

## โมเดลเชิงสัมพันธ์ (ER-Model)

204204 Database Design & Development

## วัตถุประสงค์รายครั้ง

นักศึกษาสามารถ

- ออกแบบและเขียน ER-Diagram ได้อย่างถูกต้อง
- แปลง ER-Diagram ให้เป็น Sub Schemas ได้

## หัวข้อการบรรยาย

- เอนทิตี แอททริบิวต์และความสัมพันธ์
- Weak Entity types
- คีย์ (key)
- โมเดลข้อมูลเชิงสัมพันธ์ (Entity\_Relationship Model)
- คุณลักษณะของ ER Diagram
- ขั้นตอนการ ออกแบบ ER Model
- การเขียน ER Diagram
- การแปลง ER-Diagram ให้เป็น SubSchema

## เอนทิตี แอททริบิวต์และความสัมพันธ์

- เอนทิตี (Entity) หมายถึงสิ่งต่างๆที่เราเกี่ยวข้องกับฐานข้อมูลนั้นๆ ซึ่งอาจจะเป็นรูปธรรม หรือ นามธรรมก็ได้
- แอททริบิวต์ (Attribute) จะเป็นส่วนองค์ประกอบที่แสดงคุณสมบัติของเอนทิตี
- ความสัมพันธ์
  - One-To-One
  - One-To-Many
  - Many-To-Many

## Weak Entity types

- บาง Entity อาจจะไม่มีความเฉพาะ ของตัวเองเลยนั้น หมายถึง เราอาจจะ ไม่สามารถหาความมีตัวตนของ Entity ได้ (Value ของ Attribute ไม่มีเอกลักษณ์เฉพาะของตัวเอง) เราเรียก Entity แบบนี้ว่า Weak Entity types
- Entity ที่เราเรียกว่า Weak Entity type จะถูกกำหนดโดยอีก Entity หนึ่งโดยใช้ผลรวมของบาง Attribute value ของ Entity นี้ เราเรียก Entity นี้ว่า Identifying owner และเรียก ความสัมพันธ์แบบนี้ว่า Identifying relationship

## คีย์ (key)

- **Super Key** คือ เซตของแอตทริบิวต์ที่สามารถแทน Entity
- **Candidate Key** คือ Super Key ที่ ไม่มี ซับเซตของ ตัวเอง ที่เป็น Super Key
- ตัวอย่าง : STUDENT(ID, LAST, FIRST, TEL#, ADDRESS) จะมี
  - { ID, Tel# } คือ SuperKey
  - { LAST, FIRST } คือ SuperKey
  - { LAST, FIRST, ID } คือ SuperKey
  - แต่ {ID} คือ SuperKey และ Candidate Key

## โมเดลข้อมูลเชิงสัมพันธ์ (Entity-Relationship Model)

- **เอนทิตี** คือสิ่งที่มีอยู่จริง จับต้องได้ หรือเป็นจินตภาพที่แสดงความเป็นหนึ่งเดียว เมื่อกล่าวถึงแล้วทุกคนเข้าใจตรงกัน เช่น
- **รีเลชันชิป** คือความสัมพันธ์ซึ่งเป็นลักษณะการเกี่ยวพันกันระหว่างเอนทิตีหนึ่งกับตัวมันเองหรือ เอนทิตีอื่นอาจเป็นความสัมพันธ์ที่มากกว่า 2 เอนทิตีก็ได้
- **แอตทริบิวต์** คือกลุ่มของค่าความจริงใด ๆ ที่เป็นรายละเอียดของเอนทิตีซึ่งแสดงลักษณะ และคุณสมบัติของเอนทิตี ทำให้เข้าใจเอนทิตีได้ลึกซึ้งยิ่งขึ้น และเป็นสิ่งที่ไม่สามารถแยกย่อยออกไปได้อีกโดยไม่เสียความหมายไป

## คุณลักษณะของ ER Diagram

- แสดงได้ด้วยแผนภาพ (Graphical Diagrams)
- แสดงชัดเจนถึงความหมายของข้อมูล (Explicit Representation of Semantic)
- แสดงรายละเอียดในระดับที่เหมาะสม (Appropriate Level of Detail)
- ไม่พึ่งพิงกับระบบจัดการฐานข้อมูลแบบใดแบบหนึ่ง (DBMS Independence)
- ง่ายต่อการศึกษาและใช้งาน

## ขั้นตอนการ ออกแบบ ER Model

- ขั้นตอนที่ 1 การกำหนดเอนทิตีหลัก
- ขั้นตอนที่ 2 การกำหนดความสัมพันธ์ ระหว่างเอนทิตี
- ขั้นตอนที่ 3 การกำหนดคีย์หลักและคีย์รอง
- ขั้นตอนที่ 4 การกำหนดคีย์ภายนอก
- ขั้นตอนที่ 5 พิจารณาขอบเขตค่าโดเมนของแอตทริบิวต์
- ขั้นตอนที่ 6 การวิเคราะห์ความมีเสถียรภาพและการเติบโตในอนาคต

## การเขียน ER Diagram

- Data model หมายถึงกลุ่มของแนวคิดที่ช่วยเรากำหนดโครงสร้างของฐานข้อมูลและ ชุดเซตที่เกี่ยวข้องกับชุดคำสั่งที่ใช้ในการเรียกใช้ และ เปลี่ยนแปลง ข้อมูล
- แบบจำลอง Entity Relation (ER Model) สามารถอธิบายภาพรวม (Data View) ของทั้งองค์กรได้ดีในรูปแบบ ER Diagram
- แบบจำลองแบบ E-R Model ถูกออกแบบให้ง่ายต่อความเข้าใจความเข้าใจของผู้ใช้งาน โดยไม่คำนึงถึงลักษณะการเก็บทางกายภาพของข้อมูลว่าเก็บอย่างไรก็ได้

- หลังจากที่ได้ความสัมพันธ์ทั้งหมดแล้ว พยายามจัดความซับซ้อนของ Attribute ในแต่ละ Entity ให้มากที่สุดเพื่อเขียนเป็น Conceptual Schema
- ใช้สัญลักษณ์ในการเขียน ER Diagram เพื่ออธิบาย Conceptual Schema ให้ง่ายต่อการเข้าใจและตีความ

## การแปลง ER-Diagram ให้เป็น SubSchema

- ขั้นตอนที่ 1 ทุก ๆ Entity type E ใน ER Schema เราจะสร้าง Relation R ที่รวบรวม Attribute ของ E นั้นเข้าไปในรูปแบบสมการแล้วเลือกเอาหนึ่งใน Attribute ของ R เป็น Key เพื่อใช้เป็นตัวแทนและง่ายต่อการค้นหา โดยใช้สัญลักษณ์ขีดเส้นใต้ และเป็นตัวหนา ส่วนคีย์ภายนอก (Foreign Key) คือ สัญลักษณ์ตัวเอียง
- ขั้นตอนที่ 2 สำหรับ weak entity type ใน ER schema เราสร้าง relation R และเอา primary key attribute ของ owner entity W เพื่อเป็นความสัมพันธ์ระหว่างกัน

- ขั้นตอนที่ 3 ถ้าความสัมพันธ์ 1:1 ให้เลือกเอา Primary key จากหนึ่ง entity ทั้งสองมาเป็น Foreign key ของอีก Entity หนึ่ง