บทที่ 4

ระบบฐานข้อมูลของเครื่องไอบีเอ็มเอเอส/400

ระบบฐานข้อมูลของเครื่องเอเอส/400

เครื่องไอบีเอ็มเอเอส/400 ได้ถูกออกแบบระบบจัดการฐานข้อมูลให้เป็นส่วนหนึ่งของระบบปฏิบัติการ(Operating System) ทั้งนี้ด้วยเหตุผลที่ว่า เครื่องไอบีเอ็มเอเอส/400 ถูกสร้างขึ้นเพื่อวัตถุประสงค์ในการประมวลผลข้อมูลทางธุรกิจ ดังนั้นการออกแบบให้ระบบจัดการฐานข้อมูลอยู่ในระดับเดียวกับระบบปฏิบัติการจะทำให้ประสิทธิภาพของการประมวลผลข้อมูลที่ดีกว่าแยกออกมาจากกัน ระบบฐานข้อมูลของเครื่องไอบีเอ็มเอเอส/400เป็นแบบฐานข้อมูลเชิงสัมพันธ์และจะมีการใช้งานแฟ้มฐานข้อมูลอยู่สองแบบ ดังต่อไปนี้

1. แฟ้มกายภาพ (Physical File)

สำหรับแฟ้มกายภาพจะมีการกำหนดลักษณะโครงสร้างที่แท้จริงของแฟ้มข้อมูลโดยระเบียนจะมีความยาวจำกัดตายตัว และใช้เก็บข้อมูลจริงของระบบฐานข้อมูล อีกทั้งยังสามารถกำหนดกุญแจที่เป็นดัชนีเพื่อกำหนดการจัดเรียงข้อมูลใหม่แทนที่จะต้องเรียงอยู่ตามลำดับการเกิดก่อนหลังซึ่งจะใช้หมายเลขระเบียนสัมพัทธ์(Relative Record Number :RRN) ตามลำดับจากน้อยไปมาก

2. แฟ้มตรรกะหรือมุมมอง (Logical File or View)

สำหรับแฟ้มตรรกะจะเป็นการกำหนดมุมมองและดัชนีของฐานข้อมูลโดยที่ไม่มีการเก็บข้อมูลจริงไว้ แต่จะเป็นตัวกำหนดว่าระเบียนมีรูปแบบอย่างไร โดยอาจจะมาจากแฟ้มกายภาพหนึ่งแฟ้มหรือมากกว่าได้ อีกทั้งยังมีอีกหลายอย่างที่สามารถกำหนดได้ในแฟ้มตรรกะ

2.1 สามารถเปลี่ยนคุณสมบัติบางอย่างของฟิลด์ให้แตกต่างจากแฟ้มกายภาพ

2.2 สามารถกำหนดการจัดเรียงของระเบียนได้ใหม่

2.3 สามารถป้องกันการอ่านหรือแก้ไขของข้อมูลบางฟิลด์หรือหลายฟิลด์ ในแฟ้มกายภาพ

2.4 สามารถสร้างฟิลด์ใหม่ขึ้นมาจากฟิลด์ที่มีอยู่ในแฟ้มกายภาพ

2.5 สามารถรวมแฟ้มกายภาพมากกว่าหนึ่งแฟ้มให้เป็นมุมมองเดียวได้

การสร้างแฟ้มข้อมูลกายภาพ

ในการสร้างแฟ้มข้อมูลแบบกายภาพจะต้องมีการบรรยายลักษณะต่าง ๆ ในแฟ้มโครงร่าง(Source physical file) ที่เรียกว่า "แฟ้มบรรยายข้อมูล" (Data Describe Specification :DDS) จากนั้นจึงทำการแปล(compile)ให้เป็นแฟ้มข้อมูลที่ระบบรู้จัก(Object code)อีกทอดหนึ่ง ซึ่งการกำหนดโครงสร้างจะประกอบไปด้วยส่วนสำคัญ 4 ส่วน ดังนี้

```
      ********** Beginning of data ********************************************
FMT A*  .....A*. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0000.01       *                                    (1)
FMT PF  .....A.........T.Name++++++RLen++TDpB.....Functions++++++++++++++++++
0001.00       A        (2) R I412
0002.00       A            I2ITEM        25A         COLHDG('Item Number
0003.00       A            I2WRHS        12A         COLHDG('Weare House
0004.00       A            I2LOC         10A         COLHDG('Location
0005.00       A        (3) I2BIN          8A         COLHDG('Bin
0006.00       A            I2AISL        8A         COLHDG('Aisel
0007.00       A            I2QTOH        7P 0       COLHDG('Quantity On-Hand
0008.00       A            I2COST        9P 2       COLHDG('Unit Cost
0009.00       A            I2PRIC        9P 2       COLHDG('Unit Price
0010.00       A        (4) K I2WRHS
0011.00       A            K I2ITEM
      ********** End of data **********************************************
```

รูปที่ 4.1 ตัวอย่าง DDS สำหรับแฟ้มกายภาพ

1. ระดับแฟ้มข้อมูล(File level)

ในระดับแฟ้มข้อมูลจะเป็นการกำหนดคุณสมบัติต่างๆในระดับแฟ้มข้อมูล เช่น การกำหนด UNIQUE เพื่อป้องกันไม่ให้เกิดการซ้ำกันของกุญแจเกิดขึ้นได้

2. ระดับระเบียน(Record level)

ในระดับระเบียนจะเป็นการบอกให้ระบบรับรู้ว่าแฟ้มข้อมูลที่สร้างขึ้นมานี้มีระเบียนชื่อว่าอะไร เพราะในระบบของเอเอส/400แต่ละแฟ้มสามารถจะมีระเบียนได้หลายตัว การกำหนดในDDSให้ใส่ตัวอักษร "R" ที่ตำแหน่งที่ 17 เพื่อเป็นการบ่งชี้ว่าที่บรรทัดนี้จะเป็นการกำหนดระดับระเบียน โดยชื่อระเบียนจะกำหนดที่ตำแหน่ง19-28 และข้อกำหนดอื่น ๆ (Keywords)ที่เกี่ยวข้องกับการกำหนดระดับระเบียนจะใส่ไว้ที่ ตำแหน่งที่ 45 ถึง 80

3. ระดับฟิลด์(Field level)

ในระดับฟิลด์จะเป็นการกำหนดชื่อฟิลด์ ชนิดของฟิลด์ ขนาดของฟิลด์ และคุณสมบัติอื่น ๆ อีกหลายอย่าง เช่น กำหนดโดเมน กำหนดรูปแบบ เป็นต้น การกำหนดในDDSจะใส่ชื่อฟิลด์ที่ตำแหน่งที่19-28 ชนิดของฟิลด์จะกำหนดที่ตำแหน่งที่35 โดยสามารถใส่ค่าต่าง ๆ ที่ระบุชนิดของข้อมูลดังนี้

"A" = Character
"P" = Packed decimal
"S" = Zoned decimal
"B" = Binary
"F" = Floating point

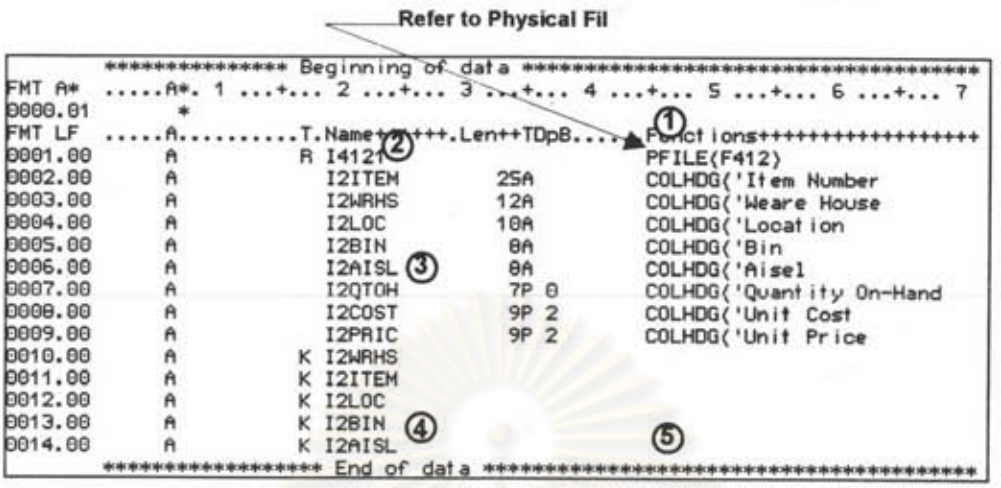
ในกรณีที่ไม่ได้ระบุชนิดของข้อมูลที่ตำแหน่งที่35 ระบบจะระบุชนิดของข้อมูลโดยใช้จุดทศนิยมที่จะใส่อยู่ที่ตำแหน่งที่ 36 ถึง 37 เป็นตัวบ่งชี้ ถ้าไม่ได้ใส่จำนวนทศนิยมแสดงว่าเป็นข้อมูลชนิดตัวอักษร แต่ถ้าใส่จำนวนทศนิยมแสดงว่าเป็นตัวเลข(Packed decimal) การกำหนดขนาดของฟิลด์จะใส่ตัวเลขแสดงขนาดของฟิลด์ที่ตำแหน่งที่30-34 ส่วน keywords อื่น ๆ ที่เกี่ยวข้องกับการกำหนดระดับฟิลด์จะใส่ไว้ที่ตำแหน่งที่ 45-80

4. ระดับกุญแจ(Key field level)

ในระดับกุญแจจะเป็นส่วนที่จะกำหนดหรือไม่ก็ได้ โดยถ้ากำหนดไว้จะทำให้การจัดเรียงของระเบียนข้อมูล เรียงลำดับตามข้อกำหนดของกุญแจแทนการจัดเรียงตามหมายเลขระเบียนสัมพัทธ์ การกำหนดในDDSจะทำได้โดยใส่ตัวอักษร "K" ที่ตำแหน่งที่17 และใส่ชื่อฟิลด์ที่จะใช้เป็นดัชนีในตำแหน่งที่19ถึง28 อีกทั้งยังสามารถกำหนดวิธีการเรียงลำดับได้ว่าเป็นแบบเรียงจากน้อยไปมากหรือเรียงจากมากไปน้อยโดยการกำหนดkeywordsที่ตำแหน่งที่45ถึง80

การสร้างแฟ้มมุมมองและดัชนี

ในการสร้างแฟ้มข้อมูลแบบตรรกะจะต้องมีการบรรยายลักษณะต่าง ๆ ในแฟ้มโครงร่าง(Source physical file) จากนั้นจึงทำการแปล(compile)ให้เป็นแฟ้มข้อมูลอีกทอดหนึ่ง เหมือนกับการสร้างแฟ้มกายภาพเช่นกัน แต่การกำหนดโครงสร้างจะประกอบไปด้วยส่วนสำคัญ 5 ส่วน ดังนี้




```
     ********** Beginning of data **************************************
FMT A*  .....A*. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0000.01      *
FMT LF  .....A..........T.Name++++++.Len++TDpB.....Functions++++++++++++++++++
0001.00      A          R I4121                   PFILE(F412)
0002.00      A            I2ITEM        2SA       COLHDG('Item Number
0003.00      A            I2WRHS        12A       COLHDG('Weare House
0004.00      A            I2LOC         10A       COLHDG('Location
0005.00      A            I2BIN          8A       COLHDG('Bin
0006.00      A            I2AISL         8A       COLHDG('Aisel
0007.00      A            I2QTOH        7P 0      COLHDG('Quantity On-Hand
0008.00      A            I2COST        9P 2      COLHDG('Unit Cost
0009.00      A            I2PRIC        9P 2      COLHDG('Unit Price
0010.00      A          K I2WRHS
0011.00      A          K I2ITEM
0012.00      A          K I2LOC
0013.00      A          K I2BIN
0014.00      A          K I2AISL
     ********** End of data ****************************************
```

รูปที่ 4.2 ตัวอย่าง DDS สำหรับแฟ้มตรรกะ

1. ระดับแฟ้มข้อมูล(File level)

จะเป็นการกำหนดคุณสมบัติต่างๆในระดับแฟ้มข้อมูล เช่น การกำหนด UNIQUE เพื่อป้องกันไม่ให้เกิดการซ้ำกันของกุญแจเกิดขึ้นได้ เหมือนกับการกำหนดในแฟ้มกายภาพ

2. ระดับระเบียน(Record format level)

จะเป็นการบอกให้ระบบรับรู้ว่าแฟ้มข้อมูลที่สร้างขึ้นมานี้มีระเบียนชื่อว่าอะไร เพราะในระบบของเอเอส/400สามารถมีระเบียนได้หลายตัวในหนึ่งแฟ้ม และเป็นการระบุชื่อแฟ้มกายภาพที่จะถูกใช้เพราะว่าแฟ้มตรรกะจะไม่มีการเก็บข้อมูลอยู่จริงแต่จะเก็บดัชนีแทน การกำหนดในDDSจะใส่ตัวอักษร 'R' ที่ตำแหน่งที่ 17 เพื่อเป็นการบ่งชี้ว่าที่บรรทัดนี้จะเป็นการกำหนดระดับระเบียน โดยชื่อระเบียนจะกำหนดที่ตำแหน่งที่ 19 ถึง 28 ส่วนการระบุว่าจะใช้แฟ้มกายภาพชื่ออะไรสามารถกำหนดได้โดยใส่ PFILE(ชื่อแฟ้มกายภาพ) ที่ตำแหน่งที่ 45 ถึง 80 ซึ่งในวงเล็บจะใส่ชื่อแฟ้มกายภาพ

3. ระดับฟิลด์(Field level)

จะเป็นการกำหนดชื่อฟิลด์ ชนิดของฟิลด์ ขนาดของฟิลด์ และคุณสมบัติอื่น ๆ อีกหลายอย่าง เช่น กำหนดโดเมน กำหนดรูปแบบ เป็นต้น การกำหนดใน DDS จะทำได้เหมือนกับการกำหนดของแฟ้มกายภาพทุกอย่างแต่ฟิลด์ที่สร้างขึ้นใหม่จะต้องอ้างถึงฟิลด์ที่ได้สร้างไว้แล้วเท่านั้น และความสะดวกอีกประการหนึ่งที่มีก็คือเราไม่จำเป็นต้องกำหนดชนิดและขนาดของข้อมูลเองเพียงแต่อ้างถึงแฟ้มกายภาพที่ระบุไว้ในระดับแฟ้มข้อมูลที่ตำแหน่งที่ 45-80 ด้วย REF(...) โดยในวงเล็บใส่ชื่อแฟ้มกายภาพ และที่ระดับฟิลด์จะใส่ตัวอักษร 'R' ที่ตำแหน่งที่ 29

4. ระดับกุญแจ(Key field level)

จะเป็นส่วนที่จะกำหนดหรือไม่ก็ได้ โดยถ้ากำหนดไว้จะทำให้การจัดเรียงของระเบียนข้อมูล เรียงลำดับตามชุดของกุญแจแทนการจัดเรียงตามหมายเลขระเบียนสัมพัทธ์ กำหนดใน DDS จะทำได้เหมือนกับการกำหนดของแฟ้มกายภาพทุกอย่าง

5. ระดับการเลือกข้อมูล(Select/Omit field level)

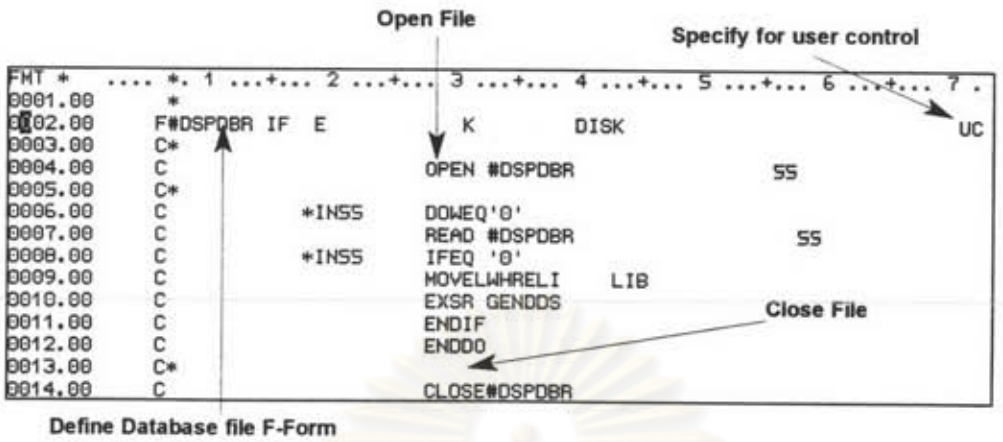
จะเป็นส่วนที่จะกำหนดหรือไม่ก็ได้ โดยถ้ากำหนดไว้จะเป็นการกรองเฉพาะข้อมูลที่สอดคล้องกับเงื่อนไขที่กำหนดด้วยคำสั่ง select "S" และomit "O" ซึ่งจะถูกกำหนดไว้ท้ายสุดของ DDS โดยใส่ "S" หรือ "O" ไว้ที่ตำแหน่งที่17 และที่ตำแหน่งที่19-28 ใส่ชื่อฟิลด์ที่จะใช้กำหนดเงื่อนไขในการเลือก ส่วนที่ตำแหน่งที่45-80 จะใส่เงื่อนไขในการเลือกข้อมูลสำหรับแต่ละฟิลด์ เช่น CMP(EQ 'N') หมายความว่าให้เลือกเฉพาะที่เท่ากับ 'N' เท่านั้น

คำสั่งในการกระทำต่อรายการข้อมูลของฐานข้อมูล

คำสั่งที่ใช้ในการปฏิบัติการต่อแฟ้มข้อมูลของภาษาRPG400 นั้นก็เหมือนกับภาษาอื่น ๆ โดยสามารถแบ่งออกเป็นกลุ่ม ๆ ดังนี้

1. คำสั่งในการเปิดและปิดแฟ้ม (OPEN, CLOSE)

ปกติแล้วโปรแกรมRPG400จะทำการเปิดแฟ้มข้อมูลเองอัตโนมัติเมื่อเราเริ่มให้โปรแกรมทำงานและจะปิดแฟ้มข้อมูลเองเมื่อสิ้นสุดการทำงานของโปรแกรม แต่เราสามารถที่จะควบคุมการเปิดและปิดของแฟ้มข้อมูลได้เองเช่นกัน โดยเราต้องระบุให้แฟ้มข้อมูลนั้น ๆ เป็นแฟ้มที่ต้องการควบคุมการเปิดปิดด้วยผู้ใช้เอง ซึ่งวิธีการก็คือใส่ตัวอักษร "UC" (User Control) บน F-form ในตำแหน่งที่71-72 ดังแสดงในรูปที่ 4.3

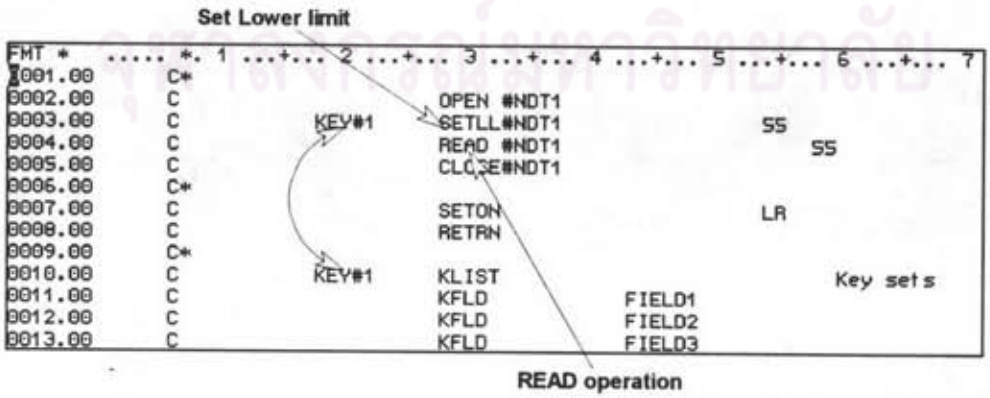


รูปที่ 4.3 ตัวอย่างการใช้คำสั่งเปิดและปิดแฟ้มข้อมูล

รูปแบบของคำสั่งเปิดและปิดแฟ้มข้อมูลคือใส่คำสั่ง OPEN หรือ CLOSE ที่ตำแหน่งที่28-32 ใน C-form และใส่ชื่อแฟ้มข้อมูลที่ได้กำหนดให้ควบคุมด้วยผู้ใช้แล้วที่ตำแหน่งที่33-42 และที่ตำแหน่งที่56-57เราสามารถใส่เลขบ่งชี้(result indicator) เพื่อใช้ตรวจสอบผลของการทำงาน ถ้าเลขบ่งชี้เท่ากับ 1(on)แสดงว่าผลการทำงานของคำสั่งเปิดหรือปิดแฟ้มข้อมูลนั้นล้มเหลว แต่ถ้าเลขบ่งชี้เท่ากับ 0(off)แสดงว่าผลการทำงานเป็นปกติ

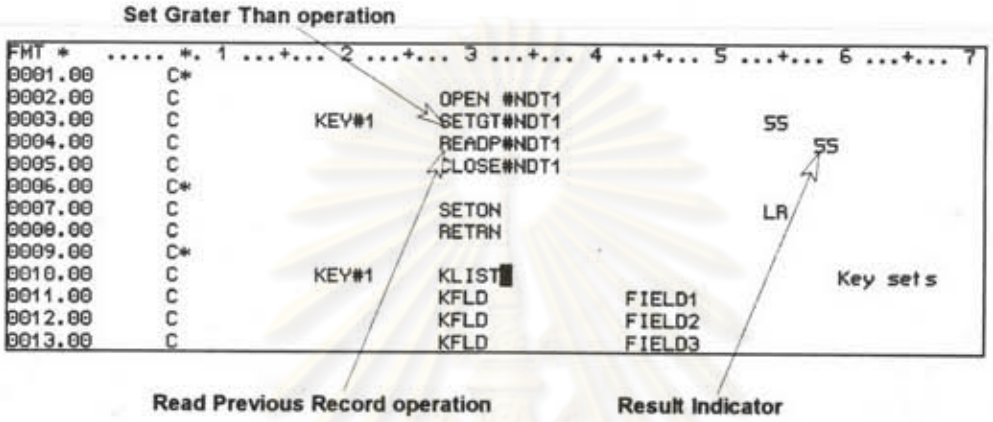
2. คำสั่งในการชี้ตำแหน่งข้อมูล (SETLL, SETGT)

คำสั่ง SETLL(Set Lower Limit) จะทำการชี้ตำแหน่งแรกของแฟ้มข้อมูลที่มีค่ามากกว่าหรือเท่ากับค่าที่กำหนด(Search argument) ดังแสดงในรูปที่ 4.5 ซึ่งค่าที่กำหนดนี้อาจเป็นชุดของกุญแจ(Key field list) หรือ หมายเลขระเบียนสัมพัทธ์ นอกจากค่าที่สามารถกำหนดได้ดังกล่าวระบบยังได้เตรียมค่า '*LOVAL' ไว้สำหรับชี้ตำแหน่งต้นแฟ้มข้อมูลเพื่อที่จะให้คำสั่งอ่านข้อมูลอย่างเรียงลำดับตั้งแต่ต้นแฟ้มเป็นต้นไป



รูปที่ 4.4 ตัวอย่างการใช้คำสั่ง SETLL และ READ

คำสั่ง SETGT(Set Greater Than) จะทำการชี้ตำแหน่งแรกของแฟ้มข้อมูลที่มีค่าน้อยกว่าหรือเท่ากับค่าที่กำหนด(Search argument) ดังแสดงใน รูปที่ 4.7 ซึ่งค่าที่กำหนดนี้อาจเป็นชุดของกุญแจ(Key field list) หรือหมายเลขระเบียนสัมพัทธ์ นอกจากค่าที่สามารถกำหนดได้ดังกล่าวระบบยังได้เตรียมค่า '*HIVAL' ไว้สำหรับชี้ตำแหน่งสุดท้ายแฟ้มข้อมูลเพื่อการอ่านข้อมูลแล้วย้อนกลับด้วยคำสั่ง READP ซึ่งจะได้กล่าวในลำดับต่อไป



รูปที่ 4.5 ตัวอย่างการใช้คำสั่ง SETGT และ READP

3. คำสั่งในการอ่านข้อมูล (READ, READE, READP)

คำสั่งอ่านทั้งหมดไม่ว่าจะเป็น READ READE หรือ READP เป็นการอ่านข้อมูลที่ตำแหน่งที่ถูกชี้อยู่ในขณะนั้นอย่างเรียงลำดับ(Sequential)ไปเรื่อย ๆ ทีละระเบียน และหลังจากอ่านสำเร็จแต่ละครั้งระบบจะทำการเลื่อนไปชี้รายการข้อมูลถัดไปโดยอัตโนมัติ แต่ถ้าอ่านไม่สำเร็จผู้ใช้จะต้องใช้คำสั่งในการชี้ตำแหน่งเองอีกครั้งเพื่อกำหนดตำแหน่งที่แน่นอนสำหรับการอ่านในครั้งต่อไป ในการอ่านแต่ละครั้งเราจะรู้จุดสิ้นสุดของแฟ้มได้ด้วยเลขบ่งชี้ที่ระบุอยู่ในตำแหน่งที่58-59 โดยถ้าเลขบ่งชี้มีสภาวะเป็น 'เปิด' หมายถึงสิ้นสุดแฟ้ม(EOF) สำหรับการอ่านด้วยคำสั่ง READ หรือหมายถึงอ่านไม่พบระเบียนที่มีค่าตรงกับ key list ที่กำหนด สำหรับการอ่านด้วยคำสั่ง READE หรือหมายถึงได้อ่านจนถึงตำแหน่งบนสุดของแฟ้มแล้ว (BOF) สำหรับการอ่านด้วยคำสั่ง READP

ตัวอย่างการใช้คำสั่ง READ ดังแสดงในตัวอย่างรูปที่4.9 จะเป็นการอ่านอย่างเรียงลำดับ(Sequential)ไปเรื่อยๆทีละระเบียนจากตำแหน่งที่ถูกชี้อยู่ในขณะนั้นเรื่อยไป โดยไม่มีการกำหนดเงื่อนไขในการอ่านใด ๆ ทั้งสิ้น จนกระทั่งสิ้นสุดแฟ้มข้อมูล

การใช้คำสั่ง READE(read equal key) ดังแสดงในตัวอย่าง รูปที่ 4.10 จะเป็นการอ่านอย่างเรียงลำดับ(Sequential)ไปเรื่อย ๆ ทีละระเบียน จากตำแหน่งที่ถูกชี้อยู่ในขณะ

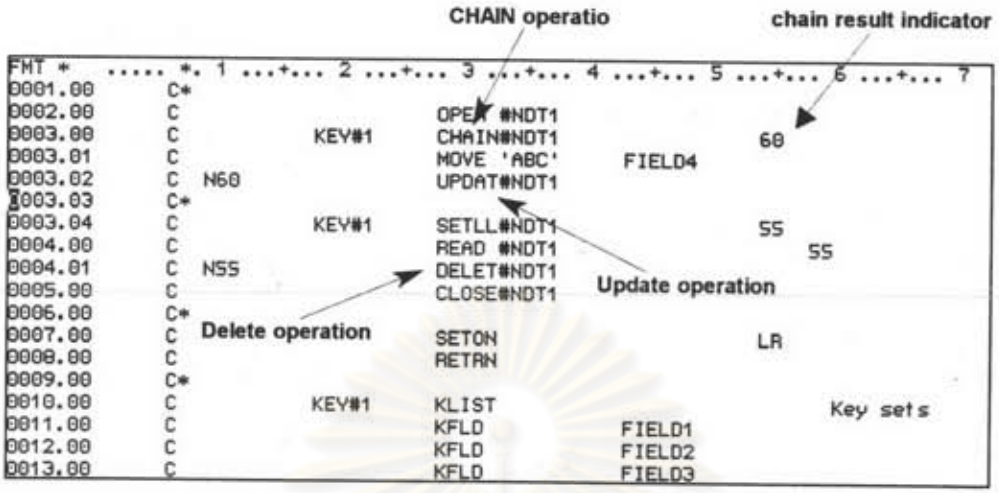
นั้นเรื่อยไป แต่ก่อนที่จะเริ่มอ่านจะต้องใช้คำสั่งในการชี้ตำแหน่ง(Set Lower Limit :SETLL) เพื่อชี้ไปยังตำแหน่งที่ตรงกับ key list เพราะการอ่านด้วยคำสั่งนี้จะเป็นการอ่านแบบมีเงื่อนไข ซึ่งเงื่อนไขก็คือจะอ่านเฉพาะข้อมูลที่มีค่าในฟิลด์ที่ตรงการ key list ที่กำหนดอยู่หน้าคำสั่ง READP ถ้ามีฟิลด์ใดฟิลด์หนึ่งที่กำหนดอยู่ใน key list ไม่ตรงกันจะทำให้เลขบ่งชี้มีสภาวะเป็น "เปิด"

การใช้คำสั่ง READP(read prior record) ดังแสดงในตัวอย่าง รูปที่ 4.11 จะเป็นการอ่านอย่างเรียงลำดับ(Sequential) จากตำแหน่งที่ถูกชี้อยู่ในขณะถอยหลังไปเรื่อย ๆ ทีละระเบียน แต่ก่อนที่จะเริ่มอ่านจะต้องใช้คำสั่งในการชี้ตำแหน่ง (Set Greater Than :SETGT) เพื่อชี้ไปยังตำแหน่งที่ต้องการ หลังจากการอ่านแต่ละครั้งระบบจะทำการเลื่อนไปชี้อยู่ที่ตำแหน่งก่อนหน้าระเบียนที่อ่านหนึ่งตำแหน่งเสมอ และเมื่ออ่านจนกระทั่งถึงบนสุดของแฟ้มแล้วจะทำให้เลขบ่งชี้มีสภาวะเป็น "เปิด"

4. คำสั่งในการค้นหาข้อมูล (CHAIN)

คำสั่ง CHAIN เป็นคำสั่งที่ใช้ในการค้นหาและเรียกใช้ข้อมูล ซึ่งสามารถใช้ชุดของกุญแจและเลขระเบียนสัมพัทธ์เป็นตัวเรียกค้น โดยที่การใช้เลขระเบียนสัมพัทธ์เพียงระบุหมายเลขของระเบียนที่ต้องการเรียกใช้ซึ่งต้องเป็นเลขจำนวนเต็มเท่านั้น ส่วนการใช้ชุดของกุญแจในการเรียกค้นจะต้องมีการกำหนดชุดของกุญแจขึ้นมาก่อนโดยใช้คำสั่ง KLIST เพื่อกำหนดชื่อของกุญแจแล้วใช้คำสั่ง KFLD เพื่อสร้างการเรียงตัวของตัวแปรต่าง ๆ ที่จะประกอบกันเป็นกุญแจ แต่กฎเกณฑ์ที่สำคัญในการสร้างชุดของกุญแจนี้ก็คือตัวแปรที่ใช้ประกอบเป็นกุญแจและฟิลด์ที่เป็นกุญแจของแฟ้มข้อมูลที่ต้องการใช้งาน จะต้องเป็นข้อมูลชนิดเดียวกันและขนาดเดียวกัน อีกทั้งการเรียงตัวของตัวแปรของชุดกุญแจก็ต้องเหมือนกับการเรียงตัวของกุญแจที่ปรากฏในแฟ้มข้อมูล เพียงแต่จำนวนของตัวแปรที่ใช้ประกอบเป็นกุญแจจะต้องน้อยกว่าหรือเท่ากับจำนวนฟิลด์ของกุญแจที่ปรากฏในแฟ้มข้อมูล

การใช้งานโดยการระบุชื่อกุญแจที่จะใช้ในการค้นข้อมูลไว้ที่ตำแหน่งที่19-27 ซึ่งได้มีการเตรียมข้อมูลต่าง ๆ ในตัวแปรที่ใช้ประกอบเป็นกุญแจนั้น ๆ เรียบร้อยแล้ว และใส่คำสั่ง CHAIN ไว้ที่ตำแหน่ง 28-32 พร้อมทั้งระบุชื่อแฟ้มข้อมูลหรือชื่อระเบียนของแฟ้มข้อมูลนั้นๆไว้ที่ตำแหน่ง 33-42 และที่ขาดไม่ได้อีกอย่างก็คือต้องระบุหมายเลขบ่งชี้ที่ตำแหน่งที่ 54-55 เพื่อเตรียมไว้ใช้ตรวจสอบผลการทำงานของคำสั่ง ถ้าเลขบ่งชี้มีสภาวะเป็น "เปิด" แสดงว่าการสืบค้นไม่พบข้อมูลตามเงื่อนไขที่ระบุอยู่ในชุดของกุญแจ แต่ถ้ามีสภาวะเป็น "ปิด" แสดงว่าการสืบค้นประสพความสำเร็จ ดังแสดงตัวอย่างใน รูปที่ 4.12




```
FMT *   ..... *. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0001.00     C*
0002.00     C                       OPEN #NDT1
0003.00     C           KEY#1       CHAIN#NDT1                    60
0003.01     C                       MOVE 'ABC'      FIELD4
0003.02     C  N60                  UPDAT#NDT1
0003.03     C*
0003.04     C           KEY#1       SETLL#NDT1                    55
0004.00     C                       READ #NDT1                        55
0004.01     C  N55                  DELET#NDT1
0005.00     C                       CLOSE#NDT1
0006.00     C*
0007.00     C                       SETON                         LR
0008.00     C                       RETRN
0009.00     C*
0010.00     C           KEY#1       KLIST
0011.00     C                       KFLD            FIELD1
0012.00     C                       KFLD            FIELD2
0013.00     C                       KFLD            FIELD3
```

รูปที่ 4.6 ตัวอย่างการใช้คำสั่ง CHAIN UPDAT และ DELET

5. คำสั่งในการปรับปรุงข้อมูล (UPDAT)

คำสั่ง UPDAT จะเป็นการปรับปรุงระเบียนที่ถูกใช้อยู่ในขณะนั้นลงแฟ้มข้อมูล ซึ่งแฟ้มข้อมูลที่จะใช้กับคำสั่งนี้ได้จะต้องเป็นแฟ้มที่ได้ประกาศไว้แล้วว่าเป็นแฟ้มที่อนุญาติให้ปรับปรุงรายการข้อมูลได้ โดยใส่ตัวอักษร "U" ไว้ที่ตำแหน่ง 15 ของการประกาศแฟ้มข้อมูลใน F-form การใช้งานก็เพียงใส่คำสั่ง UPDAT ไว้ที่ตำแหน่ง 29-32 พร้อมทั้งระบุชื่อแฟ้มข้อมูลหรือชื่อระเบียนของแฟ้มข้อมูลไว้ที่ตำแหน่ง 33-42 อีกทั้งยังสามารถกำหนดเลขบ่งชี้ไว้ที่ตำแหน่ง56-57เพื่อใช้ในการตรวจสอบว่าผลการปรับปรุงรายการเสร็จสมบูรณ์หรือไม่ ถ้าเลขบ่งชี้ "เปิด" แสดงว่าการทำงานไม่สำเร็จ

6. คำสั่งในการลบข้อมูล (DELET)

คำสั่ง DELET จะเป็นการลบระเบียนที่ถูกใช้อยู่ในขณะนั้นออกจากแฟ้มข้อมูล ซึ่งแฟ้มข้อมูลที่จะใช้กับคำสั่งนี้ได้จะต้องเป็นแฟ้มที่ได้ประกาศไว้แล้วว่าเป็นแฟ้มที่อนุญาติให้ปรับปรุงรายการข้อมูลได้ โดยใส่ตัวอักษร "U" ไว้ที่ตำแหน่ง15ของการประกาศแฟ้มข้อมูลใน F-form การใช้งานก็เพียงใส่คำสั่ง UPDAT ไว้ที่ตำแหน่ง 29-32 พร้อมทั้งระบุชื่อแฟ้มข้อมูลหรือชื่อระเบียนของแฟ้มข้อมูลไว้ที่ตำแหน่ง 33-42 อีกทั้งยังสามารถกำหนดเลขบ่งชี้ไว้ที่ตำแหน่ง 56-57 เพื่อใช้ในการตรวจสอบว่าผลการลบข้อมูลเสร็จสมบูรณ์หรือไม่ ถ้าเลขบ่งชี้ "ปิด" แสดงว่าการทำงานสำเร็จ

7. คำสั่งในการเพิ่มข้อมูล (WRITE)

คำสั่ง WRITE จะเป็นการเพิ่มระเบียนใหม่เข้าไปในแฟ้มข้อมูล ซึ่งแฟ้มข้อมูลที่จะใช้กับคำสั่งนี้ได้จะต้องเป็นแฟ้มที่ได้ประกาศไว้แล้วว่าเป็นแฟ้มที่อนุญาติให้เพิ่มรายการ

ข้อมูลได้ โดยใส่ตัวอักษร "O" หรือ "I" หรือ "U" ไว้ที่ตำแหน่ง 15 ของการประกาศแฟ้มข้อมูล ใน F-form ในกรณีที่ประกาศโดยใช้ "I" หรือ "U" จะต้องใส่ตัวอักษร "A" ไว้ที่ตำแหน่งที่ 66 เพื่อบังคับให้สามารถใช้คำสั่ง WRITE ได้ในขณะที่เปิดแฟ้มข้อมูลแบบอ่านอย่างเดียว (I) หรือ อ่านและแก้ไขได้ (U)

การใช้งานก็เพียงใส่คำสั่ง WRITE ไว้ที่ตำแหน่ง 29-32 พร้อมทั้งระบุชื่อแฟ้มข้อมูลหรือชื่อระเบียนของแฟ้มข้อมูลไว้ที่ตำแหน่ง33-42 อีกทั้งยังสามารถกำหนดเลขบ่งชี้ไว้ที่ตำแหน่ง 56-57 เพื่อใช้ในการตรวจสอบว่าผลการลบข้อมูลเสร็จสมบูรณ์หรือไม่ ถ้าเลขบ่งชี้ "เปิด" แสดงว่าการทำงานไม่สำเร็จ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย