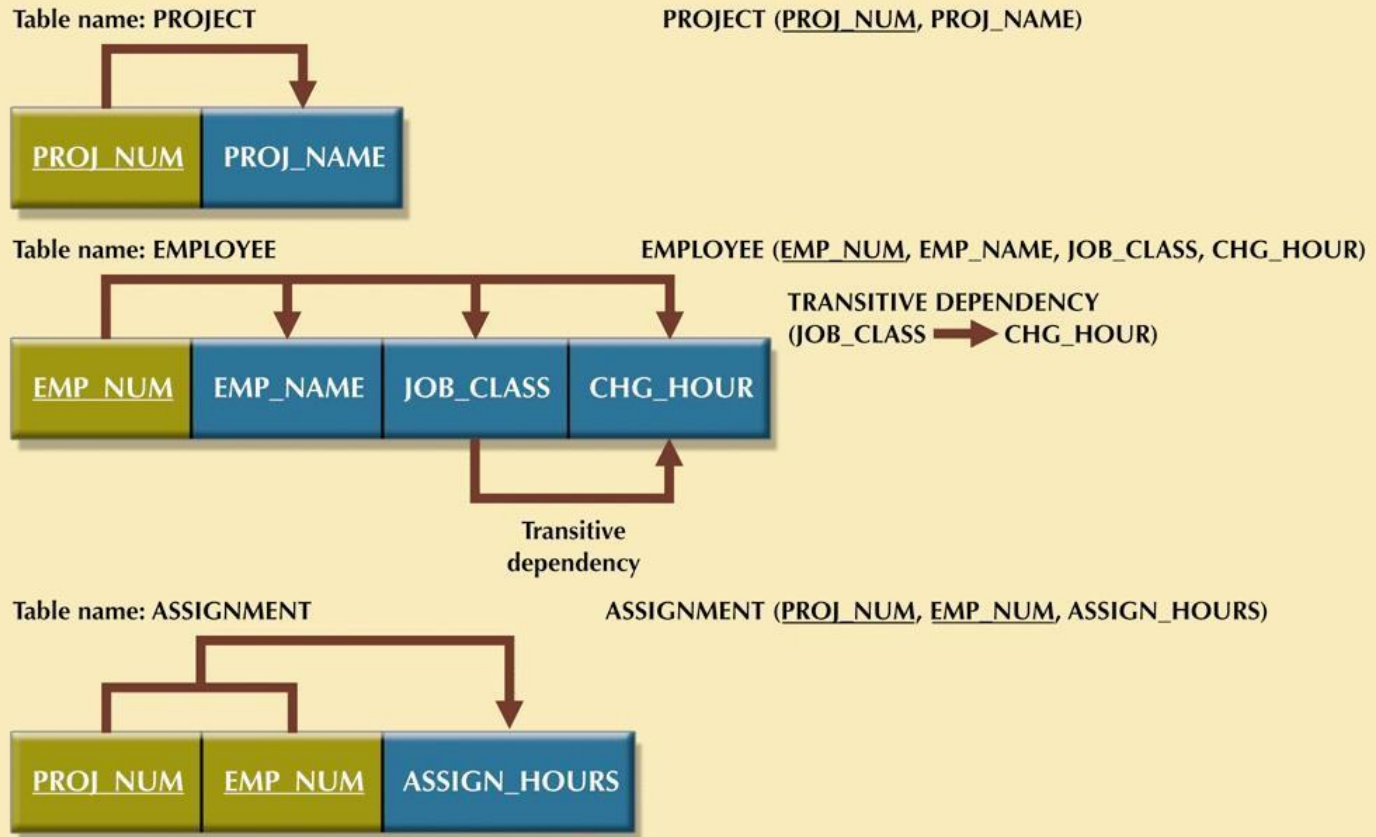
TRANSITIVE DEPENDENCY:

(JOB\_CLASS → CHG\_HOUR)

# Ví dụ chuyển đổi thành 2NF

Second normal form (2NF) conversion results

FIGURE 6.4





# Ví dụ chuyển đổi thành 2NF

- Sau khi chuyển sang 2NF được các bảng sau:
  - **PROJECT**(PROJ\_NUM, PROJ\_NAME)
  - **EMPLOYEE**(EMP\_NUM, EMP\_NAME, JOB\_CLASS, CHG\_HOUR)
  - **ASSIGNMENT**(PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)
- Một bảng đạt 2NF nếu thoả mãn 2 điều kiện:
  - Bảng đạt 1NF
  - Mỗi thuộc tính không khóa phụ thuộc đầy đủ vào khóa

# Dạng chuẩn 3 – 3NF

## ➤ Phụ thuộc hàm bắc cầu:

Một phụ thuộc hàm  $X \rightarrow Z$  là một phụ thuộc hàm bắc cầu nếu có một tập thuộc tính  $Y$  không phải là một khóa dự tuyển và cũng không phải là tập con của một khóa nào mà  $X \rightarrow Y, Y \rightarrow Z$ .

$Y$  được gọi là tập thuộc tính bắc cầu.

# Ví dụ

- Cho lược đồ quan hệ:  
*Hóa đơn bán hàng (Số hóa đơn, Mã người bán hàng, Tên người bán hàng, số tiền)*
- Các phụ thuộc hàm:
  - **Số hóa đơn** → Mã người bán hàng
  - **Mã người bán hàng** → Tên người bán hàng
- Tên người bán hàng phụ thuộc hàm bắc cầu vào Số hóa đơn.

# Dạng chuẩn 3 – 3NF

- Dạng chuẩn 3 dựa trên khái niệm phụ thuộc hàm bậc cầu.
- Một lược đồ quan hệ ở dạng chuẩn 3 nếu nó thỏa mãn dạng chuẩn 2 và không có thuộc tính không khóa phụ thuộc hàm bậc cầu vào khóa chính.
  - Lược đồ quan hệ *Hóa đơn bán hàng* đã nêu ở trên không đạt chuẩn 3NF.

# Dạng chuẩn 3 – 3NF

- *Định nghĩa hình thức dạng chuẩn 3:* Một lược đồ quan hệ R là ở dạng chuẩn 3 (3NF) nếu mọi phụ thuộc hàm  $X \rightarrow A$  thỏa mãn trong R thì:
- Hoặc X là một siêu khóa của R;
  - Hoặc A là một thuộc tính khóa của R.

# Dạng chuẩn 3NF

- Thuật toán kiểm tra lược đồ ở dạng chuẩn 3NF hay không?
  - **Input:** Lược đồ quan hệ  $R = (U, F)$
  - **Output:** Khẳng định  $R$  có đạt dạng chuẩn 3NF hay không
- Phương pháp:
  - B1: Tìm tất cả khóa của  $R$
  - B2: Từ  $F$  tạo tập phụ thuộc hàm tương đương  $G$  có vẻ phải một thuộc tính.
  - B3: Nếu mọi phụ thuộc hàm  $X \rightarrow A$  với  $A \notin X$  đều có hoặc  $X$  là siêu khóa hoặc  $A$  chứa thuộc tính khóa thì  $R$  đạt chuẩn 3NF, ngược lại  $R$  không đạt chuẩn 3NF.

# Dạng chuẩn 3 – 3NF

- Ví dụ :  $R(\underline{A}, B, C, D, E, F)$ 
  - Với  $F = \{A \rightarrow BCDE; BC \rightarrow ADEF; B \rightarrow F; D \rightarrow E\}$
- Do có phụ thuộc hàm  $D \rightarrow E$ , trong đó  $D$  không phải là siêu khóa,  $E$  cũng không phải là thuộc tính khóa, nên  $R$  vi phạm chuẩn 3NF.

# Dạng chuẩn 3 – 3NF

- Ví dụ :  $R(\underline{A}, B, C, D, E, F)$ 
  - Với  $F = \{A \rightarrow BCDE; BC \rightarrow ADEF; B \rightarrow F; D \rightarrow E\}$
- Do có phụ thuộc hàm  $D \rightarrow E$ , trong đó  $D$  không phải là siêu khóa,  $E$  cũng không phải là thuộc tính khóa, nên  $R$  vi phạm chuẩn 3NF.

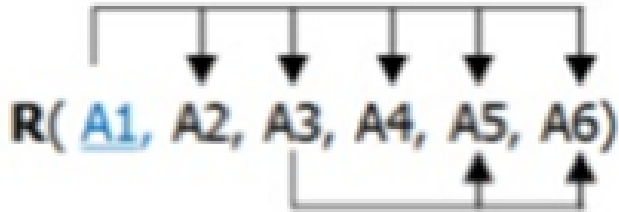


# Quy tắc chuẩn hóa về 3NF

## ➤ Quy tắc chuẩn hóa về 3NF:

- **Bước 1:** Loại bỏ các thuộc tính phụ thuộc bậc cầu ra khỏi quan hệ và tách chúng thành một quan hệ riêng có khóa chính là thuộc tính bậc cầu. (các thuộc tính ko khóa phải phụ thuộc trực tiếp vào khóa chính.)
- **Bước 2:** Các thuộc tính còn lại lập thành một quan hệ có khóa chính là quan hệ ban đầu.

# Quy tắc chuẩn hóa về 3NF



- Khóa chính là  $\{A_1\}$  (tức là từ  $\{A_1\} \rightarrow \{A_2, A_3, A_4, A_5, A_6\}$ )
- Có phụ thuộc hàm phụ thuộc bắc cầu  $\{A_3\} \rightarrow \{A_5, A_6\}$ .
- Quan hệ được tách thành hai quan hệ sau:
  - $R_1(A_3, A_5, A_6)$  - Tách thuộc tính phụ thuộc bắc.
  - $R_2(A_1, A_2, A_3, A_4)$  - Bảng ban đầu còn lại

# Ví dụ 1

- Chuyển quan hệ sau về dạng chuẩn 3NF
  - Quan hệ Order cho trong bảng dưới đây: OrderID là khóa (OrderID->OrderDate, CustomerID, CustomerName, CustomerCountry, ProductID)  
CustomerID-> CustomerName, CustomerCountry

Order					
<u>OrderID</u>	OrderDate	CustomerID	CustomerName	CustomerCountry	ProductID
1	09/30/2022	AA	Jack	UK	112
2	09/30/2022	BB	Mary	France	80
3	04/04/2021	AA	Jack	UK	67
4	07/21/2022	DD	David	USA	20

# Ví dụ 1

## ➤ Chuyển quan hệ sau về dạng chuẩn 3NF

Order					
<u>OrderID</u>	OrderDate	CustomerID	CustomerName	CustomerCountry	ProductID
1	09/03/2024	AA	Jack	UK	112
2	10/03/2024	BB	Mary	France	90
3	11/03/2024	AA	Jack	UK	68
4	12/03/2024	DD	David	USA	25

- Quan hệ Order cho trong bảng dưới đây: OrderID là khóa (
- Các phụ thuộc hàm:
  - OrderID->OrderDate, CustomerID, CustomerName, CustomerCountry, ProductID
  - CustomerID-> CustomerName, CustomerCountry

# Ví dụ chuyển đổi thành 3NF

FIGURE 6.2 A TABLE IN FIRST NORMAL FORM

Table name: DATA\_ORG\_1NF

Database name: Ch06\_ConstructCo

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
15	Evergreen	101	John G. News	Database Designer	105.00	19.4
15	Evergreen	105	Alice K. Johnson *	Database Designer	105.00	35.7
15	Evergreen	106	William Smithfield	Programmer	35.75	12.6
15	Evergreen	102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
18	Amber Wave	118	James J. Frommer	General Support	18.36	45.3
18	Amber Wave	104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
18	Amber Wave	112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
22	Rolling Tide	104	Anne K. Ramoras	Systems Analyst	96.75	48.4
22	Rolling Tide	113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
22	Rolling Tide	111	Geoff B. Wabash	Clerical Support	26.87	22.0
22	Rolling Tide	106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
25	Starflight	115	Travis B. Bawangi	Systems Analyst	96.75	45.8
25	Starflight	101	John G. News *	Database Designer	105.00	56.3
25	Starflight	114	Annelise Jones	Applications Designer	48.10	33.1
25	Starflight	108	Ralph B. Washington	Systems Analyst	96.75	23.6
25	Starflight	118	James J. Frommer	General Support	18.36	30.5
25	Starflight	112	Darlene M. Smithson	DSS Analyst	45.95	41.4

# Ví dụ chuyển đổi thành 3NF

FIGURE 6.5 THIRD NORMAL FORM (3NF) CONVERSION RESULTS



Table name: PROJECT

PROJECT (PROJ\_NUM, PROJ\_NAME)



Table name: JOB

JOB (JOB\_CLASS, CHG\_HOUR)

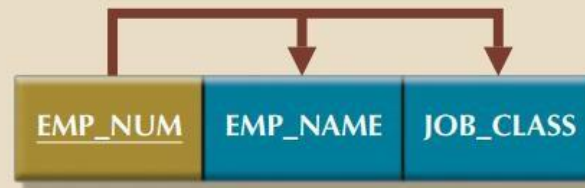


Table name: EMPLOYEE

EMPLOYEE (EMP\_NUM, EMP\_NAME, JOB\_CLASS)



Table name: ASSIGNMENT

ASSIGNMENT (PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)

# Ví dụ chuyển đổi thành 3NF

- Sau khi chuyển sang 3NF được các bảng sau:
  - **PROJECT**(PROJ\_NUM, PROJ\_NAME)
  - **EMPLOYEE**(EMP\_NUM, EMP\_NAME, JOB\_CLASS)
  - **JOB**(JOB\_CLASS, CHG\_HOUR)
  - **ASSIGNMENT**(PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)
- Một bảng đạt 3NF nếu thoả mãn 2 điều kiện:
  - Bảng đạt 2NF
  - Không có các PTH bắc cầu

# Cải thiện thiết kế

- Cấu trúc bảng cần được làm sạch để loại bỏ sự **PTH bộ phận** và **PTH bắc cầu**.
- Chuẩn hóa không thể tự nó có thể tạo ra thiết kế tốt, nhưng nó giúp loại bỏ dư thừa dữ liệu.
- Các vấn đề cần giải quyết để tạo ra tập các bảng tốt:
  - Đánh giá khóa chính - PK:
    - Với bảng JOB (JOB\_CLASS, CHG\_HOUR), nếu JOB\_CLASS lúc thì nhập “DB Designer”, lúc thì “Database Desgner” thì sao?
    - Tốt hơn hết nên tạo thêm JOB\_CODE để đảm bảo tính duy nhất
  - Đánh giá quy ước đặt tên:
    - Nên tuân thủ 1 quy tắc đặt tên
    - VD: CHG\_HOUR nên đổi thành JOB\_CHG\_HOUR thể hiện liên kết với JOB
    - JOB\_CLASS nên đổi thành JOB\_DESCRIPTION
    - HOURS nên đổi thành ASSIGN\_HOURS

JOB_CLASS	JOB_CHG_HOUR
Database Designer	10
DB Designer	12
DBDesigner	10



# Cải thiện thiết kế

- Các vấn đề cần giải quyết để tạo ra tập các bảng tốt:
  - Tính chỉnh thuộc tính nguyên tử:
    - Là thuộc tính không thể chia nhỏ hơn nữa
    - VD: Họ và Tên có thể tách thành Họ, Đệm và Tên
    - Có thể tạo ra tính linh hoạt khi tìm kiếm
    - VD khi tách thành Họ, Đệm và Tên thì có thể dễ dàng sắp xếp theo tên, họ, đệm.
    - Nhà thiết kế hay sử dụng thuộc tính đơn giản có giá trị duy nhất.
    - Tùy vào quy tắc nghiệp vụ của bài toán
  - Xác định các thuộc tính mới:
    - Trong bài toán thực tế có thể thêm một số trường để đáp ứng nhu cầu báo cáo, thống kê
    - Ví dụ, bảng EMPLOYEE thêm: thuộc tính ngày vào làm (EMP\_HIREDATE)

# Cải thiện thiết kế

- Các vấn đề cần giải quyết để tạo ra tập các bảng tốt:
  - Xác định mối quan hệ mới:
    - Ví dụ, cần theo dõi nhân viên nào làm quản lý mã dự án
    - Điều này dẫn đến thêm có mối quan hệ giữa EMPLOYEE và PROJECT
    - Chỉ việc thêm EMP\_NUM vào PROJECT là khóa ngoại
  - Tính chỉnh các PK khi được yêu cầu cho mức độ chi tiết của dữ liệu:
    - Ví dụ, trong bảng ASSIGNMENT có thuộc tính ASSIGN\_HOURS
    - Nhưng đó là tổng số giờ theo ngày, tuần, tháng hay năm?
    - Trong trường hợp này cần đặt câu hỏi: “có cần lưu số giờ chi tiết theo từng tuần, tháng hay năm hay không?”

# Cải thiện thiết kế

- Các vấn đề cần giải quyết để tạo ra tập các bảng tốt:
  - Duy trì độ chính xác lịch sử:
    - Trong bảng ASSIGNMENT có thuộc tính JOB\_CHG\_HOUR thể hiện mức phí làm việc theo giờ.
    - Nếu phí đó thay đổi theo thời gian thì sao? Khi lập báo cáo dự án phí đó nhân với giờ làm việc thì ra lương. Nhưng lương này tính trên phí hiện tại, phí cũ có thể thấp hơn thì sao?
  - Đánh giá việc sử dụng các thuộc tính dẫn xuất
    - Giả sử trong bảng ASSIGNMENT có thuộc tính  $ASSIGN\_CHARGE = ASSIGN\_HOURS * ASSIGN\_CHG\_HOUR$ .
    - Điều này dẫn đến phụ thuộc bắc cầu  $ASSIGN\_CHG\_HOUR * ASSIGN\_HOURS \rightarrow ASSIGN\_CHARGE$ .
    - Tuy nhiên, lưu thêm thuộc tính dẫn xuất giúp thực hiện nhanh

# Tổng hợp các cải thiện thiết kế

FIGURE 6.6 THE COMPLETED DATABASE

Table name: PROJECT

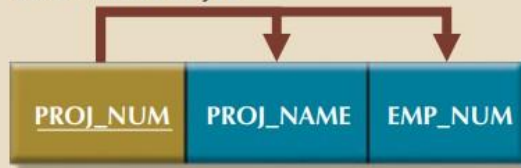


Table name: PROJECT

PROJ_NUM	PROJ_NAME	EMP_NUM
15	Evergreen	105
18	Amber Wave	104
22	Rolling Tide	113
25	Starflight	101

Table name: JOB

Database name: Ch06\_ConstructCo



Table name: JOB

JOB_CODE	JOB_DESCRIPTION	JOB_CHG_HOUR
500	Programmer	35.75
501	Systems Analyst	96.75
502	Database Designer	105.00
503	Electrical Engineer	84.50
504	Mechanical Engineer	67.90
505	Civil Engineer	55.78
506	Clerical Support	26.87
507	DSS Analyst	45.95
508	Applications Designer	48.10
509	Bio Technician	34.55
510	General Support	18.36

# Tổng hợp các cải thiện thiết kế

Table name: ASSIGNMENT

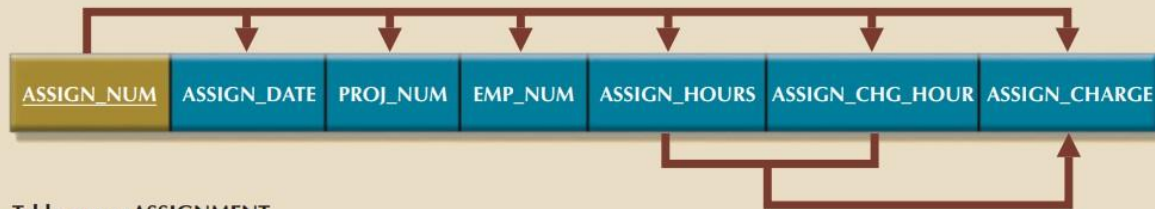


Table name: ASSIGNMENT

ASSIGN_NUM	ASSIGN_DATE	PROJ_NUM	EMP_NUM	ASSIGN_HOURS	ASSIGN_CHG_HOUR	ASSIGN_CHARGE
1001	04-Mar-18	15	103	2.6	84.50	219.70
1002	04-Mar-18	18	118	1.4	18.36	25.70
1003	05-Mar-18	15	101	3.6	105.00	378.00
1004	05-Mar-18	22	113	2.5	48.10	120.25
1005	05-Mar-18	15	103	1.9	84.50	160.55
1006	05-Mar-18	25	115	4.2	96.75	406.35
1007	05-Mar-18	22	105	5.2	105.00	546.00
1008	05-Mar-18	25	101	1.7	105.00	178.50
1009	05-Mar-18	15	105	2.0	105.00	210.00
1010	06-Mar-18	15	102	3.8	96.75	367.65
1011	06-Mar-18	22	104	2.6	96.75	251.55
1012	06-Mar-18	15	101	2.3	105.00	241.50
1013	06-Mar-18	25	114	1.8	48.10	86.58
1014	06-Mar-18	22	111	4.0	26.87	107.48
1015	06-Mar-18	25	114	3.4	48.10	163.54
1016	06-Mar-18	18	112	1.2	45.95	55.14
1017	06-Mar-18	18	118	2.0	18.36	36.72
1018	06-Mar-18	18	104	2.6	96.75	251.55
1019	06-Mar-18	15	103	3.0	84.50	253.50
1020	07-Mar-18	22	105	2.7	105.00	283.50
1021	08-Mar-18	25	108	4.2	96.75	406.35
1022	07-Mar-18	25	114	5.8	48.10	278.98
1023	07-Mar-18	22	106	2.4	35.75	85.80

# Tổng hợp các cải thiện thiết kế

FIGURE 6.6 THE COMPLETED DATABASE (CONTINUED)

Table name: EMPLOYEE

Database name: Ch06\_ConstructCo



Table name: EMPLOYEE

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_CODE
101	News	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-97	503
104	Ramoras	Anne	K	15-Nov-88	501
105	Johnson	Alice	K	01-Feb-94	502
106	Smithfield	William		22-Jun-05	500
107	Alonzo	Maria	D	10-Oct-94	500
108	Washington	Ralph	B	22-Aug-89	501
109	Smith	Larry	W	18-Jul-99	501
110	Olenko	Gerald	A	11-Dec-96	505
111	Wabash	Geoff	B	04-Apr-89	506
112	Smithson	Darlene	M	23-Oct-95	507
113	Joebrood	Delbert	K	15-Nov-94	508
114	Jones	Annelise		20-Aug-91	508
115	Bawangi	Travis	B	25-Jan-90	501
116	Pratt	Gerald	L	05-Mar-95	510
117	Williamson	Angie	H	19-Jun-94	509
118	Frommer	James	J	04-Jan-06	510

# Cân nhắc khóa thay thế

- Khi khóa chính được coi là không phù hợp, các nhà thiết kế sử dụng khóa thay thế
  - Khóa chính gồm quá nhiều thuộc tính trở nên cồng kềnh
  - Sẽ khó tạo FK khi bảng tham chiếu sử dụng khóa chính tổng hợp
  - Hoặc PK thể hiện nhiều nội dung cần bảo mật
  - Thường sử dụng khóa thay thế bằng trường tự tăng (AutoNumber)
- Không vi phạm tính toàn vẹn tham chiếu
- Bảng 6.4. dữ liệu không phù hợp vì lặp lại bản ghi nhưng không vi phạm tính toàn vẹn tham chiếu

TABLE 6.4

**DUPLICATE ENTRIES IN THE JOB TABLE**

JOB_CODE	JOB_DESCRIPTION	JOB_CHG_HOUR
511	Programmer	\$35.75
512	Programmer	\$35.75

# Cân nhắc khóa thay thế

- Thiết kế cơ sở dữ liệu thường liên quan *quan đến việc đánh đổi* việc thực hiện các đánh giá chuyên môn
  - Trong mô trường thực tế, cần cân nhắc giữa tính toàn vẹn của thiết kế và tính linh hoạt của dữ liệu.
  - Ví dụ ASSIGNMENT(PROJ\_NUM, EMP\_NUM, ASSIGN\_DATE)
    - Với ràng buộc trên thì mỗi nhân viên chỉ được tham gia 1 lần vào một ngày nhất định nào đó.
    - Nhưng với quan điểm của người quản lý là một nhân viên có thể tham gia nhiều lần vào cùng dự án
      - Điều này sẽ dẫn đến có 2 dòng (NV1, DA1, 1/1/2021) và (NV1, DA1, 30/04/2021) → sai. Không đảm bảo tính toàn vẹn.
    - → Giải pháp có thể lấy thêm ASSIGN\_DATE để làm thuộc tính P K
      - ASSIGNMENT(PROJ\_NUM, EMP\_NUM, ASSIGN\_DATE)

<u>PROJ_NUM</u>	<u>EMP_NUM</u>	<u>ASSIGN_DATE</u>
111	12	1/1/2020
111	13	2/2/2020
111	13	3/3/2021



# Chuẩn mức cao

- Các bảng đạt 3NF hoạt động phù hợp với cơ sở dữ liệu giao dịch kinh doanh.
- Tuy nhiên trong lý thuyết, nghiên cứu cần những chuẩn cao hơn:
  - BCNF - Boyce-Codd normal form
  - 4NF - Fourth normal form

# Dạng chuẩn Boyce Codd – BCNF

- Một lược đồ quan hệ ở dạng chuẩn BCNF nếu nó thỏa mãn dạng chuẩn 3 và không có thuộc tính khóa phụ thuộc hàm vào thuộc tính không khóa.
- *Định nghĩa hình thức dạng chuẩn BCNF*: nếu khi mỗi phụ thuộc hàm  $X \rightarrow A$  thỏa mãn trong R thì X là một siêu khóa của R.
- Ví dụ: Xét lược đồ R (A, B, C, D) có A là khóa chính và BC là khóa dự tuyển. Nếu tồn tại một phụ thuộc hàm  $D \rightarrow B$  thì lược đồ này vi phạm BCNF vì D không phải là siêu khóa.

# Dạng chuẩn Boyce Codd – BCNF

- Thuật toán kiểm tra lược đồ ở dạng chuẩn BCNF hay không?
  - **Input:** Lược đồ quan hệ  $R = (U, F)$
  - **Output:** Khẳng định  $R$  có đạt dạng chuẩn BCNF hay không
- Phương pháp:
  - B1: Tìm tất cả khóa của  $R$
  - B2: Từ  $F$  tạo tập phụ thuộc hàm tương đương  $G$  có vẻ phải một thuộc tính.
  - B3: Nếu mọi phụ thuộc hàm  $X \rightarrow A$  với  $A \notin X$  đều có  $X$  là siêu khóa thì  $R$  đạt chuẩn BCNF, ngược lại  $R$  không đạt chuẩn BCNF.



# Dạng chuẩn Boyce Codd – BCNF

- Ví dụ: Cho lược đồ quan hệ  $R = \{U, F\}$ .
  - $U = \{A, B, C, D, E, I\}$
  - $F = \{ACD \rightarrow EBI; CE \rightarrow AD\}$
- Hỏi  $R$  có đạt chuẩn BCNF hay không?

# Dạng chuẩn Boyce Codd – BCNF

Xi	Xi U TN	(Xi U TN) <sup>+</sup>	SK	Khóa
0	C	C		
A	AC	AC		
D	CD	CD		
E	CE	ABCDEI	CE	CE
AD	ACD	ABCDEI	ACD	ACD
AE	ACE	ABCDEI	ACE	
DE	CDE	ABCDEI	CDE	
ADE	ACDE	ABCDEI	ACDE	

- $F = F_{tt} = \{ACD \rightarrow E, ACD \rightarrow B, ACD \rightarrow I, CE \rightarrow A, CE \rightarrow D\}$
- Mọi phụ thuộc hàm của  $F_{tt}$  đều có vế trái là siêu khóa  $\Rightarrow Q$  đạt dạng chuẩn BCNF

# Phụ thuộc hàm đa trị và dạng chuẩn 4 (4NF)

## a) Phụ thuộc hàm đa trị (multivalued dependency)

Ký hiệu:  $X \twoheadrightarrow Y$ , đọc: "X xác định hàm đa trị Y" hay "Y phụ thuộc hàm đa trị vào X"

# Phụ thuộc hàm đa trị

➤ Ta nói:  $X \twoheadrightarrow Y$  đúng trong  $R$  nếu với  $r$  là quan hệ của  $R$ ;  $t_1$  và  $t_2$  là hai bộ trong  $r$  với  $t_1[X] = t_2[X]$  và  $Z = R - XY$  thì  $r$  chứa các bộ  $t_3$  và  $t_4$ , trong đó:

$$(1) \ t_3[X] = t_4[X] = t_1[X] = t_2[X]$$

$$(2) \ t_3[Y] = t_1[Y] \text{ và } t_3[Z] = t_2[Z]$$

# Phụ thuộc hàm đa trị

- Ví dụ: Mã khách hàng →→ Số hóa đơn  
Mã khách hàng →→ Tên hàng

Mã khách hàng	Số hóa đơn	Tên hàng
KH100	1021	Ipad2
KH100	1022	Iphone5S
KH100	1021	Iphone5S
KH100	1022	Ipad2



# Phụ thuộc hàm đa trị tầm thường

- $X \twoheadrightarrow Y$  được gọi là một phụ thuộc hàm đa trị tầm thường nếu nó thỏa mãn điều kiện sau:
  - $Y$  là một tập con của  $X$ .
  - Hoặc  $X \cup Y = R$
- $X \twoheadrightarrow Y$  được gọi là một phụ thuộc hàm đa trị không tầm thường nếu chúng không thỏa mãn các điều kiện trên.

# Phụ thuộc hàm đa trị

## b) Các tiên đề cho phụ thuộc hàm và phụ thuộc đa trị

Với  $X, Y, Z, V, W \subseteq U$

A1: Tính phản xạ: Nếu  $Y \subseteq X$  thì  $X \rightarrow Y$

A2: Tính tăng trưởng: Nếu  $X \rightarrow Y$  thì  $XZ \rightarrow YZ$

A3: Tính bắc cầu: Nếu  $X \rightarrow Y$  và  $Y \rightarrow Z$  thì  $X \rightarrow Z$

A4: Tính bù: Nếu  $X \twoheadrightarrow Y$  thì  $X \twoheadrightarrow U - XY$

A5: Tính tăng trưởng của phụ thuộc đa trị:

Nếu  $X \twoheadrightarrow Y$  và  $V \subseteq W$  thì  $WX \twoheadrightarrow VY$

A6: Tính bắc cầu của phụ thuộc đa trị:

Nếu  $X \twoheadrightarrow Y$  và  $Y \twoheadrightarrow Z$  thì  $X \twoheadrightarrow Z - Y$

# Phụ thuộc hàm đa trị

A7: Nếu  $X \rightarrow Y$  thì  $X \twoheadrightarrow Y$

A8: Nếu  $X \twoheadrightarrow Y$  và  $W \cap Y = \emptyset$ ,  $W \rightarrow Z$ ,  $Z \subseteq Y$  thì  $X \twoheadrightarrow Z$

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

- Định nghĩa: Bao đóng của tập các phụ thuộc hàm và phụ thuộc hàm đa trị  $D$  là tập tất cả các phụ thuộc hàm và các phụ thuộc hàm đa trị được suy diễn logic từ  $D$ .
- Ký hiệu:  $D^+$

# Phụ thuộc hàm đa trị

## c) Dạng chuẩn 4NF

Một lược đồ quan hệ ở dạng chuẩn 4 đối với một tập các phụ thuộc D nếu với mỗi phụ thuộc hàm đa trị không tầm thường  $X \twoheadrightarrow Y$  trong  $D^+$ ,  $X$  là một siêu khóa của  $R$ .

# Ví dụ

Mã khách hàng	Số hóa đơn
KH100	1021
KH100	1022

Mã khách hàng	Tên hàng
KH100	Ipad2
KH100	Iphone5S

Mã khách hàng →→ Số hóa đơn

Mã khách hàng →→ Tên hàng

# Thuật toán tách quan hệ về dạng chuẩn 4 với tính chất nối không mất

**Input:** Lược đồ R và tập phụ thuộc hàm đa trị D chưa ở dạng chuẩn 4.

**Output:** Tách R thành các lược đồ dạng 4 có tính chất nối không mất thông tin.

**Phương pháp:** Khi có một lược đồ quan hệ không thuộc dạng 4, ta tìm một phụ thuộc đa trị không tầm thường  $X \twoheadrightarrow Y$  vi phạm 4NF, thay thế nó bằng hai lược đồ XY và R – Y.

# Dạng chuẩn 4 (Fourth Normal Form - 4NF)

- 4NF tránh việc dư thừa do phụ thuộc đa trị
- Ví dụ: bảng Dangky(MaSV, MaMon, MaSach) không phải 4NF do tồn tại MVD MaMon  $\rightarrow\rightarrow$  MaSV|MaSach
- Để chuẩn hoá thành 4NF, chia bảng Dangky thành 2 bảng
  - Dangky(MaSV, MaMon)
  - Quydingh(MaMon, MaSach)



# Chuẩn hóa và Thiết kế cơ sở dữ liệu

- Chuẩn hóa phải là một phần của quá trình thiết kế
- Đảm bảo rằng các thực thể được đề xuất đáp ứng chuẩn trước khi tạo cấu trúc bảng
- Nhiều cơ sở dữ liệu trong thực tế đã thiết kế không đúng hoặc có những bất thường dữ liệu
  - Có thể được yêu cầu thiết kế lại hoặc sửa đổi lại cơ sở dữ liệu hiện có.
- Do đó, nên biết các nguyên tắc và quy trình thiết kế tốt cũng như các quy trình chuẩn hóa.

# Chuẩn hóa và Thiết kế cơ sở dữ liệu

- Sơ đồ ER – ERD:
  - Xác định các thực thể có liên quan, thuộc tính và mối quan hệ của thực thể
  - Xác định các thực thể và thuộc tính bổ sung
- Quy trình chuẩn hóa:
  - Tập trung vào đặc điểm của các thực thể cụ thể
  - Xem xét chi tiết từng thực thể với ERD
- Khó tách quy trình chuẩn hóa khỏi quy trình ERD

# Phi chuẩn - Denormalization

- Tạo các bảng là mục tiêu quan trọng của thiết kế cơ sở dữ liệu
- Xử lý các yêu cầu dữ liệu cũng là một mục tiêu
- Nếu bảng được phân rã để phù hợp với các yêu cầu chuẩn hóa:
  - Sẽ thêm các bảng trong cơ sở dữ liệu
- Kết nối trên nhiều bảng sẽ làm giảm tốc độ của hệ thống
- Sự xung đột trên thường được giải quyết thông qua các *thỏa hiệp* có thể bao gồm việc không chuẩn hóa
- Các khiếm khuyết của bảng không chuẩn:
  - Cập nhật dữ liệu kém hiệu quả hơn vì các bảng lớn hơn
  - Lập chỉ mục phức tạp hơn
  - Tạo bảng ảo (khung nhìn) phức tạp

# **Danh sách kiểm mô hình dữ liệu (Checklist)**

- Mô hình hóa dữ liệu chuyển môi trường thế giới thực thành mô hình dữ liệu
- Đại diện cho dữ liệu trong thế giới thực, người dùng, quy trình, tương tác
- Danh sách kiểm tra mô hình hóa dữ liệu giúp đảm bảo rằng các tác vụ mô hình hóa dữ liệu được thực hiện thành công
- Dựa trên các khái niệm và công cụ đã học

# Danh sách kiểm mô hình dữ liệu (Checklist)

## DATA-MODELING CHECKLIST

### Quy tắc nghiệp vụ - Bussiness Rule

- Tài liệu thích hợp để xác minh tất cả các quy tắc nghiệp vụ với người dùng cuối.
- Đảm bảo rằng tất cả các quy tắc nghiệp vụ được viết chính xác, rõ ràng và đơn giản. Các quy tắc nghiệp vụ phải giúp xác định các thực thể, thuộc tính, mối quan hệ và các ràng buộc.
- Xác định nguồn gốc của tất cả các quy tắc nghiệp vụ và đảm bảo rằng mỗi quy tắc nghiệp vụ đều được xác minh, ghi ngày và ký bởi cơ quan phê duyệt.

### Mô hình hóa dữ liệu – Data Modeling

**Quy ước đặt tên:** Tất cả các tên phải được giới hạn độ dài (kích thước phụ thuộc vào cơ sở dữ liệu)

#### -Tên thực thể:

- Nên là những danh từ quen thuộc với kinh doanh và phải ngắn gọn và có nghĩa
- Nên ghi lại các từ viết tắt, từ đồng nghĩa và bí danh cho từng thực thể
- Phải là duy nhất trong mô hình
- Đối với các đối tượng kết hợp, có thể bao gồm tổ hợp các tên viết tắt của các đối tượng được liên kết thông qua đối tượng kết hợp

#### -Tên thuộc tính:

- Phải là duy nhất trong thực thể
- Nên sử dụng chữ viết tắt của thực thể làm tiền tố
- Nên mô tả đặc tính
- Nên sử dụng các hậu tố như \_ID, \_NUM hoặc \_CODE cho thuộc tính PK
- Không nên là một từ dành riêng
- Không được chứa khoảng trắng hoặc các ký tự đặc biệt như @,!, Hoặc &

#### -Tên mối quan hệ:

- Nên là động từ chủ động hoặc bị động chỉ rõ bản chất của mối quan hệ

# Danh sách kiểm mô hình dữ liệu (Checklist)

## DATA-MODELING CHECKLIST

### Thực thể - Entities:

- Mỗi thực thể nên đại diện cho một chủ thể duy nhất.
- Mỗi thực thể phải đại diện cho một tập hợp các cá thể thực thể có thể phân biệt được.
- Tất cả các thực thể phải ở 3NF hoặc cao hơn. Bất kỳ thực thể nào dưới 3NF phải được chứng minh.
- Mức độ chi tiết của cá thể thực thể cần được xác định rõ ràng.
- PK phải được xác định rõ ràng và hỗ trợ độ chi tiết của dữ liệu đã chọn.

### Thuộc tính –Attributes:

- Phải đơn giản và có giá trị đơn (dữ liệu nguyên tử)
- Nên ghi lại các giá trị, ràng buộc, từ đồng nghĩa và bí danh mặc định
- Các thuộc tính có nguồn gốc phải được xác định rõ ràng và bao gồm (các) nguồn
- Không nên thừa trừ khi điều này là cần thiết để đảm bảo tính chính xác, hiệu suất của giao dịch hoặc duy trì lịch sử
- Thuộc tính nonkey phải hoàn toàn phụ thuộc vào thuộc tính PK

### Mối quan hệ - Relationships:

- Nên xác định rõ những người tham gia mối quan hệ
- Nên xác định rõ ràng vai trò tham gia và ~~lượng~~

### Mô hình ER –ER Model:

- Cần được xác thực theo các quy trình dự kiến: chèn, cập nhật và xóa
- Nên đánh giá địa điểm, khi nào và làm thế nào để duy trì lịch sử
- Không nên chứa các mối quan hệ thừa trừ khi được yêu cầu (xem thuộc tính)
- Nên giảm thiểu dư thừa dữ liệu để đảm bảo cập nhật ở một nơi duy nhất
- Phải tuân theo quy tắc dữ liệu tối thiểu: Tất cả những gì cần thiết là có, và tất cả những gì cần thiết là có