

# **DATABASE**

Noppanut Ploywong  
TO3EIT 60 KMITL

## Introduction

นับวันโลกของเรายังมีข้อมูลเป็นจำนวนมากขึ้นเรื่อย ๆ หากจะเก็บข้อมูลเป็นเอกสารเหมือนสมัยก่อน ก็คงยาก จึงมีสิ่งที่เรียกว่า “**ฐานข้อมูล**” (Database) เข้ามาช่วยให้การจัดการกับข้อมูลเหล่านั้น สะดวกและเป็นระบบมากขึ้น โดยเนื้อหาการสอนจะครอบคลุมทฤษฎีพื้นฐานของฐานข้อมูล แฟ้มข้อมูล การออกแบบฐานข้อมูล ไปจนถึงซอฟต์แวร์ต่าง ๆ ที่ใช้ในการจัดการฐานข้อมูล

## ภาพรวมของเนื้อหา (Course Overview)

- ข้อมูล (Data) & สารสนเทศ (Information)
- การจัดโครงสร้างของแฟ้มข้อมูล (File Organization)
  - แบบเรียงลำดับ (Sequential)
  - แบบสุ่ม (Random)
  - แบบลำดับเชิงตรรกะ (Index sequential)
- ลำดับชั้นของข้อมูล (Data hierarchy)
  - บิต (Bit)
  - ไบต์ (Byte)
  - ฟิลด์ (Field)
  - เรคคอร์ด (Record)
  - ไฟล์ (File)
- ระบบฐานข้อมูล (Database System)
  - โครงสร้างแบบตามลำดับชั้น (Hierarchical Model)
  - โครงสร้างแบบเครือข่าย (Network Model)
  - โครงสร้างแบบเชิงสัมพันธ์ (Relational Model)
- แบบจำลองโครงสร้างของฐานข้อมูล (ER-Diagram)
  - เอนทิตี (Entity)
  - แอททริบิว (Attribute)
  - ความสัมพันธ์ (Relationship)
- ระบบจัดการฐานข้อมูล (Database Management System : DBMS)
  - SQL
  - NoSQL

## ข้อมูล (Data)

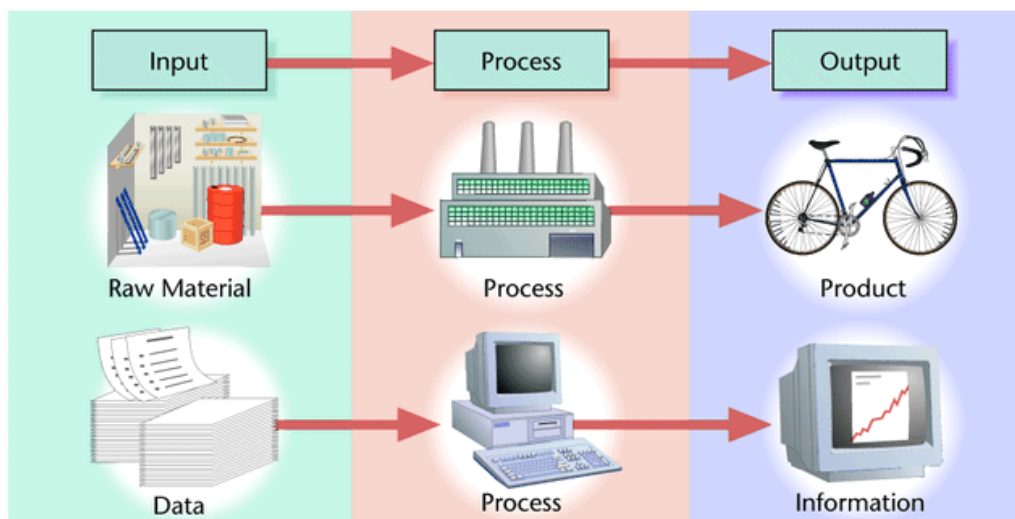
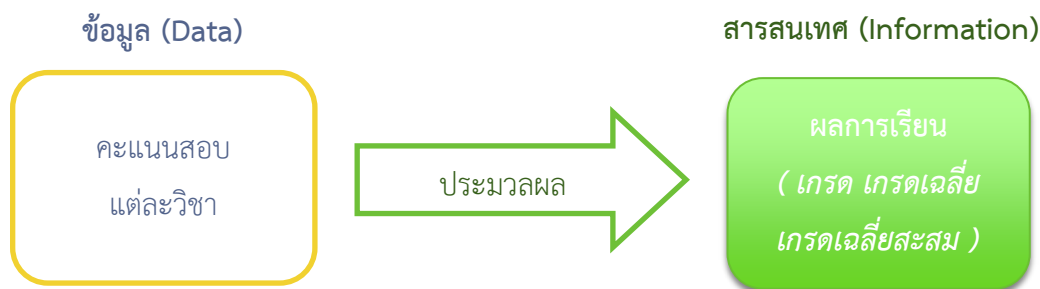
ข้อมูล หมายถึง ข่าวดสาร เอกสาร ข้อเท็จจริงเกี่ยวกับบุคคล สิ่งของหรือเหตุการณ์ที่มีอยู่ในรูปของ **ตัวเลข อักขระ ภาพ เสียง** ที่มีความหมายเฉพาะตัว ซึ่งยังไม่ผ่านการประมวลผลใด ๆ เช่น อายุ น้ำหนัก ส่วนสูง หมายเลข โทรศัพท์ เลขประจำตัวประชาชน เป็นต้น

### คุณสมบัติของข้อมูลที่ดี

1. ถูกต้องแม่นยำ (Accuracy)
2. มีความเป็นปัจจุบัน (Update)
3. สมบูรณ์ครบถ้วน (Complete)
4. ตรงกับความต้องการของผู้ใช้ (Relevance)
5. สามารถตรวจสอบได้ (Verifiable)

## สารสนเทศ (Information)

สารสนเทศ (Information) หมายถึง ข้อมูลที่ผ่านการ **จัดเรียง วิเคราะห์ คำนวณ ประมวลผล** แล้ว ซึ่งสารสนเทศนั้นมีคุณค่า มีสาระ นำไปใช้งานได้



## การจัดโครงสร้างของแฟ้มข้อมูล (File Organization)

โครงสร้างแฟ้ม	ข้อดี	ข้อเสีย	สื่อที่ใช้จัดเก็บ
แบบเรียงลำดับ (Sequential)	<ul style="list-style-type: none"> <li>- เสียค่าใช้จ่ายน้อยและใช้งานได้ง่ายกว่าวิธีอื่น ๆ</li> <li>- เหมาะกับงานประมวลผลที่มีการอ่านข้อมูลแบบเรียงลำดับและมีข้อมูลในปริมาณมาก</li> <li>- สื่อที่ใช้เก็บเป็นเทปซึ่งมีราคาถูก</li> </ul>	<ul style="list-style-type: none"> <li>- การทำงานเพื่อค้นหาข้อมูลจะต้อง เริ่มทำตั้งแต่ต้นไฟล์เรียงลำดับไปเรื่อย จนกว่าจะหาข้อมูลนั้นเจอ ทำให้เสียเวลาค่อนข้างมาก</li> <li>- ข้อมูลที่ใช้ต้องมี การจัดเรียงลำดับก่อนเสมอ</li> <li>- ไม่เหมาะกับงานที่ต้อง แก้ไขเพิ่ม ลบข้อมูลเป็นประจำเช่น งานธุรกรรมออนไลน์</li> </ul>	เทปแม่เหล็ก เช่น เทปคาสเซต
แบบสุ่ม (Random)	<ul style="list-style-type: none"> <li>- สามารถทำงานได้เร็ว มีการเข้าถึงข้อมูลเรคคอร์ดแบบเร็วมาก เพราะไม่ต้องเรียงลำดับข้อมูลก่อนเก็บลงไฟล์</li> <li>- เหมาะสมกับการใช้งานธุรกรรมออนไลน์ หรืองานที่ต้องการแก้ไขเพิ่ม ลบรายการเป็นประจำ</li> </ul>	<ul style="list-style-type: none"> <li>- ไม่เหมาะกับงานประมวลผลที่อ่านข้อมูลในปริมาณมาก</li> <li>- การเขียนโปรแกรมเพื่อค้นหาข้อมูลจะซับซ้อน</li> <li>- ไม่สามารถเข้าถึงข้อมูลแบบเรียงลำดับได้</li> </ul>	จานแม่เหล็ก เช่น ดิสเก็ตต์ ฮาร์ดดิสก์ CD-ROM
แบบลำดับเชิงตรรกะ (Index sequential)	<ul style="list-style-type: none"> <li>- สามารถรองรับการประมวลผลได้ทั้ง 2 แบบคือ แบบเรียงลำดับและแบบสุ่ม</li> <li>- เหมาะสมกับงานที่ต้องการแก้ไขเพิ่ม ลบรายการเป็นประจำ</li> </ul>	<ul style="list-style-type: none"> <li>- สิ้นเปลืองเนื้อที่ในการจัดเก็บตรรกะที่ใช้อ้างอิงถึงตำแหน่งของข้อมูล</li> <li>- การเขียนโปรแกรมเพื่อ ค้นหาข้อมูล จะซับซ้อน</li> <li>- การทำงานช้ากว่าแบบสุ่ม และมีค่าใช้จ่ายสูง</li> </ul>	จานแม่เหล็ก เช่น ดิสเก็ตต์ ฮาร์ดดิสก์ CD-ROM

## ลำดับชั้นของข้อมูล (Data hierarchy)

### 1. บิต (Bit : Binary Digit)

หน่วยเก็บข้อมูลที่เล็กที่สุด เป็นสัญญาณดิจิทัล ซึ่งประกอบด้วย สัญญาณไฟฟ้า 2 สถานะ ได้แก่ 1 กับ 0 หรือ ปิดกับเปิด หรือ จริงกับเท็จ

### 2. ไบต์ (Byte)

การนำค่าบิตจำนวน 8 บิตมาเรียงต่อกันจะแทนค่าตัวอักษรได้ 1 ตัวขึ้นไป  
เช่น 01000001 แทนตัวอักษร "A"

8 bit = 1 byte = 1 character

### 3. ฟیلด์ (Field)

ขอบเขตของข้อมูล ซึ่งประกอบด้วยตัวอักษร (Byte) ตั้งแต่ 1 ตัวขึ้นไป

### 4. เรคคอร์ด (Record)

ชุดข้อมูล 1 ชุด ประกอบด้วยหลาย ๆ ฟیلด์ ที่มีความสัมพันธ์กัน

### 5. ไฟล์ หรือ แฟ้มตารางข้อมูล (File)

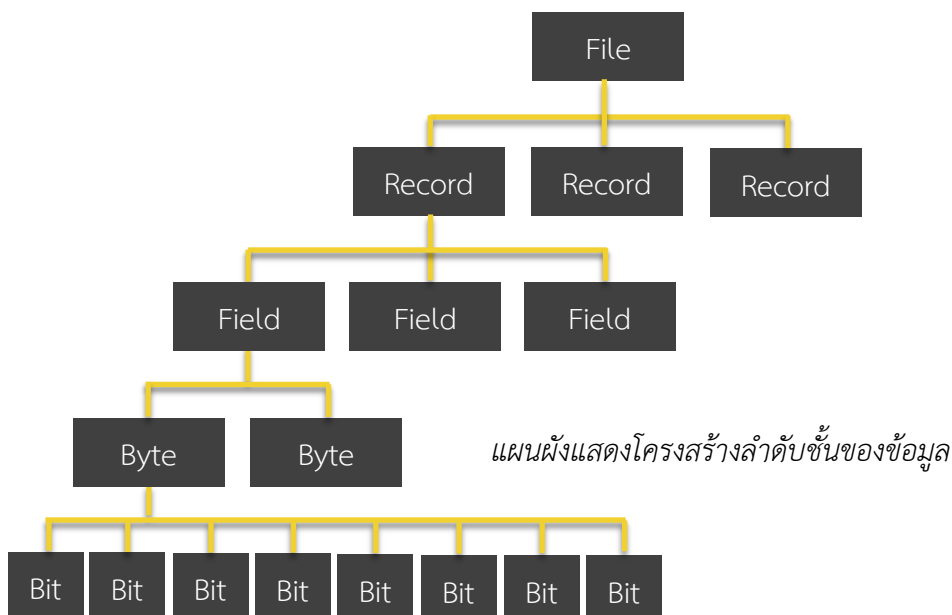
การนำเอาข้อมูลหลาย ๆ เรคคอร์ดมารวมกัน จะกลายเป็นแฟ้มข้อมูล

Fields

File — **STUDENT**

Student ID	Firstname	Lastname	Phone Number
10001	วีรภัทร	ทรัพย์สมบูรณ์	0834567890
10002	นพณัฐ	พลอยวงศ์	0812345467
10003	สัทยา	สิงห์กุล	0823456789
10004	ภูภูมิ	ทองอยู่	0845678901

Record



## ระบบฐานข้อมูล (Database System)

**ฐานข้อมูล (Database)** เกิดจากการรวบรวมเอาไฟล์ หรือแฟ้มตารางข้อมูลหลาย ๆ แฟ้มที่มีความสัมพันธ์กันมาเก็บรวบรวมกันไว้ที่เดียว

### ประโยชน์ของระบบฐานข้อมูล

#### 1. ลดการเก็บข้อมูลที่ซ้ำซ้อน

ในระบบฐานข้อมูล การเรียก เพิ่มเติม แก้ไขข้อมูล จะทำในระบบเดียวกัน ทำให้ลดความซ้ำซ้อนของข้อมูลได้

#### 2. รักษาความถูกต้องของข้อมูล

เนื่องจากข้อมูลรวมอยู่ในฐานข้อมูลเดียว เมื่อต้องการแก้ไขจะทำให้ข้อมูลทั้งหมดถูกแก้ไขตามไปด้วย

#### 3. ข้อมูลมีความปลอดภัย

ภายในระบบฐานข้อมูล จะมีการกำหนดสิทธิในการเข้าถึง ให้เฉพาะผู้ที่เกี่ยวข้องเท่านั้น เช่น ต้องยืนยันตัวตนด้วย Username และ Password ก่อนแก้ไขข้อมูล

### ประเภทของฐานข้อมูล

1. โครงสร้างแบบตามลำดับชั้น (Hierarchical Model)
2. โครงสร้างแบบเครือข่าย (Network Model)
3. โครงสร้างแบบเชิงสัมพันธ์ (Relational Model)

## แบบจำลองโครงสร้างของฐานข้อมูล (ER-Diagram)

ER-Diagram (Entity Relationship Diagram) คือ แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของแผนภาพ อธิบายโครงสร้างและความสัมพันธ์ของข้อมูล (Relationship) ประกอบด้วย เอนทิตี (Entity), แอททริบิว (Attribute) และความสัมพันธ์ (Relationship)

### 1. เอนทิตี (Entity)

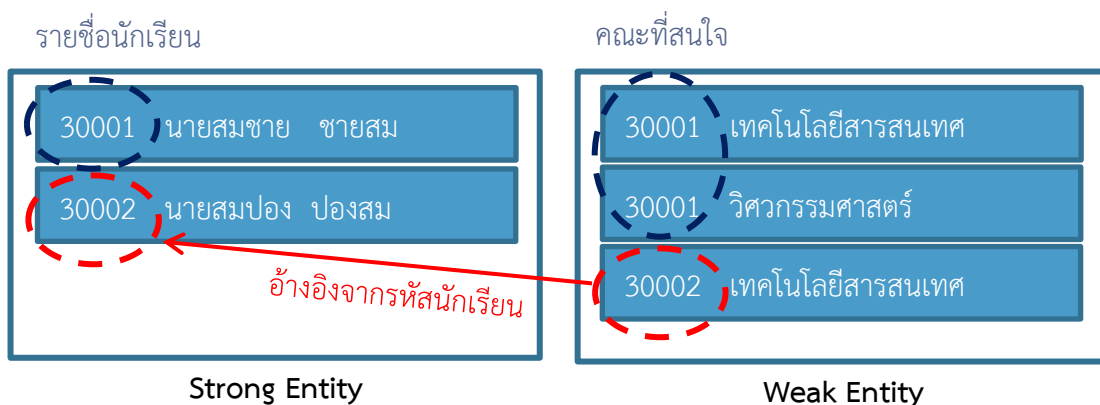
เอนทิตีคือสิ่งที่เราสนใจ เป็นได้ทั้งบุคคล สถานที่ วัตถุ เหตุการณ์ หรือแนวคิดต่าง ๆ ที่เป็น “คำนาม” และก่อให้เกิดกลุ่มของข้อมูลที่ต้องการ แบ่งออกเป็น 2 ชนิด ได้แก่

**1.1 Strong Entity** คือ เอนทิตีที่ทั่ว ๆ ไปที่มีคุณสมบัติในตัวเอง และไม่ขึ้นกับเอนทิตีใด ทำให้สามารถระบุความแตกต่างของข้อมูลได้ด้วยตนเอง

สัญลักษณ์ : สีเหลี่ยมผืนผ้า

**1.2 Weak Entity** คือ เอนทิตีที่ต้องอาศัยแอททริบิวบนเอนทิตีอื่น ๆ เพื่อระบุความแตกต่างของข้อมูล (ข้อมูลมักจะซ้ำ)

สัญลักษณ์ : สีเหลี่ยมผืนผ้าเส้นคู่



จากรูปแบบของข้อมูลด้านบน สามารถนำมาเขียนเป็นสัญลักษณ์ ได้เป็น



## 2. แอททริบิว (Attribute)

แอททริบิวคือคุณสมบัติต่าง ๆ ของเอนทิตี สัญลักษณ์คือรูปวงรี

### 2.1 Simple Attribute และ Composite Attribute

- **Simple Attribute** (Atomic Attribute) คือ แอททริบิวที่มีองค์ประกอบเดียว ไม่สามารถแตกเป็นข้อมูลย่อยได้อีก เช่น เพศ, อายุ, เงินเดือน เป็นต้น  
สัญลักษณ์ : วงรี
- **Composite Attribute** คือ แอททริบิวที่มีข้อมูลย่อยหลายตัว ภายในแอททริบิวนั้น เช่น แอททริบิว “ที่อยู่บ้าน” ที่ภายในประกอบไปด้วยแอททริบิว เลขที่, ถนน, ตำบล, อำเภอ, จังหวัด เป็นต้น  
สัญลักษณ์ : วงรีลากเชื่อมไปยังแอททริบิวหลัก

### 2.2 Single-Valued Attribute และ Multivalued Attribute

- **Single-Valued Attribute** คือ แอททริบิวที่บรรจุเพียงค่าเดียว เช่น เพศ, ชื่อ, สกุล, รหัสพนักงาน เป็นต้น  
สัญลักษณ์ : วงรี
- **Multivalued Attribute** คือ แอททริบิวที่สามารถมีหลายค่าได้ เช่น วุฒิกการศึกษา (พนักงานคนหนึ่งสามารถมีวุฒิกการศึกษาได้หลายระดับ)  
สัญลักษณ์ : วงรีเส้นคู่

### 2.3 Derived Attribute

แอททริบิวที่เกิดจากการคำนวณ เช่น แอททริบิว “อายุ” ที่เกิดจากการคำนวณ โดยนำปีปัจจุบันมาหักลบกับปีเกิด  
สัญลักษณ์ : วงรีเส้นประ

### 2.4 Primary Key (Key Attribute)

แอททริบิวที่ถูกเลือกมาโดยมีข้อมูลไม่ซ้ำกันเลย เอาไว้ใช้อ้างอิงในแต่ละเรคคอร์ด เช่น รหัสนักเรียน, เลขบัตรประชาชน เป็นต้น  
สัญลักษณ์ : แอททริบิวที่ถูกขีดเส้นใต้

### 2.5 Foreign Key

Primary Key ของอีกเอนทิตีหนึ่ง ที่ถูกนำมาอ้างในอีกเอนทิตีหนึ่ง ยังคงเป็น Primary Key เหมือนเดิม เพียงเรียกชื่อเปลี่ยนไป

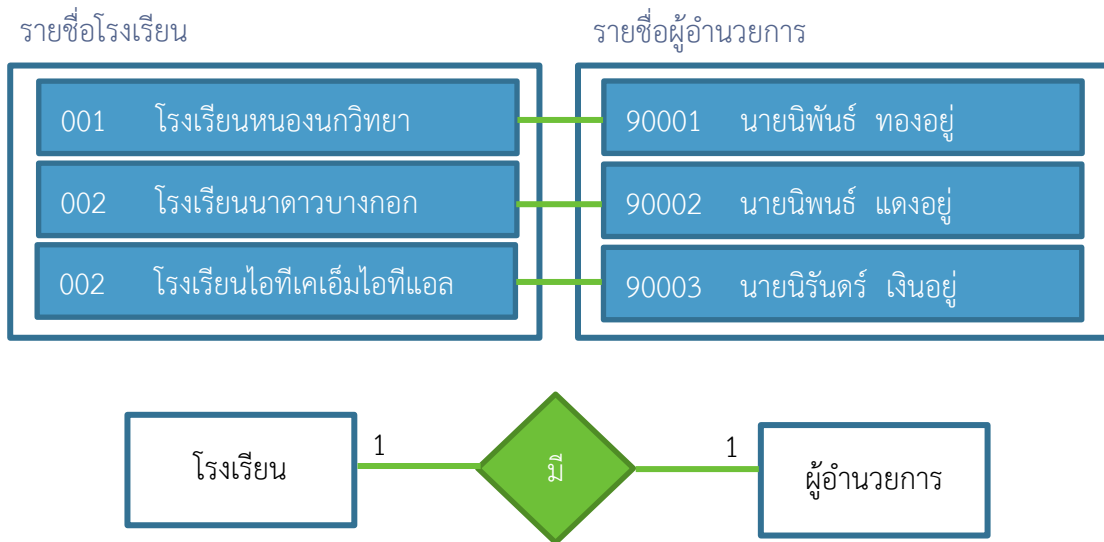


### 3. ความสัมพันธ์ (Relationship)

Relationship คือความสัมพันธ์ระหว่างเอนทิตีที่ มีความสัมพันธ์ของข้อมูลกันอย่างไร ใช้สัญลักษณ์สี่เหลี่ยมข้าวหลามตัด และอ่านจากซ้ายไปขวา

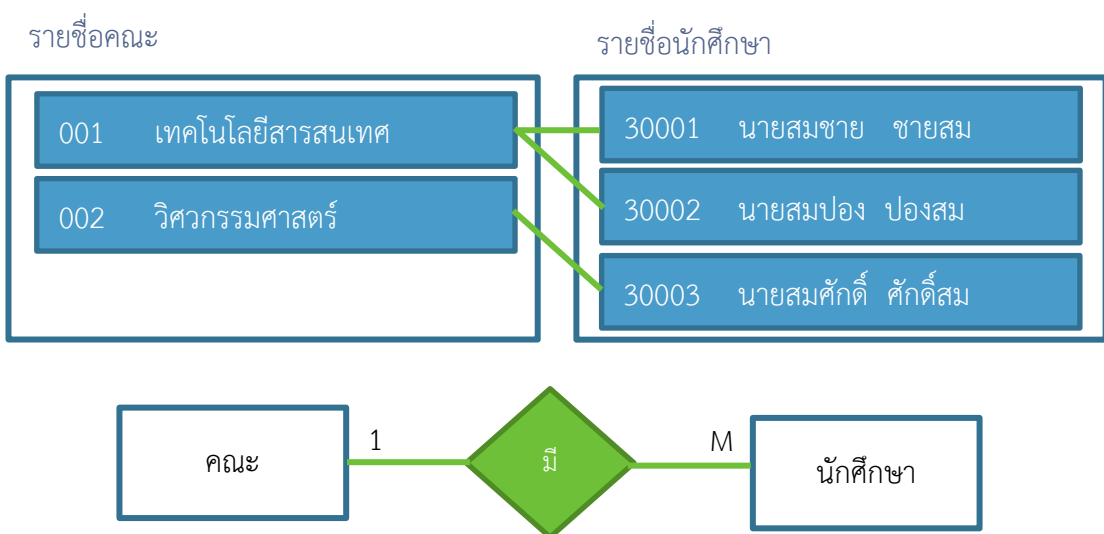
#### 3.1 One to One

ความสัมพันธ์แบบหนึ่งต่อหนึ่ง เป็นความสัมพันธ์ระหว่างสิ่งหนึ่งกับอีกสิ่งหนึ่งที่มีเพียงหนึ่งเดียว เช่น โรงเรียนแต่ละโรงเรียน จะมีผู้อำนวยการได้เพียงคนเดียวเท่านั้น



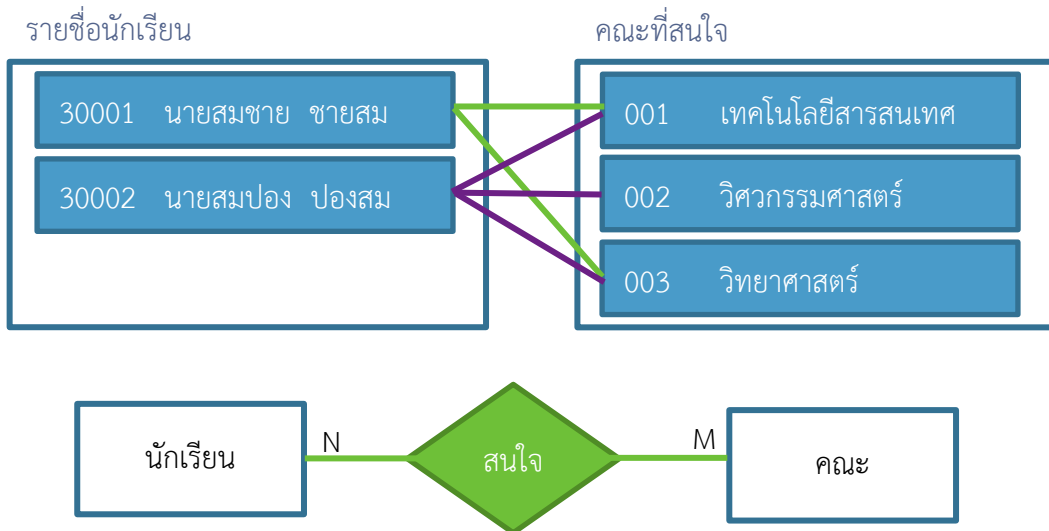
#### 3.2 One to Many

ความสัมพันธ์แบบหนึ่งต่อกลุ่ม เป็นความสัมพันธ์ระหว่างสิ่งหนึ่งกับอีกด้านหนึ่งที่เป็นกลุ่ม เช่น ในแต่ละคณะ จะมีนักศึกษาอยู่หลายคน



### 3.3 Many to Many

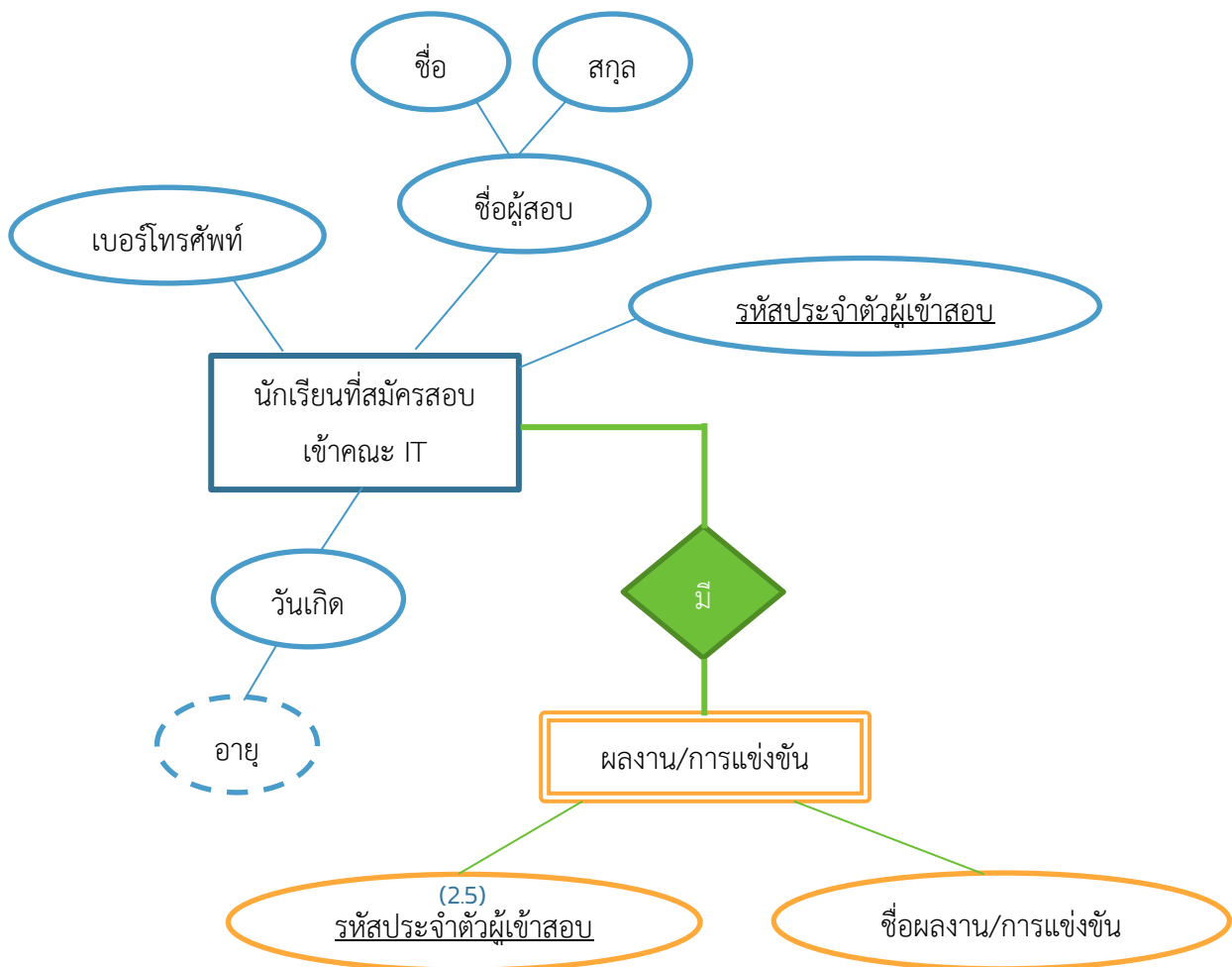
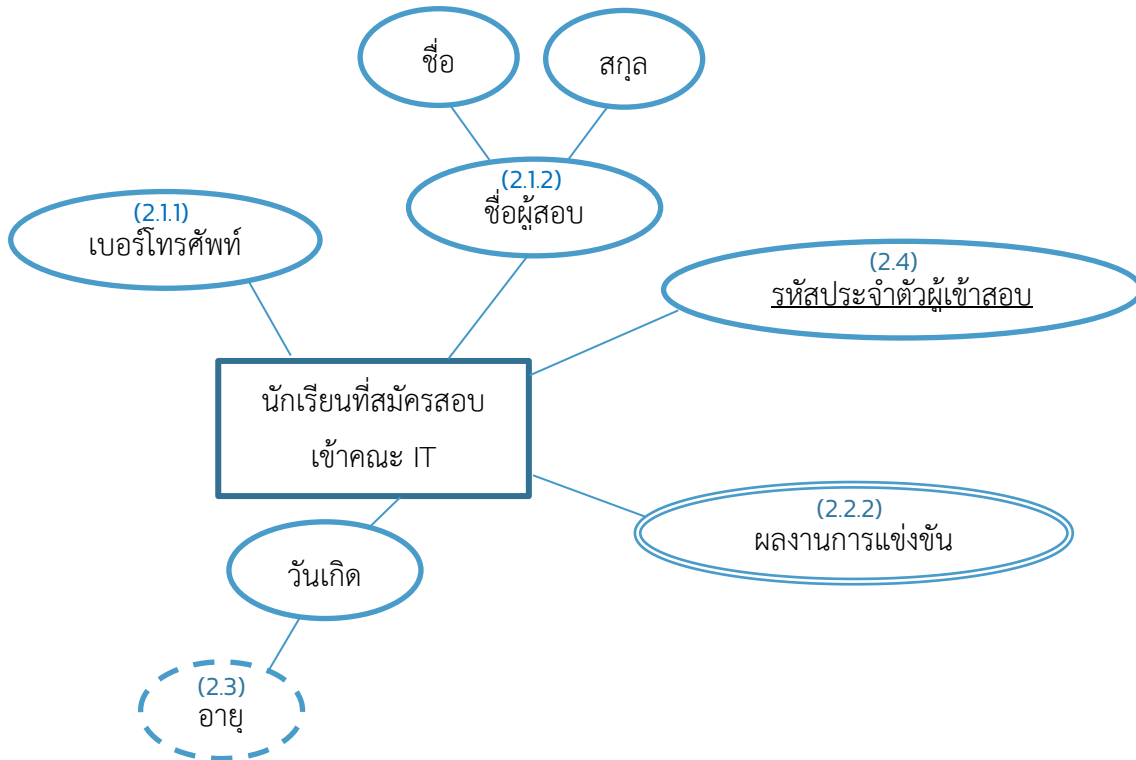
ความสัมพันธ์แบบกลุ่มต่อกลุ่ม เป็นความสัมพันธ์ระหว่างสิ่งหนึ่งกับสิ่งหนึ่งที่มีได้หลาย ๆ อย่าง เช่น นักเรียน 1 คนมีคณะที่ตนสนใจอยู่หลายคณะ และในแต่ละคณะก็มีนักเรียนสนใจอยู่หลายคน



#### ขั้นตอนการเขียน ER-Diagram

1. หาเอนทิตีจากผู้ใช้ระบบว่าต้องการเก็บข้อมูลอะไร
2. กำหนดแอททริบิวให้เอนทิตีแต่ละตัวเท่าที่จำเป็น โดยกำหนดแอททริบิว 1 ตัวเป็น Primary key โดยดูจากคุณสมบัติของแอททริบิว
3. พิจารณารูปแบบความสัมพันธ์ระหว่างเอนทิตี

## ตัวอย่าง ER-Diagram



## ระบบจัดการฐานข้อมูล (Database Management System : DBMS)

ระบบจัดการฐานข้อมูล (DBMS) คือซอฟต์แวร์จัดการฐานข้อมูลที่นำมาใช้เป็นเครื่องมือเพื่อให้ผู้ใช้สามารถโต้ตอบกับฐานข้อมูลได้ โดยพื้นฐาน DBMS ถูกออกแบบมาให้สร้าง แก้ไข เพิ่มเติม ค้นหา เรียกใช้ข้อมูลได้ โดย DBMS ที่รู้จักกันทั่วไป ได้แก่ MySQL, PostgreSQL, Microsoft Access, Oracle และ MongoDB เป็นต้น



### หน้าที่ของระบบจัดการฐานข้อมูล

1. แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูลให้อยู่ในรูปแบบที่เข้าใจ และนำคำสั่ง ต่าง ๆ ไปใช้งาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) และการเพิ่มข้อมูล (Add) เป็นต้น
2. ป้องกันความเสียหาย (มีฟังก์ชันการสำรองและกู้คืนข้อมูล)
3. ควบคุมให้ฐานข้อมูลทำงานได้อย่างถูกต้องและมีประสิทธิภาพ

## SQL



SQL ย่อมาจาก Structured Query Language คือภาษาที่ใช้ในการเขียนโปรแกรม เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน มีประสิทธิภาพการทำงานสูง สามารถทำงานที่ซับซ้อนได้โดยใช้คำสั่งเพียงไม่กี่คำสั่ง ใช้กับระบบฐานข้อมูลเชิงสัมพันธ์

### ประเภทของคำสั่งภาษา SQL

1. ภาษานิยามข้อมูล (Data Definition Language : DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูล ชนิดของข้อมูล รวมทั้งการเปลี่ยนแปลงตาราง และการสร้างดัชนี
  - CREATE สร้างตารางข้อมูล
  - DROP ลบตารางข้อมูล
  - ALTER แก้ไขตารางข้อมูล
2. ภาษาจัดการข้อมูล (Data Manipulation Language : DML) เป็นคำสั่งที่ใช้ในการเรียกใช้ เพิ่ม ลบ และเปลี่ยนแปลงข้อมูลในตาราง
  - SELECT เลือกข้อมูล
  - INSERT เพิ่มข้อมูล
  - UPDATE แก้ไขข้อมูล
  - DELETE ลบข้อมูล
3. ภาษาควบคุมข้อมูล (Data Control Language : DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิการอนุญาต หรือยกเลิก การเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของฐานข้อมูล
  - GRANT กำหนดสิทธิการเข้าถึงข้อมูล
  - REVOKE ยกเลิกสิทธิการเข้าถึงข้อมูล

### NoSQL



NoSQL (Not Only SQL) คือ แนวทางหนึ่งสำหรับการจัดการข้อมูล และออกแบบฐานข้อมูลสำหรับข้อมูลขนาดใหญ่ โดยเป็นระบบจัดการฐานข้อมูลแบบไม่ใช้ SQL

NoSQL ของแต่ละยี่ห้อ จะมีการพัฒนาภาษาสำหรับใช้งานเป็นของตนเอง ซึ่ง NoSQL จะมีจุดมุ่งหมายโดยเน้นความเร็วในการทำงานเป็นหลัก จะไม่เน้นในการสร้างความสัมพันธ์ของข้อมูล



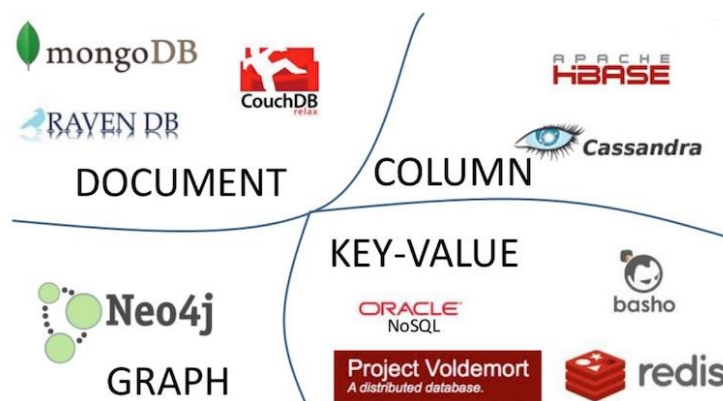
ระบบฐานข้อมูลแบบ NoSQL มักถูกนำไปใช้งานกับโครงสร้างฐานข้อมูลที่มีขนาดใหญ่ หรือต้องรองรับการขยายตัวของข้อมูลมหาศาล เช่น Pantip, Google, EA, eBay และ Facebook เป็นต้น

#### ข้อดีของ NoSQL

1. สามารถขยายระบบได้ง่าย
2. รองรับข้อมูลขนาดใหญ่
3. รองรับรูปแบบข้อมูลที่หลากหลาย หรือมีความยืดหยุ่นสูงได้

#### การจัดหมวดหมู่ของ NoSQL ประกอบไปด้วย 4 หมวด

1. Column-based databases คือฐานข้อมูลที่เก็บข้อมูลในรูปของตาราง คล้าย ๆ กับฐานข้อมูลเชิงสัมพันธ์ แต่จะเก็บข้อมูลในรูปแบบ คอลัมน์แทนแบบแถว ตัวอย่างของซอฟต์แวร์ฐานข้อมูลที่เป็นลักษณะนี้ ได้แก่ Hbase, Cassandra, Hypertable เป็นต้น
2. Document-based databases คือฐานข้อมูลที่เก็บข้อมูลในรูปแบบกลุ่มของเอกสารซึ่งง่ายต่อการแก้ไข โครงสร้างโดยแต่ละเอกสารจะมี จำนวนฟิลด์ที่แตกต่างกันได้ ตัวอย่างของซอฟต์แวร์ฐานข้อมูลที่เป็นลักษณะนี้ ได้แก่ CouchDB, MongoDB เป็นต้น
3. Key-value data stores คือฐานข้อมูลที่เก็บข้อมูลในรูปแบบ Key และ Value ที่สัมพันธ์กันอยู่ ซึ่ง Value จะถูกค้นเจอได้โดยค่า Key ตัวอย่างของซอฟต์แวร์ฐานข้อมูลที่เป็นลักษณะนี้ ได้แก่ Redis, Voldemort เป็นต้น
4. Graph-based data-stores คือฐานข้อมูลที่เก็บข้อมูลในรูปแบบของ กราฟที่มีความสัมพันธ์กัน ทำให้สามารถใช้ทฤษฎีกราฟได้ในการจัดการกับข้อมูลได้เป็นอย่างดี ตัวอย่างของซอฟต์แวร์ฐานข้อมูลที่เป็นลักษณะนี้ ได้แก่ Neo4j เป็นต้น



## สรุปเนื้อหาทั้งหมด

## ข้อมูล (Data)

สิ่งที่ยังไม่ผ่านการประมวลผล (ข้อมูลดิบ)

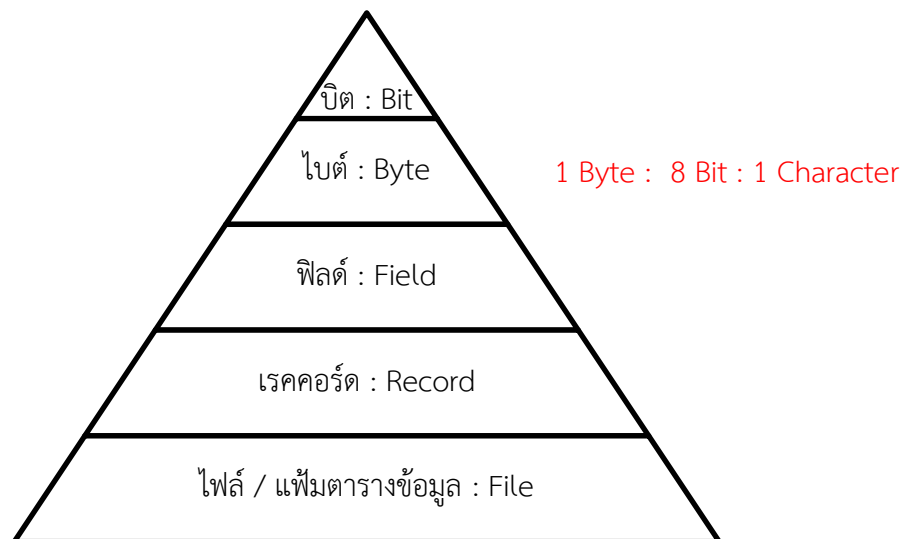
## สารสนเทศ (Information)

Data ที่ผ่านการประมวลผลแล้ว

## การจัดโครงสร้างของแฟ้มข้อมูล (File Organization)

- แบบเรียงลำดับ (Sequential)  
อ่านตามลำดับตั้งแต่ต้นจนจบ เช่น เทปคาสเซต จะข้ามไปฟังเพลงที่ 3, 4 แบบ CD ไม่ได้
- แบบสุ่ม (Random)  
เข้าถึงข้อมูลได้รวดเร็วเพราะไม่เรียงลำดับข้อมูลก่อนเก็บลงไฟล์ เช่น CD, ฮาร์ดดิสก์
- แบบลำดับเชิงตรรกะ (Index sequential)  
แบบเรียงลำดับ + แบบสุ่ม มีตรรกะใช้อ้างถึงตำแหน่งของข้อมูล เช่น CD, ฮาร์ดดิสก์

## ลำดับชั้นของข้อมูล (Data hierarchy)



### ระบบฐานข้อมูล (Database System)

**ฐานข้อมูล :** ไฟล์ / แฟ้มตารางข้อมูลหลาย ๆ แฟ้มที่มีความสัมพันธ์กันมาเก็บรวบรวมกันไว้ที่เดียว  
ประโยชน์

- ลดความซ้ำซ้อน
- มีความถูกต้อง เพียงตรง
- มีความปลอดภัย

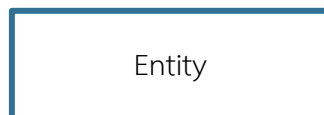
ประเภท

- โครงสร้างแบบตามลำดับชั้น (Hierarchical Model)
- โครงสร้างแบบเครือข่าย (Network Model)
- โครงสร้างแบบเชิงสัมพันธ์ (Relational Model)

### แบบจำลองโครงสร้างของฐานข้อมูล (ER-Diagram)

แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูล (เวลาจะออกแบบฐานข้อมูลขึ้นมาสักตัว ก็ต้องร่างโดยใช้ ER-Diagram ก่อน)

- เอนทิตี (Entity) : สิ่งที่เราสนใจ (เป็นคำนาม)
  - Strong : มีคุณสมบัติในตัวเอง เห็นปุ๊บก็สามารถเข้าใจได้ในตัวมันเอง
  - Weak : ต้องพึ่งพา Strong จึงจะอ่านแล้วเข้าใจ

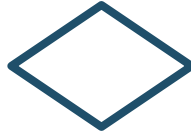


- แอททริบิว (Attribute) : คุณสมบัติของเอนทิตี
  - Simple Attribute : มีองค์ประกอบเดียว แยกย่อยอีกไม่ได้
  - Composite Attribute : แยกย่อยได้หลายตัว
  - Single-Valued Attribute : มีองค์ประกอบเดียว อาจแยกย่อยได้
  - Multivalued Attribute : มีได้หลายค่า
  - Derived Attribute : เกิดจากการคำนวณ
  - Primary Key : ข้อมูลที่เป็นเอกลักษณ์เฉพาะตัว (ไม่ซ้ำกัน)
  - Foreign Key : Primary Key ของเอนทิตีอื่น





- ความสัมพันธ์ (Relationship) : ความสัมพันธ์ระหว่างเอนทิตี
  - One to One
  - One to Many
  - Many to Many



### ระบบจัดการฐานข้อมูล (Database Management System : DBMS)

คือ ซอฟต์แวร์จัดการฐานข้อมูล เช่น MySQL, MongoDB, Oracle

หน้าที่

- แปลงคำสั่งที่ใช้จัดการฐานข้อมูลที่มนุษย์พิมพ์ (พวก SELECT, UPDATE, ... ) ให้ระบบเข้าใจ และทำงานตามที่สั่ง
- สำรอง/กู้คืนข้อมูล
- ควบคุมฐานข้อมูล

### SQL

คือ ภาษาที่ใช้เขียนโปรแกรม (สั่งการระบบฐานข้อมูล พวก SELECT, UPDATE, ... )

คำสั่งภาษาSQL

- ภาษานิยามข้อมูล (Data Definition Language : DDL)
  - CREATE    สร้างตาราง
  - DROP     ลบตาราง
  - ALTER    แก้ไขตาราง
- ภาษาจัดการข้อมูล (Data Manipulation Language : DML)
  - SELECT   เลือกข้อมูล
  - INSERT   เพิ่มข้อมูล
  - UPDATE   แก้ไขข้อมูล
  - DELETE   ลบข้อมูล
- ภาษาควบคุมข้อมูล (Data Control Language : DCL)
  - GRANT    กำหนดสิทธิการเข้าถึงข้อมูล
  - REVOKE   ยกเลิกสิทธิการเข้าถึงข้อมูล

## NoSQL

ระบบจัดการฐานข้อมูลแบบไม่ใช่ SQL มุ่งเน้นความเร็ว ไม่เน้นความสัมพันธ์ของข้อมูล มักนำไปใช้จัดการกับข้อมูลที่มีจำนวนมหาศาล เช่น Pantip, Facebook, Google (ใช้ซอฟต์แวร์ชื่อ MongoDB)

NoSQL 4 ประเภท

- Column-based
- Document-based
- Key-value
- Graph-based