

การพัฒนาระบบฐานข้อมูลบัณฑิต



นายวิชัย พิชิตนารัตน์ 37054139
นายเสรี รักรวงษ์ 37054150
นายอภิชาติ พิรินทร์ 37054151



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน..... 33859
วัน, เดือน, ปี 17 ก.ย. 2542

ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVELOPMENT DATABASE SYSTEM OF GRADUATED STUDENT

MR. WICHAI PICHITTANARAT 37054139

MR. SEREE RAKWONG 37054150

MR. APICHAT PIRINANT 37054151

A Special Project Submitted in Partial Fulfillment of the
Requirement for the Degree of Bachelor of Science
Department of Mathematics and Computer Sciences
Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
1997

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาพิเศษเรื่อง การพัฒนาระบบฐานข้อมูลบัณฑิต
ชื่อนักศึกษา นายวิชัย พิชิตชนาวรัตน์ 37054139
นายเสรี รัทวงษ์ 37054150
นายอภิชาติ พิรินันท์ 37054151
ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษา อาจารย์วีระชัย ดันยะสิทธิ์

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นำปัญหาพิเศษฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาคณิตศาสตร์ประยุกต์ ประจำปีการศึกษา 2540



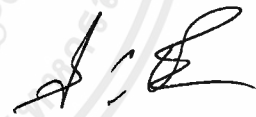
(รศ. ภัคคินี ชิตสกุล)

หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

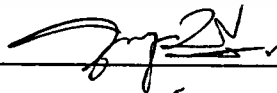
คณะกรรมการโครงการพิเศษ



(อาจารย์กฤษฎา ไตรสุรัตน์)
ประธานกรรมการ



(อาจารย์วีระชัย ดันยะสิทธิ์)
กรรมการและอาจารย์ที่ปรึกษา



(อาจารย์กฤษฎา บุศรา)

กรรมการ

ลิขสิทธิ์ของภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	การพัฒนาระบบฐานข้อมูลบัณฑิต		
ชื่อนักศึกษา	นายวิชัย	พิชิตธนารัตน์	37054139
	นายเสรี	รังกวงษ์	37054150
	นายอภิชาติ	พิรินันท์	37054151

อาจารย์ที่ปรึกษา อาจารย์วีระชัย ดันยะสิทธิ์

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์

ปีการศึกษา 2540

บทคัดย่อ

ระบบฐานข้อมูลบัณฑิต เป็นโปรแกรมที่พัฒนาเพื่อ จัดเก็บข้อมูลบัณฑิต ซึ่งเป็นประโยชน์ต่อบัณฑิต และ ทางคณะ เช่น การติดต่อแลกเปลี่ยนข้อมูลข่าวสาร เป็นต้น ซึ่งทำให้ผู้ใช้มีการใช้งานได้อย่างมีประสิทธิภาพ ประกอบด้วยการเพิ่ม ลบ แก้ไข และ ค้นหาข้อมูลซึ่งเป็นการสะดวกและง่ายต่อการใช้งานของผู้ใช้ มีรูปแบบการทำรายงานที่น่าใช้ สำหรับนำไปใช้งานต่าง ๆ ได้อย่างเพียงพอ เช่นรายงานจลลภาพปิดของจดหมาย เป็นต้น

Special Topic Development Database System of Graduated Student

Student

MR. WICHAI	PICHITTANARAT	37054139
MR. SEREE	RAKWONG	37054150
MR. APICHAT	PIRINANT	37054151

Advisor MR. WEERACHAI TANYASIT

Department Mathematics and Computer Sciences

Year 1997

Abstract

Database System of Graduated Student was programmed for collect data by expectation to advantage between graduated student and faculty. Following: connection of exchange information, etc. For user can use efficiently program. Such as inserting and query as simple to use for user Moreover, there is pattern to report for enough to make it, such as letter labels, etc.

กิตติกรรมประกาศ

ปัญหาพิเศษฉบับนี้สำเร็จได้ด้วยดี ด้วยความอนุเคราะห์และช่วยเหลือของบุคคลต่าง ๆ เหล่านี้

1. อาจารย์วีระชัย ดันยะสิทธิ์

ที่ได้ให้แนวทางในการวิจัย ตำรา ตลอดจนคำปรึกษาที่สามารถแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้น ในระหว่างทำการวิจัย นอกจากนี้ยังช่วยแนะนำแนวทางในการดำเนินงาน และ ตรวจสอบ แก้ไข

2. อาจารย์ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ที่ได้ช่วยประสาทวิชาความรู้ต่าง ๆ ที่นำมาใช้ในการทำปัญหาพิเศษ

3. เจ้าหน้าที่ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ที่สนับสนุนในการให้ความสะดวกในการใช้และเบิกอุปกรณ์ต่าง ๆ ที่ใช้ในการทำวิจัย

4. บิดา มารดา และผู้มีอุปการคุณ

ที่เป็นผู้ส่งเสริมและสนับสนุนทางด้านการศึกษา

5. เพื่อน ๆ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ที่ช่วยให้คำแนะนำ และ ให้ความช่วยเหลือในหลาย ๆ ด้านในการทำวิจัยนี้

คณะผู้จัดทำ

สารบัญ

หน้า

หน้าอนุมัติ	i
บทคัดย่อภาษาไทย	ii
บทคัดย่อภาษาอังกฤษ	iii
กิตติกรรมประกาศ	iv
บทที่ 1 บทนำ	
1. ความสำคัญและที่มาของปัญหา	1
2. วัตถุประสงค์ของปัญหา	1
3. ขอบเขตของปัญหา	1
4. ประโยชน์ที่คาดว่าจะได้รับ	2
5. ขั้นตอนในการดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักเกณฑ์ที่เกี่ยวข้อง	
1. ความหมาย และ ประเภทของระบบฐานข้อมูล	3
2. สถาปัตยกรรมมาตรฐานของระบบฐานข้อมูล	4
3. ข้อได้เปรียบของระบบฐานข้อมูลที่มีต่อระบบไฟล์	6
4. รูปแบบเชิงสัมพันธ์	7
5. ระบบจัดการฐานข้อมูลเชิงสัมพันธ์(RDBMS)	17
6. การทำตารางให้อยู่ใน รูปแบบนอร์มัล	20
บทที่ 3 การวิจัยและการดำเนินงาน	
1. ผลการศึกษาและวิเคราะห์ระบบงาน	30
2. การออกแบบ Data Flow Diagram (DFD)	33
บทที่ 4 ผลการวิเคราะห์ปัญหาพิเศษ	
ผลการวิจัยและวิจารณ์	37
บทที่ 5 บทสรุปผลการศึกษาวิจัยและข้อเสนอแนะ	
1. สรุปผลการศึกษาวิจัย	38
2. ข้อเสนอแนะ	38
ภาคผนวก ก. คู่มือการใช้โปรแกรม	
ภาคผนวก ข. แบบฟอร์มกรอกประวัติบัณฑิต	
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1	แสดงตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูลเป็นแบบโครงข่าย	3
รูปที่ 2	แสดงตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูลเป็นแบบแผนภูมิต้นไม้	4
รูปที่ 3	แสดงตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูล เป็นแบบเชิงสัมพันธ์	4
รูปที่ 4	สถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture)	5
รูปที่ 5	แสดงรายละเอียดของรูปแบบเชิงสัมพันธ์	7
รูปที่ 6	ตัวอย่างของข้อมูลที่เป็น 1NF	9
รูปที่ 7	แสดงความสัมพันธ์ของการกระทำ	12
รูปที่ 8	แสดงรีเลชัน ที่ยังไม่เป็น 1NF	20
รูปที่ 9	แสดงรีเลชัน First ที่ปรับให้เป็น 1NF แล้ว	21
รูปที่ 10	แสดงฟังก์ชันการขึ้นต่อกันของรีเลชัน First	21
รูปที่ 11	แสดงฟังก์ชันการขึ้นต่อกันใน รีเลชัน SECOND และ SP	22
รูปที่ 12	แสดงรีเลชัน SECOND และ SP.	23
รูปที่ 13	แสดงฟังก์ชันการขึ้นต่อกัน ของรีเลชัน SC และ CS	24
รูปที่ 14	แสดงรีเลชันที่ได้จากการแยกรีเลชัน SECOND	24
รูปที่ 15	แสดงรีเลชัน SUPPLIER3 ที่ยังไม่เป็นรูปแบบ BCNF	25
รูปที่ 16	แสดงฟังก์ชันการขึ้นต่อกันของ รีเลชัน SUPPLIER3	25
รูปที่ 17	แสดงรูปแบบการตรวจสอบ BCNF	26
รูปที่ 18	แสดงการแปลงรีเลชัน SPJC ให้อยู่ในรูปแบบ BCNF	27
รูปที่ 19	แสดงรีเลชัน SPJ ที่เป็น 4NF	27
รูปที่ 20	แสดงรีเลชัน SC ที่เป็น 4NF	28
รูปที่ 21	แสดงรีเลชัน SPP ที่เป็นรูปแบบ 4NF	28
รูปที่ 22	แสดงการแตกรีเลชัน SPP เป็นรีเลชัน SP , PPJ ,PJS และการนำมา join กลับ	29
รูปที่ 23	แสดงตารางที่ทำให้อยู่ในรูปแบบนอร์มัล ระดับที่ 5	29
รูปที่ 24	แสดงแผนผัง (E-R Model) ระบบฐานข้อมูลบัณฑิต	30
รูปที่ 25	แสดง Context Diagram อธิบายความสัมพันธ์ของระบบฐานข้อมูลบัณฑิต	33
รูปที่ 26	แสดง Level 0 ที่แสดงการทำงานของระบบภายในฐานข้อมูล	34
รูปที่ 27	แสดง Level 1.1 ที่แสดงรายละเอียดของระบบการทำงานที่ได้จากการแตก Level0	35
รูปที่ 28	แสดง Level 1.2 ที่แสดงรายละเอียดของรายงานของระบบฐานข้อมูล	36

รูปที่ ก-1 แสดงระบบฐานข้อมูลบัณฑิต	ก-1
รูปที่ ก-2 แสดงเมนูเปิดแฟ้มข้อมูล	ก-2
รูปที่ ก-3 แสดงเมนูแก้ไข	ก-2
รูปที่ ก-4 แสดงเมนูรายงาน	ก-3
รูปที่ ก-5 แสดงเมนูเครื่องมือ	ก-3
รูปที่ ก-6 แสดง Toolbar ระบบฐานข้อมูลบัณฑิต	ก-4
รูปที่ ก-7 แสดงการเปิดแฟ้มข้อมูลหน้าประวัติบัณฑิต	ก-5
รูปที่ ก-8 แสดงการเปิดแฟ้มข้อมูลหน้าที่อยู่	ก-6
รูปที่ ก-9 แสดงการเปิดแฟ้มข้อมูลหน้าประวัติการศึกษา	ก-7
รูปที่ ก-10 แสดงการเปิดแฟ้มข้อมูลหน้าประวัติการทำงาน	ก-8
รูปที่ ก-11 รูปแสดงตัวอย่างของการเพิ่มข้อมูลหน้าของประวัติบัณฑิต	ก-9
รูปที่ ก-12 รูปแสดงตัวอย่างของการแก้ไขข้อมูลหน้าประวัติประวัติบัณฑิต	ก-10
รูปที่ ก-13 แสดงรูปภาพเมื่อต้องการลบข้อมูล	ก-11
รูปที่ ก-14 แสดงรูปภาพเมื่อต้องการลบข้อมูลหลายคน	ก-11
รูปที่ ก-15 แสดงรูปภาพการค้นหาข้อมูล	ก-12
รูปที่ ก-16 แสดงรูปภาพการสำรองข้อมูล	ก-13
รูปที่ ก-17 แสดงรูปภาพการฟื้นฟูสภาพข้อมูล	ก-14
รูปที่ ก-18 แสดงรูปภาพการกู้ข้อมูล	ก-15
รูปที่ ก-19 แสดงรูปภาพทำการ Login เพื่อแก้ไขข้อมูล	ก-16
รูปที่ ก-20 แสดงรูปภาพการทำรายงานแบบฉลากปิดของจดหมาย	ก-16
รูปที่ ก-21 แสดงรูปภาพการทำรายงานแบบบัญชีรายชื่อ	ก-17
รูปที่ ก-22 แสดงรูปภาพการทำรายงานแบบรายบุคคล	ก-18
รูปที่ ก-23 แสดงรูปภาพแฟ้มข้อมูลระดับการศึกษา	ก-19
รูปที่ ก-24 แสดงรูปภาพแฟ้มข้อมูลภาควิชา	ก-20
รูปที่ ก-25 แสดงรูปภาพแฟ้มข้อมูลสาขาวิชา	ก-21

บทที่ 1

บทนำ

การพัฒนาระบบฐานข้อมูลบัณฑิต

Development Database System of Graduated Student

1. ความสำคัญ / ที่มาของปัญหาพิเศษ

เนื่องจากว่าในปัจจุบันนี้เมื่อต้องการที่จะทราบข้อมูลของบัณฑิตในแต่ละครั้ง เราจะต้องทำการค้นหาในแฟ้มเอกสาร ซึ่งเป็นการไม่สะดวกอย่างยิ่ง อีกทั้งอาจเป็นไปได้ว่า อาจจะไม่มียุขมูลที่ทำการค้นหา ซึ่งจะทำให้เสียเวลามาก ในการค้นหาเพื่อที่จะได้ข้อมูลมา

ทางคณะผู้จัดทำจึงได้เสนอการพัฒนาระบบฐานข้อมูลบัณฑิต เพื่อให้สามารถ ช่วยค้นหาข้อมูลเกี่ยวกับบัณฑิตได้รวดเร็วยิ่งขึ้น อีกทั้งยังช่วยส่งเสริมกิจกรรมต่าง ๆ ของคณะ และทางสถาบันในด้านการสร้างความสัมพันธ์กับบัณฑิต โดยการสนับสนุน ทางด้านข้อมูลของบัณฑิต ซึ่งจะทำให้ได้ประโยชน์กับทางสถาบัน คณะ นักศึกษา ที่กำลังศึกษาอยู่ในสถาบันและบัณฑิตที่จบการศึกษาแล้ว

2. วัตถุประสงค์ของปัญหาพิเศษ

- 2.1 เพื่อความสะดวกในการค้นหาข้อมูลบัณฑิตเพื่อติดต่อขอความร่วมมือ
- 2.2 เพื่อประโยชน์ในการขอทราบรายละเอียดด้านการทำงานจากบัณฑิต
- 2.3 เพื่อเป็นข้อมูลเบื้องต้นสำหรับงานของสมาคมศิษย์เก่า KMITL
- 2.4 เพื่อเป็นข้อมูลเบื้องต้นสำหรับงานคืนสู่เหย้าของคณะวิทยาศาสตร์ KMITL

3. ขอบเขตของปัญหาพิเศษ

3.1 เก็บข้อมูลประวัติบัณฑิต ในด้านข้อมูลส่วนตัว และรายละเอียดของงานที่ทำงาน โดยจะจัดเก็บข้อมูลดังต่อไปนี้

- รหัสนักศึกษาเมื่อมีสภาพเป็นนักศึกษาของสถาบัน
- ชื่อภาษาไทย ชื่อภาษาอังกฤษและนามสกุลของบัณฑิตได้
- ที่อยู่ตามภูมิลำเนา ที่อยู่ปัจจุบัน และที่อยู่สถานที่ทำงานของบัณฑิต
- เบอร์โทรศัพท์ เบอร์แฟกซ์และเบอร์วิทยุติดตามตัว (Pager)

3.2 สามารถแสดงรายงานตามความต้องการของผู้ใช้ เช่น แสดงชื่อ นามสกุล รหัสนักศึกษา ที่อยู่ตามภูมิลำเนา ที่อยู่ปัจจุบัน ที่อยู่ทำงาน โดยที่สามารถ ระบุจังหวัด ที่จะค้นหาได้

3.3 สามารถพิมพ์รายงานตามผลที่แสดงการค้นหาได้

4. ประโยชน์ที่คาดว่าจะได้รับ

- 4.1 สามารถสร้างความสัมพันธ์ระหว่างบัณฑิตกับทางคณะวิทยาศาสตร์ นักศึกษาปัจจุบัน และ ทางสถาบัน
- 4.2 ได้รับความร่วมมือกับบัณฑิตทั้งในและนอกประเทศ
- 4.3 เพื่อประโยชน์ของนักศึกษาปัจจุบันและอนาคตในด้านตลาดแรงงาน
- 4.4 เพื่อประโยชน์ของบัณฑิตเองในด้านการติดต่อกับเพื่อนบัณฑิตด้วยตนเอง

5. ขั้นตอนในการดำเนินงาน

- 5.1 ศึกษาลักษณะของข้อมูลและการสร้างฐานข้อมูล
- 5.2 วิเคราะห์ระบบงาน
- 5.3 ออกแบบรายละเอียดของระบบ
- 5.4 พัฒนาโปรแกรม
- 5.5 ทดสอบโปรแกรม
- 5.6 ปรับปรุงระบบ
- 5.7 จัดทำคู่มือในการใช้งาน

ซึ่งมีการแสดงตารางขั้นตอนในการดำเนินงานดังนี้

ตารางเวลาการทำปัญหาพิเศษ

ลำดับงาน	2540						2541		
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1 ศึกษาลักษณะของข้อมูลและการสร้างฐานข้อมูล	_____								
2 วิเคราะห์ระบบงาน		_____							
3 ออกแบบรายละเอียดของระบบ			_____						
4 พัฒนาโปรแกรม				_____					
5 ทดสอบโปรแกรม					_____				
6 ปรับปรุงระบบ					_____				
7 จัดทำคู่มือในการใช้งาน							_____		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักเกณฑ์ที่เกี่ยวข้อง

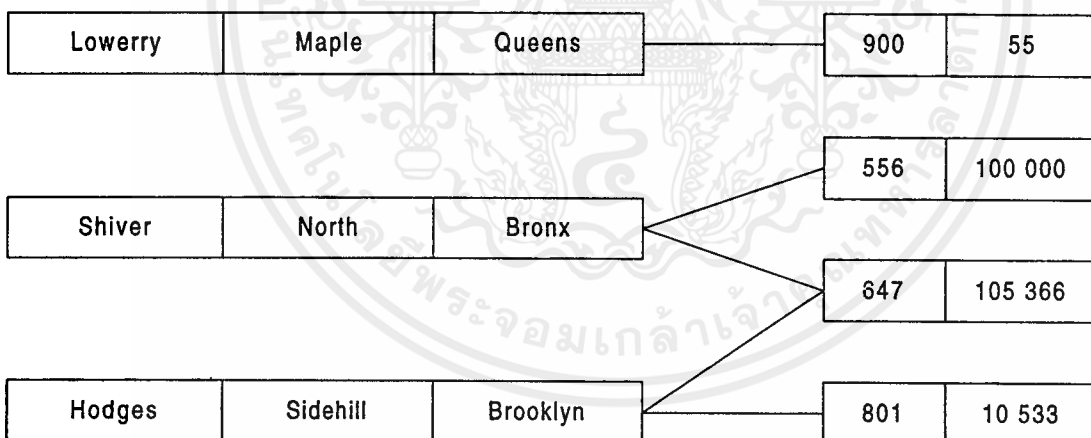
1. ระบบฐานข้อมูล (Database System)

ฐานข้อมูล คือ แหล่งเก็บข้อมูลและความสัมพันธ์ระหว่างข้อมูลนั้น และได้รับการออกแบบ และควบคุมเป็นพิเศษให้มีความซ้ำซ้อนของข้อมูลน้อยที่สุด และ มีความถูกต้องของข้อมูลสูงสุด

ความสัมพันธ์ระหว่างข้อมูลในฐานข้อมูลที่ปัจจุบันนิยมใช้กันอย่างแพร่หลายมีอยู่ 3 รูปแบบคือ

1.1 แบบโครงข่าย (Network Model)

จัดข้อมูลในแบบโครงข่ายนี้จะแสดงด้วยกลุ่มของเรคอร์ด (Record) ที่มีส่วนเชื่อมต่อ (Link) หรือตัวชี้ (Pointer) แสดงความสัมพันธ์ของข้อมูล โดยที่โครงสร้างความสัมพันธ์ของเรคอร์ดที่อยู่ในฐานข้อมูลจะมีสภาพเป็นกลุ่มของความสัมพันธ์ที่ไม่มีข้อกำหนดที่แน่นอน (Collection of arbitrary graph) ซึ่งมีความสัมพันธ์เป็นแบบกลุ่มต่อกลุ่ม (many to many) ดังแสดงความสัมพันธ์ดังรูปที่ 1

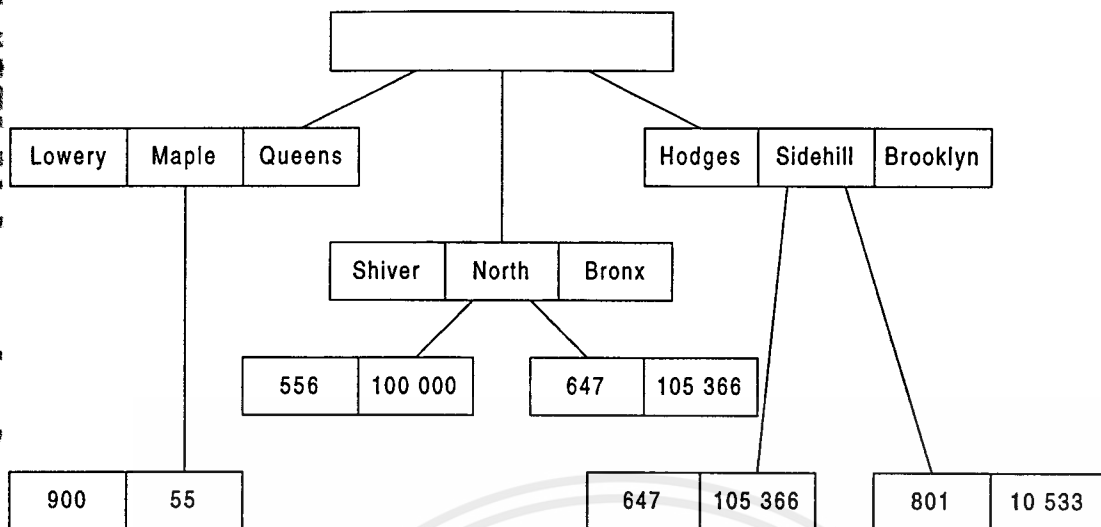


รูปที่ 1 ตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูลเป็นแบบโครงข่าย

1.2 แบบแผนภูมิต้นไม้ (Hierarchical Model)

แบบแผนภูมิต้นไม้จะมีลักษณะใกล้เคียงกับแบบโครงข่าย แต่จะแตกต่างกัน ที่โครงสร้างความสัมพันธ์ของเรคอร์ดที่อยู่ในฐานข้อมูล ซึ่งจะมีความสัมพันธ์เป็นแบบ หนึ่งต่อกลุ่ม (one to many) ซึ่งจะมีลักษณะคล้ายต้นไม้กลับหัว และ การค้นหาข้อมูลที่ต้องการ จะต้องเริ่มจากตัวแม่ (Root) และไล่ความสัมพันธ์ลงมาตามตัวลูก (Child) แสดงความสัมพันธ์ ดังรูปที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 ตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูลเป็นแบบแผนภูมิต้นไม้

1.3 แบบเชิงสัมพันธ์ (Relational Model)

จะแสดงรายละเอียดของข้อมูลและความสัมพันธ์ระหว่างข้อมูล อยู่ในรูปกลุ่มของตาราง ซึ่งในแต่ละตารางจะประกอบด้วยคอลัมน์ (Column) ต่าง ๆ โดยชื่อของคอลัมน์เหล่านั้น จะต้อง มีชื่อไม่ซ้ำกันและสามารถแสดงความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปของตารางได้ โดยไม่มีตัวชี้ หรือลิงค์ลิสต์มาเกี่ยวข้องในการแสดงความสัมพันธ์นี้ แต่สามารถมีตัวชี้ (index) มาเกี่ยวข้องได้ เพื่อประโยชน์ในการเพิ่มความเร็วในการจัดการข้อมูลเท่านั้น ซึ่งจะไม่เกี่ยวข้อง กับความสัมพันธ์ระหว่างข้อมูล สามารถแสดงความสัมพันธ์ดังรูปที่ 3

name	street	city	number
Lowery	Maple	Queens	900
Shiver	North	Bronx	556
Shiver	North	Bronx	647
Hodges	Sidehill	Brooklyn	801
Hodges	Sidehill	Brooklyn	647

number	balance
900	55
556	100 000
647	105 366
801	10 533

รูปที่ 3 ตัวอย่างข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูล เป็นแบบเชิงสัมพันธ์

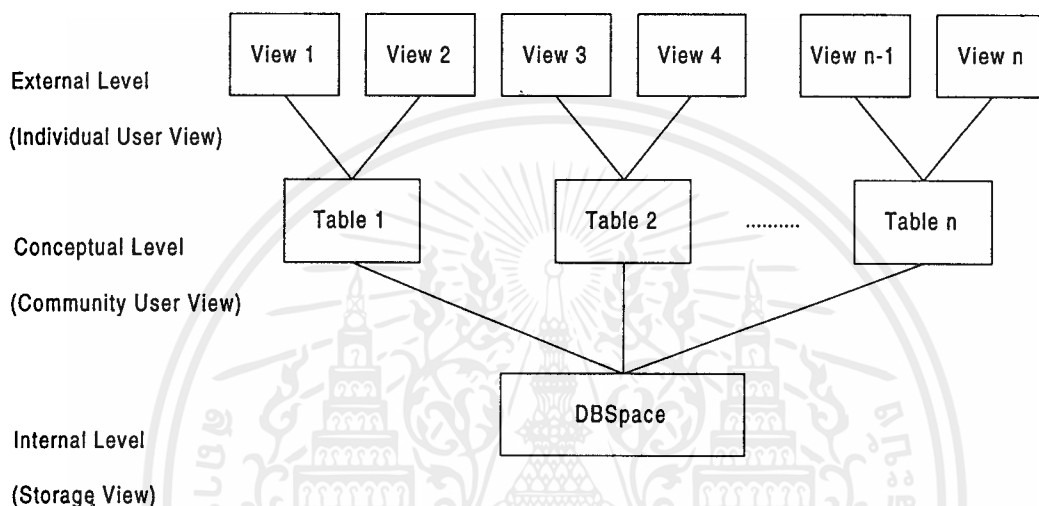
2. สถาปัตยกรรมมาตรฐานของระบบฐานข้อมูล

ได้มีกลุ่มนักวิทยาศาสตร์ของทั้งยุโรปและอเมริกา พยายามกำหนดสถาปัตยกรรม ที่เป็น มาตรฐานของระบบฐานข้อมูล เพื่อให้ฐานข้อมูลก่อให้เกิดประโยชน์แก่ผู้ใช้และหน่วยงานที่ใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูงสุด โดยในที่สุดทั้ง 3 องค์กรหลักคือ

- ISO (International Standard Organization)
- IFIP (International Federation for Information Processing)
- ANSI (American National Standard Institute)

ได้ยอมรับสถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture) ดังแสดงในรูปที่ 4 และมีรายละเอียดดังต่อไปนี้



รูปที่ 4 สถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture)

2.1 นิยามข้อมูลระดับภายนอก (External Schema) จะเป็นการกำหนดโครงสร้างข้อมูล ที่ให้ผู้ใช้เห็น ซึ่งอาจเป็นบางส่วนของนิยามข้อมูลระดับแนวคิด เช่น ในระบบฐานข้อมูลเชิงสัมพันธ์นั้นผู้ใช้บางคนอาจต้องใช้ข้อมูลผ่านวิว (View) ซึ่งจะมีสิทธิใช้ข้อมูลบางแถว (Row) หรือบางคอลัมน์ (Column) ของตารางเท่านั้น ดังนั้นผู้ใช้เหล่านี้จะมองเห็น เฉพาะข้อมูลที่ผู้ดูแลและควบคุมฐานข้อมูล (DBA : Database Administrator) หรือผู้ที่ มีอำนาจกำหนดสิทธิของตารางนั้น ๆ กำหนดขอบเขตการใช้ข้อมูลในตารางต่าง ๆ ให้เท่านั้น

2.2 นิยามข้อมูลระดับแนวคิด (Conceptual Schema) จะเป็นการกำหนดลักษณะ รูปแบบ ข้อมูล ขนาดของข้อมูลและความสัมพันธ์ของข้อมูลทั้งหมดในระบบงาน นั่นคือ ไม่ว่าฐานข้อมูล จะมีความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปแบบใด ๆ ก็ตาม จะต้องกำหนด การแทนรูปแบบของข้อมูลในนิยามข้อมูลระดับแนวคิดนี้ เช่น ถ้าความสัมพันธ์ระหว่างข้อมูล อยู่ในรูปแบบเชิงสัมพันธ์ (Relation Model) ในระดับนี้จะแสดงชื่อตาราง ชื่อคอลัมน์ ชนิดข้อมูลในแต่ละคอลัมน์ ตลอดจนชื่อของคีย์หลัก (Primary Key) และชื่อของคีย์นอก (Foreign Key)

2.3 นิยามข้อมูลระดับภายใน (Internal Schema) จะเป็นการกำหนดลักษณะ โครงสร้าง ข้อมูลที่ถูกจัดเก็บจริงในอุปกรณ์เก็บข้อมูล เช่น ถ้าความสัมพันธ์ระหว่างข้อมูล อยู่ในรูปแบบเชิงสัมพันธ์ซึ่งในระดับแนวคิดและระดับภายนอกจะแสดงอยู่ในรูปแบบของตาราง แต่เมื่อข้อมูลของตารางนั้น ๆ ถูกจัดเก็บจริงในฮาร์ดดิสก์ (Hard disk) ข้อมูล อาจจะถูกจัดเก็บด้วยรูปแบบของ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บีทรี (B-tree) ซีไอแซม (C-ISAM) หรือลิงค์ลิสต์ (Link list) ก็ได้ ซึ่งการจัดเก็บข้อมูลในระดับนี้ระบบการจัดการฐานข้อมูล (DBMS : Database Management System) จะจัดการให้โดยที่ผู้ใช้ไม่ต้องจัดการเอง

3. ข้อได้เปรียบของระบบฐานข้อมูลที่มีต่อระบบไฟล์

3.1 สามารถลดความซ้ำซ้อนของข้อมูล

ในระบบไฟล์นั้นแต่ละแบบงานจะมีแฟ้มข้อมูลของตัวเอง ซึ่งก็ให้เกิดความซ้ำซ้อน ในการเก็บข้อมูลได้และเป็นสาเหตุให้เปลืองเนื้อที่ในการจัดเก็บข้อมูล แต่ในฐานข้อมูล จะเก็บรวบรวมข้อมูลของระบบงานต่าง ๆ ไว้ในฐานข้อมูลเดียวกันจึงมีความซ้ำซ้อนน้อยลง เนื่องจากบางกรณีจะยอมให้ซ้ำซ้อนได้ เพื่อประโยชน์ในการเพิ่มประสิทธิภาพ ในการค้นหาข้อมูล (Query) ดังนั้นประโยชน์ของการลดความซ้ำซ้อนนี้ จะช่วยให้ เปลืองเนื้อที่จัดเก็บน้อยลง และ เอื้อประโยชน์ด้านการจัดการเกี่ยวกับข้อมูล ในด้านการเพิ่มข้อมูล (insert) การลบข้อมูล (delete) การเปลี่ยนแปลงแก้ไขข้อมูล (update) ให้มีประสิทธิภาพมากที่สุด

3.2 สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง

ในฐานข้อมูลนั้นเมื่อไม่เกิดความซ้ำซ้อนของข้อมูลในระดับหนึ่งแล้ว ปัญหาความขัดแย้งของข้อมูลจะไม่เกิดขึ้นเนื่องจากจะมีการเปลี่ยนแปลงแก้ไขข้อมูลเพียงที่เดียว แต่ถ้า ยังมีความซ้ำซ้อนของข้อมูลอยู่ เมื่อมีการเปลี่ยนแปลงแก้ไขข้อมูล จะต้องตามแก้ไขข้อมูล ที่ซ้ำซ้อนในที่เก็บข้อมูลนั้น ๆ ทั้งหมด

3.3 สามารถใช้ข้อมูลร่วมกันได้

เนื่องจากระบบการจัดการฐานข้อมูล(DBMS)นั้น จะมีคุณสมบัติการจัดการ ให้ผู้ใช้สามารถทำงานพร้อม ๆ กันได้หลายคน ดังนั้น โปรแกรมที่พัฒนาขึ้นมา ภายใต้การดูแลของระบบการจัดการฐานข้อมูลเชิงสัมพันธ์จะสามารถใช้ข้อมูลร่วมกันในฐานข้อมูลเดียวกันได้

3.4 สามารถควบคุมมาตรฐานได้

ในระบบฐานข้อมูล ผู้ดูแลและควบคุมฐานข้อมูล(DBA : Database Administrator) จะสามารถกำหนดมาตรฐานของขนาดข้อมูลในตัวแปรต่าง ๆ ชื่อของตัวแปร และ สิทธิการใช้ข้อมูลต่าง ๆ ได้ เพื่อประโยชน์ในการพัฒนาระบบงาน การย้ายข้อมูล ระหว่างระบบงาน และ การกำหนดรูปแบบข้อมูลให้สามารถใช้งานร่วมกันได้

3.5 มีระบบรักษาความปลอดภัยให้ใช้งาน

ในระบบฐานข้อมูล ผู้ดูแลและควบคุม ฐานข้อมูล(DBA) จะสามารถกำหนดสิทธิ และ อำนาจการใช้งานข้อมูลในฐานข้อมูลให้กับผู้ใช้ต่าง ๆ ได้ซึ่งระดับสิทธิและ อำนาจการใช้งาน ข้อมูล อาจจะไม่เหมือนกันขึ้นอยู่กับความเหมาะสม สำหรับฐานข้อมูลเชิงสัมพันธ์นั้น จะมีวิว(View) ซึ่งเป็นตารางสมมติ(Virtual Table) ช่วยกำหนดสิทธิการมองเห็น และ ใช้งานข้อมูลในตารางที่เก็บข้อมูลจริง (Base Table)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 สามารถธำรงรักษาความถูกต้องของข้อมูลได้

ในระบบการจัดการฐานข้อมูล (DBMS) สามารถรองรับการใช้งาน ของผู้ใช้หลายคน พร้อมกันได้ ดังนั้นความถูกต้องของข้อมูลจึงมีความสำคัญมากและต้องควบคุมดูแลให้ดี เนื่องจากผู้ใช้ต่าง ๆ สามารถใช้งานและเปลี่ยนแปลงแก้ไขข้อมูลในฐานข้อมูลได้ ซึ่งถ้ามีความผิดพลาดเกิดขึ้นจะส่งผลกระทบต่อแผนงานโดยรวม แต่ปัญหานี้ ในระบบไฟล์จะไม่มี เนื่องจากผู้ใช้แต่ละคนจะมีไฟล์ส่วนตัว และ ปัญหาข้อนี้สามารถป้องกันได้ ครอบคลุมเพียงใดนั้น จะขึ้นอยู่กับความสามารถ ของแต่ละระบบการจัดการฐานข้อมูล (DBMS)

3.7 สามารถปรับความต้องการของผู้ใช้ให้เป็นในแนวทางเดียวกัน

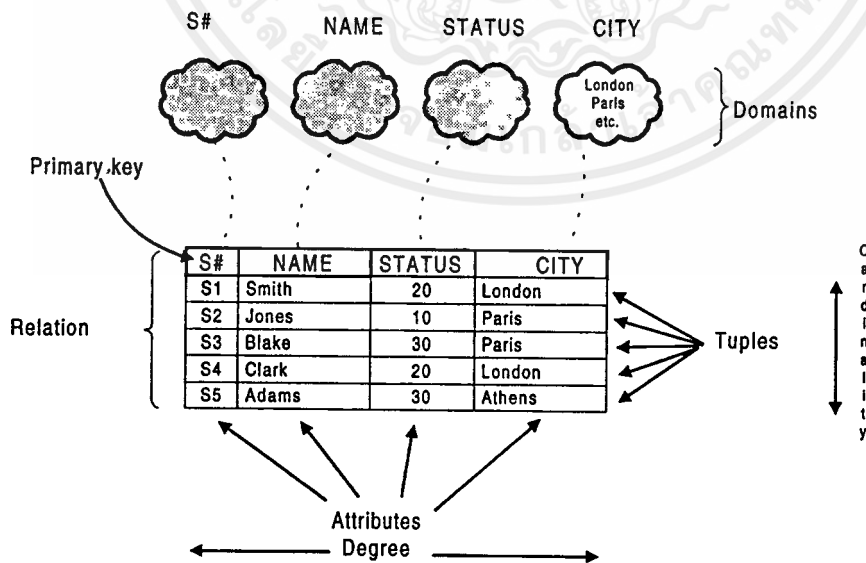
เนื่องจากผู้ดูแลและควบคุมฐานข้อมูล (DBA) เมื่อได้รับรู้ความต้องการ และ ลำดับความสำคัญ ของผู้ใช้ในระบบทั้งหมดก็จะสามารถปรับแต่งความต้องการ ของผู้ใช้เหล่านี้ ให้ไปในแนวทางเดียวกันเพื่อก่อให้เกิดประโยชน์ของส่วนรวมให้สูงสุด เช่น จัดเตรียมอุปกรณ์ ในการเก็บข้อมูลที่มีความเร็วในการเรียกใช้และเก็บข้อมูลสูงให้กับระบบงานที่มีความสำคัญต่อบริษัทสูง

4. รูปแบบเชิงสัมพันธ์ (Relation Model)

ฐานข้อมูลที่มีความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปแบบเชิงสัมพันธ์นั้น จะต้องมีคุณสมบัติต่าง ๆ ดังต่อไปนี้

4.1 โครงสร้างข้อมูล

โครงสร้างข้อมูลจะต้องเป็นเชิงสัมพันธ์เท่านั้น ซึ่งฐานข้อมูลที่มีรูปแบบเชิงสัมพันธ์ นั้น ในทางตรรกะจะใช้ตาราง (table) หรือเชิงสัมพันธ์ (Relation) แสดงความสัมพันธ์ของ โครงสร้างข้อมูลในฐานข้อมูลดังได้แสดงดังรูปที่ 5



รูปที่ 5 แสดงรายละเอียดของรูปแบบเชิงสัมพันธ์

และสามารถเปรียบเทียบคำศัพท์ที่ใช้ในเชิงวิชาการกับศัพท์ที่ใช้ทั่ว ๆ ไป ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ศัพท์เชิงวิชาการ

ศัพท์เชิงธุรกิจ

Relation	table ที่มี row ไม่ซ้ำกัน
Tuple	row หรือ record
Cardinality	จำนวนของ row ในแต่ละ table
Attribute	column หรือ field
Degree	จำนวน column ในแต่ละ table
primary key	column ที่อาจจะเป็น 1 column หรือ มากกว่า 1 column ที่เมื่อระบุค่าของข้อมูลใน column ที่เป็น primary key 1 ค่า จะได้ข้อมูลใน table นั้นออกมาเพียง 1 row
domain	ค่าของข้อมูลที่เป็นไปได้ทั้งหมดในแต่ละ column

4.1.1 เชิงสัมพันธ์

เชิงสัมพันธ์ (Relation) จะเป็นที่เกิดเขตของค่าโดเมน D_1, D_2, \dots, D_n ซึ่งจะขึ้นอยู่กับจำนวนแอททริบิวต์ (attribute) ของแต่ละเชิงสัมพันธ์ โดยที่เชิงสัมพันธ์ จะประกอบไปด้วย 2 ส่วนคือ ส่วนหัว (heading) และส่วนตัว (body)

4.1.1.1 ส่วนหัว (heading) จะประกอบด้วยเขตจำกัดของแอททริบิวต์ เขียนความสัมพันธ์ได้คือ $\{(A_1:D_1), (A_2:D_2), \dots, (A_n:D_n)\}$ ดังนั้น ค่าของแอททริบิวต์ A_j ใด ๆ จะมีขอบเขตของข้อมูลสัมพันธ์กับโดเมน D_j ใด ๆ โดยที่ $j = 1, 2, 3, \dots, n$ เมื่อ n คือ จำนวนแอททริบิวต์ของแต่ละเชิงสัมพันธ์ หรือดีกรีของเชิงสัมพันธ์

4.1.1.2 ส่วนตัว (body) จะประกอบด้วยจำนวนของทัพเพิล (tuple) ซึ่งจะมี จำนวนทัพเพิลเปลี่ยนแปลงไปตามระยะเวลาใด ๆ โดยที่ทัพเพิลจะประกอบด้วยเขต ของแอททริบิวต์ ของแต่ละเชิงสัมพันธ์และค่าของแอททริบิวต์ (value) สามารถเขียนความสัมพันธ์ดังนี้ $\{(A_1:v_{11}), (A_2:v_{12}), \dots, (A_n:v_{1n})\}$ โดยที่ $i = 1, 2, 3, \dots, m$ เมื่อ m คือจำนวนทัพเพิลในแต่ละเชิงสัมพันธ์ หรือคาร์ดินอลิตี้ (cardinality) และ n คือจำนวนแอททริบิวต์ของแต่ละเชิงสัมพันธ์ หรือดีกรีของรีเลชัน

ดังนั้นสำหรับทัพเพิลใด ๆ จะมี $(A_j:v_{ij})$ 1 ค่าสำหรับแอททริบิวต์ A_j ที่อยู่ในส่วนหัว และ $(A_j:v_{ij})$ ใด ๆ ในส่วนของส่วนตัว (body) นั้น ค่า v_{ij} จะอยู่ในขอบเขตของโดเมน D_j

4.1.2 คุณสมบัติของเชิงสัมพันธ์

คุณสมบัติของเชิงสัมพันธ์มีอยู่ 4 ข้อดังต่อไปนี้

4.1.2.1 จะไม่มีทัพเพิลที่ซ้ำกันอยู่ในเชิงสัมพันธ์เดียวกันคุณสมบัติข้อนี้เป็นจริง เนื่องจากส่วนตัวของเชิงสัมพันธ์แสดงความสัมพันธ์อยู่ในรูปของเซต ซึ่งเป็นทฤษฎี ทางคณิตศาสตร์ และโดยนิยามของทฤษฎีเซตนั้น จะไม่รวมข้อมูลที่มีการซ้ำซ้อนกัน ดังนั้นคำว่าเชิงสัมพันธ์ หรือตารางสำหรับฐานข้อมูลเชิงสัมพันธ์จะมีความหมายรวมถึง ค่าของข้อมูลในแถวของตารางจะต้องไม่ซ้ำกันใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ซ้ำซ้อนกัน นั้นแสดงถึงต้องมีคีย์หลัก (Primary key) เพื่อควบคุมไม่ให้ข้อมูลทั้งหมดในแต่ละทัวเพิล ซ้ำกัน

4.1.2.2 ลำดับของทัวเพิลจากบนลงล่าง โดยอยู่ในเชิงสัมพันธ์เดียวกันนั้น จะไม่มีความสำคัญ

คุณสมบัติข้อนี้เป็นจริงเนื่องจากส่วนตัวของ เชิงสัมพันธ์ แสดงความสัมพันธ์ อยู่ในรูปของเซตซึ่งเป็นทฤษฎีทางคณิตศาสตร์และสมาชิกในเซตนั้นจะไม่มีลำดับ ดังนั้น คำว่าเชิงสัมพันธ์หรือตารางสำหรับฐานข้อมูลเชิงสัมพันธ์จะมีความหมายรวมถึง คุณสมบัติการมีลำดับของข้อมูลในแถวของตารางจากบนลงล่างจะไม่มีผลต่อความสัมพันธ์ระหว่างข้อมูลในตาราง

4.1.2.3 ลำดับของแอททริบิวต์จากซ้ายไปขวา โดยอยู่ในเชิงสัมพันธ์เดียวกันนั้น จะไม่มีความสำคัญ

คุณสมบัติข้อนี้เป็นจริงเนื่องจากส่วนหัว ของเชิงสัมพันธ์ แสดงความสัมพันธ์ อยู่ในรูปของเซตซึ่งเป็นทฤษฎีทางคณิตศาสตร์และสมาชิกในเซตนั้นจะไม่มีลำดับ ดังนั้น เมื่อระบุชื่อของแอททริบิวต์ใด ๆ ที่มากกว่า 1 ตัวเหมือนกันแต่ลำดับของแอททริบิวต์ต่างกัน ก็จะได้ข้อมูลตัวเดียวกัน ดังนั้นคำว่าเชิงสัมพันธ์หรือตารางสำหรับฐานข้อมูลเชิงสัมพันธ์ จะมีความหมายรวมถึง คุณสมบัติลำดับของคอลัมน์ในตารางจากซ้ายไปขวา จะไม่มีผลต่อ ความสัมพันธ์ระหว่างข้อมูลในตาราง

4.1.2.4 ค่าของแอททริบิวต์ทุกค่าจะเป็นค่าเดียว

คุณสมบัติข้อนี้ก็คือไม่ให้มีข้อมูลที่มีลักษณะเป็นกลุ่มซ้ำ เก็บอยู่ในเชิง สัมพันธ์ ดังนั้น จะต้องทำการนอร์มอลไลเซชัน (Normalization) ระดับที่ 1 (1NF) ก่อน เพื่อให้ค่าของแอททริบิวต์ทุกตัวเป็นค่าเดียว ดังรูปที่ 6

BEFORE	S#	PQ		AFTER	S#	P#	QTY
		P#	QTY				
S1		P1	300	S1	P1	300	
		P2	200	S1	P2	200	
		P3	400	S1	P3	400	
		P4	200	S1	P4	200	
		P5	100	S1	P5	100	
		P6	100	S1	P6	100	
S2		P1	300	S2	P1	300	
		P2	400	S2	P2	400	
S3		P2	200	S3	P2	200	
S4		P2	200	S4	P2	200	
		P4	300	S4	P4	300	
		P5	400	S4	P5	400	

ดังนั้นคำว่าตารางที่ใช้กันทั่วไปในเชิงธุรกิจสำหรับงานฐานข้อมูลแบบเชิงสัมพันธ์จะมีความหมายรวมถึง ตารางที่จะเก็บข้อมูลที่ผ่านการทำนอร์มอลไลเซชันระดับที่ 1 เท่านั้น

4.2 กฎควบคุมความถูกต้องของข้อมูล

จะต้องมีกฎควบคุมความถูกต้องของข้อมูล เป็นการจัดการเกี่ยวกับขอบเขต ค่าของโดเมน และเงื่อนไขการกระทำของคีย์หลัก (Primary key) และคีย์นอก (Foreign key) โดยสามารถให้ความหมายของคำว่าคีย์แข่งขัน (Candidate key) คีย์หลัก (Primary key) คีย์รอง (Alternate key) และคีย์นอก (Foreign key) ได้ดังนี้

จะให้ k แทนแอททริบิวต์ 1 ตัวหรือกลุ่มของแอททริบิวต์ในเชิงสัมพันธ์ R เป็นคีย์แข่งขันของเชิงสัมพันธ์ R ได้ก็ต่อเมื่อ k มีคุณสมบัติของการเป็นอิสระต่อเวลา 2 อย่าง คือ

- คุณสมบัติความเป็นหนึ่ง (Uniqueness) หมายถึง เมื่อเวลาใด ๆ ค่าของข้อมูลใน k จะไม่มีค่าซ้ำในเชิงสัมพันธ์ R

- คุณสมบัติของการน้อยที่สุด (Minimality) หมายถึงถ้า k แทนด้วยกลุ่มของ แอททริบิวต์ และจะไม่มีแอททริบิวต์ใด ๆ ที่เป็นสมาชิกใน k แล้วทำให้คุณสมบัติ ความเป็นหนึ่งของ k สูญเสียไป

เมื่อได้ค่าจำกัดความของคีย์ แข่งขันแล้วจะได้ว่า คีย์หลักคือคีย์แข่งขันตัวหนึ่ง ที่ถูกเลือกขึ้นมาเพื่อความเหมาะสมในการใช้งานมากที่สุด ส่วนคีย์แข่งขันที่เหลือจะเรียกว่า คีย์รอง ส่วนคีย์นอกคือแอททริบิวต์หรือกลุ่มของแอททริบิวต์ที่อยู่ในเชิงสัมพันธ์ R_2 ที่เขตข้อมูลของแอททริบิวต์หรือกลุ่มของแอททริบิวต์นั้นจะเป็นสับเซตของเขตข้อมูลของคีย์หลักในเชิงสัมพันธ์ R_1 นั่นก็คือค่าข้อมูลของคีย์นอกจะแทนการอ้างอิงหรือเชื่อมโยง ไปยังทัพเพิล ที่มีค่าข้อมูลของคีย์หลักตรงกับค่าข้อมูลของคีย์นอก ดังนั้นสามารถแสดงการเชื่อมโยง ดังความสัมพันธ์

$$R_2 \rightarrow R_1$$

ดังนั้นถ้ามีเชิงสัมพันธ์ n เชิงสัมพันธ์แล้วเชิงสัมพันธ์ที่ n จะเชื่อมโยงกับเชิงสัมพันธ์ที่ $n-1$ สามารถแสดงการเชื่อมโยงดังความสัมพันธ์

$$R_n \rightarrow R_{(n-1)} \rightarrow R_{(n-2)} \rightarrow \dots \rightarrow R_2 \rightarrow R_1$$

จากความหมายและความสัมพันธ์ต่าง ๆ ในเชิงสัมพันธ์เหล่านี้ จึงต้องมีกฎควบคุม ความถูกต้องของข้อมูลในฐานข้อมูลเชิงสัมพันธ์ 2 ข้อคือ

4.2.1 กฎควบคุมความถูกต้องของเอนติตี

นิยาม จะไม่มีแอททริบิวต์ที่เป็นส่วนประกอบของคีย์หลักตัวใดในเชิงสัมพันธ์ ได้รับอนุญาตให้มีค่าเป็นค่าว่าง โดยที่ค่าว่างค่านี้จะเป็นค่าที่ไม่สามารถทราบค่าได้ ซึ่งจะไม่ใช้ค่าศูนย์หรือช่องว่าง

จะเห็นได้ว่ากฎข้อนี้เป็นการควบคุมความ ถูกต้องของคีย์หลัก

4.2.2 กฎควบคุมความถูกต้องของการอ้างอิง

นิยาม ในฐานข้อมูลจะต้องไม่มีการเก็บค่าของคีย์รองที่ไม่มีความสัมพันธ์กับค่าของ คีย์หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่ากฎข้อนี้จะใช้ควบคุมความถูกต้องของคีย์นอกที่มีความสัมพันธ์กับคีย์หลัก โดยที่จะไม่ครอบคลุมความสัมพันธ์กับคีย์รอง

จากกฎควบคุมความถูกต้องของการอ้างอิงนี้ จะนำไปสู่ปัญหาของการทำงาน ที่มีผลกระทบต่อคีย์นอก ส่วนการแก้ไขข้อมูลหรือการลบข้อมูลที่เป็นคีย์หลักในเชิงสัมพันธ์ต่าง ๆ และปัญหาของการอนุญาตให้ค่าข้อมูลของคีย์นอกมีค่าเป็นค่าว่างได้

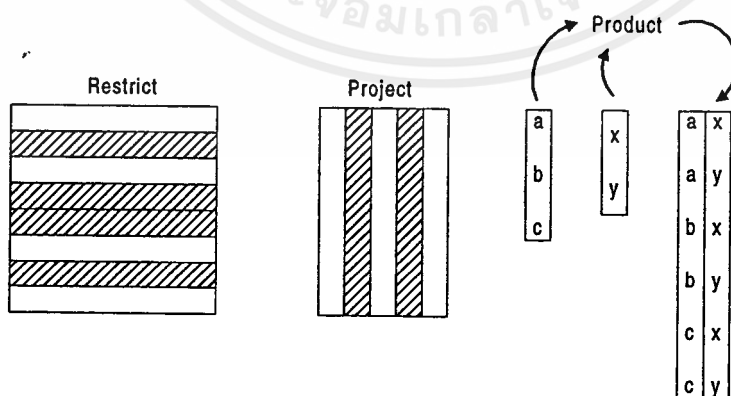
4.3 ภาษาจัดการข้อมูล

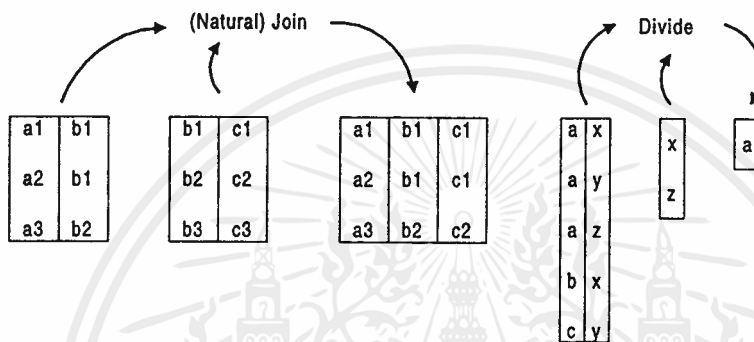
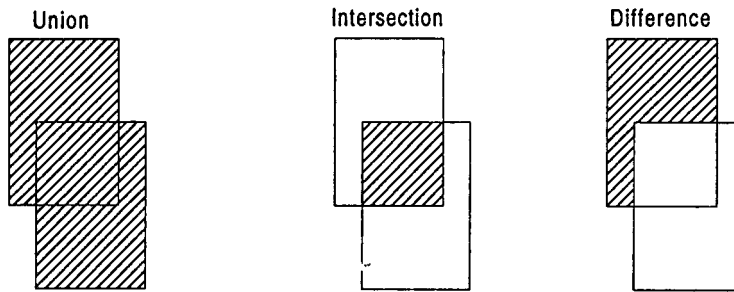
ภาษาจัดการข้อมูลจะต้องมีคุณสมบัติของการเป็น เชิงสัมพันธ์อย่างสมบูรณ์ คือ อย่างน้อยต้องมีความสามารถเทียบเท่าคณิตศาสตร์ของเชิงสัมพันธ์ หรือพีชคณิต ของเชิงสัมพันธ์

พีชคณิตของเชิงสัมพันธ์จะมีรูปแบบคล้ายกับการเขียนโปรแกรม หรือโปรแกรม คือ จะต้องสั่งให้ทำงานตามลำดับขั้นตอนเพื่อให้ได้ผลลัพธ์ที่ต้องการออกมา หรืออนโปรแกรม คือ จะบอกเฉพาะผลลัพธ์ที่ต้องการเท่านั้น ในแบบการจัดการฐาน ข้อมูลเชิงสัมพันธ์ ที่ทำขึ้นในเชิงพาณิชย์นั้นจะมีภาษาสอบถามข้อมูลทั้งแบบโปรแกรม ซึ่งมีพื้นฐานมาจาก พีชคณิตของเชิงสัมพันธ์ และแบบอนโปรแกรม ซึ่งมีพื้นฐานมาจากคณิตศาสตร์ของ ทัพเพิลเชิงสัมพันธ์ หรือคณิตศาสตร์ของโดเมนเชิงสัมพันธ์ โดยมีรายละเอียดดังนี้

4.3.1 พีชคณิตของเชิงสัมพันธ์

พีชคณิตของเชิงสัมพันธ์เป็นภาษาสอบถามข้อมูลแบบโปรแกรม ซึ่งเกิดจากกลุ่ม ของการกระทำ ที่กระทำกับเชิงสัมพันธ์ต่าง ๆ โดยต้องบอกลำดับการกระทำ เพื่อให้ได้ ผลลัพธ์ที่ต้องการออกมา การกระทำเหล่านั้น ก็คือ Select (Restrict), Project, Cartesian product, Rename, Union, Set difference, Set intersection, Natural join, Division (Divide) และ Assignment สามารถแสดงรูปภาพประกอบความสัมพันธ์ได้ดังรูปที่ 7





รูปที่ 7 แสดงความสัมพันธ์ของการกระทำ

ให้เชิงสัมพันธ์ borrow มีโครงสร้างข้อมูลคือ

```
borrow(branch_name,loan_number,customer_name,amount)
```

ให้เชิงสัมพันธ์ client มีโครงสร้างข้อมูลคือ

```
client(customer_name,banker_name)
```

ให้เชิงสัมพันธ์ customer มีโครงสร้างข้อมูลคือ

```
customer(customer_name,street,customer_city)
```

ให้เชิงสัมพันธ์ deposit และเชิงสัมพันธ์ borrow มีโครงสร้างข้อมูลคือ

```
deposit(branch_name,account_number,customer_name,balance)
```

4.3.1.1 The Select Operation

การกระทำตัวนี้จะเลือกทัพเฟิลในเชิงสัมพันธ์ออกมาตามเงื่อนไขที่ระบุโดยที่ใช้ Sigma

เอกสา (σ) แทน select operation และเงื่อนไขที่ระบุจะเป็นตัวห้อยของ σ และชื่อของเชิง

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัมพันธ์จะอยู่ในวงเล็บหลัง σ โดยที่เงื่อนไขการเปรียบเทียบ สามารถใช้การเท่ากับ (=) ไม่เท่ากับ (\neq) น้อยกว่า (<) น้อยกว่าหรือเท่ากับ (\leq) มากกว่า(>) มากกว่าหรือเท่ากับ (\geq) และสามารถรวมเงื่อนไขต่าง ๆ เข้าด้วยกันโดยใช้เงื่อนไข "และ (\wedge)" และเงื่อนไข "หรือ (\vee)"

เมื่อต้องการทัพเฟิลของเชิงสัมพันธ์ borrow ในฐานข้อมูลโดยมีเงื่อนไขว่า branch_name = Perryridge และ amount > \$1200 สามารถแสดงโดย select operation ได้คือ

$$\sigma_{(\text{branch_name} = \text{"Perryridge"} \wedge \text{amount} > 1200)}(\text{borrow})$$

4.3.1.2 The Project Operation

การกระทำตัวนี้จะเลือกแอททริบิวต์ที่ต้องการเลือกในเชิงสัมพันธ์ออกมาตามเงื่อนไขที่ระบุโดยใช้ $\pi(\Pi)$ แทน project operation ตัวอย่างเช่น ต้องการเลือกข้อมูลของแอททริบิวต์ที่ชื่อ customer_name ออกมาโดยมีเงื่อนไขว่า branch_name = Perryridge และ amount > \$1200 สามารถแสดงโดยใช้ project operation คือ

$$\Pi_{\text{customer_name}}(\sigma_{(\text{branch_name} = \text{"Perryridge"} \wedge \text{amount} > 1200)}(\text{borrow}))$$

4.3.1.3 The Cartesian Product Operation

การกระทำตัวนี้จะรวมข้อมูล ทั้งหมดที่เป็นไปได้จากเชิงสัมพันธ์ต่าง ๆ ที่ต้องการโดยใช้ cross (\times) แทน Cartesian product เนื่องจากสามารถกำหนดเชิงสัมพันธ์ เป็นสับเซตของการทำ Cartesian product ของเซตของโดเมนต่าง ๆ ของแต่ละแอททริบิวต์ ในเชิงสัมพันธ์ได้ ดังนั้นถ้าเชิงสัมพันธ์ R เป็น Cartesian product ของเชิงสัมพันธ์ client และเชิงสัมพันธ์ customer แล้วจะได้ว่า

$$r = \text{client} \times \text{customer}$$

4.3.1.4 The Rename Operation

การกระทำนี้จะใช้เปลี่ยนชื่อของเชิงสัมพันธ์เป็นชื่อใหม่เพื่อประโยชน์ในการสอบถามข้อมูลโดยใช้ $\rho_x(r)$ แทนการ rename เชิงสัมพันธ์ r เป็นชื่อเชิงสัมพันธ์ x ตัวอย่างเช่น เมื่อต้องการหาชื่อของลูกค้าทุกคนที่อาศัยอยู่บนถนนในเมืองเดียวกันกับลูกค้าที่ชื่อ Smith จะแสดงการกระทำต่าง ๆ ได้ดังนี้

$$\Pi_{\text{customer.customer_name}}(\sigma_{A.\text{street}=\text{customer.street} \wedge A.\text{customer_city}=\text{customer.customer_city}}(\text{customer} \times (\Pi_{\text{street, customer_city}}(\sigma_{\text{customer_name}=\text{"Smith"}}(\rho_A(\text{customer}))))))$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.5 The Union Operation

การกระทำตัวนี้จะทำการรวมข้อมูลของผลลัพธ์ที่ได้จากการกระทำในแต่ละเชิงสัมพันธ์เข้าด้วยกันโดยมีข้อจำกัดว่า แต่ละเชิงสัมพันธ์นั้นจะต้องมีจำนวนแอททริบิวต์เท่ากัน และต้องมีโดเมนเดียวกันในแอททริบิวต์ที่มีลำดับเดียวกันของแต่ละรีเลชัน ซึ่งการรวมข้อมูลนี้ ถ้ามีการซ้ำกันจะเลือกเอามาเพียงตัวเดียวเท่านั้นจะใช้ "∪" แทนการ union ตัวอย่างเช่น ต้องการข้อมูลลูกค้าทั้งหมดที่อยู่ในสาขา Perryridge ของเชิงสัมพันธ์ deposit และ borrow จะได้คำตอบโดยแสดงความสัมพันธ์ต่าง ๆ ดังนี้

$$\Pi_{\text{customer_name}} (\sigma_{\text{branch_name} = \text{"Perryridge"}} (\text{borrow})) \cup \Pi_{\text{customer_name}} (\sigma_{\text{branch_name} = \text{"Perryridge"}} (\text{deposit}))$$

4.3.1.6 The Set Difference Operation

การกระทำตัวนี้จะหาข้อมูลที่อยู่ในเชิงสัมพันธ์หนึ่งแต่ไม่อยู่ในอีกเชิงสัมพันธ์หนึ่ง จะให้ "-" แทน set difference นั่นคือ เมื่อมี $r - s$ จะได้ผลลัพธ์ออกมาเป็นทัพเฟิล ที่อยู่ในเชิงสัมพันธ์ r แต่ไม่อยู่ในเชิงสัมพันธ์ s ตัวอย่างเช่น ต้องการทราบจำนวนเงินฝาก ในบัญชีที่มากที่สุดของลูกค้าในเชิงสัมพันธ์ deposit จะได้ความสัมพันธ์ของการกระทำต่าง ๆ ดังนี้

$$\Pi_{\text{balance}(\text{deposit})} - \Pi_{\text{deposit.balance}} (\sigma_{\text{deposit.balance} < \text{d.balance}} (\text{deposit} \times \rho_d (\text{deposit})))$$

4.3.1.7 The Set Intersection Operation

การกระทำตัวนี้จะเอาข้อมูลเฉพาะข้อมูลแต่ละเชิงสัมพันธ์ที่เหมือนกันออกมาจะใช้ "∩" แทน set intersection และสามารถแปลงอยู่ในความสัมพันธ์ของ set difference ได้ดังนี้

$$r \cap s = r - (r - s)$$

และเมื่อต้องการข้อมูลของลูกค้าที่มีบัญชีอยู่ในสาขา Perryridge ทั้งใน borrow และ deposit จะสามารถแสดงความสัมพันธ์ของการกระทำต่าง ๆ ดังนี้

$$\Pi_{\text{customer_name}} (\sigma_{\text{branch_name} = \text{"Perryridge"}} (\text{borrow})) \cap \Pi_{\text{customer_name}} (\sigma_{\text{branch_name} = \text{"Perryridge"}} (\text{deposit}))$$

4.3.1.8 The Natural Join Operation

การกระทำตัวนี้จะรวมการกระทำ 2 อย่างเข้าด้วยกันคือ จะทำ Cartesian product ระหว่างเชิงสัมพันธ์เป็นอันดับแรก หลังจากนั้นจะทำ select โดยจะเอาเฉพาะข้อมูล ที่มีค่าของแอททริบิวต์ ในแต่ละเชิงสัมพันธ์ซึ่งอยู่ในโดเมนเดียวกันและมีค่าเท่ากันออกมา เมื่อได้ผลลัพธ์แล้วจะเอาคอลัมน์ที่ซ้ำกันออกไป จะใช้สัญลักษณ์ " \bowtie " แทน natural join ตัวอย่างเช่น ต้องการข้อมูลของลูกค้าทุกคนที่กู้เงินธนาคารในสาขาที่อยู่ในเมือง ที่ลูกค้าอาศัยอยู่ จะได้ความสัมพันธ์ของการกระทำต่าง ๆ ที่ไม่ใช่ natural join คือ

$$\Pi_{\text{borrow.customer_name, customer_city}} (\sigma_{\text{borrow.customer_name} = \text{customer.customer_name}} (\text{borrow} \times \text{customer}))$$

และเมื่อใช้ natural join จะได้ความสัมพันธ์ของการกระทำต่าง ๆ คือ

$$\Pi_{\text{customer_name, customer_city}} (\text{borrow} \bowtie \text{customer})$$

เพื่อให้เห็นชัดเจนยิ่งขึ้นจะให้ R และ S เป็นเซตของแอททริบิวในเชิงสัมพันธ์ r และ s ตามลำดับ ดังนั้นแอททริบิวที่มีอยู่ทั้งใน R และ S ก็คือ $R \cap S$ และแอททริบิวทั้งหมด ที่อยู่ใน R หรือ S หรืออยู่ในทั้ง R และ S ก็คือ $R \cup S$ ดังนั้น natural join ของเชิงสัมพันธ์ r และ s ก็คือ

$$r \bowtie s = \Pi_{R \cup S} (\sigma_{(r.A1=s.A1) \wedge (r.A2=s.A2) \wedge \dots \wedge (r.An=s.An)} (r \times s))$$

เมื่อ $R \cap S = \{A1, A2, \dots, An\}$

4.3.1.9 The Division Operation

การกระทำตัวนี้จะจัดการค้นหาข้อมูลที่มีเงื่อนไข "for all" มาเกี่ยวข้องด้วย โดยจะใช้สัญลักษณ์ " \div " แทน division ตัวอย่างเช่น ต้องการข้อมูลของลูกค้าทุกคน ที่มีบัญชีอยู่ทุกสาขาในเมือง Brooklyn จะได้ความสัมพันธ์ของการกระทำต่าง ๆ คือ

$$\Pi_{\text{customer_name, branch_name}} (\text{deposit}) \div \Pi_{\text{branch_name}} (\sigma_{\text{branch_city} = \text{"Brooklyn"}})$$

ถ้าให้ r(R) และ s(S) เป็นเชิงสัมพันธ์ และให้ $S \subseteq R$ จะได้

$$r \div s = \Pi_{R-S} (r) - \Pi_{R-S} (r \times s) - r$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.10 The Assignment Operation

การกระทำตัวนี้ไม่ได้ช่วยให้ได้ผลลัพธ์ของกลุ่มการกระทำต่าง ๆ ที่มี assignment มาเกี่ยวข้อง แต่จะช่วยให้สามารถแสดงความสัมพันธ์ของการกระทำต่าง ๆ ให้อยู่ในรูปแบบที่ซับซ้อนน้อยลง โดยจะใช้สัญลักษณ์ " \leftarrow " แทน assignment โดยที่ทางขวาของสัญลักษณ์นี้จะเป็นกลุ่มของการกระทำต่าง ๆ และทางซ้ายของสัญลักษณ์ จะเป็นคำตอบของกลุ่มการกระทำต่าง ๆ เหล่านั้นซึ่งเขียนแทนด้วยตัวแปรเชิงสัมพันธ์ ที่กำหนดขึ้นชั่วคราวตัวอย่างเช่น จากการทำงานของ division operation นั้นสามารถ จัดรูปแบบของ $r \div s$ ได้ใหม่เพื่อความสะดวกต่อการแสดงความสัมพันธ์ได้ดังนี้

$$\begin{aligned} \text{temp} &\leftarrow \prod_{R-s}(r) \\ \text{temp} &= \prod_{R-s}((\text{temp} \times s) - r) \end{aligned}$$

4.3.2 คณิตศาสตร์ของเชิงสัมพันธ์

คณิตศาสตร์ของเชิงสัมพันธ์เป็นภาษาสอบถามข้อมูลแบบนอนโพรซีเจอร์ (nonprocedure) ซึ่งจะบอกถึงข้อมูลที่ต้องการเท่านั้น โดยจะไม่บอกถึง ขั้นตอนวิธีการ ที่ทำให้ได้ข้อมูลที่ต้องการออกมาซึ่งคณิตศาสตร์ของเชิงสัมพันธ์แบ่งได้ 2 ประเภทคือ

4.3.2.1 คณิตศาสตร์ของเชิงสัมพันธ์แบบทัพเพิล

คณิตศาสตร์ของเชิงสัมพันธ์แบบทัพเพิลจะอยู่ในรูปแบบ

$$\{t / P(t)\}$$

เมื่อ t แทนเซตของทัพเพิลในเชิงสัมพันธ์ r ที่ทำให้ $P(t)$ เป็นจริงและ $t[A]$ แทนค่าของแอททริบิว A ในทัพเพิล t นั่นคือ ในรูปแบบนี้จะสนใจค่าข้อมูล ในแต่ละทัพเพิลที่สนใจ

ตัวอย่างการดึงข้อมูลโดยใช้คณิตศาสตร์ของเชิงสัมพันธ์แบบทัพเพิล เมื่อต้องการค้นหา branch_name, loan_number, customer_name และ amount ที่มีการกู้ยืมเงินมากกว่า \$1200 ซึ่งจะได้คำตอบอยู่ในรูปแบบดังนี้

$$\{t / t \in \text{borrow} \wedge t[\text{amount}] > 1200\}$$

และถ้าต้องการข้อมูลเฉพาะแอททริบิว customer_name เท่านั้น สามารถปรับรูปแบบ ของคำตอบได้ดังนี้

$$\begin{aligned} \{t / \exists s \in \text{borrow} (t[\text{customer_name}] = s[\text{customer_name}] \wedge \\ s[\text{amount}] > 1,200)\} \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีความหมายว่า เซตของทุกทัพเฟิล t ที่ประกอบด้วยทัพเฟิล s ในเชิงสัมพันธ์ borrow ที่ค่าของ t และ s ในแอททริบิว $customer_name$ เท่ากัน และค่าของ s ในแอททริบิว $amount$ มีค่ามากกว่า \$1200 จะเห็นว่าทัพเฟิล t จะถูกกำหนดเฉพาะแอททริบิว $customer_name$ เท่านั้น

4.3.2.2 คณิตศาสตร์ของเชิงสัมพันธ์แบบโดเมน

ในกรณีนี้จะสนใจค่าข้อมูลในโดเมนของแอททริบิวเท่านั้น โดยจะอยู่ในรูปแบบ

$$\{ \langle X_1, X_2, \dots, X_n \rangle / P(X_1, X_2, \dots, X_n) \}$$

เมื่อ X_1, X_2, \dots, X_n แทนตัวแปรโดเมน

ตัวอย่างการดึงข้อมูลโดยใช้คณิตศาสตร์ของเชิงสัมพันธ์แบบโดเมน เมื่อต้องการค้นหา $branch_name, loan_number, customer_name$ และ $amount$ ที่มีการกู้ยืมเงินมากกว่า \$1200 ซึ่งจะได้คำตอบอยู่ในรูปแบบ ดังนี้

$$\{ \langle b, l, c, a \rangle / \langle b, l, c, a \rangle \in borrow \wedge a > 1200 \}$$

และถ้าต้องการเฉพาะข้อมูลของแอททริบิว $customer$ เท่านั้น จะได้รูปแบบของการแสดง การค้นหาข้อมูลคือ

$$\{ \langle c \rangle / \exists b, l, a (\langle b, l, c, a \rangle \in borrow \wedge a > 1200) \}$$

5. ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System)

ระบบจัดการฐานข้อมูลเชิงสัมพันธ์เป็น ซอฟต์แวร์ (Software) สำหรับจัดการข้อมูล และความสัมพันธ์ของข้อมูลที่เป็นแบบเชิงสัมพันธ์ รวมทั้งมีภาษาสำหรับกำหนด โครงสร้างของข้อมูลในระบบต่าง ๆ ตามสถาปัตยกรรมมาตรฐานของระบบฐานข้อมูล การเข้าถึงข้อมูลและการติดต่อกับผู้ใช้และโปรแกรมของผู้ใช้ที่เขียนด้วยโปรแกรมอื่น ดังมีรายละเอียดของประโยชน์ของการใช้ระบบจัดการข้อมูลเชิงสัมพันธ์ดังนี้

5.1 ภาษาเอสคิวแอล (SQL : Structure Query Language)

ช่วยจัดการการทำงานเกี่ยวกับข้อมูล สามารถแบ่งตามลักษณะการทำงานได้ 2 ส่วน คือ

5.1.1 การจัดการเกี่ยวกับการกำหนดโครงสร้างข้อมูล (Data Definition Language : DDL) มีหน้าที่ในการกำหนดโครงสร้างข้อมูลที่ใช้แต่ละคนมองเห็น โครงสร้างข้อมูล ที่นี้บอกแบบฐานข้อมูลมองเห็น และโครงสร้างข้อมูลที่จัดเก็บในอุปกรณ์เก็บข้อมูล ซึ่งผลของการแปล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'DDL จะเก็บในไฟล์พิเศษที่เรียกว่าพจนานุกรมของข้อมูล (Data Dictionary) ตัวอย่างคำสั่ง เช่น CREATE DATABASE, CREATE TABLE และ CREATE INDEX เป็นต้น

5.1.2 การจัดการเกี่ยวกับข้อมูล (Data Manipulation Language : DML) มีหน้าที่ในด้านจัดการเข้าถึงข้อมูล ได้แก่ การสอบถามหรือการเรียกค้นข้อมูล ที่อยู่ในฐานข้อมูลการเพิ่มเติมข้อมูลใหม่เข้าไปในฐานข้อมูล การลบข้อมูลออกจากฐานข้อมูล การเปลี่ยนแปลงแก้ไขข้อมูลที่อยู่ในฐานข้อมูล ตัวอย่างคำสั่ง เช่น SELECT, INSERT, DELETE และ UPDATE เป็นต้น

5.2 ประสิทธิภาพสูงและสะดวกรวดเร็วในการพัฒนาระบบงาน

เนื่องจากเมื่อต้องการข้อมูลจะอ่านข้อมูลขึ้นมาเป็นเซตของข้อมูล ทำให้สามารถทำงานได้สะดวกรวดเร็วและสามารถพัฒนาระบบงานได้รวดเร็วขึ้น นอกจากนี้โครงสร้างการทำงานเป็นลักษณะที่รองรับให้ผู้ใช้ใช้ได้หลายคน ดังนั้นจึงไม่ต้องเขียนโปรแกรมที่ควบคุม การให้ผู้ใช้สามารถทำงานได้พร้อม ๆ กันหลายคน ควบคุมด้านความปลอดภัยของข้อมูล ควบคุมด้านอำนาจสิทธิของผู้ใช้แต่ละคน นอกจากนี้ยังสามารถกู้ข้อมูลได้โดยอัตโนมัติ เมื่อระบบงานเกิดความเสียหาย

5.3 การจัดการด้านความเป็นอิสระของข้อมูล

เมื่อมีการเปลี่ยนแปลงแก้ไขโครงสร้างข้อมูล ในนิยามข้อมูลระดับแนวคิด หรือในนิยามข้อมูลระดับภายในนั้น จะไม่มีผลกระทบต่อโปรแกรมที่ให้ผู้ใช้งาน ในนิยามข้อมูลระดับภายนอก เนื่องจากระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ จะช่วยจัดการ ให้ผู้ใช้มองเห็นโครงสร้างข้อมูลในนิยามข้อมูลระดับภายนอกเหมือนเดิม และสามารถใช้งาน ได้ตามปกติ ซึ่งการจัดการด้านความเป็นอิสระของข้อมูลแบ่งออกเป็น 2 ลักษณะคือ

5.3.1 ความเป็นอิสระของข้อมูลในระดับตรรกะ เมื่อมีการแก้ไขโครงสร้างข้อมูล ในนิยามข้อมูลระดับแนวคิด จะไม่มีผลกระทบต่อโครงสร้างข้อมูลในนิยามข้อมูลระดับภายนอก ที่ผู้ใช้ใช้งานกันอยู่ เช่น การเปลี่ยนแปลงขนาดคอลัมน์ หรือการเพิ่มคอลัมน์เข้าไป

5.3.2 ความเป็นอิสระของข้อมูลในระดับกายภาพ เมื่อมีการแก้ไขโครงสร้างข้อมูล ในนิยามข้อมูลระดับภายใน จะไม่มีผลกระทบต่อโครงสร้างข้อมูลในนิยามข้อมูลระดับแนวคิด เช่น มีการเปลี่ยนแปลงวิธีการเข้าถึงข้อมูลในนิยามข้อมูลระดับภายในโดยการเปลี่ยนแปลง ลักษณะการจัดเก็บจาก ซี-ไอแซม (C - ISAM) เป็นบีทรี(BTree) หรือแม้กระทั่งเปลี่ยนอุปกรณ์ที่ใช้เก็บข้อมูลอย่างเช่น ฮาร์ดดิสก์ (Harddisk) หรือการอัปเกรด (Upgrade) ระบบปฏิบัติการ

5.4 ความปลอดภัยของข้อมูล

ระบบการจัดการฐานข้อมูลได้มี ระบบการสอบถามรายชื่อพร้อมทั้งรหัสลับ ของผู้เข้ามาใช้งานในระบบงานเพื่อให้เข้าทำงานในส่วนที่ผู้ใช้เกี่ยวข้องเท่านั้น นอกจากนี้ข้อมูลที่เก็บอยู่ในฐาน

ข้อมูลเชิงสัมพันธ์นั้นจะถูกทำการเข้ารหัส (Encryption) เพื่อป้องกันการเข้าถึงข้อมูลภาษาอื่น ๆ โดยไม่ผ่านระบบการจัดการฐานข้อมูลเชิงสัมพันธ์

5.5 การจัดอำนาจในการใช้ข้อมูล

ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ สามารถทำการกำหนดสิทธิอำนาจ การใช้งานข้อมูลให้แก่ผู้ใช้ต่าง ๆ ได้โดยการใช้คำสั่ง GRANT หรือเพิกถอนสิทธิได้ โดยการใช้คำสั่ง REVOKE นอกจากนี้ยังใช้ View (View) เพื่อช่วยกำหนดสิทธิการใช้ข้อมูล ของแต่ละคนในนิยามข้อมูลระดับภายนอกได้อีกด้วย

5.6 ระบบการกู้ข้อมูลโดยอัตโนมัติเมื่อระบบเกิดความเสียหาย

ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ ได้เตรียมการรองรับการทำการประมวลผล กลุ่มงาน โดยกลุ่มงานนี้มีความหมายถึงกลุ่มของงานในทางตรรกะ ซึ่งยอมให้มีการละเมิดกฎ สำหรับควบคุมความถูกต้องของข้อมูลได้ภายในกลุ่มของงานและการทำงานทั้งหมดในกลุ่มงานนี้ ถ้าทำงานเสร็จตามความต้องการจะต้องเสร็จด้วยกันทั้งหมด ถ้าเกิดความบกพร่องขึ้น ภายในกลุ่มงานจะต้องยกเลิกทั้งหมดโดยที่ COMMIT คือคำสั่งยืนยันการทำงานสำเร็จส่วน ROLLBACK คือคำสั่งยกเลิกการทำงานของกลุ่มงานนั้น ๆ ดังนั้น ถ้า commit แล้วระบบเกิดความเสียหายขึ้น ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์จะรับรองว่าข้อมูลที่ commit แล้วจะไม่สูญหายไป ในทางกลับ

กันกรณีของ rollback และการทำงานที่ยังไม่ commit นั้นระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ จะรับรองว่าข้อมูลเดิมก่อนการทำงานของกลุ่มงานยังไม่สูญหายไปไหน

5.7 การดูและผู้ใช้หลายคนให้สามารถทำงานได้พร้อม ๆ กัน

ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ จะมีการรองรับการทำงานพร้อมกัน ของผู้ใช้หลายคนได้โดยอาจมีปัญหาคิดขึ้นได้ 3 ประเภทใหญ่ ๆ คือ

5.7.1 ปัญหาการสูญเสียการเปลี่ยนแปลงข้อมูล เกิดจากการอ่านข้อมูลตัวเดียวกัน ไปใช้งาน โดยที่การเปลี่ยนแปลงข้อมูลตัวนั้นในเวลาที่ไม่พร้อมกัน

5.7.2 ปัญหาที่เกิดจากการยกเลิกการเปลี่ยนแปลงข้อมูล เกิดจากมีการนำเอาข้อมูลไปใช้ในขณะการประมวลผลกลุ่มงานยังไม่เสร็จสิ้นและมีการยกเลิกการประมวลผลกลุ่มงานนั้น

5.7.3 ปัญหาที่เกิดจากการขัดแย้งของการทำงาน เกิดจากมีการนำเอาข้อมูล ไปประมวลผล และมีการยอมรับความเปลี่ยนแปลงนั้นโดยที่การเปลี่ยนแปลงที่เกิดขึ้น มีผลต่ออีกการทำงานหนึ่ง

ซึ่งปัญหาต่าง ๆ เหล่านี้สามารถป้องกันได้โดยวิธีการจองข้อมูล (Lock) ก่อนการใช้งาน ซึ่งการจองข้อมูลนี้มีอยู่ 2 แนวทางคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จองข้อมูลแบบใช้งานร่วมกัน (Share - mode Lock) หมายถึง ถ้ากลุ่มงาน Ti จองแบบใช้งานร่วมกัน (Share - mode Lock แทนด้วย S) บนข้อมูล Q ดังนั้น Ti สามารถอ่านข้อมูล Q ได้โดยไม่สามารถเขียนลง Q ได้ ซึ่งกลุ่มงานอื่นที่เข้ามาใช้งาน Q หลังจากนี้ก็จะสามารถอ่านข้อมูลจาก Q ได้อย่างเดียวเท่านั้น

- จองข้อมูลแบบพิเศษ (Exclusive - mode Lock) หมายถึง ถ้ากลุ่มงาน Ti จองแบบพิเศษ (Exclusive - mode Lock แทนด้วย X) บนข้อมูล Q ดังนั้น Ti สามารถอ่าน และเขียนข้อมูล Q ได้ ซึ่งกลุ่มงานอื่นที่เข้ามาใช้งาน Q หลังจากนี้ก็จะต้องรอให้เลิกการจอง (Unlock) ก่อนจึงจะสามารถใช้งานข้อมูลจาก Q ได้

6. การทำตารางให้อยู่ในรูปแบบ นอร์มัล (Normal Forms)

รูปแบบนอร์มัลระดับที่ 1 (first normal form)

นิยาม รีเลชันใด ๆ อยู่ในรูปแบบนอร์มัลระดับที่ 1 (1NF) ก็ต่อเมื่อรีเลชันนั้นไม่มีกลุ่มที่ซ้ำกัน

S#	STATUS	CITY	P#	QTY
S1	20	London	P1	300
			P2	200
			P3	400
			P4	200
			P5	100
			P6	100
S2	10	Paris	P1	300
			P2	400
S3	10	Paris	P2	200
S4	20	London	P2	200
			P4	300
			P5	400

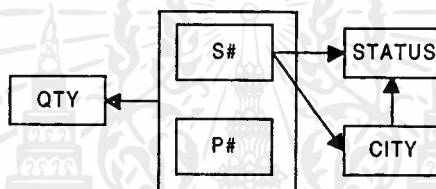
รูปที่ 8 แสดงรีเลชัน ที่ยังไม่เป็น 1NF

การปรับในระดับนี้คือ การขจัดกลุ่มที่ซ้ำกัน (Repeating groups) ซึ่งจะเห็นว่า แอตทริบิว P# และ QTY เป็นกลุ่มที่ซ้ำ ต้องทำการแยกข้อมูลโดย แยกข้อมูล S1 ออกเป็น 6 แถว S2 เป็น 2 แถว และ S4 เป็น 3 แถว โดยที่ S3 ยังคงเดิมเพราะมีเพียงแถวเดียว การปรับให้อยู่ในระดับ 1NF ดังกล่าวจะแสดงเป็นรีเลชัน First (S# , STATUS , CITY , P# , QTY) ดังรูปที่ 9

First

S#	STATUS	CITY	P#	QTY
S1	20	London	P1	300
S1	20	London	P2	200
S1	20	London	P3	400
S1	20	London	P4	200
S1	20	London	P5	100
S1	20	London	P6	100
S2	10	Paris	P1	300
S2	10	Paris	P2	400
S3	10	Paris	P2	200
S4	20	London	P2	200
S4	20	London	P4	300
S4	20	London	P5	400

รูปที่ 9 แสดงรีเลชัน First ที่ปรับให้เป็น 1NF แล้ว



รูปที่ 10 แสดงฟังก์ชันการขึ้นต่อกันของรีเลชัน First

จากรูปที่ 10 จะเห็นว่าเราต้องเพิ่ม แอดทริบิวต์ P# ร่วมกับคีย์หลัก S# ตัวเดิมเพื่อเป็นคีย์หลักตัวใหม่ คือ S# P# เพื่อสามารถที่จะกำหนด QTY ได้

จากการทำรูปที่ 8 ให้อยู่ในรูปแบบนอร์มัลระดับที่ 1 (1NF) ซึ่งสามารถขจัดกลุ่มซ้ำลงได้แล้ว แต่ยังคงเกิดปัญหาต่าง ๆ ตามมาอีก คือ ข้อมูลมีมากเกินไปจนเกิดความซ้ำซ้อนของข้อมูล จากตัวอย่าง เช่น S1, STATUS, CITY มีค่าของข้อมูลที่เหมือนกันอยู่ถึง 6 แถว ซึ่งนอกจากจะเป็นการสิ้นเปลืองเนื้อที่จัดเก็บข้อมูลในดิสก์ไดรฟ์แล้ว ยังเป็นปัญหาในการแก้ไขข้อมูลด้วย เราจะกล่าวถึงปัญหาต่าง ๆ ที่ตามมาดังนี้

การเพิ่มข้อมูล (Insert)

จากการทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 1 จะเห็นว่าคีย์หลักจะประกอบด้วย S# กับ P# เมื่อเราเพิ่มพนักงานชายคนใหม่คือ S5 ซึ่งอาศัยอยู่ที่ Athens เราจะไม่สามารถเพิ่มข้อมูลได้ จนกว่าพนักงานชาย S5 คนนั้น จะมีการขายสินค้าไปแล้ว (ต้องมี P#) (จากกฎข้อ 1 ของ โมเดลเชิงสัมพันธ์ ค่าของคีย์หลักจะต้องไม่เป็นค่าว่าง ซึ่งจากตัวอย่างนี้ คีย์หลักเป็นคีย์ประกอบ ซึ่งประกอบด้วย S# และ P#)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การลบข้อมูล (Delete)

ปัญหาจะเกิดขึ้น เมื่อมีการลบข้อมูลที่มีรหัสพนักงานขายคนนั้นเพียงแถวเดียว ตัวอย่างเช่น ถ้าต้องการจะลบข้อมูลการขาย ซึ่งสัมพันธ์กับแอตทริบิวต์ S# และ P# เมื่อลบแถวใดแถวหนึ่ง อาจสูญเสียข้อมูลที่สำคัญไป เช่น เมื่อลบข้อมูล รหัสพนักงานขาย S# ค่า S3 และรหัสการขาย P2 ออกจะทำให้เราสูญเสียข้อมูลพนักงานขายคนนั้นที่อาศัยอยู่ Paris

การแก้ไขข้อมูล (Update)

พิจารณาค่าของ CITY สำหรับแต่ละพนักงานขายซึ่งมีข้อมูลซ้ำหลายครั้ง จะเป็นปัญหาในการแก้ไขข้อมูล เช่น ถ้า S1 ย้ายที่อยู่จาก London ไป Amsterdam จะต้องทำการค้นหา S1 ทุกแถว ซึ่งในตัวอย่างมี 6 แถว แล้วทำการแก้ไข ค่าของ CITY จาก London เป็น Amsterdam ทุก ๆ แถว ซึ่งเป็นการเสียเวลามาก และหากเกิดการแก้ไข ค่าของแถวที่มี S1 เพียงบางแถว(ไม่ครบทุกแถว) ข้อมูลที่ได้จะเกิดความไม่สอดคล้องกันทันที

จากที่ได้กล่าวมาข้างต้น จะพบว่าปัญหาที่เกิดขึ้นนั้นจะเกิดจากการที่มีแอตทริบิวต์บางตัวขึ้นอยู่กับเพียงบางส่วนของคีย์หลัก เช่นตามตัวอย่างนี้ STATUS และ CITY ขึ้นกับ S# เพียงตัวเดียว แต่คีย์หลักประกอบด้วย S# และ P# รวมกัน ดังนั้นเราจะจัดการขึ้นต่อกัน เพียงบางส่วน โดยทำให้เป็น รูปแบบนอร์มัลระดับที่ 2

นิยาม เราเรียก แอตทริบิวต์ที่ไม่ใช่ส่วนหนึ่งของคีย์หลักว่า แอตทริบิวต์ นันคีย์ (nonkey attribute)

นิยาม รีเลชันใดๆจะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 2 ถ้ารีเลชันนั้นเป็น 1NF และไม่มีแอตทริบิวต์ นันคีย์ตัวใดขึ้นกับส่วนใดส่วนหนึ่งของคีย์หลัก

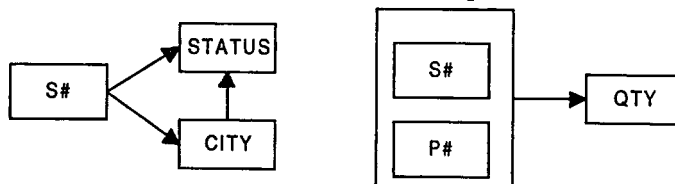
ปัญหาดังที่กล่าวมาแล้วในรีเลชัน First เราสามารถแก้ไขได้โดยแยกรีเลชัน First ออกเป็น 2 รีเลชันใหม่คือ

SECOND(S# , STATUS , CITY)

และ

SP(S# , P# , QTY)

ซึ่งฟังก์ชันการขึ้นต่อกันแสดงดังรูปที่ 11 และข้อมูลแสดงได้ดังรูปที่ 12 ดังนี้



รูปที่ 11 แสดงฟังก์ชันการขึ้นต่อกันใน รีเลชัน SECOND และ SP

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของ บริษัท ไทยพาณิชย์ จำกัด (มหาชน) ขอสงวนสิทธิ์ในเนื้อหาและข้อมูลทั้งหมดโดยไม่มีการอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SECOND

S#	STATUS	CITY
S1	20	London
S2	10	Paris
S3	10	Paris
S4	20	London
S5	30	Athens

SP

S#	P#	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

รูปที่ 12 แสดงรีเลชัน SECOND และ SP.

จากแผนภาพรูปที่ 12 แสดงตาราง SECOND และ SP ที่สามารถแก้ปัญหา ที่กล่าวมา ข้างต้น ได้ดังนี้

การเพิ่มข้อมูล (Insert)

เราสามารถเพิ่มรหัสพนักงานชายคนใหม่ S5 ซึ่งอาศัยอยู่ที่ Athens ได้ โดยที่ ไม่ต้องมีข้อมูลการขาย (P# รหัสการขาย) หรือต้องเคยขายสินค้าเกิดขึ้นมาก่อน

การลบข้อมูล (Delete)

เราสามารถลบข้อมูลการขายที่สัมพันธ์กันระหว่าง S3 และ P2 โดยลบแถวในตาราง SP โดยไม่ทำให้ข้อมูลประวัติของพนักงานชายที่อาศัยอยู่ใน Paris สูญหายไปด้วย

การแก้ไขข้อมูล (Update)

เราจะเห็นว่าแต่ละแถวไม่มีค่าของคีย์หลัก S# ซ้ำ จากปัญหาเดิมที่มีค่า S# ซ้ำหลายแถว ต้องตามไปแก้ทุกแถวจึงหมดไป แต่จะสังเกตเห็นว่าถ้าให้ค่า CITY มาหนึ่งค่า สามารถจะกำหนด หรือแสดงค่า STATUS หลายค่า(ข้อมูลซ้ำซ้อนมากเกินไปจนเกิดความจำเป็น redundancy) ถ้าเราเปลี่ยนค่า STATUS ที่ตรงกับ CITY = LONDON จาก 20 เป็น 30 เราจะต้องตามไปเปลี่ยนถึง 2 แถวคือ แถว S1 และ S4 เป็นการเสียเวลา และถ้าหากแก้ไขข้อมูลไม่ครบทุกแถวแล้ว จะทำให้ข้อมูล ไม่สอดคล้องกัน

วิธีแก้ไขปัญหาดังกล่าว คือ แยกรีเลชัน SECOND เป็น 2 รีเลชันใหม่ จะได้

SC (S# , CITY)

และ

CS (CITY , STATUS)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนภาพแสดงฟังก์ชันการขึ้นต่อกัน แสดงดังรูปที่ 13 และ ตารางข้อมูลของรีเลชัน SC และ CS แสดงดังรูปที่ 14



รูปที่ 13 แสดงฟังก์ชันการขึ้นต่อกัน ของรีเลชัน SC และ CS

S#	CITY
S1	London
S2	Paris
S3	Paris
S4	London
S5	Athens

CITY	STATUS
Athens	30
London	20
Paris	10
Rome	50

รูปที่ 14 แสดงรีเลชันที่ได้จากการแยกรีเลชัน SECOND

ตาราง รีเลชัน SC กับ CS จะมีการเชื่อมกันด้วยแอตทริบิวต์ CITY ซึ่งสามารถ แก้ปัญหาที่กล่าวมาข้างต้นที่ STATUS ขึ้นต่อกันกับ CITY ที่ไม่ใช่คีย์หลัก แล้วทำให้เกิด ความซ้ำซ้อนของ ข้อมูล

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 3 (3NF) ถ้ารีเลชันนั้นเป็น 2NF และ ทุก ๆ นันคีย์ในรีเลชันจะไม่ขึ้นกับนันคีย์

สำหรับวิธีการทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 3 ทำได้ดังนี้

1. ดึง แอตทริบิวต์นันคีย์ที่ขึ้นกับนันคีย์ด้วยกันไปสู่รีเลชันใหม่ ในตัวอย่าง คือ CITY → STATUS ดึงมาสร้างรีเลชัน CS

2. กำหนดให้นันคีย์ในรีเลชันใหม่ที่กำหนดนันคีย์อื่น เป็นคีย์ของรีเลชันใหม่

ในตัวอย่าง คือ จะให้ CITY เป็นคีย์หลักของรีเลชัน CS และกำหนด CITY ในรีเลชัน SC เป็น Foreign key

รูปแบบนอร์มัล ของบอยส์ คอดด์ (Boyce/Codd Normal Form : BCNF)

นิยาม รีเลชันหนึ่ง ๆ จะอยู่ในรูปแบบนอร์มัลของ บอยส์ คอดด์ ก็ต่อเมื่อ รีเลชันนั้น ๆ อยู่ใน รูปแบบนอร์มัลระดับที่ 3 และไม่มีแอตทริบิวต์อื่นในรีเลชันที่สามารถระบุค่า ของแอตทริบิวต์ที่เป็น คีย์หลัก หรือ ส่วนหนึ่งส่วนใดของคีย์หลักในกรณีที่คีย์หลักเป็นคีย์ผสม

โดยทั่วไป รูปแบบของรีเลชันที่มีโอกาสที่จะต้องผ่านการทำให้เป็นรูปแบบ BCNF มัก จะมีคุณสมบัติดังนี้ คือ

“ เป็นรีเลชันที่มีคีย์คู่แข่งหลายคีย์ (Multiple Candidate Key) โดยที่

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์คู่แข่งเป็นคีย์ผสม (Composite Key) และ
คีย์คู่แข่งนั้นมีความซ้ำซ้อนกัน (Overlapped) "

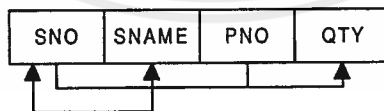
สำหรับตัวอย่างจะใช้ รีเลชัน SUPPLIER3 ในการอธิบายโดยสมมุติว่าชื่อของผู้ผลิต (SNAME) เป็นค่าไม่ซ้ำกัน และมีคุณสมบัติเป็นคีย์หลักได้เช่นกัน ดังรูปที่ 15

SUPPLIER3

SNO	SNAME	PNO	QTY
S1	SERI	P1	300
S1	SERI	P2	200
S1	SERI	P3	400
S1	SERI	P4	200
S1	SERI	P5	100
S1	SERI	P6	100
S2	WANIDA	P1	300
S2	WANIDA	P2	400
S3	SOMCHAI	P2	200
S4	ORAPIN	P2	200
S4	ORAPIN	P4	300
S4	ORAPIN	P5	400

รูปที่ 15 แสดงรีเลชัน SUPPLIER3 ที่ยังไม่เป็นรูปแบบ BCNF

จากรูปที่ 15 เป็นรีเลชันที่มีปัญหาเพราะมีคีย์คู่แข่งเป็นคีย์ผสม อาจจะเป็นแอตทริบิวต์รหัสผู้ผลิต และรหัสสินค้า (SNO และ PNO) หรือแอตทริบิวต์ชื่อผู้ผลิตและรหัสสินค้า (SNAME และ PNO) รีเลชันนี้ไม่ได้อยู่ในรูปแบบ BCNF เพราะเมื่อเลือกคีย์คู่แข่งใด เป็นคีย์หลักแล้ว คีย์คู่แข่ง ที่ไม่ได้ถูกเลือกจะยังปรากฏซ้ำซ้อนอยู่ในรีเลชันนี้ และมีคุณสมบัติในการระบุค่าของแอตทริบิวต์ที่เป็นคีย์หลัก ตัวอย่างเช่นหากเลือกแอตทริบิวต์ SNO และ PNO เป็นคีย์หลักแล้ว จะเกิดปัญหาในรีเลชันนี้ คือ แอตทริบิวต์ SNAME จะมีคุณสมบัติในการระบุค่าของแอตทริบิวต์ SNO ดังรูปที่ 16 นี้



รูปที่ 16 แสดงฟังก์ชันการขึ้นต่อกันของ รีเลชัน SUPPLIER3

นอกจากนี้ รีเลชันนี้ยังเป็นรีเลชันที่ประกอบด้วยข้อมูลที่ซ้ำซ้อนกัน อันจะก่อให้เกิดความผิดพลาดที่เกิดจากการเพิ่ม แก้ไขหรือปรับปรุงข้อมูล ตัวอย่างเช่น การเปลี่ยนชื่อ ของผู้ผลิต S1 จะต้อง แก้ไขหลายแถว ซึ่งอาจเกิดความผิดพลาดที่แก้ไขข้อมูลไม่ครบถ้วนได้

ดังนั้น รีเลชัน SUPPLIER3 จึงต้องทำการแตกรีเลชันออก โดยแยกแอตทริบิวต์ที่สามารถ ระบุค่าของคีย์คู่แข่งแยกเป็นอีกรีเลชันหนึ่งอาจทำได้สองกรณี คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีทีหนึ่ง SUPPLIER4 (SNO , SNAME)

มี SNO SNAME เป็นคีย์หลัก

ORDER2 (SNO , PNO , QTY)

มี SNO และ PNO เป็นคีย์หลัก

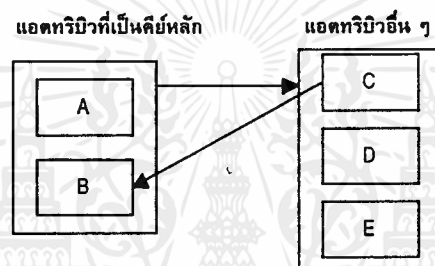
หรือ กรณีที่สอง SUPPLIER4 (SNO , SNAME)

มี SNO และ SNAME เป็นคีย์หลัก

ORDER2 (SNAME , PNO , QTY)

มี SNAME และ PNO เป็นคีย์หลัก

กล่าวโดยสรุปรูปแบบ BCNF เป็นรูปแบบที่ขยายขอบเขตของรูปแบบบรรทัดฐานขั้นที่ 3 โดยมี การตรวจสอบบรีเลชันว่ามีแอตทริบิวต์ที่ไม่ใช่คีย์หลัก แต่สามารถระบุค่าของคีย์หลักได้ ดังนี้



รูปที่ 17 แสดงรูปแบบการตรวจสอบ BCNF

รูปแบบนอร์มัลระดับที่ 4 (Fourth Normal Form : 4NF)

นิยาม รีเลชันจะอยู่ในรูปแบบนอร์มัลระดับที่ 4 ก็ต่อเมื่อ “ รีเลชันนั้น ๆ อยู่ในรูปแบบ BCNF และเป็นรีเลชันที่ไม่มีความสัมพันธ์ในการระบุค่าของแอตทริบิวต์แบบหลายค่า โดยที่แอตทริบิวต์ที่ถูกระบุค่าหลายค่าเหล่านี้ไม่มีความสัมพันธ์กัน (Independently Multivalued Dependency)

รีเลชันที่ต้องผ่านการทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 4 นี้จะต้องมีแอตทริบิวต์ อย่างน้อย สามแอตทริบิวต์ ดังรูป 18 SPJC เป็นรีเลชันที่ประกอบด้วยแอตทริบิวต์ รหัสผู้ผลิต (SNO) รหัสโครงการที่ส่งผลิต (PJNO) และชื่อ จังหวัดที่โรงงานของผู้ผลิตตั้งอยู่ (CITY) โดยกำหนดให้ ผู้ผลิตหลายโครงการ และผู้ผลิตหนึ่ง ๆ มีโรงงานตั้งอยู่ที่หลายจังหวัด (CITY)

SPJC

SNO	PJNO	CITY
S1	PJ01	BANGKOK
	PJ02	SAMUTHPRAKARN
S2	PJ03	RAYONG
	PJ04	CHONBURE
	PJ05	

→ ทำให้อยู่ใน
รูปแบบ BCNF โดย
ให้ทุกแอดทริบิว
ประกอบกันเป็น
คีย์หลัก

SPJC

SNO	PJNO	CITY
S1	PJ01	BANGKOK
S1	PJ01	SAMUTHPRAKARN
S1	PJ02	BANGKOK
S1	PJ02	SAMUTHPRAKARN
S2	PJ03	RAYONG
S2	PJ05	CHONBURE
S3	PJ04	RAYONG
S2	PJ04	CHONBURE
S2	PJ05	RAYONG
S2	PJ05	CHONBURE

รูปที่ 18 แสดงการแปลงรีเลชัน SPJCให้อยู่ในรูปแบบ BCNF

จากรูปที่ 18 รีเลชัน SPJC ประกอบด้วยแอดทริบิวท์สผู้ผลิต (SNO) ซึ่งมีความสัมพันธ์ในการระบุค่าแอดทริบิวท์สโครงการที่สั่งผลิต (PJNO) และแอดทริบิวท์จังหวัด ที่โรงงานของผู้ผลิตตั้งอยู่(CITY) ได้หลายค่า รีเลชัน SPJC อยู่ในรูปแบบ BCNF เมื่อมี แอดทริบิวท์ ทุกแอดทริบิวท์ เป็นคีย์ นอกจากนี้ รหัสผู้ผลิต (SNO) มีความสัมพันธ์ในการระบุค่า ของ รหัสโครงการที่สั่งผลิต (PJNO) แบบหลายค่า (SNO -> PJNO) ในขณะที่เดียวกัน รหัสผู้ผลิต (SNO) ก็มี ความสัมพันธ์ในการระบุค่าแบบหลายค่า กับชื่อจังหวัดที่ผู้ผลิตตั้งอยู่ (CITY) การที่แอดทริบิวท์ ทั้งสองไม่มีความสัมพันธ์กัน แต่มาอยู่ในรีเลชันเดียวกัน จะก่อให้เกิดความซ้ำซ้อนของข้อมูลเกิดขึ้น เพราะ ค่าของชื่อจังหวัดจะต้องปรากฏ ในรีเลชันซ้ำกัน เพื่อให้ข้อมูลครบถ้วน ถึงแม้ว่ารีเลชัน SPJC จะอยู่ในรูปแบบ BCNF โดยมี แอดทริบิวท์ทุกแอดทริบิวท์ประกอบกันเป็นคีย์หลักก็ตาม รีเลชัน นี้ก็ยังมีปัญหาแฝงอยู่เช่นกัน

ปัญหาความผิดพลาดที่เกิดขึ้นจากการเพิ่ม ลบ หรือปรับปรุงข้อมูลใน รีเลชัน SPJC มีโอกาสที่เกิดขึ้นมากมาย เช่น หากผู้ผลิตรหัส S2 ย้ายโรงงาน RAYONG ไปอยู่จังหวัดอื่น ก็ต้องทำการแก้ไขข้อมูลหลายแถว หรือความผิดพลาดที่เกิดจากการเพิ่มข้อมูลอาจจะทำไม่ได้ เช่น การเก็บข้อมูลชื่อจังหวัดที่ตั้งของผู้ผลิตคนใหม่ที่ยังไม่เคยได้รับงานโครงการใดจะทำไม่ได้ เป็นต้น การแก้ปัญหาทำได้โดยการแตกรีเลชัน SPJC ออกเป็นสองรีเลชัน ดังนี้

รีเลชัน SPJ ประกอบด้วยแอดทริบิวท์ SNO และ PJNO และมีแอดทริบิวท์ทั้งสองประกอบ กับ คีย์หลัก ดังรูปที่ 19 คือ

SPJ

SNO	PJNO
S1	PJ01
S1	PJ02
S2	PJ03
S2	PJ04
S2	PJ05

รูปที่ 19 แสดงรีเลชัน SPJ ที่เป็น 4NF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีเลชัน SC ประกอบด้วย แอตทริบิวต์ SNO และ CITY และมีแอตทริบิวต์ทั้งสองประกอบกันเป็น คีย์หลัก ดังรูปที่ 20 คือ

SC

SNO	CITY
S1	BANGKOK
S1	SAMUTHPRAKARN
S2	RAYONG
S2	CHONBURE

รูปที่ 20 แสดงรีเลชัน SC ที่เป็น 4NF

รูปแบบนอร์มัลระดับที่ 5 (Fifth Normal Form : 5NF)

นิยาม รีเลชันหนึ่ง ๆ จะอยู่ในรูปแบบนอร์มัลระดับที่ 5 หรือที่เรียกว่า Project-Join Normal Form (PJ/NF) ก็ต่อเมื่อ

“ รีเลชันนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 4 และเป็นรีเลชันที่มีคีย์หลัก เป็นคีย์ผสมที่ประกอบด้วยแอตทริบิวต์ตั้งแต่สามแอตทริบิวต์เป็นต้นไป หากมีการแตกรีเลชันออกเป็น รีเลชันย่อย สามรีเลชันย่อย (หรือมากกว่า) (Projection) ซึ่งเกิดจากการจับคู่แอตทริบิวต์แต่ละคู่ ของรีเลชัน เกิดเป็นคีย์ผสม และเมื่อทำการเชื่อมโยงรีเลชันย่อยทั้งหมด (Join) จะไม่ก่อให้เกิด ข้อมูลใหม่ ที่ไม่เหมือนรีเลชันเดิม ”

จากรูป 21 รีเลชัน SPP ประกอบด้วยแอตทริบิวต์ชื่อผู้ผลิต (SNAME) ชื่อสินค้า (PNAME) และชื่อโครงการ (PJNAME) โดยมีแอตทริบิวต์ทั้งสามเป็นคีย์หลัก

SPP

SNAME	PNAME	PJNAME
SERI	PEN	PROJ2
SERI	TABLE	PRCJ1
WANIDA	PEN	PROJ1
SERI	PEN	PROJ1

รูปที่ 21 แสดงรีเลชัน SPP ที่เป็นรูปแบบ 4NF

รีเลชัน SPP อยู่ในรูปแบบนอร์มัลระดับที่ 4 แต่ยังประสบปัญหาในการปรับปรุงแก้ไข ข้อมูล เช่น การลบข้อมูลของแถวที่ 3 (WANIDA , PEN , PROJ1) ข้อมูลของชื่อผู้ผลิตชื่อ WANIDA ก็จะไปหายจากฐานข้อมูล หรือ หากมีการเปลี่ยนชื่อสินค้าใดสินค้าหนึ่ง ก็จะต้อง ทำการปรับปรุง ข้อมูลหลายแถว เป็นต้น

เพื่อหลีกเลี่ยงปัญหาดังกล่าว จึงทำการแตกรีเลชัน SPP เป็นสามรีเลชันย่อย (PROJECTION) โดยในแต่ละรีเลชันย่อยจะประกอบด้วยแอตทริบิวต์แต่ละคู่เป็นคีย์ผสม ของรีเลชันย่อย ดังรูป 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SP:

SNAME	PNAME
SERI	PEN
SERI	TABLE
WANIDA	PEN

PPJ:

PNAME	PJNAME
PEN	PROJ2
TABLE	PROJ1
PEN	PROJ1

PJS:

PJNAME	SNAME
PROJ2	SERI
PROJ1	SERI
PROJ1	WANIDA

SNAME	PNAME	PJNAME
SERI	PEN	PROJ2
SERI	TABLE	PROJ1
WANIDA	PEN	PROJ1
SERI	PEN	PROJ1

รูปที่ 22 แสดงการแตกรีเลชัน SPP เป็นรีเลชัน SP , PPJ ,PJS และการนำมา join กลับ เมื่อแตกรีเลชัน SPP ออกเป็นรีเลชัน SP และ PPJ และ PJS แล้วหากนำ รีเลชันทั้งสามมาเชื่อมโยงกัน (JOIN) จะมีข้อมูลเหมือนในรีเลชันเดิมคือ SPP ไม่มีข้อมูลเหมือน ในรีเลชันเดิมคือ SPP ไม่มีลักษณะของแถวที่เกินมาที่เรียกว่า Spurious Tuple ดังรูปที่ 22

ดังนั้น เมื่อมีการแตกรีเลชันออกมา แล้วทำการเชื่อมโยงรีเลชันย่อยนั้นใหม่ หากไม่มีข้อมูลที่แตกต่างไปจากรีเลชันเดิม ก็สามารถจะแตกรีเลชันนั้นได้ แต่ถ้าหากแตกเป็นรีเลชันย่อยแล้วเกิดข้อมูลไม่เหมือนกับรีเลชันเดิม ก็ให้ถือว่ารีเลชันเดิมนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 5 แล้ว ตัวอย่าง เช่นจากรูป 23 รีเลชัน SCPJ เป็นรีเลชันที่อยู่ใน รูปแบบนอร์มัลระดับที่ 4 โดยมีแอตทริบิวท์สผู้ผลิต (SNO) ชื่อจังหวัดของผู้ผลิต (CITY) และรหัสโครงการ (PJNO) ประกอบกันเป็นคีย์หลัก รีเลชันนี้ยังมีปัญหาที่อาจเกิดความผิดพลาดในการเพิ่ม ปรับปรุง หรือ ลบข้อมูล จึงทำการแตกรีเลชัน SCPJ เป็นรีเลชัน SC CPJ และ PJS ซึ่งดูเหมือนจะเหมาะสม แต่ปัญหาที่เกิดขึ้นคือ เมื่อนำรีเลชันทั้งสามมาเชื่อมโยงจะมีข้อมูลเกินมา(Spurious Tuple) คือ S1 RAYONG PJ01 ซึ่งไม่มีในรีเลชันเดิมก็ให้ถือว่ารีเลชันนี้อยู่ในรูปแบบนอร์มัลระดับที่ 5

SCPJ

SNO	CITY	PJNO
S1	BANGKOK	PJ01
S1	RAYONG	PJ02
S3	RAYONG	PJ01

SC

SNAME	PNAME
SERI	PEN
SERI	TABLE
WANIDA	PEN

CPJ

PNAME	PJNAME
PEN	PROJ2
TABLE	PROJ1
PEN	PROJ1

PJS

PJNAME	SNAME
PROJ2	SERI
PROJ1	SERI
PROJ1	WANIDA

SNAME	PNAME	PJNAME
SERI	PEN	PROJ2
SERI	TABLE	PROJ1
WANIDA	PEN	PROJ1
SERI	PEN	PROJ1

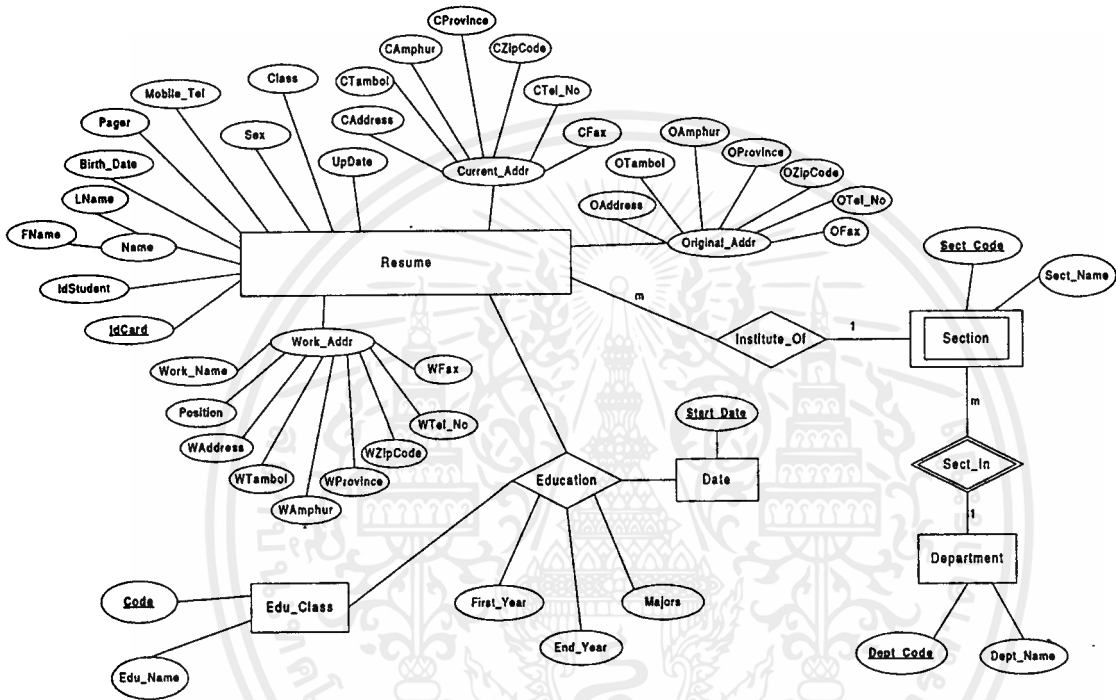
รูปที่ 23 แสดงตารางที่ทำให้อยู่ในรูปแบบนอร์มัล ระดับที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การวิจัยและการดำเนินงาน

1. ผลการศึกษาและวิเคราะห์ระบบงาน สามารถแสดงดังรูปที่ 24



รูปที่ 24 แสดงแผนผัง (E-R Model) ระบบฐานข้อมูลบัณฑิต

จากรูป สามารถทำการแยกออกเป็นตาราง และทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 5 ได้ดังนี้

Resume (ตารางประวัติ)

IdCard	หมายถึง	เลขที่บัตรประจำตัวประชาชน โดยจะเป็น คีย์หลัก
IdStudent	"	รหัสนักศึกษา
TFirstName	"	ชื่อ เป็นอักษรภาษาไทย
TLastName	"	นามสกุล เป็นอักษรภาษาไทย
BirthDate	"	วันเดือนปีเกิด
Email	"	ที่อยู่จดหมายอิเล็กทรอนิกส์
Class	"	รุ่นของบัณฑิตเมื่อตอนที่เป็นนักศึกษาในสถาบัน
Sex	"	เพศ
Mobile	"	หมายเลข เบอร์โทรศัพท์มือถือ
Pager	"	หมายเลขวิทยุติดตามตัว
Update	"	วันที่ทำการเปลี่ยนแปลงข้อมูล
OAddress	"	ที่อยู่ (บ้านเลขที่ ถนน หมู่) ภูมิลำเนา
OTambol	"	ตำบล/แขวง ภูมิลำเนา
OAmphur	"	อำเภอ/เขต ภูมิลำเนา
OProvince	"	จังหวัด ภูมิลำเนา
OZipCode	"	รหัสไปรษณีย์ ภูมิลำเนา
OTel_No	"	เบอร์โทรศัพท์ ภูมิลำเนา
OFax	"	เบอร์แฟกซ์ ภูมิลำเนา
CAddress	"	ที่อยู่ (บ้านเลขที่ ถนน หมู่) ปัจจุบัน
CTambol	"	ตำบล/แขวง ปัจจุบัน
CAmphur	"	อำเภอ/เขต ปัจจุบัน
CProvince	"	จังหวัด ปัจจุบัน
CZipCode	"	รหัสไปรษณีย์ ปัจจุบัน
CTel_No	"	เบอร์โทรศัพท์ ปัจจุบัน
CFax	"	เบอร์แฟกซ์ ปัจจุบัน
Work_Name	"	ชื่อสถานที่ทำงาน
Position	"	ตำแหน่ง/หน้าที่
WAddress	"	ที่อยู่ (บ้านเลขที่ ถนน หมู่) ที่ทำงาน
WTambol	"	ตำบล/แขวง ที่ทำงาน
WAmphur	"	อำเภอ/เขต ที่ทำงาน
WProvince	"	จังหวัด ที่ทำงาน
WZipCode	"	รหัสไปรษณีย์ ที่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WTel_No	หมายถึง	เบอร์โทรศัพท์ ที่ทำงาน
WFax	"	เบอร์แฟกซ์ ที่ทำงาน
Sect_Code	"	รหัสสาขาวิชาในสถาบัน
Dept_Code	"	รหัสภาควิชาในสถาบัน ทั้งสองรวมกันเป็นคีย์อ้างอิง

Department (ตารางภาควิชาในคณะที่เคยศึกษาในสถาบัน)

Dept_Code	หมายถึง	รหัสภาควิชา โดยจะเป็น คีย์หลัก
Dept_Name	"	ชื่อภาควิชา

Section (ตารางสาขาวิชาในคณะที่เคยศึกษาในสถาบัน)

Sect_Code	หมายถึง	รหัสสาขาวิชา
Dept_Code	"	รหัสภาควิชาเป็นคีย์อ้างอิง และทั้งสองคีย์รวมกันเป็นคีย์หลัก
Sect_Name	"	ชื่อ สาขาวิชา

Edu_Class(ตารางระดับการศึกษา)

Code	หมายถึง	รหัสระดับการศึกษา โดยจะเป็น คีย์หลัก
Edu_Name	"	ชื่อระดับการศึกษา เช่น มัธยมศึกษา , ปริญญาตรี

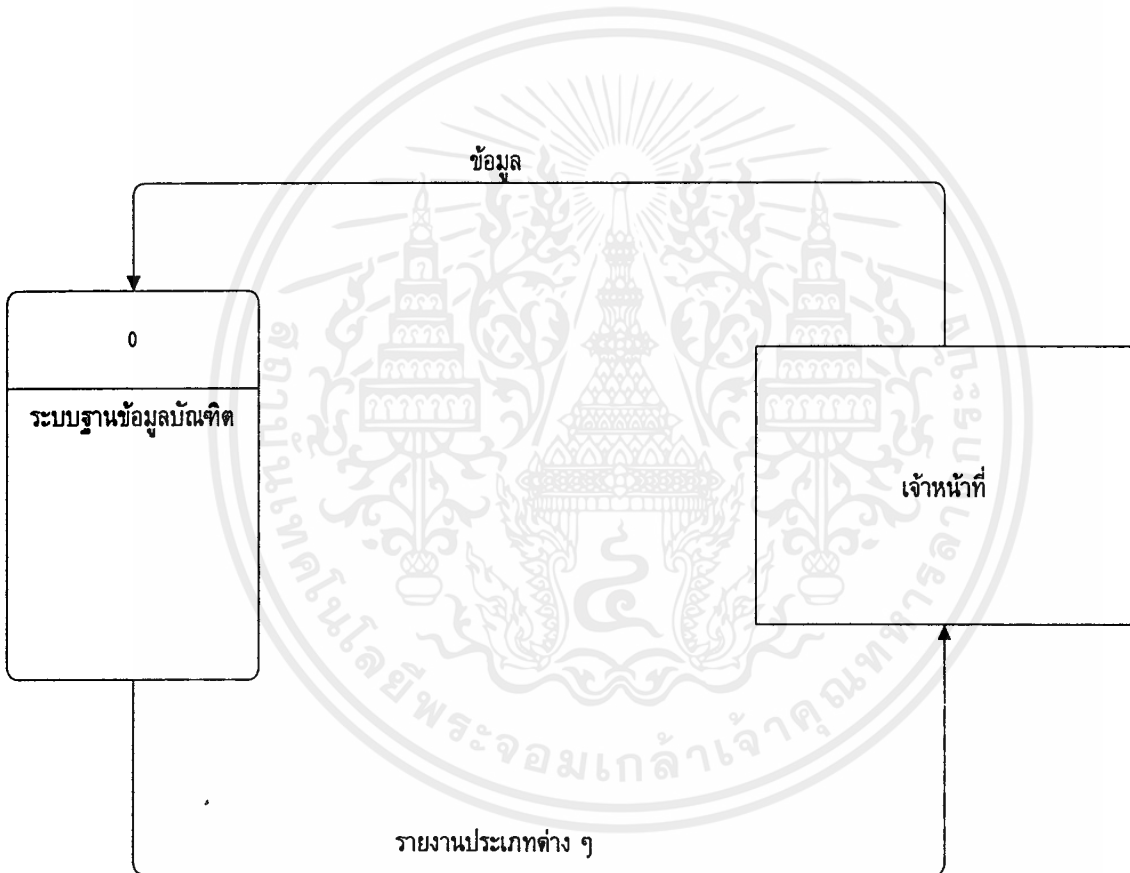
Education (ตารางการศึกษาทุกระดับ)

IdCard	หมายถึง	เลขที่บัตรประจำตัวประชาชนเป็นคีย์อ้างอิง
Code	"	รหัสระดับการศึกษาเป็นคีย์อ้างอิง
Start_Date	"	วันที่เริ่มศึกษา โดยทั้งสามคีย์รวมกันเป็น คีย์หลัก
First_Year	"	ปีที่เริ่มศึกษา
End_Year	"	ปีที่จบการศึกษา
Majors	"	สาขาวิชา

2. การออกแบบ Data Flow Diagram

ผลการวิเคราะห์ระบบสามารถนำมาออกแบบระบบฐานข้อมูลบัณฑิต โดยแสดงการไหลผ่านของข้อมูลในระบบงานได้ ดังรูปที่ 25

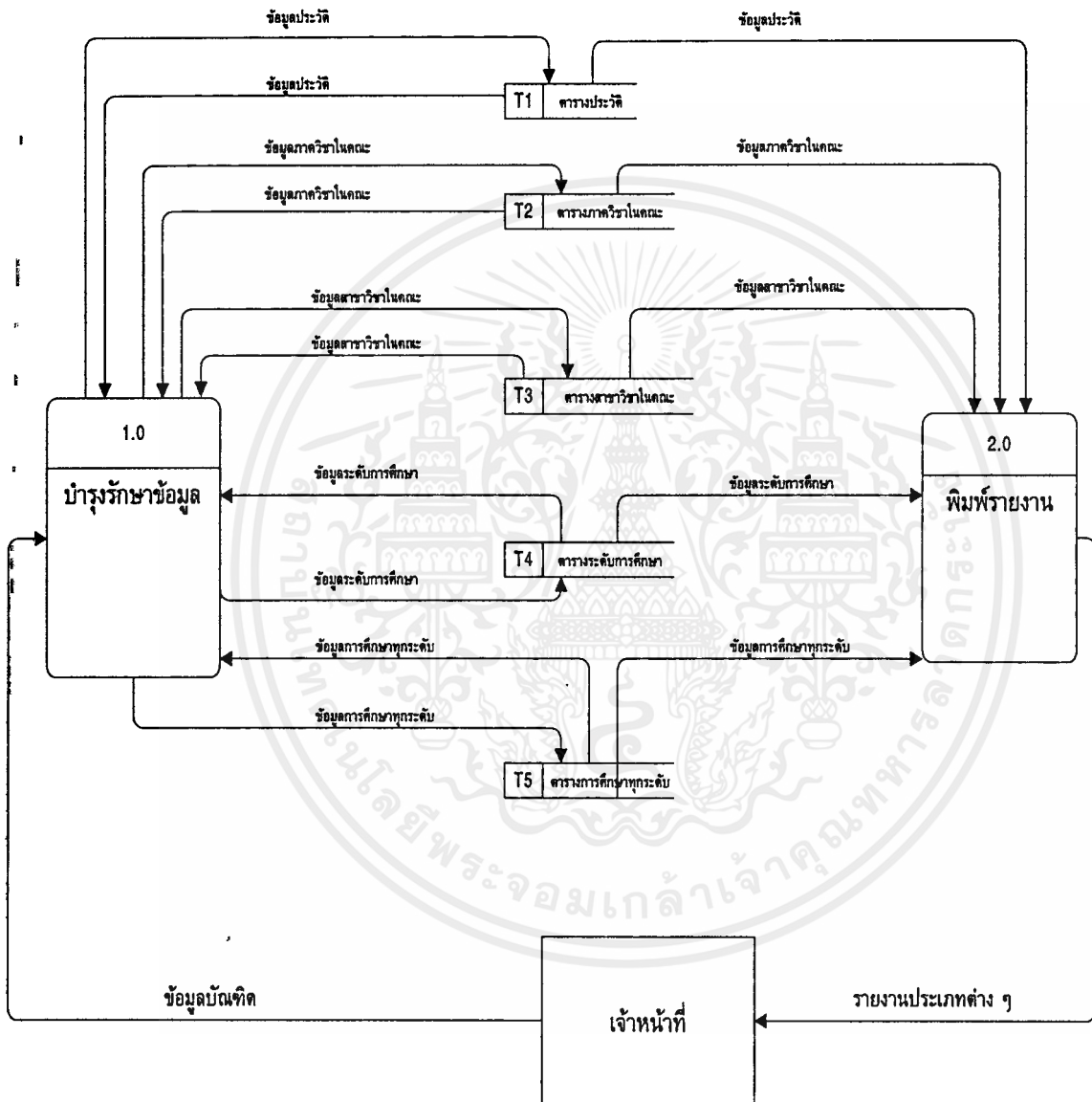
Context Diagram



รูปที่ 25 แสดง Context Diagram อธิบายความสัมพันธ์ของระบบฐานข้อมูลบัณฑิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

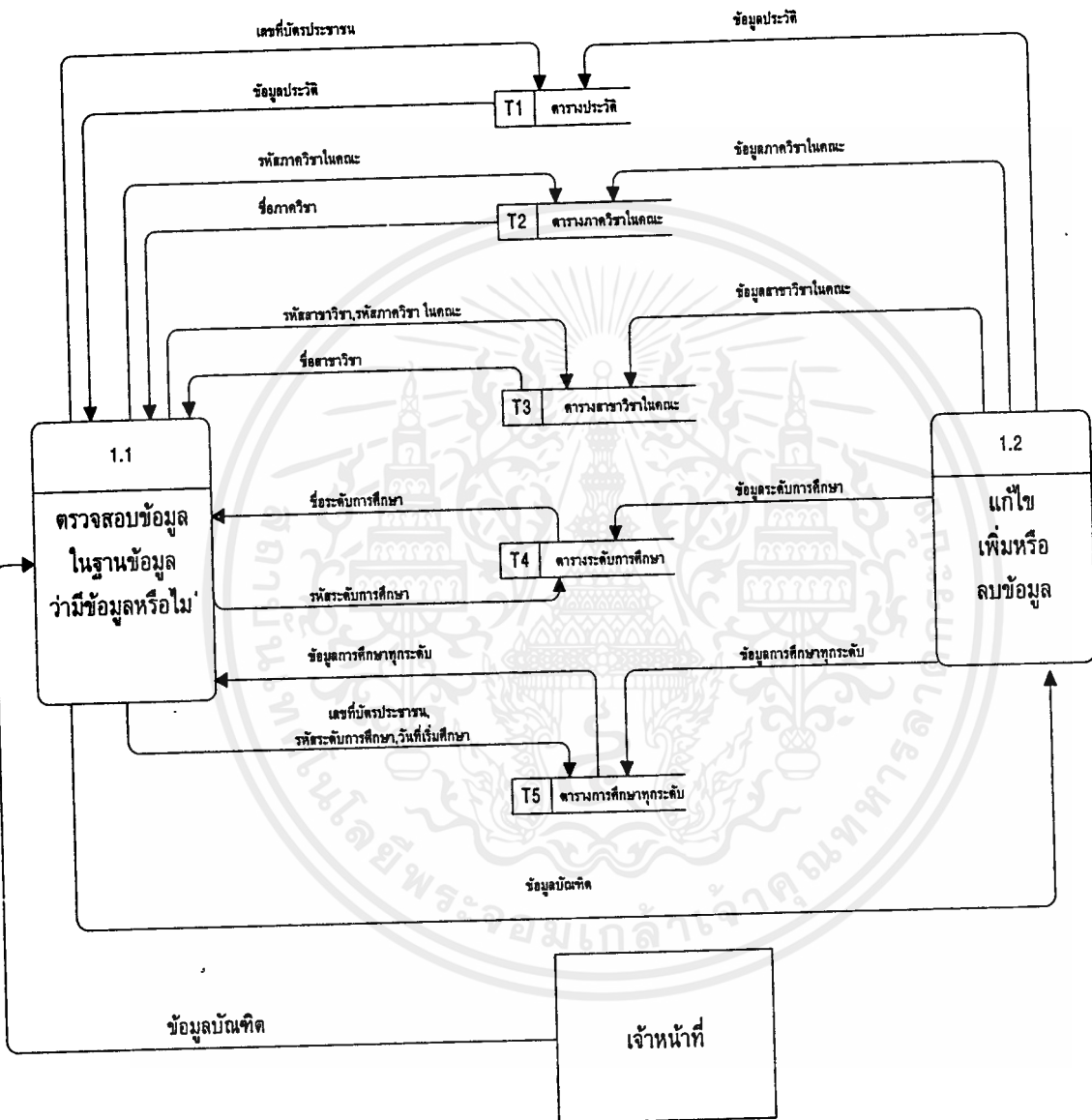
Level 0



รูปที่ 26 แสดง Level 0 ที่แสดงการทำงานของระบบภายในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

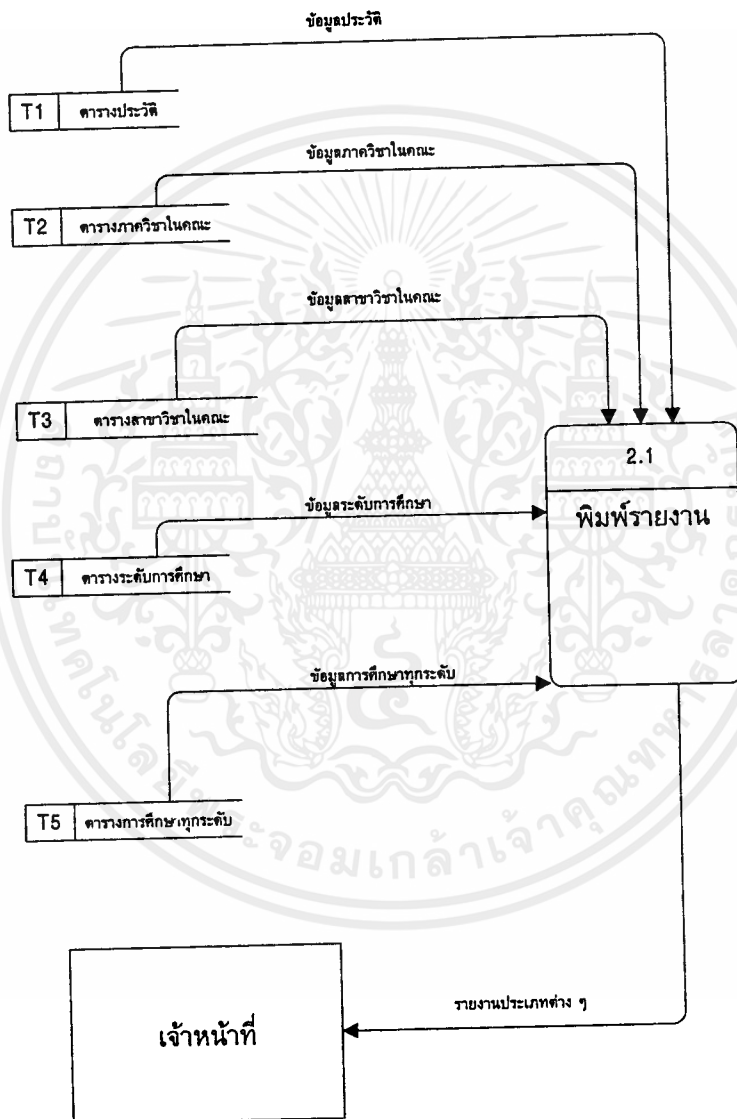
Level 1.1



รูปที่ 27 แสดง Level 1.1 ที่แสดงรายละเอียดของระบบการทำงานที่ได้จากการแตก Level 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Level 1.2



รูปที่ 28 แสดง Level 1.2 ที่แสดงรายละเอียดของรายงานของระบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการวิเคราะห์ปัญหาพิเศษ

ผลการวิจัยและวิจารณ์

ปัจจุบัน คณะวิทยาศาสตร์ของสถาบันได้ผลิตบัณฑิตใหม่เพิ่มมากขึ้นทุก ๆ ปี ซึ่งทางคณะได้มีการทำบันทึก และจัดเก็บประวัติบัณฑิตในรูปแบบของเอกสาร ซึ่งเมื่อทำการค้นหาเพิ่ม หรือปรับปรุงเปลี่ยนแปลงข้อมูล จะเกิดความล่าช้ามาก นอกจากนั้นการออกเอกสาร ใบรายงานข้อมูลประวัติส่วนตัวในแต่ละครั้ง เจ้าหน้าที่ทะเบียนประวัติต้องพิมพ์ข้อมูลใหม่ ทุกครั้ง ซึ่งทำให้เกิดความเบื่อหน่ายต่อหน้าที่ของเจ้าหน้าที่ฝ่ายทะเบียนประวัติ และ อาจทำให้เกิดความผิดพลาดได้ง่ายเมื่อกรอกข้อมูลเป็นจำนวนมาก

ข้อมูลที่ฝ่ายทะเบียนประวัติ ได้บันทึกประวัติบัณฑิตในปัจจุบัน ได้มีการจัดเก็บข้อมูลเพียงบางส่วนเท่านั้น เช่น ชื่อ สกุล ที่อยู่ (ภูมิลำเนา) ซึ่งจากข้อมูลดังกล่าว ยังขาดข้อมูลที่สามารถนำมาใช้เพื่อเป็นประโยชน์ต่อทางสถาบัน และตัวบัณฑิตเอง เช่น ควรจะมี ที่อยู่ ปัจจุบัน ตำแหน่ง และหน้าที่การงานที่กำลังทำอยู่ สถานะภาพกำลังศึกษาต่อหรือไม่และ อื่น ๆ ซึ่งข้อมูลเหล่านี้เป็นประโยชน์ ทำให้สามารถติดต่อบัณฑิตได้รวดเร็ว เพื่อขอความช่วยเหลือ และ แลกเปลี่ยนความรู้ระหว่างสถาบันกับสายงานที่บัณฑิตทำอยู่

จากปัญหาต่าง ๆ ที่กล่าวมาจึงควรมีการนำคอมพิวเตอร์เข้ามาช่วยในการดำเนินงานวิจัยขั้นนี้ได้เขียนด้วย โปรแกรมภาษา Delphi ซึ่งเป็นโปรแกรมระบบปฏิบัติการ Windows จึงมีคุณสมบัติพิเศษ คือ เป็นโปรแกรมลักษณะ Graphics User Interface (GUI) แสดงผลต่อผู้ใช้โปรแกรมในลักษณะรูปภาพ ทำให้ผู้ใช้สามารถเรียนรู้การใช้งานของโปรแกรมได้ง่าย สะดวก รวดเร็ว และข้อมูลมีความถูกต้องสูง

บทที่ 5

บทสรุปผลการศึกษาวิจัย และ ข้อเสนอแนะ

1. สรุปผลการศึกษาวิจัย

- 1.1 เนื่องจากแผนกทะเบียนประวัติ ต้องจัดเก็บประวัติบัณฑิตเป็นจำนวนมาก และคาดว่าจะเพิ่มมากขึ้นทุกๆ ปี ดังนั้นการจัดเก็บข้อมูลในรูปแบบเอกสาร จะทำให้เกิดความล่าช้า และเกิดข้อผิดพลาดได้ง่าย อีกทั้งยังใช้เนื้อที่เอกสาร ในการจัดเก็บเป็นจำนวนมาก
- 1.2 การจัดเก็บประวัติบัณฑิต ลงในคอมพิวเตอร์โดยใช้รูปแบบการจัดการฐานข้อมูลเชิงสัมพันธ์ โดยนำข้อมูลมาจัดเก็บบนระบบฐานข้อมูลเดียวกัน จึงทำให้การเรียกดูแก้ไข และเปลี่ยนแปลงข้อมูลสามารถทำได้รวดเร็ว และลดข้อผิดพลาดให้น้อยลง
- 1.3 ในโครงการปัญหาพิเศษนี้ มีขั้นตอนการดำเนินงาน 3 ขั้นตอน
 - 1.3.1 บันทึกประวัติบัณฑิต
 - 1.3.2 ค้นหา และ เพิ่ม ลบ หรือแก้ไขข้อมูล
 - 1.3.3 ค้นหา และ ออกใบรายงานตามการค้นหา
 - 1.3.3.1 รายงาน แบบแสดงข้อมูลประวัติทั้งหมด
 - 1.3.3.2 รายงาน แบบสลากชื่อ ที่อยู่ติดหน้าของจดหมาย
 - 1.3.3.3 รายงาน แบบจัดกลุ่มตามการค้นหา เช่น ตามจังหวัดเดียวกัน
- 1.4 เนื่องจากงานของแผนกทะเบียนประวัติ ต้องทำซ้ำ ๆ กันหลายครั้ง ซึ่งก่อให้เกิดความเบื่อหน่ายดังนั้นงานวิจัยนี้จึงได้เลือกใช้ตัวโปรแกรมที่แสดงผลในแบบ Graphical User Interface(GUI) คือติดต่อผู้ใช้ในลักษณะของรูปภาพ ซึ่งจะทำให้ตัวโปรแกรม ใช้งานง่าย ถูกต้อง แม่นยำ ทำให้สามารถลด ความเบื่อหน่ายของผู้ใช้ลงได้

2. ข้อเสนอแนะ

- 2.1 ควรมีการวิเคราะห์ระบบ โดยศึกษารายละเอียดของข้อมูลให้ถ่องแท้ ข้อมูลใด จำเป็นต่อการจัดเก็บ และเหมาะสมเพียงใด เพราะหากนำข้อมูลที่ได้รวบรวม มาออกแบบระบบ และเขียนโปรแกรมแล้วพบว่าข้อมูลที่ได้ไม่เหมาะสมต้องแก้ไข จะต้องออกแบบระบบใหม่ และ เขียนโปรแกรมใหม่ จะทำให้งานล่าช้ามาก
- 2.2 ควรมีระบบไฟฟ้าสำรอง เพราะเมื่อเกิดไฟดับก่อนที่จะทำการจัดเก็บข้อมูล จะทำให้ข้อมูลใหม่ไม่ได้มีการจัดเก็บ และอาจทำให้ระบบเสียหายทั้งระบบได้
- 2.3 ผู้ใช้ควรทำการ สำรองข้อมูลด้วยตัวเอง (Back up) ให้ทันสมัยอยู่เสมอ ซึ่งตัวโปรแกรมในงานวิจัยนี้มีส่วนให้ผู้ใช้ได้เลือกสำรองข้อมูล
- 2.4 โครงการปัญหาพิเศษนี้ได้วิเคราะห์ ออกแบบระบบงาน และเขียนโปรแกรม โดยการบันทึกข้อมูลประวัติบัณฑิต ดังนั้นหากต้องการเพิ่มเติมระบบงาน อื่นเช่น เพิ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบงานแผนกสมาชิกศิษย์เก่า ควรนำฐานข้อมูลประวัติบัณฑิตที่มีอยู่แล้ว ไปใช้ประโยชน์ในแผนกสมาชิกศิษย์เก่า หรือ แผนกอื่น ๆ ด้วย ตัวอย่างเช่น เชื่อมข้อมูลประวัติ เข้ากับข้อมูลสมาชิก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.
คู่มือการใช้โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

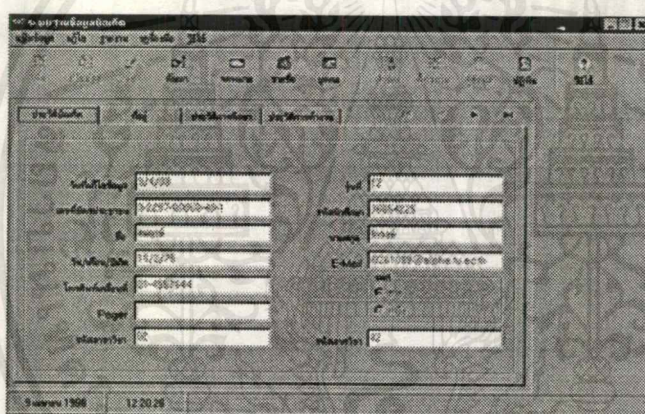
ระบบฐานข้อมูลบัณฑิตจัดทำขึ้นเพื่อจัดเก็บข้อมูลประวัติบัณฑิตเอาไว้เพื่อให้ง่ายต่อการใช้งานของผู้ใช้เมื่อต้องการค้นหา หรือตรวจสอบข้อมูลของบัณฑิตที่จบออกไปแล้ว ซึ่งทำให้การทำงานในด้านที่เกี่ยวกับ ประวัติบัณฑิตที่จบไปแล้วนั้นมีประสิทธิภาพมากขึ้น

การใช้ระบบฐานข้อมูลบัณฑิต

ช่วยแนะนำการใช้ระบบฐานข้อมูลให้แก่ผู้ใช้เพื่อให้ผู้ใช้สามารถที่จะใช้งานระบบฐานข้อมูลบัณฑิตนี้ได้ อย่างถูกต้องและมีประสิทธิภาพมากขึ้น และเมื่อผู้ใช้เกิดปัญหาสามารถที่จะแก้ปัญหาได้อย่างถูกต้อง

เปิดระบบฐานข้อมูลบัณฑิต

เมื่อผู้ใช้เริ่มต้นเรียกใช้ระบบฐานข้อมูลบัณฑิต ระบบฐานข้อมูลจะปรากฏหน้าจอตั้งแสดงในรูป



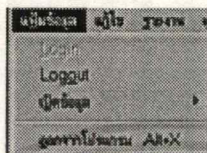
รูปที่ ก-1 แสดงระบบฐานข้อมูลบัณฑิต

ซึ่งระบบฐานข้อมูลจะเริ่มต้นโดยการแสดงหน้าจอของ ประวัติบัณฑิต และผู้ใช้สามารถที่จะเปลี่ยน ให้ระบบฐานข้อมูล แสดงหน้าจอของ ที่อยู่ ประวัติการศึกษา หรือ ประวัติการทำงาน ของบัณฑิตได้ โดยใช้ Mouse คลิกไปที่ ที่อยู่ ประวัติการศึกษา หรือ ประวัติการทำงาน ได้ หรือ เลือกที่เมนู เปิดเพิ่มข้อมูล ในเมนู หลักและเลือกที่เปิดข้อมูล เลือกเปิดข้อมูลตามความต้องการของผู้ใช้ ในส่วนของรหัสสาขาวิชา และ รหัสภาควิชานั้นผู้ใช้สามารถเรียกดูได้จากหมวดของ สาขาวิชา และ ภาควิชา ตามลำดับ

ผู้ใช้ที่เข้ามาทำการเรียกใช้ระบบฐานข้อมูลนั้นโดยปกติแล้วจะไม่สามารถทำการ เพิ่มเติม แก้ไข หรือ ลบข้อมูล ได้ แต่สามารถที่จะทำการ ค้นหา หรือ รายงาน ได้หากผู้ใช้ต้องการที่จะทำการแก้

ไข เพิ่มเติม หรือ ลบข้อมูลของ บัณฑิตนั้นจะต้องทำการ **ใส่รหัสผ่าน** เสียก่อนจึงจะสามารถแก้ไขข้อมูลได้

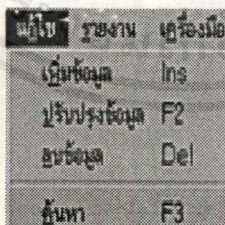
เมนูระบบฐานข้อมูลบัณฑิต
เพิ่มข้อมูล



รูปที่ ก-2 แสดงเมนูเปิดเพิ่มข้อมูล

- **Login** เลือกเมนูนี้เมื่อต้องการที่จะทำการแก้ไข เพิ่มเติม หรือ ลบข้อมูล
- **Logout** เลือกเมนูนี้เมื่อทำการ Login เข้าไปแก้ไข เพิ่มเติม หรือ ลบข้อมูลเสร็จเรียบร้อยแล้ว
- **เปิดข้อมูล** เลือกเมนูนี้เมื่อต้องการเปิดข้อมูลในหน้าต่าง ๆ คือ หน้า ประวัติบัณฑิต ที่อยู่ ประวัติ การศึกษา หรือ ประวัติการทำงาน หรือ หากต้องการเปิดดูหมวดของ ภาควิชา สาขาวิชา ระดับ การศึกษา
- **ออกจากโปรแกรม** เลือกเมนูนี้เมื่อต้องการที่จะออกจากการทำงานของระบบฐานข้อมูลบัณฑิต

แก้ไข



รูปที่ ก-3 แสดงเมนูแก้ไข

- **เพิ่มข้อมูล** เลือกเมนูนี้เมื่อต้องการเพิ่มข้อมูลประวัติของบัณฑิตใหม่
- **ปรับปรุงข้อมูล** เลือกเมนูนี้เมื่อต้องการที่จะทำการแก้ไขข้อมูลของบัณฑิต เมื่อบัณฑิตผู้นั้นมี การเปลี่ยนแปลงข้อมูล เช่น ย้ายที่อยู่ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ลบข้อมูล เลือกเมนูนี้เมื่อต้องการลบข้อมูลของบันทึกထွက်
- ค้นหา เลือกเมนูนี้เมื่อต้องการค้นหาข้อมูลของบันทึกเมื่อต้องการทราบประวัติของบันทึกบางคน

รายงาน

အမျိုးအမည်	ဖိနှိပ်
အချက်အလက်များ	Ctrl+E
အချက်အလက်	Ctrl+I
အချက်အလက်	Ctrl+F

รูปที่ ก-4 แสดงเมนูรายงาน

- แบบฉลากปิดของจดหมาย เลือกเมนูนี้เมื่อผู้ใช้ต้องการทำฉลากปิดของจดหมาย เพื่อปิดของจดหมายเมื่อต้องการที่จะส่งจดหมายติดต่อกับบันทึก
- แบบบัญชีรายชื่อ เลือกเมนูนี้เมื่อต้องการที่จะแสดงรายงานออกมาเป็นรายชื่อของบันทึก
- แบบรายบุคคล เลือกเมนูนี้เมื่อต้องการให้แสดงข้อมูลของบันทึกแต่ละบุคคล

เครื่องมือ

အမျိုးအမည်	ဖိနှိပ်
အချက်အလက်	Ctrl+B
အချက်အလက်	Ctrl+R
အချက်အလက်	Ctrl+Z
အချက်အလက်	Ctrl+L

รูปที่ ก-5 แสดงเมนูเครื่องมือ

- สำรองข้อมูล เลือกเมนูนี้เมื่อต้องการที่จะทำการสำรองข้อมูลในระบบฐานข้อมูลบันทึกเอาไว้
- ฟื้นฟูสภาพข้อมูล เลือกเมนูนี้เมื่อต้องการเรียกข้อมูลจากการสำรองข้อมูลเอาไว้กลับคืนมา
- กู้ข้อมูล เลือกเมนูนี้เมื่อต้องการเรียกข้อมูลบันทึกที่ลบไปแล้วกลับคืนมา
- ปฏิทิน เลือกเมนูนี้เมื่อต้องการเรียกดูวัน เดือน ปี

Toolbar ในระบบฐานข้อมูลบันทึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก-6 แสดง Toolbar ระบบฐานข้อมูลบัณฑิต

- เพิ่ม เลือกปุ่มนี้เมื่อต้องการเพิ่มข้อมูลของบัณฑิตใหม่
- ปรับปรุง เลือกปุ่มนี้เมื่อต้องการทำการแก้ไขข้อมูลของบัณฑิต เมื่อมีการเปลี่ยนแปลงข้อมูล
- ลบ เลือกปุ่มนี้เมื่อต้องการที่จะลบข้อมูล
- ค้นหา เลือกปุ่มนี้เมื่อต้องการค้นหาข้อมูลของบัณฑิตคนนั้น ๆ
- จดหมาย เลือกปุ่มนี้เมื่อต้องการให้ทำรายงานแบบฉลากปิดของจดหมาย เพื่อทำเป็นฉลากติดของจดหมาย เมื่อต้องการส่งจดหมายถึงบัณฑิต
- รายชื่อ เลือกปุ่มนี้เมื่อต้องการให้ทำรายงานออกมาเป็นแบบบัญชีรายชื่อ
- บุคคล เลือกปุ่มนี้เมื่อต้องการให้ทำรายงานประวัติบัณฑิตออกมาเป็นรายบุคคล
- สำรอง เลือกปุ่มนี้เมื่อต้องการทำการสำรองข้อมูลในฐานข้อมูลเอาไว้
- ฟื้นฟูสภาพ เลือกปุ่มนี้เมื่อต้องการที่จะเรียกข้อมูลที่ทำการสำรองเอาไว้กลับคืนมาใช้ใหม่
- กู้ข้อมูล เลือกปุ่มนี้เมื่อต้องการเรียกข้อมูลของบัณฑิตที่ลบไปแล้วกลับคืนมา
- ปฏิทิน เลือกปุ่มนี้เมื่อต้องการดู วัน เดือน ปี
- วิธีใช้ เลือกปุ่มนี้เพื่อแนะนำวิธีการใช้ระบบฐานข้อมูล

ประวัติบัณฑิต

ในส่วนของหน้าประวัติบัณฑิตนี้จะแสดงรายละเอียดของประวัติบัณฑิตดังรูป

รหัสบัณฑิต	114200	ปี	12
เลขที่บัตรประชาชน	7-2222-00000-00-1	วันเกิด	06/04/25
ชื่อ	สมชาย	นามสกุล	สมใจ
วันเดือนปีเกิด	15/02/75	E-Mail	0281899@ajphs.ac.th
โทรศัพท์มือถือ	01-6557604	ชื่อ	สมชาย
Page		ชื่อ	สมชาย
รหัสสาขา	12	รหัสภาควิชา	12

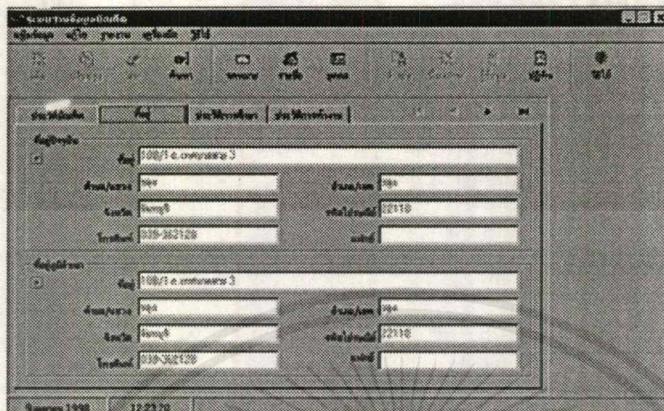
รูปที่ ก-7 แสดงการเปิดเพิ่มข้อมูลหน้าประวัติบัณฑิต

ผู้ใช้สามารถทำการ ค้นหา หรือ ทำรายงาน จากส่วนนี้ได้แต่ผู้ใช้ไม่สามารถที่จะ เพิ่มเติม แก้ไข หรือ ลบ ข้อมูลจากส่วนนี้ได้หากว่าไม่ได้ทำการใส่รหัสผ่าน เพื่อป้องกันมิสิทธิในการเพิ่มเติม แก้ไข หรือ ลบข้อมูล

ผู้ใช้สามารถที่จะเปิดหน้าของประวัติบัณฑิตได้โดยเลือกจาก เมนูเพิ่มข้อมูลและเลือกที่เปิดข้อมูล หรือเลือก คลิกที่ข้อความประวัติบัณฑิตในจอภาพ ในส่วนของรหัสสาขาวิชา และรหัสภาควิชา นั้น ผู้ใช้สามารถดูได้จากหมวด สาขาวิชา และ ภาควิชา ตามลำดับ

ที่อยู่

ในส่วนของหน้าที่อยู่นี้จะแสดงรายละเอียดที่อยู่บันทึกดังรูป



รูปที่ ก-8 แสดงการเปิดแฟ้มข้อมูลหน้าที่อยู่

ผู้ใช้สามารถทำการ ค้นหา หรือ ทำรายงาน จากส่วนนี้ได้แต่ผู้ใช้ไม่สามารถที่จะ เพิ่มเติม แก้ไข หรือ ลบ ข้อมูลจากส่วนนี้ได้หากว่าไม่ได้ทำการใส่ รหัสผ่าน เพื่อป้องกันมิสิทธิในการเพิ่มเติม แก้ไข หรือ ลบข้อมูล

ผู้ใช้สามารถที่จะเปิดหน้าของที่อยู่ได้โดยเลือกจาก เมนูแฟ้มข้อมูลและเลือกที่เปิดข้อมูล หรือเลือกคลิกที่ข้อความที่อยู่ ที่ปรากฏในจอภาพ

ประวัติการศึกษา

ในส่วนของหน้าประวัติการศึกษามันทีนี้จะแสดงรายละเอียดของประวัติประวัติการศึกษามันทีติดังรูป

เลขที่บัตรประชาชน	ชื่อและนามสกุล	วันเดือนปีเกิด	ปีการศึกษา	ชื่อโรงเรียน/มหาวิทยาลัย
3229700360491	01	15/5/90	1990	วิภา - ศศิธร
3229700360491	02	6/6/95	1995	พวงทองพวงทอง

รูปที่ ก-9 แสดงการเปิดแฟ้มข้อมูลหน้าประวัติการศึกษา

ผู้ใช้สามารถทำการ ค้นหา หรือ ทำรายงาน จากส่วนนี้ได้แต่ผู้ใช้ไม่สามารถที่จะ เพิ่มเติม แก้ไข หรือ ลบ ข้อมูลจากส่วนนี้ได้หากว่าไม่ได้ทำการใส่ รหัสผ่าน เพื่อบ่งบอกว่ามีสิทธิในการเพิ่มเติม แก้ไข หรือ ลบข้อมูล

ผู้ใช้สามารถที่จะเปิดหน้าของประวัติการศึกษาได้โดยเลือกจาก เมนูแฟ้มข้อมูลและเลือกที่เปิดข้อมูล หรือเลือก คลิกที่ข้อความประวัติการศึกษาในจอภาพ

ประวัติการทำงาน

ในส่วนของหน้าประวัติการทำงานบัณฑิตนี้จะแสดงรายละเอียดของประวัติการทำงานบัณฑิตหากว่าบัณฑิตผู้นั้น ได้ทำงานแล้วดังรูป

รูปที่ ก-10 แสดงการเปิดเพิ่มข้อมูลหน้าประวัติการทำงาน

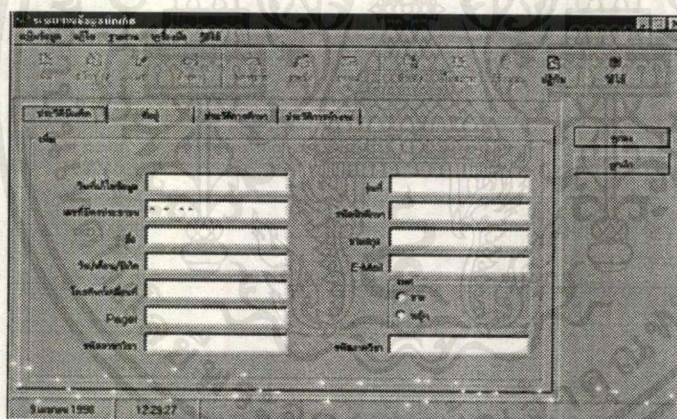
ผู้ใช้สามารถทำการ ค้นหา หรือ ทำรายงาน จากส่วนนี้ได้แต่ผู้ใช้ไม่สามารถที่จะ เพิ่มเติม แก้ไข หรือ ลบ ข้อมูลจากส่วนนี้ได้หากว่าไม่ได้ทำการใส่ รหัสผ่าน เพื่อบ่งบอกว่ามีสิทธิในการเพิ่มเติม แก้ไข หรือ ลบข้อมูล

ผู้ใช้สามารถที่จะเปิดหน้าของประวัติการทำงานได้โดยเลือกจาก เมนูเพิ่มข้อมูลและเลือกที่เปิดข้อมูล หรือเลือก คลิกที่ข้อความประวัติการทำงานในจอภาพ

การเพิ่มข้อมูล

ในการเพิ่มข้อมูลบัณฑิตนั้น เนื่องจากว่าระบบฐานข้อมูลบัณฑิตได้แสดงประวัติของบัณฑิตแยกกันไว้เป็นหน้า ๆ ออกไป ซึ่งแบ่งเป็นหน้าของ ประวัติบัณฑิต ที่อยู่ ประวัติการศึกษา และ ประวัติการทำงาน ดังนั้นหากว่าผู้ใช้ต้องการเพิ่มข้อมูลของบัณฑิตลงไปกระทำไดดังนี้

- เลือก เมนู แก้ไข ในเมนูหลัก และเลือกที่เพิ่มข้อมูล หรือ เลือกที่ปุ่มเพิ่มข้อมูลในToolbar เมื่ออยู่ในหน้าใด ๆ ที่ต้องการเพิ่มข้อมูล เช่น ต้องการเพิ่มข้อมูลประวัติบัณฑิต ก็ให้ระบบฐานข้อมูลอยู่ในหน้าของประวัติบัณฑิต และเลือกที่ปุ่มเพิ่มข้อมูล
- ถ้าเราต้องการเพิ่มข้อมูลในหน้าอื่นต่อไปให้เลือกที่หน้าที่ต้องการเพิ่มข้อมูลได้เลยไม่ต้องเลือกที่ปุ่มเพิ่มข้อมูลอีก
- เลือกปุ่มตกลงเมื่อต้องการเพิ่มข้อมูลจริง ๆ หรือ เลือกปุ่มยกเลิก เมื่อต้องการยกเลิกการเพิ่มข้อมูล ของบัณฑิตผู้นั้น
- ผู้ใช้ที่จะสามารถ เพิ่มข้อมูลได้จะต้องทำการ ใส่รหัสผ่าน เข้ามาเสียก่อน จึงจะสามารถเพิ่มข้อมูลได้



รูปที่ ก-11 รูปแสดงตัวอย่างของการเพิ่มข้อมูลหน้าของประวัติบัณฑิต

การแก้ไขข้อมูล

ในการแก้ไขข้อมูลบัณฑิตนั้น เนื่องจากว่าระบบฐานข้อมูลบัณฑิตได้แสดงประวัติของบัณฑิตแยกกันไว้เป็นหน้า ๆ ออกไป ซึ่งแบ่งเป็นหน้าของ ประวัติบัณฑิต ที่อยู่ ประวัติการศึกษา และ ประวัติการทำงาน ดังนั้นหากว่าผู้ใช้ต้องการเพิ่มข้อมูลของบัณฑิตลงไปกระทำได้ดังนี้

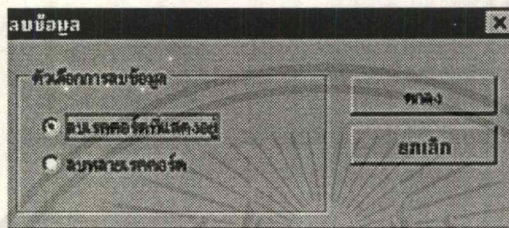
- เลือก เมนู แก้ไข ในเมนูหลัก และเลือกที่แก้ไขข้อมูล หรือ เลือกที่ปุ่มแก้ไขข้อมูลในToolbar เมื่ออยู่ในหน้าใดๆ ที่ต้องการแก้ไขข้อมูล เช่น ต้องการแก้ไขข้อมูลประวัติบัณฑิต ก็ให้ระบบฐานข้อมูลอยู่ในหน้าของประวัติบัณฑิต และเลือกที่ปุ่มแก้ไขข้อมูล
- ถ้าเราต้องการแก้ไขข้อมูลในหน้าอื่นต่อไปให้เลือกที่หน้าที่ต้องการแก้ไขข้อมูลได้เลยไม่ต้องเลือกที่ปุ่มแก้ไขข้อมูลอีก
- เลือกปุ่มตกลงเมื่อต้องการเพิ่มข้อมูลจริง ๆ หรือ เลือกปุ่มยกเลิก เมื่อต้องการยกเลิกการเพิ่มข้อมูล ของบัณฑิตผู้นั้น
- ผู้ใช้ที่จะสามารถ เพิ่มข้อมูลได้จะต้องทำการ ใส่รหัสผ่าน เข้ามาเสียก่อน จึงจะสามารถเพิ่มข้อมูลได้

รูปที่ ก-12 รูปแสดงตัวอย่างของการแก้ไขข้อมูลหน้าประวัติประวัติบัณฑิต

การลบข้อมูล

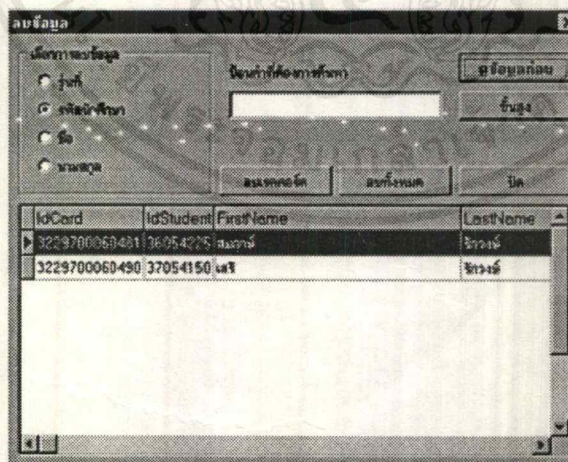
เนื่องจากระบบฐานข้อมูลบันทึกได้แบ่งประวัติบันทึกออกเป็นหน้า ๆ คือ หน้าของ ประวัติบันทึก ที่อยู่ ประวัติการศึกษา และ ประวัติการทำงาน ดังนั้นในการลบข้อมูลนั้นสามารถทำได้เมื่ออยู่ในหน้าใดก็ได้ โดยทำดังนี้

- เลือกที่เมนู แก้ไข ในเมนูหลัก และ เลือกที่ลบข้อมูล หรือ เลือกที่ ปุ่มลบข้อมูลใน Toolbar ระบบฐานข้อมูลจะปรากฏจอภาพดังรูป



รูปที่ ก-13 แสดงรูปภาพเมื่อต้องการลบข้อมูล

- ถ้าต้องการลบข้อมูลที่แสดงอยู่ ให้ใช้ Mouse เลือกที่ข้อความลบข้อมูลที่กำลังแสดงอยู่ และเลือกปุ่มตกลง
- ถ้าต้องการลบข้อมูลหลายเรคคอร์ด ให้ใช้ Mouse เลือกที่ข้อความลบหลายเรคคอร์ด และ เลือกที่ปุ่มตกลงจะ แสดงจอภาพดังรูป



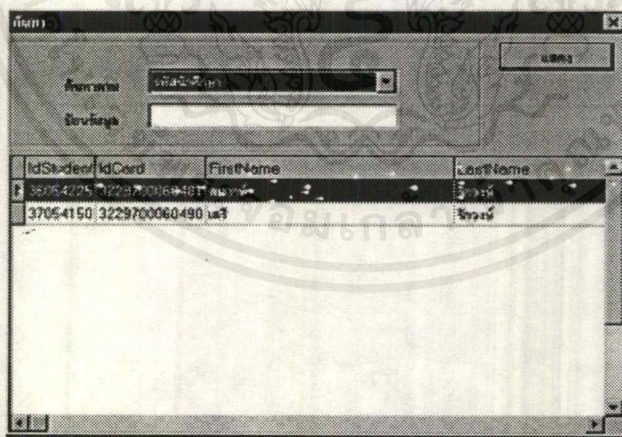
รูปที่ ก-14 แสดงรูปภาพเมื่อต้องการลบข้อมูลหลายคน

- เลือกลักษณะที่ต้องการทำการลบข้อมูล เช่น ลบเป็นรุ่นให้ เลือกที่รุ่นที่ โดยใช้ Mouse เลือกที่รุ่นที่
 - ป้อนค่าตามลักษณะที่ต้องการลบ ที่ช่อง ป้อนค่าเพื่อค้นหา เพื่อทำการค้นหาข้อมูลที่ต้องการลบ
 - เลือกที่ปุ่มดูข้อมูลก่อน เพื่อทำการดูข้อมูลที่ต้องการลบ ข้อมูลจะปรากฏที่ตารางข้างล่าง
 - เลือกข้อมูลที่ต้องการลบโดยใช้ Mouse เลือกที่ข้อมูลของบันทึกตัวนั้น
 - เลือกที่ปุ่ม ลบเรคอร์ด เพื่อทำการลบข้อมูลของบันทึกที่เลือกเอาไว้
 - หากต้องการเลือกข้อมูลทั้งหมดที่แสดงอยู่ในตารางให้เลือกที่ปุ่มลบทั้งหมด
 - เลือกที่ปุ่มปิดเพื่อออกจากการลบข้อมูล
- ข้อมูลที่ทำการลบไปแล้วสามารถที่จะ กู้ข้อมูล ขึ้นมาใหม่ได้

การค้นหาข้อมูล

เนื่องจากระบบฐานข้อมูลบันทึกได้แบ่งประวัติบันทึกออกเป็นหน้า ๆ คือ หน้าของ ประวัติบันทึกที่อยู่ ประวัติการศึกษา และ ประวัติการทำงาน แต่ในการค้นหาข้อมูลนั้นสามารถทำได้เมื่ออยู่ในหน้าใดก็ได้ โดยทำดังนี้

- เลือกที่เมนู แก๊ไข ในเมนูหลัก และ เลือกที่ ค้นหาข้อมูล หรือ เลือกที่ ปุ่มค้นหาข้อมูลใน Toolbar ซึ่งจะแสดงจอภาพดังรูป



รูปที่ ก-15 แสดงรูปภาพการค้นหาข้อมูล

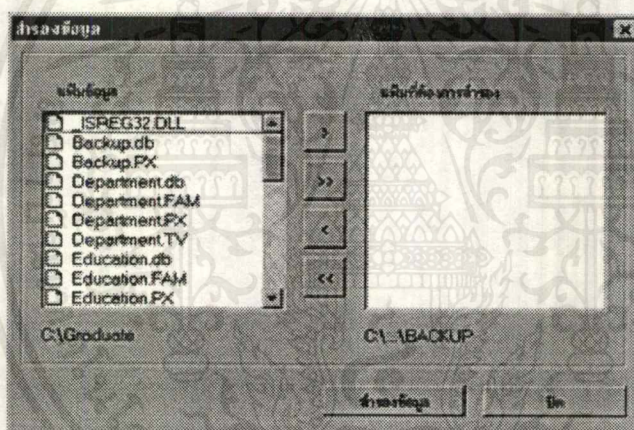
- ผู้ใช้สามารถที่จะค้นหาได้ตามรายการที่มีอยู่ในช่องค้นหาตามใน Dialog Box ค้นหาข้อมูล

- เลือกลักษณะที่ต้องการให้ค้นหาซึ่งค้นหาตาม เช่นค้นหาตามรหัสนักศึกษา ให้เลือกรหัสนักศึกษา
- กรอกข้อมูลที่ต้องการให้ค้นหาใน ช่องป้อนข้อมูล เช่น ในที่นี้เลือกค้นหาตามรหัสนักศึกษา ให้ป้อนข้อมูลรหัสของนักศึกษา
- เลือกที่ปุ่มแสดงเพื่อแสดงข้อมูลของบันทึกที่ต้องการทราบข้อมูล





สำรองข้อมูล

ผู้ใช้ที่จะสามารถที่จะทำการสำรองข้อมูลได้นั้นจะต้องใส่รหัสผ่านเข้ามาเสียก่อน เพื่อป้องกันกับระบบฐานข้อมูลบันทึกว่ามีสิทธิในการสำรองข้อมูล ผู้ใช้สามารถที่จะทำการสำรองข้อมูลได้โดย

- เลือกที่ เมฆูเครื่องมือและเลือกที่สำรองข้อมูล หรือเลือกที่ ปุ่มสำรองข้อมูลใน Toolbar และจะปรากฏจอภาพดังรูป



รูปที่ ก-16 แสดงรูปปรกพการสำรองข้อมูล

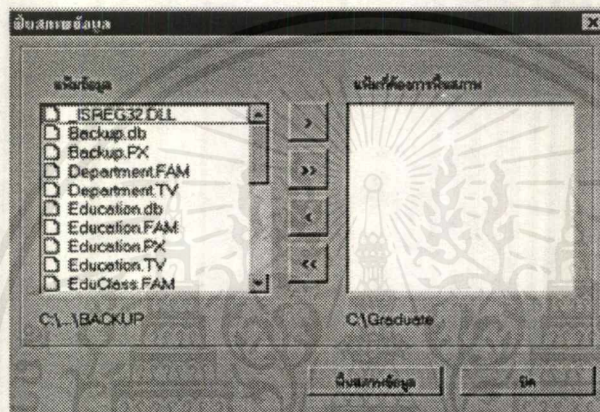
- เลือกไฟล์ที่ต้องการสำรองข้อมูลใช้ Mouse เลือกไฟล์ที่ต้องการสำรองและ เลือกที่ปุ่ม 
- หากต้องการสำรองข้อมูลทั้งหมดให้เลือกที่ปุ่ม 
- หากต้องการยกเลิกการสำรองข้อมูลไฟล์ใดให้เลือกไฟล์ที่ต้องการยกเลิกทางขวามือ และ เลือกปุ่ม 
- หากต้องการยกเลิกการสำรองข้อมูลทั้งหมดให้เลือกที่ปุ่ม 
- เลือกปุ่มสำรองข้อมูลเพื่อทำการสำรองข้อมูล
- เลือกปุ่มปิดเพื่อออกจากสำรองข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





ฟื้นฟูสภาพข้อมูล

ผู้ใช้ที่จะสามารถที่จะทำการฟื้นฟูสภาพข้อมูลได้นั้นจะต้อง ใส่รหัสผ่าน เข้ามาเสียก่อน เพื่อป้องกันกับ ระบบฐาน ข้อมูลบัณฑิตว่ามีสิทธิในการฟื้นฟูสภาพข้อมูล ในการฟื้นฟูสภาพข้อมูลนั้นระบบฐานข้อมูลจะทำ การเรียกข้อมูลที่ทำการ สำรองข้อมูล เอาไว้กลับขึ้นมาใช้อีกครั้ง ผู้ใช้สามารถที่จะทำการฟื้นฟูสภาพ ข้อมูลได้โดย

- เลือกที่ เมนูเครื่องมือและเลือกที่ฟื้นฟูสภาพข้อมูล หรือเลือกที่ ปุ่มฟื้นฟูสภาพข้อมูลใน Toolbar จะปรากฏจอภาพดังรูป



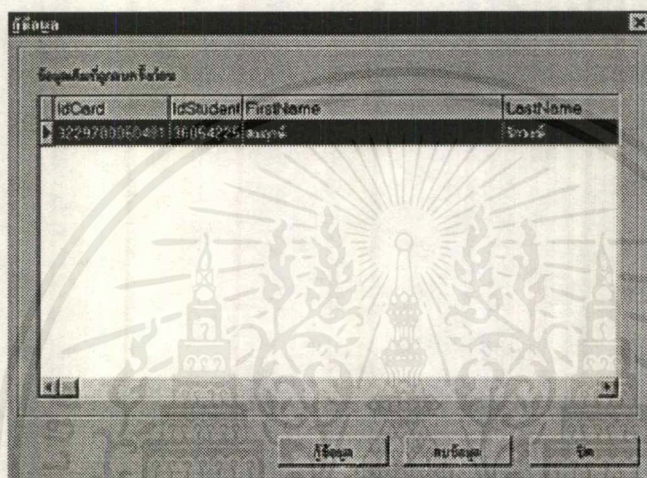
รูปที่ ก-17 แสดงรูปภาพการฟื้นฟูสภาพข้อมูล

- เลือกไฟล์ที่ต้องการฟื้นฟูสภาพขึ้นมาใหม่จากรายการไฟล์ทางซ้ายมือ โดยใช้ Mouse เลือกที่ไฟล์ที่ต้องการ
- เลือกที่ปุ่ม  ระบบฐานข้อมูลจะนำไฟล์นั้นมาไว้ทางขวามือของผู้ใช้
- เลือกปุ่ม  หากต้องการฟื้นฟูสภาพข้อมูลทั้งหมด
- เลือกปุ่มฟื้นฟูสภาพเพื่อทำการฟื้นฟูสภาพข้อมูล
- หากไม่ต้องการฟื้นฟูสภาพไฟล์ที่เลือกเอาไว้ให้เลือกไฟล์ทางขวามือ และ เลือกที่ปุ่ม 
- ถ้าไม่ต้องการฟื้นฟูสภาพข้อมูลทั้งหมดให้เลือกที่ปุ่ม 
- เลือกที่ปุ่มปิดเพื่อออกจากการฟื้นฟูสภาพข้อมูล

กู้ข้อมูล

ผู้ใช้ที่จะสามารถที่จะทำการกู้ข้อมูลได้นั้นจะต้อง ใส่รหัสผ่าน เข้ามาเสียก่อนเพื่อป้องกันระบบฐานข้อมูลบัณฑิตว่ามีสิทธิในการสำรองข้อมูล การกู้ข้อมูลนั้นจะทำการกู้ข้อมูลที่ผู้ใช้ทำการ ลบออกไปกลับคืนมา ผู้ใช้สามารถที่จะทำการกู้ข้อมูลได้โดย

- เลือกที่ เมนูเครื่องมือและเลือกที่กู้ข้อมูล หรือเลือกที่ ปุ่มกู้ข้อมูลใน Toolbar และจะปรากฏจอภาพดังรูป



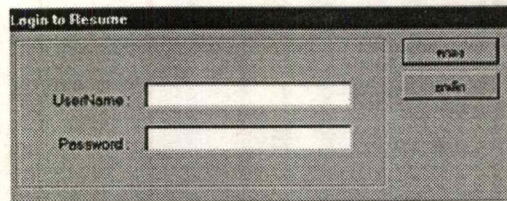
รูปที่ ก-18 แสดงรูปภาพการกู้ข้อมูล

- เลือกข้อมูลของบัณฑิตที่ต้องการกู้ขึ้นมา โดยใช้ Mouse เลือกที่บัณฑิตผู้นั้น
- เลือกที่ปุ่มกู้ข้อมูลเพื่อทำการนำข้อมูลของบัณฑิตผู้นั้นกลับขึ้นมาใช้ใหม่
- ถ้าต้องการลบข้อมูลนั้นโดยไม่ต้องการกู้ขึ้นมาอีกให้เลือกที่ปุ่มลบข้อมูล
- เลือกที่ปุ่มปิดเพื่อออกจากการกู้ข้อมูล

การรักษาความปลอดภัยระบบฐานข้อมูลบัณฑิต

จะกำหนดให้ผู้ใช้ที่เหมาะสมเท่านั้นที่มีสิทธิในการ เพิ่มเติม แก้ไข ลบ สำรองข้อมูล ฟื้นฟูสภาพข้อมูล กู้ข้อมูล โดยจะมีการกำหนด รหัสผ่านขึ้นมา ผู้ใช้สามารถใส่รหัสผ่านโดย

- เลือกที่ เมนูเพิ่มข้อมูลใน เมนูหลัก และ เลือก Login เพื่อทำการ Login เข้ามาแก้ไขข้อมูล โดยจะปรากฏจอภาพดังรูป



รูปที่ ก-19 แสดงรูปภาพทำการ Login เพื่อแก้ไขข้อมูล

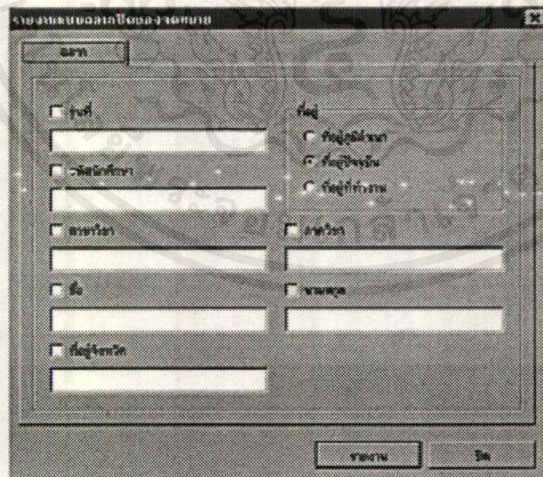
- ให้ใส่ UserName และ Password ลงในช่องที่ Dialog Box
- ทำการ Logout เมื่อทำการแก้ไขข้อมูลเสร็จ โดย เลือกที่ เมนูเพิ่มข้อมูลในเมนูหลัก และ เลือกที่ เมนู Logout
- หากใส่รหัสผ่านผิดจะไม่สามารถเข้าไปทำการแก้ไขข้อมูลได้

รายงาน

การทำรายงานข้อมูลบันทึกนั้นสามารถทำได้ในลักษณะดังต่อไปนี้

รายงานแบบฉลากปิดของจดหมาย

- เลือกที่เมนูรายงานใน เมนูหลัก และ เลือกที่ เมนูทำรายงานแบบฉลากปิดของจดหมาย หรือ เลือกที่ปุ่มจดหมาย ใน Toolbar ซึ่งจะปรากฏจภาพดังรูป



รูปที่ ก-20 แสดงรูปภาพการทำรายงานแบบฉลากปิดของจดหมาย

- เลือกที่อยู่ที่ต้องการส่งจดหมายให้บันทึกที่ช่องที่อยู่ ใช้ Mouse เลือกที่อยู่

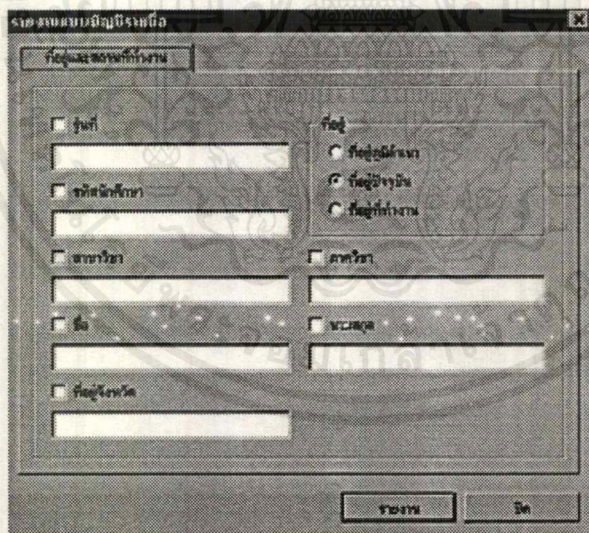
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือกลักษณะการทำรายงานจะเป็นตามรุ่น รหัสนักศึกษาของบัณฑิต สาขาวิชา ภาควิชา ชื่อ นามสกุล หรือ จังหวัด โดยสามารถที่จะเลือกได้มากกว่า 1 ชนิด เช่น ผู้ใช้ต้องการให้รายงานตามรุ่นที่ 13 และชื่อ เสรี เป็นต้น
- ใช้ Mouse คลิกที่หน้าช่องของลักษณะที่ต้องการให้ทำรายงาน
- กรอกข้อมูลในช่องด้านล่างของรูปแบบการทำรายงานที่เลือกเอาไว้
- ส่วนของรหัสสาขาวิชาและรหัสภาควิชาสามารถดูได้จากหมวด สาขาวิชา และ ภาควิชา ตามลำดับ
- เลือกปุ่มทำรายงานเพื่อทำรายงาน
- เลือกปุ่มปิดเพื่อออกจากการทำรายงานแบบฉลากปิดของจดหมาย

รายงานแบบบัญชีรายชื่อ

เมื่อผู้ต้องการให้มีการทำรายงานแบบบัญชีรายชื่อสามารถทำได้ดังนี้

- เลือกเมนูรายงาน ในเมนูหลัก และ เลือก เมนูบัญชีรายชื่อ หรือ เลือกจาก ปุ่มรายชื่อใน Toolbar จะปรากฏจอภาพดังรูป



รูปที่ ก-21 แสดงรูปภาพการทำรายงานแบบบัญชีรายชื่อ

- เลือกที่อยู่ที่ต้องการให้ทำรายงานที่ช่องที่อยู่ ใช้ Mouse เลือกที่อยู่

- เลือกลักษณะการทำรายงานจะเป็นตามรุ่น รหัสนักศึกษาของบัณฑิต สาขาวิชา ภาควิชา ชื่อ นามสกุล หรือ จังหวัด โดยสามารถที่จะเลือกได้มากกว่า 1 ชนิด เช่น ผู้ใช้ต้องการให้รายงานตามรุ่นที่ 13 และชื่อ เสรี เป็นต้น
- ใช้ Mouse คลิกที่หน้าช่องของลักษณะที่ต้องการให้ทำรายงาน
- กรอกข้อมูลในช่องด้านล่างของรูปแบบการทำรายงานที่เลือกเอาไว้
- ใสส่วนของรหัสสาขาวิชาและรหัสภาควิชาอันสามารถดูได้จากหมวด สาขาวิชา และ ภาควิชาตามลำดับ
- เลือกปุ่มทำรายงานเพื่อทำรายงาน
- เลือกปุ่มปิดเพื่อออกจากการทำรายงานแบบบัญชีรายชื่อ

รายงานแบบรายบุคคล

ผู้ใช้ต้องการให้ทำรายงานออกมาเป็นแบบรายบุคคลสามารถทำได้ดังนี้

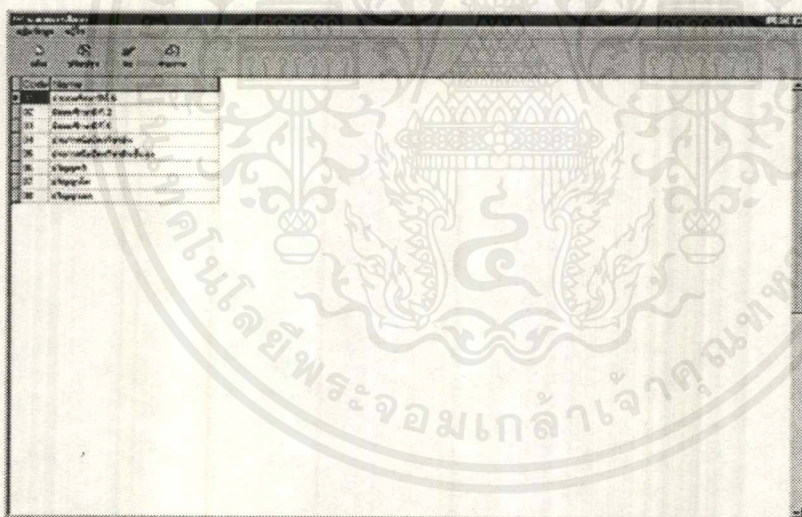
- เลือก เมนูรายงาน จากเมนูหลัก และเลือกเมนูทำรายงานแบบรายบุคคล หรือ เลือกจากคำว่าบุคคลใน Toolbar จะปรากฏจอภาพดังรูป

รูปที่ ก-22 แสดงรูปภาพการทำรายงานแบบรายบุคคล

- เลือกลักษณะที่ต้องการให้ทำรายงานโดยใช้ Mouse เลือกที่ตัวเลือกที่ตัวเลือกที่ต้องการให้ทำรายงาน เช่น ถ้าต้องการให้ทำรายงานออกมาเป็นรุ่น ก็ให้ใช้ Mouse เลือกที่รุ่นที่เพื่อให้ทำรายงานบันทึก ออกมาเป็นรุ่น
- กรอกข้อมูลในช่องว่าด้านล่างตามลักษณะที่เลือกเอาไว้ เช่น ต้องการให้ทำรายงานตามรุ่น ก็กรอกข้อมูลรุ่นที่ลงไป
- ในส่วนของภาควิชา และสาขาวิชานั้นสามารถดูรหัสภาควิชา และรหัสสาขาวิชาได้จากหมวดของภาควิชา หรือ สาขาวิชา ตามลำดับ
- เลือกที่ปุ่มแสดงเพื่อให้ฐานข้อมูลแสดงรายงานตามที่ใช้เลือกไว้
- เลือกที่ปุ่มปิดเพื่อออกจากการทำรายงานเป็นรายบุคคล

ระดับการศึกษา

- เลือกที่เมนูเพิ่มข้อมูล ในเมนูหลัก และ เลือก เปิดข้อมูล และ เลือก เมนูระดับการศึกษา จะปรากฏจอภาพดังรูป



รูปที่ ก-23 แสดงรูปภาพเพิ่มข้อมูลระดับการศึกษา

- หากต้องการแก้ไข หรือ เพิ่มเติมข้อมูลให้เลือกเมนูแก้ไข เลือกเพิ่มเติม หากต้องการเพิ่มเติมข้อมูล หรือ เลือก แก้ไข หากต้องการ แก้ไขข้อมูล หรือ เลือก ลบข้อมูล หากต้องการลบข้อมูลระดับการศึกษา หรือ เลือก ใน Toolbar ก็ได้
- เลือกที่ปุ่ม รายงาน ใน Toolbar เพื่อทำการเพิ่มรายงานขึ้นมา หากผู้ใช้ต้องการทำรายงาน

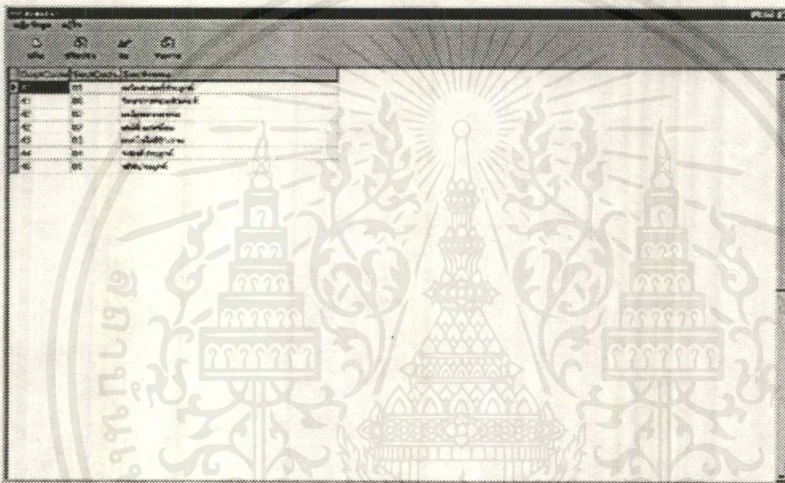
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือก เมนูเพิ่มข้อมูล และ เลือกเมนู ออกจากโปรแกรมหากต้องการออกจากหมวดระดับการศึกษา

ภาควิชา

ถ้าผู้ใช้ต้องการที่จะ เพิ่มเติม แก้ไข หรือ ดูรายชื่อ รหัสภาควิชาสามารถที่จะดูได้โดยทำดังต่อไปนี้

- เลือกที่เมนูเพิ่มข้อมูล ในเมนูหลัก และ เลือก เปิดข้อมูล และ เลือก เมนูภาควิชา จะปรากฏจอภาพดังรูป



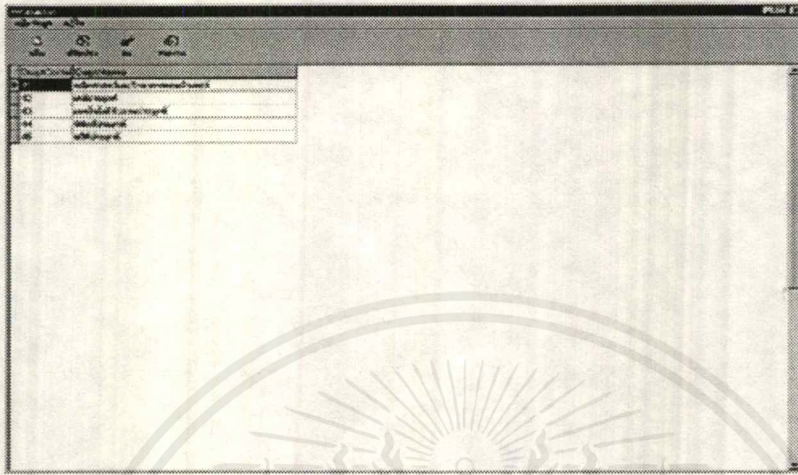
รูปที่ ก-24 แสดงรูปภาพเพิ่มข้อมูลภาควิชา

- หากต้องการแก้ไข หรือ เพิ่มเติมข้อมูลให้เลือกเมนูแก้ไข เลือกเพิ่มเติม หากต้องการเพิ่มเติมข้อมูล หรือ เลือก แก้ไข หากต้องการ แก้ไขข้อมูล หรือ เลือก ลบข้อมูล หากต้องการลบข้อมูล ภาควิชา หรือ เลือก ใน Toolbar ก็ได้
- เลือกที่ปุ่ม รายงาน ใน Toolbar เพื่อทำการเพิ่มรายงานขึ้นมา หากผู้ใช้ต้องการทำรายงาน
- เลือก เมนูเพิ่มข้อมูล และ เลือกเมนู ออกจากโปรแกรมหากต้องการออกจากหน้าภาควิชา

สาขาวิชา

ถ้าผู้ใช้ต้องการที่จะ เพิ่มเติม แก้ไข หรือ ดูรายชื่อ รหัสสาขาวิชาสามารถที่จะดูได้โดยทำดังต่อไปนี้

- เลือกที่เมนูเพิ่มข้อมูล ในเมนูหลัก และ เลือก เปิดข้อมูล และ เลือก เมนูสาขาวิชา จะปรากฏจอภาพดังรูป



รูปที่ ก-25 แสดงรูปภาพเพิ่มข้อมูลสาขาวิชา

- หากต้องการแก้ไข หรือ เพิ่มเติมข้อมูลให้เลือกเมนูแก้ไข เลือกเพิ่มเติม หากต้องการเพิ่มเติมข้อมูล หรือ เลือก แก้ไข หากต้องการ แก้ไขข้อมูล หรือ เลือก ลบข้อมูล หากต้องการลบข้อมูลสาขาวิชา หรือ เลือก ใน Toolbar ก็ได้
- เลือกที่ปุ่ม รายงาน ใน Toolbar เพื่อทำการเพิ่มรายงานขึ้นมา หากผู้ใช้ต้องการทำรายงาน
- เลือก เมนูเพิ่มข้อมูล และ เลือกเมนู ออกจากโปรแกรมหากต้องการออกจากหมวดสาขาวิชา



ภาคผนวก ข.
แบบฟอร์มกรอกประวัติบัณฑิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบฟอร์มกรอกประวัติบัณฑิต
เขียนวันที่..... เดือน..... พ.ศ.

ประวัติส่วนตัว :

ชื่อ - สกุล(นาย/นาง/นางสาว).....

วัน/เดือน/ปีเกิด.....

เลขที่บัตรประชาชน (โปรดใส่) ()-() () ()-() () () ()-() ()-()

ที่อยู่ปัจจุบัน :

เลขที่ หมู่ ซอย..... ถนน.....

ตำบล/แขวง อำเภอ/เขต จังหวัด.....

รหัสไปรษณีย์..... โทรศัพท์ Fax.....

โทรศัพท์เคลื่อนที่ วิทยุติดตามตัว(Pager).....

ที่อยู่ตามทะเบียนบ้าน :

เลขที่ หมู่ ซอย..... ถนน.....

ตำบล/แขวง อำเภอ/เขต จังหวัด.....

รหัสไปรษณีย์..... โทรศัพท์ Fax.....

E-Mail Address :

ประวัติการศึกษา :

ระดับการศึกษา	ปีจบการศึกษา	สถาบันการศึกษา	สาขาวิชา
มัธยมปลาย			
ปริญญาตรี		KMITL	
ปริญญาโท			
ปริญญาเอก			

กรณีที่ได้รับปริญญาบัตรหลายใบ โปรดระบุเพิ่ม

.....
.....
.....

หมายเหตุ กรณีที่ชำระค่าประจำตัวที่ใช้ในระหว่างการศึกษาใน KMITL ได้

ระดับการศึกษา	รหัสประจำตัว	วันที่
ปริญญาตรี/ปริญญาโท		

ประวัติการทำงาน :

ที่ทำงานปัจจุบัน : (ทำงาน)

ชื่อสถานที่ทำงาน ตำแหน่ง

เลขที่ หมู่ ซอย..... ถนน.....

ตำบล/แขวง อำเภอ/เขต จังหวัด.....

รหัสไปรษณีย์ โทรศัพท์..... Fax.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Nathan Wallace, Steve Tendon, Delphi2 Developers' Solutions, Waite Group Press, California, 1996, (pp 200-477)
2. Borland Delphi for Windows: User's Guide, Borland International., Inc, California, 1995
3. Borland Delphi for Windows: Database Application developer's Guide, Borland International., Inc, California, 1995
4. ดร. อำไพ พรประเสริฐสกุล, การวิเคราะห์และออกแบบระบบ, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ , 1994
5. ทรงศักดิ์ บรรจงมณี, แรกเริ่มเรียนรู้เรื่องการเขียนโปรแกรมด้วย DELPHI, บริษัทซีเอ็ดยูเคชั่น จำกัด(มหาชน) ,1997

