

OOAD



www.siam2dev.com

วิชา การวิเคราะห์และออกแบบระบบเชิงวัตถุ (4122506)

Lec02_4_แอสโซซิเอชันแอปแตรกชัน
(Association Abstraction)

By

ผู้ช่วยศาสตราจารย์ ดร. นัฐพงศ์ ส่องเนียม
Asst. Prof. Dr. Nattapong Songneam
เว็บไซต์ :: <http://www.siam2dev.com>
อีเมล :: siam2dev@gmail.com



Abstractions

Classification

Is member of
Is member of
Is member of

Generalization/ Specialization

Is kind of
Is kind of

Aggregation

Is part of
Is part of
Is part of

Association

Is relate to
Is relate to

ผู้ช่วยศาสตราจารย์ ดร. นัฐพงศ์ ส่งเนียม

- Website : <http://www.siam2dev.com>
- E-mail1 : dr.nattapong_s@hotmail.com
- E-mail2 : siam2dev@hotmail.com
- E-mail3 : xnattapong@hotmail.com
- Facebook : siam2dev@hotmail.com

Lec02_4_Association Abstraction

- **Objectives (วัตถุประสงค์)**
 1. เพื่อให้สามารถสร้างความสัมพันธ์ระหว่าง class ต่าง ๆ ใน problem domain โดยใช้ Association Abstraction ได้
 2. เพื่อให้สามารถใส่ Cardinality ของความสัมพันธ์ในเชิง Association ได้อย่างถูกต้อง
 3. สามารถเขียนแผนภาพเพื่อแสดงความสัมพันธ์ระหว่างคลาส ได้

Abstraction

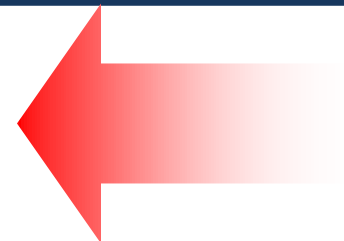
Classification Abstraction

Aggregation Abstraction

Generalization Abstraction

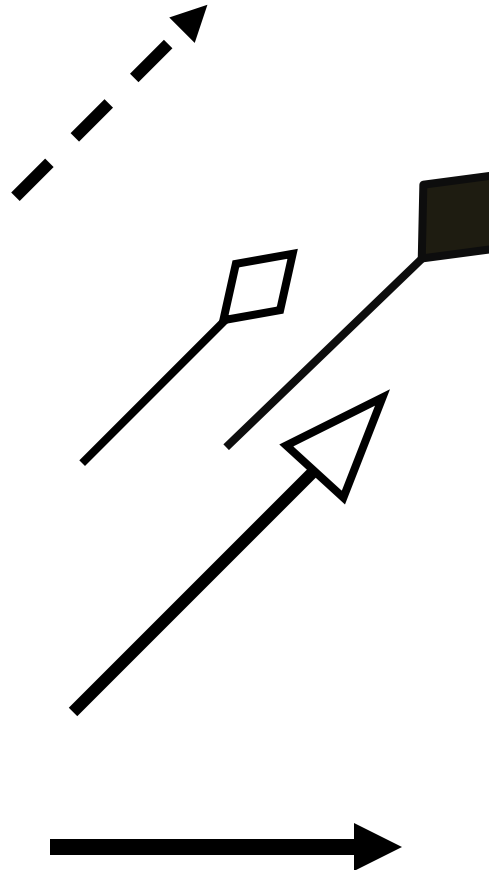
Association Abstraction

Problem Domain
Concept



Reviews

- **Is Member Of**
- **Is Part Of**
- **Is Kind Of**
- **Is related to**



วัตถุเป็นสมาชิกของ
คลาส

คลาสมีส่วนประกอบ
ย่อยอะไรบ้าง

คลาสรวมกับคลาสอื่น
ได้หรือไม่

คลาสเกี่ยวข้องกับ
คลาสอื่นอย่างไร

การระบุวัตถุ ต้องระบุให้ชัดเจน

- ปากกา
- รถยนต์
- โทรศัพท์มือถือ



- **คลาส**

- วัตถุ



- เช่น ปากกา สีแดง ของ อ.ดร. นัฐ
- รถยนต์ ยี่ห้อ Toyota รุ่น camry สีแดง เลขทะเบียน 4ก 7877

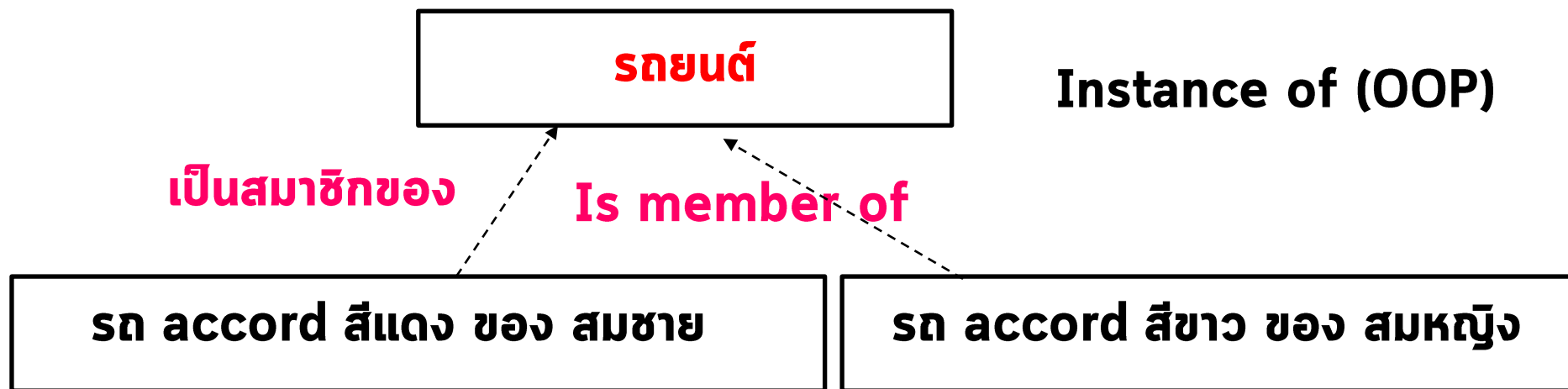
จงแปลงข้อความต่อไปนี้ให้เป็นวัตถุ

- คอมพิวเตอร์? **คอมพิวเตอร์ ASUS สีแดงของนายสมชาย** → เป็นวัตถุ
- โทรศัพท์มือถือ ...?
- ลูกค้า ...?
- คนใช้ ...? นายสมชาย
- อาจารย์ ...?
- นักเรียน ... **นาย สมชาย รหัส 650111101**
- **แมว**
- **ปากกา**

จงระบุว่าข้อใดเป็นคลาส และข้อใดเป็นวัตถุ

คอมพิวเตอร์ , คอมพิวเตอร์ ASUS สีแดงของนายสมชาย
โทรศัพท์มือถือ , ลูกค้า, คนใช้ , นายสมชาย, อาจารย์
นักเรียน , นาย สมชาย รหัส 650111101
แมว , ปากกา

คลาสสี่เฟเชียนแอบแทรกชัน



วัตถุจะมีคุณสมบัติที่เป็น Identity คือ อัตลักษณ์ ที่มีแค่ชิ้นเดียวเท่านั้น

ถ้าเราระบุ รถ accord ไม่สามารถระบุได้ว่าเป็นของใคร ดังนั้นไม่เป็น identity

แอกกรีเกชันแอบแทรคชัน



ห้องเรียน

Optional Component

องค์ประกอบที่ ไม่จำเป็น

เก้าอี้

Required Component

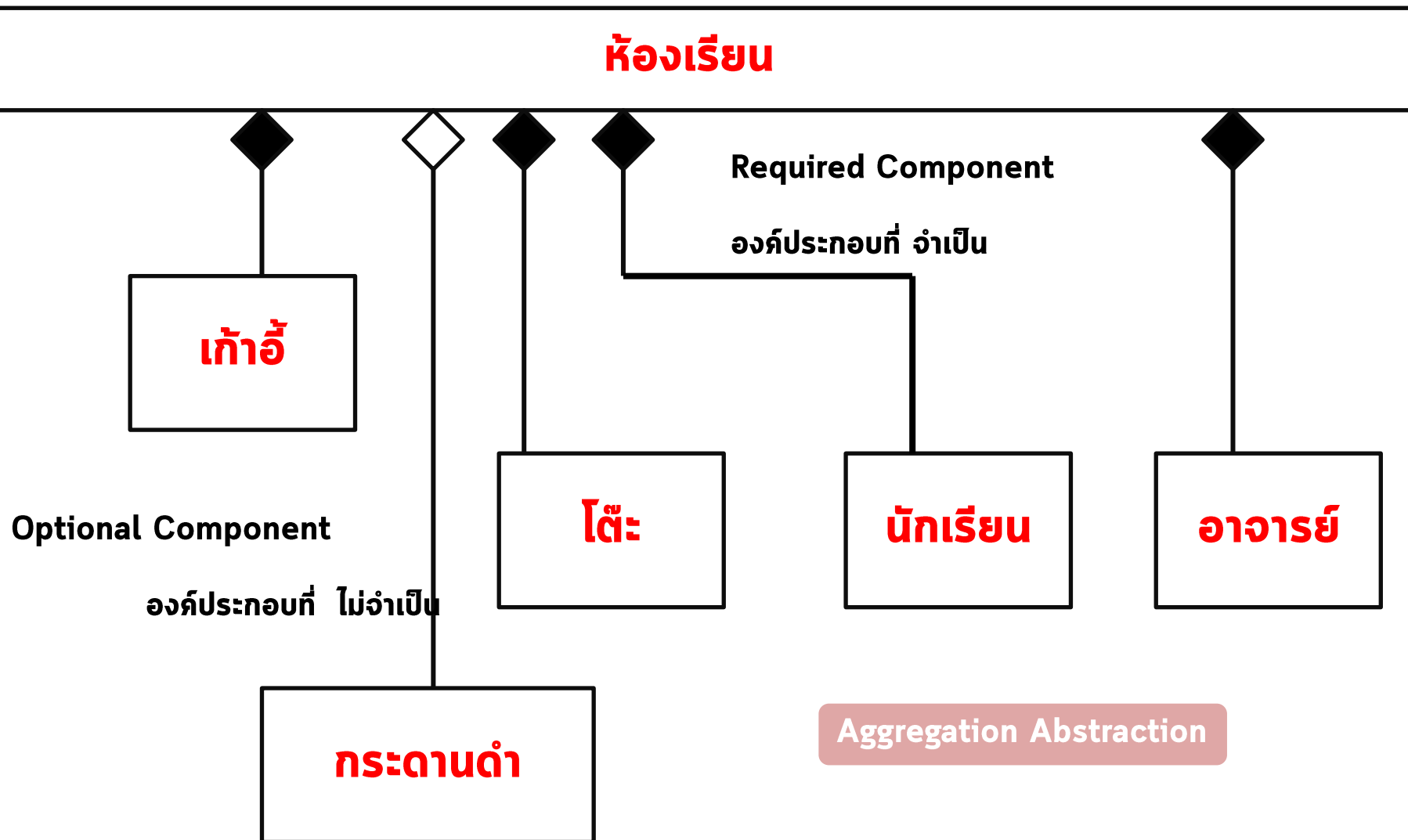
องค์ประกอบที่ จำเป็น

โต๊ะ

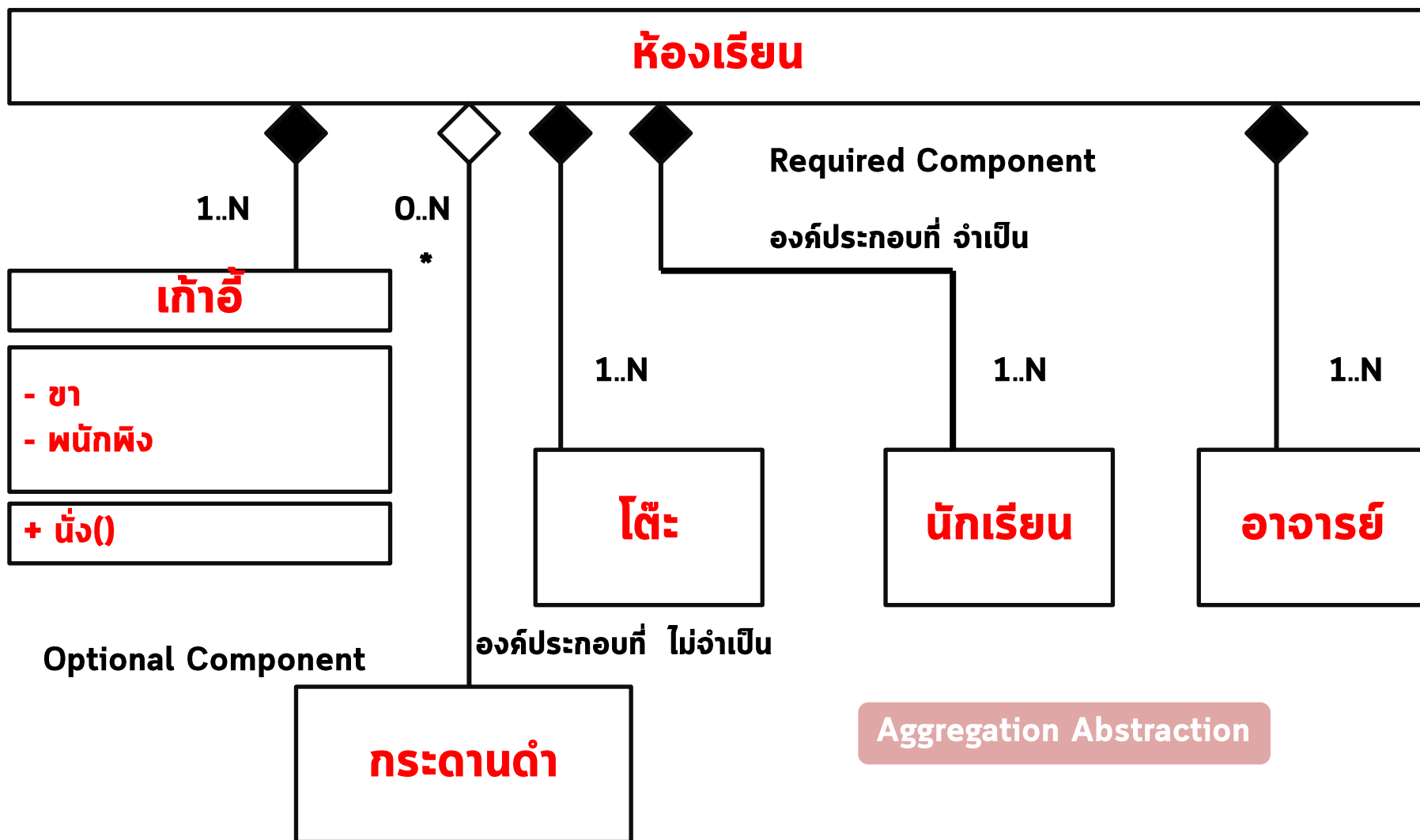
นักเรียน

อาจารย์

กระดานดำ

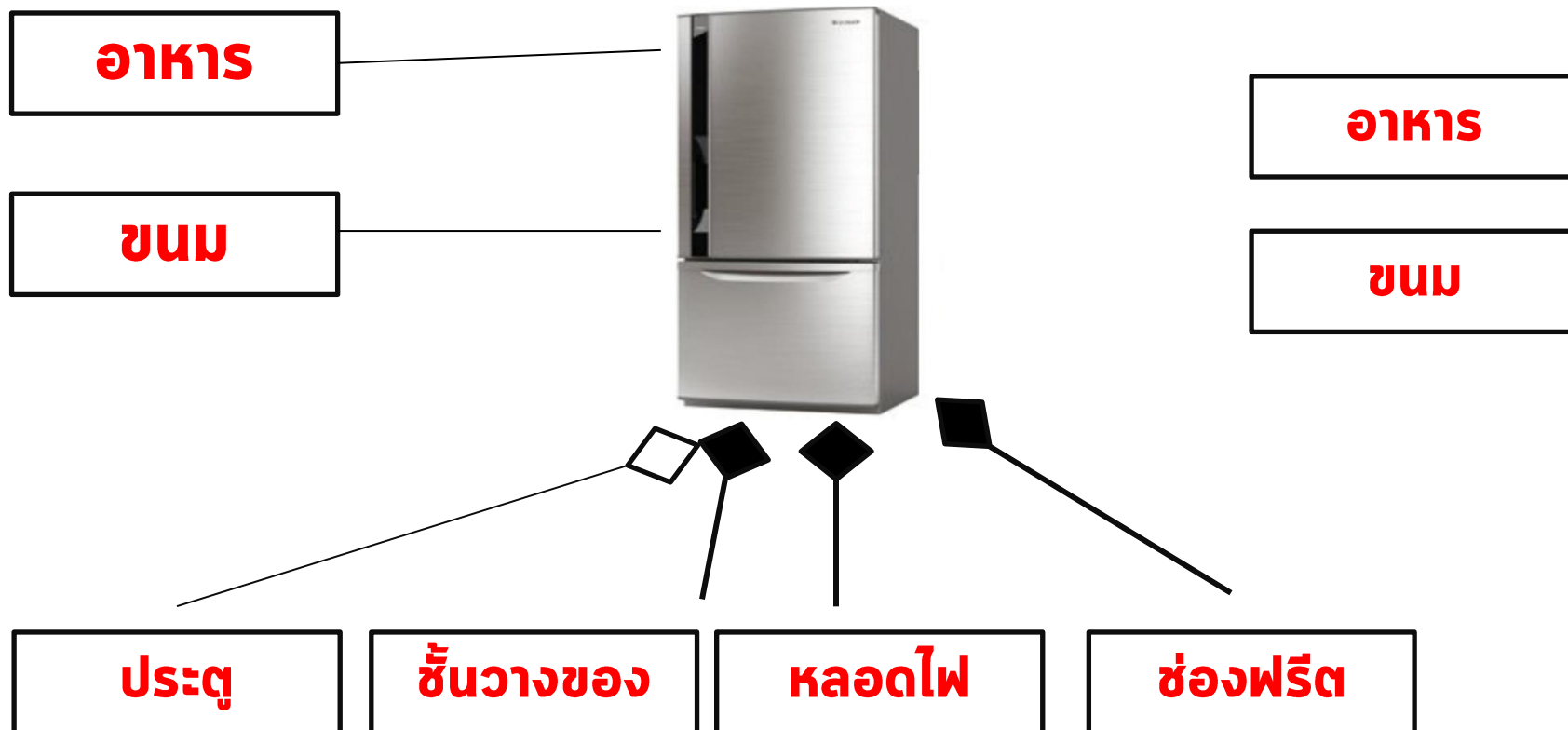


***** ห้องเรียน ทางกายภาพ(physical) ประกอบไปด้วย เสา หลังคา เพดาน พื้น ผนัง**



*** ห้องเรียน ทางกายภาพ(physical) ประกอบไปด้วย เสา หลังคา เพดาน พื้น ผ้าม่าน

Association relationship



อาหาร ขนม นม หรือ ผัก ไม่ใช่ส่วนประกอบของตู้เย็น

class

ห้องคอม

- หมายเลขห้อง
- อาจารย์
- นักศึกษา
- คอมพิวเตอร์
- โต๊ะคอมพิวเตอร์
- แอร์

object

- เปิดได้
- ปิดได้
- ใช้เรียน/สอน

ห้อง 844

- หมายเลขห้อง
- อาจารย์
- นักศึกษา
- คอมพิวเตอร์
- โต๊ะคอมพิวเตอร์
- แอร์

- เปิดได้
- ปิดได้
- ใช้เรียน/สอน

Is member of

ห้อง 845

11/01/68

<http://www.siam2dev.com>

ห้อง 843 เป็นห้องพัก
อาจารย์ ไม่สามารถใช้
สอนคอมพิวเตอร์ได้

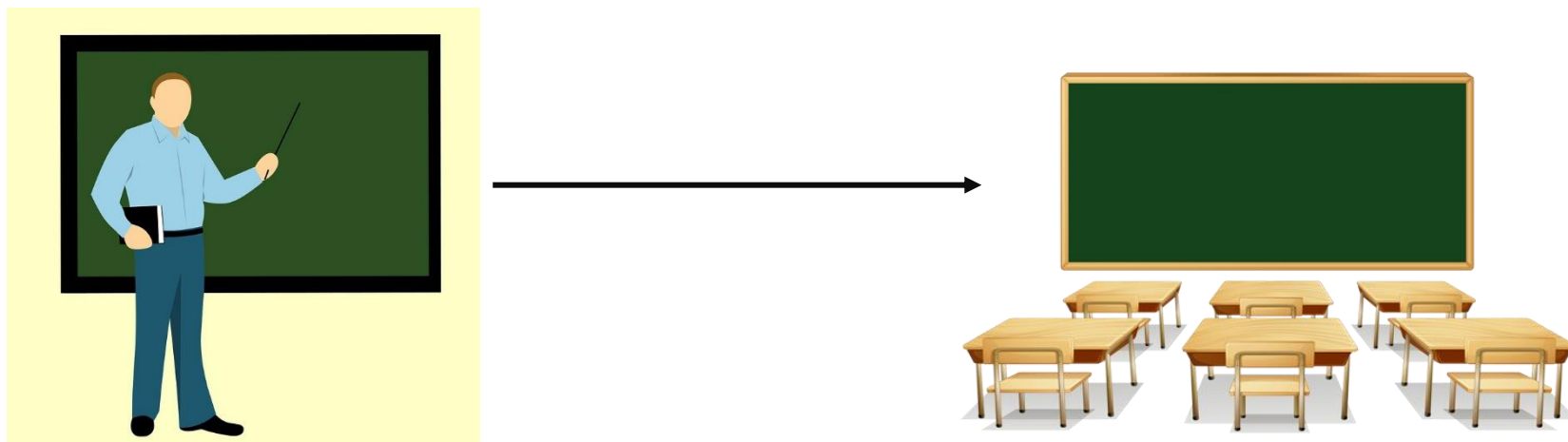
ห้อง 822

ห้อง 833

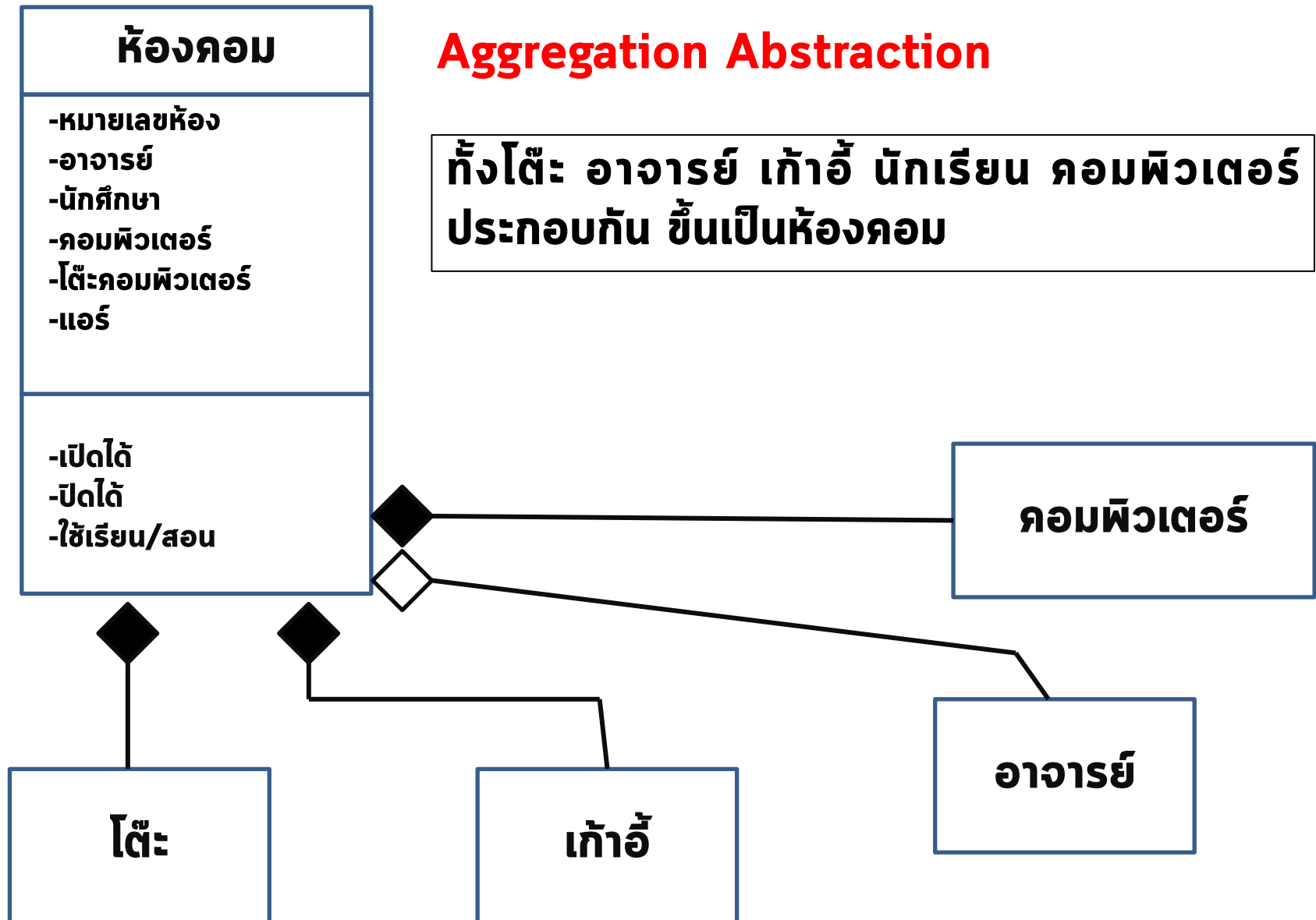
ห้อง 833 เป็น
ห้องเรียนธรรมดาไม่มี
คอมพิวเตอร์ จึงไม่
สามารถใช้สอน
คอมพิวเตอร์ได้

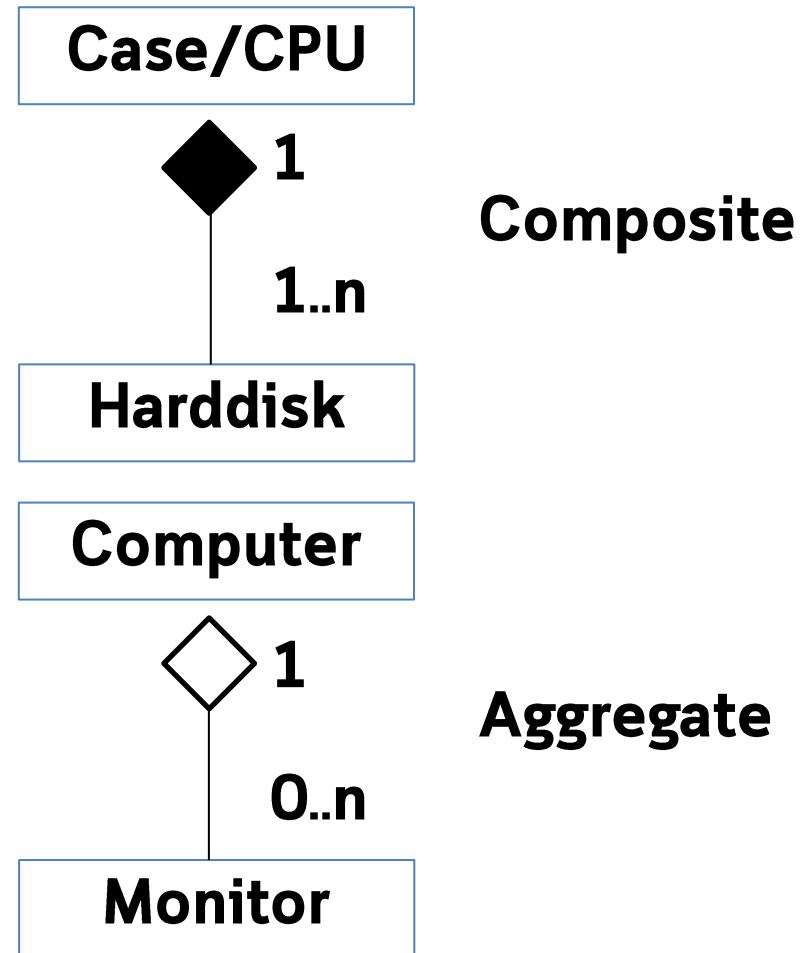
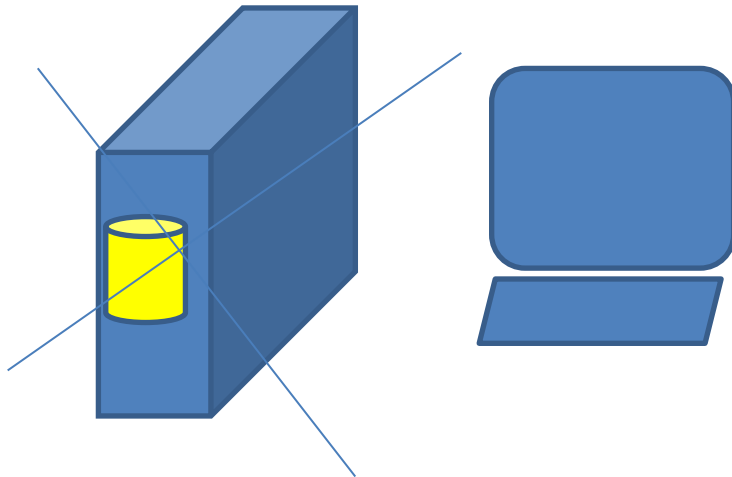
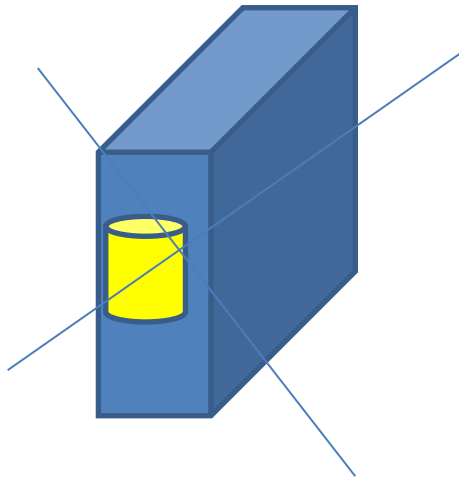
ห้องเรียน

***** ห้องเรียน ทางกายภาพ(physical) ประกอบไปด้วย เสา หลังคา เพดาน พื้น ผนัง**



**อาจจะมองว่า อาจารย์ ไม่ใช่ส่วนหนึ่งของห้องเรียน อาจารย์เป็นเพียงแค่
ผู้ใช้(association)ห้องเรียน**

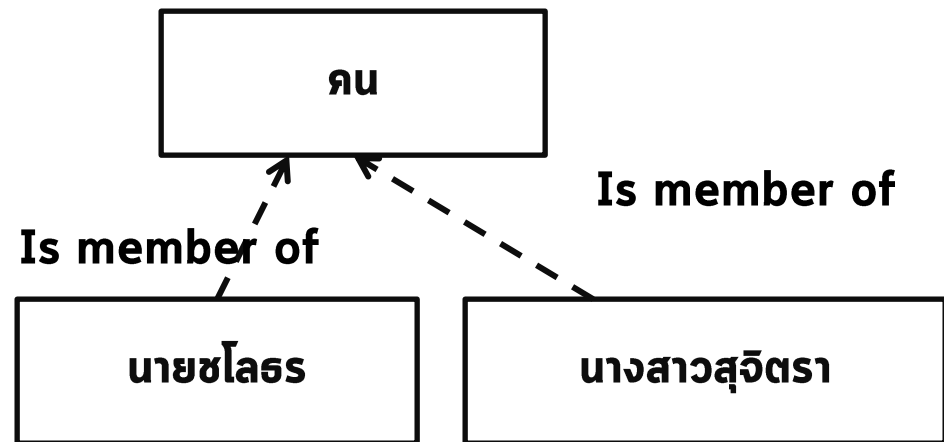




Classification Abstraction

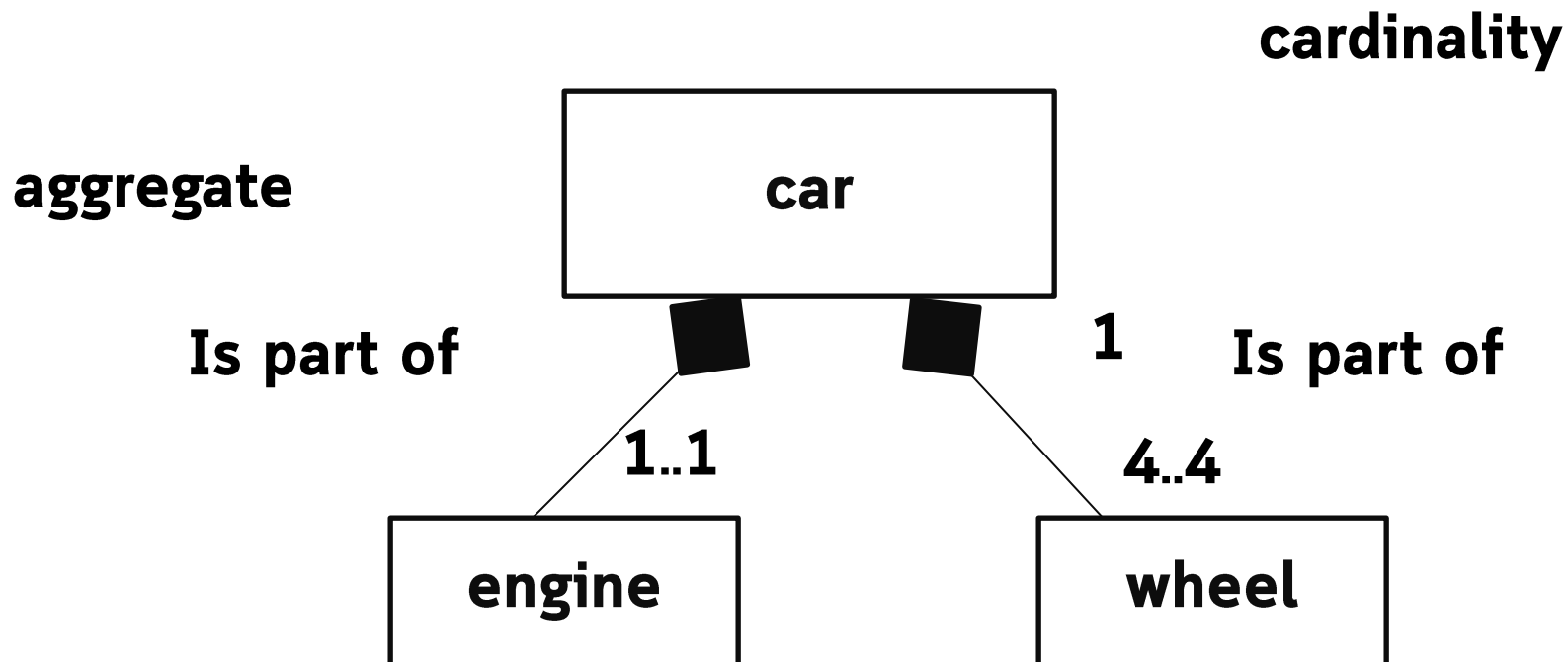
- Classification abstraction การหาคลาสและวัตถุ แล้วก็ระบุ วัตถุนั้นอยู่ในคลาสใด เช่น นายก. มีแขน มีขา มีหัว มีลำตัว เดินได้() กินได้() นอน() เรียกว่า **คลาสคน เป็นการจัดกลุ่มของวัตถุที่มีลักษณะเหมือนกัน และมีฟังก์ชันเหมือนกัน จัดให้อยู่ในกลุ่มเดียว เรียกว่าคลาส(class)**

*** สี่ไม่เน้น หรือไม่ใช่สาระ
เน้นสัญลักษณ์



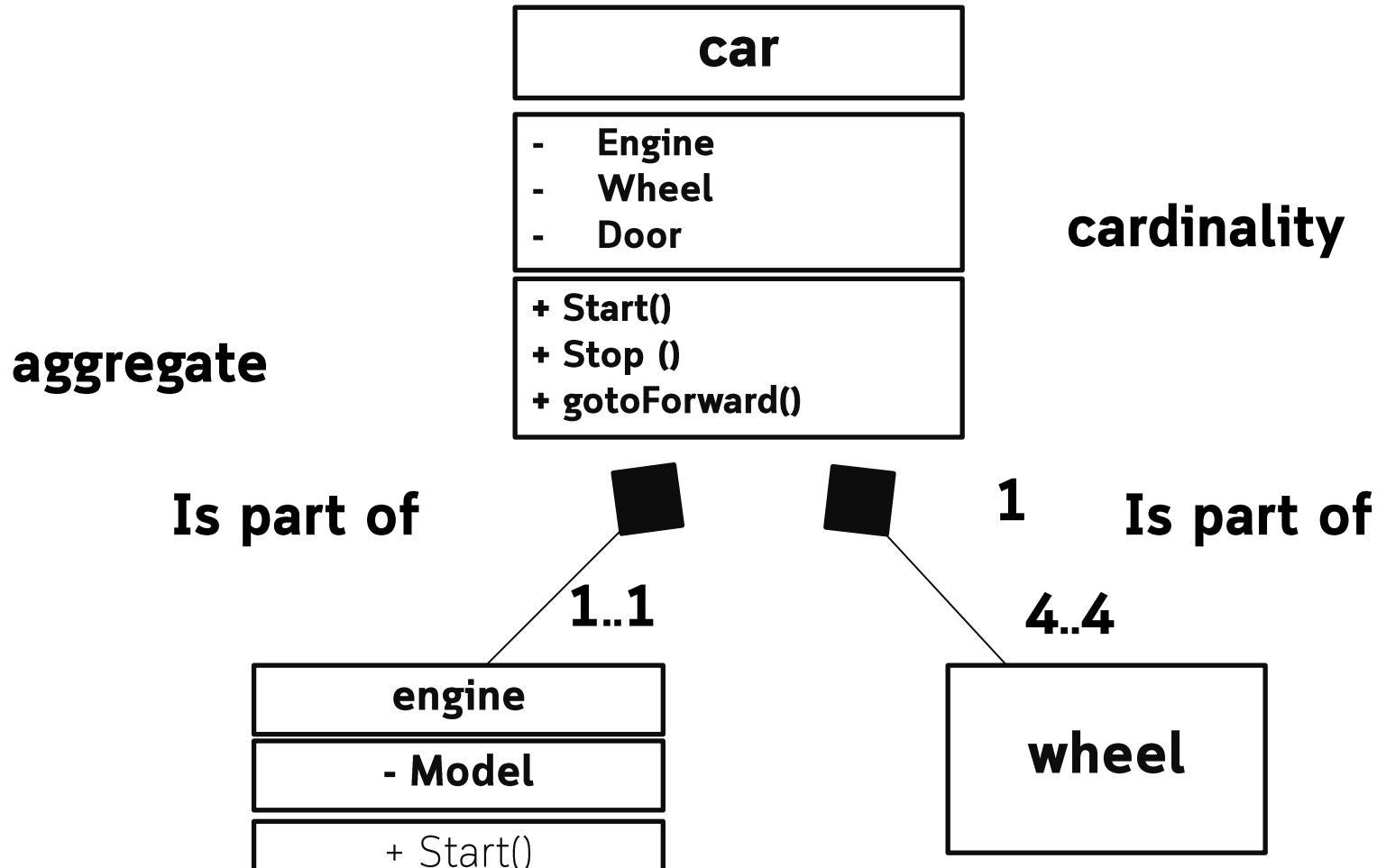
เมื่อได้คลาสต่าง ๆ ที่เป็นพื้นฐาน (**fundamental classes**)

- เราใช้ **classification** เพื่อสร้างคลาสพื้นฐาน จากนั้นนำมาพิจารณาหาความสัมพันธ์ระหว่างคลาส หรือภายในคลาสนั้น **relationship**

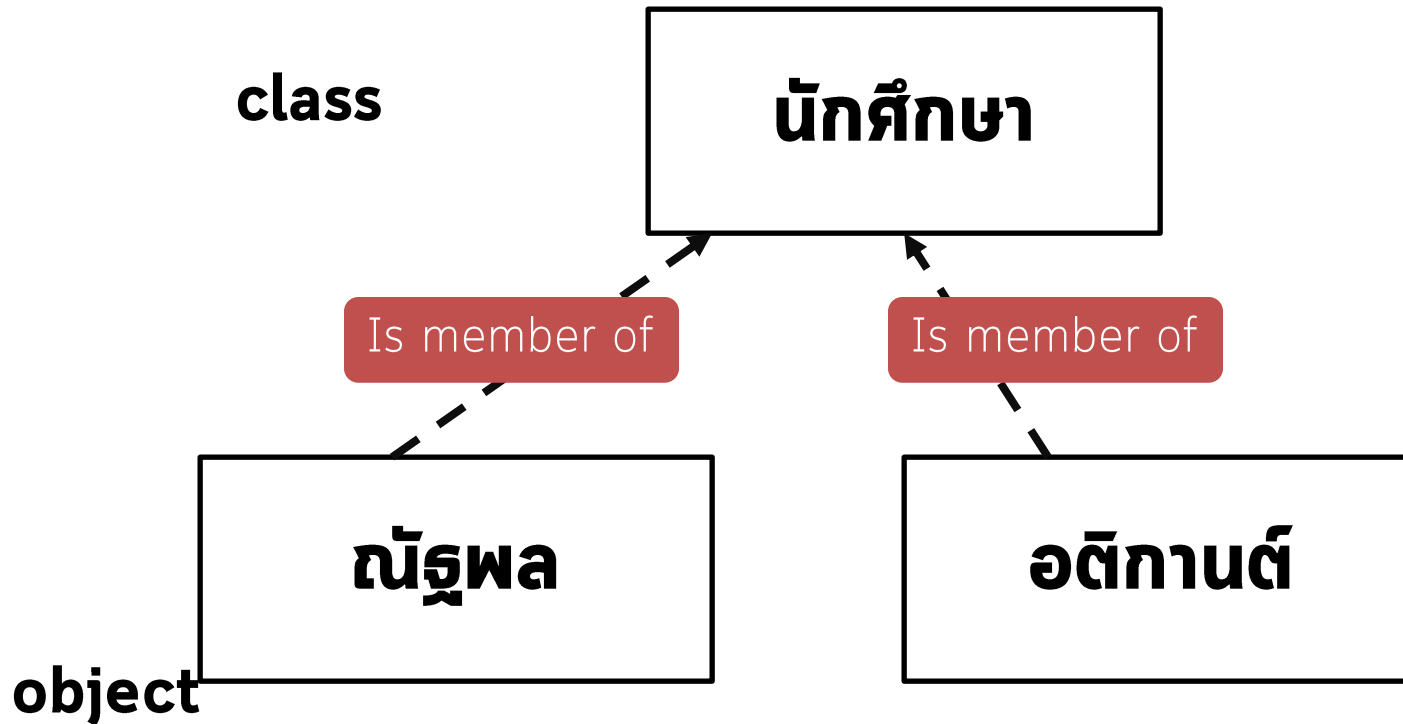


***พิจารณาแบบนี้ไปเรื่อยๆ จนหมดทุกๆ คลาสที่มีอยู่ใน Problem Domain

Aggregation Abstraction

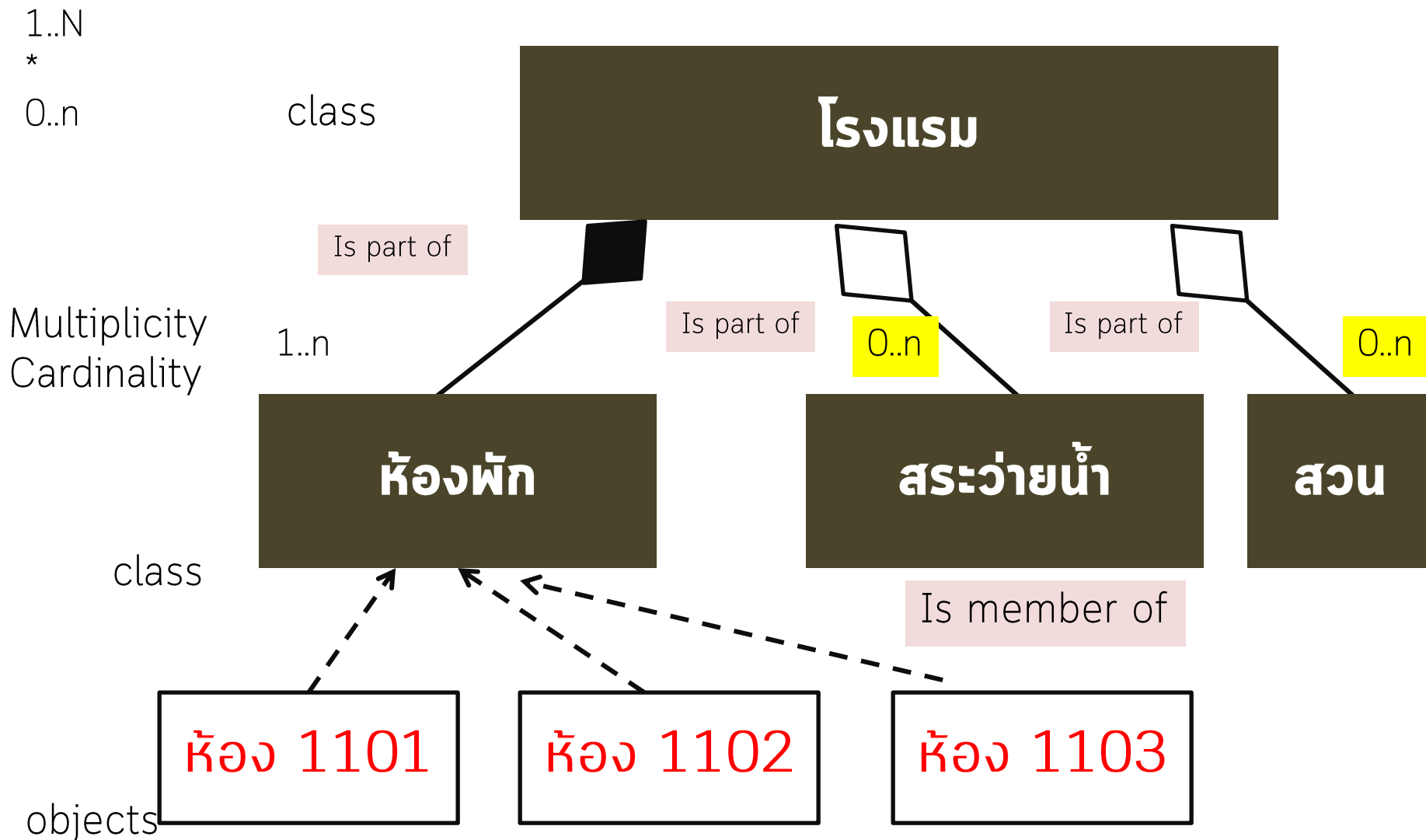


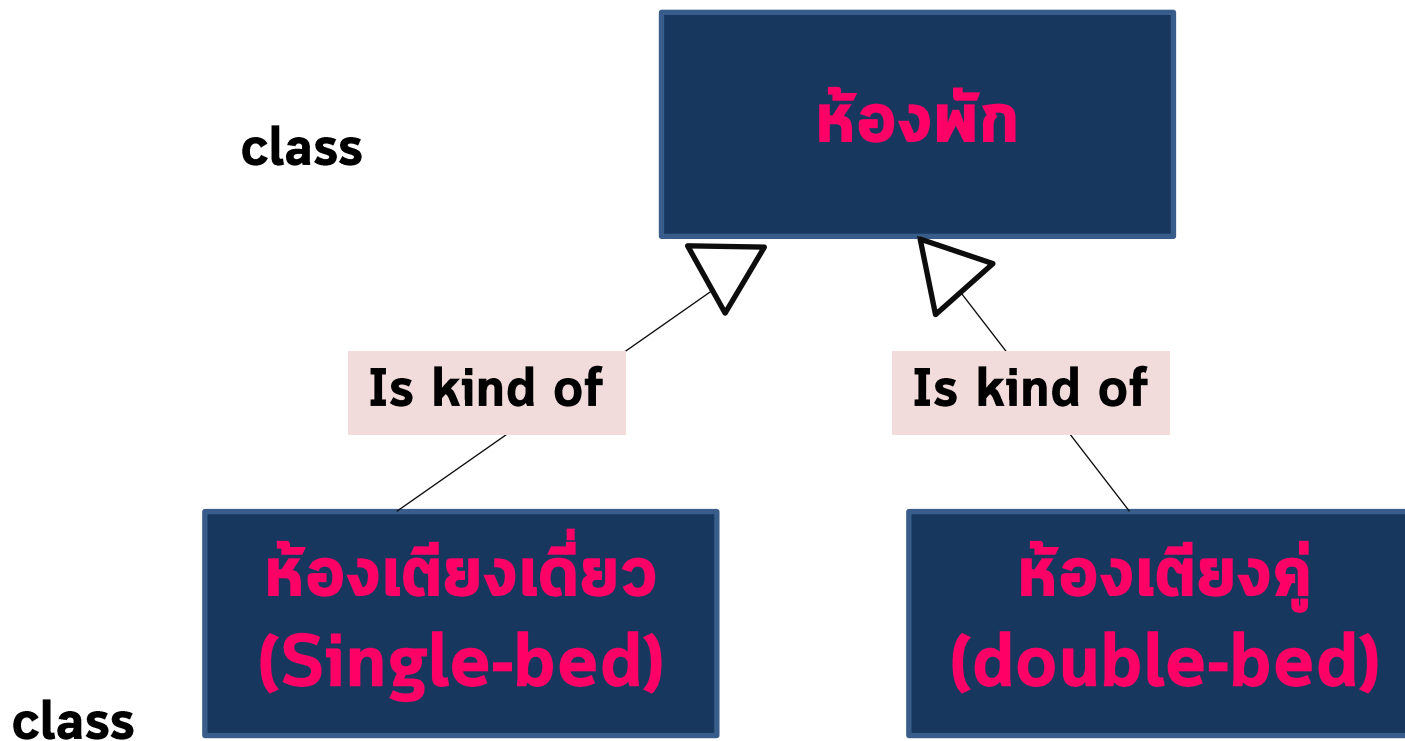
***พิจารณาแบบนี้ไปเรื่อยๆ จนหมดทุกๆ คลาสที่มีอยู่ใน Problem Domain

Classification Abstractions

***พิจารณาแบบนี้ไปเรื่อยๆ จนหมดทุกๆ คลาสที่มีอยู่ใน Problem Domain

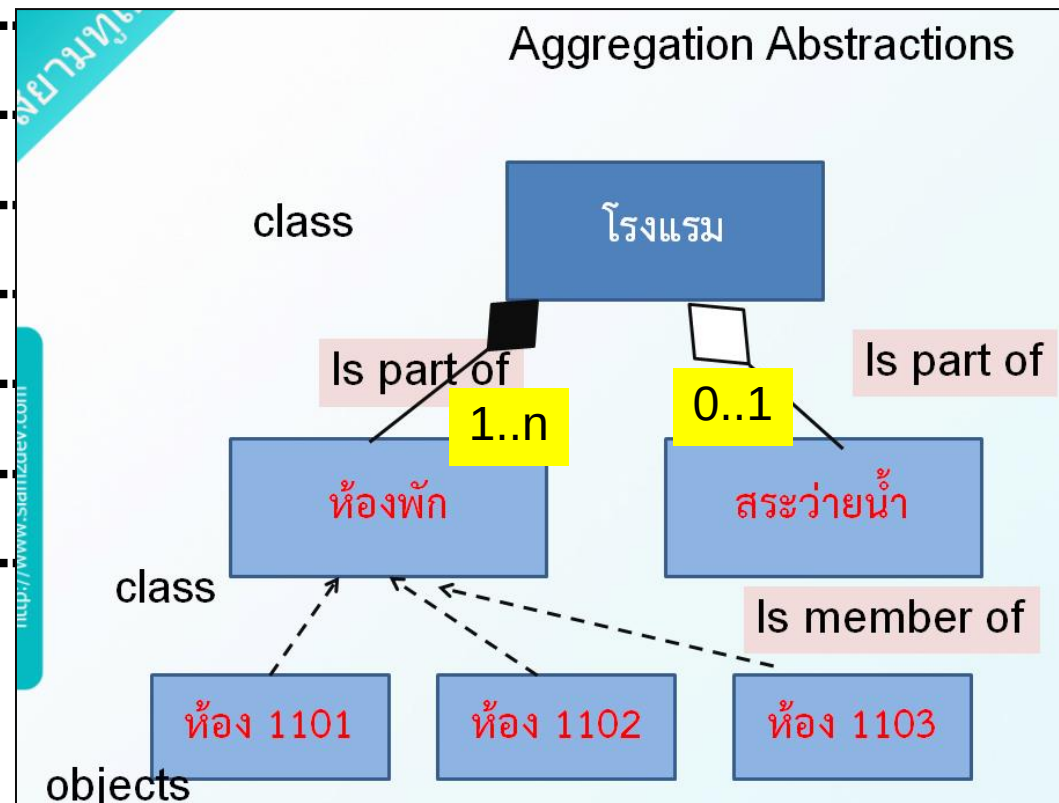
Aggregation Abstractions



Generalization Abstractions

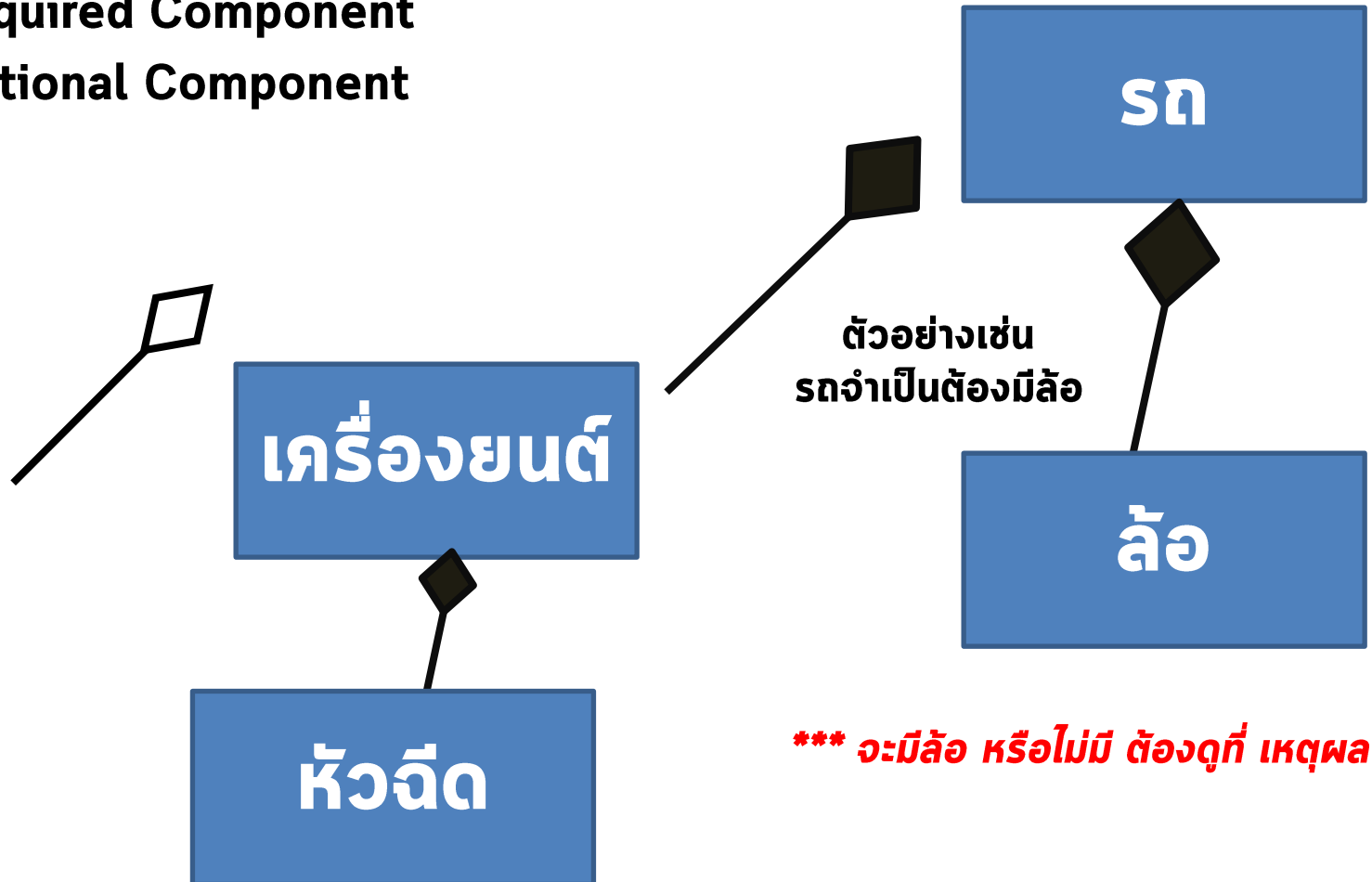
ตัวอย่าง จงอธิบายรูปนี้

- อธิบายในเชิงของ **aggregation** ได้ว่า

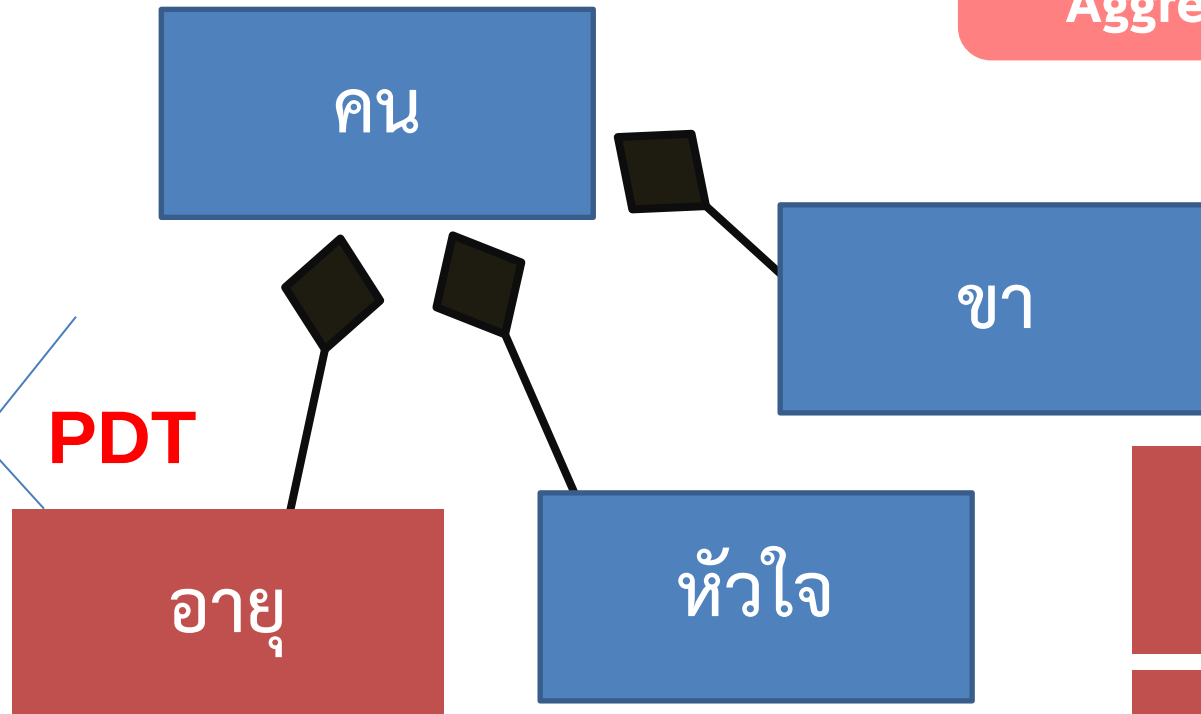


Aggregate

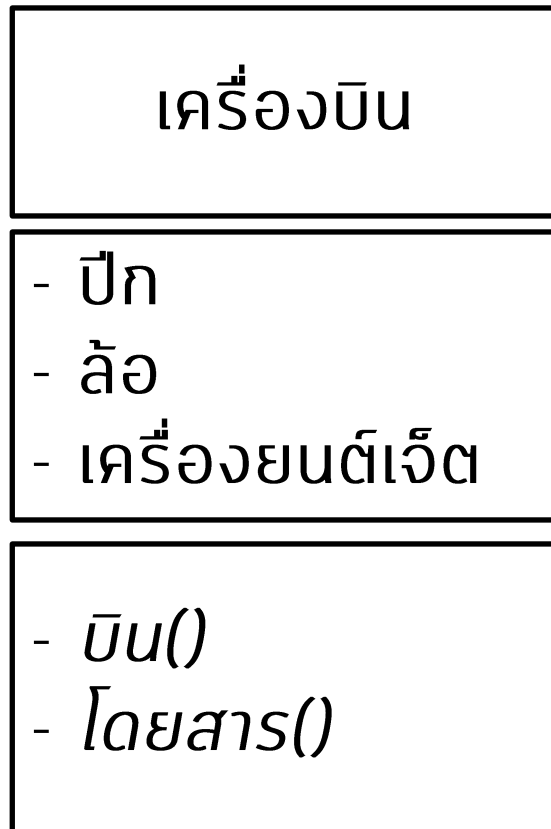
- Required Component
- Optional Component



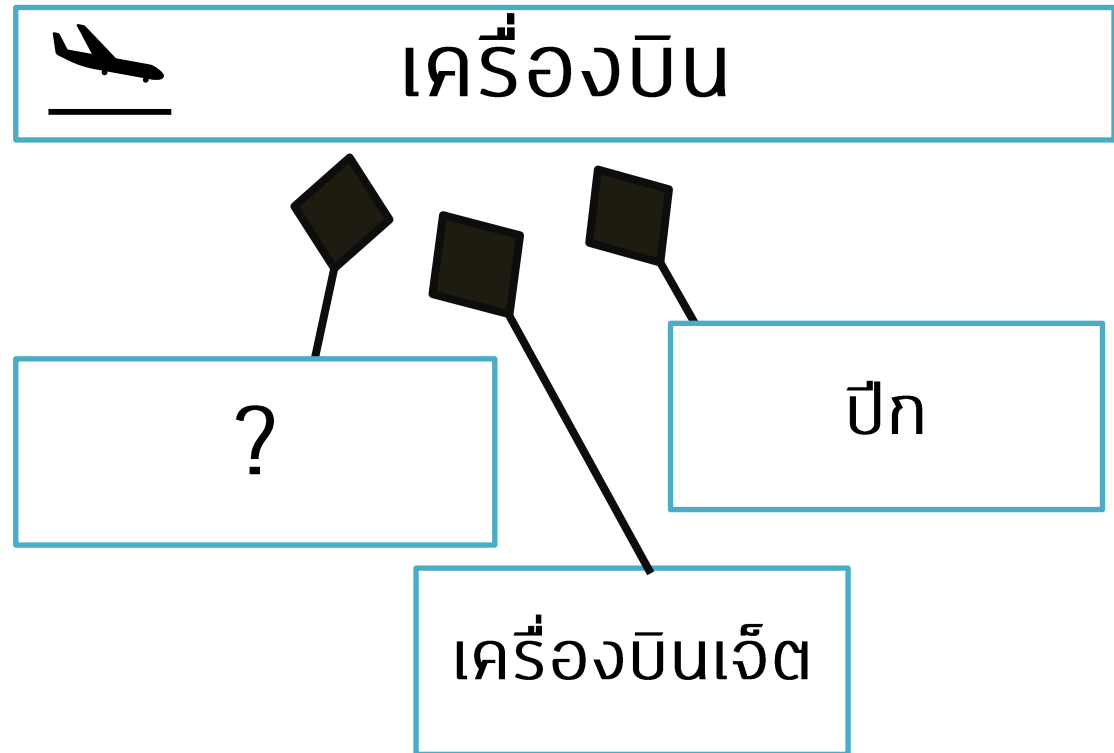
Aggregation Abstraction



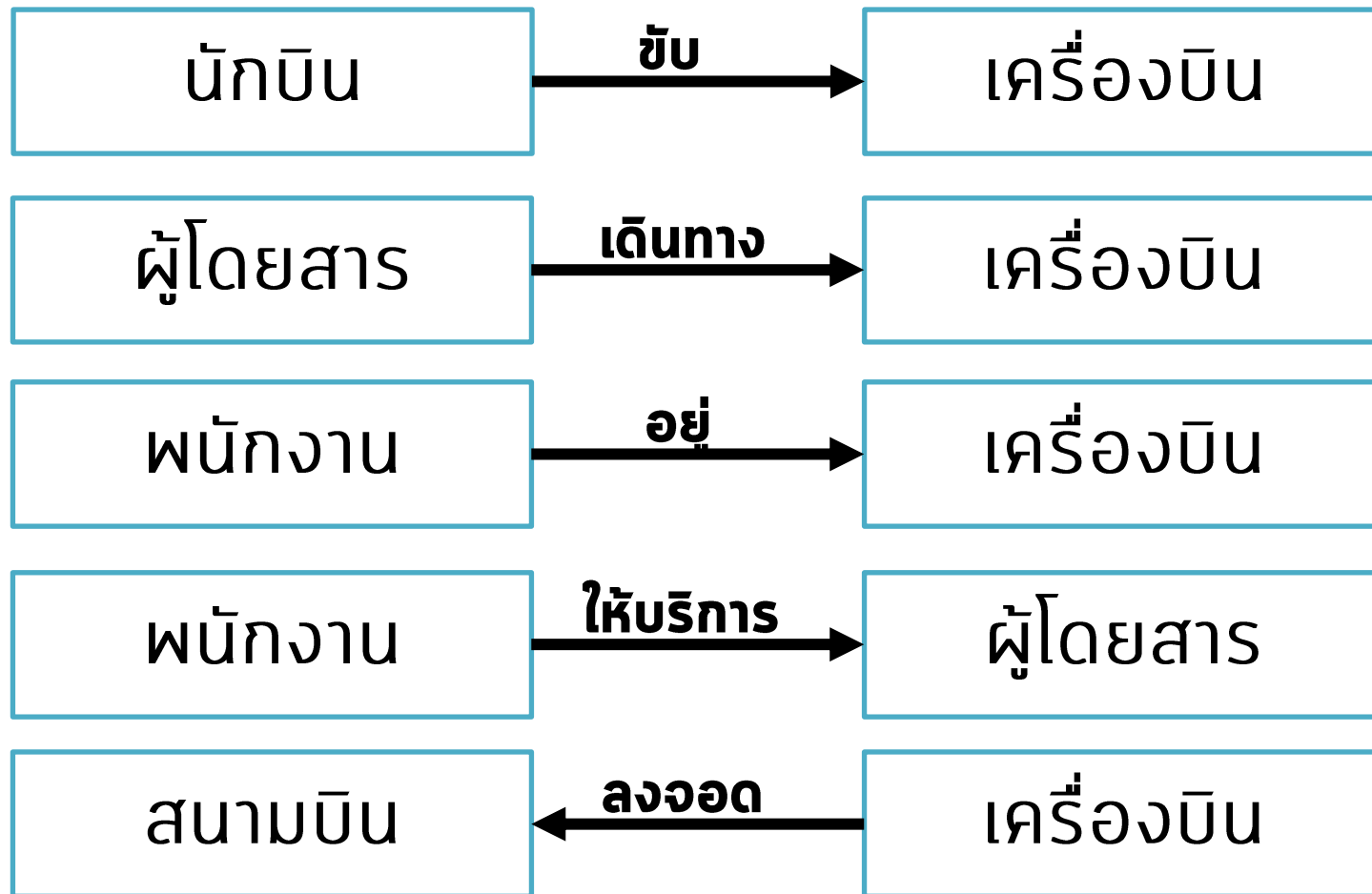
คน มี attributes
objects



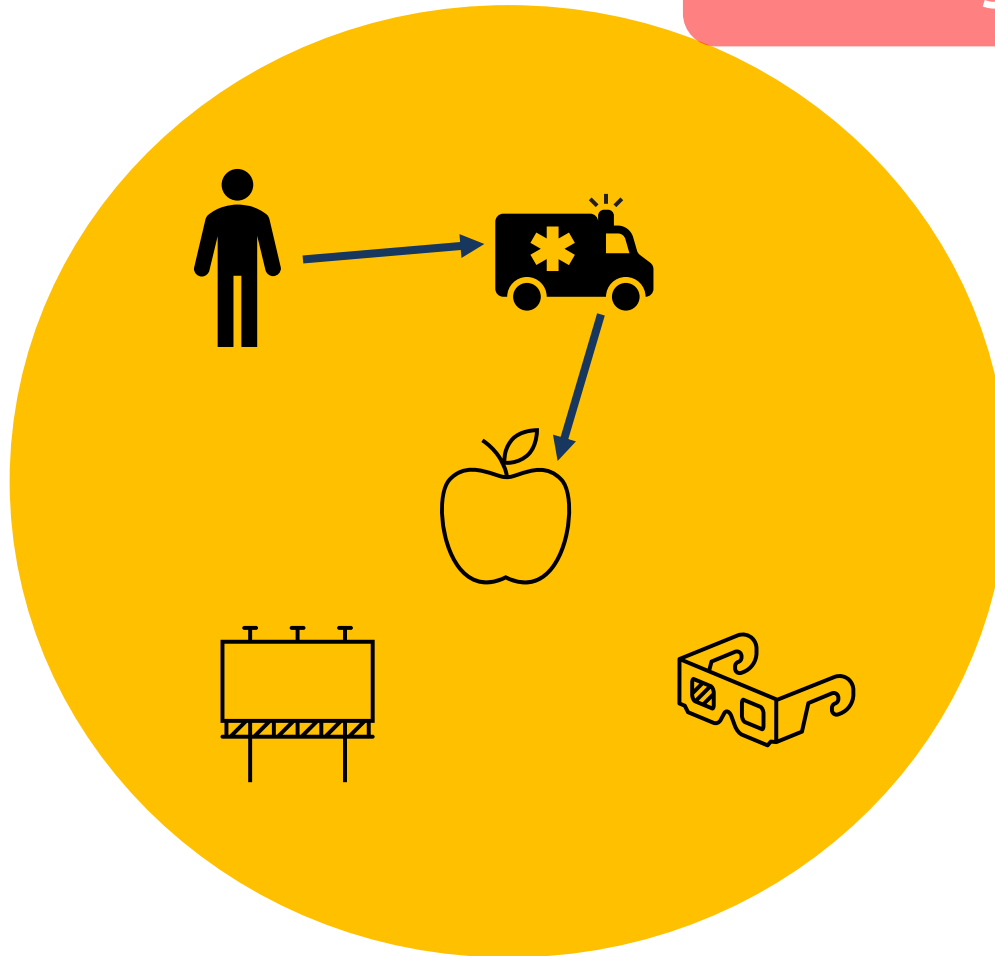
Aggregation Abstraction



นักบิน ไม่ใช่ ส่วนประกอบของเครื่องบิน



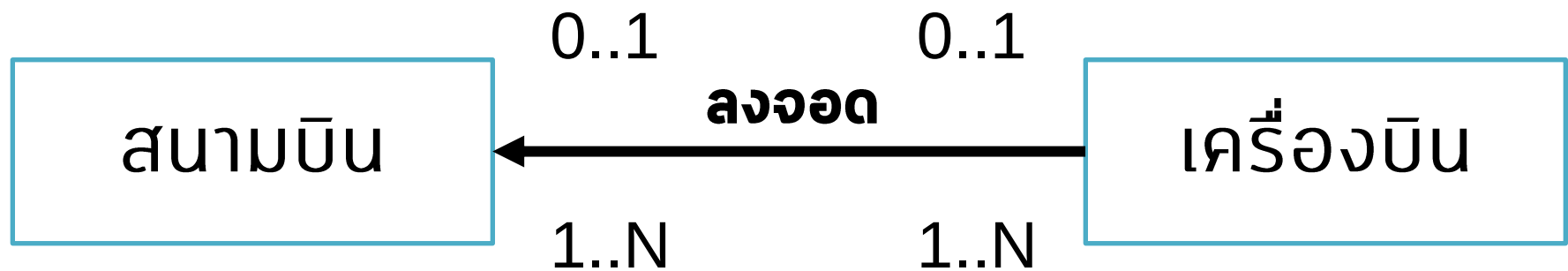
ระบบขนส่ง



ระบบเดินทางด้วยเครื่องบิน

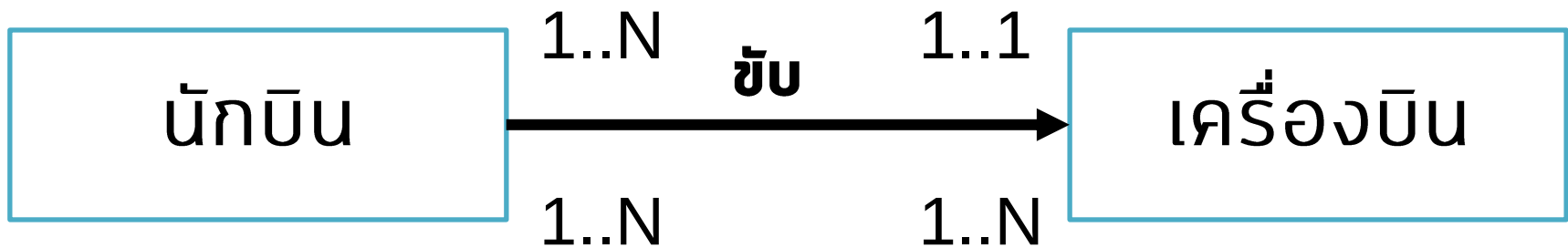


ระบบเดินทางด้วยเครื่องบิน



Minimum Cardinality
Maximum Cardinality

ระบบเดินทางด้วยเครื่องบิน



จากภาพนี้ จงอธิบายในเชิง Association Abstraction

.....

.....

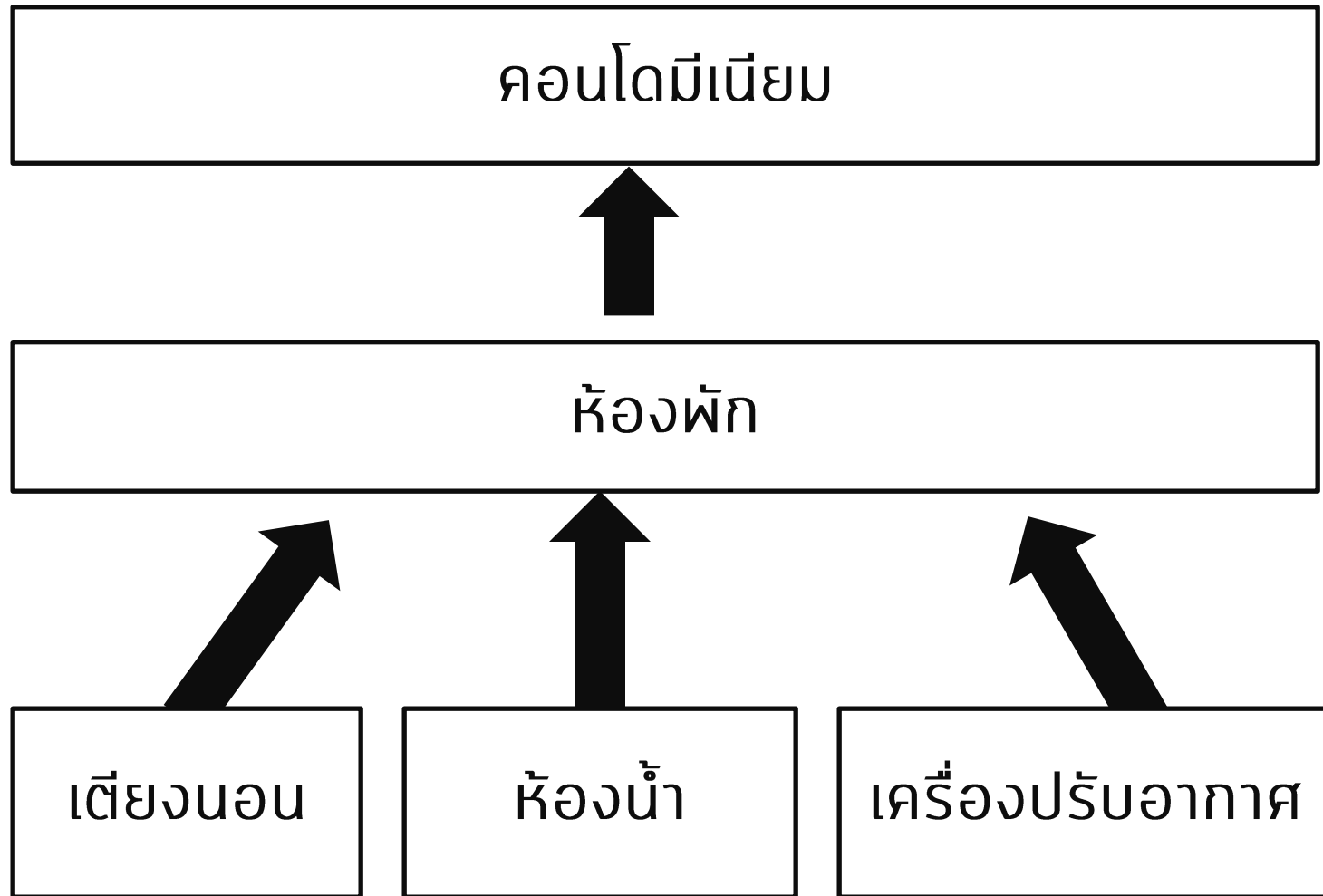
.....

.....

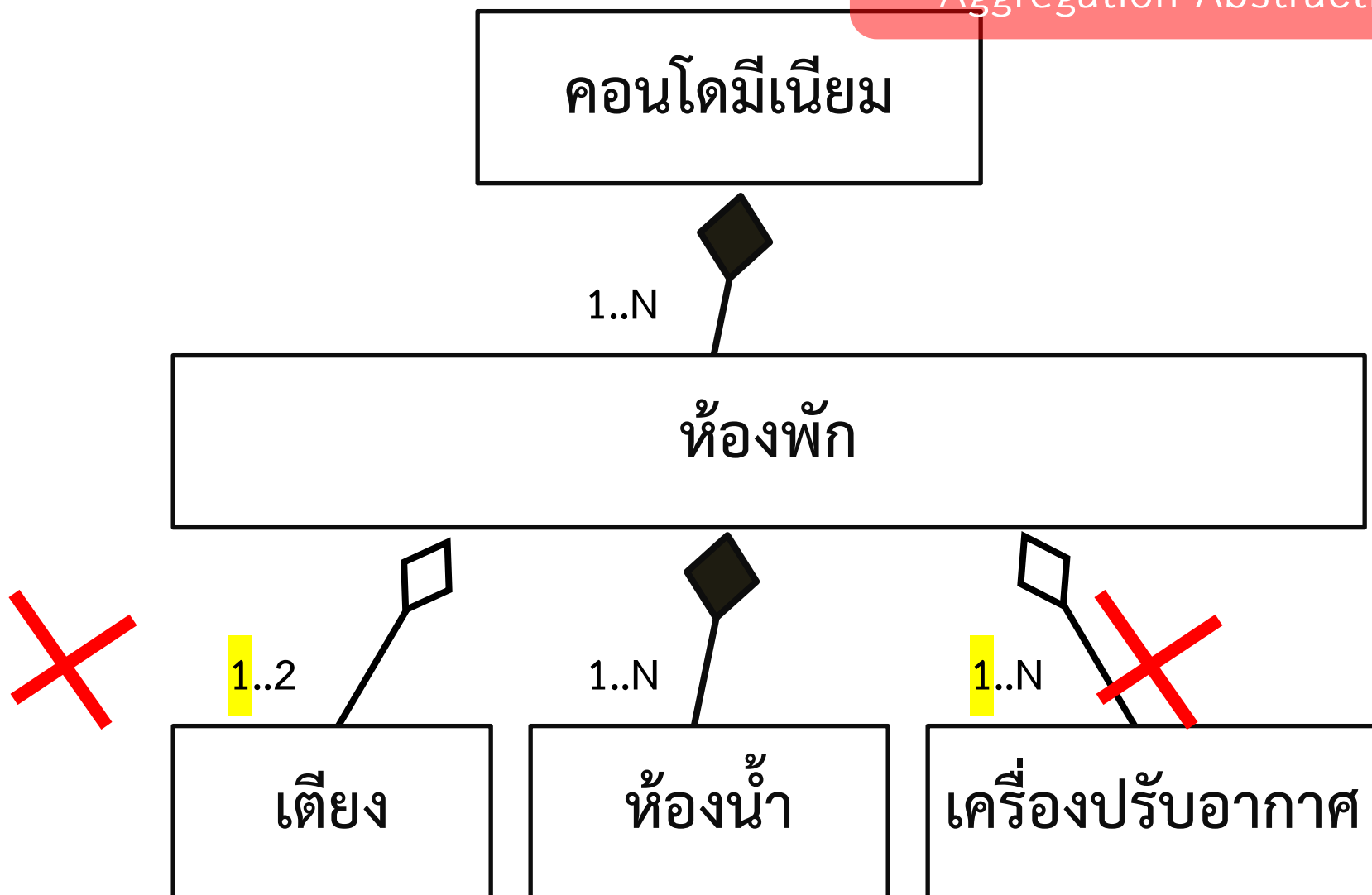
.....

.....

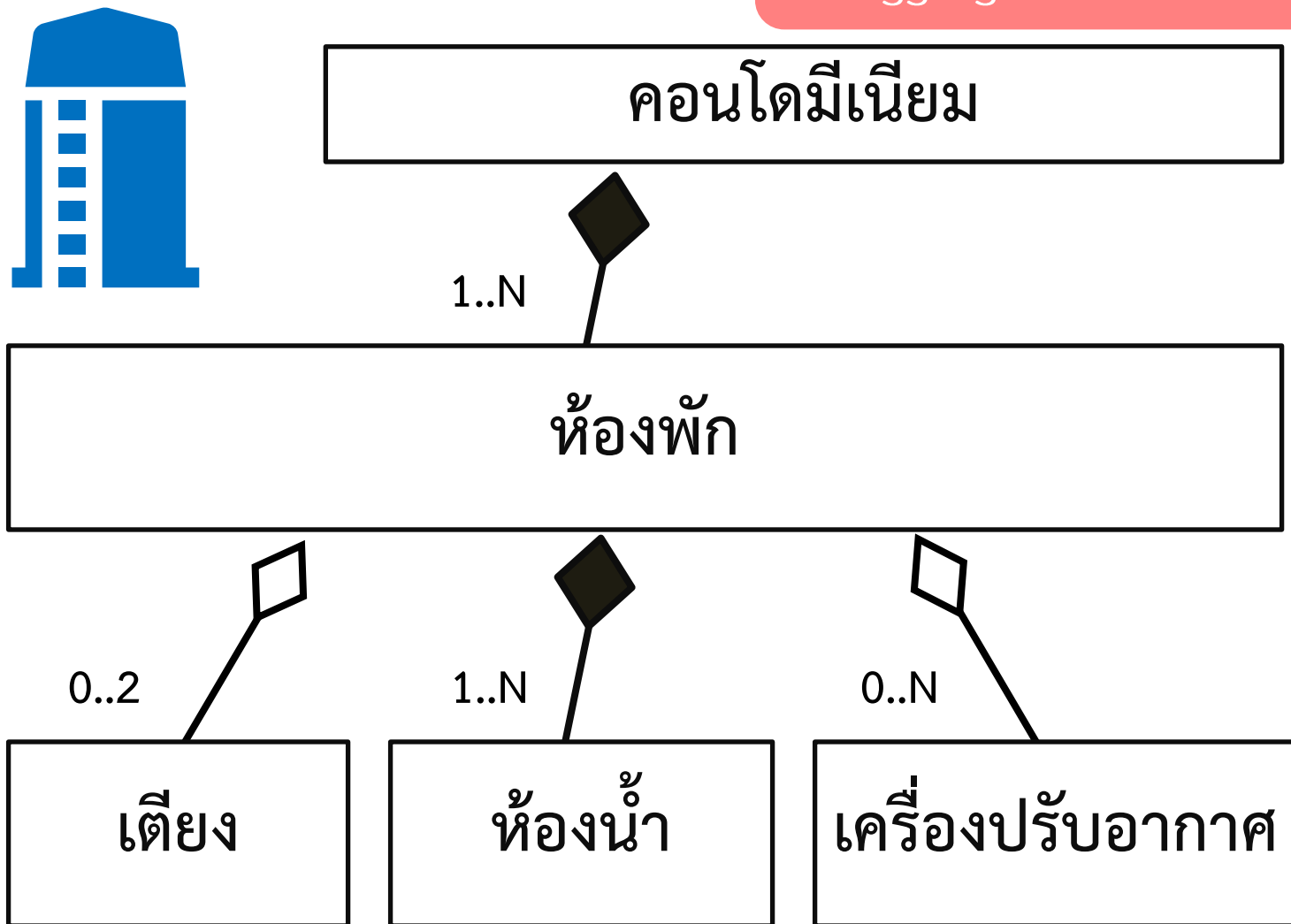
Aggregation Abstraction



Aggregation Abstraction

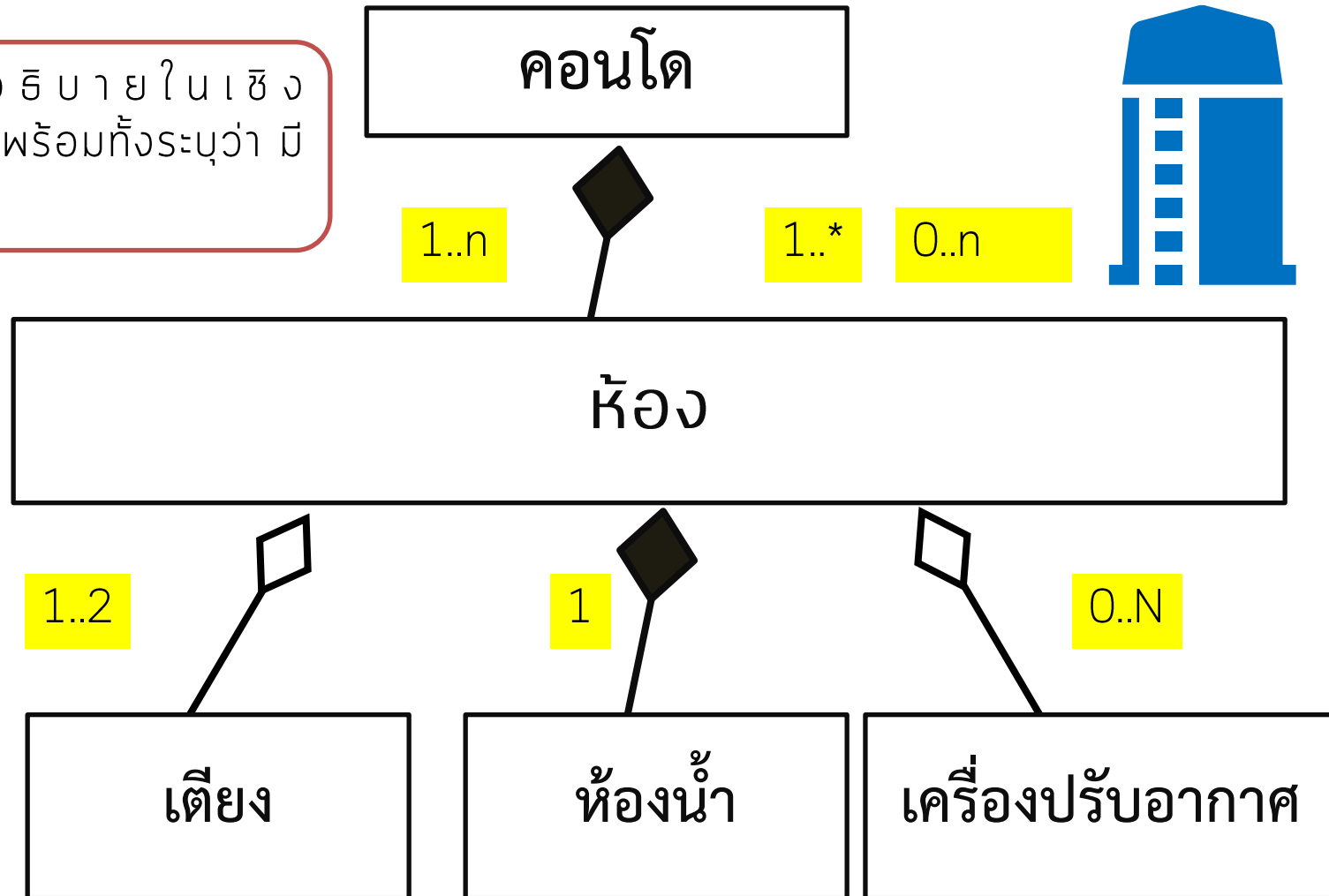


Aggregation Abstraction



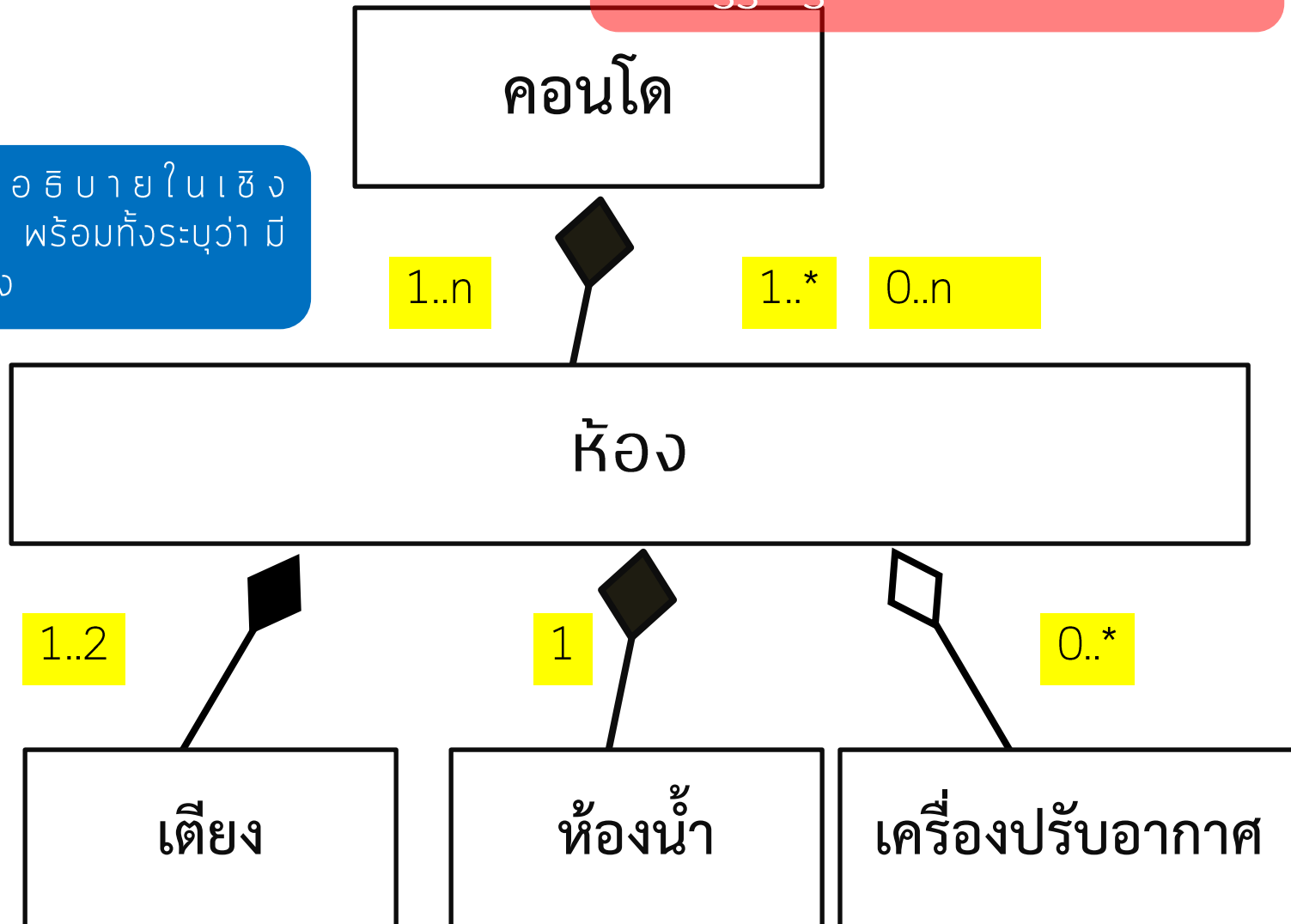
Aggregation Abstraction

จากรูปนี้จงอธิบายในเชิง aggregation พร้อมทั้งระบุว่า มีอะไรผิดพลาดบ้าง



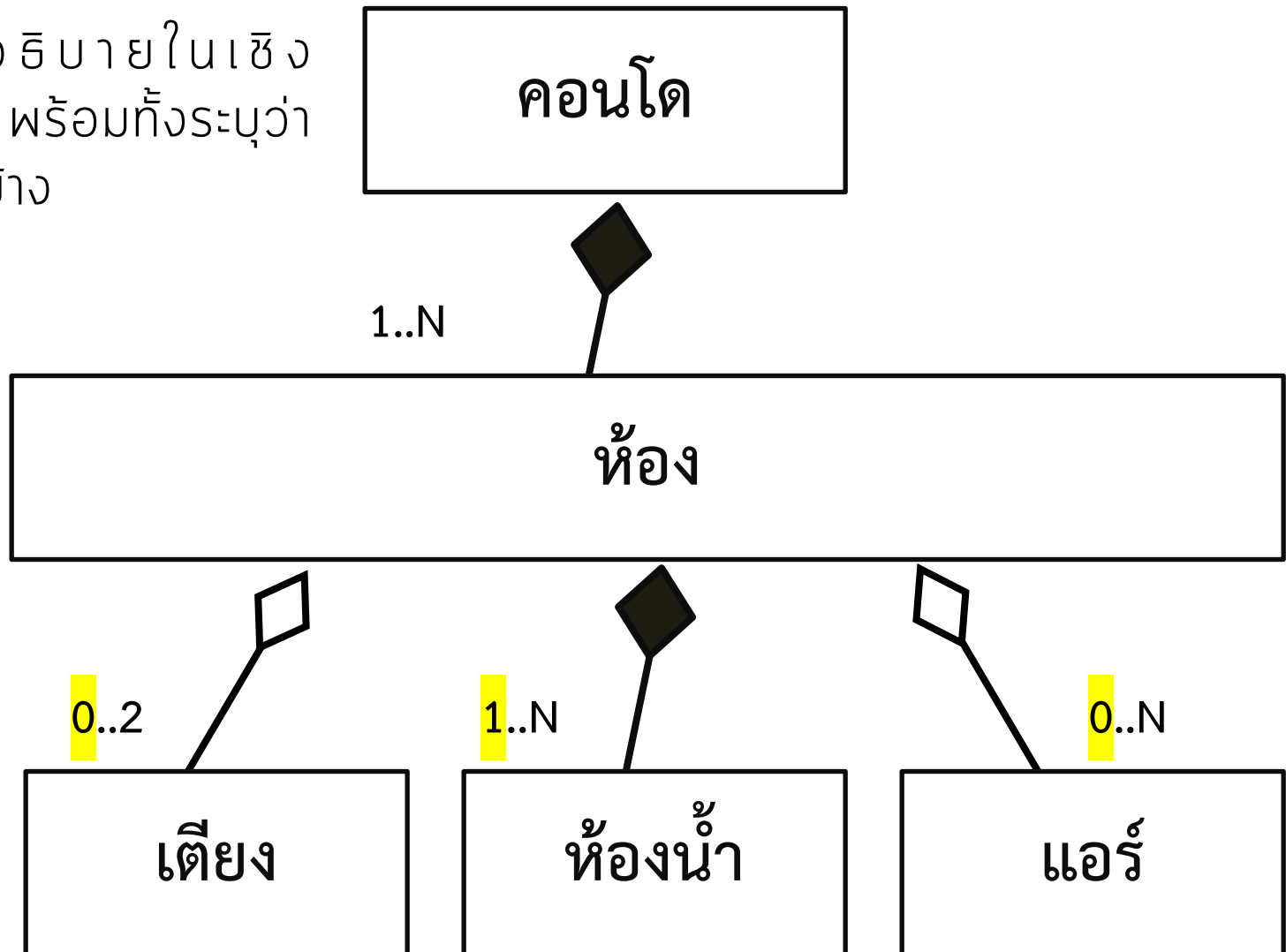
Aggregation Abstraction

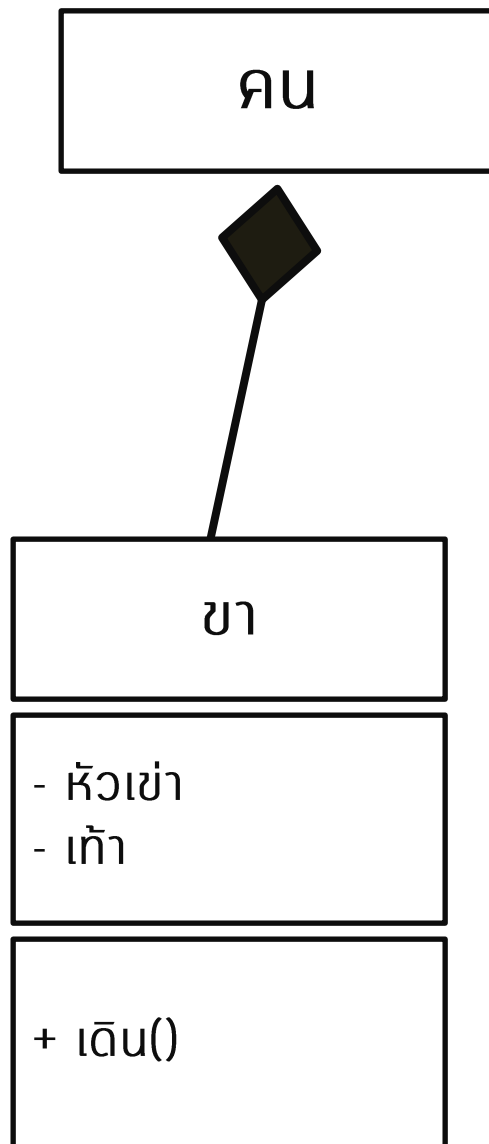
จากรูปนี้จงอธิบายในเชิง aggregation พร้อมทั้งระบุว่า มีอะไรผิดพลาดบ้าง



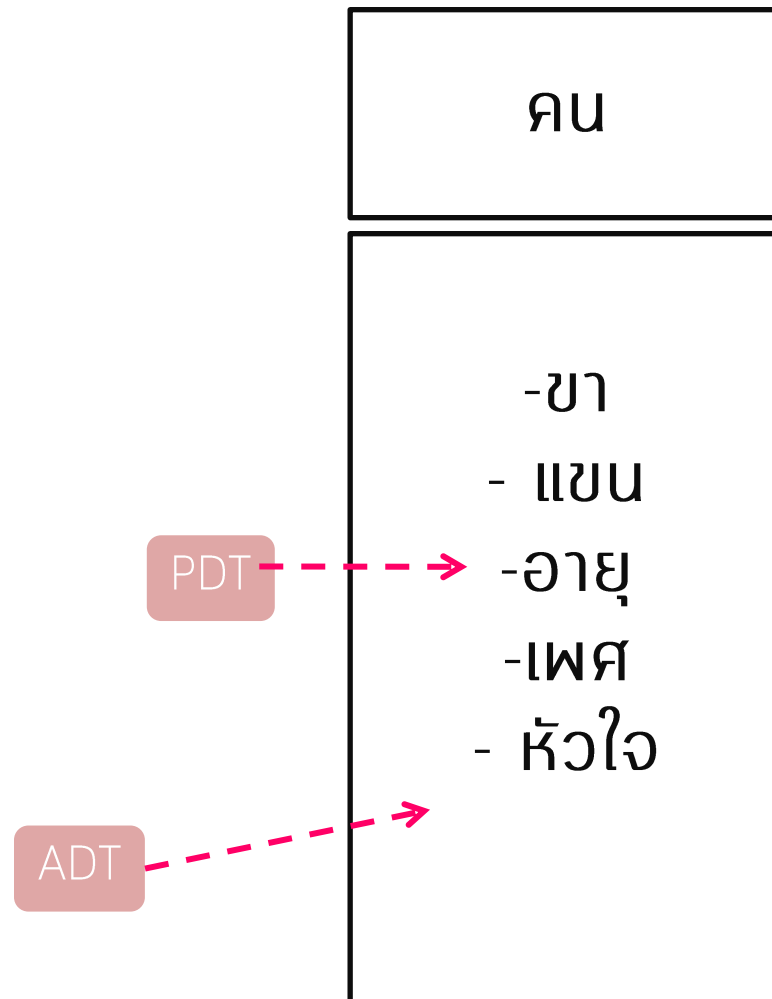
Aggregation Abstraction

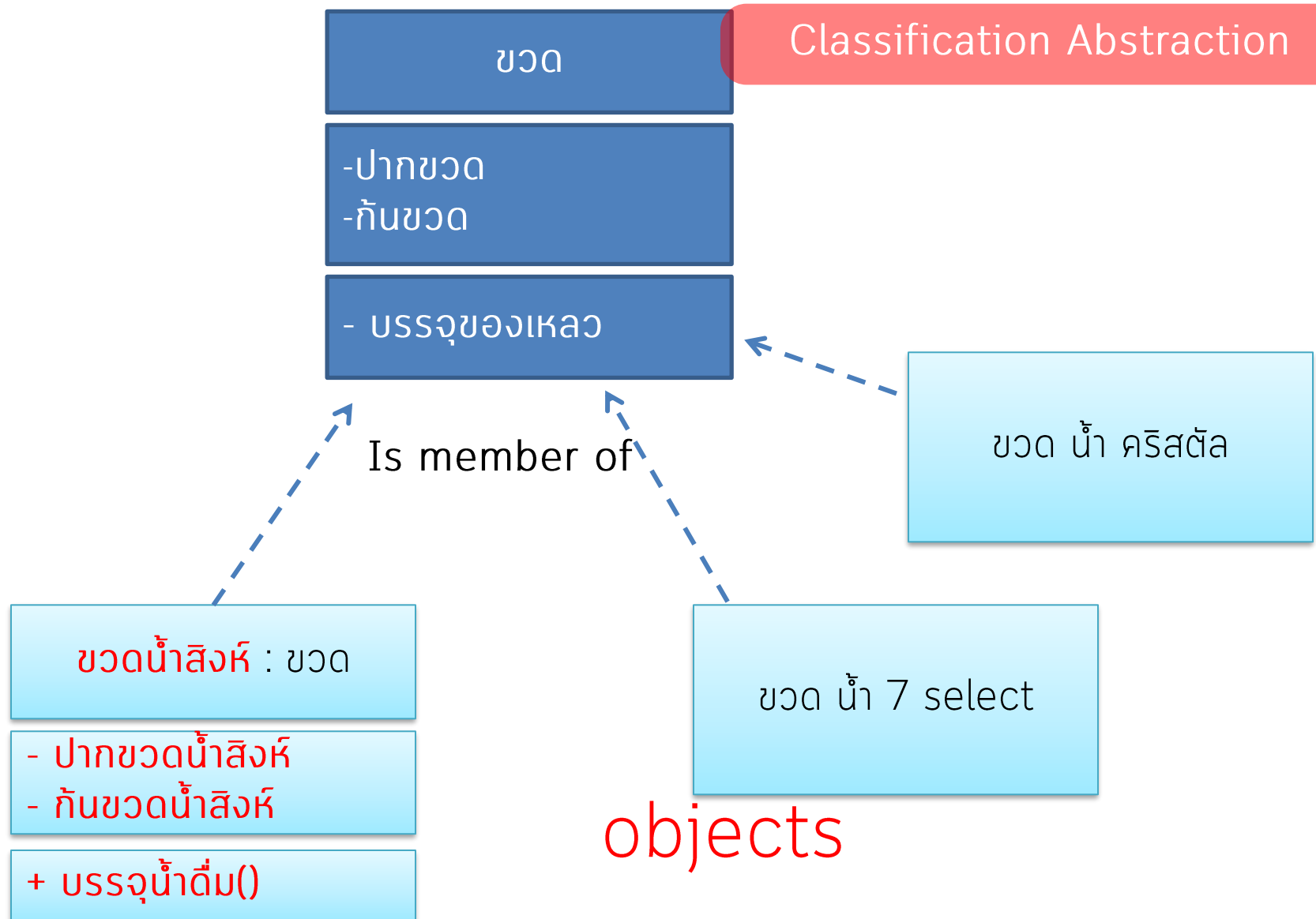
จากรูปนี้จงอธิบายในเชิง aggregation พร้อมทั้งระบุว่า มีอะไรผิดพลาดบ้าง



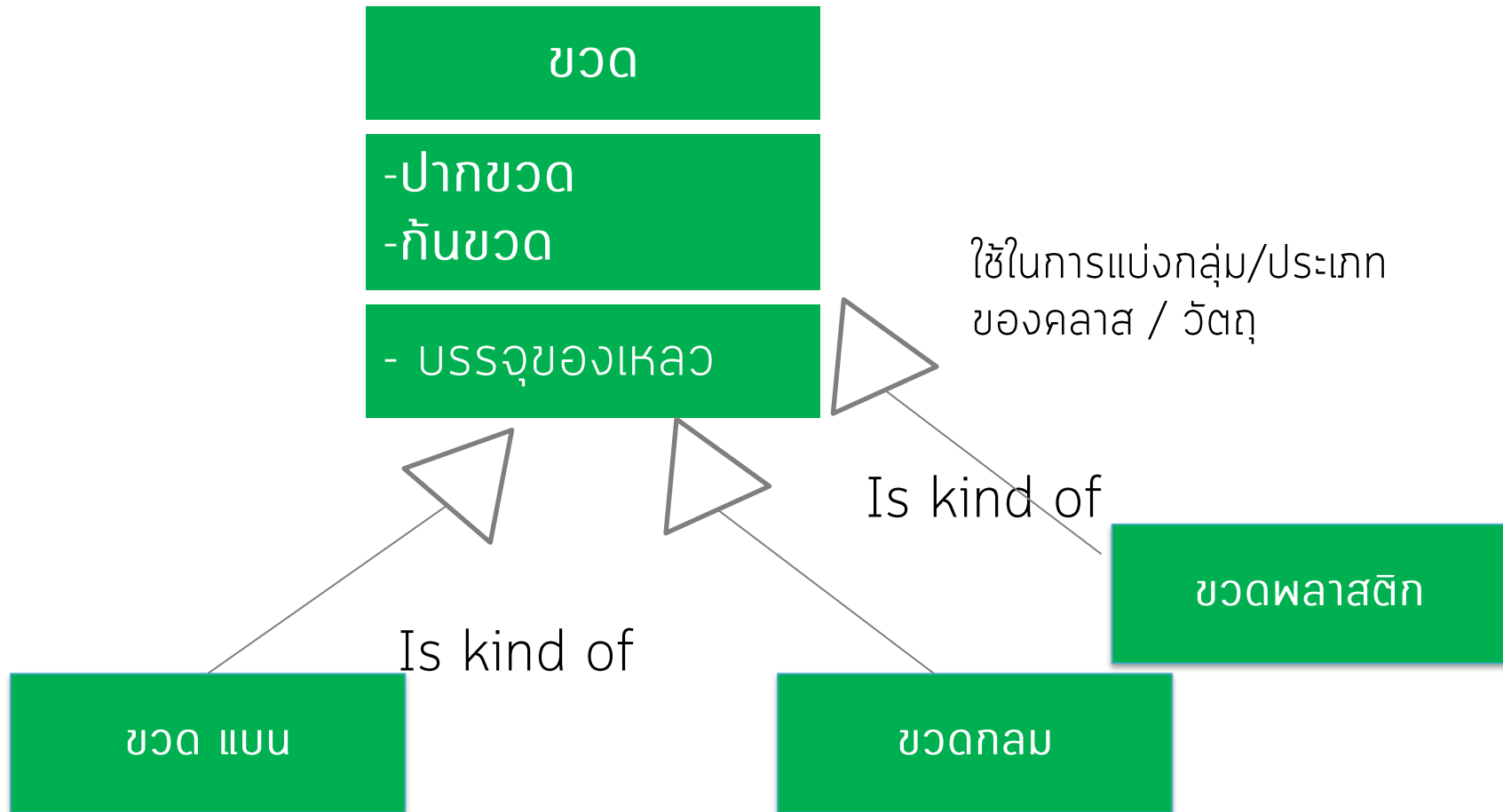


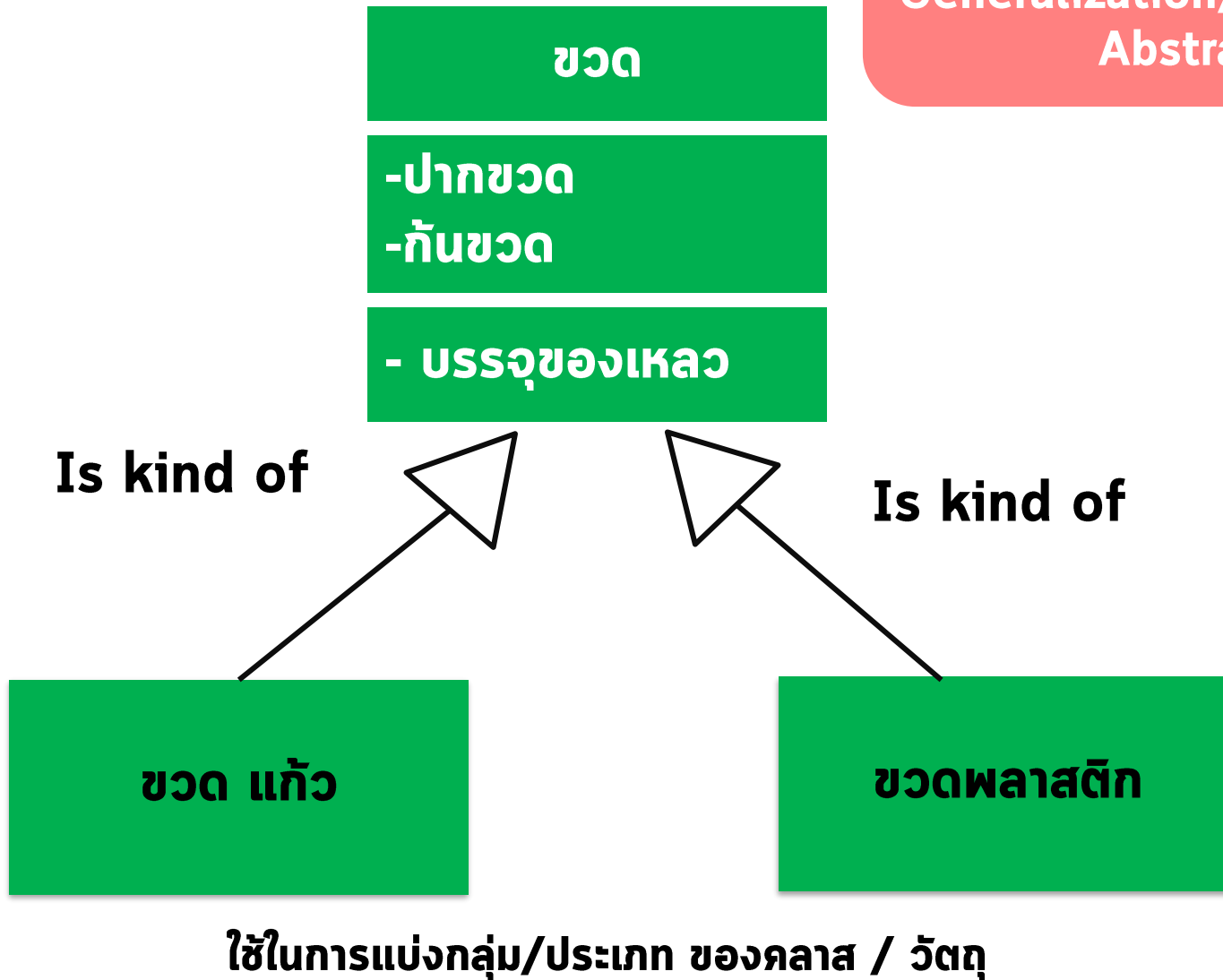
Aggregation Abstraction





Generalization/Specialization Abstraction



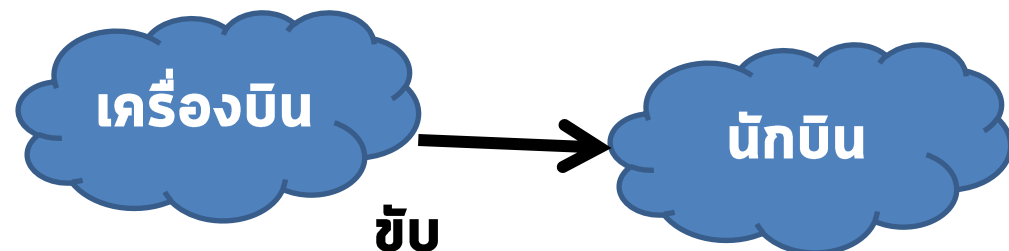
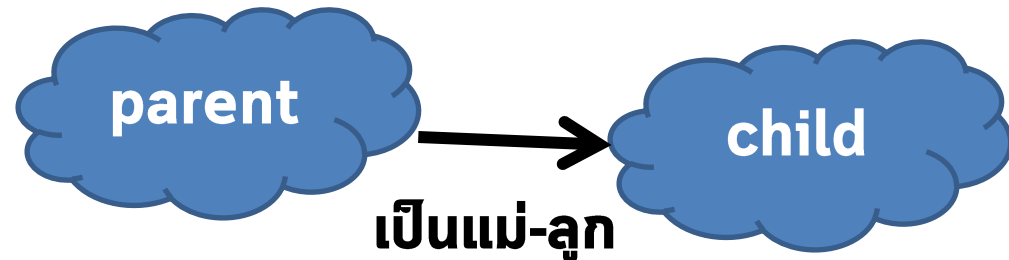
**Generalization/Specialization
Abstraction**

Association Abstraction

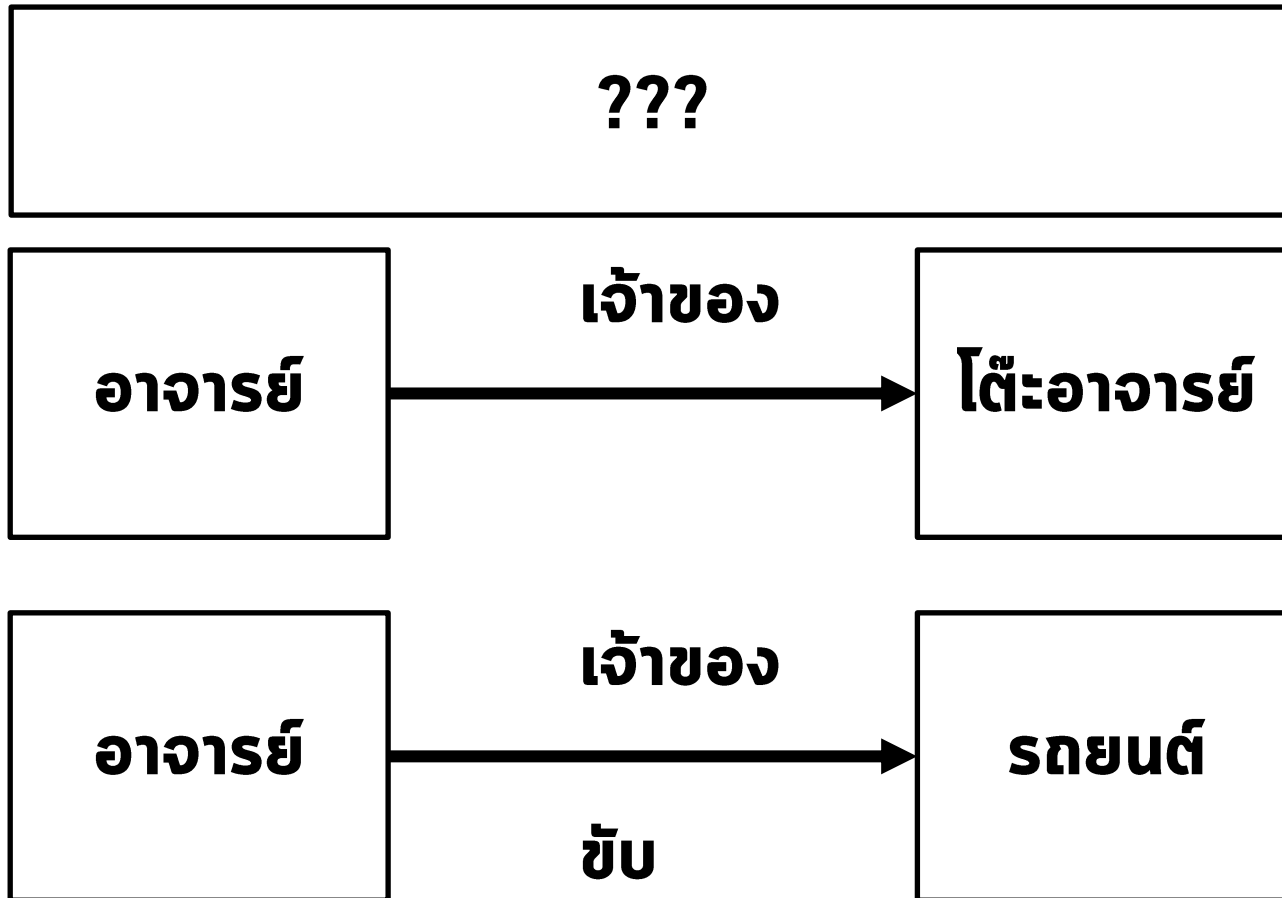
Relationship (ความสัมพันธ์)

- ตัวอย่างความสัมพันธ์ ต่าง ๆ ในชีวิตประจำวัน

- พ่อ-แม่กับลูก
- สามีกับภรรยา
- นายกรัฐมนตรี
- ตำรวจทำงานอยู่ที่สถานีตำรวจ
- น้าอยู่ในทะเล
- อาหารอยู่ในตู้เย็น
- คนสวนกำลังรดน้ำต้นไม้
- นักบินกำลังขับเครื่องบิน
- ตำรวจใส่เครื่องแบบ
- สมชายกำลังขับรถยนต์
- ปากกาอยู่บนโต๊ะทำงาน
- นักเรียนลงทะเลเรียนเรียน
- ลูกค้าสั่งซื้อสินค้า



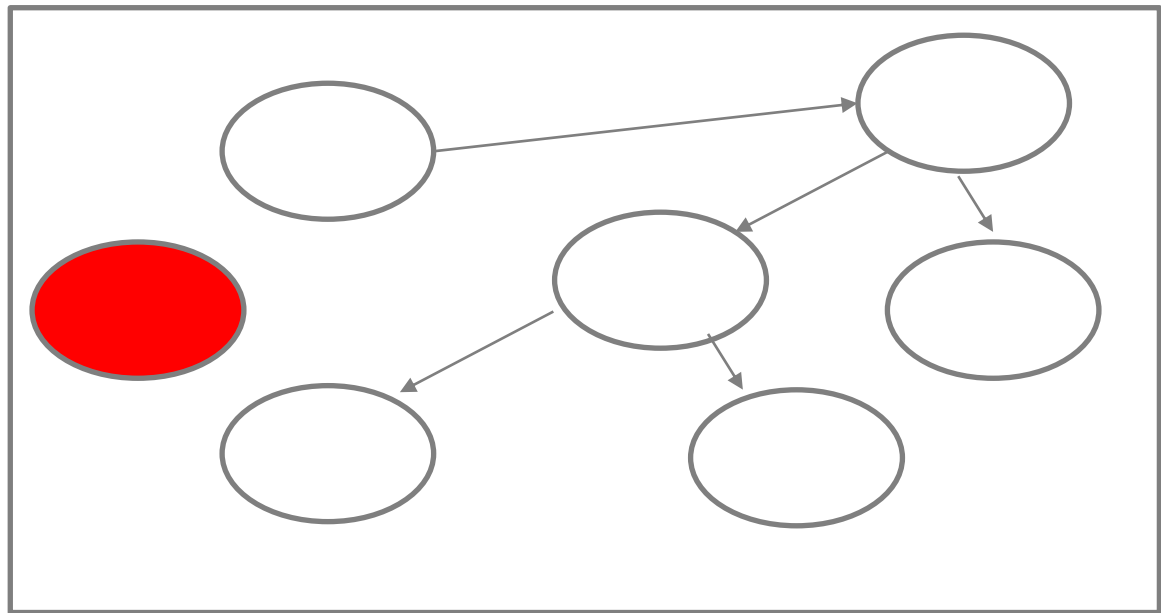
Association Abstraction



Association Abstraction

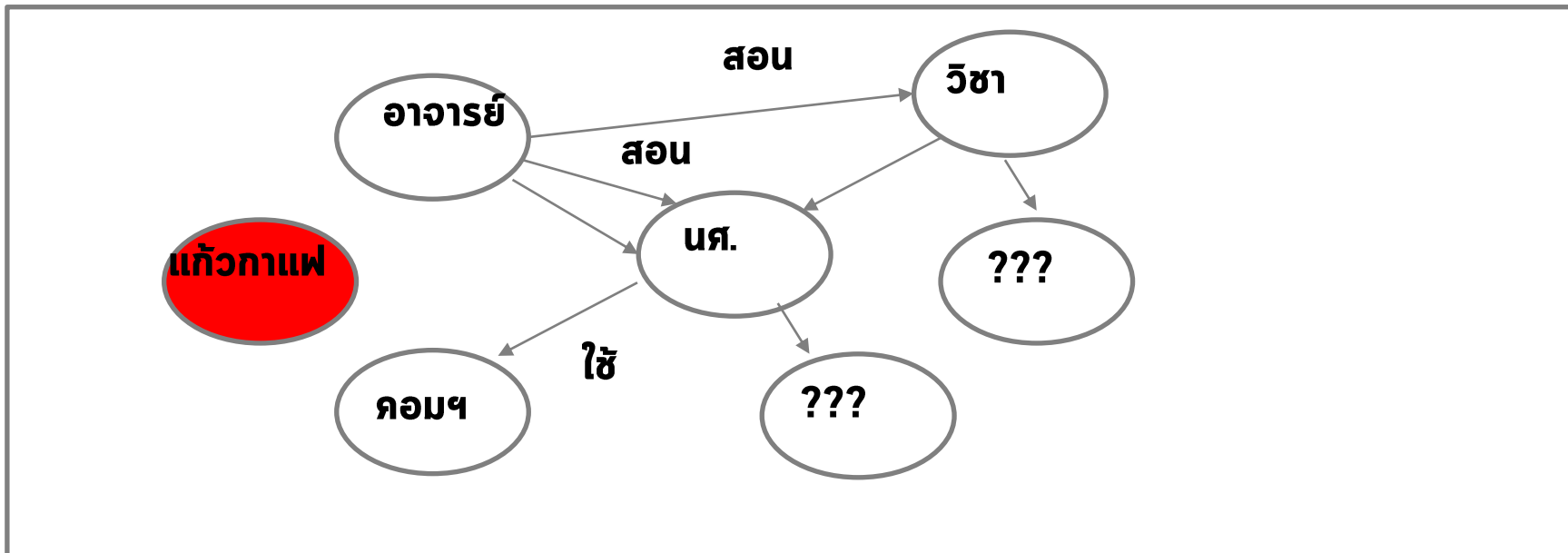
แนวคิด

- วัตถุ อยู่โดเมนเดียวกัน ย่อมมีความสัมพันธ์หรือความเกี่ยวข้อง ไม่เชิงใดก็เชิงหนึ่ง หรือในแง่ใดก็แง่หนึ่ง
 - ความเป็นเจ้าของ เช่น สมชายเป็นเจ้าของรถยนต์โตโยต้า altis เลขทะเบียน 4ก 4575
 - การมี เช่น นักศึกษา มีบัตรนักศึกษา , พนักงานมีบัตรประจำตัวพนักงาน ฯลฯ
 - การมีส่วนประกอบ(**aggregation**) รถยนต์มีน้ำมันอยู่ในถังน้ำมัน
 - การอยู่ ข้างใน อยู่ข้างบน รถยนต์วิ่งไปบนถนน อาหารอยู่ในตู้เย็น กาแฟอยู่ในแก้ว กาแฟ อาจารย์ทำงานอยู่ในมหาวิทยาลัย controlbox อยู่ใน Titlebar
 - การใช้ประโยชน์ สมหญิงกำลังพิมพ์รายงาน สมชายกำลังเล่นโทรศัพท์ อาจารย์กำลังขับรถยนต์ อาจารย์เปิดคอมพิวเตอร์
 - ความสัมพันธ์ระหว่างบุคคล เช่น ครอบครัว พี่น้อง เพื่อน เพื่อนร่วมงาน อาจารย์กับลูกศิษย์ ฯลฯ

Association Abstraction**Association abstraction**

Association Abstraction

Association abstraction

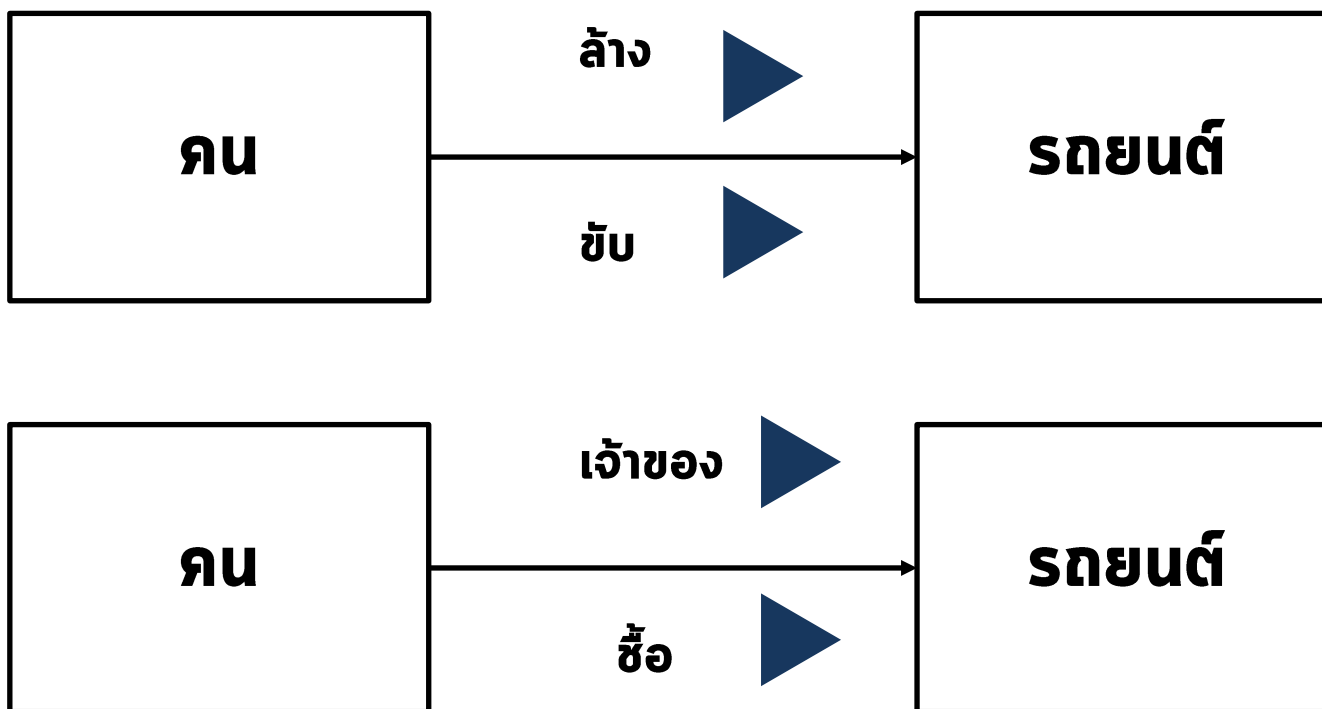


ระบบห้องเรียน

ระบบห้องเรียน ระบบการเรียนการสอน

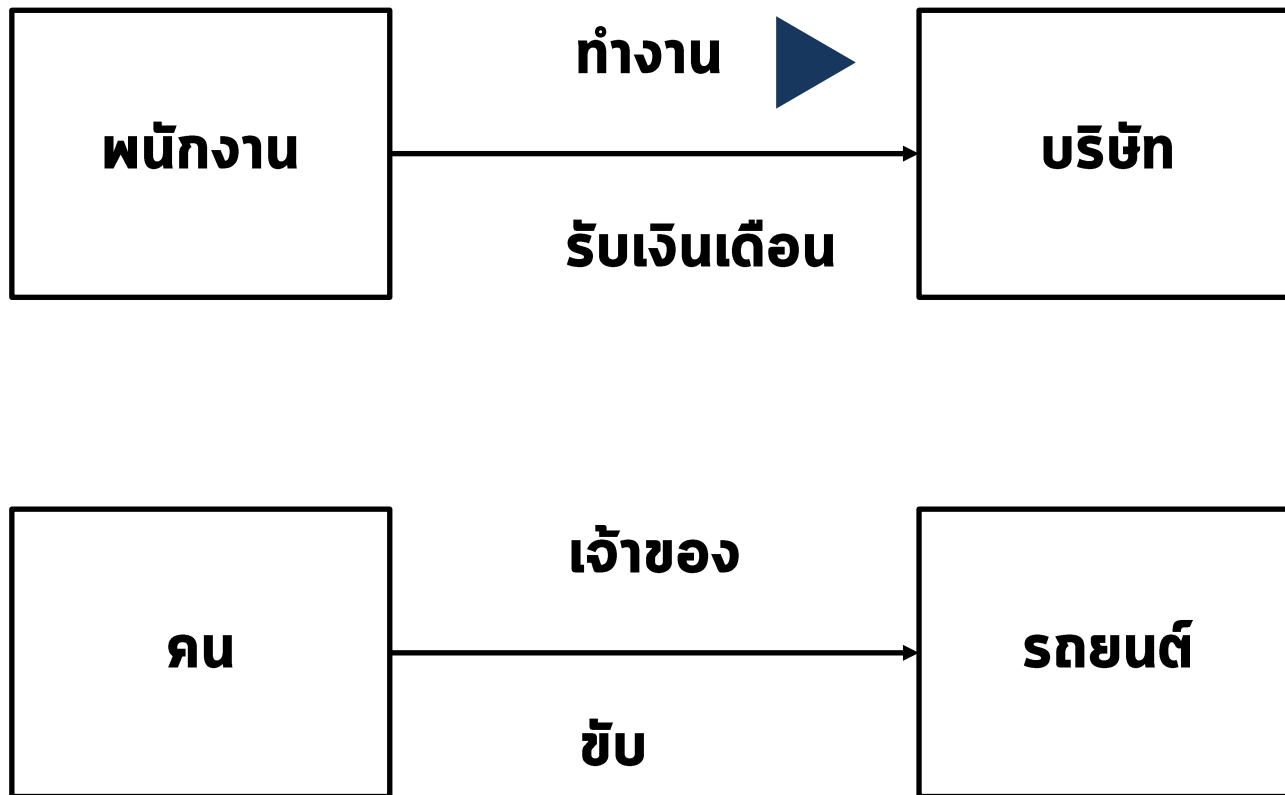
Association Abstraction

Association abstraction



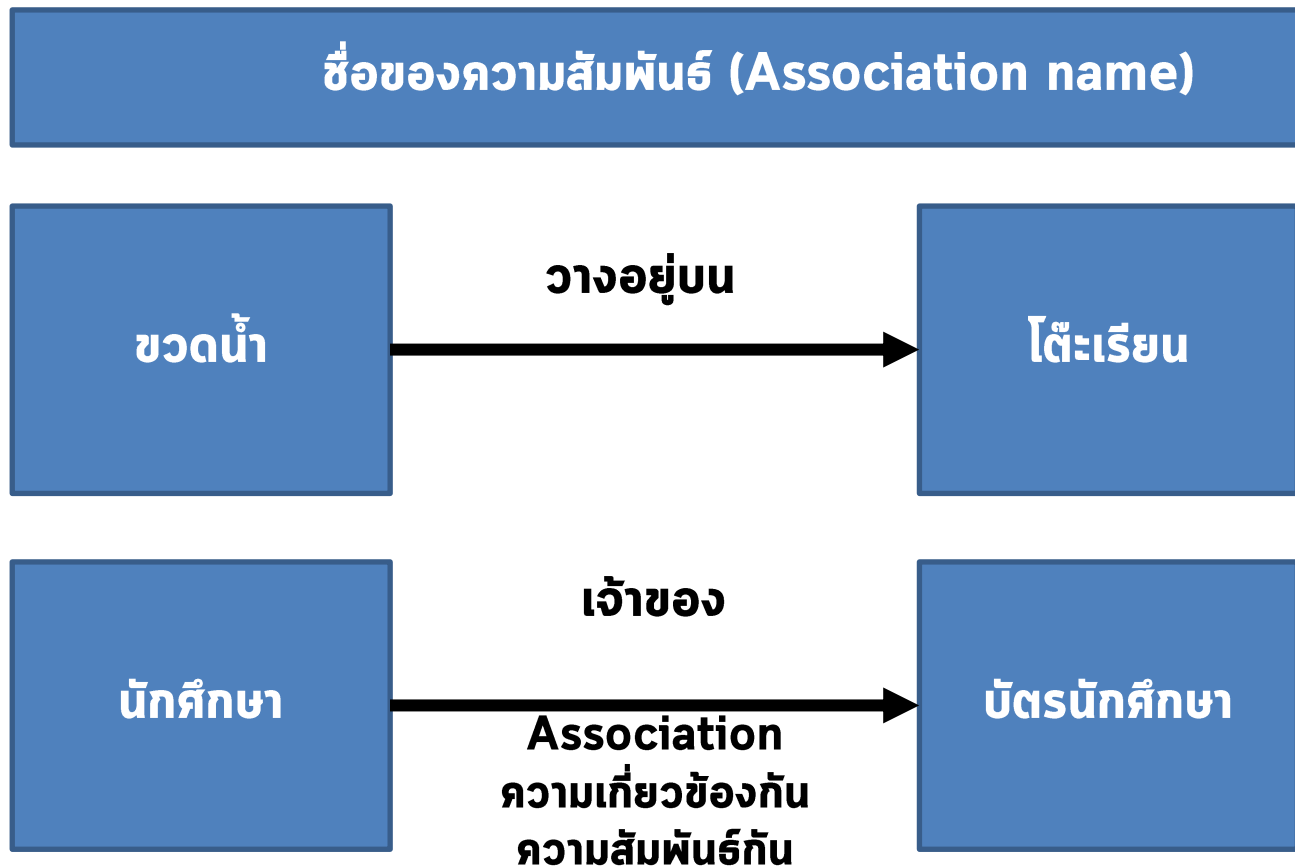
Association Abstraction

Association abstraction



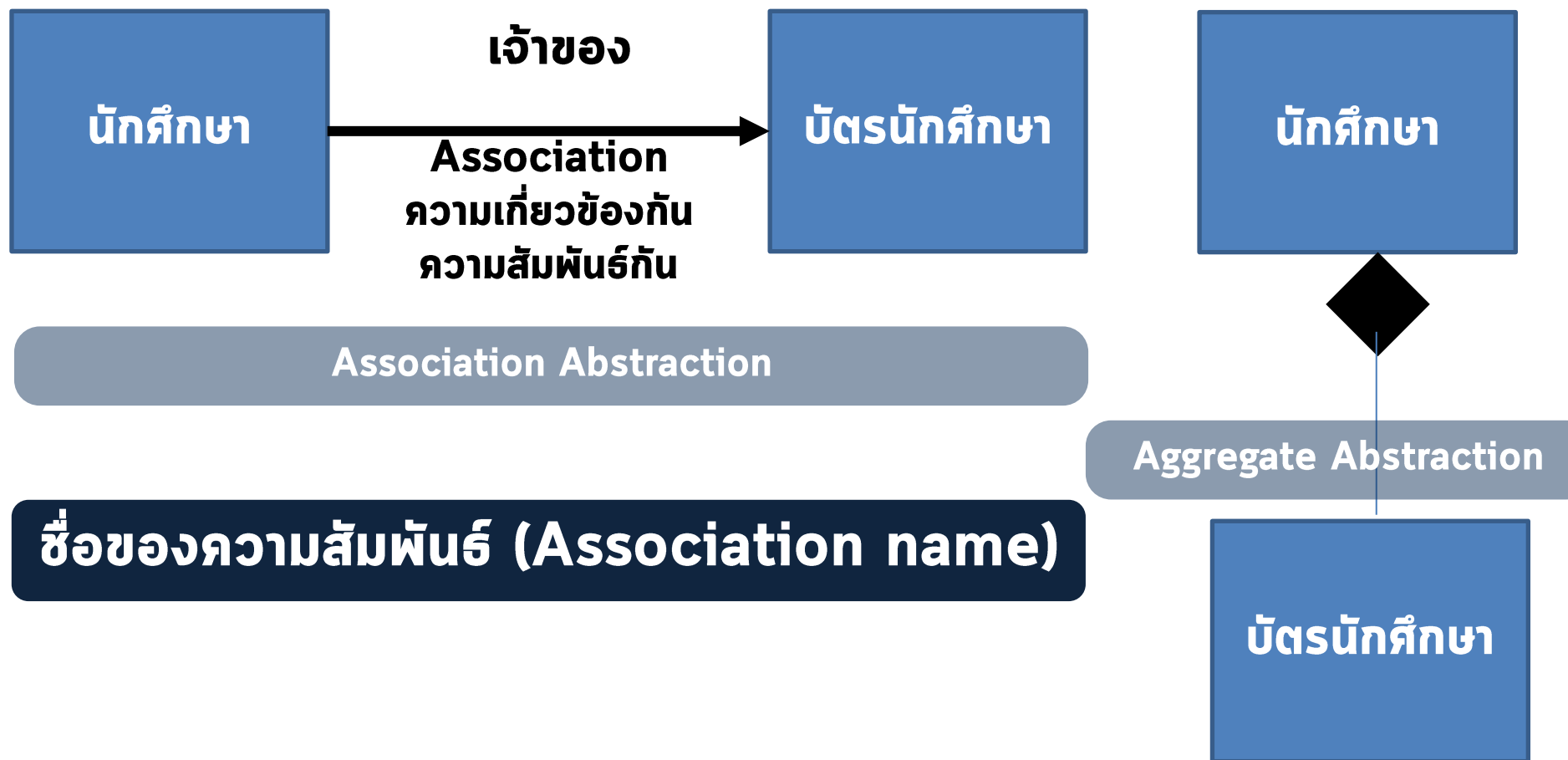
Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)



Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)

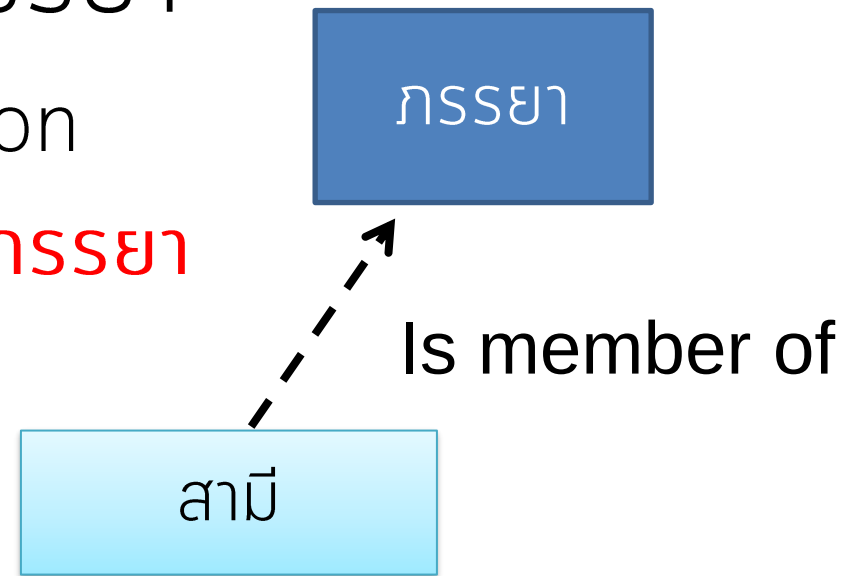
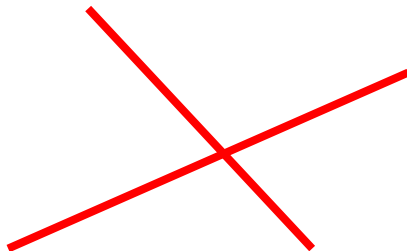


Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)

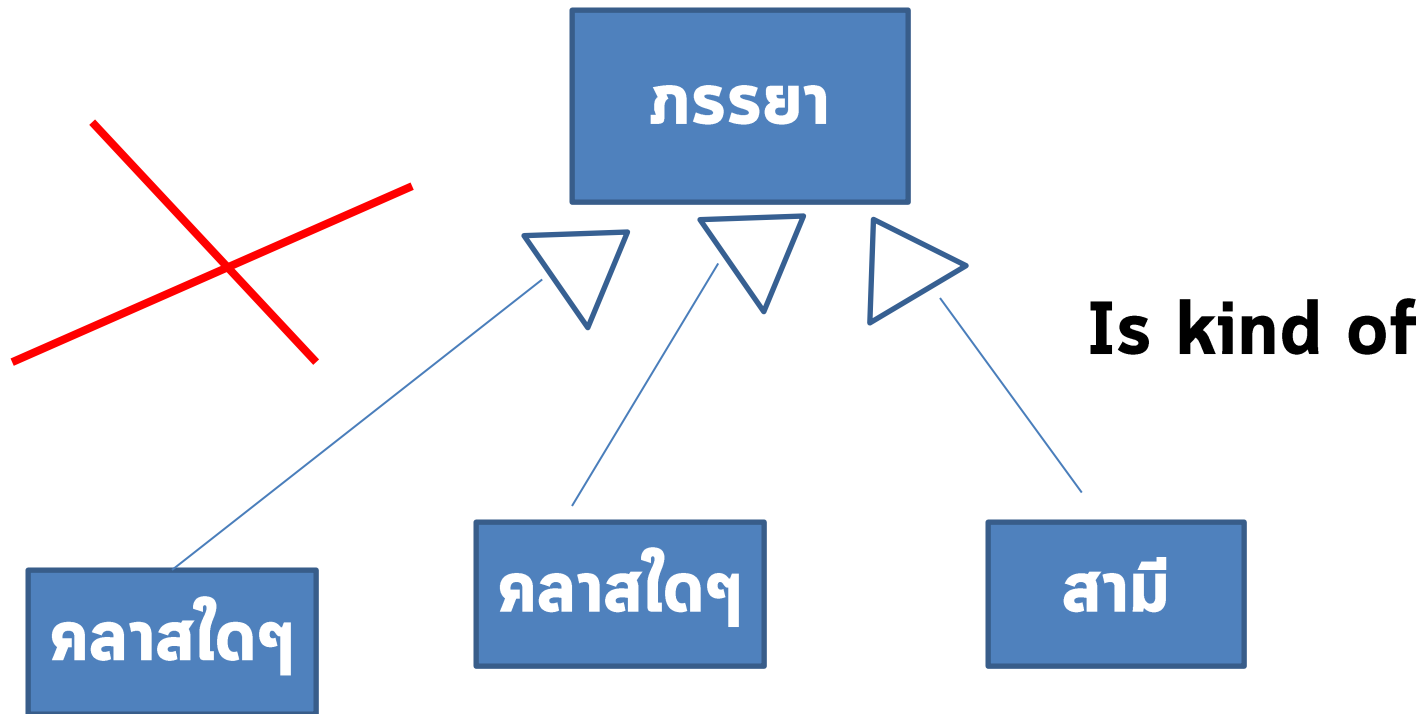
สมาชิกกับภรรยา

- Classification Abstraction
 - สมาชิกเป็นสมาชิกของคลาสภรรยา



สามีกับภรรยา

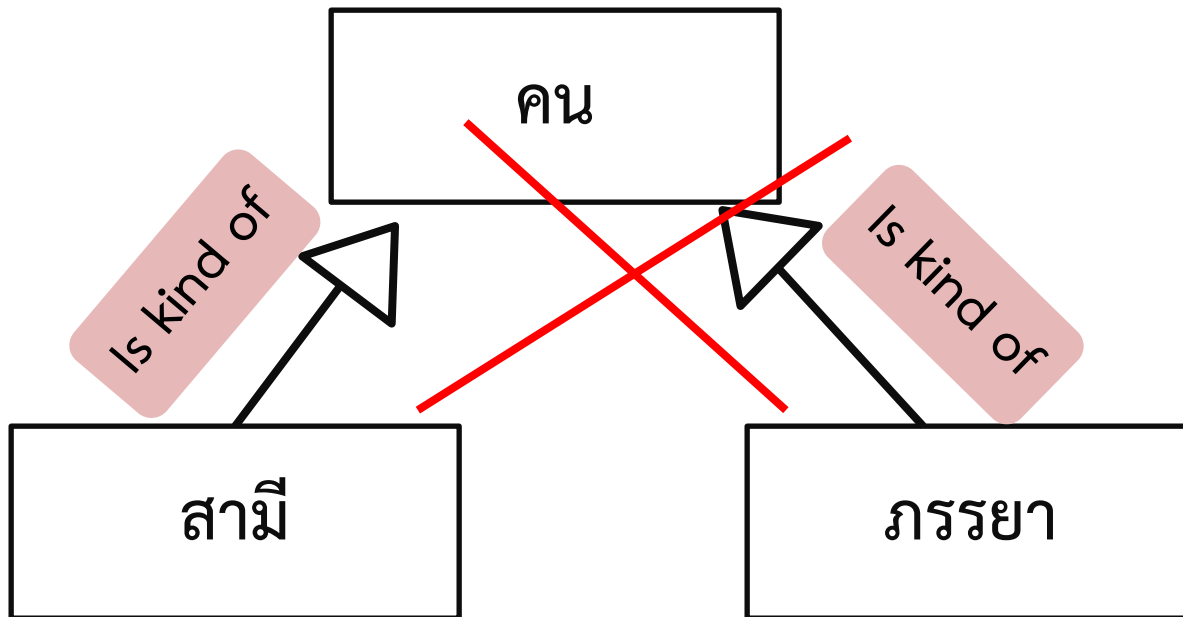
- Generalization Abstraction
 - สามีเป็นรวมกันกับคลาสใดกลายเป็นภรรยา



ສາມີກັບກຣຣຍາ



ไม่แนะนำให้เขียนแบบนี้ หรือไม่แนะนำให้อธิบายโดยใช้ Generalization Abstraction

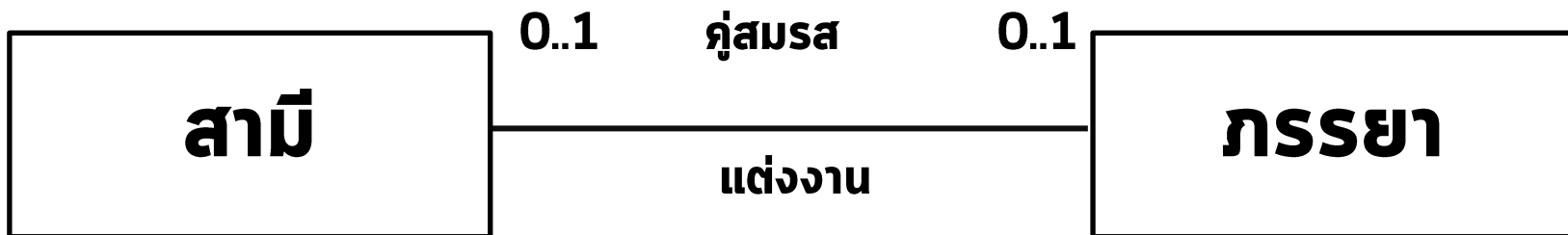


ควรอธิบายโดยใช้ *association abstraction* จะดีกว่า

Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)

การสมรส

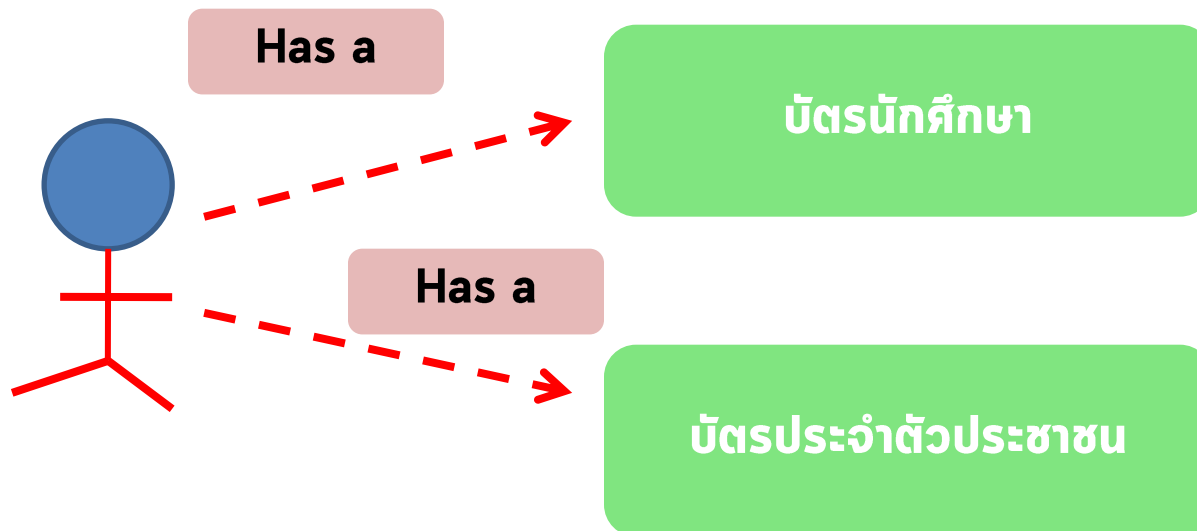


Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)

ข้อสังเกต (notice)

- เราสามารถจัดเอาความสัมพันธ์แบบ Aggregation เป็นความสัมพันธ์แบบ “**is related to**” ได้เช่นกัน เพราะการเป็นส่วนประกอบ หรือ การมีส่วนประกอบ ก็คือความสัมพันธ์เป็นเจ้าของ (has a) นั่นเอง



Association Abstraction

Association abstraction : การตั้งชื่อความสัมพันธ์ (Association naming)



การมองว่าบัตรนักศึกษา เป็นส่วนหนึ่งของนักศึกษา ก็มีความสัมพันธ์แบบ
Is part of หรือ *is relate to* ก็ได้เช่นกัน



การมองว่าเก้าอี้เป็นส่วนหนึ่งของห้องเรียน ก็มีความสัมพันธ์แบบ
Is part of หรือ *is relate to* ก็ได้เช่นกัน

Association Abstraction

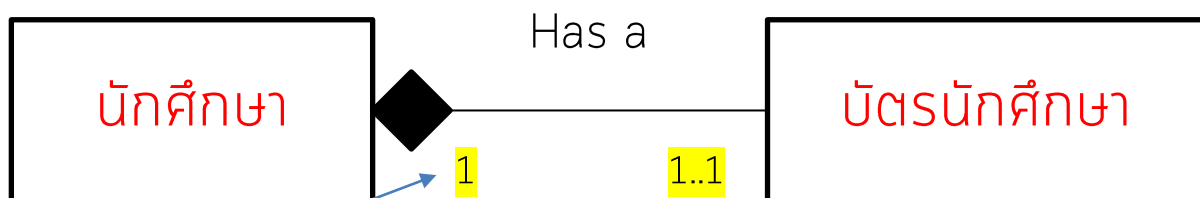
Association abstraction : Cardinality ใน Association Abstraction

- ตัวอย่าง 3.1.4.1 , 3.1.4.2

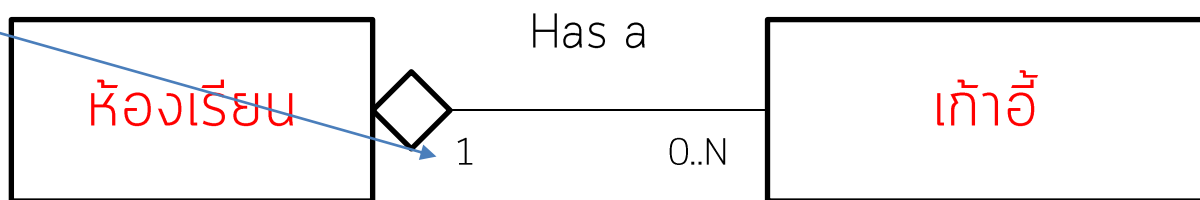
Class	Min Card	Max Card
ผู้หญิง	0 (โสด)	1 (แต่งงาน)
ผู้ชาย	0 (โสด)	1 (แต่งงาน)
แม่	1 (ลูกมีแม่ได้คนเดียว)	1 (ลูกมีแม่ได้คนเดียว)
ลูก	0 (เพราะแม่อาจไม่มีลูกก็ได้)	N (มีได้มากกว่า 1 คน)

Association Abstraction

Association abstraction : Cardinality ใน Association Abstraction



การมองว่าบัตรนักศึกษา เป็นส่วนหนึ่งของนักศึกษา ที่มีความสัมพันธ์แบบ
Is part of หรือ *is relate to* ก็ได้เช่นกัน



การมองว่าเก้าอี้เป็นส่วนหนึ่งของห้องเรียน ที่มีความสัมพันธ์แบบ
Is part of หรือ *is relate to* ก็ได้เช่นกัน

Association Abstraction

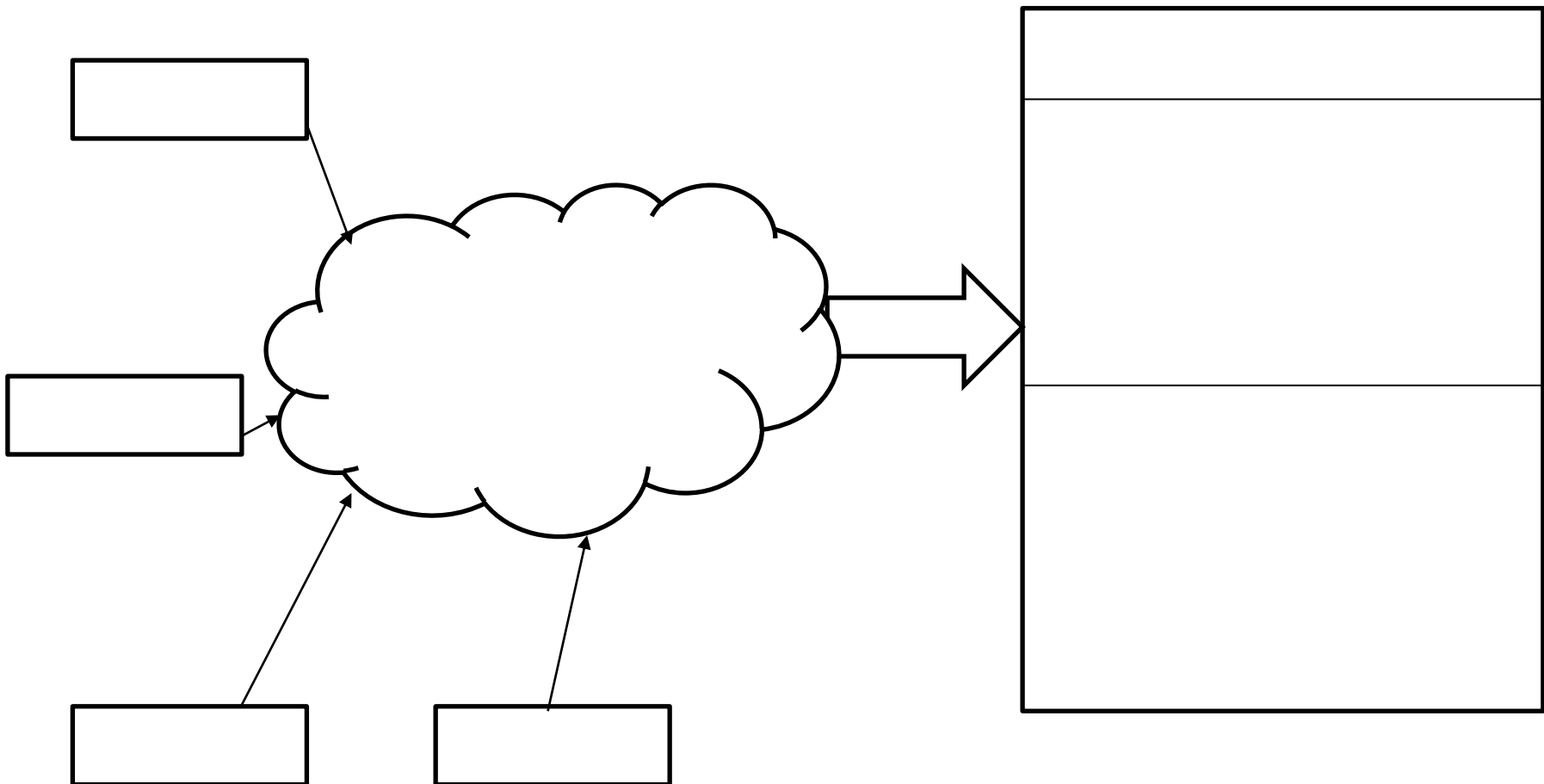
Association abstraction : **หลักในการเขียน Diagram แสดง Association**

ในการเขียนภาพเพื่อแสดง Association นั้น มีหลักการเขียนดังนี้

1. หาดำนาม หรือ หาดลาส จากการทำ classification abstraction
2. ลากเชื่อมต่อระหว่าง class ทั้งสอง
3. เขียนลูกศรเพื่อบอกทิศทางของการอ่านความสัมพันธ์
4. พิจารณา Class ที่อยู่ติดกับหัวลูกศร หาค่า Min Card, Max Card
5. พิจารณา Class ที่อยู่อีกด้านของหัวลูกศร หาค่า Min Card, Max Card
6. เขียนความสัมพันธ์ระหว่างทุกคลาส
7. ใส่รายละเอียดทั้ง attribute , function , relationship
8. จนได้ภาพของการแสดง Association ดังรูป

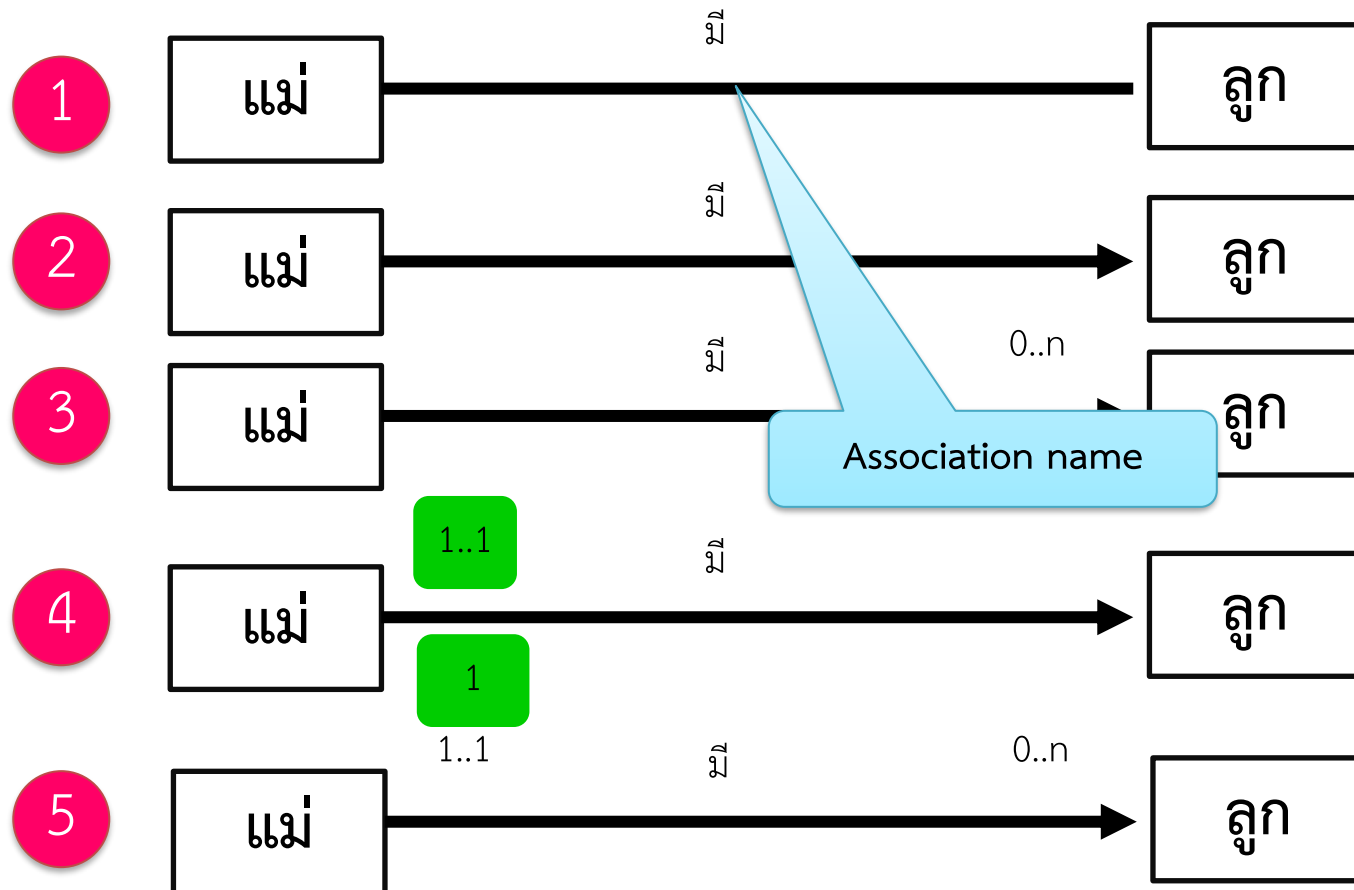
Association Abstraction

Association abstraction : หาคำนามหรือหาคลาสจากการทำ classification abstraction



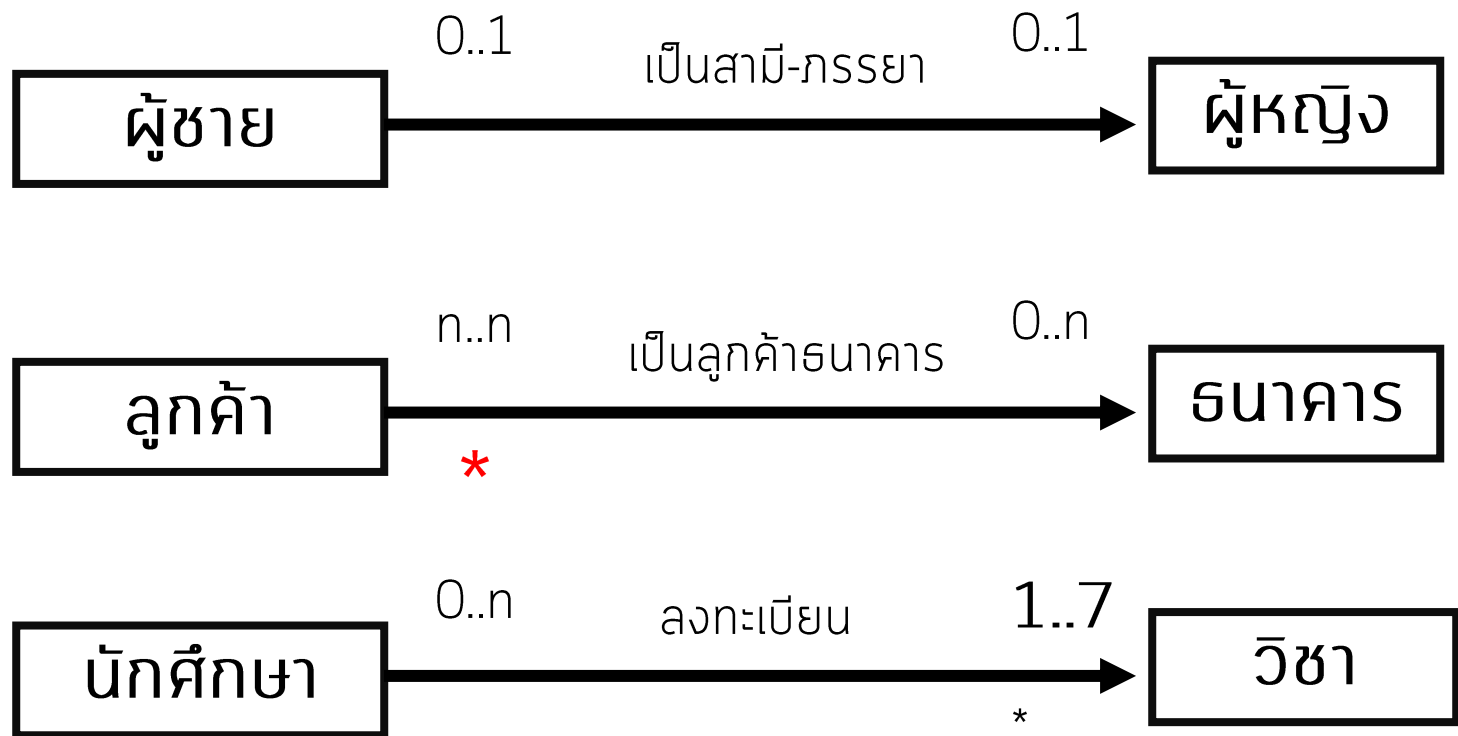
Association Abstraction

Association abstraction : **ลากเชื่อมต่อระหว่าง class ทั้งสอง**



Association Abstraction

Association abstraction : ตัวอย่าง Association ต่าง ๆ



Association Abstraction

Association abstraction : ตัวอย่าง Association ต่าง ๆ



Business Rule

Association Abstraction

Association abstraction : ตัวอย่าง Association ต่าง ๆ



Association Abstraction

Association abstraction : แบบฝึกหัด 2.4.2 จงเขียนแผนภาพ association จากโดเมนที่กำหนดให้ต่อไปนี้

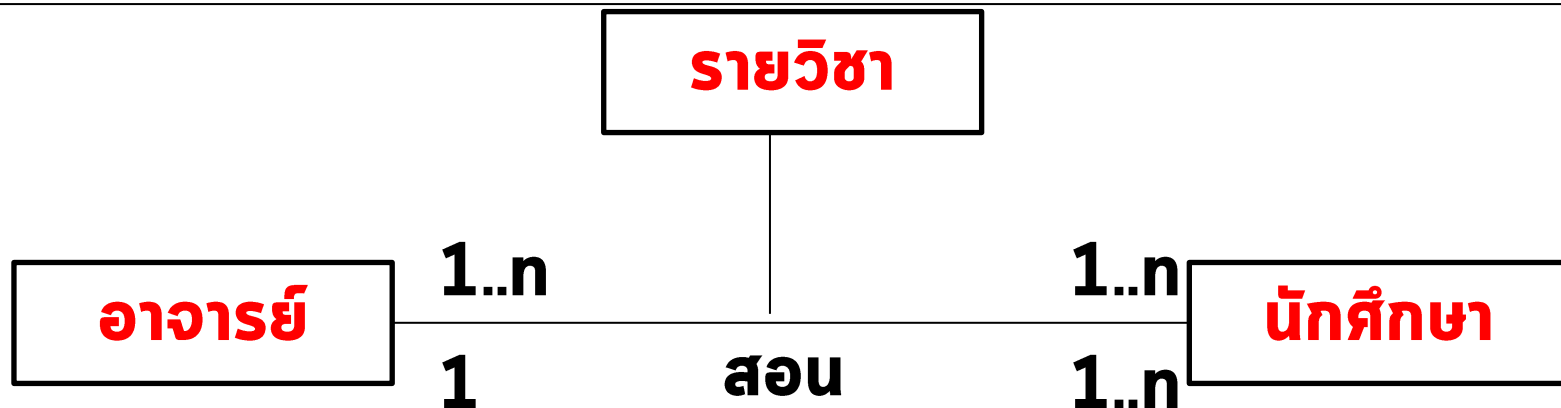
1. อาจารย์สอนนักศึกษา
2. อาจารย์นัฐพงศ์ สอนวิชา ooad ให้ นักศึกษาวิทยาลัยคอมรูน CS55
3. สมชาย ไปกดเงินสดที่ตู้ ATM ของธนาคารกสิกร
4. พนักงาน ลาหยุดงาน
5. สมหญิงคุยโทรศัพท์มือถือผ่านทางทรูออนไลน์
6. สมศรีพิมพ์รายงานด้วยโปรแกรม ms-word แล้วส่งพิมพ์ไปยังเครื่องพิมพ์

Association Abstraction

Association abstraction : แบบฝึกหัด 2.4.2 จงเขียนแผนภาพ association จากโดเมนที่กำหนดให้ต่อไป

อาจารย์นัฐพงศ์ สอนวิชา ooad ให้ นักศึกษาวิทย์คอมรุ่น 63

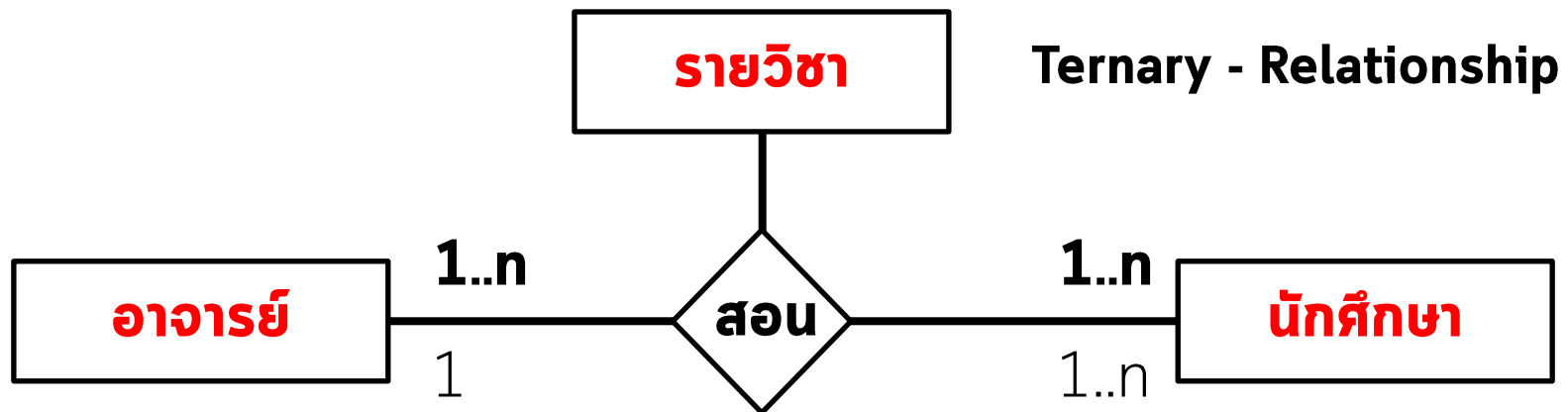
วิเคราะห์ นัฐพงศ์ เป็นวัตถุของคลาส อาจารย์
 ooad เป็นวัตถุของคลาส รายวิชา
 วิทย์คอมรุ่น 63 เป็นวัตถุของคลาส นักศึกษา
 สรุป มี 3 คลาสที่เกี่ยวข้องกัน



Association Abstraction

Association abstraction : แบบฝึกหัด 2.4.2 จงเขียนแผนภาพ association จากโดเมนที่กำหนดให้ต่อไปนี้

วิเคราะห์ นั้ฐพงศ์ เป็นวัตถุของคลาส อาจารย์
 อดิ เป็นวัตถุของคลาส รายวิชา
 วิทยากรณ์ 63 เป็นวัตถุของคลาส นักศึกษา
 สรุป มี 3 คลาสที่เกี่ยวข้องกัน

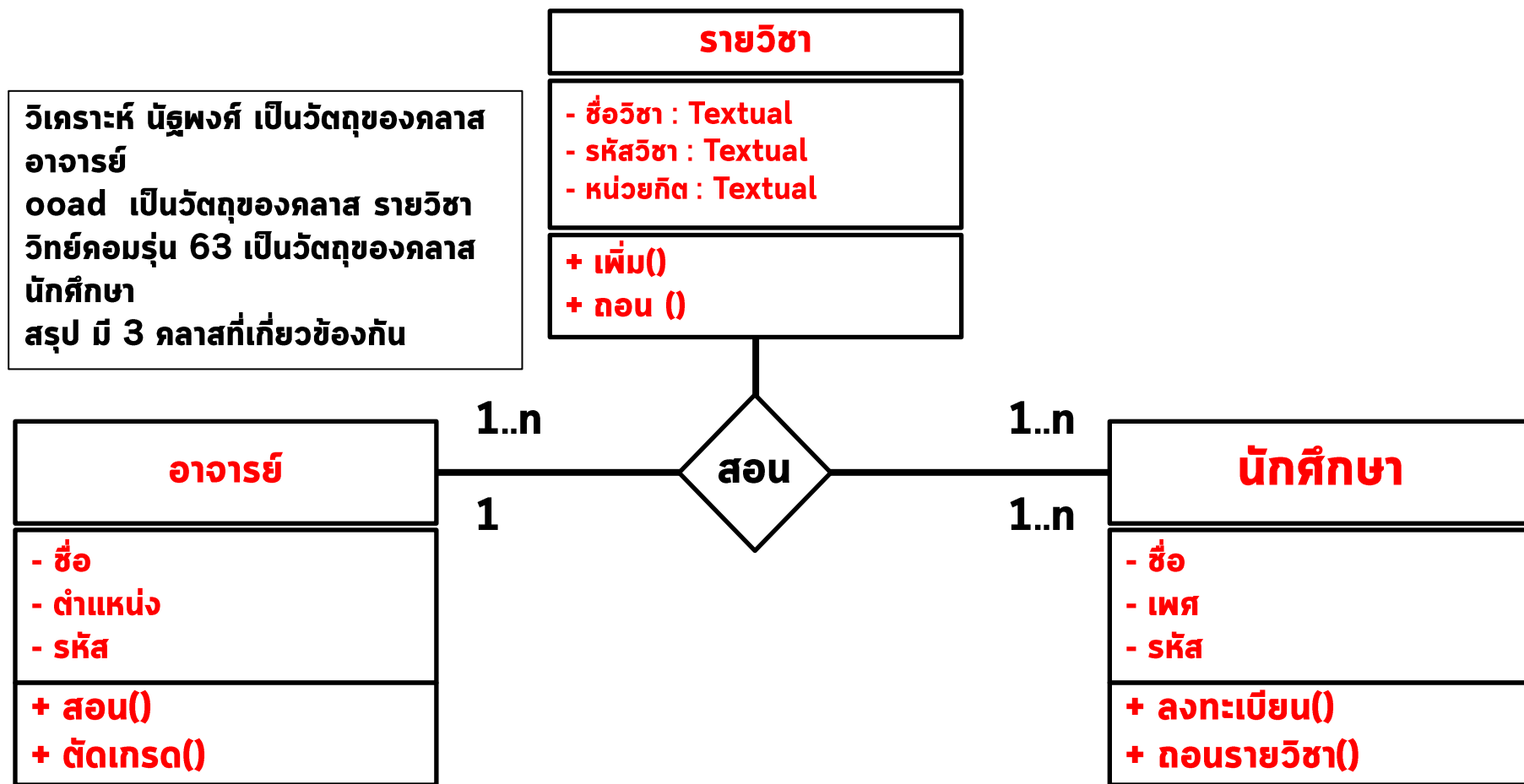


แผนภาพนี้เป็นแผนภาพคลาส ไม่ใช่ ERD

Association Abstraction

Association abstraction : แบบฝึกหัด 2.4.2 จงเขียนแผนภาพ association จากโดเมนที่กำหนดให้ต่อไปนี้

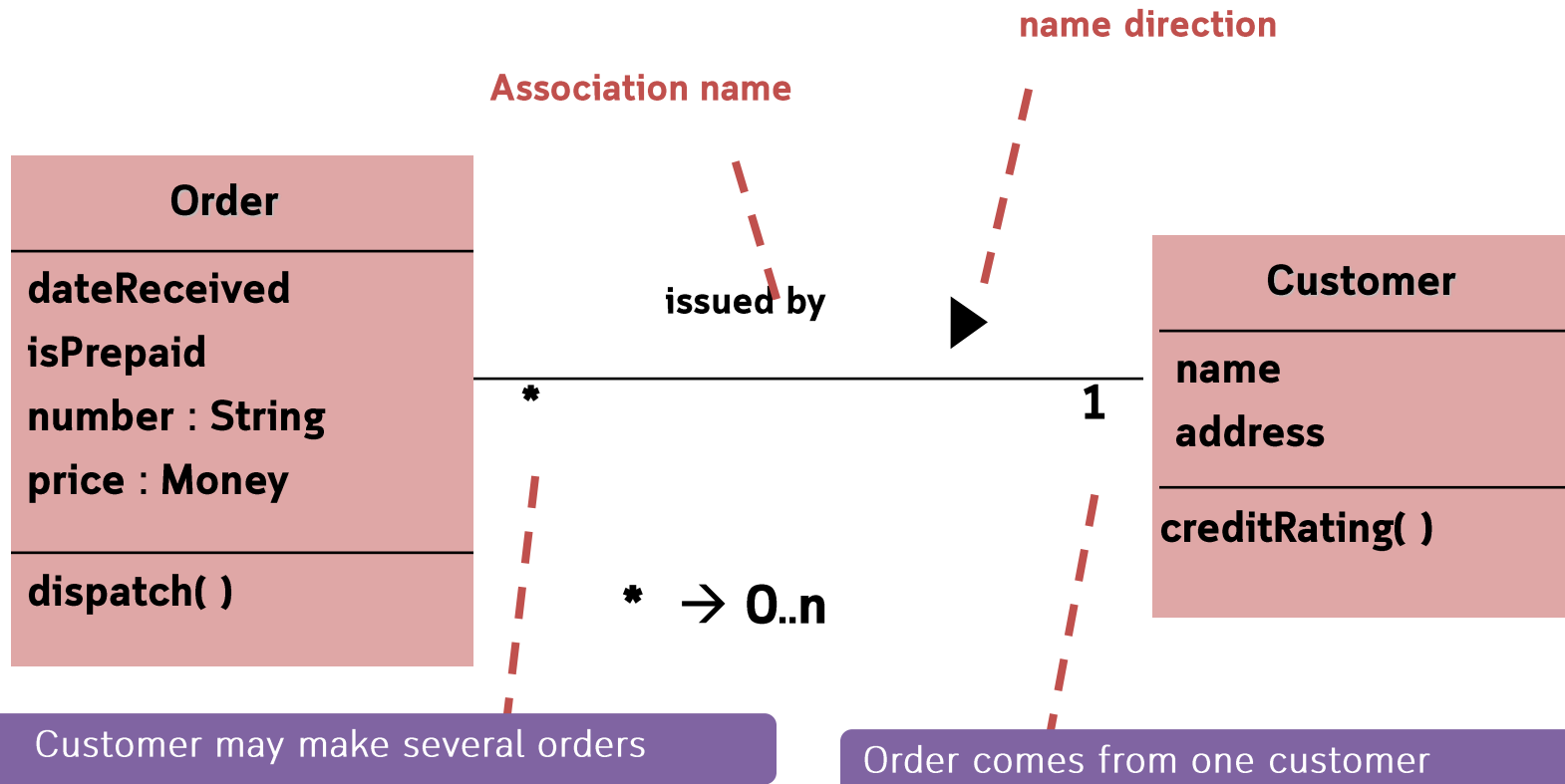
อาจารย์นัฐพงศ์ สอนวิชา ooad ให้ นักศึกษาวิทยาคอมรุ่น 63



Association Abstraction

Association abstraction : ความสัมพันธ์ (Relationship) ระหว่าง instances ของ classes

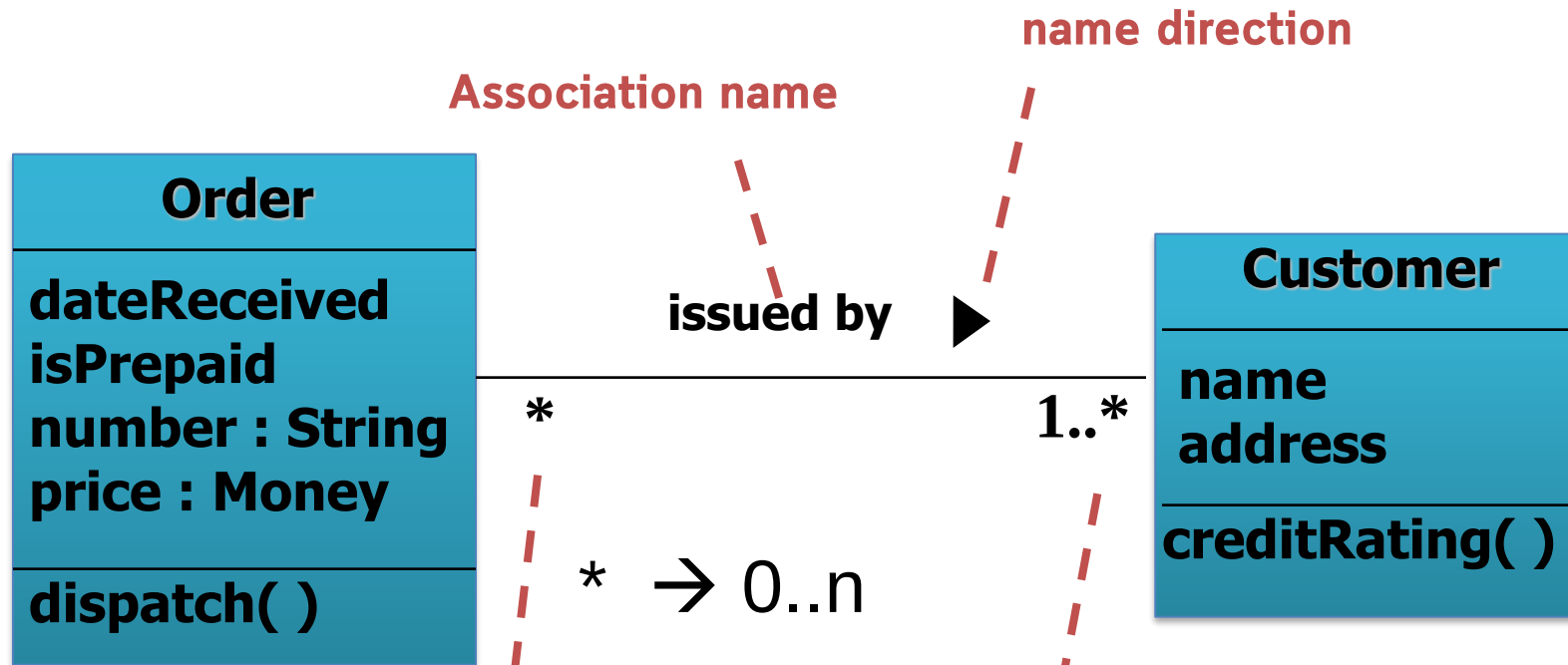
- ความสัมพันธ์ (Relationship) ระหว่าง instances ของ classes



Association Abstraction

Association abstraction : ความสัมพันธ์ (Relationship) ระหว่าง instances ของ classes

- ความสัมพันธ์ (Relationship) ระหว่าง instances ของ classes



Customer may make several orders

Order comes from several customers

Association Abstraction

Association abstraction : ความสัมพันธ์ (Relationship)

- คลาสมากกว่า 1 คลาส สัมพันธ์กันด้วย “associations”
- **Association**
 - เทียบได้กับ ER relationship ที่เชื่อมโยงความสัมพันธ์ระหว่างคลาส และแสดง จุดเชื่อมโยง (“links”) ระหว่าง วัตถุที่สร้างขึ้นจากคลาส
- ระหว่างวัตถุกับวัตถุ จะเรียกว่า Link
- ระหว่างคลาสกับคลาส เรียกว่า Association

Association Abstraction

Association abstraction : กฎการตั้งชื่อความสัมพันธ์ (Association Naming)

- **การตั้งชื่อ association**

- หลีกเลี่ยงชื่อที่ไม่สื่อความหมาย

- associated_with, has, is_related_to

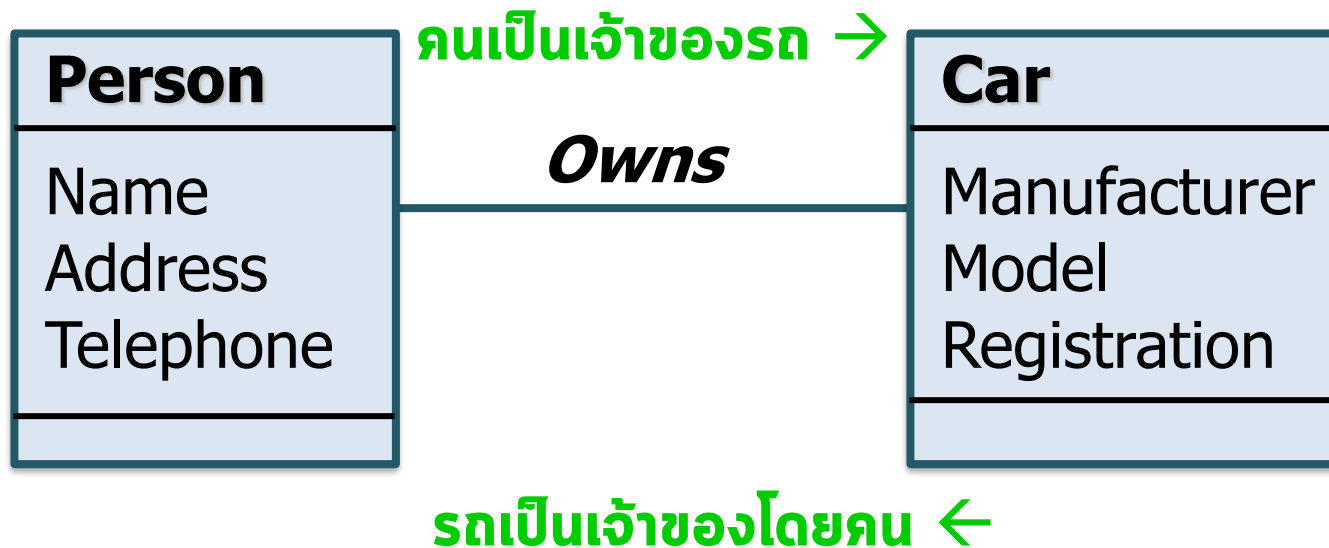
- มักขึ้นต้นด้วยคำกริยา เช่น

- works_for, owns , issued by

- มักตั้งชื่อในกรณีที่มีอง Association จากซ้ายไปขวา หรือจากบนลงล่างของ diagram

Association Abstraction

Association abstraction : Associations on Class Diagrams



ไม่ได้เขียนหรือกำกับหัวลูกศรแสดงว่าเป็นได้ 2 ทิศทาง

Association Abstraction

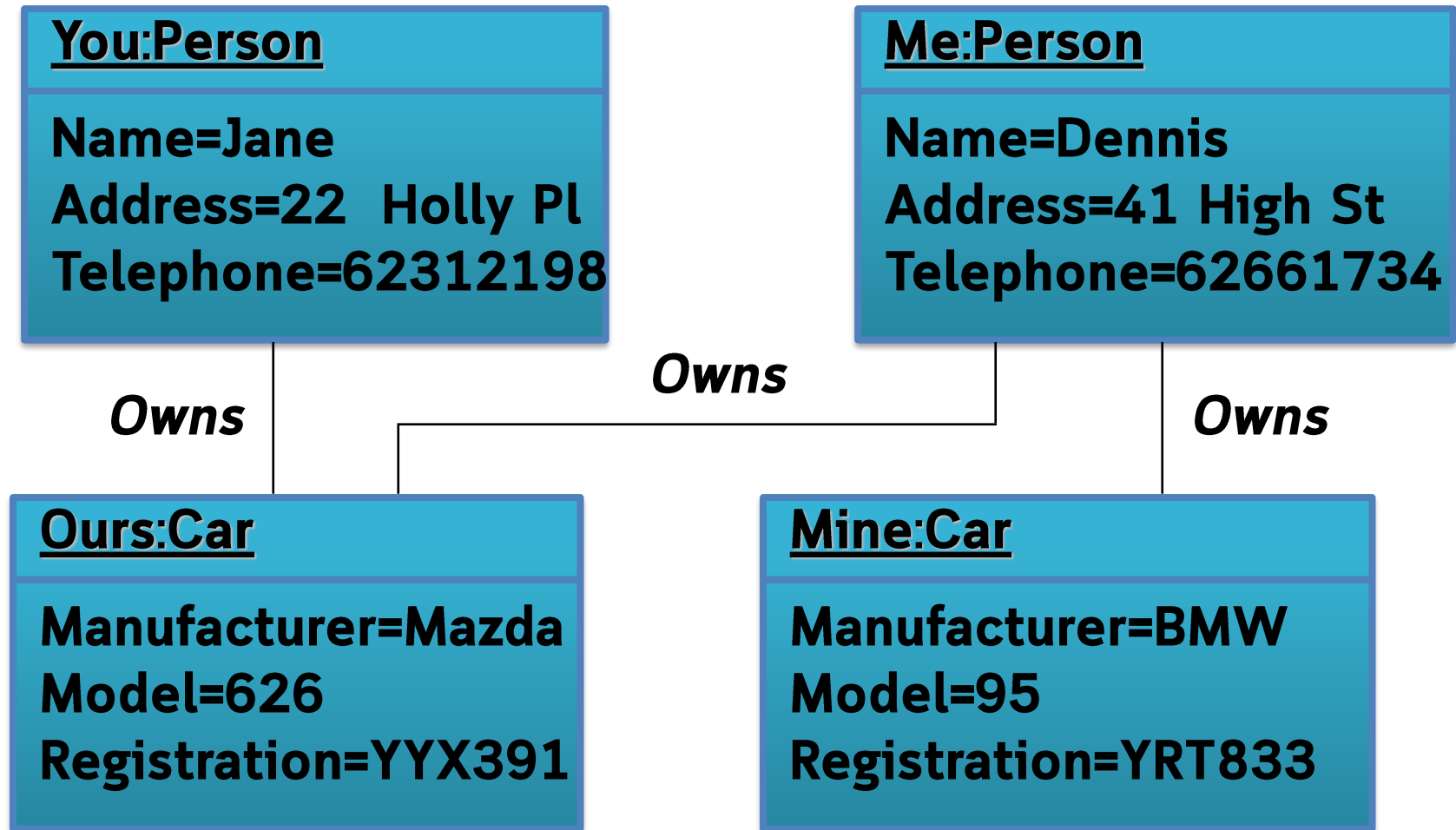
Association abstraction : **Links**

- วัตถุแต่ละวัตถุ สามารถเชื่อมโยงกันได้ โดยใช้ “links”
- **A link**
 - เป็น instance ของ association
- **หมายเหตุ**
 - an association สามารถดำรงอยู่ได้ ถึงแม้ว่าจะไม่มี instance (links) ของ association ดังกล่าว เช่นเดียวกับ คลาสที่สามารถดำรงอยู่ได้ ถึงแม้ว่าจะไม่มีวัตถุใด ๆ ถูกสร้างขึ้นจากคลาสนั้น ๆ

Class = Person , Car

Object = You,Me,Ours , Mine

Objects & Links

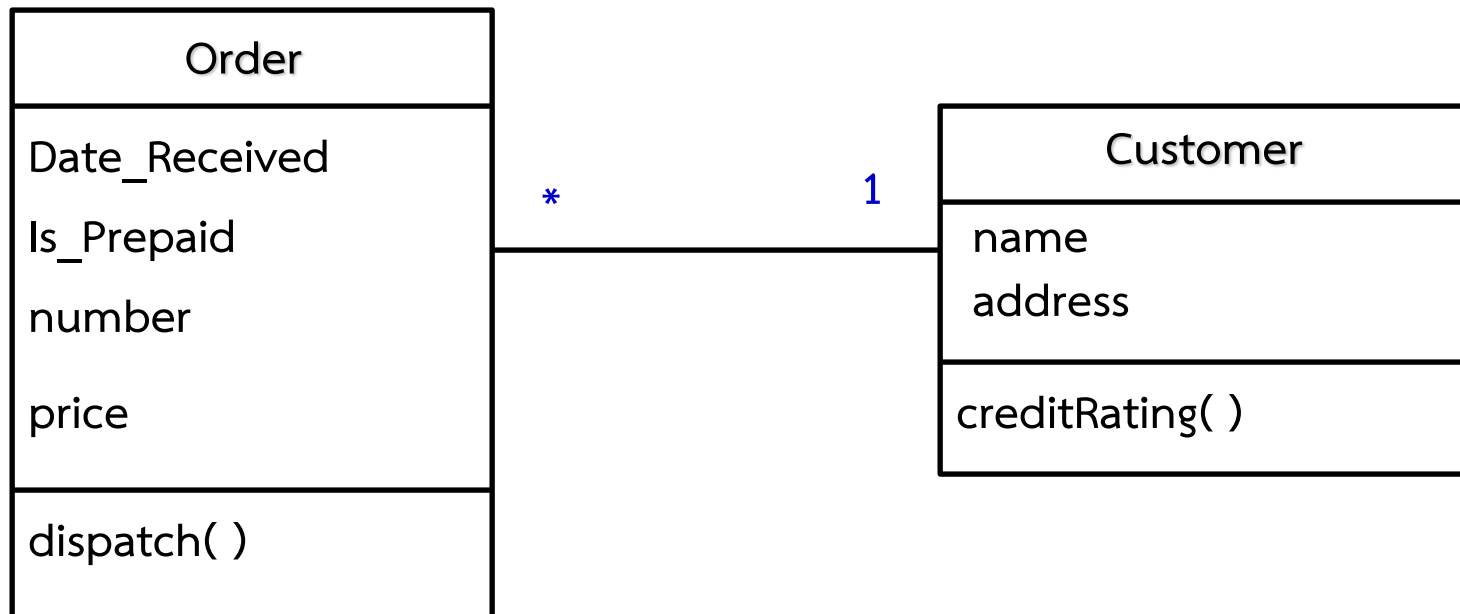


Association Abstraction

Association abstraction : **Association Multiplicity**

- **Multiplicity/cardinality**

- หมายถึง การพิจารณาจำนวน instances (objects) ของคลาสหนึ่ง ที่สามารถเชื่อมโยงกับ instance (object) ของคลาสที่เกี่ยวข้อง



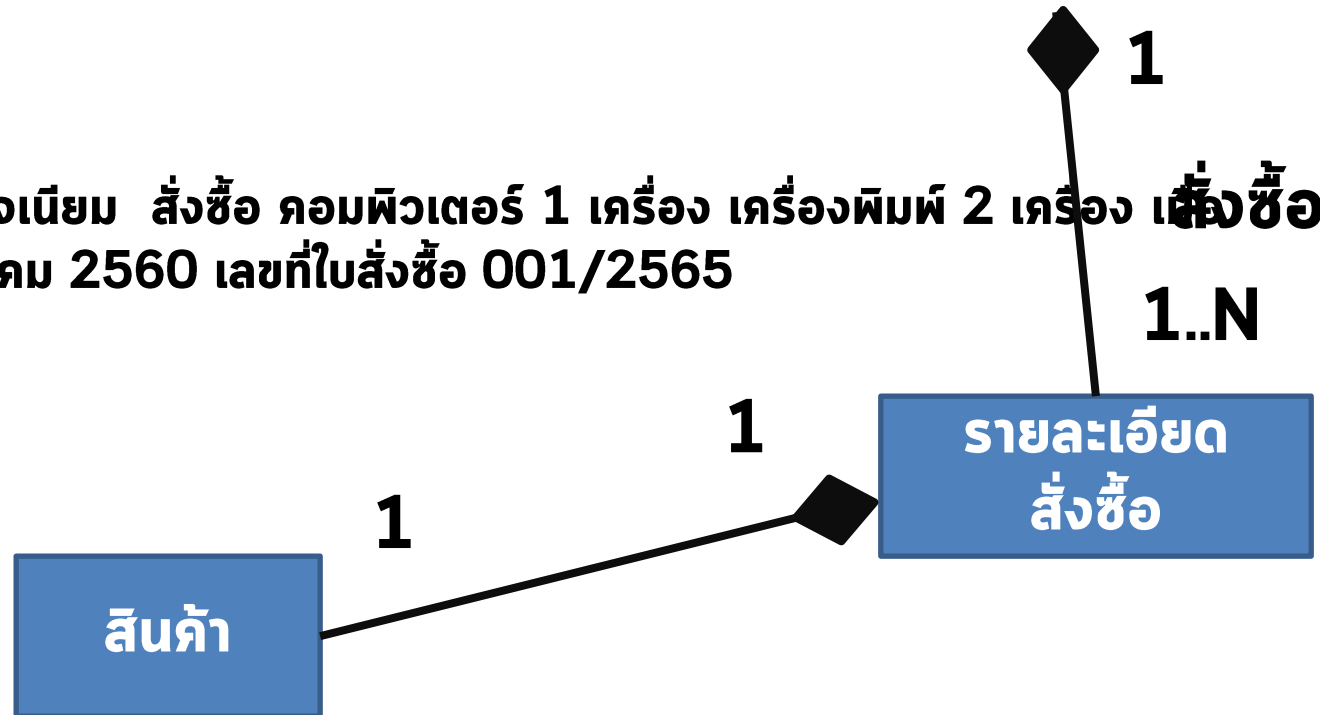
ความสัมพันธ์

- ลูกค้า กับ ใบสั่งซื้อ

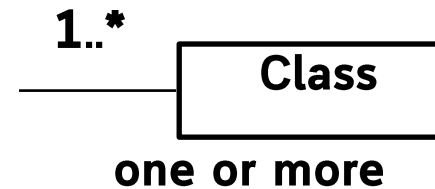
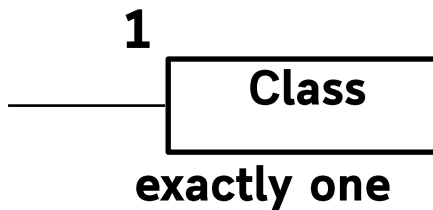
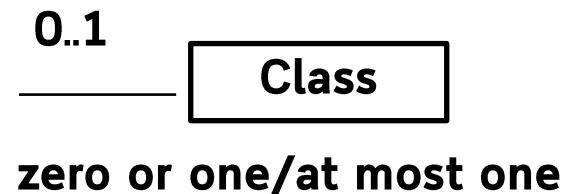
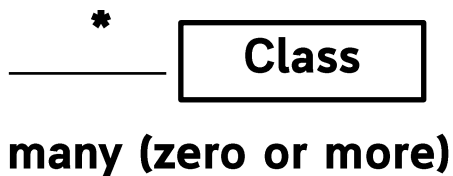


- Scenario

- นาย นัฐพงศ์ ส่งเนียม สั่งซื้อ คอมพิวเตอร์ 1 เครื่อง เครื่องพิมพ์ 2 เครื่อง ใบสั่งซื้อ วันที่ 20 สิงหาคม 2560 เลขที่ใบสั่งซื้อ 001/2565

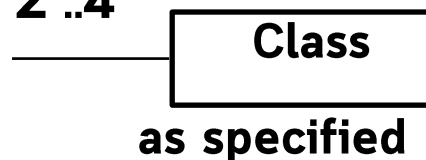


Association Abstraction

Association abstraction : **Association Multiplicity****0..n**

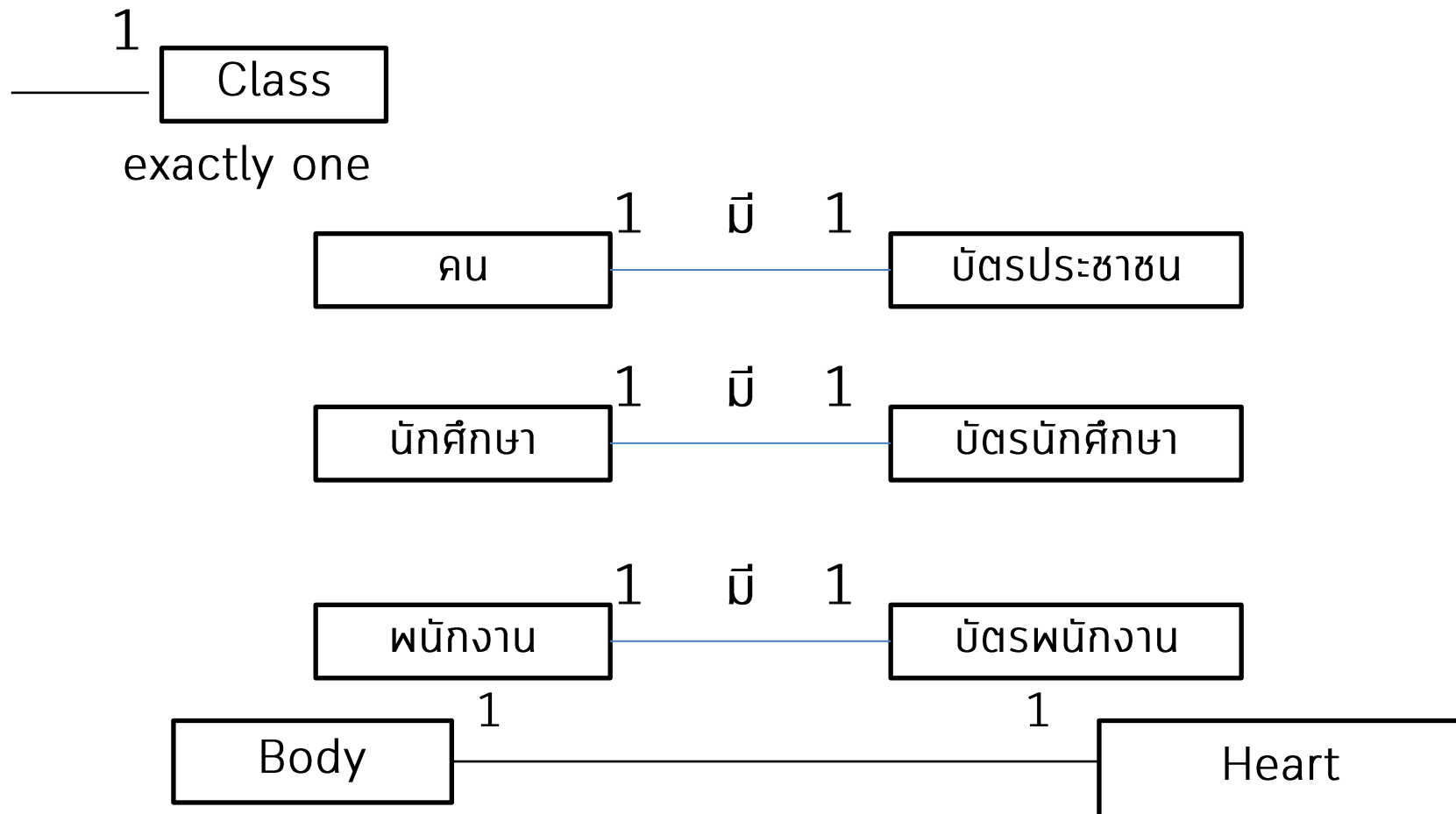
5 ..9

2 ..4

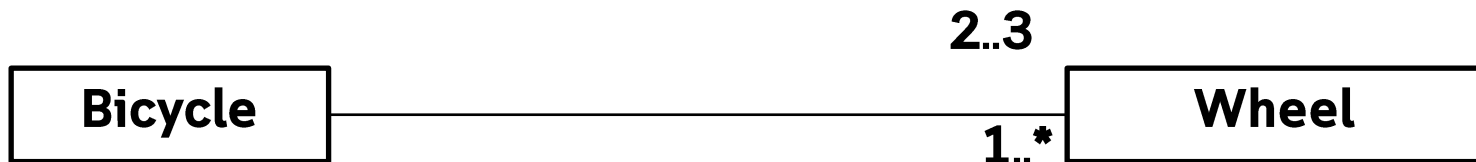
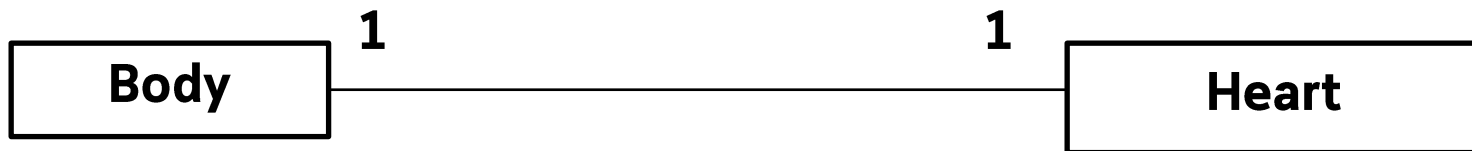
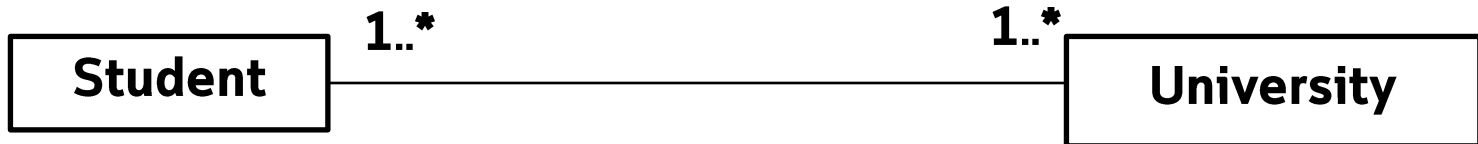


Association Abstraction

Association abstraction : ตัวอย่าง ความสัมพันธ์หนึ่งเท่านั้น



ตัวอย่าง ความสัมพันธ์แบบอื่น ๆ



Associations, Objects & Classes

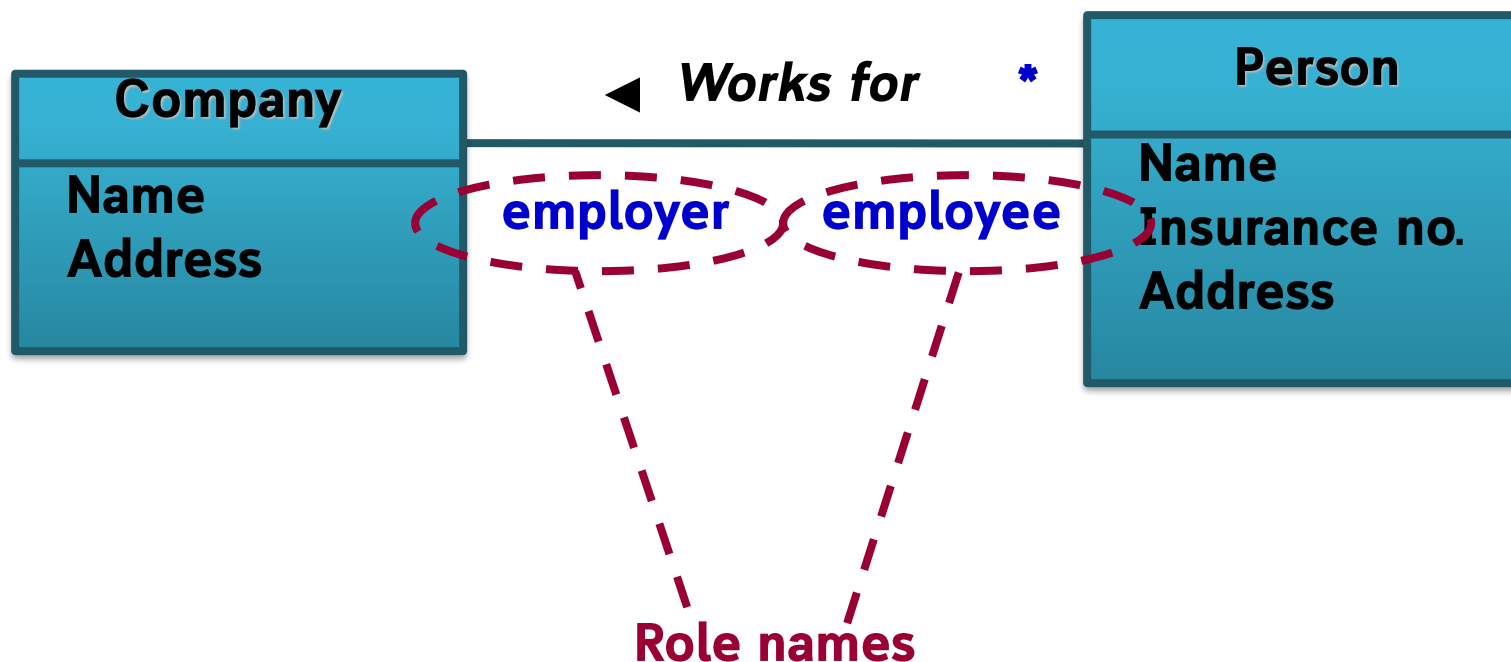
- สำหรับ association ใด ๆ ระหว่าง 2 วัตถุ สามารถมี link ได้ตั้งแต่ 0 ถึง 1 link
- ระหว่าง 2 วัตถุ สามารถมี association ได้มากกว่า 1 association (Multiple Associations) แต่ association เหล่านั้นจะต้องมีความแตกต่างในด้านความหมายของ association
- เช่น คนเป็นเจ้าของรถ และ คนขับรถ คนซื้อรถ
- อาจารย์ สอน นศ. อาจารย์ให้เกรดนศ.
- Multiple associations ระหว่างคลาสกำหนดให้ ตั้งชื่อ associations

Association Abstraction

Association abstraction : **Roles**

- **role**
 - หมายถึง ชื่อที่กำหนดให้กับด้านปลายของ association ซึ่งระบุวิธีการในการที่คลาสมีส่วนร่วมใน association
- ปกติชื่อของ role มักเกิดขึ้นเป็นคู่ (เช่นทั้งสองด้านของ association จะมีชื่อของ role)
- บังคับให้กำหนดชื่อของ Role กับ associations แบบ reflexive (Reflexive associations : ย้อนกลับ)

Association Abstraction

การตั้งชื่อโรว์ (*Roles Names*)

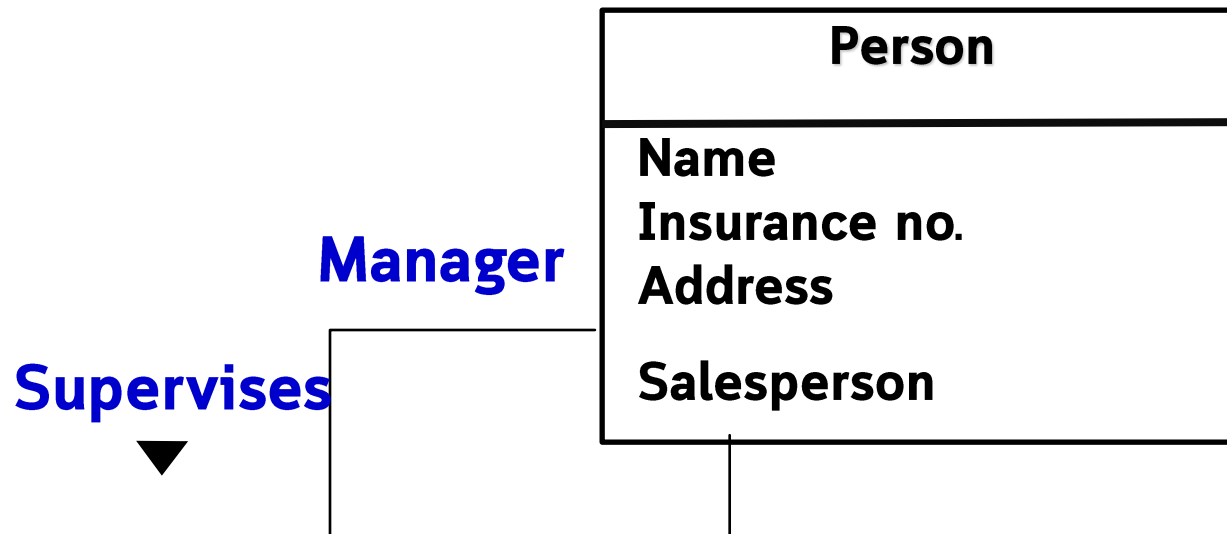
Association Abstraction

Association Names & Roles

- **association อาจมีชื่อได้เช่นเดียวกับชื่อของ role ที่กำหนดให้กับด้านปลายของความสัมพันธ์**
 - **ถ้ามีการตั้งชื่อ association แล้วมักไม่มีการตั้งชื่อ role อีก**
 - **ถ้ามีการตั้งชื่อที่ปลายของ association (ได้แก่ roles) แล้วมักไม่มีการตั้งชื่อ association อีก**

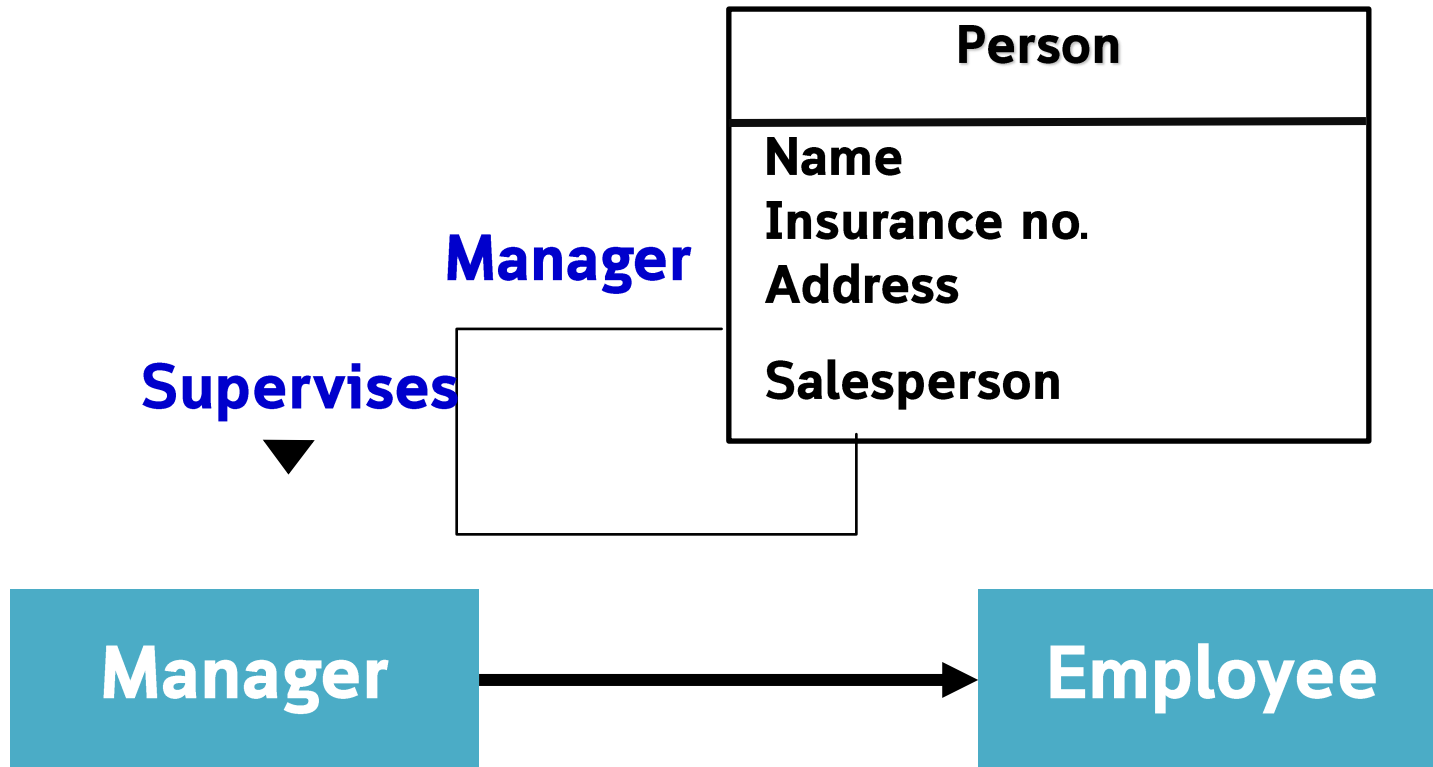
Role names

- บังคับให้กำหนดชื่อของ Role ให้กับที่เกิดขึ้นระหว่าง link ของวัตถุที่สร้างจากคลาสเดียวกัน



Unary Association : คลาสสัมพันธ์กับคลาสตัวเอง

Role names

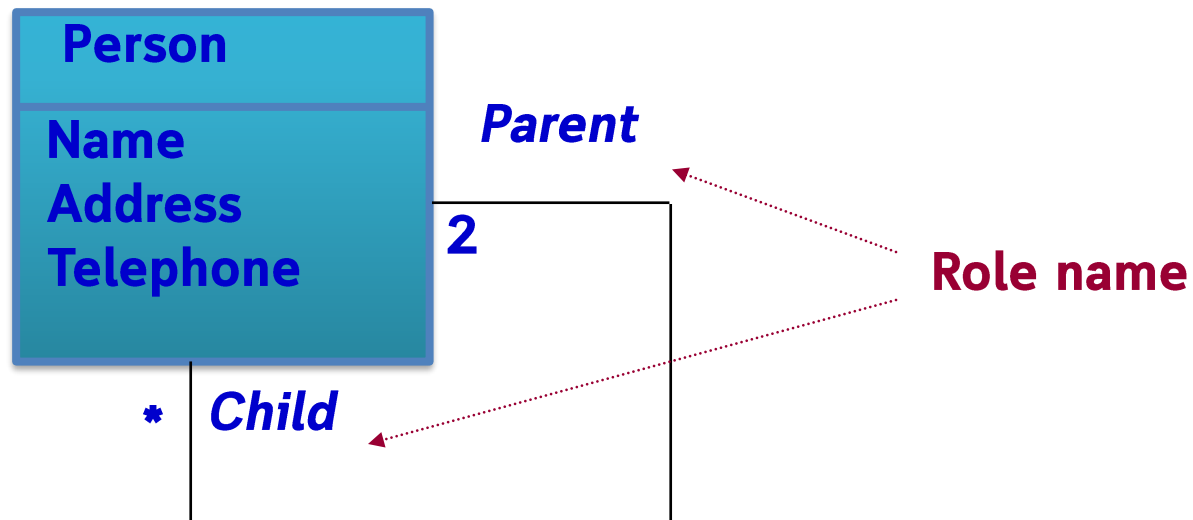


Unary Association : คลาสสัมพันธ์กับคลาสตัวเอง

Binary Association : คลาสสัมพันธ์กับคลาส สองคลาส

A Reflexive Association

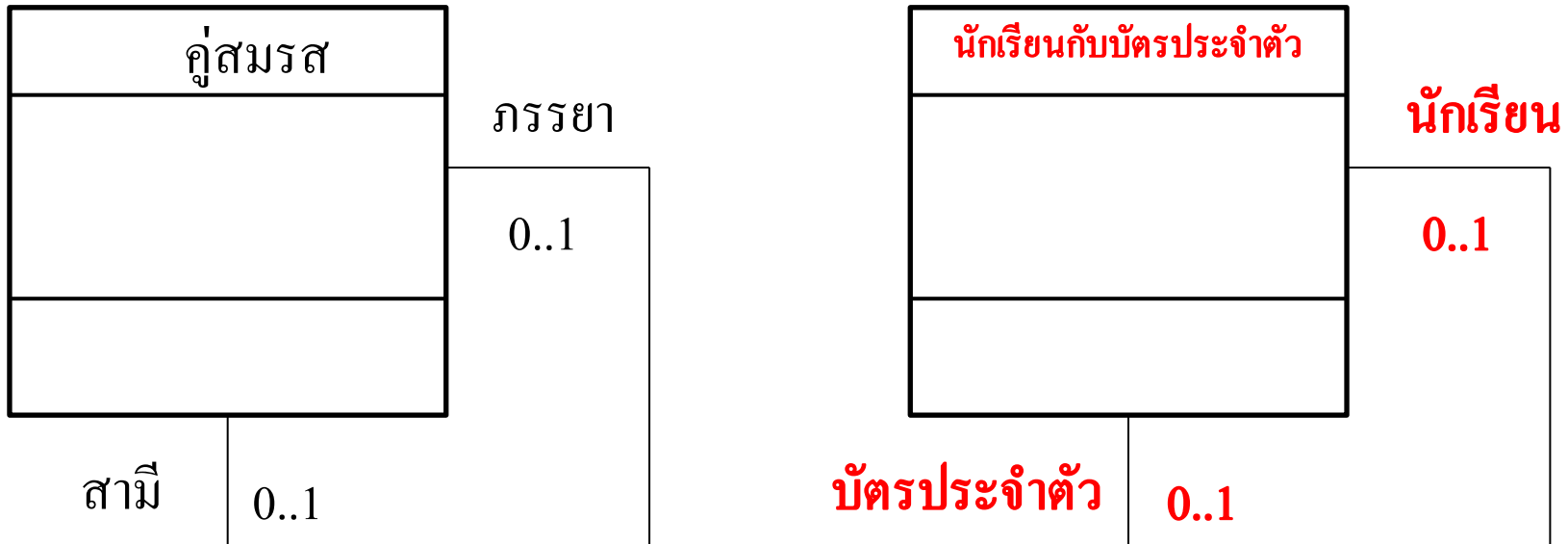
- association ที่สร้างจากคลาสเดียวกัน



**Unary Association : คลาสสัมพันธ์กับ
คลาสตัวเอง**

จยกตัวอย่าง unary association มา 2 ตัวอย่าง พร้อมอธิบาย

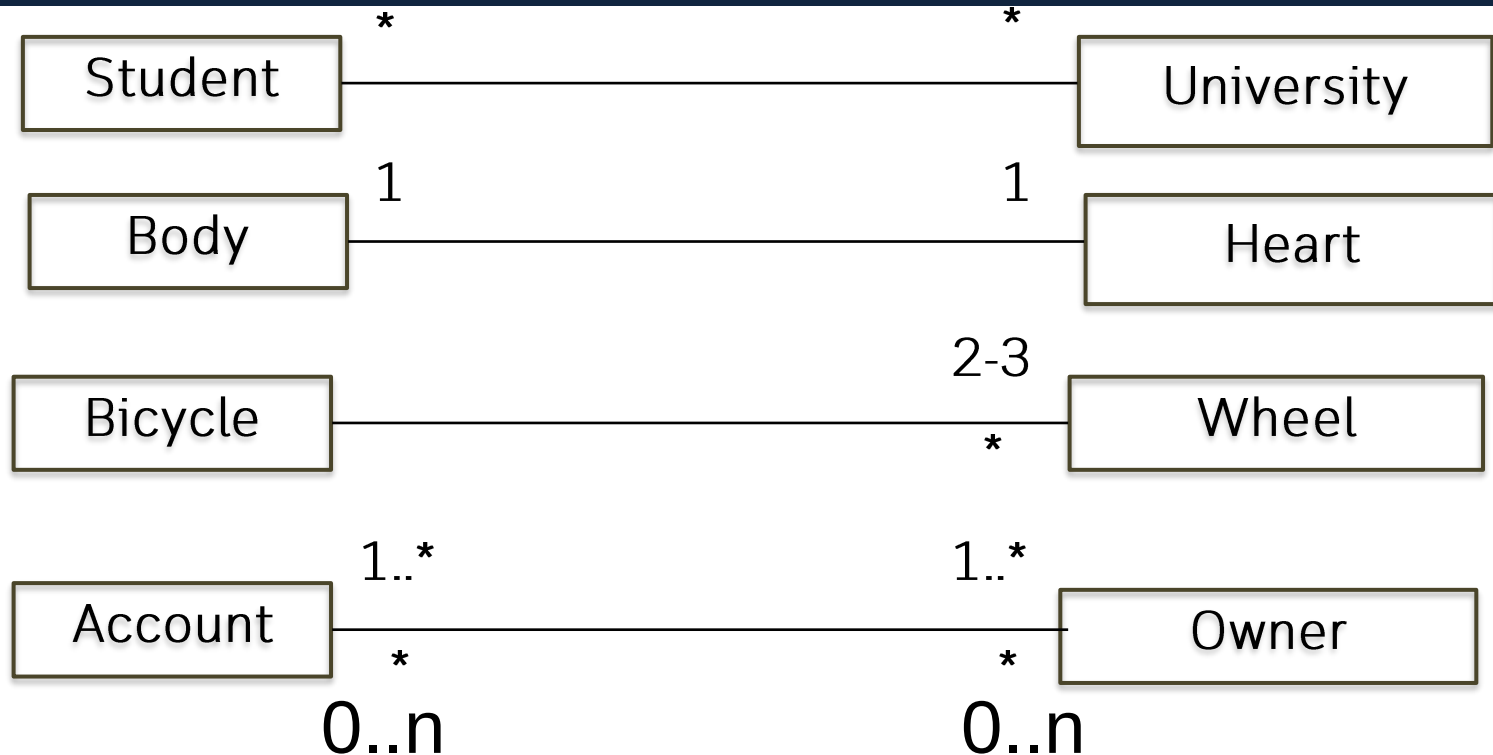
ผิด



นักเรียน — บัตรประจำตัว

บ้าน — คน

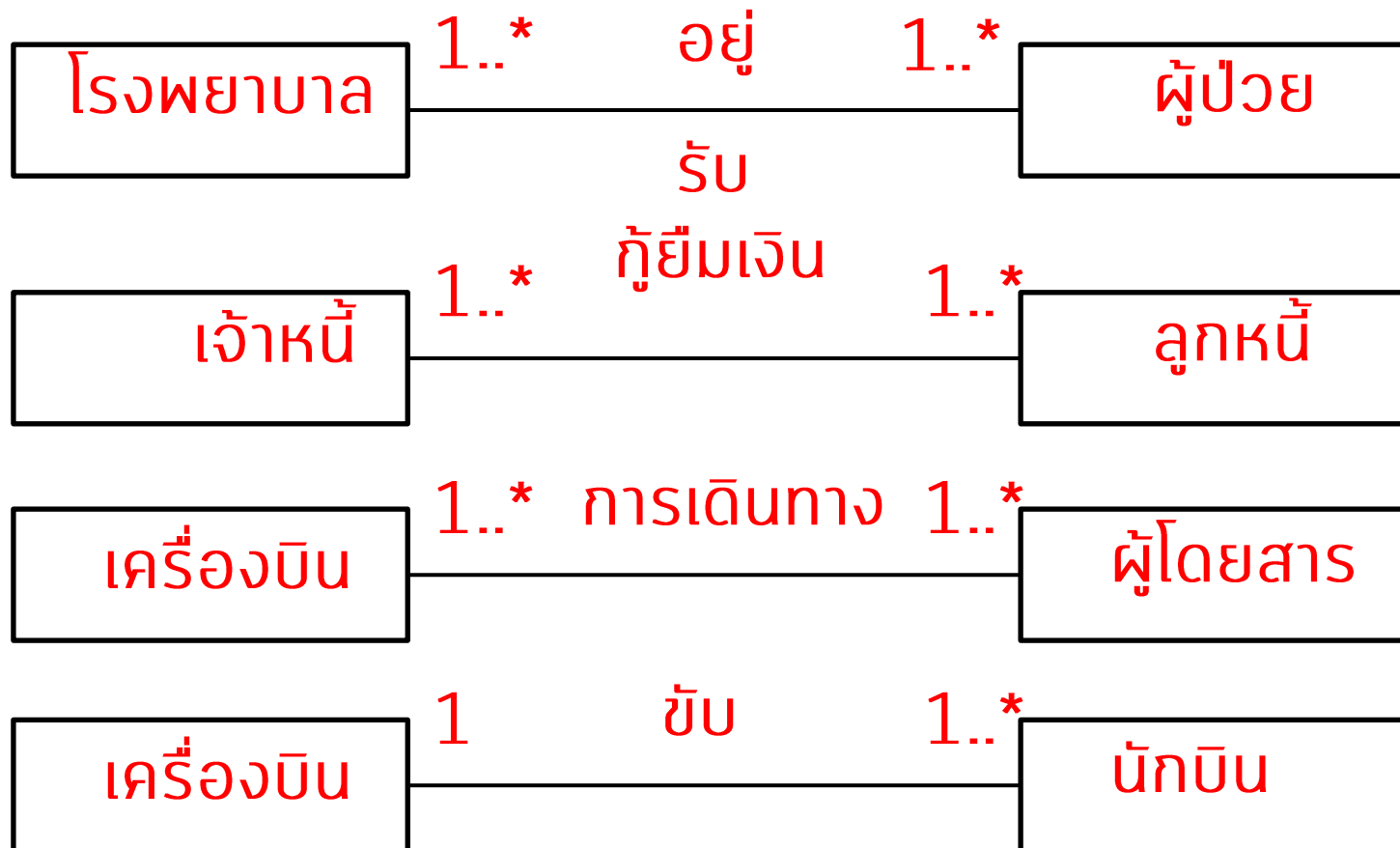
Association Abstraction

Example of Multiplicities

Binary Association : คลาสสัมพันธ์ระหว่าง 2 คลาส

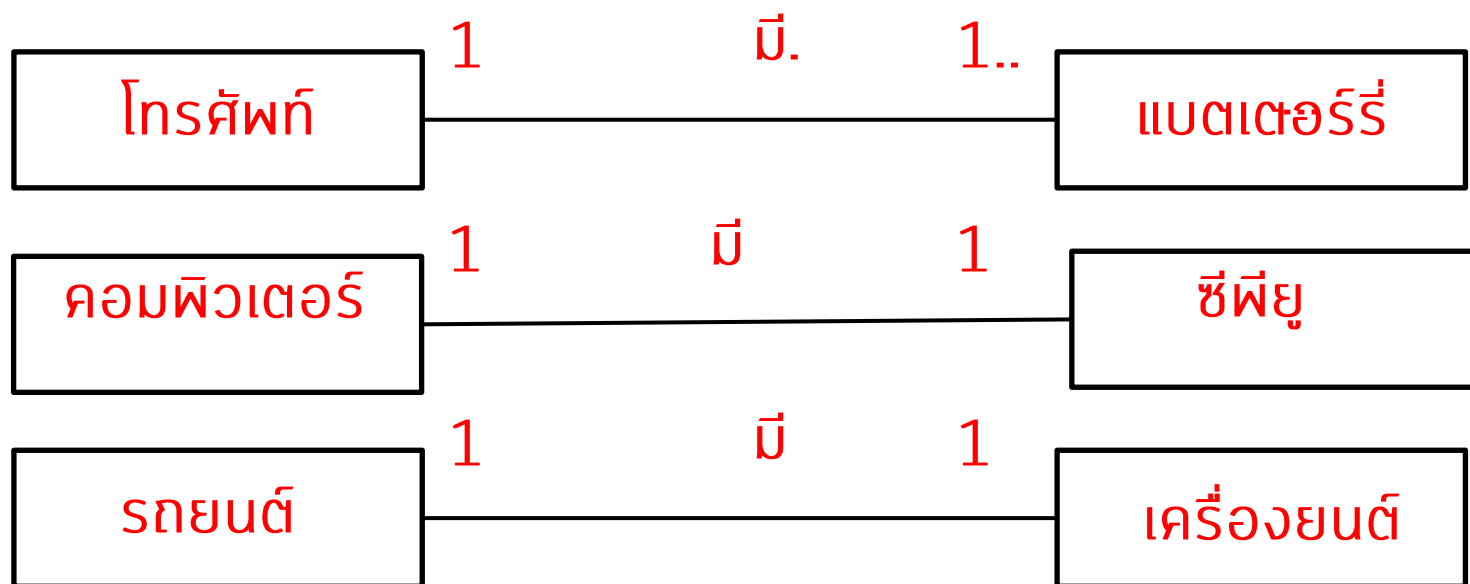
Association Abstraction

จงยกตัวอย่าง *Binary Association* มา 3 ตัวอย่าง



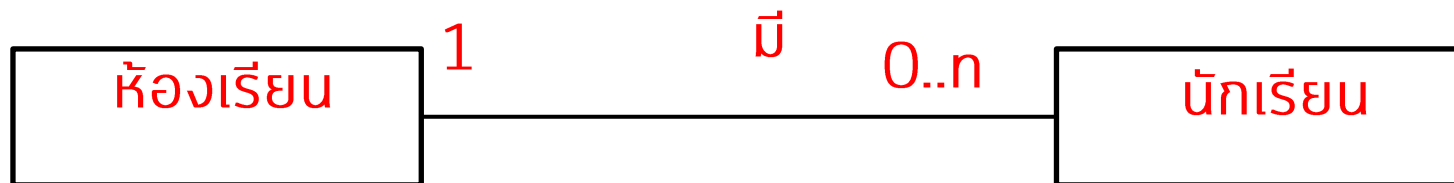
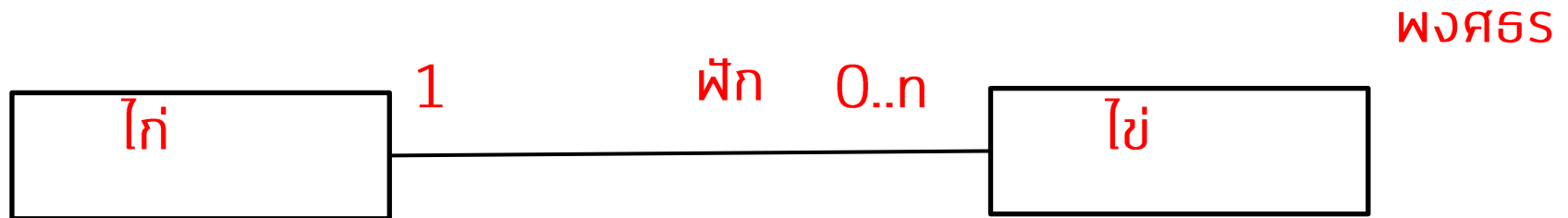
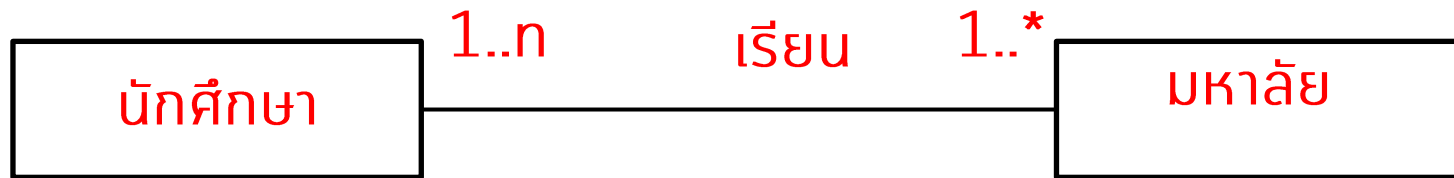
Association Abstraction

จยกตัวอย่าง Binary Association มา 3 ตัวอย่าง



Association Abstraction

จยกตัวอย่าง Binary Association มา 3 ตัวอย่าง



กนกวรรณ

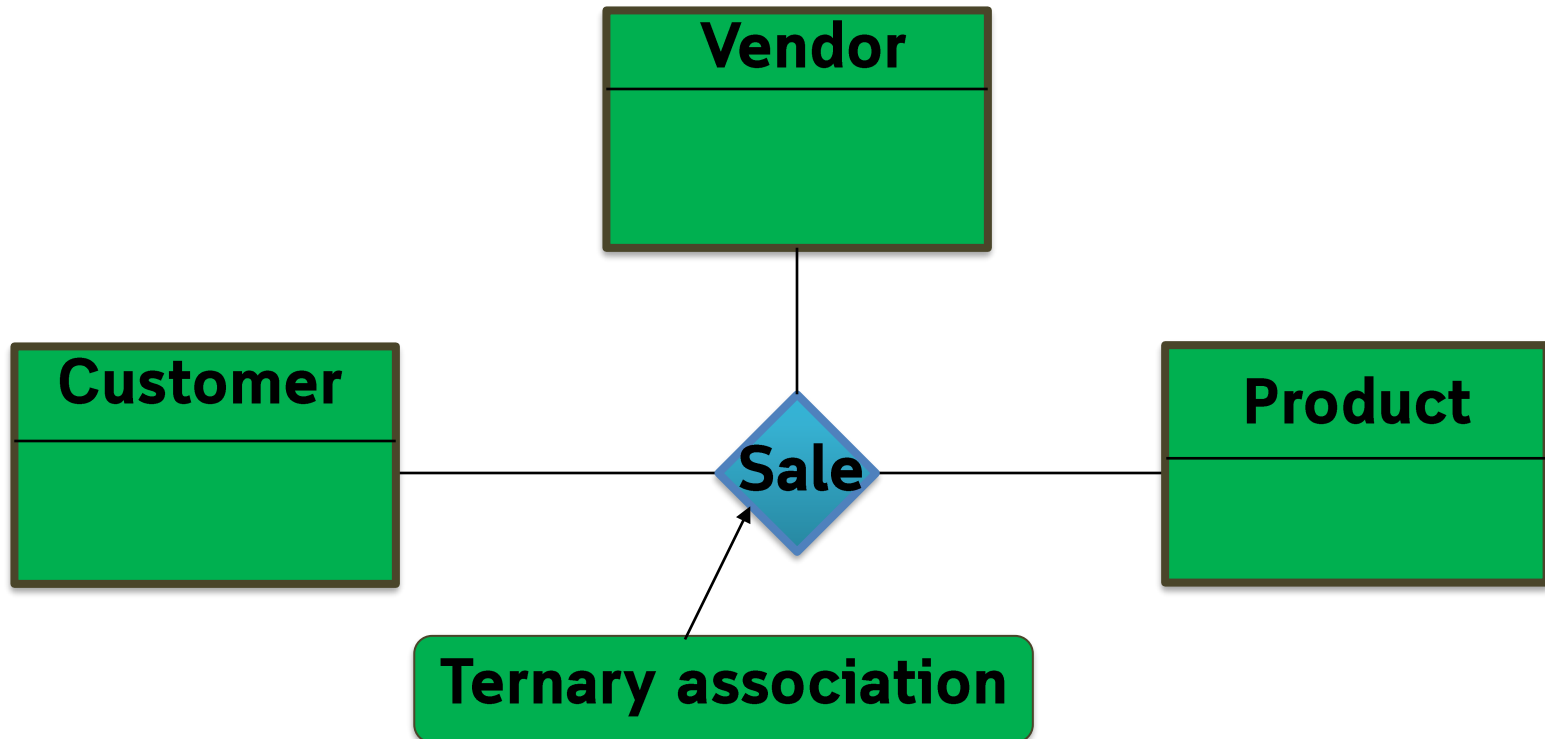
Association Abstraction

Ternary Associations

- **A ternary association**
 - เป็น association ที่ประกอบด้วย 3 คลาส
- เช่นเดียวกับแนวคิดของ ternary relationship ใน ER modelling และแนวคิดคล้าย ๆ กัน
- นำเสนอโดยใช้รูปสี่เหลี่ยมขนมเปียกปูน เชื่อมโยงกับคลาสที่เกี่ยวข้อง

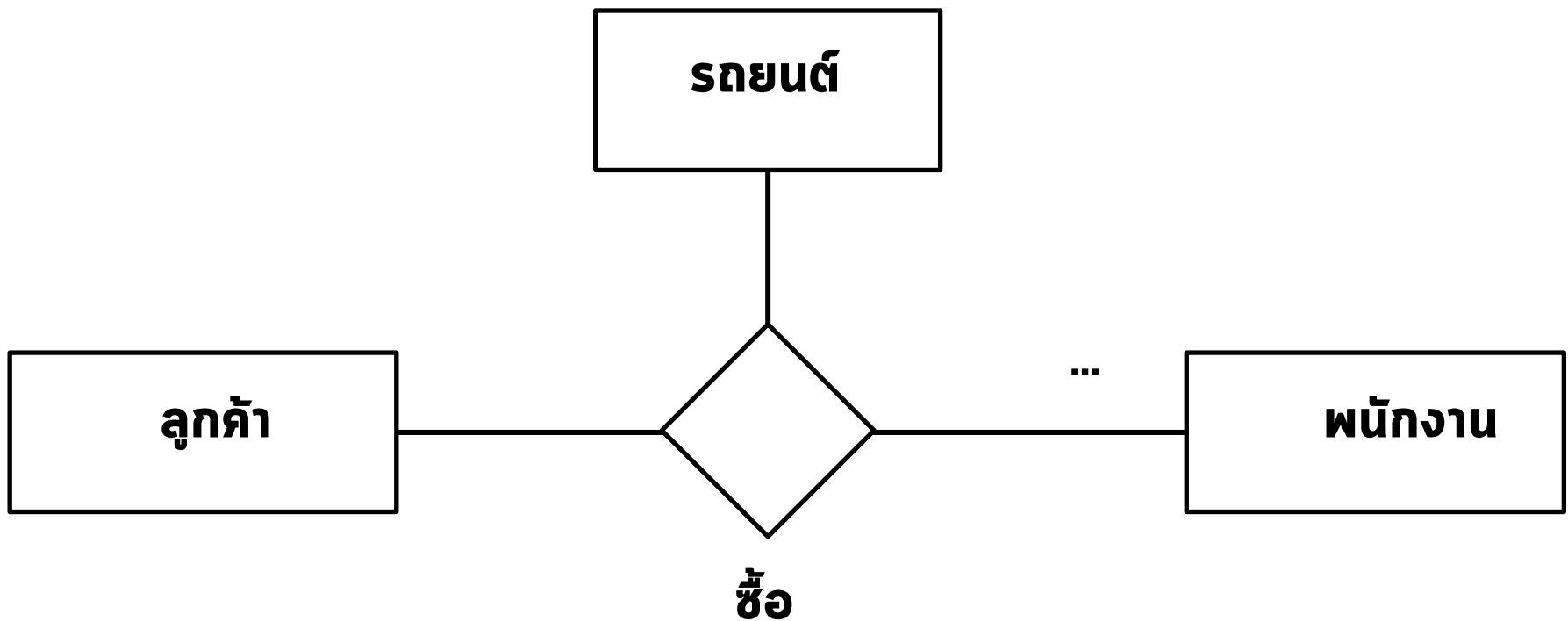
Association Abstraction

Ternary Associations

**Ternary Association : คลาสสัมพันธ์ของ 3 คลาส**

Association Abstraction

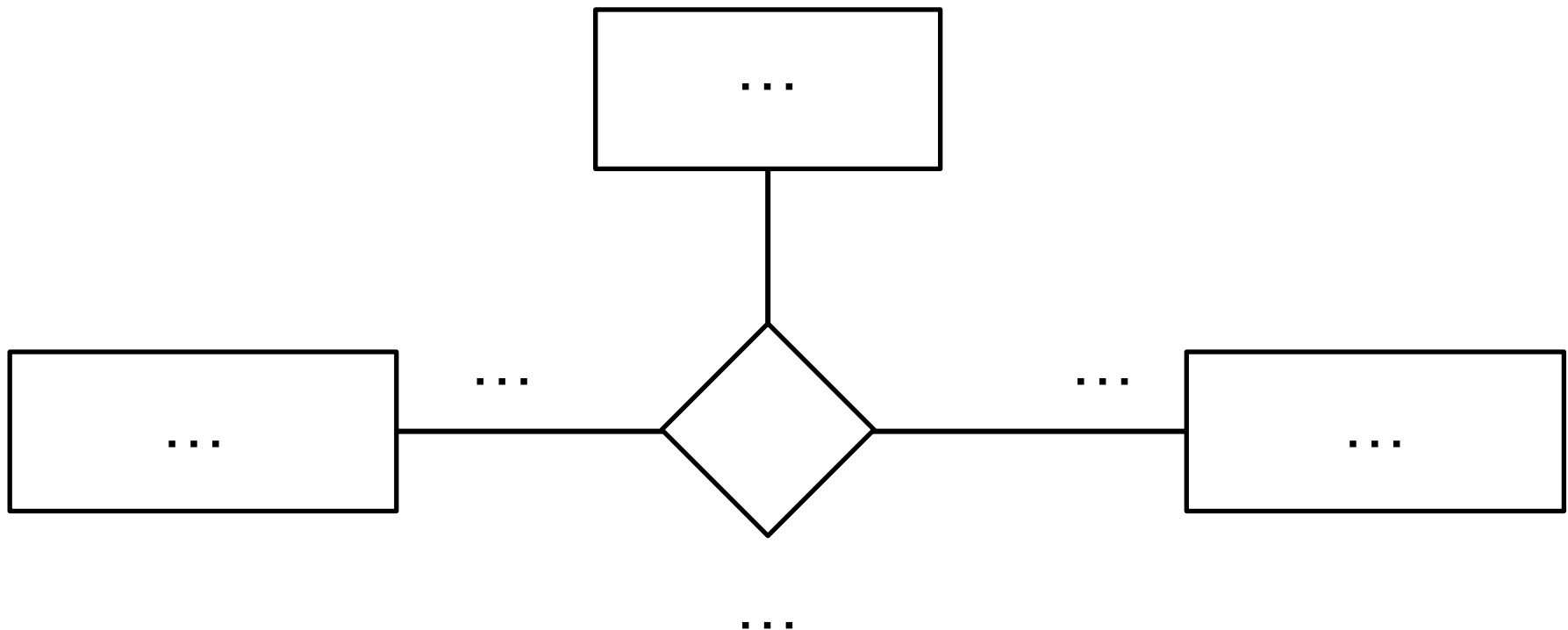
จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย



Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

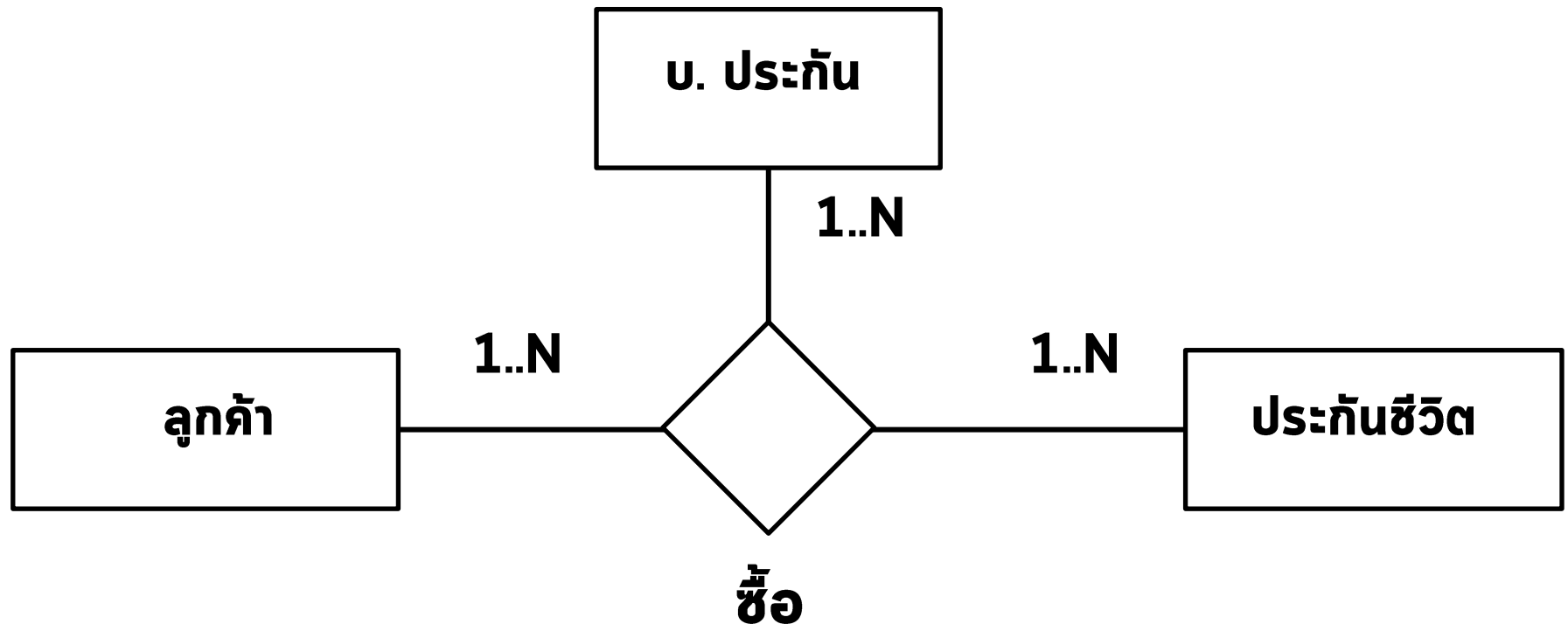
จยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย



Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย

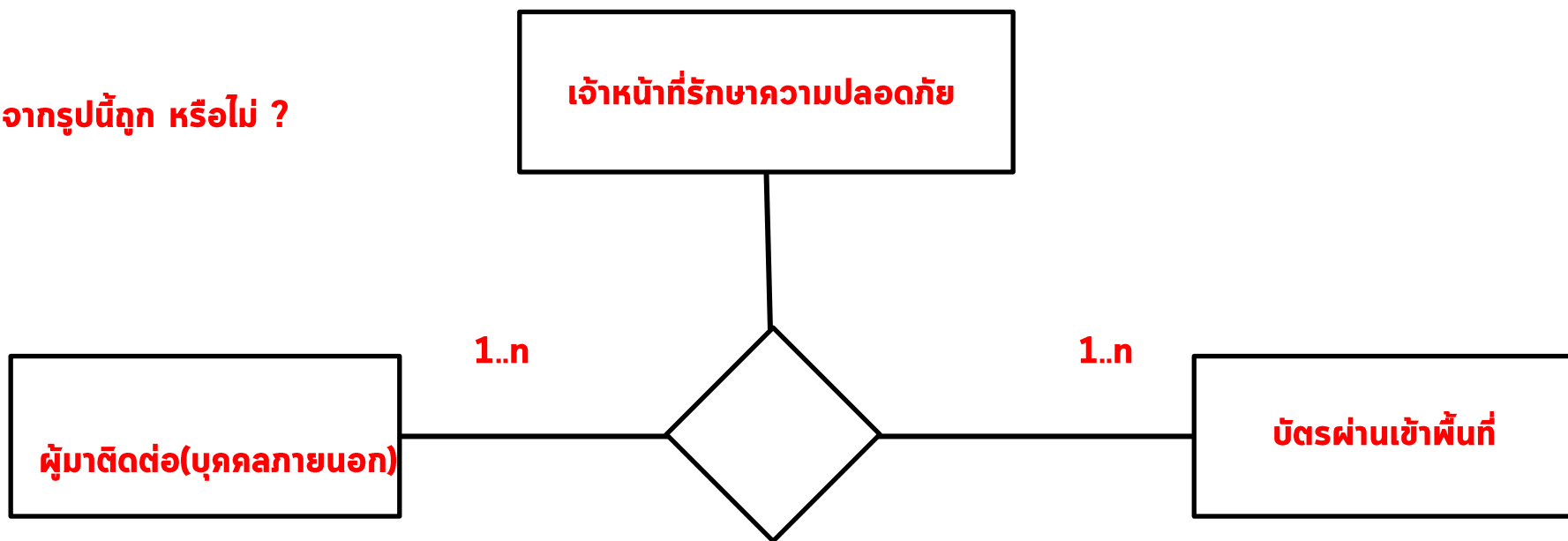


Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย

จากรูปนี้ถูก หรือไม่ ?



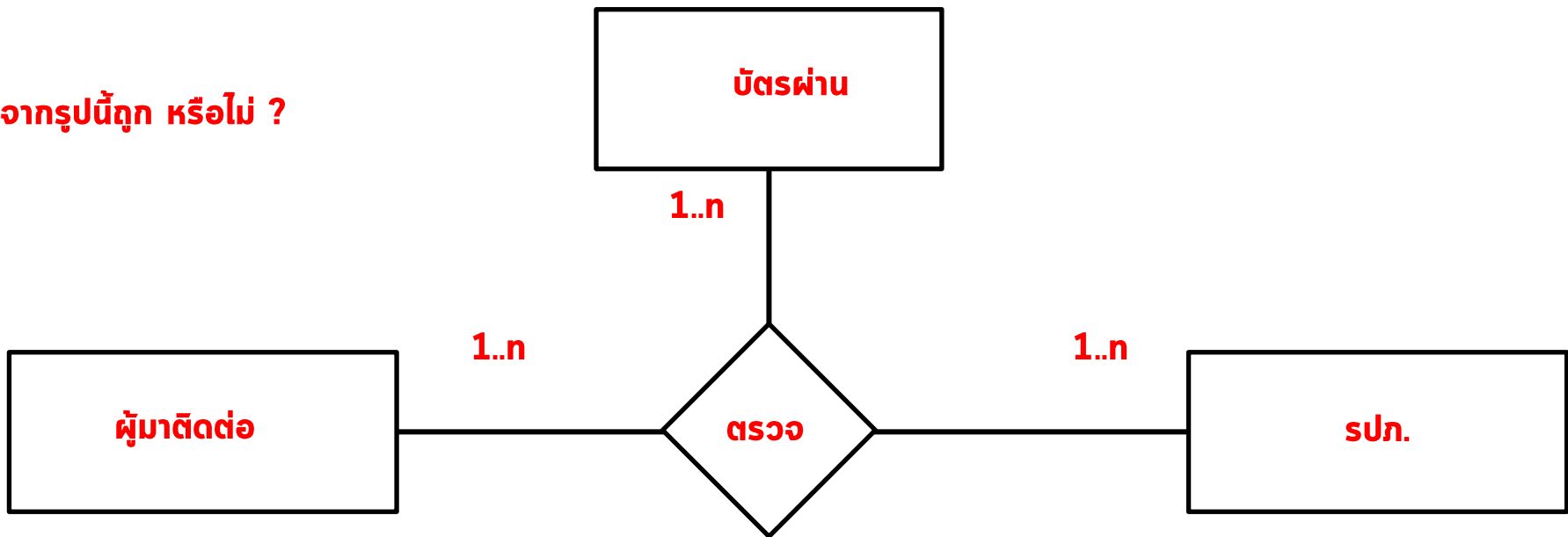
การขออนุญาตเข้าสถานที่ราชการหรือส่วนตัว

Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย

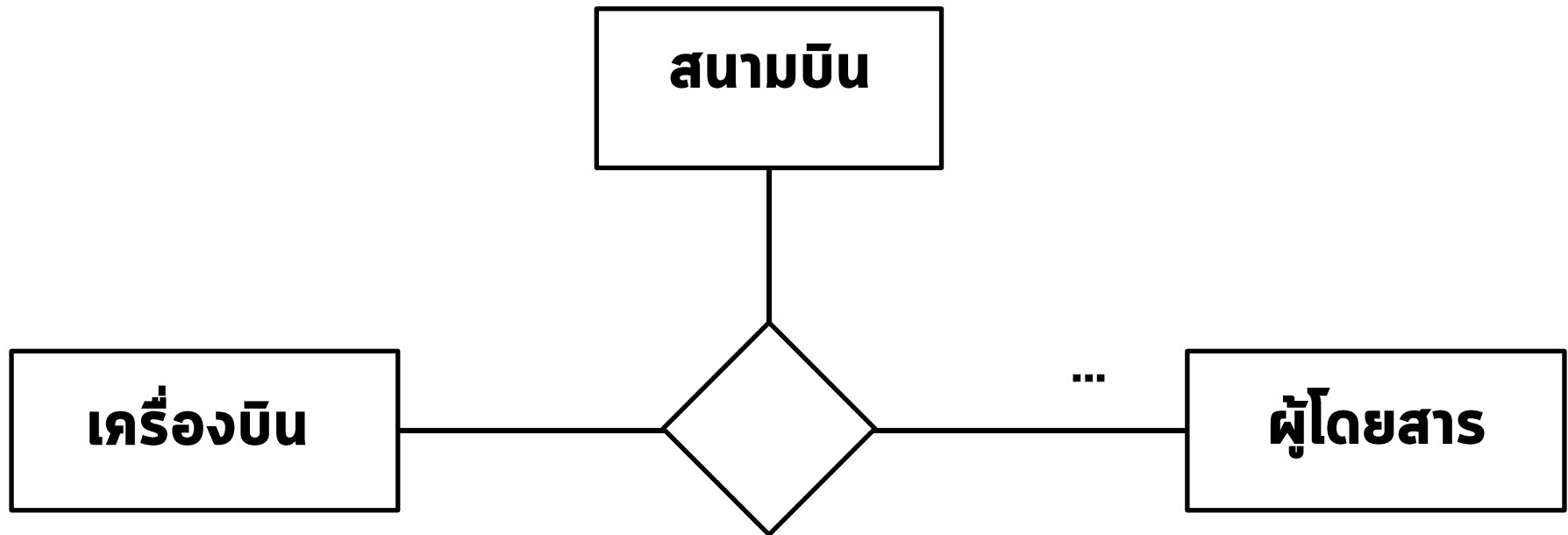
จากรูปนี้ถูก หรือไม่ ?



Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

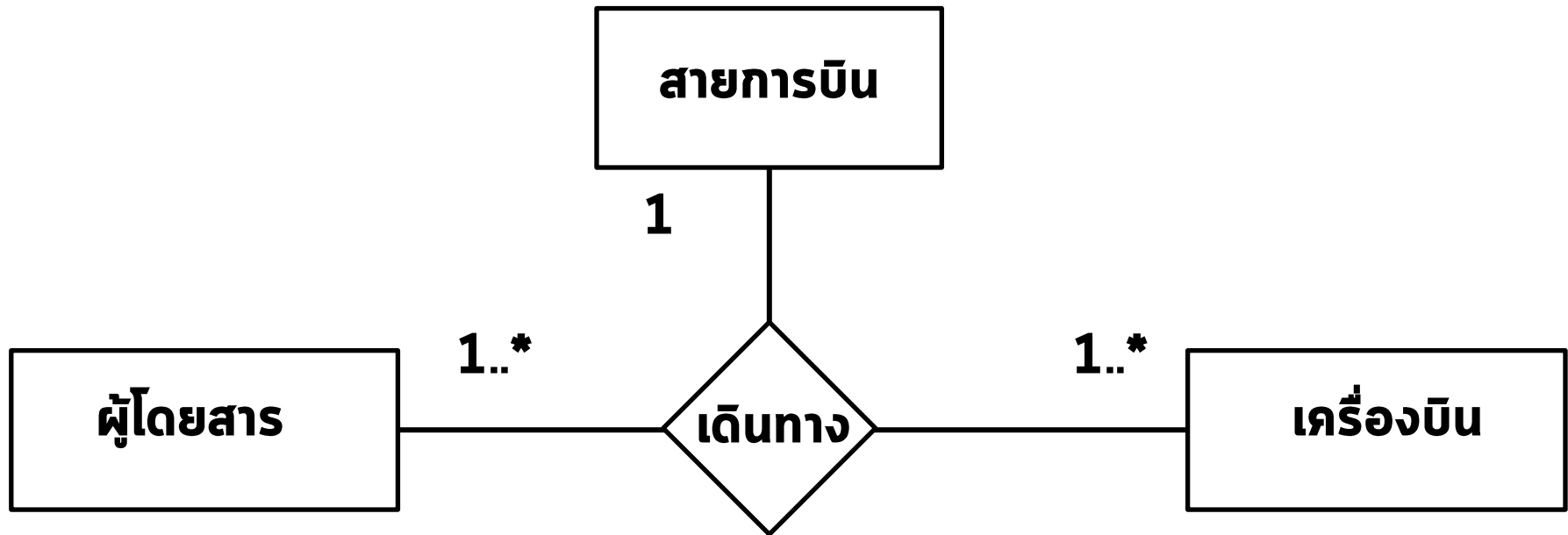
จยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย



แผนภาพคลาสนี้แสดงถึงความสัมพันธ์แบบ ternary ระหว่างคลาสสามคลาส ได้แก่ สนามบิน เครื่องบิน และ ผู้โดยสาร ความสัมพันธ์แบบ ternary เป็นความสัมพันธ์ระหว่างสามคลาส โดยคลาสหนึ่งสามารถมีความสัมพันธ์กับคลาสอื่นสองคลาสได้ ความสัมพันธ์แบบ ternary มีหลายประเภท ในที่นี้ ความสัมพันธ์แบบ ternary ที่ใช้คือ association

Association Abstraction

จยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย

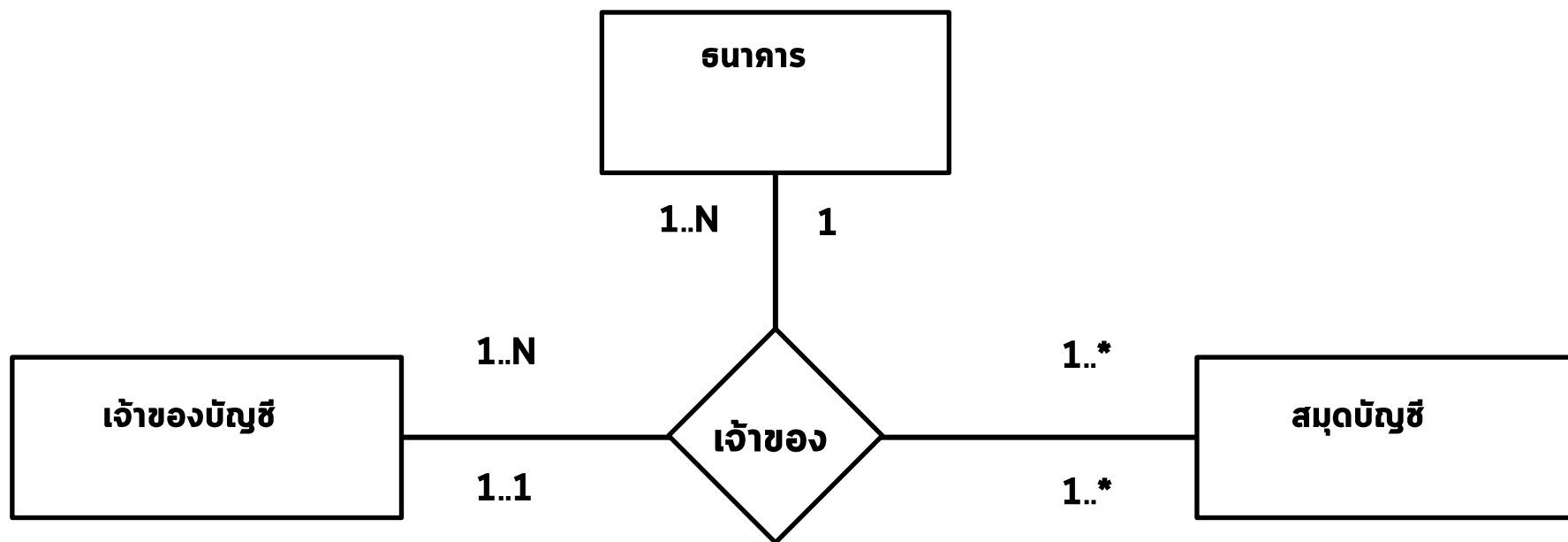


สรุปแล้ว แผนภาพคลาสนี้แสดงถึงความสัมพันธ์ระหว่างคลาสสามคลาส ได้แก่ สายการบิน ผู้โดยสาร และ เครื่องบิน ความสัมพันธ์ระหว่างคลาสเหล่านี้สามารถอธิบายได้ดังนี้

- สายการบินสามารถมีเครื่องบินหลายลำ
- เครื่องบินสามารถมีผู้โดยสารหลายคน
- ผู้โดยสารสามารถขึ้นเครื่องบินที่สายการบินใดสายการบินหนึ่งได้
- ผู้โดยสารจะต้องมีตัวโดยสารอย่างน้อยหนึ่งใบ
- ตัวโดยสารสามารถออกให้แก่ผู้โดยสารได้หลายราย

Association Abstraction

จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย



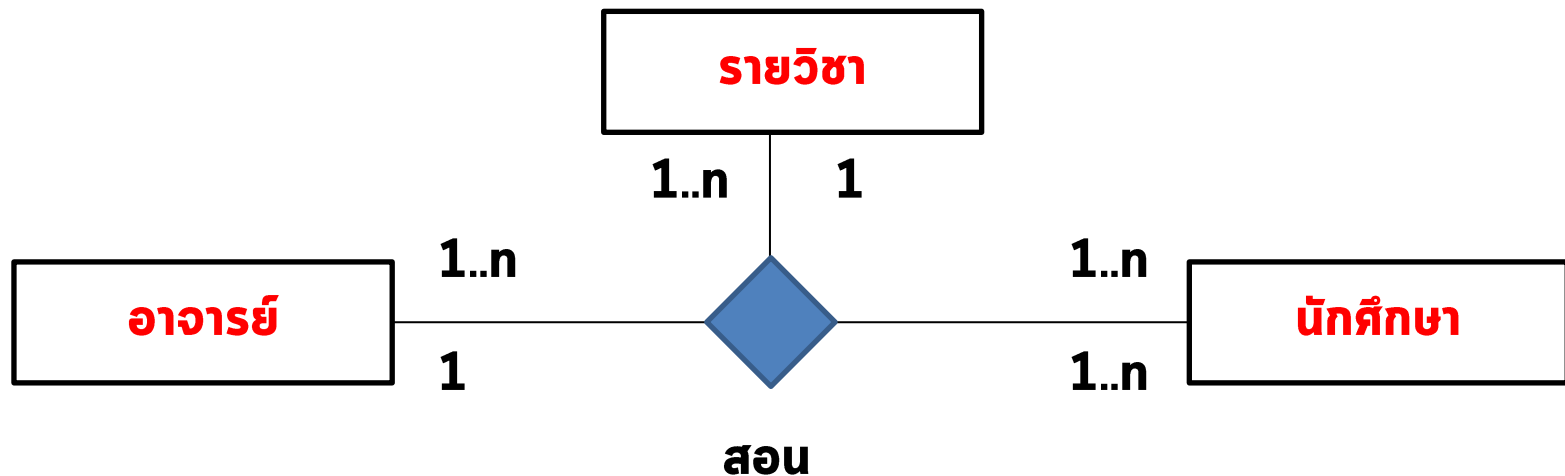
ในกรณีของแผนภาพคลาสนี้ ความสัมพันธ์แบบ association แสดงด้วยเส้นเชื่อมระหว่างคลาสธนาคารกับคลาสเจ้าของบัญชี และระหว่างคลาสเจ้าของบัญชีกับคลาสสมุดบัญชี เส้นเชื่อมเหล่านี้ไม่มีลูกศรชี้ไปทางคลาสใดคลาสหนึ่ง แสดงว่าความสัมพันธ์เป็นแบบ non-directional ความสัมพันธ์แบบ association นี้สามารถอธิบายได้ดังนี้

ธนาคารสามารถมีเจ้าของบัญชีหลายราย
 เจ้าของบัญชีสามารถมีสมุดบัญชีหลายเล่ม
 สมุดบัญชีสามารถเป็นของเจ้าของบัญชีได้หนึ่งราย

Association Abstraction

อาจารย์นัฐพงศ์ สอนวิชา ooad ให้ นักศึกษาวิทยาลัยคอมรูน 44

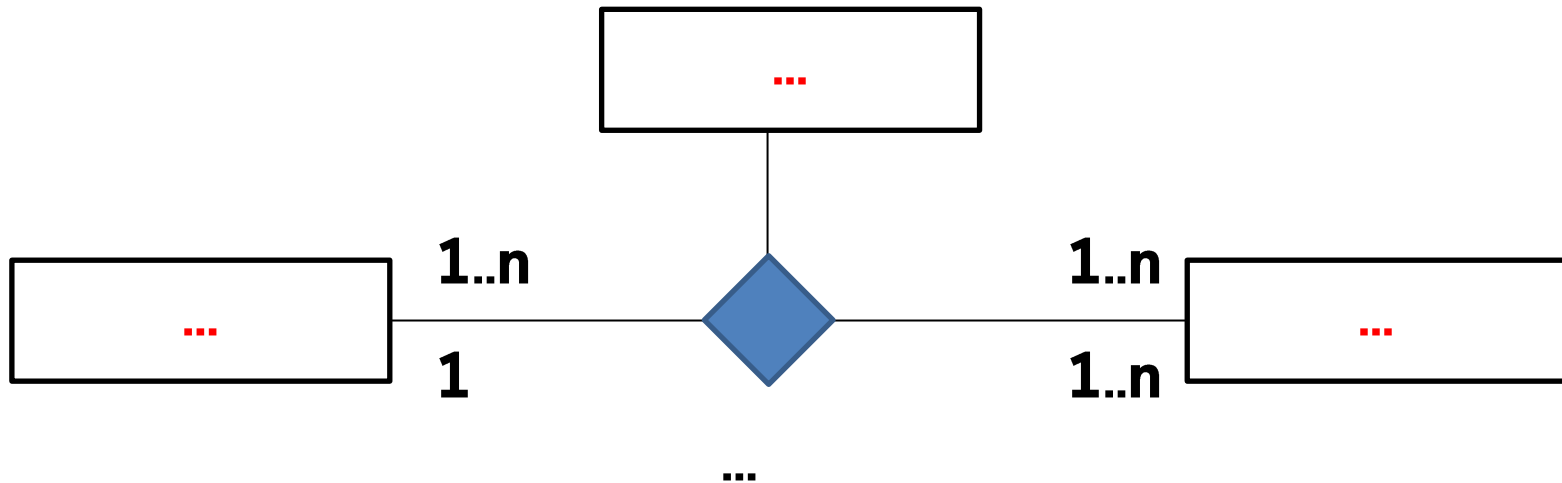
วิเคราะห์ นัฐพงศ์ เป็นวัตถุของคลาส อาจารย์
 ooad เป็นวัตถุของคลาส รายวิชา
 วิทยาลัยคอมรูน 44 เป็นวัตถุของคลาส นักศึกษา
 สรุป มี 3 คลาสที่เกี่ยวข้องกัน



Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

จงยกตัวอย่างของ Ternary Association มา 2 ตัวอย่าง พร้อมทั้งอธิบาย

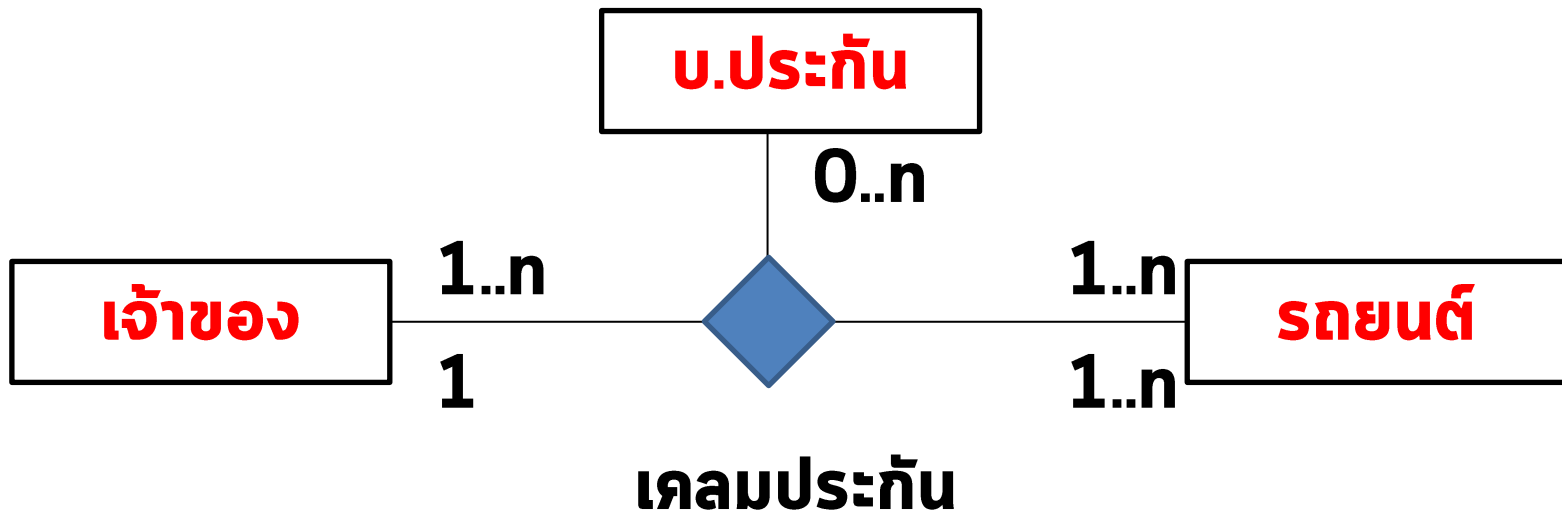


จงยกตัวอย่างความสัมพันธ์แบบ Ternary พร้อมอธิบาย

Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

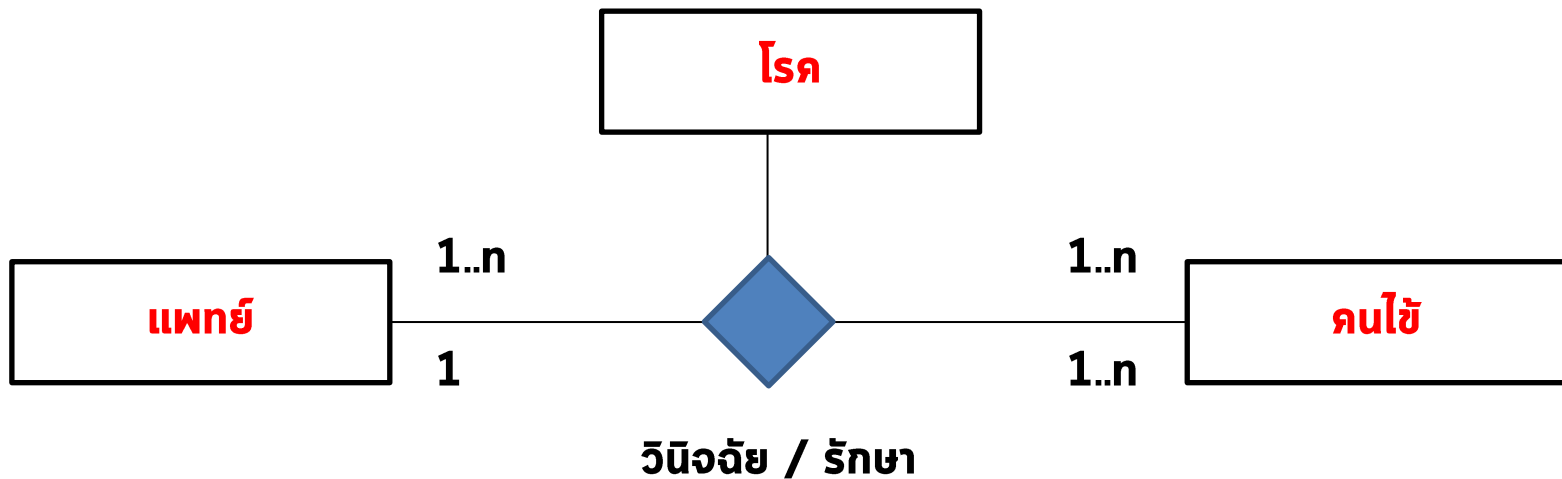
จงยกตัวอย่างความสัมพันธ์แบบ Ternary พร้อมอธิบาย

จยกตัวอย่างความสัมพันธ์แบบ Ternary พร้อมอธิบาย



Ternary Association : คลาสสัมพันธ์ของ 3 คลาส

Association Abstraction

*Ternary Association Multiplicity***จงยกตัวอย่างความสัมพันธ์แบบ Ternary พร้อมอธิบาย****Ternary Association : คลาสสัมพันธ์ของ 3 คลาส**

Association Abstraction

Ternary Association Multiplicity

- multiplicity ของ ternary associations ซับซ้อนกว่า binary และ unary associations มักเป็น แบบ “many” สำหรับคลาสที่เกี่ยวข้อง participating

Association Abstraction

N-ary Associations

- โดยทั่วไปแล้ว **association** อาจเชื่อมโยงกับคลาสจำนวนเท่าใดก็ได้
- ถ้าเชื่อมโยงคลาสที่มากกว่า 2 คลาสขึ้นไป (รวมทั้ง ternary) เรียกว่า “**n-ary**” associations
- แต่ **Associations** ที่เชื่อมโยงมากกว่า 2 คลาสมักจะหาได้ยาก

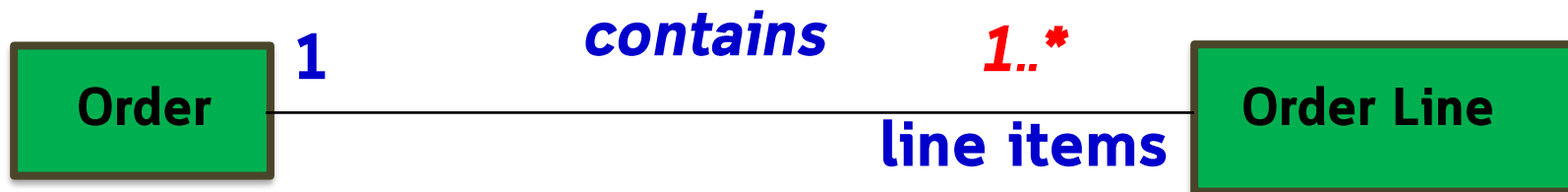
Association Abstraction

Summary: Basic notation for associations



• ตัวอย่างเช่น

***** ในใบสั่งซื้อจำเป็นต้องมี อย่างน้อย 1 รายการ**



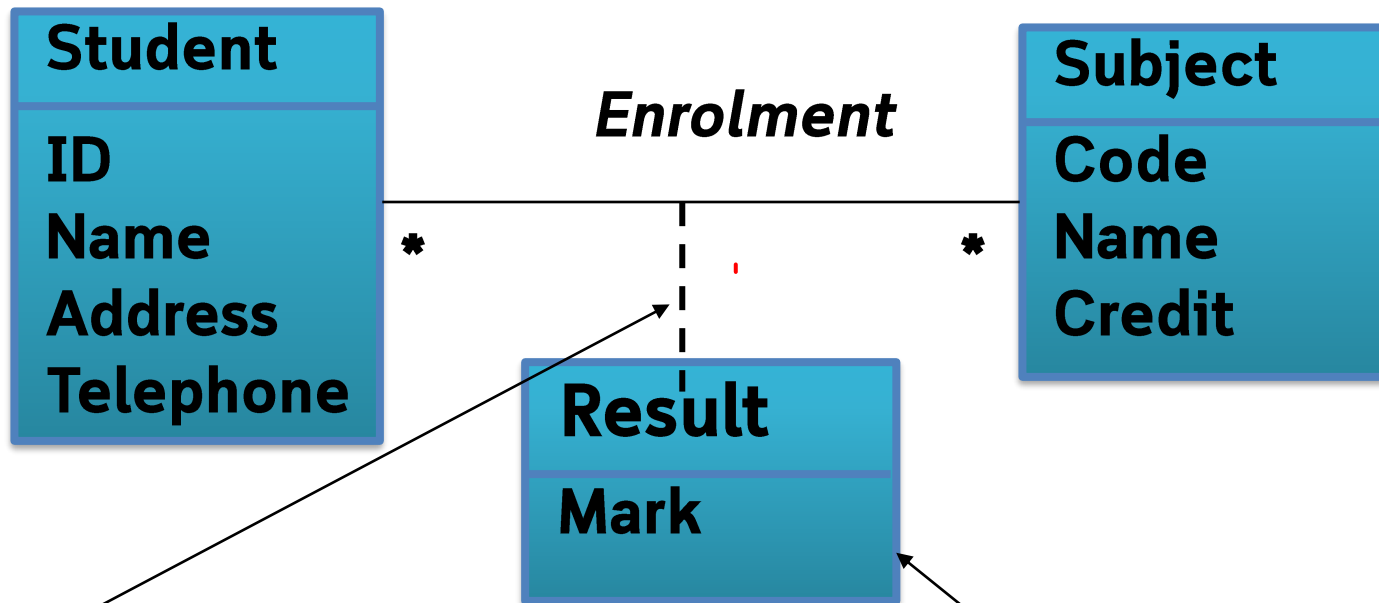
Association Abstraction

Association Classes

- **Link จะถูกพิจารณา เช่นเดียวกับการที่วัตถุถูกพิจารณาจาก attribute ของวัตถุนั้น**
- **attribute ควรผูกติดอยู่กับ association โดยใช้ association class มากกว่าที่จะเป็นของวัตถุที่เชื่อมโยง เมื่อใดก็ตามที่ whenever it exists or has meaning when the link itself exists (rather than just the objects linked)**
- **มักเกิดขึ้นบ่อยกับ many-to-many associations**

Association Classes

***** ในบางครั้งคลาส จะเกิดทางอ้อม หรือเกิดขึ้นภายหลัง จากความสัมพันธ์ของคลาสอื่นๆ เราเรียกคลาสที่เกิดขึ้นแบบนี้ ว่า *association classes***



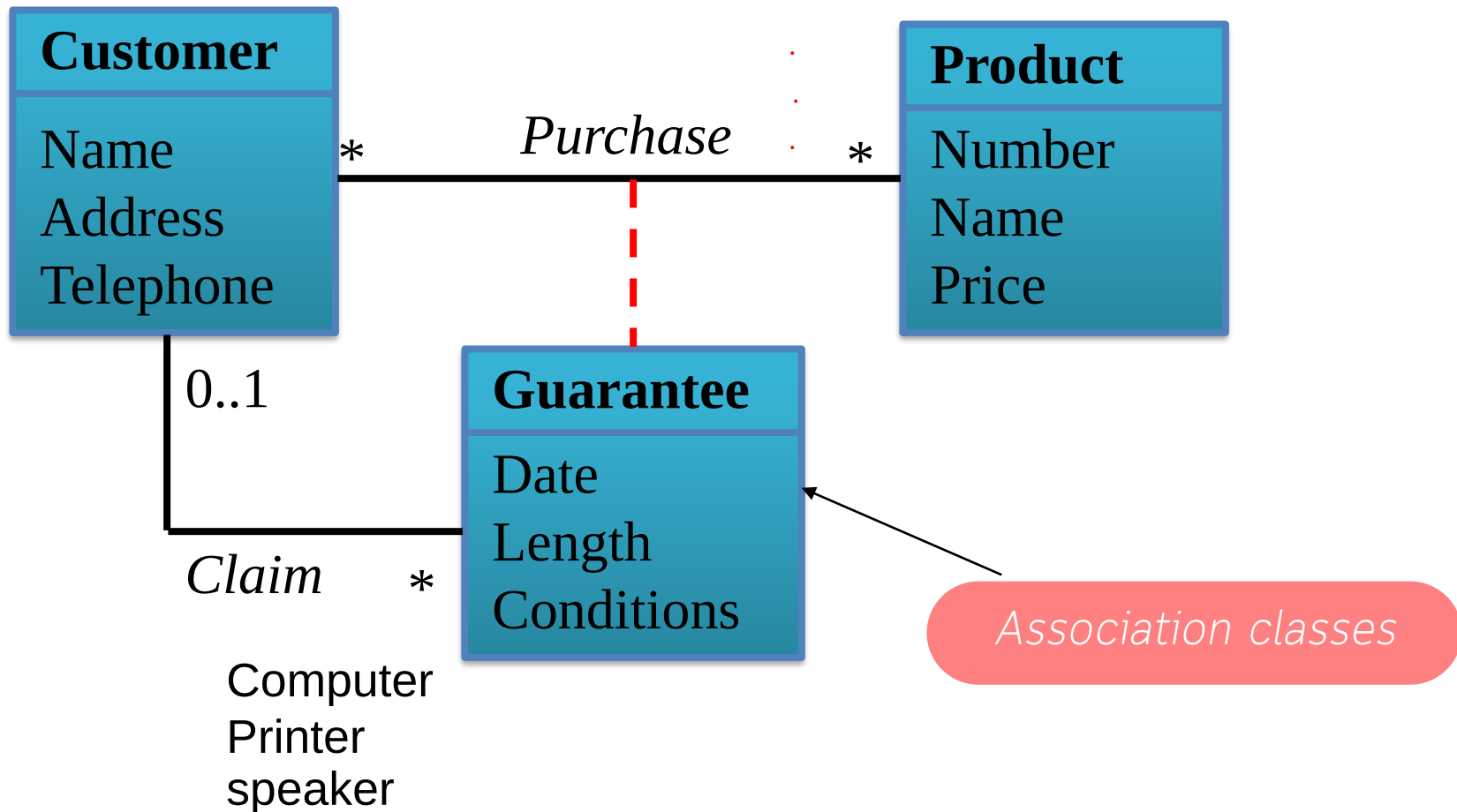
Dependency relationship

Association classes

***** คลาสผลการเรียน result เกิดจากความสัมพันธ์ระหว่าง คลาส นักเรียน และ วิชา ที่เป็นแบบ many to many**

Association Classes

- association class สามารถมีส่วนร่วมใน associations อื่น ๆ เช่นเดียวกับที่คลาสปกติสามารถทำได้



Customer

CusID	CusName	CustAddr

*



*

Guarantee

ProID	ProName	ProPrice

Product

ProID	ProName	ProPrice

Customer

CusID	CusName	CustAddr
C001	somchai	BKK
C003	somying	BKK
C003	somsee	BKK

*

Order

OrderID	OrdDate	CustID
Or001	23/12/2556	C001
Or002	24/12/2556	C001
Or003	25/12/2556	C001
Or004	25/12/2556	C002

OrderLine

*

Product

ProID	ProName	ProPrice
p0001	Computer	35000
p0002	Printer	5000
p0003	Speaker	1000
P0004	Mouse	500

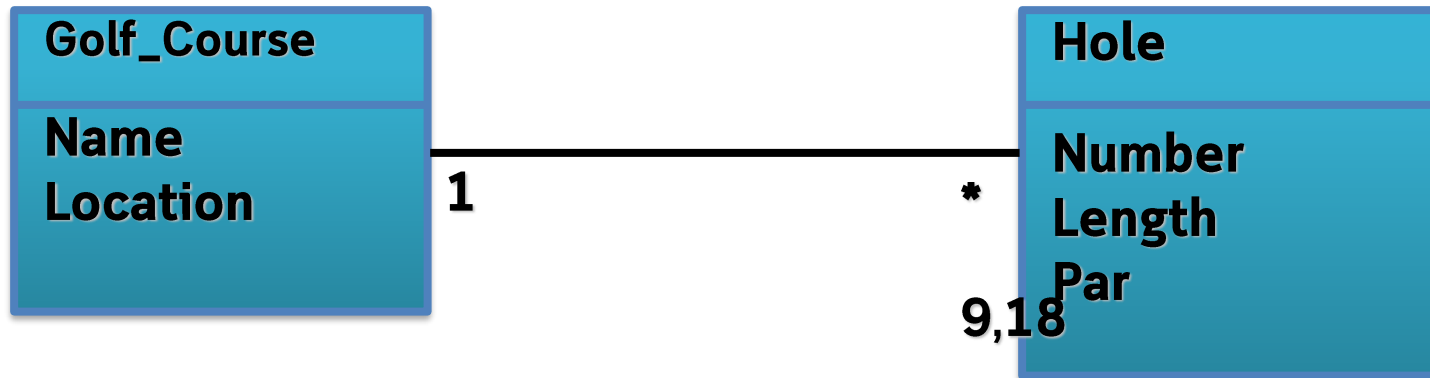
LineID	ProID	ProQTY	OrdID
D0001	p0001	5	Or001
D0002	p0002	5	Or001
D0003	p0003	5	Or001
D0004	p0001	5	Or002
D0005	p0002	5	Or002

Qualified Associations

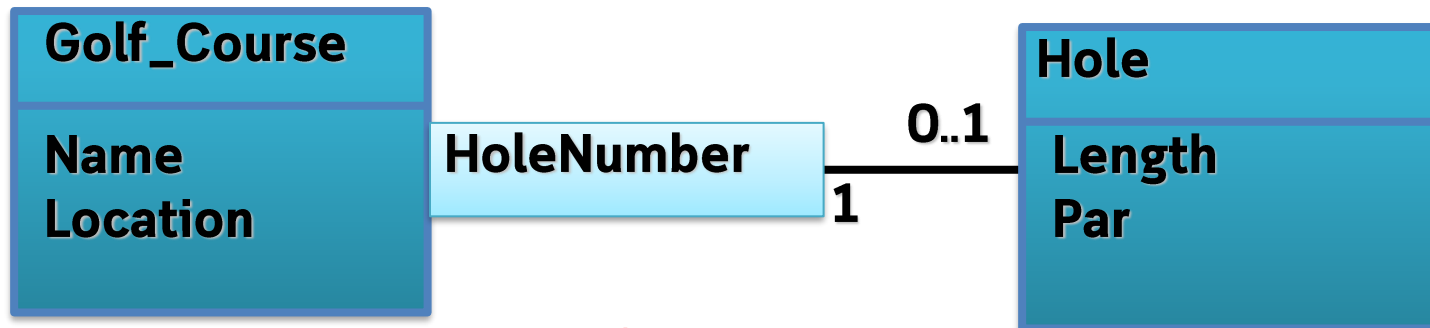
- **qualified association**
 - หมายถึง association ที่ถูกทำให้ชัดเจนโดย attribute ที่เรียกว่า qualifier
 - พิจารณา **qualified association** เช่นเดียวกับแนวคิดของ weak entity type ใน ER

Qualified Associations

Unqualified

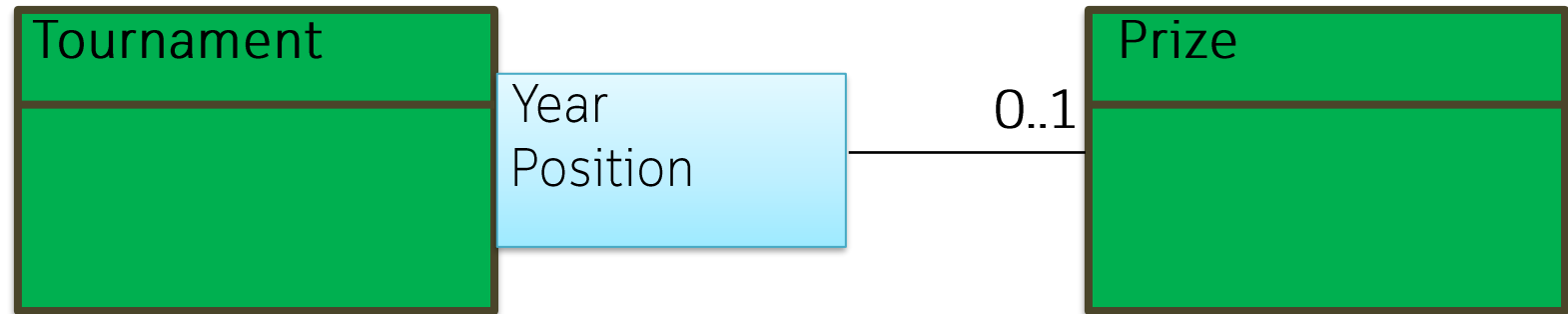


Qualified

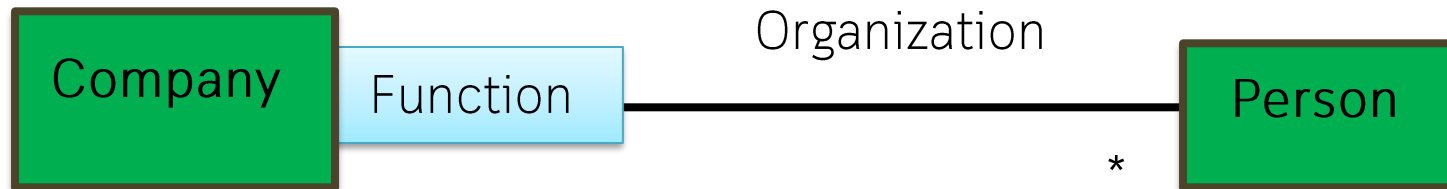


*** การทำแบบคล้ายๆ การหา PK ในการออกแบบฐานข้อมูล

Qualified Associations



Qualified Associations



ABC Inc.	President	Roger Rabbit
ABC Inc.	Vice President Finances	Joe Savemoney
ABC Inc.	Member of board	John Walker
ABC Inc.	Member of board	Susi Sanssouci
ABC Inc.	Member of board	Karl Eichbaum
XYZ Inc.	President	Donald Duck

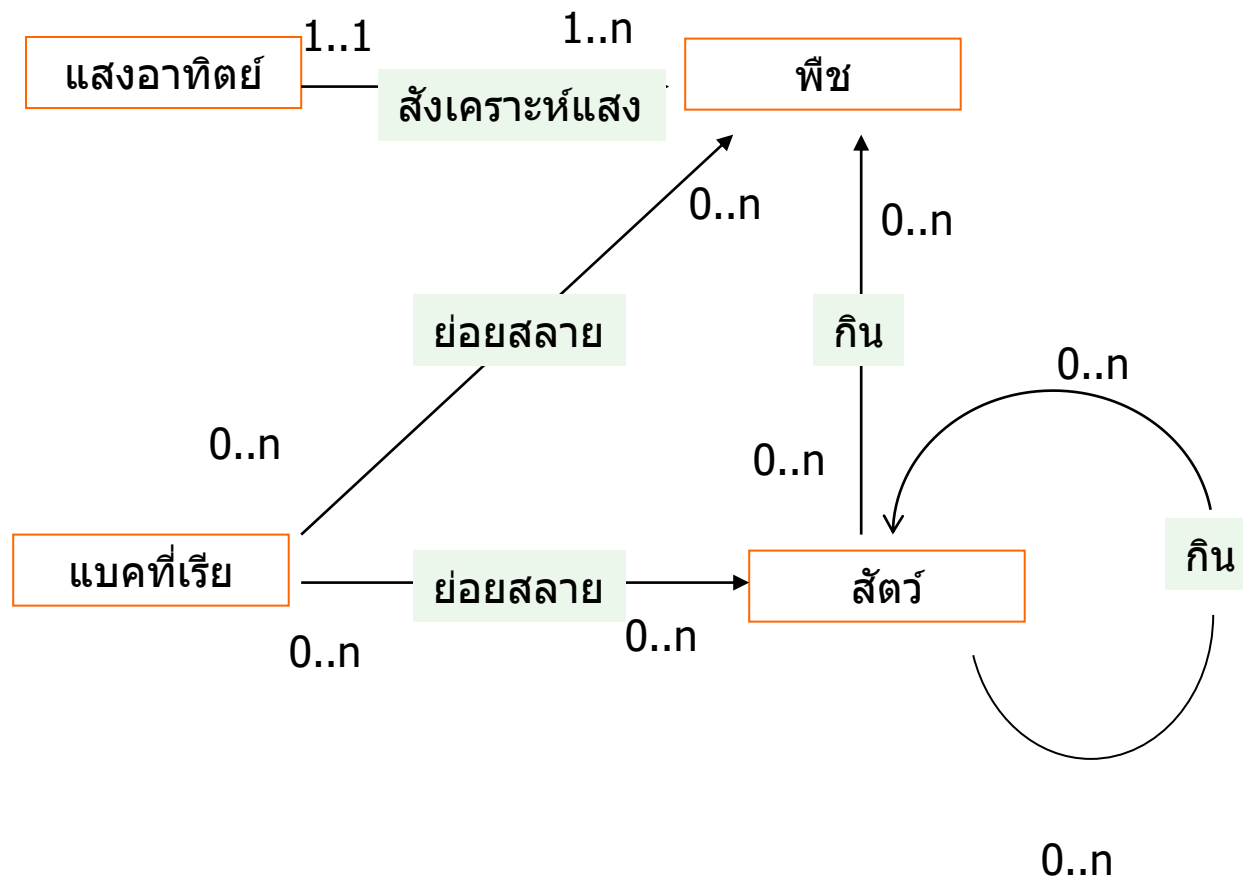
*** ความสัมพันธ์ระหว่าง บริษัท กับพนักงาน ไม่ชัดเจน แต่ถ้าระบุ ตำแหน่งไป จะชัดเจนขึ้น ทำให้รู้ได้ว่าหมายถึงใครใน บริษัท

Class Diagrams: Hints

- **class หนึ่งสามารถเป็นส่วนร่วมได้ในหลาย diagrams**
- **Diagrams ควรจะแสดงแง่มุมเฉพาะด้าน**
 - **จำนวนคลาสไม่มากเกินไป**
 - **จำนวน associations ไม่มากเกินไป**
 - **ซ่อน attributes และ operations ที่ไม่จำเป็นต้องแสดง**
- **อาจต้องมีการสร้าง diagram ซ้ำ ๆ กันหลายรอบ**

Difference between attributes and associations

- **Conceptual perspective**
 - ไม่แตกต่างกันมากนัก
 - **Attributes** เก็บค่าเดียว (single-valued)
- **Specification/implementation perspective**
 - **Attribute** เก็บค่า (Value) ไม่ใช่ตัวอ้างอิง (References)
 - ไม่มีการ share ส่วนของ attribute values ในระหว่าง instances ที่สร้างจากคลาสเดียวกัน
- **Often: Stores simple objects**
 - **Numbers, Strings, Dates, Money objects**



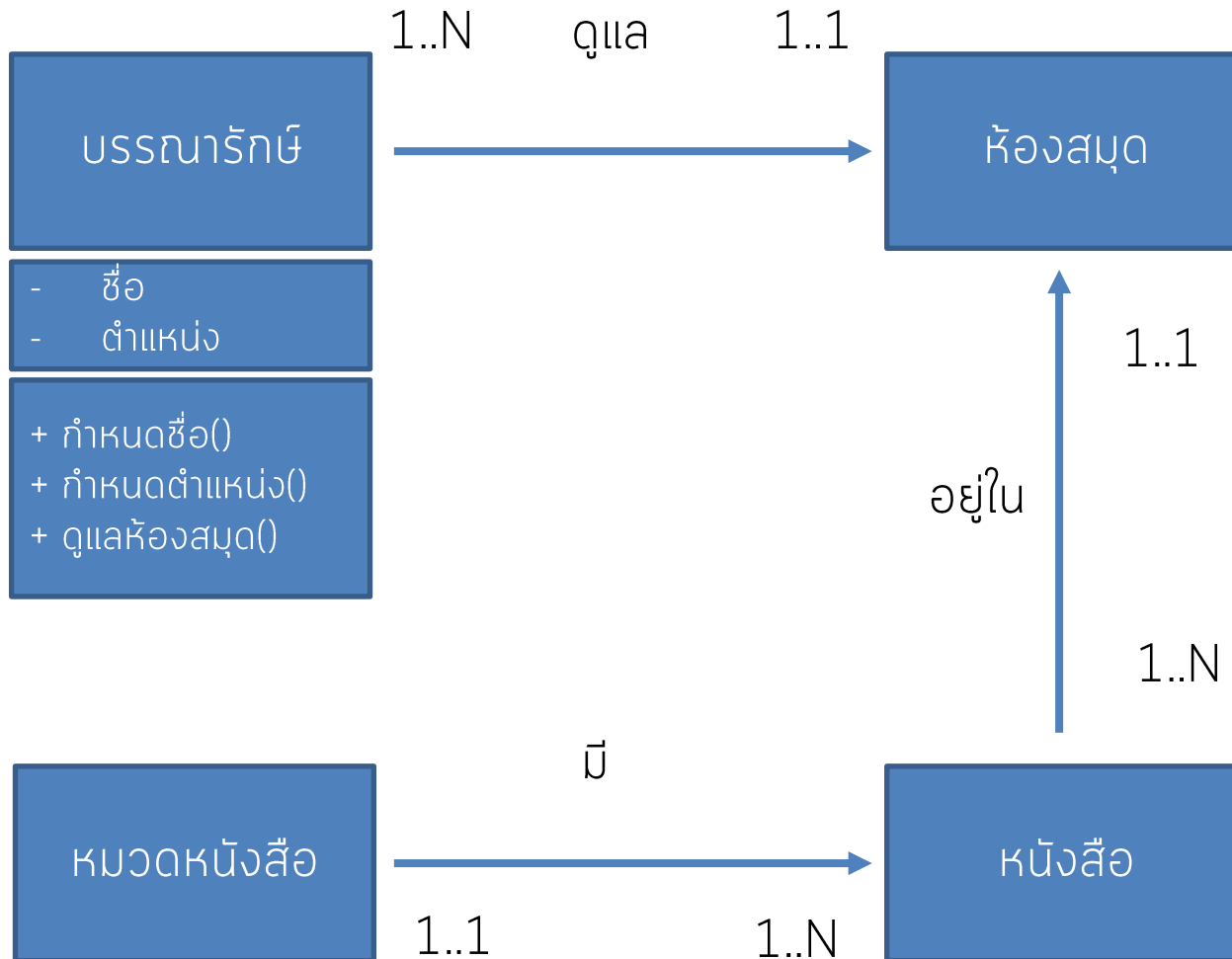
รูปแสดง Association ของห่วงโซ่อาหาร

แบบฝึกหัดท้ายบทที่ 2.4

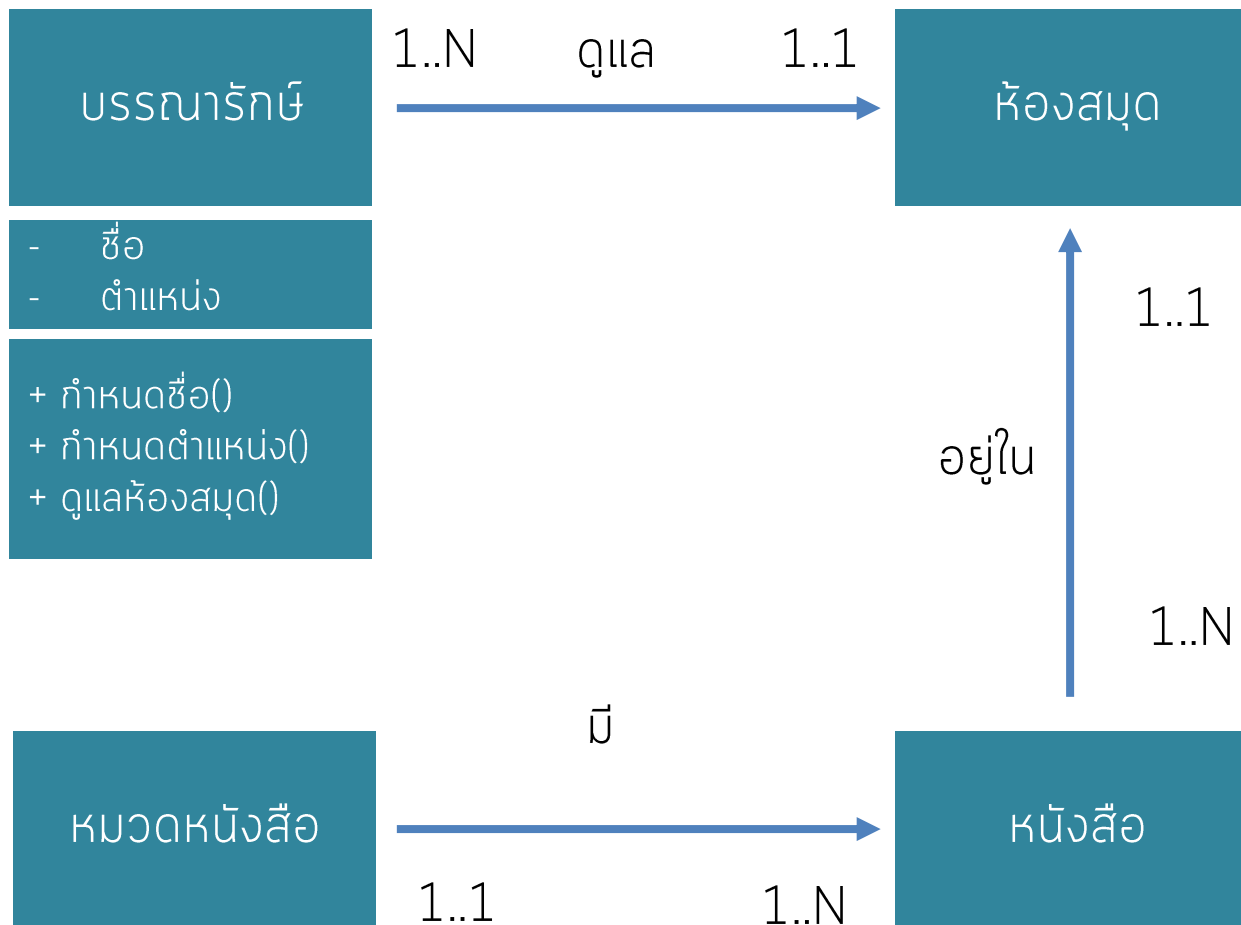
จงสร้าง แผนภาพ Association ที่แสดงถึง Association ที่มี Cardinality ที่ถูกต้องของความสัมพันธ์ดังต่อไปนี้

1. นักเรียน - วิชาเรียน - อาจารย์ผู้สอน
2. ธนาคาร - สมุดบัญชี - ผู้ฝากเงิน
3. ห้องสมุด - บรรณารักษ์ - หนังสือ - หมวดหนังสือ
4. เครื่องบิน - พนักงาน - ผู้โดยสาร - อาหาร
5. ผู้จัดการ - ร้านอาหาร - ลูกค้า - อาหาร - โต๊ะอาหาร

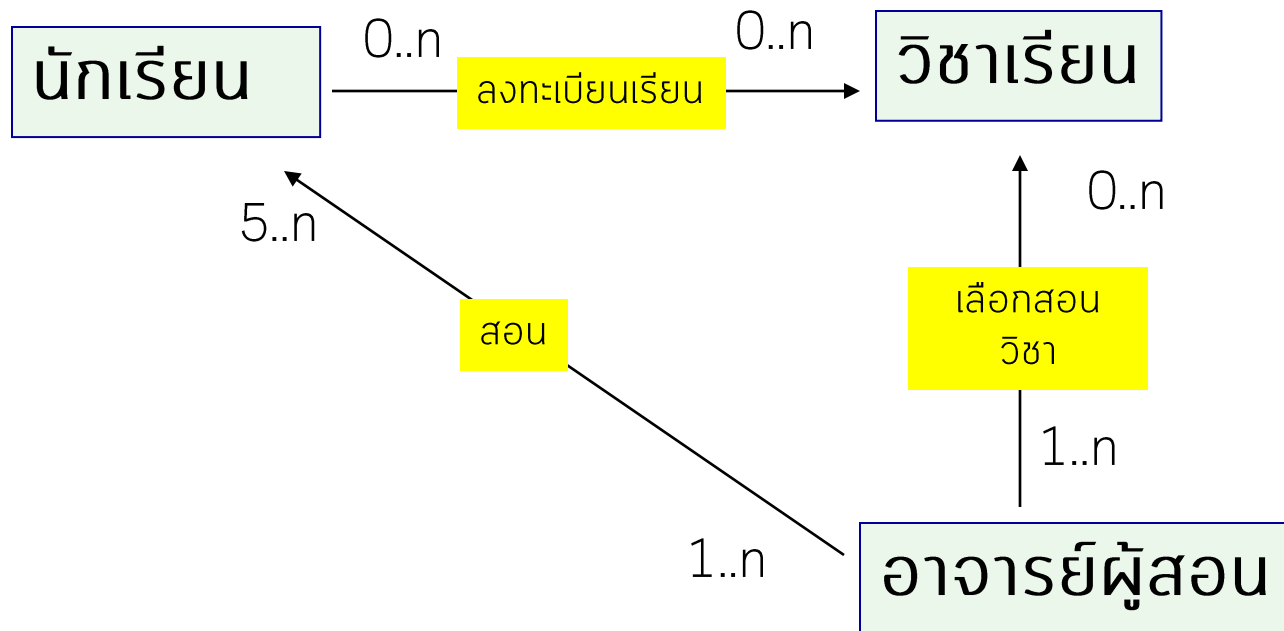
ห้องสมุด - บรรณารักษ์ - หนังสือ - หมวดหนังสือ



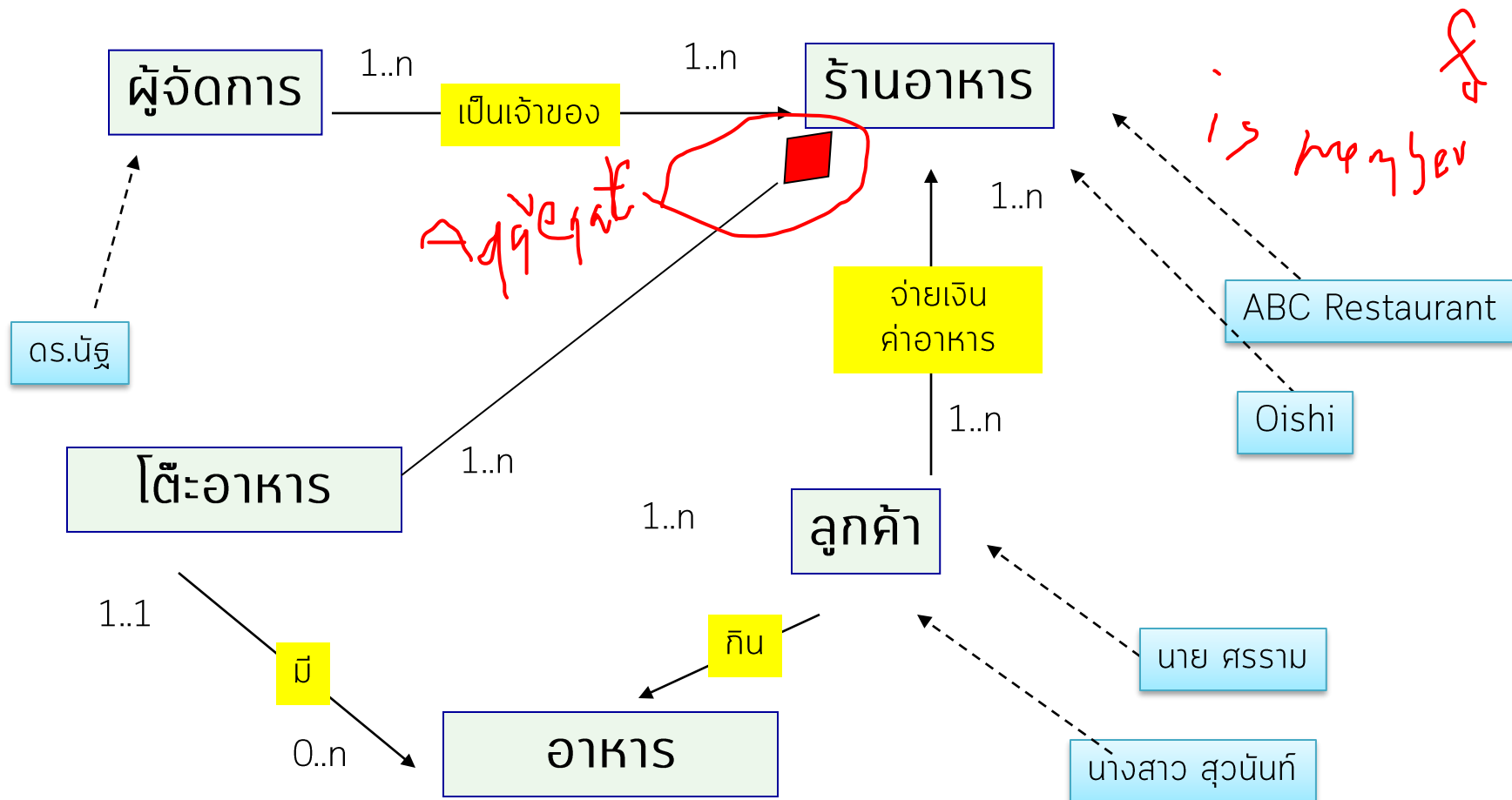
ห้องสมุด - บรรณารักษ์ - หนังสือ - หมวดหนังสือ



นักเรียน - วิชาเรียน - อาจารย์ผู้สอน



ผู้จัดการ - ร้านอาหาร - ลูกค้า - อาหาร - โต๊ะอาหาร



ผู้จัดการ - ร้านอาหาร - ลูกค้า - อาหาร - โต๊ะอาหาร

