

บทที่ 2

การเข้าถึงฐานข้อมูลการลงทะเบียนของนักศึกษา

การทำงานในระบบงานเดิมนั้น การจัดเก็บและการเรียกค้นฐานข้อมูลแบบสัมพันธ์ได้ใช้ภาษาสอบถามเชิงโครงสร้าง ซึ่งเป็นภาษาฐานข้อมูล ที่คล้ายภาษาธรรมชาติ และได้ใช้ระบบฐานข้อมูลการลงทะเบียนของนักศึกษาสถาบันราชภัฏสวนสุนันทา เป็นกรณีศึกษาดังนี้

ฐานข้อมูลที่ใช้เรียกค้น ประกอบด้วย 10 ตาราง (Table) คือ

1. ตารางอาจารย์ (Teacher Table)
2. ตารางนักศึกษา (Stud Table)
3. ตารางวิชา (Sub_ject Table)
4. ตารางอาจารย์-นักศึกษา (Tea-Stu Table)
5. ตารางอาจารย์-วิชา (Tea-Sub Table)
6. ตารางนักศึกษา-วิชา (Stu-Sub Table)
7. ตารางวิชาเอก (Major Table)
8. ตารางวิชาโท (Minor Table)
9. ตารางคุณวุฒิอาจารย์ (Degree Table)
10. ตารางคณะวิชา (Faculty Table)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางอาจารย์ ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
TEAID	Character	4	-	รหัสประจำตัวอาจารย์
TEANAME	Character	50	-	ชื่อ-สกุลอาจารย์
TEA DEG	Character	1	-	ระดับคุณวุฒิอาจารย์
TEAFAC	Numeric	2	-	สังกัดคณะของอาจารย์
SALARY	Numeric	9	2	เงินเดือน

ตารางที่ 1 แสดงโครงสร้างของตารางอาจารย์

ตารางนักศึกษา ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
STUID	Character	7	-	รหัสประจำตัวนักศึกษา
STUNAME	Character	50	-	ชื่อ-สกุลนักศึกษา
STUMAJ	Character	2	-	รหัสวิชาเอก
STUMIN	Character	2	-	รหัสวิชาโท

ตารางที่ 2 แสดงโครงสร้างของตารางนักศึกษา

ตารางวิชา ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
SUBID	Character	3	-	รหัสประจำตัววิชา
SUBNAME	Character	50	-	ชื่อวิชา
SUBCRE	Character	1	-	หน่วยกิจของวิชา
SUBROOM	Character	3	-	ห้องเรียน

ตารางที่ 3 แสดงโครงสร้างของตารางวิชา

ตารางอาจารย์-นักศึกษา ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
TEA_ID	Character	4	-	รหัสประจำตัวอาจารย์
STU_ID	Character	7	-	รหัสประจำตัวนักศึกษา

ตารางที่ 4 แสดงโครงสร้างของตารางอาจารย์-นักศึกษา

ตารางอาจารย์-วิชา ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
TEAI_D	Character	4	-	รหัสประจำตัวอาจารย์
SUB_ID	Character	3	-	รหัสประจำวิชา

ตารางที่ 5 แสดงโครงสร้างของตารางอาจารย์-วิชา

ตารางนักศึกษา-วิชา ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
STUI_D	Character	7	-	รหัสประจำตัวนักศึกษา
SUBI_D	Character	3	-	รหัสประจำวิชา

ตารางที่ 6 แสดงโครงสร้างของตารางนักศึกษา-วิชา

ตารางวิชาเอก ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
STU_MAJ	Character	2	-	รหัสวิชาเอก
MAJNAME	Character	50	-	ชื่อวิชาเอก

ตารางที่ 7 แสดงโครงสร้างของตารางวิชาเอก

ตารางวิชาโท ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
STU_MIN	Character	2	-	รหัสวิชาโท
MINNAME	Character	50	-	ชื่อวิชาโท

ตารางที่ 8 แสดงโครงสร้างของตารางวิชาโท

ตารางคุณวุฒิอาจารย์ ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
TEA_DEG	Character	1	-	คุณวุฒิอาจารย์
DEGNAME	Character	50	-	ระดับคุณวุฒิ

ตารางที่ 9 แสดงโครงสร้างของตารางคุณวุฒิอาจารย์

ตารางคณะวิชา ประกอบด้วยเขตข้อมูลดังนี้

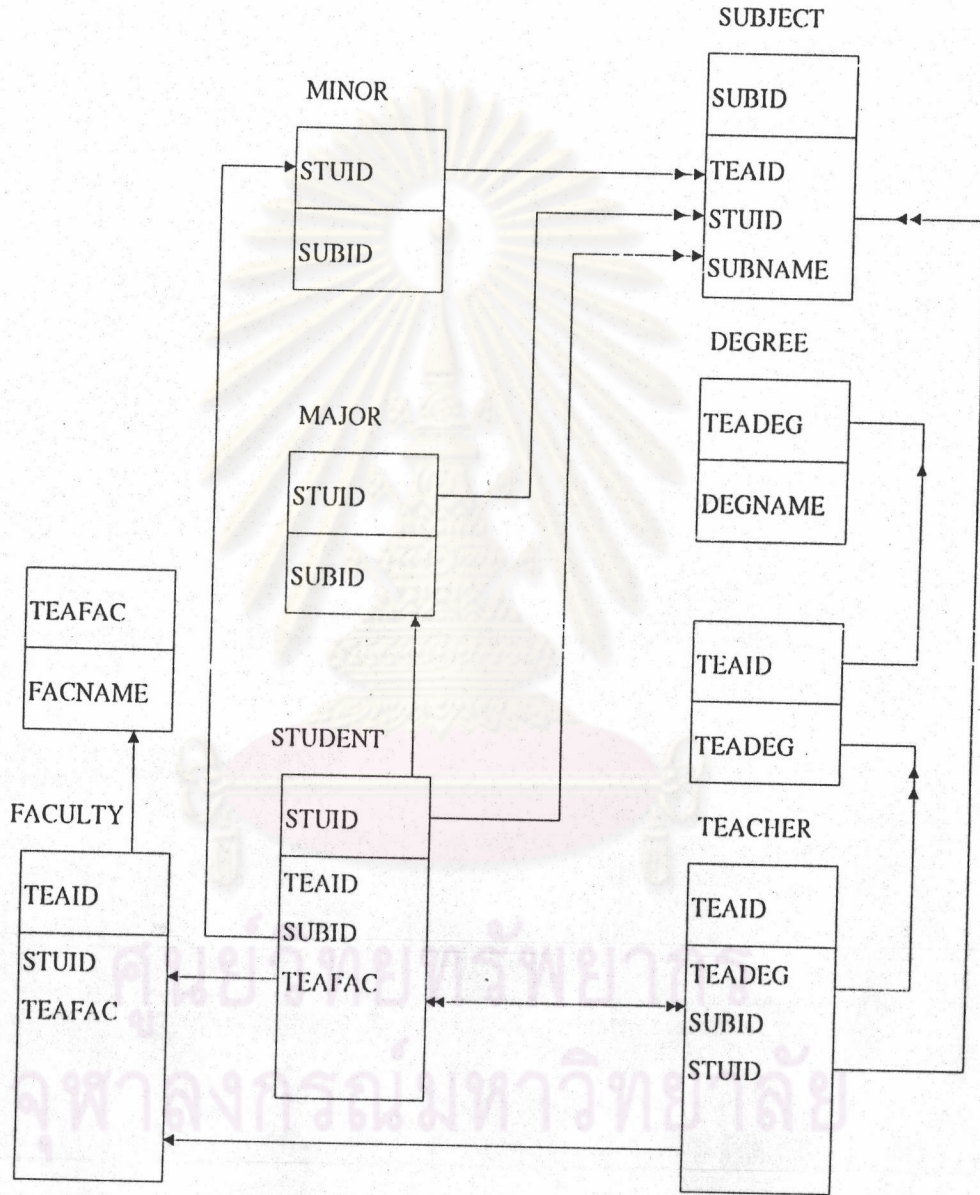
เขตข้อมูล	ชนิด	ความกว้าง	ทศนิยม	คำอธิบาย
TEA_FAC	Numeric	2	-	สังกัดคณะของอาจารย์
FACNAME	Character	50	-	ชื่อคณะวิชา

ตารางที่ 10 แสดงโครงสร้างของตารางคณะวิชา

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางข้อมูล ดังกล่าวมาแล้ว มีความสัมพันธ์ ดังตัวแบบความสัมพันธ์ของข้อมูล ดังนี้

ตัวแบบความสัมพันธ์ของข้อมูล



รูปที่ 1 แสดงตัวแบบความสัมพันธ์ของข้อมูลการลงทะเบียน



การใช้ภาษาสอบถามเชิงโครงสร้างในระบบงานเดิม

ระบบงานการลงทะเบียนเรียน ของสถาบันราชภัฏสวนสุนันทา ได้มีการใช้ภาษาฐานข้อมูลแบบสัมพันธ์ที่ประกอบด้วยประโยคคำสั่งเพื่อ เพิ่ม แทรก เพิ่ม แก้ไข ลบ เรียกค้น และป้องกันข้อมูล ภาษาสอบถามเชิงโครงสร้างไม่ใช่โปรแกรมจัดการฐานข้อมูลโดยตรง แต่ภาษาสอบถามเชิงโครงสร้างเป็นเพียงการแก้ไข ลบ เรียกค้น และป้องกันข้อมูลภาษาสอบถามเชิงโครงสร้างภาษาสอบถามเชิงโครงสร้าง เป็นเพียงฐานข้อมูลหนึ่งที่นิยมใช้ควบคู่ไปกับระบบการจัดการฐานข้อมูลแบบสัมพันธ์ เมื่อเราต้องการเรียกค้นข้อมูล เราเพียงแต่ใช้ภาษาสอบถามเชิงโครงสร้าง บอกความต้องการข้อมูลผ่านโปรแกรมจัดการฐานข้อมูลหรือ ระบบจัดการฐานข้อมูล ซึ่งจะดำเนินการเรียกค้นข้อมูลให้เรา การเรียกค้น (Query) จึงเป็นที่มาของชื่อ Structure Query Language มีชื่อย่อว่า SQL ซึ่งให้ความหมายที่ไม่ถูกต้องนัก เพราะภาษาสอบถามเชิงโครงสร้างไม่ใช่เป็นเครื่องมือในการเรียกค้นข้อมูลอย่างเดียว แต่ยังใช้ในการควบคุมการทำงานของโปรแกรมฐานข้อมูล ในด้านต่าง ๆ อีกหลายอย่าง ให้กับผู้ใช้งานหรือโปรแกรมประยุกต์ต่าง ๆ ได้แก่

1. การกำหนดโครงสร้าง การจัดเก็บ และการโยกความสัมพันธ์ของตารางข้อมูล
2. การเรียกค้นข้อมูลและใช้งาน ข้อมูลที่เก็บในฐานข้อมูล
3. การเพิ่ม แทรกข้อมูล การลบ และการตัดแปลงแก้ไขข้อมูลที่เก็บไว้แล้วในฐานข้อมูล
4. การควบคุมความปลอดภัยของการใช้ข้อมูล การกำหนดสิทธิการเรียกค้น การแก้ไข การตัดแปลงข้อมูล
5. การควบคุมการใช้ข้อมูลร่วมกันระหว่างผู้ใช้หลาย ๆ คน
6. การควบคุมความถูกต้องของข้อมูล ป้องกันการล่มสลายของระบบฐานข้อมูล

ภาษาสอบถามเชิงโครงสร้าง ไม่ใช่ภาษาโครงสร้างเหมือนกับภาษาโครงสร้างอื่น ๆ เช่น ภาษาซี หรือ ภาษาปาสคาล ภาษาสอบถามเชิงโครงสร้างจัดเป็นภาษา แบบภาษาไร้กระบวนการ คำสั่ง (Non-Procedural Language) กล่าวคือผู้ใช้เพียงระบุว่าต้องการข้อมูลอะไร โดยไม่ต้องบอกขั้นตอนวิธีการค้นหาว่าได้มาอย่างไร ต่างกับภาษาระดับสูงอื่นๆ เช่น โคบอล ซี เบสิก ปาสคาล ฯลฯ ที่เป็นภาษาแบบภาษาเชิงกระบวนการ (Procedural Language) ซึ่งต้องบอกขั้นตอนวิธีการค้นหา

ภาษาสอบถามเชิงโครงสร้าง ไม่ใช่ภาษาที่ครบสมบูรณ์เหมือนภาษาคอมพิวเตอร์อื่น ๆ ตรงที่ไม่มีคำสั่ง กระโดด เลือกลงทาง การทำงานของคำสั่ง If ตามเงื่อนไข คำสั่ง Goto หรือ Branch ไม่มีคำสั่งประเภท ทำงานเป็นวงรอบของประโยคคำสั่ง Do หรือ For ภาษาสอบถามเชิงโครงสร้าง จึงเป็นภาษาย่อยที่อาจใช้แบบได้ตอบเพื่อเรียกใช้งานติดต่อระบบฐานข้อมูลทันที หรือใช้แบบการฝังตัว (Embedded SQL) โดยนำประโยคคำสั่งภาษาสอบถามเชิงโครงสร้าง ไปฝังรวมในภาษาอื่นเช่น ภาษาโคบอล ฟอรัแทรน ซี เป็นต้น เพื่อขยายให้ภาษานั้นเรียกใช้ติดต่อฐานข้อมูล ได้ง่ายและสะดวกขึ้น

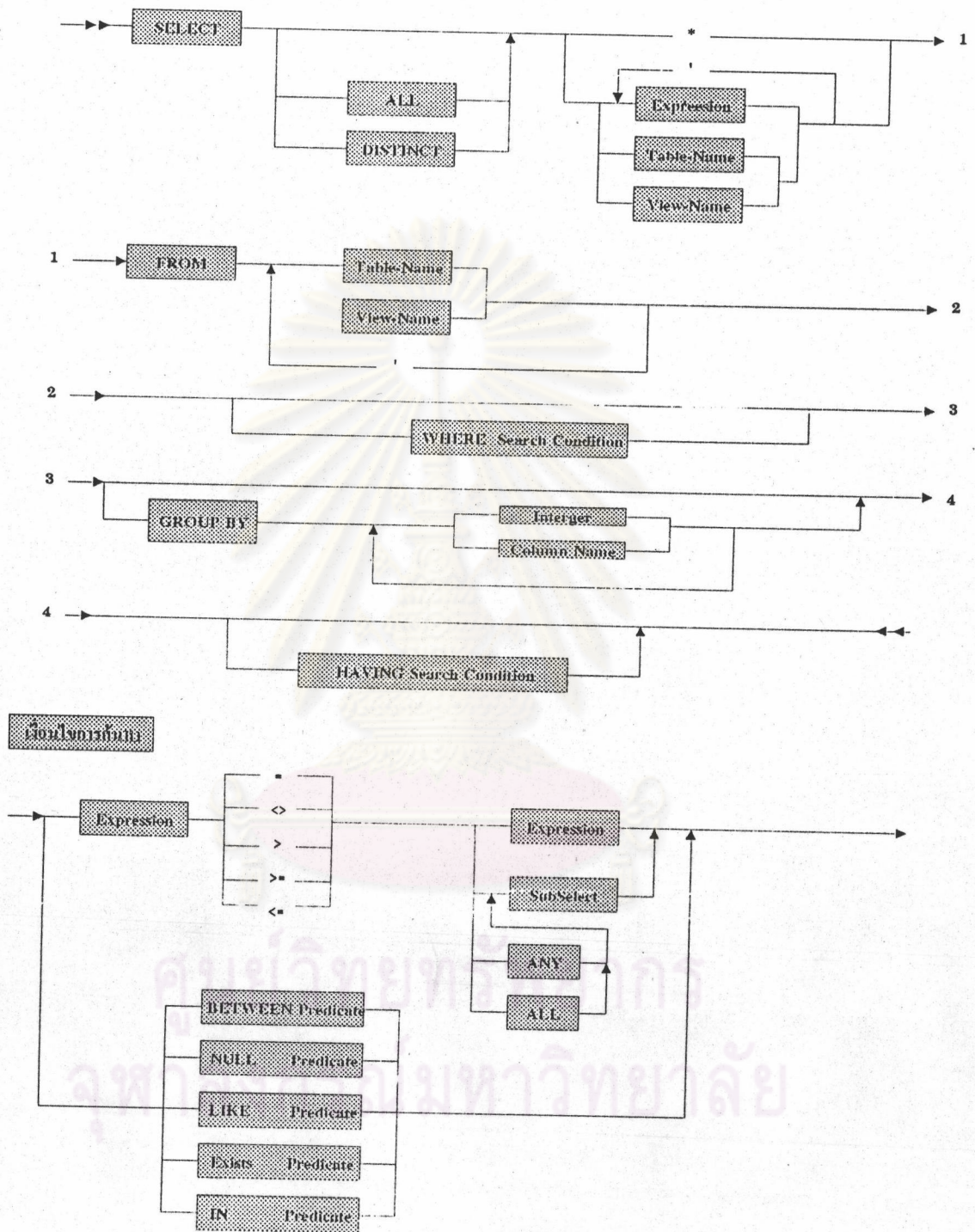
ความสามารถของภาษาสอบถามเชิงโครงสร้าง

ความสามารถของภาษาสอบถามเชิงโครงสร้างมีดังนี้

1. เป็นภาษาที่ใช้เรียกค้นได้ตอบได้ทันทีโดยเขียนคำสั่งภาษาสอบถามเชิงโครงสร้าง ผ่านโปรแกรมไปเรียกค้นข้อมูลทางเครื่องประมวลฐานข้อมูล (Database Engine) ไปสู่ฐานข้อมูลในดิสก์ แล้วแสดงผลลัพธ์ ออกมาที่หน้าจอ
2. เป็นภาษาโปรแกรมฐานข้อมูลที่ผู้เขียนโปรแกรมประยุกต์สามารถเขียนคำสั่งภาษาสอบถามเชิงโครงสร้าง ฝังเข้าไป รวมในโปรแกรมภาษาอื่น ๆ เพื่อเรียกค้นข้อมูลจากฐานข้อมูล
3. เป็นภาษาสอบถามเชิงโครงสร้าง เพื่อกำหนดโครงสร้างและควบคุมการเรียกค้นข้อมูลที่เก็บไว้
4. เป็นภาษาของระบบไคลเอนต์/เซิร์ฟเวอร์ (Client/Server) โดยไคลเอนต์จะใช้ภาษาสอบถามเชิงโครงสร้างในการติดต่อผ่านเครือข่ายท้องถิ่น (Local Area Network) ไปยังแหล่งบริการฐานข้อมูล (Database Server)
5. เป็นภาษาฐานข้อมูลที่ช่วยในการกระจายข้อมูลไปยังระบบคอมพิวเตอร์ ที่ต่ออยู่ในระบบจัดการฐานข้อมูล
6. เป็นภาษาช่องประตูไว้ติดต่อกับฐานข้อมูลอื่น ๆ (Database Gateway Language) โดยภาษาภาษาสอบถามเชิงโครงสร้าง จะเป็นสื่อกลางในการติดต่อ

สำหรับการวิจัยนี้ ได้กำหนดขอบเขตของการวิจัย โดยใช้คำสั่งภาษาสอบถามเชิงโครงสร้างเพียงคำสั่งเดียวเท่านั้น คือ คำสั่ง SELECT ซึ่งมีการใช้หลายรูปแบบดังนี้

ตัวอย่างผังไวยากรณ์คำสั่ง SELECT และเงื่อนไขการค้นหา



รูปที่ 2 ผังไวยากรณ์คำสั่ง SELECT

ตัวอย่างการใช้คำสั่งภาษาสอบถามเชิงโครงสร้างในระบบเดิม

1. แสดงข้อมูลทั้งหมดจากตารางนักศึกษา

คำสั่ง `Select * from stud`

2. แสดงข้อมูลทั้งหมดจากตารางอาจารย์-รายละเอียดวิชา

คำสั่ง `Select * from tea_sub`

3. แสดงรหัสนักศึกษา , ชื่อ , วิชาเอก จากตารางนักศึกษาที่มีวิชาเอกคอมพิวเตอร์ศึกษา (09)

คำสั่ง `Select Stuid Stuname Stumaj from stud where Stumaj = '09'`

4. แสดงชื่ออาจารย์จากตารางอาจารย์ โดยที่เงินเดือนมากกว่า 10000 และ อยู่คณะวิทยาศาสตร์และเทคโนโลยี

คำสั่ง `Select Teaname Teafac Salary from teacher where Salary > 10000 and Teafac = '01'`

5. แสดงรหัสนักศึกษา ชื่อ วิชาเอก จากตาราง นักศึกษา โดยเรียงรหัสนักศึกษาจากมากไปหาน้อย

คำสั่ง `Select Stuid Stuname Stumaj from Stud order by Stuid descending`

6. แสดง รหัสวิชา ชื่อวิชา จากตารางรายละเอียดวิชา โดยที่ ชื่อวิชาต้องขึ้นต้นด้วย " ก " และตามด้วยอักษรอะไรก็ได้

คำสั่ง `Select Subid Subname from Sub_ject where Subname Like 'ก*'`

7. ให้แสดงผลรวมเงินเดือนของอาจารย์ทั้งหมดจากตารางอาจารย์

คำสั่ง `Select Sum(Salary) from teacher`

8. ให้นำค่าเฉลี่ยของเงินเดือนของอาจารย์ทั้งหมดจากตารางอาจารย์
คำสั่ง `Select Avg(Salary) from teacher`
9. ให้นำจำนวนอาจารย์ที่สังกัดอยู่คณะวิทยาศาสตร์และเทคโนโลยีจากตารางอาจารย์
คำสั่ง `Select Count(Teafac) from teacher where Teafac = 01`
10. การคิดเงินเดือนอาจารย์แต่ละคนใน 1 ปี
คำสั่ง `Select Teaid Teaname SALARY*12 from Teacher`
11. ให้นำแสดงชื่อห้องว่ามีเรียนกี่ครั้ง
คำสั่ง `Select SubRoom from sub_ject where Subroom = '942'`
12. ให้นำแสดงชื่ออาจารย์ ที่มีเงินเดือนระหว่าง 10000 ถึง 50000
คำสั่ง `Select Teaname, Salary from teacher where SALARY >= 10000 and SALARY <= 50000`
13. ให้นำแสดงชื่ออาจารย์ ที่มีเงินเดือนมากกว่า 5000 และอยู่คณะครุศาสตร์ และมีคุณวุฒิปริญญาเอก
คำสั่ง `Select Teaname , Teadege from teacher where Teadege = '3' and Teafac = '02' and Sal > 5000`
14. ให้นำแสดงชื่อนักศึกษา ที่มีวิชาโท 42 หรือ วิชาเอก 09
คำสั่ง `Select Stuname from StuD where Minor = '42' or Major = '09'`
15. ให้นำแสดงวิชา ห้องเรียน ที่มีหน่วยกิตเท่ากับ 3
คำสั่ง `Select Subject Subroom from Sub by where Subcredit = '3'`

จะเห็นว่าในระบบงานเดิมนั้นใช้คำสั่งภาษาสอบถามเชิงโครงสร้าง ทำให้ผู้ใช้ที่ไม่คุ้นเคยกับคอมพิวเตอร์มากนักใช้คอมพิวเตอร์ไม่สะดวกเท่าที่ควร

ในการทำการวิจัยนี้เลือกระบบการจัดการฐานข้อมูลโดยพิจารณาจากความสามารถในการเก็บข้อมูลรูปภาพ และความสามารถในการเรียกค้นข้อมูล โปรแกรมจัดการฐานข้อมูลที่มีความสามารถดังกล่าวมีหลายโปรแกรมด้วยกัน และทำงานภายใต้ระบบปฏิบัติการวินโดวส์

โปรแกรมจัดการฐานข้อมูลเดลไฟ (Delphi) โดยบริษัทออร์แลนดเป็นผู้พัฒนา ภายใต้ระบบปฏิบัติการวินโดวส์ มีคุณสมบัติที่ดีทางด้านฐานข้อมูล และทำให้ผู้ใช้สามารถเตรียมโปรแกรมได้อย่างง่ายดาย ผู้วิจัยจึงเลือกโปรแกรมเดลไฟมาใช้ในการพัฒนาโปรแกรมการเข้าถึงฐานข้อมูลโดยภาษาธรรมชาติ

โปรแกรมเดลไฟ

โปรแกรมเดลไฟเป็นโปรแกรมที่น่าสนใจ เพราะนอกจากจะมีภาษาแบบวิซวลปาสคาล (Visual Pascal) แล้วยังมีองค์ประกอบอื่น ๆ ด้วย ไม่ว่าจะเป็น โปรแกรมReportSmith (Report Smith) ซึ่งเป็นโปรแกรมออกแบบรายงานที่มีประสิทธิภาพสูงมากและมีโปรแกรมอินเตอร์เบส (Interbase) ซึ่งเป็นโปรแกรมฐานข้อมูลทำงานอยู่ในเครื่องหลายระดับคือมีทั้งบนเครื่องระดับมินิคอมพิวเตอร์ เวิร์กสเตชัน และ ไมโครคอมพิวเตอร์ จึงเป็นโปรแกรมฐานข้อมูลที่มีชื่อเสียงมากตัวหนึ่ง นอกจากนั้นยังสามารถทำงานในระดับ ไคลเอนต์-เซิร์ฟเวอร์ (Client-Server)

โปรแกรมเดลไฟ เป็นโปรแกรมภาษาปาสคาลที่ให้สภาพแวดล้อมการทำงานแบบวิซวล ซึ่งโดยทั่วไปการเขียนโปรแกรมวิซวลจะประกอบด้วยการสร้างฟอร์ม แล้วลากออกแบบต่าง ๆ มาวางบนฟอร์ม จากนั้นทำการแต่งออกแบบนั้น ๆ ให้ทำงานตามต้องการและเขียนโปรแกรมภาษาปาสคาล เพื่อควบคุมเหตุการณ์ต่าง ๆ สำหรับความสามารถที่เป็นจุดเด่นของเดลไฟมีดังนี้

1. โปรแกรมเดลไฟ จะเป็น Native Code Compiler หมายความว่า เดลไฟ จะทำการแปลโปรแกรมภาษาปาสคาล ให้เป็นภาษาเครื่องโดยตรงในขณะที่โปรแกรมวิซวลเบสิกเพียงแต่แปลมาเป็นคำสั่งเทียม (Pseudo - Compiler) เท่านั้น ดังนั้นโปรแกรมที่เขียนด้วยภาษาวิซวลเบสิก จึงต้องมีการแปลเป็นคำสั่งจริงทุกครั้งที่มีการเรียกใช้งาน จะทำให้โปรแกรมที่เขียนด้วยเดลไฟสามารถทำงานได้เร็วกว่าโปรแกรมลักษณะเดียวกับที่เขียนด้วยวิซวลเบสิก ประมาณ 10 เท่า (ธนา หงษ์สุวรรณ, 2539)

2. โปรแกรมเคลฟไฟ มีแกนกลางเป็นภาษาปาสคาล แต่ โปรแกรมวิซวลเบสิก ใช้ภาษาเบสิกเป็นแกนกลาง ซึ่งภาษาปาสคาลนั้นได้รับการยอมรับว่าเป็นภาษาที่มีโครงสร้างที่ดีมากกว่าเบสิก และมีความยืดหยุ่นต่อการใช้งานมากกว่าภาษาเบสิก ทำให้การเขียนโปรแกรมขึ้นใช้งานทำได้ง่ายมากกว่า และหาจุดผิดพลาดได้ง่าย

3. โปรแกรมเคลฟไฟ ในเวอร์ชัน 1.0 ได้ให้ความสามารถทางด้านฐานข้อมูลมาให้อย่างเต็มที่ โดยสามารถเรียกใช้ฐานข้อมูลได้ 2 แบบ ด้วยกัน

โดยแบบแรก เป็นการเรียกผ่าน ODBC (Object Database Connectivity) ซึ่งเป็นมาตรฐานของการเชื่อมต่อระหว่างโปรแกรมฐานข้อมูล ที่กำหนดโดยไมโครซอฟต์ และอีกแบบหนึ่ง ผ่านทางตัวเชื่อมภาษาสอบถามเชิงโครงสร้าง(SQL Link) ซึ่งจะมีประสิทธิภาพที่ดีกว่า ODBC แต่จะมีข้อจำกัดที่สามารถใช้กับโปรแกรมฐานข้อมูลได้เพียง 4 ตัว คือ Oracle, Informix, Sybase และ Interbase ซึ่งเป็นของบริษัทเทอร์เนตเอง

โดยในเคลฟไฟ ได้ให้โปรแกรมอินเทอร์เบสที่เป็นเวอร์ชันสำหรับผู้ใช้คนเดียวของอินเทอร์เบส 4.0 ทำการพัฒนาโปรแกรมที่ใช้กับฐานข้อมูลอินเทอร์เบส สามารถพัฒนามนเครื่องที่ใช้คนเดียวก่อนแล้วจึงย้ายไปใช้กับฐานข้อมูลในเครื่องที่ใหญ่กว่าในลักษณะของไคลเอนต์/เซิร์ฟเวอร์ได้โดยไม่จำเป็นต้องแก้ไขโปรแกรม

4. โปรแกรมเคลฟไฟ มีโปรแกรมสร้างรายงานReporSmith ที่มีประสิทธิภาพสูง ซึ่งจะช่วยให้สามารถออกรายงานได้ง่ายขึ้น แทนที่จะต้องเขียนโปรแกรมขึ้นมาเองทั้งหมด ซึ่งพบว่าเป็นการเชื่อมต่อที่เหมาะสม เพราะในขณะที่เคลฟไฟมีความโดดเด่นในเรื่องการออกแบบ และการเขียนโปรแกรมที่เกี่ยวข้องกับส่วนติดต่อผู้ใช้ผ่านทางจอภาพ แต่มีข้อด้อยในการแสดงผลแบบอื่น ๆ ที่ไม่ได้ผ่านทางจอภาพ โปรแกรมReporSmithจะเข้ามาเสริมจุดอ่อนในข้อนี้ของเคลฟไฟ ซึ่งเมื่อรวมเอาความสามารถทั้งหมดเข้าด้วยกันแล้ว จะพบว่าการเขียน โปรแกรมบนวินโดวส์ไม่ยาก

5. โปรแกรมเคลฟไฟได้ให้ปุ่มควบคุม ซึ่งถือเป็น เครื่องมือที่ช่วยในการพัฒนาโปรแกรมมาให้ถึง 72 ตัว โดยได้ออกแบบมาให้มีลักษณะของการตั้งแท็บ ทำให้ประหยัดเนื้อที่ในการใช้งาน และยังสามารถจัดหมวดหมู่ของคอนโทรลได้ ทำให้การใช้งานสะดวกมากขึ้นกว่าวิซวลเบสิก ที่คอนโทรลถูกนำมารวมไว้ที่เดียวกัน

นอกจากนั้นเคลฟ ยังสามารถเรียกใช้คอนโทรลของวิซวลเบสิก ที่อยู่ในรูปของ VBX ได้อีกด้วย และที่สำคัญคือเคลฟสามารถพัฒนาสร้างคอนโทรลได้โดยตรง โดยใช้ภาษาปาสคาล ทำให้การพัฒนาได้สะดวกกว่า

6. รูปแบบสภาพแวดล้อมในการเขียนโปรแกรม ซึ่งทางบอร์แลนด์ออกแบบมาได้อย่างดีมาก โดยเคลฟได้นำรูปแบบการใช้งานในลักษณะการตั้งแท็บมาใช้ ทำให้ประหยัดเนื้อที่ในการใช้งานได้มาก และทำให้ไม่มี ToolBox กับ Properties มากเต็มหน้าจอ และอีกอย่างหนึ่งคือ การเอาส่วนรูทีนย่อยต่าง ๆ มาไว้ในส่วนเดียวกัน ทำให้สามารถเลื่อนขึ้น-ลง เพื่อดูโปรแกรมทั้งหมดในยูนิิตได้ทั้งหมด ในขณะที่ยังคงความสามารถทางด้าน การแก้ไขส่วนต่าง ๆ ได้อย่างรวดเร็ว

หลักการเขียนโปรแกรมแบบวิซวล (Visual)

ในการเขียนโปรแกรมแบบวิซวลนั้นการเขียนโปรแกรมจะต่างออกไปจากการเขียนโปรแกรมภาษาปาสคาลตามปกติ คือ ในการเขียนโปรแกรมแบบเดิม จะต้องออกแบบหน้าต่างของการแสดงผลที่จอภาพระบุตำแหน่งต่าง ๆ ที่ใช้ในการแสดงผล จากนั้นก็วิเคราะห์ปัญหาเพื่อหาวิธีการในการเขียนโปรแกรมให้ดีที่สุดแล้วจึงทำการเขียนโปรแกรม ซึ่งโปรแกรมที่เขียนมีลักษณะของการทำงานตามขั้นตอนที่กำหนด ตามลำดับที่ได้กำหนดไว้ด้วย

พบว่า การเขียนโปรแกรมลักษณะข้างต้นที่กล่าวมา เหมาะสำหรับสภาพแวดล้อมตามแบบคอสที่โปรแกรมทำงานที่ละโปรแกรมเท่านั้น แต่กับสภาพแวดล้อมแบบวินโดวส์แล้วการเขียนโปรแกรมตามแบบเดิมไม่เหมาะสมอีกต่อไป เพราะในสภาพแวดล้อมแบบวินโดวส์นั้นสามารถมีโปรแกรมทำงานพร้อม ๆ กันได้หลายโปรแกรมในหน้าจอเดียวกัน เพียงแค่ลากเมาส์จากวินโดวส์หนึ่งไปยังอีกวินโดวส์หนึ่งก็เป็นการเปลี่ยนโปรแกรมทำงานแล้ว ดังนั้นการเขียนโปรแกรมที่เหมาะสมกับสภาพแวดล้อมแบบวินโดวส์ เป็นแบบที่เรียกว่า Event-Driven Programming ซึ่งการเขียนโปรแกรมในเคลฟ จะเป็นแบบนี้ การเขียนโปรแกรมใน เคลฟจะใช้หลักการของการมองเห็น โดยเริ่มจากออกแบบส่วนประกอบของโปรแกรมไว้ในวินโดวส์ย่อยที่เรียกว่า ฟอรั่ม (Form) ภายในฟอรั่มจะประกอบด้วยสิ่งต่าง ๆ ที่ประกอบเป็นวินโดวส์ ที่เรียกว่า ออบเจค (Object) เช่น ข้อความ ช่องรับข้อความ รูปภาพ สกรอลบาร์ และปุ่มต่าง ๆ



การสร้างโปรแกรมจะเริ่มจากการกำหนดองค์ประกอบที่ต้องการลงในวินโดวส์ จากนั้นจึงกำหนดการทำงานให้กับองค์ประกอบ โดยการเขียนโปรแกรมกำกับออบเจกต์เหล่านั้น เช่น การเลือกปุ่ม OK ก็จะทำงานอย่างหนึ่ง เป็นต้น โปรแกรมที่มีการทำงานในลักษณะ เช่นนี้ เรียกว่า Event-Driven Programming หรือ โปรแกรมที่ทำงานตามเหตุการณ์ โดยทุกครั้ง ที่มีเหตุการณ์เกิดขึ้นออบเจกต์ ต่าง ๆ จะตอบสนองกับเหตุการณ์นั้น ๆ เช่น หากมีการเลื่อนเมาส์ผ่านวินโดวส์ใด วินโดวส์นั้นก็จะมี active ขึ้นมา หรือหากมีการกดคีย์ที่ออบเจกต์ที่รับคีย์ก็จะมีแสดงตัวอักษรบนจอภาพ ในการทำงานของไมโครซอฟท์วินโดวส์จะมีเหตุการณ์ต่าง ๆ ขึ้น ออบเจกต์ต่าง ๆ จะตอบสนองกับเหตุการณ์ได้ไม่เหมือนกัน เช่น ออบเจกต์ที่เป็นปุ่มก็อาจจะไม่ตอบสนองกับการคีย์ เป็นต้น ดังนั้น จะต้องเขียนโปรแกรมกำกับออบเจกต์ให้ทำงานเฉพาะเหตุการณ์ที่สนใจเท่านั้น เช่น หากเป็นออบเจกต์ที่รับคีย์บอร์ดก็อาจจะไม่ต้องสนใจกับการเคลื่อนที่ของเมาส์เลย

จะเห็นได้ว่า การเขียนโปรแกรมแบบนี้จะแตกต่าง กับการเขียนโปรแกรมแบบเดิมอย่างสิ้นเชิง ซึ่งจริงแล้วก็เหมือนกับการแบ่งโปรแกรมออกเป็นส่วนตัวต่าง ๆ ดังนั้นแทนที่จะเขียนโปรแกรมครั้งเดียวยาว ๆ ก็จะกลายเป็นโปรแกรมสั้น ๆ หลายครั้ง ซึ่งจะง่ายต่อการเขียนและการตรวจสอบความผิดพลาดมากกว่าการเขียนโปรแกรมแบบเดิม

คุณสมบัติของออบเจกต์ (Object)

เนื่องจากโปรแกรมในไมโครซอฟท์ วินโดวส์ จะประกอบด้วยวินโดวส์ และวินโดวส์เป็นออบเจกต์ ดังนั้นออบเจกต์ จึงเป็นส่วนประกอบพื้นฐาน ที่สำคัญๆ ของโปรแกรม ออบเจกต์แต่ละออบเจกต์ จะประกอบด้วยลักษณะ 2 อย่าง ดังนี้

1. รูปแบบของการกระทำที่ตอบสนองต่อเหตุการณ์ของแต่ละเหตุการณ์ที่เกิดขึ้นกับออบเจกต์นั้น ซึ่งรูปแบบของการกระทำนี้จะถูกกำหนด โดยโปรแกรมที่เขียนขึ้นเพื่อกำกับออบเจกต์นั้น และเรียก โปรแกรมที่เขียนเพื่อกำกับแต่ละออบเจกต์ว่าวิธีการดำเนินการ (Method)

2. คุณสมบัติ (Property) ของแต่ละออบเจกต์ได้แก่ ขนาด ความกว้าง ความสูง สี ลวดลาย ตำแหน่งของออบเจกต์นั้น และคุณสมบัติอื่น ๆ ซึ่งออบเจกต์ที่เราจะเห็นบนจอภาพจะถูกกำหนด โดยคุณสมบัติต่าง ๆ ซึ่งอาจจะแตกต่างกันไปบ้างในออบเจกต์แต่ละอย่างการกำหนดค่าต่าง ๆ ให้กับออบเจกต์นี้จะสามารถกำหนดไว้ในโปรแกรม หรือจะกำหนดขึ้นภายหลังจากที่

โปรแกรมเริ่มทำงานแล้วได้ โดยหากมีการกำหนดคุณสมบัติไว้ก่อน เมื่อโปรแกรมทำงานก็จะมี การนำเอาคุณสมบัติเหล่านี้มาใช้ในการแสดงผล

คุณสมบัติของออบเจกต์นี้เป็นสิ่งที่เราจะต้องทำความเข้าใจ เพราะจะต้องกำหนดคุณสมบัติแต่ละอย่างให้กับออบเจกต์แต่ละออบเจกต์อย่างละเอียดโดยในเคลฟ จะมีวินโดวส์หนึ่งที ออกแบบมาเพื่อใช้กำหนดคุณสมบัติโดยเฉพาะ และ วินโดวส์ทำให้การเขียนโปรแกรมแบบวิซวล ง่ายขึ้นมาก

รายการคำสั่งของเคลฟ

ในวินโดวส์เมนู จะทำหน้าที่เก็บคำสั่งที่สามารถใช้งานเอาไว้ทั้งหมด รวมทั้งเก็บคอนโทรลต่างๆ ที่จะใช้ในโปรแกรม นอกจากนั้นในวินโดวส์เมนู ยังมี Speed Bar ไว้ให้เรียกใช้คำสั่งที่ใช้งานบ่อยๆ อย่างรวดเร็ว ซึ่งหน้าที่ของแต่ละปุ่มของ Speed Bar มีรายละเอียดการทำงานของ แต่ละปุ่มดังนี้

- | | |
|------------------------------|-----------------------------------|
| - ปุ่ม Open Project | : เปิดไฟล์โปรเจก |
| - ปุ่ม Save Project | : เซฟไฟล์โปรเจก |
| - ปุ่ม Add File to Project | : เพิ่มไฟล์เข้าไปในโปรเจก |
| - ปุ่ม Open File | : เปิดไฟล์โปรแกรม |
| - ปุ่ม Save File | : เซฟไฟล์โปรแกรม |
| - ปุ่ม Remove File | : ลบไฟล์ออกจากโปรเจก |
| - ปุ่ม Select Unit from list | : เลือกคูนิตจากรายชื่อทั้งหมด |
| - ปุ่ม Toggle Form/Unit | : เลือกคูนิตจากรายชื่อทั้งหมด |
| - ปุ่ม New Form | : สร้างคูนิตใหม่ |
| - ปุ่ม Run | : สร้างฟอร์มใหม่ |
| - ปุ่ม Pause | : สั่งให้โปรแกรมทำงาน |
| - ปุ่ม Trace Into | : สั่งให้โปรแกรมหยุดทำงานชั่วคราว |
| - ปุ่ม Step Over | : ให้โปรแกรมทำงานทีละคำสั่ง |

ในการทำงานของเคลไพจะเป็นโปรแกรมแบบง่าย ๆ ที่ไม่ซับซ้อน แต่ก็มีมีความจำเป็น ต้องใช้ไฟล์หลายไฟล์ เพราะอย่างน้อยในแอปพลิเคชันหนึ่ง จะต้องมีโปรแกรม 1 โปรแกรม กับ ฟอรัม 1 ฟอรัม ดังนั้น การเขียนโปรแกรมภายใต้สภาพแวดล้อมแบบเคลไพจะต้องเริ่มดำเนินการ ทำงานโดยการสร้างไฟล์โปรเจก (Project) ก่อน จากนั้นจึงค่อยเพิ่มยูนิท หรือฟอรัมเข้าไปใน โปรเจกอีกครั้งหนึ่ง

ในการเขียนโปรแกรมโดยใช้เคลไพจะต้องทำบนฟอรัม และยูนิท ควบคู่กัน โดยฟอรัม จะใช้ในการออกแบบรูปแบบการแสดงผล หน้าตาของวินโดวส์ทั้งหมด ตามต้องการ โดยยูนิทจะ ทำหน้าที่เก็บโปรแกรมทั้งหมด ที่กำหนดการกระทำของออบเจกต์ทั้งหมดที่อยู่ในฟอรัมนั้น ที่จะ ตอบสนองกับเหตุการณ์ต่างๆ

ส่วนการควบคุม (Control) (VCP: Visual Component) มีอยู่มากถึง 72 ตัว ทาง บอร์แลนค์ออกแบบได้ดี โดยนำเอารูปแบบของการตั้งแท็บหรือชุดของออบเจกต์มาใช้ ทำให้ ประหยัดเนื้อที่ในการแสดงผลได้มาก โดยส่วนควบคุมทั้งหมด จะแบ่งออกเป็น 7 กลุ่ม คือ

- Standard ซึ่งเป็นส่วนควบคุมที่ใช้งานทั่วไป
- Additional เป็นส่วนควบคุมที่ใช้งานทั่วไปอีกกลุ่มหนึ่ง
- Data Access และ Data Control เป็นส่วนควบคุมที่ใช้ในการติดต่อกับฐานข้อมูล สำหรับ Dialogs นั้น เป็นฟอรัมมาตรฐานของกล่องสอบถาม (Dialogs Box) ที่ใช้ งานบ่อยๆ
- System เป็นส่วนควบคุมที่เกี่ยวข้องกับการใช้งานของระบบ
- VBX เป็นส่วนควบคุมที่เรียกใช้งานไฟล์ VBX

คอนโทรลแต่ละตัวของเคลไพจะบรรจุอยู่ในยูนิทใดยูนิทหนึ่ง ซึ่งไม่จำเป็นต้อง สอดคล้องกับการจัดกลุ่มในการตั้งแท็บ และเมื่อใดที่เรียกใช้ส่วนควบคุมนั้น ๆ โปรแกรมเคลไพ จะนำยูนิทของส่วนควบคุมนั้นรวมเข้ากับโปรเจกที่เขียนโดยอัตโนมัติ

วินโดวส์ฟอร์ม (Windows Form)

เป็นวินโดวส์ว่าง ๆ หรือเป็นตัวฟอร์มเปล่า ใช้สำหรับสร้างองค์ประกอบของระบบ โดยนำออบเจกต์มาใส่ในฟอร์ม หรือ เป็นวินโดวส์ของแอปพลิเคชันที่จะสร้าง โดยเมื่อเริ่มทำงาน โปรแกรมเคลฟจะมีฟอร์มเปล่านี้มาให้เสมอ

วินโดวส์ออบเจกต์อินสเปกเตอร์ (Windows Object Inspector)

เป็นวินโดวส์ที่ทำหน้าที่ 2 อย่าง คือ

1. เป็นวินโดวส์ที่จะแสดงคุณสมบัติทั้งหมดของออบเจกต์ที่ถูกเลือก โดยเมื่อเลือกที่ออบเจกต์ใดก็ตาม วินโดวส์นี้จะแสดงคุณสมบัติทั้งหมดของออบเจกต์นั้นออกมา และเมื่อเลือกที่ออบเจกต์อื่น คุณสมบัติในวินโดวส์จะเปลี่ยนแปลงตามไปด้วย
2. กำหนดการกระทำที่จะเกิดขึ้น เมื่อมีเหตุการณ์เกิดขึ้นกับออบเจกต์นั้น