บทที่ 3
แบบจำลองฐานข้อมูลเชิงสัมพันธ์

แบบจำลองฐานข้อมูลเชิงสัมพันธ์เป็นแหล่งรวมของแนวความคิดที่นำเสนอความเป็นจริงของวัตถุ ข้อมูล และเหตุการณ์ รวมถึงความสัมพันธ์ระหว่างข้อมูลที่มีความสอดคล้องตรงกันโดยจุดประสงค์ของแบบจำลองฐานข้อมูลคือ การนำแนวความคิดต่างๆ มานำเสนอให้เกิดเป็นรูปแบบจำลองขึ้นมา เพื่อใช้ในการสื่อสารระหว่างผู้ออกแบบฐานข้อมูลกับผู้ใช้ให้เกิดความเข้าใจตรงกัน (Date, 2013, p. 135)

เนื้อหาในบทนี้จะอธิบายถึง ความหมายของแบบจำลองฐานข้อมูลหรือตัวแบบของข้อมูล แบบจำลองฐานข้อมูล ประเภทของแบบจำลองฐานข้อมูล โครงสร้างฐานข้อมูลเชิงสัมพันธ์ ฐานข้อมูลเชิงสัมพันธ์ รีเลชั่น และคีย์ของข้อมูล

ความหมายของแบบจำลองฐานข้อมูลหรือตัวแบบของข้อมูล

โอภาส เอี่ยมสิริวงศ์ (2558, น. 78) ให้ความหมายของแบบจำลองฐานข้อมูลว่า หมายถึง แนวทางที่ใช้อธิบายแบบร่างเชิงตรรกะของข้อมูลและความสัมพันธ์ในส่วนต่างๆ ที่เกี่ยวข้องกัน แบบจำลองฐานข้อมูลเป็นแหล่งรวมของแนวความคิดที่นำเสนอความเป็นจริงของวัตถุ ข้อมูล และเหตุการณ์ รวมถึงความสัมพันธ์ระหว่างข้อมูลที่มีความสอดคล้องตรงกัน จุดประสงค์ของแบบจำลองฐานข้อมูลคือ การนำแนวความคิดต่างๆ มานำเสนอให้เกิดเป็นรูปแบบจำลองขึ้นมา เพื่อใช้สื่อสารระหว่างผู้ออกแบบฐานข้อมูลกับผู้ใช้ให้เกิดความเข้าใจตรงกัน

พนิดา พานิชกุล และ ณัฐพงษ์ วารีประเสริฐ (2552, น. 29) กล่าวว่า แบบจำลองฐานข้อมูล เป็นเครื่องมือที่ใช้อธิบายหรือแสดงให้เห็นข้อมูลทั้งหมดของระบบ ด้วยรูปภาพ สัญลักษณ์ หรือคำอธิบายต่างๆ ตามวิธีการสร้างแบบจำลองที่เลือกใช้ เนื่องจากแบบจำลองฐานข้อมูลแสดงให้เห็นข้อมูลของระบบด้วยรูปภาพ สัญลักษณ์ และคำอธิบายต่างๆ จึงสามารถใช้เป็นตัวแทนในการนำเสนอกระบวนการทางความคิดในการออกแบบฐานข้อมูลของนักออกแบบได้เป็นอย่างดี และยังจำลองให้เห็นสภาพจริงของข้อมูล ทั้งในด้านของโครงสร้างและความสัมพันธ์ระหว่างข้อมูล

ปราลี มณีรัตน์ (2552, 75) กล่าวว่า แบบจำลองข้อมูลคือแบบจำลองที่มีรูปแบบในการนำเสนอรายละเอียดต่างๆ ที่เกี่ยวข้องกับฐานข้อมูลให้เป็นมาตรฐานเดียวกัน เพื่อนำเสนอกับผู้ใช้ในแต่ละระดับ แต่ละมุมมอง ที่มองฐานข้อมูลแตกต่างกันออกไป โดยการใช้รูปภาพแสดงโครงสร้างของข้อมูล (data structure) ความสัมพันธ์ (relations) ข้อบังคับต่างๆ (constraints) และการแปลงรูปข้อมูล (transformation)

อาจสรุปได้ว่า แบบจำลองข้อมูลหมายถึง เทคนิคที่นำมาใช้จัดการโครงสร้างและสัมพันธ์ระหว่างข้อมูลในระบบ คำว่าแบบจำลองข้อมูลและแบบจำลองฐานข้อมูลส่วนใหญ่มักเรียกแทนกันเสมอ กล่าวคือ ในบางครั้งแบบจำลองข้อมูลอาจเรียกว่าแบบจำลองฐานข้อมูลก็ได้ เนื่องจากแบบจำลองข้อมูลที่สร้างขึ้นก็จะต้องนำไปพัฒนาในฐานข้อมูลนั่นเอง

แบบจำลองฐานข้อมูล

แบบจำลองฐานข้อมูลแบ่งออกเป็น 2 ประเภทคือ แบบจำลองเชิงแนวคิดและแบบจำลองเพื่อการนำไปใช้ โดยมีรายละเอียดดังต่อไปนี้ (Ramez & Shamkant, 2013, p. 127)

แบบจำลองเชิงแนวคิด (Conceptual Data Models) ใช้สำหรับแสดงลักษณะโดยรวมของข้อมูลทั้งหมดในระบบ โดยนำเสนอในลักษณะแผนภาพหรือไดอะแกรมที่ประกอบไปด้วยเอนทิตีต่างๆ และความสัมพันธ์ระหว่างเอนทิตีในระบบ จุดประสงค์ของแบบจำลองเชิงแนวคิดคือ ต้องการนำเสนอให้เกิดความเข้าใจระหว่างผู้ออกแบบและผู้ใช้งาน กล่าวคือ เมื่อเห็นแผนภาพแบบจำลองดังกล่าวจะทำให้เข้าใจถึงความสัมพันธ์ของข้อมูลต่างๆ ในระบบ แบบจำลองเชิงแนวคิดนี้ไม่ได้ขึ้นกับประเภทของ DBMS ดังนั้นจึงสามารถนำไปแปลงให้สอดคล้องกับ DBMS ที่เลือกใช้งานได้ในภายหลัง ตัวอย่างแบบจำลองเชิงแนวคิด เช่น แบบจำลอง E-R (Entity Relationship Model) ซึ่งจะกล่าวรายละเอียดต่อในบทที่ 5

แบบจำลองเพื่อการนำไปใช้ (Implementation Data Models) เป็นแบบจำลองที่ใช้อธิบายถึงโครงสร้างข้อมูลของฐานข้อมูล ด้วยการแสดงถึงรูปแบบที่อิงกับระบบจัดการฐานข้อมูลที่ใช้ โดยแบบจำลองเพื่อการนำไปใช้แบ่งออกเป็น 5 ประเภทตามรูปแบบของแต่ละโครงสร้างของฐานข้อมูล ซึ่งรายละเอียดจะกล่าวในหัวข้อแบบจำลองข้อมูล (Database Model) ต่อไป

พื้นฐานการสร้างแบบจำลองข้อมูลประกอบด้วยเอนทิตี แอตทริบิวต์ ความสัมพันธ์และข้อบังคับ (Hoffer, Prescott & McFadden, 2005, p. 127)

1. เอนทิตี

เอนทิตี (entities) คือ บุคคล สถานที่ สิ่งของ หรือเหตุการณ์ ที่เกี่ยวข้องกับข้อมูลที่รวบรวมไว้เพื่อการจัดเก็บ เอนทิตีถือเป็นตัวแทนของวัตถุในโลกแห่งความเป็นจริง ตัวอย่างเช่น เอนทิตีสินค้า นอกจากนี้แล้วเอนทิตียังอาจเป็นนามธรรม (abstractions) ก็ได้ เช่น เอนทิตีเที่ยวบิน เป็นต้น

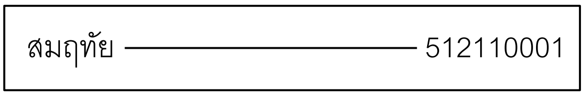
2. แอตทริบิวต์

แอตทริบิวต์ (attributes) คือ คุณลักษณะของเอนทิตี ตัวอย่างเช่น เอนทิตีลูกค้า ที่ประกอบด้วยแอตทริบิวต์รหัสลูกค้า ชื่อ-นามสกุล ที่อยู่ โทรศัพท์ วงเงินเครดิต อาจกล่าวได้ว่าแอตทริบิวต์คือฟิลด์ในระบบไฟล์นั่นเอง

3. ความสัมพันธ์

ความสัมพันธ์ (relationships) คือ ความสัมพันธ์ระหว่างเอนทิตี ตัวอย่างเช่น ความสัมพันธ์ระหว่างลูกค้ากับบัญชีธนาคาร เช่นลูกค้าสามารถเปิดบัญธนาคารได้หลายบัญชี เป็นต้น โดยแบบจำลองข้อมูลจะมีความสัมพันธ์อยู่ 3 ชนิดคือ ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (one-to-one) ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (one-to-many) และความสัมพันธ์แบบกลุ่มต่อกลุ่ม (many-to-many) โดยปกตินักออกแบบฐานข้อมูลมักใช้สัญลักษณ์ย่อเพื่อสะดวกต่อการใช้งานคือ 1:1, 1:M และ M:N ความแตกต่างด้านความสัมพันธ์ทั้งสามชนิดมีรายละเอียดดังต่อไปนี้

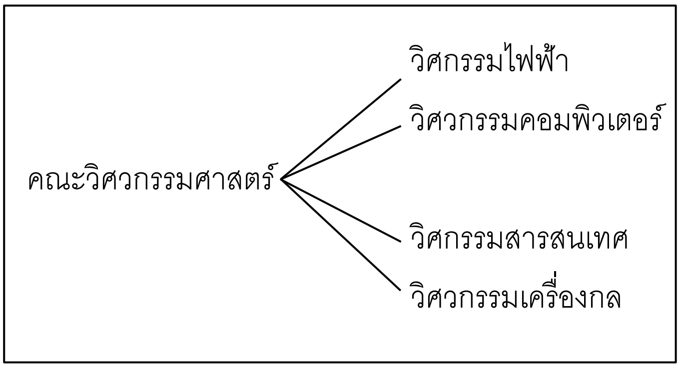
3.1 **ความสัมพันธ์แบบหนึ่งต่อหนึ่ง** (1:1) เป็นความสัมพันธ์ที่แต่ละรายการของเอนทิตี x มีความสัมพันธ์กับข้อมูล y เพียงหนึ่งรายการ ตัวอย่างเช่น นักศึกษามีรหัสประจำตัวได้เพียงหนึ่งรหัส และรหัสนักศึกษาหนึ่งรหัสจะนำไปอ้างอิงนักศึกษาได้เพียงหนึ่งคนเท่านั้น ดังแสดงในภาพที่ 3.1



ภาพที่ 3.1 แสดงความสัมพันธ์แบบ one-to-one

3.2 **ความสัมพันธ์แบบหนึ่งต่อกลุ่ม** (1:M) เป็นความสัมพันธ์ที่แต่ละรายการของเอนทิตี x มีความสัมพันธ์กับเอนทิตี y มากกว่าหนึ่งรายการ โดยแต่ละรายการของเอนทิตี y จะมี

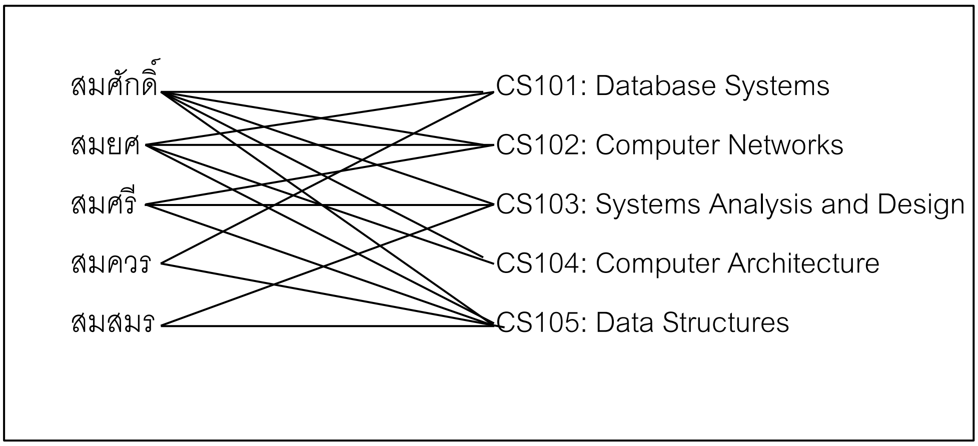
ความสัมพันธ์กับเอนทิตี x ได้เพียงหนึ่งรายการเท่านั้น ตัวอย่างเช่น คณะหนึ่งมีหลายสาขาวิชา ในขณะที่แต่ละสาขาวิชาจะต้องสังกัดเพียงหนึ่งคณะ



ภาพที่ 3.2 แสดงความสัมพันธ์แบบ one-to-many

จากภาพที่ 3.2 อธิบายได้ว่า คณะวิศวกรรมศาสตร์ประกอบด้วยหลายสาขาด้วยกัน คือ สาขาวิศกรรมไฟฟ้า วิศวกรรมคอมพิวเตอร์ วิศวกรรมสารสนเทศ และวิศวกรรมเครื่องกล ในขณะที่สาขาดังกล่าวจะสังกัดเพียงหนึ่งคณะเท่านั้นคือคณะวิศวกรรมศาสตร์

3.3 **ความสัมพันธ์แบบกลุ่มต่อกลุ่ม** (M:N) เป็นความสัมพันธ์ที่แต่ละรายการของเอนทิตี x มีความสัมพันธ์กับข้อมูล y มากกว่าหนึ่งรายการ ในขณะเดียวกันแต่ละรายการของเอนทิตี y ก็มีความสัมพันธ์กับข้อมูล x ได้มากกว่าหนึ่งรายการเช่นกัน ตัวอย่างเช่น นักศึกษาลงทะเบียนเรียนได้หลายวิชา โดยแต่ละวิชาก็มีนักศึกษาลงทะเบียนได้มากกว่าหนึ่งคน ดังแสดงในภาพที่ 3.3



ภาพที่ 3.3 แสดงความสัมพันธ์แบบ many-to-many

จากภาพที่ 3.3 อธิบายได้ว่า นักศึกษาชื่อสมศักดิ์ได้ลงทะเบียนเรียนครบทุกวิชา ซึ่งประกอบด้วยรหัส CS101, CS104 และ CS105 ในขณะที่รหัสวิชา CS101 ก็มีนักศึกษาหลายคนที่ลงทะเบียนเรียน ซึ่งประกอบด้วยสมศักดิ์ สมยศ และสมควร

4. ข้อบังคับ

ข้อบังคับ (constraints) คือ กฎเกณฑ์เพื่อการบรรจุข้อมูล ซึ่งกฎข้อบังคับนี้ถือว่าเป็นสิ่งสำคัญ เพราะจะช่วยให้เกิดความสอดคล้องตรงกันของข้อมูล ตัวอย่างเช่น เงินเดือนพนักงานจะต้องมีค่าระหว่าง 6,000 ถึง 150,000 บาท หรือนักศึกษาสามารถลงทะเบียนเรียนได้หลายหน่วยกิต แต่รวมแล้วต้องไม่เกิน 21 หน่วยกิต เป็นต้น

ประเภทของแบบจำลองฐานข้อมูล

แบบจำลองฐานข้อมูลแบ่งออกเป็น 5 รูปแบบคือ แบบจำลองฐานข้อมูลลำดับชั้น (hierarchical database model) แบบจำลองฐานข้อมูลเครือข่าย (network database model) แบบจำลองฐานข้อมูลเชิงสัมพันธ์ (relational database model) แบบจำลองฐานข้อมูลเชิงวัตถุ (object-oriented database model) และแบบจำลองฐานข้อมูลแบบมัลติไดเมนชั่น (multidimensional database model) ซึ่งจะอธิบายดังนี้ (Teorey, 2006, p.85)

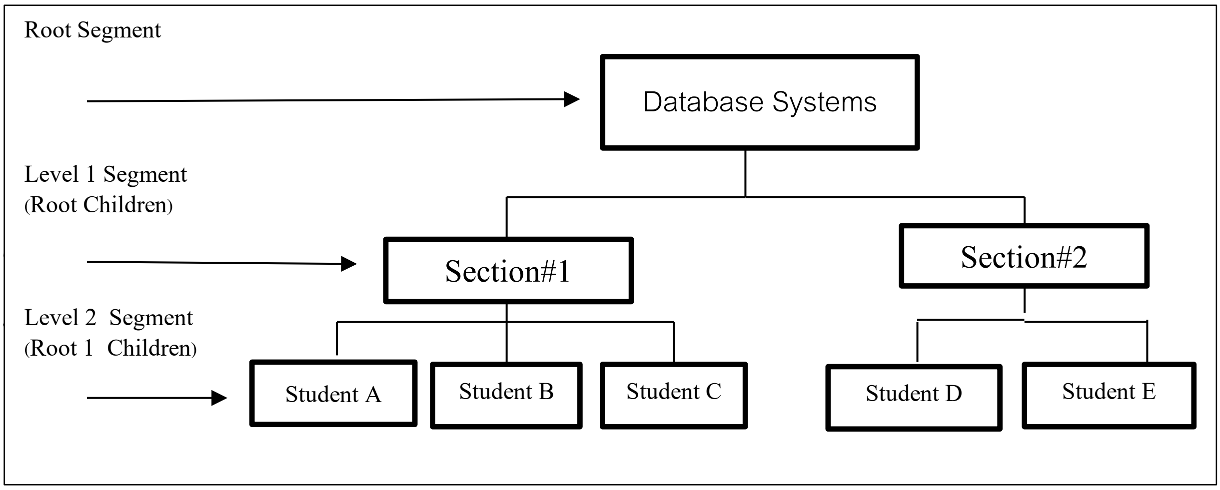
1. แบบจำลองฐานข้อมูลลำดับชั้น

แบบจำลองฐานข้อมูลลำดับชั้น (hierarchical database model) มีการจัดไฟล์ข้อมูลเป็นโครงสร้างแบบบนลงล่าง (top-down) ซึ่งมีลักษณะคล้ายกับโครงสร้างต้นไม้ (tree structure) ที่มีการสืบทอดแบบเป็นลำดับชั้น โหนดระดับสูงสุดจะเรียกว่าราก (root) และโหนดระดับล่างสุดจะเรียกว่าใบ (leaves)

โครงสร้างฐานข้อมูลลำดับชั้นจะเป็นลำดับหรือเรียกว่าเซกเมนต์ (segments) โดยเซกเมนต์เปรียบเสมือนกับเรคอร์ดในระบบแฟ้มข้อมูล แต่ละเซกเมนต์ที่อยู่ลำดับล่างลงไปก็คือลูกของเซกเมนต์ที่อยู่ลำดับก่อนหน้า ด้วยหลักการนี้แบบจำลองฐานข้อมูลชนิดนี้จึงมีความสัมพันธ์แบบ one-to-many ซึ่งโหนดพ่อสามารถแตกสาขาออกเป็นโหนดลูกได้หลายโหนด ในขณะที่โหนดลูกจะมีเพียงพ่อเดียวเท่านั้นจึงไม่รองรับความสัมพันธ์แบบ many-to-many

จากภาพที่ 3.4 จะเห็นได้ว่ามีคอร์สวิชา Database Systems เป็น root และมีการแตกสาขาด้วยการเปิดสอนวิชานี้อยู่ 2 section ด้วยกัน แต่ละ section ก็จะมีนักศึกษาหลายคน

ที่ลงทะเบียนเรียนคอร์สวิชาดังกล่าว ดังนั้น หากต้องการทราบว่า Section#2 ของวิชา Database Systems นี้มีนักศึกษาคนใดเรียนบ้างก็จะเข้าไปค้นหาตั้งแต่รากซึ่งอยู่บนสุดและไต่ลำดับลงมา ซึ่งเป็นรูปแบบการท่องไปยังลำดับชั้นถัดไปเรื่อยๆ นั่นเอง



ภาพที่ 3.4 แบบจำลองฐานข้อมูลลำดับชั้น (Hierarchical Database Model)
ที่มา : Teorey (2006, p.86)

แบบจำลองฐานข้อมูลลำดับชั้นเป็นสถาปัตยกรรมฐานข้อมูลที่เก่าแก่ที่สุด และเนื่องจากมีความสัมพันธ์แบบพ่อลูก ดังนั้น ความถูกต้องในข้อมูลย่อมมีความคงสภาพสูง แต่ในปัจจุบันไม่นิยมใช้กันแล้ว เนื่องจากความยากต่อการพัฒนาแอปพลิเคชันเพื่อใช้งานบนฐานข้อมูลชนิดนี้ และการปรับปรุงโครงสร้างมีความยืดหยุ่นต่ำ รวมทั้งเป็นโครงสร้างที่ไม่สามารถกำหนดความสัมพันธ์แบบกลุ่มต่อกลุ่มได้ สำหรับข้อดีและข้อเสียของแบบจำลองฐานข้อมูลชนิดนี้คือ

ข้อดี

1. มีรูปแบบโครงสร้างที่เข้าใจง่าย ซึ่งเป็นในลักษณะต้นไม้
2. มีโครงสร้างที่ซับซ้อนน้อยที่สุด เหมาะกับข้อมูลที่มีความสัมพันธ์แบบ one-to-many
3. ป้องกันความปลอดภัยในข้อมูลที่ดี เนื่องจากต้องอ่านข้อมูลที่เป็นต้นกำเนิดก่อน ทำให้ข้อมูลมีความคงสภาพที่ดี (data integrity)
4. เหมาะกับข้อมูลที่มีการเรียงลำดับแบบต่อเนื่อง

ข้อเสีย

1. ยากต่อการพัฒนา เพราะต้องมีความรู้ความเข้าใจถึงโครงสร้างทางกายภาพของข้อมูลที่จัดเก็บอยู่ภายในฐานข้อมูล

2. มีข้อจำกัดด้านการนำไปใช้ โดยเฉพาะไม่รองรับความสัมพันธ์แบบ many-to-many

3. เมื่อมีการเปลี่ยนแปลงโครงสร้าง แอปพลิเคชันโปรแกรมทั้งหมดต้องเปลี่ยนแปลงตาม เนื่องจากขาดอิสระในโครงสร้าง (structural independent)

4. ในการเรียกใช้ข้อมูลจำเป็นต้องผ่าน root เสมอ ดังนั้นหากต้องการค้นหาข้อมูลซึ่งอยู่ในระดับล่างๆ ก็จะต้องค้นหาทั้งแฟ้ม

5. ไม่มีภาษาที่ใช้สำหรับจัดการข้อมูล (DML) ใน DBMS

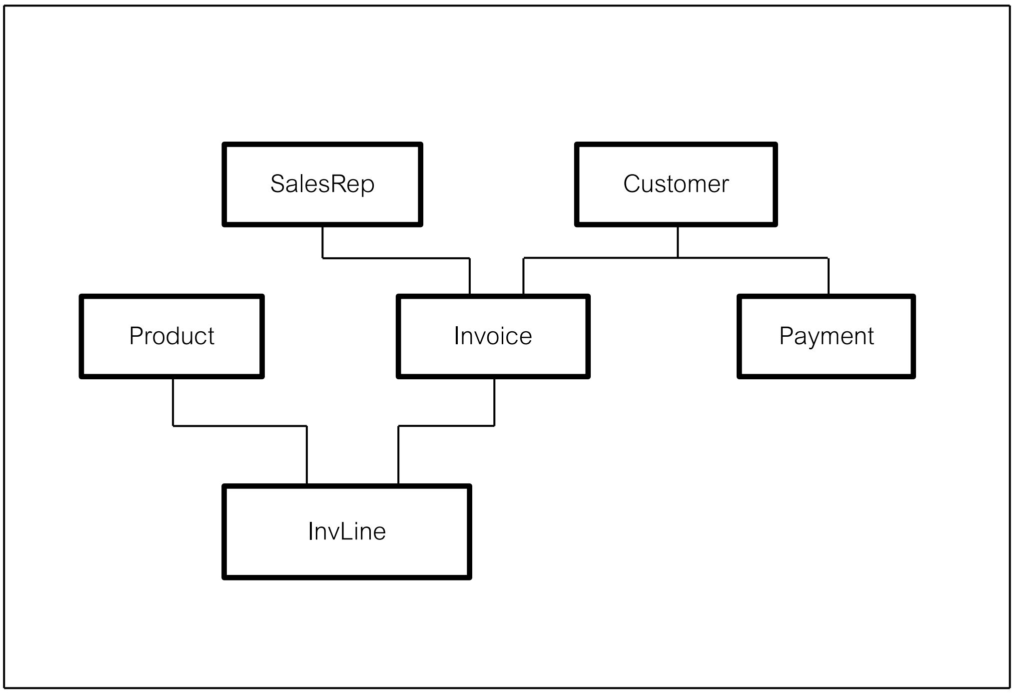
6. ขาดมาตรฐานการรองรับที่ชัดเจน

2. แบบจำลองฐานข้อมูลเครือข่าย

แบบจำลองฐานข้อมูลเครือข่าย (network database model) สามารถรองรับความสัมพันธ์ของข้อมูลที่มีความซับซ้อนยิ่งขึ้นและมีประสิทธิภาพสูงกว่าแบบลำดับชั้น แต่อย่างไรก็ตาม แบบโครงสร้างฐานข้อมูลเครือข่ายยังคงมีโครงสร้างที่คล้ายกับโครงสร้างแบบลำดับชั้น ซึ่งยังคงมีลำดับชั้นแบบบนลงล่าง แต่จะแตกต่างตรงที่แต่ละโหนดสามารถมีความสัมพันธ์กับโหนดอื่นๆ ได้หลายโหนด กล่าวคือ แต่ละโหนดสามารถมีหลายพ่อได้ซึ่งแตกต่างจากแบบจำลองฐานข้อมูลลำดับชั้นที่สามารถมีได้เพียงพ่อเดียวเท่านั้น ดังนั้นแบบจำลองนี้จึงมีความยืดหยุ่นที่สูงกว่าแบบแรก ดังตัวอย่างที่แสดงในภาพที่ 3.5

ความสัมพันธ์ระหว่างข้อมูลในแบบจำลองฐานข้อมูลเครือข่ายจะเรียกว่าเซต (set) แต่ละเซตประกอบด้วยเรคอร์ด 2 ชนิดด้วยกันคือ owner record ที่เปรียบเสมือนกับโหนดพ่อและ member record ที่เปรียบเสมือนกับโหนดลูก โดยตัวแทนของเซตที่มีความสัมพันธ์แบบ one-to-many จะสามารถเชื่อมโยงไปมาระหว่าง owner record และ member record ได้

ปัจจุบันแบบจำลองชนิดนี้ยังคงมีใช้งานอยู่บนเครื่องคอมพิวเตอร์ระดับเมนเฟรม โดยจะใช้ตัวชี้หรือพอยน์เตอร์เป็นตัวโยงความสัมพันธ์ระหว่างเรคอร์ด และสนับสนุนความสัมพันธ์ทั้งแบบ one-to-many และ many-to-many สำหรับข้อดีและข้อเสียของแบบจำลองฐานข้อมูลชนิดนี้คือ



ภาพที่ 3.5 แบบจำลองฐานข้อมูลเครือข่าย (Network Database Model)
ที่มา : Teorey (2006, p. 85)

ข้อดี

1. มีหลักการที่ง่าย ซึ่งใกล้เคียงกับแบบจำลองฐานข้อมูลลำดับชั้น
2. สนับสนุนความสัมพันธ์แบบ many-to-many
3. การเข้าถึงข้อมูลมีความยืดหยุ่นสูงกว่าแบบลำดับชั้นและระบบแฟ้มข้อมูล
4. ความสัมพันธ์แบบ owner / member relationship ทำให้ข้อมูลมีความคงสภาพที่ดี
5. มีภาษานิยามข้อมูล (DDI) และภาษาจัดการข้อมูล (DML) ใน DBMS
6. มีมาตรฐานเพื่อนำไปปฏิบัติชัดเจน

ข้อเสีย

1. ระบบโดยรวมยังซับซ้อน อีกทั้งมีข้อจำกัดด้านประสิทธิภาพ
2. ยากต่อการนำไปใช้ทั้งในด้านการพัฒนาแอปพลิเคชันและการจัดการ
3. หากโครงสร้างมีการเปลี่ยนแปลงแอปพลิเคชันโปรแกรมทั้งหมดต้องเปลี่ยนตาม

3. แบบจำลองฐานข้อมูลเชิงสัมพันธ์

แบบจำลองฐานข้อมูลเชิงสัมพันธ์ (relational database model) ถูกนำไปใช้อย่างแพร่หลายมากที่สุดในปัจจุบัน เนื่องจากแบบจำลองฐานข้อมูลเชิงสัมพันธ์นำเสนอมุมมองของข้อมูลในลักษณะตาราง (table) ซึ่งสามารถสื่อสัมพันธ์กับมนุษย์ได้เข้าใจง่าย ตารางหนึ่งประกอบด้วยแถวและคอลัมน์ ข้อมูลที่จัดเก็บอยู่ในตารางสามารถเชื่อมโยงความสัมพันธ์กับตารางต่างอื่นๆ ได้ไม่ว่าจะเป็นความสัมพันธ์แบบ one-to-many หรือแบบ many-to-many โดยจะใช้คีย์ในการอ้างอิงถึงตารางอื่นๆ ที่เกี่ยวข้อง ซึ่งคีย์ดังกล่าวเป็นได้ทั้งคีย์หลัก (primary key) และคีย์รอง (secondary key) รวมถึงการกำหนดลำดับดัชนี เพื่อเข้าถึงข้อมูลได้อย่างรวดเร็ว ดังแสดงภาพที่ 3.6

ปัจจุบันแบบจำลองฐานข้อมูลเชิงสัมพันธ์ได้พัฒนาเพื่อใช้งานบนโปรแกรมต่างๆ มากมาย รวมทั้งโปรแกรมระบบจัดการฐานข้อมูล (DBMS) ที่ส่วนใหญ่สนับสนุนการทำงานของแบบจำลองดังกล่าวด้วยการใช้ชุดคำสั่ง SQL ในการจัดการกับฐานข้อมูล สำหรับข้อดีและข้อเสียของแบบจำลองฐานข้อมูลชนิดนี้คือ

ข้อดี

1. มีความเป็นอิสระในโครงสร้าง โดยหากมีการเปลี่ยนแปลงโครงสร้างตาราง จะไม่ส่งผลต่อแอปพลิเคชันโปรแกรมที่ใช้งาน

2. การนำเสนอข้อมูลในรูปแบบของตาราง ก่อให้เกิดมโนภาพถึงข้อมูลที่จัดเก็บ ทำให้งานต่อการออกแบบฐานข้อมูล การนำไปใช้ และการจัดการ

3. การเรียกดูข้อมูลสามารถเรียกใช้ได้ด้วยชุดคำสั่ง SQL

4. มีระบบความปลอดภัยที่ดี เนื่องจากโครงสร้างนี้ผู้ใช้งานจะไม่ทราบถึงกระบวนการจัดเก็บข้อมูลภายในฐานข้อมูลแท้จริงว่าเป็นอย่างไร

5. DBMS ที่พัฒนาขึ้นมาในปัจจุบันล้วนรองรับเทคโนโลยีฐานข้อมูลเชิงสัมพันธ์

ข้อเสีย

1. จำเป็นต้องเสียค่าใช้จ่ายในระบบค่อนข้างสูง เนื่องจากทรัพยากรทั้งฮาร์ดแวร์และซอฟต์แวร์ที่นำมาใช้ต้องมีความสามารถสูง

2. เนื่องจากแนวคิดฐานข้อมูลเชิงสัมพันธ์ในภาพรวมนั้นง่ายต่อการนำไปใช้งาน ดังนั้นบุคลากรที่ไม่ได้รับการฝึกอบรมหรือเป็นผู้ที่มีความรู้ไม่ดีพอ ได้นำเครื่องมือไปใช้งานที่ผิด ทำให้ระบบที่ดีต้องแย่ลง และหากไม่ได้รับการตรวจสอบ อาจทำให้เกิดข้อมูลซ้ำซ้อนได้เช่นเดียวกับระบบแฟ้มข้อมูล

BRANCH

branchNo	street	city	postcode
B005	21 ถ.ห้วยแก้ว	เชียงใหม่	50300
B007	56 ถ.พหลโยธิน	พิษณุโลก	65150
B003	143 ถ.วิภาวดีรังสิต	กรุงเทพฯ	10110
B004	22 ถ.สหมิตร	เชียงราย	57000
B002	11 ถ.พหลโยธิน	เชียงใหม่	50300

Link through branchNo

EMPLOYEE

empNo	empName	position	sex	birthDate	salary	branchNo
SL21	ชูชัย สุขศรี	ผู้จัดการ	M	01/10/2508	30000	B005
SG37	ศิริทัตร มณีจันทร์	ผู้ช่วย	F	10/09/2515	12000	B003
SG14	สมศักดิ์ แซ่ตั้ง	ซูเปอร์ไวเซอร์	M	24/03/2520	18000	B003
SA09	ปิยะฉัตร เอี่ยมสุข	ผู้ช่วย	F	19/02/2521	9000	B007
SG05	พรรัตน์ ธนะศิลป์	ผู้จัดการ	F	03/06/2511	24000	B003
SL41	ลัลดา วงศ์ดี	ผู้ช่วย	F	13/06/2524	9000	B005

ภาพที่ 3.6 แบบจำลองฐานข้อมูลเชิงสัมพันธ์ (Relational Database Model)
ที่มา : โอภาส เอี่ยมสิริวงศ์ (2558, น. 85)

4. แบบจำลองฐานข้อมูลเชิงวัตถุ

แบบจำลองฐานข้อมูลเชิงวัตถุ (object-oriented database model) ถือเป็นเทคโนโลยีใหม่ของการจัดการฐานข้อมูลเชิงวัตถุ ซึ่งเกิดจากแนวคิดการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Program : OOP) ด้วยการมองทุกสิ่งเป็นวัตถุ โดยแต่ละวัตถุจะเป็นแหล่งรวมของข้อมูลและโอเปอเรชั่น (data and operation) มีคลาส (class) เป็นตัวกำหนดคุณสมบัติหรือรายละเอียดของวัตถุ รวมทั้งคุณสมบัติปกปิดความลับของวัตถุ (encapsulation) กล่าวคือการเข้าถึงข้อมูลจะต้องมีการตอบรับจากเมธอดในวัตถุนั้นว่าจะอนุญาตหรือไม่ที่จะให้วัตถุที่ส่งเมสเสจร้องขอเพื่อเข้าข้อมูลตน ข้อดีของแบบจำลองฐานข้อมูลเชิงวัตถุคือ สามารถจัดการกับข้อมูลชนิดต่างๆ ที่มีความสลับซับซ้อนได้เป็นอย่างดี ไม่ว่าจะเป็นข้อมูลประเภทกราฟิก วิดีโอและเสียง นอกจากนี้ยังสนับสนุนคุณสมบัติของการนำกลับมาใช้ใหม่ (reusable) แบบจำลอง

ฐานข้อมูลเชิงวัตถุจึงเป็นเทคโนโลยีใหม่ของ DBMS ที่มักนำไปใช้กับหน่วยงานขนาดใหญ่ สำหรับข้อดีและข้อเสียของแบบจำลองฐานข้อมูลชนิดนี้คือ

ข้อดี

1. คุณสมบัติด้านการสืบทอด (inheritance) ทำให้ข้อมูลมีความคงสภาพสูง
2. มีคุณสมบัติในการนำกลับมาใช้ใหม่
3. การนำเสนอเป็นรูปแบบ visual ทำให้อธิบายหัวข้อของความหมายได้ดี

ข้อเสีย

1. ต้องพึ่งพาผู้เชี่ยวชาญโดยเฉพาะ และมีค่าใช้จ่ายเกี่ยวกับระบบค่อนข้างสูง
2. ยังไม่มีมาตรฐานรองรับที่ชัดเจนเมื่อเทียบกับแบบจำลองฐานข้อมูลเชิงสัมพันธ์ อีกทั้งผลิตภัณฑ์ DBMS ที่ใช้งานบนแบบจำลองฐานข้อมูลเชิงสัมพันธ์ได้พัฒนาขีดความสามารถก้าวไปอีกขั้นหนึ่งด้วยการรวมเทคโนโลยีเชิงวัตถุเข้าไป ที่เรียกว่า Object-Relational Database
3. ผู้เชี่ยวชาญส่วนใหญ่คุ้นเคยกับเทคโนโลยีฐานข้อมูลเชิงสัมพันธ์มากกว่า

5. แบบจำลองฐานข้อมูลมัลติไดเมนชั่น

แบบจำลองฐานข้อมูลมัลติไดเมนชั่น (multidimensional database) ใช้งานกับคลังข้อมูล (data warehousing) โดยนำเสนอข้อมูลในลักษณะไดเมนชั่นทำให้วิวข้อมูลได้ 2 ทาง เพื่อให้สามารถมองเห็นปัญหาทางธุรกิจและสร้างวิธีการแก้ไขปัญหาได้ดียิ่งขึ้น แบบจำลองฐานข้อมูลมัลติไดเมนชั่นมีการนำกระบวนการทำงานทางธุรกิจมาจัดการให้อยู่ในรูปของมิติ เช่น การนำข้อมูลผลิตภัณฑ์สินค้ากับข้อมูลพื้นที่การขาย มาประมวลเป็นตารางในรูปแบบของมัลติไดเมนชั่น ทำให้ผู้ใช้สามารถตัดขวางหรือแบ่งข้อมูลเป็นส่วนๆ (data cube) มาวิเคราะห์ใช้งานได้ตามความต้องการ ตัวอย่างแบบจำลองฐานข้อมูลมัลติไดเมนชั่นที่ถูกสร้างขึ้นเพื่อใช้งานในฐานข้อมูลเชิงสัมพันธ์คือ star schema ซึ่งจะมี fact table หนึ่งตารางอยู่ที่ส่วนกลาง และบรรจุแอตทริบิวต์ทั้งหมดที่จะถูกวัด ส่วนตารางที่เหลือจะเป็นตาราง dimension ที่จะนำมารวมกันเพื่อทำการ join ระหว่างตาราง fact กับตาราง dimension สำหรับข้อดีและข้อเสียของแบบจำลองฐานข้อมูลชนิดนี้คือ

ข้อดี

1. สามารถนำมาใช้เพื่อวางแผนกลยุทธ์และสร้างวิธีแก้ไขปัญหาทางธุรกิจได้
2. ข้อมูลที่นำเสนอสามารถนำมุมมองได้หลายมิติ

ข้อเสีย

1. ใช้เงินลงทุน ทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ที่นำมาใช้เพื่อการวิเคราะห์

2. คลังข้อมูลต้องได้รับการออกแบบที่ดี มิฉะนั้นอาจไม่สามารถนำมาใช้ประโยชน์ได้อย่างคุ้มค่า

3. ผู้เชี่ยวชาญเฉพาะด้านในปัจจุบันยังคงมีไม่มาก

4. เหมาะกับธุรกิจขนาดใหญ่

โครงสร้างฐานข้อมูลเชิงสัมพันธ์

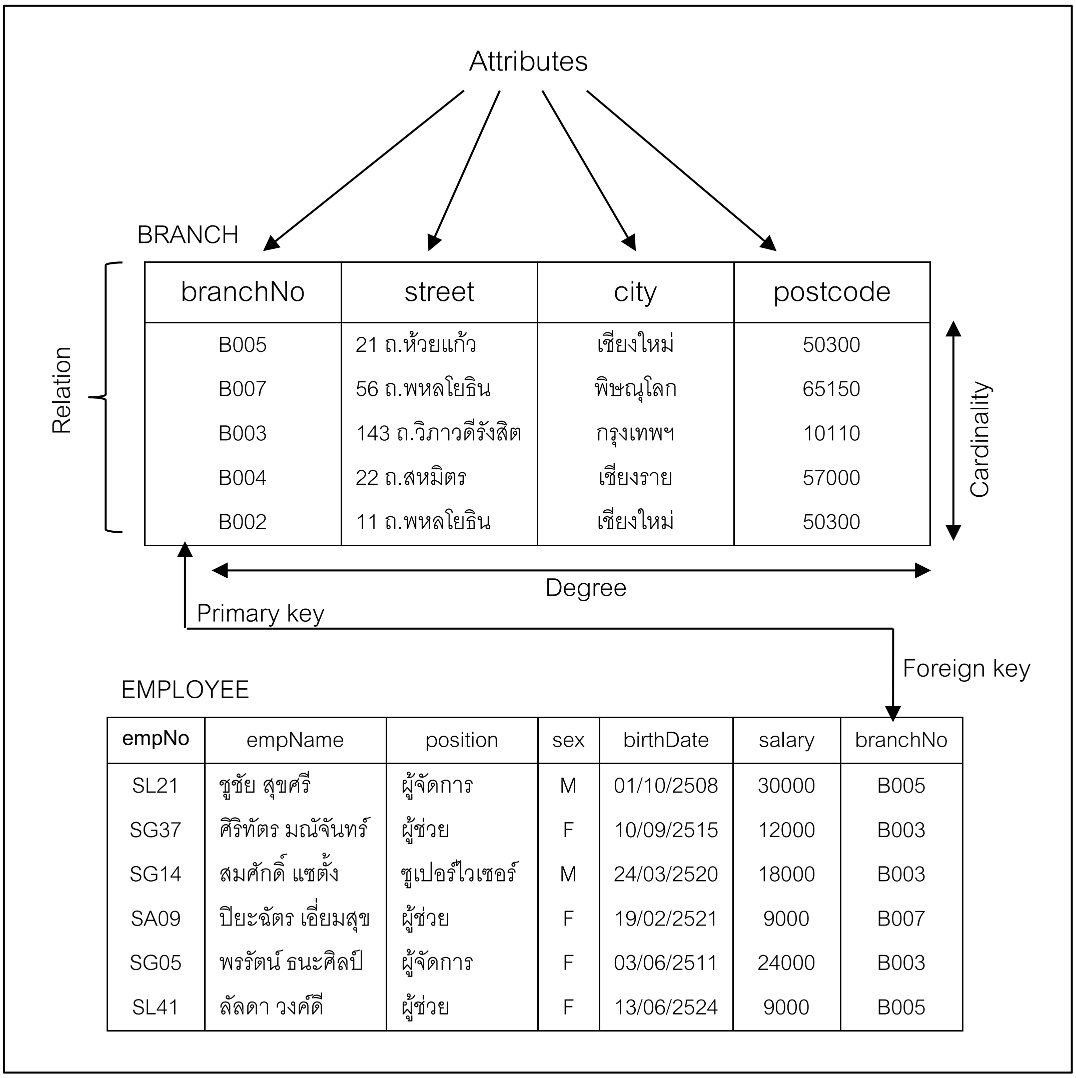
โครงสร้างฐานข้อมูลเชิงสัมพันธ์ประกอบด้วย รีเลชั่น แอตทริบิวต์ โดเมน ทูเพิล ดีกรี และคาร์ดินัลลิตี้ ซึ่งมีรายละเอียดดังต่อไปนี้ (Sikha & Richard, 2012, p. 124)

1. รีเลชั่น

รีเลชั่น (relation) ในระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDBMS) จะนำไปใช้บนฐานข้อมูลด้วยการให้ผู้ใช้เห็นภาพของข้อมูลในรูปแบบของตาราง โดยข้อมูลและโครงสร้างข้อมูลมีความอิสระจากตัวโปรแกรม การอ้างอิงโครงสร้างข้อมูลเพื่อใช้งานจะเป็นระดับแนวคิดมากกว่าโครงสร้างในระดับกายภาพ ทั้งนี้โครงสร้างทางกายภาพของฐานข้อมูลที่แท้จริงอาจถูกสร้างด้วยโครงสร้างแฟ้มข้อมูลแบบ B-Tree หรือ indexed sequential ก็ได้ อาจสรุปได้ว่ารีเลชั่นหมายถึงตารางสองมิติ ซึ่งประกอบด้วยคอลัมน์ (columns) และแถว (rows)

2. แอตทริบิวต์

ในแบบจำลองเชิงสัมพันธ์ รีเลชั่นใช้สำหรับบรรจุสารสนเทศที่เกี่ยวข้องกับวัตถุต่างๆ เพื่อนำไปใช้เป็นตัวแทนในฐานข้อมูล ซึ่งตัวแทนของรีเลชั่นก็คือตาราง 2 มิติที่ประกอบไปด้วยแถว ซึ่งก็คือเรคอร์ดที่ประกอบไปด้วยข้อมูลเฉพาะแถวของรายการนั้นๆ ในทำนองเดียวกันคอลัมน์ในตารางก็คือแอตทริบิวต์ (attribute) โดยสามารถพิจารณาจากภาพที่ 3.7 จะเห็นได้ว่ารีเลชั่น BRANCH จะประกอบไปด้วยแอตทริบิวต์ชื่อ branchNo, street, city และ postcode ในขณะที่รีเลชั่น EMPLOYEE ก็จะประกอบไปด้วยแอตทริบิวต์ชื่อ empNo, empName, position, sex, birthdate, salary และ branchNo ซึ่งตารางทั้งสองมีการเชื่อมโยงความสัมพันธ์ด้วยแอตทริบิวต์branchNo สรุปได้ว่า แอตทริบิวต์ (attribute) คือชื่อคอลัมน์ต่างๆ ในรีเลชั่นที่ใช้อธิบายคุณลักษณะของแต่ละเอนทิตี



branchNo	street	city	postcode
B005	21 ถ.ห้วยแก้ว	เชียงใหม่	50300
B007	56 ถ.พหลโยธิน	พิษณุโลก	65150
B003	143 ถ.วิภาวดีรังสิต	กรุงเทพฯ	10110
B004	22 ถ.สหมิตร	เชียงราย	57000
B002	11 ถ.พหลโยธิน	เชียงใหม่	50300

empNo	empName	position	sex	birthDate	salary	branchNo
SL21	ชูชัย สุขศรี	ผู้จัดการ	M	01/10/2508	30000	B005
SG37	ศิริทัตร มณีจันทร์	ผู้ช่วย	F	10/09/2515	12000	B003
SG14	สมศักดิ์ แซ่ตั้ง	ซูเปอร์ไวเซอร์	M	24/03/2520	18000	B003
SA09	ปิยะฉัตร เอี่ยมสุข	ผู้ช่วย	F	19/02/2521	9000	B007
SG05	พรรัตน์ ธนะศิลป์	ผู้จัดการ	F	03/06/2511	24000	B003
SL41	ลัลดา วงศ์ดี	ผู้ช่วย	F	13/06/2524	9000	B005

ภาพที่ 3.7 ตัวอย่างรีเลชั่น BRANCH และ EMPLOYEE

ที่มา : โอภาส เอี่ยมสิริวงศ์ (2558, น. 90)

3. โดเมน

แนวคิดของโดเมน (domain) เป็นการกำหนดขอบเขตค่าข้อมูลที่เป็นไปได้ และรูปแบบชนิดข้อมูลของแต่ละแอตทริบิวต์ ซึ่งโดเมนอาจกำหนดให้มีความแตกต่างกันในแต่ละแอตทริบิวต์ หรือหลายๆ แอตทริบิวต์อาจจะมีการกำหนดให้อยู่บนโดเมนเดียวกันก็ได้ โดยพิจารณาจากภาพที่ 3.8 ที่แสดงถึงโดเมนในบางแอตทริบิวต์ของรีเลชั่น BRANCH และ EMPLOYEE โดยโดเมนของแอตทริบิวต์ชื่อ branchNo จะสามารถกำหนดค่าได้ในช่วงระหว่าง B001 - B999 ซึ่งหมายความว่ารหัสสาขานี้สามารถกำหนดค่าได้ในช่วงดังกล่าวเท่านั้น โดเมนของแอตทริบิวต์ชื่อ sex ค่าที่เป็นไปได้ต้องเป็นค่า M หรือ F เท่านั้น สำหรับแนวคิดโดเมนยังอนุญาตให้ผู้ใช้สามารถ

กำหนดขึ้นเองได้ ซึ่งถือเป็นการหลีกเลี่ยงการป้อนข้อมูลผิดพลาด ตัวอย่างการกำหนดโดเมนให้กับรีเลชั่นแสดงในตารางที่ 3.1

ตารางที่ 3.1 ตัวอย่างการกำหนดโดเมนของรีเลชั่น BRANCH และ EMPLOYEE

ที่มา : Sikha & Richard (2012, p. 126)

Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	Character: size 4, rang B001-B999
street	StreetNames	The set of all street names in Thailand	Character: size 25
city	CityNames	The set of all city names in Thailand	Character: size 15
postcode	Postcodes	The set of all postcodes in Thailand	Character: size 8
sex	Sex	The sex of a person	Character: size 1, value M or F
birthDate	BirthDate	Possible value of employee birth dates	date, format dd-mm-yyyy
salary	Salaries	Possible values of employee salaries	Monetary: 7 digits, range 6000.00-40000.00

4. ทูเพิล

ทูเพิล (tuple) คือแถวของรีเลชั่น หรืออาจเรียกว่าเรคอร์ด พิจารณาจากรีเลชั่น BRANCH จากภาพที่ 3.7 จะเห็นได้ว่าแถวแต่ละแถวจะประกอบไปด้วยค่า 4 ค่า โดยทูเพิลของสาขา B003 ประกอบด้วยข้อมูลดังต่อไปนี้

branchNo	street	city	postcode
B003	143 ถ.วิภาวดีรังสิต	กรุงเทพฯ	10110

5. ดีกรี

ดีกรี (degree) ของรีเลชั่นคือจำนวนแอตทริบิวต์ที่บรรจุอยู่ ตัวอย่างเช่น ในภาพที่ 3.7 รีเลชั่น BRANCH มี 4 แอตทริบิวต์ หรือเรียกว่ามี 4 ดีกรี ซึ่งหมายความว่าในแต่ละแถวในตาราง BRANCH จะประกอบไปด้วย 4 ดีกรีที่บรรจุข้อมูลจำนวน 4 ค่าด้วยกัน

6. คาร์ดินัลลิตี้

คาร์ดินัลลิตี้ (cardinality) ของรีเลชั่นคือ จำนวนของทูเพิลที่บรรจุอยู่ โดยจำนวนของคาร์ดินัลลิตี้สามารถเปลี่ยนแปลงได้จากการเพิ่มทูเพิลหรือลบทูเพิลในรีเลชั่น

ฐานข้อมูลเชิงสัมพันธ์

พนิดา พานิชกุล และ ณัฐพงษ์ วารีประเสริฐ (2552, น. 36) ให้ความหมายของฐานข้อมูลเชิงสัมพันธ์ (relational database) ว่าเป็นฐานข้อมูลที่อาศัยแนวคิดเรื่องความสัมพันธ์ หรือรีเลชั่น (relation) ทางคณิตศาสตร์เข้ามาอธิบายลักษณะของข้อมูล แล้วนำเสนอออกมาในรูปแบบของรีเลชั่น คืออยู่ในรูปแบบของตาราง (table) 2 มิติ ที่แสดงคุณลักษณะของข้อมูลในแนวคอลัมน์ และแสดงค่าของข้อมูลนั้นในแนวนอน

วิเชียร เปรมชัยสวัสดิ์ (2550, น. 37) กล่าวว่า ฐานข้อมูลเชิงสัมพันธ์ใช้โครงสร้างข้อมูลเชิงตรรกะ ซึ่งข้อมูลทั้งหมดที่อยู่ในฐานข้อมูลเชิงสัมพันธ์จะมีรูปแบบเดียวกันทั้งหมดคือ รีเลชั่นที่สนับสนุนการเข้าถึงข้อมูลโดยผ่านทางภาษาคิวรี (query)

ปราลี มณีรัตน์ (2552, น. 39) กล่าวว่า ฐานข้อมูลเชิงสัมพันธ์เป็นฐานข้อมูลที่ประกอบด้วยตารางหลายๆ ตาราง ฐานข้อมูลเชิงสัมพันธ์ใช้แอตทริบิวต์ (attribute) เชื่อมความสัมพันธ์ระหว่างตารางเข้าไว้ด้วยกัน ซึ่งการทำงานของฐานข้อมูลเชิงสัมพันธ์จะช่วยควบคุมการเก็บข้อมูลที่ซ้ำซ้อน

Date (2004, p. 64) กล่าวว่าฐานข้อมูลเชิงสัมพันธ์ประกอบด้วยตารางต่างๆ ที่มีความสัมพันธ์กันโดยนำทฤษฎีทางคณิตศาสตร์ ซึ่งได้แก่ทฤษฎีเซตและตรรกศาสตร์ มาใช้ในการดำเนินการเพื่อหาความสัมพันธ์ระหว่างตาราง

อาจสรุปได้ว่า ฐานข้อมูลเชิงสัมพันธ์ (relational database) คือ การรวมกลุ่มของรีเลชั่นซึ่งได้ผ่านการนอร์มัลไลเซชั่น โดยแต่ละรีเลชั่นจะมีการกำหนดชื่อที่แตกต่างกันฐานข้อมูลเชิงสัมพันธ์ประกอบด้วยรีเลชั่นต่างๆ ที่มีการกำหนดความสัมพันธ์เข้าด้วยกันไว้อย่างมีแบบแผน ฐานข้อมูลเชิงสัมพันธ์สามารถนำเสนอความสัมพันธ์ระหว่างข้อมูลได้อย่างครอบคลุม และมีกฎเกณฑ์สำหรับการควบคุมที่ไม่ทำให้เกิดปัญหาการซ้ำซ้อนกันของข้อมูล ฐานข้อมูลเชิงสัมพันธ์จึงได้รับความนิยมเรื่อยมาจากอดีตจนถึงปัจจุบัน

ศัพท์ทางเทคนิคที่ใช้ในฐานข้อมูล

สำหรับศัพท์เทคนิคในฐานข้อมูลจะมีชื่อเรียกที่แตกต่างจากที่เราเคยคุ้นเคยกัน แต่มีความหมายเดียวกัน เช่น ในแต่ละแถวของรีเลชั่นจะเรียกว่า ทูเพิล (tuple) หรือแถว (row) ขณะที่คอลัมน์จะเรียกว่า แอตทริบิวต์ (attribute) หรือฟิลด์ (field) ดังแสดงในตารางที่ 3.2

ตารางที่ 3.2 ศัพท์ทางเทคนิคที่ใช้ในฐานข้อมูล
ที่มา : Eric & Jim (2012, p. 124)

ศัพท์เทคนิค	แบบที่ 1	แบบที่ 2
รีเลชั่น (relation)	ตาราง (table)	ไฟล์ (file)
ทูเพิล (tuple)	แถว (row)	ระเบียน (record)
แอตทริบิวต์ (attribute)	คอลัมน์ (column)	ฟิลด์ (field)

กฎของคอดด์

ในราวปี ค.ศ. 1985 Dr. E.F. Codd ได้เผยแพร่ผลงานเกี่ยวกับกฎ 12 ข้อบนฐานข้อมูลเชิงสัมพันธ์ของคอดด์ (Code's 12 Relationship Database Rules) ซึ่งมีเนื้อหารายการเกี่ยวกับกฎ 12 ข้อที่ได้บัญญัติไว้เพื่อนำไปนิยามบนระบบฐานข้อมูลเชิงสัมพันธ์ โดยมีวัตถุประสงค์เพื่อให้ผู้ผลิตต่างๆ ที่สร้างผลิตภัณฑ์ DBMS สามารถนำกฎดังกล่าวไปใช้เพื่อการอ้างอิงว่า DBMS ที่ได้สร้างขึ้นมานั้นมีความเป็นฐานข้อมูลเชิงสัมพันธ์มากน้อยเพียงไร ซึ่งแต่ละกฎประกอบด้วยรายละเอียดต่อไปนี้ (Eric & Jim, 2012, p. 124)

1. **กฎด้านสารสนเทศ** (Information) เป็นกฎที่ว่าด้วย สารสนเทศทั้งหมดในฐานข้อมูลเชิงสัมพันธ์ในเชิงตรรกะแล้วจะต้องนำเสนอในรูปแบบของตารางที่ประกอบไปด้วยคอลัมน์และแถว

2. **กฎการรับประกันและการเข้าถึง** (Guaranteed Access) เป็นกฎที่ว่าด้วย ทุกค่าในตาราง สามารถอ้างอิงเพื่อการเข้าถึงได้ด้วยการระบุชื่อตาราง คีย์หลัก และชื่อคอลัมน์

3. **กฎการปฏิบัติกับค่าว่าง** (Null) **อย่างมีแบบแผน** (Systematic Treatment of Null) เป็นกฎที่ว่าด้วย ค่าว่างที่นำไปใช้กับข้อมูลบางค่าที่ละไว้เป็นค่าว่าง จะต้องถูกปฏิบัติอย่างมีแบบแผนและมีความอิสระจากชนิดข้อมูล (data type)

4. **กฎการออนไลน์แคตาล็อกแบบไดนามิกที่ต้องอยู่บนพื้นฐานของแบบจำลองเชิงสัมพันธ์** (Dynamic On-Line Catalog Based on the Relational Model) เป็นกฎที่ว่าด้วยตารางต่างๆ ที่บรรจุภายในฐานข้อมูล ซึ่งภายในจะมี metadata ที่สามารถนำมาใช้เพื่องานด้านสิทธิการเข้าถึงฐานข้อมูลที่เป็นไปตามมาตรฐานภาษาฐานข้อมูลเชิงสัมพันธ์

5. **กฎด้านภาษาข้อมูล** (Comprehensive Data Sublanguage) เป็นกฎที่ว่าด้วยฐานข้อมูลเชิงสัมพันธ์อาจสนับสนุนภาษาต่างๆ ได้หลากหลาย แต่อย่างไรก็ตาม จะต้องสนับสนุนภาษาที่เกี่ยวข้องกับการนิยามข้อมูล การวิว การจัดการข้อมูล (ทำได้ทั้งแบบอินเตอร์

แอกทีฟ หรือผ่านโปรแกรม) การกำหนดข้อบังคับด้านความคงสภาพ การกำหนดสิทธิ์ของผู้ใช้และการจัดการทรานแซกชั่น

6. **กฎการปรับปรุงวิว** (View Updating) เป็นกฎที่ว่าด้วย วิวทุกวิวที่สามารถอัปเดตหรือปรับปรุงได้ ดังนั้นในเชิงทฤษฎีวิวจึงสามารถอัปเดตได้โดยผ่านระบบ

7. **กฎการแทรก การปรับปรุง และการลบข้อมูลระดับสูง** (High-Level Insert, Update and Delete) เป็นกฎที่ว่าด้วย ฐานข้อมูลจะต้องสนับสนุนกลุ่มคำสั่งระดับสูงที่สามารถนำมาใช้เพื่อการแทรก อัปเดต และการลบข้อมูล

8. **กฎความอิสระของข้อมูลเชิงกายภาพ** (Physical Data Independence) เป็นกฎที่ว่าด้วย แอปพลิเคชันโปรแกรมและฐานข้อมูลในระดับตรรกะจะไม่ได้รับผลกระทบใดๆ จากการเปลี่ยนแปลงวิธีการเข้าถึง และการปรับเปลี่ยนโครงสร้างของสื่อบันทึกข้อมูล

9. **กฎความอิสระของข้อมูงเชิงตรรกะ** (Logical Data Independence) เป็นกฎที่ว่าด้วย แอปพลิเคชันโปรแกรมจะไม่ได้รับผลกระทบใดๆ จากการเปลี่ยนแปลงโครงสร้างตารางในระดับตรรกะ

10. **กฎความคงสภาพที่เป็นอิสระ** (Integrity Independence) เป็นกฎที่ว่าด้วยข้อบังคับด้านกฎการคงสภาพ (integrity constraints) ในทุกรีเลชั่น จะต้องถูกกำหนดขึ้นด้วยภาษาเชิงสัมพันธ์ และจัดเก็บเชิงแค็ตตาล็อกของระบบ ซึ่งไม่ใช่ถูกกำหนดขึ้นจากระดับประยุกต์

11. **กฎการกระจายที่เป็นอิสระ** (Distribution Independence) เป็นกฎที่ว่าด้วย ผู้ใช้และโปรแกรมประยุกต์จะไม่รู้ตัวเลย รวมถึงไม่รับผลกระทบใดๆ ทั้งสิ้นจากตำแหน่งที่ตั้งของข้อมูลไม่ว่าจะเป็นฐานข้อมูลแบบโลคัลหรือฐานข้อมูลแบบกระจาย ที่ได้จัดเก็บฐานข้อมูลไปยังพื้นที่ต่างๆ ผ่านระบบเครือข่าย

12. **กฎปราศจากเวอร์ชันย่อย** (Nonsubversion) เป็นกฎที่ว่าด้วย หากระบบสนับสนุนการเข้าถึงข้อมูลระดับต่ำ จะต้องไม่สามารถใช้เป็นทางผ่านกฎความคงสภาพในฐานข้อมูลได้

รีเลชั่น

สคีมาฐานข้อมูลเชิงสัมพันธ์ (Relational Database Schema) คือ การนำรีเลชั่นสคีมามารวมกลุ่มกันอย่างมีความสัมพันธ์ โดยชื่อของแต่ละรีเลชั่นสคีมาต้องแตกต่างกัน

สคีมาเชิงสัมพันธ์ที่เป็นส่วนหนึ่งในฐานข้อมูล DreamHome ที่นำมาใช้เป็นกรณีศึกษา เพื่อประกอบเป็นตัวอย่างฐานข้อมูล ประกอบด้วยสคีมาต่างๆ ตามภาพที่ 3.8

DreamHome เป็นบริษัทบริการบ้านพักอาศัยแก่ลูกค้าที่ต้องการหาบ้านเช่า ซึ่ง DreamHome เป็นบริษัทตัวแทนที่บริการบ้านเช่าเพื่อการพักอาศัยให้แก่ลูกค้า โดยลูกค้าสามารถเลือกบ้านเช่าที่ต้องการในแต่ละจังหวัดได้

DreamHome ได้เปิดสำนักงานตามสาขาต่างๆ กระจายอยู่ทั่วภูมิภาคของประเทศไทย โดยมีบ้านเช่าที่เจ้าของบ้านตามจังหวัดต่างๆ มากมายทั่วประเทศเข้าร่วมโครงการกับทางบริษัท ดังนั้นลูกค้าที่ต้องการบ้านเช่าก็เพียงแต่เข้ามาติดต่อกับบริษัทสำนักงานที่เปิดในสาขาใกล้บ้าน โดยแจ้งความจำนงต่อเจ้าหน้านี่ว่าต้องการบ้านเช่าเพื่อพักอาศัยในจังหวัดใด เป็นบ้านพักแบบไหน งบประมาณค่าเช่าต่อวันเท่าไร เป็นต้น จากนั้นเจ้าหน้าที่ก็จะค้นหาและแจ้งข้อมูลกับที่พักอาศัยให้ เพื่อให้ลูกค้านำไปใช้เป็นข้อมูลประกอบการพิจารณาในการพักอาศัยต่อไป

และต่อไปนี้คือรีเลชั่นสคีมาเพียงบางส่วนของฐานข้อมูลเชิงสัมพันธ์ของบริษัท DreamHome โดยแอตทริบิวต์ที่ขีดเส้นใต้หมายถึงแอตทริบิวต์ที่กำหนดให้เป็นคีย์หลัก

BRANCH	(<u>branchNo</u>, street, city, postcode)
EMPLOYEE	(<u>empNo</u>, empName, position, sex, birthdate, salary, branchNo)
PROPERTY_FOR_RENT	(<u>propertyNo</u>, street, city, postcode, type, rooms, rent, ownerNo, empNo, branchNo)
CLIENT	(<u>clientNo</u>, clientName, telNo, prefType, maxRent)
PRIVATE_OWNER	(<u>ownerNo</u>, ownerName, address, telNo)
VIEWING	(<u>clientNo</u>, propertyNo, viewDate, comment)

โดยที่

BRANCH คือสาขาที่เปิดให้บริการของบริษัท DreamHome

EMPLOYEE คือพนักงานของบริษัท DreamHome ที่ประจำอยู่ในแต่ละสาขา

PROPERTY_FOR_RENT คือรายละเอียดบ้านที่ปล่อยให้เช่า

CLIENT คือข้อมูลลูกค้าที่เช่าบ้าน

PRIVATE_OWNER คือเจ้าบ้านที่นำบ้านตนเองมาเข้าร่วมโครงการกับ DreamHome

VIEWING คือข้อมูลที่ลูกค้าเข้ามาติดต่อเพื่อเยี่ยมชมบ้านเช่า

BRANCH

branchNo	street	city	postcode
B005	21 ถ.ห้วยแก้ว	เชียงใหม่	50300
B007	56 ถ.พหลโยธิน	พิษณุโลก	65150
B003	143 ถ.วิภาวดีรังสิต	กรุงเทพฯ	10110
B004	22 ถ.สหมิตร	เชียงราย	57000
B002	11 ถ.พหลโยธิน	เชียงใหม่	50300

EMPLOYEE

empNo	empName	position	sex	birthDate	salary	branchNo
SL21	ชูชัย สุขศรี	ผู้จัดการ	M	01/10/2508	30000	B005
SG37	ศิริทัตร มณีจันทร์	ผู้ช่วย	F	10/09/2515	12000	B003
SG14	สมศักดิ์ แซ่ตั้ง	ซูเปอร์ไวเซอร์	M	24/03/2520	18000	B003
SG09	ปิยะฉัตร เอี่ยมสุข	ผู้ช่วย	F	19/02/2521	9000	B007
SG05	พรรัตน์ ธนะศิลป์	ผู้จัดการ	F	03/06/2511	24000	B003
SL41	ลัดดา วงค์ดี	ผู้ช่วย	F	13/06/2524	9000	B005

PROPERTY_FOR_RENT

Property No	street	city	Post code	type	rooms	rent	Owner No	Emp No	Branch No
PA14	19 ถ.โยธิน	พิษณุโลก	65000	บ้าน	6	650	CO46	SA09	B007
PL94	14 ถ.ลำห้วย	เชียงใหม่	50310	แฟลต	4	400	CO87	SL41	B005
PG04	6 ถ.วิภาวดีรังสิต	กรุงเทพฯ	10200	แฟลต	3	350	CO40		B003
PG36	2 ถ.ประชาอุทิศ	กรุงเทพฯ	18160	แฟลต	3	375	CO93	SG37	B003
PG21	18 ถ.พญาไท	กรุงเทพฯ	10400	บ้าน	5	600	CO87	SG37	B003
PG16	5 ถ.พญาไท	กรุงเทพฯ	10400	แฟลต	4	450	CO93	SG14	B003

ภาพที่ 3.8 รีเลชั่นต่างๆ ในฐานข้อมูล DreamHome

CLIENT

clientNo	ClientName	telNo	prefType	maxRent
CR76	ยงยุทธ ธนเลิศ	02-5568891	แฟลต	425
CR56	สิราณี พรมจรรย์	02-4456328	แฟลต	350
CR74	ศรีสมร หิรัญพงศ์	02-4505568	บ้าน	750
CR62	ตะวัน ส่งศรีสุข	081-7655588	แฟลต	600

PRIVATE_OWNER

ownerNo	ownerName	address	telNo
CO46	นิรมล พัฒนา	พิษณุโลก	081-4445568
CO87	ฉัตรชัย ขุนศรี	กรุงเทพฯ	02-5369980
CO40	กานดา ใจมา	กรุงเทพฯ	02-4868891
CO93	สุขใจ แซ่ลี้	กรุงเทพฯ	02-5678811

VIEWING

clientNo	propertyNo	viewDate	comment
CR56	PA14	24/06/2550	พื้นที่ขนาดเล็ก
CR76	PG04	20/04/2550	ระยะทางไกลเกิน
CR56	PG04	26/05/2550	
CR62	PA14	14/05/2550	ไม่มีห้องครัว
CR56	PG36	28/04/2550	

ภาพที่ 3.8 **(ต่อ)** รีเลชั่นต่างๆ ในฐานข้อมูล DreamHome

คุณสมบัติของรีเลชั่น

รีเลชั่นประกอบไปด้วยคุณสมบัติต่างๆ ดังต่อไปนี้

1. รีเลชั่นจะต้องมีชื่อกำกับ โดยแต่ละรีเลชั่นจะมีชื่อที่แตกต่างกันซ้ำกันไม่ได้

2. แต่ละแอตทริบิวต์ของรีเลชั่นจะบรรจุค่าเพียงค่าเดียวกล่าวคือ ค่าที่บรรจุนั้นจะไม่สามารถแบ่งย่อยออกไปได้อีก

3. แต่ละแอตทริบิวต์ในรีเลชั่นนั้นต้องมีชื่อแตกต่างกันจะมีแอตทริบิวต์ซ้ำกันไม่ได้

4. ข้อมูลที่บรรจุลงในแอตทริบิวต์เป็นไปตามข้อกำหนดของโดเมนที่กำหนดขึ้น

5. ข้อมูลในแต่ละทูเพิลต้องมีความแตกต่างกัน จะไม่มีทูเพิลที่ซ้ำกัน

6. การเรียงลำดับของแต่ละแอตทริบิวต์ไม่มีความสำคัญใดๆ

7. การเรียงลำดับของทูเพิลไม่มีความสำคัญใดๆ (มุมมองเชิงทฤษฎี) แต่อย่างไรก็ตามสำหรับมุมมองเชิงปฏิบัติแล้วการเรียงลำดับของทูเพิลส่งผลต่อประสิทธิภาพในการเข้าถึงข้อมูล

คีย์ของข้อมูล

ฐานข้อมูลเชิงสัมพันธ์จะจัดเก็บข้อมูลในรูปแบบของตาราง 2 มิติที่ประกอบด้วยแถวและคอลัมน์ และด้วยคุณสมบัติของรีเลชั่นที่ได้กล่าวไว้แล้วในข้างต้นก็คือ ข้อมูลในแต่ละทูเพิลต้องมีความแตกต่างกันไม่สามารถมีทูเพิลที่ซ้ำกันได้ ดังนั้นจึงจำเป็นต้องมีการกำหนดแอตทริบิวต์หรือกลุ่มของแอตทริบิวต์ที่เรียกว่าคีย์ (key) เพื่อให้แต่ละทูเพิลมีความแตกต่างกัน เพื่อสามารถระบุความเป็นเอกลักษณ์ในแต่ละทูเพิลกล่าวคือ การกำหนดคีย์ให้กับรีเลชั่น ทำให้เราสามารถนำคีย์เหล่านี้ไประบุทูเพิลในรีเลชั่นตามที่เราต้องการได้นั่นเอง โดยคีย์ที่ใช้งานในฐานข้อมูลประกอบด้วยคีย์ชนิดต่างๆ ดังต่อไปนี้ (Chittayasothorn, 2007, p. 135)

1. ซูเปอร์คีย์

ซูเปอร์คีย์ (superkey) คือแอตทริบิวต์หรือกลุ่มของแอตทริบิวต์ที่ใช้ระบุถึงความเป็นเอกลักษณ์ในแต่ละทูเพิลของรีเลชั่นนั้นได้ ดังนั้นในรีเลชั่นหนึ่งๆ จึงสามารถมีซูเปอร์คีย์ได้หลายตัว ตัวอย่างเช่น รีเลชั่น EMPLOYEE ที่ประกอบไปด้วยแอตทริบิวต์ต่างๆ ดังต่อไปนี้

EMPLOYEE (empNo, empName, position)

จากนิยามของซูเปอร์คีย์ที่กล่าวมาแล้วในข้างต้น จึงทำให้เราสามารถกำหนดซูเปอร์คีย์ให้กับรีเลชั่น EMPLOYEE เพื่อใช้ระบุความแตกต่างหรือความเป็นเอกลักษณ์ของแต่ละทูเพิลได้ดังนี้

(empNo)
(empNo, empName)
(empName)

2. คีย์คู่แข่ง

ในรีเลชั่นหนึ่งๆ อาจมีคีย์คู่แข่งได้หลายตัว โดยคีย์คู่แข่ง (candidate key : CK) คือซูเปอร์คีย์ที่มีขนาดเล็กที่สุดที่สามารถระบุความแตกต่างในแต่ละทูเพิลบนรีเลชั่นได้ กล่าวคือคีย์คู่แข่งเป็นซูเปอร์คีย์ที่ไม่มีซับเซต (subset) ของตัวเองที่นำมาใช้เป็นซูเปอร์คีย์นั่นเอง ดังนั้นกลุ่มของแอตทริบิวต์ (empNo, empName) คีย์เหล่านี้จึงไม่ถือว่าเป็นคีย์คู่แข่งเนื่องจากมีซับเซตเป็นคีย์คู่แข่ง ดังนั้นรีเลชั่น EMPLOYEE จึงมีแอตทริบิวต์ที่เหลือคือ empNo, และ empName ที่สามารถนำมาใช้เป็นคีย์คู่แข่งได้

3. คีย์หลัก

คีย์หลัก (primary key : PK) คือคีย์คู่แข่งที่ถูกคัดเลือกให้เป็นคีย์หลัก ที่สามารถนำไปใช้เพื่อการอ้างอิงความเป็นเอกลักษณ์ของแต่ละทูเพิลในรีเลชั่นนั้นๆ ได้ คีย์หลักต้องไม่บรรจุข้อมูลว่างเปล่า (null) โดยในที่นี้ได้คัดเลือกแอตทริบิวต์ empNo เป็นคีย์หลัก

4. คีย์รอง

คีย์คู่แข่งที่ไม่ได้ผ่านการคัดเลือก จะเรียกว่า คีย์รอง (alternate key / secondary key) ซึ่งเมื่อนำไปใช้ในการค้นหาข้อมูลจากความสัมพันธ์ อาจได้ทูเพิลมากกว่าหนึ่งเรคอร์ด เพราะคีย์รองนั้นไม่มีความเป็นเอกลักษณ์ ตัวอย่างเช่น ให้แอตทริบิวต์ empName เป็นคีย์รอง ดังนั้น ในตารางพนักงานอาจมีพนักงานที่มีชื่อเหมือนกันก็ได้

5. คีย์นอก

คีย์นอก (foreign key : FK) ประกอบด้วย แอตทริบิวต์ หรือกลุ่มของแอตทริบิวต์ในรีเลชั่นหนึ่งที่มีคุณสมบัติเป็นคีย์หลัก และไปปรากฏในอีกรีเลชั่นหนึ่งเพื่อใช้สำหรับการเชื่อมโยงระหว่างกัน ดังนั้นคีย์นอกจึงจัดเป็นคีย์ที่มีความสำคัญมากในฐานข้อมูลเชิงสัมพันธ์ เนื่องจาก

เป็นคีย์ที่ใช้สำหรับเชื่องโยงความสัมพันธ์ระหว่างตาราง โดยพิจารณาจากรูปที่ 3.7 ที่ผ่านมา รีเลชั่น BRANCH จะมีคีย์หลักคือ แอตทริบิวต์ branchNo ในขณะที่รีเลชั่น EMOLOYEE จะมีแอตทริบิวต์ branchNo เช่นกัน แต่ถูกกำหนดเป็นคีย์นอกเพื่อใช้เชื่อมโยงไปยังรีเลชั่น BRANCH นั่นเอง

6. คีย์รวม

คีย์รวม (composite key) ประกอบด้วยกลุ่มของแอตทริบิวต์ที่ถูกนำมารวมกัน ซึ่งคีย์ดังกล่าวยังสามารถเรียกได้อีกหลายชื่อด้วยกัน เช่น concatenated key หรือ compound Key

สรุป

ในบทนี้ได้กล่าวถึงแบบจำลองข้อมูล ซึ่งแบ่งออกเป็น 2 ประเภทคือ แบบจำลองเชิงแนวคิด และแบบจำลองเพื่อการนำไปใช้ พื้นฐานการสร้างแบบจำลองข้อมูลประกอบด้วยเอนทิตี แอตทริบิวต์ ความสัมพันธ์ และข้อบังคับ โดยความสัมพันธ์ระหว่างเอนทิตีประกอบด้วยความสัมพันธ์ในรูปแบบต่างๆ ได้แก่ ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (1:1) ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1:M) และความสัมพันธ์แบบกลุ่มต่อกลุ่ม (M:N) แบบจำลองฐานข้อมูลประกอบด้วย แบบจำลองฐานข้อมูลลำดับชั้น แบบจำลองฐานข้อมูลเครือข่าย แบบจำลองฐานข้อมูลเชิงสัมพันธ์ แบบจำลองฐานข้อมูลเชิงวัตถุ และแบบจำลองฐานข้อมูลแบบมัลติไดเมนชั่น ทั้งนี้ โครงสร้างข้อมูลเชิงสัมพันธ์ประกอบด้วย รีเลชั่น แอตทริบิวต์ โดเมน ทูเพิล ดีกรี คาร์ดินัลลิตี้ และฐานข้อมูลเชิงสัมพันธ์ นอกจากนี้ได้กล่าวถึงกฎ 12 ข้อบนฐานข้อมูลเชิงสัมพันธ์ของคอดด์ รีเลชั่น คุณสมบัติของรีเลชั่น และคีย์ของข้อมูลชนิดต่างๆ ได้แก่ ซูเปอร์คีย์ คีย์คู่แข่ง คีย์หลัก คีย์รอง คีย์นอก และคีย์รวม

แบบฝึกหัดทบทวน

1. จงบอกจุดประสงค์ของการใช้แบบจำลองเชิงแนวคิดมาให้เข้าใจพอสังเขป
2. จงอธิบายความหมายของคำต่อไปนี้
    2.1 เอนทิตี (Entities)
    2.2 แอตทริบิวต์ (Attributes)
    2.3 ความสัมพันธ์ (Relationships)
    2.4 ข้อบังคับ (Constraints)
3. จงยกตัวอย่างความสัมพันธ์แบบหนึ่งต่อหนึ่ง พร้อมคำอธิบาย
4. จงยกตัวอย่างความสัมพันธ์แบบหนึ่งต่อกลุ่ม พร้อมคำอธิบาย
5. จงยกตัวอย่างความสัมพันธ์แบบกลุ่มต่อกลุ่ม พร้อมคำอธิบาย
6. แบบจำลองฐานข้อมูลมีกี่ประเภท อะไรบ้าง จงสรุปมาให้เข้าใจพอสังเขปพร้อมข้อดีและข้อเสีย
7. จงอธิบายส่วนต่างๆ ในโครงสร้างข้อมูลเชิงสัมพันธ์ต่อไปนี้
    7.1 รีเลชั่น (Relation)
    7.2 แอตทริบิวต์ (Attributes)
    7.3 โดเมน (Domain)
    7.4 ทูเพิล (Tuple)
    7.5 ดีกรี (Degree)
    7.6 คาร์ดินัลลิตี้ (Cardinality)
    7.7 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)
8. กฎ 12 ข้อบนฐานข้อมูลเชิงสัมพันธ์ของคอดด์ ประกอบด้วยอะไรบ้าง จงอธิบาย
9. จงบอกคุณสมบัติของรีเลชั่น
10. จงอธิบายคีย์ต่อไปนี้
    10.1 ซูเปอร์คีย์ (Superkey)
    10.2 คีย์คู่แข่ง (Candidated Key)
    10.3 คีย์หลัก (Primary Key)
    10.4 คีย์นอก (Foreign Key)
    10.5 คีย์รวม (Composite Key)

เอกสารอ้างอิง

ปราลี มณีรัตน์. (2552). *การจัดการฐานข้อมูลธุรกิจ* (พิมพ์ครั้งที่ 1). กรุงเทพฯ: มหาวิทยาลัยศรีปทุม.

พนิดา พานิชกุล และ ณัฐพงษ์ วารีประเสริฐ. (2552). *การออกแบบ พัฒนา และดูแลระบบฐานข้อมูล* (พิมพ์ครั้งที่ 1). กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์.

วิเชียร เปรมชัยสวัสดิ์. (2550). *ระบบฐานข้อมูล* (พิมพ์ครั้งที่ 11). กรุงเทพฯ: สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น).

โอภาส เอี่ยมสิริวงศ์. (2558). *ระบบฐานข้อมูล.* กรุงเทพฯ: ซี เอ็ด ยูเคชั่น.

Chittayasothorn, S. (2007). *Relational Database Design.* Thailand: Institute of Thai Information Technology.

Date, C. J. (2004). *Introduction to Database Systems* (8th ed.). USA: Pearson Education.

Date, C. J. (2013). *Database Design and Relational Theory : Normals Forms and All That Jazz*. USA: O'Reilly Media.

Eric, R., & Jim, R. W. (2012). *Seven Databases in Seven Weeks : A Guide to Modern Databases and the NoSQL Movement.* USA: The Pragmatic Programmers.

Hoffer, J. A., Prescott, M. B., & McFadden, F. R. (2005). *Modern Database Management* (7th ed.). USA: Pearson Education.

Ramez, E., & Shamkant, B. N. (2013). *Fundamentals of Database Systems* (6th ed.). USA: Pearson Education.

Sikha, B., & Richard, E. (2012). *Database Design Using Entity-Relationship Diagrams* (2nd ed.). UK: Taylor & Francis Ltd.