



Database System



Entity-Relationship Model

DB Application Life Cycle

- ▶ วิเคราะห์ความต้องการ (Requirement Analysis)
 - ▶ วิเคราะห์ระบบปัจจุบัน (ข้อมูลและการทำงาน)
 - ▶ วิเคราะห์ความต้องการของผู้ใช้
- ▶ ออกแบบฐานข้อมูลและโปรแกรม (Design)
 - ▶ Conceptual > Logical > Physical & Functional
- ▶ ดำเนินงานสร้างฐานข้อมูลและโปรแกรม (Implement)
 - ▶ System Development and Testing
- ▶ ใช้งานฐานข้อมูลและโปรแกรม (Deployment)
 - ▶ Installation, Migration/Conversion, Training
- ▶ บำรุงรักษา (Maintenance)
Monitoring, Tuning, Upgrading, Achieving

วิเคราะห์ความต้องการ

▶ ทรัพยากร

- ▶ ฐานข้อมูลและโปรแกรมที่มีอยู่เดิม
- ▶ ผู้ใช้งานในทุกระดับและผู้เชี่ยวชาญในงานนั้น
- ▶ เอกสาร, รายงาน, คู่มือ, แบบฟอร์ม, กฎ, สูตร

▶ การประเมินผล : กำหนด

- ▶ ผู้ใช้หรือโปรแกรมนำข้อมูลไปใช้อย่างไร
- ▶ อะไรคือสารสนเทศที่ต้องการ

▶ ผลลัพธ์

- ▶ การวิเคราะห์ข้อมูลนำไปสู่การออกแบบฐานข้อมูล
 - ▶ **Business Rule/Logic** เปลี่ยนเป็น **Integrity Constraints**
- ▶ การวิเคราะห์การทำงานนำไปสู่การออกแบบโปรแกรม
 - ▶ **Tools: Data Flow Diagram; UML Use Case**

ออกแบบฐานข้อมูล

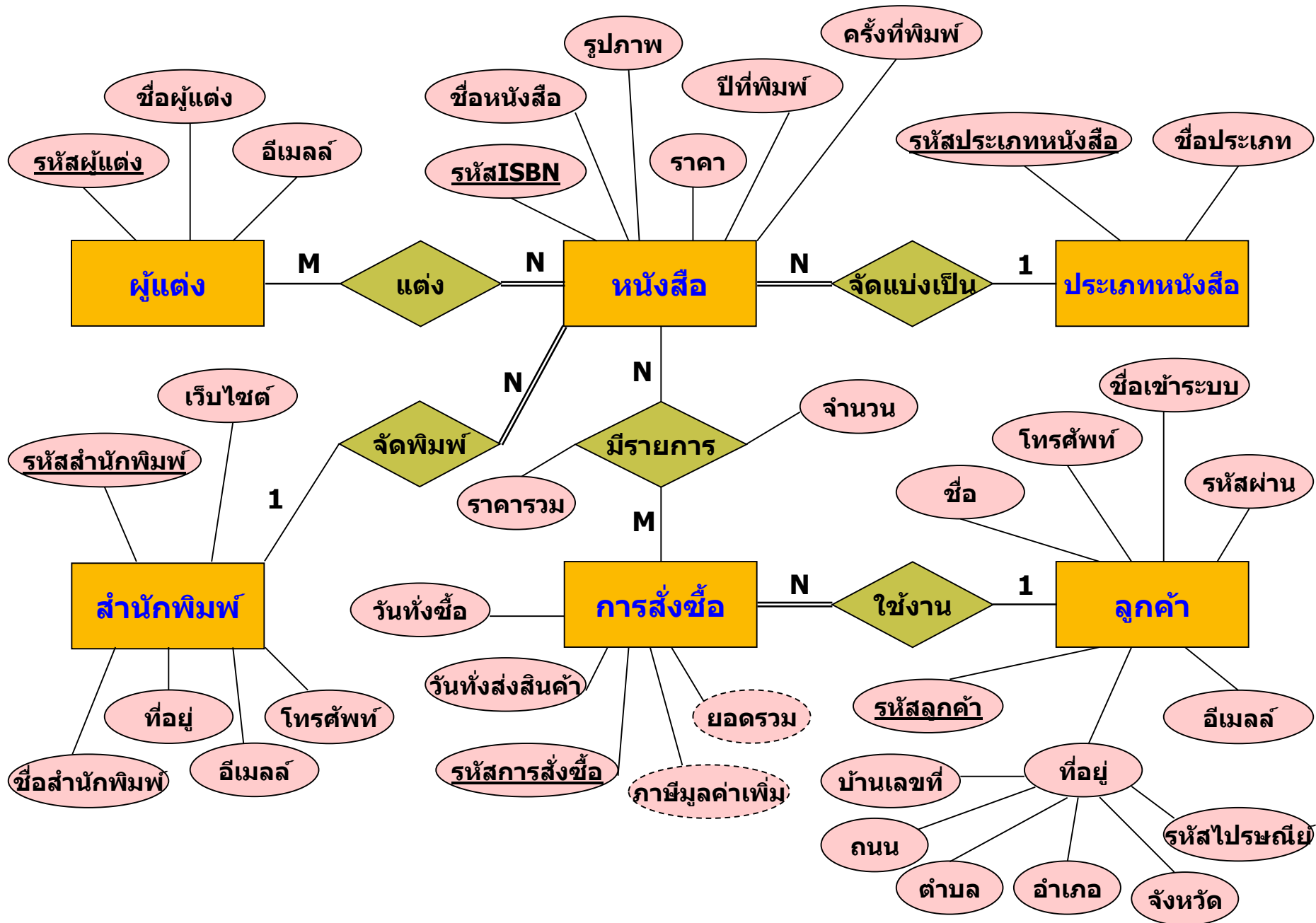
- ▶ ออกแบบฐานข้อมูลในระดับแนวคิด(Conceptual Database Design)
- ▶ การออกแบบฐานข้อมูลในระดับตรรกะ(Logical Database Design)
- ▶ การออกแบบฐานข้อมูลในระดับกายภาพ(Physical Database Design)

Conceptual Design

- ▶ เป็นขั้นตอนในการวิเคราะห์ข้อมูลที่จำเป็นทั้งหมดที่เกี่ยวข้องกับระบบสารสนเทศ
- ▶ โดยกำหนดกลุ่มข้อมูล(Entity)และกำหนดความสัมพันธ์(Relationship)ระหว่างข้อมูลต่าง ๆ จากนั้นจึงระบุรายละเอียด(Attribute)ต่าง ๆ ของข้อมูลทั้งหมด และระบุกฎที่จะควบคุมความคงสภาพของข้อมูล(Data Integrity)
- ▶ แสดงแนวคิดการออกแบบโดยใช้ ER Model แล้วพัฒนา ER Diagram

Conceptual Design (cont.)

- ▶ **Input: Data Requirement-** จากผู้ใช้, ผู้เชี่ยวชาญระบบ, ระบบเดิม
- ▶ **Output: Conceptual Schema – Logical data models**
- ▶ **Process:** ระบุความต้องการ, กำหนดโมเดลที่รองรับความต้องการ



Logical Design

- ▶ เป็นการแปลง ER Diagram ให้เป็นเค้าร่างรีเลชัน(Relational schema) ตามทฤษฎีฐานข้อมูลเชิงสัมพันธ์
- ▶ นำรีเลชันมาปรับบรรทัดฐาน(Normalization) ให้อยู่ในรูปแบบบรรทัดฐานที่เหมาะสม

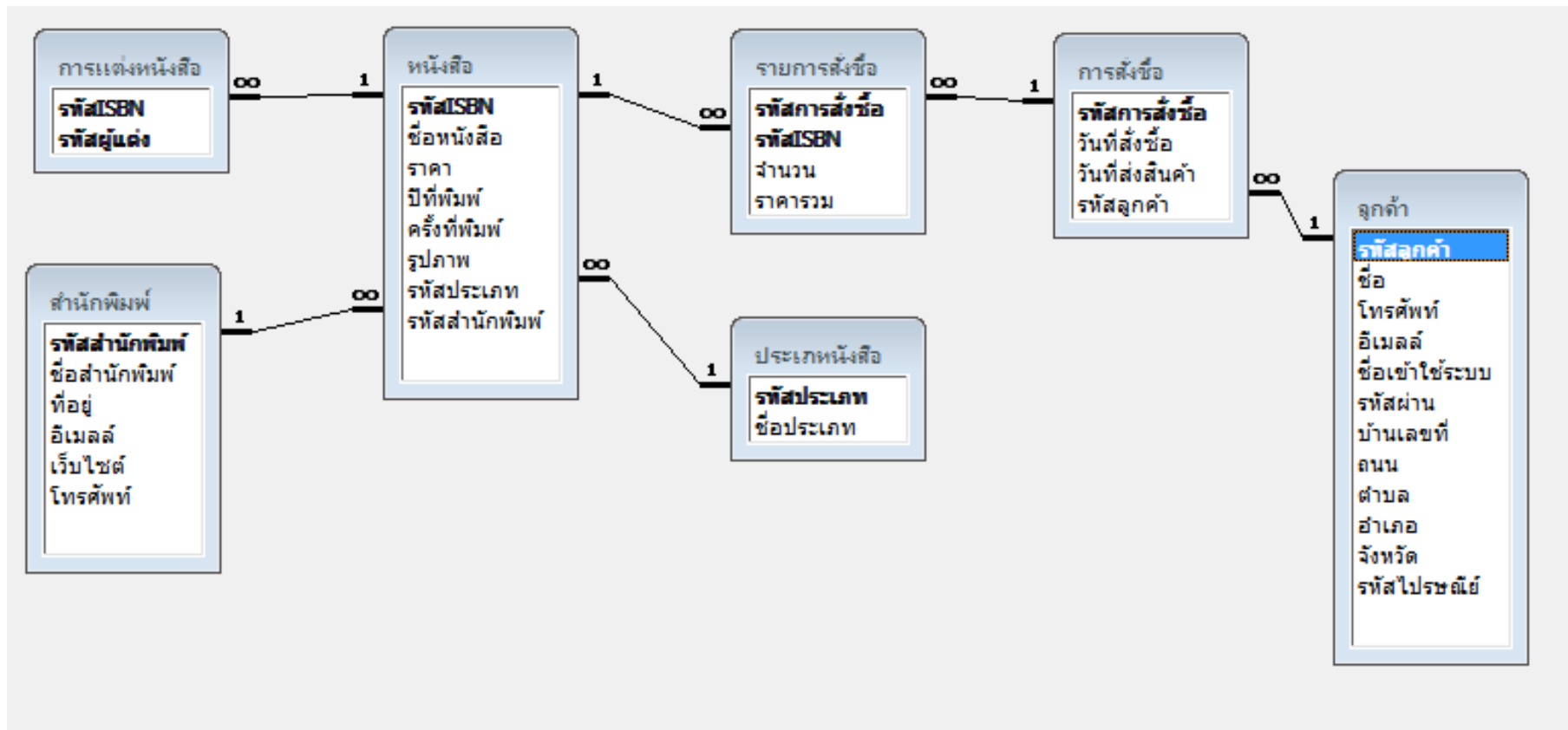
Logical Design (cont.)

- ▶ Input: Conceptual Schema
- ▶ Output: **Logical Schema** – DBMS products
- ▶ Process: ค้นหา data model ที่เหมาะสม, convert the schema to model

Relational Schema

- ▶ จาก ER Diagram แปลงเป็นเค้าร่างฐานข้อมูลได้ดังนี้
- ▶ ประเภทหนังสือ(รหัสประเภท , ชื่อประเภท)
- ▶ หนังสือ(รหัสISBN , ชื่อหนังสือ , ราคา , ปีที่พิมพ์ , ครั้งที่พิมพ์ , รูปภาพ , รหัสประเภท , รหัสสำนักพิมพ์)
- ▶ สำนักพิมพ์(รหัสสำนักพิมพ์ , ชื่อสำนักพิมพ์ , ที่อยู่ , โทรศัพท์ , เว็บไซต์ , อีเมลล์)
- ▶ ผู้แต่ง(รหัสผู้แต่ง , ชื่อผู้แต่ง , อีเมลล์)
- ▶ ลูกค้า(รหัสลูกค้า , ชื่อลูกค้า , โทรศัพท์ , อีเมลล์ , ชื่อเข้าใช้ระบบ , รหัสผ่าน , บ้านเลขที่ , ถนน , ตำบล , อำเภอ , จังหวัด , รหัสไปรษณีย์)
- ▶ การสั่งซื้อ(รหัสการสั่งซื้อ , วันที่สั่งซื้อ , วันที่ส่งสินค้า , รหัสลูกค้า)
- ▶ การแต่งหนังสือ(รหัสISBN , รหัสผู้แต่ง)
- ▶ รายการสั่งซื้อ(รหัสการสั่งซื้อ , รหัส ISBN , จำนวน , ราคารวม)

Relational Schema



Physical Design

- ▶ เป็นการออกแบบโครงสร้างข้อมูลที่จะนำไปจัดเก็บในฐานข้อมูล โดยกำหนดโครงสร้างตามคุณสมบัติข้อมูลของระบบจัดการฐานข้อมูลที่จะใช้ในการจัดการข้อมูล
- ▶ ระบุพจนานุกรมของข้อมูล(Data Dictionary)

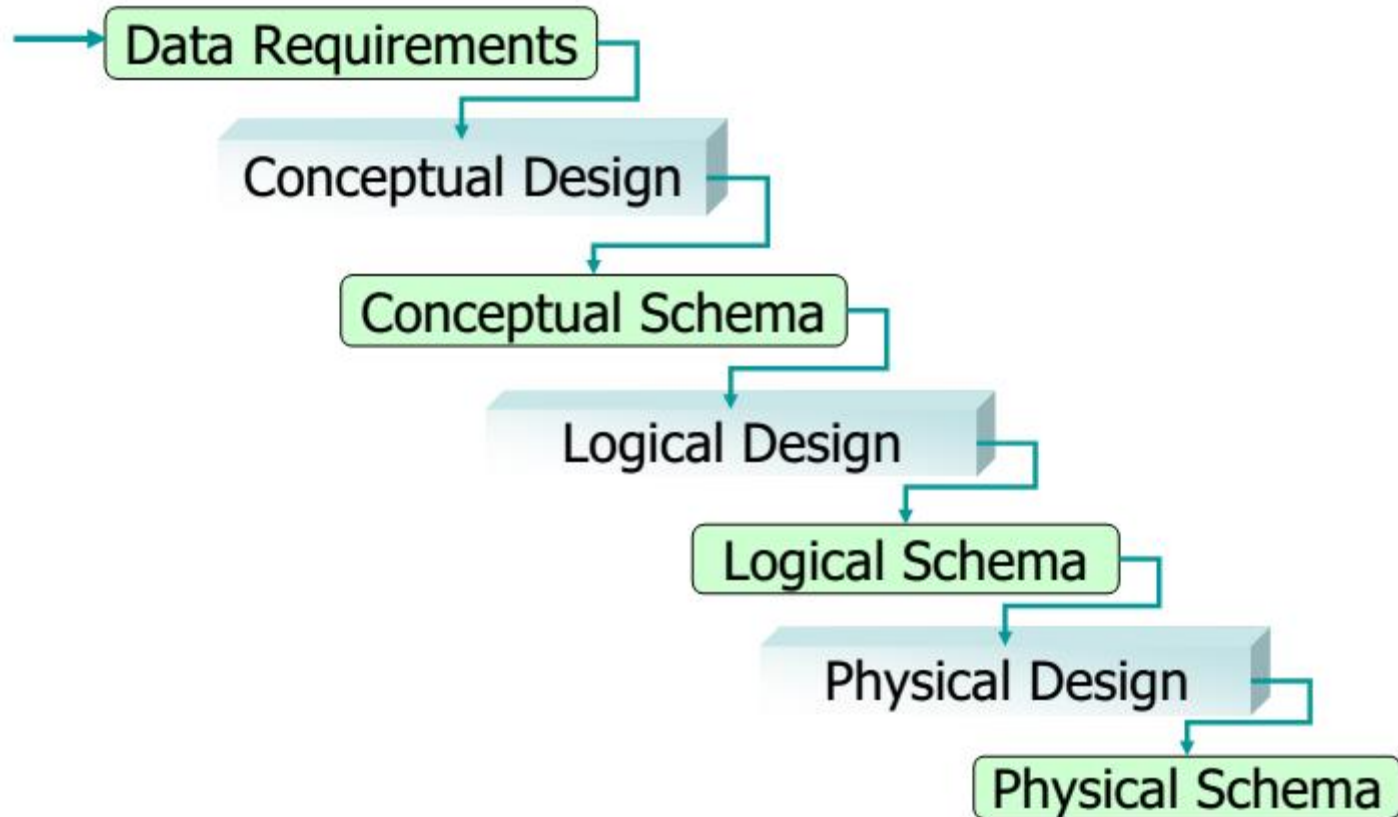
Physical Design (cont.)

- ▶ Input: Logical Schema
- ▶ Output: **Physical Schema** – schema ที่เฉพาะเจาะจงเพื่อเจาะจง DBMS
- ▶ Process: ค้นหาผลิตภัณฑ์ DBMS และดำเนินการ

พจนานุกรมข้อมูล(Data Dictionary)

Table name	Attribute name	Description	Data Type	Size	Key	References
BookType (ประเภทหนังสือ)	BTypeID	รหัสประเภท	Char	5	PK	
	BTypeName	ชื่อประเภท	Varchar	30		
Book (หนังสือ)	ISBN	รหัสISBN	Char	15	PK	
	BName	ชื่อหนังสือ	Varchar	50		
	Price	ราคา	Float			
	Edition	ครั้งที่พิมพ์	Int			
	Published	ปีที่พิมพ์	Int			
	Picture	รูปภาพ	Blob			
	BTypeID	รหัสประเภท	Char	5	FK	BookType

Database Design



Entity-Relationship Model (ER-Model)

- ▶ เป็นแบบจำลองที่ใช้แสดงแนวคิดการออกแบบฐานข้อมูลในระดับแนวคิด (Conceptual schema) โดยจะแสดงความสัมพันธ์ระหว่างข้อมูลที่ประกอบด้วย เอนทิตี , แอททริบิวต์ และความสัมพันธ์
- ▶ คิดค้นโดย Peter Chen ในปี 1976
- ▶ เป็นอิสระจากระบบการจัดการฐานข้อมูล(DBMS)
- ▶ นักวิเคราะห์และออกแบบระบบจะใช้ ERM เป็นสื่อกลางในการสื่อสารระหว่างผู้ใช้และผู้พัฒนาโปรแกรม

ER Model (cont.)

- ▶ ER-Model เป็นการออกแบบฐานข้อมูลในระดับแนวคิดในลักษณะจากบนลงล่าง (Top-Down Strategy)
- ▶ ผลจากการออกแบบฐานข้อมูลจะได้เค้าร่างในระดับแนวคิดที่ประกอบด้วย
 - ▶ เอนทิตีที่ควรจะมีในระบบ
 - ▶ ความสัมพันธ์ระหว่างเอนทิตีว่าเป็นอย่างไร
 - ▶ แอททริบิวต์ซึ่งเป็นรายละเอียดที่อธิบายเอนทิตี และมีความสัมพันธ์กันอย่างไร

องค์ประกอบของ ER Model

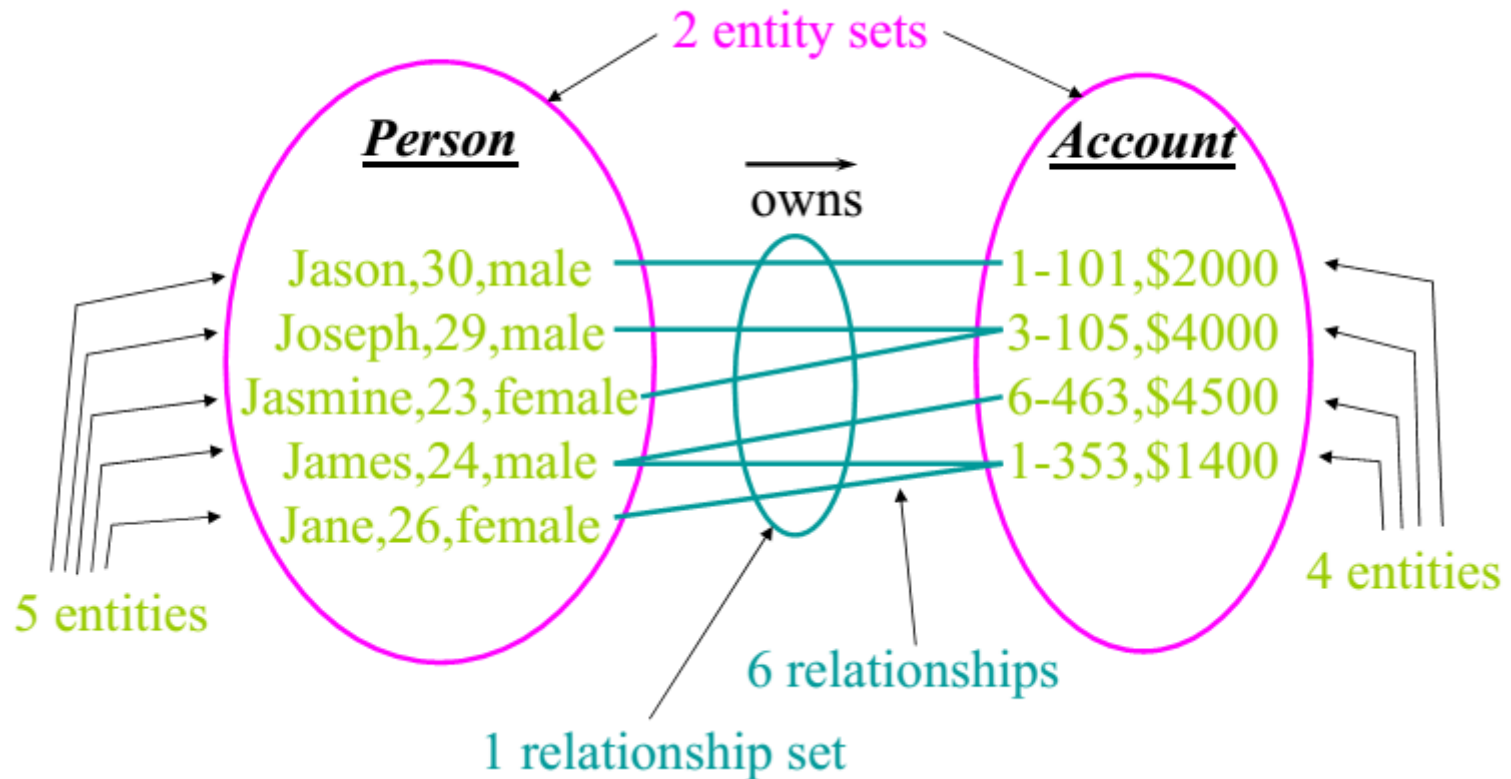
▶ เอนทิตี(Entity Sets)

- ▶ กลุ่มของสิ่งต่างๆ ในโลกความเป็นจริง ที่ต้องการจัดเก็บเป็นข้อมูลไว้ในฐานข้อมูล
- ▶ เช่น พนักงาน , ลูกค้า , นักศึกษา , อาจารย์ , สมาชิก , สุนัข , พืช , สัตว์
- ▶ หรืออาจเป็น การเช่า , การสั่งซื้อ , การลงทะเบียน
- ▶ **Entity sets** คือกลุ่มของเอนทิตีประเภทเดียวกันและมีคุณสมบัติหรือคุณลักษณะร่วมกัน

▶ ความสัมพันธ์(Relationships)

- ▶ ความเกี่ยวข้องกันระหว่างหลาย ๆ เอนทิตี
- ▶ **Relationship set** คือกลุ่มของความสัมพันธ์ที่เป็นประเภทเดียวกัน

Entities and Relationships



Entity Sets

- ▶ **Entity** คือ กลุ่มของแอตทริบิวต์ (**Attribute**)
- ▶ **Attributes** คือคุณสมบัติหรือคุณลักษณะ ที่ใช้อธิบายรายละเอียดของเอนทิตี
- ▶ **Domain** คือค่าที่ **attribute** สามารถบรรจุได้

Entity, Attribute, Domain

- ▶ Person
- ▶ {name, age, gender}
 - ▶ (Jason, 30, male)
 - ▶ (Jane, 26, female)
- ▶ Domain
 - ▶ name: character string
 - ▶ age: $0 < \text{age} < 200$
 - ▶ gender: {male, female}

- ▶ Account
- ▶ {acct number, balance}
 - ▶ (1-101, 2000 ₪)
 - ▶ (3-105, 4000 ₪)
- ▶ Domain
 - ▶ Acct number: 9-999
 - ▶ balance: numeric

ประเภทของ Entity

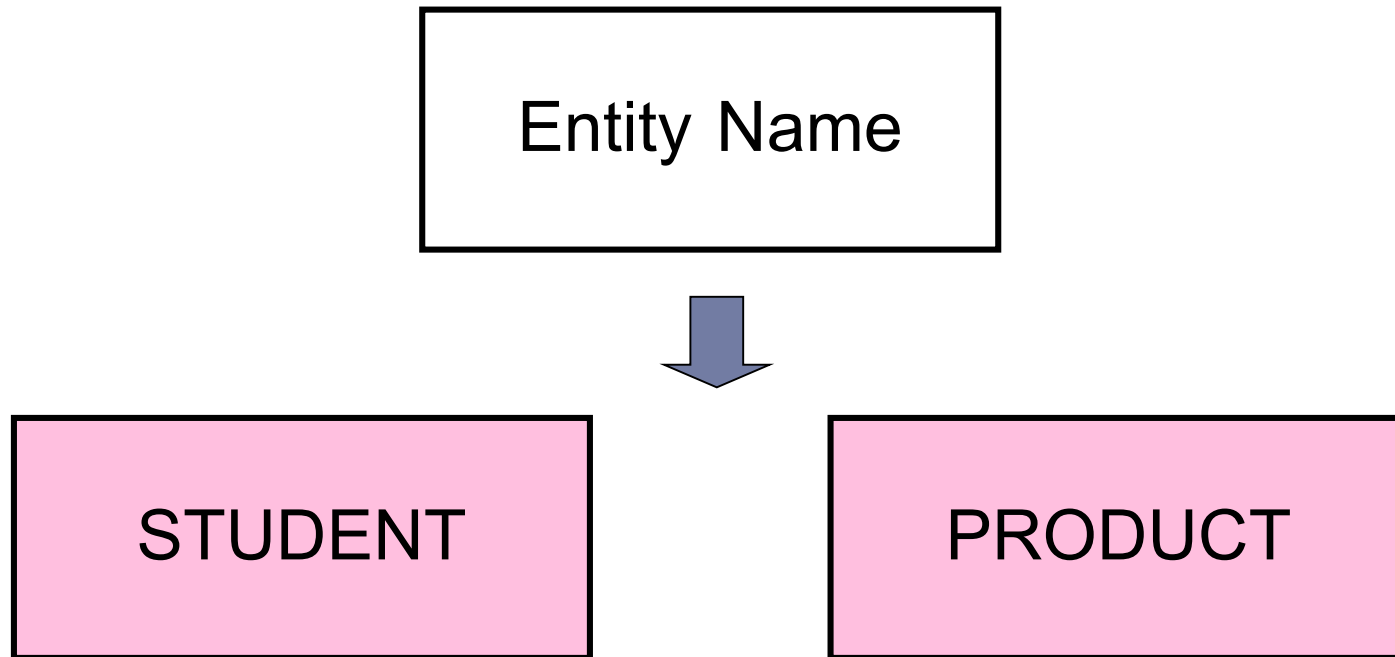
- ▶ แบ่งออกเป็น 2 ประเภท คือ
 1. เอนทิตีแบบปกติ(Strong Entity หรือ Regular Entity)
 2. เอนทิตีแบบอ่อน (Weak Entity)

Strong Entity

- ▶ เป็นเอนทิตีที่เราสามารถกำหนดให้มีในระบบได้อย่างอิสระ ไม่ขึ้นกับข้อมูลจากเอนทิตีอื่น เช่น นักศึกษา , พนักงาน , รถ และ วิชาเรียน เป็นต้น
- ▶ ข้อมูลในเอนทิตีแบบปกติมีลักษณะเป็นเอกลักษณ์(Unique) นั่นคือ สามารถเลือกข้อมูลบางแอททริบิวต์ที่อยู่ในเอนทิตีนั้นๆ มาเป็นกุญแจหลัก(Primary Key) ของเอนทิตีนั้น เพื่อจำแนกข้อมูลแต่ละรายการได้

Strong Entity (cont.)

- ▶ สัญลักษณ์ที่ใช้แทน เอนทิตีแบบปกติ

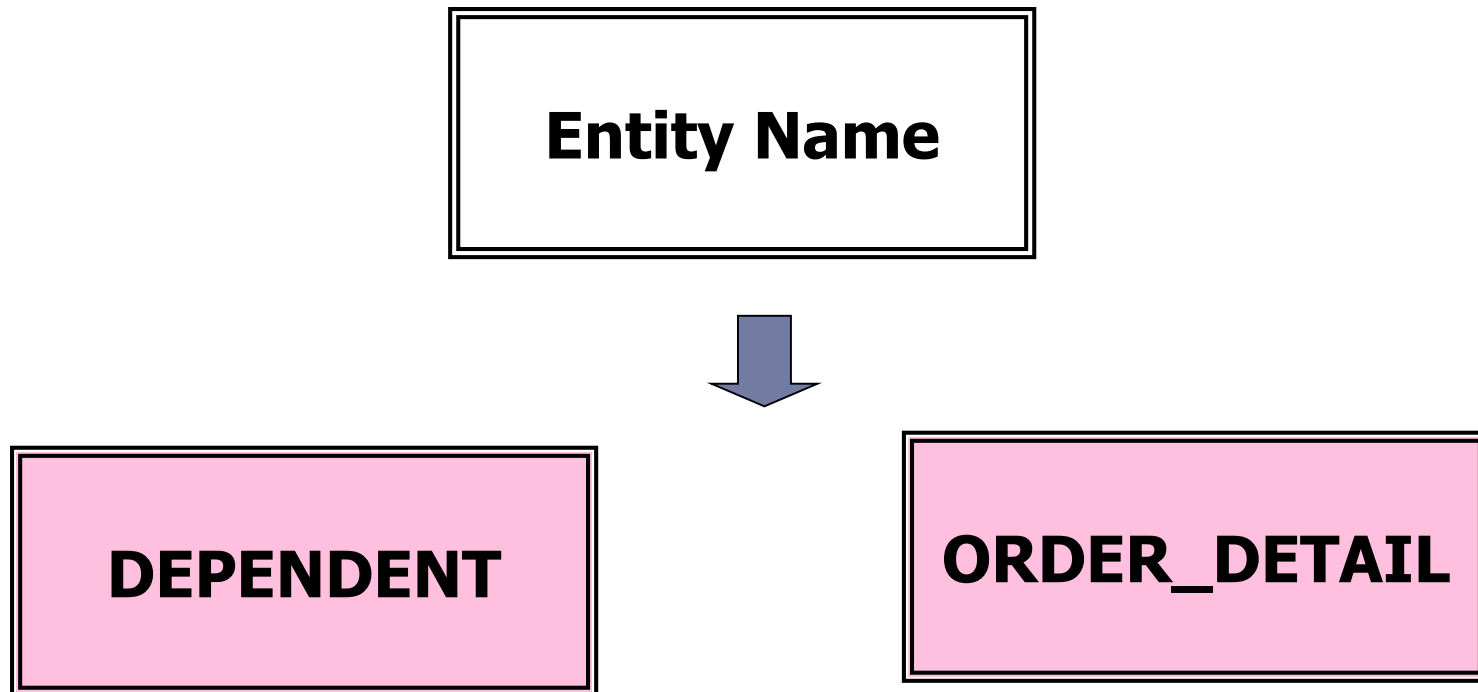


Weak Entity

- ▶ เอนทิตีแบบอ่อนเป็นเอนทิตีที่ขึ้นต่อข้อมูลในเอนทิตีอื่น ๆ บางเอนทิตี กล่าวคือ เอนทิตีแบบอ่อนจะไม่มีข้อมูลในฐานะข้อมูล หากไม่มีข้อมูลในอีกเอนทิตีหนึ่ง (เรียกว่า Owner Entity) เกิดขึ้นในระบบก่อน เรียกว่า “การขึ้นต่อกันเชิงปรากฏ (Existence Dependency)”
- ▶ เอนทิตีแบบอ่อนจะไม่สามารถกำหนดคีย์หลักโดยใช้แอททริบิวต์ในเอนทิตีเพียงลำพัง แต่ต้องอาศัยคีย์หลักจาก Owner Entity มาประกอบกันกับแอททริบิวต์ที่เป็นส่วนประกอบของคีย์

Weak Entity (cont.)

- ▶ สัญลักษณ์ที่ใช้แทน เอนทิตีแบบอ่อน



ตัวอย่างเอนทิตี



รหัส	ชื่อ	นามสกุล	ที่อยู่	ชื่อ	นามสกุล	ความสัมพันธ์
S1	วนิดา	สุขสันต์	อุดรธานี	สุดาวัลย์	สุขสันต์	บุตรสาว
S2	สมชาย	รักดี	อุดรธานี	วิภา	รักดี	ภรรยา
S3	จริงใจ	รักชีพ	กรุงเทพฯ	จันจิรา	รักชีพ	บุตรสาว
S4	สถาพร	ช่วงโชติ	เชียงใหม่	สมชาย	สุขสม	บิดา
S5	จิราพร	แก้วมณี	ภูเก็ต	สมชาย	สุขสม	บิดา
S6	ลินดา	ใจอ่อน	กรุงเทพฯ	สมชาย	แก้วมณี	สามี
S7	จิระศักดิ์	ปานพุ่ม	ระยอง	นิโคล	ปานพุ่ม	ภรรยา

Attribute

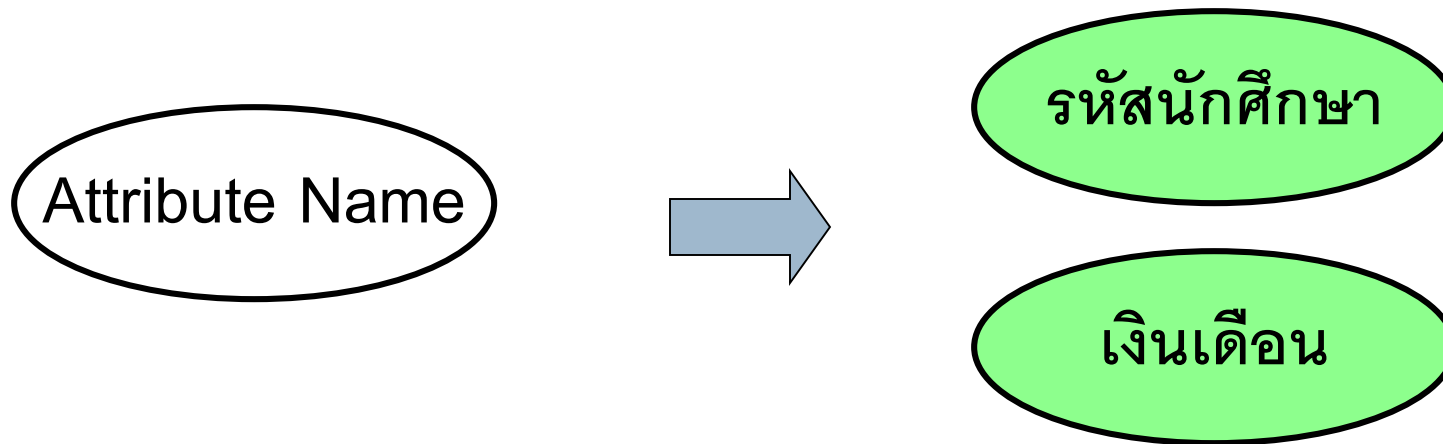
- ▶ แอททริบิวต์ หมายถึง คุณสมบัติ หรือ คุณลักษณะ ที่ใช้อธิบายรายละเอียดของเอนทิตี และ ความสัมพันธ์
- ▶ ตัวอย่าง เอนทิตี “**นักศึกษา**” ประกอบด้วยข้อมูล “รหัสนักศึกษา , ชื่อ , นามสกุล , ที่อยู่ , วันเกิด , อายุ , เกรดเฉลี่ยสะสม”
- ▶ ตัวอย่าง เอนทิตี “**สินค้า**” ประกอบด้วยข้อมูล “รหัสสินค้า , ชื่อสินค้า , ราคาต่อหน่วย , จำนวนคงเหลือ”

ประเภท Attribute

- ▶ Simple Attribute หรือ Atomic Attribute
- ▶ Composite Attribute
- ▶ Single-value Attribute
- ▶ Multivalued Attribute
- ▶ Derived Attribute

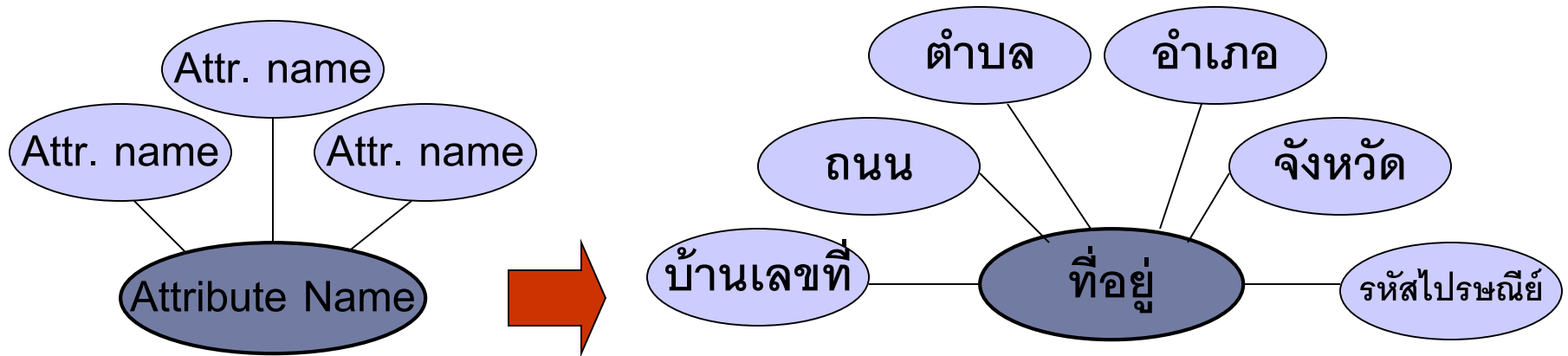
Simple Attribute

- ▶ หมายถึง แอททริบิวต์ที่ไม่มีการแยกออกเป็นแอททริบิวต์ย่อยๆ อีก เช่น รหัสนักศึกษา , เงินเดือน , เกรดเป็นต้น



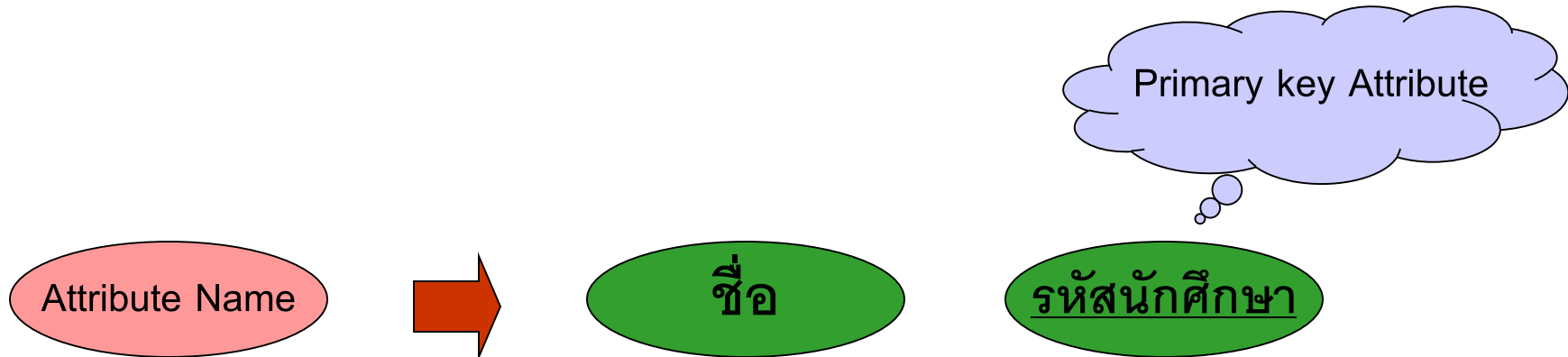
Composite Attribute

- ▶ หมายถึง แอททริบิวต์ที่สามารถแยกออกเป็นแอททริบิวต์ย่อยออกไปอีก เช่น ที่อยู่ ประกอบด้วย บ้านเลขที่ , ถนน , ตำบล , อำเภอ , จังหวัด , รหัสไปรษณีย์



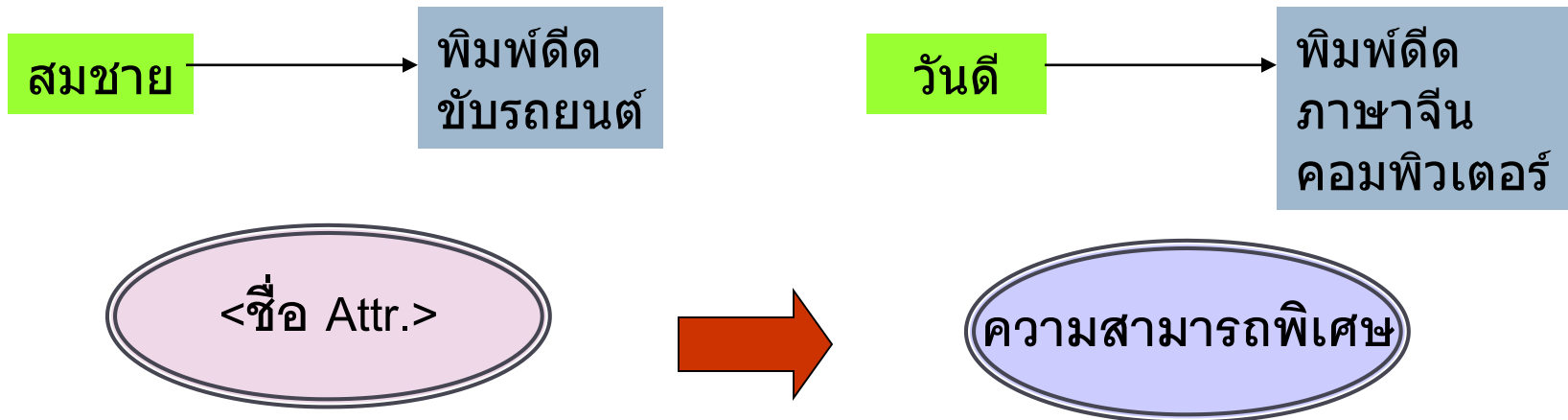
Single value Attribute

- ▶ หมายถึง แอททริบิวต์ที่มีค่าข้อมูลได้เพียงค่าเดียวในแต่ละ แอททริบิวต์ สำหรับแต่ละรายการข้อมูล
- ▶ ถ้ากำหนดให้เป็นกุญแจหลัก(Primary key) ให้ขีดเส้นใต้ที่บ่งชี้ชื่อแอททริบิวต์



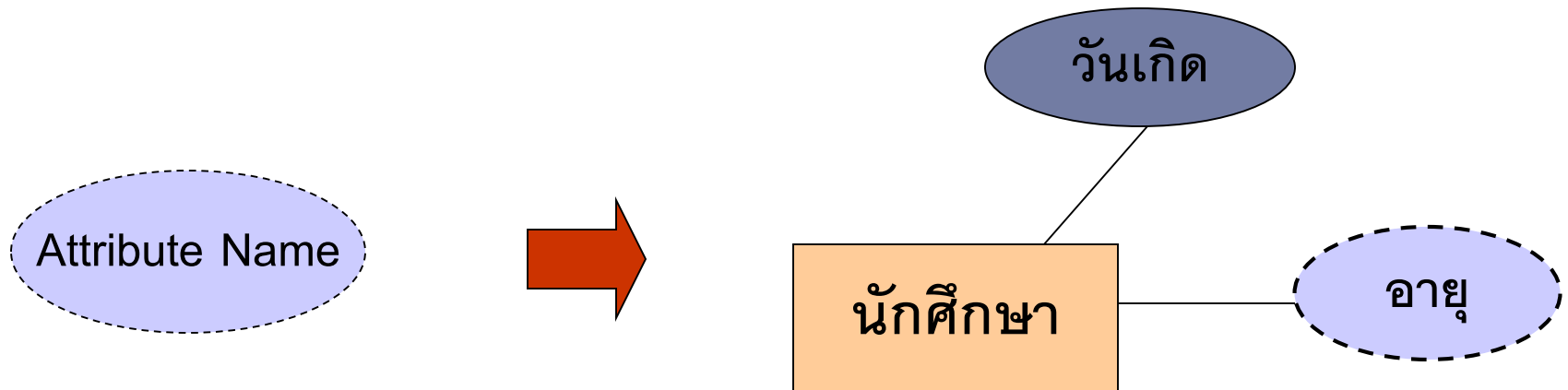
Multivalued Attribute

- ▶ หมายถึง แอททริบิวต์ที่สามารถกำหนดค่าข้อมูลได้มากกว่า 1 ค่า สำหรับข้อมูลแต่ละรายการ
- ▶ เช่น พนักงานแต่ละคนสามารถมีความสามารถพิเศษได้หลายอย่าง

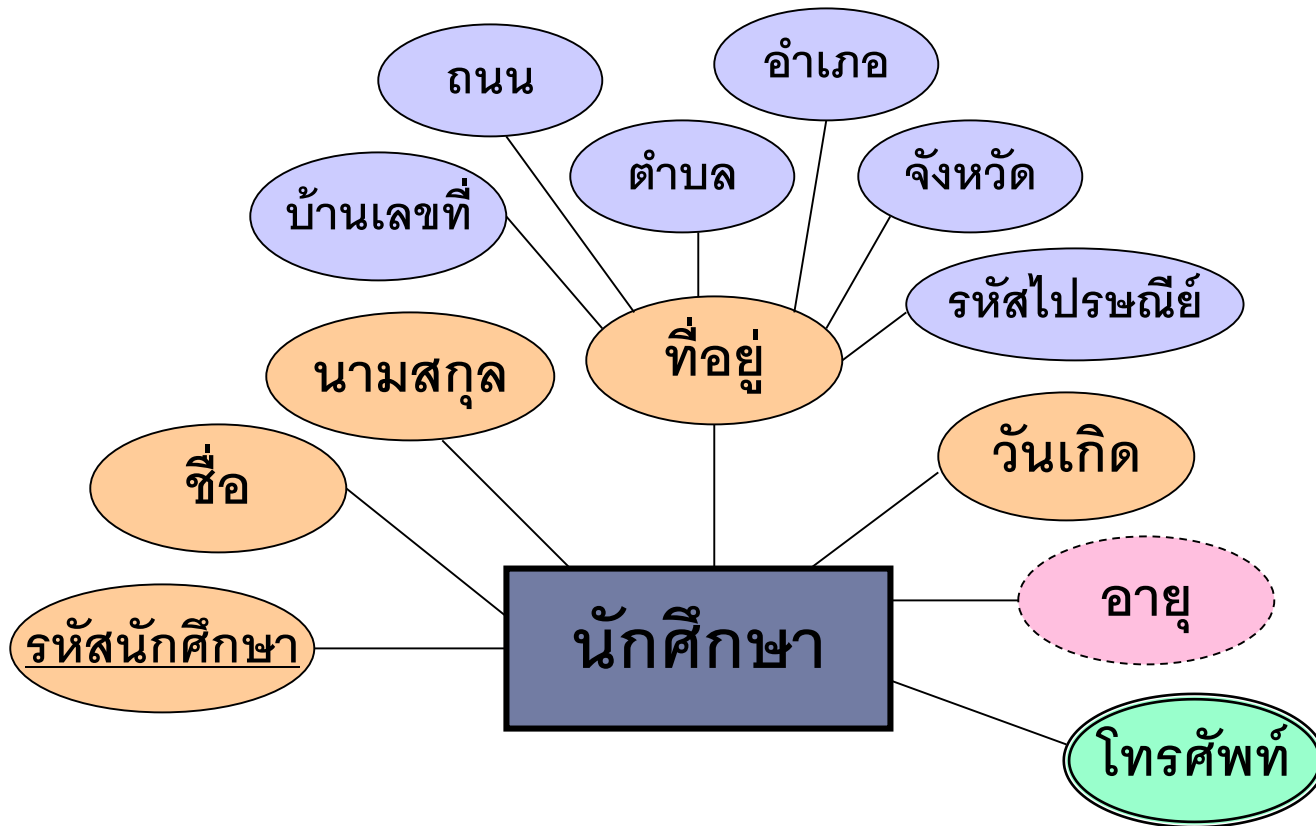


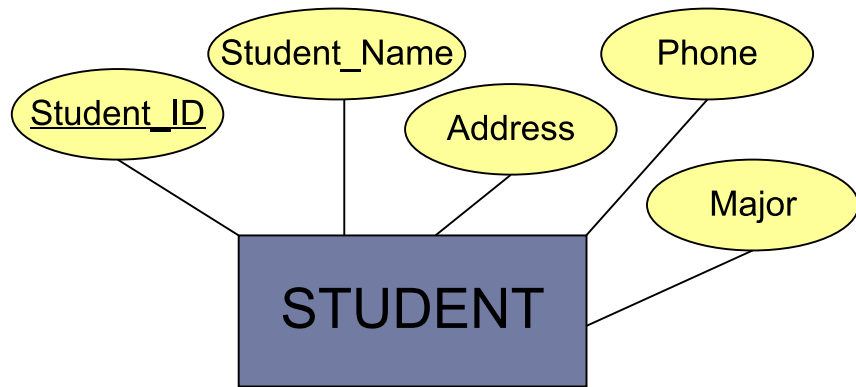
Derived Attribute

- ▶ หมายถึง แอททริบิวต์ที่ค่าของแอททริบิวต์ประกอบขึ้นหรือคำนวณค่าจากค่าของแอททริบิวต์อื่น เช่น อายุ คำนวณจาก วันเกิด



Entity & Attribute





```

CREATE TABLE STUDENT
( Student_ID      CHAR(11) NOT NULL UNIQUE,
  Student_Name    VARCHAR(50) ,
  Address         VARCHAR(100),
  Phone           VARCHAR(10) ,
  Major           VARCHAR(20) ,
  PRIMARY KEY(Student_ID)
);
  
```

Student_ID : 63812489101
 Student_Name : วันดี ใจสะอาด
 Address : 45/2 ถ.อุดรดุขฎฐฎี ต.
 หมากแฉัง อ.เมือง จ.อุดรธานี
 41000
 Phone : 042211563
 Major : เคมี

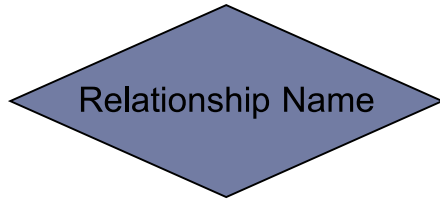
Student_ID : 63856245231
 Student_Name : สมศักดิ์ ใจดี
 Address : 50 ถ.ทหาร ต.หมากแฉัง
 อ.เมือง จ.อุดรธานี 41000
 Phone : 042244121
 Major : คอมพิวเตอร์

Relationship Sets

- ▶ ความสัมพันธ์ หมายถึง ความสัมพันธ์ที่เชื่อมโยงระหว่างแต่ละเอนทิตีตามเงื่อนไขของระบบงาน เช่นเอนทิตี “อาจารย์” และเอนทิตี “นักศึกษา” มีความสัมพันธ์กันชื่อ “เป็นที่ปรึกษา”

Relationship

▶ สัญลักษณ์แทน ความสัมพันธ์

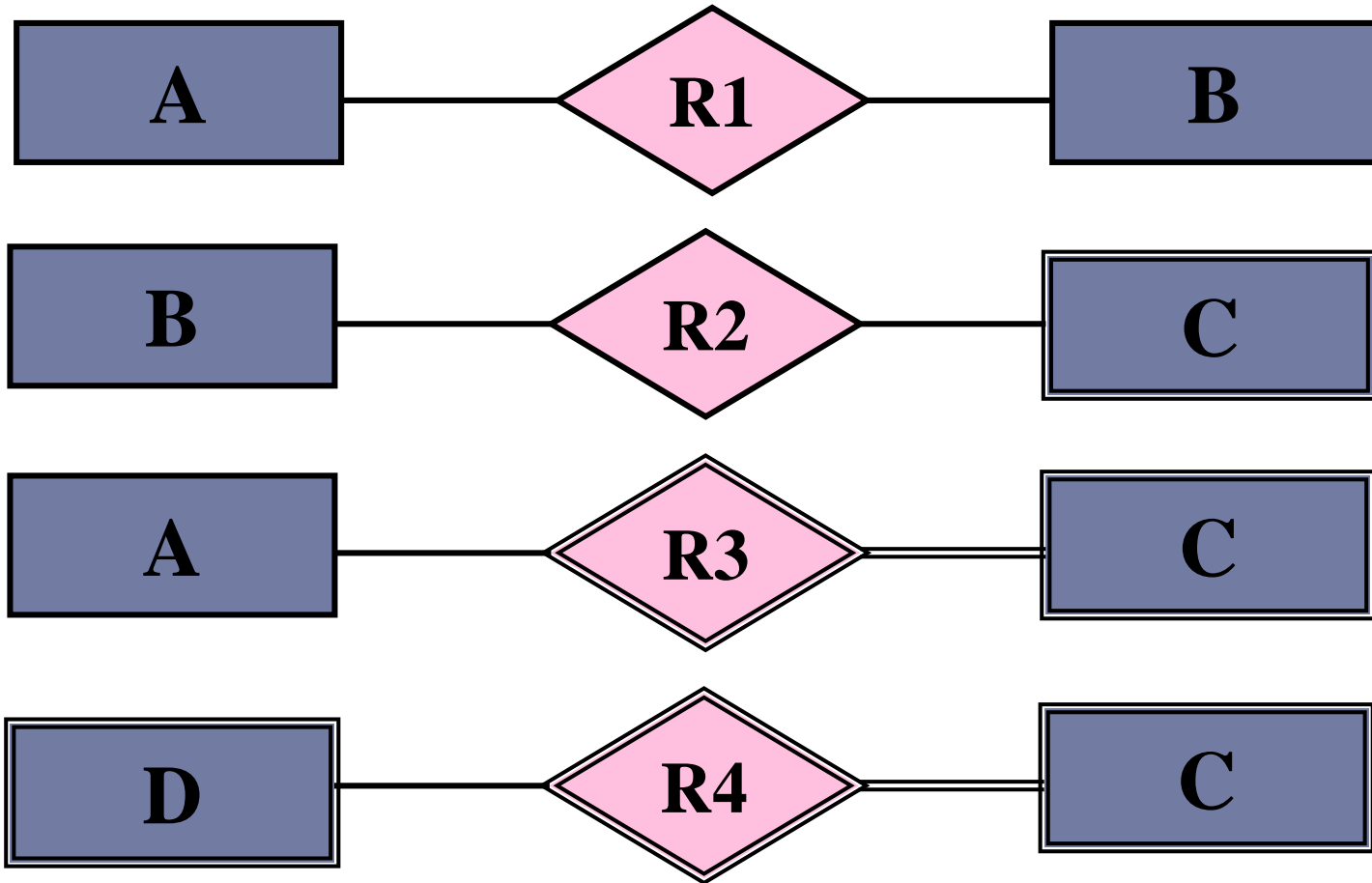


แสดงความสัมพันธ์ระหว่างเอนทิตีที่มี
ความสัมพันธ์กันแบบปกติ



แสดงความสัมพันธ์ระหว่างเอนทิตีแบบอ่อนกับ
เอนทิตีอื่น (Owner Entity) ที่มีความสัมพันธ์กัน
แบบเชิงปรากฏและเชิงระบุ เรียกความสัมพันธ์
นี้ว่า Identifying Relationship

Relationship

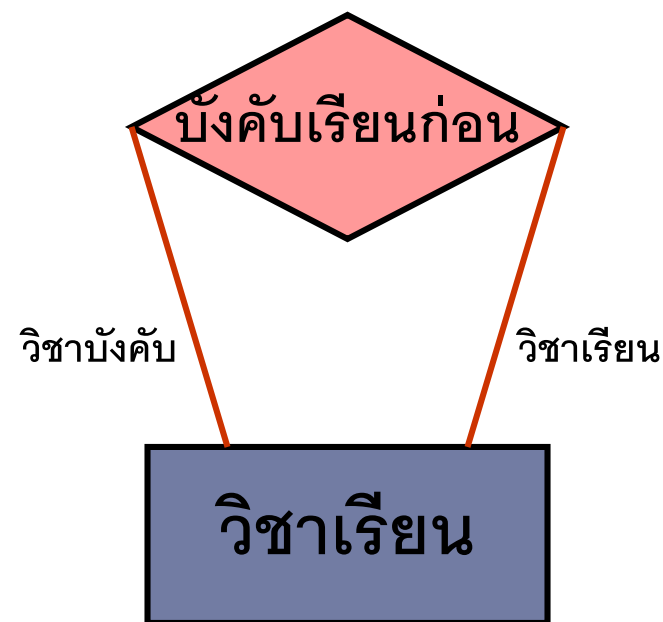
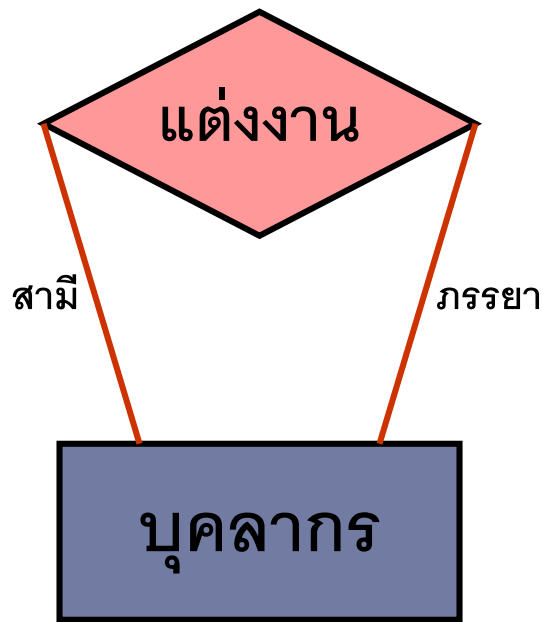
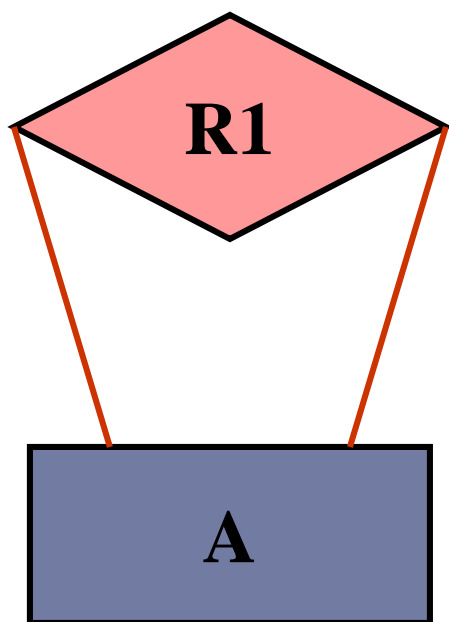


Degree of Relationship Set

- ▶ หมายถึง จำนวนของเอนทิตีที่เข้ามามีส่วนร่วม (Participating) กับความสัมพันธ์ใด ๆ
- ▶ มีความเป็นไปได้ 4 แบบ ดังนี้
 - ▶ Unary Degree
 - ▶ Binary Degree
 - ▶ Ternary
 - ▶ Quaternary

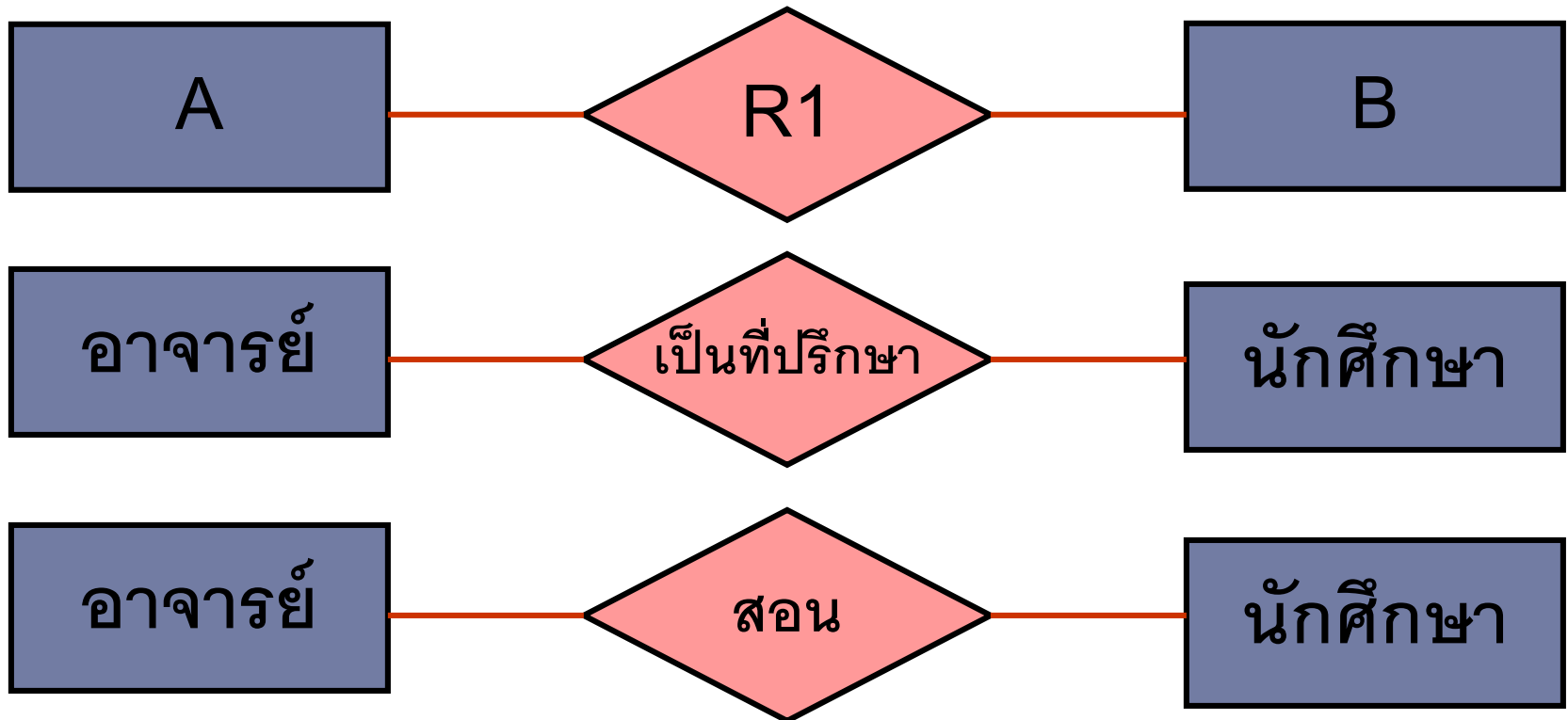
Degree of Relationship : Unary Relationship

- เป็นความสัมพันธ์เกิดขึ้นกับเอนทิตี 1 เอนทิตี โดยสมาชิกในเอนทิตีมีความสัมพันธ์ระหว่างกันเอง (recursive)

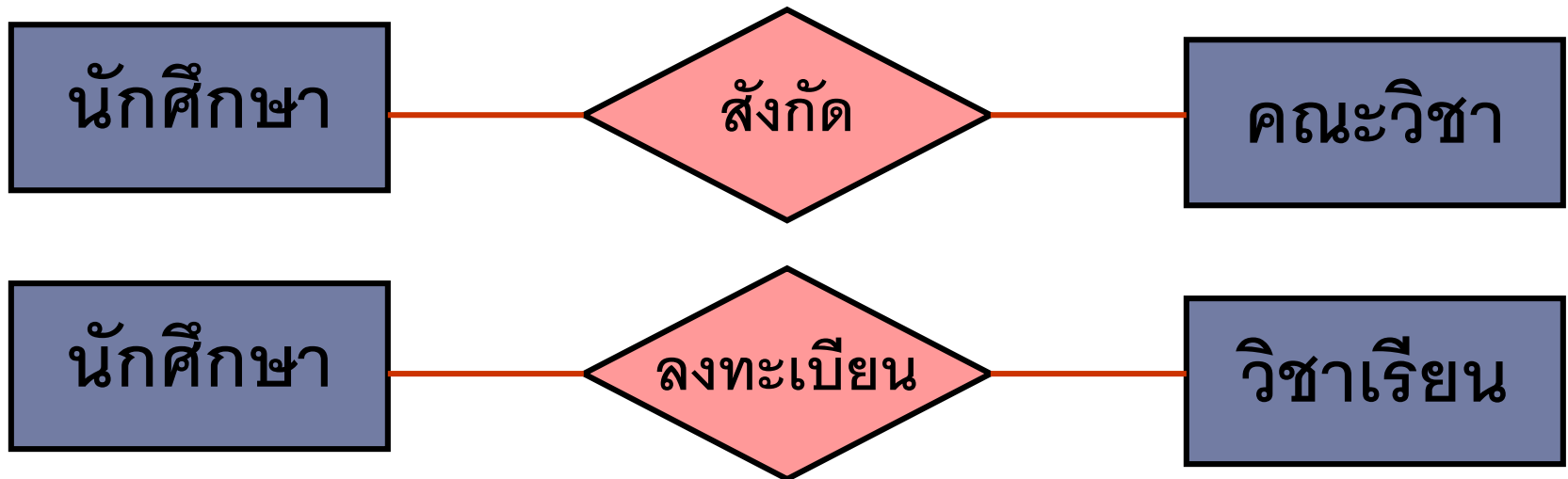


Degree of Relationship : Binary Relationship

- ▶ เป็นความสัมพันธ์ระหว่างสองเอนทิตี

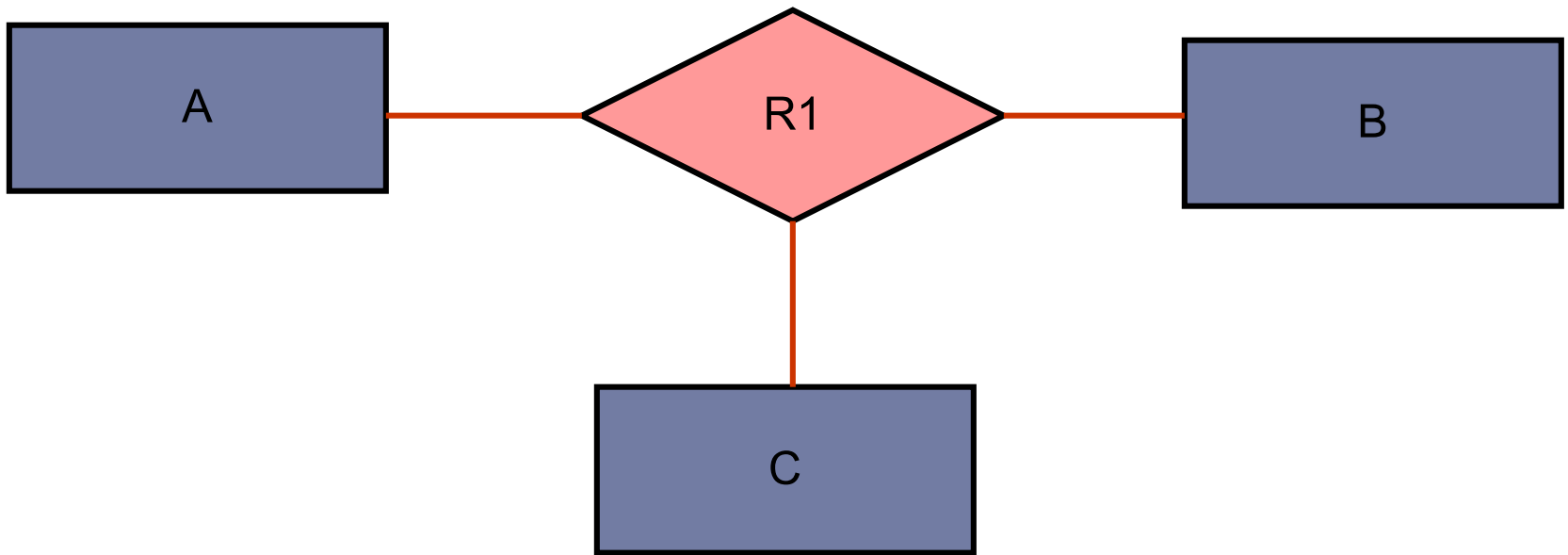


Degree of Relationship : Binary Relationship

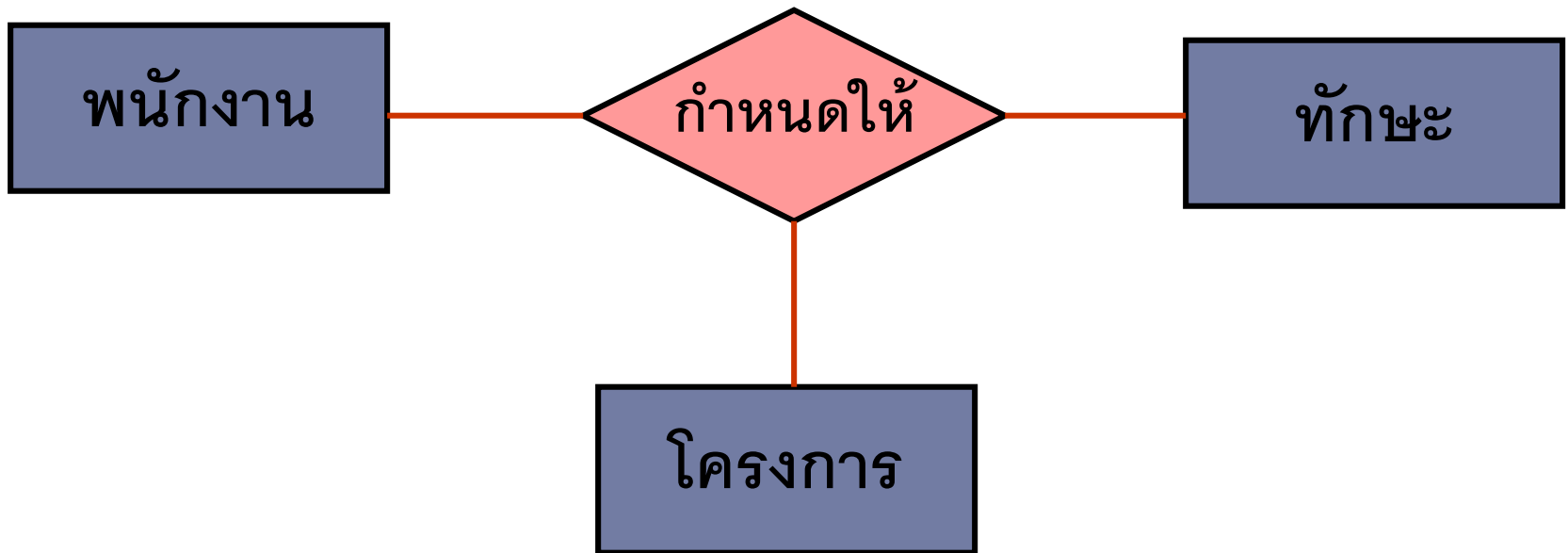


Degree of Relationship : Ternary Relationship

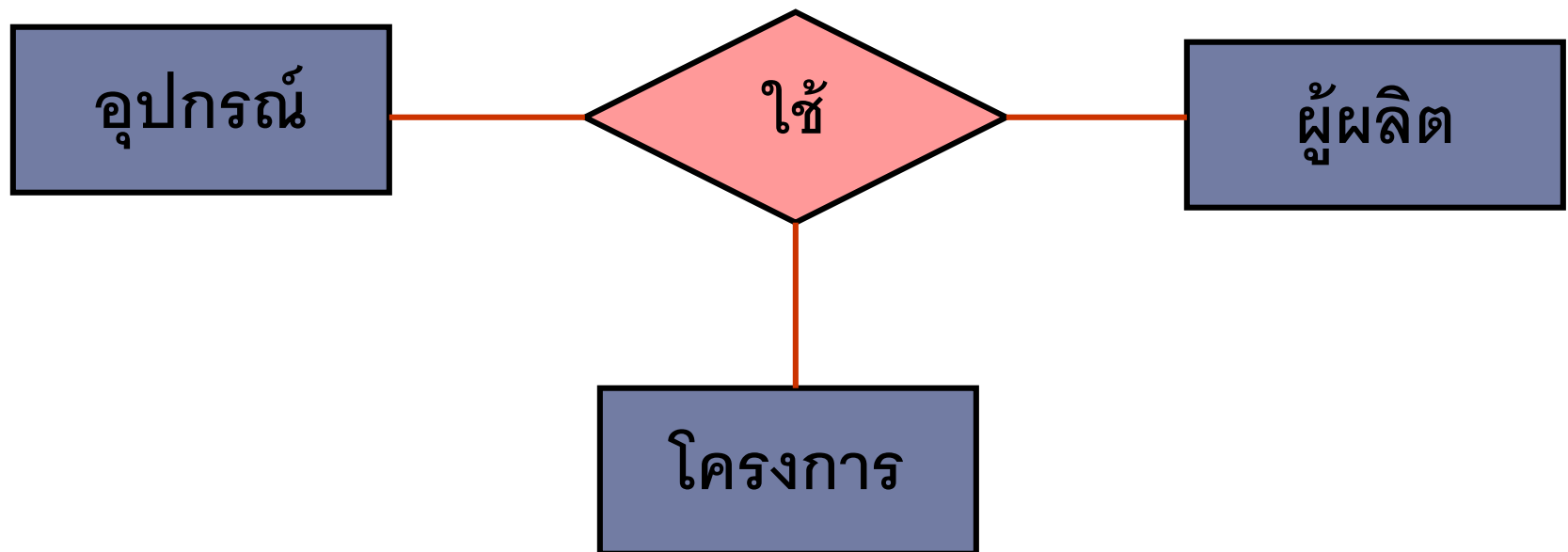
- ▶ เป็นความสัมพันธ์ระหว่างเอนทิตี 3 เอนทิตี



Degree of Relationship : Ternary Relationship



Degree of Relationship : Ternary Relationship



Constraints

- ▶ Mapping Cardinalities (Multiplicity)
 - ▶ One-to-one
 - ▶ One-to-many or many-to-one
 - ▶ Many-to-many
- ▶ Participation Constraints
 - ▶ Total participation
 - ▶ Partial participation

Cardinality Constraints

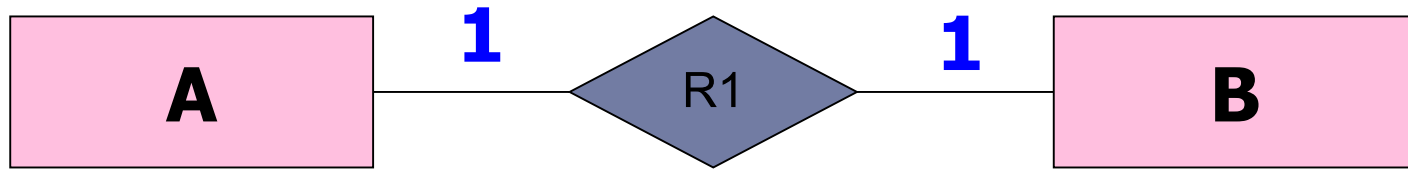
- ▶ หมายถึง การระบุจำนวนสมาชิกของเอนทิตีที่สัมพันธ์กันในแต่ละความสัมพันธ์
- ▶ ประกอบด้วย
 - ▶ ความสัมพันธ์แบบ หนึ่งต่อหนึ่ง(One-to-One Relationship)
 - ▶ ความสัมพันธ์แบบ หนึ่งต่อกลุ่ม(One-to-Many Relationship)
 - ▶ ความสัมพันธ์แบบ กลุ่มต่อกลุ่ม(Many-to-Many Relationship)

1 : 1 Relationship

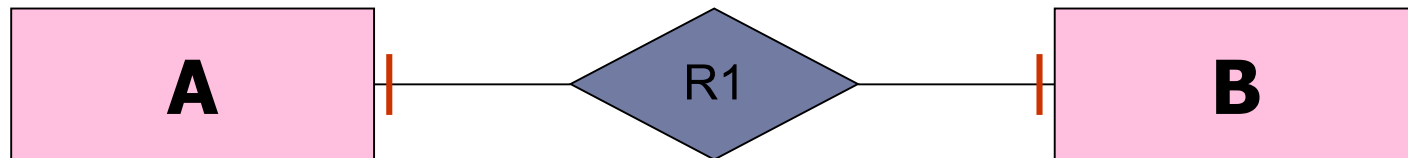
- ▶ เป็นการแสดงความสัมพันธ์ระหว่างข้อมูลของเอนทิตีหนึ่งกับข้อมูลของอีกเอนทิตีหนึ่งเพียง 1 รายการเท่านั้น
- ▶ ตัวอย่างเช่น อาจารย์ 1 คน เป็นคณบดีได้ 1 คณะ ในขณะที่ คณะวิชาแต่ละคณะมีคณบดี ได้ 1 คน/คณะ
- ▶ พนักงาน 1 คนมีที่จอดรถเพียง 1 ที่ และที่จอดรถแต่ละที่สามารถจอดรถโดยพนักงาน 1 คนเท่านั้น
- ▶ พนักงาน 1 คนเป็นผู้จัดการแผนกได้ 1 แผนกและแต่ละแผนกก็มีผู้จัดการได้เพียง 1 คน

1 : 1 Relationship

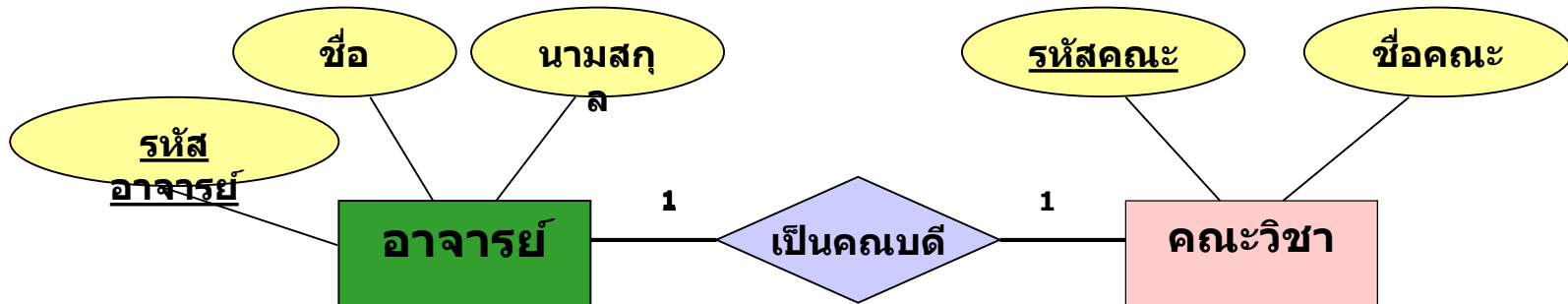
สัญลักษณ์แบบ Chen



สัญลักษณ์แบบ Crow's Foot



1 : 1 Relationship

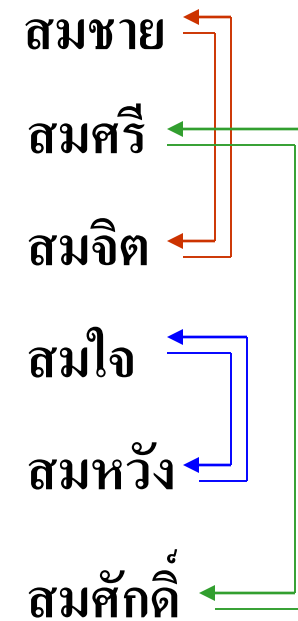
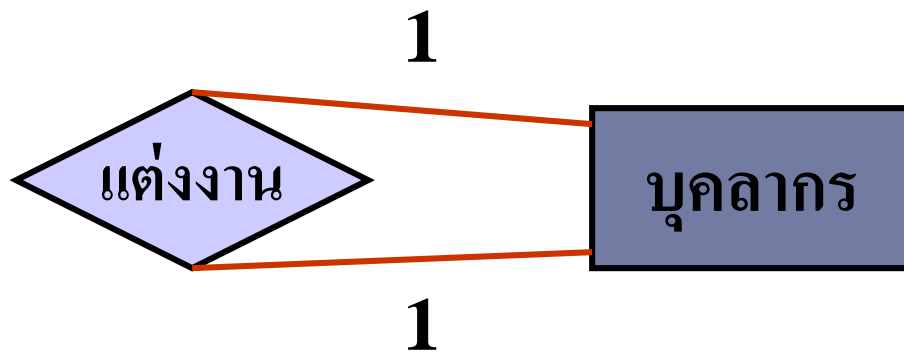


รหัส	ชื่อ	นามสกุล	รหัส	ชื่อคณะ
T1	เรวดี	พิพัฒน์สูงเนิน	F1	วิทยาศาสตร์
T2	คุณาวุฒิ	บุญกว้าง	F2	บริหารธุรกิจ
T3	นิพล	สังสุทธิ	F3	วิศวกรรมศาสตร์
T4	อารีรัตน์	วุฒิเสน	F4	ศิลปกรรมศาสตร์
T5	ปัจจุัย	พวงสุวรรณ	F5	เกษตรศาสตร์

อาจารย์

คณะวิชา

1 : 1 Relationship

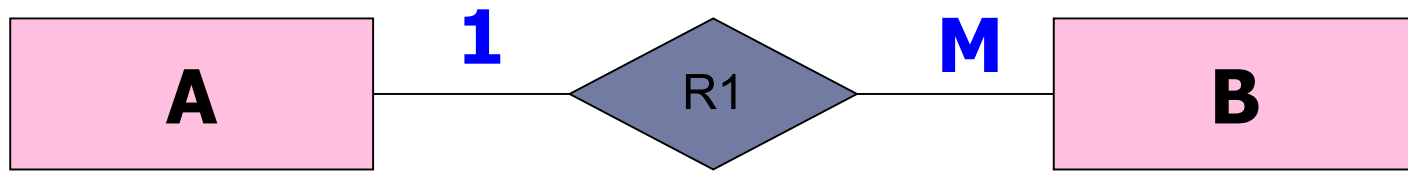


1 : M Relationship

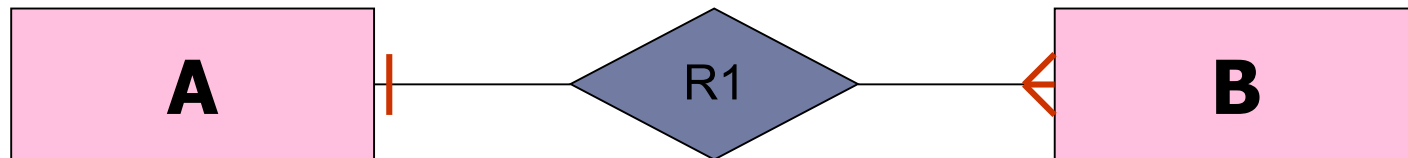
- ▶ เป็นการแสดงความสัมพันธ์ของข้อมูลเอนทิตีหนึ่งว่ามีความสัมพันธ์กับข้อมูลของอีกเอนทิตีหนึ่งมากกว่า 1 รายการ
- ▶ ตัวอย่างเช่น อาจารย์ 1 คน เป็นอาจารย์ที่ปรึกษาให้กับนักศึกษาได้หลายคน ในขณะที่นักศึกษาแต่ละคนมีอาจารย์ที่ปรึกษาเพียง 1 คน
- ▶ แผนก 1 แผนกมีพนักงานสังกัดได้หลายคน ในขณะที่พนักงานแต่ละคนสามารถสังกัดแผนกได้เพียง 1 แผนก

1 : N Relationship

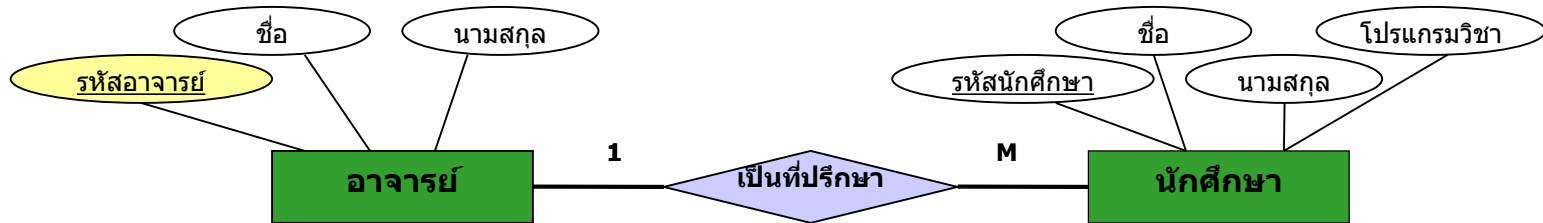
สัญลักษณ์แบบ Chen



สัญลักษณ์แบบ Crow's Foot



1 : N Relationship



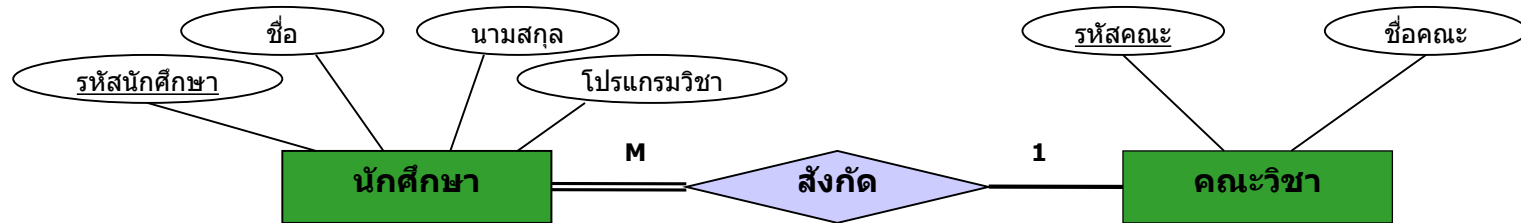
รหัส	ชื่อ	นามสกุล
T1	เรวดี	ทองอ่อน
T2	พนารัตน์	ศรีเชษฐา
T3	นิพล	สังสุทธิ
T4	อารีรัตน์	บุญทอง
T5	ปัจจุัย	พวงสุวรรณ

อาจารย์

รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จริงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
s6	ลินดา	ใจอ่อน	โยธา
s7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

นักศึกษา

1 : N Relationship



รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จรงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
S6	ลินดา	ใจอ่อน	โยธา
S7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

รหัส	ชื่อคณะ
F1	วิทยาศาสตร์
F2	บริหารธุรกิจ
F3	วิศวกรรมศาสตร์
F4	ศิลปกรรมศาสตร์
F5	เกษตรศาสตร์

คณะวิชา

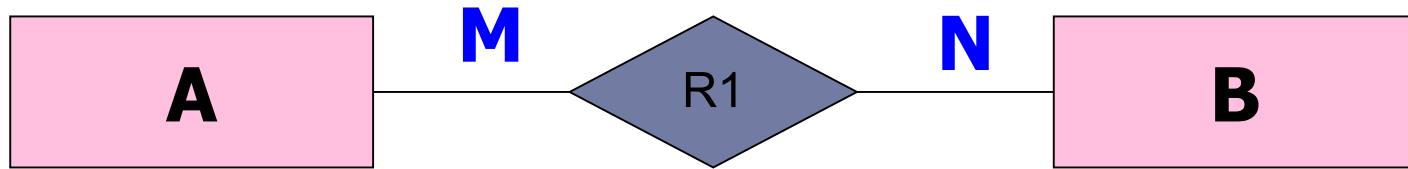
นักศึกษา

M : N Relationship

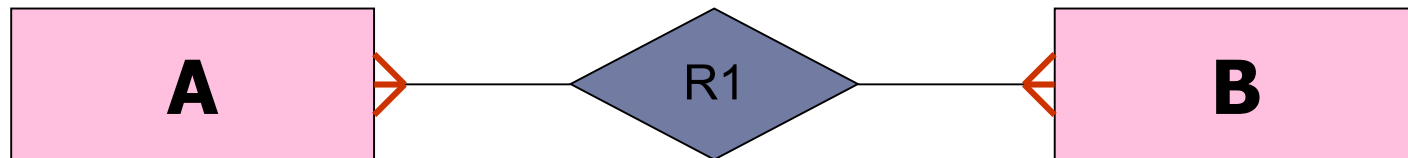
- ▶ เป็นการแสดงความสัมพันธ์ของข้อมูลระหว่างเอนทิตีแบบกลุ่มต่อกลุ่ม
- ▶ ตัวอย่างเช่น นักศึกษา 1 คนสามารถลงทะเบียนเรียนได้หลายวิชา ในขณะที่แต่ละวิชา ก็สามารถมีนักศึกษามาลงทะเบียนเรียนได้หลายคน

M : N Relationship

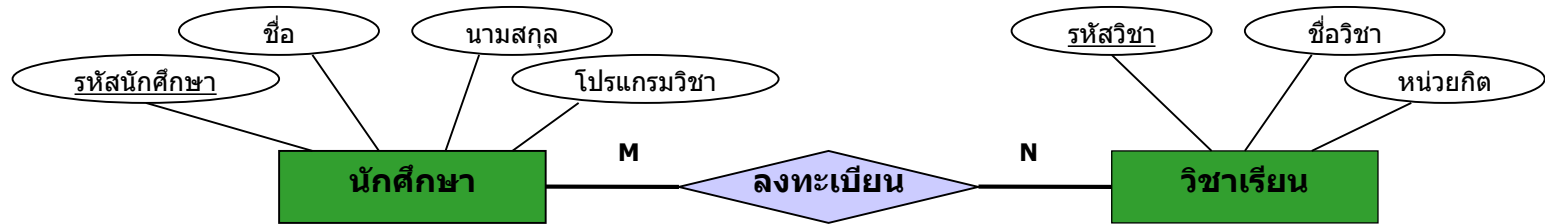
สัญลักษณ์แบบ Chen



สัญลักษณ์แบบ Crow's Foot



M : N Relationship



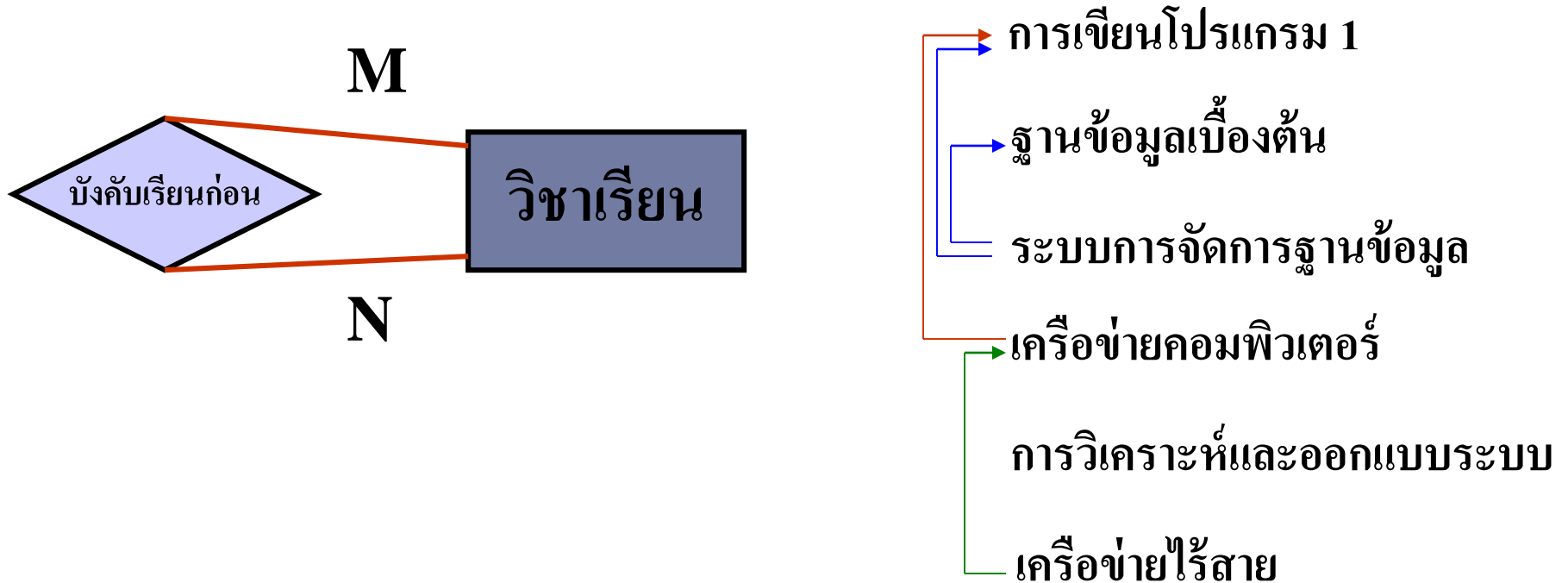
รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จริงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
S6	ลินดา	ใจอ่อน	โยธา
S7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

รหัส	ชื่อวิชา	หน่วยกิต
Su1	ฐานข้อมูล	3
Su2	สื่อสารข้อมูล	3
Su3	แคลคูลัส	3
Su4	บัญชี1	3
Su5	อังกฤษ1	2

วิชาเรียน

นักศึกษา

M : N Relationship

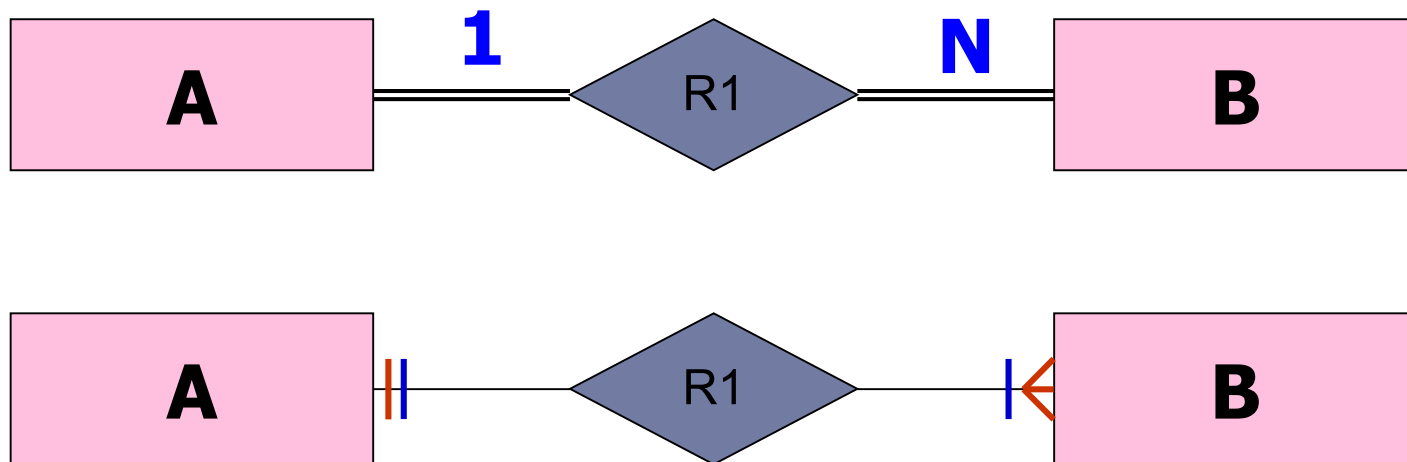


Participation Constraints

- ▶ หมายถึง ข้อกำหนดในการมีส่วนร่วมในความสัมพันธ์ของสมาชิกในเอนทิตี
- ▶ ประกอบด้วย 2 ข้อกำหนด คือ
 - ▶ การมีส่วนร่วมแบบทั้งหมด(Total Participation หรือ Mandatory)
 - ▶ การมีส่วนร่วมแบบบางส่วน(Partial Participation หรือ Optional)

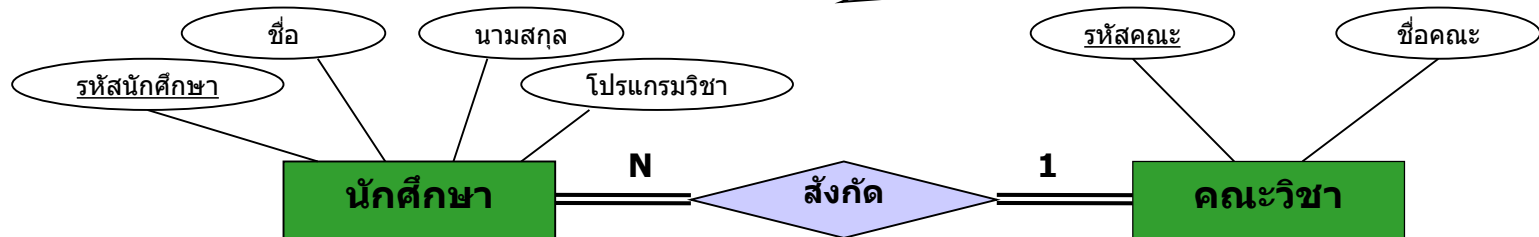
Total Participation

- ▶ หมายถึง ทุกรายการในเอนทิตี A ต้องมีความสัมพันธ์แบบ R1 กับสมาชิกเอนทิตี B เสมอ



Total Participation

นักศึกษา**ทุกคน**ต้องสังกัดคณะวิชาได้คนละ 1 คณะ
และแต่ละคณะ**ต้อง**มีนักศึกษามาสังกัดอย่างน้อย 1 คนหรือหลายคน



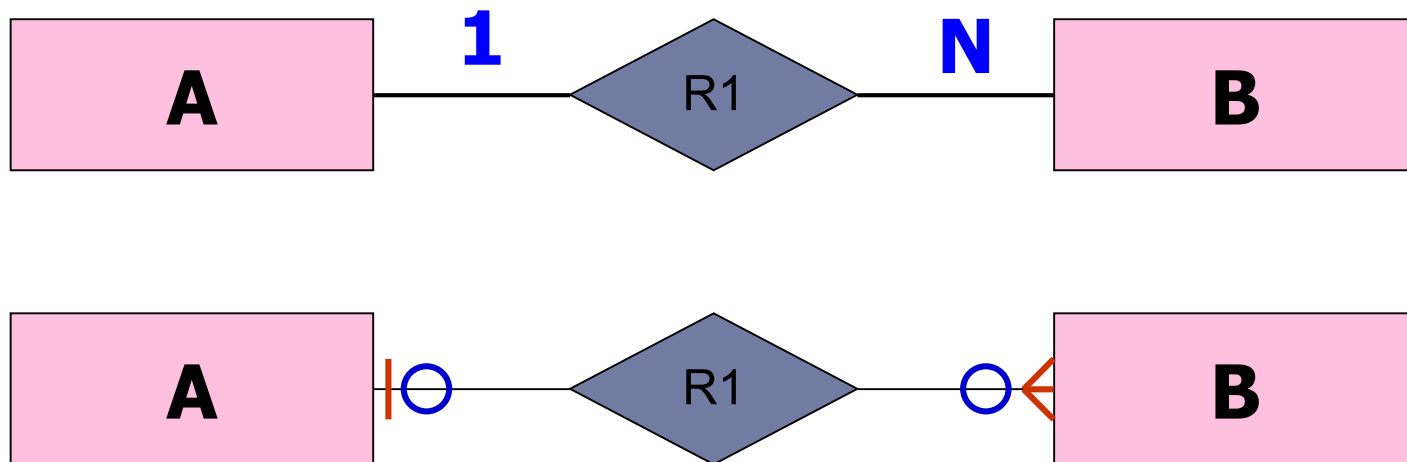
รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จริงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
S6	ลินดา	ใจอ่อน	โยธา
S7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

รหัส	ชื่อคณะ
F1	วิทยาศาสตร์
F2	บริหารธุรกิจ
F3	วิศวกรรมศาสตร์
F4	ศิลปกรรมศาสตร์
F5	เกษตรศาสตร์

คณะวิชา

Partial Participation

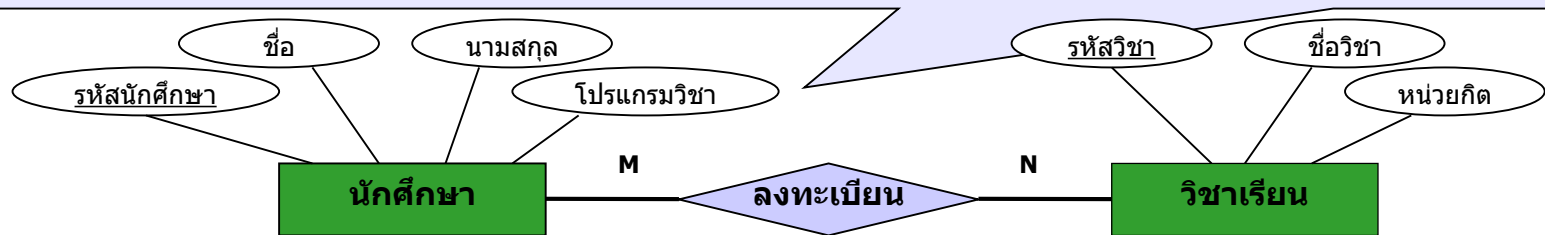
- ▶ หมายถึง รายการในเอนทิตี A อาจจะมีความสัมพันธ์แบบ R1 กับสมาชิกเอนทิตี B หรือไม่ก็ได้



Partial Participation

นักศึกษา(บางคน) สามารถลงทะเบียนเรียนหรือไม่ลงทะเบียนก็ได้
และสามารถลงทะเบียนเรียนได้หลายวิชา แต่ละวิชาก็อนุญาตให้ลงทะเบียนได้มากกว่า 1

คน



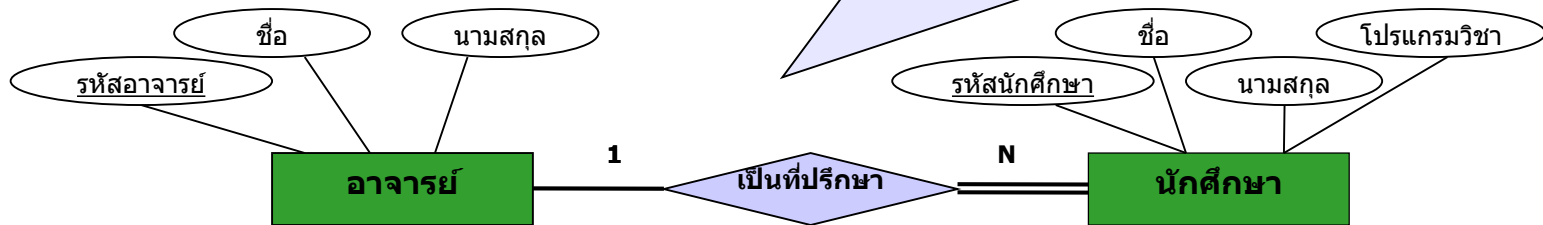
รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จริงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
S6	ลินดา	ใจอ่อน	โยธา
S7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

รหัส	ชื่อวิชา	หน่วยกิต
Su1	ฐานข้อมูล	3
Su2	สื่อสารข้อมูล	3
Su3	แคลคูลัส	3
Su4	บัญชี1	3
Su5	อังกฤษ1	2

วิชาเรียน

Participation

อาจารย์(บางคน) เป็นที่ปรึกษาให้แก่นักศึกษาได้หลายคน แต่ในขณะที่นักศึกษา**ทุกคน** ต้องมีอาจารย์ที่ปรึกษา 1 คน



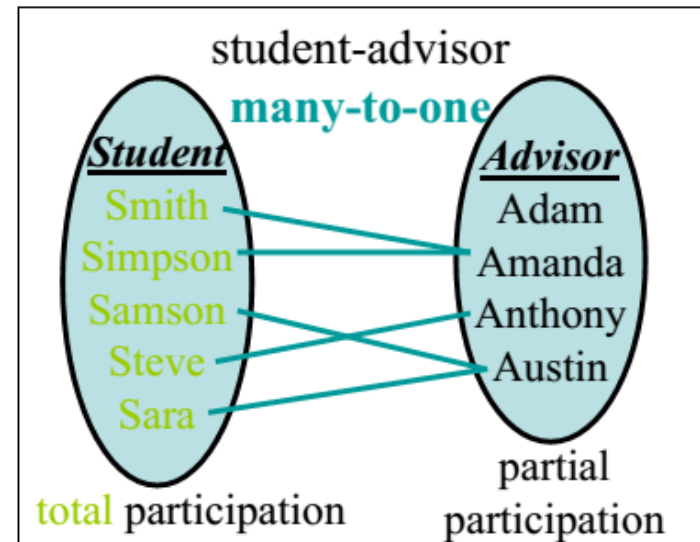
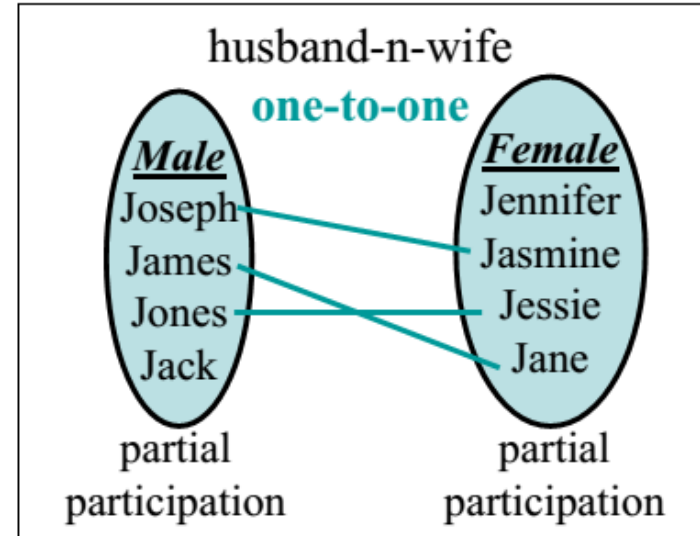
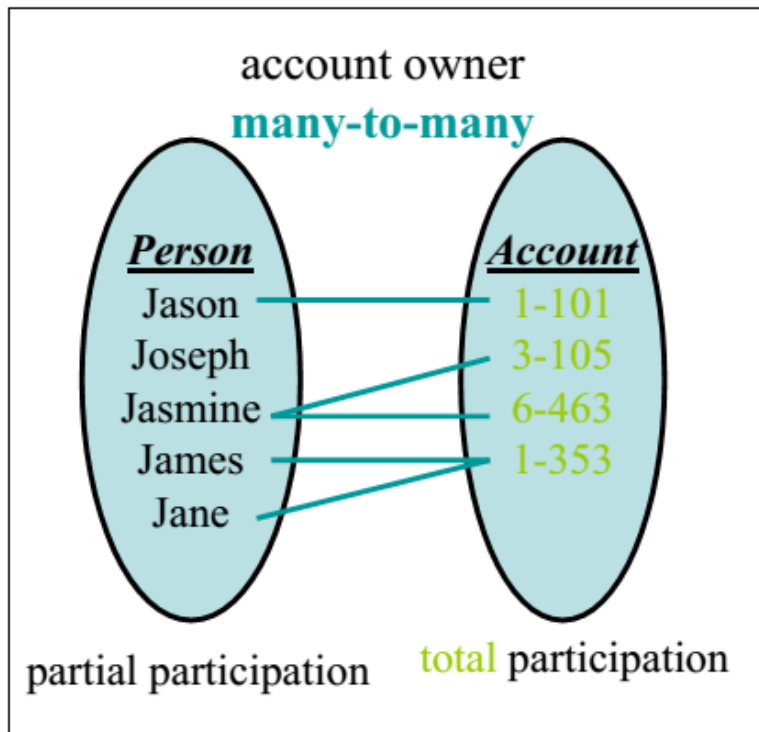
รหัส	ชื่อ	นามสกุล
T1	เรวดี	ทองอ่อน
T2	พนารัตน์	ศรีเชษฐา
T3	นิพล	สังสุทธิ
T4	อารีรัตน์	บุญทอง
T5	ปัจจุัย	พวงสุวรรณ

อาจารย์

รหัส	ชื่อ	นามสกุล	โปรแกรมวิชา
s1	วนิดา	สุขสันต์	คอมพิวเตอร์
s2	สมชาย	รักดี	โยธา
s3	จริงใจ	รักชีพ	คอมพิวเตอร์
s4	สถาพร	ช่วงโชติ	สัตวบาล
s5	จิราพร	แก้วมณี	โยธา
S6	ลินดา	ใจอ่อน	โยธา
S7	ชาติชาย	ปานพุ่ม	คอมพิวเตอร์

นักศึกษา

Multiplicity and Participation Constraints



End of part 1

