

## บทที่ 2

### สถาปัตยกรรมระบบฐานข้อมูล (Database System Architecture)

#### จุดประสงค์เชิงพฤติกรรม

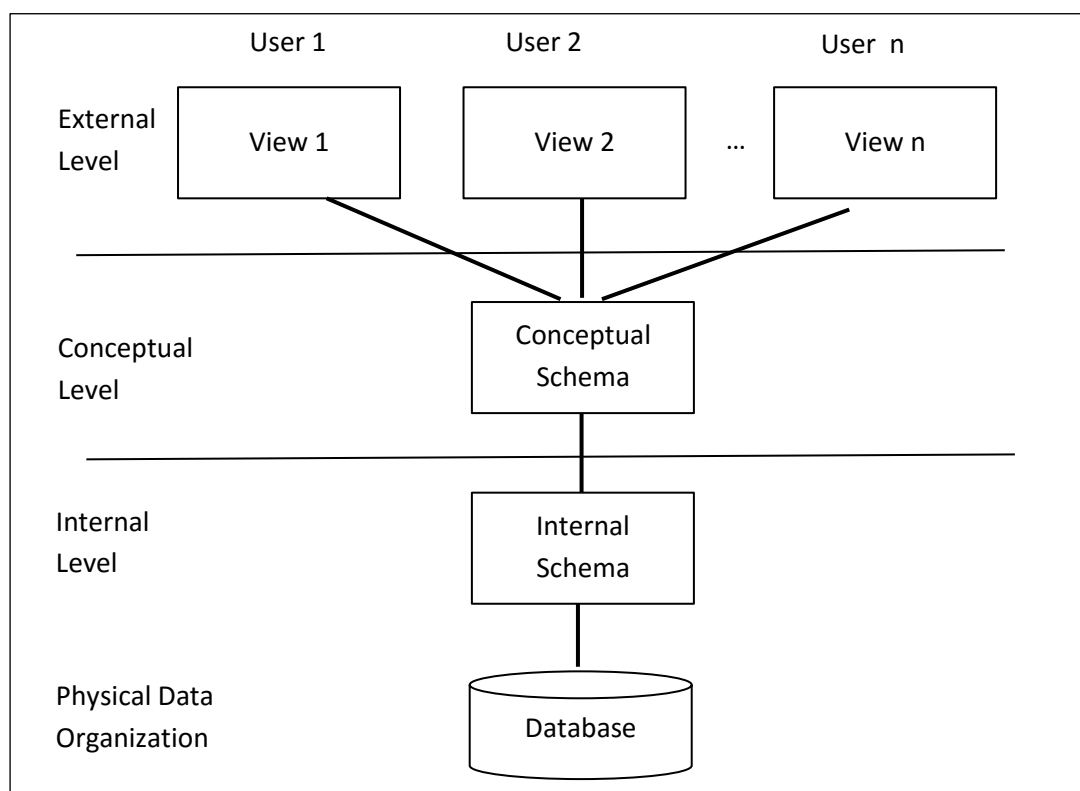
1. อธิบายการทำงานของสถาปัตยกรรม 3 ระดับได้
2. อธิบายความเป็นอิสระของข้อมูลได้
3. อธิบายแบบจำลองข้อมูลแต่ละแบบได้
4. อธิบายการทำงานของฐานข้อมูลเชิงสัมพันธ์ได้
5. อธิบายการกฎการทำงานของ อีเอฟคอตต์ได้

## บทนำ

สถาปัตยกรรมของฐานข้อมูล เป็นการอธิบายโครงสร้างและส่วนประกอบหลักที่นำมาประกอบรวมกันเป็นฐานข้อมูล เพื่อแยกมุมมองของผู้ใช้ออกจากฐานข้อมูลจริงทำให้การทำงานเป็นอิสระจากกัน จากที่กล่าวมาเบื้องต้นจะได้อธิบายรายละเอียดในบทที่ 2 ซึ่งมีเนื้อหาเกี่ยวกับสถาปัตยกรรมฐานข้อมูลทั้ง 3 ระดับชั้น ความเป็นอิสระของข้อมูลเชิงกายภาพและเชิงตรรกะแบบจำลองข้อมูล ฐานข้อมูลเชิงสัมพันธ์ โครงสร้างฐานข้อมูลเชิงสัมพันธ์ คุณสมบัติและประเภทคีย์ของรีเลชัน และกฎ 12 ข้อของอี เอฟ คอตต์

## 2.1 สถาปัตยกรรมฐานข้อมูล (Database Architectures)

สถาบันมาตรฐานแห่งชาติของสหรัฐอเมริกาและคณะกรรมการมาตรฐานการวางแผนและความต้องการ (ANSI/SPARC) ได้แบ่งโครงสร้างของสถาปัตยกรรมฐานข้อมูลออกเป็น 3 ระดับชั้น ประกอบด้วย ระดับภายใน (Internal Level) ระดับแนวคิด (Conceptual Level) และ ระดับภายนอก (External Level) (โอภาส เอี่ยมสิริวงศ์, 2558, หน้า 69 - 72) ดังแสดงในภาพที่ 2.1



ภาพที่ 2.1 สถาปัตยกรรมของระบบฐานข้อมูล ANSI/SPARC 3 ระดับ  
(ที่มา: โอภาส เอี่ยมสิริวงศ์, 2558, หน้า 69)

### 2.1.1 ระดับภายใน (Internal Level)

ระดับภายในเป็นมุมมองของการจัดเก็บข้อมูลจริงในหน่วยความจำสำรอง โดยจะเลือกสื่อเก็บข้อมูลที่เหมาะสม เช่น ฮาร์ดดิสก์ เทปแม่เหล็ก ฯลฯ มุมมองนี้เป็นหน้าที่ของผู้บริหารฐานข้อมูลที่ได้รับผิดชอบทำงานผ่านโปรแกรมระบบปฏิบัติการและโปรแกรมจัดการฐานข้อมูล โดยจะเลือกโครงสร้างการจัดเก็บที่เหมาะสม เช่น โครงสร้างข้อมูลแบบดัชนีหรือแบบบีทรี (B-Tree) วิธีการจัดระเบียบแฟ้มข้อมูลทางกายภาพ วิธีการเข้าถึงข้อมูล การดูแลรักษาความปลอดภัยข้อมูล ซึ่งจะมีผลต่อความเร็วและประสิทธิภาพการทำงานของระบบฐานข้อมูล

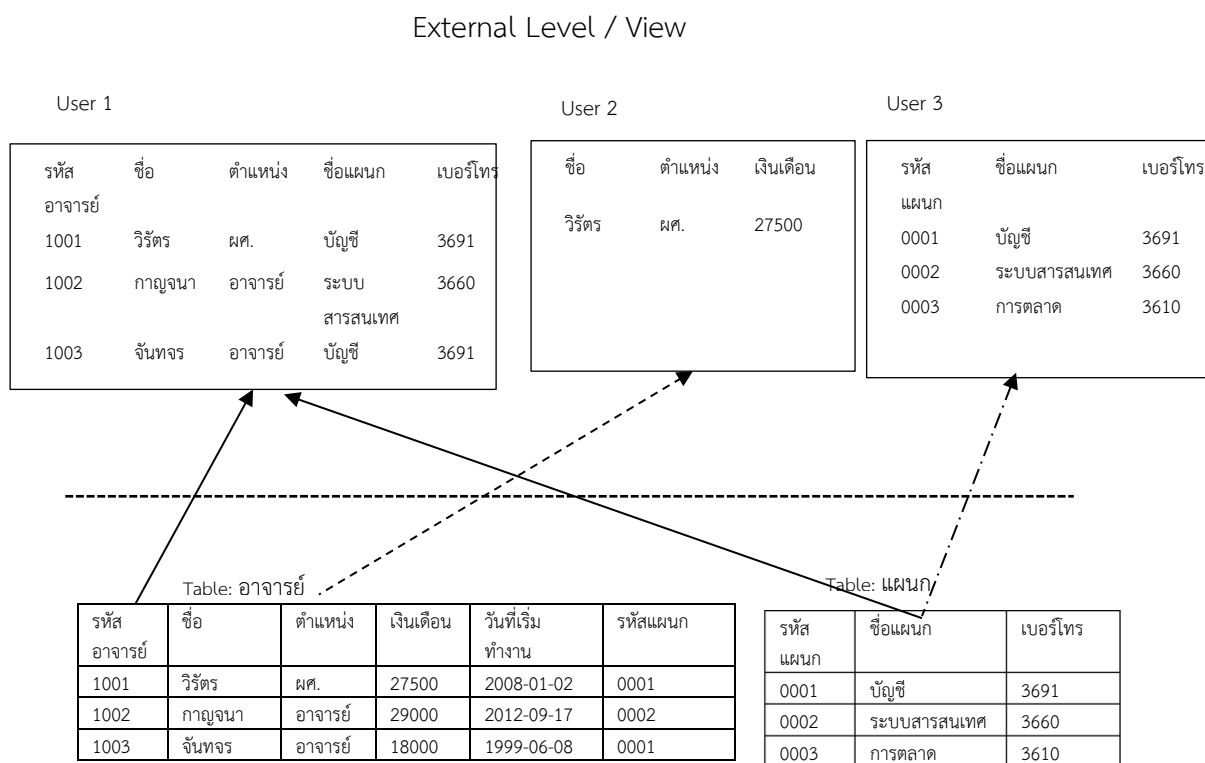
### 2.1.2 ระดับแนวคิด (Conceptual Level)

ระดับแนวคิดเป็นมุมมองของผู้บริหารฐานข้อมูลและโปรแกรมเมอร์ จะมองเห็นรายละเอียดทั้งหมดในการจัดเก็บของฐานข้อมูล เช่น ชื่อเอนทิตี ชื่อแอตทริบิวต์ ชนิดข้อมูลที่เก็บขนาดของข้อมูลที่เก็บ แต่การทำงานในระดับแนวคิดจะเป็นมุมมองเกี่ยวข้องกับ การกำหนดเอนทิตี (Entity) การกำหนดแอตทริบิวต์ (Attribute) การกำหนดความสัมพันธ์ (Relationship) ที่เกิดขึ้นระหว่างเอนทิตี กฎเกณฑ์หรือข้อบังคับในข้อมูล (Constraints) ระบบความปลอดภัยและกฎความคงสภาพในข้อมูล (Integrity) รวมถึงกฎเกณฑ์อื่นที่เกี่ยวข้องกับข้อมูล

ระดับแนวคิดจะมีมุมมองที่เรียกว่า แบบจำลองข้อมูล (Data Model) ซึ่งจะอธิบายว่ามีข้อมูลอะไรบ้างที่จะเก็บลงในฐานข้อมูล มีเอนทิตีอะไรบ้าง และแต่ละเอนทิตีมีแอตทริบิวต์อะไรที่เป็นรายละเอียดของเอนทิตี แต่ละเอนทิตีมีความสัมพันธ์กันในรูปแบบใด

### 2.1.3 ระดับภายนอก (External Level)

ระดับภายนอก หรือมุมมองของผู้ใช้ (View) เป็นการแสดงระดับมุมมองของผู้ใช้ (End User) โดยข้อมูลที่แสดงอาจมาจากข้อมูลบางส่วนของเอนทิตีในฐานข้อมูลหรือมาจากหลายเอนทิตีในฐานข้อมูล ขึ้นอยู่กับความต้องการรายละเอียดในการแสดงผลของผู้ใช้แต่ละคน ซึ่งอาจจะไม่ใช่การแสดงผลข้อมูลทั้งหมดจากมุมมองแนวคิดก็ได้ ยกตัวอย่างเช่น ในตารางข้อมูลอาจารย์ประกอบไปด้วยหลายฟิลด์ แต่อาจารย์หนึ่งท่านไม่จำเป็นต้องมองเห็นข้อมูลทุกฟิลด์ตามที่เก็บอยู่จริงในระดับแนวคิด แต่จะมองเห็นแค่ฟิลด์ ชื่อ ตำแหน่ง แผนก และเบอร์โทร หรืออาจารย์ท่านที่สองต้องการรายละเอียดเพิ่มขึ้นคือ ต้องการมองเห็นเฉพาะข้อมูลตัวเองในฟิลด์ ชื่อ ตำแหน่ง และเงินเดือนเท่านั้น โดยข้อมูลที่ถูกลำมาแสดงเป็นข้อมูลที่ได้มาจากตารางข้อมูลเดียวหรือจากหลายตารางข้อมูลที่เชื่อมความสัมพันธ์กัน แต่ผู้ใช้ในมุมมองภายนอกไม่จำเป็นต้องรู้ว่าฐานข้อมูลจริงที่เก็บอยู่มีตารางเก็บข้อมูลอะไรบ้าง แต่ละตารางมีแอตทริบิวต์อะไร มีชนิดข้อมูลแบบใดหรือขนาดความจุเท่าไร และตารางข้อมูลแต่ละตารางมีความสัมพันธ์กันอย่างไร ผู้ใช้แค่ระบุข้อมูลที่ต้องการเรียกดู ก็จะสามารถแสดงข้อมูลนั้นได้ ตัวอย่างดังภาพที่ 2.2



### Concept Level

**ภาพที่ 2.2** ข้อมูลระดับภายนอกที่ผู้ใช้แต่ละคนมีมุมมองของข้อมูลที่ต่างกัน

จากภาพที่ 2.2 เป็นการแสดงมุมมองระดับแนวคิด มีข้อมูลจัดเก็บอยู่ 2 ตาราง ประกอบด้วย ตารางอาจารย์กับตารางแผนก ซึ่งมีการถ่ายทอดมุมมอง (Mapping) ไปยังมุมมองระดับภายนอก 3 มุมมอง คือ มุมมองของ User 1 จะแสดงข้อมูล รหัสอาจารย์ ชื่อ ตำแหน่ง ชื่อแผนก และเบอร์โทร User 2 แสดงข้อมูล ชื่อ ตำแหน่ง และเงินเดือน User 3 แสดงข้อมูล รหัสแผนก ชื่อแผนกและเบอร์โทร จะเห็นว่าทั้ง 3 มุมมองจะแสดงผลข้อมูลที่แตกต่างกัน

การแสดงผลมุมมองระหว่างมุมมองแนวคิดเพื่อนำเสนอไปยังมุมมองภายนอก เรียกว่า การถ่ายทอดมุมมอง (Mapping) เป็นการนำเสนอข้อมูลจากมุมมองระดับล่างเพื่อให้ระดับที่สูงกว่าเข้าใจและมองเห็นข้อมูลในส่วนที่ตนเองเข้าใจได้ ดังนั้นการถ่ายทอดมุมมองก็แบ่งออกตามสถาปัตยกรรมของฐานข้อมูลคือ มีการถ่ายทอดมุมมองจากระดับของฐานข้อมูลที่เก็บอยู่จริงในสื่อเก็บข้อมูลสำรองไปยังมุมมองแนวคิดเพื่อให้ผู้บริหารฐานข้อมูลมองเห็น ตารางข้อมูล แอตทริบิวต์ความสัมพันธ์และเงื่อนไขการบันทึกข้อมูลของฐานข้อมูลและการถ่ายทอดมุมมองแนวคิดไปยังมุมมองภายนอก เพื่อแสดงผลหน้าจอของผู้ใช้แต่ละคน นั่นเอง

## 2.2 ความเป็นอิสระของข้อมูล (Data Independence)

สถาปัตยกรรมของ ANSI/SPARC เป็นสถาปัตยกรรมที่มีการแยกมุมมองในการมองเห็นฐานข้อมูลออกเป็น 3 ระดับ เพื่อให้การทำงานของฐานข้อมูลมีความเป็นอิสระของข้อมูล โดยแยกมุมมองของผู้ใช้ออกจากฐานข้อมูลที่เก็บอยู่จริงทางกายภาพ และทำให้ข้อมูลที่ถูกจัดเก็บมีความเป็นอิสระ แบ่งได้เป็น 2 ลักษณะคือ

### 2.2.1 ความเป็นอิสระของข้อมูลเชิงกายภาพ (Physical Data Independence)

ความเป็นอิสระของข้อมูลเชิงกายภาพ หมายถึง ถ้ามีการเปลี่ยนแปลงหรือแก้ไขข้อมูลที่อยู่ในโครงสร้างระดับภายในจะไม่ส่งผลให้การทำงานของมุมมองโครงสร้างระดับที่อยู่เหนือขึ้นไป ซึ่งก็คือโครงสร้างระดับแนวคิดและโครงสร้างระดับภายนอก การเปลี่ยนแปลงข้อมูลเชิงกายภาพ อาทิ การใช้โครงสร้างเพิ่มข้อมูลใหม่ การย้ายการจัดเก็บข้อมูลไปหน่วยเก็บข้อมูลใหม่ หรือเปลี่ยนไปใช้หน่วยเก็บข้อมูลแบบอื่น เปลี่ยนวิธีการจัดเก็บข้อมูลจากแบบลำดับไปเป็นแบบดัชนี ตลอดจนการเปลี่ยนแปลงอัลกอริทึมการเรียกใช้ข้อมูลใหม่ เพื่อให้การทำงานเร็วขึ้น ก็จะไม่ส่งผลต่อการทำงานของโครงสร้างระดับแนวคิดและโครงสร้างระดับภายนอกที่อยู่สูงขึ้นไปแต่อย่างใด

### 2.2.2 ความเป็นอิสระของข้อมูลเชิงตรรกะ (Logical Data Independence)

ความเป็นอิสระของข้อมูลเชิงตรรกะ หมายถึง ถ้ามีการเปลี่ยนแปลงหรือแก้ไขข้อมูลที่อยู่ในโครงสร้างระดับแนวคิดจะไม่มีผลต่อมุมมองโครงสร้างระดับภายนอกที่ผู้ใช้นั้นใช้งานอยู่แต่อย่างใด ผู้ใช้ยังคงเรียกดูข้อมูลในรูปแบบที่แต่ละคนต้องการได้ปกติ การเปลี่ยนแปลงข้อมูลในระดับแนวคิด เช่น เพิ่มตารางใหม่ เพิ่มฟิลด์ในตารางเดิมเพื่อให้ได้สารสนเทศที่ครบถ้วนเพิ่มขึ้น การลบตารางข้อมูลบางตารางออก แก้ไขความสัมพันธ์ของตารางข้อมูลใหม่ ฯลฯ ก็ไม่มีผลกระทบต่อการทำงานในระดับโครงสร้างภายนอกแต่อย่างใด โปรแกรมเมอร์ที่เขียนโปรแกรมประยุกต์เพื่อเรียกใช้งานข้อมูลที่เกิดขึ้นในฐานข้อมูลก็ไม่ต้องมานั่งเขียนคำสั่งใหม่แต่อย่างใด โปรแกรมก็ยังคงสามารถใช้งานได้ปกติ

จากความเป็นอิสระของข้อมูลทั้ง 2 ลักษณะ ทำให้ผู้ใช้แต่ละคนสามารถสร้างมุมมองของข้อมูลได้แตกต่างกันได้ หรือสามารถออกรายงานข้อมูลของผู้ใช้แต่ละคนได้ไม่จำเป็นต้องเป็นข้อมูลรายงานที่เหมือนกัน หรืออาจจะเหมือนกันก็ได้ จากฐานข้อมูลเดียวกัน ผู้ใช้ไม่ต้องทำงานกับฐานข้อมูลโดยตรงและไม่จำเป็นต้องรู้ว่าฐานข้อมูลจริงมีการจัดเก็บข้อมูลอะไรบ้างและจัดเก็บอย่างไร เป็นหน้าที่ของผู้บริหารฐานข้อมูลและโปรแกรมเมอร์ที่จะสร้างมุมมองการแสดงผลหรือรายงานให้ผู้ใช้ตามความต้องการ ซึ่งจะทำงานผ่านโปรแกรมจัดการฐานข้อมูล (DBMS) ร่วมกับการเขียนโปรแกรมภาษาต่าง ๆ อาทิ ภาษา JAVA, ภาษา PHP ฯลฯ และฐานข้อมูลชุดเดียวกันสามารถให้บริการกับโปรแกรมประยุกต์ต่าง ๆ ที่ถูกสร้างจากภาษาคอมพิวเตอร์ต่างภาษากันได้โดยไม่ส่งผล

กระทบต่อโปรแกรมประยุกต์อื่นแต่อย่างใด โปรแกรมจัดการฐานข้อมูลจะทำหน้าที่ในการควบคุมการเข้าถึงข้อมูลพร้อมกันเพื่อรักษาความคงสภาพของข้อมูล

## 2.3 แบบจำลองฐานข้อมูล (Database Model)

แบบจำลองข้อมูล มีนักวิชาการหลาย ๆ ท่านได้ให้ความหมายของแบบจำลองฐานข้อมูล ไว้ดังนี้

แบบจำลองฐานข้อมูล หมายถึง แบบจำลองที่ใช้อธิบายและจัดการข้อมูล การนำเสนอข้อมูล (Data Type) ความสัมพันธ์ระหว่างข้อมูล (Relation) กฎเกณฑ์และข้อบังคับของข้อมูลในระบบที่ผู้ใช้เห็นและเข้าใจได้ (สายสุนีย์ จัปโจร, 2555, หน้า 34)

แบบจำลองฐานข้อมูล หมายถึง โครงสร้างข้อมูลในระดับตรรกะที่อาจเป็นการนำเสนอข้อมูลในรูปแบบกราฟิก หรือรูปแบบอื่น ที่นำเสนอข้อมูล ความสัมพันธ์ระหว่างข้อมูลที่ง่ายต่อการทำความเข้าใจ (สุจิตรา อุดลย์เกษม และวรัฐา นพพรเจริญกุล, 2560, หน้า 56)

แบบจำลองฐานข้อมูล หมายถึง รูปแบบที่ใช้อธิบายลักษณะของการจัดเก็บข้อมูล ความสัมพันธ์ระหว่างข้อมูล วิธีการจัดการข้อมูลรวมทั้งเงื่อนไขและข้อกำหนดเกี่ยวกับข้อมูล (ศุภชัย จิระรังสิณี และ ขจรศักดิ์ สังข์เสรีญ, 2549, หน้า 3)

แบบจำลองฐานข้อมูล หมายถึง โครงสร้างระดับตรรกะ (Logical) ที่นำเสนอข้อมูลและความสัมพันธ์ระหว่างข้อมูลให้ผู้ใช้เห็นและเข้าใจได้ (ครรชิต มาลัยวงศ์, สมลักษณ์ ละอองศรี และ ทศนีย์วรรณ ศรีประดิษฐ์, 2544, หน้า 94)

นอกจากนี้ Hector Garcia-Molina Jeffery D. Ullman และ Jennifer Widom (2009, Page 17 - 18) ให้ความหมายแบบจำลองฐานข้อมูลว่า แบบจำลองฐานข้อมูล เป็นเอกสารที่อธิบายข้อมูลข่าวสารซึ่งมีส่วนประกอบหลัก 3 อย่าง คือ 1) โครงสร้างที่อธิบายการจัดเก็บข้อมูลที่อธิบายถึงฐานข้อมูลในเชิงกายภาพและระดับแนวคิด 2) ตัวดำเนินการกับข้อมูล เพื่อใช้ในการเข้าถึงข้อมูลในฐานข้อมูล และ 3) ข้อบังคับหรือเงื่อนไขในการบันทึกข้อมูลลงไปในฐานข้อมูล ซึ่งจะอธิบายข้อจำกัดของข้อมูลที่จะสามารถบันทึกข้อมูลได้

แบบจำลองฐานข้อมูล หมายถึง แผนภาพโครงสร้างของแบบจำลองฐานข้อมูลที่ถูกออกแบบเพื่อจัดเก็บข้อมูล เพื่อให้ผู้ใช้เข้าใจถึงโครงสร้างที่จัดเก็บข้อมูล เห็นความสัมพันธ์ระหว่างข้อมูลและเงื่อนไขหรือข้อบังคับเกี่ยวกับข้อมูลในระบบ

แบบจำลองฐานข้อมูล เป็นเครื่องมือที่นักวิเคราะห์และออกแบบฐานข้อมูลใช้สื่อสารและทำความเข้าใจกับโปรแกรมเมอร์และผู้ใช้ให้เข้าใจตรงกันถึงฐานข้อมูลที่ได้ออกแบบ โดยจะเขียนอยู่ในแผนภาพจำลองที่ง่ายต่อการเข้าใจ (โอภาส เอี่ยมสิริวงศ์, 2558, หน้า 88) สามารถแบ่งรูปแบบของแบบจำลองฐานข้อมูลออกเป็น 4 รูปแบบ รายละเอียดดังนี้

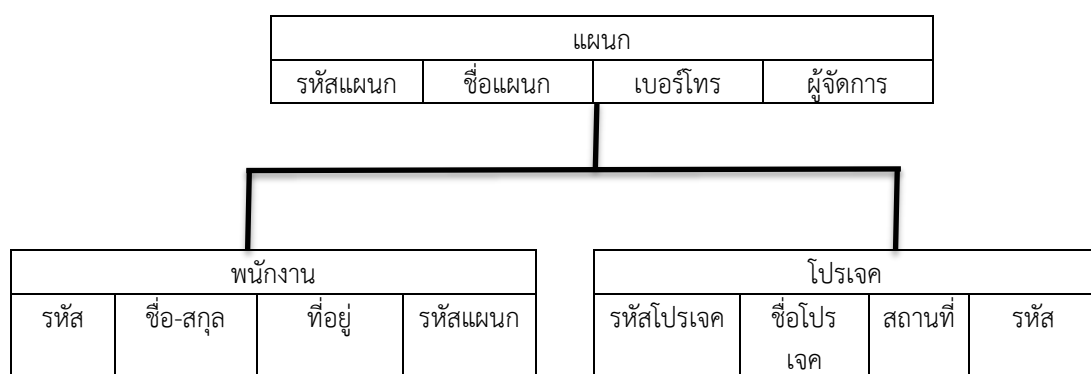
### 2.3.1 แบบจำลองฐานข้อมูลแบบลำดับชั้น (Hierarchical Database Model)

เป็นแบบจำลองของฐานข้อมูลที่เกิดขึ้นมาก่อนแบบจำลองอื่น ถูกคิดค้นโดยบริษัท North America Rockwell ทำงานร่วมกับโครงการสำรวจดวงจันทร์ด้วยยานอวกาศ Apollo ได้พัฒนาแบบจำลองฐานข้อมูลแบบลำดับชั้นขึ้นมา (Peter Rob and Carlos Coronel ,2009 ,Page 36) ซึ่งมีนักวิชาการหลาย ๆ ท่านให้ความหมายของแบบจำลองฐานข้อมูลแบบลำดับชั้นไว้ดังนี้

แบบจำลองฐานข้อมูลแบบลำดับชั้น หมายถึง แบบจำลองที่เหมือนกับโครงสร้างแบบต้นไม้ ตารางจะมีความสัมพันธ์แบบพ่อกับลูก ตารางลูกแต่ละตารางจะมีความสัมพันธ์แบบเดียวกับตารางพ่อ แต่ตารางพ่ออาจมีความสัมพันธ์กับตารางลูกหลายตาราง ตารางลูกจะอยู่ได้ตามตารางพ่อและมีข้อมูลที่สอดคล้องกัน (Gavin Powell, 2006, Page 7)

แบบจำลองฐานข้อมูลแบบลำดับชั้น หมายถึง แบบจำลองที่มีโครงสร้างการจัดเป็นโครงสร้างแบบต้นไม้ รากของต้นไม้ (Root) จะทำหน้าที่เป็นโหนดแม่ ซึ่งประกอบด้วยลูก โดยที่ลูกไม่สามารถมีโหนดแม่ได้หลายโหนดพร้อมกัน แต่โหนดแม่จะมีโหนดลูกได้หลายโหนด (รัฐสิทธิ์ สุขะหุต, 2555, หน้า 25)

แบบจำลองฐานข้อมูลแบบลำดับชั้น หมายถึง แบบจำลองที่มีการออกแบบการจัดเก็บข้อมูลแบบโครงสร้างต้นไม้ (Tree Structure) มีการจัดการข้อมูลจากบนลงล่าง มีไฟล์ที่อยู่บนสุดเรียกโหนดพ่อ และข้อมูลที่มีความสัมพันธ์โหนดพ่อเรียกว่าโหนดลูก โดยโหนดพ่อสามารถมีโหนดลูกได้หลายโหนด แต่โหนดลูกจะสัมพันธ์กับโหนดพ่อเพียงโหนดเดียว โหนดลูกสามารถมีโหนดลูกแยกย่อยลงไปได้อีกเรื่อย โดยจะเชื่อมความสัมพันธ์ระหว่างโหนดพ่อกับโหนดลูกลงไปตามลำดับชั้น ดังภาพที่ 2.3



ภาพที่ 2.3 แบบจำลองฐานข้อมูลแบบลำดับชั้น

ดัดแปลงจาก ครรชิต มัลลียงศ์, สมลักษณ์ ละอองศรี และทัศนีย์วรรณ  
ศรีประดิษฐ์ (2544, หน้า 103)

จากภาพที่ 2.3 เป็นแบบจำลองฐานข้อมูลแบบลำดับชั้น ซึ่งแสดงความสัมพันธ์ของแผนกมีความสัมพันธ์กับโหนดลูก 2 โหนด คือโหนดพนักงานกับโหนดโปรเจกต์ แต่โหนดพนักงานจะมีความสัมพันธ์กับโหนดแผนกซึ่งเป็นโหนดพ่อเพียงโหนดเดียว เป็นลักษณะความสัมพันธ์แบบ 1:M

แบบจำลองฐานข้อมูลแบบลำดับชั้นจะรองรับความสัมพันธ์ของข้อมูลได้เฉพาะความสัมพันธ์แบบ หนึ่งต่อหนึ่ง (1:1) และ หนึ่งต่อกลุ่ม (1:M) ไม่รองรับความสัมพันธ์แบบกลุ่มต่อกลุ่ม (M:N) นอกจากนี้ไม่มีการพัฒนาโปรแกรมจัดการฐานข้อมูลที่ใช้กับแบบจำลองนี้ จึงทำให้มีข้อจำกัดสำหรับการใช้งานในปัจจุบัน จึงไม่ได้รับความนิยม

### 2.3.2 แบบจำลองฐานข้อมูลแบบเครือข่าย (Network Database Model)

แบบจำลองฐานข้อมูลแบบเครือข่าย ได้ออกข้อกำหนดเกี่ยวกับฐานข้อมูลนี้ขึ้นมาในปี ค.ศ. 1969 โดยหน่วยงานชื่อ CODASYL (Committee on Data System languages) เพื่อกำหนดข้อกำหนดเกี่ยวกับการประมวลผลและภาษาที่ใช้จัดการข้อมูลแบบเครือข่าย (รัฐสิทธิ์ สุขะหุต, 2555, หน้า 25) มีนักวิชาการหลาย ๆ ท่านให้ความหมายของแบบจำลองฐานข้อมูลแบบเครือข่ายไว้ดังนี้

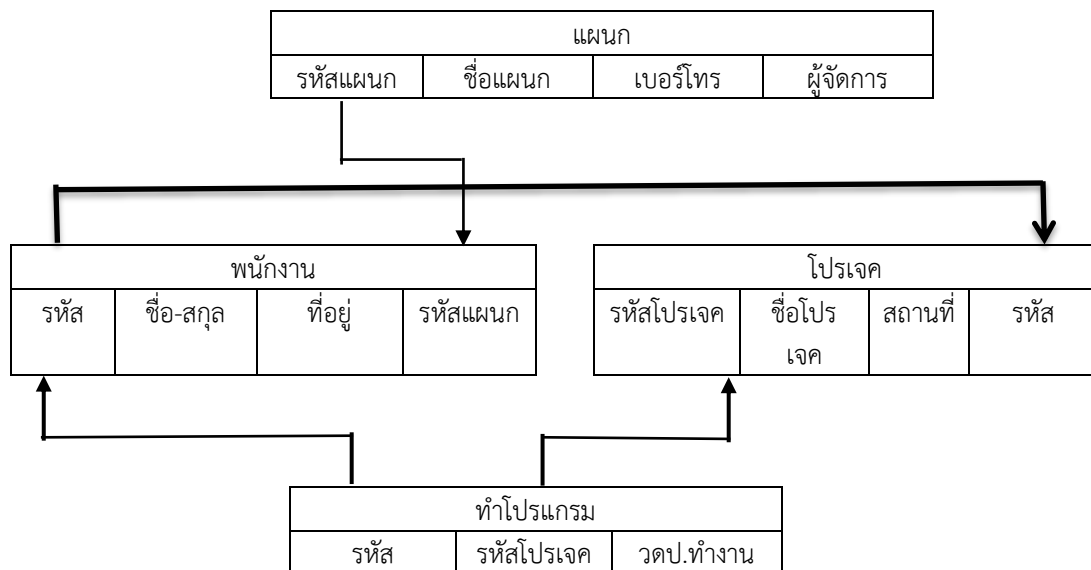
แบบจำลองฐานข้อมูลแบบเครือข่าย หมายถึง แบบจำลองที่ปรับแต่งรูปแบบมาจากฐานข้อมูลแบบลำดับชั้น แต่ทำให้ตารางลูกสามารถมีความสัมพันธ์กับตารางพ่อได้มากกว่า 1 ตาราง ทำให้รองรับความสัมพันธ์แบบหลายต่อหลายกลุ่ม (M:N) ได้ (Gavin Powell, 2006, Page 8)

แบบจำลองฐานข้อมูลแบบเครือข่าย หมายถึง แบบจำลองฐานข้อมูลที่รองรับความสัมพันธ์ข้อมูลที่มีความซับซ้อนยิ่งขึ้นและมีประสิทธิภาพสูงกว่าแบบลำดับชั้น แต่ยังมีโครงสร้างที่คล้ายกัน คือมีลำดับชั้นแบบบนลงล่าง แต่จะแตกต่างตรงที่ โหนดจะมีความสัมพันธ์กับโหนดอื่น ๆ ได้หลายโหนด (สิทธิชัย ชูสำโรง, 2558, หน้า 45)

แบบจำลองฐานข้อมูลแบบเครือข่าย หมายถึง แบบจำลองฐานข้อมูลที่มีลักษณะคล้ายกับแบบจำลองฐานข้อมูลแบบลำดับชั้น แต่จะสามารถรองรับความสัมพันธ์ที่ซับซ้อนมากกว่า คือรองรับความสัมพันธ์แบบโหนดลูกสามารถมีความสัมพันธ์แบบหนึ่งต่อหลายโหนดแล้วยังรองรับความสัมพันธ์แบบหลายต่อหลายโหนดได้ โดยโหนดลูกสามารถมีความสัมพันธ์กับโหนดพ่อได้หลายโหนด นั่นเอง

เป็นแบบจำลองฐานข้อมูลแบบเครือข่าย เป็นแบบจำลองที่ออกแบบมาเพื่อแก้ปัญหาของแบบจำลองข้อมูลแบบลำดับชั้นที่ไม่สามารถรองรับความสัมพันธ์ของข้อมูลในรูปแบบแบบหลายต่อหลายโหนด โดยแบบจำลองนี้จะมีลักษณะโครงสร้างการจัดเก็บเหมือนแบบจำลองแบบลำดับชั้น แต่รองรับความสัมพันธ์ที่โหนดลูกสามารถมีความสัมพันธ์เพื่อเชื่อมต่อกับโหนดพ่อได้มากกว่า 1 โหนด ทำให้การจัดเก็บข้อมูลสามารถลดความซ้ำซ้อนได้และเส้นทางการเข้าถึงข้อมูลมีหลายเส้นทางและความยืดหยุ่นมากขึ้น ดังภาพที่ 2.4





ภาพที่ 2.4 แบบจำลองฐานข้อมูลแบบเครือข่าย

ดัดแปลงจาก ครรชิต มัลลียงค์, สมลักษณ์ ละอองศรี และทัศนีย์วรรณ  
ศรีประดิษฐ์ (2544, หน้า 110)

จากภาพที่ 2.4 แสดงแบบจำลองฐานข้อมูลแบบเครือข่าย โดยตารางแผนกจะมีความสัมพันธ์กับพนักงาน ตารางพนักงานมีความสัมพันธ์กับตารางโครงการ ส่วนตารางการทำโปรแกรมซึ่งเป็นโหนดลูกสามารถมีความสัมพันธ์กับโหนดพ่อคือตารางพนักงานและตารางโครงการ ซึ่งเป็นความสัมพันธ์แบบหลายต่อหลายกลุ่มนั่นเอง

แบบจำลองฐานข้อมูลแบบเครือข่ายในปัจจุบันไม่ได้รับความนิยมใช้งานเช่นเดียวกับแบบจำลองแบบลำดับชั้น เพราะข้อจำกัดเรื่องของโปรแกรมจัดการฐานข้อมูลที่ไม่ได้พัฒนาต่อเนื่องจากฐานข้อมูลไม่มีความเป็นอิสระจากโครงสร้าง เนื่องจากถ้ามีการเปลี่ยนแปลงโครงสร้างต้องมีการแก้ไขโปรแกรมตามไปด้วยจึงเกิดความยุ่งยากในการพัฒนาโปรแกรมมาใช้งาน

### 2.3.3 แบบจำลองฐานข้อมูลเชิงสัมพันธ์ (Relation Database Model)

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ เป็นแบบจำลองฐานข้อมูลที่ได้รับการนิยมนำมาใช้งานกันมากที่สุดในปัจจุบัน มีซอฟต์แวร์จัดการฐานข้อมูลที่พัฒนาขึ้นมารับการทำงานจำนวนมาก ซึ่งมีซอฟต์แวร์ที่ใช้งานกับระบบฐานข้อมูลขนาดเล็กจนถึงระบบฐานข้อมูลที่จัดเก็บข้อมูลเป็นฐานข้อมูลขนาดใหญ่ ตลอดจนมีภาษาคอมพิวเตอร์ที่สามารถเรียกใช้งานฐานข้อมูลเชิงสัมพันธ์คือ ภาษา เอสคิวแอล (SQL: Structure Query Language) เป็นภาษาที่มีโครงสร้างที่ง่ายต่อการใช้งานแบบจำลองฐานข้อมูลเชิงสัมพันธ์ ได้รับการพัฒนาขึ้นมาตั้งแต่ปี ค.ศ. 1970 เป็นแบบจำลองที่แสดง

ในรูปแบบของตาราง ถูกนำมาใช้ในเชิงพาณิชย์ครั้งแรกในปี 1980 (Ramez Elmasri and Shamkant B. Navathe, 2011, Page 59) และได้รับความนิยมจนถึงปัจจุบัน แบบจำลองฐานข้อมูลเชิงสัมพันธ์ได้มีนักวิชาการหลาย ๆ ท่านให้ความหมายของไว้ดังนี้

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ หมายถึง แบบจำลองฐานข้อมูลที่แสดงอยู่ในรูปแบบตาราง (Table) 2 มิติ ที่ประกอบด้วยแถว (Row) และคอลัมน์ (Column) โดยคอลัมน์แสดงถึงคุณสมบัติของแถว (Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer Widom, 2009, Page 21 – 22)

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ หมายถึง การนำเสนอความสัมพันธ์ระหว่างข้อมูลในรีเลชัน แบบ 2 มิติ ประกอบด้วยแถวและคอลัมน์ (สุจิตรา อุดุลย์เกษม, 2553, หน้า 52)

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ หมายถึง แบบจำลองที่จัดเก็บข้อมูลเป็นเอนทิตีหรือแฟ้มข้อมูล มีการกำหนดความสัมพันธ์ระหว่างแฟ้มข้อมูลที่อยู่ในฐานข้อมูลเดียวกัน (พลพฐ์ ปิยวรรณ และ สุภาพร เชิงเอี่ยม, 2554, หน้า 130)

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ หมายถึง แบบจำลองฐานข้อมูลที่แสดงอยู่ในรูปของตารางแบบ 2 มิติ ประกอบด้วย แถวและคอลัมน์

แบบจำลองฐานข้อมูล จะถูกจัดเก็บในรูปแบบตาราง และแต่ละตารางจะมีความสัมพันธ์กับตารางอื่น ๆ โดยใช้ คีย์ (Key) ในการอ้างอิงกับตาราง ความสัมพันธ์ที่เกิดกับตารางอื่นจะสามารถรองรับความสัมพันธ์ระหว่างตารางได้ทั้งความสัมพันธ์แบบหนึ่งต่อหนึ่ง (1:1) หนึ่งต่อกลุ่ม (1:M) และ กลุ่มต่อกลุ่ม (M:N) ดังแสดงในภาพที่ 2.3

ตารางพนักงาน

รหัสพนักงาน	ชื่อ-สกุล	เงินเดือน	ตำแหน่ง	รหัสแผนก
1001	วิชัย ธรรมเริง	26,900	วิเคราะห์ลูกค้า	1
1002	สมศักดิ์ ระวิหังโส	19,900	ขาย	1
2001	ทองใบ ไชยดา	25,600	บัญชี	2
3001	เพิ่มพูน คุณวงศ์	29,700	โปรแกรมเมอร์	3

ตารางแผนก

รหัสแผนก	ชื่อแผนก	เบอร์โทร
1	การตลาด	2560
2	บัญชี	3500
3	เทคโนโลยีสารสนเทศ	3650

ภาพที่ 2.5 แบบจำลองฐานข้อมูลเชิงสัมพันธ์

จากภาพที่ 2.3 เป็นแบบจำลองฐานข้อมูลเชิงสัมพันธ์ ประกอบด้วยตารางพนักงาน จัดเก็บรายละเอียดของพนักงาน มีแอตทริบิวต์รหัสพนักงานเป็นคีย์หลัก (Primary Key) และตารางแผนกมีแอตทริบิวต์รหัสแผนกเป็นคีย์หลัก โดยรหัสแผนกที่เป็นคีย์หลักในตารางแผนกไปเป็นแอตทริบิวต์ในตารางพนักงาน จึงทำให้ตารางทั้งสองเกิดความสัมพันธ์ จึงทำให้ทราบว่า วิชัย ธรรมเริง สังกัดแผนกการตลาด เบอร์โทรภายในที่ใช้ติดต่อคือเบอร์ 2560

#### 2.3.4 แบบจำลองฐานข้อมูลเชิงวัตถุ (Object-Oriented Database Model)

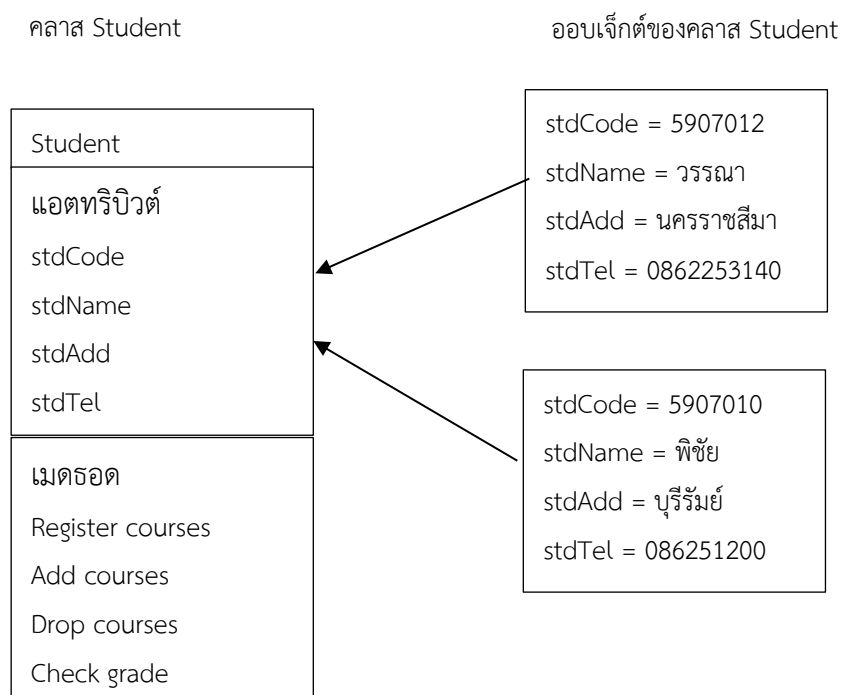
แบบจำลองฐานข้อมูลเชิงวัตถุ ได้มีนักวิชาการหลาย ๆ ท่านให้ความหมายของไว้ดังนี้

แบบจำลองฐานข้อมูลเชิงวัตถุ หมายถึง แบบจำลองฐานข้อมูลที่พัฒนาขึ้นมาเพื่อรองรับการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) ด้วยการมองทุกอย่างเป็นวัตถุที่เป็นแหล่งรวบรวมข้อมูลและโอเปอเรชันการจัดเก็บและเข้าถึงข้อมูล (โอภาส เอี่ยมสิริวงศ์, 2558, หน้า 94)

แบบจำลองฐานข้อมูลเชิงวัตถุ หมายถึง แบบจำลองที่เกิดจากแนวคิดของการเขียนโปรแกรมเชิงวัตถุ มีการมองของทุกสิ่งเป็นวัตถุซึ่งเป็นแหล่งรวมของข้อมูล (Data) และปฏิบัติงาน มีคลาสเป็นตัวกำหนดคุณสมบัติของวัตถุและมีการปกปิดความลับของวัตถุ (สุจิตรา อุดุลย์เกษม, 2553, หน้า 52)

แบบจำลองฐานข้อมูลเชิงวัตถุหมายถึง แบบจำลองที่พัฒนาขึ้นมาเพื่อรองรับการเขียนโปรแกรมเชิงวัตถุ มองข้อมูลที่จะถูกจัดเก็บในฐานข้อมูลเป็นวัตถุ มีคลาสเป็นตัวกำหนดคุณสมบัติของวัตถุและมีการปกปิดความลับของข้อมูล (Encapsulation)

แบบจำลองฐานข้อมูลเชิงวัตถุ เป็นแบบจำลองที่รองรับฐานข้อมูลขนาดใหญ่มีโครงสร้างที่มีความสลับซับซ้อน ตัวอย่างเช่น งานด้านวิศวกรรมไฟฟ้าที่ต้องการจัดเก็บรายละเอียดที่มีความซับซ้อนที่แบบจำลองเชิงสัมพันธ์ไม่สามารถออกแบบเพื่อจัดเก็บได้อย่างครบถ้วน โดยฐานข้อมูลนี้จะมีการออกแบบการจัดเก็บข้อมูลและความสัมพันธ์ของข้อมูลของข้อมูลในรูปแบบของออบเจกต์ (Object) แต่ฐานข้อมูลนี้ยังถือว่าเป็นฐานข้อมูลเชิงสัมพันธ์ประเภทหนึ่ง โครงสร้างฐานข้อมูลเชิงวัตถุประกอบด้วย ออบเจกต์ (Object) เป็นส่วนที่จัดการกับข้อมูลที่ถูกจัดเก็บหรือตารางข้อมูลและภายในออบเจกต์จะมีรายละเอียดของออบเจกต์ที่เป็นโครงสร้างของการจัดเก็บ เช่น ออบเจกต์พนักงานประกอบด้วยโครงสร้างออบเจกต์คือ รหัสพนักงาน ชื่อพนักงาน ตำแหน่งเงินเดือน ที่อยู่ เป็นต้น ส่วนคลาส (Class) เป็นกลุ่มของออบเจกต์ที่ลักษณะและวิธีการที่เหมือนกันสิ่งที่ยึดเก็บในคลาสจะเป็นข้อมูล การติดต่อกับคลาสจะติดต่อผ่านเมธอด (Method) ซึ่งทำหน้าที่อธิบายวิธีการจัดเก็บข้อมูล โครงสร้างของแบบจำลองฐานข้อมูลเชิงวัตถุแสดงดังภาพที่ 2.6



ภาพที่ 2.6 โครงสร้างแบบจำลองฐานข้อมูลเชิงวัตถุ  
ที่มา สุจิตรา อุดุ้ยเกษม (2553, หน้า 258)

## 2.4 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

ฐานข้อมูลเชิงสัมพันธ์ เป็นฐานข้อมูลที่มีรูปแบบการจัดเก็บข้อมูลเป็นคอลัมน์และแถว ซึ่งมีวิธีการออกแบบฐานข้อมูลที่รองรับความสัมพันธ์ของข้อมูลทั้งรูปแบบความสัมพันธ์แบบ หนึ่งต่อหนึ่ง (1:1) หนึ่งต่อกลุ่ม (1:M) และกลุ่มต่อกลุ่ม (M:N) มีเครื่องมือช่วยในการพัฒนาระบบและมีโปรแกรมจัดการฐานข้อมูลที่รองรับฐานข้อมูลเชิงสัมพันธ์จำนวนมาก ที่นิยมใช้งานกันในปัจจุบัน เช่น MySQL, Oracle, MS-Access, Microsoft SQL Server ฯลฯ สามารถทำงานรองรับการพัฒนาโปรแกรม ทั้งแบบเว็บแอปพลิเคชันและโปรแกรมขนาดเล็กที่ทำงานกับเครื่องคอมพิวเตอร์เครื่องเดียว ฐานข้อมูลเชิงสัมพันธ์มีโครงสร้างของการจัดเก็บข้อมูลดังนี้

### 2.4.1 โครงสร้างของฐานข้อมูลเชิงสัมพันธ์ (Relational Data Structure)

โครงสร้างของฐานข้อมูลเชิงสัมพันธ์ เป็นการออกแบบการจัดเก็บข้อมูลที่อยู่ในรูปแบบตารางแบบ 2 มิติ คือแถวกับคอลัมน์ ข้อมูลทุกตารางจะเชื่อมความสัมพันธ์กัน ดังแสดงในภาพที่ 2.7 ดังนี้

Table Name: Product

P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Unit (หน่วยนับ)	Amount (จำนวน)	Price (ราคา)	Min (จุดสั่งซื้อ)
1	คัมภีร์ระบบฐานข้อมูล	เล่ม	85	249	5
2	Server2000	เล่ม	69	345	5
3	MS Access	เล่ม	74	299	5

Table Name: Supplier

Code (รหัสผู้ค้า)	Company (ชื่อบริษัท)	Add (ที่อยู่)	Tel (เบอร์โทร)
1001	Success Media	19/20 ซ. เจริญสุข เขตบางกอกน้อย กรุงเทพฯ 10700	0-2434-8270
1002	ศูนย์ส่งเสริมอาชีพศึกษา	89 ถ. มหารณพ เสาชิงช้า พระนคร กรุงเทพฯ 10200	0-2224-1129
1003	ซีเอ็ดยูเคชั่น	90 ถ. บางนา เขตบางนา กรุงเทพฯ 10260	0-2739-8000
1004	เอ-บู๊ค	99 ซ. ติวานนท์ 4 อ. เมือง จ. นนทบุรี 11000	0-2968-9337

Purchase

Pure_Id (รหัสใบสั่งซื้อ)	Code (รหัสผู้ค้า)	Pure_date (วันที่สั่งซื้อ)	Total_Amount (ยอดรวม)	Vat (ภาษี)	Net (สุทธิ)
4901	1001	15/02/59	4695	328.65	4366.35
4902	1003	15/02/59	7050	493.50	6556.50
4903	1004	20/02/59	8372	586.04	7785.96
4904	1003	11/03/59	2990	209.30	2,780.70

Purchase\_detail

Pure_Id (รหัสใบสั่งซื้อ)	P_Code (รหัสสินค้า)	Qty (จำนวน)	P_PerUnit (ราคา)	Total (รวม)
4901	1	5	249	1245
4901	2	10	345	3450
4902	2	30	235	7050
4903	3	18	299	5382
4904	1	15	245	3675

ภาพที่ 2.7 ระบบฐานข้อมูลของระบบการสั่งซื้อหนังสือ

จากภาพที่ 2.7 เป็นฐานข้อมูลของระบบการสั่งซื้อหนังสือ ประกอบด้วยตาราง Product เก็บรายละเอียดของสินค้า ตาราง Supplier เก็บข้อมูลคู่ค้า ตาราง Purchase เก็บรายละเอียดการสั่งซื้อ และ ตาราง Purchase\_detail เก็บรายละเอียดของสินค้าที่สั่งซื้อ โดยตาราง Product มีความสัมพันธ์กับตาราง Purchase\_detail ผ่านคีย์นอก (Foreign Key) คือ ฟیلด์ P\_code ในตาราง Purchase\_detail

ฐานข้อมูลเชิงสัมพันธ์มีส่วนประกอบของโครงสร้างการจัดเก็บ ดังนี้

### 1. รีเลชัน (Relation)

รีเลชัน หมายถึง ตารางที่ใช้จัดเก็บข้อมูล (Table) ประกอบด้วยคอลัมน์ (Column) กับแถว (Row) เพื่ออธิบายถึงรายละเอียดของข้อมูลที่จัดเก็บไว้ในรีเลชัน ให้สามารถชี้เฉพาะเจาะจงข้อมูลที่จัดเก็บไว้ได้ ตัวอย่างของรีเลชัน อาทิ รีเลชันบุคคล สิ่งของ สถานที่ หรือเหตุการณ์ ฯลฯ การออกแบบรีเลชันที่จะต้องครอบคลุมถึงข้อมูลทั้งหมดของรีเลชัน เพื่อให้ครบถ้วนของข้อมูลที่จะนำไปใช้ประโยชน์ รีเลชันมีส่วนประกอบ 3 ส่วน ดังนี้

1) แถว (Rows) หมายถึง รายการข้อมูลที่ใน 1 แถวที่ได้เพิ่มข้อมูลเข้าไปในแต่ละฟیلด์หรือบางครั้งจะเรียกว่าเรคอร์ด (Records) ตัวอย่างข้อมูลในรีเลชัน Product ประกอบด้วยข้อมูลจำนวน 4 แถว โดยที่ข้อมูลในแต่ละแถวจะต้องไม่ซ้ำซ้อนกัน ตัวอย่างดังภาพที่ 2.8

Table Name: Product

Column/Field					
P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Unit (หน่วยนับ)	Amount (จำนวน)	Price (ราคา)	Min (จุดสั่งซื้อ)
1	คัมภีร์ระบบฐานข้อมูล	เล่ม	85	249	5
2	Server2000	เล่ม	69	345	5
3	MS Access	เล่ม	74	299	5
4	Windows XP	เล่ม	50	299	5

Domain  
Min >=5

Rows/Records

ภาพที่ 2.8 ตาราง Product

จากภาพที่ 2.8 เป็นตาราง Product เป็นตารางที่ใช้เก็บข้อมูลของหนังสือ จะเก็บรายละเอียดของหนังสือมีข้อมูลที่บันทึกลงไป 4 แถว

2) คอลัมน์ (Column) หมายถึง รายละเอียดของการจัดเก็บข้อมูลในรีเลชัน คอลัมน์จะมีชื่อเรียกด้วยกันหลายชื่อ คือ ฟیلด์ (Field) และแอตทริบิวต์ (Attribute) เป็นต้น ในหนึ่งตารางจะประกอบด้วยหลายคอลัมน์ ดังภาพที่ 2.8 ตารางหนังสือ (Product) ประกอบด้วยฟیلด์ P\_code,

P\_name, Unit, Amount, Price และ Min โดยมีฟิลด์ P\_code เป็นฟิลด์ที่จัดเก็บข้อมูลไม่ซ้ำกัน ในแถวเรียกว่า คีย์หลัก (Primary key) จะแสดงค่าความเป็นเอกลักษณ์ของข้อมูลที่ไม่ซ้ำซ้อนกันในแต่ละแถวเก็บข้อมูลในตาราง ส่วนจำนวนฟิลด์ในตารางจะเรียกว่า ดีกรี (Degree) ซึ่งในตารางหนังสือ (Product) จะมี 6 ดีกรี เป็นต้น

3) โดเมน (Domain) หมายถึง ขอบเขตหรือเงื่อนไขของข้อมูลที่จะสามารถบันทึกข้อมูลลงไปในฟิลด์ การกำหนดเงื่อนไขกับฟิลด์ใด ๆ จะได้จากขั้นตอนที่ไปวิเคราะห์ปัญหาและค้นพบเงื่อนไขของการบันทึกข้อมูลบางฟิลด์ ถึงจะกำหนดให้ฟิลด์นั้นเป็นโดเมน เช่น ในตารางหนังสือ (Product) มีการกำหนดโดเมนให้กับฟิลด์ Min ซึ่งเป็นฟิลด์ที่เก็บข้อมูลจุดสั่งซื้อว่าจะต้องมีจุดสั่งซื้อหนังสือต้องไม่น้อยกว่าหรือเท่ากับ 5 เล่ม ถ้ามีการบันทึกข้อมูลในฟิลด์นี้เป็น 4 เล่ม ก็ไม่สามารถบันทึกได้ นอกจากจะบันทึกตั้งแต่ 5 เล่มขึ้นไปตามเงื่อนไขที่กำหนด เป็นต้น

ในการกำหนดโดเมนให้กับฟิลด์ ขึ้นอยู่กับเงื่อนไขของแต่ละฟิลด์ซึ่งอาจจะไม่เหมือนกันก็ได้ เช่น ในการลงทะเบียนนักศึกษาจำนวนหน่วยกิตรวมต้องไม่เกิน 22 หน่วยกิต ซึ่งเป็นเงื่อนไขทางมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ได้กำหนดไว้ แต่ก็ไม่ได้หมายความว่าทุกมหาวิทยาลัยจะมีเงื่อนไขเหมือนกัน บางมหาวิทยาลัยอาจจะลงทะเบียนเรียนได้ไม่เกิน 18 หน่วยกิต หรือ 21 หน่วยกิตก็ได้

## 2. คุณสมบัติของรีเลชัน (Properties of Relations)

ในการออกแบบรีเลชันเพื่อจัดเก็บข้อมูลในฐานข้อมูลเชิงสัมพันธ์ รีเลชันจะต้องมีคุณสมบัติดังนี้

1) ทุกรีเลชันในฐานข้อมูลเชิงสัมพันธ์จะมีชื่อกำกับเสมอ และชื่อของรีเลชันจะต้องไม่ซ้ำกัน เช่น ในฐานข้อมูลของการสั่งซื้อหนังสือประกอบด้วย รีเลชันหนังสือ (Product) รีเลชันผู้ค้า (Supplier) รีเลชันการสั่งซื้อ (Purchase) และรีเลชันรายละเอียดการสั่งซื้อ (P\_order) เป็นต้น วิธีการตั้งชื่อรีเลชันก็จะคล้ายกับการตั้งชื่อตัวแปรในการเขียนโปรแกรมคือ ห้ามมีช่องว่างและถ้ามีให้ใช้ขีดกลางแทน ห้ามใช้ตัวเลขขึ้นก่อนตัวอักษร เป็นต้น

2) การตั้งชื่อฟิลด์จะไม่นิยมตั้งชื่อให้ซ้ำกันถึงแม้จะไม่ได้อยู่ในรีเลชันเดียวกันก็ตาม ใน 1 ฟิลด์จะมีข้อมูลที่จัดเก็บไว้เพียงข้อมูลเดียว เช่น ฟิลด์รหัสสินค้า ก็จะเก็บเฉพาะข้อมูลรหัสสินค้าเท่านั้น จะไม่เก็บชื่อสินค้าไว้รวมกันจะต้องแยกเก็บฟิลด์ชื่อสินค้าไว้ต่างหาก แต่ยกเว้นบางฟิลด์ที่มีการจัดเก็บไว้รวมกันเช่น ฟิลด์ชื่อ ซึ่งบางครั้งคนที่ออกแบบอาจจะมองว่าเป็นการเก็บชื่อกับสกุลไว้ในฟิลด์เดียวกัน ขึ้นอยู่กับผู้ออกแบบว่าจะมองเห็นว่าชื่อกับสกุลเป็นข้อมูลเดียวกัน แต่บางครั้งก็มีการแยกเก็บคนละฟิลด์เพื่อสะดวกต่อการค้นหาข้อมูล

3) ลำดับของฟิลด์ในตาราง ไม่จำเป็นต้องเรียงลำดับฟิลด์อะไรมาก่อนและมาหลังก็ได้ เพราะเมื่อสร้างฟิลด์ลงไปในตารางและบันทึกลงฐานข้อมูลแล้ว เวลาผู้ใช้เรียกดูจะสามารถ

เรียงลำดับได้เองจากการใช้คำสั่ง SQL ในการเรียกดูข้อมูล ดังภาพที่ 2.9 จะเห็นว่าฟิลด์ถึงแม้ไม่ได้ถูกเรียงลำดับแต่สามารถให้ผลลัพธ์ที่เหมือนกัน

ตาราง Product

P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Unit (หน่วยนับ)	Amount (จำนวน)	Price (ราคา)	Min (จุดสั่งซื้อ)
1	คอมพิวเตอร์ระบบฐานข้อมูล	เล่ม	85	249	5
2	Server2000	เล่ม	69	345	5

ตาราง Product

P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Price (ราคา)	Min (จุดสั่งซื้อ)	Unit (หน่วยนับ)	Amount (จำนวน)
1	คอมพิวเตอร์ระบบฐานข้อมูล	249	5	เล่ม	85
2	Server2000	345	5	เล่ม	69

ภาพที่ 2.9 ตัวอย่างตารางสินค้า (Product) ฟิลด์ไม่จำเป็นต้องเรียงลำดับ

4) ในทุกริเลชันจะมีคีย์หลัก (Primary Key) เพื่อไม่ทำให้ข้อมูลในแต่ละแถวมีค่าซ้ำกันเกิดขึ้น ข้อมูลที่บันทึกลงไปในตารางจะไม่สามารถซ้ำซ้อนกันได้จากคุณสมบัติของการกำหนดคีย์หลัก ซึ่งคีย์หลักอาจจะเป็น 1 ฟิลด์ หรือหลาย ๆ ฟิลด์รวมกันก็ได้ตัวอย่างของคีย์หลักแสดงดังภาพที่ 2.9 ตารางสินค้า (Product) มีฟิลด์ P\_code มีคุณสมบัติเป็นคีย์หลัก

5) ข้อมูลที่บันทึกลงไปในแต่ละแถวไม่จำเป็นต้องจัดเรียงลำดับก่อน สามารถบันทึกข้อมูลลงไปได้เลย เช่น ในตารางข้อมูลสินค้า (Product) การบันทึกข้อมูลสินค้าลงไปในตาราง ไม่จำเป็นต้องบันทึกข้อมูลของสินค้ารหัสสินค้า 1 ก่อนรหัสสินค้า 2 สามารถบันทึก รหัสสินค้า 2 และ 4 ก่อนบันทึกรหัสสินค้าที่ 1 ก่อนได้เลย ดังภาพที่ 2.10

P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Unit (หน่วยนับ)	Amount (จำนวน)	Price (ราคา)	Min (จุดสั่งซื้อ)
2	Server2000	เล่ม	69	345	5
4	WindowsXP	เล่ม	50	299	5
1	คอมพิวเตอร์ระบบฐานข้อมูล	เล่ม	85	249	5
3	MS Access	เล่ม	74	299	5

ภาพที่ 2.10 ตัวอย่างข้อมูลที่บันทึกในตารางไม่จำเป็นต้องจัดเรียงลำดับ



### 2.4.2 ประเภทของคีย์ (Keys)

ในฐานข้อมูลเชิงสัมพันธ์ ข้อมูลจะถูกจัดเก็บในรูปแบบของตาราง แบบ 2 มิติ คือมีคอลัมน์กับแถว ข้อมูลในแต่ละแถวจะมีการจัดเก็บไม่ซ้ำซ้อนกัน ดังนั้นจึงมีคีย์เพื่อเป็นตัวระบุความเป็นเอกลักษณ์ (Uniquely) ของข้อมูลในแต่ละแถว และคีย์อื่น ๆ ที่ใช้งานในฐานข้อมูลเชิงสัมพันธ์มีหลายชนิดด้วยกัน คือ

1. คีย์หลัก (Primary Key) เป็นคีย์ที่ถูกกำหนดขึ้นมาเพื่ออ้างอิงหรือกำหนด (Determine) ข้อมูลในแต่ละแถวและทำให้ข้อมูลในแต่ละแถวที่จัดเก็บไว้ในตารางเก็บข้อมูลนั้นไม่ซ้ำกัน ใน 1 รีเลชัน อาจมีคีย์หลัก 1 แอตทริบิวต์หรือหลายแอตทริบิวต์รวมกันเป็นคีย์หลักก็ได้ ถ้าคีย์หลักที่มีการนำเอาแอตทริบิวต์ตั้งแต่ 2 ตัวมาประกอบกันจะเรียกว่า คีย์ผสม (Composite Key) และคีย์หลักจะเป็นค่าว่าง (Null) ไม่ได้ ตัวอย่างเช่น ในตารางสินค้า (Product) จะมี 1 แอตทริบิวต์เป็นคีย์หลักคือ แอตทริบิวต์ P\_code ส่วนในตารางรายละเอียดการสั่งซื้อ (Purchase\_detail) ถ้าใช้ Pure\_Id เพียงแอตทริบิวต์เดียวเป็นคีย์หลักไม่ได้เพราะจะทำให้ข้อมูลในแต่ละแถวยังซ้ำซ้อนกันอยู่ จึงต้องใช้ Pure\_Id และ P\_code รวมกันเป็นคีย์หลัก ถึงจะทำให้ข้อมูลไม่ซ้ำซ้อนกัน ดังตัวอย่างที่ 2.11

ตาราง Product

P_code (รหัสสินค้า)	P_name (ชื่อสินค้า)	Unit (หน่วยนับ)	Amount (จำนวน)	Price (ราคา)	Min (จุดสั่งซื้อ)
1	คัมภีร์ระบบฐานข้อมูล	เล่ม	85	249	5
2	Server2000	เล่ม	69	345	5
3	MS Access	เล่ม	74	299	5
4	WindowsXP	เล่ม	50	299	5

ก. ตาราง Product ใช้แอตทริบิวต์ 1 ตัวเป็นคีย์หลัก คือ P\_code

ตาราง Purchase\_detail

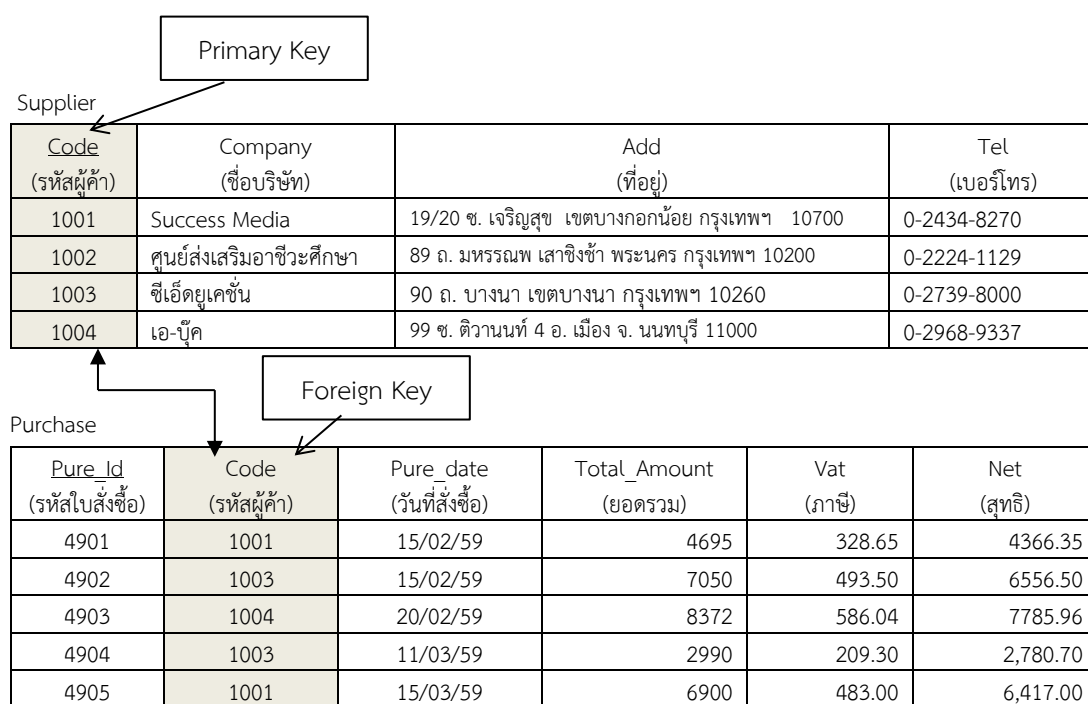
Pure_Id	P_Code	Qty	P_PerUnit	Total
4901	1	5	249	1245
4901	2	10	345	3450
4902	2	30	235	7050
4903	3	18	299	5382
4903	4	10	299	2990

ข. ตาราง Purchase\_detail ใช้แอตทริบิวต์ 2 ตัวเป็นคีย์หลัก คือ Pure\_Id และ P\_Code

**ภาพที่ 2.11** ตัวอย่างการกำหนดคีย์หลักแบบ 1 แอตทริบิวต์และ 2 แอตทริบิวต์

2. ซูเปอร์คีย์ (Super Key) เป็นแอตทริบิวต์หรือกลุ่มของแอตทริบิวต์ที่ทำให้ข้อมูลเป็นเอกลักษณ์ ไม่ซ้ำกันในแต่ละแถวของรีเลชัน ซึ่งจะมีลักษณะเหมือนกับคีย์หลัก (Primary Key) ดังภาพที่ 2.12 ตาราง สินค้า (Product) จะมี P\_code เป็นซูเปอร์คีย์

3. คีย์นอก (Foreign Key) เกิดจากคีย์หลักของรีเลชันหนึ่งแล้วนำมาใส่ไว้ในอีกรีเลชันหนึ่งเพื่อเชื่อม 2 รีเลชันให้เกิดความสัมพันธ์กัน ฐานข้อมูลจะมีการจัดเก็บข้อมูลหลายรีเลชันที่มีความสัมพันธ์กัน การเชื่อมความสัมพันธ์ระหว่างรีเลชันที่มีความสัมพันธ์กันจะเชื่อมความสัมพันธ์ผ่านคีย์นอก คือต้องเอาคีย์หลักจากรีเลชันหนึ่งไปไว้ในอีกรีเลชันหนึ่งนั่นเอง ตัวอย่างเช่น ในตารางผู้ค้า (Supplier) กับ ตารางการสั่งซื้อ (Purchase) มีความสัมพันธ์กันแบบหนึ่งต่อกลุ่ม (1:M) จึงเอาคีย์หลักจากรายการผู้ค้า คือ รหัสผู้ค้า (Code) ไปไว้ในตารางการสั่งซื้อ เพื่อเชื่อมความสัมพันธ์ระหว่างรายการผู้ค้ากับตารางการสั่งซื้อเข้าด้วยกัน ดังนั้น รหัสผู้ค้า (Code) ซึ่งเป็นคีย์หลักในรีเลชันผู้ค้าเมื่อไปอยู่ในรีเลชันการสั่งซื้อจึงเป็นคีย์นอกนั่นเอง ดังตัวอย่างภาพที่ 2.12



ภาพที่ 2.12 แสดงความสัมพันธ์ระหว่างรีเลชันผ่านคีย์นอก (Foreign Key)

4. คีย์คู่แข่ง (Candidate Key) เป็นคีย์ที่สามารถระบุความแตกต่างของข้อมูลในแต่ละแถวทำให้ข้อมูลไม่ซ้ำกัน ซึ่งมีคุณสมบัติที่จะเป็นคีย์หลัก (Primary Key) แต่ไม่ได้เลือกให้เป็นคีย์หลัก อาจจะด้วยไม่ได้ใช้ในการอ้างอิงถึงข้อมูลในระบบฐานข้อมูลนั้น ตัวอย่างเช่น ในตาราง

Student มีแอตทริบิวต์ที่ไม่ซ้ำกันเป็นคีย์คู่แข่ง 3 แอตทริบิวต์ คือ รหัสนักศึกษา, เลขที่บัตรประชาชน และ เบอร์โทร ดังตัวอย่างที่ ภาพที่ 2.13

**ตาราง Student**

Primary Key		Candidate Key		Candidate Key	
↓		↓		↓	
รหัสนักศึกษา	ชื่อ-สกุล	เลขที่บัตรประชาชน	เพศ	ที่อยู่	เบอร์โทร
59272001	ลักขณาพร สุริยะ	3440801568101	หญิง	นครราชสีมา	0865813211
59242002	การัน สุขเกษม	3440129201003	ชาย	ขอนแก่น	0912451200
59242003	วิไลพร รักดี	3440245100109	หญิง	นครราชสีมา	0861115666
59242004	รดา กองโคกกรวด	3104022100112	หญิง	บุรีรัมย์	0916651250

ภาพที่ 2.13 แสดงตาราง Student ที่มีคีย์คู่แข่ง 3 แอตทริบิวต์

จากภาพที่ 2.13 ตาราง Student ซึ่งมี 3 แอตทริบิวต์ที่มีคุณสมบัติเป็นค่าเอกลักษณ์ คือ รหัสนักศึกษา, เลขที่บัตรประชาชน และ เบอร์โทร โดยแอตทริบิวต์ รหัสนักศึกษาถูกเลือกให้เป็นคีย์หลักเนื่องจากใช้อ้างอิงข้อมูลในตารางของระบบฐานข้อมูล ส่วนแอตทริบิวต์ เลขที่บัตรประชาชนและเบอร์โทร ซึ่งไม่ถูกเลือกให้เป็นคีย์หลักก็จะเป็นคีย์คู่แข่ง หรือจะเรียกอีกชื่อว่าเป็น คีย์รอง (Alternate Key) นั่นเอง

## 2.5 กฎ 12 ข้อของ อี เอฟ คอดด์ (E.F. Codd 12 Rules)

ในปี ค.ศ. 1985 ดร. อี เอฟ คอดด์ (Dr. Edgar Frank Codd) ผู้ให้กำเนิดแบบจำลองข้อมูลเชิงสัมพันธ์ได้วางกฎไว้สำหรับเป็นบรรทัดฐานขั้นต่ำของการทำงานของระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Database Management System) ต่าง ๆ เพื่อให้เป็นระบบจัดการฐานข้อมูลสามารถรองรับการทำงานของฐานข้อมูลเชิงสัมพันธ์โดยสมบูรณ์ (Peter Rob and Carlos Coronel, 2009, P. 91) ได้กำหนดเป็นกฎ 12 ข้อ (แต่มีจริง 13 ข้อ) ของการออกแบบฐานข้อมูลเชิงสัมพันธ์ดังนี้

1. กฎข้อที่ 0 กฎพื้นฐาน (Foundation Rule) ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ ต้องมีความสามารถในการจัดการฐานข้อมูลทั้งหมดโดยใช้ความสามารถเชิงสัมพันธ์ กล่าวคือ การจัดการข้อมูลใด ๆ ในฐานข้อมูลเชิงสัมพันธ์จะต้องอ้างอิงถึงทฤษฎีเชิงสัมพันธ์เท่านั้น

2. กฎข้อที่ 1 กฎสารสนเทศ (Information Rule) สารสนเทศทั้งหมดที่อยู่ในฐานข้อมูลเชิงสัมพันธ์ ในมุมมองเชิงตรรกะจะต้องแสดงในรูปแบบตารางแบบ 2 มิติที่ประกอบด้วยคอลัมน์และแถว

3. กฎข้อที่ 2 กฎการรับประกันการเข้าถึงข้อมูล (Guaranteed Access Rule) ข้อมูลทุก ๆ ค่าที่เก็บไว้ในฐานข้อมูลจะต้องสามารถเข้าถึงได้โดยการระบุชื่อตาราง คีย์หลักและชื่อคอลัมน์

4. กฎข้อที่ 3 กฎรับรองค่าว่าง (Systematic Null Value Support) ระบบจัดการฐานข้อมูลต้องรองรับค่าว่าง หรือ null value โดยค่าดังกล่าวต้องแสดงว่าข้อมูลยังคงว่าเปล่าอยู่ต่างจากค่า 0 หรือช่องว่าง ซึ่งค่าว่างจะต้องถูกปฏิบัติอย่างมีแบบแผนให้แตกต่างจากค่าของข้อมูลชนิดอื่นและมีความเป็นอิสระจากชนิดข้อมูล

5. กฎข้อที่ 4 กฎการออนไลน์แคตตาล็อกที่อยู่บนพื้นฐานของแบบจำลองเชิงสัมพันธ์ (Dynamic On-line Catalog Based on the Relation Model) โครงสร้างของรีเลชันสามารถเรียกดูได้ ทุก ๆ ตารางและคอลัมน์ที่จัดเก็บอยู่ในระบบจัดการฐานข้อมูล จะต้องสามารถเรียกแสดงโครงสร้างและจัดการแก้ไขได้โดยภาษามาตรฐานที่ใช้จัดการกับฐานข้อมูลเชิงสัมพันธ์ และผู้ใช้งานที่ใช้งานข้อมูลจะอาศัยโครงสร้างเดียวกันนี้ในการใช้งานฐานข้อมูล

6. กฎข้อที่ 5 กฎด้านภาษาข้อมูล (Comprehensive data Sublanguage rule) จะต้องมีการอย่างน้อยหนึ่งภาษาที่ออกแบบมาสนับสนุนการทำงานกับฐานข้อมูลเชิงสัมพันธ์อย่างเต็มความสามารถของแบบจำลอง โดยความสามารถที่ภาษาจะต้องรองรับได้แก่

- 1) การนิยามข้อมูล
- 2) การนิยามวิว
- 3) การแก้ไขข้อมูล ได้แก่ การเพิ่ม ลบ และแก้ไข
- 4) การกำหนดกฎบูรณภาพหรือกฎเพื่อความคงความถูกต้องของข้อมูล
- 5) การกำหนดสิทธิ์ในการเข้าใช้ฐานข้อมูล
- 6) การจัดการทรานแซกชัน

7. กฎข้อที่ 6 กฎการปรับปรุงข้อมูลผ่านวิว (View Updating Rule) วิวหรือมุมมองการแสดงผลของผู้ใช้แต่ละคนมองเห็นจะต้องสามารถปรับปรุงได้ผ่านระบบได้เช่นกัน กล่าวคือ โดยปกติวิวจะเกิดจากรีเลชันหลักในฐานข้อมูล อาจเกิดจากหลาย ๆ รีเลชันมารวมกัน เมื่อมีการแก้ไขข้อมูลในวิว ข้อมูลที่สัมพันธ์กับข้อมูลในวิว นั้น ก็ต้องได้รับการแก้ไขให้ตรงกันด้วย

8. กฎข้อที่ 7 กฎการแทรก การปรับปรุง และการลบข้อมูลระดับสูง (High-Level Insertion, Update and Deletion Rule) ระบบจัดการฐานข้อมูลไม่ใช่ว่าแค่มีการแสดงข้อมูลเท่านั้นจะต้องมีคำสั่งระดับสูงที่รองรับการเพิ่ม การแก้ไขหรือปรับปรุงข้อมูล และการลบข้อมูลได้ด้วย

9. กฎข้อที่ 8 เป็นกฎของความเป็นอิสระของข้อมูลระดับกายภาพ (Physical Data Independence Rule) โปรแกรมประยุกต์ใด ๆ ที่มีการใช้งานระบบจัดการฐานข้อมูลจะต้องไม่ได้รับผลกระทบจากการปรับเปลี่ยนข้อกำหนดด้านกายภาพ เช่น เมื่อมีการเปลี่ยนแปลงลำดับของไฟล์ใหม่ มีการสร้างดัชนีเข้าถึงไฟล์ใหม่ เป็นต้น

10. กฎข้อที่ 9 เป็นกฎของความเป็นอิสระของข้อมูลระดับตรรกะ (Logical Data Independence Rule) โปรแกรมอื่นที่ใช้งานระบบจัดการฐานข้อมูลจะต้องไม่ได้รับผลกระทบจากการปรับเปลี่ยนข้อกำหนดด้านตรรกะ เช่น เมื่อมีการเปลี่ยนโครงสร้างตารางข้อมูล

11. กฎข้อที่ 10 กฎความคงสภาพที่เป็นอิสระ (Integrity Independence Rule) ภาษาฐานข้อมูลต้องรองรับการกำหนดข้อกำหนด หรือกฎต่าง ๆ ที่บังคับให้ข้อมูลมีความถูกต้องตามข้อกำหนด ข้อกำหนดเหล่านี้จะถูกจัดเก็บในระบบจัดการฐานข้อมูลและไม่สามารถที่จะละเมิดข้อกำหนดนี้ได้

12. กฎข้อที่ 11 กฎการกระจายที่เป็นอิสระ (Distribution Independence Rule) ผู้ใช้โปรแกรมประยุกต์จะไม่ได้รับผลกระทบใดๆ ถึงแม้จะย้ายที่เก็บฐานข้อมูลไปไว้แบบไหน อย่างไร หรือจะกระจายการจัดเก็บข้อมูลไปไว้พื้นที่ต่างๆ บนระบบเครือข่ายคอมพิวเตอร์ก็ตาม

13. กฎข้อที่ 12 กฎปราศจากเวอร์ชันย่อย (No Subversion Rule) ถ้าระบบสนับสนุนการเข้าถึงข้อมูลด้วยภาษาระดับต่ำ ในที่นี้หมายถึงภาษาที่จัดการข้อมูลครั้งละข้อมูล เช่น ครั้งละระเบียน ครั้งละคอลัมน์ในระเบียน ฯลฯ จะต้องไม่สามารถเลี่ยงกฎต่างๆ ที่ตั้งไว้เพื่อบูรณาการของข้อมูลได้

## สรุป

สถาปัตยกรรมระบบฐานข้อมูล แบ่งออกเป็น 3 ระดับด้วยกันคือ

1. ระดับภายใน (Internal Level) เป็นมุมมองที่ฐานข้อมูลเก็บอยู่จริงภายในเครื่องซึ่งจะเกี่ยวข้องกับรูปแบบการจัดเก็บข้อมูล วิธีการเข้าถึงข้อมูล ประสิทธิภาพในการทำงานของฐานข้อมูล ผู้บริหารฐานข้อมูลจะเข้าถึงข้อมูลผ่านโปรแกรมระบบปฏิบัติการและโปรแกรมจัดการฐานข้อมูล
2. ระดับแนวคิด เป็นมุมมองที่ผู้บริหารฐานข้อมูลมองเห็นเป็นแผนภาพแบบจำลองฐานข้อมูล มองเห็นตาราง แอตทริบิวต์ และความสัมพันธ์ของตาราง
3. ระดับภายนอก เป็นมุมมองของผู้ใช้มองเห็นข้อมูลที่ตัวเองต้องการ ซึ่งแต่ละคนอาจมีความต้องการใช้ข้อมูลที่ไม่เหมือนกันก็ได้

เหตุผลที่ต้องมีการแบ่งสถาปัตยกรรมออกเป็น 3 ระดับ ก็เพื่อให้ข้อมูลที่ถูกรับเก็บอยู่ในฐานข้อมูลมีความเป็นอิสระจากกัน หมายความว่า ถ้าข้อมูลที่อยู่ในระดับล่างได้รับการปรับปรุงแก้ไขใหม่ก็并不会ส่งผลกระทบต่อการทำงานของระดับที่อยู่สูงขึ้นไปนั่นเอง

แบบจำลองฐานข้อมูล (Data Model) เป็นแผนภาพที่นำมาใช้อธิบายถึงข้อมูลที่ถูกจัดเก็บในฐานข้อมูล เพื่อใช้สื่อสารกันระหว่างผู้ออกแบบและผู้ใช้ที่เกี่ยวข้องกับฐานข้อมูล ซึ่งมีแบบจำลองอยู่หลายแบบจำลองด้วยกัน ได้แก่ แบบจำลองแบบลำดับชั้น แบบจำลองแบบเครือข่าย แบบจำลองเชิงสัมพันธ์ แบบจำลองเชิงวัตถุ และแบบจำลองแบบมัลติไดเมนชัน

ฐานข้อมูลเชิงสัมพันธ์ เป็นฐานข้อมูลที่ออกแบบมาเพื่อรองรับการทำงานของฐานข้อมูลที่อยู่ในรูปแบบของตารางเก็บข้อมูล ประกอบด้วย แถว (Row) กับคอลัมน์ (Column) แต่ละตารางที่ถูกเก็บอยู่ในฐานข้อมูลจะต้องมีความสัมพันธ์ซึ่งกันและกัน ทั้งความสัมพันธ์แบบ หนึ่งต่อหนึ่ง (1:1) หนึ่งต่อกลุ่ม (1:M) และกลุ่มต่อกลุ่ม (M:N) นอกจากความสัมพันธ์แล้วในฐานข้อมูลเชิงสัมพันธ์ยังมีคีย์ (Keys) ประเภทต่างๆ ที่มีในตาราง เช่น Primary Key, Candidate Key, Foreign Key เป็นต้น

กฎ 12 ข้อของ อี เอฟ คอตต์ เป็นข้อกำหนดที่ถูกกำหนดขึ้นมาเพื่อให้เป็นบรรทัดฐานของการออกแบบฐานข้อมูลเชิงสัมพันธ์ จะต้องปฏิบัติตามกฎ 12 ข้อทุกครั้งที่ออกแบบฐานข้อมูลเชิงสัมพันธ์

### แบบฝึกหัดท้ายบท

1. จับคู่ความหมายของคำศัพท์ต่อไปนี้ โดยให้นำข้อคำศัพท์วางไว้หน้าข้อที่เห็นว่าถูกต้อง

1. เป็นคีย์ที่ถูกกำหนดขึ้นมาเพื่ออ้างอิง ข้อมูลในแต่ละแถวและทำให้ข้อมูลในแต่ละแถวที่จัดเก็บไว้ในตารางเก็บข้อมูลไม่ซ้ำกัน	ก. Degree
2. รายการข้อมูลทีใน 1 แถวที่ได้เพิ่มข้อมูลเข้าไปในแต่ละฟิลด์	ข. Table
3. เป็นคีย์ที่เป็นคีย์หลักจากรีเลชันอื่นแล้วนำมาใส่ไว้ในอีกรีเลชันหนึ่งเพื่อเชื่อม 2 รีเลชันนั้นให้เกิดความสัมพันธ์กัน	ค. Alternate key
4. จำนวนของฟิลด์ในตาราง	ง. Domain
5. ขอบเขตหรือเงื่อนไขของการบันทึกข้อมูลลงไปในฟิลด์ เช่น อายุ > 18 หรือ เงินเดือน > 0	จ. Foreign Key
6. คีย์หลักที่มีแอตทริบิวต์ตั้งแต่ 2 ตัวมาประกอบกันเป็น Primary Key	ฉ. Composite Key
7. จำนวนแถวที่มีอยู่ในตาราง	ช. Cardinality
8. การเปลี่ยนแปลงข้อมูลในระดับแนวคิดไม่ส่งผลกระทบต่อการทำงานของผู้ใช้ภายนอก	ซ. Logical Data Independence
9. คีย์ที่มีคุณสมบัติไม่ซ้ำเหมือนกับคีย์หลัก แต่ไม่ได้ถูกเลือกให้เป็นคีย์หลัก	ณ. Primary key
10. การจัดเก็บข้อมูลแบบ 2 มิติ ประกอบด้วยแถวกับคอลัมน์	ญ. Record

2. ตอบข้อคำถามต่อไปนี้

2.1 อธิบายความหมายของความเป็นอิสระต่อไปนี้

1. Physical Data Independence
2. Logical Data Independence

2.2 อธิบายแบบจำลองฐานข้อมูลแบบลำดับชั้น มีวิธีการจัดเก็บข้อมูลในรูปแบบใด

2.3 เหตุใดแบบจำลองฐานข้อมูลเชิงสัมพันธ์ถึงได้รับความนิยมใช้งานกันในปัจจุบัน

2.4 ระบบฐานข้อมูลเชิงสัมพันธ์มีโครงสร้างการจัดเก็บข้อมูลอย่างไร

2.5 ความเป็นอิสระของข้อมูลคืออะไร อธิบาย

2.6 ให้อธิบายถึงคุณสมบัติของรีเลชัน

2.7 แบบจำลองใดเหมาะกับการจัดเก็บข้อมูลขนาดใหญ่

2.8 ให้นักศึกษาอธิบายภาพสถาปัตยกรรมคอมพิวเตอร์แบ่งได้เป็นกี่ระดับ อะไรบ้าง อธิบายการวิธีการทำงานแต่ละระดับ

