

ภาคผนวก

ภาคผนวก ก

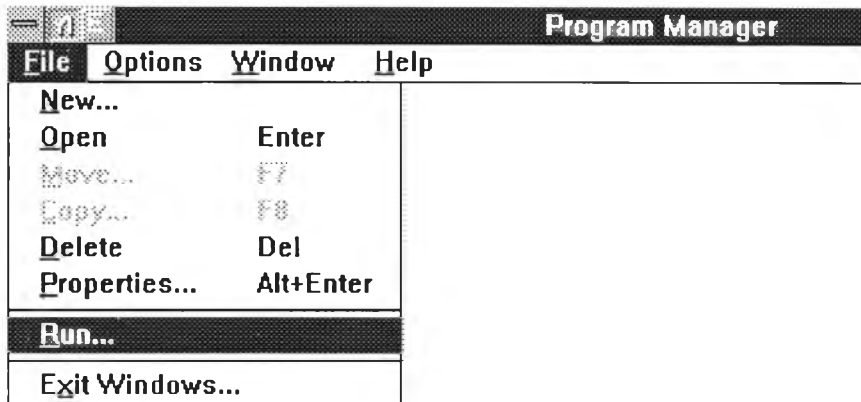
การติดตั้ง CFDTool

สำหรับ CFDTool ที่ได้ทำการพัฒนาขึ้นได้ทำการสร้างเป็นโปรแกรมแบบติดตั้งได้ด้วย โดยใช้โปรแกรมวิชวลเซทอัพของวิลซวลเบสิกรุ่น 3.0 ทำให้สามารถติดตั้งลงบนเครื่องคอมพิวเตอร์ได้โดยง่าย และมีขั้นตอนที่ต้องดำเนินการ เพื่อทำการติดตั้ง ดังนี้

1. เปิดเครื่อง แล้วเข้าสู่ระบบ วินโดว์ โดยการพิมพ์คำสั่ง WIN ดังตัวอย่างด้านล่าง

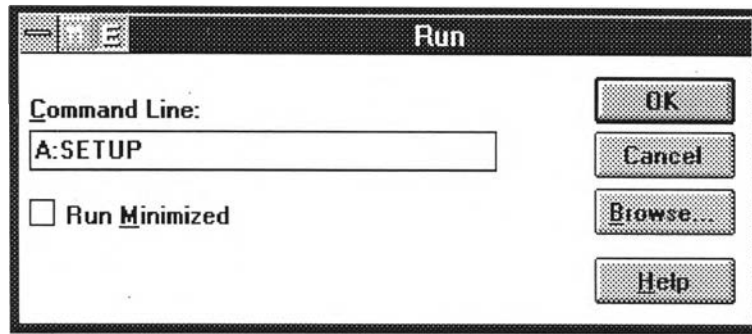
C:\> WIN กดปุ่ม ENTER

2. จากระบบวินโดว์ ให้ผู้ใช้เลือกเมนู File จะปรากฏ เมนูแบบผุด (popup menu) ขึ้นมา ต่อจากนั้นให้ผู้ใช้เลือกรายการ run ดังรูปที่ ก.1



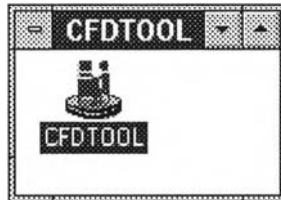
รูปที่ ก.1 แสดงการเลือกรายการ RUN เพื่อทำการติดตั้ง CFDTool ที่ได้พัฒนาขึ้น

3. นำแผ่นติดตั้งแผ่นที่ 1 ใส่ที่ไดรฟ์ A หรือ B
4. พิมพ์คำสั่งติดตั้ง ถ้าใช้ไดรฟ์ A ให้พิมพ์ A:\SETUP ถ้าใช้ไดรฟ์ B ให้พิมพ์ B:\SETUP จากนั้นให้นำแผ่นติดตั้งแผ่นที่ 2 ใส่ยังไดรฟ์เดียวกัน เมื่อโปรแกรมถาม ตัวอย่างในรูปที่ ก.2 เป็นการติดตั้งจากไดรฟ์ A ดังนั้นผู้ใช้ต้องนำแผ่นติดตั้งแผ่น 1 ใส่ไว้ที่ ไดรฟ์ A



รูปที่ ก.2 แสดงการเรียกใช้โปรแกรมติดตั้งจากไดรฟ์ A

หลังจากได้ดำเนินการครบ 4 ขั้นตอนแล้วโปรแกรมจะทำการติดตั้งตนเอง โดยสร้าง กลุ่มการใช้งานใหม่ ชื่อ CFDTool และสร้างไอคอน CFDTool ขึ้น ดังรูปที่ ก.3



รูปที่ ก.3 แสดงไอคอนของโปรแกรม CFDTool

ภาคผนวก ข

การแก้ไขโปรแกรม CFDTtools

สำหรับรายละเอียดในการแก้ไขโปรแกรม CFDTtool เพื่อให้สามารถใช้งานกับซอฟต์แวร์บริหารฐานข้อมูลประเภทอื่นได้ ให้ผู้ที่ต้องการแก้ไขโปรแกรม ดำเนินการดังต่อไปนี้ คือ

1.1 ดำเนินการแก้ไข เอ็มดีไอฟอร์มชื่อ mdiThesis

ให้ทำการแก้ไขในโมดูลชื่อ MDIForm_Load ในส่วนของการกำหนดค่า ซึ่งโดยปกติแล้ว จะมีการกำหนดค่าดังนี้

```
.....  
DefaultDT$ = "CHARACTER"  
aDBMS$(1) = "SQL"  
aDBMS$(2) = "ORACLE"  
aDBMS$(3) = "PROGRESS"  
aDBMS$(4) = "SYBASE"  
MaxDbms% = 4  
.....
```

รูปที่ ข.1 แสดงรายละเอียดตัวแปรที่ต้องแก้ไข

จากรูปที่ ข.1 ให้ดำเนินการแก้ไขดังนี้

1.1.1 ทำการเพิ่มชื่อของซอฟต์แวร์ฐานข้อมูลใหม่เข้าไปโดยกำหนดให้ Array aDBMS\$() ตัวต่อไป(จากตัวอย่างข้างต้นเป็นตัวที่ 5) เท่ากับชื่อของ ซอฟต์แวร์

1.1.2 เพิ่มค่าของตัวแปร MaxDbms% ขึ้นหนึ่งสำหรับการเพิ่มชื่อซอฟต์แวร์ 1 ชื่อซอฟต์แวร์ ในตัวอย่างจาก 4 ก็ให้เพิ่มเป็น 5

สมมุติต้องการเพิ่มซอฟต์แวร์ชื่อ newSoft หลังดำเนินการแล้ว รูปที่ ข.1 จะเปลี่ยนไปเป็น รูปที่ ข.2

```

.....
DefaultDT$ = "CHARACTER"
aDBMS$(1) = "SQL"
aDBMS$(2) = "ORACLE"
aDBMS$(3) = "PROGRESS"
aDBMS$(4) = "SYBASE"
aDBMS$(5) = "newSoft" -----> ชื่อซอฟต์แวร์ใหม่
MaxDbms% = 5 -----> เพิ่มเป็น 5
.....

```

รูปที่ ข.2 แสดงรายละเอียดตัวแปรที่หลังทำการแก้ไขโดยเพิ่มซอฟต์แวร์ชื่อ newSoft

1.2 ทำการแก้ไขฟอร์ม ชื่อ frmSql

โดยให้แก้ไขโมดูลชื่อ cmdSendSql_Click ซึ่งเดิมมีข้อมูลปรากฏในรูปที่ ข.3 คือ

```

Sub cmdSendSql_Click ()
  Select Case DBMS
  Case "SQL"
    doSQL
  Case "ORACLE"
    doOracle
  Case "SYBASE"
  Case "PROGRESS"
    doProgress
  End Select
End Sub

```

รูปที่ ข.3 แสดงรายละเอียดของโปรแกรมเดิมของโมดูล cmdSendSql_click

สำหรับโมดูลนี้ให้ดำเนินการแก้ไข ดังนี้

1.2.1 เพิ่ม Case ที่เท่ากับ ชื่อซอฟต์แวร์ที่ได้กำหนดไว้ในหัวข้อที่ 1.1 เข้าไป

1.2.2 กำหนดโพซีเตอร์ที่ต้องการให้โปรแกรมไปเรียกใช้งานเมื่อกรณีที่ใช้

ซอฟต์แวร์ที่เพิ่มเข้าไป

สมมุติต้องการเพิ่มซอฟต์แวร์ชื่อ newSoft ก็จะได้ผลลัพธ์ปรากฏในรูปที่ ข.4

```

Sub cmdSendSql_Click ()
  Select Case DBMS
  Case "SQL"
    doSQL
  Case "ORACLE"
    doOracle
  Case "SYBASE"
  Case "PROGRESS"
    doProgress
  Case "newSoft" -----> Case ที่เพิ่มขึ้น
    doNewSoft -----> ชื่อโมดูลที่ต้องการให้ทำงาน
  End Select
End Sub

```

รูปที่ ข.4 แสดงรายละเอียดโปรแกรมของโมดูล cmdSendSql_click หลังทำการแก้ไข

1.3 ดำเนินการเขียนโมดูลที่ใช้สำหรับการสร้างฐานข้อมูลของซอฟต์แวร์ ตามที่กำหนดชื่อไว้ใน หัวข้อที่ 1.2 ตัวอย่างการเขียนโปรแกรมในโมดูลให้ศึกษาจาก โมดูล doOracle , doProgeess และ doSQL ที่ได้พัฒนาจนสามารถใช้งานได้ โดยจากรูปที่ ข.4 ผู้ใช้ต้องทำการเขียนโมดูลชื่อ doNewSoft

ภาคผนวก ค

การรวบรวมข้อมูลและการกำหนดความสัมพันธ์

การรวบรวมข้อมูล

สำหรับการเก็บรวบรวมข้อมูลที่อยู่ในระบบงานเดิมแล้ว มีขั้นตอนย่อยต่าง ๆ ดังนี้

1 การรวบรวมเพิ่มข้อมูล

ขั้นตอนนี้เป็นการรวบรวมเพิ่มข้อมูลที่มีอยู่ในระบบเดิม โดยให้ทำการจัดบันทึกชื่อเพิ่มข้อมูล เพื่อเก็บเป็นข้อมูลของขั้นตอนต่อไป โดยการกำหนดชื่อในการจัดบันทึกนั้นควรกำหนดชื่อของเพิ่มข้อมูล ให้มีความหมายถึงสิ่งที่เก็บอยู่ในเพิ่มข้อมูล แต่บางครั้งการกำหนดชื่อเพิ่มข้อมูลให้มีความหมายถึงสิ่งที่เก็บนั้นอาจทำให้ชื่อเพิ่มข้อมูลที่ใช้มีขนาดยาวมากเกินไปจึงควรย่อชื่อเพิ่มข้อมูลเพื่อให้ได้ชื่อเพิ่มข้อมูลที่สั้นและกระชับรัดกุมเหมาะสมกับการจัดบันทึกเพื่อการใช้งาน ตัวอย่าง เพิ่มข้อมูลรายวิชาเปิดสำหรับการลงทะเบียนที่ต้องการใช้ชื่อว่า course offer for enrollment โดยหากต้องการจัดบันทึกให้ง่ายต่อการใช้งานก็ไม่ควรใช้ชื่อทั้งหมดมาเป็นชื่อเพิ่มข้อมูลควรทำการย่อชื่อเสียก่อน โดยสำหรับตัวอย่างที่ยกขึ้นมานี้อาจย่อและได้ชื่อเพิ่มข้อมูลเพื่อการใช้งานว่า crse_nrlmnt

2 การรวบรวมเขตข้อมูล

หลังจากทำการรวบรวมรายชื่อเพิ่มข้อมูลที่อยู่ในระบบเดิมแล้ว ในขั้นตอนต่อไปที่ต้องดำเนินการคือการรวบรวมเขตข้อมูล การรวบรวมเขตข้อมูลให้พิจารณาจัดบันทึกเขตข้อมูลตามลำดับการจัดบันทึกชื่อเพิ่มข้อมูล โดยให้เริ่มจากเพิ่มข้อมูลแรกที่จะบันทึกว่ามีเขตข้อมูลใดเป็นองค์ประกอบของเพิ่มข้อมูลอยู่บ้าง สำหรับรายละเอียดของเขตข้อมูลที่ต้องจัดบันทึกมีดังนี้

2.1 ชื่อเขตข้อมูล

ชื่อเขตข้อมูลเป็นรายละเอียดแรกของเขตข้อมูลที่ต้องจัดบันทึก ชื่อเขตข้อมูลที่ดีควรเป็นชื่อที่บอกให้ทราบว่าเขตข้อมูลนี้คืออะไร และเป็นชื่อที่กระชับไม่ยาวมาก และไม่ควรมีชื่อเขตข้อมูล 2 ชื่อที่อ้างถึงเขตข้อมูลเดียวกัน ตัวอย่างเช่น เขตข้อมูลรหัสนักศึกษาที่ใช้ชื่อว่า student_num แล้วก็ควรมีเขตข้อมูลอื่นที่มีชื่อที่แตกต่าง แต่หมายถึงเขตข้อมูลนี้อีก เพื่อความง่ายต่อการรวบรวมและจัดบันทึก ได้กำหนดวิธีการเขียนชื่อมาตรฐานสำหรับการจัดบันทึก ไว้ในรูปแบบต่อไปนี้

ชื่อเพิ่มข้อมูล.คำขยาย_คำจำแนกประเภท

และจากรูปแบบข้างต้นมีรายละเอียดของส่วนต่าง ๆ ดังนี้

1. ชื่อเพิ่มข้อมูล

หมายถึงชื่อของเพิ่มข้อมูลที่เขตข้อมูลนี้อยู่ ในการจัดบันทึกเพื่อโปรแกรมที่จัดสร้างขึ้นนี้ไม่ต้องทำการเขียนหรือจัดบันทึก

2. คำขยาย (Modify)

หมายถึง คำที่บ่งบอกคุณลักษณะของเขตข้อมูล เป็นคำที่บอกให้ทราบว่าเขตข้อมูลนี้คืออะไร ชื่อมาตรฐานที่สร้างขึ้นสามารถประกอบขึ้นจากคำขยายตั้งแต่ 1 คำขึ้นไป โดยอาจจะมีการใช้คำขยายก็ได้ แต่ไม่ควรยาวเกินไป และหากมีการใช้คำขยายมากกว่า 1 คำ ก็ให้ใช้สัญลักษณ์เส้นใต้ เป็นตัวคั่นระหว่างคำขยายเหล่านั้น ตัวอย่างเช่น คำขยาย student บอกให้ทราบว่าเป็นเรื่องราวเกี่ยวกับนักศึกษา หรือ คำขยาย thai บอกให้ทราบเกี่ยวกับประเทศไทย และหากต้องการสร้างคำขยายที่หมายถึงนักศึกษาไทย ก็อาจนำคำขยายทั้งสองมาเขียนต่อกันโดยใช้สัญลักษณ์เส้นใต้เป็นตัวคั่น โดยเขียนได้เป็น thai_student

3. คำจำแนกประเภท (Class word)

หมายถึง คำที่นำมาใช้แบ่งประเภทของเขตข้อมูลว่าเป็นอะไรโดยให้เขียนไว้เป็นส่วนสุดท้ายของชื่อมาตรฐาน และคั่นระหว่างคำขยายคำสุดท้ายกับคำจำแนกประเภทด้วยสัญลักษณ์เส้นใต้ ตัวอย่างหากต้องการตั้งชื่อมาตรฐานสำหรับเขตข้อมูลชื่อ *นักศึกษาภาษาไทย* ที่ประกอบด้วยคำขยาย "นักศึกษา" ที่ใช้คำว่า student แทน และคำขยาย "ภาษาไทย" ที่ใช้คำว่า thai แทน โดยมีคำจำแนกประเภทคือ "ชื่อ" ที่ใช้คำว่า name แทน เมื่อนำมารวมกันเป็นชื่อเขตข้อมูลได้ว่า student_thai_name อย่างนี้เป็นต้น

สำหรับคำจำแนกประเภทที่ใช้เพื่อจัดประเภทของเขตข้อมูลที่ ได้แนะนำไว้ได้แก่ คำดังต่อไปนี้

คำจำแนกประเภท	ความหมาย
AMOUNT	จำนวน
UNIT	หน่วยนับ
CODE	รหัส
DATE	วันที่
HOUR	ชั่วโมง
PERCENT	ร้อยละ
CONSTANT	ค่าคงที่
COUNT	ตัวนับ
NAME	ชื่อ
NUMBER	ตัวเลข หรือ หมายเลข
TEXT	ตัวอักษร หรือ ข้อความ
ADDRESS	ที่อยู่
TIME	เวลา

แม้ว่าการใช้ชื่อมาตรฐานทำให้ง่ายต่อการสร้างชื่อเขตข้อมูลและยังสื่อถึงความหมายของเขตข้อมูล แต่ยังมีปัญหาที่เกิดขึ้นจากการใช้ชื่อมาตรฐานเอง นั่นก็คือ หากชื่อมาตรฐานชื่อนั้นเป็นการนำคำขยายที่มีความยาวมากหลาย ๆ คำมารวมกับคำจำแนกประเภทเพื่อใช้เป็นชื่อเขตข้อมูล จะทำให้ได้ชื่อมาตรฐานที่มีความยาวมาก ไม่เหมาะสมกับการเก็บบันทึกไว้ ตัวอย่างเช่น เขตข้อมูลหมายเลขบัตรประจำตัวประชาชนของนักศึกษา ที่สามารถนำมากำหนดเป็นชื่อเขตข้อมูลที่เป็นไปตามรูปแบบที่

กำหนดไว้ คือ student_identifier_card_number ที่เป็นชื่อที่ยาวมาก ปัญหาที่กล่าวมานี้สามารถแก้ไขได้ โดยสร้างคำย่อขึ้นมาแทนคำขยายหรือ คำจำแนกประเภท ที่จะนำมาประกอบเป็นชื่อมาตรฐาน

การสร้างคำย่อเป็นเรื่องสำคัญ เพราะคำย่อจะถูกนำมาใช้ตลอดทั้งระบบงาน และไม่ควรมีคำขยาย หรือ คำจำแนกประเภทใดที่สามารถย่อได้หลายแบบ และควรมีการจดบันทึกคำย่อที่ได้สร้างขึ้นไว้เป็น พจนานุกรมคำย่อ เพื่อใช้เป็นข้อมูลอ้างอิงของการย่อคำอื่นต่อไป Candance C. Fleming และ Barbara von Halle (1989) ได้แนะนำกฎที่ใช้ในการย่อคำอย่างมีระเบียบไว้ 2 กฎ คือ

1. การย่อคำโดยกำหนดจำนวนตัวอักษร

เป็นวิธีง่าย ๆ ที่ทำงานโดยกำหนดว่าจะใช้ตัวอักษรกี่ตัวแรกของคำ มาเป็นคำย่อของคำนั้น มีข้อแนะนำเพิ่มเติมสำหรับกฎข้อนี้คือ จำนวนตัวอักษรที่ใช้ไม่ควรเกิน 5 ตัวอักษรแรกของคำ ตัวอย่างเช่น กำหนดกฎที่ใช้ตัวอักษรไม่เกิน 5 ตัวแรกเป็นคำย่อ เพื่อย่อคำ ดังนั้น คำว่า student ย่อเป็น stu หรือ stud ก็ได้

2. การย่อคำโดยตัดตัวอักษรที่เป็นสระในคำ

เป็นวิธีการย่อคำโดยตัดตัวอักษรที่เป็นสระ(ได้แก่ A E I O U)ออก จากคำที่ต้องการย่อ ตัวอย่าง ต้องการย่อคำว่า student โดยการตัดตัวอักษรที่เป็นสระออกก็สามารถย่อได้ เป็น stdnt

ในการย่อคำนั้นกฎของการย่อคำเป็นอะไรนั้นไม่ใช่เรื่องสำคัญ แต่ ความสำคัญของกฎของการย่อคำอยู่ที่ กฎที่ใช้ในการสร้างคำย่อต้องใช้กฎเดียวกันตลอดทั้งระบบงาน และ กฎที่ใช้สร้างคำย่อต้องได้รับการยอมรับจากทุกหน่วยงานกับกลุ่มผู้พัฒนาระบบงานด้วย

2.2 ประเภทของเขตข้อมูล

ประเภทข้อมูลเป็นข้อกำหนดที่ถูกกำหนดโดย ซอฟต์แวร์บริหารฐานข้อมูลที่จะนำ โมเดลข้อมูลไปใช้ ประเภทข้อมูลเป็นสิ่งที่บอกว่าเขตข้อมูลที่กำลังพิจารณาอยู่นี้สามารถเก็บข้อมูลแบบใด ได้ หากไม่ทราบว่าจะกำหนดประเภทข้อมูลเป็นอะไรได้บ้างก็ให้ใช้มาตรฐานของภาษาเอสคิวแอลก็ได้

2.3 ขนาดของเขตข้อมูล

เป็นการบอกให้ทราบว่าเขตข้อมูลที่กำลังพิจารณาอยู่ในขณะนี้ มีความยาวของ เขตข้อมูลเป็นเท่าไร และในกรณีที่เขตข้อมูลนั้นเป็นเขตข้อมูลที่สามารถเก็บข้อมูลที่เป็นทศนิยมได้ ต้องมี การจดบันทึกด้วยว่าเขตข้อมูลนั้น เขตข้อมูลนั้นสามารถมีทศนิยมได้สูงสุดกี่ตำแหน่ง ตัวอย่าง เขตข้อมูล ความสูงของนักศึกษา ที่มีหน่วยการวัดเป็นเมตร ก็เป็นเขตข้อมูลที่ยอมให้มีทศนิยมได้ โดยมีความยาวได้ 5 ตำแหน่งและมีทศนิยมได้ 2 ตำแหน่ง การจดบันทึกให้บันทึกเป็น 5.2 อย่างนี้เป็นต้น

2.4 ความจำเป็นต้องมีข้อมูลในเขตข้อมูลทุกระเบียนในแฟ้มข้อมูล

เป็นคุณสมบัติที่สำคัญมาก โดยเฉพาะอย่างยิ่ง เขตข้อมูลที่เป็นคีย์หลัก หรือ เขต ข้อมูลที่เป็นองค์ประกอบของคีย์หลัก เพราะเขตข้อมูลเหล่านั้นต้องมีค่าอยู่ ตัวอย่างเช่น ในระบบทะเบียน แฟ้มข้อมูลชื่อ ประวัตินักศึกษา ที่มีเขตข้อมูลรหัสนักศึกษาเป็นคีย์หลักของแฟ้มข้อมูล รหัสนักศึกษาจำเป็นต้องมีค่าทุกระเบียนข้อมูล เพราะว่าในความเป็นจริงแล้วไม่มีนักศึกษาคนใดในสถาบัน ๆ ที่ไม่มีรหัส ประจำตัว

2.5 มีค่าของเขตข้อมูลซ้ำกันไม่ได้ในแฟ้มข้อมูล

เป็นคุณสมบัติที่ต้องจดบันทึกไว้คุณสมบัติหนึ่ง เพราะว่ามีเขตข้อมูลบางเขตที่มีค่าซ้ำกันไม่ได้เลย โดยความเป็นจริงเขตข้อมูลเหล่านี้ได้แก่ เขตข้อมูลที่เป็นคีย์หลักของแฟ้มข้อมูล และเป็นคีย์หลักประเภทคีย์เดี่ยวด้วย สำหรับตัวอย่างขอใช้ตัวอย่างที่ผ่านมาแล้ว คือ เขตข้อมูลรหัสนักศึกษาของแฟ้มข้อมูลประวัตินักศึกษาในระบบทะเบียน ที่ไม่สามารถมีค่าซ้ำกันได้ เพราะในสถาบัน ใช้รหัสนักศึกษาในการจัดลำดับและแยกแยะระเบียบวนประวัติของนักศึกษาในแฟ้มประวัติและใช้อ้างอิงในการค้นหาด้วย

2.6 ชื่อแฟ้มข้อมูล

ชื่อแฟ้มข้อมูล เป็นการจดบันทึกเพื่อให้ทราบว่าเขตข้อมูลที่กำลังพิจารณาอยู่นั้นเป็นเขตข้อมูลที่อยู่ในแฟ้มข้อมูลใดบ้าง ในทางการดำเนินการนั้นพบว่าเขตข้อมูลหนึ่งเขตข้อมูลสามารถปรากฏอยู่ในแฟ้มข้อมูลได้หลายแฟ้มข้อมูล ตัวอย่าง ในระบบทะเบียน ที่มีเขตข้อมูลรหัสนักศึกษา เป็นเขตข้อมูลที่สามารถปรากฏอยู่ได้ทั้ง แฟ้มข้อมูลนักศึกษา และแฟ้มข้อมูลการลงทะเบียน ทำให้ในการจดบันทึกเขตข้อมูลนี้ต้องระบุลงไปว่าเป็นเขตข้อมูลที่อยู่ในแฟ้มข้อมูลทั้ง 2 แฟ้มข้อมูล

2.7 เป็นคีย์หลัก หรือองค์ประกอบของคีย์หลัก

เป็นคุณสมบัติที่บ่งบอกให้ทราบว่าเขตข้อมูลที่กำลังพิจารณาอยู่นี้เป็นคีย์หลักหรือเป็นองค์ประกอบของคีย์หลักของแฟ้มข้อมูลที่เขตข้อมูลนี้อยู่ ตัวอย่าง แฟ้มข้อมูลรายวิชาที่เปิดให้นักศึกษาลงทะเบียน ซึ่งประกอบด้วยเขตข้อมูลดังต่อไปนี้

```
crse_nrlmnt = (aca_c, sbjct_c, sctn_n, fcity_c, cmpls_c)
```

แล้วคีย์หลักของแฟ้มข้อมูลนี้ ได้แก่ aca_c + sbjct_c + sctn_n ซึ่งหากกำลังพิจารณาจดบันทึกเขตข้อมูล aca_c แล้วก็ต้องดำเนินการจดบันทึกว่าเขตข้อมูลนี้เป็นองค์ประกอบของคีย์หลักด้วย

3 การรวบรวมความสัมพันธ์ในแฟ้มข้อมูล

เป็นการจดบันทึกว่าแต่ละเขตข้อมูลในแฟ้มข้อมูลหนึ่ง ๆ มีความสัมพันธ์กับเขตข้อมูล หรือกลุ่มของเขตข้อมูลใดในแฟ้มข้อมูลเดียวกัน โดยให้ทำการจดบันทึกที่ละแฟ้มข้อมูลตัวอย่างเช่น กำหนดแฟ้มข้อมูลนักศึกษา(student) ที่มีโครงสร้างข้อมูลดังต่อไปนี้

```
student = (stdnt_c, stdnt_th_m, stdnt_english_m, cmpls_c,
           cmpls_th_m, cmpls_english_m, mjr_c, mjr_th_m,
           mjr_english_m, admtd_d, scss_f)
```

ซึ่งสามารถจดบันทึกความสัมพันธ์ได้เป็น

เขตข้อมูล	มีความสัมพันธ์กับ
stdnt_c	stdnt_c
stdnt_th_m	stdnt_c
stdnt_english_m	stdnt_c
cmpr_c	cmpr_c
cmpr_th_m	cmpr_c
cmpr_english_m	cmpr_c
mjr_c	mjr_c
mjr_th_m	mjr_c
mjr_english_m	mjr_c
admt_d	stdnt_c
sccss_f	stdnt_c

การดำเนินการเปลี่ยนข้อมูลที่เก็บรวบรวมมาเป็นโมเดลฐานข้อมูล

ในส่วนนี้เป็นการสมมุติตัวอย่างเพื่อช่วยให้ผู้อ่านเข้าใจความเหมาะสมของการเก็บรวบรวมข้อมูล และเพื่อให้เข้าใจว่าสามารถนำข้อมูลที่เก็บรวบรวมมาใช้งานได้อย่างไร โดยหลังจากเก็บรวบรวมข้อมูลจากระบบงานเดิมเรียบร้อยแล้ว ต้องนำข้อมูลเหล่านั้นมาผ่านขั้นตอนต่อไปนี้ คือ

1 การตรวจสอบความสัมพันธ์

เนื่องจากการรวบรวมข้อมูลในขั้นตอนที่ผ่านมาได้มีการกำหนดความสัมพันธ์ของเขตข้อมูลต่าง ๆ ที่ไปปรากฏอยู่ในแฟ้มข้อมูลว่า เขตข้อมูลเหล่านั้นมีความสัมพันธ์อะไรจึงไปปรากฏในแฟ้มข้อมูลนั้น ๆ เมื่อทำการเก็บรวบรวมจนครบทุกแฟ้มข้อมูลอาจพบว่ามีเขตข้อมูลบางเขตข้อมูลที่มีความสัมพันธ์หลายแบบ เช่น

แฟ้มใบทะเบียน = (ชื่อนักศึกษา, ภาคการศึกษา, วิชาที่เรียน, ตอนเรียน,)

แฟ้มประวัตินักศึกษา = (รหัสนักศึกษา, ชื่อนักศึกษา, วันเกิด,)

ในแฟ้มข้อมูลใบทะเบียน เขตข้อมูลชื่อนักศึกษานั้นมีความสัมพันธ์อยู่กับ ตัวเอง คือเขตข้อมูลชื่อนักศึกษา แต่ในแฟ้มข้อมูลประวัตินักศึกษา เขตข้อมูลชื่อนักศึกษานั้นมีความสัมพันธ์อยู่กับ เขตข้อมูลรหัสนักศึกษา ลักษณะนี้หมายความว่าเขตข้อมูลชื่อนักศึกษามีความสัมพันธ์ในระบบงานนี้ 2 แบบ

การดำเนินการในขั้นตอนนี้ให้พิจารณาเฉพาะเขตข้อมูลที่มีความสัมพันธ์หลายแบบโดยถ้าไม่มีเขตข้อมูลใดที่มีความสัมพันธ์หลายแบบให้ผ่านขั้นตอนนี้ไป แต่ถ้ามีเขตข้อมูลใดเป็นเขตข้อมูลที่มีความสัมพันธ์หลายแบบให้เลือกเพียงแบบใดแบบหนึ่ง เพียงแบบเดียวเท่านั้น โดยจากตัวอย่างด้านบนขอเลือกความสัมพันธ์ของชื่อเขตข้อมูลว่ามีความสัมพันธ์กับเขตข้อมูลรหัสนักศึกษา

2 การสร้างตารางข้อมูล

จากข้อมูลที่ได้เก็บรวบรวมให้ทำการจับกลุ่มของเขตข้อมูล ตามความสัมพันธ์ที่ได้จัดบันทึกไว้ โดยเมื่อดำเนินการเรียบร้อยแล้วจะได้ เขตข้อมูลที่มีความสัมพันธ์เดียวกันจับกันเป็นกลุ่ม จากนั้นให้ทำการจัดบันทึกความสัมพันธ์ของเขตข้อมูลที่เป็นกลุ่มนั้นแยกเป็นตาราง ๆ และให้กำหนดหมายเลขกำกับไว้เพื่อให้อ้างอิงได้โดยง่าย โดยการดำเนินการเป็นไปดังตัวอย่างต่อไปนี้

ชื่อเพิ่มข้อมูล	ชื่อเขตข้อมูล	ความสัมพันธ์	เป็นคีย์หลัก
student	stdnt_c	stdnt_c	yes
student	stdnt_th_m	stdnt_c	no
student	stdnt_nglsh_m	stdnt_c	no
student	fcly_c	fcly_c	no
student	fcly_m	fcly_c	no
student	cmpr_c	cmpr_c	no
student	cmpr_m	cmpr_c	no
student	mjr_c	mjr_c	no
student	mjr_m	mjr_c	no
student	optn_m	mjr_c	no
subject	sbjct_c	sbjct_c	yes
subject	sbjct_th_m	sbjct_c	no
subject	sbjct_nglsh_m	sbjct_c	no
subject	sbjct_lb_crdt_a	sbjct_c	no
subject	sbjct_thry_crdt_a	sbjct_c	no
crse_nrll	aca_c	aca_c + sbjct_c + sctn_n	yes
crse_nrll	sbjct_c	sbjct_c	yes
crse_nrll	sctn_n	aca_c + sbjct_c + sctn_n	yes
crse_nrll	fcly_c	fcly_c	no
crse_nrll	cmpr_c	cmpr_c	no
register	aca_c	aca_c + sbjct_c + sctn_n	yes
register	sbjct_c	sbjct_c	yes
register	sctn_n	aca_c + sbjct_c + sctn_n	yes
register	stdnt_c	stdnt_c	yes
register	fcly_c	fcly_c	no
register	cmpr_c	cmpr_c	no
register	grd_c	aca_c + sbjct_c + sctn_n + stdnt_c	no

จากตัวอย่างตารางแสดงข้อมูลของเพิ่มข้อมูล และเขตข้อมูลที่ได้จากการจัดบันทึก ด้านบนนำมาจับกลุ่มตามความสัมพันธ์ แยกเป็นกลุ่ม พร้อม กำหนดหมายเลขกำกับได้ผลดังนี้

หมายเลขตาราง	ความสัมพันธ์
1	(stdnt_c)
2	(fcly_c)
3	(cmpr_c)
4	(mjr_c)
5	(sbjct_c)
6	(aca_c,sbjct_c,sctn_n)
7	(aca_c,sbjct_c,sctn_n,stdnt_c)

และเมื่อนำมาเขียนเป็น โครงสร้างแบบ schema ก็จะได้ผลลัพธ์เป็น

- 1 = (stdnt_c)
- 2 = (fcly_c)
- 3 = (cmpr_c)
- 4 = (mjr_c)
- 5 = (sbjct_c)
- 6 = (aca_c,sbjct_c,sctn_n)
- 7 = (aca_c,sbjct_c,sctn_n,stdnt_c)

3 กำหนดรายละเอียดตารางข้อมูล

ในขั้นตอนนี้ให้นำเขตข้อมูลต่าง ๆ พร้อมรายละเอียดของเขตข้อมูล เช่น สามารถมีค่าซ้ำได้หรือไม่ การที่ต้องมีค่าทุกกระเบียน ฯ ที่มีความสัมพันธ์เกี่ยวกับความสัมพันธ์ของตารางมากำหนดให้กับเขตข้อมูลในตารางข้อมูล โดยให้พิจารณาค่าเหล่านี้จากแฟ้มข้อมูลเดิมของเขตข้อมูล ตัวอย่างจากข้อ 2 เมื่อเพิ่มเขตข้อมูลและรายละเอียดแล้วจะได้ผลลัพธ์เขียนเป็นโครงสร้าง schema ดังนี้

- 1 = (stdnt_c, stdnt_th_m, stdnt_nglsh_m)
- 2 = (fclty_c, fclty_m)
- 3 = (cmpr_c, cmpr_m)
- 4 = (mjr_c, mjr_m, optn_m)
- 5 = (sbjct_c, sbjct_th_m, sbjct_nglsh_m, sbjct_lb_crtd_a, sbjct_thry_crtd_a)
- 6 = (aca_c, sbjct_c, sctn_n)
- 7 = (aca_c, sbjct_c, sctn_n, stdnt_c, grd_c)

4 กำหนดคีย์หลักของตารางข้อมูล

การกำหนดคีย์หลักของตารางข้อมูล ทำได้โดยกำหนดให้เขตข้อมูลหรือกลุ่มของเขตข้อมูลที่เป็นความสัมพันธ์ที่นำมาสร้างตารางเป็นคีย์หลักของตารางข้อมูลนั้น จากตัวอย่างในหัวข้อ 3 สามารถกำหนดคีย์หลักของตารางใหม่ได้เป็น (การกำหนดคีย์หลักใช้การขีดเส้นใต้เพื่อระบุว่าเป็นคีย์หลัก)

- 1 = (stdnt_c, stdnt_th_m, stdnt_nglsh_m)
- 2 = (fclty_c, fclty_m)
- 3 = (cmpr_c, cmpr_m)
- 4 = (mjr_c, mjr_m, optn_m)
- 5 = (sbjct_c, sbjct_th_m, sbjct_nglsh_m, sbjct_lb_crtd_a, sbjct_thry_crtd_a)
- 6 = (aca_c, sbjct_c, sctn_n)
- 7 = (aca_c, sbjct_c, sctn_n, stdnt_c, grd_c)

5 กำหนดชื่อตารางข้อมูล

การกำหนดชื่อตารางข้อมูลสามารถทำได้โดยให้เทียบคีย์หลักของตารางข้อมูลว่าตรงกับคีย์หลักของแฟ้มข้อมูลข้อมูลใดที่จัดบันทึกไว้ หากมีคีย์หลักตรงกันก็ให้นำชื่อของแฟ้มข้อมูลนั้นมากำหนดเป็นชื่อตารางข้อมูล หากไม่พบแฟ้มข้อมูลที่มีคีย์หลักตรงกับคีย์หลักของตารางข้อมูลก็ให้ผ่านไปยังตารางต่อไป จากตัวอย่างในข้อ 4 เมื่อเปรียบเทียบคีย์หลักของตารางข้อมูล กับแฟ้มข้อมูลที่จัดบันทึกไว้ ได้เป็นดังตารางต่อไปนี้

หมายเลขตารางข้อมูล	ชื่อแฟ้มข้อมูล	คีย์หลัก
1	student	stdnt_c
2	-----	fclty_c
3	-----	cmpr_c
4	-----	mjr_c
5	subject	sbjct_c
6	crse_nrl	aca_c + sbjct_c + sctn_n
7	register	aca_c + sbjct_c + sctn_n + stdnt_c

ดังนั้น จะได้ตารางข้อมูลเป็น

```

student = (stdnt_c, stdnt_th_m, stdnt_nglsh_m)
          2 = (fclty_c, fclty_m)
          3 = (cmpr_c, cmpr_m)
          4 = (mjr_c, mjr_m, optn_m)
subject = (sbjct_c, sbjct_th_m, sbjct_nglsh_m, sbjct_lb_crdt_a, sbjct_thry_crdt_a)
crse_nrl = (aca_c, sbjct_c, sctn_n)
register = (aca_c, sbjct_c, sctn_n, stdnt_c, grd_c)

```

6 กำหนดคีย์อ้างอิง

ขั้นตอนนี้เป็นขั้นตอนที่ทำให้ตารางข้อมูลพื้นฐานที่สร้างขึ้นมีความสมบูรณ์ขึ้น โดยเริ่มจากการเทียบคีย์หลักของแฟ้มข้อมูลแรกทีจัดบันทึกไว้ กับตารางข้อมูลทีสร้างขึ้นว่ามีตารางข้อมูลตารางใดทีมีคีย์หลักตรงกับกับแฟ้มข้อมูลทีกำลังพิจารณา หากไม่มีให้ผ่านไปทำงานกับแฟ้มข้อมูลอื่นต่อไป หากมีให้ทำการเปรียบเทียบเขตข้อมูลทีมีอยู่ในแฟ้มข้อมูลตั้งแต่เขตข้อมูลแรกว่าปรากฏในตารางข้อมูลทีกำลังพิจารณานี้หรือไม่ ถ้ามีให้ผ่านไปยังเขตข้อมูลถัดไป หากไม่มีให้ค้นหาว่าเขตข้อมูลนีปรากฏอยู่ในตารางข้อมูลใดหรือไม่ หากปรากฏว่าเขตข้อมูลนีปรากฏอยู่ในตารางข้อมูลใด ให้นำเขตข้อมูลหรือกลุ่มของเขตข้อมูลทีเป็นคีย์หลักของตารางข้อมูลนั้นมาเป็นเขตข้อมูลของตารางข้อมูลทีกำลังพิจารณา จากนั้นกำหนดให้เขตข้อมูลหรือกลุ่มข้อมูลเหล่านั้นเป็นคีย์นอกของตารางข้อมูลทีกำลังพิจารณา ทำเช่นนี้จนกระทั่งครบทุกเขตข้อมูลในแฟ้มข้อมูลจากนั้นก็ให้เลื่อนไปยังแฟ้มข้อมูลถัดไป จากตัวอย่างในข้อ 4 เมื่อเปรียบเทียบแล้วทำการสร้างคีย์นอกของตารางข้อมูล ผลที่ได้คือ

ชื่อตาราง	ชื่อแฟ้มข้อมูล	เขตข้อมูลทีขาดไป	คีย์ของเขตข้อมูลทีขาดไป
student	student	fclty_c fclty_m cmpr_c cmpr_m mjr_c mjr_m optn_m	fclty_c fclty_c cmpr_c cmpr_c mjr_c mjr_c mjr_c
subject	subject	-----	-----
crse_nrl	crse_nrl	fclty_c cmpr_c	fclty_c cmpr_c
register	register	-----	-----

ดังนั้นตารางสมบูรณ์ทีได้จากการเพิ่มคีย์นอก ก็คือ

```

student = (stdnt_c, stdnt_th_m, stdnt_nglsh_m, fclty_c, cmpr_c, mjr_c)
2       = (fclty_c, fclty_m)
3       = (cmpr_c, cmpr_m)
4       = (mjr_c, mjr_m, optn_m)
subject = (sbjct_c, sbjct_th_m, sbjct_nglsh_m, sbjct_lb_crdt_a, sbjct_thry_crdt_a)
crse_nrl = (aca_c, sbjct_c, sctn_n, fclty_c, cmpr_c)
register = (aca_c, sbjct_c, sctn_n, stdnt_c, grd_c)

```

7 ตรวจสอบตารางข้อมูลที่สร้างขึ้น

ขั้นตอนนี้เป็น ส่วนสุดท้ายที่หากมีเขตข้อมูลใดไม่มีตารางข้อมูลก็ให้เพิ่มเข้าไปในตาราง หรือในกรณีมีเขตข้อมูลใดที่ไม่ต้องการให้มีอยู่ในตารางก็ให้ลบออกเสีย อีกทั้งหากตารางข้อมูลใดที่ยังไม่มีชื่อตารางก็ให้กำหนดชื่อตารางที่ได้ตามความเหมาะสม ตัวอย่าง จากตัวอย่างข้อ 6 ที่ผ่านมา เมื่อไม่ต้องการเพิ่มหรือตัดเขตข้อมูลใดออกจากตารางก็ไม่ต้องทำ แต่อาจจะต้องการตั้งชื่อตารางข้อมูล เช่น ตารางที่ 2 ตั้งชื่อว่า faculty ตารางที่ 3 ตั้งชื่อว่า campus ตารางที่ 4 ตั้งชื่อว่า major ก็จะได้ตารางข้อมูลใหม่เป็น

student	=	(<u>stdnt_c</u> , stdnt_th_m, stdnt_nglsh_m, fcly_c, cmpr_c, mjr_c)
faculty	=	(<u>fcly_c</u> , fcly_m)
campus	=	(<u>cmpr_c</u> , cmpr_m)
major	=	(<u>mjr_c</u> , mjr_m, optn_m)
subject	=	(<u>sbjct_c</u> , sbjct_th_m, sbjct_nglsh_m, sbjct_lb_crtd_a, sbjct_thry_crtd_a)
crse_nrl	=	(<u>aca_c</u> , sbjct_c, sctn_n, fcly_c, cmpr_c)
register	=	(<u>aca_c</u> , sbjct_c, sctn_n, stdnt_c, grd_c)

รายการอ้างอิง

ภาษาไทย

สำนักการศึกษาระบบสารสนเทศ สถาบันบัณฑิตพัฒนบริหารศาสตร์. เอกสารประกอบ

โครงการฝึกอบรมเกี่ยวกับการวิเคราะห์ข้อมูล. ม.ป.ท., 2534

ทักษิณา สนวนานนท์,ศาสตราจารย์. พจนานุกรมศัพท์คอมพิวเตอร์. พิมพ์ครั้งที่ 4. กรุงเทพฯ: ไสเทค
พรินติ้ง จำกัด, 2536

ภาษาอังกฤษ

DAVID S. BOWERS. From data to database. 2nd ed. London Chapman & Hall,1993

ELMASRI RAMEZ , and NAVATHE SHAMKANT B. Fundamentals of Database systems.

Newyork The Beniamin/cummings publishing company,1989

Eric Armstrong, Steven Bobrowski, Cynthia Closkey, Brian Linden.Oracle 7 Sever Administrator's
Guide. Oracle Corporation Redwood City, CA USA:Oracle Corporation, 1993

EVEREST GORDON C. Database management objctives.system functions & administration.

Singapore Chong Moh Offset Printing Pte Ltd, 1986

FLEMING CANDACE C. and HALLE BARBARA V. Handbook of Relational Database Design.

Addison-Wesley Publishing Company Inc, 1989

JENNINGS ROGER. Database Developer's Guide with Visual Basic 3. Indianapolis SAMS, 1994

JOHN J. DONOVAN. System Programing. Newyork McGrow-Hill inc, 1989

KORTH H.F., and SILBERSCHATZ A. Database system concepts. 2nd ed.

Newyork McGraw-Hill inc, 1991.

Microsoft Corporation. Professional Features Book 2. United State of America, 1993

ORACLE.SQL Language Reference Manual Version 6.0. USA:Oracle Corporation, 1990

Progress Software Corporation.PROGRESS On-line Reference., Bedford, Messachusetts

USA:Progress Software Corporation,1995

VAN DER LANS RICK F. The SQL standard. Prentice Hall International(UK) Ltd, 1989

070

ประวัติผู้เขียน

นายสันต์ กันอ่ำ เกิดเมื่อวันที่ 29 ตุลาคม พ.ศ. 2511 ที่กรุงเทพฯ สำเร็จการศึกษาปริญญาตรี
วิทยาศาสตร์บัณฑิต สาขาศาสตร์คอมพิวเตอร์ ภาควิชาศาสตร์คอมพิวเตอร์ คณะวิทยาศาสตร์
มหาวิทยาลัยรามคำแหง ในปีการศึกษา 2532 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต ที่
จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2537 ปัจจุบันรับราชการที่ สถาบันเทคโนโลยีราชมงคล ศูนย์กลาง
สถาบันเทคโนโลยีราชมงคล ตำบลคลองหก อำเภอธัญบุรี จังหวัดปทุมธานี

