



บทที่ 5

การออกแบบและพัฒนา การสร้าง, การลบ คัดนี้ฐานข้อมูล

จากบทที่ 4 หลังจากสร้างโครงสร้างฐานข้อมูลแล้ว เราจะทราบว่าโครงสร้างข้อมูลเป็นอย่างไร และสามารถที่จะสอบถามข้อมูลแบบใดต่อกับแฟ้มข้อมูลอักขระได้ โดยที่การประมวลผลข้อมูลจะเป็นแบบเรียงลำดับ แต่ในบางครั้งเราอาจจะต้องการค้นหาข้อมูลให้รวดเร็วขึ้น เราจึงจำเป็นต้องนำข้อมูลในรูปแฟ้มข้อมูลอักขระมาสร้างเป็นคดันี้ฐานข้อมูลขึ้น ซึ่งเป็นการสร้างแฟ้มคดันี้และตัวชี้ที่อยู่ให้เรียกข้อมูลได้โดยตรงและเร็วยิ่งขึ้น

5.1 รูปแบบของคำสั่งการสร้างคดันี้ฐานข้อมูล (CREATE INDEX, CREATE UNIQUE INDEX)

คำสั่งที่ใช้มี 2 รูปแบบ คือ

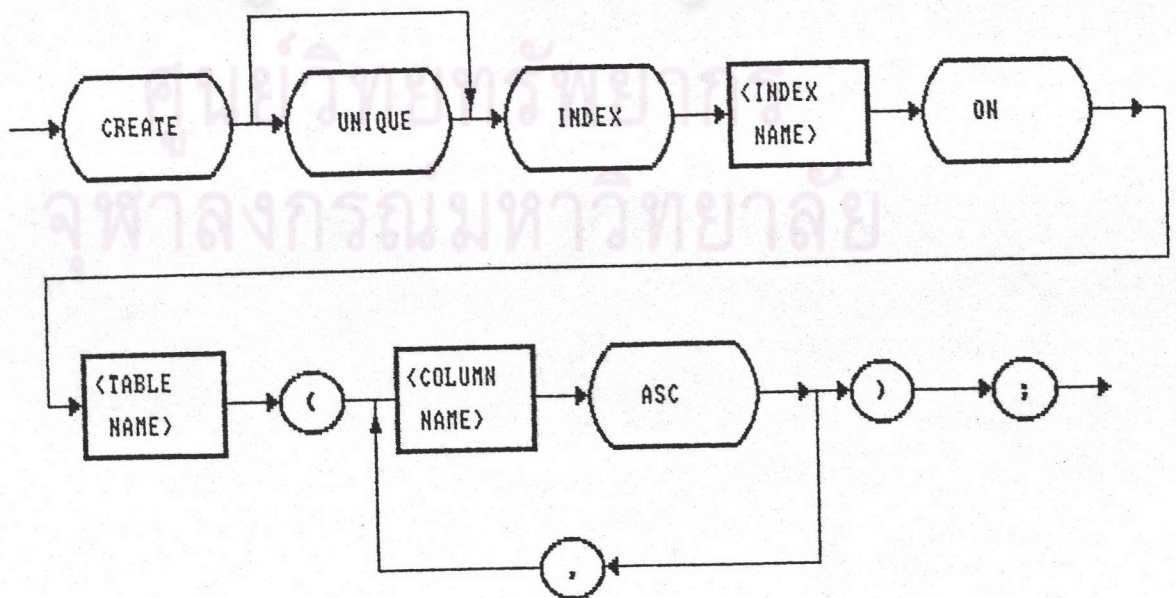
1. CREATE INDEX (2)

เป็นคำสั่งที่ใช้ในการสร้างคดันี้ฐานข้อมูล สำหรับข้อมูลที่มีความซ้ำซ้อนได้

2. CREATE UNIQUE INDEX (2)

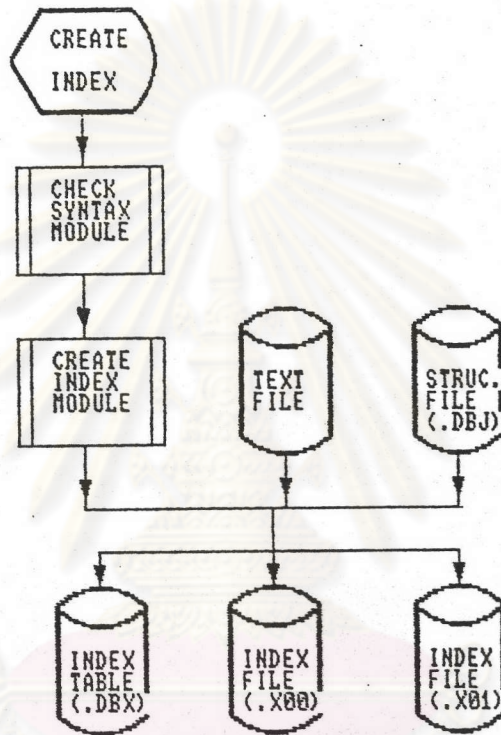
เป็นคำสั่งที่ใช้ในการสร้างคดันี้ฐานข้อมูล สำหรับข้อมูลที่ไม่มีความซ้ำซ้อน

มีรูปแบบของคำสั่งดังนี้



รูป 5.1 แสดงรูปแบบคำสั่งการสร้างคดันี้ฐานตารางข้อมูล

5.2 วิธีการสร้างดัชนีฐานข้อมูลและการใช้งาน



รูปที่ 5.2 แสดงการสร้างดัชนีฐานข้อมูล

การสร้างดัชนีฐานข้อมูลจะต้องมีการระบุชื่อ เขตข้อมูลที่ต้องการให้เรียงข้อมูลโดยโปรแกรม จะทำหน้าที่นำชื่อ เขตข้อมูลที่ระบุในคำสั่ง CREATE INDEX ไปอ่านโครงสร้างข้อมูลจากแฟ้มข้อมูลที่มีชนิด ของแฟ้มข้อมูลเป็น .DBJ เพื่อจะได้ทราบว่า เขตข้อมูลนั้นอยู่ตำแหน่ง ไบนารีที่เท่าไรนับจาก ไบนารีแรก และจะไปอ่านข้อมูลจากแฟ้มข้อมูลอีกชุดละละละ เบียน เพื่อดึง เขตข้อมูลส่วนที่เป็นคีย์ (KEY) ขึ้นมาสร้าง เป็นแฟ้มข้อมูลดัชนีขึ้น

การสร้างดัชนีฐานข้อมูล จะหาการสร้างแฟ้มข้อมูล 2 ลักษณะด้วยกันคือ

5.2.1 แฟ้มข้อมูลตารางดัชนีเก็บชื่อเขตข้อมูลที่เป็นคีย์ดัชนี (INDEX KEY) จะเก็บชื่อแฟ้มข้อมูลเป็น Filename.DBX เป็นแฟ้มข้อมูลแบบเรียงลำดับ และ ความยาวระเบียบนามคงที่ มีรูปแบบการเก็บดังนี้

ระเบียบ ที่ 1 เก็บ ชื่อตารางดัชนีตามด้วยเขตข้อมูลควบคุม ODOA

ระเบียบ ที่ 2 เก็บชื่อเขตข้อมูลที่เป็นคีย์ดัชนี สำหรับตารางดัชนีแรก (.X00) ตามด้วยเขตข้อมูลควบคุม ODOA

ระเบียบ ที่ 3 เก็บ พารามิเตอร์ 'ASC' ตามด้วยเขตข้อมูลควบคุม ODOA

ระเบียบ ที่ 4 เก็บ ชื่อตารางดัชนีที่ 2 ตามด้วยเขตข้อมูลควบคุม ODOA

ระเบียบ ที่ 5 เก็บชื่อเขตข้อมูลที่เป็นคีย์ดัชนี สำหรับตารางดัชนี ที่ 2 (.X01) ตามด้วยเขตข้อมูลควบคุม ODOA

และ ระเบียบถัดไปก็จะเก็บชื่อตารางดัชนีและชื่อ เขตข้อมูลที่เป็นคีย์ดัชนีของ ตารางดัชนีถัดไปเรื่อยๆ

ตัวอย่าง เช่น

```
CREATE INDEX TESTIDX ON TESTTAB (NAME ASC,SALARY ASC);
```

มีรูปแบบการเก็บแฟ้มข้อมูลดัชนี TESTTAB.DBX เป็น

```
TESTIDXODOANAMEODOAASCODOASALARYODOAASCODOA
```

5.2.2 แฟ้มข้อมูลดัชนี จะเก็บคีย์ดัชนีตามด้วยตัวชี้ที่เป็นที่อยู่สัมพันธ์กับจุดเริ่มต้นแฟ้มข้อมูล โดยใช้โครงสร้างข้อมูลของคั่นนั้นแบบทวิภาค (4) ในการเก็บแฟ้มข้อมูลนี้สามารถมีได้มากกว่า 1 แฟ้มข้อมูล ขึ้นอยู่กับความต้องการรูปแบบของการเรียงข้อมูลในลักษณะต่างๆ โดยจะสร้างแฟ้มข้อมูลดัชนีเท่ากับจำนวนคั่นที่ต้องการ และเก็บชื่อแฟ้มข้อมูลดัชนีเป็น Filename.X00, Filename.X01, Filename.x02,... ตามลำดับ

แฟ้มข้อมูลดัชนี มีส่วนประกอบ 4 ส่วนคือ

- 1) คีย์ดัชนี
- 2) ส่วนของตัวชี้ ของระเบียนนั้น เก็บที่อยู่สัมพันธ์กับจุดเริ่มต้นแฟ้มข้อมูล
- 3) ส่วนของ RBA ของระเบียนก่อนหน้า (RBA PREVIOUS RECORD)
- 4) ส่วนของ RBA ของระเบียนถัดไป (RBA NEXT RECORD)

INDEX KEY	RBA CURRENT REC.	RBA PREVIOUS REC.	RBA NEXT REC.
-----------	------------------	-------------------	---------------

รูปที่ 5.3 แสดงส่วนประกอบของแฟ้มข้อมูลดัชนี

ตัวอย่างเช่น โครงสร้างข้อมูลเป็นดังนี้

ID	10	ไบต์
NAME	40	ไบต์
ADDRESS	40	ไบต์

และมีค่าของข้อมูลดังนี้

ID	NAME	ADDRESS	CONTROL FIELD
2222222222	MALAIWAN	18/64 PAHONYOTIN RD. BKK.	ODOA
1111111111	PISSAMAI	281/48 PRARAMSEE RD. BKK.	ODOA
3333333333	YUWAREE	10/5 PHETBURI RD. BANGKOK.	ODOA

5.3 การคำนวณตัวชี้ตำแหน่ง

ตัวชี้ที่เก็บจะเป็นรูปแบบของที่อยู่สัมพันธ์กับจุดเริ่มต้น โดยมีการคำนวณตัวชี้เป็นดังนี้

(ความยาวระเบียบ + สองไบต์ ODOA) * (ลำดับที่ระเบียบ - 1)

วนที่นี้ ระเบียบ ที่ 1 จะมี RBA เป็น $(90 + 2) * (1 - 1) = 0$

ระเบียบ ที่ 2 จะมี RBA เป็น $(90 + 2) * (2 - 1) = 92$

ระเบียบ ที่ 3 จะมี RBA เป็น $(90 + 2) * (3 - 1) = 184$

หลังจากคำนวณ RBA แล้วจะมีการสร้างแฟ้มข้อมูลดังนี้ โดยให้โครงสร้างข้อมูลของคันทันมี
แบบทวิภาค

ตัวอย่าง เช่น

ต้องการสร้างเซตข้อมูล ชื่อ NAME เป็นดัชนี จะสร้างรูปแบบการเก็บเป็น

INDEX KEY	RBA CURRENT REC.	RBA PREVIOUS REC.	RBA NEXT REC.
PISSAMAI	0	92	184
MALAIWAN	92	-	0
YUWAREE	184	0	-

START PTR -->

5.4 รายละเอียดของขั้นตอนการสร้างดัชนีฐานข้อมูล

หลังจากผ่านการเช็คความถูกต้องเรียบร้อยแล้ว และเช็คว่าเป็นคำสั่ง CREATE INDEX แล้ว จะทราบชื่อแฟ้มข้อมูลตารางดัชนีและชื่อเซกซ์ข้อมูลที่จะสร้างดัชนี หลังจากนั้นโปรแกรมจะนำทางานตามลำดับดังนี้

- เปิดแฟ้มข้อมูลอักขระที่มีชนิดของแฟ้มข้อมูลเป็น .DB
- เปิดแฟ้มข้อมูลดัชนีที่มีชนิดของแฟ้มข้อมูลเป็น .X00, .X01, ... ถ้ามีแฟ้มข้อมูลนี้อยู่แล้ว จะทำการลบแฟ้มข้อมูลนี้ทิ้ง แล้วสร้างขึ้นมาใหม่เป็นแฟ้มข้อมูลคัมมี่ (DUMMY FILE) พร้อมกับทำการเริ่มคันทัวชี้ (INITIAL POINTER) ของ โครงสร้างข้อมูลแบบคันทัวชี้
- สร้างแฟ้มข้อมูลตารางดัชนีที่มีชนิดของแฟ้มข้อมูลเป็น .DBX เก็บชื่อเซกซ์ข้อมูลที่เป็นคันทัวชี้
- อ่านโครงสร้างข้อมูลจากแฟ้มข้อมูลโครงสร้างที่มีชนิดของแฟ้มข้อมูลเป็น .DBJ เพื่อทราบ

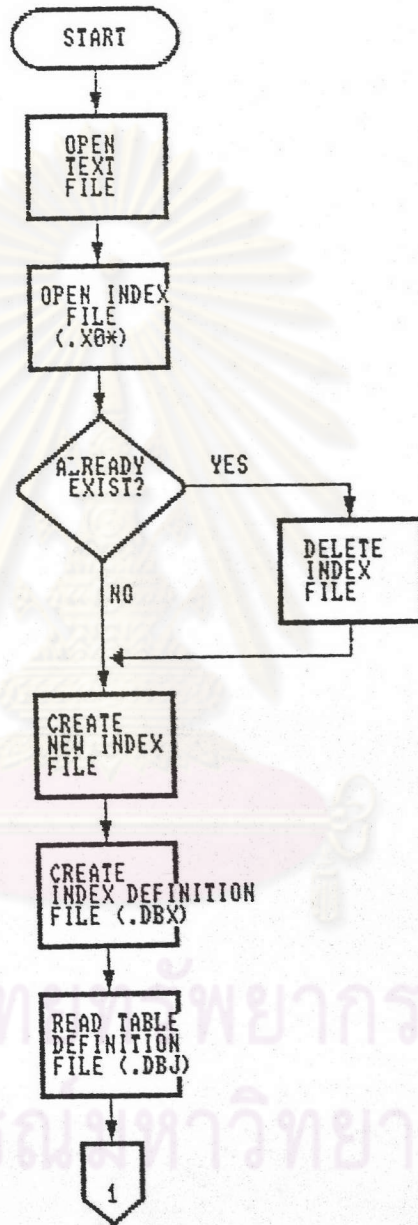
RBA

- อ่านข้อมูลจากแฟ้มข้อมูลอักขระที่ละระเบียบ และกวาดตรวจหา RBA ที่เป็นคีย์ และทำการเพิ่มเติม (INSERT) คีย์ใหม่พร้อมกับ RBA ซึ่งชี้ไปยังตำแหน่งเก็บของข้อมูลในแฟ้มข้อมูลเข้าไปในโครงสร้างข้อมูลแบบต้นไม้ทวิภาค

- ในกรณีที่เพิ่มเติมเข้ามาแล้ว จุดแตกกิ่ง (NODE) ของต้นไม้ (TREE) เริ่มไม่สมดุล จะทำการแยกจุดแตกกิ่ง และจัดการสมดุลต้นไม้ (BALANCE TREE) และเลื่อนตำแหน่งของแต่ละจุดแตกกิ่ง เพื่อให้จำนวนจุดแตกกิ่ง ในแต่ละข้างของต้นไม้สั้นที่สุด

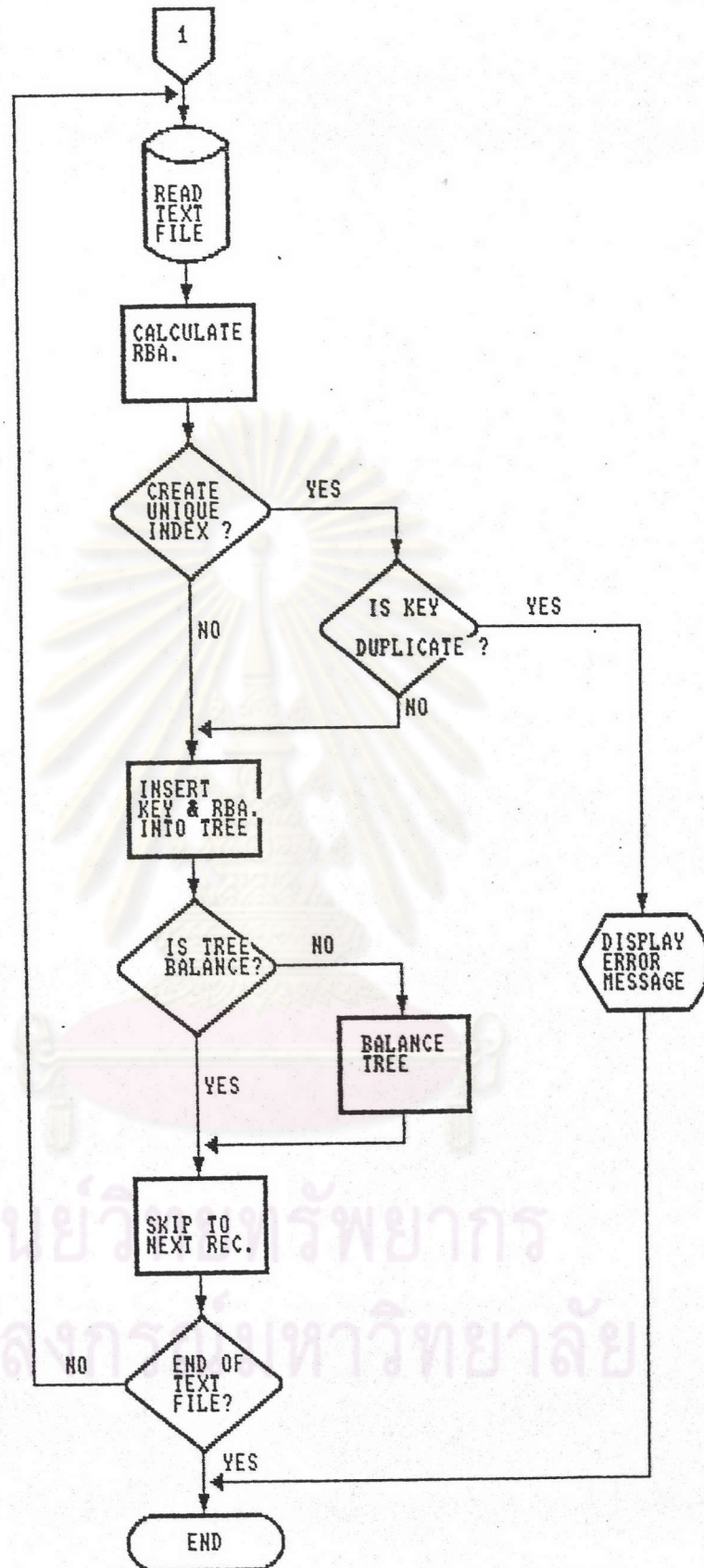
ในกรณีที่เป็นการสร้าง CREATE UNIQUE INDEX จะทำการเช็คเพิ่มเติมก่อนเพิ่มเติมระเบียบลงในโครงสร้างข้อมูลแบบต้นไม้ทวิภาค โดยจะเช็คคีย์ซ้ำ (DUPLICATE KEY) เสมอ ถ้าเกิดคีย์ซ้ำขึ้นจะแสดงข้อความผิดพลาดขึ้นบนหน้าจอภาพ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

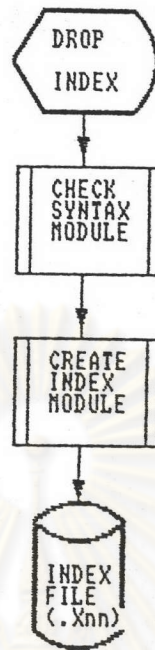


ศูนย์วิทยุพยากรณ์
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 5.4 แสดงแผนผังขั้นตอนการสร้างคีย์พื้นฐานข้อมูล



รูปที่ 5.4 แสดงแผนผังขั้นตอนการสร้างดัชนีฐานข้อมูล (ต่อ)



รูปที่ 5.6 แสดงการลบดัชนีฐานข้อมูล

5.7 รายละเอียดของโมดูลการลบดัชนีฐานข้อมูล

หลังจากบรรทัดคำสั่งผ่านการเช็คความถูกต้องเรียบร้อยแล้ว และเช็คว่าเป็นคำสั่งลบดัชนีแล้วโปรแกรมจะไปทำงานตามโมดูลชื่อ DROP_INDEX ซึ่งมีรายละเอียดการทำงานดังนี้

DROP_INDEX เป็นโมดูลซึ่งทำหน้าที่เปิดแฟ้มข้อมูลดัชนีที่มีชนิดของแฟ้มข้อมูลเป็น .Xnn และลบแฟ้มข้อมูลนั้นทิ้งไป พร้อมทั้งแสดงข้อความบนจอภาพว่า 'INDEX FILE HAS BEEN DROPPED'

จุฬาลงกรณ์มหาวิทยาลัย