

การใช้ JAVA ในการติดต่อฐานข้อมูล
JAVA DATABASE CONNECTIVITY



ปริญญาโทสำหรับปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

เลขหมู่.....
เลขทะเบียน.....
วัน, เดือน, ปี.....

การใช้ JAVA ในการติดต่อฐานข้อมูล
JAVA DATABASE CONNECTIVITY



ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

ปริญญาโทปีการศึกษา 2540

ภาควิชา วิชาวิศวกรรมคอมพิวเตอร์

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง การใช้ JAVA ในการติดต่อฐานข้อมูล

ผู้จัดทำ

1. นาย พรชัย ชรรรมรัตนนันทน์ รหัสนักศึกษา 38013277
2. นาย เอกรินทร์ กัญจนุศย์ รหัสนักศึกษา 38013309

.....
อาจารย์ที่ปรึกษา
(ดร.วรวัฒน์ ลิ้มโกคา)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้ JAVA ในการติดต่อฐานข้อมูล

นายพรชัย ธรรมรัตนนันท

นายเอกรินทร์ ภัฏจนบุศย์

ดร.วรวัฒน์ ลิ้ม โภคา อาจารย์ที่ปรึกษา
ปีการศึกษา 2540

บทคัดย่อ

โครงการนี้เป็น การนำเสนอวิธีการ ในการเชื่อมต่อกับฐานข้อมูล โดยใช้ภาษาจาวา โดยเชื่อมต่อกับฐานข้อมูลผ่านทางบราวเซอร์ของบริการเว็ลด์ไวด์เว็บในเครือข่ายอินเทอร์เน็ต ซึ่งในปัจจุบันเครือข่ายอินเทอร์เน็ตเป็นเครือข่ายที่มีความนิยมสูงสุด การนำเสนอวิธีการ ในการเชื่อมต่อกับฐานข้อมูลผ่านทางเครือข่ายอินเทอร์เน็ตเพื่อใช้จัดการกับข้อมูลในยุค โลกาภิวัตน์จึงถือว่าการประยุกต์ใช้งานเครือข่ายอินเทอร์เน็ตให้เกิดประโยชน์หรือคุ้มค่าน่ามากที่สุด โดยใช้ภาษาจาวาซึ่งเป็นภาษาคอมพิวเตอร์หนึ่งที่ได้รับ ความนิยมใช้ควบคู่กับการประยุกต์งาน ใช้ในงานด้านต่างๆ บนเครือข่ายอินเทอร์เน็ตเรื่อยมาจนเป็นที่รู้จักอย่างกว้างขวาง

การเชื่อมต่อกับฐานข้อมูลสามารถทำงานกับฐานข้อมูลชนิดต่างๆ เช่น ไมโครซอฟท์เอ็กเซล, ไซเบส, ออราเคิล, ไมโครซอฟท์เอสคิวแอลเซิร์ฟเวอร์ เป็นต้น วิธีในการเชื่อมต่อกับฐานข้อมูลเหล่านี้ผ่าน โอเดบีซีซึ่งถือเป็นมาตรฐานหนึ่งในการเชื่อมต่อกับฐานข้อมูล ผู้ใช้สามารถเรียกใช้งานอยู่ ไม่จำกัดชนิดของระบบปฏิบัติการ เพียงแต่เครื่องคอมพิวเตอร์ที่ผู้ใช้งาน ใช้อยู่ได้เชื่อมต่อกับระบบเครือข่าย อินเทอร์เน็ต หรืออินทราเน็ตก็เพียงพอแล้ว

JAVA DATABASE CONNECTIVITY

Pornchai Tummaruttananont

Akarin Kunchanubudth

Dr.Voravat Limpoka Advisor

1997

ABSTRACT

This project is presents how to connect to Database by Java Language. The Connectivity to Database is call and work through the Web browser of the WWW service in the Internet. At present in Internet is to be popular. The method is present to connect with Database work through Internet for to manage data in globalization period is to apply use Internet is most advantage. By Java language is computer language is popular to use with application of Internet

The Connectivity to Database is able to work with several Databases such as Microsoft Access, Sybase, Oracle, Microsoft SQL Server. The method is connectivity to Database with ODBC that is standardization to connect Database. User can call the application for running and working with databases without the limited of the computer platform, or the operating system. The computer should be connect to the Internet or Intranet.

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ภาษาจาวา (JAVA Language)	3
2.1 ความสามารถของภาษาจาวา	3
2.2 การนำจาวาไปประยุกต์สร้างและพัฒนาในงานด้านต่างๆ	4
2.3 JAVA VIRTUAL MACHINE	5
2.4 หลักการทำงานของภาษาจาวา	6
2.4.1 การคอมไพล์	6
2.4.2 การวางตำแหน่งในหน่วยความจำ	7
2.4.3 การรันโปรแกรม	7
2.4.4 Class Loader	8
2.4.5 ตรวจสอบไบนารีโค้ด	8
2.4.6 การทำงานตามโค้ด	9
2.4.7 การสร้างและรันโปรแกรม	9
2.5 Web Browser Support Java	10
2.6 JAVA TOOL	11
2.6.1 Java Development Kit	11
2.6.2 Sun Java Workshop	11
2.6.3 Microsoft Visual J++, Professional Edition	12
2.6.4 Symantec Visual Café	13
2.7 ความแตกต่างระหว่าง JAVA APPLET และ JAVA APPLICATION	17
2.8 ความแตกต่างระหว่าง JAVA และ JAVASCRIPT	17
2.9 ทิศทางของภาษา JAVA	18
บทที่ 3 JAVA DATABASE CONNECTIVITY (JDBC)	19
3.1 ความหมายของ JDBC	21
3.2 รูปแบบการเชื่อมต่อฐานข้อมูล	21
3.3 โครงสร้างของ JDBC	23
3.4 รูปแบบของ JDBC ไดรฟ์เวอร์	24
3.4.1 JDBC/ODBC Bridge	25
3.4.2 Native-API, Partly Java Driver	26

3.4.3	Network-Protocol, All-Java Driver	27
3.4.4	Native-Protocol, All Java Driver	30
3.5	ขั้นตอนพื้นฐานสำหรับการเชื่อมต่อกับฐานข้อมูล	31
บทที่ 4	การวางแผนและการสร้าง	35
4.1	ขอบเขตการดำเนินงาน	35
4.2	แผนการดำเนินงาน	35
บทที่ 5	ผลการดำเนินงาน	48
5.1	ระบบแสดงผลการเรียนผ่านเครือข่ายอินเทอร์เน็ต	48
5.2	ระบบลงทะเบียนผ่านเครือข่ายอินเทอร์เน็ต	49
บทที่ 6	บทวิจารณ์และสรุป	51
6.1	สรุปผลการดำเนินงาน	51
6.2	แนวทางในการพัฒนาต่อ	51
ภาคผนวก ก	IDS Server	53
ภาคผนวก ข	ODBC Open Database Connectivity	58
ภาคผนวก ค	SQL Syntax	61
ภาคผนวก ง	โปรแกรมระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต	63
ภาคผนวก จ	โปรแกรมระบบงานลงทะเบียนการเรียนผ่านเครือข่ายอินเทอร์เน็ต	81
ภาคผนวก ฉ	คู่มือการใช้งานโปรแกรม	96
กิตติกรรมประกาศ		109
หนังสืออ้างอิง		110

สารบัญรูปภาพ

หน้า

รูปที่ 2 – 1	การประมวลผลของโปรแกรมทั่วไปแต่ระบบปฏิบัติการที่ต่างกัน	7
รูปที่ 2 – 2	การประมวลผลของโปรแกรมภาษา Java	7
รูปที่ 2 – 3	โปรแกรม Sun Microsystems: Java WorkShop	12
รูปที่ 2 – 4	โปรแกรม Microsoft Visual J++	13
รูปที่ 2 – 5	โปรแกรม Symantec Visual Café	14
รูปที่ 3 -- 1	Two-Tier Model	22
รูปที่ 3 – 2	Three-Tier Model	23
รูปที่ 3 – 3	ระดับการเชื่อมต่อของ JDBC API	23
รูปที่ 3 – 4	JDBC driver types	24
รูปที่ 3 – 5	การติดต่อ JDBC โดยรูปแบบที่ 1 JDBC-ODBC Bridge	26
รูปที่ 3 -- 6	การติดต่อ JDBC โดยรูปแบบที่ 2 Native-API, partly Java driver	27
รูปที่ 3 – 7	การติดต่อ JDBC โดยรูปแบบที่ 3 Network-protocol, all-Java driver ยกตัวอย่างของ dbAnywhere, SequelLink	28
รูปที่ 3 - 8	การติดต่อ JDBC โดยรูปแบบที่ 3 Network-protocol, all-Java driver ยกตัวอย่างของ WebLogic	29
รูปที่ 3 – 9	การติดต่อ JDBC โดยรูปแบบที่ 4 Native-protocol, all-Java driver	30
รูปที่ 4 -- 1	รูปแสดงความสัมพันธ์ระหว่างฐานข้อมูล	36
รูปที่ 4 -- 2	รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต	37
รูปที่ 4 -- 3	รูปแสดง FLOWCHART ระบบลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต	43
รูปที่ 4 – 4	รูปแสดง FLOWCHART ระบบรักษาความปลอดภัยของโปรแกรม	45
รูปที่ 5 -- 1	รูปแสดงระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต	48
รูปที่ 5 -- 2	รูปแสดงระบบลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต	49
รูปที่ ก – 1	Directory Hierachy	54
รูปที่ ก – 2	Running IDS Server บน Windows NT	55
รูปที่ ก – 3	รูปแบบ Network ของ IDS Server	57
รูปที่ ข – 1	องค์ประกอบของ ODBC	59
รูปที่ ฉ – 1	การเชื่อมต่อ ODBC Data Source	96
รูปที่ ฉ – 2	การเลือก NEW Data Source	97
รูปที่ ฉ – 3	การเชื่อมต่อ Data Source ของ Microsoft Access	97

รูปที่ ฉ – 4	การระบุชื่อไฟล์ฐานข้อมูลของ Microsoft Access	98
รูปที่ ฉ – 5	การระบุรายละเอียดในการติดต่อของ Microsoft Access	98
รูปที่ ฉ – 6	การระบุรายละเอียดในการติดต่อของ Sybase SQL Anywhere	99
รูปที่ ฉ – 7	การระบุรายละเอียดในการติดต่อของ SQL Server	100
รูปที่ ฉ – 8	การระบุรายละเอียดในการติดต่อของ Oracle	100
รูปที่ ฉ – 9	แสดงเว็บเพจที่สร้าง	101
รูปที่ ฉ – 10	แสดงรายชื่อฐานข้อมูลที่จัดไว้	102
รูปที่ ฉ – 11	แสดงหน้าต่างรหัสนักศึกษา และรหัสผ่าน	102
รูปที่ ฉ – 12	แสดงหน้าต่างในกรณีที่ป้อนรหัสนักศึกษา หรือรหัสผ่านผิด	103
รูปที่ ฉ – 13	แสดงผลการศึกษาของปีล่าสุด	103
รูปที่ ฉ – 14	แสดงหน้าต่างของแบบฟอร์มในการลงทะเบียน	104
รูปที่ ฉ – 15	แสดงตัวอย่างการป้อนข้อมูล	105
รูปที่ ฉ – 16	แสดงรายวิชาเมื่อได้ป้อนชั้นปีและเทอม	105
รูปที่ ฉ – 17	แสดงผลเมื่อมีการ Add หรือ Add All	106
รูปที่ ฉ – 18	แสดงการลงทะเบียนของชั้นปีอื่น	107
รูปที่ ฉ – 19	แสดงรายละเอียดของการลงทะเบียน	108
รูปที่ ฉ – 20	แสดงหน้าต่างแจ้งเตือนการลงทะเบียนไม่สมบูรณ์	108

สารบัญตาราง

	หน้า
ตารางที่ 2 – 1 ตารางเปรียบเทียบเครื่องมือในการพัฒนาภาษาจาวา	16
ตารางที่ 2 – 2 ตารางเปรียบเทียบ Java Script และ Java	17
ตารางที่ 4 – 1 ตารางที่ 4 – 1 เปรียบเทียบ JDBC Driver	34



บทที่ 1

บทนำ

การใช้งานระบบเครือข่ายอินเทอร์เน็ตสามารถเชื่อมต่อเครื่องคอมพิวเตอร์ที่อยู่ในสถานที่ต่างๆ ที่ห่างกันไกลได้ทั่วโลก ทำให้สามารถใช้งานและแบ่งปันฐานข้อมูลต่างๆ ซึ่งช่วยให้เกิดการพัฒนาด้านต่างๆ ไม่ว่าจะเป็นวงการทางการศึกษาหรือวงการทางด้านธุรกิจอย่างรวดเร็ว เช่นการค้นคว้าข้อมูลในการทำวิจัยของนักวิชาการ นักวิทยาศาสตร์ การค้นคว้าข้อมูลสำหรับการทำธุรกิจ การค้นคว้าข้อมูลของนักศึกษาในการเรียนการศึกษา หรือแม้กระทั่งการค้นหาข้อมูลต่างๆ เพื่อความบันเทิงของผู้ใช้งานทั่วไป ดังนั้นจึงทำให้ระบบเครือข่ายอินเทอร์เน็ต จึงมีผู้ใช้งานมากมาย และมีแนวโน้มที่จะมีผู้ใช้งานเพิ่มขึ้นเรื่อยๆ อย่างต่อเนื่อง

ปัจจุบันการบริการ World Wide Web กำลังเป็นที่นิยมและมีการขยายตัวใช้กันอย่างแพร่หลายมากที่สุดบน Internet ทั้งนี้เนื่องจากมีรูปแบบการนำเสนอที่เข้าใจง่ายมีการแสดงผลเป็นหน้าๆ (page) เหมือนกับเอกสารที่อ่านกันบนกระดาษ โดยเรียกว่าเป็น “เว็บเพจ” (WebPages) ซึ่งอาจประกอบไปด้วยตัวอักษร, กราฟิก, เสียง, ภาพเคลื่อนไหวและที่สำคัญคือแต่ละเว็บเพจจะเป็นสื่อแบบมัลติมีเดียที่สามารถโต้ตอบกับผู้ใช้ได้ ในเว็บเพจ นอกจากจะเป็นแหล่งแลกเปลี่ยนความรู้ เป็นสื่อโฆษณา ยังเป็นศูนย์กลางธุรกิจทางการค้าที่สำคัญที่มีการขยายตัวทางการตลาดอย่างรวดเร็วบนเว็ลด์ไวด์เน็ตเวิร์ค

ในปัจจุบันมีภาษาระดับสูงหนึ่งที่มีความสามารถในการควบคุมและประยุกต์ใช้เว็บเพจ ได้อย่างมากมายคือจาวา (JAVA) ซึ่งเป็นภาษาแรกที่ใช้สำหรับการพัฒนาระบบคอมพิวเตอร์เน็ตเวิร์ค ใช้ประโยชน์ของการเขียน โปรแกรมอ้างอิงเชิงวัตถุ (Object Oriented Programming) และความสามารถอีกหลายๆ ด้านซึ่งเหมาะกับการศึกษาและประยุกต์ใช้เทคโนโลยีสารสนเทศนี้ เพื่อเชื่อมต่ออินเทอร์เน็ต ให้เหมาะกับการใช้งานและเกิดประโยชน์สูงสุดในชีวิตประจำวันให้มากที่สุด ผู้นำเทคโนโลยีทางด้าน Server และ Networking ส่วนมากเห็นความสำคัญของการบริการ World Wide Web จึงได้พัฒนาและประยุกต์ใช้จาวากับเว็บเพจ ซึ่งได้ถูกออกแบบมาให้ง่ายต่อการ ใช้งาน ผู้ใช้สามารถเรียนรู้การใช้งานได้ด้วยตนเองและสามารถทำ Business Transactions บนเครือข่ายอินเทอร์เน็ตหรืออินเทอร์เน็ต โดยใช้เว็บเบราว์เซอร์มาตรฐานทั่วไปเช่น Netscape, Mosaic หรือ Internet Explorer โดยที่ไม่จำเป็นต้อง Install Application ที่ Client

เนื้อหาในปฏิญญาฉบับนี้จะอธิบายถึงหลักการเบื้องต้นในการใช้งานภาษาจาวา วิธีการสร้างโปรแกรม การรันโปรแกรม ความสามารถของภาษาจาวาและแนวโน้มการใช้งานภาษาจาวาในอนาคต การนำภาษาจาวาไปประยุกต์ใช้งานในด้านการเชื่อมต่อกับฐานข้อมูล โดยออกแบบ โปรแกรมแสดงผลการศึกษาและตรวจสอบการลงทะเบียนของนักศึกษาโดยติดต่อกับฐานข้อมูลเช่น Microsoft Access และระบบจัดการฐานข้อมูล (Database Management System : DBMS) เช่น Microsoft SQL Server , Oracle , Sybase โดยผ่านทาง ODBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากเนื้อหาทั้งหมดที่กล่าวถึงในปฏิญญานี้นั้นจะช่วยให้สามารถสร้างระบบงานในการเชื่อมต่อกับฐานข้อมูลโดยการประยุกต์ผ่านเว็บเพจในเครือข่ายอินเทอร์เน็ตได้ ซึ่งการนำภาษาจาวามาประยุกต์ใช้งานฐานข้อมูลดังกล่าวจะมีข้อดีคือสามารถติดต่อกับฐานข้อมูลได้หลากหลาย การนำเสนอโปรแกรมผ่านเว็บเพจในเครือข่ายอินเทอร์เน็ต โปรแกรมจะเป็นอิสระไม่ขึ้นกับเครื่องคอมพิวเตอร์ของผู้ใช้ ไม่ขึ้นกับระบบปฏิบัติการที่ผู้ใช้ใช้งานอยู่ เช่น Windows, Unix, Macintosh, OS/2 เป็นต้น



บทที่ 2

ภาษาจาวา (JAVA LANGUAGE)

ภาษาจาวา หรือ Java Programming Language ถูกพัฒนาขึ้นโดยบริษัท ซันไมโครซิสเต็มส์ เริ่มแรกซันได้คิดค้นภาษาจาวาขึ้นมาเพื่อใช้กับระบบปฏิบัติการ Set Top Box คือโครงการของระบบตลาดอิเล็กทรอนิกส์ ที่ต้องการให้ระบบสามารถทำการควบคุมอุปกรณ์อิเล็กทรอนิกส์ได้โดยไม่ขึ้นอยู่กับแพลตฟอร์ม (Platform) ที่ใช้ คือการพัฒนาระบบปฏิบัติการหรือ ซอฟต์แวร์ให้สามารถทำงานได้ โดยไม่สนใจว่าไมโครโพรเซสเซอร์ที่ใช้อยู่จะเป็นรุ่นใด ยี่ห้อไหน ฮาร์ดแวร์เป็นของอะไร ทั้งยังได้รวมความสามารถไปถึงการเชื่อมต่อ การเปลี่ยนข้อมูลข่าวสารต่างๆ ไปและทางด้านธุรกิจ ผ่านทางระบบเครือข่ายของคอมพิวเตอร์อีกด้วย

ทางทีมงานซึ่งพัฒนาโปรแกรมนี้ได้สังเกตเห็นถึงการเจริญเติบโตและความสามารถในการแลกเปลี่ยนข่าวสารบนอินเทอร์เน็ตจึงได้วางแผนและพัฒนา ระบบ Interface แบบใหม่เพื่อใช้กับอินเทอร์เน็ต ขึ้น ด้วยสาเหตุนี้เองภาษาจาวาจึงได้บังเกิดขึ้น รวมทั้งยังได้มีการเปิดตัวเว็บเบราว์เซอร์ (Web Browser) ที่ชื่อว่า Hotjava เพื่อใช้สำหรับแสดงความสามารถของภาษาจาวา

2.1 ความสามารถของภาษาจาวา

ลักษณะของภาษา Java จะเป็นการเขียนโปรแกรมอ้างอิงเชิงวัตถุ OOP (Object Oriented Programming) สามารถสรุปความสามารถการทำงาน ได้ดังต่อไปนี้

1. การทำงานที่ไม่ขึ้นอยู่กับแพลตฟอร์ม โดยใช้วิธีการแปลงข้อมูลแบบ byte code
2. การใช้ Methods จัดการกับข้อมูลแทนการใช้ฟังก์ชัน
3. ความสามารถในการโต้ตอบแบบ InterActive กับงานด้านมัลติมีเดีย
4. ความสามารถในการดึงเอาคลาสไลบรารีจากที่ต่างๆ ผ่านทาง HTTP และ FTP
5. ความสามารถในการเล่นไฟล์เสียง ไฟล์วิดีโอและไฟล์ภาพกราฟิกเคลื่อนไหว
6. การโต้ตอบแบบเรียลไทม์
7. เป็นระบบ Multithreaded คือสามารถทำงานได้หลายงานพร้อมกันในเวลาเดียว
8. ไม่มีตัวแปรประเภทพอยน์เตอร์จึงหมดปัญหาเกี่ยวกับการขอใช้หน่วยความจำโดยตรง
9. มีความปลอดภัยสูง ด้วยความสามารถที่ไม่ให้ผู้ใช้เข้าไปยุ่งเกี่ยวกับหน่วยความจำของระบบ จึงปลอดภัยจากไวรัส
10. การดาวน์โหลดซอฟต์แวร์ใหม่ๆ และอัปเดตเวอร์ชันใหม่ได้อย่างง่ายดาย โดยผ่านทางเว็บ

เพจ

2.2 การนำจาวาไปประยุกต์สร้างและพัฒนาในงานด้านต่างๆ

จากความสามารถที่กล่าวผ่านมาข้างต้นคงพอจะเป็นข้อสรุปได้แล้วว่า Java เป็นภาษาโปรแกรมที่มีความสามารถรอบด้าน อีกทั้งยังมีความยืดหยุ่นภายในตัวสูง การที่จะนำมาสร้างและพัฒนางานใดๆ มีประสิทธิภาพคงจะทำได้ไม่ยากนัก เราสามารถใช้ Java สร้างและพัฒนาแอปพลิเคชันต่างๆ โดยยกตัวอย่างพอสังเขปได้ดังนี้

Client

จาวาสามารถเป็นเครื่องมือในการพัฒนาซอฟต์แวร์ด้าน Client ได้เพียงพอโดยเฉพาะด้านกราฟิกและระบบติดต่อกับผู้ใช้ เพราะปัจจุบันโปรแกรมที่เขียนขึ้นจากภาษา Java จะทำงานที่ฝั่ง Client เป็นส่วนมากจะเห็นได้จาก Java Applet ต่างๆบน Web Browser แต่จะให้ดีและสมบูรณ์ยิ่งขึ้นคือความเป็น Client Software ที่ดีก็คือ การติดต่อบริการฐานข้อมูลจาก Client ไปเป็น Server แม้ว่าในปัจจุบัน Java จะสามารถทำได้แล้วก็ตามแต่ก็ต้องใช้เครื่องมือชนิดอื่นเป็นตัวช่วย เช่น การเชื่อมต่อกับฐานข้อมูลแบบ ODBC (Object-Oriented Database Connectivity) ซึ่งการทำงานของทางฝั่ง Server ต้องใช้ภาษาคอมพิวเตอร์ชนิดอื่นช่วยเพื่อการติดต่อกับฐานข้อมูล

Server

จาวาเมื่อใช้สร้าง โปรแกรมประยุกต์ที่ติดบน Server นั้น Java จำเป็นที่จะต้องได้รับการปรับปรุงการทำงานอยู่เสมออย่างคือ การเชื่อมต่อกับฐานข้อมูลโดยตรง การทำงานให้สอดคล้องกับอินเทอร์เน็ต และความเร็วของการรันต้องเพิ่มขึ้นเรื่อยๆ กับ โปรแกรมประยุกต์ทั่วไป

ระบบ JDBC (Java Database Connectivity) ได้เข้ามาแก้ปัญหาในส่วนของการทำงานติดต่อกับฐานข้อมูลโดยตรง โดยการร่วมมือกันของบริษัทซอฟต์แวร์ที่มีชื่อเสียงหลายบริษัท ได้สร้างมาตรฐานการเชื่อมต่อกับฐานข้อมูลของ Java ขึ้น แต่ใจขณะนี้ก็ยัง ไม่เป็นที่รู้จักแพร่หลายนักสำหรับ JDBC คาดว่าในอนาคตอันใกล้การเชื่อมต่อฐานข้อมูลจะทำได้ง่ายขึ้นจาก Client สู่ Server โดยใช้ Java

Security

Java ได้สร้างสภาพแวดล้อมเสมือน “Java Virtual Machine” ซึ่งจะอนุญาตให้เฉพาะ Java Applet หรือ Java Application ซึ่งเป็น Byte Code เท่านั้นที่สามารถรันได้โดยที่ Byte Code จะปราศจากไวรัสหรือส่วนที่จะทำอันตรายต่อระบบ อันเนื่องจาก Java Virtual Machine จะจำกัดสิทธิ์การเข้าใช้ทรัพยากรของ Byte Code ในอนาคตการรักษาความปลอดภัยอีกระบบหนึ่งที่กำลังถูกนำมาปรับใช้คือ “การเข้ารหัสข้อมูล” เพิ่มความปลอดภัยของข้อมูล

การใช้จาวาประยุกต์สร้างและพัฒนางานยังสามารถแบ่งเป็นประเภทตามลักษณะของกลุ่มงานได้ดังนี้

- งานด้านการศึกษา โดยจัดทำเป็นลักษณะสั่งการสอนคล้ายกับ CAI มีความสามารถในเชิงโต้ตอบระหว่างผู้ใช้งานกับคอมพิวเตอร์

- คอบสนองงานทางด้าน Web Page ทำให้งานสร้าง Web Page คู่มือชีวิตชีวาขึ้น โดยการนำเอาความสามารถทางด้านมัลติมีเดียมาใช้อย่างเต็มที่
- สร้างสรรค์เครื่องมือพัฒนา (Software Developer tool kits)
- สร้างแอปพลิเคชันทางธุรกิจ มีความสามารถในการประมวลผลข้อมูลด้วยตัวเอง
- สร้างตัวจัดการบนอินเทอร์เน็ต มีความสามารถในการเรียกหาข้อมูลผ่านทางระบบเครือข่าย
- พัฒนาเกม จะมีความเป็นมัลติ คือสามารถเล่นได้หลายๆ คนพร้อมกันผ่านทางระบบเครือข่ายหลากหลายพื้นที่ไม่ว่าจะเล่นด้วยกัน ณ ตำแหน่งใด ทั่วทุกมุมโลก

2.3 Java Virtual Machine

Java Virtual Machine เป็นหลักการสร้างคอมพิวเตอร์จำลองของจาวา โดยการสมมติให้มีคอมพิวเตอร์อีกเครื่องหนึ่งขึ้นมาโดยเครื่องนี้จะใช้ในการคอมไพล์โปรแกรมภาษาจาวาทุกโปรแกรม เมื่อต้องการให้โปรแกรมภาษาจาวาไปทำงานบนคอมพิวเตอร์จริงๆ เครื่องใด เราก็กึ่งแค่สร้างตัวอินเทอร์พรีเตอร์ (Interpreter) ของคอมพิวเตอร์จำลองตัวนั้นบนเครื่องนั้นๆ ภาษาจาวาทุกโปรแกรมก็จะสามารถทำงานบนระบบคอมพิวเตอร์นั้นๆ ได้ตามต้องการ ด้วยหลักการนี้ก็เป็นที่มาของคุณสมบัติของจาวา ที่ไม่ขึ้นกับแพลตฟอร์มและฮาร์ดแวร์ใดๆ หรือพอร์ตเทเบิล (Portable) เราอาจจะเรียก Java Virtual Machine สั้นๆ ว่า JVM ก็ได้

สำหรับเหตุผลที่ต้องมี JVM นี้ จริงๆ แล้วเป็นการกำหนดค่าขึ้นมา เพื่อเป็นคำจำกัดความเฉพาะในความคิดเท่านั้น เพื่อให้ นักพัฒนาจะได้ไม่ถูกบังคับให้ต้องสร้างตัวอินเทอร์พรีเตอร์ตามแนวทางแบบใดแบบหนึ่งโดยเฉพาะ แต่ตัวอินเทอร์พรีเตอร์ที่สร้างขึ้นตามข้อกำหนดดังกล่าว ไม่ว่าจะอยู่บนแพลตฟอร์มใด จะสามารถรันโปรแกรมที่เขียนขึ้นในภาษาจาวาได้ โดยให้ผลลัพธ์ออกมาเหมือนกัน

แต่ในข้อกำหนดของ JVM ก็มีการให้นิยามที่ชัดเจนมากๆ เกี่ยวกับการออกแบบอินเทอร์พรีเตอร์ในหลายๆ ส่วน โดยเฉพาะอย่างยิ่งส่วนที่เกี่ยวข้องกับการกระจายโค้ดของจาวาไปใส่ไว้ในรูปแบบที่กำหนด ข้อกำหนดนี้ได้แก่ไวยากรณ์ของออบเจกต์และโอเปอเรนด์ พร้อมกับค่าประจำตัว การจัดโครงสร้างของโค้ด การจัดวางรูปแบบของออบเจกต์ของจาวา ข้อกำหนดต่างๆ เหล่านี้จะทำให้นักพัฒนาอินเทอร์พรีเตอร์ทั้งหลาย สามารถจะสร้างอินเทอร์พรีเตอร์ขึ้นมาใหม่บนแพลตฟอร์มใดๆ ก็ได้ ดังนั้น การพัฒนาภาษาจาวา จึงไม่ได้ถูกปิดกั้นอยู่กับระบบของซันซึ่งเป็นผู้คิดค้นภาษาจาวานี้ขึ้นมาเท่านั้น

จากที่กล่าวมาข้างต้นถึงคุณสมบัติของจาวาที่ไม่ยึดติดอยู่กับแพลตฟอร์มใดๆ ทำให้การทำงานกับระบบคอมพิวเตอร์ที่ทำงานแบบกระจาย (distributed Computing) ได้รับการตอบสนองอย่างเหมาะสม นั่นคือ ทำให้เกิดความพอร์ตเทเบิลนั่นเอง

อย่างไรก็ตาม JVM ก็ยังมีข้อจำกัดอยู่ ข้อจำกัดของ JVM นั้นจะอยู่ที่ข้อจำกัดของการออกแบบตัวอินเทอร์พรีเตอร์แทน เช่น การจำกัดค่าของโอเปอเรนด์ และขนาดของสแต็ก เป็นต้น ข้อกำหนดเหล่านี้หมายถึงว่า JVM สามารถจะอ้างถึงหน่วยความจำได้เฉพาะในห่วงแอดเดรสเท่าที่มีอยู่เท่านั้น

ข้อจำกัดภายในของตัว JVM เองนั้นมีห่วงแอดเดรสให้ใช้ได้ถึง 4 GB เพราะขนาดของความกว้างของการอ้างแอดเดรสเป็น 32 บิต Method ต่างๆ ของจาวามีขนาดได้เพียง 32 KB เพราะมีข้อจำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของการใช้คำสั่งกระโดด เป็นแบบ 16 บิต (โดยมีบิตแรกเป็นตัวบอกว่ากระโดดไปข้างหน้า หรือข้างหลัง และบิตต่อๆ ไป บอกระยะทางการกระโดดจากจุดที่หล่านี้มีขนาดเพียง 8 บิต นอกจากนั้น จำนวนของคำสั่งที่อยู่ในส่วนกลาง ยังถูกจำกัดด้วยขนาดของดัชนี 16 บิต ทำให้มีจำนวนได้เพียง 32,000 คำในแต่ละ method

การที่เราถือว่า การมีค่าหล่านี้ เป็นข้อจำกัด ก็อาจจะเป็นการมองการณ์ไกลไปสักหน่อย เนื่องจากปัจจุบันเครื่องคอมพิวเตอร์ส่วนใหญ่มีหน่วยความจำจำกัดเพียง 16 หรือ 32 MB กันเท่านั้น หน่วยความจำขนาด 4 GB จึงยังเป็นเรื่องที่ไม่ต้องคิดหนักในตอนนี ในขณะที่ขนาดของ method ที่จำกัดที่ 32 KB ก็เป็นเพียง method เดียวเท่านั้น

2.4 หลักการทำงานของภาษาจาวา

ในที่นี้ขออธิบายการทำงานของภาษาจาวาเริ่มตั้งแต่การคอมไพล์ตัวโปรแกรมจนกระทั่งเราเรียกใช้งานในเว็บเพจ เป็นดังนี้

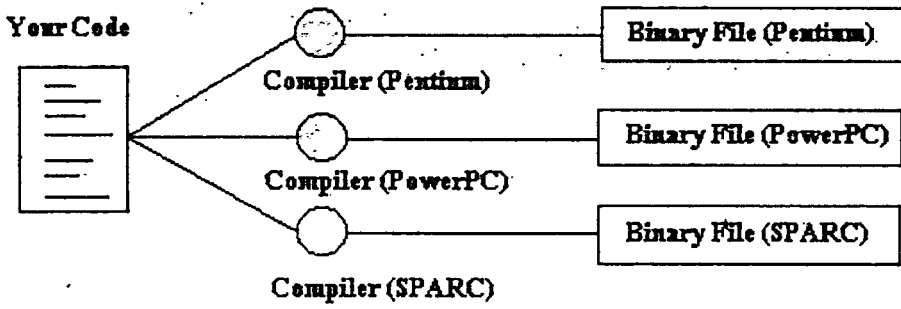
2.4.1 การคอมไพล์

คอมไพเลอร์ของภาษาจาวาก็เช่นเดียวกับคอมไพเลอร์ในภาษาอื่นๆ นั่นคือ มันจะสร้างรหัสภาษาเครื่อง (Machine Code หรือ Assembler Code) จากภาษาในระดับที่สูงกว่าเพื่อให้ซีพียู (CPU : Central Processing Unit) สามารถนำไปใช้งานได้

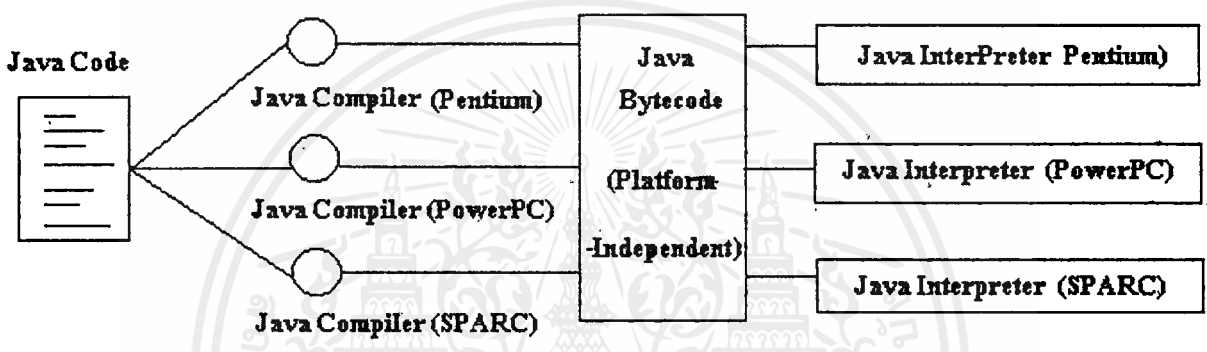
แต่ข้อแตกต่างที่สำคัญระหว่างคอมไพเลอร์ของภาษาจาวากับภาษาอื่นๆ คือ ซีพียูหรือโปรเซสเซอร์ที่จะทำหน้าที่ในการปฏิบัติตามคำสั่งที่ได้จากการคอมไพล์ภาษาจาวานั้นไม่มีอยู่จริง เป็นเพียงสิ่งที่สมมติขึ้นมาที่เรียกว่า Java Virtual Machine นอกจากนี้การอ้างถึงส่วนต่างๆ ของโปรแกรมที่คอมไพล์ด้วยคอมไพเลอร์ของภาษาจาวาก็จะมีวิธีการที่แตกต่างออกไป

คอมไพเลอร์ของภาษาจาวาจะไม่เปลี่ยนการอ้างถึงส่วนของโปรแกรมจากการใช้ชื่อแบบในภาษาสูง ไปเป็นตัวเลขเหมือนคอมไพเลอร์ภาษาอื่นๆ ทำกัน และคอมไพเลอร์ภาษาจาวาก็จะไม่มีการสร้างแผนที่ของการจัดวาง โปรแกรมบนหน่วย ความจำขึ้นมาในระหว่างการคอมไพล์ด้วยเหตุผลที่สำคัญคือ เพื่อเป็นการสร้างความพอร์ดเทเบิลให้กับตัวโปรแกรมเพราะการจัดวางตำแหน่งของโปรแกรมจะต้องขึ้นอยู่กับลักษณะการทำงานของโปรเซสเซอร์ตัวใดตัวหนึ่ง การยังไม่จัดวางตำแหน่งช่วยทำให้โปรแกรมที่ได้จากการคอมไพล์มีความเป็นกลาง สามารถนำไปใช้บนคอมพิวเตอร์แพลตฟอร์มอื่นๆ ได้ นอกจากนี้มันยังทำให้เกิดความปลอดภัยอีกด้วย

สิ่งที่ได้จากการคอมไพล์ในภาษาจาวาเราเรียกว่า ไบท์โค้ด (ByteCode)



รูปที่ 2-1 การประมวลผลของโปรแกรมทั่วไปแก่ระบบปฏิบัติการที่ต่างกัน



รูปที่ 2-2 การประมวลผลของโปรแกรมภาษา Java

2.4.2 การวางตำแหน่งในหน่วยความจำ

ในภาษาจาวาจะ ไม่มีการลดรูปแบบการอ้างถึงส่วนต่างๆ ของโปรแกรมจากการเรียกเป็นชื่อ ให้เหลือเพียงตัวเลขหรือแอดเดรสที่กำหนดขึ้นจากการจัดวางตำแหน่งของโปรแกรมลงในหน่วยความจำ คอมไพเลอร์ภาษาจาวาจะตั้งชื่อของแต่ละส่วนของ โปรแกรม (โดยเฉพาะ method) เอาไว้ในตัวโปรแกรมที่สร้างขึ้น เมื่อโปรแกรมทำงาน จะเป็นหน้าที่ของตัว อินเทอร์เน็ตอร์ที่จะคอยเปิดตารางค้นหาที่อยู่ของ method ที่ต้องการเรียกใช้งาน โดยก่อนที่จะเริ่มทำงานจริง อินเทอร์เน็ตอร์จะต้องสร้างแผนที่ในการจัดวางสิ่งต่างๆ ลงในหน่วยความจำขึ้นมาก่อนแล้วจึงสร้างตารางขึ้นมา เพื่อช่วยหาค่าตำแหน่งของ method เมื่อมีการเรียกใช้งานโดยใช้ชื่อของ method

2.4.3 การรันโปรแกรม

การรันโค้ดที่คอมไพล์เอาไว้สำหรับ Java Virtual Machine เป็นหน้าที่ของตัวอินเทอร์เน็ตอร์ การรันโปรแกรมจะแบ่งได้เป็น 3 ขั้นตอนหลักๆ คือ การอ่าน การตรวจสอบความถูกต้อง และการทำตามโค้ด หน้าที่ในการอ่านโค้ดเข้าสู่ระบบจะเป็นของ Class Loader หน้าที่การทำงานในส่วนนี้จะไม่ได้เข้ามาเฉพาะๆ ฟังก์ชันที่กำลังจะเรียกใช้เท่านั้น แต่จะอ่านคลาสที่มีการอ้างถึงและคลาสที่มีการ inherited มาโดยคลาสที่อ้างถึง เมื่อผ่านขั้นตอนี้แล้ว โค้ดทั้งหมดก็จะถูกส่งผ่านตัวตรวจสอบไบท์โค้ดเพื่อให้แน่ใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใจว่าโค้ดที่ส่งมามีความถูกต้องตามมาตรฐานของจาวา และจะไม่รบกวนเสถียรภาพของระบบ เมื่อผ่านการตรวจสอบแล้วโค้ดก็จะถูกส่งต่อไปยังระบบปรันใหม่ (Run-Time System) ซึ่งจะส่งงานไปยังฮาร์ดแวร์อีกต่อหนึ่งซึ่งหลักการทำงานในแต่ละขั้นตอนที่กล่าวมาข้างต้นเป็นดังนี้

2.4.4 Class Loader

ตัวคลาสโหลดเดอร์ จำทำหน้าที่ดึงโค้ดทั้งหมดที่จำเป็นในการทำงานของแอปพลิเคชัน ไม่ว่าจะ เป็นคลาสที่ถูก Inherited มาหรือคลาสอื่นๆ ที่มีการเรียกใช้ เมื่อโหลดเดอร์ดึงคลาสใดเข้ามาแล้วก็จัด คลาสนั้นๆ ใส่เข้าไปใน namespace ของมันเอง โดยการเก็บ จะใช้ชื่อของคลาสเป็นสำคัญ ไม่ได้ใช้การ อ้างถึงเป็นตัวเลข หลักการนี้ก็เหมือนกับการทำงานของ Virtual Machine ที่โอเอส (OS : Operating System) สร้างขึ้นให้แอปพลิเคชันแต่ละตัวทำงาน ถ้าไม่ได้มีการเจาะจงเรียกใช้คลาสที่อยู่นอก namespace นี้ การเรียกใช้ชื่อต่างๆ ในคลาส ก็จะไม่มีการรบกวนกันระหว่างคลาสเลย

คลาสทั้งหมดที่อยู่บนเครื่อง โลกของตัวเอง จะได้รับห้วงแอดเดรส (Address Space) เป็นของตัวเอง ส่วนคลาสต่างๆ ที่ดึงมาจากภายนอกจะได้รับ namespace เป็นของตัวเอง การทำงานลักษณะนี้จะช่วยให้ คลาสที่อยู่บนโลกการทำงาน ได้ประสิทธิภาพดีขึ้น เพราะใช้ namespace ร่วมกันได้ แต่ก็ยังมีการป้องกัน ความผิดพลาดที่อาจจะเกิดจากคลาสที่ดึงเข้ามาจากภายนอก และในทางกลับกัน คลาสที่อิมพอร์ตเข้ามา ก็ปลอดภัยจากความผิดพลาดที่อาจจะเกิดขึ้นจากคลาส โลกของตัวเอง

เมื่อคลาสทั้งหมดที่เกี่ยวข้องกับการทำงานถูกอิมพอร์ตเข้ามาเรียบร้อยแล้ว การจัดวางหน่วย ความจำสำหรับเริ่มต้นการทำงานก็จะเกิดขึ้นได้ การเรียกชื่อต่างๆ ก็จะสามารถจับคู่กับแอดเดรสจริงๆ ของหน่วยความจำได้ แล้วตัวโหลดเดอร์จะสร้างตารางสำหรับค้นหาที่อยู่เสี่ยงจากการทำงานผิดปกติของ ซุปเปอร์คลาส (Super class) และการอ้างแอดเดรสที่ไม่ถูกต้องได้

2.4.5 ตรวจสอบไบท์โค้ด

เมื่อโค้ดเดินทางมาถึงขั้นตอนการสร้างตารางจับคู่ชื่อกับแอดเดรสแล้ว ก็ยังไม่สามารถแน่ใจ ได้ว่าโค้ดที่อ่านเข้ามาจะมีความปลอดภัย ดังนั้นจึงต้องมีตัว verifier หรือตัวตรวจสอบไบท์โค้ด ทำหน้าที่ ตรวจสอบความถูกต้องที่ละบรรทัดว่าเป็นไปตามข้อกำหนดของจาวา และสอดคล้องกับการทำงานของ ตัวโปรแกรมเองหรือไม่ การตรวจสอบโค้ดในเชิงทฤษฎีจะสร้างค้นหาปัญหาต่างๆ ได้หลายอย่าง เช่น จะไม่มีการสร้างพอยต์เตอร์ที่เกินกว่าหน่วยความจำจริง ไม่มีคำสั่งใดสามารถละเมิดสิทธิ์การทำงานของ ตัวโปรแกรมได้ ไม่มีการจับคู่ออบเจ็คผิด จะไม่มีการให้โอเปอเรเตอร์มากหรือน้อยเกินไป การกำหนด ค่าต่างๆ สำหรับไบท์โค้ดจะต้องถูกต้องครบถ้วน และจะไม่มีการแปลงข้อมูลผิดรูปแบบ

การใช้ตัวตรวจสอบของจุดประสงค์ 2 ประการสำคัญ คือ สิ่งต่างๆ ดังที่กล่าวมาแล้วจะถูก ตรวจสอบก่อนทำให้ตัวอินเตอร์พรีเตอร์มั่นใจได้ว่า ไบท์โค้ดที่ส่งเข้าไปทำงานจะไม่มีขั้นตอนการ ทำงานที่สร้างปัญหาให้กับตัวระบบ และจุดประสงค์ที่สองก็คือ ตัวอินเตอร์พรีเตอร์จะทำงานตามไบท์ โค้ดได้รวดเร็วกว่า เพราะไม่ต้องคอยระแวงว่าจะมีปัญหาเกิดขึ้น และไม่ต้องหยุดเป็นช่วงๆ เมื่อพบปัญหา และต้องแก้ไข

ในการทำงาน ไบท์โค้ดจะถูกตรวจสอบเพียงครั้งเดียวเท่านั้น และจะทำงานไปได้ตลอดไม่ต้อง มีการตรวจสอบซ้ำอีกเมื่อมีการเรียกกลับมาทำงานที่ส่วนเดิมของ โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6 การทำงานตามโค้ด

เมื่อตัวโหลดเคอร์ได้รับรวบรวมโค้ดเข้ามาสู่ระบบทำการจัดวางในหน่วยความจำ และตัวตรวจสอบได้ทำการตรวจสอบความถูกต้องแล้ว โค้ดก็จะถูกส่งต่อไปยังตัวอินเตอร์พรีเตอร์เพื่อทำงานตามคำสั่ง การทำงานตามคำสั่งของโค้ดก็คือการเปลี่ยนโค้ดให้กลายเป็นคำสั่ง การทำงานจริงที่ตัวระบบไคลเอ็นท์ที่รันโค้ดนี้สามารถทำงานได้ ซึ่งวิธีการที่ทำได้ก็มีอยู่ 2 วิธีด้วยกันคือ ตัวอินเตอร์พรีเตอร์ทำการคอมไพล์โค้ดเหล่านี้ให้กลายเป็นเนทีฟโค้ดที่ตัวเครื่องไคลเอ็นท์เข้าใจ แล้วค่อยทำงาน เพื่อให้ได้ความเร็วสูงสุดในการทำงาน กับอีกวิธีหนึ่งก็คือ ตัวอินเตอร์พรีเตอร์อ่านโค้ดเข้ามาแล้วตีความทำงานไปที่ละคำสั่ง และทำการตีความไปเรื่อยๆ ตลอดเวลาที่มีการทำงาน

โดยปกติแล้ว ผู้สร้างตัวอินเตอร์พรีเตอร์มักจะเลือกใช้วิธีการหลัง รูปแบบของไบท์โค้ดในภาษาจาวามีความยืดหยุ่นเพียงพอที่จะสามารถเปลี่ยนไปทำงานบนเครื่องไคลเอ็นท์แบบต่างๆ ได้โดยไม่มีการก่อให้เกิดโอเวอร์เฮดมากมายเกินความจำเป็น อย่างไรก็ตามไคลเอ็นท์ของจาวาบางระบบจะมีความสามารถในการทำงานได้ทั้งสองวิธี คือ โปรแกรมเมอร์สามารถจะเลือกใช้วิธีการคอมไพล์กับงานที่เน้นการคำนวณมากๆ เพื่อเป็นการเพิ่มสมรรถนะในการทำงานให้ได้เต็มที่ ซึ่งไคลเอ็นท์แบบนี้จะให้ถึงความพอร์ทเทเบิล และสมรรถนะที่ดี

การสร้างระบบรันไทม์ที่ดีจะต้องถ่วงดุลย์ความสำคัญ 3 ประการให้ได้พอเหมาะ นั่นคือ ความพอร์ทเทเบิล ความปลอดภัย และสมรรถนะ เรื่องของความพอร์ทเทเบิลทำได้โดยการใช้รูปแบบของไบท์โค้ดที่มีความเป็นกลางเพียงพอ สามารถนำไปใช้รันบนเครื่องคอมพิวเตอร์แบบต่างๆ ได้โดยง่าย นอกจากนี้ การที่ตัวอินเตอร์พรีเตอร์ทำการกำหนดการจัดวางตำแหน่งในหน่วยความจำในช่วงรันไทม์ (แทนที่จะเป็นระหว่างการคอมไพล์เหมือนภาษาอื่นๆ) ก็เป็นการเพิ่มความแน่นอนว่า คลาสต่างๆ ที่อิมพอร์ตเข้ามาจะยังคงใช้ได้อยู่ตลอดเวลา เรื่องของความปลอดภัยนั้น เป็นสิ่งที่มีการคำนึงถึงอยู่ตลอดกระบวนการทำงานของระบบรันไทม์ โดยเฉพาะอย่างยิ่งในส่วนของตัวตรวจสอบไบท์โค้ด ที่ทำให้แน่ใจได้ว่าโปรแกรมจะทำงานได้ถูกต้องตามข้อกำหนดของจาวา ส่วนเรื่องของสมรรถนะนั้นก็สามารถจัดการได้ในสองระยะ คือ พยายามเอาโอเวอร์เฮดทั้งหลายไปใส่ไว้ที่ตอนเริ่มต้น โหลดโปรแกรมเข้ามาสู่ระบบหรือไม่ก็กำหนดให้ทำงานเป็นแบบแบ็คกราวนด์ (Back-Ground Thread)

ด้วยสิ่งต่างๆ เหล่านี้ ทำให้จาวาสามารถปล่อยสมรรถนะระดับที่น่าพอใจออกมาได้โดยที่ยังคงไว้ซึ่งความปลอดภัย และสภาพแวดล้อมที่ปลอดภัย นอกจากนี้ ยังสามารถจะดึงสมรรถนะระดับสูงสุดออกมาใช้ได้ทันทีเมื่อต้องการ

2.4.7 การสร้างและรันภาษาจาวา

การสร้างโปรแกรมจาก Java ก็เหมือนกับภาษา Programming อื่นๆ คือเขียน Code โปรแกรมจาก Editor ใดๆ ก็ได้ บันทึกอยู่ในนามสกุล .java จากนั้นจึงนำไปคอมไพล์ด้วยคอมไพเลอร์ของ Java จะได้ไฟล์ใหม่ในนามสกุล .class ที่เก็บรหัสของการคอมไพล์ไว้จาวาเรียกรูปแบบข้อมูลที่อยู่ในไฟล์ใหม่นี้ว่า Byte Code ไฟล์ที่ได้คือแอฟเพล็ตนั่นเองแอฟเพล็ตยังไม่สามารถรันได้ทันทีเหมือนไฟล์นามสกุล .exe หรือ .com ที่เรารู้กันเคยกันเพราะข้อมูลแบบ Byte-Code จะมีรูปแบบข้อมูลที่อยู่กึ่งกลางระหว่างโค้ดโปรแกรม (Source Code) กับ โค้ดที่คอมพิวเตอร์อ่านแล้วนำไปปฏิบัติงานได้ทันที (Machine Code) หาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะรันแอปพลิเคชันบนระบบใดๆ จะต้องใช้อินเทอร์พรีเตอร์จาวาของระบบนั้นๆ เพื่อตรวจสอบความถูกต้องของ Byte-Code แล้วแปลให้เป็นรหัสภาษาเครื่องส่งให้ระบบปฏิบัติการนำไปรันต่อไปขั้นตอนการทำงาน

2.5 WEB BROWSER SUPPORT JAVA

HTML (Hypertext Markup Language) เป็นภาษาที่มีรูปแบบหนึ่งของภาษา SGML (Standard Generalized Markup Language) นิยมใช้กันทั่วไปบน Internet เป็นภาษาที่ใช้ในการเขียน โปรแกรมภาษาหนึ่งของคอมพิวเตอร์มีลักษณะของโปรแกรมเป็นไฟล์ตัวอักษร ในมาตรฐานของรหัสแอสกี ประกอบด้วย Reserve Markup Directive หรือ คำสั่งต่างๆ ซึ่งมีลักษณะจะเป็นคำสั่งที่ง่ายต่อการทำความเข้าใจ เพื่อให้การสร้างเว็บเพจมีความสะดวกและง่ายมากยิ่งขึ้น ทั้งนี้เพื่อส่งผ่าน โครงสร้างของข้อมูลระหว่างผู้ใช้

ในปัจจุบันภาษา HTML ได้ถูกกำหนดมาตรฐานขึ้นมาเป็นรุ่นที่ 3.0 HyperText ที่ถูกสร้างขึ้นมา จะอยู่ในรูปของแฟ้มเอกสาร HTML (Document File) ที่มีการกำหนดคุณสมบัติ (markup) ของเว็บเพจเข้าไป ยังไม่สามารถแสดงผลข้อมูลออกมาให้ใช้งาน ได้โดยตรง ถ้าต้องการผลลัพธ์ที่เกิดจากการสร้างต้องผ่านโปรแกรมที่ทำหน้าที่แปลคำสั่งนั้นก่อน เราเรียกโปรแกรมที่ทำหน้าที่นี้ว่า “โปรแกรมเว็บเบราว์เซอร์” (Web Browser Program) หน้าที่หลักของโปรแกรมนี้คือเป็นตัวแปลคำสั่งของ HyperText แล้วแสดงผลออกมาเป็นรูปภาพ เสียงข่าวสารและข้อมูล และคุณสมบัติอื่นๆ ของโปรแกรมเว็บเบราว์เซอร์ ไม่ว่าจะเป็นการดาวน์โหลดไฟล์ การดึงรูปภาพมาใช้งาน การพิมพ์เอกสาร HTML ออกทางเครื่องพิมพ์ การส่งจดหมาย (e-mail) และความสามารถด้านอื่นๆ อีกมากมาย ทำให้เราสามารถใช้งาน Internet หรือ Intranet ได้อย่างสะดวก รวดเร็ว และสนุกสนาน ซึ่งที่พัฒนาภาษา JAVA ได้วางแผนและพัฒนาระบบ Interface โดยสังเกตเห็นถึงการเจริญเติบโตและความสามารถของภาษา JAVA จึงได้เปิดตัว WebBrowser มีชื่อว่า Hotjava เพื่อใช้สำหรับแสดงศักยภาพของ Java ที่มีอยู่อย่างเต็มที่ราวปี 2538 เพราะในขณะนั้นยังไม่มี Browser ตัวใดที่รู้จักและสามารถทำงานร่วมกับภาษา JAVA ได้เลย และแน่นอนว่า Browser ที่ได้รับการพัฒนาต่อๆ มาย่อมจะมีการเพิ่มเติมความสามารถที่ดีกว่า HotJava อย่างแน่นอนคือ Netscape Navigator เป็น Browser ที่บริษัท Netscape Communications ได้สร้างและเป็นบริษัทแรกที่ได้ลิขสิทธิ์ ภาษา JAVA จากบริษัท Sun Web Browser โดย Netscape เวอร์ชันแรกที่ Support คือ Netscape 3.0 เพื่อใช้สำหรับการทำงานบน Windows3.X, Windows95, WindowsNT, Solaris และ Apple Macintosh และยังมี Web Browser ของอีกบริษัทหนึ่งที่มีชื่อเสียงไม่แพ้ Netscape คือบริษัท Microsoft ซึ่งได้ออก WebBrowser ที่ Support Java ในชื่อ Internet Explorer 3.0 เพื่อให้ WebBrowser ที่บริษัทพัฒนาเป็นเวอร์ชันล่าสุดในขณะนั้น สามารถรองรับการออกแบบ Homepage ที่ถูกออกแบบด้วยภาษาจาวา

ปัจจุบันนี้มีบริษัทที่สร้าง WebBrowser อื่นๆ อีกเช่น AOL (America online) ,IBM ซึ่งแน่นอนว่าปัจจุบันนี้ WebBrowser เหล่านี้สามารถ Support กับ Homepage ในเครือข่าย Internet ที่ใช้ภาษา Java แน่แน่นอนเพราะปัจจุบันได้มีการนำ Java ไปประยุกต์ใช้งานในงานด้านต่างๆ ทางธุรกิจ การศึกษา บันเทิง ฯลฯ อย่างมากมาย

2.6 Java Tool

การศึกษาและพัฒนาโปรแกรมภาษาจาวานั้นในช่วงเริ่มแรกนั้นมีลักษณะที่เรียกว่า command line driven compiler เครื่องมือแรกที่ใช้ในการประยุกต์เพื่อสร้าง applet คือ Sun's Java Development Kit (JDK) ซึ่ง applets ที่ได้มานั้นมีลักษณะที่มีการเคลื่อนไหวไม่มากนักหรือ applet ที่มีความยุ่งยากไม่มากนัก และการประยุกต์เพื่อสร้าง standalone Java applications นั้นก็เช่นกัน นักพัฒนาโปรแกรมก็ยังคงต้องการเครื่องมือที่ใช้ช่วยในการศึกษาและพัฒนาโปรแกรมด้วย ในปัจจุบันนี้มีเครื่องมือที่ช่วยในการพัฒนาโปรแกรมภาษา Java ซึ่งต่างได้รวมเอาคุณสมบัติของ GUI (Graphic User Interface) ไว้ในการพัฒนา Java Application โดย Java Tool เหล่านั้น Compatible กับ Java Language Standards. เครื่องมือที่ใช้ศึกษาและพัฒนาโปรแกรมภาษา Java จึงมีความง่ายและสะดวกกว่าในการที่จะศึกษาและใช้งานภาษา Java มากขึ้น เครื่องมือสำหรับพัฒนาโปรแกรมภาษาจาวานั้นมีหลายบริษัทที่พัฒนาเครื่องมือเหล่านั้นออกมา ตัวอย่างโปรแกรมที่มีชื่อเสียงพลึงขบดังต่อไปนี้

2.6.1 JDK หรือ Java Development Kit

JDK เป็นเครื่องมือสำหรับช่วยในการสร้างและพัฒนา Java ประกอบไปด้วยไฟล์ต่างๆ ดังต่อไปนี้

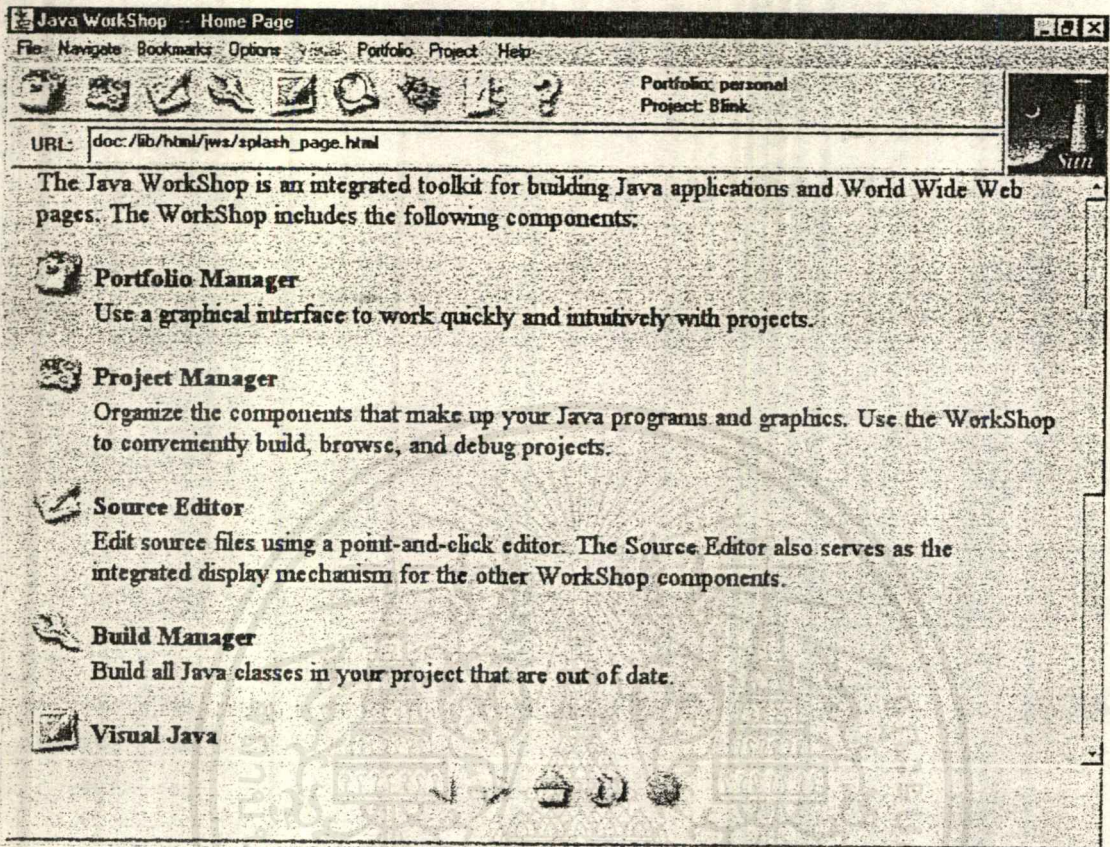
1. ไฟล์ appletviewer.exe ใช้สำหรับทดสอบ Java Applet เพื่อ View ผลการทำงาน โดยไม่ต้องสั่ง Run ผ่านทาง Browser
2. ไฟล์ java.exe เป็น Interpreter ใช้ทดสอบการทำงานของ Java Application
3. ไฟล์ javac.exe มาจากคำว่า Java Compiler ใช้สำหรับแปล Source Program ที่เราเขียนขึ้นแบบ Ascii Code (.java) เป็น Byte Code (.class)
4. ไฟล์ javadoc.exe ใช้สำหรับสร้างไฟล์เอกสาร Document จาก Source code ของ Java ในรูปแบบของ HTML ภายในจะเป็นการบอกรายละเอียดของคลาสต่างๆ ที่ผู้ใช้เขียนขึ้น กฎการเขียนที่ควรรู้ไว้กรณีที่มีผู้สนใจอยากจะทำคือ ผู้ใช้จะต้องใส่คำสั่ง `/**` เข้าไปด้วยเสมอตรงส่วนบนของโปรแกรมก่อนการประกาศคลาส ลักษณะการใช้งานจะเหมือนกับคำสั่ง `//` และ `/* */` (สำหรับใส่ Comment คำอธิบายต่างๆ ไป)
5. ไฟล์ javap.exe ใช้แปลคลาสไฟล์ของ Java ที่เป็นภาษาแอสเซมบลี กลับมาเป็น Source code

2.6.2 Sun's Java WorkShop

Java WorkShop 1.0 เป็นผลิตภัณฑ์ที่สามารถใช้ในการเขียน โปรแกรม Java และ HTML ที่สามารถแสดงผลได้ในรูปแบบหรือลักษณะที่คล้ายกับ Browser ซึ่งมาจาก Sun's HotJava Web browser. ทำให้ Java components ที่ใช้รันภายใต้ Microsoft Windows 95, Windows NT, and Sun Solaris 2.4 ซึ่งใช้ Platform เดียวกันคือ Java virtual machine (VM) และมี compiler components บางตัวที่มาจาก Sun Java

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

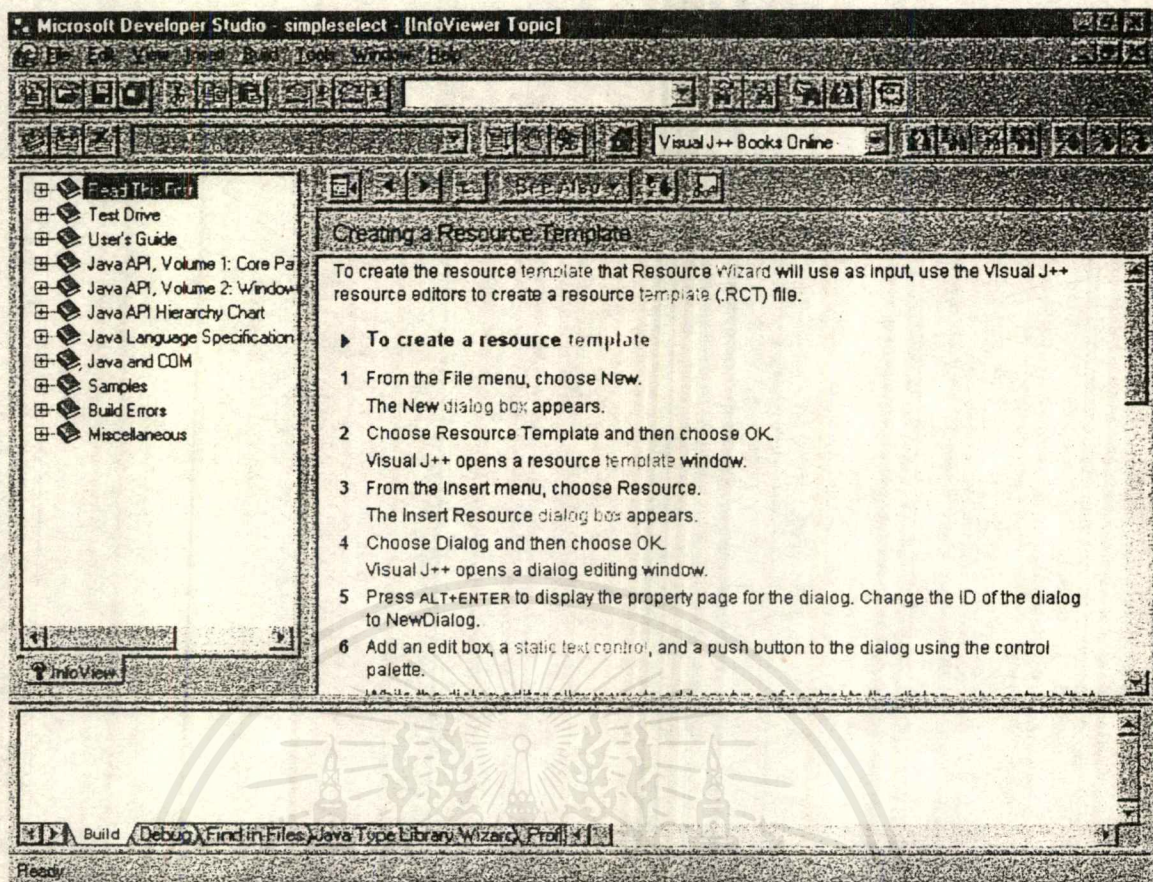
Development Kit (JDK). ซึ่งผลิตภัณฑ์นี้ต้องการใช้ Hardware ในการพัฒนาที่มีประสิทธิภาพค่อนข้างมาก เช่น ต้องการ RAM มากถึง 32MB ที่จะทำให้ประสิทธิภาพในการทำงานนั้นเป็นที่ยอมรับได้



รูปที่ 2 - 3 โปรแกรม Sun Microsystems: Java WorkShop

2.6.3 Microsoft Visual J++, Professional Edition

Visual J++ คือเครื่องมือที่ใช้พัฒนาการเขียน Java applets และ applications โดยบริษัท ไมโครซอฟท์. Visual J++ ยังมีคุณสมบัติอีกอย่างที่อยู่ใน Java Language ของ Sun ไม่มีคือในส่วนที่เรียกว่า **Component Object Model (COM)** และ **Type Library Wizard** ที่ทำงานโดย type libraries ของ COM objects (รวมถึง ActiveX controls) ที่ User สามารถที่จะ import มาใช้ได้. ด้วยเหตุนี้เอง User จึงสามารถใช้ Methods และกำหนด Properties ได้อย่างง่ายดาย



รูปที่ 2-4 โปรแกรม Microsoft Visual J++

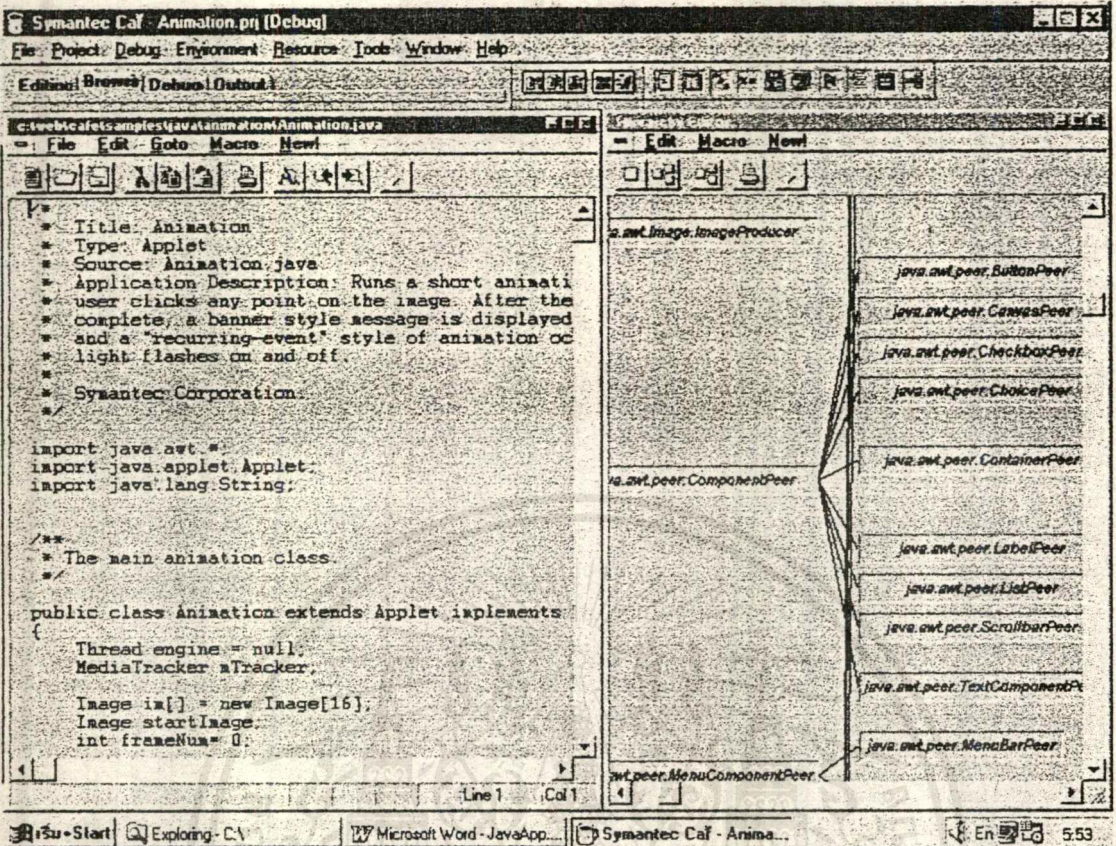
Sun JDK 1.1 ได้ถูกรวมอยู่ใน Visual J++ ตั้งแต่ version 1.0 ซึ่งปัจจุบันมีถึง version 1.1 รวมทั้ง International และ UNICODE features, Digital signing ของ applets, the JAR packaging capability (our .CAB files), Java Native Method Interface capabilities, Java Beans หรือตลอดจน Support ActiveX รวมทั้งการติดต่อกับ Database ด้วยวิธี RDO (Remote Access Object) หรือ DAO (Data Access Object)

Visual J++ ยังมีความสามารถในการด้าน visually create Java forms โดย resource editor ขอมให้ user ใช้ visually layout ในการ interface application ของ user เอง. resource editor ก็เหมือนกับ editor โดยทั่วไปที่ใช้กันอยู่ใน Microsoft Visual C++ ซึ่ง User ยังสามารถที่จะ convert เป็นรูปแบบของ Visual Basic และ Visual C++

2.6.4 Symantec Visual Cafe

Visual Cafe ก็เป็นเครื่องมืออีกชนิดหนึ่งที่มีรูปแบบการ Interface คล้ายกับ tools ตัวอื่นๆ แต่จะมี wizard ใหม่สำหรับช่วยทางด้าน component-based programming ซึ่งเรียกว่า RAD (Rapid Application Development) Tool ยกตัวอย่างเช่น Borland Delphi หรือ Microsoft Visual Basic. และ Visual Cafe ยังมี JavaBeans specification. ซึ่งนักพัฒนาโปรแกรมสามารถเขียน โปรแกรมด้วย components ที่ถูกสร้างขึ้นมาแล้ว dragging มายังรูปแบบที่ต้องการได้ สามารถ setting properties และ events ได้โดยตรง.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 - 5 โปรแกรม Symantec Visual Café

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถสรุปเป็นตารางเปรียบเทียบความสามารถของเครื่องมือในการพัฒนาภาษา Java ได้ดังต่อไปนี้

Summary of features			
y YES n NO			
Products listed in alphabetical order	Java WorkShop 1.0	Microsoft Visual J++ 1.0	Visual Café 1.0
Development Environment			
Context-sensitive help	Y	y	Y
Wizards or equivalents	Y	y	Y
Cut-and-paste from online help	N	y	Y
User can customize and reuse desktop	Y	y	Y
Third-party program launch from IDE	Y	y	Y
Include JIT compiler	N	y	Y
Supports JDK tools	Y	y	Y
Editor			
Syntax highlighting	n	y	Y
Remappable keys	n	y	Y
Macros	n	y	Y
Brief compatibility	n	y	y
Epsilon Compatibility	y	y	y
Programming Tools			
Wizard can generate applets	y	y	y
Wizard can generate applications	y	y	y
Wizard can add control events	y	n	y
Wizard adds animation automatically	n	y	y
Class browser	y	y	y
Can browse uncompiled projects	y	y	y
Graphical class-designer tool	n	n	y
GUI resource designer	y	y	y

Attaches code to controls in resource editor	y	n	y
Supports .BMP/GIF/JPG graphics files	Yyy	yyy	yyy
Can import Windows resource script files	n	y	y
HTML editor included	n	n	n
Supports ActiveX components	n	y	n
ActiveX creation in Java	N/A	y	N/A
Supports JavaBeans	n	n	y
Can target reusable components	y	y	y
Provides classes for database connectivity	n	y	n
Provides resolution-independent form design	y	y	y
Debugging			
Integrated debugger	y	y	y
Includes browser with debugging hooks	y	y	y
Debugger catches Java exceptions	y	y	y
Breakpoints	y	y	y
Watch variables	y	y	y
Support multithreaded debugging	y	y	y
Supports remote debugging	n	n	y
Integrated source control	y	y	n
Disassembles Java byte code	n	y	n
Applet Viewer	y	y	y
JavaScript/HTML debugging	nn	nn	nn

ตารางที่ 2 – 1 ตารางเปรียบเทียบเครื่องมือในการพัฒนาภาษาจาวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ความแตกต่างระหว่าง Java Applet และ Java Application

Java Applet กับ Java Application เป็นภาษาคอมพิวเตอร์ที่ถูกออกแบบมาเพื่อใช้งานบนอินเทอร์เน็ตเหมือนกัน ทำงานอยู่บนโฮมเพจร่วมกับภาษา HTML ที่ใช้สร้าง โฮมเพจเหมือนกันลักษณะที่แตกต่างกันก็คือ

- Java Applet ไม่สามารถทำงานเดี่ยวๆ ด้วยตนเองได้จะทำงานร่วมกันกับไฟล์ .html โดยเราจะเรียกใช้ไฟล์ที่เขียนจาก Java นี้ผ่านทาง Tag คำสั่ง <APPLET>...</APPLET> ภายในไฟล์ HTML จากนั้นก็จะเรียกใช้ไฟล์ โปรแกรม javac.exe มาทำการคอมไพล์เพื่อที่จะได้ไฟล์ที่มีนามสกุล .class ไว้สำหรับให้ไฟล์ HTML เรียกผ่านวิธีสั่งแสดงผลการทำงานของ Java Applet เราจะใช้โปรแกรม appletviewer.exe หรือ Web Browser เช่น Netscape, Hotjava, Internet Exploere ฯลฯ โดยเลือกใช้ตัวใดตัวหนึ่งก็ได้
- Java Application ตัวที่ 2 นี้มันจะสามารถปฏิบัติงานเดี่ยวๆ ได้ด้วยตัวเองคล้ายกับการเขียนโค้ดในภาษาโปรแกรมทั่วไป สิ่งที่เราควรจดจำไว้ให้ดีในการเขียน Java Application ก็คือ เราจะต้องมี Method ของ Main(string args[]) อย่างน้อยๆ 1 Method เพื่อใช้เป็นจุดเริ่มต้นของการโปรแกรม อันนี้ถ้าหากสังเกตให้ดีจะเหมือนกับการเขียนโปรแกรมภาษาซีต่างๆ ไป วิธีการคอมไพล์ก็จะใช้โปรแกรม javac.exe เช่นเดียวกับ Java Applet ส่วนการแสดงผลเราจะใช้โปรแกรม java.exe แทน ซึ่งจะแตกต่างจาก Java Applet ตรงจุดนี้

2.8 ความแตกต่างระหว่าง Java และ JavaScript

สามารถสรุปเป็นตารางแสดงการเปรียบเทียบระหว่าง JavaScript และ Java ได้ดังนี้

JavaScript	Java
1. ไม่มีการถูก compile ก่อน Run แต่จะ Run ได้เลยโดยใช้ interpreter บน client	1. ถูก compile ที่ Server ก่อนรันบน Client
2. Object-based, รหัสใช้ object ภายในซึ่งมีอยู่แล้วและไม่มีการประกาศ class หรือการสืบทอด class	2. Object-oriented, Applet มีการสร้าง Object และสืบทอดจาก class
3. รหัสรวมอยู่ในเนื้อ Script HTML	3. Applet จะแยกต่างหากจาก Script HTML
4. ไม่มีการประกาศชนิดของตัวแปร	4. ต้องมีการประกาศชนิดของตัวแปร
5. ตรวจสอบความถูกต้องในขณะที่รัน ถ้ามีข้อผิดพลาดเกิดขึ้น interpreter จะแจ้งให้ทราบในขณะที่ Run ไม่สามารถเขียนข้อมูลลงฮาร์ดดิสก์ได้	5. ต้องถูกตรวจสอบความถูกต้องก่อนการ Run คือในขณะที่ compile ถ้า compile ไม่ผ่านไม่สามารถรันได้ไม่สามารถเขียนข้อมูลลงฮาร์ดดิสก์ได้

ตารางที่ 2 – 2 ตารางเปรียบเทียบ Java Script และ Java

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 ทิศทางของ JAVA

JAVA ได้เริ่มต้นจากการเป็นภาษาคอมพิวเตอร์ง่าย ๆ สำหรับสร้าง Applet หรือ โปรแกรมเล็กๆ ซึ่งสามารถทำงานบนเครื่องคอมพิวเตอร์ใดๆ ก็ได้ โดยไม่จำเป็นต้องสร้าง โปรแกรมที่ใหญ่โตมากนักขึ้นมา ปัจจุบันนี้กระแสความต้องการและความนิยมของจาวากำลังสูงขึ้นอย่างรวดเร็วส่งผลให้เกิดพัฒนา โปรแกรมประยุกต์ หรือ Application ใหม่ ๆ โดยใช้ภาษา JAVA และ JAVA ไม่ได้เป็นเพียงแต่ Applet เล็กๆ อีกต่อไปแล้วเพราะเราสามารถ ใช้ JAVA เพื่อเป็นเครื่องมือสำหรับสร้างซอฟต์แวร์ขนาดใหญ่ ซึ่งผู้ที่กำหนดทิศทางหรือมองภาพของ JAVA ได้ดีในอนาคตอันใกล้ก็คือผู้อยู่เบื้องหลังความสำเร็จของ JAVA ซึ่งบทความต่อไปนี้เป็น การสรุปบทความเกี่ยวกับบุคคล 2 ท่านซึ่งบุคคลแรกคือผู้ให้กำเนิด JAVA คือ Jame Gosling และบุคคลที่สองคือผู้ที่ทำให้ชาวโลกได้รู้จักกับ JAVA โดยเข้ามาพัฒนาระบบการตลาด และสนับสนุนโปรดัคชั่นต่างๆ ของเทคโนโลยี JAVA ให้ออกไปสู่ลูกค้าทั่วโลก คือ Dr.Alan Baratz ประธานบริษัท JavaSoft ซึ่งบทความนี้ได้เรียบเรียงสรุปจากวารสารทางอินเทอร์เน็ตที่ชื่อ JavaWorld ดังต่อไปนี้

ในปี 2539 ที่ผ่านมาเป็นปีแห่งการพัฒนาซอฟต์แวร์ประเภท Client หลายต่อหลายฝ่าย ต่างพุ่งเป้าความสนใจไปที่เครื่องคอมพิวเตอร์ประเภทโต้ตอบกับผู้ใช้ในฝั่ง Client กับ Application ต่างๆ ที่สำหรับใช้ภายในองค์กร ในปี 2540 จะเป็นปีทองที่ Software จะก้าวไปไกลมากกว่า Client โดยจะเคลื่อนที่เข้าไปใกล้ชิดกับผู้ใช้งานยิ่งขึ้นในรูปแบบของอุปกรณ์ส่วนบุคคลต่าง ๆ โดยการใช้ JAVA ควบคุม อุปกรณ์ไฟฟ้า และเครื่องจักรขนาดเล็ก เช่น PDA (Personal Digital Assistant) หรือ เครื่องมือช่วยระบบ ดิจิตอลส่วนบุคคล, อุปกรณ์ Set-top Boxes (กล่องควบคุมอนเนกประสงค์) Smart Phones, ระบบฝังตัวใน อุปกรณ์ต่างๆ เช่นใน เครื่องวิทยุติดตามตัว, เครื่องพิมพ์, เครื่องส่งแฟกซ์, , โทรศัพท์เว็บ, โทรศัพท์เว็บ บราวเซอร์, รวมไปถึงเครื่องใช้ไฟฟ้าต่างๆ ซึ่งอุปกรณ์เหล่านี้จะถูกเชื่อมการติดต่อกับระบบ Smart Cards นั้นหมายถึงว่าผู้บริโภคสามารถที่จะสั่ง โปรแกรมควบคุมระยะไกลกับสวิตช์ไฟฟ้าผ่านทางอุปกรณ์ ช่วย (PDAs) เล็กๆ เหล่านี้ ซึ่งขณะนี้มันักประดิษฐ์จำนวนมากต่างก็พยายามที่จะสร้างสรรค์อุปกรณ์ แปลกๆ ใหม่ ๆ ขึ้นมา ซึ่งจะทำให้ JAVA มีศักยภาพสูงสุด ในการเป็นเครื่องมือสนับสนุนอุปกรณ์เหล่านี้

บทที่ 3

Java DataBase Connectivity

ภาษาจาวาเป็นภาษาคอมพิวเตอร์ที่ได้รับการพัฒนาอย่างรวดเร็วและต่อเนื่องนอกเหนือจากการสร้างกราฟิกหรือองค์ประกอบที่ใช้ในการประดับโฮมเพจ (Homepage) ยังมีการพัฒนาองค์ประกอบใหม่ ๆ ในการปฏิบัติงานในด้านต่างๆ ได้ถูกเพิ่มเติมประสิทธิภาพให้แก่จาวาคือ JDBC (Java Database Connectivity) ซึ่งทำหน้าที่อำนวยความสะดวกในการเชื่อมต่อและสื่อสารกับระบบฐานข้อมูล โดยพื้นฐานของภาษาจาวาที่ได้รับการออกแบบมาเพื่อปฏิบัติงานกับระบบเครือข่ายคอมพิวเตอร์โดยตรง องค์ประกอบนี้จึงทำให้จาวาเหมาะสมเป็นอย่างยิ่งในการเป็นเครื่องมือสำหรับสร้าง โปรแกรมประยุกต์เพื่อเข้าถึงข้อมูลแบบไคลเอนท์ที่เซิร์ฟเวอร์ โดยมี JDBC เป็นองค์ประกอบในการเชื่อมในการถ่ายทอดคำสั่งการเรียกค้นข้อมูลจากไคลเอนท์ที่ ไปสู่ เซิร์ฟเวอร์ และรับข้อมูลผลลัพธ์จาก เซิร์ฟเวอร์ กลับคืน ไปสู่ไคลเอนท์

ระบบไคลเอนท์ที่เซิร์ฟเวอร์ (Client-Server) ในปัจจุบันมีการเพิ่มขึ้นของค่าใช้จ่ายของบุคลากรที่ มาทำหน้าที่ดูแลบำรุงรักษา Database Server ซึ่งได้แก่บุคลากรด้านการจัดการและด้านบริการสนับสนุน สำหรับ Network ขนาดเล็ก (โดยทั่วไปน้อยกว่า 20 ผู้ใช้) โดยปกติผู้จัดการ Network เพียงคนเดียวสามารถ รับภาระหน้าที่ดูแลบำรุงรักษา Database Server ควบคุมการเข้าใช้ Database Server จากบรรดาผู้ใช้ทั้งหลาย และให้บริการสนับสนุน Front End เมื่อจำนวนผู้ใช้งานข้อมูลเพิ่มจำนวนขึ้น หรือฐานข้อมูลคือตัว ของมันเองมีขนาดใหญ่ขึ้น โดยปกติแล้วก็เริ่มจำเป็นที่ต้องจ้างผู้จัดการฐานข้อมูลเพิ่มอีกคนเพื่อมาทำ หน้าที่รันระบบจัดการฐานข้อมูลและให้บริการสนับสนุนในส่วน Front End ด้วย นอกจากนี้แล้วในกรณี ที่ระบบจัดการฐานข้อมูลรันอยู่บนระบบปฏิบัติการที่บุคลากรด้านบริการสนับสนุนไม่คุ้นเคยนั้นก็ต้องการอบรม ซึ่งเป็นค่าใช้จ่ายที่ต้องเพิ่มเข้าไปในค่าใช้จ่ายเริ่มแรกด้วย

นอกจากนี้แล้วค่าใช้จ่ายด้านฮาร์ดแวร์ก็เพิ่มขึ้นด้วย ในขณะที่มีฐานข้อมูล ไคลเอนท์ที่เซิร์ฟเวอร์ เป็นจำนวนมากที่รันอยู่ภายใต้ระบบปฏิบัติการที่นิยมใช้กันเช่น Netware OS2 และ Unix เป็นต้น และ บริษัทผู้ขายส่วนใหญ่ก็ประกาศว่าระบบจัดการฐานข้อมูลของตนเองนั้นสามารถรันอยู่บนฮาร์ดแวร์เดียวกับที่ File Server Software รันอยู่ได้ก็ตาม แต่ว่าโดยปกติแล้วถ้าต้องการสมรรถนะที่สูงและความถูกต้อง ของข้อมูลที่สูงก็สมควรให้ Database Server แยกไปรันบนคอมพิวเตอร์เฉพาะกิจของ Database เอง สิ่งนี้ หมายความว่าต้องทำการซื้อคอมพิวเตอร์เพิ่มอีกเครื่อง เพื่อมาทำเป็น Database Server โดยเฉพาะ คอมพิวเตอร์นี้ต้องมีกำลังและประสิทธิภาพสูง มีแรมเป็นจำนวนมากและมีฮาร์ดดิสก์ขนาดใหญ่ พร้อมทั้งอุปกรณ์สนับสนุนเช่น เครื่องป้องกัน ไฟฟ้า (UPS) เพื่อป้องกัน เซิร์ฟเวอร์ เสียหายจากเหตุการณ์ไฟตก หรือ ไฟดับ

ค่าใช้จ่ายโดยรวมของ Software ของระบบไคลเอนท์ที่เซิร์ฟเวอร์ ปกติแล้วสูงกว่าระบบจัดการ ฐานข้อมูลแบบหลายผู้ใช้งานพีซี ค่าใช้จ่ายต่อหนึ่ง เซิร์ฟเวอร์ สำหรับฐานข้อมูลไคลเอนท์ที่เซิร์ฟเวอร์ ที่ สามารถบริการผู้ใช้ได้แบบไม่จำกัดจำนวน มีราคาสูง บวกกับค่าใช้จ่ายแยกออกไปของ front-end หรือ

เครื่องมือพัฒนา software พร้อมทั้งค่าใช้จ่ายบุคลากรสำหรับการฝึกอบรมโปรแกรมเมอร์ในระบบใหม่ กล่าวได้ว่าค่าใช้จ่ายของระบบไคลเอนท์ที่เซอร์เวอร์ สูงกว่าของระบบจัดการฐานข้อมูลบนพีซีมาก

ยังมีเรื่องที่น่ายุ่งยากอีกเรื่องหนึ่งคือ ระบบไคลเอนท์ที่เซอร์เวอร์ นั้นมีส่วนประกอบต่างๆ เป็นจำนวนมากตามหลักที่ไปที่กล่าวว่า ยิ่งระบบใดมีส่วนประกอบเป็นจำนวนมากโอกาสที่ส่วนประกอบต่างๆ ในระบบนั้นจะเสียก็ยิ่งมีมากตามไปด้วย เมื่อเกิดเหตุการณ์ร้ายแรงและระบบเกิดพังเสียหายก็เป็นการยากที่จะชี้ชัดถึงปัญหาที่แท้จริงได้ ระยะเวลาที่ใช้ในการทำให้ทุกสิ่งทุกอย่างกลับสู่สภาพปกติและเริ่มทำงานได้อีกก็เป็นเวลาที่ยาวนาน สิ่งต่างๆ ที่เกิดขึ้นเช่นนี้มาจากการขาดประสบการณ์และความชำนาญของบุคลากรด้านบริการสนับสนุนและโปรแกรมเมอร์ ทั้งนี้เนื่องจากไคลเอนท์ที่เซอร์เวอร์ เป็นเทคโนโลยีใหม่ เมื่อเวลาผ่านไปและระบบไคลเอนท์ที่เซอร์เวอร์ เริ่มเป็นที่นิยมใช้มากขึ้นปัญหาที่ได้กล่าวมาแล้วก็จะลดน้อยลง

องค์ประกอบทางด้าน Hardware และ Software รวมกัน ซึ่งระบบจัดการฐานข้อมูล ไคลเอนท์ที่เซอร์เวอร์ รันอยู่ใน Platform ต่างๆ เช่น พีซี, Unix Workstation, Super Server, Minicomputer, Mainframe ซึ่งพีซีเป็น Platform ที่นิยมใช้กันมากที่สุดสำหรับไคลเอนท์ที่เซอร์เวอร์

บรรดาระบบฮาร์ดแวร์ต่างๆ ทั้งหลายก็มีความแตกต่างทั้งในเรื่องคุณลักษณะและความสามารถ อย่างไรก็ตามก็มีคุณลักษณะร่วมที่จำเป็นสำหรับระบบปฏิบัติการ โดยปกติแล้ว Application ถูกเขียนให้รันภายใต้ระบบปฏิบัติการใดระบบหนึ่งโดยเฉพาะ ซอฟต์แวร์ระบบปฏิบัติการที่นิยมใช้กัน เช่น MS-Dos, PC-Dos, OS/2, Unix, VMS ของ Digital, MVS/XA ที่รันบน IBM Mainframe เป็นต้น

คุณลักษณะเบื้องต้นของระบบปฏิบัติการที่จำเป็นสำหรับระบบจัดการฐานข้อมูล ไคลเอนท์ที่เซอร์เวอร์ คือ ความสามารถในการทำ Multitasking หรือการรัน Application หลาย Application พร้อมกัน Multitasking ทำให้ซอฟต์แวร์ระบบจัดการฐานข้อมูลสามารถบริการการสอบถามต่างๆ และคำขอต่างๆ ของบรรดาผู้ใช้ทั้งหลายได้อย่างถูกต้อง โดยปราศจากการรบกวนซึ่งกันและกัน multitasking ทำงานโดยแบ่งเวลาประมวลผลของซีพียูออกเป็นช่วงๆ (time Slices) ให้แก่บรรดา Task ต่างๆ และบรรดา Process ต่างๆ ระบบปฏิบัติการ Multitasking ทั้งที่เป็นแบบ Preemptive และแบบ Non-Preemptive

นอกจากนี้บางระบบปฏิบัติการยังสามารถเป็นระบบ Multi-user ได้ (คือยอมให้มีผู้ใช้หลายคนทำงานที่แตกต่างกันได้พร้อมกัน) โดยเฉพาะเมื่อมีการใช้เทอร์มินอลธรรมดาทั้งหลายเป็นทางเข้าสู่ฐานข้อมูล ไอเอสแบบ multi-user ไม่ได้ ก่อให้เกิดข้อผิดพลาดโดยเฉพาะเมื่อถูกใช้เป็น Platform สำหรับระบบจัดการฐานข้อมูล ไคลเอนท์ที่เซอร์เวอร์

การเพิ่มความสามารถด้าน Multithread เข้าไปในซอฟต์แวร์ไอเอส Multitask แบบ Preemptive ความสามารถ Multithread ทำให้ Application ทำ Multitask ได้ภายในตัวเองได้ ตัวอย่างเช่น ระบบจัดการฐานข้อมูลผู้ใช้คนเดียวแบบ Multithread สามารถทำการ Start ใหม่ (Process หรือ Task) ให้ทำงานออกรายงานที่ซับซ้อนโดยทำอยู่ใน Background ในขณะที่ผู้ใช้กำลังทำการสอบถาม ซึ่งทำอยู่ใน foreground Multithread เป็นปัจจัยที่สำคัญสำหรับการออกแบบระบบจัดการฐานข้อมูล ไคลเอนท์ที่เซอร์เวอร์ ที่ซับซ้อน Multithread ทำให้ Application สามารถควบคุมให้ Task ใหม่เริ่มทำงานหรือหยุดทำงานได้ Application สามารถถูกออกแบบให้มีความชาญฉลาดภายในที่ตัดสินใจว่า Process ใดหรือ Task ใดมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับความสำคัญและควรได้รับเวลาซีทียูมาก ตัวอย่างเช่น ระบบจัดการฐานข้อมูลให้การแก้ไขข้อมูลมีลำดับความสำคัญสูงกว่าการสอบถามข้อมูล เป็นต้น

เนื่องจากจุดด้อยในเรื่องการไม่มี Platform ใดที่สามารถสนองความต้องการได้ทุกประการหรือเหมาะสมในทุกสถานการณ์ในระบบไคลเอนท์เซิร์ฟเวอร์ ในปัจจุบันมีการประยุกต์ใช้ระบบที่เรียกว่า “Network Computing” เพื่อช่วยแก้ไขในปัญหาของระบบไคลเอนท์เซิร์ฟเวอร์ ในเรื่องการใช้บุคลากรในการดูแลระบบน้อยลงคือจะไม่ต้องไปดูแลในส่วนของไคลเอนท์ ที่ในอนาคตจะมีจำนวนเพิ่มขึ้นเรื่อยๆ การง่ายในการปรับปรุงระบบไม่ต้องคอยตามแก้ไขที่ทุกเครื่องไคลเอนท์ และจากปัญหาการทำงานไม่ครบทุก Platform ในปัจจุบันมีภาษาคอมพิวเตอร์หนึ่งที่สามารถเขียนโปรแกรมคอมพิวเตอร์โดยไม่ว่าจนถึง Platform ประกอบกับความสามารถทางด้านระบบ Network Computing ที่ภาษานี้สามารถนำมาประยุกต์ใช้ได้อย่างดี ภาษาดังกล่าวคือ ภาษาจาวา

3.1 ความหมายของ JDBC

JDBC (Java DataBase Connectivity) ถูกพัฒนาโดย JavaSoft Department ของบริษัท Sun Microsystems ซึ่งก็คือฟังก์ชันมาตรฐานหรือ Java Application Programming Interface (API) สำหรับการเชื่อมต่อกับระบบฐานข้อมูล. นักพัฒนาสามารถใช้ JDBC API และยังสามารถประกอบด้วย packages อื่นๆ ด้วย ซึ่งนำเสนอในรูปแบบ function พิเศษหรือ higher level API เพื่อเขียนโปรแกรมอิสระในการกำหนด DBMS หรือ database connectivity mechanism ในการเริ่มใช้งาน. โดยทั่วไปการใช้ SQL database ในการติดต่อ (Interface) กับ framework เพื่อที่จะจัดมาตรฐานในการติดต่อในส่วนบนสุดของชนิดต่างๆ ของ database connectivity modules ซึ่งก็คือมาตรฐานของ ANSI SQL-2 Entry level database. เพราะว่า relational databases เกือบจะทั้งหมดในปัจจุบันใช้มาตรฐานของ SQL-2 Entry level

JDBC สร้างระดับการเชื่อมต่อเพื่อสื่อสารกับฐานข้อมูลในรูปแบบที่คล้ายคลึงกับ ODBC DataBase Connectivity ของบริษัท Microsoft) ซึ่งในปัจจุบันได้ถือว่าเป็นมาตรฐานของการเชื่อมต่อกับฐานข้อมูลสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล (PC) และระบบเครือข่ายท้องถิ่น (LAN) แต่ตามหลักการทำงานของทั้ง JDBC และ ODBC ตั้งอยู่บนมาตรฐานเดียวกันคือ X/Open SQL Call-Level Interface ของระบบ X-Windows และ JDBC driver ต้องเข้ากันได้กับระดับมาตรฐานในการเข้าถึง SQL (ANSI SQL Entry Level Standard) และต้องผ่าน Conformance test ซึ่ง JavaSoft เป็นผู้กำหนดขึ้น

3.2 รูปแบบการเชื่อมต่อฐานข้อมูลของ JDBC

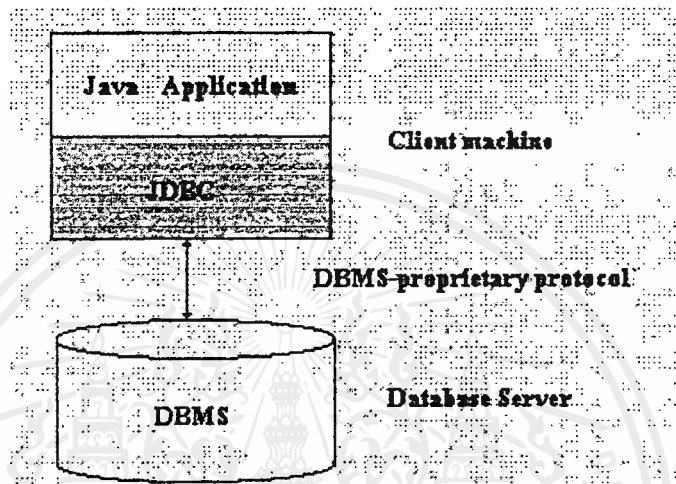
JDBC API สนับสนุนรูปแบบการเชื่อมต่อฐานข้อมูลทั้งแบบ Two-Tier model และ Three-Tier model

Two-Tier model

จาวาแอปเพล็ต (Applet) หรือจาวาแอปพลิเคชัน (Application) จะติดต่อกับฐานข้อมูลโดยตรง จึงมีความจำเป็นที่โปรแกรมจาวาต้องการ JDBC ไดรฟ์เวอร์พิเศษที่สามารถสื่อสารกับระบบจัดการฐานข้อมูลชนิดนั้นได้ รูปแบบการเชื่อมต่อแสดงดังรูปที่ 3-1 คำสั่งในการเรียกค้นข้อมูลในรูปแบบของภาษา SQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

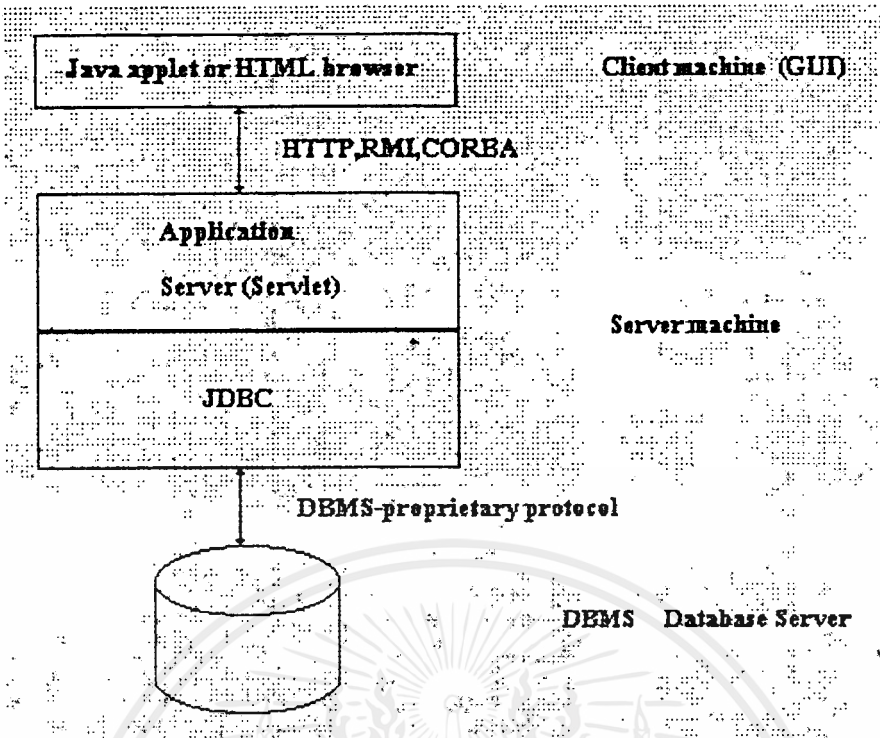
(Structured Query Language) จะถูกส่งจากผู้ใช้ไปสู่ฐานข้อมูล หลังจากนั้นผลจากการประมวลผลของระบบจัดการฐานข้อมูล ก็จะถูกส่งกลับมาสู่ผู้ใช้ฐานข้อมูลดังกล่าวนี้ส่วนมากจะติดตั้งอยู่ต่างเครื่องคอมพิวเตอร์ โดยผู้ใช้สามารถเชื่อมต่อ ผ่านระบบเน็ตเวิร์ค (Network) รูปแบบ Two Tier นี้ใช้หลักการทำงานเช่นเดียวกับรูปแบบไคลเอนท์/เซิร์ฟเวอร์ ที่เรารู้จักกันดี โดยที่เครื่องคอมพิวเตอร์ของผู้ใช้คือไคลเอนท์และเครื่องคอมพิวเตอร์ ที่ให้บริการฐานข้อมูลคือเซิร์ฟเวอร์ เครื่องข่ายคอมพิวเตอร์ที่นิยมใช้รูปแบบ Two-Tier มักจะเป็นเครือข่ายอินทราเน็ต (Intranet) สำหรับดำเนินธุรกรรมภายในองค์กร



รูปที่ 3 - 1 Two-Tier Model

Three-Tier model

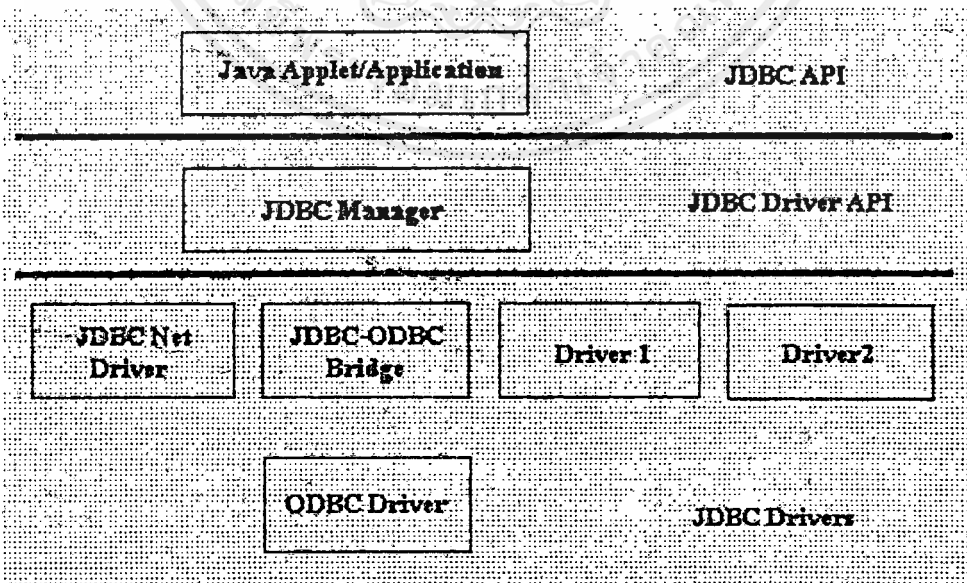
คำสั่งเรียกค้นต่างๆ จากผู้ใช้จะถูกส่งไปให้กับ Middle Tier หรือส่วนกลางของการบริการเสียก่อน หลังจากนั้น Middle Tier จะแปลงคำสั่งเหล่านี้ให้เป็นภาษา SQL เพื่อส่งไปให้กับระบบจัดการฐานข้อมูลเพื่อทำการประมวลผล ข้อมูลผลลัพธ์ที่ได้ก็จะถูกส่งกลับคืนไปให้กับ Middle Tier และส่งต่อไปให้ผู้ใช้ในที่สุด หลักการทำงานเช่นนี้มักจะพบในเครือข่ายอินเทอร์เน็ตซึ่งประกอบด้วยเครื่องคอมพิวเตอร์หลากหลายชนิด และเว็บเซิร์ฟเวอร์ (Web-Server) ซึ่งทำหน้าที่เป็น Middle Tier ก็จะเป็นตัวกลางในการจัดการให้คอมพิวเตอร์ทั้งไคลเอนท์ และเซิร์ฟเวอร์ฐานข้อมูลสามารถพูดคุยกัน ได้การปรับเปลี่ยนระบบคอมพิวเตอร์ของไคลเอนท์และเซิร์ฟเวอร์ หรือแม้กระทั่งการเปลี่ยนฐานข้อมูลตัวใหม่จะไม่มีผลกระทบต่อซึ่งกันและกันเกิดขึ้น



รูปที่ 3 - 2 Three-Tier Model

3.3 โครงสร้างของ JDBC

โครงสร้างการเชื่อมต่อภายใน JDBC ประกอบด้วย 3 ระดับหลักคือ JDBC API, JDBC Driver API และ JDBC Driver ดังรูปที่ 3-3 ระดับบนสุด JDBC API เป็นระดับของฟังก์ชัน API ที่อำนวยความสะดวกให้แก่โปรแกรมประยุกต์ ระดับกลาง JDBC Driver (มีไคร์ฟเวอร์ที่ต่างกันอยู่ 4 ชนิด) ที่เหมาะสม



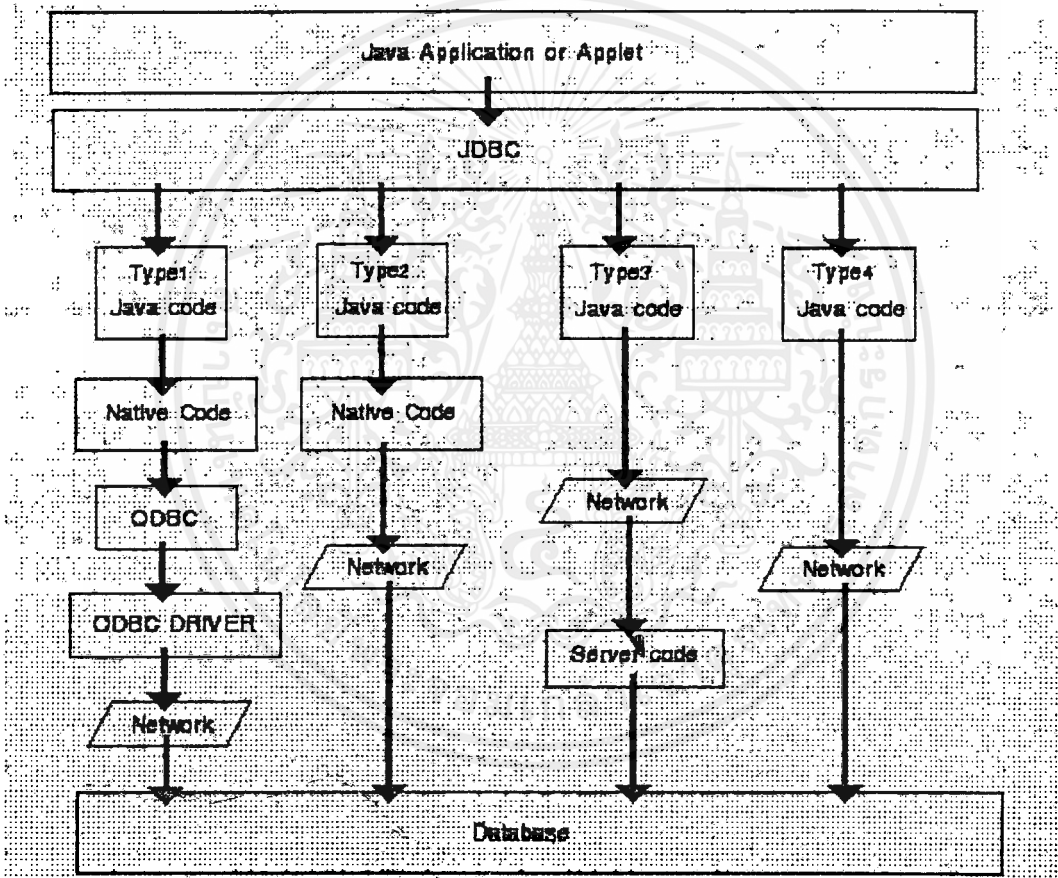
รูปที่ 3 - 3 ระดับการเชื่อมต่อของ JDBC API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 รูปแบบของ JDBC ไดรฟ์เวอร์ (Driver)

JDBC ไดรฟ์เวอร์หรือส่วนที่อยู่เบื้องหลังการทำงานของฟังก์ชัน API ต่างๆ ของ JDBC ถูกจัดแบ่งออกเป็นทั้งหมด 4 ชนิดคือ

1. JDBC-ODBC Bridge
2. Native-API (Partly-Java) Drivers
3. Net-Protocol (All-Java) Drivers
4. Native-Protocol (All-Java) Drivers

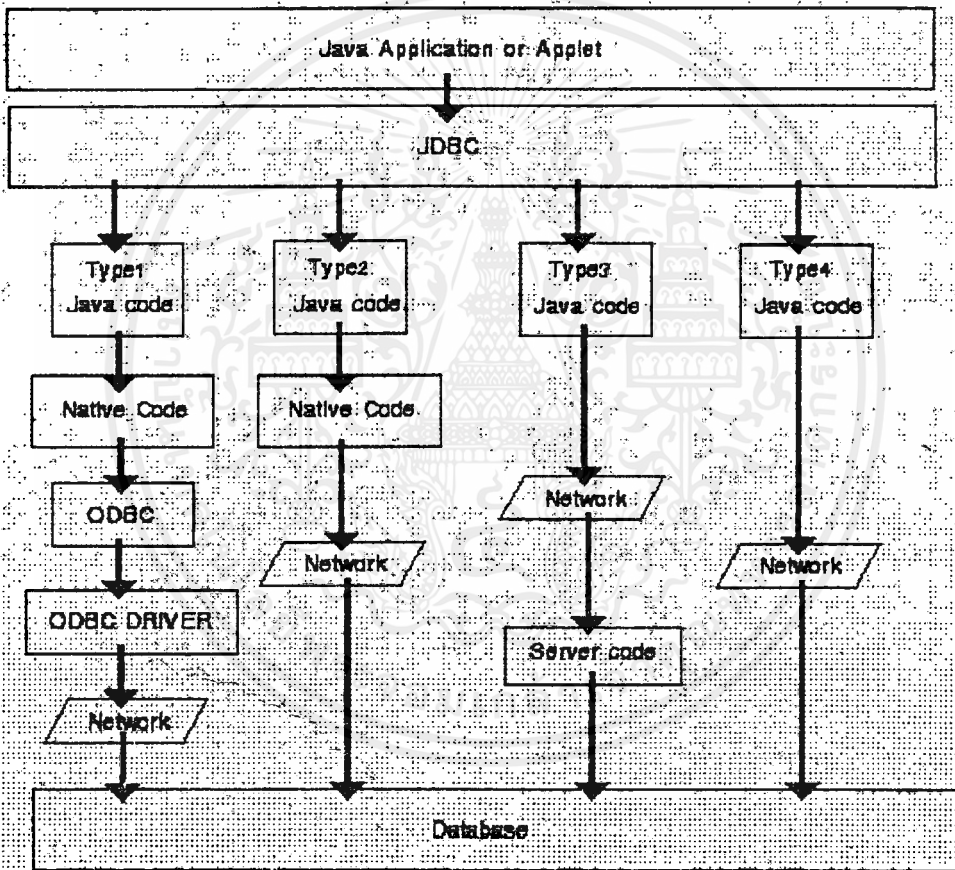


รูปที่ 3-4 JDBC driver types

3.4 รูปแบบของ JDBC ไดรฟ์เวอร์ (Driver)

JDBC ไดรฟ์เวอร์หรือส่วนที่อยู่เบื้องหลังการทำงานของฟังก์ชัน API ต่างๆ ของ JDBC ถูกจัดแบ่งออกเป็นทั้งหมด 4 ชนิดคือ

1. JDBC-ODBC Bridge
2. Native-API (Partly-Java) Drivers
3. Net-Protocol (All-Java) Drivers
4. Native-Protocol (All-Java) Drivers



รูปที่ 3-4 JDBC driver types

รายละเอียดการทำงานของไคร์ฟเวอร์แต่ละชนิดอธิบายไว้ดังนี้

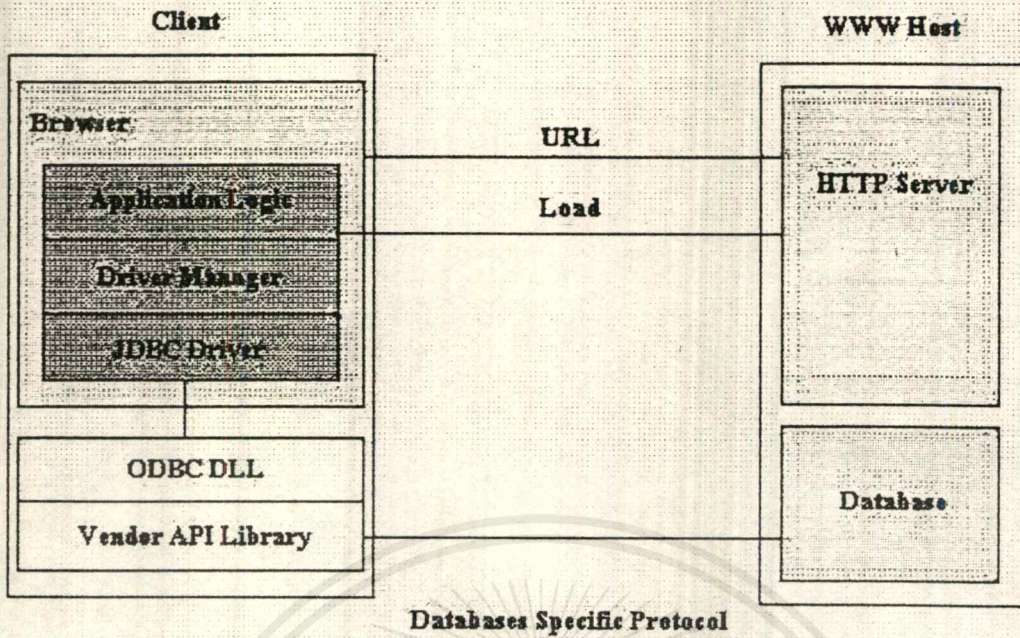
3.4.1 JDBC/ODBC bridge

JDBC/ODBC Bridge ถูกพัฒนาโดย JavaSoft และ Intersolv ทำหน้าที่เป็นตัวกลางในการเข้าถึงฐานข้อมูลได้โดยผ่านการทำงานของ ODBC โดชนำข้อดีของ ODBC-enabled data sources ที่มีใช้อยู่โดยทั่วไปอย่างมากมาย. ฟังก์ชันไคลเอนท์ Java applet หรือ application จะถูกเขียนโดยใช้ JDBC API. Bridge จะทำการแปลงการเรียกใช้ JDBC ไปยัง ODBC และส่งส่งค่า ODBC driver ที่เหมาะสมสำหรับ back-end database

ข้อดีของ bridge ทำให้ application สามารถติดต่อกับ databases ได้อย่างง่ายดายโดยจาก vendors ที่มากมายโดยเลือก ODBC driver ที่เหมาะสม อย่างไรก็ตามการติดต่อกับ database ประเภทนี้ต้องพิจารณาค่าใช้จ่าย (Overhead) และ ความซับซ้อน (Complexity) เพราะว่าการเรียกใช้จะมีลำดับการเรียกใช้ดังนี้คือ

JDBC ↔ bridge ↔ ODBC ↔ Native client-API ↔ database

การติดต่อกับใช้ประเภทนี้ชนิดของ Driver จะไม่อนุญาตสำหรับการย้ายข้อมูลแบบ just-in-time ของ Java applet . Native code ต้องมีการเตรียมการติดตั้งบนเครื่องไคลเอนท์ โดยตรงเพราะว่า JDBC-ODBC Bridge เป็นไคร์ฟเวอร์ที่ถูกเขียนขึ้นมาจากภาษา C/C++ ในบางส่วนจึงทำให้ไม่สามารถดาวน์โหลด (Download) ข้ามเครือข่ายอินเทอร์เน็ตและแปล (Interpret) บนเว็บเบราว์เซอร์ (Webbrowser) ได้ เพราะภาษา C/C++ ต้องรันเฉพาะแต่ละระบบเท่านั้น ดังนั้น ODBC ไคร์ฟเวอร์ในรูปแบบของไลบรารีไฟล์หรือ DLL (Dynamic Link Library) และ JDBC-ODBC Bridge ไคร์ฟเวอร์ต้องได้รับการติดตั้งบนไคลเอนท์แต่ละเครื่องเพื่อให้สามารถติดต่อกับฐานข้อมูลที่รู้จักได้ มีความต้องการ ในการเตรียมการติดตั้ง software โดย Administrative รูปแบบเหมือนไคลเอนท์-server application ดังนั้น JDBC/ODBC Bridge จะไม่สามารถแก้ไขปัญหาของไคลเอนท์ที่ program ดังกล่าวได้จึงเหมาะสำหรับการใช้งานกับระบบ Three-Tier โดยไคร์ฟเวอร์ทั้งสองจะได้รับการติดตั้งอยู่ที่ Middle-Tier หรือเครื่องคอมพิวเตอร์ตัวกลางเช่น Web-Server เป็นต้น



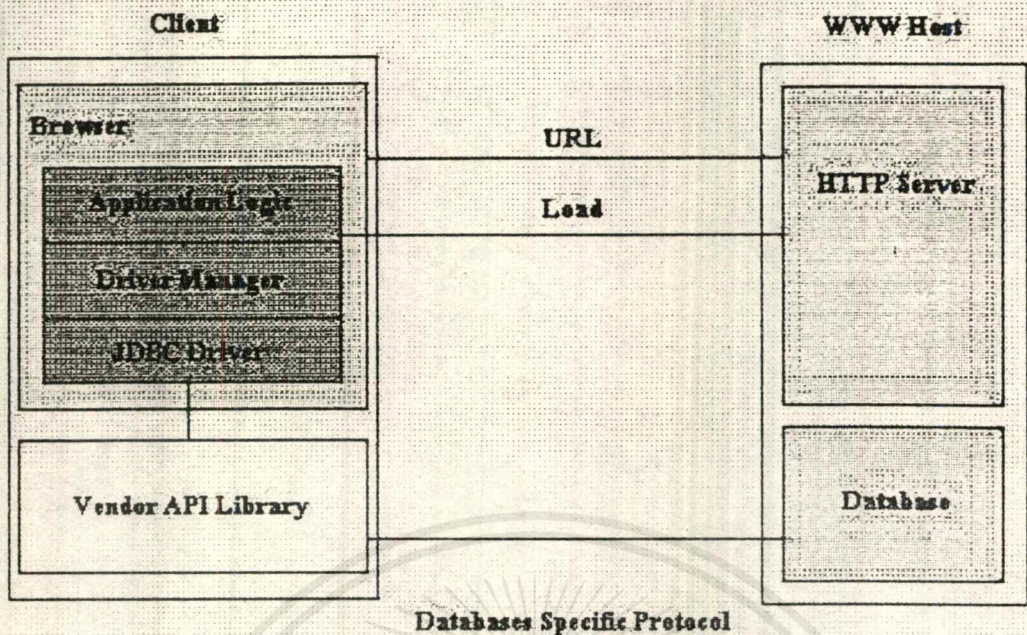
รูปที่ 3-5 การติดต่อ JDBC โดยรูปแบบที่ 1 JDBC-ODBC Bridge

3.4.2 Native-API, partly Java driver

Native-API, partly Java driver เป็น two-tier ที่ซึ่ง JDBC driver ต้องการ Vendor Library ในการแปลง JDBC function ไปยังคุณลักษณะของการใช้ภาษาในการ Query เรียกใช้ DBMS ยกตัวอย่างเช่น library สำหรับ Sybase คือ dblib , สำหรับ Oracle คือ ocilib ซึ่ง driver เหล่านี้จะถูกใช้เพื่อเขียนในการรวมของภาษา Java และ C/C++ การติดต่อโดย JDBC ประเภทนี้จะเหมือนกับ JDBC/ODBC Bridge คือต้องการ code ซึ่งก็คือ vendor library เพื่อที่จะใช้ในการติดตั้งบนเครื่องไคลเอนท์ ดังนั้นต้องมีการบำรุงรักษา (maintenance) ปัญหาเหมือนกับ bridge ใดๆก็ตาม Driver ประเภทนี้จะเร็วกว่าประเภทแรก เพราะว่ามี Layer พิเศษของการแปลงเป็น ODBC ถูกกำจัดออกไป

JDBC \leftrightarrow Native client-API \leftrightarrow database

ข้อดีของไครฟ์เวอร์ชนิดนี้คือ สามารถใช้ Native-API Driver ติดต่อกับฐานข้อมูลนั้นๆ โดยตรง โดยผ่านโปรโตคอลเดิมที่ใช้อยู่ก่อนแล้ว ทำให้เหมาะสมกับการเชื่อมต่อกับฐานข้อมูลแบบ Two-Tier Client/Server แต่ก็มีข้อเสียที่คล้ายคลึงกับ JDBC-ODBC Bridge คือต้องติดตั้งไครฟ์เวอร์ (DLL) และโปรโตคอล ไครฟ์เวอร์เช่น SQL-NET, INET, Open Client, ฯลฯ บนเครื่องคอมพิวเตอร์นั้นๆ ที่สำคัญส่วนหนึ่งของ Native-API Driver ถูกเขียนขึ้นจากภาษา C++ (Partly-Java) จึงไม่สามารถดาวน์โหลดผ่านเครือข่ายอินเทอร์เน็ตได้และไม่สามารถเชื่อมต่อฐานข้อมูลข้ามชนิดกันได้



รูปที่ 3-6 การติดต่อ JDBC โดยรูปแบบที่ 2 Native-API, partly Java driver

3.4.3 Network-protocol, all-Java driver

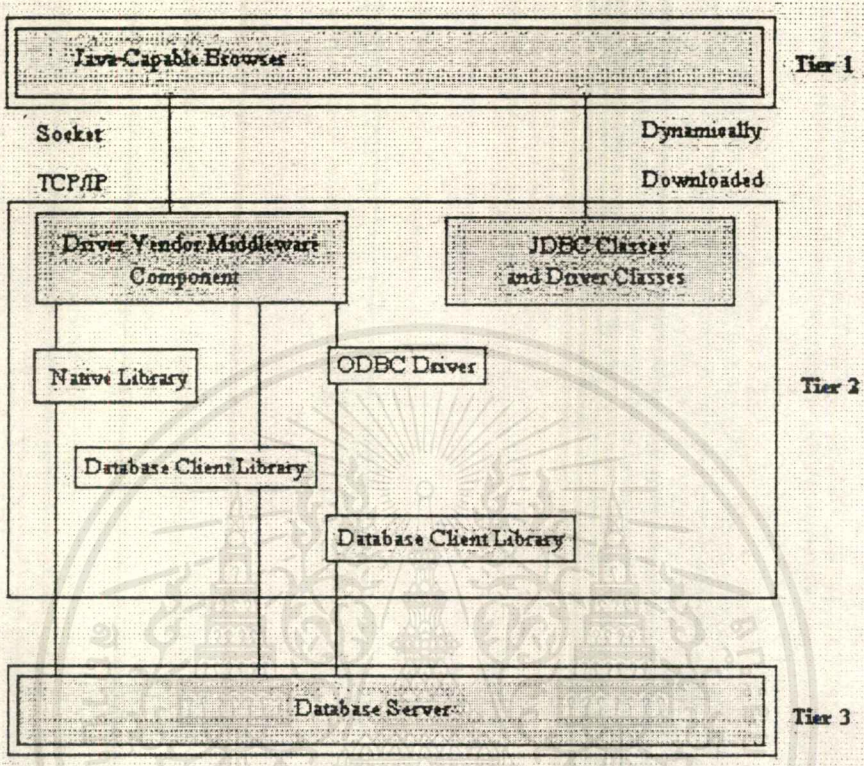
Network-protocol, all-Java ประเภทนี้ driver จะทำการแปลงการเรียกใช้ JDBC ให้อยู่ในรูปของเน็ตเวิร์คโปรโตคอลร่วม (DBMS-independent network Protocol) ซึ่งหลังจากนั้นจะถูกแปลงให้อยู่ในรูปเฉพาะของแต่ละฐานข้อมูล (database-specific API) บนเซิร์ฟเวอร์นั้นๆ Middle-tier server รูปแบบการเชื่อมต่อฐานข้อมูลจะเป็นลักษณะ three tier : JDBC driver (ปรกติ 200 KB หรือน้อยกว่านั้น) ทำการ execute บนไคลเอนท์และถูกใช้เครื่องมือทาง logic ส่งคำสั่ง SQL ไปยัง Network และไปยัง JDBC เซอร์เวอร์,เมื่อได้รับข้อมูลกลับมาจาก เซอร์เวอร์ และถูกจัดการ connection . Driver ประเภทนี้อ่อนุญาตสำหรับการใช้ Just-in-time บนเครื่องไคลเอนท์

JDBC เซอร์เวอร์จะจัดการ connection ที่มีมาไปยัง database ซึ่งดีเหมือน exception และ status events resulting จาก SQL execution . Packages ของ data สำหรับการติดต่อข้าม network ไปยัง JDBC ไคลเอนท์

Middleware server component สามารถอธิบายเหมือน native component หรือถูกเขียนด้วย Java. การอธิบาย Native ในการ connect ไปยัง database server โดยใช้ library ของ vendor หรือ ODBC . Symantec's dbAnywhere และ Intersolv's SequeLink ทั้งคู่ถูกจัดให้อยู่ในประเภทนี้, SequeLink ไม่ต้องการ database client libraries ที่จะใช้ในการติดตั้งบน เซอร์เวอร์ แต่จะใช้ Libraries ของตนเอง. เซอร์เวอร์ ต้องมีการติดตั้ง Configuration สำหรับ database ที่จะใช้ติดต่อเช่น ติดตั้งหมายเลขของ Port, คำแปรของ database specific environment สำหรับ instance, DSQuery กับ Sybase, Database specific parameters เช่น logging , translation และ parameter อื่นๆ ที่ซึ่ง เซอร์เวอร์ อาจจะต้องการ ถ้า Middleware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

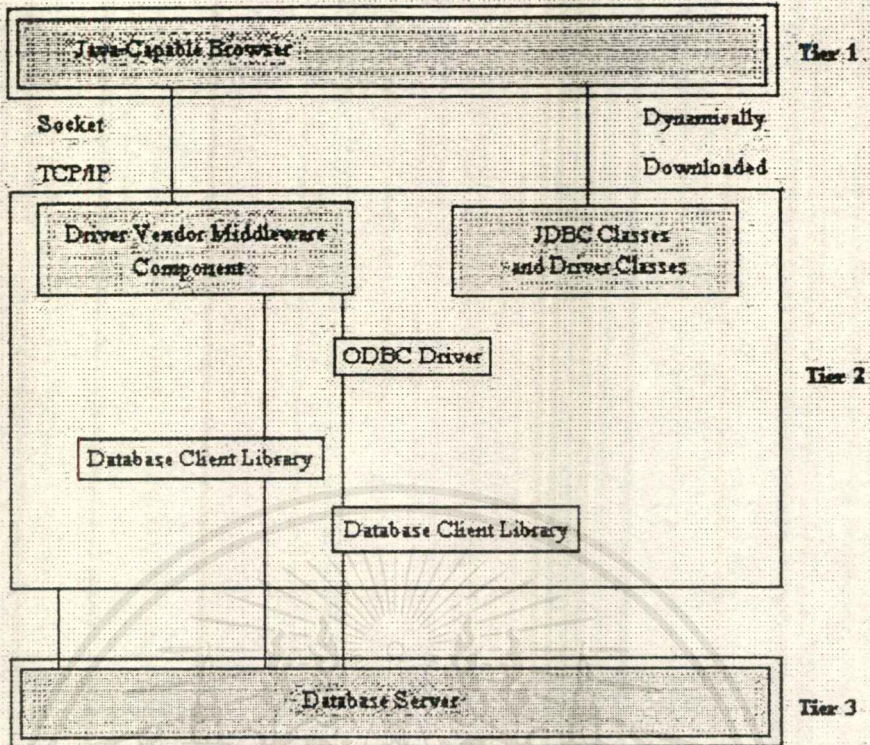
server ถูกเขียนด้วยภาษา Java สามารถใช้ JDBC-compliant driver ในการติดต่อกับ DBMS โดย Database Proprietary Protocol ของ vendor ไม่ต้องการที่จะ configure ที่ เซอร์เวอร์ สำหรับ database



รูปที่ 3-7 การติดต่อ JDBC โดยรูปแบบที่ 3 Network-protocol, all-Java driver ยกตัวอย่างของ dbAnywhere, SequelLink

สำหรับคุณสมบัติเหมือนการติดต่อกับ Caching, จำนวนของ listener threads , และ อื่นๆ , ความต้องการค่าคงที่ที่มีการติดตั้ง. Weblogic's T3 server ถูกจัดให้อยู่ในประเภทนี้. JDBC T3 driver จะถูก executing บน ไคลเอนท์ โดยจะนำชื่อของ driver ใช้ในการติดต่อสื่อสารกับ DBMS ซึ่งดีเหมือนกับคุณสมบัติ ที่ต้องการ โดย driver เช่น username, password, database name, และ อื่นๆ ดังนั้นการส่งข้อมูลไปยัง T3 server จะ Load ผ่าน ไปยัง driver class และ ใช้ในการติดต่อกับ database . Server ยังต้องมีการ load ด้วย database vendor libraries และ/หรือ ODBC driver ในกรณีนี้การส่งค่า driver class name ในประเภทที่ 1 หรือประเภทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-8 การติดต่อ JDBC โดยรูปแบบที่ 3 Network-protocol, all-Java driver
ยกตัวอย่างของ WebLogic

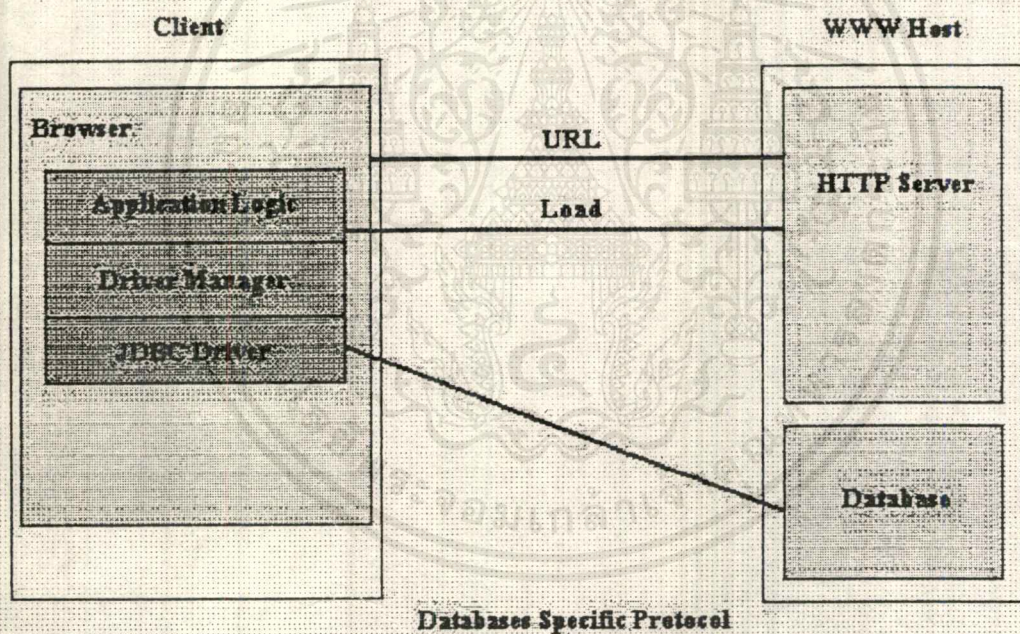
Driver ประเภทนี้ถือว่าเหมาะสมมากที่สุดสำหรับ Internet/intranet-based, multi-user data-intensive application, ที่ซึ่งมีจำนวนของ concurrent data operations มากๆ เพราะมีความยืดหยุ่นคล่องตัวที่สุด เพราะว่าเขียนขึ้นจากภาษาจาวาทั้งหมด ทำให้ไม่จำเป็นต้องมีไคร์ฟแวร์ร่วมที่เขียนขึ้นจากภาษาอื่น ซึ่งต้องติดตั้งเฉพาะบน โคล์เอ็นท์นั้น และสามารถรันบนระบบใดก็ได้ที่สนับสนุนสภาพแวดล้อมเสมือนของจาวาหรือ JVM (Java Virtual Machine) การปฏิบัติงานทุกรูปแบบทั้งคาน์โหลดข้ามเครือข่ายหรือรันบนเครื่องเดี่ยวๆ สามารถกระทำได้หมดเช่นการ queries searches และอื่นๆ และยังมีความสามารถทางด้าน scalability และประสิทธิภาพซึ่งถือเป็นส่วนประกอบหลัก เซอร์เวอร์ สามารถจัดการ multiplexing management ระหว่าง database ที่หลากหลาย, สามารถจัดหา logging และ administration facilities สามารถ load balancing features, และยังสามารถรองรับ catalog และการ Query caches. Three-tier web database application ส่วนใหญ่มี security, firewalls, และ proxies

ข้อดีของ network-centric drivers คือองค์ประกอบของ เซอร์เวอร์ คือ proprietary middleware; ตัวแทนจำหน่ายแต่ละรายใช้ middleware ของตนเองสำหรับการติดต่อสื่อสารข้ามผ่าน network ดังนั้น enterprise users จึงมีความสามารถที่จะจัดการ ได้โดย CORBA products และการ upgrade paths ในอนาคต

Openjdbc จาก I-Kinetics คือ Network-centric driver ที่ใช้ CORBA คือ middleware จะเชื่อม JDBC ไคลเอนท์ และ เซอร์เวอร์. เครื่องมือชนิดนี้มีข้อดีของ CORBA's distributed computing infrastructure คือวงการอุตสาหกรรมยอมรับเป็นมาตรฐานในการ Open ที่จะติดต่อไปยังการสื่อสารอื่นๆ Driver มาเป็น assembly ของ cooperating objects ที่ค่อนข้างจะมากกว่าหนึ่ง เซอร์เวอร์.

3.4.4 Native-protocol, all-Java driver

Native-protocol, all-Java drivers จะแปลงการเรียกใช้คำสั่งของ JDBC ให้อยู่ในรูปแบบของเน็ตเวิร์คโปรโตคอลเฉพาะของฐานข้อมูลนั้นโดยตรงไป โดยคุณสมบัติของตัวแทนจำหน่าย database. Driver เหล่านี้สามารถเขียนในภาษา Java และสามารถติดต่อกับ applet แบบ just-in-time เพราะว่า drivers เหล่านี้จะแปลง JDBC ตรงไปยัง native protocol โดยปราศจากการใช้ ODBC หรือ Native APIs ซึ่งสามารถจัดหามาสำหรับการติดต่อ database ที่มีประสิทธิภาพสูง. Driver เหล่านี้สามารถสร้างจากตัวแทนจำหน่าย DBMS เท่านั้นเนื่องจากความจริงที่ซึ่งความรู้ของ Protocol lies กับตัวแทนจำหน่าย. การนำการติดต่อประเภทนี้มาใช้งานมีอยู่เล็กน้อยมากในปัจจุบัน แต่ในอนาคตคาดว่าจะเพิ่มจำนวนขึ้นเรื่อยๆ



รูปที่ 3-9 การติดต่อ JDBC โดยรูปแบบที่ 4 Native-protocol, all-Java driver

ข้อดีของไคร์ฟเวอร์ชนิดนี้คือไม่ต้องมีการปรับเปลี่ยนระบบฐานข้อมูลเดิมที่ใช้งานอยู่แล้วในแต่ละองค์กร และไม่มีคามจำเป็นที่ต้องติดตั้งไคร์ฟเวอร์ตัวกลาง เช่น ODBC บน ไคลเอนท์ที่แต่ละเครื่อง จึงเหมาะสำหรับการพัฒนาการเชื่อมข้อมูลแบบ Two-Tier สำหรับเครือข่ายอินทราเน็ตภายในองค์กร ไคร์ฟเวอร์ชนิดนี้มีข้อเสียคือ ไม่เหมาะที่จะนำไปใช้บนเครือข่ายอินเทอร์เน็ต เพราะปัญหาเรื่องความ

ปลอดภัย การเชื่อมต่อฐานข้อมูลไม่สามารถกระทำข้ามระบบที่แตกต่างกันได้ ต้องใช้ไคร์ฟเวอร์เฉพาะสำหรับฐานข้อมูลตัวนั้นๆ

JDBC ไคร์ฟเวอร์ชนิดที่ 3 และ 4 คือไคร์ฟเวอร์ที่คาดว่าจะจะเป็นสิ่งที่ต้องการสำหรับการเชื่อมต่อกับฐานข้อมูลในอนาคตเพราะถูกเขียนขึ้นจากภาษาจาวาทั้งหมดซึ่งมีความปลอดภัยและความคล่องตัวมากกว่า เช่นมีการติดตั้งไคร์ฟเวอร์โดยอัตโนมัติเมื่อมีการดาวน์โหลดจาวาแอฟเฟร์ท สำหรับไคร์ฟเวอร์ชนิดที่ 1 และ 2 คือไคร์ฟเวอร์ เพื่อพัฒนาขึ้นมาเพื่อใช้งานร่วมกับไคร์ฟเวอร์คั้งเดิม (ODBC) หรือเน็ตเวิร์คโปรโตคอลเฉพาะฐานข้อมูล ที่มีใช้อยู่แล้วในปัจจุบันไปพลางๆ ก่อน ซึ่งแน่นอนว่าไคร์ฟเวอร์ชนิดที่ 1 และ 2 ชนิดนี้มีข้อจำกัดในการใช้งานมากกว่า 2 ชนิดแรกโดยเฉพาะเรื่องความปลอดภัยและ Bridge ร่วมกับ ODBC ไคร์ฟเวอร์บน Window 95/NT ผลปรากฏว่าไม่ยุ่งยากและให้ประสิทธิภาพดีพอสมควร เหมาะสมในการเป็นเครื่องมือในการทดลองและศึกษาขั้นตอนในการเชื่อมต่อกับฐานข้อมูลจนถึงขั้นพื้นฐานจนถึงขั้นตอนใช้ปฏิบัติงานได้จริง

3.5 ขั้นตอนพื้นฐานสำหรับการเชื่อมต่อกับฐานข้อมูล

1. เปิดการเชื่อมต่อกับฐานข้อมูล
2. ส่งคำสั่ง SQL ไปให้แก่ฐานข้อมูล
3. จัดการกับผลลัพธ์ที่ได้รับ
4. ปิดการเชื่อมต่อ

และเพื่อให้เข้าใจการทำงานของ JDBC ยิ่งขึ้นตัวอย่างโปรแกรมบางส่วนต่อไปนี้จะแสดงให้เห็นถึงขั้นตอนทั้ง 4 ในการติดต่อกับฐานข้อมูลผ่าน ODBC ไคร์ฟเวอร์โดยใช้ JDBC-ODBC Bridge (ฟังก์ชัน API ต่างๆ ของ JDBC ที่มาพร้อมกับ JDK เวอร์ชัน 1.1 ขึ้นไปถูกเก็บรวบรวมไว้ในแพ็คเกจ `java.sql.*`)

```

// (1) เปิดการเชื่อมต่อกับฐานข้อมูล
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:myDB",login,password);
// (2) ส่งคำสั่ง SQL ไปให้แกฐานข้อมูล.
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("SELECT a,b,c FROM Table1");
// (3) จัดการกับผลลัพธ์ที่ได้
while(rs.next()){
int I=rs.getInt("a");
String s=rs.getString("b");
Float f=rs.getFloat("c");
}
// (4) ปิดการเชื่อมต่อ
stmt.close();
con.close();

```

จากตัวอย่างสิ่งแรกที่ทำคือเลือกชนิดของ JDBC ไดรฟ์เวอร์ (JDBC-ODBC Bridge) ระบุชื่อฐานข้อมูล (DSN : Data Source Name) รวมถึงชื่อผู้ใช้ (Login) และรหัสผ่าน (Password) จากนั้นเป็นการสร้างอ็อบเจกต์ stmt (คลาส Statement) สำหรับการเรียกค้นข้อมูล และจัดส่งคำสั่ง SQL ไปในรูปแบบสตริง (String) ผ่านเม็ทฮอด executeQuery() ผลลัพธ์ที่ได้จากเม็ทฮอดนี้คือผลลัพธ์จากการประมวลผลของฐานข้อมูลซึ่งจะเก็บอยู่ในอ็อบเจกต์ rs (คลาส ResultSet) จากนั้นข้อมูลที่ได้รับนี้จะถูกถ่ายทอด (Mapping) ให้อยู่ในชนิดข้อมูลที่จำารูจัก เช่น int, String, float, ฯลฯ เพื่อนำไปใช้ประโยชน์อื่นต่อไป สุดท้ายคือการปิดการเชื่อมต่อกับฐานข้อมูล

ภาษาจาวาและ JDBC คือเครื่องมือที่จะทำให้การเชื่อมต่อสื่อสารกับระบบฐานข้อมูลเป็นไปอย่างสะดวกรวดเร็วและเป็นหนึ่งเดียว (Database-neutral Communication) อย่างไรก็ตามพื้นฐานที่จะขาดเสียไม่ได้สำหรับการสร้างระบบเชื่อมต่อฐานข้อมูลนั้นก็คือความรู้ความเข้าใจเกี่ยวกับหลักการทำงานหลักการออกแบบของฐานข้อมูลชนิดสัมพันธ์ (Relational Database) ระบบจัดการฐานข้อมูล (DBMS : Database Management Systems) และภาษา SQL

จากชนิดของ JDBC Drivers สามารถแสดงตัวแทนจำหน่ายและฐานข้อมูลที่สามารถปฏิบัติการ
แสดงดังตาราง ได้ดังนี้

Vendor	Type	DBMSs supported
Agave Software Design	3	Oracle, Sybase, Informix, others via ODBC
Asgard Software	3	Unisys A series DMSII database
Borland	4	InterBase 4.0
Caribou Lake Software	3	Ingres
Connect Software	4	Sybase, MS SQL Server
DataRamp	3	Several dozen through ODBC drivers.*
Ensodex, Inc.	3	Several through ODBC.*
IBM	2,3	IBM DB2 Version 2
IBM	4	DB2 for OS/400
GWE Technologies	4	mysql
IDS Software	3	Oracle, Sybase, MS SQL Server, MS Access, Informix, Watcom, and others via ODBC
I-Kinetics, Inc.	3	Oracle, Informix, Sybase, and others via ODBC
Imaginary	4	mSQL
InterSoft	3	Essentia
Intersolv	3	DB2, Ingres, Informix, Oracle, Microsoft SQL Server, Sybase 10/11
JavaSoft	1	Several dozen through ODBC drivers.*
KonaSoft, Inc.	3,4	Sybase, Oracle, Informix
NetAway	3	Oracle, Informix, Sybase, MS SQL Server, DB2, others via ODBC
OpenLink	3	Oracle, Informix, Sybase, MS SQL Server, CA-Ingres, Progress, Unify, PostgreSQL, Solid, and others via ODBC.
Oracle Corporation	2	Oracle
SAS Institute Inc.	3,4	SAS, and via SAS/ACCESS, Oracle, Informix, Ingres, and ADABAS
SCO	3	Informix, Oracle, Ingres, Sybase, Interbase
StormCloud Development	3	Any DBMS that is accessible via ODBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sybase, Inc	3,4	Sybase SQL Server, SQL Anywhere, Sybase IQ, Replication Server and more than 25 enterprise and legacy data servers via Sybase OmniCONNECT
Symantec	3	Oracle, Sybase, MS SQL Server, MS Access, Watcom and others via ODBC
Trifox, Inc.	3	ADABAS, DB2, Informix, Ingres, Oracle, Rdb, SQL Server, Sybase, and legacy systems via GENESIS.
Visigenic	3	Several dozen through ODBC drivers.*
WebLogic	2	Oracle, Sybase, MS SQL Server
WebLogic	3	Several dozen through ODBC drivers.*
XDB Systems, Inc.	1,3	Many databases through ODBC.*
Yard Software GmbH	4	YARD-SQL Database

ตารางที่ 3 - 1 เปรียบเทียบ JDBC Driver

* รายการของ ODBC drivers สำหรับการเชื่อมต่อกับ JDBC-ODBC Bridge หรือการเชื่อมต่อกับชนิดที่ 3 drivers สำหรับตารางแสดงให้เห็นถึงผู้แทนจำหน่าย DBMS หรือ ODBC vendors ยกตัวอย่างเช่น Intersolv, Visigenic, SCO, และ DataRamp. JDBC drivers ต่างๆ สามารถตรวจสอบได้ที่ <http://splash.javasoft.com/jdbc/jdbc.vendors.html>

บทที่ 4

การวางแผนและการสร้าง

4.1 ขอบเขตการดำเนินงาน

โครงการการใช้ภาษาจาวาในการติดต่อกับฐานข้อมูลนี้ คณะผู้จัดทำโครงการได้ใช้ภาษาจาวาในการติดต่อฐานข้อมูลโดยใช้ JDBC ซึ่งเป็น API (Application Programming Interface) หนึ่งของภาษาจาวาเอง โดยทำการติดต่อฐานข้อมูลโดยผ่านทาง ODBC ซึ่งได้ออกแบบระบบงานเพื่อใช้ในการแสดงผลในการประยุกต์ใช้ในการติดต่อฐานข้อมูล โดยระบบงานดังกล่าวที่คณะผู้จัดทำได้เลือกนำมาใช้พัฒนาระบบงานคือ

- ระบบแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต
- ระบบการลงทะเบียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต

ทั้งนี้เพื่อต้องการให้สามารถนำเอาทฤษฎีในการใช้ภาษาจาวาในการติดต่อกับฐานข้อมูลมาประยุกต์ใช้กับการออกแบบระบบงานได้จริง เขียนเป็นโปรแกรมใช้งานได้ตามที่ได้ออกแบบไว้จริง โดยในด้านรูปแบบของภาษาจาวาเองนั้นสามารถเชื่อมต่อกับ โปรแกรมฐานข้อมูล (Database Management System : DBMS) ได้มากมาย โดยยกตัวอย่าง โปรแกรมฐานข้อมูล (DBMS) เพื่อเชื่อมต่อกับฐานข้อมูลเพื่อประยุกต์ใช้ระบบงานดังกล่าวดังนี้คือ

- Microsoft Access
- ORACLE
- SyBase
- Microsoft SQL Server

4.2 แผนการดำเนินงาน

4.2.1 ศึกษาภาษาจาวา

การดำเนินงานในขั้นตอนนี้เป็นขั้นต้นเริ่มต้นในการดำเนินโครงการ โดยคณะผู้จัดทำเริ่มจากการศึกษาภาษาจาวา ลักษณะการเขียนโปรแกรม และศึกษารูปแบบของภาษาจาวาที่ควรรู้จักเพื่อช่วยในการพัฒนาโปรแกรม ซึ่งคณะผู้จัดทำได้จัดทำทฤษฎีเพื่อประกอบเป็นแนวทางในการศึกษานั้นตามเนื้อหาในบทที่ 2 ดังกล่าวในข้างต้นแล้ว

4.2.2 ศึกษาเครื่องมือที่ช่วยในการพัฒนาโปรแกรมภาษาจาวา

จากการศึกษาโปรแกรมภาษาจาวาแล้วเนื่องจากคุณสมบัติของภาษาจาวาเองนั้นมีลักษณะที่เรียกว่า Command Line Driven ซึ่งยากในการพัฒนาโปรแกรมประยุกต์ใช้ภาษาจาวาในลักษณะของโครงการที่มีขนาดใหญ่และสลับ ซับซ้อน

คณะผู้จัดทำได้ศึกษาเครื่องมือ (Tools) ที่ใช้พัฒนาภาษาจาวาโดยสรุปพอสังเขปในส่วนท้ายของบทที่ 2 ดังกล่าวในข้างต้น ซึ่งคณะผู้จัดทำได้ทดลองใช้แล้ว และได้สรุปผลการทดลองแล้วว่าโปรแกรม Symantec Visual Café Pro นั้นเหมาะสมกับโครงงานนี้มากที่สุด แล้วได้ตัดสินใจเลือกใช้เครื่องมือนี้เพื่อเป็นเครื่องมือที่ช่วยในการพัฒนาโปรแกรมภาษาจาวาในการติดต่อฐานข้อมูล

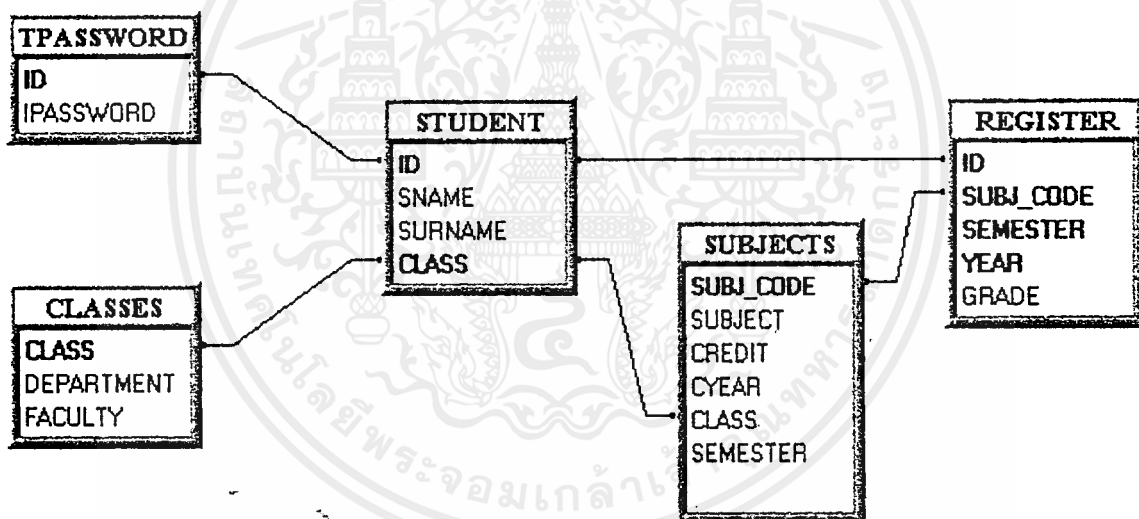
4.2.3 ออกแบบฐานข้อมูล

โครงงานการใช้ภาษาจาวาในการติดต่อฐานข้อมูลนั้นเพื่อให้เห็นการใช้งาน และสามารถทำงานได้จริงนั้นทางคณะผู้จัดทำจึงได้คิดระบบงานเพื่อมาประยุกต์ใช้งานนั้น โดยจะแสดงโปรแกรมประยุกต์ดังนี้คือ

4.2.3.1 ระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

4.2.3.2 ระบบงานลงทะเบียนเรียนผ่านเครือข่ายอินเทอร์เน็ต

และเพื่อการออกแบบระบบทั้งสองที่ได้ออกแบบมานั้นจึงได้ออกแบบฐานข้อมูลโดยแสดงความสัมพันธ์ระหว่างตารางดังต่อไปนี้

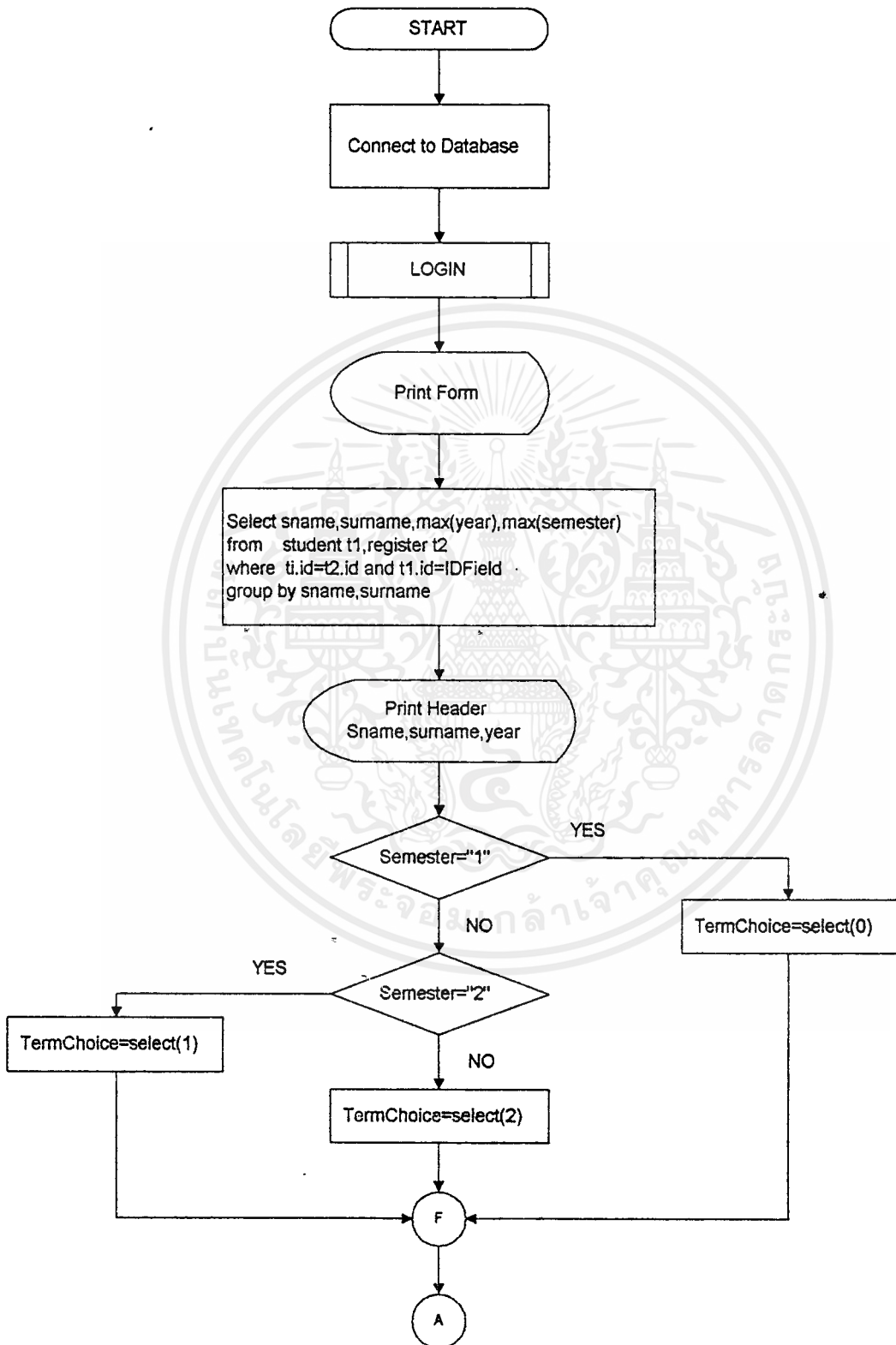


รูปที่ 4-1 รูปแสดงความสัมพันธ์ระหว่างฐานข้อมูล

4.2.4 ศึกษาระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

ทางคณะผู้จัดทำได้ศึกษาระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต โดยมีขั้นตอนการตาม FlowChart ดังต่อไปนี้

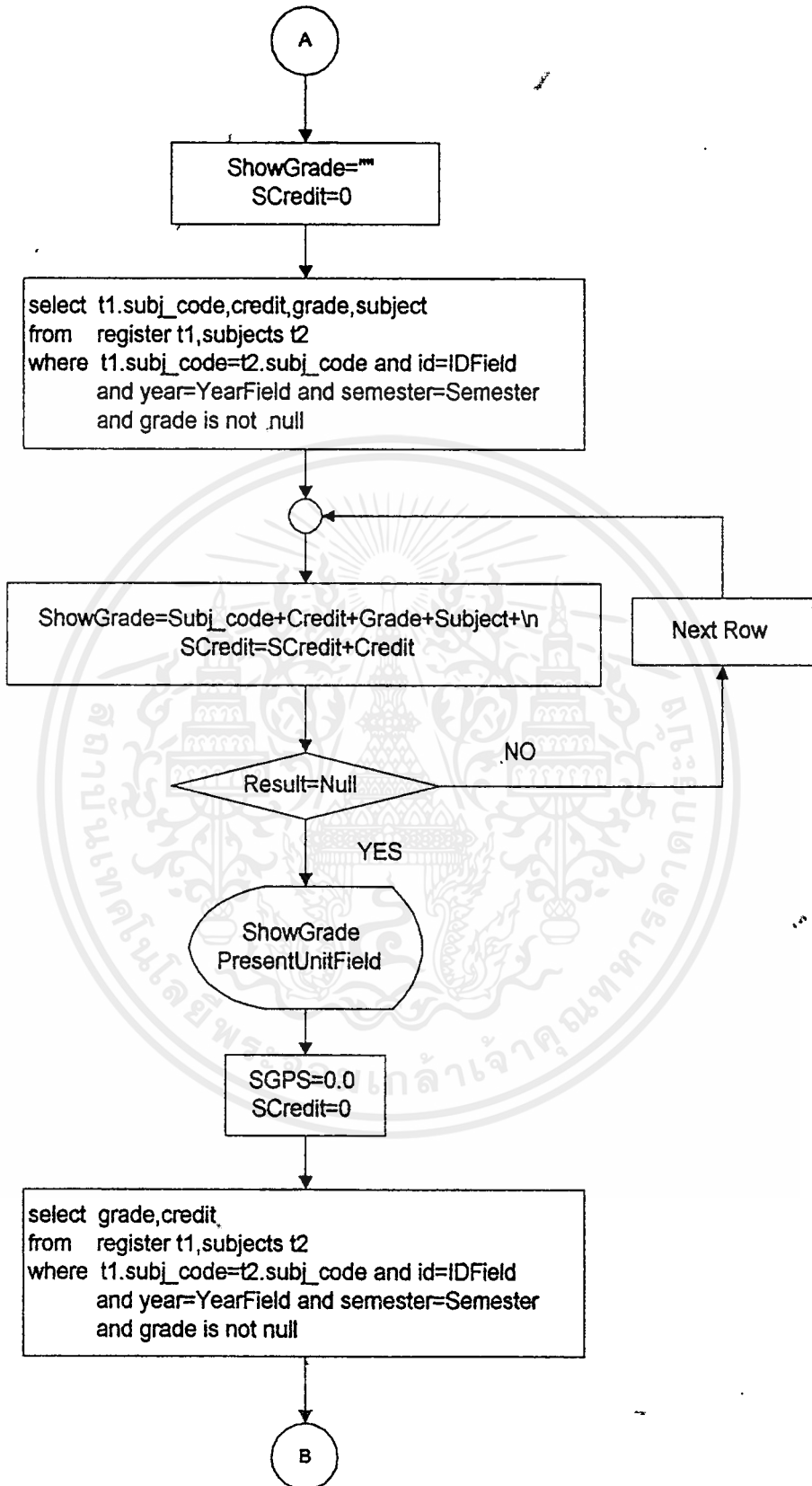
FLOWCHART

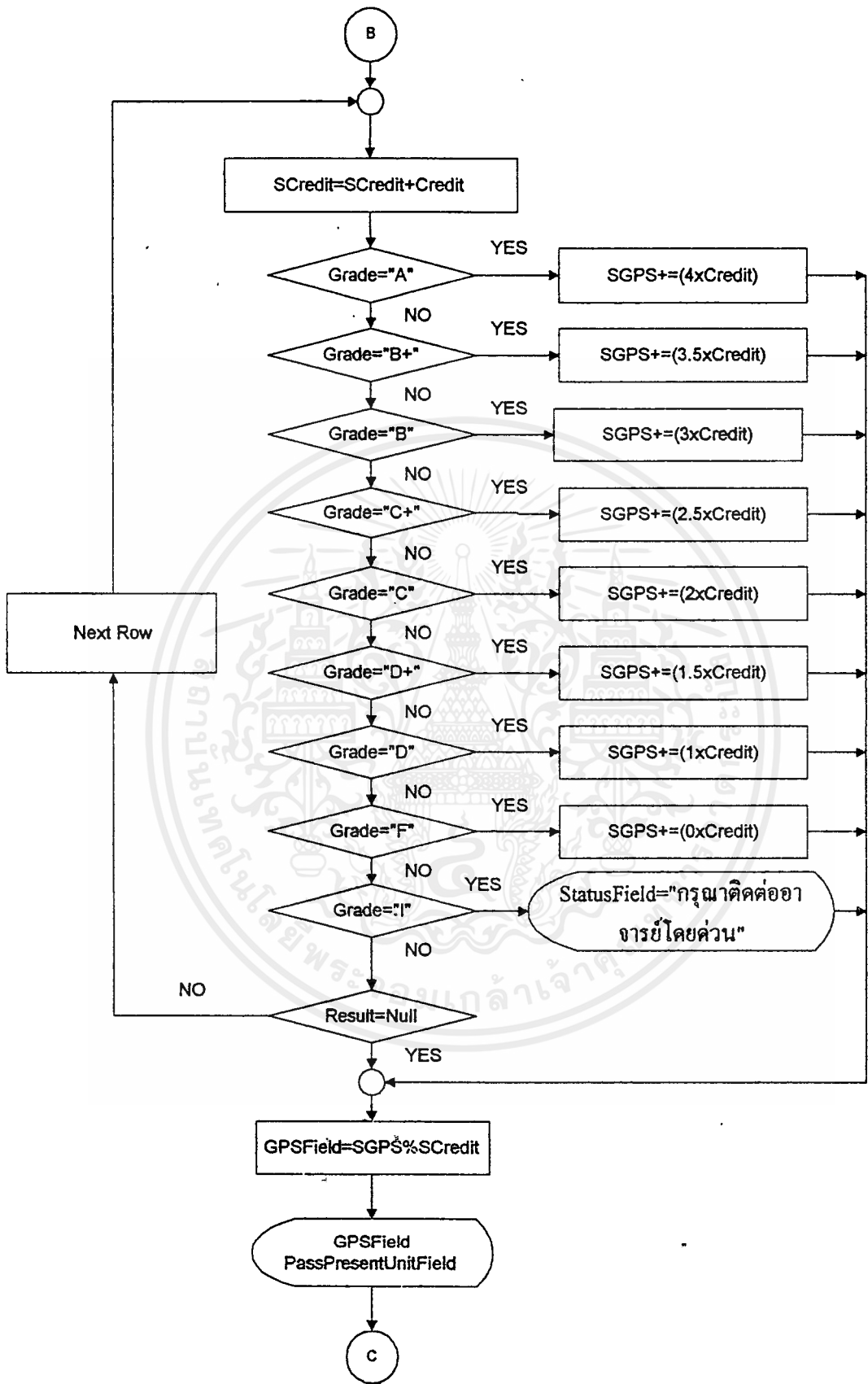


รูปที่ 4 – 2 รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

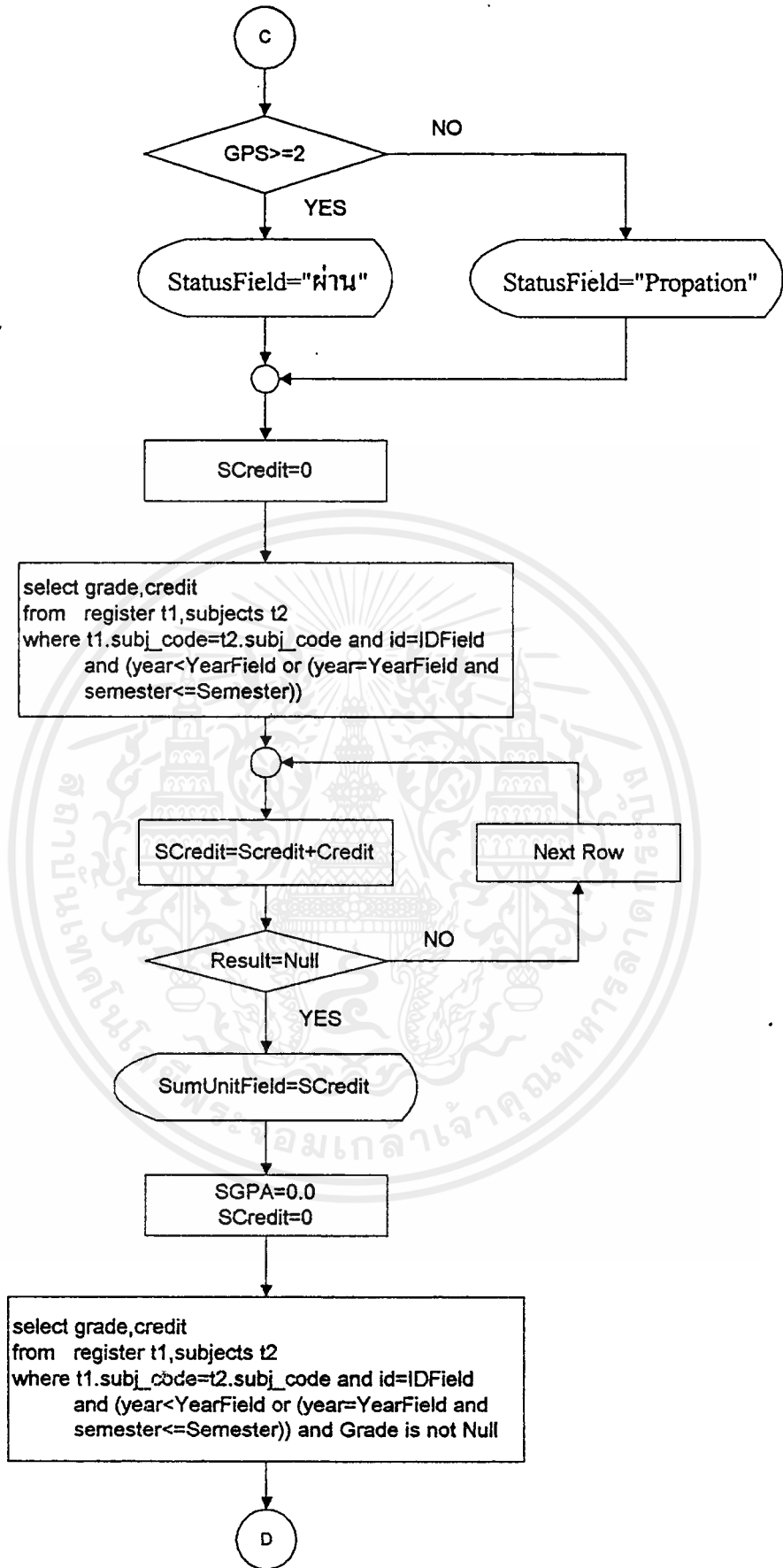
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





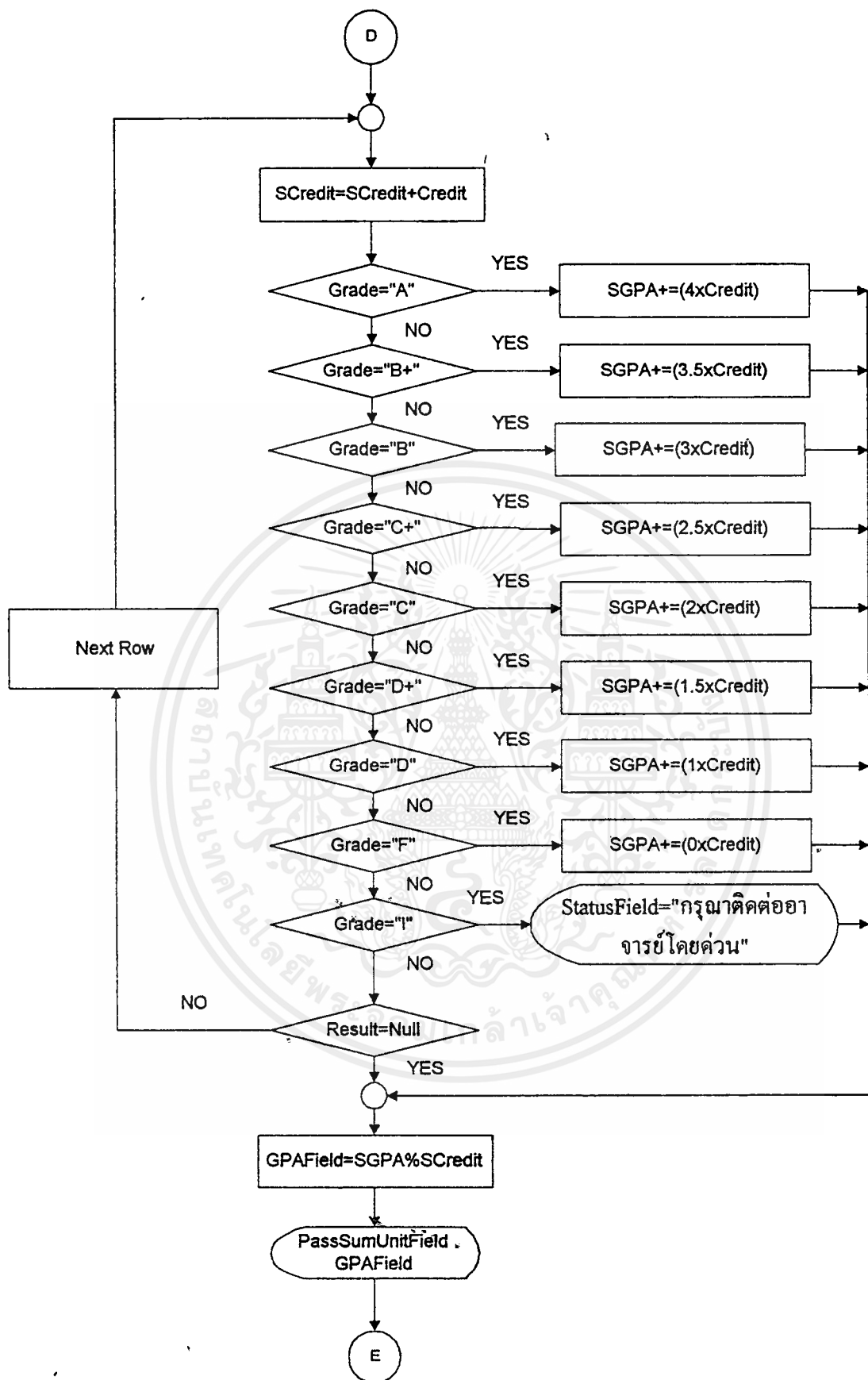
รูปที่ 4 - 2 (ต่อ) รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



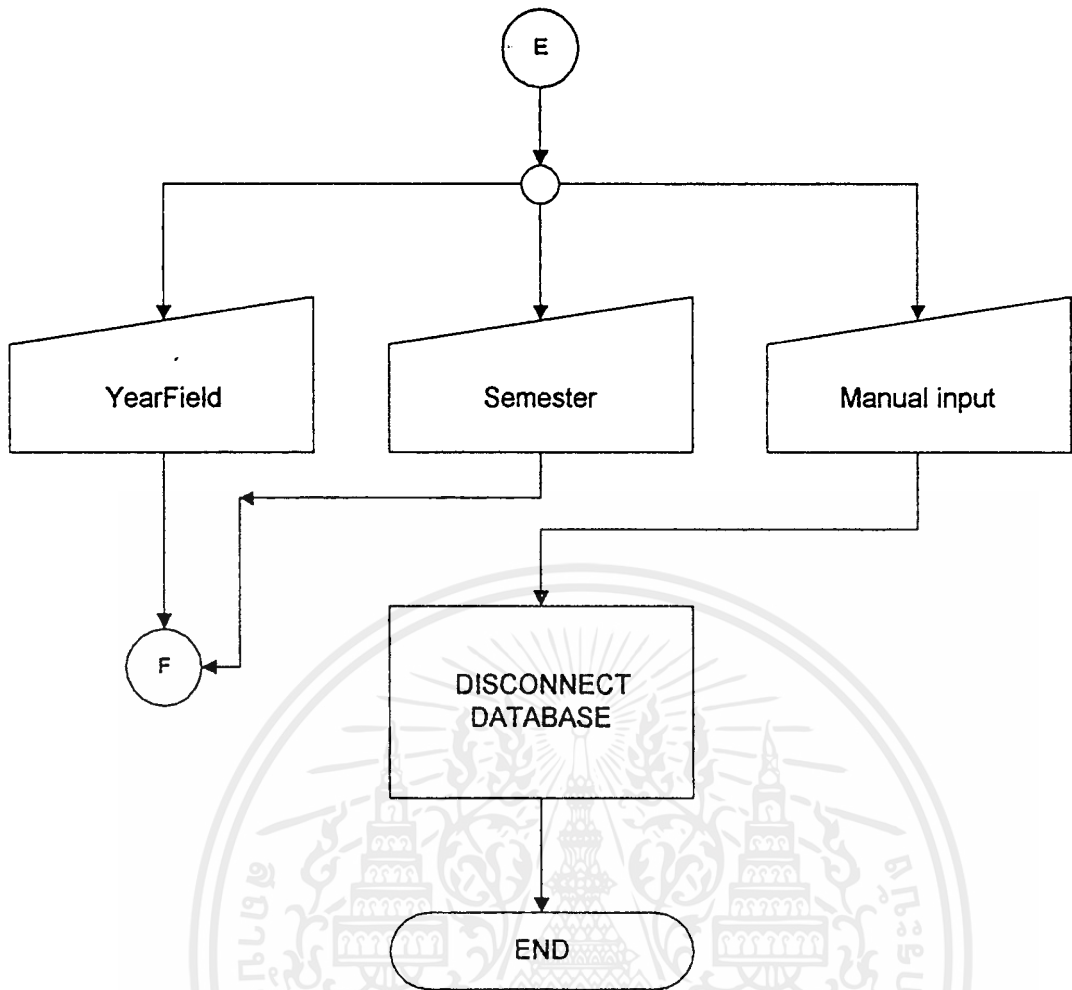
รูปที่ 4-2 (ต่อ) รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-2 (ต่อ) รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



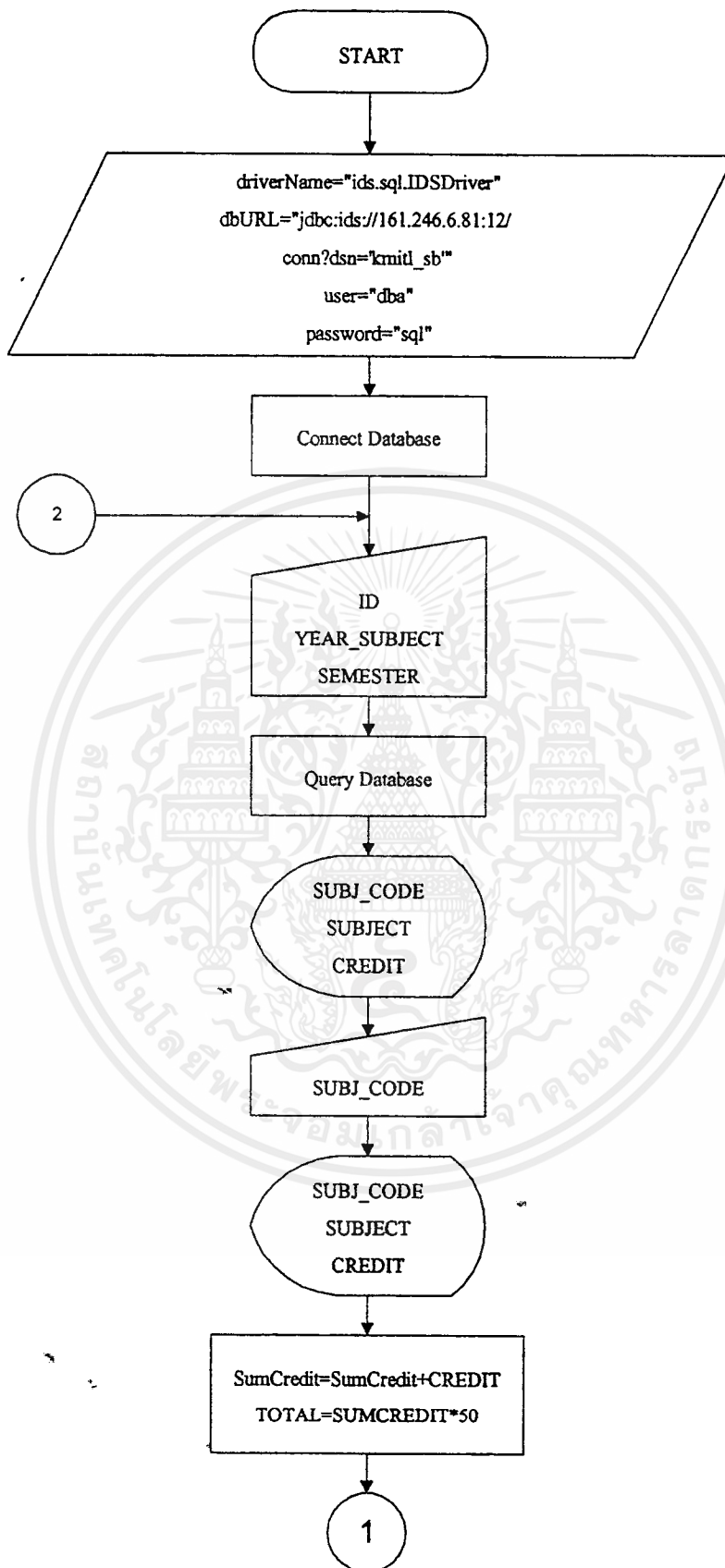
รูปที่ 4-2 (ต่อ) รูปแสดง FLOWCHART ระบบแสดงผลการศึกษาผ่านอินเทอร์เน็ต

4.2.5 พัฒนาและปรับปรุงระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

4.2.6 ศึกษากระบวนการลงทะเบียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต

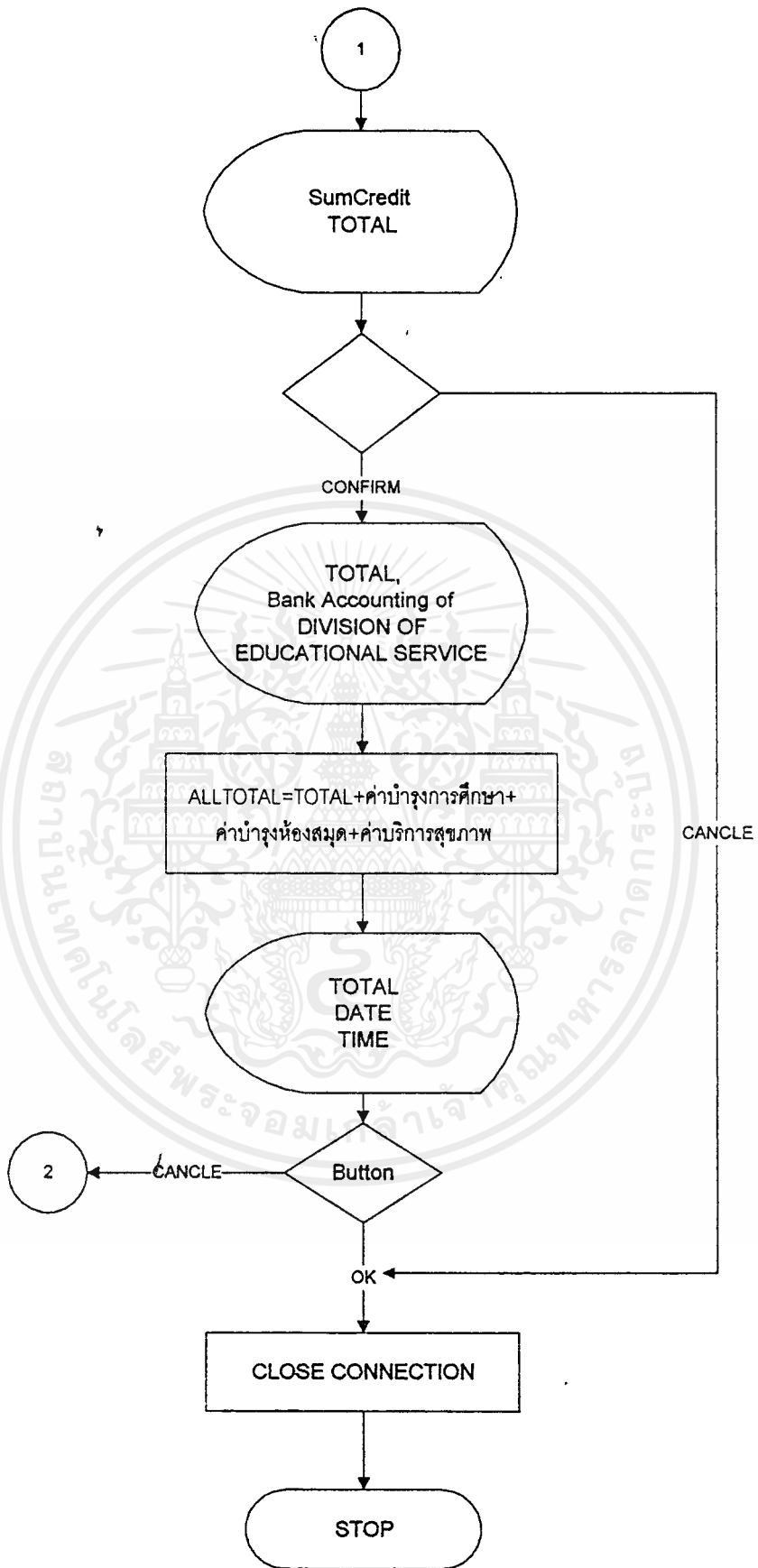
ทางคณะผู้จัดทำได้ศึกษากระบวนการลงทะเบียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต โดยมีขั้นตอนการตาม FlowChart ดังต่อไปนี้

FLOWCHART



รูปที่ 4-3 รูปแสดง FLOWCHART ระบบลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

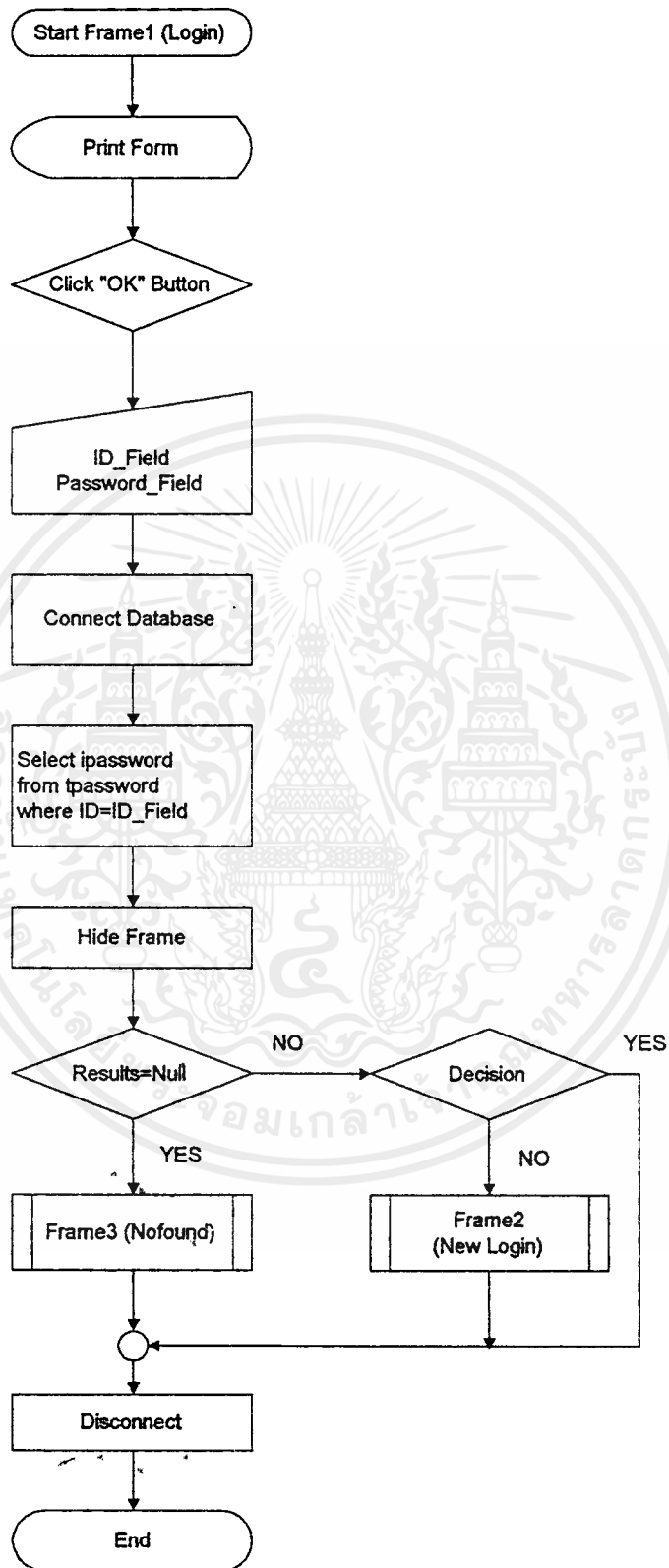


รูปที่ 4 - 3 (ต่อ) รูปแสดง FLOWCHART ระบบลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

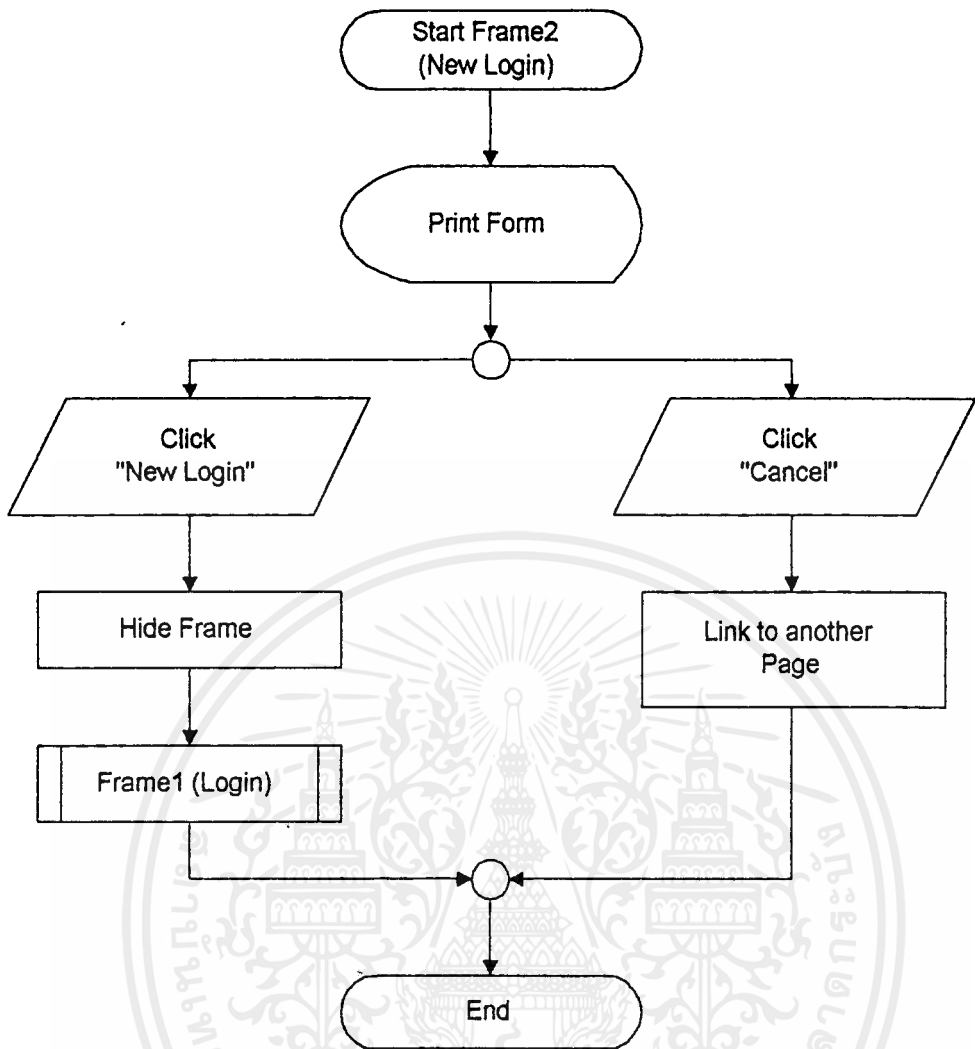
4.2.7 พัฒนาและปรับปรุงระบบการลงทะเบียนนิสิตผ่านเครือข่ายอินเทอร์เน็ต

4.2.8 ระบบรักษาความปลอดภัย

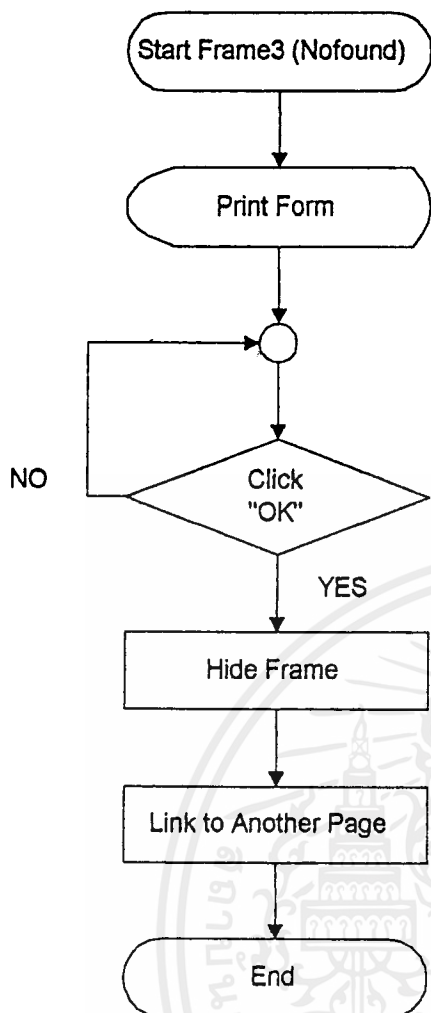


รูป 4 - 4 รูปแสดง FLOWCHART ระบบรักษาความปลอดภัยของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4 - 4 (ต่อ) รูปแสดง FLOWCHART ระบบรักษาความปลอดภัยของโปรแกรม



รูป 4 - 4 (ต่อ) รูปแสดง FLOWCHART ระบบรักษาความปลอดภัยของโปรแกรม

4.2.9 การติดตั้งระบบงานบนเครือข่ายอินเทอร์เน็ต

- 4.2.9.1 ติดตั้ง Microsoft Window NT Server
- 4.2.9.2 ติดตั้งโปรแกรมฐานข้อมูล (DBMS) บน Server
- 4.2.9.3 สร้างฐานข้อมูลบน DBMS
- 4.2.9.4 ติดตั้ง ODBC กับฐานข้อมูลบน DBMS
- 4.2.9.5 ติดตั้ง Homepage เพื่อใช้แสดงโปรแกรมจาวาเอ็ปเพล็ทบนเครือข่ายอินเทอร์เน็ต

ทั้งหมดคือขั้นตอนโดยสรุปที่ทางคณะผู้จัดทำได้นำเสนอเป็นแนวทาง ขั้นตอนในการที่ใช้ปฏิบัติจริง
งานนี้

บทที่ 5

ผลการดำเนินงาน

การดำเนินงานของโครงการนี้จากแนวทางการปฏิบัติดังกล่าวมาแล้วในบทที่ 4 ดังที่ได้เสนอระบบงานเพื่อประยุกต์ใช้ภาษาจาวาในการติดต่อฐานข้อมูลกับฐานข้อมูลคือ

- ระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต
- ระบบงานลงทะเบียนการเรียนผ่านเครือข่ายอินเทอร์เน็ต

จากการที่ได้ดำเนินระบบงานดังกล่าว จึงขอนำเสนอและวิธีการใช้งานระบบงานเรียงลำดับดังนี้คือ

5.1 ระบบแสดงผลการเรียนผ่านเครือข่ายอินเทอร์เน็ต

The screenshot shows a Java Applet Viewer window titled "Applet Viewer: Student class". The applet content is a web form titled "ระบบแสดงผลการเรียนผ่านเครือข่ายอินเทอร์เน็ต". The form contains the following fields and labels:

- เลขทะเบียนนักศึกษา: [text input]
- ปี: [text input]
- เทอม: [dropdown menu, value: 1]
- ชื่อ: [text input]
- นามสกุล: [text input]
- รหัสวิชา: [text input]
- หน่วยกิต: [text input]
- ผลการศึกษา: [text input]
- ชื่อวิชา: [text input]
- หน่วยกิต: [text input]
- หน่วยกิตที่ผ่าน: [text input]
- เกรด: [text input]
- สถานะ: [text input]
- คะแนนปัจจุบัน: [text input]
- คะแนน: [text input]
- GPA: [text input]

รูปที่ 5 - 1 รูปแสดงระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.1 วิธีใช้ระบบงานแสดงผลการเรียนผ่านเครือข่ายอินเทอร์เน็ต

จากรูปที่ 5 - 1 แสดงให้เห็นหน้าจอที่ผู้ใช้เรียกใช้ในการแสดงผลการเรียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต หลังจากผ่านการป้อนรหัสนักศึกษาแล้วโปรแกรมจะแสดงผลการเรียนที่นักศึกษาได้ลงทะเบียนล่าสุด หรือในเทอมปัจจุบัน ซึ่งผลคะแนนอาจจะยังไม่ครบก็ได้

การแสดงผลการเรียนนั้นถ้านักศึกษาต้องการที่จะเรียกดูผลการเรียนในเทอมการศึกษาที่ผ่านมา โดยระบบงานแสดงผลการศึกษผ่านเครือข่ายอินเทอร์เน็ตนี้ นักศึกษาสามารถที่จะเปลี่ยนปีการศึกษา และเทอมการศึกษา เพื่อเรียกดูผลการเรียนในปีการศึกษา หรือเทอมที่ผ่านมาได้ โดยการแสดงผลนั้นจะแสดงรหัสวิชา หน่วยกิต ผลคะแนน และชื่อวิชา โดยจะเรียงลำดับตามชื่อวิชา

5.2 ระบบลงทะเบียนผ่านเครือข่ายอินเทอร์เน็ต

Applet Viewer: Account class

ระบบลงทะเบียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เลขทะเบียนนักศึกษา ชื่อ นามสกุล

ห้อง ภาควิชา คณะ

กรุณาดำเนินการลงทะเบียนและรายชื่อวิชาให้ถูกต้อง

ชั้นปีที่ ภาคการศึกษา ปีการศึกษา

รหัสวิชา	รายชื่อวิชา	จำนวนหน่วยกิต
<input type="text"/>	<input type="text"/>	<input type="text"/>

เลือกวิชา

รายชื่อวิชาที่นักศึกษาต้องการลงทะเบียน โปรดตรวจสอบด้วยความระมัดระวัง

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
----------------------	----------------------	----------------------	----------------------

รวม หน่วยกิต รวมเงิน บาท

รูปที่ 5 - 2 รูปแสดงระบบลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1 วิธีใช้ระบบงานลงทะเบียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต

จากรูปที่ 5 - 2 แสดงให้เห็นหน้าจอที่ผู้ใช้เรียกใช้ในการลงทะเบียนเป็นการเรียนนักศึกษาผ่านเครือข่ายอินเทอร์เน็ต หลังจากผ่านการป้อนรหัสนักศึกษาแล้วโปรแกรมจะแสดงรูปแบบของระบบลงทะเบียนการเรียนโดยจะแสดงรหัสนักศึกษา ชื่อ สกุล ห้อง ภาควิชา และ คณะ ซึ่งถือเป็นข้อมูลทั่วไปของนักศึกษา

ขั้นตอนต่อไปจากที่ระบบค้นพบข้อมูลของนักศึกษาแล้ว ในส่วนต่อไปของระบบจะแสดงรายชื่อวิชาที่แผนกทะเบียนอนุญาตให้นักศึกษาลงทะเบียน โดยนักศึกษาสามารถเลือกให้แสดงรายชื่อวิชาในชั้นปีการศึกษา หรือภาคการเรียนที่ต้องการได้ (เพื่อที่จะเลือกในการลงทะเบียนวิชาที่ต้องการลงทะเบียนเมื่อวิชานั้น ไม่ผ่านการสอบได้)

จากรายชื่อวิชาที่นักศึกษาเลือกที่จะลงทะเบียน ในส่วนถัดมาจะมีปุ่มเพื่อเลือกวิชาที่แสดงนั้นเพื่อที่จะเลือกลงทะเบียนที่แท้จริง ซึ่งมีทั้งเลือกทั้งหมด (Add All) เลือกเป็นรายวิชา (Add) ถอนวิชา (Remove)

ในส่วนถัดไปจะแสดงถึงรายชื่อวิชาที่นักศึกษาเลือกตัดสินใจในการลงทะเบียนในภาคการศึกษาปัจจุบันนี้ถ้านักศึกษาได้ตรวจสอบแล้วด้วยความระมัดระวังแล้วก็สามารถคลิกปุ่มยืนยัน (Confirm) หรือ ยกเลิก (Cancel)

จากนั้นระบบจะได้รับข้อมูลในการลงทะเบียนและจะแสดงจำนวนเงินที่ใช้ในการลงทะเบียนทั้งหมดไม่ว่าจะเป็นค่าหน่วยกิต ค่าบำรุงการศึกษา ค่าห้องสมุด ค่าบริการสุขภาพ โดยจะแสดงข้อบัญญัติธนาคาร เพื่อให้ศึกษานั้น โอนเงินเข้าบัญชีธนาคารดังกล่าวพร้อมทั้งบอกให้นักศึกษาทราบว่า การลงทะเบียนในครั้งนี้จะสมบูรณ์ก็ต่อเมื่อทางสถาบันฯ ได้รับเงินจากการโอนเงินของนักศึกษาเรียบร้อยแล้วจึงถือเป็นการลงทะเบียนที่สมบูรณ์

บทที่ 6

บทวิจารณ์และสรุป

6.1 สรุปผลการดำเนินงาน

โครงการภาษาจาวาในการติดต่อฐานข้อมูล เป็นระบบงานแนวใหม่ที่เหมาะสมกับการประยุกต์ใช้ระบบงานกับงานฐานข้อมูลบนเครือข่ายอินเทอร์เน็ต ทั้งนี้เพราะคุณสมบัติของภาษาจาวาที่มีความโดดเด่นอย่างมากในปัจจุบันในการใช้งานร่วมกับ Homepage ในบริการ World Wide Web ซึ่งแสดงผลโดยเว็บเบราว์เซอร์ (Web Browser) นั้นเป็นการนำเสนองานที่ได้รับความนิยมสูงสุดและมีแนวโน้มจะได้รับความนิยมเพิ่มขึ้นเรื่อยๆ ในเครือข่ายอินเทอร์เน็ต เพราะปัจจุบันการใช้ภาษา HTML ในการแสดงผลนั้นไม่สามารถนำเสนองานในบางลักษณะได้ ประกอบกับมีคุณสมบัติที่ไม่เพียงพอต่อการประยุกต์ใช้งานฐานข้อมูล หรือภาษาอื่นเช่น PERL, CGI นั้นก็ยังมีรูปแบบการเขียน โปรแกรมที่ผู้พัฒนายังต้องศึกษาศึกษาใหม่เพราะเป็นรูปแบบที่ไม่เหมือนกับโปรแกรมประยุกต์ทั่วๆ ไปซึ่งนักเขียน โปรแกรมทั่วไปจะคุ้นเคยเช่น Delphi, Visual C, Visual Basic ภาษาจาวาจึงถือว่าเป็นภาษาที่เหมาะสม มีรูปแบบการเขียนโปรแกรมที่คุ้นเคยกับ โปรแกรมทั่วไปบน Windows ที่ได้รับความนิยมหรือรูปแบบที่ศึกษาเพิ่มเติมได้ไม่ยากนัก ประกอบกับมีความยืดหยุ่นมีระบบความปลอดภัยโดยตัวภาษาเอง และความโดดเด่นที่การทำงานนั้นไม่ขึ้นอยู่กับรูปแบบในการนำเสนอ (Platform) ไม่ว่าจะเป็น Micro Computer, Unix หรือ Macintosh ดังได้กล่าวมาแล้วในทฤษฎีข้างต้น คณะผู้จัดทำจึงถือว่าโครงการนี้มีประโยชน์และประยุกต์ใช้งานฐานข้อมูลได้จริงในยุค Network Computing ในระบบเครือข่ายอินเทอร์เน็ตในปัจจุบัน

6.2 แนวทางในการพัฒนาต่อ

แนวทางในการพัฒนาหรือประยุกต์ใช้งาน โครงการการใช้ภาษาจาวาในการติดต่อกับฐานข้อมูลนั้น คณะผู้จัดทำคิดว่าการนำเสนองานในเครือข่ายอินเทอร์เน็ตมีการประยุกต์ใช้งานหรือมีแนวทางการประยุกต์ใช้งานบนเครือข่ายอินเทอร์เน็ตนั้นมีความหลากหลายเพิ่มขึ้นเรื่อยๆ เพราะว่าเครือข่ายอินเทอร์เน็ตนั้นในปัจจุบันมีเครือข่ายอยู่ในทุกมุมโลก โดยเฉพาะอย่างยิ่งการนำประยุกต์ใช้กับงานทางด้านฐานข้อมูลในวงการธุรกิจที่มีการประยุกต์ใช้งานเครือข่ายอินเทอร์เน็ตนั้น ไม่ว่าจะเป็นการนำเสนอสินค้าหรือบริการก็มีความเกี่ยวข้องกับฐานข้อมูลด้วยทั้งสิ้น โครงการนี้จึงถือเป็นแนวทางที่ใช้เริ่มต้นที่ดีในการประยุกต์ใช้งานฐานข้อมูลกับวงการธุรกิจดังกล่าว

สำหรับแนวทางในการพัฒนาต่อของนักศึกษานั้น คณะผู้จัดทำขอเสนอแนวความคิดเพื่อที่จะเป็นแนวทางให้นักศึกษานอกเหนือจากการนำเสนองานในการติดต่อทางธุรกิจกับระบบฐานข้อมูล โดยทั่วไปดังกล่าวข้างต้นแล้วคือ ในส่วนของภาษาจาวาเองมีคุณสมบัติ และมีความโดดเด่นอย่างมากมายซึ่งทางคณะผู้จัดทำเองยังศึกษาได้ไม่ครบถ้วนดีนัก ยกตัวอย่างเช่นการติดต่อกับ Socket โดยตรง, การโอนเงิน ถอนเงิน การติดต่อซื้อขายผ่านเครือข่ายอินเทอร์เน็ต หรือกิจกรรมที่เกี่ยวข้องกับเงินตรา ซึ่งถือเป็นเรื่องที่ยุ่งยากต้องมีความละเอียดรอบคอบ และมีระบบรักษาความปลอดภัยค่อนข้างสูงกับการถูกนักโจรกรรม

เอกละไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางคอมพิวเตอร์ (Hacker) ในการเจาะระบบการเงินของคน เพราะฉะนั้นคณะผู้จัดทำจึงมีความเห็นว่า เรื่องดังกล่าวเป็นสิ่งที่น่าสนใจและมีความท้าทายอยู่ไม่น้อยสำหรับนักศึกษา จึงขอเสนอแนวคิดนี้เป็นแนวทางในการพัฒนาโครงการภาษาจาวาในการติดต่อกับฐานข้อมูลนี้คือ ไปในอนาคต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

IDS Server

IDS Server คือ Server ที่ใช้ในการติดต่อกับ Internet Database ซึ่งทำงานได้ทั้ง HTML และ Java ในการเขียน Web Pages เพื่อติดต่อกับฐานข้อมูลสำหรับโปรแกรมประยุกต์ในอินเทอร์เน็ตและอินทราเน็ต และ IDS Server นั้นสามารถติดต่อกับ ODBC บน Windows NT และ Windows 95. ฐานข้อมูลนั้นสามารถที่จะนำไปใช้ได้รวมถึง Oracle, Sybase, SQL Server และระบบฐานข้อมูลที่โด่งดังอื่นๆ ด้วย

IDS Server มี API ของภาษาที่ใช้ติดต่อกับ JDBC ด้วยคือ IDS JDBC Driver. โปรแกรมประยุกต์ใช้ภาษาจาวาสามารถใช้ JDBC API นี้ในการติดต่อกับฐานข้อมูลโดยผ่าน IDS Server. IDS JDBC Driver นี้มีคุณสมบัติมากมายทั้งยังมีขนาดเล็กมีประสิทธิภาพสูง โดยใช้ Type-3 JDBC driver. เพราะว่า JDBC driver นี้สามารถทำงานได้ทั้ง Java applets and Java programs. และยังสามารถใช้กับบราวเซอร์ทุกประเภทที่สามารถทำงานกับภาษาจาวาได้ (All Java WebBrowser, JDK ทุกรุ่น (Version), Microsoft SDK สำหรับภาษา Java

คุณสมบัติอื่นๆ ที่ถือว่าเป็นจุดเด่นหนึ่งของ IDS Server ที่ถือว่ามีน่าสนใจ พอสรุปได้ดังต่อไปนี้

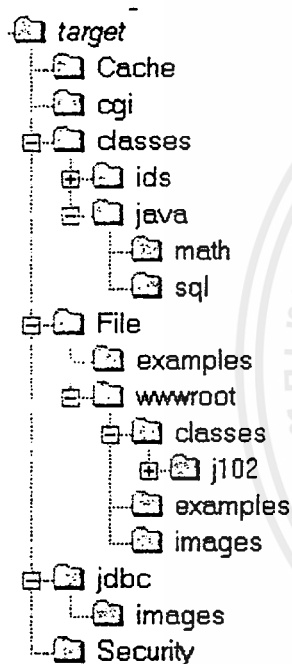
- Define/prepare/execute any SQL statement and stored procedures
- Seamless HTML table integration
- Create master/detail reports
- View results in small groups
- Easy onbreak processing
- Conditional HTML output
- Dynamic SQL query
- Flexible variable handling
- Use JavaScript to process results
- Heterogeneous database access
- Transaction control
- Integrated IDS JDBC Driver
- Public-Key and Secret-Key Secure JDBC
- Complete firewall support for JDBC applications
- JDBC ResultSet Caching
- Supports Zero Installation Applet Deployment

การติดตั้ง IDS Server

IDS Server เป็นไฟล์ที่ขยายได้เองโดยอัตโนมัติ (one self-extracting executable file). โดยเรียกใช้คำสั่ง “IDSSxxxx.EXE” ในการติดตั้ง IDS Server, ดับเบิลคลิกบน ไอคอนและทำงานตามขั้นตอนที่ละขั้นตอนตามหน้าจอ.

1. Directory Hierarchy

ในการติดตั้ง IDS Server จะสร้างลำดับไดเรกทอรี (directory hierarchy) ภายใต้ไดเรกทอรีที่ผู้ติดตั้งเป็นผู้เลือก (Target Directory) ดังต่อไปนี้



- “Cache” directory ใช้สำหรับเก็บไฟล์ประเภท HTX
- “cgi” directory ใช้สำหรับเก็บ IDS CGI Gateway program “idsc.exe” และสิ่งต่างๆ ที่จำเป็นสำหรับการสร้าง custom gateway programs.
- “classes” directory ใช้เก็บ IDS JDBC Driver และ โปรแกรมสำหรับทดสอบ คือ “IDSTest.class”.
- “jdbc” directory ใช้เก็บ JDBC API HTML documentation.
- “File” directory ใช้เก็บ 2 subdirectories. Subdirectory แรกคือ “examples” ใช้เก็บ ไฟล์ HTX ทั้งหมดของ IDS Server Examples. Subdirectory ที่สองคือ “wwwroot” เป็น document root เมื่อบริการ IDS Server services ใดๆ ไปเมื่อเว็บต้องการติดต่อกับเว็บเบราว์เซอร์
- Directory “wwwroot\examples” ใช้เก็บ HTML files ทั้งหมดของ IDS Server Examples.
- “www\classes” directory ใช้เก็บ j102.sql package.

รูปที่ ก – 1 Directory Hierarchy

2. Install Databases

ก่อนการทดสอบโปรแกรม ผู้ใช้ควรจะต้องติดตั้ง ODBC system data source ตามขั้นตอนต่อไปนี้

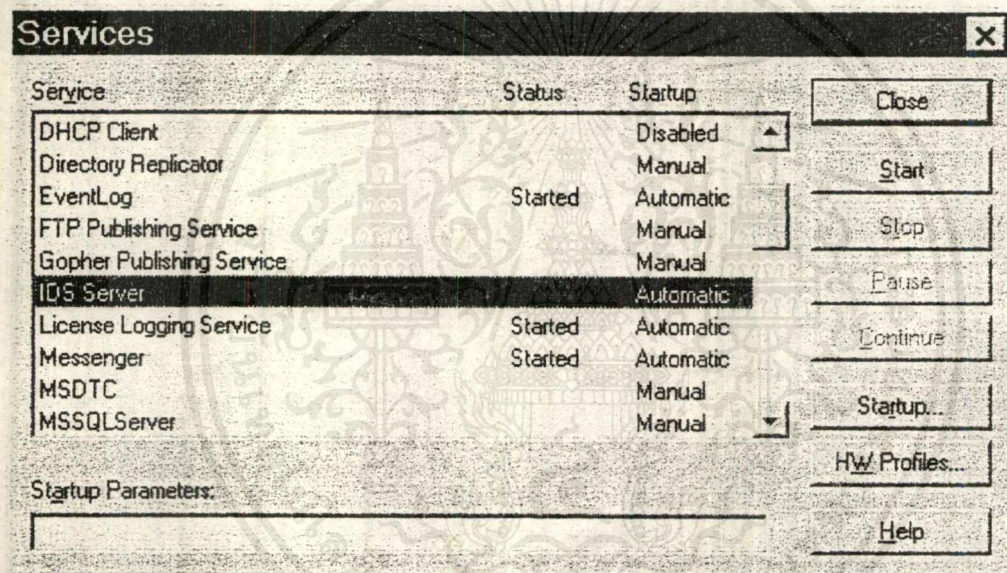
- 1) เปิด Control Panel และคลิกที่ไอคอน 32-bit ODBC icon.
- 2) คลิกบนปุ่ม “System DSN” button at the bottom.
- 3) คลิกบนปุ่ม “Add”
- 4) เลือก Driver ยกตัวอย่างเช่น “Microsoft Access Driver (*.mdb)” คลิก OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5) คลิกบนปุ่ม “Select...” แล้วใช้ dialog เพื่อหาไฟล์ Database
- 6) ที่ฟิลด์ “Data Source Name:” ให้ใส่ชื่อเพื่อใช้อ้างถึง ODBC Driver นี้
- 7) คลิกปุ่ม “OK” และคลิกปุ่ม “Close”

3. Running IDS Server

สำหรับ Windows NT⁴, IDS Server จะทำงานบน NT service. ใน Windows NT 4.0, การติดตั้งนั้นใน “Start” menu – “Programs” folder – “IDS Server” folder, a shortcut called “Shortcut to Services”. ให้เลือก shortcut นี้บริการที่ใช้ในการควบคุม window จะปรากฏขึ้น. ใน Windows NT 3.51, ผู้ใช้ยังต้องค้นหาใน Control Panel ใน Main program group และคลิกที่ไอคอน “Services” icon เพื่อที่จะให้บริการในการควบคุม window นั้นทำงาน



รูปที่ ก - 2 Running IDS Server บน Windows NT

ใน main list box, highlight “IDS Server”. ให้คลิกเมาส์ข้างขวามุมปุ่ม “Start” เพื่อที่จะเริ่มดำเนินการทำงานของ server. ในการติดตั้ง IDS Server “Automatic” ให้คลิกไปดูลูกปุ่ม “Startup...” เพื่อเข้าไปแก้ไข option ต่อไป.

สำหรับ Windows 95⁴, IDS Server จะทำงานด้วย console program. การติดตั้งจะถูกสร้างใน “Start” menu – “Programs” folder – “IDS Server” folder, a shortcut called “IDS Server”. ถ้าเลือก shortcut จะเริ่มดำเนินการทำงานของ IDS Server ใน console window. ผู้ใช้สามารถจะสั่งเริ่มดำเนินการทำงานด้วย command line ก็ได้.

```
C:\>installation_path\idss -con ↵
```

IDS Server Version 2.5 Lite Copyright (c) 1997 IDS Software.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IDS Server started...

ในการจบการทำงานของ IDS Server ใน Windows 95⁴ นั้น, ผู้ใช้ที่เพียงแค่กดปุ่ม Ctrl-C ใน console window หรือเปิด console window ลงเท่านั้นaltogether.

กระบวนการของ JDBC

กระบวนการของ IDS Server ในการติดต่อฐานข้อมูลสำหรับโปรแกรมภาษาจาวานั้นคือใช้ IDS JDBC Driver. Driver นี้มีพื้นฐานมาจาก JDBC API ล่าสุดใน JDK 1.1.

1. Overview

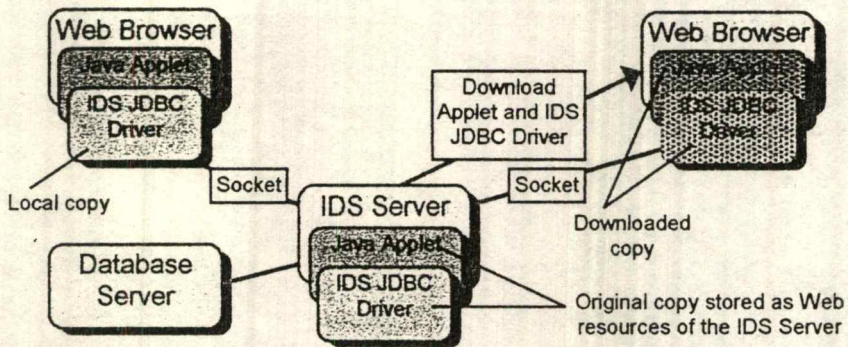
JavaSoft, เป็นผู้สร้าง JDBC API, ประเภทของ JDBC drivers แบ่งเป็น 4 ประเภทดังต่อไปนี้

- 1) JDBC-ODBC bridge
- 2) native-API partly-Java driver
- 3) net-protocol all-Java driver
- 4) native-protocol all-Java driver.

JDBC Driver ของ IDS Server ถือว่าเป็น Driver ชนิดที่ 3 “3. *net-protocol all-Java driver* ในการแปลงการเรียกใช้ JDBC ไปยัง DBMS โดย protocol อีกระยะที่ซึ่งการแปลงไปสู่ DBMS protocol นั้นทำโดย server. *Net server middleware* สามารถที่จะติดต่อไปยัง Java clients และได้ในหลายๆ ฐานข้อมูล. ข้อกำหนดของ protocol ในการใช้ขึ้นอยู่กับตัวแทนจำหน่าย (vendor) โดยทั่วไปแล้วทางเลือกของ JDBC นั้นจะยืดหยุ่นมาก. มันจึงดูเหมือนว่าตัวแทนจำหน่ายจะจัดการให้ผลิตภัณฑ์ของตนนั้นเหมาะสมสำหรับการใช้งานบนเครือข่าย ซึ่งยังต้องเพิ่มความต้องการสำหรับความปลอดภัย (security), เพื่อใช้ในการผ่าน ระบบรักษาความปลอดภัย (firewalls), และอื่นๆ อีกมากมาย

2. IDS JDBC Driver

IDS JDBC Driver เป็น Driver ประเภทที่ 3. การใช้ *net server middleware* โดย IDS JDBC Driver คือ IDS Server, ที่ซึ่งถือว่าเป็นที่ทำงานที่แท้จริงหลังจาก IDS JDBC Driver. การติดต่อไปยังฐานข้อมูลจะทำโดย driver เพื่อที่จะส่งกลับไปมาระหว่าง driver และ IDS Server. JDBC driver จำทำการตัดสินใจเลือก Java Socket ในการเชื่อมต่อ (connection) กลับไปยัง IDS Server. รูปที่ ก-3 แสดงให้เห็นถึงรูปแบบของ network ในการติดต่อไปยัง IDS Server, database server และ 2 เว็บเบราว์เซอร์



รูปที่ ก-3 รูปแบบ Network ของ IDS Server

IDS JDBC Driver คือ Java software package ที่ใช้อธิบายถึง JDBC API. Driver จะถูกเขียนด้วยภาษาจาวาเพียงอย่างเดียว ดังนั้น Platform ของ Drivers จึงเป็นอิสระและสามารถกระจายโดย Web servers หรือติดตั้งบน local Java package ใน client computers ทั่วไป. ดังรูป

ภาคผนวก ข

Open Database Connectivity (ODBC)

ODBC คืออะไร

Open Database Connectivity คือวิธีการติดต่อและเข้าถึงจากแอปพลิเคชันสู่ระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL เป็นมาตรฐานการเข้าถึงฐานข้อมูล ความสามารถในการต่อเชื่อมแบบนี้ทำให้แอปพลิเคชันสามารถเข้าถึงฐานข้อมูลได้หลายรูปแบบ ซึ่งทำให้ผู้พัฒนาโปรแกรมสามารถพัฒนาโปรแกรมไปได้โดยไม่ต้องทำการระบุชนิดของระบบจัดการฐานข้อมูล

แต่เดิมนั้นการพัฒนาโปรแกรมประยุกต์ที่ใช้งานเกี่ยวกับฐานข้อมูล การเข้าใช้ฐานข้อมูลของโปรแกรมเหล่านี้จะทำผ่านการเรียกใช้ SQL แบบฝังตัว (Embedded SQL) ซึ่งในขณะนั้นวิถีทางแบบนี้ก็ดูจะไปได้ดีทีเดียว เพราะว่าตัวโปรแกรมสามารถทำการเปลี่ยนรูปแบบของระบบไม่ว่าจะเป็นทางด้านฮาร์ดแวร์ หรือซอฟต์แวร์ได้หลายรูปแบบ รวมทั้งระบบปฏิบัติการด้วย (โดยการคอมไพล์ใหม่ทุกครั้งที่มีการย้ายระบบ)

อย่างไรก็ตามในการพัฒนาโปรแกรมในระบบที่ความแตกต่างกัน เช่นการเรียกใช้ข้อมูลของออราเคิลจากไมโครซอฟต์เอ็กเซล (Microsoft Excel) วิธีการเข้าถึงข้อมูลแบบเดิมนั้นจะต้องทำการคอมไพล์โค้ดของเอ็กเซลและออราเคิล โดยใช้ IBM precompiler และ Oracle precompiler ตามลำดับ ซึ่งจะเห็นว่าเป็นการยุ่งยากมากทีเดียว

วิธีการต่อเชื่อมแบบ ODBC จะให้ความสะดวกในการติดต่อข้อมูลมากกว่าวิธีการดั้งเดิมโดยการกำหนดมาตรฐานการต่อเชื่อมของข้อมูล (Data protocol, DBMS capability) และแนวทางนี้ได้ทำให้เกิดความคิดที่จะสร้างไครฟ์เวอร์สำหรับการติดต่อกับของฐานข้อมูลขึ้นมา (DLL)

ทฤษฎีการทำงานของ ODBC.

ส่วนติดต่อของการเชื่อมต่อฐานข้อมูลเปิด (ODBC : Open Database Connectivity) เป็นตัวทำให้โปรแกรมประยุกต์สามารถที่จะเข้าถึงข้อมูลในระบบจัดการฐานข้อมูล (DBMS : Database Management System) โดยใช้ภาษา SQL (Structure Query Language) เป็นมาตรฐานหลักในการเข้าถึงข้อมูล

ข้อดีของการติดต่อโดยใช้ ODBC

1. ฟังก์ชันของ ODBC อนุญาตให้แอปพลิเคชันติดต่อกับระบบจัดการฐานข้อมูลได้โดยสะดวก (การทำคำสั่ง SQL และการรับผลลัพธ์)
2. ใช้ภาษา SQL ตามมาตรฐาน SQL CAE, X/Open และ SQL Access Group (SAGX)
3. มีการกำหนดการส่งกลับรหัสความผิดพลาด (Error Code) เป็นมาตรฐานเดียวกัน
4. เป็นวิธีการมาตรฐานในการติดต่อกับระบบจัดการฐานข้อมูล

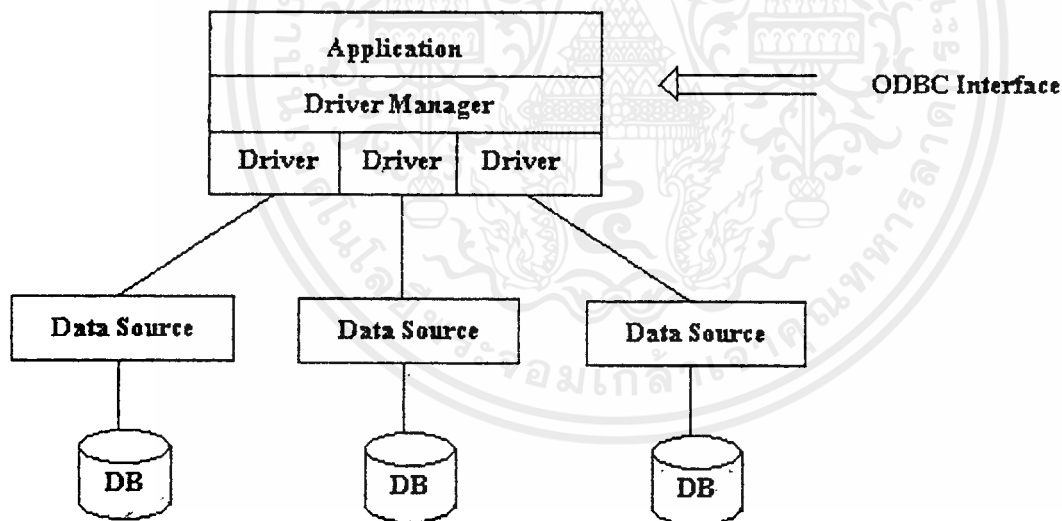
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. มีการกำหนดชนิดข้อมูล (Data Type) เป็นมาตรฐาน
6. ชุดคำสั่ง SQL สามารถกำหนดได้แม้ในขณะที่ Runtime
7. สามารถเขียนโปรแกรมชุดเดียว แต่สามารถเข้าใช้ระบบจัดการฐานข้อมูลได้หลายตัว
8. ตัวโปรแกรมไม่ต้องรับผิดชอบในการดูแลการติดต่อข้อมูลกับระบบจัดการฐานข้อมูล,
9. ค่าข้อมูลสามารถถูกส่งหรือรับได้ในรูปแบบที่สะดวกขึ้น

องค์ประกอบของ ODBC

สถาปัตยกรรมของ ODBC ประกอบด้วย 4 ส่วนสำคัญ

1. แอปพลิเคชัน ทำหน้าที่ประมวลผลและเรียกใช้ฟังก์ชันของ ODBC ตามคำสั่งภาษา SQL พร้อมทั้งทำการรับผลลัพธ์ด้วย
2. ตัวจัดการไดรฟ์เวอร์ (Driver Manager) ทำหน้าที่โหลดไดรฟ์เวอร์เชื่อมต่อกับแหล่งข้อมูล
3. ไดรฟ์เวอร์ (Driver) ทำหน้าที่ประมวลผลการเรียกใช้ฟังก์ชันของ ODBC ,ส่งคำสั่ง SQL ไปสู่แหล่งข้อมูลที่ต้องการ และทำการแปลงคำสั่งที่ส่งมาให้อยู่ในรูปแบบที่สนับสนุนคยระบบจัดการฐานข้อมูลแต่ละชนิดอีกด้วย
4. แหล่งข้อมูล (Data Source) เป็นแหล่งข้อมูลที่ใช้ต้องการเข้าถึง



รูปที่ ข - 1 องค์ประกอบของ ODBC

แอปพลิเคชัน

ตัวโปรแกรมจะเรียกใช้การต่อเชื่อม ODBC ในการทำงานต่อไปนี้

1. ร้องขอการต่อเชื่อมกับแหล่งข้อมูล
 2. ส่งคำสั่ง SQL สู่แหล่งข้อมูล
 3. กำหนดพื้นที่การจัดเก็บและรูปแบบของข้อมูลที่เป็นผลลัพธ์จาก SQL Request ร้องขอผลลัพธ์
- เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ร้องขอผลลัพธ์
5. ประมวลผลและจัดการกับข้อผิดพลาด
6. รายงานผลให้กับผู้ใช้ (ถ้าจำเป็น)
7. ร้องขอการ Commit หรือ Rollback สำหรับควบคุมการประมวลผล Transaction
8. ยกเลิกการติดต่อกับแหล่งข้อมูล

ตัวจัดการไครฟ์เวอร์

ตัวจัดการไครฟ์เวอร์ คือ DLL (Dynamic Link Library) และไลบรารีอื่น ๆ หน้าที่หลักของตัวจัดการไครฟ์เวอร์ก็คือการโหลดไครฟ์เวอร์ ส่วนหน้าที่อื่น ๆ ก็มีดังนี้

1. เรียกใช้ไฟล์ ODBC.INI เพื่อกำหนดชื่อของแหล่งข้อมูล (Data Source name) ให้กับไครฟ์เวอร์ DLL
2. ทำการประมวลผลการเริ่มต้นการเชื่อมต่อของ ODBC
3. เป็นจุดต่อเชื่อมระหว่างฟังก์ชันของ ODBC กับไครฟ์เวอร์แต่ละตัว
4. ทำการตรวจสอบพารามิเตอร์และลำดับการเรียกใช้ ODBC

ไครฟ์เวอร์

ไครฟ์เวอร์คือ DLL ที่สร้างฟังก์ชันของ ODBC และทำการโต้ตอบกับแหล่งข้อมูล ไครฟ์เวอร์ทำการตอบสนองการเรียกใช้ฟังก์ชันของ ODBC โดยจะทำงานต่อไปนี้

1. สร้างการเชื่อมต่อเชื่อมกับแหล่งข้อมูล
2. ส่งคำขอร้องให้กับแหล่งข้อมูล
3. แปลงข้อมูลจากรูปแบบหนึ่งสู่อีกรูปแบบหนึ่ง
4. ส่งผลลัพธ์กลับให้แอปพลิเคชัน
5. จัดการส่งข้อมูลความผิดพลาดให้อยู่ในรูปแบบรหัสมาตรฐานแล้วส่งกลับไปให้แอปพลิเคชัน
6. ทำหน้าที่จัดการและดูแลเคอร์เซอร์ (Cursor)

แหล่งข้อมูล (Data Source)

แหล่งข้อมูล หมายถึงการรวมกันของระบบจัดการฐานข้อมูล, ระบบปฏิบัติการ และระบบเครือข่าย โดยมีกรณีเฉพาะชนิด และประเภทลงไป หรืออีกนัยหนึ่งก็หมายความว่ากรณีที่แอปพลิเคชันทำการติดต่อกับระบบจัดการฐานข้อมูล ยี่ห้อนั้นบนระบบปฏิบัติการหนึ่งและเข้าถึงโดยระบบเครือข่ายชนิดหนึ่ง เช่น ออราเคิลที่วิ่งบนระบบปฏิบัติการ OS/2 โดยใช้ระบบเครือข่ายของ Novell Netware เป็นต้น

ภาคผนวก ค

SQL Syntax

SELECT Statement

```

SELECT [ALL | DISTINCT] select_list
      [INTO [[database.]owner.]table_name]
      [FROM
          [(database.)owner.]{table_name | view_name} [HOLDLOCK]
          ,[[database.]owner.]{table_name | view_name} [HOLDLOCK]]...]
      [WHERE search_conditions]
      [GROUP BY [ALL]
          aggregate_free_expression
          [,aggregate_free_expression]...]
      [HAVING search_conditions]
      [ORDER BY
          {[[[database.]owner.]{table_name. | view_name.}] column_name |
          select_list_number | expression [ASC | DESC]
          ,[[[database.]owner.]{table_name. | view_name.}] column_name |
          select_list_number | expression} [ASC | DESC]]...]
      [COMPUTE
          row_aggregate(column_name)
          [,row_aggregate(column_name)]]
      [BY column_name [, column_name]...]
      [FOR BROWSE]

```

INSERT Statement

```

INSERT [INTO]
      [[database.]owner.]{table_name | view_name}
      [(column_list)]
      {VALUES (constant_expression [, constant_expression]...) | select_statement}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UPDATE Statement

```

UPDATE [[database.]owner.]{table_name | view_name}
SET [[[[database.]owner.]{table_name. | view_name.}]
    column_name1 = {expression1 | NULL | (select_statement)}
    [,column_name2 = {expression2 | NULL | (select_statement)}...]
[WHERE search_conditions]

```

DELETE Statement

```

DELETE [FROM] [[database.]owner.]{table_name | view_name}
[WHERE search_conditions]

```



ภาคผนวก ง

โปรแกรมแสดงระบบงานแสดงผลการศึกษา ผ่านเครือข่ายอินเทอร์เน็ต

ในส่วนของโปรแกรมจะแสดงโปรแกรมของ Microsoft Access เป็นตัวอย่าง

```
import java.awt.*;
import java.applet.*;
import java.sql.*;
import javax.swing.*;

public class Student extends Applet {
    String driverName="ids.sql.IDSDriver";
    String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn='studentdb'";
    String Semester,sql;
    String user="pornchai",password="porn";
    boolean idPass;
    Connection con;
    // select ID : select by ID and newest Year and Term
    void textField1_EnterHit(Event event) {
        queryID();
        showPart();
    }
    // select Year : select by Year and Term=1
    void textField4_EnterHit(Event event) {
        Termchoice.select(0);
        showPart();
    }
    // select Term : select by Term is selected
    void choice1_Action(Event event) {
        showPart();
    }
    // Show Header Part : show id,sname,surname,year,semester
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public void queryID() {
    try {
        sql= "select sname,surname,max(year),semester "+
            "from student t1,register t2 "+
            "where t1.id=t2.id and t1.id= ? "+
            "group by sname,surname,semester ";
        PreparedStatement sqlStatement=con.prepareStatement(sql);
        sqlStatement.setString(1,IDField.getText());
        ResultSet results=sqlStatement.executeQuery();
        String Semester="1";
        if (results.next()){
            NameField.setText(results.getString(1));
            SufnameField.setText(results.getString(2));
            YearField.setText(results.getString(3));
            Semester=results.getString(4);
            if (Semester.compareTo("1")==0)
                Termchoice.select(0);
            else if (Semester.compareTo("2")==0)
                Termchoice.select(1);
            else Termchoice.select(2);
        }
        else {
            NameField.setText("");
            SurnameField.setText("");
            YearField.setText("");
            Termchoice.select(0);
        }
        results.close();
        sqlStatement.close();
    }
    catch (SQLException ex){
        while (ex !=null){

```



```

byte I=0;
while (results.next()){
    // MultiListaddTextCell(int, int, String)
    multiList1.addTextCell(I,0,results.getString(1));
    multiList1.addTextCell(I,1,results.getString(2));
    multiList1.addTextCell(I,2,results.getString(3));
    multiList1.addTextCell(I++,3,results.getString(4));
    SCredit+=results.getInt(2);
}

multiList1.redraw();
PresentUnitField.setText(""+SCredit);
results.close();
sqlStatement.close();
}
catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());
        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){
    ex.printStackTrace();
}
// ๕ Part of GPS
try{
    sql= "select grade,credit "+
        "from register t1,subjects t2 "+
        "where t1.subj_code=t2.subj_codè and id=? and year=? and t1.semester=? and status=? "+
        "and grade is not null";
    PreparedStatement sqlStatement=con.prepareStatement(sql);
    sqlStatement.setString(1,IDField.getText());

```

```

sqlStatement.setString(2,YearField.getText());
sqlStatement.setString(3,Semester);
sqlStatement.setString(4,"c");
ResultSet results=sqlStatement.executeQuery();
short SCredit=0;
float SGPS=0;
while (results.next()){
    SCredit+=results.getInt(2);
    if (((results.getString(1)).compareTo("A"))==0)
        SGPS+=(4*results.getInt(2));
    else if (((results.getString(1)).compareTo("B+"))==0)
        SGPS+=(3.5*results.getInt(2));
    else if (((results.getString(1)).compareTo("B"))==0)
        SGPS+=(3*results.getInt(2));
    else if (((results.getString(1)).compareTo("C+"))==0)
        SGPS+=(2.5*results.getInt(2));
    else if (((results.getString(1)).compareTo("C"))==0)
        SGPS+=(2*results.getInt(2));
    else if (((results.getString(1)).compareTo("D+"))==0)
        SGPS+=(1.5*results.getInt(2));
    else if (((results.getString(1)).compareTo("D"))==0)
        SGPS+=(1*results.getInt(2));
    else if (((results.getString(1)).compareTo("F"))==0)
        SGPS+=(0*results.getInt(2));
    else if (((results.getString(1)).compareTo("I"))==0)
        StatusField.setText("กรุณาติดต่ออาจารย์ประจำวิชา โดยด่วน");
    else StatusField.setText("ยังไม่สมบูรณ์");
}
PassPresentUnitField.setText(""+SCredit);
if (((""+SCredit).compareTo("0"))==0){
    StatusField.setText("ยังไม่สมบูรณ์");
}

```

```

        GPSField.setText("0");
    }
    else
    if ((SGPS % SCredit)==0)
        GPSField.setText(""+SGPS/SCredit);
    else    GPSField.setText(""+SGPS/SCredit).substring(0,4);
    if (SGPS/SCredit>=2.0)
        StatusField.setText("ผ่าน");
    else
    if (SGPS==0)
        StatusField.setText("ไม่มีวิชาที่ลงทะเบียน");
    else    StatusField.setText("PROPATION");
    results.close();
    sqlStatement.close();
}
catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());
        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){    ex.printStackTrace();
}
//        Part of GPA
try{
    sql=    "select grade,credit "+
            "from register t1,subjects t2 "+
            "where t1.subj_code=t2.subj_code and id=? and (year<? or "+
            "(year=? and t1.semester<=?)) and status=? ";
    PreparedStatement sqlStatement=con.prepareStatement(sql);

```

```

sqlStatement.setString(1,IDField.getText());
sqlStatement.setString(2,YearField.getText());
sqlStatement.setString(3,YearField.getText());
sqlStatement.setString(4,Semester);
sqlStatement.setString(5,"c");
ResultSet results=sqlStatement.executeQuery();
short SCredit=0;
while (results.next()){
    SCredit+=results.getInt(2);
}
SumUnitField.setText(""+SCredit);
results.close();
sqlStatement.close();
}
catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());
        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){    ex.printStackTrace();
}
try{
    sql=        "select grade,credit "+
                "from register t1,subjects t2 "+
                "where t1.subj_code=t2.subj_code and id=? and (year<? or "+
                "(year=? and t1.semester<=?)) and grade is not null and status=? ";
    PreparedStatement sqlStatement=con.prepareStatement(sql);
    sqlStatement.setString(1,IDField.getText());
    sqlStatement.setString(2,YearField.getText());

```

```

sqlStatement.setString(3,YearField.getText());
sqlStatement.setString(4,Semester);
sqlStatement.setString(5,"c");
ResultSet results=sqlStatement.executeQuery();
short SCredit=0;
float SGPA=0;
while (results.next()){
    SCredit+=results.getInt(2);
    if (((results.getString(1)).compareTo("A"))==0){
        SGPA+=(4*results.getInt(2))
    }
    else
    if (((results.getString(1)).compareTo("B+"))==0)
        SGPA+=(3.5*results.getInt(2));
    else
    if (((results.getString(1)).compareTo("B"))==0)
        SGPA+=(3*results.getInt(2));
    else
    if (((results.getString(1)).compareTo("C+"))==0)
        SGPA+=(2.5*results.getInt(2));
    else
    if (((results.getString(1)).compareTo("C"))==0)
        SGPA+=(2*results.getInt(2));
    else if (((results.getString(1)).compareTo("D+"))==0)
        SGPA+=(1.5*results.getInt(2));
    else if (((results.getString(1)).compareTo("D"))==0)
        SGPA+=(1*results.getInt(2));
    else if (((results.getString(1)).compareTo("F"))==0)
        SGPA+=(0*results.getInt(2));
    else if (((results.getString(1)).compareTo("I"))==0)
        StatusField.setText("กรุณาติดต่อบริษัทโดยด่วน");
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
PassSumUnitField.setText(""+SCredit);
if (((""+SCredit).compareTo("0")==0)
    GPAField.setText("0");
else GPAField.setText(""+SGPA/SCredit.substring(0,4));
results.close();
sqlStatement.close();
}
catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());
        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){    ex.printStackTrace();
}
}
public void init() {
    super.init();
    // Take out this line if you don't use symantec.itools.net.RelativeURL
    symantec.itools.lang.Context.setDocumentBase(getDocumentBase());
    //{{INIT_CONTROLS
    setLayout(null);
    addNotify();
    resize(541,453);
    setBackground(new Color(2138847));
    NameField = new java.awt.TextField();
    NameField.disable();
    NameField.reshape(72,108,156,22);
    add(NameField);

```

```

label3 = new java.awt.Label("ชื่อ");
label3.reshape(12,108,37,24);
add(label3);

Termchoice = new java.awt.Choice();
Termchoice.addItem("1");
Termchoice.addItem("2");
Termchoice.addItem("SUMMER");
add(Termchoice);

Termchoice.reshape(444,72,83,21);
label5 = new java.awt.Label("เทอม");
label5.reshape(384,72,49,25);
add(label5);

YearField = new java.awt.TextField();
YearField.reshape(300,72,65,21);
add(YearField);

label4 = new java.awt.Label("นามสกุล");
label4.reshape(228,108,73,24);
add(label4);

label2 = new java.awt.Label("ปี");
label2.reshape(252,72,36,25);
add(label2);

label1 = new java.awt.Label("เลขทะเบียนนักศึกษา");
label1.reshape(12,72,120,26);
add(label1);

IDField = new java.awt.TextField();
IDField.disable();
IDField.reshape(156,72,72,21);
add(IDField);

label11 = new java.awt.Label("หน่วยกิต");
label11.reshape(108,336,61,23);
add(label11);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PassPresentUnitField.disable();
PassPresentUnitField.reshape(204,360,50,22);
add(PassPresentUnitField);
GPSField = new java.awt.TextField();
GPSField.disable();
GPSField.reshape(324,360,50,22);
add(GPSField);
GPAField = new java.awt.TextField();
GPAField.disable();
GPAField.reshape(324,396,50,22);
add(GPAField);
label13 = new java.awt.Label("ทอมปัจจุบัน");
label13.reshape(12,360,98,26);
add(label13);
SurnameField = new java.awt.TextField();
SurnameField.disable();
SurnameField.reshape(300,108,228,22);
add(SurnameField);
StatusField = new java.awt.TextField();
StatusField.disable();
StatusField.reshape(396,360,134,22);
add(StatusField);
SumUnitField = new java.awt.TextField();
SumUnitField.disable();
SumUnitField.reshape(120,396,50,22);
add(SumUnitField);
PassSumUnitField = new java.awt.TextField();
PassSumUnitField.disable();
PassSumUnitField.reshape(204,396,50,22);
add(PassSumUnitField);
multiList1 = new symantec.itools.awt.MultiList();

```

```

multiList1.reshape(12,144,517,181);
add(multiList1);
multiList1.setCellFont(new java.awt.Font("Dialog",java.awt.Font.PLAIN,10));
multiList1.setHeadingFont(new java.awt.Font("Dialog",java.awt.Font.PLAIN,10));
{
java.lang.String[] tempString = new java.lang.String[4];
tempString[0] = new java.lang.String("รหัสวิชา");
tempString[1] = new java.lang.String("หน่วยกิต");
tempString[2] = new java.lang.String("ผลการศึกษา");
tempString[3] = new java.lang.String("รายชื่อวิชา");
multiList1.setHeadings(tempString);
}
label6 = new java.awt.Label("ระบบแสดงผลการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ต");
label6.reshape(60,12,432,42);
label6.setFont(new Font("Dialog", Font.BOLD, 20));
add(label6);
//}}
}

private OurFrame m_frame = null;
public void paint(Graphics g)
{
if (idPass==false){      DialogData data = m_frame.getData();
if (data != null){
try {
sql= "select id,ipassword from tpassword "+
"where id= ? and ipassword=?";
PreparedStatement sqlStatement=con.prepareStatement(sql);
sqlStatement.setString(1,data.toNameString());
sqlStatement.setString(2,data.toPassString());
ResultSet results=sqlStatement.executeQuery();
if (results.next()){

```

```

        IDField.setText(data.toNameString());
        queryID();
        showPart();
    }

else{
    errb = new ErrorBox("Error Login","ไม่มีรายชื่อนักศึกษาอยู่ในทะเบียน");
    errb.show();
    }
    results.close();
    sqlStatement.close();
    idPass=true;
    }
catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());
        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){
    ex.printStackTrace();
}}}}

public void start() {
try {
    Class.forName(driverName);
    con=DriverManager.getConnection(dbURL,user,password);
    m_frame = new OurFrame(this);
    }

catch (SQLException ex){
    while (ex !=null){
        errb = new ErrorBox("Error",ex.getMessage());

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        errb.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){    ex.printStackTrace();
}
}

public void stop() {
    try {        con.close();
    }
    catch (SQLException ex){
        while (ex !=null){
            errb = new ErrorBox("Error",ex.getMessage());
            errb.show();
            ex=ex.getNextException();
        }
    }
    catch (java.lang.Exception ex){    ex.printStackTrace();
    }
}

public boolean handleEvent(Event event) {
    if (event.target == IDField && event.id == Event.ACTION_EVENT) {
        textField1_EnterHit(event);
        return true;
    }
    if (event.target == Termchoice && event.id == Event.ACTION_EVENT) {
        choice1_Action(event);
        return true;
    }
    if (event.target == YearField && event.id == Event.ACTION_EVENT) {

```

```

        textField4_EnterHit(event);
        return true;
    }

    return super.handleEvent(event);
}

//{{DECLARE_CONTROLS
java.awt.TextField NameField;
java.awt.Label label3;
java.awt.Choice Termchoice;
java.awt.Label label5;
java.awt.TextField YearField;
java.awt.Label label4;
java.awt.Label label2;
java.awt.Label label1;
java.awt.TextField IDField;
java.awt.Label label11;
java.awt.Label label12;
java.awt.TextField PresentUnitField;
java.awt.Label label14;
java.awt.Label label15;
java.awt.Label label16;
java.awt.Label label17;
java.awt.Label label18;
java.awt.TextField PassPresentUnitField;
java.awt.TextField GPSField;
java.awt.TextField GPAField;
java.awt.Label label13;
java.awt.TextField SurnameField;
java.awt.TextField StatusField;
java.awt.TextField SumUnitField;
java.awt.TextField PassSumUnitField;

```

```

    symantec.itools.awt.MultiList multiList1;
    java.awt.Label label6;
    ErrorBox errb;
    //}}
}

class OurFrame extends Frame
{
    // child dialog box
    private AutoDialog m_autodialog = null;
    // parent applet
    private Applet m_parent = null;
    // data contained in dialog box
    private DialogData m_data = null;
    public OurFrame(Applet parent)
    {
        super("Auto Dialog");
        m_parent = parent;
        // be sure to set a font (the size is unimportant
        // because it will be changed by the AutoDialog class)
        setFont(new Font("Dialog", Font.PLAIN, 12));
        // now create the AutoDialog and let it add the
        // components to our frame
        m_autodialog = new AutoDialog(this);
        m_autodialog.CreateControls();
        show();
    }
    public boolean action(Event event, Object obj)
    {
        // if this is our button...
        Object target = event.target;
        if (target instanceof Button)

```

```

{
    Button button = (Button)target;
    String buttonLabel = button.getLabel();
    hide();
    if (buttonLabel.compareTo("OK") == 0)
    {
        // ...get the data from the dialog box...
        String sFirstName=m_autodialog.IDC_EDIT1.getText();
        String sLastName =m_autodialog.IDC_EDIT2.getText();
        // ...create a DialogData object...
        m_data = new DialogData(sFirstName, sLastName);
        // ...and force the applet to output the data
        m_parent.repaint();
    }
    else{        dispose();
    }    // hide the dialog box if either button is pressed
    return true;
}
return false;
}
public DialogData getData()
{    return m_data;
}
}

class DialogData
{
private String m_sFirstName;
    private String m_sLastName;
    public DialogData(String sFN, String sLN)
    {
        m_sFirstName = sFN;

```

```

        m_sLastName = sLN;
    }

    public String toNameString(){
        String s;
        s = m_sFirstName;
        return s;
    }

    public String toPassString(){
        String s;
        s = m_sLastName;
        return s;
    }
}

```

ในส่วนของฐานข้อมูลอื่นจะต้องเซตโปรแกรมดังนี้

ฐานข้อมูล SQL Server

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn=sqlserverdb";
```

```
String user = "sa",password="";
```

ฐานข้อมูล Sybase SQL Anywhere

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn=kmitl_db";
```

```
String user = "dba",password="sql";
```

ฐานข้อมูล Oracle

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn=oracledb";
```

```
String user = "Pornchai",password="porny";
```

ภาคผนวก จ

โปรแกรมระบบลงทะเบียนการเรียนผ่านเครือข่ายอินเทอร์เน็ต

โปรแกรมลงทะเบียนผ่านเครือข่ายอินเทอร์เน็ตในส่วนของ Microsoft Access

```
import java.awt.*;
import java.applet.*;
import java.util.*;
import java.sql.*;

public class Account extends Applet {
    String driverName="ids.sql.IDSDriver";
    String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn='studentdb'";
    String user="pornchai",password="porn";
    Connection con;
    String Semester,sql,id;
    int SumCredit=0;
    int SubTotal=0;           //SumCredit*50
    java.util.Date d = new java.util.Date();    // use class Date of packgate java.util
    String y = "19"+d.getYear();    // get year from system

void IDField_EnterHit(Event event) {
    try{
        Class.forName(driverName);
        con=DriverManager.getConnection(dbURL,user,password);
        String sql=        "select sname,surname,t1.class,department,faculty "+
            "from student t1,classes t2 " +
            "where t1.class=t2.class and t1.id= ? " +
            "group by sname,surname,t1.class,department,faculty";
        PreparedStatement sqlStatement=con.prepareStatement(sql);
        sqlStatement.setString(1,IDField.getText());
        ResultSet results=sqlStatement.executeQuery();
        if (results.next()) {
```

```

SurnameField.setText(results.getString(2));
ClassField.setText(results.getString(3));
DepartField.setText(results.getString(4));
FacultyField.setText(results.getString(5));
CyearField.setText("");
CyearField.setEditable(true);
CyearField.setEnabled();
TermChoice.setEnabled();
IDField.disable();
} else { NameField.setText("");
SurnameField.setText("");
ClassField.setText("");
DepartField.setText("");
FacultyField.setText("");
CyearField.disable();
TermChoice.disable();
}
YearField.setText(y);
results.close();
sqlStatement.close();
con.close();
} //close try
catch (SQLException ex){
while (ex!=null){
er = new ErrorBox("Error", "ไม่พบรหัสนักศึกษา");
er.show();
ex=ex.getNextException();
}
}
catch (java.lang.Exception ex){ ex.printStackTrace();
}
}

```

```

} //close scope
void Clearbutton_Clicked(Event event){
    CyearField.setText("");
    CyearField.disable();
    ListSubject.clear();
    ListSelected.clear();
    IDField.enable();
    SubtotalField.setText("");
    SumCreditField.setText("");
    SumCredit=0;
    SubTotal=0;
}
void ConfirmButton_Clicked(Event event) {
    ListSubject.clear();
    int num=ListSelected.countItems();
    if(num != 0){
        int c=ListSelected.countItems();
        try{
            Class.forName(driverName);
            con=DriverManager.getConnection(dbURL,user,password);
            for(int i=0; i<c; ++i)
            {
                sql= "insert into register "
                    +"values(" +IDField.getText()+","
                    +(ListSelected.getItem(i)).substring(0,8)+"',"
                    +TermChoice.getSelectedItem()+"',"
                    +YearField.getText()+"','',N)";
                PreparedStatement sqlStatement=con.prepareStatement(sql);
                int count=sqlStatement.executeUpdate();
                sqlStatement.close();
            }
        }
    }
}

```

```

DetailFrame df = new DetailFrame(SumCredit);
df.show();
con.close();
}
catch (SQLException ex){
    while (ex!=null){
        er = new ErrorBox("Error","การลงทะเบียนไม่สมบูรณ์
        โปรดตรวจสอบข้อมูลก่อนการลงทะเบียน");
        er.show();
        ex=ex.getNextException();
    }
}
catch (java.lang.Exception ex){    ex.printStackTrace();
}
}
void RemoveButton_Clicked(Event event) {
    int i=ListSelected.getSelectedIndex();
    if (i>=0){
        char ch=(ListSelected.getItem(i)).charAt(19);
        int credit=ch-48;
        SumCredit=SumCredit-credit;//integer
        SubTotal=SumCredit*50;
        SumCreditField.setText(""+SumCredit);//display in SumCredit textField
        SubtotalField.setText(""+SubTotal);//display in Subtotal textField
        ListSelected.delItem(i);
    }
}
}
void Addbutton_Clicked(Event event) {
    boolean nohad=true;
    int i=ListSubject.getSelectedIndex();
    if (i>=0){
        String s = ListSubject.getSelectedItem();

```



```

else { for(int j=0; j<num; ++j){
    boolean nohad=true;

    int n=ListSelected.countItems();//include new item
    String s=ListSubject.getItem(j);//it already ?
    for(int k=0; k<n; ++k){ //sequential search in ListSelected
        String str=ListSelected.getItem(k);
        if((s.compareTo(str))==0) nohad=false;
    }
    if(nohad){
        ListSelected.addItem(s);
        char ch=s.charAt(19);
        int credit=ch-48;
        SumCredit=SumCredit+credit;
        SubTotal=SumCredit*50;
        SumCreditField.setText(""+SumCredit);
        SubtotalField.setText(""+SubTotal);
    }
}

for(int j=0; j<num; ++j){ ListSubject.delItem(0); }
}

void CancelButton_Clicked(Event event) {
    System.exit(0);
}

void CyearField_EnterHit(Event event) {
    ListSubject.clear();
    try{
        Class.forName(driverName);
        con=DriverManager.getConnection(dbURL,user,password);
        byte i=0;
        sql= "select subj_code,credit,subject "+
            "from subjects "+
            "where class=? and cyear=? and semester=? "+

```

```

        "order by subj_code";

        PreparedStatement sqlStatement=con.prepareStatement(sql);
        sqlStatement.setString(1,ClassField.getText());
        sqlStatement.setString(2,CyearField.getText());
        sqlStatement.setString(3,TermChoice.getSelectedItemId());
        ResultSet results=sqlStatement.executeQuery();
        while (results.next()){
            ListSubject.addItem(results.getString(1)+" "+
            results.getString(2)+" "+
            results.getString(3));
        }
        results.close();
        sqlStatement.close();
        con.close();
    }
    catch (SQLException ex){
        while (ex!=null){
            er = new ErrorBox("Error","การลงไม่สมบูรณ์
            โปรดตรวจสอบข้อมูลก่อนการลงทะเบียน");
            er.show();
            ex=ex.getNextException();
        }
    }
    catch (java.lang.Exception ex){
        ex.printStackTrace();
    }
    IDField.disable();
}

public boolean handleEvent(Event event) {
    if (event.target == CyearField && event.id == Event.ACTION_EVENT) {
        Allbutton.enable();
        Addbutton.enable();
    }
}

```

```

        ConfirmButton.enable();
        TermChoice.disable();
        CyearField_EnterHit(event);
        return true;
    }

    if (event.target == TermChoice && event.id == Event.ACTION_EVENT) {
        Allbutton.enable();
        Addbutton.enable();
        ConfirmButton.enable();
        TermChoice.disable();
        CyearField_EnterHit(event);
        return true;
    }

    if (event.target == Allbutton && event.id == Event.ACTION_EVENT) {
        RemoveButton.enable();
        Allbutton_Clicked(event);
        return true;
    }

    if (event.target == Addbutton && event.id == Event.ACTION_EVENT) {
        RemoveButton.enable();
        Addbutton_Clicked(event);
        return true;
    }

    if (event.target == RemoveButton && event.id == Event.ACTION_EVENT) {
        RemoveButton_Clicked(event);
        return true;
    }

    if (event.target == ConfirmButton && event.id == Event.ACTION_EVENT) {
        ConfirmButton_Clicked(event);
        return true;
    }
}

```

```

if (event.target == Clearbutton && event.id == Event.ACTION_EVENT) {
    Clearbutton_Clicked(event);
    return true;
}

if (event.target == IDField && event.id == Event.ACTION_EVENT) {
    IDField_EnterHit(event);
    return true;
}

return super.handleEvent(event);
}

public void init() {
    super.init();
    // Take out this line if you don't use symantec.itools.net.RelativeURL
    symantec.itools.lang.Context.setDocumentBase(getDocumentBase());
    //{{INIT_CONTROLS
    setLayout(null);
    addNotify();
    resize(610,538);
    setForeground(new Color(4210752));
    setBackground(new Color(12632256));
    label5 = new java.awt.Label("ภาควิชา");
    label5.reshape(120,72,54,21);
    add(label5);

    ClassField = new java.awt.TextField();
    ClassField.disable();
    ClassField.reshape(60,72,31,25);
    add(ClassField);
    label4 = new java.awt.Label("ห้อง");
    label4.reshape(12,72,36,25);
    add(label4);

    SurnameField = new java.awt.TextField();

```

```

SurnameField.disable();
SurnameField.reshape(444,36,156,25);
add(SurnameField);
FacultyField = new java.awt.TextField();
FacultyField.disable();
FacultyField.reshape(444,72,156,25);
add(FacultyField);
DepartField = new java.awt.TextField();
DepartField.disable();
DepartField.reshape(180,72,204,25);
add(DepartField);
label6 = new java.awt.Label("คณะ");
label6.reshape(396,72,36,21);
add(label6);
label3 = new java.awt.Label("นามสกุล");
label3.reshape(384,36,57,25);
add(label3);
label2 = new java.awt.Label("ชื่อ");
label2.reshape(228,36,33,21);
add(label2);
label1 = new java.awt.Label("เลขทะเบียนนักศึกษา");
label1.reshape(12,36,114,21);
label1.setBackground(new Color(12632256));
add(label1);
label8 = new java.awt.Label("กรุณตรวจสอบข้อมูลการลงทะเบียนและรายชื่อวิชาให้ถูกต้อง");
label8.reshape(12,108,342,24);
label8.setFont(new Font("Dialog", Font.BOLD, 15));
label8.setForeground(new Color(255));
add(label8);
label9 = new java.awt.Label("ชั้นปีที่");
label9.reshape(12,144,46,25);

```

```

add(label9);
CyearField = new java.awt.TextField();
CyearField.disable();
CyearField.reshape(60,144,31,25);
add(CyearField);
label10 = new java.awt.Label("ภาคการเวียนที่ ");
label10.reshape(120,144,91,25);
add(label10);
label11 = new java.awt.Label("ปีการศึกษา");
label11.reshape(324,144,72,25);
add(label11);
YearField = new java.awt.TextField();
YearField.disable();
YearField.reshape(396,144,77,25);
add(YearField);
TermChoice = new java.awt.Choice();
TermChoice.addItem("1");
TermChoice.addItem("2");
TermChoice.addItem("Summer");
TermChoice.disable();
add(TermChoice);
TermChoice.reshape(216,144,108,25);
label13 = new java.awt.Label("รายชื่อวิชา");
label13.reshape(216,180,68,21);
add(label13);
label14 = new java.awt.Label("จำนวนหน่วยกิต");
label14.reshape(72,180,95,21);
add(label14);
label12 = new java.awt.Label("รหัสวิชา");
label12.reshape(12,180,56,21);
add(label12);

```

```

Allbutton = new java.awt.Button("Add All");
Allbutton.disable();
Allbutton.reshape(168,312,96,25);
add(Allbutton);
label7 = new java.awt.Label("ระบบลงทะเบียนนิสิตศึกษาผ่านเครือข่ายอินเทอร์เน็ต");
label7.reshape(132,0,360,28);
label7.setFont(new Font("Dialog", Font.BOLD, 18));
add(label7);
label15 = new java.awt.Label("รายวิชาที่นักศึกษาต้องการลงทะเบียน  
โปรดตรวจสอบด้วยความระมัดระวัง");
label15.reshape(12,348,433,24);
label15.setFont(new Font("Dialog", Font.BOLD, 15));
label15.setForeground(new Color(255));
add(label15);
label16 = new java.awt.Label("รวม");
label16.reshape(12,504,36,25);
add(label16);
label17 = new java.awt.Label("หน่วยกิต");
label17.reshape(96,504,59,25);
add(label17);
SumCreditField = new java.awt.TextField();
SumCreditField.disable();
SumCreditField.reshape(48,504,48,25);
add(SumCreditField);
ConfirmButton = new java.awt.Button("Confirm");
ConfirmButton.disable();
ConfirmButton.reshape(396,504,96,25);
add(ConfirmButton);
RemoveButton = new java.awt.Button("Remove");
RemoveButton.disable();
RemoveButton.reshape(396,312,96,25);

```

```

add(RemoveButton);
label18 = new java.awt.Label("รวมเงิน");
label18.reshape(168,504,55,25);
add(label18);
label19 = new java.awt.Label("บาท");
label19.reshape(324,504,35,25);
add(label19);
label20 = new java.awt.Label("เลือรายวิชาใน List");
label20.reshape(12,312,125,24);
label20.setFont(new Font("Dialog", Font.BOLD, 15));
add(label20);
IDField = new java.awt.TextField();
IDField.reshape(132,36,84,25);
add(IDField);
Addbutton = new java.awt.Button("Add");
Addbutton.disable();
Addbutton.reshape(288,312,96,25);
add(Addbutton);
SubtotalField = new java.awt.TextField();
SubtotalField.disable();
SubtotalField.reshape(228,504,87,25);
add(SubtotalField);
ListSubject = new java.awt.List(0,false);
add(ListSubject);
ListSubject.reshape(12,204,588,97);
ListSelected = new java.awt.List(0,false);
add(ListSelected);
ListSelected.reshape(12,384,588,110);
Clearbutton = new java.awt.Button("Clear");
Clearbutton.reshape(504,312,96,25);
add(Clearbutton);

```

```

NameField = new java.awt.TextField();
NameField.disable();
NameField.reshape(264,36,120,25);
add(NameField);
//}}
}

//{{{DECLARE_CONTROLS
java.awt.Label label5;
java.awt.TextField ClassField;
java.awt.Label label4;
java.awt.TextField SurnameField;
java.awt.TextField FacultyField;
java.awt.TextField DepartField;
java.awt.Label label6;
java.awt.Label label3;
java.awt.Label label2;
java.awt.Label label1;
java.awt.Label label8;
java.awt.Label label9;
java.awt.TextField CyearField;
java.awt.Label label10;
java.awt.Label label11;
java.awt.TextField YearField;
java.awt.Choice TermChoice;
java.awt.Label label13;
java.awt.Label label14;
java.awt.Label label12;
java.awt.Button Allbutton;
java.awt.Label label7;
java.awt.Label label15;
java.awt.Label label16;

```

```

java.awt.Label label17;
java.awt.TextField SumCreditField;
java.awt.Button ConfirmButton;
java.awt.Button RemoveButton;
java.awt.Label label18;
java.awt.Label label19;
java.awt.Label label20;
java.awt.TextField IDField;
java.awt.Button Addbutton;
java.awt.TextField SubtotalField;
java.awt.List ListSubject;
java.awt.List ListSelected;
java.awt.Button Clearbutton;
java.awt.TextField NameField;
//}}
ErrorBox er;
}

```

ในส่วนของฐานข้อมูลอื่นจะต้องเซต โปรแกรมดังนี้

ฐานข้อมูล SQL Server

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn='sqlserverdb'";
```

```
String user="sa",password="";
```

ฐานข้อมูล Sybase SQL Anywhere

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn='kmitl_db'";
```

```
String user="dba",password="sql";
```

ฐานข้อมูล Oracle

```
String dbURL="jdbc:ids://161.246.6.81:12/conn?dsn='oracledb'";
```

```
String user="Pornchai",password="porny";
```

ภาคผนวก ฉ

คู่มือการใช้งาน

การติดตั้งโปรแกรม

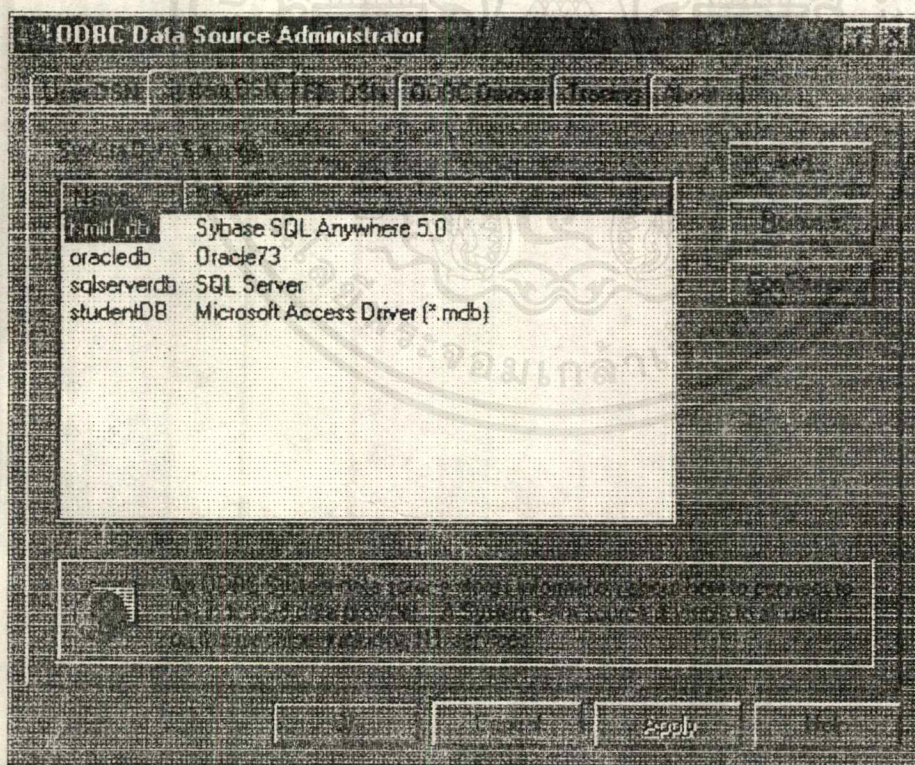
การติดตั้งโปรแกรม จะใช้เซิร์ฟเวอร์ที่เป็น Window NT และบน Window NT นี้จะติดตั้งโปรแกรมต่อไปนี้

1. Microsoft Access
2. Sybase SQL Anywhere
3. SQL Server
4. Personal Oracle
5. IDS

การติดตั้ง ODBC

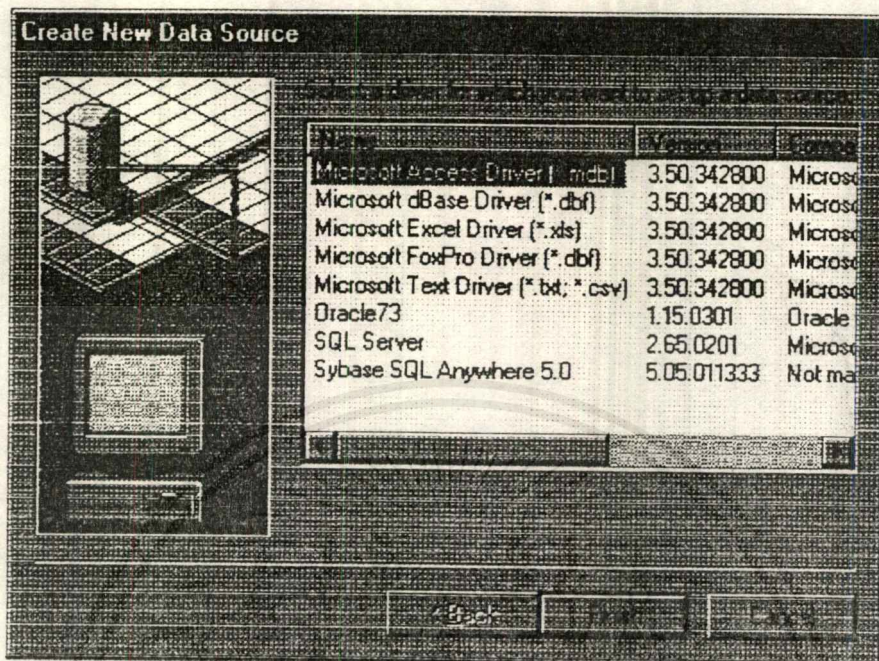
เมื่อทำการติดตั้งโปรแกรมเสร็จเรียบร้อยแล้วจะทำการติดตั้ง ODBC ซึ่งมีขั้นตอนดังนี้

- 1 เลือก Control Panel และ 32 bit ODBC
- 2 จะขึ้นหน้าต่างดังรูป



รูปที่ ฉ-1 การเซ็ท ODBC Data Source

3 ทำการเพิ่ม Data Source Name โดยกดปุ่ม Add จะแสดงหน้าต่างดังรูป

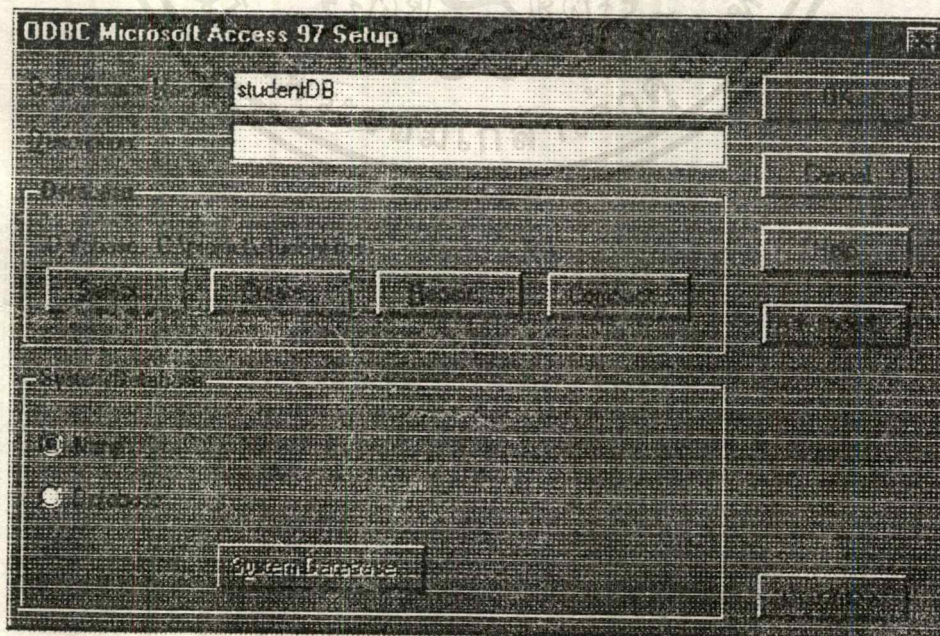


รูปที่ ๓-2 การเลือก New Data Source

4 ทำการเชื่อมต่อของแต่ละ database ดังนี้

4.1 Microsoft Access ทำการเชื่อมต่อดังนี้

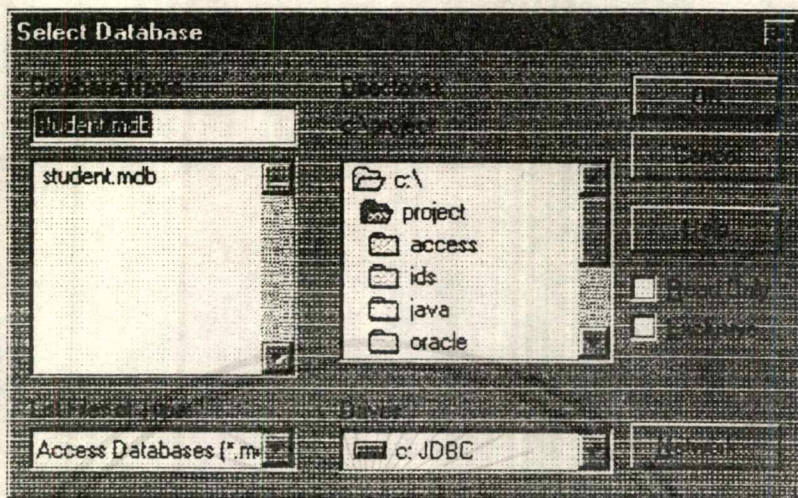
4.1.1 เลือก Microsoft Access Driver (*.mdb) กดปุ่ม Finish จะแสดงหน้าต่างดังรูป



รูปที่ ๓-3 การเชื่อมต่อ Data Source ของ Microsoft Access

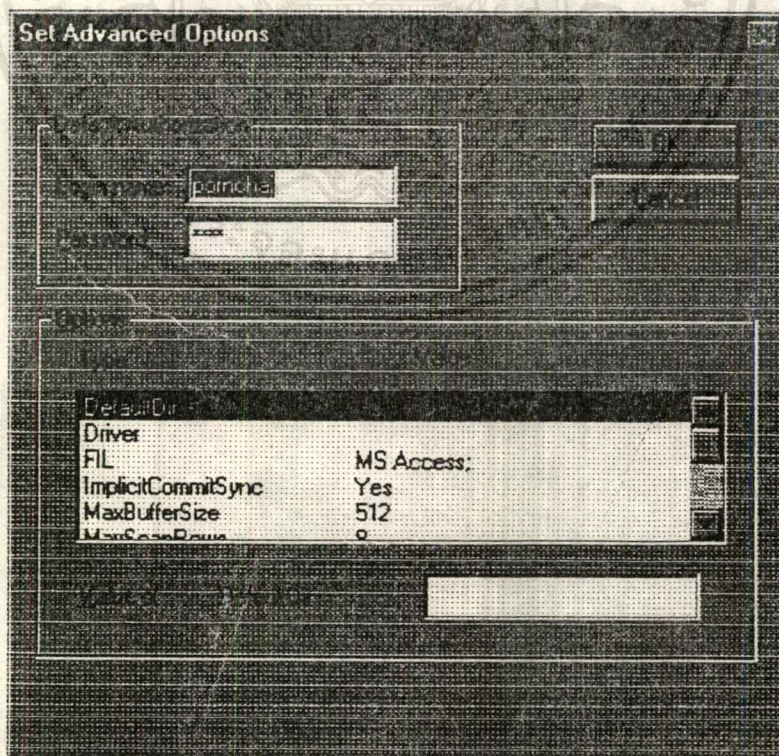
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีลายน้ำใบนี้

4.1.2 ป้อนชื่อของไฟล์ฐานข้อมูลของ Access หรือจะกดปุ่ม select เพื่อเลือกหาชื่อไฟล์ข้อมูลดังแสดงในรูป



รูปที่ ๓-4 การระบุชื่อไฟล์ฐานข้อมูล ของ Microsoft Access

4.1.3 เมื่อเลือกได้แล้วกดปุ่ม OK จากนั้นกดปุ่ม Advance เพื่อที่จะใส่ข้อมูลที่ใช้ในการติดต่อ



รูปที่ ๓-5 การระบุรายละเอียดในการติดต่อ ของ Microsoft Access

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้มาใช้ประโยชน์ดังการตั้ง
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 ป้อน Login Name และ Password ของผู้สร้างฐานข้อมูลนั้น จากนั้นกดปุ่ม OK

4.2 Sybase SQL Anywhere ทำการเชื่อมต่อโดยกำหนด Data Source Name และกำหนด ไฟล์ฐานข้อมูล โดยการ กดปุ่ม Browse จากนั้นกำหนด ผู้ใช้ไฟล์ฐานข้อมูลและรหัสลับ

SQL Anywhere ODBC Configuration

Data Source Name:

Description:

Connection Information

User ID:

Password:

Server Name:

Database Name:

Database Startup

Database File:

Local Network Custom

Additional Connection Options

Translator Name:

Microsoft Applications (Keys in SQLStatistics)

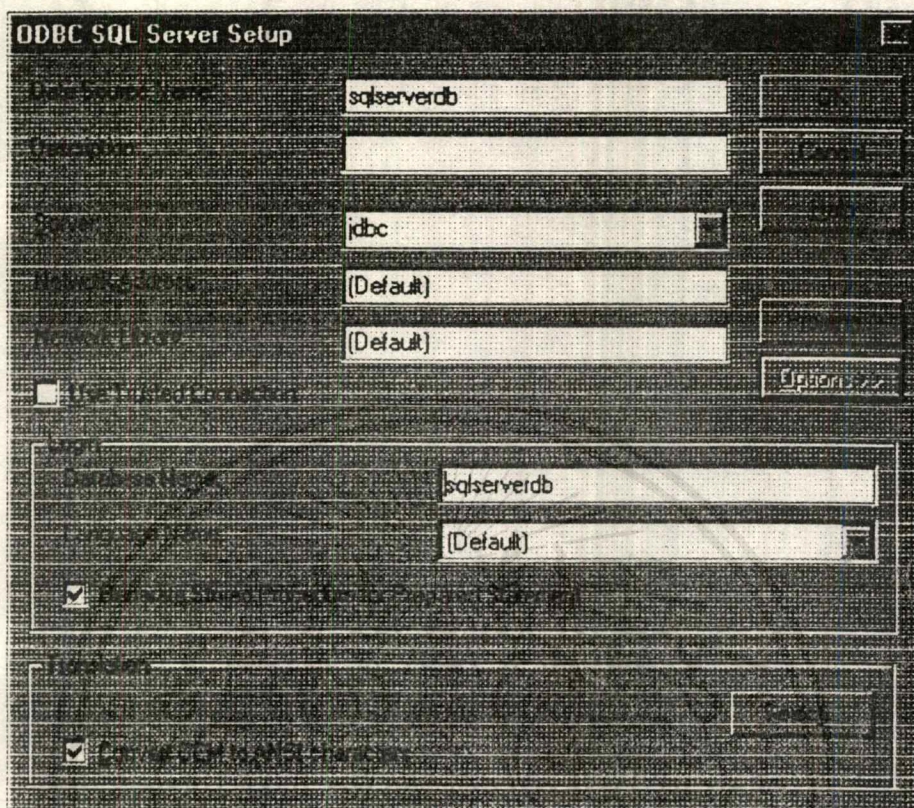
Prevent Driver not Capable errors

Delay AutoCommit until statement close

รูปที่ ๓-6 การระบุรายละเอียดในการติดต่อ ของ Sybase SQL Anywhere

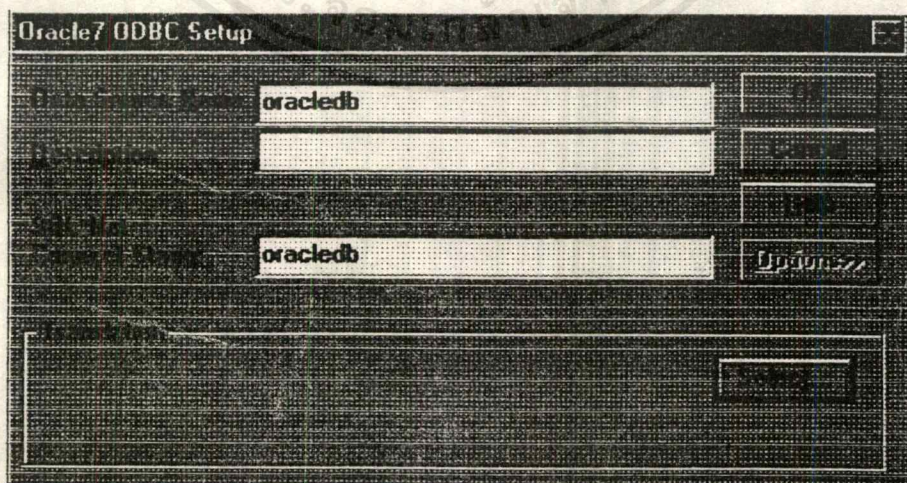
4.3 SQL Server ทำการเชื่อมต่อ

โดยการกำหนด Data Source Name เซิร์ฟเวอร์และไฟล์ฐานข้อมูล



รูปที่ ๓-7 การระบุรายละเอียดในการติดต่อ ของ SQL Server

4.4 Oracle ทำการกำหนดชื่อของ Data Source Name และชื่อของไฟล์ฐานข้อมูล

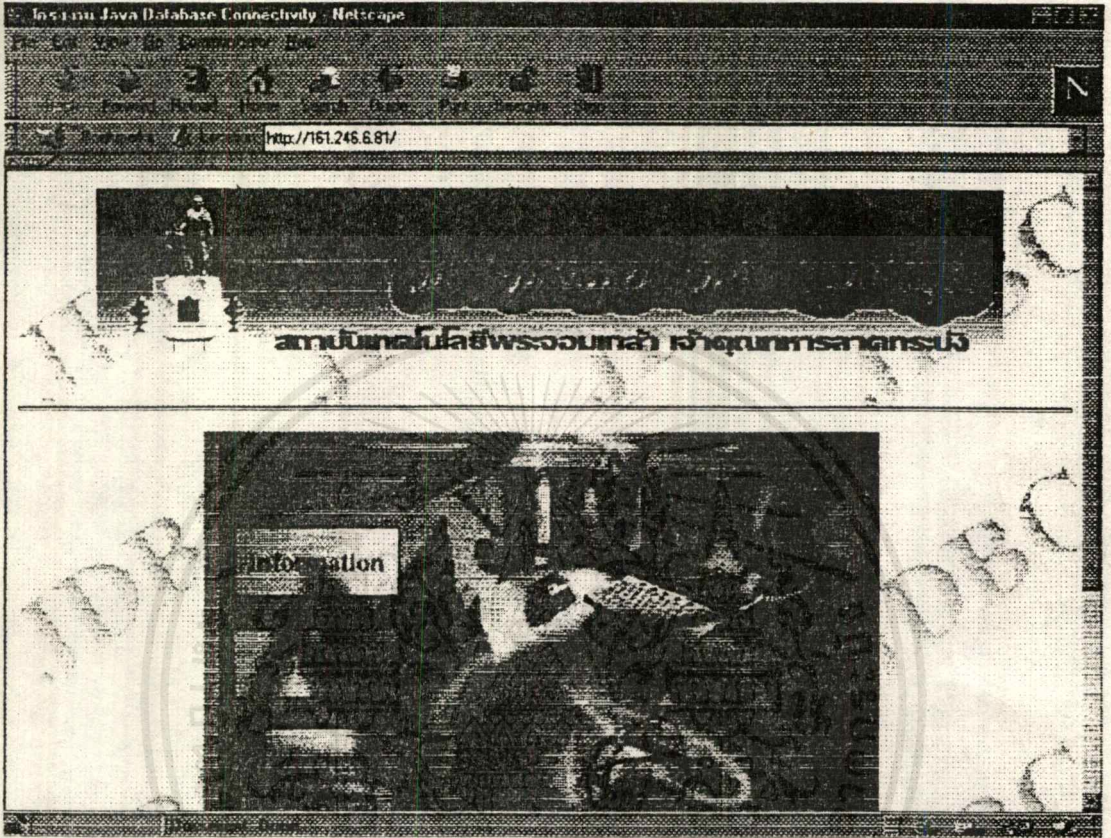


รูปที่ ๓-8 การระบุรายละเอียดในการติดต่อ ของ Oracle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานโปรแกรม

1 กำหนด URL ของเซิร์ฟเวอร์ ในที่นี้กำหนดเป็น 161.246.6.81 เป็นตัวที่ให้บริการเกี่ยวฐานข้อมูลทั้งหมด ดังรูป



รูปที่ ๑-๑ แสดงเว็บเพจที่สร้าง

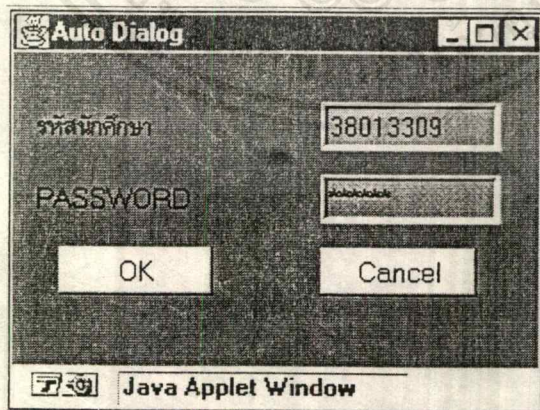
- 2 เลือก DEMO เพื่อแสดงโปรแกรมตัวอย่าง
- 3 ถ้าเลือกระบบงานแสดงผลการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



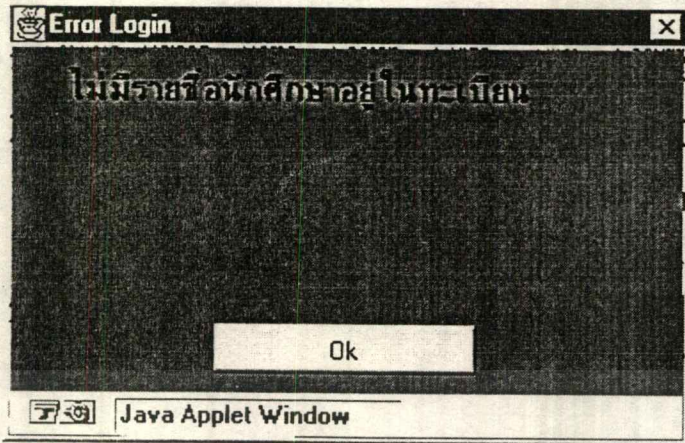
รูปที่ ๑-10 แสดงรายชื่อฐานข้อมูลที่จัดไว้

- 3.1 เลือกว่าจะดูจากระบบใดในที่นี่จะเลือกระบบ Access และเมื่อเลือกจะปรากฏหน้าต่างดังรูปเพื่อป้อน รหัสนักศึกษาและรหัสผ่าน



รูปที่ ๑-11 แสดงหน้าต่างของรหัสนักศึกษาและรหัสผ่าน

3.2 แต่ถ้าไม่มีหรือใส่ผิดจะแจ้งข้อผิดพลาดขึ้นดังรูป



รูปที่ ๑-12 แสดงหน้าต่างในกรณีที่ป้อนรหัสนักศึกษาหรือรหัสผ่านผิด

3.3 ถ้ารหัสนักศึกษาและรหัสผ่าน ผ่านจะแสดงผลการศึกษาของปีการศึกษาล่าสุด

ระบบแสดงผลการเรียนผ่านเครือข่ายอินเทอร์เน็ต

เลขทะเบียนนักศึกษา ปี เทอม

ชื่อ นามสกุล

รหัสวิชา	หน่วยกิต	ผลการเรียน	รายชื้อวิชา
01191106	3	C	Computer Organization and Architecture
03100002	2	B	Industrial Economics
01193114	3	D+	Management Information System
01193101	3	A	Project I
01193118	3	B	Robotic Engineering
01193124	3	B+	Select Topic in Information Technology

เทอมปัจจุบัน: หน่วยกิตที่ผ่าน: เกรด: สถานะ:

สะสม: GPA:

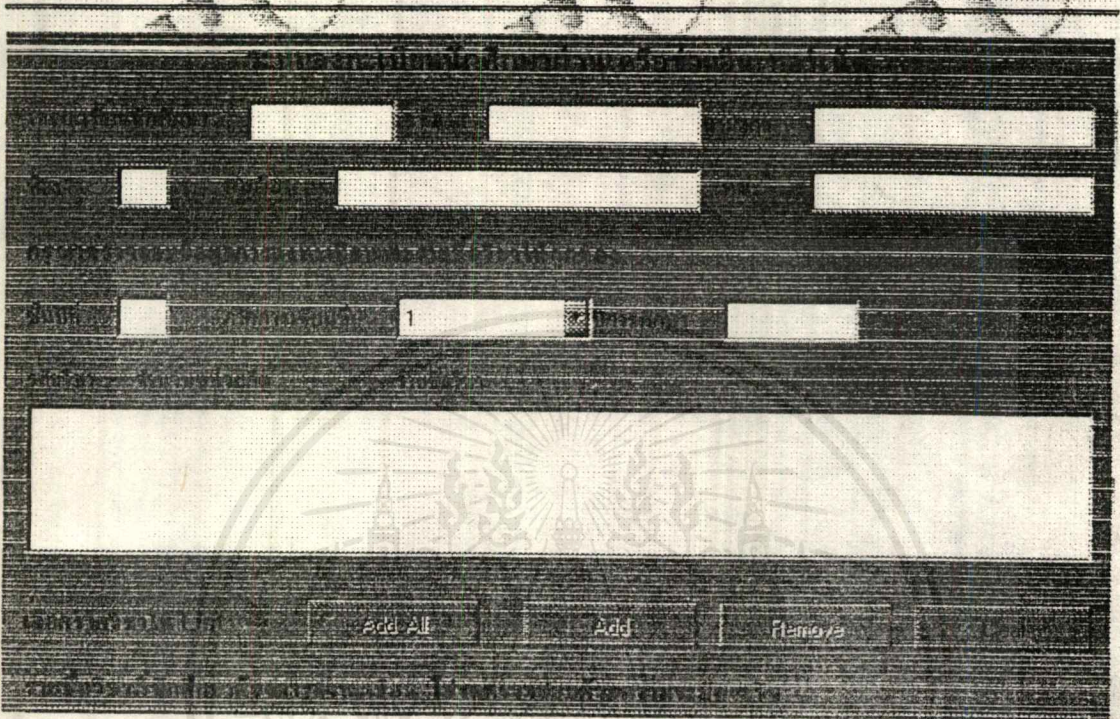
รูปที่ ๑-13 แสดงผลการศึกษาของปีล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

4 ถ้าเลือกระบบงานลงทะเบียนการศึกษาผ่านเครือข่ายอินเทอร์เน็ต

4.1 เมื่อเลือกแล้วจะแสดงหน้าต่างดังนี้

โปรแกรมตัวอย่างการใช้ Sybase SQL Anywhere 5.0 เป็น DBMS



รูปที่ ๑-14 แสดงหน้าต่างของแบบฟอร์มในการลงทะเบียน

4.2 ทำการ ป้อนรหัสนักศึกษาลงในกรอบเลขทะเบียนนักศึกษา จะแสดงชื่อ นามสกุล
ชั้น ภาควิชา และคณะ ในที่นี้จะระบุปีการศึกษาไว้ที่ปีปัจจุบัน

โปรแกรมตัวอย่างการใช้ Sybase SQL Anywhere 5.0 เป็น DBMS

รูปที่ ๑-15 แสดงตัวอย่างการป้อนข้อมูล

4.3 ทำการกำหนดชั้นปีที่ของวิชาว่าจะเลือกลงปีไหน และภาคการศึกษาใด ดังรูป

01192102	2	Computer Project II
01192103	2	Computer Laboratory III
01192106	3	Computer Network
01192107	3	Operating System I
01192108	3	Compiler Construction
01192109	2	Data Base Systems

รูปที่ ๑-16 แสดงรายวิชาเมื่อป้อนชั้นปีที่ และเทอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจะแสดงรายละเอียดของรายวิชาในชั้นปีและในภาคการศึกษานั้นๆ

- 4.4 ทำการเลือกรายวิชาที่ต้องการลงทะเบียนโดยการกด Add เมื่อต้องการเลือกทีละวิชา หรือ กด Add All ในกรณีที่ต้องการเลือกทั้งหมด และ จะทำการหาผลรวมของหน่วยกิตรวมทั้งคำนวณหาค่าหน่วยกิตด้วย ดังรูป

2	2	1998																		
<table border="1"> <tr> <td>01192102</td> <td>2</td> <td>Computer Project II</td> </tr> <tr> <td>01192103</td> <td>2</td> <td>Computer Laboratory III</td> </tr> <tr> <td>01192106</td> <td>3</td> <td>Computer Network</td> </tr> <tr> <td>01192107</td> <td>3</td> <td>Operating System I</td> </tr> <tr> <td>01192108</td> <td>3</td> <td>Compiler Construction</td> </tr> <tr> <td>01192109</td> <td>3</td> <td>Data Base Systems</td> </tr> </table>			01192102	2	Computer Project II	01192103	2	Computer Laboratory III	01192106	3	Computer Network	01192107	3	Operating System I	01192108	3	Compiler Construction	01192109	3	Data Base Systems
01192102	2	Computer Project II																		
01192103	2	Computer Laboratory III																		
01192106	3	Computer Network																		
01192107	3	Operating System I																		
01192108	3	Compiler Construction																		
01192109	3	Data Base Systems																		
16	800																			

รูปที่ ๑-17 แสดงผลเมื่อมีการกด Add หรือ Add All

- 4.5 ถ้าต้องการลงทะเบียนในของชั้นปีอื่นก็ป้อนชั้นปีที่ต้องการ ในกรอบชั้นปีที่โดยที่ไม่สามารถที่จะเปลี่ยนภาคการศึกษาได้ แต่ถ้าต้องการเปลี่ยนให้กด Clear นั้น แสดงว่าต้องการทำใหม่ ในที่นี้ต้องการลงทะเบียนของชั้นปีที่ 1 ใหม่ ดังรูป

1

2

1998

01001005	3	Engineering Mathematics II
01191104	3	Switch Theory
01191107	2	Computer Laboratory
01192104	3	Assembly and System Programming
03010004	2	Technical English II

01192102	2	Computer Project II
01192103	2	Computer Laboratory III
01192106	3	Computer Network
01192107	3	Operating System I
01192108	3	Compiler Construction
01192109	3	Data Base Systems
01191103	3	Data Structure and Algorithms

19

950

รูปที่ ๑-18 แสดงการลงทะเบียนของชั้นปีอื่น

- 4.6 เมื่อเลือกวิชาเสร็จเรียบร้อยแล้วจะต้อง Confirm เพื่อยืนยันสิ่งที่เลือก และเมื่อกด Confirm จะแสดงรายละเอียดต่างๆ ในการลงทะเบียนดังรูป

Detail Message

รายละเอียดเกี่ยวกับการลงทะเบียน

กรุณาโอนเงินค่าบำรุงการศึกษา จำนวน บาท

ผ่านบัญชีเลขที่

ธนาคาร สาขา

ตามรายการดังนี้

ค่าบำรุงการศึกษา	<input type="text" value="250"/>	บาท
ค่าลงทะเบียนวิชาเรียน	<input type="text" value="950"/>	บาท
ค่าบำรุงทั้ลงสมุด	<input type="text" value="100"/>	บาท
ค่าบริการสุขภาพ	<input type="text" value="50"/>	บาท
รวมเงิน	<input type="text" value="1350"/>	บาท

สำนักทะเบียนได้รับการลงทะเบียนเมื่อวันที่ เวลา

การลงทะเบียนจะสมบูรณ์ก็ต่อเมื่อนักทะเบียนฯ ได้รับเงินเรียบร้อยแล้ว

OK

Java Applet Window

รูปที่ ๑-19 แสดงรายละเอียดของการลงทะเบียน

4.7 และถ้ารายวิชานั้นได้ทำการลงทะเบียนไปแล้วจะแจ้งเตือนขึ้น

Error

การลงทะเบียนไม่สมบูรณ์
โปรดตรวจสอบข้อมูลก่อนการลงทะเบียน

Ok

Java Applet Window

รูปที่ ๑-20 แสดงหน้าต่างแจ้งเตือนการลงทะเบียนที่ไม่สมบูรณ์

กิตติกรรมประกาศ

โครงการนี้สามารถสำเร็จลุล่วงไปได้ด้วยดีนั้นก็เพราะได้รับความช่วยเหลือ และการสนับสนุนจาก
บุคคลหลายๆ ท่าน

ขอขอบพระคุณ บิดา-มารดา ที่ได้อบรมสั่งสอน ให้การเลี้ยงดูลูกมาอย่างดีที่สุดและให้ความสนับสนุนใน
ทุกๆ เรื่อง

ขอขอบพระคุณ ครูบาอาจารย์ทุกท่าน ที่ได้อบรมสั่งสอนทั้งด้านวิชาการและจริยธรรม

ขอขอบพระคุณ ดร.วราวัฒน์ ลิ้มโกศา อาจารย์ที่ปรึกษา ที่ได้ให้คำชี้แนะ คำแนะนำต่างๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

- [1] Cornell Gary And Horstmann Cay S.: "Core Java", Prentice Hall PTR, California, 1997
- [2] Hobbs Ashton : "Database Programing with JDBC", Sams net Publishing, Indiana, USA, 1997
- [3] Fronckowiak John W. And McMillan Gordon B., "Database Programming with Visual J++", Sams net Publishing, Indiana, USA, 1997
- [4] Sood Mukul : "Examining JDBC Drivers", Dr. Dobb's Journal, January, page 82-87, 1998
- [5] สุกฤษณ์ สุ่มมาตย์ : "Java เชื่อมต่อฐานข้อมูลด้วย JDBC ", อินเทอร์เน็ต-อินทราเน็ต ปีที่ 2 ฉบับที่ 10, หน้าที่ 79-84, บริษัท อินทรา เน็ตคอม จำกัด, กรุงเทพฯ, 2541

