



ภาควิชาครุศาสตร์วิศวกรรม  
 คณะครุศาสตร์อุตสาหกรรม  
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 ใบรับรองปริญญาโท

ชื่อหัวข้อ อีซีจีบนอินเทอร์เน็ต  
 ECG on Internet

ชื่อนักศึกษา 1. นางสาวทาริกา ศรีมงคล รหัสประจำตัว 43035608  
 2. นางสาวปิยาภรณ์ นาคลำภา รหัสประจำตัว 43035611

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา อิเล็กทรอนิกส์และคอมพิวเตอร์

อาจารย์ที่ปรึกษา อาจารย์ปิยะ จิตธรรมมาภิรมย์

อาจารย์ที่ปรึกษาร่วม อาจารย์อำพล ทองระอา

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อาจารย์ปิยะ จิตธรรมมาภิรมย์	
2. อาจารย์อำพล ทองระอา	
3. อาจารย์กิติพงศ์ มะโน	
4. อาจารย์อมรชัย ชัยชนะ	
5. อาจารย์สุระชัย พิมพ์สาตี	

วัน/เดือน/ปีที่สอบ วันศุกร์ที่ 22 พฤศจิกายน พ.ศ. 2544 เวลา 09.00 น.

สถานที่สอบ ห้อง ค.311 คณะครุศาสตร์อุตสาหกรรม สจล.



<BT4402122>  
 อีซีจีบนอินเทอร์เน็ต

ภาควิชารับรองแล้ว  
 ลงนาม.....

(ผศ.วิสุทธิ์ อธิพรธรรม)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 11 เดือน 8 พ.ศ. 2544



# ปริญญานิพนธ์

อีซีจีบนอินเทอร์เน็ต

ECG ON INTERNET



นางสาวทาริกา

ศรีมงคล

นางสาวปิยาภรณ์

นาคลำภา

เลขที่.....  
เลขทะเบียน 43154  
วัน, เดือน, ปี 23 ก.ค. 2545

b.....  
i.....

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์  
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ปริญญานิพนธ์

เรื่อง อีซีจีบนอินเทอร์เน็ต  
ECG On Internet

## วัตถุประสงค์

1. เพื่อศึกษาระบบจัดการฐานข้อมูล
2. เพื่อศึกษาการเขียนเว็บเพจด้วย ASP
3. เพื่อออกแบบระบบจัดการฐานข้อมูล
4. สามารถออกแบบให้ระบบจัดการฐานข้อมูลเชื่อมโยงกับระบบอินเทอร์เน็ตได้
5. เพื่อสามารถใช้งานระบบจัดการฐานข้อมูลบนระบบอินเทอร์เน็ตได้

## ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถใช้งานระบบจัดการฐานข้อมูลได้
2. สามารถทำงานบนระบบอินเทอร์เน็ตได้
3. ได้แบบระบบจัดการฐานข้อมูล
4. ได้แบบระบบจัดการฐานข้อมูลที่เชื่อมโยงกับระบบอินเทอร์เน็ตได้
5. สามารถนำระบบจัดการฐานข้อมูลบนระบบอินเทอร์เน็ตไปใช้งานได้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	อีซีจีบนอินเทอร์เน็ต	
	นางสาวทาริกา	ศรีมงคล
	นางสาวปิยาภรณ์	นาคลำภา
อาจารย์ที่ปรึกษา	อาจารย์ปิยะ	จิตรธรรมมาภิรมย์
อาจารย์ที่ปรึกษาร่วม	อาจารย์อำพล	ทองระอา
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต	
สาขาวิชา	อิเล็กทรอนิกส์และคอมพิวเตอร์	
ปีการศึกษา	2544	

### บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้เสนอการออกแบบและการสร้าง ECG On Internet ซึ่งเป็นเว็บเพจที่รวบรวมข้อมูลของอุปกรณ์อิเล็กทรอนิกส์ชนิดต่าง ๆ ในหนังสือ ECG Replacement Guide โดยใช้ Microsoft Access และ ASP เป็นตัวเก็บข้อมูลและแสดงผลของข้อมูล ซึ่งผู้ใช้สามารถใช้เว็บเพจนี้ในการค้นหาคุณสมบัติและอุปกรณ์ทดแทนของอุปกรณ์ทางอิเล็กทรอนิกส์ได้

## II

<b>Thesis Title</b>	ECG On Internet
<b>Students</b>	Miss Tarika Seemongkol Miss Piyaporn Naklumpa
<b>Advisor</b>	Mr.Piya Jitthamapirom
<b>Co-Advisor</b>	Mr.Amphon Thongra-ar
<b>Education Level</b>	Bachelor of Science in Industrial Education
<b>Program in</b>	Electronic and Computer
<b>Academic Year</b>	2001

### ABSTRACT

The thesis presents design and construction of the ECG On Internet, which is an web page. It shows detail of electronic devices in ECG Replacement Guide book . It uses MS Access and ASP to be an engine to store and display information Results. Users can use the web page to find characteristics of designed devices and to find the Replacement devices .

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ลุล่วงไปด้วยดี เนื่องจากความร่วมมือของสมาชิกภายในกลุ่มและเพื่อน  
ในห้อง ขอขอบคุณคณาจารย์ภาควิชาครุศาสตร์วิศวกรรมทุกท่านที่ให้ความอนุเคราะห์เครื่องมือ และคำ  
แนะนำ แนวความคิด ความรู้ต่าง ๆ แนวทางการแก้ไขปัญหา ในการจัดปริญญานิพนธ์ สุดท้ายที่ควร  
ระลึกถึงอย่างยิ่ง บิดา และมารดาที่เป็นผู้ให้การสนับสนุนด้านการศึกษา และเป็นผู้ให้กำลังใจด้วยดี  
ตลอดมา ตั้งแต่อดีตจนปัจจุบัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์	2
1.2 จีัดความสามารถของโครงการ	2
1.3 เนื้อหาพอสั่งเขป	2
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 บทนำ	4
2.2 ประวัติเป็นมาของการจัดการฐานข้อมูล	4
2.3 ความรู้พื้นฐานเกี่ยวกับฐานข้อมูลฯ	5
2.3.1 คำศัพท์พื้นฐาน	5
2.3.2 Entity, Attribute และความสัมพันธ์	6
2.3.3 รีเลชัน	8
2.3.4 ฐานข้อมูล	8
2.3.5 ระบบการจัดเก็บฐานข้อมูล	9
2.3.6 โมเดล (Model)	10
2.4 ประโยชน์จากการประมวลผลด้วยฐานข้อมูล	13
2.5 สถาปัตยกรรมของระบบฐานข้อมูลฟังก์ชันการขึ้นต่อกัน, คีย์และนอร์มัลไลซ์	14
2.5.1 ความเป็นอิสระของข้อมูล	15
2.5.2 Instance และ Database Schema	15
2.5.3 ระบบการจัดการฐานข้อมูล	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.5.4 ผู้บริหารฐานข้อมูล	16
2.5.5 ผู้ใช้ระบบ	17
2.5.6 การจัดเก็บข้อมูล	17
2.5.7 ภาษาสำหรับนิยามข้อมูล	20
2.5.8 ภาษาสำหรับการใช้ข้อมูล	20
2.6 ฟังก์ชันการขึ้นต่อกัน, คีย์และนอร์มัลไลซ์	21
2.6.1 กฎของความคงสภาพ (Integrity Rule)	21
2.6.2 ฟังก์ชันการขึ้นต่อกัน	22
2.6.3 คีย์	22
2.6.4 รูปแบบนอร์มัลระดับที่ 1, 2 และ 3	23
2.7 การออกแบบฐานข้อมูล	26
2.7.1 เป้าหมายของการออกแบบฐานข้อมูล	27
2.7.2 ความต้องการของผู้ใช้แต่ละคน	28
2.7.3 หลักการพื้นฐานของการออกแบบฐานข้อมูล	29
2.8 ภาษา สคริปต์ ASP (Active Server Page )	30
2.8.1 การเขียน Script (ActiveX Server Scripting )	31
2.8.2 การสร้างออบเจกต์ต่างๆ (ActiveX Server Component )	31
2.8.3 ความสามารถของ Script	33
2.8.4 โครงสร้างของ VBScript	33
2.8.5 ข้อดีของ ASP	35
2.9 การสร้างฐานข้อมูลด้วย Microft Access	35
2.9.1 ส่วนประกอบในฐานข้อมูล Access	36
2.9.2 การติดต่อเชื่อมโยงโดยใช้ ODBC Data Sources	37

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.10 วิชาลอินเทอร์เคฟ (Visual Interdev)	37
2.10.1 การเตรียมตัวกับวิชาลอินเทอร์เคฟ	37
2.10.2 การพัฒนาเว็บเพจโดยใช้วิชาลอินเทอร์เคฟ	38
บทที่ 3 การออกแบบการสร้างและการทำงาน	40
3.1 การสร้างฐานข้อมูลต่างๆ	40
3.2 การออกแบบโฮมเพจด้วยภาษา HTML	51
3.3 การการทำงานเชื่อมต่อฐานข้อมูลกับโฮมเพจโดยใช้ภาษา	55
บทที่ 4 ผลการทดลอง	59
1. Search By No.	60
2. Search By Property	62
3. Replacement devices	64
4. Guest Book	66
5. Testing	67
6. Code	67
7. Symbols	68
8. About me	68
บทที่ 5 บทสรุป ปัญหาแนวทางแก้ไขและการพัฒนา	69
5.1 บทสรุป	69
5.2 ปัญหาและแนวทางแก้ไข	69
5.3 แนวการพัฒนา	69
บรรณานุกรม	70
ประวัติผู้แต่ง	71

## สารบัญตาราง

ตาราง		หน้า
ตารางที่ 2.1	การสร้างความสัมพันธ์แบบหนึ่งต่อหนึ่งโดยใช้ตารางเดียว	18
ตารางที่ 2.2	การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม	18
ตารางที่ 2.3	การสร้างความสัมพันธ์แบบกลุ่มต่อกลุ่ม	19
ตารางที่ 2.4	ตัวอย่างรีเลชัน 1 NF	24
ตารางที่ 2.5	รีเลชัน 2 NF	25
ตารางที่ 2.6	ออบเจกต์ต่างๆ ใน ASP	32
ตารางที่ 2.7	ชนิดของตัวแปร	34
ตารางที่ 2.8	เปรียบเทียบการติดตั้งเว็บเซิร์ฟเวอร์ในระบบต่างๆ	38



## สารบัญรูป

รูป	หน้า
รูปที่ 2.1 ตัวอย่าง Entity และ Attribute	7
รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรคู่แสดงความเป็นกลุ่มแผนกวิชา	7
รูปที่ 2.3 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรเดี่ยวแสดงความเป็นกลุ่มแผนกวิชา	8
รูปที่ 2.4 การจักระบบไฟล์	8
รูปที่ 2.5 การใช้ระบบฐานข้อมูล	9
รูปที่ 2.6 การใช้ระบบฐานข้อมูลด้วย DBMS	9
รูปที่ 2.7 โครงสร้างฐานข้อมูลแบบเน็ตเวิร์ค	11
รูปที่ 2.8 ลักษณะของการเก็บข้อมูล	12
รูปที่ 2.9 ระดับข้อมูล 3 ระดับ	15
รูปที่ 2.10 ขั้นตอนการออกแบบฐานข้อมูล	28
รูปที่ 3.1 ชื่อฐานข้อมูลทั้งหมดที่ใช้ติดต่อกับโฮมเพจ	40
รูปที่ 3.2 รายละเอียดของ ตาราง NUMBER	41
รูปที่ 3.3 รายละเอียดของ ตาราง PRODUCK_MAIN1	41
รูปที่ 3.4 รายละเอียดของ ตาราง Zener_Diodes	42
รูปที่ 3.5 ความสัมพันธ์ระหว่างตาราง NUMBER และ ตารางPRODUCK_MAIN1	43
รูปที่ 3.6 ความสัมพันธ์ระหว่าง ตาราง NUMBER และ ตาราง NUMBER	43
รูปที่ 3.7 ความสัมพันธ์ของ ตาราง Zener_Diodes กับตาราง NUMBER	44
รูปที่ 3.8 โปรแกรม Macromedia Dreamweaver 4	44
รูปที่ 3.9 Apply Templates to Page	45
รูปที่ 3.10 หน้าตาของ Templates	45
รูปที่ 3.11 ส่วนของ Tag Body	46
รูปที่ 3.12 มีจำนวนไฟล์ทั้งหมด	46
รูปที่ 3.13 โปรแกรม Microsoft Visual InterDev 6.0	47
รูปที่ 3.14 การสร้าง SQL ใน โปรแกรม Microsoft Visual InterDev 6.0	47

## สารบัญรูป(ต่อ)

รูป	หน้า
รูปที่ 3.15 การนำเข้าตาราง	48
รูปที่ 3.16 การเปลี่ยนชื่อ Field Name	48
รูปที่ 3.17 การเลือก ID	49
รูปที่ 3.18 การใส่ชื่อตาราง	49
รูปที่ 3.19 ข้อมูลที่ได้จากการนำเข้า	50
รูปที่ 3.20 ข้อมูลในตาราง	50
รูปที่ 3.21 โปรแกรม Macromedia Dreamweaver 4	51
รูปที่ 3.22 Apply Templates to Page	52
รูปที่ 3.23 หน้าตาของ Templates	52
รูปที่ 3.24 ส่วนของ Tag Body	53
รูปที่ 3.25 มีจำนวนไฟล์ทั้งหมด	54
รูปที่ 3.26 โปรแกรม Microsoft Visual InterDev 6.0	55
รูปที่ 3.27 การสร้าง SQL ในโปรแกรม Microsoft Visual InterDev 6.0	56
รูปที่ 4.1 เมนูการทำงาน	59
รูปที่ 4.2 การค้นหาจาก Search By No.	60
รูปที่ 4.3 การเลือก Search จาก Number	60
รูปที่ 4.4 ผลลัพธ์ของการ Search จาก Number	61
รูปที่ 4.5 การเลือก Search จาก ECG Number	61
รูปที่ 4.6 ผลลัพธ์ของการ Search จาก ECG Number	62
รูปที่ 4.7 การเลือก Search จาก Property	62
รูปที่ 4.8 การเลือก Search จาก Zener Diodes	63

## สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 4.9 ผลลัพธ์ของการ Search	63
รูปที่ 4.10 การเลือก Search จาก Number ของ Replacement devices	64
รูปที่ 4.11 ผลลัพธ์ของการ Search จาก Number	64
รูปที่ 4.12 การเลือก Search จาก ECG Number ของ Replacement devices	65
รูปที่ 4.13 ผลลัพธ์ของการ Search จาก ECG Number	65
รูปที่ 4.14 การทำงานของ Guest Book	66
รูปที่ 4.15 การทำงานของ Testing	66
รูปที่ 4.16 การทำงานของ Code	67
รูปที่ 4.17 การทำงานของ Symbols	67
รูปที่ 4.18 การทำงานของ About me	68

### 1.3 เนื้อหาพอสังเขป

เนื้อหาภายในปฏิญานิพนธ์ฉบับนี้ ได้จัดแบ่งออกเป็นบทต่างๆ ทั้งหมดด้วยกัน 5 บท เพื่อความสะดวกในการศึกษา และทำความเข้าใจ โดยในแต่ละบทนั้นจะประกอบไปด้วยเนื้อหาที่สำคัญๆ ดังนี้

บทที่ 1 บทนำ ในส่วนของบทนี้จะกล่าวถึงความสำคัญและที่มาของปฏิญานิพนธ์ฉบับนี้ รวมถึงความจำเป็นในการจัดการฐานข้อมูล ตลอดจนลักษณะโดยทั่วไป ไปของปฏิญานิพนธ์อย่างกว้างๆ เพื่อให้ผู้ที่ศึกษา ได้ทราบถึงลักษณะทั่วไป และส่วนประกอบของเนื้อหาสาระที่สำคัญภายในปฏิญานิพนธ์นี้

บทที่ 2 ทฤษฎี และหลักการ ในส่วนของบทที่ 2 นี้จะประกอบไปด้วยเนื้อหาทางทฤษฎี และหลักการต่างๆ ที่เกี่ยวข้อง เช่น ประโยชน์จากการประมวลผลด้วยฐานข้อมูล ทฤษฎีและการออกแบบฐานข้อมูล การเขียนโฮมเพจด้วย ASP การสร้างฐานข้อมูลด้วย Microsoft Access และการใช้งาน Visual Interdev

บทที่ 3 การออกแบบการสร้างและการทำงาน ในส่วนของบทที่ 3 นี้ จะกล่าวถึงหลักการออกแบบการสร้างและการทำงาน ลำดับขั้นตอนของการออกแบบ รวมไปถึงขั้นตอนการสร้าง โฮมเพจและฐานข้อมูลในส่วนต่างๆ

บทที่ 4 การทดลองและผลการทดลอง ในส่วนของบทที่ 4 นี้ จะกล่าวถึงการใช้งาน และการทดลองต่าง ๆ เช่น การทดลองค้นหาข้อมูล

บทที่ 5 บทสรุป ปัญหาแนวทางแก้ไข และการพัฒนา ในบทที่ 5 นี้จะกล่าวถึง ปัญหาที่เกิดขึ้นในการระหว่างทำโครงการ และแนวทางในการแก้ปัญหาที่เกิดขึ้น รวมไปถึงแนวทางในการที่จะพัฒนาตัวโปรแกรม

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 บทนำ

ในการบริหารงานด้านต่าง ๆ ไม่ว่าจะเป็นงานด้านธุรกิจ หรืองานด้านการศึกษา มีความจำเป็นอย่างยิ่งที่จะต้องเกี่ยวข้องกับงานข้อมูล การจัดเก็บข้อมูล ในรูปแบบต่าง ๆ ไม่ว่าจะเป็นเอกสารหรือบันทึก ข้อมูลบางอย่างมีแบบฟอร์มเป็นมาตรฐาน ซึ่งอาจจัดเก็บไว้ในรูปแบบที่สามารถค้นหาได้ง่าย การจัดเก็บโดยวิธีปกติอาจจะสามารถใช้ได้ในงานที่มีขนาดเล็กเท่านั้น แต่สำหรับระบบงานขนาดใหญ่ ที่มีข้อมูลที่จะจัดเก็บ และมีความซับซ้อนของงานมาก ๆ การนำเครื่องคอมพิวเตอร์มาใช้จะช่วยอำนวยความสะดวกได้เป็นอย่างดี และช่วยให้การบริหารงานระบบข้อมูลเป็นไปอย่างมีประสิทธิภาพมากยิ่งขึ้น

#### 2.2 ประวัติความเป็นมาของการจัดการฐานข้อมูล

ความจริงแล้วนับเป็นเรื่องยากทีเดียวที่จะกล่าวลงไปอย่างแน่ชัดว่า ระบบฐานข้อมูลได้ถือกำเนิดขึ้นมาเมื่อใด แต่มีเหตุผลที่น่าเชื่อถือได้ว่า ต้นกำเนิดของระบบฐานข้อมูลเกิดขึ้นจากโครงการอพอลโลของสหรัฐอเมริกา อันเป็นโครงการส่งมนุษย์อวกาศไปดวงจันทร์ในช่วงเวลา 20 – 30 ปี ที่แล้ว แน่แน่นอนว่าข้อมูลที่ใช้ในงานดังกล่าวจะต้องมีจำนวนมากทีเดียว การจัดระบบข้อมูลในโครงการนี้เกิดจากการว่าจ้างบริษัทไอบีเอ็ม ให้พัฒนาระบบการดูแลข้อมูลขึ้นมาอันได้แก่ ระบบ GUAM (Generalized Update Access Method) ซึ่งเราต้องถือเป็นต้นกำเนิดของระบบการจัดการฐานข้อมูล บริษัทไอบีเอ็มพบว่าการทำงานของระบบ GUAM ไม่เพียงแต่จะมีประโยชน์เฉพาะกับโครงการอวกาศเท่านั้น แต่ยังมีประโยชน์ในการประมวลผลในงานธุรกิจอื่น ๆ ดังนั้นในอีก 2 ปีต่อมา ไอบีเอ็มได้พัฒนาการจัดการข้อมูลขึ้นมาใหม่เพื่อการใช้งานในวงการธุรกิจทั่ว ๆ ไป อันได้แก่ระบบ DLI (Data Language/I) จากนั้น ไอบีเอ็มก็ได้เสริมสร้าง DLI เพิ่มเติมขึ้น และในที่สุดก็ได้ระบบ IMS (Information Management System) ซึ่งใช้กันมาจนถึงปัจจุบัน

นอกจากบริษัทไอบีเอ็ม แล้วยังมีบริษัท GE (General Electric) ที่มีส่วนในการพัฒนาระบบฐานข้อมูล ได้แก่ระบบ IDS (Integrated Data Store) ระบบนี้ได้เริ่มใช้ในช่วงปี พ.ศ. 2509 และเป็นต้นกำเนิดของระบบโคดาซิล (CODASYL) หรือโมเดลแบบเน็ตเวิร์คที่ยังนิยมใช้กันแพร่หลายจนถึงปัจจุบัน ระบบโคดาซิล เกิดขึ้น ตั้งแต่ พ.ศ. 2517 โดยคณะกรรมการชุดหนึ่งอันเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทีมเดียวกับชุดที่พัฒนาภาษาโคบอล และได้ร่างข้อกำหนดมาตรฐานสำหรับโคดาชีวิตขึ้นสำเร็จเมื่อปี พ.ศ. 2515

ในช่วงเวลาใกล้เคียงกับที่โคดาชีวิตฉบับมาตรฐาน ได้ถูกกำหนดขึ้น ดร.คอดด์ (E.F. Codd) ได้เสนอผลงานทางวิชาการเกี่ยวกับโมเดลใหม่อีกโมเดลหนึ่ง อันได้แก่โมเดลเชิงสัมพันธ์ หลักการของ ดร.คอดด์ ได้รับความสนใจจากนักวิจัยและบริษัทผู้ค้าเครื่องคอมพิวเตอร์เป็นอย่างมาก และผลงานที่มีความสำคัญยิ่งต่อการค้าเบสที่พัฒนาขึ้นตามแนวความคิดของ ดร.คอดด์ ได้แก่ระบบ R ซึ่งเป็นระบบที่สร้างขึ้นโดยบริษัทไอบีเอ็ม แต่กว่าที่โมเดลเชิงสัมพันธ์จะได้ก้าวเข้าสู่วงการธุรกิจก็ใช้เวลาถึง 10 ปี และระบบ R ก็เป็นเพียงระบบต้นแบบที่ยังคงใช้โดยเฉพาะในห้องวิจัยเท่านั้น โดยไอบีเอ็มได้สร้างระบบ DB2 ขึ้นมาแทน เพื่อนำออกสู่งานธุรกิจ

ในช่วง พ.ศ. 2525 เป็นต้นมา ถือเป็นยุคทองของระบบฐานข้อมูลที่ได้ก้าวเข้าสู่ตลาดคอมพิวเตอร์อย่างเต็มตัว จนกระทั่งในปัจจุบัน ก็ได้มีซอฟต์แวร์ที่เกิดขึ้นมาเกื้อหนุนการทำงานของระบบการจัดการฐานข้อมูลอีกมากมาย ไม่ว่าจะเป็นระบบพจนานุกรมข้อมูล ซอฟต์แวร์ที่ช่วยในการออกแบบและสร้างรายงาน และภาษาเรียกค้น เป็นต้น

การเจริญเติบโตของระบบฐานข้อมูลยังคงไม่หยุดนิ่ง พร้อมกับเทคโนโลยีคอมพิวเตอร์รุดหน้าไปอย่างรวดเร็ว เครื่องไม้เครื่องมือทางซอฟต์แวร์ที่จะช่วยในการประมวลผลข้อมูลก็เกิดขึ้นเรื่อยๆ ไม่ว่าจะเป็นภาษารุ่นที่ 4 หรือที่นับว่าใหม่ที่สุดในขณะนี้ก็คือระบบ CASE (Computer Aided Software Engineering) และการผนวกเอาระบบฐานข้อมูลรวมกับระบบฐานรอบรู้ เป็นต้น

## 2.3 ความรู้พื้นฐานเกี่ยวกับระบบฐานข้อมูล

ในหัวข้อนี้ จะกล่าวถึงความรู้พื้นฐานที่ควรทราบ และแนะนำให้รู้จักคำศัพท์ที่เกี่ยวข้องกับระบบฐานข้อมูล ซึ่งในหลายส่วนอาจเป็นศัพท์ที่คุ้นเคยกันแล้ว

### 2.3.1 คำศัพท์พื้นฐาน

- 1) ข้อมูล (Data) หมายถึง ข้อเท็จจริงขั้นต้น ซึ่งอาจจะเรียกว่าเป็นวัตถุดิบของสารสนเทศ (Information) เมื่อข้อมูลถูกนำมาประมวลผล (เรียงลำดับ, แยกประเภท, เชื่อมโยง, คำนวณหรือสรุปผล) และจัดให้อยู่ในรูปแบบที่นำมาใช้ประโยชน์ได้ เราจึงเรียกว่าเป็น สารสนเทศ
- 2) บิต (Bit) หมายถึง หน่วยของข้อมูลที่มีขนาดเล็กที่สุด
- 3) ไบต์ (Byte) หมายถึง หน่วยของข้อมูลที่เกิดจากการนำบิตหลายๆ บิตมารวมกันเป็นตัวอักษร

4) ฟیلด์ (Field) หมายถึง หน่วยของข้อมูลที่ประกอบด้วยตัวอักษรหลายๆ ตัวอักษร เพื่อแทนความหมายของสิ่งๆ หนึ่ง

5) เรคคอร์ด (Record) หมายถึง หน่วยของข้อมูลที่เกิดจาก การนำเอาฟิลด์หลายๆฟิลด์ มา รวมกัน เพื่อแสดงรายละเอียดของข้อมูลเรื่องใดเรื่องหนึ่ง

6) เพิ่มข้อมูล (File) หมายถึง หน่วยของข้อมูลที่เกิดจากการนำเอาเรคคอร์ด หลายๆ เรคคอร์ดมารวมกันสำหรับในระบบฐานข้อมูล

7) ฐานข้อมูล (Database) คือ โครงสร้างของสารสนเทศ (Information) ที่ประกอบด้วย Entity หลายๆตัว ซึ่ง Entity เหล่านี้จะมีความสัมพันธ์กัน

8) คีย์หลัก (Primary Key) คือ ฟิลด์ใดๆ ในตารางข้อมูลที่มีข้อมูลในฟิลด์ไม่ซ้ำกันเลยในแต่ละเรคคอร์ด

### 2.3.2 Entity, Attribute และความสัมพัทธ์

Entity ได้แก่ สิ่งต่างๆ ที่สามารถระบุได้ในความเป็นจริงซึ่งอาจจะเป็นสิ่งที่จับต้องได้ หรือ อาจเป็นเพียงสิ่งที่อยู่ในรูปนามธรรมที่ไม่สามารถจับต้องได้ ซึ่งเมื่อนำแต่ละ Entity มารวมกันภายใต้คุณลักษณะใดคุณลักษณะหนึ่งเหมือนกันแล้ว Entity เหล่านั้นจะถูกเรียกว่า “Entity Set” จะแบ่งออกเป็น 2 ประเภทคือ

1) Regular Entity หรือบางครั้งเรียกว่า Strong Entity ได้แก่ Entity ส่วนใหญ่ที่ปรากฏอยู่ในระบบฐานข้อมูลซึ่งภายใน Entity ประเภทนี้สามารถมีคุณสมบัติ Identity ได้ด้วยตัวมันเอง

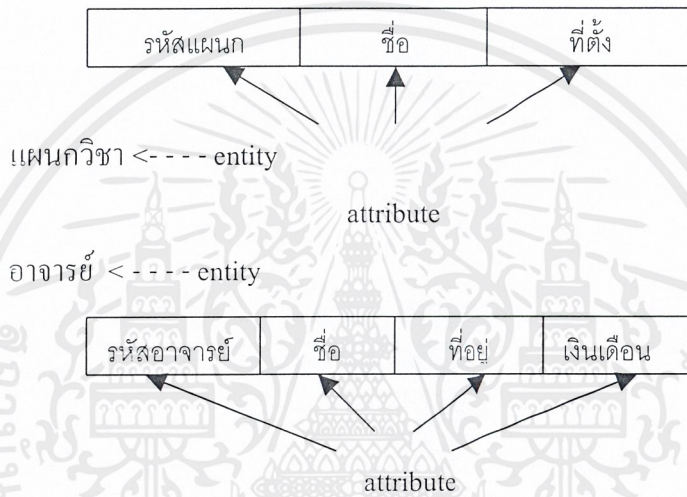
2) Weak Entity เป็น Entity ที่มีลักษณะตรงข้ามกับ Regular Entity กล่าวคือ สมาชิกของ Entity ประเภทนี้จะสามารถมีคุณสมบัติ Identity ได้ จะต้องอาศัย Property หนึ่งของ Regular Entity มาประกอบกับ Property ของตัวมันเอง

ซึ่ง Entity ในระบบฐานข้อมูลก็ได้แก่ อาจารย์ แผนกวิชา แผนกการประกันสุขภาพ ประวัติการทำงาน หรือถ้าเรากำลังสนใจในการสร้างระบบฐานข้อมูล เกี่ยวกับระบบการขายของบริษัทแห่งหนึ่ง Entity ของระบบนี้ก็จะได้แก่ พนักงานขาย ลูกค้า การสั่งซื้อ และสินค้า เป็นต้น

Attribute ก็คือ ข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เช่น Attribute ของบุคคลก็อาจจะได้แก่ สีมผม สีตา อายุ เพศ ชื่อ ฯลฯ และถ้าจะออกแบบ Attribute ของอาจารย์ที่เรา สนใจได้แก่ รหัสอาจารย์ ชื่อ ที่อยู่ เงินเดือนและอื่นๆ ดังที่แสดงในรูปที่ 2.1 เป็น Entity 2 ตัว ได้แก่อาจารย์ และแผนกวิชา โดยที่แผนกวิชาเป็น Entity ที่ประกอบด้วย Attribute 3 ตัว คือ รหัสแผนก ชื่อแผนก และสถานที่ตั้ง ส่วนอาจารย์ก็เป็น Entity ที่ประกอบด้วย Attribute 4 ตัว คือรหัสอาจารย์ ชื่อ ที่อยู่ และเงินเดือน

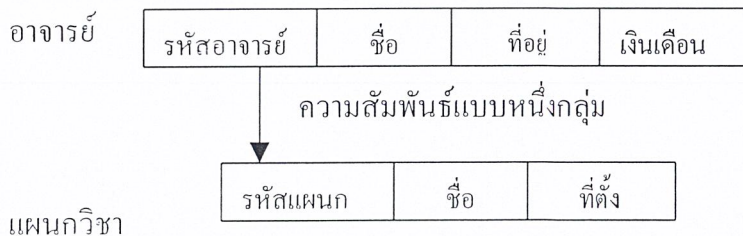
ส่วนความสัมพันธ์นั้นหมายถึง ความสัมพันธ์ระหว่าง Entity ตัวอย่างเช่น ความสัมพันธ์ระหว่างแผนกวิชาและอาจารย์ ก็เป็นในลักษณะที่ว่าแผนกวิชาที่อาจารย์นั้นสังกัดอยู่ ส่วนความสัมพันธ์ระหว่างอาจารย์และแผนกวิชา เป็นในลักษณะที่ว่าอาจารย์ที่ทำงานอยู่กับแผนกวิชานั้นๆ

ในรูปที่ 2.1 เป็นการแสดงความสัมพันธ์ระหว่าง Entity ทั้งสองนี้โดยใช้หัวลูกศรแสดงความสัมพันธ์ ซึ่งในรูปนี้ความสัมพันธ์จากแผนกวิชาไปสู่อาจารย์เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-To-Many) กล่าวคือ ในแผนกวิชา 1 แผนกจะมีอาจารย์สังกัดอยู่ได้มากกว่า 1 คน



รูปที่ 2.1 ตัวอย่าง Entity และ Attribute

ในทางกลับกันความสัมพันธ์จากอาจารย์ไปแผนกวิชาเป็นลักษณะแบบกลุ่มต่อหนึ่ง (Many-To-one) ทั้งนี้ก็เพราะมีอาจารย์มากกว่า 1 คน ทำงานอยู่ในแผนกวิชา 1 แผนก ขอให้สังเกตหัวลูกศรที่แสดงความสัมพันธ์แบบนี้ในรูปที่ 2.2 จะเห็นว่า หัวลูกศรเดียวแสดงความเป็น “หนึ่ง” และหัวลูกศร 2 หัว แสดงความเป็น “กลุ่ม”



รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรคู่แสดงความเป็นกลุ่มแผนกวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจาก 1 แผนกวิชาไปยังอาจารย์หลายคน หรือในบางครั้งเราอาจจะใช้หัวลูกศรเดียว แสดงความเป็นกลุ่ม และไม่มีหัวลูกศรแสดงความเป็น 1 ก็ได้ ดังเช่นที่แสดงในรูปที่ 2.3



รูปที่ 2.3 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรเดียว

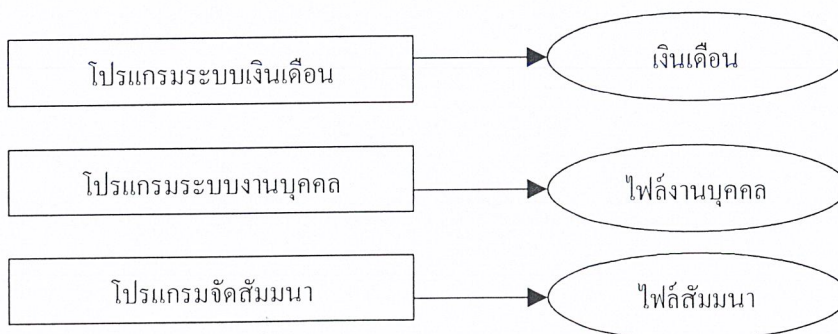
### 2.3.3 รีเลชัน (Relation)

รีเลชัน คือ ตาราง 2 มิติ ที่มีลักษณะดังนี้

- 1) แต่ละช่องของตารางจะบรรจุข้อมูลเพียงค่าเดียว
- 2) ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน ได้แก่ชื่อของ Attribute
- 3) ค่าข้อมูลที่อยู่ในแต่ละคอลัมน์ ได้แก่ค่าของ Attribute ที่ระบุไว้ในหัวข้อคอลัมน์นั้นๆ
- 4) การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
- 5) ข้อมูลในแต่ละแถวจะต้องแตกต่างกัน
- 6) การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ

### 2.3.4 ฐานข้อมูล (Database)

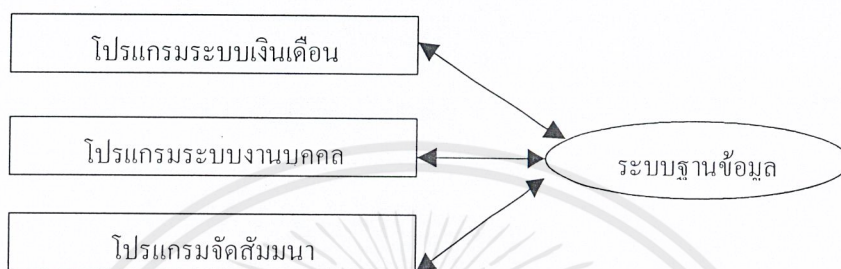
ในการประมวลผลไฟล์ต่างๆ ไปนั้น ผู้ใช้แต่ละคนจะมีไฟล์เฉพาะงาน และโปรแกรมที่เขียนขึ้นมา เพื่อดึงเอาข้อมูลจากไฟล์ส่วนตัวมาใช้งานตามต้องการดังรูปที่ 2.4 การที่ผู้ใช้แต่ละกลุ่มต่างก็เก็บข้อมูลไว้ในไฟล์เช่นนี้ ก่อให้เกิดปัญหาที่เด่นชัด นั่นก็คือเกิดการซ้ำซ้อนของข้อมูลขึ้น



รูปที่ 2.4 การจัดระบบไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

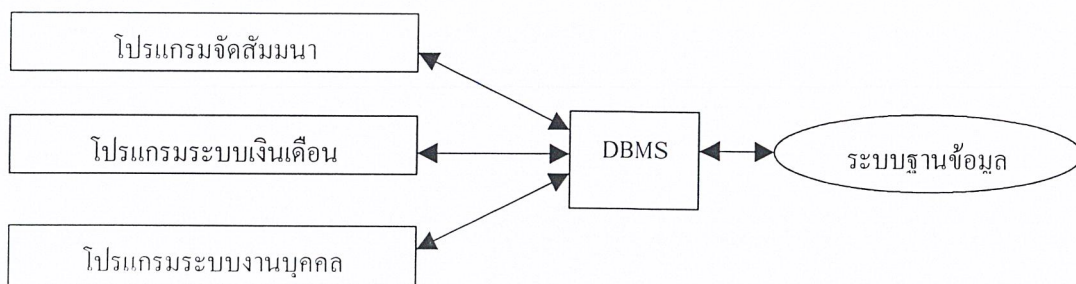
นอกจากจะทำให้เกิดความสับสนเปลืองเนื้อที่ในการจัดเก็บข้อมูลแล้ว ยังก่อให้เกิดปัญหาขึ้นเมื่อมีการเปลี่ยนแปลงค่าข้อมูล เพราะอาจจะหลงลืมเปลี่ยนข้อมูลในบางไฟล์ ทำให้ค่าข้อมูลเดียวกันที่เก็บในแต่ละไฟล์มีค่าไม่ตรงกัน ปัญหาดังกล่าว ทำให้เกิดความคิดที่ว่าจะนำข้อมูลเหล่านี้มาเก็บรวบรวมไว้ในที่เดียวกันหรือจะเก็บไว้ในฐานข้อมูลแทนที่จะเก็บไว้ในไฟล์ ดังรูปที่ 2.5



รูปที่ 2.5 การใช้ระบบฐานข้อมูล

### 2.3.5 ระบบการจัดเก็บฐานข้อมูล

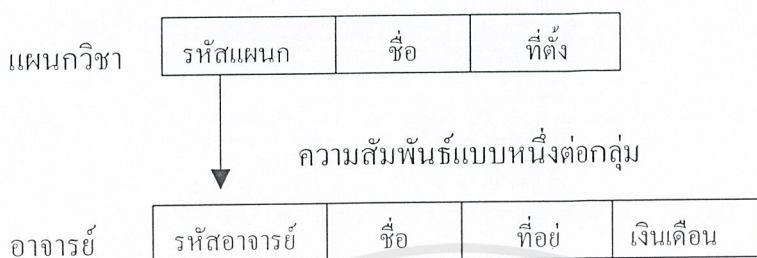
การควบคุมดูแลการใช้ฐานข้อมูลนั้นเป็นเรื่องที่ยุ่ยากกว่าการใช้ไฟล์มาก เพราะว่าเราจะต้องตัดสินใจว่าโครงสร้างในการเก็บข้อมูลควรจะเป็นเช่นไร และการเขียนโปรแกรมเพื่อสร้างและเรียกใช้ข้อมูลจากโครงสร้างไฟล์เหล่านี้ ก็เป็นเรื่องที่ยุ่ยากพอๆ และถ้าเกิดโปรแกรมเหล่านี้ทำงานผิดพลาดขึ้นมา ก็จะทำให้เกิดความเสียหายต่อโครงสร้างของข้อมูลทั้งหมดได้ เพื่อเป็นการลดภาระการทำงานของผู้ใช้ จึงได้มีซอฟต์แวร์ขึ้นมาตัวหนึ่งมีชื่อเรียกว่า ระบบการจัดการฐานข้อมูล หรือ DBMS ซึ่งจะทำหน้าที่ในการควบคุมดูแลการสร้างและเรียกใช้งานฐานข้อมูล โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล เปรียบเสมือนว่า DBMS นี้จะเป็นสื่อกลางระหว่างผู้ใช้ และ โปรแกรมต่างๆที่เกี่ยวข้องกับการใช้ฐานข้อมูลดังรูปที่ 2.6



รูปที่ 2.6 การใช้ระบบฐานข้อมูลด้วย DBMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจาก 1 แผนกวิชาไปยังอาจารย์หลายคน หรือในบางครั้งเราอาจจะใช้หัวลูกศรเดียว แสดงความเป็นกลุ่ม และไม่มีหัวลูกศรแสดงความเป็น 1 ก็ได้ ดังเช่นที่แสดงในรูปที่ 2.3



รูปที่ 2.3 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรเดียว

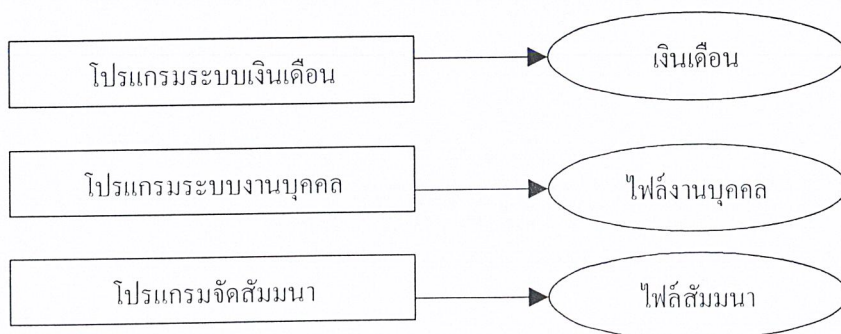
### 2.3.3 รีเลชัน (Relation)

รีเลชัน คือ ตาราง 2 มิติ ที่มีลักษณะดังนี้

- 1) แต่ละช่องของตารางจะบรรจุข้อมูลเพียงค่าเดียว
- 2) ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน ได้แก่ชื่อของ Attribute
- 3) ค่าข้อมูลที่อยู่ในแต่ละคอลัมน์ ได้แก่ค่าของ Attribute ที่ระบุไว้ในหัวข้อคอลัมน์นั้นๆ
- 4) การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
- 5) ข้อมูลในแต่ละแถวจะต้องแตกต่างกัน
- 6) การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ

### 2.3.4 ฐานข้อมูล (Database)

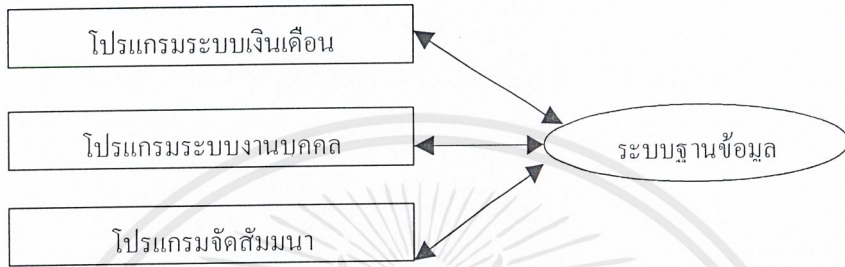
ในการประมวลผลไฟล์ต่างๆ ไปนั้น ผู้ใช้แต่ละคนจะมีไฟล์เฉพาะงาน และโปรแกรมที่เขียนขึ้นมา เพื่อดึงเอาข้อมูลจากไฟล์ส่วนตัวมาใช้งานตามต้องการดังรูปที่ 2.4 การที่ผู้ใช้แต่ละกลุ่มต่างก็เก็บข้อมูลไว้ในไฟล์เช่นนี้ ก่อให้เกิดปัญหาที่เด่นชัด นั่นก็คือเกิดการซ้ำซ้อนของข้อมูลขึ้น



รูปที่ 2.4 การจัดระบบไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

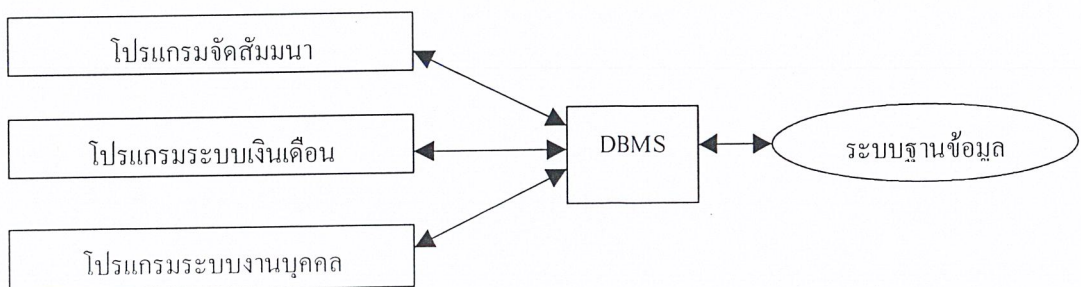
นอกจากจะทำให้เกิดความสับสนเปลืองเนื้อที่ในการจัดเก็บข้อมูลแล้ว ยังก่อให้เกิดปัญหาขึ้นเมื่อมีการเปลี่ยนแปลงค่าข้อมูล เพราะอาจจะหลงลืมเปลี่ยนข้อมูลในบางไฟล์ ทำให้ค่าข้อมูลเดียวกันที่เก็บในแต่ละไฟล์มีค่าไม่ตรงกัน ปัญหาดังกล่าว ทำให้เกิดความคิดที่ว่าจะนำข้อมูลเหล่านี้มาเก็บรวบรวมไว้ในที่เดียวกันหรือจะเก็บไว้ในฐานข้อมูลแทนที่จะเก็บไว้ในไฟล์ ดังรูปที่ 2.5



รูปที่ 2.5 การใช้ระบบฐานข้อมูล

### 2.3.5 ระบบการจัดเก็บฐานข้อมูล

การควบคุมดูแลการใช้ฐานข้อมูลนั้นเป็นเรื่องที่ยุ่ยากกว่าการใช้ไฟล์มาก เพราะที่เราจะต้องตัดสินใจว่าโครงสร้างในการเก็บข้อมูลควรจะเป็นเช่นไร และการเขียนโปรแกรมเพื่อสร้างและเรียกใช้ข้อมูลจากโครงสร้างไฟล์เหล่านี้ ก็เป็นเรื่องที่ยุ่ยากพอๆ และถ้าเกิดโปรแกรมเหล่านี้ทำงานผิดพลาดขึ้นมา ก็จะทำให้เกิดความเสียหายต่อโครงสร้างของข้อมูลทั้งหมดได้ เพื่อเป็นการลดภาระการทำงานของผู้ใช้ จึงได้มีการพัฒนาตัวหนึ่งที่มีชื่อเรียกว่า ระบบการจัดการฐานข้อมูล หรือ DBMS ซึ่งจะทำหน้าที่ในการควบคุมดูแลการสร้างและเรียกใช้งานฐานข้อมูล โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล เปรียบเสมือนว่า DBMS นี้จะเป็นสื่อกลางระหว่างผู้ใช้ และโปรแกรมต่างๆที่เกี่ยวข้องกับการใช้ฐานข้อมูลดังรูปที่ 2.6



รูปที่ 2.6 การใช้ระบบฐานข้อมูลด้วย DBMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.6 โมเดล (Model)

ประเภทของระบบการจัดการฐานข้อมูล แบ่งตามชนิดของโมเดลซึ่งโมเดลทุกชนิดจะประกอบไปด้วยองค์ประกอบ 2 อย่างคือ โครงสร้าง (Structure) และการใช้งาน (Operation)

โครงสร้าง หมายถึง โครงสร้างของระบบข้อมูล ซึ่งอาจไม่ใช่โครงสร้างที่จัดเก็บจริงๆ ก็ได้ แต่อย่างน้อยก็คือ โครงสร้างในแง่การมองของผู้ใช้ DBMS

การใช้งาน คือ วิธีการที่จะให้ผู้ใช้สามารถเรียกดู และแก้ไขข้อมูลในระบบได้ โดยไม่ต้องรู้ว่าจริงๆ แล้วข้อมูลมีการจัดเก็บอย่างไร

ลักษณะของโมเดลมีดังนี้

#### 1) โมเดลเชิงสัมพันธ์ (Relational model)

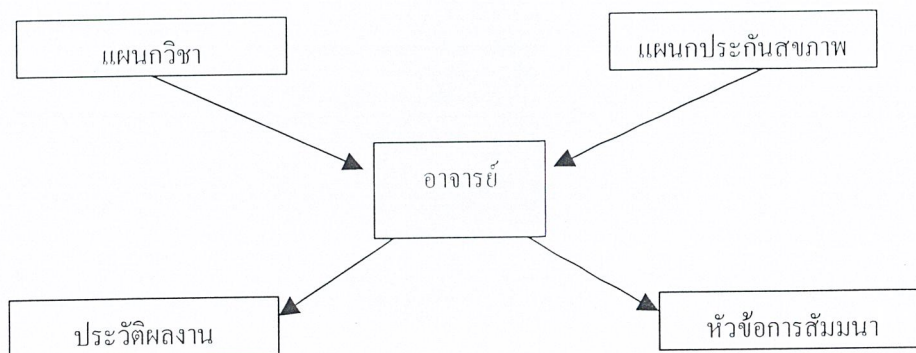
เป็นการเก็บข้อมูลแบบเป็นตาราง (Table) ซึ่งตารางนี้ก็คือ รีเลชัน (Relation) มีลักษณะเป็น 2 มิติ คือ ด้านแถว (Row) และด้านคอลัมน์ (Column) ดังตัวอย่างที่แสดงในรูปที่ 2.7 เป็นฐานข้อมูลเชิงสัมพันธ์ที่ประกอบด้วย Entity ต่าง ๆ ซึ่งข้อมูลของแต่ละ Entity จะถูกจัดเก็บในลักษณะของตาราง Entity

ข้อดีของโมเดลเชิงสัมพันธ์คือ ทั้งข้อมูลและโครงสร้างของข้อมูลมีความเป็นอิสระจากโปรแกรม เนื่องจากโครงสร้างของฐานข้อมูลแบบเชิงสัมพันธ์จะเป็นเพียงโครงสร้างข้อมูลระดับแนวความคิดมากกว่าโครงสร้างในระดับกายภาพ

ส่วนข้อเสีย ได้แก่ ฮาร์ดแวร์และระบบปฏิบัติการที่นำมาใช้จะมีความสามารถที่สูงกว่าโครงสร้างข้อมูลแบบอื่น เนื่องจากโปรแกรม DBMS ของโมเดลเชิงสัมพันธ์จะทำหน้าที่ในการจัดการกับโครงสร้างของข้อมูลแทนผู้ใช้ จึงส่งผลทำให้การทำงานของโปรแกรม DBMS มีความซับซ้อน และต้องใช้ทรัพยากรของฮาร์ดแวร์และระบบปฏิบัติการมากกว่าโครงสร้างข้อมูลในแบบอื่น

#### 2) โมเดลเน็ตเวิร์ค (Network model)

โมเดลแบบนี้เป็นลักษณะของการรวบรวมเรคคอร์ดต่างๆ และความสัมพันธ์ระหว่างเรคคอร์ด ความแตกต่างที่เห็นได้ชัดระหว่างโมเดลเชิงสัมพันธ์และแบบเน็ตเวิร์คคือ ในโมเดลเชิงสัมพันธ์จะแฝง (Implicit) การแสดงความสัมพันธ์เอาไว้ (หมายความว่าเรคคอร์ดที่มีความสัมพันธ์กันจะต้องมีค่าของข้อมูลในฟิลด์ใดฟิลด์หนึ่งเหมือนกัน) ส่วนการแสดงความสัมพันธ์ในโมเดลแบบเน็ตเวิร์คจะเป็นไปอย่างชัดเจน (Explicit) คือการแสดงไว้ในโครงสร้างอย่างชัดเจนดังรูปที่ 2.7



รูปที่ 2.7 โครงสร้างฐานข้อมูลแบบเน็ตเวิร์ค

จากรูปจะเห็นได้ว่า กรอบสี่เหลี่ยมแสดงถึงชนิดของเรคคอร์ดในฐานข้อมูลซึ่งจะมี 1 อันต่อ 1 Entity เครื่องหมายลูกศรแสดงถึงความสัมพันธ์ ซึ่งในรูปนี้แสดงความสัมพันธ์แบบหนึ่งต่อกลุ่มโดยหัวลูกศรจะออกจากส่วนของ “หนึ่ง” ไปยังส่วนของ “กลุ่ม” จะเห็นว่าแผนกวิชามีความสัมพันธ์กับอาจารย์ โดยที่ใน 1 แผนกวิชาอาจมีจำนวนอาจารย์ที่สังกัดได้มากกว่าหนึ่งคน วิธีการค้นหาข้อมูลสามารถทำได้โดยการใช้หัวลูกศรในการเชื่อมโยงความสัมพันธ์ ซึ่งบางครั้งอาจจะวิ่งวนหัวลูกศรก็ได้

ข้อดีของโมเดลเน็ตเวิร์คคือ สามารถรองรับข้อมูลที่มีความสัมพันธ์ในแบบกลุ่มต่อกลุ่มได้ สามารถเข้าถึงข้อมูลที่เป็นแบบ Member Record ได้ทุก Record สามารถควบคุมให้ข้อมูลเป็นไปตามกฎที่ใช้ในการควบคุมความถูกต้องให้กับข้อมูลได้ง่าย และยังสนับสนุนให้โปรแกรมมีความเป็นอิสระจากข้อมูลมากกว่าโมเดลแบบแตกสาขา เนื่องจากมีโครงสร้างซับซ้อนน้อยกว่า

ข้อเสียคือการออกแบบฐานข้อมูลกระทำยากขึ้นข้างยาก เนื่องจากจะต้องกำหนด set ของความสัมพันธ์ให้ครอบคลุมทุกข้อมูลในฐานข้อมูล และการเปลี่ยนแปลงโครงสร้างของข้อมูลก็กระทำยากเช่นกัน เนื่องจากจะต้องคำนึงถึง set ของความสัมพันธ์ต่างๆ ที่กำหนดไว้

### 3) โมเดลแบบแตกสาขา (Hierarchical Modal)

ในโมเดลแบบนี้มีโครงสร้างเหมือนต้นไม้ (Tree) ที่จริงแล้วโมเดลแบบนี้ก็เหมือนกับโมเดลแบบเน็ตเวิร์ค แต่ต่างกันตรงที่ว่า โมเดลแบบแตกสาขานี้มีกฎเกณฑ์ที่เพิ่มขึ้นมาอีก 1 ข้อคือในแต่ละกรอบจะมีหัวลูกศรเข้าหาได้ไม่เกิน 1 หัว จากตัวอย่างในรูปที่ 2.7 จะเห็นว่ากรอบของอาจารย์มีลูกศรเข้ามา 2 ทาง ดังนั้นเราจึงไม่อาจสร้างฐานข้อมูลสำหรับตัวอย่างนี้ โดยใช้โมเดลแบบแตกสาขาคู่วิธีปกติได้

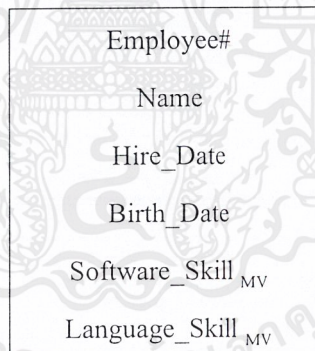
ข้อดีของการจัดเก็บฐานข้อมูลแบบ Hierarchical นี้คือ สามารถกำหนดกฎที่ใช้ควบคุมความถูกต้องให้กับข้อมูลภายในฐานข้อมูลได้ง่าย และมีโครงสร้างที่เหมาะสมต่อข้อมูลที่มีความสัมพันธ์ในแบบหนึ่งต่อกลุ่ม

ส่วนข้อเสียคือ ผู้ใช้ฐานข้อมูลจะต้องทราบถึงโครงสร้างทางกายภาพของข้อมูลที่จัดเก็บอยู่ในฐานข้อมูล จึงจะสามารถเรียกใช้ข้อมูลได้ ถึงแม้ว่าโปรแกรม DBMS จะทำให้การเข้าถึงข้อมูลเป็นอิสระจากตัวฐานข้อมูลก็ตาม และโครงสร้างของฐานข้อมูลไม่สามารถรองรับข้อมูลที่มีความสัมพันธ์ในแบบกลุ่มต่อกลุ่มได้

#### 4) Semantic Object Model (SOM)

ลักษณะของโมเดลนี้ภายใน SOM จะใช้รูปสี่เหลี่ยมแทน Object Class ต่างๆ โดยจะมีการกำหนดชื่อของ Object ไว้ด้านบนส่วนภายในสี่เหลี่ยมจะประกอบด้วย Attribute Class ต่างๆ และ Object อื่นที่ Object เกี่ยวข้อง เช่น Object “EMPLOYEE” ที่ประกอบด้วย Attribute “Employee#”, “Name”, “Hire\_Date”, “Birth\_Date”, “Software\_Skill” และ “Language\_Skill” จะถูกแทนด้วยรูปที่ 2.8

EMPLOYEE Object



รูปที่ 2.8 ลักษณะของการเก็บข้อมูล

ในกรณีที่ Attribute ใน Object เป็น Attribute ในแบบ multi-value จะต้องกำหนดคำว่า “MV” ไว้หลัง Attribute นั้น เช่น Attribute ที่ใช้กำหนดความสามารถทางการใช้งานโปรแกรม (Attribute Software\_Skill) และภาษา (Attribute Language\_Skill) ของพนักงาน ซึ่งพนักงานแต่ละคน สามารถที่จะมีความสามารถทางด้านโปรแกรมได้มากกว่า 1 โปรแกรม และได้มากกว่า 1 ภาษาดังนั้นทั้ง 2 Attribute จึงต้องมีการกำหนดคำว่า “MV” ไว้ด้านหลัง

## 5) Object-oriented

Object-oriented จะต้องประกอบด้วย Attribute และ Method รวมทั้งจะต้องมีคุณลักษณะที่เป็นเอกลักษณ์ของตนเอง ดังนั้นแต่ละ Object จะต้องมีการมี Attribute ที่เรียกว่า Object ID (OID) เพื่อให้ได้แต่ละ Object ที่แตกต่างจาก Object อื่นๆ OID นี้จะแตกต่างจากคีย์หลัก เนื่องจาก OID ของ Object จะถูกกำหนดขึ้นเพียงครั้งเดียว และจะไม่สามารถเปลี่ยนแปลงค่าได้อีก รวมทั้งจะไม่ถูกนำกลับมาใช้ใหม่ ถึงแม้ว่า Object นั้นจะถูกเลิกใช้งาน นอกจากนี้ OID นี้ ยังถูกใช้เชื่อมโยงกับ Object อื่นที่เกี่ยวข้อง แต่การเชื่อมโยงนี้จะไม่เชื่อมโยงโดยทางกายภาพ ดังนั้น OID จึงไม่ถูกกระทบจากการเปลี่ยนแปลงโครงสร้างของหน่วยจัดเก็บ (Storage) ส่วนคีย์หลักจะถูกกำหนดขึ้นโดยผู้ใช้ ดังนั้นจึงสามารถเปลี่ยนแปลงได้ ประกอบกับคีย์หลักถูกกำหนดขึ้นเพื่อช่วยในการกำหนดคุณลักษณะเฉพาะ (Identity) ของ Entity ต่างๆ กระทำได้ง่ายขึ้นเท่านั้น แต่ไม่ได้ก่อให้เกิดความสัมพันธ์ระหว่าง Entity เหล่านั้นเกิดขึ้นแต่อย่างใด

## 2.4 ประโยชน์จากการประมวลผลด้วยฐานข้อมูล

ประโยชน์จากการใช้ฐานข้อมูลในการประมวลผล สามารถสรุปได้ดังนี้

1) ลดความซ้ำซ้อนของข้อมูล การประมวลผลโดยใช้ระบบไฟล์นั้นจำเป็นที่ผู้ใช้แต่ละคนจะต้องมีไฟล์ส่วนตัวเอาไว้ ดังนั้นจึงเกิดเหตุการณ์ที่ข้อมูลชนิดเดียวกันถูกเก็บไว้หลายๆแห่งหรือที่เรียกว่าความซ้ำซ้อน การนำข้อมูลทั้งหมดมาเก็บไว้ที่เดียวกันในฐานข้อมูลนี้เป็นการลดความซ้ำซ้อนของข้อมูลได้

2) สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง ประโยชน์ในข้อนี้สืบเนื่องมาจากข้อที่แล้ว เพราะการเก็บข้อมูลไว้หลายๆแห่งอาจจะก่อให้เกิดปัญหาที่ว่า การแก้ไขข้อมูลเดียวกันทำไม่เหมือนกันในทุกๆแห่งทำให้ข้อมูลชุดเดียวกันอาจมีค่าไม่ตรงกัน ดังนั้นการใช้ระบบฐานข้อมูลทำให้เราสามารถลดความซ้ำซ้อนลง โดยมี DBMS เมื่อเกิดการแก้ไขข้อมูลขึ้น ก็จะแก้ไขให้เหมือนกันครบทุกแห่ง

3) สามารถใช้ข้อมูลร่วมกันได้ ซึ่งการใช้ข้อมูลร่วมกันได้นี้ ไม่ได้จำกัดอยู่เฉพาะโปรแกรมที่ใช้ข้อมูลอยู่ในปัจจุบันเท่านั้น แต่กินความถึงโปรแกรมประยุกต์ที่จะพัฒนาขึ้นมาใหม่ด้วยที่สามารถใช้ข้อมูลที่มีอยู่ได้เสีย โดยไม่จำเป็นต้องเพิ่มเติมข้อมูลเข้าไปในระบบอีก

4) สามารถควบคุมความเป็นมาตรฐานได้ จากการนำข้อมูลมารวมกันไว้ในฐานข้อมูล ทำให้ผู้มีหน้าที่ควบคุมดูแลการใช้ระบบฐานข้อมูล สามารถกำหนดมาตรฐานของข้อมูลขึ้นมาได้ เช่น การให้หน่วยวัดมาตรฐานที่เหมือนกัน รูปแบบการเขียนวันที่ให้เหมือนกัน เป็นต้น ซึ่งการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐานเดียวกันนี้ ทำให้การแลกเปลี่ยนข้อมูลระหว่างระบบเป็นไปอย่างสะดวกและถูกต้อง เราเรียกผู้ควบคุมระบบว่า ผู้บริหารฐานข้อมูล (Data Base Administrator) หรือ DBA

5) สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้ คำว่า ระบบความปลอดภัยนี้ หมายถึง การป้องกันไม่ให้ผู้ที่ไม่มีสิทธิ์มาใช้ข้อมูลในระบบได้ เนื่องจาก DBA เป็นผู้ควบคุมดูแลการใช้ข้อมูล จึงสามารถกำหนดสิทธิ์การใช้ให้แก่ผู้ใช้คนใดก็ได้ตามความเหมาะสม และผู้ใช้แต่ละคนก็อาจจะใช้ข้อมูลได้ระดับที่ต่างกัน

6) สามารถควบคุมความคงสภาพของข้อมูลได้ ความไม่คงสภาพของข้อมูล คือการที่เกิดความขัดแย้งของข้อมูลดังที่ได้กล่าวไปแล้ว ซึ่งในกรณีของความขัดแย้งนี้จะเกิดขึ้นได้ก็เมื่อข้อมูลมีความซับซ้อนเท่านั้น แต่ในอีกแง่หนึ่งของความคงสภาพอาจเกิดขึ้นได้ แม้ว่าจะไม่มีความซ้ำซ้อน เช่น ข้อมูลเกี่ยวกับอายุของพนักงานของบริษัทอาจจะมีค่า 300 แทนที่จะเป็น 30 ซึ่งความผิดพลาดนี้เกิดขึ้นได้ง่ายๆจากความสะเพร่าในการกรอกข้อมูลก็ได้ ในลักษณะของความไม่ถูกต้องนี้ ผู้ที่ออกแบบระบบฐานข้อมูลสามารถใส่กฎเกณฑ์ลงไปเพื่อควบคุมความคงสภาพไว้เช่นตามตัวอย่างนี้อาจจะใส่กฎว่า ค่าของอายุจะต้องเป็นเลขระหว่าง 16 ถึง 60 เป็นต้น

7) สามารถสร้างความสมดุล ในความขัดแย้งของความต้องการได้ การที่ผู้ใช้ทั้งหมดขององค์กรใช้ข้อมูลร่วมกันเช่นนี้ ทำให้ DBA ทราบถึงความต้องการและความสำคัญของผู้ใช้งานทั้งหมด จึงสามารถกำหนดโครงสร้างของฐานข้อมูลเพื่อการให้บริการที่ดีที่สุดได้ เช่น เลือเก็บข้อมูลจะต้องใช้บ่อยๆไว้ในสื่อข้อมูลที่มีความเร็วเป็นพิเศษ เป็นต้น

8) เกิดความเป็นอิสระของข้อมูล วิธีการที่จะทำให้ความเข้าใจว่า ความเป็นอิสระของข้อมูลนั้นคืออะไร ก็โดยการลองดูในด้านตรงข้ามก่อนว่า ข้อมูลที่ไม่เป็นอิสระนั้นอย่างไร ข้อมูลที่ไม่เป็นอิสระก็คือข้อมูลที่ถูกนำมาประยุกต์ใช้ยังมีความผูกพันอยู่กับวิธีการจัดเก็บและเรียกใช้ข้อมูลซึ่งในลักษณะการเขียนโปรแกรมประยุกต์บางประเภท เราอาจจำเป็นต้องใส่เทคนิคการจัดเก็บ และเรียกใช้ข้อมูลไว้ในตัวโปรแกรมด้วย นั่นหมายความว่า ถ้าเกิดต้องมีการเปลี่ยนแปลงวิธีการจัดเก็บหรือเรียกใช้แล้ว ผู้ใช้จำเป็นที่จะต้องสร้างวิธีการประยุกต์ใช้ขึ้นมาใหม่ ซึ่งเป็นความไม่สะดวกอย่างยิ่ง และทำให้เราหมดโอกาสที่จะปรับปรุงโครงสร้างของข้อมูล เพื่อให้ใช้งานได้อย่างมีประสิทธิภาพมากขึ้น

## 2.5 สถาปัตยกรรมของระบบฐานข้อมูล

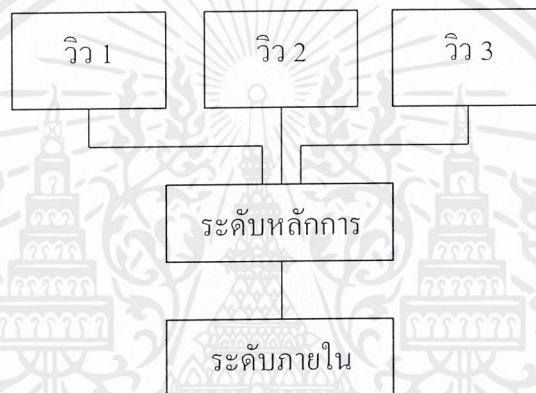
สถาปัตยกรรมของระบบฐานข้อมูล หมายถึง องค์ประกอบต่างๆของระบบฐานข้อมูลซึ่งสามารถแบ่งเป็นระดับได้ 3 ระดับ ดังรูปที่ 2.9 ซึ่งได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ระดับภายใน (Internal หรือ Physical level) เป็นระดับที่ต่ำที่สุด ได้แก่ ระดับของการจัดเก็บข้อมูลจริงๆ

2) ระดับหลักการ ( Conceptual level ) เป็นระดับที่อยู่ถัดลงมา ได้แก่ ระดับของการมอง Entity และความสัมพันธ์ระหว่าง Entity ทั้งหมดรวมทั้งกฎเกณฑ์ต่างๆเกี่ยวกับข้อมูล และผู้มีสิทธิ์จะใช้ เป็นต้น

3) ระดับภายนอก (External หรือ View level) เป็นระดับที่อยู่สูงที่สุด เป็นข้อมูลที่จะมองเห็นจากการใช้งานของผู้ใช้แต่ละคน



รูปที่ 2.9 ระดับของข้อมูล 3 ระดับ

### 2.5.1 ความเป็นอิสระของข้อมูล

หมายถึง การที่ข้อมูลในแต่ละส่วนจะไม่มีผลกระทบต่อกันเมื่อมีการเปลี่ยนแปลงต่างๆเกิดขึ้น สามารถแบ่งเป็น 2 ชนิดคือ

1) ความเป็นอิสระแบบกายภาพ (Physical data independence) คือลักษณะของการเปลี่ยนแปลงแก้ไขข้อมูลในระดับภายใน จะไม่มีผลกระทบต่อโครงสร้างของระดับหลักการและระดับภายนอก

2) ความเป็นอิสระแบบตรรก (Logical data independence) คือ ลักษณะของการเปลี่ยนแปลงข้อมูลในระดับหลักการ โดยที่จะไม่มีผลกระทบต่อระดับของผู้ใช้ภายนอก

### 2.5.2 Instance และ Database Schema

1) Instance หมายถึง ข้อมูลขณะใดขณะหนึ่งของฐานข้อมูล ในระหว่างที่มีการเรียกใช้และแก้ไขข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) **Database schema** หมายถึง เค້าร่างที่ได้จากการออกแบบฐานข้อมูลโดยรวม สามารถแบ่งออกได้เป็นระดับคล้าย ๆ กับระดับของข้อมูล คือ ระดับต่ำสุด ได้แก่ ระดับภายในหรือภายในภาพ ระดับกลาง ได้แก่ ระดับหลักการ และระดับนอกสุด หรือระดับที่เรียกว่า Subschema โดยทั่ว ๆ ไป ระบบฐานข้อมูลจะประกอบด้วย Schema แบบกายภาพ 1 ตัว แบบหลักการ 1 ตัว และมี Subschema ได้หลาย ๆ ตัว

### 2.5.3 ระบบการจัดการฐานข้อมูล

ในการใช้ระบบฐานข้อมูลนั้น ผู้ใช้ไม่จำเป็นต้องทราบถึงรายละเอียดของการจัดเก็บข้อมูลจริง ๆ แต่เป็นหน้าที่ของระบบการจัดการฐานข้อมูล หรือ DBMS ซึ่งเป็นซอฟต์แวร์ที่จะทำหน้าที่ดูแลการใช้งานให้กับผู้ใช้เหล่านี้ โดยที่ DBMS มีหน้าที่ต่าง ๆ ดังนี้

1) **ทำหน้าที่ติดต่อกับตัวจัดเก็บระบบไฟล์** เนื่องจากการใช้งานส่วนใหญ่ของระบบฐานข้อมูล คือลักษณะการใช้งานกับข้อมูลขนาดใหญ่ ซึ่งก็เป็นที่แน่นอนว่า เราไม่สามารถนำข้อมูลทั้งหมดนี้เข้ามาไว้ในหน่วยความจำหลักพร้อม ๆ กันทั้งหมดได้ กล่าวคือ ข้อมูลทั้งหมดจะเก็บไว้ในดิสก์ และจะถูกนำมาสู่หน่วยความจำหลักเฉพาะส่วนที่ต้องการใช้งาน หน้าที่ในการค้นหาว่าข้อมูลที่เราต้องการนั้นเก็บอยู่ที่ตำแหน่งใดในดิสก์ เป็นฟังก์ชันการทำงานส่วนหนึ่งของระบบดำเนินงาน (Operationg System) อันได้แก่ส่วนที่เรียกว่า ตัวจัดเก็บระบบไฟล์ (File Manager) นั่นก็คือ DBMS จะทำหน้าที่ประสานงานกับตัวจัดเก็บระบบไฟล์ในการจัดเก็บ เรียกใช้ และแก้ไขข้อมูล

2) **การควบคุมความคงสภาพ** เป็นหน้าที่ของ DBMS ที่จะต้องควบคุมค่าของข้อมูลในระบบให้อยู่ในกรอบที่ถูกต้องตามที่กำหนดไว้ในส่วนของ Schema เช่น ยอดเงินในบัญชีเงินฝากประจำจะต้องไม่น้อยกว่า 100 บาท หรือรหัสของอาจารย์ที่ปรึกษาของนักเรียนต้องเป็นรหัสที่มีปรากฏอยู่ในเรคคอร์ดของอาจารย์ที่ปรึกษา ดังนั้นการเปลี่ยนแปลงแก้ไขข้อมูลทุกครั้งจึงเป็นหน้าที่ของ DBMS ที่จะต้องดูแลให้ผลลัพธ์ถูกต้องอยู่เสมอ

3) **การควบคุมระบบความปลอดภัย** ได้แก่ การป้องกันไม่ให้ผู้ที่ไม่ได้รับอนุญาตเข้ามาเห็นหรือแก้ไขข้อมูลในส่วนที่ต้องการปกป้องเอาไว้

4) **การสร้างระบบสำรองและการฟื้นฟูสภาพ** ได้แก่ ฟังก์ชันในการจัดทำข้อมูลสำรอง และเมื่อใดก็ตามที่มีปัญหาเกิดขึ้น ไม่ว่าจะเป็นการขัดข้องของระบบไฟล์หรือเครื่องเกิดความเสียหาย DBMS จะต้องใช้ระบบข้อมูลสำรองในการฟื้นฟูสภาพ ให้ระบบข้อมูลกลับเข้าสู่สภาพที่ถูกต้องสมบูรณ์ได้

5) **การควบคุมภาวะพร้อมกัน** หน้าที่ที่สำคัญที่สุดอีกประการหนึ่ง คือ การควบคุมการใช้ข้อมูลในสภาพที่มีผู้ใช้พร้อม ๆ กันหลายคน อันได้แก่ การควบคุมภาวะพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Concurrency Control) กล่าวคือ DBMS จะต้องควบคุมลำดับการทำงานให้เป็นไปอย่างถูกต้อง เช่น ระหว่างที่กำลังแก้ไขข้อมูลส่วนหนึ่งยังไม่เสร็จก็จะไม่อนุญาตให้ผู้อื่นเข้ามาเปลี่ยนแปลงแก้ไขข้อมูลนั้นได้

#### 2.5.4 ผู้บริหารฐานข้อมูล

ผู้บริหารฐานข้อมูล หรือ DBA คือ ผู้ที่มีหน้าที่ควบคุมการบริหารของระบบฐานข้อมูลทั้งหมด หน้าที่ความรับผิดชอบของผู้บริหารฐานข้อมูลได้แก่

- 1) ตัดสินใจว่าจะรวมข้อมูลใด เข้าไว้ในระบบบ้าง
- 2) วิเคราะห์และตัดสินใจว่าควรจัดเก็บข้อมูลด้วยวิธีใด และใช้เทคนิคใดในการเรียกใช้ข้อมูล

- 3) ประสานงานกับผู้ใช้
- 4) กำหนดระบบความปลอดภัย และความคงสภาพของข้อมูล
- 5) กำหนดแผนในการสร้างระบบข้อมูลสำรองและการฟื้นฟูสภาพ
- 6) คอยสำรวจคุณภาพการทำงาน และตรวจตราความต้องการของผู้ใช้

#### 2.5.5 ผู้ใช้ระบบ

ระบบฐานข้อมูลเกิดขึ้นมา เพื่อที่จะสร้างสภาพการใช้ข้อมูลให้สะดวกสบายและให้ประสิทธิภาพแก่ผู้ใช้ให้ดีที่สุด ลักษณะของผู้ใช้ระบบแบ่งได้ 4 กลุ่มคือ

1) ผู้เขียนโปรแกรมประยุกต์ ผู้ใช้กลุ่มนี้ได้แก่ นักเขียนโปรแกรมมืออาชีพ ที่เรียกใช้ข้อมูลในระบบ โดยใช้การเขียนโปรแกรมด้วยภาษาชั้นสูง และแฝงเอาไว้ในโปรแกรม ในลักษณะเช่นนี้เราจะเรียกส่วนของภาษาชั้นสูงว่าเป็นภาษาหลัก (Host Language)

2) ผู้ใช้ทั่วไป ได้แก่ เหล่าผู้ใช้ที่เรียกใช้ข้อมูลโดยผ่านทาง DML เท่านั้น ผู้ใช้เหล่านี้ไม่จำเป็นต้องมีความรู้เกี่ยวกับการเขียนโปรแกรม เนื่องจากคำสั่งของ DML จะเป็นคำสั่งง่าย ๆ และสามารถเรียกใช้งานได้โดยสะดวก

3) โอเปอร์เรเตอร์ คือ กลุ่มผู้ใช้ที่จะทำการใส่ข้อมูลอย่างเดียว กลุ่มนี้ไม่จำเป็นต้องมีความรู้ใด ๆ ทั้งสิ้นที่เกี่ยวกับระบบคอมพิวเตอร์ ลักษณะการทำงานก็ได้แก่การใช้โปรแกรมประยุกต์ที่ผู้ใช้กลุ่มแรกสร้างไว้ แล้วยกเพียงแต่ทำตามทีที่จอภาพระบุ เช่น การใส่ข้อมูลต่าง ๆ เข้าไปเท่านั้น

4) ผู้ใช้พิเศษ ได้แก่กลุ่มของผู้ใช้ที่นำระบบไปใช้ในลักษณะงานที่ไม่ใช่งานประมวลผลข้อมูล เช่น การสร้างโปรแกรมช่วยการออกแบบ (Computer Aided Design) หรือ CAD

### 2.5.6 การจัดเก็บข้อมูล

ข้อมูลที่เก็บอยู่ในฐานข้อมูลใด ๆ ก็คือข้อมูลของ Entity การสร้างความสัมพันธ์ระหว่าง Entity ในฐานข้อมูลเชิงสัมพันธ์กระทำได้โดยการกำหนดให้ Entity ที่มีความสัมพันธ์กันมี Attribute ที่เหมือนกัน และใช้ค่าของ Attribute ในส่วนที่เหมือนกันนี้เป็นตัวระบุข้อมูลใน Entity ที่มีความสัมพันธ์กัน ซึ่งความสัมพันธ์ระหว่าง Entity แบ่งได้ 3 ชนิด ได้แก่

#### 1) การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบหนึ่งต่อหนึ่ง

สมมติว่า พนักงานขายแต่ละคนมีสิทธิขายของให้แก่ลูกค้าเพียงคนเดียวเท่านั้น ซึ่งเป็นการแสดงถึงความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างพนักงานขายกับลูกค้า ในลักษณะเช่นนี้ แทนที่เราจะใช้ตาราง 2 ตารางสำหรับเก็บค่าข้อมูลของ Entity ทั้งสอง เราสามารถรวม Entity ทั้ง 2 นี้เข้าด้วยกัน โดยใช้ตารางเก็บข้อมูลเพียงตารางเดียวดังที่แสดงในตารางที่ 2.1

ตารางที่ 2.1 การสร้างความสัมพันธ์แบบหนึ่งต่อหนึ่งโดยใช้ตารางเดียว

รหัสพนักงาน	ชื่อพนักงานขาย	รหัสลูกค้า	ชื่อลูกค้า
3	สุรศักดิ์	123	สมศักดิ์
6	เอกนรินทร์	456	พีรพัฒ
9	สโรจพันธ์	789	ธนา

#### 2) การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบหนึ่งต่อกลุ่ม

สมมติความสัมพันธ์ระหว่างพนักงานขาย และลูกค้า โดยที่พนักงานขายแต่ละคนสามารถติดต่อกับลูกค้าได้มากกว่า 1 คน แต่ลูกค้าแต่ละคนต้องซื้อของจากพนักงานขายเพียงคนเดียวเท่านั้น ซึ่งเป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม ในลักษณะนี้เราสามารถใช่ตาราง 1 ตารางสำหรับเก็บข้อมูลของพนักงานขาย และแยกเก็บข้อมูลของลูกค้าไว้อีกตารางหนึ่ง โดยตารางของลูกค้านี้จะเพิ่มขึ้นอีก 1 คอลัมน์สำหรับจัดเก็บค่ารหัสพนักงานขายที่ลูกค้าคนนั้นใช้บริการอยู่ โดยที่ค่ารหัสพนักงานขายนี้ถือเป็นคีย์หลักในตารางของพนักงานขายดังตารางที่ 2.2

ตารางที่ 2.2 การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม

ข้อมูลลูกค้า		
รหัสลูกค้า	ชื่อลูกค้า	รหัสพนักงาน
123	ปราณี	15
456	นันทนา	7
789	สินใจ	23
ข้อมูลพนักงานขาย		
รหัสพนักงาน	ชื่อพนักงาน	
7	วิมล	
15	ดวงตา	
23	สมบูรณ์	

จะเห็นว่าข้อมูลของลูกค้าแต่ละคน จะปรากฏในแถวของตารางลูกค้าเพียงคนเดียวเท่านั้น รหัสพนักงานขายในแต่ละแถวของตารางนี้ก็จะมีเพียงคนเดียว ซึ่งเป็นตัวยืนยันว่าลูกค้าแต่ละคนมีสิทธิติดต่อกับพนักงานขายคนเดียว ส่วนในตารางของพนักงานขายนั้น ไม่มีข้อมูลที่เกี่ยวข้องกับลูกค้าเลย ดังนั้นถ้าเราต้องการทราบว่า พนักงานคนใดคนหนึ่งมีใครเป็นลูกค้าอยู่บ้างก็ต้องไล่ดูรหัสของพนักงานคนนั้นในตารางลูกค้าเอง ซึ่งอาจปรากฏอยู่มากกว่า 1 แถวก็ได้ แสดงว่าพนักงานแต่ละคนมีสิทธิที่จะมีลูกค้าได้มากกว่า 1 คน

### 3) การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบกลุ่มต่อกลุ่ม

สมมติให้ลูกค้าสามารถสั่งซื้อสินค้าได้มากกว่า 1 อย่างในการสั่งซื้อแต่ละครั้ง และในขณะเดียวกันก็อนุญาตให้สินค้า 1 อย่างปรากฏอยู่ในรายการสั่งซื้อได้มากกว่า 1 ครั้ง จึงเห็นได้ว่าลักษณะความสัมพันธ์ระหว่างการสั่ง และสินค้า เป็นแบบกลุ่มต่อกลุ่ม ในกรณีเช่นนี้ เราต้องสร้างตารางขึ้นมาเป็นกรณีพิเศษ เพื่อใช้บรรจุค่าคีย์ของตารางทั้งสองนี้อันได้แก่ รหัสการสั่ง และรหัสสินค้า และตารางนี้ยังอาจประกอบด้วยข้อมูลอื่น ๆ ที่มีค่าขึ้นอยู่กับค่าของทั้งการสั่ง และรหัสสินค้า เช่น จำนวนที่สั่ง และราคา ดังตารางที่ 2.3

ตารางที่ 2.3 การสร้างความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ข้อมูลการสั่งซื้อ			
รหัสสินค้า	วันที่สั่ง	รหัสลูกค้า	
12498	020841	124	
12491	031241	311	
12494	040941	315	
ข้อมูลสินค้า			
รหัสสินค้า	ชื่อสินค้า		
AX12	เตารีด		
AF52	ลูกบอล		
ZZ00	จักรยาน		
บัญชีการสั่ง			
รหัสการสั่ง	รหัสสินค้า	จำนวนสินค้า	ราคา
12498	AX12	1	580
12491	AF52	2	450
12494	ZZ00	3	2150

ในการค้นหาว่า การสั่งตามหมายเลขที่เราสนใจนั้นมีการสั่งซื้อสินค้าอะไรบ้างก็เริ่มด้วยการดูข้อมูลในตารางบัญชีการสั่ง โดยดูในแถวที่มีรหัสการสั่งที่เราสนใจ ซึ่งเราจะได้รับรหัสของสินค้าที่อยู่ในแถวเดียวกัน และเราสามารถไปดูข้อมูลของสินค้าจากตารางสินค้าได้

### 2.5.7 ภาษาสำหรับนิยามข้อมูล

การกำหนด Schema ของฐานข้อมูลทั้ง 3 ระดับกระทำได้โดยการใช้ภาษาพิเศษ ที่เรียกว่าภาษาสำหรับนิยามข้อมูล DDL (Data Definition Language) ผลจากการคอมไพล์โปรแกรมที่เขียนด้วย DDL จะทำให้เกิดตารางที่จะจัดเก็บข้อมูลในไฟล์ชนิดหนึ่งที่เรียกว่า พจนานุกรมข้อมูล (Data Dictionary) ซึ่งจะเก็บข้อมูลที่เกี่ยวข้องกับโครงสร้างที่ได้จากการออกแบบฐานข้อมูลนั้น ๆ

### 2.5.8 ภาษาสำหรับการใช้ข้อมูล

การใช้ข้อมูลในระบบฐานข้อมูล แบ่งออกได้เป็น 3 ประเภท คือ

#### 1) การเรียกดูข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การเพิ่มข้อมูลใหม่เข้าไป

3) การลบข้อมูล

ในระดับของผู้ใช้ จะไม่ทราบและไม่สนใจว่าวิธีการจัดเก็บข้อมูลจริง ๆ นั้นเป็นอย่างไร ดังนั้นการใช้ฐานข้อมูลในระดับนี้สามารถกระทำได้โดยผ่านทาง DBMS (Database Management System) โดยการใช้ภาษาสำหรับการใช้ข้อมูล DML (Data Manipulation Language)

เราสามารถแบ่งชนิดของ DML ออกได้เป็น 2 ชนิด คือ

1) Procedural ในการใช้ DML แบบนี้ เป็นหน้าที่ของผู้ใช้ที่จะต้องระบุว่า ต้องการข้อมูลอะไร และจะเอาข้อมูลนั้นมาได้โดยวิธีใด

2) Nonprocedural สำหรับ DML แบบนี้ผู้ใช้เพียงแต่ระบุว่าต้องการข้อมูลอะไร โดยไม่ต้องบอกวิธีการเลย

## 2.6 ฟังก์ชันการขึ้นต่อกัน กิย์ และนอร์มัลไลซ์

เป้าหมายของการใช้ระบบฐานข้อมูลเชิงสัมพันธ์ ก็คือการออกแบบรีเลชันเพื่อที่จะสามารถเรียกใช้ข้อมูลในระบบได้อย่างสะดวกและมีประสิทธิภาพ โดยพยายามให้เกิดความซ้ำซ้อนของข้อมูลน้อยที่สุด ศาสตร์แขนงหนึ่งที่เป็นพื้นฐานของการออกแบบฐานข้อมูล ได้แก่ การออกแบบระบบให้อยู่ใน รูปแบบนอร์มัล (Normal Form) โดยอาศัยความรู้เกี่ยวกับ ฟังก์ชันการขึ้นต่อกัน (Functional Dependency) ของข้อมูล ก่อนที่จะกล่าวถึงวิธีการทำให้ระบบอยู่ในรูปแบบนอร์มัล จะกล่าวถึงกฎของความคงสภาพที่เกิดขึ้นใน โมเดลเชิงสัมพันธ์ก่อน

### 2.6.1 กฎของความคงสภาพ (Integrity Rule)

เมื่อกกล่าวถึงกฎของความคงสภาพของโมเดลเชิงสัมพันธ์ โดยทั่ว ๆ ไปแล้วจะหมายถึง กฎความคงสภาพ 2 กรณี คือ

1) กฎความคงสภาพของ Entity (Entity Integrity Rule) กำหนดไว้ว่า Attribute ทุกตัวที่เป็นส่วนของกิย์หลักจะไม่อนุญาตให้มีค่าว่าง ความหมายของค่าว่างก็คือ ค่าที่ไม่ทราบแน่ชัดหรือค่าที่ไม่เหมาะสม

2) กฎความคงสภาพของการอ้างอิง (Referential Integrity Rule) กำหนดไว้ว่า ถ้าเรามีรีเลชัน R2 ซึ่งมี FK เป็นคีย์นอกที่อ้างอิงถึงคีย์หลัก PK ในรีเลชัน R1 สำหรับทุก ๆ ค่าของ FK ใน R2 จะต้องมิตักษณะดังนี้

2.1) มีค่าเท่ากับค่า PK ในทัพเพิลใดทัพเพิลหนึ่งใน R1

2.2) ค่าของ Attribute ทุกตัวใน FK เป็นค่าว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสำคัญของกฎข้อนี้คือ เมื่อมีการอ้างอิงจากระดับหนึ่ง ไปยังระดับหนึ่งแล้ว เราต้องรับประกันว่าข้อมูลในรึเลขชั้นที่สองจะต้องไม่เป็นค่าว่าง

เมื่อก้าวถึงกฎความคงสภาพทั้งสองนี้ เราต้องการที่จะประกันความคงสภาพนี้ในทุก ๆ สถานะและทุก ๆ ขณะของระบบฐานข้อมูล นั่นหมายความว่าเมื่อมีการแก้ไขข้อมูลในระบบเกิดขึ้น เมื่อไรก็เป็นหน้าที่ของระบบการจัดการฐานข้อมูลที่จะต้องตรวจสอบอยู่เสมอว่ายังมีความคงสภาพที่ถูกต้องตามกฎเกณฑ์หรือไม่ และถ้าการแก้ไขข้อมูลครั้งใดที่ผิดกฎเกณฑ์ความคงสภาพไปแล้ว ก็ขึ้นอยู่กับวิธีการออกแบบของ DBMS แต่ละตัวว่าจะแก้ไขสถานการณ์อย่างไร

วิธีการแก้ไขสถานการณ์นี้แบ่งออกได้เป็น 2 วิธี คือ

**วิธีที่ 1** ระบบจะไม่ยอมรับการเปลี่ยนแปลงแก้ไขข้อมูลที่ทำให้เกิดปัญหานั้น ๆ โดยอาจจะแสดงข้อความออกมาบอกผู้ใช้

**วิธีที่ 2** ระบบจะอนุญาตให้การแก้ไขนั้น ๆ เกิดขึ้นได้ โดยระบบจะทำการปรับผลลัพธ์ให้ยังคงมีความคงสภาพตามกฎทั้ง 2 โดยอาจจะเพิ่มเติมการกระทำอะไรบางอย่างกับฐานข้อมูลก็ได้

ระบบควรจะให้ผู้ใช้ หรือ DBA สามารถตัดสินใจได้ในแต่ละครั้งว่าจะใช้แบบไหน ซึ่งระบบ DBMS ส่วนใหญ่ในปัจจุบันก็ถูกออกแบบให้เป็นเช่นนี้ ดังนั้นในกรณีนี้ DBA ควรจะมีความรู้เกี่ยวกับการประยุกต์ใช้งานของฐานข้อมูลเพียงพอที่จะตอบทั้ง 3 ข้อต่อไปนี้ คือ

1) ค่าของคีย์นอกสามารถเป็นค่าว่างได้หรือไม่

2) จะเกิดอะไรขึ้น ถ้ามีการออกคำสั่งให้ลบข้อมูลในรึเลขชั้นที่ถูกอ้างอิงจากระดับอื่น ซึ่ง

คำตอบสำหรับคำถามนี้เป็นไปได้ 3 ทางคือ

2.1) กระทำแบบเป็นลูกโซ่ (Cascade)

2.2) ขอมแบบมีเงื่อนไข (Restricted)

2.3) เปลี่ยนเป็นค่าว่าง (Nullify)

3) จะเกิดอะไรขึ้นถ้ามีการออกคำสั่งให้แก้ไขค่าของคีย์หลักที่ถูกอ้างอิงด้วยคีย์ นอกจากรึเลขชั้นอื่น ซึ่งคำตอบก็จะมี 3 ทางคือ

3.1) กระทำแบบเป็นลูกโซ่ (Cascade)

3.2) ขอมแบบมีเงื่อนไข (Restricted)

3.3) เปลี่ยนเป็นค่าว่าง (Nullify)

## 2.6.2 ฟังก์ชันการขึ้นต่อกัน

เนื่องจากปัญหาจะเกิดขึ้นเมื่อเราแก้ไขข้อมูลในระบบ มักจะมีสาเหตุมาจากการที่ข้อมูลในส่วนต่าง ๆ ของระบบมีการขึ้นต่อกันที่ไม่เหมาะสม

นิยาม สมมติให้ B เป็น Attribute ตัวหนึ่ง และ A เป็น Attribute อีกตัวหนึ่ง (ซึ่ง A อาจจะประกอบด้วย Attribute มากกว่า 1 ตัวก็ได้) เรากล่าวว่า B มีฟังก์ชันการขึ้นอยู่กับ A ก็ต่อเมื่อค่าของ A สามารถใช้ในการเลือก (Determine) ค่าของ B ได้เพียง 1 ค่าเสมอ

สัญลักษณ์ของการขึ้นต่อกันในลักษณะเช่นนี้ จะเขียนว่า

$A \rightarrow B$  ซึ่งอ่านว่า B มีฟังก์ชันการขึ้นอยู่กับ A หรือ A มีฟังก์ชันในการเลือก B หรือจะพูดสั้น ๆ ว่า B ขึ้นกับ A

### 2.6.3 คีย์

นิยาม เรากล่าวว่า Attribute A (ซึ่ง A อาจจะประกอบด้วย Attribute หลายตัวก็ได้) เป็นคีย์หลักของรีเลชัน R ถ้า

- 1) Attribute ทั้งหมดใน R มีฟังก์ชันการขึ้นอยู่กับ A
- 2) ในกรณีที่ A ประกอบด้วย Attribute หลาย ๆ ตัว จะต้องไม่มี Attribute ที่เป็นส่วนใดส่วนหนึ่งของ A ที่มีคุณสมบัติตั้งที่ระบุในข้อแรก

### 2.6.4 รูปแบบนอร์มัลไลซ์รีเลชัน

รูปแบบนอร์มัลทั้ง 5 รูปแบบนี้ถูกนิยามขึ้นมาโดย คอดด์ ในปี พ.ศ. 2515 แต่หลังจากนั้น 2 ปี รูปแบบระดับที่ 3 ได้ถูกนิยามให้มีความรัดกุมขึ้น โดย บอยส์ (Boyce) และคอดด์ จากนั้นจึงได้ตั้งชื่อรูปแบบใหม่นี้ว่า Boyce Codd Normal Form หรือ BCNF

#### 1) รูปแบบนอร์มัลระดับที่ 1

การปรับรีเลชันให้อยู่ในรูปแบบนอร์มัลระดับที่ 1 คือ การปรับจากรีเลชันที่ไม่นอร์มัล (Unnormalized Relation) ซึ่งได้แก่อรีเลชันที่มีข้อมูลในบางช่องมากกว่า 1 ค่า และจะต้องมีการเพิ่ม Attribute ของคีย์เสมอ โดยสามารถกำหนดได้ว่าคีย์ตัวใหม่จะประกอบด้วยคีย์เดิมผนวกกับ Attribute ที่ถือเป็นคีย์หลักของกลุ่มที่ซ้ำ

1.1) การแก้ไขข้อมูล ถ้าค่าของข้อมูลที่ซ้ำซ้อนกันไม่ครบถ้วน อาจก่อให้เกิดความขัดแย้งของข้อมูลที่ซ้ำซ้อนกันได้

1.2) การเพิ่มเติมข้อมูล จะทำได้ก็ต่อเมื่อลูกค้านั้นมีรายการสั่งซื้อสินค้าแล้วเท่านั้น ซึ่งหมายความว่า รีเลชันนี้จะไม่สามารถจัดเก็บข้อมูลของลูกค้าที่ยังไม่มีการสั่งซื้อได้

1.3) การลบข้อมูล ถอลบข้อมูลใน การสั่งซื้อก็จะทำให้ข้อมูลของลูกค้าบางคนสูญหายไปด้วย

นิยาม รีเลชันใด ๆ กล่าวได้ว่าอยู่ในแบบนอร์มัลระดับที่ 1 (1NF) ถ้ารีเลชันไม่มีกลุ่มที่ซ้ำกัน

## 2) รูปแบบนอร์มัลระดับที่ 2

ถึงจะจัดรูปแบบของรีเลชันให้เป็น 1 NF แล้วก็ตาม แต่ในบางครั้งก็ยังมีปัญหาต่าง ๆ เกิดขึ้น สังเกตจากตารางที่ 2.4 จะเห็นว่าเกิดความซ้ำซ้อนเมื่อมีการสั่งซื้อสินค้าชนิดเดียวกันขึ้นมา จะเห็นว่าข้อมูลที่เกี่ยวข้องกับ BT04 มีการซ้ำกัน ทำให้เปลืองเนื้อที่ และเกิดปัญหาในการแก้ไขข้อมูลซึ่งสามารถจัดแบ่งลักษณะของปัญหาออกได้เป็น 4 ประเภท คือ

2.1) การแก้ไขข้อมูล ถ้าเราต้องการแก้ไขข้อมูล เกี่ยวกับ BT04 เราจำเป็นต้องตามแก้ไขในหลายแถว นอกจากจะทำให้เสียเวลาแล้ว ยังไม่สามารถรับประกันได้ว่าการแก้ไขจะเกิดขึ้นครบทุกแถวที่มี BT04 ตามหลักแล้วการแก้ไขข้อมูลอย่างเดียวกันควรจะทำเพียงแห่งเดียว

ตารางที่ 2.4 ตัวอย่างรีเลชัน 1NF

รหัสการสั่ง	วันที่สั่ง	รหัสสินค้า	ชื่อสินค้า	จำนวนที่สั่ง	ราคา
12489	020941	AX12	เตารีด	2	280
12491	020941	BT04	เตา	1	8000
12494	020941	BZ66	เครื่องซักผ้า	5	30000
12500	050941	BT04	เตา	1	8500
12504	050941	CZ81	ค้อนน้ำหนัก	2	2160

2.2) ความขัดแย้งของข้อมูล เนื่องจากเรามีข้อมูลที่กล่าวถึง สินค้าที่มีชนิดเดียวกัน ปรากฏอยู่ในหลายแถว จึงมีความเป็นไปได้ว่าค่าของข้อมูลชนิดเดียวกันในแต่ละแถวอาจไม่ตรงกัน เช่น ในตาราง จะเห็นว่า ในบางแถวราคาของของเตา มีค่าเท่ากับ 8000 ในขณะที่อีกแถวหนึ่งระบุว่าเท่ากับ 8500 เป็นต้น

2.3) การเพิ่มเติมข้อมูล ถ้ามีการเพิ่มเติมข้อมูลลงไปในเรคคอร์ดเนื่องจากรีเลชันของเรากำหนดให้รหัสการสั่ง และรหัสสินค้าเป็นคีย์ร่วมกัน นั่นหมายความว่า การเพิ่มเติมรหัสสินค้าจะต้องมีรหัสการสั่งด้วย อาจทำให้เกิดปัญหาได้

2.4) การลบข้อมูล ถ้ามีการลบข้อมูลรหัสการสั่งออกไปจากตาราง ก็จะทำให้ระบบเสียข้อมูลของรหัสสินค้าไปด้วย

จะเห็นว่าตัวการที่ทำให้เกิดปัญหาก็คือ การที่ Attribute บางตัวขึ้นอยู่กับเพียงบางส่วนของคีย์ ดังนั้นหลักการของการทำให้เป็น 2NF ก็คือ การขจัดการขึ้นต่อกันเพียงบางส่วน

นิยาม เราเรียก Attribute ที่ไม่ใช่ส่วนหนึ่งของคีย์หลักว่า Attribute นันคีย์ (Nonkey Attribute)

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 2 (2NF) ถ้ารีเลชันนั้นเป็น 1NF และไม่มี Attribute นันคีย์ตัวใดขึ้นกับส่วนใดส่วนหนึ่งของคีย์

วิธีการที่จะทำให้รีเลชันอยู่ในรูปแบบของ 2NF กระทำได้ โดยการสร้างรีเลชันขึ้นมาใหม่สำหรับการขึ้นกันที่เป็นปัญหา

### 3) รูปแบบนอร์มัลระดับที่ 3

ถึงแม้เราจะได้นอร์มัลไลซ์ความสัมพันธ์ให้อยู่ในระดับที่ 2 แล้ว แต่ปัญหาการซ้ำซ้อนก็ยังคงมีอยู่ในบางครั้ง ดังตารางที่ 2.5 ดังนี้

ตารางที่ 2.5 รีเลชัน 2NF

รหัสลูกค้า	ชื่อ	ที่อยู่	รหัสพนักงาน	ชื่อพนักงานขาย
124	เคลียว	18 สุขสวัสดิ์	3	ธนา
256	ยงยุทธ	21 สุขุมวิท 49	12	ประจวบ
311	พิชัย	58 เจริญกรุง	6	มณีรัตน์
405	มณี	46 วิทญ์	12	ประจวบ
587	วัชรวิ	25 ลาดพร้าว	3	ธนา

3.1) การแก้ไขข้อมูล ถ้ามีการแก้ไขข้อมูลเกี่ยวกับพนักงานขาย จะต้องทำการแก้ไขหลายแห่ง

3.2) ความขัดแย้งของข้อมูล ผลสืบเนื่องจากการแก้ไขข้อมูลที่ปรากฏหลายแห่ง อาจจะทำให้แก้ไขไม่ครบถ้วนทุกแห่ง ทำให้ข้อมูลแต่ละแห่งไม่ตรงกัน

3.3) การเพิ่มเติมข้อมูล จะไม่สามารถเพิ่มพนักงานขายเข้าไปได้ ตราบใดที่พนักงานคนใหม่นี้ยังไม่มีลูกค้า

3.4) การลบข้อมูล ถ้ามีการลบข้อมูลเกี่ยวกับลูกค้าของพนักงานขายหมายเลข 12 ออกจากระบบ จะทำให้ข้อมูลของพนักงานขายคนนี้หายไปด้วย

นิยาม เราเรียก Attribute (หรือกลุ่มของ Attribute) ใด ๆ ก็ตามที่สามารถเลือก (Determine) Attribute ตัวอื่น ๆ ได้ว่า ตัวเลือก (Determinant)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 3 (3NF) ถ้ารีเลชันนั้นเป็น 2NF และตัวเลือกทุกตัวจะต้องเป็นคีย์คู่แข่ง

การจัดตัวเลือกที่ไม่ใช่คีย์คู่แข่งออกไปมีวิธีการดังนี้

- 1) ดึง Attribute ที่ขึ้นกับตัวเลือกที่ไม่ใช่คีย์คู่แข่งออกไปสู่รีเลชันใหม่
- 2) กำหนดให้ตัวเลือกที่เกี่ยวข้องเป็นคีย์ของรีเลชันใหม่นี้

#### 4) รูปแบบนอร์มัลแบบ Boyce-Codd Normal Form (BCNF)

การนอร์มัลไลซ์รีเลชัน จะต้องมีคุณสมบัติของ 3NF และ Attribute ที่เป็น Determinant จะต้องเป็น Relation Key

4.1) การแก้ไขข้อมูล จะต้องเปลี่ยนแปลงข้อมูลนั้นใน Table อื่นที่มีค่าของข้อมูลเดียวกันทั้งหมด

4.2) การเพิ่มเติมข้อมูล จะต้องเพิ่มข้อมูลให้กับ Attribute พร้อมๆ กันจึงจะสามารถกระทำได้

4.3) การลบข้อมูล ถ้ามีการลบข้อมูลออกไป อาจส่งผลทำให้ค่าใน Attribute ใด Attribute หนึ่งสูญหายไปด้วย

#### 5) รูปแบบนอร์มัลระดับที่ 4

การนอร์มัลไลซ์รีเลชัน ถ้าทำได้ถึง ระดับ 3NF แล้วจะสามารถแก้ไขปัญหาที่เกิดขึ้นได้จากฟังก์ชันการขึ้นต่อกันไปอย่างหมดสิ้น แต่จริง ๆ แล้วยังมีลักษณะของการขึ้นอยู่กันชนิดหนึ่งที่สามารถก่อให้เกิดปัญหาประเภทเดียวกันนี้ได้อีก ซึ่งการแก้ปัญหาก็จะต้องทำโดยการนอร์มัลไลซ์ให้ถึงระดับที่ 4

สาเหตุที่ทำการนอร์มัลไลซ์ให้ถึงระดับที่ 3 แล้วยังมีปัญหาเกิดขึ้นอีก เพราะไม่มีฟังก์ชันของการขึ้นต่อกันระหว่าง Attribute หรือจะมีก็แต่ลักษณะของการขึ้นต่อกันเชิงกลุ่ม

นิยาม ในรีเลชันที่ประกอบด้วย Attribute 3 ตัวคือ A, B และ C การขึ้นต่อกันเชิงกลุ่มระหว่าง B และ A โดย B ขึ้นต่อ A หมายความว่า ค่าหนึ่งค่าของ A จะอิงกับกลุ่มของ B โดยการขึ้นกันที่วันนี้จะเป็นอิสระกับค่าของ C เราสามารถแสดงการขึ้นต่อกันเชิงกลุ่ม โดย B ขึ้นต่อ A ได้ด้วยสัญลักษณ์  $A \twoheadrightarrow B$

นิยาม รีเลชันที่เป็น 4NF ได้แก่อรีเลชันที่เป็น 3NF และเป็นรีเลชันที่ไม่มีกรขึ้นต่อกันเชิงกลุ่ม

การปรับรีเลชันให้เป็น 4NF สามารถทำได้โดยการแยกกริเลชันเดิม โดยให้แต่ละรีเลชันประกอบด้วย Attribute ที่เกี่ยวข้องกันอยู่เท่านั้น

## 6) รูปแบบนอร์มัลระดับที่ 5

การนอร์มัลไลซ์ระดับที่ 5 ถ้าทำได้ถึง ระดับ 4NF และต้องมีคุณสมบัติ Join Dependency ซึ่งเป็นคุณสมบัติของการนำ Relation ย่อยที่เกิดจากการแตก Relation เดิมมารวมกันแล้วได้ข้อมูลเช่นเดียวกับ Relation เดิม

## 2.7 การออกแบบฐานข้อมูล

สิ่งสำคัญที่สุดประการหนึ่ง ในการประมวลสารสนเทศด้วยระบบฐานข้อมูลคือ การออกแบบฐานข้อมูล ความหมายของการออกแบบฐานข้อมูล คือการที่ผู้พัฒนาระบบจะต้องพิจารณาว่าเรคคอร์ดแต่ละตัวควรจะประกอบด้วยฟิลด์อะไรบ้าง แต่ละฟิลด์ควรจะมิชชนิดอะไร ขนาดเท่าไร เรคคอร์ดแต่ละชนิดควรมีความสัมพันธ์กันอย่างไร

โดยทั่ว ๆ ไปแล้ว การออกแบบระบบฐานข้อมูลสามารถแบ่งออกได้เป็น 2 ระดับ ระดับแรกเรียกว่า การออกแบบระดับสารสนเทศ (Information-Level Design) คือ ส่วนของการศึกษาวิเคราะห์รวบรวมความต้องการของผู้ใช้เอาไว้ โดยที่การออกแบบในระดับนี้มีเป้าหมายเพื่อให้การใช้งานเป็นไปตามความต้องการของผู้ใช้มากที่สุด ในจุดนี้จะไม่ให้ความสำคัญกับชนิด และยี่ห้อของ DBMS ที่จะใช้ โดยเราจะเริ่มศึกษาเกี่ยวกับ DBMS ที่จะใช้ใน ระดับที่ 2 ซึ่งเรียกว่าระดับกายภาพ (Physical-Level Design) อันเป็นระดับที่เราเริ่มให้ความสำคัญต่อประสิทธิภาพของระบบ

การออกแบบใน 2 ระดับนี้มีความสำคัญเท่าเทียมกัน เพราะว่าการออกแบบในระดับข่าวสารที่ไม่ดีย่อมจะมีผลกระทบต่อประสิทธิภาพของระบบด้วย และในขณะเดียวกัน ถึงแม้จะออกแบบในระดับข่าวสารไว้อย่างดีเลิศ แต่ถ้าการออกแบบในระดับกายภาพไม่ดีพอ ก็จะทำให้การใช้งานของระบบล้มเหลวไปด้วย

### 2.7.1 เป้าหมายของการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลมีเป้าหมายอยู่ที่การสร้างประสิทธิภาพของการใช้งานให้แก่ผู้ใช้งาน ดังนั้นจึงพอที่จะเดากันได้ว่าขั้นตอนแรกของการออกแบบข้อมูลก็คือการศึกษา วิเคราะห์และรวบรวมเอาความต้องการของผู้ใช้ให้สมบูรณ์ที่สุด ทั้งนี้นอกจากความต้องการแล้วผู้ออกแบบก็จะต้องรวบรวมกฎเกณฑ์และข้อบังคับต่าง ๆ เอาไว้ด้วย ซึ่งสามารถสรุปรายละเอียดต่าง ๆ ที่จำเป็นต้องทราบในส่วนนี้ได้ ดังนี้

- 1) ลักษณะของรายงานทั้งหมด
- 2) การค้นหาข้อมูลในทุกลักษณะ
- 3) เอาท์พุทที่ต้องส่งให้แผนกอื่น หรือระบบอื่น

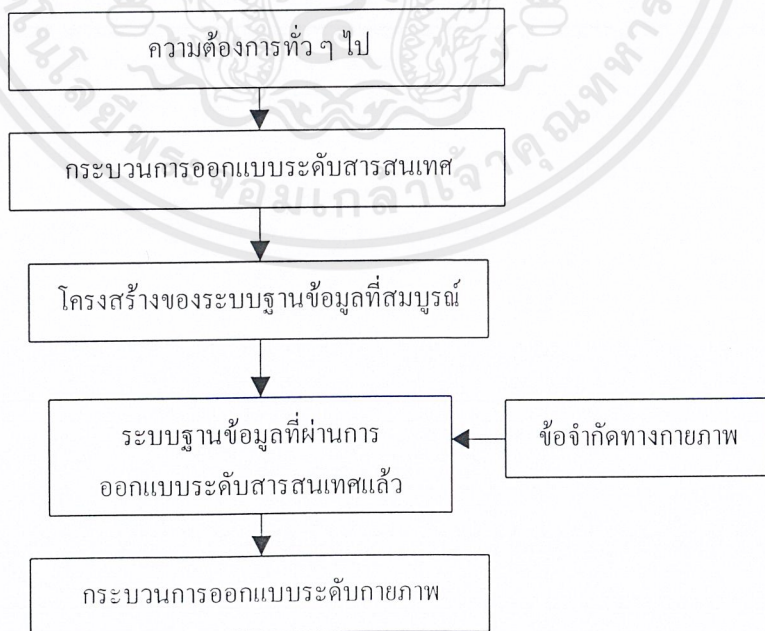
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4) การประมวลและแก้ไขข้อมูลทั้งหมด
- 5) การคำนวณทุกอย่าง
- 6) กฎเกณฑ์ข้อบังคับต่าง ๆ เช่น การห้ามไม่ให้ลูกค้าคนใดใช้บริการของพนักงานขายที่ไม่มีหมายเลขปรากฏอยู่ในตารางของพนักงานขาย
- 7) การตั้งชื่อพ้อง (Synonym) ต่าง ๆ เช่น ในแต่ละแผนกหรือผู้ใช้แต่ละคนอาจจะเรียกชื่อของ Attribute ตัวเดียวกันแตกต่างกัน

ข้อมูลเหล่านี้จะถูกนำมาประมวลในการออกแบบฐานข้อมูลในระดับสารสนเทศ และเมื่อถึงเวลาของการออกแบบในระดับกายภาพแล้ว เรายังต้องอาศัยข้อมูลอื่น ๆ ที่เกี่ยวข้องกับการทำงานของระบบในระดับกายภาพอีก เช่น

- 7.1) จำนวนของแต่ละ Entity
- 7.2) ความถี่ในการพิมพ์รายงาน
- 7.3) กฎเกณฑ์ในการควบคุมความปลอดภัยในการใช้ข้อมูล

ข้อมูลเหล่านี้จะถูกนำมาวิเคราะห์พร้อมกับระบบฐานข้อมูลที่ได้ออกแบบในระดับสารสนเทศไว้แล้ว เพื่อการออกแบบในระดับกายภาพ โดยในระดับนี้เราจะเริ่มคำนึงถึงความสามารถ DBMS ที่กำลังใช้อยู่ด้วยแล้ว เพื่อให้ผลการทำงานของระบบที่ออกแบบมานี้สมบูรณ์และมีประสิทธิภาพมากที่สุด โดยสรุปแล้วขั้นตอนของการออกแบบระบบฐานข้อมูลจะเป็นไปตามรูปที่ 2.10



รูปที่ 2.10 ขั้นตอนการออกแบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.2 ความต้องการของผู้ใช้แต่ละคน

ไม่ว่าผู้ออกแบบจะใช้เทคนิคในการออกแบบฐานข้อมูล การออกแบบเพียงครั้งเดียวให้ได้มาซึ่งระบบที่ถูกต้องสมบูรณ์ตามความต้องการของผู้ใช้แต่ละคนนั้นนับว่าเป็นเรื่องยาก ยกเว้นในกรณีของฐานข้อมูลขนาดเล็ก หลักการของการออกแบบฐานข้อมูลก็เหมือนกับการออกแบบและพัฒนาซอฟต์แวร์อื่น ๆ คือจะแบ่งงานใหญ่ออกเป็นงานย่อย ๆ หลาย ๆ ชิ้น และค่อย ๆ เริ่มออกแบบจากจุดเล็ก ๆ นี้ไปก่อน ก็คือเริ่มจากการออกแบบสำหรับความต้องการของผู้ใช้ทีละคนหรือกลุ่มย่อย ๆ ก่อน ซึ่งจะมีความยุ่งยากน้อยกว่า แล้วจึงเพิ่มขอบเขตของผู้ใช้ออกไป

### 2.7.3 หลักการพื้นฐานของการออกแบบฐานข้อมูล

ในกรณีที่ได้รวบรวมความต้องการของกฎข้อบังคับต่าง ๆ ไว้จากผู้ใช้ระบบเรียบร้อยแล้ว กระบวนการในการออกแบบมี 5 ขั้นตอน ดังนี้

- 1) เปลี่ยนรูปแบบของความต้องการให้อยู่ในลักษณะของรีเลชัน
- 2) นอร์มัลไลซ์รีเลชัน
- 3) กำหนดฟิลด์ที่จะเป็นคีย์ต่าง ๆ และคุณสมบัติของคีย์แต่ละตัว
- 4) พิจารณาข้อจำกัดต่าง ๆ และกฎเกณฑ์อื่น ๆ
- 5) นำผลที่ได้จากการออกแบบในขั้นตอนแรกมาผนวกกัน ซึ่งในแต่ละขั้นตอนจะมีรายละเอียดดังนี้

ขั้นตอนที่ 1 การเปลี่ยนรูปแบบของความต้องการให้อยู่ลักษณะของรีเลชัน

การสร้างรีเลชันก็คือการวิเคราะห์ว่าฐานข้อมูลควรมีรีเลชันอะไรบ้าง และในรีเลชันแต่ละตัวควรมีฟิลด์ใดเป็นคีย์ บางครั้งการออกแบบในส่วนนี้ออกจะตรงไปตรงมา และง่ายจนน่าตกใจ เช่น ถ้าเราต้องการเก็บข้อมูลเกี่ยวกับพนักงาน และแผนกที่พนักงานนั้นสังกัดอยู่ก็จะเห็นว่าฐานข้อมูลควรจะประกอบด้วยรีเลชัน 2 ตัวคือ แผนก และ พนักงาน ดังนี้คือ

แผนก (รหัสแผนก , ชื่อแผนก)

พนักงาน (รหัสพนักงาน , ชื่อ , ที่อยู่ , เงินเดือน , รหัสแผนก)

โดยมีฟิลด์ที่ขีดเส้นใต้เป็นคีย์ของรีเลชันนั้น ๆ

- 1) สร้างรีเลชันขึ้นมาสำหรับ Entity แต่ละตัว ต้องพิจารณาว่าฐานข้อมูลที่จะสร้างต้องใช้รีเลชันจำนวนเท่าใด
- 2) พิจารณาว่ารีเลชันแต่ละตัวควรจะใช้ฟิลด์ใดเป็นคีย์หลัก
- 3) พิจารณาคุณสมบัติของ Entity แต่ละตัว โดยศึกษาจากความต้องการของผู้ใช้ว่า Entity ว่าจะแต่ละตัวควรประกอบด้วยฟิลด์อะไรบ้าง

4) พิจารณาความสัมพันธ์ระหว่างแต่ละ Entity ในส่วนนี้ผู้ออกแบบจะต้องพิจารณาว่า Entity แต่ละตัวมีความสัมพันธ์กันอย่างไร

**ขั้นตอนที่ 2** นอร์มัลไลซ์รีเลชัน

ในขั้นตอนนี้เราจะนอร์มัลไลซ์รีเลชันแต่ละตัว โดยมีเป้าหมายให้เป็น 3NF

**ขั้นตอนที่ 3** กำหนดฟิลด์ที่จะเป็นคีย์ต่าง ๆ และคุณสมบัติของคีย์แต่ละตัว

กำหนดคีย์ทั้งหมดอันได้แก่ คีย์หลัก, คีย์คู่แข่ง, คีย์รอง และคีย์นอก ซึ่งการกำหนดว่าจะให้ข้อมูลนั้น รวมถึงความสัมพันธ์ระหว่าง Entity แต่ละตัว โดยเฉพาะอย่างยิ่งคีย์นอก เพราะผู้ออกแบบจะต้องใช้วิจารณ์ญาณ และเหตุผลในการตัดสินใจถึงการออกแบบจุดต่าง ๆ ซึ่งจะเป็นตัวตัดสินใจถึงความสัมพันธ์ที่จะเกิดขึ้นระหว่าง Entity ในฐานข้อมูล

**ขั้นตอนที่ 4** พิจารณาข้อจำกัดและกฎเกณฑ์อื่น ๆ

ในขั้นตอนนี้ ผู้ออกแบบจะต้องรวบรวมความต้องการจากผู้ใช้ว่ามีข้อจำกัดอะไรบ้าง เช่น อนุญาตให้นำลูกค้าคนใดมียอดเงินค้างชำระสูงกว่าวงเงินในเครดิต เป็นต้น ในลักษณะเช่นนี้ DBMS บางตัวจะอนุญาตให้เราระบุข้อจำกัดนี้ในส่วนของการสร้าง Schema ได้เลย แต่สำหรับ DBMS บางตัวที่ไม่มีขีดความสามารถนี้ ผู้ออกแบบก็จำเป็นต้องรายงานข้อกำหนดนี้ไว้ในโปรแกรมที่เกี่ยวข้องกับการประมวลผลข้อมูลนั้น ๆ

**ขั้นตอนที่ 5** นำผลที่ได้จากการออกแบบใน 4 ขั้นตอนแรกมาผนวกกัน

หลังจากที่เราได้ผลของการออกแบบทั้ง 4 ขั้นตอนสำหรับผู้ใช้นั้นหรือกลุ่มหนึ่งแล้ว เราจะนำการออกแบบนี้ไปผนวกกับการออกแบบที่สร้างขึ้นสำหรับผู้ใช้อื่น ๆ หรือกลุ่มอื่น ๆ

## 2.8 ภาษาสคริปต์ ASP (Active Server Page)

ภาษาสคริปต์ ASP เป็นรูปแบบการสร้างสรรค์เว็บรูปแบบหนึ่ง ที่นิยมใช้กันมากในการสื่อสารทางอินเทอร์เน็ต ซึ่งความสามารถของ ASP นั้นทำให้เว็บไม่เป็นเพียงหน้าเอกสารให้เราอ่านเท่านั้น แต่ยังสามารถเพิ่มลูกเล่นต่างๆ มากมาย เช่น การ Search หาข้อมูล, การคำนวณ ค่าใช้จ่ายในการซื้อขายผ่านอินเทอร์เน็ต, การใช้บริการต่างเช่นเมล์ฟรี และอื่นๆ อีกมากมายไม่จำกัด ซึ่งขึ้นอยู่กับความคิดสร้างสรรค์ต่างๆ ของเรา

ASP (Active Server Page) พัฒนาโดยบริษัทไมโครซอฟท์ โดยไฟล์ ASP จะมีองค์ประกอบ 3 ส่วนคือ

- 1) แท็ก HTML ซึ่งเป็นภาษาพื้นฐานในการทำเว็บที่เราๆ ได้รู้จักกันดี
- 2) Script language ทั่วๆไป ไม่ว่าจะเป็น VBScript, Jscript หรือแม้กระทั่ง Perl นอกจากนี้

ยังสามารถนำไปใช้กับ ActiveX ของ VB6 ได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) Object และ Component เราสามารถสร้างออบเจกต์ขึ้นมาเพื่อใช้ในการทำงานเฉพาะ  
 อย่งได้ เช่น การติดต่อฐานข้อมูล การเปิดปิดไฟล์ เป็นต้น

แต่ส่วนที่สามารถใช้ในการประมวลผลของ ASP จะมีอยู่ 2 ส่วนด้วยกันคือ

- 1) การเขียน Script (ActiveX Server Scripting)
- 2) การสร้างออบเจกต์ต่างๆ (ActiveX Server Component)

ส่วนภาษา HTML เราไม่นับเป็นการประมวลผลของ ASP เพราะภาษา HTML จะเป็นการ  
 ประมวลผลที่ฝั่ง Client

### 2.8.1 การเขียน Script (ActiveX Server Scripting)

เป็นวิธีหนึ่งในการประมวลผลของ ASP ให้เราดูโค้ดเพื่อทำความเข้าใจด้วยกัน ดังนี้  
ขั้นตอนที่ 1 เปิด Notepad แล้วสร้างโค้ดตามนี้

```
<Html>
<Head><Title>Test ASP</Title></Head>
<SCRIPT LANGUAGE = "VBScript">
  Sub PageStart
    MsgBox "Page loaded !"
  End Sub
< /SCRIPT>
<Body LANGUAGE = "VBScript" Onload ="PageStart">
  <% for n = 1 to 3%>
    <FONT SIZE =<%= n*3%>>Love me love Man U.</Font><BR>
  <%Next%>
< /Body>
< /Html>
```

แล้วบันทึกเป็นชื่อไฟล์ test.asp ในไดเรกทอรี C:\inetpub\wwwroot ซึ่งเป็น Home  
 Directory ในโปรแกรม PWS

ขั้นตอนที่ 2 เรียก Browser ในเครื่องของเรา แล้วพิมพ์ URL ดังนี้ <http://ชื่อเครื่อง/test.asp> จะแสดง  
 ผลลัพธ์ (ชื่อเครื่องของเราสามารถดูได้ที่ Control Panel>Network ที่เป็น Identification ในช่อง  
 Computer Name)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8.2 การสร้างออบเจกต์ต่างๆ (ActiveX Server Component)

ในหัวข้อที่ผ่านมาเราได้รู้จักการใช้ Script ในการช่วยประมวลผลมาแล้ว และในหัวข้อนี้เราจะมาดูการประมวลผลของ ASP อีกรูปแบบหนึ่ง คือการเรียกใช้ออบเจกต์ในการเรียกใช้เมธอดต่างๆ ในการทำงาน

**ออบเจกต์ (Object)** คือรูปแบบของการเขียนโปรแกรมชนิดหนึ่งที่มีผู้นิยมใช้อย่างแพร่หลายในปัจจุบัน ซึ่งรูปแบบการเขียนโปรแกรมนี้เราเรียกว่า Object Oriented Programming หรือที่มีผู้นิยมเรียกว่าการเขียนโปรแกรมเชิงวัตถุ แต่ใช้คำว่า การเขียน โปรแกรมสร้างวัตถุแทน

ก่อนอื่นเราคงต้องเข้าใจความหมายของวัตถุเสียก่อน โดยเราจะยกตัวอย่างวัตถุต่างๆ เช่น ลูกบอลซึ่งถือว่าเป็นวัตถุหนึ่งซึ่งมีคุณสมบัติ(Properties) ต่างๆ เช่น สีแดง มีรูปทรงกลม พื้นผิวอ่อนนุ่ม และมีรูปแบบการใช้งาน(Method) เช่น กลิ้ง โยน ปา และเตะได้นอกจากนี้ลูกบอลแต่ละลูกยังเป็นอิสระต่อกัน ถ้าลูกบอลลูกบอลหนึ่งแตก อีกลูกก็ไม่จำเป็นต้องแตกตาม เป็นต้น

**พรอปเพอร์ตี้ (Properties)** คือลักษณะต่างๆ ของ Instance เช่น ชื่อ Instance, ขนาด Instance

**เมธอด (Method)** คือฟังก์ชันการทำงานต่างๆ ของ Instance เช่น เปิดปิดไฟได้

ลองดูตัวอย่างโปรแกรมที่มีลักษณะเป็นวัตถุเพื่อให้เข้าใจการทำงานมากขึ้น ลองเปิดหน้าต่างต่างๆ Browser ขึ้นมา 3 หน้าต่าง และลองไปเว็บไซต์ต่างๆ ที่ไม่ซ้ำกันจะพบว่าแต่ละ Browser เป็นอิสระต่อกัน และนอกจากนี้ยังมีคุณสมบัติเฉพาะตัวที่แตกต่างกัน เช่น ขนาดของวินโดวส์ ชื่อของหน้าต่าง Browser เป็นต้น

จากที่กล่าวมาทั้งหมด เป็นเพียงพื้นฐานที่ช่วยให้เราเข้าใจการทำงานในส่วนของออบเจกต์มากขึ้น แต่ในการเขียนโปรแกรม ASP นั้นไม่จำเป็นต้องสร้างโปรแกรม Object Oriented ขึ้น เพราะว่าใน ASP มีออบเจกต์ให้เราเรียกใช้พร้อมอยู่แล้ว เพียงแต่นำออบเจกต์ต่างๆ นั้นมาใช้งานโดยการกำหนดคุณสมบัติและเมธอดเท่านั้นเอง เราลองมาดูตัวอย่างการเรียกใช้ออบเจกต์ในโปรแกรม ASP กัน

```
<% for I = 1 to 5
    response.write ("Love me love Man U.<br>")
next
%>
```

จากโค้ดตัวอย่างเป็นการใช้ออบเจกต์ response เรียกเมธอด write ในการพิมพ์ประโยค

“Love me love Man U.<br>” 5 ใน ASP มีออบเจกต์ต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 2.6 ออบเจกต์ต่างๆ ใน ASP

Object	Component
Application	Database Access
Session	File Access
Request	Browser Capabilities
Response	Ad Rotator
Server	Content Linking

จากตาราง ASP ได้เตรียมออบเจกต์ให้เรา 5 ตัว และคอมโพเนนต์ให้อีก 5 ตัว ซึ่งการใช้งานออบเจกต์นั้นเราสามารถเรียกใช้งานได้เลย(คงโค้ดตัวอย่างด้านบน) แต่การใช้งานคอมโพเนนต์นั้นเราต้องใช้คำสั่ง Create Object แปลงคอมโพเนนต์นั้นเป็นออบเจกต์เสียก่อนเราจึงจะสามารถเรียกใช้งานได้

ในการเขียนโปรแกรม ASP นั้นจะเกิดจากการผสมผสานกันระหว่าง Script และ ActiveX Component ซึ่งส่วน ActiveX Component นั้นใช้ในการทำงานด้านต่างๆ เช่น การเปิดไฟล์, การติดต่อ Database เป็นต้น และอีกส่วนของ Script ซึ่งส่วนที่ใช้ในการควบคุมการทำงานของโปรแกรม เช่นการทำซ้ำ การเลือกเงื่อนไข การกำหนดตัวแปร

### 2.8.3 ความสามารถของ Script

Script คือ โครงสร้างภาษาที่มีรูปแบบง่ายๆ และมีความสามารถในการทำงาน ดังนี้

- 1) ใช้ในการสร้างและกำหนดค่าตัวแปร
- 2) ส่งค่าตัวแปรผ่านไปยัง Browser ฟังก์ชัน Client เพื่อแสดงผลลัพธ์
- 3) เป็นชุดคำสั่งที่ใช้ควบคุมการทำงานเช่น การใช้เงื่อนไข การทำซ้ำ

ใน ASP เราสามารถเขียน Script ได้จากภาษาต่างๆ ไม่ว่าจะเป็น VBScript, Jscript, Javascript และ PerlScript ก็ได้ ซึ่งถ้าเราเขียนโดยไม่กำหนดภาษาที่นำมาใช้ ASP จะตีความว่าเราใช้ภาษา VBScript ในการเขียน ก็จะใช้การเขียนโค้ดคำสั่งโดยใช้ VBScript เป็นหลัก ในการเขียน Script ภาษาใดๆ นั้น ต้องตรวจสอบว่าเครื่องฝั่งเซิร์ฟเวอร์ที่จะประมวลผลจะต้องมีตัวแปลภาษานั้นเพื่อใช้ในการ Compile และรันอยู่ด้วย

## 2.8.4 โครงสร้างของ VBScript

เราอาจสงสัยว่า VBScript กับ VB (Visual Basic) แตกต่างกันอย่างไรมาก่อนแล้ว VBScript ก็คือ VB นั่นเอง เพียงแต่เป็นบางส่วนหรือซบเซตที่อยู่ภายใน VB เท่านั้น ดังนั้นโครงสร้างการเขียนจึงไม่ได้แตกต่างกันจาก VB เท่าไร ทำให้ผู้ที่มีความรู้พื้นฐานการเขียนโปรแกรมด้วย VB มาก่อนก็สามารถทำความเข้าใจกับ VBScript ได้ไม่ยาก

### 1) ชนิดของตัวแปร

ใน VBScript จะสนับสนุนตัวแปรเพียงชนิดเดียวคือ Variant ซึ่งเป็นตัวแปรที่สามารถแสดงได้ทั้งข้อมูลตัวเลขและตัวอักษร โดยข้อมูลต่างๆ ที่สามารถเก็บในตัวแปร Variant ได้มีดังนี้

ตารางที่ 2.7 ชนิดของตัวแปร

ชนิดของข้อมูล	ความหมาย
Empty	ข้อมูลชนิดว่างคือเป็นค่า 0 สำหรับข้อมูลตัวเลขและค่า zero-length string(“”)สำหรับ String
Null	เป็นข้อมูลที่ถือว่าไม่มีค่าใดๆ
Boolean	เป็นข้อมูลทางตรรกะคือ True หรือ False
Byte	ข้อมูลจำนวนเต็ม (8bit) 0 ถึง 255
Integer	ข้อมูลจำนวนเต็ม (16bit) ตั้งแต่ -32,768 ถึง 32,767
Long	ข้อมูลจำนวนเต็ม (32bit) ตั้งแต่ -2,147,483,648 ถึง 2,147,483,647 และเลขทศนิยม(แบบสั้น) ตั้งแต่ 3.402823E38 ถึง -1.401298E-45 สำหรับค่าลบและ 1.401298E-45 ถึง 3.402823E38 สำหรับค่าบวก
Double	ข้อมูลเลขทศนิยม (แบบยาว) เริ่มตั้งแต่ ค่าลบ -1.79769313486232E308 ถึง 1.79769313486232E308
Date(Time)	ใช้เก็บวันที่ตั้งแต่ 1 มกราคม ค.ศ. 100 ถึง 31 ธันวาคม 9999
String	สตริงใช้เก็บตัวอักษร ได้ถึงสองล้านตัวอักษร
Object	เป็นตัวแปร Object
Error	จะเป็นชนิดของข้อมูลที่ใช้ในการอ้างอิงถึงคอนโทรลต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2) การประกาศตัวแปร (Variable Declaration)

การประกาศตัวแปร จะเป็นการบอกโปรแกรมว่าเราต้องการนำตัวแปรนี้มาใช้ โดยการประกาศตัวแปร จะใช้คำสั่งดังรูปแบบต่อไปนี้

```
Dim<ชื่อตัวแปร>
```

## 3) สำหรับการตั้งชื่อตัวแปร ต้องตั้งชื่อตามกฎต่อไปนี้

- 3.1) ชื่อตัวแปรต้องไม่ซ้ำกันในโปรแกรมเดียวกัน
- 3.2) ชื่อต้องไม่ซ้ำกับคีย์เวิร์ด (คำสงวน) ของ VBScript เช่นคำว่า Dim, Integer เป็นต้น
- 3.3) ความยาวต้องไม่เกิน 255 ตัวอักษร
- 3.4) จะต้องเริ่มต้นด้วยอักษร A-Z หรือ a-z

ตัวอย่างการประกาศตัวแปร

```
Dim Name, LastName
```

จากโค้ดเป็นการสร้างตัวแปร 2 ตัวชื่อ Name และ LastName เป็นชนิด Variant

## 4) ชนิดของการประกาศตัวแปร

การประกาศตัวแปรใน VBScript จะมีอยู่ 2 แบบ คือ

- 4.1) การประกาศตัวแปรโดยละคำสั่ง Dim

```
Name = "ASP is easy!"
```

```
Num = 1234567890
```

จากคำสั่งข้างต้น เราละคำสั่ง Dim ในการประกาศตัวแปร และใช้การกำหนดค่าให้กับตัวแปรแทน ซึ่งถ้าตัวแปรที่ถูกกำหนดค่ายังไม่ถูกประกาศ คำสั่งนี้ก็จะทำการประกาศแลกำหนดค่าตัวแปรให้ด้วย

- 4.2) การประกาศตัวแปรโดยใช้คำสั่ง Dim เหมือนที่กล่าวมาแล้ว

### 2.8.5 ข้อดีของ ASP

1) ASP ทำให้เว็บของคุณเป็นเว็บแบบไดนามิก (Dynamic) นั่นคือรูปแบบเว็บที่แสดงผลออกมานั้นสามารถเปลี่ยนแปลงได้ ตามข้อมูลที่ ASP ได้รับ เช่นตัวอย่างการ Search ข้อมูลในเว็บไซค์ Yahoo ที่ผลลัพธ์จะเปลี่ยนไปตามคำที่เราใช้ Search

2) เพิ่มความเร็วในการดูเว็บนั้น เรามักจะสูญเสียเวลาส่วนใหญ่มากกับการรอข้อมูลที่มาจากรินเตอร์เน็ต ยิ่งข้อมูลยิ่งมากยิ่งรอนาน ซึ่ง ASP สามารถช่วยในจุดนี้ได้ กล่าวคือ ASP จะทำการคำนวณต่างๆ จนเสร็จและส่งเฉพาะผลลัพธ์ที่เราต้องการเท่านั้น ทำให้ปริมาณการส่งข้อมูลน้อยลงเราก็เสียเวลารอข้อมูลน้อยลงและสามารถดูเว็บได้เร็วขึ้น

3) เพิ่มความปลอดภัยให้ระบบของเรา ในการเขียนโปรแกรมต่างๆ บางครั้งเราต้องอ้างถึงไอดีเรกทอรีเก็บฐานข้อมูล อย่างเช่นเว็บไซต์ Yahoo เป็นต้น ซึ่งการใช้ ASP ไอดีเรกทอรีต่างๆ จะไม่ถูกแสดงที่ฝั่งผู้ดูเว็บเราได้ง่าย และป้องกันผู้ไม่หวังดีมาเจาะระบบของเราด้วย

4) ลดปัญหาความสามารถของเครื่องที่ใช้ดูเว็บ เนื่องจาก ASP จะส่งเฉพาะผลลัพธ์สุดท้ายมาแสดงผลเท่านั้น ดังนั้นไม่ว่าเครื่องคุณจะทันสมัยหรือล้าสมัยเพียงใด ก็ไม่ทำให้เวลาที่ใช้เปิดดูเว็บแตกต่างกันมากนัก เพราะว่าการประมวลผลทั้งหมดเสร็จสิ้นที่ฝั่ง Server แล้ว

## 2.9 การสร้างฐานข้อมูลด้วย Microsoft Access

เริ่มแรกจะมาทำการสร้างฐานข้อมูลโดยใช้ Microsoft Access ซึ่งจะเป็น Access97 หรือ Access2000 ก็ได้ แต่ในตัวอย่างที่เราจะแสดงนั้นจะเป็นการใช้ Access97 ในการเริ่มต้นสร้างฐานข้อมูลโดยให้เราเปิดโปรแกรม Access และทำตามขั้นตอนของการสร้างฐานข้อมูล

### 2.9.1 ส่วนประกอบในฐานข้อมูล Access

ในฐานข้อมูลของ Access จะมีส่วนประกอบย่อยๆ ตามแต่ละแท็บที่ปรากฏในวินโดว์ Database ดังนี้

1) Table คือ “ตาราง” ที่ใช้เก็บข้อมูลจริงแต่ละอย่าง เช่น ในฐานข้อมูลของบริษัท รายชื่อและที่อยู่ของลูกค้าก็จะเป็นตารางหนึ่ง รายชื่อพนักงานก็เป็นอีกตารางหนึ่ง แต่ในที่นี้จะขอใช้ทับศัพท์ว่า Table เพื่อไม่ให้สับสนกับลักษณะการจัดหรือนำเสนอข้อมูลเป็นตารางที่อาจพบได้ในระหว่างการทำงานกับข้อมูลบนแบบฟอร์มหรือรายงานต่างๆ ต่อไป

2) Query คือ “ตารางเสมือน” ที่เกิดจากการดึงข้อมูล Table เดิมออกมาเป็นเสมือนกับ Table ใหม่อีกอันหนึ่ง โดยการดึงข้อมูลนี้อาจดึงมาตรงๆ เหมือน Table เดิมเลยก็ได้ (ซึ่งก็ไม่จำเป็นต้องทำ เพราะไม่เห็นมีประโยชน์อะไร) หรือดึงโดยมีการกำหนดเชื่อมโยงเพิ่มเติมเข้าไปว่าให้ดึงข้อมูลประเภทไหนบ้าง (เช่นดึงเฉพาะชื่อลูกค้าที่ขึ้นด้วย “บริษัท”) หรือดึงแล้วนำมาจัดเรียงอย่างไร (เช่นเรียงตามที่อยู่หรือรหัสไปรษณีย์) หรือมีการหาผลรวม , ค่าเฉลี่ย , ค่ามากที่สุด/น้อยสุด ฯลฯ ของแต่ละพวกแล้วนำมาแสดงเหมือนกับเป็น Table ใหม่อีกอันหนึ่ง แต่ถ้ามีการแก้ไขข้อมูลใด ๆ ใน Query ก็จะทำกับไปแก้ไขข้อมูลจริงใน Table นั้นเอง

3) Form คือ “แบบฟอร์ม” ที่ใช้สำหรับแสดงและให้กรอกหรือแก้ไขข้อมูลบนจอภาพ โดยตรงนั่นเอง แบบฟอร์มนี้อาจประกอบด้วยชิ้นส่วนต่างๆเป็นจำนวนมาก บ้างก็เป็นชิ้นส่วนที่ใช้เพื่อประดับหรือบอกกล่าวถึงการใช้งาน Form นั้นๆ เช่น ชื่อของ Form หรือลวดลาย เส้นกรอบ ที่ใช้แบ่งข้อมูลออกจากกัน บ้างก็เป็นพื้นที่สำหรับแสดงและให้แก้ไขข้อมูลจริงใน Table

หรือ Query ที่เกี่ยวข้องกัน หรือข้างก็เป็นตัวรับคำสั่งหรือการกระทำเพื่อเชื่อมโยงกับการทำงานอื่นๆ เช่น ปุ่มกด หรือข้อความที่เป็นไฮเปอร์ลิงก์ (ดูบทที่ว่าด้วย Form) ซึ่งเมื่อคลิกตรงนั้นๆ ก็จะมีการทำงานเกิดขึ้นตามที่ตั้งไว้ เป็นต้น

4) Report คือ “รายงาน” ที่ใช้แสดงข้อมูลต่างออกมาให้ดูได้ทั้งบนจอภาพและพิมพ์ออกบนกระดาษ แต่ไม่มีที่ใช้แก้ไขใดๆ (ถ้าจะให้แก้ไขได้ต้องใช้ Form แทน) เพียงแต่นำเสนอผลลัพธ์เท่านั้น แต่ Report จะมีข้อความสามารถในการรวบรวมผลและนำเสนอข้อมูลสรุปแยกเป็นกลุ่มๆ ได้ดีกว่า Form

5) Macro คือ “ชุดคำสั่ง” ย่อยๆ ที่ใช้เสริมการทำงานต่างๆ ภายใน Access 97 โดยใช้คำสั่งต่างๆ ภายใน Access 97 เองมาเก็บเป็นชุดและสั่งการโดยอัตโนมัติคล้ายๆ กับแมโครของ Excel หรือ Word นั่นเอง เพียงแต่เป็นการนำแมโครที่สร้างขึ้นมาแสดงไว้ให้เรียกใช้ได้ในวินโดว Database โดยตรง

6) Module คือ “โปรแกรมย่อย” ที่เขียนขึ้นเป็นภาษา Access Basic หรือ Visual Basic for Application (VBA) ซึ่งจะใช้สำหรับการทำงานอัตโนมัติต่างๆ ที่ซับซ้อนเกินกว่า Macro จะทำได้

### 2.9.2 การติดต่อฐานข้อมูลโดยใช้ ODBC Data Sources

ODBC Open Database Connectivity เป็นมาตรฐานหลักที่ให้เชื่อมต่อกับฐานข้อมูลชนิดต่างๆ ซึ่งจะเป็นประโยชน์สำหรับนักพัฒนาโปรแกรมที่จะสร้างโปรแกรมที่สามารถรองรับฐานข้อมูลต่างๆ ได้โดยการเขียนโปรแกรมเพียงครั้งเดียว

หลักการทำงานกับ ODBC นั้น ต้องเขียนโปรแกรมเพื่อติดต่อกับ ODBC ให้ได้เสียก่อน จากนั้นส่วนของการจัดการฐานข้อมูลจะเป็นหน้าที่ของ ODBC

จากนั้นการทำงานในลักษณะนี้จะเห็นได้ว่าทำการเปลี่ยนชนิดของฐานข้อมูล ก็จะไม่มีการแก้ไขโปรแกรมเพราะว่าโปรแกรมที่เราเขียนนั้นติดต่อผ่าน ODBC ไม่ได้ติดต่อกับฐานข้อมูลโดยตรงดังนั้นถึงมีการแก้ไขใดๆ ที่ฐานข้อมูลก็ไม่จำเป็นต้องแก้ไขโปรแกรมที่เขียน เพียงแต่แก้ไขที่ตัว ODBC เท่านั้น

## 2.10 วิซวลอินเตอร์เดฟ (Visual InterDev)

วิซวลอินเตอร์เดฟ (Visual InterDev) คือ ซอฟต์แวร์ที่ช่วยสร้าง และพัฒนาเว็บเพจให้มีประสิทธิภาพด้วยวิธีการที่สะดวกและรวดเร็ว พร้อมกับความสามารถที่ทำได้มากมาย เช่น การสร้างสคริปต์เพื่อรันที่ฝั่งไคลเอนต์ การสร้างไฟล์ ASP เพื่อติดต่อกับงานฐานข้อมูล หรือแม้แต่การ

สร้าง ActiveX Controls บนเว็บเพจ เหล่านี้สามารถกระทำได้ด้วยวีซาร์ดที่จัดเตรียมไว้ให้แล้วได้ ง่าย ๆ โดยปกติแล้วเราสามารถทำการติดตั้งวิวอลอินเตอร์เฟสไว้ที่ฝั่งไคลเอนต์เพื่อใช้ในการพัฒนาเว็บเพจได้ โดยจะมีการซิงโครไนซ์ (Synchronize) กันกับเซิร์ฟเวอร์ในแต่ละเว็บเพจที่สร้างขึ้น ทำให้สามารถทดสอบผลของการสร้าง หรือแก้ไข ได้ทันทีหรืออาจติดตั้งไว้ที่ฝั่งของเซิร์ฟเวอร์ โดยตรงก็ได้

### 2.10.1 การเตรียมการกับวิวอลอินเตอร์เฟส

วิวอลอินเตอร์เฟส เป็นหนึ่งในซอฟต์แวร์ที่อยู่ในชุดของวิวอลสตูดิโอ (Visual Studio) ซึ่งในช่วงของการติดตั้ง สิ่งที่จะต้องจำสำหรับฝั่งเซิร์ฟเวอร์คือ เว็บเซิร์ฟเวอร์ ซึ่งสามารถทำการติดตั้งในระบบที่แตกต่างกันดังนี้

ตารางที่ 2.8 เปรียบเทียบการติดตั้งเว็บเซิร์ฟเวอร์ในระบบต่าง ๆ

ระบบปฏิบัติการ	เว็บเซิร์ฟเวอร์
Windows NT 4.0 Sever	Internet Information Sever 3.0 ขึ้นไป + ASP
Windows NT 4.0 Workstation	Peer-to-Peer Web Sever
Windows 95	Personal Web Server (PWS)
Windows 98	เลือกติดตั้ง Personal Web Server จากการติดตั้ง Windows 98 ได้

สำหรับ Windows NT Server 4.0 นอกจาก IIS 3.0 (Internet Information Server) หรือเวอร์ชันที่สูงกว่าแล้ว ยังต้องการ ASP ซึ่งเป็น Add-in ที่ช่วยเสริมการทำงานของ IIS ให้สามารถรับแอฟพลิเคชัน ASP ได้ ส่วน Windows NT Workstation 4.0 ใช้เพียง (Peer-to-Peer Web Server) ซึ่งได้รวมความสามารถของ ASP ไว้ในตัวอยู่แล้วก็สามารถทำการรับแอฟพลิเคชัน ASP ได้ทันที สำหรับ Windows 95 Thai Edition คงต้องใช้ PWS สำหรับเวอร์ชันภาษาไทยโดยเฉพาะด้วย ส่วน Windows 98 นั้นมี PWS อยู่ในตัวแล้ว

นอกจากนี้แล้ววิวอลอินเตอร์เฟสยังสามารถลงโปรแกรมเสริมอื่นๆ เพื่อช่วยให้การสร้างเว็บเพจมีลูกเล่นมากขึ้นได้ เช่น Image Composer เป็นเครื่องมือสำหรับการทำงานตกแต่ง รูปภาพ Media Manager ช่วยให้ Windows Explorer มีความสามารถกับ Media File เพิ่มมากขึ้น และ Music Producer ช่วยในการทำงานเกี่ยวกับระบบเสียงเป็นไปโดยง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10.2 การพัฒนาเว็บเพจโดยใช้วิซวลอินเทอร์เฟส

**Workspace Pane** ใช้แสดงโครงสร้างหลักต่างๆของการทำงาน ซึ่งจะแบ่งออกเป็นแท็บ ได้แก่

1) แท็บ Info View เป็นแท็บเริ่มต้นไว้แสดงเอกสารอ้างอิงต่างๆ ที่ช่วยในการสร้างเว็บเพจด้วยวิธีการที่หลากหลาย ซึ่งจะได้อธิบายในส่วนถัดไป

2) แท็บ File View เพื่อแสดงไฟล์ที่เกี่ยวข้องสำหรับโปรเจกต์นั้นๆ

3) แท็บ Data View ในกรณีใช้งานเกี่ยวข้องกับฐานข้อมูล

**Windows Pane** ใช้แสดงรายละเอียดที่ได้จากการเลือกเปิดไฟล์ หรือเอกสารจากทางด้าน Workspace Pane โดยจะแสดงให้เห็นข้อมูลที่มีอยู่และสามารถทำการแก้ไขข้อมูลที่เปิดขึ้นมาได้เสมือนเป็นอิดิเตอร์ (Editor) ของแต่ละประเภทไปในตัว แต่ละไฟล์ที่เปิดถือเป็นหน้าต่างต่างโดยปริยาย

วิซวลอินเทอร์เฟส สามารถสร้างเว็บเพจให้มีความสามารถได้หลากหลาย แต่ยังคงไว้ซึ่งความเป็นเครื่องมือพัฒนาเว็บเพจที่มีความง่าย สะดวกต่อการใช้งาน โดยพยายามลดความยุ่งยากในส่วนของการเขียนโค้ดที่ซับซ้อนออกไป

การเตรียมวิธีการใช้งานในแต่ละลักษณะที่สำคัญไว้มากมาย เพื่อเป็นพื้นฐานต่อการใช้งานในขั้นที่สูงขึ้นต่อไป โดยประกอบไปด้วย

1) การสร้างเว็บโปรเจกต์ เพื่อกำหนดขอบเขตการทำงานทั้งหมดให้เกิดขึ้น

2) การสร้างไฟล์ HTML นอกจากจะมีเท็กซ์อิดิเตอร์ปกติแล้ว ยังมี Frontpage อิดิเตอร์ ที่ช่วยให้การสร้างเว็บเพจที่เกี่ยวข้องกับตาราง หรือแม่แต่ฟอร์ม เป็นไปอย่างง่ายดาย

3) การสร้างไฟล์ ASP เช่นเดียวกับเท็กซ์อิดิเตอร์ที่มีให้ ซึ่งเรียกว่า Source Editor แยกโค้ดแต่ละประเภทออกเป็นสีสันทึบให้เห็นอย่างชัดเจน ทำให้ไม่สับสนในขณะที่แก้ไขโค้ด

4) การสร้างไฟล์ Layout ด้วย Layout Editor ทำให้สามารถสร้างไฟล์ Layout และเพิ่มความสามารถของเว็บเพจธรรมดา ให้มีลูกเล่นเพิ่มเติมมากยิ่งขึ้น

5) การสร้าง ActiveX Controls และ Design-Time Controls ความสามารถที่โดยเด่นทำให้เว็บเพจแสดง ActiveX ควบคุมด้วยวิธีการที่ง่ายอย่างไม่น่าเชื่อรวมถึงการใช้ Design-Time สำหรับงานติดต่อกับฐานข้อมูล ด้วยการกำหนดค่าเพียงไม่กี่ที่เท่านั้น

6) การทำงานร่วมกับฐานข้อมูล การติดต่อกับฐานข้อมูลผ่าน ODBC ด้วย ADO (ActiveX Data Objects) และ การใช้ Query Builder ช่วยเพิ่มประสิทธิภาพการทำงาน ร่วมกับฐานข้อมูลมากยิ่งขึ้น

7) การใช้ Script Wizard สคริปต์ที่ยุ่งยากสามารถสร้างอย่างง่ายดาย ด้วยวิซาร์ดที่เตรียมไว้

ให้ จึงเหมาะกับการสร้างสคริปต์ที่ทำการรันเพื่อตรวจสอบข้อมูลก่อนส่งไปยังเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 8) การสร้าง Form Wizard ออกแบบส่วนติดต่อผู้ใช้ด้วยวิซาร์ด เพียงทำตามแต่ละขั้นตอนก็สามารถสร้างเว็บเพจให้ติดต่อกับฐานข้อมูลได้ทั้งแก้ไข เพิ่ม หรือลบเร็คคอร์ด เป็นไปโดยอัตโนมัติ
- 9) การใช้ Link View เพื่อดูแลและจัดการเว็บไซต์ โดยสามารถตรวจสอบแต่ละไฟล์ที่เกี่ยวข้อง หรือลิงก์ที่ถูกเชื่อมโยงของแต่ละเว็บเพจให้เป็นไปอย่างสมบูรณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบ การสร้างและการทำงาน

ECG On Internet นี้การออกแบบ และการสร้างนั้นสามารถแบบออกเป็น 3 ส่วนด้วยกันดังต่อไปนี้

1. การสร้างฐานข้อมูลต่างๆ
2. การออกแบบโฮมเพจด้วยภาษา HTML
3. การทำการเชื่อมต่อฐานข้อมูลกับโฮมเพจโดยใช้ภาษา ASP

#### 3.1 การสร้างฐานข้อมูลต่างๆ

การสร้างฐานข้อมูลต่างๆในโครงงานนี้สร้างขึ้นมาจากโปรแกรม Microsoft Access โดยมีฐานข้อมูลทั้งหมดดังนี้

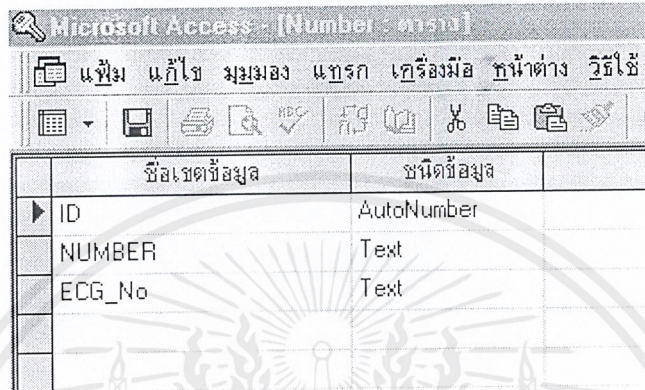
Bi_Polar_TR_1_39	Phase Control-SCR	ตาราง_Sockets
Bidirectional Diode Thyristors (SYDAC)	PIN Diodes_111	ตาราง2_IC
Bilateral Trigger Diodes (DIAC's)	Power_MOSFETS_1_67	
Bridge Rectifiers	PRODUCK_MAIN1	
Complementary_Pairs_1_61	Quartz Crystals	
Complementary_Pairs_1_611	RF Transistor	
Darlington_Power_1_52	SCR Phase	
Diodes and Rectifiers	Silicon Bilateral Switch (SBS)	
Fast Recovery_120	Silicon Controlled Switch (SCS)	
Gate Turn-Off	Silicon Unijunction Transistor (UJT)	
Germanium_T_1_85	Silicon_Power_1_74	
Heat Sink	Silicon_Small_(cont'd)	
High Speed	Silicon_Small_1_81	
High Voltage Devices	Sockets	
HV Rectifiers Microwave	Surge Arresters	
HV Selenium	Thermal Cutoffs	
HV Solid - Silicon	TRIACS	
IC Protectors	TTL Compatible Photo Coupled Logic Gates_160	
Industrial Power Modules	TV Horizontal Deflection	
Industrial Rectifiers	TV Power Supply	
Insulated_1_72	Ultra Fast Recovery_120	
Integrated Thyristor/Rectifier (ITR)	Voltage Regulator	
Matched_1_61_2	Zener_Diodes	
Microwave Mixer Diodes_111	ตาราง_Darlington	
Number	ตาราง_DISCONTINUED	
Optoisolators	ตาราง_FET	
Oversvoltage (MOV)	ตาราง_Heat_Sinks	
Oversvoltage Unidirectional (DC)	ตาราง_MOSFET	

รูปที่ 3.1 ฐานข้อมูลทั้งหมดที่ใช้ติดต่อกับ โฮมเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของตารางทั้งหมดจะมีส่วนที่สำคัญอยู่ 3 ส่วนคือ

1) ส่วนของตาราง NUMBER คือตาราง NUMBER นั้นจะเป็นตารางที่ทำการเก็บข้อมูลของเบอร์ และ ECG NUMBER ทั้งหมดของ ECG โดยมีรายละเอียดดังรูป



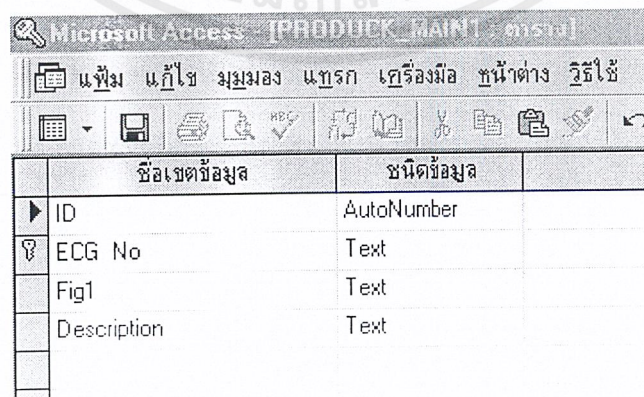
ชื่อเขตข้อมูล	ชนิดข้อมูล
ID	AutoNumber
NUMBER	Text
ECG_No	Text

รูปที่ 3.2 รายละเอียดของ ตาราง NUMBER

โดยกำหนดให้

ID หมายถึง ค่าลำดับที่ของเบอร์ที่อยู่ภายในตาราง NUMBER  
 NUMBER หมายถึง เบอร์ของตัวอุปกรณ์ทั้งหมด  
 ECG\_No หมายถึง ECG NUMBER ของตัวอุปกรณ์ทั้งหมด

2) ส่วนของตาราง PRODUCT\_MAIN1 คือตารางที่แสดงคุณสมบัติทั้งหมดของตัวอุปกรณ์ตั้งแต่แสดงรายละเอียดดังรูป



ชื่อเขตข้อมูล	ชนิดข้อมูล
ID	AutoNumber
ECG No	Text
Fig1	Text
Description	Text

รูปที่ 3.3 รายละเอียดของ ตาราง PRODUCT\_MAIN1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยกำหนดให้

ID	หมายถึง	ค่าลำดับที่ของเบอร์ที่อยู่ภายในตารางPRODUCT_MAIN1
ECG_No	หมายถึง	ECG NUMBER ของตัวอุปกรณ์ทั้งหมด
Fig1	หมายถึง	รูปภาพ Fig ต่างๆของตัวอุปกรณ์ทั้งหมด
Description	หมายถึง	คุณสมบัติของตัวอุปกรณ์ทั้งหมด

3) ส่วนนี้เป็นส่วนของตารางที่บ่งบอกถึงค่าต่างๆของอุปกรณ์แต่ละชนิด โดยจะนำมาแสดงเป็นตัวอย่างดังตาราง Zener\_Diodes

ชื่อเขตข้อมูล	ชนิดข้อมูล
ID	AutoNumber
ECG No	Text
Axial Lead	Text
Voltage	Currency

รูปที่ 3.4 รายละเอียดของ ตาราง Zener\_Diodes

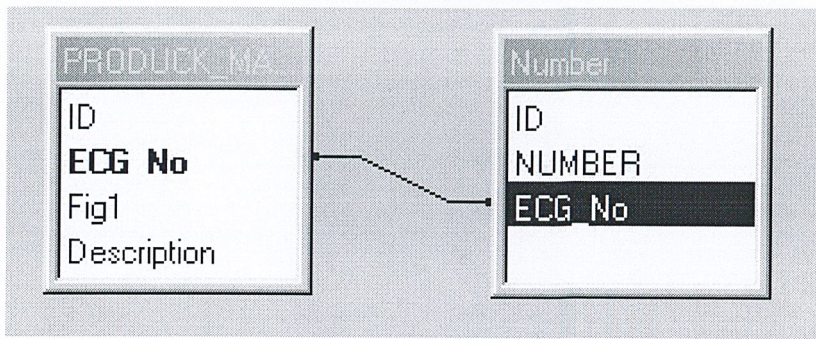
โดยกำหนดให้

ID	หมายถึง	ค่าลำดับที่ของเบอร์ที่อยู่ภายในตาราง Zener_Diodes
ECG_No	หมายถึง	ECG NUMBER ของตัวอุปกรณ์ทั้งหมด
Axial Lead	หมายถึง	ค่ากำลัง POWER ที่ตัว Zener Diodes จะทนได้
Voltage	หมายถึง	ค่าแรงดัน Zener Diodes หรือ (Vz)

ความสัมพันธ์ของฐานข้อมูลสามารถแบ่งออกเป็น 3 ส่วนใหญ่ๆ เช่นกัน

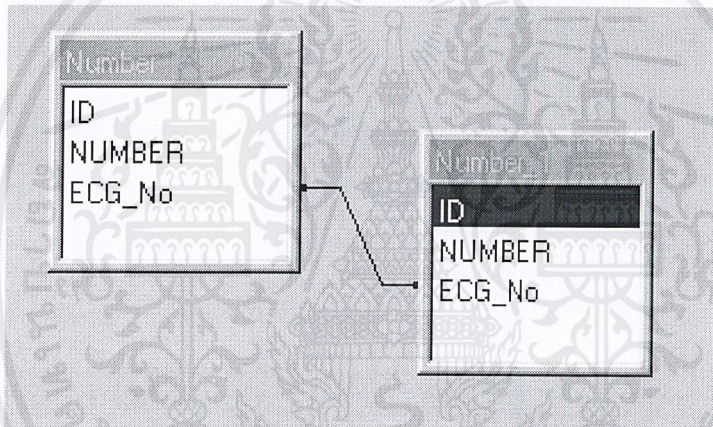
1) ส่วนแรก เป็นการแสดงส่วน ของความสัมพันธ์ระหว่าง ตาราง NUMBER และ ตาราง PRODUCT\_MAIN1 ประโยชน์เพื่อทำการค้นหาข้อมูลจากเบอร์ที่ทราบไปหาคุณสมบัติต่างๆแสดง ได้ดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



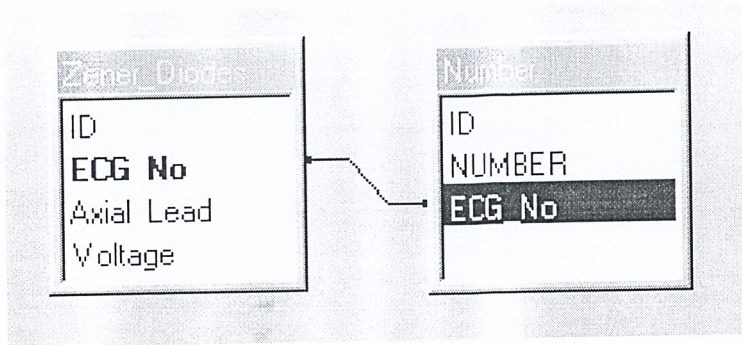
รูปที่ 3.5 ความสัมพันธ์ระหว่างตาราง NUMBER และ ตารางPRODUCK\_MAIN1

2) ส่วนที่สอง คือการแสดงความสัมพันธ์ระหว่างตาราง NUMBERและตาราง NUMBER เป็นการสร้างเพื่อทำการค้นหาเบอร์แทน โดยแสดงได้ดังรูปที่ 3.6



รูปที่ 3.6 ความสัมพันธ์ระหว่าง ตาราง NUMBER และ ตารางNUMBER

3) ส่วนที่สาม เป็นการแสดงความสัมพันธ์ในกลุ่มของการค้นหาจากคุณสมบัติไปหาเบอร์ที่ต้องการ ในที่นี้จะยกตัวอย่างความสัมพันธ์ของ ตาราง Zener\_Diodes กับตาราง NUMBER ดังรูปที่ 3.7



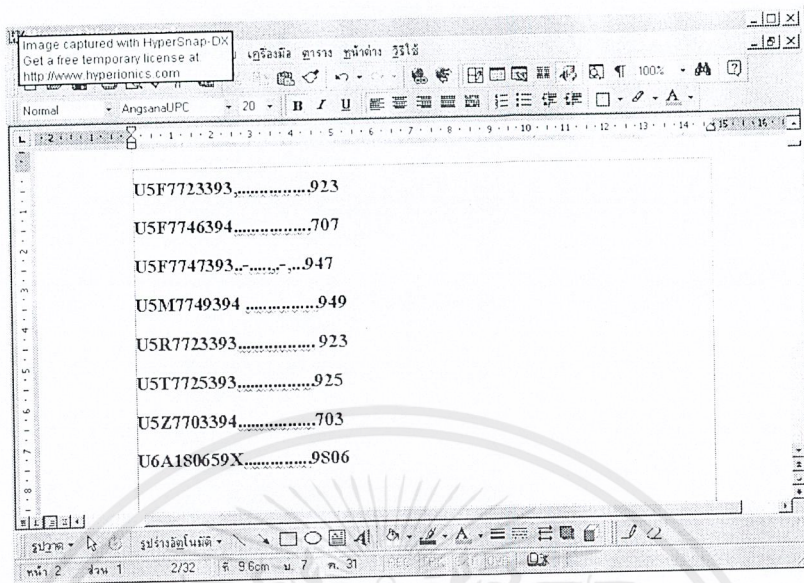
รูปที่ 3.7 ความสัมพันธ์ของ ตาราง Zener\_Diodes กับตาราง NUMBER

ในส่วนของการเก็บข้อมูลจะมีขั้นตอนดังนี้

U5F7746394.....	707
U5F7747393.....	947
U5M7749394.....	949
U5R7723393.....	923
U5T7725393.....	925
U5Z7703394.....	703
U6A180659X.....	9806
U6A740059X.....	7400
U6A909359X.....	9093
U6A909459X.....	9094
U6A909759X.....	9097
U6A909959X.....	9099
U6A910959X.....	9109
U6A911059X.....	9110
U6A911159X.....	9111
U6A911259X.....	9112
U6A913559X.....	9135
U6A915759X.....	9157
U6A915859X.....	9158
U6A939059X.....	7490
U6A993059X.....	9930

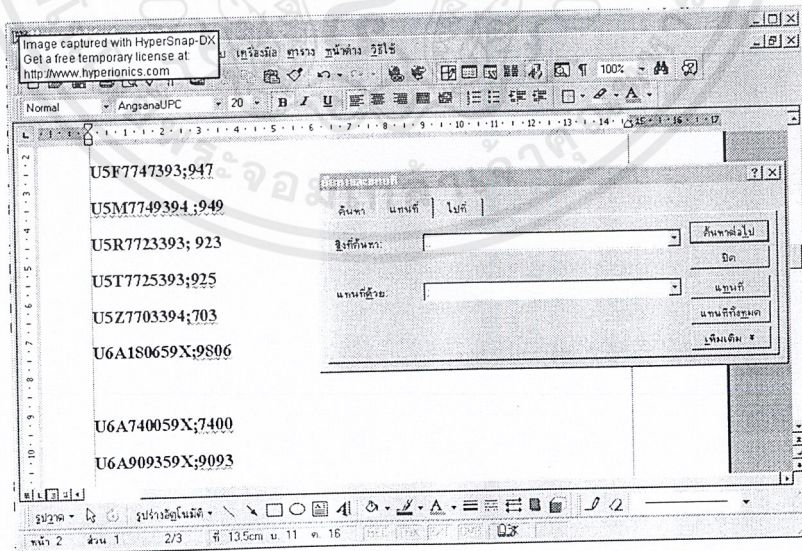
รูปที่ 3.8 การเก็บข้อมูลของเบอร์อุปกรณ์

ซึ่งเป็นการสแกนรูปโดยเป็นไฟล์ภาพจากนั้นแปลงเป็นไฟล์ Text แล้วนำเข้า โปรแกรม Notepad และนำไปเข้าโปรแกรม Microsoft Word เพื่อทำการแก้ไขเปลี่ยนแปลงจุดไขปลาทอก แล้วแทนด้วยเครื่องหมาย ; โดยทำตามขั้นตอนดังรูปที่ 3.9 และ 3.10



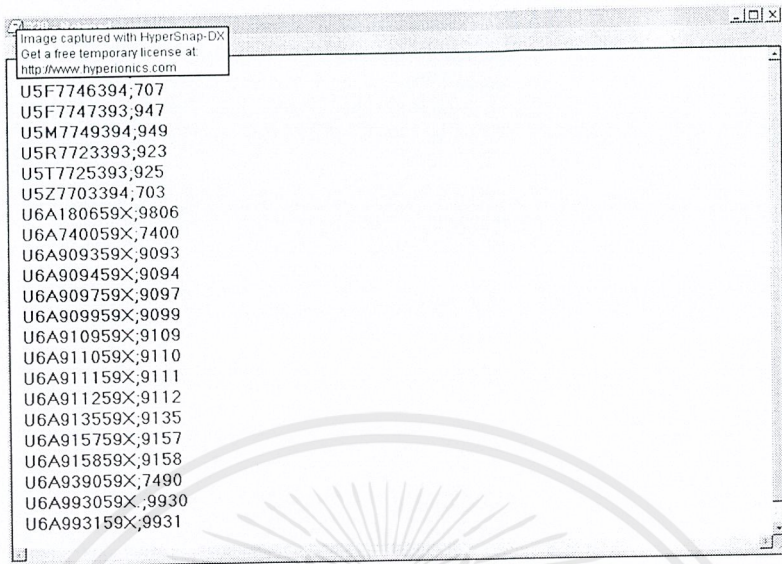
รูปที่ 3.9 การนำข้อมูลไปเข้าโปรแกรม Microsoft Word

เมื่อนำข้อมูลลงในโปรแกรม Microsoft Word แล้วดังรูปที่ 3.9 ให้ไปที่ menu bar แล้วคลิกที่แก้ไข แล้วเลือกเมนูแทนที่จากนั้นก็ทำการแก้ไขเปลี่ยนไปลาออก แล้วแทนด้วยเครื่องหมาย ; จะได้ดังรูปที่ 3.10



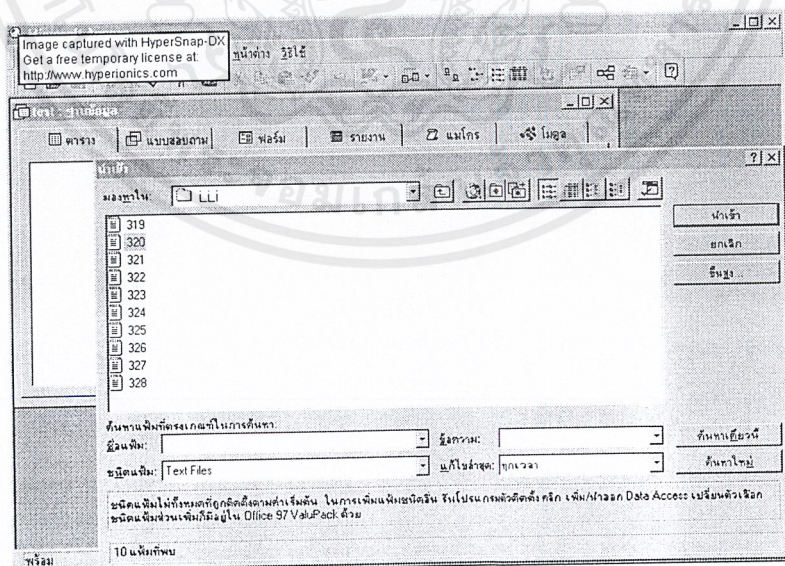
รูปที่ 3.10 การแทนที่ของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



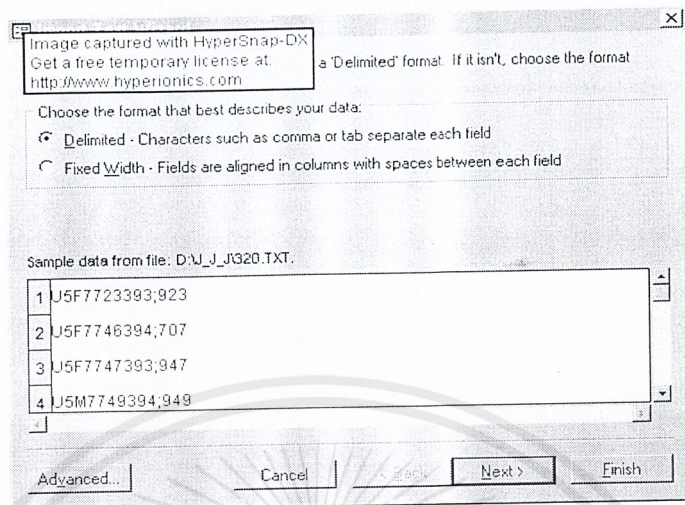
รูปที่ 3.11 ข้อมูลที่ทำการแก้ไข

เมื่อได้ข้อมูลที่ตรวจเช็ค จากโปรแกรม Microsoft Word ให้นำมา Save ที่ โปรแกรม Notepad อีกที จากนั้นให้เปิดโปรแกรม Microsoft Access เพื่อนำข้อมูลที่ทำการแก้ไขแล้วลงตาราง โดยเลือกชื่อ ไฟล์ที่ต้องการนำข้อมูลเข้า



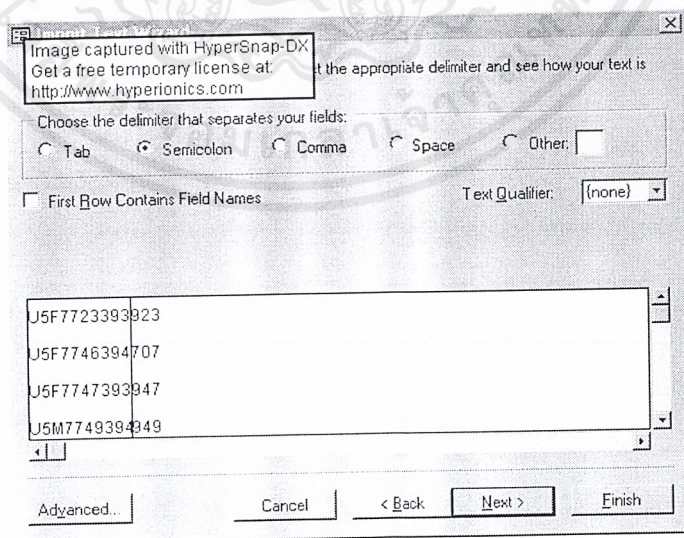
รูปที่ 3.12 การนำข้อมูลเข้าโดย Microsoft Access

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



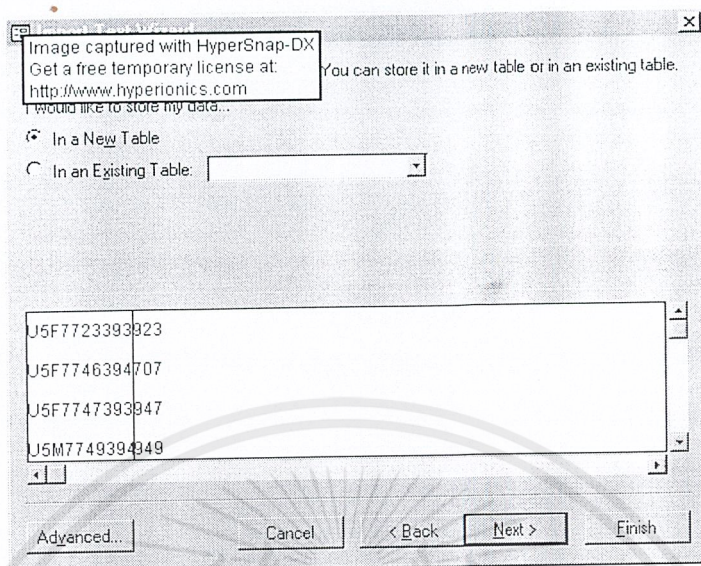
รูปที่ 3.13 การเลือกชนิดของ Access

ในการนำข้อมูลเข้าในโปรแกรม Microsoft Access การสร้างตารางข้อมูลด้วยวิธีนี้ ให้เลือก Menu แล้วคลิกที่ Import Table แล้วเลือกไฟล์ของข้อมูลที่ต้องการ ดังรูปที่ 3.12 จากนั้นเลือกที่ Menu นำเข้า คลิกเลือกที่ Delimited-Characters Such as comma of tab separate each field ตามรูปที่ 3.13 แล้วคลิกที่ปุ่ม Next หน้าจอจะปรากฏตามรูปที่ 3.14 ให้เลือกที่ semicolon แล้วก็คลิกที่ปุ่ม Next อีกครั้ง



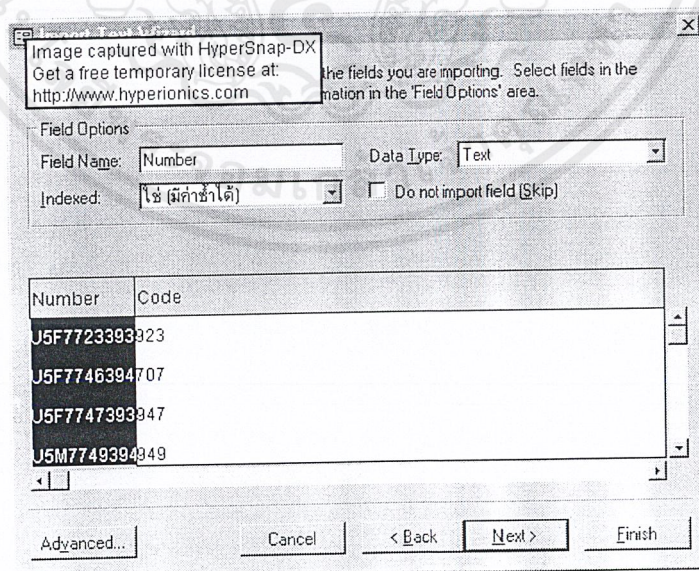
รูปที่ 3.14 การเลือกแบบ Semicolon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 การนำเข้าตาราง

เมื่อทำตามขั้นตอนดังรูป ต่อมาในรูปที่ 3.15 จะเป็นการนำเข้าของตาราง Access โดยถ้าเป็นข้อมูลชุดแรกให้เลือกที่ In a new Table แต่ถ้ามีข้อมูลอยู่แล้วให้เลือก In a Existing Table แล้วเลือกชื่อตารางที่ต้องการ

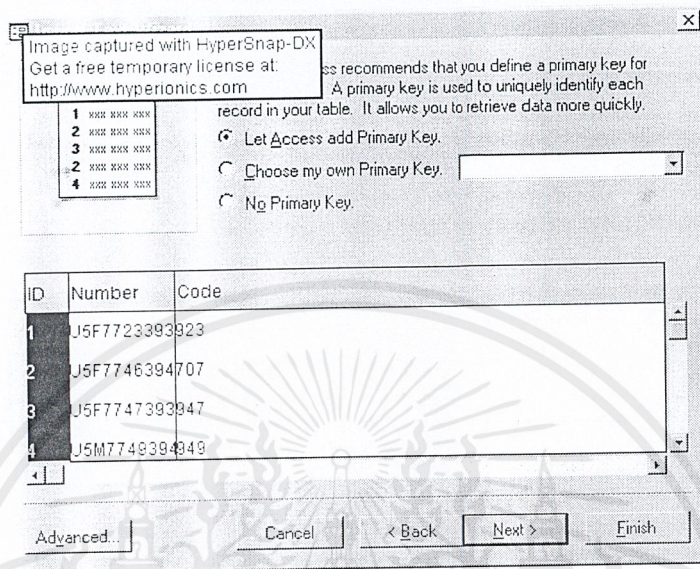


รูปที่ 3.16 การเปลี่ยนชื่อ Field Name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

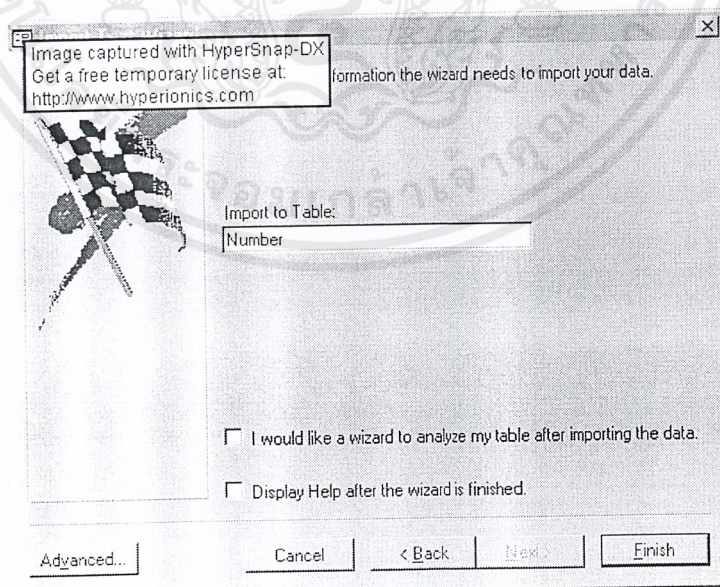
ขั้นตอนต่อไป ก็เป็นการเลือกคีย์หลัก พร้อมทั้งแสดง ID ด้วยดังรูปที่ 3.17 แล้วคลิกที่ปุ่ม

Next



รูปที่ 3.17 การเลือก ID

ขั้นตอนสุดท้าย ใส่ชื่อตารางที่กำหนดไว้แล้วคลิก Finish รูปที่ 3.18



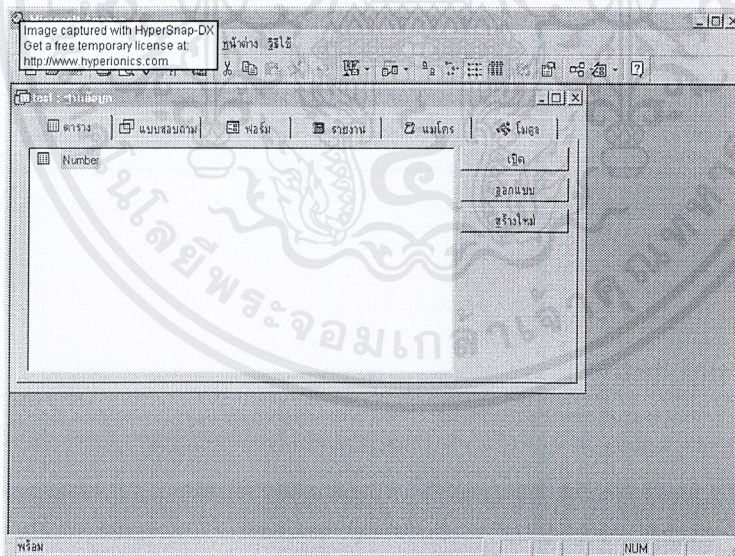
รูปที่ 3.18 การใส่ชื่อตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นจะได้ตาราง Number ที่มีข้อมูลของ ECG อยู่ ดังรูปที่ 3.19

ID	Number	Code
1	U5F7723393	923
2	U5F7746394	707
3	U5F7747393	947
4	U5M7749394	949
5	U5R7723393	923
6	U5T7725393	925
7	U5Z7703394	703
8	U6A180659X	9806
9	U6A740059X	7400
10	U6A909359X	9093
11	U6A909459X	9094
12	U6A909759X	9097
13	U6A909959X	9099

รูปที่ 3.19 ข้อมูลที่ได้จากการนำเข้า



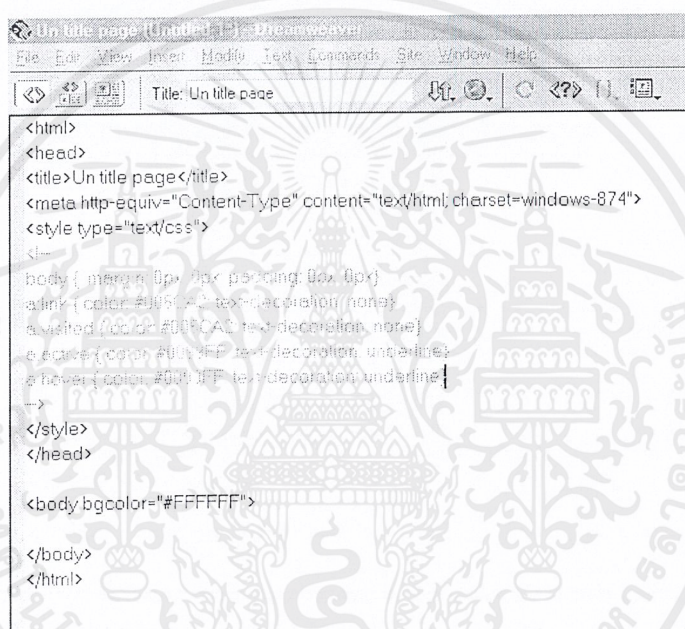
รูปที่ 3.20 ข้อมูลในตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำตามขั้นตอนดังรูป ก็จะได้ข้อมูลที่ต้องการ ซึ่งการเก็บข้อมูลแบบนี้จะช่วยลดเวลาไปได้มากและยังได้ข้อมูลที่ละจำนวนมาก ๆ อีกด้วย ซึ่งอาจจะเสียเวลาในการสแกนเพียงเล็กน้อยเท่านั้นเมื่อเทียบกับ การที่จะต้องกรอกข้อมูลที่ละตัว

### 3.2 การออกแบบโฮมเพจด้วยภาษา HTML

การออกแบบโฮมเพจเราสามารถออกแบบได้จากโปรแกรม Macromedia Dreamweaver 4



```

<html>
<head>
<title>Un title page</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874">
<style type="text/css">
<!--
body { margin: 0px; padding: 0px;
a:link { color: #0000FF; text-decoration: none;
a:visited { color: #0000FF; text-decoration: none;
a:hover { color: #0000FF; text-decoration: underline;
a:active { color: #0000FF; text-decoration: underline;
-->
</style>
</head>

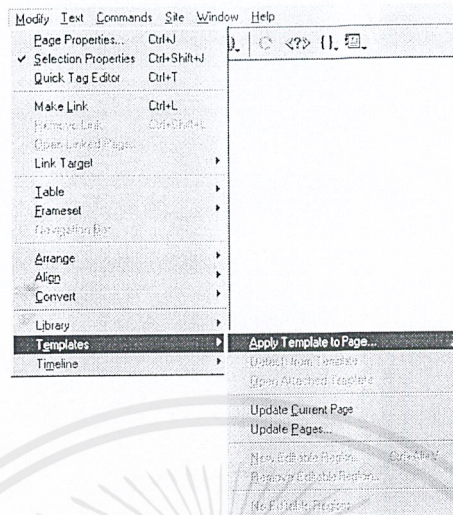
<body bgcolor="#FFFFFF">

</body>
</html>
  
```

รูปที่ 3.21 โปรแกรม Macromedia Dreamweaver 4

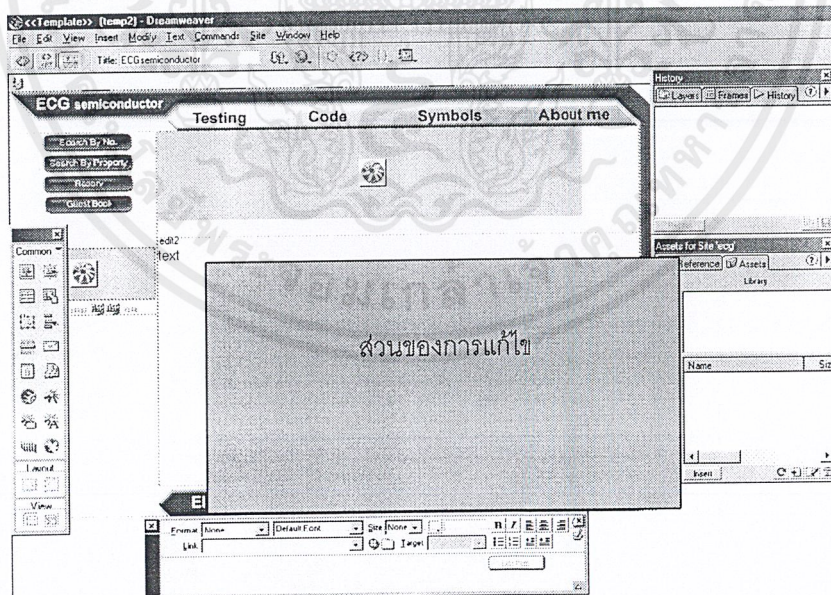
ทำการออกแบบโฮมเพจโดยใช้หลักการของ Templates คือ ออกแบบหน้าจอหลักไว้ก่อน แล้วทำการ Apply Templates to Page เพื่อสะดวกในการสร้างโฮมเพจดังรูปที่ 3.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 Apply Templates to Page

หน้าตาของ Templates สามารถออกแบบและกำหนดให้เป็น Templates โดยการ Save As Templates หน้าตาของ Templates แสดงได้ดังรูปที่ 3.23

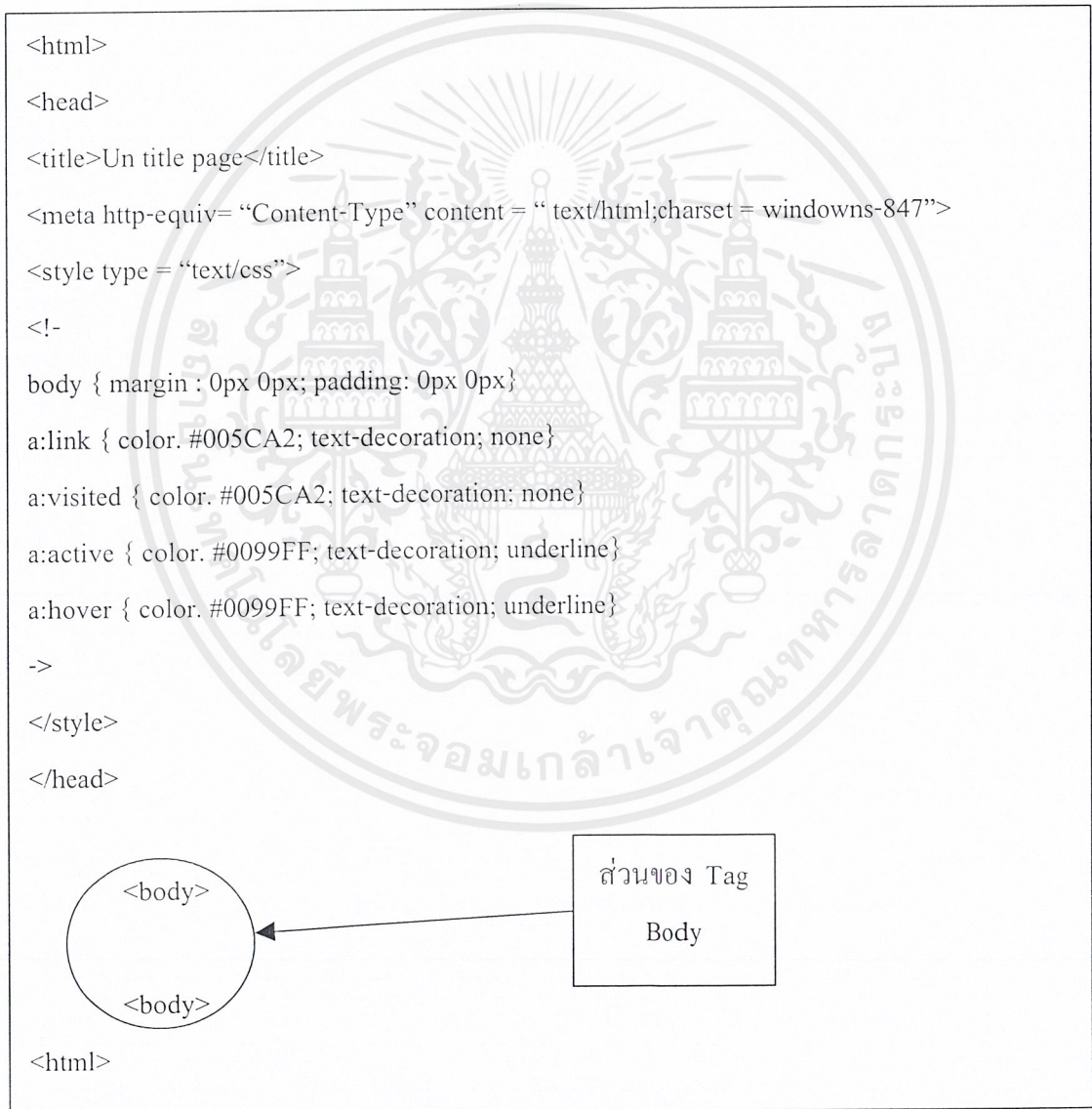


รูปที่ 3.23 หน้าตาของ Templates

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นได้ว่าจะมีส่วนการแก้ไขอยู่ใน Templates ส่วนนี้มีไว้เพื่อสะดวกในการทำหน้าตาของโฮมเพจ เพื่อเป็นมาตรฐานเดียวกันทั้งหมด ถ้าเราต้องการนำเอาข้อความหรือส่วนของ Form หรืออื่นๆแทรกก็ทำได้ง่ายโดยใช้วิธีการ Apply Templates to Page ตามที่กล่าวมาแล้ว

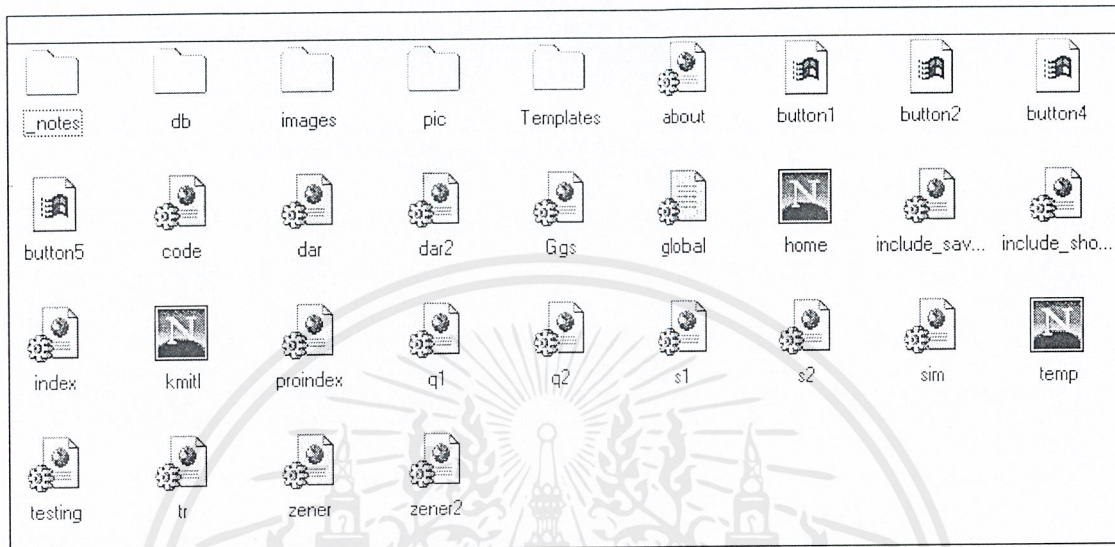
ส่วนของ Code Program โปรแกรม Macromedia Dreamweaver 4 จะทำการเขียนภาษา HTML ให้เองโดยอัตโนมัติ แต่ส่วนใดที่ทำการพัฒนาขึ้นเองก็สามารถแทรก Code Program ได้ใน Tag <Body></Body> ดังรูปที่ 3.24



รูปที่ 3.24 ส่วนของ Tag Body

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฮมเพจทั้งหมดของโครงการมีจำนวนไฟล์ทั้งหมดดังรูปที่ 3.25



รูปที่ 3.25 มีจำนวนไฟล์ทั้งหมด

จากรูปจะเห็นได้ว่าจะมี File และ Folder ทั้งหมดดังรูป จะมี File ที่เป็น .ASP , .HTML สามารถอธิบายได้คือ

.ASP ความหมาย เป็น File ที่มีส่วนของ Tag ASP ร่วมอยู่ด้วยเพื่อ File ที่เป็น ASP นั้นจะทำการ RUN ที่ Server จะมีหน้าตาดังรูป



code

.HTML ความหมาย เป็น File ที่มีส่วนของ Tag HTML ทั้งหมด จะมีหน้าตา  
ดังรูป



kmitl

Folder ต่างๆสามารถอธิบายได้ดังนี้

Db ความหมาย เป็น Folder ที่เก็บเอาส่วนของ Database เอาไว้เพื่อติดต่อกับโฮมเพจ

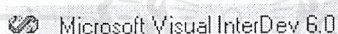
Images ความหมาย เป็น Folder ที่เก็บเอาส่วนของ รูปภาพที่เป็นส่วนของหน้าตาของเว็บเพจไว้ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pic ความหมาย เป็น Folder ที่เก็บเอาส่วนของ รูปภาพที่เป็นตัวอุปกรณ์  
 Templates ความหมาย เป็น Folder ที่เก็บเอาส่วนของ Templates เอาไว้ทั้ง  
 หหมด

### 3.3 การทำการเชื่อมต่อฐานข้อมูลกับโฮมเพจโดยใช้ภาษา ASP

ส่วนของการเชื่อมโยงฐานข้อมูลกับโฮมเพจนั้นในโครงการนี้ทำขึ้นโดยใช้ ภาษา  
 ของ ASP เขียนขึ้นมาเราสามารถพัฒนาโปรแกรมได้โดยทำการเขียน Tag ASP ในโปรแกรม  
 Microsoft Visual InterDev 6.0



#### รูปที่ 3.26 โปรแกรม Microsoft Visual InterDev 6.0

โปรแกรมนี้ช่วยในการเขียนโปรแกรมได้คือในโปรแกรมนี้จะมีเครื่องมือที่อำนวยความสะดวก  
 สะดวกอยู่มากเช่นการเขียน Code ก็จะทำการแยกสีของ Tag แต่ละ Tag ไว้สามารถดูง่ายและหา  
 ตำแหน่งที่ผิดได้ง่ายมาก และการเขียนติดต่อกับ Database ก็สามารถทำการเขียนได้ง่ายเพราะจะมี  
 เครื่องมือที่ใช้ในการติดต่อกับ ฐานข้อมูลอยู่แล้วเพียงแต่ทำการดึงเอามาใช้เท่านั้น และในส่วนของ  
 การเขียน SQL ในโปรแกรม Microsoft Visual InterDev 6.0 นั้นได้มีส่วนของ SQL เพื่ออำนวยความสะดวก  
 ในการเขียนโปรแกรมด้วย

การเขียนโปรแกรม ASP จะมีขั้นตอนต่อไปนี้จะแสดงการเขียน ในส่วนของ การค้นหา  
 ข้อมูลจาก Number จะมีด้วยกันอยู่ 5 ส่วนด้วยกันคือ

1. การเขียนส่วนของ Form เพื่อให้ผู้ใช้กรอกความต้องการ
2. การเขียนส่วนของการนำเอาค่าที่ส่งมาจาก Form มาเก็บในตัวแปร
3. การเขียนส่วนของการติดต่อกับฐานข้อมูล
4. การเขียนเพื่อกรองข้อมูล
5. การเขียนในส่วนของการแสดงผล

1) การเขียนส่วนของ Form เพื่อให้ผู้ใช้กรอกความต้องการ คือ ต้องทำการเขียน Form  
 ขึ้นมาแล้วต้องทำการเขียน `<FORM action=s1.asp id=form1 method=post name=form1>` โดย  
 ส่วน ของaction=s1.asp จำเป็นที่จะต้องบอกว่าทำการส่งค่าไปที่โฮมเพจใด

2) การเขียนส่วนของการนำเอาค่าที่ส่งมาจาก Form มาเก็บในตัวแปร การเขียนในส่วน  
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

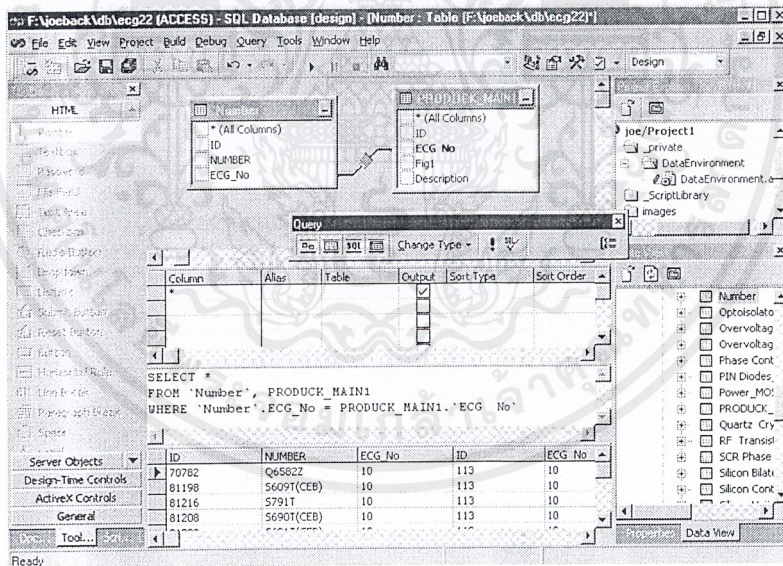
นี้สามารถเขียนได้โดย `ecg = Request ("ecg")` และ `st = Request ("st")` ก็คือการ Request ecg ค่าที่ส่งมาแล้วมาเก็บในตัวแปรที่ชื่อว่า ecg เช่นเดียวกัน

3) การเขียนส่วนของการติดต่อกับฐานข้อมูล สามารถเขียนได้หลายแบบทั้งที่ใช้ DSN และไม่ใช่ DSN แต่ในโครงการนี้ทำการเขียนติดต่อกับฐานข้อมูลโดยไม่ใช่ DSN เพื่อสะดวกในการ Up Load ขึ้นบนอินเทอร์เน็ตสามารถเขียนได้โดย

```
Set Conn=Server.CreateObject("ADODB.Connection")
conn.open "DRIVER={Microsoft Access Driver
(*.mdb)};DBQ=" & Server.mappath("db/ecg22.mdb")
```

จะเห็นได้ว่าโปรแกรมที่เขียนขึ้นมาใช้นี้ DRIVER เป็น Microsoft Access Driver และอยู่ที่ db/ecg22.mdb มีชื่อว่า ecg22.mdb

4) การเขียนเพื่อกรองข้อมูลสามารถเขียนได้โดยการใช้ภาษา SQL ในการบรรยายเราสามารถทำได้โดยทำการสร้างในโปรแกรม Microsoft Visual InterDev 6.0 ดังรูป



รูปที่ 3.27 การสร้าง SQL ในโปรแกรม Microsoft Visual InterDev 6.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) การเขียนในส่วนของการแสดงผล สามารถเขียนได้โดย

```

<%
total=rs.recordcount
if total = 0 then
%>ไม่มีค่าที่ต้องการค้นหา<%
else
pagesize=5
totalpage=total/pagesize
if (total mod pagesize) <> "0" then
totalpage= totalpage+1
end if
RS.Move pagesize*(pageno-1)
a=1
if trim(pageno) <> trim(totalpage) then
endrecord = pagesize
else
if (total mod pagesize)=0 then
endrecord=5
else
endrecord=(total mod pagesize)
end if
end if
For a=1 to endrecord
cont=cont+1
number1=rs("NUMBER")
des=rs("Description")
pic=rs("Fig1")
%>
<table border=1 cellpadding=1 cellspacing=1 width="200" align="center">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<tr>
<td width="40" align="center"><%=cont%></td>
<td &nbsp;๒๒๐๖ <%=number1%></td>
</tr>
</table>
<div align="center">
<h2><font size="4" face="MS Sans Serif">Description
<%=des%><BR>
FIG <%=pic%></font>
</h2>
</div>
<p align="center"><img height=250 src ="pic/<%=pic%>.jpg" width=350 >
<%=rs.MoveNext
next
rs.close
conn.close
%>

```

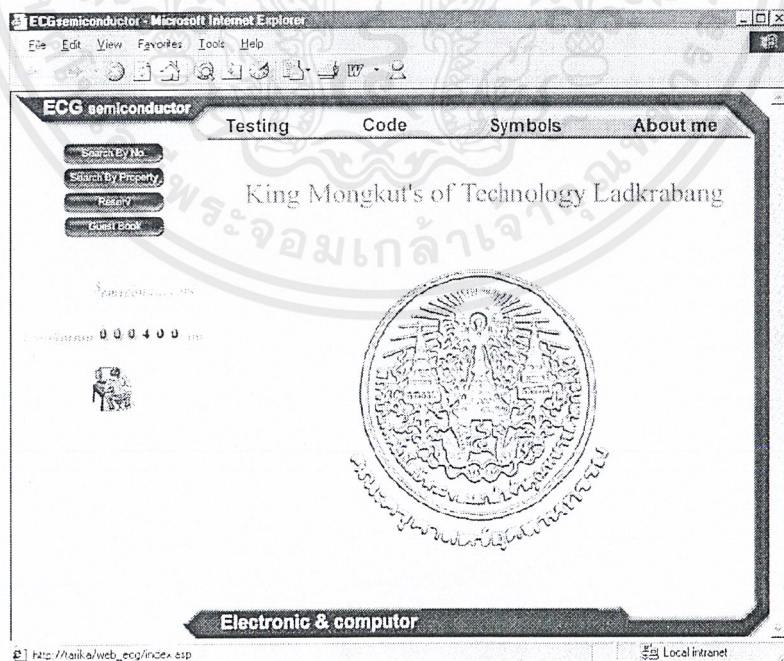
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

จากการทำงานของ Web ECG On Internet ผลที่ได้ของหน้ามีดังนี้  
เป็นส่วนของ Menu การทำงานทั้งหมดของ Web ECG On Internet ดังรูปที่ 4.1 ซึ่งมีปุ่ม  
ต่าง ๆ ดังนี้

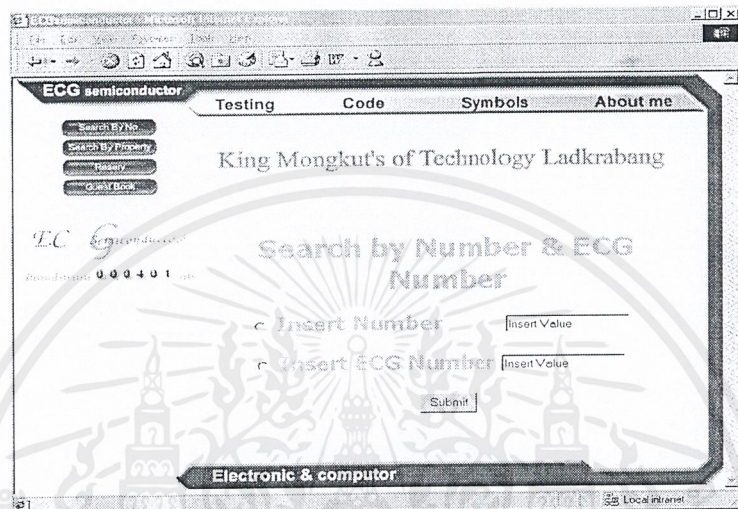
1. Search By No.
2. Search By Property
3. Replacement devices
4. Guest Book
5. Testing
6. Code
7. Symbols
8. About me



รูปที่ 4.1 เมนูการทำงาน

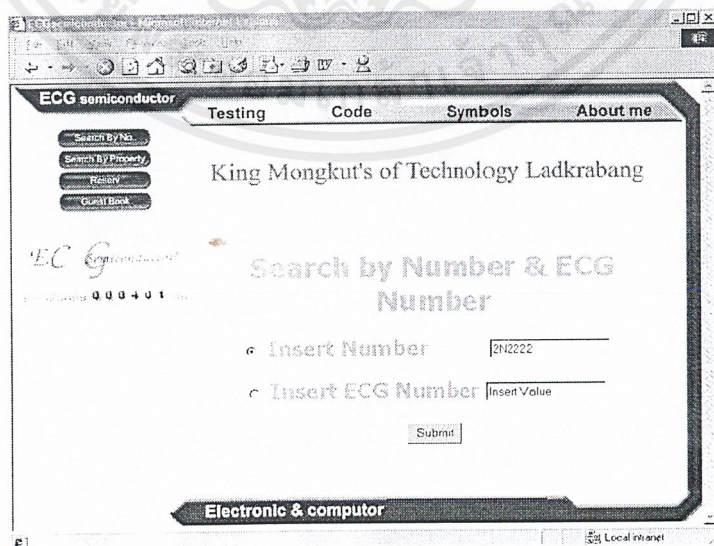
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) เมื่อคลิกที่ปุ่ม Search By No. จะปรากฏรูปแบบดังรูปที่ 4.2 โดยมี Text box ให้ป้อนข้อมูลคือ เบอร์ของอุปกรณ์ สามารถทำการ Search ได้ 2 วิธี คือ Insert Number และ Insert ECG Number



รูปที่ 4.2 การค้นหาจาก Search By No.

1.1) วิธีแรกคือ ทำได้โดยป้อนเบอร์ของอุปกรณ์ ดังรูปที่ 4.3 ลงใน Text box ของ Insert Number จากนั้นก็คลิกที่ปุ่ม Submit

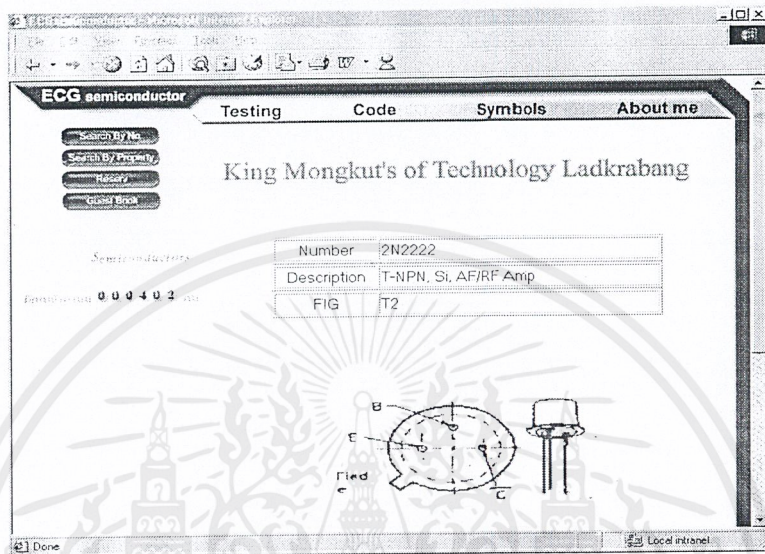


รูปที่ 4.3 การเลือก Search จาก Number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

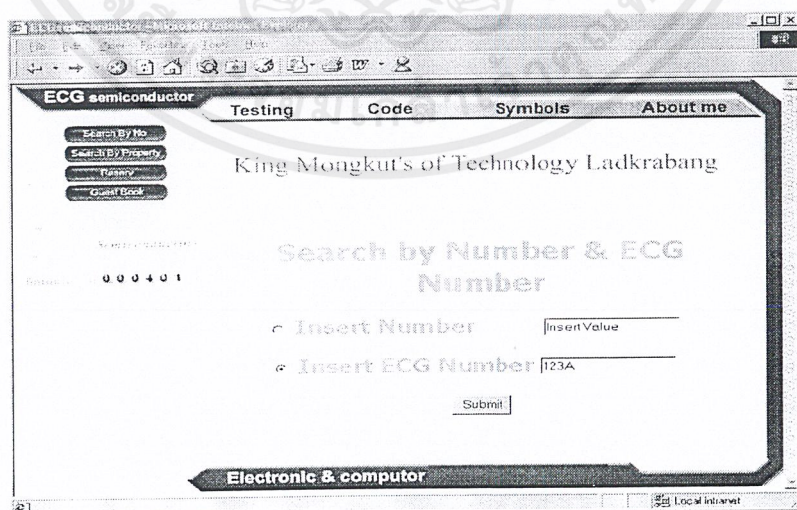
ผลที่ได้คือ แสดงเบอร์อุปกรณ์ที่ทำการค้นหา คุณสมบัติต่าง ๆ พร้อมรูปของอุปกรณ์ ดังรูป

ที่ 4.4



รูปที่ 4.4 ผลลัพธ์ของการ Search จาก Number

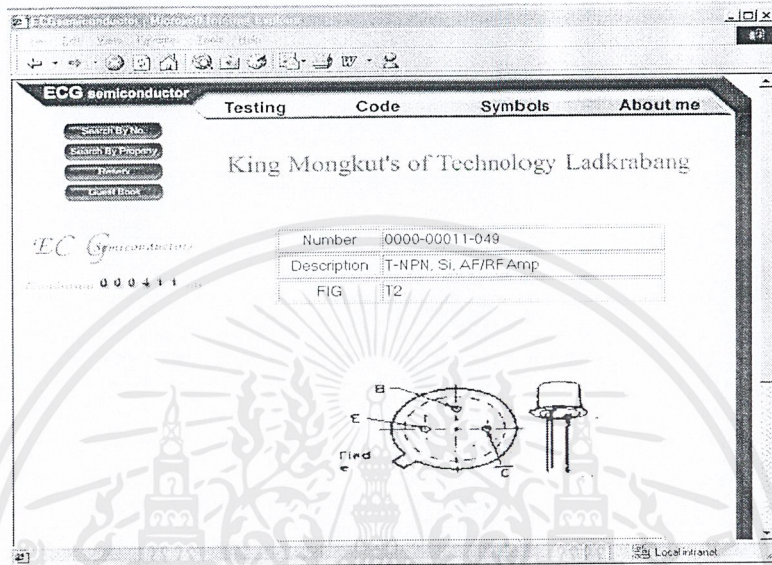
1.2) วิธีที่สองคือ Search จาก ECG Number สามารถทำได้โดย ป้อนข้อมูลของอุปกรณ์ใน Text box ของ Insert ECG Number จากนั้นก็คลิกปุ่ม Submit ดังรูปที่ 4.5



รูปที่ 4.5 การเลือก Search จาก ECG Number

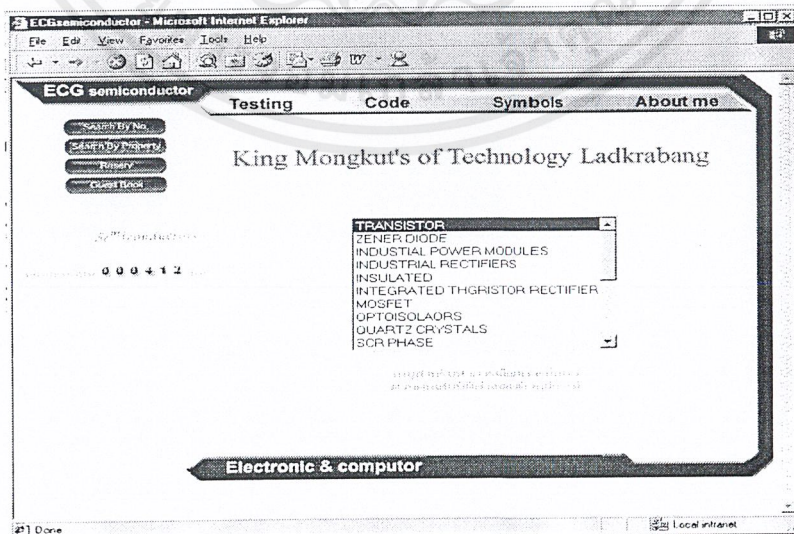
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้จะเหมือนกับวิธีแรก คือ แสดงเบอร์อุปกรณ์ที่ค้นหา พร้อมทั้งคุณสมบัติต่าง ๆ และรูปของตัวอุปกรณ์ ดังรูปที่ 4.6



รูปที่ 4.6 ผลลัพธ์ของการ Search จาก ECG Number

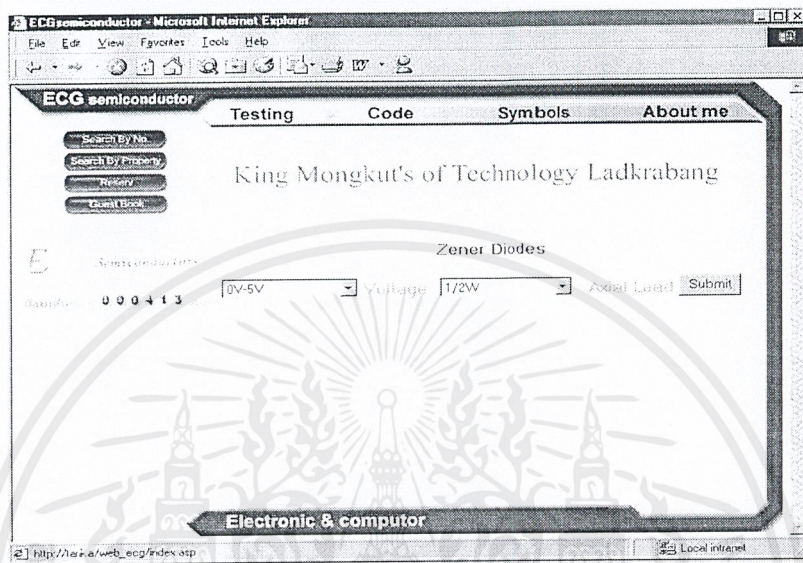
2) ปุ่ม Search By Property จะปรากฏรูปแบบดังรูปที่ 4.7 โดยจะให้เลือกรค้นชนิดของอุปกรณ์



รูปที่ 4.7 การเลือก Search จาก Property

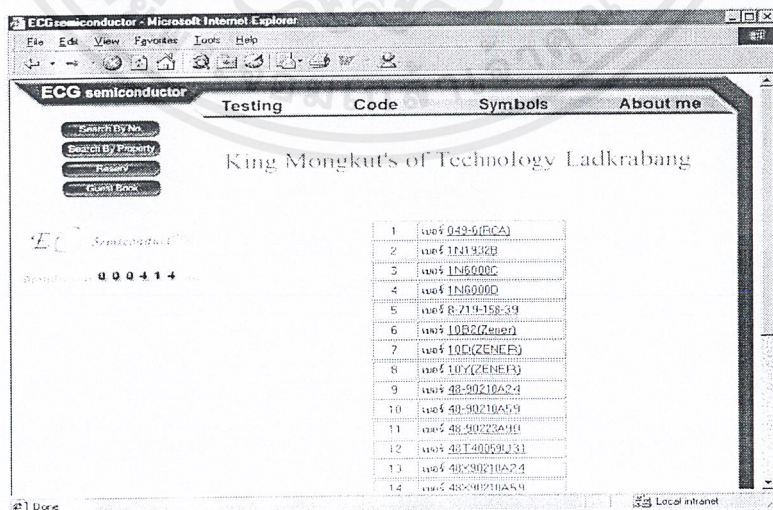
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1) เมื่อทำการเลือกชนิดเรียบร้อยแล้ว จากตัวอย่างเลือก Zener Diode โดยจะมีคุณสมบัติเกี่ยวกับ Voltage และ Axial Lead ให้เลือก ดังรูปที่ 4.8



รูปที่ 4.8 การเลือก Search จาก Zener Diodes

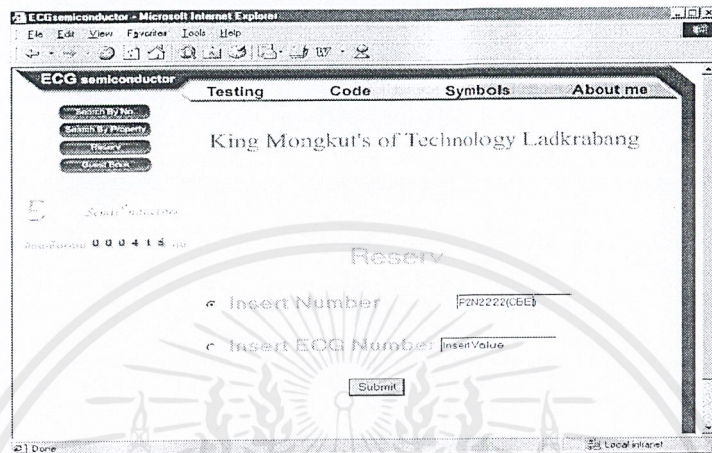
ผลที่ได้คือ แสดงเบอร์ต่าง ๆ ของอุปกรณ์ ดังรูปที่ 4.9 จากนั้นก็เลือกเบอร์ของอุปกรณ์ที่ต้องการ



รูปที่ 4.9 ผลลัพธ์ของการ Search

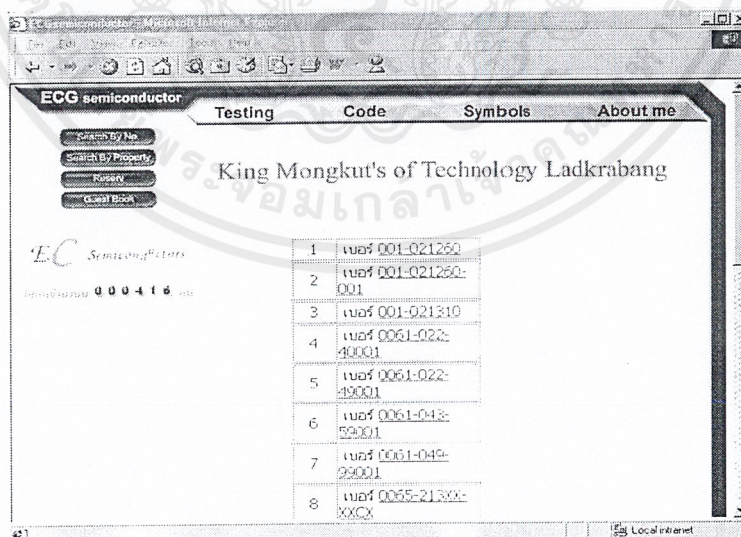
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ปุ่ม Reserve คือการหาเบอร์อุปกรณ์ทดแทน ทำได้ 2 วิธี คือ Search จาก Insert Number และ Insert ECG Number ดังรูปที่ 4.10



รูปที่ 4.10 การเลือก Search จาก Number ของ Replacement devices

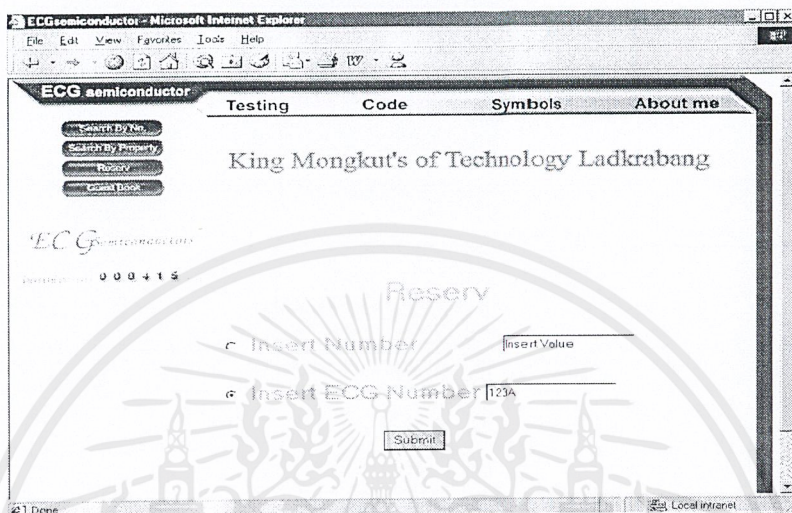
3.1) วิธีแรก ทำได้โดยป้อนเบอร์ของอุปกรณ์ (เบอร์ 2N2222) ลงใน Text box ของ Insert Number แล้วคลิกที่ปุ่ม Submit ดังรูปที่ 4.11



รูปที่ 4.11 ผลลัพธ์ของการ Search จาก Number

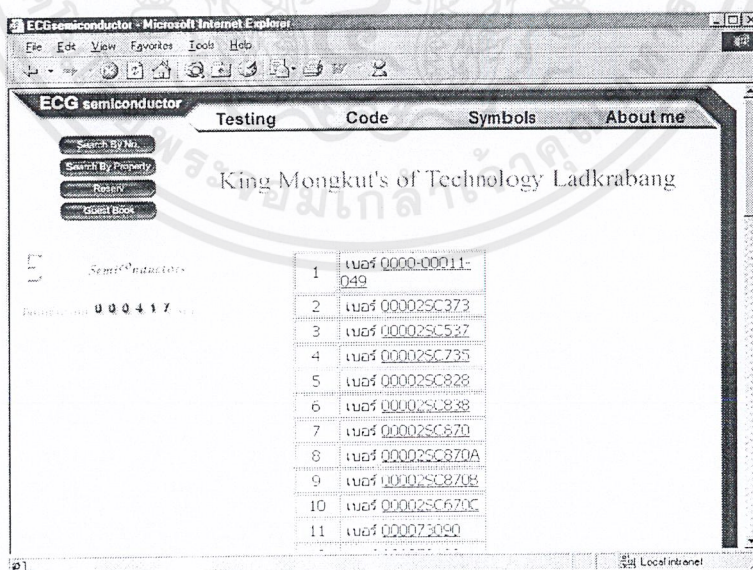
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2) วิธีที่สอง ทำได้โดยป้อนเบอร์ของอุปกรณ์ (123A) ลงใน Text box ของ Insert ECG Number แล้วคลิกที่ปุ่ม Submit ดังรูปที่ 4.12



รูปที่ 4.12 การเลือก Search จาก ECG Number ของ Replacement devices

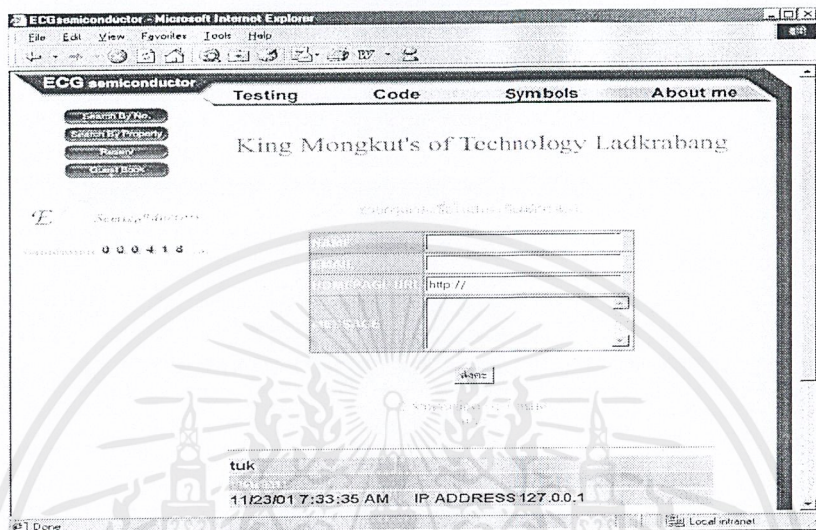
ดังนั้นผลที่ได้ คือ แสดงเบอร์ที่ใช้ทดแทนเบอร์ของอุปกรณ์ (เบอร์ 2N2222) ดังรูปที่ 4.13



รูปที่ 4.13 ผลลัพธ์ของการ Search จาก ECG Number

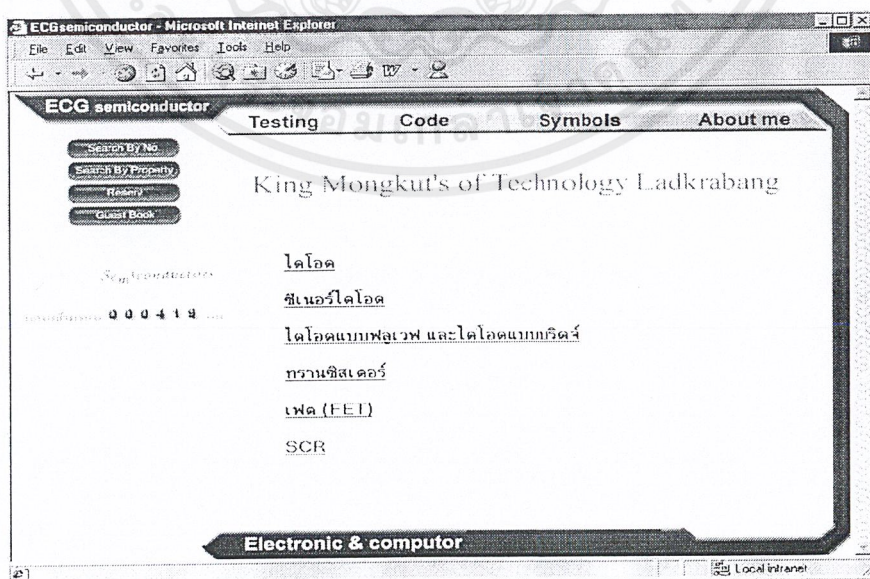
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ปุ่ม Guest Book จะมี Text box โดยใช้ป้อนข้อมูลของ NAME , EMAIL , HOME PAGE URL , และ MESSAGE ของผู้เข้ามาเยี่ยมชม ดังรูปที่ 4.14



รูปที่ 4.14 การทำงานของ Guest Book

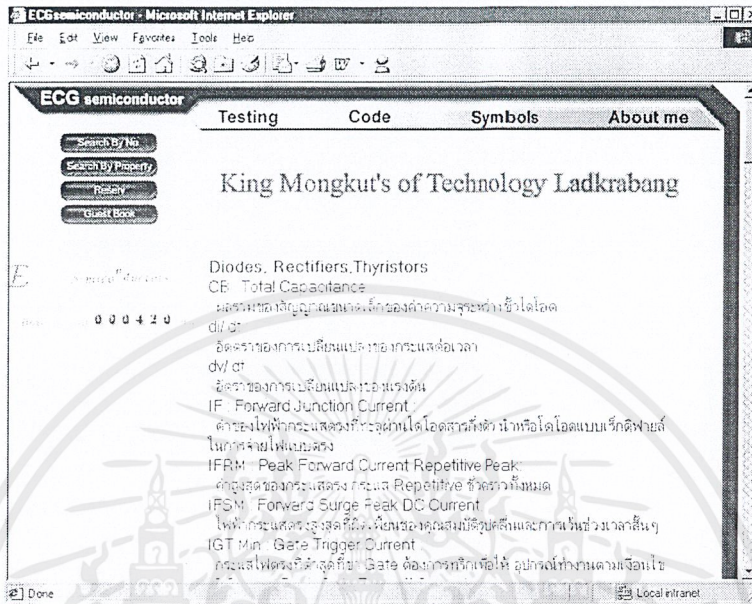
5) ปุ่ม Testing ใช้แสดงการวัดค่าต่าง ๆ ของอุปกรณ์ โดยมีวิธีการวัดและเนื้อหาต่าง ๆ เกี่ยวกับตัวของอุปกรณ์ ดังรูปที่ 4.15



รูปที่ 4.15 การทำงานของ Testing

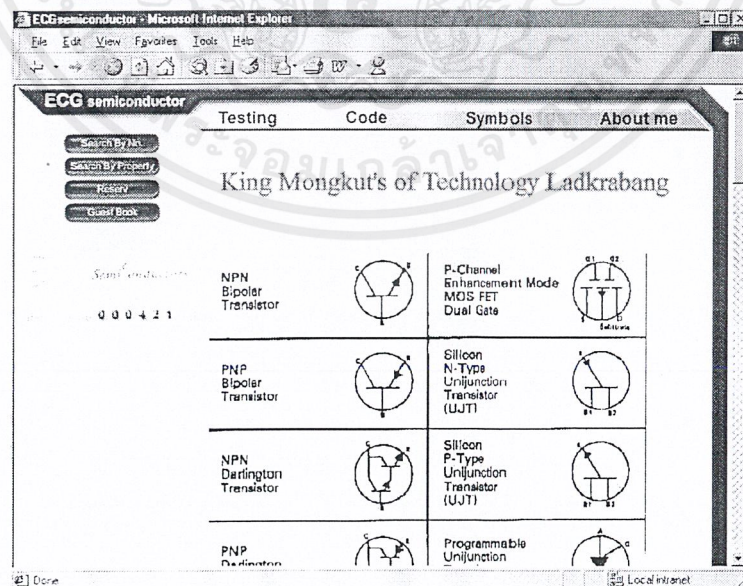
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) ปุ่ม Code ใช้แสดงความหมายของอักษรย่อต่าง ๆ ที่ใช้ในข้อมูล ดังรูปที่ 4.16



รูปที่ 4.16 การทำงานของ Code

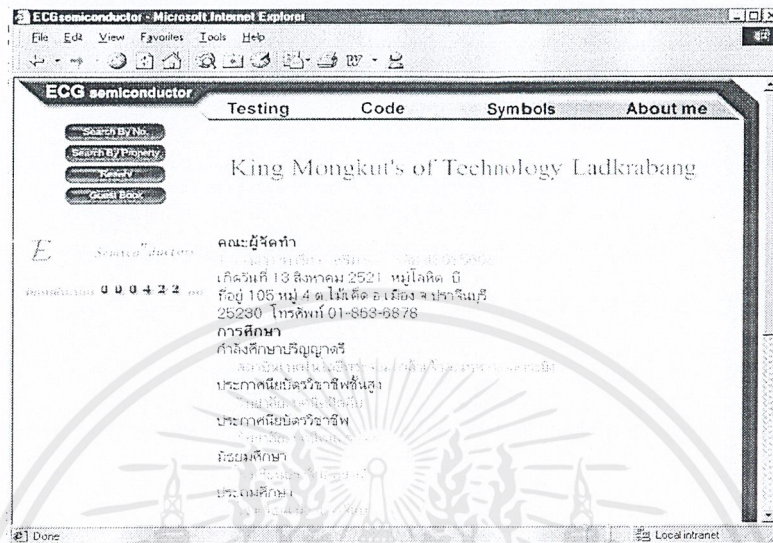
7) ปุ่ม Symbol ใช้แสดงสัญลักษณ์ต่าง ๆ ของอุปกรณ์ ดังรูปที่ 4.17



รูปที่ 4.17 การทำงานของ Symbols

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) ปุ่ม About me ใช้แสดงประวัติของคณะผู้จัดทำ ดังรูปที่ 4.18



รูปที่ 4.18 การทำงานของ About me

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป ปัญหา แนวทางแก้ไขและการพัฒนา

#### 5.1 บทสรุป

โปรแกรม ECG เป็นโปรแกรมที่สร้างขึ้นมาเพื่อประโยชน์ในการค้นหาข้อมูลอุปกรณ์อิเล็กทรอนิกส์ สะดวกในการค้นหา ในการออกแบบโปรแกรมจะออกแบบเพื่อให้ผู้ใช้ได้รับความสะดวก รวดเร็วสูงสุดในการทำงานของโปรแกรม เนื่องจากการค้นหาด้วยหนังสือมีปัญหามากมายจากการที่ผู้ใช้ไม่มีความชำนาญในการใช้หนังสือ ดังนั้นแม้ว่าผู้ใช้โปรแกรมจะไม่เคยใช้หนังสือ ECG ก็สามารถจะใช้โปรแกรมนี้เพื่อหาข้อมูลได้เช่นเดียวกับการใช้หนังสือ ECG

ส่วนในการออกแบบฐานข้อมูลออกแบบโดยให้ข้อมูลเรียงกันเป็นลำดับ เพราะข้อมูลหาความสัมพันธ์กันได้น้อยจึงไม่เหมาะกับการใช้วิธี Normalize เนื่องจากการ Normalize เหมาะกับข้อมูลที่มีความสัมพันธ์กัน

#### 5.2 ปัญหาและแนวทางแก้ไข

- 1) เนื่องจากยังไม่สามารถ Up Load เพราะ Hosting ไม่สามารถรองรับข้อมูลของเว็บที่มีมากได้ จึงยังไม่ได้ทำการ Up Load
- 2) สำหรับเบอร์ที่ทำการค้นหาได้มี 299,477 ตัว ส่วนรูปยังไม่ครบเพราะเนื่องจากยังมีรูปที่สแกนเสียและยังไม่ได้สแกนซ่อม และส่วนฐานข้อมูลยังไม่ครบเนื่องจากข้อมูลมีจำนวนมาก

#### 5.3 แนวทางการพัฒนา

การพัฒนาของโปรแกรมนี้สามารถทำได้โดย ทำการจัดตั้งชื่อเว็บ Hosting ที่มีความจุพอ และสามารถรองรับเว็บที่สร้างขึ้นได้

การพัฒนาด้านฐานข้อมูลควรตรวจสอบข้อตรวจสอบข้อมูลอย่างละเอียด

## บรรณานุกรม

- กิตติ ภักดีวัฒนกุลและจำลอง ทรูอดุตสาหะ. กัมภีร์ระบบฐานข้อมูล. พิมพ์ครั้งที่ 4 .ไทยเจริญการพิมพ์ : กรุงเทพฯ , 2544 .
- กิตติ ภักดีวัฒนกุลและจำลอง ทรูอดุตสาหะ. Visual Interdev. พิมพ์ครั้งที่ 4 .ไทยเจริญการพิมพ์ : กรุงเทพฯ , 2543 .
- พลรังสี คู่ความดีและประชา พฤกษ์ประเสริฐ . ASP เพื่อการประยุกต์ใช้งาน . ชัคเชส มีเดีย จำกัด : กรุงเทพฯ , 2544 .
- พันจันทร์ ธนวัฒน์เสถียรและคณะ . Macromedia Dreamweaver 4 . พิมพ์ครั้งที่ 2 . ชัคเชส มีเดีย จำกัด : กรุงเทพฯ , 2544 .



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อผู้ทำปริญญาโท	นางสาวทาริกา ศรีมงคล
วันเดือนปีเกิด	13 สิงหาคม 2521
สถานที่เกิด	จังหวัดปราจีนบุรี
ภูมิลำเนาเดิม	105 หมู่ 4 ตำบลไม้เค็ด อำเภอเมือง จังหวัดปราจีนบุรี 25230
ที่อยู่ปัจจุบัน	397/1 หมู่ 1 ซ.จินดาภิเษก 10 ถ.อ่อนนุช แขวงลาดกระบัง เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(01) 8636878
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนเมืองปราจีนบุรี
มัธยมศึกษาตอนต้น	โรงเรียนปราจีนกัลยาณี
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคนครนายก
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคสัทหีบ
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	อย่าทำตัวเหมือนเด็กเลี้ยงแกะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานិพนธ์	นางสาวปิยาภรณ์ นาคคำภา
วันเดือนปีเกิด	15 มีนาคม 2522
สถานที่เกิด	จังหวัดลพบุรี
ภูมิลำเนาเดิม	43 หมู่ 2 ตำบลวัดดาว อำเภอ บางปลาม้า จังหวัดสุพรรณบุรี
ที่อยู่ปัจจุบัน	72150 556-557 หมู่ 1 ซอยริมสวน ถ.อ่อนนุช แขวงลาดกระบัง เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(01)7217664
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวัดโพธิ์ตะควน
มัธยมศึกษาตอนต้น	โรงเรียนสงวนหญิง
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคสุพรรณบุรี
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	สถาบันเทคโนโลยีราชมงคลวิทยาเขต เทคนิคกรุงเทพฯ
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	กองทุนกู้ยืมเพื่อการศึกษา
คติพจน์	รู้หน้าไม่รู้ใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้