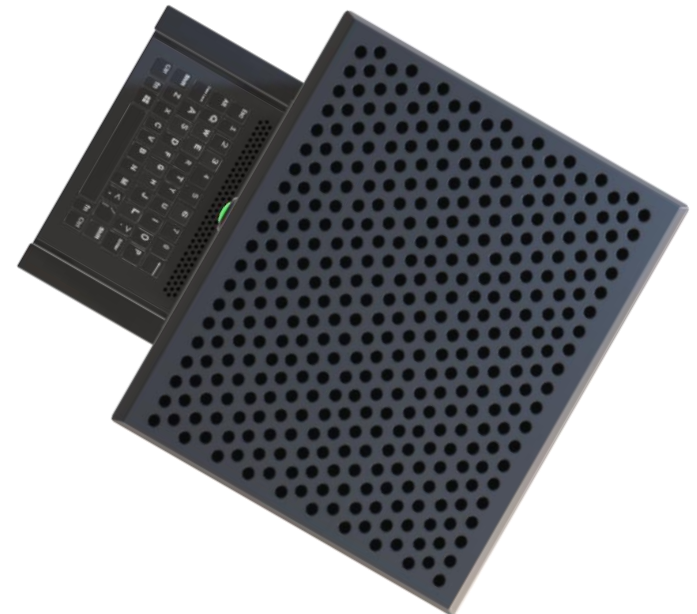




: DATABASE :

MID QUIZ



LESSON : 1

ฐานข้อมูล(DATABASE) คือ โครงสร้างข้อมูลแบบพิเศษใช้กักเก็บ จัดการ และเรียกใช้ข้อมูลระบบ

หลักการของฐานข้อมูล

**มีไว้สำหรับเก็บข้อมูลของ User โดยระบบจัดฐานข้อมูลจะ
สร้างการเก็บและการเรียกใช้ข้อมูล เพื่อให้เกิดความปลอดภัยของข้อมูล
และเพิ่มประสิทธิภาพของการใช้งาน**

ข้อมูล

**คือ ข้อมูลดิบที่ไม่ได้ถูก ประมวลผล หรือ
วิเคราะห์ใดๆ สารสนเทศ คือ ข้อมูลที่ถูก ประมวลผล หรือ
วิเคราะห์ข้อมูลแล้ว**

**ความสำคัญของการออกแบบฐานข้อมูล
คือ การใช้ประโยชน์ของมัน**

ทำไมจึงไม่ใช้ระบบ Folder หรือ แฟ้มข้อมูล

**การประมวลผลข้อมูลจะต้องมีโปรแกรมเรียกใช้เพื่อดึงข้อมูล การเขียนโปรแกรมจะ
เป็นในรูป Specific หรือเฉพาะกิจจึงเรียกใช้กันที่ไม่ได้ อีกทั้งยังสับสนและไม่ปลอดภัยมาก
เท่าที่ควร**

Redundancy

**คือ ความ "ซ้ำซ้อน" ของข้อมูล ทำให้เกิดกระบวนการ
Abnormally ทั้งสาม คือ Insert Update และ Delete**

ปัจจัยหาของฐานข้อมูล

คือ Hardware Software People Procedure และ Data

ข้อเสียของระบบ DATABASE

คือ มีค่าใช้จ่าย ซ้ำซ้อน ต้องการการบำรุงรักษา และ อาจมี License ตามมา

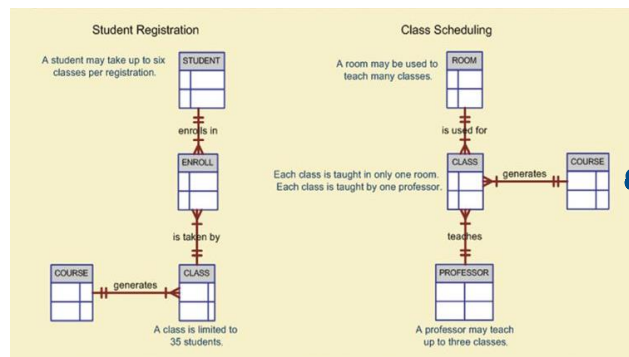
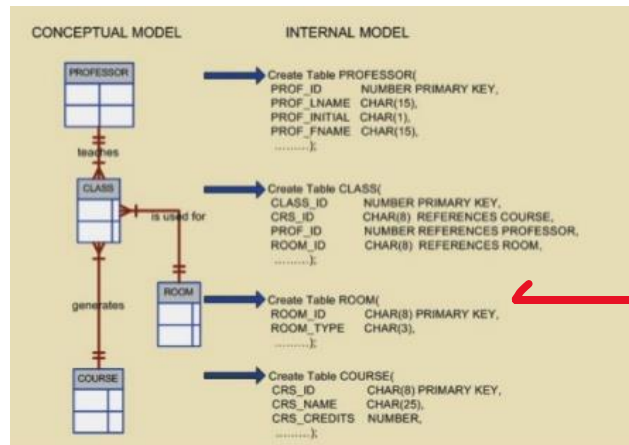
หน้าที่เช่น จัดพจนานุกรม เก็บข้อมูล แปลงข้อมูล จัดฐานข้อมูลเป็นต้น



พื้นฐานของระบบจำลองข้อมูล ประกอบด้วย
Entity(วัตถุ)
Attributes(คุณลักษณะของวัตถุ)
Relation(ความสัมพันธ์)
Condition(เงื่อนไข)

LESSON : 2

ปัจจุบัน แบบจำลองข้อมูลเชิงสัมพันธ์ จะเป็นที่นิยมในการใช้



กฎเกณฑ์ทางธุรกิจ
คือเงื่อนไขทางธุรกิจ เพื่อเป็นกฎเกณฑ์ในการเข้าถึงข้อมูลและทำให้เป็นมาตรฐานเดียวกันของทั้งองค์กร

แบบจำลองฐานข้อมูลแบบเครือข่าย กับ แบบจำลองแบบฐานข้อมูล
ต่างกันที่ แบบ เครือข่ายมีความยืดหยุ่นกว่า และมีความสัมพันธ์แบบ Owner & Member (ฟลอก)

ในการออกแบบสถาปัตยกรรมของข้อมูล จะมีทั้งหมด 4 ระดับ คือ ภายนอก แนวคิด ภายใน และ กายภาพ โดย

ระดับภายนอก
ระดับแนวคิด
ระดับภายใน
ระดับกายภาพ

จะเป็นมุมมองส่วนใดส่วนหนึ่ง
จะเป็นภาพรวมทั้งหมด การทำงานในระดับนี้ จะไม่ขึ้นกับ Hardware หรือ Software ใดๆ
จะเริ่มมองไม่เห็นในรูปแบบที่เป็นกายภาพ (รูปแบบที่จินตนาการลำบาก) ขึ้นอยู่กับ Software
Hardware : Tape, Disk ทั้งนี้จะขึ้นอยู่กับ Software ด้วย

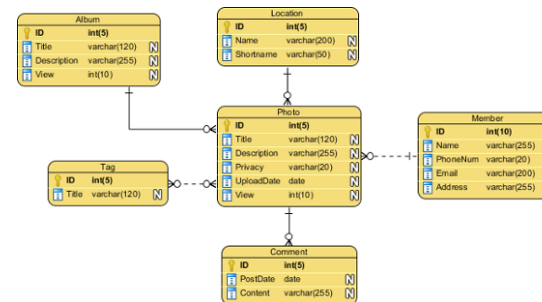


TABLE 2.4 Levels of Data Abstraction

MODEL	DEGREE OF ABSTRACTION	FOCUS	INDEPENDENT OF
External	High	End-user views	Hardware and software
Conceptual		Global view of data (database model independent)	Hardware and software
Internal	Low	Specific database model	Hardware
Physical		Storage and access methods	Neither hardware nor software

Composite Key(คีย์ประกอบ) : Key ที่มี Attribute มากกว่า 1
Key Attribute : Attribute ใดๆ ของ Key
Super Key : Key ที่บอกถึง เอกลักษณ์
Candidate Key(คีย์คู่แข่ง) : Super key ที่เล็กที่สุด ไม่มี Attribute อยู่
Primary Key : Key บอกเอกลักษณ์ของระเบียบ
Secondary Key(คีย์รอง) : Key สำหรับดึงข้อมูล
Foreign Key(คีย์นอก) : Primary Key ของอีกตารางหนึ่ง
NuLL : ว่างไม่มีข้อมูล

LESSON : 3

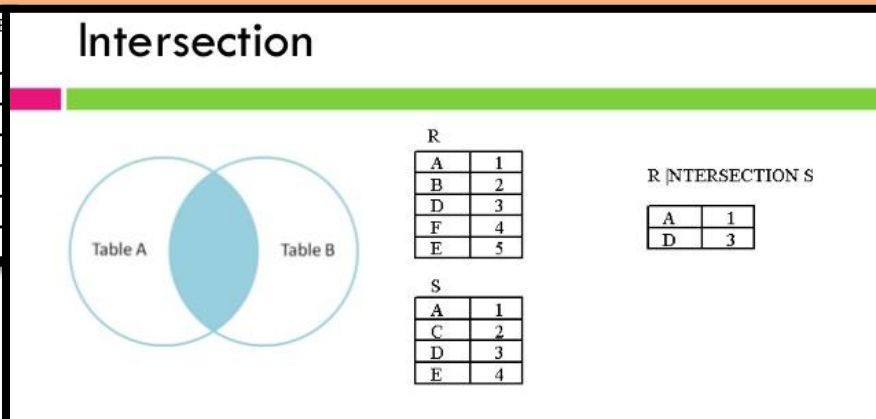
การควบคุม Redundancy
 คือ ทำให้ฐานข้อมูลเชิงสัมพันธ์ทำงาน
 ได้ผล โดยใช้ Attribute รวม ในการเชื่อมตาราง ทำ
 ให้ค่าที่โผล่มาไม่จัดเป็น Redundancy ซึ่งเกิดจาก
 การ ที่มีข้อมูลที่ มี Attribute ไม่จำเป็นซ้ำซ้อนกัน
 อยู่

การบูรณาภาพของการอ้างอิง
 คือ การที่มี คีย์นอกที่มีค่า ของข้อมูลค่านั้นๆ
 สามารถอ้างอิงไปที่ระเบียบอื่นได้

A B A UNION ALL B

1	3	1
2	4	2
3	5	3
		4
		5

การคุณให้เปิดหนังสือ
หน้า 41



Yield เหมือนเป็น Select ในรูปแบบ Where ซึ่ง จะต้อง
 ระบุว่า เอาแบบไหน และ And Or อย่างไร

DIVIDE

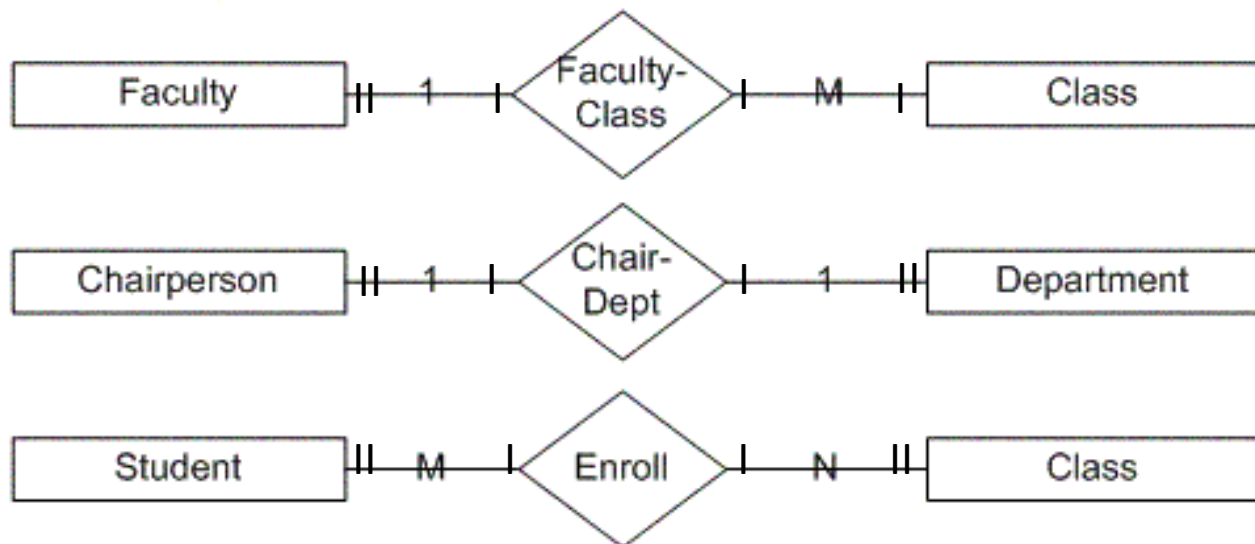
CODE	LOC
A	5
A	9
A	4
B	5
B	3
C	6
D	7
D	8
E	8

CODE A B

yields

LOC
5

Theta Join คือ การรวมกันแบบ เทียบค่าแบบฟังก์
 เครื่องหมาย >, <, =, >=, <=, !=
Equi Join คือ การรวมกันแบบ = เท่านั้น
Natural Join คือ Join แบบ Equi แต่จะติดร่างแห
 ของตารางทั้งตารางจะคล้ายกับการคูณ
Outer Join คือ Join แบบไม่ต้องใช้เงื่อนไข



Method 1: One: 1
Many M,N

||
|

คือ
คือ

1
Many

Entity
ประกอบ

ตัวอย่าง RELATION		
1 คณะ	มีได้หลาย	ห้องเรียน
1 คน	มีได้แค่	1 สาขา
นักเรียน	มีได้หลาย	ห้องเรียน
ห้องเรียน	มีได้หลาย	นักเรียน

Entity ประกอบ

คือ Entity ที่เกิดขึ้นเมื่อมี M:N ทำให้เกิดการแปลงความสัมพันธ์แบบ 1:M เพื่อรับ Primary Key จาก Entity

ระบบจำลอง ER

คือ ระบบนิยมใช้ในการออกแบบโครงสร้างฐานข้อมูลเชิงแนวความคิด เพราะง่ายต่อการทำความเข้าใจ และไม่ขึ้นกับ Hardware / Software

ประกอบด้วย : Entity -> วัตถุ Object
 : Attribute -> คุณลักษณะของ Entity
 : Relationship -> ความสัมพันธ์แบ่งได้เป็น 1:1 / 1:M / M:N

Degree ความสัมพันธ์
 UNARY(1 Self-Called)
 BINARY(2 Entity)
 TERNARY(3 Entity)

Derived Attribute

เป็น Attributes ที่มาจากการคำนวณ Attribute อื่นๆ ทำให้ประหยัดในการจัดพื้นที่ แต่ จะทำให้ CPU ทำงานหนักเพราะต้องคำนวณทุกค่า

Cardinality

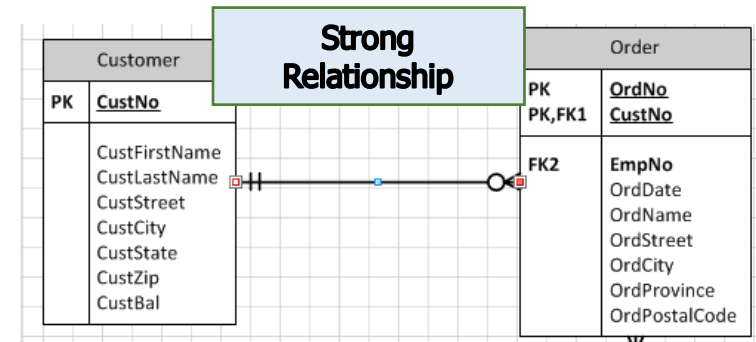
คือ ภาวะเชื่อมต่อที่แสดงถึงจำนวนที่น้อยที่สุดและมากที่สุดของ Entity instance กับ Entity instance อีกอันหนึ่ง

ความสัมพันธ์แบ่งได้ สองแบบ

คือ
 อ่อนแอ : คีย์หลักของ Entity ไม่ได้มาจาก Entity พ่อแม่ ทำให้มีความอิสระ
 แข็งแรง : คีย์หลักของ Entity ประกอบด้วยคีย์หลักของ Entity พ่อแม่

การมีส่วนร่วมของความสัมพันธ์

จำเป็น : Entity instance ต้องสัมพันธ์กับอีก Entity instance หนึ่ง
 ไม่จำเป็น : ตรงข้าม ^



ตัวอย่างความสัมพันธ์แบบ มีก็ได้หรือไม่มีก็ได้

One order record must be associated with a maximum of one Customer

A customer can have many orders

An order record must have a valid customer and it must NOT be Null

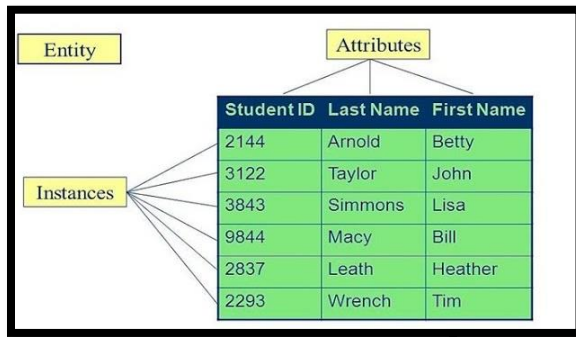
A customer does not need to have an order to be a customer

ต้องมี
หนึ่งคนเท่านั้น

Customer มี ได้หลาย Order

บอกว่าต้องไม่ NULL

บ่งบอกว่า Customer ไม่ต้องมี Order ก็ได้



Total Completeness

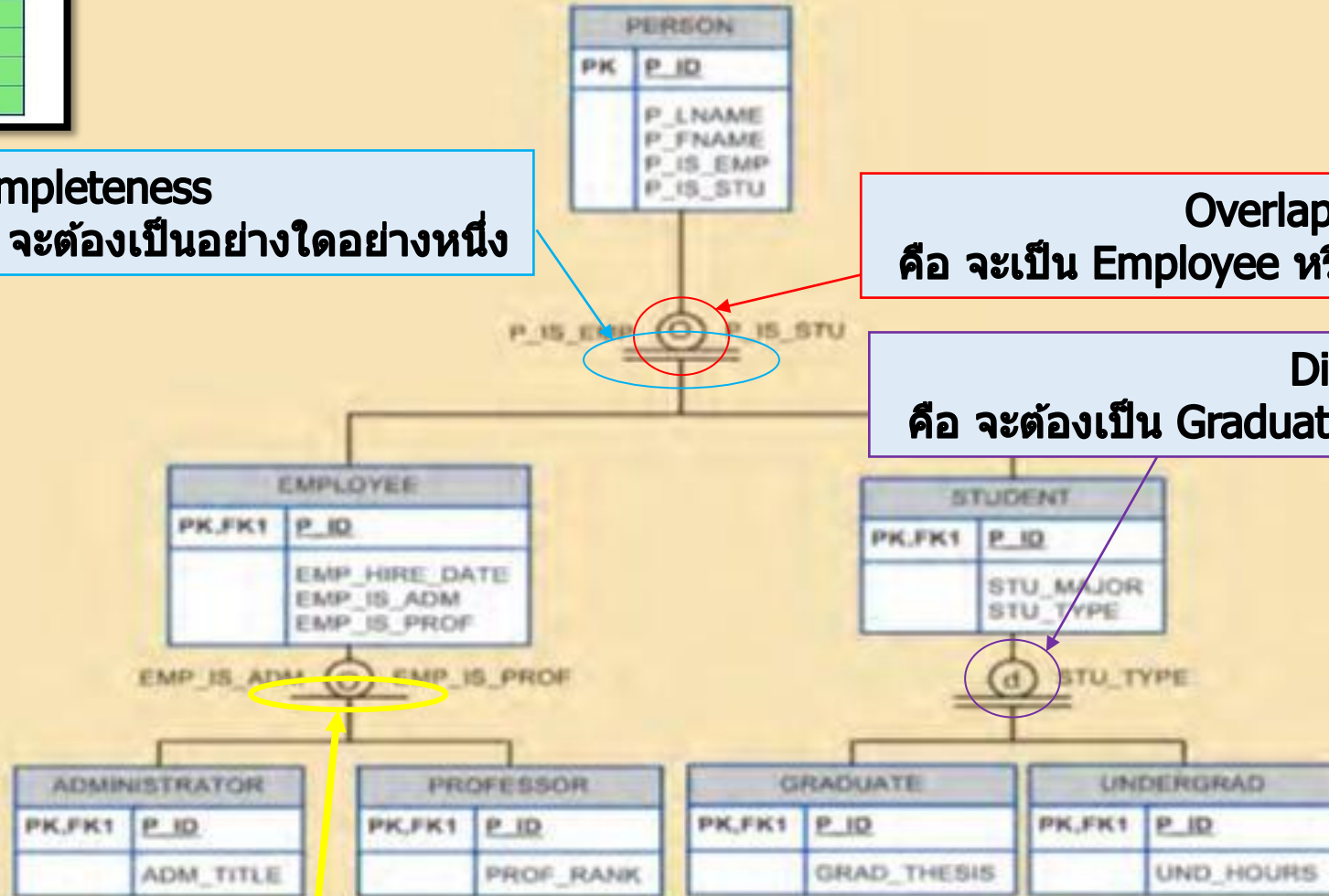
คือ อินสแตนซ์ของ Entity จะต้องเป็นอย่างใดอย่างหนึ่ง

Overlap

คือ จะเป็น Employee หรือ Student ก็ได้

Disjoint

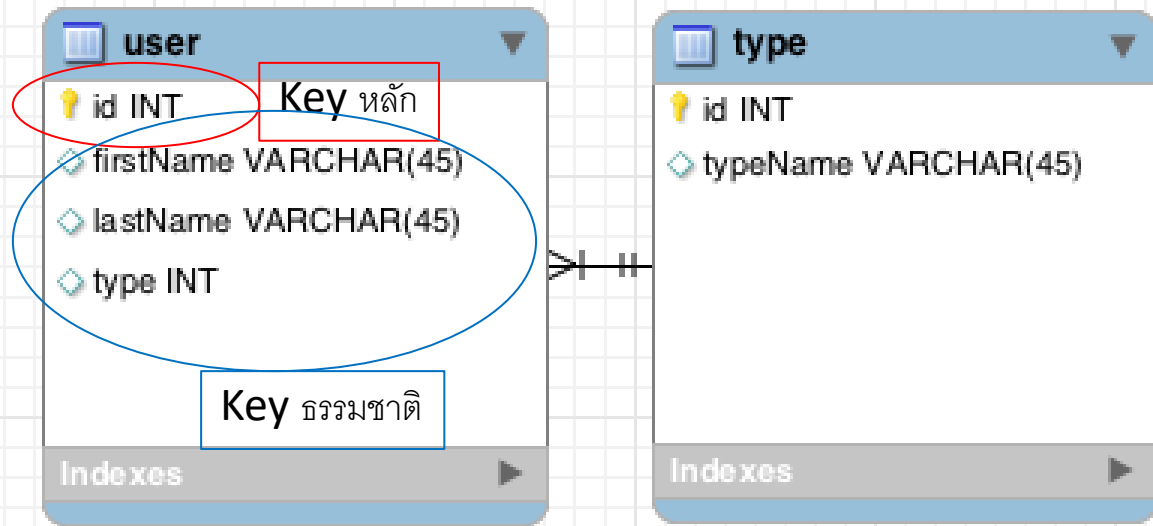
คือ จะต้องเป็น Graduate หรือ Undergrade เท่านั้น



Partial Completeness

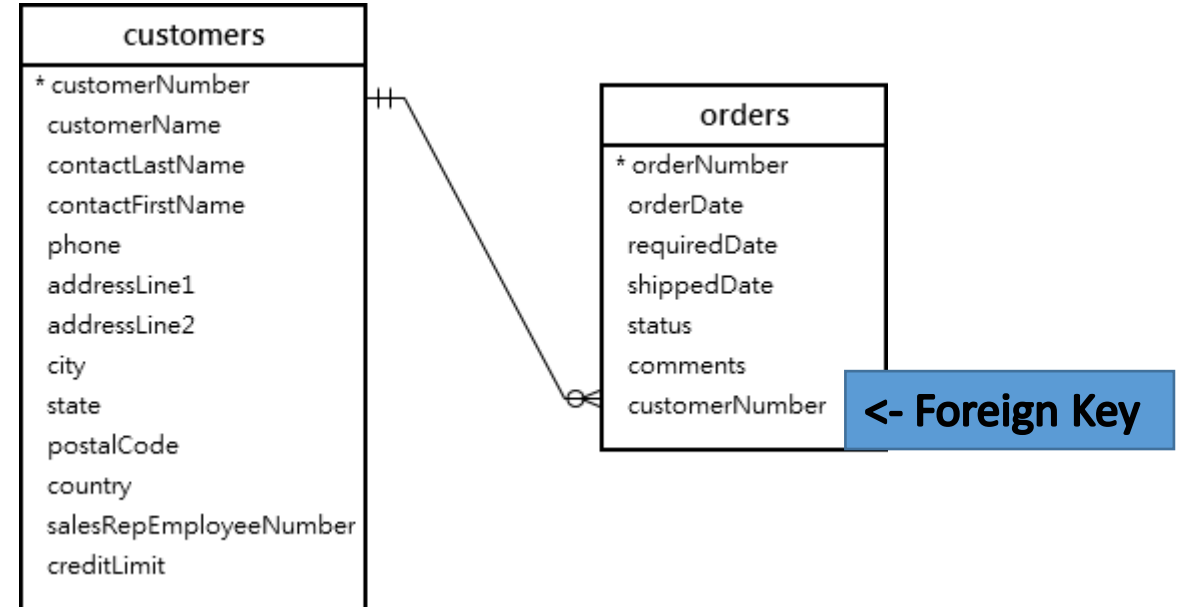
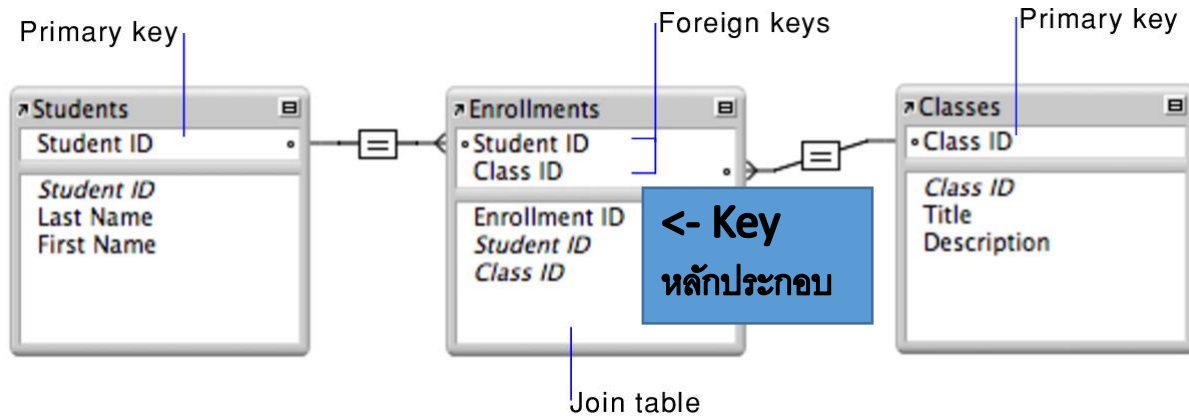
คือ อินสแตนซ์ของ Entity ไม่จำเป็นต้องเป็นอย่างใดอย่างหนึ่ง

overlapping subtypes



Key ธรรมชาติ : Natural Key ก็ key ที่เราได้ยินแล้วก็รู้ว่าเป็นอะไรนั่นแหละ

Key หลัก : Key ที่เป็นพวกรหัส ที่ได้ยินก็ต้อง สงสัยเปิดสมุด



รหัสพนักงาน EmpID	ชื่อ FirstName	นามสกุล LastName	รหัสแผนก DepID	ตำแหน่ง Position
E991005	บุญโชค	เอื้ออำนวย	D1	ผู้จัดการแผนก
E991102	ปรีชา	เปรมปรีดิ์	D2	รองผู้จัดการแผนก
E010109	ธีระพงษ์	ทองมาก	D2	พนักงาน
E020401	บุญญา	นิคม	D3	พนักงาน
E030202	กัญญา	ทรัพย์สมบัติ	D1	พนักงาน
E031101	จักรินส์	เปรมปรีดิ์	D3	ผู้จัดการแผนก

Key ทดแทนกรณีให้อะไรเป็น Key ไม่ได้เลย ^

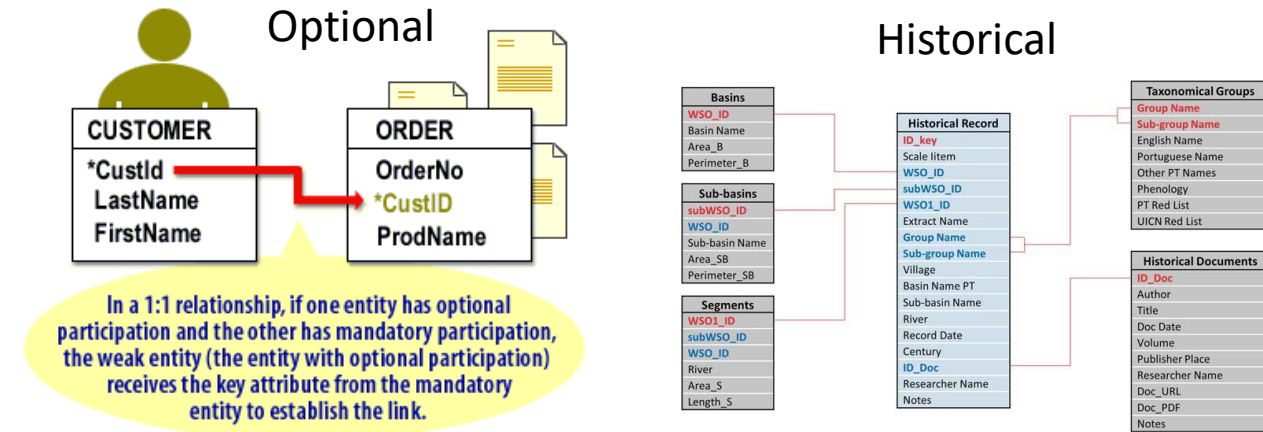
การออกแบบ

กรณี 1 : ดำเนินความสัมพันธ์ การกำหนด Relationship และ มีการใช้ Key แบบ Optional เพื่อลดการเกิด NULL

กรณี 2 : การบำรุงรักษาแบบข้อมูลผันตามเวลา

กรณี 3 : ปัญหาการติดกับการออกแบบ Fan trap

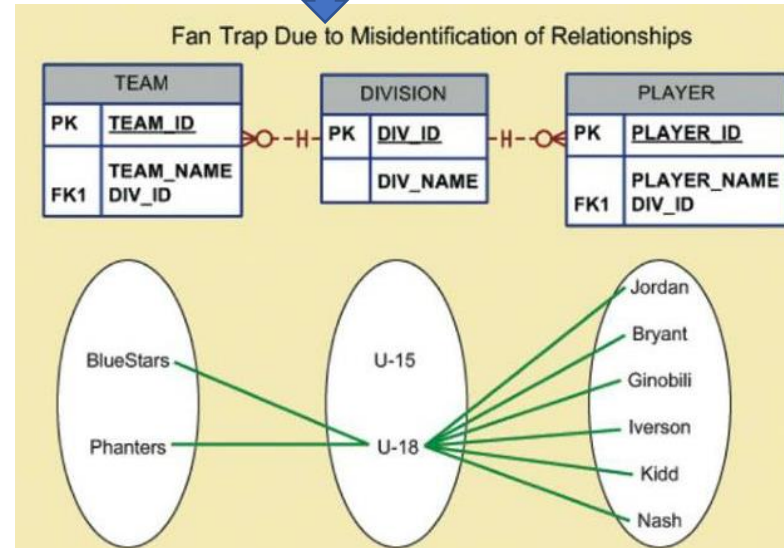
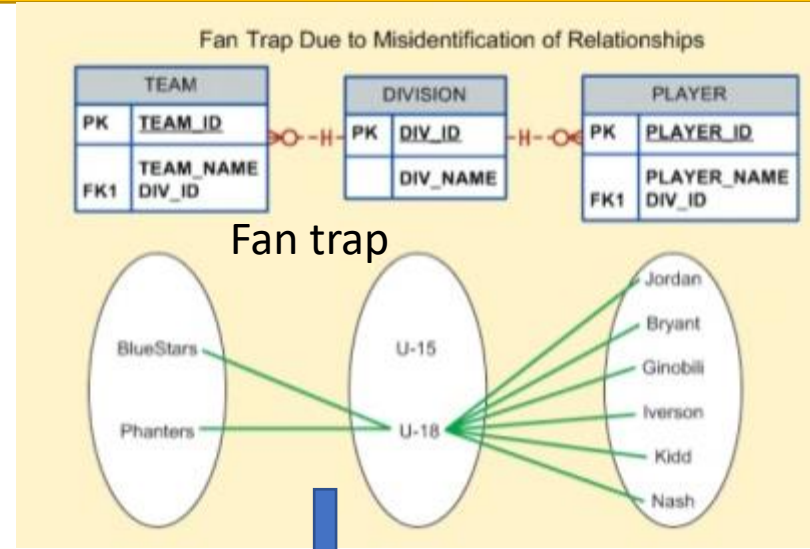
กรณี 4 : Redundancy อาจเกิดขึ้นถ้ามีการเก็บฐานข้อมูลไว้หลายๆที่



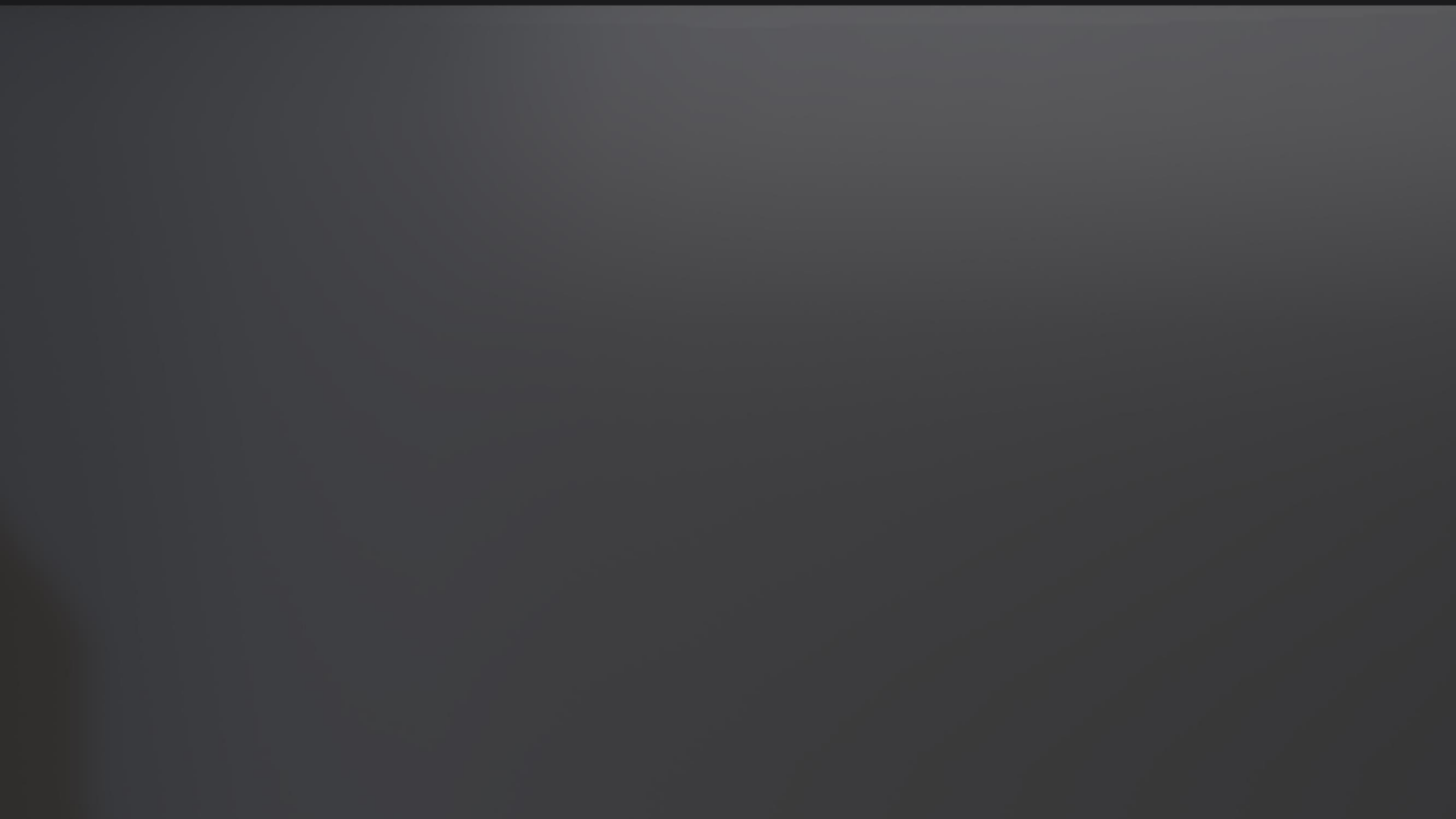
Student_ID	Name	Contact	College	Course	Rank
100	Himanshu	7300934851	GEU	Btech	1
101	Ankit	7900734858	GEU	Btech	1
102	Aysuh	7300936759	GEU	Btech	1
103	Ravi	7300901556	GEU	Btech	1

All places should be updated

Redundancy







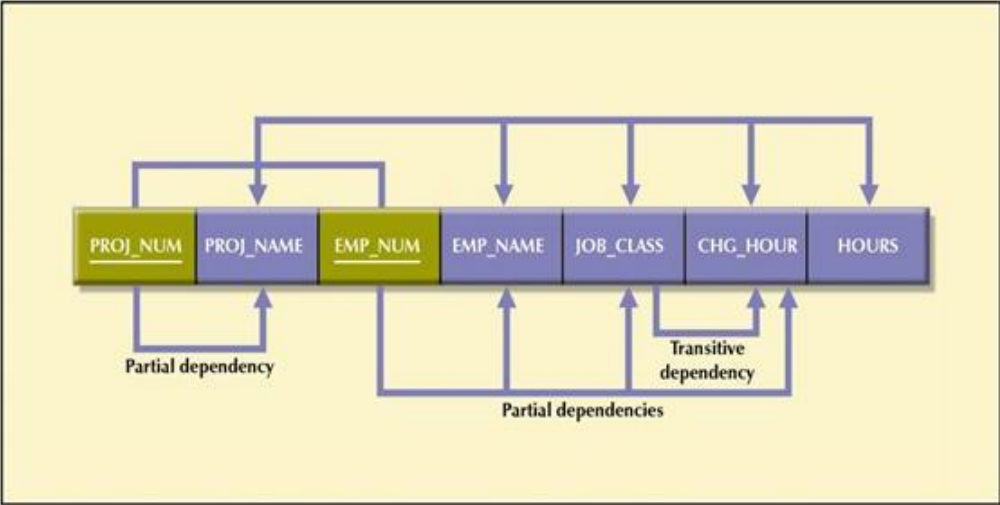
Normalization
คือ การทำให้โครงสร้างตารางข้อมูลมีความถูกต้องและลดความซ้ำซ้อน

Denormalization
คือ การทำย้อนกลับของการทำให้โครงสร้างตารางข้อมูลมีความถูกต้องและลดความซ้ำซ้อน

Form of Normalization	Explanation
1 NF	เป็นรูปตาราง ไม่มีกลุ่มซ้ำ มีการระบุคีย์หลัก
2 NF	เป็น 1NF ที่ไม่มี Partial Dependent
3 NF	เป็น 2NF ที่ไม่มี Transitive Dependent
Boyce-Codd (BCNF)	Determinant ทุกตัวเป็น Key คู่แข่ง
4 NF	3 NF ที่ไม่มี Attribute หลายค่า

Partial Dependent
 $(A,B) \rightarrow (C,D)$ และ (B,C) ดังนั้น $B \rightarrow C$ คือ Partial D.

Transitive Dependent
 $X \rightarrow Y$ และ $Y \rightarrow Z$ ดังนั้น $X \rightarrow Z$ ด้วย



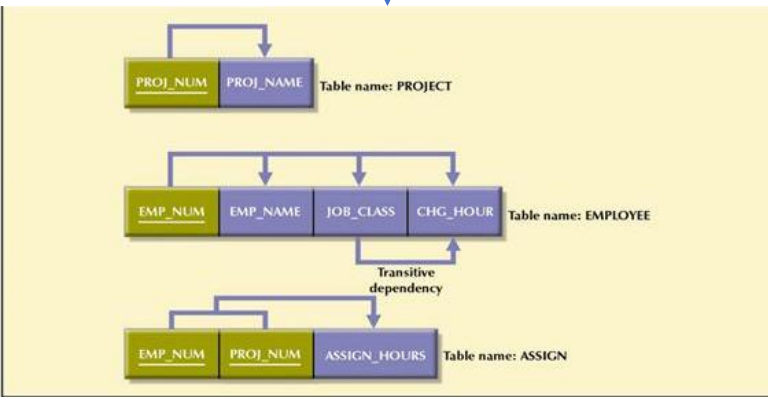
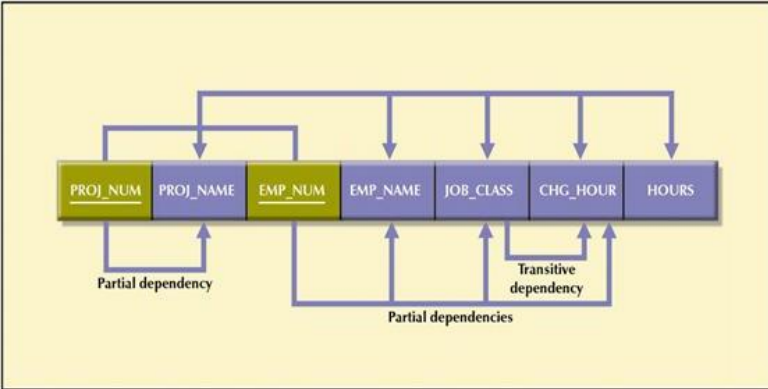
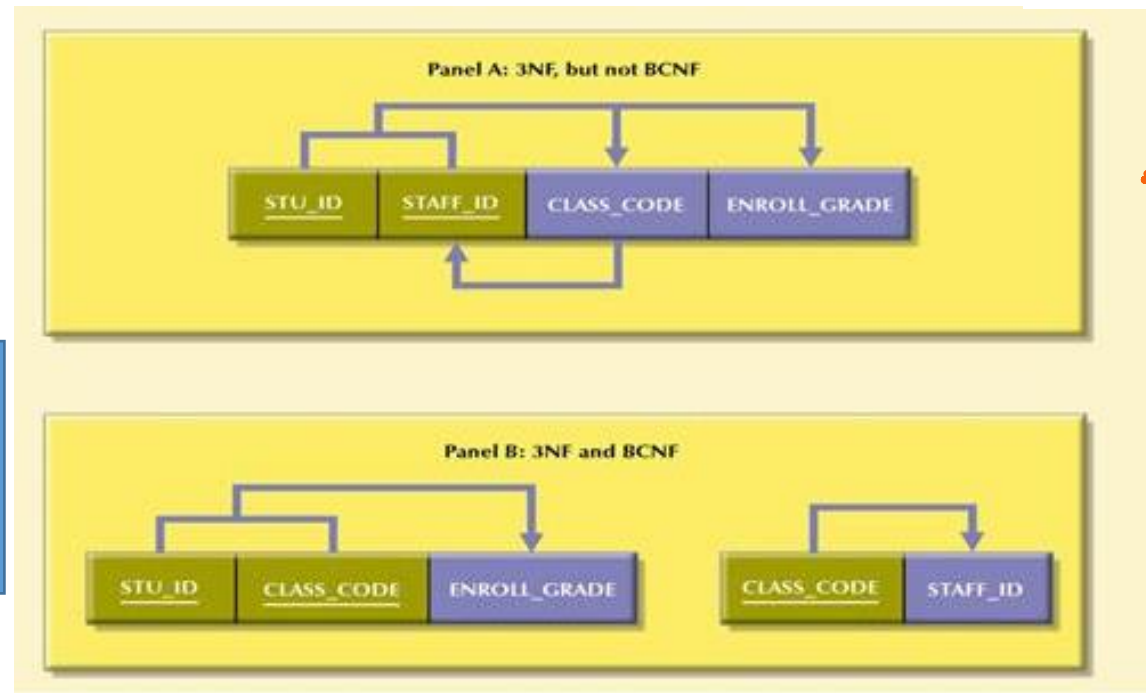
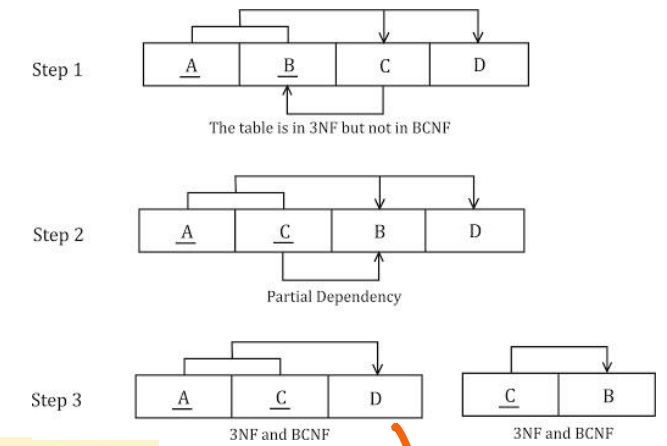
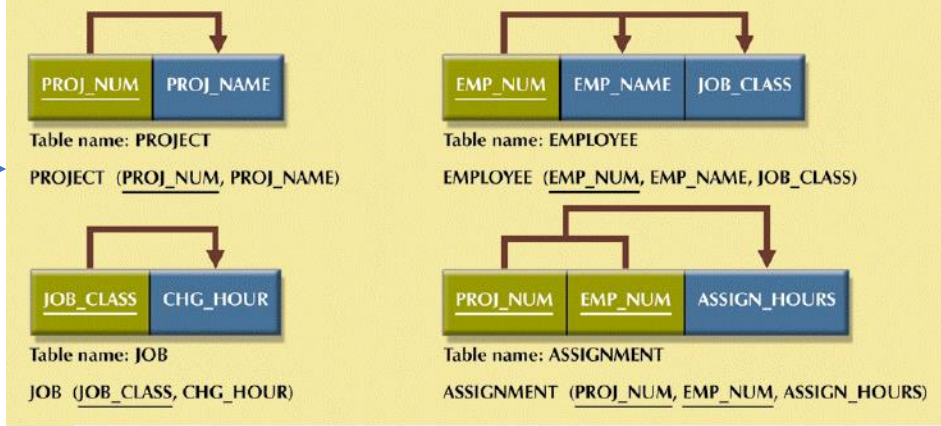


FIGURE 5.5 Third normal form (3NF) conversion results



4NF จะเหมือน 3NF
 ต่างที่
 ไม่มี Dependent กับ
 Attribute ที่มีค่าหลายตัว