



ภาควิชาครุศาสตร์วิศวกรรม  
 คณะครุศาสตร์อุตสาหกรรม  
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 ใบรับรองปริญญาโท

ชื่อหัวข้อ อีซีจีบนอินทราเน็ต  
 ECG on Intranet

ชื่อนักศึกษา 1. นางสาวประกายเพชร อินทราวุธ รหัสประจำตัว 42035307  
 2. นางสาวปิยฉัตร สุขสุรัส รหัสประจำตัว 42035309  
 3. นางสาวสุกวรรณ ปลั่งพงษ์พันธ์ รหัสประจำตัว 42035320

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา อิเล็กทรอนิกส์และคอมพิวเตอร์

อาจารย์ที่ปรึกษา อาจารย์พงษ์เกียรติ เชษฐพิทักษ์สกุล

อาจารย์ที่ปรึกษาร่วม อาจารย์สุระชัย พิมพ์สาดี

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อาจารย์พงษ์เกียรติ เชษฐพิทักษ์สกุล	
2. อาจารย์สุระชัย พิมพ์สาดี	
3. อาจารย์สุชิน อาจหาญ	
4. อาจารย์ไพฑูรย์ พวงวงศ์ตระกูล	
5. อาจารย์กิตติพงศ์ มะโน	

วันเดือนปีที่สอบ วันเสาร์ที่ 9 ธันวาคม พ.ศ. 2543 เวลา 10.00 น.

สถานที่สอบ ห้อง ค.311 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว  
 ลงนาม.....   
 (ผศ.วิสุทธิ์ อธิพรธรรม)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 16 เดือน พฤษภาคม พ.ศ. 2543



# ปฏิญานิพนธ์

อีซีจีบนอินทราเน็ต

ECG ON INTRANET



นางสาวประกายเพชร	อินทราวุธ
นางสาวปิยฉัตร	สุขสุรัส
นางสาวสุภวรรณ	ปลั่งพงษ์พันธ์

ปฏิญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต

ปี.....

สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์

ปี.....

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

ปี.....

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

40174

b.....	11099549
i.....	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงแหล่งเอกสารทุกครั้งที่มีการนำไปใช้

# ปริญญานิพนธ์

เรื่อง อีซีจีบนอินทราเน็ต

ECG On Intranet

## วัตถุประสงค์

1. เพื่อศึกษาการใช้งานฐานข้อมูลผ่านระบบ Intranet
2. เพื่อศึกษาระบบฐานข้อมูลแบบ Client/Server
3. เพื่อออกแบบโปรแกรมใช้งาน ECG บน Intranet
4. เพื่อสร้างโปรแกรมใช้งาน ECG บน Intranet
5. เพื่อนำโปรแกรม ECG บน Intranet ไปใช้งานจริง

## ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้เกี่ยวกับระบบฐานข้อมูลแบบ Client/Server
2. ได้รับความรู้เกี่ยวกับการใช้ฐานข้อมูลผ่านระบบ Intranet
3. สามารถออกแบบโปรแกรมใช้งาน ECG บน Intranet
4. สามารถสร้างโปรแกรมใช้งาน ECG บน Intranet
5. สามารถนำโปรแกรม ECG บน Intranet ไปใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I

ชื่อหัวข้อ	อีซีจีบนอินทราเน็ต
นักศึกษา	นางสาวประกายเพชร อินทราวุธ นางสาวปิยฉัตร สุขสุรัส นางสาวสุภวรรณ ปลั่งพงษ์พันธ์
อาจารย์ที่ปรึกษา	นายพงษ์เกียรติ เชมฐพิทักษ์สกุล
อาจารย์ที่ปรึกษาร่วม	นายสุระชัย พิมป์สาตี
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	อิเล็กทรอนิกส์และคอมพิวเตอร์
ปีการศึกษา	2543

บทคัดย่อ

ปฏิญญาพันธบัตรฉบับนี้เสนอการออกแบบและการสร้างอีซีจีบนอินทราเน็ต ซึ่งเป็นทำการรวบรวมข้อมูลของอุปกรณ์อิเล็กทรอนิกส์ชนิดต่างๆ ในหนังสืออีซีจีไว้เป็นฐานข้อมูล เพื่อใช้ในการสืบค้นข้อมูล ของอุปกรณ์ในแบบต่างๆ คือ การค้นหาจากเบอร์อุปกรณ์และจากคุณสมบัติของตัวอุปกรณ์ เพื่อให้เกิดความสะดวกในการใช้งานแก่นักอิเล็กทรอนิกส์หรือผู้ใช้ทั่วไปยิ่งขึ้น

## II

<b>Thesis Title</b>	ECG On Intranet
<b>Student</b>	Miss Prakaiphet Intrawut Miss Piyachat Suksurat Miss Supawan Plungphonphong
<b>Advisor</b>	Mr. Pongkeat Shetpituksakul
<b>Co-Advisor</b>	Mr. Surachai Pimsalee
<b>Education Level</b>	Bachelor of Science in Industrial Education
<b>Program in</b>	Electronics and Computer
<b>Academic Year</b>	2000

### ABSTRACT

The thesis present the designing and implementation of an ECG on Intranet. this system is a web database of electronics devices which uses the data from ECG replacement guide book. The Web database of electronic equipment that in ECG replacement guide Book. User can easily search the data sheets of the electronics devices by the part number and properties.

## กิตติกรรมประกาศ

ปริยญาณิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากพระคุณของบิดา มารดาที่ให้การสนับสนุนในทุกๆ ด้านและเป็นกำลังใจให้เสมอมา ขอขอบคุณอาจารย์พงษ์เกียรติ เศษฐพิทักษ์สกุล, อาจารย์สุระชัย พิมพ์สาติ ที่ให้ความช่วยเหลือและให้คำปรึกษาเป็นอย่างดี อีกทั้งคณาจารย์ภาควิชาครุศาสตร์วิสุทกรรมทุกท่านที่ให้คำแนะนำ ความรู้ต่างๆ รวมถึงแนวทางการแก้ไขปัญหาในการจัดทำปริยญาณิพนธ์ ขอขอบคุณนางสาวทาริกา ศรีมงคล ที่ให้คำแนะนำจัดการกับฐานข้อมูล อีกทั้งเพื่อนๆ ทุกคนที่ให้แนวคิดต่างๆ มากมาย รับฟังปัญหาและคอยเป็นกำลังใจให้อยู่เสมอ คณะผู้จัดทำจึงขอขอบคุณไว้ ณ โอกาสนี้ด้วย



## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มาของปริญญาานิพนธ์	2
1.2 ซัดความสามารถของโครงการ	2
1.3 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 บทนำ	4
2.2 ประวัติเป็นมาของการจัดการฐานข้อมูล	4
2.3 ความรู้พื้นฐานเกี่ยวกับฐานข้อมูล	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	11
2.5 สถาปัตยกรรมของระบบฐานข้อมูล	13
2.5.1 ความเป็นอิสระของข้อมูล	14
2.5.2 Instance และ Database Schema	14
2.5.3 ระบบการจัดการฐานข้อมูล	14
2.5.4 ผู้บริหารฐานข้อมูล	15
2.5.5 ผู้ใช้ระบบ	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.5.6 การจัดเก็บข้อมูล	16
2.5.7 ภาษาสำหรับนิยามข้อมูล	19
2.5.8 ภาษาสำหรับการใช้ข้อมูล	19
2.6 ฟังก์ชันการขึ้นต่อกัน, คีย์และนอร์มัลไลซ์	19
2.6.1 กฎของความคงสภาพ (Integrity Rule)	20
2.6.2 ฟังก์ชันการขึ้นต่อกัน	21
2.6.3 คีย์	21
2.6.4 รูปแบบนอร์มัลระดับที่ 1, 2 และ 3	21
2.7 การออกแบบฐานข้อมูล	24
2.7.1 เป้าหมายของการออกแบบฐานข้อมูล	25
2.7.2 ความต้องการของผู้ใช้แต่ละคน	26
2.7.3 หลักการพื้นฐานของการออกแบบฐานข้อมูล	27
2.8 ภาษา SQL	28
2.9 ภาษาสคริปต์ PHP (Professional Home Page)	32
2.9.1 การติดต่อฐานข้อมูลโดยอาศัย ODBC driver	33
2.9.2 การเปิดการติดต่อกับฐานข้อมูล	34
บทที่ 3 การออกแบบการสร้าง และการทำงาน	36
3.1 การออกแบบและการสร้างฐานข้อมูล	36
3.1.1 ระบบฐานข้อมูล	36
3.1.2 การจัดการระบบฐานข้อมูล	37
3.1.3 การสร้างฐานข้อมูล	37
3.1.4 การออกแบบฐานข้อมูล	41
3.1.5 การสร้างตาราง	42
3.1.6 การออกแบบโครงสร้างของตาราง	42
3.1.7 ข้อมูล ECG	43
3.1.8 ข้อมูลอุปกรณ์	43
3.1.9 การสร้างตารางใน SQL	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
3.1.10 การใช้ SQL Server Enterprise Manager สร้างตาราง	43
3.1.11 การกำหนดค่า DSN	45
3.2 การออกแบบและการสร้างโฮมเพจ	48
3.2.1 การสร้างโฮมเพจ โดย Macromedia Dreamweaver3	49
3.3 การเชื่อมต่อฐานข้อมูลด้วย PHP	52
บทที่ 4 การทดลองและผลการทดลอง	53
บทที่ 5 บทสรุป ปัญหาแนวทางแก้ไขและการพัฒนา	62
5.1 บทสรุป	62
5.2 ปัญหาและแนวทางแก้ไข	62
5.3 แนวทางการพัฒนา	63
ภาคผนวก ผังการทำงานและโปรแกรม	64
บรรณานุกรม	90
ประวัติผู้แต่ง	91

## สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 การสร้างความสัมพันธ์แบบหนึ่งต่อหนึ่งโดยใช้ตารางเดียว	16
ตารางที่ 2.2 การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม	17
ตารางที่ 2.3 การสร้างความสัมพันธ์แบบกลุ่มต่อกลุ่ม	18
ตารางที่ 2.4 ตัวอย่างรีเลชัน 1 NF	22
ตารางที่ 2.5 ตัวอย่างรีเลชัน 2 NF	23
ตารางที่ 2.6 สรุปคำสั่งของ SQL	29



## สารบัญรูป

รูป	หน้า
รูปที่ 2.1 ตัวอย่าง Entity และ Attribute	7
รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรคู่แสดงความเป็นกลุ่ม	7
รูปที่ 2.3 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรเดียวแสดงความเป็นกลุ่ม	8
รูปที่ 2.4 การจัดระบบไฟล์	9
รูปที่ 2.5 การใช้ระบบฐานข้อมูล	9
รูปที่ 2.6 การใช้ระบบฐานข้อมูลด้วย DBMS	10
รูปที่ 2.7 โครงสร้างฐานข้อมูลแบบเน็ตเวิร์ค	11
รูปที่ 2.8 ระดับของข้อมูล 3 ระดับ	13
รูปที่ 2.9 ขั้นตอนการออกแบบฐานข้อมูล	26
รูปที่ 2.10 โครงสร้างการทำงานของ ODBC	34
รูปที่ 3.1 โครงสร้างระบบฐานข้อมูลของ SQL Server 7.0	37
รูปที่ 3.2 การสร้างฐานข้อมูลโดยคลิกขวามบนพื้นที่ว่าง	38
รูปที่ 3.3 การสร้างฐานข้อมูลด้วยเมนู	38
รูปที่ 3.4 การเปลี่ยนไดเรกทอรี	39
รูปที่ 3.5 การเลือกที่เก็บ Transaction Log	40
รูปที่ 3.6 สัญลักษณ์ของฐานข้อมูลด้วยรูปทรงกระบอกสี่เหลี่ยม	41
รูปที่ 3.7 ความสัมพันธ์ระหว่างข้อมูลทั้ง 3 ส่วน	42
รูปที่ 3.8 การเข้าสู่ SQL Server Enterprise Manager	44
รูปที่ 3.9 หน้าต่างของ Design Table	44
รูปที่ 3.10 ชื่อตาราง Main ใน project Table	45
รูปที่ 3.11 หน้าต่าง ODBC Administrator ใน Windows98	46
รูปที่ 3.12 หน้าต่าง Create New Data Source ของ ODBC	47
รูปที่ 3.13 หน้าจอหลัก	48
รูปที่ 3.14 การเลือก Object	49
รูปที่ 3.15 การเลือกรูปแบบของเฟรม	50
รูปที่ 3.16 แบบฟอร์มต่างๆ ในเมนูฟอร์ม	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 3.17 ผลการออกแบบและกำหนดฟอร์มเรียบร้อยแล้ว	51
รูปที่ 4.1 เมนูการทำงานของโปรแกรม ECG on Intranet	53
รูปที่ 4.2 เมนู SEARCH (ค้นหา)	54
รูปที่ 4.3 การค้นหาแบบป้อนหมายเลขอุปกรณ์	55
รูปที่ 4.4 ผลการทดลองการป้อนหมายเลขอุปกรณ์	56
รูปที่ 4.5 หน้าจอการค้นหาโดยใช้คุณสมบัติเลือกจากไดอะล็อกบ็อกซ์	56
รูปที่ 4.6 ผลการทดลองการค้นหาด้วยคุณสมบัติของอุปกรณ์	58
รูปที่ 4.7 การทำงานของเมนู LINK	59
รูปที่ 4.8 การทำงานของเว็บบอร์ด	59
รูปที่ 4.9 การทำงานของเว็บบอร์ดหลังได้รับข้อความ	60
รูปที่ 4.10 การทำงานของ GUEST BOOK	60
รูปที่ 4.11 การทำงานของเมนู ABOUT ME	61

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของปริญาานิพนธ์

ในวงการอิเล็กทรอนิกส์นั้นเป็นที่รู้กันคืออยู่แล้วว่ามีความเจริญก้าวหน้าเป็นอันมาก โดยได้มีการใช้เทคโนโลยีที่ทันสมัยในการผลิตอุปกรณ์อิเล็กทรอนิกส์ชนิดต่างๆ มากมาย นับล้านชนิด เพื่อความสะดวกสบายในการสร้างวงจรและการออกแบบอุปกรณ์ต่างๆ จนในปัจจุบันได้มีอุปกรณ์ทางด้านอิเล็กทรอนิกส์มากมายหลายชนิด โดยแต่ละชนิดนั้นจะมีคุณสมบัติและคุณภาพแตกต่างกันไป ปัจจุบันได้มีการรวบรวมคุณสมบัติของอุปกรณ์อิเล็กทรอนิกส์ไว้ในหนังสือที่เป็นที่รู้จักกันดีคือ อีซีจี (Electronic Component Replacement Guide) หากเรามีความต้องการที่จะค้นหาคุณสมบัติต่างๆ เกี่ยวกับอุปกรณ์อิเล็กทรอนิกส์ เรา จำเป็นต้องทราบวิธีการเปิดหนังสือ นับว่าไม่ใช่เรื่องง่ายนักที่จะค้นหาคุณสมบัติเหล่านั้นของตัวอุปกรณ์ เราจำเป็นต้องมีความรู้ในเรื่องของอุปกรณ์แต่ละตัวพอสมควร จึงจะสามารถค้นหาตัวอุปกรณ์ชนิดนั้นๆ ได้ นอกจากนั้นยังพบว่าเกิดปัญหาในเรื่องของการหาซื้ออุปกรณ์ หากอุปกรณ์ที่เราต้องการไม่สามารถหาซื้อได้ตามท้องตลาด จึงจะต้องมีการกลับไปเปิดค้นหาอุปกรณ์ที่สามารถนำมาใช้แทนใหม่อีกครั้งหนึ่ง

จากการที่ได้ศึกษาคุณสมบัติของระบบฐานข้อมูลถึงการจัดเก็บรวบรวมข้อมูลต่างๆ อย่างเป็นระบบ ตลอดจนการค้นหาที่เก็บในระบบได้อย่างสะดวกและรวดเร็ว การประหยัดเนื้อที่ในการจัดเก็บข้อมูล และยังสามารถแก้ไขหรือเพิ่มเติมข้อมูลใหม่ๆ ได้ จึงได้มีการจัดสร้างระบบฐานข้อมูลที่ใช้จัดเก็บข้อมูลของอุปกรณ์ทางด้านอิเล็กทรอนิกส์ขึ้น เพื่อนำมาใช้ภายในหน่วยงาน ที่มีการใช้งานเกี่ยวกับอุปกรณ์อิเล็กทรอนิกส์เป็นประจำ เพื่อให้บุคลากรในหน่วยงานนั้นๆ ได้มีความสะดวกรวดเร็วในการทำงานและพัฒนาหน่วยงานต่อไป

## 1.2 ขีดความสามารถของโครงการ

โครงการนี้มีขีดความสามารถดังต่อไปนี้

1. รวบรวมคุณสมบัติต่างๆ ของอุปกรณ์อิเล็กทรอนิกส์ที่มีในคู่มือ ECG
2. ค้นหาข้อมูลตามเบอร์ของอุปกรณ์ได้
3. ค้นหาข้อมูลตามชนิดของอุปกรณ์ได้
4. สามารถใช้ฐานข้อมูลได้โดยผ่านอินเทอร์เน็ต
5. สามารถหาอุปกรณ์ทดแทนได้

## 1.3 เนื้อหาสังเขป

เนื้อหาภายในปฏิญานิพนธ์ฉบับนี้ ได้จัดแบ่งออกเป็นบทต่างๆ ทั้งหมดด้วยกัน 5 บท เพื่อความสะดวกในการศึกษา และทำความเข้าใจ โดยในแต่ละบทนั้นจะประกอบไปด้วยเนื้อหาที่สำคัญๆ ดังนี้

บทที่ 1 บทนำ ในส่วนของบทนี้จะกล่าวถึงความสำคัญและที่มาของปฏิญานิพนธ์ฉบับนี้ รวมไปถึงความจำเป็นในการจัดการฐานข้อมูล ตลอดจนลักษณะโดยทั่วไป ของปฏิญานิพนธ์อย่างกว้างๆ เพื่อให้ผู้ที่ศึกษา ได้ทราบถึงลักษณะทั่วไป และส่วนประกอบของเนื้อหาสาระที่สำคัญภายในปฏิญานิพนธ์นี้

บทที่ 2 ทฤษฎี และหลักการ ประกอบด้วยเนื้อหา ดังนี้ คือ ความรู้พื้นฐานทั่วไปเกี่ยวกับระบบฐานข้อมูล ประโยชน์จากการประมวลผลด้วยฐานข้อมูล การออกแบบฐานข้อมูล ภาษา SQL การเขียนโฮมเพจจาก PHP

บทที่ 3 การออกแบบ การสร้าง และการทำงาน กล่าวถึงเนื้อหาที่เกี่ยวกับ หลักการออกแบบ การสร้างและการทำงาน ลำดับขั้นตอนของการออกแบบ การกำหนดสิทธิ์ใช้งานฐานข้อมูล รวมไปถึงขั้นตอนการสร้าง โฮมเพจและฐานข้อมูลในส่วนต่างๆ

บทที่ 4 การทดลองและผลการทดลอง ประกอบด้วย การทดลองและผลการทดลองการค้นหาข้อมูล การใช้งาน Webboard และ GuestBook

บทที่ 5 บทสรุป ปัญหา แนวทางแก้ไข และพัฒนา ขั้นตอนการสรุปผล ในการจัดทำโครงการ ปัญหาที่เกิดขึ้นในระหว่างการทำโครงการ และได้เสนอแนวทางในการแก้ปัญหา รวมทั้งแนวทางการพัฒนาให้มีประสิทธิภาพมากยิ่งขึ้น

ภาคผนวก ก ผังการทำงานและโปรแกรม

สำหรับส่วนที่นอกเหนือไปจากเนื้อหาในบทที่ 1 ถึงบทที่ 5 ผู้อ่านสามารถศึกษาค้นคว้า  
ข้อมูลเพิ่มเติมได้จากหนังสือ และเอกสารอ้างอิงที่แสดงเอาไว้ในบรรณานุกรม หรืออาจจะศึกษา  
เพิ่มเติมจากหนังสือเล่มอื่นๆ ที่เกี่ยวข้อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 บทนำ

ในการบริหารงานด้านต่างๆ ไม่ว่าจะเป็นงานด้านธุรกิจหรืองานด้านการศึกษา มีความจำเป็นอย่างยิ่งที่จะต้องเกี่ยวข้องกับงานข้อมูล การจัดเก็บข้อมูล ในรูปแบบต่างๆ ไม่ว่าจะเป็น เอกสารหรือบันทึก ข้อมูลบางอย่างมีแบบฟอร์มเป็นมาตรฐาน ซึ่งอาจจัดเก็บไว้ในรูปแบบที่สามารถค้นหาได้ง่าย การจัดเก็บโดยวิธีปกติก็น่าจะสมารถใช้ได้ในงานที่มีขนาดเล็กเท่านั้น แต่สำหรับระบบงานขนาดใหญ่ที่มีข้อมูลที่จะจัดเก็บ และมีความซับซ้อนของงานหลายๆ การนำเครื่องคอมพิวเตอร์มาใช้จะช่วยอำนวยความสะดวกได้เป็นอย่างดี และช่วยให้การบริหารงานระบบข้อมูลเป็นไปอย่างมีประสิทธิภาพมากยิ่งขึ้น

#### 2.2 ประวัติความเป็นมาของการจัดการฐานข้อมูล

ความจริงแล้วนับเป็นเรื่องยากทีเดียวที่จะกล่าวลงไปอย่างแน่ชัดว่า ระบบฐานข้อมูลได้ถือกำเนิดขึ้นมาเมื่อใด แต่มีเหตุผลที่น่าเชื่อถือได้ว่าต้นกำเนิดของระบบฐานข้อมูลเกิดขึ้นจากโครงการอพอลโลของสหรัฐอเมริกา อันเป็นโครงการส่งมนุษย์อวกาศไปดวงจันทร์ในช่วงเวลา 20 – 30 ปีที่แล้ว แน่หนอนว่าข้อมูลที่ใช้ในงานดังกล่าว จะต้องมิจำนวนมากทีเดียว การจัดระบบข้อมูลในโครงการนี้เกิดจากการว่าจ้างบริษัทไอบีเอ็ม ให้พัฒนาระบบการดูแลข้อมูลขึ้นมาอันได้แก่ ระบบ GUAM (Generalized Update Access Method) ซึ่งเราต้องถือเป็นต้นกำเนิดของ ระบบการจัดการฐานข้อมูล บริษัทไอบีเอ็มพบว่าการทำงานของระบบ GUAM ไม่เพียงแต่จะมีประโยชน์เฉพาะกับโครงการอวกาศเท่านั้น แต่ยังมีประโยชน์ในการประมวลผลในงานธุรกิจอื่นๆ ดังนั้นในอีก 2 ปีต่อมา ไอบีเอ็มได้พัฒนาการจัดการข้อมูลขึ้นมาใหม่เพื่อการใช้งานในวงการธุรกิจต่างๆ ไป อันได้แก่ระบบ DL/I (Data Language/I) จากนั้นไอบีเอ็มก็ได้เสริมสร้าง DL/I เพิ่มเติมขึ้น และในที่สุดก็ได้ระบบ IMS (Information Management System) ซึ่งใช้กันมาจนถึงปัจจุบัน

นอกจากบริษัทไอบีเอ็มแล้วยังมีบริษัท GE (General Electric) ที่มีส่วนในการพัฒนาระบบฐานข้อมูล ได้แก่ระบบ IDS (Integrated Data Store) ระบบนี้ได้เริ่มใช้ในช่วงปี พ.ศ. 2509 และเป็นต้นกำเนิดของระบบโคดาซิล (CODASYL) หรือโมเดลแบบเน็ตเวิร์คที่ยังนิยมใช้กันแพร่หลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในปัจจุบัน ระบบโคดาชีวล์เกิดขึ้นตั้งแต่ พ.ศ. 2517 โดยคณะกรรมการชุดหนึ่งอันเป็นทีมเดียวกับชุดที่พัฒนาภาษาโคบอลและได้ร่างข้อกำหนดมาตรฐานสำหรับโคดาชีวล์ขึ้นสำเร็จเมื่อปี พ.ศ. 2515

ในช่วงเวลาไล่เลี่ยกับที่โคดาชีวล์ฉบับมาตรฐาน ได้ถูกบัญญัติขึ้น ดร.คอดด์ (E.F.Codd) ได้เสนอผลงานทางวิชาการเกี่ยวกับโมเดลใหม่อีกโมเดลหนึ่ง อันได้แก่โมเดลเชิงสัมพันธ์ หลักการของ ดร.คอดด์ ได้รับความสนใจจากนักวิจัยและบริษัทผู้ค้าเครื่องคอมพิวเตอร์เป็นอย่างมาก และผลงานที่มีความสำคัญยิ่งต่อวงการฐานข้อมูลพัฒนาขึ้นตามแนวความคิดของ ดร.คอดด์ ได้แก่ระบบ R ซึ่งเป็นระบบที่สร้างขึ้นโดยบริษัทไอบีเอ็ม แต่กว่าที่โมเดลเชิงสัมพันธ์จะได้ก้าวเข้าสู่วงการธุรกิจก็ใช้เวลาถึง 10 ปี และระบบ R ก็เป็นเพียงระบบต้นแบบที่ยังคงใช้อยู่เฉพาะในห้องวิจัยเท่านั้น โดยไอบีเอ็มได้สร้างระบบ DB2 ขึ้นมาแทน เพื่อนำออกสู่งานธุรกิจ

ในช่วงปี พ.ศ. 2525 เป็นต้นมา ถือเป็นยุคทองของระบบฐานข้อมูลที่ได้ก้าวเข้าสู่ตลาดคอมพิวเตอร์อย่างเต็มตัว จนกระทั่งในปัจจุบัน ก็ได้มีซอฟต์แวร์ที่เกิดขึ้นมาเกื้อหนุนการทำงานของระบบการจัดการฐานข้อมูลอีกมากมาย ไม่ว่าจะเป็นระบบพจนานุกรมข้อมูล ซอฟต์แวร์ที่ช่วยในการออกแบบและสร้างข่ายงาน และภาษาเรียกค้น เป็นต้น

การเจริญเติบโตของระบบฐานข้อมูลยังคงไม่หยุดนิ่ง พร้อมๆ กับเทคโนโลยีคอมพิวเตอร์ รุดหน้าไปอย่างรวดเร็ว เครื่องไม้เครื่องมือทางซอฟต์แวร์ที่จะช่วยในการประมวลผลข้อมูลก็เกิดขึ้นเรื่อยๆ ไม่ว่าจะเป็นภาษารุ่นที่ 4 หรือที่นับว่าใหม่ที่สุดในขณะนี้ก็คือ ระบบ CASE (Computer Aided Software Engineering)

## 2.3 ความรู้พื้นฐานเกี่ยวกับระบบฐานข้อมูล

ในหัวข้อนี้ จะกล่าวถึงความรู้พื้นฐานที่ควรทราบ และแนะนำให้ผู้จักคำศัพท์ที่เกี่ยวข้องกับระบบฐานข้อมูล ซึ่งในหลายส่วนอาจเป็นศัพท์ที่คุ้นเคยกันแล้ว

### 2.3.1 คำศัพท์พื้นฐาน

1) ข้อมูล (Data) หมายถึง ข้อเท็จจริงขั้นต้น ซึ่งอาจจะเรียกว่าเป็นวัตถุดิบของสารสนเทศ เมื่อข้อมูลถูกนำมาประมวลผล (เรียงลำดับ, แยกประเภท, เชื่อมโยง จำนวนหรือสรุปผล) และจัดให้อยู่ในรูปแบบที่นำมาใช้ประโยชน์ได้เราจึงเรียกว่าเป็น สารสนเทศ

2) บิต (Bit) หมายถึง หน่วยของข้อมูลที่มีขนาดเล็กที่สุด

3) ไบต์ (Byte) หมายถึง หน่วยของข้อมูลที่เกิดจากการนำบิตหลายๆ บิตมารวมกันเป็น ตัวอักษร

4) ฟิลด์ (Field) หมายถึง หน่วยของข้อมูลที่ประกอบด้วยตัวอักษรหลายๆ ตัวอักษร เพื่อ

แทนความหมายของสิ่งๆ หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) เรคอร์ด (Record) หมายถึง หน่วยของข้อมูลที่เกิดจาก การนำเอาฟิลด์หลายๆ ฟิลด์ มา รวมกัน เพื่อแสดงรายละเอียดของข้อมูลเรื่องใดเรื่องหนึ่ง

6) เพิ่มข้อมูล (File) หมายถึง หน่วยของข้อมูลที่เกิดจากการนำเอาเรคอร์ดหลายๆ เรคอร์ด มารวมกันสำหรับในระบบฐานข้อมูล

7) ฐานข้อมูล (Database) คือ โครงสร้างของสารสนเทศ (Information) ที่ประกอบด้วย Entity หลายๆ ตัว ซึ่ง Entity เหล่านี้มีความสัมพันธ์กัน

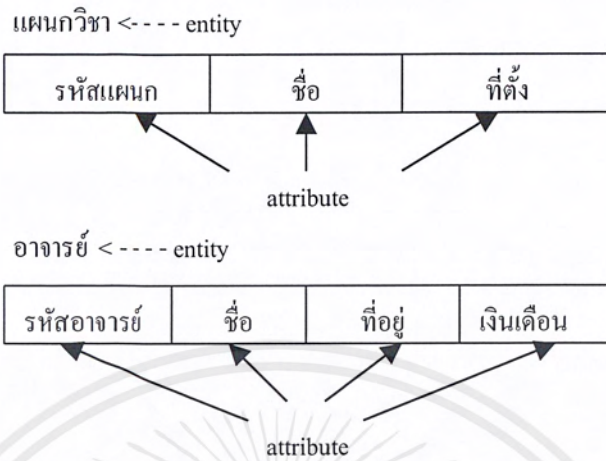
### 2.3.2 Entity, Attribute และความสัมพันธ์

คำว่า Entity เปรียบเสมือนกับเป็นคำนาม อันได้แก่ บุคคล สถานที่ และสิ่งของ เช่น Entity ในระบบฐานข้อมูลก็ได้แก่ อาจารย์ แผนกวิชา แผนกการประกันสุขภาพ ประวัติการทำงาน หรือถ้า เรากำลังสนใจในการสร้างระบบฐานข้อมูล เกี่ยวกับระบบการขายของบริษัทแห่งหนึ่ง Entity ของ ระบบนี้ก็จะได้แก่ พนักงานขาย ลูกค้า การสั่งซื้อ และสินค้า เป็นต้น

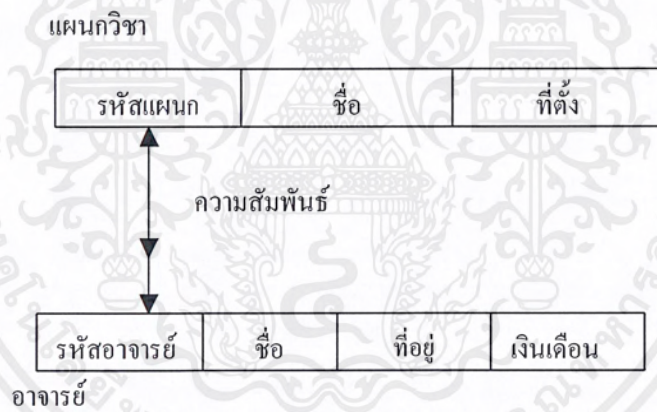
Attribute ก็คือ ข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เช่น Attribute ของบุคคลก็ อาจจะได้แก่ สีมผม สีตา อายุ เพศ ชื่อ ฯลฯ และถ้าจะออกแบบ Attribute ของอาจารย์ที่เรา สนใจได้ แก่ รหัสอาจารย์ ชื่อ ที่อยู่ เงินเดือนและอื่นๆ ดังที่แสดงในรูปที่ 2.1 เป็น Entity 2 ตัว ได้แก่อาจารย์ และแผนกวิชา โดยที่แผนกวิชาเป็น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

ส่วนความสัมพันธ์นั้นหมายถึง ความสัมพันธ์ระหว่าง Entity ตัวอย่างเช่น ความสัมพันธ์ ระหว่างแผนกวิชาและอาจารย์ ก็เป็นในลักษณะที่ว่าแผนกวิชาที่อาจารย์นั้นสังกัดอยู่ ส่วนความ สัมพันธ์ระหว่างอาจารย์และแผนกวิชา เป็นในลักษณะที่ว่าอาจารย์ที่ทำงานอยู่กับแผนกวิชานั้นๆ

ในรูปที่ 2.1 เป็นการแสดงความสัมพันธ์ระหว่าง Entity ทั้งสองนี้โดยใช้หัวลูกศรแสดง ความสัมพันธ์ ซึ่งในรูปนี้ความสัมพันธ์จากแผนกวิชาไปสู่อาจารย์เป็นความสัมพันธ์แบบหนึ่งต่อ กลุ่ม (One-To-Many) กล่าวคือ ในแผนกวิชา 1 แผนกจะมีอาจารย์สังกัดอยู่ได้มากกว่า 1 คน ในทาง กลับกันความสัมพันธ์จากอาจารย์ไปแผนกวิชาเป็นลักษณะแบบกลุ่มต่อหนึ่ง (Many-To-one) ทั้งนี้ก็ เพราะมีอาจารย์มากกว่า 1 คน ทำงานอยู่ในแผนกวิชา 1 แผนก ขอให้สังเกตหัวลูกศรที่แสดงความ สัมพันธ์แบบนี้ในรูปที่ 2.2 จะเห็นว่า หัวลูกศรเดียวแสดงความเป็น “หนึ่ง” และหัวลูกศร 2 หัว แสดงความเป็น “กลุ่ม” คือ จาก 1 แผนกวิชาไปยังอาจารย์หลายคน หรือในบางครั้งเราอาจจะใช้หัว ลูกศรเดียว แสดงความเป็นกลุ่ม และไม่มีหัวลูกศรแสดงความเป็น 1 ก็ได้ ดังเช่นที่แสดงในรูปที่ 2.3

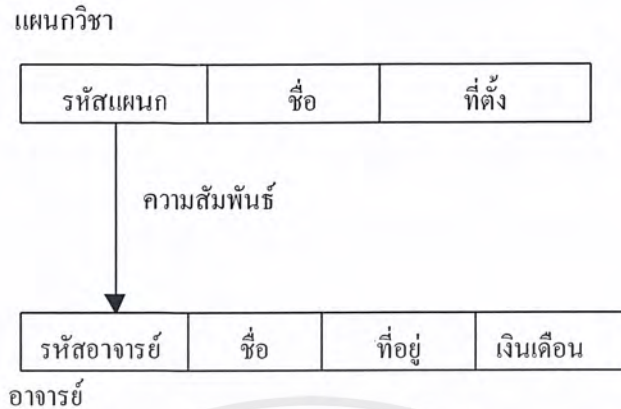


รูปที่ 2.1 ตัวอย่าง Entity และ Attribute



รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรคู่แสดงความเป็นกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยใช้หัวลูกศรเดียวแสดงความเป็นกลุ่ม

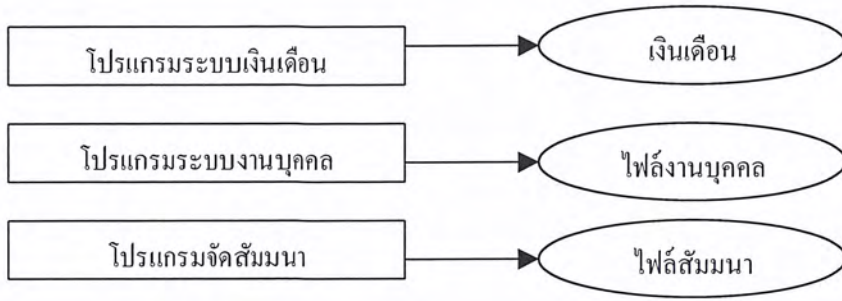
### 2.3.3 รีเลชัน (Relation)

รีเลชัน คือ ตาราง 2 มิติ ที่มีลักษณะดังนี้

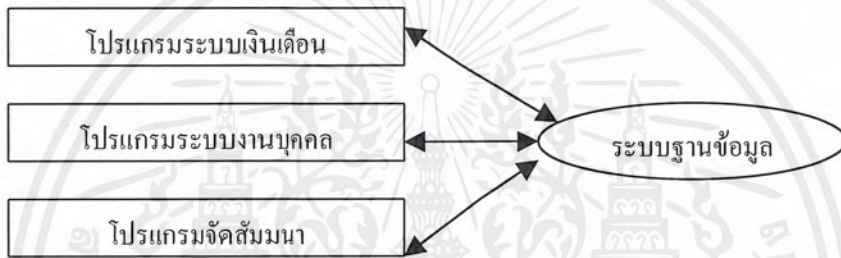
- 1) แต่ละช่องของตารางจะบรรจุข้อมูลเพียงค่าเดียว
- 2) ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน ได้แก่ ชื่อของ Attribute
- 3) ค่าข้อมูลที่อยู่ในแต่ละคอลัมน์ ได้แก่ ค่าของ Attribute ที่ระบุไว้ในหัวข้อคอลัมน์นั้นๆ
- 4) การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
- 5) ข้อมูลในแต่ละแถวจะต้องแตกต่างกัน
- 6) การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ

### 2.3.4 ฐานข้อมูล (Database)

ในการประมวลผลไฟล์ต่างๆ ไปนั้น ผู้ใช้แต่ละคนจะมีไฟล์เฉพาะงาน และโปรแกรมที่เขียนขึ้นมา เพื่อดึงเอาข้อมูลจากไฟล์ส่วนตัวมาใช้งานตามต้องการดังรูปที่ 2.4 การที่ผู้ใช้แต่ละกลุ่มต่างก็เก็บข้อมูลไว้ในไฟล์เช่นนี้ ก่อให้เกิดปัญหาที่เด่นชัด นั่นก็คือเกิดการซ้ำซ้อนของข้อมูลขึ้น เช่น ที่อยู่อาจารย์แต่ละคนจะปรากฏซ้ำๆ กันอยู่ในหลายๆ ไฟล์ความซับซ้อนเหล่านี้ นอกจากจะทำให้เกิดความสิ้นเปลืองเนื้อที่ในการจัดเก็บข้อมูลแล้ว ยังก่อให้เกิดปัญหาขึ้นเมื่อมีการเปลี่ยนแปลงค่าข้อมูล เพราะอาจจะหลงลืมเปลี่ยนข้อมูลในบางไฟล์ ทำให้ค่าข้อมูลเดียวกันที่เก็บในแต่ละไฟล์มีค่าไม่ตรงกัน ปัญหาดังกล่าว ทำให้เกิดความคิดที่ว่าจะนำข้อมูลเหล่านี้มาเก็บรวบรวมไว้ในที่เดียวกัน หรือจะเก็บไว้ในฐานข้อมูลแทนที่จะเก็บไว้ในไฟล์ ดังรูปที่ 2.5



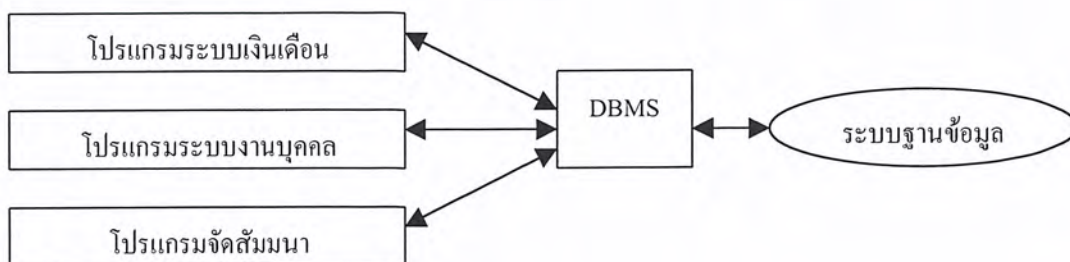
รูปที่ 2.4 การจัดระบบไฟล์



รูปที่ 2.5 การใช้ระบบฐานข้อมูล

### 2.3.5 ระบบการจัดเก็บฐานข้อมูล

การควบคุมดูแลการใช้ฐานข้อมูลนั้นเป็นเรื่องที่ยุงยากกว่าการใช้ไฟล์มาก เพราะว่าเราจะต้องตัดสินใจว่าโครงสร้างในการเก็บข้อมูลควรจะเป็นเช่นไร และการเขียนโปรแกรมเพื่อสร้างและเรียกใช้ข้อมูลจากโครงสร้างไฟล์เหล่านี้ ก็เป็นเรื่องที่ยุงยากพอๆ และถ้าเกิดโปรแกรมเหล่านี้ทำงานผิดพลาดขึ้นมา ก็จะทำให้เกิดความเสียหายต่อโครงสร้างของข้อมูลทั้งหมดได้ เพื่อเป็นการลดภาระการทำงานของผู้ใช้ จึงได้มีซอฟต์แวร์ขึ้นมาตัวหนึ่งมีชื่อเรียกว่า ระบบการจัดการฐานข้อมูล หรือ DBMS ซึ่งจะทำหน้าที่ในการควบคุมดูแลการสร้างและเรียกใช้งานฐานข้อมูล โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายใน โครงสร้างของฐานข้อมูล เปรียบเสมือนว่า DBMS นี้จะเป็นสื่อกลางระหว่างผู้ใช้ และ โปรแกรมต่างๆที่เกี่ยวข้องกับการใช้ฐานข้อมูลดังรูปที่ 2.6



รูปที่ 2.6 การใช้ระบบฐานข้อมูลด้วย DBMS

### 2.3.6 โมเดล (Model)

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

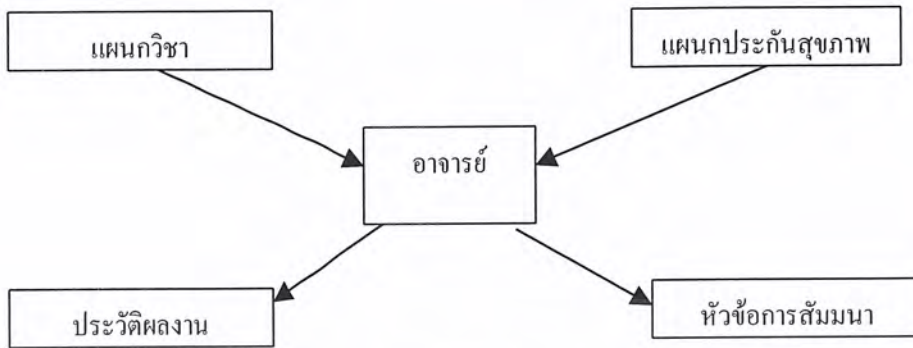
โครงสร้าง หมายถึง โครงสร้างของระบบข้อมูล ซึ่งอาจไม่ใช่โครงสร้างที่จัดเก็บจริงๆ ก็ได้ แต่อย่างน้อยก็คือ โครงสร้างในแง่การมองของผู้ใช้ DBMS

การใช้งาน คือ วิธีการที่จะให้ผู้ใช้สามารถเรียกดู และแก้ไขข้อมูลในระบบได้ โดยไม่ต้องรู้ว่าจริงๆ แล้วข้อมูลมีการจัดเก็บอย่างไร

ในปัจจุบันมีโมเดลอยู่ 3 ชนิด

1) โมเดลเชิงสัมพันธ์ (Relational model) เป็นการเก็บข้อมูลแบบเป็นตาราง (Table) ซึ่งตารางนี้ก็คือ รีเลชัน (Relation) มีลักษณะเป็น 2 มิติ คือ ด้านแถว (Row) และด้านคอลัมน์ (Column) ดังตัวอย่างที่แสดงในรูปที่ 2.7 เป็นฐานข้อมูลเชิงสัมพันธ์ที่ประกอบด้วย Entity ต่างๆ ซึ่งข้อมูลของแต่ละ Entity จะถูกจัดเก็บในลักษณะของตาราง Entity

2) โมเดลเน็ตเวิร์ค (Network model) โมเดลแบบนี้เป็นลักษณะของการรวบรวมเรคคอร์ดต่างๆ และความสัมพันธ์ระหว่างเรคคอร์ด ความแตกต่างที่เห็นได้ชัดระหว่างโมเดลเชิงสัมพันธ์และแบบเน็ตเวิร์คคือ ในโมเดลเชิงสัมพันธ์จะแฝง (Implicit) การแสดงความสัมพันธ์เอาไว้ (หมายความว่าเรคคอร์ดที่มีความสัมพันธ์กันจะต้องมีค่าของข้อมูลในฟิลด์ใดฟิลด์หนึ่งเหมือนกัน) ส่วนการแสดงความสัมพันธ์ในโมเดลแบบเน็ตเวิร์คจะเป็นไปอย่างชัดเจน (Explicit) คือ การแสดงไว้ในโครงสร้างอย่างชัดเจนดังรูปที่ 2.7



รูปที่ 2.7 โครงสร้างฐานข้อมูลแบบเน็ตเวิร์ค

จากรูปจะเห็นได้ว่า กรอบสี่เหลี่ยมแสดงถึงชนิดของเรคอร์ดในฐานข้อมูลซึ่งจะมี 1 อันต่อ 1 Entity เครื่องหมายลูกศรแสดงถึงความสัมพันธ์ ซึ่งในรูปนี้แสดงความสัมพันธ์แบบหนึ่งต่อกลุ่ม โดยหัวลูกศรจะออกจากส่วนของ “หนึ่ง” ไปยังส่วนของ “กลุ่ม” จะเห็นว่าแผนกวิชามีความสัมพันธ์กับอาจารย์ โดยที่ 1 แผนกวิชาอาจมีจำนวนอาจารย์ที่สังกัดได้มากกว่าหนึ่งคน วิธีการค้นหาข้อมูลสามารถทำได้โดยการใช้หัวลูกศรในการเชื่อมโยงความสัมพันธ์ ซึ่งบางครั้งอาจจะวิ่งทวนหัวลูกศรก็ได้

3) โมเดลแบบแตกสาขา (Hierar Chical Modal) ในโมเดลแบบนี้มีโครงสร้างเหมือนต้นไม้ (Tree) ที่จริงแล้วโมเดลแบบนี้ก็เหมือนกับโมเดลแบบเน็ตเวิร์ค แต่ต่างกันตรงที่ว่า โมเดลแบบแตกสาขานี้มีกฎเกณฑ์ที่เพิ่มขึ้นมาอีก 1 ข้อคือ ในแต่ละกรอบจะมีหัวลูกศรเข้าหาได้ไม่เกิน 1 หัว จากตัวอย่างในรูปที่ 2.7 จะเห็นว่ากรอบของอาจารย์มีลูกศรเข้ามา 2 ทาง ดังนั้นเราจึงไม่อาจสร้างฐานข้อมูลสำหรับตัวอย่างนี้ โดยใช้โมเดลแบบแตกสาขาด้วยวิธีปกติได้

## 2.4 ประโยชน์จากการประมวลผลด้วยฐานข้อมูล

ประโยชน์จากการใช้ฐานข้อมูลในการประมวลผล สามารถสรุปได้ดังนี้

1) ลดความซ้ำซ้อนของข้อมูล การประมวลผลโดยใช้ระบบไฟล์นั้นจำเป็นที่ผู้ใช้แต่ละคนจะต้องมีไฟล์ส่วนตัวเอาไว้ ดังนั้นจึงเกิดเหตุการณ์ที่ข้อมูลชนิดเดียวกันถูกเก็บไว้หลายๆ แห่งหรือที่เรียกว่าความซ้ำซ้อน การนำข้อมูลทั้งหมดมาเก็บไว้ที่เดียวกันในฐานข้อมูลนี้ เป็นการลดความซ้ำซ้อนของข้อมูลได้

2) สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง ประโยชน์ในข้อนี้ สืบเนื่องมาจากข้อที่แล้ว เพราะการเก็บข้อมูลไว้หลายๆ แห่งอาจจะก่อให้เกิดปัญหาที่ว่า การแก้ไขข้อมูลเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันทำไม่เหมือนกันในทุกๆ แห่งทำให้ข้อมูลชุดเดียวกันอาจมีค่าไม่ตรงกัน ดังนั้นการใช้ระบบฐานข้อมูลทำให้เราสามารถลดความซ้ำซ้อนลง โดยมี DBMS เมื่อเกิดการแก้ไขข้อมูลขึ้น ก็จะแก้ไขให้เหมือนกันครบทุกแห่ง

3) สามารถใช้ข้อมูลร่วมกันได้ ซึ่งการใช้ข้อมูลร่วมกันได้นี้ ไม่ได้จำกัดอยู่เฉพาะโปรแกรมที่ใช้ข้อมูลอยู่ในปัจจุบันเท่านั้น แต่กินความถึงโปรแกรมประยุกต์ที่จะพัฒนาขึ้นมาใหม่ด้วยที่สามารถใช้ข้อมูลที่มีอยู่ได้เลย โดยไม่จำเป็นต้องเพิ่มเติมข้อมูลเข้าไปในระบบอีก

4) สามารถควบคุมความเป็นมาตรฐานได้ จากการนำข้อมูลมารวมกันไว้ในฐานข้อมูล ทำให้ผู้มีหน้าที่ควบคุมดูแลการใช้ระบบฐานข้อมูลสามารถกำหนดมาตรฐานของข้อมูลขึ้นมาได้ เช่น การให้หน่วยวัดมาตรฐานที่เหมือนกัน รูปแบบการเขียนวันที่ให้เหมือนกัน เป็นต้น การใช้มาตรฐานเดียวกันนี้ ทำให้การแลกเปลี่ยนข้อมูลระหว่างระบบเป็นไปอย่างสะดวกและถูกต้อง เราเรียกผู้ควบคุมระบบว่า ผู้บริหารฐานข้อมูล (Data Base Administrator) หรือ DBA

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

6) สามารถควบคุมความคงสภาพของข้อมูลได้ โดยความไม่คงสภาพของข้อมูล คือการที่เกิดความขัดแย้งของข้อมูลดังที่ได้กล่าวไปแล้ว ซึ่งในกรณีของความขัดแย้งนี้จะเกิดขึ้นได้ก็เมื่อข้อมูลมีความซ้ำซ้อนเท่านั้น แต่ในอีกแง่หนึ่งของความคงสภาพอาจเกิดขึ้นได้ แม้ว่าจะไม่มีความซ้ำซ้อน เช่น ข้อมูลเกี่ยวกับอายุของพนักงานของบริษัทอาจจะมีค่า 300 แทนที่จะเป็น 30 ซึ่งความผิดพลาดนี้เกิดขึ้นได้ง่ายๆ จากความสะเพร่าในการกรอกข้อมูลก็ได้ ในลักษณะของความไม่ถูกต้องนี้ ผู้ที่ออกแบบระบบฐานข้อมูลสามารถใส่กฎเกณฑ์ลงไปเพื่อควบคุมความคงสภาพไว้ เช่น ตามตัวอย่างนี้อาจจะใส่กฎว่า ค่าของอายุจะต้องเป็นเลขระหว่าง 16 ถึง 60 เป็นต้น

7) สามารถสร้างความสมดุล ในความขัดแย้งของความต้องการได้ การที่ผู้ใช้ทั้งหมดขององค์กรใช้ข้อมูลจากฐานข้อมูลร่วมกันเช่นนี้ ทำให้ DBA ทราบถึงความต้องการและความสำคัญของผู้ใช้งานทั้งหมด จึงสามารถกำหนดโครงสร้างของฐานข้อมูลเพื่อการให้บริการที่ดีที่สุดได้ เช่น เลือกเก็บข้อมูลจะต้องใช้บ่อยๆ ไว้ในสื่อข้อมูลที่มีความเร็วเป็นพิเศษ เป็นต้น

8) เกิดความเป็นอิสระของข้อมูล วิธีการที่จะทำความเข้าใจว่า ความเป็นอิสระของข้อมูลนั้นคืออะไร ก็โดยการลองดูในด้านตรงข้ามก่อนว่า ข้อมูลที่ไม่เป็นอิสระนั้นอย่างไร ข้อมูลที่ไม่เป็นอิสระก็คือข้อมูลที่ถูกนำมาประยุกต์ใช้ยังมีความผูกพันอยู่กับวิธีการจัดเก็บและเรียกใช้ข้อมูลซึ่งในลักษณะการเขียนโปรแกรมประยุกต์บางประเภท เราอาจจำเป็นต้องใส่เทคนิคการจัดเก็บ และ

เรียกใช้ข้อมูลไว้ในตัวโปรแกรมด้วย นั่นหมายความว่า ถ้าเกิดต้องมีการเปลี่ยนแปลงวิธีการจัดเก็บ หรือเรียกใช้แล้ว ผู้ใช้จำเป็นที่จะต้องสร้างวิธีการประยุกต์ใช้ขึ้นมาใหม่ ซึ่งเป็นความไม่สะดวกอย่างยิ่ง และทำให้เราหมดโอกาสที่จะปรับปรุงโครงสร้างของข้อมูล เพื่อให้ใช้งานได้อย่างมีประสิทธิภาพมากขึ้น

## 2.5 สถาปัตยกรรมของระบบฐานข้อมูล

สถาปัตยกรรมของระบบฐานข้อมูล หมายถึง องค์ประกอบต่างๆ ของระบบฐานข้อมูลซึ่งสามารถแบ่งเป็นระดับได้ 3 ระดับ ดังรูปที่ 2.8 ซึ่งได้แก่

- 1) ระดับภายใน (Internal หรือ Physical level) เป็นระดับที่ต่ำที่สุด ได้แก่ ระดับของการจัดเก็บข้อมูลจริงๆ
- 2) ระดับหลักการ (Conceptual level) เป็นระดับที่อยู่ถัดลงมา ได้แก่ ระดับของการมอง Entity และความสัมพันธ์ระหว่าง Entity ทั้งหมดรวมทั้งกฎเกณฑ์ต่างๆ เกี่ยวกับข้อมูล และผู้มีสิทธิ์จะใช้ เป็นต้น
- 3) ระดับภายนอก (External หรือ View level) เป็นระดับที่อยู่สูงที่สุด เป็นข้อมูลที่จะมองเห็นจากการใช้งานของผู้ใช้แต่ละคน



รูปที่ 2.8 ระดับของข้อมูล 3 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.1 ความเป็นอิสระของข้อมูล

หมายถึง การที่ข้อมูลในแต่ละส่วนจะไม่มีผลกระทบต่อกันเมื่อมีการเปลี่ยนแปลงต่างๆ เกิดขึ้น สามารถแบ่งเป็น 2 ชนิดคือ

1) ความเป็นอิสระแบบกายภาพ (Physical data independence) คือ ลักษณะของการเปลี่ยนแปลงแก้ไขข้อมูลในระดับภายใน จะไม่มีผลกระทบต่อโครงสร้างของระดับหลักการและระดับภายนอก

2) ความเป็นอิสระแบบตรรก (Logical data independence) คือ ลักษณะของการเปลี่ยนแปลงข้อมูลในระดับหลักการ โดยที่จะไม่มีผลกระทบต่อระดับของผู้ใช้ภายนอก

### 2.5.2 Instance และ Database Schema

1) Instance หมายถึง ข้อมูลขณะใดขณะหนึ่งของฐานข้อมูล ในระหว่างที่มีการเรียกใช้และแก้ไขข้อมูล

2) Database schema หมายถึง เคี้ยวร่างที่ได้จากการออกแบบฐานข้อมูลโดยรวม สามารถแบ่งออกได้เป็นระดับคล้ายๆ กับระดับของข้อมูล คือ ระดับต่ำสุด ได้แก่ ระดับภายในหรือกายภาพ ระดับกลาง ได้แก่ ระดับหลักการ และระดับนอกสุด หรือระดับที่เรียกว่า Subschema โดยทั่วๆ ไป ระบบฐานข้อมูลจะประกอบด้วย Schema แบบกายภาพ 1 ตัว แบบหลักการ 1 ตัว และมี Subschema ได้หลายๆ ตัว

### 2.5.3 ระบบการจัดการฐานข้อมูล

ในการใช้ระบบฐานข้อมูลนั้น ผู้ใช้ไม่จำเป็นต้องทราบถึงรายละเอียดของการจัดเก็บข้อมูลจริงๆ แต่เป็นหน้าที่ของระบบการจัดเก็บฐานข้อมูล หรือ DBMS ซึ่งเป็นซอฟต์แวร์ที่จะทำหน้าที่ดูแลการใช้งานให้กับผู้ใช้เหล่านี้ โดยที่ DBMS มีหน้าที่ต่างๆ ดังนี้

1) ทำหน้าที่ติดต่อกับตัวจัดเก็บระบบไฟล์ เนื่องจากการใช้งานส่วนใหญ่ของระบบฐานข้อมูล คือ ลักษณะการใช้งานกับข้อมูลขนาดใหญ่ ซึ่งก็เป็นที่แน่นอนว่า เราไม่สามารถนำข้อมูลทั้งหมดนี้เข้ามาไว้ในหน่วยความจำหลักพร้อมๆ กันทั้งหมดได้ กล่าวคือ ข้อมูลทั้งหมดจะเก็บไว้ในดิสก์ และจะถูกนำมาสู่หน่วยความจำหลักเฉพาะส่วนที่ต้องการใช้งาน หน้าที่ในการค้นหาข้อมูลที่เรากำลังต้องการนั้นเก็บอยู่ที่ตำแหน่งใดในดิสก์ เป็นฟังก์ชันการทำงานส่วนหนึ่งของระบบดำเนินงาน (Operating System) อันได้แก่ส่วนที่เรียกว่า ตัวจัดเก็บระบบไฟล์ (File Manager) นั่นก็คือ DBMS จะทำหน้าที่ประสานงานกับตัวจัดเก็บระบบไฟล์ในการจัดเก็บ เรียกใช้ และแก้ไขข้อมูล

2) การควบคุมคุณภาพ เป็นหน้าที่ของ DBMS ที่จะต้องควบคุมค่าของข้อมูลในระบบให้อยู่ในกรอบที่ถูกต้องตามที่กำหนดไว้ในส่วนของ Schema เช่น ยอดเงินในบัญชีเงินฝากประจำจะ

ต้องไม่น้อยกว่า 100 บาท หรือรหัสของอาจารย์ที่ปรึกษาของนักเรียนต้องเป็นรหัสที่มีปรากฏอยู่ในเรคคอร์ดของอาจารย์ที่ปรึกษา ดังนั้นการเปลี่ยนแปลงแก้ไขข้อมูลทุกครั้งจึงเป็นหน้าที่ของ DBMS ที่จะต้องดูแลให้ผลลัพธ์ถูกต้องอยู่เสมอ

3) การควบคุมระบบความปลอดภัย ได้แก่ การป้องกันไม่ให้ผู้ที่มิได้รับอนุญาตเข้ามาเห็นหรือแก้ไขข้อมูลในส่วนที่ต้องการปกป้องเอาไว้

4) การสร้างระบบสำรองและการฟื้นฟูสภาพ ได้แก่ ฟังก์ชันในการจัดทำข้อมูลสำรอง และเมื่อใดก็ตามที่มีปัญหาเกิดขึ้น ไม่ว่าจะเป็นการขัดข้องของระบบไฟล์ หรือเครื่องเกิดความเสียหาย DBMS จะต้องใช้ระบบข้อมูลสำรองในการฟื้นฟูสภาพ ให้ระบบข้อมูลกลับเข้าสู่สภาพที่ถูกต้องสมบูรณ์ได้

5) การควบคุมภาวะพร้อมกัน หน้าที่ที่สำคัญที่สุดอีกประการหนึ่ง คือ การควบคุมการใช้ข้อมูลในสภาพที่มีผู้ใช้พร้อมๆ กันหลายคน อันได้แก่ การควบคุมภาวะพร้อมกัน (Concurrency Control) กล่าวคือ DBMS จะต้องควบคุมลำดับการทำงานให้เป็นไปอย่างถูกต้อง เช่น ระหว่างที่กำลังแก้ไขข้อมูลส่วนหนึ่งยังไม่เสร็จก็จะไม่อนุญาตให้ผู้อื่นเข้ามาเปลี่ยนแปลงแก้ไขข้อมูลนั้นได้

#### 2.5.4 ผู้บริหารฐานข้อมูล

ผู้บริหารฐานข้อมูล หรือ DBA คือ ผู้ที่มีหน้าที่ควบคุมการบริหารของระบบฐานข้อมูลทั้งหมด หน้าที่ความรับผิดชอบของผู้บริหารฐานข้อมูลได้แก่

- ตัดสินว่าจะรวมข้อมูลใด เข้าไว้ในระบบบ้าง
- วิเคราะห์ และตัดสินใจว่า ควรจะจัดเก็บข้อมูลด้วยวิธีใด และใช้เทคนิคใด ในการเรียกใช้ข้อมูล
- ประสานงานกับผู้ใช้
- กำหนดระบบความปลอดภัย และความคงสภาพของข้อมูล
- กำหนดแผนในการสร้างระบบข้อมูลสำรองและการฟื้นฟูสภาพ
- คอยสำรวจดูผลการทำงาน และตรวจตราความต้องการของผู้ใช้

#### 2.5.5 ผู้ใช้ระบบ

ระบบฐานข้อมูลเกิดขึ้นมา เพื่อที่จะสร้างสภาพการใช้ข้อมูลให้สะดวกสบายและให้ประสิทธิภาพแก่ผู้ใช้ให้ดีที่สุด ลักษณะของผู้ใช้ระบบแบ่งได้ 4 กลุ่มคือ

1) ผู้เขียนโปรแกรมประยุกต์ ผู้ใช้กลุ่มนี้ได้แก่ นักเขียนโปรแกรมมืออาชีพ ที่เรียกใช้ ข้อมูลในระบบ โดยใช้การเขียนโปรแกรมด้วยภาษาชั้นสูง และแฝงเอาไว้ในโปรแกรม ในลักษณะเช่นนี้ เราจะเรียกส่วนของภาษาชั้นสูงว่าเป็นภาษาหลัก (Host Language)

2) ผู้ใช้ทั่วไป ได้แก่ เหล่าผู้ใช้ที่เรียกใช้ข้อมูลโดยผ่านทาง DML เท่านั้น ผู้ใช้เหล่านี้ไม่จำเป็นต้องมีความรู้เกี่ยวกับการเขียนโปรแกรม เนื่องจากคำสั่งของ DML จะเป็นคำสั่งง่ายๆ และสามารถเรียกใช้งานได้โดยสะดวก

3) โอเปอร์เรเตอร์ คือ กลุ่มผู้ใช้ที่จะทำการใส่ข้อมูลอย่างเดียว กลุ่มนี้ไม่จำเป็นต้องมีความรู้ใดๆ ทั้งสิ้นที่เกี่ยวกับระบบคอมพิวเตอร์ ลักษณะการทำงานก็ได้แก่ การใช้โปรแกรมประยุกต์ที่ผู้ใช้กลุ่มแรกสร้างไว้ แล้วก็เพียงแต่ทำตามที่จอภาพระบุ เช่น การใส่ข้อมูลต่างๆ เข้าไปเท่านั้น

4) ผู้ใช้พิเศษ ได้แก่ กลุ่มของผู้ใช้ที่นำระบบไปใช้ในลักษณะงาน ที่ไม่ใช้งานประมวลผลข้อมูล เช่น การสร้างโปรแกรมช่วยการออกแบบ (Computer Aided Design) หรือ CAD

### 2.5.6 การจัดเก็บข้อมูล

ข้อมูลที่เก็บอยู่ในฐานข้อมูลใดๆ ก็คือ ข้อมูลของ Entity การสร้างความสัมพันธ์ระหว่าง Entity ในฐานข้อมูลเชิงสัมพันธ์กระทำได้โดยการ กำหนดให้ Entity ที่มีความสัมพันธ์กัน มี 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	พีรพัฒน์
9	สโรจน์	789	ธนา

2) การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบหนึ่งต่อกลุ่ม สมมติความสัมพันธ์ ระหว่างพนักงานขาย และลูกค้า โดยที่พนักงานขายแต่ละคนสามารถติดต่อกับลูกค้าได้มากกว่า 1 คน แต่ลูกค้าแต่ละคนต้องซื้อของจากพนักงานขายเพียงคนเดียวเท่านั้น ซึ่งเป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม ในลักษณะนี้เราสามารถใช่ตาราง 1 ตาราง สำหรับเก็บข้อมูลของพนักงานขาย และแยกเก็บเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลของลูกค้าไว้อีกตารางหนึ่ง โดยตารางของลูกค้านี้จะเพิ่มขึ้นอีก 1 คอลัมน์สำหรับจัดเก็บค่ารหัสพนักงานขายที่ลูกค้าคนนั้นใช้บริการอยู่ โดยที่ค่ารหัสพนักงานขายนี้ ถือเป็นคีย์หลักในตารางของพนักงานขายดังตารางที่ 2.2

ตารางที่ 2.2 การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม

รหัสลูกค้า	ชื่อลูกค้า	รหัสลูกค้า
123	ปราณี	15
456	นันทนา	7
789	สินใจ	23

(ก) ข้อมูลลูกค้า

รหัสพนักงาน	ชื่อพนักงาน
7	วิมล
15	ดวงตา
23	สมบุรณ์

(ข) ข้อมูลพนักงานขาย

จะเห็นว่าข้อมูลของลูกค้าแต่ละคน จะปรากฏ ในแถวของตารางลูกค้าเพียงคนเดียวเท่านั้น รหัสพนักงานขายในแต่ละแถวของตารางนี้ก็จะมีเพียงคนเดียว ซึ่งเป็นตัวยืนยันว่าลูกค้าแต่ละคนมีสิทธิติดต่อกับพนักงานขายคนเดียว ส่วนในตารางของพนักงานขายนั้น ไม่มีข้อมูลที่เกี่ยวข้องกับลูกค้าเลย ดังนั้นถ้าเราต้องการทราบว่า พนักงานคนใดคนหนึ่งมีใครเป็นลูกค้าอยู่บ้าง ก็ต้องไล่ดูรหัสของพนักงานคนนั้นในตารางลูกค้าเอง ซึ่งอาจปรากฏอยู่มากกว่า 1 แถวก็ได้ แสดงว่าพนักงานแต่ละคนมีสิทธิที่จะมีลูกค้าได้มากกว่า 1 คน

3) การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบกลุ่มต่อกลุ่ม สมมติให้ลูกค้าสามารถสั่งซื้อสินค้าได้มากกว่า 1 อย่างในการสั่งซื้อแต่ละครั้ง และในขณะเดียวกันก็อนุญาตให้สินค้า 1 อย่างปรากฏอยู่ในรายการสั่งซื้อได้มากกว่า 1 ครั้ง จึงเห็นได้ว่าลักษณะความสัมพันธ์ระหว่างการสั่งและสินค้าเป็นแบบกลุ่มต่อกลุ่ม ในกรณีเช่นนี้ เราต้องสร้างตารางขึ้นมาเป็นกรณีพิเศษ เพื่อใช้บรรจุค่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์ของตารางทั้งสองนี้ อันได้แก่ รหัสการสั่ง และรหัสสินค้า และตารางนี้ยังอาจประกอบด้วยข้อมูลอื่นๆ ที่มีค่าขึ้นอยู่กับค่าของทั้งการสั่ง และรหัสสินค้า เช่น จำนวนที่สั่ง และราคา ดังตารางที่ 2.3

ตารางที่ 2.3 การสร้างความสัมพันธ์แบบกลุ่มต่อกุ่ม

รหัสสินค้า	วันที่สั่ง	รหัสลูกค้า
12498	020841	124
12491	031241	311
12494	040941	315

(ก) ข้อมูลการสั่งซื้อ

รหัสสินค้า	ชื่อสินค้า
AX12	เตารีด
AF52	ลูกบอล
ZZ00	จักรยาน

(ข) ข้อมูลสินค้า

รหัสการสั่ง	รหัสสินค้า	จำนวนสินค้า	ราคา
12498	AX12	1	580
12491	AF52	2	450
12494	ZZ00	3	2150

(ค) บัญชีการสั่ง

ในการค้นหาว่า การสั่งตามหมายเลขที่เราสนใจนั้นมีการสั่งสินค้าอะไรบ้าง ก็เริ่มด้วยการดูข้อมูลในตารางบัญชีการสั่ง โดยดูในแถวที่มีรหัสการสั่งที่เราสนใจ ซึ่งเราจะได้รหัสของสินค้าที่อยู่ในแถวเดียวกัน และเราสามารถใช้รหัสสินค้านี้ไปดูข้อมูลของสินค้าจากตารางสินค้าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.7 ภาษาสำหรับนิยามข้อมูล

การกำหนด Schema ของฐานข้อมูลทั้ง 3 ระดับกระทำได้โดยใช้ภาษาพิเศษ ที่เรียกว่า ภาษาสำหรับนิยามข้อมูล DDL (Data Definition Language) ผลจากการคอมไพล์โปรแกรมที่เขียนด้วย DDL จะทำให้เกิดตารางที่จะจัดเก็บข้อมูลในไฟล์ชนิดหนึ่งที่เรียกว่า พจนานุกรมข้อมูล (Data Dictionary) ซึ่งจะเก็บข้อมูลเกี่ยวกับโครงสร้างที่ได้จากการออกแบบฐานข้อมูลนั้นๆ

### 2.5.8 ภาษาสำหรับการใช้ข้อมูล

การใช้ข้อมูลในระบบฐานข้อมูล แบ่งออกได้เป็น 3 ประเภท คือ

- 1) การเรียกดูข้อมูล
- 2) การเพิ่มข้อมูลใหม่เข้าไป
- 3) การลบข้อมูล

ในระดับของผู้ใช้จะไม่ทราบและไม่สนใจว่า วิธีการจัดเก็บข้อมูลจริง 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, คีย์ และนอร์มัลไลซ์

เป้าหมายของการใช้ระบบฐานข้อมูลเชิงสัมพันธ์ ก็คือ การออกแบบริเลชันเพื่อที่จะสามารถเรียกใช้ข้อมูลในระบบได้อย่างสะดวกและมีประสิทธิภาพ โดยพยายามให้เกิดความซ้ำซ้อนของข้อมูลน้อยที่สุด ศาสตร์แขนงหนึ่งที่เป็นพื้นฐานของการออกแบบฐานข้อมูล ได้แก่ การออกแบบระบบให้อยู่ในรูปแบบนอร์มัล (Normal Form) โดยอาศัยความรู้เกี่ยวกับฟังก์ชันการขึ้นต่อกัน (Functional Dependency) ของข้อมูล ก่อนที่จะกล่าวถึงวิธีการทำให้ระบบอยู่ในรูปแบบนอร์มัล จะกล่าวถึงกฎของความสัมพันธ์ที่เกิดขึ้นในโมเดลเชิงสัมพันธ์ก่อน

### 2.6.1 กฎของความคงสภาพ (Integrity Rule)

เมื่อกล่าวถึงกฎของความคงสภาพของโมเดลเชิงสัมพันธ์ โดยทั่วไปแล้วจะหมายถึง กฎความคงสภาพ 2 กรณี คือ

1) กฎความคงสภาพของ Entity (Entity Integrity Rule) บัญญัติไว้ว่า Attribute ทุกตัวที่เป็น ส่วนของคีย์หลักจะไม่อนุญาตให้มีค่าว่าง ความหมายของค่าว่างก็คือ ค่าที่ไม่ทราบแน่ชัด หรือค่าที่ไม่เหมาะสม

2) กฎความคงสภาพของการอ้างอิง (Referential Integrity Rule) บัญญัติไว้ว่าถ้าเรามีรีเลชัน R2 ซึ่งมี FK เป็นคีย์นอกที่อ้างอิงถึงคีย์หลัก PK ในรีเลชัน R1 สำหรับทุกๆ ค่าของ FK ใน R2 จะต้องมีลักษณะดังนี้

2.1) มีค่าเท่ากับค่า PK ใน tupleใด tuple หนึ่ง ใน R1

2.2) ค่าของ Attribute ทุกตัวใน FK เป็นค่าว่าง

ความสำคัญของกฎข้อนี้คือ เมื่อมีการอ้างอิงจากรีเลชันหนึ่งไปยังรีเลชันหนึ่งแล้ว เราต้องรับประกันว่าข้อมูลในรีเลชันที่สองจะต้องไม่เป็นค่าว่าง

เมื่อกล่าวถึงกฎความคงสภาพทั้งสองนี้ เราต้องการที่จะประกันความคงสภาพนี้ในทุกๆ สถานะและทุกๆ ขณะของระบบฐานข้อมูล นั่นหมายความว่า เมื่อมีการแก้ไขข้อมูลในระบบเกิดขึ้นเมื่อไร ก็เป็นหน้าที่ของระบบการจัดการฐานข้อมูลที่จะต้องตรวจสอบอยู่เสมอว่า ยังมีความคงสภาพที่ถูกต้องตามกฎเกณฑ์หรือไม่ และถ้าการแก้ไขข้อมูลครั้งใดที่ผิดกฎเกณฑ์ความคงสภาพไปแล้ว ก็ขึ้นอยู่กับวิธีการออกแบบของ DBMS แต่ละตัวว่าจะแก้ไขสถานการณ์อย่างไร

วิธีการแก้ไขสถานการณ์นี้แบ่งออกได้เป็น 2 วิธี คือ

วิธีที่ 1 ระบบจะไม่ยอมรับการเปลี่ยนแปลงแก้ไขข้อมูลที่ทำให้เกิดปัญหานั้นๆ โดยอาจจะแสดงข้อความออกมาบอกผู้ใช้

วิธีที่ 2 ระบบจะอนุญาตให้การแก้ไขนั้นๆ เกิดขึ้นได้ โดยระบบจะทำการปรับผลลัพธ์ให้ยังคงมีความคงสภาพตามกฎทั้ง 2 โดยอาจจะเพิ่มเติมการกระทำอะไรบางอย่างกับฐานข้อมูลก็ได้

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

1) ค่าของคีย์นอกสามารถเป็นค่าว่างได้หรือไม่

2) จะเกิดอะไรขึ้น ถ้ามีการออกคำสั่งให้ลบข้อมูลในรีเลชันที่ถูกอ้างอิงจากรีเลชันอื่น ซึ่งคำตอบสำหรับคำถามนี้เป็นไปได้ 3 ทางคือ

2.1) กระทำแบบเป็นทอดๆ (Cascade)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2) ยอมแบบมีเงื่อนไข (Restricted)

2.3) เปลี่ยนเป็นค่าว่าง (Nullify)

3) จะเกิดอะไรขึ้นถ้ามีการออกคำสั่งให้แก้ไขค่าของคีย์หลักที่ถูกอ้างอิงด้วยคีย์ นอกจากรีเลชันอื่น ซึ่งคำตอบก็จะมี 3 ทาง คือ

3.1) กระทบแบบเป็นทอด (Cascade)

3.2) ยอมแบบมีเงื่อนไข (Restricted)

3.3) เปลี่ยนเป็นค่าว่าง (Nullify)

## 2.6.2 ฟังก์ชันการขึ้นต่อกัน

เนื่องจากปัญหาจะเกิดขึ้นเมื่อเราแก้ไขข้อมูลในระบบ มักจะมีสาเหตุมาจาก การที่ข้อมูลในส่วนต่างๆ 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 หรือจะพูดสั้นๆ 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 หลายๆ ตัว จะต้องไม่มี Attribute ที่เป็นส่วนใดส่วนหนึ่งของ A ที่มีคุณสมบัติดังที่ระบุในข้อแรก

## 2.6.4 รูปแบบนอร์มัลระดับที่ 1, 2 และ 3 (First, Second And Third Normal Form)

รูปแบบนอร์มัลทั้ง 3 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) ซึ่งได้แก่ รีเลชันที่มีข้อมูลในบางช่องมากกว่า 1 ค่า และจะต้องมีการเพิ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Attribute ของคีย์เสมอ โดยสามารถกำหนดได้ว่าคีย์ตัวใหม่จะประกอบด้วยคีย์เดิมผนวกกับ Attribute ที่ถือเป็นคีย์หลักของกลุ่มที่ซ้ำ

นิยาม รีเลชันใด ๆ กล่าวได้ว่าอยู่ในแบบนอร์มัลระดับที่ 1(1NF) ถ้ารีเลชันไม่มีกลุ่มที่ซ้ำกัน

## 2) รูปแบบนอร์มัลระดับที่ 2

ถึงจะจัดรูปแบบของรีเลชันให้เป็น 1 NF แล้วก็ตาม แต่ในบางครั้งก็ยังมีปัญหาต่างๆ เกิดขึ้นสังเกตจากตารางที่ 2.4 จะเห็นว่าเกิดความซ้ำซ้อนเมื่อมีการสั่งซื้อสินค้าชนิดเดียวกันขึ้นมา จะเห็นว่าข้อมูลที่เกี่ยวข้องกับ BT04 มีการซ้ำกัน ทำให้เปลืองเนื้อที่ และเกิดปัญหาในการแก้ไขข้อมูลซึ่งสามารถจัดแบ่งลักษณะของปัญหาออกได้เป็น 4 ประเภท คือ

2.1) การแก้ไขข้อมูล ถ้าเราต้องการแก้ไขข้อมูลเกี่ยวกับ BT04 เราจำเป็นต้องตามแก้ไขในหลายแถว นอกจากจะทำให้เสียเวลาแล้ว ยังไม่สามารถรับประกันได้ว่าการแก้ไขจะเกิดขึ้นครบทุกแถวที่มี BT04 ตามหลักแล้วการแก้ไขข้อมูลอย่างเดียวควรจะทำเพียงแห่งเดียว

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	ตุ้มน้ำหนัก	2	2160

2.2) ความขัดแย้งของข้อมูล เนื่องจากเรามีข้อมูลที่กล่าวถึง สินค้าที่มีชนิดเดียวกันปรากฏอยู่ในหลายแถว จึงมีความเป็นไปได้ว่าค่าของข้อมูลชนิดเดียวกันในแต่ละแถวอาจไม่ตรงกัน เช่น ในตารางจะเห็นว่า ในบางแถวราคาของของเตา มีค่าเท่ากับ 8000 ในขณะที่อีกแถวหนึ่งระบุว่ามีค่าเท่ากับ 8500 เป็นต้น

2.3) การเพิ่มเติมข้อมูล ถ้ามีการเพิ่มเติมข้อมูลลงไปในเรคคอร์ด เนื่องจากรีเลชันของเรากำหนดให้รหัสการสั่ง และรหัสสินค้าเป็นคีย์ร่วมกัน นั่นหมายความว่า การเพิ่มเติมรหัสสินค้าจะต้องมีรหัสการสั่งด้วย อาจทำให้เกิดปัญหาได้

2.4) การลบข้อมูล ถ้ามีการลบข้อมูลรหัสการสั่งออกไปจากตาราง ก็จะทำให้ระบบเสียข้อมูลของรหัสสินค้าไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าตัวการที่ทำให้เกิดปัญหาก็คือ การที่ Attribute บางตัวขึ้นอยู่กับเพียงบางส่วนของคีย์ ดังนั้นหลักการของการทำให้เป็น 2NF ก็คือ การจัดการขึ้นต่อกันเพียงบางส่วน

นิยาม เราเรียก Attribute ที่ไม่ใช่ส่วนหนึ่งส่วนใดของคีย์หลักว่า Attribute นันคีย์ (Nonkey Attribute)

นิยาม รีเลชันใดๆ 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	เฉลี่ย	18 สุขสวัสดิ์	3	ธนา
256	ยงยุทธ	21 สุขุมวิท 49	12	ประจวบ
311	พิชัย	58 เจริญกรุง	6	มณีรัตน์
405	มณี	46 วิทยู	12	ประจวบ
587	วัชร	25 ลาดพร้าว	3	ธนา

3.1) การแก้ไขข้อมูล ถ้ามีการแก้ไขข้อมูลเกี่ยวกับพนักงานขาย จะต้องทำการแก้ไขหลายแห่ง

3.2) ความขัดแย้งของข้อมูล ผลสืบเนื่องจากการแก้ไขข้อมูลที่ปรากฏหลายแห่ง อาจจะทำให้แก้ไขไม่ครบถ้วนทุกแห่ง ทำให้ข้อมูลแต่ละแห่งไม่ตรงกัน

3.3) การเพิ่มเติมข้อมูล จะไม่สามารถเพิ่มพนักงานขายเข้าไปได้ตราบใดที่พนักงานคนใหม่นี้ยังไม่มีลูกค้า

3.4) การลบข้อมูล ถ้ามีการลบข้อมูลเกี่ยวกับลูกค้าของพนักงานขายหมายเลข 12 ออกจากระบบ จะทำให้ข้อมูลของพนักงานขายคนนี้หายไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

นิยาม รีเลชันใดๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 3 (3NF) ถ้ารีเลชันนั้นเป็น 2NF และ ตัวเลือกทุกตัวจะต้องเป็นคีย์คู่แข่ง

การขจัดตัวเลือกที่ไม่ใช่คีย์คู่แข่งออกไปมีวิธีการดังนี้

- 1) คึง Attribute ที่ขึ้นกับตัวเลือกที่ไม่ใช่คีย์คู่แข่งออกไปสู่รีเลชันใหม่
- 2) กำหนดให้ตัวเลือกที่เกี่ยวข้องเป็นคีย์ของรีเลชันใหม่นี้

#### 4) รูปแบบนอร์มัลระดับที่ 4

การนอร์มัลไลซ์รีเลชัน ถ้าทำได้ถึงระดับ 3NF แล้วจะสามารถแก้ไขปัญหาที่เกิดขึ้นได้จาก ฟังก์ชันการขึ้นต่อกันไปอย่างหมดสิ้น แต่จริงๆ แล้วยังมีลักษณะของการขึ้นอยู่อีกชนิดหนึ่ง ที่สามารถก่อให้เกิดปัญหาประเภทเดียวกันนี้อีก ซึ่งการแก้ปัญหาจะต้องทำโดยการนอร์มัลไลซ์ให้ ถึงระดับที่ 4

สาเหตุที่ทำการนอร์มัลไลซ์ให้ถึงระดับที่ 3 แล้วยังมีปัญหาเกิดขึ้นอีก เพราะว่ามีฟังก์ชันของการขึ้นต่อกันระหว่าง Attribute หรือจะมีก็แต่ลักษณะของการขึ้นต่อกันเชิงกลุ่ม

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

นิยาม รีเลชันที่เป็น 4NF ได้แก่ รีเลชันที่เป็น 3NF และเป็นรีเลชันที่ไม่มีการขึ้นต่อกันเชิงกลุ่ม

การปรับรีเลชันให้เป็น 4NF สามารถทำได้โดยการแยกรีเลชันเดิม โดยให้แต่ละรีเลชันประกอบด้วย Attribute ที่เกี่ยวข้องกันอยู่เท่านั้น

## 2.7 การออกแบบฐานข้อมูล

สิ่งสำคัญที่สุดประการหนึ่ง ในการประมวลสารสนเทศด้วยระบบฐานข้อมูล คือ การออกแบบฐานข้อมูล ความหมายของการออกแบบฐานข้อมูล คือ การที่ผู้พัฒนาระบบจะต้องพิจารณาว่า เรคคอร์ดแต่ละตัวควรประกอบด้วยฟิลด์อะไรบ้าง แต่ละฟิลด์ควรจะมีชนิดอะไร ขนาดเท่าไร เรคคอร์ดแต่ละชนิดควรมีความสัมพันธ์กันอย่างไร

โดยทั่วไปแล้ว การออกแบบระบบฐานข้อมูล สามารถแบ่งออกได้เป็น 2 ระดับ ระดับแรกเรียกว่า การออกแบบระดับสารสนเทศ (Information-Level Design) คือ ส่วนของการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการสงวนสิทธิ์ในเนื้อหา ไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์รวบรวมความต้องการของผู้ใช้เอาไว้ โดยที่การออกแบบในระดับนี้มีเป้าหมายเพื่อให้การใช้งานเป็นไปตามความต้องการของผู้ใช้มากที่สุด ในจุดนี้จะไม่ให้ความสำคัญกับชนิด และชื่อของ DBMS ที่จะใช้ โดยเราจะเริ่มศึกษาเกี่ยวกับ DBMS ที่จะใช้ในลำดับที่ 2 ซึ่งเรียกว่าระดับกายภาพ (Physical-Level Design) อันเป็นระดับที่เราเริ่มให้ความสำคัญต่อประสิทธิภาพของระบบ

การออกแบบใน 2 ระดับนี้มีความสำคัญเท่าเทียมกัน เพราะว่าการออกแบบในระดับข่าวสารที่ไม่ดีย่อมจะมีผลกระทบต่อประสิทธิภาพของระบบด้วย และในขณะเดียวกัน ถึงแม้จะออกแบบในระดับข่าวสารไว้อย่างดีเลิศ แต่ถ้าวการออกแบบในระดับกายภาพไม่ดีพอ ก็จะทำให้การใช้งานของระบบล้มเหลวไปด้วย

### 2.7.1 เป้าหมายของการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลมีเป้าหมายอยู่ที่การสร้างประสิทธิภาพของการใช้งานให้แก่ผู้ใช้ ดังนั้นจึงพอที่จะเดากันได้ว่าขั้นตอนแรกของการออกแบบข้อมูลก็คือการศึกษา วิเคราะห์และรวบรวมเอาความต้องการของผู้ใช้ให้สมบูรณ์ที่สุด ทั้งนี้นอกจากความต้องการแล้ว ผู้ออกแบบก็จะต้องรวบรวมกฎเกณฑ์และข้อบังคับต่างๆ เอาไว้ด้วย ซึ่งสามารถสรุปรายละเอียดต่างๆ ที่จำเป็นต้องทราบในส่วนนี้ได้ ดังนี้

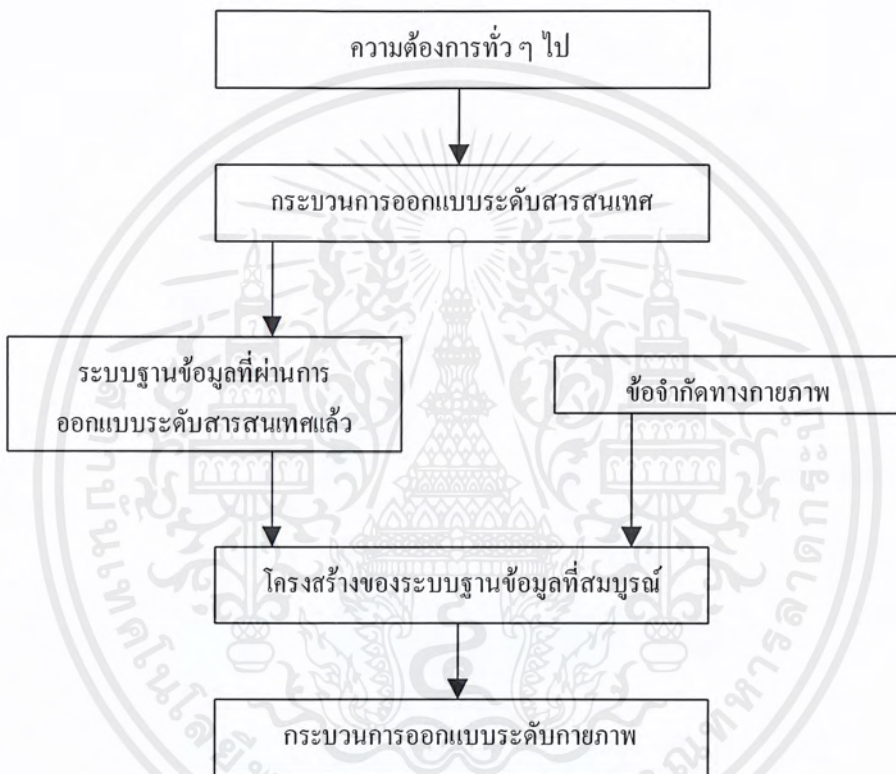
- 1) ลักษณะของรายงานทั้งหมด
- 2) การค้นหาข้อมูลในทุกลักษณะ
- 3) เอาต์พุตที่ต้องส่งให้แผนกอื่น หรือระบบอื่น
- 4) การประมวลและแก้ไขข้อมูลทั้งหมด
- 5) การคำนวณทุกอย่าง
- 6) กฎเกณฑ์ข้อบังคับต่างๆ เช่น การห้ามไม่ให้ลูกค้านใดใช้บริการของพนักงานขายที่ไม่มีหมายเลขปรากฏอยู่ในตารางของพนักงานขาย
- 7) การตั้งชื่อพ้อง (Synonym) ต่างๆ เช่น ในแต่ละแผนกหรือผู้ใช้แต่ละคนอาจจะเรียกชื่อของ Attribute ตัวเดียวกันแตกต่างกัน

ข้อมูลเหล่านี้จะถูกนำมาประมวลในการออกแบบฐานข้อมูลในระดับสารสนเทศ และเมื่อถึงเวลาของการออกแบบในระดับกายภาพแล้ว เรายังต้องอาศัยข้อมูลอื่นๆ ที่เกี่ยวข้องกับการทำงานของระบบในระดับกายภาพอีก เช่น

- จำนวนของแต่ละ Entity
- ความถี่ในการพิมพ์รายงาน
- กฎเกณฑ์ในการควบคุมความปลอดภัยในการใช้ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลเหล่านี้จะถูกนำมาวิเคราะห์พร้อมกับระบบฐานข้อมูลที่ได้ออกแบบในระดับสารสนเทศไว้แล้ว เพื่อการออกแบบในระดับกายภาพ โดยในระดับนี้เราจะเริ่มคำนึงถึงความสามารถ DBMS ที่กำลังใช้อยู่ด้วยแล้ว เพื่อให้ผลการทำงานของระบบที่ออกแบบมานี้สมบูรณ์และมีประสิทธิภาพมากที่สุด โดยสรุปแล้วขั้นตอนของการออกแบบระบบฐานข้อมูลจะเป็นไปตามรูปที่ 2.9



รูป 2.9 ขั้นตอนการออกแบบฐานข้อมูล

### 2.7.2 ความต้องการของผู้ใช้แต่ละคน

ไม่ว่าผู้ออกแบบจะใช้เทคนิคในการออกแบบฐานข้อมูล การออกแบบเพียงครั้งเดียวให้ได้มาซึ่งระบบที่ถูกต้องสมบูรณ์ตามความต้องการของผู้ใช้แต่ละคนนั้นนับว่าเป็นเรื่องยาก ยกเว้นในกรณีของฐานข้อมูลขนาดเล็ก หลักการของการออกแบบฐานข้อมูลก็เหมือนกับการออกแบบและพัฒนาซอฟต์แวร์อื่นๆ ก็คือแบ่งงานใหญ่ออกเป็นงานย่อยๆ หลายๆ ชิ้น และค่อยๆ เริ่มออกแบบจากจุดเล็กๆ นี้ไปก่อน ก็คือเริ่มจากการออกแบบสำหรับความต้องการของผู้ใช้ทีละคนหรือกลุ่มย่อยๆ ก่อน ซึ่งจะมีความยุ่งยากน้อยกว่า แล้วจึงเพิ่มขอบเขตของผู้ใช้ออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.3 หลักการพื้นฐานของการออกแบบฐานข้อมูล

ในกรณีที่ได้รวบรวมความต้องการของกฎข้อบังคับต่างๆ ไว้จากผู้ใช้งานระบบเรียบร้อยแล้ว กระบวนการในการออกแบบมี 5 ขั้นตอน ดังนี้

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

ขั้นตอนที่ 1 การเปลี่ยนรูปแบบของความต้องการให้อยู่ลักษณะของรีเลชัน

การสร้างรีเลชันก็คือ การวิเคราะห์ว่าฐานข้อมูลควรมีรีเลชันอะไรบ้าง และในรีเลชันแต่ละตัวควรมีฟิลด์ใดเป็นคีย์ บางครั้งการออกแบบในส่วนนี้ออกจะตรงไปตรงมา และง่ายจนน่าตกใจ เช่น ถ้าเราต้องการเก็บข้อมูลเกี่ยวกับพนักงาน และ แผนกที่พนักงานนั้นสังกัดอยู่ จะเห็นว่าฐานข้อมูลควรประกอบด้วยรีเลชัน 2 ตัวคือ แผนก และ พนักงาน ดังนี้คือ

แผนก (รหัสแผนก, ชื่อแผนก)

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

โดยมีฟิลด์ที่ขีดเส้นใต้เป็นคีย์ของรีเลชันนั้นๆ

- 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 เพราะ

ผู้ออกแบบจะต้องใช้วิจารณญาณ และเหตุผลในการตัดสินใจถึงการออกแบบจุดต่างๆ ซึ่งจะเป็นตัวตัดสินใจถึงความสัมพันธ์ที่จะเกิดขึ้นระหว่าง Entity ในฐานข้อมูล

ขั้นตอนที่ 4 พิจารณาข้อจำกัดและกฎเกณฑ์อื่นๆ

ในขั้นตอนนี้ ผู้ออกแบบจะต้องรวบรวมความต้องการจากผู้ใช้ว่ามีข้อจำกัดอะไรบ้าง เช่น อนุญาตให้นำค่าให้ลูกค้าคนใดมียอดเงินค้างชำระสูงกว่าวงเงินในเครดิต เป็นต้น ในลักษณะเช่นนี้ DBMS บางตัวจะอนุญาตให้เราระบุข้อจำกัดนี้ในส่วนของตาราง Schema ได้เลย แต่สำหรับ DBMS บางตัวที่ไม่มีขีดความสามารถนี้ ผู้ออกแบบก็จำเป็นต้องรายงานข้อกำหนดนี้ไว้ในโปรแกรมที่เกี่ยวข้องกับการประมวลผลข้อมูลนั้นๆ

ขั้นตอนที่ 5 นำผลที่ได้จากการออกแบบใน 4 ขั้นตอนแรกมาผนวกกัน

หลังจากที่เราได้ผลของการออกแบบทั้ง 4 ขั้นตอนสำหรับผู้ใช้คนหนึ่งหรือกลุ่มหนึ่งแล้ว เราจะนำการออกแบบนี้ไปผนวกกับการออกแบบที่สร้างขึ้นสำหรับผู้ใช้คนอื่นๆ หรือกลุ่มอื่นๆ

## 2.8 ภาษา SQL

ภาษาสำหรับการใช้ข้อมูล (DML) ที่เป็นที่ยอมรับสำหรับโมเดลเชิงสัมพันธ์ คุณสมบัติสำคัญของภาษาเหล่านี้ คือ สามารถใช้เรียกดูข้อมูลตามที่ระบุได้ สามารถใช้ในการเปลี่ยนแปลงเพิ่มเติมและลบข้อมูลออกจากระบบ นอกจากนี้ยังสามารถใช้ในการสร้างระบบฐานข้อมูลด้วย ดังนั้นเราจึงสามารถพูดได้ว่าภาษาสำหรับการใช้ข้อมูลนี้เป็นภาษาสำหรับนิยามข้อมูลโดยเสร็จสรรพด้วย

SQL เป็นภาษาที่แทบจะกล่าวได้ว่าเป็นที่ยอมรับมากที่สุด ถือกำเนิดขึ้นในห้องปฏิบัติการวิจัยของบริษัทไอบีเอ็ม ที่เมือง San Jose ของมลรัฐแคลิฟอร์เนีย ในช่วงกลางทศวรรษที่ 1970 โดยมี Chamberlin เป็นผู้บัญญัตินิยามขึ้นมา และได้เริ่มพัฒนาระบบทดลองขึ้นบนระบบ R โดยในตอนนั้นได้ใช้ชื่อว่า SEQUEL (ถูกเปลี่ยนเป็น SQL ในปี พ.ศ. 2523) ผลการทดลองครั้งนั้นเป็นที่น่าพอใจมาก ดังนั้นบริษัทไอบีเอ็ม จึงได้ให้ความสนับสนุนอย่างจริงจังในการพัฒนาระบบดังกล่าว และในปี พ.ศ. 2525 ไอบีเอ็มก็ได้ปล่อยตัวระบบดังกล่าวออกสู่วงการธุรกิจภายใต้ชื่อของ SQL/DS (Structured Query Language/Data System) เพื่อใช้งานกับเครื่องระดับ 370, 30XX, 43XX, รวมทั้งระบบใหม่ล่าสุดคือ 937X ซึ่งทั้งหมดนี้ทำงานภายใต้ระบบการดำเนินงาน DOS/VSE และ VM

นอกจาก SQL/DS แล้วระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ของไอบีเอ็มอีกตัวหนึ่งที่มีชื่อเสียงไม่แพ้กัน ก็คือ DB2 ซึ่งก็มีรากฐานการพัฒนาโดยอาศัยต้นแบบจาก SQL ที่ทำงานภายใต้ระบบ R เช่นเดียวกัน โดยที่ DB2 เป็นระบบที่ทำงานภายใต้ระบบการดำเนินงาน MVS

รูปแบบคำสั่ง SQL กระทำได้ 3 วิธี ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ออกคำสั่งแบบออนไลน์ กล่าวคือ ผู้ใช้สามารถพิมพ์ประโยคคำสั่งผ่านทางเทอร์มินัล โดยที่คำสั่งเหล่านี้จะปฏิบัติงานโดยทันที

2) ส่งคำสั่งในลักษณะงานออฟไลน์ หรืองาน Batch ลักษณะการใช้งานประเภทนี้เหมาะกับการสร้างรายงาน หรือประเภทของงานที่ไม่ต้องการทราบผลโดยทันที

3) สอดแทรกประโยคคำสั่งไว้ใน โปรแกรมประยุกต์ที่เขียนขึ้นมา สำหรับการใช้ระบบฐานข้อมูล

เราสามารถสรุปคำสั่ง SQL ได้ดังตารางที่ 2.6

ตารางที่ 2.6 สรุปคำสั่งของ SQL

คำสั่งสำหรับเรียกดูข้อมูล	
SELECT	เรียกดูข้อมูลจากตารางได้จาก 1 หรือมากกว่า 1 ตาราง
คำสั่งสำหรับการใช้ข้อมูล	
INSERT	เพิ่มข้อมูล 1 แถว หรือมากกว่า 1 แถวลงในตาราง
UPDATE	แก้ไขข้อมูลใน 1 แถว หรือมากกว่า 1 แถว
DELETE	ลบข้อมูล 1 แถว หรือมากกว่า 1 แถวจากตาราง
คำสั่งสำหรับนิยามข้อมูล	
CREATE TABLE	นิยาม โครงสร้างข้อมูลในตารางที่จะสร้างขึ้นใหม่
DROP TABLE	ลบ โครงสร้างของตารางข้อมูลออกจากระบบ
ALTER TABLE	เพิ่มคอลัมน์ใหม่ในตาราง
CREATE INDEX	สร้างดัชนีของตารางซึ่งประกอบด้วย 1 คอลัมน์ หรือมากกว่า
DROP INDEX	ยกเลิกดัชนีออกจากระบบ
CREATE VIEW	นิยามวิวของผู้ใช้
DROP VIEW	ลบวิวออกจากระบบ

คำสั่ง SQL แบ่งได้ดังนี้

1) การเรียกดูข้อมูล ประโยคคำสั่งสำหรับการเรียกดูข้อมูลได้แก่ คำสั่ง SELECT รูปแบบของคำสั่งนี้คือ

SELECT.....FROM.....WHERE.....ORDER BY.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้สามารถระบุส่วนต่างๆ ในคำสั่งได้ดังนี้

ส่วนหลัง SELECT	ใช้ระบุคอลัมน์หรือกลุ่มของคอลัมน์ที่ต้องการดูข้อมูล
ส่วนหลัง FROM	ใช้ระบุชื่อของตารางที่ต้องการดูข้อมูล
ส่วนหลัง WHERE	ใช้ระบุเงื่อนไขของข้อมูลที่สนใจ ซึ่งอาจจะมีหรือไม่มีก็ได้
ส่วนหลัง ORDER BY	ใช้ระบุวิธีการเรียงลำดับการแสดงผลข้อมูล

2) การใช้ข้อมูล คำสั่งสำหรับการใช้ข้อมูลได้แก่กลุ่มคำสั่งที่ใช้ในการเพิ่มเติมแก้ไข และลบข้อมูลออกจากตาราง ซึ่งตารางเหล่านี้จะต้องได้รับการสร้างไว้แล้วด้วยคำสั่งสำหรับการนิยามข้อมูล

2.1) คำสั่ง INSERT คำสั่งนี้ใช้ในการเพิ่มเติมข้อมูลแถวใหม่ลงในตาราง โดยอาจจะระบุข้อมูลไว้ในคำสั่งเลย หรือดึงข้อมูลออกมาจากตารางอื่นก็ได้ โดยบังคับไว้ว่า ข้อมูลที่จะเติมลงไปนี้จะต้องมีข้อมูลในแต่ละคอลัมน์ตรงตามคอลัมน์ในตารางที่จะเพิ่มเติมลงไปด้วย

2.2) คำสั่ง UPDATE คำสั่งนี้ใช้ในการเปลี่ยนแปลงแก้ไขค่าของคอลัมน์ที่ต้องการเฉพาะในแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้ โดยเราสามารถแก้ไขได้มากกว่า 1 คอลัมน์ในคำสั่งเดียวกัน

รูปแบบของคำสั่งนี้คือ

```
UPDATE ชื่อตาราง
SET ชื่อคอลัมน์ = ค่าข้อมูล
WHERE เงื่อนไข
```

2.3) คำสั่ง DELETE คำสั่งนี้ใช้ในการลบแถวของข้อมูลที่สอดคล้องกับเงื่อนไขที่ระบุออกจากตาราง

รูปแบบของคำสั่งนี้คือ

```
DELETE ชื่อตาราง WHERE เงื่อนไข
```

3) ฟังก์ชันต่างๆ คำสั่งที่ใช้ในการเรียกดูข้อมูลมีอยู่เพียงคำสั่งเดียวคือ คำสั่ง SELECT ซึ่งถ้าเราต้องการความสามารถในด้านการคำนวณและการนับแล้ว จะต้องใช้ฟังก์ชันต่างๆ เข้ามาช่วยเพื่อเพิ่มความสะดวกในการเรียกดูข้อมูล ฟังก์ชันต่างๆ นี้ได้แก่

COUNT	ใช้ในการนับจำนวน
SUM	ใช้รวมค่าในคอลัมน์ที่กำหนด
AVG	แสดงค่าเฉลี่ยของคอลัมน์ที่กำหนด
MAX	แสดงค่ามากที่สุดของข้อมูลในคอลัมน์ที่กำหนด
MIN	แสดงค่าน้อยที่สุดของข้อมูลในคอลัมน์ที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GROUP BY      จับกลุ่มของข้อมูลที่น่าสนใจเข้าด้วยกัน  
 HAVING        เหมือนคำสั่ง WHERE แต่ใช้กับกรณีที่มีการจัดกลุ่ม

3.1) คำสั่ง **CREATE TABLE** การสร้างตารางในระบบฐานข้อมูลกระทำได้โดยใช้คำสั่ง **CREATE TABLE** ซึ่งคำสั่งนี้เราต้องใส่ชื่อตารางพร้อมทั้งชื่อของคอลัมน์ ชนิดและขนาดของข้อมูลของแต่ละคอลัมน์

รูปแบบของคำสั่งนี้คือ

**CREATE TABLE** ชื่อตาราง

(ชื่อคอลัมน์1    ชนิด(ขนาดของข้อมูล),  
 ชื่อคอลัมน์1    ชนิด(ขนาดของข้อมูล),  
 ชื่อคอลัมน์1    ชนิด(ขนาดของข้อมูล),.....)

3.2) คำสั่ง **DROP TABLE** เป็นคำสั่งที่ใช้ในการลบตารางออกจากระบบ

รูปแบบของคำสั่งนี้คือ

**DROP TABLE** ชื่อตาราง

3.3) คำสั่ง **CREATE VIEW** เป็นคำสั่งในการนิยามวิว ซึ่งสามารถระบุได้ว่าประกอบด้วยคอลัมน์ใดบ้าง โดยคอลัมน์เหล่านี้อาจจะมาจากตารางมากกว่า 1 ตาราง นอกจากนี้ยังสามารถระบุได้ว่าวิวที่นิยามขึ้นมาแล้ว ผู้มีสิทธิใช้วิวก็จะมองเห็นว่าวิวเหล่านี้ก็คือตารางนั่นเอง เราถือว่าวิวเป็นเพียง ตารางสมมติ (Virtual Table) เท่านั้น ดังนั้นผู้ที่มีสิทธิใช้วิวจะสามารถใช้คำสั่ง SQL เรียกดูข้อมูลจากวิวนี้ได้เต็มที่ แต่ถ้าผู้ใช้ต้องการจะเปลี่ยนแปลง เพิ่มเติม หรือลบข้อมูลออกจากวิว ก็อาจจะติดขัดข้อบังคับบางอย่าง

รูปแบบของคำสั่งนี้คือ

**CREATE VIEW** ชื่อวิว

[(ชื่อคอลัมน์1, ชื่อคอลัมน์2,.....)]

AS SELECT ประโยคคำสั่ง Select

3.4) คำสั่ง **ALTER TABLE** เป็นคำสั่งที่ใช้เพิ่มเติมฟิลด์ใหม่เข้าไปในตารางได้ โดยคอลัมน์ใหม่จะมีค่าเริ่มต้นเป็นค่าว่าง

รูปแบบของคำสั่งนี้คือ

**ALTER TABLE** ชื่อตาราง

ADD ชื่อคอลัมน์ ชนิด(ขนาดของข้อมูล)

3.5) คำสั่ง **CREATE INDEX** ในการเรียกใช้ข้อมูล การระบุดัชนีที่เหมาะสม จะช่วยให้การค้นหาข้อมูลเป็นไปอย่างรวดเร็วยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของคำสั่งนี้คือ

CREATE INDEX ชื่อดัชนี

ON ชื่อตาราง (ชื่อคอลัมน์1 [,ชื่อคอลัมน์2]...)

### 3.6) คำสั่ง DROP INDEX ใช้ในการยกเลิกการระบุดัชนี

รูปแบบของคำสั่งนี้คือ

DROP INDEX ชื่อดัชนี

## 2.9 ภาษาสคริปต์ PHP (Professional Home Page)

ภาษาสคริปต์ PHP พัฒนาโดย Rasmus Lerdorf ทั้งนี้เพราะ Rasmus เคยเขียนเว็บเพจด้วยภาษา Perl มาก่อน แล้วพบว่าเว็บเพจผลลัพธ์ที่ได้นั้นยังไม่เป็นที่น่าพอใจ เขาจึงได้พัฒนาภาษาสคริปต์ขึ้นมาเพื่อใช้ในการเขียนเว็บเพจเอง และได้อาศัยเค้าโครงของภาษา Perl เป็นต้นแบบในการสร้างภาษาสคริปต์ดังกล่าวขึ้นมา โดยใช้โปรแกรมภาษา C++ เป็นเครื่องมือพัฒนา แรกเริ่มเดิมทีเขาเรียกภาษาสคริปต์นี้ว่า Personal Home Page หรือ PHP ก็เนื่องด้วยวัตถุประสงค์ของการนำไปใช้ในการเขียนโฮมเพจนั่นเอง

ต่อมาได้มีผู้ร่วมงานอีกหลายคนเข้ามาช่วยพัฒนา PHP โดยเพิ่มขีดความสามารถมากยิ่งขึ้น จนถูกกล่าวขานว่าน่าจะเป็น Professional Home Page มากกว่า Personal Home Pages ดังนั้นถ้าพูดถึง PHP ในปัจจุบันจะหมายถึงคำที่ย่อมาจาก Professional Home Page

PHP เป็นโปรแกรมที่แจกจ่ายฟรี ไม่มีปัญหาเรื่องลิขสิทธิ์ ปัจจุบันมีผู้นิยมใช้กันมากขึ้นเรื่อยๆ จากการสำรวจเมื่อประมาณปลายปี 2542 โดย [www.netcraft.com](http://www.netcraft.com) มีเว็บไซต์ที่ใช้ PHP จำนวนมากหลายแสนราย

ทั้งนี้ไม่เพียงเพราะว่าเป็น โปรแกรมที่แจกจ่ายฟรีเท่านั้น แต่มีประสิทธิภาพและสามารถทำงานได้หลากหลายรูปแบบ คือใช้ได้กับระบบปฏิบัติการหลายระบบ และทำงานได้หลากหลายรูปแบบ คือใช้ได้กับระบบปฏิบัติการหลายระบบ และทำงานร่วมกับโปรแกรมเว็บเซิร์ฟเวอร์ได้หลากหลาย ไม่ว่าจะเป็น Personal Web Server (PWS) ซึ่งใช้กับระบบปฏิบัติการ Windows 95 กับ 98 หรือ Internet Information Server (IIS) ซึ่งใช้กับ Windows NT หรือ จะใช้กับ Apache Web Server ภายใต้ระบบปฏิบัติการ Linux และระบบปฏิบัติการอื่นๆ ก็ได้

ตัวสคริปต์ที่เขียนขึ้นมาก็สามารถนำไปใช้งานข้ามระบบปฏิบัติการได้เลย โดยการเขียนและทดสอบใน Windows 98 (ใช้ Apache 1.3.9 win32 + PHP 3.0.11) แล้วคัดลอกตัวสคริปต์ใช้กับ Linux โดยแทบไม่ต้องแก้ไขตัวสคริปต์เลย หากในสคริปต์นั้นไม่ระบุว่าต้องอ้างอิงการใช้งานบางอย่างที่เป็นคุณสมบัติเฉพาะที่ต้องทำงานใน Windows เท่านั้น เช่น การใช้งาน ODBC เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะการเขียนเว็บเพจให้มีสคริปต์ PHP จะอาศัยวิธีการเขียนซอร์ซโค้ดให้อยู่ในรูปแบบของภาษาสคริปต์ PHP ทั้งหมดเลยก็ได้ (เหมือนกับที่เขียนเว็บเพจด้วยภาษา Perl) หรืออาจจะเขียนในรูปแบบการฝัง (embed) คำสั่งหรือฟังก์ชันของ PHP ลงไปเฉพาะในตำแหน่งที่ต้องการ ซึ่งก็เหมือนกับการเขียนเว็บเพจทั่วไปที่มีการฝังสคริปต์ภาษา HTML นั่นเอง

สคริปต์ PHP จะใช้แท็กในการกำหนดขอบเขตของสคริปต์ ซึ่งอาจเรียกว่า PHP script tag โดยประกอบไปด้วยแท็กเปิดและแท็กปิด แท็กเปิดของ PHP เขียนได้ 2 แบบ คือ <? หรือ <?php ส่วนแท็กปิดอยู่ในรูป ?> ในกรณีที่เป็น <?php ต้องปิดในรูป php?> เท่านั้น ห้ามเขียนเป็นแบบ <? ทั้งนี้เนื่องจากแท็กของ XML เขียนอยู่ในรูป <? และ ?> เหมือนกัน จึงต้องเลี่ยงการเขียนแท็กเปิดของ PHP ไปเป็น <?php เพื่อความแตกต่าง แต่ไม่ว่าจะเปิดแบบ <? หรือ <?php ก็ตาม วิธีการเขียนเว็บเพจแบบนี้เรียกว่าเป็นการเขียนในลักษณะฝังสคริปต์ หรือ embedded script นั่นเอง

เราจะพบเห็นการนำวิธีการฝังสคริปต์มาใช้ในการเขียนเว็บเพจมากขึ้นเรื่อยๆ ตัวอย่างเช่น การเขียนสคริปต์ ASP (Active Server Pages) ฝังลงในเว็บเพจ ก็จะมีเครื่องหมาย <% และ %> ใช้กำกับในการเปิดปิดส่วนที่เป็นสคริปต์ ASP เป็นต้น ทั้งนี้เพราะเป็นวิธีการเขียนเว็บเพจที่สะดวกต่อผู้เขียนในการตรวจสอบการทำงานของเว็บเพจ โดยส่วนของเว็บเพจที่ไม่ได้กำกับด้วยสคริปต์ใดๆ ก็จะแสดงผลไปตามข้อความนั้นๆ โดยตรง หากเราจะเปลี่ยนแปลงแก้ไขข้อความใดๆ ก็จะกระทำได้เลย โดยไม่ต้องกลัวว่าเว็บเพจจะทำงานไม่ถูกต้อง และเมื่อเว็บเพจแจ้งข้อความว่าเกิดข้อผิดพลาดอันเนื่องมาจากการทำงานของสคริปต์ เราก็เพียงไปแก้ไขหรือ ปรับปรุงเฉพาะจุดที่เป็นสคริปต์นั้นๆ

ข้อแตกต่างของสคริปต์ PHP กับสคริปต์ภาษา HTML คือ สคริปต์ PHP เป็น Server side script โดยถูกเรียกให้ทำงานทางฝั่งเซิร์ฟเวอร์ ส่วนสคริปต์ภาษา HTML เป็น Client side script นั่นคือ สคริปต์จะถูกเรียกทำงานทางฝั่งไคลเอนต์หรือฝั่งของเบราว์เซอร์

### 2.9.1 ติดต่อฐานข้อมูลโดยอาศัย ODBC driver

ODBC หรือ Open Data Base Connectivity คือ การติดต่อระหว่างโปรแกรมสองโปรแกรม เพื่อให้สามารถเรียกใช้ข้อมูลร่วมกันได้ แม้ว่าต่างก็เก็บข้อมูลในรูปแบบที่ไม่เหมือนกันก็ตาม อย่างเช่น การเขียนสคริปต์ PHP เพื่อเรียกใช้แฟ้มข้อมูลที่เก็บบันทึกในรูปแบบของ Microsoft Access

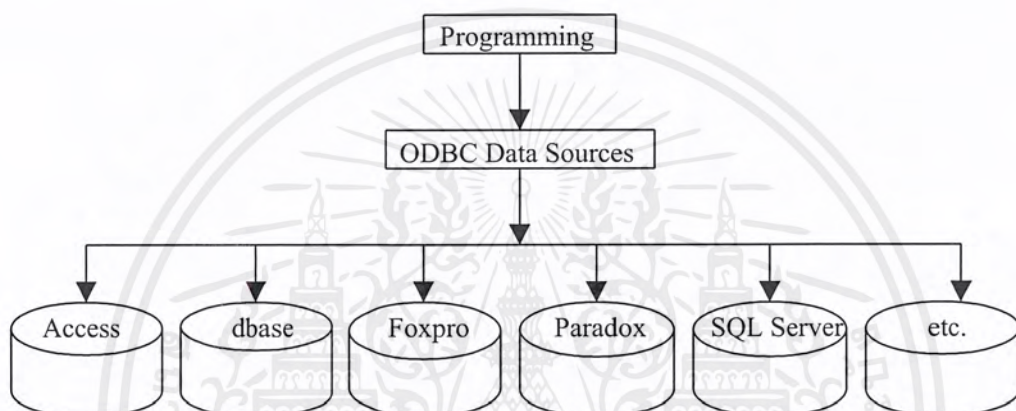
PHP มีฟังก์ชันสำหรับการติดต่อฐานข้อมูล ซึ่งสามารถใช้กับโปรแกรมจัดการระบบฐานข้อมูลยอดนิยมหลายๆ ระบบที่มีคุณสมบัติรองรับการติดต่อผ่านทาง ODBC (Open Data Base Connectivity) โดยจะอาศัยโปรแกรมประเภทไดรเวอร์ ที่มีมาพร้อมกับโปรแกรมจัดการระบบฐานข้อมูลทั้งหลาย ในการทำงาน

ข้อดีอย่างหนึ่งของ PHP คือ สามารถใช้ได้ทั้งระบบปฏิบัติการ Windows และ Unix เช่น Linux ในการติดต่อฐานข้อมูลชนิด ODBC ก็คือ DSN หรือ Data Source Name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทำงานกับ ODBC นั้นเราต้องเขียนโปรแกรมเพื่อติดต่อกับ ODBC ให้ได้เสียก่อน จากนั้นส่วนของการจัดการฐานข้อมูลจะเป็นหน้าที่ของ ODBC

จากหลักการทำงานดังกล่าวจะเห็นได้ว่าถ้าเราทำการเปลี่ยนชนิดของฐานข้อมูลก็จะมีผลต่อการแก้ไขโปรแกรม เพราะเนื่องจากโปรแกรมที่เราเขียนนั้นติดต่อกับ ODBC ไม่ได้ติดต่อกับฐานข้อมูลโดยตรง ดังนั้นถึงมีการแก้ไขใดๆ ที่ฐานข้อมูล เราก็ไม่จำเป็นที่จะต้องแก้ไขโปรแกรมที่เราเขียน เพียงแต่แก้ที่ตัว ODBC เท่านั้น



รูปที่ 2.10 โครงสร้างการทำงานของ ODBC

## 2.9.2 การเปิดการติดต่อกับฐานข้อมูล

สำหรับ ODBC นั้น PHP จะใช้ API (Application Programming Interface) ของ ODBC ในกาติดต่อซึ่งมีฟังก์ชันที่สำคัญดังนี้

`odbc_connect (dsn, user, password)`

ฟังก์ชันนี้เป็นการเปิดการติดต่อกับ ODBC ซึ่งต้องส่งค่า DSN, Username และ Password ให้กับฟังก์ชัน ส่วน Username และ Password นั้นถ้าฐานข้อมูลกำหนดไว้ก็ให้ใส่เข้าไปด้วย รูปแบบการใช้งานคือ

`odbc_connect (dsn, user, password);`

dsn เป็นข้อมูลชนิดข้อความ หมายถึง ชื่อ Data Source Name

user เป็นข้อมูลชนิดข้อความ หมายถึง ชื่อผู้ใช้

password เป็นข้อมูลชนิดข้อความ หมายถึง รหัสผ่านของผู้ใช้

ฟังก์ชันนี้จะคืนค่าออกมาเป็น False ถ้าการติดต่อกับ ODBC ไม่สำเร็จ และถ้าติดต่อกับ

ODBC ได้สำเร็จจะคืนค่าออกมาเป็นตัวเลข (Connection ID) ซึ่งเอาไว้ใช้ในการอ้างอิงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิด ODBC ได้เป็นที่เรียบร้อย ขั้นต่อไปก็คือใช้งานฐานข้อมูลได้ทันที ซึ่งการใช้งานส่วนใหญ่แล้วก็จะใช้ภาษา SQL ในการสั่งงาน โดยจะใช้ฟังก์ชันต่อไปนี้ในการสั่งงานด้วย SQL

`odbc_exec (connection_id, query_string)`

ผลลัพธ์ที่ได้จากฟังก์ชันนี้จะเป็นค่าตัวเลข (Result ID) ที่ใช้ในการอ้างอิงให้กับฟังก์ชันอื่นๆ อีกต่อไป ซึ่งค่านี้ก็จะเปรียบเสมือนตัวเลขอ้างอิงไปยังชุดข้อมูล(Recordset) ในฐานข้อมูลถ้ามีการใช้คำสั่ง SQL เป็น `Select Target from TableName` หรือค่านี้อาจไม่มีประโยชน์เลย ถ้ามีการใช้คำสั่งอื่นๆ ที่ไม่ใช่ `Select` เช่น `Alter ,... Insert,... Update...` เป็นต้น ซึ่งอาจจะบอกให้ทราบว่าฟังก์ชันทำงานสำเร็จหรือไม่โดยส่งค่ากลับมาเป็น `True` หรือ `False` เท่านั้น



## บทที่ 3

### การออกแบบ การสร้างและการทำงาน

โปรแกรม ECG บน Intranet นี้จะแบ่งส่วนของการออกแบบ การสร้างและการทำงานออกได้เป็น 3 ส่วนด้วยกันดังต่อไปนี้

- 1) การออกแบบและการสร้างฐานข้อมูล
- 2) การออกแบบและการสร้างโฮมเพจ
- 3) การเชื่อมต่อฐานข้อมูลกับโฮมเพจโดยใช้ PHP

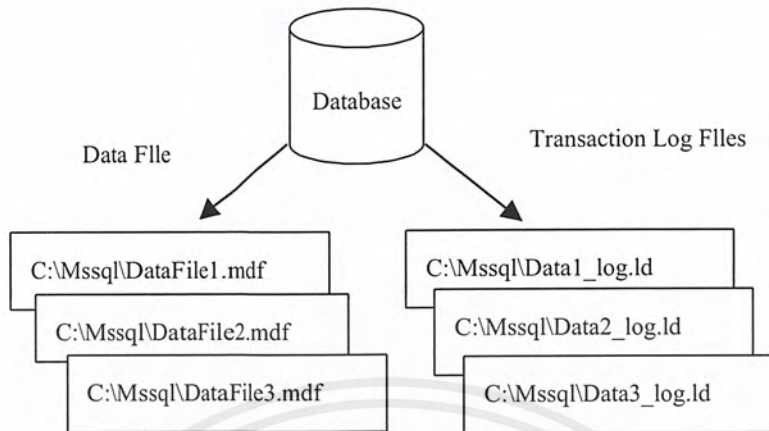
#### 3.1 การออกแบบและการสร้างฐานข้อมูล

##### 3.1.1 ระบบฐานข้อมูล

ก่อนที่จะเริ่มใช้งานเพื่อเก็บข้อมูลได้นั้น จะต้องมีการสร้าง และจัดการกับระบบฐานข้อมูลให้เรียบร้อยเสียก่อน ในการสร้างฐานข้อมูล (Database) หนึ่งๆ โดยปกติแล้วจะประกอบไปด้วยไฟล์อย่างน้อย 2 ไฟล์ คือ

- 1) ไฟล์ข้อมูล (Data File) เป็นไฟล์ที่เก็บข้อมูลจริง
- 2) ไฟล์ทรานแซคชันล็อก (Transaction Log File) เป็นไฟล์สำหรับเก็บโอเปอเรชัน หรือข้อมูลล็อก (log) ซึ่งเกิดขึ้นจากการทำงานกับข้อมูลจริง

สำหรับ SQL Server 7.0 แล้ว สามารถสร้างไฟล์ข้อมูล หรือไฟล์ทรานแซคชันล็อก มีมากกว่า 1 ไฟล์ได้ ทั้งนี้เพื่อแยกการทำงานที่ชัดเจน และประโยชน์ในแง่ของการบำรุงรักษาอีกด้วย



รูปที่ 3.1 โครงสร้างระบบฐานข้อมูลของ SQL Server 7.0

จากรูปที่ 3.1 แสดงให้เห็นถึงโครงสร้างของระบบฐานข้อมูล ซึ่งประกอบด้วยไฟล์หลาย ๆ ไฟล์ ทั้งในส่วนที่เป็นไฟล์ข้อมูล และไฟล์ทรานแซกชันล็อก โดยในแต่ละไฟล์สามารถเก็บอยู่ต่าง พาร หรือแม้แต่ว่าใคร่กันก็ได้ ซึ่งช่วยให้การเก็บข้อมูลสามารถทำได้หลายที่ และไม่ต้องกังวลกับ ขนาดเนื้อที่ว่างของใคร่ที่มีอยู่อย่างจำกัด รวมทั้งช่วยเพิ่มประสิทธิภาพในการเข้าถึงข้อมูลให้ดีขึ้น อีกด้วย

### 3.1.2 การจัดการระบบระบบฐานข้อมูล

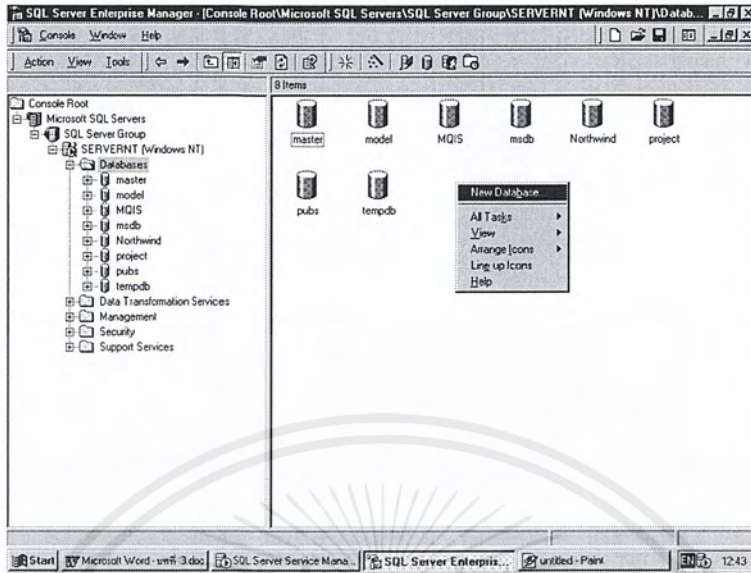
SQL Server 7.0 มีวิธีการจัดการระบบฐานข้อมูล ทั้งการสร้าง, เปลี่ยนแปลง หรือทำลาย รวมถึงการกำหนดคอบช้ันต่างๆ ของฐานข้อมูล โดยใช้เครื่องมือสำหรับบริหารงานอย่าง Enterprise Manager ซึ่งง่ายต่อการใช้งานมากกว่าการใช้คำสั่งของ Transact – SQL

### 3.1.3 การสร้างฐานข้อมูล

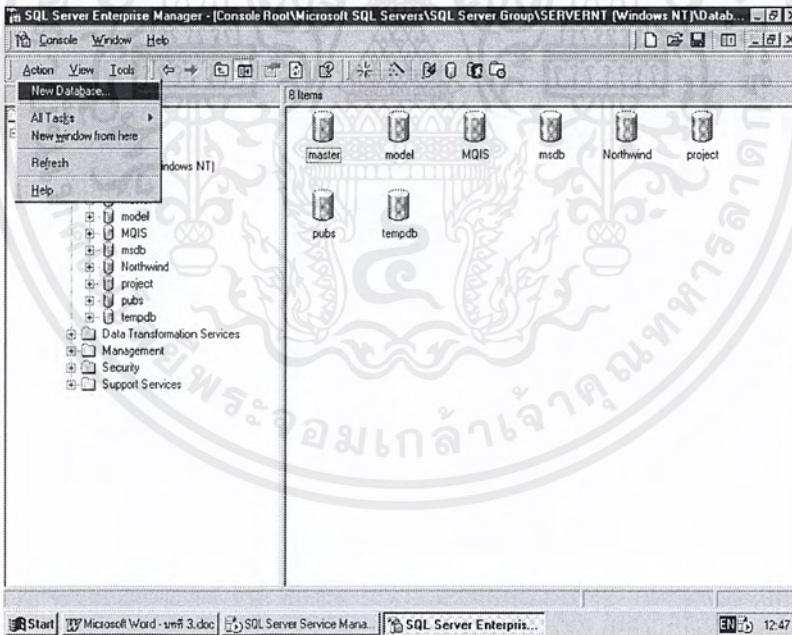
สำหรับการสร้างฐานข้อมูลโดยใช้ Enterprise Manager มีขั้นตอน ดังนี้

- 1) ติดต่อ SQL Server ด้วย Enterprise Manager
- 2) ขยายโพลเดอร์ไปยัง Database
- 3) คลิกขวายังพื้นที่ว่างของเฟรมขวา พร้อมทั้งเลือก New Database ดังรูปที่ 3.2 หรือเลือกที่

เมนู Action > New Database ซึ่งทำให้เด้งเือกสำหรับการสร้างข้อมูลปรากฏขึ้น ดังรูปที่ 3.3




รูปที่ 3.2 การสร้างฐานข้อมูลโดยคลิกขวาบนพื้นที่ว่าง

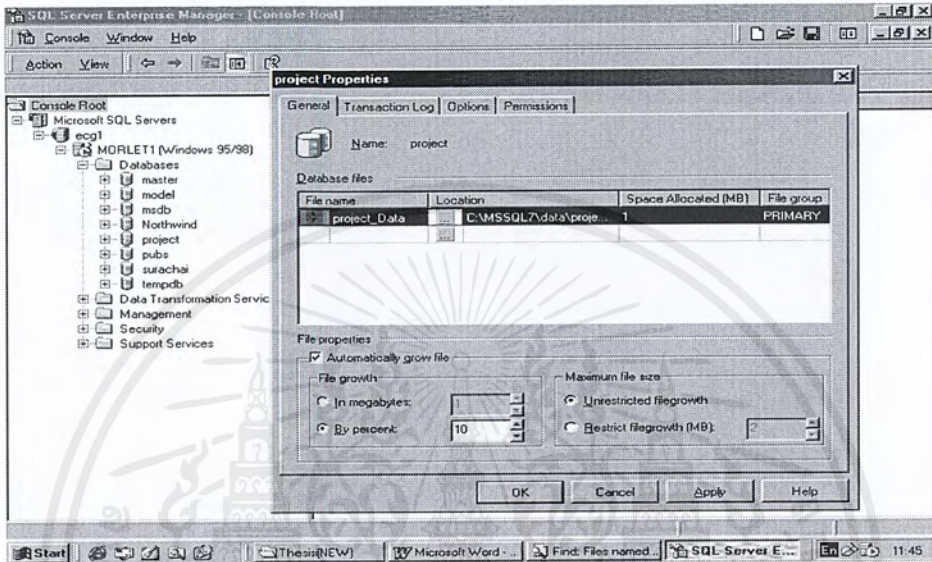


รูปที่ 3.3 การสร้างฐานข้อมูลด้วยเมนู

4) ที่แท็บ General ให้ใส่ชื่อของฐานข้อมูลที่ต้องการสร้างยังช่อง Name: ซึ่งจะมีผลทำให้การตั้งชื่อไฟล์ของฐานข้อมูล ขึ้นต้นด้วยชื่อที่ระบุนี้แล้วตามด้วย \_Data และมีนามสกุลเป็น .MDF โดยอัตโนมัติ หากไม่เหมาะสมก็สามารถเปลี่ยนได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) นอกจากนี้ ในส่วนของ Database files สามารถเปลี่ยนตำแหน่งที่เก็บไฟล์ของฐานข้อมูลได้ ปกติจะอยู่ภายใต้ไดเรกทอรี \MSSQL7\Data โดยการคลิกที่ปุ่ม  ซึ่งจะได้ไดอะล็อกบ็อกซ์สำหรับเลือกเปลี่ยนพาทที่ต้องการ ดังรูปที่ 3.4



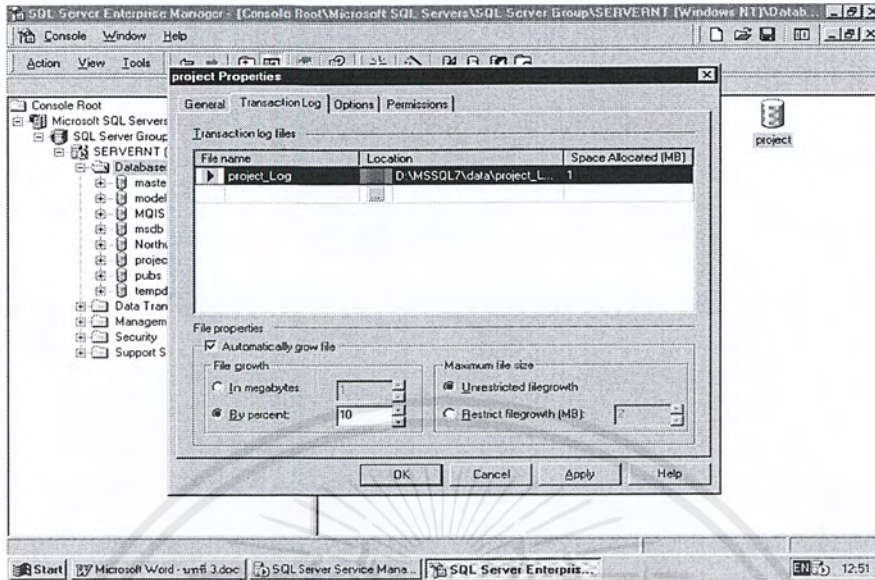
รูปที่ 3.4 การเปลี่ยนพาทของDirectory

6) กำหนดขนาดเริ่มต้นของฐานข้อมูลที่ช่อง Initial size (MB) ซึ่งมีค่าดีฟอลต์เป็น 1 MB ส่วนช่อง File group สำหรับสร้างฐานข้อมูลให้เป็นแบบไฟล์กรุป (Filegroups)

7) ในส่วนของ File properties เช็คบ็อกซ์ที่ Automatically grow file ได้ถูกเลือกไว้โดยดีฟอลต์ ซึ่งทำให้สามารถกำหนดลักษณะการเพิ่มขนาดของฐานข้อมูล เมื่อฐานข้อมูลเกิดเต็มขึ้นมาได้ ซึ่งแบ่งออกเป็น

- **File growth** สำหรับระบุการเพิ่มในแต่ละครั้ง ซึ่งสามารถเลือกได้เป็น In megabytes: คือเพิ่มตามจำนวนที่กำหนดไว้ หน่วยเป็น MB และ By percent เพิ่มขึ้นเป็นจำนวนเปอร์เซ็นต์จากขนาดของฐานข้อมูลเดิม

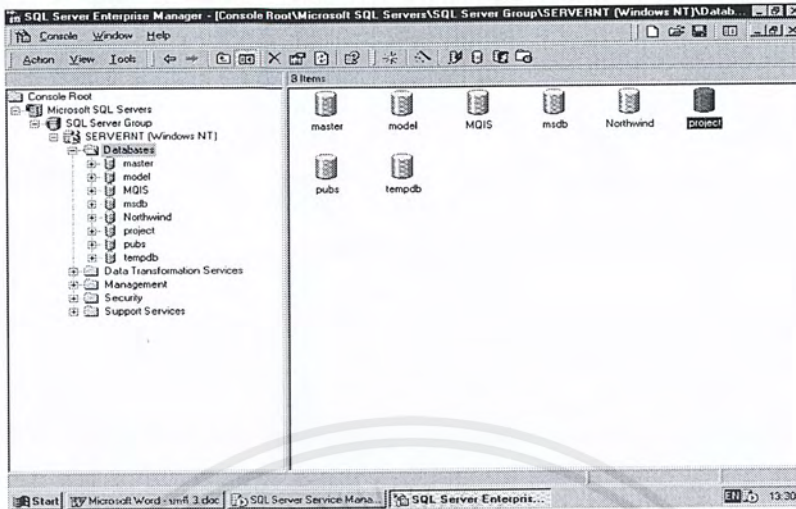
- **Maximum file size** สำหรับกำหนดขนาดของฐานข้อมูลสูงสุดที่สามารถเป็นไปได้ทั้งสิ้น ซึ่งสามารถเลือกได้เป็น Unrestricted filegrowth คือไม่จำกัดขนาด หรือจนกว่าเนื้อที่ว่างบนดิสก์จะเต็ม และ Restricted filegrowth (MB): เพื่อจำกัดขนาดของฐานข้อมูลที่สามารถเป็นไปได้สูงสุด



### รูปที่ 3.5 การเลือกที่เก็บ Transaction Log

8) เลือกที่เก็บ Transaction Log จะพบว่าลักษณะการกำหนดชื่อ และตำแหน่งมีลักษณะคล้ายตามการกำหนดฐานข้อมูลอย่างอัตโนมัติ แสดงดังรูปที่ 3.5 โดยขึ้นต้นด้วยชื่อของฐานข้อมูลตามด้วย \_Log และมีนามสกุลเป็น .LDF หากไม่เหมาะสมสามารถเปลี่ยนชื่อรวมทั้งตำแหน่งของพาร์ตที่เก็บได้ตามต้องการ

9) เมื่อกำหนดข้อมูลเสร็จให้คลิกปุ่ม  เป็นอันเสร็จสิ้นขั้นตอนของการสร้างข้อมูล ซึ่งจะได้สัญลักษณ์รูปทรงกระบอกสีเหลือง แสดงถึงฐานข้อมูลที่เพิ่มสร้างปรากฏขึ้น ดังแสดงในรูปที่ 3.6



รูปที่ 3.6 สัญลักษณ์ของฐานข้อมูลด้วยรูปทรงกระบอกสีเหลือง

### 3.1.4 การออกแบบฐานข้อมูล

ในการออกแบบฐานข้อมูลควรมีการวางแผน และใช้กฎการ Normalization เข้ามาช่วย เพื่อลดความซ้ำซ้อนของข้อมูล มีการกำหนดความสัมพันธ์ระหว่างเอนทิตีของแต่ละตารางให้เหมาะสม และจัดแบ่งประเภทของข้อมูลให้ถูกต้อง หลักการพื้นฐานสำหรับออกแบบฐานข้อมูลมีดังนี้

- 1) รวบรวมข้อมูลที่เกี่ยวข้องกับตารางนั้น ๆ ให้มากที่สุด เช่น ตาราง Customers ควรจะมีข้อมูลที่เกี่ยวข้องกับชื่อ-นามสกุล ที่อยู่ วัน/เดือน/ปี เบอร์โทรศัพท์ ฯลฯ ตาราง Products มีข้อมูลเกี่ยวกับ รหัสสินค้า จำนวน ราคา ฯลฯ
- 2) ทำการ Normalization เพื่อให้เกิดความซ้ำซ้อนน้อยที่สุด เนื่องจากในตารางจะประกอบไปด้วยฟิลด์ข้อมูลหลายฟิลด์
- 3) กำหนดค่าคีย์หลัก (Primary Key) และคีย์นอก (Foreign Key) สำหรับตารางที่เกี่ยวข้อง หรือมีความสัมพันธ์กัน
- 4) ให้กำหนดประเภทของข้อมูลอย่างเหมาะสมกับแต่ละฟิลด์ เพราะประเภทข้อมูลจะมีผลต่อการทำงานโดยตรง เช่น ฟิลด์ Name ควรเป็น Character ฟิลด์ Salary ควรเป็น Numeric ฟิลด์ Order\_date ควรเป็น Datetime

### 3.1.5 การสร้างตาราง

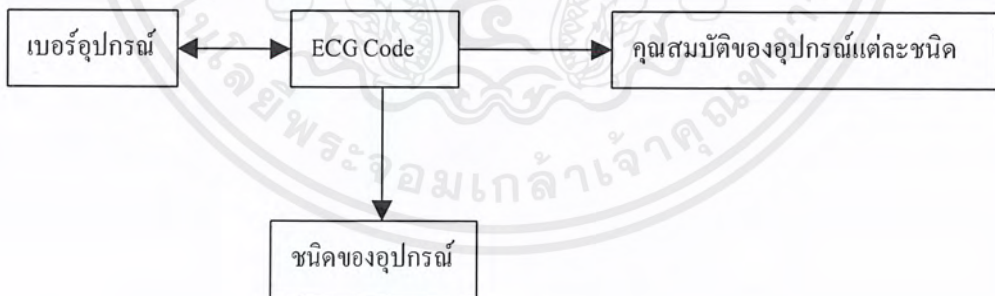
ฐานข้อมูลเชิงสัมพันธ์จะเก็บข้อมูลเอาไว้ในรูปของตาราง (Table) ซึ่งทุกตารางสามารถที่จะเชื่อมโยงข้อมูลระหว่างกันได้ ในการสร้างตารางด้วยภาษา SQL จะใช้ส่วนภาษานิยามข้อมูล (Data Definition Language: DDL) เพื่อกำหนดโครงสร้างของข้อมูล รูปแบบคอลัมน์ชนิดของข้อมูล ก่อนที่จะเริ่มสร้างตารางควรจะออกแบบตารางคร่าวๆ บนแผ่นกระดาษ โดยเขียนชื่อคอลัมน์ ชนิดข้อมูล ความกว้าง กำหนดคีย์หลัก และความสัมพันธ์ระหว่างตารางให้ถูกต้อง

### 3.1.6 การออกแบบโครงสร้างของตาราง

ในการออกแบบโครงสร้างของตารางจะต้องใช้ทฤษฎีและความเป็นจริงร่วมกัน เพื่อให้ได้ตารางข้อมูลที่มีประสิทธิภาพ สามารถเปลี่ยนแปลงแก้ไขได้ในอนาคตและไม่กระทบกระเทือนกับส่วนอื่น ที่สำคัญจะต้องทราบวัตถุประสงค์ในการทำงานว่า ตารางนี้จะใช้เก็บข้อมูลเกี่ยวกับอะไร? เช่น ประวัตินักศึกษา ผลสอบแต่ละวิชา ประวัติพนักงาน ข้อมูลการขายสินค้า เป็นต้น ฯลฯ

จากการศึกษาถึงขอบเขต และหน้าที่ของระบบงาน ทำให้เราได้ทราบถึงโครงสร้างของระบบฐานข้อมูล ดังนั้น โปรแกรมจะมียังประกอบอยู่ 3 ส่วน ได้แก่

- 1) รหัสของอุปกรณ์ (ECG Code)
- 2) เบอร์อุปกรณ์ (Code)
- 3) คุณสมบัติของอุปกรณ์แต่ละชนิด



รูปที่ 3.7 ความสัมพันธ์ระหว่างข้อมูลทั้ง 3 ส่วน

รายละเอียดของข้อมูลทั้ง 3 ส่วนมีดังนี้

### 3.1.7 ข้อมูล EGC

ระบบฐานข้อมูลประกอบด้วยข้อมูลต่างๆ ดังนี้

- 1) รหัสอุปกรณ์ (ECG Code)
- 2) เบอร์อุปกรณ์ (Code)
- 3) ชนิดของอุปกรณ์ (TYPE)

### 3.1.8 ข้อมูลอุปกรณ์

ในแต่ละตารางจะประกอบด้วยคุณสมบัติของอุปกรณ์แต่ละชนิดที่มีความแตกต่างกัน แต่ส่วนประกอบที่สำคัญประกอบด้วย

- 1) รหัสอุปกรณ์ (ECG Code)
- 2) เบอร์อุปกรณ์ (Code)

3) รูปภาพภายนอก (Case) และวงจรภายในของอุปกรณ์ (Circuit or Fig) แต่ละอุปกรณ์มีรายละเอียดปลีกย่อยไม่เหมือนกันสามารถดูได้ที่ภาคผนวก

### 3.1.9 การสร้างตารางใน SQL

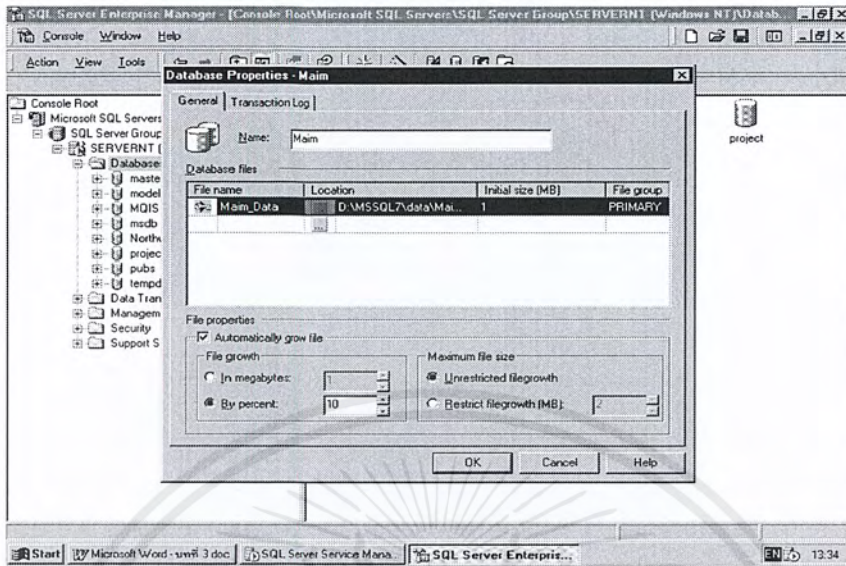
การสร้างตารางสามารถทำได้ 2 วิธีด้วยกัน คือ

- 1) การใช้ SQL Server Enterprise Manager สร้างตาราง
- 2) การใช้สแตทเมนต์คำสั่ง CREATE TABLE สร้างตาราง


### 3.1.10 การใช้ SQL Server Enterprise Manager สร้างตาราง

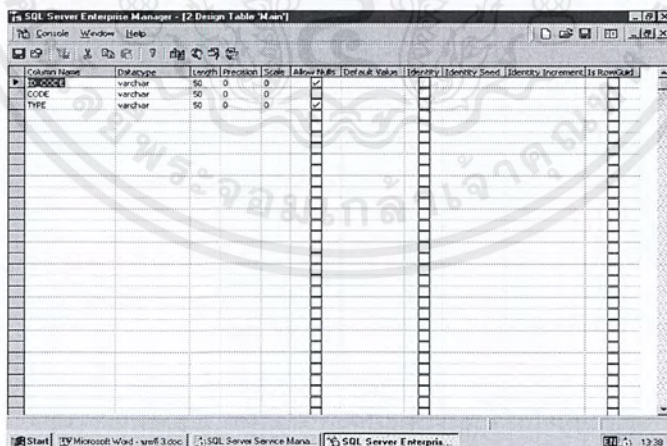
เป็นการสร้างตารางที่ง่ายและรวดเร็ว มีกราฟิกส์ทูลช่วยในการทำงาน ขั้นตอนการสร้างตารางจะยกตัวอย่างมาเพียงบางส่วน ดังนี้

- 1) เข้าสู่ SQL Server Enterprise Manager แล้วคลิกเมาท์ปุ่มขวาตรงฐานข้อมูลที่ต้องการเลือกเมนูคำสั่ง New \ Table ดังรูปที่ 3.8 จะปรากฏกรอบ Choose Name ให้ใส่ชื่อตารางลงไปแล้วคลิกปุ่ม



รูปที่ 3.8 การเข้าสู่ SQL Server Enterprise Manager

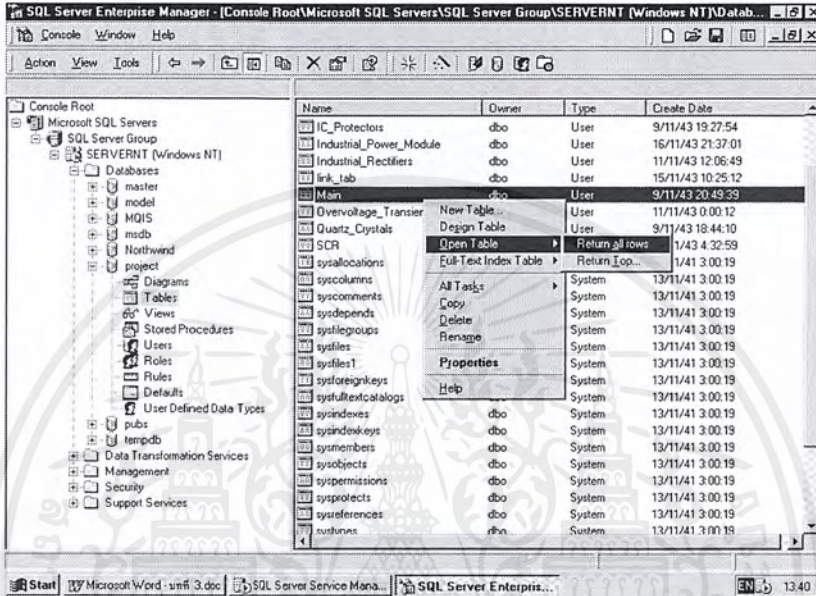
2) จะปรากฏหน้าต่างของ Design Table ซึ่งจะมีชื่อของคอลัมน์ ชนิดข้อมูล ความยาวให้ทำการพิมพ์ชื่อคอลัมน์เป็นภาษาไทย แล้วกำหนดชนิดข้อมูล ขนาดความกว้าง ดังรูปที่ 3.9 เสร็จแล้วคลิก  เพื่อบันทึกโครงสร้างของตารางเอาไว้



รูปที่ 3.9 หน้าต่างของ Design Table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) กลับมาที่ SQL Server Enterprise Manager แล้วเลือกออบเจ็กต์ Tables จะเห็นว่ามียี่ห้อตาราง Main ปรากฏอยู่ด้วย ดังรูปที่ 3.10 ให้คลิกเมาท์ปุ่มขวาตรงตาราง Main แล้วเลือกเมนูคำสั่ง Open Table \ Return all rows



รูปที่ 3. 10 ชื่อตาราง Main ในproject Table

### 3.1.11 การกำหนดค่า DSN

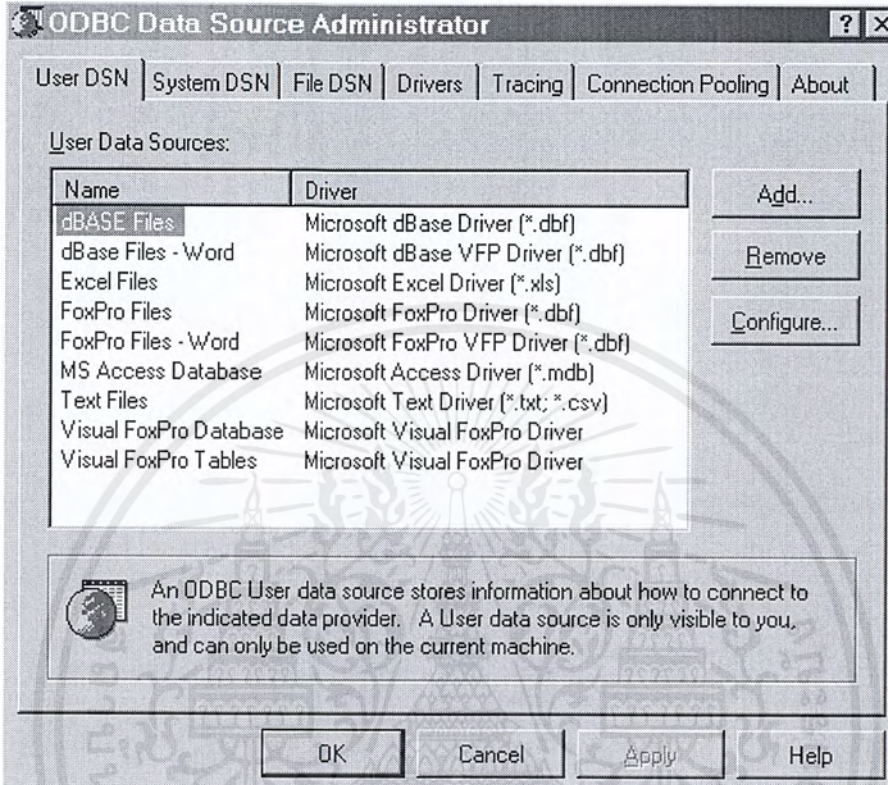
การที่จะใช้ ODBC ในการติดต่อกับฐานข้อมูลได้นั้น จำเป็นต้องมีการกำหนดค่า DSN (Data Source Name) ก่อน เพื่อที่จะระบุว่า ฐานข้อมูลอะไรที่ต้องการใช้งาน เป็นของผู้ผลิตรายใด เป็นต้น ซึ่งเราสามารถเข้าไปกำหนดค่าได้ที่ Control Panel (ในระบบ Windows) ซึ่งจะพบ Icon



ลักษณะนี้อยู่

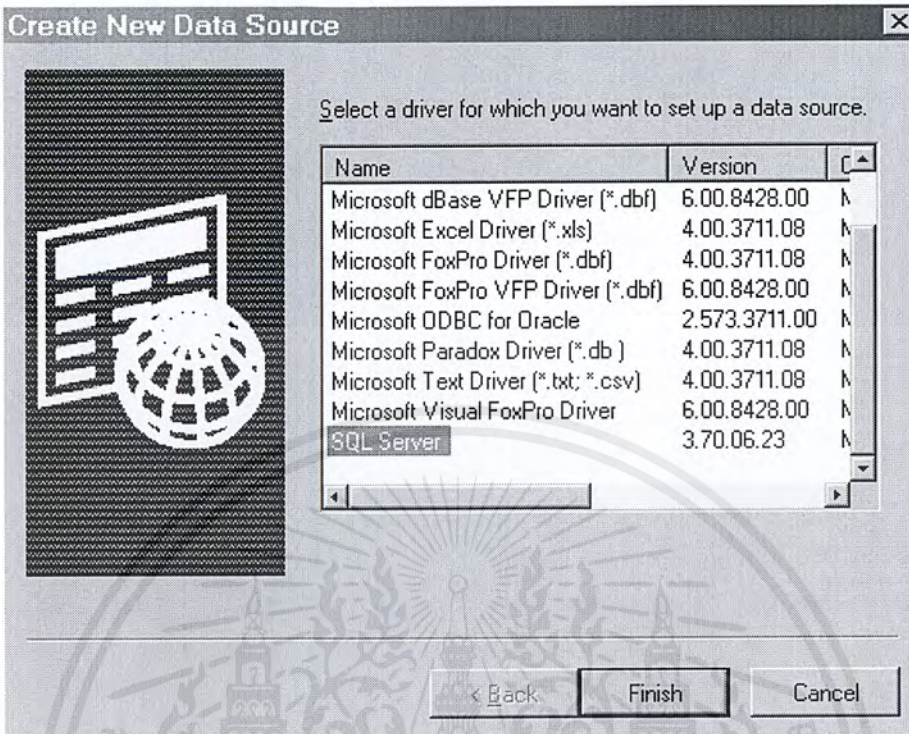
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเข้าไปที่ ODBC แล้วจะปรากฏหน้าต่างดังรูปที่ 3.11



รูปที่ 3.11 หน้าต่าง ODBC Administrator ใน Windows98

- 1) เลือกที่ System DSN หลังจากนั้น คลิกที่ปุ่ม Add เพื่อสร้าง DSN ใหม่ไว้ใช้ในการติดต่อกับฐานข้อมูลที่ต้องการแล้วจะปรากฏหน้าต่าง Create New Data Source ดังรูปที่ 3.12



รูปที่ 3.12 หน้าต่าง Create New Data Source ของ ODBC

2) ให้เลือกชนิดของฐานข้อมูลที่ต้องการติดต่อ ซึ่ง ODBC แต่ถ้าไม่มีในรายการให้หา Driver นั้นมาติดตั้งเพื่อเติม โดยในที่นี้จะเลือกติดต่อกับฐานข้อมูลประเภท SQL Server แล้วคลิกที่ปุ่ม Finish

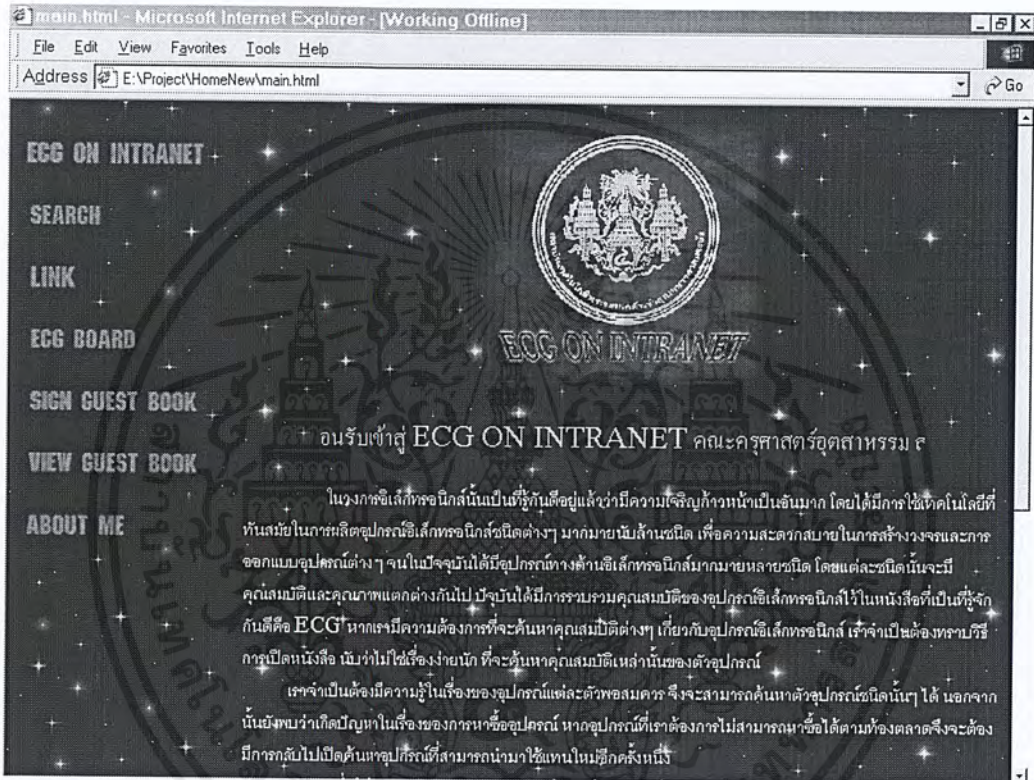
3) จากนั้นจะปรากฏหน้าต่าง ODBC ของ SQL Server ขึ้นมาให้เติมชื่อ Data Source Name ที่ต้องการ

4) ให้คลิกที่ปุ่ม Select.. ซึ่งเป็นการเลือกฐานข้อมูลที่ต้องการติดต่อ ซึ่งเมื่อกำหนดฐานข้อมูลเสร็จเรียบร้อยแล้วจะปรากฏชื่อที่เราตั้งไว้ใน ODBC

การกำหนด ODBC ในแต่ละฐานข้อมูลนั้นจะไม่เหมือนกันแล้วแต่ Driver ของแต่ละฐานข้อมูลว่าจะกำหนดมาเป็นอย่างไร ถ้าเป็นพวก Database Server อาจจะต้องกำหนด ชื่อ Server, Username, Password เป็นต้น

### 3.2 การออกแบบและการสร้างโฮมเพจ

การออกแบบหน้าโฮมเพจนั้น เราได้ทำการศึกษาภาษา HTML, JAVA SCRIPT และรวมถึงโปรแกรมสำเร็จรูปอย่าง Macromedia Dreamweaver3 ด้วย ซึ่งได้ทำการออกแบบหน้าจอหลักโดยแบ่งออกเป็น 2 ส่วนด้วยกัน คือ ส่วนของเมนูและส่วนของการแสดงผล



รูปที่ 3.13 หน้าจอหลัก

โดยในส่วนของเมนูนั้นเราได้ทำการแบ่งออกเป็นหัวข้อต่างๆ รวมทั้งสิ้น 7 หัวข้อด้วยกัน ดังนี้คือ

- 1) ECG ON INTRANET เป็นการกล่าวถึงประวัติ ความสำคัญ และประโยชน์ที่ได้รับจากโปรแกรมนี้
- 2) SEARCH เป็นการนำท่านเข้าสู่ระบบการค้นหาข้อมูล โดยโปรแกรม
- 3) LINK เป็นส่วนของหน้ารวมลิงค์ต่างๆ เช่น บทความหรือเว็บต่างๆ ที่เกี่ยวข้องกับ การเขียนโฮมเพจ อินเทอร์เน็ต คอมพิวเตอร์ และรวมถึงลิงค์ต่างๆ ไป เป็นต้น
- 4) ECG BOARD คือหน้าของการตั้งกระทู้คำถามและข้อสงสัยต่างๆ

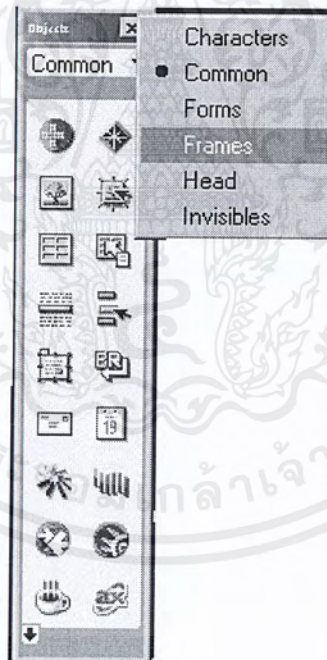
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5) SIGN GUEST BOOK คือหน้าที่สามารถให้ผู้เยี่ยมชมได้ลงชื่อแนะนำติชม
- 6) VIEW GUEST BOOK คือหน้าแสดงข้อความต่างๆ ที่ผู้เยี่ยมชมแต่ละท่านได้ให้ข้อวิจารณ์ ข้อเสนอแนะต่างๆ ไว้
- 7) ABOUT ME เป็นหน้าของการแนะนำตัว ประวัติโดยสังเขปของผู้จัดทำแต่ละท่าน

### 3.2.1 การสร้างโฮมเพจโดย Macromedia Dreamweaver3

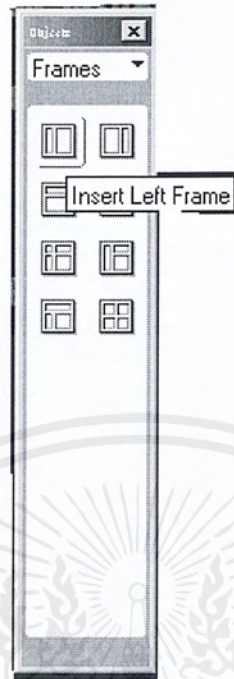
ในการสร้างหน้าต่างๆ ของโฮมเพจนั้น เราสามารถทำได้ทั้งทางการเขียนด้วยภาษา HTML โดยใช้โปรแกรม Editor ทั่วๆ ไปและการใช้โปรแกรมสำเร็จรูปอย่าง Macromedia Dreamweaver3 ซึ่งในการใช้โปรแกรมสำเร็จรูปทำให้สามารถลดระยะเวลาในการดำเนินการลงไปได้เป็นอย่างมาก เราจึงจะขอยกตัวอย่างวิธีการสร้าง โดยการใช้โปรแกรมสำเร็จรูป Macromedia Deamweaver3

จากรูปที่ 3.13 เราจะเห็นว่าหน้าจอกถูกแบ่งออกเป็น 2 ส่วนคือส่วนของเมนูและส่วนแสดงผลซึ่งการแบ่งหน้าจอหรือการแบ่งเฟรมนี้เราสามารถทำได้โดย



รูปที่ 3.14 การเลือก Object

- 1) เข้าไปที่เมนู Object และเลือก Frames เพื่อเป็นการเข้าสู่ฟังก์ชันการสร้างหรือการแบ่งหน้าจอของโฮมเพจ



รูปที่ 3.15 การเลือกรูปแบบเฟรม

2) เมื่อเข้าสู่เมนูเฟรมเราก็สามารถที่จะเลือกรูปแบบของเฟรมตามที่ต้องการนำไปใช้งานได้

ในการสร้างฟอร์มเพื่อรับส่งข้อมูลต่างๆ นั้นมีอยู่ด้วยกันหลายรูปแบบ ซึ่งใน Macromedia Dreamweaver3 ก็มีการให้เลือกรูปแบบฟอร์มได้เช่นกัน โดยมีให้เลือก เช่น Text Box, Select Box, Button, Checkbox และ Radio Button เป็นต้น สำหรับการสร้างฟอร์มนี้ก็เช่นกัน เราจะเข้าไปที่ Object เลือกที่ Forms จากนั้นจึงเลือกรูปแบบของฟอร์มตามที่เราต้องการและกำหนดค่าให้กับฟอร์มที่เราได้เลือกมาด้วย



อย่างไรก็ตามเมื่อทำการสร้างแต่ละหน้าเสร็จแล้วเรายังคงต้องมาทำการตรวจสอบความถูกต้องจาก HTML ໄ໊ດໄດ້ด้วย เพื่อทำการแก้ไขในส่วนที่โปรแกรมสำเร็จรูปจัดการให้ไม่ได้ หรือในส่วนที่เราต้องการเพิ่มเติมให้มีประสิทธิภาพมากยิ่งขึ้น

### 3.3 การเชื่อมต่อฐานข้อมูลด้วย PHP

การทำงานของ PHP มีลักษณะการทำงานคือ เมื่อต้องการใช้งานคำสั่งของ PHP ก็จะใช้แท็กชุดคำสั่งของ PHP เข้าไปในชุดคำสั่งการทำงานของ HTML โดยมีเครื่องหมายเปิด-ปิดชุดคำสั่งคือ `<? ชุดคำสั่ง PHP ?>` การทำงานในส่วนของ PHP จะถูกใช้ในการติดต่อกับฐานข้อมูล โดยที่แทรกไปไว้ในชุดคำสั่งที่เป็น HTML เพื่อใช้ในการควบคุมการทำงานและติดต่อกับผู้ใช้ โปรแกรม ECG การใช้งาน PHP นี้จะมีการใช้ร่วมกับภาษา SQL ด้วยเพื่อติดต่อกับฐานข้อมูล

วิธีการทำงานของ PHP นี้จะต้องมีการกำหนด DSN หรือฐานข้อมูล เพื่อให้ทราบว่าต้องไปติดต่อกับฐานข้อมูลตัวใด การติดต่อกับฐานข้อมูลเพื่อนำข้อมูลขึ้นมาแสดงจะใช้ชุดคำสั่งภาษา SQL เพื่อใช้งานตัวอย่างเช่น `sql = "select * from project "`; ซึ่งในที่นี้ project ก็คือชื่อฐานข้อมูลที่ต้องการนำมาแสดง อีกทั้งการค้นหา ข้อมูล ECG มีการทำงานที่ต้องการการค้นหาเป็นลำดับ จึงทำให้ต้องใช้คำสั่งในการเรียงข้อมูลและคำสั่ง เพื่อทำงานเป็นรูป ในการนำตารางมาแสดงก็จะทำการสร้างตารางไว้เป็น form โดยให้มีจำนวนคอลัมน์เท่ากับจำนวนฟิลด์ภายในตารางที่ต้องการนำมาแสดงผล เมื่อนำไปเปรียบเทียบและค้นหาข้อมูลแล้ว ก็จะนำมาแสดงให้ผู้ใช้ดูโดยผ่านทางหน้าจอ Homepage

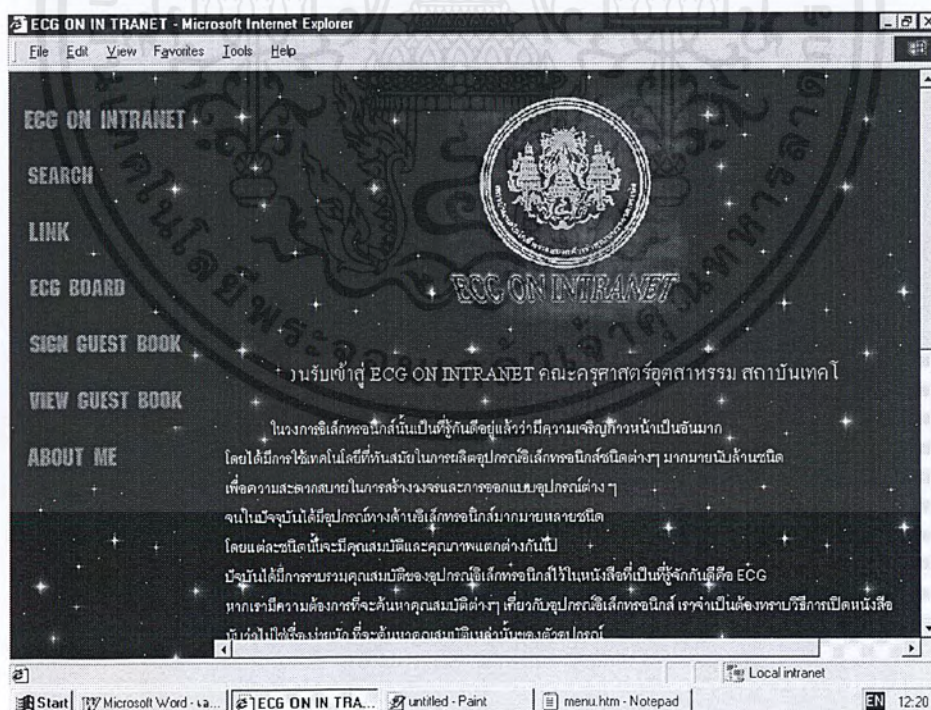
การติดต่อกับ ODBC จะต้องไปกำหนดให้ ODBC DATA SOURCE ทราบว่าเราต้องการใช้ ฐานข้อมูลตัวใดอีกทั้งยังต้องไปกำหนดให้ทราบว่าเราจะใช้ PHP ในการติดต่ออีกด้วย

## บทที่ 4

### การทดลองและผลการทดลอง

จากการทดลองใช้งาน โปรแกรม ECG on Intranet ผลที่ได้คือจะได้หน้าจอแสดงจากผลทางด้านซ้ายมือเรียงกันมา คือ

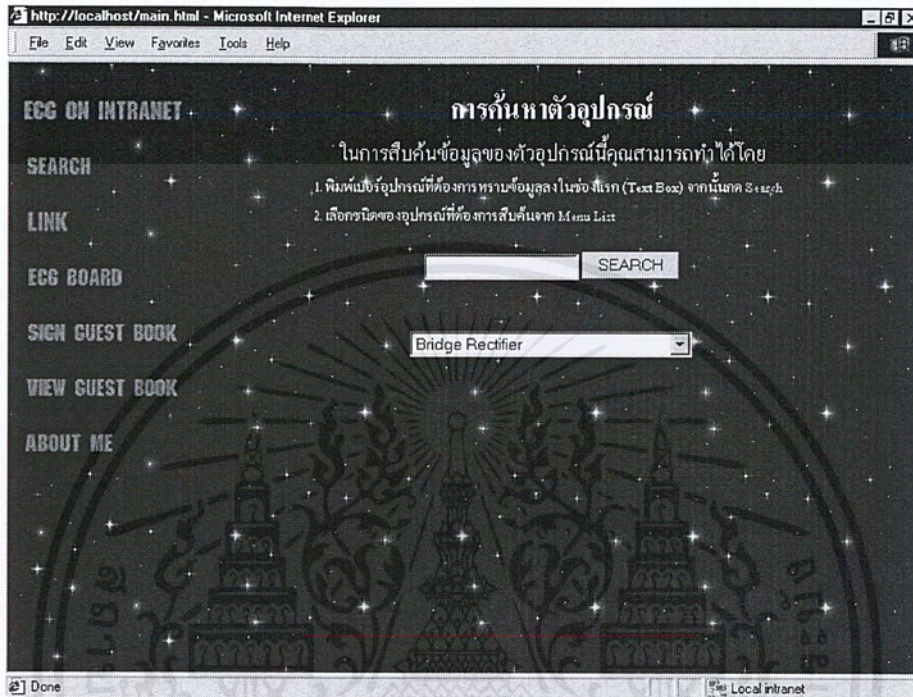
- 1) ECG ON INTRANET
- 2) SEARCH
- 3) LINK
- 4) ECG BOARD
- 5) SIGN GUEST BOOK
- 6) VIEW GUEST BOOK
- 7) ABOUT ME



รูปที่ 4.1 เมนูการทำงานของโปรแกรม ECG on Intranet

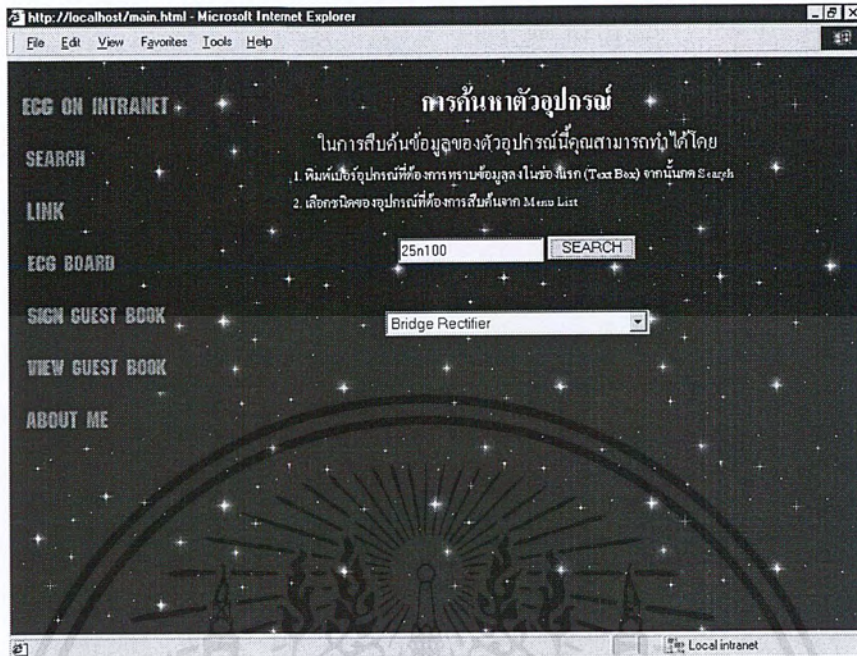
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวเลือก ECG ON INTRANET เป็นการกล่าวถึง ประวัติความเป็นมาและเหตุผลที่ต้องมีการจัดทำโปรแกรมนี้ และประโยชน์ของ โปรแกรมนี้รวมทั้งแนวทางการพัฒนาโปรแกรม



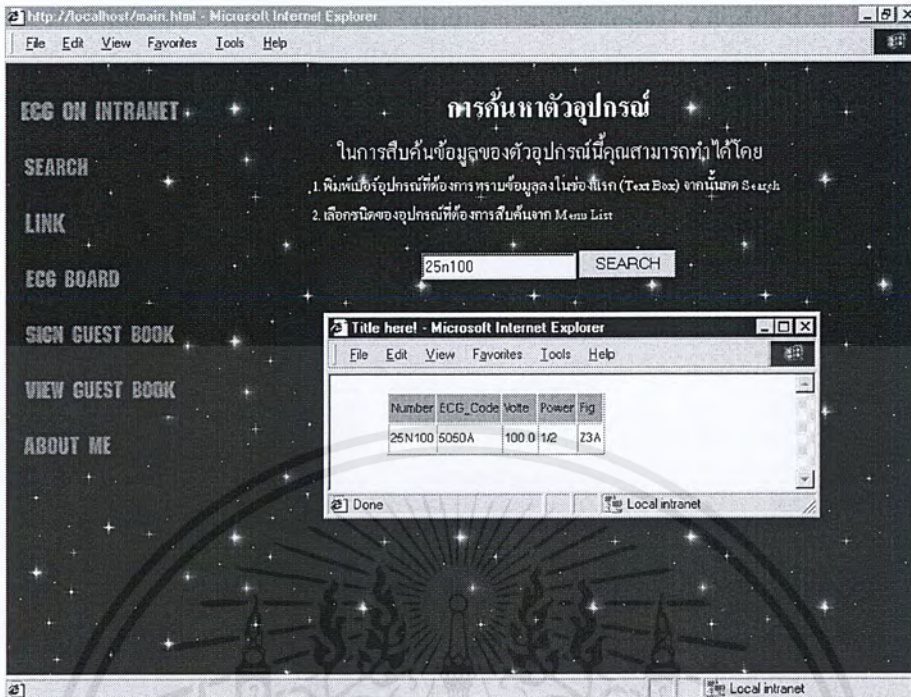
รูปที่ 4.2 เมนู SEARCH (ค้นหา)

จากตัวเลือก SEARCH เป็นเมนูบอกถึงการค้นหา โดยแบ่งการค้นหาเป็น 2 รูปแบบ คือ รูปแบบที่ 1 แสดงการค้นหาโดยใช้หมายเลขอุปกรณ์ รูปแบบที่ 2 แสดงการค้นหาโดยรับคุณสมบัติของอุปกรณ์

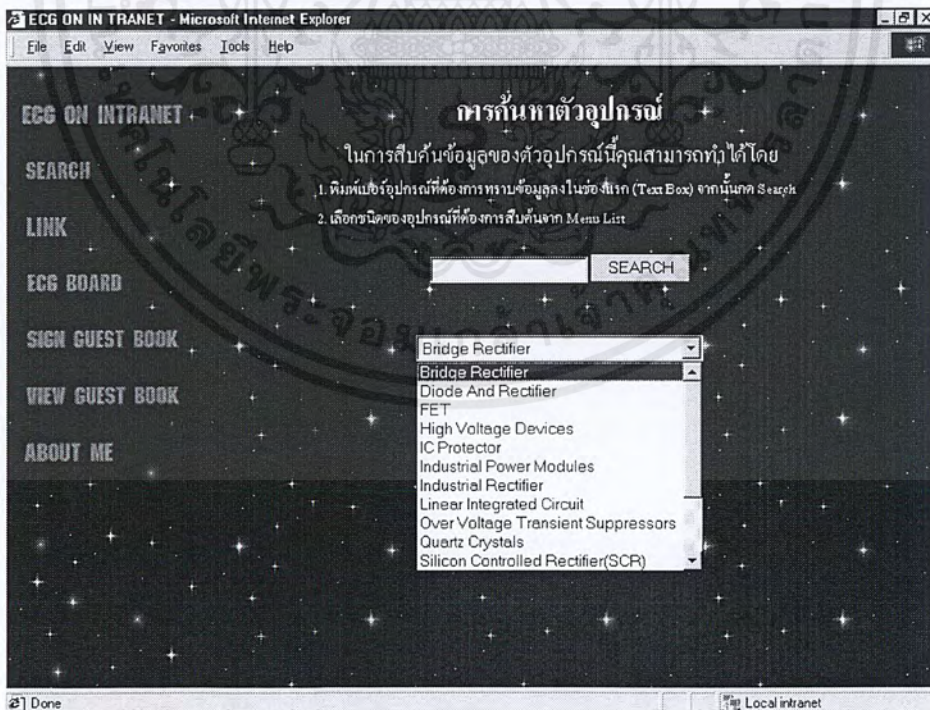


รูปที่ 4.3 การค้นหาแบบป้อนหมายเลขอุปกรณ์

รูปแบบที่ 1 แสดงการค้นหาโดยใช้หมายเลขอุปกรณ์ การทำงานของเมนูนี้เริ่มต้นให้ผู้ใช้ป้อนหมายเลขของอุปกรณ์เข้าไปในช่องที่กำหนดให้ เมื่อได้หมายเลขของอุปกรณ์ที่ต้องการแล้ว ก็กดปุ่ม SEARCH ที่อยู่ทางด้านหลัง ผลที่ได้คือ จะแสดงตารางค่าคุณสมบัติทั้งหมดและรูปของอุปกรณ์ให้ผู้ใช้ดู โดยรูปจะดูได้เมื่อกด LINK ไปที่รูปภาพ



รูปที่ 4.4 ผลการทดลองการป้อนหมายเลขอุปกรณ์



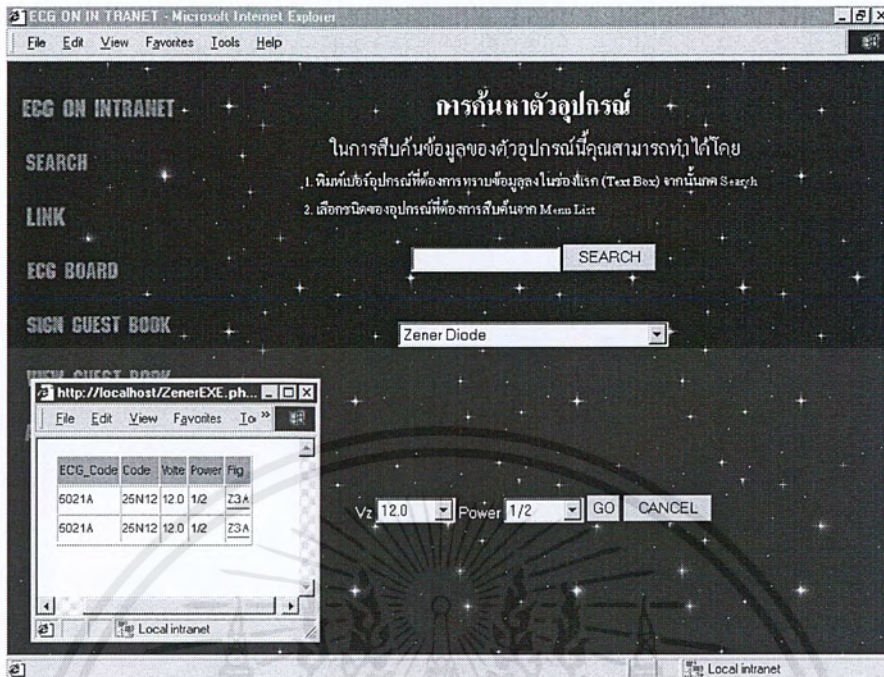
รูปที่ 4.5 หน้าจอการค้นหาโดยใช้คุณสมบัติเลือกจากไดอะล็อกบ็อกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบที่ 2 แสดงการค้นหาโดยรับคุณสมบัติของอุปกรณ์แล้วเข้าไปทำการค้นหา ผู้ใช้ป้อนคุณสมบัติจากไดอะล็อกบ็อกซ์ที่กำหนดให้ หลังจากนั้นจะนำข้อมูลที่ได้มาไปเปรียบเทียบเพื่อหาหมายเลขอุปกรณ์ที่ต้องการและทำการหาอุปกรณ์ที่สามารถใช้ทดแทนกันได้ด้วย ตัวอย่างของคุณสมบัติของอุปกรณ์ คือ

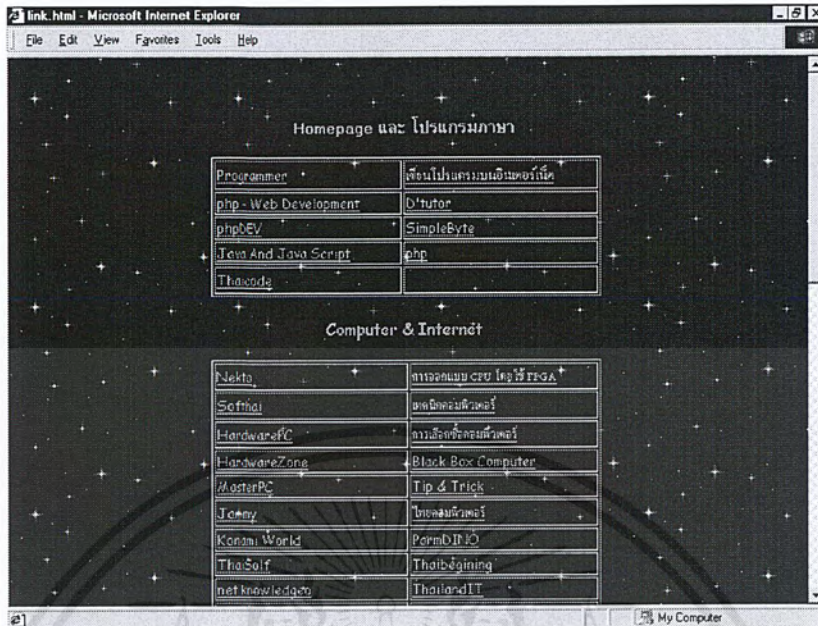
Bridge Rectifier	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	PRV, IO, PeakRSC
Diode And Rectifier	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	TYPE, IOMAX, VF
FET	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	GFS, VGS, IDSS
Industrial Rectifier	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	PRV, IO
Quartz Crystals	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	MHZ, FT
SCR	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	VDRM, RMS, AV
Transistor	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	Type(PNP หรือ NPN), HFE, FT
TRIACS	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	VRRM, FT
Zener Diode	คุณสมบัติที่ผู้ใช้ป้อนได้คือ	VZ, POWER

ตามคุณสมบัติเหล่านี้ที่ผู้ใช้สามารถ เลือกให้มีตัวใดตัวหนึ่งหรือทุกตัวก็ได้แล้วแต่ความต้องการของผู้ใช้ดังมีตัวอย่างดังรูป ดังนี้



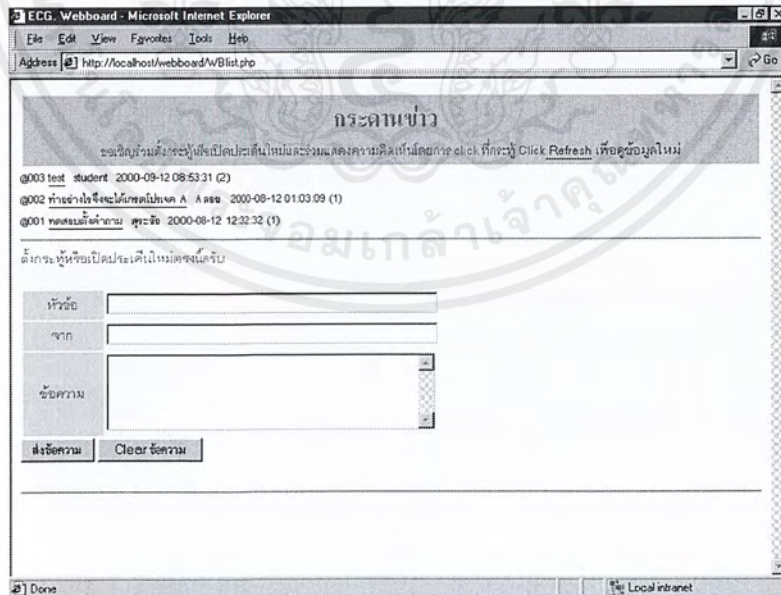
รูปที่ 4.6 ผลการทดลองการค้นหาคิวด้วยคุณสมบัติของอุปกรณ์

จากตัวเลือก LINK เป็นเมนูบอกถึงการ link กับหน้าโฮมเพจอื่น ที่กำหนดไว้โดยโฮมเพจเหล่านี้มีทั้งที่เกี่ยวข้องกับโปรแกรมนี้โดยตรงและทางอ้อม รวมไปถึงที่ไม่เกี่ยวข้องแต่ผู้ใช้สามารถเข้าไปศึกษาเรื่องต่างๆ ได้โดยสะดวก



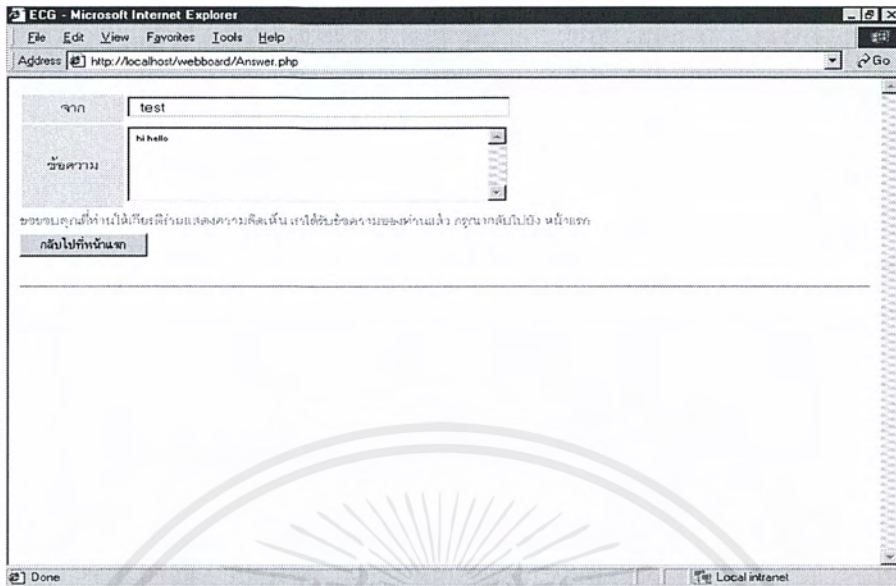
รูปที่ 4.7 การทำงานของเมนู LINK

จากตัวเลือก ECG BOARD เป็นเมนูบอกถึงการเรียกใช้งาน WEBBOARD โดยรองรับชื่อนามสกุล, E-mail และข้อความที่ผู้ใช้ต้องการสื่อสาร ไปยังผู้อื่น



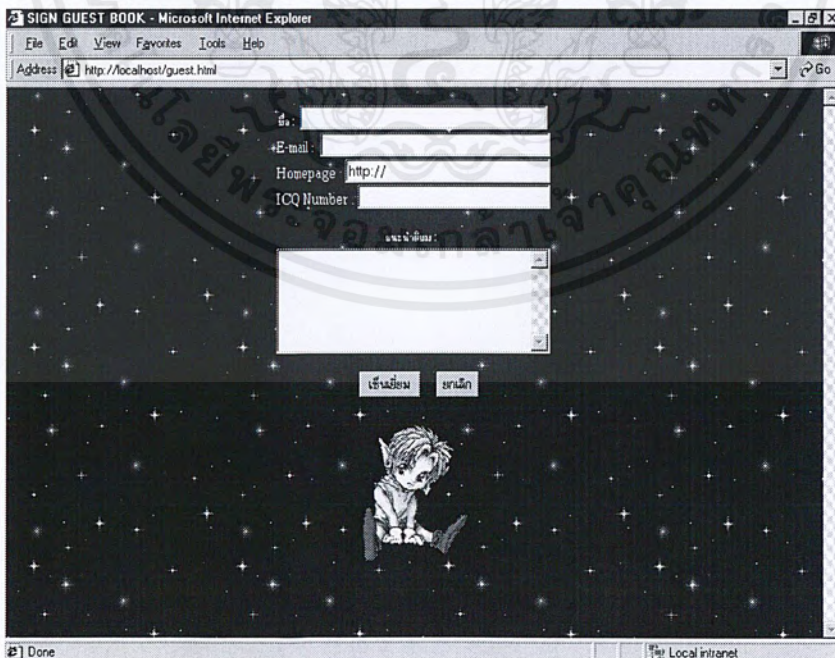
รูปที่ 4.8 การทำงานของเว็บบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



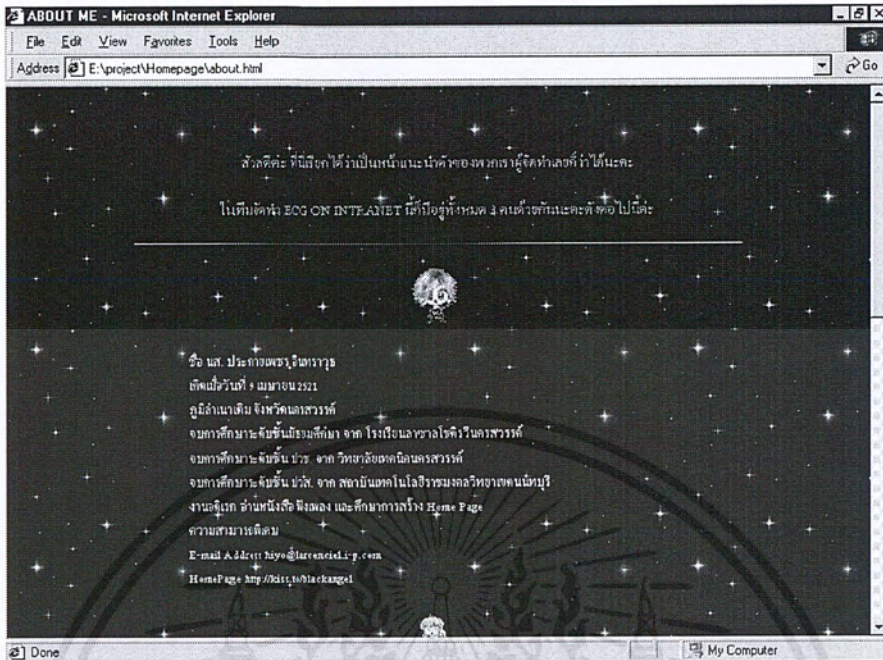
รูปที่ 4.9 การทำงานของเว็บบอร์ดหลังได้รับข้อความ

จากตัวเลือก SIGN GUEST BOOK เป็นเมนูบอกถึงการเรียกใช้งาน GUEST BOOK โดยรองรับชื่อ-นามสกุล, E-mail และข้อความที่ผู้ใช้ต้องการสื่อสารไปยังผู้จัดทำโปรแกรมและเป็นการแสดงความคิดเห็นเกี่ยวกับโปรแกรมนี้



รูปที่ 4.10 การทำงานของเมนู GUEST BOOK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 การทำงานของเมนู ABOUT ME

จากตัวเลือก ABOUT ME เป็นเมนูที่แสดงให้ทราบเกี่ยวกับรายละเอียดของผู้จัดทำโปรแกรมนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป ปัญหา แนวทางแก้ไขและการพัฒนา

#### 5.1 บทสรุป

โปรแกรม ECG เป็นโปรแกรมที่สร้างขึ้นมาเพื่อประโยชน์ในการค้นหาข้อมูลอุปกรณ์อิเล็กทรอนิกส์ สะดวกในการค้นหา ในการออกแบบโปรแกรมจะออกแบบเพื่อให้ผู้ใช้ได้รับความสะดวก รวดเร็วสูงสุดในการทำงานของโปรแกรม เนื่องจากการค้นหาด้วยหนังสือมีปัญหามากมายจากการที่ผู้ใช้ไม่มีความชำนาญในการใช้หนังสือ ดังนั้นแม้ว่าผู้ใช้โปรแกรมจะไม่เคยใช้หนังสือ ECG เลยก็สามารถใช้โปรแกรมนี้เพื่อหาข้อมูลได้เช่นเดียวกับการใช้หนังสือ ECG

#### 5.2 ปัญหาและแนวทางแก้ไข

1) การเขียนโปรแกรม PHP ไม่สามารถทำได้ทุกที่ เนื่องจากต้องติดตั้งโปรแกรม Personal Web Server และต้องใช้ MS SQL เป็น Database Server ที่ทำงานบนระบบปฏิบัติการ Windows NT ดังนั้นจึงต้องมีเครื่องที่ใช้ประจำไม่สามารถใช้เครื่องได้ทั่วไป และเครื่องที่มีอยู่มีความสามารถต่ำ

2) การใช้โปรแกรม SQL SERVER เพื่อจัดการระบบฐานข้อมูล สามารถทำการปรับปรุงแก้ไขทำได้ยากหากมีการใส่ข้อมูลผิดพลาด

แนวทางแก้ไข ทำการป้อนข้อมูลใน MS Access แล้วทำการ Import เข้ามาใน SQL SERVER แทน

3) ข้อมูลที่นำมาสร้างฐานข้อมูลนั้น มีความแตกต่างกันมาก ความสัมพันธ์ต่อกันของข้อมูลนั้นมีต่ำ จึงทำให้ยากต่อการออกแบบ และหลังจากที่ทำการออกแบบฐานข้อมูลแล้ว เมื่อสร้างเป็นตารางเพื่อใช้งานจริง พบปัญหาที่ทำให้ต้องทำการแก้ไขหลายต่อหลายครั้ง

แนวทางแก้ไข ทำได้โดยสร้างตารางขึ้นใหม่หากข้อมูลมีความแตกต่างกันมาก และรวมตารางกันสำหรับข้อมูลที่ไม่แตกต่างกันมาก

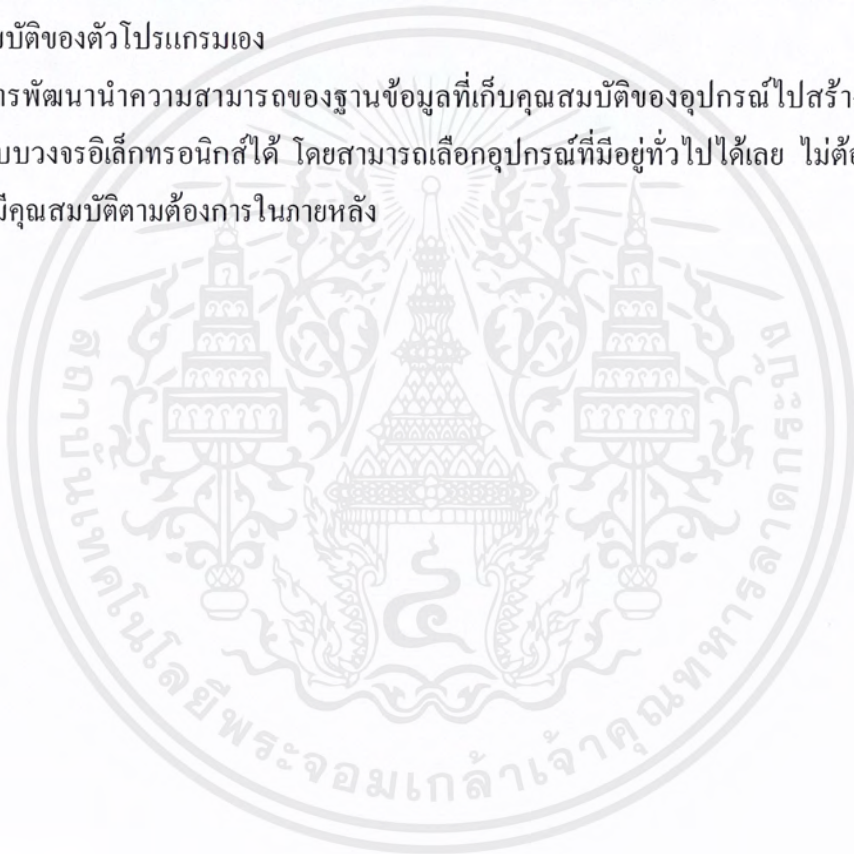
4) ไม่สามารถนำรูปวงจรหรือลักษณะภายนอกของอุปกรณ์ไปแสดงผลได้ เนื่องจากไม่มีตัวแปลงไฟล์รูปออกมา แม้จะทดลองนำรูปไปไว้ใน MS Access แล้วทำการ Import เข้ามาใน SQL SERVER แล้วก็ไม่สามารถอ่านไฟล์รูปได้

แนวทางแก้ไข ใช้วิธีใส่ชื่อไฟล์ในตารางแล้วเขียนโปรแกรมเข้าไปอ่านชื่อไฟล์ เพื่ออ่านไฟล์แล้วนำไปแสดงผลอีกที

### 5.3 แนวทางการพัฒนา

การพัฒนาของโปรแกรมนี้อาจทำได้โดย เมื่อมีการเปลี่ยนแปลงฐานข้อมูลสามารถทำได้จากทางเครือข่ายได้ เนื่องจากการติดต่อข้อมูลในขณะนี้ จะเป็นลักษณะของการทำงานโดยเน็ตเวิร์กเครือข่าย และรวมถึงตัวโปรแกรม SQL SERVER นั้น ก็มีความปลอดภัยในการติดต่อกับฐานข้อมูลด้วยคุณสมบัติของตัวโปรแกรมเอง

การพัฒนานำความสามารถของฐานข้อมูลที่เก็บคุณสมบัติของอุปกรณ์ไปสร้างโปรแกรมเพื่อออกแบบวงจรอิเล็กทรอนิกส์ได้ โดยสามารถเลือกอุปกรณ์ที่มีอยู่ทั่วไปได้เลย ไม่ต้องไปค้นหาอุปกรณ์ที่มีคุณสมบัติตามต้องการในภายหลัง





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?&#x26;dsn="ecgproject";
echo "<body bgcolor=\&#xFF6699\">";
echo "<blockquote><table border=1 cellspacing=0><tr
bgcolor=SKYBLUE>";
&#x26;connect=odbc_connect(&#x26;dsn,"","");
if((&#x26;VOLTE=="none")&#x26;&(&#x26;POWER=="none")){
&#x26;sql="select * from Z_diode";
}elseif((&#x26;VOLTE=="none")&#x26;&(&#x26;POWER !="none")){
&#x26;sql="select * from Z_diode where power='&#x26;.POWER.'";
}elseif((&#x26;VOLTE!="none")&#x26;&(&#x26;POWER =="none")){
&#x26;sql="select * from Z_diode where volte='&#x26;.VOLTE.'";
}else{&#x26;sql="select * from Z_diode where volte='&#x26;.VOLTE.'
and power='&#x26;.POWER.'";}
&#x26;rn=odbc_exec(&#x26;connect,&#x26;sql);
&#x26;sql3="select Code from main where ECG_Code='&#x26;.odbc_result
(&#x26;rn,1).'";
&#x26;rn3=odbc_exec(&#x26;connect,&#x26;sql3);
echo "<body bgcolor=\&#xFF6699\">";
echo "<blockquote><table border=1 cellspacing=0><tr
bgcolor=\&#xFF6699\">";echo "<blockquote>
<table border=1 cellspacing=0><tr bgcolor=\&#xFF6699\">";
echo "<td>",&#x26;odbc_field_name(&#x26;rn,1),"</td>";
echo "<td>",&#x26;odbc_field_name(&#x26;rn3,1),"</td>";
echo "<td>",&#x26;odbc_field_name(&#x26;rn,2),"</td>";
echo "<td>",&#x26;odbc_field_name(&#x26;rn,3),"</td>";
echo "<td>",&#x26;odbc_field_name(&#x26;rn,4),"</td>";
do{
//for serial number
&#x26;sql3="select Code from main where ECG_Code='&#x26;.odbc_result
(&#x26;rn,1).'";
&#x26;rn3=odbc_exec(&#x26;connect,&#x26;sql3);
//echo &#x26;sql3;
echo"<tr>";
echo "<td>",&#x26;odbc_result(&#x26;rn,1),"</td>";
// show serial number
echo "<td>",&#x26;odbc_result(&#x26;rn3,1),"</td>";
echo "<td>",&#x26;odbc_result(&#x26;rn,2),"</td>";
echo "<td>",&#x26;odbc_result(&#x26;rn,3),"</td>";
&#x26;text="<a href='&#x26;/04_Zener_Diode/&#x26;.odbc_result(&#x26;rn,4).'.jpg'>";
echo "<td>",&#x26;text,&#x26;odbc_result(&#x26;rn,4),"</a>",&#x26;,&#x26;,&#x26;";
echo"</tr>";
}
while(odbc_fetch_row(&#x26;rn));
&#x26;bgc=(&#x26;bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=&#x26;bgc>";
echo "</table>";?>

```

### รูปที่ 1 โปรแกรม Zener Diode

```

<?&#x26;dsn=&#x26;ecgproject';
echo <body bgcolor=\&#xFF6699\">";
echo <blockquote><table border=1 cellspacing=0><tr

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

bgcolor=SKYBLUE>;
$connect=odbc_connect($dsn, '', '');
if (($VRRM==none) && ($IT_RRM==none)) {
  $sql=select * from triacs;
}elseif (($VRRM==none) && ($IT_RRM !=none)) {
  $sql=select * from triacs where IT_RRM=' '.$IT_RRM.' ';
}elseif (($VRRM!=none) && ($IT_RRM ==none)) {
  $sql=select * from triacs where V_RRM=' '.$VRRM.' ';
}else{$sql=select * from triacs where V_RRM=' '.$VRRM.' ' And
IT_RRM=' '.$IT_RRM.' ';}
$rn=odbc_exec($connect,$sql);
$sql3=select Code from main where ECG_Code=' '.odbc_result
($rn,1).' ';
$rn3=odbc_exec($connect,$sql3);
echo <body bgcolor=#FFFFFF>;
echo <blockquote><table border=1 cellspacing=0><tr
bgcolor=#FF6699>;echo <blockquote>
  <table border=1 cellspacing=0><tr bgcolor=#FF6699>;
//
echo <td>,odbc_field_name($rn,1), </td>;
echo <td>,odbc_field_name($rn3,1), </td>;
echo <td>,odbc_field_name($rn,2), </td>;
echo <td>,odbc_field_name($rn,3), </td>;
echo <td>,odbc_field_name($rn,4), </td>;
echo <td>,odbc_field_name($rn,5), </td>;
echo <td>,odbc_field_name($rn,6), </td>;
echo <td>,odbc_field_name($rn,7), </td>;
echo <td>,odbc_field_name($rn,8), </td>;
echo <td>,odbc_field_name($rn,9), </td>;
echo <td>,odbc_field_name($rn,10), </td>;
echo <td>,odbc_field_name($rn,11), </td>;
echo <td>,odbc_field_name($rn,12), </td>;
echo <td>,odbc_field_name($rn,13), </td>;
echo <td>,odbc_field_name($rn,14), </td>;
echo <td>,odbc_field_name($rn,16), </td>;
echo <td>,odbc_field_name($rn,15), </td>;
do{
//
-for serial number
$sql3=select Code from main where ECG_Code=' '.odbc_result
($rn,1).' ';
$rn3=odbc_exec($connect,$sql3);
//
echo $sql3;
echo<tr>;
echo <td>,odbc_result($rn,1), </td>;
//
show serial number
echo <td>,odbc_result($rn3,1), </td>;
echo <td>,odbc_result($rn,2), </td>;
echo <td>,odbc_result($rn,3), </td>;
echo <td>,odbc_result($rn,4), </td>;
echo <td>,odbc_result($rn,5), </td>;
echo <td>,odbc_result($rn,6), </td>;
echo <td>,odbc_result($rn,7), </td>;
echo <td>,odbc_result($rn,8), </td>;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

echo "<td>",odbc_result ($rn,9) , "</td>";
echo "<td>",odbc_result ($rn,10) , "</td>";
echo "<td>",odbc_result ($rn,11) , "</td>";
echo "<td>",odbc_result ($rn,12) , "</td>";
echo "<td>",odbc_result ($rn,13) , "</td>";
echo "<td>",odbc_result ($rn,14) , "</td>";
echo "<td>",odbc_result ($rn,16) , "</td>";
$text="<a href='/08_Triacs/'.odbc_result ($rn,15) . '.jpg'>";
echo "<td>",$text,odbc_result ($rn,15) , "</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn)) ;
$bgc=($bgc=="beige")? "cyan": "beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

## รูปที่ 2 โปรแกรม triacs

```

<?
$dns="ecgproject";
echo "<body bgcolor=#FFFFFFA\>";
echo "<blockquote><table border=1 cellspacing=0><tr
bgcolor=SKYBLUE>";
$connect=odbc_connect ($dns, "", "");
If (($HFE=="none") && ($FT=="none")) {
$sql="select * from transistor where DECRIP_AP='".$$TYPE."'";
}elseif (($HFE=="none") && ($FT != "none")) {
$sql="select * from transistor where FT='".$$FT."' and
DECRIP_AP='".$$TYPE."'";
}elseif (($HFE!="none") && ($FT == "none")) {
$sql="select * from transistor where HFE='".$$HFE."' and
DECRIP_AP='".$$TYPE."'";}
else{$sql="select * from transistor where HFE='".$$HFE."' and
FT='".$$FT."' and DECRIP_AP='".$$TYPE."'";}
$rn=odbc_exec ($connect, $sql);
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1) . "'";
$rn3=odbc_exec ($connect, $sql3);
echo "<body bgcolor=#FFFFFFA\>";
echo "<blockquote><table border=1 cellspacing=0><tr
bgcolor=#FF6699\>";echo "<blockquote>
<table border=1 cellspacing=0><tr bgcolor=#FF6699\>";//
echo "<td>",odbc_field_name ($rn,1) , "</td>";
echo "<td>",odbc_field_name ($rn3,1) , "</td>";
echo "<td>",odbc_field_name ($rn,2) , "</td>";
echo "<td>",odbc_field_name ($rn,3) , "</td>";
echo "<td>",odbc_field_name ($rn,4) , "</td>";
echo "<td>",odbc_field_name ($rn,5) , "</td>";
echo "<td>",odbc_field_name ($rn,6) , "</td>";
echo "<td>",odbc_field_name ($rn,7) , "</td>";
echo "<td>",odbc_field_name ($rn,8) , "</td>";

```

```

echo "<td>",odbc_field_name($rn,9),"</td>";
echo "<td>",odbc_field_name($rn,10),"</td>";
echo "<td>",odbc_field_name($rn,11),"</td>";
do{
//for serial number
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)."'";
$rn3=odbc_exec($connect,$sql3);
//echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1),"</td>";
// show serial number
echo "<td>",odbc_result($rn3,1),"</td>";
//
echo "<td>",odbc_result($rn,2),"</td>";
echo "<td>",odbc_result($rn,3),"</td>";
echo "<td>",odbc_result($rn,4),"</td>";
echo "<td>",odbc_result($rn,5),"</td>";
echo "<td>",odbc_result($rn,6),"</td>";
echo "<td>",odbc_result($rn,7),"</td>";
echo "<td>",odbc_result($rn,8),"</td>";
echo "<td>",odbc_result($rn,9),"</td>";
echo "<td>",odbc_result($rn,10),"</td>";
$text="<a href='/01_Transistors/"'.odbc_result($rn,11).".jpg'>";
echo "<td>",$text,odbc_result($rn,11),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

### รูปที่ 3 โปรแกรม Transistor

```

<? $dsn="ecgproject";
echo "<body bgcolor=#FFFFFF>";
echo "<blockquote><table border=1 cellpadding=0 cellspacing=0 border=1
bgcolor=SKYBLUE>";
$connect=odbc_connect($dsn,"","");
if (($VDRM=="none") && ($RMS=="none") && ($AV=="none")) {
$sql="select * from scr";
}elseif (($VDRM=="none") && ($RMS=="none") && ($AV!="none")) {
$sql="select * from scr where AV='". $AV. "'";
}elseif (($VDRM=="none") && ($RMS!="none") && ($AV=="none")) {
$sql="select * from scr where RMS='". $RMS. "'";
}elseif (($VDRM!="none") && ($RMS=="none") && ($AV=="none")) {
$sql="select * from scr where V_DRM='". $VDRM. "'";
}elseif (($VDRM=="none") && ($RMS!="none") && ($AV!="none")) {
$sql="select * from scr where RMS='". $RMS. "' and AV='". $AV. "'";
}elseif (($VDRM!="none") && ($RMS=="none") && ($AV!="none")) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

$sql="select * from scr where V_DRM='". $VDRM. "' and
AV='". $AV. "'";
}elseif (($VDRM!="none") && ($RMS!="none") && ($AV=="none")) {
$sql="select * from scr where RMS='". $RMS. "' and
V_DRM='". $VDRM. "'";
}else{$sql="select * from scr where V_DRM='". $VDRM. "' And
RMS='". $RMS. "' and AV='". $AV. "'";}
$rn=odbc_exec($connect,$sql);
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)."'";
$rn3=odbc_exec($connect,$sql3);
echo "<body bgcolor=\\"#FFFFFFA\>";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=\\"#FF6699\>";echo "<blockquote>
<table border=1 cellpadding=0><tr bgcolor=\\"#FF6699\>";
//
echo "<td>",odbc_field_name($rn,1), "</td>";
echo "<td>",odbc_field_name($rn3,1), "</td>";
echo "<td>",odbc_field_name($rn,2), "</td>";
echo "<td>",odbc_field_name($rn,3), "</td>";
echo "<td>",odbc_field_name($rn,4), "</td>";
echo "<td>",odbc_field_name($rn,5), "</td>";
echo "<td>",odbc_field_name($rn,6), "</td>";
echo "<td>",odbc_field_name($rn,7), "</td>";
echo "<td>",odbc_field_name($rn,8), "</td>";
echo "<td>",odbc_field_name($rn,9), "</td>";
echo "<td>",odbc_field_name($rn,10), "</td>";
echo "<td>",odbc_field_name($rn,11), "</td>";
echo "<td>",odbc_field_name($rn,12), "</td>";
echo "<td>",odbc_field_name($rn,13), "</td>";
echo "<td>",odbc_field_name($rn,14), "</td>";
echo "<td>",odbc_field_name($rn,15), "</td>";
echo "<td>",odbc_field_name($rn,16), "</td>";
echo "<td>",odbc_field_name($rn,17), "</td>";
do{
// -for serial number
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)."'";
$rn3=odbc_exec($connect,$sql3);
//echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1), "</td>";
// show serial number
echo "<td>",odbc_result($rn3,1), "</td>";//
echo "<td>",odbc_result($rn,2), "</td>";
echo "<td>",odbc_result($rn,3), "</td>";
echo "<td>",odbc_result($rn,4), "</td>";
echo "<td>",odbc_result($rn,5), "</td>";
echo "<td>",odbc_result($rn,6), "</td>";
echo "<td>",odbc_result($rn,7), "</td>";
echo "<td>",odbc_result($rn,8), "</td>";
echo "<td>",odbc_result($rn,9), "</td>";
echo "<td>",odbc_result($rn,10), "</td>";
echo "<td>",odbc_result($rn,11), "</td>";
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

echo "<td>",odbc_result($rn,12),"</td>";
echo "<td>",odbc_result($rn,13),"</td>";
echo "<td>",odbc_result($rn,14),"</td>";
echo "<td>",odbc_result($rn,15),"</td>";
echo "<td>",odbc_result($rn,16),"</td>";
$text="<a href='/07_scr/" .odbc_result($rn,17) . ".jpg'>";
echo "<td>",$text,odbc_result($rn,17),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

#### รูปที่ 4 โปรแกรม SCR

```

<? $dsn="ecgproject";
echo "<body bgcolor=\ "#FFFFFFA\">";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=SKYBLUE>";
$connect=odbc_connect($dsn,"","");
if(($PRV=="none")&&($IO=="none")){
$sql="select * from industrial_rectifiers";
}elseif(($PRV=="none")&&($IO!="none")){
$sql="select * from industrial_rectifiers where IO=' ".$IO." '";
}elseif(($PRV!="none")&&($IO=="none")){
$sql="select * from industrial_rectifiers where PRV=' ".$PRV." '";
}else{$sql="select * from industrial_rectifiers where
PRV=' ".$PRV." ' and IO=' ".$IO." '";}
$rn=odbc_exec($connect,$sql);
$sql3="select Code from main where ECG_Code=' ".odbc_result
($rn,1)." ";
$rn3=odbc_exec($connect,$sql3);
echo "<body bgcolor=\ "#FFFFFFA\">";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=\ "#FF6699\">";echo "<blockquote>
<table border=1 cellpadding=0><tr bgcolor=\ "#FF6699\">";
//
echo "<td>",odbc_field_name($rn,1),"</td>";
echo "<td>",odbc_field_name($rn3,1),"</td>";
echo "<td>",odbc_field_name($rn,2),"</td>";
echo "<td>",odbc_field_name($rn,3),"</td>";
echo "<td>",odbc_field_name($rn,4),"</td>";
echo "<td>",odbc_field_name($rn,5),"</td>";
echo "<td>",odbc_field_name($rn,6),"</td>";
echo "<td>",odbc_field_name($rn,7),"</td>";
echo "<td>",odbc_field_name($rn,8),"</td>";
echo "<td>",odbc_field_name($rn,9),"</td>";
echo "<td>",odbc_field_name($rn,11),"</td>";
echo "<td>",odbc_field_name($rn,10),"</td>";
do{

```

```

// -for serial number
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)." ";
$rn3=odbc_exec($connect,$sql3);
//echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1),"</td>";
// show serial number
echo "<td>",odbc_result($rn3,1),"</td>";
//
echo "<td>",odbc_result($rn,2),"</td>";
echo "<td>",odbc_result($rn,3),"</td>";
echo "<td>",odbc_result($rn,4),"</td>";
echo "<td>",odbc_result($rn,5),"</td>";
echo "<td>",odbc_result($rn,6),"</td>";
echo "<td>",odbc_result($rn,7),"</td>";
echo "<td>",odbc_result($rn,8),"</td>";
echo "<td>",odbc_result($rn,9),"</td>";
echo "<td>",odbc_result($rn,11),"</td>";
$text="<a href='/06_IndustrialRectifier/"'.odbc_result
($rn,10)."'.jpg'>";
echo "<td>",$text,odbc_result($rn,10),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

### รูปที่ 5 โปรแกรม Industrial Rectifier

```

<html>
<body>
  <?&dsn="project";
  $connect=odbc_connect($dsn,"","");
  $sql="select * from high_voltage_devices ";
  $rn=odbc_exec($connect,$sql);
  $sql2="select count (*) from high_voltage_devices ";
  $rn2=odbc_exec($connect,$sql2);
  $nf=odbc_num_fields($rn);
  $nr=odbc_result($rn2,1);
  echo"<table border=1 ><tr>";
  for($i=1;$i<=$nf;$i++){
  echo "<td>",odbc_field_name($rn,$i),"</td>";
  }
  do{echo"<tr>";
  for($i=1;$i<=$nf;$i++){
  echo "<td>",odbc_result($rn,$i),"</td>";
  }
  echo"</tr>";}
  while(odbc_fetch_row($rn));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
echo "</table>";
?></body>
</html>
```

## รูปที่ 6 โปรแกรม High Voltage

```
<?&#x2D;dsn="ecgproject";
echo "<body bgcolor=\&#xFF6699\">";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=SKYBLUE>";
&#x2D;connect=odbc_connect (&#x2D;dsn, "", "");
if ((&#x2D;GFS=="none") && (&#x2D;VGS=="none") && (&#x2D;IDSS=="none")) {
&#x2D;sql="select * from fet";
}elseif ((&#x2D;GFS=="none") && (&#x2D;VGS == "none") && (&#x2D;IDSS!="none")) {
&#x2D;sql="select * from fet where IDSS_NIN_TO_MAX='&#x2D;IDSS.'";
}elseif ((&#x2D;GFS=="none") && (&#x2D;VGS!="none") && (&#x2D;IDSS=="none")) {
&#x2D;sql="select * from fet where VGS_OFF='&#x2D;VGS.'";
}elseif ((&#x2D;GFS!="none") && (&#x2D;VGS=="none") && (&#x2D;IDSS=="none")) {
&#x2D;sql="select * from fet where GFS='&#x2D;GFS.'";
}elseif ((&#x2D;GFS=="none") && (&#x2D;VGS!="none") && (&#x2D;IDSS!="none")) {
&#x2D;sql="select * from fet where VGS_OFF='&#x2D;VGS.'" and
IDSS_NIN_TO_MAX='&#x2D;IDSS.'";
}elseif ((&#x2D;GFS!="none") && (&#x2D;VGS=="none") && (&#x2D;IDSS!="none")) {
&#x2D;sql="select * from fet where GFS='&#x2D;GFS.'" and
IDSS_NIN_TO_MAX='&#x2D;IDSS.'";
}elseif ((&#x2D;GFS!="none") && (&#x2D;VGS!="none") && (&#x2D;IDSS=="none")) {
&#x2D;sql="select * from fet where GFS='&#x2D;GFS.'" and
VGS_OFF='&#x2D;VGS.'";
}else{&#x2D;sql="select * from fet where GFS='&#x2D;GFS.'" and
VGS_OFF='&#x2D;VGS.'" and IDSS_NIN_TO_MAX='&#x2D;IDSS.'";}
&#x2D;r=odbc_exec (&#x2D;connect, &#x2D;sql);
&#x2D;sql3="select Code from main where ECG_Code='&#x2D;odbc_result
(&#x2D;r,1)';
&#x2D;r3=odbc_exec (&#x2D;connect, &#x2D;sql3);
echo "<body bgcolor=\&#xFF6699\">";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=\&#xFF6699\">";echo "<blockquote>
<table border=1 cellpadding=0><tr bgcolor=\&#xFF6699\">";
//
echo "<td>",odbc_field_name (&#x2D;r,1) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r3,1) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,2) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,3) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,4) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,5) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,6) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,7) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,8) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,9) , "</td>";
echo "<td>",odbc_field_name (&#x2D;r,10) , "</td>";
do{
//&#x2D;for serial number
```

```

sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)."'" ;
$rn3=odbc_exec($connect,$sql3);
//echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1),"</td>";
// show serial number
echo "<td>",odbc_result($rn3,1),"</td>";
//
echo "<td>",odbc_result($rn,2),"</td>";
echo "<td>",odbc_result($rn,3),"</td>";
echo "<td>",odbc_result($rn,4),"</td>";
echo "<td>",odbc_result($rn,5),"</td>";
echo "<td>",odbc_result($rn,6),"</td>";
echo "<td>",odbc_result($rn,7),"</td>";
echo "<td>",odbc_result($rn,8),"</td>";
echo "<td>",odbc_result($rn,9),"</td>";
$text="<a href='/01_Transistors/'".odbc_result($rn,10)."'.jpg'>";
echo "<td>",$text,odbc_result($rn,10),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

### รูปที่ 7 โปรแกรม FET

```

<?&#x26;dsn="ecgproject";
echo "<body bgcolor=#FFFFFFA\>";
echo "<blockquote><table border=1 cellspacing=0<tr
bgcolor=SKYBLUE>";
$connect=odbc_connect($dsn,"","");
if (($TYPE=="none") && ($IO=="none") && ($VF=="none")) {
$sql="select * from diode";
}elseif (($TYPE=="none") && ($IO == "none") && ($VF!="none")) {
$sql="select * from diode where VF='". $VF. "'";
}elseif (($TYPE=="none") && ($IO!="none") && ($VF=="none")) {
$sql="select * from diode where IO='". $IO. "'";
}elseif (($TYPE!="none") && ($IO=="none") && ($VF=="none")) {
$sql="select * from diode where description='". $TYPE. "'";
}elseif (($TYPE=="none") && ($IO!="none") && ($VF!="none")) {
$sql="select * from diode where IO='". $IO. "' and VF='". $VF. "'";
}elseif (($TYPE!="none") && ($IO=="none") && ($VF!="none")) {
$sql="select * from diode where description='". $TYPE. "' and
VF='". $VF. "'";
}elseif (($TYPE!="none") && ($IO!="none") && ($VF=="none")) {
$sql="select * from diode where description='". $TYPE. "' and
IO='". $IO. "'";
}else{$sql="select * from diode where description='". $TYPE. "'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

and IO="'".$IO."' and VF="'".$VF."''";}
$rn=odbc_exec($connect,$sql);
$sql3="select Code from main where ECG_Code='".$odbc_result
($rn,1)."'";
$rn3=odbc_exec($connect,$sql3);
echo "<body bgcolor=\"#FFFFFFA\">";
echo "<blockquote><table border=1 cellspacing=0><tr
bgcolor=\"#FF6699\">;echo "<blockquote>
<table border=1 cellspacing=0><tr bgcolor=\"#FF6699\">";
//
echo "<td>",odbc_field_name($rn,1),"</td>";
echo "<td>",odbc_field_name($rn3,1),"</td>";
echo "<td>",odbc_field_name($rn,2),"</td>";
echo "<td>",odbc_field_name($rn,3),"</td>";
echo "<td>",odbc_field_name($rn,4),"</td>";
echo "<td>",odbc_field_name($rn,5),"</td>";
echo "<td>",odbc_field_name($rn,6),"</td>";
echo "<td>",odbc_field_name($rn,7),"</td>";
echo "<td>",odbc_field_name($rn,8),"</td>";
echo "<td>",odbc_field_name($rn,9),"</td>";
echo "<td>",odbc_field_name($rn,10),"</td>";
echo "<td>",odbc_field_name($rn,11),"</td>";
echo "<td>",odbc_field_name($rn,12),"</td>";
do{ //for serial number
$sql3="select Code from main where ECG_Code='".$odbc_result
($rn,1)."'";
$rn3=odbc_exec($connect,$sql3); //echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1),"</td>"; // show serial number
echo "<td>",odbc_result($rn3,1),"</td>"; //
echo "<td>",odbc_result($rn,2),"</td>";
echo "<td>",odbc_result($rn,3),"</td>";
echo "<td>",odbc_result($rn,4),"</td>";
echo "<td>",odbc_result($rn,5),"</td>";
echo "<td>",odbc_result($rn,6),"</td>";
echo "<td>",odbc_result($rn,7),"</td>";
echo "<td>",odbc_result($rn,8),"</td>";
echo "<td>",odbc_result($rn,9),"</td>";
echo "<td>",odbc_result($rn,10),"</td>";
echo "<td>",odbc_result($rn,11),"</td>";
$text="<a href='/03_Diode/'."."odbc_result($rn,12).".jpg'>";
echo "<td>",$text,odbc_result($rn,12),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

## รูปที่ 8 โปรแกรม Diode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<? $dsn="ecgproject" ;
echo "<body bgcolor=\ "#FFFFFFA\ ">" ;
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=SKYBLUE>" ;
$connect=odbc_connect($dsn, "", "");
if (($F_Mhz=="none") && ($FT=="none")) {
$sql="select * from quartz_crystals";
}elseif (($F_Mhz=="none") && ($FT != "none")) {
$sql="select * from quartz_crystals where FT='".$FT."'";
}elseif (($F_Mhz!="none") && ($FT == "none")) {
$sql="select * from quartz_crystals where F_Mhz='".$F_Mhz."'";
}else {$sql="select * from quartz_crystals where
F_Mhz='".$F_Mhz."' and FT='".$FT."'";}
$rn=odbc_exec($connect, $sql);
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)." ";
$rn3=odbc_exec($connect, $sql3);
echo "<body bgcolor=\ "#FFFFFFA\ ">" ;
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=\ "#FF6699\ ">" ;echo "<blockquote>
<table border=1 cellpadding=0><tr bgcolor=\ "#FF6699\ ">" ;
//
echo "<td>",odbc_field_name($rn,1), "</td>";
echo "<td>",odbc_field_name($rn3,1), "</td>";
echo "<td>",odbc_field_name($rn,2), "</td>";
echo "<td>",odbc_field_name($rn,3), "</td>";
echo "<td>",odbc_field_name($rn,4), "</td>";
echo "<td>",odbc_field_name($rn,5), "</td>";
echo "<td>",odbc_field_name($rn,6), "</td>";
echo "<td>",odbc_field_name($rn,7), "</td>";
echo "<td>",odbc_field_name($rn,8), "</td>";
echo "<td>",odbc_field_name($rn,9), "</td>";
echo "<td>",odbc_field_name($rn,10), "</td>";
do{
//-for serial number
$sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)." ";
$rn3=odbc_exec($connect, $sql3);
//echo $sql3;
echo"<tr>";
echo "<td>",odbc_result($rn,1), "</td>";
// show serial number
echo "<td>",odbc_result($rn3,1), "</td>";
//
echo "<td>",odbc_result($rn,2), "</td>";
echo "<td>",odbc_result($rn,3), "</td>";
echo "<td>",odbc_result($rn,4), "</td>";
echo "<td>",odbc_result($rn,5), "</td>";
echo "<td>",odbc_result($rn,6), "</td>";
echo "<td>",odbc_result($rn,7), "</td>";
echo "<td>",odbc_result($rn,8), "</td>";
echo "<td>",odbc_result($rn,9), "</td>";
$text="<a
href='/11_Quartz_crystal/'.odbc_result($rn,10)." .'.jpg'>";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

echo "<td>",$text,odbc_result($rn,10),"</a>","</td>";
echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

### รูปที่ 9 โปรแกรม Crystal

```

<?$dsn="ecgproject";
echo "<body bgcolor=#FFFFFFA\>";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=SKYBLUE>";
$connect=odbc_connect($dsn,"","");
if(($PRV=="none")&&($IO=="none")&&($Peak_FSC=="none")){
$sql="select * from bridge_rectifiers";
}elseif(($PRV=="none")&&($IO=="none")&&($Peak_FSC!="none")){
$sql="select * from bridge_rectifiers where
peak_FSC='".$Peak_FSC.'"";
}elseif(($PRV=="none")&&($IO!="none")&&($Peak_FSC=="none")){
$sql="select * from bridge_rectifiers where IO='".$IO.'"";
}elseif(($PRV!="none")&&($IO=="none")&&($Peak_FSC=="none")){
$sql="select * from bridge_rectifiers where PRV='".$PRV.'"";
}elseif(($PRV=="none")&&($IO!="none")&&($Peak_FSC!="none")){
$sql="select * from bridge_rectifiers where IO='".$IO.'" and
peak_FSC='".$Peak_FSC.'"";
}elseif(($PRV!="none")&&($IO=="none")&&($Peak_FSC!="none")){
$sql="select * from bridge_rectifiers where PRV='".$PRV.'" and
peak_FSC='".$Peak_FSC.'"";
}elseif(($PRV!="none")&&($IO!="none")&&($Peak_FSC=="none")){
$sql="select * from bridge_rectifiers where PRV='".$PRV.'" and
IO='".$IO.'"";
}else{$sql="select * from bridge_rectifiers where PRV='".$PRV.'"
and IO='".$IO.'" and Peak_FSC='".$Peak_FSC.'"";}
$rn=odbc_exec($connect,$sql);
$sql3="select Code from main where ECG_Code='".$odbc_result
($rn,1).'"";
$rn3=odbc_exec($connect,$sql3);
echo "<body bgcolor=#FFFFFFA\>";
echo "<blockquote><table border=1 cellpadding=0><tr
bgcolor=#FF6699\>";echo "<blockquote>
<table border=1 cellpadding=0><tr bgcolor=#FF6699\>";
//
echo "<td>",$odbc_field_name($rn,1),"</td>";
echo "<td>",$odbc_field_name($rn3,1),"</td>";
echo "<td>",$odbc_field_name($rn,2),"</td>";
echo "<td>",$odbc_field_name($rn,3),"</td>";
echo "<td>",$odbc_field_name($rn,4),"</td>";
echo "<td>",$odbc_field_name($rn,5),"</td>";
echo "<td>",$odbc_field_name($rn,6),"</td>";

```

```

echo "<td>",odbc_field_name($rn,7),"</td>";
do{
  //-for serial number
  $sql3="select Code from main where ECG_Code='".odbc_result
($rn,1)."'";
  $rn3=odbc_exec($connect,$sql3);
  //echo $sql3;
  echo"<tr>";
  echo "<td>",odbc_result($rn,1),"</td>";
  // show serial number
  echo "<td>",odbc_result($rn3,1),"</td>";
  //
  echo "<td>",odbc_result($rn,2),"</td>";
  echo "<td>",odbc_result($rn,3),"</td>";
  echo "<td>",odbc_result($rn,4),"</td>";
  echo "<td>",odbc_result($rn,5),"</td>";
  echo "<td>",odbc_result($rn,6),"</td>";
  $text="<a
href='/05_BridgeRectifier/'.odbc_result($rn,7).'.jpg'>";
  echo "<td>",$text,odbc_result($rn,7),"</a>","</td>";
  echo"</tr>";
}
while(odbc_fetch_row($rn));
$bgc=($bgc=="beige")?"cyan":"beige";
echo"<tr bgcolor=$bgc>";
echo "</table>";
?>

```

### รูปที่ 10 โปรแกรม Bridge Rectifier

```

<html>
<head>
  <title>ECG ON IN TRANET</title>
</head>
<FRAMESET border=0 frameBorder=0
  frameSpacing=0 cols="22%,*">
<FRAME src="menu.htm" name=left>
<FRAME src="ecg.html" name=right>
</frameset><noframes></noframes>
</html>

```

รูปที่ 11 ซอร์สโค้ด main.html

```

<html>
<head>
  <title>Untitled Document</title>
<script language="JavaScript">
<!--function MM_swapImgRestore() { //v3.0
  var i,x,a=document.MM_sr; for(i=0;a&&I
<a.length&&(x=a[i])&&x.oSrc;i++)
  x.src=x.oSrc;
}
function MM_preloadImages() { //v3.0
var d=document; if(d.images){ if(!d.MM_p)
d.MM_p=new Array();
var i,j=d.MM_p.length,
a=MM_preloadImages.arguments; for(i=0;
i<a.length; i++)
  if (a[i].indexOf("#")!=0){ d.MM_p[j]=new
  Image; d.MM_p[j++].src=a[i];}}
function MM_findObj(n, d) { //v3.0
var p,i,x;  if(!d) d=document; if
((p=n.indexOf("?"))>0&&parent.frames.length)
  {d=parent.frames[n.substring
(p+1)].document; n=n.substring(0,p);}
if (!(x=d[n])&&d.all) x=d.all[n]; for
(i=0;!x&&i<d.forms.length;i++) x=d.forms[i]
[n];
for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj
(n,d.layers[i].document); return x;}

function MM_swapImage() { //v3.0
  var i,j=0,x,a=MM_swapImage.arguments; document.MM_sr=new Array;
for(i=0;i<(a.length-2);i+=3)
  if ((x=MM_findObj(a[i]))!=null){document.MM_sr[j++]=x;
  if(!x.oSrc) x.oSrc=x.src; x.src=a[i+2];}}//-->
</script>
</head>
<body bgcolor="#FFFFFF" background="bg1.gif"
onLoad="MM_preloadImages
('ANIMATE/SEARCH2.jpg', 'ANIMATE/ECG2.jpg',

```

```
'ANIMATE/LINK2.jpg', 'ANIMATE/BOARD2.jpg',
'ANIMATE/SIGN2.jpg', 'ANIMATE/GUEST2.jpg',
'ANIMATE/ABOUT2.jpg')">
<div align="left">
<a href="ecg.html" target="right" onMouseOut="MM_swapImgRestore()
" onMouseOver="MM_swapImage('MEI1','',
'ANIMATE/ECG2.jpg',1)">
</a></div>
<div align="left"><a href="content01.html"
target="right" onMouseOut="MM_swapImgRestore
()" onMouseOver="MM_swapImage
('MEI3','', 'ANIMATE/SEARCH2.jpg',1)">
</a></div>
<div align="left"><a href="link.html"
target=right onMouseOut="MM_swapImgRestore
()" onMouseOver="MM_swapImage
('MEI5','', 'ANIMATE/LINK2.jpg',1)">
</a></div>
<div align="left"><a href="#"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('MEI6','',
'ANIMATE/BOARD2.jpg',1)">
</a></div>
<div align="left"><a href="guest.html"
target="right"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('MEI7','',
'ANIMATE/SIGN2.jpg',1)">
</a></div>
<div align="left"><a href="#"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('MEI8','',
'ANIMATE/GUEST2.jpg',1)">
</a></div>
<div align="left"><a href="about.html"
target=right onMouseOut="MM_swapImgRestore
()" onMouseOver="MM_swapImage
('MEI9','', 'ANIMATE/ABOUT2.jpg',1)">
</a></div>
```







```

<div align="center">
<select name="menu2" onChange="MM_jumpMenu
('parent.frames[\ 'down\'],'',this,0)">
<option value="bridge.php" selected>Bridge
  Rectifier</option>
<option value="diode.php">Diode And Rectifier
</option>
<option value="fet.php">FET</option>
<option value="#">High Voltage Devices
</option>
<option value="#">IC Protector</option>
<option value="#">Industrial Power Modules
</option>
<option value="industrial.php">Industrial
  Rectifier</option>
<option value="#">Linear Integrated Circuit
</option>
<option value="#">Over Voltage Transient
  Suppressors</option>

<option value="crystals.php">Quartz Crystals</option>
<option value="scr.php">Silicon Controlled Rectifier
(SCR)</option>
<option value="#">Special Purpose Devices</option>
<option value="#">Thermal Cut-Offs (TCOs)</option>
<option value="tr.php">Transistor</option>
<option value="triac.php">TRIACS</option>
<option value="zener.php">Zener Diode</option></select>
</div></form></body></html>

```

รูปที่ 15 ซอร์สโค้ด content.html

```

<html>
<head>
  <title>LINK ME</title>
</head>
<body bgcolor="#FFFFFF" background="bg1.gif" text="#9999FF"
  link="#00CC66" vlink="#FF3333" alink="#FFFF99"
  bgproperties="fixed">
<div align="center">
  <p>&nbsp;</p>
  <p><font face="MS Sans Serif, AngsanaUPC" color="#CC99CC"
  size="2"><b><font color="#9966FF">Homepage<font size="3">
  และ โปรแกรมภาษา</font></font></b></font></p>
  <table width="50%" border="1">
<tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.go.to/programmer.com">Programmer</a>
</font></td>
<td><font face="AngsanaUPC, CordiaUPC" size="4">
<a href="http://crclab.hypermart.net">เขียน โปรแกรมบนอินเทอร์เน็ต

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</a></font></td>
</tr>
<tr><td><font face="Comic Sans MS" size="-1">
<a href="http://cgi-bin.spaceports.com/%7Esansak">php
- Web Development</a></font></td>
<td><a href="http://tutor.dserver.org"><font size="-1"
face="Comic Sans MS">D'tutor</font></a></td>
</tr>
<tr>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.phpdev.f2s.com">phpDEV</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://simplebyte.hypermart.net">SimpleByte</a>
</font></td></tr>
<tr>
<td><font face="Comic Sans MS" size="-1">
<a href="http://i.am/monchai">Java And Java Script</a>
</font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.php.net">php</a></font></td>
</tr><tr>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.thaicode.com">Thaicode</a></font></td>
<td>&nbsp;</td>
</tr>
</table>
<p><font face="Comic Sans MS" color="#0099FF">Computer
& Internet</font></p>
<table width="50%" border="1" name="Computer"
align="center">
<tr><td width="51%">
<div align="left"><font face="Comic Sans MS"
color="#CCCCCC"><a href="http://members.xoom.com/nekto">
<font size="-1">Nekto</font></a></font></div>
</td><td width="49%">
<font face="AngsanaUPC, CordiaUPC, MS Serif">
<a href="http://www.geocities.com/MadisonAvenue/Newsstand/
5556/index.htm">การออกแม่บ
CPU โดยใช้ FPGA</a></font></td>
</tr>
<tr>
<td width="51%">
<div align="left"><a href="http://siam.to/softhai">
<font face="Comic Sans MS" size="-1">Softhai</font></a>
</div>
</td>
<td width="49%"><font face="AngsanaUPC, CordiaUPC, MS
Serif">
<a href="http://www.krirk.ac.th/comboard/comboard.html">
เทคนิคคอมพิวเตอร์</a></font></td>
</tr>
<tr>

```

```

<td width="51%">
<div align="left"><font face="Comic Sans MS" size="-1">
<a href="http://hwpc.cjb.net/">HardwarePC</a></font></div>
</td><td width="49%">
<font face="AngsanaUPC, CordiaUPC, MS Serif">
<a href="http://www.fortunecity.com/skyscraper/chaos/917/">
การเลือกซื้อคอมพิวเตอร์ </a> </font></td>
</tr>
<tr>
<td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://hardzone.hypermart.net/">HardwareZone</a>
</font></td>
<td width="49%"><font face="Comic Sans MS" size="-1">
<a href="http://come.to/black_box">Black Box Computer</a>
</font></td>
</tr><tr>
<td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://home.nakhon.net/masterpc">MasterPC</a>
</font></td>
<td width="49%"><font face="Comic Sans MS">
<a href="http://www.geocities.com/SiliconValley/Bay/1115">
<font size="-1">Tip & Trick</font></a></font></td></tr>
<tr><td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://i.am/jammy007">Jammy</a></font></td>
<td width="49%"><font face="AngsanaUPC, CordiaUPC, MS
Serif"><a href="http://www.thaicomputer.com/">
ไทยคอมพิวเตอร์</a></font></td></tr><tr>
<td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://w3.to/konamiworld">Konami World</a></font>
</td><td width="49%"><font face="Comic Sans MS" size="-1">
<a href="http://www.palmdino.com/">ParmDINO</a></font></td>
</tr><tr>
<td width="51%"><font face="Comic Sans MS">
<font face="Comic Sans MS"><font size="-1">
<a href="http://thaisoft.123go.cx/">ThaiSolf</a></font>
</font></font></td>
<td width="49%"><font face="Comic Sans MS" size="-1">
<a href="http://www.thaibeginner.net/">Thaibegining</a>
</font></td></tr>
<tr><td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://www.geocities.com/form_th">net knowledgeo
</a></font></td>
<td width="49%"><font face="Comic Sans MS" size="-1">
<a href="http://www.bangkokhome.com/%7Ewattana">ThailandIT
</a></font></td>
</tr>
<tr>
<td width="51%"><font face="Comic Sans MS" size="-1">
<a href="http://parinya.cjb.net/">Parinya's Homepage</a>
</font></td>
<td width="49%"><font size="-1" face="Comic Sans MS">
<a href="http://goldenship.com/">GoldenShip</a></font></td>
</tr><tr><td width="51%">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div align="left"><font face="Comic Sans MS">
<font face="Comic Sans MS" size="-1">
<a href="http://download2000.hypermart.net/">Download</a>
</font><font face="Comic Sans MS"></font></font></div>
</td>
<td width="49%"><font face="Comic Sans MS" size="-1">
<a href="http://passkorn.hypermart.net">ThaiWBI</a></font>
</td></tr></table>
<p><font face="Comic Sans MS" color="#FFFF00">Search Web
</font></p>
<table width="51%" border="1"><tr>
<td height="15" width="49%"><font face="AngsanaUPC,
CordiaUPC" size="4"><a href="http://www.linksiam.com/">
ลิงก์สยาม</a></font></td>
<td height="15" width="51%"><font face="AngsanaUPC,
CordiaUPC" size="4"><a href="http://go.to/hooray">ฮูเร่</a>
</font></td></tr>
<tr><td width="49%"><font face="MS Sans Serif, AngsanaUPC"
size="-1"><a href="http://www.siamguru.com/">Siamguru</a>
</font></td>
<td width="51%"><font face="AngsanaUPC, CordiaUPC" size="4">
<a href="http://www.janenet.com/"><font face="Comic Sans MS"
size="-1">Jananet</font></a></font></td>
</tr><tr><td width="49%">
<font face="Comic Sans MS" size="-1">
<a href="http://www.thailinkcenter.com/">Thailinkcenter</a>
</font></td><td width="51%">
<font face="Comic Sans MS" size="-1">
<a href="http://www.jamseek.com/">Jamseek</a></font></td>
</tr><tr>
<td width="49%" height="24">
<font face="Comic Sans MS" size="-1">
<a href="http://www.noseek.com/">Noseek</a></font></td>
<td width="51%" height="24">
<font face="Comic Sans MS" size="-1">
<a href="http://www.thaitop.com/">Thaitop</a></font></td>
</tr><tr><td width="49%">
<font face="Comic Sans MS" size="-1">
<a href="http://www.technopolis.or.th/">Smart Search</a>
</font></td>
<td width="51%"><font face="AngsanaUPC, CordiaUPC" size="4">
<a href="http://www.yoyoo.com/">โยโย่</a></font></td>
</tr></table>
<p><font face="Comic Sans MS" color="#FF6633">General</font>
</p><table width="50%" border="1">
<tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.hunsa.com">Hunsa</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.sanook.com">Sanook</a></font></td>
</tr><tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.narak.com">Narak</a></font></td>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<td><font face="Comic Sans MS" size="-1">
<a href="http://www.saim2you.com">Siam2you</a></font></td>
</tr><tr>
<td><font face="MS Sans Serif, AngsanaUPC" size="-1">
<a href="http://www.lemononline.com">Lemononline</a></font>
</td><td><font face="Comic Sans MS" size="-1">
<a href="http://www.thaiicq.com">Thaiicq</a></font></td>
</tr><tr>
<td><a href="http://www.catcha.co.th">Catcha</a></td>
<td>&nbsp;</td>
</tr></table>
<p><font face="Comic Sans MS" color="#FF0000">Free E-Mail
</font></p>
<table width="50%" border="1"> <tr>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.chaiyo.com">Chaiyo</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.i-kool.com">i-kool</a></font></td>
</tr><tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.hotmail.com">Hotmail</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.thaimail.com">Thaimail</a></font></td>
</tr><tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.kittymail.com">Kitty</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.doraemonmail.com">Doraemon</a></font>
</td></tr>
<tr><td><a href="http://www.lycos.com">
<font face="Comic Sans MS" size="-1">Lycos</font></a></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.yahoo.com">Yahoo</a></font></td>
</tr><tr><td><font face="Comic Sans MS" size="-1">
<a href="http://hide.i-p.com">Youname@hide.i-p.com</a>
</font></td>
<td><a href="http://xjapan.i-p.com">Youname@xjapan.i-p.com
</a></td></tr></table>
<p><font face="Comic Sans MS" color="#FFFFFF">
Free HomePage</font></p>
<table width="50%" border="1">
<tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.geocities.com">Geocities</a></font></td>
<td><font face="Comic Sans MS" size="-1">
<a href="http://www.sanook.to">sanook.to</a></font></td>
</tr><tr><td><font face="Comic Sans MS" size="-1">
<a href="http://www.thai.to">Thai.to</a></font></td>
<td><font face="Comic Sans MS" size="-1"></font></td>
</tr></table>
<p>&nbsp;</p>
</div> </body>
</html>

```

รูปที่ 16 ซอร์สโค้ด link.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





## บรรณานุกรม

วุฒิพงษ์ พงศ์สุวรรณ และทีมงานวิชาการ. **Web Mastering ด้วย PHP**. กรุงเทพฯ :

DLS กรุงเทพฯ. 2543

มนัท พูลสวัสดิ์ และเกียรติประถม สีนรุ่งเรืองกุล. **Microsoft SQL Server**. กรุงเทพฯ : ว.เพ็ชรสกุล . 2543

จิตเกษม พัฒนาศิริ. **เริ่มสร้างโฮมเพจด้วย<HTML>**. กรุงเทพฯ : วิตดี กรุ๊ป. 2543

กิตติ ภัคดีวัฒนากุล และจำลอง ครูอุตสาหะ. **คำภีร์ระบบฐานข้อมูล**. พิมพ์ครั้งที่ 2. กรุงเทพฯ :

ดวงกมล. 2542

กิตติศักดิ์ เจริญ โภคานนท์. **สร้างเว็บได้ตั้งใจนี้กับด้วย PHP**. กรุงเทพฯ : ซัคเซส มีเดีย. 2543

กิตติภูมิ วรฉัตร. **PHP เปลี่ยนวิธีการสร้างโฮมเพจอย่างมือโปร**. กรุงเทพฯ : วิตดี กรุ๊ป. 2543

บัณฑิต จามรภูติ. **การใช้งานฐานข้อมูลเชิงสัมพันธ์ Microsoft SQL Server**. กรุงเทพฯ : ว. เพ็ชร  
สกุล. 2541

สมพร จิวรสกุล. **คู่มือการติดตั้งและใช้งาน SQL Server 7.0 ฉบับสมบูรณ์**. กรุงเทพฯ : อินโฟเพรส.  
2543

## ประวัติผู้แต่ง



ชื่อผู้ทำปริญญาบัตร	นางสาวประกายเพชร อินทรารุช
วันเดือนปีเกิด	9 เมษายน 2521
สถานที่เกิด	จังหวัดนครสวรรค์
ภูมิลำเนาเดิม	67 หมู่ 6 อ. เมือง จ.นครสวรรค์ 60000
ที่อยู่ปัจจุบัน	801/2205 ถ.หลวงแพ่ง เขตลาดกระบัง กทม. 10520
โทรศัพท์	(02) 7391455
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนลาซาลโชติรวินนครสวรรค์
มัธยมศึกษาตอนต้น	โรงเรียนลาซาลโชติรวินนครสวรรค์
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคนครสวรรค์
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	สถาบันเทคโนโลยีราชมงคลวิทยาเขต นนทบุรี
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	รองชนะเลิศอันดับ 2 การแข่งขันขับร้อง เพลงสากล อชท. ระดับประเทศ ปี 2538
ทุนการศึกษา	-
คติพจน์	คบเพื่อนดีเป็นศรีแก่ตัว คบเพื่อนชั่วพาตัวมัวหมอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานิพนธ์	นางสาวปิยฉัตร สุขสุรัส
วันเดือนปีเกิด	22 ตุลาคม 2518
สถานที่เกิด	จังหวัดนนทบุรี
ภูมิลำเนาเดิม	54/21 อ.เมือง จ.นนทบุรี 11000
ที่อยู่ปัจจุบัน	397/1 หมู่ 1 ซ.จินดาภิเวศน์ 10 ถ.อ่อนนุช เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(02) 3268456 ห้อง 3 ใหม่
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนสตรีบูรณวิทย์
มัธยมศึกษาตอนต้น	โรงเรียนสตรีบูรณวิทย์
มัธยมศึกษาปีที่ 4	โรงเรียนรัตนาธิเบศร์
ประกาศนียบัตรวิชาชีพชั้นสูงหลักสูตร 4 ปี	สถาบันเทคโนโลยีราชมงคลวิทยาเขตนนทบุรี
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	กองทุนกู้ยืมเพื่อการศึกษา
คติพจน์	คนที่ไม่เคยทำอะไรผิด คือคนที่ไม่เคย ทำอะไรเลย การทำอะไรก็ตาม ทำให้ดี ที่สุดก็พอแล้ว แต่ต้องดีที่สุคจริงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อผู้ทำปริญญาบัตร	นางสาวสุภวรรณ ปลั่งพงษ์พันธ์
วันเดือนปีเกิด	24 มีนาคม 2521
สถานที่เกิด	จังหวัดเลย
ภูมิลำเนาเดิม	175/4 หมู่ 10 อ.วังสะพุง จ.เลย 42130
ที่อยู่ปัจจุบัน	397/1 ซ.จินดาภิเษก 10 ถ.อ่อนนุช เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(02) 3268456 ห้อง 20 เก้า
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนชุมชนวังสะพุง
มัธยมศึกษาตอนต้น	โรงเรียนเลยพิทยาคม
มัธยมศึกษาปีที่ 4	โรงเรียนแก่นนครวิทยาลัย
ประกาศนียบัตรวิชาชีพชั้นสูงหลักสูตร 4 ปี	สถาบันเทคโนโลยีราชมงคลวิทยาเขต ขอนแก่น
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	- เหรียญเงินเทวณีสหคิงส์ กิจการฯ มงคล
ทุนการศึกษา	กองทุนกู้ยืมเพื่อการศึกษา
คติพจน์	จงอย่าให้อภัยผู้ที่ทำผิดอยู่เสมอไม่เช่นนั้น ท่านจะกลายเป็นผู้ผิดเสียเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้