

Korea IT School 2022

# CƠ SỞ DỮ LIỆU

Unit 6: Chuẩn hóa các bảng dữ liệu

# Nội dung

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

- Thiết kế cơ sở liệu 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 chuẩn không?
- Có thể dựa trên các tiêu chuẩn về bảng để có định hướng thiết kế tốt hơn

- Chuẩn hóa:
  - Là quá trình đánh giá và hiệu chỉnh cấu trúc bảng để giảm thiểu những dư thừa dữ liệu
    - Giảm các bất thường với thao tác dữ liệu
  - Gồm có 1 chuỗi các giai đoạn chuẩn hóa:
    - 1NF (First Normal Form) – Dạng chuẩn 1
    - 2NF (Second Normal Form) - Dạng chuẩn thứ 2
    - 3NF (Third normal form) – Dạng chuẩn 3
    - BCNF (Boyce-Codd normal form) – Dạng chuẩn Boyce-Code
    - 4NF (Fourth Normal Form) – Dạng chuẩn 4
    - ....



- Chuẩn hóa:
  - 2NF tốt hơn 1NF; 3NF tốt hơn 2NF ;...
  - Đối với hầu hết mục đích thiết kế cơ sở dữ liệu thì 3NF là mức cao nhất cần thiết.
  - Mức độ chuẩn hóa cao nhất không hẳn lúc nào cũng mong muốn nhất
    - Chuẩn hóa càng cao thì yêu cầu tách (phân rã) thành nhiều bảng, dẫn đến chi phí kết nối giữa các bảng cao.
- Phi chuẩn sẽ tạo ra mức chuẩn thấp hơn
  - Tăng tốc độ thực thi, nhưng dư thừa dữ liệu lớn hơn

- Người thiết kế CSDL sử dụng chuẩn hóa trong 2 trường hợp
  - Sau khi vẽ ERD, sử dụng chuẩn hóa để phân tích mối quan hệ bên trong mỗi thực thể và xác định xem cấu trúc có cải thiện qua chuẩn hóa không?
  - Sửa đổi cấu trúc dữ liệu hiện có: tệp, bảng tính, cấu trúc cũ thành các cấu trúc phù hợp
- Thông thường:
  - Bắt đầu xác định các quy tắc nghiệp vụ và ràng buộc dữ liệu, xác định các phụ thuộc hàm, thực thể và thuộc tính
  - Sau đó, bạn áp dụng các khái niệm chuẩn hóa để xác thực và tinh chỉnh thêm mô hình.

- Qua quá trình chuẩn hóa sẽ:
  - Biết cách thiết kế CSDL như thế nào để đáp ứng được các ràng buộc toàn vẹn dữ liệu, báo cáo dữ liệu và yêu cầu thực thi
  - Các bước trong quá trình thiết kế:
    - Xác định quy tắc nghiệp vụ
    - Xác định và định nghĩa các ràng buộc nghiệp vụ và dữ liệu
    - Xác định các phụ thuộc hàm
    - Xác định các thực thể và mối quan hệ
    - Loại bỏ các thuộc tính đa trị
- Mục tiêu chính của chuẩn hóa là loại bỏ các bất thường về dữ liệu
  - Loại bỏ đi các dư thừa dữ liệu dựa trên các phụ thuộc hàm

- Ví dụ, hoạt động công ty quản lý các dự án xây dựng
  - Mỗi dự án có số dự án riêng, tên, nhân viên được chỉ định, v.v.
  - Mỗi nhân viên có một mã số, tên và phân loại công việc, chẳng hạn như kỹ sư hoặc kỹ thuật viên máy tính.
  - Tính phí khách hàng dựa trên số giờ sử dụng cho mỗi hợp đồng.
  - Tỷ lệ thanh toán hàng giờ phụ thuộc vào phân loại công việc của nhân viên. Ví dụ: một giờ dành cho kỹ thuật viên máy tính được tính phí theo tỷ lệ khác với một giờ dành cho kỹ sư.
  - Định kỳ, một báo cáo dự án được tạo có chứa thông tin hiển thị trong như hình dưới



# Nhu cầu chuẩn hóa

- Giả sử có bảng như hình bên

A SAMPLE REPORT LAYOUT

PROJECT NUMBER	PROJECT NAME	EMPLOYEE NUMBER	EMPLOYEE NAME	JOB CLASS	CHARGE/HOUR	HOURS BILLED	TOTAL CHARGE
15	Evergreen	103	June E. Arbough	Elec. Engineer	\$ 84.50	23.8	\$ 2,011.10
		101	John G. News	Database Designer	\$105.00	19.4	\$ 2,037.00
		105	Alice K. Johnson *	Database Designer	\$105.00	35.7	\$ 3,748.50
		106	William Smithfield	Programmer	\$ 35.75	12.6	\$ 450.45
		102	David H. Senior	Systems Analyst	\$ 96.75	23.8	\$ 2,302.65
				<b>Subtotal</b>			<b>\$10,549.70</b>
18	Amber Wave	114	Annelise Jones	Applications Designer	\$ 48.10	24.6	\$ 1,183.26
		118	James J. Frommer	General Support	\$ 18.36	45.3	\$ 831.71
		104	Anne K. Ramoras *	Systems Analyst	\$ 96.75	32.4	\$ 3,134.70
		112	Darlene M. Smithson	DSS Analyst	\$ 45.95	44.0	\$ 2,021.80
				<b>Subtotal</b>			<b>\$ 7,171.47</b>
22	Rolling Tide	105	Alice K. Johnson	Database Designer	\$105.00	64.7	\$ 6,793.50
		104	Anne K. Ramoras	Systems Analyst	\$96.75	48.4	\$ 4,682.70
		113	Delbert K. Joenbrood *	Applications Designer	\$48.10	23.6	\$ 1,135.16
		111	Geoff B. Wabash	Clerical Support	\$26.87	22.0	\$ 591.14
		106	William Smithfield	Programmer	\$35.75	12.8	\$ 457.60
				<b>Subtotal</b>			<b>\$13,660.10</b>
25	Starflight	107	Maria D. Alonzo	Programmer	\$ 35.75	24.6	\$ 879.45
		115	Travis B. Bawangi	Systems Analyst	\$ 96.75	45.8	\$ 4,431.15
		101	John G. News *	Database Designer	\$105.00	56.3	\$ 5,911.50
		114	Annelise Jones	Applications Designer	\$ 48.10	33.1	\$ 1,592.11
		108	Ralph B. Washington	Systems Analyst	\$ 96.75	23.6	\$ 2,283.30
		118	James J. Frommer	General Support	\$ 18.36	30.5	\$ 559.98
		112	Darlene M. Smithson	DSS Analyst	\$ 45.95	41.4	\$ 1,902.33
				<b>Subtotal</b>			<b>\$17,559.82</b>
				<b>Total</b>			<b>\$48,941.09</b>

Note: \* indicates the project leader.

- Nhận xét bảng trên:
  - Bảng trên không phù hợp với yêu cầu của bảng dữ liệu
  - Cấu trúc dữ liệu đưa ra những mâu thuẫn
    - Giá trị JOB\_CLASS là “Elect. Engineer” cũng có thể nhập là: “Elect.Eng.”, “El. Eng.” hoặc “ EE”,...
    - John G. News và Alice K. Johnson cùng công việc nhưng có phí khác nhau
  - Cấu trúc dữ liệu chứa một số thuộc tính đa giá trị làm cho các nhiệm vụ quản lý dữ liệu trở nên rất khó khăn
    - Không trả lời được: "Có bao nhiêu nhân viên đang làm việc trong dự án Starlight?"
  - Dữ liệu nhân viên bị dư thừa trong bảng vì nhân viên có thể làm việc trên nhiều dự án.

- Nhận xét bảng trên:
  - Cấu trúc dữ liệu này tạo ra sự mâu thuẫn dữ liệu
  - Báo cáo có thể mang lại các kết quả khác nhau tùy thuộc vào sự bất thường của dữ liệu.
  - Không thể lập báo cáo: tổng số giờ của công việc “Database Design”
- Các vấn đề phải được giải quyết trong quá trình thiết kế cơ sở dữ liệu:
  - Tính toàn vẹn dữ liệu,
  - Dư thừa dữ liệu và
  - Không nhất quán dữ liệu

- Mỗi bảng đại diện cho một chủ đề duy nhất
  - Bảng SINHVIEN, bảng MONHOC, bảng GIAOVIEN,...
- Giao nhau giữa dòng và cột chỉ gồm 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

Chuẩn	Đặc điểm
1NF	Bảng định dạng, không lặp lại nhóm, đã có khóa chính (PK)
2NF	1NF và không có phụ thuộc hàm bộ phận
3NF	2NF và không có phụ thuộc bắc cầu
BCNF	Mỗi yếu tố quyết định là một khóa ứng viên (trường hợp đặc biệt của 3NF)
4NF	3NF và không có phụ thuộc nhiều giá trị độc lập

- **Mục tiêu** của chuẩn hóa là đảm bảo rằng tất cả các bảng ít nhất đạt **3NF**
- Các chuẩn cao hơn (BCNF, 4NF,..) 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
- Có 2 phụ thuộc hàm cần quan tâm:
  - **Phụ thuộc hàm bộ phận**
    - Bảng(Mã SV, Họ tên, Mã môn, Tên môn, Điểm)
    - PTH={Mã SV  $\rightarrow$  Họ tên; Mã môn  $\rightarrow$  Tên môn; Mã SV, Mã môn  $\rightarrow$  Điểm}
    - Khóa: Mã SV, Mã môn
    - Ta có: Họ tên, Tên môn chỉ phụ thuộc vào 1 phần của Khóa
  - **Phụ thuộc hàm bắc cầu**
    - Mã môn  $\rightarrow$  Tên môn, Tên GV; Tên GV  $\rightarrow$  Phòng
    - PTH bắc cầu: Mã môn  $\rightarrow$  Tên GV  $\rightarrow$  Phòng



Khái niệm	Định nghĩa
Phụ thuộc hàm	<p>Kí hiệu: <math>A \rightarrow B</math>. 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</p> <p>VD: <math>PROJ\_NUM \rightarrow PROJ\_NAME</math>, PROJ_NUM xác định thuộc tính PROJ_NAME</p> <ul style="list-style-type: none"><li>- PROJ_NUM là thuộc tính xác định</li><li>- PROJ_NAME là thuộc tính phụ thuộc</li></ul>
Phụ thuộc hàm (định nghĩa khái quát)	<p>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</p>
Phụ thuộc hàm đầy đủ (khóa tổng hợp)	<p>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</p>

- Bắt đầu bằng cách trình bày dữ liệu ở dạng bảng
  - Mỗi ô (giao nhau giữa hàng và cột) chỉ có 1 giá trị duy nhất và không có nhóm lặp
- Nhóm lặp:
  - Nhóm nhiều thực thể có cùng kiểu có thể tồn tại cho 1 thuộc tính
- Bảng quan hệ không chứa các nhóm lặp
  - NHANVIEN(MaNV, TenNV, GioiTinh, KyNang1, KyNang2, KyNang3)
- Chuẩn hóa cấu trúc bảng sẽ làm giảm dư thừa dữ liệu
- Chuẩn hóa là quy trình 3 bước:

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

- 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 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
  - Bước 3, Xác định tất cả các yếu tố phụ thuộc
    - 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

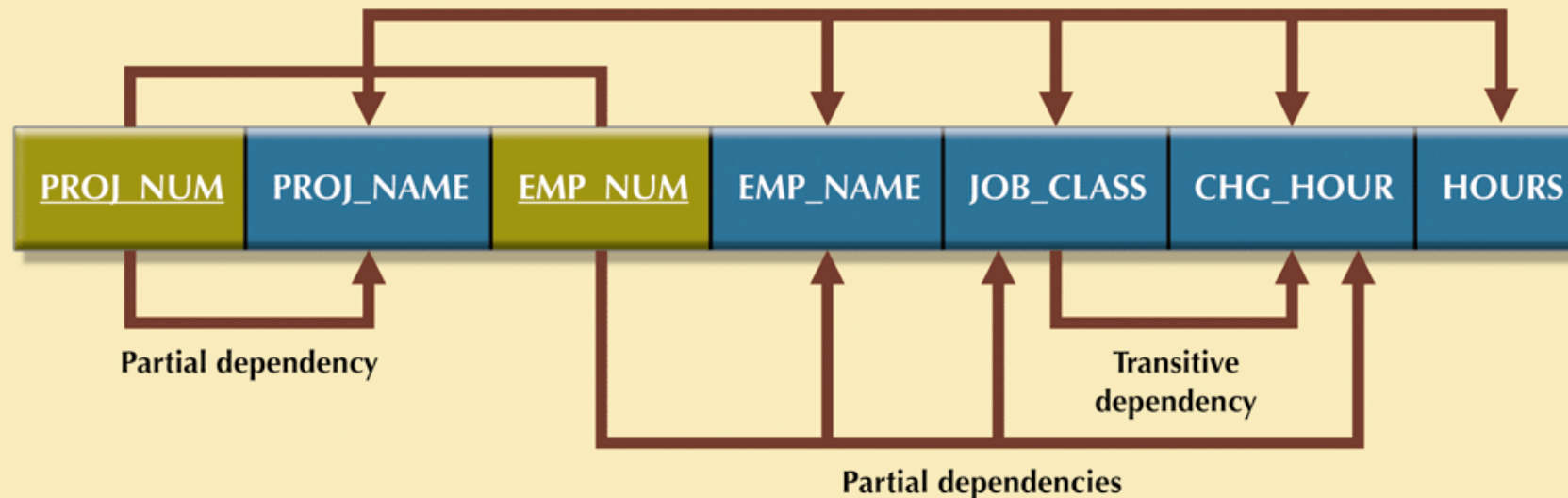
- Sơ đồ phụ thuộc:
  - Là mô tả tất cả các phụ thuộc được tìm thấy trong cấu trúc bảng nhất định
  - Hữu ích trong việc có được cái nhìn tổng thể về tất cả các mối quan hệ giữa các thuộc tính của bảng
  - Làm cho nó ít có khả năng bạn sẽ bỏ qua một phụ thuộc quan trọng



- 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  $\Rightarrow$  PROJ\_NAME)

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

TRANSITIVE DEPENDENCY:

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

SOURCE: Course Technology/Cengage Learning

Copyright by Multicampus Co., Ltd. All right reserved

- 1NF mô tả định dạng bảng:
  - Tất cả các thuộc tính khóa được **xác định**
  - Không có **nhóm lặp** lại trong bảng
  - Tất cả các thuộc tính đều dựa vào khóa chính
- 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
- Kiểm tra các bất thường khi: chèn, sửa, xóa???

- Bước 1: Tạo bảng mới để loại bỏ sự **phụ thuộc hàm bộ phận**
  - Viết mỗi khóa thành phần trên một dòng, khóa tổng hợp dòng cuối
  - Mỗi thành phần này sẽ trở thành bảng mới
  - Ví dụ, với bảng 6.2
    - PROJ\_NUM → ???
    - EMP\_NUM → ???
    - PROJ\_NUM, EMP\_NUM → ???
  - Mỗi thuộc tính trên kết hợp với thuộc tính tương ứng tạo thành các bảng: PROJECT, EMPLOYEE, và ASSIGNMENT

- Bước 2: Gán thêm các thuộc tính phụ thuộc tương ứng
  - Xác định các thuộc tính phụ thuộc vào các thuộc tính khóa bước 1
    - Dựa vào hình 6.3
  - Các thuộc tính phụ thuộc được xóa khỏi bảng gốc
    - Tạo bảng mới với PTH mới
    - VD: PROJECT(PROJ\_NUM, PROJ\_NAME)
      - Với PTH={PROJ\_NUM→PROJ\_NAME}
    - EMPLOYEE (EMP\_NUM, EMP\_NAME, JOB\_CLASS, CHG\_HOUR)
      - Với PTH = {EMP\_NUM → EMP\_NAME, JOB\_CLASS, CHG\_HOUR; JOB\_CLASS → CHG\_HOUR}
    - ASSIGNMENT (PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)
      - Với PTH = {PROJ\_NUM, EMP\_NUM → ASSIGN\_HOURS}
- Tại thời điểm này, **hầu hết** các dị thường đã được loại bỏ

# Chuyển đổi thành 2NF

Second normal form (2NF) conversion results

FIGURE 6.4

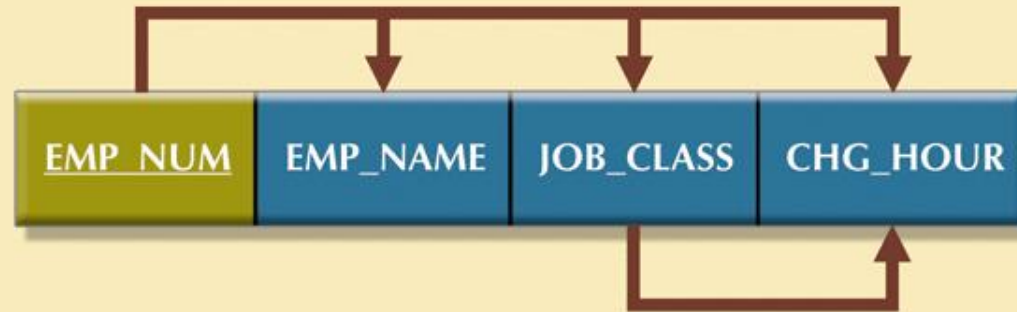
Table name: PROJECT

PROJECT (PROJ\_NUM, PROJ\_NAME)



Table name: EMPLOYEE

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



TRANSITIVE DEPENDENCY  
(JOB\_CLASS → CHG\_HOUR)

Transitive  
dependency

Table name: ASSIGNMENT

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





- Bảng đạt 2NF khi:
  - Bảng đạt 1NF
  - **Không bao gồm PTH bộ phận**
    - Không có thuộc tính nào chỉ phụ thuộc vào 1 phần của khóa chính
- Hình 6.4. có PTH bắc cầu trong bảng EMPLOYEE
  - EMPLOYEE (EMP\_NUM, EMP\_NAME, JOB\_CLASS, CHG\_HOUR)
    - Với PTH = {**EMP\_NUM** → EMP\_NAME, **JOB\_CLASS**, CHG\_HOUR;  
**JOB\_CLASS** → **CHG\_HOUR**}
  - Nếu mức phí mỗi giờ (**CHG\_HOUR**) thay đổi đối với phân loại công việc (**JOB\_CLASS**) thì cần cập nhật cho tất cả nhân viên
  - Nếu quên cập nhật sẽ dẫn đến cùng công việc nhưng mức phí khác nhau

- Bước 1, tạo bảng mới để **loại bỏ PTH bắc cầu**
  - Với mọi PTH bắc cầu thì viết định thức như PK của bảng mới
  - **Định thức:** bất kỳ thuộc tính nào có giá trị xác định giá trị khác
    - VD: Định thức JOB\_CLASS có giá trị xác định giá trị của CHG\_HOUR
- Bước 2, gán lại các thuộc tính phụ thuộc tương ứng
  - Xác định các thuộc tính phụ thuộc vào từng định thức ở bước 1
    - CHG\_HOUR phụ thuộc vào JOB\_CLASS
  - Xác định sự PTH
    - JOB\_CLASS  $\rightarrow$  CHG\_HOUR
  - Đặt tên bảng mới phản ánh nội dung của bảng mới
    - JOB(JOB\_CLASS, CHG\_HOUR)
  - Bảng mới EMPLOYEE (EMP\_NUM, EMP\_NAME, JOB\_CLASS)
    - Với PTH={EMP\_NUM  $\rightarrow$  EMP\_NAME, JOB\_CLASS}

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)

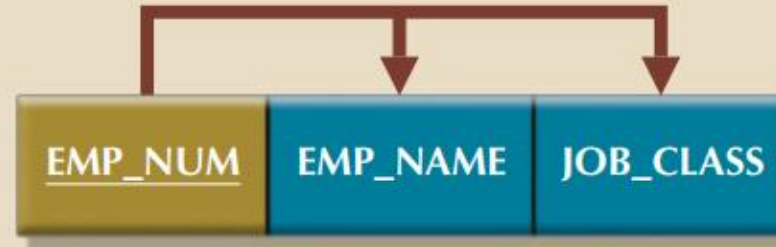


Table name: EMPLOYEE

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



Table name: ASSIGNMENT

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

- 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 thỏa mãn 2 điều kiện sau:**
  - Bảng đạt 2NF
  - Không bao gồm các PTH bắc cầu

- 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



- 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 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)

- 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 mỗi 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
  - Tinh chỉnh các PK khi được yêu cầu cho mức độ chi tiết của dữ liệu:
    - Ví dụ, tong 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?”

- 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 thuộc tính bắc cầu
      - $ASSIGN\_CHARGE + ASSIGN\_HOURS \rightarrow ASSIGN\_CHG\_HOUR$
    - Tuy nhiên, lưu thêm thuộc tính dẫn xuất giúp thực hiện nhanh

FIGURE 6.6 THE COMPLETED DATABASE

Table name: PROJECT

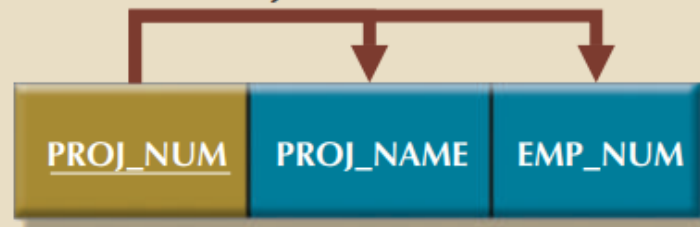


Table name: PROJECT

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

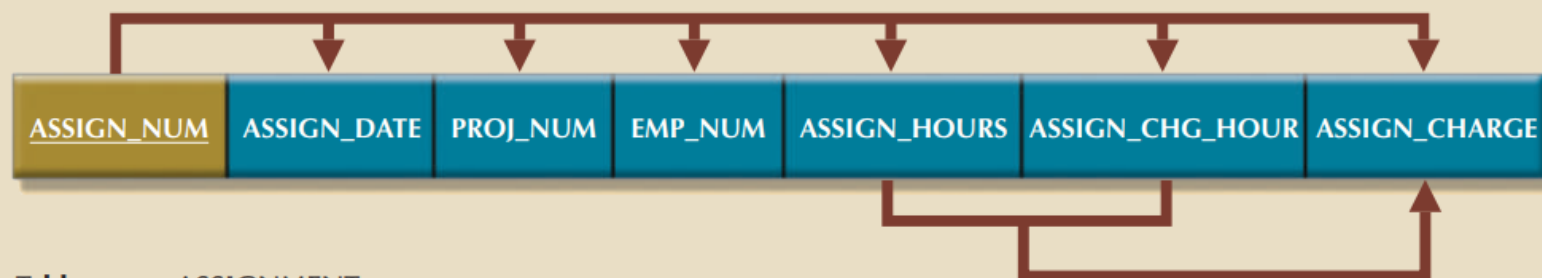


Table name: ASSIGNMENT

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



FIGURE 6.6 THE COMPLETED DATABASE (CONTINUED)

Table name: EMPLOYEE

Database name: Ch06\_ConstructCo

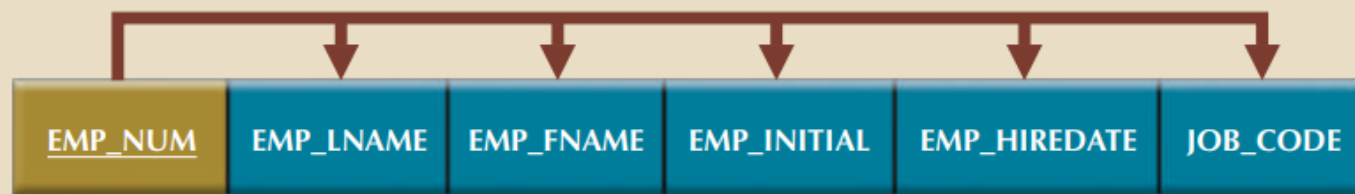


Table name: EMPLOYEE

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

- 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 sẽ 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

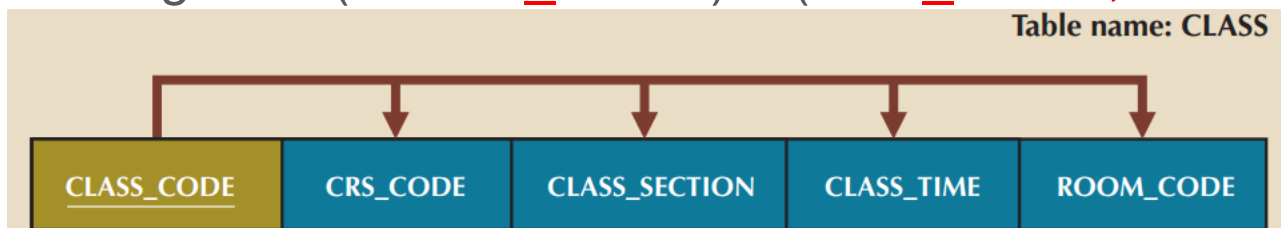
**DUPLICATE ENTRIES IN THE JOB TABLE**

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

- Thiết kế cơ sở dữ liệu thường liên quan quan đến việc **đánh đổi** việc thực hiện các đánh giá chuyên môn
  - Trong môi 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 PK
      - ASSIGNMENT(PROJ\_NUM, EMP\_NUM, ASSIGN\_DATE)

- 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

- Một bảng đạt chuẩn BCNF khi mọi định thức đều là khóa ứng viên
  - **Định thức:** bất kỳ thuộc tính nào có giá trị xác định giá trị khác
  - **Khóa ứng viên:** 1 hay 1 tập thuộc tính suy tất cả các thuộc tính
    - Từ tập khóa ứng viên có thể chọn ra khóa chính
- Một bảng R đạt chuẩn BCNF thì mọi PTH  $X \rightarrow A$  trong bảng R thì X phải là **khóa ứng viên**
- Khi 1 bảng chỉ có 1 khóa ứng viên thì chuẩn 3NF cũng đạt BCNF luôn
- Ví dụ, CLASS(CLASS\_CODE, CRS\_CODE, CLASS\_SECTION, CLASS\_TIME, ROOM\_CODE)
  - 2 khóa ứng viên: (**CLASS\_CODE**) & (**CRS\_CODE, CLASS\_SECTION**)

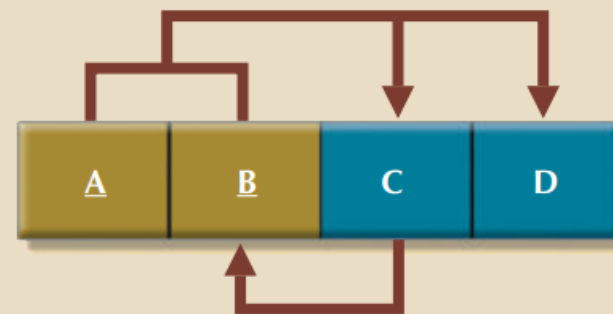




- Đa số các thiết kế đều coi BCNF là trường hợp đặc biệt của 3NF.
- Bảng ở 3NF khi ở 2NF và không có PTH bậc cầu
- Bảng có thể ở 3NF, nhưng không ở BCNF khi:
  - Không có PTH bộ phận, **không có PTH bậc cầu**
  - Thuộc tính **không khóa** phụ thuộc vào thuộc tính khóa
    - SINHVIEN(MaSV, Hoten, Ngaysinh),
    - PTH={Masv  $\rightarrow$  Hoten, Ngaysinh},
    - Khóa: MaSV.
    - Thuộc tính khóa: MaSV; Thuộc tính không khóa: Hoten, Ngaysinh

- Bảng với PTH hình 6.8 đạt 3NF, nhưng không đạt BCNF

FIGURE 6.8 A TABLE THAT IS IN 3NF BUT NOT IN BCNF

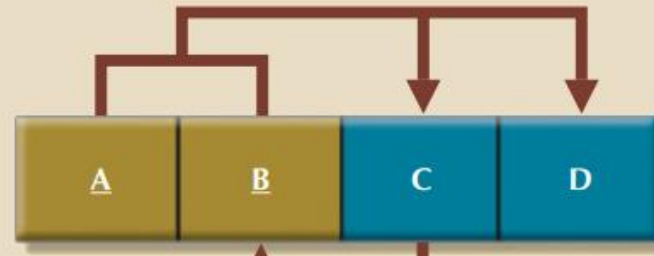


- Trong hình trên có PTH:
  - $A, B \rightarrow C, D$ ;  $A, C \rightarrow B, D$ ;  $C \rightarrow B$
  - $(A, B)$  và  $(A, C)$  là khóa ứng viên
  - $C \rightarrow B$  không phải là PTH bắc cầu vì B là thuộc tính nguyên tố
    - Thuộc tính nguyên tố là thuộc tính của khóa
    - $A, B \rightarrow C, D$  &  $C \rightarrow B$ , nhưng B đã nằm trong thuộc tính của khóa rồi
  - Nên bảng chứa PTH trên đạt chuẩn 3NF, nhưng không đạt BCNF vì:
    - Có  $C \rightarrow B$  nhưng C không phải là khóa ứng viên

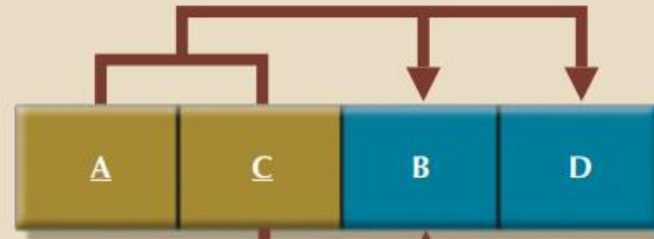
- 3NF:
  - Mọi  $X \rightarrow A$  thì  $X$  là khóa ứng viên hoặc  $A$  nguyên tố
    - Thuộc tính nguyên tố là 1 thuộc tính của khóa
- BCNF:
  - Mọi  $X \rightarrow A$  thì  $X$  là khóa ứng viên

FIGURE 6.9 DECOMPOSITION TO BCNF

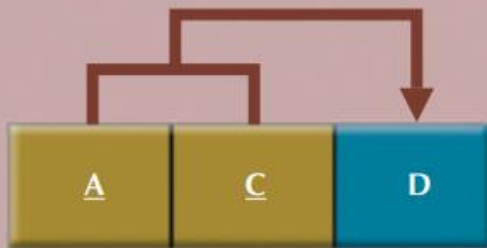
3NF, but not BCNF



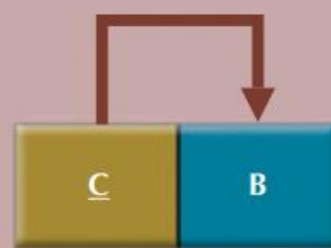
1NF



Partial dependency



3NF and BCNF



3NF and BCNF

# Phân rã thành BCNF

- Giả sử có bảng

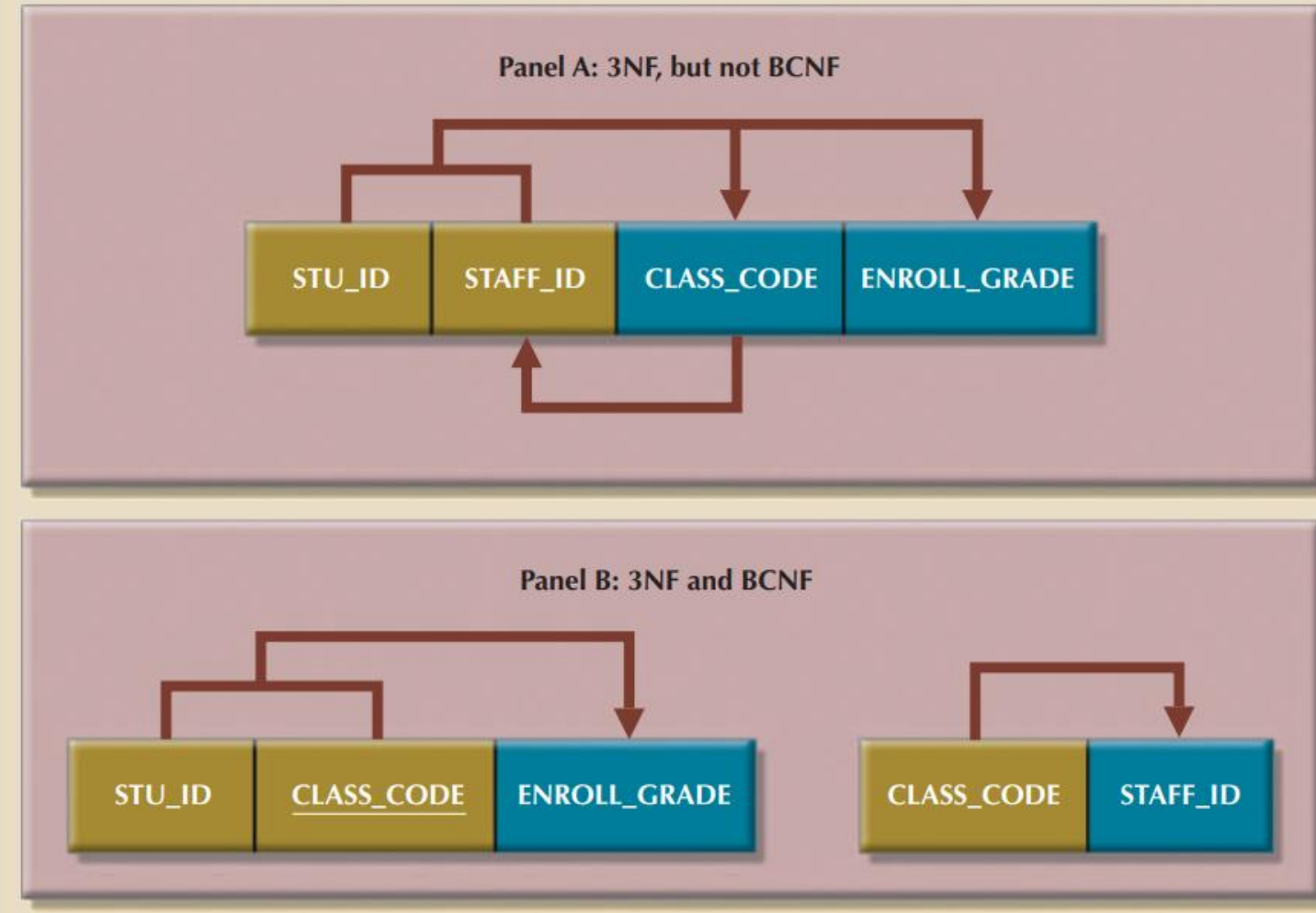
STU_ID	STAFF_ID	CLASS_CODE	ENROLL_GRADE
125	25	21334	A
125	20	32456	C
135	20	28458	B
144	25	27563	C
144	20	32456	B

- Phân rã thành 2 bảng:

STU_ID	STAFF_ID	ENROLL_GRADE
125	25	A
125	20	C
135	20	B
144	25	C
144	20	B

CLASS_CODE	STAFF_ID
21334	25
32456	20
28458	20
27563	25

FIGURE 6.10 ANOTHER BCNF DECOMPOSITION





- Bảng đạt chuẩn 4NF khi thỏa mãn cả 2 điều kiện sau:
  - Bảng đạt 3NF
  - Không có nhiều bộ phụ thuộc nhiều giá trị
- Bảng đạt 4NF nếu thỏa mãn 2 quy tắc:
  - Tất cả phụ thuộc vào khóa chính, độc lập với nhau
  - Không có hàng nào chứa 2 hoặc nhiều dữ kiện có nhiều giá trị về một thực thể

- Trong thực tế có gặp:
  - Thiết kế CSDL kém
  - Chuyển bảng tính → CSDL
- Nhận xét bảng 6.11:
  - 10123 tham gia tình nguyện RC và UW.
  - 10123 chỉ định tham gia dự án: 1,3,4
  - Không thể EMP\_NUM không thể kết hợp ORG\_CODE, ASSGN\_NUM là khóa được vì có những giá trị NULL
  - Với cách lưu như trên thì ORG\_CODE và ASSIGN\_NUM có thể có nhiều giá trị khác nhau.
    - EMP\_NUM → ORG\_CODE, ASSIGN\_NUM gọi là phụ thuộc nhiều giá trị
    - Thuộc tính khóa xác định nhiều giá trị của 2 thuộc tính và độc lập

FIGURE 6.11 TABLES WITH MULTIVALUED DEPENDENCIES

Table name: VOLUNTEER\_V1

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	UW	3
10123		4

Table name: VOLUNTEER\_V3

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	RC	3
10123	UW	4

Table name: VOLUNTEER\_V2

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	
10123	UW	
10123		1
10123		3
10123		4

Database nam

# Chuẩn 4NF

- Chuyển sang 4NF

FIGURE 6.11 TABLES WITH MULTIVALUED DEPENDENCIES

Database nam

Table name: VOLUNTEER_V1		
EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	UW	3
10123		4

Table name: VOLUNTEER_V2		
EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	
10123	UW	
10123		1
10123		3
10123		4

Table name: VOLUNTEER_V3		
EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	RC	3
10123	UW	4

FIGURE 6.12 A SET OF TABLES IN 4NF

Database nam

Table name: PROJECT		
PROJ_CODE	PROJ_NAME	PROJ_BUDGET
1	BeThere	1023245.00
2	BlueMoon	20198608.00
3	GreenThumb	3234456.00
4	GoFast	5674000.00
5	GoSlow	1002500.00

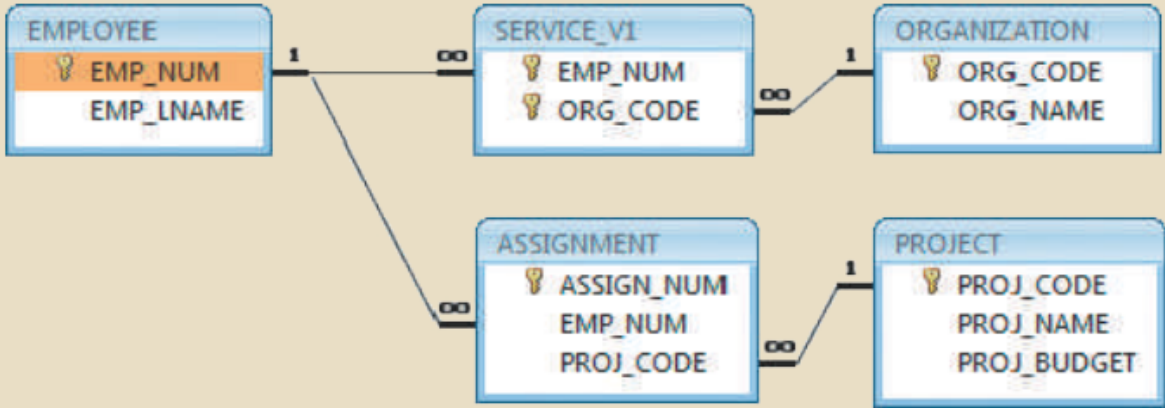
Table name: EMPLOYEE	
EMP_NUM	EMP_LNAME
10121	Rogers
10122	O'Leery
10123	Panera
10124	Johnson

Table name: ORGANIZATION	
ORG_CODE	ORG_NAME
RC	Red Cross
UW	United Way
WF	Wildlife Fund

Table name: ASSIGNMENT		
ASSIGN_NUM	EMP_NUM	PROJ_CODE
1	10123	1
2	10121	2
3	10123	3
4	10123	4
5	10121	1
6	10124	2
7	10124	3
8	10124	5

Table name: SERVICE_V1	
EMP_NUM	ORG_CODE
10123	RC
10123	UW
10123	WF

The relational diagram



- 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
  - Bạn có thể được yêu cầu thiết kế lại hoặc sửa đổi lại cơ sở dữ liệu hiện có.
- Do đó, bạn nên biết các nguyên tắc và quy trình thiết kế tốt cũng như các quy trình chuẩn hóa.

- 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



- 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) multicampus

- 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) multicampus

## 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) multicampus

## 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 lực 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

atomic attribute

atomicity

Boyce-Codd normal form  
(BCNF)

denormalization

dependency diagram

determinant

first normal form (1NF)

fourth normal form (4NF)

granularity

key attribute

nonkey attribute

nonprime attribute

normalization

partial dependency

prime attribute

repeating group

second normal form (2NF)

third normal form (3NF)

transitive dependency

unnormalized data



- Sinh viên tạo tệp:
  - STT\_HoVaTen\_Chapter6\_Normalization of Database Tables.docx
- Copy các câu hỏi trong tệp
  - Chapter6\_Normalization of Database Tables.docx
  - Trả lời các câu hỏi và làm bài tập phía dưới câu hỏi
- Nộp bài lên hệ thống



# THANK YOU

**multicampus**

Copyright by Multicampus Co., Ltd. All right reserved