



การสร้างแบบจำลองข้อมูล

Data Modeling



การสร้างแบบจำลองข้อมูล (Data modeling)

- การสร้างแบบจำลองข้อมูล คือ กระบวนการในการวิเคราะห์ถึงข้อมูลที่ จำเป็นต้องจัดเก็บในระบบ
- โดยการจัดเก็บข้อมูลนั้นจะต้องสอดคล้องกับกฎการดำเนินธุรกิจ (Business rules) ขององค์กรนั้นๆ
- เป้าหมายของการสร้างแบบจำลองข้อมูล คือ ใช้สำหรับแสดงความหมายของวิธีการจัดเก็บข้อมูลในฐานข้อมูลในเชิงแนวคิด (Conceptual Data Model) และใช้เป็นข้อกำหนดของการพัฒนาฐานข้อมูล



แบบจำลองข้อมูลเชิงแนวคิด

(Conceptual Data Model)

- แบบจำลองข้อมูลเชิงแนวคิด คือแบบจำลองที่แสดงโครงสร้างของข้อมูลและเงื่อนไขในเชิงความสัมพันธ์ของข้อมูลเหล่านั้น โดยแบบจำลองจะต้องสอดคล้องกับกฎการดำเนินธุรกิจ (Business rules) ขององค์กรนั้นๆ
- แบบจำลองข้อมูลเชิงแนวคิด จะต้องสามารถทำความเข้าใจได้ง่าย และเป็นอิสระกับ DBMS หรือวิธีการจัดการข้อมูลแบบใดๆ
- แบบจำลองข้อมูลเชิงแนวคิดจะถูกแปลงไปเป็นแบบจำลองฐานข้อมูลเชิงตรรกะ (Logical database model) และแบบจำลองฐานข้อมูลเชิงกายภาพ (Physical database model) ในขั้นตอนการออกแบบ เพื่อนำไปใช้เป็นข้อกำหนดของการสร้างเป็นฐานข้อมูลต่อไป



ตัวอย่างของ Business rules

- นิสิตสามารถลงทะเบียนได้หลายวิชาในหนึ่งภาคการศึกษา
- นิสิตจะลงทะเบียนเรียนรายวิชาหลายวิชาในเวลาเรียนเดียวกันไม่ได้
- อาจารย์สามารถสอนได้หลายวิชาในหนึ่งภาคการศึกษา
- นิสิตสามารถยืมหนังสือได้ครั้งละไม่เกิน 5 เล่ม
- นิสิตสามารถมีอาจารย์ที่ปรึกษาได้เพียงคนเดียว
- นิสิตแต่ละคนจะมี **User Account** เดียวในการเข้าระบบของมหาวิทยาลัย
- ฯลฯ

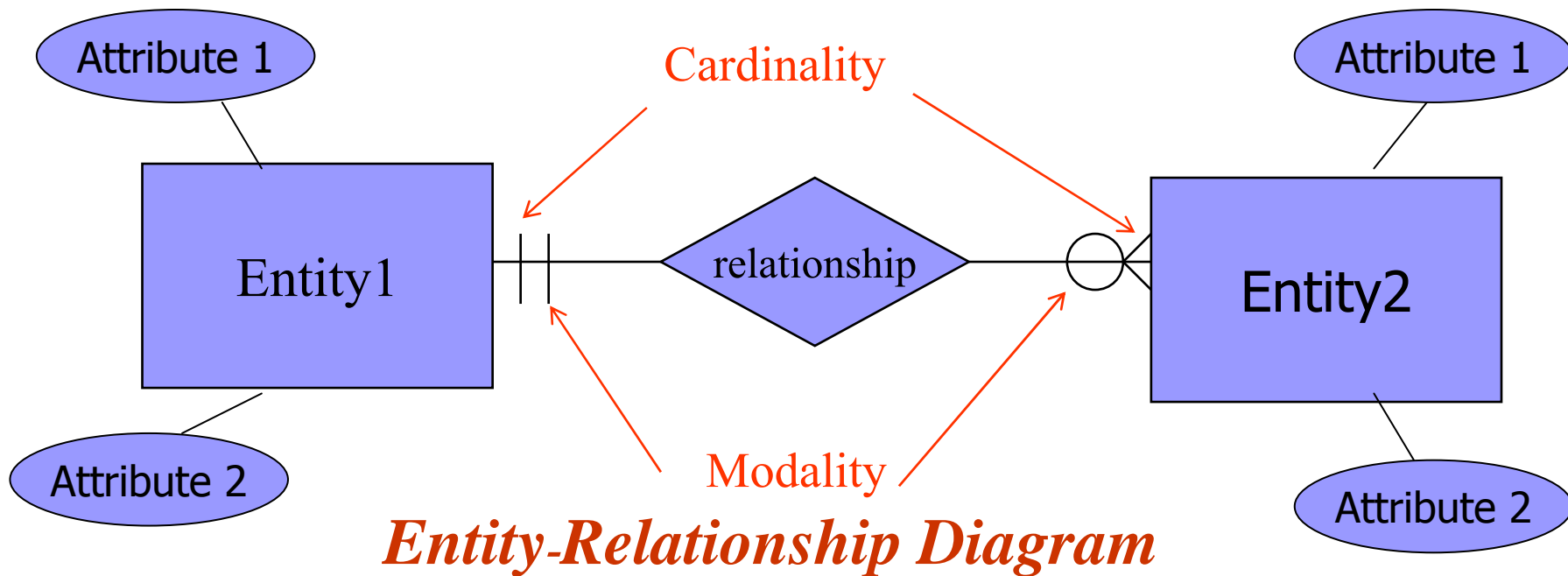
Data Entity

- **Entity** คือสิ่งที่เป็นตัวแทนของข้อมูลในเชิงนามธรรม มีความเป็นเอกเทศในตนเอง โดยข้อมูลที่ถูกแสดงด้วย **Entity** นั้นคือข้อมูลที่จะต้องเก็บไว้ในระบบ
- **Entity** จะต้องถูกอธิบายด้วยคุณลักษณะ (**Attributes**) ต่างๆ เพื่อให้ทราบถึงโครงสร้างข้อมูลของ **Entity** นั้นๆ
- ตัวอย่างของ **Entity** เช่น
 - ลูกค้า ผู้ผลิต พนักงาน สินค้า ฯ
 - นิสิต อาจารย์ วิชาเรียน ฯ
 - สถานที่เช่น โกดังเก็บสินค้า สาขาย่อยขององค์กร
 - เหตุการณ์ต่างๆ ที่เกิดขึ้น เช่น การลงทะเบียน การใช้โทรศัพท์ การเข้าทำงาน
 - ฯลฯ
- ในระบบหนึ่งจะมีการเก็บข้อมูลหลายชนิด และการจัดเก็บข้อมูลแต่ละชนิดจะต้องถูกบังคับให้เป็นไปตาม **Business rules** ที่กำหนดไว้ ดังนั้นจึงจำเป็นต้องกำหนดโครงสร้างของการจัดเก็บข้อมูลด้วยโครงสร้างความสัมพันธ์ของ **Entities** ต่างๆ

แผนภาพเอนทิตีและความสัมพันธ์

(Entity-Relationship Diagram: ERD)

- แผนภาพเอนทิตีและความสัมพันธ์ ERD ใช้สำหรับแสดงแบบจำลองข้อมูลเชิงแนวคิดในแบบแผนภาพ โดยมีสัญลักษณ์ดังนี้



องค์ประกอบของ ERD

- **Entity** คือ ชนิดของตัวข้อมูลที่มีความเป็นเอกเทศในตนเอง เช่น คน สถานที่ สิ่งของ หน่วยงาน เป็นต้น
- **Attributes** คือ องค์ประกอบของตัวข้อมูล แสดงถึงคุณลักษณะต่างๆ ของ **Entity** เช่น **Attributes** ของคน ได้แก่ ชื่อ ความสูง น้ำหนัก สีผม เชื้อชาติ ฯ
- **Relationship** คือ ข้อกำหนดในแบบจำลองข้อมูลที่แสดงถึงความสัมพันธ์ในแง่มุมต่างๆ ของตัวข้อมูลต่างๆ (**Entities**) ในระบบ ซึ่ง สามารถระบุแง่มุมต่อไปนี้ได้
 - **Cardinality** คือการระบุถึงความสัมพันธ์ในเชิงจำนวนระหว่าง **Entities**
 - **Modality** การระบุถึงความจำเป็นในการมีอยู่ของตัวข้อมูลที่สัมพันธ์กัน



ลักษณะของ Attributes

■ Candidate keys Attributes

- คือ **Attributes** บางส่วนที่ถูกเลือกมาสำหรับอธิบายลักษณะของ **Entity** ที่กำลังทำการวิเคราะห์
- เช่น **Entity** ข้อมูลนิสิต อาจมี **Candidate keys** ได้แก่ รหัสประจำตัว ชื่อ นามสกุล ที่อยู่ สาขาที่เรียน เท่านั้น ส่วนลักษณะอื่นๆ เช่น ความสูง สีผิว ชื่อเล่น ไม่ได้ อยู่ในความสนใจก็ไม่จำเป็นต้องเลือกมา

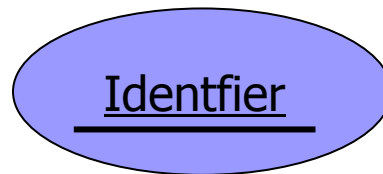
■ Identifier Attributes

- คือบาง **Candidate key attributes** ของ **Entity** ซึ่งสามารถใช้แยกแยะความแตกต่างของตัวข้อมูลในกลุ่มได้ เช่น ข้อมูลนิสิต เราอาจจะใช้ รหัสนิสิต เป็น **Identifier** ในการจำแนกความแตกต่างของตัวข้อมูลแต่ละตัว

หลักในการเลือก Identifier Attribute

■ หลักในการเลือก Identifier Attribute

- เลือก Attribute ที่ไม่มีการเปลี่ยนแปลงค่า
- เลือก Attribute ที่ไม่เป็นค่าว่าง Null
- เลือก Attribute ที่มีค่าไม่ซ้ำกัน



สัญลักษณ์ของ Identifier จะมีขีดเส้นใต้



Multi-valued Attributes

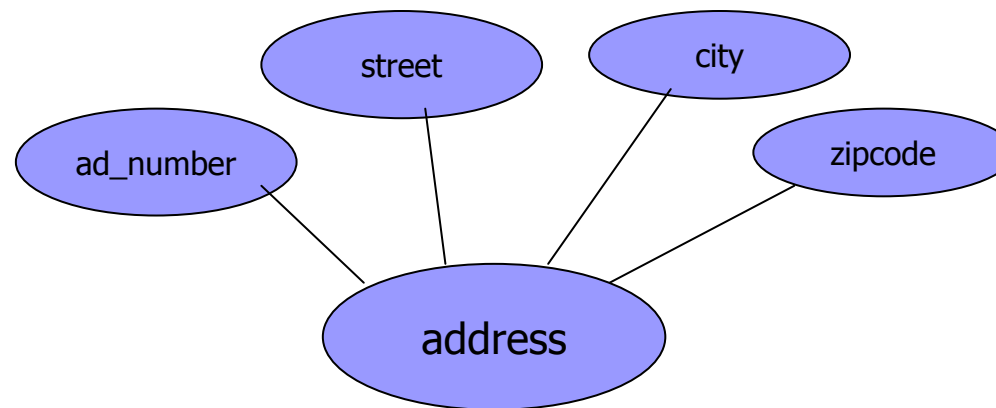
- คือ **Attribute** ที่สามารถมีค่าได้หลายค่า
 - เช่น นิสิตหนึ่งคนอาจมีความสามารถพิเศษหลายด้าน
 - ใบเสร็จหนึ่งใบมีรายการชำระหลายรายการ
- สัญลักษณ์สำหรับ **Multi-valued Attributes**



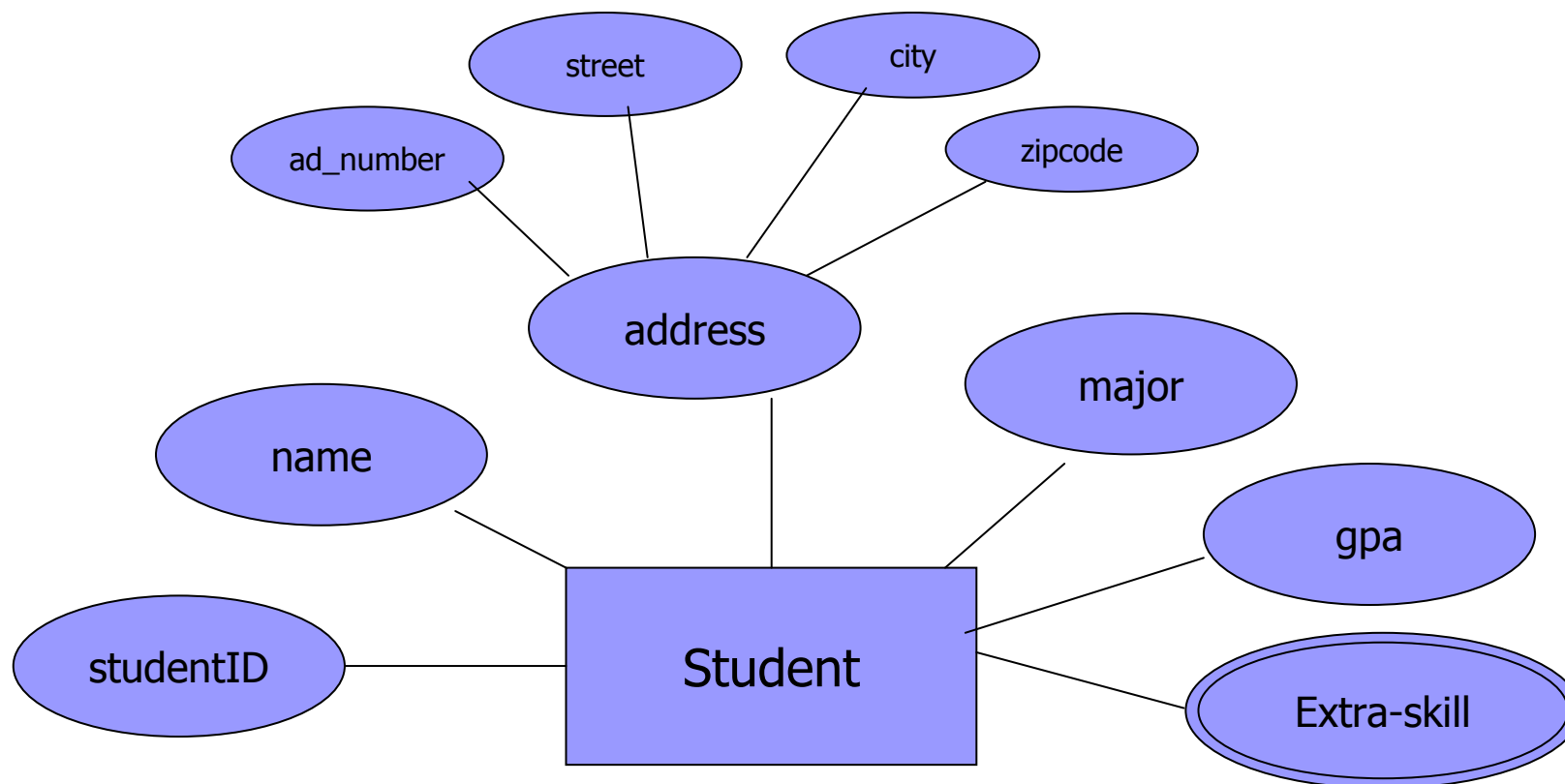
Multi-value

Composite Attributes

- **Composite Attribute** คือ **Attribute** ที่สามารถอธิบายด้วย **Attribute** ย่อยๆ ได้
- เช่น ที่อยู่ (**Address**) สามารถแตกรายละเอียดเป็น เลขที่บ้าน ชื่อถนน ตำบล อำเภอ จังหวัด และรหัสไปรษณีย์ เป็นต้น

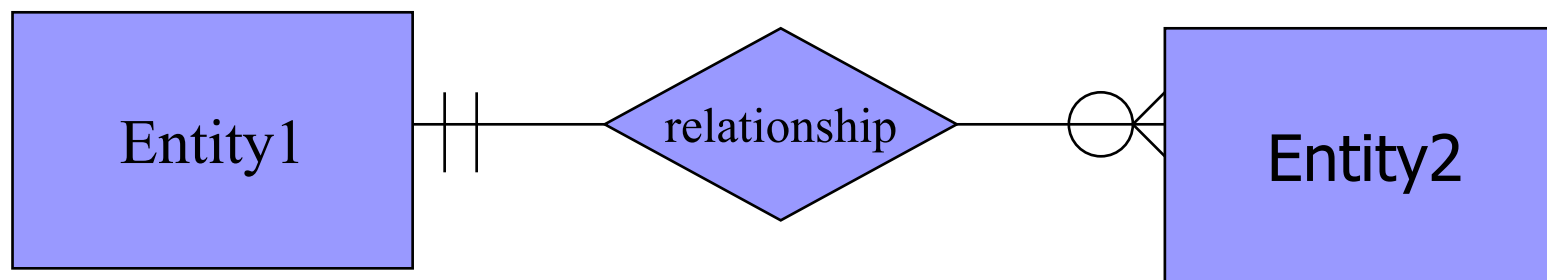


ตัวอย่าง ERD ของนิสิต



ความสัมพันธ์ (Relationship)

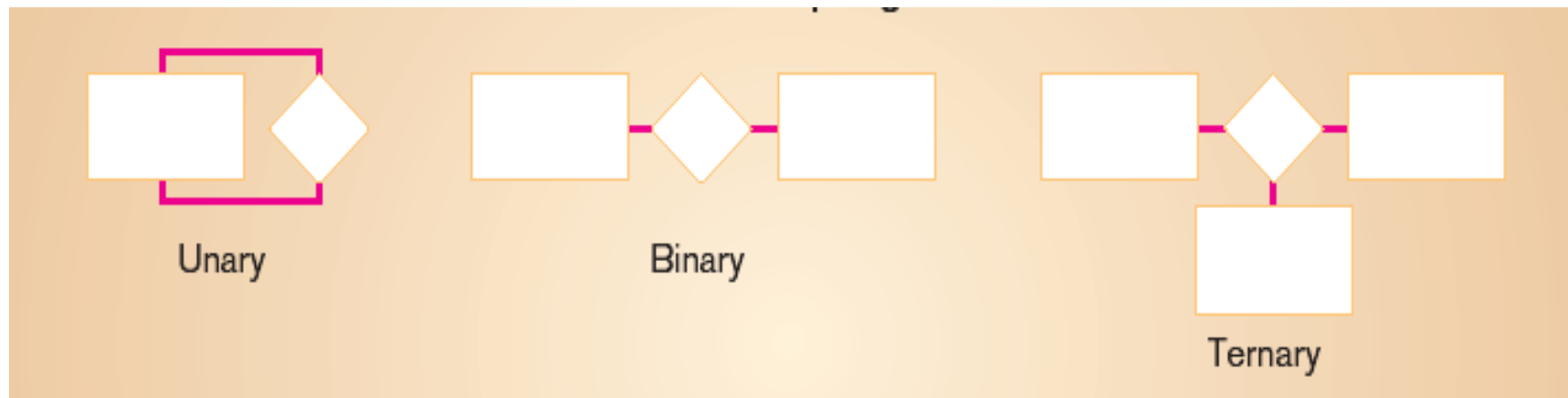
- **Relationship** คือ ข้อกำหนดในแบบจำลองข้อมูลที่แสดงถึงความสัมพันธ์ในแง่มุมต่างๆ ของตัวข้อมูล



องศาแห่งความสัมพันธ์ในแบบจำลองข้อมูล

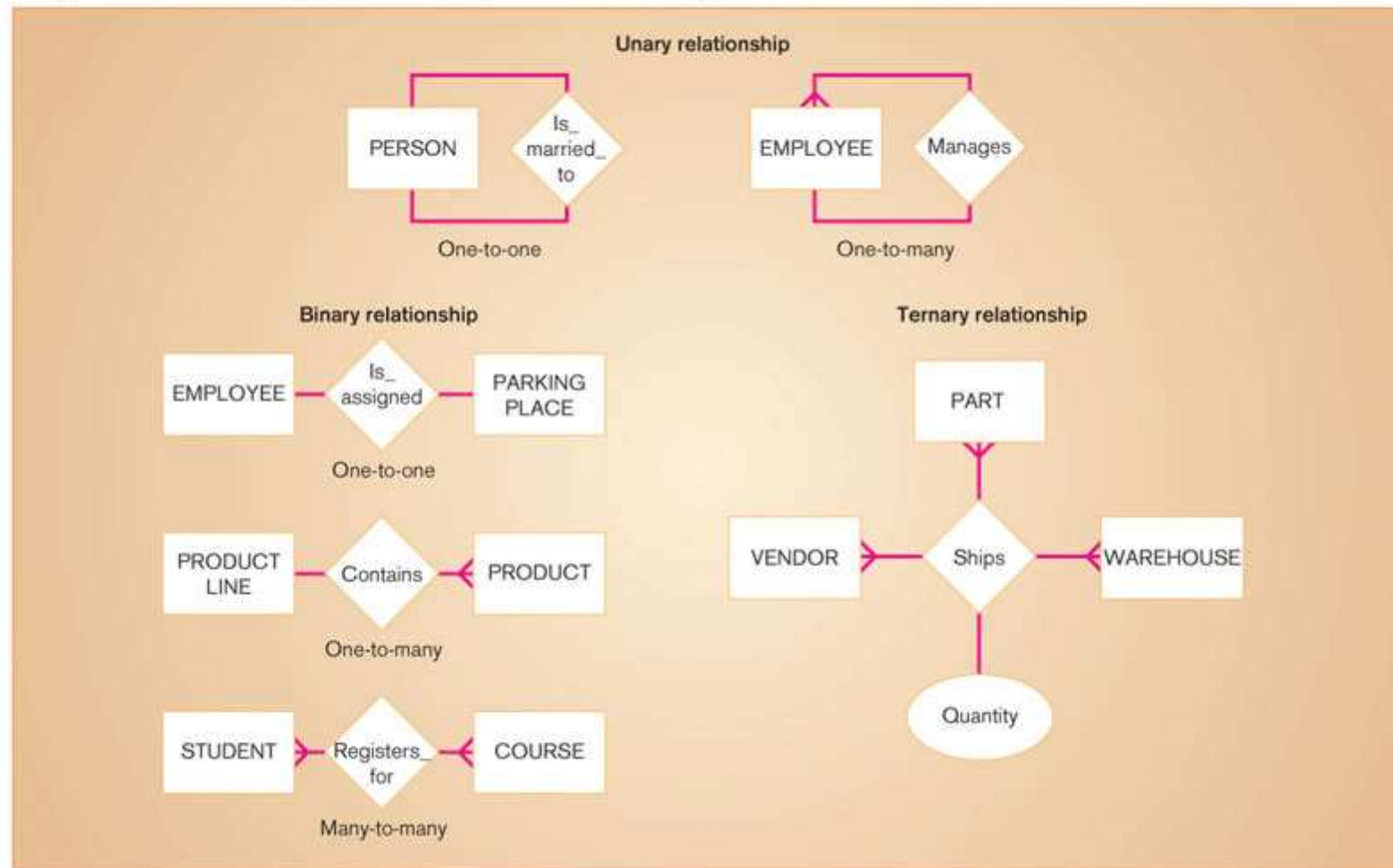
Degree of relationship

- องศาแห่งความสัมพันธ์ของ **Entity** สามารถจำแนกได้ 3 ลักษณะ
 - **Unary relationship** (ความสัมพันธ์ในตัวเอง)
 - **Binary relationship** (ความสัมพันธ์ระหว่าง 2 **Entities**)
 - **Ternary relationship** (ความสัมพันธ์ระหว่าง 3 **Entities**)



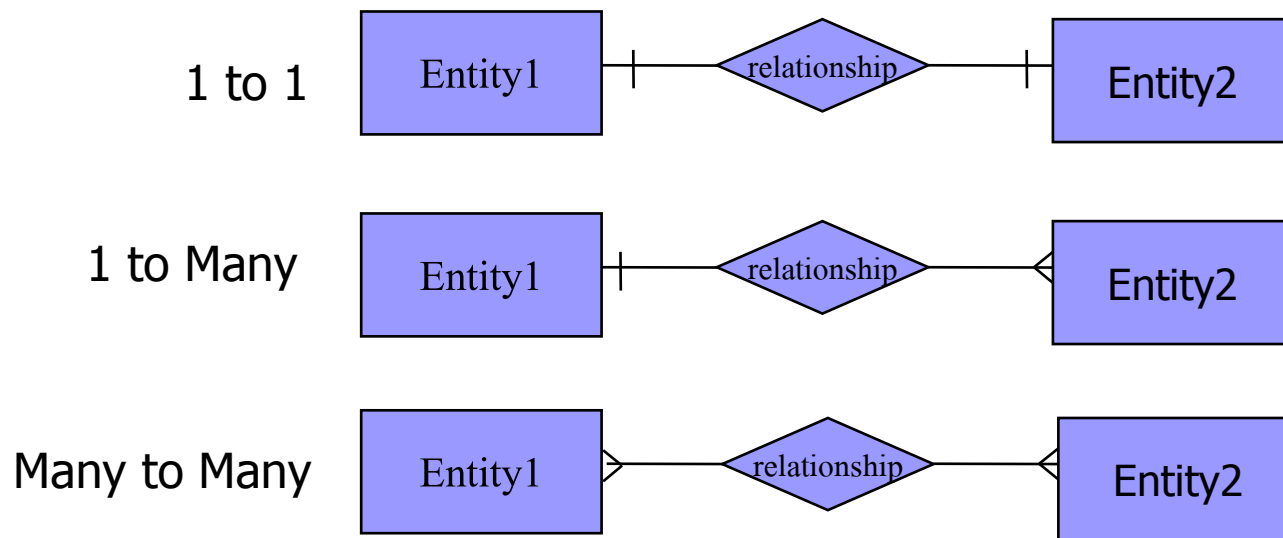
ตัวอย่าง

Figure 9-6 Example relationships of different degrees



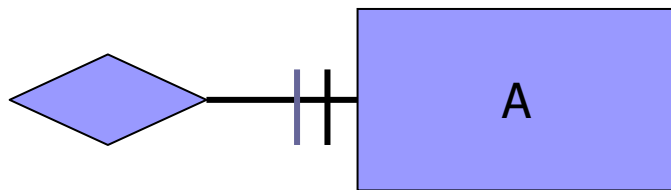
Cardinality บนความสัมพันธ์

- **Cardinality** คือการระบุถึงความสัมพันธ์ในเชิงจำนวนของตัวข้อมูล (data instance) ระหว่าง **Entities**

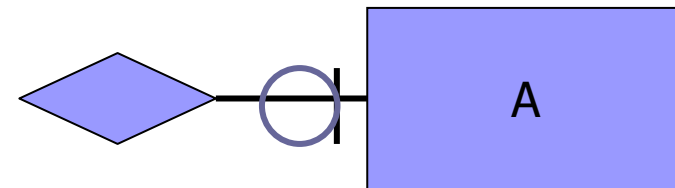


การระบุถึงความจำเป็นในการมีอยู่ของตัวข้อมูล

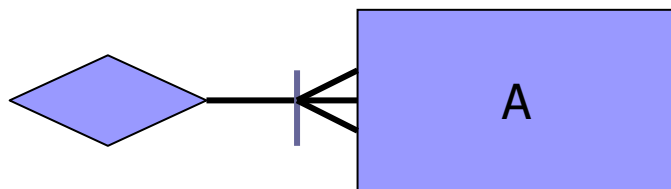
- **Mandatory Cardinality** คือการระบุว่าตัวข้อมูลนั้นจำเป็นต้องมีอยู่ในความสัมพันธ์
- **Optional Cardinality** คือการระบุว่าตัวข้อมูลนั้นอาจจะมีหรือไม่มีในความสัมพันธ์ก็ได้



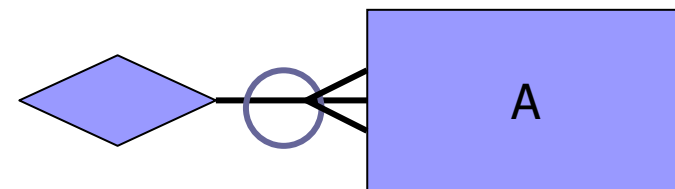
จะต้องมี A แน่นอน 1



อาจไม่มี A เลย แต่ถ้ามีก็มีได้แค่ 1 เท่านั้น



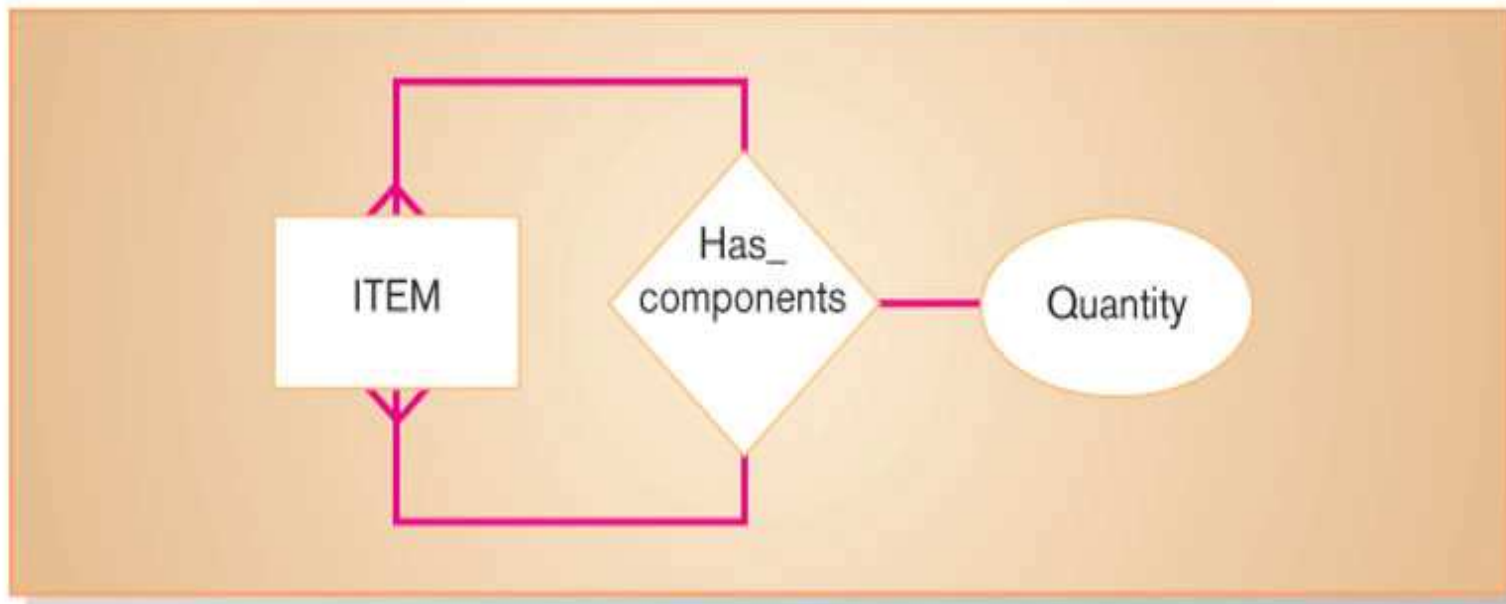
มี A แน่นอนอย่างน้อย 1 หรืออาจมีมากกว่านั้น



อาจมี A ได้มากกว่า 1 หรือ ไม่มีเลยก็ได้

Unary Relationship Example

Figure 9-7a Bill-of-materials unary relationship - Many-to-many relationship



Binary Relationship Examples

Figure 9-8a Examples of cardinalities in relationships -
Mandatory cardinalities

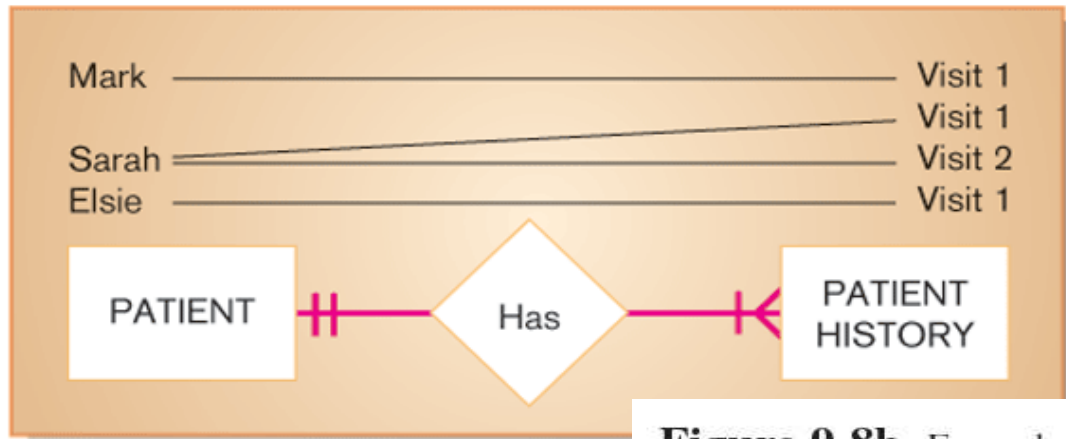
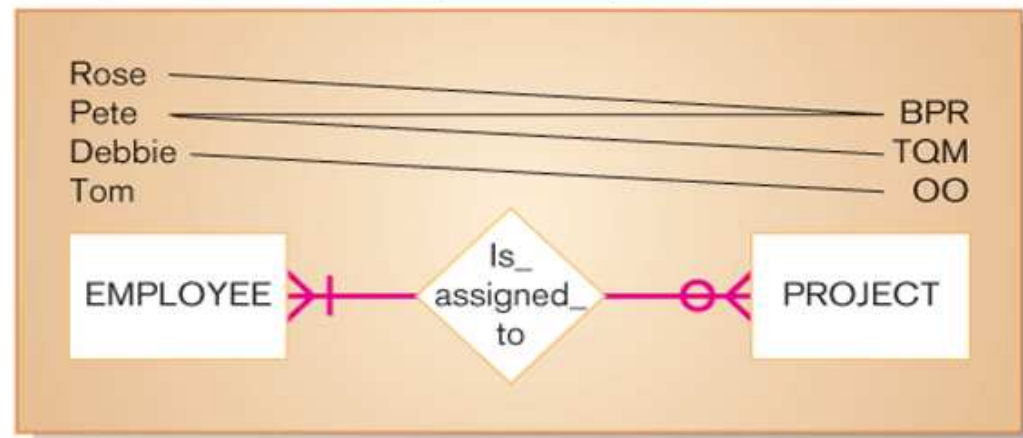
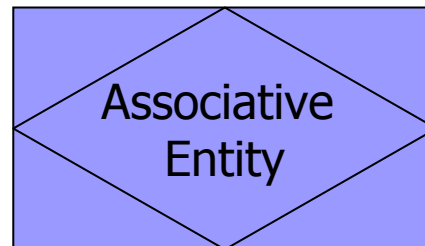


Figure 9-8b Examples of cardinalities in relationships -
One optional, one mandatory cardinality



Associative Entities

- **Associative Entity** คือ **Entity** พิเศษที่ทำหน้าที่เหมือน **Relationship** ซึ่งจะเป็นตัวกลางในการโยงความสัมพันธ์ให้กับ **Entities** อื่นที่ไม่มีความสัมพันธ์โดยตรง
- **Associative Entity** สามารถใช้แทน **Relationship** ที่มี **Attributes** ได้ ขึ้นอยู่กับนักวิเคราะห์ว่าจะเลือกใช้แบบใด



สัญลักษณ์

A relationship with an attribute

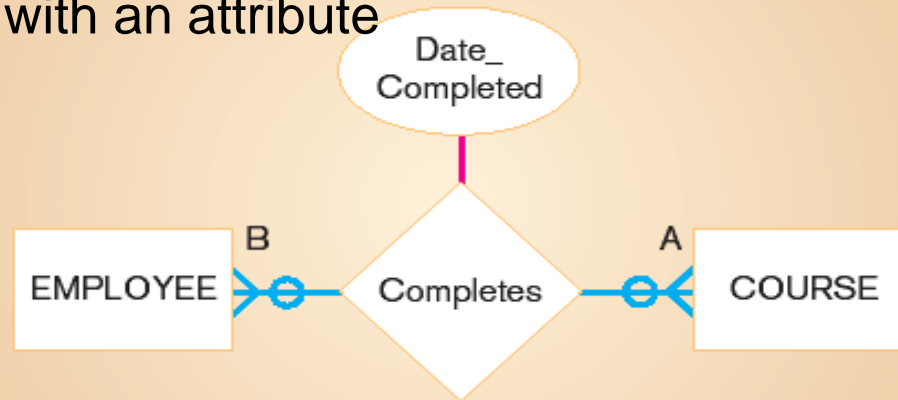
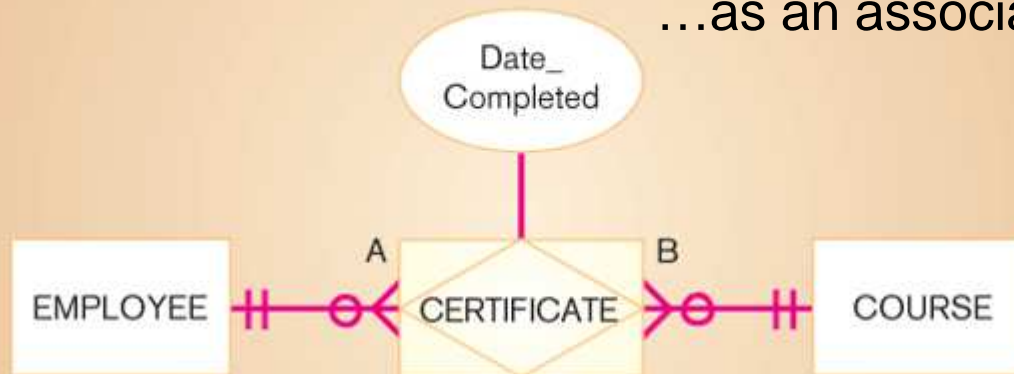
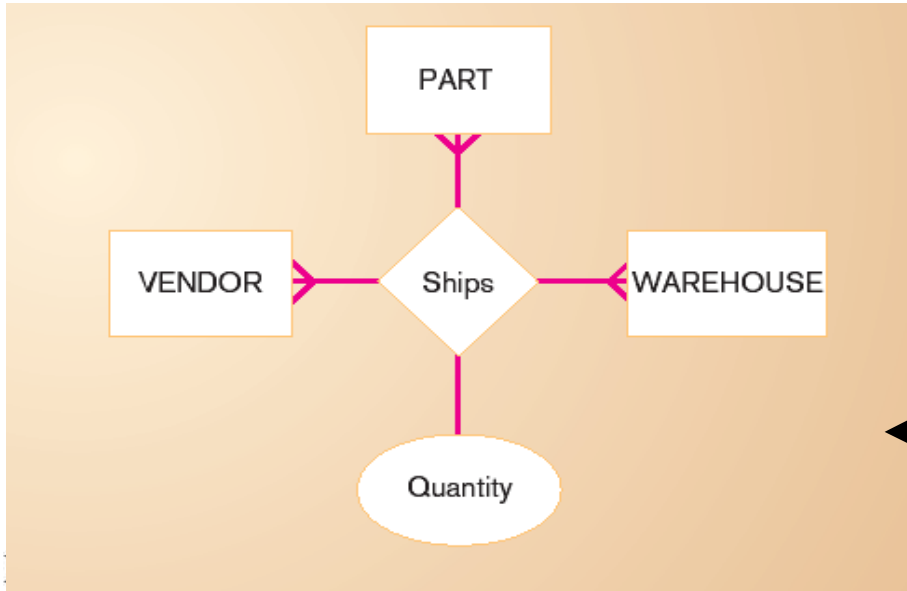


Figure 9-9 Example associative entity

เท่าเทียมกัน

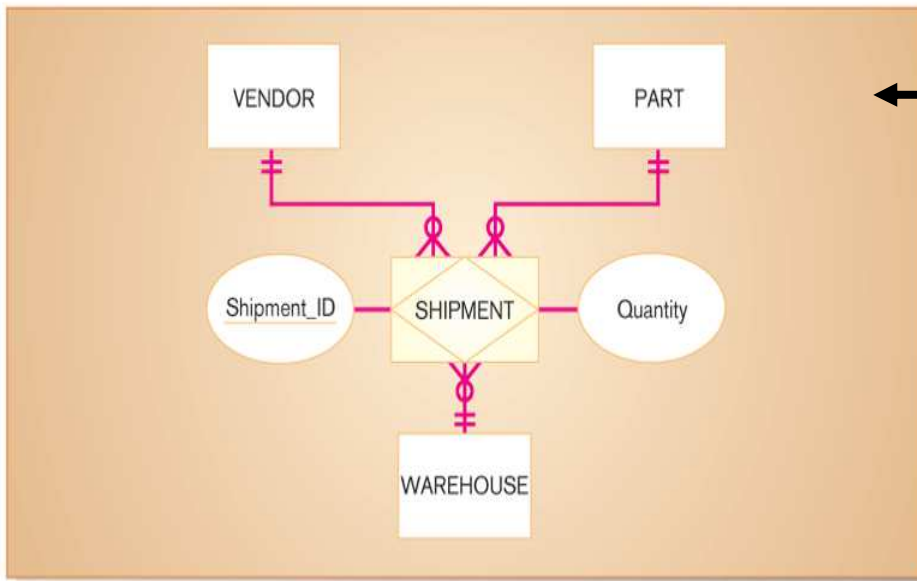
...as an associative entity





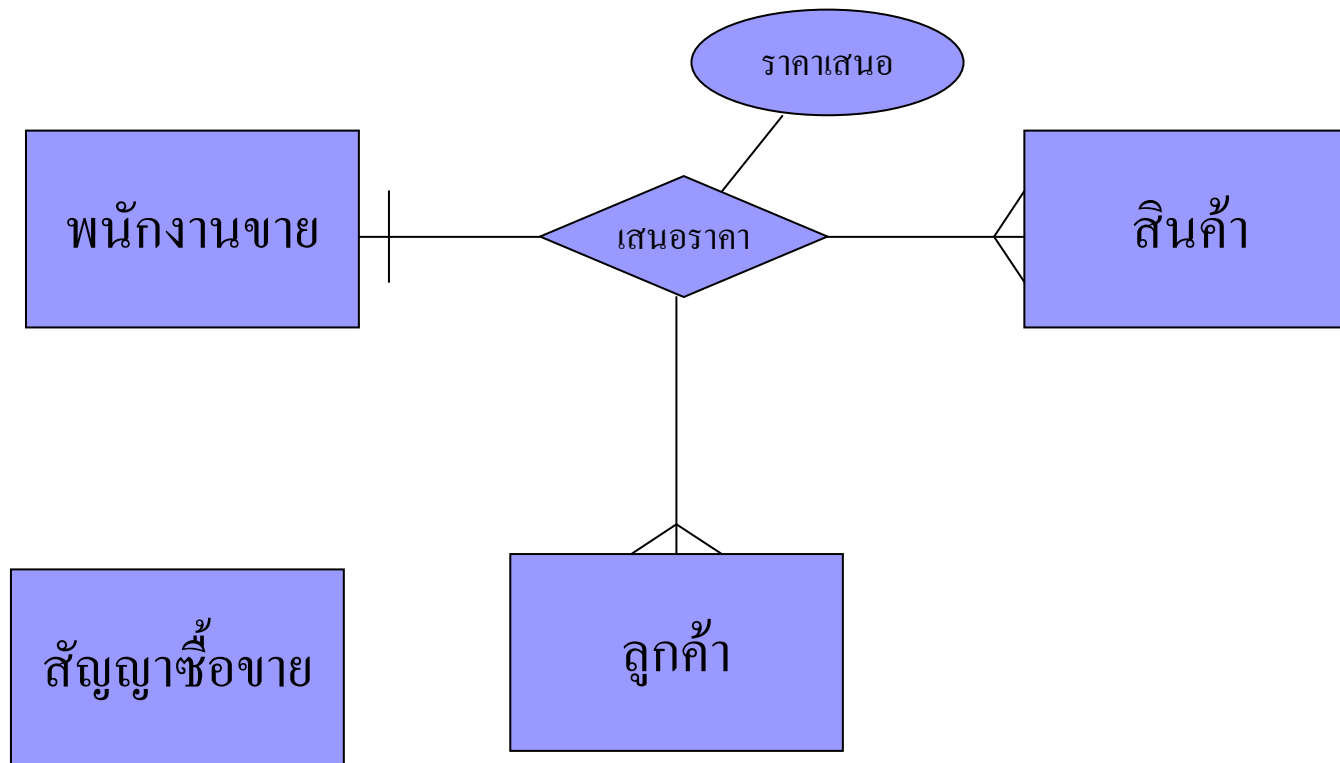
Ternary relationship

เท่าเทียมกัน



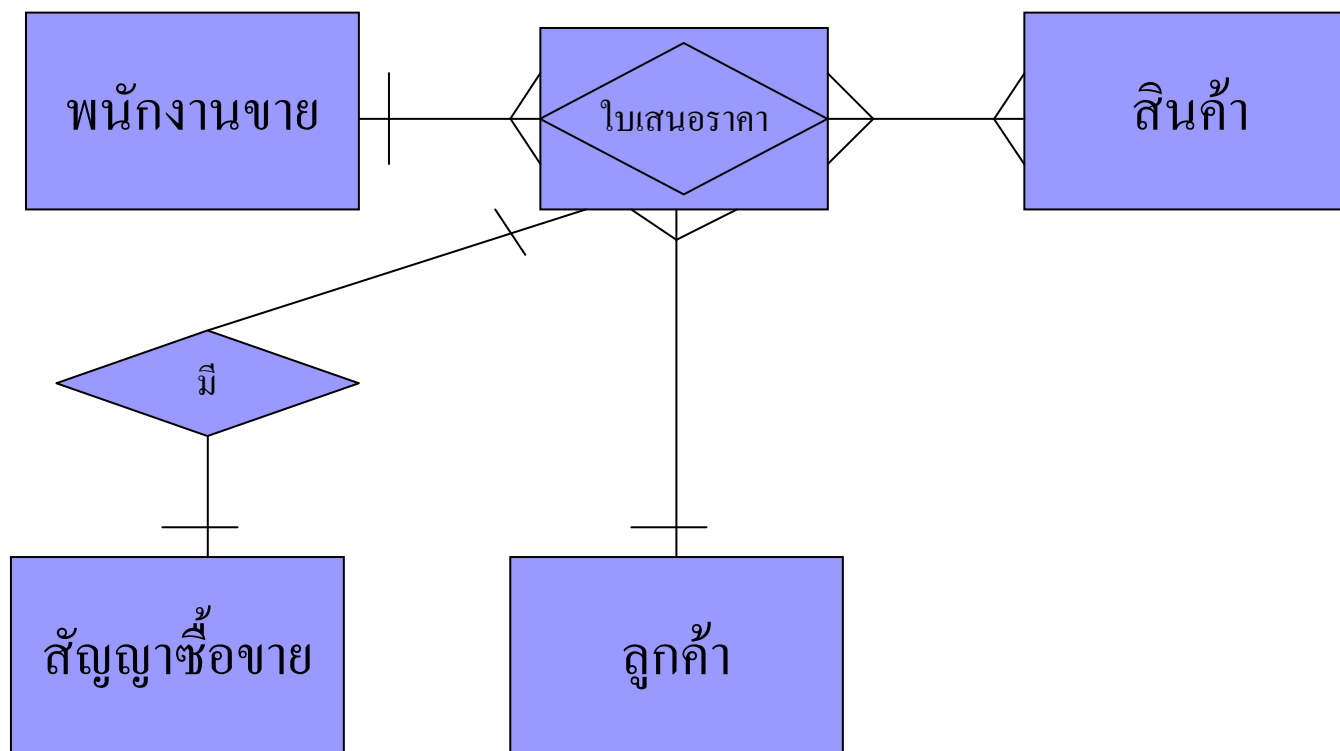
...as an associative entity

ตัวอย่างกรณีที่ใช้ Associative Entity



ต้องการเชื่อมความสัมพันธ์ของสัญญาซื้อขายเข้าร่วมกับการเสนอราคาด้วย

ตัวอย่างกรณีที่ใช้ Associative Entity



Supertypes/Subtypes Relationship

- ความสัมพันธ์อีกรูปแบบหนึ่งในแผนภาพ ERD คือความสัมพันธ์ในเชิง SuperType และ SubType
- SuperType Entities คือ Entities ทั่วไปที่มี Attributes ไว้สำหรับถ่ายทอดให้ SubTypes Entities ต่างๆของตน
- Subtype Entities คือ Entities ซึ่งสืบทอด Attributes จาก SuperType Entities และมี Attributes ของตนเองเพิ่มเติมขึ้นมา

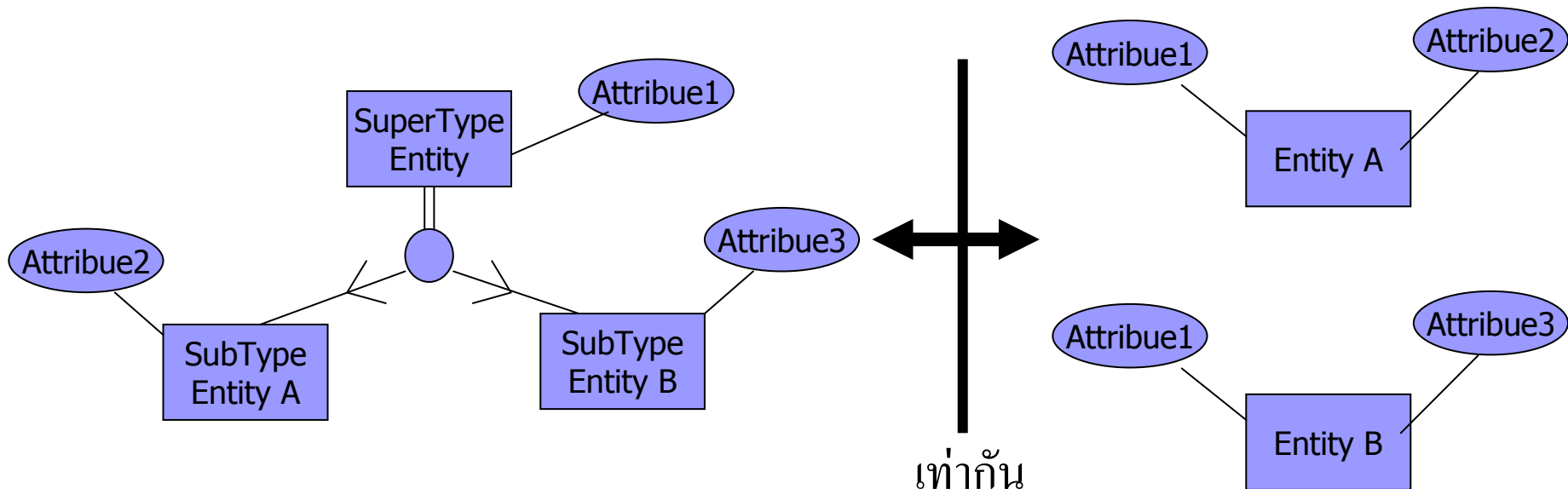
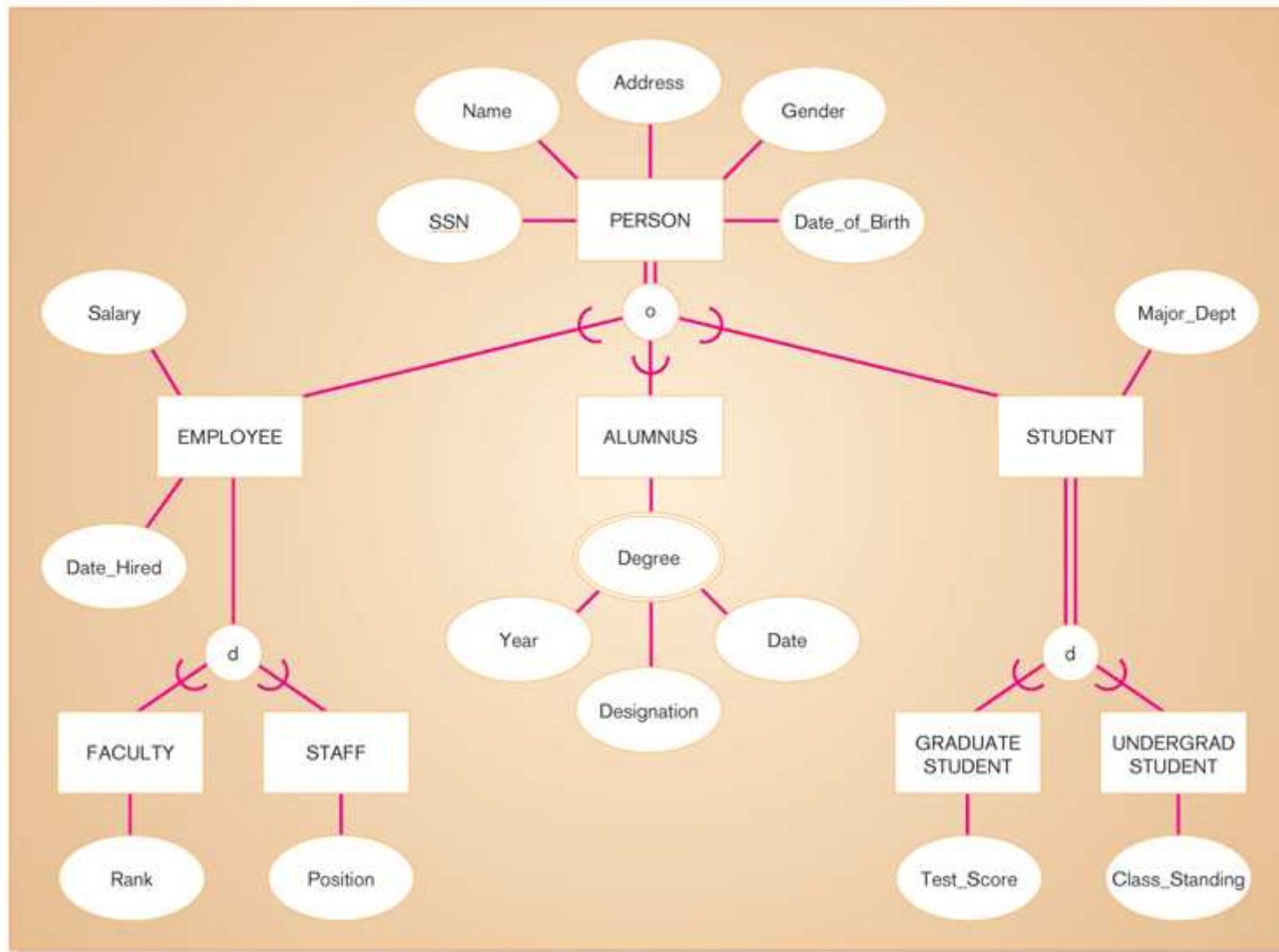


Figure 9-12 Example of supertype/subtype heirarchy





ขั้นตอนการวิเคราะห์แบบจำลองข้อมูล

1. พิจารณาข้อมูลนำเข้า (Input Data) ต่างๆ ที่จำเป็นต้องจัดเก็บในระบบ
2. วิเคราะห์โครงสร้างของแต่ละข้อมูลนำเข้า
3. วิเคราะห์ **Entities** และ **Attributes**
4. พิจารณา **Business rules** ของแต่ละ **Entity**
5. กำหนดความสัมพันธ์ตาม **Business rules**
6. สร้าง **ER Diagram**
7. ตรวจสอบความถูกต้อง และทำการปรับให้เหมาะสม (**Optimization**)



การวิเคราะห์แบบจำลองข้อมูลจาก Input Data

- พิจารณาจาก **Input Forms** ต่างๆ ว่ามีการบันทึกข้อมูลอะไรบ้าง
- ตัวอย่างเช่นใน ระบบทะเบียน จะบันทึกข้อมูลต่อไปนี้
 - ข้อมูลนิติศ
 - ข้อมูลอาจารย์
 - ข้อมูลรายวิชา
 - ข้อมูลการลงทะเบียน

วิเคราะห์โครงสร้างของแต่ละข้อมูลนำเข้า

■ Input Data ที่ต้องเก็บในระบบ

- ข้อมูลนิสิต = รหัสนิสิต+ชื่อ+สาขาวิชา+คณะ+ปีที่เข้าศึกษา
- อาจารย์ผู้สอน = รหัสอาจารย์+ชื่อ+ภาควิชา+คณะ
- รายวิชา = รหัสวิชา+ชื่อวิชา+จำนวนหน่วยกิต +คำอธิบายรายวิชา+สาขาวิชา+คณะ+{อาจารย์ผู้สอน}^{1..*}
- การลงทะเบียน = รหัสนิสิต+ภาคการศึกษา+ปีการศึกษา+วันที่ลงทะเบียน +{รายวิชาที่ลง}^{1..*}

วิเคราะห์ Entities และ Attributes

■ พิจารณาที่โครงสร้างข้อมูล

■ ข้อมูลนิสิต = รหัสนิสิต+ชื่อ+สาขาวิชา+คณะ+ปีที่เข้าศึกษา

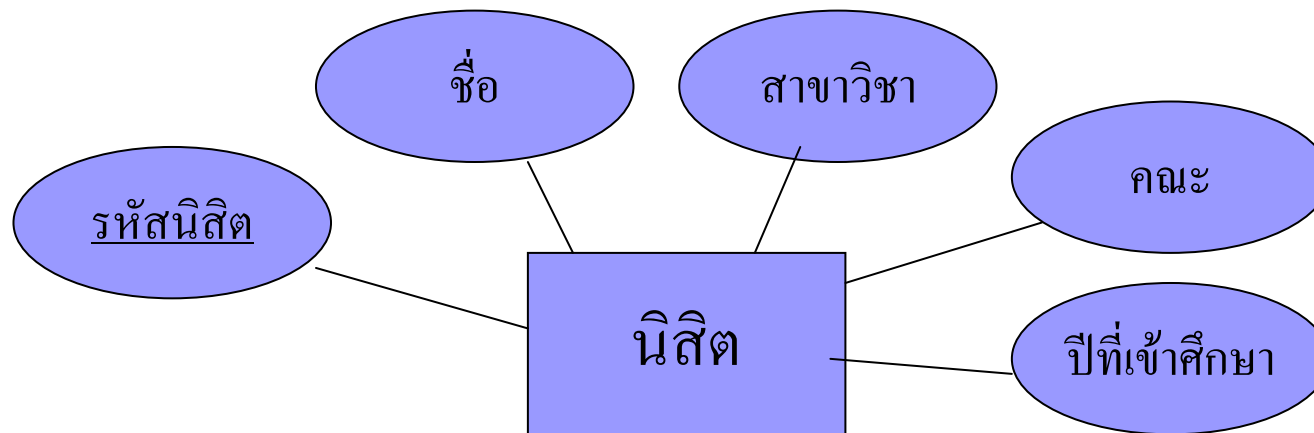
■ อาจารย์ผู้สอน = รหัสอาจารย์+ชื่อ+ภาควิชา+คณะ

Entity ■ รายวิชา = รหัสวิชา+ชื่อวิชา+จำนวนหน่วยกิต +คำอธิบายรายวิชา+สาขาวิชา+คณะ+{อาจารย์ผู้สอน}^{1..*}

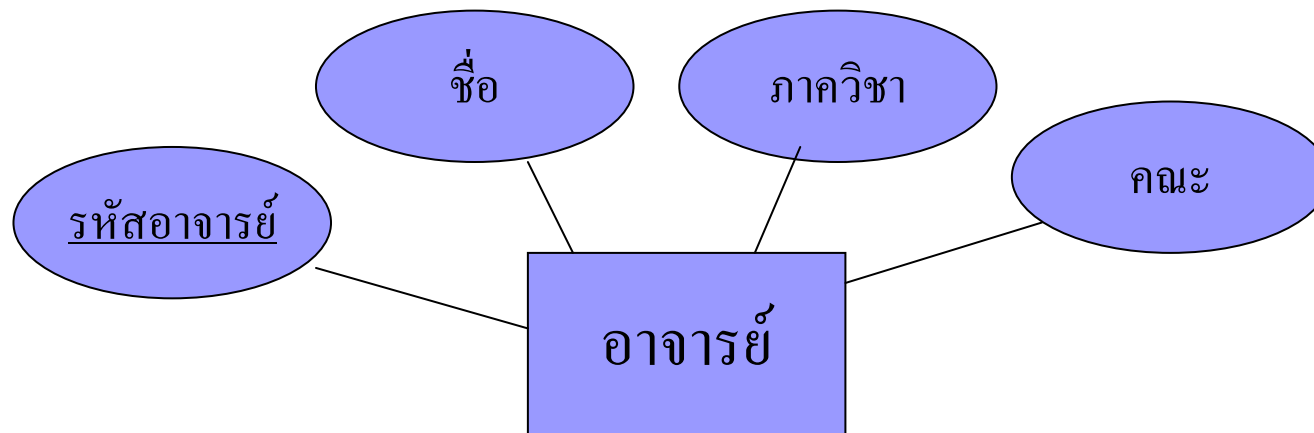
■ การลงทะเบียน = รหัสนิสิต+ภาคการศึกษา+ปีการศึกษา+วันที่

ลงทะเบียน + {รายวิชาที่ลง}^{1..*} Multi Value Attributes

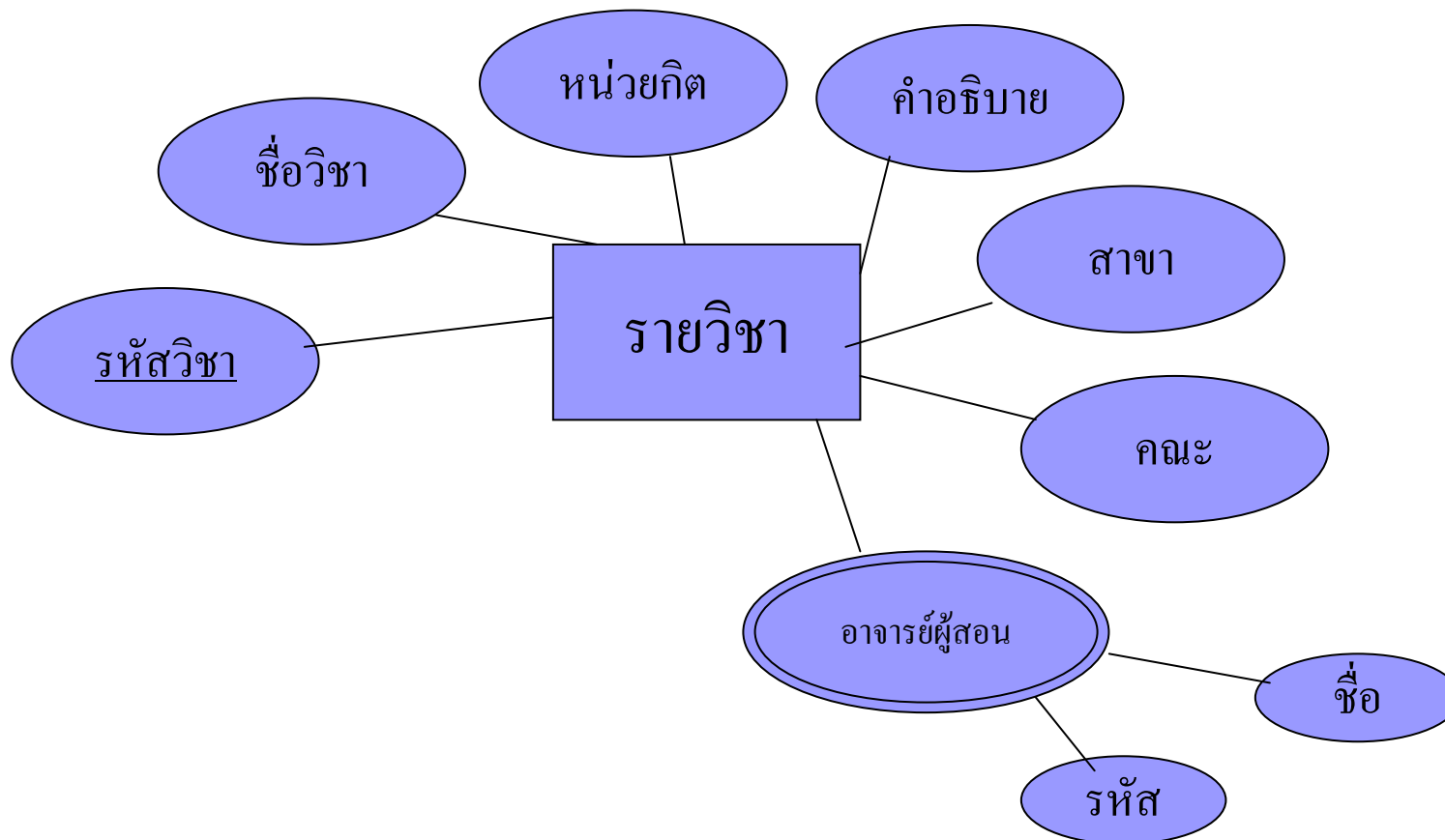
Entities ที่วิเคราะห์ได้



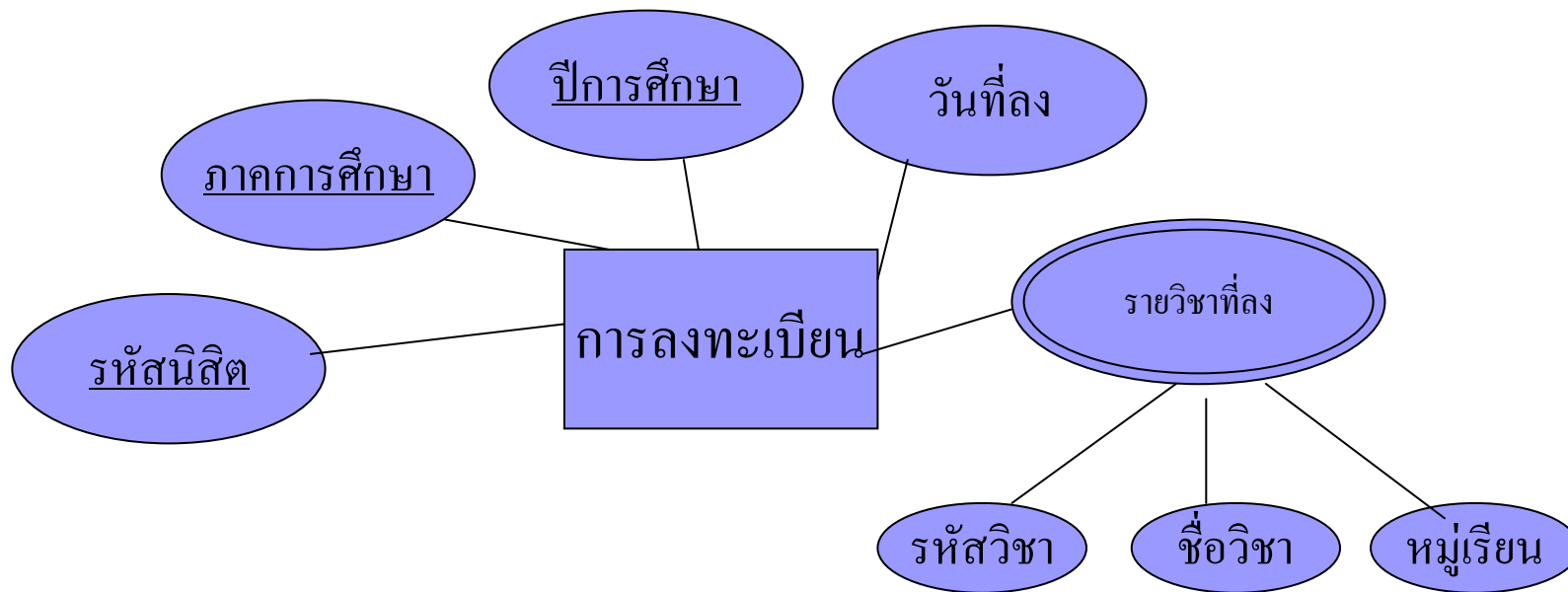
Entities ที่วิเคราะห์ได้



Entities ที่วิเคราะห์ได้



Entities ที่วิเคราะห์ได้

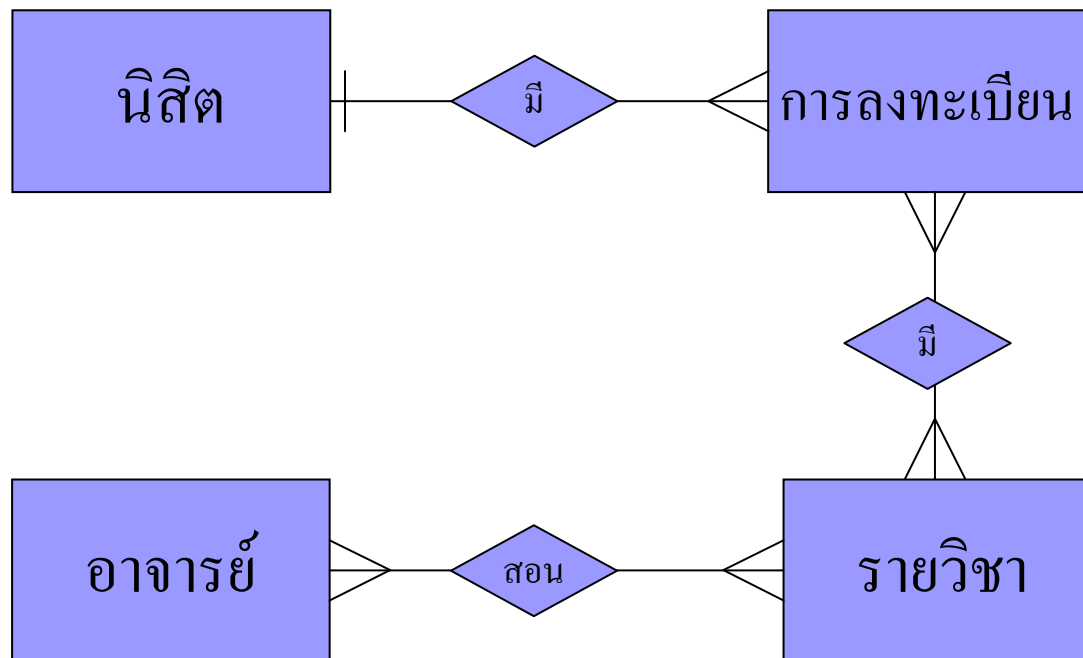




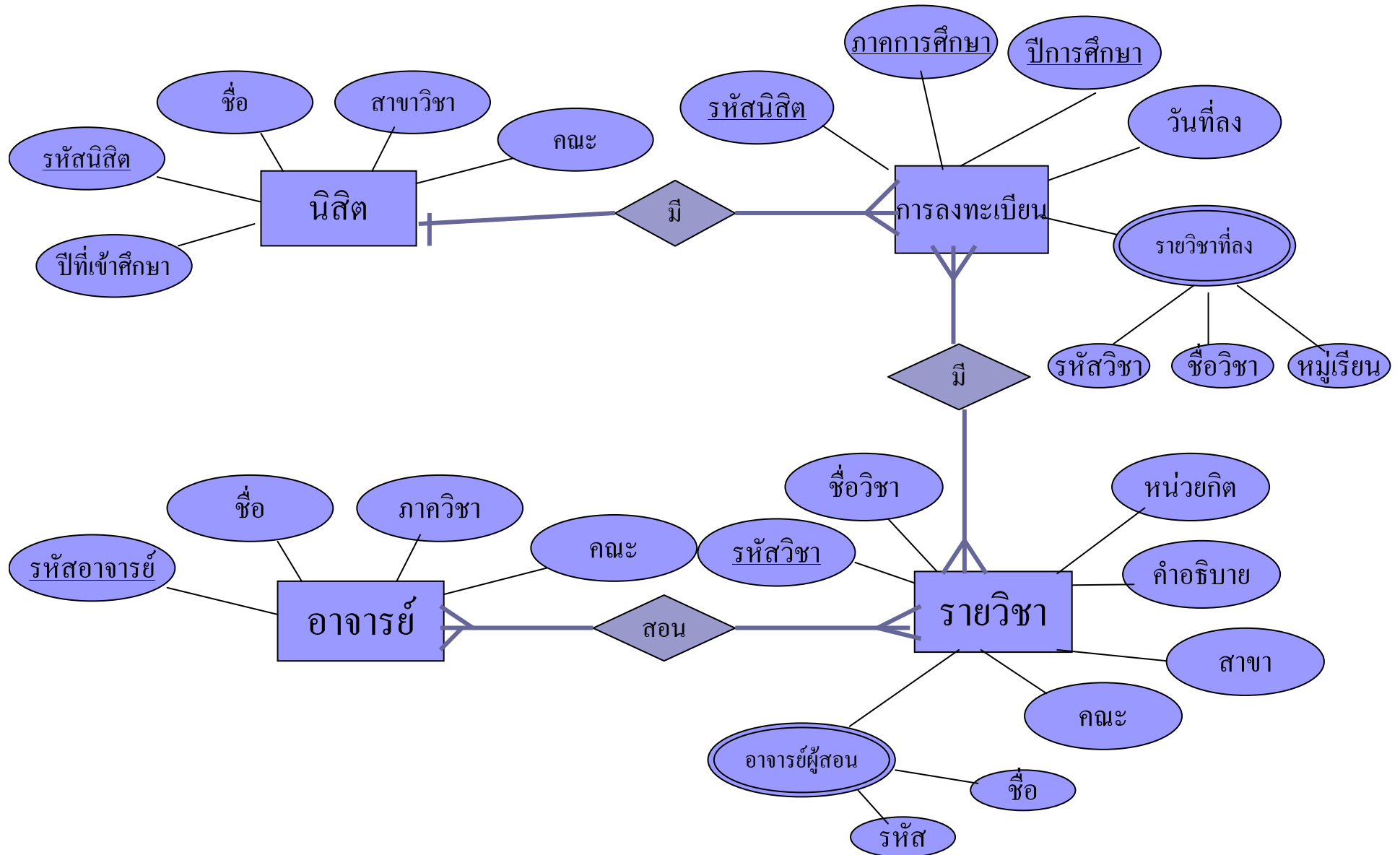
Business rules

- นิสิตสามารถทำการลงทะเบียนได้หลายภาคการศึกษา
- ลงทะเบียนแต่ละครั้งสามารถลงได้หลายวิชา
- แต่ละรายวิชาสามารถลงได้หลายครั้ง
- อาจารย์สามารถสอนได้หลายวิชา
- แต่ละวิชาสามารถมีอาจารย์สอนได้มากกว่า 1 ท่าน
- ฯลฯ

วิเคราะห์ Relationship



รายละเอียด Data Model





Note

- ยังไม่ต้องนึกถึงตารางฐานข้อมูล
- ยังไม่ต้องนึกถึงการ **Normalization**
- ตอนนี้แค่เข้าใจความหมายของตัวข้อมูลแต่ละตัว
และความสัมพันธ์ของข้อมูลแต่ละตัวก็พอ