

บทที่ 5

นอร์มัลไลเซชัน NORMALIZATION

หัวข้อการบรรยาย

2

- ความหมายและจุดประสงค์ของการนอร์มัลไลเซชัน
- ฟังก์ชันการขึ้นต่อกัน (Function Dependencies)
- กระบวนการนอร์มัลไลเซชัน

ความหมายและจุดประสงค์ของการนอร์มัลไลเซชัน

3

- นอร์มัลไลเซชัน เป็นทฤษฎีที่ผู้ออกแบบฐานข้อมูลจะต้องนำมาใช้ในการแปลงข้อมูลที่อยู่ในรูปแบบที่ซับซ้อน ให้อยู่ในรูปแบบที่ง่ายต่อการนำไปใช้งานและก่อให้เกิดปัญหาน้อยที่สุด

- ในบทนี้จะสอนกระบวนการนอร์มัลไลเซชัน ใน 3 ระดับด้วยกันคือ
 - นอร์มัลไลเซชันระดับที่ 1 หรือเรียกว่า 1NF
 - นอร์มัลไลเซชันระดับที่ 2 หรือเรียกว่า 2NF
 - นอร์มัลไลเซชันระดับที่ 3 หรือเรียกว่า 3NF

วิเคราะห์ความต้องการของผู้ใช้

E-R Diagram

รีเลชันที่มีรูปแบบไม่เป็นบรรทัดฐาน
(Unnormalized relation)

1 NF

2 NF

3 NF

บอยด์ คอตต์

4 NF

กระบวนการปรับ
บรรทัดฐาน
(Normalization)

รีเลชันที่มีเป็นบรรทัดฐาน
(Normalized relation)

วัตถุประสงค์ของนอร์มัลไลซ์ (Normalization)

5

- ❑ ลดความซ้ำซ้อนของข้อมูล เมื่อลดความซ้ำซ้อนก็ทำให้ลดเนื้อหาที่ใช้ในการจัดเก็บข้อมูล
- ❑ ลดปัญหาความไม่ถูกต้องของข้อมูล เมื่อข้อมูลไม่เกิดความซ้ำซ้อนทำให้การปรับปรุงข้อมูลสามารถทำได้จากแหล่งข้อมูลเพียงแหล่งเดียว
- ❑ ลดความผิดพลาดที่อาจเกิดจากการปรับปรุงข้อมูล (update anomalies) ซึ่งประกอบด้วย

ความซ้ำซ้อนและข้อผิดพลาดจากการปรับปรุงข้อมูล

6

- แนวคิดหลักอันสำคัญของการออกแบบฐานข้อมูลเชิงสัมพันธ์ คือ การออกแบบให้มีการเก็บข้อมูลซ้ำซ้อนน้อยที่สุด
 - เพื่อประหยัดเนื้อที่ในการเก็บข้อมูลและลดปัญหาที่จะเกิดดังตัวอย่างต่อไปนี้

ตัวอย่างการออกแบบฐานข้อมูลที่ดี

Employee (พนักงาน)

รหัสพนักงาน	ชื่อ-สกุล	ตำแหน่ง	เงินเดือน	รหัสสาขา
SG21	ชูชาติ สุขศรี	ผู้จัดการ	30000	B005
SG37	ศิริ ดวงเด่น	ผู้ช่วย	20000	B003
SG14	ดวงใจ มีสุข	เลขานุการ	20000	B003
SG09	อังฉรา เขียวแก้ว	ผู้จัดการ	30000	B007

Branch (สาขา)

รหัสสาขา	ที่อยู่
B005	เชียงใหม่
B003	กรุงเทพ
B007	พิษณุโลก

ตัวอย่างการออกแบบฐานข้อมูลที่จะมีปัญหาของความซ้ำซ้อนตามมา

Employee_Branch (รวมรายละเอียดของพนักงานไว้ด้วยกันกับรายละเอียดของสาขา)

รหัสพนักงาน	ชื่อ-สกุล	ตำแหน่ง	เงินเดือน	รหัสสาขา	ที่อยู่
SG21	ชูชาติ สุขศรี	ผู้จัดการ	30000	B005	เชียงใหม่
SG37	ศิริ ดวงเด่น	ผู้ช่วย	20000	B003	กรุงเทพ
SG14	ดวงใจ มีสุข	เลขานุการ	20000	B003	กรุงเทพ
SG09	อังฉรา เขียวแก้ว	ผู้จัดการ	30000	B007	พิษณุโลก

ตัวอย่างปัญหาความซ้ำซ้อนในข้อมูล รีเลชั่น EMPLOYEE_BRANCE

<u>รหัสพนักงาน</u>	<u>ชื่อ-สกุล</u>	<u>ตำแหน่ง</u>	<u>เงินเดือน</u>	<u>รหัสสาขา</u>	<u>ที่อยู่</u>
SG21	ชูชาติ สุขศรี	ผู้จัดการ	30000	B005	เชียงใหม่
SG37	ศิริ ดวงเด่น	ผู้ช่วย	20000	B003	กรุงเทพ
SG14	ดวงใจ มีสุข	เลขานุการ	20000	B003	กรุงเทพ
SG09	อัจฉรา เขียวแก้ว	ผู้จัดการ	30000	B007	พิษณุโลก

ความผิดพลาดจากการเพิ่ม

- ถ้าต้องการเพิ่มพนักงานใหม่ ที่อยู่สาขา B005
- จะต้องกรอก B005 และที่อยู่สาขา คือ เชียงใหม่ เพิ่มอีก

ตัวอย่างปัญหาความซ้ำซ้อนในข้อมูล รีเลชั่น EMPLOYEE_BRANCE

รหัสพนักงาน	ชื่อ-สกุล	ตำแหน่ง	เงินเดือน	รหัสสาขา	ที่อยู่
SG21	ชูชาติ สุขศรี	ผู้จัดการ	30000	B005	เชียงใหม่
SG37	ศิริ ดวงเด่น	ผู้ช่วย	20000	B003	กรุงเทพ
SG14	ดวงใจ มีสุข	เลขานุการ	20000	B003	กรุงเทพ
SG09	อัจฉรา เขียวแก้ว	ผู้จัดการ	30000	B007	พิษณุโลก

ความผิดพลาดจากการเพิ่ม

- ถ้าต้องการเพิ่มสาขา จะมีปัญหาคือ ตารางนี้มีทั้งข้อมูลพนักงานและข้อมูลสาขาอยู่รวมกัน
- หากจะเพิ่มเฉพาะ รหัสสาขา และ ที่อยู่ ก็ไม่ได้เพราะ รหัสพนักงาน จะมีค่าว่างไม่ได้เพราะเป็น Primary Key ของตาราง
- ดังนั้นจะบันทึกได้ก็ต่อเมื่อมีพนักงานแล้ว

ความผิดพลาดจากการลบข้อมูล

- ❑ ถ้าลบข้อมูลหนึ่งแล้วส่งผลกระทบต่อข้อมูลอื่น ที่ต้องถูกลบตาม
 - ❑ เช่น พนักงานรหัส SG21 ลาออก ก็ลบแถวนั้นออก
 - ❑ ข้อมูลสาขา B005 ก็จะหายไปด้วย

ข้อผิดพลาดจากการเปลี่ยนแปลง

- ❑ ในกรณีที่ต้องการเปลี่ยนแปลงข้อมูลบางตัวของสาขา
 - ❑ เช่น เปลี่ยนที่อยู่ของ B003 ก็ต้องเปลี่ยนหลายที่
 - ❑ ถ้าหากมีพนักงานสังกัดสาขานี้หลายที่ก็ต้องไปตามแก้ทุก ๆ ที่

ดังนั้นเราควรแยกตาราง Employee_Branch ออกเป็นสองตาราง คือ ตารางพนักงาน และตารางสาขา

พนักงาน

<u>รหัสพนักงาน</u>	<u>ชื่อ-สกุล</u>	<u>ตำแหน่ง</u>	<u>เงินเดือน</u>	<u>รหัสสาขา</u>
SG21	ชูชาติ สุขศรี	ผู้จัดการ	30000	B005
SG37	ศิริ ดวงเด่น	ผู้ช่วย	20000	B003
SG14	ดวงใจ มีสุข	เลขานุการ	20000	B003
SG09	อังฉรา เขียวแก้ว	ผู้จัดการ	30000	B007

สาขา

<u>รหัสสาขา</u>	<u>ที่อยู่</u>
B005	เชียงใหม่
B003	กรุงเทพ
B007	พิษณุโลก

ฟังก์ชันการขึ้นต่อกัน (Functional Dependency : FD)

12

- ถ้าให้ X และ Y เป็น Attribute ใน Relation ใดๆ แทนด้วย $R(X,Y)$ Attribute Y เป็น จะถูกเรียกว่ามีฟังก์ชันการขึ้นต่อกันกับแอททริบิวต์ X ก็ต่อเมื่อ แต่ละค่าที่ไม่ซ้ำกันของแอททริบิวต์ X มีข้อมูลของ Y ที่เกี่ยวข้องกับ X เพียง 1 ค่า
- เขียนแทนด้วย สัญลักษณ์ $X \rightarrow Y$

ตัวอย่าง

<u>EmployeeNo</u>	Name	Position
S01	ฉัตรชัย มีสมบัติ	Manager
S02	เอกชัย ใจดี	Manager Assistant
S03	มนิรัตน์ เจริญสุข	Manager
S04	ขวัญชัย ใจเพชร	Manager Assistant
S05	มานพ เกตุแก้ว	Staff
S06	ดวงกมล ทิพย์เทพ	Staff

EmployeeNo \longrightarrow Position

ตัวอย่าง A) employeeNo



Position

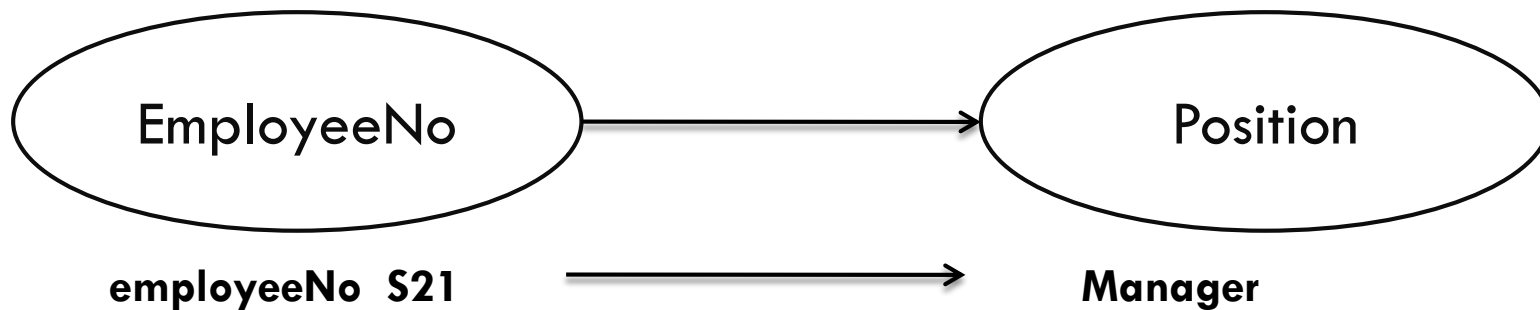
B) Position (not)



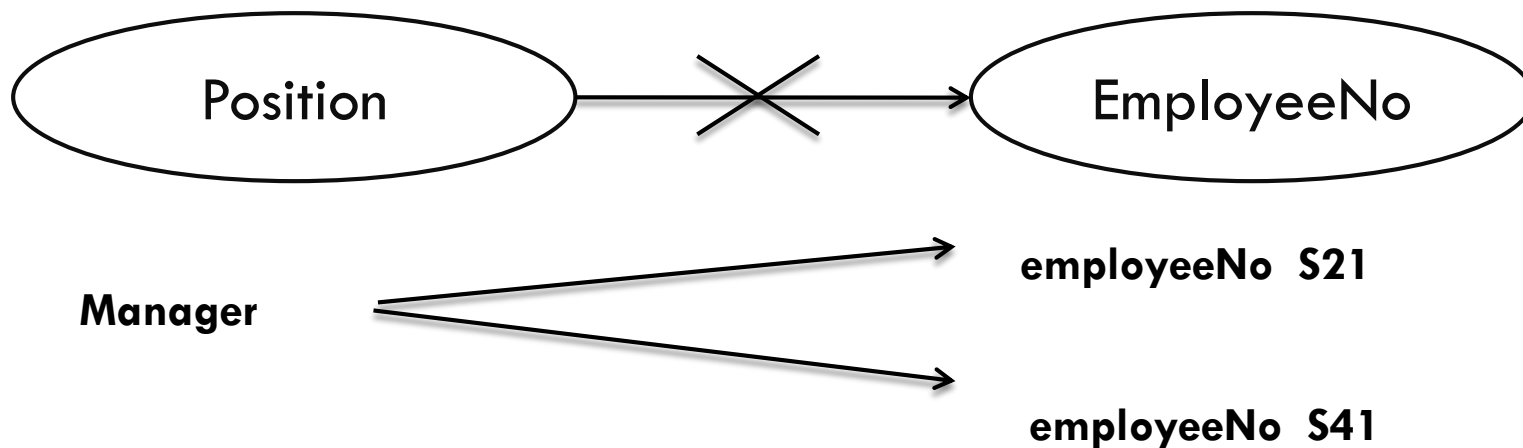
employeeNo

14

A



B



การขึ้นต่อกัน ค่า X จะต้องกำหนดค่า Y ได้ 1:1

ชนิดของฟังก์ชันการขึ้นต่อกัน (Functional Dependency :FD)

15

- 1) Complete dependencies การขึ้นต่อกันอย่างสมบูรณ์
- แอททริบิวต์ที่ไม่ใช่คีย์หลัก ขึ้นต่อ แอททริบิวต์หรือกลุ่มของแอททริบิวต์ที่เป็นคีย์หลัก
- ตัวอย่าง ตารางที่มีแอทริบิวต์ค่าเดียวทำหน้าที่เป็นคีย์หลัก คือ หมายเลขบัตรประชาชน

<u>หมายเลขบัตรประชาชน</u>	<u>ชื่อเจ้าของบัตร</u>
3440100634931	กนกวรรณ พ่วงพงษ์
3437283420343	ชาติชาย เตชะวงศ์
2938742039485	กิงกาญ เดชาทรัพย์

หมายเลขบัตรประชาชน → ชื่อเจ้าของบัตร

ชนิดของฟังก์ชันการขึ้นต่อกัน (Functional Dependency :FD)

16

- ตัวอย่าง ตารางที่มีแอทริบิวต์หลายตัวขึ้นกับคีย์หลักตัวเดียว

หมายเลขบัตรประชาชน	ชื่อเจ้าของบัตร	วันเกิด	วันที่ทำบัตร
3440100634931	กนกวรรณ พ่วงพงษ์	27/03/2520	28/04/2553
3437283420343	ชาติชาย เตชะวงศ์	23/06/2522	25/02/2553
2938742039485	กิงกาญ เตชาทรัพย์	21/04/2525	19/0125/52

หมายเลขบัตรประชาชน → ชื่อเจ้าของบัตร,วันเกิด,วันที่ทำบัตร

ชนิดของฟังก์ชันการขึ้นต่อกัน (Functional Dependency :FD)

17

- ตัวอย่าง ตารางที่มีแอทริบิวหลายตัวรวมกันเป็นคีย์หลัก คือ รหัสนักศึกษา รหัสวิชา

ตารางการลงทะเบียน

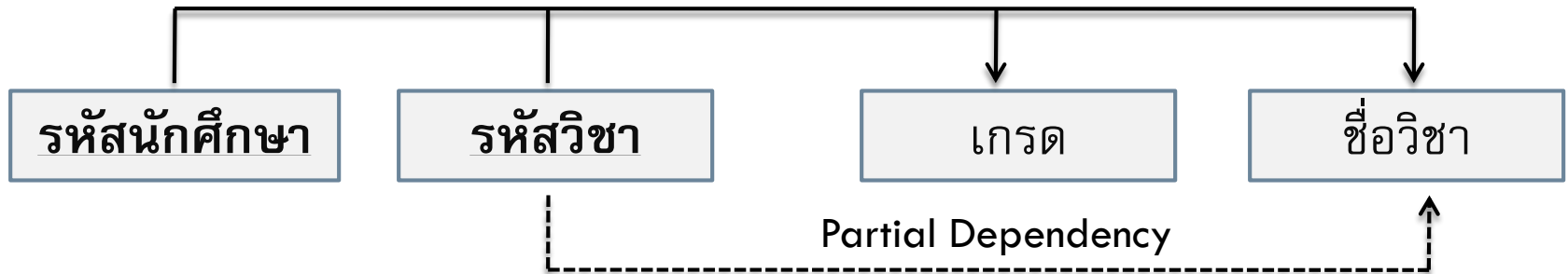
<u>รหัสนักศึกษา</u>	<u>รหัสวิชา</u>	เกรด
520014001	S001	A
520014001	S002	B
520014002	S001	C
520014002	S002	A

รหัสนักศึกษา,รหัสวิชา → เกรด

ชนิดของฟังก์ชันการขึ้นต่อกัน (Functional Dependency :FD)

18

- 2) Partial Dependency (การขึ้นต่อกันบางส่วน)
- เกิดขึ้นเมื่อกีย์หลักประกอบด้วยหลาย Attribute รวมกัน
- เมื่อแอตทริบิวต์บางส่วนของคีย์หลัก สามารถไประบุค่าแอตทริบิวต์ตัวอื่น ๆ ที่ไม่ใช่คีย์หลักของรีเลชันได้



รหัสนักศึกษา, รหัสวิชา → เกรด , ชื่อวิชา
รหัสวิชา → ชื่อวิชา

ตัวอย่างฟังก์ชันการขึ้นต่อกันแบบ Partial

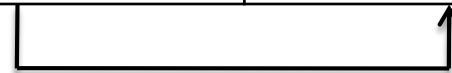
<u>รหัสนักศึกษา</u>	<u>รหัสวิชา</u>	<u>เกรด</u>	<u>ชื่อวิชา</u>
534267001	F01	A	การเขียนโปรแกรม
534267001	F02	B	การออกแบบฐานข้อมูล
534267002	F01	D	การเขียนโปรแกรม
534267002	F02	A	การออกแบบฐานข้อมูล
534267003	F01	A	การเขียนโปรแกรม
534267003	F02	C	การออกแบบฐานข้อมูล

ชนิดของฟังก์ชันการขึ้นต่อกัน (Functional Dependency : FD)

20

- **3 Transitive Dependency** เกิดขึ้นเมื่อ Attribute ที่ไม่ใช่ Primary Key ไปขึ้นอยู่กับ Attribute อื่นที่ไม่ใช่ Primary Key ในรีเลชันนั้น ๆ

เลขประจำตัว	ชื่อ สกุล	ที่อยู่	ตำแหน่ง	รถประจำตำแหน่ง
01	ฉัตรชัย มีสมบัติ	กรุงเทพ	ผู้จัดการ	BMW
02	เอกชัย ใจดี	นนทบุรี	ผู้ช่วยผู้จัดการ	Honda
03	มนีรัตน์ เจริญสุข	เชียงใหม่	ผู้จัดการ	BMW
04	ขวัญชัย ใจเพชร	ราชบุรี	ผู้ช่วยผู้จัดการ	Honda



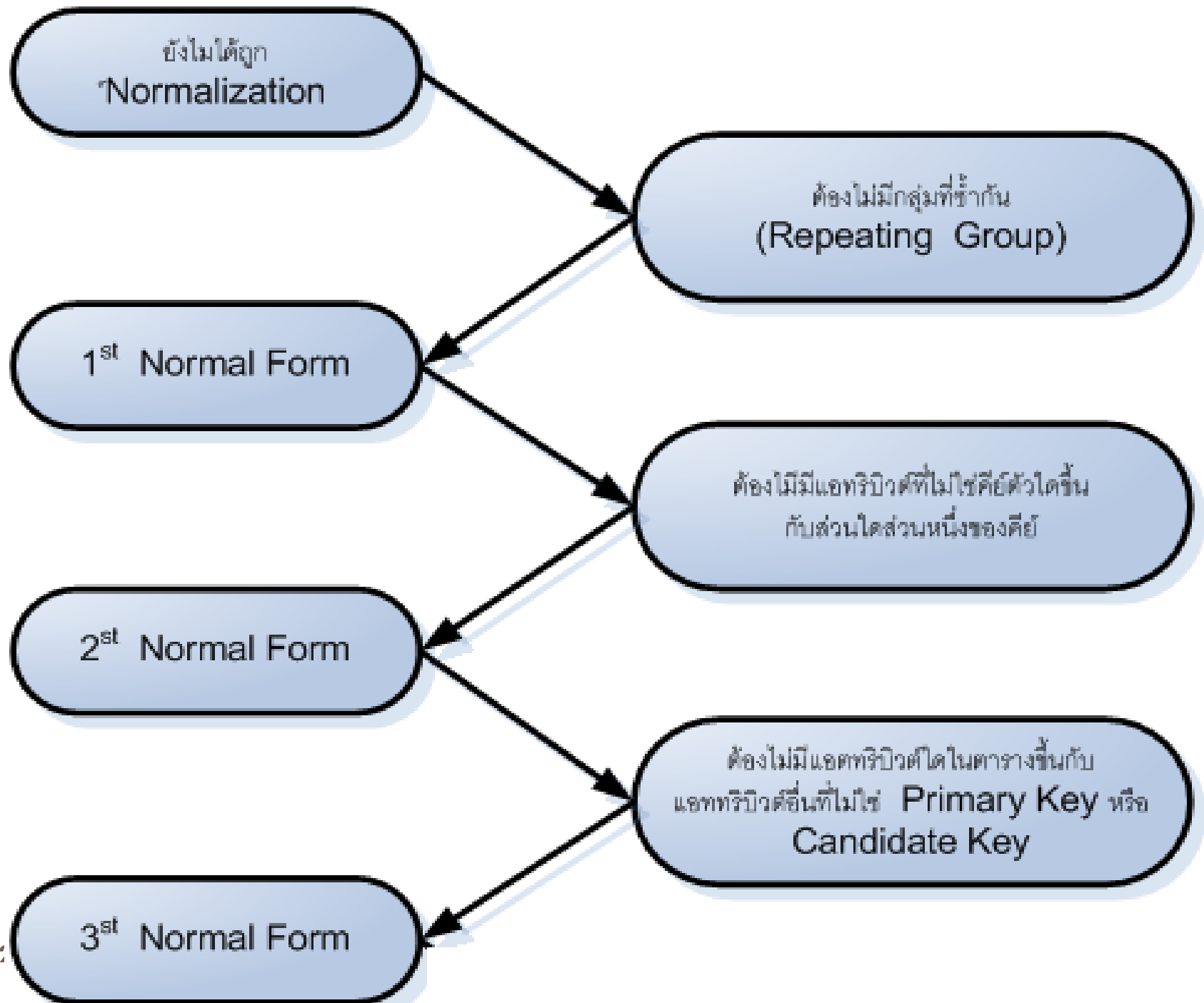
คำอธิบาย เลขประจำตัว เป็นคีย์หลัก (Primary Key) ของตาราง

เลขประจำตัว → ชื่อสกุล, ที่อยู่, ตำแหน่ง
ตำแหน่ง → รถประจำตำแหน่ง

นอร์มัลไลเซชัน (Normalization)

21

- Normalization คือ กระบวนการปรับปรุงโครงสร้างข้อมูลของฐานข้อมูลที่มีความซ้ำซ้อนให้อยู่ในรูปแบบที่เป็นบรรทัดฐาน (Normal Form)
- การนอร์มัลไลเซชันมีได้ถึง 5 ระดับ ในระดับที่ 3 ก็จัดว่าเพียงพอสำหรับการออกแบบฐานข้อมูลในปัจจุบัน
 - 1NF - กำจัด repeating group
(กำจัดกลุ่มของข้อมูลที่มีความซ้ำซ้อน)
 - 2NF - กำจัด partial dependency
(กำจัดการขึ้นต่อกันบางส่วน)
 - 3NF - กำจัด transitive dependency
(กำจัดการขึ้นต่อกันของแอตทริบิวต์ที่ไม่ใช่คีย์หลัก)



First Normal Form (1NF)

23

- ทุก Attribute ในแต่ละ record จะเป็น single value ไม่มี ค่าของกลุ่ม ข้อมูลที่ซ้ำกัน (Repeating Group)
- ข้อมูลทุกแถว (Tuple) ต้องมีค่าไม่ซ้ำกัน

ตารางที่มีลักษณะข้อมูลเป็น Repeating group

รหัสนักศึกษา	ชื่อ	นามสกุล	รหัสวิชาที่ลงทะเบียน
001	สมชาย	สมใจนึก	204-101 204-204 204-205
002	ธีรชาย	บุญมาศ	204-102 204-204

Repeating Group

ตารางที่มีลักษณะข้อมูลเป็น Repeating group

<u>รหัสนักศึกษา</u>	<u>ชื่อ</u>	<u>นามสกุล</u>	<u>รหัสวิชาที่ลงทะเบียน</u>
001	สมชาย	สมใจนึก	204-101 204-204 204-205
002	ธีรชาย	บุญมาศ	204-102 204-204

เราสามารถทำให้อยู่ในรูป 1NF ได้ดังนี้

<u>รหัสนักศึกษา</u>	<u>ชื่อ</u>	<u>นามสกุล</u>	<u>รหัสวิชาที่ลงทะเบียน</u>
001	สมชาย	สมใจนึก	204-100
001	สมชาย	สมใจนึก	204-204
001	สมชาย	สมใจนึก	204-125
002	ธีรชาย	บุญมาศ	204-102
002	ธีรชาย	บุญมาศ	204-204

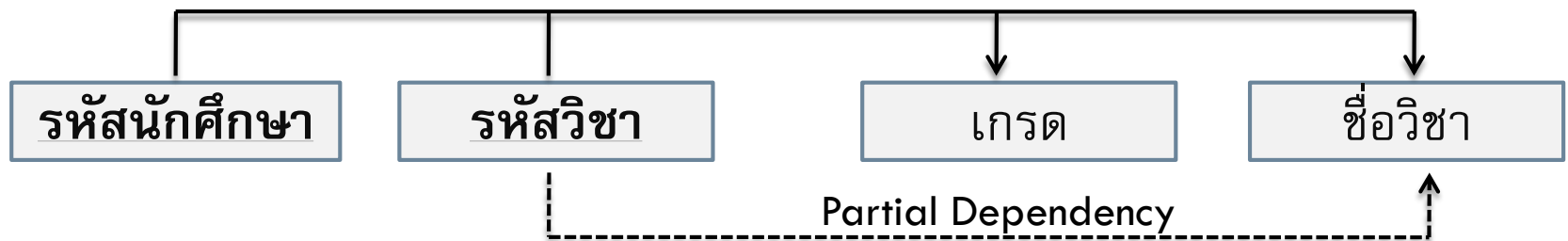
Second Normal Form (2NF)

25

- 1. ต้องเป็น First Normal Form (1NF) มาก่อน
- 2. ต้องไม่มี Partial Dependency (การขึ้นต่อกันบางส่วน)
- สรุปก็คือ นอร์มัลไลเซชันระดับที่ 2 (Second normal form : 2NF) เป็นการ ขจัดแอดตริบิวที่ **ไม่ขึ้นกับทั้งส่วนของคีย์หลัก** ออกไป เพื่อให้แอดตริบิวอื่นทั้งหมดขึ้นตรงกับส่วนที่เป็นคีย์หลักทั้งหมดเท่านั้น

ตัวอย่างตารางที่ Partial Dependency (การขึ้นต่อกันบางส่วน)

รหัสนักศึกษา	รหัสวิชา	เกรด	ชื่อวิชา
534267001	F01	A	การเขียนโปรแกรม
534267001	F02	B	การออกแบบฐานข้อมูล
534267002	F01	D	การเขียนโปรแกรม
534267002	F02	A	การออกแบบฐานข้อมูล
534267003	F01	A	การเขียนโปรแกรม
534267003	F02	C	การออกแบบฐานข้อมูล



รหัสนักศึกษา, รหัสวิชา → เกรด, ชื่อวิชา
 รหัสวิชา → ชื่อวิชา

Second Normal Form (2NF)

27

□ วิธีขจัดปัญหา

- 1) ต้องสร้างตารางเพิ่ม
- 2) นำคอลัมน์ที่มีปัญหาไปไว้ในตารางที่สร้างเพิ่ม
- 3) กำหนดคีย์หลักให้กับตารางที่สร้างใหม่
- 4) แอททริบิวต์ใดในตารางเดิม เมื่อนำไปไว้ในตารางใหม่ให้ตัดออกจากตารางเดิม ยกเว้น ส่วนของคีย์หลัก คงไว้ในตารางเดิม

ตารางผลการเรียน

รหัสนักศึกษา	รหัสวิชา	เกรด	ชื่อวิชา
534267001	F01	A	การเขียนโปรแกรม
534267001	F02	B	การออกแบบฐานข้อมูล
534267002	F01	D	การเขียนโปรแกรม
534267002	F02	A	การออกแบบฐานข้อมูล
534267003	F01	A	การเขียนโปรแกรม
534267003	F02	C	การออกแบบฐานข้อมูล

ตารางนี้เมื่อทำให้อยู่ในรูป 2 NF จะได้ 2 ตารางดังนี้

ตารางผลการเรียน

รหัสนักศึกษา	รหัสวิชา	เกรด
534267001	F01	A
534267001	F02	B
534267002	F01	D
534267002	F02	A
534267003	F01	A
534267003	F02	C

ตารางวิชา

รหัสวิชา	ชื่อวิชา
F01	การเขียนโปรแกรม
F02	การออกแบบฐานข้อมูล

ให้นักศึกษานอมนัลไลซ์ตารางนี้ให้อยู่ในรูปแบบ 2NF

ตารางผลการอบรม

<u>รหัสผู้เข้า อบรม</u>	<u>รหัสครอส อบรม</u>	<u>ชื่อผู้เข้าอบรม</u>	<u>ชื่อครอสอบรม</u>	<u>ผลการทดสอบ</u>
0001	TR01	นายเอ ใจดี	การซ่อมไฟฟ้า	ผ่าน
0001	TR05	นายเอ ใจดี	การซ่อมตู้เย็น	ผ่าน
0002	TR03	นางบี ใจกล้า	การทำอาหาร	ไม่ผ่าน
0002	TR09	นางบี ใจกล้า	การเลี้ยงเด็ก	ผ่าน
0003	TR01	นายรวย มีเงิน	การซ่อมไฟฟ้า	ผ่าน
0003	TR05	นายรวย มีเงิน	การซ่อมตู้เย็น	ไม่ผ่าน

Third Normal Form (3NF)

30

- 1. Relation นั้นจะต้องมีคุณสมบัติ 2NF
- 2. ต้องไม่มีความสัมพันธ์ระหว่าง Non-key Attribute หรือ
ไม่มี Transitive Dependency

สรุป : แอททริบิวต์ที่ไม่ใช่คีย์หลัก ต้องไม่ขึ้นต่อกันเอง

Third Normal Form (3NF)

31

□ วิธีขจัดปัญหา

1. สร้างตารางเพิ่ม
2. นำแอททริบิวต์ที่มีปัญหามาใส่ในตารางใหม่
3. กำหนดคีย์หลัก
4. แอททริบิวต์ที่ย้ายจากตารางเดิมไปใส่ในตารางใหม่ให้ตัดออกจากตารางเดิม
5. นำคีย์หลักในข้อ 3 ไปใส่ในตารางเดิม

<u>รหัสพนักงาน</u>	ชื่อสกุล	รหัสแผนก	ชื่อแผนก	เงินเดือน
P001	นพเกศ แก้วใส	A001	บัญชี	25000
P002	วารุณี รวดเร็ว	F001	การเงิน	30000

- คีย์หลักของตารางนี้คือ รหัสพนักงาน
- จากตารางยังมีฟังก์ชันการขึ้นต่อกันแบบ Transitive Dependency อยู่ คือ
- รหัสแผนก ซึ่งไม่ใช่คีย์หลักของตาราง แต่สามารถระบุค่า ชื่อแผนก ได้ คือ ถ้ารู้รหัสแผนก ก็ารู้ชื่อแผนก

จากตารางข้างบน ทำให้อยู่ในรูป 3 NF จะได้ 2 ตารางข้างล่างนี้

<u>รหัสพนักงาน</u>	ชื่อสกุล	เงินเดือน	รหัสแผนก
P001	นพเกศ แก้วใส	25000	A001
P002	วารุณี รวดเร็ว	30000	F001

<u>รหัสแผนก</u>	ชื่อแผนก
A001	บัญชี
F001	การเงิน

สรุป Normalization

33

- 1NF ทุกแอททริบิวต์ในแต่ละแถวมีค่าของข้อมูลเพียงค่าเดียว
- 2NF รีเลชันนั้นต้องไม่มีความสัมพันธ์ระหว่างแอททริบิวต์แบบบางส่วน (แอททริบิวต์ทุกตัวต้องขึ้นกับคีย์หลักทุกตัว ไม่ขึ้นอยู่กับตัวใดตัวหนึ่ง)
- 3NF ทุกแอททริบิวต์ที่ไม่ใช่คีย์หลักไม่มีคุณสมบัติในการกำหนดค่าของแอททริบิวต์อื่น

แบบฝึกหัด

ให้นักศึกษาแปลงตารางต่อไปนี้ให้อยู่ในรูป NF1-NF3 โดยละเอียด

1) การลงทะเบียนเรียน

รหัส นักศึกษา	ชื่อนักศึกษา	รหัสวิชา	ชื่อวิชา	หน่วยกิต	เกรด
53001	วนิดา	AB12	บัญชี	3	A
		CD01	การเงิน	3	A
		PC09	สถิติ	3	B
53009	สุมาลี	AB12	บัญชี	3	B
		CD01	การเงิน	3	A

แบบฝึกหัด

ให้นักศึกษาแปลงตารางต่อไปนี้ให้อยู่ในรูป NF1-NF3 โดยละเอียด

2) การลงทะเบียนเรียน

รหัส นักศึกษา	ชื่อนักศึกษา	รหัส คณะ	ชื่อคณะ	รหัสวิชา	ชื่อวิชา	หน่วยกิต	เกรด
53001	วนิดา	BC	บริหารธุรกิจ	AB12	บัญชี	3	A
				CD01	การเงิน	3	A
				PC09	สถิติ	3	B
53009	สุมาลี	HT	การโรงแรม	AB12	บัญชี	3	B
				CD01	การเงิน	3	A

แบบฝึกหัด

ให้นักศึกษาแปลงตารางต่อไปนี้ให้อยู่ในรูป NF1-NF3 โดยละเอียด

3) การตั้งสินค้า

เลขที่ใบสั่งซื้อ	วันที่ซื้อ	รหัสลูกค้า	ชื่อผู้สั่งซื้อ	รหัสสินค้า	ชื่อสินค้า	จำนวนที่ซื้อ	ราคาต่อหน่วย
OR001	02/09/2552	C001	เดวิด	AB12	ตู้เย็น	4	4000
				CD01	พัดลม	3	2000
				PC09	แอร์	4	6000
OR002	02/09/2552	C005	ไมเคิล	TP01	เตาอบ	3	3000
				CD01	พัดลม	2	2000