

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ฐานข้อมูลบนเว็ลด์ไวด์เว็บ

DATABASE ON WORLD WIDE WEB



นางสาวสายสุดา ธนาธรรมนันท์
นางสาวสิริพร ตั้งพัฒนากิจเจริญ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลบนเว็ลด์ไวด์เว็บ
DATABASE ON WORLD WIDE WEB



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาคามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2542

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ฐานข้อมูลบนเว็ลด์ไวด์เว็บ

DATABASE ON WORLD WIDE WEB

ผู้จัดทำ

1.นางสาวสายสุดา ชนาธรรมนันท์ รหัสประจำตัว 39014568

2.นางสาวสิริพร ตั้งพัฒนากิจเจริญ รหัสประจำตัว 39014576



อาจารย์ที่ปรึกษา

(อาจารย์บัณฑิต พัสยา)

20/03/43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลบนเว็ลด์ไวด์เว็บ

สายสุคา หนาธรรมเนียม 39014568

สิริพร คังพัฒนากิจเจริญ 39014576

อาจารย์บัณฑิต พัทธา อาจารย์ที่ปรึกษา

ปีการศึกษา 2542

บทคัดย่อ

โครงการฐานข้อมูลบนเว็ลด์ไวด์เว็บจัดทำขึ้นเพื่อให้เป็นเว็บไซต์ตั้งชื่ออะไหล่รถยนต์ โดยรวบรวมข้อมูลของอะไหล่รถยนต์รุ่นต่างๆ เก็บไว้ในฐานข้อมูล เพื่อให้ผู้ใช้สามารถทำการเลือกซื้อและตั้งชื่ออะไหล่รถยนต์ผ่านบริการทางเว็ลด์ไวด์เว็บได้ โดยผู้ใช้สามารถค้นหาอะไหล่รถยนต์ที่ต้องการได้จากชื่ออะไหล่รถยนต์ และจากรุ่นของรถยนต์ที่ใช้กับอะไหล่ นั้น เมื่อผู้ใช้ยืนยันการตั้งชื่ออะไหล่แล้ว จะมีการออกไปกำกับสินค้าให้กับลูกค้า และแสดงรายละเอียดของใบกำกับสินค้าออกมาที่เว็บเพจ

โครงการนี้แบ่งเป็น 2 ส่วนหลักๆ คือ เว็บเซิร์ฟเวอร์และ คาตาเบสเซิร์ฟเวอร์ เว็บเซิร์ฟเวอร์จะเป็นส่วนที่ให้บริการทางเว็บเพื่อรองรับความต้องการของผู้ใช้ ส่วนคาตาเบสเซิร์ฟเวอร์นั้นจะเป็นระบบจัดการฐานข้อมูล ในโครงการนี้ใช้แอปเปิ้ล-เจเซิร์ฟเป็นเว็บเซิร์ฟเวอร์ และออราเคิลเป็นคาตาเบสเซิร์ฟเวอร์ การพัฒนาแอปพลิเคชันทำโดยเขียนโปรแกรมจาวาเซิร์ฟเล็ต ซึ่งเป็นส่วนที่เชื่อมโยงเว็บเซิร์ฟเวอร์ให้สามารถค้นหาและจัดการข้อมูลต่างๆ ที่เก็บอยู่ในฐานข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Database on World Wide Web

Saisuda Thanathammanunt

Siriporn Tangpattanakitcharoen

Bundit Pasaya Advisor

ABSTRACT

Database on World Wide Web Project is arranged for sale spare parts web site. This project collects data of spare parts into database for any users through World Wide Web. The users could find spare parts by searching from their names and their manufacturers model of the cars that are compatible with these spare parts. After users confirm to buy, the invoice will be shown on web page.

The main parts of this project can be divided into two parts which are Web Server and Database Server. Web server will serve web page which user's request. Database server is the Database Management System which will manage all data. This project uses Apache-Jserv as Web Server and Oracle as Database Server. The application development will be written in Java Servlet that enable web server to search and manage data in the database.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณบุคคลต่อไปนี้ที่ให้ความช่วยเหลือในการทำโครงการให้สำเร็จไปได้ด้วยดี

อาจารย์บัณฑิต พัสชา	อาจารย์ที่ปรึกษาที่ให้คำแนะนำ เอาใจใส่ดูแล และช่วยเหลือเราเสมอมา
คุณฟูเกียรติ จุลนวล	พี่จากบริษัท Oracle ที่ให้คำแนะนำเกี่ยวกับแนวทางในการติดตั้งระบบ และความรู้ต่างๆ อีกมากมายเกี่ยวกับออราเคิล นอกจากนี้ยังให้ยืมซีดี Oracle บนลินุกซ์ไปติดตั้ง
เฮียไข่	ที่ให้คำปรึกษาค้นแนวทางในการทำโครงการ และคอยช่วยเหลือในเรื่องต่างๆ
พี่เต๋า	ที่ช่วยคิดค้นหาเอกสารเกี่ยวกับออราเคิล และให้คำปรึกษาเกี่ยวกับการเขียนโปรแกรมภาษาจาวา
พี่หนุ่ม	กับคำปรึกษาเกี่ยวกับ Oracle บนลินุกซ์ พร้อมทั้งแนะนำให้รู้จักกับพี่กรและพี่ฟูเกียรติ
พี่กร	ที่ให้ยืมซีดี Oracle บนลินุกซ์
พี่ไอ้	ที่ให้ยืมหนังสือคู่มือติดตั้งและปรับแต่งออราเคิล
พี่จ็อกกี้	ที่แนะนำแนวทางเกี่ยวกับการเลือกทำโครงการ และให้คำปรึกษาทางเทคนิค
พี่ก้อง & พี่เซวี่ & น้องกระบี่	ที่ให้ความช่วยเหลือเมื่อเครื่องมีปัญหา และให้ยืมไขควง ซีดี เทปขาว ฯลฯ จากห้อง ISAG เสมอ
เอ	ที่ให้คำปรึกษาเกี่ยวกับการเขียนจาวาเซิร์ฟเล็ต
ฮาร์ท	ที่ให้ข้อมูลเกี่ยวกับอะไหล่รถยนต์ และความรู้เกี่ยวกับรถยนต์
ปิ่น	ที่ช่วยดูเครื่อง เวลาระบบปฏิบัติการลินุกซ์มีปัญหา
ลม	ที่ให้คำปรึกษาเกี่ยวกับลินุกซ์
หลิว	เพื่อนที่แสนดี คอยให้กำลังใจและให้ที่พักพิงตอนทำโครงการ
คีน้อย	ที่ให้เราหยิบหนังสือลินุกซ์ไปใช้
พี่ดม	ที่ให้ share drive เพื่อเก็บไฟล์
เพื่อนๆ พี่ๆ ห้อง Database	ที่ให้กำลังใจ และความรู้ดีๆ ที่ได้ทำงานร่วมกันตลอดปี
ภาควิชาวิศวกรรมคอมพิวเตอร์	ที่ให้สถานที่ในการทำโครงการ

และบุคคลสำคัญที่ต้องขอขอบคุณ คือ คุณพ่อ คุณแม่ พี่น้อง และเพื่อนๆ ทุกคนที่ให้ความรัก เอาใจใส่และห่วงใยเราเสมอมาในทุกๆ เรื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VII
สารบัญตาราง	IX
✓บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ลิנקซ์	3
2.1 ความหมายของลิנקซ์	3
2.2 ประวัติของลิנקซ์	4
2.3 จุดเด่นของลิנקซ์	4
2.4 셸ล์ (Shell)	5
2.5 โครงสร้างของระบบไฟล์และไครกทอรี	6
2.6 การใช้งานลิנקซ์เบื้องต้น	6
2.7 การดูแลรักษาระบบเบื้องต้น	10
2.8 การใช้งานภาษาไทยบนลิנקซ์	13
2.8.1 การติดตั้งภาษาไทยบนลิנקซ์	13
2.8.2 การปรับแต่งให้เน็ตสเคปแสดงผลภาษาไทยได้	13
✓บทที่ 3 ระบบจัดการฐานข้อมูล	14
3.1 ความหมายและที่มา	14
3.2 สถาปัตยกรรมภายในของออราเคิล	15
3.3 โครงสร้างภายใน (Internal Structure)	17
3.3.1 ไฟล์ฐานข้อมูล	17
3.3.2 โครงสร้างของหน่วยความจำ (Memory Structures)	18
3.3.3 โพรเซสของออราเคิล	18
3.4 ตาราง (Table)	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 SQL (Structure Query Language)	20
3.5.1 ไวยากรณ์ของภาษา SQL	21
3.5.2 รูปแบบการใช้คำสั่ง SQL	21
3.6 ทรานแซคชัน (Transaction)	24
3.7 ระบบรักษาความปลอดภัย	24
3.7.1 มาตรการรักษาความปลอดภัย (Security Policy)	24
3.7.2 การให้สิทธิ์และการอนุญาตการใช้งานอ็อบเจ็กต์	25
3.8 การเข้าถึงผ่านเครือข่าย (Network Access)	26
3.9 การใช้งานฐานข้อมูลออรากเคิล	27
3.10 การออกแบบฐานข้อมูลด้วยวิธีโนแอม	28
๗ บทที่ 4 เวิลด์ไวด์เว็บ (World Wide Web)	30
4.1 เทคโนโลยีเครือข่ายอินเทอร์เน็ต	30
4.1.1 โครงสร้างของเครือข่าย	30
4.1.2 ความรู้เบื้องต้นเกี่ยวกับ TCP/IP	33
4.2 เว็บเซิร์ฟเวอร์	36
4.2.1 โครงสร้างของเอกสารเฮชทีเอ็มแอล (HTML)	37
4.2.2 ซีจีไอ (Common Gateway Interface-CGI)	37
4.3 อพาเช่เว็บเซิร์ฟเวอร์ (Apache Web Server)	38
4.4 อพาเช่เจเซิร์ฟ (ApacheJServ)	39
4.4.1 ลักษณะของอพาเช่เจเซิร์ฟ	41
4.4.2 คำศัพท์ที่เกี่ยวข้องกับอพาเช่เจเซิร์ฟ	41
4.4.3 หลักการทำงานของเซิร์ฟเล็ตเอ็นจิน (Servlet Engine)	42
4.4.4 ระบบรักษาความปลอดภัยของอพาเช่เจเซิร์ฟ	43
๘ บทที่ 5 จาวา (JAVA)	44
5.1 ความสามารถของภาษาจาวา	44
5.2 การนำจาวาไปประยุกต์สร้างและพัฒนาในงานด้านต่างๆ	44
5.3 หลักการทำงานของภาษาจาวา	45
5.3.1 การคอมไพล์	45
5.3.2 การวางตำแหน่งในหน่วยความจำ	46
5.3.3 การรันโปรแกรม	47
5.3.4 คลาสโหลดเดอร์ (Class Loader)	47
5.3.5 ตรวจสอบไบท์โค้ด	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.6 การทำงานตามโค้ด	48
5.4 การสร้างและรันภาษาจาวา	49
5.5 JDBC (Java Database Connectivity)	52
5.5.1 JDBC/ODBC bridge	52
5.5.2 Native-API, partly Java driver	53
5.5.3 Network-protocol, all-Java driver	54
5.5.4 Native-protocol, all-Java driver	55
5.6 จาวาเซิร์ฟเล็ต	56
5.6.1 ความสามารถของเซิร์ฟเล็ต	57
5.6.2 พื้นฐานของ Http Servlet	57
5.6.3 วงจรชีวิตของเซิร์ฟเล็ต	58
5.6.4 GET และ POST	59
5.6.5 การเชื่อมต่อกับฐานข้อมูล	63
✓ บทที่ 6 หลักการออกแบบ	65
6.1 การออกแบบฐานข้อมูล	65
6.1.1 ไนแอม (NIAM –Nijssens 's Information Analysis Methodology) ของฐานข้อมูล	65
6.1.2 ตารางฐานข้อมูลในโครงการ	67
6.1.3 คำอธิบายตารางฐานข้อมูล (Data Dictionary)	67
6.2 การออกแบบเว็บเพจ	69
6.2.1 แผนภาพแสดงการเชื่อมต่อระหว่างเว็บเพจ	69
6.2.2 เว็บเพจในโครงการ	73
6.3 ปัญหาที่พบและการแก้ไข	78
บรรณานุกรม	81
ภาคผนวก	
ก. การติดตั้งระบบปฏิบัติการลินุกซ์	82
ข. การติดตั้งอแปเช่-เจเซิร์ฟ	83
ค. การติดตั้งออราเคิล	86
ง. Source Code	91

สารบัญรูปภาพ

หน้า

รูปที่ 2.1	เซลล์บนระบบฐานข้อมูลและลินุกซ์	5
รูปที่ 2.2	โครงสร้างของไครกทอรีในลินุกซ์ (บางส่วน)	6
รูปที่ 2.3	รายละเอียดของสิทธิ์ในการเข้าถึงแฟ้มข้อมูล	9
รูปที่ 3.1	ฐานข้อมูลและระบบจัดการฐานข้อมูล	14
รูปที่ 3.2	องค์ประกอบทั้ง 4 ส่วน และวิธีเชื่อมต่อกัน	15
รูปที่ 3.3	สถาปัตยกรรมของ Oracle Workgroup Server	17
รูปที่ 3.4	System Global Area	18
รูปที่ 3.5	ตัวอย่างของตารางชื่อ EMPLOYEES	19
รูปที่ 3.6	ระบบรักษาความปลอดภัยที่เกี่ยวข้องกับผู้ใช้ (user)	25
รูปที่ 3.7	ลักษณะการติดต่อผ่านระบบเครือข่ายของออราเคิล	27
รูปที่ 3.8	สัญลักษณ์ของชนิดเอนติตีภาควิชา	28
รูปที่ 3.9	สัญลักษณ์ของชนิดเลเวลรหัสภาควิชา	28
รูปที่ 3.10	ความสัมพันธ์อ้างอิงแบบ one to one	28
รูปที่ 3.11	ความสัมพันธ์อ้างอิงแบบ one to one อย่างย่อ	29
รูปที่ 3.12	ความสัมพันธ์อ้างอิงแบบ many to one	29
รูปที่ 3.13	ความสัมพันธ์แบบอ้างอิงแบบ many to many	29
รูปที่ 4.1	การเชื่อมต่อแบบจุดถึงจุด	30
รูปที่ 4.2	การเชื่อมต่อแบบกระจาย	31
รูปที่ 4.3	OSI 7 Layer	33
รูปที่ 4.4	การให้บริการเว็บไซต์ไคว์เว็บ	36
รูปที่ 4.5	การทำงานของเซิร์ฟเวอร์ที่อยู่ในอาปาเช่-เจเซิร์ฟ	40
รูปที่ 4.6	เว็บเซิร์ฟเวอร์ที่มีเซิร์ฟเวอร์เอนจินซึ่งจัดการกับการร้องขอของไคลเอนท์	40
รูปที่ 5.1	การประมวลผลของ โปรแกรมทั่วไปแต่ระบบปฏิบัติการที่ต่างกัน	46
รูปที่ 5.2	การประมวลผลของ โปรแกรมภาษาจาวา	46
รูปที่ 5.3	เอกสาร HTML ที่มีการเรียกแอปพลิเคชันมาจากเน็ตสเค๊ป	51
รูปที่ 5.4	เอกสาร HTML ที่มีการเรียกแอปพลิเคชันมาจาก Appletviewer	51
รูปที่ 5.5	การติดต่อ JDBC โคจรูปแบบที่1 JDBC-ODBC Bridge	52
รูปที่ 5.6	การติดต่อ JDBC โคจรูปแบบที่2 Native-API, partly Java driver	53
รูปที่ 5.7	การติดต่อ JDBC โคจรูปแบบที่3 Network-protocol, all-Java driver	54
รูปที่ 5.8	การติดต่อ JDBC โคจรูปแบบที่4 Native-protocol, all-Java driver	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.9	การทำงานของจาวาเซิร์ฟเอร์แบบมัลติเซรค	56
รูปที่ 5.10	เซรคใช้งานเซิร์ฟเอร์ได้จากเซิร์ฟเอร์พูล	59
รูปที่ 5.11	เว็บเพจแสดงฟอร์มรับข้อมูล	61
รูปที่ 5.12	เว็บเพจโต้ตอบกับผู้ใช้หลังจากกรอกชื่อ	62
รูปที่ 5.13	หน้าจอเว็บเพจแสดงข้อมูลในฐานข้อมูลเมื่อเรียกคลาส DBPhoneLookup	64
รูปที่ 6.1	ในแอมไคอะแกรมของฐานข้อมูล	66
รูปที่ 6.2	ขั้นตอนการสั่งซื้ออะไหล่	70
รูปที่ 6.3	การเชื่อมต่อระหว่างเว็บเพจ	71
รูปที่ 6.3	(ต่อ)การเชื่อมต่อระหว่างเว็บเพจ	72
รูปที่ 6.4	เว็บเพจแสดงชั้นรูปภาพเพื่อให้เลือกชิ้นส่วนของอะไหล่รถ	73
รูปที่ 6.5	เว็บเพจแสดงรายการของอะไหล่ชิ้นย่อยๆให้เราเลือก	73
รูปที่ 6.6	เว็บเพจเลือกชนิดและรุ่นของรถที่ใช้กับอะไหล่ที่เราต้องการ	74
รูปที่ 6.7	เว็บเพจเลือกชนิดและรุ่นของรถที่ใช้กับอะไหล่ที่เราต้องการ	74
รูปที่ 6.8	เว็บเพจแสดงรายการสินค้าที่ลูกค้าต้องการ	75
รูปที่ 6.9	เว็บเพจแสดงรายการสินค้าและจำนวนสินค้าที่เราเลือกซื้อ	75
รูปที่ 6.10	เว็บเพจแสดงหน้าจอนี้สำหรับลูกค้าทั่วไปกรอกรายละเอียด	76
รูปที่ 6.11	เว็บเพจแสดงหน้าจอลูกค้าที่เป็นสมาชิกกรอก username และ password	76
รูปที่ 6.12	เว็บเพจแสดงใบกำกับสินค้า	77
รูปที่ 6.13	เว็บเพจแสดงหน้าจอในกรณีที่ไม่มีอะไหล่ลูกค้าต้องการ	77

สารบัญตาราง

หน้า

ตารางที่ 2.1	เงื่อนไขที่ใช้ในการตรวจหาไฟล์	12
ตารางที่ 2.2	คำสั่งที่กระทำเมื่อค้นหาพบ	12
ตารางที่ 3.1	โปรแกรมลูกข่ายที่สำคัญของออราเคิล	16
ตารางที่ 3.2	โปรแกรมเครือข่ายที่สำคัญของออราเคิล	16
ตารางที่ 3.3	เครื่องหมายในภาษา (SQL Operator)	22
ตารางที่ 4.1	ขอบเขตหมายเลขแอคเคสของเน็ตเวิร์กในแต่ละคลาส	35
ตารางที่ 6.1	ตารางแสดงรหัสของระบบที่ได้จากการแปลงในเอมโคโนแกรม	67
ตารางที่ 6.2	Data Dictionary ของตาราง INVOICE	67
ตารางที่ 6.3	Data Dictionary ของตาราง SALE	67
ตารางที่ 6.4	Data Dictionary ของตาราง CUSTOMER	68
ตารางที่ 6.5	Data Dictionary ของตาราง CAR	68
ตารางที่ 6.6	Data Dictionary ของตาราง MATCH	68
ตารางที่ 6.7	Data Dictionary ของตาราง SPARE_PART	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

ในปัจจุบันนี้การใช้งานระบบเครือข่ายอินเทอร์เน็ต (internet) ได้รับความนิยมน้อยอย่างแพร่หลายภายในระยะเวลาอันรวดเร็ว สาเหตุเนื่องมาจากระบบเครือข่ายอินเทอร์เน็ตกลายเป็นแหล่งข้อมูลขนาดใหญ่สำหรับผู้เข้ามาใช้บริการระบบเครือข่าย ทั้งนี้เพราะแต่ละไซต์จัดตั้งขึ้นเพื่อให้บริการข้อมูล ตามจุดประสงค์ที่แตกต่างกัน โดยอาจจะเป็นแหล่งที่ให้ข้อมูลเกี่ยวกับ วิชาการ กีฬา ความบันเทิง การเมือง เศรษฐกิจ รวมถึงการทำธุรกรรมที่เกิดขึ้นมากมาย เนื่องจากอำนวยความสะดวกให้กับผู้ซื้อ เพราะสามารถทำการสั่งซื้อของผ่านอินเทอร์เน็ตได้โดยไม่ต้องเดินทางไปถึงร้านขายสินค้า

โครงการนี้เป็นเว็บไซต์หนึ่งที่ทำการซื้อขายอะไหล่รถยนต์ ซึ่งลูกค้าสามารถค้นหาอะไหล่ที่ต้องการ โดยอาจจะหาคำยี่ห้อ รุ่น หรือปีของรถที่ใช้อะไหล่ชิ้นนั้นได้ หรือสามารถเลือกกรายการอะไหล่ตามรายชื่อที่ต้องการได้ โดยค้นหาเทคโนโลยีจาซึ่งเป็นเทคโนโลยีใหม่มาใช้ระบบของโครงการนี้ ซึ่งประกอบด้วยจาวาเซิร์ฟเล็ต (Java Servlet) ซึ่งเป็นการนำเสนอเว็บเพจแทนที่ใช้ซีจีไอ (CGI) การเชื่อมต่อฐานข้อมูลด้วย JDBC โดยใช้ภาษา SQL ที่ฝังไว้ในภาษาจาวา

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อนำมาเป็นเว็บไซต์ที่มีการเลือกซื้อและสั่งซื้อสินค้าบนอินเทอร์เน็ต
2. เพื่อเพื่อศึกษาเทคโนโลยีในการพัฒนาเว็บเพจ
3. เพื่อศึกษาระบบจัดการฐานข้อมูลและการติดต่อฐานข้อมูลผ่านทางเว็บเพจ
4. เพื่ออำนวยความสะดวกแก่ผู้ซื้ออะไหล่รถยนต์ในการสั่งซื้อ และผู้ขายอะไหล่ในเรื่องของการเพิ่มช่องทางการขายสินค้า

1.3 ขอบเขตของโครงการ

1. เป็นเว็บไซต์สั่งซื้ออะไหล่ โดยสามารถค้นหาอะไหล่ที่ต้องการได้อย่างง่ายดาย และมีขั้นตอนการสั่งซื้อที่ไม่ยุ่งยาก
2. สามารถค้นหาอะไหล่จาก ชื่อของอะไหล่, ชนิดของรถที่เข้ากันได้กับอะไหล่
3. ลูกค้าที่สั่งซื้ออะไหล่มี 2 ประเภท คือ ลูกค้าทั่วไป และสมาชิก โดยสมาชิกจะได้รับส่วนลด 30%
4. เป็นเว็บไซต์สำหรับคนไทย เนื่องจากชื่ออะไหล่ที่เก็บในฐานข้อมูลเป็นภาษาไทย ยังไม่มีการเก็บชื่ออะไหล่เป็นภาษาอังกฤษ
5. รวบรวมฐานข้อมูลเกี่ยวกับอะไหล่รถ และฐานข้อมูลอื่นๆที่จำเป็นในการซื้อขายสินค้า
6. สามารถย้อนกลับไปแก้ไขรายการอะไหล่ที่เลือกซื้อแล้วไม่ต้องการได้
7. สามารถค้นหาอะไหล่จากรูปรถยนต์ได้ ซึ่งใช้เทคนิคการแมปภาพ (Image Map)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีการดำเนินงาน

โครงการนี้เริ่มต้นด้วยการศึกษาทฤษฎีพื้นฐานต่าง ๆ ที่เกี่ยวข้องกับโครงการนี้ คือการทำงานและสถาปัตยกรรมของเว็บเซิร์ฟเวอร์ (Web Server) เปรียบเทียบข้อดีข้อเสียของจาวา และ แอ็คทีฟ-เอ็กซ์ (Active X) เพื่อคัดเลือกใช้ภาษาที่นำมาพัฒนาแอปพลิเคชัน ซึ่งได้เลือกใช้ภาษาจาวา ซึ่งมีข้อดีหลายอย่าง เช่น การใช้งานที่ไม่ขึ้นกับแพลตฟอร์ม และศึกษาการสร้างเว็บเพจ (web page)

การคิดระบบจะประกอบด้วย 3 ส่วนหลักๆ คือ การติดตั้งลินุกซ์ (Linux) เป็นระบบปฏิบัติการ, การติดตั้งอพาเช่-เจเซิร์ฟ (Apache JServ) เป็นเว็บเซิร์ฟเวอร์ และการติดตั้งออราเคิล (Oracle) เป็นดาตาเบสเซิร์ฟเวอร์ ซึ่งเป็นการติดตั้งฐานข้อมูลออราเคิลบนลินุกซ์ พร้อมกับศึกษาวิธีการปรับแต่งและใช้งานให้ระบบสามารถทำงานได้ในสภาพแวดล้อมที่เหมาะสม ปลอดภัย

หลังจากที่ติดตั้งระบบเรียบร้อยแล้ว จึงเริ่มพัฒนาเว็บไซค์ขายอะไหล่รถยนต์ โดยผู้ใช้งานค้นหาและสั่งซื้ออะไหล่ที่ต้องการได้ การออกแบบฐานข้อมูลทำโดยใช้วิธีในแอม (NIAM – Nijssens 's Information Analysis Methodology) โดยมีภารกิจรวบรวมข้อมูลที่ทำเป็นไว้ในฐานข้อมูล

การพัฒนาแอปพลิเคชันทำโดยเขียนโปรแกรมจาวาเซิร์ฟได้ให้ทำงานอย่างสมบูรณ์ เช่น ติดต่อกับฐานข้อมูล การเขียนโปรแกรมทำไปพร้อมกับการตรวจสอบโปรแกรมเพื่อหาข้อผิดพลาด และแก้ไขเมื่อพบข้อผิดพลาด

บทที่ 2

ลินุกซ์

2.1 ความหมายของลินุกซ์

ลินุกซ์ (Linux) เป็นระบบปฏิบัติการแบบ 32 บิต ที่พัฒนาขึ้นมาจากระบบปฏิบัติการยูนิกซ์ สำหรับเครื่องพีซี และแจกจ่ายให้ใช้ฟรี สนับสนุนการใช้งานแบบหลายงานพร้อมๆกัน (Multi Tasking), หลายผู้ใช้พร้อมๆกัน (Multi User), มีระบบเอ็กซ์วินโดวส์ (X Windows) ซึ่งเป็นระบบการติดต่อผู้ใช้แบบกราฟฟิกที่ไม่ขึ้นกับโอเอสหรือฮาร์ดแวร์ใดๆ (ใช้กันมากในระบบยูนิกซ์) และมาตรฐานการสื่อสาร TCP/IP ที่ใช้เป็นมาตรฐานการสื่อสารในอินเทอร์เน็ตมาให้อยู่ในตัว นอกจากนี้ ยังสนับสนุนการทำงานแบบมัลติโพรเซสเซอร์ (Multi Processor) ด้วย

ลินุกซ์มีความเข้ากันได้ (compatible) กับมาตรฐาน POSIX ซึ่งเป็นมาตรฐานอินเทอร์เน็ตเฟซที่ระบบยูนิกซ์ส่วนใหญ่จะต้องมี และมีรูปแบบบางส่วนที่คล้ายกับระบบปฏิบัติการยูนิกซ์จากค่าย Berkeley และ System V โดยความหมายทางเทคนิคแล้ว ลินุกซ์เป็นเพียง “แก่น” หรือเคอร์เนล (kernel) ของระบบปฏิบัติการ ซึ่งจะทำหน้าที่ในด้านของการจัดสรรทรัพยากรในระบบ, การประมวลผล, การจัดการ ไฟล์และอุปกรณ์ I/O ต่างๆ แต่ผู้ใช้ทั่วไปจะรู้จักลินุกซ์ผ่านทางแอปพลิเคชันและระบบอินเทอร์เน็ตเฟซที่เขาเหล่านั้นเห็น (เช่น เอ็กซ์วินโดวส์ หรือ เซลล์)

ปัจจุบันลินุกซ์ยังได้ทำการพัฒนาระบบเพื่อให้สามารถใช้งานได้บนแพลตฟอร์มอื่นๆด้วย เช่น DEC Alpha, Motorola Power-PC, MIPS เมื่อคุณสร้างแอปพลิเคชันขึ้นมาบนแพลตฟอร์มใดแพลตฟอร์มหนึ่งแล้ว คุณก็สามารถย้ายแอปพลิเคชันของคุณ ไปวิ่งบนแพลตฟอร์มอื่นได้ไม่ยาก

บรรดาผู้ใช้งานบนลินุกซ์มีได้หลากหลาย ไม่ว่าจะเป็นระดับเคอร์เนลแฮกเกอร์ ซึ่งจะทำการศึกษายุ่งเกี่ยวกับการทำงานของระบบปฏิบัติการในระดับลึก ไปจนถึงเอนคีย์เซออร์หรือผู้ใช้ทั่วไป ผู้ใช้สามารถใช้ลินุกซ์ทำประโยชน์ได้หลายอย่าง ไม่ว่าจะเป็นเอาไว้ทำการศึกษาระบบยูนิกซ์ หรือศึกษาตัวอย่างการเขียนรหัสโปรแกรมที่ดีได้ หากต้องการจะใช้แอปพลิเคชันบนคอส หรือบนวินโดวส์

แอปพลิเคชันที่พัฒนามาเพื่อใช้งานบนลินุกซ์ที่น่าสนใจก็มีมากมาย ได้แก่ ตัวอย่างของเว็บเบราว์เซอร์ อะเรนา (Arena), เน็ตสเคป (Netscape) และ โมเสค (Mosaic) หรือเกมส์ต่างๆ เช่น DOOM เป็นต้น นอกจากนี้ ลินุกซ์ยังมีโปรแกรมประเภทออฟฟิศทูล (Office Tools) ที่ใช้ในการจัดพิมพ์งานด้านเอกสาร, และ โปรแกรมตกแต่งภาพที่คล้ายๆ กับ PhotoShop

แอปพลิเคชันที่กล่าวถึงข้างต้นนี้ ส่วนใหญ่จะเป็นแอปพลิเคชันที่แจกจ่ายฟรี ผ่านทางอินเทอร์เน็ต แต่ในปัจจุบันสำหรับลินุกซ์แล้วก็เริ่มที่จะมีตลาดของตัวเองมากขึ้นเรื่อยๆ ทำให้มีบริษัทต่างๆ ได้เริ่มทำการพัฒนาแอปพลิเคชันที่เป็นเชิงการค้า แอปพลิเคชันเหล่านี้มีมากมาย และผู้พัฒนาก็มีทั้งในยุโรปและอเมริกา ตัวอย่างเช่น คาตาเบสเซิร์ฟเวอร์ YardSQL, JustLogic SQL, MySQL, NEXUS และเวิร์ดโพรเซสเซอร์ หรือในปัจจุบันบริษัทผู้ผลิตต่างๆ ไม่ว่าจะเป็น IBM, Oracle หรือ Informix ต่างก็มีผลิตภัณฑ์ของตนซึ่งได้พัฒนาให้มาใช้กับลินุกซ์เป็นที่เรียบร้อยแล้วเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้สามารถจะสื่อสารกับอินเทอร์เน็ต, ทำบีบีเอสส่วนตัว, ทำระบบงานแบ็คออฟฟิศที่ใช้งานจริง, ใช้ทำการศึกษา หรือแม้แต่ใช้เป็นอินเทอร์เน็ตเซิร์ฟเวอร์ (Internet Server), เว็บเซิร์ฟเวอร์ (Web Server) ก็ได้

2.2 ประวัติของลินุกซ์

ลินุกซ์ถือกำเนิดขึ้นในฟินแลนด์ ปี ค.ศ. 1980 โดยลินุส โทรวาลด์ส (Linus Trovalds) นักศึกษาภาควิชาวิทยาการคอมพิวเตอร์ (Computer Science) ในมหาวิทยาลัยเฮลซิงกิ

ลินุสเห็นว่าระบบมินิกซ์ (Minix) ที่เป็นระบบยูนิกซ์บนพีซีในขณะนั้น ซึ่งทำการพัฒนาโดย ศ.แอนดรูว์ ทานเนนบาวม (Andrew S. Tanenbaum) ยังมีความสามารถไม่เพียงพอแก่ความต้องการ จึงได้เริ่มต้นทำการพัฒนาระบบยูนิกซ์ของตนเองขึ้นมา โดยจุดประสงค์อีกประการ ก็คือต้องการทำความเข้าใจในวิจาาระบบปฏิบัติการคอมพิวเตอร์ เมื่อเขาเริ่มพัฒนาลินุกซ์ไปช่วงหนึ่งแล้ว เขาก็ได้ทำการชักชวนให้นักพัฒนาโปรแกรมอื่นๆ มาช่วยทำการพัฒนาลินุกซ์ ซึ่งความร่วมมือส่วนใหญ่ก็จะเป็นความร่วมมือผ่านทางอินเทอร์เน็ต

ลินุสจะเป็นคนรวบรวมโปรแกรมที่ผู้พัฒนาต่างๆ ได้ร่วมกันทำการพัฒนาขึ้นมาและแจกจ่ายให้ทดลองใช้เพื่อทดสอบหาข้อบกพร่อง ที่น่าสนใจก็คืองานต่างๆ เหล่านี้ผู้คนที่หมดต่างก็ทำงานโดยไม่คิดค่าตอบแทน และทำงานผ่านอินเทอร์เน็ตทั้งหมด

ปัจจุบันเวอร์ชันล่าสุดของระบบลินุกซ์ที่ได้ประกาศออกมาคือเวอร์ชัน 2.0.13 ข้อสังเกตในเรื่องเลขรหัสเวอร์ชันนี้ก็คือ ถ้ารหัสเวอร์ชันหลังทศนิยมตัวแรกเป็นเลขคู่เช่น 1.0.x, 1.2.x เวอร์ชันเหล่านี้จะถือว่าเป็นเวอร์ชันที่เสถียรแล้ว และมีความมั่นคงในระดับหนึ่ง แต่ถ้าเป็นเลขคี่เช่น 1.1.x, 1.3.x จะถือว่าเป็นเวอร์ชันทดสอบ ซึ่งในเวอร์ชันเหล่านี้จะมีการเพิ่มเติมความสามารถใหม่ๆ ลงไป และยังคงทำการทดสอบหาข้อผิดพลาดต่างๆ อยู่

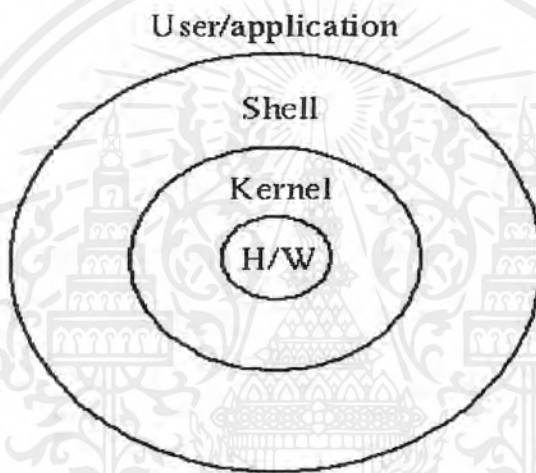
2.3 จุดเด่นของลินุกซ์

- ลินุกซ์ เป็นระบบปฏิบัติการที่ใช้งานได้ฟรี
- ลินุกซ์ ทำงานได้บนเครื่องพีซีทั่วไป
- ความสามารถแบบยูนิกซ์
- การใช้งานร่วมกับดอส (DOS) และ ไมโครซอฟท์วินโดวส์ (Microsoft Windows)
- ลินุกซ์ เป็นระบบปฏิบัติการแบบเปิด
- สนับสนุนการใช้งานแบบหลายงานพร้อมๆ กัน (Multi Tasking), หลายผู้ใช้พร้อมๆ กัน (Multi User) ที่สมบูรณ์แบบ
- ประสิทธิภาพ ลินุกซ์ถูกออกแบบมาให้ใช้งานอุปกรณ์ฮาร์ดแวร์ของเครื่องอย่างเต็มประสิทธิภาพ
- คุณภาพ ลินุกซ์เป็นความร่วมมือจากนักพัฒนานับพันที่ติดต่อกันทางอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 เชลล์ (Shell)

เนื่องจากลินุกซ์ถูกพัฒนาขึ้นมาโดยอาศัยรูปแบบส่วนใหญ่มาจากยูนิกซ์ และผู้พัฒนาระยะแรกๆ ก็เป็นผู้เชี่ยวชาญยูนิกซ์เป็นส่วนใหญ่ ดังนั้นหากเรียนรู้อินิกซ์เราก็จะรู้ลินุกซ์ด้วยเช่นเดียวกัน เมื่อเราทำการล็อกอินเข้ามาใช้งานในระบบ เราสามารถจะติดต่อกับเครื่องคอมพิวเตอร์โดยผ่านโปรแกรมที่เรียกว่า "เชลล์" โดยที่เชลล์จะทำหน้าที่ตีความคำสั่งที่ได้รับจากผู้ใช้งานส่งต่อไปยังคอมพิวเตอร์ และรับผลลัพธ์จากคอมพิวเตอร์มาแสดงให้กับผู้ใช้อีกทีหนึ่ง นั่นคือเชลล์จะทำหน้าที่เป็นตัวกลางคอยประสานงานการใช้งานระหว่างผู้ใช้และคอมพิวเตอร์ ในคอสมจะมีโปรแกรมอยู่ตัวหนึ่งที่ทำหน้าที่คล้ายกับเชลล์ ก็คือโปรแกรม COMMAND.COM แต่มีความซับซ้อนน้อยกว่าเชลล์ในยูนิกซ์มาก เราสามารถจะเขียนโปรแกรมให้ใช้งานกับเชลล์ได้ ซึ่งจะเรียกว่า เชลล์สคริปต์ (shell script) จะคล้ายกับแบทไฟล์ (Batch file) บนคอสมแต่มีประสิทธิภาพสูงกว่า



รูปที่ 2.1 เชลล์บนระบบยูนิกซ์และลินุกซ์

เชลล์บนยูนิกซ์จะมีที่นิยมใช้กันอยู่สองแบบคือ บอร์นเชลล์ (Bourne shells) ซึ่งถูกเรียกตามผู้คิดค้นแรกก็คือ Steven Bourne และเชลล์อีกแบบก็คือ ซีเชลล์ (C shells) โดยผู้เขียนคนแรก ก็คือ Bill Joy (เป็นคนเขียน vi บนยูนิกซ์ด้วยเช่นกัน) ปัจจุบัน Bill Joy ทำงานให้กับบริษัทซันไมโครซิสเต็ม (เขาเป็นผู้ร่วมก่อตั้งบริษัทคนหนึ่งด้วย)

สำหรับเชลล์ที่นิยมใช้งานกันอยู่บนลินุกซ์ก็คือ แบช (bash หรือ Bourne Again Shell) ซึ่งเป็นเชลล์ที่พัฒนามาจากบอร์นเชลล์ โดย Free Software Foundation มีการเพิ่มความสามารถบางอย่างจากบอร์นเชลล์ และใส่คุณลักษณะบางอย่างของซีเชลล์เข้ามาด้วย เมื่อคุณทำการล็อกอินได้สำเร็จ แบชจะทำการแสดงพร้อมพร้อมต่อออกมาบรรทัดคำสั่ง นั่นคือ คุณได้สั่งให้มีการใช้งาน โปรแกรมแรกบนลินุกซ์แล้ว นั่นก็คือ โปรแกรมแบชนั่นเอง

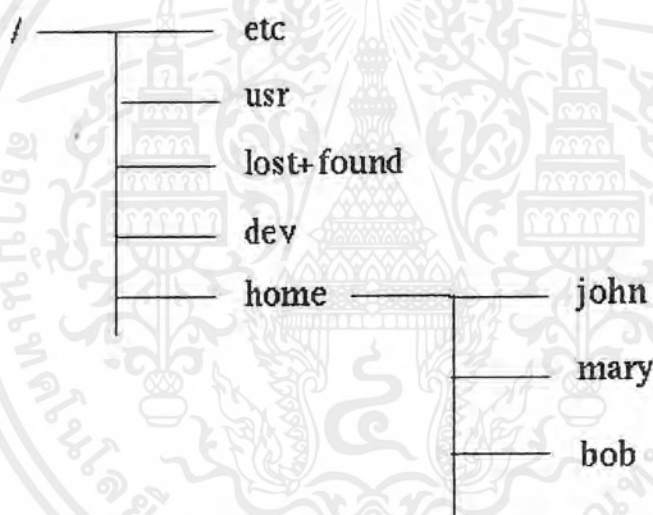
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โครงสร้างของระบบไฟล์และไคลเรกทอรี

ในระบบยูนิกซ์ทำการเก็บข้อมูลโดยใช้ ไฟล์ และ ไคลเรกทอรี เข้ามาช่วย โดยจะมีลักษณะเป็นรูปแบบของลำดับชั้น (heirarchy) หรือ โครงสร้างแบบต้นไม้ หากคุณค้นเลขกับคอสมาก่อน จะเห็นว่ามีการเก็บข้อมูลที่คล้ายกันเพียงแต่การใช้งานจะแตกต่างกันบ้าง ไคลเรกทอรีจะเปรียบเสมือนแฟ้ม ที่สามารถเก็บไฟล์ต่างๆ (เหมือนกับกระดาษ) ในไคลเรกทอรีลำดับบนๆ ก็เหมือนกับแฟ้มขนาดใหญ่ ซึ่งนอกจากจะเก็บไฟล์ได้แล้วก็ยังสามารถเก็บไคลเรกทอรีอื่นๆ ได้ด้วย

ไคลเรกทอรีลำดับบนสุดจะถูกเรียกว่า ไคลเรกทอรีราก (root directory) ซึ่งจะประกอบไปด้วยไฟล์และไคลเรกทอรีต่างๆ ในไคลเรกทอรีที่ชื่อของมากี้อาจจะประกอบไปด้วยไฟล์และไคลเรกทอรีไปเรื่อยๆ ในไฟล์แต่ละไฟล์จะต้องมีชื่อกำกับอยู่ จะมีการอ้างชื่ออยู่สองแบบที่ใช้อ้างถึงไฟล์ได้คือ การอ้างชื่อแบบยาวและแบบสั้น ตัวอย่างชื่อแบบสั้นก็คือ "note" และถ้าจะอ้างถึงชื่อแบบยาวก็อาจจะอ้างได้เป็น "/home/mary/note"

สรุปคือ ในการอ้างถึงชื่อแบบยาว ชื่อทั้งหมดก่อนหน้าชื่อไฟล์ (คือ "/home/mary") จะเป็นชื่อของไคลเรกทอรี ส่วนชื่อ "note" จึงจะเป็นชื่อของไฟล์จริงๆ



รูปที่ 2.2 โครงสร้างของไคลเรกทอรีในยูนิกซ์ (บางส่วน)

2.6 การใช้งานลินุกซ์เบื้องต้น

การเริ่มต้นใช้ลินุกซ์เริ่ม โดยบูตระบบ จากนั้นทำการล็อกอินเข้าสู่ระบบ โดยหน้าจอจะรับชื่อผู้ใช้และรหัสผ่าน ถ้าคุณเป็นคนติดตั้งระบบเอง แน่แน่นอนว่าคุณจะต้องใส่ชื่อผู้ใช้เป็น root ซึ่งเป็นเจ้าของระบบ และตามด้วยรหัสผ่านที่คุณ ได้ให้ไว้ในขั้นตอนการติดตั้ง

Login : root

Password : *****

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อคุณใส่รหัสผ่านได้ถูกต้อง ระบบจะอนุญาตให้คุณเข้ามาใช้งานในระบบได้ ซึ่งจะแสดงเครื่องหมายพร้อมพีซีหรือรับคำสั่ง ในกรณีที่ทำการล็อกอินเข้ามาด้วย super user เครื่องหมายพร้อมพีซีจะแสดงเป็น "#" แต่ถ้าเข้ามาในฐานะผู้ใช้ปกติเครื่องหมายจะแสดงเป็น "\$"

ถ้าต้องการพักการใช้งาน เพื่อเข้าไปใช้อีกครั้ง โดยใช้ชื่อล็อกอินของผู้ใช้คนอื่น ต้องใช้คำสั่ง

```
#logout หรือ #exit
```

คุณจะสามารถกลับมาดูหน้าจอของการรอรับชื่อผู้ใช้และรหัสผ่านอีกครั้งหนึ่ง แต่ในกรณีที่คุณต้องการจะเลิกใช้งานเลย คุณจะปิดเครื่องหลายๆแบบเหมือนกับการทำงานคอสมิกไม่ได้ คุณจะต้องสั่งให้ระบบทำการเก็บข้อมูลทุกอย่างทุกอย่างจากหน่วยความจำลงไปเก็บไว้ในฮาร์ดดิสก์ทั้งหมด และบางทีอาจจะต้องการหยุดโปรแกรมที่รันทิ้งไว้ในหน่วยความจำบางส่วน ไม่ต้องเป็นกังวลกับงานพวกนี้ คุณใช้แค่คำสั่งเดียวเท่านั้น แต่อย่าลืม คุณจำเป็นจะต้องเป็น root หรือเป็นผู้ใช้ที่มีฐานะเทียบเท่ากับ root

```
#/sbin/shutdown -h now    สั่ง Shutdown (ยุติการทำงานระบบ) และหยุดเครื่อง
# /sbin/shutdown -r now    สั่ง Shutdown และให้ระบบทำการบูตใหม่
```

พารามิเตอร์ "now" ข้างหลังคือบอกให้กระทำคำสั่งในทันที ถ้าใส่ตัวเลขก็จะถือเป็นการ หน่วงเวลาก่อนจะกระทำคำสั่ง เมื่อ Shutdown เสร็จแล้วและถ้าเป็นการสั่งแบบหยุดเครื่อง คุณก็สามารถปิดสวิทช์เครื่องของคุณได้เลย ในกรณีที่เข้ามาใช้งานด้วย ชื่อที่เป็นผู้ใช้ธรรมดา และต้องการเปลี่ยนระดับ (permission) ไปเป็น root ก็ให้ใช้คำสั่ง

```
$ su root (หรือ su เลขๆ)
```

```
password : *****
```

```
# ← สัญลักษณ์ '#' หมายความว่า คุณมีฐานะเทียบเท่า root แล้ว
```

การใช้คำสั่งในยูนิกซ์จะแตกต่างจากคอสมิก คือเป็นลักษณะที่เรียกว่า "case-sensitive" หมายความว่าตัวอักษรตัวใหญ่ และตัวอักษรตัวเล็กจะมีความแตกต่างกัน เช่น cat และ Cat จะมีความแตกต่างกัน ดังนั้นเราจะต้องระวังสะกดตัวอักษร ให้ถูกต้องตามคำสั่งที่มีอยู่ในคู่มือด้วย

ในการใช้งานยูนิกซ์จะมีคำพูดวลีหนึ่งคือ "No news is good news" นั่นคือระบบยูนิกซ์ จะมีการแจ้งข้อความออกมาให้น้อยที่สุด ถ้ามีการแจ้งมาแล้วหมายความว่าอาจมีปัญหาเกิดขึ้น

คำสั่งที่ใช้บ่อยๆ มีดังนี้

- **pwd** (print work directory) เป็นคำสั่งแสดงชื่อของไดเรกทอรีที่คุณกำลังอยู่ในขณะนั้นออกมาได้ (current directory)
- **mkdir** (make directory) เป็นคำสั่งในการสร้างไดเรกทอรีใหม่
มีรูปแบบคือ `mkdir [directory]`
- **rm** (remove) เป็นคำสั่งที่ใช้ในการลบไดเรกทอรีและไฟล์
มีรูปแบบคือ `rm [directory]` หรือ `rm filename`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **cd** (change directory) เป็นคำสั่งที่ใช้เปลี่ยน โดเรกทอรี

มีรูปแบบคือ `cd [directory]`

- `cd` หมายถึง กลับไปที่โดเรกทอรีแรกสุดที่เข้ามาหลังจากทำการลือกอิน (home directory)
- `cd..` หมายถึง ต้องการย้าย ไปอยู่โดเรกทอรีที่อยู่ลำดับเหนือขึ้นไป
- `cd.` หมายถึง ระบุโดเรกทอรีตนเอง (current directory)

- **cp** (copy) เป็นคำสั่งให้ทำการสำเนาเพิ่มข้อมูล (หรืออาจจะเป็น โดเรกทอรีด้วยก็ได้) โดยใช้คำสั่ง ตัวอย่างเช่น คำสั่งสำเนาเพิ่มข้อมูล (copy) จาก file ไปเป็นชื่อ file2

มีรูปแบบคือ `cp file1 file2`

- **mv** (move) เป็นคำสั่งให้ทำการย้ายเพิ่มจากโดเรกทอรีหนึ่ง ไปสู่อีกโดเรกทอรีหนึ่ง ตัวอย่างเช่น คำสั่งย้ายเพิ่มชื่อ filename จากโดเรกทอรีปัจจุบัน ไปสู่โดเรกทอรีที่ ชื่อ directory

มีรูปแบบคือ `mv filename directory`

นอกจากนี้ คำสั่งเปลี่ยนชื่อไฟล์จาก file1 ไปเป็นชื่อ file2

รูปแบบคือ `mv file1 file2`

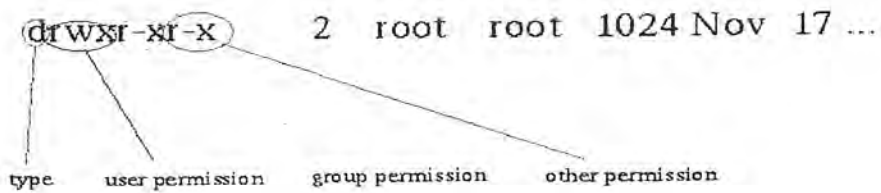
- **ls** (list directory) จะคล้ายกับคำสั่ง DIR ในดอส ซึ่งจะทำการแสดงชื่อของไฟล์และโดเรกทอรี ที่มีอยู่ในโดเรกทอรีที่คุณอยู่ในขณะนั้น หรือ โดเรกทอรีปัจจุบัน(current directory) การดูรายชื่อไฟล์และโดเรกทอรีที่มีอยู่ในโดเรกทอรีรากทำได้โดยใช้คำสั่ง

```
# ls /
bin    etc    lost+found  opt    sbin    var
boot  home  mnt        proc  tmp
dev    lib    net        root  usr
```

โดยทั่วไปแล้วคำสั่งในยูนิกซ์ส่วนใหญ่ จะต้องมีการระบุ option ด้วย ซึ่งถ้าคุณใช้ man คุณอาจจะเห็นรูปแบบของการใช้งานคำสั่งคล้ายๆ กับดังต่อไปนี้ `# ls [-arF] [directory]` นี่คือ รูปแบบ (template) ของคำสั่ง คุณจะสังเกตเห็นเครื่องหมาย "[" และ "]" สิ่งที่จะระบุไว้ในเครื่องหมายกำกับดังกล่าว หมายความว่า เป็น option เราอาจจะใส่หรือไม่ใส่ไว้ก็ได้

ในระบบยูนิกซ์เป็นระบบที่เกี่ยวข้องกับผู้ใช้หลายคน ดังนั้นระบบยูนิกซ์จึงมีการกำหนดเกี่ยวกับเรื่องสิทธิ์ที่จะเข้าถึงเพิ่มข้อมูล (file permission) ในระบบได้ ซึ่งคุณสามารถจะขอลูได้จากคำสั่ง

```
$ ls -l
drwxr-xr-x 2 root root 1024 Nov 17 21:49 axhome
drwxr-xr-x 2 root root 1024 Dec 28 13:57 backup
-rw----- 1 root root 6 Mar 31 23:26 dead.letter
-rw----- 1 root root 3075 Apr 15 03:49 mbox
```



รูปที่ 2.3 รายละเอียดของสิทธิในการเข้าถึงแฟ้มข้อมูล

รายละเอียดของสิทธิในการเข้าถึงแฟ้มข้อมูลแสดงดังรูปที่ 2.3 คอลัมน์แรกเป็นคอลัมน์ที่ระบุเกี่ยวกับเรื่องสิทธิในการเข้าถึงแฟ้มข้อมูล แบ่งเป็นกลุ่มได้ ดังนี้

- *type* คือชนิดของไฟล์ ถ้าหากไม่ระบุ (-) จะหมายความว่าเป็นไฟล์ปกติ แต่ถ้าหากเป็น (d) จะหมายความว่า เป็นไดเรกทอรี และถ้าเป็น (b) จะหมายถึงเป็น device file แบบ block หากเป็น (c) หมายถึง device file แบบ character นอกจากนี้ยังมีรูปแบบชนิดอื่นๆ อีก
- *user permission* จะระบุระดับการอนุญาตให้เข้าถึงไฟล์ในระดับผู้ใช้ (ตัวเอง)
 - r หมายถึง "อ่านไฟล์ได้" (readable)
 - w หมายถึง "เขียนไฟล์ได้" (writeable)
 - x หมายถึง "สามารถเรียกใช้งานได้ หรือรันได้" (executable)
- *group permission* จะระบุระดับการอนุญาตให้เข้าถึงไฟล์ในระดับกลุ่มผู้ใช้ (group) ซึ่งจะมีรูปแบบเหมือนกับ *user permission*
- *other permission* จะระบุระดับการอนุญาตให้เข้าถึงไฟล์ในระดับผู้ใช้รายอื่นๆ ที่ไม่ได้ในกลุ่มเดียวกัน รูปแบบก็จะเหมือนกันกับ *user permission*

ในระบบยูนิกซ์จะมีไฟล์ที่มองไม่เห็นอยู่ด้วย (hidden file) ซึ่งไฟล์เหล่านี้เมื่อ ใช้คำสั่ง ls ธรรมดา จะไม่สามารถมองเห็นได้ เราจะต้องใช้ option -a ดังตัวอย่าง

```
# ls -a
.
..
.saves-622-daffiy.kaiwal.com~
.FVWM95-errors
.Xauthority
.Xclients
.Xclients,bkp
.Xdefaults
.bash_history
.bash_logout
.bash_profile
.bashrc
.cshrc
.seyon
.tcshrc
.tin
.xboing-scores
.xfm
.xsession
.xsession-errors
axhome
backup
dead.letter
mbox
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

hidden file เหล่านี้จะถูกนำหน้าด้วย จุด "." ซึ่งถ้าเราไม่ใช้ option -a ก็จะไม่เห็นไฟล์เหล่านั้น โดยทั่วไปมักจะใช้ไฟล์แบบนี้เพื่อเก็บข้อมูลเฉพาะของรูปแบบ การทำงานของผู้ใช้ (user environment) เช่น .bash_profile, .xsession เป็นต้น

2.7 การดูแลรักษาระบบเบื้องต้น

สำหรับในระบบลินุกซ์จะมีความแตกต่างจากระบบยูนิกซ์เซิร์ฟเวอร์ขนาดใหญ่ เช่นระบบของซันไมโครซิสเต็ม หรือระบบของ HP/UX ซึ่งระบบเหล่านี้ถูกออกแบบให้มาใช้งานในลักษณะของเซิร์ฟเวอร์ (หรือผู้ให้บริการ) เดิมรูปแบบจริงๆ ระบบเหล่านี้ต้องการผู้ดูแลโดยเฉพาะ ซึ่งในแต่ละวันจะมีหน้าที่คอยตรวจสอบ, บริหารระบบ และทำการปรับปรุงระบบ ซึ่งอาจจะเป็นงานที่เกี่ยวข้องกับการดูแลระบบฐานข้อมูล หรือการบริหารระบบเครือข่าย เป็นต้น การใช้งานในลักษณะของพีซีหรืองานส่วนตัว โดยทั่วไปมักจะไม่นิยมกระทำกันบนเครื่องเซิร์ฟเวอร์ ซึ่งสังเกตได้จากไมโครซอฟต์จะแบ่งระบบปฏิบัติการของวินโดวส์เอ็นที (Window NT) ออกเป็นสองระดับก็คือ NT server และ NT workstation โดยจะวาง NT server ในระดับของเครื่องเซิร์ฟเวอร์ขององค์กรซึ่งอาจจะมีบริการระบบฐานข้อมูล และ NT workstation จะนำไปใช้ในลักษณะของงานส่วนบุคคล เช่น การพัฒนาระบบงาน, การทำงานสำนักงาน เป็นต้น

ในระบบลินุกซ์ จะไม่ได้มีการแบ่งแยกอย่างชัดเจนว่าจะอยู่ในส่วนของเซิร์ฟเวอร์หรือ ทั้งนี้ก็ขึ้นอยู่กับรูปแบบในการปรับแต่งของผู้เป็นเจ้าของระบบเองว่า ต้องการจะใช้งานลินุกซ์ในฐานะอะไร หากคุณต้องการจะใช้งานในฐานะของเซิร์ฟเวอร์ไม่ว่าจะเป็น ดาตาเบสเซิร์ฟเวอร์ (ระบบฐานข้อมูล), อินทราเน็ตเซิร์ฟเวอร์ หรืออินเทอร์เน็ตเซิร์ฟเวอร์ ก็ตาม การใช้งานในระดับนี้คุณก็ต้องมีความรู้ในฐานะของผู้ดูแลระบบ (System Administrator) พอสมควร แต่ถ้าคุณใช้ลินุกซ์ในฐานะของเวิร์กสเตชัน คุณอาจจะไม่จำเป็นจะต้องมีความรู้ในฐานะของผู้ดูแลระบบมากนักได้ เพราะซอฟต์แวร์ส่วนใหญ่บนลินุกซ์ที่ใช้ในฐานะของเวิร์กสเตชัน จะถูกออกแบบมาให้ทำการดูแล รักษาและใช้งานง่าย รวมทั้งการจัดเตรียมระบบมาตรฐานที่มีมาให้จากดิสทริบิวเตอร์ ก็สามารถทำให้คุณใช้งานได้อย่างสะดวกสบาย และมีสถิติที่บ่งบอกว่ามีผู้ใช้งานลินุกซ์จำนวนมากที่ไม่ได้เคยผ่าน

การใช้งานยูนิกซ์มาก่อน ซึ่งเขาเหล่านั้นก็สามารถใช้งานลินุกซ์ได้โดยไม่มีปัญหาอะไร แต่อย่างไรก็ตามหากคุณเรียนรู้เรื่องของการดูแลระบบไว้บ้างคุณก็จะมีความเข้าใจและปรับแต่งระบบให้ทำงานได้อย่างมีประสิทธิภาพขึ้น

ผู้ใช้ทั่วไปจะถูกจำกัดการใช้งาน ให้อยู่ในขอบเขตตามที่ผู้ดูแลระบบกำหนด แต่สำหรับงานที่เกี่ยวข้องกับการดูแลรักษาระบบแล้ว จำเป็นต้องใช้สิทธิในระดับของ super user การเป็น super user หรือ root นั้นหมายถึงการมีสิทธิอำนาจสูงสุดในระบบ คุณสามารถที่จะทำการแก้ไขและเปลี่ยนแปลงได้ทุกอย่าง แต่คุณก็ต้องระมัดระวังเมื่อจะทำการแก้ไขดังกล่าวนี้ด้วย เนื่องจากหากมีความผิดพลาดอาจหมายถึงการทำลายระบบลงได้

คุณสามารถจะเป็น super user ได้สองวิธีคือ ทำการล็อกอินเข้าด้วยชื่อผู้ใช้ "root" หรือใช้คำสั่ง "su" ที่คอมมานด์ไลน์เพื่อขอเปลี่ยนสิทธิผู้ใช้ให้เป็นระดับของ superuser

เราสามารถจะใช้เครื่องมือเล็กๆสองตัวคือ grep กับ find เพื่อช่วยในการบริหารงานระบบได้ สำหรับ grep จะใช้เลือก output ที่มีความสอดคล้องกับที่เราระบุไว้ขึ้นมาและ find จะใช้สำหรับตรวจหาไฟล์ที่ต้องการ

คำสั่ง grep

ตัวอย่างการตรวจหาโปรแกรมที่ถูกละทิ้งไว้ในหน่วยความจำของผู้ใช้ชื่อ "twg"

```
# ps -aux | grep twg
twg  524 0.0 2.3 2416 728 ?    S  14:03  0:04  fvwm95-2
twg  531 0.0 1.6 2208 512 ?    S  14:03  0:00  /usr/X11R6/lib/X11/fv
twg  532 0.0 2.4 2292 740 ?    S  14:03  0:01  /usr/X11R6/lib/X11/fv
twg  536 0.0 1.4 2192 432 ?    S  14:03  0:00  /usr/X11R6/lib/X11/fv
twg  537 0.0 2.9 3336 896 ?    S  14:03  0:00  xclock -bg #c0c0c0 -p
twg  538 0.0 2.8 3372 880 ?    S  14:03  0:00  xload -nolabel -bg gr
twg  541 0.0 1.8 1268 572 p1   S  14:03  0:00  -bash
twg  554 0.0 1.9 1264 612 p2   S  14:03  0:00  -bash
twg  649 0.0 0.6 1552 204 ?    S  14:16  0:00  /opt/applix/axdata/ax
twg 1325 1.0 36.7 16968 11304 ?    S  15:09  0:45  netscape -geometry 79
twg 1326 0.0 3.0 10276 936 ?    S  15:09  0:00  (dns helper)
twg 1648 0.0 2.0 1212 636 p2   S  16:15  0:00  vi sysadmin1.txt
twg 1661 0.0 1.6 1188 500 p2   S  16:19  0:00  /bin/bash -c (ps -aux
twg 1662 0.0 1.7 1192 540 p2   S  16:19  0:00  /bin/bash -c (ps -aux
twg 1663 0.0 1.3 928 416 p2   R  16:19  0:00  ps -aux
twg 1664 0.0 1.0 912 308 p2   R  16:19  0:00  grep twg
```

หากต้องการให้ระบุส่วนหัวซึ่งจะอธิบายความหมายของคอลัมน์แต่ละคอลัมน์ ให้ใช้คำสั่งดังต่อไปนี้

```
# ps -aux | egrep 'twg|PID'
```

คำสั่ง find

คำสั่ง find จะใช้ในการตรวจค้นหาไฟล์ ซึ่งมีรูปแบบในการใช้งานดังนี้

```
# find <ไดเรกทอรีที่เริ่มต้นค้นหา> <เงื่อนไขที่ใช้ตรวจหาและคำสั่งที่ทำเมื่อพบ>
```

การใช้คำสั่ง find จะช่วยผู้ดูแลระบบในเรื่องต่อไปนี้

- ตรวจสอบการใช้เนื้อที่ของดิสก์เพื่อบริหารการใช้งานดิสก์อย่างมีประสิทธิภาพ
- ค้นหาไฟล์ที่อาจมีปัญหาเกี่ยวกับเรื่องระบบการรักษาความปลอดภัย
- ทำงานที่ซ้ำๆกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง	ความหมาย
Atime n	ไฟล์ที่ถูกใช้งาน (accessed) ไปเมื่อ n วันที่ผ่านมาแล้ว
Mtime n	ไฟล์ที่ถูกแก้ไข (modified) ไปเมื่อ n วันที่ผ่านมาแล้ว
size n	ไฟล์ที่มีขนาดความยาวเท่ากับ n X 512 ไบต์
type c	ไฟล์ที่มีชนิดเป็น : f = ไฟล์ธรรมดา, d = โดเรกทอรี เป็นต้น
Fstype typ	ระบุประเภทของระบบไฟล์ : เช่น nfs (Network file system)
Name nam	ไฟล์ที่มีชื่อเป็น nam
user usr	ไฟล์ที่มีเจ้าของชื่อ usr
Group grp	ไฟล์ที่เป็นของกลุ่มที่ชื่อ grp
Perm p	ไฟล์ที่มีรูปแบบการอนุญาตการใช้งานเป็น p

ตารางที่ 2.1 เงื่อนไขที่ใช้ในการค้นหาไฟล์

คุณสามารถระบุเครื่องหมาย + (มากกว่า) และเครื่องหมาย - (น้อยกว่า) ประกอบไปกับเงื่อนไขต่างๆ ได้ ตัวอย่าง

mtime +7 หาไฟล์ที่มีการแก้ไขมาแล้วมากกว่า 7 วัน
 atime -2 หาไฟล์ที่มีการใช้งานน้อยกว่า 2 วัน
 size +100 ไฟล์ที่มีขนาดใหญ่กว่า 50K (512*100)

คำสั่ง	ความหมาย
Print	ให้พิมพ์ชื่อแบบยาวของไฟล์ที่ค้นหาเจอ
Exec cmd	ให้กระทำการคำสั่ง cmd กับไฟล์ที่หาเจอ
Ok cmd	ให้ถามยืนยันก่อนที่จะทำการคำสั่ง cmd กับไฟล์ที่หาเจอ

ตารางที่ 2.2 คำสั่งที่กระทำเมื่อค้นพบ

ตัวอย่างเช่น ให้หาไฟล์ที่เป็นซอร์สโค้ดของภาษาซี (จะลงท้ายด้วย .c) แล้วพิมพ์รายชื่อของไฟล์ที่เจอออกมา ใช้คำสั่งดังนี้

```
# find . -name \*.c -print
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การใช้งานภาษาไทยบนลินุกซ์

2.8.1 การติดตั้งภาษาไทยบนลินุกซ์

การใช้งานภาษาไทยบนลินุกซ์นั้นมีหลายวิธีด้วยกัน ในระบบนี้ เราเลือกใช้โปรแกรม Thai Extension for Linux ซึ่งติดตั้งได้ง่ายแล้วใช้งานได้ดี รายละเอียดการติดตั้งและการใช้งานภาษาไทย มีดังนี้

Download โปรแกรม TE_Jan22.tar.gz จากเว็บไซต์ <http://www.fcd.ucc.ac.jp/Zzz/Linux/te.html>

ขยายไฟล์ออกมา โดยใช้คำสั่ง `$tar xfvz TE_Jan22.tar.gz`

ทำการติดตั้ง โดยเรียก `Tinst.tcl` ในไดเรกทอรี `TE_Jan22` จากเทอร์มินอลดังนี้

```
$. /Tinst.tcl
```

ในการติดตั้งให้เลือกติดตั้งโปรแกรมทุกตัว ซึ่งได้แก่

- likit โปรแกรมลิขิตเป็นอิลิคเตอร์ภาษาไทย
- txterm โปรแกรมเทอร์มินอลที่แสดงอักษรไทย
- xzthai โปรแกรมควบคุมการแสดงผลภาษาไทยและอิลิคเตอร์

โปรแกรมทั้งสามจะติดตั้งอยู่ในไดเรกทอรี `/usr/local/bin` เราสามารถเรียกใช้งานโปรแกรมภาษาไทยได้โดยใช้คำสั่งต่อไปนี้

```
$/usr/local/bin/likit                   ; ในกรณีเรียกใช้โปรแกรม likit
```

```
$/usr/local/bin/txterm               ; ในกรณีเรียกใช้โปรแกรม txterm
```

```
$/usr/local/bin/xzthai               ; ในกรณีเรียกใช้โปรแกรม xzthai
```

การใช้งาน ปุ่มควบคุมที่ใช้ในการเปลี่ยนจากภาษาไทยเป็นอังกฤษ คือ

<Alt+Shift ปุ่มซ้าย> เปลี่ยนเป็นภาษาอังกฤษ

<Alt+Shift ปุ่มขวา> เปลี่ยนเป็นภาษาไทย

2.8.2 การปรับแต่งให้เนตสเคปแสดงผลภาษาไทยได้

การปรับแต่งให้เนตสเคปแสดงผลภาษาไทย ใช้วิธีเดียวกันกับที่ใช้กับเนตสเคปบนวินโดวส์ เมื่อเปิดเนตสเคปขึ้นมา ให้เลือกไปที่ `Edit>Preference` จากนั้น เลือกไปที่ช่อง `Font` ในช่องตัวเลือก `For the Encoding` ให้เลือก `User – Defined` ส่วนตัวเลือก `Variable Width Font` ให้เลือก `Font` ที่เป็นภาษาไทย เช่น `Fixed (Thai)` หรือ `PsThai` และ กด `OK` จากนั้นเลือกเมนู `View>Encoding` แล้วเลือกไปที่ `User Defined` เพียงเท่านั้น เนตสเคป ก็สามารถแสดงผลเป็นภาษาไทยได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

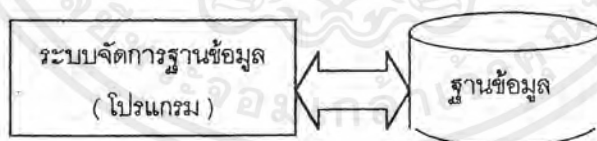
ระบบจัดการฐานข้อมูล

3.1 ความหมายและที่มา

ฐานข้อมูล (Database) คือ แหล่งเก็บข้อมูล และความสัมพันธ์ระหว่างข้อมูลเหล่านั้น ในอดีตการ จัดเก็บและค้นหาข้อมูลยังเก็บอยู่ในรูปแบบของไฟล์ (file) ซึ่งก็ใช้งานได้ในระบบขนาดเล็ก มีผู้ใช้งานอยู่ เพียงไม่กี่คน แต่ในระบบที่มีขนาดใหญ่ขึ้น มีผู้ใช้มากขึ้น การค้นหาและจัดเก็บข้อมูลจึงต้องการประสิทธิภาพ ที่สูงขึ้น

สาเหตุที่ทำให้มีการนำระบบฐานข้อมูลมาใช้แทนไฟล์ระบบเก่า มีดังนี้

- ลดความซ้ำซ้อนของข้อมูล (Minimum Redundancy) การที่แผนกต่างๆ ต้องการข้อมูลชนิดเดียวกัน แต่ต่างแยกกันเก็บ ทำให้สิ้นเปลืองเนื้อที่ในการจัดเก็บข้อมูล
- หลีกเลี่ยงการขัดแย้งกันเองของข้อมูล (Data Inconsistency)
- เมื่อมีการเปลี่ยนแปลงข้อมูลที่จัดเก็บในหลายๆ ที่ หลายๆ ไฟล์ จะต้องตามทำการเปลี่ยนแปลงทุก ไฟล์ที่จัดเก็บ ถ้าทำไม่ครบทุกไฟล์ ก็จะทำให้ข้อมูลมีความผิดพลาด ไปจากความจริง และขัดแย้งกัน
- แก้ปัญหาการรักษาความปลอดภัยของข้อมูล (Data Security Problem) เนื่องจากมีการเก็บข้อมูล กระจัดกระจาย ทำให้ยากต่อการรักษาความปลอดภัย เลี่ยงต่อการรั่วไหลและดัดแปลงข้อมูล
- ทำให้ข้อมูลมีความถูกต้องของข้อมูลมากที่สุด (Maximum Integrity)
- ทำให้สามารถควบคุมการใช้งาน โดยส่วนกลางได้ (Centralized Control)
- สามารถใช้ข้อมูลร่วมกัน ได้
- มีความคล่องตัว และความยืดหยุ่นในการใช้งาน (Centralized Independent)
- มีการควบคุมมาตรฐานร่วมกัน



รูปที่ 3.1 ฐานข้อมูลและระบบจัดการฐานข้อมูล

ระบบฐานข้อมูลเป็นระบบที่ใช้จัดเก็บข้อมูลอย่างมีระเบียบ เราสามารถนำข้อมูลมาใช้ประโยชน์ ได้ทันทีด้วยวิธีการที่กำหนดไว้ สิ่งที่ใช้ในการบันทึกข้อมูลลงไป ในฐานข้อมูลและดูแลข้อมูลไม่ให้สูญหาย เรียกว่า ระบบจัดการฐานข้อมูล (Database Management System หรือ DBMS)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 สถาปัตยกรรมภายในของออรากิล

ออรากิล (Oracle) เป็นระบบจัดการฐานข้อมูลที่ไม่ขึ้นกับแพลตฟอร์ม สถาปัตยกรรมภายในของออรากิล ไม่ว่าจะอยู่บนระบบปฏิบัติการใดก็ตาม จะเหมือนกันกว่า 80% ส่วนที่แตกต่างกันจะเป็นส่วนติดต่อกับผู้ใช้ (User Interface) การใช้งานคำสั่งหรือหน่วยความจำจะเหมือนกัน ดังนั้น เราสามารถออกแบบระบบงานที่ใช้ออรากิลเป็นฐานข้อมูลบนเครื่องคอมพิวเตอร์ขนาดเล็กที่ใช้วินโดวส์เป็นระบบปฏิบัติการแล้วนำไปใช้งานบนระบบยูนิกซ์ขนาดใหญ่ได้โดยไม่ต้องแก้ไขเลย

สถาปัตยกรรมของระบบจัดการฐานข้อมูลออรากิล ถูกออกแบบมาในลักษณะ Client-Server Relational DBMS (RDBMS) ประกอบด้วย 4 ส่วนหลัก (ส่วนเชื่อมโยง 3 ส่วน และส่วนข้อมูล 1 ส่วน) แสดงดังรูป 3.2



รูปที่ 3.2 องค์ประกอบทั้ง 4 ส่วน และวิธีเชื่อมต่อกัน

ผู้ใช้ คือ บุคคลที่กำลังใช้โปรแกรมลูกข่าย (Client Program) ซึ่งกำลังติดต่อกับ โปรแกรม DBMS อาจเรียกว่าเซิร์ฟเวอร์ก็ได้ ซึ่งทำหน้าที่ติดต่อกับตัวฐานข้อมูล จากรูป 3.2 ผู้ใช้, โปรแกรมลูกข่าย, ระบบจัดการฐานข้อมูล และฐานข้อมูล ถูกเชื่อมโยงเข้าด้วยกันโดยตัวเชื่อม 3 ส่วน (The Three Interface) ดังนี้

- ตัวเชื่อมที่ 1 *User Interface* คือส่วนที่แสดงบนจอแสดงผลเมื่อผู้ใช้ส่งคำสั่งเข้าไปผ่านแป้นพิมพ์
- ตัวเชื่อมที่ 2 *Client-Server Interface* เป็นตัวเชื่อมระหว่าง โปรแกรมแม่ข่ายและ โปรแกรมลูกข่าย สำหรับ ORACLE Workgroup Server ตัวเชื่อมนี้ก็คือภาษา SQL หรือ Structure Query Language

ในการทำงานจริง โปรแกรมลูกข่ายและ โปรแกรมแม่ข่ายจะอยู่บนเครื่องคอมพิวเตอร์คนละเครื่อง ติดต่อกันด้วยระบบเครือข่ายในสำนักงาน (LAN) โปรแกรมที่ทำให้โปรแกรมลูกข่ายคุยกับ โปรแกรมแม่ข่ายผ่านระบบเครือข่ายแบบนี้ก็คือ โปรแกรม SQL*Net (ในกรณีที่โปรแกรมลูกข่ายและ โปรแกรมแม่ข่ายอยู่ในเครื่องเดียวกันก็สามารถคุยกันด้วย SQL*Net ได้เช่นกัน)

- ตัวเชื่อมที่ 3 *I/O Interface* เป็นตัวเชื่อมระหว่าง โปรแกรมแม่ข่ายกับข้อมูล คือการอ่านหรือเขียนข้อมูลบนจานแม่เหล็กหรือดิสก์

ข้อดีของออรากิลอีกข้อหนึ่ง คือ การออกแบบระบบฐานข้อมูลมีลักษณะที่เป็นแบบ Client-Server ดังนั้น สามารถใช้ front-end Tool ต่างๆ กับออรากิลที่วิ่งบนระบบยูนิกซ์ได้ โดยเปลี่ยนชื่อ Connect String ที่ใช้ติดต่อกัน

ออราเคิลทำหน้าที่เป็นเซิร์ฟเวอร์เพื่อคอยให้บริการด้านฐานข้อมูล ซึ่งต้องมีโปรแกรมบริหารฐานข้อมูล นอกจากนี้ ออราเคิลยังมีโปรแกรมอื่นๆ อีก เช่น โปรแกรมลูกข่าย และโปรแกรมทางด้านเครือข่าย (SQL*Net) เพื่อให้โปรแกรมลูกข่ายสามารถคุยกับโปรแกรมแม่ข่ายที่อยู่อีกเครื่องหนึ่งได้ โดยโปรแกรมแม่ข่ายคือ Oracle Server นั่นเอง ส่วนโปรแกรมลูกข่ายและโปรแกรมแม่ข่ายนั้นแสดงในตาราง 3.1 และ 3.2 ตามลำดับ

โปรแกรมลูกข่าย	คำอธิบาย (ประโยชน์)
Backup Manager	สำรองไฟล์ฐานข้อมูล
Database Manager	เริ่มหรือหยุดการใช้งานฐานข้อมูล, กำหนดค่าเริ่มต้นของพารามิเตอร์สำหรับปรับแต่งฐานข้อมูล
Recovery Manager	กู้ฐานข้อมูล โดยการนำไฟล์ที่ได้จากการสำรองข้อมูลข้อมูลด้วยโปรแกรม Backup Manager ในกรณีที่เกิดปัญหาเกี่ยวกับฐานข้อมูล เช่น คิส์ก์เสีย
Session Manager	ดูแลการติดต่อของผู้ใช้, ตรวจสอบการใช้งานของผู้ใช้งานระบบในขณะนั้น
Server Manager	ทำคำสั่ง SQL และใช้ควบคุมฐานข้อมูล
SQL*Loader	นำข้อมูลจาก Text File บันทึกลงในตาราง
SQL*Plus	ทำคำสั่ง SQL ต่างๆ มีความสามารถด้านการออกรายงานอย่างง่าย
User Manager	การสร้างและแก้ไขลักษณะของ Username และ Role

ตารางที่ 3.1 โปรแกรมลูกข่ายที่สำคัญของออราเคิล

โปรแกรมเครือข่าย (SQL*Net)	คำอธิบาย (ประโยชน์)
APPC/LU6.2 Driver/Listener	ติดต่อใช้งานฐานข้อมูลผ่านระบบเครือข่ายโปรโตคอล APPC/LU6.2 (ส่วนของ Driver จะติดตั้งที่ฝั่งเครื่องลูกข่าย ส่วนของ Listener จะติดตั้งที่ฝั่งเครื่องแม่ข่าย ในการติดต่อผ่านระบบเครือข่ายจะต้องมีส่วนนี้ควบคู่กันไปเสมอ ไม่ว่าจะใช้โปรโตคอลชนิดใด โดยเลือกชนิดของ Driver และ Listener ให้ตรงกับชนิดของโปรโตคอล)
Listener Controller	ควบคุม SQL*Net Listener
TCP/IP driver/listener	ติดต่อใช้งานฐานข้อมูลผ่านระบบเครือข่ายโปรโตคอล TCP/IP
TNS Listener	ช่วยให้ระบบจัดการฐานข้อมูล สามารถติดต่อกับโปรแกรมลูกข่ายผ่านระบบเครือข่ายใด (ที่ออราเคิลสนับสนุน) ก็ได้

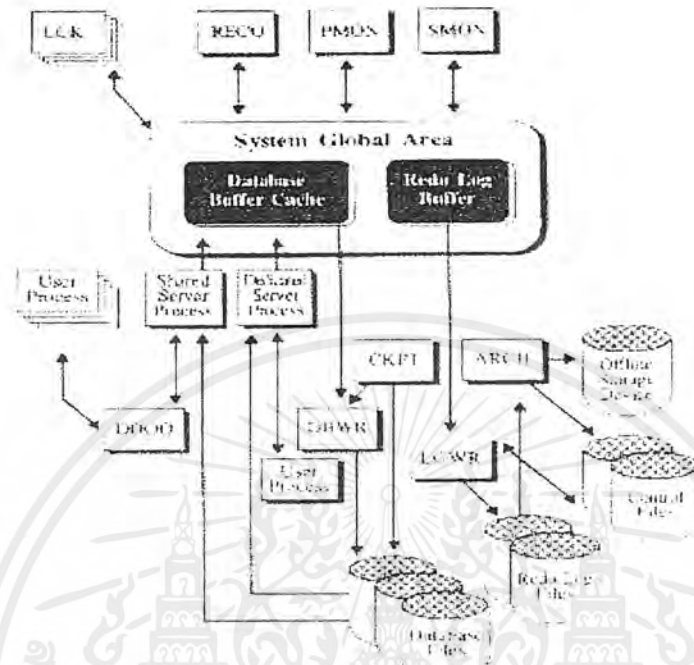
ตารางที่ 3.2 โปรแกรมเครือข่ายที่สำคัญของออราเคิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

3.3 โครงสร้างภายใน (Internal Structure)

โครงสร้างภายในหลักๆ ของออร์ราเคิลประกอบด้วย 3 ส่วน คือ ไฟล์ฐานข้อมูล หน่วยความจำ และ โพรเซส (Process) ของออร์ราเคิล ดังรูป



รูปที่ 3.3 สถาปัตยกรรมของ Oracle Workgroup Server

3.3.1 ไฟล์ฐานข้อมูล

ขั้นตอนการทำงานของออร์ราเคิลในการจัดการกับไฟล์ฐานข้อมูลมีอยู่หลายแบบขึ้นอยู่กับชนิดของไฟล์ ซึ่งแบ่งเป็นชนิดใหญ่ๆ ได้คือ

- ไฟล์ข้อมูล (Data File) จัดเก็บข้อมูลต่างๆ ที่เราใส่เข้าไป
- ไฟล์ล็อก (Log File) ใช้จัดเก็บคำสั่งและข้อมูลที่มีการป้อนเข้าไปในช่วงเวลานั้น เพื่อใช้ในการกู้ข้อมูลหากเกิดปัญหา
- ไฟล์ควบคุม (Control File) เป็นไฟล์ที่ใช้บอกถึงโครงสร้างภายในของระบบฐานข้อมูล โดยบอกถึงตำแหน่งของแฟ้มข้อมูลในดิสก์
- ไฟล์คอนฟิกูเรชัน (Configuration File) จัดเก็บพารามิเตอร์ต่างๆ ของออร์ราเคิล บอกถึงขนาดหน่วยความจำที่ใช้

3.3.2 โครงสร้างของหน่วยความจำ (Memory Structures)

เมื่อเราเริ่มค้นทำงานกับฐานข้อมูล ออราเคิลจะจองเนื้อที่ว่างในหน่วยความจำเพื่อใช้ทำงาน หน่วยความจำส่วนนี้เรียกว่า System Global Area (SGA) ซึ่งแบ่งเป็น 3 ส่วนใหญ่ๆ คือ

Database Buffer Cache

Cache เป็นหน่วยความจำที่ไว้เก็บข้อมูลที่มีการใช้บ่อยๆ เพื่อจะได้ไม่จำเป็นต้องอ่านจากดิสก์ทุกครั้งที่ต้องการใช้ ทำให้ประหยัดเวลาได้มาก การทำงานก็จะเร็วขึ้น

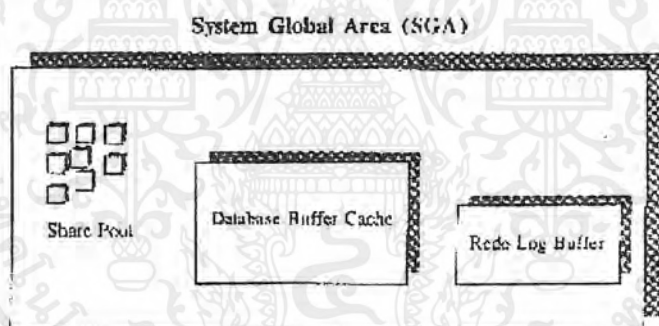
Database Buffer Cache คือที่เก็บข้อมูลในหน่วยความจำ เพื่อใช้สืบค้น (Query) หรือทำตามคำสั่ง SQL ได้อย่างรวดเร็ว

Redo Log Buffer

ก่อนที่จะบันทึกข้อมูลเกี่ยวกับคำสั่งลงในไฟล์ล็อก จะต้องบันทึกลงใน Redo Log Buffer เสียก่อน ลักษณะการบันทึกจะเป็นตามลำดับ (Sequential Write) และเมื่ออ่านจาก Redo Log Buffer ไปบันทึกลงใน Redo Log File ก็จะเป็นแบบตามลำดับเช่นกัน

Share Pool Buffer

เป็นเนื้อที่สำหรับเก็บข้อมูลหลายอย่าง เช่น คำสั่ง SQL ที่ถูกเรียกใช้บ่อยๆ Data Dictionary ฯลฯ ข้อมูลเหล่านี้เมื่ออ่านขึ้นมาจากดิสก์แล้วจะค้างอยู่ในหน่วยความจำเพื่อให้ผู้อื่นได้ใช้ด้วย นอกจากนี้จะกันหน่วยความจำส่วนหนึ่ง เพื่อไว้เก็บค่าตัวแปรที่ใช้ โดยผู้ใช้แต่ละคนไม่ใช้ร่วมกัน (Private Area)



รูปที่ 3.4 System Global Area

3.3.3 โพรเซสของออราเคิล

เมื่อเริ่มใช้ฐานข้อมูล ออราเคิลจะสร้างโพรเซสขึ้นมาชุดหนึ่งเพื่อรองรับงานฐานข้อมูล โดยจะประกอบด้วยโพรเซสมาตรฐานที่สำคัญ ดังนี้

- Server Process ทำหน้าที่ดูแลความต้องการของผู้ใช้ที่ส่งเข้ามาให้ออราเคิลปฏิบัติ
- PMON (Process Monitor) ทำหน้าที่ Process Recovery กล่าวคือ นำหน่วยความจำที่ใช้ในการประมวลผลของผู้ใช้คืน เมื่อผู้ใช้ปิดเซสชันแล้ว

- SMON (System Monitor) ทำหน้าที่กู้คืนฐานข้อมูล, ลบข้อมูลใน Temporary Segment ที่ไม่จำเป็นคือใช้ และคอยรวบรวมส่วนของไฟล์ข้อมูลที่ไม่ได้ใช้ให้ต่อเนื่องกัน เพื่อให้มีที่ว่างมากๆ
- DBWR (Database Writer) ทำหน้าที่ดูแล Buffer Cache ให้เพียงพอกับการใช้งาน User Process และ Server Process
- LGWR (Log Writer) ทำหน้าที่ดูแลจัดการ Redo Log Buffer

3.4 ตาราง (Table)

ตาราง คือ โครงสร้างที่ใช้จัดเก็บข้อมูลที่สัมพันธ์กันไว้ด้วยกัน โดยแบ่งข้อมูลที่สัมพันธ์กันออกเป็น ส่วน ๆ ตามแนวตั้งที่เรียกว่า คอลัมน์ (Column) โดยมีเมตาดาตา (Metadata) เพื่ออธิบายข้อมูลดิบ จากรูป เมตาดาตา ได้แก่ EMPLOYEE, NAME, JOB ฯลฯ

EMPLOYEES				
NAME	JOB	BOSS	SAL	OFFICE
Jones	Clerk	Smith	7,000	Dallas
Smith	President		5,000	Dallas
Ford	Sales Rep	Clark	3,000	Chicago
Clark	Manager	Smith	4,000	Chicago

รูปที่ 3.5 ตัวอย่างของตารางชื่อ EMPLOYEES

ข้อมูลที่ใช้อธิบายข้อมูลดิบ เราเรียกว่า เมตาดาตา ซึ่งจัดเก็บไว้ในตารางพิเศษกลุ่มหนึ่ง เราเรียกดังกล่าวกันว่า คอลเลกชันนารี (Data Dictionary)

TAB เป็นวิวที่เป็นคอลเลกชันนารีวิวหนึ่ง เมื่อเรียกใช้จะแสดงรายชื่อตารางหรือวิวของผู้ใช้ที่เรา Connect เข้าไปในฐานข้อมูล คอลเลกชันนารีนี้ได้รับการดูแลโดย RDBMS ผู้ใช้สามารถเรียกดูได้ คำสั่งที่ใช้ในการเรียกดูโครงสร้างของ TAB คือคำสั่ง describe

```
SQL > describe tab;
```

คำสั่งนี้จะเรียกดูคำอธิบายคอลเลกชันนารีที่เป็นวิวที่ชื่อว่า TAB ข้อมูลที่แสดงนี้ได้มาจากคอลเลกชันนารีเป็นดังนี้

Name	Null?	Type
-----	-----	-----
TNAME	NOT NULL	VARCHAR2(30)
TABTYPE		VARCHAR2(7)
CLUSTERID		NUMBER

หมายเหตุ เนื่องจาก TAB เป็นวิว หากเรา Connect เป็นผู้ใช้อื่น เราอาจจะเห็นข้อมูลที่แตกต่างกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของข้อมูล (Data Type) ที่จัดเก็บในตาราง มีดังต่อไปนี้

- **CHAR** ข้อมูลชนิด CHAR เป็นข้อมูลแบบตัวอักษรหรือตัวเลขที่ไม่ใช่เพื่อการคำนวณ เมื่อเราระบุขนาดความกว้างของคอลัมน์แล้ว ขนาดของที่บรรจุหรือที่จัดเก็บจะคงที่แม้ว่าเราจะใส่ข้อมูลไม่เต็มช่องก็ตาม สำหรับช่องที่เหลือนั้น ORACLE Workgroup Server จะเติมช่องว่างให้จนเต็ม แต่หากเราใส่ข้อมูลจนเกินขนาดของคอลัมน์ ORACLE Workgroup Server ก็จะตัดส่วนที่เกินทิ้งให้เช่นกัน ข้อมูลแบบ CHAR นี้สามารถบรรจุข้อมูลได้ 255 ตัวอักษร
- **DATE** ชนิดข้อมูลแบบ DATE นี้จะจัดเก็บข้อมูลทั้งวันและเวลา คือ ปี เดือน วัน ชั่วโมง นาที วินาที และ เศษของวินาที รูปแบบของ DATE จะจัดเก็บข้อมูลโดยบีบอัดในที่เก็บขนาด 7 ไบต์ รูปแบบมาตรฐานที่ใช้อ้างอิงข้อมูลแบบ DATE คือ “DD-MON-YY”
- **VARCHAR2** ชนิดข้อมูลแบบ VARCHAR2 จะคล้ายกับ CHAR แต่จะจัดเก็บตามขนาดข้อมูลที่เป็นจริง เช่นถ้าเรากำหนดชนิดของข้อมูลเป็น VARCHAR2 ขนาด 20 ตัวอักษร เราสามารถเก็บข้อมูลที่ตัวอักษรก็ได้ แต่จะต้องไม่เกิน 20 ตัว จำนวนไบต์ที่ใช้จัดเก็บก็จะเท่ากับจำนวนตัวอักษร ไม่มีการเติมช่องว่างให้เปลืองที่จัดเก็บ ซึ่งทำให้ประหยัดกว่าแบบ CHAR มาก เราสามารถกำหนดขนาดของ VARCHAR2 ได้สูงสุด 2000 ตัวอักษร
- **RAW** ชนิดข้อมูลแบบ RAW จะเหมือนกับ VARCHAR2 แต่สามารถเก็บข้อมูลนอกเหนือจากตัวเลขและตัวอักษร ได้ สามารถเก็บได้สูงสุด 255 ไบต์
- **LONG** ข้อมูลชนิด LONG จะคล้ายกับ VARCHAR2 แต่จะขยายขนาดของข้อมูลได้ถึง 2 พันล้านตัวอักษร ส่วนมากใช้จัดเก็บข้อมูลประเภทข้อความที่มีขนาดใหญ่
- **LONG RAW** มีลักษณะเหมือนกับ LONG เรามักจัดเก็บข้อมูลประเภท ภาพ เสียง ภาพเคลื่อนไหว เพลงต่าง ๆ ไว้ในคอลัมน์ชนิดนี้
- **NUMBER** ข้อมูลชนิด NUMBER ใช้เก็บตัวเลขที่ใช้ในการคำนวณ มีเพียงแบบเดียวใน ORACLE Workgroup Server แต่สามารถเก็บตัวเลขได้ทุกชนิด ขนาดของตัวเลขที่สามารถเก็บได้คือ 38 ตำแหน่ง หรืออยู่ในช่วง -9.99×10^{125} $+9.99 \times 10^{125}$ และค่าที่เล็กที่สุดที่สามารถอ้างอิงได้ก็คือ 1×10^{-130} เมื่อเรากำหนดให้คอลัมน์ใดคอลัมน์หนึ่งใช้ข้อมูลแบบ NUMBER เราสามารถระบุได้ทั้งจำนวน Precision และ Scale โดยที่ Precision คือจำนวนตำแหน่งทั้งหมดที่จัดเก็บ ส่วน Scale คือจำนวนตำแหน่งสูงสุดหลังจุดทศนิยม

3.5 SQL (Structure Query Language)

SQL เป็นภาษาที่ใช้จัดการกับข้อมูลที่เก็บอยู่ในรีเลชันเนลดาตาเบส (Relational Database) เราใช้ SQL ในการจัดการกับข้อมูลได้ 5 รูปแบบ ดังนี้

1. Data Definition : การสร้างดาตาเบสและ โครงสร้างของตาราง
2. Data Manipulation : การ INSERT, DELETE, และ UPDATE ข้อมูลที่อยู่ในตาราง
3. Data Query : การค้นหาและแสดงผลข้อมูลที่ต้องการจากฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Transaction Processing : การทำทรานแซกชันโดยใช้คำสั่ง ROLLBACK, COMMIT เป็นต้น
5. Data Control : การควบคุมการเข้าถึงข้อมูลโดยใช้สิทธิ์ และอำนาจ เช่น คำสั่ง GRANT

โหมดการประมวลผลคำสั่ง SQL มี 2 โหมด คือ

1. Interactive Processing เป็นการประมวลผลคำสั่ง SQL ที่ SQL prompt ซึ่งเป็นวิธีที่สะดวก รูปแบบของ SQL prompt ได้แก่ SQL> SQL>_ SQL>_
2. Embedded SQL Commands เป็นการนำคำสั่ง SQL ติดเอาไว้ในโปรแกรมต่างๆ เช่น FORTRAN, COBOL, C เป็นต้น

3.5.1 ไวยากรณ์ของภาษา SQL

ภาษา SQL เป็น nonprocedural language มีไวยากรณ์ของภาษาที่สำคัญ ดังนี้

- ตัวอักษรใหญ่หรือเล็กมีความหมายเหมือนกัน
- เว้นช่องว่างอย่างน้อย 1 ช่องในการแยกคำแต่ละคำออกจากกัน
- ความยาวของคำสั่งไม่เกิน 256 ตัวอักษร
- จบประโยคคำสั่งด้วยเครื่องหมาย “;”
- เครื่องหมาย “;” ใช้แยกประโยควลี
- เครื่องหมายวงเล็บ ใช้สำหรับกำหนดแอททริบิวต์ของคอลัมน์, แสดงการคำนวณทางคณิตศาสตร์
- เครื่องหมาย “'” ใช้ครอบคำที่มีชนิดเป็น string

3.5.2 รูปแบบการใช้คำสั่ง SQL

การใช้คำสั่ง SQL มีรูปแบบมากมายแล้วแต่ความต้องการของผู้ใช้ ในที่นี่จะอธิบายรูปแบบคำสั่ง SQL ที่สำคัญ เช่น CREATE, SELECT, INSERT, DELETE, UPDATE และ คำสั่งเกี่ยวกับการทำทรานแซกชัน

รูปแบบคำสั่งที่ใช้ในการสร้างตาราง	CREATE TABLE <name of base table> (column definition [,column definition] ..);
-----------------------------------	---

ตัวอย่างคำสั่ง SQL ในการสร้างตาราง โดยกำหนดให้ชื่อของตารางคือ department และชื่อของคอลัมน์คือ DEPT_NUM, DEPT_NAME, LOCATION ตามลำดับ คอลัมน์แรกเป็นชนิดตัวเลขสองหลัก คอลัมน์ที่สองและสามเป็นชนิดตัวอักษร ใช้คำสั่งดังนี้

```
SQL> CREATE TABLE DEPARTMENT (
DEPT_NUM    number (2) primary key,
DEPT_NAME   char (10),
LOCATION     char (8) );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำสั่งที่ใช้ในการสืบค้นข้อมูล	SELECT <column list> FROM <name of table>
-------------------------------------	--

ตัวอย่างคำสั่ง SQL ในการสืบค้นข้อมูลในตาราง ชื่อ department โดยสืบค้นคอลัมน์ DEPT_NAME และ LOCATION ใช้คำสั่งดังนี้

```
SQL> SELECT DEPT_NAME, LOCATION
FROM DEPARTMENT;
```

ในกรณีที่ต้องการสืบค้นทุกๆ คอลัมน์ในตาราง สามารถใช้คำสั่งง่ายๆ ดังนี้

```
SQL> SELECT *
FROM DEPARTMENT;
```

นอกจากนี้ เราสามารถสืบค้นข้อมูลเฉพาะแถว (row) ที่เราต้องการ โดยใช้คำสั่งดังนี้

SELECT <column list> FROM <name of table> WHERE <filter conditions>

Where ใช้ในการกำหนดเงื่อนไขในการเรียกดูข้อมูลเฉพาะบางรายการ คำสั่ง Where สามารถมีนิพจน์หรือเงื่อนไขได้มากกว่าหนึ่งโดยใช้ And หรือ Or เป็นตัวเชื่อม หากมีเงื่อนไขไปมากอาจใช้วงเล็บช่วยบอกลำดับในการเปรียบเทียบได้ เช่น

```
SQL> SELECT *
FROM DEPARTMENT
WHERE (DEPT_NUM < 10)
And (LOCATION = 'Bangkok');
```

เครื่องหมาย	ความหมาย
=	เท่ากับ
<>, !=, ^=	ไม่เท่ากับ
>, >=, <, <=	มากกว่า, มากกว่าหรือเท่ากับ, น้อยกว่า, น้อยกว่าหรือเท่ากับ
BETWEEN....AND	ระหว่างค่าสองค่า
LIKE	เหมือนกัน (รูปแบบตัวอักษร)
IS NULL	เป็น null
IN	ค่าในรายชื่อ
ANY	ค่าใดๆในรายชื่อ

ตารางที่ 3.3 เครื่องหมายในภาษา (SQL Operator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้เครื่องหมาย ดังนี้

```
SELECT.... FROM.... WHERE l_name BETWEEN 'A' AND 'D';
WHERE ss_no LIKE '___45%';
WHERE dept IN ('Accounting', 'Sales', 'Personel');
WHERE f_name IN (SELECT f_name FROM female);
ซึ่งมีค่าเท่ากับ WHERE f_name = ANY(SELECT f_name FROM female);
```

รูปแบบคำสั่งที่ใช้ในการใส่ข้อมูลในตาราง `INSERT INTO <table name> [<column list>]`
`VALUES (data values from the columns);`

ตัวอย่างต่อไปนี้เป็นคำสั่งที่ใช้ใส่ข้อมูลลงในตารางที่ชื่อ employee โดยใส่ข้อมูลลงในคอลัมน์ที่ชื่อ ss_no, l_name, f_name และ title ตามลำดับ

```
INSERT INTO employee (ss_no,l_name,f_name,title)
VALUE ('909-10-8910','Parker','Susan C.','Trainee');
```

นอกจากนี้ การใส่ข้อมูลหลายๆแถวลงในตารางสามารถทำได้โดยคัดลอกจากตารางอื่น เช่น

```
INSERT INTO staff
SELECT * FROM employee
WHERE DEPT= 'Sales';
```

รูปแบบคำสั่งที่ใช้ในการลบข้อมูลออกจากตาราง `DELETE FROM <table name>`
`VALUES <filter condition>;`

ตัวอย่างต่อไปนี้เป็นคำสั่งที่ใช้ลบข้อมูลออกจากตารางที่ชื่อ employee โดยแถวที่ลบออกไปคือข้อมูลของพนักงานที่อยู่ในแผนกการขาย หรือแผนกบัญชี

```
DELETE FROM staff
WHERE dept = 'Sales' OR dept = 'Accounting';
```

รูปแบบคำสั่งที่ใช้ในการแก้ไขข้อมูลในตาราง `UPDATE <table name>`
`SET <column name> = <data value>;`

ตัวอย่างต่อไปนี้เป็นคำสั่งที่ใช้ในการแก้ไขข้อมูลในตารางที่ชื่อ payroll โดยแก้ไขข้อมูลในคอลัมน์ที่ชื่อ salary ให้มีค่าเพิ่มขึ้น 5% ของค่าเดิม ในพนักงานที่ชื่อ Baker

```
UPDATE payroll
SET salary = salary * 1.05
WHERE l_name = 'Baker';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ทรานแซกชัน (Transaction)

ในการออกคำสั่ง Insert, Delete, Update ข้อมูลจากตาราง เราจะออกคำสั่งหลายๆ ไม่ได้ ต้องมีการระบุยืนยันคำสั่งหรือยกเลิกคำสั่งโดยใช้คำสั่งเหล่านี้ปิดท้าย

Commit ;ยืนยันคำสั่ง
Rollback ;ยกเลิกคำสั่ง

งานบางงานต้องมีการออกคำสั่งหลายคำสั่งประกอบกัน เพื่อให้รับประกันได้ว่างานแต่ละชิ้นมีคำสั่งทุกคำสั่งสอดคล้องกัน กล่าวคือ หากสำเร็จต้องสำเร็จทุกคำสั่ง มิฉะนั้นก็จะต้องไม่มีคำสั่งใดเลยที่ทำงานได้ เช่นในงานธนาคารที่มีการโอนเงินจากบัญชีหนึ่งไปยังอีกบัญชีหนึ่ง ต้องรับประกันได้ว่าจะต้องไม่มีการสูญหายของเงินระหว่างการโอน จึงต้องมีข้อกำหนด कुछหนึ่งเกิดขึ้นเพื่อรองรับงานประเภทนี้ เราเรียกว่า Transaction

ดังนั้นทุกครั้งที่เราคำสั่ง Insert, Delete, Update จะก็คำสั่งก็ตามจนพอใจแล้ว ต้องปิดท้ายด้วยคำสั่ง Commit (ยืนยันการกระทำคำสั่ง) หรือ Rollback (ยกเลิกคำสั่ง) เสมอ หากระบบเกิดเสียหายในขณะนั้น เมื่อเปิดใช้อีกครั้ง ออราเคิลจะทำการยกเลิกคำสั่งที่ทำค้างไว้

3.7 ระบบรักษาความปลอดภัย

3.7.1 มาตรการรักษาความปลอดภัย (Security Policy)

สิ่งที่สำคัญที่สุดในการติดตั้งระบบรักษาความปลอดภัยก็คือ การกำหนดมาตรการต่างๆ ทางด้านการรักษาความปลอดภัย ว่าเราจะให้ใครบ้างที่สามารถมาทำงานกับข้อมูลที่มีความสำคัญมากกับองค์กรของเรา และเมื่อเข้ามาแล้วจะมีสิทธิ์ทำอะไรได้บ้าง พร้อมทั้งต้องมีส่วนจัดเก็บการทำงานต่างๆ เพื่อใช้เป็นข้อมูลในการตรวจสอบภายหลังได้ เราอาจสรุปสิ่งที่กล่าวมาข้างต้นเป็นข้อๆ ดังนี้คือ

- การกำหนดผู้ที่สามารถเข้ามาใช้งานฐานข้อมูล
- การกำหนดรูปแบบในการใช้งานฐานข้อมูลของผู้ที่เข้ามาในระบบได้
- การตรวจสอบย้อนหลังหากเกิดสิ่งผิดปกติ หรือเกิดความเสียหาย

ในอราเคิล Schema และ User จะบ่งบอกถึงสถานะการเป็นเจ้าของ Object ที่ถูกสร้างอยู่ภายใต้มัน หรืออาจมีความหมายในการจัดกลุ่มของ Object ที่ถูกสร้างขึ้น Object ในที่นี้หมายถึง ตาราง, อินเด็กซ์, วิว ฯลฯ เช่น ตาราง emp ถูกสร้างโดยผู้ใช้ชื่อ Scott เรากล่าวอ้างได้ว่าตารางอยู่ใน Schema ชื่อ Scott

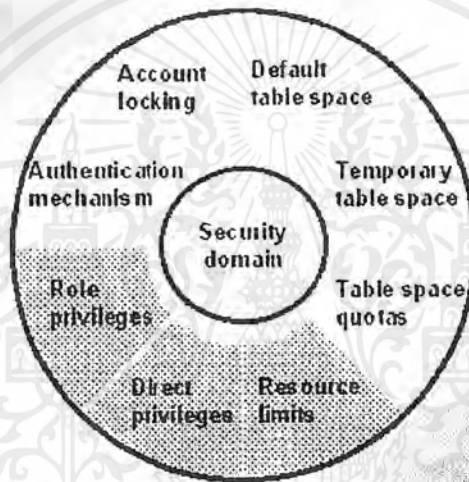
ชื่อผู้ใช้ (Username) ทุกชื่อในฐานข้อมูลจะต้องไม่ซ้ำกัน ทุกๆ ชื่อจะมีหนึ่ง Schema และการที่จะดูว่าใน Schema ของ Scott มี Object อะไรบ้าง ต้องใช้คำสั่ง

```
SQL > select object_name, Object_type
      From all_objects
      Where owner = 'SCOTT';
```

Object ต่างๆ ที่ถูกสร้างขึ้นภายใต้ User หรือ Schema เดียวกันจะต้องตั้งชื่อไม่ซ้ำกันแม้จะเป็นคนละชนิดก็ตาม แต่หากอยู่คนละ Schema ก็สามารถมี Object ที่ชื่อซ้ำกันได้ ทุก Object จะมีชื่อเต็มๆ ว่า Schema.Object เช่น ตารางชื่อ emp ของ Bill เรียกเต็มๆ ว่า Bill.emp

การตั้งชื่อคอลัมน์ในตารางหนึ่งก็ห้ามซ้ำกัน แต่ซ้ำกับชื่อคอลัมน์ในตารางอื่นหรือ Schema อื่นๆ ได้และเวลาอ้างอิงก็อ้างอิงแบบเต็มยศคือ Schema.Table.Column เช่น คอลัมน์ ename ในตารางชื่อ emp ของ Bill เรียกเต็มๆ ว่า Bill.emp.ename

วิว (View) เป็นรูปแบบการเข้าถึงข้อมูลชนิดหนึ่งที่ถูกออกแบบมาให้ผู้ใช้สามารถลดการออกคำสั่งสืบค้นที่ซับซ้อน สร้างระบบการรักษาความปลอดภัยของข้อมูลอย่างง่ายได้ อาจกล่าวได้ว่า วิวเป็น Virtual Table เพราะวิวไม่ได้เก็บข้อมูลอะไร เพียงแค่สร้างโครงกรอบไว้บนตารางอีกทีหนึ่ง กล่าวคือแทนที่เราจะติดต่อกับตารางโดยตรง เราสามารถติดต่อผ่านวิวแทน การใช้งานวิวจะเหมือนกับตารางทุกประการ



รูปที่ 3.6 ระบบรักษาความปลอดภัยที่เกี่ยวข้องกับผู้ใช้ (user)

3.7.2 การให้สิทธิ์และการอนุญาตการใช้งานอ็อบเจ็กต์

การให้ผู้อื่นสามารถใช้งานตารางของเราได้นั้น เราต้องอนุญาตสิทธิ์ให้กับผู้ใช้คนนั้นด้วยคำสั่ง grant สิทธิ์ในการจัดการกับตารางที่สำคัญ มีดังต่อไปนี้

- Select อนุญาตให้สืบค้นข้อมูลในตารางได้
- Alter อนุญาตให้เข้ามาแก้ไขโครงสร้างของ Object ที่ระบุไว้ในคำสั่งได้ เช่น เพิ่มคอลัมน์ให้กับตาราง
- Delete อนุญาตให้เข้ามาลบข้อมูล (Row) ในตารางหรือวิวได้
- Index อนุญาตให้เข้ามาสร้างดัชนีเพื่อช่วยค้นข้อมูลในตารางได้
- Insert อนุญาตให้เข้ามาเพิ่มข้อมูลในตารางหรือวิวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- References อนุญาตให้ใครมาอ้างอิง Primary Key ในตารางเพื่อใช้เป็นตัวอ้างอิงกับ Foreign Key ของอีกตารางได้
- Update อนุญาตให้เข้ามาปรับปรุงข้อมูลในตารางหรือวิวได้
- Execute อนุญาตให้เข้ามา Execute Procedure, Function หรือ Package ของเราได้
- All อนุญาตสิทธิ์ทั้งหมดข้างต้นให้กับผู้ใช้ เป็นการสะดวกในการให้สิทธิ์หลายๆ อย่างพร้อมกันทีเดียว

นอกจากนี้ยังมีสิทธิ์ที่เกี่ยวข้องกับระบบ หรือมีการขอใช้ทรัพยากรระบบที่เรียกว่า System Privileges ด้วย เช่น การสร้างตารางการติดต่อ (connect) เข้ามาใช้งาน การให้สิทธิ์ทำได้โดยใช้คำสั่ง GRANT และการเพิกถอนสิทธิ์ทำโดยใช้คำสั่ง REVOKE

ข้อมูลที่เกี่ยวข้องกับการให้สิทธิ์ต่าง ๆ ของออรากิลนี้จะเก็บอยู่ในคานาดิกชันนารี (Data Dictionary) คำสั่งที่ใช้ดูโครงสร้างของตาราง คือ `SQL>describe user_tab_grants_made`

ออรากิลถูกออกแบบให้มีการจัดกลุ่มสิทธิ์ต่างๆ หรือกลุ่มผู้ใช้เสียก่อน เพื่อให้ง่ายในการดูแล เมื่อเรากำหนดกลุ่มของผู้ใช้หรือของสิทธิ์เรียบร้อยแล้ว จึงให้สิทธิ์ตามกลุ่มของผู้ใช้

ออรากิลมีบทบาท (Role) มาตรฐาน 5 บทบาทที่ให้เราได้ใช้งานดังต่อไปนี้

DBA	สำหรับเจ้าหน้าที่ผู้ดูแลระบบ
CONNECT	สำหรับผู้ใช้ทั่วไป
RESOURCE	สำหรับผู้ใช้ที่เป็นนักพัฒนาระบบ
EXP_FULL_DATABASE	สำหรับเจ้าหน้าที่เครื่องทำสำรองข้อมูล
IMP_FULL_DATABASE	สำหรับเจ้าหน้าที่เครื่องนำคืนข้อมูล (restore)

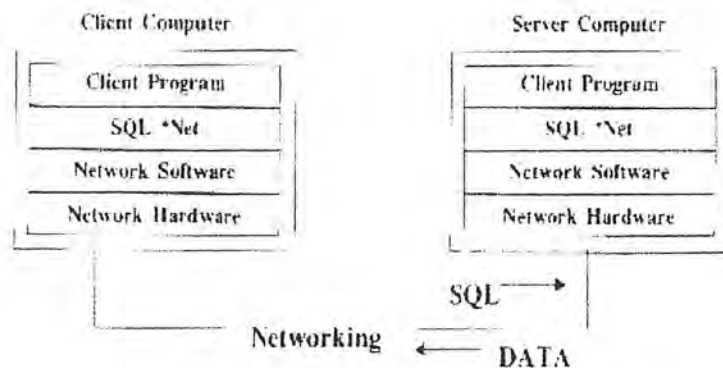
3.8 การเข้าถึงผ่านเครือข่าย (Network Access)

การติดต่อผ่านเครือข่ายเป็นการแยกเครื่องมือ (tool) กับออรากิลให้วิ่งอยู่บนเครื่องคอมพิวเตอร์คนละเครื่อง และใช้เครือข่ายที่ติดตั้งอยู่นั้นเป็นสื่อในการแลกเปลี่ยนข้อมูลซึ่งกันและกัน ทำให้เครื่องแม่ข่ายวิ่งเฉพาะออรากิล และกระจายโปรแกรมต่างๆ ไปยังเครื่องลูกข่ายของผู้ใช้ เพื่อแบ่งเบาภาระของการประมวลผลให้กระจายกันออกไป วิธีการนี้จำเป็นต้องมีการจัดการระบบเครือข่าย และมี SQL*Net เป็นตัวเชื่อมอีกต่อหนึ่ง

จากรูปที่ 3.7 จะเห็นได้ว่าส่วนของเซิร์ฟเวอร์และไคลเอนต์ประกอบด้วยซอฟต์แวร์หลายตัวซ้อนกันเป็นชั้น ซึ่งออกแบบตามโมเดล OSI (7 เลเยอร์) มาตรฐานที่ใช้ติดต่อระหว่างเครื่องคอมพิวเตอร์ในระบบเครือข่ายมีอยู่ด้วยกันหลายชนิด เช่น TCP/IP, Netbios, SPX/SPX เป็นต้น จึงทำให้ออรากิลผลิต SQL*Net หลายๆ แบบเพื่อรองรับชนิดของโปรโตคอลที่ใช้งานอยู่ ทำให้บางครั้งการออกแบบโปรแกรมมีความซ้ำซ้อนกันบางส่วน ต่อมาในระยะหลังออรากิลจึงแบ่ง SQL*Net ออกเป็นสองส่วน ดังนี้

- ส่วนที่เป็น SQL*Net ที่ใช้ติดต่อกับโปรแกรมลูกข่ายหรือโปรแกรมแม่ข่ายออรากิล
- ส่วนที่เป็น Protocol Adapter ที่ใช้ติดต่อกับโปรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 ลักษณะการติดต่อผ่านระบบเครือข่ายของออรากิล

3.9 การใช้งานฐานข้อมูลออรากิล

การเปิดฐานข้อมูล

การเริ่มต้นใช้งานระบบฐานข้อมูลทำได้โดยใช้โปรแกรม Server Manager เรียกใช้ได้โดยคำสั่ง svrmgrl จากนั้นทำการ connect แบบ internal แล้วจึงสามารถสั่งเปิดฐานข้อมูลได้ ถ้า Startup Database สำเร็จจะมีข้อความ “Database opened” แสดงทางหน้าจอ

```
$ svrmgrl
svrmgrl> connect internal
svrmgrl> startup
svrmgrl> exit ; คำสั่งออกจาก Server Manager
```

การปิดฐานข้อมูล

การปิดฐานข้อมูลทำได้โดยใช้โปรแกรม Server Manager เช่นกัน หลังจาก Shutdown Database แล้วจะมีข้อความ “Database closed” แสดงทางหน้าจอ

```
$ svrmgrl
svrmgrl> connect internal
svrmgrl> shutdown
svrmgrl> exit
```

การเปิดปิด TNS Listener

การให้ Database ทำงานแบบ Client-Server เพื่อให้โปรแกรมฝั่งเครื่องลูกข่ายสามารถติดต่อกับฐานข้อมูลได้นั้น ต้องมีการ Start Process ขึ้นมาอีก 1 process ซึ่งก็คือ TNS Listener นั่นเอง

```
คำสั่งเปิด TNS Listener      $ lsnrctl start
คำสั่งปิด TNS Listener       $ lsnrctl stop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้าโปรแกรม SQL Plus

การใช้คำสั่ง SQL สามารถทำได้ โดยเรียกโปรแกรม SQL Plus ซึ่งรูปแบบคำสั่งที่ใช้เป็นดังนี้

คำสั่งเรียก โปรแกรม SQL Plus `sqlplus`

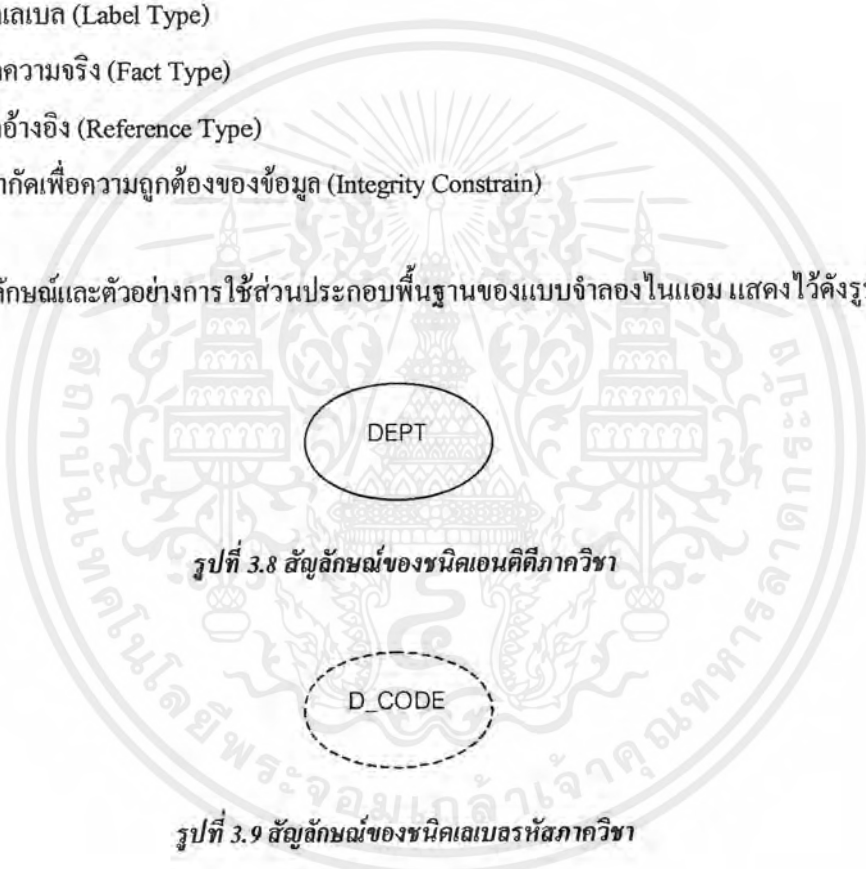
คำสั่งออกจากโปรแกรม SQL Plus `sqlplus> exit`

3.10 การออกแบบฐานข้อมูลด้วยวิธีในแอม

ในแอม (NIAM – Nijssens 's Information Analysis Methodology) เป็นวิธีการออกแบบ แบบ bottom up ทำให้โมเดลที่ออกแบบไว้ไม่มีความซ้ำซ้อน และเป็นไปตามกฎของ ฐานข้อมูลสัมพันธ์ (Relational database) ในแอมโมเดลจะมีองค์ประกอบพื้นฐานดังนี้ คือ

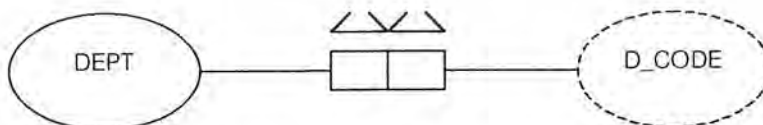
- ชนิดเอนติตี (Entity Type)
- ชนิดเลเบล (Label Type)
- ชนิดความจริง (Fact Type)
- ชนิดอ้างอิง (Reference Type)
- ข้อจำกัดเพื่อความถูกต้องของข้อมูล (Integrity Constraint)

สัญลักษณ์และตัวอย่างการใช้ส่วนประกอบพื้นฐานของแบบจำลองในแอม แสดงไว้ดังรูป



รูปที่ 3.8 สัญลักษณ์ของชนิดเอนติตีภาควิชา

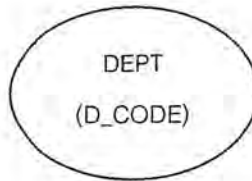
รูปที่ 3.9 สัญลักษณ์ของชนิดเลเบลรหัสภาควิชา



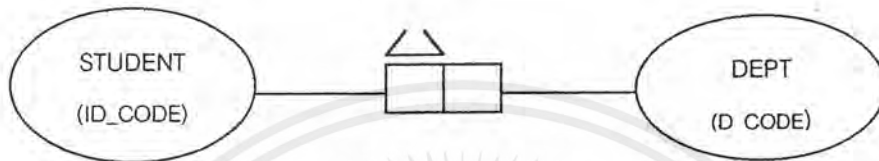
รูปที่ 3.10 ความสัมพันธ์อ้างอิงแบบ one to one

จากรูป หมายความว่า ภาควิชาใดๆ จะมีรหัสภาควิชาเพียงรหัสเดียวเท่านั้นและไม่ซ้ำกับภาควิชาอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

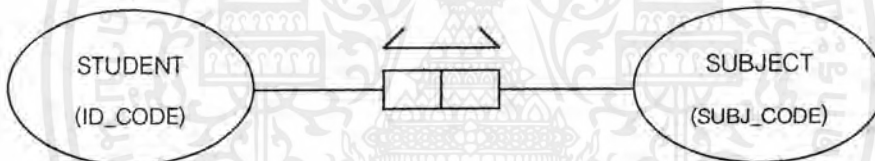


รูปที่ 3.11 ความสัมพันธ์อ้างอิงแบบ *one to one* อย่างย่อ



รูปที่ 3.12 ความสัมพันธ์อ้างอิงแบบ *many to one*

จากรูป หมายความว่า นักศึกษาหนึ่งคนจะสังกัดภาควิชาได้เพียงภาคเดียว แต่ภาควิชาใดๆ แล้ว สามารถมีนักศึกษาในสังกัดได้มากกว่าหนึ่งคน



รูปที่ 3.13 ความสัมพันธ์แบบอ้างอิงแบบ *many to many*

จากรูป หมายความว่า นักศึกษาหนึ่งคนสามารถลงทะเบียนเรียนได้หลายวิชา และแต่ละวิชาที่เปิดสอนสามารถรับจำนวนนักศึกษาได้มากกว่าหนึ่งคน แต่นักศึกษาที่ลงทะเบียนเรียนวิชาใดๆ แล้ว จะลงทะเบียนซ้ำวิชาเดิมไม่ได้

ชนิดเอนทิตีเป็นเซต (set) ซึ่งมีสมาชิกเป็นตัวอย่างเอนทิตี (Entity Instance) ส่วนเครื่องหมายความสัมพันธ์ที่เป็นส่วนเชื่อมโยงระหว่างชนิดเอนทิตี และชนิดเอนทิตี หรือชนิดเลเบลนั้น เรียกว่า บทบาท (role) โดยสามารถเขียนความหมายของบทบาทนั้นไว้ภายในหรือข้างๆ สัญลักษณ์ของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

เว็ลด์ไวด์เว็บ (World Wide Web)

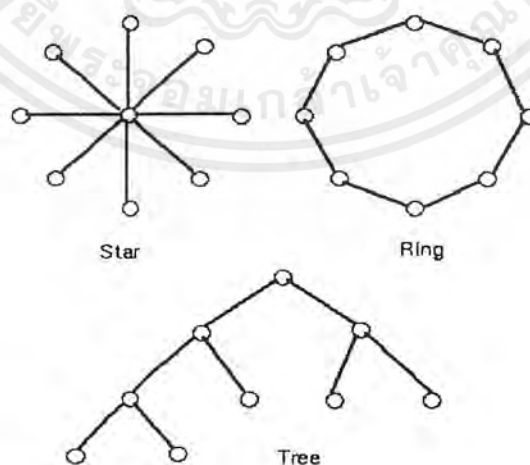
4.1 เทคโนโลยีเครือข่ายอินเทอร์เน็ต

เครือข่ายอินเทอร์เน็ต คือ เครือข่ายของคอมพิวเตอร์ หรืออุปกรณ์ต่างๆ ที่สื่อสารกันในระดับ Network layer ด้วยโปรโตคอล IP การเชื่อมต่อคอมพิวเตอร์เข้าด้วยกัน จะมีจุดประสงค์เพื่อสามารถให้ใช้ทรัพยากรร่วมกัน ผู้ที่ใช้งานอยู่บนคอมพิวเตอร์เครื่องหนึ่ง จะสามารถใช้บริการทรัพยากรของเครื่องคอมพิวเตอร์อีกเครื่องหนึ่งซึ่งอยู่ห่างไกลออกไปได้ สำหรับจุดมุ่งหมายอื่น เช่น เพื่อเน้น ความน่าเชื่อถือของระบบ และการลดค่าใช้จ่ายลง

4.1.1 โครงสร้างของเครือข่าย

ภายในเน็ตเวิร์กจะมี เครื่องคอมพิวเตอร์ประเภทต่างๆ ที่ทำหน้าที่ตามคำสั่งของผู้ใช้จะเรียกว่า Host (end system) ซึ่งระบบแรกๆจะถูกเชื่อมผ่านซับเน็ต แต่ในระยะหลังเนื่องจากมีขนาดเครือข่ายกว้างขวางขึ้น รวมทั้งมีความซับซ้อนของการสื่อสารข้อมูลมากขึ้น จะมีระบบสลับข้อมูลเข้ามาช่วยในการสื่อสารข้อมูลระบบสลับข้อมูลนี้จะเรียกว่า IMP (Interface Message processor) บางทีอาจจะเรียกว่า package switching node หรือ intermediate system ปกติรูปแบบการเชื่อมต่อจะเป็นแบบจุดถึงจุด (point to point) และแบบกระจาย (broadcast channel)

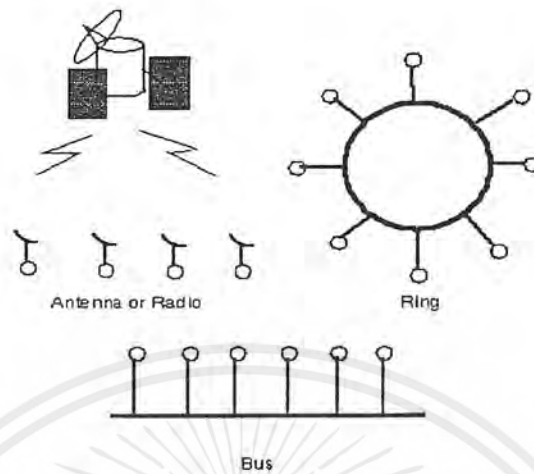
การเชื่อมต่อแบบจุดถึงจุดประกอบด้วยสายเคเบิล หรือสายสื่อสารเชื่อมต่อระหว่าง IMP ของแต่ละจุดเข้าด้วยกันในกรณีต้องการจะทำการสื่อสารระหว่าง IMP ซึ่งไม่มีสายสื่อสารเชื่อมต่อเข้าด้วยกัน จะต้องทำการส่งผ่าน IMP ตัวอื่นๆที่อยู่ใกล้เคียงแทน เมื่อกลุ่มของข้อมูล (message หรือ packet) ถูกส่งจาก IMP ออกมาแล้ว IMP ตัวกลางจะทำการเก็บแพคเกจเก็บเอาไว้ รอจนกระทั่งสายที่ใช้ส่งว่างแล้วจึงส่งแพคเกจออกไป วิธีแบบนี้จะถูกเรียกว่า store and forward หรือ packet switched



รูปที่ 4.1 การเชื่อมต่อแบบจุดถึงจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อแบบกระจาย จะมีช่องทางเชื่อมสัญญาณเพียงช่องทางเดียว และต้องใช้ร่วมกับเนตเวิร์กอื่นๆ เมื่อแพคเกจถูกส่งออกไปแล้ว IMP อื่นๆจะได้รับสัญญาณในเวลาใกล้เคียงกัน เมื่อ IMP ได้รับสัญญาณแล้วจะตรวจสอบว่าเป็นแพคเกจของตนหรือไม่ ได้จาก address ภายในแพคเกจนั้น หากไม่ตรงกับ address ของตนเอง แพคเกจนั้นก็จะถูกทิ้งไป



รูปที่ 4.2 การเชื่อมต่อแบบกระจาย

International Standard Organization (ISO) กำหนดมาตรฐานการสื่อสารข้อมูลบนเครือข่ายระหว่างจุดปลาย 2 จุดใดๆ ที่เรียกว่า OSI (Open System Interconnection) ซึ่งจะมีทั้งหมด 7 เลเยอร์ ดังนี้

เลเยอร์ 1 : Physical layer

เลเยอร์นี้จะเกี่ยวข้องกับการส่งข้อมูลระดับบิต ผ่านช่องสื่อสารข้อมูล โดยการออกแบบจะต้องแน่ใจว่าจะสามารถส่งข้อมูลออกไป และปลายทางจะต้องรับข้อมูลนั้นได้อย่างถูกต้อง โดยส่วนใหญ่จะเป็นข้อกำหนดเกี่ยวกับเรื่องของแรงดันไฟฟ้า ว่าจะต้องใช้แรงดันเท่าไรสำหรับแทนเลข "1" และเท่าใดสำหรับแทนเลข "0" ระยะเวลาในการส่งแต่ละบิตจะต้องห่างกันเท่าใด รวมถึงรูปแบบของคอนเนคเตอร์ วิศวกรไฟฟ้าจะเกี่ยวข้อง โดยตรงกับเลเยอร์ในลำดับขั้นนี้

เลเยอร์ 2 : Data link layer

จุดประสงค์หลักของเลเยอร์นี้คือ จะทำการควบคุมการส่งข้อมูลคิบบให้เหมือนกับว่า ไม่มี ข้อผิดพลาดเกิดขึ้น ทำให้เลเยอร์ในลำดับถัดไปไม่ต้องสนใจเรื่องนี้ วิธีการคือจะต้อง ทำการแตกข้อมูลออกเป็นก้อนๆเรียกว่า เฟรมข้อมูล (data-frame) แล้วทำการส่งออกไปทีละชุด และรอการตอบรับ (acknowledge frame) กลับมา

ในกรณีที่มีสัญญาณรบกวน ทำให้สัญญาณขาดหายไป จะต้องมีการบอกให้เครื่องต้นทางทำการส่งข้อมูลที่หายไปนั้นกลับมาให้ใหม่ นอกจากนี้จะต้องมีการหักข้อมูลไว้ในบัฟเฟอร์ หากความเร็วในการส่งข้อมูลของทั้งสองฝ่ายไม่เท่ากัน

เลเยอร์ 3 : Network layer

หน้าที่หลักของเลเยอร์นี้ จะเกี่ยวข้องกับการหาเส้นทาง (route) เพื่อพิจารณาว่าแพคเกจจะถูกส่งจากต้นทาง ไปยังปลายทาง ได้อย่างไร การกำหนดเส้นทางอาจจะกำหนดตั้งแต่เริ่มต้นติดต่อกัน หรืออาจจะสามารถ เปลี่ยนแปลงตลอดเวลา (dynamic) ก็ได้ นอกจากนี้ก็มีเรื่องที่จะต้องพิจารณาหากมีการส่งข้อมูลข้ามเน็ตเวิร์ก แล้วมีความแตกต่างระหว่างเน็ตเวิร์ก หรือใช้โปรโตคอลแตกต่างกัน เน็ตเวิร์กเลเยอร์จะต้องทำการจัดการกับปัญหาเหล่านี้เพื่อให้แต่ละเน็ตเวิร์กสามารถเชื่อมต่อกันได้ เสมือนเป็นเน็ตเวิร์กเดียวกัน

เลเยอร์ 4 : Transport layer

เลเยอร์นี้จะคอยทำการติดต่อกับเลเยอร์ถัดไป (session layer) เพื่อคอยแยกข้อมูล ให้มีขนาดพอเหมาะและส่งต่อไปกับ network layer พร้อมทั้งตรวจสอบว่าข้อมูลได้ถูกส่งไปถึงยังปลายทางได้อย่างเรียบร้อยหรือไม่ โดยเลเยอร์นี้จะต้องจัดการได้อย่างมีประสิทธิภาพเพื่อเป็นการแยกให้ session layer เป็นอิสระจากการ เปลี่ยนแปลงทางด้านฮาร์ดแวร์

เลเยอร์ 5 : Session layer

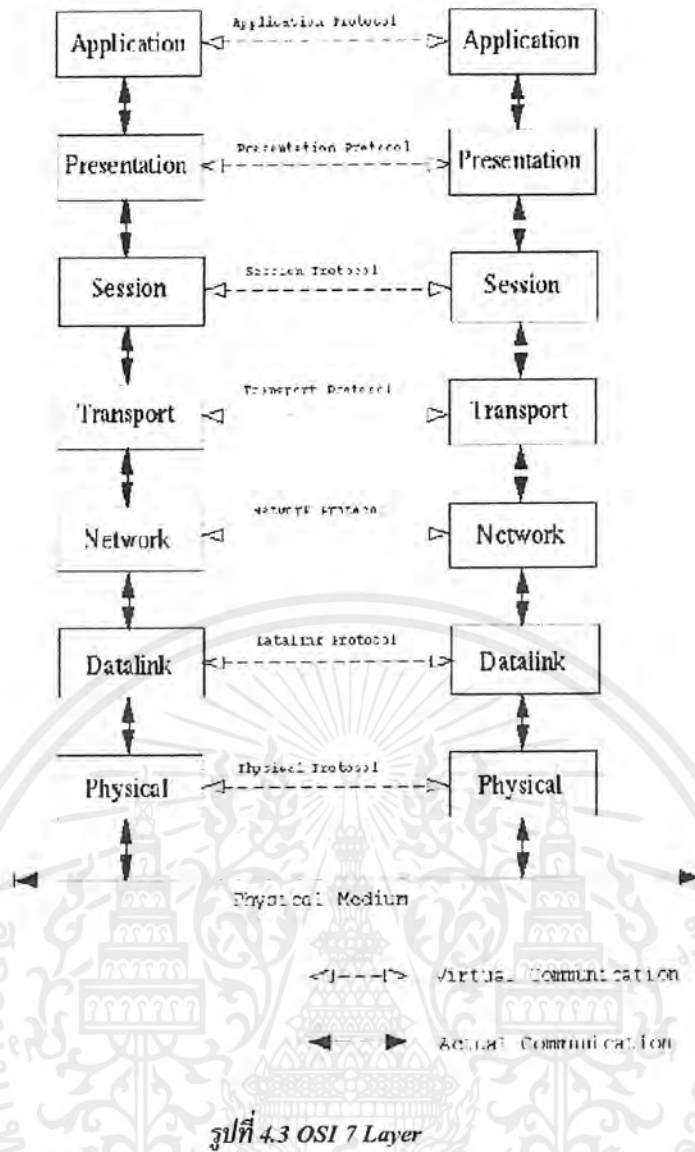
นอกจากจะทำการส่งข้อมูลแบบเดียวกับ transport layer แล้ว ยังมีการให้บริการอื่นๆเช่น การยอมให้ผู้ใช้งานเข้าไปใช้งานยังเครื่องที่อยู่ห่างไกลออกไป (remote login) หรือทำการถ่ายโอนไฟล์ระหว่างเครื่อง นอก จากนี้ยังทำหน้าที่เกี่ยวกับการซิงโครไนซ์เซชัน (synchronization) หรือทำให้สองระบบทำงานสัมพันธ์กัน

เลเยอร์ 6 : Presentation layer

เลเยอร์นี้จะสนใจในเรื่องของรูปแบบของข้อมูล เช่นการเปลี่ยนรหัสข้ามจาก ASCII เป็นรหัส EBCDIC เพื่อให้คอมพิวเตอร์ที่มีการแทนรหัสต่างกันสามารถสื่อสารกันได้ นอกจากนี้ก็อาจทำการลดขนาดของข้อมูล (data compression) หรือทำการเข้ารหัสของข้อมูล (data encryption) เพื่อป้องกันการโจรกรรมข้อมูลได้ด้วย

เลเยอร์ 7 : Application layer

เลเยอร์บนสุดจะเกี่ยวข้องกับ โปรโตคอลมากมาย ซึ่งจะมีการใช้งานที่แตกต่างกัน โดยเฉพาะ ตัวอย่างเช่นการควบคุมเทอร์มินอลชนิดต่างๆ รูปแบบการแสดงผลทางจอภาพอาจมีความแตกต่างกัน ก็อาจมีการกำหนดเทอร์มินอลเสมือน เพื่อเป็นตัวกลางในการควบคุมการทำงานของเทอร์มินอลจริงๆ (คล้ายกับ java bytecode เป็นรหัสกลางสำหรับ java compiler จะทำการตีความให้สามารถทำงานที่เครื่องต่างชนิดกันได้)



4.1.2 ความรู้เบื้องต้นเกี่ยวกับ TCP/IP

การจะใช้งาน TCP/IP ได้นั้นจำเป็นจะต้องมีการกำหนดค่าของ IP address หรือหมายเลขประจำตัวให้กับเครื่องคอมพิวเตอร์แต่ละเครื่องในเครือข่าย ซึ่งจะคล้ายกับเลขที่บ้านของคุณ ที่จะทำหน้าที่บอกตำแหน่งที่อยู่ของคุณให้คนอื่นๆ ได้รับทราบ

นอกจากนี้ก็ยังจะมีค่าอื่นๆอีก แต่ที่จำเป็นอีกอย่างก็คือค่าของ MTU หรือ Maximum Transfer Unit ซึ่งค่านี้ก็จะบอกถึงจำนวนสูงสุดของข้อมูล (datagram) ที่จะใช้ทำการประมวลผลของโปรแกรม

โปรโตคอลของ TCP/IP

โปรโตคอลของ TCP/IP จะแบ่งออกเป็นสองส่วน คือส่วนของ IP (Internet Protocol) และส่วนของ TCP (Transport Control Protocol)

โปรโตคอลในส่วนของ IP จะรับผิดชอบเกี่ยวกับการเชื่อมต่อระหว่างระบบ ซึ่งสามารถเทียบได้กับ Network layer ในโมเดลของ OSI และไม่มีการรับประกันการสูญหายของข้อมูล (หน้าที่นี้จะเป็นของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอลในระดับสูงขึ้นไป) นอกจากนี้โปรโตคอล IP ยังสามารถแตกข้อมูลออกเป็นชิ้นย่อยๆ และประกอบข้อมูลย่อยเหล่านั้นกลับคืนได้อย่างถูกต้อง ณ จุดรับข้อมูลปลายทาง

โปรโตคอลในส่วนของ TCP จะมีอยู่สองแบบ คือ TCP (Transport Control Protocol) และ UDP (User Datagram Protocol) โปรโตคอลในส่วนนี้จะเทียบได้กับ Session layer และ Transport layer ในโมเดลของ OSI ข้อมูลที่เกิดจากโปรโตคอลในเลเยอร์นี้จะประกอบด้วยสองส่วนคือ ข้อมูลในส่วนหัว (header) และเป็นข้อมูลที่เป็นเนื้อข้อมูลจริงๆ (data area) โดยข้อมูลในส่วนหัวจะประกอบด้วยที่อยู่, ความยาวของข้อมูล และตัวเลขตรวจสอบความถูกต้อง (checksum) โปรโตคอลแบบ UDP จะส่งข้อมูลแบบไม่รับประกันการสูญหายของข้อมูลในการรับส่งระหว่างโฮสต์ ในขณะที่ TCP จะรับประกันการสูญหายของข้อมูลในระหว่างหารรับส่งข้อมูลระหว่างโฮสต์ให้ด้วย

IP Address

IP address เป็นตัวเลขความยาว 32 บิต และจะต้องกำหนดให้แต่ละเครื่องมีความแตกต่างกัน ถ้าคุณใช้งานเนตเวิร์กของคุณเพียงแค่ในหน่วยงานของคุณเอง ในกรณีนี้คุณสามารถกำหนดหมายเลขของ IP ให้กับเครื่องต่างๆของคุณเองได้ แต่ถ้าหากเป็นเนตเวิร์กที่ติดต่อกับสาธารณะเมื่อใด จะต้องมีการกำหนดมาจากหน่วยงานที่ทำการควบคุมหมายเลข IP โดยเฉพาะซึ่งจะเรียกว่า the Network Information Center (NIC)

หมายเลข IP จะถูกแบ่งออกเป็นตัวเลข 8 บิตจำนวน 4 ส่วน (octets) ตัวอย่างเช่น IP address 0x954C0C04 จะถูกเขียนอย่างง่ายได้เป็น 149.76.12.4 การเขียนแบบนี้จะเรียกว่า dotted quad notation ประโยชน์อีกอย่างของการเขียนแบบแบ่งเป็นตัวเลข 4 ชุดนี้คือเพื่อแสดงถึง หมายเลขของเนตเวิร์ก และหมายเลขของโฮสต์ โดยปกติแล้วมีการแบ่งเนตเวิร์กออกเป็นขนาดต่างๆ ซึ่งจะมีผลต่อจำนวนของโฮสต์ที่มีอยู่ได้ ในเนตเวิร์กแต่ละชนิดด้วย

Class A

Class A ประกอบด้วยเนตเวิร์กหมายเลข 0.0.0.0 จนถึง 127.255.255.255 หมายเลขของโฮสต์จะประกอบด้วยตัวเลข 24 บิต มีเนตเวิร์กย่อยได้ 128 เนตเวิร์ก และมีโฮสต์ถึง 1.6 ล้าน โฮสต์

Class B

Class B ประกอบด้วยเนตเวิร์กหมายเลข 128.0.0.0 จนถึง 191.255.255.255 หมายเลขของโฮสต์จะมี 16 บิต มีเนตเวิร์กย่อยได้ 16,320 เนตเวิร์ก และมีโฮสต์ได้ 65,024 โฮสต์ สำหรับแต่ละเนตเวิร์ก

Class C

Class C ประกอบด้วยเนตเวิร์กตั้งแต่ 192.0.0.0 จนถึง 223.255.255.255 จะมีเนตเวิร์กได้จำนวน 2 ล้านเนตเวิร์ก ซึ่งแต่ละเนตเวิร์กจะมีจำนวนโฮสต์ได้ 254 โฮสต์

Class D, E และ F

มีแอดเรสตั้งแต่ 224.0.0.0 ถึง 254.0.0.0 อยู่ในระหว่างการทดลองและสำรองไว้เพื่ออนาคต ไม่ได้กำหนดจำนวนเนตเวิร์กใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลาสของเครือข่าย	แอดเดรสที่น้อยที่สุด	แอดเดรสที่สูงที่สุด
A	0.0.0.0	127.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255
D	244.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

ตารางที่ 4.1 ขอบเขตหมายเลขแอดเดรสของเครือข่ายในแต่ละคลาส

ตัวอย่างเช่น IP หมายเลข 149.76.12.4 จะหมายถึงโฮสต์ 12.4 บนเครือข่าย Class B หมายเลขที่ 149.76.0.0 ในกรณีหมายเลขเป็น 149.76.255.255 จะหมายถึงโฮสต์ทุกตัวที่อยู่บนเครือข่ายหมายเลข 147.76.0.0

มีเครือข่ายอีกสองหมายเลขที่ถูกสำรองไว้ใช้งานเฉพาะด้านคือ เครือข่ายหมายเลข 0.0.0.0 (default root) และ 127.0.0.0 (loopback address) loopback จะเป็นช่องทางพิเศษที่ใช้ติดต่อกับโฮสต์เครื่องนั้นเอง บางทีจะใช้สำหรับการทดสอบเครือข่ายบางอย่าง โดยไม่จำเป็นต้องต่อกับเครือข่ายจริง

เน็ตมาสก์ (Net Mask)

เน็ตมาสก์เป็นตัวบอกให้เราทราบได้ว่า IP address ใดอยู่ในเครือข่ายเดียวกันบ้าง ตัวอย่างเช่น ถ้าหากเน็ตมาสก์มีค่าเป็น 255.255.255.0 ดังนั้นโฮสต์ที่มีหมายเลข IP เป็น 149.108.10.1 ที่มีค่าเน็ตมาสก์เป็นค่าข้างต้น เรารู้ว่าโฮสต์ตัวนี้จะอยู่ในเครือข่ายเดียวกับ โฮสต์ตัวอื่นๆที่มีหมายเลข IP ขึ้นต้นด้วย 149.108.10

เกตเวย์ (Gateway)

เกตเวย์เป็นหนทางในการเชื่อมต่อระหว่างเครือข่ายสองเครือข่ายเข้าด้วยกัน สำหรับในกรณีที่โฮสต์ที่อยู่ในเครือข่ายที่เป็นซับเน็ต (subnet : เครือข่ายย่อย) เดียวกัน โฮสต์ต่างๆเหล่านั้นจะสามารถทำการติดต่อกันได้ แต่ถ้าหากจะต้องการติดต่อกับโฮสต์ที่อยู่คนละซับเน็ตกันแล้วจำเป็นจะต้องอาศัยเกตเวย์เข้ามาช่วย เกตเวย์จะเปรียบเสมือนเป็นโฮสต์ที่เป็นที่รับรู้ว่าทั้งองซับเน็ต และจะคอยทำการส่งผ่านข้อมูลระหว่างซับเน็ตทั้งสองไปมา สำหรับการหาเส้นทางการสื่อสารของเครือข่ายจำเป็นจะต้องอาศัยเครื่องหาเส้นทาง (router) เข้ามาช่วย

ในกรณีที่เครือข่ายมีขนาดใหญ่และซับซ้อนมากขึ้น หน้าที่ของเกตเวย์ก็จะมีมากขึ้นตามไปด้วย ซึ่งนอกจากจะช่วยส่งข้อมูลข้ามเครือข่ายไปมาแล้ว ยังต้องตัดสินใจหาเส้นทางที่ดีที่สุดในการส่งผ่านข้อมูลอีกด้วย ถึงแม้ว่าเกตเวย์จะทำหน้าที่ซับซ้อนและต้องส่งผ่านข้อมูลเป็นจำนวนมากจนดูเหมือนว่าคอมพิวเตอร์ที่ทำหน้าที่เป็นเกตเวย์จะต้องมีความจุของฮาร์ดดิสก์และหน่วยความจำขนาดใหญ่มาก แต่ในความเป็นจริงแล้วคอมพิวเตอร์ที่ใช้เป็นเกตเวย์นั้นเราสามารถเลือกใช้คอมพิวเตอร์ขนาดเล็กๆ ได้ เนื่องจากการออกแบบเกตเวย์นั้นมีลักษณะที่อยู่ที่เกตเวย์จะส่งผ่านข้อมูลไปยังเน็ตเวิร์กปลายทาง ไม่ใช่

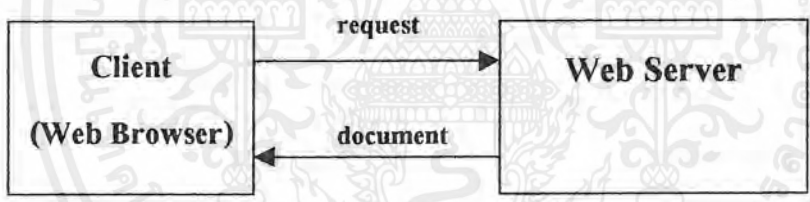
โสตป์ปลายทาง และจากการที่เกิดเวทย์ให้ความสนใจกับเน็ตเวิร์กแทนที่จะเป็น โสตต์นั้นก็มีส่วนทำให้ข้อมูลที่เกิดเวทย์คุณเองมีสัดส่วนน้อยลงไปด้วย

โดเมนเนมเซิร์ฟเวอร์ (Domain Name Server : DNS)

โดเมนเนมเซิร์ฟเวอร์เป็นระบบของการแปลงกลับไปกลับมาระหว่างชื่อ โสตต์ให้เป็น IP Address ซึ่งจะทำหน้าที่คล้ายกับสมุดโทรศัพท์ ซึ่งเราสามารถจะเปิดหาชื่อของบุคคลที่เราต้องการจะติดต่อ เพื่อค้นหาเบอร์โทรศัพท์ของเขาได้ ในทำนองเดียวกันหมายเลข IP address มีลักษณะเป็นตัวเลขที่มนุษย์ค่อนข้างจะจดจำได้ยาก จึงมีการเปลี่ยนมาใช้เป็นตัวอักษรสำหรับแทนหมายเลข IP address ต่างๆ และมีเครื่องเซิร์ฟเวอร์ ที่คอยทำหน้าที่บริการเปลี่ยนจากตัวเลข IP ให้กลายเป็นตัวอักษรได้

4.2 เว็บเซิร์ฟเวอร์

เวิลด์ไวด์เว็บ (World Wide Web - WWW) เป็นบริการอันหนึ่งที่มีอยู่ในอินเทอร์เน็ต ทำให้การใช้งานอินเทอร์เน็ตเป็นสิ่งที่ง่ายขึ้น ข่าวสาร ข้อมูล หรือ เอกสารที่เก็บอยู่ในอินเทอร์เน็ตจะอยู่ในรูปแบบของสื่อผสม (Multimedia) ที่เรียกว่า “เว็บเพจ” (web page) ซึ่งประกอบไปด้วยตัวอักษรข้อความที่มีรูปแบบต่างๆ เช่น รูปภาพ เสียง ภาพเคลื่อนไหว วิดีโอ และ ไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นการแสดงข้อความที่มีการเชื่อมโยงถึงกันได้ โดยผู้ใช้สามารถดูเอกสารหนึ่งจากเอกสารหนึ่งได้ ซึ่งเป็นลักษณะการเชื่อมต่อที่คล้ายกับใยแมงมุม จึงเรียกบริการ WWW ว่าเป็นเครือข่ายใยแมงมุม



รูปที่ 4.4 การให้บริการเวิลด์ไวด์เว็บ

การใช้บริการเวิลด์ไวด์เว็บได้นั้นจำเป็นต้องมีส่วนประกอบ 2 ส่วน ดังนี้

1. แหล่งข้อมูลหรือเว็บเซิร์ฟเวอร์

เว็บเซิร์ฟเวอร์เป็นระบบคอมพิวเตอร์ที่เก็บเว็บเพจ ซึ่งผู้ให้บริการสามารถเรียกดูเว็บเพจที่เก็บอยู่ในเว็บไซต์นั้นได้ เครื่องที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ต้องมีการติดตั้งโปรแกรมการจัดการ ซึ่งจะทำการจัดการคอมพิวเตอร์เครื่องนั้นทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ เช่น ในกรณีที่เครื่องคอมพิวเตอร์เครื่องนั้นใช้ระบบปฏิบัติการเป็นยูนิกซ์ อาจใช้เว็บเซิร์ฟเวอร์เป็น NCSA httpd, Apache เป็นต้น

2. โปรแกรมเว็บเบราว์เซอร์ (Web Browser)

เป็นซอฟต์แวร์ที่ใช้ในการเข้าสู่ WWW และเปิดดูเว็บเพจที่เก็บอยู่ในเว็บไซต์ใดๆ นอกจากนี้ ยังมีความสามารถอื่นๆ เช่น การค้นหาข้อมูล การบริการส่งจดหมายอิเล็กทรอนิกส์ เป็นต้น ซอฟต์แวร์เหล่านี้ ได้แก่ Netscape Communications, Microsoft Internet Explorer, Opera เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่นำมาใช้เป็นเว็บเซิร์ฟเวอร์ มีให้เลือกใช้ได้หลายชนิด เช่น

1. IIS (Internet Information Server) ซึ่งทำงานบน Windows NT สามารถติดตั้งพร้อมระบบปฏิบัติการได้เลย
2. Netscape Enterprise Server 3
3. Oracle Web Server
4. Net Object Cold Fusion
5. Apache Web Server

4.2.1 โครงสร้างของเอกสารเลขที่เอ็มแอล (HTML)

โฮมเพจและเว็บเพจใดๆ ส่วนมากจะสร้างด้วยภาษาที่เรียกว่า HTML (Hyper Text Markup Language) การสร้างเว็บเพจ นอกจากจะสร้างโดยการเขียน HTML บนโปรแกรมอิดิตเตอร์ตัวใดตัวหนึ่ง เช่น Notepad บนวินโดวส์แล้วยังสามารถใช้โปรแกรมช่วยสร้างเว็บเพจ โดยที่ผู้สร้างไม่จำเป็นต้องรู้ HTML แต่อย่างใด ในปัจจุบันมีโปรแกรมช่วยสร้างเว็บเพจหลายตัว เช่น โปรแกรม Microsoft FrontPage เป็นต้น

เอกสาร HTML ประกอบด้วยคำสั่งที่เรียกว่า "Tags" ซึ่งอยู่ภายใต้เครื่องหมาย '<' และ '>' และมีข้อความที่เป็นแท็กซ์ทั่วไป เพื่อบอกให้บราวเซอร์ทราบว่าจะต้องแสดงผลออกมาเป็นอย่างไร

โครงสร้างของ HTML มี 2 ส่วน คือ ส่วนที่เป็นส่วนหัว (Head) และส่วนที่เป็นเนื้อหา (Body) ดัง

```
<HTML>
<HEAD>
<TITLE> ชื่อ โปรแกรมหรือหัวเรื่องที่จะแสดง</TITLE>
</HEAD>
<BODY>
.....
คำสั่งหรือข้อความที่ต้องการแสดง
.....
</BODY>
</HTML>
```

4.2.2 ซีจีไอ (Common Gateway Interface-CGI)

ซีจีไอเป็นโปรโตคอลที่ใช้ควบคุมการส่งข้อมูลระหว่างเว็บเบราว์เซอร์ และเซิร์ฟเวอร์เพื่อให้ได้ตอบกันในลักษณะ 2 ทิศทาง ข้อดีของซีจีไอ คือ ช่วยเพิ่มความสามารถให้กับเว็บเพจ โดยสามารถโต้ตอบกับผู้ใช้ได้

โปรแกรมที่ถูกเขียนขึ้นตามมาตรฐานของโปรโตคอลซีจีไอนี้ เรียกว่า ซีจีไอสคริปต์ (CGI Script) ซึ่งทำให้ผู้ใช้สามารถส่งข้อมูลโต้ตอบกับเว็บเซิร์ฟเวอร์ได้ และอนุญาตให้เว็บเบราว์เซอร์สามารถสอบถาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือค้นหาข้อมูลจากฐานข้อมูลที่เก็บอยู่ในเครื่องเซิร์ฟเวอร์ได้ ซีจีไอสามารถเขียนได้หลายภาษา ได้แก่ Perl, C, JAVA เป็นต้น

ส่วนประกอบของซีจีไอมีขั้นตอนใหญ่ ๆ ดังนี้

1. สร้างฟอร์มรับข้อมูลจากผู้ใช้โดยใช้แท็กของ HTML สำหรับการสร้างแบบฟอร์มและปุ่ม submit เมื่อป้อนข้อมูลเสร็จ
2. เขียนซีจีไอสคริปต์เก็บไว้ที่เซิร์ฟเวอร์

การส่งข้อมูลจากแบบฟอร์มให้กับซีจีไอสคริปต์ ต้องกำหนดรูปแบบดังนี้

`< FORM METHOD="GET" หรือ "POST" ACTION="URL">`

ซึ่งเมื่อผู้ใช้กดปุ่ม submit เครื่องเซิร์ฟเวอร์จะค้นหาสคริปต์ที่มีชื่อตาม URL

GET และ POST เป็นวิธีการที่เซิร์ฟเวอร์นำข้อมูลจากแบบฟอร์มมาส่งให้กับซีจีไอสคริปต์

POST : ส่งมาทีละ Character เรียงต่อแถวกันในรูปของ Standard input (STDIN)

GET : รับข้อมูลเก็บไว้ใน Environment Variable ชื่อ QUERY_STRING

4.3 อาปาเช่เว็บเซิร์ฟเวอร์ (Apache Web Server)

โครงการนี้ใช้โปรแกรมอาปาเช่เว็บเซิร์ฟเวอร์ทำหน้าที่ควบคุมการให้บริการ โฮมเพจซึ่งสามารถติดตั้งโปรแกรมนี้ในขั้นตอนการติดตั้ง Red Hat Linux โดยกาดำเลือกเว็บเซิร์ฟเวอร์ลงไป ไคลเรกทอรีหลักของเว็บเซิร์ฟเวอร์ที่ใช้ในการเก็บโฮมเพจจะอยู่ที่ /home/httpd โดยภายใน ไคลเรกทอรี httpd นี้จะมีอีก 3 ไคลเรกทอรี คือ

/home/httpd/html	เก็บโฮมเพจของ Server
/home/httpd/cgi-bin	เก็บไฟล์โปรแกรม CGI
/home/httpd/icons	เก็บไฟล์รูปภาพไอคอนต่าง ๆ

อาปาเช่เว็บเซิร์ฟเวอร์จะกำหนดไฟล์แรกของโฮมเพจ เป็นไฟล์ชื่อ index.html ซึ่งถ้ามีผู้ใดเข้ามาเปิดโฮมเพจของ Server นี้ ไฟล์ index.html จะถูกเปิดเป็นไฟล์แรก

ไฟล์ที่ใช้ในการปรับแต่งจะถูกเก็บไว้ใน /etc/httpd/conf การปรับแต่งส่วนมากจะทำในไฟล์ httpd.conf และ srm.conf ซึ่งมีรายละเอียด ดังนี้

httpd.conf

ใช้โปรแกรม vi เปิดดูข้อมูลภายในไฟล์ ถ้าเริ่มต้นบรรทัดมีเครื่องหมาย # นั่นคือ คำอธิบาย (comment) ในไฟล์นี้มีการประกาศตัวแปรระบบหลายตัว เช่น Server Root, Server Admin และ Server Name ที่เป็นตัวแปรที่ใช้ในการปรับแต่งระบบที่น่าสนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Server Root กำหนดไคลเรกทอรีที่ใช้ในการปรับแต่งเว็บเซิร์ฟเวอร์ซึ่ง default จะอยู่ที่ `/etc/httpd`
- Server Admin เป็นการกำหนด email ของ root ที่จะใช้ในการติดต่อเมื่อ Server มีปัญหา เช่น `webmaster@aquakmitl.ac.th`
- Server Name กำหนดชื่อของเซิร์ฟเวอร์ที่ใช้ในการอ้างอิง เช่น `aquakmitl.ac.th` หรือเป็น IP ของเครื่องก็ได้ เช่น `161.246.6.156`

srm.conf

ไฟล์นี้จะเก็บ configuration เกี่ยวกับ resource และ path directory ต่างๆ ของเว็บเซิร์ฟเวอร์มีตัวแปรต่างที่น่าสนใจดังนี้

- Document Root กำหนดไคลเรกทอรีที่เป็น HTML ของระบบ default อยู่ที่ `/home/httpd/html` ไฟล์ที่เป็นโฮมเพจของระบบจะเก็บไว้ที่นี่
- UserDir กำหนดชื่อของไคลเรกทอรีที่จะให้ User แต่ละคนเก็บโฮมเพจ
- DirectoryIndex กำหนดชื่อไฟล์ที่เป็นไฟล์แรกของโฮมเพจ ปกติจะเป็น `index.html`
- ScriptAlias ใช้ในการกำหนดชื่อเรียกของไคลเรกทอรี เพื่อให้สามารถเข้าถึงได้โดยเร็ว เราสามารถกำหนด Alias นี้ได้หลายชื่อ ซึ่งเขียนเรียงกับเป็นบรรทัดต่อกันไป เช่น ถ้าเราชี้ไปที่ `cgi-bin` ก็จะเป็นการอ้างอิงถึง `/home/httpd/cgi-bin` นั่นที่

การปรับแต่งเว็บเซิร์ฟเวอร์ให้รองรับซิจิไอได้นั้น ต้องแก้ไขที่ `srm.conf` โดยใช้ `vi` และแก้ไขบรรทัดต่อไปนี้

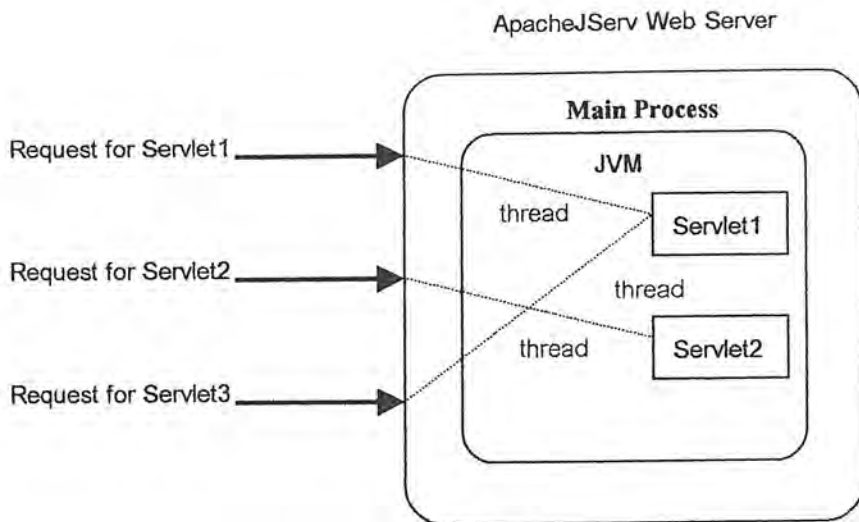
```
#AddHandler cgi-script.cgi
```

ให้นำเอา comment ในต้นบรรทัดออกไป และบันทึกไฟล์ จากนั้นก็ reboot เครื่องใหม่ ซึ่งเป็นการขอให้อ่าปาเซ่ ทราบดีว่า เราอนุญาตให้โปรแกรมที่มีนามสกุลเป็น `.cgi` นั้นเป็นซิจิไอสคริปต์

4.4 อ่าปาเซ่-เจเซิร์ฟ (ApacheJServ)

อ่าปาเซ่-เจเซิร์ฟเป็นเว็บเซิร์ฟเวอร์ที่เพิ่มความสามารถโดยใช้เซิร์ฟเล็ต (servlet) ซึ่งออกแบบมาเพื่อสนับสนุนการร้องขอและตอบสนองของเว็บเซิร์ฟเวอร์ โดยอยู่ในสถานะแวดล้อมเสมือนจาวา หรือ JVM เช่น การประมวลผลข้อมูลที่รับมาจากฟอร์มที่เขียนด้วย HTML หรือเป็นคัตกลาง (Middle-tier) ในการติดต่อกับฐานข้อมูล นอกจากนี้ยังทำให้อ่าปาเซ่-เจเซิร์ฟเป็นเว็บเซิร์ฟเวอร์ที่มีความสามารถในการให้บริการดีกว่าการใช้เทคโนโลยีซิจิไออีกด้วยในเรื่องของความเร็วและความง่ายในการพัฒนาโปรแกรมที่รองรับการร้องขอ และจัดหาไดนามิกเว็บเพจส่งกลับไป

อ่าปาเซ่-เจเซิร์ฟสามารถรองรับการร้องขอที่มีเข้ามาพร้อมกันได้ เนื่องจากมีการทำงานแบบมัลติเธรด (Multithread) แสดงดังรูปที่ 4.5



รูปที่ 4.5 การทำงานของเซิร์ฟเวอร์ที่อยู่ในอาปาเช่-เจเซิร์ฟ

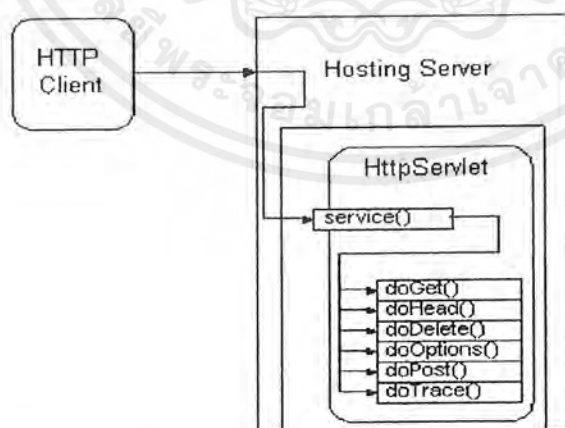
ผู้ใช้สามารถเรียกเซิร์ฟเวอร์ได้จากบราวเซอร์ โดยมีรูปแบบดังนี้

`http://localhost/servlet/servlet-name`

ตัวอย่างเช่น ถ้าต้องการเรียกเซิร์ฟเวอร์ที่ชื่อ `DateTimeServlet` ทำได้โดยเรียก URL ดังต่อไปนี้

`http://localhost/servlet/DateTimeServlet`

อาปาเช่-เจเซิร์ฟสามารถรองรับการร้องขอที่เข้ามาได้หลายรูปแบบ เนื่องจากโปรแกรมที่เขียนขึ้นมา นั้น เป็นคลาสที่มีการ extend มาจากคลาส `HttpServlet` ซึ่งคลาสนี้มีเมธอดที่ชื่อว่า `service()` คอยรองรับการร้องขอที่เข้ามาในรูปแบบต่างๆ ดังรูปที่ 4.6 เมธอด `doGet()` รองรับการร้องขอที่มีเข้ามาแบบ GET , เมธอด `doPost()` รองรับการร้องขอที่มีเข้ามาแบบ POST และเมธอดอื่นๆ อีก เช่น `doHead()`, `doDelete`, `doOptions()` และ `doTrace()` เป็นต้น



รูปที่ 4.6 เว็บเซิร์ฟเวอร์ที่มีเซิร์ฟเวอร์เจ็นจินซึ่งจัดการกับการร้องขอของไคลเอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.1 ลักษณะของอาปาเช่-เจเซิร์ฟ

- เป็น JavaSoft Servlet API 2.0 ทำให้สามารถนำโค้ดที่เขียนบนเซิร์ฟเล็ตหนึ่ง ไปรันบนเซิร์ฟเล็ตอื่นได้ (portable)
- สามารถนำโค้ดที่เขียนด้วยจาวาแอปพลิเคชัน 100% มาใช้กับเซิร์ฟเล็ตเอ็นจิน (servlet engine) ได้อย่างสมบูรณ์
- มีการทำงานแบบ Multi-threaded Servlets และแยกเซิร์ฟเล็ตเซิร์ฟเวอร์ออกจากเว็บเซิร์ฟเวอร์ เพื่อให้เว็บเซิร์ฟเวอร์มีความคงที่ในการทำงาน (stability) และใช้ทรัพยากรของระบบไม่มาก
- เซิร์ฟเล็ตสามารถให้บริการเป็นจำนวนมากได้ เพราะมีการทำโหลดบาลานซิง (Load Balancing)
- มีการจัดการส่งต่อรีเควสที่เข้ามา ได้อย่างฉลาด
- ใช้โปรโตคอลเฉพาะ AJP ทำให้สามารถสร้างเครือข่ายที่ซับซ้อนได้ (เชื่อมต่อเว็บเซิร์ฟเวอร์หลายตัวเข้ากับเซิร์ฟเล็ตเอ็นจินหลายตัวที่อยู่คนละเครื่องกัน)
- เซิร์ฟเล็ตสามารถแบ่งย่อยได้เป็นเขต เรียกว่า เซิร์ฟเล็ตโซน (servlet zones) โดยแต่ละเขตมีลักษณะเฉพาะของตัวเอง ทำให้ง่ายในการดูแล
- เซิร์ฟเล็ตสามารถรันจากไคลเรกทอรี หรือ ไฟล์ที่มีนามสกุลเป็น jar หรือ zip ได้
- คลาสเซิร์ฟเล็ตสามารถรีโหลดได้ โดยอัตโนมัติเมื่อมีการเปลี่ยนแปลง (autoreloading)
- สามารถใช้กับ Apache Web Server 1.2.x และ 1.3.x ได้อย่างสมบูรณ์ โดยใช้โมดูล mod_jserv
- มีเพจแสดงสถานะและไฟล์คอนฟิกูเรชันของระบบ สำหรับผู้ดูแลระบบ
- มีความสามารถในการทำดีบัก (debugging)
- เป็นระบบที่เปิดให้ผู้สนใจเข้ามาลองใช้ และพัฒนาต่อไปได้

4.4.2 คำศัพท์ที่เกี่ยวข้องกับอาปาเช่-เจเซิร์ฟ

Apache JServ

เป็นเซิร์ฟเวอร์ที่ใช้แอปพลิเคชันภาษาจาวา 100 %

mod_jserv

เป็น Apache module ที่เปลี่ยน HTTP requests ไปเป็น servlet requests ซึ่งเชื่อมต่อกับเซิร์ฟเล็ตเอ็นจินและส่ง HTTP requests กลับไปยังไคลเอ็นท์

Apache JServ Protocol

เป็นโปรโตคอลที่ใช้เชื่อมต่อระหว่างเว็บเซิร์ฟเวอร์และเซิร์ฟเล็ตเอ็นจินบนโปรโตคอล TCP/IP

Local Mode

เว็บเซิร์ฟเวอร์และเซิร์ฟเล็ตเอ็นจินอยู่บนเครื่องเดียวกันและใช้ทรัพยากรระบบร่วมกัน จะมีโหมดการทำงานอยู่ 2 แบบ คือ

- *integrated/automatic* เว็บเซิร์ฟเวอร์ควบคุมการทำงานของเซิร์ฟเล็ตเอ็นจิน, สร้างสภาวะแวดล้อมเสมือนของจาวา, เริ่มการทำงานใหม่ถ้ามีปัญหาเกิดขึ้นกับระบบ(crash) และหยุดการทำงานถ้าเว็บเซิร์ฟเวอร์ปิดทำงาน
- *standalone/manual* เว็บเซิร์ฟเวอร์และเซิร์ฟเล็ตเอ็นจินมีการทำงานที่ไม่ขึ้นต่อกัน ไม่มีการควบคุมซึ่งกันและกัน

Remote Mode

เว็บเซิร์ฟเวอร์และเซิร์ฟเล็ตเอ็นจินอยู่คนละเครื่องกัน เชื่อมต่อกันผ่านโปรโตคอล TCP/IP จะช่วยลดโอเวอร์เฮดที่เกิดจากการทำงานของเซิร์ฟเล็ตให้กับเว็บเซิร์ฟเวอร์

Servlet

เป็นคลาสอินเทอร์เฟซ `javax.servlet.Servlet` และคลาสที่มีการเรียกใช้อินเทอร์เฟซ `javax.servlet.Servlet` ดังนั้นจึงสามารถทำงานที่เซิร์ฟเล็ตเอ็นจินได้

Servlet Repository

เป็นกลุ่มของเซิร์ฟเล็ตที่ผ่านการคอมไพล์ เซิร์ฟเล็ต อาจเป็น ไคเรกทอรีเดี่ยว หรือ ไฟล์นามสกุล jar หรือ zip

Servlet Zone

เป็นเซิร์ฟเล็ตเอ็นจินที่เปรียบเสมือนเวอร์ชวลโฮสต์ของเว็บเซิร์ฟเวอร์

4.4.3 หลักการทำงานของเซิร์ฟเล็ตเอ็นจิน (Servlet Engine)

เมื่ออาปาเช่-เจเซิร์ฟเริ่มการทำงาน คลาส `org.apache.jserv.JServ` จะอ่านตัวแปรต่างๆ จากไฟล์พรอพเพอร์ตี้ที่ชื่อ `jserv.properties` เพื่อเริ่มการทำงาน ตัวแปรเหล่านี้ ได้แก่ พอร์ต, เซิร์ฟเล็ตโซน, ไฟล์คอนฟิกูเรชันของเซิร์ฟเล็ตโซน และตัวแปรต่างๆ ที่เกี่ยวข้องกับความปลอดภัย รวมถึง ความสามารถในการทำเทรซซิง(tracing capability) เพื่อทำการดีบัก (debugging) และ ล็อกไฟล์ (logging)

จากนั้น เซิร์ฟเล็ตโซนแต่ละโซนจะสร้างคลาส `org.apache.jserv.JServServletManager` และเซิร์ฟเวอร์จะรอรับการเชื่อมต่อจากพอร์ตที่กำหนด เมื่อมีการขอเชื่อมต่อขึ้นที่พอร์ต ถ้าการเชื่อมต่อมาจาก IP address ที่ไม่ได้รับอนุญาต มันจะปิดการเชื่อมต่อทันที เพื่อป้องกันการโจมตี แต่ถ้ามาจาก IP address ที่ได้รับอนุญาต มันจะทำการเชื่อมต่อโดยสร้างคลาส `org.apache.jserv.JServConnection` เพื่อจัดการกับรีเควสที่เข้ามา เนื่องจากคลาสนี้มีการเอ็กซ์เทนด (extend) มาจาก `Runnable` ทำให้มีความสามารถในการทำมัลติเธรด (Multithread) ซึ่งสามารถจัดการกับรีเควสที่มีเข้ามาพร้อมกันได้

หลังจากที่คลาส `JServConnection` รับตัวแปรต่างๆของรีเควสแล้ว มันจะเรียกคลาส `JServServletManager` (เฉพาะของแต่ละโซน) เมื่อได้รับเซิร์ฟเล็ตที่ถูกขอรีเควส มันจะสร้างสภาวะแวดล้อมในการประมวลผล และเรียกเมธอด `service()` ของเซิร์ฟเล็ต คลาส `JservConnection`, `JServContext` และ `JServSession` จะจัดหาเมธอดที่จำเป็นในการประมวลผล คลาส `JServConnection` จะจัดการสตรีมข้อมูลด้วย เมื่อเอาท์พุทของข้อมูลถูกส่งผ่านไปยังไคลเอ็นท์เรียบร้อยแล้ว คลาส `JServConnection` จะปิดซ็อกเก็ตที่เชื่อมต่อ แสดงว่าสิ้นสุดการรีเควส

4.4.4 ระบบรักษาความปลอดภัยของอาปาเช่-เจเซิร์ฟ

- **เว็บเซิร์ฟเวอร์ (Web server)**

โดยทั่วไป อาปาเช่เว็บเซิร์ฟเวอร์ และเครื่องที่มันวิ่งจะอยู่ในสภาพแวดล้อมที่มีความปลอดภัยในระดับหนึ่ง ในกรณีที่ต้องการความปลอดภัยเพิ่ม ระบบสามารถหา SSL มาใช้กับอาปาเช่ได้

- **โปรโตคอลอาปาเช่-เจเซิร์ฟ (Apache JServ Protocol)**

ข้อกีดกันที่เชื่อมต่อบetween อาปาเช่และอาปาเช่-เจเซิร์ฟมีการป้องกันการติดต่อที่มาจากเครื่องภายนอก (firewall) ปัจจุบันอาปาเช่-เจเซิร์ฟมีการกรอง IP และการให้อนุญาตการเชื่อมต่อ ซึ่งป้องกันการ

- **Hostile Servlet**

ป้องกันการ โหลด โค้ดเซิร์ฟเล็ตที่ไม่ได้รับอนุญาต

- **Network-load code**

อาปาเช่-เจเซิร์ฟไม่สนับสนุนการ โหลด โค้ดผ่านเครือข่าย

- **Cookies and sessions**

คุกกี้ถูกใช้ในการกำหนดเซสชัน (session) อีอบเจ็คจะถูกส่งผ่านเครือข่าย การโจมตีระยะไกลสามารถเดาได้ว่าเป็นการปลอมตัวมาจากผู้อื่นๆ โดยปกติคุกกี้ไม่ถูกเก็บอยู่ในดิสก์เมื่อผู้ใช้เลิกใช้เบราว์เซอร์ เจเซิร์ฟสามารถเชื่อมต่อไปยังข้อมูลบนฝั่งเซิร์ฟเวอร์ โดยไม่เกิดความเสียหายต่อเบราว์เซอร์

- **System classes**

อาจเกิดปัญหาขึ้นได้ ถ้าผู้ใช้ติดตั้งคลาสเซิร์ฟเล็ตทับกับคลาสของจาวารัน-ไทม์ (Java Runtime)

- **Native code**

อาปาเช่-เจเซิร์ฟในปัจจุบันอนุญาตให้มีการ โหลดเนทีฟโค้ดจากภายในเซิร์ฟเล็ตเองและตัวจาวา โดยไม่มีข้อบังคับใดๆ

บทที่ 5

จาวา (JAVA)

จาวาเป็นโปรแกรมขั้นสูง ถูกพัฒนาขึ้นโดยบริษัท ซัน ไมโครซิสเต็มส์ เริ่มแรกขึ้นได้คิดค้นภาษาจาวาขึ้นมาเพื่อใช้กับระบบปฏิบัติการ Set Top Box คือ โครงการของระบบคลาสิคอิเล็กทรอนิกส์ ที่ต้องการให้ระบบสามารถทำการควบคุมอุปกรณ์อิเล็กทรอนิกส์ได้โดยไม่ขึ้นกับแพลตฟอร์ม (Platform Independent) คือการพัฒนากระบวนการหรือซอฟต์แวร์ให้สามารถทำงานได้ โดยไม่สนใจว่าไมโครโพรเซสเซอร์ที่ใช้อยู่จะเป็นรุ่นใด ยี่ห้อไหน ฮาร์ดแวร์เป็นของอะไร ทั้งยังได้รวมความสามารถไปถึงการเชื่อมต่อ การแลกเปลี่ยนข้อมูลข่าวสารทั่วไปและทางด้านธุรกิจผ่านทางระบบเครือข่ายคอมพิวเตอร์อีกด้วย

ทางทีมงานซึ่งพัฒนาโปรแกรมนี้ได้สังเกตเห็นถึงการเจริญเติบโตและความสามารถในการแลกเปลี่ยนข่าวสารบนอินเทอร์เน็ต จึงได้วางแผนพัฒนาระบบอินเทอร์เฟซ (Interface) แบบใหม่เพื่อใช้กับอินเทอร์เน็ตเช่น ด้วยสาเหตุนี้เองภาษาจาวาจึงได้บังเกิดขึ้น รวมทั้งยังได้มีการเปิดตัวเว็บเบราว์เซอร์ที่ชื่อว่า Hotjava เพื่อใช้สำหรับแสดงความสามารถของภาษาจาวา

5.1 ความสามารถของภาษาจาวา

ความสามารถของภาษาจาวาสรุปได้ดังต่อไปนี้

- การใช้งานที่ไม่ขึ้นกับแพลตฟอร์ม โดยใช้วิธีการแปลงข้อมูลแบบไบนารีโค้ด (Byte code)
- การใช้เมธอดจัดการกับข้อมูลแทนการใช้ฟังก์ชัน
- ความสามารถในการโต้ตอบแบบ Interactive กับงานด้านมัลติมีเดีย
- ความสามารถในการดึงเอาคลาสไลบรารีจากที่ต่างๆ ผ่านทาง HTTP และ FTP
- ความสามารถในการเล่นไฟล์เสียง ไฟล์วิดีโอและไฟล์ภาพกราฟฟิกเคลื่อนไหว
- การโต้ตอบแบบเรียลไทม์
- เป็นระบบมัลติเทรด (Multithread) คือสามารถทำงานได้หลายงานพร้อมกันในเวลาเดียว
- ไม่มีตัวแปรแบบพอยน์เตอร์ จึงหมดปัญหาเกี่ยวกับการขอใช้หน่วยความจำโดยตรง
- มีความปลอดภัยสูง ด้วยความสามารถที่ไม่ให้ผู้ใช้เข้าไปยุ่งเกี่ยวกับหน่วยความจำของระบบจึงปลอดภัยจากไวรัส

5.2 การนำจาวาไปประยุกต์สร้างและพัฒนาในงานด้านต่างๆ

เราสามารถนำจาวาสร้างและพัฒนาแอปพลิเคชันต่างๆ โดยยกตัวอย่างพอสังเขปดังนี้

ไคลเอนท์ (Client)

จาวาสถาปัตยกรรมเป็นเครื่องมือในการพัฒนาซอฟต์แวร์ด้านไคลเอนต์ โดยเฉพาะด้านกราฟิก และระบบคิดคําคับผู้ใช้ เพราะปัจจุบัน โปรแกรมที่เขียนขึ้นจากภาษาจาวาจะทำงานที่ฝั่งไคลเอนต์เป็นส่วนมาก เช่น จาวาแอปเพล็ต (Java Applet) ต่าง ๆ บนเว็บเบราว์เซอร์

เมื่อสร้างโปรแกรมประยุกต์บนเซิร์ฟเวอร์นั้น จำเป็นต้องได้รับการปรับปรุงการทำงานอยู่สามอย่าง คือ การเชื่อมต่อฐานข้อมูลโดยตรง การทำงานให้สอดคล้องกับอินพุท เอาท์พุท และความเร็วของการรันต้องเพิ่มขึ้นเท่าๆ กับโปรแกรมประยุกต์ทั่วไป

JDBC (Java Database Connectivity) ได้เข้ามาแก้ปัญหาในส่วนของการติดต่อกับฐานข้อมูลโดยตรง โดยการร่วมมือกันของบริษัทซอฟต์แวร์ที่มีชื่อเสียงหลายบริษัทที่สร้างมาตรฐานการเชื่อมต่อกับฐานข้อมูลของจาวาขึ้น

จาวาได้สร้างสภาพแวดล้อมเสมือน Java Virtual Machine (JVM) ซึ่งอนุญาตเฉพาะ จาวาแอปเพล็ต (Java Applet) และจาวาแอปพลิเคชัน (Java Application) ซึ่งเป็น ไบท์โค้ดเท่านั้นที่สามารถรันได้ โดยที่ไบท์โค้ดจะปราศจากไวรัสหรือส่วนที่จะทำอันตรายต่อระบบ อันเนื่องจาก JVM จะจำกัดสิทธิ์การเข้าใช้ทรัพยากรของไบท์โค้ด ในอนาคตการรักษาความปลอดภัยอีกระบบหนึ่งก็กำลังถูกนำมาปรับใช้ คือ “การเข้ารหัสข้อมูล” เพิ่มความปลอดภัยของข้อมูล

5.3 หลักการทำงานของภาษาจาวา

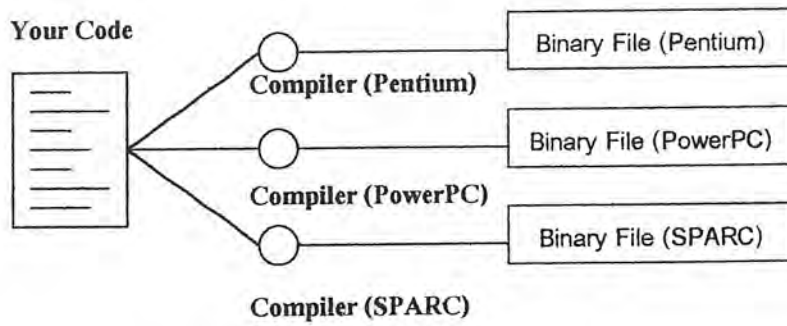
ในที่นี้ขออธิบายการทำงานของภาษาจาวาเริ่มตั้งแต่การคอมไพล์ตั้งโปรแกรมจนกระทั่งเราเรียกใช้งานในเว็บเพจ เป็นดังนี้

5.3.1 การคอมไพล์

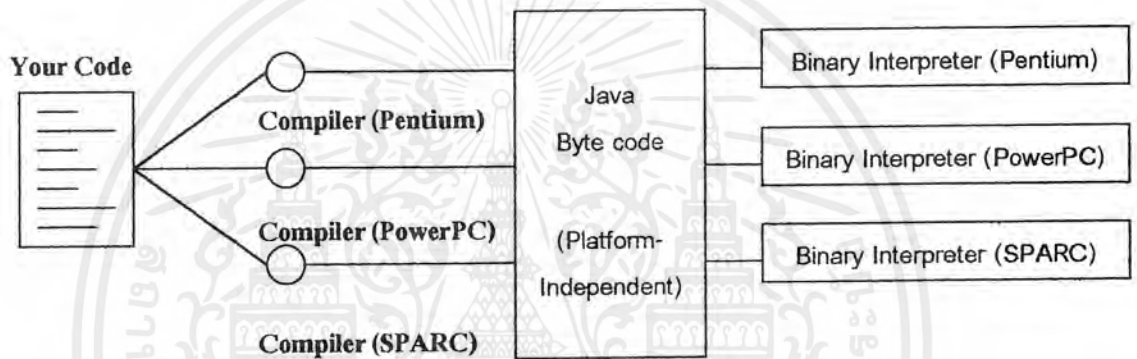
การคอมไพล์ของภาษาจาวาก็เช่นเดียวกับคอมไพเลอร์ในภาษาอื่นๆ นั่นคือ มันจะสร้างรหัสภาษาเครื่อง (Machine Code หรือ Assemble Code) จากภาษาในระดับที่สูงกว่าเพื่อให้อุปกรณ์ (CPU: Central Processing Unit) สามารถนำไปใช้งานได้

แต่ข้อแตกต่างที่สำคัญระหว่างคอมไพเลอร์ของภาษาจาวากับภาษาอื่นๆ คือ ซีพียูหรือโปรเซสเซอร์ที่จะทำหน้าที่ในการปฏิบัติตามคำสั่งที่ได้จากการคอมไพล์ภาษาจาวานั้น ไม่มีอยู่จริง เป็นเพียงสิ่งที่สมมติขึ้นมาที่เรียกว่า Java Virtual Machine นอกจากนี้การอ้างถึงส่วนต่างๆ ของโปรแกรมที่คอมไพล์ด้วยคอมไพเลอร์ของภาษาจาวาก็จะมีวิธีการที่แตกต่างออกไป

คอมไพเลอร์ของภาษาจาวาจะไม่เปลี่ยนการอ้างถึงส่วนของโปรแกรมจากการใช้ชื่อแบบในภาษาสูง ไปเป็นตัวเลขเหมือนคอมไพเลอร์ภาษาอื่นๆ ทำกัน และคอมไพเลอร์ของภาษาจาวาก็จะไม่มีการสร้างแผนที่ของการจัดวางโปรแกรมบนหน่วยความจำขึ้นมาในระหว่างการคอมไพล์ ด้วยเหตุผลที่สำคัญคือเพื่อเป็นการสร้างความพอร์ดตารางให้กับตัวโปรแกรม เพราะการจัดวางตำแหน่งของโปรแกรมจะต้องขึ้นอยู่กับลักษณะการทำงานของโปรเซสเซอร์ตัวใดตัวหนึ่ง การยังไม่จัดวางตำแหน่งช่วยทำให้โปรแกรมที่ได้จากการคอมไพล์มีความเป็นกลาง สามารถนำไปใช้บนคอมพิวเตอร์แพลตฟอร์มอื่นๆ ได้ นอกจากนี้ยังทำให้เกิดความปลอดภัยอีกด้วย สิ่งที่ได้จากการคอมไพล์ในภาษาจาวาเรียกว่า ไบท์โค้ด (ByteCode)



รูปที่ 5.1 การประมวลผลของโปรแกรมทั่วไปแต่ละระบบปฏิบัติการที่ต่างกัน



รูปที่ 5.2 การประมวลผลของโปรแกรมภาษาจาวา

5.3.2 การวางตำแหน่งในหน่วยความจำ

ในภาษาจาวาจะไม่มีกรลสรุปแบบการอ้างถึงส่วนต่างๆ ของโปรแกรมจากการเรียกเป็นชื่อให้เหลือเพียงตัวเลขหรือแอดเดรสที่กำหนดขึ้นจากการจัดวางตำแหน่งของโปรแกรมลงในหน่วยความจำ คอมไพเลอร์ภาษาจาวาจะทิ้งชื่อของแต่ละส่วนของโปรแกรม (โดยเฉพาะเมธอด) เอาวางไว้ในตัวโปรแกรมที่สร้างขึ้น เมื่อโปรแกรมทำงาน จะเป็นหน้าที่ของตัวอินเตอร์พรีเตอร์ ที่จะคอยเปิดตารางค้นหาที่อยู่ของเมธอดที่ต้องการเรียกใช้งาน โดยก่อนที่จะเริ่มทำงานจริง อินเตอร์พรีเตอร์ จะต้องสร้างแผนที่ในการจัดวางสิ่งต่างๆ ลงในหน่วยความจำขึ้นมาก่อนแล้วจึงสร้างตารางขึ้นมา เพื่อช่วยหาด้านตำแหน่งของเมธอดเมื่อมีการเรียกใช้งาน โดยใช้ชื่อของเมธอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.3 การรันโปรแกรม

การรันโค้ดที่คอมไพล์เอาไว้สำหรับ Java Virtual Machine เป็นหน้าที่ของตัวอินเทอร์พรีเตอร์ การรันโปรแกรมจะแบ่งได้เป็น 3 ชั้นหลักๆ คือ การอ่าน การตรวจสอบความถูกต้อง และการทำตามโค้ด หน้าที่ในการอ่าน โค้ดเข้าสู่ระบบจะเป็นของ Class Loader หน้าที่การทำงานในส่วนนี้จะไม่ได้อ่านเข้ามา เฉพาะไฟล์จาวาที่กำลังจะเรียกใช้เท่านั้น แต่จะอ่านคลาสที่มีการอ้างอิง และคลาสที่มีการ inherited มาโดย คลาสที่อ้างอิง เมื่อผ่านชั้นตอนนี้แล้ว โค้ดทั้งหมดก็จะถูกส่งผ่านตัวตรวจสอบไบท์โค้ดเพื่อให้แน่ใจว่า โค้ดที่ส่งมามีความถูกต้องตามมาตรฐานของจาวา และจะไม่รบกวนเสถียรภาพของระบบ เมื่อผ่านการ ตรวจสอบแล้ว โค้ดก็จะถูกส่งต่อไปยังระบบรันใหม่ (Run-Time System) ซึ่งจะตั้งงานไปยังฮาร์ดแวร์อีก ต่อหนึ่ง ซึ่งหลักการทำงานในแต่ละชั้นตอนที่กล่าวมาขั้นต้นเป็นดังนี้

5.3.4 คลาสโหลดเดอร์ (Class Loader)

ตัวคลาสโหลดเดอร์ จะทำหน้าที่ดึงโค้ดทั้งหมดที่จำเป็นในการทำงานของแอปพลิเคชัน ไม่ว่าจะ เป็นคลาสที่ถูก Inherited มาหรือคลาสอื่นๆ ที่มีการเรียกใช้ เมื่อโหลดเดอร์ดึงคลาสใดเข้าแล้วก็จะจัดคลาสนั้นๆ ใส่เข้าไปใน namespace ของมันเอง โดยการเก็บ จะใช้ชื่อของคลาสเป็นสำคัญ ไม่ได้ใช้การอ้างอิง เป็นตัวเลข หลักการนี้ก็จะเหมือนกันกับการทำงานของ Virtual Machine ที่โอเอส (OS: Operating System) สร้างขึ้นให้แอปพลิเคชันแต่ละตัวทำงาน ถ้าไม่ได้มีการเจาะจงเรียกใช้คลาสมีอยู่นอก namespace นี้ การเรียกใช้ชื่อต่างๆ ในคลาส ก็จะไม่มีการกบเกี่ยวกันระหว่างคลาสเลย

คลาสทั้งหมดที่อยู่บนเครื่องโกลบอลเอง จะได้รับห้วงแอดเดรส (Address Space) เป็นของตัวเอง ส่วนคลาสต่างๆที่ดึงมาจากภายนอกจะได้รับ namespace เป็นของตัวเอง การทำงานลักษณะนี้จะช่วยให้ คลาสที่อยู่บนโกลบอลทำงานได้ประสิทธิภาพดีขึ้น เพราะใช้ namespace ร่วมกันได้ แต่ก็ยังมีทางป้องกัน ความผิดพลาดที่อาจเกิดจากคลาสที่ดึงเข้ามาจากภายนอก และในทางกลับกัน คลาสที่อิมพอร์ตเข้ามา ก็ปลอดภัยจากความผิดพลาดที่อาจจะเกิดขึ้นจากคลาสโกลบอลด้วย

เมื่อคลาสทั้งหมดที่เกี่ยวข้องกับการทำงานถูกอิมพอร์ตเข้ามาเรียบร้อยแล้ว การจัดวางหน่วยความ จำสำหรับเริ่มต้นการทำงานก็จะเกิดขึ้นได้ การเรียกชื่อต่างๆก็จะสามารถจับคู่กับแอดเดรสจริงๆ ของ หน่วยความจำได้ แล้วตัวโหลดเดอร์จะสร้างตารางสำหรับค้นหาที่อยู่ซึ่งจากการทำงานผิดพลาดของซูเปอร์คลาส (Super class) และการอ้างอิงแอดเดรสที่ไม่ถูกต้องได้

5.3.5 ตรวจสอบไบท์โค้ด

เมื่อโค้ดเดินทางมาถึงขั้นตอนการสร้างตารางจับคู่ชื่อกับแอดเดรสแล้ว ก็ยังไม่สามารถแน่ใจ ได้ว่าโค้ดที่อ่านเข้ามาจะมีความปลอดภัย ดังนั้นจึงต้องมีตัว verifier หรือตัวตรวจสอบไบท์โค้ด ทำหน้าที่ ตรวจสอบความถูกต้องที่ละบรรทัดว่าเป็นไปตามข้อกำหนดของจาวา และสอดคล้องกับการทำงานของ ตัวโปรแกรมหรือไม่ การตรวจสอบโค้ดในเชิงทฤษฎีจะสร้างค้นหาปัญหาต่างๆ ได้หลายอย่าง เช่น จะ ไม่มีการสร้างพอยน์เตอร์ที่เกินกว่าหน่วยความจำจริง ไม่มีคำสั่งใดสามารถละเมิดสิทธิ์การทำงานของตัว

โปรแกรมได้ ไม่มีการจับคู่ อีอบเจ็คตึค จะไม่มีการให้โอเปอเรชั่นรันค้มากหรือน้อยเกินไป การกำหนดค่าต่างๆ สำหรับไบท์โค้ดจะต้องถูกค้องครบถ้วน และจะไม่มีการแปลงข้อมูลค้ีกรูบแบบ

การใช้ค้ีตรวจสอบค้องค้องจุดประสงค้ 2 ประการสำคัญ ค้ี ค้ีงค้่างๆ ค้ีงที่กล่าวมาแล้วจะถูกตรวจสอบก่อนทำให้ค้ีวอินเตอร้พรีเตอร้มันใจได้ว้า ไบท์โค้ค้ที่ส่งเข้าไปท้างานจะไม่ม้ีข้ันค้องการท้างานที่สร้างปัญหาให้กับค้ีระบบ และจุดประสงค้ที่สองก็ค้ี ค้ีวอินเตอร้พรีเตอร้จะท้างานตามไบท์โค้ค้ได้เร็วกว้า เพราะไม่ค้องค้องระวังว้าจะมีปัญหาเกิดข้ัน และไม่ค้องหยุดเป็นข้่วงๆ เมื่อพบปัญหาและค้องแก้ไข

ในการท้างาน ไบท์โค้ค้จะถูกตรวจสอบเพียงครั้งเดียวเท่านั้น และจะท้างานไปได้ค้ลค้ไม่ค้องม้ีการตรวจสอบข้้าอีกเมื่อมีการเรียกกลับมาท้างานที่ส่วนเดิมของโปรแกรม

5.3.6 การท้างานตามโค้ค้

เมื่อค้ีวโหลค้เลอร้ได้รวบรวมโค้ค้เข้ามาสู่ระบบท้างานจัดวางในหน่วยความจำ และค้ีวตรวจสอบได้ท้่าการตรวจสอบความถูกค้องแล้ว โค้ค้ก็จะถูกส่งค้่อไปยังค้ีวอินเตอร้พรีเตอร้เพื่อท้างานตามค้่าสั่ง การท้างานตามค้่าสั่งของโค้ค้ก็ค้ี การเปล้ียนโค้ค้ให้กลายเป็นค้่าสั่ง การท้างานจริงที่ค้ีระบบค้ลเอน้ที่ท้รันโค้ค้นี้สามารถท้างานได้ ซึ่งวิธีการที่ท้่าได้ก็มีอยู่ 2 วิธีค้้วยกันค้ี ค้ีวอินเตอร้พรีเตอร้ท้่าการคอมไพล์โค้ค้เหล่านี้ให้กลายเป็นเนทิวโค้ค้ที่ค้ีวเครื่องค้ลเอน้ที่เข้าใจ แล้วค้่อยท้างาน เพื่อให้ได้ความเร็วสูงค้สุดในการท้างาน กับอีกวิธีหนึ่งก็ค้ี อินเตอร้พรีเตอร้อ่านโค้ค้เข้ามาแล้วค้ีความท้างานไปที่ละค้่าสั่ง และท้่าการค้ีความไปเรื่อยๆ ค้ลค้เวลาที่มีการท้างาน

โดยปกติแล้ว ผู้สร้างอินเตอร้พรีเตอร้มักจะเลือกใช้วิธีการหลัง รูบแบบของไบท์โค้ค้ในภาษาจาวามีความยืดหยุ่นเพียงพอที่จะสามารถเปล้ียนไปท้างานบนเครื่องค้ลเอน้ที่แบบค้่างๆ ได้โดยไม่มีการก้่อให้เกิด โอเวอร์เซคมากมายนเกินความจำเป็น อย่างไรก็ตามค้ลเอน้ที่ของจาวาบางระบบจะมีความสามารถในการท้างานได้ทั้งสองวิธี ค้ี โปรแกรมเมอร์สามารถจะเลือกใช้วิธีการคอมไพล์กับงานที่เน้นการค้่านวมมากๆ เพื่อเป็นการเพื่อสมรรถนะในการท้างานให้ได้เต็มที่ ซึ่งค้ลเอน้ที่แบบนี้จะให้ค้ีทั้งความพอร์ค้ดราจ และสมรรถนะที่ค้ี

การสร้างระบบรันไทม์ที่ค้ีจะค้องถ้่วงค้ลค้ความสำคัญ 3 ประการให้ค้ีพอเหมาะะ นั่นค้ี ความพอร์ค้ดราจ ความปลอดภัย และสมรรถนะ เรื่องของความ พอร์ค้ดราจท้่าได้โดยการ ใช้รูบแบบของไบท์โค้ค้ที่มีความเป็นกลางเพียงพอ สามารถนำไปใช้รันบนเครื่องคอมพิวเตอร้แบบค้่างๆ ได้โดยง่ายนอกจากนั้น การที่ค้ีวอินเตอร้พรีเตอร้ท้่าการกำหนดการจ้ดวางค้่าแห่งในหน่วยความจำในช่วงรันไทม์ (แทนที่จะเป็นระหว้่างการคอมไพล์เหมือนภาษาอื่น) ก็เป็นการเพิ่มความแน่นอนว้า คลาสค้่างๆ ที่อินพอร์ค้เข้ามาจะยังคงใช้ค้ีอยู่ค้ลค้เวลา เรื่องของความปลอดภัยนั้น เป็นค้ีงที่มีการค้่านึงถ้ิงอยู่ค้ลค้ ภาระบวการท้างานของระบบรันไทม์ โดยเฉพาะอย่างยิ่งในส่วนของค้ีวตรวจสอบไบท์โค้ค้ ที่ท้่าให้แน่ใจได้ว้าโปรแกรมจะท้างานได้ถูกค้องค้่ากำหนดค้องจาวา ส่วนเรื่องของสมรรถนะนั้นก็สามารถจ้ดการได้ไ้สองระยะ ค้ี พยายามเอาโอเวอร์เซคทั้งหลายไปไว้ที่ค้องเริ่มค้ัน โหลค้โปรแกรมเข้ามาสู่ระบบ หรือไม่ก้่าหนดให้ท้างานเป็นแบบแบ้คกราวนค้ (Back-Ground Thread)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว้ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คล้ายดังต่างๆ เหล่านี้ ทำให้จาวาสามารถปล่อยสมรรถนะระดับที่น่าพอใจออกมาได้ โดยที่ยังคงไว้ซึ่งความปลอดภัย และสภาพแวดล้อมที่ปลอดภัย นอกจากนี้ ยังสามารถจะดึงสมรรถนะระดับสูงสุดออกมาใช้ได้ทันทีเมื่อต้องการ

5.4 การสร้างและรันภาษาจาวา

การสร้าง โปรแกรมจาก Java ก็เหมือนกับภาษา Programming อื่นๆ คือเขียน Code โปรแกรมจาก Editor ใดๆ ก็ได้ บันทึกที่อยู่ในนามสกุล .java จากนั้นจึงนำไปคอมไพล์ด้วยคอมไพเลอร์ของ Java จะได้ไฟล์ใหม่ในนามสกุล .class ที่เก็บรหัสของการคอมไพล์ไว้จาวาเรียกรูปแบบข้อมูลที่อยู่ในไฟล์ใหม่นี้ว่า Byte Code ไฟล์ที่ได้คือแอปเพล็ตนั่นเอง แอปเพล็ตยังไม่สามารถรันได้ทันทีเหมือนไฟล์นามสกุล .exe หรือ .com ที่เรารู้คุ้นเคยกัน เพราะข้อมูลแบบ Byte Code จะมีรูปแบบข้อมูลที่อยู่ที่กึ่งกลางระหว่างโค้ดโปรแกรม (Source Code) กับโค้ดที่คอมพิวเตอร์อ่านแล้วนำไปปฏิบัติได้ทันที (Machine Code) หากจะรันแอปเพล็ตบนระบบใดๆ ต้องใช้อินเตอร์พรีเตอร์จาวาของระบบนั้นๆ เพื่อตรวจสอบความถูกต้องของ Byte Code แล้วแปลให้เป็นรหัสภาษาเครื่องส่งให้ระบบปฏิบัติการนำไปรันต่อไปขั้นตอนการทำงาน

ลักษณะของการเขียนโปรแกรมภาษาจาวาเป็นการเขียนโปรแกรมเชิงวัตถุ OOP (Object Oriented Programming) ตัวโปรแกรมต้องถูกคอมไพล์เป็นโค้ด ซึ่งเรียกว่า Java byte code เครื่องมือที่ใช้พัฒนาภาษาโปรแกรมจาวามีการจำลองตัวมันเองเป็น JVM (Java virtual machine) ซึ่งเป็นการสมมติให้คอมพิวเตอร์อีกเครื่องหนึ่งขึ้นมา โดยเครื่องนี้จะใช้คอมไพล์ภาษาจาวาทุกโปรแกรม และเอ็กซีคิวต์คำสั่งแต่ละคำสั่งในจาวาโค้ดให้คอมพิวเตอร์ทำงานตามคำสั่งได้ ข้อดีของการเขียนภาษาจาวามีดังนี้

- เป็นภาษาที่สามารถเรียนรู้ได้อย่างรวดเร็ว
- ในการที่จะเขียน โปรแกรมหนึ่ง ๆ ขนาด โค้ดของจาวาจะเล็ก
- สามารถนำไปใช้งานกับเครื่องประเภทอื่นได้

การเขียนโปรแกรมภาษาจาวาแบ่งเป็น 2 ประเภท

- จาวาแอปพลิเคชัน (Java application) เป็นโปรแกรมที่มีลักษณะ standalone ไม่สามารถใช้งานบนเว็บเบราว์เซอร์ได้
- จาวาแอปเพล็ต (Java applet) สามารถใช้งานบนเว็บเบราว์เซอร์ได้

การเขียนโปรแกรมภาษาจาวานั้น เราต้องคิดตั้งตัวแปลภาษาจาวาก่อน ซึ่งในระบบนี้ เราได้เลือก JDK1.2.2 เป็นคอมไพเลอร์ เราสามารถใช้คำสั่ง javac เพื่อคอมไพล์โปรแกรม ผลที่ได้จากการคอมไพล์โปรแกรมจาวา คือ ไฟล์นามสกุล .class เป็นเหมือนกับไฟล์เอ็กซีคิวต์ของจาวา การเรียกให้ .class นี้ทำงานไม่ต้องใส่นามสกุล .class

การเขียนโปรแกรมภาษาจาวา จะต้องกำหนดให้ชื่อไฟล์กับชื่อคลาสเป็นชื่อเดียวกัน เพราะในการคอมไพล์ หรือรันโปรแกรม เราจะเรียกโดยใช้ชื่อคลาสนั้นเอง ซึ่งไฟล์ .class ของจาวาสามารถทำงานข้ามแพลตฟอร์มได้ และตรงนี้เองที่ทำให้ภาษาจาวาได้รับความนิยม และนำไปใช้มากขึ้น

ตัวอย่าง การใช้ภาษาจาวา แสดงข้อความ Easy Java ออกมาที่คอมมานไลน์

1. เขียนซอร์สโค้ดชื่อ Easy.java (เขียนบนอิดิตเตอร์ตัวใดก็ได้ แล้วเซฟไฟล์เป็นนามสกุล .java) ตัวอย่างโค้ดเป็นดังนี้

```
class Easy {
    public static void main(String arg[]) {
        System.out.println( "Easy Java\n" );
    }
}
```

2. คอมไพล์ซอร์สโค้ด โดยใช้คำสั่ง `#javac Easy.java`
(ถ้าคอมไพล์สำเร็จ คอมไพล์เลอร์จะสร้างไฟล์ Easy.class ให้)
3. รัน Easy.class ที่ได้ โดยใช้คำสั่ง `#java Easy`

ผลที่ได้จากการรัน Easy.class เป็นดังนี้

```
#java Easy
Easy Java
```

ตัวอย่าง การเขียนจาวาแอปเพล็ต แสดงข้อความ Hello World ออกมาที่บราวเซอร์

1. เขียนซอร์สโค้ดของจาวาแอปเพล็ต ชื่อ HelloWorld.java โดยมี import class Applet และ Graphics เข้ามา ตัวอย่างโค้ดเป็นดังนี้

```
import java.applet.Applet;
import java.awt.Graphics;
public class HelloWorld extends Applet
{
    public void paint(Graphics g) { g.drawString("Hello World! ", 50, 20); }
}
```

2. คอมไพล์ซอร์สโค้ด โดยใช้คำสั่ง `#javac HelloWorld.java`
ถ้าคอมไพล์สำเร็จ คอมไพล์เลอร์จะสร้างไฟล์ HelloWorld.class ให้ ซึ่งเป็นจาวาไบต์โค้ด และเป็นไฟล์ที่เป็นแอปเพล็ต
3. สร้างเอกสาร HTML ที่แทรกแท็ก `<APPLET>...</APPLET>` โดยใช้ชื่อไฟล์ "HelloWorld.class" แทรกลงไปในแท็ก `<APPLET>`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโค้ด "hello.html" แสดงดังต่อไปนี้

```
<HTML>
<HEAD><TITLE>A Sample Program</TITLE></HEAD>
<BODY>
<APPLET CODE="HelloWorld.class" WIDTH=400 HEIGHT=400></APPLET>
</BODY>
</HTML>
```

ทำการรันแอปพลิเคชันที่สร้างขึ้น เราสามารถเรียก "hello.html" จากบราวเซอร์ได้โดยตรง หรือใช้ `appletviewer` ที่มีมากับ JDK เพื่อผลลัพธ์เฉพาะส่วนที่เป็นแอปพลิเคชันที่สร้างขึ้นจากคำสั่ง โดยใช้คำสั่ง

```
#appletviewer hello.html
```



รูปที่ 5.3 เอกสาร HTML ที่มีการเรียกแอปพลิเคชันขึ้นมาจากเน็ตสเคป



รูปที่ 5.4 เอกสาร HTML ที่มีการเรียกแอปพลิเคชันขึ้นมาจาก Appletviewer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5 JDBC (Java Database Connectivity)

JDBC (Java Database Connectivity) ถูกพัฒนาโดย JavaSoft Department ของบริษัท Sun Microsystems ซึ่งก็คือฟังก์ชันมาตรฐาน หรือ Java Application Programming Interface (API) สำหรับการเชื่อมต่อกับระบบฐานข้อมูล นักพัฒนาสามารถใช้ JDBC API และยังสามารถใช้แพ็คเกจอื่นๆ ด้วย ซึ่งนำเสนอในรูปแบบฟังก์ชันพิเศษ โดยทั่วไปการใช้ SQL database ในการติดต่อกับเฟรมเวิร์ค (framework) เพื่อพัฒนามาตรฐานในการติดต่อกับส่วนบนสุดของชนิดต่างๆ ของ database connectivity modules ซึ่งก็คือมาตรฐานของ ANSI SQL-2 Entry level database เพราะว่า relational databases เกือบทั้งหมดในปัจจุบันใช้มาตรฐานของ SQL-2 Entry level

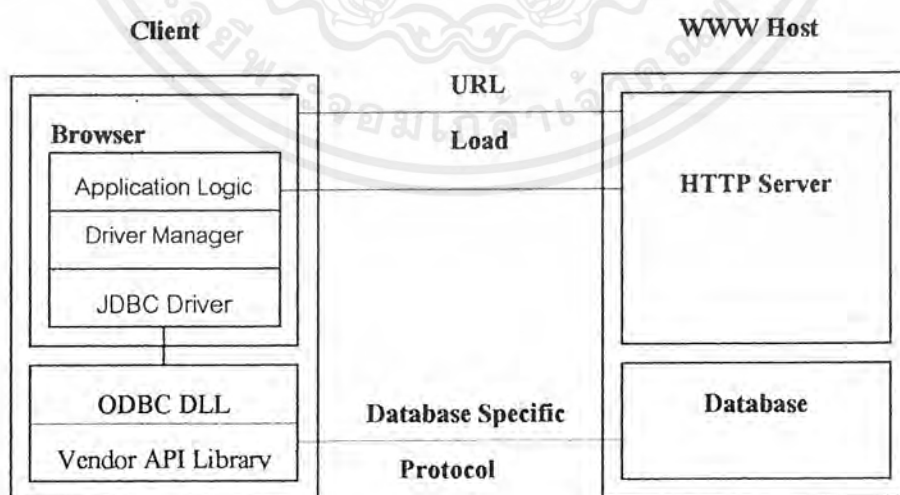
JDBC สร้างระดับการเชื่อมต่อเพื่อการสื่อสารกับฐานข้อมูลในรูปแบบที่คล้ายคลึงกับ ODBC (Open Database Connectivity ของบริษัท ไมโครซอฟท์) หลักการทำงานของทั้ง JDBC และ ODBC ตั้งอยู่บนมาตรฐานเดียวกัน คือ X/Open SQL Call-Level Interface ของระบบ X Window

โครงสร้างของ JDBC

โครงสร้างของการเชื่อมต่อภายใน JDBC ประกอบด้วย 3 ระดับหลัก คือ JDBC API, JDBC Driver API และ JDBC Driver ดังรูป ระดับบนสุด JDBC API เป็นระดับฟังก์ชัน API ที่อำนวยความสะดวกให้แก่โปรแกรมประยุกต์ ระดับล่าง JDBC Driver (มีไคร์ฟเวอร์ที่ต่างกันอยู่ 4 ชนิด) รายละเอียดการทำงานของไคร์ฟเวอร์แต่ละชนิดอธิบายได้ดังนี้

5.5.1 JDBC/ODBC bridge

JDBC/ODBC bridge ทำหน้าที่เป็นตัวกลางในการเข้าถึงฐานข้อมูลได้โดยผ่านการทำงานของ ODBC โดยนำข้อดีของ ODBC-enabled data sources ที่มีใช้อยู่โดยทั่วไปอย่างมากมา ผังไคลเอนท์ จาวา แอปพลิเคชันหรือจาวาแอปพลิเคชันจะถูกเขียนโดยใช้ JDBC API บริษัทจะทำการแปลงการเรียกใช้ JDBC ไปยัง ODBC และส่งค่า ODBC driver ที่เหมาะสมสำหรับ back-end database



รูปที่ 5.5 การติดต่ JDBC โดยรูปแบบที่ JDBC-ODBC Bridge

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของบริดจ์ ทำให้แอปพลิเคชันสามารถติดต่อกับฐานข้อมูลได้อย่างง่ายดาย โดยจากผู้ผลิตที่มีมากมาย โดยสามารถเลือก ODBC driver ที่เหมาะสม อย่างไรก็ตามการติดต่อกับฐานข้อมูลประเภทนี้ต้องพิจารณาค่าใช้จ่าย (overhead) และความซับซ้อน (complexity) เพราะว่าการเรียกใช้จะมีลำดับดังนี้คือ

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

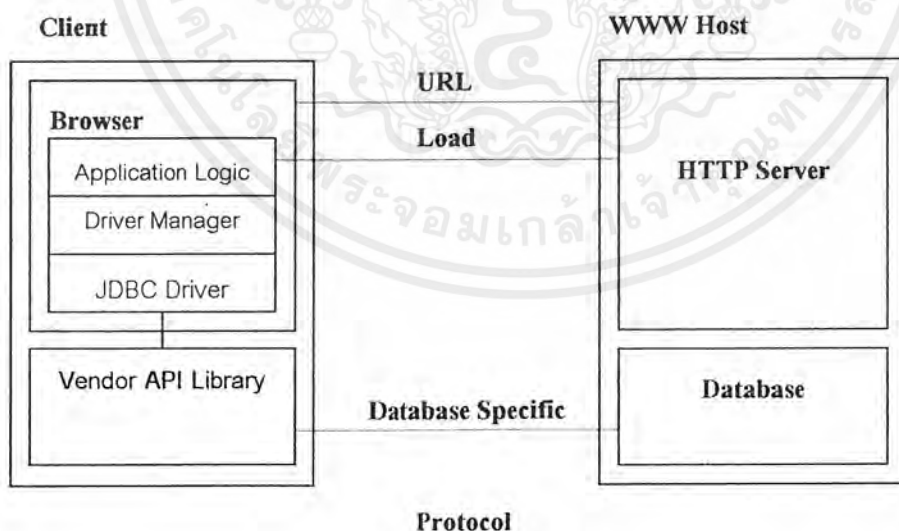
5.5.2 Native-API, partly Java driver

Native-API, partly Java driver ใช้ Vendor Library ในการแปลง JDBC function ไปยังคุณลักษณะของการใช้ภาษาในการ Query เช่นไลบรารีสำหรับออรากเคิล คือ ocilib การติดต่อกับ JDBC ประเภทนี้ จะเหมือนกับ JDBC/ODBC Bridge คือต้องติดตั้งโค้ดในเครื่องไคลเอนท์

ข้อดีของไคร์ฟเวอร์ชนิดนี้ คือ สามารถใช้ Native-API Driver ติดต่อกับฐานข้อมูลนั้นโดยตรง โดยผ่านโปรโตคอลเดิมที่ใช้อยู่ก่อนแล้ว ทำให้เหมาะสมกับการเชื่อมต่อฐานข้อมูลแบบ Two-tier Client Server นอกจากนี้ ยังเร็วกว่าไคร์ฟเวอร์แบบแรก เพราะว่ามีเลเซอร์พิเศษของการแปลงเป็น ODBC ถูกจำกัดออกไป

JDBC ↔ Native client-API ↔ database

แต่มีบางส่วนของ Native-API Driver ถูกเขียนขึ้นจากภาษา C++ (Partly-Java) จึงไม่สามารถดาวน์โหลดผ่านเครือข่ายอินเทอร์เน็ตได้ และไม่สามารถเชื่อมต่อฐานข้อมูลข้ามชนิดกันได้



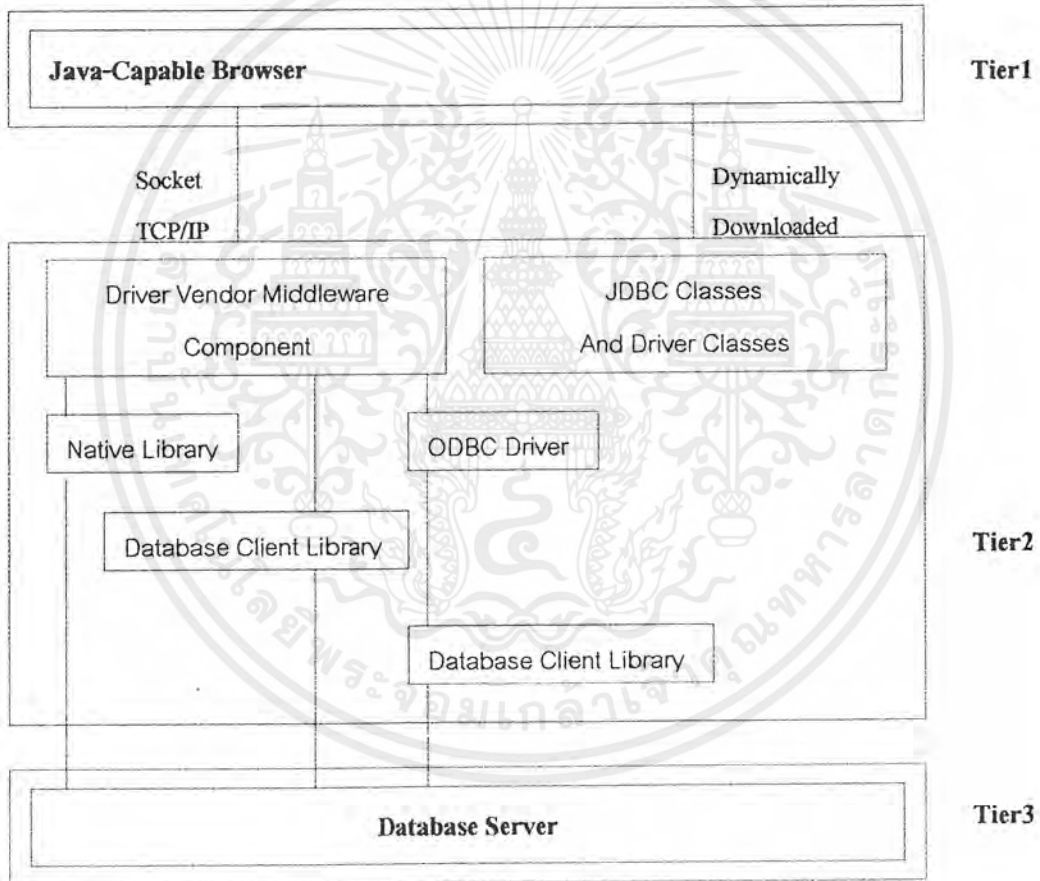
รูปที่ 5.6 การติดต่อกับ JDBC โดยรูปแบบที่ 2 Native-API, partly Java driver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5.3 Network-protocol, all-Java driver

Network-protocol, all-Java driver จะทำการแปลงการเรียกใช้ JDBC ให้อยู่ในรูปของเน็ตเวิร์ค โพรโทคอลร่วม (DBMS-independent network Protocol) ซึ่งหลังจากนั้นจะถูกแปลงให้อยู่ในรูปเฉพาะของแต่ละฐานข้อมูล (database-specific API) บนเซิร์ฟเวอร์นั้นๆ รูปแบบการเชื่อมต่อจะเป็นลักษณะของ three tier ไคล์เวอร์ชนิดนี้จะทำการเอ็ชคิวทีบนไคล์เอนท์ และส่งคำสั่ง SQL ไปยังเน็ตเวิร์ค เมื่อเซิร์ฟเวอร์ได้รับข้อมูล ก็จะจัดการการเชื่อมต่อที่มีมาไปยังฐานข้อมูล

ไคล์เวอร์ประเภทนี้เหมาะสมมากสำหรับระบบเครือข่ายอินเทอร์เน็ต-อินทราเน็ต และการทำงานที่มีผู้ใช้หลายคน เพราะมีความยืดหยุ่นคล่องตัวที่สุด เพราะว่าเขียนขึ้นจากภาษาจาวาทั้งหมด ทำให้ไม่จำเป็นต้องมีไคล์เวอร์ร่วมที่เขียนขึ้นจากภาษาอื่น ซึ่งต้องติดตั้งเฉพาะฝั่งไคล์เอนท์นั้น และสามารถรันบนระบบใดก็ได้ที่สนับสนุนสภาพแวดล้อมเสมือนของจาวาหรือ JVM (Java Virtual Machine)



รูปที่ 5.7 การติดต่อ JDBC โดยรูปแบบที่ 3 Network-protocol, all-Java driver

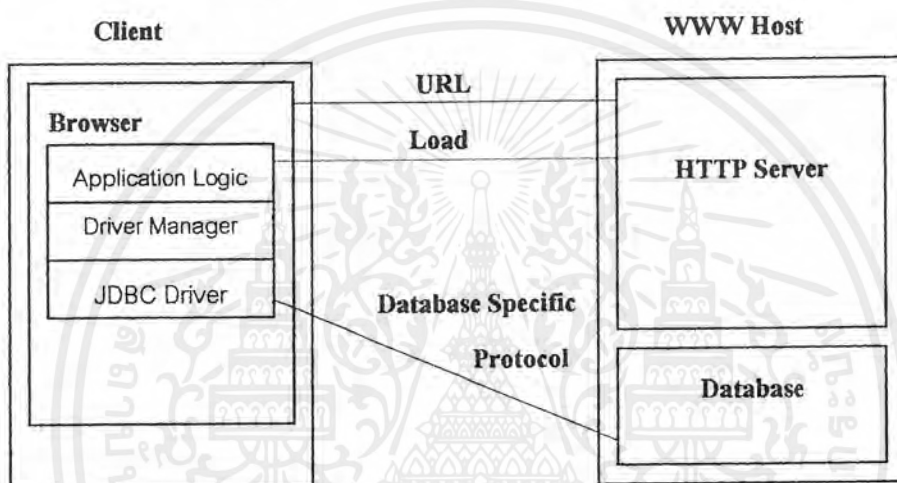
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5.4 Native-protocol, all-Java driver

Native-protocol, all-Java driver จะแปลงการเรียกใช้คำสั่งของ JDBC ให้อยู่ในรูปแบบของเน็ตเวิร์คโปรโตคอลเฉพาะของฐานข้อมูลนั้นโดยตรง เพราะใช้ไคร์ฟเวอร์ของตัวแทนจำหน่ายฐานข้อมูล ไคร์ฟเวอร์เหล่านี้สามารถเขียนในภาษาจาวา และติดต่อกับแอปพลิเคชันแบบ just-in-time เพราะว่าไคร์ฟเวอร์เหล่านี้จะแปลง JDBC ตรงไปตรง native protocol โดยปราศจากการใช้ ODBC หรือ Native APIs ซึ่งสามารถจัดหามาสำหรับการติดต่อกับฐานข้อมูลที่มีประสิทธิภาพสูง

ข้อดีของไคร์ฟเวอร์ชนิดนี้คือ ไม่ต้องมีการปรับเปลี่ยนระบบฐานข้อมูลเดิมที่ใช้งานอยู่แล้วในแต่ละองค์กร และไม่มีควมจำเป็นต้องติดตั้ง ไคร์ฟเวอร์ตัวกลาง

JDBC Driver ชนิดที่ 3 และ 4 คือไคร์ฟเวอร์ที่คาดว่าจะจะเป็นสิ่งที่ต้องการสำหรับการเชื่อมต่อกับฐานข้อมูลในอนาคต เพราะถูกเขียนขึ้นจากภาษาจาวาทั้งหมดซึ่งมีความปลอดภัยและคล่องตัวมากกว่า



รูปที่ 5.8 การติดต่อ JDBC โดยรูปแบบที่ 4 Native-protocol, all-Java driver

ออราเคิลมี JDBC Driver 2 ประเภท คือ

- JDBC Thin เป็น JDBC Driver ชนิดที่ 4 ซึ่งใช้ซ็อกเก็ต (socket) เชื่อมต่อโดยตรงกับออราเคิล โดยผ่านโปรโตคอล TCP/IP การติดต่อทำโดยใช้ภาษาจาวา ได้แก่ จาวาแอปพลิเคชันและจาวาแอปพลิเคชัน ทำให้ไคร์ฟเวอร์ชนิดนี้มีคุณสมบัติคือ ไม่ขึ้นกับแพลตฟอร์ม (platform-independent)
- JDBC OCI เป็น JDBC Driver ชนิดที่สอง ไคร์ฟเวอร์ชนิดนี้ใช้กับจาวาแอปพลิเคชัน

โครงการนี้ใช้ JDBC Thin Driver หรือ Native-protocol, all-Java driver ในการติดต่อกับฐานข้อมูลออราเคิล ขั้นตอนการเขียนภาษาจาวาให้ติดต่อกับออราเคิล โดยใช้ JDBC Thin Driver มีดังนี้

■ อิมพอร์ตคลาสของ JDBC

```
import java.sql.*;
import java.math.*;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รีจิสเตอร์ไดรฟ์เวอร์ JDBC

```
DriverManager.registerDriver (new oracle.jdbc.driver.OracleDriver ());
```

- เปิดการเชื่อมต่อกับฐานข้อมูล

ตัวอย่างการเชื่อมต่อกับฐานข้อมูลที่มีโฮสต์ชื่อว่า “myhost” และมี TCP/IP listener ที่พอร์ต “1521” ของฐานข้อมูลที่ชื่อว่า “orcl” โดยล็อกอินเป็นผู้ใช้ที่ชื่อ “scott” โดยมีพาสเวิร์ดคือ “tiger” ดังนี้

```
Connection conn =
```

```
    DriverManager.getConnection
```

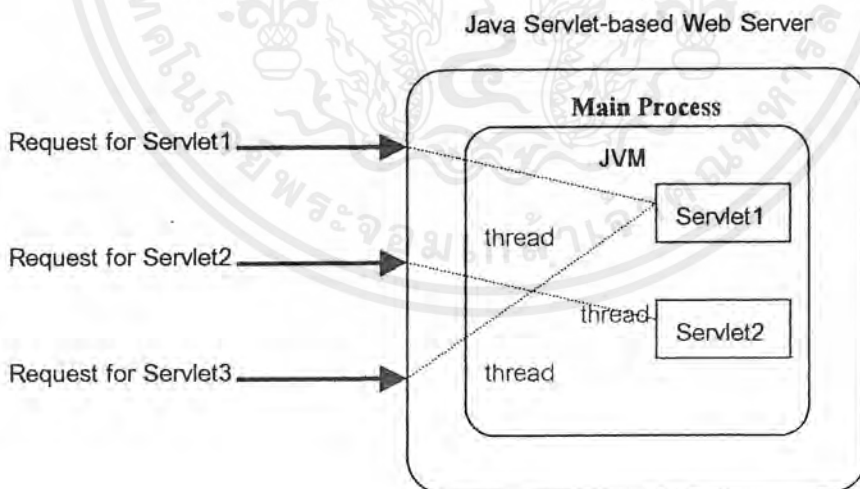
```
    (“jdbc:oracle:thin:@myhost:1521:orcl”, “scott”, “tiger”);
```

- ปิดการเชื่อมต่อกับฐานข้อมูล

```
conn.close();
```

5.6 จาวาเซิร์ฟเวสต์

เซิร์ฟเวสต์มีการทำงานแบบซิงโครนัส โดยทำงานบนจาวาเวอร์ชวลแมชชีน (JVM) บนเซิร์ฟเวอร์ ซึ่งต่างจากจาวาแอปเพล็ต คือ บราวเซอร์ไม่จำเป็นต้องสนับสนุนการใช้งาน และต่างจากซีจีไอตรงที่ ซีจีไอจะใช้ Multiple-process เพื่อจัดการกับโปรแกรมหลายๆ โปรแกรมและรีเควสหลายๆ ครั้ง ส่วนเซิร์ฟเวสต์จะจัดการโดยแยกเป็นหลายๆ เธรด (Thread) ในการประมวลผลของเว็บเซิร์ฟเวอร์



รูปที่ 5.9 การทำงานของจาวาเซิร์ฟเวสต์แบบมัลติเธรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.1 ความสามารถของเซิร์ฟเล็ต

- ทำงานข้ามแพลตฟอร์ม

เซิร์ฟเล็ตเป็นภาษาจาวา ซึ่งมีความสามารถในการทำงานข้ามแพลตฟอร์มได้ เช่นเราอาจพัฒนาเซิร์ฟเล็ตบนวินโดวส์แล้วนำไปรันบนลินุกซ์ได้

- ความสามารถของจาวา

เซิร์ฟเล็ตจะสามารถใช้ความสามารถของจาวาได้อย่างเต็มที่ เช่นการเข้าถึงระบบเน็ตเวิร์ก และ URL การทำมัลติเธรด การเชื่อมต่อกับฐานข้อมูล การใช้ remote method invocation (RMI)

- ความมีประสิทธิภาพ

เซิร์ฟเล็ตมีการทำงานอย่างมีประสิทธิภาพ โดยเมื่อโหลดเซิร์ฟเล็ตขึ้นมาในหน่วยความจำในลักษณะของ single object instance เซิร์ฟเล็ตจะจัดการการร้องขอพร้อมๆ กัน โดยใช้เธรด และสามารถคงสถานะเดิมของมันต่อไปได้ เช่นการเชื่อมต่อกับฐานข้อมูล เพราะตัวของเซิร์ฟเล็ตเองจะอยู่ในหน่วยความจำของเซิร์ฟเวอร์

- ความปลอดภัย

ในส่วนของเซิร์ฟเล็ตนั้นจะมีการรักษาความปลอดภัยที่สืบทอดมาจากคุณลักษณะของจาวา และยังมีคุณสมบัติในตัวมันเองของ API ด้วย มีระบบกำจัดข้อมูลที่เป็นขยะต่างๆ หรือพอยท์เตอร์ด้วยในการดูแลหน่วยความจำ มีการรองรับความผิดพลาดต่างๆ ที่อาจเกิดขึ้น โดยใช้ Exception-handling mechanism ซึ่งจะ throw exception โดยไม่ทำให้ระบบเสียหาย

- ความสะดวกในการพัฒนา

ในการเขียนจาวาเซิร์ฟเล็ตนั้นสามารถเรียก Servlet API มาใช้ได้ โดยมันจะประกอบไปด้วยคลาสต่างๆ ที่ช่วยในการทำงานของเซิร์ฟเล็ตทำให้พัฒนาโปรแกรมได้ง่ายขึ้น

- การทำงานร่วมกัน

เซิร์ฟเล็ตสามารถสร้างการทำงานร่วมกันได้อย่างสมบูรณ์กับตัวเซิร์ฟเวอร์ เช่น ใช้เซิร์ฟเวอร์ในการแปลงพารามิเตอร์ของข้อมูล แสดงล็อกไฟล์หรือ MIME type

- ความยืดหยุ่น

Servlet API นั้นถูกออกแบบมาเพื่อรองรับการขยายงานในรูปแบบต่างๆ ได้ง่าย เช่นในปัจจุบันนี้มีการพัฒนาเพื่อรองรับการทำงานสำหรับ HTTP Servlet

5.6.2 พื้นฐานของ Http Servlet

พื้นฐานของ Http Requests, Response และ Headers นั้น การทำงานของโปรโตคอล Http นั้นฝั่งไคลเอนท์(ในที่นี้คือเว็บเบราว์เซอร์)จะทำการรีเควส และเว็บเซิร์ฟเวอร์ก็จะตอบสนองกลับมา ในการส่งรีเควสของไคลเอนท์ สิ่งแรกที่ต้องระบุถึง คือ Http command ซึ่งเป็นเมธอดที่จะบอกเซิร์ฟเวอร์ถึงเอนต์รี่ที่มันจะทำ บรรทัดแรกของการรีเควสจะกำหนด URL และเวอร์ชันของโปรโตคอล Http ดังตัวอย่าง

```
GET /intro.html HTTP/1.0
```

ในการรีควีสต์นี้จะใช้เมธอด GET เพื่อร้องขอ intro.html โดยใช้ Http เวอร์ชัน 1.0 หลังจากส่งรีควีสต์แล้วไคลเอนต์อาจส่งข้อมูลอื่นๆ เช่น ซอฟต์แวร์ที่ไคลเอนต์ทำงานอยู่ ดังตัวอย่าง

```
User-Agent:Mazilla/4.0 (Compatible; MSIE4.0;Windows95)
```

หลังจากได้รับรีควีสต์ของไคลเอนต์แล้ว เซอร์เวอร์จะทำการประมวลผลและส่ง response กลับไป บรรทัดแรกของการตอบสนองจะแสดงสถานะ(status line) ที่ระบุเวอร์ชันของ โปรโตคอล HTTP เช่น

```
HTTP/1.0 200 OK
```

หลังจากรับ status line แล้ว เซิร์ฟเวอร์จะส่ง response เพื่อบอกรายละเอียดเกี่ยวกับตัวมัน ดังตัวอย่าง

```
Date          : Saturday,23-May-98 03:25:12 GMT
Server        : JavaWebServer/1.1.1
MIME-version  : 1.0
Content-type  : text/html
Content-length : 1029
Last-modified : Thursday, 7-May-98 12:15:35
```

5.6.3 วงจรชีวิตของเซิร์ฟเล็ต

วงจรชีวิตของเซิร์ฟเล็ต ประกอบด้วย

1. สร้างและกำหนดค่าเริ่มต้นของเซิร์ฟเล็ต
2. จัดการบริการให้กับไคลเอนต์
3. ทำลายเซิร์ฟเล็ตนั้น

โดยที่เซิร์ฟเล็ตจะถูกสร้าง, โหลด และกำหนดค่าเริ่มต้นโดยจาวาเวอร์ชวลแมชชีน ซึ่งเซิร์ฟเล็ตเป็นชิ้นส่วนใหญ่มักจะประมวลผลเซิร์ฟเล็ตทั้งหมดที่จาวาเวอร์ชวลแมชชีนใน Single-process Multithreaded Web Server ที่เขียนโดยใช้ภาษาอื่น จาวาเวอร์ชวลแมชชีนจะถูกฝังไว้ภายในโทรมเซตของเซิร์ฟเวอร์นั้น ส่วน Multiprocess Web Server ที่ทำงานหลายๆ โทรมเซต เพื่อรองรับเซอร์เวอร์ไม่จำเป็นต้องฝังจาวาเวอร์ชวลแมชชีนเข้าไป โดยมันทำงานด้วยจาวาเวอร์ชวลแมชชีนภายนอก (external JVM) โดยแต่ละโทรมเซตจะมาใช้ร่วมกันแทน

Init and Destroy

เซิร์ฟเล็ตจะมีเมธอด init() และ destroy() โดยเซิร์ฟเวอร์จะเรียกเมธอด init() หลังจากสร้างอินสแตนซ์ของเซิร์ฟเล็ตขึ้นมา โดยขึ้นอยู่กับเว็บเซิร์ฟเวอร์และการติดตั้งของมัน

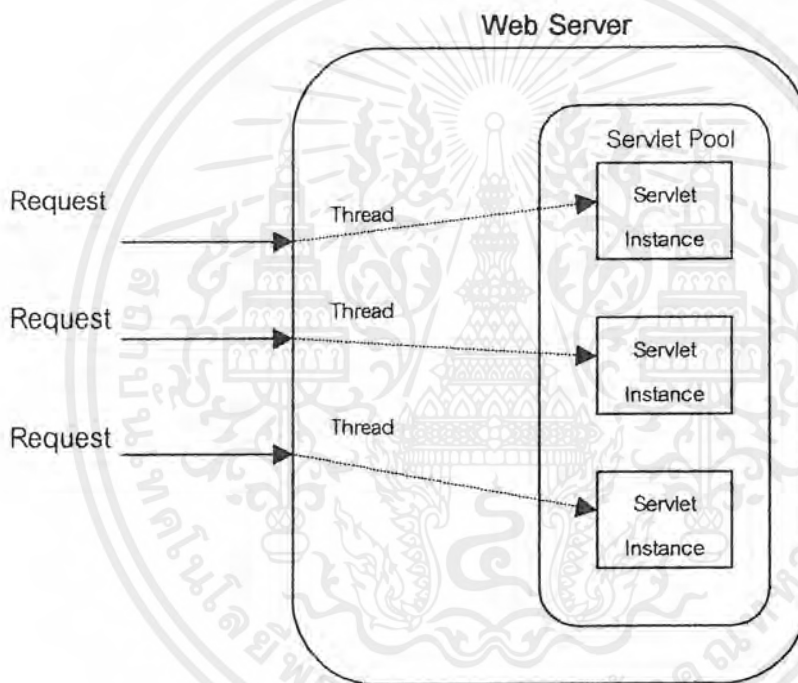
เมธอด `init()` ใช้เพื่อกำหนดค่าเริ่มต้นของเซิร์ฟเล็ต เมธอด `init()` จะถูกเรียกเมื่อ

- เซิร์ฟเวอร์เริ่มทำงาน
- เซิร์ฟเล็ตถูกร้องขอเป็นครั้งแรกก่อนเมธอด `service()` จะถูกเรียก
- โดยการเรียกของผู้ดูแลเซิร์ฟเวอร์เท่านั้น

เมธอด `destroy()` ถูกเรียกเพื่อให้เซิร์ฟเล็ตปล่อยทรัพยากรทั้งหมดที่มัน ใช้อยู่

Single-Thread Model

เซิร์ฟเล็ตสามารถมี pool สำหรับอินสแตนซ์ที่สร้างขึ้นมา และแบ่งหน้าที่กันจัดการกับการร้องขอที่เรียกเข้ามาโดยการอิมพลิเมนต์ `javax.servlet.SingleThreadModel` อินเตอร์เฟส โดยเซิร์ฟเวอร์ที่โหลด `SingleThreadModel` จะมีเพียงเซรเคเดียวที่ทำงานในเวลาใดเวลาหนึ่งเท่านั้น โดยจะไม่มีการทำงานของเซรเคสองตัวขึ้นไปพร้อมกัน โดยเซรเคจะใช้งานเซิร์ฟเล็ตอินสแตนซ์จาก pool ดังรูปที่ 5.10



รูปที่ 5.10 เซรเคใช้งานเซิร์ฟเล็ตจากเซิร์ฟเล็ตพูล

5.6.4 GET และ POST

เมื่อไคลเอนต์ติดต่อกับเซิร์ฟเวอร์และทำ HTTP request รูปแบบของการร้องขอมีหลายรูปแบบที่พบบ่อยๆ คือ เมธอด GET และ POST

เซิร์ฟเล็ตจะใช้คลาสและอินเตอร์เฟสจาก 2 แพคเกจ คือ `javax.servlet` และ `javax.servlet.http` การเขียนเซิร์ฟเล็ตคืออิมพลิเมนต์อินเตอร์เฟส `javax.servlet.Service` หรือ `javax.servlet.http.HttpServlet`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละครั้งที่เซิร์ฟเวอร์ส่งผ่านการร้องขอ (request) ไปยังเซิร์ฟเล็ตมันจะไปเรียกเมธอด service() ของเซิร์ฟเวอร์

พื้นฐานอย่างหนึ่งของ HTTP เซิร์ฟเล็ต คือการแสดงผลหน้าเว็บเพจ HTML ซึ่งมันจะสามารถทำงานได้เหมือนกับซีจีไอทั้งด้านการทำฟอร์ม HTML หรือการติดต่อกับฐานข้อมูล คือตัวอย่างจะเป็นการใช้เซิร์ฟเล็ตเขียน โปรแกรมให้แสดงออกมาในรูปแบบของ HTML

ตัวอย่าง โปรแกรม Hello World

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWorld extends HttpServlet
{
    public void doGet(HttpServletRequest req,HttpServletResponse res)
    throws ServletException,IOException
    {
        res.setContentType("text/html");
        PrintWriter out=res.getWriter();
        out.println("<HTML>");
        out.println("<HEAD><TITLE>Hello World</TITLE></HEAD>");
        out.println("<BODY>");
        out.println("<BIG>Hello World<BIG>");
        out.println("</BODY></HTML>");
    }
}
```

เซิร์ฟเล็ตจะเอ็กซ์เทนดึคลาส HttpServlet และโอเวอร์โหลดเมธอด doGet() แต่ละครั้งที่เซิร์ฟเวอร์ได้รับรีควีสแบบ GET แล้วเซิร์ฟเวอร์จะเรียกเมธอด doGet() เพื่อผ่านค่าของอ็อบเจ็กต์ HttpServletRequest และ HttpServletResponse

HttpServletRequest จะรองรับการร้องขอต่างๆของไคลเอนท์ โดยอ็อบเจ็กต์นี้จะเข้าถึงข้อมูลเกี่ยวกับไคลเอนท์ และพารามิเตอร์ที่เกี่ยวข้องกับการรีควีส, HttpHeaders

HttpServletResponse จะรองรับการตอบสนองของเซิร์ฟเล็ต เซิร์ฟเล็ตจะใช้อ็อบเจ็กต์เพื่อคืนข้อมูลไปยังไคลเอนท์ และยังสามารถใช้อ็อบเจ็กต์เพื่อการเซต HttpServletResponse header ด้วย

เมธอดที่ตัวอย่างโปรแกรมใช้คอนแรก คือเมธอด setContentType() ของอ็อบเจ็กต์ response เพื่อคิงค่าของ content type ของการตอบสนองที่เป็น "text/html" และใช้เมธอด getWriter() เพื่อเก็บค่าของ PrintWriter ในการที่จะส่ง "Hello World" ในรูปแบบของ HTML ไปยังไคลเอนท์

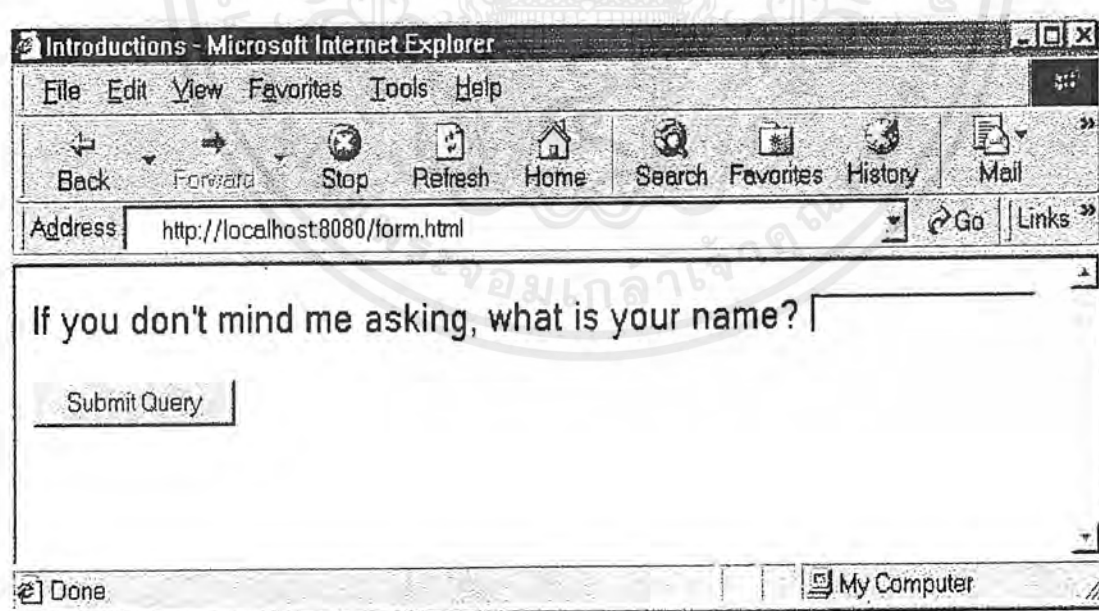
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซิร์ฟเว็ตสามารถแสดงการรับข้อมูลจากฟอร์มของ HTML ได้ โดยขั้นแรกเราจะมีฟอร์มในภาษา HTML ดังนี้

```
<HTML>
<HEAD>
<TITLE> Introductions </TITLE>
</HEAD>
<BODY>
<FORM METHOD=GET ACTION="/servlet/Hello">
If you don't mind me asking, what is your name?
<INPUT TYPE=TEXT NAME="name"><P>
<INPUT TYPE=SUBMIT>
</FORM>
</BODY>
</HTML>
```

โดยผลลัพธ์จะออกมาเป็นหน้าตาของเพจ ดังรูปที่ 5.11 ซึ่งแสดงการรับข้อมูลจากฟอร์มของ HTML เมื่อผู้ใช้คลิกปุ่ม (submit) ฟอร์ม ข้อมูลที่กรอกจะส่งไปยังเซิร์ฟเว็ตที่ชื่อ Hello Servlet ซึ่งจะใช้เมธอด GET ข้อมูลที่ส่งจะ ไปต่อท้าย URL

เช่น <http://server:8080/servlet/Hello?name=Saisuda+Thanathammanunt>



รูปที่ 5.11 เว็บเพจแสดงฟอร์มรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง โปรแกรม Hello

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Hello extends HttpServlet {

    public void doGet(HttpServletRequest req,HttpServletResponse res)
    throws ServletException,IOException
    {
        res.setContentType("text/html");

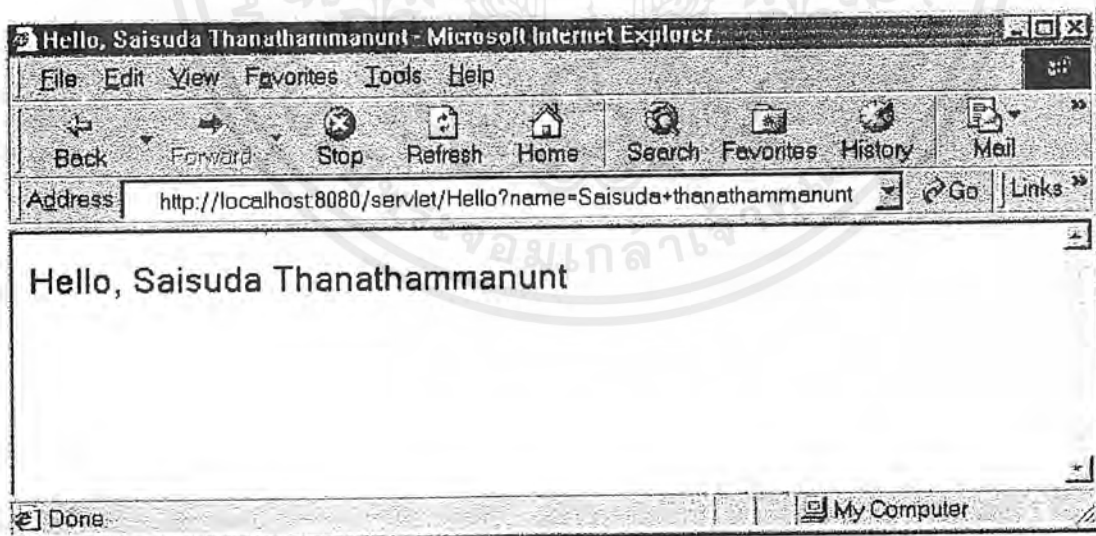
        PrintWriter out=res.getWriter();

        String name=req.getParameter("name");

        out.println("<HTML>");
        out.println("<HEAD><TITLE>Hello, "+ name + "</TITLE></HEAD>");
        out.println("<BODY>");
        out.println("Hello, "+ name);
        out.println("</BODY></HTML>");
    }
}

```

เมธอด req.getParameter("name") จะใช้เป็นตัวแปรรับค่า name ของผู้ใช้ โดยผลของโปรแกรม จะได้ดังรูปที่ 5.12



รูปที่ 5.12 เว็บเพจได้ตอบกับผู้ใช้หลังจากกรอกชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ารับเมธอด POST จะเขียนเป็นเมธอดในรูปของเซิร์ฟเล็ตได้ ดังนี้

```
public void doPost(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {
}
}
```

โดยในฟอร์มของการส่ง HTML นั้นจะอยู่ในรูปแบบ

```
<FORM METHOD=POST ACTION="/servlet/Hello">
```

5.6.5 การเชื่อมต่อกับฐานข้อมูล

ตามที่ได้อธิบายถึงการเชื่อมต่อกับฐานข้อมูลด้วย JDBC มาแล้วในบทข้างต้น บทนี้เราจะยกตัวอย่างการนำโปรแกรมเซิร์ฟเล็ตมารวมกับการเชื่อมต่อกับฐานข้อมูล ดังตัวอย่างต่อไปนี้ เป็นโปรแกรมที่เอาข้อมูลจากฐานข้อมูลขึ้นมาแสดงบนเว็บเพจ โดยนำการเชื่อมต่อกับฐานข้อมูลในลักษณะของ JDBC มาใช้ร่วมกับเซิร์ฟเล็ตด้วย ผลที่ได้แสดงดังรูปที่ 5.13

```
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class DBPhoneLookup extends HttpServlet {
    public void doGet(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {
        Connection con = null;
        Statement stmt = null;
        ResultSet rs = null;
        res.setContentType("text/html");
        PrintWriter out = res.getWriter();

        try {
            // Load (and therefore register) the Oracle Driver
            Class.forName("oracle.jdbc.driver.OracleDriver");
            // DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
            // Get a Connection to the database
            con = DriverManager.getConnection(
                "jdbc:oracle:thin:@aqua:1521:ORCL", "scott", "tiger");
            // Create a Statement object
            stmt = con.createStatement();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Execute an SQL query, get a ResultSet
rs = stmt.executeQuery("SELECT * FROM dept");

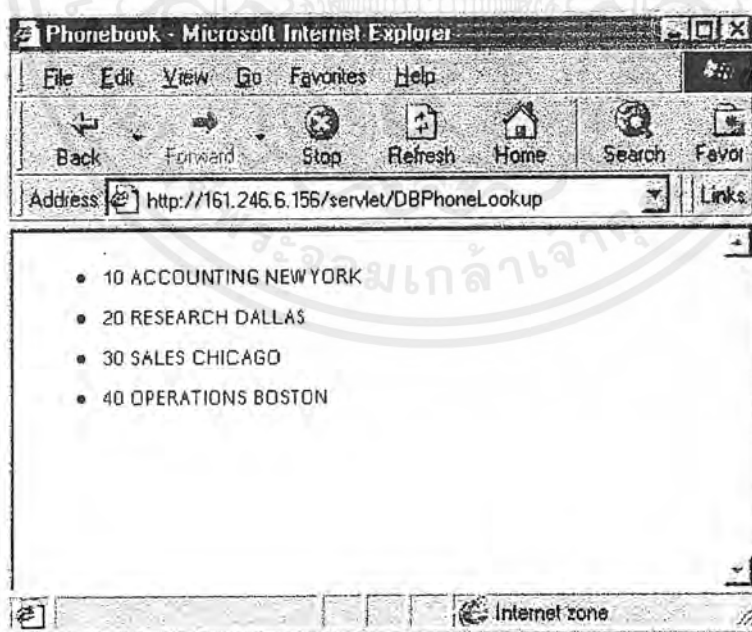
// Display the result set as a list
out.println("<HTML><HEAD><TITLE>Phonebook</TITLE></HEAD>");
out.println("<BODY>");
out.println("<UL>");

while(rs.next() )
out.println("<LI> " + rs.getString("DEPTNO") + " " +rs.getString("DNAME")+ " " +rs.getString("LOC")); }

    out.println("</UL>");
    out.println("</BODY></HTML>");
}

catch(ClassNotFoundException e) {
    out.println("Couldn't load database driver: " + e.getMessage()); }
catch(SQLException e) {
    out.println("SQLException caught: " + e.getMessage()); }
finally { // Always close the database connection.
    try {
        if (con != null) con.close(); }
    catch (SQLException ignored) { }
}
}}

```



รูปที่ 5.13 หน้าจอเว็บเพจแสดงข้อมูลในฐานข้อมูลเมื่อเรียกคลาส DBPhoneLookup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

หลักการออกแบบ

6.1 การออกแบบฐานข้อมูล

6.1.1 ไนแอม (NIAM –Nijssens ‘s Information Analysis Methodology) ของฐานข้อมูล

ระบบฐานข้อมูลในโครงการนี้มีข้อมูลเกี่ยวกับสิ่งต่างๆ ดังต่อไปนี้

- ข้อมูลเกี่ยวกับอะไหล่ โดยจะมี ATTRIBUTE คือ SP_CODE, SP_NAME, SP_MASTER, SP_MENUFAC, SP_FB, SP_LR, SP_UD, SP_PRICE, SP_STOCK , SP_FEA, SP_DES, SP_PIC และ SP_PICCODE
- ข้อมูลเกี่ยวกับรถยนต์ ซึ่งแสดงรายละเอียดเกี่ยวกับบริษัทผู้ผลิตรถยนต์, รุ่นของรถ และปีที่ออกรุ่นรถ โดยจะมี ATTRIBUTE คือ CAR_CODE, CAR_MANUFACTURER, MODEL และ YEAR
- ความสัมพันธ์ของข้อมูลที่บอกว่าอะไหล่แต่ละชิ้นใช้ได้กับรถยนต์รุ่นไหน
- ข้อมูลเกี่ยวกับรายการขายแต่ละรายการที่เกิดขึ้น ทำให้รู้ว่าขายอะไหล่อะไร ไปจำนวนเท่าไร และรายการขายนี้อยู่ในใบกำกับสินค้าใบไหน โดยจะมี ATTRIBUTE คือ SALE_ID, INVOICE_ID, SP_CODE และ ITEMS
- ข้อมูลเกี่ยวกับใบกำกับสินค้า ซึ่งมีรายละเอียดของใบกำกับสินค้าว่าออกให้ลูกค้าคนไหน, วันที่ออกใบกำกับสินค้า, ยอดรวมราคาของใบกำกับสินค้า โดยจะมี ATTRIBUTE คือ INVOICE_ID, CUSTOMER_ID, INVOICE_DATE และ TOTAL_PRICE
- ข้อมูลเกี่ยวกับลูกค้า เก็บรายละเอียดของลูกค้าว่าลูกค้าเป็นสมาชิกหรือเป็นลูกค้าทั่วไป ถ้าเป็นสมาชิกต้องมี USERNAME และ PASSWORD เพื่อให้สมาชิกสามารถล็อกอินเข้าระบบ นอกจากนี้ ยังเก็บรายละเอียดอื่นๆ อีก เช่น ชื่อ, เบอร์โทรศัพท์, ที่อยู่ ฯลฯ โดยจะมี ATTRIBUTE คือ CUTOMER_ID, TYPE, USERNAME, PASSWORD, FULLNAME, TELEPHONE, EMAIL, ADDRESS และ POSTCODE

6.1.2 ตารางฐานข้อมูลในโครงการ

INVOICE_ID	CUSTOMER_ID	INVOICE_DATE	TOTAL_PRICE
------------	-------------	--------------	-------------

SALE_ID	INVOICE_ID	SP_CODE	ITEMS
---------	------------	---------	-------

CUSTOMER_ID	TYPE	USERNAME	PASSWORD	FULLNAME	TELEPHONE	EMAIL	ADDRESS	POSTCODE
-------------	------	----------	----------	----------	-----------	-------	---------	----------

CAR_CODE	CAR_MANUFACTURER	MODEL	YEAR
----------	------------------	-------	------

SP_CODE	CAR_CODE
---------	----------

SP_CODE	SP_NAME	SP_MASTER	SP_MENUFAC	SP_FB	SP_LR	SP_UD	SP_PRICE	SP_STOCK	SP_FEA	SP_DES	SP_PIC	SP_PICCODE
---------	---------	-----------	------------	-------	-------	-------	----------	----------	--------	--------	--------	------------

ตารางที่ 6.1 ตารางแสดงรีเลชันของระบบที่ได้จากการแปลงในแอมโอดีอะแกรม

6.1.3 คำอธิบายตารางฐานข้อมูล (Data Dictionary)

ชื่อ Field	Null ?	ชนิด	รายละเอียด
INVOICE_ID	NOT NULL	CHAR(4)	หมายเลขใบกำกับสินค้า (เป็นคีย์หลัก)
CUSTOMER_ID	NOT NULL	CHAR(4)	หมายเลขลูกค้า
INVOICE_DATE	NOT NULL	DATE	วันที่ออกใบกำกับสินค้า
TOTAL_PRICE	NOT NULL	SMALLINT	ยอดรวมราคาของใบกำกับสินค้า

ตารางที่ 6.2 Data Dictionary ของตาราง INVOICE

ชื่อ Field	Null ?	ชนิด	รายละเอียด
SALE_ID	NOT NULL	CHAR(6)	หมายเลขรายการขาย (เป็นคีย์หลัก)
INVOICE_ID	NOT NULL	CHAR(4)	หมายเลขใบกำกับสินค้า
SP_CODE	NOT NULL	CHAR(5)	รหัสอะไหล่
ITEMS	NOT NULL	SMALLINT	จำนวนชิ้นของอะไหล่ที่สั่งซื้อในรายการขาย

ตารางที่ 6.3 Data Dictionary ของตาราง SALE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ Field	Null ?	ชนิด	รายละเอียด
CUTOMER_ID	NOT NULL	CHAR(4)	หมายเลขลูกค้า (เป็นคีย์หลัก)
TYPE	NOT NULL	CHAR(1)	ชนิดของลูกค้า 0 : สมาชิก 1 : ลูกค้าทั่วไป
USERNAME		VARCHAR2(20)	ชื่อที่ใช้ในการล็อกอินเข้าระบบของสมาชิก
PASSWORD		VARCHAR2(20)	พาสเวิร์ดที่ใช้ในการล็อกอินเข้าระบบของสมาชิก
FULLNAME	NOT NULL	VARCHAR2(60)	ชื่อเต็มของลูกค้า
TELEPHONE	NOT NULL	VARCHAR2(20)	เบอร์โทรศัพท์ของลูกค้า
EMAIL		VARCHAR2(30)	อีเมลของลูกค้า
ADDRESS	NOT NULL	VARCHAR2 (100)	ที่อยู่ของลูกค้า (ที่อยู่จัดส่งสินค้าไปให้)
POSTCODE	NOT NULL	VARCHAR2(20)	รหัสไปรษณีย์ของลูกค้า

ตารางที่ 6.4 Data Dictionary ของตาราง CUSTOMER

ชื่อ Field	Null ?	ชนิด	รายละเอียด
CAR_CODE	NOT NULL	CHAR(4)	รหัสรถ (เป็นคีย์หลัก)
CAR_MANUFACTURER	NOT NULL	VARCHAR2(15)	บริษัทผู้ผลิตรถ
MODEL	NOT NULL	VARCHAR2(15)	รุ่นของรถ
YEAR	NOT NULL	SMALLINT	ปีของรถ

ตารางที่ 6.5 Data Dictionary ของตาราง CAR

ชื่อ Field	Null ?	ชนิด	รายละเอียด
SP_CODE	NOT NULL	CHAR(5)	รหัสอะไหล่
CAR_CODE	NOT NULL	CHAR(4)	รหัสรถ

ตารางที่ 6.6 Data Dictionary ของตาราง MATCH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ Field	Null ?	ชนิด	รายละเอียด
SP_CODE	NOT NULL	CHAR(5)	รหัสอะไหล่ (เป็นคีย์หลัก)
SP_NAME	NOT NULL	VARCHAR2(30)	ชื่ออะไหล่
SP_MASTER		VARCHAR2(15)	ชิ้นส่วนหลักของอะไหล่
SP_MENUFAC	NOT NULL	VARCHAR2(15)	ชื่อบริษัทผู้ผลิตอะไหล่
SP_FB		CHAR(1)	ตำแหน่งของอะไหล่เทียบกับรถ F : หน้า B : หลัง
SP_LR		CHAR(1)	ตำแหน่งของอะไหล่เทียบกับรถ L : ซ้าย R : ขวา
SP_UD		CHAR(1)	ตำแหน่งของอะไหล่ U : บน D : ล่าง
SP_PRICE	NOT NULL	SMALLINT	ราคาอะไหล่
SP_STOCK	NOT NULL	SMALLINT	จำนวนอะไหล่คงเหลือในสต็อก
SP_FEA		VARCHAR2(15)	ลักษณะของอะไหล่ เช่น ไฟ, คิว, กระจก ฯลฯ
SP_DES		VARCHAR2(30)	คำอธิบายอะไหล่เพิ่มเติม เช่น เก้า, สีขาว, หรือ อะไหล่มาจาก ไต้หวัน เป็นต้น
SP_PIC		VARCHAR2(8)	รูปของอะไหล่
SP_PICCODE		VARCHAR2(45)	รหัสรูปของอะไหล่

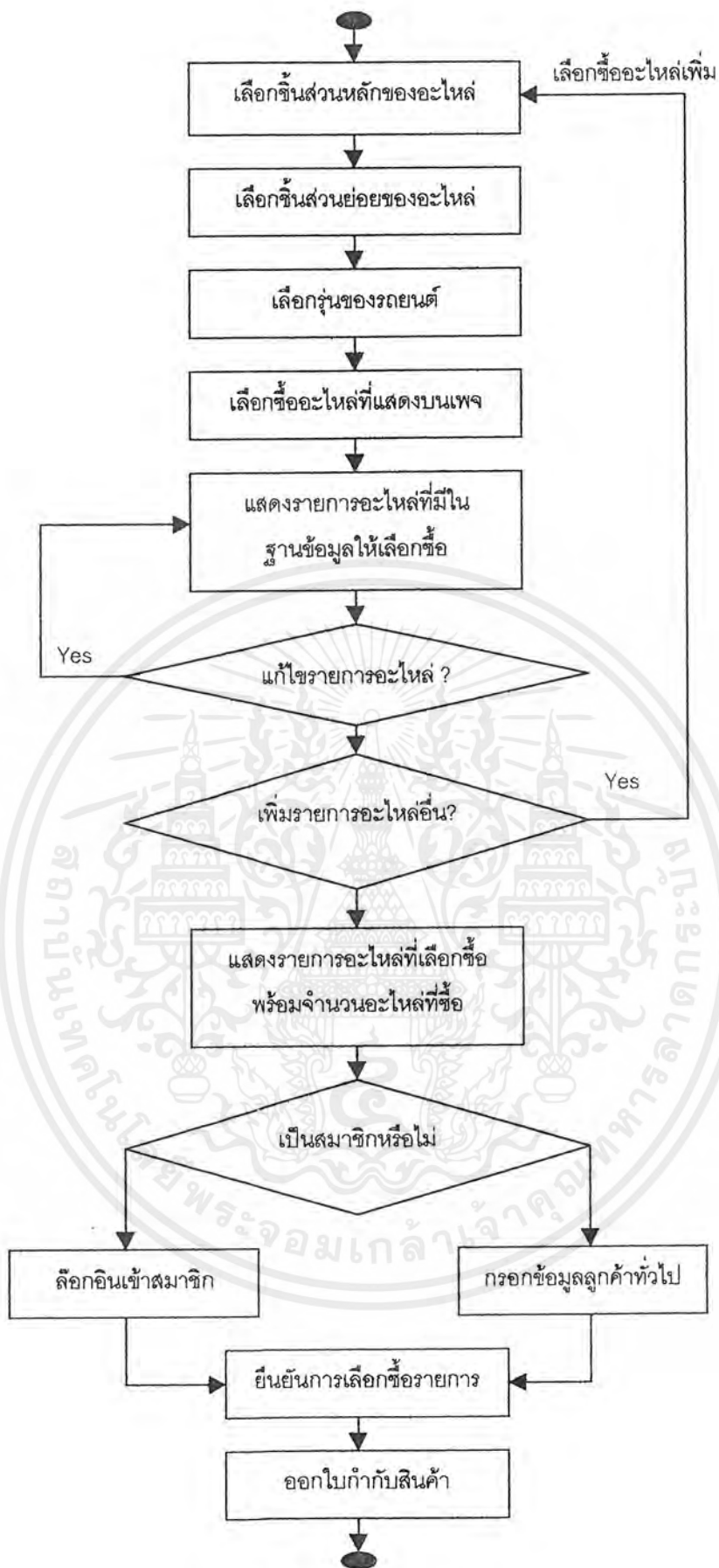
ตารางที่ 6.7 Data Dictionary ของตาราง SPARE_PART

6.2 การออกแบบเว็บเพจ

6.2.1 แผนภาพแสดงการเชื่อมต่อระหว่างเว็บเพจ

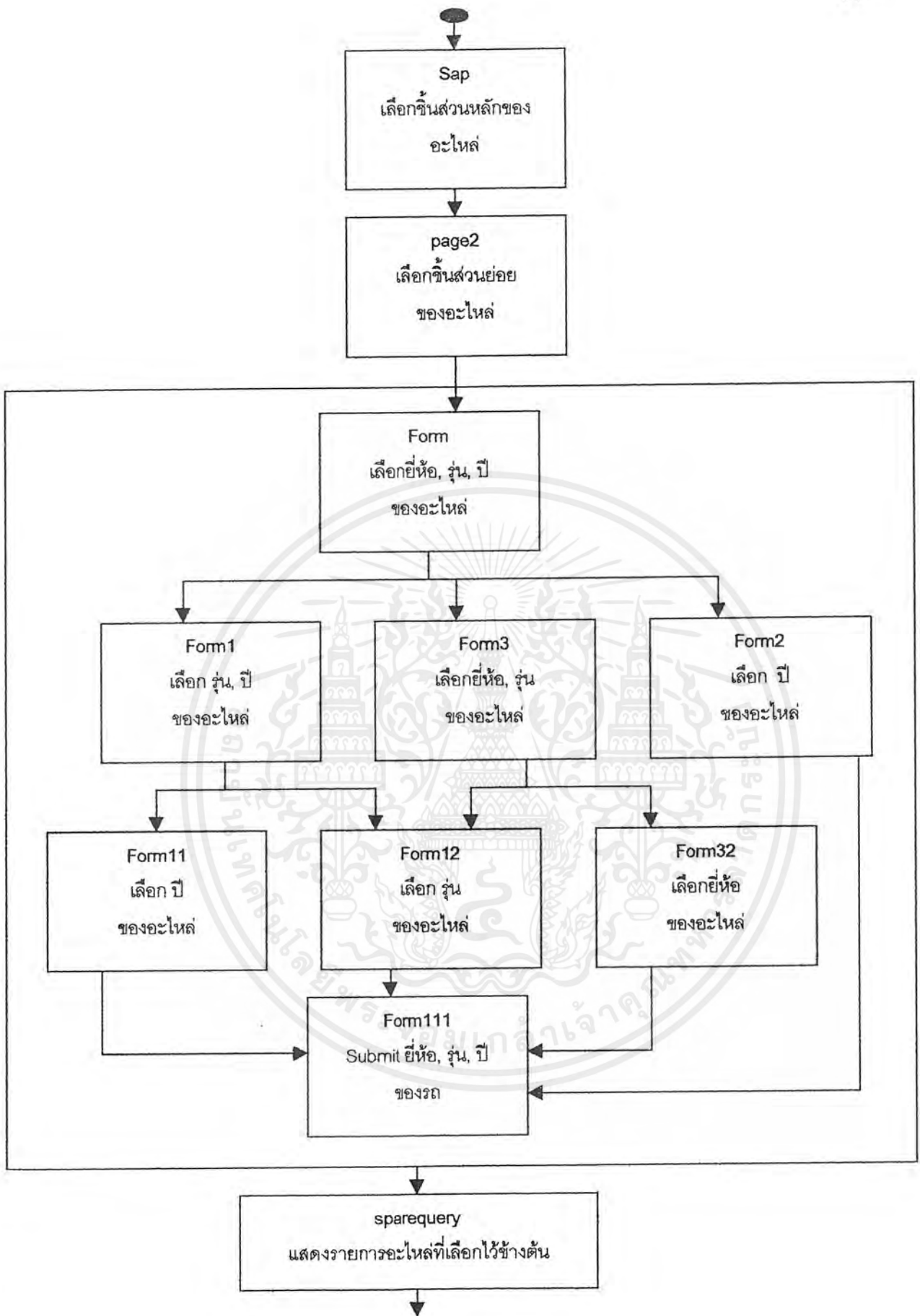
การเลือกซื้อและสั่งซื้ออะไหล่สามารถแบ่งเป็นขั้นตอน ดังรูปที่ 6.2 ส่วนแผนภาพแสดงการเชื่อมต่อระหว่างเว็บเพจและแนวทางในการสร้างเว็บไซต์ แสดงดังรูปที่ 6.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



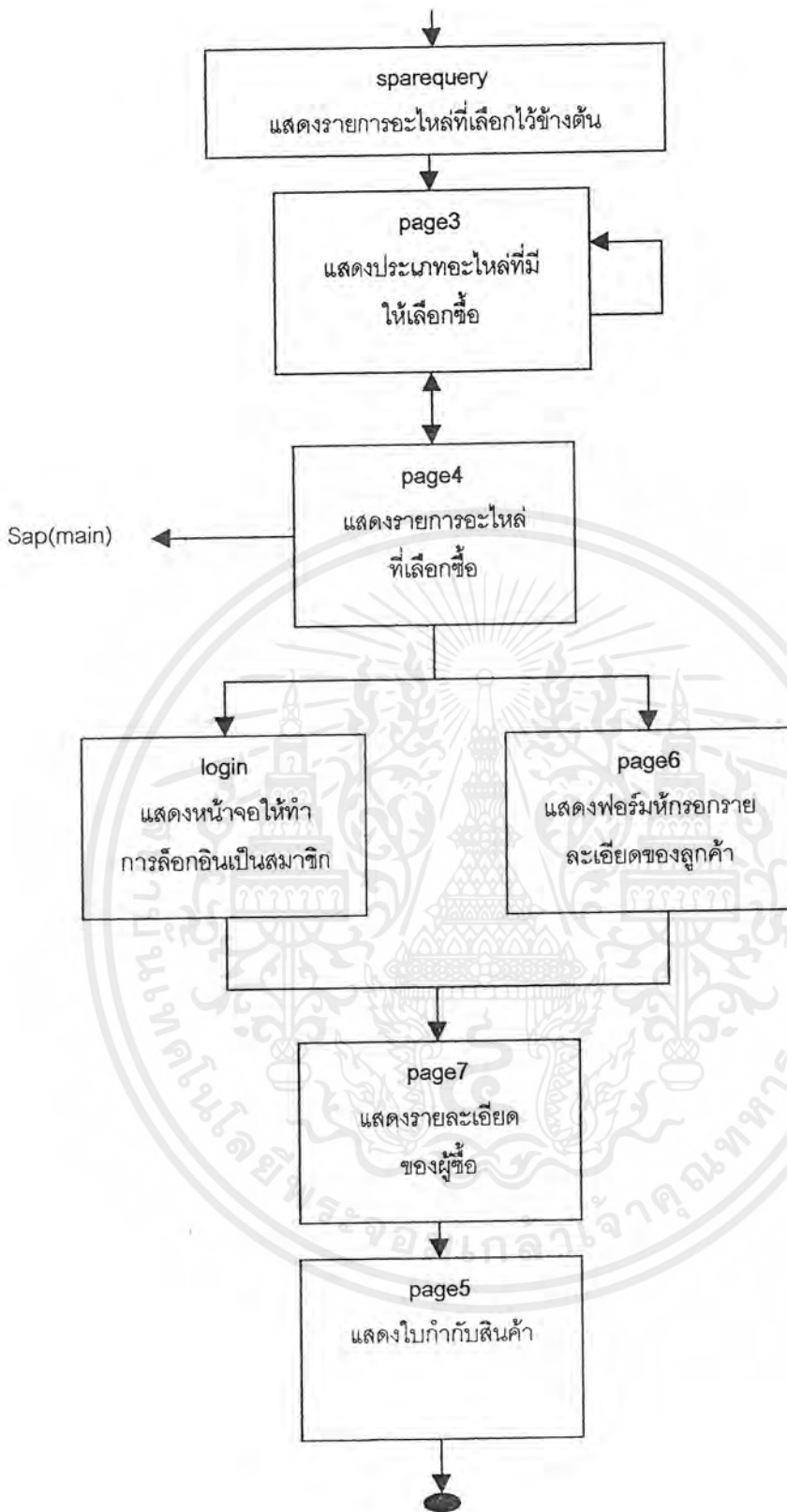
รูปที่ 6.2 ขั้นตอนการสั่งซื้ออะไหล่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.3 การเชื่อมต่อระหว่างเว็บเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

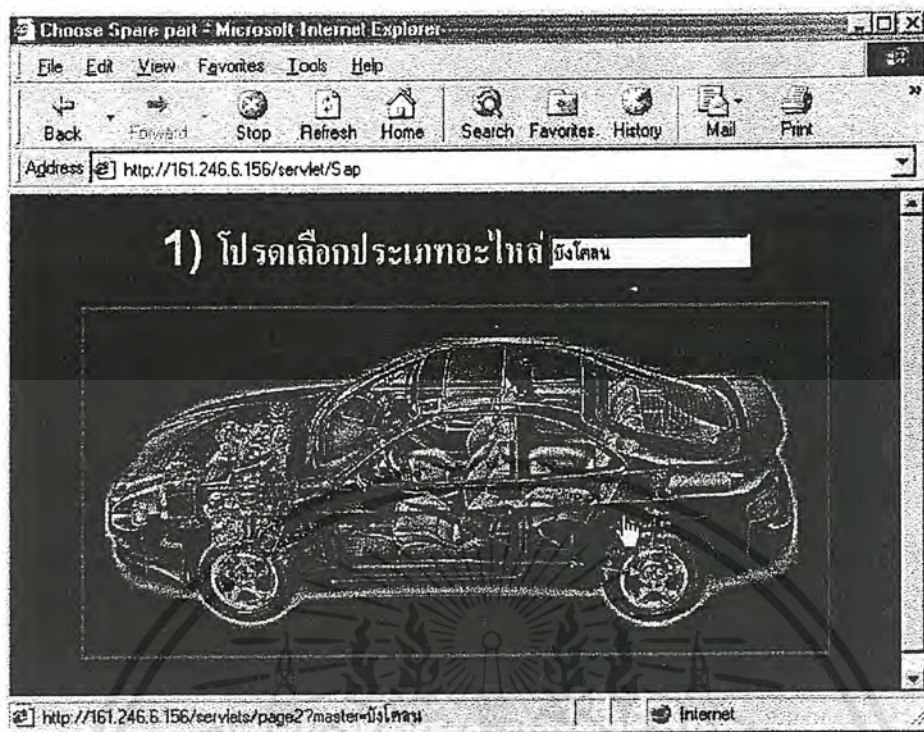


รูปที่ 6.3(ต่อ) การเชื่อมต่อระหว่างเว็บเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

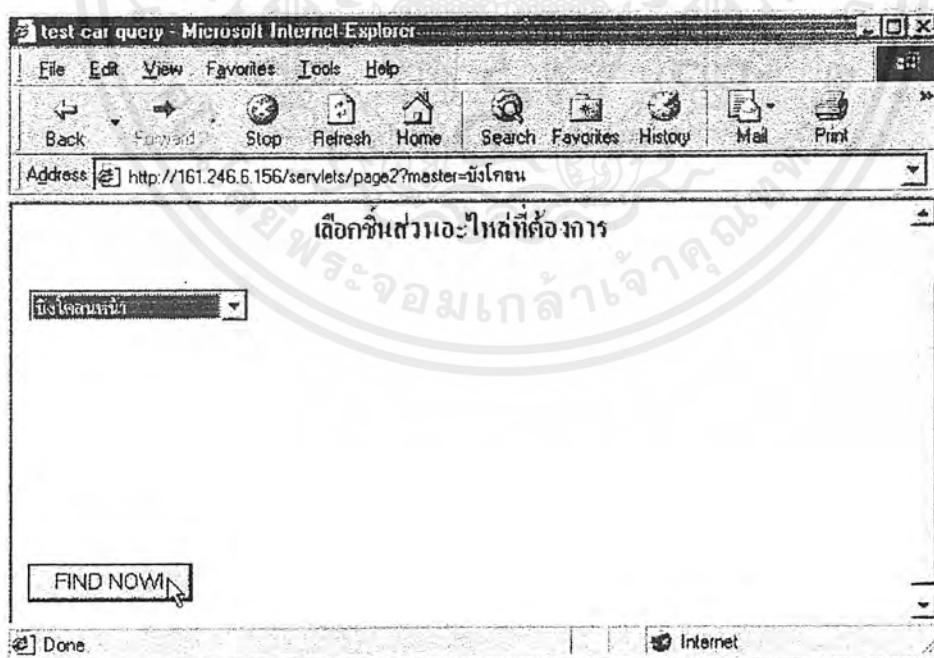
6.2.2 เว็บเพจในโครงการงาน

หน้าจอ โฮมเพจของ โครงการงานมีดังนี้



รูปที่ 6.4 เว็บเพจแสดงขึ้นรูปภาพเพื่อให้เลือกชิ้นส่วนของอะไหล่รถ

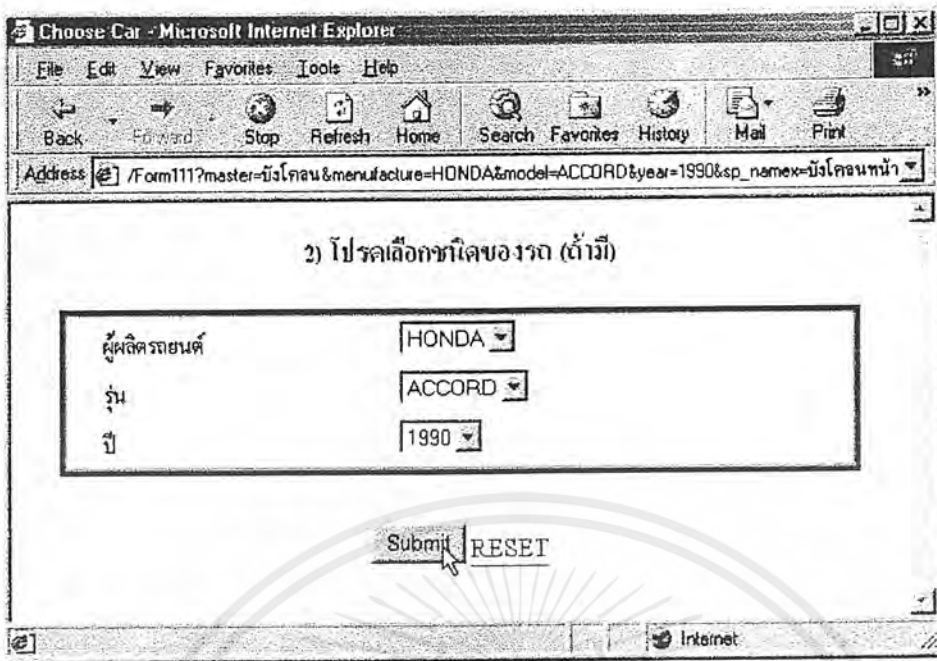
จากรูปที่ 6.4 เมื่อคลิกเลือกชิ้นส่วนจะแสดงหน้าจอถัดไปดังรูปที่ 6.5



รูปที่ 6.5 เว็บเพจแสดงรายการของอะไหล่ชิ้นย่อยๆให้เราเลือก

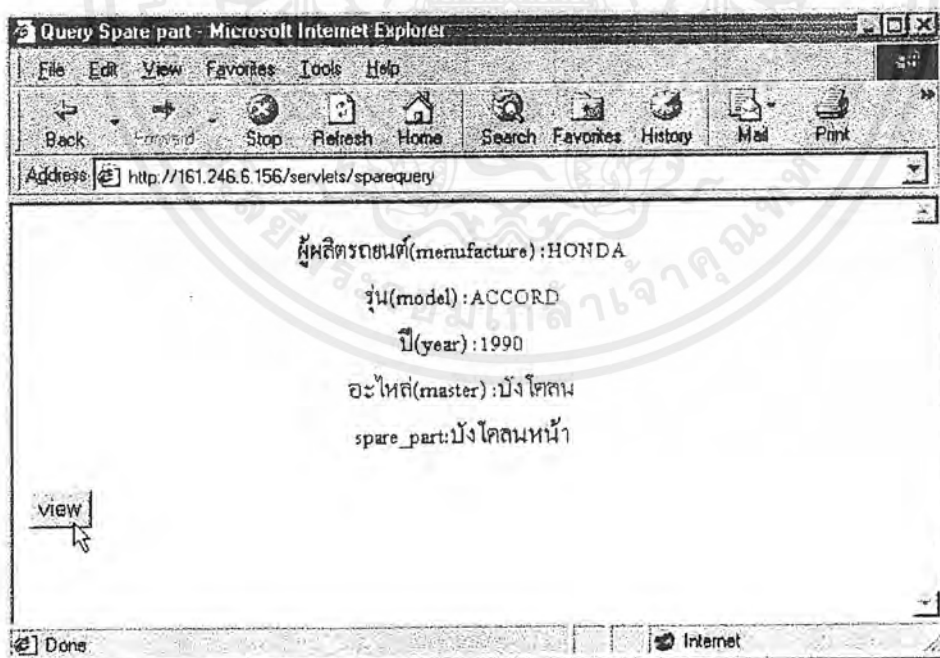
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 6.5 เมื่อคลิกปุ่ม FIND NOW! เว็บเพจจะแสดงหน้าจอดังรูป 6.6



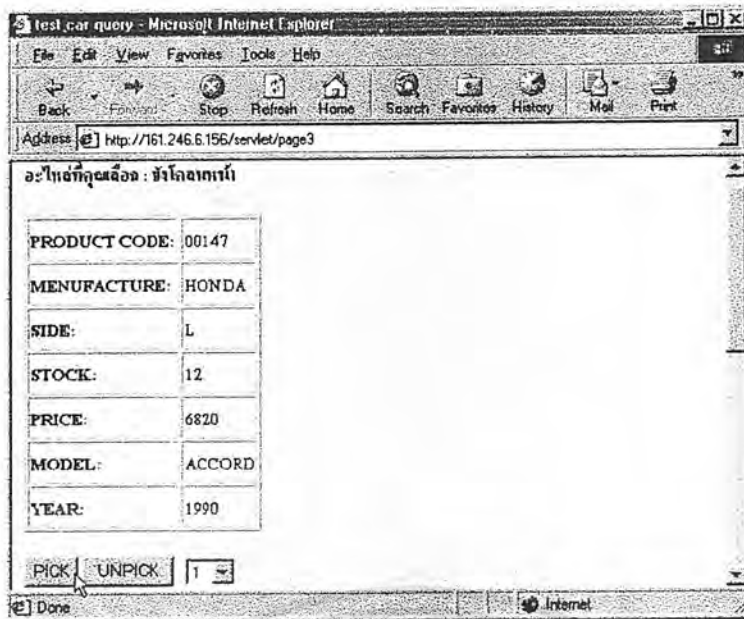
รูปที่ 6.6 เว็บเพจเลือกชนิดและรุ่นของรถที่ใช้กับอะไหล่ที่เราต้องการ

ในกรณีที่ไม่ทราบรุ่นรถแน่ชัด สามารถคลิกปุ่ม Submit ได้ จากนั้นเว็บเพจจะแสดงหน้าจอดังรูปที่ 6.7 ซึ่งจะแสดงชื่อของอะไหล่และรถที่เลือกเอาไว้ ซึ่งเมื่อคลิกปุ่ม view แล้ว เว็บเพจจะแสดงรายการสินค้า ดังรูป 6.8



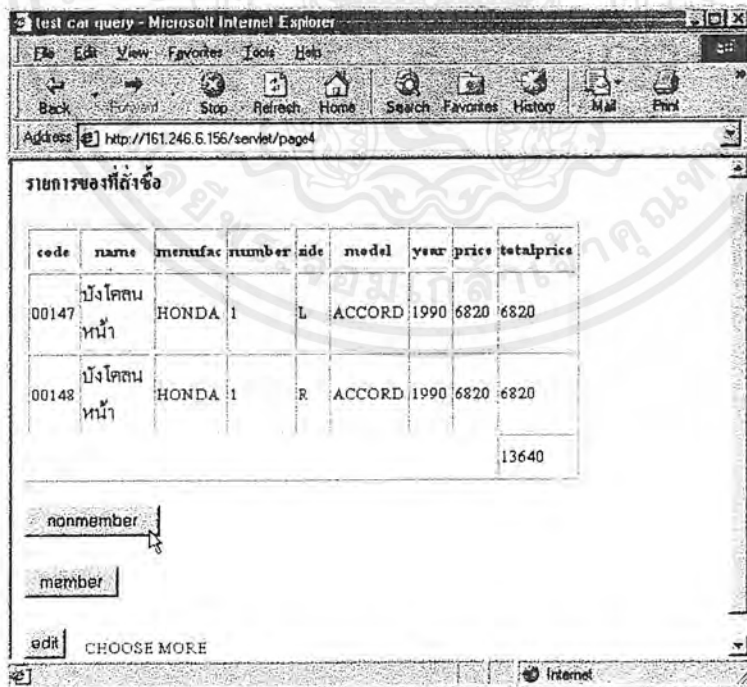
รูปที่ 6.7 เว็บเพจเลือกชนิดและรุ่นของรถที่ใช้กับอะไหล่ที่เราต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.8 เว็บไซต์แสดงรายการสินค้าที่ถูกคัดเลือก

จากรูปที่ 6.8 ในการเลือกซื้อทำได้โดยเลือกจำนวนและ กดปุ่ม PICK หรือถ้ายกเลิกสินค้านั้นก็ กดUNPICK จากหน้าจอเมื่อกดปุ่ม view แล้ว โคมเพจจะแสดงรายการของสินค้าและจำนวนสินค้าที่เรา เลือกซื้อดังรูปที่ 6.9 ซึ่งแสดงรายการสินค้าที่เราตั้งชื่อและขอรวมเมื่อตกลงซื้อก็จะเลือกเข้าหน้าจอของ ลูกค้าที่เป็นสมาชิกและไม่ใช่สมาชิก หรือถ้าอยากแก้ไขรายการที่ซื้อก็กดปุ่ม edit เพื่อย้อนกลับไปแก้ไขรายการนั้น ซึ่งเว็บเพจจะกลับไปเพจก่อนหน้านี้ซึ่งเป็นรูป 6.8



รูปที่ 6.9 เว็บไซต์แสดงรายการสินค้าและจำนวนสินค้าที่เราเลือกซื้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่ตัดสินใจเลือกซื้ออะไหล่ ถ้าหากผู้ซื้อเป็นลูกค้าทั่วไปก็สามารถกดปุ่ม nonmember เพื่อกรอกรายละเอียดอื่นๆ (รูป 6.10) แต่ถ้าหากเป็นสมาชิกสามารถกดปุ่ม member ซึ่งจะต้องล็อกอินเข้าเป็นสมาชิกเพื่อยืนยัน โดยใส่รหัสผ่าน (รูป 6.11)

ถ้าต้องการอะไหล่ประเภทอื่นเพิ่มเติมกด CHOOSE MORE ซึ่งจะกลับไปหน้าแรก ให้เลือกอะไหล่ใหม่ (รูป 6.4)

The screenshot shows a Microsoft Internet Explorer window titled "Home Page - Microsoft Internet Explorer". The address bar contains "161.246.6.156/servlet/page6". The main content area displays a form titled "ไปกรอกรายละเอียด" (Fill in details). The form fields are as follows:

- ชื่อลูกค้า (ชื่อเต็ม): กิจเจริญการช่าง
- ที่อยู่ (เลขที่บ้าน/ชื่อ): 81 ซอยแม่งสาร ถนนบำรุงเมือง กรุงเทพฯ
- รหัสไปรษณีย์: 10100
- เบอร์โทรศัพท์: 71 29204
- e-mail address: pomchai@kichearn.co.th

At the bottom of the form is a "Continue" button. The status bar at the bottom indicates "Local intranet zone".

รูปที่ 6.10 เว็บเพจแสดงหน้าจอนี้สำหรับลูกค้าทั่วไปไปกรอกรายละเอียด

The screenshot shows a Microsoft Internet Explorer window titled "Home Page - Microsoft Internet Explorer". The address bar contains "http://161.246.6.156/servlet/login". The main content area displays a form titled "ไปคลิกล็อกอินระบบ" (Click to log in to the system). The form fields are as follows:

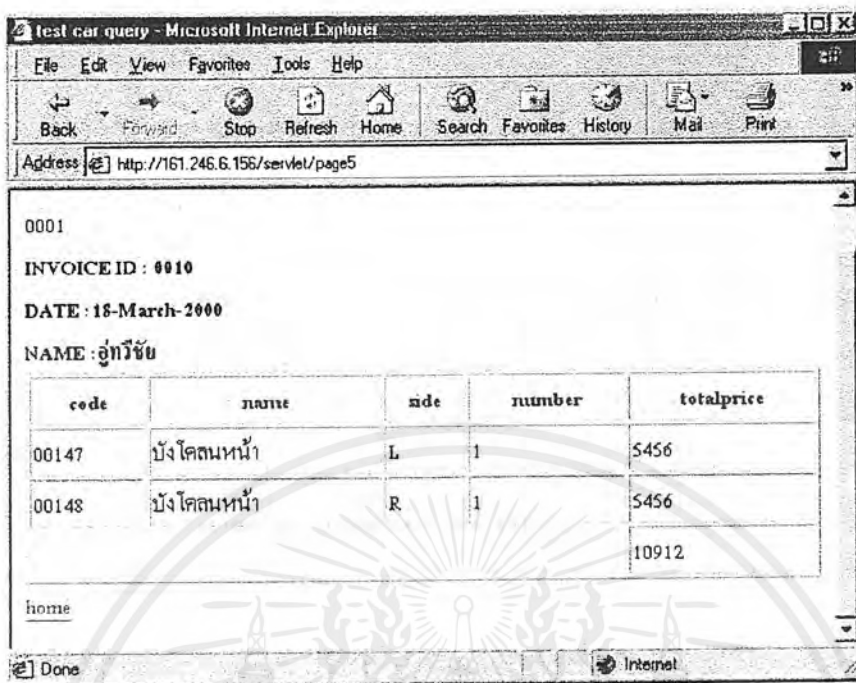
- USERNAME: member1
- PASSWORD: [Redacted]

At the bottom of the form are "Submit" and "Reset" buttons. The status bar at the bottom indicates "Done" and "Internet".

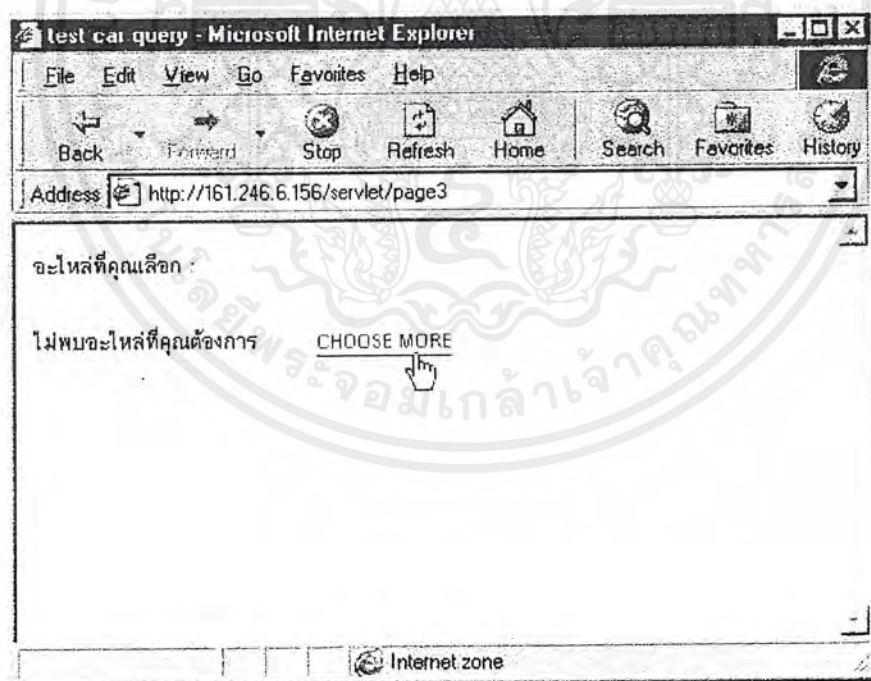
รูปที่ 6.11 เว็บเพจแสดงหน้าจอลูกค้าที่เป็นสมาชิกกรอก username และ password

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 6.10 และ 6.11 เมื่อกด Submit เว็บเพจจะแสดงใบกำกับสินค้าซึ่งมีสินค้าที่เลือกซื้อ พร้อมยอดรวมราคา ดังรูปที่ 6.12 ซึ่งถือว่าการซื้อขายเสร็จสิ้นแล้ว



รูปที่ 6.12 เว็บเพจแสดงใบกำกับสินค้า



รูปที่ 6.13 เว็บเพจแสดงหน้าจอในกรณีที่ไม่มีอะไหล่ลูกค้าต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ปัญหาที่พบและการแก้ไข

ปัญหา หลังจากทีลงออราเคิลแล้ว ไม่สามารถเปิด TNS Listener ได้

วิธีแก้ คู่มือไฟล์ (log file) ของออราเคิล พบว่าการเปิด TNS Listener นั้น มีการอ่านไฟล์ที่ชื่อว่า listener.ora และ tnsnames.ora จึงแก้ไขรายละเอียดไฟล์ทั้งสอง โดยเปลี่ยนคำว่า "<ORACLE_SID>" เป็น "ORCL" ซึ่งเป็นชื่อของฐานข้อมูลที่เราสร้างขึ้นมา

ปัญหา ทดลองเก็บข้อมูลที่เป็นภาษาไทยในตาราง ปรากฏว่า เมื่อเรียกดูข้อมูลที่เก็บในตารางฐานข้อมูล ผ่านโปรแกรม sqlplus ผลคือ สามารถแสดงผลเป็นภาษาไทยได้ แต่พอเขียน โปรแกรมจาวาเซิร์ฟเอร์คลิกค้อกับฐานข้อมูล โดยเรียกข้อมูลจากตารางออกมาแสดงที่เน็คสเค็ป ปรากฏว่า ตัวอักษรภาษาไทยแต่ละตัว กลายเป็นเครื่องหมาย "?" แสดงออกมาที่เพจ

วิธีแก้ จากการศึกษาค้นพบว่า การเก็บข้อมูลโดยใช้ตัวอักษรอื่นๆ ที่ไม่ใช่ภาษาอังกฤษ หรือเป็นภาษาอังกฤษนั้น จะเกี่ยวข้องกับตัวแปรที่ชื่อว่า Character Set และ National Character Set ที่กำหนดให้กับฐานข้อมูล ซึ่งเลือกให้มีค่าเป็น "TH8TISASCII" ทั้งสองตัว เพื่อต้องการให้ฐานข้อมูลสามารถซอร์ท(sort) ตัวอักษร ไทย ได้ แต่ใช้ยังไม่ได้เนื่องจาก ต้องมีการเซตตัวแปรแวดล้อมที่ชื่อว่า NLS_LANG ให้เข้ากับฐานข้อมูลอีก และยังศึกษาไม่พอ ทำให้ต้องสร้างฐานข้อมูลใหม่ ซึ่งกำหนดตัวแปร Character Set และ National Character Set ให้เป็นค่า default (US7ASCII) ซึ่งสามารถเก็บตัวอักษรภาษาไทยได้

ปัญหา หลังจากทีลงลินุกซ์แล้ว บุคลินุกซ์ไม่ขึ้น โดยคอนลงจะมีให้เลือกว่า ลงตัวจัดการบูต (Lilo) ที่ตำแหน่งไหนในฮาร์ดดิสก์ โดยมีให้เลือกคือ "Master Boot Record" และ "First sector of boot partiuon" และต้องการให้ Lilo จัดการดูแลการบูตระบบจึงเลือกที่ Master Boot Record

วิธีแก้ ทำการลงใหม่อีกหลายครั้ง แต่ผลเหมือนเดิมคือ บูตไม่ขึ้น สันนิษฐานว่าตำแหน่ง Master Boot Record ในฮาร์ดดิสก์มีปัญหา ดังนั้น การบูคลินุกซ์ขึ้นมาใช้งานจึงใช้แผ่นบูต

ปัญหา หลังจากลงอาปาเช่แล้ว คิดตั้งอาปาเช่-เจเซิร์ฟ ตามขั้นคอนแล้ว ทดลองเปิด apachectl มีข้อความแจ้งขึ้นมาว่า "httpd not restart (pid is still running)"

วิธีแก้ เนื่องจากขั้นตอนคิดตั้งลินุกซ์นั้น ได้เลือกเพ็คเก็จเว็บเซิร์ฟเวอร์ด้วย ทำให้มีการคิดตั้งอาปาเช่ในระบบอยู่แล้ว และเริ่มทำงาน โดยอัตโนมัติเมื่อเริ่มเปิดเครื่อง จึงต้องลงลินุกซ์ใหม่ โดยไม่เลือกเพ็คเก็จเว็บเซิร์ฟเวอร์ทำให้ไม่ลงอาปาเช่ซ้ำซ้อน

ปัญหา ใช้คำสั่งคอมไพล์ไฟล์ที่มีนามสกุล .java โดยใช้คำสั่ง javac ไม่ได้ ทั้งทีหลังจากทีลง JDK1.2.2 แล้วได้มีการทดลองใช้คำสั่งต่างๆ แล้วปรากฏว่าใช้ได้

วิธีแก้ เกิดจากการเซตตัวแปร CLASSPATH = /u01/app/oracle/product/8.0.5/jdbc/lib หลังจากคิดตั้งออราเคิล เพื่อให้เห็น ไคร์ฟเวอร์ JDBC ที่ติดตั้งมากับออราเคิล ทำให้ไม่สามารถรันได้ ดังนั้นจึงลบตัวแปร CLASSPATH ออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหา ติดตั้งออรากิลบนลินุกซ์ตามขั้นตอนที่ได้จาก <http://jordan.forwayne.com/oracle> ระหว่างติดตั้ง Oracle8 Server มีข้อความแจ้งความผิดพลาด (error message) หลายข้อความ ดังนี้

```
"There are still sgafef<SID>.dbf file(s) in $ORACLE_HOME/dbs/
You must shut down all database instances associated
with this ORACLE_HOME before upgrading the Server"
```

O/S Error

"Error during action "Looking for system network libraries to install TCP/IP Protocol Adapter"

O/S Error

```
"Error during action "Createing rdbms/lib/psoliblist"
Command : make -f ins_rdbms.mk
/u01/app/oracle/project/8.0.5/rdbms/lib/psoliblist
ins_rdbms.mk : 15 : /u01/app/oracle/project/8.0.5/rdbms/lib/
env_rdbms.mk : No such file or directory
ins_rdbms.mk : 45 : *** command commence before first target.Stop
```

วิธีแก้ จากการที่ทดลองลงออรากิลใหม่หลายครั้ง พบว่าขั้นตอนที่ทำถูกต้องทุกขั้นตอน จึงสรุปว่า ซีดีที่ใช้ติดตั้งออรากิลมีปัญหา อาจเกิดจากชื่อไฟล์ในแผ่นซีดีเป็นตัวใหญ่หมด (เกิดความผิดพลาดกับซีดีที่ถือไปมา) และไฟล์บางไฟล์อาจสูญหายไป จึงติดต่อกับบริษัทออรากิลในประเทศไทย เพื่อขอซีดีติดตั้งออรากิลสำหรับระบบปฏิบัติการลินุกซ์ ผลปรากฏว่าติดตั้งได้ โดยไม่มีข้อความแจ้งความผิดพลาดใดๆ

ปัญหา หลังจากสร้าง User ที่ชื่อ PROJECT แล้วและให้สิทธิ์อนุญาตให้สามารถทำการติดต่อกับฐานข้อมูลได้ โดยใช้คำสั่ง

```
CREATE USER PROJECT
IDENTIFIED BY PROJECT; ← รหัสผ่านของ User ที่ชื่อ Project
GRANT CONNECT TO PROJECT;
```

จากนั้น ทดลองสร้างตารางของ User ที่ชื่อ PROJECT ปรากฏว่าไม่สามารถสร้างตารางได้

วิธีแก้ ให้สิทธิ์กับ User ที่ชื่อ PROJECT โดยใช้คำสั่ง

```
GRANT RESOURCE TO PROJECT;
```

ปัญหา โปรแกรมที่เขียนขึ้นไม่สามารถเช็คสถานะของลูกค้าที่เลือกสินค้า คือ ถ้าลูกค้าเปลี่ยนไปที่เพจอื่น รายการของลูกค้าจะหายไปหมด

วิธีแก้ ใช้ session tracking

ปัญหา เขียนโปรแกรมจาวาเซิร์ฟเล็ต โดยให้คิวรีข้อมูลจากฐานข้อมูล เมื่อเรียกพจน์นี้หลายครั้งทางเว็บเบราว์เซอร์ หน้าจอจะแสดงข้อความแจ้งความผิดพลาดว่า 500 Internal Server Error

วิธีแก้ เขียนโปรแกรมเพิ่มให้แสดงข้อความแจ้งความผิดพลาดออกมาที่ Catch Exception พบว่ามีข้อความแสดงข้อผิดพลาด คือ “ORA-01000 : maximum open cursors exceeded”

ซึ่งจำนวนของเคอร์เซอร์ที่คลเอนท์หลายๆ สามารถเปิดได้ในเวลาหนึ่งมีจำกัด (ปกติคือ 50 ครั้ง) ดังนั้นจึงเขียนโปรแกรมให้มีการปิดเคอร์เซอร์ทุกครั้งที่มีการเลิกใช้แล้ว โดยใช้คำสั่ง stmt.close();



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] RedHat Linux ฉบับเพื่อการใช้งานจริง, นิรุช อำนวยศิลป์, บริษัท ชัคเซสมิเคีย จำกัด
- [2] คู่มือ ORACLE8, 1999, พิชัย จันทร์จรัสทอง, แมคกรอ-ฮิล อินเทอร์เน็ตเนชันแนล เอ็นเตอร์ไพรส์ ینگ์
- [3] การเขียนโปรแกรมบนเว็บ, งามนิง อาออินทร์, บริษัทดวงกมลสมัยจำกัด
- [4] The SQL Guide to ORACLE, Rick F. van der Lans, ADDISON-WESLEY
- [5] JAVA Servlets, Karl Moss, McGraw-Hill
- [6] Java Servlets Programming, JasonHunter with William Crawford, OREILLY
- [7] Official Netscape JavaScript 1.2 Programmer's Reference, Peter Kent Kent Mutter, Ventana Communication Group
- [8] Database Design and SQL, NECTEC
- [9] สร้างอินเทอร์เน็ตเซิร์ฟเวอร์ด้วย Linux, กัทธิระ เกียรติเสวี และทีมงานห้องปฏิบัติการเครือข่ายคอมพิวเตอร์ (NTL), ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC), บริษัท ซีเอ็ดยูเคชั่นจำกัด
- [10] Database Programming with JDBC and JAVA, George Reese, OREILLY
- [11] <http://apache.org>
- [12] <http://java.apache.org>
- [13] <http://technet.oracle.com>
- [14] <http://java.sun.com/products/Servlet>
- [15] <http://java.sun.com>
- [16] <http://jordan.fortwayne.com/oracle/rh60-805.html>
- [17] <http://www.wv.com/presentations/IntroJavaScript/fromcheck.htm>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก. การติดตั้งระบบปฏิบัติการลินุกซ์

ความต้องการขั้นพื้นฐาน

- ซีพียูรุ่น 386SX หรือสูงกว่า
- RAM อย่างน้อย 4 MB (ควรเป็น 8 MB ขึ้นไปถ้าต้องการใช้งานแบบกราฟิกด้วย)
- ฮาร์ดดิสก์ขนาด 100 MB (200 MB ขึ้นไปถ้าต้องการใช้งานแบบกราฟิกด้วย)

ขั้นตอนการติดตั้ง

สร้างแผ่น boot ของลินุกซ์ โดยใช้คำสั่ง `rewrite -f \images\Boot.img` ที่ subdirectory ของ DOSUTILS ในแผ่น CD ที่ติดตั้งลินุกซ์

บู๊ตจากแผ่นเพื่อเข้าสู่หน้าจอของการติดตั้ง

- เลือกชนิดของคีย์บอร์ด เป็น us
- เลือกสื่อที่เราใช้ติดตั้ง เป็น local CDROM เนื่องจาก ติดตั้งจาก CD
- เลือกรูปแบบของการติดตั้ง เป็นแบบ Install
- เลือกชนิดของ SCSI Adapter ซึ่งไม่มี
- แบ่ง partition ใน Hard Disk โดยใช้ Disk Druid หรือ fdisk ก็ได้ เราต้องสร้าง partition ขึ้นมา 2 แบบ ทั้ง Linux native (เป็น Root Partition ไว้เก็บข้อมูลและโปรแกรมต่างๆ ของระบบ) และ Linux Swap (เป็นหน่วยความจำสำรอง หรือที่เรียกว่า “Virtual Memory”) โดยขนาดของ Linux native ควรจะมีขนาดใหญ่ ส่วน Swap partition ให้ใช้เนื้อที่เท่ากับขนาดของ RAM
- ทำการ Mount Point โดยใช้เครื่องหมาย / ลงไป เพื่อบอกว่า Partition นี้ คือ Root Partition
- เลือก Root Partition
- เลือก Swap Partition
- ฟออร์แมต Partition
- เลือก Component ที่ต้องการติดตั้ง
- เลือกชนิดของ Mouse เป็น Generic Mouse
- เลือกชนิดของการ์ดจอ
- เลือกชนิดของจอภาพ
- ปรับแต่งระบบ Network โดยเตรียมค่าต่าง ๆ ให้พร้อม ได้แก่ IP Address, Netmask, Default Gateway, Primary Nameserver, Domain Name และ Host Name
- เขตเวลาของระบบ
- กำหนดการเริ่มทำงานของ Service
- ติดตั้ง LILO เลือกติดตั้งลงไปใน MBR (Master Boot Record)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. การติดตั้งอปาเซ-เจเซิร์ฟ

ความต้องการก่อนการติดตั้ง

- ระบบต้องมีแพ็คเกจของ C (ในขั้นตอนนี้ install RedHat เลือก package ของ C Development)
- Java Development Kit จาก <http://www.java.sun.com/>
- Apache1.3.9 – download “apache_1.3.9.tar.gz” จาก <http://www.apache.org/dist/>
- ApacheJServ 1.1b2 – download “ApacheJServ-1.1b2.tar.gz” จาก <http://java.apache.org/>
- Java Servlet Development Kit 2.0 – download “jsdk20-solaris2-sparc[2].tar.Z” จากเว็บไซต์ java.sun.com โดยเข้าไปที่ <http://www.java.sun.com/products/servlet/download.html>

ขั้นตอนการติดตั้ง

- นำไฟล์ที่ download มาเหล่านี้ใส่ไว้ใน /usr/local (JDK , JSDK , Apache, Apache JServ)
- ลง JDK ในที่นี้ ใช้ JDK1.2 โดยอยู่ในรูปของ bunzip2 และ tar

```
# cd /usr/local
# bunzip2 -fv jdk1.2pre-v2.tar.bz2
# tar xvf jdk1.2pre-v2.tar
# ln -s /usr/local/jdk1.2 /usr/local/java
```

แก้ /etc/profile เพื่อเพิ่ม PATH ให้มี /usr/local/java/bin หลังจากนั้น
logout แล้ว login ใหม่

ทดสอบว่า JDK ที่ลงไปใช้ได้หรือเปล่าโดยพิมพ์

```
# java หรือ # javac
```

ถ้าใช้ได้จะแสดงวิธีการใช้ และ option ให้ โดยไม่มีการแจ้ง error อะไร

- ลง JSDK2.0

```
# cd /usr/local
# tar xvfz jsdk20-solaris2-sparc[2].tar.Z
```

- ลง Apache ใหม่ ในที่นี้จะใช้ DSO ซึ่งเป็นวิธีที่สะดวก คือ ไม่ต้อง compile ใหม่ทุกครั้ง
ที่มีการเปลี่ยนแปลง

```
# cd /usr/local
# tar xvfz apache_1.3.9.tar.gz
# cd apache_1.3.9
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#./configure \
--prefix=/usr/local/apache \
--enable-rule=SHARED_CORE \
--enable-module=so
# make install

```

แก้ไข /etc/profile เพื่อเพิ่ม PATH ให้มี /usr/local/apache/bin

logout แล้ว login ใหม่

ทดลอง start Apache Web Server โดยพิมพ์

```
# apachectl start
```

```
# lynx http://localhost
```

ถ้าขึ้นคำว่า “It Worked! The Apache Web Server is installed on this Web Site!”

แสดงว่าลงสำเร็จได้แล้ว

จากนั้น stop apache ก่อนที่จะทำการ configuring Apache JServ

```
# apachectl stop
```

- ลง ApacheJServ 1.1b2

```

# cd /usr/local
# tar xvzf ApacheJServ-1.1b2.tar.gz
# cd ApacheJServ-1.1b2
# ./configure \
--prefix=/usr/local/jserv \
--with-apxs=/usr/local/apache/bin/apxs \
--with-jdk-home=/usr/local/java \
--with-JSDK=/usr/local/JSDK2.0/lib/jsdk.jar \
--disable-dubugging
# make
# make install
# cd /usr/local/JavaJServ-1.1b2/src/java
# cp *.jar /usr/local/jserv

```

- แก้ไฟล์ httpd.conf ใน /usr/local/apache/conf โดยเพิ่มบรรทัด

```
Include /usr/local/apache/conf/jserv/jserv.conf
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดสอบ ApacheJServ

```
# apachectl start
```

```
# lynx http://localhost/servlets/Hello
```

ถ้ามีข้อความ “Example Apache Jserv Servlet

Congratulations, ApacheJServ 1.1b2 is working”

แสดงว่าติดตั้งสำเร็จแล้วเว็บไซต์เวิร์กได้ถูกต้อง และสามารถทำงานได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก. การติดตั้งออราเคิล

เตรียม Environment ก่อน install (root user)

- Mount Redhat6 CD แล้วติดตั้ง Packages ที่จำเป็นต้องมีก่อนทำการติดตั้งออราเคิล ซึ่ง Package เหล่านี้มีอยู่ในแผ่น RedHat6 โดยใช้คำสั่งต่อไปนี้

```
# mkdir /cdrom
# chmod 777 /cdrom
# mount -t iso9660 /dev/cdrom /cdrom
# cd /cdrom/RedHat/RPMS
# rpm -ivh compat-binutils-5.2-2.9.1.0.23.1.i386.rpm
# rpm -ivh compat-glibc-5.2-2.0.7.1.i386.rpm
# rpm -ivh compat-egcs-5.2-1.0.3a.1.i386.rpm
# rpm -ivh compat-egcs-c++-5.2-1.0.3a.1.i386.rpm
# rpm -ivh compat-libs-5.2-1.i386.rpm
# cd /
# umount /cdrom
```

- เปลี่ยน shell ของ root ให้เป็น /bin/sh ในไฟล์ /etc/passwd
- logout แล้ว login ใหม่
- สร้าง group DBA ขึ้นมา
- สร้าง oracle user โดยให้อยู่ในกลุ่ม DBA ซึ่ง oracle user จะเป็นผู้ทำการติดตั้งออราเคิล
- กำหนด password ให้กับ oracle user
- เปลี่ยน shell ของ oracle ให้เป็น /bin/sh ในไฟล์ /etc/passwd
- สร้าง ไดรเรกทอรี(Mount Points) ให้กับออราเคิล แล้วเปลี่ยน owner ให้เป็นของ oracle user

```
# mkdir /u01
# chown -R oracle.dba /u01
```

- mount Oracle Server CD และรันไฟล์สคริปต์ชื่อ oratab.sh เพื่อสร้างไฟล์ oratab ในไดเรกทอรี /etc โดยต้องกำหนดตัวแปร ORACLE_OWNER ให้เป็น oracle ก่อน

```
# mount -t iso9660 /dev/cdrom /cdrom
# cd /cdrom/orainst
# ORACLE_OWNER=oracle; export ORACLE_OWNER
# sh oratab.sh
```

- logout

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตรียม Environment ก่อนลง (oracle user)

- ดาวน์โหลด glibcpatch.tgz (เฉพาะ Linux version 6)
 - \$ mkdir ~/orapatch
 - \$ cd ~/orapatch
 - \$ ftp ftp.oracle.com
 - \$ cd /pub/www/otn/linux
 - \$ get glibcpatch.tgz
 - \$ quit
- ตรวจสอบค่า umask = 022 หรือเปล่า ถ้าไม่ใช่ ต้องเซตให้เป็น 022 ก่อน
- set ค่าต่างๆ ใน .profile ที่ home directory
 - ORACLE_HOME=/u01/app/oracle/product/8.0.5; export ORACLE_HOME
 - LD_LIBRARY_PATH=/u01/app/oracle/product/8.0.5/lib; export LD_LIBRARY_PATH
 - ORACLE_BASE=/u01/app/oracle; export ORACLE_BASE
 - ORACLE_SID=ORCL; export ORACLE_SID
 - ORACLE_TERM=386; export ORACLE_TERM
 - PATH=\$PATH: /u01/app/oracle/product/8.0.5/bin; export PATH
 - TMPDIR=/var/tmp; export TMPDIR
 - NLS_LANG=THAI_THAILAND.TH8YISASCII; export NLS_LANG
- logout ออก แล้ว login ใหม่

ทำการ Install Oracle (oracle user)

- เริ่มต้นโดยการเรียก โปรแกรมติดตั้ง
 - \$ cd /cdrom/orainst
 - \$./orainst /c
- ในการลง install oracle ควรเลือกแบบ Custom
- มี choice ใน installation activity อยู่ 3 ข้อ ให้เลือก
 - Install, Upgrade, or De-install Software
 - Create/Upgrade Database Objects
 - Perform Administrative Tasks
- เลือก Install Software แต่ยังไม่ต้อง Create Database Object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในหน้าจอ Software Asset Manager จะเป็นการเลือก Software ที่เราต้องการ Install ซึ่งมี Product ให้เลือก 11 Products ดังต่อไปนี้

1. Client Software
2. Net8 8.0.5
3. Oracle Intelligent Agent
4. Oracle JDBC Driver
5. Oracle Unix Installer
6. Oracle8 Server, Release8.0.5
7. Oracle 8 Standard (RDBMS) 8.0.5
8. PL/SQL 8.0.5
9. SQL* Plus 8.0.5
10. Net8 Protocol Adapter 8.0.5
11. Precompiler 8.0.5

- ในการเลือกลง Oracle JDBC Driver ต้องมีตัวแปลภาษาจาวาก่อน ซึ่งก็คือ Java Development Kit หรือ JDK (ในระบบใช้ JDK1.2 version 2) แล้วกำหนดค่า CLASSPATH ให้เห็น ไคลเรททอรี /u01/app/oracle/product/8.0.5/jdbc/lib/classes111.zip
- การเลือก JDBC Components โดยเลือก JDK1.1 JDBC OCI Driver และ JDK 1.1 JDBC Thin Driver
- ไม่ควรลง Oracle Intelligent Agent เพราะจะทำให้มีปัญหาตามมาภายหลัง
- ยังไม่ต้องลง Oracle8 Server, Documentation Release เนื่องจากจะเจอปัญหาในการหา ไคลเรททอรี ไม่พบ ต้อง link directory ก่อน ทำได้ดังนี้

```
$ mkdir /u01/app/oracle/doc
```

```
$ ln -s /u01/app/oracle/doc /u01/app/oracle/product/8.0.5
```

- ถ้าได้รับข้อความ “The requested action has been performed for selected products” หมายความว่า Product ที่เราเลือกไว้ถูก install เสร็จแล้ว
- ออกจาก Installer
- ทำการ Patch Binaries โดยการ run glibcpatch.tgz (ทำโดย oracle user)

```
$ cd ~/orapatch
```

```
$ tar xvfz glibcpatch.tgz
```

```
$ sh glibcpatch.tgz
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้าง Database Objects

โดยการ run installer แล้วเลือก Create/Upgrade Database Object

- ในการ Create Database Object ต้องกำหนด Character Set และ National Character Set ให้เป็น TH8TISASCII เพื่อให้สามารถ sort ข้อมูลตามอักษรภาษาไทยได้
- กำหนด password ของ SYSTEM, SYS, TNS listener
- ไม่ต้อง Configure MTS Listener
- ถ้าได้รับข้อความ “The requested action has been performed for selected products” หมายความว่า สร้างเสร็จ Database Object เสร็จแล้ว

ขั้นตอนหลังการ Install

- root ทำการก๊อปปี้ไฟล์ .profile ของ oracle user จากนั้น logout แล้ว login เป็น root เข้ามาใหม่
- รันสคริปต์ไฟล์ชื่อ root.sh ในไดเรกทอรี \$ORACLE_HOME/orainst
- ทำการ Stop Server


```
$ svrmgrl
svrmgrl> connect internal
svrmgrl> shutdown
svrmgrl> quit
```
- เปลี่ยนแปลงและปรับปรุง initORCL.ora ซึ่งเป็น ไฟล์คอนฟิกูเรชัน ให้เหมาะสมกับสภาพแวดล้อม
- TNS Listener เป็นตัวที่คอยดูว่ามีใครจากภายนอก Connect เข้ามายัง Database Instance จำเป็นต้องมีการเปลี่ยน permission เป็นดังนี้

```
$ su root
# chown oracle.dba $ORACLE_HOME/bin/tnslsnr
# chmod 750 $ORACLE_HOME/bin/tnslsnr
# chown oracle.dba $ORACLE_HOME/network/log
# chmod 775 $ORACLE_HOME/network/log
# exit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากติดตั้งออร์ราเคิลเรียบร้อยแล้ว เราสามารถศึกษาการทำงานของระบบฐานข้อมูลได้จาก user “scott” (มี password เป็น tiger) ที่ติดตั้งมาให้พร้อมกับออร์ราเคิล ก่อนอื่นต้องสร้าง user profile โดยรันสคริปต์ชื่อ “pupbld.sql” ก่อน โดยใช้โปรแกรม sqlplus

```
$ cd $ORACLE_HOME/sqlplus/admin
$ sqlplus
sqlplus> connect system
sqlplus> @pupbld.sql
sqlplus> quit
```

ถ้าต้องการสร้างตารางให้กับ “scott” ต้องทำดังนี้

```
$ cd $ORACLE_HOME/rdbms/admin
$ sqlplus (Connect as SYS)
sqlplus> @utlsamp1.sql
sqlplus> quit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

Source Code

ไฟล์ index.html

```

<html>
<head>
<title>Muai & Sai SparePart Homepage</title>
</head>
<body background="/image/41.gif">
<center><a href="http://161.246.6.156/servlets/Sap"></a></center>
</body>
</html>

```

ไฟล์ connect.java

```

public class connect
{
public final static String hostname="161.246.6.156";
public final static String portnumber="1521";
public final static String databaseSID="ORCL";
public final static String username="project";
public final static String password="project";
}

```

ไฟล์ open_connection.java

```

import java.io.*;
import java.sql.*;

public class open_connection
{
Connection con = null;
Statement stmt = null;

public void connectdb()
{
try
{
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+":"+
    connect.portnumber+":"+connect.databaseSID
    ,connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
    System.out.println("error while initializing database connection"+ex.toString());
}
}
}
}

```

ไฟล์ ConnectionPool.java

```

import java.sql.*;
import java.util.*;

public class ConnectionPool {
    private Hashtable connections;
    private int increment;
    private String dbURL, user, password;

    public ConnectionPool (String dbURL,
        String user,
        String password,
        String driverClassName,
        int initialConnections,
        int increment)
        throws SQLException, ClassNotFoundException {

        // Load the specified driver class
        Class.forName(driverClassName);
        this.dbURL = dbURL;
        this.user = user;
        this.password = password;
        this.increment = increment;

        connections = new Hashtable();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Put our pool of Connections in the Hashtable
// The FALSE value indicates they're unused
for(int i = 0; i < initialConnections; i++) {
    connections.put(DriverManager.getConnection(dbURL, user, password), Boolean.FALSE);
}
}

public Connection getConnection() throws SQLException {
    Connection con = null;

    Enumeration cons = connections.keys();

    synchronized (connections) {
        while(cons.hasMoreElements()) {
            con = (Connection)cons.nextElement();

            Boolean b = (Boolean)connections.get(con);
            if(b == Boolean.FALSE) {
                // So we found an unused connection.
                try {
                    con.setAutoCommit(true);
                }
                catch(SQLException e) {
                    // Problem with the connection, replace it.
                    con = DriverManager.getConnection(dbURL, user, password);
                }
                // Update the Hashtable to show this one's taken
                connections.put(con, Boolean.TRUE);
                // Return the connection
                return con;
            }
        }
    }

    // If we get here, there were no free connections.
    // We've got to make more.
    for(int i = 0; i < increment; i++) {
        connections.put(DriverManager.getConnection(dbURL, user, password), Boolean.FALSE);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Recurse to get one of the new connections.
return getConnection();
}

public void returnConnection(Connection returned) {
    Connection con;
    Enumeration cons = connections.keys();
    while (cons.hasMoreElements()) {
        con = (Connection)cons.nextElement();
        if (con == returned) {
            connections.put(con, Boolean.FALSE);
            break;
        }
    }
}
}
}

```

ไฟล์ Sap.java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Sap extends HttpServlet
{
    public void service(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {

        res.setContentType("text/html");
        PrintWriter out = res.getWriter();
        String values[];
        String lastpage="";
        values=req.getParameterValues("lastpage");
        if(values!=null)
        {
            lastpage=values[0];
        }
        if(lastpage.compareTo("yes")!=0)
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lastpage="no";
}

    out.println("<HTML><HEAD><TITLE>Choose Spare part</TITLE></HEAD>");
out.println("<BODY bgcolor=#000102>");

out.println("<center><img src='http://"+connect.hostname+"/image/bar1.gif'></center>");
out.println("<br>");

    out.println("<CENTER><FORM NAME=spare METHOD='GET'
ACTION='http://"+connect.hostname+"/servlets/Form'><FONT face=Arial size=2 color=#AA0000>");

    out.println("<font color=#c1c1ff size=2><strong>1) โปรดเลือกประเภทอะไหล่</strong></font>");
    out.println("<INPUT name=master value='ประเภทอะไหล่!'></FONT></FORM>");

    out.println("<p><img src='http://161.246.6.156/image/sap.jpg' border='0' width='493' height='227'
alt='sap.jpg (27461 bytes)' usemap='#sap'></p>");
    out.println("<MAP name='sap'>");

    out.println("<area shape='polygon"
coords='199,81,194,85,203,107,214,116,217,113,215,101,207,87,199,79,197,80'");
    out.println("<href='http://"+connect.hostname+"/servlets/page2/?master=พวงมาลัย&lastpage="+lastpage+"\"");
    out.println("<onmouseout='spare.master.value=spare.master.defaultValue;return true'");
    out.println("<onmouseover='window.status='พวงมาลัย';spare.master.value='พวงมาลัย';return true' >");

    out.println("<area shape='polygon"
coords='101,104,117,105,117,133,114,135,113,154,117,153,117,170,100,169,101,151,109,150,107,135,97,135,97,103,
100,105'");
    out.println("<href='http://"+connect.hostname+"/servlets/page2/?master=กันสะเทือน&lastpage="+lastpage+"\"");
    out.println("<onmouseout='spare.master.value=spare.master.defaultValue;return true'");
    out.println("<onmouseover='window.status='กันสะเทือน';spare.master.value='กันสะเทือน';return true' >");

    out.println("<area shape='polygon"
coords='380,105,390,105,387,126,393,125,391,136,384,136,382,155,388,155,387,167,370,167,370,153,376,153,380,1
33,371,133,372,120,381,122,380,104'");
    out.println("<href='http://"+connect.hostname+"/servlets/page2/?master=กันสะเทือน&lastpage="+lastpage+"\"");
    out.println("<onmouseout='spare.master.value=spare.master.defaultValue;return true'");
    out.println("<onmouseover='window.status='กันสะเทือน';spare.master.value='กันสะเทือน';return true' >");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<area shape='rect' coords='404,116,427,138'");
out.println("href='http://"+connect.hostname+"/servlets/page2?master=ถึงน้ำมัน&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=ถึงน้ำมัน';spare.master.value=ถึงน้ำมัน;return true' >");

out.println("<area shape='polygon"
coords='424,46,433,54,440,71,441,80,439,91,421,100,407,104,446,100,462,88,465,79,456,53,446,44,423,44,427,47'"
);
out.println("href='http://"+connect.hostname+"/servlets/page2?master=สปีชเฮอร์&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=สปีชเฮอร์;spare.master.value=สปีชเฮอร์;return true' >");

out.println("<area shape='polygon"
coords='133,163,138,176,137,190,136,192,149,185,141,158,139,155,132,162,134,166'"
);
out.println("href='http://"+connect.hostname+"/servlets/page2?master=ข้างกั้น โคลน&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=ข้างกั้น โคลน;spare.master.value=ข้างกั้น โคลน;return true' >");

out.println("<area shape='polygon"
coords='400,149,410,155,416,164,418,171,419,183,409,190,411,170,396,149,400,149'"
);
out.println("href='http://"+connect.hostname+"/servlets/page2?master=ข้างกั้น โคลน&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=ข้างกั้น โคลน;spare.master.value=ข้างกั้น โคลน;return true' >");

out.println("<area shape='polygon' coords='95,74,136,75,131,108,79,104,98,72'");
out.println("href='http://"+connect.hostname+"/servlets/page2?master=เครื่อง&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=เครื่อง;spare.master.value=เครื่อง;return true' >");

out.println("<area shape='polygon"
coords='160,113,148,107,148,75,155,59,170,50,156,69,153,78,154,102,160,113'"
);
out.println("href='http://"+connect.hostname+"/servlets/page2?master=ก้านปีค้ำน้ำฝน&lastpage="+lastpage+"')");
out.println("onmouseout='spare.master.value=spare.master.defaultValue;return true'");
out.println("onmouseover='window.status=ก้านปีค้ำน้ำฝน;spare.master.value=ก้านปีค้ำน้ำฝน;return true' >");

out.println("<area shape='polygon"
coords='13,134,21,142,32,145,40,140,42,134,48,135,49,141,74,141,76,133,84,154,77,163,75,176,76,188,67,189,24,17
0,19,152,17,149,12,133,20,139'"
);
out.println("href='http://"+connect.hostname+"/servlets/page2?master=กั้นชน&lastpage="+lastpage+"')");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='กินข้าว';spare.master.value='กินข้าว';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"461,90,454,106,463,128,460,136,442,149,442,169,439,178,458,170,466,161,472,142,470,129,468,111,460,94,
460,92,454,105\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=กินข้าว&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='กินข้าว';spare.master.value='กินข้าว';return true\" >");

out.println("<area shape=\"circle\" coords=\"31,135,14\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ไฟ&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='ไฟ';spare.master.value='ไฟ';return true\" >");

out.println("<area shape=\"polygon\" coords=\"442,146,441,126,458,116,463,130,462,137,443,148,442,139\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ไฟ&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='ไฟ';spare.master.value='ไฟ';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"114,153,118,153,131,163,135,157,133,112,129,106,116,106,112,156,119,153\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ไอรี&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='ไอรี';spare.master.value='ไอรี';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"134,127,155,121,157,118,152,130,150,143,154,157,158,176,162,185,149,185,132,155,135,125,145,125\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=บั้ง โคลน&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='บั้ง โคลน';spare.master.value='บั้ง โคลน';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"77,134,85,117,99,117,99,133,110,136,108,150,100,154,85,153,74,135\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=หม้อก๋วยเตี๋ยว&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='หม้อก๋วยเตี๋ยว';spare.master.value='หม้อก๋วยเตี๋ยว';return true\" >");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<area shape=\"polygon\"
coords=\"53,139,51,106,71,88,89,88,81,105,100,105,95,115,77,142,49,141,52,104\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=หม้อน้ำ&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='หม้อน้ำ';spare.master.value='หม้อน้ำ';return true\" >");

out.println("<area shape=\"polygon\" coords=\"14,136,16,117,32,99,28,119,14,137\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=หน้ากระຈัง&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='หน้ากระຈัง';spare.master.value='หน้ากระຈัง';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"31,97,48,78,81,61,141,51,171,49,156,59,149,75,151,107,161,114,155,119,141,130,42,140,23,117,36,92\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=กระโปรง&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='กระโปรง';spare.master.value='กระโปรง';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"391,45,424,46,431,48,444,76,437,93,423,99,416,105,425,74,414,52,389,46,405,45\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=กระโปรง&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='กระโปรง';spare.master.value='กระโปรง';return true\" >");

out.println("<area shape=\"polygon\"
coords=\"227,27,223,45,223,58,226,72,251,64,289,63,322,66,342,72,341,29,308,22,281,20,249,21,227,27,224,42\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=หัตถ์กา.&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='หัตถ์กา!';spare.master.value='หัตถ์กา!';return true\" >");

out.println("<area shape=\"circle\" coords=\"109,176,32\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ถั๊&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='ถั๊';spare.master.value='ถั๊';return true\" >");

out.println("<area shape=\"circle\" coords=\"380,177,34\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ถั๊&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status='ถั๊';spare.master.value='ถั๊';return true\" >");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<area shape=\"polygon\"
coords=\"170,48,198,38,238,24,227,39,223,42,224,70,181,95,162,113,151,98,152,74,170,48,179,45\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=กิจกรรมปัจฉิม&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status=กิจกรรมปัจฉิม;spare.master.value=กิจกรรมปัจฉิม;return true\" >");

out.println("<area shape=\"polygon\"
coords=\"341,28,379,42,404,48,419,57,427,75,420,94,354,77,341,74,341,37,341,26,343,30\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=กิจกรรมปัจฉิม&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status=กิจกรรมปัจฉิม;spare.master.value=กิจกรรมปัจฉิม;return true\" >");

out.println("<area shape=\"polygon\"
coords=\"162,185,151,155,150,126,163,106,235,61,288,58,331,68,362,104,360,108,308,111,288,113,288,180,282,185,
188,185,157,185,149,133,162,106\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ประวัติ&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status=ประวัติ;spare.master.value=ประวัติ;return true\" >");

out.println("<area shape=\"polygon\"
coords=\"283,184,302,185,348,185,421,183,442,177,443,125,461,117,461,88,322,67,353,103,282,109,284,181,285,186
\"");
out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=บึง ไคลิน&lastpage="+lastpage+"\"");
out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
out.println("onmouseover=\"window.status=บึง ไคลิน;spare.master.value=บึง ไคลิน;return true\" >");

// out.println("<area shape=\"default\"");
// out.println("href=\"http://"+connect.hostname+"/servlets/page2?master=ทั้งหมด!&lastpage="+lastpage+"\"");
// out.println("onmouseout=\"spare.master.value=spare.master.defaultValue;return true\"");
// out.println("onmouseover=\"window.status=ทั้งหมด!;spare.master.value=ทั้งหมด!;return true\" >");

out.println("</MAP></CENTER></BODY></HTML>");

out.close();
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์ page2.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.text.*;
import java.lang.*;

public class page2 extends HttpServlet
{
    String values[];
    Vector v;
    Vector v1;
    Connection con = null;
    Statement stmt = null;
    ResultSet rs = null;

    public void service(HttpServletRequest req,HttpServletResponse resp)
    throws ServletException,java.io.IOException
    {
        connectdb();
        getspec(req,resp);
    }

    public void connectdb()
    {
        try
        {
            DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
            // Get a Connection to the database
            con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+":"+
                connect.portnumber+"."+connect.databaseSID
                ,connect.username,connect.password);

            stmt = con.createStatement();
        }
        catch(SQLException ex)
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println("error while initializing database connection"+ex.toString());
}
}

public void getspec(HttpServletRequest req, HttpServletResponse resp)
{
try
{
resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

v=new Vector();
v1=new Vector();

resp.setHeader("Content-Language","en");
Locale locale=new Locale("en","");
DateFormat fmt=DateFormat.getDateInstance(DateFormat.LONG,DateFormat.LONG,locale);
fmt.setTimeZone(TimeZone.getDefault());
// out.println(fmt.format(new java.util.Date()));

out.println("<html>");
out.println("<head>");
out.println("<title>Choose Spare Part</title>");
out.println("</head>");
out.println("<body background='/image/41.gif'>");
out.println("<center><img src='http://"+connect.hostname+"/image/bar11.gif'></center>");
out.println("<br>");

out.println("<center><font color='#000080'><h3>2) เลือกชิ้นส่วนอะไหล่ที่ต้องการ</h3></font></center>");

// out.println("this part you choose");
String master="";
values=req.getParameterValues("master");
if(values!=null)
{
master=values[0];
master=master.replace(' ','');
master=master.trim();
//out.println(master+"kkkkkkkkkk");
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
// out.println(master);
// out.println("<br>");

String lastpage="";
values=req.getParameterValues("lastpage");
if(values!=null)
{
    lastpage=values[0];
}

String query=null;
int count=0;
query="select distinct sp_name from spare_part where sp_master='"+master+"' order by sp_name";
rs = stmt.executeQuery(query);
int temp=0;
while(rs.next())
{
    v.addElement(rs.getString(1));
// out.println(rs.getString(1));
    temp++;
// out.println(temp);
}
// out.println(temp);
int i,j;
j=v.size();
out.println("<form method=POST action='"+http://"+connect.hostname+"/servlet/Form+'>");
out.println("<select name=sp_name size=1>");
for(i=0;i<j;i++)
{
    out.println("<option value='"+(String)v.elementAt(i)+"'>"+(String)v.elementAt(i)+"");
}
clearElement(v);//remove all in vector

out.println("</select>");
out.println("<br>");
out.println("<br>");
out.println("<br>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<br>");
out.println("<br>");
out.println("<br>");
out.println("<input type=hidden name=master value='"+master+"'>");
out.println("<input type=hidden name=lastpage value='"+lastpage+"'>");
out.println("<input type=submit name=find value='"+ "เลือกวิธีนรณ " > "+" align=middle>");

out.println("</form>");
out.println("</body>");
out.println("</html>");
out.flush();
con.close();
}
catch(Exception ex)
{
ex.printStackTrace();
}
}

public void clearelement(Vector vec)
{
String dummy;
int i,j;

j=vec.size();
for(i=0;i<j;i++)
{
dummy=(String)vec.elementAt(0);
vec.removeElement(dummy);
}
}
}
}

```

ไฟล์ Form.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public class Form extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+":"+connect.portnumber+":"+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
        catch(Exception e) {
            throw new UnavailableException(this, "Couldn't create connection pool");
        }
    }

    public void service(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {
        String values[];

        String sp_namex="";
        values=req.getParameterValues("sp_namex");
        if(values!=null)
        {
            sp_namex=values[0];
            sp_namex=sp_namex+".";
        }

        String master = "";
        values = req.getParameterValues("master");
        if (values!=null) { master = values[0]; }

        String manufacture = "";
        //values = req.getParameterValues("manufacture");
        //if (values!=null) { manufacture = values[0]; }

        String model = "";
        //values = req.getParameterValues("model");
        //if (values!=null) { model = values[0]; }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String year = "";
//values = req.getParameterValues("year");
//if (values!=null) { year = values[0]; }

String lastpage="";
values=req.getParameterValues("lastpage");
if(values!=null)
{
    lastpage=values[0];
}

res.setContentType("text/html");
PrintWriter out = res.getWriter();

//out.println(sp_namex);

    out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
    out.println("<body background='\"/image/41.gif\">");
    out.println("<center><img
src='\"http://"+connect.hostname+"/image/bar2.gif\"></center>");
    out.println("<br>");

    out.println("<SCRIPT language=JavaScript>");

        out.println("function doReload1(entry) {");
        out.println("var current1,current2,current3;");
        out.println("var manufacture,model,year,master,url;");
        out.println("current1 = entry.manufacture.selectedIndex;");
        out.println("manufacture = entry.manufacture.options[current1].value;");
        out.println("current2 = entry.model.selectedIndex;");
        out.println("model = entry.model.options[current2].value;");
        out.println("current3 = entry.year.selectedIndex;");
        out.println("year = entry.year.options[current3].value;");

        out.println("master = entry.master.value;");
        out.println("sp_namex=entry.sp_namex.value;");
        out.println("lastpage=entry.lastpage.value;");

        out.println("url =
'\"http://"+connect.hostname+"/servlets/Form1?master="+master+"\"&manufacture="+manufacture+"\"&model="+mod
el+"\"&year="+year+"\"&sp_namex="+sp_namex+"\"&lastpage="+lastpage;");
        out.println("if (manufacture!="\"");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("window.location.href = url;");
out.println("}");

out.println("function doReload2(entry) {");
out.println("var current1,current2,current3;");
out.println("var manufacture,model,year,master,url;");
out.println("current1 = entry.manufacture.selectedIndex;");
out.println("manufacture = entry.manufacture.options[current1].value;");
out.println("current2 = entry.model.selectedIndex;");
out.println("model = entry.model.options[current2].value;");
out.println("current3 = entry.year.selectedIndex;");
out.println("year = entry.year.options[current3].value;");

out.println("master = entry.master.value;");
out.println("sp_namex=entry.sp_namex.value;");
out.println("lastpage=entry.lastpage.value;");
out.println("url =
"http://"+connect.hostname+"/servlets/Form2?master="+master+"&manufacture="+manufacture+"&model="+model
+"&year="+year+"&sp_namex="+sp_namex+"&lastpage="+lastpage;");
out.println("if (model!="");
out.println("window.location.href = url;");
out.println("}");

out.println("function doReload3(entry) {");
out.println("var current1,current2,current3;");
out.println("var manufacture,model,year,master,url;");
out.println("current1 = entry.manufacture.selectedIndex;");
out.println("manufacture = entry.manufacture.options[current1].value;");
out.println("current2 = entry.model.selectedIndex;");
out.println("model = entry.model.options[current2].value;");
out.println("current3 = entry.year.selectedIndex;");
out.println("year = entry.year.options[current3].value;");

out.println("master = entry.master.value;");
out.println("sp_namex=entry.sp_namex.value;");
out.println("lastpage=entry.lastpage.value;");
out.println("url =
"http://"+connect.hostname+"/servlets/Form3?master="+master+"&manufacture="+manufacture+"&model="+model
+"&year="+year+"&sp_namex="+sp_namex+"&lastpage="+lastpage;");
out.println("if (year!="");
out.println("window.location.href = url;");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        out.println("<OPTION VALUE=\"" + rs.getObject("year").toString() + "\">" + rs.getObject("year").toString() + "
</OPTION>");
    }
    out.println("</SELECT>");

    pstmt = null;
    rs = null;

//-----END-----

    out.println("</FONT></TD></TR>");
    out.println("</TBODY></TABLE></CENTER></TD></TR></TBODY></TABLE></CENTER>");

    out.println("<BR><div align=\"center\"><center><p>");
    out.println("<INPUT TYPE=\"submit\" value=\"ค้นหาชื่อหนังสือ\" name=\"B1\">");
    out.println("<A href=\"http://"+connect.hostname+"/servlets/Sap\"><FONT color=\"#0000FF\" size=4>ชื่อหนังสือ</FONT></A>");
    out.println("</p></center></div>");
    out.println("</FORM>");
    con.commit();
    pstmt.close();
    rs.close();
}

catch(Exception e) {
    try {
        con.rollback();
    }
    catch (Exception ignored) {}
    //out.println("can not create connection in FORM " + e.getMessage());
}

finally {
    if (con != null) pool.returnConnection(con);
}

    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์ Form1.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form1 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+":"+connect.portnumber+":"+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
        catch(Exception e) {
            throw new UnavailableException(this, "Couldn't create connection pool");
        }
    }

    public void service(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {

        String values[];

        String master = "";
        values = req.getParameterValues("master");
        if (values!=null) { master = values[0]; }

        String manufacture = "";
        values = req.getParameterValues("manufacture");
        if (values!=null) { manufacture = values[0]; }

        String model = "";
        values = req.getParameterValues("model");
        if (values!=null) { model = values[0]; }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String year = "";
values = req.getParameterValues("year");
if (values!=null) { year = values[0]; }

String sp_namex = "";
values = req.getParameterValues("sp_namex");
if (values!=null) { sp_namex = values[0]; }

String lastpage="";
values=req.getParameterValues("lastpage");
if(values!=null)
{
    lastpage=values[0];
}

res.setContentType("text/html");
PrintWriter out = res.getWriter();

//out.println(sp_namex);

    out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
    out.println("<body background='image/41.gif'>");
    out.println("<center><img src='http://"+connect.hostname+"/image/bar2.gif'></center>");
    out.println("<br>");

    out.println("<SCRIPT language=JavaScript>");

    out.println("function doReload11(entry) {}");
    out.println("var current1,current2,current3;");
    out.println("var manufacture,model,year,master.url;");
    out.println("current1 = entry.manufacture.selectedIndex;");
    out.println("manufacture = entry.manufacture.options[current1].value;");
    out.println("current2 = entry.model.selectedIndex;");
    out.println("model = entry.model.options[current2].value;");
    out.println("current3 = entry.year.selectedIndex;");
    out.println("year = entry.year.options[current3].value;");

    out.println("master = entry.master.value;");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("sp_nameX=entry.sp_nameX.value;");
out.println("lastpage=entry.lastpage.value;");
    out.println("url =
\"http://"+connect.hostname+"/servlets/Form11?master="+master+"&manufacture="+manufacture+"&model="+mo
del+"&year="+year+"&sp_nameX="+sp_nameX+"&lastpage="+lastpage;");
    out.println("if (model!="");
    out.println("window.location.href = url;");
    out.println("}");

    out.println(" function doReload12(entry) {}");
    out.println(" var current1,current2,current3;");
    out.println(" var manufacture,model,year,master,url;");
    out.println("current1 = entry.manufacture.selectedIndex;");
    out.println("manufacture = entry.manufacture.options[current1].value;");
    out.println("current2 = entry.model.selectedIndex;");
    out.println("model = entry.model.options[current2].value;");
    out.println("current3 = entry.year.selectedIndex;");
    out.println("year = entry.year.options[current3].value;");
out.println("master = entry.master.value;");
out.println("sp_nameX=entry.sp_nameX.value;");
out.println("lastpage=entry.lastpage.value;");
    out.println("url =
\"http://"+connect.hostname+"/servlets/Form12?master="+master+"&manufacture="+manufacture+"&model="+mo
del+"&year="+year+"&sp_nameX="+sp_nameX+"&lastpage="+lastpage;");
    out.println("if (year!="");
    out.println("window.location.href = url;");
    out.println("}");

    out.println("</SCRIPT>");

    Connection con = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
try {
    con = pool.getConnection();
    con.setAutoCommit(false);
//-----
    out.println("<FORM METHOD='POST' ACTION='"+http://"+connect.hostname+"/servlets/page3'>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        rs.close();
    }
    catch(Exception e) {
        try {
            con.rollback();
        }
        catch (Exception ignored) {}
        //out.println("can not create connection in FORM "+ e.getMessage());
    }
    finally {
        if (con != null) pool.returnConnection(con);
    }

    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
}
}

```

ไฟล์ Form2.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form2 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+": "+connect.portnumber+": "+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
        catch(Exception e) {
            throw new UnavailableException(this, "Couldn't create connection pool");
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public void service(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException {

    String values[];

    String sp_namex="";
    values=req.getParameterValues("sp_namex");
    if(values!=null)
    {
        sp_namex=values[0];
    }

    String lastpage="";
    values=req.getParameterValues("lastpage");
    if(values!=null)
    {
        lastpage=values[0];
    }

    String master = "";
    values = req.getParameterValues("master");
    if (values!=null) { master = values[0]; }

    String manufacture = "";
    values = req.getParameterValues("manufacture");
    if (values!=null) { manufacture = values[0]; }

    String model = "";
    values = req.getParameterValues("model");
    if (values!=null) { model = values[0]; }

    String year = "";
    values = req.getParameterValues("year");
    if (values!=null) { year = values[0]; }

    res.setContentType("text/html");
    PrintWriter out = res.getWriter();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//out.println(sp_name);

        out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
out.println("<body background=\"image/41.gif\">");
        out.println("<center><img src=\"http://"+connect.hostname+"/image/bar2.gif\"></center>");
        out.println("<br>");

        out.println("<SCRIPT language=JavaScript>");

        out.println("function doReload111(entry) {");
        out.println("var current1,current2,current3;");
        out.println("var manufacture,model,year,master,url;");
        out.println("current1 = entry.manufacture.selectedIndex;");
        out.println("manufacture = entry.manufacture.options[current1].value;");
        out.println("current2 = entry.model.selectedIndex;");
        out.println("model = entry.model.options[current2].value;");
        out.println("current3 = entry.year.selectedIndex;");
        out.println("year = entry.year.options[current3].value;");
        out.println("master = entry.master.value;");
        out.println("sp_name=entry.sp_name.value;");
        out.println("lastpage=entry.lastpage.value;");
        out.println("url =
\"http://"+connect.hostname+"/servlets/Form111?master="+master+"&manufacture="+manufacture+"&model="+m
odel+"&year="+year+"&sp_name="+sp_name+"&lastpage="+lastpage;");
        out.println("if (year!="");
        out.println("window.location.href = url;");
        out.println("}");

        out.println("</SCRIPT>");

        Connection con = null;
        PreparedStatement pstmt = null;
        ResultSet rs = null;
try {
        con = pool.getConnection();
        con.setAutoCommit(false);
//-----
        out.println("<FORM METHOD=\"POST\" ACTION=\"http://"+connect.hostname+"/servlets/page3\">");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

catch(Exception e) {
    try {
        con.rollback();
    }
    catch (Exception ignored) {}
    //out.println("can not create connection in FORM "+ e.getMessage());
}
finally {
    if (con != null) pool.returnConnection(con);
}

```

```

        out.println("</BODY>");
        out.println("</HTML>");
    out.close();
}
}

```

ไฟล์ Form3.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form3 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+": "+connect.portnumber+": "+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
        catch(Exception e) {
            throw new UnavailableException(this, "Couldn't create connection pool");
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public void service(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException {
```

```
    String values[];
```

```
    String sp_namex="";
```

```
    values=req.getParameterValues("sp_namex");
```

```
    if(values!=null)
```

```
    {
```

```
        sp_namex=values[0];
```

```
    }
```

```
    String lastpage="";
```

```
    values=req.getParameterValues("lastpage");
```

```
    if(values!=null)
```

```
    {
```

```
        lastpage=values[0];
```

```
    }
```

```
    String master = "";
```

```
    values = req.getParameterValues("master");
```

```
    if (values!=null) { master = values[0]; }
```

```
    String manufacture = "";
```

```
    values = req.getParameterValues("manufacture");
```

```
    if (values!=null) { manufacture = values[0]; }
```

```
    String model = "";
```

```
    values = req.getParameterValues("model");
```

```
    if (values!=null) { model = values[0]; }
```

```
    String year = "";
```

```
    values = req.getParameterValues("year");
```

```
    if (values!=null) { year = values[0]; }
```

```
    res.setContentType("text/html");
```

```
    PrintWriter out = res.getWriter();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//out.println(sp_namex);

        out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
out.println("<body background='/image/41.gif'>");
        out.println("<center><img src='http://"+connect.hostname+"/image/bar2.gif'></center>");
        out.println("<br>");
        out.println("<SCRIPT language=JavaScript>");

        out.println("function doReload31(entry) {");
        out.println("var current1,current2,current3;");
        out.println("var manufacture,model,year,master,url;");
        out.println("current1 = entry.manufacture.selectedIndex;");
        out.println("manufacture = entry.manufacture.options[current1].value;");
        out.println("current2 = entry.model.selectedIndex;");
        out.println("model = entry.model.options[current2].value;");
        out.println("current3 = entry.year.selectedIndex;");
        out.println("year = entry.year.options[current3].value;");
        out.println("master = entry.master.value;");
        out.println("sp_namex=entry.sp_namex.value;");
        out.println("lastpage=entry.lastpage.value;");
        out.println("url =
'"+http://"+connect.hostname+"/servlets/Form12?master="+master+"&manufacture="+manufacture+"&model="+mo
del+"&year="+year+"&sp_namex="+sp_namex+"&lastpage="+lastpage;");
        out.println("if (manufacture!="");
        out.println("window.location.href = url;");
        out.println("}");

        out.println("function doReload32(entry) {");
        out.println("var current1,current2,current3;");
        out.println("var manufacture,model,year,master,url;");
        out.println("current1 = entry.manufacture.selectedIndex;");
        out.println("manufacture = entry.manufacture.options[current1].value;");
        out.println("current2 = entry.model.selectedIndex;");
        out.println("model = entry.model.options[current2].value;");
        out.println("current3 = entry.year.selectedIndex;");
        out.println("year = entry.year.options[current3].value;");
        out.println("master = entry.master.value;");
        out.println("sp_namex=entry.sp_namex.value;");
        out.println("lastpage=entry.lastpage.value;");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
out.println("</SELECT>");
```

```
//-----END-----
```

```
out.println("</FONT></TD></TR>");
```

```
out.println("</TBODY></TABLE>< CENTER></TD></TR></TBODY></TABLE></CENTER>");
```

```
out.println("<BR><div align=\"center \"><center><p>");
```

```
out.println("<INPUT TYPE=\"submit\" value=\"ค้นหาะโหล่ง\" name=\"B1\">");
```

```
out.println("<A href=\"http://\"+connect.hostname+\"/servlets/Sap\"><FONT color=\"#0000FF\" size=4>ชื้อน
```

```
ถึลึบ</FONT></A>");
```

```
out.println("</p></center></div>");
```

```
out.println("</FORM>");
```

```
con.commit();
```

```
pstmt.close();
```

```
rs.close();
```

```
}
```

```
catch (Exception e) {
```

```
try {
```

```
con.rollback();
```

```
}
```

```
catch (Exception ignored) {}
```

```
//out.println("can not create connection in FORM "+ e.getMessage());
```

```
}
```

```
finally {
```

```
if (con != null) pool.returnConnection(con);
```

```
}
```

```
out.println("</BODY >");
```

```
out.println("</HTML>");
```

```
out.close();
```

```
{
```

```
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์ Form11.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form11 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+": "+connect.portnumber+" "+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
        catch(Exception e) {
            throw new UnavailableException(this, "Couldn't create connection pool");
        }
    }

    public void service(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {

        String values[];

        String sp_namex="";
        values=req.getParameterValues("sp_namex");
        if(values!=null)
        {
            sp_namex=values[0];
        }

        String lastpage="";
        values=req.getParameterValues("lastpage");
        if(values!=null)
        {
            lastpage=values[0];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

String master = "";
values = req.getParameterValues("master");
if (values!=null) { master = values[0]; }

String manufacture = "";
values = req.getParameterValues("manufacture");
if (values!=null) { manufacture = values[0]; }

String model = "";
values = req.getParameterValues("model");
if (values!=null) { model = values[0]; }

String year = "";
values = req.getParameterValues("year");
if (values!=null) { year = values[0]; }

res.setContentType("text/html");
PrintWriter out = res.getWriter();

out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
out.println("<body background='\"/image/41.gif\"'>");
out.println("<center><img src='\"http://\"+connect.hostname+\"/image/bar2.gif\"'></center>");
out.println("<br>");

out.println("<SCRIPT language=JavaScript>");

out.println("function doReload11(entry) {");
out.println("var current1,current2,current3:");
out.println("var manufacture,model,year,master,url:");
out.println("current1 = entry.manufacture.selectedIndex;");
out.println("manufacture = entry.manufacture.options[current1].value;");
out.println("current2 = entry.model.selectedIndex;");
out.println("model = entry.model.options[current2].value;");
out.println("current3 = entry.year.selectedIndex;");
out.println("year = entry.year.options[current3].value;");
out.println("master = entry.master.value;");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        out.println("<BR><div align='center'><center><p>");
        out.println("<INPUT TYPE='submit' value='ค้นหาชื่อใหม่' name='B1'>");
        out.println("<A href='http://"+connect.hostname+"/servlets/Sap'><FONT color='#0000FF' size=4>ชื่อใหม่  

กลับ</FONT></A>");
        out.println("</p></center></div>");
        out.println("</FORM>");

        con.commit();
        pstmt.close();
        rs.close();

    }
    catch(Exception e) {
        try {
            con.rollback();
        }
        catch (Exception ignored) { }
        //out.println("can not create connection in FORM "+ e.getMessage());
    }
    finally {
        if (con != null) pool.returnConnection(con);
    }

        out.println("</BODY>");
        out.println("</HTML>");
    out.close();
}
}

```

ไฟล์ Form12.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form12 extends HttpServlet {
    private ConnectionPool pool;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public void init(ServletConfig config) throws ServletException {
    super.init(config);
    try {
        pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+": "+connect.portnumber+": "+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
    }
    catch(Exception e) {
        throw new UnavailableException(this, "Couldn't create connection pool");
    }
}

public void service(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException {

    String values[];

    String sp_namex="";
    values=req.getParameterValues("sp_namex");
    if(values!=null)
    {
        sp_namex=values[0];
    }

    String lastpage="";
    values=req.getParameterValues("lastpage");
    if(values!=null)
    {
        lastpage=values[0];
    }

    String master = "";
    values = req.getParameterValues("master");
    if (values!=null) { master = values[0]; }

    String manufacture = "";
    values = req.getParameterValues("manufacture");
    if (values!=null) { manufacture = values[0]; }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String model = "";
values = req.getParameterValues("model");
if (values!=null) { model = values[0]; }

String year = "";
values = req.getParameterValues("year");
if (values!=null) { year = values[0]; }

res.setContentType("text/html");
PrintWriter out = res.getWriter();

//out.println(sp_namex);

        out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
out.println("<body background=\\\"/image/41.gif\\\">");
        out.println("<center><img src=\\\"http://"+connect.hostname+"/image/bar2.gif\\\"></center>");
        out.println("<br>");

        out.println("<SCRIPT language=JavaScript>");

        out.println("function doReload111(entry) {");
        out.println("var current1,current2,current3;");
        out.println("var manufacture,model,year,master,url;");
        out.println("current1 = entry.manufacture.selectedIndex;");
        out.println("manufacture = entry.manufacture.options[current1].value;");
        out.println("current2 = entry.model.selectedIndex;");
        out.println("model = entry.model.options[current2].value;");
        out.println("current3 = entry.year.selectedIndex;");
        out.println("year = entry.year.options[current3].value;");

        out.println("master = entry.master.value;");
        out.println("sp_namex=entry.sp_namex.value;");
        out.println("lastpage=entry.lastpage.value;");

        out.println("url =
\\\"http://"+connect.hostname+"/servlets/Form111?master=\\\"+master+\\\"&manufacture=\\\"+manufacture+\\\"&model=\\\"+m
odel+\\\"&year=\\\"+year+\\\"&sp_namex=\\\"+sp_namex+\\\"&lastpage=\\\"+lastpage;");

        out.println("if (model!=\\\"\\\")");
        out.println("window.location.href = ud:");
        out.println("}");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

}
catch(Exception e) {
    try {
        con.rollback();
    }
    catch (Exception ignored) {}
    //out.println("can not create connection in FORM "+ e.getMessage());
}
finally {
    if (con != null) pool.returnConnection(con);
}

```

```

        out.println("</BODY>");
        out.println("</HTML>");
    out.close();
}
}

```

ไฟล์ Form32.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form32 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
        try {
            pool = new
ConnectionPool("jdbc:oracle:thin:@"+connect.hostname+": "+connect.portnumber+": "+connect.databaseSID,connect.us
ername,connect.password,"oracle.jdbc.driver.OracleDriver", 10, 5);
        }
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

catch(Exception e) {
    throw new UnavailableException(this, "Couldn't create connection pool");
}
}

```

```

public void service(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException {

```

```

    String values[];

```

```

    String sp_namex="";

```

```

    values=req.getParameterValues("sp_namex");

```

```

    if(values!=null)

```

```

    {

```

```

        sp_namex=values[0];

```

```

    }

```

```

    String lastpage="";

```

```

    values=req.getParameterValues("lastpage");

```

```

    if(values!=null)

```

```

    {

```

```

        lastpage=values[0];

```

```

    }

```

```

    String master = "";

```

```

    values = req.getParameterValues("master");

```

```

    if (values!=null) { master = values[0]; }

```

```

    String manufacture = "";

```

```

    values = req.getParameterValues("manufacture");

```

```

    if (values!=null) { manufacture = values[0]; }

```

```

    String model = "";

```

```

    values = req.getParameterValues("model");

```

```

    if (values!=null) { model = values[0]; }

```

```

    String year = "";

```

```

    values = req.getParameterValues("year");

```

```

    if (values!=null) { year = values[0]; }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        out.println("</FORM>");

        con.commit();
        pstmt.close();
        rs.close();
    }
    catch(Exception e) {
        try {
            con.rollback();
        }
        catch (Exception ignored) { }
        //out.println("can not create connection in FORM "+ e.getMessage());
    }
    finally {
        if (con != null) pool.returnConnection(con);
    }

    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
}
}

```

ไฟล์ Form111.java

```

import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Form111 extends HttpServlet {
    private ConnectionPool pool;

    public void init(ServletConfig config) throws ServletException {
        super.init(config);
    }

    public void service(HttpServletRequest req, HttpServletResponse res)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
throws ServletException, IOException {
```

```
String values[];
```

```
String sp_namex="";
```

```
values=req.getParameterValues("sp_namex");
```

```
if(values!=null)
```

```
{
```

```
    sp_namex=values[0];
```

```
}
```

```
String lastpage="";
```

```
values=req.getParameterValues("lastpage");
```

```
if(values!=null)
```

```
{
```

```
    lastpage=values[0];
```

```
}
```

```
String master = "";
```

```
values = req.getParameterValues("master");
```

```
if (values!=null) { master = values[0]; }
```

```
String manufacture = "";
```

```
values = req.getParameterValues("manufacture");
```

```
if (values!=null) { manufacture = values[0]; }
```

```
String model = "";
```

```
values = req.getParameterValues("model");
```

```
if (values!=null) { model = values[0]; }
```

```
String year = "";
```

```
values = req.getParameterValues("year");
```

```
if (values!=null) { year = values[0]; }
```

```
res.setContentType("text/html");
```

```
PrintWriter out = res.getWriter();
```

```
    out.println("<HTML><HEAD><TITLE>Choose Car</TITLE></HEAD>");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
//-----3-----

    out.println("<SELECT NAME='year' SIZE='1'>");
    out.println("<OPTION SELECTED VALUE='\" + year+ "\"> + year+ "</OPTION>");
    out.println("</SELECT>");

//-----END-----

    out.println("</FONT></TD></TR>");
    out.println("</TBODY></TABLE></CENTER></TD></TR></TBODY></TABLE></CENTER>");

    out.println("<BR><div align='center'><center><p>");
    out.println("<INPUT TYPE='submit' value='ค้นหาชื่อให้สิ' name='BI'>");
    out.println("<A href='http://\" + connect.hostname+ "/servlets/Sap'><FONT color='\"#0000FF\" size=4>ย้อน  
กลับ</FONT></A>");
    out.println("</p></center></div>");
    out.println("</FORM>");
    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
}
}
```

ไฟล์ sparequery.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.lang.*;

public class sparequery extends HttpServlet
{
    public void service(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {

        String values[];

        String sp_name="";
        values=req.getParameterValues("sp_name");
        if(values!=null)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sp_namex=values[0];
sp_namex=sp_namex.replace('.', '');
sp_namex=sp_namex.trim();
}

```

```

String lastpage="";
values=req.getParameterValues("lastpage");
if(values!=null)
{
    lastpage=values[0];
}

```

```

String manufacture = "";
values = req.getParameterValues("manufacture");
if (values!=null) { manufacture = values[0]; }

```

```

String model = "";
values = req.getParameterValues("model");
if (values!=null) { model = values[0]; }

```

```

String year = "";
values = req.getParameterValues("year");
if (values!=null) { year = values[0]; }

```

```

String master = "";
values = req.getParameterValues("master");
if (values!=null) { master = values[0]; }

```

```

res.setContentType("text/html");
PrintWriter out = res.getWriter();

```

```

        out.println("<HTML><HEAD><TITLE>Query Spare part</TITLE></HEAD>");
out.println("<body background=\\'/image/61.gif\\'><center>");

```

```

        out.println("ผู้ผลิตรถยนต์(manufacture) :<font color=\\'##AA0000\\'>"+manufacture+"</font><br>");
        out.println("รุ่น(model) :<font color=\\'##AA000077\\'>"+model+"</font><br>");
        out.println("ปี(year) :<font color=\\'##AA0000\\'>"+year+"</font><br>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("ชื่อไฟ้(master) :<font color=#AA0000#>"+master+"</font><br>");
out.println("spare_part:"+sp_namex);

```

```

out.println("</center></body></html>");

```

```

out.println("<form method=POST action='"+http://"+connect.hostname+"/servlet/"+page3+"'\>");
out.println("<input type=submit name=view value='view'\>");
out.println("<input type=hidden name=sp_namex value='"+sp_namex+"'\>");
out.println("<INPUT TYPE='hidden' NAME='lastpage' VALUE='"+lastpage+"'\>");
out.println("<input type=hidden name=manufacture value='"+manufacture+"'\>");
out.println("<input type=hidden name=model value='"+model+"'\>");
out.println("<input type=hidden name=year value='"+year+"'\>");
out.println("</form>");

```

```

out.close();

```

```

}

```

```

}

```

ไฟล์ pic.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;
import java.util.*;
import java.lang.*;

```

```

public class pic extends HttpServlet

```

```

{

```

```

    String values[];

```

```

    public void service(HttpServletRequest req,HttpServletResponse resp)

```

```

    throws ServletException,java.io.IOException

```

```

    {

```

```

        resp.setContentType("text/html");

```

```

        PrintWriter out = resp.getWriter();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String pic="";
values=req.getParameterValues("pic");
if(values!=null)
{
    pic=values[0];
    //out.println(pic);
}

String code="";
values=req.getParameterValues("code");
if(values!=null)
{
    code=values[0];
    //out.println(pic);
}

out.println("<html>");
out.println("<head>");
out.println("<title>test car query</title>");
out.println("</head>");

out.println("<body background=\\'/image/41.gif\\'>");
out.println("<center><h3>รหัสสินค้า คือ "+ <font color=\\'##AA0000\\' size=5>"+code+"</font></h3></center>");
out.println("<center><p><img border = \\'1\\' src=\\'http://"+connect.hostname+"/image/"+pic+".jpg\\'></p>
</center>");

    out.println("</body>");
out.println("</html>");
}
}

```

ไฟล์ page3.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public class page3 extends HttpServlet
{
String values[];
Connection con = null;

public void service(HttpServletRequest req,HttpServletResponse resp)
throws ServletException,java.io.IOException
{
connectdb();
getsparepart(req,resp);
}

public void connectdb()
{
try
{
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database
con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+"."+
connect.portnumber+"."+connect.databaseSID
,connect.username,connect.password);
}
catch(SQLException ex)
{
System.out.println("error while initializingg database connection"+ex.toString());
}
}

public void getsparepart(HttpServletRequest req,HttpServletResponse resp)
{
try
{
resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

HttpSession session=req.getSession(true);

Statement stmt,stmt1 = null;
ResultSet rs,rs1 = null;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

stmt = con.createStatement();
stmt1 = con.createStatement();
String query="";

Vector vector_code =new Vector();
Vector vector_manufac=new Vector();
Vector vector_price =new Vector();
Vector vector_stock =new Vector();
Vector vector_model =new Vector();
Vector vector_lr =new Vector();
Vector vector_year =new Vector();
Vector vector_pic =new Vector();
Vector vector_des =new Vector();
Vector vector_piccode=new Vector();
Vector vector_codex;
Vector vector_stockx;

out.println("<html>");
out.println("<head>");
out.println("<title>Shopping Spare Part</title>");
out.println("</head>");
out.println("<body background='/image/41.gif'>");
out.println("<center><img src='http://'+connect.hostname+'/image/bar3.gif'></center>");
out.println("<br>");

out.println("<center><font color='#000080'><b3>4</b3> ตั้งซื้อชิ้นส่วนอะไหล่ที่ต้องการ</b3></font></center>");
out.println("<center><font size=2>การตั้งซื้ออะไหล่ทำได้โดยเลือกจำนวนอะไหล่ แล้วกดปุ่ม <font color=#AA0000
size=5>PICK</font> ในกรณีที่ต้องการยกเลิกการตั้งซื้อทำได้โดยกดปุ่ม <font color=#AA0000 size=5>UNPICK </font>
</font></center>");

String sp_namex="";
values=req.getParameterValues("sp_namex");
if(values!=null)
{
sp_namex=values[0];
sp_namex=sp_namex.replace(' ','');
sp_namex=sp_namex.trim();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    out.println("<h3><b>ชื่อชิ้นที่คุณเลือก : <font color='#000080'>" + sp_name + "</font></b></h3>"),
    out.println("<br>");

    out.println("<br>");

    String lastpage="";
    values=req.getParameterValues("lastpage");
    if(values!=null)
    {
        lastpage=values[0];
    }

/* out.println("your spare part you choose is");
    out.println(sp_name);
    out.println("<br>");*/

    String manufacture="";
    values=req.getParameterValues("manufacture");
    if(values!=null)
    {
        manufacture=values[0];
    }
/* out.println("your manufacture you choose is");
    out.println(manufacture);
    out.println("<br>");*/

    String model="";
    values=req.getParameterValues("model");
    if(values!=null)
    {
        model=values[0];
    }
/* out.println("your model you choose is");
    out.println(model);
    out.println("<br>");*/

    String year="";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

values=req.getParameterValues("year");
if(values!=null)
{
    year=values[0];
}
/* out.println("your year you choose is");
out.println(year);
out.println("<br>");*/

String codex="";
values=req.getParameterValues("codex");
if(values!=null)
{
    codex=values[0];
}

String stockx="";
values=req.getParameterValues("stockx");
if(values!=null)
{
    stockx=values[0];
}

String findx="";
values=req.getParameterValues("findx");
if(values!=null)
{
    findx=values[0];
}

// out.println(findx);

// out.println(findx);
vector_codex=(Vector)session.getValue("mycodex");
vector_stockx=(Vector)session.getValue("mystockx");

if(vector_codex==null)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

vector_codex=new Vector();
}
if(vector_stockx==null)
{
vector_stockx=new Vector();
}

//out.println(lastpage);

if(lastpage.compareTo("yes")==0) //if have submit page 5 before
{
lastpage="no";
clearelement(vector_codex);
clearelement(vector_stockx);
//out.println("it's have go to last page before");
}

if(findx.compareTo("PICK")==0)
{
int ff=0;
for(ff=0;ff<vector_codex.size();ff++)
{
if(codex.compareTo((String)vector_codex.elementAt(ff))==0)
{
vector_codex.removeElement((String)vector_codex.elementAt(ff));
vector_stockx.removeElement((String)vector_stockx.elementAt(ff));
//out.println("remove same value");
}
}
//out.println("OK buy this one");
vector_codex.addElement(codex);
vector_stockx.addElement(stockx);
}

if(findx.compareTo("UNPICK")==0)
{
vector_codex.removeElement(codex);
vector_stockx.removeElement(stockx);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

session.putValue("mycodex",vector_codex);
session.putValue("mystockx",vector_stockx);

//test=====
/* int test=0;
for(test=0;test<vector_codex.size();test++)
{
    out.println((String)vector_codex.elementAt(test));
    out.println((String)vector_stockx.elementAt(test));
}*/

//=====test
if((manufacture.compareTo("")==0)&&(model.compareTo("")==0)&&(year.compareTo("")==0))
{
    //out.println("000");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND car_code IN (SELECT car_code FROM match WHERE
sp_code=s.sp_code)";
}
//out.println("<br>");

if((manufacture.compareTo("")==0)&&(model.compareTo("")==0)&&(year.compareTo("")!=0))
{ //out.println("001");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND year='"+year+"' AND car_code IN (SELECT car_code FROM match
WHERE sp_code=s.sp_code)";
}
//out.println("<br>");

if((manufacture.compareTo("")==0)&&(model.compareTo("")!=0)&&(year.compareTo("")==0))
{ //out.println("010");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND model='"+model+"' AND car_code IN (SELECT car_code FROM
match WHERE sp_code=s.sp_code)";
}
//out.println("<br>");

if((manufacture.compareTo("")==0)&&(model.compareTo("")!=0)&&(year.compareTo("")!=0))
{ //out.println("011");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND year='"+year+"' AND model='"+model+"' AND car_code IN
(SELECT car_code FROM match WHERE sp_code=s.sp_code);
}
//out.println("<br>");

if((manufacture.compareTo("")!=0)&&(model.compareTo("")==0)&&(year.compareTo("")==0))
{ //out.println("100");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND sp_manufac='"+manufacture+"' AND car_code IN (SELECT car_code
FROM match WHERE sp_code=s.sp_code);
}
//out.println("<br>");

if((manufacture.compareTo("")!=0)&&(model.compareTo("")==0)&&(year.compareTo("")!=0))
{ //out.println("101");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND year='"+year+"' AND sp_manufac='"+manufacture+"' AND car_code
IN (SELECT car_code FROM match WHERE sp_code=s.sp_code);
}
//out.println("<br>");

if((manufacture.compareTo("")!=0)&&(model.compareTo("")!=0)&&(year.compareTo("")==0))
{ //out.println("110");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND sp_manufac='"+manufacture+"' AND model='"+model+"' AND
car_code IN (SELECT car_code FROM match WHERE sp_code=s.sp_code);
}
//out.println("<br>");

if((manufacture.compareTo("")!=0)&&(model.compareTo("")!=0)&&(year.compareTo("")!=0))
{ //out.println("111");
    query="select sp_code,sp_manufac,sp_lr,sp_stock,sp_price,model,year,sp_des,sp_pic,sp_piccode FROM spare_part
s,car c WHERE sp_name='"+sp_namex+"' AND sp_manufac='"+manufacture+"' AND model='"+model+"' AND
year='"+year+"' AND car_code IN (SELECT car_code FROM match WHERE sp_code=s.sp_code);
}
//out.println("<br>");

rs=stmt.executeQuery(query);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// out.println("<br>");
while(rs.next())
{
    vector_code.addElement(rs.getString(1));
    vector_manufac.addElement(rs.getString(2));
    vector_lr.addElement(rs.getString(3));
    vector_stock.addElement(rs.getString(4));
    vector_price.addElement(rs.getString(5));
    vector_model.addElement(rs.getString(6));
    vector_year.addElement(rs.getString(7));
    vector_des.addElement(rs.getString(8));
    vector_pic.addElement(rs.getString(9));
    vector_piccode.addElement(rs.getString(10));
}

int i=0;
// out.println(vector_code.size());
for(i=0;i<vector_code.size();i++)
{
    out.println("<table border width=100%>");
    out.println("<table border>");

    if((((String)vector_pic.elementAt(i)).compareTo("NULL")!=0)&&(((String)vector_pic.elementAt(i)).compareTo
("null")!=0))
    {
        out.println("<tr><td><b>PRODUCT CODE:</b></td><td>"+<a
href="http://"+connect.hostname+"/servlet/pic/?pic="+<String>vector_pic.elementAt(i)+"&code="+<String>
vector_piccode.elementAt(i)+"'">"+<String>vector_code.elementAt(i)+"</a></td></tr>");
    }
    else
    {
        out.println("<tr><td><b>PRODUCT CODE:</b></td><td>"+<String>vector_code.elementAt(i)+"</td></tr>");
    }

    out.println("<tr><td><b>MANUFACTURE:</b></td><td>"+<String>vector_manufac.elementAt(i)+"</td></tr>");
    out.println("<tr><td><b>SIDE:</b></td><td>"+<String>vector_lr.elementAt(i)+"</td></tr>");
    out.println("<tr><td><b>DESCRIBE:</b></td><td>"+<String>vector_des.elementAt(i)+"</td></tr>");
    out.println("<tr><td><b>PRICE:</b></td><td>"+<String>vector_price.elementAt(i)+"</td></tr>");
    out.println("<tr><td><b>MODEL:</b></td><td>"+<String>vector_model.elementAt(i)+"</td></tr>");
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<tr><td><b>YEAR:</b></td><td>"+(String)vector_year.elementAt(i)+"</td></tr>");

out.println("</table>");
out.println("</table>");

out.println("<form method=POST action=\""+"http://"+connect.hostname+"/servlet/page3"+"\">");
out.println("<input type=submit name=findx value=\""+"PICK"+"\">");
out.println("<input type=submit name=findx value=\""+"UNPICK"+"\">");
out.println("<input type=hidden name=sp_namex value=\""+sp_namex+"\">");
out.println("<input type=hidden name=lastpage value=\""+lastpage+"\">");
out.println("<input type=hidden name=codex value=\""+(String)vector_code.elementAt(i)+"\">");
out.println("<input type=hidden name=manufacture value=\""+manufacture+"\">");
out.println("<input type=hidden name=model value=\""+model+"\">");
out.println("<input type=hidden name=year value=\""+year+"\">");

int stk=Integer.parseInt((String)vector_stock.elementAt(i));
int k=0;
out.println("<select name=stockx size=1-");
for(k=1;k<stk+1;k++)
{
String k2=Integer.toString(k);
out.println("<option value=\""+k2+"\">"+k2+"");
}
out.println("</select>");

out.println("</form>");
}
// out.println("<br>");

if((vector_code.size())<=0))
{
out.println("<font size=3 color=#AA0000><b>ไม่พบอะไหล่ที่คุณต้องการ.</font></b>");
out.println("<br><br>");
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((vector_codex.size())<=0)&&(vector_code.size())<=0)
{
    out.println("");
}
else
{
    out.println("<form method=POST action=\""+http://"+connect.hostname+"/servlet/page4"+"\">");
    out.println("<input type=submit name=view value=\" >> ดูรายการสินค้าที่เลือกซื้อ >> \">");
    if(vector_codex.size())>0)
    {
        String send=null;
        String send2=null;
        if(vector_codex.size())>1)
        {
            send="<input type=hidden name=codex_page3 value=\"";
            send2="<input type=hidden name=stockx_page3 value=\"";
            for(i=0;i<vector_codex.size()-1;i++)
            {
                send=send+(String)vector_codex.elementAt(i);
                send2=send2+(String)vector_stockx.elementAt(i);
                send=send+" ";
                send2=send2+" ";
            }
            send=send+(String)vector_codex.elementAt(i);
            send2=send2+(String)vector_stockx.elementAt(i);
            send=send+"\">";
            send2=send2+"\">";
        }//if
    }
    else
    {
        if(vector_codex.size())=1)
        {
            send="<input type=hidden name=codex_page3 value=\""+(String)vector_codex.elementAt(0)+"\">";
            send2="<input type=hidden name=stockx_page3 value=\""+(String)vector_stockx.elementAt(0)+"\">";
        }//if
    }//else
    out.println(send);
    out.println(send2);
}//if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<input type=hidden name=sp_name value=\""+sp_name+"\">");
out.println("<input type=hidden name=lastpage value=\""+lastpage+"\">");
out.println("<input type=hidden name=manufacture value=\""+manufacture+"\">");
out.println("<input type=hidden name=model value=\""+model+"\">");
out.println("<input type=hidden name=year value=\""+year+"\">");
out.println("</form>");
}
out.println("<a href=\"http://"+connect.hostname+"/servlet/Sap\">เลือกซื้ออะไหล่เพิ่มเติม</a>");

out.println("</body>");
out.println("</html>");

clearElement(vector_code);
clearElement(vector_manufac);
clearElement(vector_price);
clearElement(vector_stock);
clearElement(vector_model);
clearElement(vector_lr);
clearElement(vector_year);
stmt.close();
con.close();
}
catch(Exception ex)
{
ex.printStackTrace();
}
}

public void clearElement(Vector vec)
{
String dummy;
int i,j;

if(vec.size()!=0)
{
j=vec.size();
for(i=0;i<j;i++)
{
dummy=(String)vec.elementAt(0);
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        vec.removeElement(dummy);
    }
}
}
}

```

ไฟล์ page4.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;
import java.util.*;
import java.lang.*;

public class page4 extends HttpServlet
{

    String values[];
    Connection con = null;
    Statement stmt = null;
    ResultSet rs = null;

    public void service(HttpServletRequest req,HttpServletResponse resp)
    throws ServletException,java.io.IOException
    {
        connectdb();
        list_goods(req,resp);
    }

    public void connectdb()
    {
        try
        {
            DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
            // Get a Connection to the database
            con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+"."+
                connect.portnumber+"."+connect.databaseSID

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        .connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
    System.out.println("error while initializingg database connection"+ex.toString());
}
}

public void list_goods(HttpServletRequest req,HttpServletResponse resp)
{
    try
    {
        Vector vector_code =new Vector();
        Vector vector_manufac=new Vector();
        Vector vector_price =new Vector();
        Vector vector_stock =new Vector();
        Vector vector_name =new Vector();
        Vector vector_lr =new Vector();
        Vector vector_model =new Vector();
        Vector vector_year =new Vector();
        Vector vector_stockx=new Vector();

        resp.setContentType("text/html");
        PrintWriter out = resp.getWriter();

        String sp_namex="";
        values=req.getParameterValues("sp_namex");
        if(values!=null)
        {
            sp_namex=values[0];
        }

        String lastpage="";
        values=req.getParameterValues("lastpage");
        if(values!=null)
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lastpage=values[0];
}

// out.println(lastpage+" pppppp");

String manufacture="";
values=req.getParameterValues("manufacture");
if(values!=null)
{
    manufacture=values[0];
}
/* out.println("your manufacture you choose is");
out.println(manufacture);
out.println("<br>");*/

String model="";
values=req.getParameterValues("model");
if(values!=null)
{
    model=values[0];
}
/* out.println("your model you choose is");
out.println(model);
out.println("<br>");*/

String year="";
values=req.getParameterValues("year");
if(values!=null)
{
    year=values[0];
}
/* out.println("your year you choose is");
out.println(year);
out.println("<br>");*/

String codex_page3="";
values=req.getParameterValues("codex_page3");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(values!=null)
{
    codex_page3=values[0];
    //out.println("found!");
}

String stockx_page3="";
values=req.getParameterValues("stockx_page3");
if(values!=null)
{
    stockx_page3=values[0];
    //out.println("found!");
}

/* out.println("your code you choose is");
out.println(codex_page3);
out.println(stockx_page3); */

StringTokenizer tk=new StringTokenizer(codex_page3);
while(tk.hasMoreTokens())
{
    vector_code.addElement(tk.nextToken());
}

StringTokenizer sk=new StringTokenizer(stockx_page3);
while(sk.hasMoreTokens())
{
    vector_stockx.addElement(sk.nextToken());
}

int i=0;
for(i=0;i<vector_code.size();i++)
{
    String query="select sp.sp_name,sp.sp_manufac,sp.sp_stock,sp.sp_lr,c.model,c.year,sp.sp_price from spare_part
sp,car c where sp.sp_code='"+(String)vector_code.elementAt(i)+"' and c.car_code=(select car_code from match where
sp_code='"+(String)vector_code.elementAt(i)+"'");

    rs = stmt.executeQuery(query);
    rs.next();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

vector_name.addElement(rs.getString(1));
vector_menufac.addElement(rs.getString(2));
vector_stock.addElement(rs.getString(3));
vector_lr.addElement(rs.getString(4));
vector_model.addElement(rs.getString(5));
vector_year.addElement(rs.getString(6));
vector_price.addElement(rs.getString(7));
}

out.println("<html>");
out.println("<head>");
    out.println("<title>Shopping Spare Part</title>");
out.println("</head>");
out.println("<body background='\"/image/41.gif\">");

out.println("<center><img src='\"http://\"+connect.hostname+\"/image/bar3.gif\"></center>");
out.println("<h4><font color='\"#000080\">รายการสินค้าที่สั่งซื้อ มีดังนี้</font><b4>");
//out.println("<br>");

out.println("<table border width= 80%>");
out.println("<tr><th>code</th><th>name</th><th>menufac</th><th>number</th><th>side</th><th>model</th>");
<th>year</th><th>price</th><th>totalprice</th></tr>");
int pn1=0;
for(i=0;i<vector_code.size();i++)
{
    int p1=Integer.parseInt((String)vector_price.elementAt(i));
    int n1=Integer.parseInt((String)vector_stockx.elementAt(i));//number that the user buy
    int n2=Integer.parseInt((String)vector_stock.elementAt(i));//number of stock
    if(n1>n2) //if user buy more than stock have
    {
        n1=n2;
    }
    int pn=p1*n1;
    pn1=pn1+pn;

out.println("<tr><td>"+(String)vector_code.elementAt(i)+"</td><td>"+(String)vector_name.elementAt(i)+"</td>");
<td>"+(String)vector_menufac.elementAt(i)+"</td><td>"+(String)vector_stockx.elementAt(i)+"</td><td>"+(String)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

vector_lr.elementAt(i)+"</td><td>"+(String)vector_model.elementAt(i)+"</td><td>"+(String)vector_year.elementAt
(i)+"</td><td>"+(String)vector_price.elementAt(i)+"</td><td>"+pn+"</td></tr>";
    }
    out.println("<tr><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+ " "+
</td><td>"+ " "</td><td>"+ " "</td><td>"+pn1+"</td></tr>");
    out.println("</table>");
// out.println("<a href='\"http://\"+connect.hostname+\"/sap2.html\">CHOOSE MORE</a>");

    out.println("<form method=POST action='\"+\"http://\"+connect.hostname+\"/servlet/\"+\"page3\"+\"'\>");
    out.println("<input type=submit name=view value='\" แก้ไขรายการอะไหล่ที่ตั้งชื่อ '\">");
    out.println("<input type=hidden name=sp_name value='\"+sp_name+\"'\>");
    out.println("<INPUT TYPE='\"hidden\"' NAME='\"lastpage\"' VALUE='\""+lastpage+\"'\>");
    out.println("<input type=hidden name=manufacture value='\""+manufacture+\"'\>");
    out.println("<input type=hidden name=model value='\""+model+\"'\>");
    out.println("<input type=hidden name=year value='\""+year+\"'\>");
    out.println("<a href='\"http://\"+connect.hostname+\"/servlet/Sap\">เลือกซื้ออะไหล่เพิ่มเติม</a>");
    out.println("</form>");

    out.println("<br>");

    out.println("<form method=POST action='\"+\"http://\"+connect.hostname+\"/servlet/\"+\"page6\"+\"'\>");
    out.println("<input type=submit name=view value='\"nonmember\"'\>");
    out.println("<input type=hidden name=codex_page4 value='\""+codex_page3+\"'\>");
    out.println("<input type=hidden name=stockx_page4 value='\""+stockx_page3+\"'\>");
    out.println("</form>");

    //out.println("<br>");

    out.println("<form method=POST action='\"+\"http://\"+connect.hostname+\"/servlet/\"+\"login\"+\"'\>");

    out.println("<input type=submit name=view value='\"member\"'\>");
    out.println("<input type=hidden name=codex_page4 value='\""+codex_page3+\"'\>");
    out.println("<input type=hidden name=stockx_page4 value='\""+stockx_page3+\"'\>");
    out.println("</form>");

    out.println("</body>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("</html>");
con.close();

}
catch(Exception ex)
{
ex.printStackTrace();
}
}
}

```

ไฟล์ login.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;
import java.util.*;
import java.lang.*;

public class login extends HttpServlet
{
String values[];
Connection con = null;
Statement stmt = null;
ResultSet rs = null;

public void service(HttpServletRequest req,HttpServletResponse resp)
throws ServletException,java.io.IOException
{
connectdb();
addinfo(req,resp);
//con.close();
}

public void connectdb()
{
try
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database
con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+"."+
                                connect.portnumber+"."+connect.databaseSID
                                ,connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
System.out.println("error while initializing database connection"+ex.toString());
}
}

public void addinfo(HttpServletRequest req,HttpServletResponse resp)
{
try
{
resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

String codex_page4="";
values=req.getParameterValues("codex_page4");
if(values!=null)
{
codex_page4=values[0];
// out.println(" found!");
}

String stockx_page4="";
values=req.getParameterValues("stockx_page4");
if(values!=null)
{
stockx_page4=values[0];
//out.println(" found!");
}

/* out.println("your code you choose is");
out.println(codex_page4);
out.println(stockx_page4);*/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String view="";
values=req.getParameterValues("view");
if(values!=null)
{
    vjew=values[0];
    //out.println(view);
}

out.println("<html>");
out.println("<head><title>Please Login as Member</title></head>");

    out.println("<body background=\\'/image/41.gif\\'>");
        out.println("<center><img src=\\'http://'+connect.hostname+'/image/bar4.gif\\'></center>");
out.println("<br>");

out.println("<CENTER><FONT color=#000080 size=5>5 โปรดคลิกอื่นเข้าระบบ</FONT>");
out.println("<TABLE bgColor=#ffff00 border=2 width=\\'100%\\'>");
out.println("<TBODY>");
out.println("<TR><TD>");

out.println("<form method=\\'POST\\' action=\\'http://'+connect.hostname+'/servlet/page7\\'>");
out.println("<div align=\\'center\\'><center><table border=\\'0\\' width=\\'100%\\' height=\\'151\\'>");
out.println("<tr>");
out.println("<td width=\\'50%\\' height=\\'27\\'><div align=\\'right\\'><p><big><big><font color=\\'#000080\\'>
USERNAME      : </font></big></big></td>");
out.println("<td width=\\'50%\\' height=\\'27\\'><input type=\\'text\\' name=\\'username\\' size=\\'20\\'></td>");
out.println("</tr>");
out.println("<tr>");
out.println("<td width=\\'50%\\' height=\\'27\\'><div align=\\'right\\'><p><big><big><font color=\\'#000080\\'>
PASSWORD      : </font></big></big></td>");
out.println("<td width=\\'50%\\' height=\\'27\\'><input type=\\'password\\' name=\\'password\\' size=\\'20\\'></td>");
out.println("</tr>");
out.println("<tr>");
out.println("<td width=\\'50%\\' height=\\'25\\'><div align=\\'right\\'><p><input type=\\'submit\\' value=\\'Submit\\'
name=\\'B1\\'></td>");
out.println("<td width=\\'50%\\' height=\\'25\\'><input type=\\'reset\\' value=\\'Reset\\' name=\\'B2\\'></td>");
out.println("</tr>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("</table>");
out.println("</center></div>");
    out.println("<input type=hidden name=codex_page4 value='"+codex_page4+"'>");
    out.println("<input type=hidden name=stockx_page4 value='"+stockx_page4+"'>");
out.println("<input type=hidden name=view value='"+view+"'>");
out.println("</form>");

out.println("</TD></TR></TBODY></TABLE></CENTER>");
out.println("</body>");
out.println("</html>");

    con.close();
}
catch(Exception ex)
{
    ex.printStackTrace();
}
}
}

```

ไฟล์ page6.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;
import java.util.*;
import java.lang.*;

public class page6 extends HttpServlet
{
    String values[];
    Connection con = null;
    Statement stmt = null;
    ResultSet rs = null;

    public void service(HttpServletRequest req,HttpServletResponse resp)
    throws ServletException,java.io.IOException
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

connectdb();
addinfo(req,resp);
//con.close();
}

public void connectdb()
{
try
{
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database
con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+"."+
connect.portnumber+"."+connect.databaseSID
,connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
System.out.println("error while initializng database connection"+ex.toString());
}
}

public void addinfo(HttpServletRequest req,HttpServletResponse resp)
{
try
{
resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

String codex_page4="";
values=req.getParameterValues("codex_page4");
if(values!=null)
{
codex_page4=values[0];
// out.println("found!");
}

String stockx_page4="";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

values=req.getParameterValues("stockx_page4");
if(values!=null)
{
    stockx_page4=values[0];
    // out.println("found!");
}
/* out.println("your code you choose is");
out.println(codex_page4);
out.println(stockx_page4);*/

String view="";
values=req.getParameterValues("view");
if(values!=null)
{
    view=values[0];
    // out.println(view);
}

out.println("<html>");
out.println("<br>");
out.println("<head>");
out.println("<title>Please enter information for none member</title>");

out.println("</head>");
out.println("<body background='/image/41.gif'>");

out.println("<SCRIPT LANGUAGE='JavaScript'>");

out.println("function checkinput(CheckForm) {");
    out.println("if (empty(CheckForm.customer_name.value)) {");
    out.println(" var nofullname='\You must include your name.';");
    out.println(" alert(nofullname);");
    out.println("}");
    out.println("else if (empty(CheckForm.address.value)) {");
    out.println(" var noaddress='\You must include your address.';");
    out.println(" alert(noaddress)");
    out.println("}");
    out.println("else if (empty(CheckForm.postcode.value)) {");
    out.println(" var nopostcode='\You must include your postcode.';");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    out.println(" alert(nopostcode) ");
    out.println("{}");
    out.println("else if (empty(CheckFom.telephone.value) {}");
    out.println(" var nophone="\You must include your phone number.\");");
    out.println(" alert(nophone) ");
    out.println("{}");
    out.println(" else return true ");
    out.println("return false ");
    out.println("{}");

    out.println("function empty(inputStr) {}");
        out.println(" if (inputStr == \"\" || inputStr == null) {}");
        out.println("     return true");
        out.println(" }");
        out.println(" return false");
        out.println(" }");
    out.println("</SCRIPT>");
    //test=====
    String pagex="page7";
    /*
    if(enter all)
        {pagex=page5;}
    else
        {pagex=page7;}
    */
    //=====test

    out.println("<center><img src=\"http://"+connect.hostname+"/image/bar4.gif\"></center>");
        out.println("<br>");
    out.println("<CENTER><FONT color=#000080 size=5>5 โปรรครกรราชละยียค</FONT>");
    out.println("<TABLE bgColor=#ffffe0 border=2 width=\"100%\">");
    out.println("<TBODY>");
    out.println("<TR><TD>");

    out.println("<center>");
    out.println("<table border=0 cellspacing=2 cellpadding=2 width=100%>");

    out.println("<form name=ThisForm method=POST action=\""+http://" +connect.hostname+"/servlet/"+pagex+"\"
    onSubmit=\"return checkinput(document.ThisForm)\">");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<div align=\"center\">");
out.println("<tr><td align=right><img alt=\"Gift!\" width=18 height=17 src=\"/image/tiny-gift-click-box.gif\"></td><td
align=left></td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>ชื่อผู้ซื้อ</font></strong><br><font
face=Arial size=2>( ชื่อเต็ม ):</font></td><td><input type=\"text\" name=customer_name value=\"\" size=50
maxlength=50></td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>นามสกุลผู้ซื้อ</font></strong><br><font
face=Arial size=2>( ถิ่น ):</font></td><td><input type=\"text\" name=lastname value=\"\" size=50 maxlength=50>
</td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>ที่อยู่</font></strong><br><font face=Arial
size=2>( ในกาจัดตั้งสินค้า ):</font></td><td><input type=\"text\" name=address value=\"\" size=50 maxlength=100>
</td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>รหัสไปรษณีย์</font></strong></td>
<td><input type=\"text\" name=postcode value=\"\" size= 20></td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>เบอร์โทรศัพท์</font></strong></td>
<td><input type=\"text\" name=telephone value=\"\" size=20></td></tr>");

out.println("<tr><td align=right><strong><font face=Arial size=1>e-mail address</font></strong>&nbsp;&nbsp;&nbsp;
</td><td valign=top><input type=text name=email maxlength=30 size=30 ></td></tr>");

out.println("<tr><td>&nbsp;&nbsp;&nbsp;</td><td>");

out.println("<input type=image src=\"/image/continuc-orange.gif\" border=0 value=\"Continue\" name=submit
alt=\"Continue\" height=23 width=89>");

out.println("</td></tr></td></tr></table></center></div>");

/* out.println("FIRST NAME");
out.println("<input type=text name=customer_name>");
out.println("LAST NAME");
out.println("<input type=text name=lastname>");
out.println("<br>");
out.println("TELEPHONE");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<input type=text name=telephone>");
out.println("<br>");
out.println("EMAIL");
out.println("<input type=text name=email>");
out.println("<br>");
out.println("ADDRESS");
out.println("<input type=text name=address>");
out.println("POSTCODE");
out.println("<input type=text name=postcode>");
out.println("<br>");
out.println("<input type=submit name=submit value='submit'>");*/
out.println("<input type=hidden name=codex_page4 value='"+codex_page4+"'>");
out.println("<input type=hidden name=stockx_page4 value='"+stockx_page4+"'>");
out.println("<input type=hidden name=view value='"+view+"'>");

out.println("</form>");
out.println("</TD></TR></TBODY></TABLE></CENTER>");

out.println("</body>");
out.println("</html>");
con.close();
}
catch(Exception ex)
{
ex.printStackTrace();
}
}
}
}

```

ไฟล์ page7.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Vector;
import java.util.*;
import java.lang.*;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public class page7 extends HttpServlet
{
String values[];
Connection con = null;
Statement stmt = null;
ResultSet rs = null;

public void service(HttpServletRequest req,HttpServletResponse resp)
throws ServletException,java.io.IOException
{
connectdb();
addinfo(req,resp);
//con.close();
}

public void connectdb()
{
try
{
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database
con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+"."+
connect.portnumber+"."+connect.databaseSID
,connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
System.out.println("error while initializing database connection"+ex.toString());
}
}

public void addinfo(HttpServletRequest req,HttpServletResponse resp)
{
try
{
resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String customer_id="";

String codex_page4="";
values=req.getParameterValues("codex_page4");
if(values!=null)
{
    codex_page4=values[0];
    //out.println("found!");
}

String stockx_page4="";
values=req.getParameterValues("stockx_page4");
if(values!=null)
{
    stockx_page4=values[0];
    //out.println("found!");
}
/* out.println("your code you choose is");
out.println(codex_page4);
out.println(stockx_page4); */

String view="";
values=req.getParameterValues("view");
if(values!=null)
{
    view=values[0];
    // out.println(view);
}

String customer_name="";
values=req.getParameterValues("customer_name");
if(values!=null)
{
    customer_name=values[0];
}

String lastname="";
values=req.getParameterValues("lastname");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(values!=null)
{
    lastname=values[0];
}

String telephone="";
values=req.getParameterValues("telephone");
if(values!=null)
{
    telephone=values[0];
}

String email="";
values=req.getParameterValues("email");
if(values!=null)
{
    email=values[0];
}

String address="";
values=req.getParameterValues("address");
if(values!=null)
{
    address=values[0];
}

String postcode="";
values=req.getParameterValues("postcode");
if(values!=null)
{
    postcode=values[0];
}

String username="";
values=req.getParameterValues("username");
if(values!=null)
{
    username=values[0];
    // out.println(username);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

String password="";
values=req.getParameterValues("password");
if(values!=null)
{
    password=values[0];
    // out.println(password);
}

out.println("<html>");
out.println("<br>");
out.println("<head>");
out.println("<title>Confirm</title>");
// out.println("please enter information for none member");
out.println("</head>");
out.println("<body background= '/image/41.gif' >");
out.println("<center><img src= 'http://'+connect.hostname+' /image/bar4.gif' ></center>");
    out.println("<br>");
    out.println("<center><font color= '#000080' ><h3>6 ยืนยันการสั่งซื้อ</h3></font></center>");

if(view.compareTo("nonmember")==0)
{
    if(customer_name.compareTo("")==0)
    {
        out.println("PLEASE ENTRY YOUR NAME");
    }
    else
    {
        out.println("<b>"+ชื่อ :<font color= '#000080' >"+customer_name+"</font><b>");
        out.println("<br>");
        out.println("<b>"+นามสกุล :<font color= '#000080' >"+lastname+"</font><b>");
        out.println("<br>");
        out.println("<b>"+เบอร์โทรศัพท์ :<font color= '#000080' >"+telephone+"</font><b>");
        out.println("<br>");
        out.println("<b>"+EMAIL :<font color= '#000080' >"+email+"</font><b>");
        out.println("<br>");
        out.println("<b>"+ที่อยู่ที่จะจัดส่งสินค้าไปที่ :<font color= '#000080' >"+address+"</font><b>");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    out.println("<bi>");
    out.println("<b>"+รหัสไปรษณีย์;<font color=\"#000080\">"+postcode+"</font><b>");
    out.println("<br>");
}
}
if(view.compareTo("member")==0)
{
    String query="select customer_id,firstname,lastname,telephone,email,address,postcode from customer where
username='"+username+"' and password='"+password+"'";

    rs = stmt.executeQuery(query);
    rs.next();

    customer_id=rs.getString(1);
    customer_name=rs.getString(2);
    lastname=rs.getString(3);
    telephone=rs.getString(4);
    email=rs.getString(5);
    address=rs.getString(6);
    postcode=rs.getString(7);

    /*
if(rs==null)
{
    out.println("WRONG ENTRY");
}
else
*/
    out.println("<b>"+NAME           :<font color=\"#AA0000\">"+customer_name+"</font><b>");
    out.println("<br>");
    out.println("<b>"+LAST NAME      :<font color=\"#AA0000\">"+lastname+"</font><b>");
    out.println("<br>");
    out.println("<b>"+TELEPHONE      :<font color=\"#AA0000\">"+telephone+"</font><b>");
    out.println("<br>");
    out.println("<b>"+EMAIL           :<font color=\"#AA0000\">"+email+"</font><b>");
    out.println("<br>");
    out.println("<b>"+ADDRESS         :<font color=\"#AA0000\">"+address+"</font><b>");
    out.println("<br>");
    out.println("<b>"+POSTCODE       :<font color=\"#AA0000\">"+postcode+"</font><b>");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        out.println("<br>");
    }
    if((customer_name.compareTo("")==0)&&(view.compareTo("nonmember")==0))
    {
    }
    else {
        out.println("<form method=POST action=\"" + http://" + connect.hostname + "/servlet/page5\">");
        out.println("<input type=hidden name=customer_name value=\"" + customer_name + "\">");
        out.println("<input type=hidden name=lastname value=\"" + lastname + "\">");
        out.println("<input type=hidden name=telephone value=\"" + telephone + "\">");
        out.println("<input type=hidden name=email value=\"" + email + "\">");
        out.println("<input type=hidden name=address value=\"" + address + "\">");
        out.println("<input type=hidden name=postcode value=\"" + postcode + "\">");
        out.println("<input type=submit name=submit value=\"" >> ขึ้นชั้นใหม่ 1 ครั้ง <<\">");
        out.println("<input type=hidden name=codex_page4 value=\"" + codex_page4 + "\">");
        out.println("<input type=hidden name=stockx_page4 value=\"" + stockx_page4 + "\">");
        out.println("<input type=hidden name=customer_id value=\"" + customer_id + "\">");
        out.println("<input type=hidden name=view value=\"" + view + "\">");
        out.println("</form>");
    }
    out.println("</body>");
    out.println("</html>");
    con.close();
}
catch(Exception ex)
{
    ex.printStackTrace();
}
}
}

```

ไฟล์ page5.java

```

import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.lang.*;
import java.text.*;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public class page5 extends HttpServlet
{
String values[];
Connection con = null;
Statement stmt = null;
ResultSet rs = null;

public void service(HttpServletRequest req,HttpServletResponse resp)
throws ServletException,java.io.IOException
{
connectdb();
change_db(req,resp);
//con.close();
}

public void connectdb()
{
try
{
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
// Get a Connection to the database
con=DriverManager.getConnection("jdbc:oracle:thin:@"+connect.hostname+": "+
connect.portnumber+": "+connect.databaseSID+
".connect.username,connect.password);

stmt = con.createStatement();
}
catch(SQLException ex)
{
System.out.println("error while initializingg database connection"+ex.toString());
}
}

public void change_db(HttpServletRequest req,HttpServletResponse resp)
{
try
{
Vector vector_code =new Vector();
Vector vector_stockx =new Vector();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Vector vector_total=new Vector();

String customer_idx;
String username="member1";
String password="iammember1";
String invoice_id;
String invoice_date;
String sale_id;
int total_price=0;

resp.setContentType("text/html");
PrintWriter out = resp.getWriter();

resp.setHeader("Content-Language","en");
Locale locale=new Locale("en","");
DateFormat fmt=DateFormat.getDateInstance(DateFormat.LONG,DateFormat.LONG,locale);
fmt.setTimeZone(TimeZone.getDefault());
//out.println(fmt.format(new java.util.Date()));
String st_date=fmt.format(new java.util.Date());
//out.println(st_date);
StringTokenizer tk_date=new StringTokenizer(st_date);
String month=tk_date.nextToken();
String day=tk_date.nextToken();
    day=day.substring(0,2);
String year=tk_date.nextToken();
//out.println(day+"-"+month+"-"+year);
invoice_date=day+"-"+month+"-"+year;

String customer_id="";
values=req.getParameterValues("customer_id");
if(values!=null)
{
    customer_id=values[0];
    //out.println("customer id "+customer_id+"");
}

String view="";
values=req.getParameterValues("view");
if(values!=null)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
view=values[0];
//out.println(view);
}

String codex_page4="";
values=req.getParameterValues("codex_page4");
if(values!=null)
{
codex_page4=values[0];
//out.println("found!");
}

String stockx_page4="";
values=req.getParameterValues("stockx_page4");
if(values!=null)
{
stockx_page4=values[0];
//out.println("found!");
}

String customer_name="";
values=req.getParameterValues("customer name");
if(values!=null)
{
customer_name=values[0];
//out.println(customer_name);
}

String lastname="";
values=req.getParameterValues("lastname");
if(values!=null)
{
lastname=values[0];
//out.println(lastname);
}

String telephone="";
values=req.getParameterValues("telephone");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(values!=null)
{
    telephone=values[0];
    //out.println(telephone);
}

String email="";
values=req.getParameterValues("email");
if(values!=null)
{
    email=values[0];
    //out.println(email);
}

String address="";
values=req.getParameterValues("address");
if(values!=null)
{
    address=values[0];
    //out.println(address);
}

String postcode="";
values=req.getParameterValues("postcode");
if(values!=null)
{
    postcode=values[0];
    //out.println(postcode);
}

/* out.println("your code you choose is");
out.println(codex_page4);
out.println(stockx_page4); */

StringTokenizer tk=new StringTokenizer(codex_page4);
while(tk.hasMoreTokens())
{
    vector code.addElement(tk.nextToken());
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

StringTokenizer sk=new StringTokenizer(stockx_page4);
while(sk.hasMoreTokens())
{
    vector_stockx.addElement(sk.nextToken());
}

//add total price to vector
int i=0;
for(i=0;i<vector_code.size();i++)
{
    String query="select sp_price from spare_part where sp_code='"+vector_code.elementAt(i)+"'";
    rs = stmt.executeQuery(query);
    rs.next();
    int p1=Integer.parseInt((String)vector_stockx.elementAt(i));
    int p2=p1*rs.getInt(1);

    if(view.compareTo("member")==0)
    {
        p2=(int)p2*80/100;
    }

    total_price=total_price+p2; //total money in the invoice

    String dummy="" + p2;
    vector_total.addElement(dummy);
    //out.println(dummy);
}
stmt.close();
rs=null;
//out.println("<br>");

//update stock
update_stock(vector_code,vector_stockx);

//check if not member
stmt = con.createStatement();
if(view.compareTo("nonmember")==0)
{
    customer_idx=insert_customertable(customer_name,lastname,telephone,email,address,postcode);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        out.println("insert non member to customer table"+customer_idx);
    }
    else //if member
    {
        customer_idx=customer_id;
    }

    out.println("<br>");
    out.println(customer_idx);

    //insert data to invoice
    invoice_id=update_invoice(customer_idx,total_price,invoice_date);

    //insert data into sale
    sale_id=update_sale(invoice_id,vector_code,vector stockx,out);
    /*out.println("sale id:");
    out.println(sale_id);
    out.println("<br>");*/

    out.println("<html>");
    out.println("<head>");
    out.println("<title>test car query</title>");
    out.println("</head>");
    out.println("<body background='\"/image/41.gif\">");
    out.println("<center><img src='\"http://"+connect.hostname+"/image/bar5.gif\"></center>");
        out.println("<br>");

    out.println("<br>");
    out.println("<b>"+"INVOICE ID : "+invoice_id+"</b>");
    out.println("<br>");
    out.println("<b>"+"DATE :"+invoice_date+"</b>");
    out.println("<br>");

    stmt = con.createStatement();
    String query="select firstname from customer where customer_id="+customer_idx+"";
    rs = stmt.executeQuery(query);
    rs.next();
    out.println("<b>"+"NAME :"+rs.getString(1)+"</b>");
    rs=null;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

out.println("<table border width=100%>");
out.println("<tr><th>code</th><th>name</th><th>side</th><th>number</th><th>totalprice</th></tr>");

i=0;
for(i=0;i<vector_code.size();i++)
{
    query="select sp_name,sp_lr from spare_part where sp_code='"+(String)vector_code.elementAt(i)+"'";
    rs = stmt.executeQuery(query);
    rs.next();
out.println("<tr><td>"+(String)vector_code.elementAt(i)+"</td><td>"+rs.getString(1)+"</td><td>"+rs.getString(2)+"
</td><td>"+(String)vector_stockx.elementAt(i)+"</td><td>"+(String)vector_total.elementAt(i)+"</td></tr>");
}
out.println("<tr><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+ " "</td><td>"+total_price+"</td>
</tr>");
out.println("</table>");

out.println("<a href='http://"+connect.hostname+"/servlet/Sap/?lastpage=yes'>home</a>");
out.println("</body>");
out.println("</html>");
con.close();
}
catch(Exception ex)
{
ex.printStackTrace();
}
}

public String insert_customerTable(String cn,String ln,String tl,String em,String ad,String pc)
{
int customer_id=0;
String zero="";
try
{
String query="select MAX(CUSTOMER_ID) from CUSTOMER";
rs = stmt.executeQuery(query);
rs.next();
customer_id=rs.getInt(1);
customer_id++;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

stmt.close();
rs=null;

zero=getzero(4,customer_id);
stmt=con.createStatement();
query="INSERT INTO CUSTOMER
VALUES('"+zero+customer_id+"','1',NULL,NULL,"+cn+"','"+ln+"','"+tl+"','"+cm+"','"+ad+"','"+pc+"')";

stmt.executeUpdate(query);
con.commit();
stmt.close();
stmt=con.createStatement();
}
catch(Exception ex)
{
ex.printStackTrace();
}
String re_type=zero+customer_id;
return re_type;
}

public void update_stock(Vector code,Vector stockx)
{
try
{
int i=0;
int stock,bought;
String query;
for(i=0;i<code.size();i++)
{
stmt=con.createStatement();
//select the old stock
query="select sp_stock from spare_part where sp_code='"+(String)code.elementAt(i)+"'";
rs = stmt.executeQuery(query);
rs.next();
bought=Integer.parseInt((String)stockx.elementAt(i));
stock=rs.getInt(1)-bought;
stmt.close();
rs=null;
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//put new stock to database
stmt=con.createStatement();
query="UPDATE SPARE_PART SET sp_stock="+stock+"where sp_code="+(String)code.elementAt(i)+"";
stmt.executeUpdate(query);
stmt.close();
}
con.commit();
}
catch(Exception ex)
{
ex.printStackTrace();
}
}

public String update_invoice(String customer_id,int total_price,String invoice_date)
{
String st_invoice_id="";
String zero;
int in_invoice_id;
try
{
//get the max invoice id
stmt=con.createStatement();
String query="select MAX(INVOICE_ID) from INVOICE";
rs = stmt.executeQuery(query);
rs.next();
in_invoice_id=rs.getInt(1);
in_invoice_id++;
stmt.close();
rs=null;

//insert
zero=getzero(4,in_invoice_id);
st_invoice_id=zero+in_invoice_id;

stmt=con.createStatement();
query="INSERT INTO INVOICE
VALUES("+st_invoice_id+","+customer_id+","+invoice_date+","+total_price+")";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

stmt.executeUpdate(query);
con.commit();
stmt.close();
rs=null;
}
catch(Exception ex)
{
ex.printStackTrace();
}
return st_invoice_id;
}

public String update_sale(String invoice_id,Vector code,Vector item,PrintWriter out)
{String zero="";
String st_sale_id="";
int in_sale_id=0;
try
{
//out.println("xxxxxxxxxxxxxxxxxxxx");
stmt=con.createStatement();
String query="select MAX(SALE_ID) from SALE";
rs = stmt.executeQuery(query);
rs.next();
in_sale_id=rs.getInt(1);
stmt.close();
rs=null;
//out.println(in_sale id);
//out.println("yyyyyyyyyyyyyyyyyy");
//insert
stmt=con.createStatement();
int i=0;
for(i=0;i<code.size();i++)
{
in_sale_id++;
zero=getzero(6,in_sale_id);
//out.println(":"+zero+":");
st_sale_id=zero+in_sale_id;
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    query="INSERT INTO SALE VALUES('"+st_sale_id+"','"+invoice_id+"','"+(String)code.elementAt(i)+"','"+(String)
item.elementAt(i)+"");
    stmt.exccuteUpdate(query);

}

con.commit();
stmt.close();
rs=null;
}
catch(SQLException ex)
{
    System.out.println("error while initializingg database connection"+ex.toString());
}
return st_sale_id;
}

public String getzero(int its digit,int any_id)
{
    String zero="";
    String id="";
    int differ;

    id=id+any_id;
    differ=its digit-id.length();
    while(differ>0)
    {
        differ--;
        zero=zero+"0";
    }
    return zero;
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้