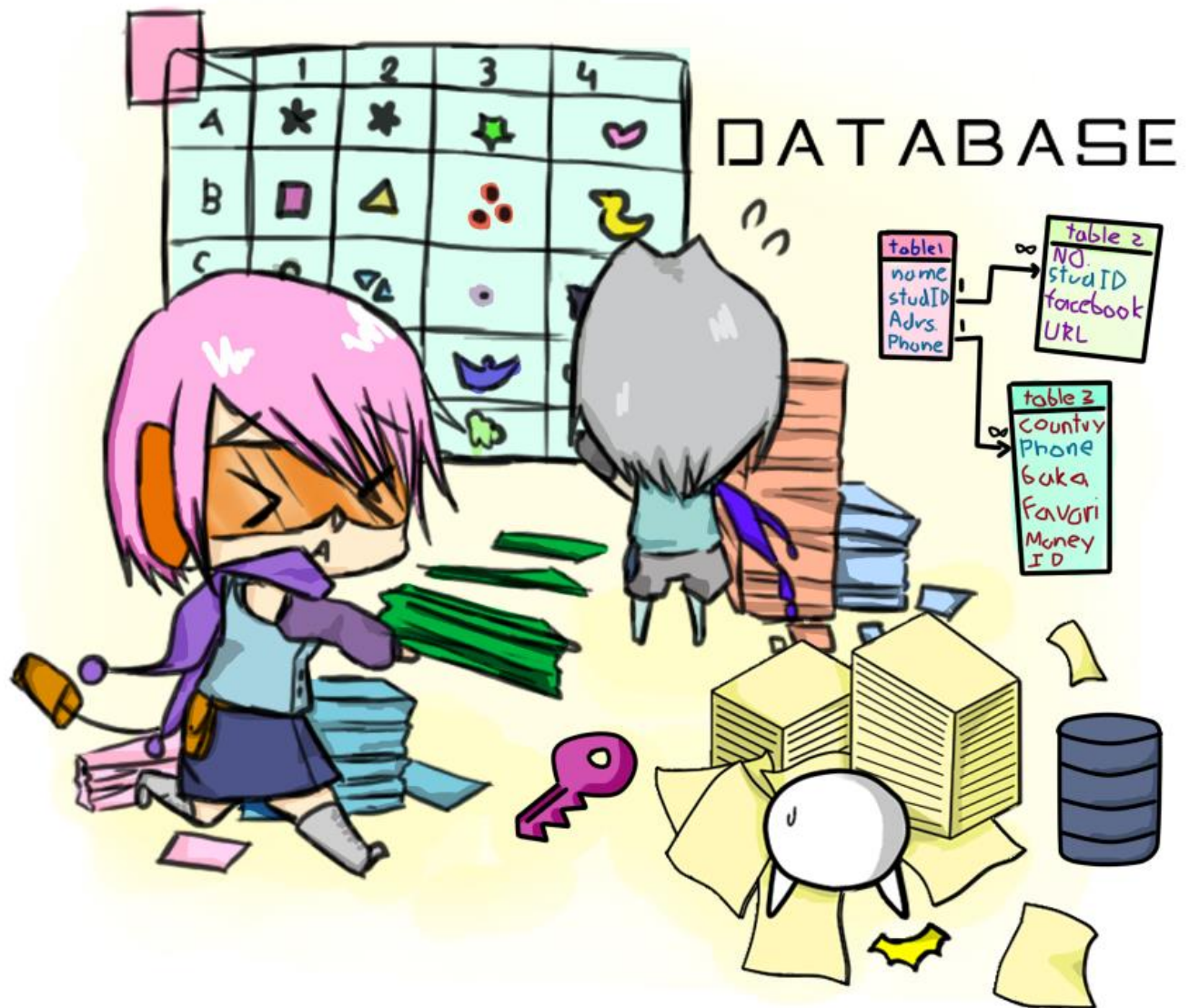


# ToBeIT

@ K M I T L 5 8



## ข้อมูล (Data)

**ข้อมูล** คือ ข้อเท็จจริงเกี่ยวกับบุคคล สิ่งของหรือเหตุการณ์ที่มีอยู่ในรูปของตัวเลข ภาษา ภาพ สัญลักษณ์ต่างๆ ที่มีความหมายเฉพาะตัว เช่น น้ำหนัก ส่วนสูง เลขประจำตัวประชาชน เบอร์โทรศัพท์ รูปเลือด หนังสือ เป็นต้น

### ชนิดของข้อมูล

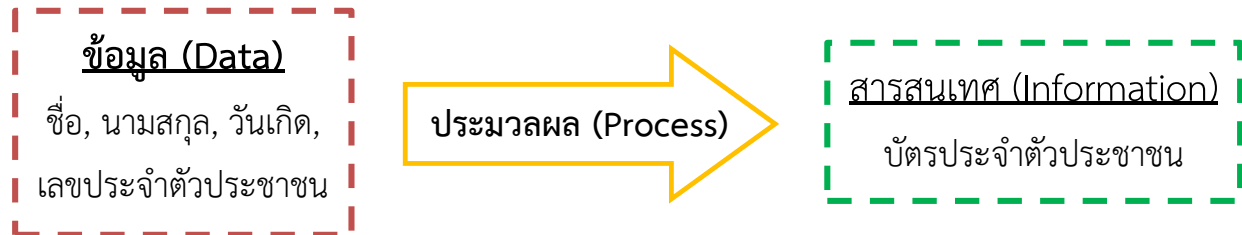
1. ข้อมูลตัวเลข ประกอบไปด้วยตัวเลขเท่านั้น มักจะนำมาใช้ในการคำนวณ
2. ข้อมูลอักขระ ประกอบด้วยตัวอักษร ตัวเลข และอักขระพิเศษหรือเครื่องหมายพิเศษต่างๆ ซึ่งตัวเลขในข้อมูลชนิดนี้จะไม่นำมาคำนวณ เช่น เลขที่บ้าน วันเกิด เป็นต้น
3. ข้อมูลภาพ รับรู้จากการมองเห็น เช่น ภาพดาราศาสตร์ เป็นต้น
4. ข้อมูลเสียง รับรู้จากการได้ยิน เช่น เสียงพูด เสียงเพลง เป็นต้น

### ข้อมูลที่ดี ต้อง.....

1. ถูกต้องแม่นยำ (Accuracy) -> ข้อมูลที่ดีควรมีความถูกต้องแม่นยำ
2. ความเป็นปัจจุบัน (Update) -> เป็นข้อมูลที่ทันสมัยและทันต่อความต้องการของผู้ใช้
3. สมบูรณ์ครบถ้วน (Complete) -> ข้อมูลที่เก็บรวบรวมต้องเป็นข้อมูลที่ให้ข้อเท็จจริง ที่ครบถ้วนทุกด้านทุกประการ ถ้าขาดส่วนใดส่วนหนึ่งไปอาจทำให้นำไปใช้การไม่ได้
4. ตรงกับความต้องการของผู้ใช้ (Relevance) -> ข้อมูลที่นำมาใช้ต้องมีความสอดคล้องกับความต้องการของผู้ใช้ให้มากที่สุด
5. สามารถตรวจสอบได้ (Verifiable) -> ข้อมูลที่ได้ควรมีแหล่งอ้างอิงที่แน่นอน เพื่อป้องกันข้อมูลที่ผิด หรือข้อมูลเท็จ ซึ่งอาจทำให้เกิดผลเสียและข้อผิดพลาดเมื่อนำข้อมูลไปใช้

## สารสนเทศ (Information)

สารสนเทศ คือ ข้อมูลที่ผ่านการจัดเรียง วิเคราะห์ คำนวณ และประมวลผลแล้ว ซึ่งสารสนเทศนั้นมีความหมาย มีคุณค่า มีสาระ สามารถนำไปใช้งานได้



## ฐานข้อมูล (Database)

ฐานข้อมูล คือ กลุ่มของข้อมูลที่รวบรวมไว้และมีการจัดการทำให้สามารถเข้าถึงข้อมูลเพื่อทำการปรับปรุงและนำมาใช้ได้ง่าย



## ระบบฐานข้อมูล (Database System)

เป็นระบบที่ใช้ในการจัดเก็บข้อมูลด้วยคอมพิวเตอร์ ซึ่งทำให้สามารถเรียกใช้ข้อมูลได้ทุกที่

ประเภทของฐานข้อมูล มี 3 ประเภทคือ

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

## โครงสร้างฐานข้อมูลแบบตามลำดับชั้น (Hierarchical Model)

1011 1110  
1001 0010  
1111 1010  
1100 0100  
0000 0001

### บิต (Bit : Binary Digit)

เป็นลำดับชั้นหน่วยข้อมูลที่เล็กที่สุด เป็น  
เลขฐานสองที่มีเพียง 1 และ 0 เท่านั้น

### ไบต์ (Byte)

a, b, c, d

(01000001, 01000010,  
01000011, 01000100)

คือหน่วยที่คอมพิวเตอร์ใช้แสดงตัวอักษร ตัวเลข

หรือสัญลักษณ์ต่างๆ

(1 byte = 1 character (ตัวอักษร) = 8 bit)

name	age	salary

### ฟิลด์ (Field)

คือขอบเขตของข้อมูล ซึ่งประกอบด้วย

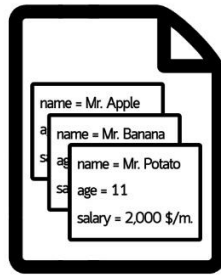
ตัวอักษร(ไบต์)ตั้งแต่ 1 ตัวขึ้นไป

name = Mr. Potato
age = 11
salary = 2,000 \$/m.

### เรคคอร์ด (Record)

คือชุดข้อมูล 1 ชุด ประกอบด้วยหลายๆ ฟิลด์

ที่มีความสัมพันธ์กัน




### ไฟล์ หรือ แฟ้มตารางข้อมูล (File)


คือการนำเอาข้อมูลหลายๆ เรคคอร์ดมารวมกัน


จะกลายเป็นแฟ้มข้อมูล

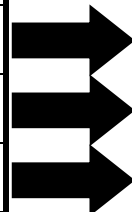
### Field & Record

name	age	salary
Mr. Apple	18	2,900 \$/m.
Mr. Banana	24	4,000 \$/m.
Mr. Potato	11	2,000 \$/m.

  
 Field: name

  
 Field: age

  
 Field: salary



Record: Mr. Apple  
 Record: Mr. Banana  
 Record: Mr. Potato

### จากตารางข้างบน

- ตารางนี้มี 3 ฟิลด์ ได้แก่ name, age, salary

โดยในแต่ละฟิลด์ก็จะมีชนิดของข้อมูลที่แตกต่างกัน (เช่น ฟิลด์ name เก็บข้อมูลเป็นแบบตัวอักษร, ฟิลด์ age และ salary เก็บข้อมูลเป็นแบบตัวเลข)

- ตารางนี้มี 3 เรคคอร์ด ได้แก่ Mr. Apple, Mr. Banana, Mr. Potato

โดยข้อมูลของแต่ละคนเป็น 1 เรคคอร์ด หรือข้อมูล 1 ชุด

(Field แนวตั้ง – Record แนวนอน)

## การจัดโครงสร้างของแฟ้มข้อมูล (File Organization)

การจัดเก็บข้อมูลจะต้องกำหนดโครงสร้างในการจัดเก็บ เพื่อให้สามารถจัดเก็บและเข้าถึงข้อมูลได้รวดเร็ว ถูกต้อง และเหมาะสมกับความต้องการ โดยการเข้าถึงและนำข้อมูลมาใช้จะอาศัยฟิลด์ที่เป็นคีย์หลักในการสืบค้นข้อมูลเสมอ การจัดโครงสร้างของแฟ้มข้อมูลแบ่งได้เป็น 3 ลักษณะ

โครงสร้าง	ข้อดี	ข้อเสีย	สื่อที่ใช้ในการเก็บ
1. แบบเรียงลำดับ (Sequential file)	<ul style="list-style-type: none"> <li>- เสียค่าใช้จ่ายน้อย</li> <li>- ใช้งานได้ง่าย</li> <li>- เหมาะกับงานประมวลผลที่มีการอ่านข้อมูลแบบเรียงลำดับ และมีข้อมูลในปริมาณมาก</li> <li>- สื่อที่ใช้เก็บมีราคาถูก</li> </ul>	<ul style="list-style-type: none"> <li>- เสียเวลาค่อนข้างมากในการหาข้อมูล</li> <li>- ข้อมูลที่ใช้ต้องมีการจัดเรียงลำดับก่อนเสมอ</li> <li>- ไม่เหมาะกับงานที่ต้องแก้ไขเพิ่มเติม ลบข้อมูลเป็นประจำ</li> </ul>	เทปแม่เหล็ก เช่น เทปคาสเซ็ท
2. แบบสุ่ม (Random file)	<ul style="list-style-type: none"> <li>- สามารถทำงานได้รวดเร็ว</li> <li>- เหมาะสมกับงานที่ต้องการแก้ไข เพิ่มเติม ลบรายการเป็นประจำ</li> </ul>	<ul style="list-style-type: none"> <li>- ไม่เหมาะกับงานประมวลผลที่อ่านข้อมูลในปริมาณมาก</li> <li>- การเขียนโปรแกรมเพื่อค้นหาข้อมูลจะซับซ้อน</li> <li>- ไม่สามารถเข้าถึงข้อมูลแบบเรียงลำดับได้</li> </ul>	จานแม่เหล็ก เช่น ดิสก์เก็ต, ฮาร์ดดิสก์ หรือ แผ่น CD
3. แบบลำดับเชิงตรรกะ (Index sequential file)	<ul style="list-style-type: none"> <li>- สามารถรองรับการประมวลผลได้ทั้ง 2 แบบคือ แบบลำดับ และแบบสุ่ม</li> <li>- เหมาะสมกับงานที่ต้องการแก้ไขเพิ่มเติม ลบรายการเป็นประจำ</li> </ul>	<ul style="list-style-type: none"> <li>- สิ้นเปลืองเนื้อที่ในการจัดเก็บตรรกะที่ใช้อ้างอิงถึงตำแหน่งของข้อมูล</li> <li>- การเขียนโปรแกรมเพื่อค้นหาข้อมูลจะซับซ้อน</li> <li>- การทำงานช้ากว่าแบบสุ่มและมีค่าใช้จ่ายสูง</li> </ul>	จานแม่เหล็ก เช่น ดิสก์เก็ต, ฮาร์ดดิสก์ หรือ แผ่น CD

## ประโยชน์ของระบบฐานข้อมูล

### - ลดความซ้ำซ้อนของข้อมูล

ในระบบฐานข้อมูล การเรียก เพิ่มเติม แก้ไขข้อมูล จะทำในระบบเดียวกัน ทำให้ลดความซ้ำซ้อนของข้อมูลได้

### - รักษาความถูกต้องของข้อมูล

เนื่องจากข้อมูลรวมอยู่ในฐานข้อมูลเดียว เมื่อต้องการแก้ไขจะทำให้ข้อมูลทั้งหมดถูกแก้ไขตามไปด้วย

### - มีความปลอดภัยของข้อมูลสูง

ในเมื่อทุกคนสามารถแก้ไขข้อมูลบางส่วนของฐานข้อมูลได้ การป้องกันและรักษาความปลอดภัยจึงต้องมีการเข้าถึงข้อมูล เช่น มีการให้ใส่ username, password

## เครื่องมือสำหรับจัดการฐานข้อมูล (DBMS)

การจัดการข้อมูลด้วยคอมพิวเตอร์โดยปกติจะใช้โปรแกรมที่เรียกว่า ระบบจัดการฐานข้อมูล (DBMS : Database Management System) ซึ่งเป็นซอฟต์แวร์ที่ดูแลจัดการเกี่ยวกับฐานข้อมูล ที่นิยมใช้ในปัจจุบันได้แก่

- |                        |                 |
|------------------------|-----------------|
| - Oracle               | - Cache         |
| - Sybase               | - PostgreSQL    |
| - Microsoft SQL Server | - InterBase     |
| - Microsoft Access     | - Firebird      |
| - MySQL                | - Pervasive SQL |
| - DB2                  | - SAP DB        |

## ลักษณะของระบบจัดการฐานข้อมูล (DBMS)

- สร้างฐานข้อมูล (Create Database)
- เพิ่ม เปลี่ยนแปลง แก้ไข และลบข้อมูล (Add, Change, Edit and Delete data)
- จัดเรียงและค้นหาข้อมูล (Sort and Retrieve data)
- สร้างฟอร์ม และรายงาน (Create Forms and Reports)

## หน้าที่ของระบบจัดการฐานข้อมูล

1. ทำหน้าที่แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูลให้อยู่ในรูปแบบที่ข้อมูลเข้าใจ
2. ทำหน้าที่ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้วไปสั่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) หรือ การเพิ่มข้อมูล (Add) เป็นต้น ฯลฯ
3. ทำหน้าที่ป้องกันความเสียหายที่จะเกิดขึ้นกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดที่สามารถทำงานได้และคำสั่งใดที่ไม่สามารถทำได้
4. ทำหน้าที่รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
5. ทำหน้าที่เก็บรายละเอียดต่างๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลไว้ใน data dictionary ซึ่งรายละเอียดเหล่านี้มักจะถูกเรียกว่า "ข้อมูลของข้อมูล" (Meta Data)
6. ทำหน้าที่ควบคุมให้ฐานข้อมูลทำงานได้อย่างถูกต้องและมีประสิทธิภาพ



## SQL

SQL ย่อมาจาก Structured Query Language คือภาษาที่ใช้ในการเขียนโปรแกรม เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ และเป็นชื่อโปรแกรมฐานข้อมูลอีกด้วย

SQL เป็นโปรแกรมฐานข้อมูลที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน มีประสิทธิภาพการทำงานสูง สามารถทำงานที่ซับซ้อนได้โดยใช้คำสั่งเพียงไม่กี่คำสั่ง

### ประเภทคำสั่งของภาษา SQL

1. ภาษานิยามข้อมูล (Data Definition Language : DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล

CREATE	สร้างตารางข้อมูล
DROP	ลบตารางข้อมูล
ALTER	แก้ไขตารางข้อมูล

2. ภาษาจัดการข้อมูล (Data Manipulation Language : DML) เป็นคำสั่งที่ใช้ในการเรียกใช้เพิ่ม ลบ และเปลี่ยนแปลงข้อมูลในตาราง

SELECT	เลือกข้อมูลในฐานข้อมูล
INSERT	เพิ่มข้อมูลเข้าไปในฐานข้อมูล
UPDATE	แก้ไขข้อมูลในฐานข้อมูล
DELETE	ลบข้อมูลในฐานข้อมูล

3. ภาษาควบคุมข้อมูล (Data Control Language : DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิการอนุญาตหรือยกเลิกการเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของข้อมูล

GRANT	กำหนดสิทธิการเข้าถึงข้อมูล
REVOKE	ยกเลิกสิทธิการเข้าถึงข้อมูล



## NoSQL

NoSQL (Not Only SQL) คือ แนวทางหนึ่งสำหรับจัดการข้อมูลและออกแบบฐานข้อมูลสำหรับข้อมูลขนาดใหญ่ ซึ่งอยู่อย่างกระจัดกระจาย หลากหลายรูปแบบ

NoSQL เป็นฐานข้อมูลที่ไม่มีความสัมพันธ์ เพราะไม่ได้เป็น Relational Database (จะไม่มี JOIN ในการค้นข้อมูล) และเน้นให้ทำงานได้เร็วเป็นหลัก ซึ่ง NoSQL เน้นใช้งานกับปริมาณข้อมูลที่มีจำนวนมาก เช่น Facebook, Twitter, Google

## ข้อดีของ NoSQL

- สามารถขยายระบบได้ง่าย
- รองรับข้อมูลขนาดใหญ่
- รองรับรูปแบบข้อมูลที่หลากหลาย หรือมีความยืดหยุ่นสูงได้

## ตัวอย่างฐานข้อมูลของ NoSQL

แบ่งออกตามประเภทของ NoSQL

1. Document เช่น CouchDB, MongoDB
2. Graph เช่น HyperGraphDB, InfoGrid, Neo4j, sonesGraphDB
3. Key/Value เช่น HBase, MemcacheDB, Project Voldemort, Redis, SimpleDB
4. Tabular เช่น Cassandra, Hypertable

## ER-Diagram (Entity Relationship Diagram)

ER-Diagram (Entity Relationship Diagram) เป็นหนึ่งในวิธีที่นิยมในการเริ่มต้นออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดยเป็นการนำรายละเอียดของข้อมูลที่เราจะนำไปสร้างเป็น database มาวิเคราะห์และเขียนเป็นแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลแต่ละตัว

### ส่วนประกอบของ ER-Diagram

#### 1. Entity

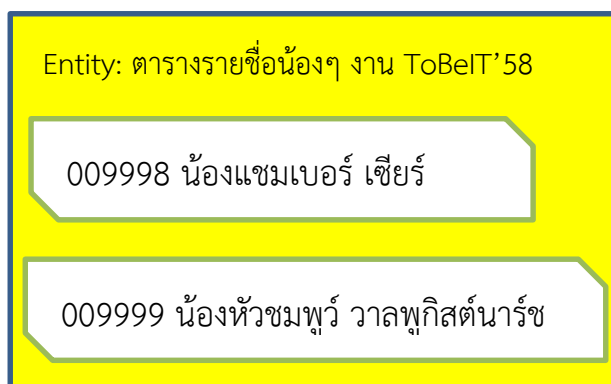
Entity คือสิ่งที่คงอยู่หรือสามารถระบุได้ในความจริง เช่น คน เหตุการณ์ สถานที่ ซึ่งทั่วไปแล้วเอนทิตีจะอยู่ในรูปของคำนาม และ เอนทิตีนี้สามารถมีคุณสมบัติได้หลายอย่าง เช่น นักเรียน มีคุณสมบัติได้หลายอย่างเช่น ชื่อ ที่อยู่ อายุ เพศ ฯลฯ ใช้รูปสี่เหลี่ยมผืนผ้า

#### ชนิดของ Entity

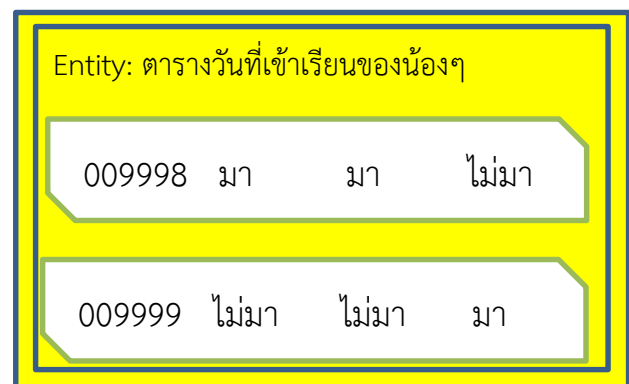
1. **Strong entity** คือ เอนทิตีตัวทั่วไปที่มีคุณสมบัติในตัวเอง สามารถแยกความแตกต่างได้จากการดูข้อมูลในแอตทริบิวต์ของตัวมันได้ เช่น นักศึกษา หรืออาจารย์ หรือสินค้า เป็นต้น

2. **Weak entity** คือขึ้นโดยอาศัย แอตทริบิวต์ในเอนทิตี อื่นจึงจะสามารถบ่งบอกความแตกต่างของข้อมูลในแต่ละ record ใน เอนทิตีนั้นๆได้ (ข้อมูลที่มีมักจะซ้ำ)

Strong entity



Weak entity



## 2. Attribute หรือ Property

Attribute คือคุณสมบัติต่างๆของเอนทิตีซึ่งแต่ละเอนทิตีจะมีคุณสมบัติแล้วแต่เราจะกำหนดว่าจะเก็บข้อมูลคุณสมบัติใดบ้าง ใช้รูปวงกลม

### ชนิดของ Attribute

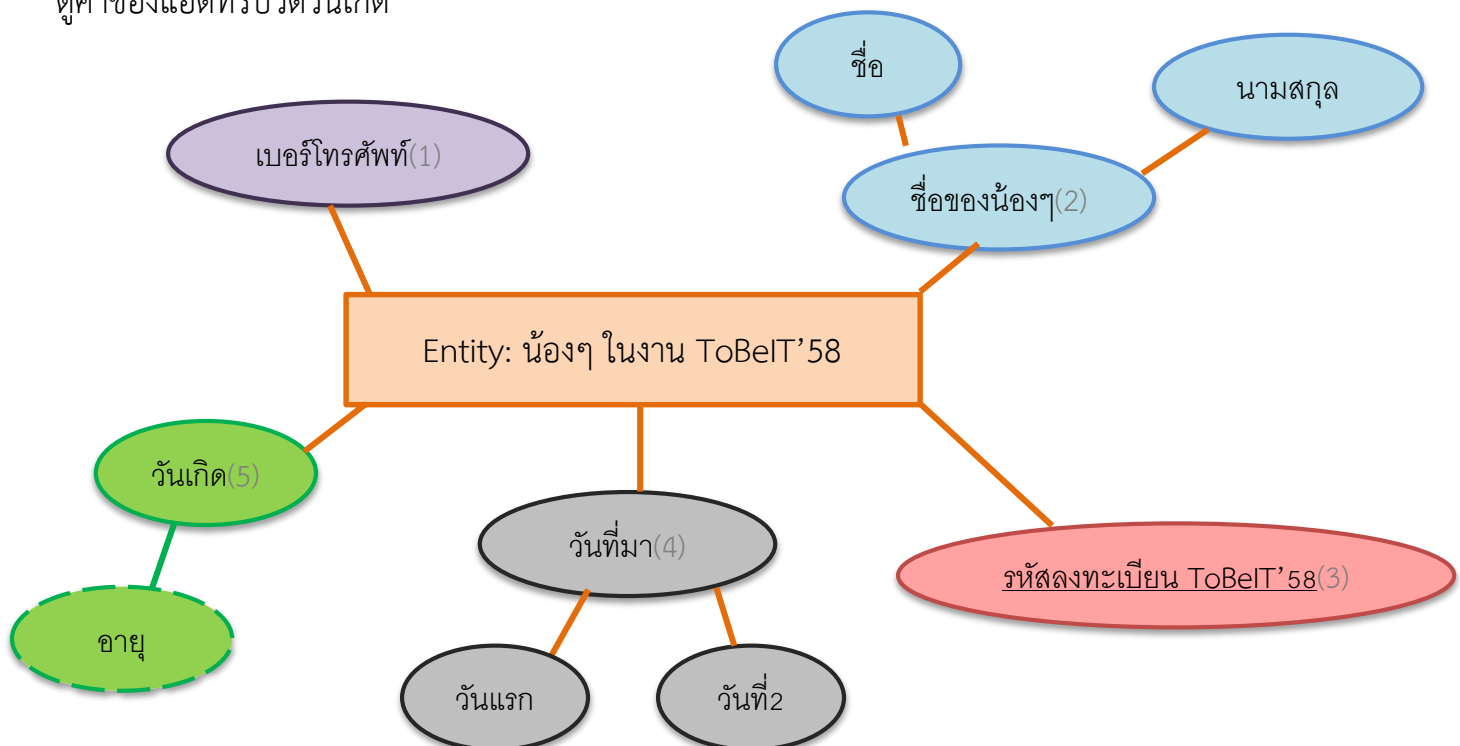
1. Atomic Attribute (Simple Attribute) คือ แอตทริบิวต์ทั่วไปที่มีค่าไม่แบ่งแยกย่อยออกมาได้อีก

2. Composite Attribute คือ แอตทริบิวต์ที่สามารถแยกย่อยออกมาได้แต่ต้องอยู่คู่กันเท่านั้นเช่น ชื่อ และ นามสกุล

3. Primary Key (Key Attribute) คือ เป็นแอตทริบิวต์ที่ถูกเลือกโดยมามีเงื่อนไขว่าจะต้องไม่มีฟิลด์ที่ไม่มีข้อมูลซ้ำกันเลย (หรืออาจซ้ำไม่ได้) ซึ่งเอาไว้ใช้เป็นตัวอ้างอิงในแต่ละเอนทิตี เช่น รหัสลงทะเบียน ToBeIT รหัสบัตรประชาชน จะขีดเส้นใต้แอตทริบิวต์ตัวนั้นเพื่อให้รู้ว่าเป็น Key Attribute

4. Multivalued Attribute คือ แอตทริบิวต์ที่มีตัวเดียวแต่สามารถเก็บค่าได้หลายๆค่าได้ เช่น แอตทริบิวต์ของสีเสื้อ โดยเสื้อตัวหนึ่งอาจจะมีทั้งสีแดงและสีน้ำเงินก็ได้ หรือ แอตทริบิวต์ของเบอร์โทรศัพท์ซึ่งเราอาจเก็บเบอร์โทรศัพท์มากกว่าหนึ่งเบอร์ก็ได้ ใช้สัญลักษณ์ วงรีซ้อนกันสองวง

5. Derived Attribute คือ แอตทริบิวต์ที่ไม่จำเป็นต้องรับค่าเข้ามา อาศัยการคำนวณจากแอตทริบิวต์ตัวอื่นๆเพื่อหาค่าของมันได้ ใช้วงรีเส้นประ เช่นเราจะรู้ค่าของแอตทริบิวต์อายุได้จากการดูค่าของแอตทริบิวต์วันเกิด



### 3. Relationship

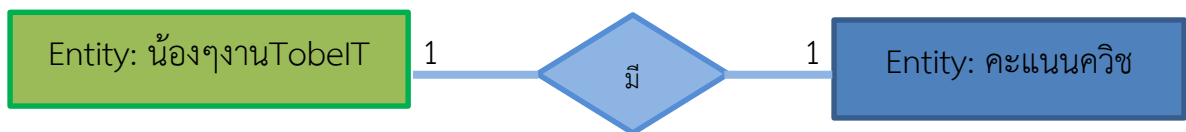
Relationship คือความสัมพันธ์ระหว่างเอนทิตีที่มีความสัมพันธ์ของข้อมูลกันอย่างไร ใช้รูปข้าวหลามตัด

#### ชนิดของ Relationship

ความสัมพันธ์ระหว่างข้อมูลมี 3 ลักษณะ ดังนี้

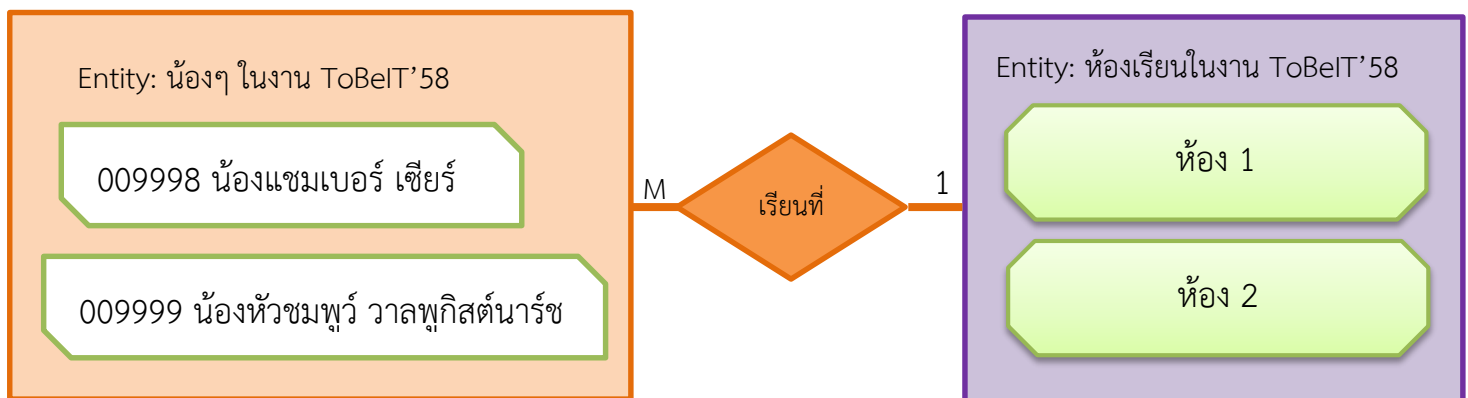
##### 1. One to One

หมายถึงข้อมูล 1 เรคคอร์ดที่อยู่ในตารางหนึ่ง มีความสัมพันธ์หรือเชื่อมโยงกับข้อมูลที่อยู่ในอีกตารางหนึ่ง โดยที่สามารถอ้างอิงได้เพียง 1 เรคคอร์ดเท่านั้น เช่น น้อง 1 คนสามารถมีคะแนนวิชได้เพียง 1 ชุดเท่านั้นเท่านั้น



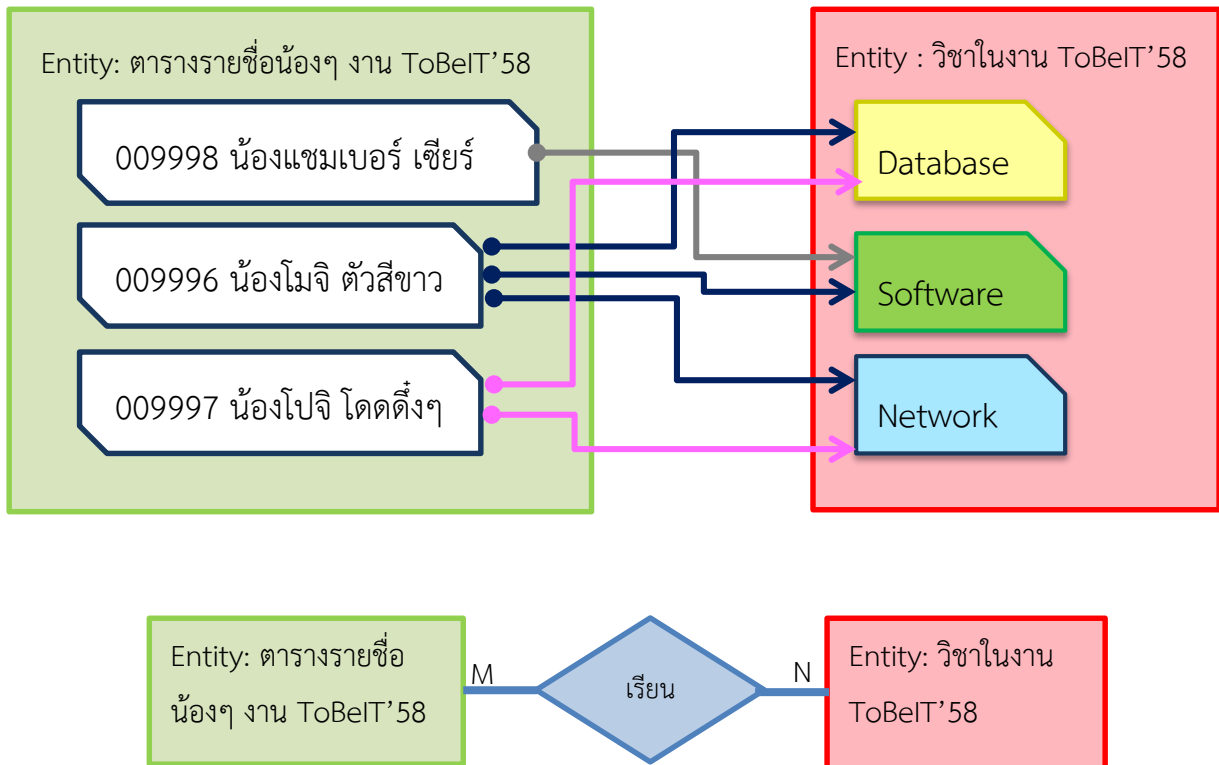
##### 2. One to Many

หมายถึงข้อมูล 1 เรคคอร์ดที่อยู่ในตารางหนึ่ง มีความสัมพันธ์หรือเชื่อมโยงกับข้อมูลที่อยู่ในอีกตารางหนึ่ง โดยที่สามารถอ้างอิงได้มากกว่า 1 เรคคอร์ด เช่น ในมหาวิทยาลัยมีนักศึกษาได้หลายคน และนักศึกษาหลายคนอยู่มหาวิทยาลัยเดียวกัน



### 3. Many to Many

หมายถึงข้อมูลหลายเรคคอร์ดที่อยู่ในตารางหนึ่ง มีความสัมพันธ์หรือเชื่อมโยงกับข้อมูลที่อยู่ในอีกตารางหนึ่ง โดยที่สามารถอ้างอิงได้หลายเรคคอร์ด เช่น นักศึกษา 1 คนสามารถลงทะเบียนได้หลายวิชา และในแต่ละวิชาก็มีนักศึกษาลงทะเบียนหลายคน



#### ขั้นตอนการเขียน ER diagram

1. กำหนด Entity โดยดูจากผู้ใช้ระบบว่าต้องการเก็บข้อมูลอะไรบ้าง
2. กำหนด Attribute แต่ละตัวสำหรับแต่ละ Entity โดยเลือกให้มี Attribute เท่าที่จำเป็น และทำการเลือก Attribute หนึ่งตัวมาเป็น Primary Key โดย Attribute ตัวนั้นต้องมีคุณสมบัติของ Primary Key คือภายใน field ไม่มีข้อมูลซ้ำกันเลย
3. ทำการวิเคราะห์ความสัมพันธ์ระหว่าง Entity แล้วเขียนตัวเชื่อมแสดงความสัมพันธ์ระหว่าง Entity เหล่านั้น