

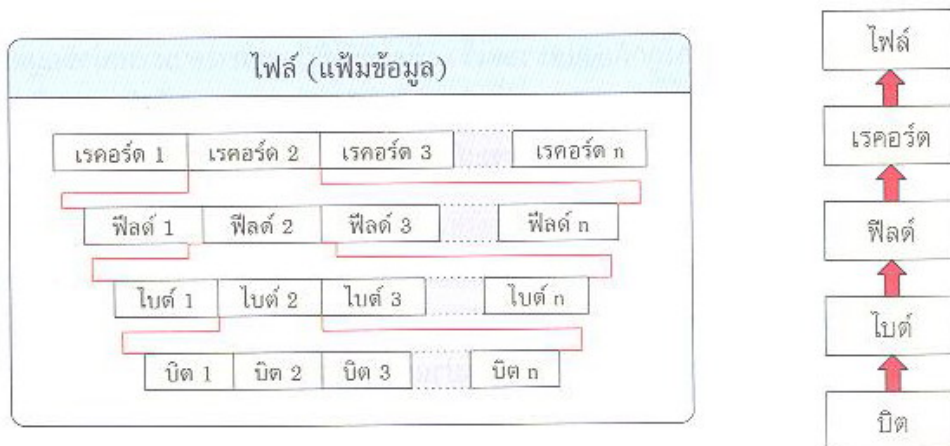
ใบความรู้ที่ 1.2

เรื่อง ความรู้เบื้องต้นเกี่ยวกับฐานข้อมูล

2. ฐานข้อมูล

ฐานข้อมูล คือ กลุ่มข้อมูลที่มีความสัมพันธ์กันและถูกเก็บรวบรวมไว้ในที่เดียวกันอย่างเป็นระบบ เพื่อนำไปใช้ในวัตถุประสงค์อย่างใดอย่างหนึ่งโดยกลุ่มผู้ใช้ตั้งแต่หนึ่งกลุ่มขึ้นไป

ตัวอย่างฐานข้อมูลที่เกี่ยวข้องกับชีวิตประจำวันที่เห็นชัด คือ สมุดโทรศัพท์ซึ่งแสดงรายชื่อผู้ใช้โทรศัพท์โดยเรียงลำดับตามชื่อและนามสกุล จาก ก-ข วิธีค้นหาเบอร์โทรศัพท์ก็คือ ต้องเริ่มต้นที่อักษรตัวแรกของชื่อที่ต้องการแล้วหาไปจนพบชื่อนั้น ถ้ามีชื่อซ้ำกันก็ต้องไปหาที่นามสกุลต่อไปอีกจนกว่าจะได้ชื่อและนามสกุลที่ต้องการ ในทางกลับกันถ้าจะค้นหาชื่อจากเบอร์โทรศัพท์ หรือนามสกุล หรือจากที่อยู่ ถ้าค้นหาด้วยมือ อาจต้องหาจากสมุดโทรศัพท์ทั้งเล่มซึ่งใช้เวลานานมาก แต่ถ้านำโปรแกรมจัดการฐานข้อมูลมาช่วยจัดเก็บข้อมูลลงในคอมพิวเตอร์ จะช่วยให้ค้นหาข้อมูลได้อย่างรวดเร็ว นอกจากนี้การบำรุงรักษาฐานข้อมูลไม่ว่าจะเป็นการเพิ่ม ลบ หรือแก้ไขข้อมูล จะทำได้ง่ายขึ้นด้วย



รูปที่ 1.3 แสดงโครงสร้างข้อมูลที่นำมาประมวลผลด้วยคอมพิวเตอร์

- บิต (Bit) ย่อมาจาก Binary digit เป็นหน่วยข้อมูลที่เล็กที่สุด แทนด้วยเลขฐานสอง (0 หรือ 1)
- ไบต์ (Byte) คือกลุ่มของบิตที่แทนตัวอักษร ตัวเลข หรือสัญลักษณ์พิเศษ 1 ตัว (character) เช่น รหัส ASCII 1 ไบต์ ซึ่งเก็บบิต 01000001 จะหมายถึงตัวอักษร A
- ฟิลด์ (Field) หรือเขตข้อมูล หมายถึงกลุ่มของตัวอักษร ตัวเลข หรือสัญลักษณ์พิเศษที่นำมารวมกันแล้วมีความหมาย เช่น ฟิลด์ชื่อสินค้า เก็บกลุ่มตัวอักษรที่แสดงชื่อสินค้า ฟิลด์ราคาสินค้า เก็บกลุ่มตัวเลขที่แสดงราคาสินค้าต่อหน่วย เป็นต้น

- เรคอร์ด (Record) หรือระเบียบข้อมูล ประกอบด้วยกลุ่มของฟิลด์ที่มีความสัมพันธ์กัน เช่น เรคอร์ดพนักงานจะมีฟิลด์รหัสพนักงาน ชื่อ นามสกุล ตำแหน่ง สถานภาพสมรส วันเข้าทำงาน ที่อยู่ และอื่นๆ โดย 1 เรคอร์ดจะเก็บข้อมูลพนักงาน 1 คน ในฟิลด์เดียวกันของทุกๆเรคอร์ดจะต้องเก็บข้อมูลชนิดเดียวกัน เช่น ฟิลด์ชื่อพนักงาน จะเก็บข้อมูลตัวอักษรเท่านั้น ส่วนฟิลด์ วันเข้าทำงาน จะเก็บข้อมูล วัน/เดือน/ปี ที่เข้าทำงานของพนักงานแต่ละคน เป็นต้น
- ไฟล์ (File) หรือแฟ้มข้อมูล คือกลุ่มเรคอร์ดหลายๆเรคอร์ดที่เก็บข้อมูลซึ่งเป็นเรื่องเดียวกัน เช่น ไฟล์ข้อมูลพนักงาน ไฟล์ข้อมูลลูกค้า ไฟล์ข้อมูลการสั่งซื้อสินค้า ไฟล์ข้อมูลสินค้า เป็นต้น ตัวอย่างไฟล์ข้อมูลที่ประกอบด้วยเรคอร์ดพนักงานทั้งหมดในบริษัทแห่งหนึ่ง โดยแต่ละเรคอร์ดจะมีฟิลด์ รหัส ชื่อ นามสกุล ที่อยู่ จังหวัด และรหัสไปรษณีย์

ชื่อฟิลด์ (Field Name)

รหัส	ชื่อ	นามสกุล	ที่อยู่	จังหวัด	รหัสไปรษณีย์
101	สุจิตรา	วัฒนไพศาล	355/100 หมู่ 2 ถนนเอกชัย-บางบอน	สมุทรสาคร	74000
102	ชลธิชา	บุญทวีพร	1234 หมู่ 7 ซอยวัดนินสุชาราม ถนนวงแหวน	กรุงเทพฯ	10160
103	อารีรัตน์	บุญเสริม	740 ถนนลำลูกกา	ปทุมธานี	12150
104	บุษบา	ทรัพย์รุ่งเรือง	110/900 ถนนวิภาวดีรังสิต 16/13	กรุงเทพฯ	10900
105	สุชาติ	โชคอนันต์	888/888 หมู่ 7 ถนนบางขุนเทียน	กรุงเทพฯ	10150
106	รุ่งทิวา	แก้วสิโล	1/1 ซอยพัฒนาเวสต์ 12	กรุงเทพฯ	10110
107	พนิดา	จันทร์แจ่มฟ้า	699 ถนนบรมราชชนนี	กรุงเทพฯ	10170
108	ชัยวัฒน์	จงประเสริฐ	1777 ซอยจรัญสนิทวงศ์ 66/1 ถนนจรัญสนิทวงศ์	กรุงเทพฯ	10700
109	การะเกด	ไชยสวัสดิ์	766 หมู่ 9 ถนนเทพารักษ์ กม.19	สมุทรปราการ	10540
110	จันทร์จิรา	งามขำ	1222 หมู่ 5 ซอยศิริชัย ถนนพุทธมณฑลสาย 5	สมุทรสาคร	74130

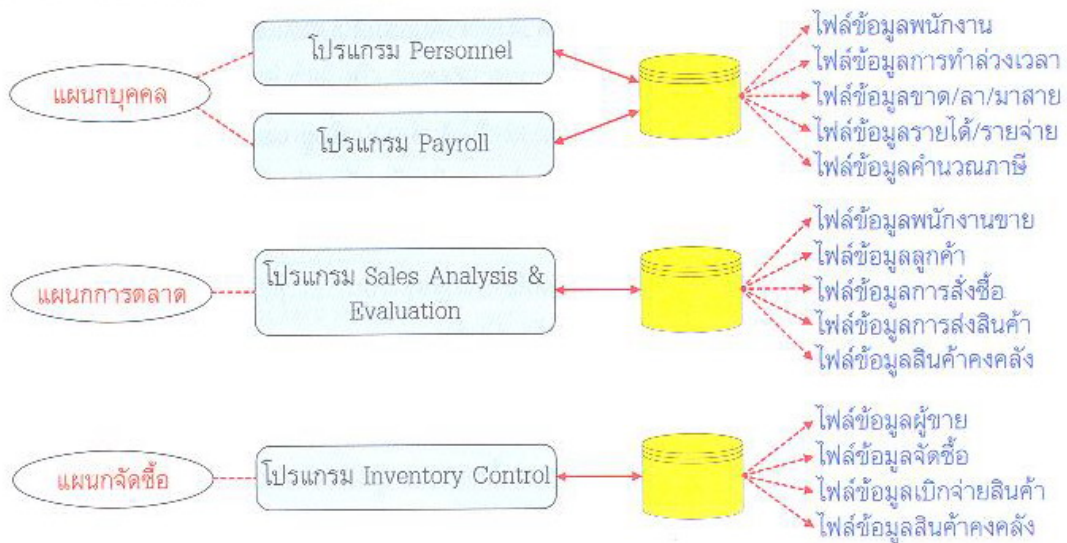
ฟิลด์ (Field)

เรคอร์ด (Record)

รูปที่ 1.4 แสดงชื่อฟิลด์ และเรคอร์ด

ความสำคัญของฐานข้อมูล

ยุคแรกก็นำระบบคอมพิวเตอร์มาใช้ประมวลผลข้อมูลนั้น โครงสร้างการจัดเก็บข้อมูลในเครื่องยังมีลักษณะคล้ายคลึงกับการจัดเก็บข้อมูลในกระดาษ คือ ข้อมูลแต่ละประเภทถูกเก็บแยกกันในลักษณะของไฟล์หรือแฟ้มข้อมูล โดยแต่ละแผนกหรือหน่วยงานจะเก็บข้อมูลและมีโปรแกรมของตนเองที่ใช้ดึงข้อมูลจากไฟล์ต่างๆมาประมวลผล



รูปที่ 1.5 แสดงระบบการจัดเก็บข้อมูลในลักษณะไฟล์หรือแฟ้มข้อมูล

แผนภาพแสดงระบบการจัดเก็บข้อมูลของบริษัทแห่งหนึ่ง จะเห็นว่าแต่ละแผนกเก็บข้อมูลของตัวเองไว้ในไฟล์ที่แยกจากกันทำให้มีข้อมูลบางส่วนซ้ำกัน ตัวอย่างเช่น ไฟล์ข้อมูลพนักงาน ของแผนกบุคคล จะเก็บข้อมูลของพนักงานทุกคนในบริษัท จึงมีข้อมูลพนักงานชายอยู่ด้วย ทำให้ซ้ำกับข้อมูลในไฟล์ข้อมูลพนักงานชายของแผนกการตลาด ส่วนไฟล์ข้อมูลสินค้าคงคลังของแผนกการตลาดและแผนกจัดซื้อจะมีข้อมูลเหมือนกัน เป็นต้น โดยแต่ละแผนกจะดึงข้อมูลจากไฟล์มาประมวลผลและออกรายงานโดยใช้โปรแกรมของตัวเอง

ข้อดีในการประมวลผลแบบไฟล์ข้อมูล คือ

- แต่ละแผนกสามารถเขียนโปรแกรมประมวลผลข้อมูลในรูปแบบที่ต้องการ ได้อย่างอิสระ
- การดึงข้อมูลมาใช้ทำได้อย่างรวดเร็วเนื่องจากต่างคนต่างเก็บ
- ลดต้นทุนในส่วนของการวางระบบคอมพิวเตอร์ เนื่องจากข้อมูลที่เก็บเป็นข้อมูลที่ใช้งานในฝ่ายเท่านั้น จึงไม่มีความซับซ้อนจนต้องใช้ระบบการจัดการข้อมูลที่มีประสิทธิภาพสูงมาก

สำหรับข้อเสียในการประมวลผลแบบนี้ก็คือ

- เกิดความซ้ำซ้อนของข้อมูล (Redundancy) ซึ่งจะนำไปสู่ปัญหาความขัดแย้งของข้อมูล (Data Inconsistency) ในภายหลังได้ ถ้ามีการแก้ไขข้อมูลในไฟล์หนึ่ง แต่ไม่ได้แก้ไขข้อมูลนั้นในไฟล์อื่นๆด้วย
- เกิดความไม่เป็นอิสระของข้อมูล (Data Dependency) เพราะโปรแกรมที่ใช้ในแต่ละแผนกจะผูกพันกับโครงสร้างการจัดเก็บและวิธีเรียกใช้ข้อมูล ถ้าเปลี่ยนแปลงโครงสร้างการจัดเก็บ

ข้อมูล โปรแกรมทุกโปรแกรมที่เรียกใช้ไฟล์ข้อมูลนั้นจะต้องถูกแก้ไขด้วย ซึ่งอาจเกิดข้อผิดพลาดได้

- ไม่มีการควบคุมจากศูนย์กลาง เนื่องจากแต่ละแผนก สามารถเรียกใช้ข้อมูลได้อย่างอิสระ โดยไม่ต้องผ่านการดูแลจากศูนย์กลาง จึงไม่สามารถควบคุมสิทธิในการแก้ไขข้อมูลของผู้ใช้แต่ละคนให้เป็นมาตรฐานเดียวกัน

จากปัญหาข้างต้น จึงได้มีการพัฒนาระบบฐานข้อมูล เพื่อรวบรวมข้อมูลที่มีความสัมพันธ์กัน แต่แยกกันอยู่แต่ละไฟล์มาเก็บไว้ที่เดียวกัน เพื่อให้แต่ละแผนกสามารถเข้าถึงและใช้ข้อมูลร่วมกันได้ การดูแลรักษาระบบงานและปรับปรุงข้อมูลให้มีความถูกต้องและทันสมัยอยู่เสมอจะทำได้ง่ายขึ้น เพราะข้อมูลไม่อยู่กระจัดกระจาย สามารถเรียกใช้ข้อมูลได้ทันทีที่ต้องการ ช่วยให้ผู้บริหารวางแผนและตัดสินใจได้อย่างรวดเร็วและทันต่อสถานการณ์

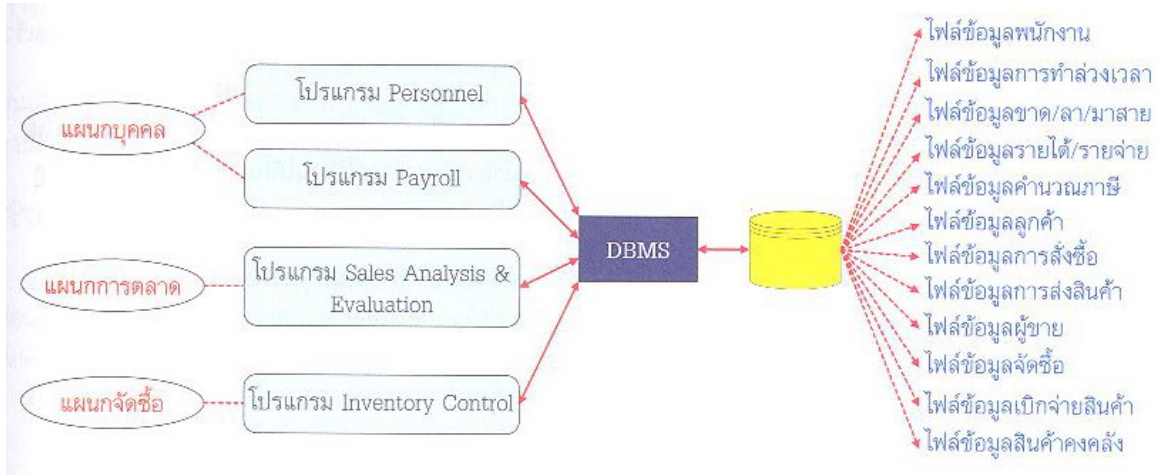
การออกแบบฐานข้อมูลที่ดีนั้น ผู้ออกแบบต้องจัดกลุ่มและแยกประเภทข้อมูลที่จะนำมาใช้ โดยให้ข้อมูลแต่ละกลุ่มมีความซ้ำซ้อนกันน้อยที่สุด (ดูรายละเอียดในหัวข้อ การทำ Normalization) เพื่อให้ฐานข้อมูลมีขนาดเล็กที่สุด แต่ในขณะเดียวกันก็ได้ความหมายมากที่สุดเช่นกัน ผู้ออกแบบต้องสามารถกำหนดความสัมพันธ์ระหว่างกลุ่มข้อมูลต่างๆ ได้ ซึ่งความสัมพันธ์นี้เองจะเป็นจุดเริ่มต้นที่นำไปสู่การพัฒนาเป็นระบบฐานข้อมูล

องค์ประกอบในการทำงานร่วมกับฐานข้อมูล

นอกจากฐานข้อมูลจะเกิดจากการนำไฟล์หรือกลุ่มข้อมูลต่างๆ มาเก็บไว้ในที่เดียวกันแล้ว ยังมีปัจจัยสำคัญอื่นๆ ที่จำเป็นต่อการทำงานร่วมกับฐานข้อมูลด้วยดังนี้

- กลุ่มข้อมูลที่นำมาเก็บรวมกันจะต้องมีความเกี่ยวข้องกัน โดยจะต้องระบุความสัมพันธ์ระหว่างกลุ่มข้อมูลและใช้ประโยชน์จากความสัมพันธ์นั้นได้ สำหรับโปรแกรมระบบจัดการฐานข้อมูล Microsoft Access 2007 จะเรียกกลุ่มข้อมูลที่ผู้ใช้มองเห็นในรูปตารางว่า Table ซึ่งต่อจากนี้ไปผู้เขียนจะขอใช้คำว่า Table แทนคำว่า ตาราง
- ต้องมีโปรแกรมระบบจัดการฐานข้อมูล โครงสร้างการจัดเก็บข้อมูลในระบบฐานข้อมูล ถ้าเขียนโปรแกรมควบคุมการทำงานขึ้นใช้เองจะค่อนข้างยุ่งยากและใช้เวลามาก ต้องใช้บุคลากรที่มีความชำนาญด้านนี้โดยเฉพาะจึงได้มีการพัฒนาเครื่องมือที่เรียกว่า โปรแกรมระบบจัดการฐานข้อมูล (DataBase Management System : DBMS) เพื่อนำมาช่วยให้การสร้าง เรียกใช้ และแก้ไขฐานข้อมูลทำได้ง่ายแต่มีประสิทธิภาพสูง DBMS จะทำหน้าที่เป็นสื่อกลางระหว่างผู้ใช้

กับฐานข้อมูล เพื่อควบคุมการทำงานที่กล่าวข้างต้น โดยผู้ใช้ไม่จำเป็นต้องทราบ โครงสร้างและวิธีจัดเก็บข้อมูลทางกายภาพ (physical storage)



รูปที่ 1.6 แสดงระบบการจัดเก็บข้อมูลในลักษณะฐานข้อมูล

ข้อดีของการนำระบบฐานข้อมูลมาใช้

- ลดปัญหาความซ้ำซ้อนกันของข้อมูล เนื่องจากข้อมูลทั้งหมดจะถูกเก็บไว้ที่เดียวกัน ไม่ได้แยกไว้คนละที่เหมือนในระบบไฟล์ข้อมูล ถ้าจำเป็นต้องเก็บข้อมูลตัวเดียวกันไว้มากกว่า 1 ที่ (มากกว่า 1 Table) เช่น ต้องเก็บรหัสลูกค้าไว้ใน Table ลูกค้าและ Table การสั่งซื้อ เพื่อให้ทราบว่ามีลูกค้ารายใดบ้างที่สั่งซื้อสินค้า ในกรณีนี้ DBMS จะทำหน้าที่ตรวจสอบได้ว่ามีข้อมูลใดบ้างที่ซ้ำซ้อนกันและถูกเก็บไว้ที่ใดบ้าง
- ลดปัญหาความขัดแย้งกันของข้อมูล จากการที่ข้อมูลมีความซ้ำซ้อนกันน้อยลง ทำให้ลดปัญหาการเก็บข้อมูลตัวเดียวกันแต่ค่าไม่ตรงกัน และถ้ามีการแก้ไขข้อมูลใน Table ใด DBMS จะสามารถควบคุมให้ข้อมูลตัวเดียวกันนั้นที่อยู่ใน Table อื่นๆถูกแก้ไขให้ตรงกันทั้งหมดได้ โดยผู้ใช้ไม่ต้องตามไปแก้ไขด้วยตัวเอง
- สามารถควบคุมการคงสภาพความถูกต้องของข้อมูล หมายถึง ความถูกต้องที่สอดคล้องและสมเหตุสมผลตรงเงื่อนไขที่กำหนดไว้ เช่น ถ้ากำหนดให้ราคาสินค้าเป็นค่าติดลบไม่ได้ เมื่อผู้ใช้ป้อนข้อมูลราคาสินค้า DBMS จะตรวจสอบว่าข้อมูลที่ป้อนเข้าไปนั้นตรงตามกฎเกณฑ์นี้หรือไม่ ถ้าไม่ตรงกัน ก็จะแสดงข้อผิดพลาดให้ทราบ และไม่บันทึกข้อมูลนั้นลงฐานข้อมูลจนกว่าผู้ใช้จะแก้ไขให้ถูกต้อง

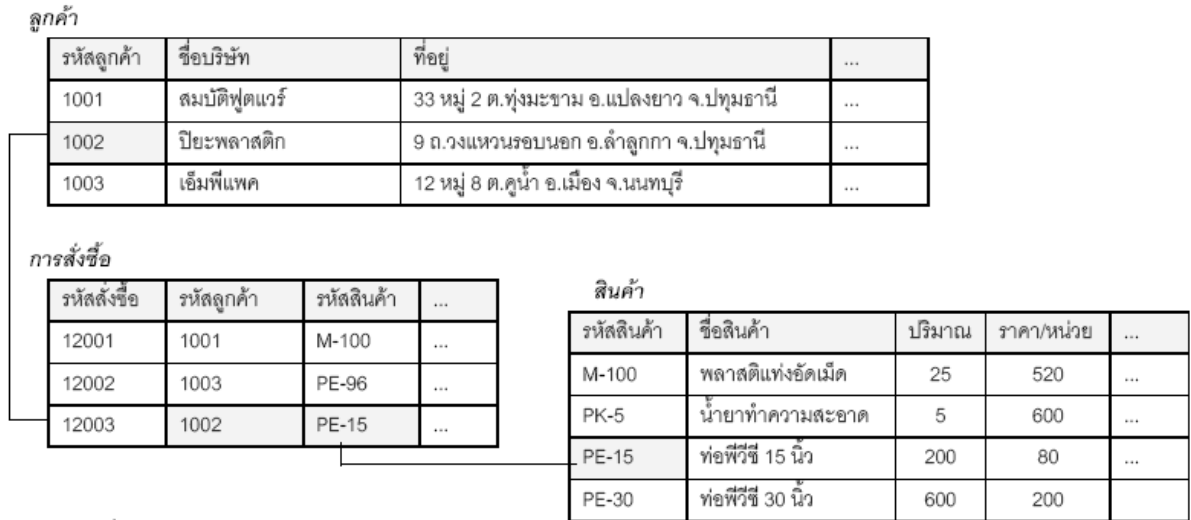
- ทำให้เกิดความเป็นอิสระของข้อมูล เนื่องจากโปรแกรมที่เขียนขึ้นมา ไม่ต้องยึดติดกับโครงสร้างการจัดเก็บและวิธีเรียกใช้ข้อมูลอีกต่อไป ทำให้การปรับปรุงโครงสร้างข้อมูลในระยะยาวทำได้ง่ายขึ้น โดยไม่ต้องกังวลกับการแก้ไขโปรแกรมให้สอดคล้องกับโครงสร้างข้อมูลนั้น หรือถ้าจำเป็นต้องแก้ไขก็จะไม่ยุ่งยากและไม่มีข้อจำกัดมากเหมือนในระบบไฟล์ข้อมูล
- ทำให้สามารถใช้ข้อมูลร่วมกันได้ ถ้าฐานข้อมูลมีความสมบูรณ์และครบถ้วน จะช่วยให้โปรแกรมที่เขียนขึ้นสามารถดึงข้อมูลมาใช้ได้ โดยไม่ต้องแก้ไขโครงสร้างข้อมูล หรือถ้าจะต้องแก้ไขก็เพียงเล็กน้อยเท่านั้น
- ข้อมูลมีความเป็นมาตรฐาน เนื่องจากสามารถกำหนดชนิดและรูปแบบของข้อมูลตัวเดียวกันให้เหมือนกันได้ ไม่ว่าจะเก็บไว้ที่ส่วนใดของฐานข้อมูล ทำให้การนำข้อมูลไปใช้หรือแลกเปลี่ยนระหว่างฐานข้อมูลทำได้ง่ายขึ้น
- สามารถตอบสนองความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ ผู้บริหารฐานข้อมูล (DataBase Administrator : DBA) ซึ่งควบคุมและบริหารระบบฐานข้อมูล สามารถจัดการฐานข้อมูลเพื่อตอบสนองให้บริการต่อผู้ใช้โดยส่วนรวมได้อย่างมีประสิทธิภาพ เช่น เก็บข้อมูลที่มีความสำคัญ และถูกเรียกใช้อยู่เสมอไว้ในสื่อที่มีความเร็วสูงเพื่อให้เรียกใช้ได้อย่างรวดเร็วเป็นต้น
- สามารถสร้างระบบความปลอดภัยให้กับ DBA สามารถกำหนดสิทธิการเข้าใช้ฐานข้อมูลให้กับผู้ใช้แต่ละคนในระดับต่างๆ เช่น กำหนดว่าจะอนุญาตให้ใครเข้าไปใช้ฐานข้อมูลได้บ้าง และสามารถใช้งานได้ในระดับใด เช่น ให้เรียกดูข้อมูลและแก้ไขได้ หรือให้เรียกดูข้อมูลได้อย่างเดียวแต่แก้ไขไม่ได้ เป็นต้น

2. ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

เป็นฐานข้อมูลที่ได้รับคามนิยมมากที่สุดในปัจจุบัน เนื่องจากใช้งานง่าย เพราะผู้ใช้จะทำงานร่วมกับข้อมูลในรูปของตารางหรือ Table ที่ข้อมูลแต่ละแถวจะหมายถึงแต่ละเรคอร์ด ส่วนข้อมูลแต่ละคอลัมน์จะหมายถึงฟิลด์

ฐานข้อมูลแบบสัมพันธ์ เป็นฐานข้อมูลที่เก็บกันเป็นรูปตาราง เป็นการเก็บข้อมูลแบบ 2 มิติ ที่ประกอบด้วยข้อมูลแต่ละแถวในแนวนอน ทำให้สามารถเชื่อมโยงหรือสร้างความสัมพันธ์ระหว่างกลุ่มข้อมูลหรือ ตารางที่เกี่ยวข้องในฐานข้อมูลเดียวกันได้

สำหรับโครงสร้างของฐานข้อมูลเชิงสัมพันธ์ที่สามารถเรียกใช้ข้อมูลได้ในรูปของตารางนี้ ผู้ใช้ไม่จำเป็นต้องทราบว่าข้อมูลจะถูกเก็บจริงในลักษณะใด แต่สามารถกำหนดความสัมพันธ์ระหว่างตารางต่าง ๆ ในฐานข้อมูลด้วยตนเอง



รูปที่ 1.7 แสดงตัวอย่างฐานข้อมูลเชิงสัมพันธ์

จากตารางจะเห็นตารางลูกค้าและตารางสั่งซื้อ มีเขตข้อมูลรหัสลูกค้าเชื่อมความสัมพันธ์ระหว่าง 2 ตารางนี้ ถ้าต้องการทราบชื่อและที่อยู่ของลูกค้ารหัส 1002 ในตารางการสั่งซื้อ เราจะนำรหัส 1002 ไปค้นตารางลูกค้าที่มีรหัสตรงกัน

2.2 ศัพท์ที่ควรรู้เกี่ยวกับฐานข้อมูล

เอนทิตี (Entity) และ แอตทริบิวต์ (Attribute)

เอนทิตี หมายถึง สิ่งต่าง ๆ ที่อ้างถึงในฐานข้อมูล ประกอบด้วยกลุ่มข้อมูลประเภทเดียวกันที่เป็นสมาชิกของเอนทิตินั้น เช่น ถ้ากล่าวเอนทิตีพนักงานจะหมายถึงกลุ่มคนทั้งหมดที่เป็นพนักงาน

แอตทริบิวต์ หรือ ฟิลด์ หมายถึง สิ่งที่ใช้บอกองค์ประกอบหรือเนื้อหา (subject) ของเอนทิตี เช่น เอนทิตีพนักงานจะประกอบด้วยแอตทริบิวต์ต่าง ๆ คือ รหัสประจำตัว ชื่อ นามสกุล แผนก วุฒิการศึกษา วันเข้าทำงาน ที่อยู่ เป็นต้น

รีเลชัน (Relation) หมายถึง รูปแบบของตารางแบบ 2 มิติ ที่ประกอบด้วยแต่ละแถวที่เรียกว่า ทูเพิล (Tuple) และแต่ละคอลัมน์ที่เรียกว่า แอตทริบิวต์ (Attribute)

ตาราง (เอนทิตี) : พนักงาน

รหัสประจำตัว	ชื่อ	นามสกุล	ตำแหน่ง	วุฒิการศึกษา	วันเข้าทำงาน	ที่อยู่
101	นายสมรักษ์	คำหอม	พนักงานขาย	ปวส.	15 ก.พ.2545
102	น.ส.มาลัย	มะลิวัลย์	พนักงานบัญชี	ปริญญาตรี	1 มี.ค. 2546
103	น.ส.ยุพิน	สมศรีคำ	พนักงานขาย	ปวส.	22 ธ.ค. 2548

ฟิลด์ (แอตทริบิวต์)

รูปที่ 1.8 แสดงตัวอย่างเอนทิตีพนักงาน และคำศัพท์ที่ควรทราบ

ความสัมพันธ์ (Relationship) การกำหนดความสัมพันธ์ระหว่างเอนทิตี หรือ ตาราง ในฐานข้อมูลเชิงสัมพันธ์จะมี 3 ลักษณะ

ความสัมพันธ์แบบ 1:1 (One-to-One) เป็นความสัมพันธ์ที่เรคคอร์ดหนึ่งเรคคอร์ดในตารางใด ๆ สามารถจับคู่กับเรคคอร์ดในอีกตารางหนึ่งได้เพียงเรคคอร์ดเดียวเท่านั้น หรือเป็นการจับคู่กันตัวต่อตัว



พนักงาน

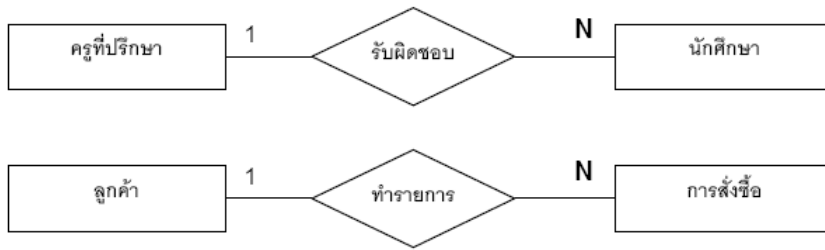
รหัสพนักงาน	ชื่อ	นามสกุล	ที่อยู่	...
1001	สมพร	ความดี	52 หมู่ 12 ต.ลาดยาว อ.เมือง ปทุมธานี	...
1002	สมหมาย	รักดี	22 หมู่ 4 ถ.ราษฎร์ อ.เมือง จ.นนทบุรี	...
1003	สุเมธ	คมขำ	122/125 แพลตตินแดง 3 เขตดินแดง กรุงเทพฯ	...

ข้อมูลส่วนตัว

รหัสพนักงาน	เงินเดือน	เลขประกันสังคม
1001	15,000	662-55-2552
1002	13,550	252-66-5259
1003	24,000	141-36-4585

รูปที่ 1.9 แสดงความสัมพันธ์แบบ หนึ่งต่อหนึ่ง

ความสัมพันธ์แบบ 1:N (One-to-Many) เป็นความสัมพันธ์ที่เรคคอร์ดในตารางใด ๆ สามารถจับคู่กับเรคคอร์ดในอีกตารางหนึ่งได้หลายเรคคอร์ด ตัวอย่างเช่น อาจารย์ที่ปรึกษาที่มีนักศึกษาในความรับผิดชอบได้หลายคน แต่นักศึกษาแต่ละคนจะมีอาจารย์ที่ปรึกษาได้เพียงคนเดียวเท่านั้น



ลูกค้า

รหัสลูกค้า	ชื่อบริษัท	ที่อยู่	...
1001	สมบัติฟูดแวร์	33 หมู่ 2 ต.ทุ่งมะขาม อ.แปลงยาว จ.ปทุมธานี	...
1002	ปิยะพลาสติก	9 ถ.วงแหวนรอบนอก อ.ลำลูกกา จ.ปทุมธานี	...
1003	เอ็มพีแพค	12 หมู่ 8 ต.คูน้ำ อ.เมือง จ.นนทบุรี	...

การสั่งซื้อ

รหัสสั่งซื้อ	รหัสลูกค้า	รหัสสินค้า	...
12001	1001	M-100	...
12002	1003	PE-96	...
12003	1002	PE-15	...

รูปที่ 1.20 แสดงตัวอย่างความสัมพันธ์แบบหนึ่งต่อกลุ่ม

ความสัมพันธ์แบบ M:N (Many-to-Many) เป็นความสัมพันธ์ที่เรคคอร์ดหลาย ๆ เรคคอร์ดในตารางหนึ่ง มีความสัมพันธ์กับอีกหลาย ๆ เรคคอร์ดในอีกตารางหนึ่งพร้อมกัน เช่น ความสัมพันธ์ระหว่างตารางลูกค้า และตารางสินค้า ลูกค้าหนึ่งคนสามารถซื้อสินค้าได้หลายชนิด ในขณะที่สินค้าแต่ละชนิดก็จะถูกซื้อโดยลูกค้าหลาย ๆ คนได้ด้วย ถ้าเรานำตารางทั้งสองนี้มาเชื่อมโยงกันโดยตรงจะเห็นว่าทำไม่ได้ เนื่องจากไม่มีฟิลด์ที่เป็นฟิลด์ร่วมของทั้ง 2 ตาราง การสร้างความสัมพันธ์แบบ M:N จึงต้องใช้ตารางอื่นมาช่วยเป็นสะพานในการเชื่อมโยง คือ ตาราง การสั่งซื้อ



รูปที่ 1.21 แสดงตัวอย่างความสัมพันธ์แบบกลุ่มต่อกลุ่ม

คีย์ (Key)

เพื่อเพิ่มประสิทธิภาพในการจัดการข้อมูลควรกำหนดคีย์ (Key) ให้กับตารางเพื่อใช้จำแนก เรคคอร์ด และกำหนดความสัมพันธ์ระหว่างตาราง เช่น กำหนดรหัสประจำตัวให้กับพนักงานทุกคนในตารางพนักงาน คีย์ที่ใช้ในระบบฐานข้อมูลมีหลายอย่างดังนี้

คีย์หลัก (Primary Key) หมายถึง เขตข้อมูลย่อยที่ไม่ซ้ำกันเลยในแต่ละแถวข้อมูล สามารถที่จะบ่งชี้ระเบียบแต่ละระเบียบได้ ค่าของคีย์หลักจะเป็นค่าว่าง (NULL) ไม่ได้

เป็นคีย์ที่กำหนดขึ้น โดยจะต้องไม่มีข้อมูลซ้ำกันโดยเด็ดขาดในตารางนั้น เช่น ฟیلด์รหัสพนักงานในตารางข้อมูลพนักงาน หรือฟیلด์รหัสสินค้าในตารางข้อมูลสินค้านำไปใช้จัดเรียงและแยกแยะข้อมูลในแต่ละเรคคอร์ดออกจากกัน

พนักงาน

รหัสพนักงาน	ชื่อ	นามสกุล	ที่อยู่	...
1001	สมพร	ความดี	52 หมู่ 12 ต.ลาดยาว อ.เมือง ปทุมธานี	...
1002	สมหมาย	รักดี	22 หมู่ 4 ถ.อารักษ์ อ.เมือง จ.นนทบุรี	...
1003	สุเมธ	คมช้ำ	122/125 แพลตดินแดง 3 เขตดินแดง กรุงเทพฯ	...

↑
ใช้ฟیلด์นี้เป็นคีย์หลัก

รูปที่ 1.22 แสดงคีย์หลัก

ดัชนี (Index) หรือ คีย์รอง (Secondary Key) เป็นคีย์ที่ใช้ค้นหาหรือจัดเรียงกลุ่มเรคคอร์ดที่มีจำนวนมากได้อย่างรวดเร็ว เช่นค้นหาชื่อและนามสกุลของพนักงานใน Table ถ้าไม่กำหนดฟیلด์ชื่อและนามสกุลเป็นดัชนีไว้ก่อน DBMS จะค้นหาตั้งแต่เรคคอร์ดแรกไปจนถึงเรคคอร์ดที่ต้องการ ฟیلด์ที่เป็นดัชนีอาจมีข้อมูลซ้ำกันได้ (ต่างกับฟیلด์ที่เป็นคีย์หลักที่ข้อมูลจะซ้ำกันไม่ได้ คีย์หลักทุกตัวจะมีคุณสมบัติเป็นดัชนี แต่ดัชนีไม่จำเป็นต้องเป็นคีย์หลัก)

คีย์คู่แข่ง (Candidate Key) หมายถึง เขตข้อมูลย่อยที่ไม่ซ้ำกันเลยในแต่ละแถวข้อมูล มีมากกว่า 1 ฟیلด์ ซึ่งสามารถทำหน้าที่เป็นคีย์หลักแทนกันได้

ถ้าในตารางหนึ่งมีฟیلด์ที่มีคุณสมบัติที่สามารถใช้คีย์หลักแทนกันได้ จะเรียกคีย์เหล่านั้นว่าคีย์คู่แข่ง เช่นในตารางพนักงาน ถ้าไม่มีชื่อพนักงานซ้ำกันเลย ก็สามารถใช้ฟیلด์รหัสพนักงาน หรือฟیلด์ชื่อพนักงานเป็นคีย์หลักได้ ทำให้ทั้งสองฟیلด์นี้กลายเป็นฟیلด์คู่แข่งกัน ถ้าใช้รหัสพนักงานเป็นคีย์หลักก็ทำให้ชื่อพนักงาน เป็นคีย์รอง (Alternate Key)

พนักงาน

รหัสพนักงาน	ชื่อ	นามสกุล	ที่อยู่	...
1001	สมพร	ความดี	52 หมู่ 12 ต.ลาดยาว อ.เมือง ปทุมธานี	...
1002	สมหมาย	รักดี	22 หมู่ 4 ถ.อารักษ์ อ.เมือง จ.นนทบุรี	...
1003	สุเมธ	คมช้ำ	122/125 แพลตดินแดง 3 เขตดินแดง กรุงเทพฯ	...

↑
2 ฟیلด์นี้เป็นฟیلด์คู่แข่ง ในกรณีนี้ที่ ฟیلด์รหัสพนักงาน หรือ ฟیلด์ชื่อ ไม่ซ้ำกัน

รูปที่ 1.23 แสดงคีย์คู่แข่ง

คีย์ร่วม (Composite Key) หรือคีย์ผสม หมายถึง การนำฟิลด์ ตั้งแต่ 2 ฟิลด์ ขึ้นไปมารวมกันเพื่อกำหนดให้เป็นคีย์หลัก

เนื่องจากในบางครั้งการสร้างคีย์หลักจากฟิลด์เดียวอาจมีโอกาที่จะเกิดข้อมูลซ้ำกันได้เช่นในตารางพนักงานด้านล่าง ตารางพนักงานหากไม่ได้กำหนดฟิลด์รหัสพนักงาน เราอาจใช้ฟิลด์ชื่อและนามสกุล ประกอบกันเป็นคีย์หลักของตารางก็ได้

พนักงาน

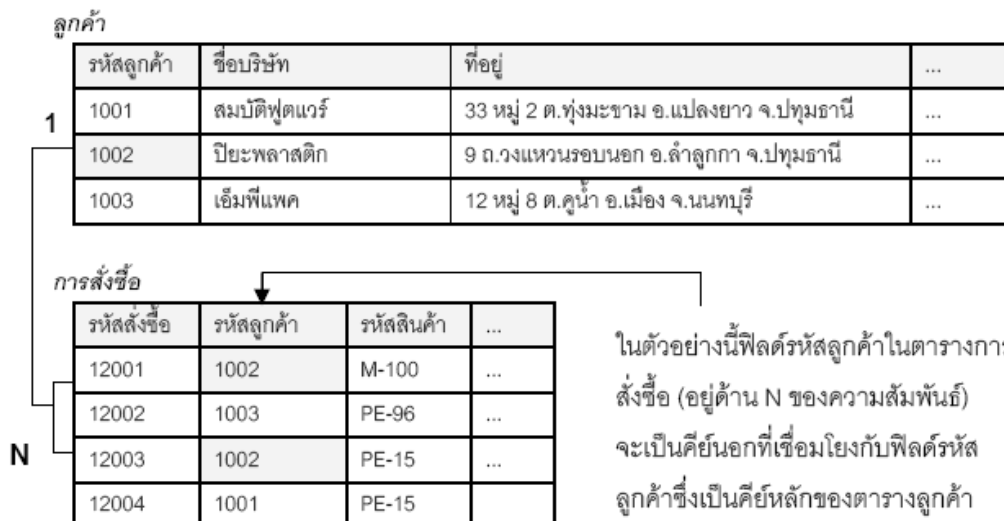
ชื่อ	นามสกุล	ที่อยู่	เบอร์โทรศัพท์	...
สมพร	ความดี	52 หมู่ 12 ต.ลาดยาว อ.เมือง ปทุมธานี	02-953-2222	...
สมหมาย	รักดี	22 หมู่ 4 ถ.อารักษ์ อ.เมือง จ.นนทบุรี	02-253-2552	...
สุเมธ	คมขำ	122/125 แฟลตดินแดง 3 เขตดินแดง กรุงเทพฯ	02-225-3987	...

หากไม่กำหนดฟิลด์รหัสพนักงาน อาจใช้ฟิลด์ชื่อและนามสกุล เป็นคีย์ร่วมกัน

รูปที่ 1.23 แสดงคีย์ร่วม

คีย์นอก (Foreign Key) หมายถึง คีย์หลักจากตารางภายนอก เป็นคีย์เพื่อสร้างความสัมพันธ์ระหว่างรีเลชัน ใช้เพื่อกำหนดให้ข้อมูลในรีเลชัน หนึ่งมีค่าตรงกับคีย์หลักของอีกรีเลชันหนึ่ง

เป็นคีย์ที่ใช้เชื่อมโยงตารางที่มีข้อมูลเกี่ยวข้องกัน เช่น ในตารางลูกค้าจะมีฟิลด์รหัสลูกค้าเป็นคีย์หลัก เราจะใช้รหัสลูกค้าในตารางลูกค้าเชื่อมโยงกับรหัสลูกค้าในตารางการสั่งซื้อ เพื่อที่จะได้ทราบชื่อและที่อยู่ของลูกค้าที่สั่งซื้อสินค้านั้น



รูปที่ 1.23 แสดงคีย์นอก



กฎที่ใช้ควบคุมการคงสภาพความถูกต้องของข้อมูล (Integrity Constraint)

เนื่องจากฐานข้อมูลเชิงสัมพันธ์ประกอบด้วยข้อมูลหลายๆ ตารางที่กำหนดให้มีความสัมพันธ์ในลักษณะต่าง ๆ ไม่ว่าจะเป็นแบบ One-to-One , One-to-Many หรือ Many-to-Many การลบเรคคอร์ดหรือการแก้ไขเปลี่ยนแปลงข้อมูลในตารางหนึ่งต้องมีผลกระทบต่อข้อมูลในตารางอื่นที่สัมพันธ์กันเพื่อให้ข้อมูลทั้งหมดที่เกี่ยวข้องมีความคงสภาพความถูกต้องของข้อมูลที่สอดคล้องตามเงื่อนไขที่กำหนดไว้ตลอดเวลาจะต้องมีกฎเกณฑ์ที่ใช้ควบคุมความถูกต้องซึ่งมี 2 ลักษณะ

- **กฎการคงสภาพของเอนทิตี (Entity Integrity Constraint)** ควบคุมการคงสภาพความถูกต้องของข้อมูลในตารางเดียวกันซึ่งจะต้องไม่มีค่าว่างในฟิลด์ใด ๆ ที่ประกอบขึ้นเป็นคีย์หลักของตารางนั้น

- **กฎการคงสภาพการอ้างอิง (Referential Integrity Constraint)** ควบคุมการคงสภาพความถูกต้องของความสัมพันธ์ระหว่างตาราง ซึ่งมีคีย์นอกอยู่ในตารางใด ข้อมูลที่อยู่ในคีย์นอกนั้นจะต้องเป็นข้อมูลที่อยู่ในคีย์หลักของอีกตารางหนึ่งด้วย ถ้าไม่เช่นนั้นแล้วข้อมูลในคีย์นอกจะต้องเป็นค่าว่าง