



มัลติมีเดียสื่อการสอนโปรแกรมภาษา DELPHI
MULTIMEDIA COMPUTER AID INSTRUCTOR FOR DELPHI



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของของการศึกษาหลักสูตรปริญญาวิทยาศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มัลติมีเดียสื่อการสอนโปรแกรมภาษา DELPHI
MULTIMEDIA COMPUTER AID INSTRUCTOR FOR DELPHI

โดย

นาย จตุรนต์	พวงนาค	37013284	3P
นาย อนุชา	บุญชูโต	37013322	3P
นาย อิทธิศักดิ์	จำรัสประเสริฐ	37013324	3P

อาจารย์ที่ปรึกษา

ดร. วรวัฒน์ ลิ้มโกศา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2539

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โครงการนวัตกิตติมศักดิ์การสอนโปรแกรมภาษา DELPHI

ผู้จัดทำ	นาย จตุรนต์	พวงนาค	37013284	3P
	นาย อนุชา	บุญชูโต	37013322	3P
	นาย อธิศักดิ์	จรัสประเสริฐ	37013324	3P



ดร. วรวัฒน์ ถิมโกศา

(.....)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	
Abstract	
บทนำ	
บทที่ 1 มัคติมี่เดีย	1
บทที่ 2 ทฤษฎีและเอกสารที่เกี่ยวข้อง	14
บทที่ 3 ความเป็นมาของ เดลไฟ	18
บทที่ 4 การใช้งาน เดลไฟ	56
บทที่ 5 Multimedia ToolBook3	79
บทที่ 6 Multimedia ToolBook CBT	87
บทที่ 7 Screencam96	94
บทที่ 8 การติดตั้งโปรแกรม	97
บทที่ 9 การทำงานของโปรแกรม MMCAI	108
บทที่ 10 โค้ดของโปรแกรม	112



สารบัญรูป

	หน้า
รูป 1 ภาพใบหน้าที่ของ Chart FX ของ page Control	21
รูป 2 หน้าต่างของโปรแกรม Explorer	23
รูป 3 แสดงการใช้ Tree view	24
รูป 4 การแสดงผลในกรอบ Listview	32
รูป 5 ไดอะล็อกบ็อกซ์ NewItem	35
รูป 6 ไดอะล็อกบ็อกซ์	39
รูป 7 Demo Inherit Form	41
รูป 8 หน้าต่าง Form	44
รูป 9 หน้าต่าง Unit	45
รูป 10 การกำหนดคอมโปเนนต์	47
รูป 11 Form Ddsigner	56
รูป 12 The Editor Windows	57
รูป 13 Component Palette	58
รูป 14 Object inspector	58
รูป 15 Online Help	59
รูป 16 องค์ประกอบของ ODBC	61
รูป 17 ลำดับขั้นฐานข้อมูล	62
รูป 18 สถาปัตยกรรมฐานข้อมูล	64
รูป 19 การติดต่อกับฐานข้อมูล	66
รูป 20 ลำดับขั้นของ Object Hierarchy	80
รูป 21 จอภาพหลักของ MTB3	80
รูป 22 เครื่องมือของ MTB3	81
รูป 23 แถบบอกสถานะของ MTB3	86
รูป 24 อินเตอร์เฟซของ Screencam96	94
รูป 25 อินเตอร์เฟซของ Screencam Player	95
รูป 26 Logo ของตัวโปรแกรม	97
รูป 27 เลือกรูปแบบการติดตั้ง	98

รูป 28 เลือก Directory	99
รูป 29 การเลือกแบบ Custom	100
รูป 30 การติดตั้งโปรแกรม	101
รูป 31 Manager Group	102
รูป 32 หน้าต่าง Manager group	103
รูป 33 หน้าจอการติดตั้งสมบุรณ์	104
รูป 34 การเข้าโปรแกรม	105
รูป 35 สารบัญ	108
รูป 36 รายการ	109
รูป 37 หน้าต่างควบคุม	110
รูป 38 โปรแกรมตัวอย่าง	111



โปรแกรมสื่อการสอนภาษาเคลไฟ

ผู้จัดทำ

1. นาย จตุรงค์ พวงนาค
2. นาย อนุชา บุญชูโต
3. นาย อธิรศิกดิ์ จำรัสประเสริฐ

อาจารย์ที่ปรึกษา

ดร. วรวัฒน์ ลิ้มโกศา

ปีการศึกษา 2539

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้จะกว้างถึง การนำเสนอข้อมูลในรูปแบบของ ระบบมัลติมีเดีย โดยสื่อออกมาเป็นโครงการโปรแกรม ภาษาเคลไฟ ซึ่งตัวโปรแกรมนี้ จะนำเสนอข้อมูลที่เป็นภาพเคลื่อนไหว และ เสียงเพื่อให้ผู้เรียนมองเห็นภาพได้อย่างชัดเจน โดยตัวโปรแกรมแบ่งเป็น 3 ส่วนหลักๆ คือส่วนของโปรแกรม Multimedia ToolBook , Delphi และ Screencam โดย โปรแกรม Screencam จะบันทึกหน้าจอที่ทำการสอนและจัดลำดับขั้นในการนำเสนอ โดย โปรแกรม Multimedia ToolBook และในส่วนของ Delphi จะเป็นส่วนของคำถามท้ายบทที่จะเก็บคะแนนสูงสุดของผู้เขียน โดยการสอนทั้งหมด 10 บท

MULTIMEDIA COMPUTER AID INSTRUCTOR FOR DELPHI

Mr. Jaturon Puangnak 37013284

Mr. Anucha Boonchuto 37013322

Mr. Ittisak Jamratparsert 37013324

Advisor

DR. Voravat Limpoka

Academic Year 1996

abstract

this thesis describe about taking computer program to help teaching in multimedia system. It's present a media for teching program Delphi language presents animation and effect sounds. It's make you clean in images. This program has 3 parts , one is the part of Multimedia ToolBook , Delphi and ScreenCam. ScreenCam program for record screen computer for teaching ,two manage processing for presentation by Multimedia ToolBook , three Delphi is a part of finally question. The total contents has 10 chapter.

บทที่ 1

มัลติมีเดีย

ปัจจัยสำคัญ ในการนำเสนอข้อมูลในยุคแห่งสารสนเทศ

การนำเสนอข้อมูล มีการแข่งขันกันอย่างมาก เพื่อตอบสนองความต้องการของผู้บริโภค หากรูปแบบในการนำเสนอข้อมูล สามารถเข้าใจได้ง่าย หรือ สามารถโต้ตอบ สื่อสารกับผู้รับได้ สื่อนั้นจะน่าสนใจมากทีเดียว

การนำเสนอข้อมูลในรูปแบบมัลติมีเดีย ในปัจจุบันกำลังเป็นที่นิยมกันอย่างมาก ไม่ว่าจะเป็นด้าน การเรียนการสอน การโฆษณาสินค้า ความบันเทิง โดยตัวกลางของสื่อนี้มักจะอยู่ในรูปของ CD-ROM หรือไฟล์ข้อมูลบนเครื่องคอมพิวเตอร์ หรืออาจจะหาสื่อรูปแบบนี้ได้ ทางอินเทอร์เน็ต

ในบทนี้ จะกล่าวถึงเรื่องเกี่ยวกับมัลติมีเดีย ไม่ว่าจะเป็นความเข้าใจเกี่ยวกับข้อมูลต่างๆ อุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการสร้างงานมัลติมีเดีย

ระบบมัลติมีเดีย คืออะไร

การให้คำจำกัดความของคำว่ามัลติมีเดีย ในขณะนี้ยังไม่สิ้นสุด ทั้งนี้เนื่องมาจากการพัฒนาเทคโนโลยี ต่างๆ ยังไม่สิ้นสุด จึงขออนุญาตอย่างกว้าง ๆ ว่าเป็น การรวบรวมสื่อต่างๆ ที่มีองค์ประกอบสำคัญใหญ่ๆ 3 อย่าง คือ

1. ภาพ
2. เสียง
3. การโต้ตอบซึ่งกันและกัน

ระบบมัลติมีเดีย ใช้กับงานอะไรได้บ้าง

ขอบเขตของระบบมัลติมีเดีย ได้รับการพัฒนากว้างขวางมากขึ้นทำให้ข้อจำกัดในการทำงานมีน้อยลง เดิมคิดกันแต่เพียงว่า ระบบมัลติมีเดียจะเหมาะสำหรับงานเสนอเท่านั้น แต่ความจริงเราสามารถนำมัลติมีเดียมาประยุกต์ใช้งานได้มากมาย ได้แก่ ประยุกต์ใช้ในการศึกษา ด้านธุรกิจ ด้านการโฆษณา ด้านความบันเทิง ด้านการให้ข้อมูลตามสถานที่ต่าง ๆ เป็นต้น

ระบบมัลติมีเดีย มีอะไรบ้าง

สิ่งที่จำเป็นสำหรับระบบมัลติมีเดีย แยกเป็น 2 ส่วนคือ

1. ฮาร์ดแวร์ องค์ประกอบที่สำคัญของระบบมัลติมีเดียคือ ภาพ เสียง และการตอบโต้กันได้ ส่วนที่เป็นฮาร์ดแวร์จึงต้องจำแนกออกไปตามองค์ประกอบดังกล่าวโดยมีคอมพิวเตอร์เป็นส่วนสำคัญ คอมพิวเตอร์ที่ใช้งานในระบบมัลติมีเดียแยกจากกันระหว่างการสร้างงานจำเป็นต้องใช้คอมพิวเตอร์ที่มีสมรรถนะสูง ถ้าเป็นระดับเวิร์กสเตชัน จะดีมาก ซึ่งจะเป็นตัวเชื่อมโยงไปหาฮาร์ดแวร์อื่น ๆ

2. ซอฟต์แวร์มัลติมีเดีย แยกออกเป็น 2 ตอน คือ ตอนสร้างกับตอนใช้งาน

2.1 ซอฟต์แวร์สร้างงานระบบมัลติมีเดีย แยกออกเป็นสองส่วนคือ ซอฟต์แวร์สำหรับสร้างภาพและเสียง กับซอฟต์แวร์จัดระบบ

ซอฟต์แวร์สำหรับสร้างภาพและเสียง มักมีการใช้และเป็นที่คุ้นเคยของผู้ใช้คอมพิวเตอร์อยู่แล้วมีการตกแต่ง ตัดต่อ ดัดแปลง สร้างใหม่ ซึ่งแยกกันทำเป็นชิ้นเป็นตอนได้

ซอฟต์แวร์จัดระบบมัลติมีเดีย เป็นซอฟต์แวร์ที่ควบคุมงานส่วนต่าง ๆ มาจัดลำดับ เพื่อให้มีการตอบโต้กันได้ เช่น บอกให้รู้ว่าถึงตอนนี้จะมีภาพต่อไป จะมีเสียง หรือถ้ากดตรงนี้จะได้อาพณ์ภาพนี้ออกมา หรือมีเสียงนั้น เสียงนี้ เป็นต้น ซอฟต์แวร์ประเภทนี้ยังไม่เป็นที่คุ้นเคยมากนัก ทำให้ยังมีน้อยราคาจึงสูงและที่สำคัญยังต้องใช้กับคอมพิวเตอร์สมรรถนะสูงด้วย

2.2 ซอฟต์แวร์ใช้งานระบบมัลติมีเดีย ระบบมัลติมีเดียนั้นถูกสร้างขึ้นมาโดยผ่านระบบคอมพิวเตอร์ ทำให้ไม่เป็นการยากนักหากจะใช้งาน แต่มีข้อจำกัดอยู่ที่ความง่ายในการใช้งาน ถ้าหากใช้คอมพิวเตอร์ไม่เป็นจะทำอย่างไร หรือมาตรฐานของระบบคอมพิวเตอร์จากหลายค่าย ซึ่งจะถูกนำไปสู่การพัฒนาระบบอื่นๆ ที่ไม่ต้องใช้คอมพิวเตอร์ เพื่อรองรับกับตลาดผู้บริโภคมากที่สุดเท่าที่จะทำได้ โดยในปัจจุบันนี้มีการพัฒนาเครื่องรับโทรทัศน์ ให้สามารถใช้งานอินเทอร์เน็ตได้ นั่นหมายความว่าระบบมัลติมีเดียนั้นกำลังได้รับการพัฒนา ให้ง่ายในการใช้งานต่อผู้บริโภคมากยิ่งขึ้น

ระบบมัลติมีเดีย กับการใช้งาน Web ในระบบเครือข่ายอินเทอร์เน็ต

ระบบเครือข่าย Web เป็นการนำเสนอข้อมูลแบบไม่จำกัดบริเวณไม่ว่าจะอยู่ที่ใดในโลกแห่งอินเทอร์เน็ต เราจะสามารถค้นหาข้อมูลต่างๆ ที่มีอยู่ได้ การใช้งานทั้งสองระบบร่วมกัน ในปัจจุบันสามารถทำได้ แต่ยังมีข้อจำกัดอยู่ที่เรื่อง ความเร็วในการรับส่งข้อมูล หากในอนาคตสามารถแก้ปัญหานี้ได้ จะทำให้การพัฒนาเทคโนโลยีการนำเสนอข้อมูลเป็นไปอย่างไม่มีที่สิ้นสุด

เทคโนโลยีที่เกี่ยวข้องกับมัลติมีเดีย

เนื่องจากระบบคอมพิวเตอร์มัลติมีเดียเป็นการรวบรวมเทคโนโลยีหลายอย่างเข้าด้วยกัน เพื่อให้เกิดความสมบูรณ์ในการทำงานเทคโนโลยีเหล่านั้น ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การพัฒนาเทคโนโลยีในการบันทึกข้อมูลการทำงานของมัลติมีเดียประกอบไปด้วยภาพและเสียง

2. การพัฒนาด้านระบบคอมพิวเตอร์เครือข่าย สิ่งที่ระบบคอมพิวเตอร์มัลติมีเดียเข้าไปมีบทบาทร่วมกับระบบคอมพิวเตอร์เครือข่าย เช่น การติดต่อสื่อสารด้วยระบบ Electronics Mail ซึ่งเดิมเป็นการติดต่อที่เป็นลักษณะ Text Base เท่านั้น เป็นการนำสองเทคโนโลยีมารวมกัน ทำให้การติดต่อสื่อสารในระบบเครือข่ายคอมพิวเตอร์ทำได้ทั้งที่เป็นภาพและเสียง

3. การพัฒนาเทคนิคการย่อหรือบีบอัดขนาดข้อมูล การย่อข้อมูลที่มีประสิทธิภาพจะเป็นปัจจัยสำคัญอย่างหนึ่ง ในการทำงานของระบบคอมพิวเตอร์มัลติมีเดีย

4. การพัฒนาไมโครคอมพิวเตอร์ การทำงานของคอมพิวเตอร์มัลติมีเดียเป็นการทำงานที่เกี่ยวข้องกับข้อมูลในปริมาณมหาศาล กระบวนการย่อและขยายขนาดข้อมูลจะต้องเกิดอย่างรวดเร็วและมากพอที่จะทำให้การติดต่อส่งข้อมูลระหว่างหน่วยความจำและอุปกรณ์ต่าง ๆ ไม่เกิดการหยุดชะงัก เพราะถ้าเกิดเหตุการณ์เช่นนี้จะทำให้การแสดงผลทั้งภาพและเสียงอาจเพี้ยนไปจากของจริงได้

5. การพัฒนาของจอภาพ

6. การพัฒนาอุปกรณ์ป้อนข้อมูล

7. การพัฒนาซอฟต์แวร์ ส่วนหนึ่งที่ทำให้โลกของคอมพิวเตอร์มัลติมีเดียเป็นจริงก็คือ การพัฒนาซอฟต์แวร์ ให้มีประสิทธิภาพสูงและมีการใช้งานได้ง่ายขึ้น การพิจารณาเลือกซอฟต์แวร์เพื่อมาทำงานด้านมัลติมีเดีย อาจพิจารณาได้จาก

- ความง่ายในการใช้งาน
- ความสามารถในการนำเสนองาน
- ความสามารถในการติดต่อกับผู้ใช้
- ความสามารถในการใช้ตัวแปรและฟังก์ชันในการคำนวณ
- ความสามารถในการใช้งานร่วมกับโปรแกรมอื่น ๆ
- มี Library ของโปรแกรมย่อยต่างๆ สนับสนุนการทำงาน
- ความสามารถในการทำเอกสารประกอบโปรแกรม
- ความสามารถในการส่งแอปพลิเคชัน(Application) ที่เสร็จแล้วให้ผู้ใช้

เรื่องของภาพนิ่ง (Still Image)

ภาพในระบบมัลติมีเดีย แบ่งออกเป็นประเภทใหญ่ ๆ ได้ดังนี้ คือ

1. ตัวอักษร (Text)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. รูปภาพ (Images)

โดยสามารถแบ่งตามลักษณะของภาพเป็นประเภทใหญ่ได้อีกคือ ภาพนิ่ง(Still Image) และ ภาพเคลื่อนไหว (Animation Image) ถ้าหากพิจารณาถึงที่มาของภาพแล้ว สามารถจำแนกออกเป็น สองลักษณะใหญ่ ๆ คือ

1. ภาพจากการจำลองแบบ (Imitated Image) ได้แก่ ภาพที่ถ่ายแบบมาจากของจริง โดยอาศัยเครื่องมือช่วยในการจำลองแบบ เช่น กล้องถ่ายภาพนิ่ง กล้องวิดีโอ เครื่องอ่านภาพสแกนเนอร์ (Scanner) เป็นต้น

2. ภาพจากการสร้าง (Creative Images) เป็นภาพที่สร้างขึ้นจากจินตนาการมนุษย์ เช่น ภาพวาด ภาพการ์ตูน เป็นต้น

เหตุที่เราไม่แยกตัวอักษรกับภาพออกจากกันก็เพราะถ้ามองในส่วนของการใช้งานคอมพิวเตอร์ในปัจจุบัน ตัวอักษรจัดเป็นภาพจากการสร้าง โดยที่การสร้างแบบตัวอักษร (Font) จะมีลักษณะการสร้างในเชิงรูปภาพ

ในการนำเสนอข้อมูลแบบมัลติมีเดีย เราอาจต้องใช้ข้อมูลชนิดภาพนิ่งจำนวนมาก เพื่ออธิบายลักษณะของโครงการ และทำให้การนำเสนอมีความน่าสนใจมากขึ้น

รูปแบบของไฟล์รูปภาพในปัจจุบันมีอยู่มากมายตามลักษณะการใช้งาน และในบางครั้งได้มีการกำหนดรูปแบบของไฟล์ขึ้นมาใหม่ เราสามารถแบ่งลักษณะของไฟล์รูปภาพเป็น 2 ประเภท

1. Bitmap File

เป็นไฟล์ที่เก็บรายละเอียดของภาพในลักษณะของจุดภาพหรือ Pixel มีการกำหนดไว้อย่างแน่นอนว่าที่ตำแหน่งใดของภาพจะมีลักษณะของสีและรายละเอียดอย่างไร เมื่อเราย่อหรือขยายภาพของไฟล์ชนิด Bitmap นี้เราจะสูญเสียความคมชัดของภาพไป

.ในปัจจุบันได้นำเทคโนโลยีการบีบข้อมูลมาใช้กันอย่างแพร่หลาย จึงได้มีการนำการบีบข้อมูลมาใช้ในการเก็บภาพนิ่งด้วย อาทิ ไฟล์ RLE(Run-Length Encoding), ไฟล์ JPEG(Joint Photographic Experts Group), ไฟล์ GIF(Graphics Image Format) ฯลฯ ในแต่ละวิธีจะมีอัลกอริทึม (Algorithm) ในการบีบข้อมูลแตกต่างกันออกไป

2. Vector File

เป็นไฟล์ที่สามารถย่อและขยายภาพได้โดยมีการสูญเสียรายละเอียดน้อยมาก แต่เรามักใช้กับภาพที่ไม่มีรายละเอียดมาก ลักษณะของภาพที่มีการย่อ-ขยายขนาด เช่น โลโก้(Logo) และภาพในงานเขียนแบบต่างๆ เป็นต้น

การพิจารณาเลือกใช้รูปแบบของไฟล์สำหรับงานมัลติมีเดีย

1. BMP

BMP เป็นรูปแบบไฟล์มาตรฐานของ Windows โปรแกรมส่วนใหญ่ของ Windows สามารถทำงานกับ ไฟล์ภาพนิ่งในรูปแบบนี้ได้ ซึ่งสามารถแสดงจำนวนสีได้มากถึง 16.7 ล้านสี (True Color 24 bits) ข้อดีของ ไฟล์รูปแบบนี้คือ ไม่มีการใช้เทคโนโลยีการบีบข้อมูลเลย ทำให้ไฟล์มีขนาดค่อนข้างใหญ่มาก เมื่อเปรียบเทียบกับ ไฟล์ในรูปแบบอื่น

2. GIF (Graphics Image Format)

GIF เป็นไฟล์รูปแบบของ CompuServe ซึ่งเป็นรูปแบบที่ใช้ในการสื่อสารข้อมูล (Down Load, Up Load) ไฟล์รูปแบบนี้จะสามารถเรียกใช้ได้โดยโปรแกรมตกแต่งภาพ (Image Retouching) หลายโปรแกรม

สามารถแสดงจำนวนสีได้ 256 สี (8 bits) มีการใช้เทคโนโลยีการบีบข้อมูลเข้ามาช่วย เพื่อให้ขนาดของ File ลดลงเพื่อประโยชน์ ในการส่งผ่านข้อมูล คุณภาพของภาพหลังจากบีบข้อมูลแล้ว ไม่มีการสูญเสียรายละเอียดของภาพแต่อย่างใด

3. JPEG (Joint Photographic Experts Group)

เป็นรูปแบบของภาพนิ่งที่กำลังเป็นที่นิยมมากในการเก็บข้อมูลขณะนี้ เพราะสามารถบีบข้อมูลได้มากมีจำนวนสี 16.7 ล้านสี สามารถกำหนด Factor ของการบีบขนาดภาพได้

ข้อเสียคือมีการสูญเสียรายละเอียดของภาพบางส่วนตามอัตราการ บีบข้อมูล

ในการทำโครงการมัลติมีเดียเราจะควรใช้รูปแบบไฟล์ JPEG เพื่อประหยัดเนื้อที่ในการเก็บภาพ แต่ในบางครั้งเราจำเป็นต้องใช้รูปแบบไฟล์ BMP หรือรูปแบบไฟล์ GIF เนื่องจากซอฟต์แวร์ที่ใช้ในการจัดการระบบมัลติมีเดีย ไม่สามารถเรียกใช้ไฟล์ในรูปแบบ JPEG ได้

เรื่องของภาพเคลื่อนไหว (Animation Image)

ในงานมัลติมีเดียข้อมูลประเภทภาพเคลื่อนไหวเป็นข้อมูลที่น่าสนใจและสร้างความประทับใจให้กับผู้ใช้ได้มาก เพราะเป็นการผสมผสานระหว่างข้อมูลภาพและเสียง

ข้อมูลประเภทภาพเคลื่อนไหวจะมีวิธีการเก็บได้หลายวิธี มีทั้งชนิดที่ไม่ได้มีการบีบ ขนาดข้อมูลและที่มีการบีบขนาดข้อมูลด้วยการใช้ซอฟต์แวร์และฮาร์ดแวร์ ทำให้เกิดมีมาตรฐานของ Movie File ขึ้นมามากมายและที่เป็นที่นิยมในปัจจุบัน เช่น มาตรฐาน MPEG, มาตรฐาน Indeo, Quicktime, Video for Windows, Motion JPEG, Fractals เป็นต้น

สำหรับโครงการมัลติมีเดียที่ผู้เขียนทำการสร้างขึ้นจะเลือกใช้มาตรฐาน Video for

Windows เพราะว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) สามารถใช้กับ Codec (Co/Decompression) ได้หลายแบบ
 - 2) เป็นมาตรฐานของภาพเคลื่อนไหวของ Windows ทำให้มีโปรแกรมที่สร้างและแก้ไข รวมถึงการนำไปใช้ได้มากมาย
 - 3) ฮาร์ดแวร์ที่ใช้ในการทำโครงการสามารถใช้มาตรฐานนี้ได้เท่านั้น
- เราจะพิจารณา Codec ต่าง ๆ ที่สามารถใช้ได้ในช่วงของ Video for Windows 1.1e ซึ่งจะมีมาตรฐานต่างๆ ดังนี้

- Cinepak Codec โดย SuperMatch
- Microsoft Video1
- Intel Indeo
- Microsoft RLE

การเปรียบเทียบ Codec ต่าง ๆ

หลังจากการทดสอบสร้างไฟล์ AVI ด้วย Adobe Premiere 4.0 ด้วย Codec ต่างๆ เพื่อเปรียบเทียบคุณสมบัติบางประการที่จะนำมาช่วยในการตัดสินใจเลือกวิธี Codec ที่จะใช้ โดยไฟล์ AVI ที่สร้างขึ้นจากไฟล์ AVI เดิมที่มีอยู่แล้วนำมาสร้าง Effect โดย Adobe Premiere ให้ได้ไฟล์ AVI ใหม่มีความยาว 20 วินาทีและมีการแทรกเสียง (Wave File) ด้วย โดยเสียงที่แทรกเข้าไปได้ใช้เทคโนโลยีการบีบข้อมูลเสียงแบบ IMA ADPCM จะได้ข้อมูลต่างๆ ที่น่าสนใจดังตารางที่ 1

	คุณภาพของภาพ	เวลาที่ใช้ในการสร้าง	ขนาดของข้อมูล	อัตราส่วนการบีบอัดข้อมูล
ไม่มีการบีบข้อมูล	100%	2 นาที	23,163,722	1:1
Cinepak Codec	75%	16 นาที	1,933,560	12:1
Video1 256 ที	75%	3 นาที	2,640,638	9:1
Video1 1024 ที	75%	3 นาที	1,917,722	14:1
Intel Indeo R3.2	75%	6 นาที	1,204,422	19:1
Intel Indeo Raw	ไม่มีข้อมูล	2 นาที	6,603,722	4:1
Microsoft RLE	75%	1 นาที	4,477,386	5:1

ตาราง 1 เปรียบเทียบ codec ต่างๆ ที่ทำให้ได้คุณสมบัติของไฟล์ต่างกัน

จากตาราง กรณีที่ไม่มีการบีบอัดข้อมูล จะได้คุณภาพของภาพสมบูรณ์มากที่สุด แต่ขนาดของข้อมูลก็มากที่สุดด้วย ซึ่งเมื่อเปรียบเทียบกับ Codec แบบ Intel Indeo R3.2 จะสามารถบีบอัดข้อมูลได้มากถึง 19 เท่า ทำให้สามารถประหยัดเนื้อที่ในการจัดเก็บ

ในด้านคุณภาพของภาพ ทดลองใช้ Picture Publisher 4.0 เรียกไฟล์ AVI เพื่อดูภาพในเฟรมเดียวกันของแต่ละ Codec แล้วทำการขยายเพื่อพิจารณาคุณภาพของภาพปรากฏว่าไม่มีความแตกต่างกันมากนัก

เรื่องของเสียง

เสียงนับได้ว่าเป็นสื่อแรกที่มนุษย์ใช้ และเสียงหรือภาษาพูดก็เป็นสื่อที่มีความสำคัญไม่น้อยไปกว่าภาษาเขียน การผลิตภาษาเชิงภาพจะมีขั้นตอนยุ่งยากกว่าภาษาเสียง ซึ่งอาศัยหูเป็นสื่อกลางเพียงอย่างเดียวก็ใช้งานได้แล้ว อย่างไรก็ตาม เสียงก็มีข้อจำกัดที่ลดลงไปมากกว่าในอดีต เนื่องจากการพัฒนาสื่อทางเสียงได้รับการพัฒนาไปเร็วกว่าสื่อทางภาพ และที่สำคัญการพัฒนาสื่อทางเสียงจะไม่ร้ายแรงเท่า กับความผิดพลาดในเชิงภาพ ทั้งนี้เพราะความผิดพลาดของเสียงจะมีผลแต่เพียงทำให้ความชัดเจนของเสียงลดลงหรือเพี้ยนไปจากเดิม เช่น หูมกเกินไป แลลมกเกินไป เป็นต้น แต่ถ้าเป็นความผิดพลาดในลักษณะของสื่อทางภาพแล้ว อาจถึงขั้นทำให้ใช้งานไม่ได้เลยทีเดียว ระบบมัลติมีเดียจึงเป็นการประสานกันระหว่างสื่อทางภาพกับสื่อทางเสียงนั่นเอง

Wave File

ในงานมัลติมีเดียเสียงมีความสำคัญมากในการนำเสนอ ความรู้เกี่ยวกับการใช้งาน Wave File ก็คือการเลือกชนิดของ Wave File และวิธีการบีบข้อมูลของมัน

การใช้งาน Wave File ใน Windows เราจะใช้ Wave File มาตรฐานของ Windows ซึ่งทุก ๆ Application บน Windows สามารถเรียกใช้ได้ คุณภาพเสียงสามารถบันทึกและเล่นกลับได้ตามความสามารถของ SoundCard ที่ต้องสามารถบันทึกและเล่นกลับเสียง ที่มีคุณภาพเสียง 16 Bit (CD Quality) Sampling Rate สูงสุด 44.4 KHz

การบีบขนาดข้อมูลเสียง

การใช้ข้อมูลเสียง ถ้าเราไม่นำวิธีการบีบข้อมูลเข้ามาช่วยอาจทำให้เราต้องใช้เนื้อที่ในการเก็บข้อมูลขนาดมหาศาล ในทางปฏิบัติโดยทั่วไปจะมีการบีบอัดข้อมูลเสียงเพื่อลดขนาดของเนื้อที่ที่ใช้เก็บ Wave File

การบีบขนาดข้อมูลเสียงในปัจจุบันมีวิธีการอยู่หลายชนิดเช่น

- PCM (Pulse Code Modulation)
- Microsoft ADPCM (Adaptive Delta Pulse Code Modulation)
- IMA ADPCM

การเปรียบเทียบ Wave File ที่ใช้วิธีการบีบอัดเสียงแบบต่าง

สำหรับชนิดของข้อมูลเสียงที่นำมาทดสอบปัจจัยที่ต่างกันดังนี้

- Sampling Rate เป็นความถี่ของการ Sampling มีค่าโดยทั่วไปเท่ากับ 11.025 KHz, 22.05 KHz, 44.1 KHz ถ้า Sampling Rate มีค่ามากความคมชัดของเสียงจะมีมากขึ้น (44.1 KHz คือคุณภาพระดับ CD Quality) ใน Sound card 16 Bit โดยทั่วไปจะสามารถบันทึกเสียงที่มี Sampling Rate ได้สูงเท่ากับ 44.1 KHz

- Channel สามารถกำหนดได้ว่าต้องการบันทึกเสียงแบบ Mono หรือ Stereo ทั้งนี้ต้องขึ้นอยู่กับความสามารถของ Sound Card ที่เราใช้ด้วย

- Sample Size ขนาดของการ Sampling กำหนดได้ว่าต้องการ Sampling Size เป็น 8 bit หรือ 16 bit

ในการทดสอบจะทำการบันทึกเสียงผ่าน Microphone ให้มีคุณลักษณะดังนี้

- Sampling Rate 44.1 KHz

- Channel Stereo

- Sampling 16 Bit

โดยใช้เป็นพื้นฐานแล้วทำการแปลงให้อยู่ในรูปแบบที่เราต้องการ โดยใช้ Media Studio Audio Editor เพื่อเปรียบเทียบความแตกต่างของ

- ขนาดของ File

- คุณภาพของเสียง

เสียงที่ใช้ทดสอบเป็นเสียงนับ 1 ถึง 10 เป็นเวลาทั้งหมดเท่ากับ 8.686 วินาที เมื่อผ่านการบีบข้อมูลด้วยวิธีต่าง ๆ แล้ว ได้ข้อมูลที่น่าสนใจดังตารางข้างล่าง

ชนิดของการบันทึกข้อมูล	16 Bits	8 Bits
PCM Stereo 44 KHz	1,532,380	766,212
PCM Stereo 11 KHz	383,128	191,586
Microsoft ADPCM Stereo 11 KHz	-	98,394
IMA ADPCM Stereo 11 KHz	-	98,364
PCM Mono 11 KHz	-	95,816
Microsoft Mono 11 KHz	-	49,242
IMA ADPCM Mono 11 KHz	-	49,212

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบคุณภาพของเสียงที่ได้ยินพบว่าในระดับงาน Presentation ที่เป็นการพูดบรรยายนั้น เราสามารถใช้ ปัจจัยที่ทำให้ File มีขนาดเล็กที่สุดได้ คือ

- 11.025 KHz
- Mono Channel
- 8 Bits Sampling Size

จากการทดสอบ โดยการฟังเสียงที่ทดสอบพบว่าคุณภาพของเสียงอยู่ในระดับที่น่าพอใจแม้ว่าเราจะใช้การบันทึกเสียงแบบ 8 Bits, Mono, 11.025 KHz ก็ตาม

และจากการทดสอบวิธีการในการบีบอัดข้อมูลเสียงสามารถสรุปได้ดังตารางข้างล่าง

	PCM	MS-ADPCM	IMA ADPCM
ความเร็วในการบันทึกข้อมูล	มากที่สุด	ต่ำที่สุด	ค่อนข้างมาก
ความเร็วในการเรียกข้อมูล	เร็วที่สุด	ช้ากว่าเล็กน้อย	ช้ากว่าเล็กน้อย
ขนาดของข้อมูลหลังการบันทึกข้อมูล	ใหญ่ที่สุด	เล็ก	เล็กที่สุด

ตาราง 3 เปรียบเทียบวิธีการบีบอัด

จากข้อมูลดังกล่าวทำให้เราสามารถพิจารณาเลือกใช้วิธี IMA ADPCM ในการบีบอัดข้อมูลเสียง เพราะว่ามีอัตราการบีบอัดข้อมูลที่ดีและมีความเร็วในการบีบและขยายข้อมูลที่ดี

การพิจารณาวิธีการบีบอัดข้อมูลนี้ยังมีประโยชน์กับการสร้าง AVI File อีกด้วยเพราะในการสร้าง AVI File นอกจากเราจะกำหนดวิธีการบีบอัดข้อมูลภาพแล้วยังสามารถระบุวิธีการบีบอัดข้อมูลเสียงใน AVI File ดังกล่าวด้วย

การตอบโต้ซึ่งกันและกัน

คำว่า ตอบโต้กันได้ (Interactive) หมายถึง แสดงผลตอบสนองการทำงานว่าถูกต้องหรือไม่ ใช้ได้หรือไม่ได้ในขณะนั้น ระบบมัลติมีเดียต้องมีการตอบโต้กันได้จึงจะสมบูรณ์ จากลักษณะอันนี้เองที่ทำให้ขอบเขตของมัลติมีเดียกว้างขวางออกไปจำแนกออกได้ดังนี้

1. การตอบโต้กันได้ ในลักษณะตัวเลือก มีการจำลองสถานการณ์ (Simulation) จากปัญหาหนึ่งไว้หลายรูปแบบ โดยพิจารณาจากเงื่อนไขที่แตกต่างกัน หากสามารถจำลองสถานการณ์ได้มาก และซับซ้อนครอบคลุมได้มากเท่าไร จะทำให้ระบบมัลติมีเดียนั้นสมบูรณ์มากขึ้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การตอบโต้กันได้ในลักษณะการติดต่อสื่อสารใช้สมรรถนะของการคมนาคมที่ทันสมัย
 เข้าช่วยผลลัพธ์ที่ได้จะปรับเปลี่ยนตามเหตุการณ์ที่เปลี่ยนไป เช่น เดิมถ้าต้องการทราบข้อมูลเพียง
 ว่า มีโรงแรมอะไรบ้างเท่านั้น แต่ปัจจุบันเมื่อเราเลือกโรงแรมใดแล้ว เราจะทราบถึงขนาดว่า
 โรงแรมนั้นมีห้องพักกี่ห้องและมีห้องว่างหรือไม่ด้วย เป็นต้น

สำหรับในงานมัลติมีเดียเราควรจะทำความรู้จักกับ ไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นรูป
 แบบของการจัดข้อมูล แต่จะมีความแตกต่างจาก Database โดยทั่วไปในด้านรูปแบบ และหน้าที่
 การทำงานต่างๆ เช่น การ Browsing, nodes และ การเชื่อมต่อ (Link)

ไฮเปอร์เท็กซ์ คืออะไร

ไฮเปอร์เท็กซ์ คือ สารบัญญัตินำเสนอในเทคโนโลยีใหม่ ที่สามารถกระโดดไปยังส่วนที่เกี่ยวข้อง
 ช้องกัน เพื่อแสดงความหมาย ความสัมพันธ์ หรือ แสดงการตอบโต้ (Interactive) กับผู้ใช้ เช่น ใน
 On-line Help ในวินโดว์

ในปัจจุบันไฮเปอร์เท็กซ์ถูกนำเสนอในรูปแบบของหน้าต่าง ที่สามารถจะใส่ข้อมูล เช่น ตัว
 อักษร (text), กราฟิก (graphic), รูปภาพ, เสียง, ภาพเคลื่อนไหว ซึ่งได้มีการรวบรวมมาตรฐานต่างๆ
 ของไฮเปอร์เท็กซ์ เรียกว่า “Standard General Markup Language (SGML)” ซึ่งจะอนุญาตให้
 โปรแกรมสามารถเชื่อมต่อไปยัง แอปพลิเคชัน ต่างๆ ได้
 ความเข้าใจกับสแกนเนอร์

สแกนเนอร์เป็นอุปกรณ์ที่จะถ่ายทอดภาพนิ่งเข้าสู่คอมพิวเตอร์ เช่น การสแกนภาพถ่าย
 หรือตัวอักษรที่ต้องการความละเอียด ดังนั้น จึงสามารถให้สแกนเนอร์จับภาพแบบกราฟิกหรืออื่น
 ๆ ลงในงานเอกสารหรือในแอปพลิเคชันต่าง ๆ ที่ต้องการได้ เมื่อสแกนภาพเข้าไปในแอปพลิเคชัน
 ภาพจะถูกสร้างขึ้นในลักษณะของจุด(dot) เรียกว่า reset image ซึ่งจะแตกต่างกับ Vector Image ที่
 ถูกสร้างจากเส้นตรง ส่วนของเส้นตรงและลักษณะทางเลขาคณิตต่าง ๆ

สแกนเนอร์มีระดับความสามารถในการสแกนต่าง ๆ กัน ตั้งแต่ 300 dpi (dots per inch) ถึง
 800 dpi ซึ่งมีราคาและประสิทธิภาพต่างกันตามแต่การใช้งาน สแกนเนอร์ที่มีราคาค่อนข้างต่ำคือ
 แบบ Line-art โดยมากจะเป็นแบบใช้มือถือซึ่งมีขนาดเล็กจับภาพที่ซับซ้อนกันมากนัก เมื่อนำมาจับ
 ภาพสีหรือภาพกราฟิก จะไม่ค่อยได้คุณภาพนัก สแกนเนอร์ขนาดเล็กก็มีแบบที่สามารถสแกนภาพ
 สีได้ โดยเคลื่อนที่เองเพื่อลดความผิดเพี้ยนจากต้นแบบเช่น Color Mobile ซึ่งให้สีได้ถึง 16.7 ล้านสี
 ที่มีความละเอียด 400 dpi

ลักษณะของสแกนเนอร์แบ่งได้เป็น ขนาดเล็กที่มือถือได้เรียกว่าแบบ Hand-held และ
 Flabled ที่ขนาดใหญ่ขึ้น ซึ่งแบ่งเป็นประเภทการใช้งานดังนี้

Black-and-White,Line-Art Scanners เหมาะสำหรับภาพขาวดำที่ไม่ซับซ้อนมากนักเช่น ภาพลายเส้น :256- Color Gayscale and Halftone Scanners จับภาพกราฟิก แต่แปลงสีออกเป็นเฉด (Shades) ของสีเทา

: Color scanners สามารถจับภาพสีได้เต็มรูปแบบซึ่งจะมีความเพี้ยนจากต้นแบบน้อยที่สุด เหมาะสมกับงานกราฟิกที่ต้องการความละเอียดของภาพและความสมบูรณ์ของสี

หมายเหตุ ถ้าต้องการใช้งานมัลติมีเดียมีสีต้นและมีคุณภาพ การใช้สแกนเนอร์ที่มีความละเอียดและความเพี้ยนจากต้นแบบน้อย จะช่วยให้งานสมบูรณ์ขึ้น โดยพิจารณาความละเอียดจาก 300 dpi ขึ้นไป โดยที่แบบ Hand-held จะมีราคาต่ำกว่า แต่ก็มีความจำกัดในการจับภาพขนาดใหญ่ ซึ่งสแกนเนอร์แบบ Flatbed จะมีประสิทธิภาพที่ดีกว่าแต่ก็มีราคาสูง

ข้อควรระวัง ควรที่จะตรวจสอบไครเวอร์ของสแกนเนอร์ให้สอดคล้องกับแอปพลิเคชันที่ต้องการใช้เพื่อความมั่นใจว่าสแกนเนอร์ จะสามารถใช้งานร่วมกับแอปพลิเคชันที่ต้องการได้อย่างถูกต้อง

ความเข้าใจกับวิดีโอ

การผสมผสานภาพวิดีโอกับระบบคอมพิวเตอร์เป็นจุดแปลกใหม่ของการใช้งานทางคอมพิวเตอร์และนับว่าเป็นการพัฒนาทางด้านมัลติมีเดียที่น่าทึ่งความสามารถในการนำไปใช้ประโยชน์กับงาน เราสามารถเก็บภาพวิดีโอ หรือเป็นภาพนิ่งลงในงาน และบันทึกไว้เพื่อเรียกใช้งานในภายหลัง พร้อมทั้งเสียงที่เกิดขึ้น ด้วยการนำภาพจากสัญญาณวิดีโอสู่ทางมอนิเตอร์ของเครื่องพีซี ทำให้เราสามารถดูวิดีโอทางพีซีได้ และยังสามารถนำภาพวิดีโอ เหล่านั้นมารวมกับ Application ที่ใช้งานอยู่ในรูปของภาพนิ่งหรือภาพต่อเนื่องของวิดีโอก็ได้ หรือถ้ามีกล้องวิดีโอก็สามารถถ่ายภาพต่อเนื่องแบบวิดีโอ เข้ามาบนพีซีได้เช่นกัน

ความแตกต่างของสัญญาณวิดีโอกับสัญญาณคอมพิวเตอร์นั้น เป็นสิ่งที่ต้องพิจารณา เพราะสัญญาณวิดีโอ เป็นสัญญาณแบบอนาลอก แต่มอนิเตอร์ของเครื่องพีซี รับสัญญาณในรูปแบบของดิจิทัล จึงต้องมีสัญญาณในการแปลง ให้สัญญาณเข้าและออกสอดคล้องกัน และการบีบอัดสัญญาณและการเล่นกลับเรียกว่า Codec (Compression/Dedompress) ซึ่งแบ่งประเภทของสัญญาณที่ควรทราบดังนี้

1. NTSC (National Television /standards Committee)

NTSC ได้ รับการกำหนดขึ้นมาเป็นมาตรฐานของสัญญาณวิดีโอในทวีปอเมริกาเหนือ ซึ่งเครื่องเล่นในทวีปอเมริกาเหนือ ใช้สัญญาณ NTSC

2. PAL

PAL คือมาตรฐานที่กำหนดขึ้นมาจากกลุ่มประเทศทางยุโรป ซึ่งเทียบได้กับระบบ NTSC เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของอเมริกาเหนือ แต่ทั้ง PAL และ NTSC จะแตกต่างจากสัญญาณที่ใช้ในทวีปยุโรปโดยมากจะใช้ PAL เพื่อความแน่ใจทางด้าน Hardware ซึ่งจะตรวจสอบสัญญาณ Input และ Output ด้วยสัญญาณ PAL

3. AVI (Audio Video Interleave)

.AVI เป็นเทคโนโลยีที่ถูกพัฒนาขึ้นโดยไมโครซอฟต์ เพื่องานวิดีโอสำหรับผลิตภัณฑ์สำหรับวินโดวส์และสำหรับซอฟต์แวร์ ที่เล่นอย่างเดียวยุโรปสำหรับการเคลื่อนไหวกแบบวิดีโอโดยสมบูรณ์ การจับภาพวิดีโอจะต้องใช้ Hardware พิเศษสำหรับการบีบอัดข้อมูล แต่สามารถเล่น AVI ได้ โดยไม่ต้องมี Hardware พิเศษแต่อย่างใด

4. DVI (Digital Video Interactive)

DVI คือการรวมกันระหว่าง Hardware และ ซอฟต์แวร์ ซึ่งพัฒนาโดย Intel และ IBM DIV สามารถบีบอัดข้อมูลได้ในอัตราส่วน 150: 1 และเล่นกลับได้อย่างสมบูรณ์

5. JPEC (Joint Photograph Experts Group)

.เป็นมาตรฐานการบีบอัดข้อมูลสำหรับภาพนิ่ง

6. MPEC (Motion Pixruew Feoup)

เป็นมาตรฐานการบีบอัดข้อมูลสำหรับภาพเคลื่อนไหว

7. Quick Time

คือ วิดีโอ Codec ซึ่งพัฒนาโดย Apple สำหรับ เครื่อง Machintosh โดยเฉพาะ

หมายเหตุ วิดีโอบอร์ดโดยมากจะรองรับสัญญาณ NTSC และ PAL เพื่อการใช้งานที่คุณสามารถเรียกใช้ได้ตลอดตัวขึ้น ถึงแม้ภาพเคลื่อนไหวของวิดีโอจะเป็นส่วนที่น่าสนใจใน มัลติมีเดีย แต่ก็ยังมีความซับซ้อนในการใช้พอสมควร โดยเฉพาะการจับภาพวิดีโอและแปลงจากสัญญาณอนาล็อกไปเป็นสัญญาณข้อมูลภาพ เพื่อการใช้งานหรือการบีบอัดข้อมูลขนาดใหญ่

.การที่สามารถเห็นภาพเคลื่อนไหวจากมอนิเตอร์ ต้องใช้ความเร็วระหว่าง 15 ถึง 30 เฟรมต่อวินาที ซึ่งเป็นปริมาณข้อมูลขนาดใหญ่ ทำให้เกิดปัญหาต่าง ๆ ตามมา เช่น ถ้าต้องการจัดเก็บข้อมูลเพื่อการเล่นกลับ ต้องใช้พื้นที่ที่ขนาดมหาศาล และการส่งผ่านข้อมูลเพื่อการแสดงผล จึงต้องใช้ขบวนการบีบอัดข้อมูล (Data Compression) เข้ามาช่วย

หมายเหตุ การเคลื่อนไหวของวิดีโอที่เหมือนจริง สามารถดูได้อย่างราบรื่น จะต้องใช้ความเร็วที่ 30เฟรมต่อวินาที และถ้าน้อยกว่า 24 เฟรมต่อวินาที ภาพที่ออกมาจะสามารถสังเกตเห็นจังหวะการเปลี่ยนเฟรมได้ ทำให้ไม่ราบเรียบเรียกว่า ภาพ “ choppy “

มัลติมีเดียในประเทศไทย

มัลติมีเดียได้เข้ามาในประเทศไทยนานแล้วแต่การใช้งานยังจำกัดอยู่แค่การใช้เพื่อการศึกษาและใช้เพื่อความบันเทิงเท่านั้น ทั้งนี้อาจเป็นเพราะระบบสร้างงานมัลติมีเดียต้องประกอบด้วยเครื่องคอมพิวเตอร์ความเร็วสูง ซอฟต์แวร์โปรแกรมกราฟิก และซอฟต์แวร์สำหรับใช้งานด้านมัลติมีเดียที่มีราคาสูง อีกทั้งผู้ค้าชาวไทยเกรงว่า หากพัฒนาระบบมัลติมีเดียเสร็จแล้ว จะมีผู้อื่นนำไปลอกเลียนแบบอีกทอดหนึ่ง อย่างไรก็ตามปัจจุบัน เมื่อพิจารณาจากยอดขายสินค้าระบบมัลติมีเดียแล้วจะพบว่าระบบมัลติมีเดียบนเครื่องพีซีนั้น มีผู้คาดการณ์ว่าจะเติบโตขึ้นถึง 40% ในปี 2540 จากแนวโน้มของความเป็นไปได้ในการขยายตัวของตลาดมัลติมีเดียในเมืองไทย ในช่วงที่ผ่านมามียอดขายเพิ่มขึ้นเรื่อย ๆ นี้เองที่ส่งผลให้อุตสาหกรรมใช้งานด้านมัลติมีเดียย่อมมีมากขึ้นเป็นเงาตามตัวด้วย ดังนั้น ไม่ใช่เรื่องง่ายเลยที่จะทำนายทิศทางในอนาคตของระบบมัลติมีเดียว่า จะเป็นอย่างไร แต่พอจะกล่าวได้ว่า ระบบมัลติมีเดียจะเข้าสู่ตลาดเพื่อการอุปโภคและบริโภคอย่างแน่นอน

บทสรุป

ระบบมัลติมีเดียนับได้ว่าเป็นเทคโนโลยีที่เราควรจับตามอง เพราะเราจะสามารถนำภาพเสียงกราฟิก วิดีโอและข้อความมาผสมผสานเพื่อนำเสนอข้อมูลให้ดึงดูดความสนใจได้อย่างมากไม่ว่าจะเป็นการนำไปใช้ในวงการภาพยนตร์ การผลิตงานโฆษณาทางโทรทัศน์ การผลิตหนังสือ และนับวันยังจะเป็นที่นิยมขึ้นไปเรื่อย ๆ แม้ระบบที่มีอยู่ในปัจจุบันจะยังมีข้อจำกัดอยู่มาก แต่ทางผู้ผลิตก็กำลังพัฒนาเทคโนโลยีใหม่ ๆ ขึ้นมา เพื่อแก้ปัญหาที่มีอยู่ในขณะนี้ อีกทั้งบริษัทซอฟต์แวร์ก็มีการผลิตซอฟต์แวร์ของมัลติมีเดียออกสู่ตลาดมากขึ้นด้วย ดังนั้น ตลาดของมัลติมีเดียซึ่งเคยจำกัดอยู่ในวงแคบ ๆ ก็กำลังขยายตัวออกไปอย่างกว้างขวาง ระบบมัลติมีเดียจึงเป็นสิ่งที่เข้ามามีบทบาทอย่างมากต่อวงการคอมพิวเตอร์ แม้การพัฒนาเทคโนโลยีมัลติมีเดียจะเพิ่งเริ่มต้น แต่เชื่อแน่ว่า ยังมีการพัฒนาไปอีกไกลอย่างแน่นอน

บทที่ 2

ทฤษฎีและเอกสารที่เกี่ยวข้อง

บทเรียนสำเร็จรูปทางคอมพิวเตอร์ (CAI)

บทเรียนสำเร็จรูปทางคอมพิวเตอร์ (Computer Assisted Instruction หรือ CAI) เป็นการนำบทเรียนประกอบเข้ากับสื่อ คือคอมพิวเตอร์ โดยจะต้องสร้างบทเรียนก่อน จากนั้นจึงนำบทเรียนมาลำดับเนื้อหาและรูปแบบที่ต้องการแสดง โดยใช้ความสามารถของโปรแกรมคอมพิวเตอร์ช่วยในการนำเสนอ โดยอาจมีภาพเคลื่อนไหว, เสียง, มีการทดสอบและสามารถตรวจสอบผลการทดสอบได้ นับเป็นการพัฒนาบทเรียนโดยใช้สื่อที่ทันสมัยเพื่อให้เรียนสนใจยิ่งขึ้น

ชนิดของบทเรียนสำเร็จรูปทางคอมพิวเตอร์ แบ่งตามการตอบสนองของผู้เรียน ได้ 2 แบบ ดังนี้

1. แบบเชิงเส้น (Linear)

หมายถึง บทเรียนที่จัดลำดับการเรียนรู้จากการตอบสนองของผู้เรียนให้เหมือนกันทุกคน แบบเรียนมีการจัดลำดับการเรียนรู้จากง่ายไปหายาก ผู้เรียนจะต้องเริ่มจากหน่วยแรกและเรียนไปตามลำดับจนถึงหน่วยสุดท้ายเหมือนกันทุกคน

2. แบบไม่เชิงเส้น (Branching)

หมายถึง บทเรียนที่จัดลำดับการเรียนรู้ตามการตอบสนองของนักเรียนแต่ละคน ทุกคนได้มีโอกาสเรียนตามความสามารถของตนเอง ซึ่งอาจมีขบวนการเรียนรู้ไม่เหมือนกับคนอื่น ๆ ผู้เรียนอาจข้ามหน่วยย่อยใดหน่วยหนึ่งซึ่งเข้าใจแล้วไปยังหน่วยต่อไปได้ การเรียนจะไม่ดำเนินไปตามลำดับขั้นแต่จะขึ้นอยู่กับความสามารถของผู้เรียน

คุณสมบัติของ (CAI)

การใช้คอมพิวเตอร์เป็นตัวเสนอเนื้อหาสามารถให้ประโยชน์ในด้านต่างๆเหล่านี้คือ

1. เสนอเนื้อหาได้รวดเร็วกว่า แทนที่ผู้เรียนจะต้องเปิดหนังสือที่หลายๆหน้าก็เพียงแต่กดแป้นพิมพ์หรือคลิกเมาส์เพียงครั้งเดียวเท่านั้น
2. คอมพิวเตอร์สามารถเสนอรูปภาพเคลื่อนไหวได้
3. มีเสียงประกอบได้ ทำให้เกิดความน่าสนใจได้มากขึ้น
4. เก็บข้อมูลเนื้อหาได้มากกว่าหนังสือหลายเท่า
5. มีการโต้ตอบกับผู้เรียนได้
6. สามารถควบคุมไม่ให้ผู้เรียนโกงตัวเองโดยการเปิดผ่านเนื้อหาต่างๆไปได้
7. สามารถประเมินผลการเรียนและประเมินผู้เรียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. สามารถนำคิดค้นไปเรียนในที่ต่าง ที่มีเครื่องคอมพิวเตอร์ได้โดยไม่จำกัดด้านเวลาและสถานที่
9. เหมาะสำหรับการเรียนการสอนผ่านการสื่อสาร เช่น การจัดการศึกษาทางไกลผ่านดาวเทียม เป็นต้น

การออกแบบบทเรียน (Course Designing)

การออกแบบและพัฒนาบทเรียนมีขั้นตอนต่างๆดังนี้

1. การวิเคราะห์หลักสูตรและเนื้อหา (Course Analysis)

ขั้นตอนนี้เป็นขั้นที่นับว่าสำคัญที่สุดของขบวนการออกแบบบทเรียน คือการวิเคราะห์ความต้องการของหลักสูตรที่จะนำมาสร้างเป็นบทเรียน CAI ส่วนของเนื้อหาบทเรียนจะได้มาจากการศึกษาและวิเคราะห์รายวิชาแต่ละวิชา

2. กำหนดวัตถุประสงค์ของบทเรียน

เป็นการกำหนดแนวทางที่คาดหวังให้ผู้เรียนมีความสามารถในเชิงรูปธรรมหลังจากที่จบบทเรียนแล้ว วัตถุประสงค์เชิงพฤติกรรมนี้จะครอบคลุมขอบข่ายของเนื้อหาที่ได้จากการวิเคราะห์ในขั้นตอนที่1ซึ่งจะสอดคล้องกับหัวข้อเรื่องย่อยๆที่จะนำมาสร้างบทเรียน

3. การวิเคราะห์เนื้อหาและกิจกรรม

ขั้นตอนนี้จะยึดตามวัตถุประสงค์เชิงพฤติกรรมเป็นหลัก โดยทำการขยายรายละเอียด ดัง

3.1 กำหนดเนื้อหา กิจกรรมการเรียนรู้และ Concept ที่คาดหวังให้ผู้เรียนเรียนรู้

3.2 เขียนเนื้อหาสั้นๆ ทุกหัวข้อย่อยให้สอดคล้องกับวัตถุประสงค์เชิงพฤติกรรม

3.3 เขียน Concept ทุกๆหัวข้อย่อยจากนั้นนำมาดำเนินตามขั้นตอนดังนี้

3.3.1 จัดลำดับเนื้อหาได้แก่

บทนำ

ระดับของเนื้อหาและกิจกรรม

ความต่อเนื่องของเนื้อหาแต่ละเฟรม

ความยากง่ายของเนื้อหา

เลือกสื่อที่จะช่วยให้เกิดการเรียนรู้ในแต่ละกิจกรรม

3.3.2 เขียนผังงาน (Layout Content) โดยการ

แสดงการเริ่มต้นและจุดจบของเนื้อหา

แสดงการเชื่อมต่อและความสัมพันธ์ของบทเรียน

แสดงเนื้อหา เลือกใช้แบบสาขาหรือเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การดำเนินบทเรียนและวิธีการเสนอเนื้อหาและกิจกรรม

3.3.3 การออกแบบจอภาพและแสดงผล ได้แก่

บทนำและวิธีการใช้โปรแกรม

การจัดเฟรมแต่ละหน้าจอ

การให้ สี แสง เสียง ภาพ และกราฟฟิกต่างๆ

การเลือกรูปแบบตัวอักษร

การตอบสนองและการโต้ตอบ

การแสดงผลบนจอภาพ

4. การกำหนดขอบข่ายของบทเรียน

หมายถึงการกำหนดความสัมพันธ์ ของเนื้อหาแต่ละหัวข้อย่อยเพื่อหาความสัมพันธ์ ระหว่างบทเรียนจะได้ทราบถึงแนวทางที่ผู้เรียนจะต้องศึกษาต่อไป ถ้าบทเรียน CAI ที่ออกแบบขึ้น เพียงบทเรียนเรื่องเดียวก็อาจจะข้ามการพิจารณาขั้นตอนนี้ไปได้

5. การกำหนดวิธีการนำเสนอ

ได้แก่การเลือกรูปแบบการนำเสนอเนื้อหาในแต่ละเฟรมว่าจะใช้วิธีการแบบใด โดยสรุปจากขั้นตอนที่ 3 และ 4 นำมากำหนดรูปแบบการนำเสนอ เช่น การออกแบบเฟรม การแสดง ภาพกราฟฟิกบนจอ ฯลฯ

6. การสร้าง Storyboard ของบทเรียน

Storyboard หมายถึงเรื่องราวของบทเรียนที่ประกอบด้วยเนื้อหาที่แบ่งเป็นเฟรมๆตาม วัตถุประสงค์และรูปแบบการนำเสนอ โดยร่างเป็นแต่ละเฟรมย่อยเรียงลำดับตั้งแต่เฟรมที่ 1 จนถึง เฟรมสุดท้าย Storyboard จะต้องระบุภาพ พร้อมเงื่อนไขต่างๆที่เกี่ยวข้องในแต่ละเฟรม เช่น ลักษณะของภาพ เสียงประกอบ ความสัมพันธ์กับเฟรมอื่นๆของบทเรียน

Storyboard จะใช้เป็นแนวทางในการสร้างบทเรียนช่วยสอนต่อไป ดังนั้นการสร้าง Storyboard ที่ยิ่งละเอียดสมบูรณ์มากก็จะทำให้การสร้างบทเรียนจริงเป็นระบบมากขึ้น

7. การสร้างบทเรียน

ภายหลังที่ได้ออกแบบ Storyboard แล้ว ขั้นตอนนี้ จะดำเนินการตาม Storyboard ที่วางไว้ทั้งหมดตั้งแต่ การออกแบบเฟรม การกำหนดสี รูปแบบของตัวอักษร เงื่อนไขการโต้ตอบ การสร้างเสียง ความสัมพันธ์ระหว่างเฟรม ฯลฯ

8. การตรวจสอบและประเมินก่อนนำไปใช้

ขั้นตอนสุดท้ายก่อนการนำบทเรียน CAI ไปใช้งาน จำเป็นที่จะต้องผ่านขบวนการตรวจสอบและการประเมินตัวบทเรียน CAI ว่ามีคุณภาพอย่างไร ซึ่งมีข้อพิจารณา ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.1 การตรวจสอบแต่ละขั้นตอนของการออกแบบบทเรียน

8.2 การทดสอบการใช้งานบทเรียน โปรแกรมบทเรียน CAI จำเป็นต้องมีการทดสอบก่อนการนำไปใช้งาน เพื่อเป็นการตรวจสอบความถูกต้องของเนื้อหาและความเหมาะสมของบทเรียน

8.3 การประเมินผลบทเรียน มีจุดประสงค์เพื่อประเมินผลตัวบทเรียน CAI และการประเมินผลสัมฤทธิ์ของผู้เรียน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ความเป็นมาของ เดลไฟ

หลังจากที่ไมโครซอฟท์ได้ออกวินโดวส์ 95 อย่างเป็นทางการแล้ว บอร์แลนด์ได้ออก Delphi 2.0 เพื่อใช้กับวินโดวส์ 95 และวินโดวส์เอ็นที ดังที่จะนามากล่าวในภาคผนวกนี้

โดยจะกล่าวเรื่องของ เดลไฟ 1.0 กับ 2.0/ข้อแตกต่าง/แบบข้อมูลใหม่/เมนู/เทียบโปรแกรมของ 2 รุ่น/คอมโปเนนต์ใหม่/Object Repository/ การสืบสกุลของหน้าต่างและ finalization

เดลไฟรุ่น 1.0 กับ 2.0

เดลไฟรุ่นแรกซึ่งเมื่อออกรุ่น 2.0 แล้วจึงได้ให้ชื่อรุ่นเป็นรุ่น 1.0 เดลไฟ 1.0 คือเดลไฟที่ใช้สำหรับติดตั้งกับวินโดวส์ 3.x หรือวินโดวส์ 95 เพื่อใช้พัฒนาโปรแกรม เป็นแอฟพลิเคชัน 16 บิต เพื่อนำไปรันบนวินโดวส์ 3.x หรือวินโดวส์ 95

สำหรับเดลไฟ 2.0 คือเดลไฟที่ใช้สำหรับติดตั้งกับวินโดวส์ 95 หรือวินโดวส์เอ็นที ซึ่งจะขอรวมเรียกเป็นวินโดวส์ 95/เอ็นที เพื่อใช้พัฒนาโปรแกรมเป็นแอฟพลิเคชัน 32 บิต และใช้สภาพแวดล้อมเช่นไดอะล็อกบ็อกซ์ตามแบบของวินโดวส์ 95 เพื่อนำไปรันบนวินโดวส์ 95 /เอ็นที

ในการนี้ยังไม่ควรถือว่าเดลไฟ 1.0 เป็นเดลไฟรุ่นเก่า หากเป็นเดลไฟสำหรับวินโดวส์ 3.x(ในเดลไฟ 2.0 จะมีเดลไฟ 1.0 อยู่ด้วยอยู่ในไดเรคทอรี delphi 16 ในแผ่นซีดี)

ชุดของเดลไฟ 2.0

เดลไฟ 2.0 ประกอบด้วยเดลไฟชุดต่าง ๆ รวม 3 ชุด คือ

1.Delphi Desktop 2.0 ซึ่งเดิมเรียกว่า Delphi คือเดลไฟรุ่นแรกตามที่กล่าวในหนังสือนี้มาแต่ต้น และเมื่อได้ออกรุ่น 2.0 แล้วได้เปลี่ยนชื่อเป็น Delphi Desktop 1.0

2.Delphi Developer 2.0 คือ Delphi Desktop 2.0 ที่มีโปรแกรมยูทิลิตี้ที่ช่วยในการพัฒนาโปรแกรมเพิ่มขึ้น

3.Delphi Client/Server 2.0 มีข้อแตกต่างกับชุด Desktop ในเรื่องงานฐานข้อมูลเท่านั้น

ชุดที่ใช้ในการเขียน

ในการเขียนหนังสือเล่มนี้ ผู้เขียนใช้ Delphi Desktop 1.0 บนวินโดวส์ 3.x เป็นหลัก แล้วนำโปรแกรมที่คอมไพร์แล้วบางโปรแกรมไปรันบนเครื่องที่ติดตั้งวินโดวส์ 3.11 และกับเครื่องที่ติดตั้งวินโดวส์ 95 ที่ไม่ได้ติดตั้งเดลไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แล้วนำซอร์สโปรแกรมทั้งหมดไปคอมไพล์ซ้ำด้วย Delphi Developer 2.0 บนวินโดวส์ 95 มีบางโปรแกรมที่จะต้องปรับแต่ง ตามที่จะกล่าวในหัวข้อ “เทียบโปรแกรม” เพื่อให้ใช้หนังสือนี้กับเคลฟ 2.0 ได้

และได้ใช้ Delphi Developer 2.0 ในส่วนที่เป็นชุดเคสโค้ดที่ออกแบบบนวินโดวส์ 95 เพื่อเตรียมโปรแกรมแสดงการใช้คอมโปเนนต์ใหม่ในเคลฟ 2.0 ตามที่จะกล่าวในหัวข้อ “คอมโปเนนต์ใหม่”

ข้อแตกต่าง

ข้อแตกต่างของเคลฟ 2.0 กับ 1.0 หรือคือสิ่งที่เพิ่มขึ้นในเคลฟ 2.0 (เคลฟ 1.0 เป็นซับเซตของเคลฟ 2.0) โดยหลักใหญ่คือทำให้ตนเองและโปรแกรมที่คอมไพล์ด้วยเคลฟ 2.0 เป็นแอฟพลิเคชัน 32 บิต ใช้สถาปัตยกรรมของวินโดวส์ 95 และทำให้ไฟล์ .EXE มีขนาดเล็กลงและรันได้เร็วขึ้นมีเรื่องที่เกี่ยวข้องกับหนังสือนี้ ตามที่จะนำมากล่าว (เช่น ไม่กล่าวถึงเรื่อง OLE ซึ่งจะอยู่ในเล่มสาม) สามเรื่องคือ

1. ทางด้านภาษา ตามที่จะกล่าวในหัวข้อ “แบบข้อมูลใหม่” และ เทียบ “โปรแกรม”
2. การสืบสกุลจาก TForm ตามที่จะกล่าวในหัวข้อ “การสืบสกุลของหน้าต่าง”
3. ทางด้านคอมโปเนนต์ ตามที่จะกล่าวในหัวข้อ “คอมโปเนนต์ใหม่” และแสดงการใช้ใน

หัวข้อที่ตามมา

แบบข้อมูลใหม่

ในเคลฟ 2.0 ได้เพิ่มแบบข้อมูล (type) ขึ้นอีก 4 แบบ คือ

1. Character type แบ่งเป็น

- AnsiChar 8 บิต

- WideChar 16 บิต

ในปัจจุบันให้ Char เท่ากับ AnsiChar

2. Shortring type

- ShortString คือสตริงดั้งเดิม เป็นสตริงอย่างสั้น

- AnsiString เป็นสตริงอย่างยาว ซึ่งยาวได้ถึง 2 กิกะไบต์ (ค่าบวกสูงสุดของ LongInt)

เป็นสตริงที่เป็นอะเรย์ของอักษร และเริ่มอักขระแรกที่ช่อง 1 เช่นกัน แต่ไม่มีช่องศูนย์ ซึ่งของสตริงเดิมหรือ ShortString ใช้เก็บความยาวของสตริง ส่วนของ AnsiString จะปิดท้ายด้วยอักขระ null (#0) และเก็บความยาวแยกไว้ต่างหาก

สำหรับคำว่า String ได้ให้คำตีฟอลต์ไว้เป็น AnsiString คือสตริงอย่างยาว โดยที่ถ้ากำหนดตัวแปร เช่น

```
var astr : String;
    bstr : String[ 255 ];
```

จะได้ astr เป็นสตริงอย่างยาว ซึ่งความยาวจะขึ้นกับข้อความที่จะให้ ส่วน bstr จะเป็นสตริงอย่างสั้นและจะกำหนดความยาวได้ไม่เกิน 255 อักขระ

ซึ่งในเดลไฟได้ให้คำแนะนำไว้ว่าในโปรแกรมใหม่ควรใช้สตริงอย่างยาว และใช้ได้เช่นเดียวกับที่เคยใช้ เพียงแต่ไม่ต้องกำหนดความยาวของสตริง

3. Variant type เป็นแบบข้อมูลที่ไม่กำหนดแบบ อาจให้ค่าเป็นเลขจำนวนหรือสตริง หรือวันเวลาได้

4. Currency type เป็นแบบข้อมูลสำหรับการเก็บเลขจำนวนจริงที่มีทศนิยมน้อยตำแหน่งใช้หน่วยความจำ 8 ไบต์ เก็บค่าเป็นเลขจำนวนเต็ม โดยแบ่งเป็นเลขทศนิยม 4 ตำแหน่ง จึงเก็บค่าได้ประมาณ 900 ล้านล้าน เช่นบาท

เมนู

ในเดลไฟ 2.0 ได้ปรับปรุงเมนูใหม่ ที่เปลี่ยนแปลงและเพิ่มขึ้นตามที่ใช้ในหนังสือนี้คือ

1. File

- 1.1 New ตามที่จะกล่าวในหัวข้อ “สืบสกุลจาก TForm”
- 1.2 New Application เพื่อการเปิดโปรเจกต์ใหม่ คือ New Project เดิม
- 1.3 Open เพื่อเปิดโปรเจกต์เก่าคือ Open Project เดิมและใช้เปิดไฟล์ทุกชนิด
- 1.4 Reopen เพื่อใช้ในการเปิดโปรเจกต์หรือยูนิตเก่าที่เพิ่งปิด
- 1.5 Save As เพื่อการเก็บยูนิตเป็นไฟล์ .pas คือ Save File เดิม

2. Project เป็นรายการใหม่ คือ

- 2.1 Add to Project เพื่อผนวกยูนิตเข้ากับโปรเจกต์ คือ View/Project Manager
เลือกปุ่ม Add unit ทั้งของเดิมและใหม่
- 2.2 Remove from Project เพื่อถอนยูนิตออกจากโปรเจกต์ ใช้ Project Manager
เช่นกันทั้งของเดิมและใหม่
- 2.3 Add to Repository ตามที่จะกล่าวในหัวข้อ “การสืบสกุลของหน้าต่าง”

3. Tools

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 Options เพื่อการเซตค่าต่าง ๆ คือ Options/Environment เดิม

3.2 Repository เพื่อการเพิ่มเติมแบบตามที่จะกล่าวในหัวข้อ “การสืบสกุล หน้าต่าง”

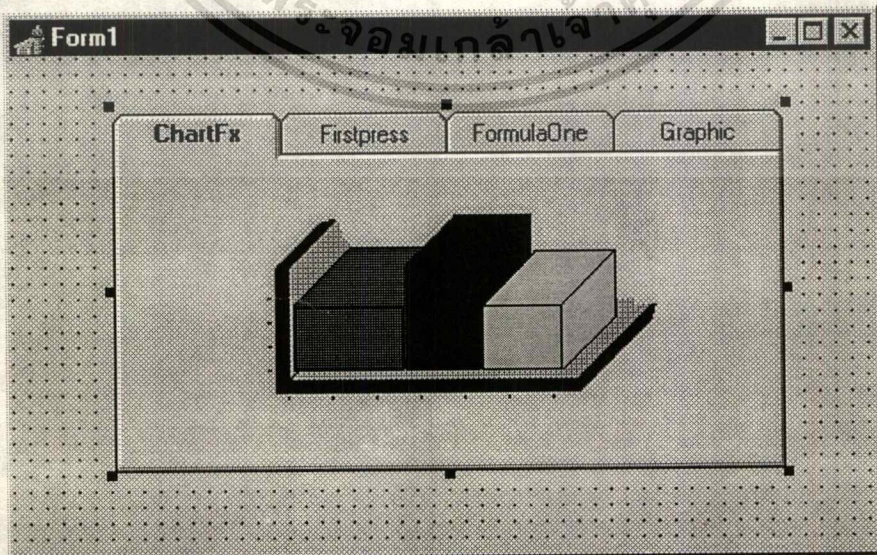
เทียบโปรแกรม

จุดประสงค์ของหัวข้อนี้เพื่อแสดง ทั้งการนำซอร์สโปรแกรมที่เตรียมด้วยเดฟ 1.0 มาใช้ คือมาคอมพายล์ด้วยเดฟ 2.0 และทั้งกับผู้ที่เตรียมโปรแกรมตามตัวอย่างในหนังสือนี้ด้วย เดฟ 2.0 โดยตรง โดยที่ในเดฟได้ให้คำแนะนำไว้ว่า

1. ในโปรแกรมจะต้องไม่ใช่ VBX เพราะใช้กับวินโดวส์ 95/เอ็นที ไม่ได้
2. หากเรียกใช้ฟังก์ชันของวินโดวส์ 3..X ที่ไม่มีในวินโดวส์ 95/เอ็นที จะคอมพายล์ไม่ผ่าน
3. ในเดฟ 2.0 ได้รวมยูนิท WinType และ WinProc ไว้ในยูนิท Windows และทำให้สามารถอ้างถึงตามเดิมได้ (แต่ถ้าป้อนโปรแกรมใหม่ควรใช้ Windows)
4. เมื่อนำโปรแกรมเก่ามาคอมพายล์ด้วยเดฟ 2.0 จะทำให้ไฟล์ .res ต่าง ๆ เปลี่ยนเป็น 32 บิต จะนำกลับไปคอมพายล์ด้วยเดฟ 1.0 ใหม่ไม่ได้

คอมโปเนนต์ใหม่

ในเดฟ 2.0 ได้เพิ่มคอมโปเนนต์ในงานทั่วไป (คือวันทางด้านฐานข้อมูล) ไว้ในหน้า Win 95 รวม 12 ชนิด เป็นคอนโทรลทั้งหมด เพื่อใช้แทนคอมโปเนนต์เดิมและเพิ่มใหม่ เพื่อให้เป็นคอมโทรลตามรูปแบบของวินโดวส์ 95 และได้นำคอมโปเนนต์เดิมที่มีของ ใหม่ใช้ แทน ไปไว้ใน หน้า Win 3.1 คอมโปเนนต์ที่เพิ่มใหม่ได้แก่

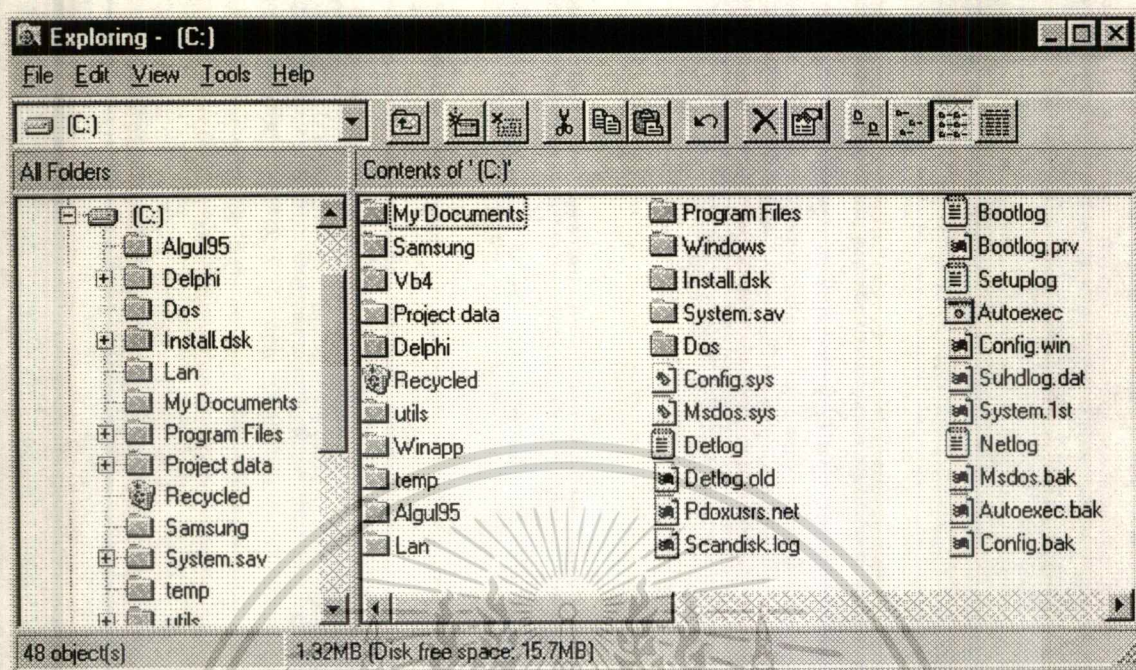


รูป 1 ภาพใบหน้าของ ChartFX ของ pageControl

TabControl เพื่อใช้แทน Tabset และมีวิธีใช้เช่นเดียวกัน (โปรแกรม Calenda)

PageControl เพื่อใช้แทน TabbedNote Boose และมีวิธีใช้คล้ายกัน แต่ทำให้กำหนดหน้าได้ง่ายกว่า33คือ อาจกำหนดหน้าทั้งหมดก่อนแล้วถึงกำหนดคอมโปเนนต์ในแต่ละหน้า หรือกำหนดหน้าและกำหนดคอมโปเนนต์ในแต่ละหน้า ได้กำหนดหน้าไว้ 4 หน้า จะเตรียมโปรแกรมที่หน้านี้ได้ดังนี้

1. กำหนดคอมโปเนนต์เป็น PageContol 1
2. กำหนดหน้าและคอมโปเนนต์ได้โดย
 - คลิกเมาส์ขวามือ PageControl 1 จะแสดงเมนูลอย
 - ให้เลือกรายการ NewPage จะเพิ่มหน้าให้
 - ให้ป้อนชื่อออปเจ็คต์เป็น ChartFx จะเป็นการให้ค่าแก่ Name ซึ่งเมื่อกดคีย์ Enter จะได้เป็นค่า Caption ด้วย
 - กำหนดคอมโปเนนต์เป็น ChartFX1 (อยู่ในหน้า OCX ซึ่งเป็นแต่เพียงให้มีการกำหนดคอมโปเนนต์ และแนะนำคอมโปเนนต์ในหน้า OCX เท่านั้น)
3. ทำซ้ำข้อ 2 โดยให้ชื่อหน้าเป็น Firstmpress, FormulaOne และ Graphic สำหรับการกำหนดคอมโปเนนต์ให้กำหนดเป็น VCFirst Impression 1, VCFormulaOne 1 และ GraphicsServer 1 ตามลำดับ
4. หากต้องการเปลี่ยนแปลงหรือกำหนดคอมโปเนนต์เพิ่มในหน้าใดให้เลือกหน้านั้นได้โดยตรง เมื่อที่เรารันคองเลือกหน้าจากที่คั่นหน้าได้เช่นกัน
 - TreeView
 - ListView
 - ImageList



รูป 2 หน้าต่างของโปรแกรม Explorer

ทั้งสามคอมพิวเตอร์นี้ได้เตรียมไว้ให้ใช้งานร่วมกัน ดังเช่นในโปรแกรม Explorer ตามรูปข้างบน ได้แบ่งหน้าต่างออกเป็น 2 บาน บานซ้ายคือ Tree pane แสดงรายการให้เลือก โดยที่

- บางรายการมีปุ่มเครื่องหมายบวก เมื่อเลือกจะแสดงรายการรอง และเปลี่ยนเครื่องหมายเป็นลบ
- แต่ละรายการจะมีไอคอนกำกับอยู่ เพื่อบ่งบอกชนิดหรือประเภทของรายการ แต่ถ้าเป็นรูปแฟ้มปิดเมื่อเลือกจะแสดงรายการรองในบาน Contents pane ทางด้านขวา จะแสดงโดยมีไอคอนกำกับเช่นกัน และให้เลือกแสดงได้ 4 รูปแบบ โดยเลือกรายการใน View คือ

- Large Icon แสดงรายการจากซ้ายไปขวาและจากบนลงล่าง โดยแสดงภาพไอคอนขนาดใหญ่

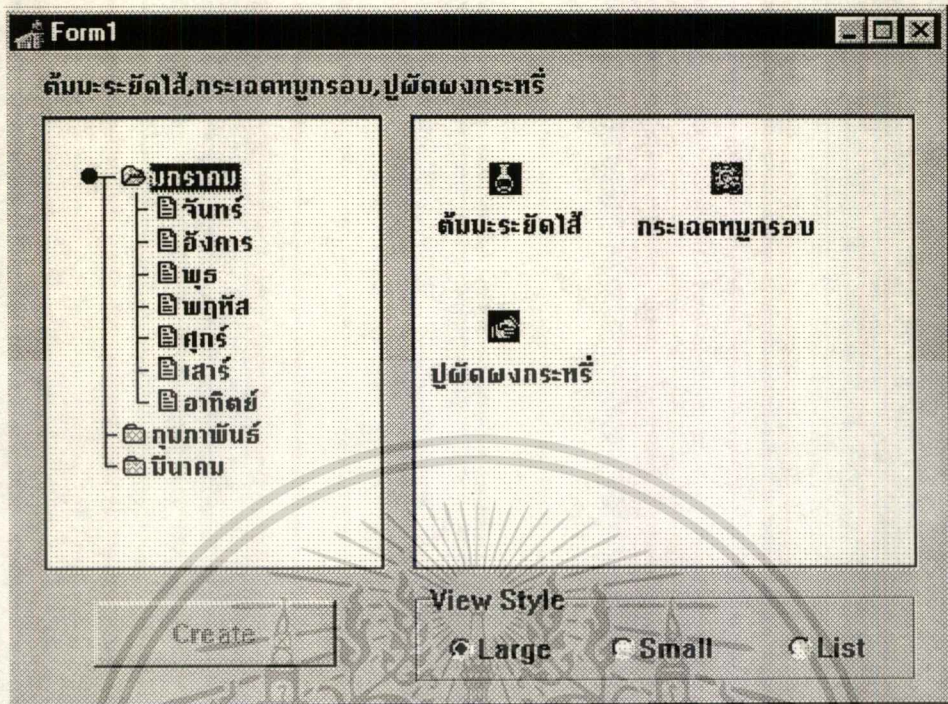
- Small Icon เช่นกันแต่กำกับด้วยภาพไอคอนขนาดเล็ก

- List เรียงลำดับจากบนลงล่างด้วยไอคอนขนาดเล็ก

- Details เช่นเดียวกับ List แต่จะให้รายละเอียด เช่น ขนาดของไฟล์

ในแคลิฟจึงจัดคอมพิวเตอร์นี้ให้ เพื่อใช้ TreeView เป็น Tree pane และ ListView เป็น Contents pane ส่วน ImageList เพื่อเก็บภาพเป็นอะเรย์ของรูปภาพ (ปิดแม่ฟไอคอน หรือ เมตาไฟล์) เพื่อแสดงเป็นไอคอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3 แสดงการใช้ TreeView กรอบซ้าย ListView กรอบทางขวาและ ImageList ซึ่งไม่แสดงตัวขณะรัน

ดังจะแสดงตัวอย่างโดยสังเขป ซึ่งเมื่อเริ่มต้นจะแสดงรายการเป็นชื่อเดือน (3เดือน) ต่อเมื่อเลือกปุ่มบวกของรายการ “มกราคม” จะแสดงรายการวันและเมื่อเลือกตรงไอคอน ของ รายการวัน เช่น ตามรูปได้เลือกของวัน “อาทิตย์” จะแสดงรายการอาหารในกรอบ ListView 1 ทางขวา

ดังจะกล่าวถึงการเตรียมโปรแกรมควบคุมไปกับคำอธิบายถึงคอมโปเนนต์ทั้งสามนี้ ดังนี้คือ

1. การใช้ ImageList
2. การใช้ TreeView
3. การใช้ ListView

การใช้ ImageList

เป็นคอมโปเนนต์ที่ไม่ใช่คอนโทรล จึงไม่แสดงตัวในขณะรัน ในการใช้งานจะต้องโหลดภาพ (บิตแมป ไอคอน หรือเมตาไฟล์) จำนวนหนึ่งให้แก่ ImageList ซึ่งอาจทำได้ทั้งขณะเตรียมโปรแกรมและในขณะรันและภาพที่จะให้อาจให้เป็นภาพเล็กหลายภาพหรือให้ภาพที่มีลักษณะเป็นแถบยาวภาพเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดั่งจะแสดงตัวอย่างทุกวิธี ขอให้กำหนดคอมโปเนนต์เป็น ImageList รวม 3 ตัวได้เป็น imglTree, imglSmall และ imglLarg3e (ให้ค่าแก่ Name) แล้วดำเนินการดังนี้

1. ของ imglTree .ซึ่งจะให้เก็บภาพไอคอนของรายการใน treView 1 โดยให้ค่าในโปรแกรม จึงต้องเตรียมภาพบิตแม็พเก็บไว้ในรีซอร์สคือในไฟล์ view1.res โดยใช้ Image Editor คือ

- เลือกรายการ Tools/Image Editor
- เลือกรายการ File/New/Resource File
- เลือกรายการ Resource/New/Bitmap

ให้ค่า Width เป็น $6 \times 16 = 96$ และค่า Height เป็น 16 จะได้ชื่อบิตแม็พรีซอร์สเป็น Bitmap 1 ให้ป้อนแก่เป็น SPOTSSIMG

ดับเบิลคลิกที่ SPOTSIMG จะแสดงหน้าต่างสำหรับวาดภาพซึ่งอาจไม่เห็นต้องขยายหน้าต่าง Image Editor

-เลือกรายการ View/Zoom In .เพื่อขยายพื้นที่วาดภาพ ซึ่งเตรียมไว้สำหรับการวาดภาพ 6 ภาพ จึงขอให้กะแบ่งพื้นที่เป็น 6 ช่องเท่า ๆ กัน และตามตัวอย่างเพื่อความง่ายจึงได้วาด เป็นรูป วงกลมสีน้ำเงิน แดง เขียว เหลือง ขาวและดำรวม 6 รูป

-เลือกรายการ File/Save As เก็บไว้ในชื่อ view.res ในไดเรกทอรีเดียวกับที่จะเก็บ

โปรแกรม
ลิสติงที่ 1

```

implementation
{ $R *.DFM }
{ $R view1.res }
procedure TForm1.FormCreate ( Sender : TObject );
begin
    oMnuList := TStringList.Create ;
    imglTree.ResourceLoad(rtBitmap, 'SPOTSIMG',c10live);  ต่อหน้าถัดไป
end;
procedure TForm1.FormDestroy ( Sender : TObject );
begin
    oMnuList.Free;
end;

```

ซึ่งจะโหลดมาให้แก่ imglTree ได้ด้วย ResourceLoad ดังในลิสติงที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ของ `imglSmall` ซึ่งจะให้เก็บภาพบิตแม็พขนาดเล็ก เพื่อใช้เป็นไอคอนของรายการใน `LisView 1` โดยให้ค่าในขณะเตรียมโปรแกรม ดังนี้คือ

- คลิกเมาส์ขวาที่ `imglSmall` เพื่อเปิดเมนูลอย แล้วเลือกรายการ `ImaglList Editor`
- จะแสดงไดอะล็อกบ็อกซ์ให้เลือกภาพ โดยเลือกรายการ `Add` ขอให้เลือกภาพขนาด `16 x 16` ซึ่งอาจวาดไว้ก่อน หรือ ตามตัวอย่างนี้ได้เลือกภาพในไดเรคทอรี

`\program File\Borland\delphi 2.0\Images\Icons\`

โดยเลือกไฟล์ที่ลงท้ายด้วยเลข `16` ทั้งหมด โดย `Add` ครั้งละ `1` ไฟล์

3. ของ `imglLarge` ซึ่งจะให้เก็บภาพบิตแม็พขนาด `32 x 32` ดังนั้นก่อน โหลดภาพขอให้ป้อนแก้ค่าพารามิเตอร์ `Heith` และ `Width` เป็น `32` คู่เสร็จแล้ว ให้โหลดภาพที่เหลือในไดเรคทอรีข้างบนให้ โดย `Add` ครั้งละ `1` ไฟล์

การใช้ TreeView

`TreeView` ก็คือ `Outline` (โปรแกรม `BrowseINI` ในบทที่ 4) ของวินโดวส์ 95 สิ่งที่เพิ่มมาคือให้เรากำหนดภาพที่จะใช้เป็นไอคอนได้ วิธีใช้จึงใกล้เคียงกันซึ่งสามารถให้ค่าได้ทั้งในขณะเตรียมโปรแกรมและขณะรัน เป็นการให้เป็นค่ารายการหรือคือเป็น โหนดหนึ่ง ๆ

โดยที่ใน `TreeView` จะมี `ItemS` เป็นค่าพารามิเตอร์ดีส์ของสกูล `TTreeNodeS` ซึ่งเป็นอะเรย์ของ `TTreeNode` ซึ่งก็คือ โหนดหนึ่ง ๆ โดยที่ในแต่ละโหนดนี้จะมี `ItemS` อีก ทำให้รายการหนึ่ง ๆ สามารถมีรายการรองและมีรายการรอง ๆ ต่อไปอีกได้ ตามที่จะกล่าวต่อไป คือ

หลังจากที่กำหนดคอมโปเนนต์เป็น `TreeView` แล้ว ให้เลือกหัวข้อ `ImglTree` และให้ค่ารายการได้ดังนี้

1. ในขณะเตรียมโปรแกรม (ตามที่จะยกมาเป็นตัวอย่างไม่ได้ใช้วิธีนี้) ให้เลือกหัวข้อ `ItemS` แล้วเลือกปุ่ม `อิลิป` จะแสดงไดอะล็อกบ็อกซ์ให้ป้อนชื่อรายการ โดย

- เลือกปุ่ม `New Item` เพื่อป้อนหัวข้อใหญ่
- เลือกหัวข้อใหญ่ (ใดก็ได้หากมีมาก) แล้วเลือกปุ่ม `New Item` เพื่อป้อนหัวข้อใหญ่ในระดับเดียวกัน

- เลือกหัวข้อใหญ่ แล้วเลือกปุ่ม `New SubItem` เพื่อหัวข้อรองของหัวข้อใหญ่นั้น

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตเห็นชอบให้ใช้เป็นการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ให้หมายเลขภาพบิดแม่พินใน Image List นับจากศูนย์ แก่ Image Index ให้เป็นภาพที่แสดงว่าไม่ถูกเลือก และแก่ Selected Index เพื่อแสดงว่าถูกเลือก

ซึ่งเมื่อเลือกปุ่ม OK จะแสดงรายการหัวข้อที่ป้อนพร้อมด้วยไอคอน (หากโหลดให้แก่ imglTree แล้ว) ในกรอบ TreeView

2. ในขณะที่การใช้งาน TreeView เมื่อไม่ได้ให้แสดงข้อมูลในระบบของวินโดวส์ (ดังเช่นของโปรแกรม Explour) ควรเป็นรายการที่ให้ผู้ใช้งานเปลี่ยนแปลงได้ดังในรูป 3 ซึ่งอาจให้ผู้ใช้งานป้อนเป็นเท็กซ์ไฟล์ ตามที่กล่าวในหัวข้อ “กรอบ Outline” แต่อาจไม่สะดวกแก่ผู้ใช้

จึงกำหนดให้ป้อนดังในลิสต์ดังที่ 2 โดยมี

- ชื่อเดือน เป็นหัวข้อใหญ่ ให้ป้อนขีดซ้าย
- ชื่อวัน เป็นหัวข้อรอง ให้กดคีย์ space bar หนึ่งครั้งแล้วป้อน
- ชื่ออาหาร ซึ่งควรเป็นหัวข้อรองของวัน คือกดคีย์ space bar สองครั้งแล้วป้อนบรรทัดละ

1 ชื่อ แต่เห็นว่าไม่สะดวกจึงจัดให้เป็นข้อมูล โดยให้กดคีย์ Tab หนึ่งครั้งแล้วป้อน บรรทัดใหม่ด้วยเช่นของวันอาทิตย์ในการป้อนชื่ออาหารให้แยกจากกันด้วยเครื่องหมาย , และวรรค

ขอให้ป้อนข้อความตามลิสต์ 2 เก็บไว้ในไฟล์ foodmenu.txt ในไดเรกทอรีเดียวกับ

โปรแกรม

ลิสต์ที่ 2

มกราคม	
จันทร์	ต้มยำกุ้ง , คะนํานํ้ามันหอย , ปลาทอด
อังคาร	แกงเขียวหวานไก่ , กุ้งชุบแป้งทอด , ผัดบุ้งไฟแดง
พุธ	ไข่ชดใส่ , แกงจืดเต้าหู้ , ผัดพริกหยวก
พฤหัสบดี	ผัดเนื้อนํ้ามันหอย , แกงเลียง , ข้าปลากระป๋อง
ศุกร์	ปลาช่อนแปะชะ , แกงป่า , ผัดถั่วงอก
เสาร์	แกงแดงเนื้อ , ไข่ลูกเขย , ผัดท๊อกรมมิตร
อาทิตย์	คัมมะระขี้ชู , กระเจตหมูกรอบ , ปูผัดผงกระหรี่ , ข้าวเหนียว , ไก่ย่าง , ส้มตำ , น้ำคอก
กุมภาพันธ์	
มีนาคม	

ลิสตริงที่ 3

```

const  chTab = #9;      chspec = #32;   cdCmma = ', ';
function  GetToken ( var txt : String ) : String ;
var      ic : integer ;
begin
    ic := Pos ( chTab , txt ) ;
    if ic = 0 then ic := Pos ( chCmma . txt ) ;
    if ic > 0 then Result := Copy(txt , 1 , ic-1 )
else
begin
    if Lengh (txt) > 0 then
    begin Result := txt ; txt := '' end
    else Result := ''
end;
    if txt <> '' then
        txt := Copy ( txt , ic + 1 , Length (txt) ) ;
    end ;
    procedure TForm1.button1click ( Sender : TObject ) ;
    var      oList : TStringList ; itm , ind : integer ;
            txt , adTxt : String ; oNde , oSub : TTreeNode ;
begin
    TreeView1.ItemS.Clear; oList := TStringList.Create;
    oList.LoadFromFile('foodmenu.txt');
    itm := 0 ;      oMnuList.Clear;
    while itm < oList.Count do
        begin
            txt := oList[itm];   inc(itm);
            if txt[1] > chSpec then
                begin
                    oNde := TreeView1.ItemS.Add(TreeView1.Selected, txt);
                    oNde.ImageIndex := 0 ; oNde.SelectedIndex := 0 ;

```

```

end
else if txt[1] = chSpce Then
begin
oSub.data := Pointer (oMnuList.count);
oSub.ImageIndex := 2 ; oSub.SelectedIndex := 3 ;
end
else
begin
txt[1] := chSpce ; ind := oMnuList.Count-1 ;
oMnuList.Deledte(ind) : oMnuList.add( adtxt);
end;
end;
oList.Free ; Button1.Ebabled := False ;
end;

```

จะให้ค่าแก่ TreeView 1 ได้ตั้งโปรแกรมของกิจกรรม Button 1 Click ในลิสต์ดิงที่ 3 คือ เมื่อผู้ใช้เลือกปุ่ม Create คือปุ่ม Button1 ซึ่งได้ให้ดำเนินการดังนี้คือ

2.1 ให้ oList โหลดข้อมูลจากไฟล์ foodmenu.txt มาเก็บไว้เป็นรายการสตริง สตริงละ 1

บรรทัด

2.2 ในวงรอบ while ได้ให้อ่านสตริงจาก oList มาครั้งละ 1 สตริง

2.3 ซึ่งถ้าอยู่ต้นบรรทัด คือชื่อเดือน จะให้เพิ่มโหนดเป็นหัวข้อใหญ่ด้วย Add โดยที่ Add

จะ

- ให้กำเนิดโหนด (โหนดเป็นออปเจกต์) ดังที่ให้ค่าแก่ oNde
- และผนวกต่อท้ายโหนดในระดับเดียวกัน ตามค่าโหนดที่ผ่านไปให้ตรง Selected
- ซึ่งในครั้งแรกนี้ Selected จะเป็น nil จึงผนวกเป็นโหนดหัวข้อใหญ่ หัวข้อแรกคือ

“มกราคม”

- ในครั้งต่อไป Selected จะเป็นโหนดของเดือน “มกราคม” จึงเป็นการบวกโหนดของเดือน “กุมภาพันธ์” ต่อท้ายเดือน “มกราคม”

- ซึ่งในการให้กำเนิดจะให้ค่า txt เป็นค่า Text ของโหนดและนำออกแสดง
- ได้ให้ค่า ImageIndex และ Selected Index เป็นศูนย์ (วงกลมสีน้ำเงิน) ทั้งคู่ เพื่อจะ

บอกผู้ใช้เป็นหัวข้อที่ไม่มีข้อมูลภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 แต่ถ้าอักขระแรกเป็นวรรค คือชื่อวัน จะให้ AddChild ผนวกให้เป็นโหนดลูก เช่น “วันจันทร์” ของโหนดแม่ เช่น ของเดือน “มกราคม” (ตามตัวอย่างนี้มีแค่เดือนนี้) โดย

- ให้ GetTokm ตัดคำเฉพาะที่เป็นชื่อวันมาให้เป็นค่า Text
- ค่าที่เหลือใน txt จะเป็นรายชื่ออาหาร ได้ให้เก็บไว้เป็นรายการสตริงใน oMmuList
- และให้ค่าอินเด็กซ์บวกหนึ่ง ของรายการนี้แก่ Data ซึ่งให้เราเก็บค่าใดก็ได้ตามต้องการแต่ต้องเก็บเป็น Pointer (ที่ให้เก็บเป็นอินเด็กซ์บวกหนึ่ง เพราะไม่ต้องการให้มีค่าศูนย์ ซึ่งจะตรงกับ nil)

2.5 ถ้าอักขระแรกเป็น Tab เช่น กรณีของวันอาทิตย์ ให้เก็บต่อท้ายสตริงเดิม

การใช้ ListView

มีลักษณะการใช้ใกล้เคียงกับ TreeView แต่จะต้องใช้ ImageList สองตัวขอให้กำหนดคอมโปเนนต์เป็น ListView1 และให้ค่า

- LargeImage เป็น imglLarge
- SmallImages เป็น imglSmall

กับกำหนดคอมโปเนนต์เป็น RedioGroup1 โดยให้ชื่อปุ่มวิทยุเป็น Large, Small, List และ

Report

จะให้ค่าแก่ ListView1 เพื่อให้แสดงรายการดังในรูปที่ 3 กรอบขวา ได้ตั้งโปรแกรมของกิจกรรม TreeView1MouseDown ในลิสติง 4 ซึ่งจะมาทำงานที่กิจกรรมนี้เมื่อผู้ใช้เลือกรายการใน TreeView ดังที่ให้คำเนนการดังนี้

ลิสตริงที่ 4

```

procedure TForm1.TreeView1MouseDown(Sender : TObject ; Button :
TMouseButton ;
    Shift : TShifState ; X,Y : Integer);
var HT : Thitests ; oNde : TTreeNode ; txt ,adTxt : String ;
    oItem : TListItem ; cnt : Integer ;
begin
    HT := Treeview1.GetHitTestInfoAt (X,Y) ;
    if not (htOnItem in HT) then Exit ;
    Label1,caption := omnuList [Integer(oNde.Data) - 1];
    txt := oMnuList [Integer(oNde,data) - 1) ;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และสงวนเพื่อวัตถุประสงค์เท่านั้น ไม่อนุญาตให้มีการนำเอกสารไปใช้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ListView1.ItemS.Clear ;
While (txt <> '') do
begin
adtxt := GetToken (txt) ;
oitem := ListView1.ItemS.Add ;
oem.SubItems.add ('10 ');
oitem.SubItems.add ('20 ');
oItem.SubItems.add ('30');
end ;
end;
procedure TForm1.RadioGroup1Click (Sender : TObject);
begin
ListView.viewStyle := Tviewsty(RadioGroup1 > Itemindex );
end;
end.

```

1. ให้ตรวจว่าถ้าผู้ใช้ไม่ได้เลือกตรงไอคอนหรือชื่อ (htOnItem) ไม่ดำเนินการต่อ
2. หาโหนด ณ ตำแหน่งที่ผู้ใช้คลิกแล้วใส่ให้แก่ oNde
3. ถ้าเป็นโหนดที่ไม่ได้ให้ค่าข้อมูลไว้แก่ Data ไม่ดำเนินการต่อ
4. นำค่าใน Data มาทำเป็นอินเด็กซ์โดยลบด้วยหนึ่ง
5. แล้วอ่านสตริงของรายการนี้ มาให้เป็นค่า Caption ของกรอบ Label 1 เพื่อเป็น ตัวอย่างเมื่อไม่ต้องการใช้ LisView และให้ค่าแก่ txt
6. นำค่าใน txt มาทำเป็นโหนดของ LisView โดย
 - ให้ GetToken แยกเป็นชื่ออาหารให้ครั้งละ 1 ชื่อให้แก่ adTxt
 - ให้ Add ให้กำเนิดโหนดและผนวก เป็นรายการหลัก (เท่านั้น)
 - ให้ชื่ออาหารเป็นค่า Caption เพื่อนำออกแสดงและให้หมายเลขไอคอนแก่ ImageIndex (จะได้ผลดังในรูปที่ 3)
7. โดยที่แต่ละโหนดหลักของ LisView จะมีโหนดลูกได้ 1 ชั้น เพื่อแสดงผลเป็นตาราง ดังในรูปที่ ค-4 ซึ่งจะต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

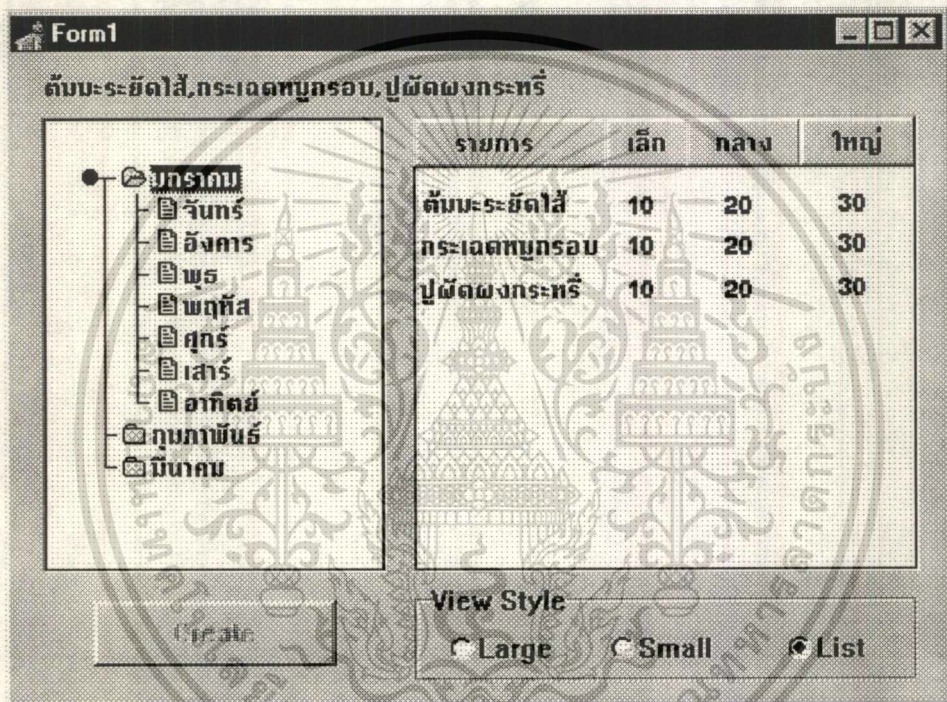
- ให้กำหนดไอคอนลูกและให้ค่าด้วย SubItemS.Add ดังที่ได้แสดงการให้ค่าไว้พอเป็นตัวอย่าง

- ให้ชื่อคอลัมน์แก่ Columns ในขณะเตรียมโปรแกรม ซึ่งตามรูป 4 ให้ไว้เป็นรายการ เด็กกลาง ใหญ่ รวม 4 คอลัมน์ (ถ้าไม่ให้จะไม่แสดงภาพ)

- เลือกรูปแบบของการแสดงคือ ViewStyle เป็น vsReport

สามารถเลือกรูปแบบของการแสดงได้ 4 แบบ ตามที่กล่าวมา ดังที่ได้จัดกรอบ RadioGroup

1 ไว้ และมีโปรแกรมอยู่ในกิจกรรม RadioGroup 1 Click



รูป 4 การแสดงผลในกรอบ ListView ในรูปของตาราง เมื่อเลือกปุ่ม Report

RichEdit

RichEdit เป็นเอดิเตอร์เช่นเดียวกับ Memo แต่มีขีดความสามารถสูงกว่าคือ ให้ผู้ใช้สามารถกำหนดรายละเอียดของฟอนต์ให้แก่อักษรเป็นตัว ๆ ได้ เช่น ทำเป็นหัวข้อ และรายละเอียดเกี่ยวกับการย่อหน้า ดังจะนำไปโปรแกรมเก่ามาเปลี่ยนเป็นใช้ RichEdit กล่าวคือ

ให้สร้างไครเรคทอรีใหม่แล้วลอกไฟล์ทุกไฟล์ของโปรแกรม TxtEdit มาไว้ในไครเรคทอรีใหม่นี้ (อาจเว้นไฟล์ที่เป็น backup) แล้วดำเนินการดังนี้คือ

1. ขุดกรอบ Memo 1 (เลือกแล้วกดคีย์ Del)
2. กำหนดคอมโปเนนต์เป็น RichEdit 1 แล้วให้ค่าพรอพเพอร์ตี้เช่นเดียวกับของ Memo 1

คือ

- Align ให้เป็น alClient

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- HideSelection ให้เป็น False

- ScrollBars ให้เป็น sBoth

3. เปลี่ยนค่าจาก Memo 1 เป็น RichEdit 1 ทุกแห่ง (ใช้ Search/Replace)

4. ป้อนแก้โปรแกรมของกิจกรรม SrchMemo ให้เป็นดังในลิสตติง 5 ทั้งนี้เพราะใน RichEdit มีกิจกรรม FindText ค้นหาคำอยู่แล้ว และต้องเปลี่ยนการให้ค่าแฟลกใน Options ของ FindDialog 1 ใหม่คือ frDisableMatchCase และ frDisableWholeWord ให้เป็น False ทั้งคู่ เพื่อให้ผู้ใช้เลือกได้

ลิสตติงที่ 5 RichEdit1.Paragraph.

```
function Tfrom1.SrchMemo(const srchTextS : OpenString) : Boolean ;
var fndAt , srchStt , srchLen : Integer ; opt ; Tsearchtypes ;
begin
    srchStt := RichEdit1.SelStart + Lengthz( srchTexts);
    rchLen := RichEdit1.SelLength ;
    if SerchLen <= Length ( SerchTexts ) Then
        SerchLen := RichEdit .GetText(Lend ;
    Opt := [] ;
with FinDislog1 do
begin if FrWholeword in Option Then
    opt := Opt + [stWholeWord] ;
if frmathCase in Options Then
    Opt := Opt + [ stMatchCase] end ;
fndAt := RichEdit1.FindTexts( srchstt , srchLen , opt) ;    ค่อนหน้าถัดไป
If fndAt >= 0 Then
begin RichEdit1.SetFosucs ;
    Ricshedit .SelSart := fndAt ;
    RichEdit1.SelLength := Llength ( srchTextS);
    lblTopMag.Caption := 'search Ffond ' ;
    lblTopMag.Font.Color := clBlue ;
    SrchMemo := True    end
else
begin lblTopMsg.captio := 'Search "' + SrchTxs + '" NOT FOND ' ;
    lblTopmsg.Font.Colo := clRed ;
    srchMemo := False end
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษานานาชาติ ในอนาคตหากมีข้อผิดพลาดประการใด

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end ;

(*-----NEW IN DELPHI 2.0 ----- *)

procedure TForm1.SpeedButton1Click(Sender: TObject);
begin
  if FontDialog1.Execute Then
  With RichEdit1.SelAttributes do
    begin
      Color := FontDialog1.Font.Color;
      Hight := FontDialog1.Font.Hight;
      Name := FontDialog1.Font.Name;
      Pitch := FontDialog1.Font.Pitch;
      Size := FontDialog1.Font.Size;
      Style := FontDialog1.Font.Style end;
end;

Procedure TForm1.SpeedButton2Click( Sender : TObject);
begin
  RichEdit1.SelAttributes.Color := clRed;
  RichEdit1.SelAttributes.Name := 'Times New Roman';
  RichEdit1.SelAttributes.Hight := 16;
  RichEdit1.SelAttributes.Style := [ fsBold, fsItalic ];
  RichEdit1.Paragraph.Alignment := { TaCenter } taLeftJustify;
  RichEdit1.Paragraph.FirstIndent := 120;
  RichEdit1.Paragraph.LeftIndent := 100;
  { RichEdit1.Paragraph.RightIndent := 100; }

end;

```

5. กำหนดคอมโปเนนต์เป็น SpeedBulton 1 และ SpeedBulton 2 แล้วป้อนโปรแกรมตาม
 ลิสต์คิง ค-7 เพื่อให้ผู้ใช้เลือกข้อความแล้วเลือกปุ่มทั้งสองนี้ดังนี้

5.1 ปุ่ม SpeedBulton 1 จะแสดงฟอนต์ไคอะล็อกบ็อกซ์ แล้วนำค่าที่ผู้ใช้เลือกใน
 FontDialog 1.Font มาให้แก่อข้อความที่ผู้ใช้เลือกคือ SelAltributes

5.2 ปุ่ม SpeedBulton 2 จะให้ค่าที่เตรียมไว้แก่อข้อความที่ผู้ใช้เลือก
 ทั้งนี้เพียงเพื่อเป็นตัวอย่าง ไม่ควรใช้ฟอนต์ไคอะล็อกบ็อกซ์ ควรจัดคอนโทรลให้ในแต่ละ
 เรื่อง ดูตัวอย่างได้จาก โปรแกรม WordPad (อยู่ใน Accessories) เช่น

- ฟอนต์ ใช้กรอบรายการฟอนต์

- สี ใช้กรอบรายการสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมโปเนนต์อื่น

คอมโปเนนต์ที่เหลือในหน้า Win 95 เป็นคอมโปเนนต์ที่ปรับปรุงจากของเดิม ในด้านความสวยงามบ้าง เพิ่มขีดความสามารถบ้าง มีวิธีการใช้เช่นเดียวกับหรือใกล้เคียงกับของเดิมได้แก่

1. HeaderControl ใช้แทน Header ของเดิม
2. StatusBar ปรับปรุงมาจาก Panel เพื่อใช้เป็นกรอบสแตตัส ทางด้านล่างของฟอร์ม
3. TrackBar มีลักษณะการใช้เช่นเดียวกับ ScrollBar แต่ดูสวยกว่า
4. ProgressBar ของเดิมคือ Gauge เป็นคอมโปเนนต์ตัวอย่างอยู่ในหน้า Samples
5. UpDown ของเดิมมี SpinBulton กับ SpinEdit สำหรับ UpDown หากใช้ตัวเดียวคือ

SpinBulton หากใช้ร่วม (associate) กับกรอบ Edit คือ SpinEdit

6. HotKey เป็นคอนโทรลใหม่ เพื่อให้ผู้ใช้กำหนดคีย์ด่วน (hot key) ให้แก่คอนโทรล (n)

Object Repasitory

ในเคลไฟ 2.0 ใช้ Object Repasitory แทน Gallery ในรุ่น 1.0 ทำหน้าที่เก็บต้นแบบของโปรเจกต์ฟอร์ม ฯลฯ เพื่อใช้งานร่วมกันทั้งในโปรเจกต์เดียวกันและต่างโปรเจกต์ มีต้นแบบที่สร้างไว้ก่อนแล้วและที่เราจะสร้างเพิ่มตามที่จะกล่าวในหัวข้อ “การสร้างฟอร์มต้นแบบ”

ให้นำมาใช้ได้โดยเลือกรายการ File/New จะแสดงไดอะล็อกบ็อกซ์ New Item ดังในรูปที่ ค-5 แสดงต้นแบบที่เก็บอยู่ใน ObjectRepository แบ่งเป็น 6 หน้า ภายในแต่ละหน้าจะมีต้นแบบต่างๆ ให้เลือกใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 5 ไอคอนบ็อกซ์ New Items ซึ่งแบ่งเป็น 6 หน้า ในหน้า Forms จะแสดงไอคอนของฟอร์มต้นแบบต่าง ๆ ตามรูปนี้ ให้เก็บเมื่อได้สร้างฟอร์มต้นแบบเป็น My Origin Form แล้ว

การสืบสกุลของหน้าต่าง

เมื่อเปิดฟอร์มใหม่ เช่น เมื่อเปิดโปรเจกต์ใหม่ ได้ฟอร์มเป็น Form 1 จะเท่ากับเป็นการสร้างสกุล Tform 1 ให้สืบสกุลจาก Tform ซึ่งเป็นหน้าต่าง เป็นการสืบสกุลของหน้าต่าง

แต่ถ้าต้องการจะสร้างสกุลใหม่สืบสกุลจาก Tform 1 ซึ่งเป็นการสืบสกุลอีกทอดหนึ่ง ในเคลฟ 1.0 ทำไม่ได้ มาทำได้ในเคลฟ 2.0 ดังที่จะกล่าวในหัวข้อนี้

จุดประสงค์

จุดประสงค์ของการสร้างสกุลคือ เพื่อใช้เก็บไว้ใช้ และเพื่อการสืบสกุล ส่วนจุดประสงค์ของการสืบสกุลคือ เพื่อใช้เปลี่ยนแปลงและต่อเติมโปรแกรมเดิมในโปรแกรมใหม่

บรรพบุรุษและการสืบสกุล

เมื่อสร้างสกุล AA สืบสกุลจากสกุล A จะได้ AA เป็นผู้สืบสกุล (descendent) ของสกุล A และสกุล A เป็นบรรพบุรุษ (ancestor) ของสกุล AA และอาจสร้างสกุล AB, AC..สืบสกุลจากสกุล A จึงได้สกุล A เป็นสกุลต้นตระกูลของสกุลเหล่านี้ เช่นกันอาจได้สกุล AA เป็นต้นตระกูลของสกุล AAA, AAB...

ฟอร์มต้นแบบ

เราอาจสร้างสกุลธรรมดาโดยทั่วไปรวมทั้งที่ให้สืบสกุลจากสกุลที่เป็นคอมโปเนนต์ไว้ในยูนิต (โดยไม่ทำสกุลใหม่นี้เป็นคอมโปเนนต์) แล้วนำยูนิตไปใช้โดยสร้างสกุลใหม่สืบสกุลจากสกุลในยูนิตได้โดยตรงไม่ต้องพึ่งเคลฟ

แต่ถ้าเป็นสกุลของฟอร์ม เช่น เมื่อเปิดโปรเจกต์ใหม่จะได้ Form 1 เป็นออกเจกต์ของ Tform 1 สืบสกุลจาก Tform ถ้าเราต้องการเก็บสกุล Tform 1 ไว้เป็นสกุลต้นตระกูล เช่น เพื่อสร้างผู้สืบสกุลเป็น TFormX มีฟอร์มเป็น FormX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากอาจต้องกำหนดคอมโปเนนต์เพิ่มในฟอร์ม FormX ในเคลฟจึงให้เราเก็บสกุลแต่ความหมายจะหนักไปทางฟอร์มคือ TFormX ซึ่งจะขอเรียกเป็นฟอร์มต้นแบบไว้ใน Object Repository ตามที่จะกล่าวในหัวข้อ “การสร้างฟอร์มต้นแบบ” และให้นำมาใช้คือนำมาสืบสกุลได้ดังที่จะกล่าวในหัวข้อ “การใช้ฟอร์มต้นแบบ”

ไคเรคทอรีของฟอร์มต้นแบบ

โดยที่ในการสร้างฟอร์มต้นแบบ จะต้องเก็บไฟล์ของฟอร์มต้นแบบ (ไฟล์ .pas .dfm และ .dcu) ไว้ในไคเรคทอรีประจำ เพื่อเก็บไฟล์ของฟอร์มต้นแบบทั้งหมดคือที่จะสร้างต่อไปในอนาคต ด้วยจึงขอให้สร้างไคเรคทอรีใหม่ เช่น

C: mydelphi \ myobjrep

และต่อไปจะขออ้างถึงไคเรคทอรีนี้ว่า myobjrep

การสร้างฟอร์มต้นแบบ

เพียงเพื่อเป็นตัวอย่างในการสืบสกุลเท่านั้น จะสร้างฟอร์มต้นแบบให้ประกอบด้วยกรอบ Edit 1 กรอบ Label 1 ปุ่ม Bultron 1 และ Bultron 2 โดยที่เมื่อผู้ใช้ป้อนข้อความในกรอบ Edit 1 แล้วเลือกปุ่ม Bultron 1 จะนำข้อความในกรอบ Edit 1 ไปแสดงในกรอบ Label 1 และถ้าเลือกปุ่ม Bultron 2 จะแสดงเมสเสจบ็อกซ์ จะดำเนินการได้ดังนี้คือ

1. เปิดโปรเจกต์ใหม่ (ยังไม่ต้องเก็บลงไฟล์)
2. เปลี่ยนชื่อ Form 1 เป็น OrgForm โดยให้ค่าแก่ Name ทั้งนี้เพื่อไม่ให้ขัดต่อการตั้งชื่อของเคลฟ และลบข้อความที่ Caption เพื่อให้เป็นฟอร์มที่ไม่มีชื่อที่ไคเดิลบาร์
3. กำหนดคอมโปเนนต์เป็น Edit1, Label 1, Bultron 1 และ Bultron 2 ณ ตำแหน่งที่เห็นเหมาะสมแล้วป้อนโปรแกรมตามลิสต์ที่ 6 เสร็จแล้วให้ทดลองรัน

ลิสต์ที่ 6

```

procedure TOrgForm.Button1Click ( Sender : Tobject );
begin
    Label1.Caption := Edit.Text ;
end ;
procerurd TorgForm.Button2Click ( Sender : Tobject ) ;
begin
    ShowMessage ('Here , I Am ');
end ;

```

4. เก็บไว้ใน Object Repasitory โดย

4.1 เลือกรายการ File/ Save As เพื่อเก็บไฟล์ของฟอร์มต้นแบบไว้ในไดเรคทอรี myobjrep โดยให้ชื่อไฟล์เป็น orgformO.pas

4.2 เลือกในฟอร์มในแอกทีฟแล้วคลิกเมาส์.ขวาเพื่อเปิดเมนูลอย เลือกรายการ Add To Repository หรือเลือกการ Project/Add To Reposibory ที่เมนูบาร์ จะแสดงไดอะล็อกบ็อกซ์ ดังในรูปที่ 6 ให้ป้อนค่าต่าง ๆ ดังนี้

- Title ป้อนชื่อที่จะให้แสดงเป็นชื่อต้นแบบ เช่น My Original Form
- Description ป้อนคำอธิบาย เช่น Exanple Form
- Page ป้อนชื่อหน้าคือ Forms
- Author ป้อนชื่อผู้เขียนโปรแกรม
- หากได้เตรียมภาพไว้ในไฟล์ .ico ให้โหลดด้วยปุ่ม Browse
- เลือกปุ่ม OK

รูปที่ 6 ไอคอนบ็อกซ์ที่ใช้ในการเก็บฟอร์มต้นแบบไว้ใน Object Repository

5. ปิดโปรเจกต์โดยเลือกรายการ File/Close All หรือเปิดโปรเจกต์ใหม่และให้ปฏิเสธคำถามที่จะให้เก็บลงไฟล์ เนื่องจากจะไม่เก็บโปรเจกต์ไว้

สำหรับการรันทดสอบ ในบางกรณีอาจต้องสร้างฟอร์มต้นแบบที่ไม่สมบูรณ์ในตัวเองรันไม่ได้ จึงเพียงแต่ให้ผ่านการคอมไพล์ และโดยทั่วไปโปรแกรมประเภทนี้จะยาว จึงควรอย่างยิ่งที่จะเก็บลงไฟล์ไว้เป็นระยะ ๆ ตามปกติ เมื่อรันทดสอบหรือคอมไพล์เรียบร้อยแล้ว จึงดำเนินการตามข้อ 4 นั่นคือสามารถนำโปรแกรมที่เขียนไว้แล้วมาทำเป็นฟอร์มต้นแบบได้ทุกโปรแกรม และถ้ามีหน้าต่างหลายบานให้เลือกบานที่ต้องการ

การยกเลิกฟอร์มต้นแบบ

ฟอร์มต้นแบบที่ไม่ใช่แล้ว ให้ยกเลิกได้โดยเลือกรายการ Tools/Repository จะแสดงไดอะล็อกบ็อกซ์ ให้

- เลือกหน้าคือ Forms
- เลือกชื่อฟอร์ม เช่น My Original Form
- เลือกปุ่ม Delete Object

การใช้ฟอร์มต้นแบบ

การใช้ฟอร์มต้นแบบเป็นไปตามที่กล่าวมาคือ เพื่อใช้ เปลี่ยนแปลง และต่อเติม ใดๆ อย่างใดอย่างหนึ่งหรือร่วมกัน ซึ่งอาจจำแนกการนำมาใช้ได้เป็น 2 เรื่อง คือ

1. วิธีการนำมาใช้ ได้แก่

- Copy เป็นการลอกฟอร์มมาใช้เช่นเดียวกับที่ใช้อยู่ในเคลฟ 1.0
- Inherit เป็นการนำมาใช้ด้วยการสืบสกุล เป็นวิธีการที่ถูกต้องตามหลักของการเขียนโปรแกรมในแบบโอโอพี ที่ใช้อยู่ในเคลฟ
- Use เป็นการนำฟอร์มต้นแบบมาใช้โดยตรง การเปลี่ยนแปลงใด ๆ จึงมีผลต่อฟอร์มต้นแบบโดยตรง เป็นวิธีการที่สวนทางกับหลักการของโอโอพี แต่ก็มีประโยชน์ เช่น เมื่อโปรแกรมที่จะเตรียมให้แก่ฟอร์มต้นแบบไปพร้อม ๆ กัน

2. ใช้ในฐานะ คือจะนำมาใช้เป็นหน้าต่างหลัก หรือหน้าต่างรอง

ดังจะแสดงตัวอย่างการนำมาใช้ด้วยการสืบสกุลในฐานะหน้าต่างหลักในหัวข้อต่อไปนี้

การสืบสกุลจากฟอร์มต้นแบบ

ตามที่กล่าวว่า การสืบสกุลจากฟอร์มต้นแบบเพื่อ ใช้เปลี่ยนแปลงและต่อเติมนั้นความจริงแล้วเป็นการยากที่จะแยกเรื่องทั้งสามนี้ออกจากกัน ดังที่ในตัวอย่างนี้พยายามจะแยกเป็น

- การใช้ จะใช้ปุ่ม Bulton2 ตามที่ออกแบบไว้คือ ให้แสดงเมสเสจบ็อกซ์
 - การเปลี่ยนแปลงจะเปลี่ยนแปลงการทำงานของปุ่ม Bulton 1 ให้นำข้อความที่ผู้ใช้ป้อนในกรอบ Edit 1 ไปเก็บไว้ในกรอบรายการ ตามที่จะกำหนดเพิ่มเป็น ListBox 1
 - การต่อเติม จะเพิ่มปุ่ม Bulton 3 เพื่อลบข้อความในกรอบ LisBox 1
- จะเตรียมโปรแกรมด้วยการสืบสกุลจากฟอร์มต้นแบบ ได้ดังนี้

1. เปิดโปรเจกต์ใหม่ แต่ยังไม่ต้องเก็บลงไฟล์ จะได้ฟอร์ม Form1 ตามปกติ
2. เลือกรายการ File/New ซึ่งจะแสดงไดอะล็อกบ็อกซ์(คล้ายกับในรูป ค-5) ให้

- เลือกหน้า Forms
- เลือกฟอร์มต้นแบบเป็น My Origin Form
- เลือกปุ่ม Inherit (จะเห็นว่ามิปุ่มให้เลือก 3 ปุ่มคือ Copy, Inherit และ Use ตามที่

กล่าวมา)

- เลือกปุ่ม OK

3. จะได้ฟอร์ม (หน้าต่าง) อีก 1 ฟอร์มมีชื่อเป็น OrgForm 2 ซึ่งอาจใช้ชื่อนี้ แต่ควรเปลี่ยน

ใหม่โดยให้ค่าแก่ Name เช่น ให้เป็น NewForm1

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเชิงในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับทางด้านของโปรแกรมในหน้าต่างยูนิท จะเห็นว่าเคลฟได้สร้างสกุล TNewForm สืบสกุลจาก TorgForm (แทนเรา) เช่นเดียวกับที่เคยสร้างสกุล Tform1 สืบสกุล Tform และเป็นการ สืบสกุลโดยสมบูรณ์แล้ว และจะใช้ฟอร์มใหม่นี้ได้ดังนี้

3.1 ใช้เป็นหน้าต่างรอง ได้เช่นเดียวกับที่เปิดด้วย File/ New Form

3.2 ใช้เป็นหน้าต่างหลัก คือตามตัวอย่างนี้ซึ่งจะต้องยุบฟอร์ม Form1 โดยเลือกรายการ View/Project Manager จะแสดงหน้าต่าง Project Manager ให้

- เลือกฟอร์ม Form 1

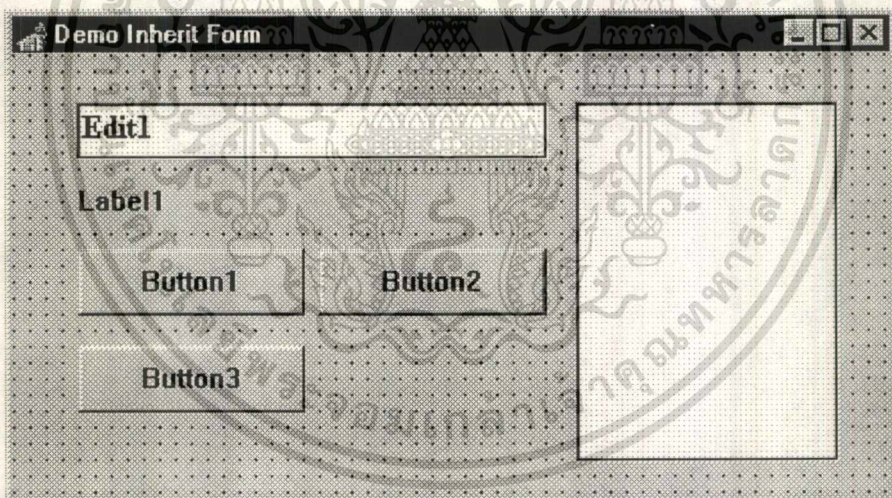
- เลือกปุ่ม Remove Unit คือปุ่มที่มีเครื่องหมายลบ (ให้ปฏิเสธการเก็บลงไฟล์)

- ฟอร์ม Form1 จะหายไปเหลือแต่ NewForm1 เป็นหน้าต่างหลัก และควรที่จะได้ เก็บลงไฟล์ไว้ครั้งหนึ่งก่อน

4. ปรับปรุงโปรแกรมโดย

- กำหนดคอมโปเนนต์เป็น ListBox 1 และ Button3 จะได้ผลดังในรูป 7

- ป้อนโปรแกรมตามลิสดิงที่ 7 ซึ่งเมื่อรันจะได้ผลตามที่กล่าวมา



รูป 7 ฟอร์มของผู้สืบสกุลหลังจากกำหนดคอมโปเนนต์เพิ่มแล้ว

ลิสดิงที่ 7

```

procedure TnewForm1.Button1Click (sender : TObject ) ;
begin
    inherited;
    ListBox1.Items.Add ( Edit1.Text);
end ;
procedure TnewForm1.Button3Click ( Sender := TObject );
begin
    inherited;

```

```

ListBox1.Clear ;
end ;

```

คำอธิบายเสริม

1. ตามที่กล่าวมานี้ถึงแม้จะยกตัวอย่างเป็นโปรแกรมขนาดเล็กมาก แต่ก็ครอบคลุมเนื้อหาของการสร้างและการใช้ฟอร์มต้นแบบทั้งหมด

2. การสร้างฟอร์มต้นแบบในงานจริงจะพบในโปรแกรมขนาดใหญ่ ที่ใช้หน้าต่างหลายบาน คือหลายฟอร์ม และในหน้าต่างเหล่านี้มีส่วนที่เหมือนกัน จึงจะนำส่วนที่เหมือนกันไปทำฟอร์มต้นแบบ

3. ในโปรแกรมนี้อาจเปลี่ยนแปลงการทำงานของปุ่ม Button 1 โดยเพิ่มกรอบรายการ ListBox1 ซึ่งถ้าต้องการตัดกรอบ Label 1 ออกจะทำได้ เพราะเคล็ฟจะไม่ให้ยุบคอมโปเนนต์ในฟอร์มต้นแบบ วิธีการคือ

- ให้กดแสดงกรอบ Label1 โดยให้ค่า Visible เป็น False ข้อเสียคือทำให้เสิร์ชอร์สไปจำนวนหนึ่งโดยไม่ได้ใช้

- หรือ เลือกวิธีการนำมาใช้เป็น Copy แต่ก็มีข้อเสียเช่นกันคือจะมีโปรแกรมเดียวกันหลายชุด

ในการสร้างฟอร์มต้นแบบจึงควรตรวจสอบที่จะไม่ให้เกิดกรณีเช่นนี้

4. เมื่อนำฟอร์มมาใช้ด้วย Use สามารถปรับปรุงโปรแกรมเดิมได้ทั้งหมด แต่จะต้องไม่เปลี่ยนค่าใน Name ของฟอร์ม

Finalization

ในเคล็ฟ 1.0 เราสามารถกำหนดส่วน initialization ไว้ตอนท้ายของยูนิท เพื่อให้รันโปรแกรมในส่วนนี้ตามลำดับของยูนิทที่บ่งชี้ใน uses ก่อนแสดงหน้าต่างหลัก

ในเคล็ฟ 2.0 ได้เพิ่มส่วน finalization อีก 1 ส่วน โดยกำหนดว่าจะต้องอยู่หลังส่วน initialization เช่น

```

initialization
ShowMessage ( 'Start ' );
finalization
MessageBeep (0)
end.

```

เอกสารนี้เป็นเอกสารที่... ให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะแสดงเมสเสจบ็อกซ์ก่อนเปิดหน้าต่างและส่งเสียงบีบหลังหน้าต่างแล้ว

เปิดโปรเจกต์ใหม่

ให้เริ่มการโปรแกรมทุกครั้งด้วยการเปิดโปรเจกต์ใหม่ โดยเลือกรายการ File/New Project หรือใช้โปรเจกต์ที่เคลฟเปิดให้เมื่อเริ่มต้นดังเช่นในรูปที่ 8 เป็นโปรเจกต์ใหม่เช่นกัน

เมื่อเปิดโปรเจกต์ใหม่เคลฟจะเตรียมหน้าต่างให้ 2 บาน คือหน้าต่างฟอร์มคือ Form 1 เพื่อให้กำหนดคอมโปเนนต์ และหน้าต่างยูนิตคือ Unit1 เพื่อใช้ป้อนโปรแกรม

หมายเหตุ

การเปิดโปรเจกต์ใหม่ตามที่กล่าวมานี้ หมายถึง ได้ติดตั้งเคลฟโดยไม่ได้เซตค่าใด หากได้เซตปุ่ม Options/Environment/Gallery/Uses on New Project ไว้ จะแสดงรูปแบบของฟอร์ม ให้เลือกเป็น Blank project

รันโปรแกรมทดสอบและยุติการรัน

สามารถรันโปรแกรมทดสอบได้ทุกโอกาสที่ต้องการ แม้แต่เมื่อเพิ่งเปิดโปรเจกต์ใหม่ โดยเลือกรายการ Run/Run หรือกดคีย์ F9 หรือเลือกปุ่ม Run ที่สปีดบาร์

เมื่อรันโปรแกรมทดสอบแล้ว สามารถยุติการรันได้โดยเลือกรายการ Close จาก System menu (คือปุ่มที่มุมบนซ้ายของหน้าต่าง หรือปุ่ม X ของวินโดวส์ 95)

ก่อนที่จะทดลองรันโปรแกรม หากเป็นโปรแกรมขนาดใหญ่ ควรได้คอมพายล์ก่อน โดยเลือกรายการ Compile/Compile แล้วเก็บลงไฟล์ตามที่กล่าวต่อไป

เมื่อรันโปรแกรมทดสอบ เคลฟจะคอมพายล์ให้ก่อนแล้วจึงรัน

โปรแกรม MyNull

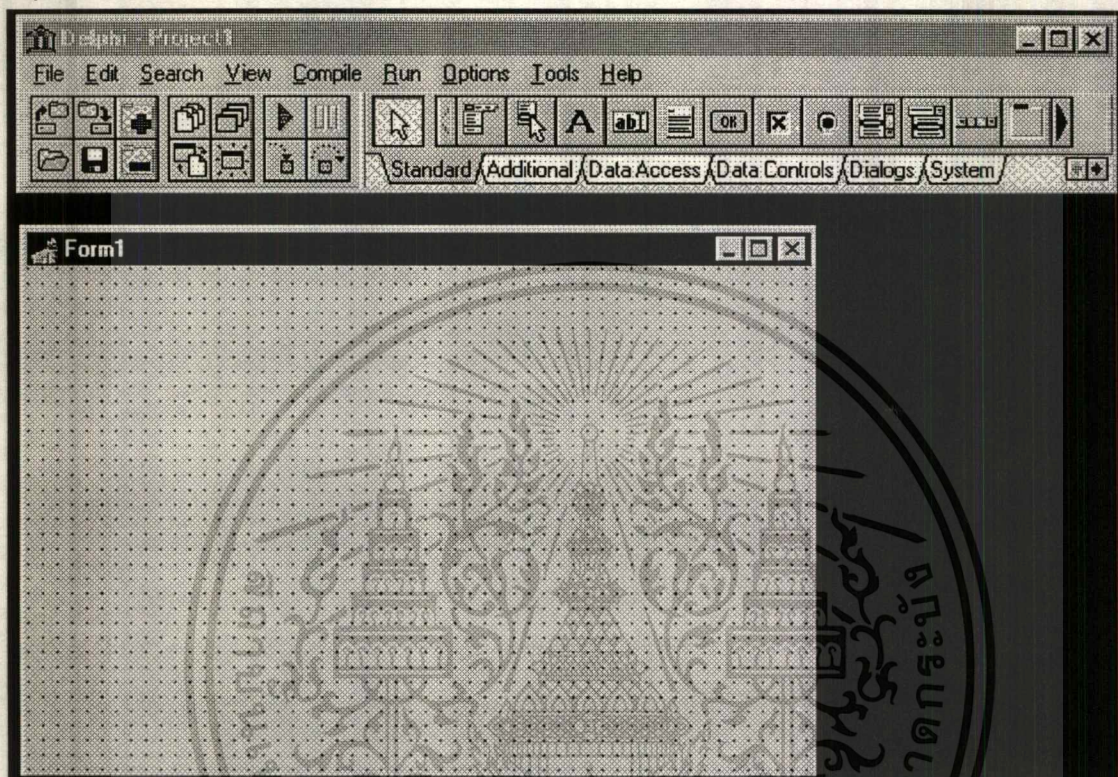
เพื่อแสดงถึงการโปรแกรมด้วยเคลฟโดยที่ยังไม่ต้องเตรียมโปรแกรมเลย ซึ่งจะได้เป็นโปรแกรมเปล่า ขอให้ดำเนินการดังนี้คือ

1. เปิดโปรเจกต์ใหม่
2. รันโปรแกรมทดสอบ

จะได้ผลบนจอภาพดังรูปที่ ก-1 แสดงหน้าต่างของโปรแกรม MyNull มีชื่อหน้าต่างว่า

Form1

จะเป็นโปรแกรมเปล่าที่ผู้ใช้ใช้งานไม่ได้ แต่ก็ เป็น Windows application คือเป็นโปรแกรมบนวินโดวส์ที่สมบูรณ์ สามารถย่อ ขยาย ย้ายหน้าต่างได้ มินิไมส์และแมกซิไมส์ได้ และยุติการรันได้โดยเลือกรายการ Close



รูป 8 แสดงหน้าต่าง Form1 ของโปรแกรม MyNull ซึ่งเป็นโปรแกรมเปล่าในขณะรันทดสอบ

โปรแกรม MyBeep

เพื่อแสดงถึงงานที่จะให้ผู้ใช้ทำได้ คือ เมื่อรันโปรแกรมแล้วหากผู้ใช้คลิกเมาส์ในหน้าต่างจะให้ส่งเสียงดัง 1 ครั้ง จะเตรียมโปรแกรมได้ดังนี้

1. เปิดโปรเจกต์ใหม่ (ขณะนี้โปรเจกต์ที่อ้างถึงเป็น โปรแกรม MyNull ยังคงเป็นโปรเจกต์ใหม่เพราะยังไม่ได้ใช้เตรียมโปรแกรม)
2. ป้อนโปรแกรม โดย
 - 2.1 เลือกหน้า Events ในหน้าต่าง Object Inspector เลือกอีเวนต์ เป็น OnClick แล้วดับเบิลคลิกในกรอบทางขวา (ของหัวข้อนี้) จะแสดงโครงโปรแกรมในหน้าต่าง Unit 1 ดังในรูปที่ 9
 - 2.2 ให้ป้อนโปรแกรมคือคำสั่ง MessageBeep ให้เป็นดังลวดดิ่งที่ 8
3. รันโปรแกรมทดสอบ จะได้ผลการรันตามที่กล่าวมา

```

UNIT1.PAS
procedure TForm1.Button1Click(Sender: TObject);
begin
|
end;

end.

```

รูป 9 แสดงหน้าต่างยูนิท Unit1 หลังจากดับเบิลคลิกที่กรอบทางขวาของหัวข้อ OnClick ในหน้า Events

ลิสต์ที่ 8

```

procedure TForm1.FormClick ( Sender : TObject );
begin
    MessageBeep (0);
end;

```

การเก็บลงไฟล์

เมื่อคอมพายล์หรือรันแล้ว หากเป็นไปตามที่กล่าวมาจะได้ไฟล์ project1.exe อยู่ในไดเรกทอรี C:\delphi ซึ่งถ้าเป็นโปรแกรมที่สมบูร์นไฟล์ .exe คือไฟล์ที่จะนำไปใช้งาน

การเก็บลงไฟล์ หมายถึง การเก็บไฟล์ในโปรเจกต์ทั้งหมด (หลายไฟล์) ไว้ในดิสก์ เพื่อเก็บไว้เป็นหลักฐาน ตัวอย่าง หรือเพื่อการปรับปรุงต่อไป

ซึ่งถ้าถือตามนี้ก็ควรจะเก็บลงไฟล์เมื่อได้ บั๊น แก่ ตรวจสอบโปรแกรมเป็นที่พอใจแล้ว แต่ในหนังสือนี้จะถือว่าและขอแนะนำให้เก็บลงไฟล์เมื่อเปิดโปรเจกต์ใหม่ 1 ครั้ง และเป็นระยะๆ ตามที่เห็นควร ทั้งนี้เพื่อ

1. ให้ได้ชื่อไฟล์ .exe
2. รวบรวมไฟล์ทุกไฟล์ของโปรแกรมหนึ่งๆ คือของโปรเจกต์หนึ่งๆ ไว้ในไดเรกทอรี

เดียวกัน

3. ถึงแม้ในเคลฟจะป้อนกันไว้แน่นอน แต่โอกาสที่จะสูญเสียไฟล์ในหน่วยความจำยังมีอยู่

เมื่อต้องการเก็บลงไฟล์ เช่น ถ้าต้องการเก็บของโปรแกรม MyBeep ลงไฟล์ จะดำเนินการได้ดังนี้

1. เตรียมไดเรคทอรีเช่น

D:\bkondphi\02start\1mybeep

2. เตรียมชื่อไฟล์ .dpr หนึ่งชื่อ เช่น ให้เป็น mybeep.dpr ซึ่งเมื่อคอมไพล์แล้วจะได้เป็นไฟล์ mybeep.exe

3. เตรียมชื่อไฟล์ .pas หนึ่งชื่อ ซึ่งในหนังสือนี้จะนำชื่อไฟล์ .dpr มาต่อท้ายด้วยเลข 1 เช่น เป็น mybeep 1.pas (ชื่อไฟล์ทั้งสองนี้จะต้องไม่ซ้ำกัน)

4. เลือกรายการ File/Save Project As จะแสดงไดอะล็อกบ็อกซ์ให้ป้อนชื่อไฟล์ ให้ดำเนินการดังนี้

4.1 เลือกไดเรคทอรีตามที่เตรียมไว้ในข้อ 1

4.2 ป้อนชื่อไฟล์เป็น mybeep1.pas เลือกปุ่ม OK

4.3 จะแสดงไดอะล็อกบ็อกซ์ใหม่ ให้ป้อนชื่อไฟล์เป็น mybeep.dpr (คือจะให้ป้อนชื่อไฟล์ .pas ก่อน .dpr)

หมายเหตุ

1. ที่ให้ชื่อไฟล์ .pas ตามที่ชื่อไฟล์ .dpr ต่อด้วยหมายเลข เช่น mybeep1.pas เพราะต่อไปอาจมีการเปิดฟอร์มหรือยูนิคเพิ่ม จะได้ให้เป็นหมายเลข 2 และต่อไป

2. ทำให้ตั้งชื่อไฟล์ .dpr ได้ไม่เกิน 7 อักขร ซึ่งชื่อนี้จะเป็นชื่อไฟล์ .exe เป็นชื่อโปรแกรมจริง แต่โดยที่ชื่อไฟล์ที่เป็นชื่อหัวข้อเช่น “โปรแกรม MyBeep” อ่อนได้ความพอสมควร ชื่อทั้งสองนี้อาจไม่ตรงกัน เช่น ButtonBeep กับ bitbeep

การกำหนดคอมโปเนนต์

หลักใหญ่ของการโปรแกรมด้วยเคลฟ คือ การนำคอมโปเนนต์มาใช้ โดยกำหนดในฟอร์มให้คำพรอพเพอร์ตี้ส์เพื่อปรับแต่งให้เป็นตามที่ต้องการ ในหัวข้อนี้จะกล่าวถึงการกำหนดคอมโปเนนต์กับเรื่องที่เกี่ยวข้องกัน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดคอมโปเนนต์

เมื่อต้องการกำหนดคอมโปเนนต์ ดังเช่นเมื่อต้องการกำหนดคอมโปเนนต์เป็นปุ่มควบคุม คือ Button ดังแสดงในรูปที่ 10 จะกำหนดได้ดังนี้

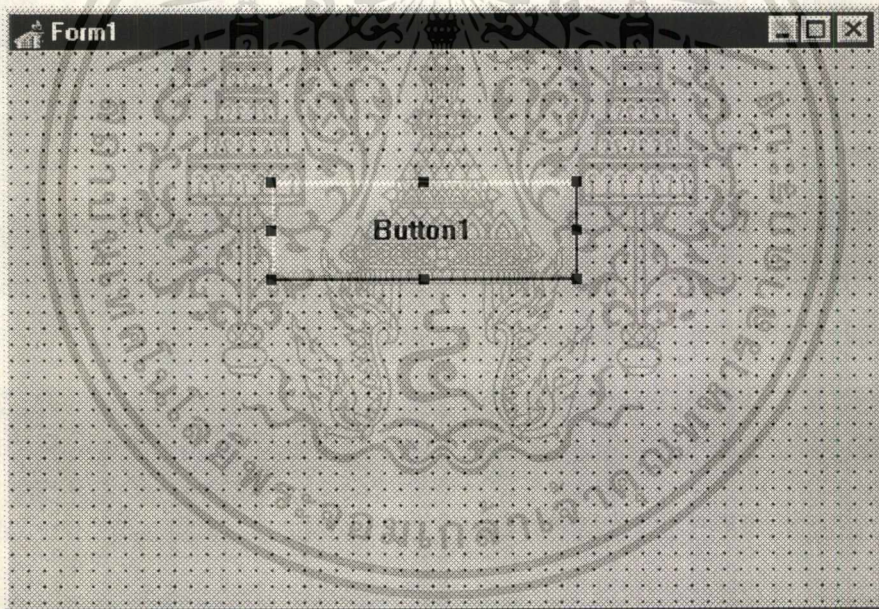
1. เลือกคอมโปเนนต์เป็น Button ในกล่องคอมโปเนนต์หน้า Standard (ปุ่มที่ 6 จากซ้ายไม่นับปุ่มหัวลูกศร)

2. นำไปกำหนดในฟอร์มโดย

2.1 คลิกเมาส์ ณ ตำแหน่ง (จุดมุมบนซ้าย) ที่ต้องการ จะได้ขนาดมาตรฐาน หรือ

2.2 ติกรอบเพื่อให้ได้ตำแหน่งและขนาดที่ต้องการ

จะได้คอมโปเนนต์ตามที่ต้องการเช่นดังรูปที่ ก-3 คือปุ่ม Button สำหรับการกำหนดขนาด และตำแหน่งขอให้ประมาณจากรูปประกอบ และขอให้ถือว่าเป็นเพียงตัวอย่างแนะนำเท่านั้น อาจกำหนดได้ตามที่เห็นควร



รูป 10 แสดงผลของการกำหนดคอมโปเนนต์เป็น ปุ่ม Button1 และอยู่ในสถานะเลือก

สถานะเลือก

เมื่อกำหนดหรือเบีอกคอมโปเนนต์ใด คอมโปเนนต์นั้นจะอยู่ในสถานะเลือก มีปุ่มเล็ก ๆ 8 ปุ่ม ซึ่งจะขอเรียกว่า “ปุ่มเลือก” ล้อมรอบ

ซึ่งถ้ากำหนดคอมโปเนนต์ใหม่ หรือเลือกคอมโปเนนต์อื่น หรือเลือกฟอร์ม คือ คลิกเมาส์ ในฟอร์ม (ตรงที่ว่าง) สถานะเลือกนี้จะหมดไป ไปอยู่คอมโปเนนต์ตัวใหม่หรือฟอร์ม แต่ฟอร์มไม่

เอกสารแสดงปุ่มเลือกที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อคอมโปเนนต์หรือฟอร์มอยู่ในสถานะเลือก จะแสดงชื่อออปเจ็กต์ (ตามที่กล่าวต่อไป) เช่นเป็น Button 1 หรือ Form 1 ที่กรอบอิดิตของกรอบคอมโป ที่ได้ไคเดิลบาร์ของหน้าต่าง Object Inspector จึงเป็นอีกแห่งหนึ่งที่จะดูว่าคอมโปเนนต์ใด หรือ โดยเฉพาะอย่างยิ่งคือ ฟอร์มอยู่ในสถานะเลือกหรือไม่

หมายเหตุ

ที่ชื่อออปเจ็กต์ตามที่กล่าวมานี้ จะแสดงชื่อสกุลกำกับด้วย เช่น TButton:Button1 ตามที่จะกล่าวต่อไป

การย้าย ย่อ และขยายขนาด

เมื่อต้องการย้ายคอมโปเนนต์ ให้คลิกเลื่อนไปไว้ ณ ตำแหน่งที่ต้องการ และถ้าต้องการย่อหรือขยายขนาดให้เล็กแล้วคลิกเมาส์ ณ ปุ่มเลือกที่อยู่ในทิศทางที่ต้องการ แล้วเลื่อนเมาส์เพื่อย่อหรือขยายขนาดตามความต้องการ แล้วปล่อยมือ

การยุบ

เมื่อกำหนดคอมโปเนนต์แล้ว หากกำหนดคิดหรือไม่ต้องการใช้ให้ยุบได้โดยเลือกคอมโปเนนต์นั้นแล้วกดคีย์ Del

หมายเหตุ

1. ขอให้ทดลองกำหนดคอมโปเนนต์โดยเฉพาะในหน้า Standard เพื่อทำความเข้าใจกับคอมโปเนนต์เหล่านั้น แล้ว ย่อ ขยาย รวมทั้งยุบด้วย แต่คงให้มีคอมโปเนนต์จำนวนหนึ่งในฟอร์มรันโปรแกรมทดสอบแล้วทดลองเลือกคอมโปเนนต์เหล่านั้นด้วยเมาส์และคีย์บอร์ด (ใช้คีย์ Tab กับ Enter)

2. เมื่อกำหนดคอมโปเนนต์ ขอให้สังเกตชื่อคอมโปเนนต์ดังเช่น Button 1 ด้วย

ชื่อคอมโปเนนต์

ในเคลฟได้สร้างสกุลไว้เป็นจำนวนมาก โดยตั้งชื่อสกุลขึ้นต้นด้วยอักษร T เช่น Tbutton แล้วนำสกุลจำนวนหนึ่งไปทำเป็นคอมโปเนนต์ ใช้ชื่อชนิดของคอมโปเนนต์ตามชื่อสกุล โดยตัดอักษร T เช่นเป็น Button

เมื่อเรากำหนดคอมโปเนนต์ในฟอร์ม จะได้ชื่อคอมโปเนนต์ตามชื่อชนิดต่อท้ายด้วยหมายเลข เช่น Button1 ซึ่งถ้ากำหนดคอมโปเนนต์ชนิดนี้อีก จะได้เป็น Button2 และต่อ ๆ ไป

โดยที่คอมโปเนนต์จะมีสถานะอยู่ 2 สถานะ คือ

1. เป็นคอนโทรลในสายตาของผู้ใช้ คือเห็นเป็นปุ่มและกรอบต่าง ๆ มีชื่อตามที่เห็น เช่น เป็นปุ่ม Button 1 ในรูปที่ 10 ซึ่งในเคลฟจะให้ค่านี้ไว้เป็นค่า Caption หรือ Text แล้วแต่ชนิดของคอมโปเนนต์ทำให้ไปปรากฏเป็นชื่อคอนโทรล (ต่อไปเมื่อกล่าวถึง Caption ขอให้หมายถึง Text ด้วยโดยอนุโลม)

2. เป็นออปเจกต์ในสายตาของผู้เขียนโปรแกรมคือ เพื่อใช้ในการเขียนโปรแกรม โดยที่ในเคลฟจะให้ค่าไว้แก่ Name เช่น เป็น Button1 เช่นกัน เป็นชื่อออปเจกต์

ค่า Caption ซึ่งใช้เป็นชื่อคอนโทรลนี้ ปกติจะให้ค่าใหม่เพื่อสื่อความหมายกับผู้ใช้ เช่น เป็นปุ่ม OK และ Cancel เป็นต้น

สำหรับค่า Name ถ้ามีคอมโปเนนต์จำนวนน้อย อาจไม่ต้องให้ค่าใหม่ได้ แต่ส่วนมากแล้วต้องให้ค่าใหม่ ซึ่งถือเป็นการตั้งชื่อเช่นกัน เพื่อให้จำง่าย อ่านโปรแกรมง่าย เช่น ถ้าให้ชื่อคอนโทรลเป็น Load, Save และ Clear (รวม 3 ปุ่ม) ก็นิยมกันและในหนังสือนี้ จะให้เป็น bttLde, bttSve และ bttClr ตามลำดับ เพื่อให้ทราบว่า เป็นคอมโปเนนต์ชนิดใด แยกไว้ว่าชื่อใดเป็นชื่อคอนโทรล ชื่อใดเป็นชื่อออปเจกต์ แต่อ้างอิงถึงกัน

หมายเหตุ

1. คอมโปเนนต์ทุกชนิดเป็นออปเจกต์แต่ไม่ทุกชนิดที่เป็นคอนโทรล
2. ตามที่กล่าวมานี้ จะเห็นว่ามีค่าถึง 3 ค่าที่อ้างถึงสิ่งเดียวกัน คือ คอมโปเนนต์ออปเจกต์ และคอนโทรลซึ่งสามารถ interchangeable ได้ โดยที่ในหนังสือนี้จะใช้แนวทางนี้คือ
 - คอมโปเนนต์ เป็นคำรวม และโดยปกติจะใช้ในการกำหนดและการเลือก
 - ออปเจกต์ในด้านการเขียนโปรแกรม
 - คอนโทรล เพื่อเน้นว่าเป็นคอนโทรลและในคำอธิบายโปรแกรม

ให้ค่าพรอพเพอร์ตี้

เมื่อกำหนดคอมโปเนนต์ เคลฟจะให้ค่าพรอพเพอร์ตี้ที่ควรให้ทั้งหมด เป็นค่าดีฟอลต์ เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับโรงเรียนโพธิ์โพธิ์ศึกษาเท่านั้น เมื่อนักเรียนได้เห็นว่าเว็บไซต์นี้เป็นการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(default) ให้ผู้เขียนโปรแกรมให้ค่าใหม่ตามต้องการ

ค่าที่ต้องให้เป็นประจำคือค่า Caption ดังจะนำมาเป็นตัวอย่างของการให้ค่าพรอพเพอร์ตี้ส์ ดังเช่น ถ้าต้องการเปลี่ยนชื่อปุ่ม Button 1 ในรูป 10 เป็น Beep จะให้ค่า Caption ใหม่ได้ดังนี้

1. เลือกคอมโปเนนต์ คือ ปุ่ม Button 1
2. เลือกหน้าต่าง Properties แล้วเลือกหัวข้อพรอพเพอร์ตี้ส์ที่ต้องการคือ Caption จะแสดง

กรอบอิดิตทางด้านขวา

(อาจเลือกหัวข้อพรอพเพอร์ตี้ส์ก่อนเลือกคอมโปเนนต์ได้)

3. ให้ป้อนค่าที่ต้องการคือ Beep ในกรอบอิดิต (โดยลบข้อความเดิมออก)

เมื่อให้ค่าพรอพเพอร์ตี้ส์แล้วจะเกิดผลแตกต่างกันดังนี้

- ปรากฏผลทันทีทุกคีย์ที่ป้อน ดังเช่นค่า Caption ข้อความที่เป็นชื่อปุ่มจะเปลี่ยนตามคีย์ที่

ป้อน

- เกิดผลเมื่อเปลี่ยนไปทำงานอื่น เช่น เลือกฟอร์ม โดยทั่วไปจะเป็นเช่นนี้

- เกิดผลเมื่อกดคีย์ Enter คือค่า Name (ดังในรูป เมื่อให้ค่า Name แล้วจะต้องกดคีย์ Enter ทุก

ครั้ง)

- เกิดผลในขณะรัน เป็นจำนวนน้อย เช่นค่า Enabled

หมายเหตุ

1. การให้ค่าพรอพเพอร์ตี้ส์ยังมีวิธีอื่นอีก เช่น ให้เลือกจากกรอบคอมโปตามทีกล่าวการในหัวข้อ “การให้ค่าเฉพาะ” ในบทที่ 2

2. การให้ค่าพรอพเพอร์ตี้ส์ในบางกรณีจะเป็นแต่เพียงให้ลบค่าที่มีอยู่เดิม

โปรแกรม ButtonBeep

เพื่อให้มีโปรแกรมไว้อ้างอิง โดยสมมติว่าได้ออกแบบโปรแกรมไว้ว่าจะแสดงหน้าต่างมีปุ่ม Beep เมื่อผู้เลือกใช้ปุ่มนี้จะส่งเสียงดังหนึ่งครั้ง

ดังจะแสดงตัวอย่างการเตรียมโปรแกรม โดยเน้นในเรื่องของการป้อนโปรแกรมตามที่จะกล่าวต่อไป

การป้อนโปรแกรม

สามารถแยกการป้อนโปรแกรมได้เป็น 2 ลักษณะคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า.

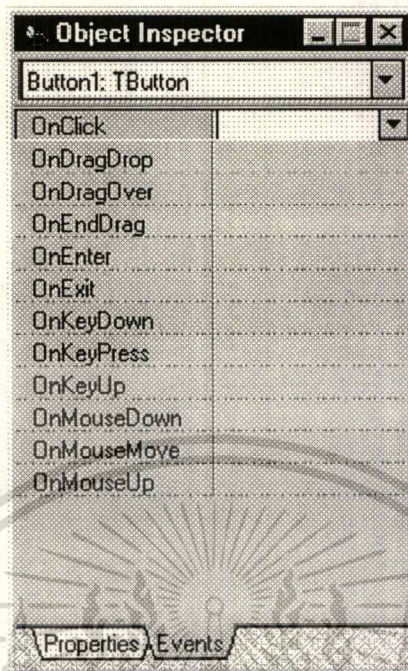
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการออกแบบ

ตั้งโปรแกรม ButtonBeep โดยขอสมมติว่าผู้อ่านเป็นผู้ออกแบบ จะเตรียมโปรแกรมได้ดังนี้

การเขียนโปรแกรมบนวินโดวส์จะเขียนจากเหตุ (เท่านั้น) ตามตัวอย่างนี้คือ “เมื่อผู้ใช้เลือกปุ่ม Beep” ดังนั้นจึงต้อง

1. กำหนดคอมโปเนนต์เป็นปุ่ม Button1 แล้วเปลี่ยนชื่อปุ่ม (ให้ค่าแก่ Caption) เป็น Beep ซึ่งโดยทั่วไปจะถือว่าเป็นต้นเหตุด้วย (ส่วนชื่อออปเจกต์คือ Name ยังคงเป็น Button1)
2. จะต้องทราบ (เช่นจากหนังสือนี้) ว่าอีเวนต์ของเหตุ “อันเกิดจากผู้ใช้เลือก” คือ OnClick และเป็นเหตุอันเกิดกับปุ่ม Button1 ดังนั้นกิจกรรมที่ต้องการคือ Button1 Click ซึ่งในแคลไฟให้เรา กำหนดกิจกรรมนี้โดย
 - เลือกปุ่มนี้ เลือกหน้า Events เลือกอีเวนต์เป็น OnClick แล้วดับเบิลคลิกในกรอบอติตทางขวา ดังที่เห็นเป็นกรอบพื้นขาวในรูป 11
 - จะได้โครงโปรแกรมของกิจกรรม Button1 Click ในหน้าต่าง VNIT.PAS ดังในลิสต์ที่ 9 ท่อนบน และขอให้ดูในรูปที่ 9 ประกอบ
3. ป้อนโปรแกรม จากการออกแบบต้องการให้ส่งเสียง 1 ครั้ง จึงให้ป้อนโปรแกรมเป็น MessageBeep (0) ดังในลิสต์ที่ 9 ท่อนล่าง



รูป 11 แสดงหน้าต่าง Object Inspector

ลิสต์ที่ 9

```

procedure TForm1.Button1Click ( Sender : TObject );
begin
end ;

-----

procedure TForm1.ButtonClick ( Sender : TObject );
begin
    MessageBeep ( 0 );
end ;

```

จากตัวอย่าง

แต่ถ้าเป็นการป้อนโปรแกรมจากตัวอย่างเช่นในหนังสือนี้ จะ แสดงโปรแกรมดังเช่นในลิสต์ที่ 9 ท่อนล่างเช่นกัน จะต้องดำเนินการคล้ายกับย้อนกลับดังนี้คือ นำชื่อกิจกรรมอีเวนต์ คือ

Button1Click

มาแยกออกเป็น Button1 คือชื่อออปเจกต์ต้นเหตุ และ Click ได้เป็น อีเวนต์ OnClick นำชื่อออปเจกต์ไปเทียบหาชื่อคอนโทรลจะได้เป็นปุ่ม Beep นำไปป้อนโปรแกรมตามที่กล่าวมาข้างต้น

โดยที่การ โปรแกรมตามหนังสือนี้ จะเป็นการป้อนโปรแกรมตามตัวอย่างทั้งสิ้น จึงขอให้ นักทบทวนด้วยว่าเป็นอีเวนต์ใด เกิดกับคอนโทรลใด คือเป็นอีเวนต์ของคอนโทรลใด

สำหรับคำว่า TForm1 ที่กำกับชื่อกิจกรรมอีเวนต์ หากมองในด้านโอโอพี จะแสดงว่า กิจกรรมนี้เป็นกิจกรรมในสกุล TForm1 หากมองทางด้านคอมโปเนนต์ จะแสดงว่า Button1 เป็นคอมโปเนนต์ในฟอร์ม Form1 (เพราะอาจเปิดฟอร์มหลายฟอร์ม)

กิจกรรมของฟอร์ม

ชื่อกิจกรรมอีเวนต์ของฟอร์มแตกต่างกับของคอมโปเนนต์ เช่น ในโปรแกรม MyBeep ได้ชื่อกิจกรรมเป็น FormClick แทนที่จะเป็น Form1Click ทั้งที่มีชื่อออปเจกต์ คือค่า Name เป็น Form1

การกำหนดคอมโปเนนต์เป็นกลุ่ม

ในโปรแกรมส่วนใหญ่มักจะต้องกำหนดคอมโปเนนต์ชนิดเดียวกันหลายตัว หรือให้ค่าคอมโปเนนต์เป็นกลุ่ม ซึ่งจะมีเรื่องเกี่ยวข้องกันดังนี้

กำหนดคอมโปเนนต์เป็นกลุ่ม

เมื่อต้องการกำหนดคอมโปเนนต์ชนิดเดียวกันหลายตัว สามารถเลือกกำหนดได้ดังนี้

1. ด้วยวิธีการตามที่กล่าวมา ซึ่งเป็นการกำหนดคอมโปเนนต์เดี่ยวทีละตัวจนได้จำนวนที่ต้องการหรือ

2. กดคีย์ Shift พร้อมทั้งคลิกเมาส์เลือกชนิดคอมโปเนนต์ในกล่องคอมโปเนนต์ นำไปกำหนดคอมโปเนนต์ให้ได้ตามจำนวนที่ต้องการแล้วเลือกคีย์ลูกศรทางซ้ายของกล่องคอมโปเนนต์ เป็นการยกเลิกการเลือกซ้ำ หรือ

3. ด้วยวิธีการลอกปะ โดยกำหนดคอมโปเนนต์แรก ซึ่งจะได้เป็นคอมโปเนนต์หลักด้วยเลือกแล้วกดคีย์ Ctrl+C เป็นการลอก แล้วกดคีย์ Ctrl+V เป็นการปะหลายครั้งเพื่อให้ได้จำนวนคอมโปเนนต์ที่ต้องการ จะได้คอมโปเนนต์ซ้อนเหลื่อมทับกัน ให้เลื่อนไปไว้ ณ ตำแหน่งที่ต้องการ

การลอกปะ จะได้คอมโปเนนต์ชนิดเดียวกันกับตัวหลัก มีค่าพารามิเตอร์ที่เหมือนกัน เว้นค่าตำแหน่งคือ Left กับ Top และถ้าได้ป้อนโปรแกรมกิจกรรมอีเวนต์ให้แก่คอมโปเนนต์หลักแล้วจะใช้กิจกรรมนี้ร่วมกัน และสามารถจะใช้ลอกคอมโปเนนต์เป็นกลุ่มคือ ลอกแล้วปะเป็นกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจับกลุ่ม

สามารถจับกลุ่มคอมโปเนนต์ชนิดเดียวกันหรือต่างชนิดกัน เพื่อการดำเนินการเป็นกลุ่ม เลือกได้ดังนี้คือ

1. ตีกรอบล้อมรอบคอมโปเนนต์ที่ต้องการ หรือ
2. เลือกคอมโปเนนต์แรก ซึ่งจะได้เป็นตัวหลักด้วย กดคีย์ Shift พร้อมกับเลือกคอมโป

เนนต์ตัวต่อไป

ยกเลิกการจับกลุ่ม โดยคลิกเมาส์ในฟอร์ม หรือเลือกคอมโปเนนต์เดี่ยว

การดำเนินการเป็นกลุ่ม

เมื่อจับกลุ่มคอมโปเนนต์แล้ว สามารถดำเนินการเป็นกลุ่มได้ดังนี้

1. ย้ายตำแหน่งด้วยการคลิกเลื่อน ซึ่งต้องคลิกเมาส์ในกลุ่ม คือที่คอมโปเนนต์ตัวใดตัวหนึ่ง
2. จัดแนว โดยเลือกรายการ Align ใน Edit หรือเลือกจากเมนูลอย (เปิดเมนูลอยได้โดยคลิกเมาส์ขวาในกลุ่ม) จะแสดงไดอะล็อกบ็อกซ์ ให้เขตค่าเพื่อการจัดแนว
3. ปรับขนาด เลือกรายการ Size ใน Edit หรือเลือกจากเมนูลอย จะแสดงไดอะล็อกบ็อกซ์ ให้เขตค่า โดยให้ป้อนขนาดใหม่ หรือเลือกให้เท่ากับคอมโปเนนต์ใหญ่สุด หรือเล็กที่สุด

การให้ค่าเป็นกลุ่ม

เมื่อต้องการให้ค่าพารามิเตอร์ดีส์เป็นกลุ่ม คือให้ใช้ค่าเดียวกันหรือให้ใช้กิจกรรมอีเวนต์ร่วมกัน ให้เลือกดำเนินการได้ดังนี้

1. จับกลุ่ม. แล้วให้ค่ากับป้อนโปรแกรมของกิจกรรมอีเวนต์ (อย่างใดอย่างหนึ่งหรือทั้งคู่) ถ้าเป็นการป้อนโปรแกรมจะได้ชื่อกิจกรรมอีเวนต์เป็นของตัวหลัก แต่จะเป็นกิจกรรมอีเวนต์ของทุกคนในกลุ่ม หรือ
2. กำหนดคอมโปเนนต์ตัวแรก แล้วให้ค่ากับป้อนโปรแกรมของกิจกรรมอีเวนต์ แล้วใช้วิธีลอกปะ

ร่วมใช้กิจกรรมอีเวนต์

เมื่อได้ป้อนกิจกรรมอีเวนต์ของคอมโปเนนต์หนึ่งหรือเป็นกลุ่มไปแล้ว หากต้องการให้คอมโปเนนต์อื่นร่วมใช้กิจกรรมด้วย ให้เลือกคอมโปเนนต์ใหม่นั้น เลือกอีเวนต์แล้วเปิดกรอบคอมโป เลือกรายการที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถอนจากกลุ่ม

เมื่อให้ค่าที่เป็นกลุ่มแล้ว หากต้องการให้ค่าอื่นหรือให้ใช้ค่าที่แตกต่างกัน รวมทั้งกิจกรรม
อีเวนต์ด้วย ทำได้โดยให้ค่าโดยตรงแก่คอมโปเนนต์นั้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การใช้งาน *DELPHI*

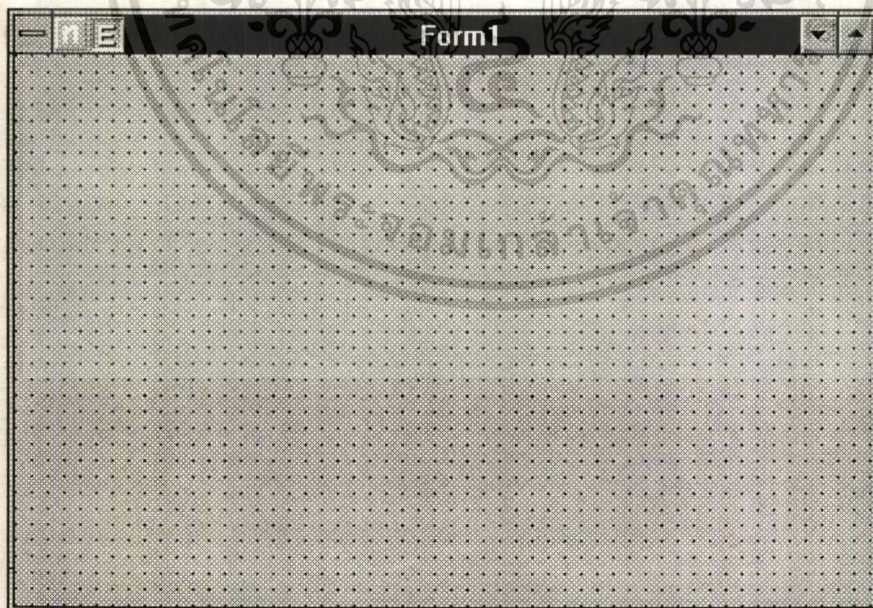
Delphi เป็นโปรแกรมที่นำมาใช้ในการพัฒนาและสร้าง Application ที่มี
ความสามารถในการจัดการกับงาน DataBase ได้อย่างมีประสิทธิภาพ และ Delphi ยัง
มีเครื่องมือที่เอื้ออำนวยความสะดวกในการสร้าง Application ต่าง ๆ ซึ่งทำให้ผู้ที่ต้อง
การสร้างงาน Application สามารถสร้างโปรแกรมได้ง่าย

ส่วนประกอบของ Delphi

ในการใช้งาน Delphi สร้าง Application นั้นเราจะทำการศึกษารายละเอียดต่าง ๆ
ที่ต้องใช้งาน ซึ่งส่วนหลัก ๆ ในการสร้างนั้นประกอบด้วย

1. *Form Designer* ของ Delphi

คือรูปแบบที่จะใช้ในการออกแบบหน้าตาของ Application ซึ่งเรา
สามารถออกแบบส่วนต่าง ๆ ที่ต้องการสำหรับติดต่อกับ User ซึ่งในครั้งแรกนั้น
Delphi จะให้ชื่อว่า Form1



รูป 11 *Form Designer*

Form : คอมพิวเตอร์ใหญ่ที่ทำหน้าที่รวบรวมคอมพิวเตอร์อื่นๆไว้ในตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Editor Window

คือส่วนที่ Delphi ใช้ในการเขียน Code ในการควบคุมเหตุการณ์ต่าง ๆ ของ Object ที่ Programmer ต้องการให้เกิดขึ้น ในส่วนนี้ Delphi ได้กำหนด รูปแบบในการเขียน Code ซึ่งเป็นรูปแบบการเขียน Code แบบ Pascal ซึ่งเราสามารถกำหนดและควบคุมโปรแกรมได้ในส่วนนี้ได้เหมือนกัน โดยมีโค้ดต่างๆ แบ่งเป็น Unit , Interface , User ,Type ,Var และ Imprementation แต่โดยทั่วไปโค้ด ส่วนใหญ่ เดลไฟจะทำการจัดการให้เราเอง

```

UNIT1.PAS
unit Unit1;

interface

uses
  SysUtils, WinTypes, WinProcs, Messages, Classes,
  Forms, Dialogs;

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

var
  Form1: TForm1;

implementation

{$R *.DFM}

end.

```

1: 1 Modified Insert

Unit1

รูป 12 The Editor Window

Unit : หน้าต่างที่คุณจะสั่งด้วยโค้ดให้เดลไฟลปฏิบัติกรใดๆ

3. Component Palette

คือเครื่องมือที่ใช้อำนวยความสะดวกในการสร้าง Application ซึ่งเป็น Icon ที่เราสามารถ Click ที่ Icon ที่เราต้องการใช้งานแล้วนำมาวางลงบน Form Designer เท่านั้น ซึ่ง Delphi จะสั่งให้มันทำงานในขณะที่ Run time ซึ่ง Component Palette นั้นยังแบ่งย่อยออกเป็นส่วนต่าง ๆ อีกหลายส่วน

ทูลบาร์ที่อยู่ด้านซ้ายของหน้าต่างหลัก แผงปุ่มลัด (Speed Button) ซึ่งจะรวบรวมคำสั่งสำคัญไว้ เช่นการปิดเปิดโปรเจกต์ การจัดเก็บ การรัน การดีบั๊ก โดยทุกปุ่มจะมีป้ายบอกตัวมันเอง (Tool Tips) ส่วนที่อยู่ทางขวาของหน้าต่างหลักคือปุ่มที่สร้างรูปอินเทอร์เฟซบนฟอร์ม

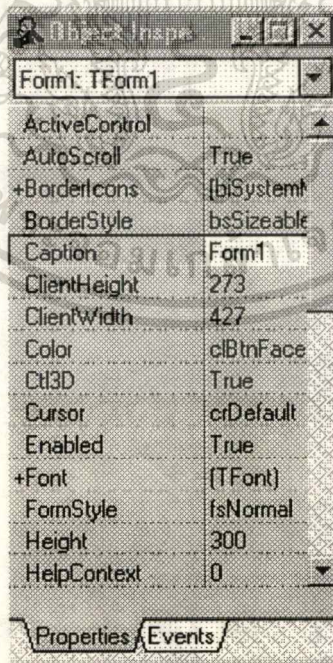


รูป 13 Component Palette

Main Menu : ส่วนที่รวบรวมคำสั่งและคอมโพเนนต์ทั้งหลายไว้

4. Object Inspector

คือส่วนที่ใช้ในการกำหนดคุณสมบัติของ Component ที่วางอยู่บน Form Designer

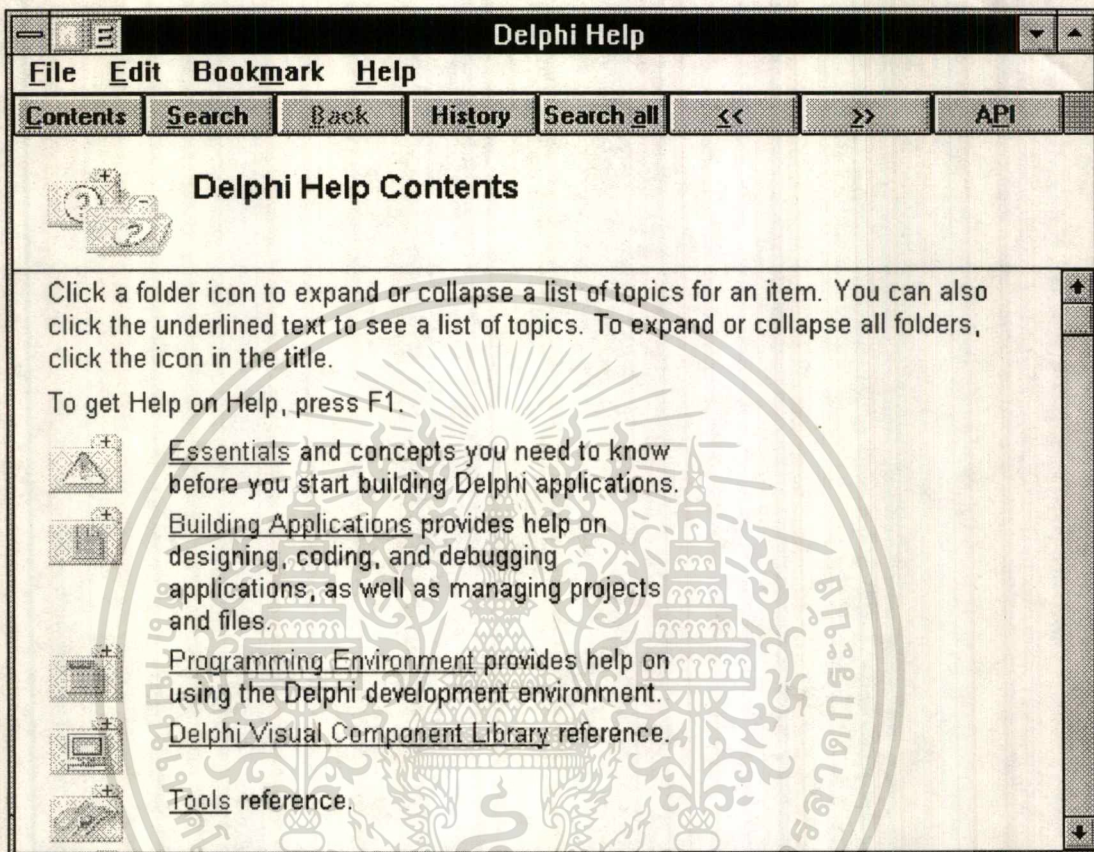


รูป 14 The Object inspector หน้าต่างที่จะนำคุณเข้าไปออกแบบการปรากฏ และการทำงานของออบเจกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Online help

คือส่วนที่ช่วยในการอธิบายรายละเอียดและวิธีการใช้งานของ Delphi



รูป 15 Online help

Delphi กับงาน DataBase

DataBase

คือแหล่งที่ใช้เก็บข้อมูลซึ่งข้อมูลเหล่านั้นมีความสัมพันธ์กันในทาง Logically ซึ่งอาจเก็บไว้ในรูปแบบความสัมพันธ์ในลักษณะต่าง ๆ ซึ่งผู้ใช้อาจเรียกดูข้อมูลใน Table โดยผ่าน View ได้ ตามขอบเขตที่กำหนด

ในทาง Physical อาจจะมีเก็บ Table ไว้ใน Drive เดียวกัน หรือต่างกัน และสามารถกระจายข้อมูลไปอย่างสม่ำเสมอ เมื่ออ่านข้อมูลจะอ่านมาแบบขนาน ซึ่งทำให้สามารถอ่านข้อมูลได้รวดเร็ว แต่เมื่อเวลาเรียกมาใช้ก็จะเห็นเป็น Table เดียวกัน

Database ส่วนใหญ่จะใช้การมองข้อมูลแบบ Logical หากมองในทาง Physical จะเป็นระดับที่ซับซ้อนมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ODBC (Open DataBase Connectivity)

คือวิธีการติดต่อและเข้าถึง Application ผู้ระบบการจัดการฐานข้อมูลโดยใช้ภาษา SQL เป็นมาตรฐานการเข้าถึงข้อมูล ความสามารถในการเชื่อมต่อแบบนี้ทำให้ Application สามารถเข้าถึงฐานข้อมูล ได้หลายรูปแบบซึ่งทำให้ผู้พัฒนาโปรแกรมสามารถพัฒนาโปรแกรมไปได้โดยไม่ต้องทำการระบุชนิดของระบบการจัดการฐานข้อมูล

แต่เดิมนั้นการพัฒนาโปรแกรมประยุกต์ที่ใช้งานเกี่ยวกับฐานข้อมูล การเข้าใช้ฐานข้อมูลโปรแกรมเหล่านี้จะผ่านการเรียกใช้ Embedded SQL ซึ่งในขณะนั้นวิธิต่างแบบนี้ก็จะเป็นไปได้ทีเดียว เพราะว่าตัวโปรแกรมสามารถ ทำการเปลี่ยนแปลงรูปแบบของระบบไม่ว่าจะเป็นทางด้านฮาร์ดแวร์ หรือ ซอฟต์แวร์ ได้หลายรูปแบบรวมทั้งระบบปฏิบัติการด้วย (โดยการคอมไพล์ไปหมดทุกครั้งที่มีการย้ายระบบ)

อย่างไรก็ตามในการพัฒนาโปรแกรมในระบบที่มีความแตกต่างกัน เช่น การเรียกใช้ข้อมูลของ Oracle จาก Microsoft Excel วิธีการเข้าถึงข้อมูลแบบเดิมนั้นจะต้องทำการคอมไพล์โค้ดของ Excel และ Oracle โดยการใช้ IBM Precompiler และ Oracle Precompiler ตามลำดับ ซึ่งจะเห็นว่าเป็นการยุ่งยากมากทีเดียว

สำหรับวิธีการต่อเชื่อมแบบ ODBC จะให้ความสะดวกในการติดต่อข้อมูลมากกว่าวิธีการดั้งเดิม โดยการกำหนดมาตรฐานการต่อเชื่อมของข้อมูล (Data Protocol,DBMS Capability) และแนวทางนี้ได้ทำให้เกิดความคิดที่จะสร้าง Driver การติดต่อกับการของานข้อมูลขึ้นมา (DLL)

ข้อดีของการติดต่อโดยใช้ ODBC

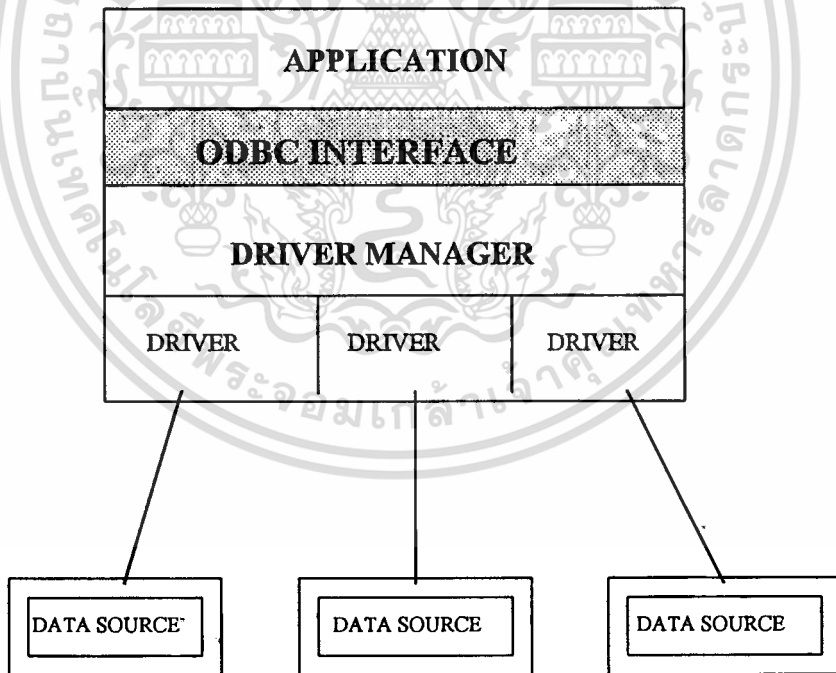
- ฟังก์ชันของ ODBC อนุญาตให้ Application ติดต่อกับระบบจัดการฐานข้อมูล ได้โดยสะดวก (การทำคำสั่ง SQL และการรับผลลัพธ์)
- ใช้ภาษา SQL ตามมาตรฐาน SQL CAE,X/OPEN และ SQL ACCESS GROUP (SAG)
- มีการกำหนดการส่งกลับรหัสความผิดพลาด (Error Code) เป็นมาตรฐานเดียวกัน
- เป็นวิธีการมาตรฐานในการติดต่อกับระบบจัดการฐานข้อมูล
- มีการกำหนดชนิดของข้อมูล (Data Type) เป็นมาตรฐาน
- ชุดคำสั่ง SQL สามารถกำหนดได้แม้ในขณะที่ Run
- สามารถเขียน โปรแกรมชุดเดียวแต่สามารถเข้าใช้ระบบจัดการฐานข้อมูล ได้หลายตัว
- ตัวโปรแกรมไม่ต้องรับผิดชอบการดูแลการติดต่อข้อมูลกับระบบจัดการฐานข้อมูล

- ค่าข้อมูลสามารถถูกส่งหรือรับได้ในรูปแบบที่สะดวกขึ้น

องค์ประกอบของ ODBC

สถาปัตยกรรมของ ODBC ประกอบด้วย 4 ส่วนสำคัญ

1. Application ทำหน้าที่ประมวลผลและเรียกใช้ฟังก์ชันของ ODBC ตามคำสั่งภาษา SQL พร้อมทั้งทำการรับผลลัพธ์ด้วย
2. ตัวจัดการ Driver Manager ทำหน้าที่โหลด Driver และเชื่อมต่อกับแหล่งข้อมูล
3. Driver ทำหน้าที่ประมวลผลการเรียกใช้ฟังก์ชันของ ODBC ส่งคำสั่ง SQL ไปสู่แหล่งข้อมูลที่ต้องการและทำการส่งผลลัพธ์กลับให้ Application และในบางครั้ง Driver จะทำหน้าที่แปลงคำสั่งที่ส่งมาให้อยู่ในรูปแบบที่สนับสนุนโดยระบบจัดการฐานข้อมูลแต่ละชนิดอีกด้วย
4. Data Source เป็นแหล่งข้อมูลที่ผู้ใช้งานต้องการเข้าถึง



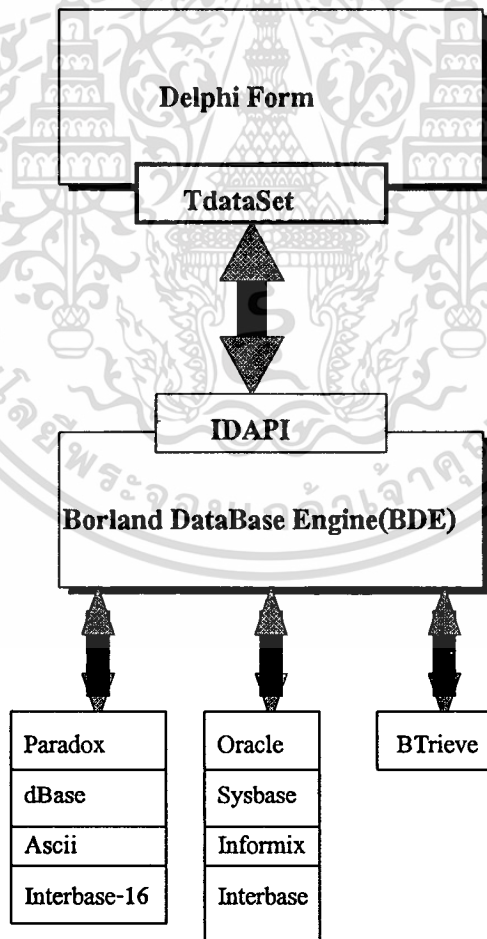
รูป 16 องค์ประกอบของ ODBC

โปรแกรมจะเรียกใช้การเชื่อม ODBC ในการทำงานต่อไปนี้

1. ร้องขอการต่อเชื่อมกับแหล่งข้อมูล
2. ส่งคำสั่ง SQL ตู่แหล่งข้อมูล
3. กำหนดพื้นที่การจัดเก็บและรูปแบบของข้อมูล ที่เป็นผลลัพธ์จาก SQL Request
4. ร้องขอผลลัพธ์
5. ประมวลผลและจัดการกับข้อผิดพลาด
6. รายงานผลให้กับผู้ใช้ (ถ้าจำเป็น)
7. ร้องขอการ Commit หรือ Roll back สำหรับควบคุมการประมวลผลTransaction
8. ยกเลิกการติดต่อกับแหล่งข้อมูล

Architecture of Delphi Database Applications

ซึ่งประกอบด้วย 3 Layers ดังรูปที่แสดง



รูป 17 ลำดับชั้นฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Layer แรก เป็น การเข้าถึง (Accessing) และจัดการกับ database สามารถทำได้โดยใช้ Standard Database Components ซึ่ง Delphi ได้จัดเตรียมไว้ให้

Layer ที่สอง เป็นตัวกลางในการจัดการกับ database จะใช้ BDE (Borland Database Engine) ทำการติดต่อ Interface กับ Database ซึ่งมี Subroutines และ Services สำหรับใช้งาน โดย BDE Application Programming Interface

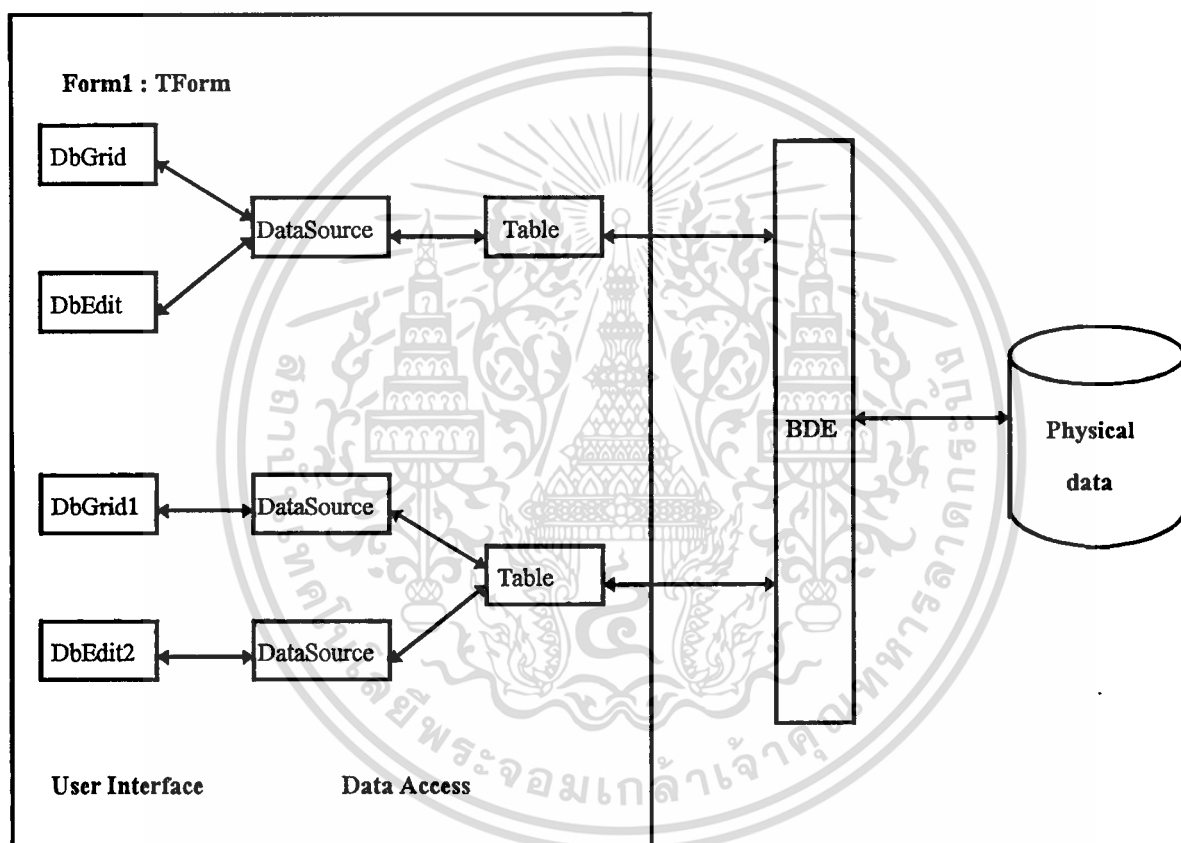
Layer ที่สาม เป็นการเข้าถึง Data ทาง Physical ซึ่งเก็บอยู่ใน Database Tables ซึ่งจะมี DBMS เป็นตัวจัดการกับ Database เช่น อาจมีรูปแบบเป็น Paradox หรือ Dbase Tables และ DBMS คือ Oracle, Sybase, Informix, or Interbase Server ซึ่งมีวิธีการจัดการแตกต่างกัน

ความสามารถและคุณลักษณะพิเศษทางด้านงานฐานข้อมูลของ Delphi

การสร้าง Application งาน DataBase โดย Delphi จะอาศัย

- เครื่องมือพัฒนางานด้านฐานข้อมูลของ Delphi (delphi database development tool)
- Delphi data access component คุณสามารถใช้มันเข้าไปกำหนดค่าของ Fields ใน records และสามารถจัดการข้อมูลได้ เช่น กำหนด table, list table ภายใน Database ซึ่งเครื่องมือเหล่านี้จะอยู่ใน Data Access page ของ Delphi's Component Palette ซึ่งประกอบด้วย TTable, TDataSource, TQuery และอื่น ๆ ซึ่งในการผ่านเข้าไปใน Low Level จะผ่าน BDE

Delphi data aware GUI component เป็น Visual Component ที่ใช้แสดงข้อมูลของ Data Field ของ Table ได้ เช่น DbGrid, DbLabel, DbListBox และอื่น ๆ ซึ่งจะเกี่ยวข้องกับ Database แต่ Tool เหล่านี้จะต้องใช้ร่วมกับ DataSource, Data Access Component โดยใช้คอม component ในการติดต่อกับบอร์เครื่องข่ายท้องถิ่น คาด้าเบส เอนจิน หรือ บีดีอี (Borland Database Engine, BDE) ซึ่งจะทำหน้าที่ติดต่อกับฐานข้อมูลอีกที ภาพต่อไปนี้จะแสดงความ สัมพันธ์ ของเครื่องมือต่าง ๆ และ Application งาน Database กับ BDE และแหล่ง ข้อมูล (DataSource)



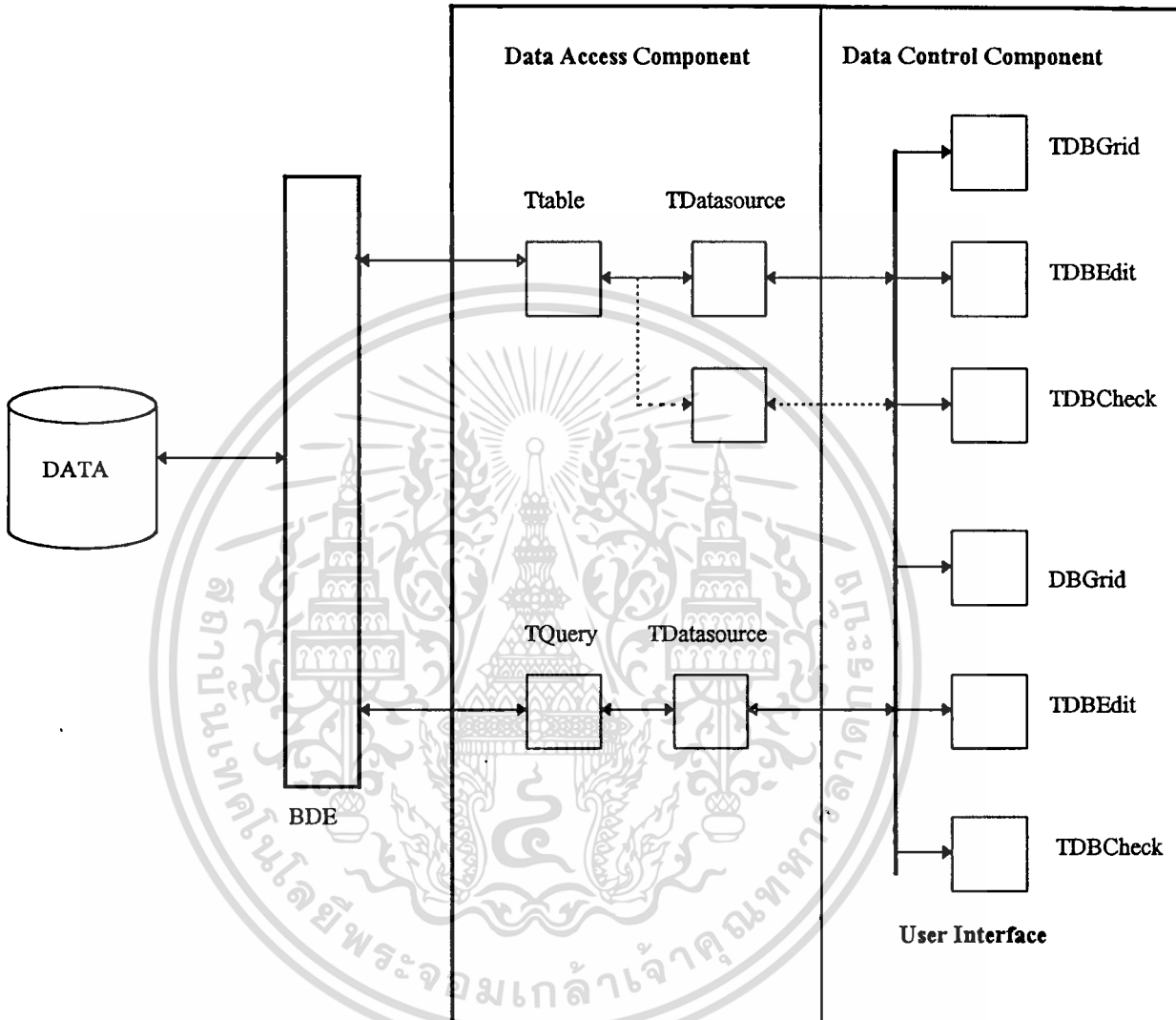
รูปที่ 18 สถาปัตยกรรมฐานข้อมูล

Delphi ใช้โอปเจตโอเรียนเตด คอมโพเนนท์ (Object-Oriented Component) ในการสร้างแอปพลิเคชันฐานข้อมูลเช่นเดียวกับแอปพลิเคชันอื่นๆที่ไม่ใช่แอปพลิเคชันฐานข้อมูล าค้าเบสคอมโพเนนท์ (Database Component) จะเหมือนกับคอมโพเนนท์มาตรฐานทั่วไป ที่มีเอทริบิวท์ หรือเรียกว่า พรอพเพอร์ตี้ (Property) ซึ่งจะถูกตั้งค่าโดยโปรแกรมเมอร์ในขณะที่ทำการออกแบบ และอาจจะตั้ง

เอกสารนี้ใช้ค่าในขณะที่กำลังรันโปรแกรมก็ได้ เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ค่าค่าเบสคอมโพเนนต์จะมีค่าที่ถูกตั้งไว้ให้แล้ว (Default) เพื่อให้มันสามารถทำงานได้ โดยไม่ต้องเขียนโปรแกรมเพิ่มหรือเพิ่มเพียงเล็กน้อย
- Delphi ประกอบด้วย 2 คอมโพเนนต์พาเลท (Component Palette) ที่มีค่าเบสคอมโพเนนต์ คือ
- หน้าสำหรับคอมโพเนนต์ที่เข้าถึงข้อมูล (Data-Access Page) จะมีออปเจกต์ที่ทำให้การเข้าถึงข้อมูลง่ายขึ้น โดยการเอนแคปซูลาท (Encapsulate) แหล่งข้อมูล เช่น ฐานข้อมูลที่จะติดต่อตารางในฐานข้อมูล หรือฟิลด์ (Field) ของข้อมูลที่ต้องการตัวอย่างของออปเจกเหล่านี้ได้แก่ Ttable Tquery ,Tdata- source และ Treport
- หน้าสำหรับคอมโพเนนต์ที่ควบคุมข้อมูล (Data-Control Page) จะมีคอมโพเนนต์ไว้ติดต่อกับผู้ใช้เพื่อแสดงข้อมูลในรูปแบบต่างๆซึ่งจะเหมือนกับคอมโพเนนต์มาตรฐานที่ใช้ติดต่อกับผู้ใช้ เพียงแต่ข้อมูลเหล่านั้นมาจากตารางในฐานข้อมูล ตัวอย่างของคอมโพเนนต์ควบคุมข้อมูลที่ใช้บ่อยๆได้แก่ TDBEdit,TDBNavigator และ TDBGrid

ค่าค่าเซตคอมโพเนนต์(Dataset Component) เช่น TTable,TQuery และ TStoreProc จะมองไม่เห็นในขณะรันโปรแกรม แต่มีไว้ให้แอปพลิเคชันติดต่อกับข้อมูลของมันผ่านบีดีคอมโพเนนต์ ควบคุมข้อมูลจะติดต่อกับ ค่าค่าเซต คอมโพเนนต์ โดยผ่านทาง Tdatasource คอมโพเนนต์ เพื่อให้การติดต่อกับข้อมูล แบบมองเห็นได้จากภาพแสดงให้เห็นว่า คอมโพเนนต์เข้าถึงข้อมูล กับ คอมโพเนนต์ควบคุมข้อมูลสัมพันธ์กับข้อมูล สัมพันธ์ซึ่งกันและกัน และสัมพันธ์ กับส่วนติดต่อกับผู้ใช้แอปพลิเคชันอย่างไร



รูป 19 การติดต่อกับฐานข้อมูล

Component Based Database Development Patterns

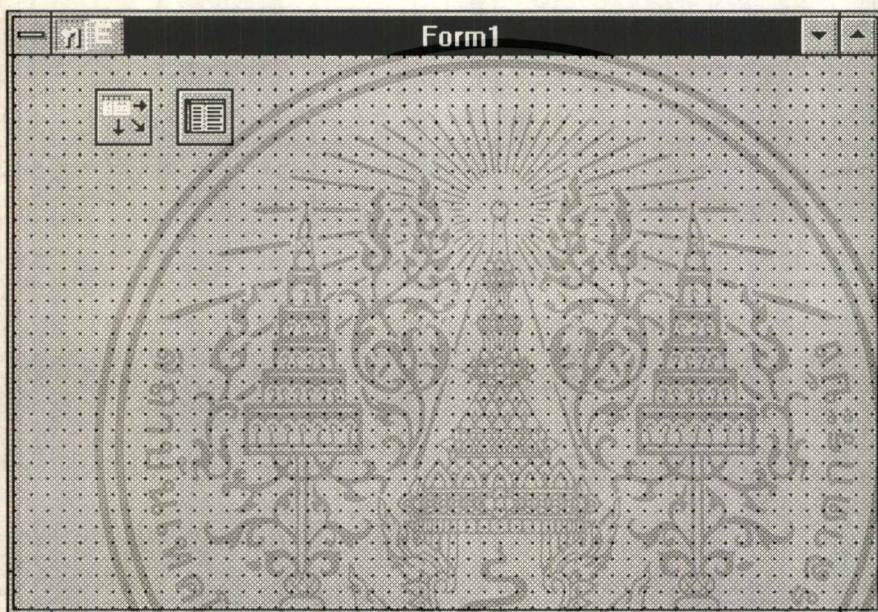
Table Browser เป็นการเรียกดู Database Table ซึ่งมีอยู่แล้ว เราสามารถใช้ Tool เพียงไม่กี่ตัว คือ TTable, TDataSource และ TDbGrid

1. สร้าง Project ใหม่ โดยการเลือก File|New ที่ Main Menu ของ Delphi ตั้งชื่อ DBROWSER.DPR จะปรากฏ Form Designer ที่หน้าจอ

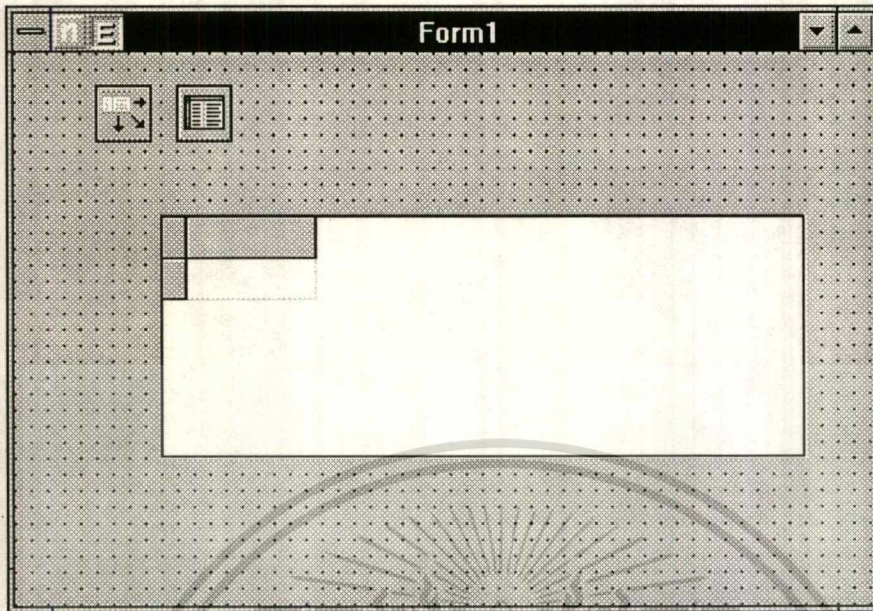
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เลือก Table Component จาก Data Access Page และวางลงบน Form Designer จากนั้น Delphi จะตั้งชื่อให้ว่า Table1 และกำหนด Type Ttable ไว้ใน Class ของ Tform

3. เลือก DataSource Component จาก Data Access Page และวางลงที่ Form Designer ข้าง ๆ Table Component ดังรูป ซึ่ง Delphi จะตั้งชื่อ ให้ว่า DataSource1



4. เลือกไปที่ Icon DbGrid จาก Data Control Page และวางลงที่ Form Designer จากนั้น Delphi จะตั้งชื่อให้ว่า DbGrid1 ซึ่งตอนนี้จะยังไม่แสดงข้อมูลออกมาเพราะ ยังไม่ได้กำหนด Property ของ Component ต่าง ๆ



5. ทำการเปลี่ยน Property ตามส่วนของข้างล่างนี้ซึ่งเป็นของแต่ละ Component ดัง

TDataSource.DataSet = Table1
TDBGrid.Align = alClient
TDBGrid.DataSource = DataSource1
TTable.DatabaseName = DBDEMOS
TTable.TableName = COUNTRY.DB
TTable.Active = True

เมื่อทำการ Set Property ตามที่กำหนดนี้แล้ว Form Designer ก็จะแสดงข้อมูลออกมาตามรูปข้างล่างนี้

Name	Capital
Argentina	Buenos Aires
Bolivia	La Paz
Brazil	Brasilia
Canada	Ottawa
Chile	Santiago
Colombia	Bagota
Cuba	Havana
Ecuador	Quito
El Salvador	San Salvador
Guyana	Georgetown
Jamaica	Kingston

หลังจาก Run Project แล้วจะปรากฏรูปข้างล่างซึ่ง TTable, TDataSource จะหายไป

Name	Capital
Argentina	Buenos Aires
Bolivia	La Paz
Brazil	Brasilia
Canada	Ottawa
Chile	Santiago
Colombia	Bagota
Cuba	Havana
Ecuador	Quito
El Salvador	San Salvador
Guyana	Georgetown
Jamaica	Kingston

จะมี highlighting bar ปรากฏด้านข้าง ซึ่งเราสามารถเลื่อนไป มาเพื่อที่จะ ดูแต่ละ ส่วนของ field ได้ ซึ่งเราจะเห็น current field ซึ่งจะมีลูกศรชี้อยู่ข้างหน้าและที่ field จะมีแถบสีแสดงอยู่ ซึ่งเราสามารถเปลี่ยนแปลงค่าใน field หรือ insert record ใหม่ได้ (edit mode=true) แต่เมื่อคุณเพิ่ม record ใหม่เข้าไป แล้วกลับไป ที่ form designer คุณ จะไม่เห็น record ที่คุณเพิ่มเข้าไป เพราะ table1's active เป็น true คือจะเก็บ ค่าเริ่ม

แรกที่มีมันอ่านเข้ามา แต่ถ้าทำการเปลี่ยนเป็น false แล้วกลับเป็น true ใหม่คุณจึง จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

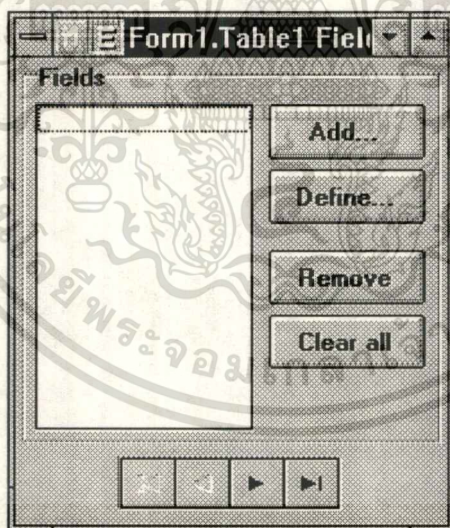
เห็นค่าที่คุณ insert ส่วนการ delete เราจะใช้ arrow key เลื่อนไปที่ record ที่ต้องการ delete แล้วกด key control delete เท่านั้นข้อมูลก็จะหายไป

Using Data Set Designer

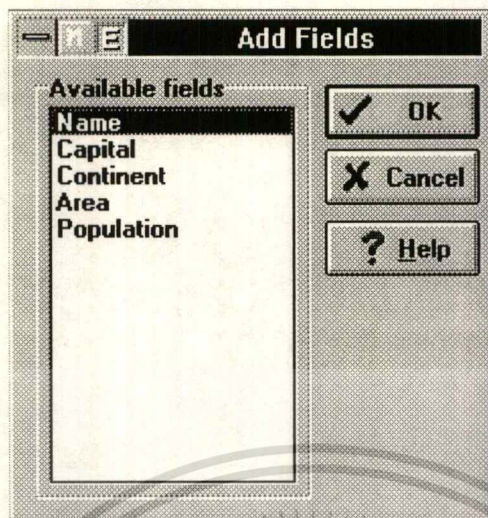
จากตัวอย่างที่ผ่านมาใช้ Physical Structer ของ Country Data Table มาแสดงที่ DBGrid ซึ่ง display ที่ DBGrid จะแสดง coloumn และลำดับตามที่อยู่ใน table จริง แต่เราสามารถที่จะกำหนดขอบเขตลำดับของ display ใน DBGrid ได้ขณะ run-time โดยใช้ Data Set Designer ซึ่งจุดประสงค์ของมัน คือ ต้องการที่จะให้คุณสามารถควบคุม data ให้แสดงเฉพาะที่จะใช้หรือจัดลำดับในการ display ได้

วิธีการใช้งาน

1. double-click ที่ TTable Component ใน Form Designer จะปรากฏรูปของ Data Set Designer



2. click ที่ปุ่ม Add จะแสดง Data Column ที่สามารถเลือกใช้ได้



3. ให้คุณเลือก Name, Area and Population column แล้ว click ปุ่ม OK.

จากนั้นจะปรากฏ Column ซึ่งคุณเลือกที่ List Box ของ Data Set

Designer และจะไปปรากฏที่ DBGrid

Name	Area	Population
Argentina	2777815	32300003
Bolivia	1098575	7300000
Brazil	85111968	150400000
Canada	9976147	26500000
Chile	756943	13200000
Colombia	1138907	33000000
Cuba	114524	10600000
Ecuador	455502	10600000
El Salvador	20865	5300000
Guyana	214969	800000
Jamaica	11424	2500000
Mexico	1967180	88600000

4. และจะมีการเปลี่ยนแปลงตามที่เรา set ไว้ที่ Data Set Designer

การเปลี่ยนแปลงนี้จะอยู่ถาวร ทุกครั้งที่คุณ run Application ที่ DBGrid ก็จะมี

ปรากฏค่าตามที่กำหนดไว้

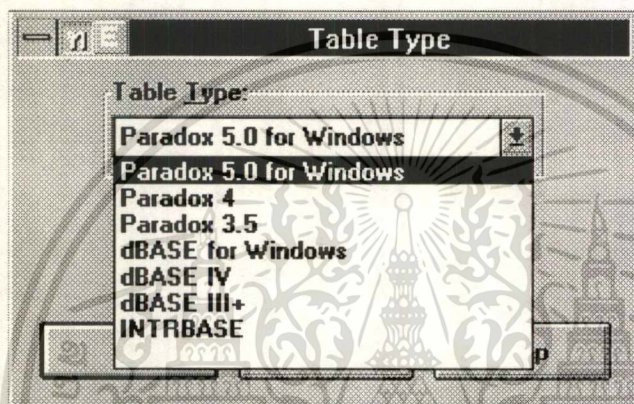
การใช้ Database Desktop สร้าง tables

มี 2 วิธีสำหรับ Delphi

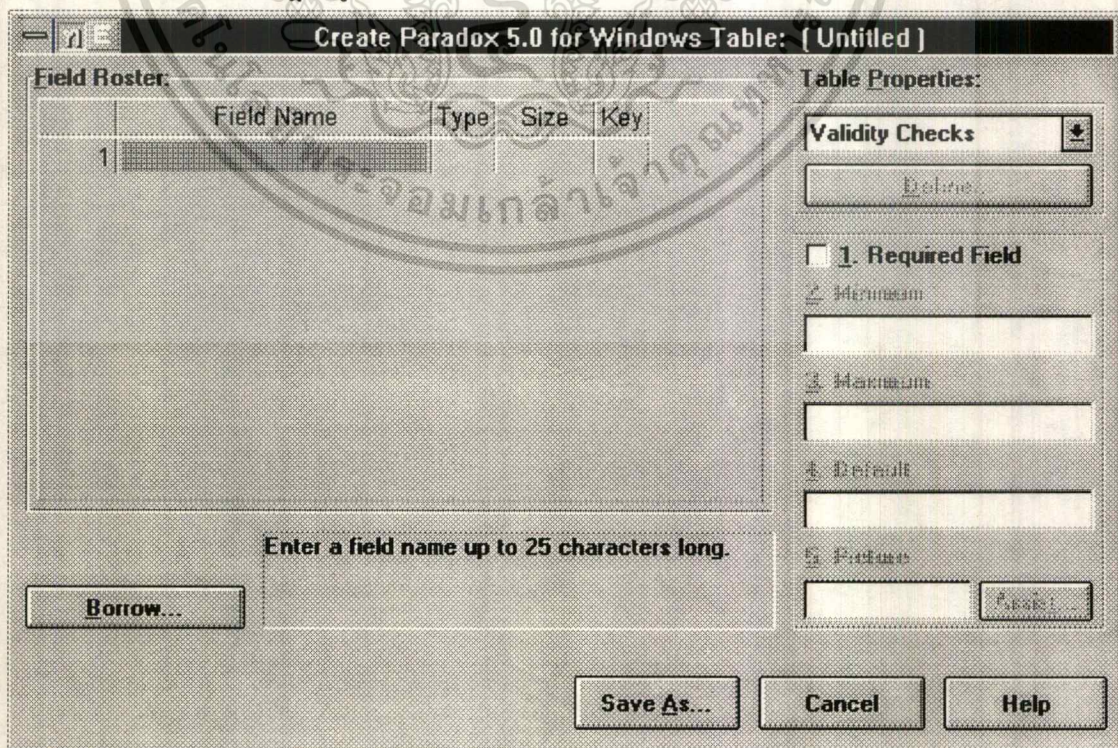
1. ใช้ DBD
2. ถ้าต้องการควบคุม table ที่สร้างขึ้นควรใช้ TQuery object และ SQL Statements

การใช้ Database DeskTop สร้าง Table หลังจากที่คุณเปิด DBD ให้เลือก

File|New|table เพื่อสร้าง Table และจะปรากฏ ชนิดของ Table ให้คุณเลือก ดังรูป



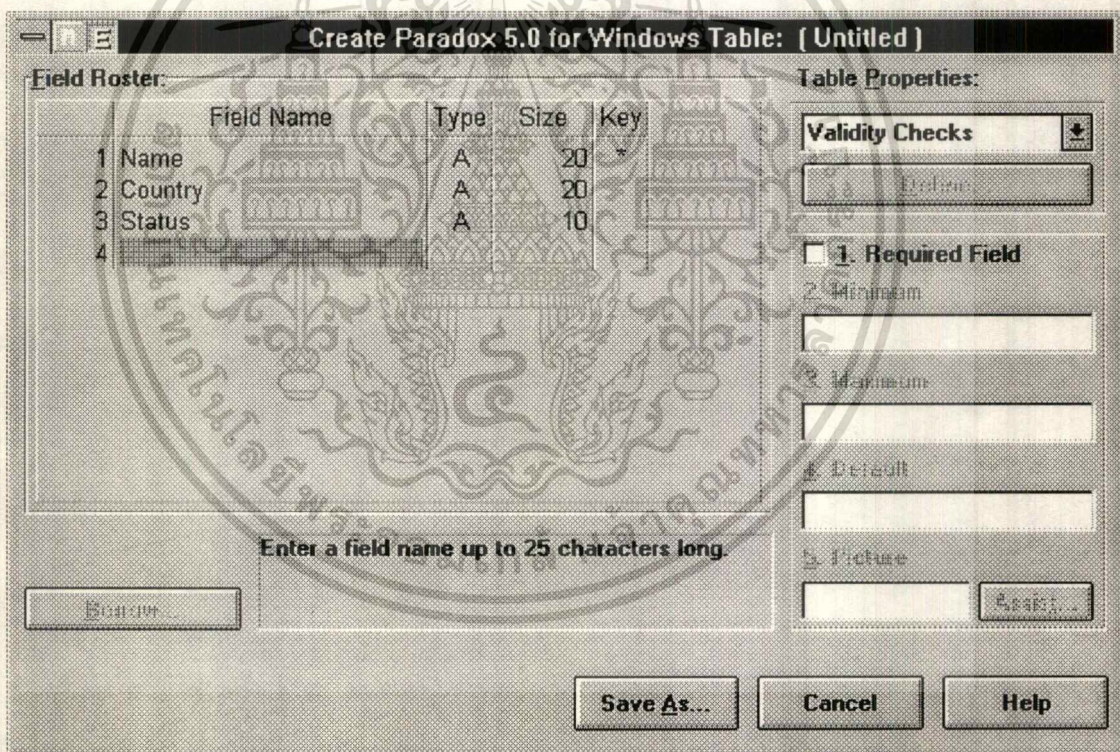
เราจะใช้ Paradox 5.0 for windows table จากนั้น DBD จะให้คุณกำหนดชื่อ และ ชนิดของ fields จะปรากฏดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อที่คุณกำหนดในรูปแบบ Paradox Field จะต้องเป็นไปตามกฎ

- Names ต้องประกอบไม่เกิน 25 ตัวอักษร
- Names ต้องไม่ขึ้นต้นด้วยช่องว่าง แต่ภายในมีได้ แต่ถ้าต้องการที่จะใช้งานร่วมกับ format อื่น ควรจะมีไม่เกิน 9 ตัวอักษร และไม่ควรมีช่องว่างภายในชื่อ
- และต้องไม่มี [], { } หรือ () สัญลักษณ์พิเศษต่าง ๆ การกำหนด Type ใช้ click ด้านขวาของ mouse แล้วเลือกตามชนิดที่ต้องการหรือจะพิมพ์เองก็ได้ ซึ่งเป็นสัญลักษณ์ของชนิดที่มีใน field type ในส่วนของการกำหนด primary index กำหนดได้โดยใส่เครื่องหมาย * ที่ key field ในรูปแบบ Paradox จะกำหนดให้ field แรกของ record จะถูกกำหนดให้เป็น Primary Key จากตัวอย่างดังรูป

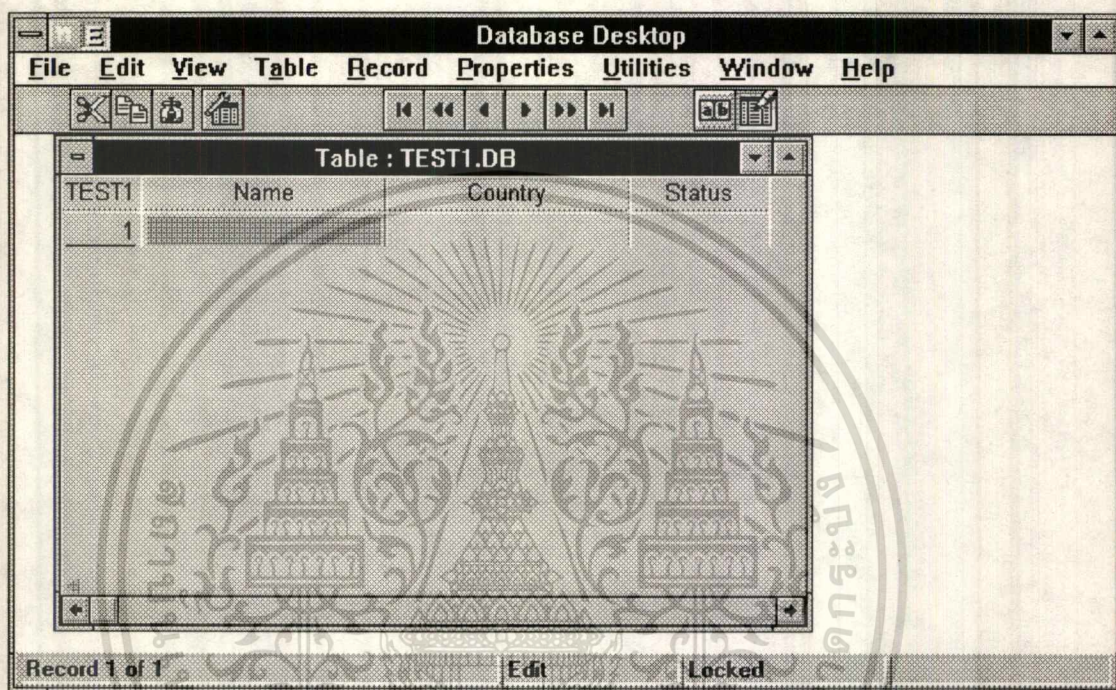


คุณอาจสร้าง a secondary index, validity check or referential integrity โดยเลือก

Utility|Restructure จาก DBD main menu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่คุณสร้าง Table เสร็จแล้วคุณสามารถเข้าไปบันทึกข้อมูล ด้วย โดยการเลือก File|Open|Table ชื่อ Table ของคุณจากนั้นเลือก Restructure จากนั้น Click ที่ icon Edit ก็จะแสดงให้คุณดังรูป



และถ้าต้องการแก้ไขก็สามารถเข้าไปแก้ไขเพิ่มเติมได้โดยใช้

Table|Table Restructure

การสร้าง Tables ด้วย SQL

ในการสร้าง Tables ด้วย SQL จะมีรายละเอียดและ Syntax ในการสร้างแตกต่างกันไปตาม Server ที่คุณใช้ แต่พื้นฐาน basic statement ส่วนใหญ่จะสามารถใช้กับ Server ทั่ว ๆ ไป

การสร้างขั้นแรกให้ drop a Tquery object ลงที่ form กำหนด DatabaseName Property to a valid alias และพิมพ์

```
create table address (Firstname char(20),Lastname char(20))
```

ใน SQL property ดังรูปถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



จากนั้น Run Program โดยเรียก Query.Open โดยใส่ที่ procedure TForm.create แล้ว server จะสร้าง table ให้

หลักการพัฒนาแอปพลิเคชันงานฐานข้อมูล

1. รูปแบบของการพัฒนา

เนื่องจากการออกแบบแอปพลิเคชัน จะต้องขึ้นกับโครงสร้างฐานข้อมูลที่จะเข้าถึง โดยฐานข้อมูลจะต้องถูกกำหนดล่วงหน้าก่อนที่จะทำการพัฒนาแอปพลิเคชัน

รูปแบบในการพัฒนาแอปพลิเคชันงานฐานข้อมูลของ Delphi เป็นไปได้ 4 รูปแบบคือ

- ไม่มีฐานข้อมูลอยู่ก่อน หรือจำเป็นต้องทำการกำหนดใหม่หมด สามารถทำได้โดย
- ใช้ ดาต้าเบส เดสก์ทอป ยูทิลิตี้ (Database Desktop Utility) เพื่อกำหนดตารางแบบพาราดอกซ์และดีเบส

- สำหรับ เอสคิวแอลเซิร์ฟเวอร์ สามารถใช้เครื่องมือที่เป็นของเซิร์ฟเวอร์ หรือของดาต้าเบส เคสค์ทอปก็ได้ ตัวอย่างเช่น สำหรับโลกอลอินเตอร์เบตเซิร์ฟเวอร์ สามารถใช้ วินโดว์ ไอเอสคิวแอล (Windows ISQL) ในการกำหนดฐานข้อมูล

- มีฐานข้อมูลปรากฏอยู่บนแหล่งข้อมูล ที่เป็นเคสค์ทอป หรือ เป็น เครือข่ายท้องถิ่น (LAN) และฐานข้อมูลเข้าถึงข้อมูล ณ ที่แห่งนั้น ถ้าบีดีอี และ แหล่งข้อมูลอยู่บนเครื่องเดียวกันกับแอปพลิเคชันแล้ว แอปพลิเคชันนั้น จะเป็น แบบสแตนด์ออล (Stand-Alone)

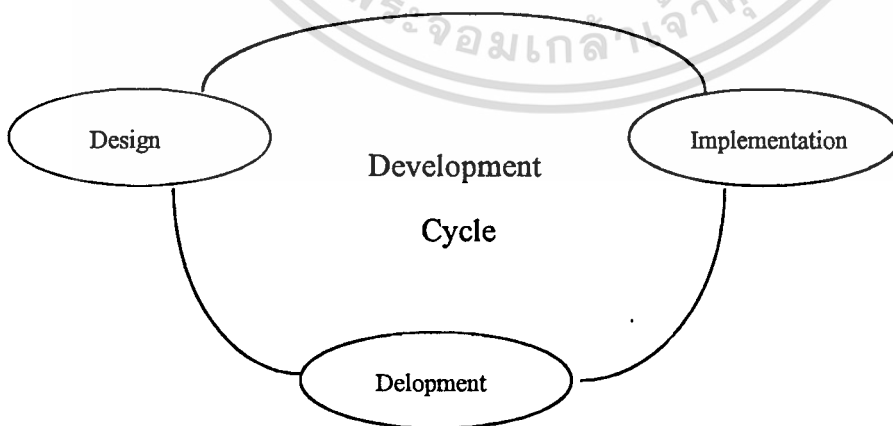
- ฐานข้อมูลปรากฏบนแหล่งข้อมูลแบบเคสค์ทอป และจะถูกอัพไซส์ (Upsize) เป็นเอสคิวแอลเซิร์ฟเวอร์แบบนี้จะเรียกว่าเป็น โลกอลเอสคิวแอลเซิร์ฟเวอร์

- ฐานข้อมูลอยู่บนเอสคิวแอลเซิร์ฟเวอร์และแอปพลิเคชันจะเข้าถึง ข้อมูลที่อยู่เซิร์ฟเวอร์ แบบนี้จะเรียกว่าเป็น ไคลเอ็นท์เซิร์ฟเวอร์แบบมาตรฐาน

2. วงจรในการพัฒนาแอปพลิเคชันงานฐานข้อมูล การพัฒนา แอปพลิเคชันงานฐานข้อมูลมี 3 ขั้นตอนหลักคือ

- การออกแบบ และ ทำต้นแบบ (Prototyping)
- การสร้าง (Implement)
- การส่งมอบระบบใช้งานจริง (Deployment) และการบำรุงรักษา

(maintenance)



วงจรในการพัฒนาสำหรับแอปพลิเคชันแบบไคลเอ็นท์เซิร์ฟเวอร์ งานของ ฐาน

ข้อมูลและงานของแอปพลิเคชันจะแยกกันเด่นชัดเนื่องจากจะรันบนแพลตฟอร์มที่ต่างกัน และระบบปฏิบัติการมักจะต่างกันด้วย เช่น ยูนิกซ์เซิร์ฟเวอร์ และวินโดวส์ 3.1 ไคลเอนท์ ดังนั้น ในการพัฒนา ความรับผิดชอบในการแบ่งว่างานใดควรทำโดยคาด้าเบส เซิร์ฟเวอร์ งานใดควรทำโดยไคลเอนท์แอปพลิเคชันจะอยู่ในขั้นตอนการออกแบบ โดยทั่วไปแล้วจะมีการแบ่งงาน กันอย่างชัดเจนระหว่าง 2 ฝั่ง แต่กรรมวิธีพื้นฐาน ข้อมูล เช่น สตอร์โพรซิเคอร์บางอย่างก็สามารถทำได้โดยไคลเอนท์แอปพลิเคชัน ดังนั้นจึงขึ้น กับลักษณะของการนำระบบไปใช้จริงที่คาดหวังไว้, ความต้องการของแอปพลิเคชัน และ ข้อควรพิจารณาอื่นๆ ดังนั้นในการออกแบบสามารถจะ กำหนด ไว้การทำงานจะอยู่ฝั่ง ไคลเอนท์ หรือฝั่งเซิร์ฟเวอร์

การส่งมอบแอปพลิเคชันเพื่อนำไปใช้จริง (Deploying an application)

หมายถึงการนำแอปพลิเคชัน ไปให้ผู้ใช้งานใช้ และจัดหาซอฟต์แวร์ที่จำเป็นที่ ผู้ใช้ จะต้องมีไว้ใช้ในการรันแอปพลิเคชัน ในสถานะแวดล้อมของผู้ใช้ สำหรับ แอปพลิเคชันที่ไม่ใช่งานฐานข้อมูลของ Delphi จะใช้เพียงไฟล์ .EXE เท่านั้น เนื่องจากแอปพลิเคชันของ Delphi ไม่ต้องการตัวแปลภาษาขณะทำงาน (run-time interpreter) หรือ ไดนามิก ลิงค์ ไบเบรารี (DLL)

ไฟล์ที่ต้องการในการส่งมอบแอปพลิเคชันเพื่อนำไปใช้จริง (แบบสแตนด์ออล)

โดยทั่วไปในการส่งมอบแอปพลิเคชันงานฐานข้อมูลเพื่อนำไปใช้จริง จะต้อง สร้างแพคเกจ (Package) ที่ประกอบด้วยไฟล์ทั้งหมดที่ผู้ใช้งานต้องการในการรัน แอปพลิเคชันและเข้าถึงแหล่งข้อมูล

ไฟล์เหล่านั้นได้แก่

- ไฟล์ .EXE และ ไฟล์ .DLL (ถ้ามี) ของแอปพลิเคชัน
- ไฟล์ที่เป็นส่วนประกอบต่าง ๆ เช่น ไฟล์ .README . ไฟล์ .HLP หรือ

ไฟล์สำหรับ ออนไลน์เฮल्प (OnLine Help)

- บีดีอี ซัพพอร์ต สำหรับการเข้าถึงฐานข้อมูล (เดสก์ทอป / เซิร์ฟเวอร์)
- รีพอร์ทสมิธ รันไทม์ (ReportSmith Runtime) สำหรับการพิมพ์รายงาน

- ถ้าแอปพลิเคชันใช้ วีบีเอ็กซ์คอนโทรล (VBX Control) จะต้องใช้ ไฟล์

BIVBX11.DLL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

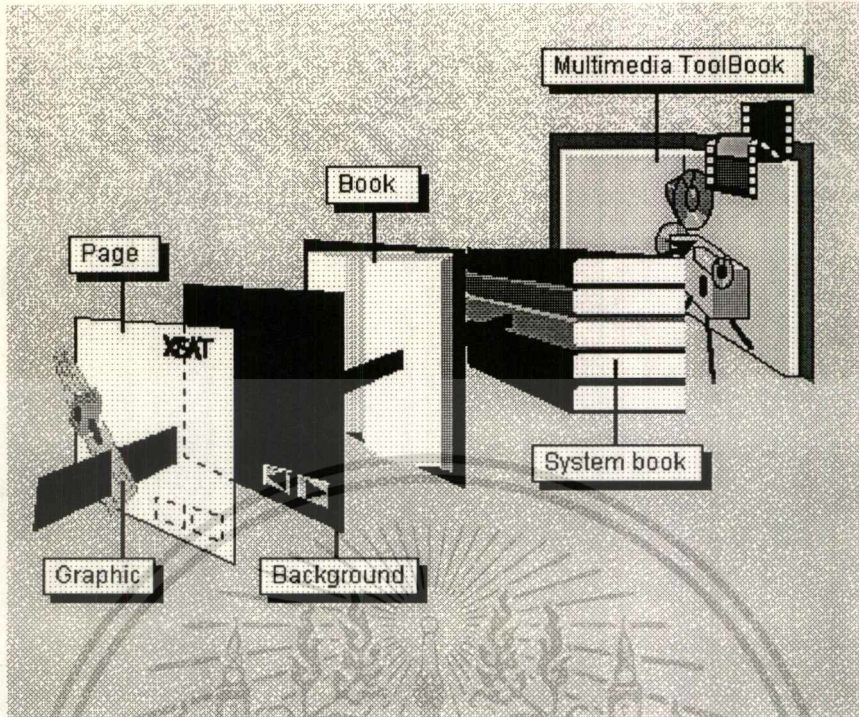
multimedia ToolBook 3

MTB 3 เป็นโปรแกรมที่ใช้สำหรับการพัฒนาระบบมัลติมีเดียภายใต้ระบบปฏิบัติการใช้Windows ได้อย่างมีประสิทธิภาพโปรแกรมหนึ่ง โดยที่ MTB 3 เป็นการเขียนโปรแกรมแบบเสมือนจริง (Visual Progaming) ที่สามารถเชื่อมต่อและเรียกใช้ทรัพยากรต่างๆของ windows รวมถึงแอปพลิเคชันต่างๆที่ทำงานบนวินโดวได้เป็นอย่างดีทำให้ MTB มีข้อได้เปรียบโปรแกรมที่ทำหน้าที่พัฒนาระบบมัลติมีเดียอื่นๆที่มีอยู่

MTB 3 มีการทำงาน 2 ระดับ คือ ระดับของออเทอร์ (Autho Mode) เป็นระดับที่ใช้ในการเขียนโปรแกรมพัฒนาระบบมัลติมีเดีย และ ระดับรีดเดอร์ (Reader Mode) ซึ่งเป็นระดับที่ใช้ในการทดสอบ และ สั่งโปรแกรมที่เขียนไว้ให้ทำงาน

MTB 3 เป็นโปรแกรมแบบอ็อบเจกต์ โอเรียนเต็ด (Object - Oriented) ม อีเวนไคร์เวน (Event - Driven) ซึ่งเป็นการเขียนโปรแกรมรูปแบบใหม่ที่ใช้กันอยู่ในปัจจุบัน การเขียนโปรแกรมควบคุมจะเขียนคำสั่งลงในสคริป (Script) โดยที่การควบคุมการส่งผ่านข้อมูลจะทำตาม ลำดับชั้นของอ็อบเจกต์ (Object Hierarchy) โดยเรียงลำดับดังนี้

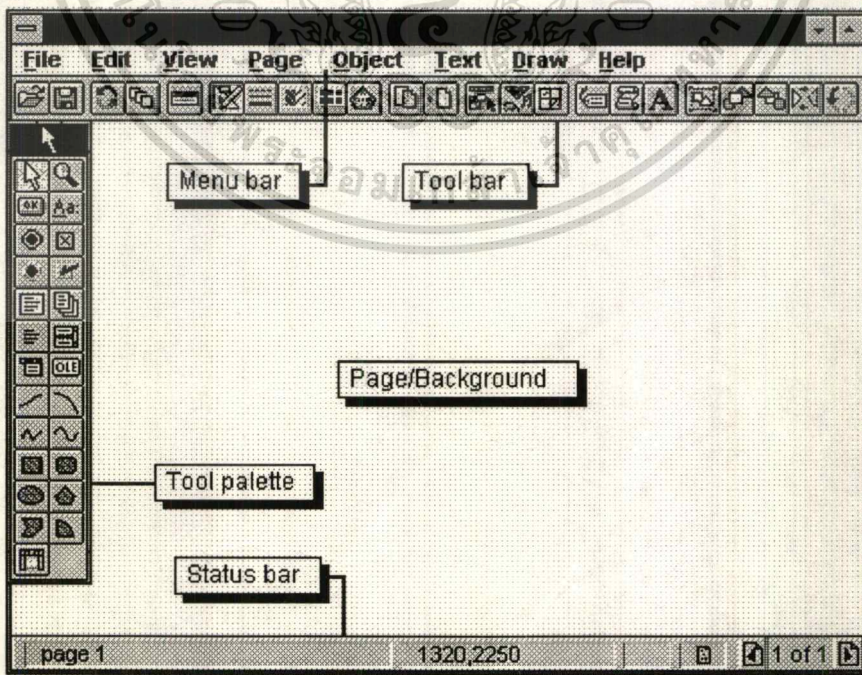
- อ็อบเจกต์ (Object)
- กรุป (Group)
- เพจ (page)
- แบ็คกราวด์ (Background)
- บ็อค (Book)
- ซิสเต็มบ็อค (System Book)
- ทูลบ็อค (Tool Books)



รูป 20 ลำดับชั้นของ Object Hierarchy

หน้าที่ต่างๆของ Multimedia ToolBook 3

MTB 3 เป็นโปรแกรมที่มีภาวะแวดล้อมของการเขียนโปรแกรมแบบเสมือนจริง (Visual Programming Environment) หน้าที่ที่ใช้สำหรับพัฒนาโปรแกรมจะประกอบไปด้วยส่วนต่างๆคือ



รูปที่ 21 จอภาพหลักของ MTB 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

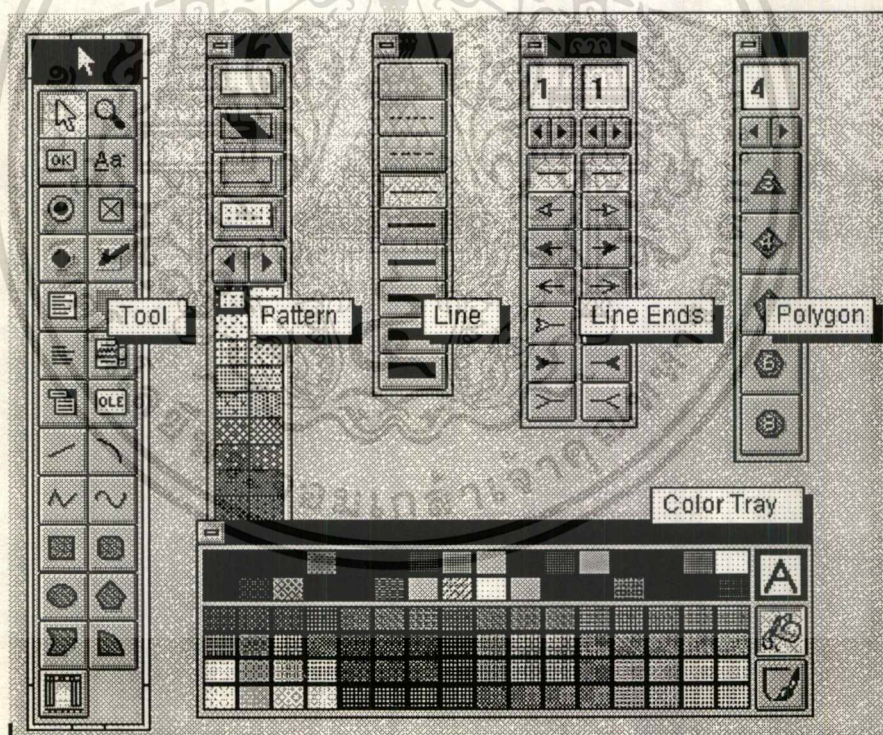
- บริเวณทำงาน (Work Area) เป็นบริเวณที่ออปเจ็ก จากการเขียน โปรแกรมของ โปรแกรมเมอร์จะรวมกับ โฟร์กราวด์ (Foreground) และแบ็กกราวด์ (Back Ground) ของ แอปพลิเคชัน

- แถบเครื่องมือ (Tool Bar) เป็นแถบที่ประกอบไปด้วยไอคอนที่ใช้สำหรับการเข้าถึงเมนู คำสั่งอย่างรวดเร็ว

- ชุดเครื่องมือ (Tool Palette) เป็นแถบที่ประกอบด้วยทูลบาร์ต่างๆ ที่ไม่สามารถเข้าถึงได้ จากเมนู

- เมนู (Menu) เป็นกลุ่มของคำสั่งที่สามารถเข้าถึงได้โดยเมนูบาร์

- แถบสถานะ (Status Bar) เป็นแถบบอกสถานะของข้อมูลต่างๆ ของ MTB 3 ซึ่งอยู่ด้านล่างของจอภาพ




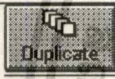




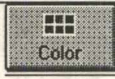






รูป 22 เครื่องมือต่างๆของ MTB 3






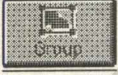
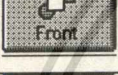
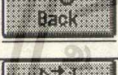


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถบเครื่องมือ (ToolBar)

แถบเครื่องมือจะประกอบด้วยไอคอนของคำสั่งที่มักใช้งานบ่อยครั้งในการทำงานซึ่งได้ออกแบบเพื่ออำนวยความสะดวกให้โปรแกรมเมอร์ แถบเครื่องมือจะประกอบด้วยกลุ่มของไอคอนดังตาราง




ไอคอน	คำสั่ง	คำอธิบาย
	OPEN	เปิดไฟล์ ของ MTB
	SAVE	บันทึกข้อมูลของไฟล์ที่กำลังทำงานอยู่
	Undo	ยกเลิกการทำคำสั่งสุดท้ายที่ทำไปแล้ว
	Duplicate	สร้างอ็อบเจกต์ที่เลือกไว้ซ้ำ
	Command	แสดงหน้าต่างคำสั่ง (command windows)
	Tool	แสดง/ซ่อน ชุดเครื่องมือ
	Line	แสดง/ซ่อน ชุดเครื่องมือเกี่ยวกับเส้น (Line Palette)
	Pattern	แสดง/ซ่อน ชุดเครื่องมือเกี่ยวกับรูปแบบ (Pattern Palette)
	Color	แสดง/ซ่อน ชุดเครื่องมือเกี่ยวกับสี (Color Palette)
	Polygon	แสดง/ซ่อน ชุดเครื่องมือเกี่ยวกับรูปเหลี่ยม (Polygon Palette)
	Background	เปลี่ยนสู่การแก้ไขแบ็คกราวด์
	New Page	สร้างหน้าใหม่
	Resources	แสดงรีซอร์สเมนเจอร์ (Ressources Manager)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

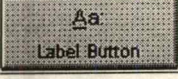




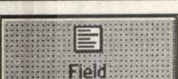
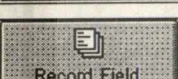


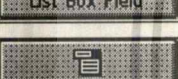
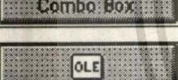
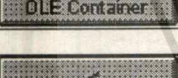

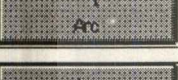
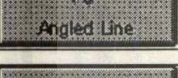
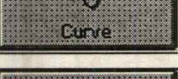

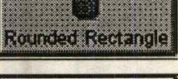
	Clips	แสดงคลิปแมนเนเจอร์ (Clip Manager)
	Viewers	แสดงการโต้ตอบของวิวเวอร์ (Viewer Dialog box)
	Properties	แสดงการตอบโต้ของคุณสมบัติของอ็อบเจ็กต์ (Object Properties)
	Script	เปิดหน้าต่างแก้ไขสคริปต์ (Script Editor) ของอ็อบเจ็กต์ที่เลือก
	Charactor	แสดงกรอบโต้ตอบของอักษร (Charsactor Dialog Box)
	Group	รวมกลุ่ม/แยกกลุ่ม อ็อบเจ็กต์ที่เลือก
	Front	นำอ็อบเจ็กต์ที่เลือกไว้ด้านบน
	Back	นำอ็อบเจ็กต์ที่เลือกไว้ด้านล่าง
	Flip	พลิกอ็อบเจ็กต์ที่เลือกตามแนวนอน
	Rotate	หมุนอ็อบเจ็กต์ที่เลือก 90 องศา ทวนเข็มนาฬิกา

ชุดเครื่องมือ (Tools Palette)




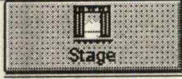
ชุดเครื่องมือประกอบด้วยอุปกรณ์ต่างๆ ที่ใช้สำหรับสร้างแผนงาน (Page) อุปกรณ์เหล่านี้เป็นอุปกรณ์เสมือน (Vistual) ที่ไม่สามารถเข้าถึงได้จากเมนู การใช้งานสามารถทำได้โดยการคลิกที่ อุปกรณ์ที่ต้องการแล้วนำมาวางบน บริเวณทำงาน

ไอคอน	คำสั่ง	คำอธิบาย
	Select	เลือกอ็อบเจ็กต์ที่ต้องการ
	Magnify	ขยายขนาดของอ็อบเจ็กต์
	Button	วาดปุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Label Button	สร้างปุ่มแบบป้ายชื่อ
	Radio Button	สร้างปุ่มเลือกแบบกลม
	Check Box	สร้างปุ่มเลือกแบบสี่เหลี่ยม
	Radio Button 3D	สร้างปุ่มเลือก 3 มิติแบบวงกลม
	Check Box 3D	สร้างปุ่มเลือก 3 มิติแบบสี่เหลี่ยม
	Field	สร้างฟิลด์ข้อความ
	Record Field	สร้างฟิลด์ข้อความแบบเรคคอร์ด (ใช้เฉพาะแบล็คกราวด์)
	Borderless Field	สร้างฟิลด์ข้อความที่ไม่มีขอบเขต
	List Box Field	สร้างฟิลด์ข้อความแบบรายการ
	Combo Box	สร้างคอมโบบ็อก
	OLE Container	สร้างออบเจกต์ที่เชื่อมโยงแบบ OLE
	Line	วาดเส้น
	Arc	วาดส่วนโค้งของวงกลม
	Angled Line	วาดเส้นตรงแบบต่อเนื่อง
	Curve	วาดเส้นโค้ง
	Rectangle	วาดสี่เหลี่ยมมุมฉาก
	Rounded Rectangle	วาดสี่เหลี่ยมที่มีมุมโค้ง
	Ellipse	วาดรูปวงรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Polygon	วาดรูปหลายเหลี่ยมที่มีด้านเท่ากัน
	Irregular Polygon	วาดรูปหลายเหลี่ยม
	Pie	วาดชิ้นส่วนของวงกลม
	Stage	วาดสเตจ (Stage)

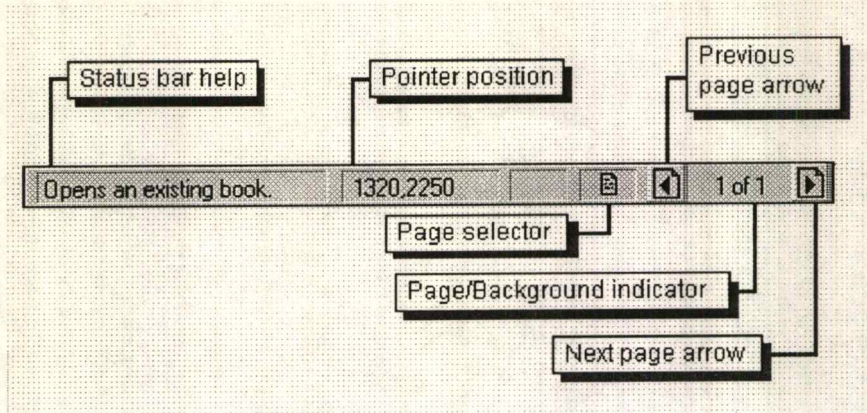
ระบบเมนูของ MTB 3

ใน MTB 3 ผู้ใช้งานสามารถเข้าถึงคำสั่งการทำงานได้หลายวิธี เช่น การเพิ่มหน้าสารบรรณทำได้โดยเลือก Object, New Page ได้จากเมนูหลักหรือทำการคลิกที่ไอคอนนิวส์เพจ (New Page) ในแถบเครื่องมือ หรือ พิมพ์คำสั่ง SEND New Page ในคอมมานด์วินโดวส์ (Command Windows) หรือกดปุ่ม **Ctrl+N** ซึ่งแต่ละวิธีจะทำให้ไดหน้าใหม่เพิ่มขึ้น

ระบบเมนูของ MTB 3 มีคุณลักษณะที่ควรทราบอยู่ 2 ประการ คือ ประการแรก เมนูจะเปลี่ยนแปลงตามอ็อบเจกต์ที่โปรแกรมเมอร์เลือก เช่น ถ้าเลือกอ็อบเจกต์หลายชั้น คำสั่ง **Group** ในเมนูจะพร้อมทำงาน (ตัวอักษรเป็นสีดำ) แต่ถ้าเลือกอ็อบเจกต์ชั้นเดียว คำสั่ง **Group** ในเมนูจะไม่สามารถใช้ได้ (ตัวอักษรเป็นสีเทา) ประการที่สองเมนูสามารถเปลี่ยนแปลงได้ตามความต้องการของโปรแกรมเมอร์ทั้งในระดับรีดเคอร์ (Reader) และออเทอร์ (Author)

แถบบอกสถานะ (Status Bar)

แถบบอกสถานะเป็นแถบสีเทา ที่อยู่บริเวณด้านล่างของจอภาพ ใช้ประโยชน์ในการบอกข้อมูลข่าวสารเกี่ยวกับอ็อบเจกต์ต่างๆ ใน MTB ตำแหน่งของเมาส์เคอร์เซอร์ และยังสามารถควบคุม, เลือกเพงที่ต้องการให้แสดง



รูป 23 แถบบอก สถานะของ MTB 3

ข้อมูลในแถบสถานะจะเปลี่ยนไปตามตำแหน่งของเมาส์เคอร์เซอร์ว่าชื่ออยู่ที่อ็อบเจกต์ใด โปรแกรมเมอร์สามารถควบคุมให้ แสดง/ซ่อน แถบบอกสถานะนี้ได้ทั้งจากคำสั่งในเมนู และ คำสั่งในสคริปต์

ตารางการเปรียบเทียบความสามารถ ToolBook กับซอฟต์แวร์อื่นๆ

	MTB	Visual Basic 3.0	Director 4.0	Authorware
OLE Support	Yes	Yes	No	Yes
RTF Support	Yes	No	No	Yes
Embedded True Type Fonts	Yes	No	No	No
Hot Words	Yes	No	No	Yes
dBase/Paradox engines	Yes	Yes	No	No
Spell Checker	Yes	No	No	Yes
Full text search and retrieval	Yes	No	No	Yes
Path-based animation	Yes	No	No	Yes
Transition effects between				
AVI,MCI and non-MCI	Yes	No	Yes	Yes
VBX controls	Yes	Yes	No	No

ตาราง 6 การเปรียบเทียบความสามารถ MTB กับซอฟต์แวร์อื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

Multimedia ToolBook CBT Edition

ToolBook CBT เป็นซอฟต์แวร์ที่เพิ่มความสามารถของ ToolBook 4.0 ในการสร้างแอปพลิเคชัน ที่ใช้สำหรับการฝึกฝนบุคลากร

Asymetrix ได้เปิดตัวซอฟต์แวร์ Multimedia ToolBook CBT Edition พร้อมกับ ToolBook 4.0 ซึ่งความจริงแล้วชุด ToolBook CBT เป็นซอฟต์แวร์ที่เพิ่มขยายความสามารถของ ToolBook 4.0 ในเรื่อง การสร้างแอปพลิเคชัน ที่ใช้สำหรับฝึกฝนการพัฒนาของ ToolBook นั้นมีมาอย่างต่อเนื่อง ToolBook เป็นซอฟต์แวร์ที่มีประสิทธิภาพสูง สามารถใช้สร้างแอปพลิเคชันที่ทำงานบนวินโดวส์ โดยมีอินเตอร์เฟซหลายอย่างที่มีประสิทธิภาพ ในการจัดการกับคำคำเบสและเท็กซ์ รวมถึงการใช้ มัลติมีเดียกราฟิก เอนิเมชัน วิดีโอ ออดีโอประกอบในแอปพลิเคชันได้ดีแอปพลิเคชัน ที่พัฒนา สามารถ ใช้ได้กับ วินโดวส์ ออบเจกต์ทุกชนิด เช่น เมนูบาร์ icoะลือกบ็อกซ์ คอมโบบ็อกซ์ บัททอนสามมิติ เป็นต้น นอกจากนี้ยังสามารถใช้คอลโทรลและอินเตอร์เฟซเฉพาะกับแพลตฟอร์ม แต่ละชนิด เช่น วินโดวส์ 3.X, 95 หรือ เอ็นทีได้ด้วย

ความต้องการของระบบที่จะใช้ ToolBook CBT

- CPU ต่ำสุดต้องเป็น 80386SX - 20 Mhz แต่จะให้ดีกว่าใช้ 80486 ขึ้นไป
- หน่วยความจำอย่างน้อย 8 เมกะไบต์แต่ควรใช้อย่างน้อย 12 เมกะไบต์จะดีกว่า
- ฮาร์ดดิสค์อย่างน้อยควรมีเนื้อที่ว่าง 25 เมกะไบต์ แต่ถ้าติดตั้งเต็มรูปแบบต้องใช้มากกว่านี้
- จอภาพอย่าง น้อยต้องแสดงได้ 256 สี ที่ความละเอียด 680 * 480 พิกเซล
- ต้องมีเมาส์ ซีดีรอมไดร์ และควรมีชาวนด์การ์ดติดตั้ง
- มีวินโดวส์ 3.1 หรือ 95 อยู่บนเครื่อง

การจัดการวิดีโอ กราฟิก และเสียง

การนำเอาวิดีโอ กราฟิก และเสียงมาใช้ในทูลบุ๊กทำได้ง่ายโดยสามารถสร้างแอปพลิเคชันแบบใช้เส้นทางเป็นหลัก โดยไม่จำเป็นต้องเขียนสคริปต์ หรือ จะใช้วิดีโอคลิป สามารถใช้งานในลักษณะของ วิดีโอ สำหรับ วินโดวส์ได้อย่างเต็มรูปแบบการใช้สามารถที่จะเล่นกราฟิกและมัลติมีเดียอื่นๆ ได้พร้อมๆ กันภายในหน้าต่างเดียว สามารถปรับขนาดและเลื่อนตำแหน่งหน้าต่างได้ตลอดเวลา ใช้แคร์ริก และ ครีอบ การควบคุมชาวนด์ของทูลบุ๊กจะมีตัวควบคุมเสียงภายในตัว สามารถควบคุมการเล่นได้ไปพร้อมๆ กับการเล่น วิดีโอ หรือ เอนิเมชัน โดยควบคุมจากเวลาเป็นหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะที่ช่วยให้พัฒนาแอปพลิเคชันได้เร็วขึ้น

ทุกบีมมีโครงสร้างหลายอย่างที่จะช่วยให้ผู้ใช้สร้างผลงานได้เร็วขึ้น ได้แก่

- มี Auto - Script ที่ผู้ใช้สามารถเลือกได้จากไลบรารีของออบเจกต์สคิปต์.
- ผู้ใช้จะพัฒนาแอปพลิเคชันได้เร็วเนื่องจากมีสคิปต์เอคิเตอร์แบบบิวอิน และมีตัวตรวจสอบ

ความถูกต้องของสคิปต์ดีบั๊กเกอร์ให้

- มี Script Recorder ซึ่งช่วยบันทึกสิ่งที่ผู้ใช้ต้องการสร้างเป็นสคิปต์ให้
- มีหน้าต่าง Command Windows ที่ผู้ใช้สามารถพิมพ์คำสั่ง OpenScript ลงไปให้ทำงานบน

อินเทอร์เน็ตแอคทีฟทันที

- มี Object browser ที่ใช้แสดงออบเจกต์ทั้งหมดที่อยู่ในบุ๊กแบบโครงสร้าง ไชราคี ผู้ใช้จะเลือกแก้ไข หรือลบออบเจกต์ได้ทันที
- Shared Script เป็นลักษณะที่ ออบเจกต์หลายๆมีความสามารถคล้ายกันสามารถใช้ร่วมกันได้โดยไม่ต้องเขียนใหม่ทุกครั้ง
- Property Editor ช่วยอำนวยความสะดวก ในการจัดการพรอพเพอร์ตี้ต่างๆของออบเจกต์
- Automated Book Specialist ที่จะช่วยสร้าง โครงสร้างของแอปพลิเคชันพร้อมทั้งงาน ที่มีจะถูกใช้ บ่อยๆ ให้ เหมาะสำหรับผู้ที่ใช้และมีความชำนาญแล้ว
- ใช้ร่วมกับ Internet browser โดยสามารถกำหนด URLs ได้จากภายในแอปพลิเคชันที่ สร้างจากทูลบุ๊กเอง

การสร้างแอปพลิเคชันทำได้ดีขึ้น

ToolBook 4.0 สามารถเรียกใช้คอนโทรลของวิซวลเบสิก (VBX) ผู้ใช้สามารถอิมพอร์ต VBX ต่างๆ มาใช้ เช่น ลิสต์บ็อกซ์ ไทม์เมอร์ เกจ กริด เป็นต้น แอปพลิเคชันจึงสามารถสร้างได้ อย่างมีประสิทธิภาพ มากขึ้น นอกจากนี้ยังสามารถใช้ DDE , OLE และ DLL ผู้ใช้จะสามารถแลกเปลี่ยน หรือเรียกใช้ออบเจกต์ของแอปพลิเคชันชนิดใดๆก็ได้ที่มีอยู่บนวินโดวส์เข้ามาใช้ในแอปพลิเคชันได้ทันที

ภาษา OpenScript

OpenScript เป็นภาษาที่ใช้เขียนโปรแกรมได้ มีคำสั่งให้ใช้งานได้หลายรูปแบบ ใช้สร้างและจัดการ ออบเจกต์ โดยสามารถเชื่อมโยงและเรียกใช้ฟังก์ชันได้จาก DLL ของ Windows คำสั่งของ OpenScript จะเหมือนกับภาษาอังกฤษธรรมดา สามารถเข้าใจได้ง่ายมีลักษณะของออบเจกต์โอเล็นเต็ค ผู้ใช้สามารถ สร้างแอปพลิเคชัน ToolBook และใช้ OpenScript ที่มีประสิทธิภาพที่ทัดเทียมกับของ C , C++ หรือ Visual BASIC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฮเปอร์เท็กซ์และดาต้าเบสได้ภายในแอปพลิเคชัน

การสร้างมัลติมีเดียแอปพลิเคชันมีความจำเป็นต้องใช้ดาต้าเบสและเท็กซ์มาก ทูลบุ๊กสนับสนุนการใช้งานเท็กซ์ชนิด Rich - Text format (RFT) ผู้ใช้สามารถอิมพอร์ตเท็กซ์ชนิดนี้ที่สร้างขึ้นสามารถกำหนดสี ใช้ตัวก ตัวห้อย มีกราฟฟิกผสมได้ มีเอนจินสำหรับใช้ค้นหาคำเดียวหรือสร้างเป็นเงื่อนไขเพื่อค้นหากลุ่มคำก็ได้ นอกจากนี้ยังสามารถตรวจสอบคำผิดได้ในตัวด้วย

ToolBook มีความสามารถจัดการเกี่ยวกับดาต้าเบสได้ดีพอสมควร ผู้ใช้สามารถผนวกข้อมูลจากดาต้าเบสประเภทต่างๆเข้ามาใช้ในมัลติมีเดียแอปพลิเคชันได้ เช่นประวัติลูกค้าแต่ละรายรายการสินค้าไม่ว่าจะเป็นร้านเล็กๆหรือธุรกิจขนาดใหญ่ก็ตาม ทูลบุ๊กมีวิวัฒนาการเอนจินของ Prodox และ bBASE

III

ให้ใช้ได้ทันที หรือจะเรียกใช้ผ่าน ODBC ก็ได้

Course Management System (CMS)

Course Management System เป็นระบบที่ใช้ใน ToolBook CBT ที่ช่วยจัดการหลักสูตรการเรียนของนักเรียนแต่ละคน ระบบดังกล่าวนี้จะช่วยให้ผู้ใช้ประหยัดเวลาในการพัฒนาแอปพลิเคชันลักษณะนี้ได้เร็วขึ้น CBS ประกอบด้วยยูทิลิตี้ 3 ตัวคือ

1. ADMIN.EXE เป็นยูทิลิตี้ซึ่งใช้จัดการหลักสูตรการเรียน โปรแกรมจะล็อกอินใช้งานโดยผู้ดูแลหลักสูตร ซึ่งจะเป็นผู้กำหนดหลักสูตรทั้งหมด กำหนดรายชื่อนักเรียนที่สามารถเข้ามาใช้งาน โปรแกรมดูแลข้อมูลที่สำคัญต่างๆของหลักสูตร

2. STUDENT.EXE เป็นยูทิลิตี้สำหรับนักเรียนแต่ละคนในหลักสูตรจะล็อกอินด้วยพาสเวิร์ดเฉพาะ นักเรียนแต่ละคนจะมีข้อมูลเฉพาะ สามารถบันทึกผลการเรียน และความต่อเนื่องในการใช้คอมพิวเตอร์ได้ โดยมีผู้ดูแลระบบเป็นคนจัดการข้อมูลทั้งหมด

3. DECRYPT.EXE เป็นยูทิลิตี้ที่ใช้สำหรับถอดรหัสล็อกไฟล์ ซึ่งเป็นไฟล์ที่ใช้เก็บข้อมูลความก้าวหน้าในการเรียนของนักเรียนแต่ละคนของหลักสูตร

Widget อุปกรณ์สำคัญในการพัฒนาซอฟต์แวร์หลักสูตร

Widget เป็นเครื่องมือที่ช่วยในการสร้างคอร์สแวร์ให้ง่ายขึ้น ความจริง Widget ก็คือออบเจกต์ชนิดหนึ่งซึ่งถูกเตรียมมาให้ประกอบด้วยสคริปต์และพรอพเพอร์ตี้ต่างๆที่ผู้ใช้งานสามารถเรียกมาใส่เพิ่มไว้ในบู้คที่สร้างขึ้นได้โดยไม่ต้องเขียนโปรแกรมเพิ่มเติม เช่น ชุดคำถามที่ใช้สำหรับทดสอบนักเรียนซึ่งมีสคริปต์ที่พร้อมจะรับสิ่งที่นักเรียนตอบมาตอบสนองด้วยผลลัพธ์ และ ตรวจสอบคะแนนที่ได้ด้วย

ToolBook CBT เตรียม Drag and Drop Widget มาให้ใช้มากกว่า 200 ชนิด โดยแยกประเภทได้คือ Question Type , Action Buttons , Bookmark , 3D layout elements , Data Validation , Media Clip, Navigation , Responsre Checking , Special Fields , Tools and Examples สำหรับ Question Type สามารถใช้สร้างคำถามได้หลายแบบเช่น ถูกผิด เลือกข้อที่ถูกที่สุด จับคู่ เติมคำในช่องว่าง เป็นต้น พารามิเตอร์ที่ใช้ของคำถามอาจเป็นคะแนน คำถามที่ถ่วงน้ำหนัก ค่าสูงสุดเวลาที่จำกัดในการตอบคำถามและการสุ่มคำถามนอกจากนี้ผู้ใช้งานยังสามารถดัดแปลง Widget ที่มีอยู่ให้มีรูปแบบตามที่ต้องการและเก็บไว้ใช้ได้ภายหลัง

มีอะไรใหม่ใน ToolBook 4.0 CBT

- ประสิทธิภาพในการทำงานเร็วกว่าเวอร์ชัน 3.0 ถึง 50 เปอร์เซ็นต์
- แอปพลิเคชันที่สร้างขึ้นมีอินเตอร์เฟสให้เลือกใช้อย่างเหมาะสมกับวินโดวส์แพลตฟอร์ม โดยเฉพาะวินโดวส์ 95
- เพิ่มประสิทธิภาพในการสร้างแอปพลิเคชันด้วยการนำคอลโทรลของวิซวลเบสิก (VBX) เข้ามาใช้ได้
- แอปพลิเคชันที่สร้างมีลักษณะออบเจกต์โอเรียนเต็ลมากขึ้น เนื่องจากสามารถใช้สคริปต์ร่วมระหว่างออบเจกต์ได้
- มี Palette Optimizer ที่ช่วยลดความคลาดเคลื่อนของสีระหว่างมัลติมีเดียออบเจกต์
- มี Object browser ที่แสดงออบเจกต์ทั้งหมดในแอปพลิเคชันในลักษณะโครงสร้างไฮราคี
- มี Automted Book specillist ช่วยลดขั้นตอนในการสร้างบู้คใหม่ง่ายขึ้น
- Script Remover เป็นยูทิลิตี้ที่ช่วยดึงส่วนที่เป็นเท็กซ์ซึ่งเก็บอยู่ในแอปพลิเคชันออกได้แก่ ส่วนสคริปต์ทั้งหมด แอปพลิเคชันที่ผ่านกระบวนการจะเหลือแต่โค้ดที่พร้อมจะทำงานเท่านั้น ช่วยป้องกันไม่ให้คนอื่นมาเรียกดูหรือแก้ไขสคริปต์ที่เขียนไว้ และยังช่วยลดขนาดไฟล์ให้เล็กลงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Course Management System จะช่วยในการพัฒนาซอฟต์แวร์ช่วยสอนในหลักสูตรการศึกษา
ใด ๆ ให้ทำได้อย่างมีประสิทธิภาพและเสียเวลาน้อย Widget ที่ทูลบู้คเตรียมมาให้มีมากกว่า 200
ชนิด สามารถดึงเข้ามาใช้ได้ทันที
- สามารถเรียกใช้เวบเพจด้วย Internet Widget และเวบบราวเซอร์ที่กำหนด

ตารางการเปรียบเทียบความสามารถ ToolBook กับซอฟต์แวร์อื่นๆ

	MMTB 4.0	Visual Basic 3.0	Director 4.0	Authorware
OLE Support	Yes	Yes	No	Yes
RTF Support	Yes	No	No	Yes
Embedded True Type Fonts	Yes	No	No	No
Hot Words	Yes	No	No	Yes
dBase/Paradox engines	Yes	Yes	No	No
Spell Checker	Yes	No	No	Yes
Full text search and retrieval	Yes	No	No	Yes
Path-based animation	Yes	No	No	Yes
Transition effects between AVI,MCI and non-MCI	Yes	No	Yes	Yes
VBX controls	Yes	Yes	No	No

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบความสามารถ ToolBook CBT กับซอฟต์แวร์อื่นๆ

	MMTB CBT 4.0	AuthorWare 3.0	IconAuthor 7.0	Quest 5.0
Authoring model	Book/Page Even Driver	Flow Chart Programming	Flow Chart Programming	Flow Chart Programming
TrueWYSIWYG authoring customizable specialists create new applications	Yes	No	No	Yes
course Management system	Yes	No	No	No
Object Oriented Full featured ,English-like scripting language	Yes OpenScript	Yes No(limited)	No No	Yes No
DLL/DEE support	Yes	Yes	Yes	Yes
OLE support	Yes	Yes	Yes	No
VBX control supported	Yes	No	No	No
direct import RTF text files	Yes	Yes	Yes	No
Embedded Ture Typ Fonts	Yes	No	No	No
Bulit-in database support	Yes	No	No	No
auto-scripting tool	Yes	No	No	No
Customizable script language	Yes	No	No	No

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Optimize for CD-ROM				
Playback	Yes	No	No	No
Hypertext support	Yes	Yes	No	Yes
Free runtime	Yes	Yes	Yes	Yes



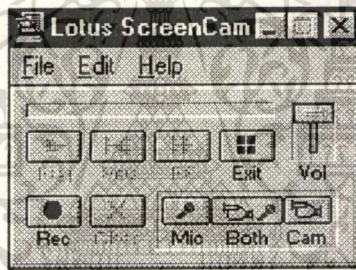
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7











ScreenCam 96

ScreenCam เป็นแอปพลิเคชันที่สามารถบันทึกภาพเหตุการณ์ต่างๆ ที่เกิดขึ้นบนเดสก์ทอปของวินโดวส์ไม่ว่าจะเป็นการเลื่อนพอยท์เตอร์ การคลิก และ ดับเบิลคลิก ภาพการปิดเปิดโปรแกรม

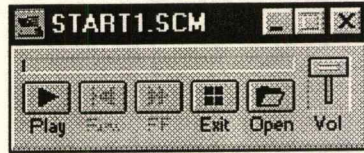
ทุกอย่างที่เกิดขึ้นบนจอภาพ และสามารถบันทึกเสียงบรรยายต่างๆ เพิ่มเข้าไปได้ด้วย ซึ่งจะเก็บอยู่ในรูปของไฟล์ .scm เพื่อสามารถที่จะนำกลับมาแสดงใหม่ในภายหลังได้ด้วยโปรแกรม ScreenCam Player ScreenCam ใช้เป็นแอปพลิเคชันที่อยู่ในชุดของโปรแกรมของ Lotus Smartsuit 96 ในการเก็บภาพเหตุการณ์บนเดสก์ทอปของวินโดวส์ 95 จำเป็นที่ต้องใช้ โปรแกรม Scwwncam 96 หรือใช้เวอร์ชันที่ใหม่กว่า เพราะ โปรแกรม ScreenCam เวอร์ชันก่อนหน้านี้ยังไม่สามารถทำงานบนวินโดวส์ 95 ได้แต่สามารถทำงานบนวินโดวส์ 3.1 ได้



รูป 24 อินเทอร์เฟซของ ScreenCam 96

-  เริ่มต้นการแสดงผลภาพ
-  ย้อนกลับไปยังจุดเริ่มต้น
-  การเร่งการแสดงผลภาพ
-  ออกจากโปรแกรม
-  เริ่มต้นการบันทึกรายการ
-  ยกเลิกการบันทึกที่ผ่านมา
-  เลือกบันทึกเฉพาะเสียง
-  เลือกบันทึกทั้งภาพและเสียง
-  เลือกบันทึกเฉพาะภาพ
-  ควบคุมระดับเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 25 อินเทอร์เฟซของ Screencam Player



เริ่มต้นการแสดงผล



ย้อนกลับจุดเริ่มต้น



ออกจากโปรแกรม



เปิดไฟล์สำหรับการแสดง



ควบคุมระดับเสียง

การบันทึกภาพ


1. เริ่มต้นด้วยการเรียกโปรแกรม Screencam กดปุ่ม  เพื่อทำการเริ่มบันทึก หรือกดปุ่ม F12 (ในกรณีที่กำหนดให้เป็นคีย์เริ่มต้น)
2. เมื่อทำการบันทึกภาพเสร็จสิ้นให้กดปุ่ม Esc (ในกรณีที่กำหนดเป็นคีย์หยุด)

การเก็บข้อมูลลงไฟล์

เมื่อทำการบันทึกภาพแล้วให้เลือก คำสั่ง Save ในเมนู File เพื่อทำการบันทึกโดยการใส่ชื่อที่ต้องการแล้วกดปุ่ม OK การบันทึกสามารถเลือกคำสั่ง Save As เพื่อเลือกการบันทึกแบบนามสกุลเป็นแบบ .EXE ซึ่งสามารถนำกลับมาแสดงใหม่ได้โดยไม่ต้องเรียกใช้โปรแกรม Screencam Player

การแสดงผลโดยใช้โปรแกรม Screencam player

การแสดงผลโดยใช้โปรแกรม Screencam Player สามารถทำได้ 2 วิธี

1. เรียกโปรแกรม Screencam Player ขึ้นมาแล้วใช้คำสั่ง Oper ในเมนู File เลือก File ที่จะแสดง แล้วกดปุ่ม 
2. แสดงการใช้คอมมานโด Scrmply.exe < ตัวเลือก > (ชื่อไฟล์ .scm) ซึ่งโปรแกรมมีตัวเลือกดังนี้

ตัวเลือก

/S ช้อน Screencam Logo

/P รอให้ผู้กด Play ก่อนแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- /R แสดงภาพซ้ำ
- /C ปิดโปรแกรมแสดงเมื่อแสดงภาพจบ
- /H ซ่อนชุดควบคุมขณะแสดงภาพ
- /F หยุดภาพหลังการแสดง
- /Z จำนวนครั้งในการแสดง

การใช้งาน Screencam ร่วมกับ Multimedia toolbox

การเรียกใช้โปรแกรม Screencam เพื่อแสดงเหตุการณ์ สามารถทำได้โดยใช้คำสั่งคอมมานไลน์ เช่น Scplyer.exe /sc Screencam เป็นการสั่งให้แสดงไฟล์ชื่อ Screen.scm พารามิเตอร์ sc เป็นการสั่งให้ซ่อน LOGO และเปิดโปรแกรมเมื่อการแสดงสิ้นสุด

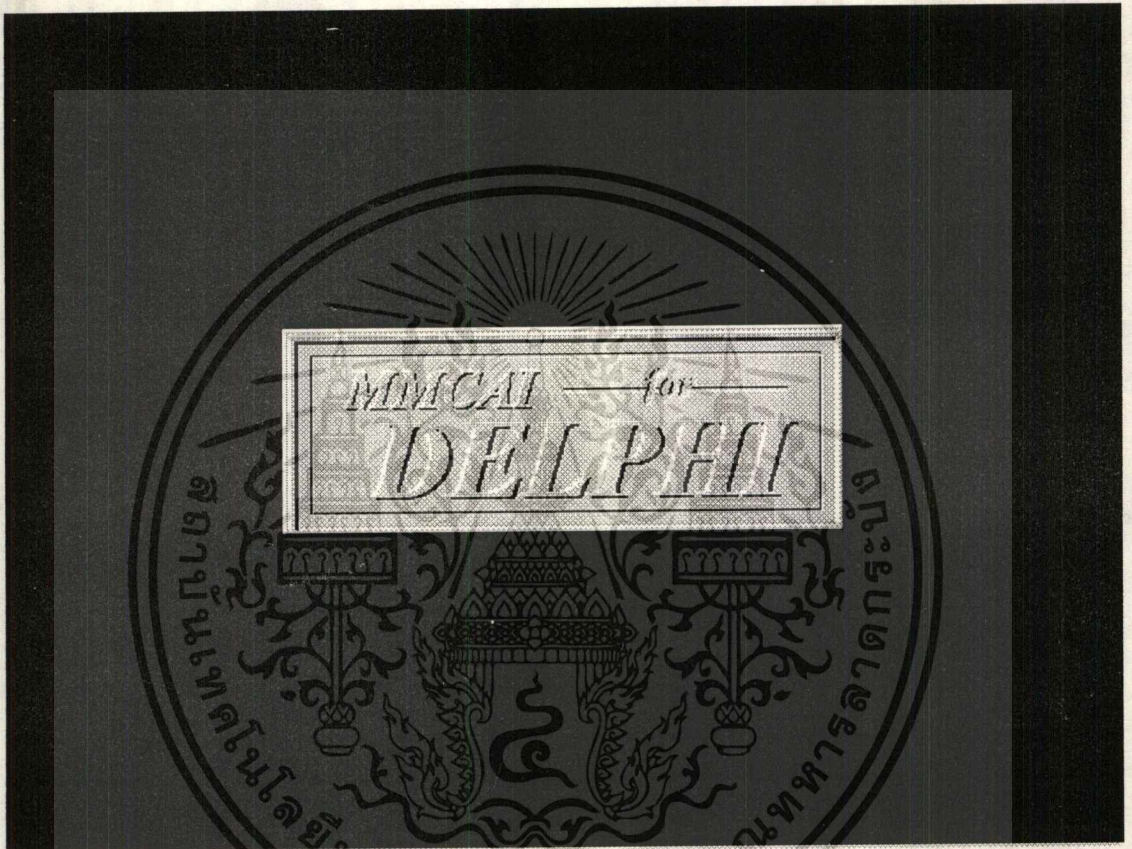
การเรียกใช้ Screencam บน MTB ทำได้โดยใช้คำสั่ง Run เช่น Run “ scplayer.exe /sc Screencam “ แต่ในการใช้งานจริงเมื่อทำคำสั่ง Run แล้ว MTB จะทำคำสั่งถัดไปโดยไม่มีการหยุดรอโปรแกรม ที่ถูกสั่ง Run เพื่อให้ MTB หยุด รอการทำงาน จึงต้องเพิ่มสคริปคำสั่งควบคุมดังนี้

```
linkDll"User"
word findWindow ( Dword,string )
end linkDll
linkDll. " TB30win.dll"
int YieldApp()
end linkDll
run " Scplyer.exe /sc Screencam.scm "
do
get YieldApp ( )
get Findwindow ( 0 ," Screen.scm ") { คำสั่งหาหมายเลขหน้าต่างที่ แอปพลิเคชัน }
until it = 0 { ชื่อ Screen.Scm ถ้าหาหน้าต่างไม่พบจะส่ง }
UnlinkDll " User " { ค่ากลับเป็น ศูนย์ซึ่งจะเก็บไว้ที่ตัวแปร it }
unlinkDll " TB30win.Dll "
```

บทที่ 8

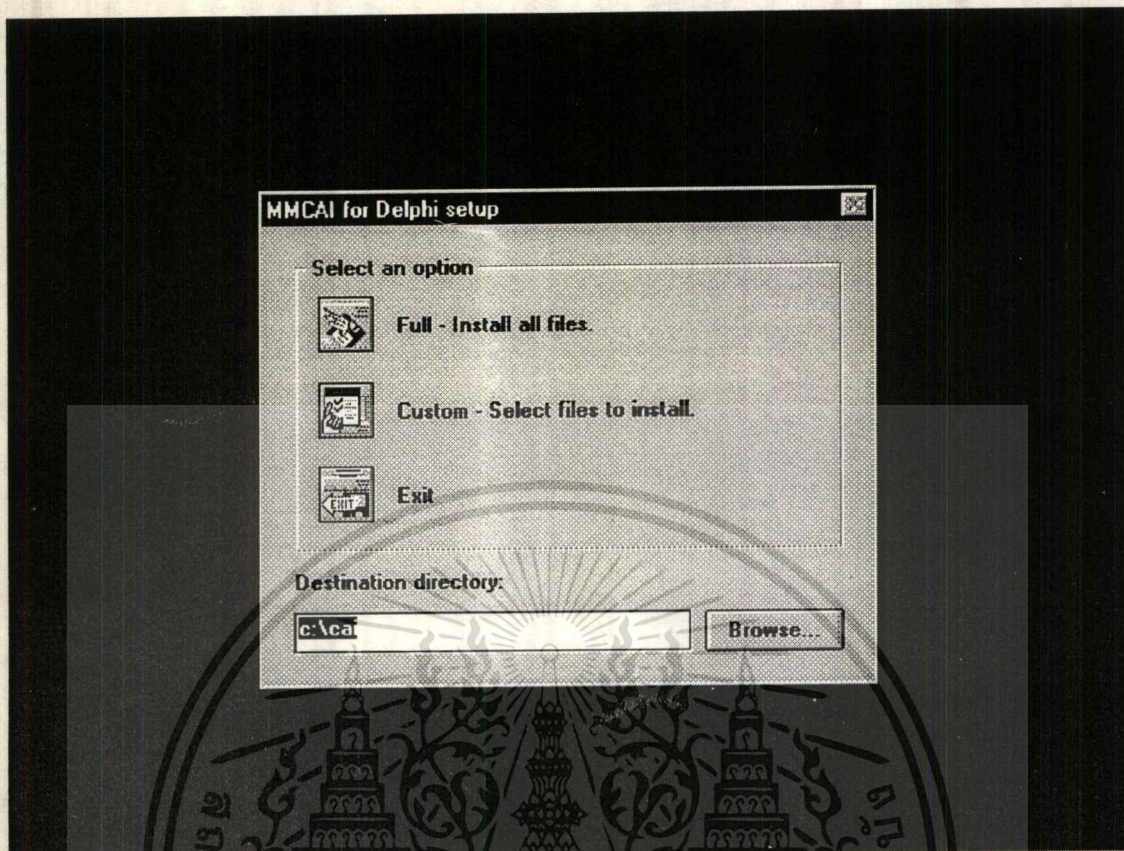
การติดตั้งโปรแกรม

ให้นำแผ่น CD ใสเข้าไปแล้วทำการเรียกคำสั่ง Setup จากนั้นโปรแกรมจะเริ่มทำการติดตั้งตัวโปรแกรมให้ โดยจะปรากฏ LOGO ของโปรแกรมขึ้นมาดังรูปข้างล่าง



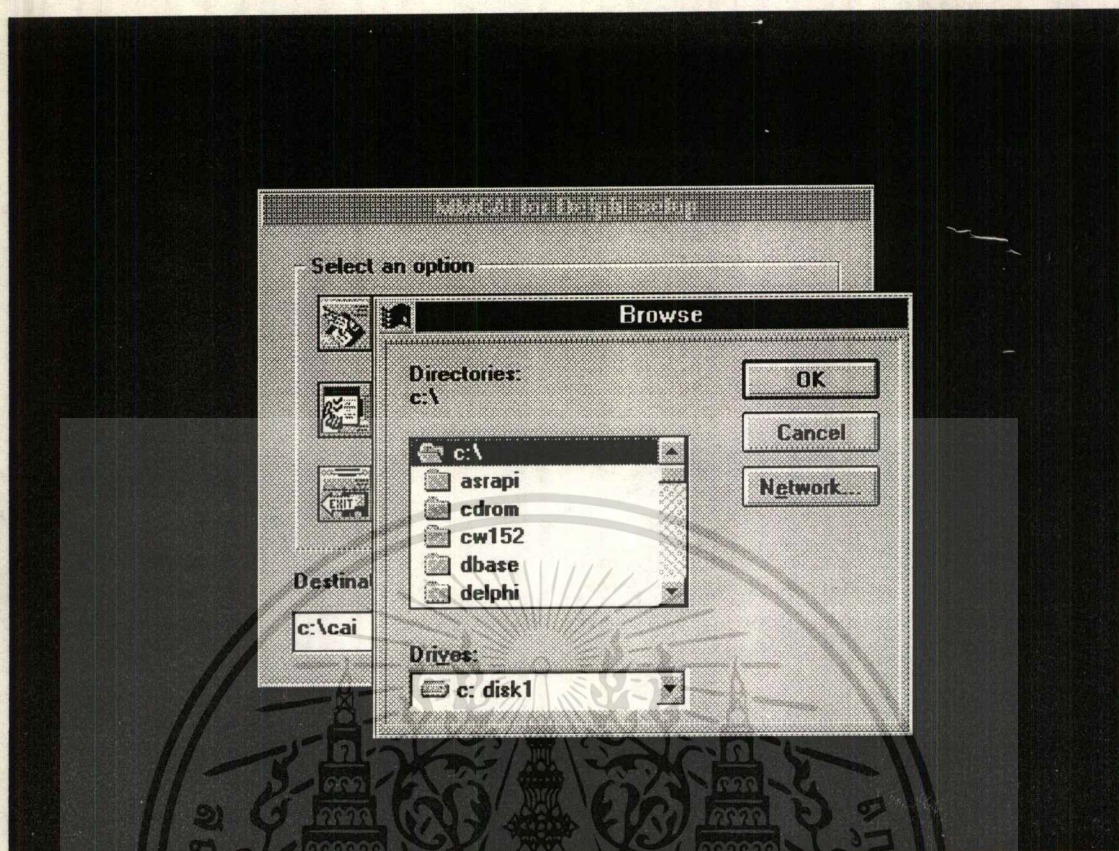
รูปที่ 26 รูป Logo ของตัวโปรแกรม

จากนั้น Logo จะหายไปแล้วจะปรากฏ หน้าจอต่อไปข้างล่างนี้อย่างอัตโนมัติ



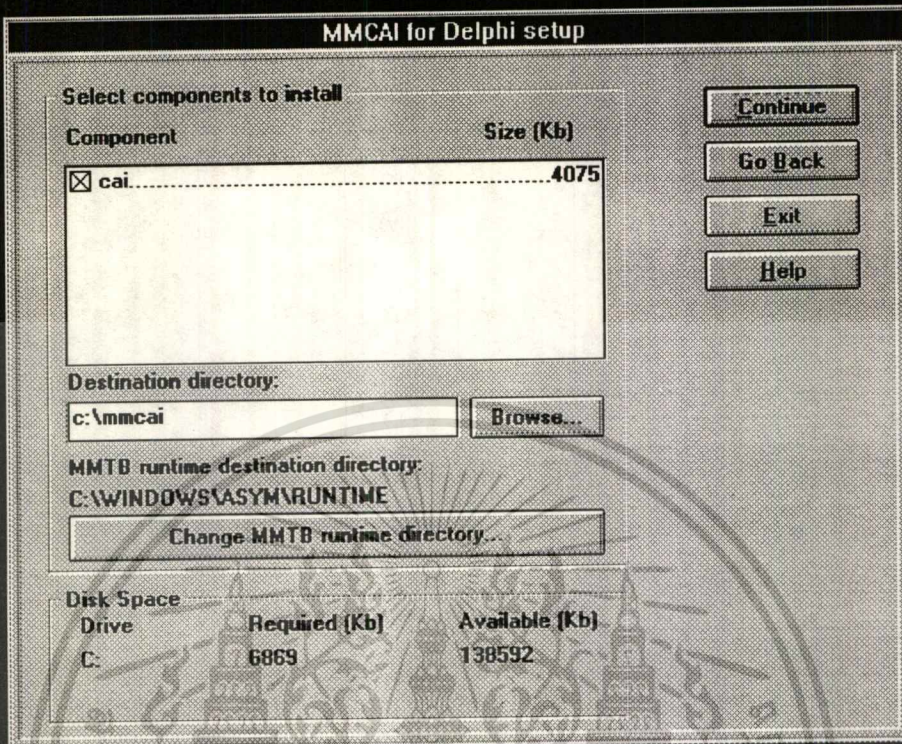
รูป 27 เลือกรูปแบบการติดตั้ง

จากนั้นถ้าไม่ต้องการให้อยู่ที่ Directory ที่โปรแกรมกำหนดมาให้ ให้กดปุ่ม Browse ปรากฏหน้าจอ ดังรูปที่ 28



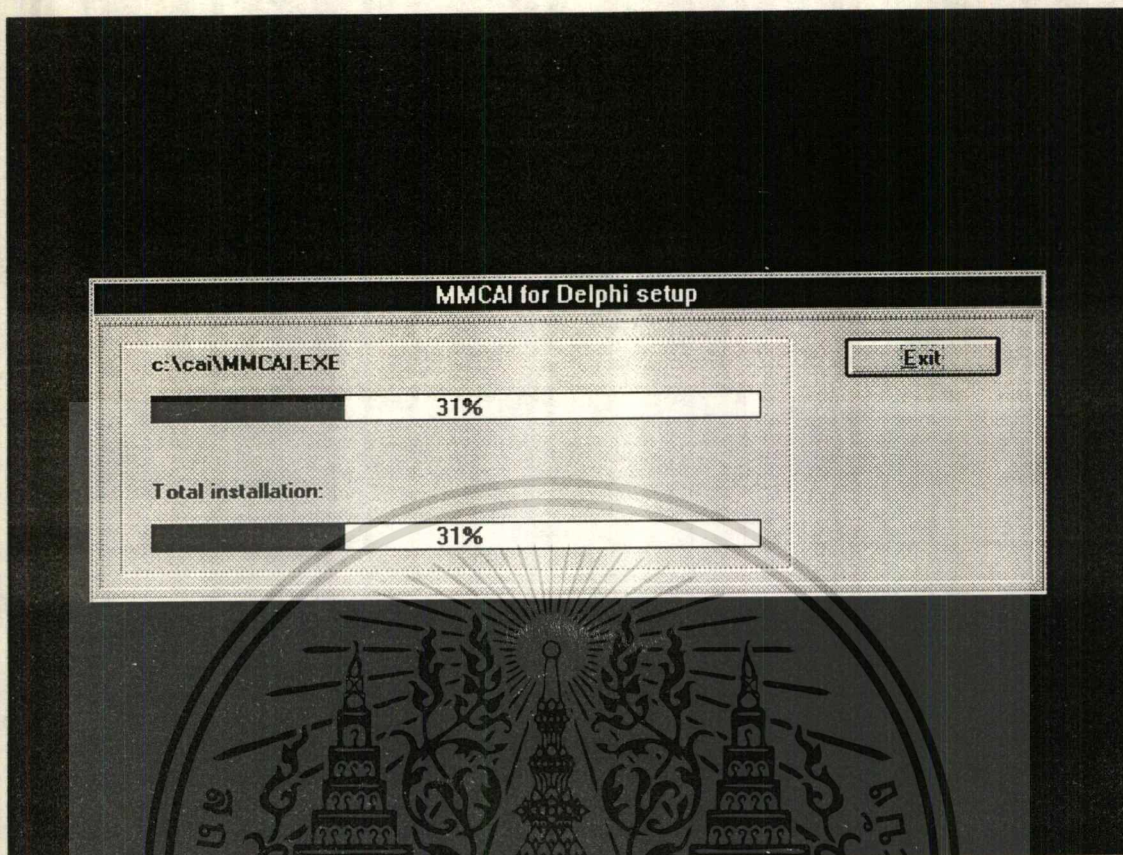
รูป 28 เลือก Directory

ให้เลือก Directory ที่ต้องการแล้วทำการกดปุ่ม OK จากนั้นจะปรากฏรูปภาพ
 ดังรูปที่ 27 อีกครั้งหนึ่ง เพื่อให้เลือกชนิดการ Setup หากเลือกแบบ Custom จะปรากฏหน้าจอตั้งรูป
 ที่ 29 แต่ถ้า เลือก Full จะปรากฏหน้าจอตั้งรูปที่ 30

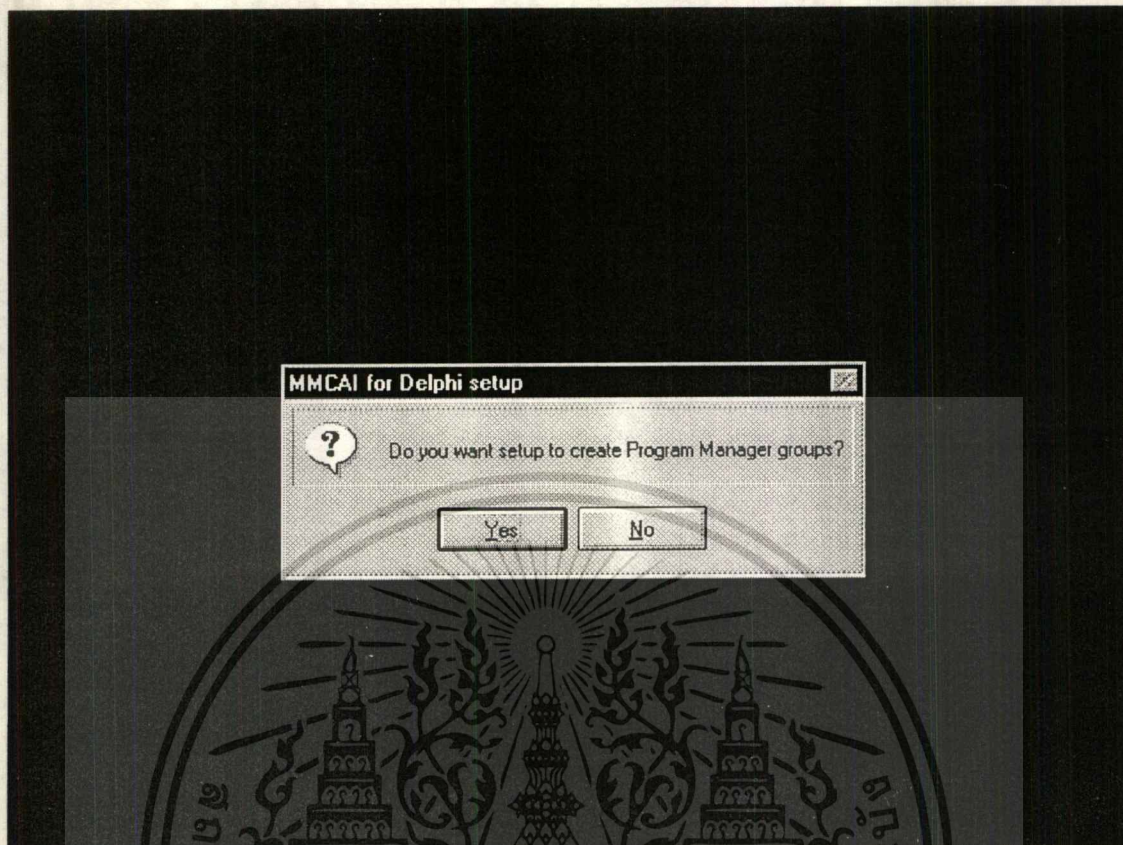


รูป 29 การเลือกแบบ Custom

ให้เลือกไฟล์ที่ต้องการ Setup แล้วทำการกดปุ่ม OK จะปรากฏดังรูปที่ 30

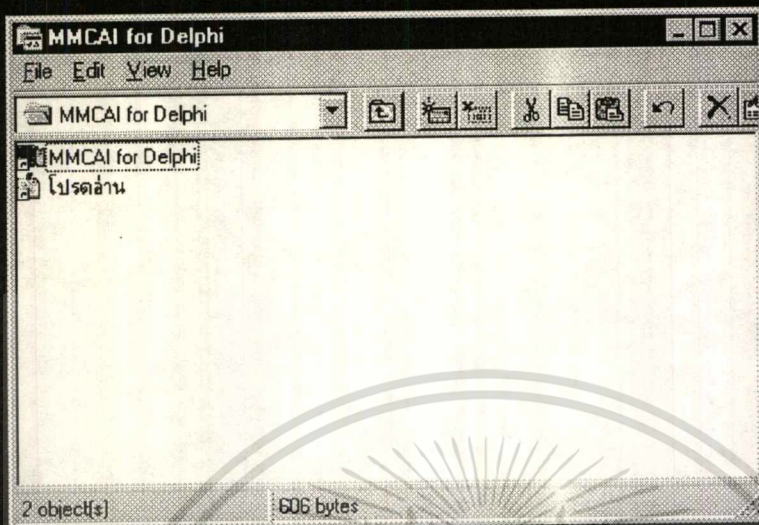


รูป 30 การติดตั้งโปรแกรม
เมื่อติดตั้งเรียบร้อยแล้วจะปรากฏหน้าจอต่อมา ดังรูปที่ 31



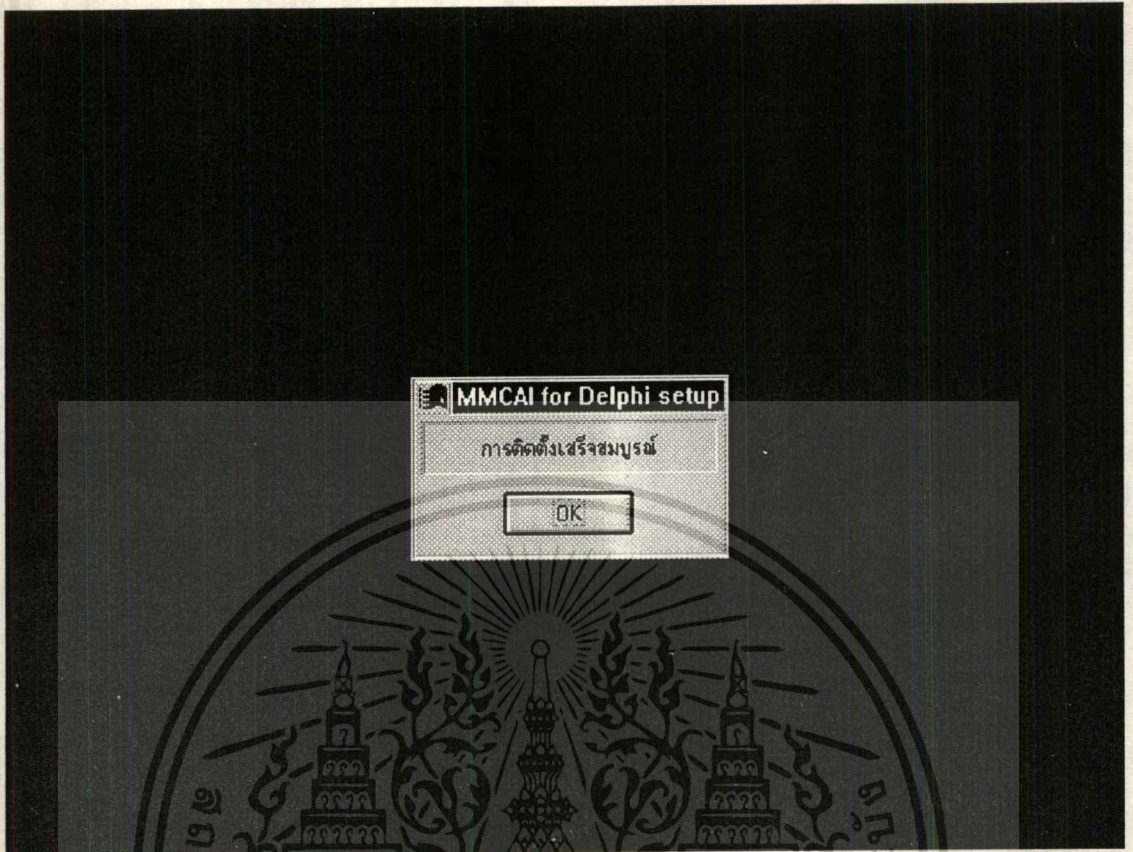
รูป 31 สร้าง Manager group

โปรแกรมจะถามว่า คุณต้องการสร้าง Manager group หรือไม่ ถ้าต้องการให้กดปุ่ม Yes จะปรากฏหน้าจอตั้งรูปที่ 32

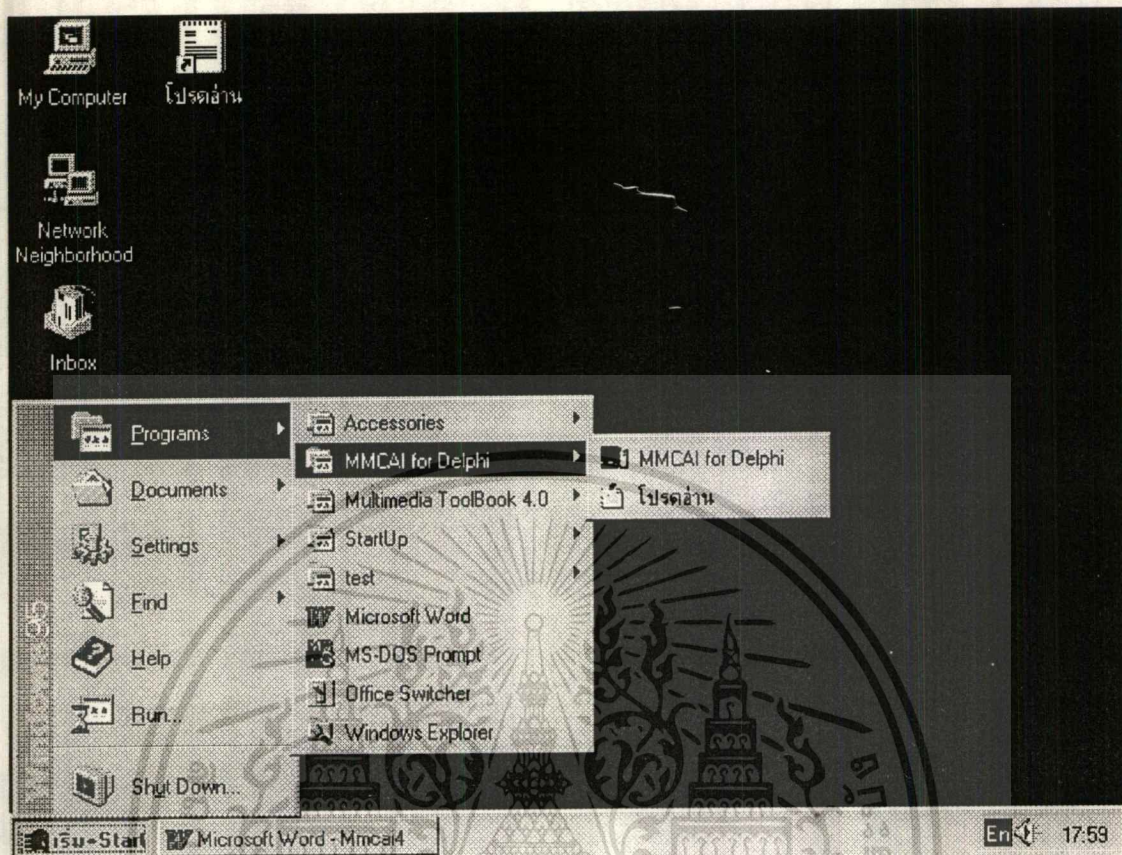


รูป 32 หน้าต่าง Manager group
จากนั้นหน้าจอจะเปลี่ยนจากรูปที่ 32 ไปเป็นรูปที่ 33 โดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 33 หน้าจอแสดงว่าการติดตั้งเสร็จสมบูรณ์
ให้ทำการกดปุ่ม OK เพื่อเป็นการเสร็จสิ้นการติดตั้งโปรแกรม และเราก็สามารถเรียกใช้
โปรแกรมจากตัว Windows ได้โดยดูจากรูปที่ 34



รูป 34 การเข้าโปรแกรม

รูปแบบการนำเสนอ

จะอ้างอิงการสอนจากหนังสือ Delphi Nuts & Bolts For Experienced Program และ Delphi ของ อ.บุญเลิศ เอี่ยมทัศนาศ ซึ่งจะเน้นหนักในผู้ใช้ระดับกลาง จนถึง ระดับสูง โดยในตอนที่ 2 จะทำการนำเสนอทั้งหมด 10 บท

บทที่ 1 ทำความรู้จักกับ Program Delphi

จะเป็นการแนะนำเกี่ยวกับการเข้าสู่โปรแกรม Delphi ในแบบต่างๆ และเมื่อเข้าสู่โปรแกรมของ Delphi แล้วจะทำการแนะนำให้รู้จักกับส่วนประกอบหลักต่างๆ คือ Form , Code Eder , Objinspector , Component Patte , Menu Bar , Tools , Title Bar ฯลฯ

บทที่ 2 เขียน Program Delphi อย่างง่าย

ในบทนี้จะอธิบายให้รู้จักการเขียนโปรแกรมอย่างง่าย เพื่อให้รู้จักการกำหนด Component-เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Patte , และการกำหนด Propoty , การป้อนโปรแกรม เพื่อควบคุมการทำงาน พร้อมทั้งทดสอบ การ Run โปรแกรม และการ Save ลง File ทั้งหมดเพื่อให้มองภาพรวมๆ ของโปรแกรม Delphi ให้ออก ถึงการเขียน และ ยังได้โปรแกรมที่สามารถใช้งานได้จริง เพื่อให้เห็นข้อดี ของ Delphi

บทที่ 3 Component ในส่วนของการ Control

จะอธิบายถึง Component ในส่วนของการ Control ซึ่งมีอยู่ด้วยกันหลายตัวได้แก่ Component ที่อยู่ในส่วนของ Additional และในส่วนของ Standard ยกเว้น Main menu และ Popuption โดยในบทนี้เราจะนำ Component Control มาบางส่วนเพื่อใช้งานให้ดู ส่วน ตัว Control อื่นๆ จะได้พบในบทต่อไป

บทที่ 4 กรอบรายการ

บทนี้จะเสนอให้รู้จักกับ กรอบรายการ ซึ่งตัวกรอบรายการ ก็คือ ตัวที่มีไว้เพื่อแสดง รายการต่างๆเอาไว้ข้างในกรอบ ซึ่งจำทำให้มีความเป็นระเบียบ และ สะดวกในการเรียกใช้งาน ซึ่ง เราจะพบเห็นตัวกรอบรายการได้บ่อยๆในตัวของ Windows เอง หรือ อาจจะเป็นกรอบ คอมโบ หรือ รายการที่เป็นโครงสร้าง และ อื่นๆ

บทที่ 5 ไลออตบ็อกซ์

การเรียนรู้ในบทที่ 5 จะกล่าวถึงการทำงานของ ไลออตบ็อกซ์ซึ่งความสามารถที่ดี ของไลออตบ็อกซ์ คือ การใช้งานง่าย และสามารถช่วยผู้เขียน โปรแกรมสามารถทำงานได้ง่ายมาก เราจะพบเห็นการใช้งานไลออตบ็อกซ์ ได้บ่อยๆในตัวของ Windows เอง

บทที่ 6 เมนู

ภายใน บทที่ 6 นี้เราจะกล่าวถึง การสร้างเมนู ซึ่งเมนู จะแบ่งเป็น 2 แบบ คือ Main Menu Popuption ซึ่งเราจะแสดงวิธีการทำงานของ ทั้ง 2 แบบ และเราจะเห็นถึงความง่ายในการเขียน โปรแกรม และการใช้งานก็สามารถทำได้อย่างง่าย เราจะพบเห็น เมนูได้บ่อยๆใน โปรแกรมต่างๆ

บทที่ 7 Data Base

ในบทที่ 7 เราจะกล่าวถึงการติดต่อกับ Data Base อย่างง่าย ซึ่งจะ เป็นประโยชน์ในการนำไปพัฒนาโปรแกรมเกี่ยวกับ Data Base ต่อไป โดยเราจะใช้ Data Base ที่ Delphi มีมาให้เรา ซึ่ง Data Base จะทำหน้าที่เก็บข้อมูลเท่านั้น แต่เราจะสร้างฟอร์ม ซึ่งจะ เป็น User Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้แจกจ่ายหรือเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8 ระบบ Multimedia

ในบทที่ 8 เราจะสอนการสร้างโปรแกรม เกี่ยวกับ Multimedia คือ จะมีการแสดงทั้ง ภาพนิ่ง ภาพเคลื่อนไหว การเรียกใช้ไฟล์ .WAV และ ไฟล์ .MIDI ซึ่งจะสามารถนำไปประยุกต์ใช้งานสร้างโปรแกรมที่เป็นระบบมัลติมีเดียได้อย่างมีประสิทธิภาพ

บทที่ 9 การ Debug โปรแกรม

การทำงานบทที่ 9 เราจะกล่าวถึงการ Debug โปรแกรม คือในการเขียนโปรแกรมหากเกิดการผิดพลาดเราจำเป็นต้องมาตรวจดูการทำงานของ โปรแกรมที่ละบรรทัดเพื่อดูว่าที่บรรทัดใดที่ทำให้เกิด Error ซึ่งก็คือการ Debug นั่นเอง

บทที่ 10 เบ็ดเตล็ด

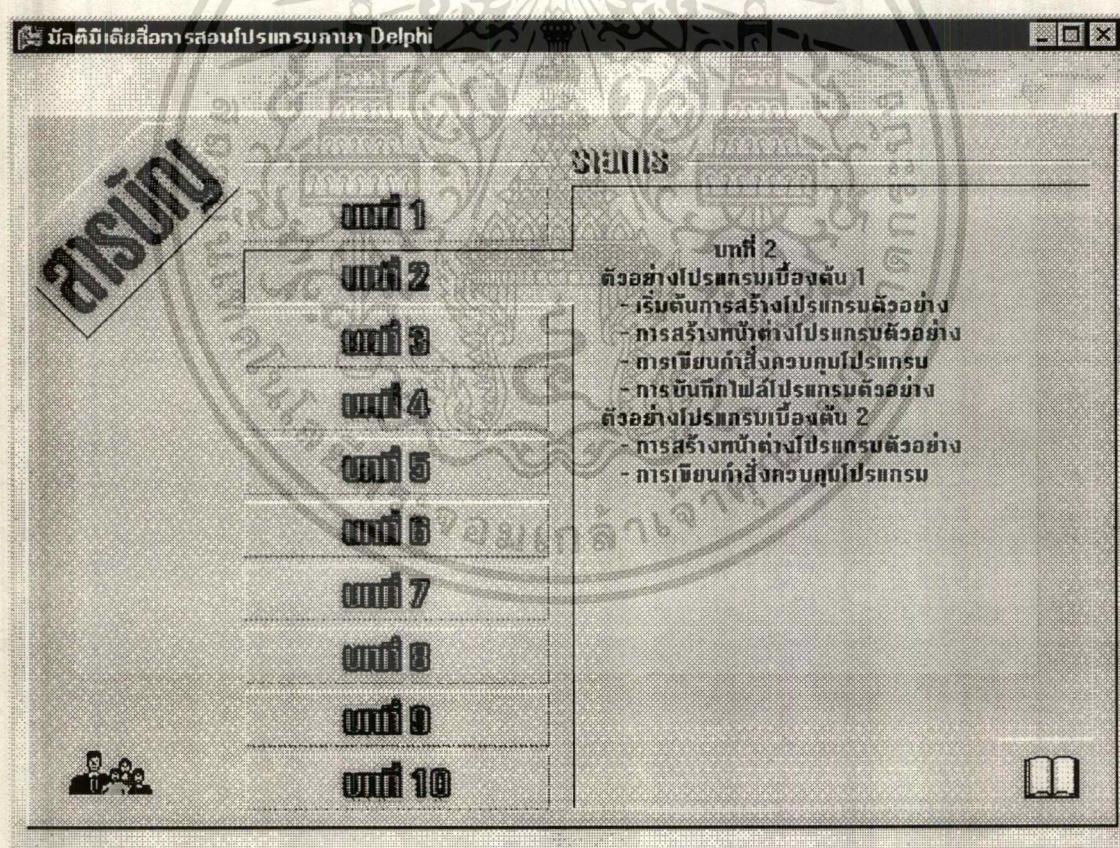
ในบทที่ 10 นี้จะเป็นการนำเทคนิคต่างในการเขียน โปรแกรมมารวมกัน เช่นการเพิ่มความสวยงามให้ กับ Form การกำหนด Icon ให้กับ Form และ อื่นๆ

ในการสอนจะใช้ Screen Crame เก็บหน้าจอ แล้วแสดงให้ดูพร้อมมีเสียงอธิบายที่ละขั้นตอน และเมื่อจบบทก็จะมีคำถามท้ายบทเพื่อทดสอบความเข้าใจของผู้ใช้ และจะเก็บคะแนนเอาไว้เมื่อเรียนจบก็จะมีการประมวลผลคะแนนว่าผู้ที่ใ้มีความเข้าใจ ในระดับใด ก่อนเริ่มต้นแต่ละบทจะมีการพูดแนะนำก่อนที่จะสอนแต่ละบทว่าจะ สอนเกี่ยวกับเรื่องอะไรต้องเตรียมตัวอย่างไรบ้างในการเรียนแต่ละบท

บทที่ 9

การทำงานของโปรแกรม MMCAI

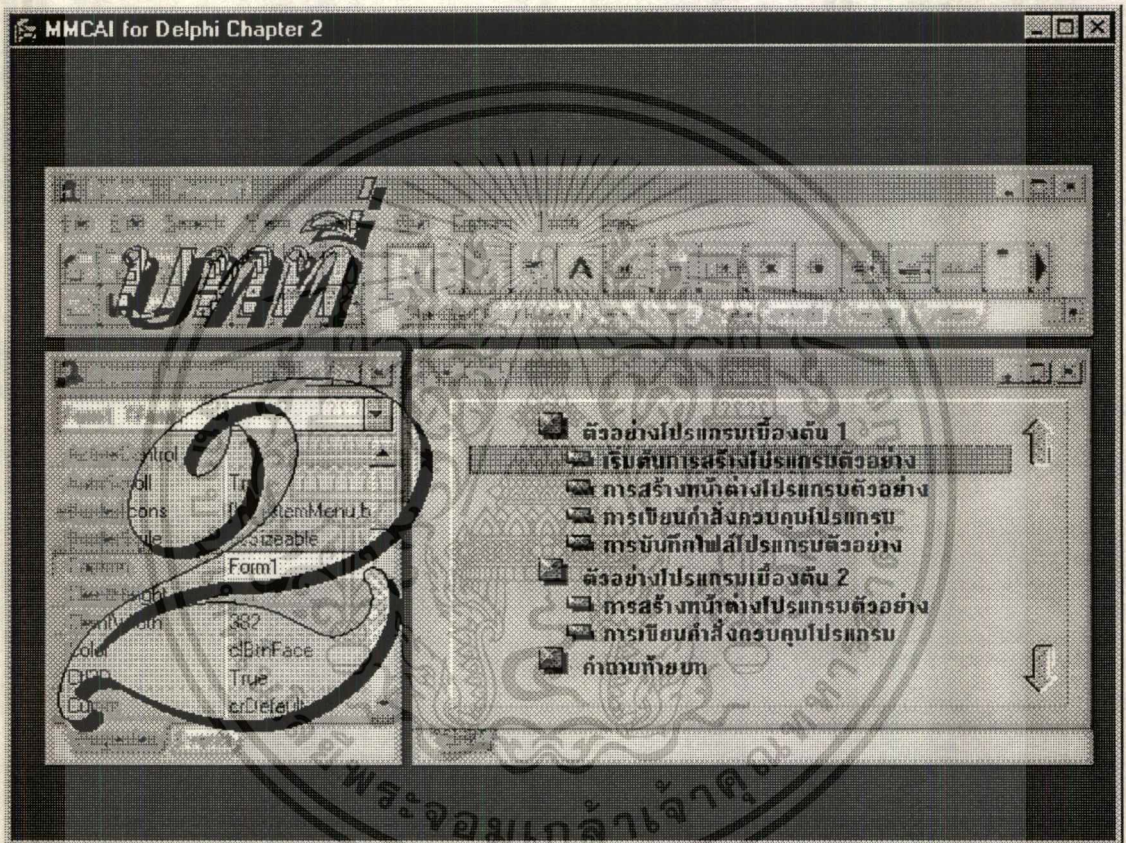
จากการทำงานเมื่อ ทำการ Setup แล้วจะสามารถเรียกใช้โปรแกรมมัลติมีเดียเพื่อการสอน โปรแกรมภาษา Delphi ได้ เมื่อเรียกโปรแกรมให้ทำงานจะปรากฏหน้าจอ รูป LOGO จากนั้นจะขึ้นหน้าจอ สารบัญ ดังรูปข้างล่างโดยจะมีปุ่มการทำงานโดยการเลื่อน Mouse ผ่านที่ตัวบทต่างๆ จะมีการเปลี่ยน จากรูปสมุดสีน้ำเงินเป็นสมุดสีแดง เพื่อแสดงว่า ขณะนี้ Mouse อยู่ที่ตำแหน่งนี้ โดยพร้อมกันนั้นใน Windows ด้านขวามือจะปรากฏรายละเอียดของแต่ละบทที่ Mouse ลากผ่าน และจะมี Icon รูปสมุดเปิดทางด้านล่างขวามือของหน้าจอ จะเป็นตัวที่ทำการออกจากโปรแกรมเมื่อมีการคลิกจะจบการทำงานแต่ถ้าต้องการเข้าสู่บทเรียนให้ทำการคลิกที่บทที่ต้องการ ในที่นี้เราเลือกกดที่บทที่ 2



รูป 35 สารบัญ

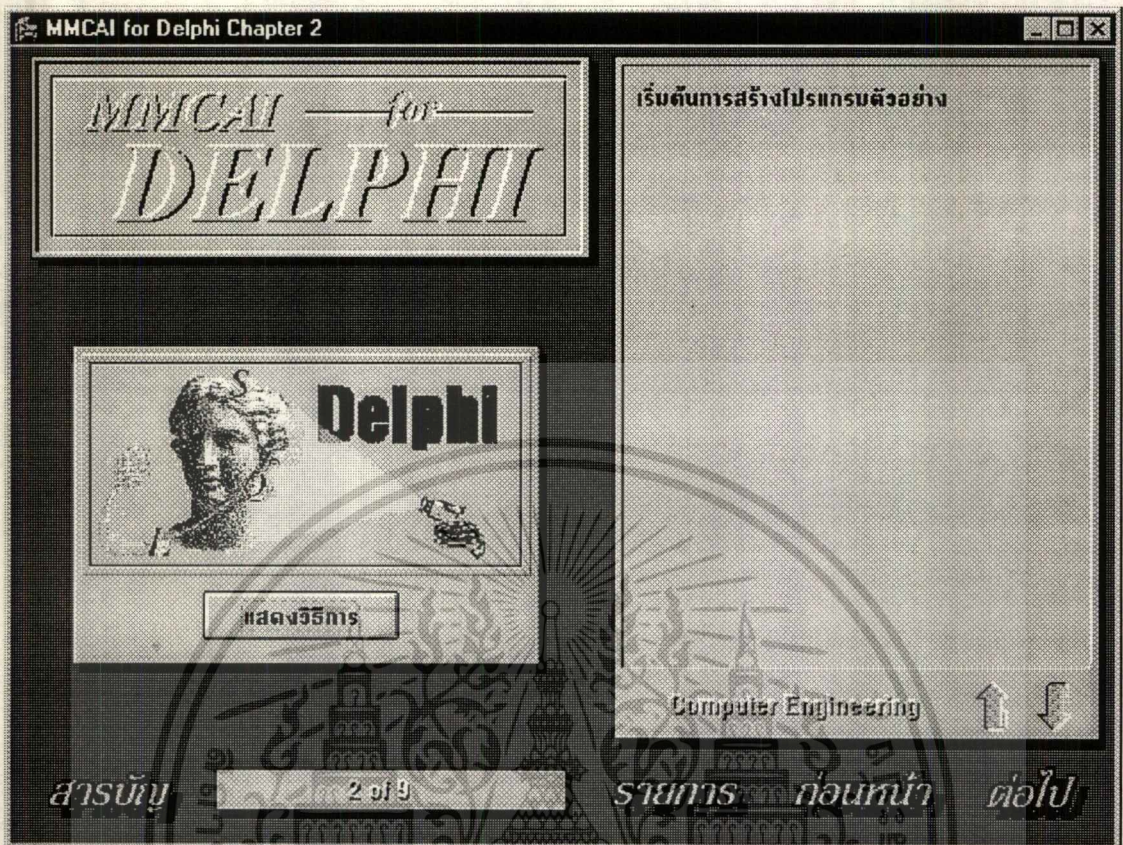
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่เราเลือกบทที่ต้องการแล้วหน้าจอจะเปลี่ยนไปดังรูปข้างล่างซึ่งจะมีเสียงในการบรรยายของแต่ละบท และจะประกอบด้วยเนื้อหาภายในบทซึ่งเราสามารถที่จะเลือกเข้าไปในแต่ละส่วนโดยการคลิก Mouse เข้าไปในเรื่องที่ต้องการจะทราบจะสังเกตเห็นว่ามีรูป ลูกศรสีเขียวซึ่งจะทำหน้าที่คล้ายกับ ScrollBar เพื่อเลื่อนดูเนื้อหาเพิ่มเติม หรือถ้าหากไม่มีการเลือก ใน 10 วินาที จะมีการทำงาน โดย อัตโนมัติ โดยจะเริ่มจากเรื่องแรกเรียงมาเป็นลำดับ



รูป 36 รายการ

ในที่นี้เราเลือกที่ 'เริ่มต้นการสร้างโปรแกรมตัวอย่าง' แล้วทำการคลิกส์ จะปรากฏหน้าต่างดังรูปต่อไปนี้

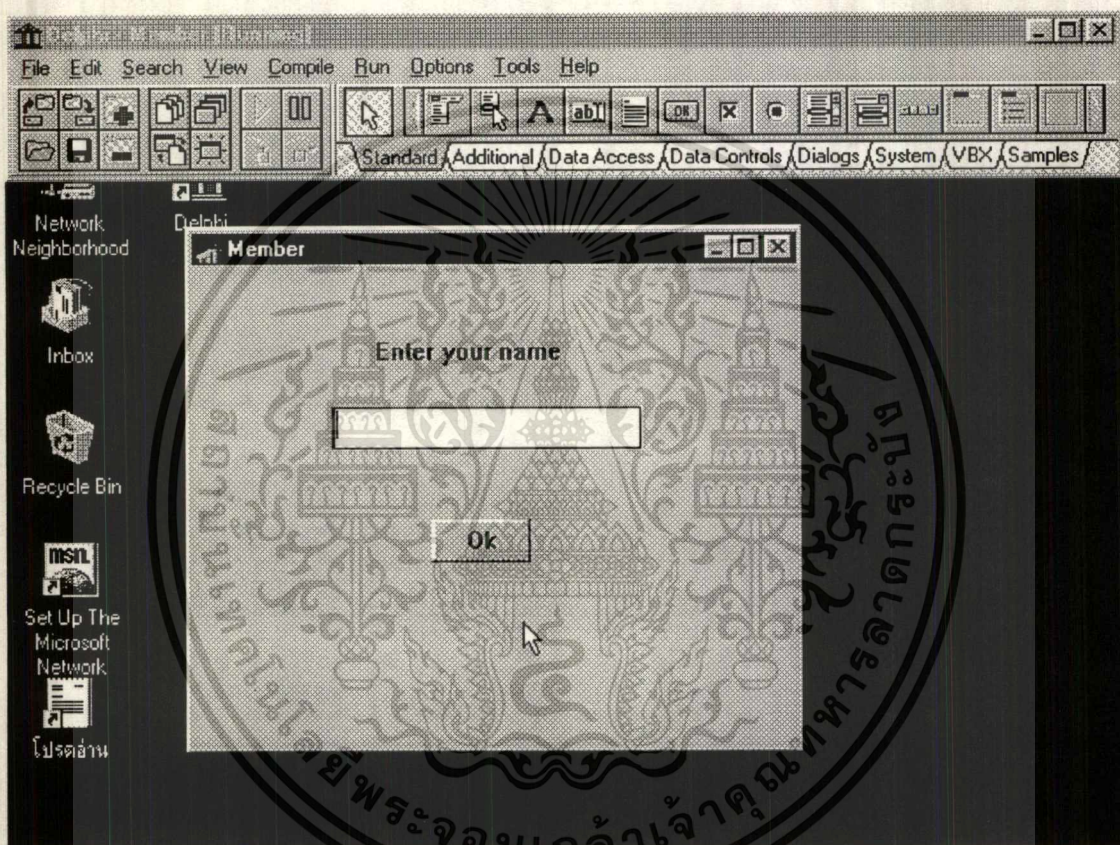


รูปที่ 37 หน้าต่างควบคุม

- จะเห็นว่า หน้าจอนี้เป็นหน้าจอที่ทำหน้าที่ควบคุมการทำงานหลายอย่าง ได้แก่ ปุ่ม
- ปุ่ม สารบัญ เมื่อมีการลาก Mouse ผ่านรูปลูกศรของ Mouse จะเปลี่ยนเป็นรูปมือ เมื่อมีการคลิก Mouse จะทำให้โปรแกรมย้อนกลับไปยัง หน้าสารบัญในรูปที่ 35 ที่นำเสนอไปแล้ว
 - 2 of 9 จะทำหน้าที่เป็นตัวแสดงว่าขณะนี้อยู่ที่หน้า 2 ของทั้งหมด 9 หน้า
 - ปุ่ม รายการ จะทำหน้าที่ให้โปรแกรมกลับไปแสดงรายละเอียดที่ดับบท ดังรูปที่ 36
 - ปุ่ม ก่อนหน้า จะทำหน้าที่กลับไปแสดงหน้าต่างก่อนหน้านี้ 1 หน้า
 - ปุ่ม ต่อไป จะทำหน้าที่แสดงหน้าจอต่อไป 1 หน้า
 - ปุ่ม แสดงวิธีการ จะหน้าที่สั่งให้โปรแกรมแสดงวิธีการทำของแต่ละส่วน
 - หน้าต่างด้านขวามือจะทำหน้าที่ในการแสดงรายละเอียดต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่เราเลือกบทเรียนแล้วกดปุ่ม **แสดงวิธีการ** ก็จะปรากฏรูปหน้าจอดังรูปที่ 38 เนื่องจากข้างต้นเราได้เลือกให้แสดง **เริ่มต้นการสร้างโปรแกรมตัวอย่าง** ก็จะมีโปรแกรมที่ทำหน้าที่สอนเราในการเขียนโปรแกรมซึ่งจะมีเสียง ประกอบ ซึ่งเราก็สามารถเรียนรู้กับโปรแกรมนี้ได้ถึง 10 บท เมื่อเราทำการเรียนจบในแต่ละบทจะมี การทดสอบท้ายบทของทุกบท และจะมีการเก็บคะแนน และชื่อของผู้ที่ทำการทดสอบ



รูป 38 โปรแกรมตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 10

โค้ดของโปรแกรม

โปรแกรมประกอบด้วยไฟล์ Source Code ดังนี้

- Source Code Multimedia ToolBook

CHP1.TBK

CHP2.TBK

CHP3.TBK

CHP4.TBK

CHP5.TBK

CHP6.TBK

CHP7.TBK

CHP8.TBK

CHP9.TBK

CHP10.TBK

MMCALTBK

- Source Code Delphi

QUESTION.PAS

-- script for Book "MMCALTBK"

-- initialize function and variable

-- in : -

-- out: C_drive #reference -save current drive

-- C_cd #reference -save first cdrom drive

-- HDMediaPath #reference -set harddisk path for media

-- setCurrentDrive #reference -set current drive to cdrom drive

-- set midi volume

to handle enterApplication

system string C_drive -- save current drive

system string C_cd -- save first cdrom drive

forward

linkDLL "user"

INT lockWindowUpdate(word)

WORD getDeskTopWindow()

WORD findWindow(DWORD,string)

int showWindow(word,int)

end linkDLL

linkDLL "TB30WIN.DLL"

int yieldApp()

end linkDLL

deskTop = GetDeskTopWindow()

get LockWindowUpdate(deskTop)

defaultStatus of mainWindow = maximized

show mainWindow

เอกสารนี้ **get lockWindowUpdate(0)** ทรัพยากรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

linkDLL "tb30dos.dll"
    string getCurrentDrive()
    int setCurrentDrive(string)
    string getCurrentDirectory(string)
    string getCDDriveList()
end linkDLL
set c_drive to getCurrentDrive()    -- get current drive
set L_cd to getCDDriveList()        -- get cdrom drive present

if item 1 of L_cd is null    -- check cdrom drive
    request "Your CDROM DRIVE not found!" with "Exit"
    send exit
end if
set c_cd to item 1 of L_cd    -- this program use first cdrom drive
do -- test special key pressed to set cdrom drive
    if keyState(keyShift) is "down"
        ask "What is Drive(CAI subdirectory)" with c_cd
        set c_cd to it
    end if
    get MMCAIcd(c_cd) -- test cdrom is MMCAI for Delphi CD
until it    -- until it = 0
get setCurrentDrive(c_cd) -- set current drive to cdrom drive
unlinkDLL "tb30dos.dll"
clear HDMediaPath    --< clear all path in HDMediaPath >
clear CDMediaPath    --< clear all path in CDMediaPath >
-- set harddisk path for media
put C_cd & ":\cai\wave;" & C_cd & ":\cai\image;" & C_cd & ":\cai\sfx;" \
& C_cd & ":\cai\midi;" before HDMediaPath of this book
-- set volume of midi background
get mixerCommand("set mixer device_in MUS control volume both 35")
saveOnClose of this book = no    -- set no ask user when exit
end enterApplication

```

```

-----
-- close all media and unlinkDLL
-- in : LeaveApplication even
-- out: close all media
-----

```

```

to handle LeaveApplication
    mmmclose all
    unlinkDLL "user"
    unlinkDLL "TB30WIN.DLL"
end

```

```

-----
-- test cd in cdrom drive is MMCAI for Delphi CD
-- in : cdDrive #value -name of cdrom drive to test
-- out: handle return true when MMCAI for Delphi is in cdrom drive

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-- false when MMCAI for Delphi not in cdrom drive
```

```
-----
to get MMCAIcd cdDrive
```

```
clear syserror
```

```
sysErrorNumber = 0
```

```
sysSuspend = false
```

```
set FileToOpen to cdDrive & ":\mmcai.id"
```

```
openFile fileToOpen
```

```
put sysErrorNumber into Openstatus
```

```
Conditions
```

```
when Openstatus = 0
```

```
closeFile fileToOpen
```

```
return true
```

```
else
```

```
closeFile fileToOpen
```

```
request "Please insert MMCAI for DELPHI CD in drive " \
& cdDrive with "OK" or "EXIT"
```

```
conditions
```

```
when it is "exit"
```

```
send exit
```

```
return true
```

```
else
```

```
return false
```

```
end conditions
```

```
end conditions
```

```
end MMCAIcd
```

```
*****
```

```
-- script for Background id 0
```

```
-----
-- run MTB file
```

```
-- in : MtbFile #value -name of MTB file
```

```
-- c_cd #reference -name of cdrom drive
```

```
-- out: run MTB files to display
```

```
-----
to handle MtbRun MtbFile
```

```
system c_cd
```

```
mmclose all -- cse all Media
```

```
run c_cd & ":\cai\chapter\ch" & MtbFile & ".exe" -- run MTB file
```

```
hide viewer "index viewer" -- hide main window
```

```
get findwindow(0,"MMCAI for Delphi Chapter " & MtbFile)
```

```
do -- loop to test MTB file run finish
```

```
get yieldApp()
```

```
get findwindow(0,"MMCAI for Delphi Chapter " & MtbFile)
```

```
until it = 0
```

```
show viewer "index viewer" -- show main window
```

```
alwaysOnTop of viewer id 0 = true
```

```
alwaysOnTop of viewer id 0 = false
```

end MtbRun

```
-----
-- run MIDI background song
-- in : mmstatus of clip "midiSong" #reference -is property of clip
--   syslevel #reference --is property of system
-- out: MIDI background song
-----
-- script for run MIDI background song
to handle idle
  if mmStatus of clip "midiSong" = "closed" and syslevel = reader then
    mmPlay clip "midisong" autoclose
    if mmstatus of clip "we_song" = "playing"
      mmclose clip "we_song"
    end if
  end if
end idle
```

```
-----
-- script for page 1
-----
```

```
-- start MIDI background song wait and go to next page
-- in : mmstatus of clip "midiSong" #reference -is property of clip
--   syslevel #reference --is property of system
-- out: MIDI background song
-----
```

```
to handle EnterPage
  if mmStatus of clip "midiSong" = "closed" and syslevel = reader then
    mmPlay clip "midisong" autoclose
  end if
  pause 3 seconds -- wait to show first page
  send next
end
```

```
-----
-- script for page 2
-----
```

```
-- wait and go to next page
-- in : -
-- out: -
-----
```

```
to handle EnterPage
  pause 3 seconds
  send next
end
```

```
-----
-- script for Page 3
-----
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-- in : c_cd #reference -name of cdrom drive
-- det_chp.txt -text file is chapter detail
-- out: det_chp #reference -chapter detail
```

to handle EnterPage

```
system c_cd
system det_chp
-- open file det_chp.txt for data of chapter
clear syserror
sysErrorNumber = 0
sysSuspend = false
set FileToOpen to c_cd & ":\cai\det_chp.txt"
openFile fileToOpen
put sysErrorNumber into Openstatus
Conditions
when Openstatus = 0
    break conditions
else
    request "det_chp.txt not found!" with "Ok"
    closeFile FiletoOpen
    set delt_chp to null
    break EnterPage
end conditions
clear sysError
readFile fileToOpen to EOF
if sysError is not null
    request "det_chp.txt not found!" with "Ok"
    set delt_chp to null
    break buttonUp
else
    put it into det_chp
end if
closeFile fileToOpen
```

end

```
-- script for button id 2 of page 3
```

```
-- change action of self and text of field "textBox"
```

```
-- in : mouse position
```

```
-- det_chp #reference -chapter detail
```

```
-- out: action of self
```

```
-- text of field "textBox" #reference -is property of field
```

to handle MouseEnter

```
system det_chp
```

```
invert of self = true
```

```
put item 1 of det_chp into text of field "textBox"
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end

```
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseLeave
  invert of self = false
  clear text of field "textBox"
end
```

```
-----
-- send message to run MTB file
-- in : mouse click
-- out: -
-----
```

```
to handle ButtonClick
  send MtbRun("1") -- send to run chapter 1
end
```

```
-----
-- set invert of self to false before enter page 3
-- in : -
-- out: self
-----
```

```
notifyBefore enterPage
  invert of self = false
end enterPage
```

```
*****
```

```
-- script for button id 8 of page 3
-----
```

```
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseEnter
  system det_chp
  invert of self = true
  put item 2 of det_chp into text of field "textBox"
end
```

```
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
-----
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-- text of field "textBox" #reference -is property of field
```

```
-----
to handle MouseLeave
```

```
    invert of self = false
```

```
    clear text of field "textBox"
```

```
end
```

```
-----
-- send message to run MTB file
```

```
-- in : mouse click
```

```
-- out: -
```

```
-----
to handle ButtonClick
```

```
    send MtbRun("2") -- send to run chapter 2
```

```
end
```

```
-----
-- set invert of self to false before enter page 3
```

```
-- in : -
```

```
-- out: self
```

```
-----
notifyBefore enterPage
```

```
    invert of self = false
```

```
end enterPage
```

```
*****
```

```
-- script for button id 10 of page 3
```

```
-----
-- change action of self and text of field "textBox"
```

```
-- in : mouse position
```

```
    det_chp #reference -chapter detail
```

```
-- out: action of self
```

```
-- text of field "textBox" #reference -is property of field
```

```
-----
to handle MouseEnter
```

```
    system det_chp
```

```
    invert of self = true
```

```
    put item 3 of det_chp into text of field "textBox"
```

```
end
```

```
-----
-- change action of self and text of field "textBox"
```

```
-- in : mouse position
```

```
    det_chp #reference -chapter detail
```

```
-- out: action of self
```

```
-- text of field "textBox" #reference -is property of field
```

```
-----
to handle MouseLeave
```

```
    invert of self = false
```

```
    clear text of field "textBox"
```

```
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-----
-- send message to run MTB file
-- in : mouse click
-- out: -
-----
```

```
to handle ButtonClick
    send MtbRun("3") -- send to run chapter 3
end
-----
```

```
-- set invert of self to false before enter page 3
-- in : -
-- out: self
-----
```

```
notifyBefore enterPage
```

```
    invert of self = false
```

```
end enterPage
```

```
*****
```

```
-- script for button id 12 of page 3
-----
```

```
-- change action of self and text of field "textBox"
-- in : mouse position
--     det_chp #reference -chapter detail
-- out: action of self
--     text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseEnter
    system det_chp
    invert of self = true
    put item 4 of det_chp into text of field "textBox"
end
-----
```

```
-- change action of self and text of field "textBox"
-- in : mouse position
--     det_chp #reference -chapter detail
-- out: action of self
--     text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseLeave
    invert of self = false
    clear text of field "textBox"
end
-----
```

```
-- send message to run MTB file
-- in : mouse click
-- out: -
-----
```

```
to handle ButtonClick
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

send MtbRun("4") -- send to run chapter 4
end
-----
-- set invert of self to false befor enter page 3
-- in : -
-- out: self
-----
notifyBefore enterPage
    invert of self = false
end enterPage
*****
-- script for button id 13 of page 3
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
to handle MouseEnter
    system det_chp
    invert of self = true
    put item 5 of det_chp into text of field "textBox"
end
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
to handle MouseLeave
    invert of self = false
    clear text of field "textBox"
end
-----
-- send message to run MTB file
-- in : mouse click
-- out: -
-----
to handle ButtonClick
    send MtbRun("5") -- send to run chapter 5
end
-----
-- set invert of self to false befor enter page 3
-- in : -
-- out: self

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-----
notifyBefore enterPage
```

```
    invert of self = false
```

```
end enterPage
```

```
*****
```

```
-- script for button id 7 of page 3
```

```
-----
-- change action of self and text of field "textBox"
```

```
-- in : mouse position
```

```
--    det_chp #reference -chapter detail
```

```
-- out: action of self
```

```
--    text of field "textBox" #reference -is property of field
```

```
-----
to handle MouseEnter
```

```
    system det_chp
```

```
    invert of self = true
```

```
    put item 6 of det_chp into text of field "textBox"
```

```
end
```

```
-----
-- change action of self and text of field "textBox"
```

```
-- in : mouse position
```

```
--    det_chp #reference -chapter detail
```

```
-- out: action of self
```

```
--    text of field "textBox" #reference -is property of field
```

```
-----
to handle MouseLeave
```

```
    invert of self = false
```

```
    clear text of field "textBox"
```

```
end
```

```
-----
-- send message to run MTB file
```

```
-- in : mouse click
```

```
-- out: -
```

```
-----
to handle ButtonClick
```

```
    send MtbRun("6") -- send to run chapter 6
```

```
end
```

```
-----
-- set invert of self to false before enter page 3
```

```
-- in : -
```

```
-- out: self
```

```
-----
notifyBefore enterPage
```

```
    invert of self = false
```

```
end enterPage
```

```
*****
```

```
-- script for button id 9 of page 3
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----

```

```

to handle MouseEnter
  system det_chp
  invert of self = true
  put item 7 of det_chp into text of field "textBox"
end
-----

```

```

-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----

```

```

to handle MouseLeave
  invert of self = false
  clear text of field "textBox"
end
-----

```

```

-- send message to run MTB file
-- in : mouse click
-- out: -
-----

```

```

to handle ButtonClick
  send MtbRun("7") -- send to run chapter 7
end
-----

```

```

-- set invert of self to false before enter page 3
-- in : -
-- out: self
-----

```

```

notifyBefore enterPage
  invert of self = false
end enterPage
-----

```

```

*****
-- script for button id 1 1 of page 3
-----

```

```

-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ซึ่งการแจ้งในหนังสือเรียนนั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-----
to handle MouseEnter
    system det_chp
    invert of self = true
    put item 8 of det_chp into text of field "textBox"
end
```

```
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseLeave
    invert of self = false
    clear text of field "textBox"
end
```

```
-----
-- send message to run MTB file
-- in : mouse click
-- out: -
-----
```

```
to handle ButtonClick
    send MtbRun("8") -- send to run chapter 8
end
```

```
-----
-- set invert of self to false before enter page 3
-- in : -
-- out: self
-----
```

```
notifyBefore enterPage
    invert of self = false
end enterPage
```

```
*****
```

```
-- script for button id 14 of page 3
-----
```

```
-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----
```

```
to handle MouseEnter
    system det_chp
    invert of self = true
    put item 9 of det_chp into text of field "textBox"
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----

```

```

to handle MouseLeave
  invert of self = false
  clear text of field "textBox"
end
-----

```

```

-- send message to run MTB file
-- in : mouse click
-- out: -
-----

```

```

to handle ButtonClick
  send MtbRun("9") -- send to run chapter 9
end
-----

```

```

-- set invert of self to false before enter page 3
-- in : -
-- out: self
-----

```

```

notifyBefore enterPage
  invert of self = false
end enterPage
-----

```

```

*****
-- script for button id 15 of page 3
-----

```

```

-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self
--   text of field "textBox" #reference -is property of field
-----

```

```

to handle MouseEnter
  system det_chp
  invert of self = true
  put item 10 of det_chp into text of field "textBox"
end
-----

```

```

-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference -chapter detail
-- out: action of self

```

```

-- text of field "textBox" #reference -is property of field

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือมีการขังในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
to handle MouseLeave
    invert of self = false
    clear text of field "textBox"
end

```

```

-----
-- send message to run MTB file
-- in : mouse click
-- out: -

```

```

-----
to handle ButtonClick
    send MtbRun("10")      -- send to run chapter 10
end

```

```

-----
-- set invert of self to false befor enter page 3
-- in : -
-- out: self

```

```

-----
notifyBefore enterPage
    invert of self = false
end enterPage

```

```

*****
-- script for button id 3 of page 3

```

```

-----
-- change action of self,text of field "textBox" and play clip "exit"
-- in : mouse position
--   det_chp #reference --chapter detail
-- out: action of self
--   text of field "textBox" #reference --is property of field

```

```

-----
to handle MouseEnter
    system det_chp
    mmclose WaveAudio
    mmyield
    mmPlay clip "exit" autoclose
    put item 1 1 of det_chp into text of field "textBox"
    invert of self = true
end

```

```

-----
-- change action of self and text of field "textBox"
-- in : mouse position
--   det_chp #reference --chapter detail
-- out: action of self
--   text of field "textBox" #reference --is property of field

```

```

-----
to handle MouseLeave

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clear text of field "textBox"
end
-----
-- to exit program
-- in : mouse click
-- out: system to exit
-----
to handle ButtonClick
    send exit
end
*****

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

unit 1

```
-- script for Book "CHP1.TBK"
```

```
-----
-- initialize function and variable
```

```
-- in : -
```

```
-- out: C_drive #reference -save current drive
```

```
-- HDMediaPath #reference -set harddisk path for media
-----
```

to handle enterApplication

```
system string C_drive          -- save current drive
system string ScmPath          -- save Path of ScreenCam files
system int Tdelay
    set Tdelay to 10000
system int statusTimer
system logical IsWavePlay      -- save ststus of playing wave
set IsWavePlay to false
forward
linkDLL "user"
    INT lockWindowUpdate(word)
    WORD getDeskTopWindow()
    WORD findWindow(DWORD,string)
    int showWindow(word,int)
end linkDLL
linkDLL "TB30WIN.DLL"
    int yieldApp()
end linkDLL
deskTop = GetDeskTopWindow()
get LockWindowUpdate(deskTop)
defaultStatus of mainWindow = maximized
show mainWindow
get lockWindowUpdate(0)
get FindCurrentDrive()
set C_drive to it
clear HDMediaPath             -- clear all path in HDMediaPath
    -- set harddisk path for media
put C_drive & ":\cai\wave;" & C_drive & ":\cai\image;" & C_drive & ":\cai\sfx;" \
& C_drive & ":\cai\midi;" before HDMediaPath of this book
ScmPath = C_drive & ":\cai\screen\"
saveOnClose of this book = no  -- set no ask user when exit
```

```
end enterApplication
-----
```

```
-- close all media and unlinkDLL
```

```
-- in : LeaveApplication even
```

```
-- out: close all media
-----
```

to handle LeaveApplication

```
mmclose all
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unlinkDLL "user"
unlinkDLL "TB30WIN.DLL"
end

```

```

-----
-- find current drive
-- in : -
-- out: function return current drive
-----

```

```

to get FindCurrentDrive
    local string temp_drive
    linkDLL "tb30dos.dll"
        string getCurrentDrive()
    end linkDLL
    set temp_drive to getCurrentDrive()
    unlinkDLL "tb30dos.dll"
    return temp_drive
end FindCurrentDrive

```

```

*****

```

```

-- script for Background id 0

```

```

-----
-- manage action for each page
-- in : EnterPage even
-- Tdelay #reference -constant of delay time
-- out: statustime #reference -timer id is used
-----

```

```

to handle EnterPage
    system Tdelay
    system statusTimer
    if syslevel is reader
        if mmstatus of clip "logo" is "playing"
            mmClose clip "logo"
        end if
        get pagenumber of this page -- get number of current page
        caption of button "status text1" of this background = \
        (it-1) &" of 1 1"
        caption of button "status text2" of this background = \
        (it-1) &" of 1 1"
        enabled of button "show" of background id 0 = false
        mmclose waveaudio
        mmyield -- update ststus of mmstatus
        get pagenumber of this page -- get number of current page
        conditions -- and store in it
            when it = 1
                send PlayWave 1 1 2
            when it = 2
                enabled of button "show" of background id 0 = true
                mmPlay clip "logo" in stage "myStage" of background id 0 hold

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-- start timer
get timerStart("single",Tdelay,0,background id 0)
statusTimer = it -- store timer id to statusTimer
timerID of this page = statusTimer
when it = 3
  enabled of button "show" of background id 0 = true
  mmPlay clip "logo" in stage "myStage" of background id 0 hold
  -- start timer
  get timerStart("single",Tdelay,0,background id 0)
  statusTimer = it -- store timer id to statusTimer
  timerID of this page = statusTimer
when it = 4
  if mmstatus of clip "logo" is "playing"
    mmClose clip "logo"
  end if
  -- start timer
  get timerStart("single",Tdelay,0,background id 0)
  statusTimer = it -- store timer id to statusTimer
  timerID of this page = statusTimer
when it = 5
  --{Play clip}
  send PlayWave 103
when it = 6
  --{Play clip}
  send PlayWave 104
when it = 7
  --{Play clip}
  send PlayWave 105
when it = 8
  --{Play clip}
  send PlayWave 106
when it = 9
  --{Play clip}
  send PlayWave 107
when it = 10
  --{Play clip}
  send PlayWave 108
when it = 11
  --{Play clip}
  send PlayWave 109
when it = 12
  send ProbRun("1") -- send to run final question
  -- send to exit
  send buttonClick to ellipse "index" of this background
else
end conditions
end if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end

```
-----
-- control playing MIDI
-- in : mmstatus of clip "midiSong" #reference -is property of clip
--   syslevel #reference --is property of system
-- out: background song
-----
```

to handle idle

```
    if mmStatus of clip "midiSong" = "closed" and syslevel = reader then
        mmPlay clip "midisong" autoclose
```

```
    end if
```

end idle

```
-----
-- control action after timer active
-- in : timeID #value -active timer id
--   statusTimer #reference -timer id is used
-- out: even to each page condition
-----
```

to handle timerNotify timerID

```
    system statusTimer--The ID number
    if timerID = statusTimer
        if (pagenumber of this page = 2) or (pagenumber of this page = 3)
            send buttonclick to button "show" of this background
        else
            go to next page
        end if
    end if
```

```
    end if
```

end timerNotify

```
-----
-- increase time of timer when user click in screen
-- in : mouse click
--   IsWavePlay #reference -status of playing wave
--   Tdelay #reference -constant of delay time
-- out: statusTimer #reference -new timer id
-----
```

to handle buttonClick

```
    system statusTimer
    system Tdelay
    system IsWavePlay
    if not IsWavePlay -- no increase if wave playing
        -- start new timer
        get   timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        -- set new timer id to statusTimer
        timerID of background id 0 = statusTimer
```

```
    end if
```

end

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
-- play wave and start timer after stoped wave
-- in : ClipId #value -id of wave clip to play
-- out: statusTimer #reference -new timer id
--     IsWavePlay #reference -status of playing wave
-----

```

```

to handle PlayWave ClipId
  system statusTimer
  system IsWavePlay
  set IsWavePlay to true
  mmPlay clip id ClipId autoclose
  count = mmEndPoint of clip id ClipId
  get timerStart("single",count,0,group "logo" of background id 0)
  statusTimer = it
  timerID of self = statusTimer
end
-----

```

```

-- run final question and wait it end
-- in : Chnum #value -Chapter number to run
--     c_Drive #reference -current drive
-- out: run final question
-----

```

```

to handle ProbRun ChNum
  system c_Drive
  mmclose all
  hide viewer id 0
  run c_Drive & "\cai\question\problem.exe /" & ChNum & " " & C_drive
  get findwindow(0,"คำถามท้ายบท")
  do
  get yieldApp()
  get findwindow(0,"คำถามท้ายบท")
  until it = 0
end ProbRun
-----

```

```

*****
-- script for button "show" of background id 0 ปุ่มแสดงวิธีการ
-----

```

```

-- run screenCam file and send to next page
-- in : buttonClick
--     statusTimer #reference -new timer id
-- out: send name of screenCam file to handle ScmRun
-----

```

```

to handle ButtonClick
  system statusTimer
  statusTimer = null -- clear statusTimer
  if sysLevel is Reader
    mmclose sequencer
    mmclose WaveAudio

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

get pagenumber of this page
conditions
    when it = 2
        send ScmRun "ch1-1.SCM"
        send next
    when it = 3
        send ScmRun "ch1-2.SCM"
        if mmstatus of clip "logo" is "playing"
            mmClose clip "logo"
        end if
    end if
    send next

else
end conditions

end if

end

-----
-- run screenCam file and wait
-- in : ScmFile #value -name of screenCam to run
--   ScmPath #reference -path of screenCam files
-- out: run screenCam
-----
to handle ScmRun ScmFile
    system ScmPath
    run Scmpath & "scplayer.exe /cs " & ScmPath & ScmFile
    step i from 1 to 200
        get YieldApp()
    end step
    hide viewer id 0
    get findwindow(0,ScmFile)
    get showWindow(it,2)
    do
        -- loop to test end run screenCam file
    get yieldApp()
    sta = it
    get findwindow(0,ScmFile)
until it = 0
show viewer id 0
    alwaysOnTop of viewer id 0 = true
    alwaysOnTop of viewer id 0 = false
end ScmRun
*****
-- script for Ellipse "index" of background id 0 ปุ่มสารบัญ
-----
-- exit program
-- in : buttonClick
-- out: -
-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

to handle ButtonClick
    mmclose all
    send exit
end
-----
-- change mouse cursor
-- in : mouse pointer position
-- out: mouse cursor
-----
to handle MouseEnter
    sysCursor = 44
end
to handle MouseLeave
    sysCursor = 1
end
-----
-- change graphics of button id 97
-- in : buttonDown even and buttonUp even
-- out: button id 97
-----
to handle Buttondown
    invert of button id 97 of this background = true
end
to handle Buttonup
    invert of button id 97 of this background = false
end
*****
-- script for Ellipse id 101 of background id 0 ปุ่มรายการ
-----
-- go to first page
-- in : buttonClick
-- out: go to first page
-----
to handle ButtonClick
    mmclose waveaudio
    mmyield
    mmPlay clip "click" wait
    if mmstatus of clip "logo" is "playing"
        mmClose clip "logo"
    end if
    go to first page
end
-----
-- change graphics of button id 98
-- in : buttonDown even and buttonUp even
-- out: button id 98
-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

to handle Buttondown
    invert of button id 98 of this background = true
end
to handle Buttonup
    invert of button id 98 of this background = false
end

```

```

-----
-- change mouse cursor
-- in : mouse pointer position
-- out: mouse cursor
-----

```

```

to handle MouseEnter
    sysCursor = 44
end

```

```

to handle MouseLeave
    sysCursor = 1
end

```

```

*****
-- script for Ellipse id 93 of background id 0 ปุ่มก่อนหน้า

```

```

-----
-- go to previous page
-- in : buttonClick
-- out: go to previous page
-----

```

```

to handle ButtonClick
    if enabled of button id 94 of this background = true
        mmclose waveaudio
        mmyield
        mmPlay clip "click" wait
        send previous
    end if
end

```

```

-----
-- change graphics of button id 94
-- in : buttonDown even and buttonUp even
-- out: button id 94
-----

```

```

to handle Buttondown
    invert of button id 94 of this background = true
end
to handle Buttonup
    invert of button id 94 of this background = false
end

```

```

-----
-- change mouse cursor
-- in : mouse pointer position
-- out: mouse cursor

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
to handle MouseEnter
    sysCursor = 44
end
to handle MouseLeave
    sysCursor = 1
end
*****
-- script for Ellipse id 100 of background id 0 ปุ่มต่อไป
-----
-- send to next page
-- in : buttonClick
-- out: send to next page
-----
to handle ButtonClick
    if enabled of button id 96 of this background = true
        mmclose waveaudio
        mmyield
        mmPlay clip "click" wait
        send next
    end if
end
-----
-- change graphics of button id 96
-- in : buttonDown even and buttonUp even
-- out: button id 96
-----
to handle Buttondown
    invert of button id 96 of this background = true
end
to handle Buttonup
    invert of button id 96 of this background = false
end
-----
-- change mouse cursor
-- in : mouse pointer position
-- out: mouse cursor
-----
to handle MouseEnter
    sysCursor = 44
end
to handle MouseLeave
    sysCursor = 1
end
*****
-- script for field "Descriptions" of page 1
-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-- go to page that user select
-- in : mouse click
-- out: go to page
```

```
to handle ButtonClick
```

```
  get selectedTextLines -- get text line is selected
```

```
  conditions
```

```
    when it = 2
```

```
      go to page 3
```

```
    when it = 3
```

```
      go to page 4
```

```
    when it = 4
```

```
      go to page 5
```

```
    when it = 5
```

```
      go to page 6
```

```
    when it = 6
```

```
      go to page 7
```

```
    when it = 7
```

```
      go to page 8
```

```
    when it = 8
```

```
      go to page 9
```

```
    when it = 9
```

```
      go to page 10
```

```
    when it = 10
```

```
      go to page 11
```

```
    when it = 11
```

```
      go to page 12
```

```
  else
```

```
    send next
```

```
  end conditions
```

```
end
```

```
*****
```

```
-- script for Group "slides" of page 1
```

```
-- scroll field "Descriptions" of page 1
```

```
-- in : nDelta #value -number of line to scroll
```

```
-- out: field "Descriptions" of page 1
```

```
-- button "scrollUp" -enable and disable it
```

```
-- button "scrollDown" -enable and disable it
```

```
to handle scrollFields nDelta
```

```
  targetField = field "descriptions" of page 1
```

```
  sysLockScreen = true
```

```
  if (nDelta = 0)
```

```
    scroll of targetField = 0
```

```
  else
```

```
    increment scroll of targetField by nDelta
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end
-- if scroll is 0, disable 'up' button:
enabled of (button "scrollUp" of self) = not (scroll of targetField = 0)
-- if no overflowing text, disable 'down' button:
enabled of (button "scrollDown" of self) = not (textOverFlow of targetField = 0)
sysLockScreen = false
end
-----
-- initial before enter this page
-- in : -
-- out: send scrollFields with 0
-----
notifyAfter enterPage
    send scrollFields 0
end
*****
-- script for button "scrollUp" of page 1
-----
-- scroll field "Descriptions" of page 1
-- in : buttonClick
-- out: send scrollFields with -1
-----
to handle buttonClick
    send scrollFields -1
    forward
end
-----
-- scroll field "Descriptions" of page 1
-- in : button Still Down
-- out: send scrollFields with -1 until enabled of self is true
-----
to handle buttonStillDown
    send scrollFields -1
    if (not enabled of self)
        invert of self = false
        break to system
    end
end
end
*****
-- script for button "scrollDown" of page 1
-----
-- scroll field "Descriptions" of page 1
-- in : buttonClick
-- out: send scrollFields with 1
-----
to handle buttonClick
    send scrollFields 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

forward
end
-----
-- scroll field "Descriptions" of page 1
-- in : button Still Down
-- out: send scrollFields with 1 until enabled of self is true
-----
to handle buttonStillDown
  if (not enabled of self)
    invert of self = false
    break to system
  end
  send scrollFields 1
end
*****
-- script for group "logo" of background id 0
-----
-- show logo
-- in : buttonClick
-- out: viewer "popup"
-----
to handle ButtonClick
  system statusTimer
  mmclose waveAudio
  defaultPage of viewer "popup" = page id 3 of this book
  open viewer "popup"
  show viewer "popup"
  statusTimer = null
end
-----
-- start new timer after end of wave play
-- in : timerID -id of timer
-- out: new timer id send to background id 0
-----
to handle timerNotify timerID
  system Tdelay
  system statusTimer--The ID number
  system IsWavePlay
  set IsWavePlay to false -- wave now is closed
  if timerID of background id 0 = statusTimer
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of background id 0 = statusTimer
  end if
end timerNotify
-----
-- send to next page

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-- in : buttonClick
-- out: send to next page
```

```
-----
to handle MouseEnter
```

```
    sysCursor = cursor "data"
```

```
end
```

```
to handle MouseLeave
```

```
    sysCursor = 1
```

```
end
```

```
#####
```

ในแต่ละบทจะใช้ script ของแต่ละออบเจ็กต์เหมือน

แต่จะมี script ของ

- Background id 0 คือ handle EnterPage

- Button "Show" of background id 0 คือ handle ButtonClick

ซึ่งจะแตกต่างกันในแต่ละบท เพื่อความประหยัดจึงขอยก script

เฉพาะส่วนที่ต่างกันมาอธิบาย

บทที่ 2

```
-- script for Book "CHP2.TBK"
```

```
script ของ Background id 0 คือ handle EnterPage
```

```
to handle EnterPage
```

```
    system Tdelay
```

```
    system statusTimer
```

```
    if syslevel is reader
```

```
        get pagenumber of this page
```

```
        caption of button "status text1" of this background = \
```

```
(it-1) &" of 9"
```

```
        caption of button "status text2" of this background = \
```

```
(it-1) &" of 9"
```

```
        enabled of button "show" of background id 0 = false
```

```
        mmclose waveaudio
```

```
        mmyield
```

```
    if pagenumber of this page is not 1
```

```
        enabled of button "show" of background id 0 = true
```

```
        mmPlay clip "logo" in stage "myStage" of background id 0 hold
```

```
    end if
```

```
    get pagenumber of this page
```

```
    conditions
```

```
        when it = 1
```

```
            --{Play clip}
```

```
            send PlayWave 1 15
```

```
        when it = 2
```

```
            get timerStart("single",Tdelay,0,background id 0)
```

```
            statusTimer = it
```

```
            timerID of this page = statusTimer
```

```
        when it = 3
```

```
            get timerStart("single",Tdelay,0,background id 0)
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        statusTimer = it
        timerID of this page = statusTimer
    when it = 4
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 5
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 6
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 7
        send PlayWave 116
    when it = 8
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 9
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 10
        send ProbRun("2")
        send buttonClick to ellipse "index" of this background
    else
    end conditions
end if
end

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

system C_Drive --<current drive>
system statusTimer
local string ScmFile
. statusTimer = null -- clear statusTimer
if sysLevel is Reader
    mmclose sequencer
    mmclose WaveAudio
    get pagenumber of this page
    conditions
        when it = 2
            send ScmRun "ch2-1.SCM"
            send next

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        send ScmRun "ch2-2.SCM"
        send next
    when it = 4
        send ScmRun "ch2-3.SCM"
        send next
    when it = 5
        send ScmRun "ch2-4.SCM"
        send next
    when it = 6
        send ScmRun "ch2-5.SCM"
        send next
    when it = 7
        send ScmRun "ch2-6.SCM"
        send next
    when it = 8
        send ScmRun "ch2-7.SCM"
        send next
    when it = 9
        send ScmRun "ch2-8.SCM"
        send next
    else
    end conditions
end if
end
*****
บทที่ 3
-- script for Book "CHP3.TBK"
script ของ Background id 0 คือ handle_EnterPage
to handle EnterPage
    local count
    system Tdelay
    system statusTimer
    if syslevel is reader
    get pagenumber of this page
    caption of button "status text1" of this background = \
(it-1) &" of 5"
    caption of button "status text2" of this background = \
(it-1) &" of 5"
    enabled of button "show" of background id 0 = false
    mmclose waveaudio
    mmyield
    if pagenumber of this page is not 1
        enabled of button "show" of background id 0 = true
        mmPlay clip "logo" in stage "myStage" of background id 0 hold
    end if
    get pagenumber of this page
    conditions

```

```

when it = 1
    --{Play clip}
    send PlayWave 1 17
when it = 2
    get    timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 3
    get    timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 4
    get    timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 5
    get    timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 6
    send ProbRun("3")
    send buttonClick to ellipse "index" of this background
else
end conditions
end if
end

```

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

```

    system C_Drive ---<current drive>
    system statusTimer
    local string ScmFile
    statusTimer = null -- clear statusTimer
    if sysLevel is Reader
    mmclose sequencer
        mmclose WaveAudio
        get pagenumber of this page
        conditions
            when it = 2
                send ScmRun "ch3-1.SCM"
                send next
            when it = 3
                send ScmRun "ch3-2.SCM"
                send next
            when it = 4
                send ScmRun "ch3-3.SCM"
                send next

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        when it = 5
            send ScmRun "ch3-4.SCM"
            send next
        else
        end conditions
    end if
end

```

บทที่ 4

-- script for Book "CHP4.TBK"

script ของ Background id 0 คือ handle EnterPage

to handle EnterPage

```

    local count
    system Tdelay
    system statusTimer
    if syslevel is reader
    get pagenumber of this page
    caption of button "status text1" of this background = \
(it-1) &" of 4"
    caption of button "status text2" of this background = \
(it-1) &" of 4"
    enabled of button "show" of background id 0 = false
    mmclose waveaudio
    mmyield
    if pagenumber of this page is not 1
        enabled of button "show" of background id 0 = true
        mmPlay clip "logo" in stage "myStage" of background id 0 hold
    end if
    get pagenumber of this page
    conditions
        when it = 1
            --(Play clip)
            send PlayWave 121
        when it = 2
            get timerStart("single",Tdelay,0,background id 0)
            statusTimer = it
            timerID of this page = statusTimer
        when it = 3
            get timerStart("single",Tdelay,0,background id 0)
            statusTimer = it
            timerID of this page = statusTimer
        when it = 4
            get timerStart("single",Tdelay,0,background id 0)
            statusTimer = it
            timerID of this page = statusTimer
        when it = 5
            get timerStart("single",Tdelay,0,background id 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        statusTimer = it
        timerID of this page = statusTimer
    when it = 6
        send ProbRun("4")
        send buttonClick to ellipse "index" of this background
    else
    end conditions
end if
end

```

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

```

    system C_Drive --<current drive>
    system statusTimer
    local string ScmFile
    statusTimer = null -- clear statusTimer
    if sysLevel is Reader
        mmclose sequencer
        mmclose WaveAudio
        get pagenumber of this page
        conditions
            when it = 2
                send ScmRun "ch4-1.SCM"
                send next
            when it = 3
                send ScmRun "ch4-2.SCM"
                send next
            when it = 4
                send ScmRun "ch4-3.SCM"
                send next
            when it = 5
                send ScmRun "ch4-4.SCM"
                send next
        else
        end conditions
    end if
end

```

```

end

```

```

*****

```

บทที่ 5

```

-- script for Book "CHP5.TBK"

```

```

script ของ Background id 0 คือ handle EnterPage
to handle EnterPage

```

```

    local count
    system Tdelay
    system statusTimer
    if syslevel is reader
        get pagenumber of this page

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

caption of button "status text1" of this background = \
(it-1) &" of 5"
caption of button "status text2" of this background = \
(it-1) &" of 5"
enabled of button "show" of background id 0 = false
mmclose waveaudio
mmyield
if pagenumber of this page is not 1
    enabled of button "show" of background id 0 = true
    mmPlay clip "logo" in stage "myStage" of background id 0 hold
end if
get pagenumber of this page
conditions
    when it = 1
        --{Play clip}
        send PlayWave 1 17
    when it = 2
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 3
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 4
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 5
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 6
        send ProbRun("5")
        send buttonClick to ellipse "index" of this background
    else
end conditions
end if
end

```

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

```

    system C_Drive --<current drive>
    system statusTimer
    local string ScmFile
    statusTimer = null -- clear statusTimer
    if sysLevel is Reader

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mmclose sequencer
mmclose WaveAudio
get pagenumber of this page
conditions
    when it = 2
        send ScmRun "ch5-1.SCM"
        send next
    when it = 3
        send ScmRun "ch5-2.SCM"
        send next
    when it = 4
        send ScmRun "ch5-3.SCM"
        send next
    when it = 5
        send ScmRun "ch5-4.SCM"
        send next
    else
end conditions
end if
end
*****
บทที่ 6
-- script for Book "CHP6.TBK"
script ของ Background id 0 คือ handle EnterPage
to handle EnterPage
    local count
    system Tdelay
    system statusTimer
    if syslevel is reader
        get pagenumber of this page
        caption of button "status text1" of this background = \
        (it-1) &" of 4"
        caption of button "status text2" of this background = \
        (it-1) &" of 4"
        enabled of button "show" of background id 0 = false
        mmclose waveaudio
        mmyield
        if pagenumber of this page is not 1
            enabled of button "show" of background id 0 = true
            mmPlay clip "logo" in stage "myStage" of background id 0 hold
        end if
        get pagenumber of this page
        conditions
            when it = 1
                --(Play clip)
                send PlayWave 1 17
            when it = 2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

get timerStart("single",Tdelay,0,background id 0)
statusTimer = it
timerID of this page = statusTimer
when it = 3
get timerStart("single",Tdelay,0,background id 0)
statusTimer = it
timerID of this page = statusTimer
when it = 4
get timerStart("single",Tdelay,0,background id 0)
statusTimer = it
timerID of this page = statusTimer
when it = 5
send ProbRun("6")
send buttonClick to ellipse "index" of this background
else
end conditions
end if
end

```

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

```

system C_Drive --<current drive>
system statusTimer
local string ScmFile
statusTimer = null -- clear statusTimer
if sysLevel is Reader
mclose sequencer
mclose WaveAudio
get pagenum of this page
conditions
when it = 2
send ScmRun "ch6-1.SCM"
send next
when it = 3
send ScmRun "ch6-2.SCM"
send next
when it = 4
send ScmRun "ch6-3.SCM"
send next
else
end conditions
end if

```

```
end
```

```
*****
```

บทที่ 7

```
-- script for Book "CHP7.TBK"
```

```
script ของ Background id 0 คือ handle EnterPage
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

to handle EnterPage

```

local count
system Tdelay
system statusTimer
if syslevel is reader
get pagenumber of this page
caption of button "status text1" of this background = \
(it-1) &" of 4"
caption of button "status text2" of this background = \
(it-1) &" of 4"
enabled of button "show" of background id 0 = false
mmclose waveaudio
mmyield
if pagenumber of this page is not 1
    enabled of button "show" of background id 0 = true
    mmPlay clip "logo" in stage "myStage" of background id 0 hold
end if
get pagenumber of this page
conditions
when it = 1
    --{Play clip}
    send PlayWave 1 17
when it = 2
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 3
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 4
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 5
    send ProbRun("7")
    send buttonClick to ellipse "index" of this background
else
end conditions
end if
end

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick

to handle ButtonClick

```

system C_Drive --<current drive>
system statusTimer

```

```

local string ScmFile

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

statusTimer = null -- clear statusTimer
if sysLevel is Reader
  mmclose sequencer
    mmclose WaveAudio
    get pagenumber of this page
    conditions
      when it = 2
        send ScmRun "ch7-1.SCM"
        send next
      when it = 3
        send ScmRun "ch7-2.SCM"
        send next
      when it = 4
        send ScmRun "ch7-3.SCM"
        send next
      when it = 5
      else
    end conditions
  end if
end
end
*****
บทที่ 8
-- script for Book "CHP8.TBK"
script ของ Background id 0 ที่ handle EnterPage
to handle EnterPage
  local count
  system Tdelay
  system statusTimer
  if syslevel is reader
    get pagenumber of this page
    caption of button "status text1" of this background = \
(it-1) &" of 5"
    caption of button "status text2" of this background = \
(it-1) &" of 5"
    enabled of button "show" of background id 0 = false
    mmclose waveaudio
    mmyield
    if pagenumber of this page is not 1
      enabled of button "show" of background id 0 = true
      mmPlay clip "logo" in stage "myStage" of background id 0 hold
    end if
    get pagenumber of this page
    conditions
      when it = 1
        --{Play clip}
        send PlayWave 1 17
      when it = 2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        get      timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 3
        get      timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 4
        get      timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 5
        get      timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 6
        send ProbRun("8")
        send buttonClick to ellipse "index" of this background
    else
    end conditions
end if
end

```

```

script ของ Button "Show" of background id 0 คือ handle ButtonClick
to handle ButtonClick

```

```

    system C_Drive --<current drive>
    system statusTimer
    local string ScmFile
    statusTimer = null -- clear statusTimer
    if sysLevel is Reader
    mmclose sequencer
    mmclose WaveAudio
    get pagenumber of this page
    conditions
        when it = 2
            send ScmRun "ch8-1.SCM"
            send next
        when it = 3
            send ScmRun "ch8-2.SCM"
            send next
        when it = 4
            send ScmRun "ch8-3.SCM"
            send next
        when it = 5
            send ScmRun "ch8-4.SCM"
            send next
    else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end conditions
    end if
end
*****

```

บทที่ 9

-- script for Book "CHP9.TBK"

script ของ Background id 0 คือ handle EnterPage

to handle EnterPage

```

    local count
    system Tdelay
    system statusTimer
    if syslevel is reader
    get pagenumber of this page
    caption of button "status text1" of this background = \
(it-1) &" of 6"
    caption of button "status text2" of this background = \
(it-1) &" of 6"
    enabled of button "show" of background id 0 = false
    mmclose waveaudio
    mmyield
    if pagenumber of this page is not 1
        enabled of button "show" of background id 0 = true
        mmPlay clip "logo" in stage "myStage" of background id 0 hold
    end if
    get pagenumber of this page
    conditions
    when it = 1
        --(Play clip)
        send PlayWave 1 17
    when it = 2
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 3
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 4
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 5
        get timerStart("single",Tdelay,0,background id 0)
        statusTimer = it
        timerID of this page = statusTimer
    when it = 6
        get timerStart("single",Tdelay,0,background id 0)

```

```

        statusTimer = it
        timerID of this page = statusTimer
    when it = 7
        send ProbRun("9")
        send buttonClick to ellipse "index" of this background
    else
    end_conditions
end if
end

```

```

script ของ Button "Show" of background id 0 ดี handle ButtonClick
to handle ButtonClick

```

```

    system C_Drive --<current drive>
    system statusTimer
    local string ScmFile
    statusTimer = null -- clear statusTimer
    if sysLevel is Reader
    mmclose sequencer
    mmclose WaveAudio
    get pagenumber of this page
    conditions
        when it = 2
            send ScmRun "ch9-1.SCM"
            send next
        when it = 3
            send ScmRun "ch9-2.SCM"
            send next
        when it = 4
            send ScmRun "ch9-3.SCM"
            send next
        when it = 5
            send ScmRun "ch9-4.SCM"
            send next
        when it = 6
            send ScmRun "ch9-5.SCM"
            send next
    else
    end_conditions
end if
end

```

บทที่ 10

-- script for Book "CHP10.TBK"

```

script ของ Background id 0 ดี handle EnterPage
to handle EnterPage

```

```

    local count

```

```

    system Tdelay

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

system statusTimer
if syslevel is reader
get pagenumber of this page
caption of button "status text1" of this background = \
(it-1) &" of 7"
caption of button "status text2" of this background = \
(it-1) &" of 7"
enabled of button "show" of background id 0 = false
mmclose waveaudio
mmyield
if pagenumber of this page is not 1
    enabled of button "show" of background id 0 = true
    mmPlay clip "logo" in stage "myStage" of background id 0 hold
end if
get pagenumber of this page
conditions
when it = 1
    --{Play clip}
    send PlayWave 1 17
when it = 2
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 3
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 4
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 5
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 6
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 7
    get timerStart("single",Tdelay,0,background id 0)
    statusTimer = it
    timerID of this page = statusTimer
when it = 8
    send ProbRun("a")
    send buttonClick to ellipse "index" of this background
else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end conditions
end if
end

```

script ของ Button "Show" of background id 0 ที่ handle ButtonClick
to handle ButtonClick

```

system C_Drive --<current drive>
system statusTimer
local string ScmFile
statusTimer = null --- clear statusTimer
if sysLevel is Reader
mmclose sequencer
    mmclose WaveAudio
    get pagenumber of this page
    conditions
        when it = 2
            send ScmRun "ch10-1.SCM"
            send next
        when it = 3
            send ScmRun "ch10-2.SCM"
            send next
        when it = 4
            send ScmRun "ch10-3.SCM"
            send next
        when it = 5
            send ScmRun "ch10-4.SCM"
            send next
        when it = 6
            send ScmRun "ch10-5.SCM"
            send next
        when it = 7
            send ScmRun "ch10-6.SCM"
            send next
    else
    end conditions
end if
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

unit Question;

interface

uses

SysUtils, WinTypes, WinProcs, Messages, Classes, Graphics, Controls,
Forms, Dialogs, StdCtrls, ExtCtrls, Buttons;

type

```
TForm_question = class(TForm)
  Nbook1: TNotebook;
  Label1: TLabel;
  Label_bestscore: TLabel;
  Label_problem: TLabel;
  RdButton1: TRadioButton;
  RdButton2: TRadioButton;
  RdButton3: TRadioButton;
  RdButton4: TRadioButton;
  BitBtn1: TBitBtn;
  Lb_choice1: TLabel;
  Lb_choice2: TLabel;
  Lb_choice3: TLabel;
  Lb_choice4: TLabel;
  Label_userscore: TLabel;
  Label3: TLabel;
  Panel1: TPanel;
  Edit_getuser: TEdit;
  Label2: TLabel;
  BitBtn_ok: TBitBtn;
  Shape1: TShape;
  BitBtn_exit: TBitBtn;
  procedure FormActivate(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure BitBtn_okClick(Sender: TObject);
procedure BitBtn_exitClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure FormDestroy(Sender: TObject);
private
  BluePh : hPalette;
  NumBlues, BlueLevel, BlueDelta : Integer;
  procedure CreateParams(VAR Params: TCreateParams);
    override;
  function GetPalette : hPalette; override;
  procedure WM_EraseBkgnd(VAR Msg: TWM_EraseBkgnd);
    message WM_ERASEBKGND;
  { Private declarations }
public
  theend,score : integer;
  chapter,dir : string;
  Procedure runquestion(chap,section : string);
  Procedure readchoice(chap,section : string);
  Function chk_choice(chap,section : string):boolean;
  Procedure putdata(chap,info : string);
  procedure readhiscore(chap,section : string);
  Function chk_score(sco : string):boolean;
  { Public declarations }
end;

var
  Form_question: TForm_question;

implementation

{$R *.DFM}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{~~~~~}
{ uses for writeing background }
procedure TForm_question.CreateParams(VAR Params: TCreateParams);
begin
  Inherited CreateParams(Params);
  WITH Params.WindowClass DO Style := Style OR CS_VREDRAW;
end;

```

```

{~~~~~}
{ uses for write data to file }
Procedure TForm_question.putdata(chap,info : string);
var myfile : textfile;
    linetext,temptext,new_line : string;
begin
  temptext:="";
  new_line:=chr(13)+chr(10);
  assignfile(myfile,'c:\windows\usercai.ini');
  {$I-}
  Reset(myfile);
  {$I+}
  if ioresult = 0 then
  begin
    repeat
      readln(myfile,linetext);
      temptext := temptext + linetext + new_line;
    until (linetext[2] = chap) or (eof(myfile)); { find section }

    if not eof(myfile) then
    begin
      readln(myfile,linetext);
      temptext := temptext + info + new_line; { chage data }
    end
  end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else
    temptext := temptext + '[' + chap + ']' + new_line + info;

while not eof(myfile) do { set new data }
begin
    readln(myfile,linetext);
    temptext := temptext + linetext + new_line;
end;
rewrite(myfile); { write new data }
write(myfile,temptext);
end { if }
else
begin
    rewrite(myfile); { if no file then write new file }
    write(myfile,['+chap+']+ new_line + info);
end;
closefile(myfile);
end;

{-----}

Function TForm_question.chk_choice(chap,section : string):boolean;
var myfile : textfile;
    linetext : string;
begin
    chk_choice := false;
    Assignfile(myfile,dir+'.\cai\question\problem'+chap+'.txt');
    reset(myfile);
    if section = '10' then
        section:='@';
    repeat
        readln(myfile,linetext);
    until linetext[2] = section; { find section }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case linetext[4] of
    '1' : if rbutton1.checked = true then chk_choice := true;
    '2' : if rbutton2.checked = true then chk_choice := true;
    '3' : if rbutton3.checked = true then chk_choice := true;
    '4' : if rbutton4.checked = true then chk_choice := true;
end; {case}

rbutton1.checked:=false;
rbutton2.checked:=false;
rbutton3.checked:=false;
rbutton4.checked:=false;
closefile(myfile);
end;

{-----}
{ uses for read question form text file }
Procedure TForm_question.readchoice(chap,section :string);
var myfile : textfile;
    linetext,quest,choice : string;
    i : integer;
begin
    linetext:="";
    quest:="";
    choice:="";
    Assignfile(myfile,dir+':\cai\question\problem'+chap+'.txt');
    reset(myfile);
    if section = '10' then
        section:='@';
    repeat
        readln(myfile,linetext);
    until linetext[2] = section; { find section }

    readln(myfile,linetext);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while linetext <> '{}' do { find problem }
begin
  quest := quest + linetext + chr(10) + chr(13);
  readln(myfile,linetext);
end;
label_problem.caption := quest;

for i:=1 to 4 do { make 4 choice }
begin
  readln(myfile,linetext);
  while linetext <> '{}' do { get one choice }
  begin
    choice := choice + linetext + chr(10) + chr(13);
    readln(myfile,linetext);
  end;

  case i of { display choice }
    1 : lb_choice1.caption:=choice;
    2 : lb_choice2.caption:=choice;
    3 : lb_choice3.caption:=choice;
    4 : lb_choice4.caption:=choice;
  end; {case}
  choice:="";
end; {for}

closefile(myfile);
end;

{~~~~~}
{ uses for read hi score form text file }
procedure TForm_question.readhiscore(chap,section : string);
var myfile : textfile;
    linetext : string;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
  Assignfile(myfile,'c:\windows\usercai.ini');
  {$i-}
  reset(myfile);
  {$i+}
  if ioresult = 0 then
    begin
      repeat
        readln(myfile,linetext);
      until (linetext[2] = chap) or (Eof(myfile)); { find section }

      if not eof(myfile) then
        begin
          readln(myfile,linetext);
          label_bestscore.caption := linetext;
        end
      else
        label_bestscore.caption := 'ยังไม่มีคะแนนสูงสุด'; { no data }
      end { if ioresult }
    else
      rewrite(myfile);

    closefile(myfile);
  end;

  {~~~~~}

```

```

Procedure TForm_question.Runquestion(chap,section : string);

```

```

begin
  if section = '10' then
    section:='@';
    readchoice(chap,section);
  end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{-----}
{ uses for set intial data and set background }
procedure TForm_question.FormActivate(Sender: TObject);
begin
score:=0;
theend:=0;
{getdir(0,dir);}
if ParamCount = 0 then
begin
Application.messagebox('Error no parameter','Error Message',mb_ok);
exit;
end; {if}
chapter := copy(paramstr(1),2,1); { delete '/' }
dir := paramstr(2);
if chapter[1] in ['1'..'9','a','A'] then { chk chapter in 1 to 10 }
begin
readhiscore(chapter,'');
runquestion(chapter,'1'); { call procedure }
end
else
Application.messagebox('Error Parameter is invalid','Error Message',mb_ok);
end;

{-----}

```

```

{ uses for user click answer }
procedure TForm_question.BitBtn1Click(Sender: TObject);
begin
theend:=theend+1;
if theend <= 10 then
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if chk_choice(chapter,inttostr(theend)) then
  begin
    score:=score+1;
    label_userscore.caption:=inttostr(score);
  end;
if theend < 10 then
  runquestion(chapter,inttostr(theend+1))
else {1}
  begin
    if chk_score(label_userscore.caption) then
      nbook1.Activepage:='getinfo'
    else {2}
      begin
        if MessageDlg('คะแนนคุณต่ำกว่า คะแนนสูงสุด?',
          mtInformation, [mbYes], 0) = mrYes then
          close;
        end; { else 2 }
      end; { else 1 }
    end
  end;
end;

{-----}
{ uses for check score less or most }
function TForm_question.chk_score(sco : string):boolean;
var linetext : string;
    myfile : textfile;
    i,oldvalue : integer;
begin
  assignfile(myfile,'c:\windows\usercai.ini');
  {$I-}
  Reset(myfile);
  {$I+}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if ioresult = 0 then
begin
  readln(myfile,linetext);
  while ( linetext <> '[' + chapter + ']') and ( not eof(myfile)) do
    readln(myfile,linetext);
  if not eof(myfile) then
    begin
      readln(myfile,linetext);
      for i := 1 to length(linetext) do { find data }
      begin
        if linetext[i] = '=' then
          oldvalue := strtoint(copy(linetext,i+1,length(linetext)));
        end;
        if oldvalue < strtoint(sco) then { old data < new data }
          begin
            chk_score:=true;
          end
        else
            chk_score:=false;
        end { if eof }
      end { if ioresult }
    else { if ioresult }
      chk_score:=true;

      closefile(myfile);

end;

```

```

procedure TForm_question.BitBtn_okClick(Sender: TObject);
begin
  putdata(chapter,edit_getuser.text + '='+ inttostr(score));
  close;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm_question.BitBtn_exitClick(Sender: TObject);
```

```
begin
```

```
  close;
```

```
end;
```

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

```
begin
```

```
begin
```

```
  IF GetDeviceCaps(Canvas.Handle, RASTERCAPS) AND RC_PALETTE=0 THEN
```

```
    NumBlues := 256
```

```
  ELSE
```

```
    NumBlues := 64; {try 16, 32, 64, 128, or 234}
```

```
  BlueDelta := 256 DIV NumBlues;
```

```
  IF BlueDelta < 1 THEN BlueDelta := 1;
```

```
  GetPalette;
```

```
  Perform(WM_ERASEBKGD, Canvas.Handle, 0);
```

```
end;
```

```
end;
```

```
{~~~~~}
```

```
procedure TForm_question.FormDestroy(Sender: TObject);
```

```
begin
```

```
  DeleteObject(BluePh);
```

```
end;
```

```
{~~~~~}
```

```
{ uses for write background }
```

```
procedure TForm_question.WMEraseBkgnd(VAR Msg: TWmEraseBkgnd);
```

```
VAR
```

```
  OldPalette : hPalette;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Colr, OldColr : Integer;
N      : LongInt;
R      : TRect;

begin
Canvas.Handle := Msg.DC;
IF BluePh = 0 THEN
  BlueLevel := 255 - (NumBlues-1)*BlueDelta
ELSE
  BEGIN
    OldPalette := SelectPalette(Canvas.Handle,BluePh,False);
    RealizePalette(Canvas.Handle);
  END;
R := Bounds(0, 0, ClientWidth, 0);
OldColr := 0;
FOR N := 0 TO ClientHeight DO
  BEGIN
    Colr := N*NumBlues DIV ClientHeight;
    IF Colr = NumBlues THEN Dec(Colr);
    IF (Colr <> OldColr) OR (N = ClientHeight) THEN
      BEGIN
        OldColr := Colr;
        R.Top := R.Bottom;
        R.Bottom := N;
        IF BluePh = 0 THEN
          Canvas.Brush.Color := RGB( 0 , 0 , BlueLevel)
        ELSE Canvas.Brush.Color := PaletteIndex(Colr+10);
        BlueLevel := BlueLevel + BlueDelta;
        Canvas.FillRect(R);
      END;
    END;
  END;
END;

```

```

IF BluePH <> 0 THEN
  SelectPalette(Canvas.Handle, OldPalette, False);
Msg.Result := 1;
Canvas.Handle := 0;
end;

{~~~~~}

{ uses for get plate of computer }

function TForm_question.GetPalette: hPalette;
VAR
  N, PalSize : Integer;
  BluePal : PLogPalette;
begin
  IF BluePH <> 0 THEN Result := BluePH
  ELSE IF GetDeviceCaps(Canvas.Handle, RASTERCAPS) AND
    RC_PALETTE=0 THEN Result := 0
  ELSE
    BEGIN
      PalSize := SizeOf(TLogPalette) +
        (10+NumBlues)*SizeOf(TPaletteEntry);
      GetMem(BluePal, PalSize);
      FillChar(BluePal^, PalSize, 0);
      BluePal^.palVersion := $300;
      BluePal^.palNumEntries := 10+NumBlues;
      GetSystemPaletteEntries(Canvas.Handle, 0, 10,
        BluePal^.PalPalEntry);
      BlueLevel := 255 - (NumBlues-1)*BlueDelta;
      {$R-}
      FOR N := 0 TO NumBlues-1 DO
        WITH BluePal^.PalPalEntry[N+10] DO
          BEGIN
            peBlue := BlueLevel;

            peFlags := PC_NOCOLLAPSE;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BlueLevel := BlueLevel + BlueDelta;
```

```
END;
```

```
{SR+}
```

```
BluePh := CreatePalette(BluePal^);
```

```
FreeMem(BluePal, PalSize);
```

```
Result := BluePh;
```

```
END;
```

```
end;
```

```
end.
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. นฤก กระจาย ,“ การเขียนโปรแกรมด้วยบอร์แลนด์ปาสคาล 7.0 ”, ซีเอ็ดยูเคชั่น จำกัด, 645 หน้า, 2537.
2. บุญเลิศ เอี่ยมทัศนาศนา, “ Delphi ” , ซีเอ็ดยูเคชั่น จำกัด, 373 หน้า, 2539.
3. Andrew Wozniwicz and Namir Shammas,“Teach Yourself Borland Delphi in 21 Days”,Sams Publishing,1995.
4. Dottie Natal and Erik Reitan,“Special Edition Using Asymetrix Multimedia ToolBook 4,” Que Corporation, 307 p., 1995.
5. Gray Cornell and Troy Strain ,“ Delphi Nuts & Bolts : For Experienced Programmers”, McGraw-Hill, 912p. 1995.
6. Neil Rubenking,“Delphi Programming Problem Solvex ”, IDG Books Worldwidge , Inc., 604 p. 1996



กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณ ดร. วรวัฒน์ ลิ้มโกศา ซึ่งเป็น อาจารย์ที่ปรึกษาที่ให้คำปรึกษาแนะนำ และช่วยแก้ไขปัญหาที่เกิดขึ้น ตรวจสอบแก้ไขสิ่งที่บกพร่องจนทำให้โครงการพิเศษครั้งนี้สำเร็จได้ด้วยดี

ขอขอบพระคุณ คุณ รมรงค์ ชนากร ซึ่งเป็นผู้ให้คำปรึกษาและสถานที่ในการปฏิบัติงาน ขอขอบพระคุณ คุณ วีระยุทธ จีวรมงคล ซึ่งช่วยอนุเคราะห์ในการเขียนแผ่น CD รวมทั้งคุณ รุ่งโรจน์ เลิศรัตนศาสตร์ ที่ให้ยืมเครื่องคอมพิวเตอร์ที่ใช้ในการปฏิบัติงานและ คุณ ธนพรรณพร้อมมูล ที่ช่วยในเรื่องการจัดพิมพ์เอกสาร

ขอบคุณเพื่อนๆ น้องๆ ที่เป็นกำลังใจให้คำแนะนำและช่วยแก้ไขปัญหาที่เกิดในการทำงาน ขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การศึกษาศึกษาและวิทยากรความรู้ในด้านต่างๆ

ท้ายนี้ขอกราบขอบพระคุณ คุณพ่อ คุณแม่และญาติพี่น้องที่ให้การสนับสนุนปัจจัยต่างๆ ในการเรียน คำสอน พร้อมทั้งกำลังใจตลอดมา

จตุรงค์ พวงนาค

อนุชา บุญชูโต

อิทธิศักดิ์ จำรัสประเสริฐ