

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ความก้าวหน้าของฐานข้อมูลเชิงวัตถุ

ADVANCES IN OBJECT DATABASE



รฟ.
ร 849 ค
2549

เลขหมู่.....
เลขทะเบียน 72679
วัน,เดือน,ปี 21 ส.ย. 2550

b. 11221252
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความก้าวหน้าของฐานข้อมูลเชิงวัตถุ
ADVANCES IN OBJECT DATABASE



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมคอมพิวเตอร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมคอมพิวเตอร์

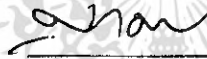
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ความก้าวหน้าของฐานข้อมูลเชิงวัตถุ

ADVANCES IN OBJECT DATABASE

ผู้จัดทำ

1. นางสาวสุมาลี แซ่เหี้ย รหัสประจำตัว 46010862
2. นางสาวสุรียพร สร้อยสังวาลย์ รหัสประจำตัว 46010885



อาจารย์ที่ปรึกษา

(รศ.ดร.ศุภมิตร จิตตะยโสธร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความก้าวหน้าของฐานข้อมูลเชิงวัตถุ

นางสาวศุมาลี	แจ่มเหี้ย	46010862
นางสาวสุรียพร	สร้อยสังวาลย์	46010885
รศ.ดร.ศุภมิตร	จิตตะยโสธร	อาจารย์ที่ปรึกษา
ปีการศึกษา 2549		

บทคัดย่อ

ในปัจจุบัน การพัฒนาโปรแกรมส่วนใหญ่ ยึดหลักการของ โปรแกรมเชิงวัตถุมากขึ้น จนหลักการดังกล่าวได้เริ่มเข้ามามีบทบาทสำคัญในการพัฒนาระบบฐานข้อมูลเชิงวัตถุ (Object – Database) โดยการนำแนวคิดเชิงวัตถุเข้ามาใช้กับระบบฐานข้อมูล ซึ่งระบบจัดการฐานข้อมูลเชิงวัตถุสามารถรองรับการใช้งานข้อมูลที่มีความซับซ้อน ได้มีประสิทธิภาพมากกว่าฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

โครงการนี้จึงได้ทำการศึกษาแนวความคิดของฐานข้อมูลเชิงวัตถุ โดยได้เลือกใช้ระบบจัดการฐานข้อมูล Cache และได้ทำการศึกษาวิธีการติดต่อใช้งานระบบจัดการฐานข้อมูลเชิงวัตถุ โดยใช้ภาษาโปรแกรมจาวาในการติดต่อกับฐานข้อมูล เนื่องจากเป็นภาษาที่สนับสนุนแนวคิดเชิงวัตถุ รวมถึงศึกษาและออกแบบคลาสไดอะแกรมเพื่อนำมาสร้างฐานข้อมูลเชิงวัตถุสำหรับทดสอบตามแนวความคิดเชิงวัตถุของฐานข้อมูลเชิงวัตถุ เพื่อให้ได้มาซึ่งพื้นฐานสำคัญที่จะนำไปพัฒนาที่มีความก้าวหน้าในขั้นสูงต่อไป หลังจากที่ได้ทำการศึกษา ก็พบว่าฐานข้อมูลเชิงวัตถุ นั้นพัฒนาขึ้นเพื่อในงานกับข้อมูลที่มีความซับซ้อน ดังเช่น ข้อมูลรูปภาพ เสียง วิดีโอ เป็นต้น ในขั้นต่อไปนั้นจึงได้ทำการออกแบบและทดสอบฐานข้อมูลเชิงวัตถุกับข้อมูลที่เป็นรูปภาพตามแนวคิดเชิงวัตถุ โดยสร้างแอปพลิเคชันการค้นหารูปภาพโดยรูปภาพขึ้น ซึ่งผลที่ได้ นั้นสนับสนุนแนวคิดเชิงวัตถุ และง่ายต่อการจัดการกับ โปรแกรมแอปพลิเคชันอีกด้วย

ADVANCES IN OBJECT DATABASE

Miss.Sumalee Saehear 46010862
Miss.Sureeporn Soysangworn 46010885
Assoc. Prof. Dr. Suphamit Chittayasothorn Advisor
Academic Year 2006

ABSTRACT

Nowadays almost programming development adheres object oriented programming principle more until it begins to have impact on object database by using object oriented concept in object database which supports complex data better than relational database.

In this project, we studied object oriented database concept by using Caché database and contact way to use object database by Java programming which supports object oriented concept. Including studied and designed class diagram to build object database for testing object oriented concept of object database to get important basic for advance project development in the future. After we have studied ,we found that object database is developed for complex data such as image, sound, VDO etc. So we designed and tested object database with image data following object oriented concept by building “search image by image” application. Finally receiving result supports object oriented concept and manages application program easier too.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้เป็นอย่างดี ด้วยคำแนะนำ คำปรึกษา และความช่วยเหลือจาก อาจารย์ ศุภมิตร จิตตะยโสธร ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการที่เอาใจใส่และเป็นห่วงเสมอมา ซึ่งขอกราบขอบพระคุณเป็นอย่างสูง และขอกราบขอบพระคุณ อาจารย์ ธีรฉัตร จิตต์โสภักตร์ ที่ให้คำแนะนำที่เป็นประโยชน์ในงานด้านอิมเมจ

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้ ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ขอกราบขอบพระคุณ บิดา มารดา และครอบครัวอันเป็นที่เคารพรักยิ่ง ที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้สามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี คุณค่าและประโยชน์อันพึงมาจากปริญญานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกท่านและผู้ที่เกี่ยวข้องศึกษา

สุมาลี แซ่เหี้ย

สุรีพร สร้อยสังวาลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูปภาพ	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	2
1.4 ขอบเขตของโครงการ	2
1.5 วิธีการดำเนินงาน	2
1.6 ส่วนประกอบของรายงาน	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 แนวความคิดเชิงวัตถุ	4
2.1.1 OID	4
2.1.2 คุณสมบัติ (Property)	4
2.1.3 เมธอด (Method)	4
2.1.4 เมสเซจ (Message)	4
2.1.5 คลาส (Class)	5
2.1.6 เอนแคปซูเลชัน (Encapsulation)	5
2.1.7 การสืบทอดคุณสมบัติ (Inheritance)	5
2.1.8 พอลิมอร์ฟิซึม (Polymorphism)	6
2.1.9 ความสัมพันธ์ระหว่างออบเจกต์ (Association)	6
2.2 ฐานข้อมูลเชิงวัตถุ (Object Oriented Database)	7
2.2.1 แนวความคิดของฐานข้อมูลเชิงสัมพันธ์ (Relational Database)	8
2.2.2 แนวความคิดของฐานข้อมูลเชิงวัตถุ (Object Oriented Database)	8
2.2.3 ข้อดีของฐานข้อมูลเชิงวัตถุ	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3 การจัดเก็บรูปภาพ	12
3.1 วิธีที่ 1 การจัดเก็บและสืบค้นภาพทั่วทัศน์ทั่วไป	12
3.1.1 การแปลง Color Space และ ฮิสโตแกรม	12
3.1.2 วิธีการจัดเก็บและค้นหาภาพค้นหาภาพ	13
3.2 วิธีที่ 2 การจัดเก็บและสืบค้นภาพใบหน้าของคน	18
3.2.1 การจัดเก็บข้อมูลรูปภาพลงในฐานข้อมูล	18
3.2.2 การเรียกค้นข้อมูลรูปภาพ	20
บทที่ 4 Caché	22
4.1 บทนำ	22
4.2 Data Modeling-Relational or Object Access	23
4.3 Caché Object Model และ Object Programming	23
4.4 SQL Access	25
4.5 Caché Object	26
4.6 Caché Application Server	27
4.7 JAVA	28
4.7.1 Object Access through Projected Classes	28
4.7.2 Caché Methods Written in Java	29
บทที่ 5 การทดสอบ	30
5.1 การทดสอบที่ 1	30
5.1.1 การทดสอบ	30
5.1.2 ผลจากการทดลองการทดสอบที่ 1	37
5.2 การทดสอบที่ 2	40
5.2.1 การทดสอบ	40
5.2.2 วิธีทำการทดสอบ	40
5.2.3 ผลจากการทดลองการทดสอบที่ 2	47
บทที่ 6 สรุป	53
6.1 สรุปผลการทดลอง	53
6.2 แนวทางการพัฒนาต่อ	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
ภาคผนวก ก. การติดตั้งและใช้งานฐานข้อมูล Caché	56
การดาวน์โหลดฐานข้อมูล Caché และเอกสารคู่มือการใช้งาน	56
การ Install ระบบฐานข้อมูล Caché	57
วิธีการใช้งาน โปรแกรม Caché Studio	62
การ Install และการใช้งาน RoseLink	77
ภาคผนวก ข. การติดต่อใช้งานฐานข้อมูล Caché กับ JAVA	84
การใช้งานกับโปรแกรมจาวา	84
การเพิ่มไลบรารีของ Caché เข้าสู่จาวา	85
ภาคผนวก ค. การเขียนโปรแกรมในการจัดเก็บแล้วค้นหารูปภาพ	91
ภายในฐานข้อมูลเชิงวัตถุ Caché	91
ภายในโปรแกรมภาษา จาวา	110
ภาคผนวก ง. การใช้งานโปรแกรมที่พัฒนาขึ้น บรรณานุกรม	113 123

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2-1 ตารางเปรียบเทียบแนวคิดเชิงวัตถุและแนวคิดเชิงสัมพันธ์	10
ตารางที่ 2-2 ตารางเปรียบเทียบฐานข้อมูลเชิงวัตถุและฐานข้อมูลเชิงสัมพันธ์	10
ตารางที่ 3-1 ตารางแสดงการจับคู่อินเตอร์เฟซชั้นของภาพในแต่ละส่วน	15
ตารางที่ 4-1 ตารางเปรียบเทียบการใช้ภาษา SQL แบบ TRADITIONAL SQL กับ OBJECT EXTENDED SQL	26



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 2-1 ความสัมพันธ์แบบ one-to-many และ many-to-one	7
รูปที่ 2-2 ความสัมพันธ์แบบ many-to-many	7
รูปที่ 3-1 แสดงแนวคิดในการแบ่งขอบเขตของภาพออกเป็น 5 ส่วน	15
รูปที่ 3-2 แสดงขั้นตอนการทำงานของการทำงานของการเก็บภาพแล้วฮิสโตแกรมของภาพหนึ่งจากฐานข้อมูล	16
รูปที่ 3-3 แสดงขั้นตอนการทำงานของการทำงานของการค้นหาและเรียกภาพหนึ่งจากฐานข้อมูล	17
รูปที่ 3-4 รูปภาพดั้งเดิม	18
รูปที่ 3-5 รูปภาพระดับสีเทา	18
รูปที่ 3-6 แสดงการสร้างค่าตัวแทนรูปภาพจากภาพอินพุท	19
รูปที่ 3-7 แสดงการนำภาพอินพุทมาแปลงเป็นข้อมูลอาเรย์ขนาด 10*10	19
รูปที่ 3-8 แสดงขั้นตอนการทำงานของการทำงานของการเก็บภาพและค่าตัวแทนของภาพหนึ่งจากฐานข้อมูล	20
รูปที่ 3-9 แสดงขั้นตอนการทำงานของการทำงานของการค้นหาและเรียกภาพหนึ่งจากฐานข้อมูล	21
รูปที่ 4-1 แสดงตัวอย่างของคลาสและคุณสมบัติภายใน	24
รูปที่ 5-1 คลาสไดอะแกรมของระบบลงทะเบียนนักศึกษา	31
รูปที่ 5-2 การ Export จาก Ration Rose ไปยัง Cache	32
รูปที่ 5-3 แสดงการเลือก คลาสที่จะ export ไปเป็นคลาสที่ใช้ในการเก็บข้อมูล	33
รูปที่ 5-4 แสดงตัวอย่างคำสั่งที่ใช้สร้างฐานข้อมูลเชิงวัตถุ	33
รูปที่ 5-5 คลาส Student ใน Cache	34
รูปที่ 5-6 คลาส Master ใน Cache Inherit มาจากคลาส Student	34
รูปที่ 5-7 ตัวอย่างโค้ด โปรแกรมที่ cache สร้างเป็นภาษาจาวา	35
รูปที่ 5-8 การเพิ่มและค้นหาประวัตินักศึกษาในฐานข้อมูล	37
รูปที่ 5-9 การลงทะเบียน	38
รูปที่ 5-10 การเพิ่ม เปลี่ยน ถอน การลงทะเบียน	39
รูปที่ 5-11 คลาสไดอะแกรมของระบบการจัดเก็บและการสืบค้นข้อมูลรูปภาพ	41
รูปที่ 5-12 แสดงตัวอย่างโครงสร้างคำสั่ง	42
รูปที่ 5-13 แสดงตัวอย่างคำสั่งที่ใช้สร้างฐานข้อมูลเชิงวัตถุ	43
รูปที่ 5-14 ตัวอย่างโค้ด โปรแกรมที่ cache สร้างเป็นภาษาจาวา	44
รูปที่ 5-15 หน้าจอสำหรับการเลือกการทำงาน	48
รูปที่ 5-16 การจัดเก็บข้อมูลรูปภาพ	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 5-17 การค้นหารูปภาพจากชื่อและประเภทของรูปภาพ	49
รูปที่ 5-18 ค่า BitmapFileHeader ที่ได้จากการ Search	49
รูปที่ 5-19 ค่า BitmapInfoHeader ที่ได้จากการ Search	50
รูปที่ 5-20 การค้นหารูปภาพจากรูปภาพ (ภาพวิว)	51
รูปที่ 5-21 การค้นหารูปภาพจากรูปภาพ (ภาพใบหน้าคน)	52
รูปที่ ก-1 แสดงเมนูการเลือกความไวของระบบฐานข้อมูล Cache	56
รูปที่ ก-2 แสดงเว็บเพจการดาวน์โหลดฐานข้อมูล Cache	56
รูปที่ ก-3 แสดงการเข้าถึงเมนูเพื่อการดาวน์โหลดเอกสารคู่มือการใช้งาน	57
รูปที่ ก-4 แสดงหน้าเว็บเพจการดาวน์โหลดเอกสารคู่มือการใช้งาน	57
รูปที่ ก-5 ไฟล์ Install ฐานข้อมูล Cache	57
รูปที่ ก-6 ไอคอนของการเลือกภาษาสำหรับการ Install ฐานข้อมูล Cache	58
รูปที่ ก-7 แสดงการเข้าสู่การ Install ฐานข้อมูล Cache	59
รูปที่ ก-8 แสดงดีฟอลต์ path ในการ Install Cache	59
รูปที่ ก-9 ระบบกำลังทำการ Install	60
รูปที่ ก-10 แสดงการ Install เสร็จสิ้น	60
รูปที่ ก-11 เว็บเพจคู่มือการใช้งาน Cache	61
รูปที่ ก-12 แสดงหน้าต่างโปรแกรม Cache Studio	62
รูปที่ ก-13 แสดงไอคอนของการ Log on เข้าสู่ระบบ	62
รูปที่ ก-14 แสดงไอคอนการเลือก Namespace	63
รูปที่ ก-15 แสดงหน้าจอของโปรแกรมที่พร้อมใช้งาน	63
รูปที่ ก-16 แสดงการเลือกเมนูการเริ่มต้นเข้าสู่การสร้างงาน	64
รูปที่ ก-17 แสดงการเลือกชนิดไฟล์งานที่ต้องการสร้าง จากรูปเป็นการเลือก CacheClassDefination	64
รูปที่ ก-18 แสดงหน้าแรกของ โปรแกรม Wizard ที่ช่วยเหลือในการสร้างคลาส	65
รูปที่ ก-19 แสดงหน้าการเลือกชนิดของคลาสของ โปรแกรม Wizard ที่ช่วยเหลือใน การสร้างคลาส	65
รูปที่ ก-20 แสดงโค้ดโปรแกรมหลังจาก โปรแกรม Wizard การสร้างคลาสเสร็จสิ้น	66
รูปที่ ก-21 แสดงการเลือกเมนูการสร้าง property	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ ก-22 แสดงหน้าแรกของโปรแกรม Wizard ที่ช่วยเหลือในการสร้าง property	67
รูปที่ ก-23 แสดงหน้าการเลือกชนิดของ property ของโปรแกรม Wizard การสร้าง property	67
รูปที่ ก-24 แสดง ไดอะล็อกเมื่อกดปุ่ม Browse...	68
รูปที่ ก-25 แสดง ใ้ค้ด โปรแกรมหลังจาก โปรแกรม Wizard การสร้าง property สำเร็จ	68
รูปที่ ก-26 การเลือกเมนูเพื่อทำการสร้างเมธอด	68
รูปที่ ก-27 แสดงหน้าแรกของโปรแกรม Wizard ที่ช่วยเหลือการสร้างเมธอด	69
รูปที่ ก-28 แสดงหน้าการเลือกค่ารีเทิร์นและค่าพารามิเตอร์ที่ต้องการพาสสู่เมธอด ของโปรแกรม Wizard การสร้างเมธอด	69
รูปที่ ก-29 แสดงหน้าการเลือกภาษาโปรแกรมที่ต้องการจะเขียนภายในเมธอด ของโปรแกรม Wizard การสร้างเมธอด	70
รูปที่ ก-30 แสดงหน้าการอิมพลีเมนต์ ใ้ค้ดภายในเมธอด ของโปรแกรม Wizard การสร้างเมธอด	70
รูปที่ ก-31 แสดง ใ้ค้ด โปรแกรมเมื่อ โปรแกรม Wizard การสร้างเมธอดเสร็จสิ้น	71
รูปที่ ก-32 แสดงการเลือกเมนูเพื่อสร้าง Query	71
รูปที่ ก-33 แสดงหน้าแรกของโปรแกรม Wizard เพื่อช่วยเหลือในการสร้าง Query	72
รูปที่ ก-34 แสดงหน้าของการ สร้างพารามิเตอร์อินพุท ของโปรแกรม Wizard การสร้าง Query	72
รูปที่ ก-35 แสดงพารามิเตอร์อินพุทที่ถูกสร้างขึ้น	73
รูปที่ ก-36 แสดงหน้าการเลือกคอลัมน์ที่ต้องการแสดงในผลลัพธ์ของการ Query ของโปรแกรม	73
รูปที่ ก-37 แสดงหน้าของการสร้างเงื่อนไขของ Query ของโปรแกรม Wizard การสร้าง Query	74
รูปที่ ก-38 แสดงหน้าของการเลือกการจัดเรียงข้อมูล ของ โปรแกรม Wizard การสร้าง Query	74
รูปที่ ก-39 แสดง ใ้ค้ด โปรแกรมเมื่อ โปรแกรม Wizard การสร้าง Query เสร็จสิ้น	75
รูปที่ ก-40 แสดงการเลือกเมนูการสร้าง Projection	75
รูปที่ ก-41 แสดงหน้าแรกของโปรแกรม Wizard ในการสร้าง Projection	76
รูปที่ ก-42 แสดงหน้าของการใส่ path ที่ต้องการเก็บ ใ้ค้ดภาษาจาวาของโปรแกรม Wizard ในการสร้าง Projection	76
รูปที่ ก-43 แสดง ใ้ค้ด โปรแกรมเมื่อ โปรแกรม Wizard ในการสร้าง Projection เสร็จสิ้น	77
รูปที่ ก-44 แสดงการเลือก Cache ภายใน Rational Rose	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ ก-45 แสดงการเลือกเมนูเพื่อต้องการ export คลาสไคอะแกรมไปที่ Cache	78
รูปที่ ก-46 แสดงคลาสตัวอย่างที่จะทำการ export ไปยัง Cache	79
รูปที่ ก-47 แสดงรูป Rose Cache Link หลังจากเลือก Export classes to Cache แล้ว	79
รูปที่ ก-48 แสดงไคอะล็อกการเลือกคลาสที่จะทำการ export ไปยัง Cache	80
รูปที่ ก-49 แสดงการเลือกคลาสที่เป็นคลาสฐานข้อมูล	80
รูปที่ ก-50 แสดงไคอะล็อกการเลือก Server Connection	81
รูปที่ ก-51 แสดงการล็อกอินเพื่อเข้าสู่ฐานข้อมูล Cache	81
รูปที่ ก-52 แสดงการเลือก Namespace	82
รูปที่ ก-53 แสดงการประมวลผลในการ export คลาสไปยัง Cache ได้สำเร็จเรียบร้อย	82
รูปที่ ก-54 แสดงคลาสไคอะแกรมที่ถูกสร้างขึ้นมาจากการ export มาจากคลาสไคอะแกรม	83
รูปที่ ก-55 แสดงตัวอย่างโค้ดที่ได้จากการ export มาจากคลาสไคอะแกรม	83
รูปที่ ข-1 แสดงโปรแกรม eclipse	84
รูปที่ ข-2 แสดงการเลือกเมนูเพื่อทำการเพิ่มไลบรารี Cache เข้าสู่ไฟล์โปรเจกจาวา	85
รูปที่ ข-3 แสดงไคอะล็อกการเพิ่มไลบรารี	86
รูปที่ ข-4 แสดงหน้าของการเข้าสู่การสร้าง User Libraries	86
รูปที่ ข-5 แสดงหน้าของการสร้าง User Libraries	87
รูปที่ ข-6 แสดงไคอะล็อกของการตั้งชื่อไลบรารี	87
รูปที่ ข-7 แสดงหน้าของการเข้าสู่การเพิ่มไฟล์ไลบรารีเข้าสู่ระบบ	88
รูปที่ ข-8 แสดงการเลือกไฟล์ไลบรารีของ Cache	89
รูปที่ ข-9 แสดงหน้าของการเพิ่มไลบรารีเสร็จเรียบร้อยแล้ว	90
รูปที่ ข-10 แสดงไลบรารีที่เพิ่มเข้าไปสู่โปรเจกงานเป็นที่เรียบร้อยแล้ว	90
รูปที่ ง-1 หน้าจอหลักในการทำงาน	113
รูปที่ ง-2 หน้าจอในการเพิ่มข้อมูลรูปภาพ	114
รูปที่ ง-3 หน้าจอการเลือกรูปที่ต้องการจัดเก็บลงในฐานข้อมูล	114
รูปที่ ง-4 การใส่ข้อมูลในการเพิ่มข้อมูลรูปภาพ	115
รูปที่ ง-5 แสดงหน้าจอการค้นหารูปภาพจากชื่อของรูปภาพ	116
รูปที่ ง-6 ค่า Histogram ของรูปภาพในฐานข้อมูล	117
รูปที่ ง-7 ค่า BitmapFileHeader ของรูปภาพในฐานข้อมูล	117

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ ง-8 แสดงค่า BitmapInfoHeader ของรูปภาพในฐานะข้อมูล	118
รูปที่ ง-9 หน้าจอการค้นหารูปภาพ จากรูปภาพ	119
รูปที่ ง-10 การค้นหารูปภาพจากรูปภาพ	120
รูปที่ ง-11 ผลจากการค้นหาภายในฐานข้อมูล	120
รูปที่ ง-12 เปรียบเทียบฮิสโตแกรมของภาพที่ต้องการหาแล้วภาพภายในฐานข้อมูล	121
รูปที่ ง-13 แสดงรูปภาพอินพุตที่จัดเก็บในฐานข้อมูล	121
รูปที่ ง-14 แสดงผลการค้นหารูปภาพใบหน้าคนที่มีความคล้ายกัน	122
รูปที่ ง-15 แสดงผลการค้นหารูปภาพวิวทัศน์ที่มีความคล้ายกัน	122



บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของโครงการ

ปัจจุบัน เทคโนโลยีสารสนเทศได้เข้ามามีบทบาทสำคัญในการดำเนินกิจกรรม และระบบงานต่างๆ ของแต่ละองค์กร ซึ่งการจะทำให้ระบบสารสนเทศเหล่านั้นมีประสิทธิภาพเพียงพอที่จะสนับสนุนระบบงานได้ จำเป็นจะต้องมีการจัดการข้อมูลอย่างมีประสิทธิภาพ ด้วยสาเหตุเหล่านี้ จึงเป็นที่มาของการพัฒนาระบบการจัดการฐานข้อมูลขึ้น โดยในอดีตที่ผ่านมา ได้มีการค้นคว้า และพัฒนาระบบฐานข้อมูลขึ้นหลายรูปแบบ และหลายแนวความคิด อย่างไรก็ตาม ระบบฐานข้อมูลที่ได้รับความนิยม และแพร่หลายสูงสุดในการใช้งานคือ ระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database System) ในขณะเดียวกันหลักการคิด และมุมมองข้อมูลในรูปแบบเชิงวัตถุ ทำให้การพัฒนาโปรแกรมส่วนใหญ่ในยุคปัจจุบัน ล้วนแต่ยึดหลักการ โปรแกรมเชิงวัตถุมากขึ้น จนหลักการดังกล่าวได้เริ่มเข้ามามีบทบาทสำคัญในการพัฒนาระบบฐานข้อมูลเชิงวัตถุ (Object-Database) ซึ่งจะสามารถรองรับการใช้งานข้อมูลที่มีความซับซ้อน ได้มากกว่าฐานข้อมูลเชิงสัมพันธ์

โครงการนี้จึงได้ทำการศึกษาแนวความคิดของฐานข้อมูลเชิงวัตถุ โดยได้เลือกใช้ระบบจัดการฐานข้อมูล Caché โดยได้ทำการศึกษาวิธีการใช้ระบบจัดการฐานข้อมูลโดยใช้ภาษาจาวาในการติดต่อกับฐานข้อมูล รวมถึงศึกษาและออกแบบคลาสไคอะแกรมเพื่อนำมาสร้างฐานข้อมูลเชิงวัตถุสำหรับทดสอบตามแนวความคิดเชิงวัตถุของฐานข้อมูลเชิงวัตถุ เพื่อให้ได้มาซึ่งพื้นฐานสำคัญที่จะนำไปพัฒนางานที่มีความก้าวหน้าในขั้นสูงต่อไป กับข้อมูลที่มีความซับซ้อนอย่างเช่น รูปภาพ เสียง วิดีโอ เป็นต้น

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อศึกษาแนวความคิดของฐานข้อมูลเชิงวัตถุ (Object-Oriented Database)
- 1.2.2 เพื่อศึกษาซอฟต์แวร์ที่ใช้จัดการระบบฐานข้อมูลเชิงวัตถุ Caché
- 1.2.3 เพื่อศึกษาการใช้ภาษาจาวาติดต่อกับฐานข้อมูลเชิงวัตถุ
- 1.2.4 เพื่อทดสอบฐานข้อมูลเชิงวัตถุตามแนวความคิดเชิงวัตถุ
- 1.2.5 เพื่อศึกษาความก้าวหน้าของฐานข้อมูลเชิงวัตถุ

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ได้รับความรู้เกี่ยวกับแนวคิดของฐานข้อมูลเชิงวัตถุ
- 1.3.2 ได้รับความรู้เกี่ยวกับการใช้ระบบจัดการฐานข้อมูลเชิงวัตถุ Caché
- 1.3.3 ได้รับความรู้ความเข้าใจเกี่ยวกับการใช้ภาษาจาวาในการติดต่อฐานข้อมูลเชิงวัตถุ
- 1.3.4 สามารถวิเคราะห์และออกแบบฐานข้อมูลเชิงวัตถุได้
- 1.3.5 ได้รับความรู้ในเทคโนโลยีใหม่ ๆ ในฐานข้อมูลเชิงวัตถุ กับการเก็บข้อมูลที่มีความซับซ้อนมากยิ่งขึ้น

1.4 ขอบเขตของโครงการงาน

ศึกษาแนวคิดเชิงวัตถุ แนวคิดของฐานข้อมูลเชิงวัตถุ และทดสอบสิ่งที่ศึกษามาทั้งหมดด้วยซอฟต์แวร์ที่สนับสนุนระบบจัดการฐานข้อมูลเชิงวัตถุ ซึ่งก็คือ Caché ศึกษาการติดต่อกับฐานข้อมูลเชิงวัตถุด้วยภาษาจาวาและทดสอบการติดต่อฐานข้อมูลโดยการสร้างแอปพลิเคชันให้รองรับฐานข้อมูลเชิงวัตถุและการเขียนโปรแกรมเชิงวัตถุ โดยเริ่มด้วยการใช้ฐานข้อมูล Caché เก็บข้อมูลที่เป็นข้อความธรรมดา แล้วพัฒนาขึ้นโดยการเก็บข้อมูลที่มีความซับซ้อนดังเช่น ข้อมูลรูปภาพ เป็นต้น

1.5 วิธีการดำเนินการงาน

- 1.5.1 ศึกษาหลักการและแนวคิดระบบฐานข้อมูลเชิงวัตถุ
- 1.5.2 ศึกษาการวิเคราะห์และออกแบบเชิงวัตถุ ตามหลักของ UML
- 1.5.3 ศึกษาซอฟต์แวร์ที่สนับสนุนระบบจัดการฐานข้อมูลเชิงวัตถุ Caché
- 1.5.4 ศึกษาการติดต่อฐานข้อมูลด้วยภาษาที่สนับสนุนการเขียน โปรแกรมเชิงวัตถุจาวา
- 1.5.5 ทดสอบระบบ โดยการสร้างแอปพลิเคชันที่มีข้อมูลเป็นข้อความธรรมดา โดยการออกแบบฐานข้อมูลเชิงวัตถุนี้ ได้ใช้คลาสไคอะแกรมของ UML เนื่องจากเป็นวิธีที่เหมาะสมกับการออกแบบฐานข้อมูลเชิงวัตถุและ OODBMS (Object-Oriented Database Management System) ได้สนับสนุนการแปลงคลาสไคอะแกรมเป็น Object Schema ที่ใช้ในการเก็บข้อมูลของฐานข้อมูลเชิงวัตถุได้ทันที
- 1.5.6 เมื่อทดสอบให้ฐานข้อมูลเชิงวัตถุเก็บข้อมูลที่เป็นข้อความธรรมดาได้แล้ว จากนั้นจะพัฒนาโดยการให้เก็บข้อมูลที่มีความซับซ้อนมากยิ่งขึ้นดังเช่น ข้อมูลรูปภาพ เป็นต้น โดยหลักการทุกอย่างยังสนับสนุนแนวคิดเชิงวัตถุ

1.6 ส่วนประกอบของรายงาน

รายงานฉบับนี้แบ่งเป็น 6 บท ได้แก่

เนื้อหาในบทที่ 1 กล่าวถึงความเป็นมาของปัญหา วัตถุประสงค์ของโครงการ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงการ วิธีการดำเนินงาน และส่วนประกอบของรายงานฉบับนี้

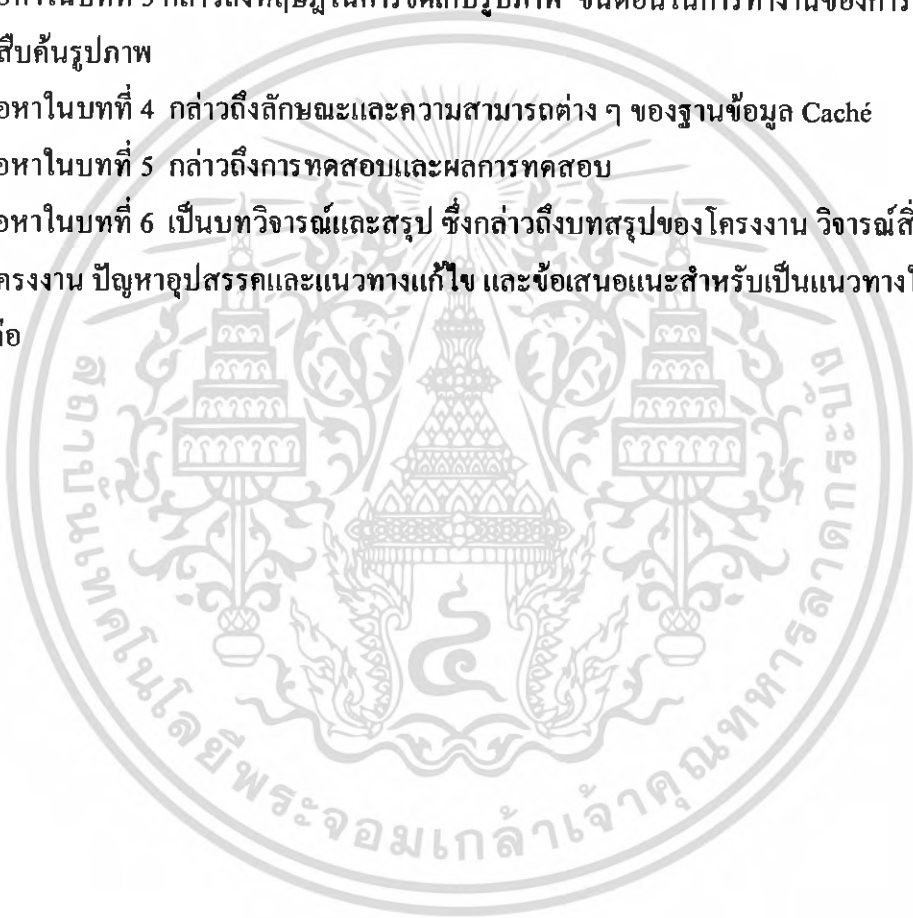
เนื้อหาในบทที่ 2 กล่าวถึงทฤษฎีพื้นฐานเชิงวัตถุ ซึ่งประกอบด้วยทฤษฎีของแนวคิดเชิงวัตถุ ฐานข้อมูลเชิงวัตถุ ระบบจัดการฐานข้อมูลเชิงวัตถุ (Cache) ภาษาจาวาที่ใช้ในการติดต่อฐานข้อมูลเชิงวัตถุ

เนื้อหาในบทที่ 3 กล่าวถึงทฤษฎีในการจัดเก็บรูปภาพ ขั้นตอนในการทำงานของการจัดเก็บและสืบค้นรูปภาพ

เนื้อหาในบทที่ 4 กล่าวถึงลักษณะและความสามารถต่าง ๆ ของฐานข้อมูล Cache

เนื้อหาในบทที่ 5 กล่าวถึงการทดสอบและผลการทดสอบ

เนื้อหาในบทที่ 6 เป็นบทวิจารณ์และสรุป ซึ่งกล่าวถึงบทสรุปของโครงการ วิจารณ์สิ่งที่ได้รับจากโครงการ ปัญหาอุปสรรคและแนวทางแก้ไข และข้อเสนอแนะสำหรับเป็นแนวทางในการพัฒนาต่อ



บทที่ 2

แนวความคิดเชิงวัตถุ

2.1 แนวความคิดเชิงวัตถุ จะประกอบด้วยองค์ประกอบเหล่านี้

Object is something which has state , behavior ,identity
ออบเจกต์ คือ สิ่งที่มี สถานะ,พฤติกรรม,เอกลักษณ์

2.1.1 OID

Object Identity หรือ Object Identifier (OID) เป็นสิ่งที่ใช้ในการระบุหรือใช้ในการอ้างอิงถึงออบเจกต์ ซึ่งในแต่ละออบเจกต์จะมีค่า OID ไม่ซ้ำกัน (Unique) เมื่อออบเจกต์ถูกสร้างขึ้นระบบจะทำการกำหนดค่า OID ให้ และไม่สามารถเปลี่ยนแปลงค่า OID ภายหลังได้ และเมื่อออบเจกต์นั้นถูกลบออกจากระบบ OID ของออบเจกต์นั้นก็ถูกลบไปด้วยไม่มีการนำค่า OID ของออบเจกต์ที่ถูกลบไปแล้วกลับมาใช้ใหม่

2.1.2 คุณสมบัติ (Property)

คือข้อมูลของออบเจกต์ เป็นได้ทั้งข้อมูลชนิดพื้นฐาน โดยทั่วไป ประกอบด้วย Integer , Float และ String เป็นต้น ส่วนข้อมูลที่ซับซ้อนจะประกอบด้วยการนำเอาข้อมูลพื้นฐานมารวมเป็นโครงสร้าง ได้แก่ Array หรืออาจจะเป็นข้อมูลที่เป็น ออบเจกต์ก็ได้

2.1.3 เมธอด (Method)

ออบเจกต์ที่เก็บจะถูกเรียกใช้งานผ่านทาง เมธอดเท่านั้น ตามคุณสมบัติเอนแคปซูเลชัน เมธอดจะถูกเรียกใช้เพื่อเรียกดูหรือเปลี่ยนแปลงค่าคุณสมบัติของออบเจกต์ เมธอดจะประกอบด้วย ชื่อเมธอด ,รูปแบบในการเรียกใช้ , และการทำงานของเมธอดนั้น

2.1.4 เมสเซจ (Message)

ออบเจกต์ที่เก็บอยู่ในระบบฐานข้อมูลเชิงวัตถุจะมีวิธีการใช้งาน โดยผ่านทาง เมธอด โดยเมธอดจะประกอบด้วย ชื่อ,รูปแบบในการใช้ และส่วนที่เป็นตัวเนื้อหาของเมธอด

ในการเรียกใช้เมธอดนั้นต้องทำการส่งเมสเซจไปยังออบเจกต์ กล่าวคือ เมสเซจที่ถูกส่งไปยังออบเจกต์จะเป็นตัวสั่งให้เมธอด ในออบเจกต์ทำงานตามรูปแบบการทำงานในส่วนเนื้อหาของเมธอด โดยเมสเซจที่ส่งจะต้องระบุออบเจกต์ที่จะรับเมสเซจนั้น ชื่อเมธอด และพารามิเตอร์ที่เกี่ยวข้องทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 คลาส (Class)

กลุ่มของออบเจกต์ที่มีโครงสร้างภายในเหมือนกัน ตอบสนองต่อเมสเสจเดียวกัน โดยที่มีรายละเอียดของคุณสมบัติและรายละเอียดของเมธอดที่เหมือนกันชนิดตัวแปรชนิดเดียวกัน ในบางกรณีอาจพบว่าเราสามารถให้ออบเจกต์ตัวหนึ่ง ๆ เป็นตัวแทนของคลาสมากกว่าหนึ่งคลาสมากกว่าหนึ่งคลาสได้ โดยการสืบทอดคุณสมบัติ ซึ่งจะอธิบายต่อไป แต่อย่างไรก็ตามออบเจกต์ตัวนั้นจะถูกสร้างขึ้นมาจากคลาสหลักเพียงหนึ่งคลาสเท่านั้น

2.1.6 เอนแคปซูเลชัน (Encapsulation)

เอนแคปซูเลชันเป็นการซ่อนรายละเอียด คุณสมบัติ (Property) และเมธอด ของคลาสไว้ ข้อมูลภายในออบเจกต์จะถูกปิดกั้นไว้ไม่ให้มองเห็นจากภายนอกคลาส โดยเมื่อสิ่งต่าง ๆ เมื่ออยู่ภายนอกคลาสดึงข้อมูลกับคลาสได้ต้องติดต่อผ่านทางเมธอดที่คลาสนั้นเตรียมไว้ให้เท่านั้น ทำให้ข้อมูลมีความปลอดภัยมากขึ้น การซ่อนรายละเอียดของคลาสมีหลายระดับ ในบ้างรายละเอียดอาจเปิดเผยให้ภายนอกสามารถมองเห็นและเรียกใช้งาน ได้โดยตรง แต่ในบางรายละเอียดต้องการปกปิดไม่ให้ภายนอกมองเห็น ขึ้นกับความจำเป็นในการใช้งาน การปกปิดดังกล่าวสามารถแบ่งได้ 3 ระดับ คือ

- 2.1.6.1 Private เมื่อถูกกำหนดให้เป็นแบบ Private ข้อมูลภายในของคลาสจะไม่ถูกเปิดเผยแก่ภายนอก และไม่สามารถเข้าถึงได้โดยตรงจากภายนอก แต่สามารถเข้าถึงได้จากเมธอดที่อยู่ภายในคลาสนั้นเท่านั้น การเรียกใช้จากภายนอกจริงต้องเรียกผ่านเมธอดเท่านั้น
- 2.1.6.2 Protect เป็นการประกาศข้อมูลแบบให้คลาสลูกเรียกใช้ได้ มีผลให้เฉพาะเมธอดในคลาสนั้นและคลาสที่สืบทอด(คลาสลูก) เท่านั้นที่สามารถอ้างอิงถึงข้อมูลภายในคลาสได้
- 2.1.6.3 Public ข้อมูลภายในคลาสจะถูกเปิดเผยและสามารถถูกเข้าถึงได้โดยตรงจากภายนอก นั่นคือไม่มีการปิดกั้นข้อมูลใด ๆ ทั้งสิ้น

2.1.7 การสืบทอดคุณสมบัติ (Inheritance)

ในระบบฐานข้อมูลเชิงวัตถุ มีแนวคิดการสืบทอดคุณสมบัติ โดย ออบเจกต์จะสืบทอดคุณสมบัติ (Property) และเมธอด มาจากคลาสที่มีอยู่แล้ว (Super Class) โดยสามารถเพิ่มคุณสมบัติพิเศษบางอย่างให้กับคลาสเดิมเพื่อให้เกิดคลาสใหม่ที่พิเศษกว่าคลาสเดิม จากคุณสมบัตินี้ทำให้ได้ประโยชน์ทางการนำกลับมาใช้ใหม่ และเพิ่มคุณสมบัติเข้าไป โดยที่ไม่ต้องเขียนคลาสขึ้นมาใหม่ทั้งหมด แบ่งออกเป็น 2 แบบคือ

Single Inheritance เป็นการสืบทอดมาจากคลาสแม่เพียงคลาสเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Multiple Inheritance เป็นการสืบทอดมาจากคลาสแม่หลาย ๆ คลาส

2.1.8 พอลิมอร์ฟิซึม (Polymorphism)

การที่เมธอดสามารถใช้ชื่อเดียวกันในหลาย ๆ คลาส โดยยอมให้มีการส่งเมสเสจเดียวกันไปยังออบเจกต์หลายตัว และออบเจกต์แต่ละตัวมีการตอบสนองต่อเมสเสจนั้นแตกต่างกัน ทำให้เราไม่ต้องใช้เมสเสจที่แตกต่างกันตามแต่ละออบเจกต์ ช่วยให้ชนิดของเมสเสจที่ส่งเหลือเพียงชนิดเดียว แบ่งออกเป็น 2 ลักษณะใหญ่คือ

2.1.8.1 แบบ Overloading คือ การที่เมธอดภายในคลาสเดียวกันมีชื่อเหมือนกัน แต่มีรูปแบบการใช้งานที่แตกต่างกัน

2.1.8.2 แบบ Overriding คือ คลาสลูกมีเมธอดที่มีชื่อเหมือนกับเมธอดที่สืบทอดมาจากคลาสแม่ โดยได้ทำการแก้ไขการทำงานภายในเมธอดให้มีความเจาะจงมากขึ้น เมธอดนี้จะถูกเรียกใช้แทนเมธอดที่สืบทอดมาจากคลาสแม่

2.1.9 ความสัมพันธ์ระหว่างออบเจกต์ (Association)

ความสัมพันธ์ มี 2 ลักษณะ คือ

2.1.9.1 ความสัมพันธ์ระหว่างคลาส และซับคลาส (Class-Subclass Relationship) เรียกความสัมพันธ์แบบนี้ว่า “is-a” เป็นคุณสมบัติการสืบทอดของคลาส

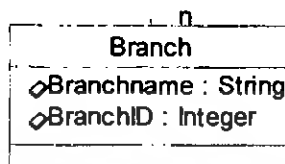
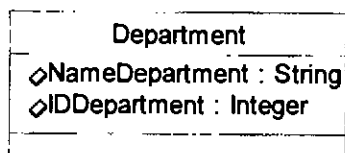
จาก Superclass

2.1.9.2 ความสัมพันธ์ระหว่างคลาส

ความสัมพันธ์แบบนี้จะเกิดขึ้นเมื่อชนิดของคุณสมบัติมีการอ้างอิงไปยังออบเจกต์อื่น

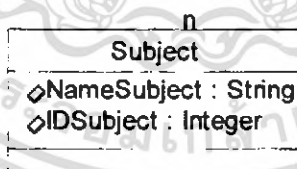
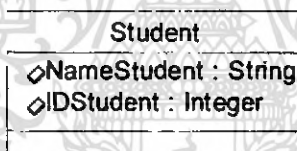
โดยมีความสัมพันธ์ 3 แบบ คือ ความสัมพันธ์แบบ one-to-one , one-to-many , many-to-many

- one-to-one ถ้าออบเจกต์มีความสัมพันธ์กันแบบนี้ หมายความว่า ออบเจกต์ตัวหนึ่งสามารถอ้างอิงไปที่ออบเจกต์อื่นได้เพียงตัวเดียวเท่านั้น ในทางกลับกันออบเจกต์ตัวนั้นก็สามารอ้างอิงถึงออบเจกต์ได้เพียงหนึ่งตัวเท่านั้น
- one-to-many และ many-to-one ถ้าออบเจกต์มีความสัมพันธ์แบบนี้หมายความว่า ออบเจกต์ตัวหนึ่ง สามารถอ้างอิงไปที่ออบเจกต์อื่นได้หลายตัว แต่ในทางกลับกันออบเจกต์ตัวนั้นถูกอ้างอิงได้จากออบเจกต์เดียว จากตัวอย่าง หนึ่งคนสามารถมีได้หลายสาขา แต่สาขาหนึ่งๆ อยู่ในคณะได้เพียงหนึ่งคณะ



รูปที่ 2-1 ความสัมพันธ์แบบ *one-to-many* และ *many-to-one*

- many-to-many ถ้าออบเจกต์มีความสัมพันธ์แบบนี้หมายความว่า ออบเจกต์ตัวหนึ่งสามารถอ้างอิงไปที่ออบเจกต์อื่นได้หลายตัว แต่ในทางกลับกันออบเจกต์ตัวนั้นถูกอ้างอิงได้จากออบเจกต์หลายตัวเช่นกัน ตัวอย่างเช่น นักเรียนคนหนึ่งสามารถลงทะเบียนเรียนได้หลายวิชา ในขณะที่วิชาหนึ่งหนึ่งก็สามารถถูกลงทะเบียนได้จากนักศึกษาหลายคน



รูปที่ 2-2 ความสัมพันธ์แบบ *many-to-many*

2.2 ฐานข้อมูลเชิงวัตถุ (Object Oriented Database)

ในอดีตที่ผ่านมา ได้มีการค้นคว้า และพัฒนาระบบฐานข้อมูลขึ้นหลายรูปแบบ หลายแนวคิด ระบบฐานข้อมูลที่ได้รับความนิยม และแพร่หลายสูงสุดในการใช้งานคือ ระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database System) ในขณะที่ปัจจุบันแนวความคิด และมุมมองข้อมูลในรูปแบบเชิงวัตถุได้รับความนิยมเพิ่มมากขึ้น ทำให้การพัฒนาโปรแกรมส่วนใหญ่ในยุคปัจจุบัน ล้วน

แต่ยึดหลักการเขียน โปรแกรมเชิงวัตถุมากขึ้น จนแนวคิดดังกล่าวได้เริ่มเข้ามามีบทบาทสำคัญในการพัฒนาระบบฐานข้อมูลเชิงวัตถุ (Object-Oriented Database)

ในที่นี้ได้เปรียบเทียบถึงความแตกต่างระหว่างเทคโนโลยีแบบรีเลชันกับเทคโนโลยีแบบออบเจกต์ ในส่วนที่ได้พัฒนาเพิ่มเข้ามาและส่วนของประสิทธิภาพของเทคโนโลยีแบบออบเจกต์ แนวความคิดของฐานข้อมูลแบบเชิงสัมพันธ์ กับ ฐานข้อมูลแบบเชิงวัตถุ

2.2.1 แนวความคิดของฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

ถ้ากล่าวโดยสรุปแล้ว โครงสร้างข้อมูลของระบบฐานข้อมูลเชิงสัมพันธ์จะจัดเก็บข้อมูลที่สามารถแสดงได้ในรูปแบบของตารางโดยประกอบด้วยแถวและคอลัมน์ โดยแถว ของตาราง หมายถึงทูเปิ้ล (Tuple) และคอลัมน์ หมายถึง แอททริบิวต์ หรือฟิลด์ โดยแต่ละคอลัมน์จะมีชนิดของข้อมูล สำหรับชนิดข้อมูลในแต่ละคอลัมน์จะต้องเหมือนกัน และมีเพียงชนิดเดียวเท่านั้น โดยที่ข้อมูลที่ถูกจัดเก็บในแต่ละฟิลด์ จะมีค่าได้เพียงค่าเดียวจากคุณสมบัติ อะตอมิก (Atomic) สำหรับความสัมพันธ์ (Relationship) ระหว่างข้อมูลแต่ละตาราง จะทำโดยใช้การจัดเก็บคีย์หลัก (Primary Key) ของอีกตารางหนึ่ง เป็น Foreign Key ในอีกตารางหนึ่ง

ระบบฐานข้อมูลเชิงสัมพันธ์ใช้ภาษาเอสคิวแอล (Structure Query Language : SQL) ซึ่ง SQL จะรวมคำสั่งที่ใช้ในการกำหนดชนิดข้อมูล , คำสั่งที่ใช้ในการเปลี่ยนแปลงข้อมูล , คำสั่งคิวรี โดยสามารถคิวรี ได้ทุกระดับ ตั้งแต่คิวรีตารางข้อมูลง่ายๆจากเพียงตารางเดียว จนถึงแบบที่มีความซับซ้อนจากหลายตาราง ที่ต้องอาศัยการใช้การรวมกันของหลายตาราง (Join) โดยข้อเสียของฐานข้อมูลเชิงสัมพันธ์ คือ ไม่สามารถใช้ชนิดข้อมูลแบบพิเศษ เช่น อาร์เรย์ (Array)

2.2.2 แนวความคิดของฐานข้อมูลเชิงวัตถุ (Object Oriented Database)

ฐานข้อมูลเชิงวัตถุ เกิดขึ้นจากการรวมเอาพื้นฐานแนวความคิดเชิงวัตถุ กับหลักการด้านภาษาสำหรับการเขียน โปรแกรมเชิงวัตถุ และความสามารถด้านฐานข้อมูลเข้าด้วยกัน ผลลัพธ์ที่ได้คือทำให้เกิดความเข้ากันได้ดีระหว่างโครงสร้างข้อมูลภายในฐานข้อมูล กับโครงสร้างข้อมูลที่ใช้ในการพัฒนาโปรแกรม ซึ่งอำนวยความสะดวกในการใช้ภาษาสำหรับการเขียน โปรแกรมเชิงวัตถุ ต่างๆ

2.2.2.1 โครงสร้างข้อมูล

จากแนวคิดการออกแบบระบบเชิงวัตถุนำไปสู่การพัฒนาฐานข้อมูลเชิงวัตถุ โดยการใช้งานข้อมูลในฐานข้อมูลเชิงวัตถุจะอยู่ในรูปแบบของคลาส ซึ่งเป็นแนวความคิดตามทฤษฎีของ Object-Oriented โดยมีลักษณะการเก็บข้อมูลและการเรียกใช้ข้อมูลแตกต่างจากระบบฐานข้อมูลเชิงสัมพันธ์ ก็จะมองข้อมูลที่ถูกจัดเก็บในรูปแบบของออบเจกต์ ซึ่งออบเจกต์จะประกอบด้วยเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติ (Property) และเมธอด ซึ่งจะมีการกำหนดค่าที่ใช้ในการอ้างอิงถึงออบเจกต์ (Object Identifier :OID) เอาไว้ นอกจากนี้ยังสนับสนุนการทำงานแบบ Encapsulation และ Inheritance ด้วย

2.2.2.2 ภาษาที่ใช้ในการจัดการฐานข้อมูล

2.2.2.2.1 ระบบฐานข้อมูลเชิงวัตถุจะใช้ภาษาสำหรับการเขียน โปรแกรมเชิงวัตถุ ทั้งในการจัดการ ฐานข้อมูล และการพัฒนาโปรแกรม เช่น C++, Java เนื่องจาก แอปพลิเคชัน และออบเจกต์ที่เก็บข้อมูลในระบบฐานข้อมูลชนิดนี้มีความสัมพันธ์กัน โดยตรง ดังนั้นการทำ กำหนดข้อมูล , การจัดการกับข้อมูล และการ Query จึงสามารถ ทำได้ด้วยภาษาสำหรับการ โปรแกรมเชิงวัตถุ คือ กระทำผ่าน ออบเจกต์ซึ่งเมื่อมีการทำ การเปลี่ยนแปลงค่าใด ๆ ในออบเจกต์แล้วจะมีผลต่อข้อมูลในฐานข้อมูล มีข้อดีคือ

- ทำให้ไม่ต้องใช้ คำสั่ง SQL ซึ่งมีความยุ่งยากซับซ้อนกว่า เนื่องจาก สามารถทำการเปลี่ยนแปลง , แก้ไข , ค้นหาข้อมูลได้โดยผ่านทาง ออบเจกต์
- ช่อนความยุ่งยากในการเขียน โปรแกรมแอปพลิเคชัน เนื่องจากการใช้ คุณสมบัติเอนแคปซูเลชันของออบเจกต์ ผู้ใช้แค่ใช้เมธอดที่ ผู้ออกแบบระบบฐานข้อมูลได้เตรียมไว้ให้โดยไม่ต้องทราบว่าจะข้างใน ออบเจกต์เป็นยังไง

2.2.2.2.2 ตามมาตรฐาน ODMG-93 ได้มีการกำหนดภาษา ขึ้นมาอีกภาษาหนึ่งคือ OQL (Object Query Language) ซึ่งใช้ในการเข้าถึงข้อมูลแบบวัตถุซึ่งจะมีความ แตกต่างกันไปจาก SQL ตามมาตรฐาน SQL 2 เพราะในมาตรฐาน SQL 2 ไม่มี ความสามารถในการเข้าถึงข้อมูลแบบวัตถุ

2.2.2.3 กลไกที่ใช้ในการเข้าถึงข้อมูล

การสร้างและแก้ไขข้อมูลในระบบฐานข้อมูลเชิงวัตถุจะใช้การเข้าถึงโดยตรงจากภาษา สำหรับการเขียน โปรแกรมเชิงวัตถุ ในระบบฐานข้อมูลเชิงวัตถุนี้ระบบจะกำหนด OID ให้ ออบเจกต์แต่ละตัวอัตโนมัติเมื่อออบเจกต์ถูกสร้าง และค่า OID ของแต่ละออบเจกต์จะไม่ซ้ำกัน ไป ตลอดการใช้งาน นอกจากนี้ในตัวออบเจกต์แต่ละออบเจกต์ยังสามารถจัดเก็บ OID ของออบเจกต์ ตัวอื่น เพื่อใช้ในการอ้างอิงถึงออบเจกต์ตัวอื่นได้ ซึ่งจะเป็นประโยชน์ในการสร้างความสัมพันธ์ ระหว่างออบเจกต์

ตารางเปรียบเทียบฐานข้อมูลเชิงวัตถุและฐานข้อมูลเชิงสัมพันธ์

ตารางที่ 2-1 ตารางเปรียบเทียบแนวคิดเชิงวัตถุและแนวคิดเชิงสัมพันธ์

แนวคิดเชิงวัตถุ	แนวคิดเชิงสัมพันธ์
Class	Entity Set
Object	Entity
Instance Variable	Attribute
OID	Primary
Method	-

ตารางที่ 2-2 ตารางเปรียบเทียบฐานข้อมูลเชิงวัตถุและฐานข้อมูลเชิงสัมพันธ์

ฐานข้อมูลเชิงวัตถุ	ฐานข้อมูลเชิงสัมพันธ์
1. เก็บข้อมูลในรูปแบบของออบเจกต์	1. เก็บข้อมูลในรูปตาราง
2. เข้าถึงข้อมูลโดยผ่านเมธอดเท่านั้น	2. เข้าถึงข้อมูลได้โดยตรง
3. ภาษาทางด้านออบเจกต์ และ ภาษา OQL	3. ใช้คำสั่ง SQL ในการเข้าถึงข้อมูล
4. รองรับความสามารถของออบเจกต์	

2.2.3 ข้อดีของฐานข้อมูลเชิงวัตถุ

2.2.3.1 มีการซ่อน โครงสร้างของข้อมูลเอาไว้ทำให้ไม่สามารถเรียกใช้งานข้อมูลได้ โดย ตรงจากภายนอก โดยการเรียกใช้งานจะมีเมธอดเรียกใช้ผ่านทางออบเจกต์ที่กำหนดไว้ เป็นแนวคิดของการเอนแคปซูเลชัน ทำให้ข้อมูลมีความปลอดภัยมากขึ้น โดยภายนอกไม่สามารถมองเห็นและเรียกใช้งานได้โดยตรง

2.2.3.2 ซ่อนความยุ่งยากในการเขียนโปรแกรมแอฟพลิเคชัน เนื่องจากการใช้คุณสมบัติเอนแคปซูเลชันของออบเจกต์ ผู้ใช้แค่ใช้เมธอดที่ผู้ออกแบบระบบฐานข้อมูลได้เตรียมไว้ให้ โดยไม่ต้องทราบว่าจะข้างในออบเจกต์เป็นยังไง

2.2.3.3 เนื่องจากการทำงานของฐานข้อมูลเชิงวัตถุไม่จำเป็นต้องรวม (join) ตาราง ซึ่งในการค้นหาข้อมูลที่มีความซับซ้อน การรวม (join) ตาราง จะมีความยุ่งยากมากขึ้น และใช้เวลาในการรวม (join) มากขึ้นด้วย โดยฐานข้อมูลเชิงวัตถุจะค้นหาข้อมูลโดยใช้ตัวชี้ที่อยู่ภายในออบเจกต์เป็นตัวทำให้

2.2.3.4 ฐานข้อมูลเชิงวัตถุสามารถนำกลับมาใช้ใหม่ได้ สามารถเพิ่มเติมคุณสมบัติให้กับคลาสที่มีอยู่เดิม มีการนำคลาสที่เคยสร้างไว้แล้วมาใช้ซ้ำช่วยให้ไม่ต้องออกแบบส่วนที่ทำหน้าที่เหมือนกันหลาย ๆ รอบ และไม่ต้องทดสอบใหม่อีกครั้ง

2.2.3.5 มีความสามารถในการสร้าง Library ของคลาสของทั้งแอปพลิเคชัน โปรแกรม และฐานข้อมูลช่วยให้การพัฒนาระบบเร็วขึ้น เพราะไม่ต้องเสียเวลาในการสร้างคลาสเดิมที่เคยทำมาก่อน

2.2.3.6 ฐานข้อมูลเชิงวัตถุมีการควบคุมการค้นหาที่ดีกว่า เพราะตัวชี้ที่ชี้ไปยังวัตถุที่มีความสัมพันธ์กันนั้นถูกเก็บในตัววัตถุเอง

2.2.3.7 ฐานข้อมูลเชิงวัตถุมีการตรวจสอบความถูกต้องในการอ้างอิง เนื่องจากการอ้างอิงถึงวัตถุอื่นนั้นไม่ใช่การอ้างอิงถึงเท่านั้นแต่เป็นการบอกที่อยู่ของวัตถุนั้นโดยตรง นอกจากนี้ฐานข้อมูลเชิงวัตถุยังแสดงถึงความสัมพันธ์ของวัตถุได้มากกว่าระบบอื่นด้วยความสามารถในการสร้างชนิดข้อมูลและรวมเอาวิธีจัดการกับข้อมูลเข้าไว้ในวัตถุ

2.2.3.8 ในระบบเชิงสัมพันธ์การกระจายกันเก็บของวัตถุ เป็นปัญหาอย่างหนึ่งของผู้ออกแบบ โดยผู้ออกแบบต้องแน่ใจว่าตารางที่รวมกันบ่อย ๆ จะถูกเก็บในที่เดียวกันและอยู่ในพื้นที่ที่ใกล้กันเพื่อลดปริมาณการเข้าถึงดิสก์ แต่ในฐานข้อมูลเชิงวัตถุวัตถุที่เป็นกลุ่มเดียวกันจะถูกเก็บไว้ด้วยกันเหมือนกับการประกอบกันของวัตถุ โดยจะนำโครงสร้างของคลาสมาเป็นหลักในการพิจารณาการแยกวัตถุออกเป็นกลุ่มในการเก็บ

2.2.3.9 โครงสร้างข้อมูลในโปรแกรมประยุกต์และฐานข้อมูลเชิงวัตถุจะมีโครงสร้างของข้อมูลเหมือนกัน คือใช้ตัวแปรในรูปแบบเดียวกันทำให้ การเข้าถึงฐานข้อมูลใช้ภาษาจากโปรแกรมประยุกต์ได้เลย

บทที่ 3

การจัดเก็บรูปภาพ

3.1 วิธีที่ 1 การจัดเก็บและสืบค้นภาพทิวทัศน์ทั่วไป

3.1.1 การแปลง Color Space และ ฮิสโตแกรม

3.1.1.1 Color Space

Color Space คือการแสดงผลของสีในมิติ (dimension) ต่าง ๆ โดยหนึ่งพิกเซลประกอบด้วย 3 มิติในโครงงานนี้มี Color Space ที่เกี่ยวข้องอยู่ 2 Color Space คือ RGB color space และ 166 สี HSV color space โดยมีลักษณะดังนี้

3.1.1.1.1 RGB color space ภาพสีที่อยู่ในรูปแบบของ RGB จะเป็นรูปแบบพื้นฐานโดย 1 พิกเซลของภาพประกอบด้วยค่าสี 3 ค่า คือ สีแดง, สีเขียว และสีน้ำเงิน ซึ่งจะเหมาะสมกับการแสดงผลบนหน้าจอคอมพิวเตอร์ แต่เนื่องจากการเก็บสีแยกเป็นคณละค่าทำให้เมื่อคำนวณแยกทีละค่าของสีแล้วสีของภาพที่ได้จะเกิดการเพี้ยนได้

3.1.1.1.2 HSV color space การกำหนดภาพสีในรูปแบบของ HSV (hue, saturation, value) โดยความสัมพันธ์จะเหมือนกับหลักการของจิตรกร คือการใช้ สี เงาม และความเข้มของสี ซึ่งเป็นการกำหนดตามลักษณะการมองเห็น ซึ่งเหมาะสมกับการแยกแยะวัตถุตามสีที่ปรากฏในการทำงานของโปรแกรมจะใช้ HSV color space แต่ในการแสดงผลหน้าจอคอมพิวเตอร์ ปกติจะอยู่ใน RGB color space ดังนั้นก่อนที่จะนำภาพมาใช้งานในโปรแกรมจึงต้องทำการแปลง color space ก่อน

3.1.1.2 การแปลง Color space

ก่อนที่จะนำภาพที่อยู่ใน RGB color space มาใช้งานในโปรแกรม ต้องทำการแปลงจาก RGB color space ให้มาเป็น 166 สี HSV color space

นิยามในการแปลงสี

ให้ $\hat{v}_c = (r, g, b)$ เป็น color point บน RGB color space
 $\hat{w}_c = (h, s, v)$ เป็น color point บน HSV color space
ซึ่ง $\hat{w}_c = T_c * \hat{v}_c$ สำหรับ $r, g, b \in [0...1]$

เมื่อผ่าน T_c จะได้ $h, s, v \in [0...1]$ โดยมีสมการการแปลงสี ดังสมการที่ 3.1-3.3

$$v = \max(r, g, b) \quad (3.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$s = \frac{v - \min(r, g, b)}{v} \quad (3.2)$$

$$6h = \begin{cases} 5 + b' & \text{ถ้า } r = \max(r, g, b) \text{ และ } g = \min(r, g, b) \\ 1 - g' & \text{ถ้า } r = \max(r, g, b) \text{ และ } g \neq \min(r, g, b) \\ 1 + r' & \text{ถ้า } g = \max(r, g, b) \text{ และ } b = \min(r, g, b) \\ 3 - b & \text{ถ้า } g = \max(r, g, b) \text{ และ } b \neq \min(r, g, b) \\ 3 + g' & \text{ถ้า } b = \max(r, g, b) \text{ และ } r = \min(r, g, b) \\ 5 - r' & \text{ถ้า ไม่ตรงกับกรณีที่กล่าวมาข้างต้น} \end{cases} \quad (3.3)$$

โดย $r' = \frac{v-r}{v - \min(r, g, b)}$, $g' = \frac{v-g}{v - \min(r, g, b)}$, $b' = \frac{v-b}{v - \min(r, g, b)}$

จากสมการแปลงสีที่กล่าวมาข้างต้น จะได้ HSV color space โดยระดับสีที่ได้ 8 ระดับสี

3.1.2 วิธีการจัดเก็บและค้นหาภาพค้นหาภาพ

3.1.2.1 วิธีที่ 1 Color Histogram

3.1.2.1.1 ฮิสโตแกรมของสี

ฮิสโตแกรมของสีประกอบไปด้วยระดับต่างๆ ของสี ซึ่งระดับต่าง ๆ นี้จะเป็นจำนวนของสีในแต่ละระดับที่ปรากฏในภาพ

การสร้างฮิสโตแกรมของสีทำได้โดยเก็บสีที่ปรากฏในภาพที่ผ่านการแปลงสีและลดระดับสีแล้ว จากนั้นทำโดยใช้วิธีการนับจำนวนพิกเซลของแต่ละสีที่ต่างกันที่ปรากฏในภาพและเก็บไว้โดยสร้างเป็นฮิสโตแกรมของแต่ละภาพขึ้นมา ในโครงการนี้ใช้เพียงฮิสโตแกรมของระดับสีเพียงอย่างเดียว

3.1.2.1.2 ฮิสโตแกรมอินเตอร์เซกชัน

ในการค้นหาภาพ จะใช้วิธีดูความคล้ายของคู่ฮิสโตแกรมของภาพเป้าหมายและฮิสโตแกรมของภาพโมเดล วิธีนี้เรียกว่าฮิสโตแกรมอินเตอร์เซกชัน วิธีนี้จะเป็นการวัดว่ามีจำนวนพิกเซลของภาพ โมเดลเท่าไรที่พบในภาพเป้าหมาย

การทำฮิสโตแกรมอินเตอร์เซกชัน

วิธีฮิสโตแกรมอินเตอร์เซกชันเป็นวิธีที่ใช้ในการวัดการซ้อนทับกันระหว่างคู่ฮิสโตแกรมที่กำลังพิจารณา

นิยาม กำหนดให้คู่ของฮิสโตแกรม I และ M

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย I ที่ เป็นฮิสโตแกรมของภาพเป้าหมาย และ
 M เป็นฮิสโตแกรมของภาพโมเดลที่เรานำมาพิจารณา
 เราจะสามารถหาอินเตอร์เซกชันระหว่าง 2 ฮิสโตแกรม นี้ได้จากสมการที่ 3.4

$$\sum_{j=1}^n \min(I_j, M_j) \quad (3.4)$$

โดยที่ n คือ จำนวนระดับทั้งหมดของฮิสโตแกรมที่สนใจ
 I_j คือ จำนวนพิกเซลของภาพเป้าหมายในระดับสีที่ j
 M_j คือ จำนวนพิกเซลของภาพโมเดลที่ระดับที่ j

ผลที่ได้จากสมการนี้เป็นจำนวนรวมพิกเซลของฮิสโตแกรมทั้ง 2 ภาพที่มีค่า
 เดียวกัน จากนั้นนำจำนวนรวมพิกเซลของภาพ โมเดลที่พิจารณามาหารกับค่าอินเตอร์เซก
 ชันที่ได้จากสมการในตอนต้น จะหาค่าความเหมือนได้ ดังสมการที่ 3.5

$$H(I, M) = \frac{\sum_{j=1}^n I_j M_j}{\sum_{j=1}^n M_j} \quad (3.5)$$

ค่าความเหมือนที่ได้จากสมการจะมีค่าอยู่ในช่วง 0 - 1 โดยที่ถ้าค่าความ
 เหมือนใกล้เคียงกับ 1 จะเป็นตัวชี้ว่าภาพใดมีความเหมือนกับภาพโมเดลมากที่สุด
ขั้นตอนการทำฮิสโตแกรมของภาพ

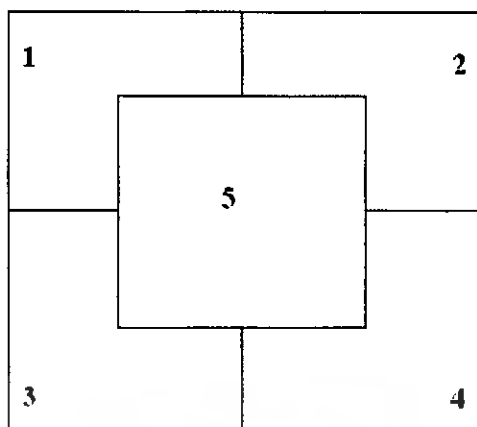
ในส่วนการเก็บภาพลงในฐานข้อมูล

- ทำฮิสโตแกรมของสีทั้งภาพ
- เก็บภาพและฮิสโตแกรมลงในฐานข้อมูล

ในส่วนการทำการค้นหาภาพในฐานข้อมูล

- จะนำภาพ Model ที่ต้องการหา มาทำฮิสโตแกรมของสี
- นำฮิสโตแกรมของภาพโมเดลที่ต้องการค้นหา มาฮิสโตแกรมอินเตอร์เซกชัน
กับฮิสโตแกรมของภาพที่เก็บอยู่ภายในฐานข้อมูล

จากการใช้สีของภาพพบว่าขาดความสามารถในการแยกแยะรูปภาพที่มีจำนวนสี
 ของภาพใกล้เคียงกัน แต่มีลักษณะการเกาะกลุ่มสีในภาพต่างกัน แนวทางในการแก้ปัญหา
 โดยได้ทำการบ่งขอบเขตของภาพที่สนใจออกเป็นส่วนเล็ก ๆ ซึ่งได้แบ่งขอบเขตของภาพ
 ออกเป็น 5 ส่วน จะแสดงไว้ในรูปที่ และจะนำแต่ละส่วนที่ได้จากการแบ่ง มาฮิสโตแกรม
 อินเตอร์เซกชัน โดยส่วนที่ 1,2,3 และ 4 สามารถหมุนแล้วนำมาทำฮิสโตแกรมอินเตอร์เซก
 ชัน



รูปที่ 3-1 แสดงแนวคิดในการแบ่งขอบเขตของภาพออกเป็น 5 ส่วน

ขั้นตอนการทำฮิสโตแกรมของภาพ

ในส่วนการเก็บภาพลงในฐานข้อมูล

- ทำการแบ่งขอบเขตของภาพที่สนใจออกเป็น 5 ส่วนดังรูปที่
- ทำฮิสโตแกรมของสีทั้ง 5 ส่วน
- เก็บ ภาพและค่าฮิสโตแกรมลงในฐานข้อมูล

ในส่วนการทำการค้นหภาพในฐานข้อมูล

- จะนำภาพ โมเดลที่ต้องการหาทำการแบ่งขอบเขตของภาพออกเป็น 5 ส่วน และทำฮิสโตแกรมของสีทั้ง 5 ส่วนของภาพ
- นำฮิสโตแกรมของภาพ โมเดลที่ต้องการค้นหาทำฮิสโตแกรมอินเตอร์เซกชันกับฮิสโตแกรมของภาพที่เก็บอยู่ภายในฐานข้อมูลซึ่งมีรูปแบบการจับคู่อินเตอร์เซกชันในแต่ละส่วนดังตาราง เพื่อหาค่าความเหมือนในแต่ละส่วนของแต่ละรูปแบบ ดังตารางที่ 3-1

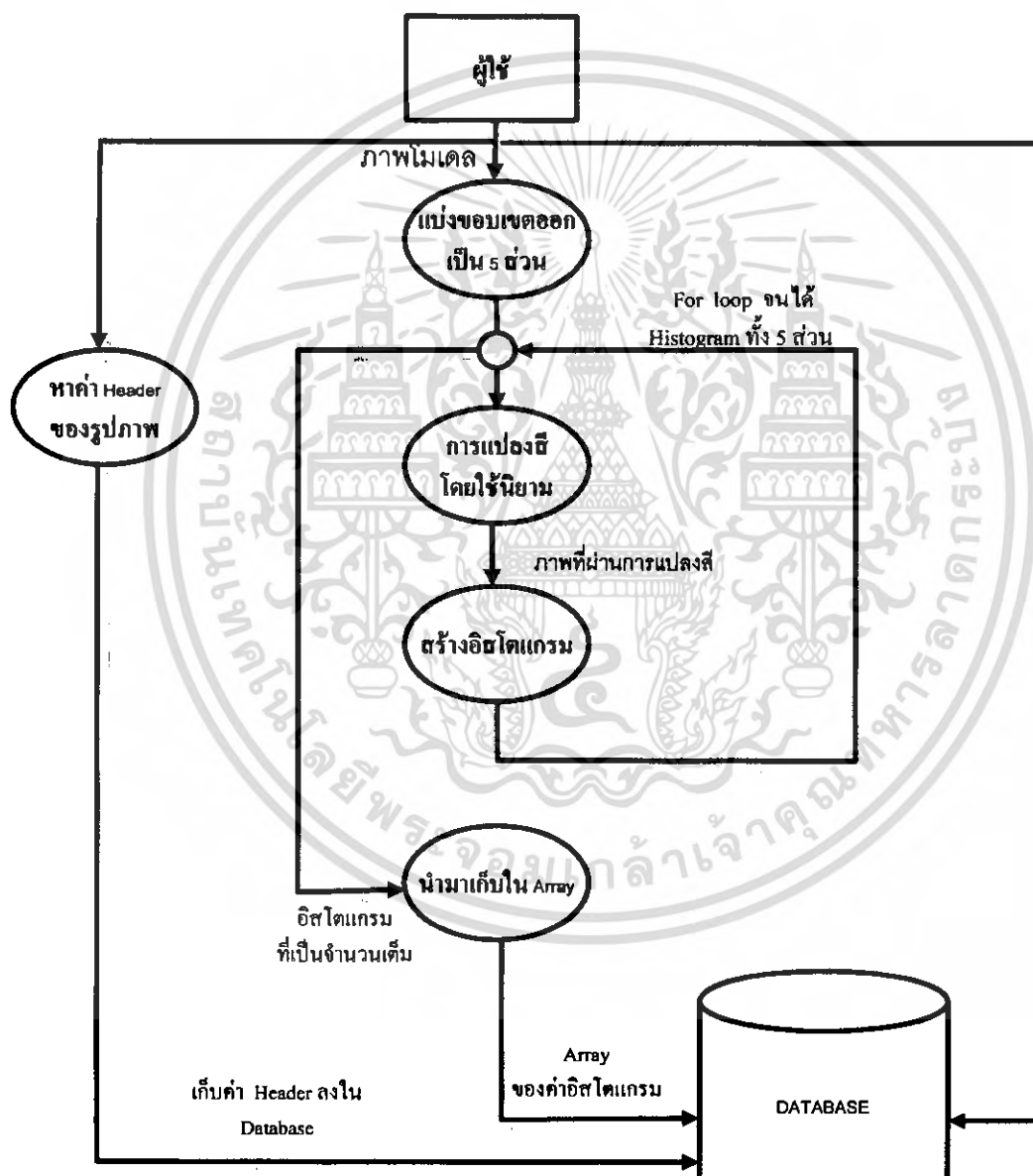
ตารางที่ 3-1 ตารางแสดงการจับคู่อินเตอร์เซกชันของภาพในแต่ละส่วน

รูปแบบ	M	T	M	T	M	T	M	T	M	T
1	1	1	2	2	3	4	4	3	5	5
2	1	3	2	1	3	2	4	4	5	5
3	1	4	2	3	3	1	4	2	5	5
4	1	2	2	4	3	3	4	1	5	5

หมายเหตุ M = ภาพโมเดลที่ต้องการค้นหา , T = ภาพเป้าหมาย

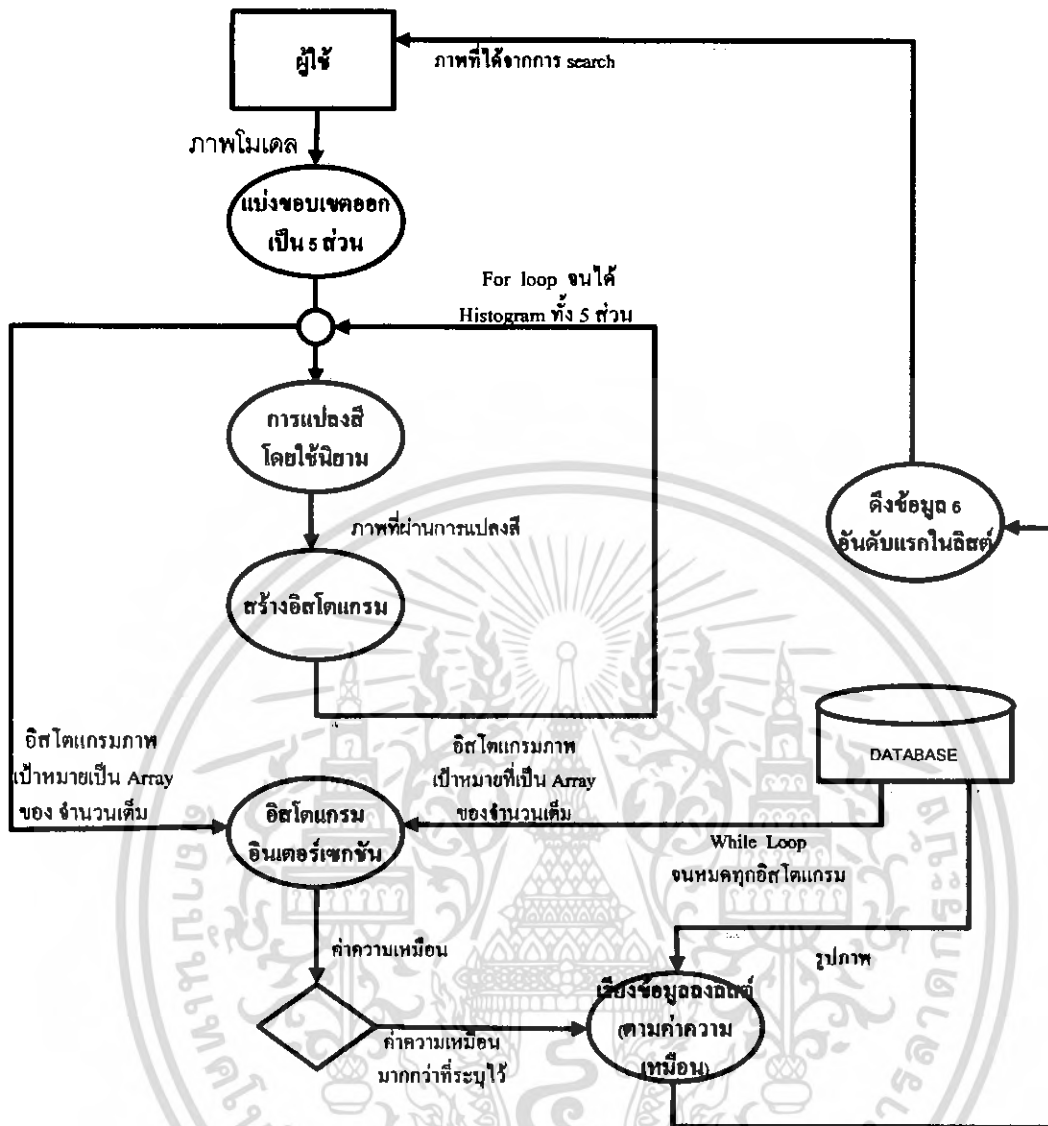
- ในแต่ละรูปแบบจะนำค่าความเหมือนใน 5 ส่วนมารวมกันและหารด้วย 5
- หาค่าความเหมือนสูงสุดจากทั้ง 4 รูปแบบ เก็บค่าที่ได้และรูปลงในรายการที่จะแสดงเป็นผลลัพธ์

แผนผังในการทำงาน



รูปที่ 3-2 แสดงขั้นตอนการทำงานของ การเก็บภาพแล้วฮิสโตแกรมของภาพนิ่งจากฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-3 แสดงขั้นตอนการทำงานของการค้นหาและเรียกภาพนิ่งจากฐานข้อมูล

72679

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 วิธีที่ 2 การจัดเก็บและสืบค้นภาพใบหน้าของคน

3.2.1 การจัดเก็บข้อมูลรูปภาพลงในฐานข้อมูล มีขั้นตอนดังต่อไปนี้

3.2.1.1. เมื่อรับรูปภาพเข้ามาจะทำการอ่านค่าแต่ละพิกเซลในภาพระดับสีเทา

การแปลงภาพสีไปเป็นภาพระดับสีเทา (Gray Scale Image)

ภาพในระดับสีเทานั้นแต่ละพิกเซลจะมีค่าอยู่ในช่วงตั้งแต่ 0 -255 ซึ่งการแปลงนั้นจะเป็นไปตามสมการที่ 3.6

$$\text{gray value} = 0.299*r + 0.587*g + 0.114*b \quad (3.6)$$

ซึ่ง r คือค่าสีแดงในแต่ละพิกเซลของภาพสี

g คือค่าสีเขียวในแต่ละพิกเซลของภาพสี

b คือค่าสีน้ำเงินในแต่ละพิกเซลของภาพสี

เมื่อทำการแปลงแล้วจะได้ภาพออกมาเป็นลักษณะดังนี้



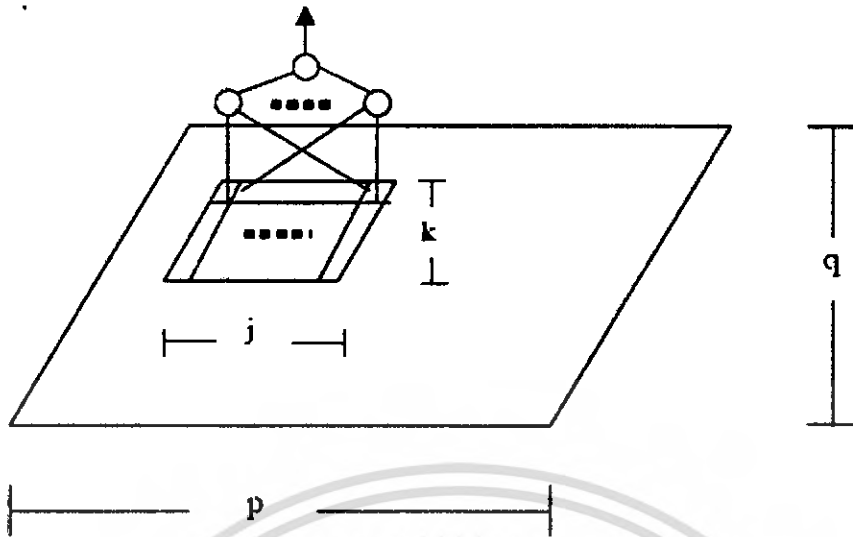
รูปที่ 3-4 รูปภาพดั้งเดิม



รูปที่ 3-5 รูปภาพระดับสีเทา

3.2.1.2. ขั้นตอนการนำค่าจากรูปภาพมาแปลงเป็นค่าที่จะนำไปใช้ในการเปรียบเทียบรูปภาพ ซึ่งจะเป็นค่าที่เก็บในฐานข้อมูลด้วย มีหลักการดังต่อไปนี้

ขนาดของภาพอินพุตมีขนาดเป็น $p*q$ พิกเซล และอาร์เรย์สำหรับเก็บค่าตัวแทนรูปภาพมีขนาด $m*n$ พิกเซล ดังนั้นกลุ่มของพิกเซลของรูปภาพอินพุตที่จะรับเพื่อเก็บลงใน 1 ช่องอาร์เรย์มีขนาดเป็น $p/m * q/n$ พิกเซล (ซึ่งต่อจากนี้จะกำหนดให้ $j = p/m$ และ $k = q/n$) ดังนั้นค่าที่จะนำไปใส่ลงในช่องอาร์เรย์ 1 ช่อง คือ ค่าที่ได้จากการนำค่าทุกพิกเซลใน $j*k$ พิกเซลมาหาค่าเฉลี่ย ยกตัวอย่างเช่น ภาพอินพุตมีขนาด $200*300$ พิกเซล และอาร์เรย์สำหรับเก็บค่าตัวแทนรูปภาพมีขนาด $10*10$ ช่องอาร์เรย์ ดังนั้นกลุ่มของพิกเซลที่รับมาเพื่อเก็บใน 1 ช่องอาร์เรย์จะมีขนาดเป็น $(200/10)*(300/10) = 20*30$ พิกเซล และค่าที่จะนำลงในช่องอาร์เรย์นี้คือค่าเฉลี่ยของทุก ๆ พิกเซลใน $20*30$ พิกเซลนี้ พิจารณาดังรูปที่ 3-6



รูปที่ 3-6 แสดงการสร้างค่าตัวแทนรูปภาพจากภาพอินพุต

สำหรับในโครงการนี้ได้กำหนดค่าสำหรับเก็บค่าตัวแทนรูปภาพเป็นอาเรย์ที่มีขนาด 10×10 พิจารณาดังรูปที่ 3-7



อาเรย์แถวที่ 1	203	202	196	152	96	91	114	175	201	200
อาเรย์แถวที่ 2	203	131	36	12	18	22	18	17	132	202

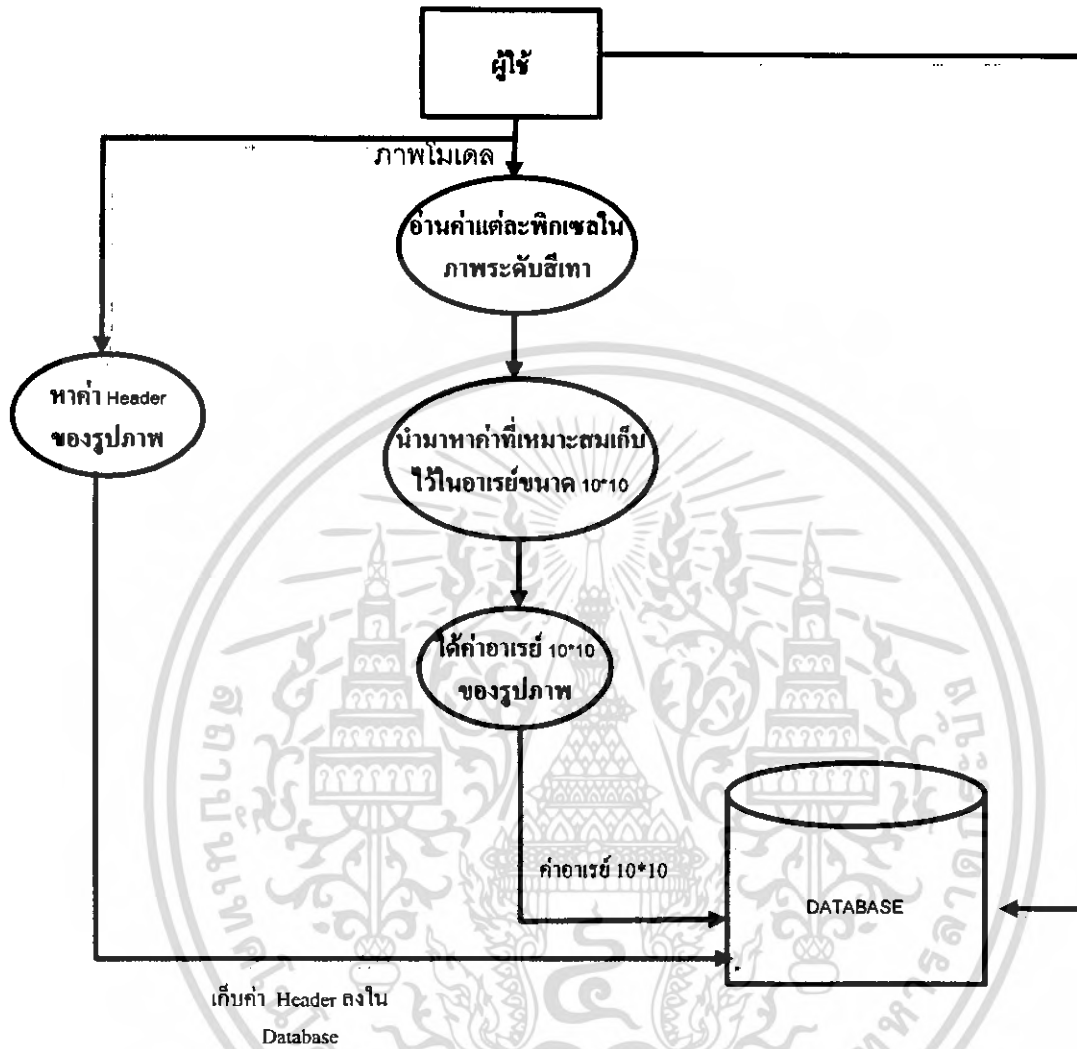
อาเรย์แถวที่ 10	89	51	72	129	111	98	103	43	22	63
-----------------	----	----	----	-----	-----	----	-----	----	----	----

รูปที่ 3-7 แสดงการนำภาพอินพุตมาแปลงเป็นข้อมูลอาเรย์ขนาด 10×10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.3. เก็บรูปภาพ ข้อมูลรูปภาพ และค่าตัวแทนรูปภาพลงฐานข้อมูล แสดงดังรูปที่

3-8



รูปที่ 3-8 แสดงขั้นตอนการทำงานของกรเก็บภาพและค่าตัวแทนของภาพนิ่งจากฐานข้อมูล

3.2.2 การเรียกค้นข้อมูลรูปภาพ

ดังแสดงในรูปที่ 3-9 โดยมีหลักการทำงานดังนี้

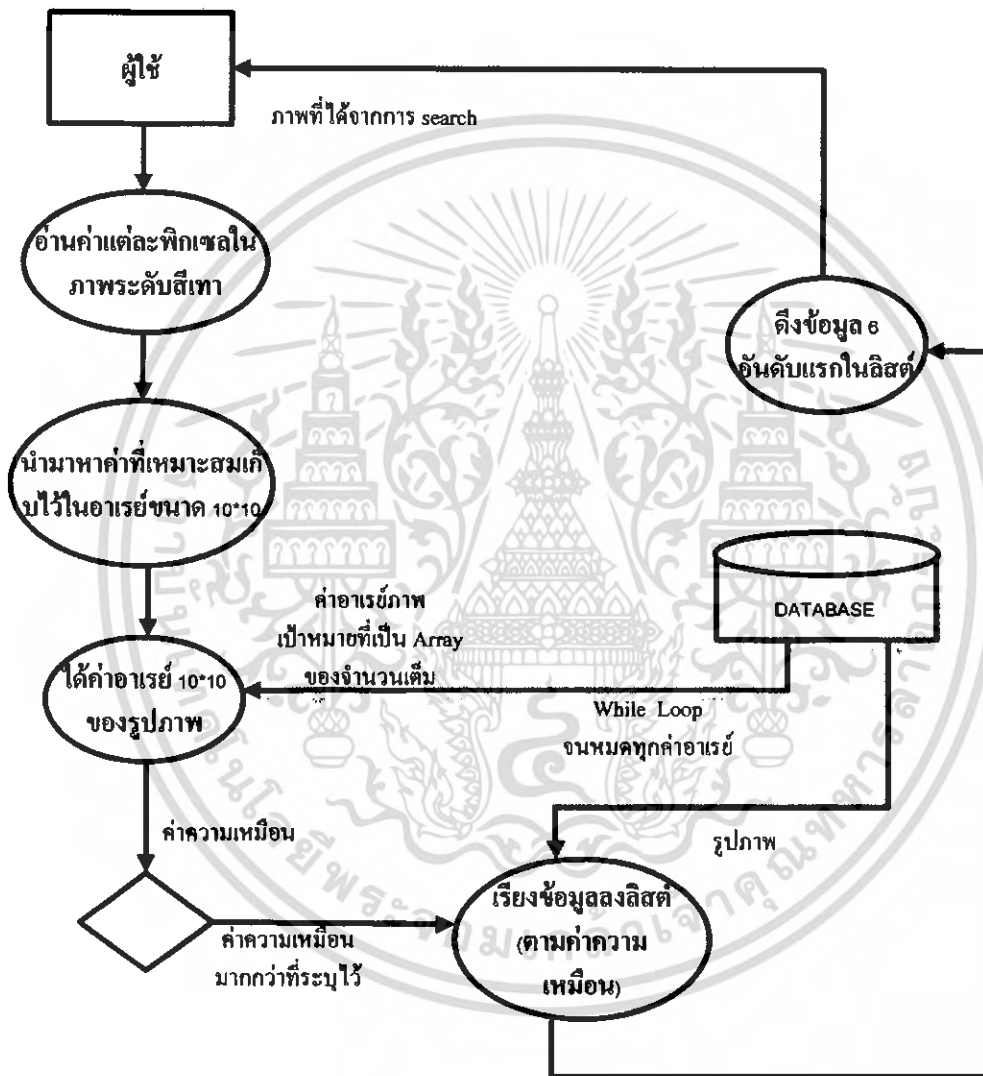
3.2.2.1 รับข้อมูลรูปภาพที่จะทำการเรียกค้นเข้ามา

3.2.2.2 อ่านค่าแต่ละพิกเซลในรูปภาพในระดับสีเทา และหาค่าตัวแทนของรูปภาพ

ตามหลักการในข้างต้นเช่นเดียวกับการเพิ่มรูปภาพลงฐานข้อมูล เพื่อที่จะนำค่าตัวแทนของรูปภาพมาเปรียบเทียบกับค่าตัวแทนของรูปภาพที่อยู่ในฐานข้อมูล

3.2.2.3 เมื่อได้ค่าตัวแทนรูปภาพของรูปภาพที่จะมาทำการเรียกคืนแล้ว ระบบจะนำค่านี้มาเปรียบเทียบกับทุก ๆ ค่าที่มีอยู่ในฐานข้อมูล โดยจะคิดหาเปอร์เซ็นต์ความเหมือนกันระหว่างรูปภาพที่มาทำการเรียกคืนกับรูปภาพแต่ละรูปที่อยู่ในฐานข้อมูล

3.2.2.4 ระบบจะแสดงรูปภาพที่อยู่ในฐานข้อมูลตามระดับเปอร์เซ็นต์ความเหมือนกันที่คำนวณได้



รูปที่ 3-9 แสดงขั้นตอนการทำงานของการค้นหาและเรียกภาพนิ่งจากฐานข้อมูล

บทที่ 4

Caché

4.1 บทนำ

ระบบจัดการฐานข้อมูลเชิงวัตถุที่ใช้ (ODBMS) ในโครงการนี้ได้แก่ Caché พัฒนาโดย InterSystems Corporation ซึ่งเป็นระบบจัดการฐานข้อมูลที่สนับสนุนเทคโนโลยีเชิงวัตถุ

InterSystems Caché® คือผลิตภัณฑ์ยุคใหม่ของเทคโนโลยีฐานข้อมูลที่มีประสิทธิภาพสูง (Ultra-high-performance database) ซึ่งได้รวบรวม Object database, high-performance SQL และ powerful multidimensional data access เอาไว้ ซึ่งทั้งหมดนั้นสามารถเข้าถึง(access)ข้อมูล (data)บนข้อมูลเดียวกันได้ในเวลาเดียวกันได้ Caché ได้รวบรวมจัดเตรียมระดับของ performance ความสามารถในการพัฒนาขยายออกไปในอนาคต การเขียนโปรแกรมที่รวดเร็ว และการใช้งานที่ง่าย ที่เทคโนโลยีของ relational ทำไม่ได้

แต่ Caché เป็นมากกว่าเทคโนโลยีฐานข้อมูลเพียงอย่างเดียว (pure database) เนื่องจาก Caché ได้รวมเอา Application Server ซึ่งมีพร้อมทั้งด้วย ความสามารถในการเขียน โปรแกรมเชิง วัตถุ และมีความสามารถในการรวมหลาย ๆ เทคโนโลยีเข้าไว้ด้วยกัน

Caché มาพร้อมกับ built-in scripting language หลายภาษาด้วยกัน คือ

-Caché ObjectScript ซึ่งมีความสามารถที่ง่ายต่อการเรียนรู้ภาษาของการเขียนโปรแกรมเชิงวัตถุ (object-oriented programming language)

-Caché Basic ซึ่งก็คือภาษาที่ครอบคลุมภาษาโปรแกรม widespread Basic ที่ได้รวมการขยายการเข้าถึงข้อมูลที่ทรงประสิทธิภาพและเทคโนโลยีเชิงวัตถุ และ

-Caché MVBasic คือภาษาที่ปรับเปลี่ยนมาจาก Basic ที่ใช้โดย MultiValue application

-ภาษาอื่น ๆ อีก เช่น Java ,C# และ C++ ซึ่งได้รับการสนับสนุนผ่านทาง call-in หรือ อินเตอร์เฟส (interface) อื่นๆ ดังเช่น ODBC ,JDBC, .NET , และ Caché provided object interface ซึ่งยอมให้เข้าถึงฐานข้อมูล Caché และสิ่งอื่น ๆ ของ Caché เช่น คุณสมบัติ (property) และ เมธอด (method)

Caché ยังได้พัฒนาไปจากฐานข้อมูลแบบเดิม ๆ โดยได้รวม rich environment สำหรับการ พัฒนา browser-based (Web) application ที่สลับซับซ้อน เทคโนโลยี Caché Server Pages(CSP) ที่ให้การพัฒนาที่รวดเร็วและรองรับการทำงานในการสร้างของ Webpages ผู้ใช้งาน Web พร้อม ๆ กันหลายพันคนสามารถเข้าถึง database application แม้ว่าอยู่บนฮาร์ดแวร์ราคาถูก

สำหรับในส่วนที่ไม่ใช่ Web application ซึ่งก็คือ user interface จะถูกโปรแกรมโดยเข้า กับแบบอย่างหนึ่งในเทคโนโลยี client user interface เช่น Java ,.NET ,Delphi ,C# หรือ C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ที่ดีที่สุด (ในที่นี้หมายถึง การเขียนโปรแกรมที่เร็วที่สุด performance ที่ดีที่สุด และการบำรุงรักษา (maintenance) ที่น้อยที่สุด) สามารถพัฒนาได้ทั้งหมดภายใน Caché อย่างไรก็ตาม Caché ก็ได้จัดหาการทำงานร่วมกันกับเทคโนโลยีอื่น ๆ และสนับสนุน development tools ปกติทั้งหมด ซึ่งรองรับการพัฒนาขยายระบบออกไปได้ในอนาคต

4.2 Data Modeling-Relational or Object Access

การดำเนินการแรก ๆ ของการออกแบบ (design) แอปพลิเคชันใหม่ ผู้พัฒนาต้องตัดสินใจที่จะเลือกการเข้าถึงยัง data model ส่วนใหญ่แล้วก็จะเลือกระหว่างโมเดลดั้งเดิมของข้อมูลที่เป็นตารางความสัมพันธ์ (Relational table) และโมเดลที่ใหม่กว่าอย่างเช่นออบเจกต์ (object) เมื่อต้องเจอกับความจำเป็นที่จะต้องจัดการกับข้อมูลที่ซับซ้อน ผู้พัฒนาหลาย ๆ คนเชื่อโมเดลแบบออบเจกต์นั้น มีประสิทธิภาพในการเข้าถึงข้อมูลมากกว่า

ซึ่งเป็นเรื่องที่น่าพอใจว่า เมื่อทำการย้ายแอปพลิเคชันที่มีอยู่ไปที่ Caché ขั้นตอนแรกคือต้องย้ายโมเดลข้อมูลที่มีอยู่ มีทางง่าย ๆ ที่จะนำเข้า (import) โมเดลข้อมูลจากรูปแบบตารางความสัมพันธ์หรือออบเจกต์ที่หลากหลายไปเป็นผลลัพธ์ที่เป็นการกำหนดข้อมูล Caché ที่เป็นมาตรฐาน เมื่อข้อมูลย้ายไปที่ Caché ข้อมูลจะสามารถถูกเข้าถึงในเวลาเดียวกันทั้งแบบออบเจกต์ ตารางความสัมพันธ์ และแบบ multidimensional array

Caché สนับสนุนทั้ง SQL และการเข้าถึงข้อมูลด้วยวิธีการเชิงวัตถุ มันจะเป็นประโยชน์อย่างมากถ้าสามารถที่จะเข้าใจได้ว่าแต่ละแบบนี้ทำงานอย่างไรและเพราะเหตุใดที่ต้องใช้แบบนี้

4.3 Caché Object Model และ Object Programming

Caché Object Model ขึ้นกับมาตรฐาน ODMG(Object Database Management Group) และสนับสนุน feature ที่ก้าวหน้าหลายอย่าง รวมถึง multiple inheritance

เทคโนโลยีของออบเจกต์ นั้นพยายามที่จะเข้าถึงพฤติกรรมการคิดและการใช้ข้อมูลของมนุษย์จริง ๆ ซึ่งไม่เหมือนตารางความสัมพันธ์ ออบเจกต์จะบรรจุทั้งข้อมูลและเมธอดเข้าไว้ด้วยกัน ตัวอย่างเช่น ออบเจกต์ของใบรายการสินค้าต้องมีข้อมูลเช่น เลขที่ใบรายการสินค้าและราคารวม และเมธอด เช่น Print()

ตามคอนเซปต์แล้ว ออบเจกต์คือแพ็คเกจ (package) ซึ่งได้รวมค่าข้อมูลของออบเจกต์ (property) และเมธอดทั้งหมด (method) เมธอดของออบเจกต์จะส่งเมสเสจ (message) เพื่อทำการสื่อสารกับเมธอดอื่น ๆ และเพื่อลดการเก็บข้อมูลที่เป็นออบเจกต์ที่มาจากคลาสเดียวกันจะใช้เมธอดร่วมกัน สำหรับใน Caché ก็เช่นกัน เมธอดจะให้ผลลัพธ์ผ่านทางฟังก์ชันที่มี

ประสิทธิภาพ มากกว่าที่จะใช้การส่งผ่านเมสเซจที่มี overhead อย่างไรก็ตามเทคนิคการเขียนโปรแกรมเหล่านี้จะถูกซ่อนโดยโปรแกรมเมอร์เนื่องจากเป็นเรื่องของ object passing message

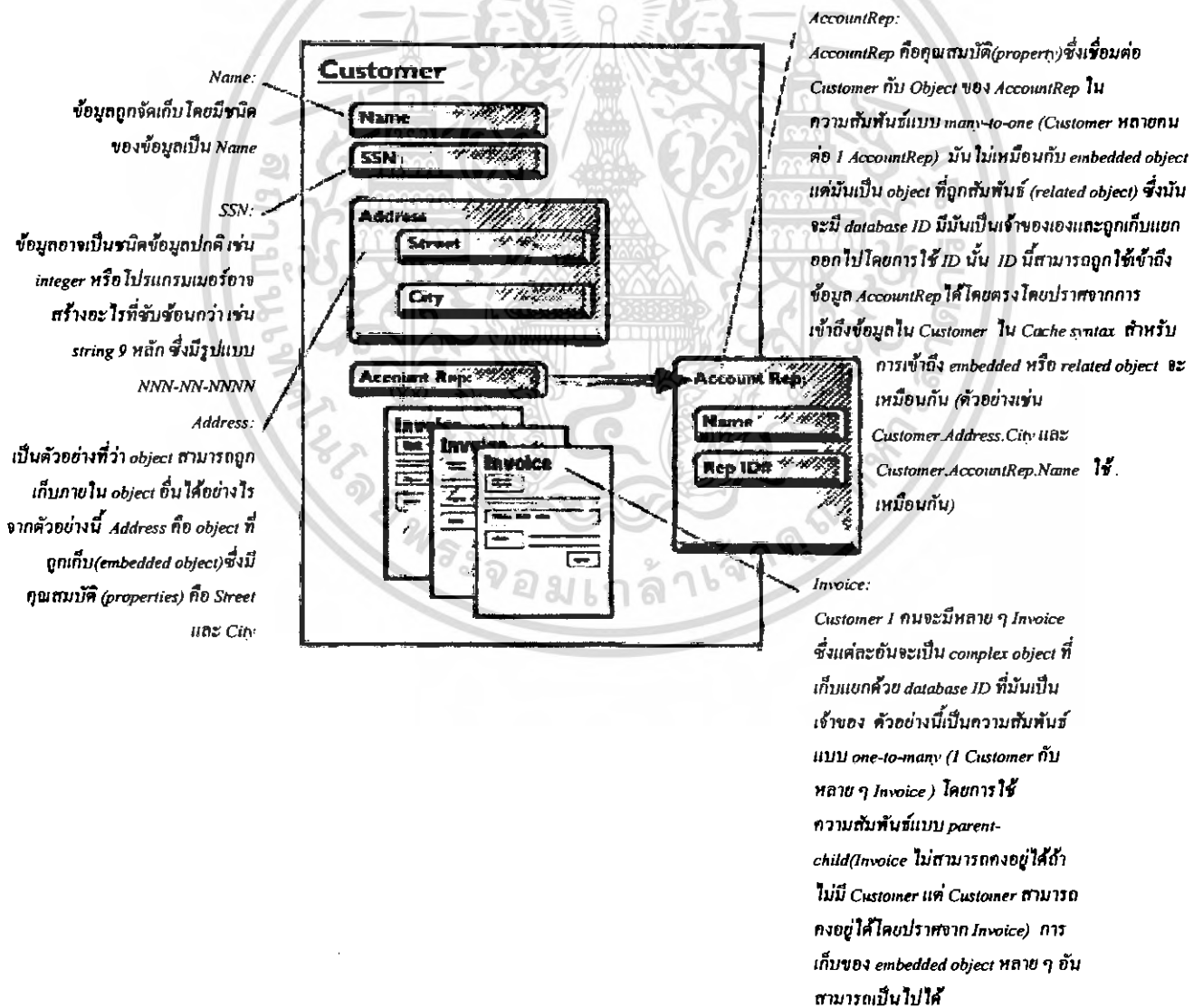
ความแตกต่างระหว่างออบเจกต์ (object) และ คลาส(class) ก็คือ

คลาส คือ โครงสร้างและเมธอดที่กำหนดโดยโปรแกรมเมอร์ มันได้รวมคำอธิบายของธรรมชาติของข้อมูล (ซึ่งก็คือชนิดของข้อมูล หรือ "type") และวิธีการ(method)

ออบเจกต์ คือ ตัวอย่างเฉพาะของคลาส

ตัวอย่างเช่น เลขที่ใบแจ้งรายการสินค้า (Invoice) #123456 คือออบเจกต์ของคลาสใบแจ้งรายการสินค้า

เทคโนโลยีของออบเจกต์แสดงข้อมูลที่เป็นธรรมชาติ ไม่จำกัดเฉพาะคุณสมบัติที่ธรรมดา ออบเจกต์อาจจะบรรจุอยู่ในออบเจกต์อื่นได้ หรือ อาจจะไปยังออบเจกต์อื่น ดังรูปที่ 4-1



รูปที่ 4-1 แสดงตัวอย่างของคลาสและคุณสมบัติภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับแอปพลิเคชันของฐานข้อมูลใหม่ ๆ ผู้พัฒนาส่วนใหญ่เลือกที่จะใช้เทคโนโลยีของ ออบเจกต์ เพราะว่าสามารถพัฒนาแอปพลิเคชันที่ซับซ้อนได้รวดเร็วกว่าและแก้ไขปรับปรุงใน ภายหลังได้ง่ายกว่า เทคโนโลยีออบเจกต์มีประโยชน์หลายอย่าง คือ

- ออบเจกต์สนับสนุนโครงสร้างของข้อมูลที่เป็นข้อมูลที่อยู่ใน โลกแห่งความเป็นจริง
- มีการเขียนโปรแกรมที่ง่ายกว่า คือ ง่ายในการรักษาขอบเขตของงาน
- มีคอนเซปต์ของ black box (กล่องดำ) หรือก็คือการเอนแคปซูเลชัน (encapsulation) ที่ โปรแกรมเมอร์สามารถแก้ไขปรับปรุงการทำงานภายในได้โดยที่ไม่กระทบกับแอปพลิเคชัน
- ออบเจกต์จัดหาทางที่ง่ายที่จะนำไปสู่การติดต่อกับเทคโนโลยีหรือแอปพลิเคชันที่แตกต่าง
- เทคโนโลยีออบเจกต์สามารถ match กับ graphical user interface ได้อย่างลงตัว
- เครื่องมือ (tool) ใหม่ ๆ สนับสนุนเทคโนโลยีออบเจกต์

ออบเจกต์ มีการแยกกันที่ชัดเจนระหว่างส่วนติดต่อกับผู้ใช้ (user interface) กับแอปพลิเคชัน ส่วนอื่น ๆ ดังนั้นเมื่อมีความจำเป็นที่จะต้องเปลี่ยนมาใช้เทคโนโลยี user interface ใหม่ ๆ (ซึ่งบางที อาจเป็นเทคโนโลยีที่คาดไม่ถึงในอนาคต) ก็จะสามารถนำโค้ด เกือบทั้งหมดกลับมาใช้ใหม่ได้

4.4 SQL Access

SQL คือ query language สำหรับ Caché และสนับสนุนความสามารถฐานข้อมูลเชิงสัมพันธ์ (relational database) อย่างเต็มรูปแบบ ซึ่งได้รวม DDL ,Transactions ,referential integrity ,triggers ,stored procedure และอื่น ๆ Caché สนับสนุนการเข้าถึงผ่าน ODBC และ JDBC (การใช้ pure Java-based driver) คำสั่งและ query ของ SQL สามารถฝังลงใน Caché ObjectScript และ ภายในเมธอดของออบเจกต์

Caché SQL ได้รวมคุณสมบัติทางออบเจกต์ไว้ซึ่งทำให้โค้ด SQL เขียนได้ง่ายกว่าและดู เข้าใจในการที่จะ read และ write มากกว่า

ตัวอย่างเช่น

ตารางที่ 4-1 ตารางเปรียบเทียบการใช้ภาษา SQL แบบ TRADITIONAL SQL กับ OBJECT

EXTENDED SQL

TRADITIONAL SQL	OBJECT EXTENDED SQL
SELECT SC.FullName, SM.Descr, MS.Value, SI.InvDate, SI.InvNumber FROM MainSales MS, SalesItem SI, SalesProduct SP, SalesCustomer SC, SalesMarket S WHERE SI.SalesItemID *= MS.SalesItem AND SP.SalesProductID *= MS.Product AND SC.SalesCustomerID *= MS.Customer AND SM.SalesMarketID *= SC.SalesMarket AND SP.Descr = 'Hammer'	SELECT Customer->FullName, Customer->SalesMarket->Descr, Value, SalesItem->InvDate, SalesItem->InvNumber FROM MainSales WHERE Product->Descr = 'Hammer'

4.5 Caché Object

Object model ของ Caché ขึ้นกับมาตรฐาน ODMG Caché สนับสนุนขอบเขตความรู้ทั้งหมดของการเขียนโปรแกรมเชิงวัตถุ ซึ่งได้รวมการเอนแคปซูลเลชัน (encapsulation) , embedded object ,multiple inheritance , พอลิมอร์ฟิซึม (polymorphism) และ collection

ภาษาที่ built-in ภายใน Caché นั้นจัดการโดยตรงกับ object เหล่านี้ และ Caché ใช้ class ของ Caché ดังเช่น Java, EJB, COM, .NET และ C++ class คลาสทั้งหลายของ Caché สามารถทำงานกับการสนับสนุนด้าน XML และ SOAP ได้อย่างอัตโนมัติโดยการคลิกที่ปุ่มใน Studio IDE ดังนั้น object ทั้งหมดของ Caché ก็พร้อมทุกเมื่อในทุก ๆ การใช้เทคโนโลยี object

มีหลายทางสำหรับการเขียนโปรแกรมภายนอก Caché Application Server เพื่อเข้าถึงคลาสของ Caché

1. Caché class ทั้งหมดสามารถถูกแสดงออกมาเหมือนดังเช่นคลาสในภาษาทั่วไป เมื่อ Java, C++, C#, หรือ โปรแกรมอื่น ๆ เข้าถึง Caché object เรียกว่า template ของคลาสในภาษาทั่วไป template class ดังกล่าวนั้น (ซึ่งถูกสร้าง(generate)อย่างอัตโนมัติโดย Caché) จะสื่อสารกับ Caché Application Server เพื่อร้องขอเมธอดบน Caché server และเข้าถึงข้อมูลหรือแก้ไขปรับปรุงคุณสมบัติต่าง ๆ สถานะของ Caché object ถูกดูแลจัดการใน Caché Application Server ในการที่จะเพิ่มความเร็วในการคำนวณประมวลผลและลดการส่งข่าวสาร (messaging) Caché จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซ่อนหรือเก็บก๊อปปี้ของข้อมูลของออบเจกต์บน client และแบกภาระการ update กับ message อื่นเมื่อเป็นไปได้

2. template class เข้าถึงฐานข้อมูลโดยตรง เกี่ยวกับ Application Server คุณสมบัติ(property) ต่างๆ ภายในหน่วยความจำ (memory) ถูกดูแลจัดการเพียงใน client การเข้าถึงนี้จัดหา throughput ที่สูงกว่าแค่การทำงานตามบทบาทหน้าที่ที่ลดลง เนื่องจากตัวอย่างเมธอดของคลาสข้าง server (เช่นเมธอดที่จำเป็นจะต้องเข้าถึงคุณสมบัติต่าง ๆ ภายในหน่วยความจำ) ไม่สามารถถูกร้องขอได้

3. เทคโนโลยี InterSystems Jalapeño อนุญาตให้ผู้พัฒนา Java ในอันดับแรกจะสร้าง Java database class เหมือนดังเช่นคลาส POJO(plain old Java object) อื่น ๆ ใน IDE และต่อมา Caché จะสร้าง database schema และ Caché class ที่ตรงกันให้โดยอัตโนมัติ การใช้การเข้าถึงนี้ Java class จะไม่ถูกเปลี่ยน และแอปพลิเคชันจะดำเนินการเข้าถึงคุณสมบัติ(property) และเมธอด (method) ต่าง ๆ ของมันต่อไป Caché จัดหา library class (“Object Manager”) เข้ากับ API ซึ่งถูกใช้เพื่อเก็บและเข้าถึงออบเจกต์ของฐานข้อมูลและสั่ง query

แต่ละการเข้าถึงในสามแบบนี้ ออบเจกต์จะปรากฏอยู่ใน local ภายในโปรแกรม ผู้ใช้งาน Caché จัดการการสื่อสารทั้งหมดอย่างเปิดเผยโดยการใช้ call-in หรือ TCP

Java template และ library ที่สนับสนุนเป็น Java-based อย่างสมบูรณ์ ดังนั้นมันสามารถถูกใช้ข้าม Web หรือบนเครื่องมือพิเศษต่าง ๆ แบบ Java

4.6 Caché Application Server

Caché Application Server ได้เสนอความสามารถของความก้าวหน้าของการเขียนโปรแกรมเชิงวัตถุ (object programming) โดยมีการจัดเก็บข้อมูลที่สลับซับซ้อน และได้รวมการเข้าถึงหลาย ๆ เทคโนโลยีอย่างง่าย ๆ Caché Application Server ทำเหล่านี้ให้เป็นไปได้ที่จะพัฒนา database application ที่ซับซ้อนได้อย่างรวดเร็ว มีการจัดการที่มีประสิทธิภาพสูง และมีการสนับสนุนอย่างง่าย ๆ

ลักษณะเฉพาะที่มากไปกว่านั้น Caché จัดหา :

- การเชื่อมต่อซอฟต์แวร์กับฝั่ง client
- SQL เข้าถึงโดยการใช้ ODBC(Open Database Connectivity) และ JDBC(Java Database Connectivity) ซึ่งได้รวมการเก็บที่สลับซับซ้อนที่ client และ application server เพื่อประสิทธิภาพที่สูงกว่า
- การเข้าถึงฐานข้อมูลเชิงสัมพันธ์ (relational database)
- Caché server page เพื่อประสิทธิภาพที่สูงขึ้น ซึ่งก็คือง่ายที่จะเขียนโปรแกรม Web Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โค้ดสำหรับภาษา script ถูกเก็บในฐานข้อมูลและสามารถเปลี่ยนแปลงขณะออนไลน์ กับ การเปลี่ยนแปลงนั้นจะถ่ายทออย่างอัตโนมัติไปสู่ application server ทั้งหมด

4.7 JAVA

Java คือ ภาษาการเขียนโปรแกรมที่เป็นที่นิยม แต่การเชื่อมต่อ Java application กับ ฐานข้อมูลส่วนใหญ่กำลังเป็นที่ทำหาย การเชื่อมต่อไปยัง relational database จะต้องทำการเขียน โค้ด SQL ที่ใหญ่มาก ซึ่งใช้เวลานานและทำให้ประโยชน์หลาย ๆ อย่างจากเทคโนโลยีเชิงวัตถุของ Java หดหายไป การเข้าถึงของ Caché ไปยังออบเจกต์ที่เก็บไว้โดยตรงปราศจากความกังวลของ ผู้พัฒนาที่ว่าข้อมูลจะถูกเก็บถาวรอย่างไรและการใช้ object syntax ในการเข้าถึงฐานข้อมูลง่ายกว่า มาก และนิยมมากกว่าด้วย

ผู้พัฒนาบางคนชอบที่จะกำหนด database schema เป็นอันดับแรกและต่อมา Java class ที่ สอดคล้องกับ database schema สำหรับแต่ละฐานข้อมูลก็จะถูกสร้างอย่างอัตโนมัติ ในขณะที่บาง คนชอบที่จะสร้าง Java class ก่อนและให้ Caché สร้าง database schema ให้อย่างอัตโนมัติ

Caché สนับสนุนการเข้าถึงเหล่านี้ทั้งหมด คือ:

- Caché class บางคลาสสามารถถูกแปลงไปเป็น Java class ดังนั้นคุณสมบัติ (property) และ เมธอด สามารถถูกเข้าถึง โดย Java object
- Caché class สามารถถูกแปลงไปเป็น Enterprise Java Beans
- JDBC จัดหา SQL ที่มีประสิทธิภาพสูง เข้าถึงโดยการใช้ไดรเวอร์ Java-based (Type 4) ที่ สมบูรณ์

4.7.1 Object Access through Projected Classes

ทุก ๆ Caché class สามารถไปเป็น Java (หรือ EJB) class ที่มีการทำงานที่สอดคล้องกัน ด้วยทุก ๆ คุณสมบัติ (property) และเมธอดของ Caché class ในการเขียนโปรแกรม Java คลาส เหล่านี้จะดูเหมือน Java class ทั่วไปอื่น ๆ Java class ที่ถูกสร้าง ใช้ Java library ซึ่งหามาโดย InterSystems เพื่อรองรับการสื่อสารระหว่าง client และ server

สถานะของทุก ๆ Caché object ถูกดูแลจัดการใน Caché Application Server ถึงแม้ว่า คุณสมบัติ (property) ของคลาสนี้จะถูกเก็บไว้ใน client เพื่อทำให้มีประสิทธิภาพ Java method จะ เรียกชื่อของเมธอดที่สอดคล้องกันบน Caché Application Server คือได้รวมเมธอดที่เก็บ ออบเจกต์ภายในฐานข้อมูลและจะเข้าถึงในเวลาต่อไป มันเปิดเผยต่อ client ดังเช่น Caché Data Server บรรจุข้อมูลหรือถึงแม้ว่าถ้าข้อมูลออบเจกต์จะถูกเก็บในฐานข้อมูลเชิงสัมพันธ์ก็就会被เข้าถึง ผ่าน Caché Application Server

4.7.2 Caché Methods Written in Java

เมธอดของ Caché class สามารถถูกเขียนด้วย Java ใน Caché Studio อย่างไรก็ตามจะไม่เหมือน Caché ObjectScript และ Basic Java method ไม่ถูกประมวลผลโดย Caché Virtual Machine แต่มันจะถูกรวบรวมตอนสร้าง Java class และ ประมวลผลใน Java Virtual Machine เมื่อเป็นเช่นนี้แล้ว โค้ดจะไม่สามารถเข้าถึงจากเมธอดที่ไม่ใช่ Java



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดสอบ

5.1 การทดสอบที่ 1

การนำเอาฐานข้อมูลเชิงวัตถุมาประยุกต์ใช้งานกับระบบลงทะเบียนนิสิตอย่างง่าย จากตอนต้นได้กล่าวถึงแนวความคิดของฐานข้อมูลเชิงวัตถุและประโยชน์ของฐานข้อมูลเชิงวัตถุ ต่อไปจะเป็นการยกตัวอย่างของการนำเอาระบบฐานข้อมูลเชิงวัตถุมาประยุกต์ใช้งานกับระบบงานเพื่อทำการทดสอบตามแนวความคิดที่ได้ศึกษามาในตอนต้นและได้ทำการสรุปถึงข้อดีและข้อด้อยของการนำแนวคิดดังกล่าวไปใช้งานกับระบบงานจริงโดยในโครงการนี้จะนำเอาระบบลงทะเบียนของนิสิตมาเป็นตัวอย่างในการศึกษา โดยซอฟต์แวร์ Cache ในการจัดการระบบฐานข้อมูลเชิงวัตถุ

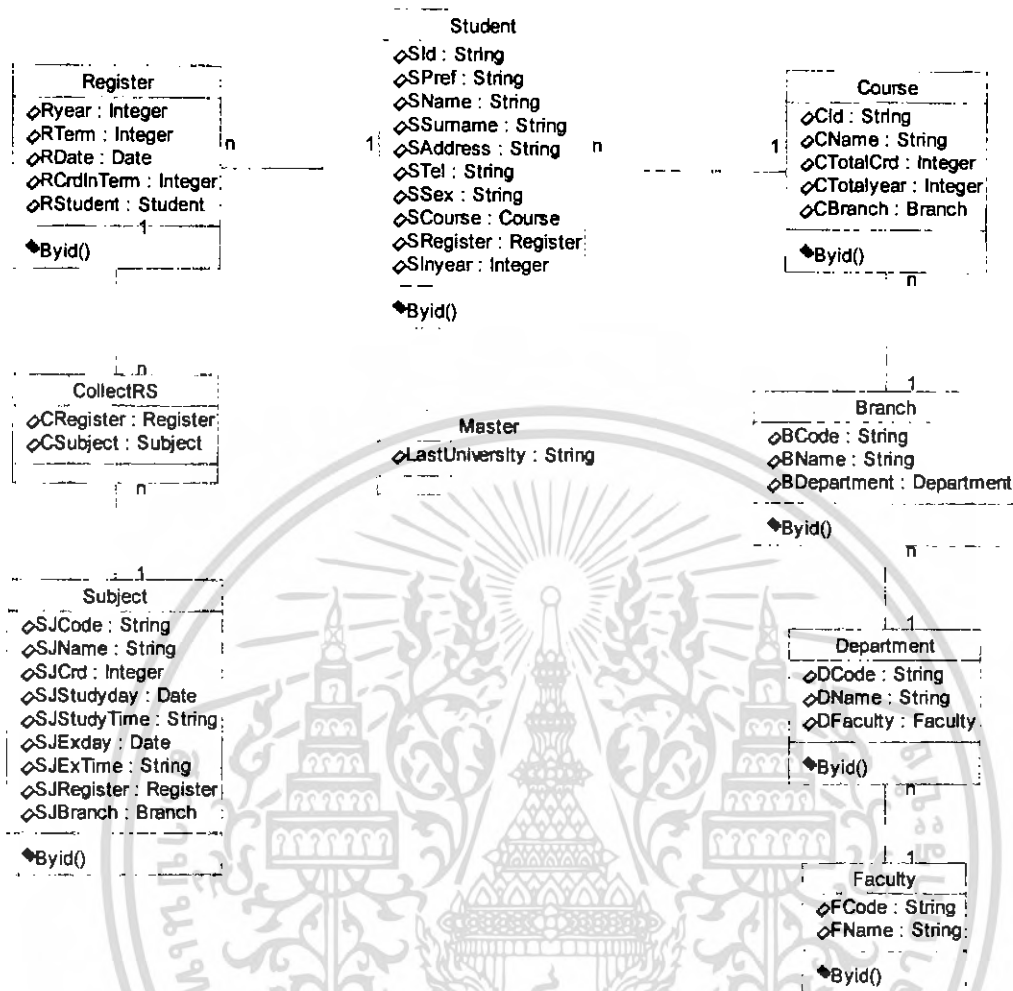
5.1.1 การทดสอบ

ระบบงานลงทะเบียนนิสิต

1. ทะเบียนประวัตินิสิต , รายละเอียดแต่ละวิชา
2. สามารถแก้ไขเกี่ยวกับวิชาเรียน
3. การลงทะเบียนเรียน
4. เพิ่ม เปลี่ยน ถอน วิชาเรียนได้

คลาสไดอะแกรมของระบบลงทะเบียนนิสิต

หลังจากที่ได้ทำการศึกษาระบบลงทะเบียนนิสิตได้ทำการออกแบบโดยใช้คลาสไดอะแกรมในการแสดงความสัมพันธ์ระหว่างคลาสต่าง ๆ จากนั้นจะนำคลาสไดอะแกรมที่ได้ออกแบบไว้มาใช้ในการทำฐานข้อมูล โดยข้อมูลที่ต้องการเก็บลงฐานข้อมูลให้กำหนดเป็นคุณสมบัติของคลาสนั้น



รูปที่ 5-1 คลาสไดอะแกรมของระบบลงทะเบียนนักศึกษา

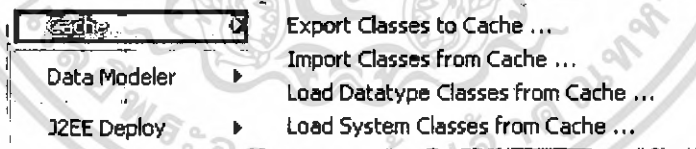
การทดลองได้ออกแบบคลาสไดอะแกรมจากโปรแกรม Rational Rose โดยมีรายละเอียดในแต่ละคลาสดังต่อไปนี้

- Class Student เป็นคลาสเกี่ยวกับนักศึกษา โดยจะมีรายละเอียดเกี่ยวกับนักศึกษา คือ รหัสนักศึกษา, ชื่อนักศึกษา, นามสกุล, ที่อยู่, เบอร์โทรศัพท์, ปีที่เข้ารับการศึกษ, หลักสูตรที่เรียน, รายละเอียดการลงทะเบียน
- Class Course เป็นคลาสเกี่ยวกับหลักสูตร โดยจะมีรายละเอียดเกี่ยวกับหลักสูตร คือ รหัสหลักสูตร, ชื่อหลักสูตร, จำนวนหน่วยกิตทั้งหมด, จำนวนปีที่ใช้ในการศึกษา, และสาขาวิชาของหลักสูตร
- Class Branch เป็นคลาสเกี่ยวกับสาขาที่เรียน โดยจะมีรายละเอียดเกี่ยวกับสาขาวิชา คือรหัสสาขา, ชื่อสาขาวิชา, และภาควิชาของสาขาวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

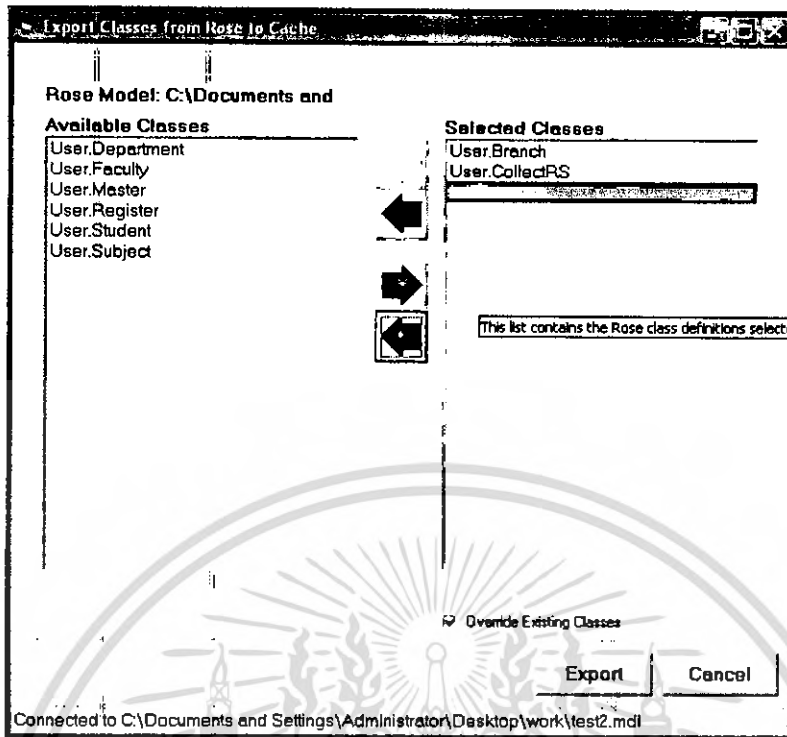
- Class Department เป็นคลาสเกี่ยวกับภาควิชา โดยจะมีรายละเอียดเกี่ยวกับภาควิชา คือ รหัสภาควิชา, ชื่อภาควิชา, และคณะของภาควิชา
- Class Faculty เป็นคลาสเกี่ยวกับคณะ โดยจะมีรายละเอียดเกี่ยวกับคณะ คือ รหัสคณะ, ชื่อคณะ
- Class Subject เป็นคลาสเกี่ยวกับวิชา โดยจะมีรายละเอียดเกี่ยวกับวิชา คือ รหัสวิชา, ชื่อวิชา, จำนวนหน่วยกิต, วันเรียน, เวลาเรียน, วันสอบ, เวลาสอบ, สาขาวิชาของวิชา, รายละเอียดการลงทะเบียน
- Class Register เป็นคลาสเกี่ยวกับการลงทะเบียน โดยจะมีรายละเอียดเกี่ยวกับรายละเอียดการลงทะเบียน คือ ภาคเรียน, ปีการศึกษา, วันที่ลงทะเบียน, นักศึกษาที่ลงทะเบียน, หน่วยกิตรวมในการลงทะเบียนในครั้งนั้น
- CollectRSJ เป็นคลาสเก็บความสัมพันธ์ของการลงทะเบียนแต่ละครั้งและวิชาที่ลง โดยจะมีรายละเอียด คือ รายละเอียดการลงทะเบียน, และวิชาในการลงทะเบียน
- Master เป็นคลาสเกี่ยวกับนักศึกษาปริญญาโท จะเก็บรายละเอียดเกี่ยวกับนักศึกษาปริญญาโท

เมื่อทำการออกแบบคลาสไดอะแกรมจาก Rational Rose แล้ว โปรแกรม Rational Rose จะมีเครื่องมือที่ใช้ในการ export คลาสไดอะแกรมไปเป็นคลาสที่ใช้ในการเก็บข้อมูลของฐานข้อมูล Caché โดยการทำการ export ไปที่ cache ได้ โดยเลือกที่ Tool -> Cache -> Export Classes to Cache ... ดังรูปที่ 5-2



รูปที่ 5-2 การ Export จาก Ration Rose ไปยัง Cache

จากนั้นจะมีให้เลือกว่าจะ Export Class อะไรไปบ้าง โดยที่ไม่จำเป็นต้อง Export ไปทั้งหมด ดังแสดงในรูปที่ 5-3



รูปที่ 5-3 แสดงการเลือก คลาสที่จะ export ไปเป็นคลาสที่ใช้ในการเก็บข้อมูล

หลังจากที่ทำการ Export มายัง Cache จะทำการสร้าง โครงร่างของ โค้ด โปรแกรมใน Cache ซึ่งต้องมาใส่เพิ่มเติมในสิ่งที่ต้องการให้ฐานข้อมูลทำเพิ่มเติมจากที่สร้างให้ ดังรูป 5-4 และ รูป 5-5 และรูปที่ 5-6 แสดงการ Cache Inherit

```

Class User.Course Extends &Persistent [ ClassType = persistent ]
(
Property CBranch As User.Branch;
Property CId As %String;
Property CName As %String;
Property CTotalCrds As %Integer;
Property CTotalyear As %Integer;
Relationship TheBranch As User.Branch [ Cardinality = one, Inverse = Courses ];
Relationship Students As User.Student [ Cardinality = many, Inverse = TheCourse ];
Projection PrjName1 As %Projection.Java(ROOTDIR = "C:\Register\User");
EQuery Byid(id As %String) As %SQLQuery(CONTAINID = 1)
(
SELECT %ID FROM Course
WHERE (Cid = :id)
)
)

```

รูปที่ 5-4 แสดงตัวอย่างคำสั่งที่ใช้สร้างฐานข้อมูลเชิงวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Class User.Student Extends %Persistent [ ClassType = persistent ]
{

Property SAddrNo As %String;

Property SAddrMoo As %String;

Property SAddrSoi As %String;

Property SAddrLoc As %String;

Property SAddrDis As %String;

Property SAddrPro As %String;

Property SAddrZipC As %String;

Property SCourse As User.Course;

Property SId As %String;

Property SInyear As %Integer;

Property SName As %String;

Property SPref As %String;

Property SSex As %String;|

Property SSurname As %String;

Property STel As %String;

Relationship TheCourse As User.Course [ Cardinality = one, Inverse = Students ];

Relationship Registers As User.Register [ Cardinality = many, Inverse = TheStudent ];

Projection PrjName1 As %Projection.Java(ROOTDIR = "C:\Register\User");

%Query Byid(id As %String) As %SQLQuery(CONTAINID = 1)
{
SELECT %ID FROM Student
WHERE (SId = :id)
}
}

```

รูปที่ 5-5 คลาส Student ใน Cache

```

Class User.Master Extends User.Student
{

Property lastuniversity As %String;

Projection PrjName1 As %Projection.Java(ROOTDIR = "C:\Register\User");
%Query Byid(name As %String) As %SQLQuery(CONTAINID = 1)
{
SELECT %ID FROM Student
WHERE (SName = :name)
}
}

```

รูปที่ 5-6 คลาส Master ใน Cache Inherit มาจากคลาส Student

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโค้ดโปรแกรมดังกล่าวเมื่อทำการ รันโปรแกรม caché จะทำการสร้างโค้ดภาษาจาวา เก็บไว้ใน “C:\Register\User\” แสดงดังรูปที่ 5-7

```

1305 public void setSName(java.lang.String value) throws com.intersys.objects.CacheException {
1306     com.intersys.cache.Dataholder dh = new com.intersys.cache.Dataholder (value);
1307     mInternal.setProperty(ii_SName, jj_SName, kk_SName, com.intersys.objects.Database.RET_PRIM, "SName", dh);
1308     return;
1309 }
1310
1311 private static int ii_SPref = 16;
1312 private static int jj_SPref = 0;
1313 private static int kk_SPref = 16;
1314 /**

```

รูปที่ 5-7 ตัวอย่างโค้ดโปรแกรมที่ cache สร้างเป็นภาษาจาวา

ซึ่งเมื่อคอมไพล์โค้ดจาวาดังกล่าวแล้วเราจะได้คลาสซึ่งเหมือนเป็นตัวแทนที่ใช้ในการติดต่อกับฐานข้อมูล ซึ่งเวลาจะจัดการกับฐานข้อมูลโดยจะทำการเปลี่ยนค่าคุณสมบัติใด ๆ จะกระทำผ่านเมธอดของออบเจกต์ที่สร้างขึ้นมาก็จะส่งผลกับข้อมูลในฐานข้อมูลด้วย ดังตัวอย่าง

1. ติดต่อกับฐานข้อมูล และการสร้างออบเจกต์ของข้อมูล

```

//ประกาศค่าที่ใช้ในการติดต่อกับฐานข้อมูล
String url = "jdbc:Cache://localhost:1972/USER";
String username = "_SYSTEM";
String password = "SYS";

// ทำการติดต่อกับฐานข้อมูล จากค่าที่ประกาศไว้คอนดั้น
Database dbconnection = CacheDatabase.getDatabase (url,
    username, password);

//สร้าง ออบเจกต์ของข้อมูล
User.Course course = new User.Course(dbconnection);

//กำหนดค่า "01010101" ให้กับ Property CId และทำการกำหนดค่า "program 4 year" ให้กับ Property
CName ให้กับออบเจกต์
course.setCId("01010101");
course.setCName("program 4 year");

//แสดงค่า Cid และ CName ของออบเจกต์

System.out.println("Course ID : " + course.getCId());
System.out.println("Course Name : " + course.getCName());

//ปิดการติดต่อกับฐานข้อมูล
Dbconnection.close();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Output

Course ID : 01010101
 Course Name: program 4 year

2. การเปิดออบเจ็คต์ที่มีอยู่แล้วในฐานข้อมูล

```
//เป็นการติดต่อกับข้อมูลออบเจ็คต์จากฐานข้อมูลเชิงวัตถุโดยใช้ค่า OID ของออบเจ็คต์นั้น
User.Course course1 = (User.Course)User.Course._open
    (dbconnection,new Oid(strid));
System.out.println(course1.getCId());
```

3.การใช้ Class Query

//เป็นการเรียกใช้ query ชื่อ Byid ของคลาส Course โดยภายในจะทำการ query จะทำการ SELECT ค่า ID ตามเงื่อนไขที่กำหนดใน query

```
CacheQuery cq = new CacheQuery(dbconnection,
    "User.Course", "Byid");
ResultSet rs = cq.execute("01010101");
while (rs.next()){
    int i = Integer.parseInt(rs.getString(1));
    Oid oid = new Oid(i);
    User.Course course2 =
    (User.Course)User.Course._open(dbconnection, oid);
    System.out.println(course2.getCName());
}
```

5.1.2 ผลจากการทดลองการทดสอบที่ 1

5.1.2.1 การเพิ่มและค้นหาประวัติของนักศึกษาในฐานข้อมูล

หน้าจอสําหรับการนำประวัติข้อมูลของนักศึกษาลงในฐานข้อมูล แสดงในรูปที่ 5-8 ทำโดยกรอกข้อมูลลงในแบบฟอร์มเมื่อกดปุ่ม Add โปรแกรมก็จะนำข้อมูลที่กรอกเก็บลงในฐานข้อมูล Caché สําหรับการหาประวัติของนักศึกษาทำให้กรอกรหัสนักศึกษาแล้วกดปุ่ม Find

The screenshot shows a web-based form for entering student information. The form is titled "Student Information" and has three tabs: "Student Info.", "Register", and "Add, Change, Delete". The form contains several input fields for student details: Faculty (Engineering), Department (Computer Engineering), Branch (Computer Engineering), Gender (Male, Female), Title (Mr., Miss, Other), ID Student (46010862), Name (Sumalee), Surname (Baehear), Course (4-5 Years Normal, 1-3 Years Con.Normal), Year (2548), Address (No. 497/61, Moo 13, Sol, Local Kongdan, District Bangbo, Province Samutprakan, Zip Code 10550, Telephone Number 023301111). At the bottom, there are buttons for "Add", "Find", and "Clear".

รูปที่ 5-8 การเพิ่มและค้นหาประวัตินักศึกษาในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2.2 การลงทะเบียน

สำหรับการลงทะเบียนให้กรอกข้อมูล คือ รหัสนักศึกษา , ปีการศึกษา , ภาคเรียน และ วันที่ทำการลงทะเบียน จากนั้นใส่รหัสวิชาที่จะทำการลงทะเบียน เมื่อกด enter จะแสดงชื่อของวิชานั้น และสามารถเรียกดูว่าวิชานี้มีนักศึกษาคนไหนบ้างที่ลงทะเบียนในวิชานี้ โดยกลุ่มที่อยู่ต่อจากชื่อวิชา (check1 ,check2,check3,check4,check5) เมื่อทำการกรอกข้อมูลครบและแน่ใจที่จะลงทะเบียน แล้วให้กดปุ่ม OK โปรแกรมก็จะนำข้อมูลที่กรอกเก็บลงในฐานข้อมูล Cache แสดงดังรูปที่ 5-9

The screenshot shows a window titled 'Student Information' with three tabs: 'Student Info.', 'Register', and 'Add,Change,Delete'. The 'Student Info.' tab is active, displaying the following information:

- Student ID: 46010862
- Student Name: Miss. Sumalee Saehear
- Year: 2549
- Term: 1
- Date: 03-04-2549

Below this information is a table with columns for ID, Subject, and Unit. Each row has a corresponding checkbox labeled 'check1' through 'check5'. A 'Total' row shows a unit count of 12. At the bottom of the window are 'OK' and 'Cancel' buttons.

ID	Subject	Unit	
01023307	DESIGN AND ANALYSIS OF ALGORITHM	3	check1
01023309	TCP/IP NETWORKS	3	check2
01023305	LOCAL AND WIDE AREA NETWORKS	3	check3
01013311	DIGITAL COMMUNICATIONS	3	check4
			check5
Total		12	

รูปที่ 5-9 การลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2.3 การค้นหาข้อมูลในการลงทะเบียน และการเพิ่ม เปลี่ยน ถอน วิชาเรียน

การค้นหาข้อมูลที่ได้ลงทะเบียน แสดงดังรูปที่ 5-10 โดยในการที่จะค้นหาข้อมูลที่ได้ลงทะเบียนไปแล้ว ให้ใส่รหัสนักศึกษา แล้ว กดปุ่ม Show จะแสดงข้อมูลที่นักศึกษาได้ลงทะเบียนไว้ โดยนักศึกษาสามารถทำการเพิ่ม เปลี่ยน ถอนวิชาเรียนได้ โดยในการเพิ่ม ให้ทำการเช็คที่หน้าช่องที่จะกรอกรายวิชาแล้วทำการกรอกรหัสวิชาเพิ่มแล้วทำการกดปุ่ม ADD เมื่อทำการเปลี่ยนวิชาเรียน ให้เช็คหน้าช่องวิชาที่ต้องการเปลี่ยนแล้วทำการกรอกรหัสวิชาใหม่ที่ต้องการเปลี่ยน จากนั้น กดปุ่ม Change เมื่อต้องการลบให้ไปเช็คหน้าช่องวิชาที่ต้องการลบ เมื่อแน่ใจที่จะลบแล้วให้กดปุ่ม Delete

รูปที่ 5-10 การเพิ่ม เปลี่ยน ถอน การลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทดสอบที่ 2

จากการทดลองข้างต้นเป็นการทดลองในการเก็บข้อมูล โดยที่เป็นชนิดข้อมูลพื้นฐานทั่วไป สำหรับการทดสอบในตอนที่ 2 นี้ ได้ทำการทดสอบการเก็บและค้นหาข้อมูลประเภทรูปภาพจากตอนต้นได้กล่าวถึง แนวความคิดของฐานข้อมูลเชิงวัตถุและประโยชน์ของฐานข้อมูลเชิงวัตถุและวิธีในการหาความเหมือนของรูปภาพ ต่อไปจะเป็นการทดสอบโดยการนำเอาระบบฐานข้อมูลเชิงวัตถุมาประยุกต์ใช้งานกับระบบการจัดเก็บแล้วค้นหารูปภาพ โดยซอฟต์แวร์ Caché ในการจัดการระบบฐานข้อมูลเชิงวัตถุ

5.2.1 การทดสอบ

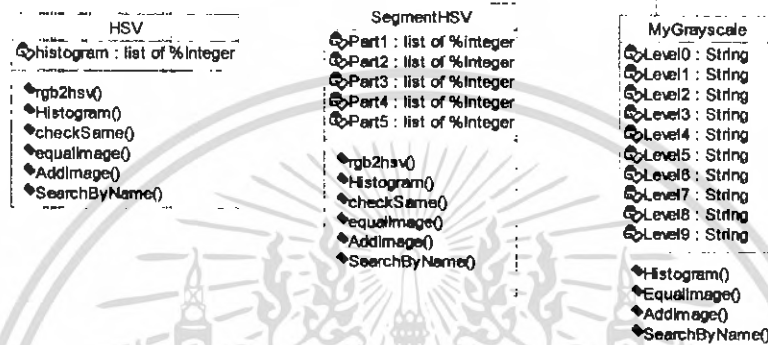
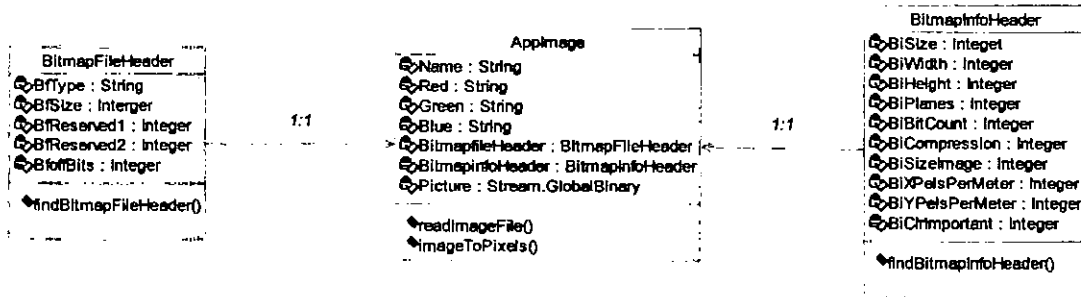
ระบบการจัดเก็บและค้นหารูปภาพ

1. ทำการจัดเก็บรูปภาพลงในฐานข้อมูล
2. ทำการจัดเก็บ BitmapFileHeader และ BitmapInfoHeader ของภาพแต่ละภาพ โดยสร้างเมทอดสำหรับหาค่าทั้งสอง และเก็บไว้ในฐานข้อมูล
3. ทำการสร้าง เมทอดเพื่อคำนวณหาค่า ฮิสโตแกรม และเก็บลงในฐานข้อมูล เพื่อใช้ในการค้นหาภาพที่มีความเหมือน
4. ทำการค้นหารูปภาพโดยใช้ ชื่อของรูปภาพในการค้นหา
5. ทำการสร้างเมทอด ในการค้นหาภาพที่มีความเหมือน เมทอด equalImage()
6. ทำการค้นหารูปภาพโดยใช้ รูปภาพในการค้นหา โดยสามารถหาภาพที่เหมือนกับภาพต้นฉบับ และค้นหาภาพที่มีความใกล้เคียงได้

5.2.2 วิธีทำการทดสอบ

5.2.2.1 ออกแบบคลาสไดอะแกรมของระบบการจัดเก็บและค้นหารูปภาพ

ได้ทำการออกแบบโดยใช้คลาสไดอะแกรมในการแสดงความสัมพันธ์ระหว่างคลาสต่าง ๆ จากนั้นจะนำคลาสไดอะแกรมที่ได้ออกแบบไว้มาใช้ในการทำฐานข้อมูล โดยข้อมูลที่ต้องการเก็บลงฐานข้อมูลให้กำหนดเป็นคุณสมบัติของคลาสนั้น ดังรูปที่ 5-11



รูปที่ 5-11 คลาสไดอะแกรมของระบบการจัดเก็บและการสืบค้นข้อมูลรูปภาพ

การทดลองได้ออกแบบคลาสไดอะแกรมจากโปรแกรม Rational Rose โดยมีรายละเอียดในแต่ละคลาสดังต่อไปนี้

- Class AppImage เป็นคลาสเกี่ยวกับรูปภาพโดยจะมีรายละเอียดรูปภาพที่จะนำมาจัดเก็บ คือ ชื่อของรูปภาพ, สีของภาพ, BitmapFileHeader, BitmapInfoHeader, และรูปภาพ
- Class BitmapFileHeader เป็นคลาสที่เก็บรายละเอียดของรูปภาพ โดยจะอยู่ในส่วนต้นของ files คือ ระบุประเภทไฟล์, ขนาดของไฟล์ในหน่วยไบต์, and และจำนวนไบต์ตั้งแต่เริ่มต้นไฟล์จนถึง bitmap data
- Class BitmapInfoHeader เป็นคลาสที่เก็บรายละเอียดของรูปภาพเป็นส่วนที่อยู่ถัดจาก BitmapFileHeader โดยเก็บรายละเอียด คือ ขนาดโครงสร้างของ BitmapInfoHeader , ความกว้างของภาพในหน่วยพิกเซล , ความยาวของภาพในหน่วยพิกเซล จำนวนบิตต่อพิกเซลของภาพ,ประเภทการบีบอัดข้อมูลที่ใช้ในภาพ ,ขนาดของภาพในหน่วยไบต์,จำนวนสีทั้งหมดในภาพ
- Class HSV เป็นคลาสเกี่ยวกับการหาค้นหารูปภาพโดยใช้วิธีฮิสโตแกรมอินเตอร์เซกชันซึ่งได้กล่าวมาในข้างต้น โดยจะเก็บ ค่าฮิสโตแกรมของรูปภาพที่ทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จัดเก็บ โดยจะมีรายละเอียดเกี่ยวกับรูปภาพ และมีเมทอดในการหาค่าฮิสโตแกรมของรูปภาพ และเมทอดที่ใช้ในการค้นหารูปภาพที่มีความเหมือน

- Class MyGrayscale เป็นคลาสเกี่ยวกับการค้นหารูปภาพของคน โดยจะมีรายละเอียดเกี่ยวกับรูปภาพ เมทอดที่ใช้ในการค้นหารูปภาพที่มีความเหมือน
- Class SegmentHSV เป็นคลาสเกี่ยวกับการค้นหารูปภาพ โดยก่อนที่จะทำการหาได้แบ่งรูปภาพออกเป็น 5 ส่วน แล้วใช้การค้นหารูปภาพโดยวิธีโดยใช้วิธีฮิสโตแกรมอินเตอร์เซกชันซึ่งได้กล่าวมาในข้างต้น โดยจะเก็บค่าฮิสโตแกรมทั้ง 5 ส่วนของรูปภาพที่ทำการจัดเก็บโดยจะมีรายละเอียดเกี่ยวกับรูปภาพ และมีเมทอดในการหาค่าฮิสโตแกรมของรูปภาพทั้ง 5 ส่วน และเมทอดที่ใช้ในการค้นหารูปภาพที่มีความเหมือน

เมื่อทำการออกแบบคลาสไดอะแกรมจาก Rational Rose แล้ว โปรแกรม Rational Rose จะมีเครื่องมือที่ใช้ในการ export คลาสไดอะแกรมไปเป็นคลาสที่ใช้ในการเก็บข้อมูลของฐานข้อมูล Caché โดยการทำการ export ไปที่ cache ได้ โดยเลือกที่ Tool -> Cache -> Export Classes to Cache ...

จากนั้นจะมีให้เลือกว่าจะ Export Class อะไรไปบ้าง โดยที่ไม่จำเป็นต้อง Export ไปทั้งหมด

หลังจากที่ทำการ Export มายัง Cache จะทำการสร้างโครงร่างของโค้ด โปรแกรมใน Cache ดังรูปที่ 5-12

```

Class MyImages.AppImage Extends %Persistent
{
Property Name As %String;
Property Red As %String;
Property Blue As %String;
Property Green As %String;
Property BitmapfileHeader As BitmapFileHeader;
Property BitmapinfoHeader As BitmapInfoHeader;

```

รูปที่ 5-12 แสดงตัวอย่างโครงร่างคำสั่ง

ซึ่งต้องมาใส่เพิ่มเติมในสิ่งที่ต้องการให้ฐานข้อมูลทำเพิ่มเติมจากที่สร้างให้ ดังรูปที่ 5-13

```

Class MyImages.AppImage Extends %Persistent
{
Property Name As %String;
Property Red As %String;
Property Blue As %String;
Property Green As %String;
Property BitmapfileHeader As BitmapFileHeader;
Property BitmapinfoHeader As BitmapInfoHeader;
Property Picture As %Stream.GlobalBinary(LOCATION = "^MyApp", STORAGE = "GLOBAL");
Projection MyTestImagr As %Projection.Java(ROCTDIR = "C:\Project");
}
Method readImageFile(fileName As %String) [ Language = java ]
{
Toolkit tk = Toolkit.getDefaultToolkit();
Image image = tk.getImage (fileName);
MediaTracker tracker = new MediaTracker (this);
tracker.addImage (image, 1);
try {
tracker.waitForID (1);
}
catch (InterruptedException e) {
System.out.println (e);
}
return image;
}
Method imageToPixels() [ Language = java ]
{
int numRows = image.getHeight (this);
int numCols = image.getWidth (this);
int[] oneDPixels = new int[numRows*numCols];

PixelGrabber grabber = new PixelGrabber(image, 0, 0, numCols, numRows,
oneDPixels ,0 ,numCols);
try {
grabber.grabPixels (0);
}
catch (InterruptedException e) {
System.out.println (e);
}
int[] pixels = new int[numRows][numCols][4];
for(int row = 0; row < numRows; row++){
int[] aRow = new int [numCols];
for(int col = 0; col < numCols; col++){
int element = row * numCols + col;
for(int col=0; col < numCols; col++){
pixels[row][col][0] = (aRow[col] >> 24) & 0xFF; // Alpha
pixels[row][col][1] = (aRow[col] >> 16) & 0xFF; // Red
pixels[row][col][2] = (aRow[col] >> 8) & 0xFF; // Green
pixels[row][col][3] = (aRow[col]) & 0xFF; // Blue
}
}
}
}
}

```

รูปที่ 5-13 แสดงตัวอย่างคำสั่งที่ใช้สร้างฐานข้อมูลเชิงวัตถุ

จากรูปที่ ได้ทำการสร้างเมธอด readImageFile(Filename As %String) ที่ทำการแปลงชื่อไฟล์ของรูปภาพไปเป็นรูปภาพ และเมธอด imageToPixels() จะทำการแปลงจาก Image เป็น Array ของพิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโค้ดโปรแกรมดังกล่าวเมื่อทำการ รันโปรแกรม cache จะทำการสร้างโค้ดภาษาจาวา เก็บไว้ใน “C:\Project” แสดงดังรูปที่ 5-14

```

public com.intersys.objects.CacheOutputStream getPictureOut() throws com.in
com.intersys.cache.Dataholder dh = mInternal.getProperty(ii_Picture,
jj_Picture,
com.intersys.objects.Database.RET_
"Picture");
com.intersys.cache.CacheObject cobj = dh.getCacheObject();
com.intersys.cache.SysDatabase db = mInternal.getDatabase();
return (db.getOutputStream (cobj));
)
private static int ii_Red = 11;
private static int jj_Red = 0;
private static int kk_Red = 11;
/**

```

รูปที่ 5-14 ตัวอย่างโค้ดโปรแกรมที่ cache สร้างเป็นภาษาจาวา

เมื่อได้โค้ดจาวาแล้วสามารถเขียนการทำงานในเมทอดเพิ่มเติมการทำงานได้ ซึ่งเมื่อคอมไพล์โค้ดจาวาดังกล่าวแล้วเราจะได้คลาสซึ่งเหมือนเป็นตัวแทนที่ใช้ในการติดต่อกับฐานข้อมูล ซึ่งเวลาจะจัดการกับฐานข้อมูลโดยจะทำการเปลี่ยนค่าคุณสมบัติใด ๆ จะกระทำผ่านเมทอดของออบเจกต์ที่สร้างขึ้นมาก็จะส่งผลกับข้อมูลในฐานข้อมูลด้วย ดังตัวอย่าง

ในส่วนของการทำงานติดต่อกับฐานข้อมูลจะใช้การติดต่อเช่นเดียวกับในระบบลงทะเบียนนักศึกษา โดยส่วนที่แตกต่างคือการเก็บและการดึงข้อมูลประเภทรูปภาพ ดังต่อไปนี้

5.2.2.2 การจัดเก็บรูปภาพ

1. ทำการสร้างการติดต่อกับฐานข้อมูล

```

String url = "jdbc:Cache://localhost:1972/USER";
String username = "SYSTEM";
String password = "SYS";
Database dbconnection =
CacheDatabase.getDatabase(url, username, password);

```

2. ทำการสร้างออบเจกต์ของข้อมูลรูปภาพ

```

MyImage.HSV pic = new MyImage.HSV(dbconnection);

```

3. ทำการหาค่าของฮิสโตแกรมของรูปภาพที่จะจัดเก็บลงในฐานข้อมูลนั้น เพื่อใช้ในการค้นหารูปภาพโดยใช้เมทอด Histogram() ซึ่งเป็นเมทอดการทำงานภายในออบเจกต์นั้น

```

MyImages.SegmentHSV.Histogram(image);

```

4. ทำการหาค่าของ BitmapFileHeader ของรูปภาพที่จะจัดเก็บลงในฐานข้อมูลนั้น

โดยใช้เมทอด findBitmapFileHeader() ซึ่งเป็นเมทอดการทำงานภายในคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BitmapFileHeader ซึ่งจะทำการนำค่า BitmapFileHeader ของรูปภาพนั้นมาสร้างเป็น
 ออบเจกต์ ซึ่งออบเจกต์ของคลาส BitmapFileHeader เป็นคุณสมบัติภายในคลาส
 รูปภาพ

```
MyImages.BitmapFileHeader fileHeader =
    MyImages.BitmapFileHeader.findBitmapFileHeader(
        dbconnection, image);
```

5. ทำการหาค่าของ BitmapInfoHeader ของรูปภาพที่จะจัดเก็บลงในฐานข้อมูลนั้น
 โดยใช้เมธอด findBitmapInfoHeader() ซึ่งเป็นเมธอดการทำงานภายใน
 คลาส BitmapInfoHeader ซึ่งจะทำการนำค่า BitmapInfoHeader ของรูปภาพนั้นมา
 สร้างเป็น ออบเจกต์ ซึ่งออบเจกต์ของคลาส BitmapInfoHeader เป็นคุณสมบัติภายใน
 คลาสรูปภาพ

```
MyImages.BitmapInfoHeader infoHeader =
    MyImages.BitmapInfoHeader.findBitmapInfoHeader(
        dbconnection, image);
```

6. ทำการจัดเก็บรูปภาพลงในฐานข้อมูล

```
//path ของไฟล์รูปที่จะทำการจัดเก็บลงในฐานข้อมูล
String inputBinaryFileName = pathFile;
//getPictureOut () เมธอดที่ cache สร้างให้สำหรับจัดเก็บรูปภาพลงในฐานข้อมูล
CacheOutputStream pl = pic.getPictureOut();

File inputBinaryFile = new File(inputBinaryFileName);
FileInputStream inputFileInputStream =
    new FileInputStream(inputBinaryFile);

//ทำการเขียน Stream ลงในฐานข้อมูล
byte[] binarybuffer=new
    byte[inputFileInputStream.available()];
int bytesRead = 0;

while ((bytesRead=inputFileInputStream.read(binarybuffer))
    !=-1){
    pl.write(binarybuffer);
}
```

7. เมื่อทำการหาค่าทั้ง 3 ค่า จากข้อ 2,3 และ4 จะนำค่าคุณสมบัติที่ได้นั้นมาจัดเก็บลง
 ในฐานข้อมูลของรูปภาพที่สร้างในข้อ 1

```
// เก็บค่าฮิสโตแกรมที่ได้จากข้อ 1
for(int i = 0;i<18;i++){
    pic.getPart1().add(i, Part1[i]);
}

// เก็บค่า BitmapFileHeader ที่ได้จากข้อ 3
pic.setBitmapinfoHeader (InfoHeader);
```

```
// เก็บค่า BitmapFileHeader ที่หาได้จากข้อ 3
pic.setBitmapfileHeader (fileHeader);
```

8. ทำการ SAVE ข้อมูลลงในฐานข้อมูล

```
pic._save();
dbconnection.close();
```

5.2.2.3 การค้นหารูปภาพ

- นำรูปภาพที่ต้องการค้นหา ผ่านเมธอด `equalImage` ซึ่งเป็นเมธอดภายในของคลาสที่เก็บรูปภาพนั้น โดยการทำงานภายในจะทำการหาค้นหาภาพที่อยู่ภายในฐานข้อมูลนั้น

```
imageicon = MyImages.HSV.equalImage(image1);
```

- เมื่อทำการเรียกเมธอดในข้อ 1 แล้วภายในเมธอดจะมีการทำงานดังนี้

2.1 ทำการหาฮิสโตแกรมของภาพ โมเดลที่ต้องการค้นหา

```
Histogram (image);
```

2.2 ทำการติดต่อกับฐานข้อมูล

```
String url = "jdbc:Driver://localhost:1972/USEF";
String username = "SYS@1";
String password = "SYS";
Database dbconnection =
CacheDatabase.getDatabase(url, username, password);
```

2.3 การเรียกออบเจกต์ของรูปภาพที่เก็บอยู่ภายในฐานข้อมูล

```
MyImages.SegmentHSV test = (MyImages.SegmentHSV)
MyImages.SegmentHSV._open(dbconnection, oid);
```

- นำค่าฮิสโตแกรมที่ได้จากข้อ 2.1 มาทำฮิสโตแกรมอินเตอร์เซกชันกับฮิสโตแกรมของรูปภาพแต่ละรูปที่เก็บอยู่ภายในฐานข้อมูล โดยมีเมธอด `checkSame` ที่ทำงานโดยการตรวจสอบความเหมือนของคู่ฮิสโตแกรมตามทฤษฎีที่กล่าวมาในตอนต้นจะได้ค่าความเหมือนของรูปที่ต้องการหาก็กับรูปที่เก็บอยู่ในฐานข้อมูล

```
// คำนวณอัตราความเหมือนของภาพที่เป็น Model กับภาพที่อยู่ในฐานข้อมูลแต่ละค่า
double checkequal =
checkSame(MHistogram, test.getHistogramStr1(), numpixel);
```

```
// คิดเป็นเปอร์เซ็นต์
```

```
per = checkequal * 100;
```

2.5 นำค่าเปอร์เซ็นต์ที่คำนวณได้และรูปภาพจากฐานข้อมูลมาเรียงลำดับจากมาก

ไปหาน้อย แล้วเก็บลงใน Array เพื่อแสดงผล

```
for(int j=0;j<6;j++){
    if(per>percents[j]){
        for(int k=5;k>j;k--){
            percents[k]=percents[k-1];
            imageout[k] = imageout[k-1];
        }
        percents[j]=per;
        imageout[j]= new ImageIcon(binarybuffer);
        j = 6;
    }
}
```

โดยการนำข้อมูลรูปภาพออกจากฐานข้อมูลสามารถทำได้ดังนี้

```
MyImage.HSV test=(MyImage.HSV)MyImage.HSV._open(
    dbconnection,oid);
//เมธอด getPictureIn() เป็นเมธอดที่ cache สร้างไว้สำหรับดึงรูปออกจาก
ฐานข้อมูล
com.intersys.objects.CacheInputStream p2 =
    test.getPictureIn();
byte[] binarybuffer = new byte[p2.available()];
int position;
for(position = 1;position <= p2.available() ; position+=
    p2.available() ) {
    bytesRead = p2.read(binarybuffer);
}
```

2.6 เมื่อทำการอินเทอร์เซกชันครบรูปภาพแล้ว จะทำการปิดการเชื่อมต่อกับ

ฐานข้อมูล

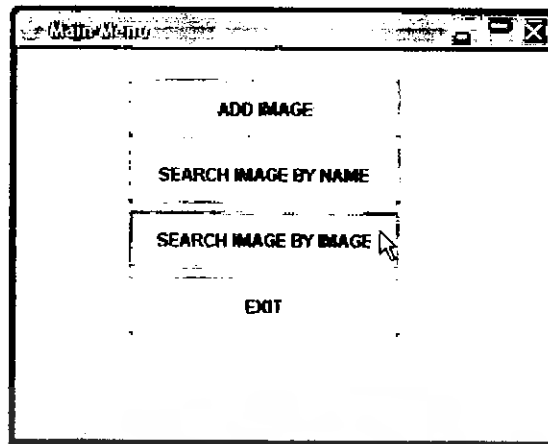
```
dbconnection.closeObject(test.getOref());
dbconnection.close();
```

วิธีการเปิดออบเจกต์ที่มีอยู่แล้วในฐานข้อมูล และการใช้ Class Query ทำเช่นเดียวกับในระบบลงทะเบียนนักศึกษาข้างต้น

5.2.3 ผลจากการทดลองการทดสอบที่ 2

5.2.3.1 หน้าจอสำหรับการเลือกการทำงาน แสดงดังรูปที่ 5-15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

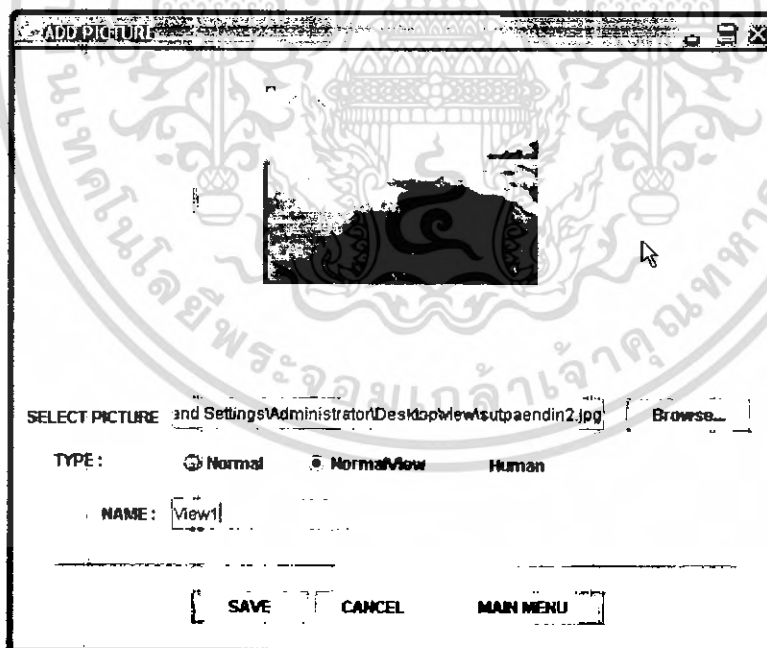


รูปที่ 5-15 หน้าจอสำหรับการเลือกการทำงาน

5.2.3.2 การจัดเก็บข้อมูลรูปภาพลงในฐานข้อมูล

หน้าจอสำหรับการจัดเก็บรูปภาพลงในฐานข้อมูล โดยทำการเลือกรูปภาพที่จะจัดเก็บลงในฐานข้อมูล, ประเภทของรูปภาพและกรอกชื่อของรูปภาพลงในแบบฟอร์มที่กำหนด

เมื่อกดปุ่ม SAVE โปรแกรมก็จะนำรูปภาพมาหาค่าฮีดรแกรม, ค่า BitmapFileHeader และค่า BitmapInfoHeader จากนั้นจะนำรูปภาพ และค่าที่ได้ รวมถึง ข้อมูลที่กรอก เก็บลงในฐานข้อมูล Caché ตามรูปที่ 5-16



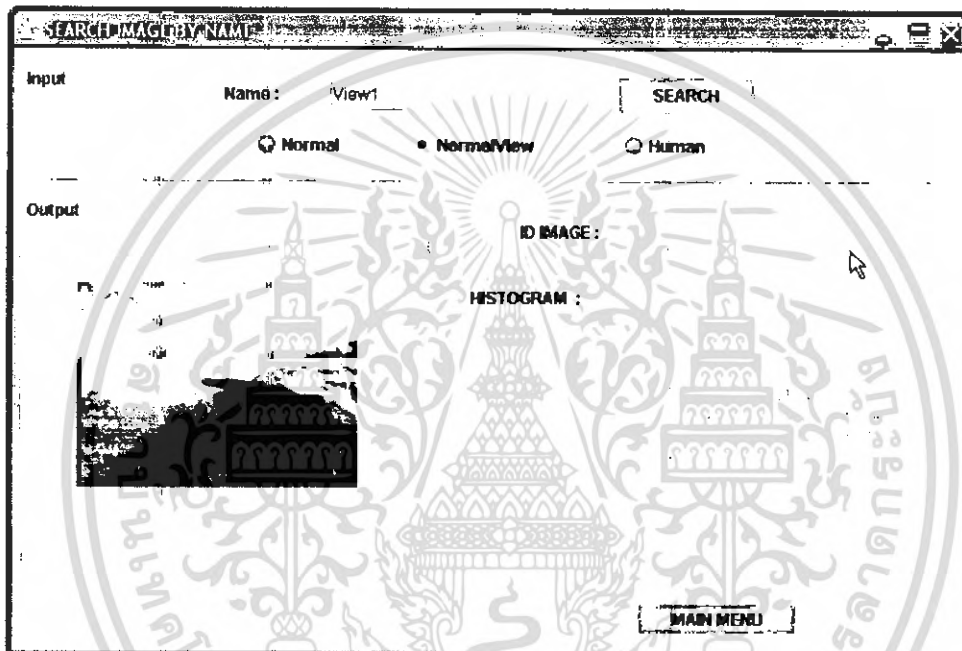
รูปที่ 5-16 การจัดเก็บข้อมูลรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

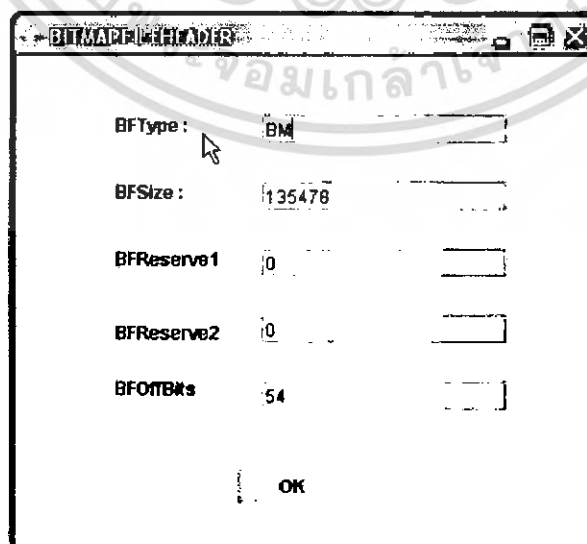
5.2.3.3 การค้นหารูปภาพจากชื่อของรูปภาพ

หน้าจอสําหรับการค้นหารูปภาพในฐานข้อมูลจากชื่อของรูปภาพแสดงดังรูปโดยใน ส่วนของอินพุตข้อมูลจะทำการอินพุต ชื่อของภาพ และประเภทของภาพในฐานข้อมูล ดังในรูปที่ 5-17

เมื่อกดปุ่ม SEARCH โปรแกรมก็จะนำชื่อของรูปภาพและประเภทของรูปภาพมาทำการค้นหาในฐานข้อมูลถ้าพบก็จะดึงรูปภาพออกจากฐานข้อมูลขึ้นมาแสดงผล พร้อมทั้งค่าฮิสโตแกรมของรูปนั้น และ ค่า BitmapFileHeader , BitmapInfoHeader ดังรูปที่ 5-18 และ 5-19 ตามลำดับ



รูปที่ 5-17 การค้นหารูปภาพจากชื่อและประเภทของรูปภาพ



รูป 5-18 ค่า BitmapFileHeader ที่ได้จากการ Search

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Field	Value
BiSize :	40
BiWidth :	245
BiHeight :	184
BiPlanes	24
BiBitCount	1
BiCompressi...	0
BiSizeImage	135424
BiXPelsPerMe...	0
BiYPelsMeter	0
BiClrUsed :	0
BiClrImportant :	0

OK

รูปที่ 5-19 ค่า *BitmapInfoHeader* ที่ได้จากการ Search

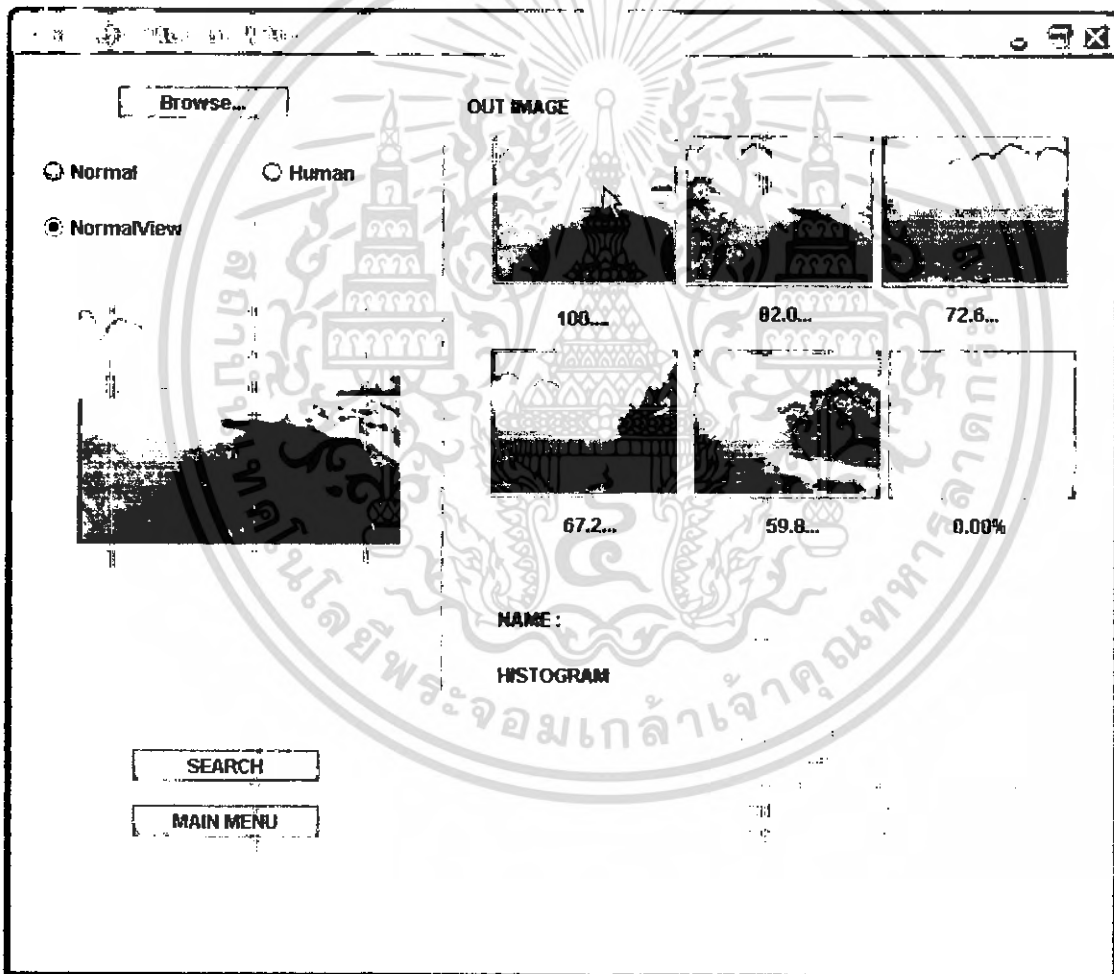
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3.4 การค้นหารูปภาพจากรูปภาพ

หน้าจอสําหรับการค้นหารูปภาพในฐานข้อมูลจากรูปภาพแสดงดังรูป โดยในส่วน ของอินพุตข้อมูลจะทำการอินพุท รูปภาพที่ต้องการค้นหา และประเภทของภาพในฐานข้อมูล ดังใน รูป

เมื่อกดปุ่ม SEARCH โปรแกรมก็จะนำรูปภาพมาทำฮิสโตแกรมอินเตอร์เซกชันกับ รูปภาพในประเภทที่ระบุมาในคอนอินพุทภายในฐานข้อมูล โดยโปรแกรมจะดึงรูปภาพที่มีความ เหมือนมากที่สุด 6 ลำดับ ขึ้นมาแสดงผลโดยเรียงจากความเหมือนมากไปยังน้อย โดยเราสามารถ เรียกดูค่า ฮิสโตแกรมของรูปภาพต่าง ๆ ได้โดยการคลิกที่รูปภาพที่ต้องการดู ดังแสดงผล ดังรูปที่

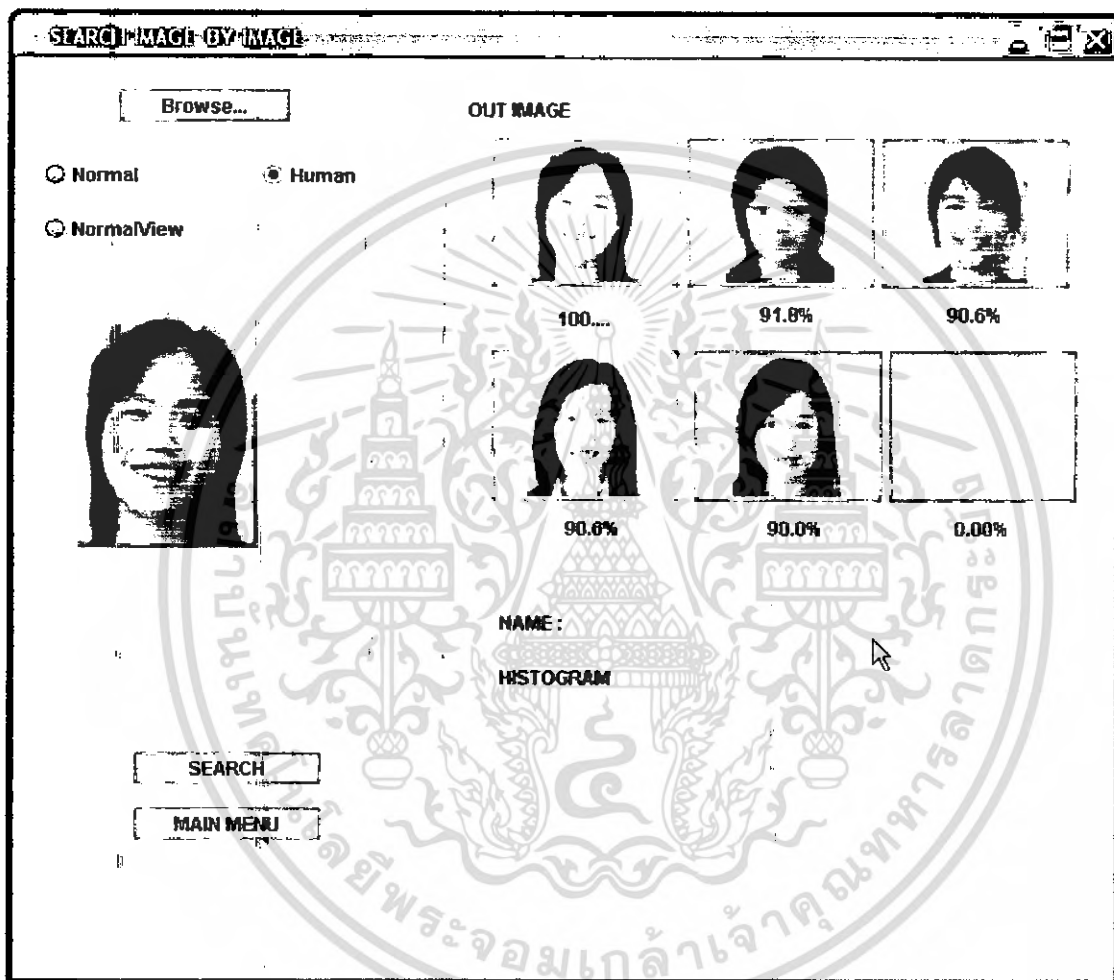
5-20



รูปที่ 5-20 การค้นหารูปภาพจากรูปภาพ (ภาพวิว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกวิธีหนึ่งของการค้นหารูปภาพในโครงการนี้ คือ ภาพใบหน้าคน แสดงผลดังรูปที่ 5-21 โดยเมื่อกดปุ่ม SEARCH โปรแกรมก็จะนำรูปภาพมาทำการประมวลผลตามทฤษฎีข้างต้น ในแบบที่ 2 ตามประเภทที่ระบุมาในตอนอินพุตภายในฐานข้อมูล โดยโปรแกรมจะดึงรูปภาพที่มีความเหมือนมากที่สุด 6 ลำดับ ขึ้นมาแสดงผลโดยเรียงจากความเหมือนมากไปยังน้อย โดยเราสามารถเรียกดูค่า ของรูปภาพต่าง ๆ ได้โดยการคลิกที่รูปภาพที่ต้องการดู



รูปที่ 5-21 การค้นหารูปภาพจากรูปภาพ (ภาพใบหน้าคน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

สรุป

6.1 สรุปผลการทดลอง

ในการทดลองที่ 1 ได้ทำการออกแบบฐานข้อมูลตามแนวคิดเชิงวัตถุเพื่อทดสอบว่าฐานข้อมูลเชิงวัตถุสามารถทำงานตามแนวคิดเชิงวัตถุได้ ผลปรากฏว่า

6.1.1 มีคุณสมบัติของการสืบทอด (Inheritance) โดยที่คลาส Master เป็น ชั้นคลาส ของคลาส Student คลาส Master จึงสืบทอดคุณสมบัติทั้งหมดของคลาส Student มา ทั้งคุณสมบัติแล้วเมธอด

ตัวอย่างเช่น

```
User.Student stu = new User.Student(dbconnection) ;
Stu.setSId("46010862");
Stu.setSName("Sumalee");
Stu.setSSurname("Saehear");
```

```
User.Master mas = new User.Master(dbconnection);
mas.setSId("461010862");
mas.setSName("Sureeporn");
mas.setSSurname("Soysangwan");
mas.setLastUniversity("KMITL");
```

จะเห็นว่า คลาส Master สืบทอดมาจากคลาส Student จากคุณสมบัติการสืบทอด จะสืบทอดเมธอด setSId(), setSname(), setSSurname() มาจากคลาส Student แล้วสามารถเพิ่มคุณสมบัติ setLastUniverty() ซึ่งเป็นคุณสมบัติพิเศษของคลาส Master

6.1.2 สามารถทำโพลิมอร์ฟิซึม (Polymorphism)ได้ จากการทดลองแต่ละคลาสมี query ซึ่ง"Byid" เหมือนกัน โดยเมื่อแต่ละคลาสใช้ query ชื่อเดียวกัน จะมีการทำงานที่ต่าง ๆ กัน ขึ้นกับการทำงานของออบเจกต์แต่ละคลาส จากตัวอย่างในข้อ 6.1.1

```
CacheQuery cq = new CacheQuery(dbconnection, "User.Student","Byid");
ResultSet rs = cq.execute("01010101");
while (rs.next()){
    int i = Integer.parseInt(rs.getString(1));
    Oid oid = new Oid(i);
    User.Student test = (User.Student) User.Student._open(
        dbconnection, oid);
    System.    .println(test.getSName());
```

```
CacheQuery cq = new CacheQuery(dbconnection, "User.Master","Byid");
ResultSet rs = cq.execute("Sureeporn");
while (rs.next()){
    int i = Integer.parseInt(rs.getString(1));
    Oid oid = new Oid(i);
    User.Master test = (User.Master)User.Master._open(
        dbconnection, oid);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
System.out.println(test.getSName());
```

จากตัวอย่างทั้งสอง คลาส Master ได้ทำการ Inherit มาจากคลาส Student ซึ่งมีเมธอดการ Query ชื่อ "Byid" เหมือนกัน โดยแต่ละเมธอดทำงานต่างกัน โดย คลาส Student จะทำการ Query จากรหัสนักศึกษา และคลาส Master จะทำการ Query จากชื่อนักศึกษา เมื่อทำการทดสอบพบว่าเมื่อชื่อเมธอดเดียวกันถูกเรียกใช้งานจากออบเจ็กต์ที่ต่างกันจะมีผลให้การทำงานต่างกัน ตามคุณสมบัติพอลิมอร์ฟิซึม

6.1.3 มีคุณสมบัติเอนแคปซูเลชัน (Encapsulation) โดยฐานข้อมูลเชิงวัตถุ ไม่สามารถแก้ไขค่าคุณสมบัติ (Property) ได้โดยตรงการทำงานแก้ไขค่าต้องกระทำผ่านเมธอดที่แต่ละคลาสนั้นระบุไว้เท่านั้น

ตัวอย่างเช่น

```
User.Course course1 = new User.Course(dbconnection);
course1.setCId("01010102");
course1.setCName("Continue");
course1.setTotalCrd(141);
course1.setTotalYear(3);
```

จากตัวอย่างพบว่าการทำงานที่จะกำหนดหรือแก้ไขค่า คุณสมบัติ (Property) ทำได้โดยกระทำผ่านทางเมธอดเท่านั้น

6.1.4 การทดลองพบว่า caché ไม่มีคุณสมบัติความสัมพันธ์แบบ many-to-many

6.1.5 การใช้ Object Access ติดต่อกับฐานข้อมูลเชิงวัตถุ (Caché) ทำให้แอปพลิเคชันที่สร้างติดต่อกับฐานข้อมูลได้ง่าย เนื่องจากใช้หลักการของออบเจ็กต์ในเรื่องการเอนแคปซูเลชันโดยผู้ใช้งานไม่รู้รายละเอียดการทำงาน เช่น เวลา query จะทำการใส่ คำสั่ง SQL ลงในเมธอด ทำให้ผู้ใช้แอปพลิเคชันไม่จำเป็นต้องรู้คำสั่ง SQL รู้เพียงว่าเมธอดนี้ทำงานอะไรก็พอ

หลังจากทำการทดลองที่ 1 แล้ว ได้ทำการศึกษาต่อถึงความก้าวหน้าของฐานข้อมูลเชิงวัตถุพบว่าฐานข้อมูลเชิงวัตถุมีความเหมาะสมกว่าฐานข้อมูลเชิงความสัมพันธ์เมื่อต้องการเก็บข้อมูลที่มีความซับซ้อนสูง เช่น รูปภาพ วิดีโอ เสียง จากการศึกษาได้ทำการทดลองโดยการสร้างระบบการจัดเก็บและค้นหารูปภาพโดยใช้ฐานข้อมูลเชิงวัตถุ cache พบว่า

ระบบฐานข้อมูลเชิงวัตถุรองรับชนิดข้อมูลที่มีความซับซ้อน ในโครงการนี้ คือ รูปภาพ โดยแต่ละรูปภาพจะมี OID ของแต่ละรูป โดยวัตถุแต่ละวัตถุจะมีค่าไม่เหมือนกันโดยฐานข้อมูลเชิงวัตถุจะสร้างเมธอดที่จะทำการจัดเก็บและดึงข้อมูลประเภทรูปภาพ รวมทั้งยังสนับสนุนชนิดข้อมูล อาร์เรย์ ลิสต์ ทำให้วัตถุหนึ่ง สามารถเก็บข้อมูลต่างๆ ไว้ในคุณสมบัติชื่อเดียวกัน โดยการใช้ลักษณะของลิสต์หรือ อาร์เรย์

ฐานข้อมูลเชิงวัตถุมีเมธอดที่ใช้ในการกระทำกับรูปภาพที่เก็บอยู่ภายในฐานข้อมูลซึ่งเมื่อผู้พัฒนาแอปพลิเคชันนำฐานข้อมูลนี้ไปใช้งาน เมื่อต้องการกระทำอะไรกับรูปภาพจะกระทำผ่านเมธอดที่สร้างไว้ในฐานข้อมูล โดยผู้พัฒนาแอปพลิเคชันไม่ต้องรู้กระบวนการเกี่ยวกับ Image

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Processing เพียงแต่รู้ว่าเมธอดนี้ให้ผลการทำงานอย่างไรก็สามารถใช้งานได้ อีกทั้งจากคุณสมบัติ การสืบทอด ทำให้เมื่อผู้พัฒนารฐานข้อมูลเชิงวัตถุทำการพัฒนารฐานข้อมูลขึ้นมาใหม่ สามารถสืบ ทอดคุณสมบัติของฐานข้อมูลเดิมที่มีอยู่แล้ว ทำให้ลดเวลา ในการออกแบบฐานข้อมูล

6.2 แนวทางการพัฒนาต่อ

6.2.1 พัฒนาให้ระบบสามารถทำการแยกประเภทของรูปภาพก่อนที่จะเก็บลงฐานข้อมูล

6.2.2 ทำการสร้างอินเด็กซ์ที่ใช้กับรูปภาพสำหรับฐานข้อมูลเชิงวัตถุ เพื่อประสิทธิภาพใน การค้นหาที่ดีขึ้น

6.2.3 พัฒนากับข้อมูลประเภทอื่นที่มีความซับซ้อนมากขึ้น เช่น วีดีโอ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Answers Corporation . “object-database”
<http://www.answers.com/topic/object-database>
- [2] InterSystems Corporation . “Caché” [Online].
Available :<http://www.intersystems.com/>.
- [3] Mark Priestley, 2004. PRACTICAL OBJECT-ORIENTED DESIGN WITH UML Second Edition , New York , The McGraw-Hill Companies
- [4] Sixth International World Wide Web Conference. “Image Processing Tool on the WWW”
<http://www.ambuehler.ethz.ch/CDstore/www6/Posters/701/poster701.html>
- [5] Sun Microsystems, Inc.” How to Use File Choosers”
<http://java.sun.com/docs/books/tutorial/uiswing/components/filechooser.html>
- [6] กิตติ ภัคดีวัฒนกุล , กิตติพงษ์ กลมกล่อม , 2548 คัมภีร์ การวิเคราะห์และออกแบบระบบเชิงวัตถุด้วย UML กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์
- [7] ธิดา เค่นพฤษธรรม , อาทิตย์ วงษ์สมบูรณ์กุล “ระบบสืบค้นข้อมูลโดยใช้รูปภาพ” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2544
- [8] ประชุมพร หงสกุล , โสภณ มงคลลักษณ์ “การกำหนดดัชนีและการเรียกภาพนิ่งโดยใช้สีเป็นพื้นฐาน” ” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2543
- [9] สัจพร ชื่นพิทยาร , สุวิทย์ สิทธิธรรมชารี “ระบบสารสนเทศแผนที่เชิงเวลา” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2544

ภาคผนวก ก.

การติดตั้งและใช้งานฐานข้อมูล Caché

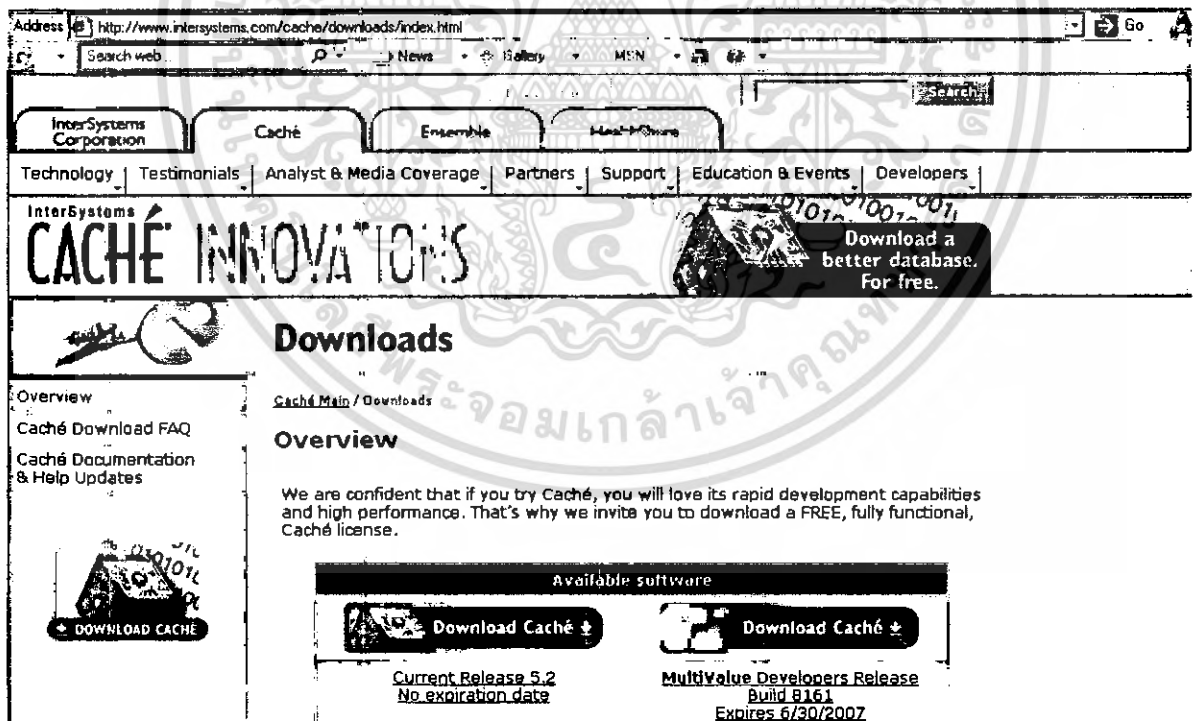
การดาวน์โหลดฐานข้อมูล Caché และเอกสารคู่มือการใช้งาน

ทำการดาวน์โหลด ฐานข้อมูล Caché จากเว็บไซต์ www.intersystems.com ซึ่งทำการดาวน์โหลดได้จากเมนู Developers->Downloads ดังรูปที่ ก-1



รูปที่ ก-1 แสดงเมนูการเลือกดาวน์โหลดระบบฐานข้อมูล Cache

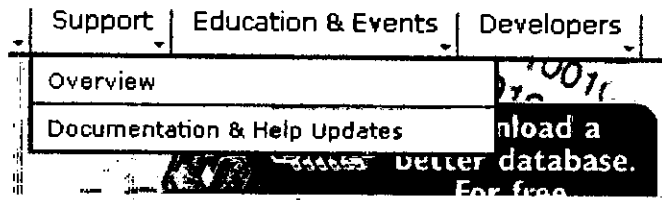
เมื่อทำการเลือกแล้ว จะเข้าสู่เว็บเพจของการดาวน์โหลดฐานข้อมูล Caché ดังรูปที่ ก-2



รูปที่ ก-2 แสดงเว็บเพจการดาวน์โหลดฐานข้อมูล Cache

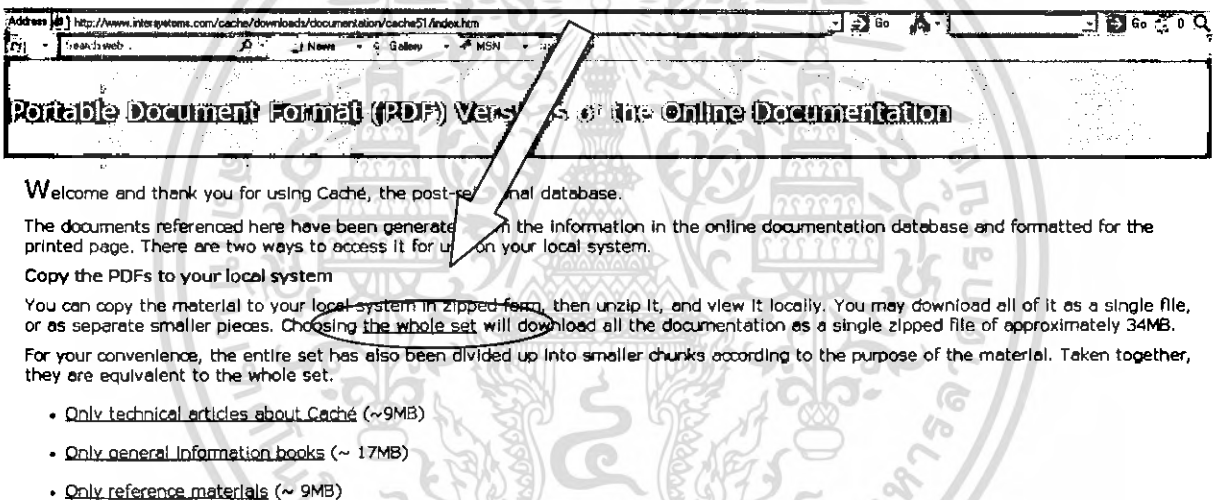
สำหรับเอกสารทุกอย่างเกี่ยวกับฐานข้อมูล Caché ที่ทำการดาวน์โหลดมาแล้วนั้น สามารถทำการดาวน์โหลดไฟล์ pdf ได้จากการเลือกที่เมนู Support->Documentation & Help Updates ดังเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก-3 ซึ่งเมื่อเข้าไปแล้ว จะสามารถเลือกดาวน์โหลดได้ตามรุ่นของฐานข้อมูลที่ทำการโหลดมาในตอนแรก



รูปที่ ก-3 แสดงการเข้าถึงเมนูเพื่อการดาวน์โหลดเอกสารคู่มือการใช้งาน

เมื่อทำการเลือกเมนูแล้วจะเข้าสู่เว็บเพจที่จะทำการดาวน์โหลดเอกสารซึ่งมีให้เลือกหลายแบบขึ้นกับความต้องการในการใช้งานของเรา แต่ในที่นี้จะทำการดาวน์โหลดเอกสารทุกอย่างที่เกี่ยวข้องกับรุ่นที่ดาวน์โหลดมาในตอนแรก โดยคลิกที่ ดังรูปที่ ก-4



รูปที่ ก-4 แสดงหน้าเว็บเพจการดาวน์โหลดเอกสารคู่มือการใช้งาน

การ Install ระบบฐานข้อมูล Caché

เมื่อทำการดาวน์โหลดฐานข้อมูล Caché เสร็จเรียบร้อยแล้ว จะได้ไฟล์ดังรูปที่ ก-5

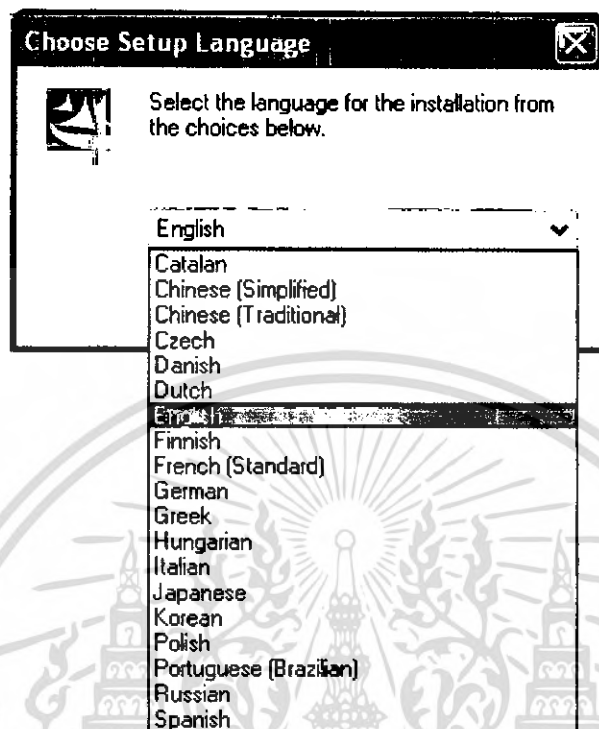


CachePckit.exe

รูปที่ ก-5 ไฟล์ Install ฐานข้อมูล Cache

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

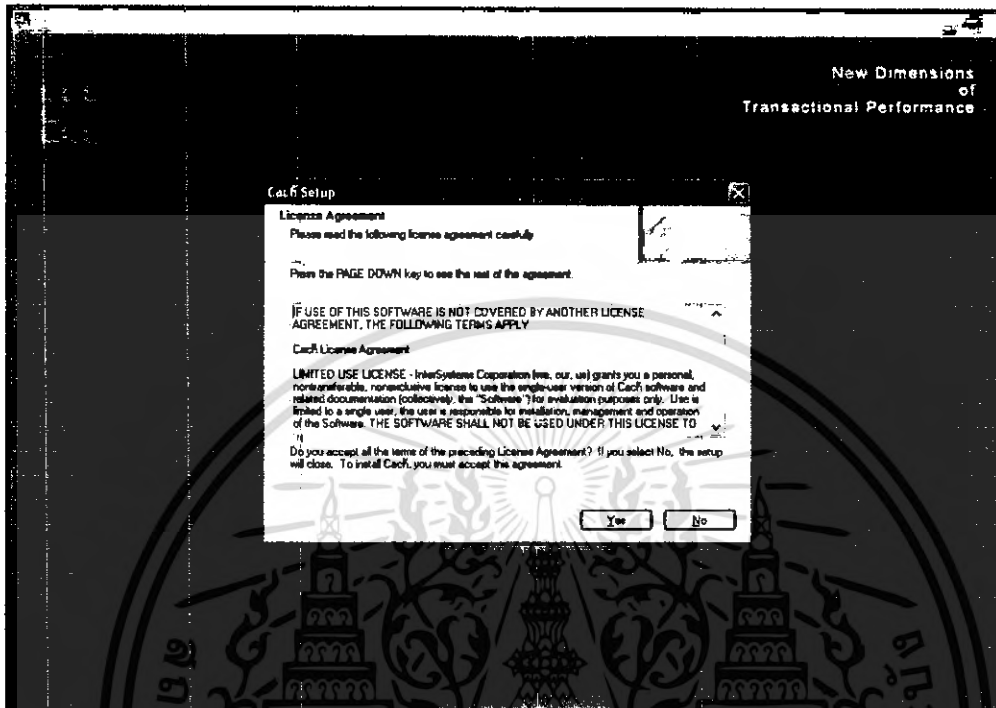
เมื่อดับเบิลคลิกที่ตัวไฟล์ จะขึ้นไดอะล็อกให้เลือกภาษา ดังรูปที่ ก-6 ซึ่งจะเห็นว่ามิให้
เลือกหลายภาษาด้วยกัน



รูปที่ ก-6 ไดอะล็อกของการเลือกภาษาสำหรับการ *Install* ฐานข้อมูล *Cache*

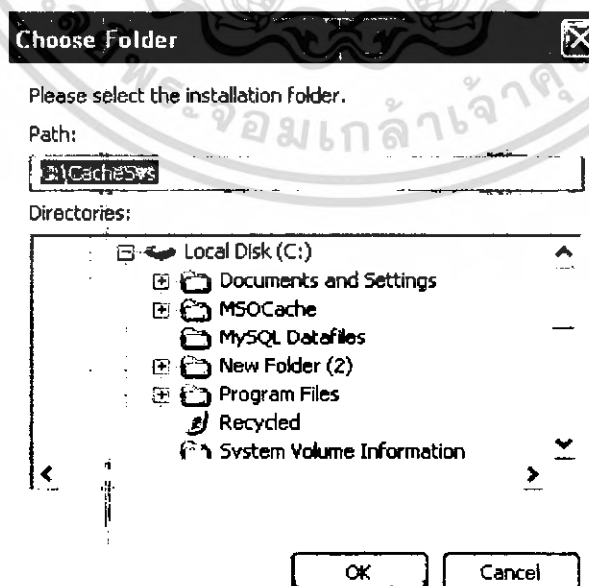
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกภาษาเรียบร้อยแล้ว จะเข้าสู่การ Install ฐานข้อมูล Caché ดังรูปที่ ก-7 ซึ่งคล้าย ๆ กับการลงโปรแกรมทั่ว ๆ ไป โดยให้ปฏิบัติตามคำสั่งในแต่ละขั้นตอน



รูปที่ ก-7 แสดงการเข้าสู่การ Install ฐานข้อมูล Caché

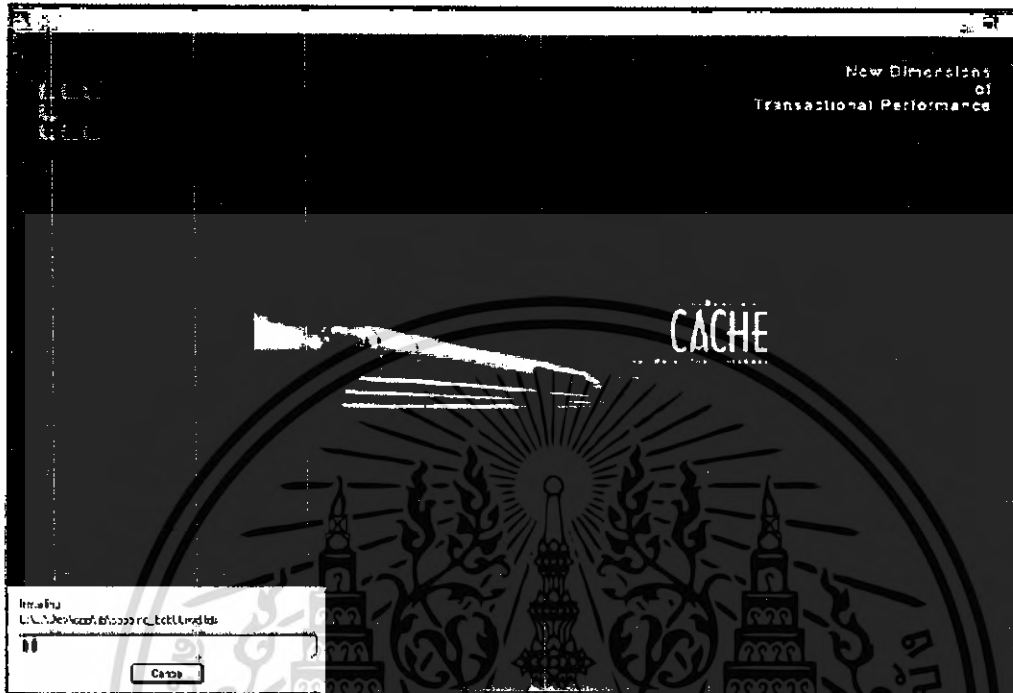
จากรูปที่ ก-8 เป็นไดอะล็อกในการเลือก path ในการ Install Caché ซึ่งดีฟอลต์ของระบบ จะอยู่ใน C:\CacheSys



รูปที่ ก-8 แสดงดีฟอลต์ path ในการ Install Caché

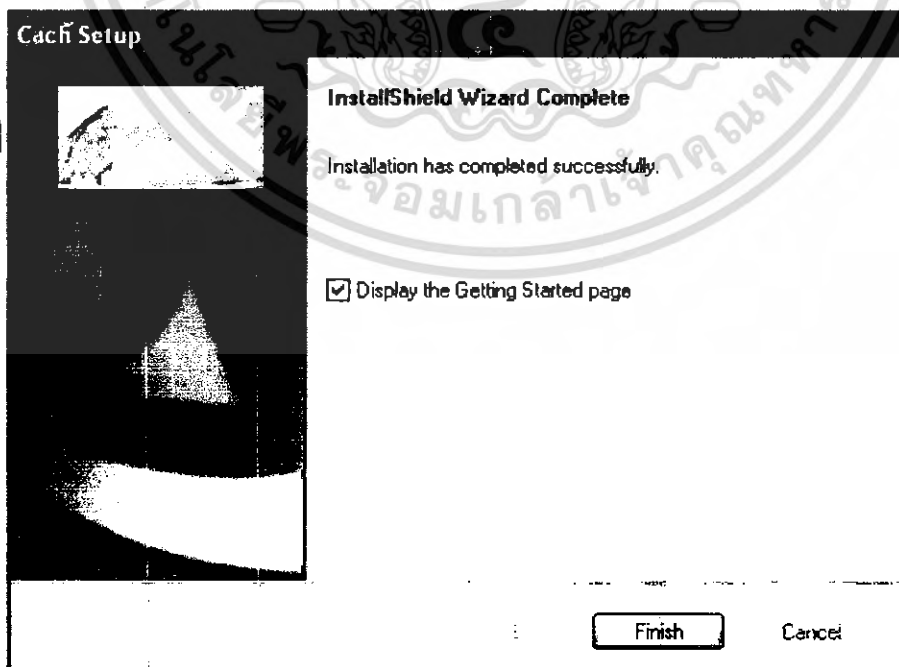
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปฏิบัติตามคำสั่งขั้นตอนในการ Install ไปเรื่อย ๆ โดยเลือกทุกอย่างตามดีฟอลต์ของระบบ จากนั้นรอสักครู่เพื่อให้ระบบทำการลง Caché ดังรูปที่ ก-9



รูปที่ ก-9 ระบบกำลังทำการ Install

เมื่อเสร็จสิ้นการ Install คลิก Finish ดังรูปที่ ก-10



รูปที่ ก-10 แสดงการ Install เสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นระบบจะขึ้นเว็บเพจคู่มือการใช้งานระบบฐานข้อมูล Caché ให้อย่างอัตโนมัติ ดังรูปที่ ก-11

InterSystems Online Documentation - Mozilla Firefox

File Edit View Go Database Tools Help

http://127.0.0.1:8772/html/inter.sys/doc/inter/Cache/Cache%20Getting%20Started%20with%20Cache.html?DCFGID=000001000011196000000000

Cache Sated | latest releases

Getting Started with Caché

The material listed below will help you to get started using InterSystems products. You can also go to the [Documentation Home Page](#) using this button:

[Home](#)

For New Caché Users

- [Introduction to Caché](#)—An overview of Caché architecture and features.
- [Building Web Applications with Caché](#)—A tutorial that takes you through the steps of building an object-based Web application using Caché.
- [Basic System Administration](#)—A guide to administering your Caché System.
- [Other Tutorial Offerings](#)—A map of the other tutorial offerings for self-paced learning.

For Experienced Caché Users

- [Cache Release Notes](#)—A listing of the new features in this release, as well as any upgrade issues.
- [Using Caché Studio](#)—Documentation for Caché Studio.
- [Cache Security Guide](#)—A description of the new security features available with this version.

For Additional Information

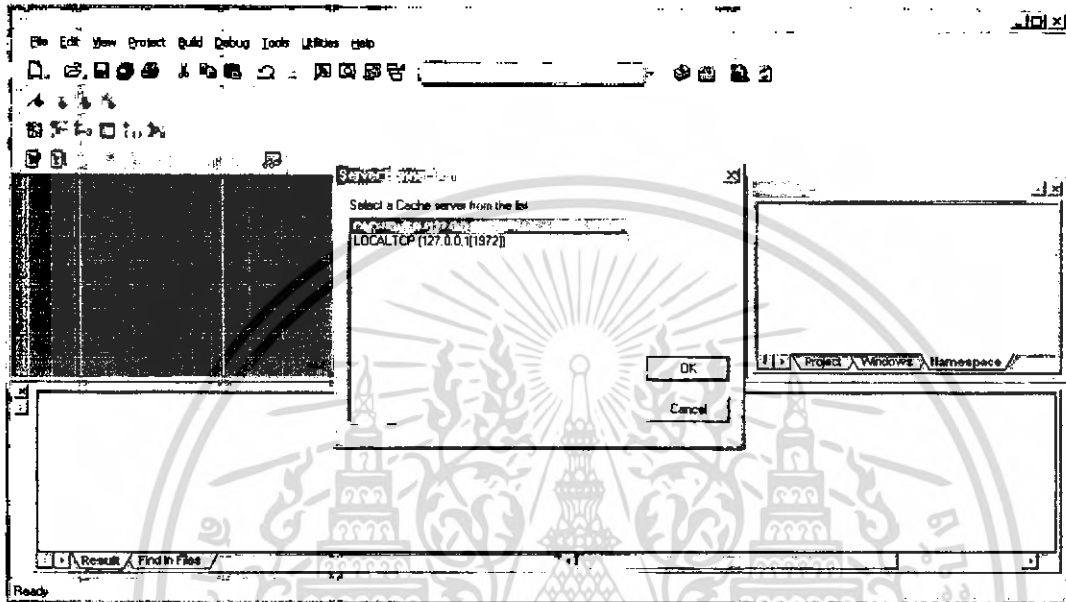
รูปที่ ก-11 เว็บเพจคู่มือการใช้งาน Caché

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการใช้งานโปรแกรม Cache Studio

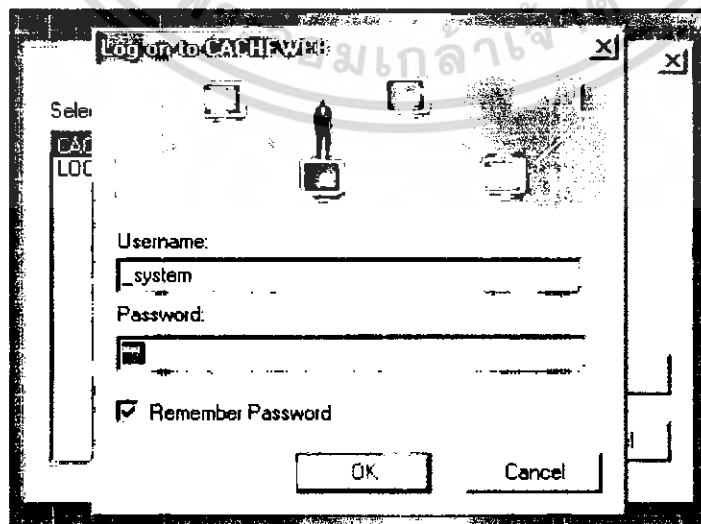
เริ่มต้นการใช้งาน

เข้าสู่โปรแกรมโดยเลือกที่ โปรแกรม Caché->CACHEWEB->Studio จากนั้นจะเข้าสู่โปรแกรมดังรูปที่ ก-12 และในไดอะล็อก Server Connection เลือก CACHEWEB แล้วคลิกที่ OK



รูปที่ ก-12 แสดงหน้าต่างโปรแกรม Cache Studio

จากนั้นจะขึ้นไดอะล็อกให้ทำการ Log on โดยคีย์พอลต์ของระบบ Username คือ _system และ Password คือ SYS ซึ่งหาข้อมูลได้จากเอกสารคู่มือการใช้งานที่ดาวน์โหลดมา แล้วคลิก OK ดังรูปที่ ก-13

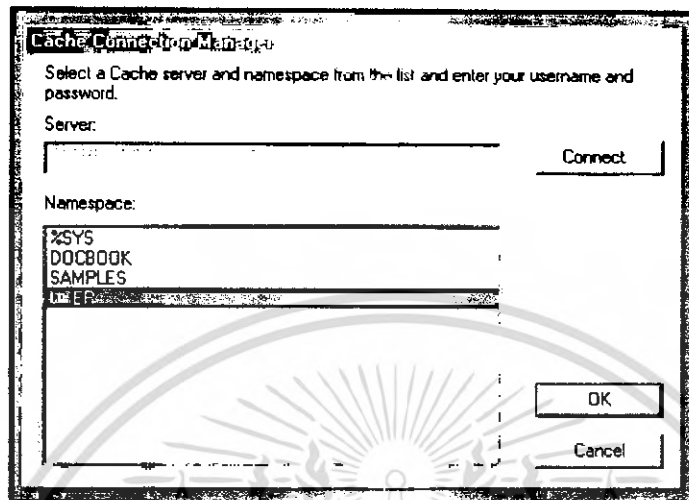


รูปที่ ก-13 แสดงไดอะล็อกของการ Log on เข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

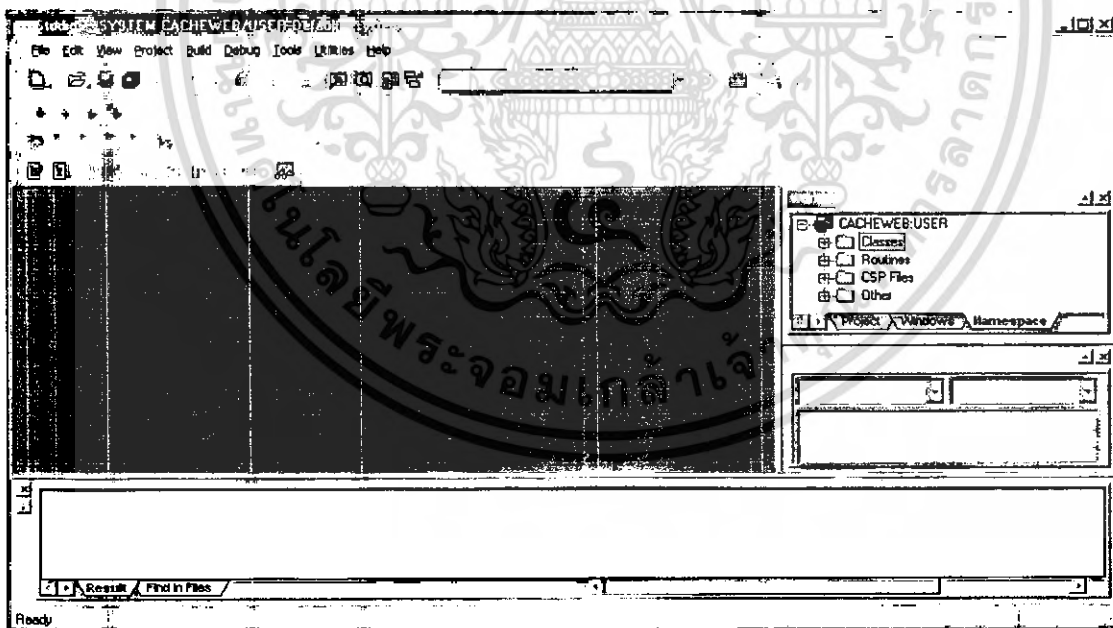
จากนั้นจะให้ทำการเลือกเข้าสู่ Namespace ซึ่งในที่นี้จะเลือก USER แล้วคลิก OK ดังรูปที่

ก-14



รูปที่ ก-14 แสดงไดอะล็อกการเลือก Namespace

จากนั้นจะเข้าสู่การใช้งานโปรแกรมดังรูปที่ ก-15

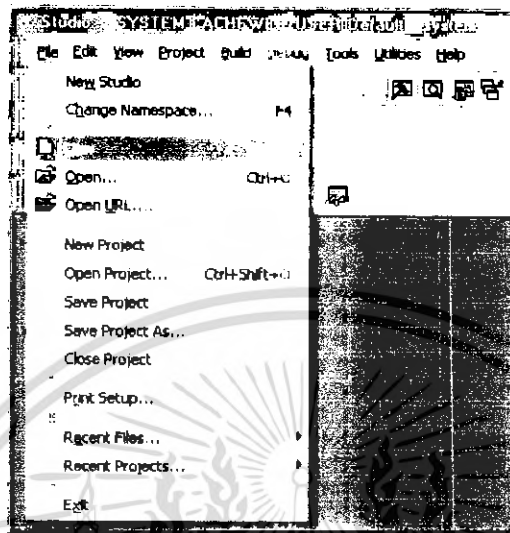


รูปที่ ก-15 แสดงหน้าของโปรแกรมที่พร้อมใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

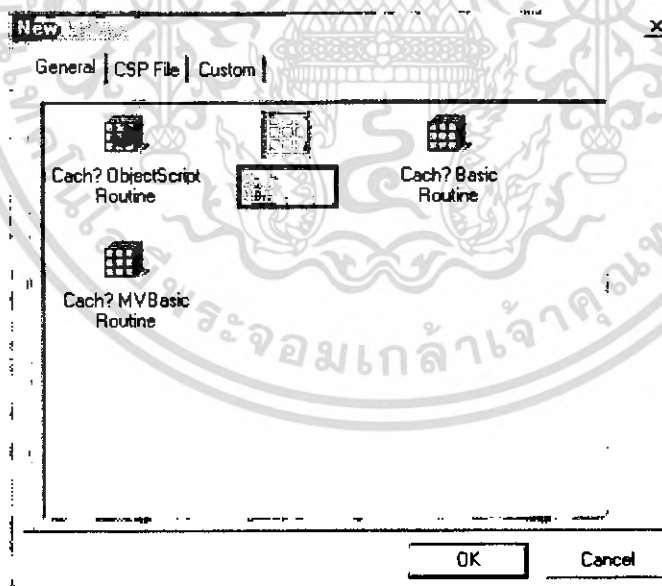
การสร้างคลาส

เริ่มเข้าสู่การใช้งาน โดยการเลือกที่เมนู File->New ดังรูปที่ ก-16



รูปที่ ก-16 แสดงการเลือกเมนูการเริ่มต้นเข้าสู่การสร้างงาน

เลือกที่ CacheClass Definition เพื่อทำการสร้างคลาสของฐานข้อมูล ดังรูปที่ ก-17



รูปที่ ก-17 แสดงการเลือกชนิดไฟล์งานที่ต้องการสร้าง จากรูปเป็นการเลือก Cache Class Definition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะมี Wizard ช่วยเหลือในการสร้างคลาส เพื่อความสะดวกต่อผู้ใช้โปรแกรมมาก
ยิ่งขึ้น โดยในหน้าแรกนี้จะให้ใส่ชื่อ package และชื่อ class แล้วคลิก Next > ดังรูปที่ ก-18

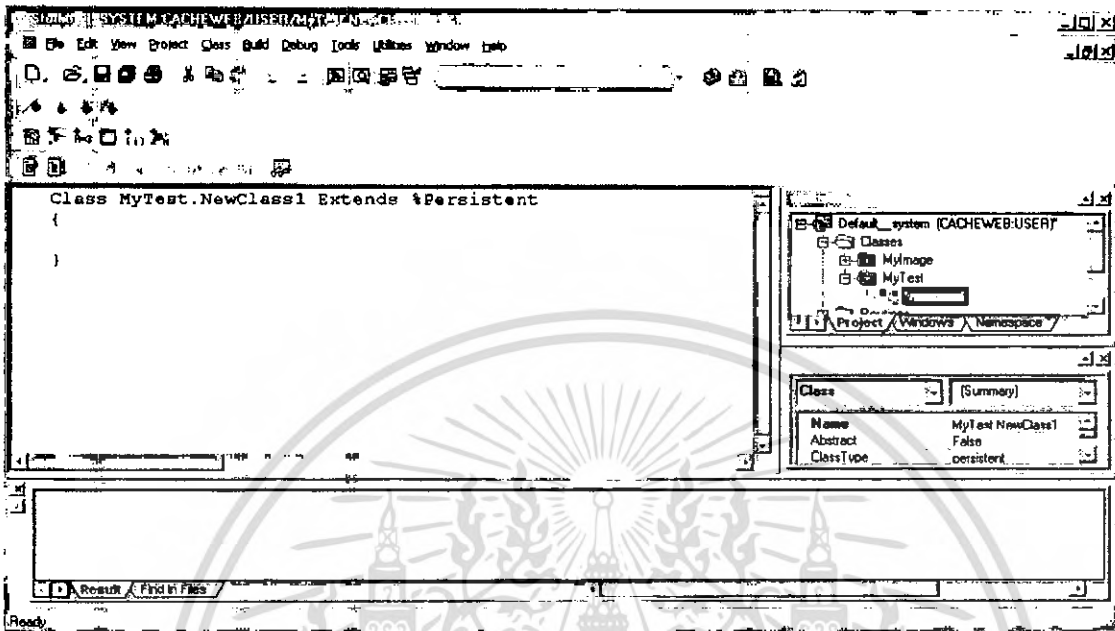
รูปที่ ก-18 แสดงหน้าแรกของโปรแกรม Wizard ที่ช่วยเหลือในการสร้างคลาส

ทำการเลือกชนิดของคลาสเป็นแบบ Persistent เนื่องจากเป็นคลาสของฐานข้อมูล แล้ว
คลิก Finish ดังรูปที่ ก-19

รูปที่ ก-19 แสดงหน้าการเลือกชนิดของคลาสของโปรแกรม Wizard ที่ช่วยเหลือในการสร้างคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Wizard จะทำการสร้างโค้ดโปรแกรมให้เป็นโครงสร้างตามที่เราต้องการ ดังรูปที่ ก-20

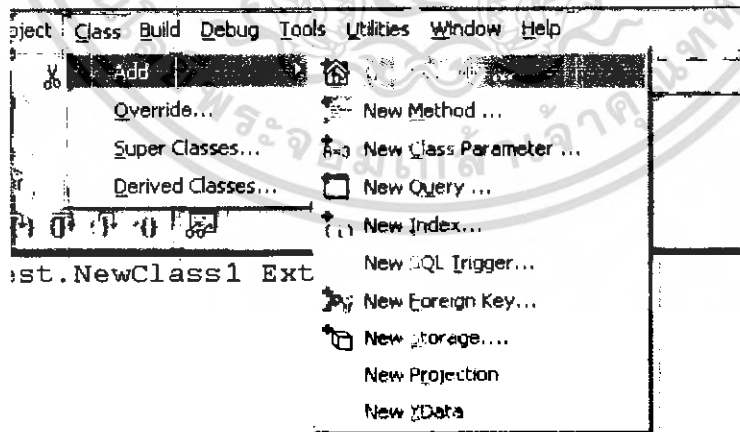


รูปที่ ก-20 แสดงโค้ดโปรแกรมหลังจากโปรแกรม Wizard การสร้างคลาสเสร็จสิ้น

การสร้าง property

ทำการสร้าง property ให้กับคลาสโดยเลือกที่เมนู Class->Add->New Property ดังรูปที่ ก-

21



รูปที่ ก-21 แสดงการเลือกเมนูการสร้าง property

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นโปรแกรม Wizard จะช่วยในการจัดการสร้าง property ให้ง่ายขึ้น โดยในหน้าแรก นี้จะเป็นการใส่ชื่อของ property แล้วคลิก Next > ดังรูปที่ ก-22

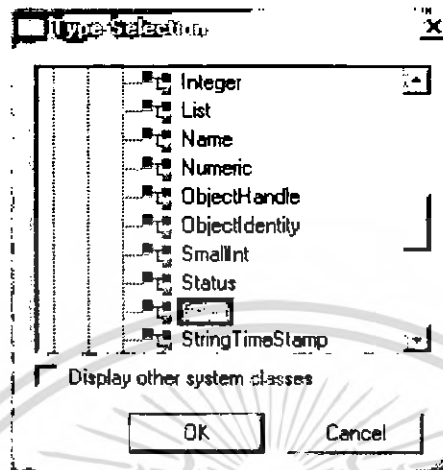
รูปที่ ก-22 แสดงหน้าแรกของโปรแกรม Wizard ที่ช่วยเหลือในการสร้าง property

ทำการเลือกชนิดของ property โดยสามารถคลิกที่ Browse... ดังรูปที่ ก-23

รูปที่ ก-23 แสดงหน้าการเลือกชนิดของ property ของโปรแกรม Wizard การสร้าง property

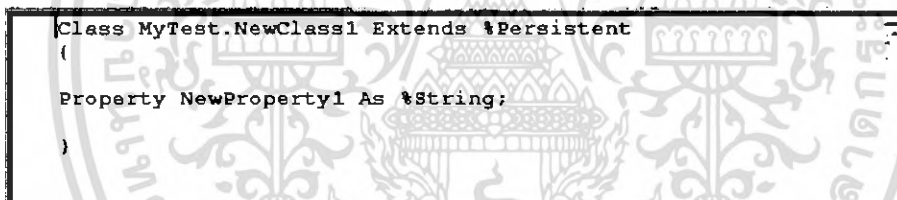
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจะขึ้นได้อะลือกให้เลือกชนิดของ property ต่าง ๆ โดยแต่ละชนิดคืออะไรนั้นสามารถหาข้อมูลได้จากเอกสารคู่มือการใช้ที่ควาน์โหลดมา แล้วคลิก OK และ Finish ดังรูปที่ ก-24



รูปที่ ก-24 แสดงได้อะลือกเมื่อคลิกปุ่ม Browse...

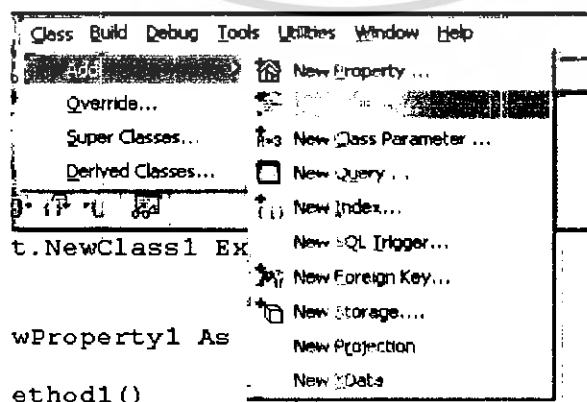
Wizard จะทำการสร้างโค้ดโปรแกรมให้ดังรูปที่ ก-25



รูปที่ ก-25 แสดงโค้ดโปรแกรมหลังจากโปรแกรม Wizard การสร้าง property สำเร็จ

การสร้างเมธอด

การสร้างเมธอดให้กับคลาสทำได้โดยการเลือก Add->New Method ดังรูปที่ ก-26



รูปที่ ก-26 การเลือกเมนูเพื่อทำการสร้างเมธอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Wizard จะทำการช่วยเหลือในการสร้างเมธอด โดยในหน้าแรกนี้จะให้ทำการใส่ชื่อเมธอด แล้วคลิก Next> ดังรูปที่ ก-27

รูปที่ ก-27 แสดงหน้าแรกของโปรแกรม Wizard ที่ช่วยเหลือการสร้างเมธอด

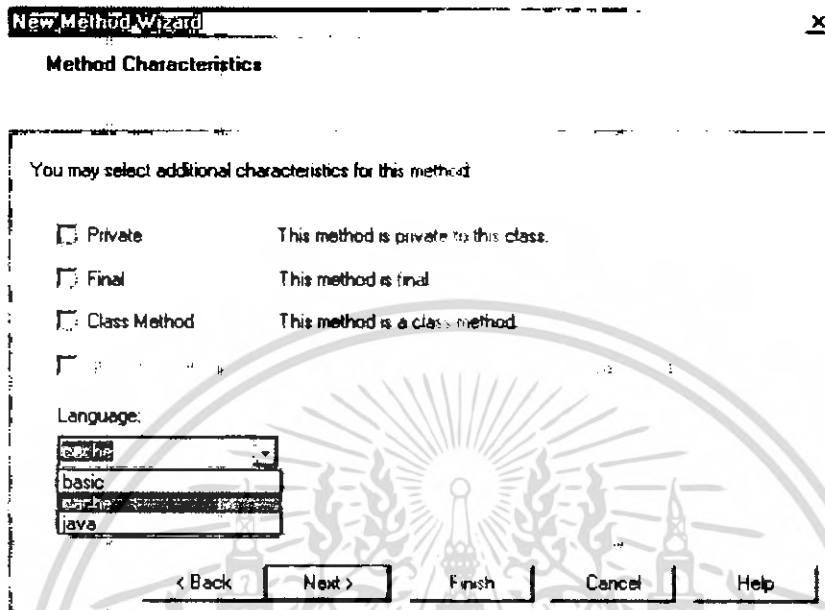
จากนั้นทำการเลือกชนิดข้อมูลที่ต้องการรีเทิร์นค่า และค่าพารามิเตอร์ที่ต้องการปาสเข้าไป โดยค่าพารามิเตอร์ที่ต้องการปาสเข้าไปนั้นให้คลิกที่ปุ่ม Browse ดังรูปที่ ก-28

รูปที่ ก-28 แสดงหน้าการเลือกคาร์เทิร์นและค่าพารามิเตอร์ที่ต้องการปาสเมธอด ของโปรแกรม

Wizard การสร้างเมธอด

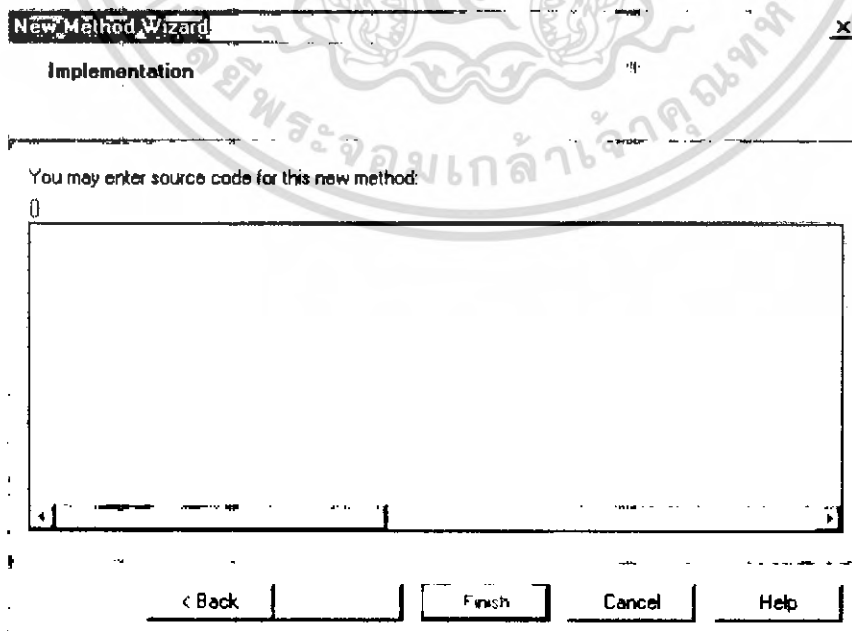
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกภาษาที่ต้องการเขียนภายในเมธอด ซึ่งมี 3 ภาษาให้เลือกคือ basic, cache, java ดังรูปที่ ก-29



รูปที่ ก-29 แสดงหน้าการเลือกภาษาโปรแกรมที่ต้องการจะเขียนภายในเมธอด ของโปรแกรม Wizard การสร้างเมธอด

ทำการใส่โค้ด โปรแกรมภายในฟังก์ชันได้ในหน้า Implement หรือสามารถนำไปเขียนภายนอก Wizard ก็ได้ จากนั้นคลิก Finish ดังรูปที่ ก-30



รูปที่ ก-30 แสดงหน้าการพิมพ์โค้ดภายในเมธอด ของโปรแกรม Wizard การสร้างเมธอด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเสร็จสิ้นโปรแกรม Wizard การสร้างเมธอดแล้ว จะปรากฏโค้ดโปรแกรมดังรูปที่ ก-

31

```

Class MyTest.NewClass1 Extends %Persistent
{
    Property NewProperty1 As %String;

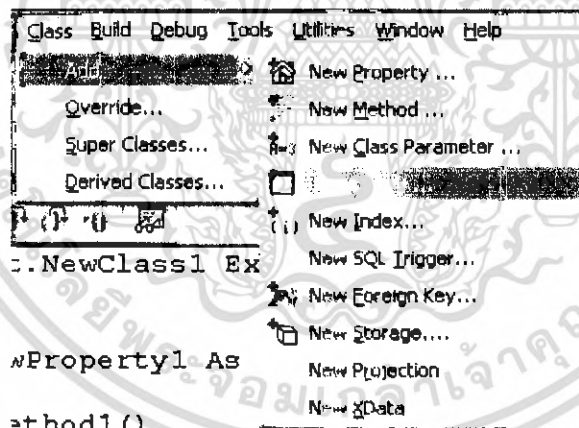
    Method NewMethod1 ()
    {
    }
}

```

รูปที่ ก-31 แสดงโค้ดโปรแกรมเมื่อโปรแกรม Wizard การสร้างเมธอดเสร็จสิ้น

การสร้าง Query

ในการสร้าง Query ก็สามารสร้างโดยใช้โปรแกรม Wizard ได้เหมือน ๆ กับในข้างต้น โดยเลือกที่เมนู Class->Add->New Query ดังรูปที่ ก-32




รูปที่ ก-32 แสดงการเลือกเมนูเพื่อสร้าง Query

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Wizard จะทำการช่วยเหลือในการสร้าง query ดังนี้โดยในหน้าแรกนี้ให้ใส่ชื่อของ query แล้วคลิก Next > ดังรูปที่ ก-33

รูปที่ ก-33 แสดงหน้าแรกของโปรแกรม Wizard เพื่อช่วยเหลือในการสร้าง Query

ทำการใส่พารามิเตอร์ที่จะอินพุตเข้าไปใน query โดยสามารถคลิกปุ่ม  เพื่อทำการตั้งชื่อพารามิเตอร์และชนิดของข้อมูลที่ต้องการ แล้วคลิก OK ดังรูปที่ ก-34

รูปที่ ก-34 แสดงหน้าของการสร้างพารามิเตอร์อินพุต ของโปรแกรม Wizard การสร้าง Query

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

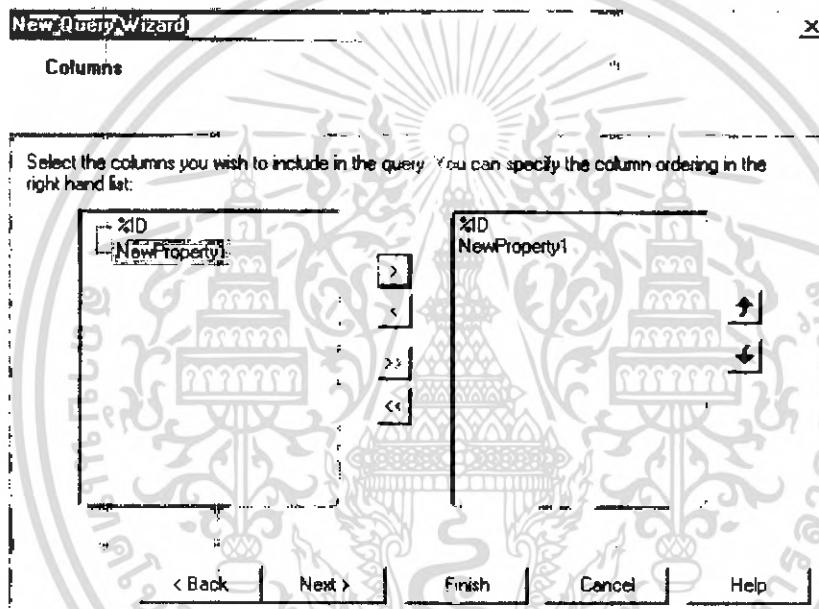
จะได้ผลลัพธ์ดังรูปที่ ก-35 ซึ่งสามารถสร้างได้มากกว่า 1 พารามิเตอร์

Name	Type	Default Value
Param1	%String	

รูปที่ ก-35 แสดงพารามิเตอร์อินพุตที่ถูกสร้างขึ้น

จากนั้นจะทำการเลือกคอลัมน์ที่ต้องการเป็นเอาท์พุทออกมา แล้วคลิก Next > ดังรูปที่ ก-

36



รูปที่ ก-36 แสดงหน้าการเลือกคอลัมน์ที่ต้องการแสดงในผลลัพธ์ของการ Query ของโปรแกรม Wizard การสร้าง Query

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการสร้างเงื่อนไขของการ query โดยเลือกตามที่ปรากฏในรูป แล้วคลิก Next >
 ดังรูปที่ ก-37

รูปที่ ก-37 แสดงหน้าของการสร้างเงื่อนไขของ Query ของโปรแกรม Wizard การสร้าง Query

จากนั้นเราก็สามารถเลือกได้ว่าจะให้จัดเรียงข้อมูลตามคอลัมน์ไหน จากมากไปน้อย หรือ
 จากน้อยไปมาก แล้วคลิก Finish ดังรูปที่ ก-38

รูปที่ ก-38 แสดงหน้าของการเลือกการจัดเรียงข้อมูล ของโปรแกรม Wizard การสร้าง Query

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเสร็จสิ้น โปรแกรม Wizard การสร้าง Query จะได้ผลลัพธ์เป็น โค้ดดังรูปที่ ก-39

```

Class MyTest.NewClass1 Extends %Persistent
{
    Property NewProperty1 As %String;

    Method NewMethod1()
    {
    }

    Query NewQuery1(Param1 As %String) As %SQLQuery (CONTAINID = 1)
    {
        SELECT %ID,NewProperty1 FROM NewClass1
        WHERE (NewProperty1 = :Param1)
    }
}

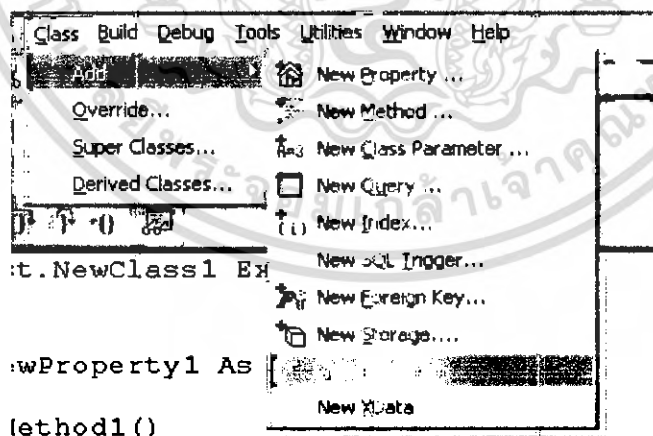
```

รูปที่ ก-39 แสดงโค้ดโปรแกรมเมื่อโปรแกรม Wizard การสร้าง Query เสร็จสิ้น

การสร้าง Projection เพื่อการ generate code เป็นภาษาจาวา

เนื่องจากในโครงการนี้ใช้ภาษาจาวาในการอิมพลิเมนต์ ดังนั้นจึงต้องสร้าง Projection ขึ้นมาซึ่งไว้สำหรับให้แปลงคลาสฐานข้อมูลใน Caché ให้ไปเป็นโค้ดภาษาจาวาซึ่งมี property และเมธอดที่เราสร้างไว้ใน Caché และมีเมธอดพื้นฐานอื่น ๆ ที่สำคัญอยู่ด้วย

การสร้าง Projection นี้ทำได้โดยการไปที่เมนู Class->Add->New Projection ดังรูปที่ ก-40



รูปที่ ก-40 แสดงการเลือกเมนูการสร้าง Projection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Wizard จะช่วยเหลือในการสร้างโค้ดโดยในหน้าแรกนี้ให้ใส่ชื่อ Projection ดังรูปที่ ก-41

รูปที่ ก-41 แสดงหน้าแรกของโปรแกรม Wizard ในการสร้าง Projection

จากนั้นทำการใส่ path ที่ต้องการเก็บโปรแกรมโค้ดภาษาจาวา ลงในช่องของ ROOTDIR ดังรูปที่ ก-42

รูปที่ ก-42 แสดงหน้าของการใส่ path ที่ต้องการเก็บโค้ดภาษาจาวาของโปรแกรม Wizard ในการสร้าง Projection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อโปรแกรม Wizard ในการสร้าง Projection เสร็จสิ้นจะแสดงโค้ดดังรูปที่ ก-43

```

Class MyTest.NewClass1 Extends %Persistent
{
    Property NewProperty1 As %String;

    %Method NewMethod1()
    {
    }

    %Query NewQuery1(Param1 As %String) As %SQLQuery (CONTAINID = 1)
    {
        SELECT %ID,NewProperty1 FROM NewClass1
        WHERE (NewProperty1 = :Param1)
    }

    Projection NewProjection1 As %Projection.Java(ROOTDIR = "C:\");
}

```

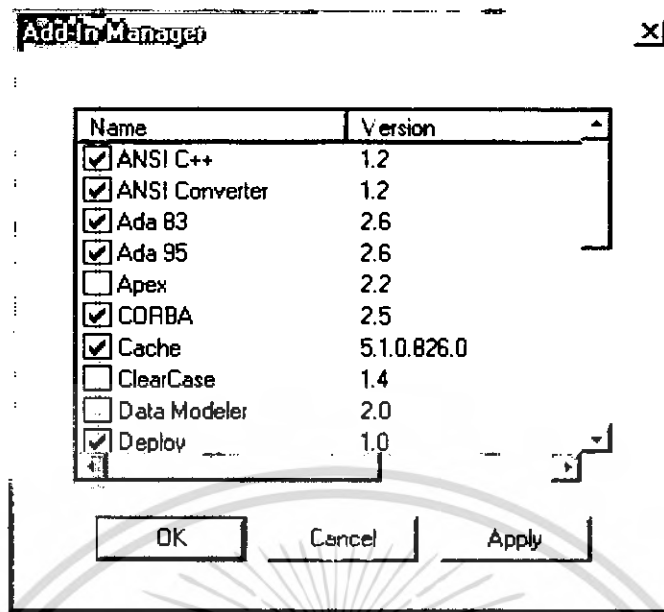
รูปที่ ก-43 แสดงโค้ดโปรแกรมเมื่อโปรแกรม Wizard ในการสร้าง Projection เสร็จสิ้น

การ Install และการใช้งาน RoseLink

RoseLink มีไว้สำหรับการเชื่อมต่อโปรแกรม Rational Rose กับโปรแกรม Caché Studio ซึ่งมีประโยชน์มากเนื่องจากการทำงานโดยใช้เทคโนโลยีเชิงออบเจกต์นั้นต้องมีการออกแบบคลาสไดอะแกรมซึ่งถ้ามี RoseLink แล้วจะสามารถทำการ export คลาสไดอะแกรมนั้นไปที่ Caché ได้เลย คือ จะทำการสร้างโครงของคลาสให้ตามที่ได้ออกแบบไว้ในคลาสไดอะแกรม โดยที่สามารถเลือกได้ด้วยว่าคลาสใดบ้างที่สมควร export ออกไปซึ่งก็ต้องเป็นคลาสที่เป็นฐานข้อมูล โดยที่ผู้ใช้งานเป็นผู้เลือก หลังจาก export เสร็จแล้ว Caché จะสร้างโครงของโค้ดโปรแกรมให้ตามคลาสไดอะแกรม โดยที่คลาสไหน export เข้ามาใน Caché คลาสนั้นก็จะถูกสร้างเป็นโค้ด 1 คลาส ซึ่งมี property และเมธอด(ในลักษณะโครงของเมธอด)ดังปรากฏในคลาสไดอะแกรม คลาสไหนมีการอินเฮอริท Caché ก็สามารถรู้ได้และทำการอิมพลีเมนต์ให้ เป็นต้น ซึ่งสร้างความสะดวกให้กับผู้ใช้งานได้ในระดับหนึ่ง

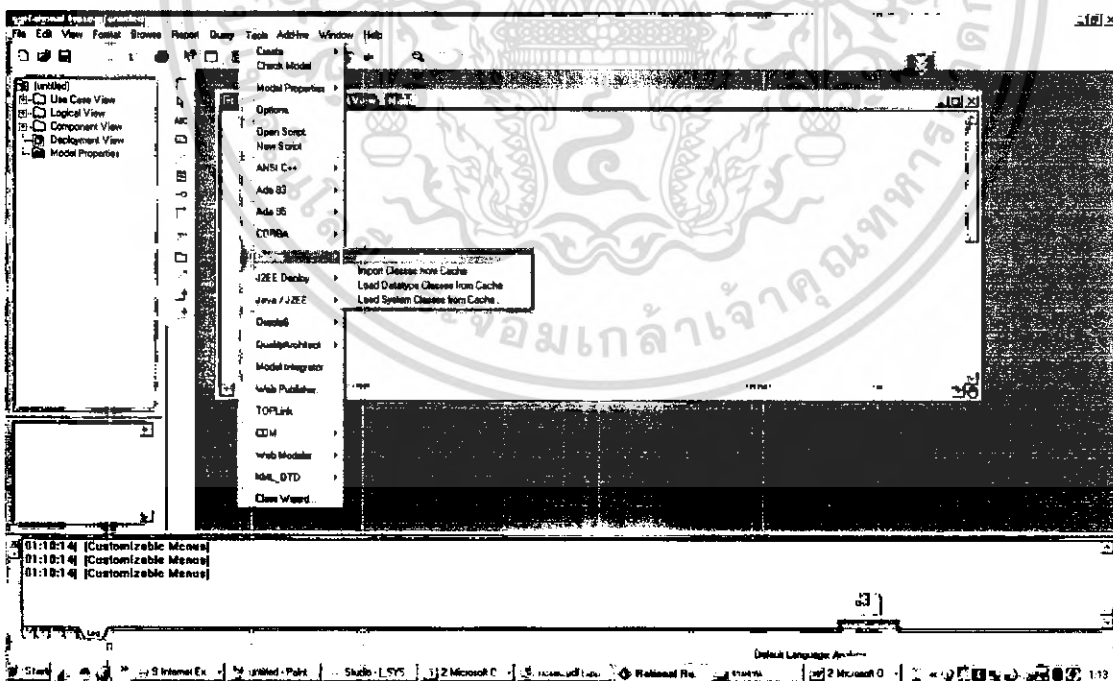
ขั้นตอนในการ Install RoseLink มีดังนี้

1. ปิดโปรแกรม Rational Rose
2. ถัดลอกไฟล์ CacheRoseLink.exe ไปที่ไดเรกทอรีที่เก็บงานที่ทำอยู่ จากนั้นรัน CacheRoseLink.exe
3. เปิดโปรแกรม Rose และเลือกที่ Add-Ins->Add-In Manager ทา Caché แล้วทำการเลือก แล้วคลิก OK แสดงดังรูปที่ ก-44



รูปที่ ก-44 แสดงการเลือก Cache ภายใน Rational Rose

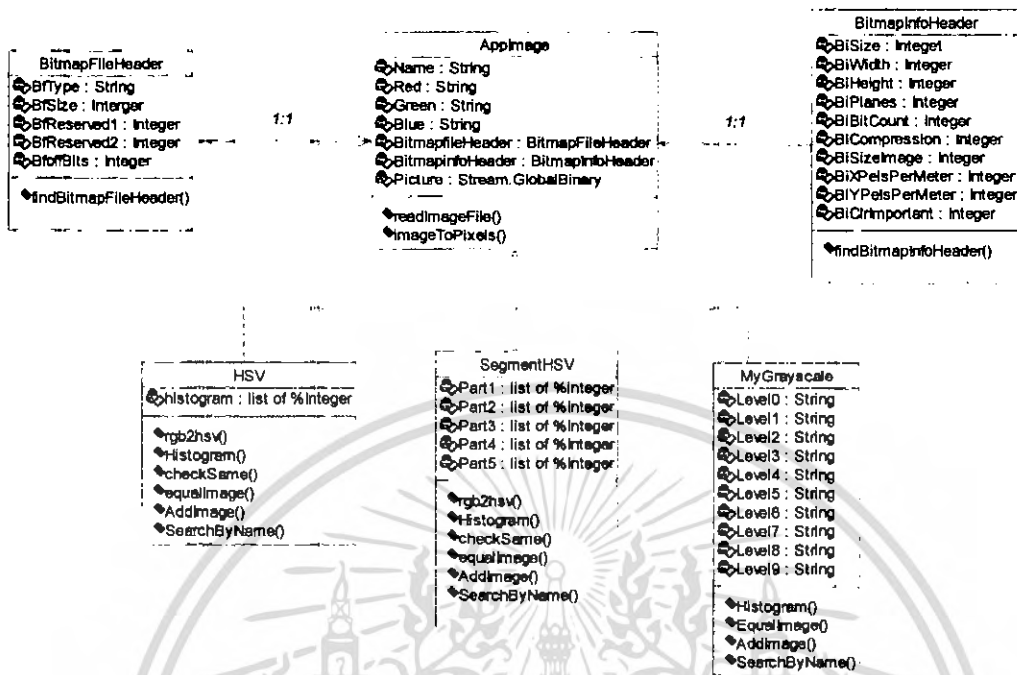
4. เมนู Cache จะปรากฏบนเมนู Tools ดังรูปที่ ซึ่งเราสามารถทำการ export คลาสไดอะแกรมได้จากเมนูนี้โดยการเลือก Tools->Cache->Export classes to Cache ดังรูปที่ ก-45



รูปที่ ก-45 แสดงการเลือกเมนูเพื่อต้องการ export คลาสไดอะแกรมไปที่ Cache

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ก-46 เป็นคลาสตัวอย่างที่จะทำการแสดงการ export ออกไปยัง Caché



รูปที่ ก-46 แสดงคลาสตัวอย่างที่จะทำการ export ไปยัง Cache

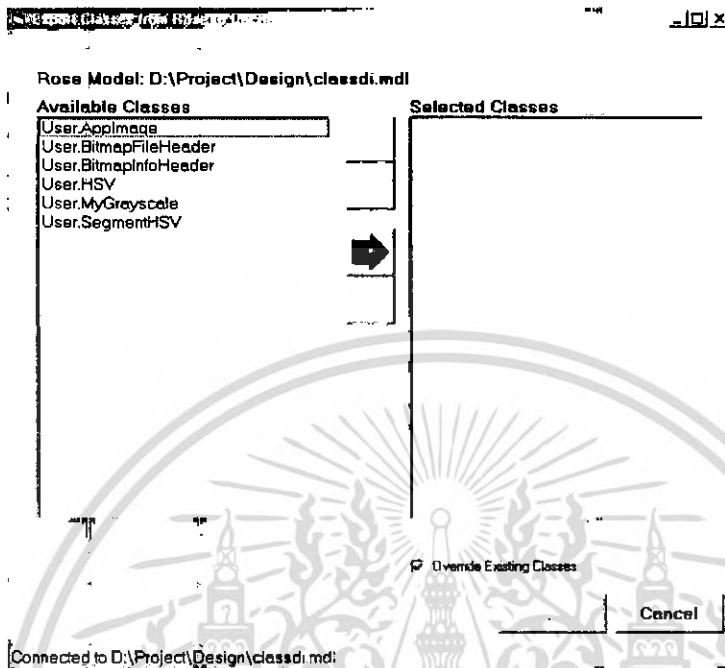
รูปที่ ก-47 เมื่อทำการเลือก Export classes to Caché แล้ว จะแสดงดังรูปที่ แล้วรอสักครู่



รูปที่ ก-47 แสดงรูป Rose Cache Link หลังจากเลือก Export classes to Cache แล้ว

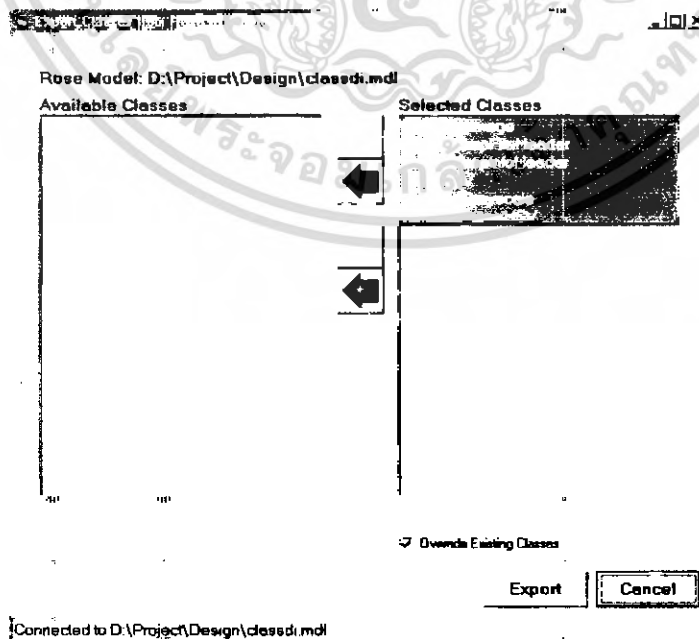
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะขึ้น ไดอะล็อกให้ทำการเลือกคลาสว่าจะเลือกคลาสไหน export ไปยัง Caché
 ดังรูปที่ ก-48



รูปที่ ก-48 แสดงไดอะล็อกการเลือกคลาสที่จะทำการ export ไปยัง Cache

เนื่องจากคลาสตัวอย่างคลาสทุกคลาสเป็นคลาสฐานข้อมูลทั้งหมด จึงเลือกทุกคลาสดังรูป
 ที่ ก-49 จากนั้นคลิก Export

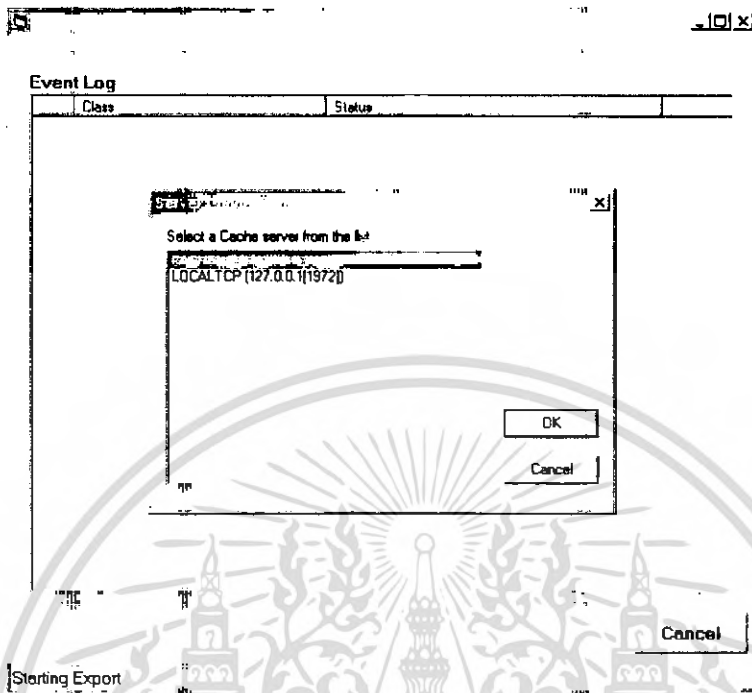


รูปที่ ก-49 แสดงการเลือกคลาสที่เป็นคลาสฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะขึ้นไดอะล็อกเพื่อทำการเลือก Server Connection ซึ่งให้เลือก CACHEWEB ดัง

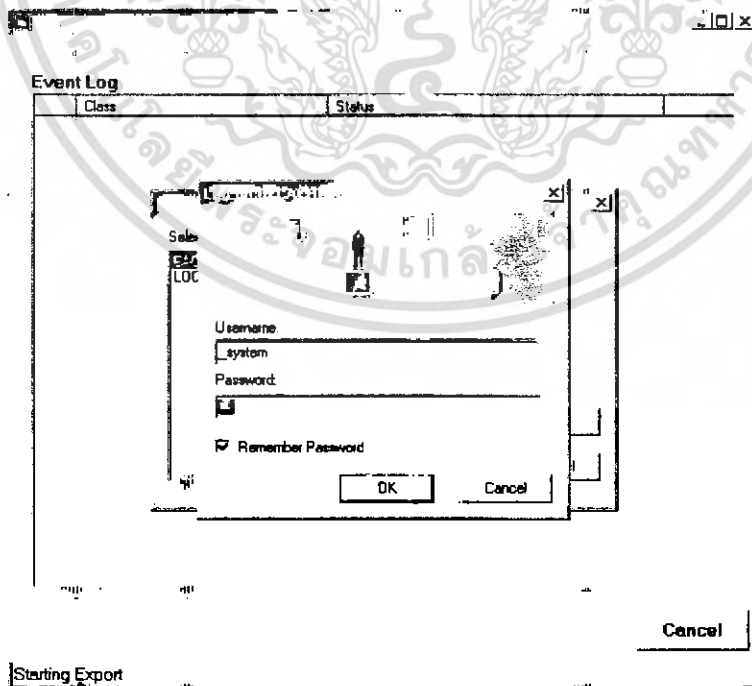
รูปที่ ก-50



รูปที่ ก-50 แสดงไดอะล็อกการเลือก Server Connection

จากนั้นทำการใส่ Username และ Password เพื่อทำการติดต่อกับฐานข้อมูล Cache ดังรูปที่

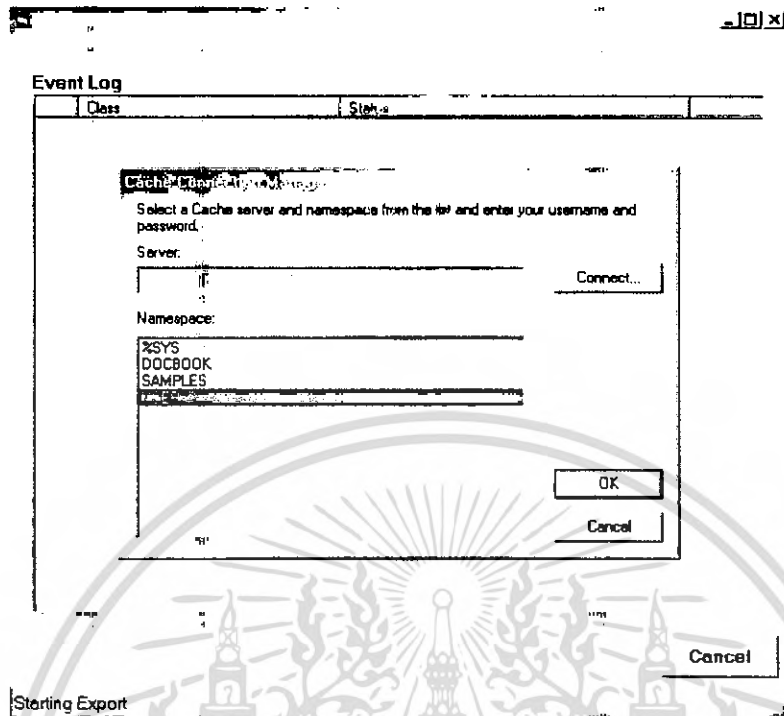
ก-51



รูปที่ ก-51 แสดงการล็อกอินเพื่อเข้าสู่ฐานข้อมูล Cache

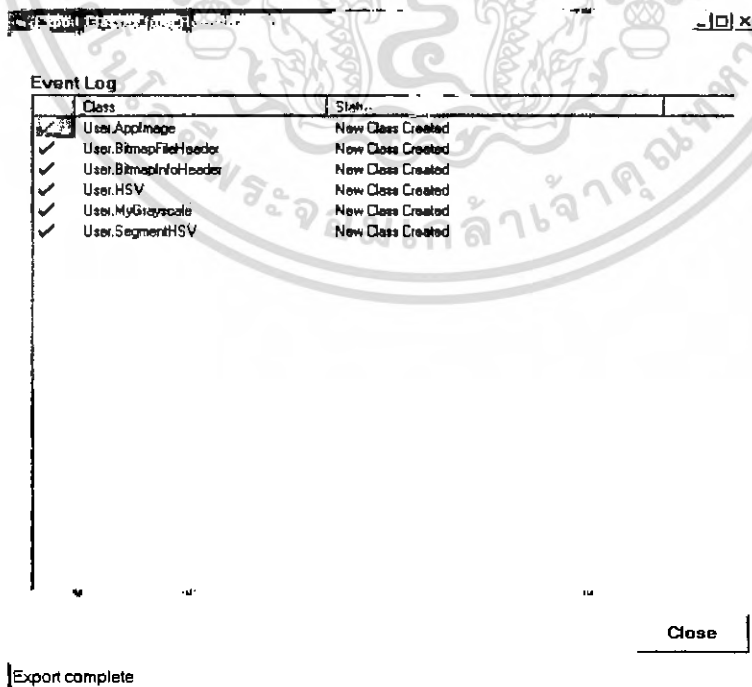
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือก Namespace ที่จะทำการสร้างโปรเจกงานของเรา ดังรูปที่ ก-52



รูปที่ ก-52 แสดงการเลือก Namespace

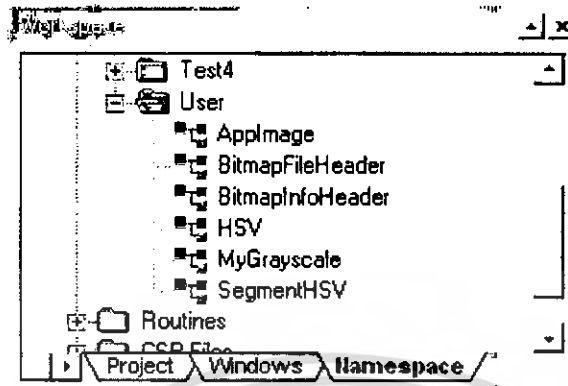
จากนั้นทำการรอเพื่อการประมวลผลจากคลาสไดอะแกรมไปสู่ Cache จนสำเร็จทุกคลาส โดยจะมีเครื่องหมายถูกแสดงอยู่ด้านหน้าชื่อคลาส ดังรูปที่ ก-53 แสดงว่าได้ทำการ export คลาส ไปที่ได้เรียบร้อยแล้ว Cache



รูปที่ ก-53 แสดงการประมวลผลในการ export คลาสไปยัง Cache ได้สำเร็จเรียบร้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน Caché Studio ก็จะมีคลาสสร้างให้ดังรูปที่ ก-54



รูปที่ ก-54 แสดงคลาสไออะแกรมที่ถูกสร้างขึ้นมาจากการ *export* มาจากคลาสไออะแกรม

และรูปที่ ก-55 แสดงตัวอย่างโค้ดของคลาส AppImage ซึ่งจะมี property และ โครงร่างของเมธอดคั้งที่ออกแบบไว้ในคลาส ไออะแกรม

```

Class User.AppImage [ Not ProcedureBlock ]
(
Property BitmapfileHeader As User.BitmapFileHeader [ Private ];
Property BitmapinfoHeader As User.BitmapInfoHeader [ Private ];
Property Blue As %String [ Private ];
Property Green As %String [ Private ];
Property Name As %String [ Private ];
Property Picture As Stream.GlobalBinary [ Private ];
Property Red As %String [ Private ];
Method imageToPixels() [ Language = cache ]
{
}
Method readImageFile() [ Language = cache ]
{
}
)

```

รูปที่ ก-55 แสดงตัวอย่างโค้ดที่ได้จากการ *export* มาจากคลาสไออะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

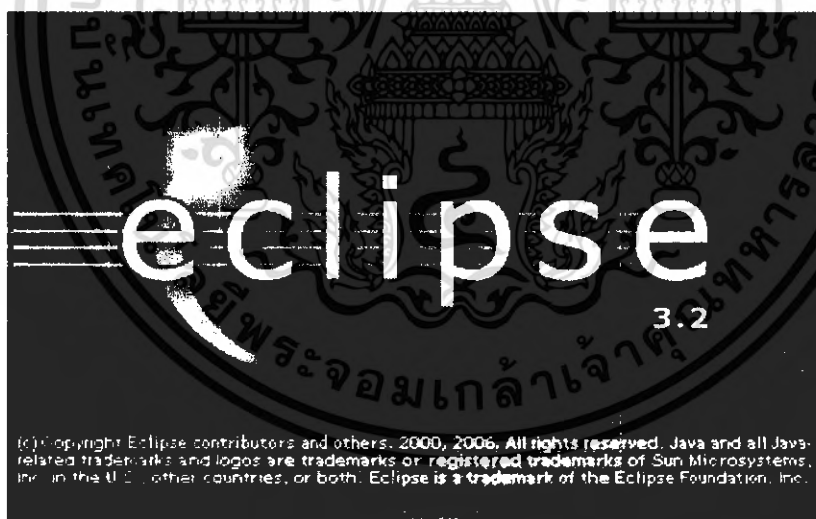
ภาคผนวก ข.

การติดต่อใช้งานฐานข้อมูล Caché กับ JAVA

การใช้งานกับโปรแกรมจาวา

เมื่อทำการสร้างคลาสที่ Caché Studio เรียบร้อยแล้ว และสร้าง Projection เพื่อให้คลาสที่สร้างใน Caché Studio เจนเนอร์เรทออกมาเป็นคลาสในโค้ดภาษาจาวา ดังที่กล่าวไว้ในภาคผนวก ก. เมื่อทำการรันที่ Caché Studio คลาสในโค้ดภาษาจาวาก็จะถูกสร้างขึ้นใน path ที่เรากำหนดไว้ ซึ่งเราจะนำคลาสภาษาจาวานั้นมาใช้งาน ซึ่งจะอธิบายต่อไปนี้

สำหรับในโครงการนี้ได้ใช้ eclipse เพื่อสร้าง โปรแกรมของโครงการโดยใช้ภาษาจาวา ซึ่ง eclipse จะช่วยอำนวยความสะดวกในการทำงานต่าง ๆ ได้พอสมควร สำหรับตัว Install โปรแกรมนี้ คือ eclipse-SDK-3.2-win32 และต้องลง jdk-1_5_0_07-windows-i586-p เพื่อใช้รันโค้ดจาวา ด้วย เมื่อทำการลงโปรแกรมทุกอย่างเสร็จเรียบร้อย และเปิดโปรแกรม eclipse ขึ้นมาจะแสดงดังรูปที่ ข-1



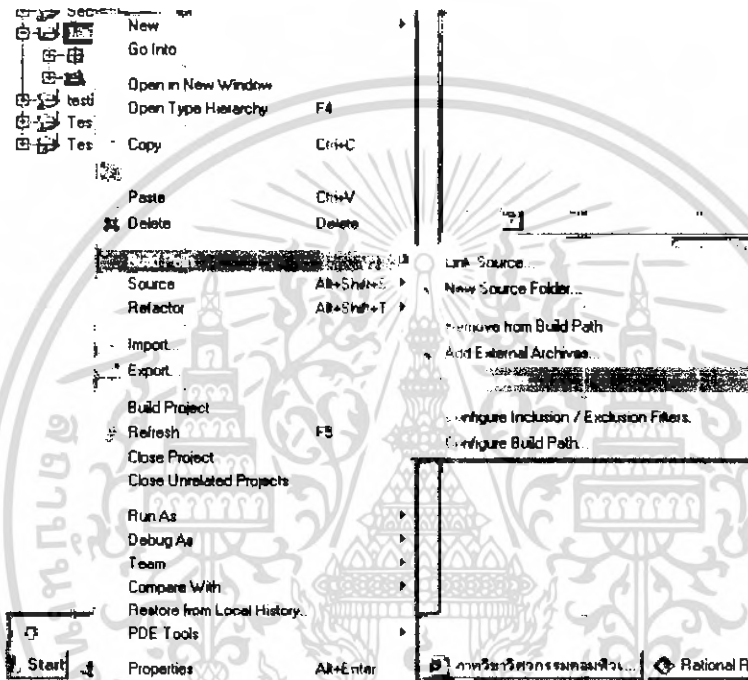
รูปที่ ข-1 แสดงโปรแกรม eclipse

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเพิ่มไลบรารีของ Caché เข้าสู่จาวา

เมื่อคลาสใน Caché Studio ถูกสร้างเป็น โค้ดจาวาเรียบร้อยแล้ว จากนั้นก็ไปทำการเอาไฟล์ โค้ดนั้นมาไว้ในโปรเจกของเราที่สร้างในโปรแกรม eclipse แต่ก่อนที่จะรันไฟล์จาวาดังกล่าวได้นั้น ต้องทำการเพิ่มไลบรารีเข้าไปในไฟล์โปรเจกที่เราสร้างงานขึ้นเสียก่อน โดยมีวิธีการดังนี้

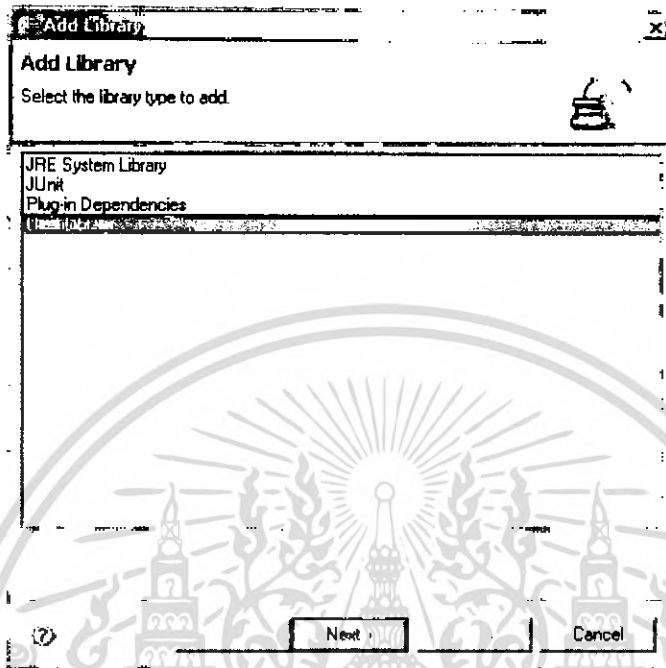
1. เมื่อทำการสร้างโปรเจกขึ้นมาเรียบร้อยแล้ว ให้นำเมาส์ไปชี้ที่ชื่อโปรเจกแล้วคลิกขวา และทำการเลือกที่ Build Path->Add Libralies แสดงดังรูปที่ ข-2



รูปที่ ข-2 แสดงการเลือกเมนูเพื่อทำการเพิ่มไลบรารี Cache เข้าสู่ไฟล์โปรเจกจาวา

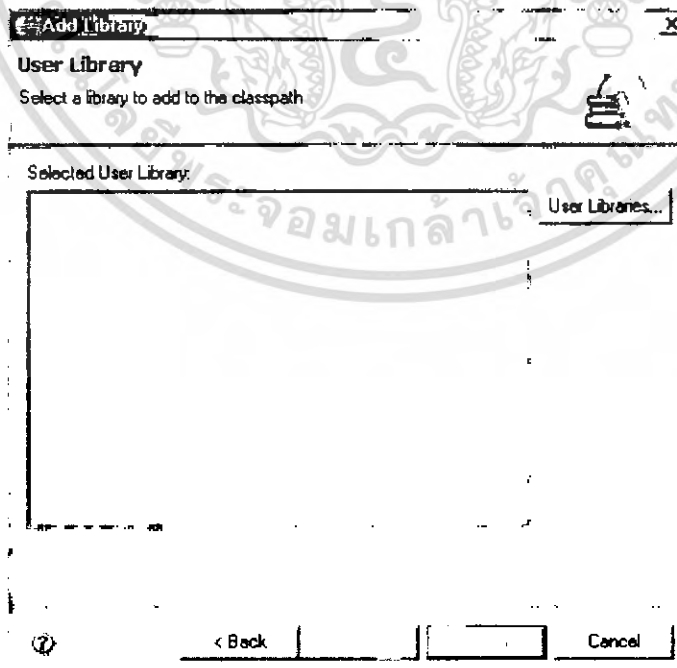
เมื่อเลือกที่ Add Library แล้วจะเข้าสู่ไดอะล็อกดังรูปที่ ข-3 เลือก User Library แล้วคลิก

Next >



รูปที่ ข-3 แสดงไดอะล็อกการเพิ่มไลบรารี

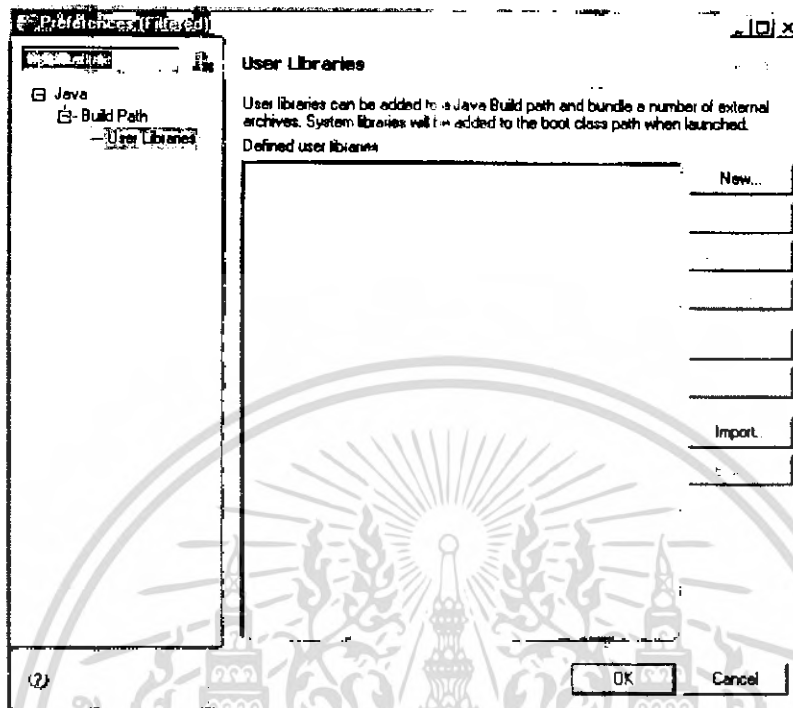
ทำการคลิกที่ปุ่ม User Libraries... เพื่อทำการสร้าง User Libraries ดังรูปที่ ข-4



รูปที่ ข-4 แสดงหน้าของการเข้าสู่การสร้าง User Libraries

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

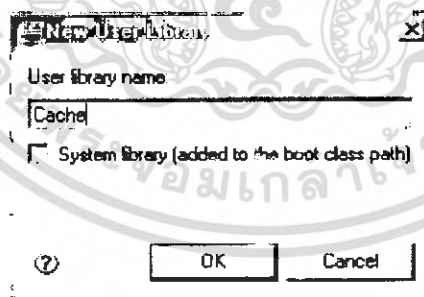
ให้ทำการคลิกที่ปุ่ม New... เพื่อสร้าง Libraries ที่เราต้องการ ดังรูปที่ ข-5



รูปที่ ข-5 แสดงหน้าของการสร้าง User Libraries

เมื่อคลิกที่ New... แล้วจะขึ้นไดอะล็อกให้ใส่ชื่อไลบรารีลงไป จากนั้นคลิก OK ดังรูปที่

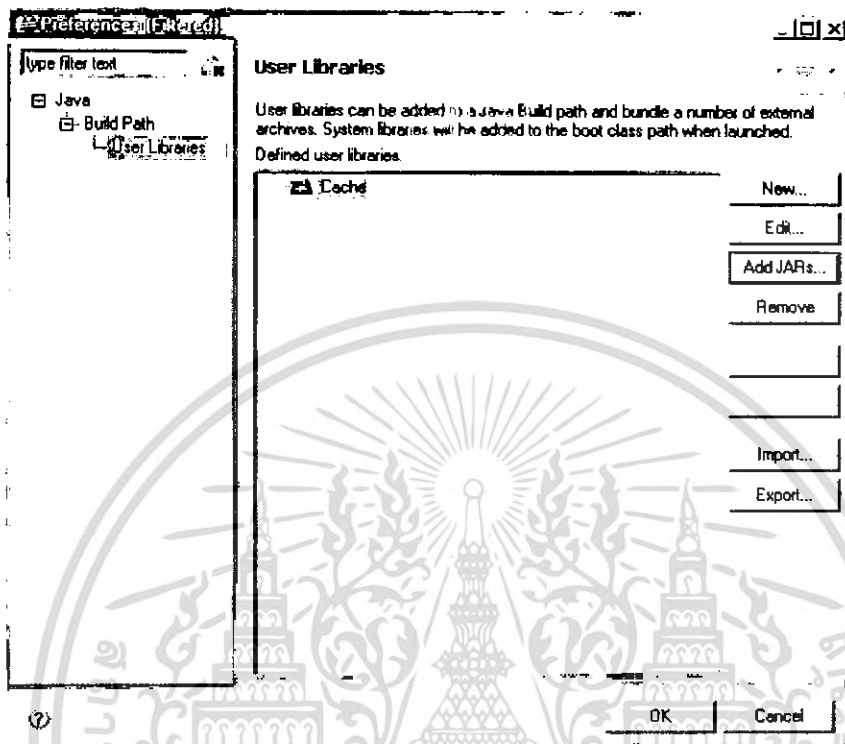
ข-6



รูปที่ ข-6 แสดงไดอะล็อกของการตั้งชื่อไลบรารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

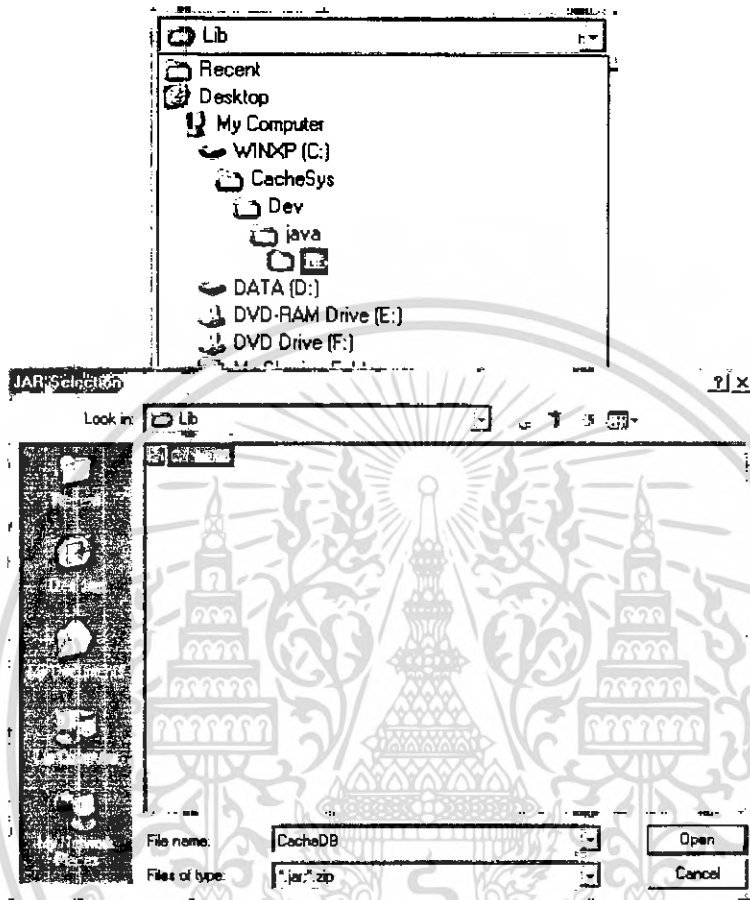
เมื่อตั้งชื่อเสร็จเรียบร้อยแล้ว ให้คลิกที่ปุ่ม Add JARs... เพื่อทำการหาไฟล์ที่เป็นไลบรารีที่เราต้องการเพิ่มเข้าไป ดังรูปที่ ข-7



รูปที่ ข-7 แสดงหน้าของการเข้าสู่การเพิ่มไฟล์ไลบรารีเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

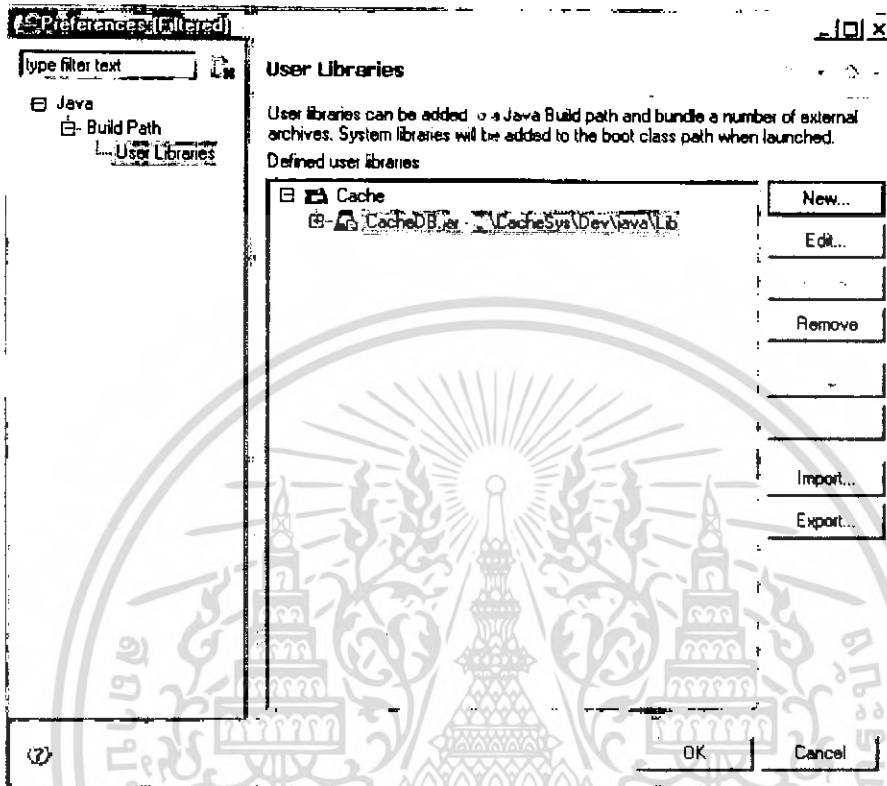
เมื่อคลิกที่ปุ่ม Add JARs... แล้วทำการค้นหาไฟล์ไลบรารีของ Cache ชื่อว่า CacheDB.jar
ซึ่งอยู่ในพาท C:\CacheSys\Dev\java\Lib ดังรูปที่ ข-8



รูปที่ ข-8 แสดงการเลือกไฟล์ไลบรารีของ Cache

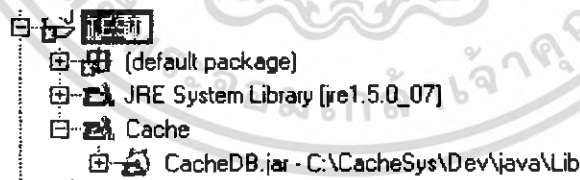
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกไฟล์เสร็จเรียบร้อยแล้ว จะขึ้นแสดงดังรูปที่ ข-9 ถ้าไม่ต้องการเพิ่มไฟล์อื่นที่
ต้องการใช้อีกแล้ว ให้คลิกที่ OK



รูปที่ ข-9 แสดงหน้าของการเพิ่มไลบรารีเสร็จเรียบร้อยแล้ว

เมื่อกระบวนการข้างต้นเสร็จเรียบร้อยแล้ว ใน path ของโปรเจกงานจะขึ้นไลบรารีที่สร้าง
ไว้ดังรูปที่ ข-10



รูปที่ ข-10 แสดงไลบรารีที่เพิ่มเข้าไปสู่โปรเจกงานเป็นที่เรียบร้อยแล้ว

จากนั้นก็ทำการสร้างคลาส main ขึ้นมาเพื่อใช้งานคลาสฐานข้อมูลต่าง ๆ โดยหลักการ
เขียนโปรแกรมตามแนวคิดเชิงวัตถุ ซึ่งรายละเอียดการทำงานพื้นฐานจะกล่าวในภาคผนวกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

การเขียนโปรแกรมในการจัดเก็บแล้วค้นหารูปภาพ

ภายในฐานข้อมูลเชิงวัตถุ Caché

การเขียนโปรแกรมในการจัดเก็บรูปภาพ ฐานข้อมูลเชิงวัตถุ Caché มีคุณสมบัติ(Property) %Stream.GlobalBinary ซึ่งเป็นคุณสมบัติ (Property) ของรูปภาพที่จะจัดเก็บลงในฐานข้อมูล ดังโค้ด

1. คลาสที่ใช้ในการการจัดเก็บรูปภาพ

```
MyImages.AppImage Extends %Persistent
{
Property Name As %String;
Property Red As %String;
Property Blue As %String;
Property Green As %String;
Property BitmapfileHeader As BitmapFileHeader;
Property BitmapinfoHeader As BitmapInfoHeader;
Property Picture As %Stream.GlobalBinary(LOCATION = "^MyApp", STORAGE =
"GLOBAL");
Projection MyTestImagr As %Projection.Java(ROOTDIR = "C:\Project");
Method readImageFile(fileName As %String) [ Language = java ]
{
    Toolkit tk = Toolkit.getDefaultToolkit();
    Image image = tk.getImage (fileName);
    MediaTracker tracker = new MediaTracker (this);
    tracker.addImage (image, 1);
    try {
        tracker.waitForID (1);
    }
    catch (InterruptedException e) {
        System.out.println (e);
    }
    return image;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Method imageToPixels() [ Language = java ]
{
    int numRows = image.getHeight (this);
    int numCols = image.getWidth (this);
    int[] oneDPixels = new int[numRows*numCols];

    PixelGrabber grabber = new PixelGrabber(image, 0, 0, numCols, numRows,
        oneDPixels ,0 ,numCols);

    try {
        grabber.grabPixels (0);
    }
    catch (InterruptedException e) {
        System.out.println (e);
    }
    int[] pixels = new int[numRows][numCols][4];
    for(int row = 0; row < numRows; row++){
        int[] aRow = new int [numCols];
        for(int col = 0; col < numCols; col++){
            int element = row * numCols + col;
            aRow[col] = oneDPixels[element];
        }
        for(int col=0; col < numCols; col++){
            pixels[row][col][0] = (aRow[col] >> 24) & 0xFF; // Alpha
            pixels[row][col][1] = (aRow[col] >> 16) & 0xFF; // Red
            pixels[row][col][2] = (aRow[col] >> 8) & 0xFF; // Green
            pixels[row][col][3] = (aRow[col]) & 0xFF; // Blue
        }
    }
}
}
}

```

จากโค้ดโปรแกรมข้างต้น เป็นโค้ดที่เขียนภายในฐานข้อมูล Caché โดย คลาส AppImage เป็นคลาสที่อยู่ภายในแพ็คเกจ MyImages โดยภายในคลาส AppImage ของฐานข้อมูลมีการจัดเก็บข้อมูลดังต่อไปนี้

- คุณสมบัติ (Property) Name , Red , Blue , Green ซึ่งมีชนิดของข้อมูลเป็นข้อความ (%String) โดยจะใช้ในการจัดเก็บชื่อของรูปภาพ และสีจากจุดพิกเซลของรูปภาพทั้ง 3 สี คือ สีแดง สีเขียว และสีน้ำเงิน
- คุณสมบัติ (Property) BitmapfileHeader และ BitmapinfoHeader ซึ่งมีชนิดของข้อมูลเป็นคลาส BitmapfileHeader และ คลาส BitmapinfoHeader ซึ่งคลาสทั้งสองเป็นคลาสภายในของฐานข้อมูล ที่จะทำการจัดเก็บค่า Header ของรูปภาพลงในฐานข้อมูล
- คุณสมบัติ (Property) Picture ซึ่งมีชนิดของข้อมูลเป็น %Stream.GlobalBinary ซึ่งเป็นชนิดของข้อมูลที่ฐานข้อมูล Caché มีไว้รองรับการจัดเก็บรูปภาพ

โดยในการกำหนดคุณสมบัติให้กับข้อมูลภายในฐานข้อมูล จะใช้คำสั่ง Property ตามด้วยชื่อคุณสมบัติ ตาม ด้วย As และชนิดของคุณสมบัติ นั้น เช่น Property Name As %String; หมายถึงเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติชื่อ Name มีชนิดข้อมูลเป็น %String หรืออาจใช้โปรแกรม Wizard ช่วยในการสร้างดังกล่าวแล้วในภาคผนวก ก.

ในส่วนของ Projection เป็นคำสั่งที่ใช้ในการระบุว่าจะทำการรันโค้ดที่เขียนดังกล่าวแล้วจะนำโค้ดภาษาจาวาที่ได้จากการรันไปเก็บไว้ที่ใด จากตัวอย่าง

```
Projection MyTestImagr As %Projection.Java (ROOTDIR = "C:\Project");
```

หมายถึงว่า เมื่อรันโค้ดข้างต้นแล้วจะได้โค้ดภาษาจาวาเก็บไว้ที่ C:\Project

- Method readImageFile(fileName As %String) มีการทำงานภายใน โดยรับชื่อไฟล์ขึ้นมาเป็นแล้วทำการเปลี่ยนไฟล์ดังกล่าวเป็น Image โดยภาษาที่ใช้ในการเขียนการทำงานภายใน Method ได้ใช้ภาษาจาวา โดยการกำหนดที่ [Language = java]

- Method imageToPixels() มีการทำงานภายใน โดยนำ Image ที่ได้มาหาค่าสี 3 สี คือ สีแดง , สีเขียว และสีน้ำเงิน ของในแต่ละพิกเซล เพราะในการจัดเก็บรูปภาพมีการจัดเก็บสีของรูปภาพเพื่อใช้ในการค้นหารูปภาพด้วย โดยภาษาที่ใช้ในการเขียนการทำงานภายในเมธอดได้ใช้ภาษาจาวา โดยการกำหนดที่ [Language = java]

2. คลาสที่ใช้ในการจัดเก็บ Header ของแต่ละรูปภาพ

- คลาสในการจัดเก็บ BitmapFileHeader ลงในฐานข้อมูล
Class MyImages.BitmapFileHeader Extends %Persistent
{

Property BfType As %String;

Property BfSize As %Integer;

Property BfReserved1 As %Integer;

Property BfReserved2 As %Integer;

Property BfOffBits As %Integer;

Projection MyTestImagr As %Projection.Java(ROOTDIR = "C:\Project");

}

```

Query byname(name As %String) As %SQLQuery(CONTAINID = 1)
{
SELECT %ID FROM HSV
WHERE (Name = :name)
}

```

```

Query byAll() As %SQLQuery(CONTAINID = 1)
{
SELECT %ID FROM HSV
}

```

```

Method Histogram() [ Language = java ]
{
}

```

```

Method equalImage(filename As %String) [ Language = java ]
{
}
}

```

จากโค้ดโปรแกรมข้างต้น เป็นโค้ดที่เขียนภายในฐานข้อมูล cache โดย คลาส HSV เป็น คลาสที่อยู่ภายในแพ็คเกจ MyImages โดยทำการสืบทอดคุณสมบัติจากคลาส AppImage เพื่อใช้ในการจัดเก็บรูปภาพ และเพิ่มเติมคุณสมบัติบางประการเพื่อใช้ในการค้นหารูปภาพโดยภายในคลาส HSV ของฐานข้อมูลได้เพิ่มเติมในส่วนของการจัดเก็บข้อมูลดังต่อไปนี้

- คุณสมบัติ (Property) histograms ซึ่งชนิดของข้อมูลเป็นลิสต์ของชนิดข้อมูลตัวเลข (list Of %Integer) โดยจะใช้ในการจัดเก็บค่าฮิสโตแกรมของทุกพิกเซลในรูปภาพโดยสามารถคำนวณหาค่าฮิสโตแกรมได้จาก Method Histogram()
- Query byid(id As %Integer) เป็นค้นหาข้อมูลจากการใช้ id
- Query byname(name As %String) เป็นค้นหาข้อมูลจากการใช้ ชื่อของรูปภาพในการค้นหา
- Query byAll() เป็นค้นหาข้อมูลทั้งหมด

จาก Method Histogram() เป็นการคำนวณค่าฮิสโตแกรมของรูปภาพที่ต้องการการจัดเก็บ
ลงในฐานข้อมูล โดยแสดงรายละเอียดได้ดังนี้

```
int H0=0,H1=0,H2=0,H3=0,H4=0,H5=0,H6=0,H7=0,H8=0,H9=0,H10=0,H11=0,H12=0,H13=0,
for (int row=0; row < pixels.length; row+=1) {
    for (int col=0; col < pixels[row].length; col++) {
        float Histogramout = rgb2hsv(pixels[row][col][1],pixels[row][col][2],p
        if((Histogramout>0)&&(Histogramout<=1)) H0++;
        else if ((Histogramout>1)&&(Histogramout<=2)) H1++;
        else if ((Histogramout>2)&&(Histogramout<=3)) H2++;
        else if ((Histogramout>3)&&(Histogramout<=4)) H3++;
        else if ((Histogramout>4)&&(Histogramout<=5)) H4++;
        else if ((Histogramout>5)&&(Histogramout<=6)) H5++;
        else if ((Histogramout>6)&&(Histogramout<=7)) H6++;
        else if ((Histogramout>7)&&(Histogramout<=8)) H7++;
        else if ((Histogramout>8)&&(Histogramout<=9)) H8++;
        else if ((Histogramout>9)&&(Histogramout<=10)) H9++;
        else if ((Histogramout>10)&&(Histogramout<=11)) H10++;
        else if ((Histogramout>11)&&(Histogramout<=12)) H11++;
        else if ((Histogramout>12)&&(Histogramout<=13)) H12++;
        else if ((Histogramout>13)&&(Histogramout<=14)) H13++;
        else if ((Histogramout>14)&&(Histogramout<=15)) H14++;
        else if ((Histogramout>15)&&(Histogramout<=16)) H15++;
        else if ((Histogramout>16)&&(Histogramout<=17)) H16++;
        else if ((Histogramout>17)&&(Histogramout<=18)) H17++;
    }
}
System.out.println();
}
```

ซึ่งมีวิธีการทำงานภายในคือ นำค่าสีทั้ง 3 สี คือสีแดง , สีเขียว และสีน้ำเงินของแต่ละ
พิกเซลมาคำนวณค่าระดับสีจาก

```
float Histogramout = rgb2hsv(pixels[row][col][1],pixels[row][col][2],pixels[row][col][3]);
```

จากนั้นนำค่าที่คำนวณได้มาเทียบว่าอยู่ในระดับไหน แล้วทำการเพิ่มค่าที่ระดับนั้น
เมื่อทำครบทั้งรูปภาพแล้วจะได้ จำนวนพิกเซลในแต่ละระดับสีเป็นค่าฮิสโตแกรมของรูปภาพนั้น
เพื่อจะจัดเก็บค่าดังกล่าว ใน Property histogram ของคลาส HSV

จาก `equalImage(filename As %String)` จะใช้ในการค้นหารูปภาพโดยรับไฟล์รูปภาพที่ต้องการหาข้อมูล โดยแสดงรายละเอียดได้ดังนี้

```
Method ImageIcon[] equalImage (ImageIcon icon,int percent)throws
    SQLException,NumberFormatException,IOException {
    Histogram(image); // นำค่ารูปภาพที่ต้องการค้นหาทำการหาค่า ฮิสโตแกรม

    // initial ImageIcon เป็นอะเรย์สำหรับเก็บรูปภาพที่ค้นหาได้ทั้งหมด 6 รูปภาพ
    ImageIcon[] imageout = new ImageIcon[6];
    ....
    // ค้นหา pixel รวมของภาพที่ต้องการค้นหา
    int numpixel = 0;
    for (int r=0; r < MHistogram.length; r++)
        numpixel = numpixel + MHistogram[r]; //Pixel ใน Image

    // ทำการติดต่อกับฐานข้อมูล
    try{
        String url = "jdbc:Cache://localhost:1972/USER";
        String username = "_SYSTEM";
        String password = "SYS";
        Database dbconnection = CacheDatabase.getDatabase(url,username,password);

        // ทำการวนค้นหารูปภาพ จากคลาส HSV โดยค้นหาทีละรูปภาพ
        CacheQuery cq = new CacheQuery(dbconnection,"MyImages.HSV","byAll");
        ResultSet rs = cq.execute();
        ...

        // จำนวนอัตราความเหมือนของภาพที่เป็น Model ะภาพที่อยู่ในคาต้าเบสแต่ละค่า
        double checkequal = checkSame(MHistogram,test.getHistograms(),numpixel);
        per = checkequal * 100; // เปอร์เซนต์ความเหมือน
        ...
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ถ้าเปอร์เซ็นต์ความเหมือนมากกว่า 70 จะทำการแสดงผล
if((per >= 70)){
...
// นำรูปภาพจากฐานข้อมูลขึ้นมาแสดงผล
com.intersys.objects.CacheInputStream p2 = test.getPictureIn();
byte[] binarybuffer = new byte[p2.available()];
for (position = 1; position <= p2.available(); position += p2.available() )
    bytesRead = p2.read(binarybuffer);

// เรียงลำดับความเหมือนของภาพที่จะแสดงผลทั้งหมด 6 ภาพจากมากไปหาน้อย
for(int j=0;j<6;j++){
    if(per>percents[j]){
        for(int k=5;k>j;k--){
            percents[k]~percents[k-1];
            imageout[k] = imageout[k-1];
            outOid[k] = outOid[k-1];
        }
        percents[j]=per;
        imageout[j]= new ImageIcon(binarybuffer);
        outOid[j] = i;
        j = 6;
    }
}

test._save();
dbconnection.closeObject(test.getOref()); // ปิด connection กับฐานข้อมูล
}

....
// ส่งผลรูปภาพที่มีความเหมือน 6 รูปภาพคืนเพื่อที่จะแสดงผล
return imageout;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วิธีที่ 2 การจัดเก็บและค้นหารูปภาพโดยใช้การแบ่งรูปภาพออกเป็น 5 ส่วนก่อนทำการหาค่าฮิสโตแกรมของพิกเซลในแต่ละส่วน แล้วจึงทำการจัดเก็บลงในฐานข้อมูล

```

Class MyImages.SegmentHSV Extends MyImages.AppImage
{
Property Part1 As list Of %Integer;
Property Part2 As list Of %Integer;
Property Part3 As list Of %Integer;
Property Part4 As list Of %Integer;
Property Part5 As list Of %Integer;
Projection MyTestImagr As %Projection.Java(ROOTDIR = "C:\Project");
Query byid(id As %Integer) As %SQLQuery(CONTAINID = 1)
{
    SELECT %ID FROM SegmentHSV
    WHERE (%ID = :id)
}
Query byname(name As %String) As %SQLQuery(CONTAINID = 1)
{
    SELECT %ID FROM SegmentHSV
    WHERE (Name = :name)
}
Query byAll() As %SQLQuery(CONTAINID = 1)
{
    SELECT %ID FROM SegmentHSV
}
Method Histogram() [ Language = java ]
{
}

Method equalImage(filename As %String) [ Language = java ]
{
}
}

```

จากโค้ดโปรแกรมข้างต้น เป็นโค้ดที่เขียนภายในฐานข้อมูล cache โดย คลาส SegmentHSV เป็นคลาสที่อยู่ภายในแพคเกจ MyImages โดยทำการสืบทอดคุณสมบัติจากคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ApplImage เพื่อใช้ในการจัดเก็บรูปภาพ และเพิ่มเติมคุณสมบัติบ้างประการเพื่อใช้ในการค้นหารูปภาพ โดยภายในคลาส SegmentHSV ของฐานข้อมูลได้เพิ่มเติมในส่วนของการจัดเก็บข้อมูลดังต่อไปนี้

- คุณสมบัติ (Property) Part1 , Part2 , Part3 , Part4 , Part5 ซึ่งชนิดของข้อมูลเป็น ลิสต์ของชนิดข้อมูลตัวเลข (list Of %Integer) โดยจะใช้ในการจัดเก็บค่าฮิสโตแกรม ของทุกพิกเซลในรูปภาพทั้ง 5 ส่วน โดยสามารถคำนวณหาค่าฮิสโตแกรมได้จาก Method Histogram()
- Query byid(id As %Integer) เป็นค้นหาข้อมูลจากการใช้ id
- Query byname(name As %String) เป็นค้นหาข้อมูลจากการใช้ ชื่อของรูปภาพในการ ค้นหา
- Query byAll() เป็นค้นหาข้อมูลทั้งหมด

จากเมธอด Histogram() เป็นการคำนวณค่าฮิสโตแกรมของรูปภาพที่ต้องการการจัดเก็บ ลงในฐานข้อมูล โดยจะมีการทำงานเช่นเดียวกับ คลาส HSV แต่จะเพิ่มเติมคือต้องมีการแบ่งรูปภาพ ออกเป็น 5 ส่วนก่อนแล้วนำแต่ละส่วนมาทำการหาค่าฮิสโตแกรมของทั้ง 5 ส่วน โดยแสดง รายละเอียดได้ดังนี้

```

for (int row=0; row < HPart*2; row+=HPart) {
    for (int col=0; col < WPart*2; col+=WPart) {
        for (int i = 0; i < HPart; i++){
            for (int j = 0; j < WPart; j++){
                float Histogramout = rgb2hsv(pixels[row+i][col+j][1],pixels[row+i][
                if ((Histogramout>0) && (Histogramout<=1)) H0++;
                else if ((Histogramout>1) && (Histogramout<=2)) H1++;
                else if ((Histogramout>2) && (Histogramout<=3)) H2++;
                else if ((Histogramout>3) && (Histogramout<=4)) H3++;
                else if ((Histogramout>4) && (Histogramout<=5)) H4++;
                else if ((Histogramout>5) && (Histogramout<=6)) H5++;
                else if ((Histogramout>6) && (Histogramout<=7)) H6++;
                else if ((Histogramout>7) && (Histogramout<=8)) H7++;
                else if ((Histogramout>8) && (Histogramout<=9)) H8++;
                else if ((Histogramout>9) && (Histogramout<=10)) H9++;
                else if ((Histogramout>10) && (Histogramout<=11)) H10++;
                else if ((Histogramout>11) && (Histogramout<=12)) H11++;
                else if ((Histogramout>12) && (Histogramout<=13)) H12++;
                else if ((Histogramout>13) && (Histogramout<=14)) H13++;
                else if ((Histogramout>14) && (Histogramout<=15)) H14++;
                else if ((Histogramout>15) && (Histogramout<=16)) H15++;
                else if ((Histogramout>16) && (Histogramout<=17)) H16++;
                else if ((Histogramout>17) && (Histogramout<=18)) H17++;
            }
        }
    }
}

```

ซึ่งมีวิธีการทำงานภายในคือ นำค่าสีทั้ง 3 สี คือสีแดง , สีเขียว และสีน้ำเงินของแต่ละ พิกเซลมาคำนวณค่าระดับสีจาก

```
float Histogramout = rgb2hsv(pixels[row][col][1],pixels[row][col][2],pixels[row][col][3]);
```

จากนั้นนำค่าที่คำนวณได้มาเทียบว่าอยู่ในระดับไหนแล้วทำการเพิ่มค่าที่ระดับสีนั้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำครบทั้งส่วนนั้นแล้วจะได้ จำนวนพิกเซลในแต่ละระดับสีเป็นค่าฮิสโตแกรมของส่วนนั้นเพื่อจะจัดเก็บค่าดังกล่าว ใน Property Part1 , Part2 , Part3 , Part4 , Part5 ของคลาส SegmentHSV

จาก equalImage(filename As %String) จะใช้ในการค้นหาภาพโดยรับไฟล์รูปภาพที่ต้องการหาข้อมูล โดยแสดงรายละเอียดได้ดังนี้

```
Method ImageIcon[] equalImage (ImageIcon icon,int percent)throws
    SQLException,NumberFormatException,IOException {
//นำค่ารูปภาพที่ต้องการค้นหาทำการหาค่า ฮิสโตแกรมทั้ง 5 ส่วน
int [] Part = Histogram(image);
histograms1 = Part[0];
histograms2 = Part[2];
histograms3 = Part[3];
histograms4 = Part[4];
histograms5 = Part[5];

// initial ImageIcon เป็นอาร์เรย์สำหรับเก็บรูปภาพที่ค้นหาได้ทั้งหมด 6 รูปภาพ
ImageIcon[] imageout = new ImageIcon[6];
...
...
try{
String url = "jdbc:Cache://localhost:1972/USER";
String username = "_SYSTEM";
String password = "SYS";
Database dbconnection = CacheDatabase.getDatabase(url,username,password);

// ทำการวนค้นหาภาพ จากคลาส SegmentHSV โดยค้นหาทีละรูปภาพ
CacheQuery cq = new CacheQuery(dbconnection,"MyImages. SegmentHSV
","byAll");

ResultSet rs = cq.execute();
...
// คำนวณอัตราความเหมือนของภาพที่เป็น Model ะภาพที่อยู่ในดาต้าเบสแต่ละค่า โดย
คำนวณในแต่ละส่วน แล้วนำมาหาค่าเฉลี่ย
double type1P1 = checkSame(MPart1,test.getPart1(),numpixelPart1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

double type1p2 = checkSame(MPart2,test.getPart2(),numpixelPart2);
double type1p3 = checkSame(MPart3,test.getPart3(),numpixelPart3);
double type1p4 = checkSame(MPart4,test.getPart4(),numpixelPart4);
double type1p5 = checkSame(MPart5,test.getPart5(),numpixelPart5);
per = ((type1P1+type1p2+type1p3+type1p4+type1p5)*100)/5; // ความเหมือน
...
...
if(per >= 70){           // ถ้าเปอร์เซ็นต์ความเหมือนมากกว่า 70 จะทำการแสดงผล
...

// นำรูปภาพจากฐานข้อมูลขึ้นมาแสดงผล
com.intersys.objects.CacheInputStream p2 = test.getPictureIn();
byte[] binarybuffer = new byte[p2.available()];
for (position = 1; position <= p2.available(); position += p2.available() )
    bytesRead = p2.read(binarybuffer);
// เรียงลำดับความเหมือนของภาพที่จะแสดงผลทั้งหมด 6 ภาพจากมาก ไปหาน้อย
for(int j=0;j<6;j++){
    if(per>percents[j]){
        for(int k=5;k>j;k--){
            percents[k]=percents[k-1];
            imageout[k] = imageout[k-1];
            outOid[k] = outOid[k-1];
        }
        percents[j]=per;
        imageout[j]= new ImageIcon(binarybuffer);
        outOid[j] = i;
        j = 6;
    }
}
}

test._save();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        dbconnection.closeObject(test.getOref()); // ปิดการติดต่อกับฐานข้อมูล
    }
    ....
    // ส่งผลรูปภาพที่มีความเหมือน 6 รูปภาพคืนเพื่อที่จะแสดงผล
    return imageout;
}

```

- วิธีที่ 3 การจัดเก็บและค้นหารูปภาพประเภทหน้าคนโดยใช้ Grayscale โดยการคำนวณหาค่าที่จะใช้แทนรูปภาพนั้น แล้วจึงทำการจัดเก็บลงในฐานข้อมูล

```

Class MyImages.MyGrayscale Extends MyImages.ApplImage
{

```

```

Property Level0 As %String;

```

```

Property Level1 As %String;

```

```

Property Level2 As %String;

```

```

Property Level3 As %String;

```

```

Property Level4 As %String;

```

```

Property Level5 As %String;

```

```

Property Level6 As %String;

```

```

Property Level7 As %String;

```

```

Property Level8 As %String;

```

```

Property Level9 As %String;

```

```

Projection Pjt As %Projection.Java(ROOTDIR = "C:\Project");

```

```

Query byid(id As %Integer) As %SQLQuery(CONTAINID = 1)

```

```

{
    SELECT %ID FROM MyGrayscale
    WHERE (%ID = :id)
}

```

```

Query byname(name As %String) As %SQLQuery(CONTAINID = 1)

```

```

{
    SELECT %ID FROM MyGrayscale
    WHERE (Name = :name)
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Query byAll() As %SQLQuery(CONTAINID = i)
{
    SELECT %ID FROM MyGrayscale
}

Method Histogram() [ Language = java ]
{
}

Method equalImage(filename As %String) [ Language = java ]
{
}
}

```

จากโค้ด โปรแกรมข้างต้น เป็นโค้ดที่เขียนภายในฐานข้อมูล Caché โดย คลาส MyGrayscale เป็นคลาสที่อยู่ภายในแพคเกจ MyImages โดยทำการสืบทอดคุณสมบัติจากคลาส AppImage เพื่อใช้ในการจัดเก็บรูปภาพ และเพิ่มเติมคุณสมบัติบ้างประการเพื่อใช้ในการค้นหารูปภาพ โดยภายในคลาส MyGrayscale ของฐานข้อมูลได้เพิ่มเติมในส่วนของการจัดเก็บข้อมูลดังต่อไปนี้

- คุณสมบัติ (Property) Level0, Level1, Level2, Level3, Level4, Level5, Level6, Level7, Level8 Level9 ซึ่งชนิดของข้อมูลเป็นข้อความ (%String) โดยจะใช้ในการจัดเก็บค่าตัวแทนของรูปภาพที่ต้องการจัดเก็บ โดยค่าที่ใช้แทนรูปภาพสามารถหาได้จาก โค้ดจาก Method Histogram()
- Query byid(id As %Integer) เป็นค้นหาข้อมูลจากการใช้ id
- Query byname(name As %String) เป็นค้นหาข้อมูลจากการใช้ ชื่อของรูปภาพในการค้นหา
- Query byAll() เป็นค้นหาข้อมูลทั้งหมด

จากเมธอด Histogram() เป็นการคำนวณค่าตัวแทนของรูปภาพที่ต้องการการจัดเก็บลงในฐานข้อมูล โดยแสดงรายละเอียดได้ดังนี้

```

public static void Histogram (Image image) {
    ...
    Width = Imagei.getIconWidth();
    Hight = Imagei.getIconHeight();
    // วิธีนี้ต้องการแบ่งรูปภาพเป็นขนาด 10*10 เพราะฉะนั้นต้องทำการหาว่าในแต่ละช่องอาร์เรย์นั้น
    รูปภาพจะมีขนาดเท่าไร โดยการหารด้วย 10 ทั้งความกว้างและความสูง
    WGray = Width/10;
    HGray = Hight/10;
    ...
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

// อ่านค่ารูปภาพที่ได้ทำการตัดแบ่งเป็น 10*10 ช่องแล้วโดยนำมาทำทีละช่องและอ่านค่าที่ละพิกเซลที่อยู่ในช่องนั้น เพื่อนำมาหาค่าเฉลี่ย และทำการจัดเก็บค่าเฉลี่ยนั้นลงในอาร์เรย์ค่าตัวแทนรูปภาพ 10*10 ตามลำดับของช่อง

```
for (int row=0; row < HGray*10; row+=HGray) {
    for (int k=0; k < WGray*10; k+=WGray) {
        System.out.println(k);
        for(int i = 0; i < HGray; i++){
            for(int j = 0; j < WGray; j++){
                sum+= (0.299*pixels[i+row][j+k][1]) +
                    (0.587*pixels[i+row][j+k][2]) + 0.114*pixels[i+row][j+k][3];
            }
        }
        ...
        Compare[m][n] = sum/(WGray*HGray);
    }
}

Level0 =Level1=Level2=Level3=Level4=Level5=Level6=Level7=Level8=Level9= "";
for(int i = 0; i <10; i++){
    if(i != 9){
        Level0 += Compare[0][i]+" ";
        Level1 += Compare[1][i]+" ";
        Level2 += Compare[2][i]+" ";
        Level3 += Compare[3][i]+" ";
        Level4 += Compare[4][i]+" ";
        Level5 += Compare[5][i]+" ";
        Level6 += Compare[6][i]+" ";
        Level7 += Compare[7][i]+" ";
        Level8 += Compare[8][i]+" ";
        Level9 += Compare[9][i]+" ";
    }else{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Level0 += Compare[0][i];
Level1 += Compare[1][i];
Level2 += Compare[2][i];
Level3 += Compare[3][i];
Level4 += Compare[4][i];
Level5 += Compare[5][i];
Level6 += Compare[6][i];
Level7 += Compare[7][i];
Level8 += Compare[8][i];
Level9 += Compare[9][i];
    }
}
}

```

จาก `equalImage(filename As %String)` จะใช้ในการค้นหารูปภาพโดยรับไฟล์รูปภาพที่ต้องการหาข้อมูล โดยแสดงรายละเอียดได้ดังนี้

```

public static ImageIcon[] equalImage (ImageIcon icon)throws
SQLException,NumberFormatException,IOException {
    Image image = icon.getImage();
    // หาค่าตัวแทนรูปภาพของรูปภาพที่ต้องการค้นหาจากเมทอด Histogram
    Histogram(image);
    // initial ImageIcon เป็นอาร์เรย์สำหรับเก็บรูปภาพที่ค้นหาได้ทั้งหมด 6 รูปภาพ
    ImageIcon[] imageout = new ImageIcon[6];
    for(int i = 0;i<6;i++){
        imageout[i] = null;
    }
    // อ่านค่าตัวแทนรูปภาพจาก String เป็น Integer
    int[] Row0 = stringToArray(Level0);
    int[] Row1 = stringToArray(Level1);
    ...
    int[] Row9 = stringToArray(Level9);

    try{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ติดต่อกับฐานข้อมูล และนำรูปที่อยู่ในฐานข้อมูลมาคำนวณหาค่าความเหมือน
String url = "jdbc:Cache://localhost:1972/USER";
String username = "_SYSTEM";
String password = "SYS";
Database dbconnection =
    CacheDatabase.getDatabase(url,username,password);
...
CacheQuery cq = new
    CacheQuery(dbconnection,"MyImages.MyGrayscale","byAll");
ResultSet rs = cq.execute();
boolean equal = false;
...
}
while(rs.next()){
    int i = Integer.parseInt(rs.getString(1));
    Oid oid = new Oid(i);
    MyImages.MyGrayscale test = (MyImages.MyGrayscale)
        MyImages.MyGrayscale._open(dbconnection, oid);
// อ่านค่าตัวแทนรูปภาพของรูปภาพที่อยู่ในฐานข้อมูล ลงในอาร์เรย์ชนิด Integer
int[] ARow0 = stringToIntArray(test.getLevel0());
...
int[] ARow9 = stringToIntArray(test.getLevel9());

...
// นำค่าในแต่ละช่องอาร์เรย์มาหาค่าเปอร์เซ็นต์ความเหมือนของแต่ละช่องอาร์เรย์โดยเทียบแต่ละ
รูปภาพในฐานข้อมูลกับภาพที่นำมาทำการค้นหา
for (int r=0; r < I0; r++){
    perc[0][r]= (255-(Row0[r]-ARow0[r]))*100.0/255.0;
    perc[1][r]= (255-(Row1[r]-ARow1[r]))/255.0*100.0;
    ...
    perc[9][r]= (255-(Row9[r]-ARow9[r]))/255.0*100.0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

...
// จากนั้นทำการหาเปอร์เซ็นต์ความเหมือนรวมของทั้งรูปภาพ
for(int d=0;d<10;d++){
    for(int e=0;e<10;e++){
        if(perc[d][e]>=90.0 && perc[d][e]<=110.0)
            counta++;
        else if (perc[d][e]>=80.0&&perc[d][e]<90.0)
            countb++;
        else if (perc[d][e]>110.0 && perc[d][e]<=120)
            countb++;
        else countc++;
        sum += perc[d][e];
    }
}
per = sum/100.0;
per = ((100.0*counta) + (90.0*countb) + (50.0*countc))/100.0;
// กำหนดค่าเปอร์เซ็นต์ความเหมือนที่เรายอมรับได้ว่ารูปภาพจะมีความคล้ายกัน
if(per >= 90.0){//(percents >85) && (percents < 120)}{
    equal = true;
    ...
// อ่านค่ารูปภาพจากฐานข้อมูลลงในตัวแปร ImageIcon เพื่อไปทำการแสดงผล
ต่อไป
com.intersys.objects.CacheInputStream p2 = test.getPictureIn();
byte[] binarybuffer = new byte[p2.available()];
int position ,bytesRead;
for (position = 1; position <= p2.available()*400; position +=
    p2.available() ) {
        bytesRead = p2.read(binarybuffer);
    }
//นำรูปภาพมาจัดลำดับความเหมือน 6 ลำดับจากมากไปน้อย
for(int j=0;j<6;j++){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(per>percents[j]){
    for(int k=5;k>j;k--){
        percents[k]=percents[k-1];
        imageout[k] = imageout[k-1];
        outOid[k] = outOid[k-1];
    }
    percents[j]=per;
    imageout[j]= new ImageIcon(binarybuffer);
    outOid[j] = i;
    j = 6;
}
}
// จบการทำงานติดต่อกับฐานข้อมูล
test._save();
dbconnection.closeObject(test.getOref());
test = null;
}
}
// กรณีถ้าไม่มีรูปใดเหมือนหรือคล้าย
if(!equal){
    System.out.println("No Match");
} else System.out.println("Match");
}
}
...
// ส่งผลรูปภาพที่มีความเหมือน 6 รูปภาพคืนเพื่อที่จะแสดงผล
return imageout;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในโปรแกรมภาษา จาวา

1. การจัดเก็บรูปภาพลงในฐานข้อมูล

```

public void AddImage (String pathFile,String namePic) throws IOException,
                                NumberFormatException, SQLException{

try{

String url = "jdbc:Cache://localhost:1972/USER";
String username = "_SYSTEM";
String password = "SYS";

// ทำการติดต่อกับฐานข้อมูลเชิงวัตถุ Cache
Database dbconnection = CacheDatabase.getDatabase(url,username,password);

// ทำการสร้างออบเจกต์ของคลาสภายในฐานข้อมูลที่จะจัดเก็บข้อมูล
MyImages.HSV pic = new MyImages.HSV(dbconnection);
pic.setName(namePic); // ทำการ เพิ่มข้อมูล Propert Name ผ่าน Method setName
String inputBinaryFileName = pathFile; // Path ของไฟล์รูปภาพที่ต้องการ เพิ่มลงในฐานข้อมูล
CacheOutputStream p1 = pic.getPictureOut(); //จัดเก็บรูปภาพโดยผ่าน Method getPictureOut()

File inputBinaryFile = new File(inputBinaryFileName);
FileInputStream inputFileInputStream = new FileInputStream(inputBinaryFile);

byte[] binarybuffer = new byte[inputFileInputStream.available()];
int bytesRead = 0;
while((bytesRead = inputFileInputStream.read(binarybuffer)) != -1){
    p1.write(binarybuffer); // เขียนลงในฐานข้อมูล
}

Image image = MyImages.HSV.readImageFile(inputBinaryFileName);
MyImages.HSV.Histogram(image); // คำนวณค่า Histogram ของรูปภาพที่ต้องการจัดเก็บ
for(int i = 0;i<18;i++){
    pic.gethistograms().add(i, Histogram[i]); // จัดเก็บค่าที่คำนวณ ได้ลงในฐานข้อมูล
}
}
}

```

```

// นำภาพที่ต้องการจัดเก็บมาคำนวณหาค่า BitmapInfoHeader ผ่านเมธอด findBitmapInfoHeader
    MyImages.BitmapInfoHeader InfoHeader =
        MyImages.BitmapInfoHeader.findBitmapInfoHeader(dbconnection,image,(new
        ImageIcon(image)).getIconWidth(),(new ImageIcon(image)).getIconHeight());

// นำภาพที่ต้องการจัดเก็บมาคำนวณหาค่า BitmapFileHeader ผ่าน Method findBitmapFileHeader
    MyImages.BitmapFileHeader fileHeader =
        MyImages.BitmapFileHeader.findBitmapFileHeader(dbconnection,image,(new
        ImageIcon(image)).getIconWidth(),(new ImageIcon(image)).getIconHeight());

// ทำการ Add ค่า BitmapInfoHeader และค่า BitmapFileHeader ผ่าน Method setBitmapInfoHeader และ
setBitmapFileHeader ตามลำดับ
    pic.setBitmapInfoHeader(InfoHeader);
    pic.setBitmapFileHeader(fileHeader);

// ทำการ save
    pic._save();

//ปิดการเชื่อมต่อ
    dbconnection.closeObject(InfoHeader.getOref());
    dbconnection.closeObject(fileHeader.getOref());
    dbconnection.close();
}
catch (CacheException ex){
    System.out.println("Caught exception: "+ex.getClass().getName() +
    " : "+ex.getMessage());
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การค้นหารูปภาพภายในในฐานข้อมูล

// initial array ของ image เพื่อรองรับค่าที่ได้จากการค้นหาในฐานข้อมูล

```
ImageIcon[] imageicon = null;
```

// ทำการเปลี่ยนชื่อไฟล์รูปภาพเป็นรูปภาพที่จะให้ในการค้นหาภายในฐานข้อมูล

```
ImageIcon imagei = new
```

```
ImageIcon(MyImages.HSV.readImageFile(fc.getSelectedFile().getAbsolutePath()));
```

// ทำการค้นหารูปภาพ โดยใช้เมธอด `equalImage(imagei)` โดยนำ รูปที่ต้องการค้นหาส่งเป็นพารามิเตอร์โดยการทำงานภายในเมธอด `equalImage` ได้กล่าวมาแล้วในข้างต้น

```
imageicon = MyImages.HSV.equalImage(imagei);
```

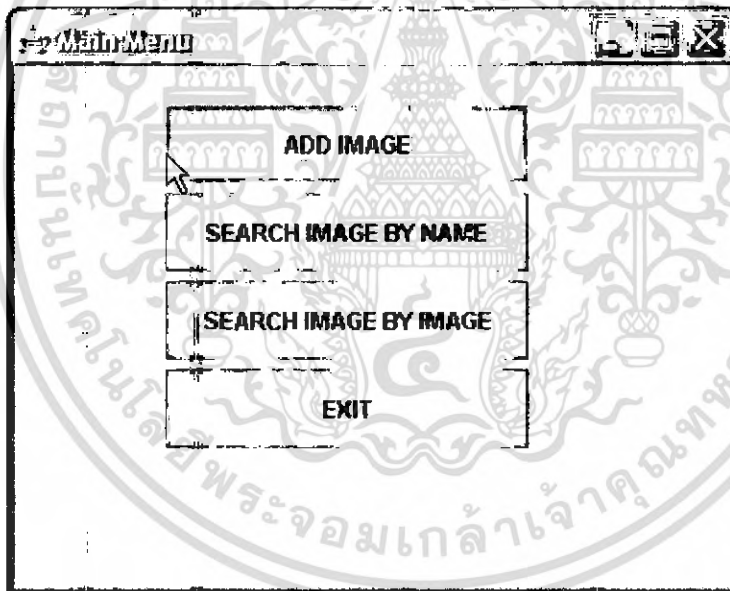
เมื่อได้รูปที่ได้จากการค้นหาแล้วจะทำการแสดงผล



ภาคผนวก ง. การใช้งานโปรแกรมที่พัฒนาขึ้น

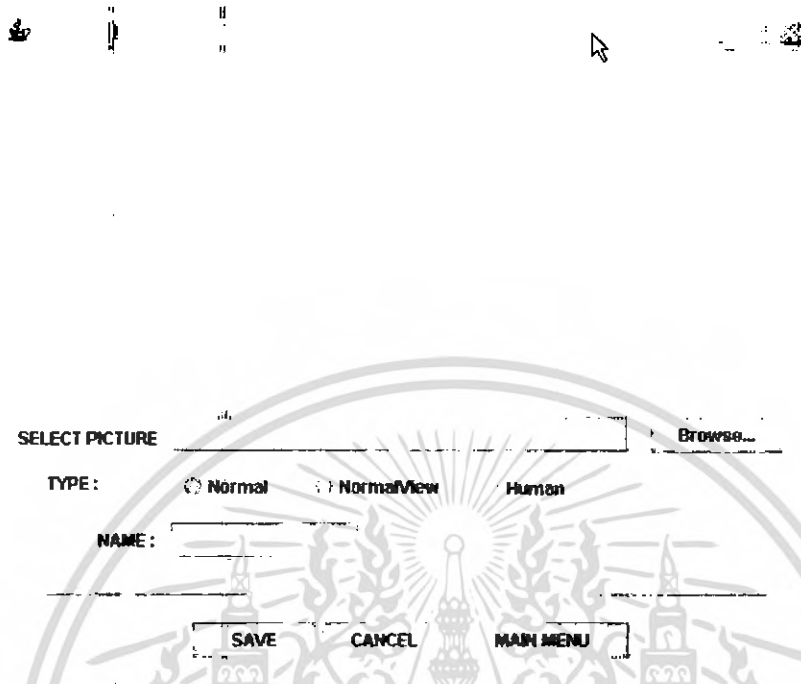
เมื่อเริ่มต้นการใช้งาน จะเข้าสู่หน้าจอหลักให้ผู้ใช้งานเลือกการทำงาน ดังแสดงในรูปที่ ง-1 ซึ่งมีรายละเอียดการทำงานคือ

- ปุ่ม ADD IMAGE เมื่อต้องการทำการเพิ่มรูปภาพลงในฐานข้อมูล
- ปุ่ม SEARCH IMAGE BY NAME เมื่อต้องการทำการค้นหารูปภาพโดยใช้ชื่อของรูปภาพในการค้นหา
- ปุ่ม SEARCH IMAGE BY IMAGE เมื่อต้องการทำการค้นหารูปภาพโดยนำรูปภาพเข้าไปค้นหาในฐานข้อมูล
- ปุ่ม EXIT เมื่อต้องการออกจากโปรแกรม



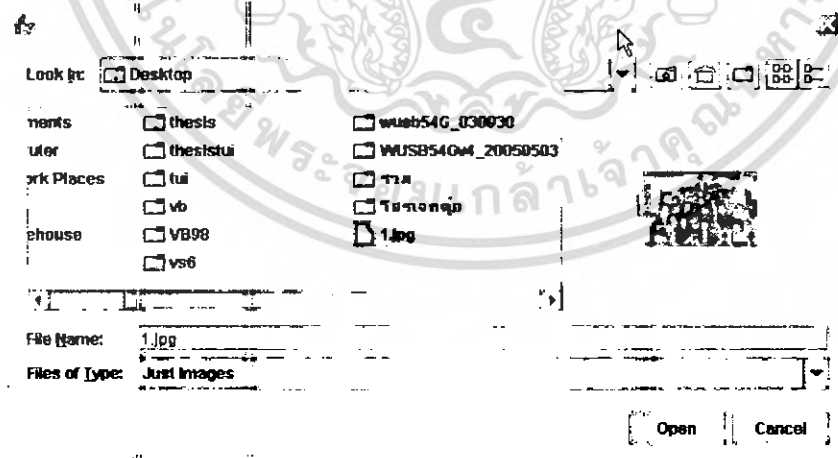
รูปที่ ง-1 หน้าจอหลักในการทำงาน

- เมื่อคลิกปุ่ม ADD IMAGE เพื่อต้องการทำการเพิ่มรูปภาพลงในฐานข้อมูลจะ
แสดงหน้าดังรูปที่ ง-2



รูปที่ ง-2 หน้าจอในการเพิ่มข้อมูลรูปภาพ

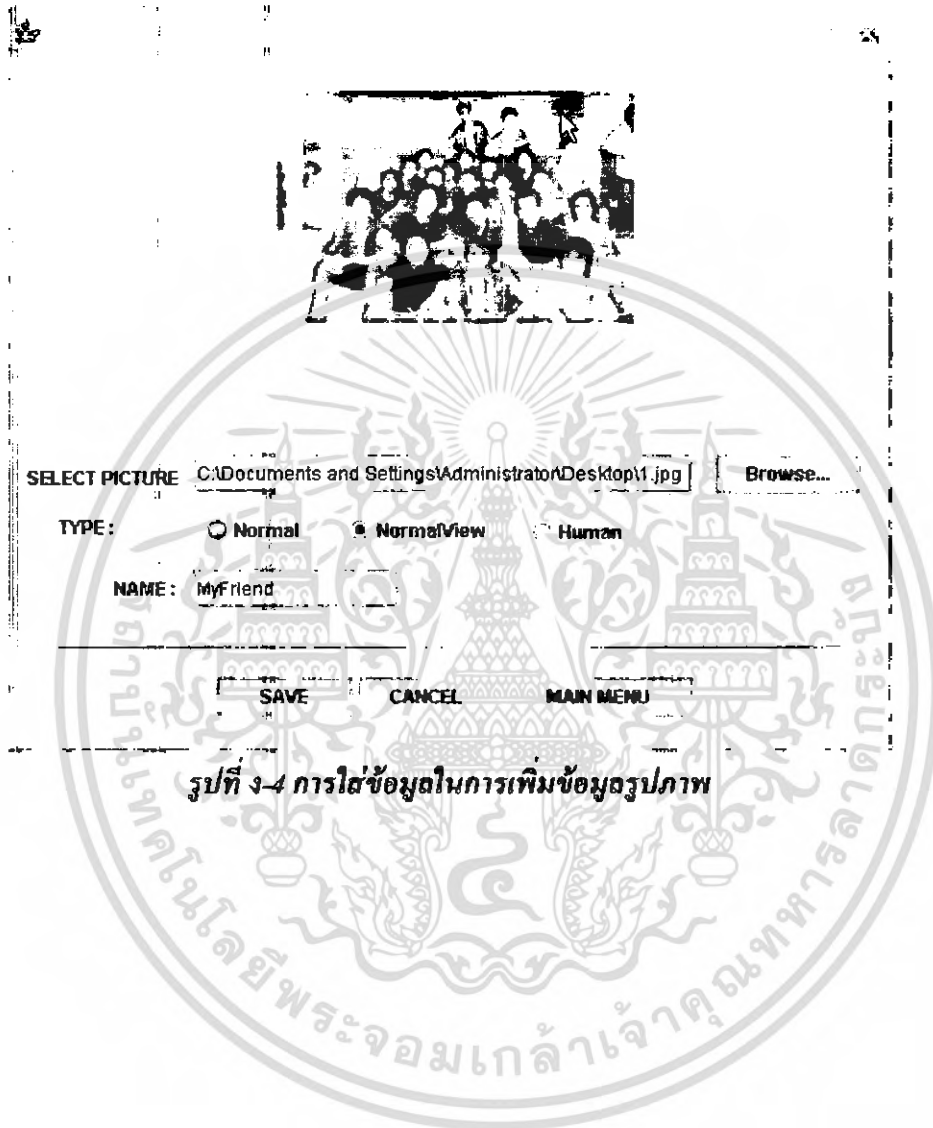
- เมื่อผู้ใช้ต้องการเพิ่มข้อมูลในฐานข้อมูลจะทำการ กดปุ่ม Browse.... เพื่อเลือกรูปที่
ต้องการจะจัดเก็บลงในฐานข้อมูลดังรูปที่ ง-3



รูปที่ ง-3 หน้าจอการเลือกรูปที่ต้องการจัดเก็บลงในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

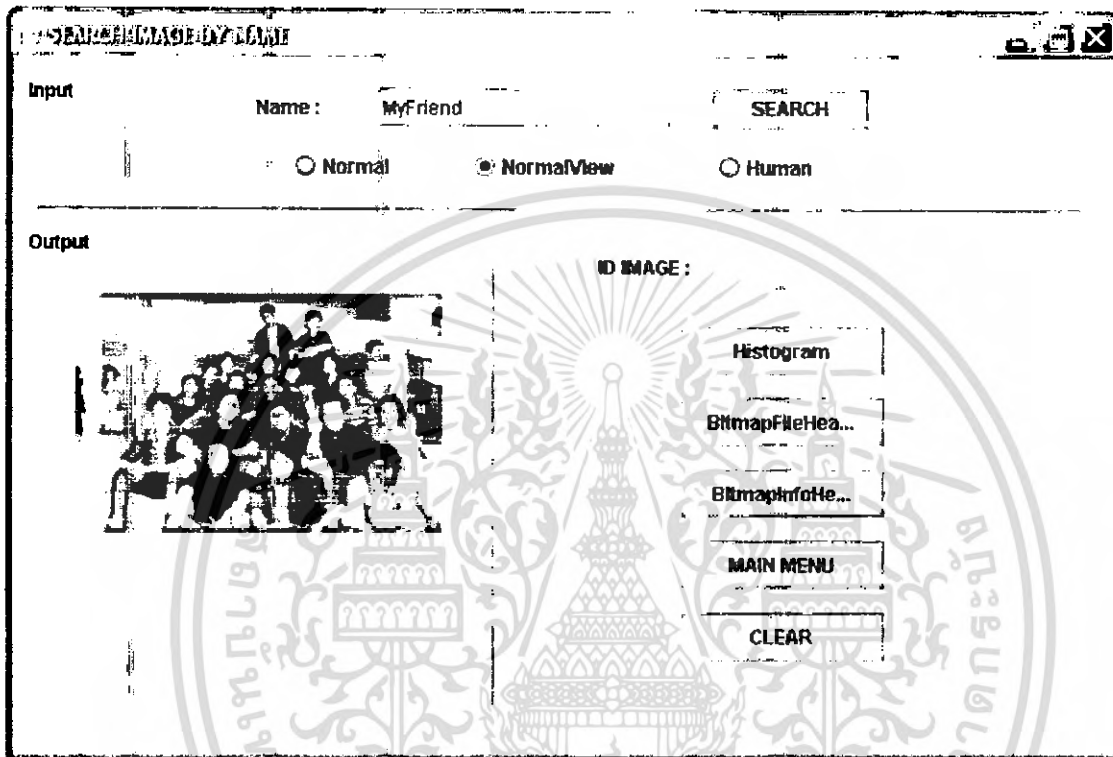
เมื่อทำการเลือกรูปที่ต้องการเพิ่มแล้ว จะทำการใส่ข้อมูลเพิ่มเติม คือชื่อของรูปภาพ และทำการเลือกประเภทว่ารูปภาพเป็นแบบใด ดังแสดงในรูป ง-4 จากนั้นทำการกดปุ่ม SAVE ข้อมูลทั้งหมดจะถูกจัดเก็บลงในฐานข้อมูล



รูปที่ ง-4 การใส่ข้อมูลในการเพิ่มข้อมูลรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

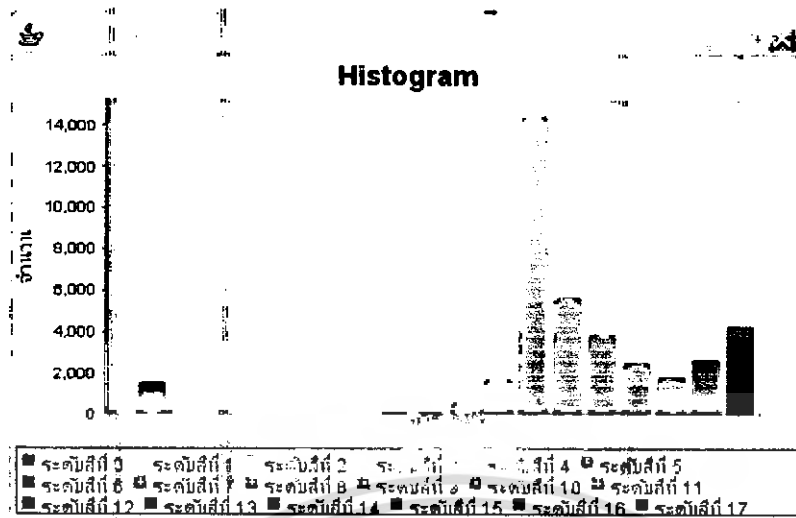
• เมื่อคลิกปุ่ม SEARCH IMAGE BY NAME ที่หน้าจอการทำงานหลัก จะแสดงหน้าจอการค้นหารูปภาพจากชื่อของรูปภาพ โดยการใส่ชื่อของรูปภาพที่ต้องการค้นหาแล้วเลือกประเภทของรูปภาพ จากนั้นทำการกดปุ่ม SEARCH เพื่อทำการค้นหารูปภาพที่มีชื่อ MyFriend ในฐานข้อมูล ดังแสดงใน รูปที่ ง-5



รูปที่ ง-5 แสดงหน้าจอการค้นหารูปภาพจากชื่อของรูปภาพ

โดยจะสามารถทำการดูค่าฮิสโตแกรม , BitmapFileHeader , BitmapInfoHeader จากฐานข้อมูล ที่การกดปุ่ม Histogram ,BitmapFileHeader ,BitmapInfoHeader ตามลำดับ โปรแกรมจะแสดงการค้นหาจากฐานข้อมูล ได้ผลดังรูป ง-6 , ง-7, ง-8 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-6 ค่า Histogram ของรูปภาพในฐานข้อมูล

BFieldType: BM

BFSIZE: 120022

BFRReserve1: 0

BFRReserve2: 0

BFOffBits: 54

OK

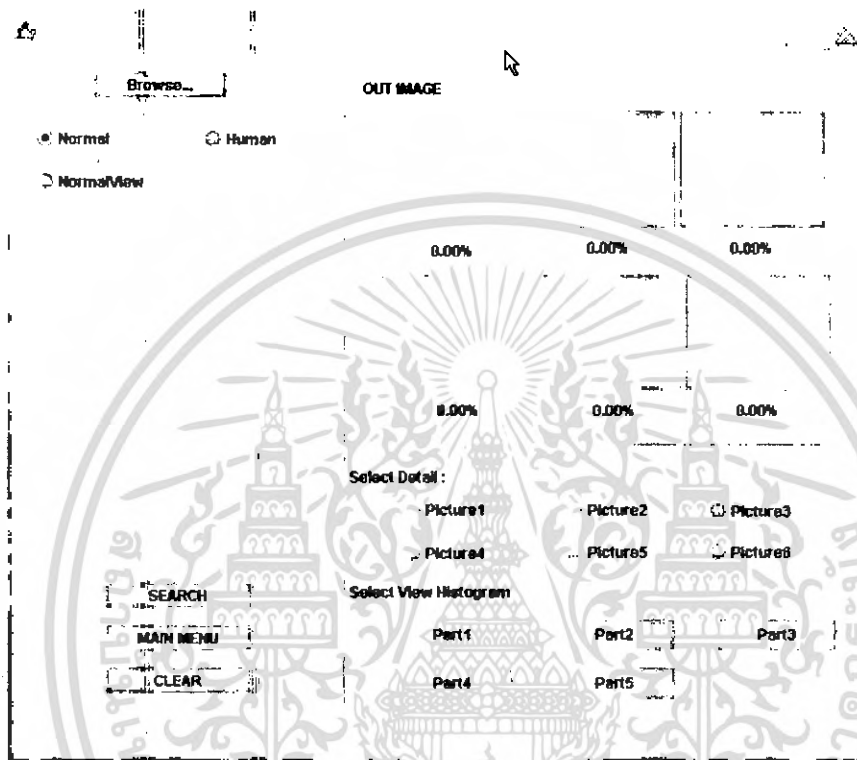
รูปที่ ง-7 ค่า BitmapFileHeader ของรูปภาพในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

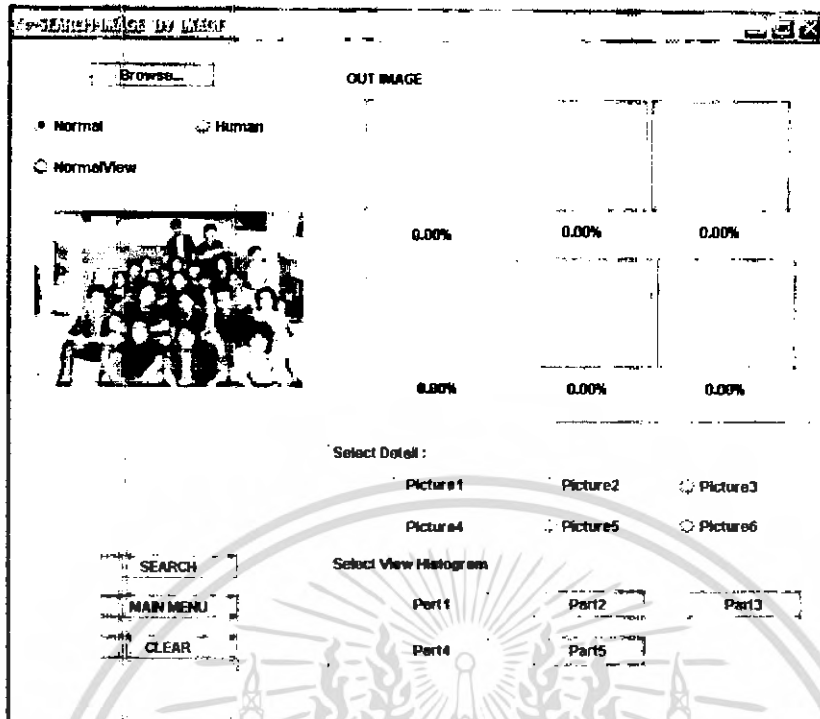
BiSize :	40
BWidth :	245
BiHeight :	163
BiPlanes :	24
BiBitCount :	1
BiCompressi...	0
BiSizeImage :	119968
BiXPelsPerMe...	0
BiYPelsMeter :	0
BiClrUsed :	0
BiClrImportant :	0
OK	

รูปที่ 3-8 แสดงค่า *BitmapInfoHeader* ของรูปภาพในฐานะข้อมูล

- เมื่อคลิกปุ่ม SEARCH IMAGE BY IMAGE ที่หน้าจอการทำงานหลัก จะแสดงหน้าจอการค้นหารูปภาพจากรูปภาพดังแสดงใน รูปที่ ๓-9 โดยทำการ Browse.. เพื่อเลือกรูปภาพที่ต้องการค้นหาแล้วเลือกประเภทของรูปภาพ จากนั้นทำการกดปุ่ม SEARCH เพื่อทำการค้นหารูปภาพในฐานข้อมูล ดังแสดงใน รูปที่ ๓-10

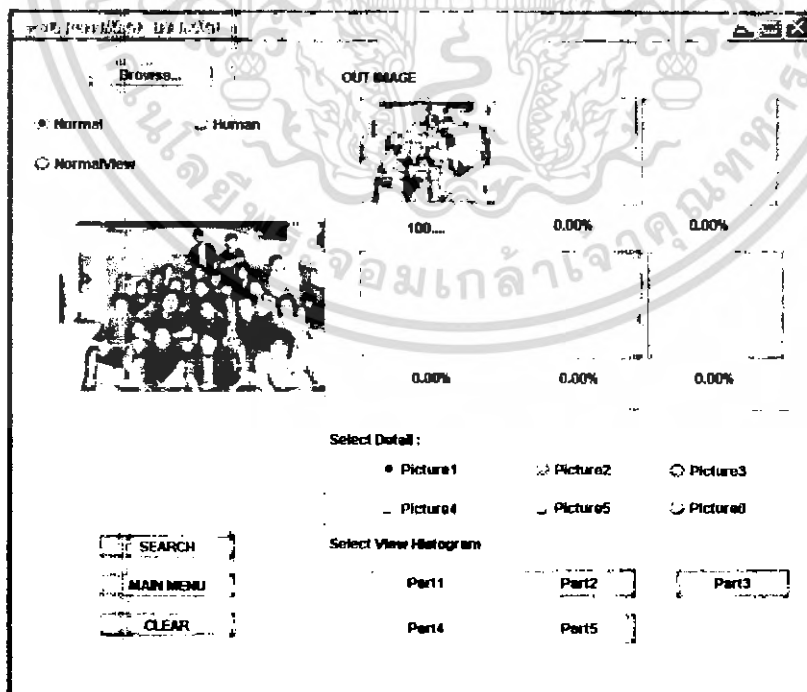


รูปที่ ๓-9 หน้าจอการค้นหารูปภาพ จากรูปภาพ



รูปที่ 3-10 การค้นหารูปภาพจากรูปภาพ

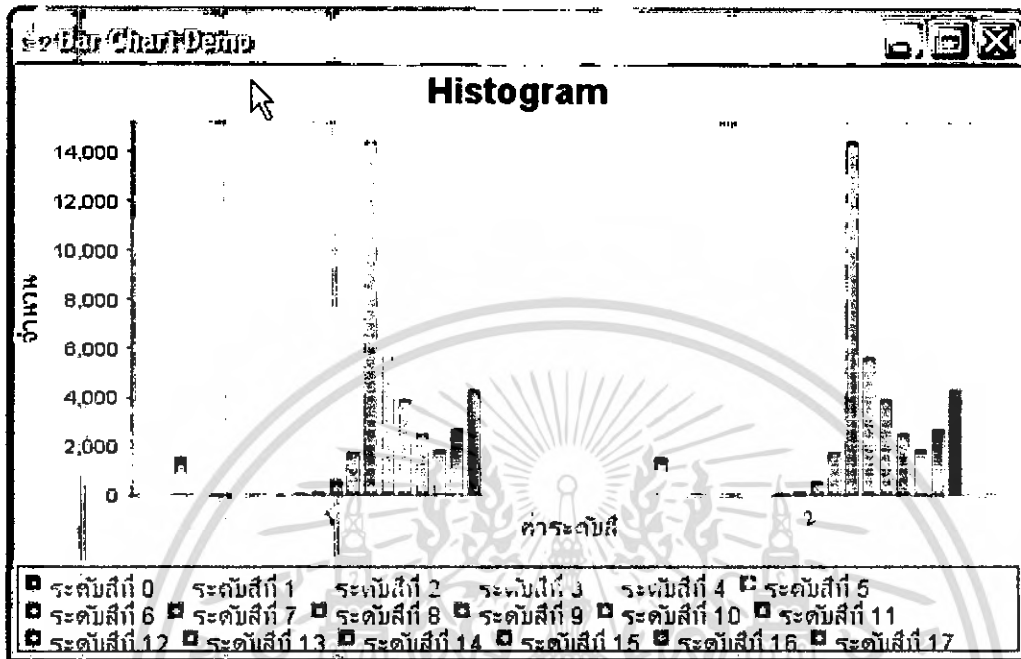
โดยผลจากการค้นหาในฐานข้อมูล แสดงในรูป 3-11 โดยได้ภาพจะเป็นเปอร์เซ็นต์ความเหมือนของภาพที่ต้องการค้นหาเทียบกับภาพในฐานข้อมูล



รูปที่ 3-11 ผลจากการค้นหาภายในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเมื่อทำการเลือกที่ Picture1 จะทำการเปรียบเทียบ ค่าฮิสโตแกรมของภาพที่ต้องการค้นหากับภาพในฐานข้อมูล เมื่อเลือก Part1 จะทำการเปรียบเทียบในส่วนที่ 1 ดังรูป ง-12



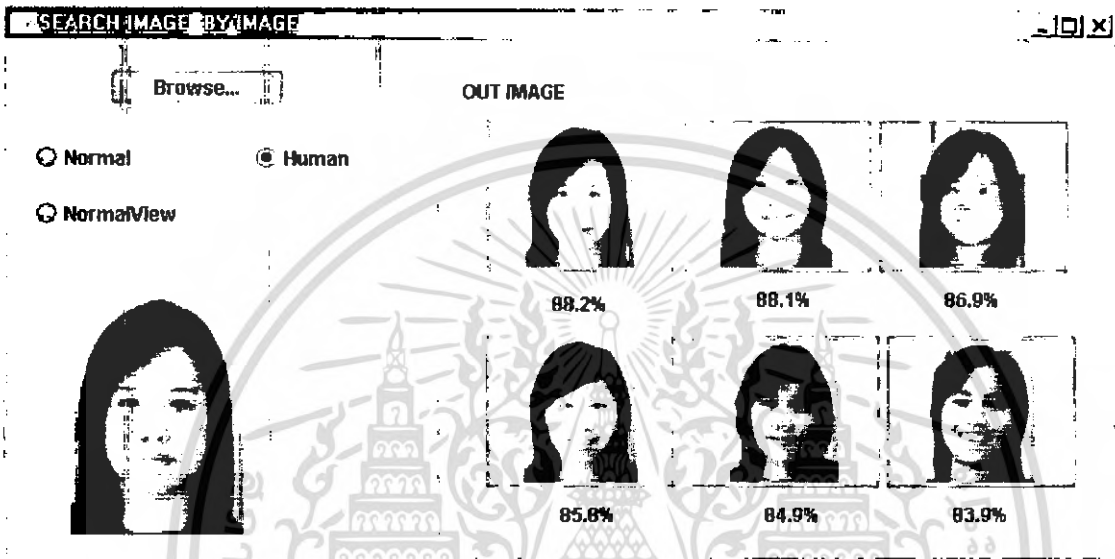
รูปที่ ง-12 เปรียบเทียบฮิสโตแกรมของภาพที่ต้องการหาแล้วภาพภายในฐานข้อมูล

จากตัวอย่างรูปข้างต้นจะมีรูปภาพที่ออกมาเพียง 1 รูป ซึ่งเหมือนเป็นรูปภาพที่เหมือนกันเลย แต่ที่จะกล่าวต่อไป จะเป็นการแสดงผลของการค้นหาที่มีหลายรูปภาพอยู่ในฐานข้อมูลดังรูปที่ ง-13 เป็นตัวอย่างรูปภาพใบหน้าคนที่ถูกบันทึกไว้ในฐานข้อมูลในหมวดของใบหน้าคน



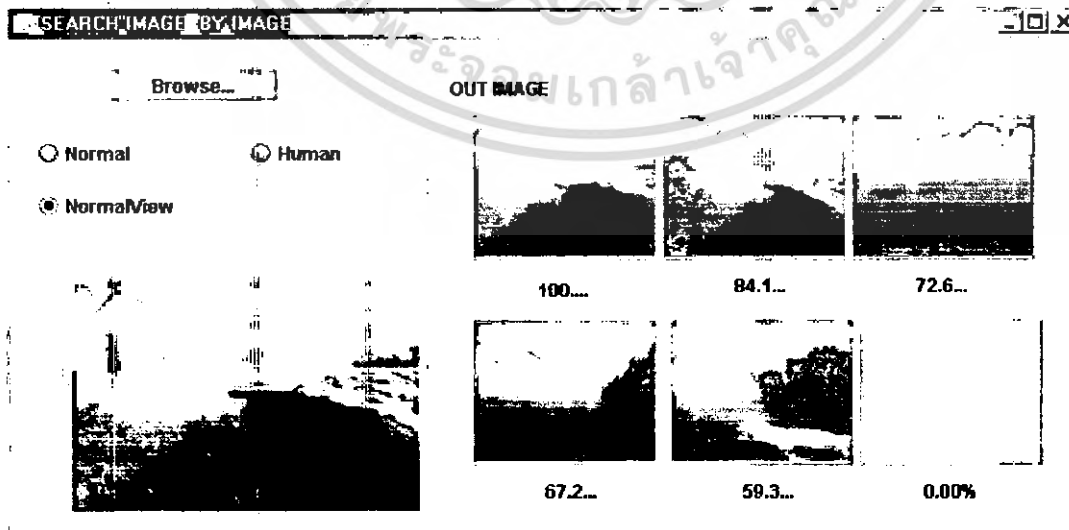
รูปที่ ง-13 แสดงรูปภาพอินพุตที่จัดเก็บในฐานข้อมูล

ซึ่งผลที่ได้จากการค้นหาภาพนั้นจะแสดงดังรูปที่ ง-14 ซึ่งภาพที่ใช้ค้นหาจากตัวอย่างนี้ไม่ได้ถูกจัดเก็บลงในฐานข้อมูล จากนั้นให้เลือกที่หมวด Human ด้วยเนื่องจากภาพที่ต้องการค้นหานั้นเป็นภาพใบหน้าคน (ตอนที่จัดเก็บรูปใบหน้าคนก็ต้องเก็บไว้ในหมวดรูปภาพใบหน้าคนเช่นกัน) ผลที่แสดงออกมาจะแสดงรูปใบหน้าคนที่ความเหมือนใกล้เคียงกันในลำดับถัด ๆ มาตามลำดับ โดยมีเปอร์เซ็นต์แสดงที่ได้ภาพนั้น ๆ



รูปที่ ง-14 แสดงผลการค้นหารูปภาพใบหน้าคนที่มีความคล้ายกัน

ซึ่งเช่นเดียวกับภาพทิวทัศน์ แสดงดังรูปที่ ง-15 ภาพเหล่านี้จะจัดเก็บไว้ในหมวด NormalView เพราะฉะนั้นในการค้นหาภาพหมวดนี้ก็ต้องทำการเลือกหมวดนี้ด้วยดังรูป จะแสดงภาพที่มีความคล้ายกันออกมาตามลำดับ



รูปที่ ง-15 แสดงผลการค้นหาภาพทิวทัศน์ที่มีความคล้ายกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้