

Attribute Database Management

หลักการพิจารณาเพื่อกำหนด Feature Type (Line , Point , Polygon)

1. ลักษณะรูปร่างของวัตถุ
2. มาตรการส่วน : ขนาดวัตถุในแผนที่/ในภูมิประเทศ
3. การใช้งานหรือประโยชน์ใช้สอย

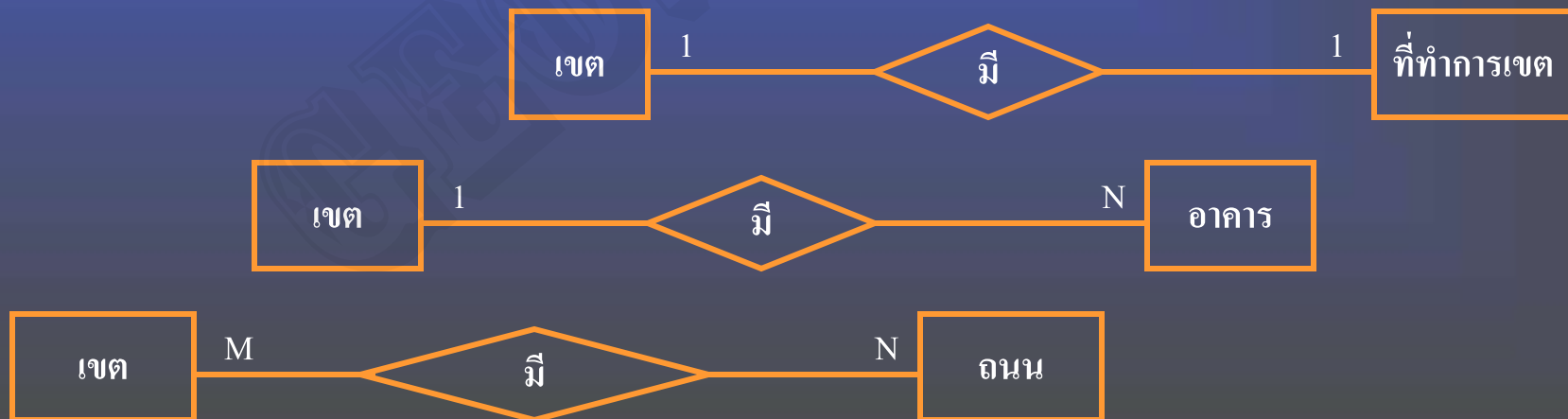


Entity : อาคาร

Attribute : House_ID House_No Road Sub_district District Province

House_ID	House_No	Road	Sub_district	District	Province
10452000001	20/5	เสรีไทย	คันนายาว	คันนายาว	กรุงเทพมหานคร
10352000025	45	เสรีไทย	คันนายาว	ปิ้งกุ่ม	กรุงเทพมหานคร
10012000025	17	ดินสอ	เสาชิงช้า	พระนคร	กรุงเทพมหานคร

Relationship :



แบบจำลองข้อมูลแบบสัมพันธ์

House_ID	House_No	Road	Sub_district	District_code	Type_c
10452000001	20/5	เสรีไทย	คันนายาว	45	S
10352000025	45	เสรีไทย	คันนายาว	35	D
10012000025	17	คินสอ	เสาชิงช้า	01	S



District_code	District_name
01	พระนคร
35	บึงกุ่ม
45	คันนายาว

Type_c	Type
S	เดี่ยว
D	แฝด

ชนิดของข้อมูล (หลักการกำหนด)

1. ข้อมูลตัวอักษรหรือข้อความ (Character or Text)
2. ข้อมูลเชิงจำนวน (Numerical Data)
3. ข้อมูลรหัส (Code Data)
4. ข้อมูลวันที่ (Date Data)
5. ข้อมูลตรรกะ (Logical Data)
6. ข้อมูลภาพ (Image Data)
7. ข้อมูลภาพเคลื่อนไหว (Moving Data)
8. ข้อมูลเสียง (Voice Data)

วิเคราะห์งานพื้นฐาน (ดูแบบฟอร์ม , ข้อมูล , ขบวนการ ฯลฯ)

1. จัดกลุ่มข้อมูล (Base map , specific data ฯลฯ)
2. แยกองค์ประกอบข้อมูลให้เล็กที่สุด (ที่อยู่ : บ้านเลขที่ หมู่ที่ ถนน ตำบล อำเภอ จังหวัด รหัสไปรษณีย์)
3. แปลงองค์ประกอบข้อมูลเป็น Field name
4. กำหนดชนิด-ขนาดข้อมูล
5. กำหนด Primary key , Foreign key

การออกแบบฐานข้อมูลของระบบงานขนาดใหญ่ที่ซับซ้อน

ต้องศึกษาโครงสร้างพื้นฐานของฐานข้อมูล

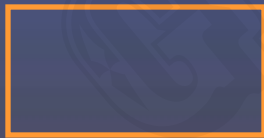
ต้องวิเคราะห์ระบบในภาพรวม

สร้างแบบจำลอง

E-R model เป็นแบบจำลองเชิงแนวคิด (Conceptual design)

E-R model หรือ Entity-Relationship model

- E-R model คือ เครื่องมือที่ใช้ในการออกแบบระบบฐานข้อมูลที่แสดงความสัมพันธ์ระหว่าง Entity ต่าง ๆ ในลักษณะรูปภาพ
- อำนวยความสะดวกในการสื่อสารระหว่างนักออกแบบฐานข้อมูล ผู้เขียนโปรแกรม และผู้ใช้
- การออกแบบ E-R model ต้องกำหนดให้ได้ว่าสิ่งใดคือ Entity Attribute และสร้างความสัมพันธ์ระหว่าง Entity



Entity



Attribute



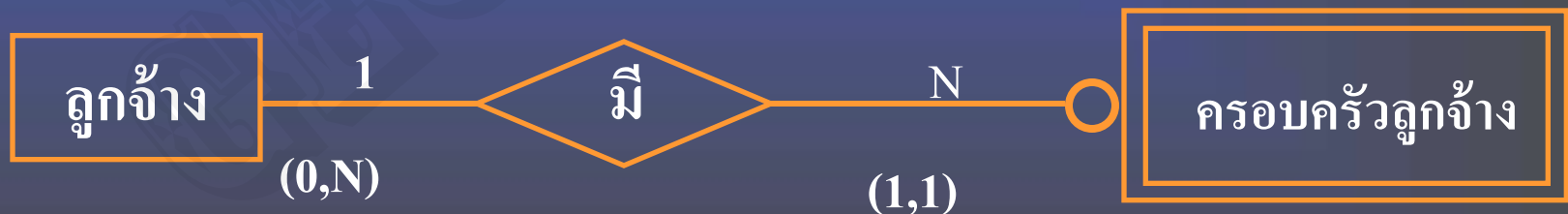
Relationship

นักเรียน

Entity ปกติ : Entity ที่สนใจและต้องการจัดเก็บข้อมูล

ครอบครัวลูกจ้าง

Entity อ่อนแอ : Entity ที่ขึ้นอยู่กับการมีอยู่ของ Entity อื่น (เรียกว่า) Parent Entity และมี Primary key เกิดจากส่วนหนึ่งหรือทั้งหมดของ Primary key ของ Parent Entity



รหัสประจำตัวนักเรียน

Primary key attribute : Attribute ที่เป็นคีย์หลัก

ชื่อ

นามสกุล

Simple attribute : Attribute ที่ไม่สามารถแบ่งย่อยได้อีก

ถนน

ตำบล

อำเภอ

Composite attribute : Attribute ที่สามารถแบ่งย่อยได้อีก

เลขที่บ้าน

ที่อยู่

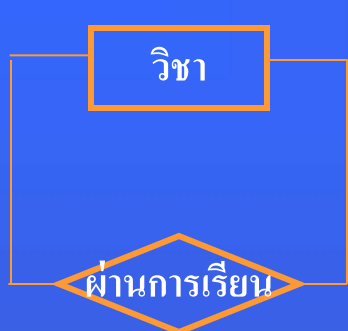
จังหวัด

ปริญญา

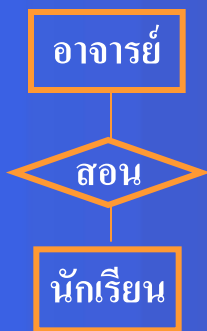
Multivalue attribute : Attribute ที่สามารถมีได้หลายค่า

อายุ

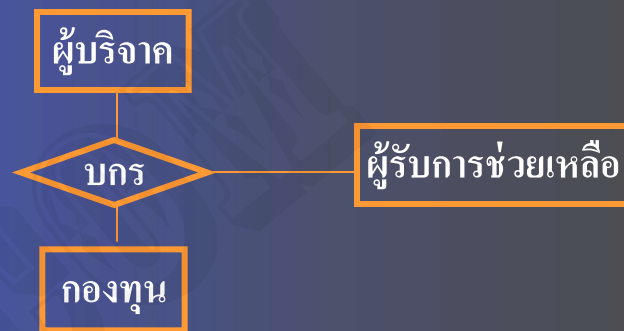
Derive attribute : Attribute ที่สามารถหาค่ามาได้จากการคำนวณจาก Attribute อื่น ที่มีอยู่

Relationship : ระดับความสัมพันธ์

Unary relationship



Binary relationship



Ternary relationship

Relationship : ลักษณะต่าง ๆ ที่เกี่ยวกับความสัมพันธ์

Cardinality : ระบุค่าสูงสุด ต่ำสุดใน Entity หนึ่งที่มีความสัมพันธ์กับ Entity หนึ่ง

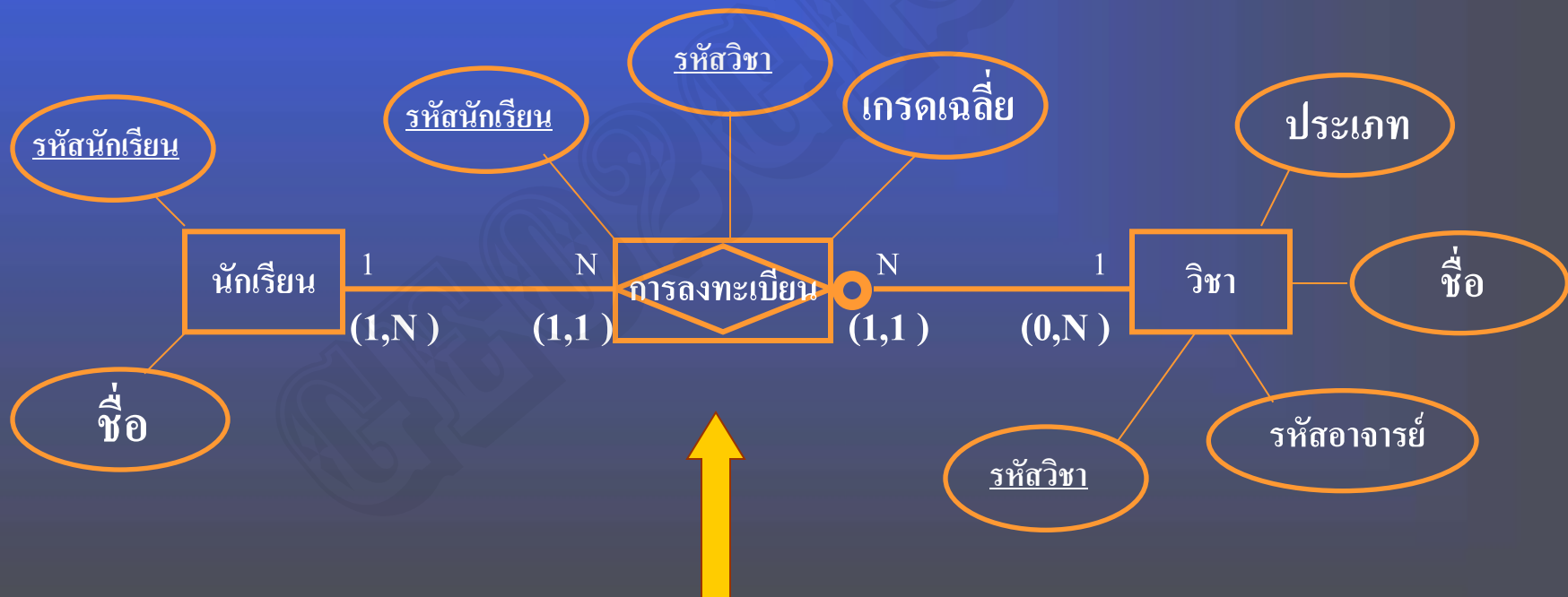


ความสัมพันธ์แบบเลือก(Optional) และแบบบังคับ(Mandatory)



Relationship : ลักษณะต่าง ๆ ที่เกี่ยวกับความสัมพันธ์ Attribute Database Management

Composite entity : Entity ที่สร้างขึ้นเพื่อแปลงสัมพันธ์แบบ $M : N$ ให้เป็น $1 : N$



Composite entity

กระบวนการออกแบบ E-R model

- กำหนด Entity ที่ต้องการในระบบ
- กำหนดความสัมพันธ์ระหว่าง Entity
- กำหนด Attribute ให้กับ Entity

กระบวนการออกแบบ E-R model

ตัวอย่างการสร้าง E-R model

1. บริษัท ที่ปรึกษา A มี 5 แผนก (Division)
2. แต่ละแผนก ทำโครงการ (Project) ได้หลายโครงการ
3. แต่ละโครงการขึ้นอยู่กับแผนกเดียว
4. แต่ละแผนกมีพนักงาน (Employee) ได้หลายคน พนักงานแต่ละคนสังกัดอยู่แผนก

ใดแผนกหนึ่งเท่านั้น

5. แต่ละแผนกมีพนักงาน 1 คนเป็นผู้จัดการ
6. พนักงานอาจจะมีหรือไม่มีครอบครัว (Family) ก็ได้
7. พนักงานแต่ละคนทำงานได้หลายโครงการหรือไม่ทำเลยก็ได้
8. แต่ละโครงการต้องมีพนักงานอย่างน้อย 1 คนทำงาน
9. บริษัทฯ รวบรวมความชำนาญ (Skill) ของพนักงานแต่ละคนไว้
10. ความชำนาญอย่างหนึ่งอาจมีพนักงานหลายคนทำได้
11. พนักงานคนหนึ่งอาจจะชำนาญหลายอย่าง

ตัวอย่างการสร้าง E-R model

แต่ละแผนก ทำโครงการ (Project) ได้หลายโครงการ แต่ละโครงการขึ้นอยู่กับแผนกเดียว



แต่ละแผนกมีพนักงาน (Employee) ได้หลายคน พนักงานแต่ละคนสังกัดอยู่แผนกใดแผนกหนึ่งเท่านั้น แต่ละแผนกมีพนักงาน 1 คนเป็นผู้จัดการ



พนักงานอาจจะมีหรือไม่มีครอบครัว (Family) ก็ได้

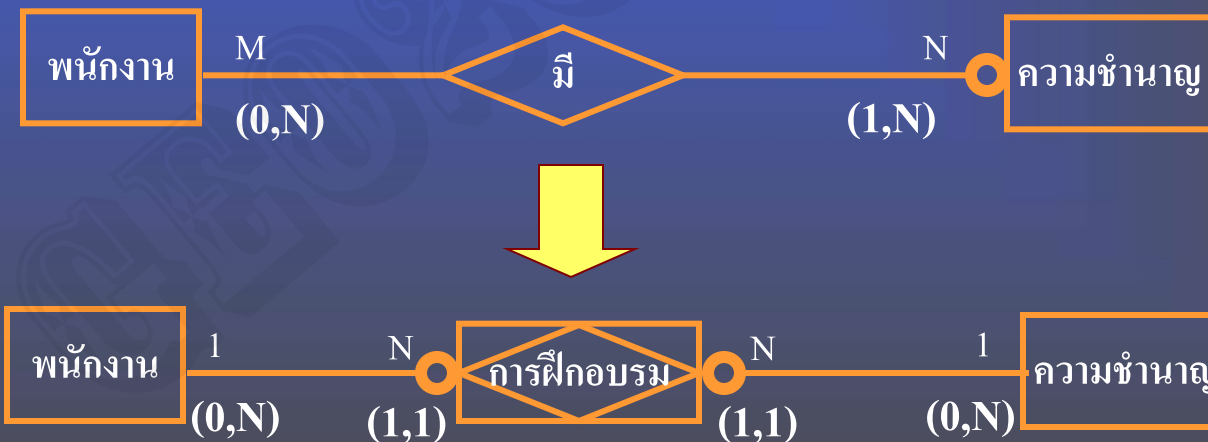


ตัวอย่างการสร้าง E-R model

พนักงานแต่ละคนทำงานได้หลายโครงการหรือไม่ทำเลยก็ได้แต่ละโครงการต้องมีพนักงานอย่างน้อย 1 คนทำงาน



บริษัทฯ รวบรวมความชำนาญ (Skill) ของพนักงานแต่ละคนไว้ ความชำนาญอย่างหนึ่งอาจมีพนักงานหลายคนทำได้ พนักงานคนหนึ่งอาจจะชำนาญหลายอย่าง



ตัวอย่างการสร้าง E-R model

<u>Entity</u>	<u>Relationship</u>	<u>Connectivity</u>	<u>Entity</u>
แผนก	ทำ	1 : N	โครงการ
แผนก	มี	1 : N	พนักงาน
แผนก	จัดการ	1 : 1	พนักงาน
พนักงาน	มี	1 : 1	ครอบครัว
พนักงาน	ทำ	1 : N	โครงการ
พนักงาน	มี	M : N	ความชำนาญ

ตัวอย่างการสร้าง E-R model



การแปลง E-R Diagram ให้เป็นโครงสร้างฐานข้อมูล

สร้างตารางจาก E-R Diagram

1. ใช้ชื่อของ Entity เป็นชื่อของตาราง จากนั้นกำหนด Attribute พร้อมกำหนด Primary key
2. กรณี Relationship ของ Entity เป็นแบบ 1 : N ให้นำ Primary key ของ Entity ที่อยู่ข้าง 1 ไปใส่เป็น Foreign key ในตารางที่อยู่ข้าง N
3. ในกรณีของ Entity อ่อนแอ ให้ใช้ Primary key ของ Parent Entity เป็น Primary key ของ Entity อ่อนแอ หรือส่วนหนึ่งของ Primary key ของ Entity อ่อนแอ

การแปลง E-R Diagram ให้เป็นโครงสร้างฐานข้อมูล

ตัวอย่างการสร้างตารางจาก E-R Diagram

แผนก

Division (D_ID, D_Name, Emp_ID)

พนักงาน

Employee (Emp_ID, D_ID, Emp_Lname, Emp_Fname, Emp_Salary)

ความชำนาญ

Skill (Skill_ID, Skill_Description)

การฝึกอบรม

Training (Emp_ID, Skill_ID, Training_Experience)

โครงการ

Project (Pj_ID, Pj_Name, D_ID, Emp_ID, Pj_budget)

ครอบครัว

Family (Emp_ID, Family_ID, Family_Insurance)

E-R model เป็นเครื่องมือในการออกแบบระบบฐานข้อมูลเชิงแนวคิด นำเสนอรายละเอียดให้ผู้ใช้และผู้ออกแบบเห็นภาพรวมฐานข้อมูลทั้งระบบ และเห็นความสัมพันธ์ของข้อมูลอย่างชัดเจน

กระบวนการ Normalization

- กระบวนการวิเคราะห์ความสัมพันธ์ระหว่าง Attribute ในแต่ละ Relationship เพื่อตรวจสอบและแก้ไขปัญหาคำซ้ำซ้อนของข้อมูลในแต่ละ Relation ให้เหลือน้อยที่สุด หรือจนกระทั่งไม่มีความซ้ำซ้อนอยู่เลย
- การที่ต้องมีกระบวนการ Normalization เนื่องจากขั้นตอนการออกแบบโดยใช้ E-R model เป็นการวิเคราะห์ความสัมพันธ์ระดับ Entity เท่านั้น แต่ไม่สามารถวิเคราะห์ความสัมพันธ์ระหว่าง Attribute ในแต่ละ Relationship ได้

แนวคิดเกี่ยวกับ Normalization

Attribute Database Management

Comp_ID	Company	City	Region	Product_ID	Prod_Name	Color
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD01	เก้าอี้	แดง
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD11	โต๊ะ	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD12	โต๊ะ	แดง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD01	เก้าอี้	แดง
C03	C Furniture	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD11	โต๊ะ	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD03	เก้าอี้	ฟ้า
C04	D Furniture	ตาก	เหนือ	PD21	ตู้เหล็ก	แดง

แนวคิดเกี่ยวกับ Normalization

Attribute Database Management

Comp_ID	Company	City	Region
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน
C02	บี เฟอร์นิเจอร์	เลย	อีสาน
C02	บี เฟอร์นิเจอร์	เลย	อีสาน
C03	C Furniture	กทม.	กลาง
C04	D Furniture	ตาก	เหนือ
C04	D Furniture	ตาก	เหนือ
C04	D Furniture	ตาก	เหนือ

Comp_ID	Company	City	Region
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน
C03	C Furniture	กทม.	กลาง
C04	D Furniture	ตาก	เหนือ

City	Region
กทม.	กลาง
เลย	อีสาน
ตาก	เหนือ

แนวคิดเกี่ยวกับ Normalization

Attribute Database Management

Product_ID	Prod_Name	Color
PD01	แก้อื้อ	แดง
PD02	แก้อื้อ	น้ำตาล
PD11	โตะ	น้ำตาล
PD12	โตะ	แดง
PD01	แก้อื้อ	แดง
PD02	แก้อื้อ	น้ำตาล
PD11	โตะ	น้ำตาล
PD03	แก้อื้อ	ฟ้า
PD21	ตุ๋นเหล็ก	แดง

Product_ID	Prod_Name	Color
PD01	แก้อื้อ	แดง
PD02	แก้อื้อ	น้ำตาล
PD03	แก้อื้อ	ฟ้า
PD11	โตะ	น้ำตาล
PD12	โตะ	แดง
PD21	ตุ๋นเหล็ก	แดง

Comp_ID	Company	City	Region	Product_ID	Prod_Name	Color
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD01	เก้าอี้	แดง
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD11	โต๊ะ	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD12	โต๊ะ	แดง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD01	เก้าอี้	แดง
C03	C Furniture	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD11	โต๊ะ	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD03	เก้าอี้	ฟ้า
C04	D Furniture	ตาก	เหนือ	PD21	ตู้เหล็ก	แดง

ปัญหาที่เกิดจากความซ้ำซ้อนของข้อมูล

1. การแก้ไขข้อมูล

เมื่อต้องการแก้ไขชื่อสินค้าจาก เก้าอี้ เป็น เก้าอี้มีพนักพิง ต้องแก้ แถวที่ 1,2,5,6,8

Comp_ID	Company	City	Region	Product_ID	Prod_Name	Color
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD01	เก้าอี้	แดง
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD11	โต๊ะ	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD12	โต๊ะ	แดง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD01	เก้าอี้	แดง
C03	C Furniture	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD11	โต๊ะ	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD03	เก้าอี้	ฟ้า
C04	D Furniture	ตาก	เหนือ	PD21	ตู้เหล็ก	แดง

ปัญหาที่เกิดจากความซ้ำซ้อนของข้อมูล

2. การเพิ่มข้อมูล

เมื่อบริษัทต้องการจำหน่ายสินค้าใหม่ คือ PD31 : เติ่ง แต่ยังไม่ได้กำหนดให้บริษัทใดเป็นผู้ผลิต ทำให้ยังไม่มีข้อมูลผู้ผลิตสินค้า ถ้า Primary key คือ Comp_ID จึงบันทึกข้อมูลไม่ได้

Comp_ID	Company	City	Region	Product_ID	Prod_Name	Color
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD01	เก้าอี้	แดง
C01	เอ เฟอร์นิเจอร์	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD11	โต๊ะ	น้ำตาล
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD12	โต๊ะ	แดง
C02	บี เฟอร์นิเจอร์	เลย	อีสาน	PD01	เก้าอี้	แดง
C03	C Furniture	กทม.	กลาง	PD02	เก้าอี้	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD11	โต๊ะ	น้ำตาล
C04	D Furniture	ตาก	เหนือ	PD03	เก้าอี้	ฟ้า
C04	D Furniture	ตาก	เหนือ	PD21	ตู้เหล็ก	แดง

ปัญหาที่เกิดจากความซ้ำซ้อนของข้อมูล

3. การลบข้อมูล

เมื่อบริษัท D Furniture ปิดกิจการ ถ้ามลบแถว 7, 8,9 อาจจะทำให้ข้อมูลสำคัญอื่นหายไปด้วย
เช่น สินค้าที่เป็น เก้าอี้ สี ฟ้า