



งานบริการห้องสมุด

LIBRARY SERVICE SYSTEM

โดย

นายวรวิทย์ ยี่สุน

นายศรายุทธ บุญเตี้ย

นายอดิศักดิ์ รักสละม

อาจารย์ที่ปรึกษา

อาจารย์มยุรี เลิศเวชกุล

วัน เดือน ปี.....	14. ธ.ค. 2541
เลขทะเบียน.....	038900
เลขเรียกหนังสือ.....	โ 201110 02859

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์อื่นใด
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038900

ปริญญาโทปีการศึกษา 2540

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผู้จัดทำ

1. นาย วรุฒิ ยี่สุน
2. นาย ศรายุทธ บุญเตี้ย
3. นาย อติศักดิ์ รักสลาม

..... อาจารย์ที่ปรึกษา

(อ. มยุรี เลิศเวชกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ งานบริการห้องสมุด

โดย

นาย วรุฒิ ยี่สุน

นาย ศรายุทธ บุญเตี้ย

นาย อติศักดิ์ รักสกลาม

อาจารย์ที่ปรึกษา อ.มยุรี เลิศเวชกุล

ภาควิชาเทคนิคอุตสาหกรรม

ปีการศึกษา 2540

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังอนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการตรวจสอบปริญญานิพนธ์

..... ประธานกรรมการ

()

..... กรรมการ

()

..... กรรมการ

()

..... กรรมการ

()

งานบริการห้องสมุด

นาย วรุฒิ ยี่สุน
นาย ศราวุธ บุญเตี้ย
นาย อติศักดิ์ รักสลาม

อ. มยุรี เลิศเวชกุล อาจารย์ที่ปรึกษา

บทคัดย่อ

ปฏิญญาพันธบัตรฉบับนี้กล่าวถึงงานบริการห้องสมุดมีวัตถุประสงค์เพื่อศึกษาและพัฒนางานบนโปรแกรม เดลไฟล์ (Delphi) ที่ทำงานบนวินโดวส์ซึ่งมีข้อดีคือการพัฒนาโปรแกรมด้วย เดลไฟล์ จะพัฒนาได้เร็วกว่าเนื่องจาก เดลไฟล์ เป็นการพัฒนาโปรแกรมแบบวิซวล โปรแกรมที่เขียนขึ้นใช้รองรับงานบริการและจัดการของห้องสมุดภาคเทคนิคอุตสาหกรรม เช่น การคืนหนังสือ, การยืมหนังสือ, การค้นหาหนังสือ, รายการสมาชิก ที่รวมถึง การแก้ไข, การลบ ,การค้นหาข้อมูลเป็นต้น ซึ่งรายละเอียดต่างๆ ได้แสดงไว้ในปฏิญญาพันธบัตรฉบับนี้แล้ว

LIBRARY SERVICE SYSTEM

MR. WORAWUT YEESUN

MR. SARAYUT BOONTIA

MR. ADISAK RAKSALAM

MRS. MAYUREE LERTWATECHAKUL ADVISOR

ABSTRACT

This project will introduce programming library service system, object is to study and develop Delphi application for windows. Delphi better using because Delphi is visual programming development . The program can keep update information of library service , such as sent book, borrowed books , find book , member: including edit ,delete , check information , booking ,etc , detail presented in this thesis.

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์ มยุรี เลิศเวชกุล อาจารย์ประจำคณะวิศวกรรมศาสตร์ ที่เป็นผู้ให้คำปรึกษา คำแนะนำในการแก้ไขข้อบกพร่องต่างๆ จนโครงการนี้สำเร็จลุล่วงไปได้ด้วยดี นอกจากนี้ท่านยังเป็นอาจารย์ที่ปรึกษาของโครงการนี้ด้วย ขอขอบคุณเพื่อนๆ ทุกคนที่ให้คำแนะนำ และคอยช่วยเหลือด้วยดีเสมอมา และขอขอบคุณบิดามารดาที่สนับสนุนด้านการเงินและคอยให้กำลังใจมาโดยตลอด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนำ

ในปัจจุบันห้องสมุดมีความจำเป็นต่อนักศึกษาและประชาชนที่มีความสนใจในการอ่านหนังสือ ซึ่งห้องสมุดจะมีการบริการการยืม การคืน หนังสือ ในระบบเก่าจะใช้เจ้าหน้าที่บรรณารักษ์ จดบันทึกในการยืม การคืน หนังสือของสมาชิกซึ่งบางครั้งสมาชิกเข้ามาใช้บริการเป็นจำนวนมากจึงทำให้เสียเวลามากในการจดบันทึกและค้นหา ในปัจจุบันเราสามารถใช้ระบบการปฏิบัติการทาง Computer เพื่อจดบันทึก การยืม การคืน หนังสือได้

เดลไฟล์ เป็นเครื่องมือในการพัฒนาโปรแกรมประยุกต์การใช้งาน เป็นการเขียนโปรแกรมบนวินโดวส์ เดลไฟล์ เป็นการทำงานในลักษณะของวิซวลซึ่งในยุคแรกๆของการเขียนโปรแกรมบนวินโดวส์เขียนด้วยภาษา C ต่อจากภาษา C ก็เป็นยุคของโอโอทีคือเป็นการเขียนโปรแกรมแบบ Object oriented Programming ด้วยเทอร์โบปาสคาล เบอร์แลนด์ปาสคาล หรือ C++สำหรับการเขียนโปรแกรมบนวินโดวส์ยุควิซวล (Visual) เป็นโปรแกรมในลักษณะด้วยภาพ

ในโปรแกรมที่ทางคณะผู้จัดทำได้ทำขึ้นมาเป็นการนำเอาโปรแกรมเดลไฟล์มาสร้างเป็นโปรแกรมงานบริการห้องสมุดซึ่งจะสามารถใช้จดบันทึกการยืม การคืน และสามารถให้จดบันทึกประวัติของสมาชิกได้ด้วย เช่นสามารถทราบได้เลยว่าสมาชิกมีหนังสือค้างอยู่ที่เล่ม และยืมมาเป็นเวลานานเท่าใด อีกทั้งยังสามารถสืบค้นหาหนังสือได้ทั้งตามรายชื่อหนังสือ และตามชื่อผู้แต่ง โดยในเนื้อหาทั้งหมดจะเขียนถึงวิธีการสร้าง วิธีการใช้โปรแกรม ให้อย่างละเอียด

ทางผู้จัดทำก็หวังเป็นอย่างยิ่งว่าโปรแกรมระบบจัดการห้องสมุดที่ได้จัดทำขึ้นคงจะมีประโยชน์กับเจ้าหน้าที่ห้องสมุดบ้างไม่มากก็น้อยและถ้ามีข้อผิดพลาดประการใดก็ขออภัยไว้ ณ. ที่นี้ด้วย

สารบัญ

	หน้า
บทที่ 1 บทนำ	
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
บทที่ 2 ทฤษฎีและหลักการเบื้องต้น	2
2.1 การทำงานของระบบห้องสมุด	2
2.2 ทฤษฎีการวิเคราะห์และออกแบบระบบ	3
2.3 วิธีการรวบรวมความต้องการของผู้ใช้	3
2.4 ระบบฐานข้อมูล	4
2.5 การจัดเก็บข้อมูลภายในฐานข้อมูล	6
บทที่ 3 เครื่องมือที่ใช้ในการพัฒนาโปรแกรมประยุกต์สำหรับฐานข้อมูล	11
3.1 SQL	11
3.2 การใช้ SQL จัดการข้อมูล	11
บทที่ 4 รายละเอียดของโปรแกรมงานบริการห้องสมุด	15
4.1 โปรแกรมเดสไพล์	15
4.2 การออกแบบฐานข้อมูล	16
บทที่ 5 การใช้งานโปรแกรมเดสไพล์	19
5.1 วินโดวส์ Form	19
5.2 การดูรายละเอียดต่างๆด้วยวินโดวส์ Object Inspector	20
5.3 Code Editor	21
5.4 วินโดวส์ Main	21
5.5 กำหนดตำแหน่งของส่วนประกอบ	22
5.6 การดึงข้อมูลจากแหล่งข้อมูล	23
5.7 คิวรี เพื่อหาข้อมูล	23
บทที่ 6 การสร้างแอปพลิเคชันของงานบริการห้องสมุด	24
6.1 กฎในการใช้บริการห้องสมุดและข้อมูลที่ต้องทราบ	24
6.2 Code editor ที่ใช้ในโปรแกรม	25
บทที่ 7 ความสามารถของโปรแกรมงานบริการห้องสมุด	125

บทที่ 8 บทสรุป	126
8.1 ผลการทดลอง	126
8.2 การวิเคราะห์ปัญหาที่เกิดขึ้น	126
8.3 การพัฒนาและปรับปรุงโปรแกรมต่อไปในอนาคต	127
เอกสารอ้างอิง	128



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 การสร้างความสัมพันธ์แบบหนึ่งต่อหนึ่ง	6
รูปที่ 2.2 การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม	7
รูปที่ 2.3 การสร้างความสัมพันธ์แบบกลุ่มต่อกลุ่ม	9
รูปที่ 4.1 การออกแบบโครงสร้างฐานข้อมูลในห้องสมุด	17
รูปที่ 4.2 แสดงตารางของฐานข้อมูลที่ใช้ในโปรแกรม	18
รูปที่ 5.1 วินโดวส์ Form	19
รูปที่ 5.2 Object Inspector	20
รูปที่ 5.3 Code Editor	21
รูปที่ 5.4 วินโดวส์ Main	22



บทที่ 1

บทนำ

1.1 ความเป็นมา

ห้องสมุดเป็นแหล่งศึกษาค้นคว้าด้วยตนเองนอกเหนือจากการศึกษาปกติในห้องเรียน เป็นที่รวมของหนังสือประกอบการเรียนและหนังสือต่างๆ ที่มีเนื้อหาออกเหนือไปจากบทเรียน จึงจำเป็นอย่างยิ่งที่ในสถานศึกษาต่างๆ จะต้องมีห้องสมุดเพื่อให้บริการแก่นักเรียนนักศึกษาอย่างพอเพียง ซึ่งในภาควิชาเทคนิคอุตสาหกรรมก็มีห้องสมุดจัดบริการให้กับนักศึกษาที่ศูนย์นนทบุรี จากการสังเกตของผู้จัดทำโครงการพบว่า การให้บริการต่างๆ ของห้องสมุดไม่ว่าจะเป็นการยืมหนังสือ คินหนังสือ การเปลี่ยนแปลงข้อมูลต่างๆ เช่น ข้อมูลหนังสือที่เข้ามาใหม่ ข้อมูลนักศึกษาที่ใช้บริการ ยังใช้การจดบันทึกข้อมูลลงในเอกสารซึ่งเป็นการล่าช้าในการให้บริการและการตรวจสอบข้อมูลต่างๆ เกิดการผิดพลาดได้ง่าย ไม่เหมาะสมกับจำนวนของหนังสือและจำนวนผู้ใช้บริการห้องสมุดของภาคเทคนิคอุตสาหกรรมที่มีเพิ่มมากขึ้นในปัจจุบัน

ดังนั้นผู้จัดทำโครงการจึงได้เกิดแนวความคิดในการพัฒนาโปรแกรมงานบริการห้องสมุดนี้ขึ้นมาเพื่อช่วยเจ้าหน้าที่ห้องสมุดในการบริการการยืม คินหนังสือ บริการเพิ่ม ลบรายชื่อหนังสือและรายชื่อนักศึกษา การค้นหาข้อมูลของหนังสือและข้อมูลของผู้ใช้บริการในห้องสมุด ให้มีความสะดวกรวดเร็วลดความผิดพลาดที่อาจเกิดขึ้นได้ เพื่อเป็นประโยชน์แก่เจ้าหน้าที่ห้องสมุดและนักศึกษาที่มาใช้บริการห้องสมุดของภาคเทคนิคอุตสาหกรรมที่ศูนย์นนทบุรี โดยใช้ เดลไฟล์ เป็นเครื่องมือในการพัฒนาโปรแกรม เนื่องจาก เดลไฟล์ เป็นการเขียนโปรแกรมที่ใช้งานบนวินโดวส์และใช้ระบบการจัดการฐานข้อมูลแบบ Relational ซึ่งเป็นระบบจัดการฐานข้อมูลที่มีประสิทธิภาพและไม่ซับซ้อน

1.2 วัตถุประสงค์ของโครงการ

1 ศึกษาประยุกต์การใช้คอมพิวเตอร์มาใช้กับระบบงานบริการห้องสมุด ซึ่งมีการยืมคืน การเพิ่มลบหนังสือและนักศึกษา และการตรวจสอบ เพื่อให้การบริการห้องสมุดมีประสิทธิภาพและเกิดความรวดเร็ว

2 ศึกษาเกี่ยวกับโปรแกรมที่จะนำมาควบคุมการทำงานโดยใช้ เดลไฟล์

3 ศึกษาการจัดการฐานข้อมูลแบบ Relational Model เพื่อนำมาใช้ในการจัดการข้อมูลต่างๆ ที่จะนำไปใช้ร่วมกับ เดลไฟล์

1.3 ขอบเขตของโครงการ

1. ศึกษาข้อมูลและรายละเอียดเกี่ยวกับงานบริการต่างๆ ในห้องสมุดที่สามารถนำไปพัฒนาโปรแกรมเพื่อช่วยงาน

2. ศึกษาการเขียนโปรแกรมโดยใช้ เดลไฟล์ และระบบฐานข้อมูลแบบ Relational ร่วมกับ SQL

3. ศึกษาวิธีการออกแบบฐานข้อมูลที่จะใช้กับงานบริการห้องสมุด

4. เขียนโปรแกรมบริการสำหรับงานบริการห้องสมุดทั้งหมด

บทที่ 2

ทฤษฎีและหลักการเบื้องต้น

2.1 การทำงานของระบบงานห้องสมุด

ระบบงานห้องสมุด แบ่งงานออกเป็น 3 ส่วนหนึ่งดังนี้

2.1.1 งานบริหาร

2.1.2 งานเทคนิค

2.1.3 งานบริการ

2.1.1 งานฝ่ายบริหาร มีงานหลักๆ 3 งานคือ

1.งานจัดทำค่าของบประมาณ ผู้รับผิดชอบในส่วนนี้คือ เจ้าหน้าที่ฝ่ายบริหาร มีหน้าที่ขอ
งบประมาณที่ห้องสมุดจะได้รับในแต่ละปี เพื่อใช้ในการจัดบริการให้แก่ผู้ใช้ห้องสมุด เช่น ค่าวัสดุตำรา
ค่าอาหารทำการนอกเวลา ค่าประชุม ค่าครุภัณฑ์

2. การจัดซื้อจัดหาหนังสือ , วารสาร เป็นหน้าที่ของผู้บริหารระดับสูงและเจ้าหน้าที่ฝ่ายบริหาร
ในการพิจารณาการจัดซื้อหนังสือ

3.เรื่องการอนุมัติต่างๆ เช่น ขออนุมัติปฏิบัติงานนอกเวลาราชการ

2.1.2 งานเทคนิค

ในระบบงานห้องสมุด สามารถแบ่งงานเทคนิคเป็น 2 ส่วน คือ

1. งานเทคนิคหนังสือ ซึ่งมีงานหลักๆ อยู่ 5 ส่วน คือ

- การลงทะเบียน เป็นหน้าที่ของ เจ้าหน้าที่ลงทะเบียน ซึ่งการลงทะเบียนจะต้องมีการลง ราย
ละเอียดเกี่ยวกับหนังสือที่สำคัญ เช่น ผู้แต่ง , ชื่อหนังสือ , สำนักพิมพ์, ราคาหนังสือ , หรือหนังสือ
ประเภท ได้รับบริจาค , ถ่ายเอกสาร . เอกสารประกอบสิ่งพิมพ์รัฐบาล (งานวิจัย) และวิทยานิพนธ์ มา
ทำการลงทะเบียนโดยแยกเป็น หนังสือ ภาษาไทย และหนังสือ ภาษาต่างประเทศหลังจากนั้นจึงนำ
หนังสือมาประทับตราโดยแยกเป็น หนังสือจากการซื้อ และ หนังสือจากการบริจาค

- การวิเคราะห์เลขหมู่และทำบัตรรายการ รวมถึงการนำบัตรเข้าตู้บัตรรายการ เพื่อให้สมาชิก
เข้าถึงสิ่งพิมพ์ได้สะดวกรวดเร็วยิ่งขึ้น

- งานตรวจสอบความถูกต้อง เป็นหน้าที่ของเจ้าหน้าที่ตรวจสอบความถูกต้อง มีหน้าที่นำ
หนังสือพร้อมเอกสารประกอบจากงานพิมพ์, บัตรครบชุด ,บัตรร่าง , และการแจ้งข้อผิดพลาดเพื่อปรับ
ปรุงแก้ไขให้ถูกต้อง

- งานพิมพ์ เป็นการจัดงานทุกอย่างที่จำเป็นต้องให้เกิดความเรียบร้อย และรวดเร็ว

- การหุ้มปกและซ่อมบำรุง เป็นหน้าที่ของเจ้าหน้าที่หุ้มปกและซ่อมบำรุง มีหน้าที่รักษาสภาพ
สิ่งพิมพ์ที่ชำรุดให้กลับสู่สภาพที่สมบูรณ์พร้อมจะให้บริการ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการอ้างอิงข้อมูลเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. งานเทคนิควารสาร คือการลงทะเบียนวารสาร เป็นหน้าที่ของเจ้าหน้าที่เทคนิควารสาร จัดทำเพื่อให้ทราบว่ามีวารสารใดบ้าง เริ่มรับเมื่อใด รับต่อเนื่องหรือไม่ โดยจะแยกเป็นภาษาไทย และ ภาษาต่างประเทศ ซึ่งจะมีการบันทึกรายละเอียดเกี่ยวกับ ชื่อวารสาร ที่อยู่ ปีที่เท่าใด ฉบับที่เท่าใด ของปีใด เดือนใด

2.1.3 งานบริการ

งานบริการหนังสือ สามารถแบ่งย่อยได้ดังนี้ การยืมหนังสือ ,การคืนหนังสือ ,การจัดชั้นหนังสือ , การจองหนังสือ , การทวงหนังสือ , การรับแจ้งบัตรหนังสือและบัตรสมาชิกหาย, การทำบัตรสมาชิก

2.2 ทฤษฎีการวิเคราะห์และออกแบบระบบ

การออกแบบระบบฐานข้อมูลสามารถแบ่งออกได้เป็น 2 ระดับ คือ

2.2.1 การออกแบบระดับสารสนเทศ (Information - level design) คือ ส่วนที่ศึกษาวิเคราะห์รวบรวมความต้องการของผู้ใช้ไว้

2.2.2 การออกแบบระดับกายภาพ (Physical - level design) เป็นระดับที่เริ่มให้ความสำคัญต่อประสิทธิภาพของระบบ

2.2.1 ระบบสารสนเทศ (Management Information System)

ระบบนี้จะช่วยจัดการกับข้อมูล ช่วยเก็บตัวเลขและช่วยเก็บข้อมูลที่มีความสำคัญต่างๆ ที่เราต้องการ เช่น ชื่อ - สกุล คณะ ที่อยู่ ชั้น เป็นต้น ระบบสารสนเทศอาจจะใช้หรือไม่ใช้คอมพิวเตอร์ก็ได้ กรณีที่ใช้คอมพิวเตอร์ก็มีเหตุผลหลายอย่าง เช่น เราต้องการทราบข้อมูลอย่างรวดเร็วหรือจำนวนสมาชิกที่มีมาก จึงสะดวกในการทำงานมากกว่าแทนที่จะใช้คนหลายๆคน (Manual) เป็นต้น การที่ผู้บริหารทราบข้อมูลต่างๆได้รวดเร็วจะช่วยให้การตัดสินใจอย่างรวดเร็วถูกต้องทำให้วางแผนงานต่างๆได้ล่วงหน้าและทันท่วงที เพราะฉะนั้นการวิเคราะห์ในระดับสารสนเทศจะคำนึงถึงความต้องการของผู้ใช้เป็นหลัก

2.3 วิธีการรวบรวมความต้องการของผู้ใช้

การศึกษา และรวบรวมความต้องการของผู้ใช้นับว่าเป็นขั้นตอนการทำงานที่ยุ่ยากทีเดียว เพราะเป็นงานที่มีความเกี่ยวข้องกับผู้คนมากมายบางครั้งใช้หลักทฤษฎีเพียงอย่างเดียวก็อาจไม่ได้ผลเต็มที่

2.3.1 การใช้แบบสอบถาม การออกแบบสอบถามกระทำได้ 2 วิธี วิธีแรกโดยการส่งแบบสอบถามให้ผู้

ใช้กรอกข้อมูลโดยตรงและเมื่อกรอกเสร็จแล้วก็ส่งคืนมายังผู้ที่จะทำการวิเคราะห์หรือออกแบบ ส่วนวิธีที่สอง

คือ การที่ผู้วิเคราะห์เป็นผู้กรอกแบบสอบถามเองโดยข้อมูลที่ได้จากการศึกษาเอกสาร และรายละเอียดงานต่างๆ รวมทั้งข้อมูลที่ได้จากการสัมภาษณ์ผู้ใช้กลุ่มต่างๆ

แต่ไม่ว่าจะให้ใครเป็นผู้กรอกแบบสอบถามก็ตาม สิ่งที่เราต้องคำนึงถึงคือ ข้อมูลที่ได้รับจากแบบสอบถามควรมีความสมบูรณ์มากที่สุด

2.3.2 การรวบรวมข้อมูลจากรายงานและเอกสารที่ให้อยู่แล้ว ผู้วิเคราะห์จะเริ่มต้นศึกษาระบบจากเอกสาร และ รายงานเหล่านี้ก็นับเป็นการเริ่มต้นที่ดีเพราะจะทำให้ผู้วิเคราะห์มีความรู้ และความเข้าใจพื้นฐานของฟังก์ชันการทำงานในองค์กรนั้นก่อนที่จะได้เริ่มสร้างแบบสอบถาม และสัมภาษณ์กับผู้ใช้

2.4 ระบบฐานข้อมูล

ประกอบด้วยส่วนประกอบสองส่วนคือ ระบบการจัดการฐานข้อมูล (Database Management System (DBMS)) ซึ่งเป็นโปรแกรมที่ใช้เพื่อจัดระเบียบและบำรุงรักษารายการของข้อมูลเหล่านี้ อีกส่วนหนึ่งคือแอปพลิเคชันฐานข้อมูล (database application) เป็นโปรแกรมที่ช่วยดึง ตรวจสอบ และ แก้ไขข้อมูลที่ถูกเก็บอยู่ใน DBMS ทั้งส่วนของ DBMS และแอปพลิเคชันฐานข้อมูลจะทำงานอยู่ในคอมพิวเตอร์เครื่องเดียวกัน ทั้งสองส่วนจะถูกรวมอยู่ในโปรแกรมเดียวกัน

หน้าที่ของระบบจัดการฐานข้อมูล Database Management System (DBMS)

- Data definition เตรียมวิธีการสำหรับการกำหนดและจัดเก็บประชากรของข้อมูล
- Data maintenance บำรุงรักษาประชากรโดยใช้เรคคอร์ดสำหรับรายการแต่ละรายการในประชากร ใช้ฟิลด์เก็บข้อมูลเฉพาะที่อธิบายถึงรายการนั้นๆ
- Data manipulation เตรียมบริการที่ให้ผู้ใ้เพิ่ม แก้ไข ลบ และจัดเรียงข้อมูลในฐานข้อมูล
- Data display อาจจะมีหรือไม่มีก็ได้ เตรียมวิธีการบางอย่างสำหรับการแสดงข้อมูลสำหรับผู้ใ้
- Data Integrity เตรียมวิธีการหนึ่งหรือหลายวิธี เพื่อให้มั่นใจว่าข้อมูลนั้นมีความถูกต้อง

โมเดล DBMS

ระบบการจัดการฐานข้อมูลที่มีอยู่ในปัจจุบันนี้สามารถที่จะจัดได้เป็นสี่กลุ่มที่แตกต่างกัน หรือที่เรียกกันทั่วไปว่า โมเดล (model) คือระบบ File Management, ระบบ Hierarchical Database, ระบบ Net Work Database และระบบ Relational Database ประเภทใหม่ของ DBMS ที่เรียกว่า

เอกสารนี้ Object - Oriented Database (OODBMS) ซึ่งมีการใช้กันอย่างกว้างขวางไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 ระบบ File Management

File Management System (FMS) เป็นโมเดลฐานข้อมูลที่ง่ายที่สุดและเป็นเพียงระบบเดียวที่อธิบายถึงว่าข้อมูลถูกจัดเก็บลงบนดิสก์อย่างไร ในโมเดล FMS ไฟล์แต่ละไฟล์หรือรายงานข้อมูลแต่ละรายการจะถูกจัดเก็บเรียงตามลำดับบนดิสก์ในไฟล์ขนาดใหญ่ไฟล์หนึ่ง การที่จะค้นหาไฟล์แต่ละไฟล์นั้น แอปพลิเคชันจะต้องเริ่มต้นจากจุดเริ่มต้นและตรวจสอบแต่ละรายการจนกระทั่งพบอันที่ตรงกัน ข้อดีของ FMS มีเพียงข้อเดียวคือความง่าย แต่มีข้อเสียหลายประการดังนี้คือ

- ไม่มีเครื่องหมายที่บ่งชี้ถึงความสัมพันธ์ระหว่างรายการที่แตกต่างกันนอกเหนือไปจากลำดับของการจัดเก็บ โปรแกรมเมอร์หรือผู้ใช้จะต้องรู้อย่างชัดเจนว่าข้อมูลนั้นถูกจัดการอย่างไร นั่นก็คือต้องมีความรู้ว่าจะระบบปฏิบัติการของเครื่องคอมพิวเตอร์นั้นจัดเก็บข้อมูลลงบนดิสก์ในทางฟิสิกส์อย่างไร ซึ่งทำให้การเข้าถึงข้อมูลนั้นเป็นเรื่องยาก
- เกิดปัญหาในเรื่องความถูกต้องของข้อมูล ค่าของไฟล์ทั้งหมดจะต้องถูกตรวจสอบโดยโปรแกรมแอปพลิเคชันก่อนที่จะถูกจัดเก็บลงไปในดิสก์ ฐานข้อมูลเดียวกันสามารถถูกเข้าถึงได้โดยแอปพลิเคชันที่แตกต่างไป
- ไม่มีวิธีการสำหรับค้นหาเรคอร์ดเฉพาะไฟล์ใดไฟล์หนึ่งโดยเร็ว การค้นหาทุกครั้งนั้นจะต้องเริ่มต้นจากจุดเริ่มต้นของไฟล์ พิจารณาแต่ละเรคอร์ดและทุกๆ เรคอร์ด
- ไม่สามารถที่จะเปลี่ยนแปลงโครงสร้างของฐานข้อมูลได้โดยง่าย ซึ่งเป็นข้อเสียมากที่สุด

2.4.2 ระบบ Hierarchical Database

ในโมเดลนี้ ข้อมูลถูกจัดระเบียบในลักษณะโครงสร้างแบบต้นไม้หรือคล้ายคลึงกับแผนผังองค์กร เป็นการกำหนดความสัมพันธ์แบบ "แพเรนต์ - ไชลด์" (parent child) จะเห็นได้ว่าโครงสร้างแบบระดับชั้นมีข้อดีกว่าระบบ FMS ในการกำหนดความสัมพันธ์แบบ one-to-many ทำการค้นหาข้อมูลได้ง่ายและเร็วขึ้น การค้นหาข้อมูลที่อยู่ในไฟล์ใดไม่จำเป็นต้องค้นหาทั้งไฟล์เพียงแต่พิจารณาคำขอ และแบ่ง คำขอนั้นให้เล็กลงเป็นส่วนประกอบของมันและเคลื่อนที่ไปตามกิ่งก้าน แต่ระบบ HDS ก็มีข้อเสียดังนี้

- โครงสร้างเริ่มต้นของฐานข้อมูลซึ่งกำหนดได้โดยอิสระ ทำให้การเปลี่ยนแปลงความสัมพันธ์แบบแพเรนต์ - ไชลด์จะทำได้ก็ต่อเมื่อต้องออกแบบโครงสร้างใหม่ทั้งหมด
- การกำหนดความสัมพันธ์แบบ many-to-many ทำได้ยาก

2.4.3 ระบบ Network Database

โมเดล Network เป็นเพียงแค่การอธิบายถึงฐานข้อมูลที่มีความสัมพันธ์แบบ many-to-many ความสัมพันธ์ระหว่างรายการข้อมูลที่แตกต่างกันมักถูกอ้างถึงโดยใช้คำว่า "เซต" เพื่อแยกความแตกต่างจากความสัมพันธ์เพเรนต์ - ไชลด์ที่กำหนดโดยโมเดล Hierarchical แต่ก็มีข้อเสียเช่นเดียวกับฐานข้อมูล Hierarchical การออกแบบเบื้องต้นสำหรับฐานข้อมูลนั้นสามารถกำหนดได้อย่างอิสระ หลังจากติดตั้งแล้ว การเปลี่ยนแปลงใดๆกับเซตที่แตกต่างกันทำให้ต้องสร้างโครงสร้างใหม่ทั้งหมด

2.4.4 โมเดล Relational Database

โมเดลรีเลชันแนลได้ยกเลิกความคิดในเรื่องความสัมพันธ์เพเรนต์-ไชลด์ ระหว่างรายการข้อมูลที่แตกต่างกัน โดยข้อมูลจะถูกจัดเป็นเซตทางคณิตศาสตร์ในโครงสร้างของตาราง ในระบบ RDM ฟิวด์ข้อมูลแต่ละตัวจะเป็นคอลัมน์ในตารางและแต่ละ เรคอร์ด จะกลายเป็นแต่ละแถวในตาราง โมเดลรีเลชันแนลมีข้อดีที่ชัดเจนกว่าโมเดลอื่น คือ ความยืดหยุ่นในการกล่าวถึงความสัมพันธ์ระหว่างรายการข้อมูลที่แตกต่างกัน โปรแกรมเมอร์กำหนดฐานข้อมูลด้วยการสร้างตารางและตัดสินใจว่า คอลัมน์ใดที่ตารางสัมพันธ์กัน การเปลี่ยนแปลงโครงสร้างของฐานข้อมูลโดยการลบหรือเพิ่มคอลัมน์ออกจากตารางจะไม่มีผลต่อตารางอื่นๆ และมีความถูกต้องของข้อมูลเนื่องจากสามารถป้องกันการเข้าถึงข้อมูลด้วยวิธีการอื่นๆ นอกเหนือจากการจัดการคิวรีโดย DBMS ได้อย่างสมบูรณ์ โดยจะต้องมีคำศัพท์ที่ควรรู้คือ รีเลชัน หมายถึง ตาราง , ทัพเพิล หมายถึง แถว , attribute หมายถึง คอลัมน์ , entity หมายถึง ชื่อตาราง , คีย์หลัก (primary key) หมายถึง attribute 1 ตัวหรือมากกว่า 1 ตัวก็ได้ที่สามารถใช้เป็นตัวเจาะจงบอกว่าการกำลังอ้างอิงถึงทัพเพิลอันไหน , คีย์นอก หมายถึง attribute ของรีเลชันอันหนึ่งที่ซ้ำกับ attribute ในอีกรีเลชันหนึ่ง attribute ที่ว่านี้จะใช้ในการจับคู่ข้อมูลในรีเลชันทั้งสองนี้

2.5 การจัดเก็บข้อมูลภายในฐานข้อมูล จะมีอยู่ด้วยกัน 3 แบบคือ

2.5.1 การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบหนึ่งต่อหนึ่ง ยกตัวอย่างสมมติให้บริษัทออกกฎมาว่าพนักงานชายแต่ละคนมีสิทธิขายของให้แก่ลูกค้าเพียงคนเดียวเท่านั้นซึ่งเป็นการแสดงถึงความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างพนักงานชาย และลูกค้า ซึ่งจะได้ตารางออกมาดังแสดงในรูป

พนักงานชาย

รหัสพนักงานชาย	ชื่อพนักงานชาย	รหัสลูกค้า
3	ธนา	124
6	มณีรัตน์	256
12	ประจวบ	311

ลูกค้า

รหัสลูกค้า	ชื่อลูกค้า	รหัสพนักงานขาย
124	เจลิยว	3
256	มุสตี	6
311	ยงยุทธ	12

รูปที่ 2.1 การสร้างความสัมพันธ์แบบหนึ่งต่อหนึ่ง

จากรูปในตารางของพนักงานขายก็มีรหัสพนักงานขายเป็นคีย์หลัก และมีรหัสลูกค้าเป็นคีย์นอก ตารางลูกค้าก็มีรหัสลูกค้าเป็นคีย์หลักและมีรหัสพนักงานขายเป็นคีย์นอกการสร้างความสัมพันธ์ระหว่าง entity หนึ่งดังแสดงในรูปโดยจะเห็นว่าคีย์นอกจะปรากฏอยู่ในแต่ละตารางเพียงครั้งเดียวเท่านั้น

2.5.2 การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบหนึ่งต่อกลุ่ม ยกตัวอย่างบริษัทได้เปลี่ยนกฎให้พนักงานขายแต่ละคนติดต่อลูกค้าได้มากกว่า 1 คน แต่ลูกค้าแต่ละคนต้องซื้อของจากพนักงานขายเพียงคนเดียวเท่านั้นซึ่งเท่ากับเป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม ในลักษณะการณีนี้นี้เราสามารถใช้อีตารางหนึ่งตาราง สำหรับเก็บข้อมูลของพนักงานขาย และเก็บข้อมูลของลูกค้าไว้ในอีตารางหนึ่งโดยในตารางนี้จะเพิ่มอีกหนึ่งคอลัมน์สำหรับจัดเก็บค่ารหัสพนักงานขายที่ลูกค้าคนนั้นใช้บริการอยู่ โดยที่ค่าของรหัสพนักงานขายนี้ก็จะเป็นคีย์หลักในตารางของพนักงานขายดังแสดงไว้ในรูป

ลูกค้า

รหัสลูกค้า	ชื่อ	รหัสพนักงานขาย
124	เจลิยว	3
256	มุสตี	6
311	ยงยุทธ	12
315	พิชัย	6
405	นันทวัน	12
412	ประเดิม	3
522	มณี	12
567	วิไล	6
587	วัชร	6
622	มณี	3

พนักงานขาย

รหัสพนักงานขาย	ชื่อพนักงานขาย
3	ธนา
6	มนีรัตน์
12	ประจวบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้รูปที่ 2.2 การสร้างความสัมพันธ์แบบหนึ่งต่อกลุ่ม ทุกครั้งที่มีการนำไปใช้

จากตัวอย่างในรูปจะเห็นว่าข้อมูลของลูกค้าแต่ละคนจะปรากฏในแถวของตารางของลูกค้าเพียงแถวเดียวเท่านั้นและค่าของรหัสพนักงานขายในแต่ละแถวของตารางนี้จะมีเพียงค่าเดียว ซึ่งเป็นการยืนยันว่าลูกค้าแต่ละคนมีสิทธิติดต่อกับพนักงานขายคนเดียว ส่วนในตารางของพนักงานขายนั้นไม่มีข้อมูลเกี่ยวกับลูกค้าเลยดังนั้นถ้าเราต้องการทราบว่า พนักงานขายคนใดคนหนึ่งมีใครเป็นลูกค้าอยู่บ้างก็ต้องไล่ดูรหัสของพนักงานคนนั้นในตารางลูกค้าเอง ซึ่งปรากฏอยู่มากกว่าหนึ่งแถวก็ได้เป็นการแสดงว่าพนักงานแต่ละคนมีสิทธิมีลูกค้าได้มากกว่าหนึ่งคน ตัวอย่างเช่น พนักงานหมายเลข 3 (คือธนา) ให้บริการลูกค้าอยู่ 3 คน คือลูกค้าหมายเลข 124 ,412 และ 622 แต่ลูกค้าแต่ละคนเช่น ประเดิม (หมายเลข 412) จะใช้บริการของธนา (หมายเลข 3) เพียงคนเดียว

2.5.3 การจัดเก็บข้อมูลสำหรับความสัมพันธ์แบบกลุ่มต่อกลุ่ม ยกตัวอย่างบริษัทมีนโยบายทำให้ลูกค้าสามารถสั่งซื้อสินค้าได้มากกว่า 1 อย่างในการสั่งซื้อแต่ละครั้ง และในทำนองเดียวกันก็อนุญาตให้สินค้า 1 อย่างปรากฏอยู่ในรายการสั่งซื้อได้มากกว่า 1 ครั้งจึงเห็นได้ว่าลักษณะความสัมพันธ์ระหว่างการสั่ง และสินค้าเป็นแบบกลุ่มต่อกลุ่มในกรณีเช่นนี้เราต้องสร้างตารางขึ้นมาเป็นพิเศษเพื่อบรรจุค่าคีย์ของตารางทั้งสองนี้อันได้แก่รหัสการสั่งและรหัสสินค้า และตารางนี้ยังประกอบด้วยข้อมูลอื่นๆที่มีค่าขึ้นอยู่กับค่าของทั้งรหัสการสั่งและรหัสสินค้า เช่น จำนวนสั่งและราคา เป็นต้น เราขอตั้งชื่อตารางที่ 3 นี้ว่า บัญชีการสั่ง ดังแสดงไว้ในรูป

การสั่ง

รหัสการสั่ง	วันที่สั่ง	รหัสลูกค้า
ORDER_NUMBER	DATE	CUSTOMER_NUMBER
12489	020931	124
12491	020931	311
12494	040931	315
12495	040931	256
12498	050931	522
12500	050931	124
12504	050931	522

สินค้า

รหัสสินค้า	ชื่อสินค้า
AX12	เตารีด
AZ52	สเกด
BA74	ลูกบอล
BH22	เตาปิ้งขนมปัง
BT04	เตา
BZ66	เครื่องซักผ้า
CA14	เตาอั้งโล่
CB03	จักรยาน
CX11	เครื่องปั่น
CZ81	ตุ้มน้ำหนัก

บัญชีการสั่ง

รหัสการสั่ง	รหัสสินค้า	จำนวนสั่ง	ราคา
12489	AX12	11	280
12491	BT04	1	800
12491	BZ66	1	600
12494	CB03	4	3500
12495	CX11	2	1040
12498	AZ52	2	440
12498	BA74	4	80
12500	BT04	1	8040
12504	CZ81	2	2160

รูปที่ 2.3 การสร้างความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ในการค้นหาว่า การสั่งตามหมายเลขที่เราสนใจนั้นมีการสั่งสินค้าอะไรบ้างก็เริ่มด้วยการดูข้อมูลในตารางบัญชีการสั่งโดยดูในแถวที่มีรหัสการสั่งที่สนใจปรากฏอยู่ซึ่งเราจะได้รับรหัสของสินค้าที่อยู่ในแถวเดียวกันนั้น และเราสามารถไต่รหัสสินค้านี้ไปดูข้อมูลของสินค้าจากรายสินค้าได้ ของให้สังเกตว่าจำนวนแถวของข้อมูลที่มีรหัสการสั่งตามที่เราสนใจนั้นอาจมีมากกว่า 1 แถวก็ได้

ตัวอย่างเช่น ถ้าอยากทราบว่ายางยુทธ์ (หมายเลข 311) สั่งสินค้าอะไรบ้างเมื่อวันที่ 2 กันยายน 2531 ก็จะเริ่มที่ตารางการสั่ง ได้รับรหัสการสั่งคือ 12491 และเราก็ไต่รหัสนี้เข้าไปดูในตารางบัญชีการสั่ง ซึ่งจะเจอว่าเป็นการสั่งสินค้า 2 ชนิดคือ BT04 และ BZ66 ซึ่งถ้ามาไล่ดูในตารางสินค้าก็จะทราบว่า เป็นการสั่ง เต้า และ เครื่องซักผ้า และในทางกลับกัน การดูว่าสินค้าที่เราสนใจนั้นมีข้อมูลเกี่ยวกับการสั่งอย่างไรก็ใช้วิธีการเดียวกัน และก็เช่นเดียวกันว่ารหัสสินค้าที่ปรากฏในตารางบัญชีการสั่งอาจมีมากกว่า 1 อย่าง เช่น ถ้าอยากทราบว่า เต้า อันได้แก่ BT04 มาจากรหัสการสั่งเบอร์อะไรบ้างก็ย้อนไปดูในตารางบัญชีการสั่ง ซึ่งจะพบว่า BT04 ปรากฏอยู่ใน 2 แถว คือ รหัสการสั่ง 12491 (สั่งโดยลูกค้ายหมายเลข 311 ได้แก่ยางยुทธ์) และรหัสการสั่ง 12500 (สั่งโดยลูกค้ายหมายเลข 124 ได้แก่เจลิยว)

ประโยชน์จากการ ประมวลผลด้วยฐานข้อมูลที่มีประสิทธิภาพ

1. ลดความซ้ำซ้อนของฐานข้อมูล
2. สามารถหลีกเลี่ยงความขัดแย้งของฐานข้อมูลได้ในระดับหนึ่ง
3. สามารถใช้ข้อมูลร่วมกันได้
4. สามารถควบคุมความเป็นมาตรฐานได้
5. สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้
6. สามารถควบคุมความคงสภาพของข้อมูลได้
7. เกิดความเป็นอิสระของข้อมูล



บทที่ 3

เครื่องมือที่ใช้ในการพัฒนาโปรแกรมประยุกต์สำหรับฐานข้อมูล

3.1 SQL

SQL เป็นภาษาสำหรับการใช้ข้อมูลที่เป็นที่นิยมสำหรับโมเดลเชิงสัมพันธ์ คุณสมบัติสำคัญของภาษาเหล่านี้ คือสามารถใช้เรียกดูข้อมูลตามที่ต้องการได้ สามารถใช้ในการเปลี่ยนแปลงเพิ่มเติมและลบข้อมูลออกจากระบบ นอกจากนี้ยังสามารถใช้ในการสร้างระบบฐานข้อมูล การใช้ SQL กระทำได้ 3 วิธีได้แก่

- 1 ออกคำสั่งแบบออนไลน์ กล่าวคือ ผู้ใช้สามารถพิมพ์ประโยคคำสั่งผ่านทางเทอร์มินัลโดยที่คำสั่งเหล่านี้จะถูกปฏิบัติงานโดยทันที
- 2 ส่งคำสั่งในลักษณะงานออฟไลน์ หรืองาน batch ลักษณะการใช้งานประเภทนี้เหมาะกับการสร้างรายงาน หรือประเภทของงานที่ไม่จำเป็นต้องการทราบผลโดยทันที
- 3 สอดแทรกประโยคคำสั่งไว้ในโปรแกรมประยุกต์ที่เขียนขึ้นมาสำหรับการใช้ระบบฐานข้อมูล

3.2 การใช้ SQL จัดการข้อมูล

การเรียกดูข้อมูล ประโยคคำสั่งสำหรับการเรียกดูข้อมูลได้แก่ คำสั่ง SELECT ซึ่งจะมีรูปแบบของคำสั่งดังนี้ SELECT...FROM...WHERE.....ORDER BY...

ซึ่งสามารถระบุส่วนต่างๆในคำสั่งดังได้ดังนี้

ส่วนหลัง SELECT ใช้ระบุ คอลัมน์หรือกลุ่มของคอลัมน์ที่เราต้องการดูข้อมูล

ส่วนหลัง FROM ใช้ระบุ ชื่อตารางที่เราต้องการดูข้อมูล

ส่วนหลัง WHERE ใช้ระบุ เงื่อนไขของข้อมูลที่เราสนใจซึ่งอาจจะมีหรือไม่มีก็ได้

ส่วนหลัง ORDER BY ใช้ระบุ วิธีการเรียงลำดับการแสดงผลข้อมูล ซึ่งอาจจะมีหรือไม่มีก็ได้

ตัวอย่างเช่น

```
SELECT รหัสนักศึกษา,คณะ
```

```
FROM สมาชิก
```

```
WHERE ชื่อนักศึกษา = 'พิระพล'
```

การใช้ข้อมูล คำสั่งสำหรับการใช้ข้อมูลได้แก่กลุ่มคำสั่งที่ใช้ในการเพิ่มเติมแก้ไข และลบข้อมูลออกจากตาราง ซึ่งตารางเหล่านี้จะต้องได้รับการสร้างไว้แล้วด้วยคำสั่งสำหรับการนิยามข้อมูล

การเพิ่มเติมข้อมูล---คำสั่ง INSERT

คำสั่ง INSERT ใช้ในการเพิ่มเติมข้อมูลแถวใหม่ลงในตาราง โดยอาจจะระบุข้อมูลไว้ในคำสั่งเลย หรือ

ดึงข้อมูลออกมาจากตารางอื่นก็ได้ โดยบังคับไว้ว่า ข้อมูลที่จะเติมลงไปนั้นจะต้องมีข้อมูลในแต่ละคอลัมน์ตรงตามคอลัมน์ในตารางที่จะเพิ่มเติมลงไปด้วย รูปแบบคำสั่ง INSERT เป็นดังนี้

```
INSERT INTO ชื่อตาราง
VALUES (ค่าที่ต้องการใส่ในคอลัมน์ต่างๆ) หรือ
INSERT INTO ชื่อตาราง : SELECT.....
```

ตัวอย่างเช่น

```
INSERT INTO หนังสือ
VALUES ("window 95",NULL,"6")
```

การแก้ไขข้อมูล--คำสั่ง UPDATE

คำสั่ง UPDATE ใช้ในการเปลี่ยนแปลงแก้ไขค่าของคอลัมน์ที่ต้องการเฉพาะในแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้ โดยเราสามารถแก้ไขได้มากกว่าหนึ่งคอลัมน์ในคำสั่งเดียวกัน รูปแบบของคำสั่งคือ

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

ตัวอย่างเช่น

```
UPDATE สมาชิก
SET ชื่อนักศึกษา = "พีระพล"
WHERE รหัสนักศึกษา = 39012021;
```

การลบข้อมูล--คำสั่ง DELETE

คำสั่ง DELETE ใช้ในการลบแถวของข้อมูลที่สอดคล้องกับเงื่อนไขที่ระบุออกจากตาราง โดยมีรูปแบบการออกคำสั่งคือ

```
DELETE ชื่อตาราง
WHERE เงื่อนไข
```

ซึ่งถ้าผู้ใช้ไม่ได้ WHERE เอาไว้ก็จะมีผลให้ลบข้อมูลทุกแถวออกจากตารางที่ระบุ

ตัวอย่างเช่น

```
DELETE สมาชิกห้อง N
WHERE ลำดับ =7 AND ลำดับ= 11
```

ฟังก์ชันต่างๆ (Built-in Function)---COUNT,SUM ฯลฯ

คำสั่งที่ใช้ในการเรียกดูข้อมูลมีอยู่เพียงคำสั่งเดียวคือ คำสั่ง SELECT ฟังก์ชันที่จะกล่าวเพิ่มเติมในที่นี้จะให้ความคล่องตัวในการเรียกดูข้อมูลเป็นอันมาก ฟังก์ชันดังกล่าวได้แก่

COUNT ใช้ในการนับจำนวน
 SUM รวมค่าในคอลัมน์ที่กำหนด
 AVG แสดงค่าเฉลี่ยของคอลัมน์ที่กำหนด
 MAX แสดงค่ามากที่สุดของข้อมูลในคอลัมน์ที่กำหนด
 MIN แสดงค่าน้อยที่สุดของข้อมูลในคอลัมน์ที่กำหนด

โดยมีรูปแบบการออกคำสั่งคือ

```
SELECT ชื่อฟังก์ชัน (ชื่อคอลัมน์)
FROM ชื่อตาราง
WHERE เงื่อนไข
```

ตัวอย่างเช่น

```
SELECT SUM ( รหัสนักศึกษา )
FROM ตารางนักศึกษา
```

คำสั่งสำหรับนิยามตาราง---คำสั่ง CREATE TABLE

การสร้างตารางในระบบฐานข้อมูลกระทำได้โดยการใช้คำสั่ง CREATE TABLE ซึ่งในคำสั่งนี้ เราต้องใส่ชื่อของตารางพร้อมทั้งชื่อคอลัมน์ ชนิด และขนาดของข้อมูลของแต่ละคอลัมน์ ตัวอย่างเช่น ถ้าเราต้องการสร้าง Table ชื่อ หนังสือ โดยมี 3 คอลัมน์คือ หัวเรื่อง,ราคา และ ครั้งที่พิมพ์

```
CREARE TABLE หนังสือ
(หัวเรื่อง CHAR (20),
ราคา MONEY (6.2),
ครั้งที่พิมพ์ CHAR (3)
```

คำสั่งนิยามวิวของผู้ใช้---คำสั่ง CREATE VIEW

ผู้ใช้งานข้อมูลแต่ละคนสามารถมองเห็นระบบในลักษณะที่ต่างกันไป เราเรียกเค้าร่างของระบบที่ผู้ใช้แต่ละคนเห็นว่า " วิว " รูปแบบของการใช้คำสั่ง CREATE VIEW คือ

```
CREATE VIEW ชื่อวิว
```

```
[ ( ชื่อคอลัมน์1,ชื่อคอลัมน์2,...) ]
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการเรียงขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ไปยังผู้อื่นโดยไม่ได้รับอนุญาตจากผู้ถือลิขสิทธิ์เอกสารทุกครั้งที่มีการนำไปใช้

```
AS SELECT ประโยคคำสั่ง SELECT
```

ซึ่งเป็นการสร้างตารางขึ้นมาใหม่ตามชื่อที่ระบุ อันประกอบไปด้วยคอลัมน์ และแถวของข้อมูลตามที่ได้จากประโยค SELECT ตารางที่สร้างขึ้นมานี้ถ้าไม่ได้ระบุชื่อคอลัมน์เอาไว้ก็จะมีชื่อคอลัมน์ตรงตามตารางเดิมที่ระบุไว้ในส่วนของประโยค SELECT

คำสั่งสำหรับเพิ่มคอลัมน์ใหม่--คำสั่ง ALTER TABLE

เป็นคำสั่งที่สามารถเพิ่มเติมฟิลด์ (หรือคอลัมน์) ใหม่เข้าไปในตารางได้โดยคอลัมน์ใหม่จะมีค่าเริ่มต้นเป็นค่าว่าง รูปแบบของคำสั่งคือ

```
ALTER TABLE ชื่อตาราง
ADD ชื่อคอลัมน์ ชนิดและขนาดของข้อมูล
```

ตัวอย่างเช่น

```
ALTER TABLE คำปรับ
ADD BOOK_TYPE CHAR ( 20 )
```

บทที่ 4

รายละเอียดของโปรแกรม งานบริการห้องสมุด

4.1 โปรแกรม เดลไฟล์

งานบริการห้องสมุดนี้พัฒนาขึ้นโดยใช้โปรแกรม เดลไฟล์ ซึ่งเป็นเครื่องมือในการพัฒนาโปรแกรมประยุกต์ใช้งาน เป็นการเขียนโปรแกรมบนวินโดวส์ ซึ่งความแตกต่างระหว่างการเขียนโปรแกรมบนวินโดวส์กับการเขียนโปรแกรมบนดอสนั้นคือ การเขียนโปรแกรมบนดอสนั้นเป็นเรื่องที่ซับซ้อนและยุ่งยากในการสร้างและใช้งาน ยกตัวอย่างเช่น ภาษาเบสิก, ภาษาปาสคาล การจะสร้างเมนูขึ้นมาสักเมนูหนึ่งจะต้องป้อนโปรแกรมกำหนดขนาดของส่วนประกอบ, ตำแหน่งของเมนู, กำหนดคุณลักษณะทั้งหมดซึ่งเป็นเรื่องที่ยุ่งยากมากถ้าโปรแกรมมีขนาดใหญ่ ต่างจากการใช้โปรแกรมแบบวิซวล เช่น เดลไฟล์ การกำหนดปุ่มกด (Button) , ช่องป้อนข้อความ (Edit) หรือส่วนประกอบต่างๆ ที่ต้องการให้มีในโปรแกรมสามารถกำหนดได้โดยตรงจากพาเลตต์คอมโพเนนต์และเปลี่ยนแปลงขนาดหรือตำแหน่งของส่วนต่างๆ ได้ด้วยการลากเมาส์ การกำหนดคุณลักษณะของส่วนประกอบต่างๆ ที่สร้างขึ้นมาเป็นโปรแกรม สามารถกำหนดได้โดยตรงจากเพจหรือฟอรัลตี้ และส่วนสำคัญที่โปรแกรมบนวินโดวส์แตกต่างจากโปรแกรมบนดอสคือการเขียนโปรแกรมบนดอสจะต้องเขียนโปรแกรมโดยกำหนดคำสั่งแบบเรียงลำดับขึ้นมาก่อน เช่น เมนูหลักในงานบริการห้องสมุด เมื่อเลื่อนแถบสีเลือกส่วนเจ้าหน้าที่ ขึ้นต่อไปคือส่วนการป้อน Pass Word จะ ปรากฏขึ้น ถ้าป้อน Pass Word ถูก ให้ไปที่เมนูยืมคืน ถ้าป้อนไม่ถูกให้ป้อนใหม่ดังนี้เป็นขั้นตอนตามลำดับ การเปลี่ยนแปลงต่างๆ ที่เกิดขึ้นจะเป็นไปตามลำดับขั้นของการโปรแกรมที่เขียนไว้อย่างยุ่งยากและยาวหลายบรรทัด แต่การทำงานของโปรแกรมบนวินโดวส์เมื่อโปรแกรมถูกโหลดเข้าไปในเมโมรี่และเริ่มทำงานในส่วนเริ่มต้นโปรแกรมจะไม่มีขั้นตอนต่อไป ไม่มีอะไรเกิดขึ้นจนกว่าจะมีอีเวนต์เกิดขึ้น (Event : เหตุการณ์ต่างๆ เช่น การคลิกเมาส์ การลากออบเจกต์) จากนั้นโปรแกรมจึงจะตอบสนองอีเวนต์เหล่านั้น และไม่ได้ใช้ชุดคำสั่งในการควบคุม แต่จะเปลี่ยนเป็นขั้นตอนต่างๆ ที่กำหนดไว้เพื่อตอบสนองอีเวนต์สำคัญๆ ที่เกิดขึ้น จากตัวอย่างเดิมในเมนูหลักของงานบริการห้องสมุด เมื่อคลิกที่ปุ่มเจ้าหน้าที่ซึ่งก็คือ Button2 อีเวนต์ Button2Click จะเป็นตัวกำหนดว่าเมื่อคลิกเมาส์ที่ Button2 แล้วจะเกิดการเปลี่ยนแปลงใดต่อไป ในที่นี้จะกำหนดให้เมื่อคลิกที่ Button2 แล้ว Form การป้อน Pass Word จะปรากฏขึ้น เช่นเดียวกับ Form เมนูหลัก การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นกับ Form Pass Word หลังจากการป้อน Pass Word ลงในช่องว่าง (Edit1) จะถูกกำหนดโดยอีเวนต์ Button1Click โดยกำหนดคำสั่งถ้าป้อน Pass Word ถูกต้องก็ให้เปิด Form เมนูย่อยที่กำหนดไว้ขึ้นมา ถ้าการป้อน Pass Word ไม่ถูกต้องก็จะไม่เกิดการเปลี่ยนแปลงใดๆ จะเห็นได้อย่างชัดเจนว่าการทำงานของโปรแกรมครบถ้วนสมบูรณ์โดยมีการเขียนคำสั่งเพียงเล็กน้อยเท่านั้น

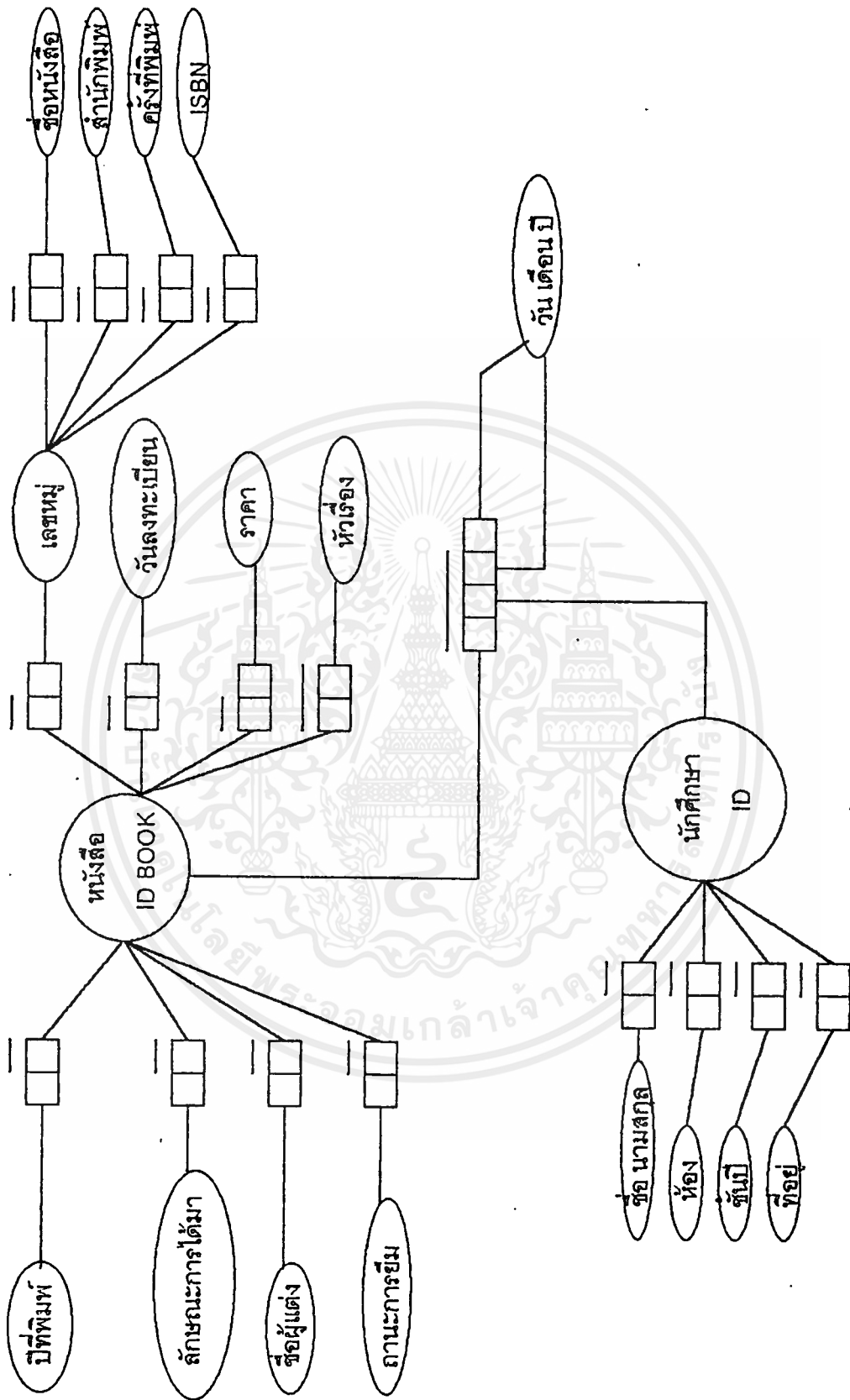
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การออกแบบฐานข้อมูล

เดลไฟล์ ใช้ระบบจัดการฐานข้อมูลแบบ Relational นั่นคือข้อมูลถูกจัดให้เป็นเซตในทางคณิตศาสตร์อยู่ในรูปของตาราง ฟิลด์ข้อมูลแต่ละตัวจะเป็นคอลัมน์ในตารางและแต่ละเรคอร์ดจะเป็นแต่ละแถวในตาราง ข้อดีของการจัดการฐานข้อมูลแบบ Relational กับระบบจัดการแบบอื่นคือมีความยืดหยุ่นในการบ่งบอกความสัมพันธ์ระหว่างรายการข้อมูลที่แตกต่างกันและง่ายต่อการเข้าใจ การเปลี่ยนแปลงโครงสร้างของฐานข้อมูลสามารถทำได้ง่ายโดยการเพิ่มหรือลบคอลัมน์ออกจากตาราง ซึ่งจะไม่ส่งผลต่อตารางอื่นและการที่ไม่จำเป็นต้องสร้างโครงสร้างของฐานข้อมูลใหม่ทั้งหมดเมื่อมีการเปลี่ยนแปลงยังทำให้เพิ่มการรักษาความถูกต้องของข้อมูลอีกด้วย หลังจากที่เรารู้วิเคราะห์และรวบรวมเอาความต้องการของผู้ใช้แล้ว เราก็ต้องรวบรวมเอากฎเกณฑ์และข้อบังคับต่างๆเอาไว้ด้วยเพื่อที่จะผ่านขั้นตอนการออกแบบซึ่งมีอยู่ 5 ขั้นตอนคือ

1. เปลี่ยนรูปแบบของความต้องการให้อยู่ในลักษณะของรีเลชัน
2. นอร์มัลไลซ์รีเลชัน
3. กำหนดฟิลด์ที่จะเป็นคีย์ต่างๆและคุณสมบัติของคีย์แต่ละตัว
4. พิจารณาข้อจำกัดและกฎเกณฑ์อื่นๆ
5. นำผลที่ได้จากการออกแบบในขั้นตอนแรกมาผนวกกัน

จากขั้นตอนต่างๆที่ได้กล่าวไปแล้วเราสามารถที่จะออกแบบโครงสร้างฐานข้อมูลในห้องสมุดซึ่งแสดงถึงการสร้างความสัมพันธ์จากความต้องการของผู้ใช้เพื่อที่จะได้นำไปสร้างตารางของฐานข้อมูลซึ่งแสดงไว้ดังแสดงในรูป



รูปที่ 4.1 การออกแบบโครงสร้างข้อมูลในท้องถิ่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง BOOK

เลขทะเบียน	เลขหมู่	ราคา	สถานะการยืม	ปีที่พิมพ์	ลักษณะที่ได้มา	ชื่อผู้แต่ง
ID BOOK	CALL	PRICE	STATUS	YEAR	GET_TYPE	WRITE

ตารางเรียกหนังสือ

เลขหมู่	ชื่อหนังสือ	สำนักพิมพ์	ครั้งที่พิมพ์	รหัสมาตรฐาน	หัวเรื่อง
CALL	BOOKNAME	PUBLISH	EDITION	ISBN	TITLE

ตารางการยืมคืน

รหัสนักศึกษา	เลขทะเบียนหนังสือ	วันที่ยืม	วันที่คืน
ID	ID BOOK	BORROW	RETRUN

ตารางนักศึกษา

รหัสนักศึกษา	ชื่อนามสกุล	ห้อง	ชั้นปี	ที่อยู่
ID	NAME	ROOM	LAYER	ADDRESS

ตารางหัวเรื่อง

เลขทะเบียนหนังสือ	หัวเรื่อง
-------------------	-----------

รูปที่ 4.2 แสดงตารางของฐานข้อมูลที่ใช้ในโปรแกรม



บทที่ 5

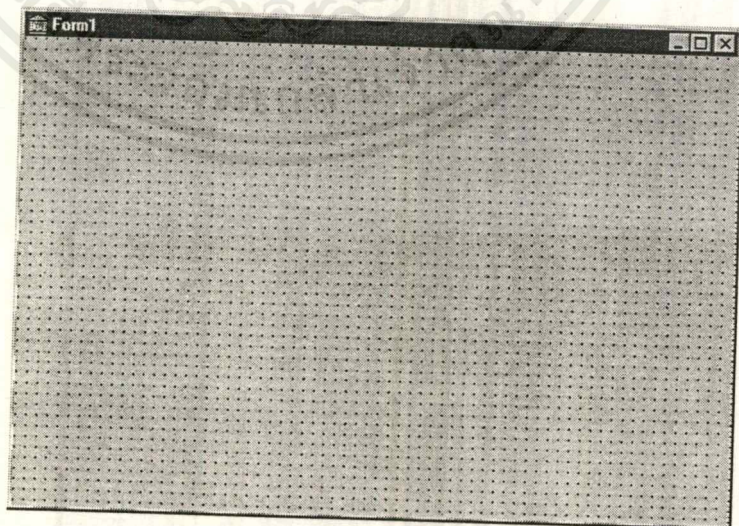
การใช้งานโปรแกรมเดลไฟล์

วินโดวส์หลักของเดลไฟล์ประกอบไปด้วย

- วินโดวส์ Form
- วินโดวส์ Object Inspector
- วินโดวส์ Code Editor
- วินโดวส์ Main

5.1 วินโดวส์ Form

วินโดวส์ Form1 ที่มีจุดๆคือ วินโดวส์ Form ที่ถูกโหลดพร้อมกับพร็อพเพอร์ตี้ดีฟอลต์ในการกำหนดค่าพื้นฐานของวินโดวส์ที่โปรแกรมใช้งาน โปรแกรมทั่วไปของเดลไฟล์จะแสดงวินโดวส์หลักเมื่อเริ่มใช้งานและมีการใช้ฟอร์มตัวที่สองในการแสดงหรือแลกเปลี่ยนข้อมูลกับยูสเซอร์ การออกแบบโปรแกรมโดยการนำเลเบล และส่วนประกอบอื่นๆมาวางลงบนฟอร์มและส่วนประกอบแต่ละตัว แล้วจึงเพิ่มคำสั่งในการตอบสนองต่ออีเวนต์หรือเหตุการณ์ต่างๆที่ยูสเซอร์กระทำต่อโปรแกรม จุดต่างๆถูกใช้อ้างอิงตำแหน่งเพื่อช่วยให้ส่วนประกอบต่างๆบนฟอร์มได้อย่างมีระเบียบเรียบร้อย

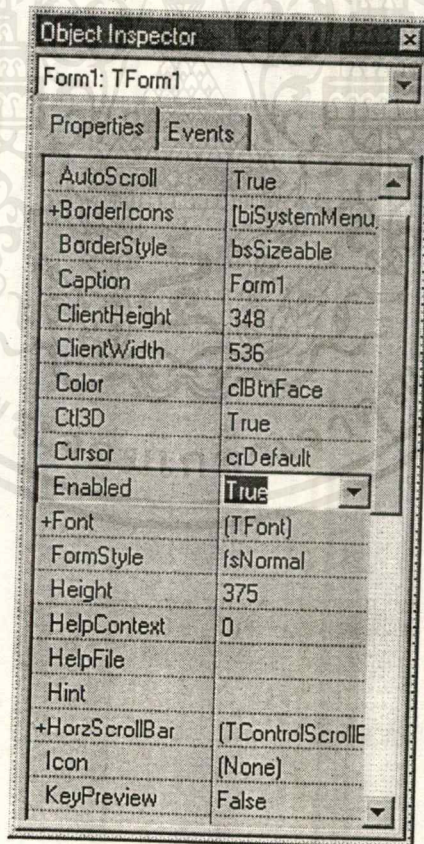


รูปที่ 5.1 วินโดวส์ Form

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์อื่นใด
038900
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การดูรายละเอียดต่างๆด้วยวินโดวส์ Object Inspector

หลังจากกำหนดรูปแบบโปรแกรมอย่างคร่าวๆในวินโดวส์ Form แล้วสามารถปรับแต่งการออกแบบเพจ Properties ของ Object Inspector จะแสดงรายละเอียดโปรแกรมหลักและฟอร์ม โดยมีชื่อวัตถุเป็น Form1 หรือพเพอร์ติ์ทุกตัวจะมีค่าดีฟอลต์เริ่มต้น สามารถปรับแต่งโปรแกรมโดยการแก้ไขหรือพเพอร์ติ์เหล่านี้ Object Inspector จะมีเพจที่สองเรียกว่า Events อีเวนต์ซึ่งมักจะซ้ำกัน (Event : เหตุการณ์ต่างๆ เช่นการคลิกเมาส์ การกดปุ่ม) เช่น OnActive , OnClick , OnCreate ฯลฯ ฟอร์มของเดสไพล์สามารถตอบสนองต่อเหตุการณ์ต่างๆที่เกิดขึ้นจากเมาส์ คีย์บอร์ด หรือจากวินโดวส์เอง สามารถควบคุมการตอบสนองของฟอร์มโดยการกำหนดส่วจัดการอีเวนต์ (Event handler)

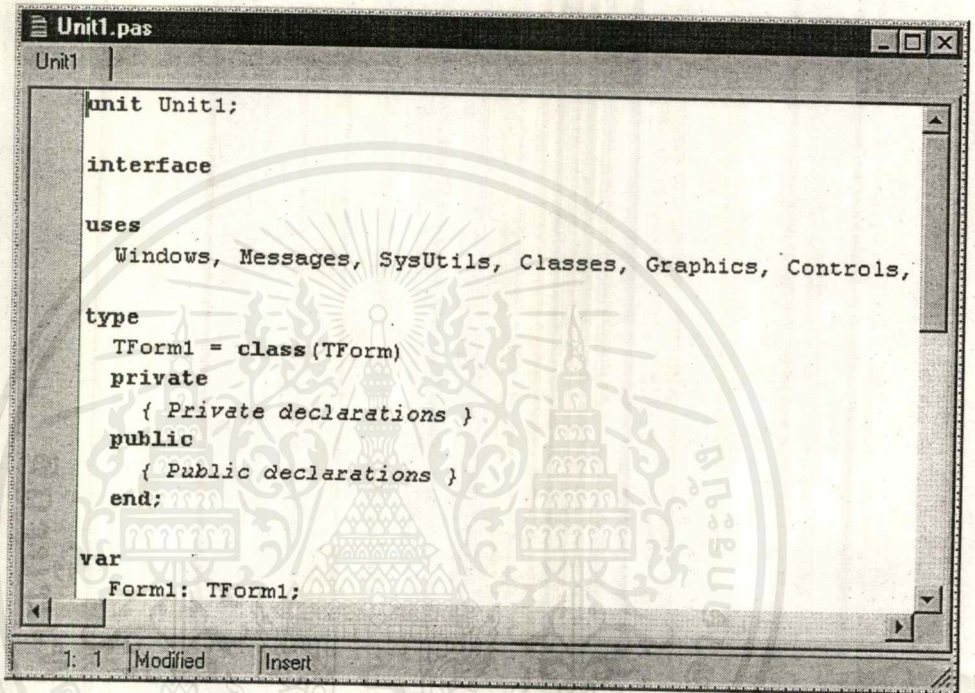


รูปที่ 5.2 Object Inspector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 Code Editor

Code Editor จะประกอบไปด้วยเพชเชอร์สโค้ดของโปรแกรมที่ถูกสร้างขึ้นโดยเดลไฟล์ ซึ่งมีชื่อเรียกว่า Unit1



```

Unit1.pas
Unit1
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,

type
  TForm1 = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  
```

รูปที่ 5.3 Code Editor

5.4 วินโดวส์ Main

ด้านบนสุดของจอภาพคือวินโดวส์หลักที่ชื่อวินโดวส์ Main สามารถเรียกว่าเป็นวินโดวส์หลัก เพราะถ้าปิดมันนั่นคือปิดโปรแกรมเดลไฟล์ไปด้วยวินโดวส์ตัวนี้คือวินโดวส์ที่ควบคุมวินโดวส์ตัวอื่นๆโดยจะแบ่งออกเป็นสามส่วนดังนี้

- วินโดวส์หลัก
- ปุ่มสปิดบาร์
- พาเลตต์คอมโพเนนต์

1. การเลือกเมนู

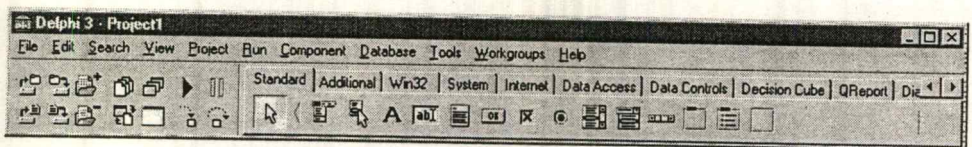
เอกสารนี้เป็นเอกสารเมนูคำสั่งของเดลไฟล์จะเป็นที่คุ้นเคยของยูสเซอร์ที่เคยใช้วินโดวส์และบางตัวก็สามารถอธิบายไม่ว่ากรณีความหมายของตัวมันเองได้แปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ปุ่มสปีดบาร์

ออกแบบมาสำหรับแก้ปัญหาในการจดจำและพิมพ์คำสั่งที่ซับซ้อนและปุ่มสปีดบาร์จะช่วยลดปัญหาในการอ่านและคลิกเมนู ปุ่มสปีดบาร์ของเดลไฟล์ได้จัดเตรียมวิธีที่รวดเร็วในการใช้งานเมนูที่มีประโยชน์มากมายหลายตัว แอปพลิเคชันใหม่ๆและขนาดใหญ่ทั้งหลายบนวินโดวส์จะมีปุ่มSpeedBar ซึ่งอาจใช้ชื่อนี้หรือชื่ออื่นๆ (Smatlcon, ปุ่ม ToolBar, HotCrossButton ฯลฯ) แต่ทั้งหมดก็มีไว้สำหรับแก้ปัญหาเดียวกัน ปุ่มทั้งหมดที่อยู่ที่กรอบทางซ้ายมือของสปีดบาร์ทำให้การทำงานของเดลไฟล์รวดเร็วขึ้น สัญลักษณ์นี้เปิดและเซฟโปรเจกต์ทั้งหมด สุดท้ายปุ่มที่มีไฟล์เดอร์และเครื่องหมายบวกและลบ ใช้สำหรับเพิ่มและลบไฟล์ออกจากโปรเจกต์ ชุดสี่เหลี่ยมตัวถัดไปช่วยนำทางในฟอร์มและไฟล์ยูนิทให้เลือกใช้ปุ่มอื่นๆ ปุ่มล่างซ้ายจะนำจากฟอร์มที่แอกทีฟไปยังยูนิทที่เกี่ยวข้องทั้งไปและกลับ สุดท้ายปุ่มทางขวาจะสร้างและเพิ่มฟอร์มว่างๆในโปรเจกต์ เมื่อมียูนิทและฟอร์มเพียงตัวเดียวในโปรเจกต์ ปุ่มเหล่านี้จะไม่สำคัญเท่าใด แต่เมื่อฟอร์มและยูนิทจำนวนมากในโปรเจกต์ ปุ่มสปีดบาร์มาตรฐานสี่ตัวสุดท้ายจะใช้ในการรันและดีบั๊กโปรแกรมปุ่มที่ดูเหมือนปุ่มเล่นและหยุดชั่วคราว คือรันโปรแกรมและหยุดการทำงานชั่วคราว ได้ปุ่มนี้คือปุ่มที่จะใช้ในการดีบั๊ก

3. พาเลตต์คอมโพเนนต์

พาเลตต์คอมโพเนนต์ของเดลไฟล์จะคล้ายๆ กับกล่องเครื่องมือใช้ในการสร้างโปรแกรมทุกๆ ไอเท็มในทุกๆ เพจของพาเลตต์คอมโพเนนต์คือการทำงานที่สามารถนำไปไว้บนฟอร์มด้วยการคลิกเมาส์และสกรอลล์บาร์ กริดของข้อมูลและเท็กซ์เอดิเตอร์ทั้งหมดอยู่ที่นี้



รูปที่ 5.4 วินโดวส์ Main

5.5 กำหนดตำแหน่งของส่วนประกอบ

โปรแกรมฐานข้อมูลต่างๆเก็บข้อมูลด้วยวิธีการที่แตกต่างกันแต่ก็ลงท้ายด้วยแถวและคอลัมน์ แต่ละคอลัมน์แทนฟิลด์ของข้อมูล กลุ่มของแถวและคอลัมน์เรียกว่าตาราง และส่วนประกอบตารางของเดลไฟล์คือกริดเชื่อมโยงของโปรแกรมไปยังตารางข้อมูลแต่ละตัว ในส่วนประกอบตารางหรือฟอรัม DatabaseName จะเป็นตัวกำหนดฐานข้อมูลที่เก็บตารางนี้ และ ทั้งเอเล็ยสหรือชื่อของไคเรกทอรีที่เก็บ

ไฟล์ตาราง TableName ก็คือชื่อของตาราง การกำหนดพรีอเพอร์ตี Active ให้เป็น True จะเป็นการเปิดตาราง กำหนดพรีอเพอร์ตี DatabaseName เป็น Demos หรือดาต้าเบสใดๆในระบบ (เมื่อคลิกปุ่มลูกศรลงที่ถัดจาก DatabaseName ใน Object Inspector จะเห็นเอเลียสของรายการเลือก) จากนั้นคลิกพรีอเพอร์ตี TableName คลิกปุ่มลูกศรที่ถัดไป แล้วทำการเลือกตารางในฐานข้อมูลที่เลือกไว้

5.6 การดึงข้อมูลจากแหล่งข้อมูล

ตารางเป็นส่วนประกอบที่เรียบร้อยสำหรับส่วนประกอบแหล่งข้อมูลจะทำหน้าที่เป็นตัวกลางในการแลกเปลี่ยนข้อมูลระหว่างตารางและส่วนประกอบข้อมูล พรีอเพอร์ตี Dataset จะชี้ไปยังตารางซึ่งเป็นที่มีการไหลของข้อมูล ถ้า AutoEdit มีค่าเป็น True การไหลของข้อมูลจะขึ้นทั้งสองด้าน

5.7 คิวรี เพื่อหาข้อมูล

ส่วนประกอบคิวรี คืออีกทางหนึ่งในการเชื่อมโยงโปรแกรมเดลไฟล์เข้ากับดาต้าเบสเช่นเดียวกับส่วนประกอบของตารางประกอบไปด้วยพรีอเพอร์ตี DatabaseName และ Active Datasource เป็นตัวการเชื่อมโยงระหว่างส่วนประกอบดาต้าอแวร เมื่อโปรแกรมเปิดส่วนประกอบคิวรี จะดึงข้อมูลตามที่กำหนดไว้ในคำสั่ง SQL (Structured Query Language) ในพรีอเพอร์ตี SQL กำหนด RequestLive ให้เป็น True คิวรีจะเชื่อมต่อเข้ากับดาต้าเบส ดังนั้นการเอ็ดิตด้วยคิวรีจะส่งคืนค่ากลับไปยังดาต้าเบส

บทที่ 6

การสร้างแอปพลิเคชันของงานบริการห้องสมุด

6.1 กฎในการใช้บริการห้องสมุดและข้อมูลที่ต้องทราบ

กฎในการใช้บริการห้องสมุด (เฉพาะกฎที่จำเป็นต้องใช้ในการเขียนโปรแกรม)

- ยืมได้ไม่เกิน 3 เล่มต่อคน
- ยืมหนังสือได้เป็นระยะเวลา 7 วัน
- ยืมหนังสือเกินระยะเวลาที่กำหนดจะต้องเสียค่าปรับเป็นจำนวน 1 บาทต่อเล่มต่อวัน
- ผู้ยืมทำหนังสือชำรุดหรือสูญหายจะต้องชดใช้เป็นจำนวนเงินตามราคาหนังสือ
- ผู้ยืมไม่ส่งหนังสือภายในระยะเวลาที่กำหนด จะไม่สามารถยืมหนังสือในครั้งต่อไปได้

ข้อมูลของหนังสือ

- เลขทะเบียนหนังสือ
- เลขหมู่หนังสือ
- ชื่อหนังสือภาษาไทย
- ชื่อหนังสือภาษาอังกฤษ
- ชื่อผู้แต่ง
- ชื่อสำนักพิมพ์
- พิมพ์ครั้งที่
- จำนวนหน้า
- ราคาหนังสือ
- สถานะ (ถูกยืมไปแล้วหรือไม่)

ข้อมูลนักศึกษา

- รหัสนักศึกษา
- ชื่อนักศึกษา
- นามสกุลนักศึกษา
- ห้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 Code editor ที่ใช้ในโปรแกรม

```
unit main;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm1 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
BitBtn3: TBitBtn;
```

```
BitBtn4: TBitBtn;
```

```
Panel2: TPanel;
```

```
procedure BitBtn2Click(Sender: TObject);
```

```
procedure BitBtn1Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form1: TForm1;
```

```
implementation
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
uses offic, find, pass;
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{$R *.DFM}
```

```
procedure TForm1.BitBtn2Click(Sender: TObject);
```

```
begin
```

```
form27.showmodal;
```

```
end;
```

```
procedure TForm1.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
form22.showmodal;
```

```
end;
```

```
end.
```

```
unit editst;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
```

```
DBTables, Db, Buttons, StdCtrls, Mask, DBCtrls, ExtCtrls;
```

```
type
```

```
TForm10 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

เอกสารนี้เป็น **Label1: TLabel**; สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด **Label2: TLabel**; มมให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
DBEdit3: TDBEdit;
Button1: TButton;
Button2: TButton;
BitBtn1: TBitBtn;
Table1: TTable;
DataSource1: TDataSource;
Query1: TQuery;
DataSource2: TDataSource;
Label8: TLabel;
DBEdit4: TDBEdit;
Label9: TLabel;
Label10: TLabel;
Edit5: TEdit;
Label11: TLabel;
Query2: TQuery;
Query3: TQuery;
Query4: TQuery;
Query5: TQuery;
Button3: TButton;

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่สามารถเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 หรือการอื่นใดได้

```

procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form10: TForm10;

implementation
  {$R *.DFM}

  procedure TForm10.Edit1Change(Sender: TObject);
  begin
    dbedit1.Clear;
    dbedit2.Clear;
    dbedit3.Clear;
    dbedit4.Clear;

  end;

  procedure TForm10.Button1Click(Sender: TObject);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ query2.active := false; ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (edit2.Text <> "")
  then
    begin
      query2.ParamByName('ids').asString := edit1.text;
      query2.ParamByName('names').asString := edit2.text;
      query2.ExecSQL;
    end
  else
    begin
      query2.ParamByName('ids').asString := edit1.text;
      query2.ParamByName('names').asString := dbedit1.Field.Text;
      query2.ExecSQL;
    end;

query3.active := false;
if (edit3.Text <> "")
  then
    begin
      query3.ParamByName('ids').asString := edit1.text;
      query3.ParamByName('layers').asString := edit3.text;
      query3.ExecSQL;
    end
  else
    begin
      query3.ParamByName('ids').asString := edit1.text;
      query3.ParamByName('layers').asString := dbedit2.Field.Text;
      query3.ExecSQL;
    end;

```

```

then
  begin
    query4.ParamByName('ids').asString := edit1.text;
    query4.ParamByName('rooms').asString := edit4.text;
    query4.ExecSQL;
  end
else
  begin
    query4.ParamByName('ids').asString := edit1.text;
    query4.ParamByName('rooms').asString := dbedit3.Field.Text;
    query4.ExecSQL;
  end;

  query5.active := false;
  if (edit5.Text <> "")
  then
    begin
      query5.ParamByName('ids').asString := edit1.text;
      query5.ParamByName('addressss').asString := edit5.text;
      query5.ExecSQL;
    end
  else
    begin
      query5.ParamByName('ids').asString := edit1.text;
      query5.ParamByName('addressss').asString := dbedit4.Field.Text;
      query5.ExecSQL;
    end;

  query1.active := false;
  query1.ParamByName('ids').asString := edit1.text;
  query1.active := true;
end;

```

```

procedure TForm10.Button2Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
edit5.Clear;
dedit1.Clear;
dedit2.Clear;
dedit3.Clear;
dedit4.Clear;
end;

procedure TForm10.Button3Click(Sender: TObject);
begin
    query1.active := false;
    query1.ParamByName('ids').asString := edit1.text;
    query1.active := true ;
end;

procedure TForm10.FormActivate(Sender: TObject);
begin
edit1.SetFocus;
end;
end.
unit mbk;

interface

```

StdCtrls, Buttons, ExtCtrls;

type

TForm11 = class(TForm)

Panel1: TPanel;

GroupBox1: TGroupBox;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

BitBtn4: TBitBtn;

Panel2: TPanel;

procedure BitBtn1Click(Sender: TObject);

procedure BitBtn2Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form11: TForm11;

implementation

uses addbk, rembk, editbk, medit;

{ \$R *.DFM }

เอกสารนี้ `procedure TForm11.BitBtn1Click(Sender: TObject);` นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณี `begin` ใด ๆ ทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

form12.showmodal;
form12.DBEdit1.Clear;
form12.Edit1.Clear;
form12.Edit2.Clear;
form12.Edit3.Clear;
form12.Edit4.Clear;
form12.Edit5.Clear;
form12.Edit6.Clear;
form12.Edit7.Clear;
form12.Edit8.Clear;
form12.Edit9.Clear;
form12.Edit10.Clear;
form12.Label8.Enabled := false;
form12.Label9.Enabled := false;
end;

procedure TForm11.BitBtn2Click(Sender: TObject);
begin
form13.showmodal;
form13.Edit1.Clear;
form13.Query1.Active :=false;
form13.BitBtn1.Enabled :=false;
end;

procedure TForm11.BitBtn3Click(Sender: TObject);
begin
form29.showmodal;

end;

```

เอกสารนี้ใช้เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit addbk;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, Mask, DBCtrls, ExtCtrls, DBTables, Db;
```

```
type
```

```
TForm12 = class(TForm)
```

```
Panel1: TPanel;
```

```
Label3: TLabel;
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label4: TLabel;
```

```
Label5: TLabel;
```

```
Label6: TLabel;
```

```
Label7: TLabel;
```

```
Label10: TLabel;
```

```
Edit1: TEdit;
```

```
DBEdit1: TDBEdit;
```

```
Edit2: TEdit;
```

```
Edit3: TEdit;
```

```
Edit4: TEdit;
```

```
BitBtn1: TBitBtn;
```

```
Edit5: TEdit;
```

```
Edit6: TEdit;
```

```
Panel3: TPanel;
```

```
Label8: TLabel;
```

```
Label9: TLabel;
```

```

BitBtn2: TBitBtn;
Table1: TTable;
DataSource1: TDataSource;
Query1: TQuery;
DataSource2: TDataSource;
Query2: TQuery;
Table2: TTable;
Label11: TLabel;
Edit7: TEdit;
Label12: TLabel;
Edit8: TEdit;
Label13: TLabel;
Edit9: TEdit;
Label14: TLabel;
Table3: TTable;
Edit10: TEdit;
Table4: TTable;
Query3: TQuery;
DataSource3: TDataSource;
DBText1: TDBText;
DBText2: TDBText;
DataSource4: TDataSource;
DBText3: TDBText;
DBText4: TDBText;
DBText5: TDBText;
DBText6: TDBText;
DBText7: TDBText;
DBText8: TDBText;
DBText9: TDBText;
DataSource5: TDataSource;
DataSource6: TDataSource;
Edit11: TEdit;

```

```

Label15: TLabel;
DBText10: TDBText;
DBText11: TDBText;
DataSource7: TDataSource;
Query5: TQuery;
DataSource8: TDataSource;
DataSource9: TDataSource;
Query6: TQuery;
DataSource10: TDataSource;
procedure BitBtn1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure Edit1Change(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form12: TForm12;

implementation

{$R *.DFM}

procedure TForm12.BitBtn1Click(Sender: TObject);
var c : integer;
begin

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ query1.Active := true; แปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
c := strtoint(dbedit1.Field.Text);
```

```
c := c+1;
```

```
query2.ParamByName('co').asString := intostr(c);
```

```
query2.ExecSQL;
```

```
if (edit1.text = "")and(dbtext10.Field.text=")and(dbtext11.Field.text=")
```

```
and(edit2.text = "") and(edit11.text = ") or (edit3.text = ")
```

```
and (dbtext1.Field.Text =")
```

```
then
```

```
begin
```

```
label8.Enabled := true;
```

```
label9.Enabled := false;
```

```
query1.Active := false;
```

```
query1.Active := true;
```

```
c := strtoint(dbedit1.Field.Text);
```

```
c := c-1;
```

```
query2.ParamByName('co').asString := intostr(c);
```

```
query2.ExecSQL;
```

```
query1.Active := false;
```

```
query1.Active := true;
```

```
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
```

```
if (dbtext1.Field.Text = "") then
```

```
begin
```

```
table2.insertrecord([dbedit1.Field.Text ,edit1.text,  
edit7.text,'Y',edit8.text,edit9.text,edit4.text]);
```

```
table3.insertrecord([edit1.Text,edit3.text,  
edit5.text,edit6.text,edit10.text]);
```

```
table4.insertrecord([dbedit1.Field.Text ,edit2.text]);
```

```
table4.insertrecord([dbedit1.Field.Text ,edit11.text]);
```

```
label8.Enabled := false;
```

```
label9.Enabled := true;
```

```
Edit1.clear;
```

```
Edit2.clear;
```

```
Edit3.clear;
```

```
Edit4.clear;
```

```
Edit5.clear;
```

```
Edit6.clear;
```

```
edit7.Clear;
```

```
Edit8.clear;
```

```
Edit9.clear;
```

```
Edit10.clear;
```

```
Edit11.clear;
```

```
end
```

```
else
```

```
begin
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

table2.insertrecord([dbedit1.Field.Text ,edit1.text,
    dbtext5.field.text,'Y',dbtext6.field.text,
    dbtext7.field.text,dbtext2.field.text]);

// table3.insertrecord([edit1.Text,dbtext1.field.text,
// dbtext3.field.text,dbtext4.field.text,dbtext8.field.text]);

table4.insertrecord([dbedit1.Field.Text ,dbtext10.field.text]);
table4.insertrecord([dbedit1.Field.Text ,dbtext11.field.text]);

label8.Enabled := false;
label9.Enabled := true;
Edit1.clear;
Edit2.clear;
Edit3.clear;
Edit4.clear;
Edit5.clear;
Edit6.clear;
edit7.Clear;
Edit8.clear;
Edit9.clear;
Edit10.clear;
Edit11.clear;

end;

```

```
query1.Active := false;
```

```
query1.Active := true;
```

```
Edit1.clear;
```

```
Edit2.clear;
```

```

Edit3.clear;
Edit4.clear;
Edit5.clear;
Edit6.clear;
edit7.Clear;
Edit8.clear;
Edit9.clear;
Edit10.clear;
edit11.Clear;

edit1.SetFocus;

end;

procedure TForm12.FormActivate(Sender: TObject);
begin
    query1.Active := false;
    query1.Active := true;
end;

procedure TForm12.Edit1Change(Sender: TObject);
begin

query3.Active := false;
query3.ParamByName('calls').asstring := edit1.Text;
query3.Active := true;

query4.Active := false;
query4.ParamByName('calls').asstring := edit1.Text;

```

```
query4.Active := true;
```

```
query5.Active := false;
```

```
query5.ParamByName('idbooks').asstring := dbtext9.Field.Text;
```

```
query5.Active := true;
```

```
query6.Active := false;
```

```
query6.ParamByName('idbooks').asstring := dbtext9.Field.Text;
```

```
query6.ParamByName('nots').asstring := dbtext10.Field.Text;
```

```
query6.Active := true;
```

```
end;
```

```
end.
```

```
unit rembk;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, Mask, DBCtrls, ExtCtrls, DBTables, Db;
```

```
type
```

```
TForm13 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label4: TLabel;
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ห้ามมิให้คัดลอกหรือเผยแพร่โดยไม่ได้รับอนุญาต
 ไม่ว่ากรณีใดๆ หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงและขอให้อ่านเงื่อนไขการใช้งานฉบับล่าสุดที่แนบมาไว้ที่ท้ายเอกสารทุกครั้งที่มีการนำไปใช้

```

Label7: TLabel;
Edit1: TEdit;
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
DBEdit3: TDBEdit;
BitBtn1: TBitBtn;
BitBtn2: TBitBtn;
DBEdit5: TDBEdit;
DBEdit6: TDBEdit;
Table1: TTable;
DataSource1: TDataSource;
Query1: TQuery;
DataSource2: TDataSource;
Button1: TButton;
DBText1: TDBText;
procedure BitBtn1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Edit1Change(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form13: TForm13;

implementation

```

```
uses wrembk;
```

```
{SR *.DFM}
```

```
procedure TForm13.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
if (edit1.Text <> '') and (dbedit1.Field.Text <> '')
```

```
then
```

```
begin
```

```
form14.showmodal;
```

```
end;
```

```
end;
```

```
procedure TForm13.Button1Click(Sender: TObject);
```

```
begin
```

```
query1.active := false ;
```

```
query1.ParamByName('idbooks').asString := edit1.text;
```

```
query1.active := true ;
```

```
if (dbtext1.Field.Text <> 'N')
```

```
then
```

```
begin
```

```
BitBtn1.Enabled := true;
```

```
BitBtn1.SetFocus;
```

```
end
```

```
else
```

```
begin
```

```
BitBtn1.Enabled := false;
```

```
showmessage('หนังสือถูกยืม');
```

```

end;
end;

```

```

procedure TForm13.Edit1Change(Sender: TObject);

```

```

begin

```

```

  dbedit1.Clear;

```

```

  dbedit2.Clear;

```

```

  dbedit3.Clear;

```

```

  dbedit5.Clear;

```

```

  dbedit6.Clear;

```

```

  BitBtn1.Enabled :=false;

```

```

end;

```

```

procedure TForm13.BitBtn2Click(Sender: TObject);

```

```

begin

```

```

  query1.Active := false;

```

```

end;

```

```

end.

```

```

unit wrembk;

```

```

interface

```

```

uses

```

```

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

```

```

  Buttons, StdCtrls, ExtCtrls, DBTables, Db, DBCtrls;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TForm14 = class(TForm)
  Panel1: TPanel;
  Button1: TButton;
  GroupBox1: TGroupBox;
  Label2: TLabel;
  BitBtn1: TBitBtn;
  Table1: TTable;
  DataSource1: TDataSource;
  Query1: TQuery;
  Query2: TQuery;
  Table2: TTable;
  DataSource2: TDataSource;
  DBText1: TDBText;
  Query3: TQuery;
  DataSource3: TDataSource;
  DBText2: TDBText;
  Label1: TLabel;
  Label3: TLabel;
  Table3: TTable;
  DataSource4: TDataSource;
  Query4: TQuery;
  DataSource5: TDataSource;
  DBText3: TDBText;
  Table4: TTable;
  DataSource6: TDataSource;
  Query5: TQuery;
  DataSource7: TDataSource;
  Query6: TQuery;
  DataSource8: TDataSource;
  DBText4: TDBText;

```

```

procedure Button1Click(Sender: TObject);

```

```

procedure FormActivate(Sender: TObject);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อผู้อื่นโดยเด็ดขาด

แจ้งให้ทราบถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    procedure BitBtn1Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form14: TForm14;

implementation

uses rembk;

{$R *.DFM}

procedure TForm14.Button1Click(Sender: TObject);
begin
    query1.ParamByName('idbooks').asString := form13.edit1.text;
    query1.ExecSQL;

    query5.ParamByName('idbooks').asString := form13.edit1.text;
    query5.ExecSQL;

    query6.Active := false;
    query6.ParamByName('calls').asString := dbtext2.Field.text;
    query6.Active := true;

    if (dbtext4.Field.Text = "")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

query2.ParamByName('calls').asString := dbtext2.Field.Text;
query2.ExecSQL;
end;

```

```

form13.Edit1.Clear;
form13.Query1.Active :=false;
close;
end;

```

```

procedure TForm14.FormActivate(Sender: TObject);
begin
query3.Active := false;
query3.ParamByName('idbooks').asString := form13.edit1.text;
query3.Active := true;

query4.Active := false;
query4.ParamByName('idbooks').asstring := dbtext1.Field.Text;
query4.Active := true;

```

```

query6.Active := false;
query6.ParamByName('calls').asstring := dbtext2.Field.text;
query6.Active := true;

```

```

if (dbtext3.Field.Text = "")
then button1.Enabled := true
else
begin
button1.Enabled := false;

```

```

showmessage('หนังสือถูกยืม');

```

```

end;

```

```

end;
procedure TForm14.BitBtn1Click(Sender: TObject);
begin
form13.Edit1.Clear;
form13.Query1.Active :=false;
end;
end.
unit editbk;

interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, Mask, DBCtrls, ExtCtrls, DBTables, Db;

type
TForm15 = class(TForm)
Panel1: TPanel;
Label10: TLabel;
GroupBox1: TGroupBox;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label11: TLabel;
Label5: TLabel;
Label6: TLabel;
Label14: TLabel;
Label15: TLabel;
Label16: TLabel;
Label17: TLabel;
Label18: TLabel;
Edit1: TEdit;

```

Edit4: TEdit;
 Edit5: TEdit;
 Edit6: TEdit;
 Edit7: TEdit;
 Edit8: TEdit;
 DBEdit3: TDBEdit;
 DBEdit4: TDBEdit;
 DBEdit5: TDBEdit;
 DBEdit6: TDBEdit;
 DBEdit7: TDBEdit;
 Button1: TButton;
 Button2: TButton;
 Table1: TTable;
 DataSource1: TDataSource;
 Query1: TQuery;
 DataSource2: TDataSource;
 Label7: TLabel;
 Label8: TLabel;
 DBEdit8: TDBEdit;
 Label19: TLabel;
 DBEdit9: TDBEdit;
 DBEdit10: TDBEdit;
 Label20: TLabel;
 Query2: TQuery;
 Table2: TTable;
 DataSource3: TDataSource;
 Query3: TQuery;
 Query4: TQuery;
 Query5: TQuery;
 Query6: TQuery;

```

Label21: TLabel;
Query9: TQuery;
Edit9: TEdit;
Label22: TLabel;
Query10: TQuery;
Edit10: TEdit;
Label23: TLabel;
Query11: TQuery;
Edit11: TEdit;
Query12: TQuery;
Button3: TButton;
Table3: TTable;
DataSource4: TDataSource;
procedure Button2Click(Sender: TObject);
procedure Edit1Change(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form15: TForm15;

implementation

{$R *.DFM}

```

```
begin
```

```
close;
```

```
end;
```

```
procedure TForm15.Edit1Change(Sender: TObject);
```

```
begin
```

```
    query1.Active := false;
```

```
if (edit1.Text = "") then button1.Enabled := false
```

```
    else button1.Enabled := true;
```

```
DBEdit3.Clear;
```

```
DBEdit4.Clear;
```

```
DBEdit5.Clear;
```

```
DBEdit6.Clear;
```

```
DBEdit7.Clear;
```

```
DBEdit8.Clear;
```

```
DBEdit9.Clear;
```

```
DBEdit10.Clear;
```

```
end;
```

```
procedure TForm15.Button1Click(Sender: TObject);
```

```
begin
```

```
(* query2.active := false;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

then
  begin
    query2.ParamByName('idbooks').asString := edit1.text;
    query2.ParamByName('calls').asString := edit2.text;

    query3.ParamByName('s').asString := dbedit1.Field.Text;
    query3.ParamByName('calls').asString := edit2.text;

    query2.ExecSQL;
    query3.ExecSQL;
  end
else
  begin
    query2.ParamByName('idbooks').asString := edit1.text;
    query2.ParamByName('calls').asString := dbedit1.Field.Text;

    query3.ParamByName('s').asString := dbedit1.Field.Text;
    query3.ParamByName('calls').asString := dbedit1.Field.Text;

    query2.ExecSQL;
    query3.ExecSQL;
  end;

query4.active := false;
if (edit3.Text <> "")
  then
    begin
      query4.ParamByName('s').asString := edit1.Text;
      query4.ParamByName('titels').asString := edit3.text;
      query4.ExecSQL;
    end

```

```

else
  begin
query4.ParamByName('s').asString := edit1.Text;
query4.ParamByName('titels').asString := dbedit2.Field.Text;
query4.ExecSQL;
  end; *)

query5.active := false;
if (edit4.Text <> "")
  then
  begin
query5.ParamByName('s').asString := edit1.Text;
query5.ParamByName('booknames').asString := edit4.text;
query5.ExecSQL;
  end
else
  begin
query5.ParamByName('s').asString := edit1.Text;
query5.ParamByName('booknames').asString := dbedit3.Field.Text;
query5.ExecSQL;
  end;

query6.active := false;
if (edit5.Text <> "")
  then
  begin
query6.ParamByName('calls').asString := edit1.text;
query6.ParamByName('writers').asString := edit5.text;
query6.ExecSQL;

```

```

begin
query6.ParamByName('calls').asString := edit1.text;
query6.ParamByName('writers').asString := dbedit4.Field.Text;
query6.ExecSQL;
end;

```

```

query7.active := false;
if (edit6.Text <> "")
then
begin
query7.ParamByName('s').asString := edit1.Text;
query7.ParamByName('publishs').asString := edit6.text;
query7.ExecSQL;
end
else
begin
query7.ParamByName('s').asString := edit1.Text;
query7.ParamByName('publishs').asString := dbedit5.Field.Text;
query7.ExecSQL;
end;

```

```

query8.active := false;
if (edit7.Text <> "")
then
begin
query8.ParamByName('s').asString := edit1.Text;
query8.ParamByName('editions').asString := edit7.text;
query8.ExecSQL;
end
else

```

```
begin
```

```
query8.ParamByName('s').asString := edit1.Text;
```

```

query8.ParamByName('editions').asString := dbedit6.Field.Text;
query8.ExecSQL;

end;

query9.active := false;
if (edit8.Text <> "")
then
begin
query9.ParamByName('calls').asString := edit1.Text;
query9.ParamByName('p_years').asString := edit8.text;
query9.ExecSQL;
end
else
begin
query9.ParamByName('calls').asString := edit1.Text;
query9.ParamByName('p_years').asString := dbedit7.Field.Text;
query9.ExecSQL;
end;

query10.active := false;
if (edit9.Text <> "")
then
begin
query10.ParamByName('s').asString := edit1.Text;
query10.ParamByName('isbns').asString := edit9.text;
query10.ExecSQL;
end
else
begin
query10.ParamByName('s').asString := edit1.Text;

```

```

end;

query11.active := false;
if (edit10.Text <> "")
then
begin
query11.ParamByName('calls').asString := edit1.Text;
query11.ParamByName('get_types').asString := edit10.text;
query11.ExecSQL;
end
else
begin
query11.ParamByName('calls').asString := edit1.Text;
query11.ParamByName('get_types').asString := dbedit9.Field.Text;
query11.ExecSQL;

end;

query12.active := false;
if (edit11.Text <> "")
then
begin
query12.ParamByName('calls').asString := edit1.Text;
query12.ParamByName('prices').asString := edit11.text;
query12.ExecSQL;
end
else
begin
query12.ParamByName('calls').asString := edit1.Text;
query12.ParamByName('prices').asString := dbedit10.Field.Text;
query12.ExecSQL;
end;

```

```

query1.active := false;
query1.ParamByName('calls').asString := edit1.text;
query1.active := true ;

```

```

edit4.Clear;
edit5.Clear;
edit6.Clear;
edit7.Clear;
edit8.Clear;
edit9.Clear;
edit10.Clear;
edit11.Clear;

```

```
end;
```

```

procedure TForm15.Button3Click(Sender: TObject);
begin
query1.Active := false;
query1.ParamByName('calls').asString := edit1.Text;
query1.Active := true;

```

```
end;
```

```
end.
```

```
unit chkbk;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBTables, Db, Buttons, StdCtrls, Grids, DBGrids, ExtCtrls;
```

```
type
```

```
TForm16 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
DBGrid1: TDBGrid;
```

```
Button1: TButton;
```

```
Button2: TButton;
```

```
BitBtn1: TBitBtn;
```

```
Table1: TTable;
```

```
Table2: TTable;
```

```
DataSource1: TDataSource;
```

```
DataSource2: TDataSource;
```

```
Query1: TQuery;
```

```
Query2: TQuery;
```

```
DataSource3: TDataSource;
```

```
DataSource4: TDataSource;
```

```
procedure Button1Click(Sender: TObject);
```

```
procedure Button2Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form16: TForm16;
```

```
implementation
```

```
{$R *.DFM}
```

```
procedure TForm16.Button1Click(Sender: TObject);
```

```
begin
```

```
query1.Active := false;
```

```
DBGrid1.DataSource := DataSource3;
```

```
query1.Active := true;
```

```
end;
```

```
procedure TForm16.Button2Click(Sender: TObject);
```

```
begin
```

```
query2.Active := false;
```

```
DBGrid1.DataSource := DataSource4;
```

```
query2.Active := true;
```

```
end;
```

```
end.
```

```
unit presen;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm17 = class(TForm)  
    Panel1: TPanel;  
    GroupBox1: TGroupBox;  
    BitBtn1: TBitBtn;  
    BitBtn2: TBitBtn;  
    BitBtn3: TBitBtn;  
    procedure BitBtn1Click(Sender: TObject);  
    procedure BitBtn2Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form17: TForm17;
```

```
implementation
```

```
uses today, p;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
{ \$R *.DFM }
ไม่รวมกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm17.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
form18.showmodal;
```

```
end;
```

```
procedure TForm17.BitBtn2Click(Sender: TObject);
```

```
begin
```

```
form20.showmodal;
```

```
end;
```

```
end.
```

```
unit today;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
```

```
StdCtrls, Buttons, DBTables, Db, Grids, DBGrids, ExtCtrls;
```

```
type
```

```
TForm18 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
DBGrid1: TDBGrid;
```

```
Table1: TTable;
```

```
DataSource1: TDataSource;
```

```
Query1: TQuery;
```

```
DataSource2: TDataSource;
```

```

BitBtn1: TBitBtn;
BitBtn2: TBitBtn;
procedure FormActivate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form18: TForm18;

implementation

uses pritoday;

{$R *.DFM}

procedure TForm18.FormActivate(Sender: TObject);
begin
  query1.Active := false;
  query1.ParamByName('dat').asdate := date;
  query1.Active := true;
end;

procedure TForm18.BitBtn1Click(Sender: TObject);
begin
  form19.QuickRep1.print;
end;

```

```
unit pritoday;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
QrCtrls, quickrpt, ExtCtrls;
```

```
type
```

```
TForm19 = class(TForm)
```

```
QuickRep1: TQuickRep;
```

```
QRBand1: TQRBand;
```

```
QRDBText1: TQRDBText;
```

```
QRDBText2: TQRDBText;
```

```
QRDBText3: TQRDBText;
```

```
QRDBText4: TQRDBText;
```

```
QRDBText5: TQRDBText;
```

```
QRDBText6: TQRDBText;
```

```
QRBand2: TQRBand;
```

```
QRLabel1: TQRLabel;
```

```
QRLabel2: TQRLabel;
```

```
QRLabel3: TQRLabel;
```

```
QRLabel4: TQRLabel;
```

```
QRLabel5: TQRLabel;
```

```
QRLabel6: TQRLabel;
```

```
QRBand3: TQRBand;
```

```
QRLabel7: TQRLabel;
```

```
private
```

```
{ Private declarations }
```

```
public
```

เอกสาร { Public declarations } นี้ให้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าก่ end; ๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
var
```

```
Form19: TForm19;
```

```
implementation
```

```
{$R *.DFM}
```

```
end.
```

```
unit offic;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm2 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
BitBtn3: TBitBtn;
```

```
BitBtn4: TBitBtn;
```

```
BitBtn5: TBitBtn;
```

```
BitBtn6: TBitBtn;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BitBtn7: TBitBtn;
BitBtn8: TBitBtn;

procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses borrow, return, mbook, mbk, chkbk, presen, money;

{$R *.DFM}

procedure TForm2.BitBtn1Click(Sender: TObject);
begin
  form3.showmodal;
  form3.query1.Active := false;
  form3.query4.Active := false;

```

```
procedure TForm2.BitBtn2Click(Sender: TObject);
```

```
begin
```

```
form4.showmodal;
```

```
form4.edit1.Clear;
```

```
form4.Label13.Caption := '';
```

```
form4.Label15.Caption := '';
```

```
form4.query1.Active :=false;
```

```
end;
```

```
procedure TForm2.BitBtn3Click(Sender: TObject);
```

```
begin
```

```
form5.showmodal;
```

```
end;
```

```
procedure TForm2.BitBtn4Click(Sender: TObject);
```

```
begin
```

```
form11.showmodal;
```

```
end;
```

```
procedure TForm2.BitBtn5Click(Sender: TObject);
```

```
begin
```

```
form16.showmodal;
```

```
form16.Query1.Active :=false;
```

```
form16.Query2.Active :=false;
```

```
end;
```

```
procedure TForm2.BitBtn6Click(Sender: TObject);
```

```
begin
```

```
form17.showmodal;
```

```
end;
```

```
procedure TForm2.BitBtn8Click(Sender: TObject);  
begin  
form28.showmodal;  
end;  
  
end.
```



```
unit p;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBTables, Db, StdCtrls, Buttons, Grids, DBGrids, ExtCtrls;
```

```
type
```

```
TForm20 = class(TForm)
```

```
  Panel1: TPanel;
```

```
  GroupBox1: TGroupBox;
```

```
  DBGrid1: TDBGrid;
```

```
  Panel2: TPanel;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  BitBtn1: TBitBtn;
```

```
  Table1: TTable;
```

```
  DataSource1: TDataSource;
```

```
  Query1: TQuery;
```

```
  DataSource2: TDataSource;
```

```
  BitBtn2: TBitBtn;
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
  procedure FormActivate(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
end;
```

implementation

uses pp;

{\$R *.DFM}

```
procedure TForm20.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
  form21.QuickRep1.print;
```

```
end;
```

```
procedure TForm20.FormActivate(Sender: TObject);
```

```
begin
```

```
  label2.Caption := datetostr(date);
```

```
  form21.QRLabel6.Caption := label2.Caption;
```

```
  query1.Active := false;
```

```
  query1.ParamByName('dat').asdate := date;
```

```
  query1.Active := true;
```

```
end;
```

```
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit pp;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
QrCtrls, quickrpt, ExtCtrls;
```

```
type
```

```
TForm21 = class(TForm)
```

```
QuickRep1: TQuickRep;
```

```
QRBand1: TQRBand;
```

```
QRDBText1: TQRDBText;
```

```
QRDBText2: TQRDBText;
```

```
QRDBText3: TQRDBText;
```

```
QRDBText4: TQRDBText;
```

```
QRDBText5: TQRDBText;
```

```
QRDBText6: TQRDBText;
```

```
QRBand2: TQRBand;
```

```
QRLabel6: TQRLabel;
```

```
QRLabel8: TQRLabel;
```

```
QRBand3: TQRBand;
```

```
QRLabel1: TQRLabel;
```

```
QRLabel2: TQRLabel;
```

```
QRLabel3: TQRLabel;
```

```
QRLabel4: TQRLabel;
```

```
QRLabel5: TQRLabel;
```

```
QRLabel7: TQRLabel;
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form21: TForm21;
```

```
implementation
```

```
{$R *.DEM}
```

```
end.
```

```
unit find;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm22 = class(TForm) -
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
BitBtn3: TBitBtn;
```

```
GroupBox2: TGroupBox;
```

```
Label1: TLabel;
```

```
BitBtn4: TBitBtn;
```

```

procedure BitBtn1Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form22: TForm22;

implementation

uses fbksnam, finwrit, tit;

{$R *.DFM}

procedure TForm22.BitBtn1Click(Sender: TObject);
begin
  form23.showmodal;
  form23.Edit1.Clear;

  form23.Query1.Active := false;
  form23.Query2.Active := false;

  form23.query1.ParamByName('booknames').asstring := '';
  form23.query1.Active := true;
  form23.query2.Active := true;

```

```

form23.label2.Caption := inttostr(form23.query1.RecordCount);

form23.Query1.Active := false;
form23.Query2.Active := false;

form23.Label4.Enabled := false;
form23.Label5.Enabled := false;

end;

procedure TForm22.BitBtn3Click(Sender: TObject);
begin
close;
end;

procedure TForm22.BitBtn2Click(Sender: TObject);
begin
form24.showmodal;
form24.Edit1.Clear;

form24.Query1.Active := false;
form24.Query2.Active := false;

form24.query1.ParamByName('writers').asstring := '';
form24.query1.Active := true;
form24.query2.Active := true;

form24.label2.Caption := inttostr(form24.query1.RecordCount);

form24.Query1.Active := false;
form24.Query2.Active := false;

```

```
form24.Label4.Enabled := false;
```

```
form24.Label5.Enabled := false;
```

```
end;
```

```
procedure TForm22.BitBtn4Click(Sender: TObject);
```

```
begin
```

```
form25.showmodal;
```

```
form25.Edit1.Clear;
```

```
form25.Query1.Active := false;
```

```
form25.Query2.Active := false;
```

```
form25.query1.ParamByName('titels').asstring := '';
```

```
form25.query1.Active := true;
```

```
form25.query2.Active := true;
```

```
form25.label2.Caption := intostr(form25.query1.RecordCount);
```

```
form25.Query1.Active := false;
```

```
form25.Query2.Active := false;
```

```
form25.Label4.Enabled := false;
```

```
form25.Label5.Enabled := false;
```

```
end;
```

```
end.
```

```
unit fbknam;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, Grids, DBGrids, ExtCtrls, DBTables, Db;
```

```
type
```

```
TForm23 = class(TForm)
```

```
Panel1: TPanel;
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label6: TLabel;
```

```
GroupBox1: TGroupBox;
```

```
DBGrid1: TDBGrid;
```

```
DBGrid2: TDBGrid;
```

```
Edit1: TEdit;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
GroupBox2: TGroupBox;
```

```
Label4: TLabel;
```

```
Label5: TLabel;
```

```
Label7: TLabel;
```

```
Table1: TTable;
```

```
DataSource1: TDataSource;
```

```
Query1: TQuery;
```

```
DataSource2: TDataSource;
```

```
Query2: TQuery;
```

```
DataSource3: TDataSource;
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure BitBtn2Click(Sender: TObject);
```

```

procedure FormActivate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form23: TForm23;

implementation

{$R *.DFM}

procedure TForm23.BitBtn2Click(Sender: TObject);
begin
  query1.Active := false;
  query2.Active := false;

  if (edit1.Text <>")
  then
    begin
      query1.ParamByName('booknames').asstring := edit1.Text+'%';
      query1.Active := true;
      query2.Active := true;

    end
  else
    begin
      query1.ParamByName('booknames').asstring := "";
    end
  end;

```

```

    query1.Active := true;
    query2.Active := true;
end;

label2.Caption := inttostr(query1.RecordCount);
if (query1.RecordCount = 0)
then
begin
    label4.Enabled := true;
    label5.Enabled := true;
end
else
begin
    label4.Enabled := false;
    label5.Enabled := false;
end;
end;

procedure TForm23.FormActivate(Sender: TObject);
begin
edit1.SetFocus;
end;

procedure TForm23.BitBtn1Click(Sender: TObject);
begin
close;
end;

end.

```

```
unit finwrit;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, Grids, DBGrids, ExtCtrls, DBTables, Db;
```

```
type
```

```
TForm24 = class(TForm)
```

```
Panel1: TPanel;
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label6: TLabel;
```

```
GroupBox1: TGroupBox;
```

```
DBGrid1: TDBGrid;
```

```
DBGrid2: TDBGrid;
```

```
Edit1: TEdit;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
GroupBox2: TGroupBox;
```

```
Label4: TLabel;
```

```
Label5: TLabel;
```

```
Label7: TLabel;
```

```
Table1: TTable;
```

```
DataSource1: TDataSource;
```

```
Query1: TQuery;
```

```
DataSource2: TDataSource;
```

```
Query2: TQuery;
```

```
DataSource3: TDataSource;
```

```

procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form24: TForm24;

implementation
  {$R *.DFM}

procedure TForm24.BitBtn1Click(Sender: TObject);
begin
close;
end;

procedure TForm24.BitBtn2Click(Sender: TObject);
begin
query1.Active := false;
  query2.Active := false;
  if (edit1.Text <>")
  then
    begin
      query1.ParamByName('writers').asstring := edit1.Text+'%';
      query1.Active := true;
      query2.Active := true;
    end;
  end;

```

```

end
else
begin
query1.ParamByName('writers').asString := "";
query1.Active := true;
query2.Active := true;
end;
label2.Caption := inttostr(query1.RecordCount);
if (query1.RecordCount = 0)
then
begin
label4.Enabled := true;
label5.Enabled := true;
end
else
begin
label4.Enabled := false;
label5.Enabled := false;
end;
end;

procedure TForm24.FormActivate(Sender: TObject);
begin
edit1.SetFocus;
end;

end.
unit tit;

```

interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่า **uses** ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, Buttons, Grids, DBGrids, ExtCtrls, DBTables, Db;

type

```
TForm25 = class(TForm)
```

```
  Panel1: TPanel;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  Label6: TLabel;
```

```
  GroupBox1: TGroupBox;
```

```
  DBGrid1: TDBGrid;
```

```
  DBGrid2: TDBGrid;
```

```
  Edit1: TEdit;
```

```
  BitBtn1: TBitBtn;
```

```
  BitBtn2: TBitBtn;
```

```
  GroupBox2: TGroupBox;
```

```
  Label4: TLabel;
```

```
  Label5: TLabel;
```

```
  Table1: TTable;
```

```
  DataSource1: TDataSource;
```

```
  Query1: TQuery;
```

```
  DataSource2: TDataSource;
```

```
  Query2: TQuery;
```

```
  DataSource3: TDataSource;
```

```
  Table2: TTable;
```

```
  DataSource4: TDataSource;
```

```
  Query3: TQuery;
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
  procedure FormActivate(Sender: TObject);
```

```
  procedure BitBtn2Click(Sender: TObject);
```

```
private
```

```

    { Private declarations }
public
    { Public declarations }
end;

var
    Form25: TForm25;

implementation

{$R *.DFM}

procedure TForm25.BitBtn1Click(Sender: TObject);
begin
close;
end;

procedure TForm25.FormActivate(Sender: TObject);
begin
edit1.SetFocus;
end;

procedure TForm25.BitBtn2Click(Sender: TObject);
begin
    query1.Active := false;
    query2.Active := false;

    if (edit1.Text <>")
    then
begin
query1.ParamByName('titels').asstring := edit1.Text+'%';

```

```

query1.Active := true;
query2.Active := true;

end
else
begin
query1.ParamByName('titels').asstring := "";
query1.Active := true;
query2.Active := true;
end;
label2.Caption := inttostr(query1.RecordCount);
if (query1.RecordCount = 0)
then
begin
label4.Enabled := true;
label5.Enabled := true;
end
else
begin
label4.Enabled := false;
label5.Enabled := false;
end;
end;
end.
end.

```

```
unit q;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs;
```

```
type
```

```
TForm26 = class(TForm)
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

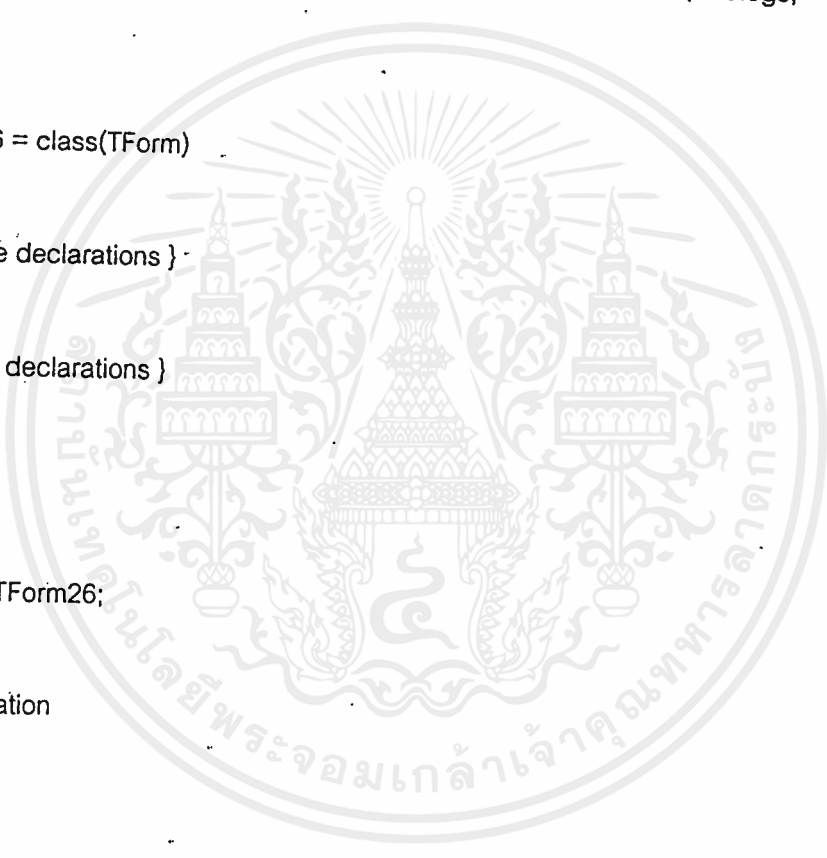
```
var
```

```
Form26: TForm26;
```

```
implementation
```

```
{ $R *.DFM }
```

```
end.
```



```
unit pass;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm27 = class(TForm)
```

```
  Panel1: TPanel;
```

```
  GroupBox1: TGroupBox;
```

```
  Edit1: TEdit;
```

```
  BitBtn1: TBitBtn;
```

```
  procedure Edit1Change(Sender: TObject);
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
end;
```

```
var
```

```
  Form27: TForm27;
```

```
implementation
```

```
uses offic;
```

```
{$R *.DFM}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm27.Edit1Change(Sender: TObject);
begin
  BitBtn1.Enabled := edit1.Text <> "";
end;

```

```

procedure TForm27.BitBtn1Click(Sender: TObject);
begin

```

```

  if (edit1.Text = 'kmitl')

```

```

  then

```

```

    begin

```

```

      form2.showmodal;

```

```

      close;

```

```

    end

```

```

  else

```

```

    begin

```

```

      edit1.Text := "";

```

```

    end;

```

```

  edit1.Text := "";

```

```

  close;

```

```

end;

```

```

end.

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit money;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBTables, Db, Grids, DBGrids, StdCtrls, ExtCtrls, Buttons;
```

```
type
```

```
TForm28 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
DBGrid1: TDBGrid;
```

```
Table1: TTable;
```

```
DataSource1: TDataSource;
```

```
Query1: TQuery;
```

```
DataSource2: TDataSource;
```

```
BitBtn1: TBitBtn;
```

```
procedure FormActivate(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form28: TForm28;
```

```
implementation
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 (\$R * DFM) ทั่วถึงกัน อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm28.FormActivate(Sender: TObject);
begin
  query1.Active := false;
  query1.Active := true;
end;

end.

unit medit;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, Buttons, ExtCtrls;

type
  TForm29 = class(TForm)
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    Panel1: TPanel;
  procedure BitBtn2Click(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

```

```

var
  Form29: TForm29;

implementation

uses editbk, editid;

{$R *.DFM}

procedure TForm29.BitBtn2Click(Sender: TObject);
begin
  form15.showmodal;
  form15.edit1.Clear;

  form15.edit4.Clear;
  form15.edit5.Clear;
  form15.edit6.Clear;
  form15.edit7.Clear;
  form15.edit8.Clear;
  form15.edit9.Clear;
  form15.edit10.Clear;
  form15.edit11.Clear;

  form15.DBEdit3.Clear;
  form15.DBEdit4.Clear;
  form15.DBEdit5.Clear;
  form15.DBEdit6.Clear;
  form15.DBEdit7.Clear;
  form15.DBEdit8.Clear;

```

```

form15.DBEdit9.Clear;
form15.DBEdit10.Clear;
end;

```

```

procedure TForm29.BitBtn1Click(Sender: TObject);
begin
form30.showmodal;
end;

end.

```

```

unit borrow;

```

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
DBTables, Db, StdCtrls, ExtCtrls, DBCtrls, Grids, DBGrids, Buttons;

```

```

type

```

```

TForm3 = class(TForm)

```

```

    Panel1: TPanel;

```

```

    GroupBox1: TGroupBox;

```

```

    Edit1: TEdit;

```

```

    Label1: TLabel;

```

```

    Button1: TButton;

```

```

    Database1: TDatabase;

```

```

    Table1: TTable;

```

```

    DataSource1: TDataSource;

```

```

    Query1: TQuery;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DataSource2: TDataSource;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 DBText1: TDBText;
 DBText2: TDBText;
 DBText3: TDBText;
 DBText4: TDBText;
 Table2: TTable;
 DataSource3: TDataSource;
 GroupBox2: TGroupBox;
 DBGrid1: TDBGrid;
 Query2: TQuery;
 DataSource4: TDataSource;
 GroupBox3: TGroupBox;
 Edit2: TEdit;
 Button2: TButton;
 Label6: TLabel;
 Label7: TLabel;
 Label8: TLabel;
 Label9: TLabel;
 Query3: TQuery;
 Label10: TLabel;
 Label11: TLabel;
 Label12: TLabel;
 Label13: TLabel;
 Label14: TLabel;
 Label15: TLabel;
 DBText5: TDBText;

DBText6: TDBText;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 DBText7: TDBText;
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DBText8: TDBText;
DBText9: TDBText;
DBText10: TDBText;
Table3: TTable;
DataSource5: TDataSource;
Query4: TQuery;
DataSource6: TDataSource;
Label16: TLabel;
DBText11: TDBText;
Query5: TQuery;
BitBtn1: TBitBtn;
Button3: TButton;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Edit1Change(Sender: TObject);
procedure Edit2Change(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;

implementation

```

```
procedure TForm3.Button1Click(Sender: TObject);
```

```
  var overDate : integer;
```

```
begin
```

```
  edit2.Enabled := false;
```

```
  button2.Enabled := false;
```

```
  query1.active := false;
```

```
  query1.ParamByName('ids').asString := edit1.text;
```

```
  query1.active := true;
```

```
  query2.active := false;
```

```
  query2.ParamByName('ids').asString := edit1.text;
```

```
  query2.active := true;
```

```
  label8.Caption := inttostr(query2.RecordCount);
```

```
  if (query2.recordcount < 3)
```

```
    then edit2.Enabled := true
```

```
    else
```

```
      begin
```

```
        edit2.Enabled := false;
```

```
        button2.Enabled := false;
```

```
      end;
```

```
  if (dbtext1.Field.text = "")
```

```
    then Edit2.Enabled := false;
```

```
  // check not over date
```

```
  if (query2.recordcount <> 0)
```

```
    then
```

```
      begin
```

```
        if (date > strtodate(query2.fields[3].text)) then overdate := 1
```

```
        then overdate := 1
```

```

        else overDate := 0;
    end;
if (query2.recordcount = 2)
then
begin
    query3.active := false;
    query3.ParamByName('ids').asString := edit1.text;
    query3.ParamByName('ids2').asString := query2.Fields[0].text;
    query3.active := true;
    if ((date > strtodate(query3.fields[1].text)) and (overDate = 0))
    then overDate := 1;
// label15.caption := query5.Fields[1].text;
end;
if ( overDate = 1)
then
begin
    showmessage('ท่านต้องทำการคืนหนังสือที่ท่านยืมมาเกินกำหนดก่อน จึงจะยืมเล่มต่อไปได้');
    edit2.Enabled := false;
    button2.Enabled := false;
end;

//end;
end;

procedure TForm3.Button2Click(Sender: TObject);
begin
if (query2.recordcount < 2)
then edit2.Enabled := true
else
begin
edit2.Enabled := false;

```

```

        button2.Enabled := false;
    end;

    query5.ParamByName('idbooks').asString := Edit2.Text;
    query5.ExecSQL;

    table2.insertrecord([edit1.text,edit2.text,
        datetostr(date),datetostr(date+7)]);

    query4.active := false;
    query4.ParamByName('idbooks').asString := edit2.text;
    query4.active := true;

    if (dbtext11.Field.Text = 'Y')
    then button2.Enabled := true
    else button2.Enabled := false;
    label8.Caption := inttostr(query2.recordcount);

end;

procedure TForm3.FormActivate(Sender: TObject);
begin
    edit1.Text := '';
    edit2.Text := '';
    Edit1.SetFocus;

    query2.Active :=false;

```

```
end;
```

```
procedure TForm3.FormCreate(Sender: TObject);
```

```
begin
```

```
    edit1.Clear;
```

```
    edit2.clear;
```

```
end;
```

```
procedure TForm3.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
    query1.Active := false;
```

```
    query4.Active := false;
```

```
end;
```

```
procedure TForm3.Button3Click(Sender: TObject);
```

```
begin
```

```
    query4.Active :=false;
```

```
    query4.ParamByName('idbooks').asstring := edit2.Text;
```

```
    query4.Active :=true;
```

```
    if (dbtext11.Field.Text = 'Y').
```

```
        then button2.Enabled := true
```

```
        else button2.Enabled := false;
```

```
end;
```

```
procedure TForm3.Edit1Change(Sender: TObject);
```

```
begin
```

```
    query1.Active := false;
```

```
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าในรูปแบบใด ๆ ทั้งสิ้น หากมีข้อผิดพลาดประการใด ขออภัยเป็นอย่างสูง และขอแจ้งถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm3.Edit2Change(Sender: TObject);
```

```

begin
query4.Active := false;
end;

end.

```

```

unit editid;

```

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
DBTables, Db, StdCtrls, Mask, DBCtrls, ExtCtrls, Buttons;

```

```

type

```

```

TForm30 = class(TForm)
  Panel1: TPanel;
  GroupBox1: TGroupBox;
  Edit1: TEdit;
  Button1: TButton;
  Table1: TTable;
  DataSource1: TDataSource;
  Query1: TQuery;
  DataSource2: TDataSource;
  Query2: TQuery;
  Query3: TQuery;

```

เอกสารนี้เป็น **Table2: TTable**; ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณี **DataSource3: TDataSource**; ลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Label4: TLabel;
 DBText1: TDBText;
 GroupBox2: TGroupBox;
 Edit2: TEdit;
 Label2: TLabel;
 Button2: TButton;
 BitBtn 1: TBitBtn;
 GroupBox3: TGroupBox;
 Edit3: TEdit;
 Button3: TButton;
 GroupBox4: TGroupBox;
 DBEdit1: TDBEdit;
 DBEdit2: TDBEdit;
 Edit4: TEdit;
 Edit5: TEdit;
 DataSource4: TDataSource;
 Query4: TQuery;
 DataSource5: TDataSource;
 Query5: TQuery;
 DataSource6: TDataSource;
 Button4: TButton;
 Table3: TTable;
 DataSource7: TDataSource;
 Query6: TQuery;
 Query7: TQuery;
 Label1: TLabel;
 Label3: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 Label7: TLabel;

เอกสารนี้ procedure Button1Click(Sender: TObject); ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใด procedure Button2Click(Sender: TObject); ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure BitBtn1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form30: TForm30;

implementation

{$R *.DFM}

procedure TForm30.Button1Click(Sender: TObject);
begin
  query1.Active := false;
  query1.ParamByName('calls').asstring := edit1.Text;
  query1.Active := true;

  edit2.SetFocus;
end;

procedure TForm30.Button2Click(Sender: TObject);
begin
  query2.Active := false;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะในรูปแบบใดก็ตาม หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (edit2.Text <> "")
  then
    begin
      query2.ParamByName('s').asString := dbtext1.Field.Text;
      query2.ParamByName('calls').asString := edit2.text;

      query3.ParamByName('s').asString := dbtext1.Field.Text;
      query3.ParamByName('calls').asString := edit2.text;

      query2.ExecSQL;
      query3.ExecSQL;
    end
  else
    begin
      query2.ParamByName('s').asString := dbtext1.Field.Text;
      query2.ParamByName('calls').asString := dbtext1.Field.Text;

      query3.ParamByName('s').asString := dbtext1.Field.Text;
      query3.ParamByName('calls').asString := dbtext1.Field.Text;

      query2.ExecSQL;
      query3.ExecSQL;
    end;

    query1.Active := false;
    query1.ParamByName('calls').asstring := edit2.Text;
    query1.Active := true;

```

```

edit1.SetFocus;
end;

```

```

procedure TForm30.BitBtn1Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
query1.Active := false;
end;

```

```

procedure TForm30.FormActivate(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
edit5.Clear;
query1.Active := false;
query4.Active := false;
query5.Active := false;
end;

```

```

procedure TForm30.Button3Click(Sender: TObject);
begin
query4.Active := false;
query4.ParamByName('idbooks').asstring := edit3.Text;
query4.Active := true;

```

```

query5.Active := false;

```

เอกสารนี้เป็นทรัพย์สินของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ กรุณาแจ้งที่ โทร. 02-261-6000 หรือที่ อีเมล: info@kmitl.ac.th

```
query5.Active := true;
```

```
end;
```

```
procedure TForm30.Button4Click(Sender: TObject);
```

```
begin
```

```
    query6.active := false;
```

```
    if (edit4.Text <> "")
```

```
    then
```

```
        begin
```

```
            query6.ParamByName('yes').asString := dbedit1.Field.Text;
```

```
            query6.ParamByName('idbooks').asString := edit3.Text;
```

```
            query6.ParamByName('titels').asString := edit4.text;
```

```
            query6.ExecSQL;
```

```
        end
```

```
    else
```

```
        begin
```

```
            query6.ParamByName('idbooks').asString := edit3.Text;
```

```
            query6.ParamByName('yes').asString := dbedit1.Field.Text;
```

```
            query6.ParamByName('titels').asString := dbedit1.Field.Text;
```

```
            query6.ExecSQL;
```

```
        end;
```

```
    query7.active := false;
```

```
    if (edit5.Text <> "")
```

```
    then
```

```
        begin
```

```
            query7.ParamByName('yes').asString := dbedit2.Field.Text;
```

```
            query7.ParamByName('idbooks').asString := edit3.Text;
```

```
            query7.ParamByName('titels').asString := edit5.text;
```

```
            query7.ExecSQL;
```

```
        end;
```

```

else
  begin

query7.ParamByName('idbooks').asString := edit3.Text;
query7.ParamByName('yes').asString := dbedit2.Field.Text;
query7.ParamByName('titels').asString := dbedit2.Field.Text;
query7.ExecSQL;
  end;

query4.Active := false;
query4.ParamByName('idbooks').asstring := edit3.Text;
query4.Active := true;
edit4.Clear;

query5.Active := false;
query5.ParamByName('idbooks').asstring := edit3.Text;
query5.ParamByName('nots').asstring := dbedit1.Field.Text;
query5.Active := true;
edit5.Clear;

edit3.SetFocus;

end;

end.

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit return;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, ExtCtrls, DBTables, Db, DBCtrls, Mask, Buttons;
```

```
type
```

```
TForm4 = class(TForm)
```

```
  Panel1: TPanel;
```

```
  GroupBox1: TGroupBox;
```

```
  Edit1: TEdit;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  Label4: TLabel;
```

```
  Label5: TLabel;
```

```
  Label6: TLabel;
```

```
  Label7: TLabel;
```

```
  Label8: TLabel;
```

```
  Label9: TLabel;
```

```
  DBText3: TDBText;
```

```
  DBText4: TDBText;
```

```
  DBText5: TDBText;
```

```
  DBText6: TDBText;
```

```
  DBText7: TDBText;
```

```
  DBText8: TDBText;
```

```
  Table1: TTable;
```

เอกสารนี้ **DataSource1: TDataSource**; ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณี **Query1: TQuery**; ห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DataSource2: TDataSource;
Panel2: TPanel;
GroupBox2: TGroupBox;
Label10: TLabel;
Label11: TLabel;
Label12: TLabel;
DBText9: TDBText;
DBText10: TDBText;
Label13: TLabel;
Label14: TLabel;
Label15: TLabel;
Label16: TLabel;
DBText1: TDBText;
Panel3: TPanel;
GroupBox3: TGroupBox;
Button2: TButton;
Query2: TQuery;
Table2: TTable;
DataSource3: TDataSource;
Query3: TQuery;
DBText2: TDBText;
BitBtn1: TBitBtn;
Button1: TButton;
Table3: TTable;
procedure Button1Click(Sender: TObject);
procedure DBEdit1Change(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Edit1Change(Sender: TObject);
private
  { Private declarations }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าการตีพิมพ์หรือการเผยแพร่ให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;

var
  Form4: TForm4;

implementation

{$R *.DFM}

procedure TForm4.Button1Click(Sender: TObject);
  var i : integer;
begin
  query1.Active :=false;
  query1.ParamByName('idbooks').asstring := edit1.text;
  query1.Active := true;

  label13.Caption := datetostr(date);

  if (dbtext1.Field.text <> "")
  then
    begin
      label15.caption:='0';
      if ( date > strtodate(dbtext10.Field.text))
      then
        begin
          i:=0;
          while date <> strtodate(dbtext10.Field.Text)+i
          do i:= i+1;
          label15.caption:=inttostr(i);
        end;
      end;
    end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
```

```
procedure TForm4.DBEdit1Change(Sender: TObject);
```

```
var i : integer;
```

```
begin
```

```
//if (dbedit1.text <> ")
```

```
// then
```

```
// begin
```

```
// label15.caption:='0';
```

```
// if ( date > strtodate(dbtext10.Field.text))
```

```
// then
```

```
// begin
```

```
// i:=0;
```

```
// while date <> strtodate(dbtext10.Field.Text)+i
```

```
// do i := i+1;
```

```
// label15.caption:=inttostr(i);
```

```
// end;
```

```
// end;
```

```
end;
```

```
procedure TForm4.Button2Click(Sender: TObject);
```

```
var m : integer;
```

```
begin
```

```
query3.active := false;
```

```
query3.ParamByName('idbooks').asString := edit1.text;
```

```
query3.ExecSQL;
```

```
query2.active := false;
```

```
query2.ParamByName('idbooks').asString := edit1.text;
```

```
query2.ExecSQL;
```

```

m := strtoint(label15.Caption);
if (m > 0)
then
begin
table3.insertrecord([(dbtext1.field.text,edit1.text,
dbtext2.field.text,label15.caption)]);
end;

query1.Active :=false;
edit1.Clear;
Label13.Caption := "";
label15.Caption := "";
end;

procedure TForm4.Edit1Change(Sender: TObject);
var i : integer;
begin
query1.Active := false;

end;

end.

```

```
unit mbook;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm5 = class(TForm)
```

```
Panel1: TPanel;
```

```
GroupBox1: TGroupBox;
```

```
BitBtn1: TBitBtn;
```

```
BitBtn2: TBitBtn;
```

```
BitBtn3: TBitBtn;
```

```
BitBtn4: TBitBtn;
```

```
Panel2: TPanel;
```

```
procedure BitBtn1Click(Sender: TObject);
```

```
procedure BitBtn2Click(Sender: TObject);
```

```
procedure BitBtn3Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form5: TForm5;
```

```
implementation
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ **uses addst, remst, editst;** ที่ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{$R *.DFM}
```

```
procedure TForm5.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
form6.showmodal;
```

```
form6.Edit1.Clear;
```

```
form6.Edit2.Clear;
```

```
form6.Edit3.Clear;
```

```
form6.Edit4.Clear;
```

```
form6.Edit5.Clear;
```

```
form6.label6.Enabled := false;
```

```
form6.label7.Enabled := false;
```

```
end;
```

```
procedure TForm5.BitBtn2Click(Sender: TObject);
```

```
begin
```

```
form7.showmodal;
```

```
form7.Edit1.Text := '';
```

```
form7.dbedit1.Clear;
```

```
form7.dbedit2.Clear;
```

```
form7.dbedit3.Clear;
```

```
form7.dbedit4.Clear;
```

```
end;
```

```
procedure TForm5.BitBtn3Click(Sender: TObject);
```

```
begin
```

```
form10.showmodal;
```

```
form10.edit1.Clear;
```

```
form10.edit2.Clear;
```

```
form10.edit3.Clear;
```

เอกสารนี้จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ หากมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

form10.edit4.Clear;
form10.edit5.Clear;
form10.dbedit1.Clear;
form10.dbedit2.Clear;
form10.dbedit3.Clear;
form10.dbedit4.Clear;
end;

```

```
end.
```

```
unit addst;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
DBTables, Db, StdCtrls, DBCtrls, ExtCtrls, Buttons;
```

```
type
```

```
TForm6 = class(TForm)
```

```
Panel1: TPanel;
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label4: TLabel;
```

```
DBText1: TDBText;
```

```
Label9: TLabel;
```

```
Edit1: TEdit;
```

```

Edit2: TEdit;
Edit3: TEdit;
Button1: TButton;
Edit4: TEdit;
Button3: TButton;
Panel3: TPanel;
Label6: TLabel;
Label7: TLabel;
Edit5: TEdit;
Label5: TLabel;
Table1: TTable;
DataSource1: TDataSource;
Query1: TQuery;
Label8: TLabel;
DataSource2: TDataSource;
BitBtn1: TBitBtn;
procedure Button1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form6: TForm6;

implementation

{$R *.DFM}

```

```

procedure TForm6.Button1Click(Sender: TObject);
begin
    query1.ParamByName('ids').asString := edit1.text;
    query1.active := true ;
    if (dbtext1.Field.Text = " )
    then
        begin
            if (edit1.text = "")or(edit2.text=")
            or(edit3.text = "")or(edit4.text=")
            //or(edit5.text = ")
            then
                begin
                    label6.Enabled := true;
                    label7.Enabled := false;
                end
            else
                begin
                    table1.insertrecord([edit1.text,edit2.text,
                    edit3.text,edit4.text,edit5.text]);
                    label6.Enabled := false;
                    label7.Enabled := true;
                    Edit1.clear;
                    Edit2.clear;
                    Edit3.clear;
                    Edit4.clear;
                    edit5.Clear;
                end
            end
        end
    else
        begin

```

```

label7.Enabled := false;

    Edit1.clear;

    Edit2.clear;

    Edit3.clear;

    Edit4.clear;

    Edit5.Clear;

query1.active := false ;

    end;

    Edit1.clear;

    Edit2.clear;

    Edit3.clear;

    Edit4.clear;

    Edit5.Clear;

query1.active := false ;

end;

procedure TForm6.Button3Click(Sender: TObject);
begin
    Edit1.clear;

    Edit2.clear;

    Edit3.clear;

    Edit4.clear;

    Edit5.Clear;

    label6.Enabled := false;

    label7.Enabled := false;

end;

end.

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit remst;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBTables, Db, Mask, DBCtrls, StdCtrls, ExtCtrls, Grids, DBGrids, Buttons;
```

```
type
```

```
TForm7 = class(TForm)
```

```
  Panel2: TPanel;
```

```
  GroupBox1: TGroupBox;
```

```
  Label5: TLabel;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  DBText1: TDBText;
```

```
  Edit1: TEdit;
```

```
  Button1: TButton;
```

```
  Button2: TButton;
```

```
  DBEdit1: TDBEdit;
```

```
  DBEdit2: TDBEdit;
```

```
  DBEdit3: TDBEdit;
```

```
  Button5: TButton;
```

```
  Label4: TLabel;
```

```
  DBEdit4: TDBEdit;
```

```
  Label6: TLabel;
```

```
  Table1: TTable;
```

```
  DataSource1: TDataSource;
```

```
  Query1: TQuery;
```

```
  DataSource2: TDataSource;
```

```
  Table2: TTable;
```

```

DataSource3: TDataSource;
Query2: TQuery;
DataSource4: TDataSource;
Panel4: TPanel;
GroupBox2: TGroupBox;
Label8: TLabel;
Label10: TLabel;
Edit2: TEdit;
Button3: TButton;
Button4: TButton;
GroupBox3: TGroupBox;
DBGrid1: TDBGrid;
Query3: TQuery;
DataSource5: TDataSource;
Edit3: TEdit;
Label7: TLabel;
Button6: TButton;
BitBtn1: TBitBtn;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Edit2Change(Sender: TObject);
procedure Edit3Change(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Edit1Change(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

```

```

var
  Form7: TForm7;

implementation

uses wst, wstall;

{$R *.DFM}

procedure TForm7.Button1Click(Sender: TObject);
begin
  if (dbedit1.Field.Text <> "")
  then
    begin
      form8.showmodal;
    end;
end;

end;

procedure TForm7.Button2Click(Sender: TObject);
begin
  edit1.Text := "";
  dbedit1.Clear;
  dbedit2.Clear;
  dbedit3.Clear;
  dbedit4.Clear;
end;

```

```

procedure TForm7.Button6Click(Sender: TObject);
begin
query3.Active := false;
query3.ParamByName('layers').asString := edit2.Text;
query3.ParamByName('rooms').asString := edit3.Text;
query3.Active := true;
end;

```

```

procedure TForm7.Button4Click(Sender: TObject);
begin
edit2.Text := "";
edit3.Text := "";
query3.Active := false;
query3.ParamByName('layers').asString := edit2.Text;
query3.ParamByName('rooms').asString := edit3.Text;
query3.Active := true;
end;

```

```

procedure TForm7.Button3Click(Sender: TObject);
begin
if (edit2.Text <> "")
then
begin
form9.showmodal;

end;
end;

```

```

procedure TForm7.Edit2Change(Sender: TObject);

```

```

begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าในรูปแบบใดๆ หากมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
```

```
procedure TForm7.Edit3Change(Sender: TObject);
```

```
begin
```

```
query3.Active := false;
```

```
end;
```

```
procedure TForm7.Button5Click(Sender: TObject);
```

```
begin
```

```
query1.Active := false;
```

```
query1.ParamByName('ids').asString := edit1.Text;
```

```
query1.Active := true;
```

```
query2.Active := false;
```

```
query2.ParamByName('ids').asString := edit1.Text;
```

```
query2.Active := true;
```

```
if (dbtext1.Field.Text = "")
```

```
then
```

```
begin
```

```
button1.Enabled := true;
```

```
button1.SetFocus;
```

```
end
```

```
else
```

```
begin
```

```
button1.Enabled :=false;
```

```
showmessage('นักศึกษายังค้างหนังสือ');
```

```
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm7.Edit1Change(Sender: TObject);
```

```
begin
```

```
dbedit1.Clear;
```

```
dbedit2.Clear;
```

```
dbedit3.Clear;
```

```
dbedit4.Clear;
```

```
button1.Enabled := false;
```

```
end;
```

```
end.
```

```
unit wst;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBTables, Db, Buttons, StdCtrls, ExtCtrls;
```

```
type
```

```
TForm8 = class(TForm)
```

```
Panel1: TPanel;
```

```
Button1: TButton;
```

```
GroupBox1: TGroupBox;
```

```
Label2: TLabel;
```

```
BitBtn1: TBitBtn;
```

```
Table1: TTable;
```

```

DataSource1: TDataSource;
Query1: TQuery;
procedure Button1Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form8: TForm8;

implementation

uses remst;

{$R *.DFM}

procedure TForm8.Button1Click(Sender: TObject);
begin
  query1.ParamByName('ids').asString := form7.edit1.text;
  query1.ExecSQL;
  form7.Edit1.Text := '';
  form7.dbedit1.Clear;
  form7.dbedit2.Clear;
  form7.dbedit3.Clear;
  form7.dbedit4.Clear;

```

close;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะในรูปแบบใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm8.BitBtn1Click(Sender: TObject);
begin
    close;
end;

end.

```

```

unit wstall;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    Buttons, StdCtrls, ExtCtrls, DBTables, Db;

type
    TForm9 = class(TForm)
        Panel1: TPanel;
        Button1: TButton;
        GroupBox1: TGroupBox;
        Label2: TLabel;
        BitBtn1: TBitBtn;
        Table1: TTable;
        DataSource1: TDataSource;
        Query1: TQuery;
        procedure Button1Click(Sender: TObject);
        procedure BitBtn1Click(Sender: TObject);
    end;

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form9: TForm9;

implementation

uses remst;

{$R *.DEM}
procedure TForm9.Button1Click(Sender: TObject);
begin
  query1.ParamByName('layers').asString := form7.edit2.text;
  query1.ParamByName('rooms').asString := form7.edit3.text;
  query1.ExecSQL;

  form7.Edit2.Text := "";
  form7.Edit3.Text := "";
  form7.Query3.Active := false;

close;
end;

procedure TForm9.BitBtn1Click(Sender: TObject);
begin
  form7.Edit2.Clear;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
form7.Edit3.Clear;  
form7.Query3.Active := false;  
end;  
  
end.
```



บทที่ 7

ความสามารถของโปรแกรมการจัดการห้องสมุด

บริการยืมหนังสือ

ในการให้บริการยืมหนังสือความต้องการพื้นฐานคือลดความยุ่งยาก ลดขั้นตอนของการเก็บข้อมูลผู้ยืมและข้อมูลหนังสือที่ถูกยืมเจ้าหน้าที่เพียงแต่ป้อนรหัสนักศึกษาที่ยืมและเลขทะเบียนหนังสือที่ถูกยืมเข้าไปเท่านั้น โปรแกรมจะแสดงผลชื่อ นามสกุล และ ห้อง ซึ่งเป็นข้อมูลของผู้ยืม แสดงเลขหมู่หนังสือ ชื่อหนังสือ ชื่อผู้แต่ง และ สถานะการยืมว่ายืมได้หรือไม่ ซึ่งเป็นข้อมูลหนังสือที่ถูกยืม ออกทางหน้าจอทันที จะเห็นได้ว่าเจ้าหน้าที่ทำงานเพียงสองขั้นตอนเท่านั้นก็เสร็จสิ้นการให้บริการยืมหนังสือ ส่วนที่สำคัญอีกส่วนหนึ่งคือการตรวจสอบว่านักศึกษาที่จะยืมหนังสือนั้นมีสถานะที่จะยืมได้หรือไม่และหนังสือที่จะยืมมีสถานะยืมได้หรือไม่ ถ้านักศึกษาที่ยืมยังไม่ได้ส่งคืนหนังสือตามกำหนดเวลาจากการยืมครั้งก่อนหรือได้ยืมหนังสือครบสามเล่มแล้วก็จะไม่สามารถทำการยืมอีกได้โดยอัตโนมัติ รวมทั้งแสดงรายการหนังสือที่ยังค้างไว้อีกด้วยและถ้าหนังสือที่ผู้ยืมต้องการถูกคนอื่นยืมไปแล้วก็ไม่สามารถทำการยืมได้เช่นกัน

บริการคืนหนังสือ

การให้บริการการคืนหนังสือนั้นมีอยู่ขั้นตอนเดียวคือการป้อนเลขทะเบียนหนังสือที่คืน โปรแกรมจะแสดงชื่อนามสกุลนักศึกษา รหัสนักศึกษา และ ห้อง ซึ่งเป็นข้อมูลของผู้ยืม แสดงเลขหมู่หนังสือ ชื่อหนังสือ และ ชื่อผู้แต่ง ซึ่งเป็นข้อมูลของหนังสือที่นำมาคืน รวมทั้งแจ้งวันที่ยืมหนังสือ วันที่จะต้องส่งคืนหนังสือ วันที่นักศึกษานำหนังสือมาส่งคืนและจำนวนเงินค่าปรับในกรณีที่นักศึกษาส่งคืนหนังสือช้ากว่ากำหนด

บริการเพิ่ม/ลบ/แก้ไข หนังสือ

ประกอบไปด้วยการเพิ่ม การลบและการแก้ไขรายการหนังสือจากฐานข้อมูลโดยที่การเพิ่มรายการหนังสือโปรแกรมจะกำหนดเลขทะเบียนหนังสือให้

บริการเพิ่ม/ลบ/แก้ไข นักศึกษา

ประกอบไปด้วยการเพิ่ม การลบและการแก้ไขรายชื่อนักศึกษาจากฐานข้อมูล

บริการตรวจสอบทั้งหมด

เป็นการตรวจสอบข้อมูลนักศึกษาและข้อมูลหนังสือทั้งหมดที่มีอยู่ในฐานข้อมูล

บริการค้นหาหนังสือ

สามารถค้นหาได้ทั้งแบบตามชื่อหนังสือและแบบตามชื่อผู้แต่ง

บทที่ 8

บทสรุป

8.1 ผลการทดลอง

ผลที่ได้จากการรันโปรแกรม เป็นไปตาม ขั้นตอน เริ่มตั้งแต่การป้อนรหัสผ่านเข้าโปรแกรม สำหรับเจ้าหน้าที่เข้าไปยังส่วนของการเพิ่ม/ลบ นักศึกษา เพิ่มรายชื่อนักศึกษาและข้อมูลต่างๆ เข้าไปเพิ่มรายชื่อและข้อมูลของหนังสือใหม่ในส่วนการเพิ่ม/ลบ หนังสือ ดูผลการเพิ่มรายชื่อนักศึกษาและรายชื่อหนังสือจากเมนู ตรวจสอบทั้งหมด จากนั้นทดสอบในส่วนของการยืมโดยการป้อนรหัสนักศึกษาและเลขทะเบียนหนังสือที่ต้องการยืม ดูว่าการทำงานของโปรแกรมถูกต้องหรือไม่ โดยการทดลองยืมหนังสือที่ถูกยืมไปแล้วจะพบว่าไม่สามารถยืมได้ ตรวจสอบการทำงานของโปรแกรมในส่วนของการคืนโดยป้อนเลขทะเบียนหนังสือ โปรแกรมจะบอกข้อมูลเกี่ยวกับผู้ยืม, ข้อมูลหนังสือ, วันเวลาที่ยืมและคืน และ ค่าปรับในกรณีที่คืนหนังสือเกินกำหนดเวลาโดยอัตโนมัติ ตรวจสอบว่าได้คืนหนังสือไปเรียบร้อยแล้วหรือไม่ โดยการย้อนกลับไปยังส่วนของการคืนซึ่งจะบอกรายการหนังสือที่ค้าง จากนั้นทำการทดสอบการลบรายชื่อนักศึกษาและรายชื่อหนังสือแล้วดูผลการทดสอบจากเมนู ตรวจสอบทั้งหมด

ส่วนสุดท้ายที่ทำการทดลองคือเมนูการค้นหาหนังสือทั้งการค้นหาตามชื่อหนังสือ ค้นหาตามชื่อผู้แต่ง และ ค้นหาตามชื่อหัวเรื่อง โปรแกรมจะแจ้งให้ทราบว่าหนังสือที่ต้องการอยู่ในห้องสมุดหรือไม่ หรือหนังสือที่ต้องการถูกยืมไปแล้วหรือไม่ ผลการทดลองส่วนต่างๆ ของโปรแกรมพบว่าสามารถทำงานได้อย่างถูกต้องแม่นยำและเป็นไปตามจุดมุ่งหมายที่ตั้งไว้

8.2 การวิเคราะห์ปัญหาที่เกิดขึ้น

1. ในการเขียนโปรแกรมเราจะใช้โปรแกรม เดลไฟล์ 3 เพื่อใช้ในการสร้างฐานข้อมูล ซึ่งจะเป็นโปรแกรมที่ใหญ่ใช้เนื้อที่หน่วยความจำมากจึงอาจเกิดปัญหาในการเรียกใช้งานบนPCที่มีความเร็วต่ำและหน่วยความจำน้อย ซึ่งจะทำให้การทดสอบการทำงานเกิดความล่าช้า

2. ปัญหาของหนังสืออ้างอิงเกี่ยวกับการเขียนแอฟลิเคชันโปรแกรมไม่ละเอียดพอ ในการศึกษา ค้นคว้า ขาดตัวอย่างในการนำไปใช้งาน และการเขียนโปรแกรมเดลไฟล์ ยังใหม่สำหรับนักศึกษา

3. จากการที่ต้องเขียนโปรแกรม เดลไฟล์ ในลักษณะของผู้หัดใหม่ที่เริ่มเรียนรู้ คำสั่งบางคำสั่งยังศึกษาไม่ละเอียดรอบคอบพอ ดังนั้นเมื่อเริ่มเขียนโปรแกรมจึงประสบปัญหาที่ต้องทำการแก้ไขใหม่อยู่เรื่อยๆ เพื่อให้เกิดความสมบูรณ์ของโปรแกรมมากที่สุด ระยะเวลาในการทำงานจึงหมดไปในส่วนนี้มาก

8.3 การพัฒนาและปรับปรุงโปรแกรมต่อไปในอนาคต

1. เพิ่มระบบบาร์โค้ดเพื่อความสะดวกและลดข้อผิดพลาดของเจ้าหน้าที่ในการป้อนข้อมูล
2. เพิ่มการพิมพ์ข้อมูลการยืม คืน ข้อมูลหนังสือและข้อมูลของนักศึกษาออกทางเครื่องพิมพ์เพื่อใช้เป็นหลักฐานในการตรวจสอบ
3. พัฒนาให้โปรแกรมลบข้อมูลนักศึกษาทั้งห้องได้ โดยที่สามารถแจ้งรายชื่อนักศึกษาที่ยังไม่ได้คืนหนังสือตามกำหนด



เอกสารอ้างอิง

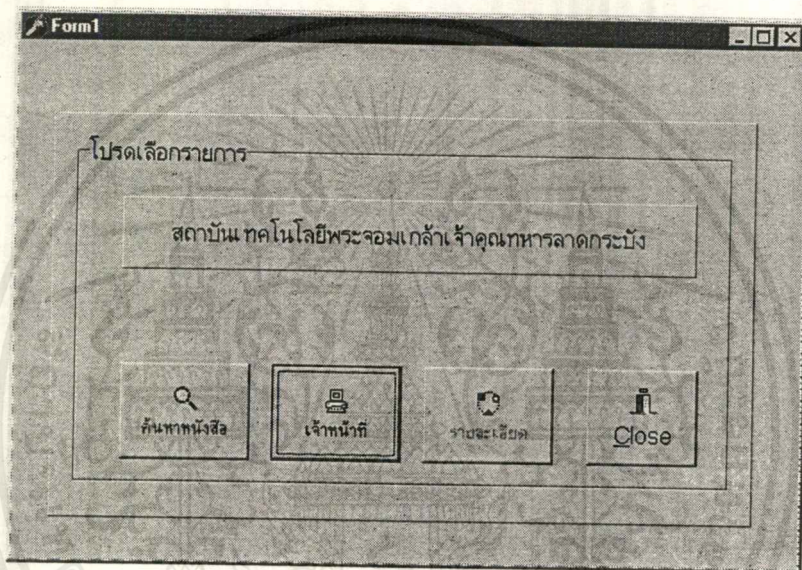
1. Delphi บุญเลิศ เอี่ยมทัศนาศนา โดย บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน) ISBN 974-512-425-7
- 2 .แรกเริ่มการเรียนรู้เรื่องการเขียนโปรแกรมด้วย Delphi เรียบเรียงจาก Delphi Programming for Dummies ของ Neil J.Rubenking for Dummies เรียบเรียง โดย ทนงศักดิ์ บรรจงมณี ISBN 974-521-760-4.
3. นำทางสู่ระบบฐานข้อมูลแบบ ไคลเอนต์/เซิร์ฟเวอร์ เรียบเรียงจาก Guide to Client / Server Database ของ JOE SALEMI เรียบเรียงโดย โชคชัย เดชพรุ่ง ISBN 974-512-474-5



คู่มือการใช้งาน

การใช้งานโปรแกรมบริการห้องสมุด

เมื่อเรียกโปรแกรมขึ้นมาใช้งานจะปรากฏ main menu แสดงเพื่อให้ผู้ใช้ได้เลือกรายการบริการ
ดังแสดงไว้ในรูปที่ 1



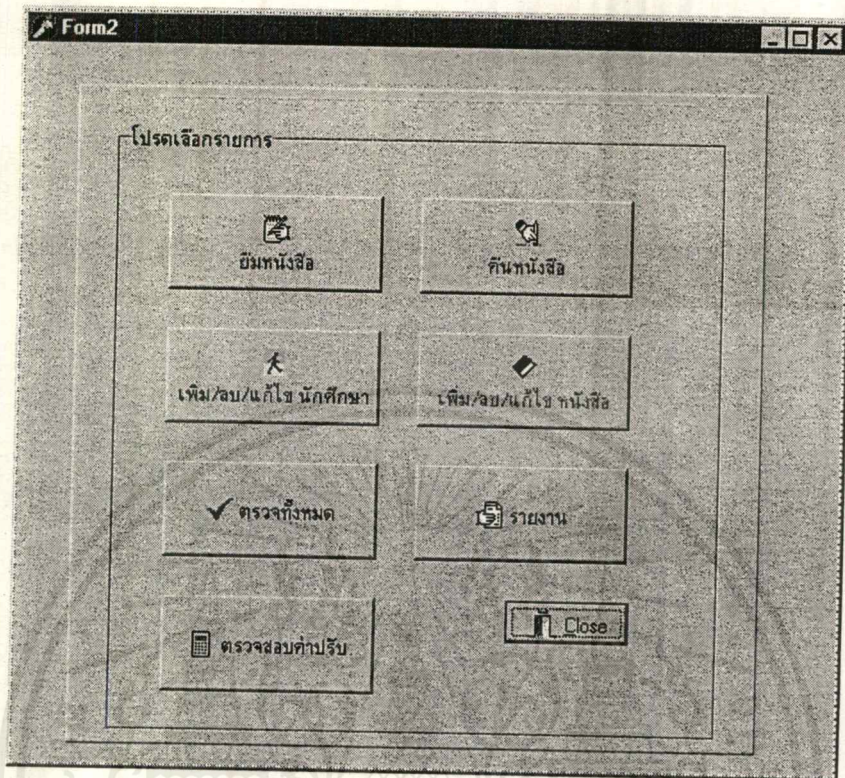
รูปที่ 1 หน้าจอ main menu

งานบริการยืม-คืน

จากจอ main menu ในรูปหากเราต้องการใช้งานบริการยืมคืนหนังสือทำได้โดยการคลิกที่ปุ่ม

" เจ้าหน้าที่ " หน้าจอจะแสดงส่วนของเจ้าหน้าที่ซึ่งจะต้องกรอก password โดยพิมพ์คำว่า

" kmitl " จะปรากฏ menu งานบริการยืม คืน เพิ่ม/ลบ/แก้ไขหนังสือ เพิ่ม/ลบ/แก้ไขนักศึกษา ตรวจ
ทั้งหมด ,รายงาน, ตรวจค่าปรับ และ Close ดังแสดงในรูปที่ 2



รูปที่ 2 หน้าจอแสดงตัวเลือกของงานบริการยืม-คืน

การใช้งานบริการยืมหนังสือ

เมื่อคลิกปุ่ม " ยืมหนังสือ " โดยผ่านทางเมาส์ หรือ ทางคีย์บอร์ดซึ่งจะได้หน้าจอที่รอรับรหัสสมาชิกดังรูปที่ 3 เสร็จแล้วโปรแกรมจะทำงานดังนี้

- จากรูปที่ 3 โปรแกรมจะรอรับรหัสสมาชิกซึ่งสามารถป้อนรหัสผ่านทางคีย์บอร์ดเมื่อทำการป้อนรหัสสมาชิกเสร็จเรียบร้อยแล้วโปรแกรมจะทำการค้นหาและเอารายละเอียดของสมาชิกมาแสดงดังรูป แต่ถ้าทำการป้อนรหัสสมาชิกผิด หรือรหัสสมาชิกที่ป้อนไม่มีอยู่ในห้องสมุด โปรแกรมจะทำการรอรับรหัสสมาชิกใหม่ พร้อมทั้งจะมีการแสดงจำนวนของหนังสือ ชื่อหนังสือ วันที่ยืม และ วันที่คืนหนังสือไว้ด้วย

The screenshot shows a window titled 'Form3' with a 'Close' button. It is divided into two main sections:

- รหัสนักศึกษา (Student ID):** A text box containing '39012025' and an 'OK' button.
- เลขทะเบียนหนังสือ (Book Registration Number):** An empty text box and an 'OK' button.
- ข้อมูลนักศึกษา (Student Information):**
 - ชื่อนักศึกษา: วรุณี มีจุ่น
 - ชั้นปี: 2
 - กัณ: 1
 - ที่อยู่นักศึกษา: kmill
- ข้อมูลหนังสือ (Book Information):**
 - ชื่อหนังสือ: (empty)
 - เลขหมู่: (empty)
 - ชื่อผู้แต่ง: (empty)
 - สำนักพิมพ์: (empty)
 - ครั้งที่พิมพ์: (empty)
 - ราคา: (empty)
 - สถานะ: (empty)
- หนังสือที่ค้าง (Overdue Books):**
 - กัณหนังสือทั้งหมด: 1 เล่ม
 - Table with columns: BOOKNAME, IDBOOK, CALL, BORROW, RETURN.

BOOKNAME	IDBOOK	CALL	BORROW	RETURN
การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี	2	la200	5/26/98	6/2/98

รูปที่ 3 หน้าจองานบริการยืมหนังสือ

- จากรูปที่ 3 โปรแกรมจะให้ป้อนรหัสนักศึกษาเมื่อป้อนเสร็จเรียบร้อยแล้วก็จะปรากฏรายละเอียดเกี่ยวกับนักศึกษาคนนั้นออกมา พร้อมกับแสดงจำนวนหนังสือและรายละเอียดเกี่ยวกับหนังสือที่ติดค้างไว้ด้วย
- การยืมหนังสือสามารถทำได้โดยป้อนรหัสหนังสือผ่านทางคีย์บอร์ดเข้าไปในหน้าจอแสดงเลขทะเบียนหนังสือ เมื่อป้อนเลขทะเบียนหนังสือเสร็จก็จะปรากฏรายละเอียดเกี่ยวกับหนังสือ คือ ชื่อหนังสือ เลขหมู่หนังสือ ชื่อของผู้แต่ง สำนักพิมพ์ ครั้งที่พิมพ์ ราคา และ สถานะของการยืมว่าสามารถที่จะยืมได้หรือไม่ ถ้ายืมได้จะปรากฏเป็นตัว Y แต่ถ้ายืมไม่ได้จะแสดงเป็นตัว N ถ้าหากมีการติดค้างหนังสือไว้จำนวน 3 เล่มจะมี menu ฟ้องขึ้นมาว่า " ท่านจะต้องคืนหนังสือที่ท่านยืมมาเกินกำหนดก่อนท่านจึงสามารถยืมเล่มต่อไปได้ " และจะไม่สามารถที่จะคีย์เลขทะเบียนหนังสือได้ในกรณีที่ยืมได้ก็คลิกไปที่ "ยืม" การยืมก็จะเสร็จเรียบร้อย

งานบริการการคืนหนังสือ

การใช้งานบริการการคืนหนังสือจากหน้าจอรูปที่ 2 เมื่อเราคลิกไปที่ปุ่ม "คืนหนังสือ" จะปรากฏหน้าจอที่จะรอรับรหัสหนังสือดังรูปที่ 4 แล้วโปรแกรมจะทำงานดังต่อไปนี้

Form4

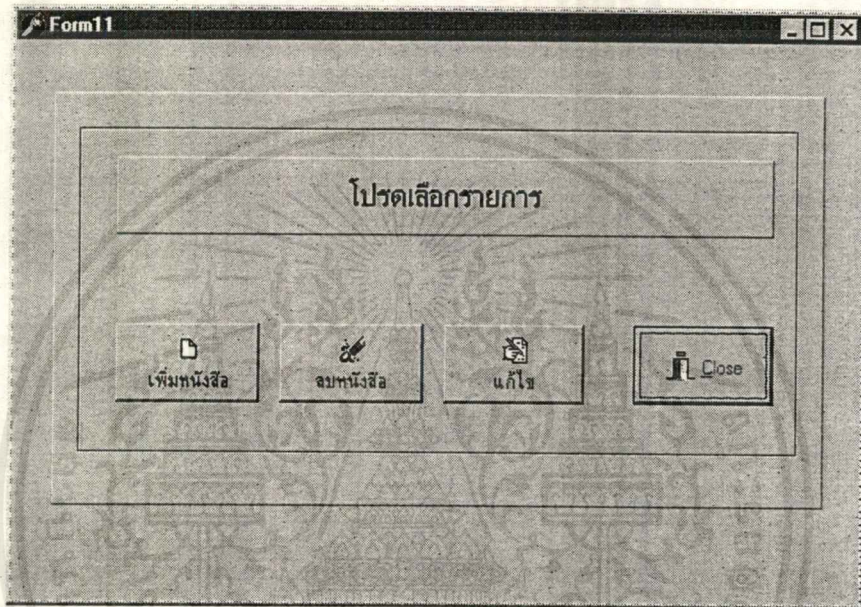
เลขทะเบียนหนังสือ		วันที่ยืมหนังสือ	6/1/98
2	OK	วันที่จะต้องส่งคืน	6/8/98
รหัสนักศึกษา	39012025	นักศึกษาส่งหนังสือวันที่	6/1/98
ชื่อนักศึกษา	วรวิทย์ ยี่สุน	สำหรับ	0 บาท
ชื่อหนังสือ	การเขียนโปรแกรม		
เลขเรียกหนังสือ	๒๒200		
ชื่อผู้แต่ง	มนตรี พจนาน		
สำนักพิมพ์	ซีเอ็ดยูเคชั่น		
ครั้งที่พิมพ์	2		
ราคา	190		
		คืน	Close

รูปที่ 4 หน้าจอการคืนหนังสือ

- จากรูปที่ 4 โปรแกรมจะรอรับเลขทะเบียนหนังสือ เมื่อทำการป้อนเสร็จเรียบร้อยแล้วโปรแกรมจะทำการค้นหา และเอารายละเอียดของสมาชิกที่คืนหนังสือเล่มนั้นออกมา
- หน้าจอของรูปที่ 4 จะมีการบันทึกวันที่ยืมหนังสือ วันที่จะต้องส่งคืนหนังสือ วันที่นักศึกษาคืนหนังสือ และค่าปรับที่ส่งหนังสือช้ากว่ากำหนดโดยปรับในอัตราวันละหนึ่งบาท
- ถ้ารหัสหนังสือที่คืนมายังไม่มีการยืมไปหรือถ้าป้อนรหัสหนังสือผิดโปรแกรมจะไม่แสดงรายละเอียดเกี่ยวกับหนังสือเล่มนั้นออกมา แล้วพร้อมที่จะรอรับรหัสหนังสือใหม่
- โปรแกรมงานบริการคืนหนังสือ จะมีความสัมพันธ์กับโปรแกรมงานบริการสืบค้นหนังสือ และส่วนอื่นๆ คือเมื่อสมาชิกมาคืนหนังสือ โปรแกรมจะทำการแก้ไขข้อมูลเก่าที่มีอยู่

งานบริการเพิ่ม/ลบ/แก้ไขหนังสือ

จาก menu รายการหนังสือในรูปที่ 2 เมื่อเราคลิกปุ่มเลือก " เพิ่ม/ลบ/แก้ไขหนังสือ " จะปรากฏจอร์ับข้อมูลใหม่ดังรูปที่ 5 ซึ่งจะแบ่งเป็นสามส่วนคืองานเพิ่มหนังสือ งานลบหนังสือ งานแก้ไข และ Exit ซึ่งมีรายละเอียดในการใช้งานดังนี้



รูปที่ 5 หน้าจอ menu รายการ เพิ่ม/ลบ/แก้ไขหนังสือ

การใช้งานบริการเพิ่มหนังสือ

- จากหน้าจอรูปที่ 5 ถ้าเราต้องการเพิ่มหนังสือก็ให้ **คลิกไปที่** "เพิ่มหนังสือ" จะปรากฏหน้าจอตั้งแสดงในรูปที่ 6

The screenshot shows a web form titled "Form12" with the following fields and buttons:

- เลขทะเบียนหนังสือ: 11
- เลขเรียกหนังสือ: Ia300
- ชื่อหัวเรื่อง_1: ฐานข้อมูล
- ชื่อหัวเรื่อง_2: Database
- ชื่อนี้หนังสือ: ระบบฐานข้อมูล
- ชื่อ นามสกุล ผู้แต่ง: ดวงแก้ว ขวามิภักดิ์
- สำนักพิมพ์: บริษัท ซีเอ็ดบุ๊คเซ็น จำกัด(มหาชน)
- พิมพ์ครั้งที่: 2
- ราคา: 110
- ปีที่พิมพ์: 2539
- ลักษณะที่ใดมา: ซีดี
- รหัสมาตรฐาน (ISBN): 974-509-135-4

Buttons: "ไม่บันทึก", "บันทึก", "เพิ่ม", "Close".

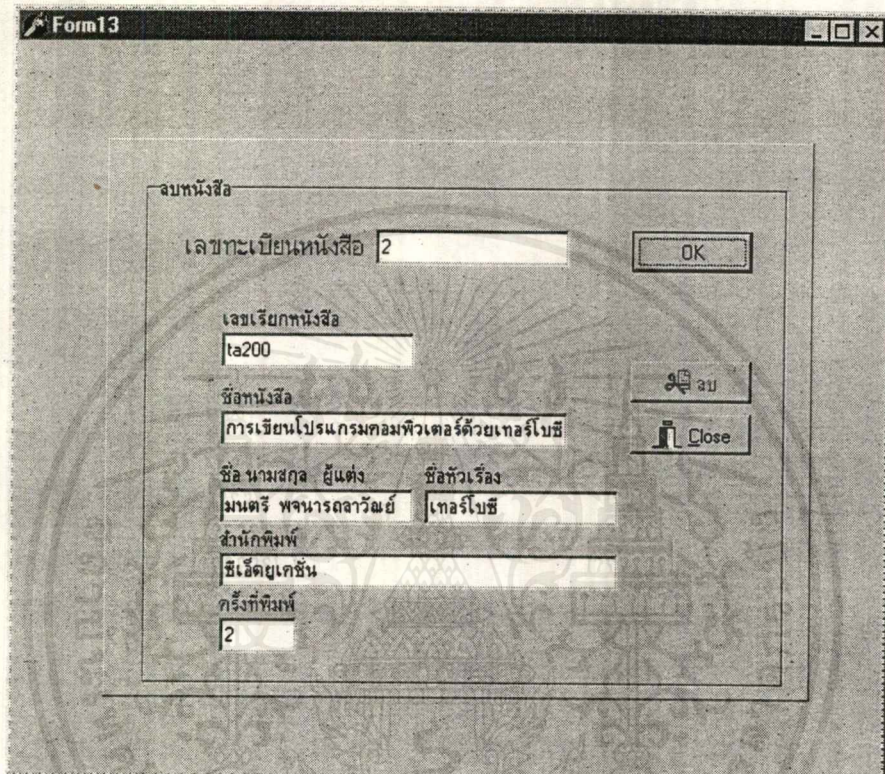
รูปที่ 6 หน้าจอแสดงรายละเอียดเกี่ยวกับการเพิ่มหนังสือ

- การเพิ่มหนังสือจะมีการเพิ่มเลขทะเบียนหนังสือโดยอัตโนมัติ
- ถ้ามีการเพิ่มหนังสือที่มีเลขเรียกหนังสือที่เหมือนกันรายละเอียดต่างๆที่เกี่ยวกับหนังสือเล่มนั้นจะปรากฏออกมาเองโดยอัตโนมัติ
- ถ้าชื่อหนังสือ หรือ ชื่อของผู้แต่งเป็นภาษาอังกฤษ ให้ใช้ตัวอักษรเป็นตัวพิมพ์เล็กเท่านั้น
- ขณะที่ป้อนข้อมูลอยู่ถ้าต้องการเปลี่ยนช่องที่จะเติมข้อมูลให้ใช้เมาส์คลิก หรือกดที่ " tab"
- ถ้ากรอกข้อมูลเกี่ยวกับหนังสือไม่ครบเช่น ไม่ได้กรอกเลขหมู่ หรือ ไม่ได้กรอกชื่อผู้แต่ง โปรแกรมจะไม่ทำการบันทึกให้โดยจะแสดงสีแดงที่คำว่า " ไม่บันทึก " แต่ถ้ากรอกข้อมูลครบถ้วนเสร็จแล้วคลิกที่ปุ่ม"เพิ่ม"โปรแกรมจะทำการบันทึกให้ โดยแสดงสีน้ำเงินที่คำว่า "บันทึก"
- ถ้าต้องการกลับไปสู่นำจอ menu ให้คลิกไปที่ปุ่ม " Close "

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานบริการลบบรายชื่อหนังสือ

จากหน้าจอรูปที่ 5 คลิกปุ่มเลือก "ลบหนังสือ" จะปรากฏหน้าจอเดียวกับรูปที่ 7



The screenshot shows a web form titled "ลบหนังสือ" (Delete Book) with the following fields and buttons:

- เลขทะเบียนหนังสือ: 2
- OK button
- เลขเรียกหนังสือ: la200
- ชื่อหนังสือ: การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี
- ลบ button
- ชื่อ นามสกุล ผู้แต่ง: มนตรี พจนารถาวณิชย์
- ชื่อตัวเรียง: เทอร์โบซี
- สำนักพิมพ์: ซีเอ็ดยุคเข็ญ
- ปีครั้งที่พิมพ์: 2
- Close button

รูปที่ 7 หน้าจอแสดงงานลบหนังสือ

- หน้าจอจะรอรับเลขทะเบียนหนังสือ ถ้าจะทำการลบให้กรอกเลขทะเบียนหนังสือเรียบร้อยแล้ว ก็คลิกไปที่ปุ่ม "OK" จากนั้นก็จะปรากฏรายละเอียดเกี่ยวกับหนังสือเล่มนั้นออกมาจากนั้นก็คลิกไปที่ปุ่ม "ลบ" ก็จะปรากฏหน้าจอแสดงการยืนยันการลบ เมื่อต้องการลบก็คลิกไปที่ปุ่ม "ลบ" การลบก็เสร็จเรียบร้อย แต่ถ้าไม่ต้องการลบก็คลิกไปที่ปุ่ม "Cancel" เพื่อกลับไปสู่งานบริการลบบรายชื่อหนังสืออีกครั้ง

การใช้งานบริการแก้ไขหนังสือ

เมื่อคลิกไปที่ปุ่ม "แก้ไข" จะปรากฏหน้าจอแสดงในรูปที่ 8 ซึ่งจะมีการทำงานดังนี้

Form15

โปรดพิมพ์อักษรด้วยตัวพิมพ์เล็ก

โปรดป้อนเลขทะเบียนเพื่อทำการแก้ไข

เลขทะเบียนหนังสือ 2 OK

เลขเรียกหนังสือ เลขเรียกหนังสือ

ใบ200 ทิวเรือ่ง

เทลโรบซี ชื่อหนังสือ

การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี ชื่อ นามสกุล ยี่แต้ง

มนตรี พจนารถวรณ์ ชื่อ นามสกุล ยี่แต้ง

สำนักพิมพ์ สำนักพิมพ์

ซีเอ็ดยูเคชั่น

ครั้งที่พิมพ์ ปีที่พิมพ์ รังษมาตรฐาน

2 2528 974-509-412-9

ระยะเวลาได้มา ราคา

3 ปี 190

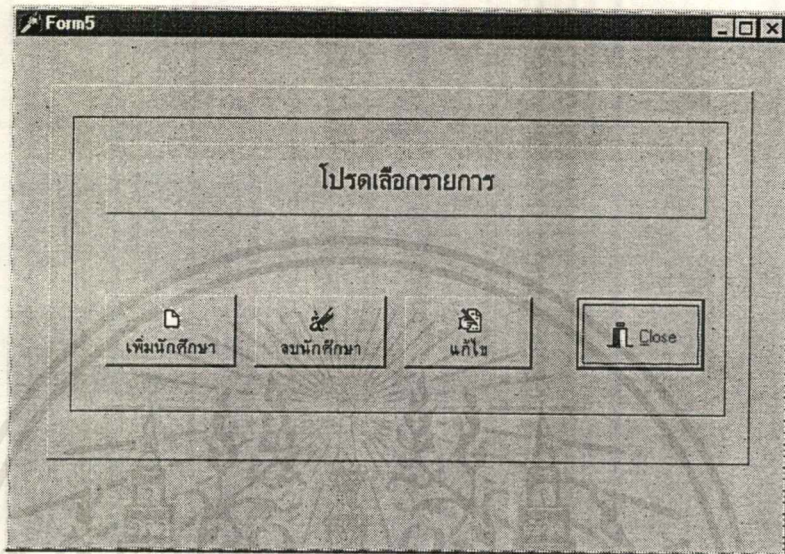
แก้ไข Cancel

รูปที่ 8 แสดงหน้าจอการแก้ไขหนังสือ

- หน้าจองานแก้ไขหนังสือจะถูกแบ่งเป็น 2 ส่วนคือทางซ้ายมือของหน้าจอจะแสดงข้อมูลของหนังสือก่อนการแก้ไข ส่วนทางขวาของจอจะแสดงส่วนที่ต้องการแก้ไข
- ใส่เลขทะเบียนหนังสือที่ต้องการแก้ไข เมื่อใส่แล้วจะปรากฏรายละเอียดต่างๆของหนังสือเล่มนั้นออกมา
- เมื่อต้องการแก้ไขตรงส่วนใดก็แก้ไขได้โดยการคลิกข้อความลงในหน้าจอที่จัดเตรียมไว้ทางขวามือสำหรับการแก้ไข จากนั้นก็คลิกไปที่ปุ่ม "แก้ไข" ก็จะปรากฏข้อมูลที่ถูกแก้ไขเรียบร้อยแล้วทางซ้ายมือ
- ถ้าต้องการออกจากหน้าจอการเพิ่มลบหนังสือนั้นให้คลิกไปที่ปุ่ม "Close"

งานบริการเพิ่ม/ลบ/แก้ไขนักศึกษา

จากหน้าจอ "เพิ่ม/ลบ/แก้ไขนักศึกษา" จะปรากฏหน้าต่างดังแสดงในรูปที่ 9



รูปที่ 9 menu งานบริการ เพิ่ม/ลบ/แก้ไขนักศึกษา

การใช้งานงานบริการเพิ่มนักศึกษา

รูปที่ 10 หน้าจองานบริการเพิ่มนักศึกษา

- จากหน้าจอรูปที่ 10 จะมีช่องให้ใส่ข้อมูลเกี่ยวกับนักศึกษาโดยจะมีช่องสำหรับใส่ รหัสนักศึกษา ชื่อ นักศึกษา นามสกุล ชั้นปี และ ห้อง เมื่อกรอกเสร็จเรียบร้อยแล้วคลิกไปที่ปุ่ม " OK " จะปรากฏสีน้ำเงินที่คำว่า " บันทึก " การเพิ่มนักศึกษาก็เสร็จเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้ผู้ใช้ไปใช้ประโยชน์อื่นใด
ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้ใช้ที่นำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจากผู้จัดทำเอกสารจะถือว่าผิดกฎหมายและจะถูกลงโทษตามกฎหมาย
โปรแกรมจะทำการป้องกันการป้อนรหัสนักศึกษาซ้ำซ้อน คือ ถ้ามีการใช้รหัสนักศึกษาที่ซ้ำซ้อนกัน โปรแกรมจะฟ้องโดยแสดงสีแดงที่คำว่า "ไม่บันทึก"

- ถ้ามีการกรอกข้อมูลไม่ครบโปรแกรมก็จะไม่ทำการบันทึกให้โดยจะแสดงสีแดงที่คำว่า "ไม่บันทึก"
- ถ้ามีการกรอกข้อมูลผิดหรือต้องการลบข้อมูลที่แสดงอยู่บนหน้าจอ ก็สามารถทำได้โดยการคลิกไปที่ปุ่ม " Cancel "
- ถ้าต้องการออกจากงานบริการนี้ก็สามารทำได้โดยการคลิกไปที่ปุ่ม " Close "

การใช้งานบริการลบนักศึกษา

จากรูปที่ 9 เมื่อคลิกไปที่ปุ่ม"ลบนักศึกษา" ก็จะปรากฏหน้าจอดังในรูปที่ 11

รูปที่ 11 แสดงหน้าจอการลบนักศึกษา

- การลบนักศึกษาเพียงใส่รหัสนักศึกษาแล้วคลิก "OK " โปรแกรมจะแสดงรายละเอียดเกี่ยวกับนักศึกษาคนนั้นออกมาแล้วจากนั้นก็คลิกไปที่ปุ่ม"ลบ"ก็จะปรากฏหน้าจอยืนยันการลบถ้าแนใจก็คลิกที่ปุ่ม "ลบ"
- การลบนักศึกษาทั้งห้องเพียงใส่ ห้องที่ต้องการจะลบ แล้วคลิก "OK " โปรแกรมจะแสดงรายละเอียดเกี่ยวกับนักศึกษาภายในห้องนั้น โดยจะแสดงจำนวนนักศึกษา ชื่อนักศึกษาและรายละเอียดต่างๆทั้งหมดของนักศึกษาภายในห้องนั้นออกมา ถ้าแนใจก็คลิกไปที่ลบก็จะปรากฏ หน้าจอยืนยันการลบถ้าแนใจก็คลิกที่ปุ่ม "ลบ"
- ถ้าต้องการออกจากหน้าจองานบริการลบศึกษาก็คลิกไปที่ปุ่ม "Close"

งานแก้ไขนักศึกษา

จากรูปที่ 9 เมื่อคลิกไปที่ปุ่ม "แก้ไข" จะปรากฏหน้าจอดังรูปที่ 12 จะเป็นการแก้ไขนักศึกษา โดยจะมีรายละเอียดการทำงานดังนี้

โปรแกรมรับรหัสนักศึกษาเพื่อทำการแก้ไข

รหัสนักศึกษา
39012025 OK

ชื่อ นามสกุล	ชั้นปี	ห้อง	ชื่อ นามสกุล	ชั้นปี	ห้อง
วรรณิ อีจัน	2	1			

ที่อยู่
kmitl

ที่อยู่
พระจอมเกล้าลาดกระบัง

แก้ไข Cancel Close

รูปที่ 12 หน้าจอการแก้ไขนักศึกษา

- หน้าจอนี้จะแบ่งออกเป็น 2 ส่วนคือ ซ้ายมือจะแสดงข้อมูลก่อนการแก้ไข ส่วนขวามือจะแสดงส่วนที่ต้องการจะแก้ไข
- ใส่รหัสนักศึกษาแล้วคลิกที่"OK"โปรแกรมจะแสดงรายละเอียดเกี่ยวกับนักศึกษาคนนั้นถ้าต้องการแก้ไขตรงส่วนใดก็แก้ลงในส่วนที่เตรียมไว้สำหรับการแก้ไข
- เมื่อกรอกข้อมูลที่ต้องการแก้ไขเสร็จเรียบร้อยแล้วก็คลิกไปที่ "แก้ไข" ก็จะปรากฏข้อมูลที่ถูกแก้ไขแล้วทางส่วนขวาของจอ

งานบริการตรวจทั้งหมด

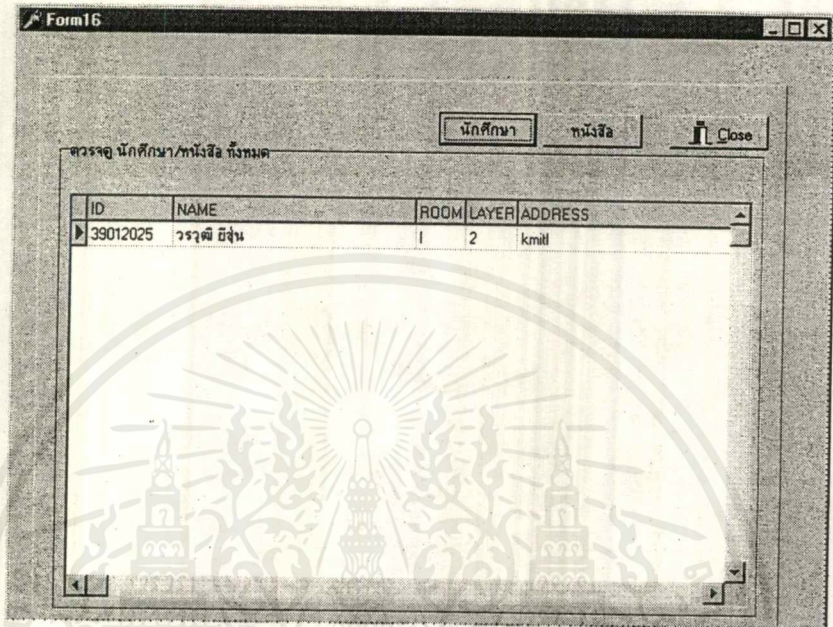
จากรูปที่ 2 เมื่อคลิกไปที่ "ตรวจทั้งหมด" จะปรากฏหน้าจอแสดงในรูปที่ 13 ซึ่งจะมีการทำงานดังนี้



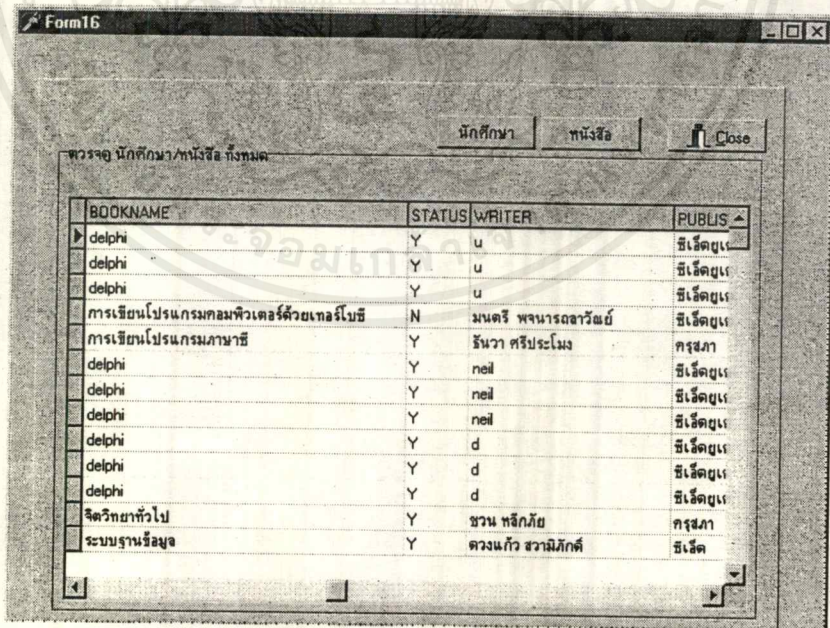
รูปที่13 แสดงหน้าจอการตรวจทั้งหมด

- จากหน้าจอที่ปรากฏจะมีให้เลือกว่าจะตรวจสอบ นักศึกษา หรือ ตรวจสอบ หนังสือ ถ้าต้องการตรวจสอบอะไรก็ให้คลิกไปที่ปุ่มนั้นก็จะปรากฏรายชื่อของข้อมูลทั้งหมดที่เก็บไว้ ถ้าคลิกเลือกที่จะตรวจสอบนักศึกษาก็จะปรากฏรายชื่อนักศึกษาที่เป็นสมาชิกทั้งหมดโดยจะแสดงให้เห็นถึง ID NAME SERNAME และ LAYER ดังแสดงในรูปที่ 14
- ถ้าคลิกที่จะเลือกตรวจสอบหนังสือจะแสดงหนังสือทั้งหมดที่มีอยู่ซึ่งแสดงให้เห็น ID BOOK ใช้แสดงเลขทะเบียนหนังสือ CALL ใช้แสดงเลขเรียกหนังสือ BOOKNAME ใช้แสดงชื่อหนังสือ STATUS ใช้แสดงสถานะว่าสามารถยืมได้หรือไม่ได้ WRITER นอกจากนี้ยังมีรายละเอียดอื่นๆอีก คือ PUBLISH , EDITION ,ISBN ,PRICE ดังในรูปที่ 15 ถ้าต้องการออกจกงานบริการก็ให้คลิกไปที่ปุ่ม "Close" หน้าจองานบริการก็จะถูกปิด

- ถ้าต้องการออกจากโปรแกรมให้คลิกไปที่ปุ่ม " Close " หน้าจองานบริการนี้ก็จะถูกปิดไป



รูปที่ 14 หน้าจอแสดงการตรวจสอบนักศึกษา

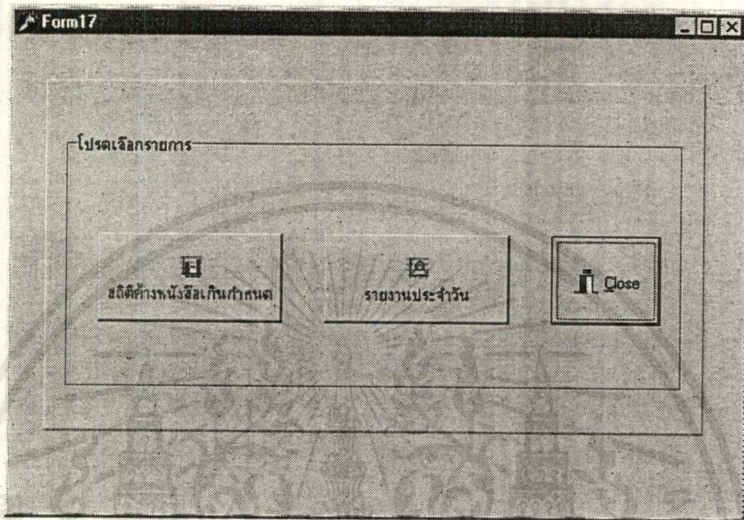


รูปที่ 15 หน้าจอแสดงการตรวจสอบหนังสือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานบริการรายงาน

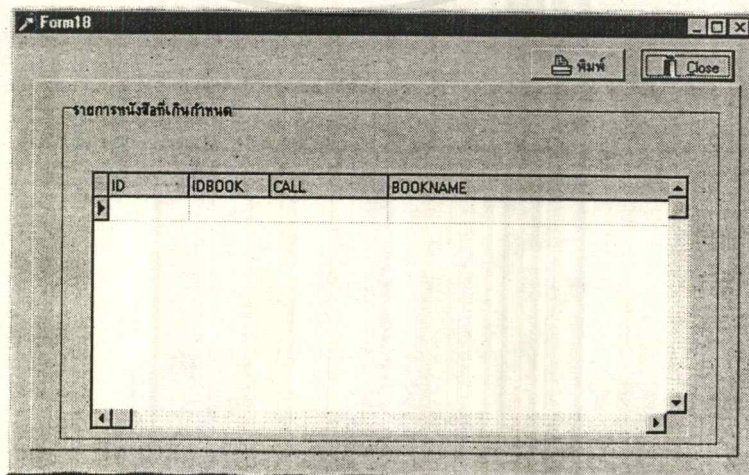
จาก menu ในรูปที่ 2 เมื่อเราคลิกปุ่มเลือก"รายงาน" จะเข้าสู่งานบริการนี้ ซึ่งจะมีหน้าจอแสดง
ในรูปที่ 16 ซึ่งจะมีการทำงานดังนี้



รูปที่ 16 งานบริการรายงาน

การใช้งาน รายงานสถิติการค้างหนังสือเกินกำหนด

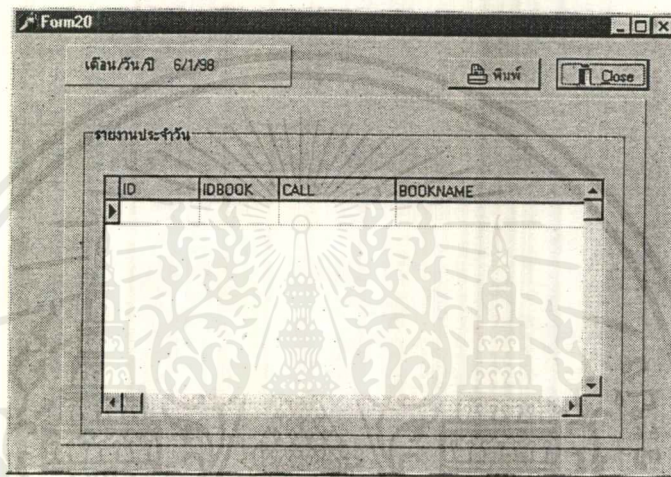
- รูปที่ 16 เมื่อคลิกไปที่ปุ่ม"สถิติการค้างหนังสือเกินกำหนด" ก็จะปรากฏหน้าจอตั้งในรูปที่ 17 ซึ่งจะแสดงชื่อ และรายละเอียดต่างๆของหนังสือที่ส่งเข้าเกินกว่ากำหนด
- ถ้าต้องการที่จะพิมพ์รายละเอียดต่างๆเหล่านี้สามารถที่จะพิมพ์ได้โดยคลิกไปที่ "พิมพ์"
- ถ้าต้องการออกจากหน้าจอนี้ก็คลิกไปที่ "Close "



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 17 แสดงหน้าจอรายงานหนังสือที่ส่งเข้าเกินกว่ากำหนด
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้รายงานประจำวัน

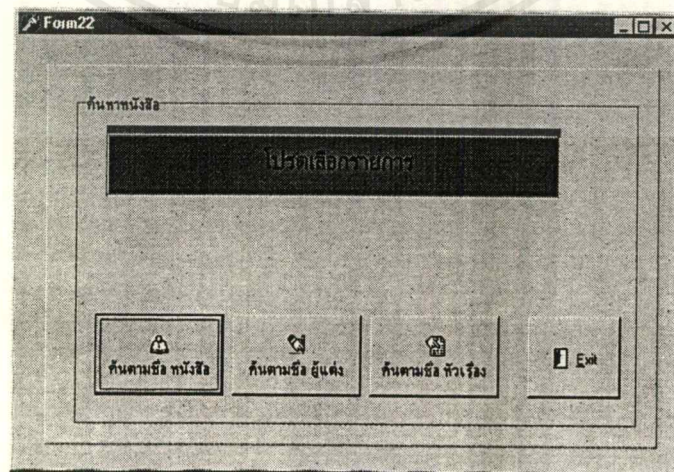
- จากรูปที่ 16 เมื่อกดคลิกไปที่ "รายงานประจำวัน" ก็จะปรากฏหน้าจอแสดงในรูป18 ซึ่งจะแสดงรายละเอียดของรายงานประจำวันของวันนั้นๆว่ามี การ ยืม - คืน หนังสืออะไรบ้าง
- ถ้าต้องการพิมพ์ข้อมูลมาใช้ก็สามารถพิมพ์ได้โดยคลิกไปที่ " พิมพ์ "
- ถ้าต้องการออกจากหน้าต่างนี้คลิกไปที่ " Close "



รูปที่ 18 แสดงหน้าจอรายงานประจำวัน

งานบริการค้นหาหนังสือ

จากหน้าจอ Main menu ดังในรูปที่ 1 คลิกเลือกปุ่ม"ค้นหาหนังสือ"จะปรากฏหน้าจอ menu รายการต่างๆให้เลือกดังในรูปที่ 19

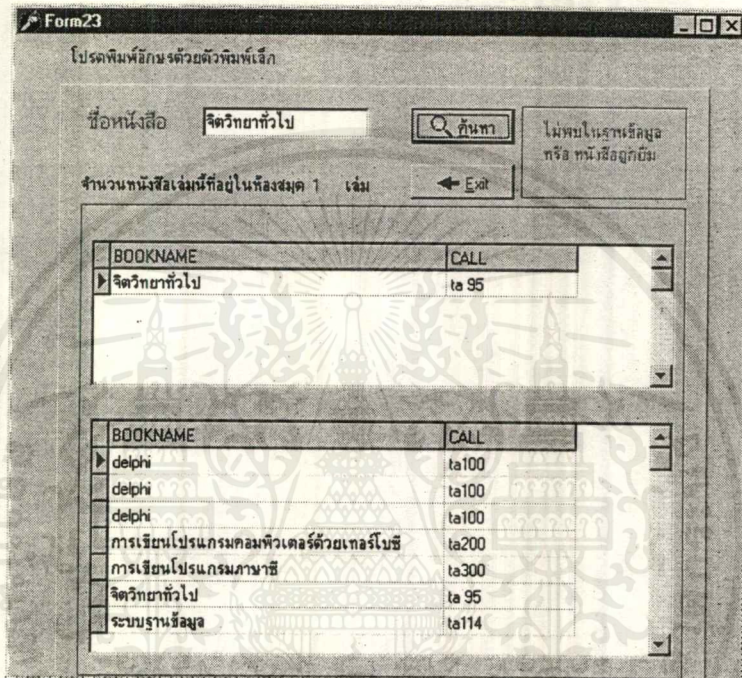


รูปที่ 19 แสดง menu ต่างๆที่ใช้เลือกค้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการใช้งานบริการเลือกค้นตามชื่อหนังสือ

จากรูปที่ 16 เมื่อเราคลิกเลือกปุ่มเลือก " การค้นหาตามชื่อหนังสือ" จะปรากฏหน้าจอแสดงในรูปที่ 20 ซึ่งจะมีวิธีการทำงานดังต่อไปนี้



รูปที่ 20 แสดงหน้าจอการค้นหาตามชื่อหนังสือ

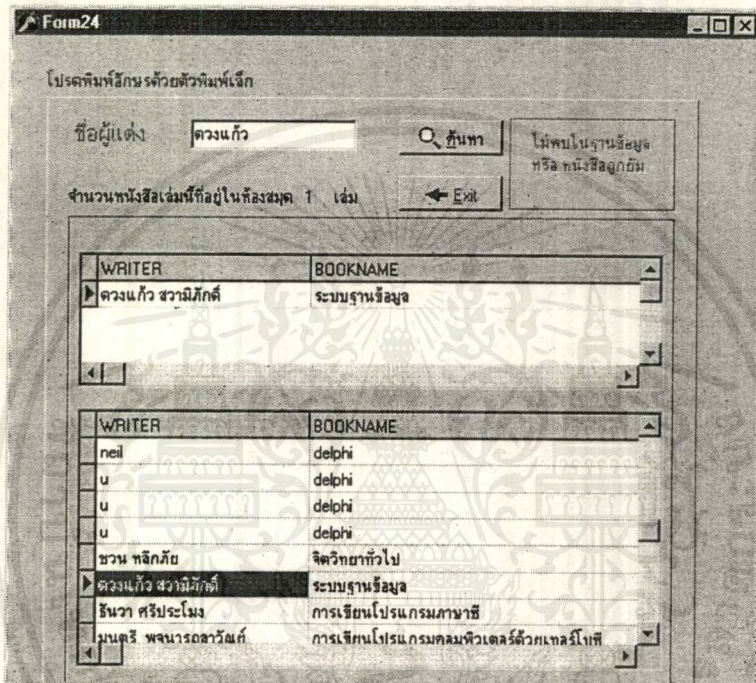
- จะปรากฏหน้าจอเพื่อรอรับการป้อน ชื่อหนังสือที่เราต้องการค้นหา เมื่อป้อนข้อมูลเข้าไปแล้ว โปรแกรมจะทำการค้นหาและแสดงการค้นหาและจะแสดงรายละเอียดเกี่ยวกับหนังสือเล่มนั้นออกมา เช่น ชื่อหนังสือ จำนวนของหนังสือที่มีอยู่ในห้องสมุด
- ถ้าหากรายชื่อหนังสือที่ต้องการค้นหาไม่มีในห้องสมุดจะเกิดสีแดงที่คำว่า " ไม่พบในฐานข้อมูลหรือถูกยืม " หากต้องการออกจากหน้าจอนี้ให้กดไปที่ปุ่ม "Exit " หน้าจอนี้ก็จะถูกปิดไป

วิธีการเลือกค้นหนังสือตามชื่อผู้แต่ง

- จากหน้าจอ menu ในรูปที่ 16 คลิกปุ่ม "ค้นตามชื่อผู้แต่ง" จะปรากฏหน้าจอเพื่อรอรับการป้อน ชื่อ - นามสกุลผู้แต่งที่เราต้องการค้นหา เมื่อป้อนข้อมูลเข้าไปแล้วโปรแกรมจะทำการค้นหาและแสดงรายละเอียดเกี่ยวกับหนังสือเล่มนั้นออกมา พร้อมทั้งแสดงจำนวนหนังสือที่ยังคงเหลืออยู่ในห้องสมุด (ที่ยังไม่ถูกยืม) ว่ายังเหลืออีกกี่เล่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าห้องสมุดไม่มีหนังสือที่ผู้แต่งคนนั้นแต่งหรือเกิดการผิดพลาดในการป้อนข้อมูลโปรแกรมจะทำการแสดงให้ทราบโดยจะมีสีแดงปรากฏที่คำว่า " ไม่พบในฐานข้อมูลหรือหนังสือถูกยืม"
- ถ้าหากต้องการออกจากหน้าจอนี้ให้คลิกไปที่ปุ่ม " Exit " หน้าจอนี้ก็จะถูกปิดลง



รูปที่ 21 แสดงหน้าจอการค้นตามชื่อผู้แต่ง



การค้นหาตามชื่อหัวเรื่อง

เมื่อคลิกไปที่ "ค้นหาตามชื่อหัวเรื่อง" จะปรากฏหน้าจอแสดงดังแสดงในรูปที่ 19 ซึ่งจะมีการทำงานดังนี้คือ

BOOKNAME	TITEL	CAL
ระบบฐานข้อมูล	ฐานข้อมูล	ta11

BOOKNAME	TITEL	CAL
delphi	del	ta10
delphi	del	ta10
การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี	เทอร์โบซี	ta20
การเขียนโปรแกรมภาษาซี	ภาษาซี	ta30
จิตวิทยาทั่วไป	จิตวิทยา	ta 9c
ระบบฐานข้อมูล	ฐานข้อมูล	ta11

รูปที่ 22 แสดงหน้าจอการค้นหาตามชื่อหัวเรื่อง

- จะปรากฏหน้าจอเพื่อรอรับการป้อน ชื่อหัวเรื่อง ที่เราต้องการค้นหา เมื่อป้อนข้อมูลเข้าไปแล้ว โปรแกรมจะทำการค้นหาและแสดงรายละเอียดเกี่ยวกับหนังสือเล่มนั้นออกมา พร้อมทั้งแสดงจำนวนหนังสือที่ยังคงเหลืออยู่ในห้องสมุด (ที่ยังไม่ถูกยืม) ว่ายังมีเหลืออีกกี่เล่ม
- ถ้าห้องสมุดไม่มีหนังสือ หรือเกิดการผิดพลาดในการป้อนข้อมูลโปรแกรมจะทำการแสดงให้ทราบโดยจะมีสีแดงปรากฏที่คำว่า "ไม่พบในฐานข้อมูลหรือหนังสือถูกยืม"
- ถ้าหากต้องการออกจากหน้าจอนี้ให้คลิกไปที่ปุ่ม " Exit " หน้าจอนี้ก็จะถูกปิดลง