

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์

Flight Ticket Reservation via Telephone System



โดย นางสาวไพรัตน์ มิ่งเมือง
นางสาวภัทรภรณ์ เลิศวิทย์วรเทพ
นางสาวมาตา คุ้มทุกข์

ตรวจออกหนังสือ

เลขหมู่.....
เลขทะเบียน 61996
วัน,เดือน,ปี 25 ก.ค. 2549

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

๒๗



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อเวลาผ่านไปให้เข้าใจใช้ประโยชน์ด้านการทำ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์

Flight Ticket Reservation via Telephone System



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทบริหารการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์

Flight Ticket Reservation via Telephone System

ผู้จัดทำ

1. นางสาวไพรัตน์ มิ่งเมือง 44010353
2. นางสาวภัทรภรณ์ เลิศวิทย์วรเทพ 44010359
3. นางสาวมาตา กุ่มทุกข์ 44010376

.....
(ผศ. นภัทร สระเอี่ยม) อาจารย์ที่ปรึกษา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์
FLIGHT TICKET RESERVATION VIA
TELEPHONE SYSTEM

โดย	นางสาวไพรัตน์	มิ่งเมือง	44010353
	นางสาวภัทรภรณ์	เลิศวิทย์วรเทพ	44010359
	นางสาวมาตา	กุ่มทุกซ์	44010376

อาจารย์ที่ปรึกษา ผศ.นภัทร สระเอี่ยม

บทคัดย่อ

การให้บริการลูกค้าผ่านทางโทรศัพท์ เป็นการบริการอีกรูปแบบหนึ่งที่เป็นที่นิยม และสามารถทำงานได้อย่างมีประสิทธิภาพจึงทำให้นักธุรกิจต่างๆ หันมาให้ความสนใจนำไปประยุกต์ใช้งานกับธุรกิจที่ทำอยู่ ปริญญาบัตรนี้จึงทำการศึกษาและออกแบบระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์ เพื่อติดต่อระหว่างโทรศัพท์กับฐานข้อมูลที่มีอยู่ และเป็นแนวทางนำไปประยุกต์ใช้ในการทำธุรกิจต่างๆ เพื่อให้ผู้ใช้บริการสามารถเข้าถึงข้อมูลและบริการต่างๆ ได้อย่างรวดเร็วยิ่งขึ้น

ABSTRACT

Services via telephone, is popular and efficiently useful system. Thus, many businesses are interested in this system and apply it for commerce. This project studies and designs flight ticket reservation via telephone system to connect telephone and database. This will be guideline to apply in business for subscribers can reach the information and services fastly.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์ของปฏิญานิพนธ์	1
1.3 ขอบเขตของโครงการ	1
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักการ	
2.1 ระบบโทรศัพท์	3
2.1.1 ความรู้เบื้องต้นเกี่ยวกับโทรศัพท์	3
2.1.2 ระบบ DTMF (Dual Tone Multi Frequency dialing)	4
2.1.3 สัญญาณพื้นฐานของโทรศัพท์	5
2.1.3.1 สัญญาณให้หมุน (DT : Dial Tone)	5
2.1.3.2 สัญญาณสายไม่ว่าง (BT : Busy Tone)	5
2.1.3.3 สัญญาณเรียกกลับ (RBT : Ring Back Tone)	5
2.1.4 โครงข่ายโทรศัพท์สวิตช์สาธารณะ (PSTN: Public Switched Telephone Network)	5
2.1.5 ระบบบริการข้อมูลจากฐานข้อมูลเป็นเสียง (IVR : Interactive Voice Response)	6
2.2 โมเด็ม (Modem)	6
2.2.1 ส่วนประกอบของโมเด็ม	6
2.2.2 ประเภทของโมเด็ม	7
2.2.3 ข้อเปรียบเทียบระหว่างโมเด็มแบบติดตั้งภายใน และโมเด็มแบบติดตั้งภายนอก	8
2.2.4 ประโยชน์ของโมเด็ม	8
2.2.5 มาตรฐานของโมเด็ม	8
2.2.6 โมเด็มเสียง (Voice Modem)	
2.2.6.1 ลักษณะการใช้งานของโมเด็มเสียง	9
2.2.6.2 โมเด็มความเร็ว 56 Kbps	9
2.2.6.3 โมเด็ม ADSL (Asymmetric Digital Subscriber Line)	10
2.3 จาวา	10
2.3.1 Java Virtual Machine (JVM)	11
2.3.2 Java Runtime Environment (JRE)	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า	
2.3.3	การสร้างแอปพลิเคชันของจาวา	12
2.3.4	แบบจำลองเชิงวัตถุ (Object Model)	13
2.3.4.1	แนวคิดแบบ Object Model Technology (OMT)	13
2.3.5	ข้อดีของการเลือกใช้ภาษาจาวา	14
2.4	จาวาเทเลโฟนี่ API (Java Telephony API : JTAPI)	14
2.4.1	รูปแบบการพัฒนาจาวาเทเลโฟนี่ API (Configuration)	15
2.4.1.1	รูปแบบการพัฒนาเพื่อใช้งานแบบเน็ตเวิร์ค (Network Computer (NC) Configuration)	15
2.4.1.2	รูปแบบการพัฒนาเพื่อใช้งานบนเครื่องคอมพิวเตอร์ส่วนตัว (Desktop Computer Configuration)	16
2.4.2	โครงสร้างของแพคเกจ	16
2.4.2.1	แพคเกจคอร์ (Core package)	16
2.4.2.2	แพคเกจสแตนด์การ์ดเอ็กซ์เทนชัน (Standard extension package)	17
2.4.3	แบบจำลองการเรียก (Call model)	17
2.4.4	เมธอดในแพคเกจคอร์ (Core Package Methods)	19
2.4.4.1	เมธอด Call.connect()	19
2.4.4.2	เมธอด TerminalConnection.answer()	19
2.4.4.3	เมธอด Connection.disconnect()	19
2.4.5	สถานะของ Connection Object	19
2.4.6	สถานะของ TerminalConnection Object	21
2.4.7	การวางเส้นทางการเรียกของโทรศัพท์ (Placing a Telephone Call)	22
2.4.8	แบบจำลองการรายงานเหตุการณ์ของจาวาเทเลโฟนี่ (The Java Telephony Event Delivery Model)	23
2.4.9	การระบุและการจัดหา Providers (Locating and obtaining Providers)	24
2.4.10	แพคเกจ javax.telephony.media	24
2.4.10.1	หลักการ Bind และ Release	26
2.4.10.2	อินเตอร์เฟซ MediaService	26
2.4.10.3	ECTF Resource interfaces	27
2.4.11	เจเนอริก JTAPI (Generic JTAPI : GJTAPI)	28
2.5	SQL Server 2000	29
2.5.1	ส่วนประกอบของตารางฐานข้อมูล	30
2.5.1.1	คอลัมน์	30
2.5.1.2	แถว	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.5.1.3 คีย์ (Key)	30
2.5.2 การจัดการและทำงานกับข้อมูลในฐานข้อมูล	31
2.5.2.1 ประโยคคำสั่ง SELECT	32
2.5.2.2 ประโยคย่อ FROM	34
2.5.2.3 ประโยคย่อ WHERE	34
2.5.2.4 ประโยคย่อ ORDER BY	35
2.5.2.5 ประโยคคำสั่ง INSERT	35
2.5.2.6 ประโยคคำสั่ง UPDATE	36
2.5.2.7 ประโยคคำสั่ง DELETE	37
2.5.2.8 การรวมข้อมูลในแต่ละคอลัมน์จากหลายตาราง (Merging Data from Multiple Table)	37
2.6 JDBC	38
2.6.1 การเขียนโปรแกรมติดต่อกับฐานข้อมูลโดยใช้ JDBC	39
2.6.1.1 สร้างการติดต่อกับฐานข้อมูล	39
2.6.1.2 ดำเนินการกับฐานข้อมูล	39
2.6.1.3 แสดงผลลัพธ์ให้กับผู้ใช้หรือการสอบถามข้อมูล	40
2.6.1.4 ยกเลิกการติดต่อกับฐานข้อมูล	41
บทที่ 3 การออกแบบ	
3.1 หลักการทำงานของระบบ	42
3.2 การออกแบบระบบฐานข้อมูล	48
บทที่ 4 การแสดงขั้นตอนการทำงานและผลการทดลองของระบบ	
4.1 ส่วนที่ใช้ในการติดต่อสื่อสารกับโทรศัพท์ของผู้ใช้บริการ และตรวจจับสัญญาณ DTMF	51
4.2 ส่วนที่ทำการติดต่อกับฐานข้อมูลและค้นหาข้อมูลที่ต้องการ	55
4.3 ส่วนที่ใช้สำหรับผู้ดูแลระบบซึ่งจะประกอบไปด้วยแอปพลิเคชัน ที่ใช้ในการรับตัว	63
บทที่ 5 สรุปและวิจารณ์	65
ภาคผนวก	66
หนังสือและเอกสารอ้างอิง	168

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 1.1 บล็อกไดอะแกรมการทำงาน	2
รูปที่ 2.1 การแบ่งกลุ่มความถี่ตามแฉวและหลัก	4
รูปที่ 2.2 สัญญาณผสมคลื่นความถี่	4
รูปที่ 2.3 โครงข่ายโทรศัพท์สวิตซ์สาธารณะ	5
รูปที่ 2.4 การเชื่อมต่อของโมเด็มความเร็ว 56 Kbps	7
รูปที่ 2.5 Java Virtual Machine	12
รูปที่ 2.6 ลำดับการทำงานของกรณีโปรแกรมจาวา	12
รูปที่ 2.7 รูปแบบการพัฒนาเพื่อใช้งานแบบเน็ตเวิร์ค	15
รูปที่ 2.8 รูปแบบการพัฒนาเพื่อใช้งานบนเครื่องคอมพิวเตอร์ส่วนตัว	16
รูปที่ 2.9 แพคเกจจอร์ และ แพคเกจสแตนด์ออลเอ็กซ์เทนชัน	16
รูปที่ 2.10 แบบจำลองการเรียกของ JTAPI	18
รูปที่ 2.11 การเปลี่ยนแปลงสถานะของ Connection	20
รูปที่ 2.12 การเปลี่ยนแปลงสถานะของ TerminalConnection	21
รูปที่ 2.13 ไดอะแกรมแสดงเหตุการณ์ที่เกิดขึ้นของแบบจำลองการเรียก	22
รูปที่ 2.14 ความสัมพันธ์ทางโครงสร้างของ JTAPI Media และ ECTF	25
รูปที่ 2.15 ส่วนประกอบของตารางฐานข้อมูล	30
รูปที่ 2.16 การติดต่อระหว่างโปรแกรมกับระบบฐานข้อมูล	38
รูปที่ 3.1 ขั้นตอนการทำงานของระบบจองตั๋วเครื่องบิน	42
รูปที่ 3.2 ฐานข้อมูลของระบบจองตั๋วเครื่องบิน	48
รูปที่ 4.1 ผลที่ได้จากการรันโปรแกรมและการรายงานผลการเชื่อมต่ออุปกรณ์	51
รูปที่ 4.2 การรายงานเหตุการณ์ต่างๆ ที่เกิดขึ้นจากการเรียกเข้า	52
รูปที่ 4.3 การรายงานเหตุการณ์การสิ้นสุดการเรียกเข้า	54
รูปที่ 4.4 ผลการตรวจจับสัญญาณ DTMF	55
รูปที่ 4.5 ผลในขั้นตอนการเลือกสถานีต้นทาง	56
รูปที่ 4.6 ผลในขั้นตอนการเลือกสถานีปลายทางเมื่อสถานีต้นทางเป็น Bangkok	56
รูปที่ 4.7 ผลในขั้นตอนการเลือกสถานีปลายทางเมื่อสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok	57
รูปที่ 4.8 ผลจากการเลือกสถานีต้นทางและสถานีปลายทางเมื่อสถานีต้นทางเป็น Bangkok	57
รูปที่ 4.9 ผลจากการเลือกสถานีต้นทางและสถานีปลายทางเมื่อสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok	58
รูปที่ 4.10 ผลที่ได้จากการเลือก วัน/เดือน/ปี ที่ต้องการออกเดินทาง	58
รูปที่ 4.11 ผลการแสดงเที่ยวบินและรายละเอียดเกี่ยวกับสายการบินต่างๆ	59
รูปที่ 4.12 ผลจากการเลือกเที่ยวบินและการเลือกรายการต่างๆ เมื่อเดินทางไปเที่ยวเดียว	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 4.13 ผลจากการเลือกเที่ยวบินและการเลือกรายการต่างๆ เมื่อเดินทางไป – กลับ	60
รูปที่ 4.14 ผลการแสดงผลข้อมูลและรายละเอียดของผู้ใช้บริการ	61
รูปที่ 4.15 ตารางข้อมูลเกี่ยวกับการจองตั๋วเครื่องบินของผู้ใช้บริการ	61
รูปที่ 4.16 ตารางข้อมูลเกี่ยวกับผู้ใช้บริการ	62
รูปที่ 4.17 ข้อมูลในตาราง Flight Instance	62
รูปที่ 4.18 ผู้ดูแลระบบทำการล็อกอินเพื่อใช้แอปพลิเคชัน	63
รูปที่ 4.19 หน้าต่างที่ใช้กรอกรายละเอียดข้อมูลในส่วน password และ ID ของลูกค้า ที่ต้องการรับตั๋ว	63
รูปที่ 4.20 ข้อมูลต่างๆ ของลูกค้า	64



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า	
ตารางที่ 2.1	ข้อเปรียบเทียบระหว่างโมเต็มแบบติดตั้งภายในโมเต็มแบบติดตั้งภายนอก	8
ตารางที่ 2.2	เครื่องมือของ Java SDK	11
ตารางที่ 2.3	ตัวอย่างข้อมูลจากตาราง author	31
ตารางที่ 2.4	ตัวอย่างข้อมูลจากตาราง Order Detail	31
ตารางที่ 2.5	ตัวอย่างข้อมูลจากตาราง Flight Instance	32
ตารางที่ 2.6	ตัวอย่างข้อมูลจากตาราง Flight schedule	32
ตารางที่ 2.7	ตัวอย่างข้อมูลจากตาราง Airport	32
ตารางที่ 2.8	ผลการคิวรีเมื่อระบุคอลัมน์ที่จะเรียกดูโดยใช้ประโยคคำสั่ง SELECT	33
ตารางที่ 2.9	ผลการคิวรีเมื่อต้องการเรียกดูคอลัมน์ทั้งหมดโดยใช้ประโยคคำสั่ง SELECT	33
ตารางที่ 2.10	ผลการคิวรีเมื่อเรียกดูข้อมูลแบบมีเงื่อนไขโดยใช้ประโยคคำสั่ง SELECT	34
ตารางที่ 2.11	โอเปอเรเตอร์ใน SQL	34
ตารางที่ 2.12	ผลการคิวรีเมื่อต้องการเรียกดูคอลัมน์โดยใช้ร่วมกับประโยคย่อย ORDER BY	35
ตารางที่ 2.13	การคิวรีเมื่อใช้คำสั่ง INSERT	35
ตารางที่ 2.14	ผลการคิวรีเมื่อใช้คำสั่ง UPDATE	36
ตารางที่ 2.15	ผลการคิวรีเมื่อใช้คำสั่ง UPDATE ร่วมกับประโยคย่อย WHERE	36
ตารางที่ 2.16	ผลการคิวรีเมื่อใช้คำสั่ง DELETE ร่วมกับประโยคย่อย WHERE	37
ตารางที่ 2.17	ผลการรวมข้อมูลระหว่างตาราง Flight Instance กับ Flight schedule	38
ตารางที่ 3.1	รายละเอียดของข้อมูลในฐานข้อมูล	49

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันเทคโนโลยีการติดต่อสื่อสารได้เข้ามามีบทบาทต่อชีวิตประจำวันมากขึ้น จะเห็นได้ว่ามีธุรกิจหลายรูปแบบที่มุ่งเน้น และมีความพยายามที่จะจัดหาอุปกรณ์และบริการเพื่อตอบสนองต่อความต้องการ พร้อมทั้งอำนวยความสะดวกให้กับผู้ใช้ได้ทันทั่วถึง ซึ่งการให้บริการลูกค้าผ่านทางโทรศัพท์เป็นการบริการอีกรูปแบบหนึ่งที่เป็นที่นิยม ดังนั้นปริญญาานิพนธ์นี้จึงทำการศึกษาและออกแบบระบบการจองตั๋วเครื่องบินผ่านทางโทรศัพท์ เพื่อติดต่อระหว่างโทรศัพท์กับฐานข้อมูลที่มีอยู่และเป็นแนวทางนำไปประยุกต์ใช้ในการทำธุรกิจต่างๆ เพื่อให้ผู้ใช้บริการสามารถเข้าถึงข้อมูลและบริการต่างๆ ได้อย่างรวดเร็วยิ่งขึ้น

1.2 วัตถุประสงค์ของปริญญาานิพนธ์

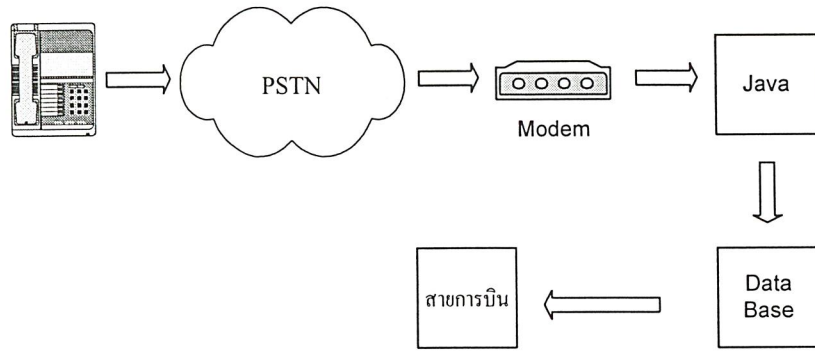
1. เพื่อศึกษาและสร้างระบบจำลองของการจองตั๋วเครื่องบินผ่านทางระบบโทรศัพท์ ซึ่งเป็นระบบตอบรับอัตโนมัติ
2. พัฒนาแอปพลิเคชัน (application) บนระบบการสื่อสาร
3. สามารถให้ผู้ใช้โทรศัพท์ทั่วไปเข้าใช้บริการของระบบโดยผ่านโครงข่ายโทรศัพท์ได้
4. สามารถนำไปประยุกต์ใช้งานกับการบริการทางผ่านโทรศัพท์ได้

1.3 ขอบเขตของโครงการ

1. สามารถนำสัญญาณจากโทรศัพท์มาใช้เป็นสื่อที่ใช้ในการติดต่อระหว่างผู้ใช้บริการกับระบบ
2. ทำการสร้างระบบที่สามารถให้ผู้ใช้โทรศัพท์ทั่วไปเข้ามาใช้บริการของระบบ โดยผ่านทางโครงข่ายโทรศัพท์
3. สร้างฐานข้อมูลขึ้นมาใช้ในระบบที่สร้างขึ้น
4. สร้างแอปพลิเคชันที่สามารถติดต่อกับฐานข้อมูลและโทรศัพท์ด้วยภาษาจาวา (Java)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบการจ้องตัวเครื่องบิน มีบล็อกไดอะแกรม (Block diagram) การทำงานในรูปที่ 1.1



รูปที่ 1.1 บล็อกไดอะแกรมการทำงาน

การเข้าใช้บริการระบบที่สร้างขึ้นนี้ สามารถทำได้โดยการกดเลขหมายที่ทำการให้บริการอยู่นั้นผ่านโครงข่ายโทรศัพท์เพื่อเข้าสู่ระบบ ระบบจะตอบรับสัญญาณที่เรียกเข้ามาผ่านทางโมเด็ม (Modem) จากนั้นเมื่อผู้ใช้บริการกดเลือกรายการที่ต้องการ สัญญาณที่ได้จะถูกตรวจจับเพื่อนำไปเป็นข้อมูลให้กับแอปพลิเคชันเพื่อเข้าสู่กระบวนการขั้นตอนต่อไป เช่น การค้นหาข้อมูลจากฐานข้อมูล โดยการทำงานดังกล่าวจะใช้จาวาเป็นเครื่องมือในการสร้างแอปพลิเคชันเพื่อให้ส่วนต่างๆ ของระบบติดต่อกันได้ และอาจจะทำการเชื่อมต่อฐานข้อมูลนี้ไปยังสาขาการบินที่เกี่ยวข้องกับระบบ

1.4 วิธีการดำเนินงาน

1. ศึกษาหลักการการทำงานของระบบโทรศัพท์และฮาร์ดแวร์ (Hardware) ที่สามารถใช้ร่วมกับระบบโทรศัพท์
2. ศึกษาการเขียนโปรแกรมแบบเชิงวัตถุ (Object Oriented Programming) ด้วยภาษาจาวาเพื่อเป็นพื้นฐานในการสร้างแอปพลิเคชัน
3. ศึกษา JTAPI (Java Telephony Application Programming Interface) ซึ่งเป็น API ที่ใช้งานในระบบโทรศัพท์ และ JDBC ซึ่งเป็น API ที่ใช้ติดต่อกับฐานข้อมูลเพื่อนำมาใช้ในโครงการนี้
4. ศึกษาข้อมูลเกี่ยวกับสาขาการบินในเรื่องต่างๆ เพื่อใช้เป็นข้อมูลสำหรับการออกแบบระบบ
5. การออกแบบระบบและเขียนแอปพลิเคชัน
6. ปรับปรุงระบบให้สามารถใช้งานได้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

ระบบจำลองการจ้องตัวเครื่องบินในปริศยานิพนธ์นี้ ได้ทำการสร้างขึ้นโดยประกอบด้วยส่วนหลักๆ ด้วยกัน 3 ส่วน คือ

- ส่วนที่ทำการติดต่อไปยังโครงข่ายโทรศัพท์
ในส่วนนี้จะใช้ JTAPI มาใช้ในการสร้างแอปพลิเคชันเพื่อติดต่อไปยังอุปกรณ์ที่เชื่อมต่อระหว่างเครื่องคอมพิวเตอร์และโครงข่ายโทรศัพท์ซึ่งก็คือ โมเด็ม
- ฐานข้อมูลของระบบ
ในส่วนนี้จะใช้ Microsoft SQL Server 2000 มาใช้ในการสร้างฐานข้อมูลบนเครื่องเซิร์ฟเวอร์ (server)
- ส่วนที่ทำการติดต่อไปยังฐานข้อมูล
ในส่วนนี้ได้นำ JDBC (Java Data Base Connection) มาใช้ในแอปพลิเคชัน เพื่อให้แอปพลิเคชันที่สร้างขึ้นมานั้นสามารถจัดการกับฐานข้อมูลที่มีอยู่ เช่น การดึงข้อมูลจากฐานข้อมูลมาใช้งานในโปรแกรมหรือการนำค่าที่ได้จากแอปพลิเคชัน ไปเก็บไว้ในฐานข้อมูล โดยเนื้อหาและรายละเอียดของส่วนต่างๆ นี้ จะกล่าวในส่วนต่อไป

2.1 ระบบโทรศัพท์

2.1.1 ความรู้เบื้องต้นเกี่ยวกับโทรศัพท์

ในปัจจุบันนี้การสื่อสารได้เข้ามามีบทบาทเป็นอย่างมากในชีวิตประจำวันของเรา เรียกได้ว่าต้องมีการติดต่อสื่อสารกันอยู่ตลอดเวลาที่ทำได้ และระบบโทรศัพท์ก็จัดว่าเป็นระบบสื่อสารที่ใกล้ตัวเรามาก โทรศัพท์ที่ใช้กันโดยทั่วๆ ไปมีอยู่ด้วยกัน 2 แบบ คือ แบบกดปุ่มและแบบระบบหมุน หน้าที่ของทั้ง 2 ระบบนี้จะเหมือนๆ กัน จะต่างกันตรงที่แบบกดปุ่มจะส่งสัญญาณออกไปเป็นความถี่ที่แตกต่าง ส่วนแบบระบบหมุนจะส่งสัญญาณเป็นจำนวนพัลส์ (pulse) หน้าที่หลักๆ ของทั้งสองแบบนี้จะเหมือนกันสามารถสรุปได้ดังนี้

- ระบบโทรศัพท์จะรับรู้ว่ามีผู้ต้องการใช้โทรศัพท์เมื่อมีการยกหูโทรศัพท์
- ระบบโทรศัพท์จะส่งสัญญาณที่เรียกว่า “สัญญาณหมุน” (dial tone) เป็นการบอกว่าพร้อมที่จะให้ทำการกดหรือหมุนหมายเลขที่จะติดต่อได้ ซึ่งก็คือเสียงที่ได้ยินเมื่อเวลายกหู
- ระบบโทรศัพท์จะทำหน้าที่ส่งรหัสหมายเลขที่ผู้เรียกต้องการจะติดต่อด้วยไปยังชุมสายที่ควบคุม
- ระบบโทรศัพท์จะส่งสัญญาณบอกผู้เรียกว่าหมายเลขที่ต้องการติดต่อดูว่างหรือไม่ ถ้าว่างก็จะส่งสัญญาณตอบกลับ (ring back)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบโทรศัพท์สามารถเปลี่ยนรูปพลังงานเสียงเป็นสัญญาณไฟฟ้าและเปลี่ยนสัญญาณไฟฟ้ากลับมาเป็นพลังงานเสียง
- ระบบโทรศัพท์จะปรับระดับแรงดันอย่างอัตโนมัติในกรณีที่เกิดการเปลี่ยนแปลงของแรงดันขึ้น
- ระบบโทรศัพท์จะส่งสัญญาณไปยังชุมสายเพื่อแจ้งให้ทราบว่าสิ้นสุดการใช้งานแล้ว และให้ชุมสายเลิกทำการติดต่อกับอีกฝ่ายหนึ่งได้

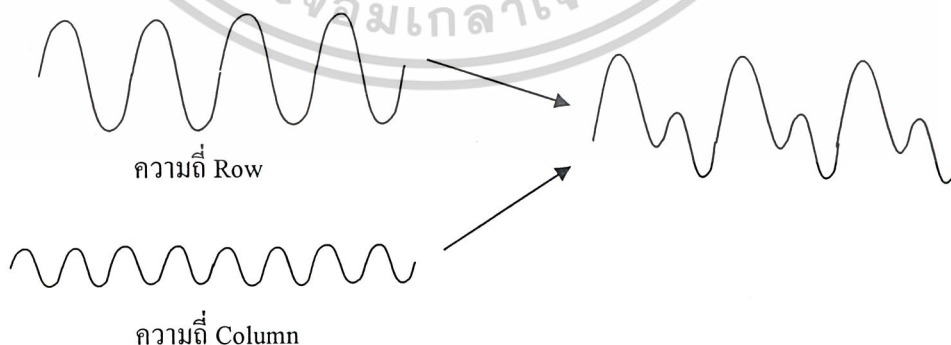
2.1.2 ระบบ DTMF (Dual Tone Multi Frequency dialing)

เครื่องโทรศัพท์ที่มีหน้าปัดแบบปุ่มกดและใช้กรรมวิธีของ DTMF ในการส่งหมายเลขโทรศัพท์ สัญญาณที่ได้จากการกดปุ่มบนแป้นโทรศัพท์ปุ่มใดปุ่มหนึ่ง จะเป็นสัญญาณซึ่งประกอบด้วยโทนเสียง 2 ความถี่ผสมกัน โดยทั่วไปจะมี 12 ปุ่ม แบ่งเป็น 4 แถว และ 3 คอลัมน์ บางรุ่นอาจจะมี 16 ปุ่ม โดยเพิ่มคอลัมน์ที่ 4 ดังรูปที่ 2.1

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

รูปที่ 2.1 การแบ่งกลุ่มความถี่ตามแถวและหลัก

จากรูปที่ 2.1 หากกดหมายเลข 1 ซึ่งตรงกับ แถวที่ 1 หลักที่ 1 วงจรภายในเครื่องจะทำการเข้ารหัสข้อมูลที่ได้แล้วส่งให้วงจรผลิตสัญญาณผสม 2 ความถี่ระหว่างความถี่ 697 Hz กับความถี่ 1209 Hz ออกไป หรือในกรณีหมายเลข 8 ก็จะประกอบด้วยความถี่ในแถวที่ 3 คือ 852 Hz กับความถี่ในหลักที่ 2 คือ 1336 Hz ผสมออกไปยังคู่สายโทรศัพท์ ซึ่งถ้าเราจับสัญญาณที่ส่งออกไปจะได้ดังรูปที่ 2.2



รูปที่ 2.2 สัญญาณผสมคลื่นความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 สัญญาณพื้นฐานของโทรศัพท์

2.1.3.1 สัญญาณให้หมุน (DT : Dial Tone)

เป็นสัญญาณต่อเนื่องความถี่ประมาณ 400 Hz ใช้เพื่อแสดงให้ผู้เรียกทราบว่า “พร้อมรับหมายเลขโทรศัพท์ได้”

2.1.3.2 สัญญาณสายไม่ว่าง (BT : Busy Tone)

เป็นสัญญาณ 400 Hz ดังเป็นช่วงเวลาประมาณ 0.25 วินาที เงียบเป็นช่วงเวลา 0.25 วินาที ใช้เพื่อเตือนให้ผู้เรียกทราบว่า “สายไม่ว่าง” ผู้เรียกควรวางหูสักระยะแล้วค่อยโทรใหม่

2.1.3.3 สัญญาณเรียกกลับ (RBT : Ring Back Tone)

เป็นสัญญาณ 400 Hz ดังเป็นช่วงเวลาประมาณ 1 วินาที เงียบประมาณ 3 วินาที เป็นสัญญาณที่ชุมสายแจ้งให้ผู้รับทราบว่า “มีคนโทรเข้ามา” ให้ไปรับสายได้โดยจะส่งเป็นเสียงกริ่งให้ได้ยินประมาณ 15 ครั้ง

2.14 โครงข่ายโทรศัพท์สวิตซ์สาธารณะ (PSTN : Public Switched Telephone Network)

โครงข่ายโทรศัพท์สวิตซ์สาธารณะ คือ ระบบชุมสายโทรศัพท์พื้นฐานที่ใช้กันโดยทั่วไปในปัจจุบันมีระบบการทำงานเป็นแบบสวิตซ์ (switching) ระบบ PSTN ในครั้งเริ่มต้นมีการเชื่อมต่อระหว่างจุดเป็นแบบอะนาล็อก (analog) จากนั้นจึงค่อยๆ เปลี่ยนบางส่วนมาเป็นระบบดิจิทัล (digital) ซึ่งในปัจจุบันการเชื่อมต่อระหว่างชุมสายในเขตกรุงเทพฯ และจังหวัดใหญ่ๆ ของประเทศไทยก็เปลี่ยนมาใช้ระบบดิจิทัลกันมาก จะเห็นได้จากผู้ให้บริการอินเทอร์เน็ตสามารถให้บริการที่ความเร็ว 56 kbps ได้ แต่ในส่วนที่เชื่อมโยงระหว่างชุมสายท้องถิ่นและผู้ใช้ยังคงเป็นสายทองแดงอยู่



รูปที่ 2.3 โครงข่ายโทรศัพท์สวิตซ์สาธารณะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 ระบบบริการข้อมูลจากฐานข้อมูลเป็นเสียง (IVR : Interactive Voice Response)

ระบบนี้จะทำงานโดยอ่านข้อมูลที่เป็นตัวเลขหรือข้อความ ในฐานข้อมูลออกมาเป็นเสียงผ่านทางโทรศัพท์โดยเชื่อมระบบโทรศัพท์เข้ากับฐานข้อมูล ผู้โทรสามารถรับฟังข้อมูลและทำรายการต่างๆ ที่ต้องมีการเปลี่ยนแปลงข้อมูลในฐานข้อมูลด้วยการกดปุ่มบนเป็นโทรศัพท์ เช่น การสั่งจองสินค้า การตรวจสอบยอดและโอนเงินระหว่างบัญชี ฯลฯ

- การเปลี่ยนแปลงของข้อความเสียงที่รายงานต่อผู้โทรจะเปลี่ยนไปตามความเคลื่อนไหวของข้อมูลในฐานข้อมูลโดยอัตโนมัติ
- ผู้โทรสามารถเปลี่ยนข้อมูลในฐานข้อมูลของระบบผ่านทางโทรศัพท์ได้
- เจ้าของระบบสามารถกำหนดเงื่อนไขให้กรหัสผ่าน ก่อนการเข้าฟังข้อมูลในฐานข้อมูลได้
- สามารถเชื่อมต่อกับโปรแกรมระบบฐานข้อมูลที่มีอยู่แล้ว เช่น dBase, FoxBase, FoxPro, Access, Paradox, Microsoft SQL Server, Sybase, Informix, Oracle, Excel และ Text File
- รายงานสามารถเชื่อมต่อกับฐานข้อมูล ทั้งแบบตัวเลขและข้อความ
- สามารถกำหนดช่วงเวลาในการให้บริการข้อมูลได้
- รองรับข้อมูล ได้ทั้งภาษาไทยและภาษาอังกฤษ
- สดใสหรือบริการสายด่วน สำหรับให้ข้อมูลสินค้าและบริการ
- ศูนย์บริการลูกค้าสัมพันธ์
- ศูนย์บริการข้อมูลเฉพาะสำหรับสมาชิก ลูกค้า หรือตัวแทนจำหน่าย
- บริการธนาคารทางโทรศัพท์

2.2 โมเด็ม

อุปกรณ์ที่รองรับในทางระบบโทรศัพท์มีอยู่ด้วยกันหลายประเภท การเลือกใช้อุปกรณ์ในระบบโทรศัพท์นั้นต้องพิจารณาให้เหมาะสมกับการใช้งาน ดังนั้นในที่นี้จะกล่าวถึงเฉพาะอุปกรณ์ที่เกี่ยวข้องกับปฏิญาณพันธ์เท่านั้นซึ่งก็คือ โมเด็ม

โมเด็ม ย่อมาจากคำว่า “Modulation/Demodulation” เป็นอุปกรณ์ที่ทำหน้าที่แปลงสัญญาณอนาล็อก (analog signal) ให้เป็นสัญญาณดิจิทัล (digital signal) ซึ่งการทำงานของโมเด็มภายในจะส่งสัญญาณไปมาระหว่างเครื่องรับสัญญาณและเครื่องส่งสัญญาณแต่สัญญาณที่เห็นนั้นจะเป็น 0 และ 1

2.2.1 ส่วนประกอบของโมเด็ม

เนื่องจากขั้วสายโทรศัพท์ได้ออกแบบมาเพื่อใช้กับการส่งสัญญาณเสียง หรือสัญญาณอนาล็อก ดังนั้น สัญญาณดิจิทัลจึงไม่สามารถใช้กับขั้วสายโทรศัพท์โดยตรงได้ จึงต้องใช้โมเด็มเป็นตัวเปลี่ยนสัญญาณดิจิทัลให้เป็นสัญญาณอนาล็อกก่อนที่จะส่งไปตามขั้วสายโทรศัพท์ได้

ดังนั้น โมเด็มจึงประกอบด้วยส่วนที่สำคัญ 3 ส่วน คือ

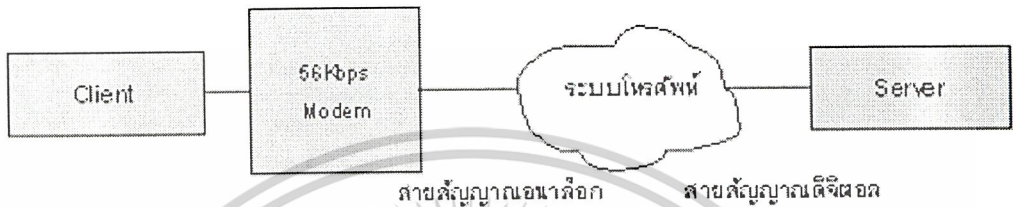
1. ส่วนที่ทำหน้าที่เปลี่ยนสัญญาณดิจิทัลที่รับเข้ามาจาก Data Terminal Equipment (DTE) ให้กลายเป็นสัญญาณอนาล็อกเพื่อส่งออกไปตามขั้วสายโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนที่ทำการเปลี่ยนสัญญาณอนาล็อกที่รับเข้ามาทางสายโทรศัพท์ ให้กลับไปเป็นสัญญาณดิจิทัลเพื่อส่งต่อไปยัง DTE ต่อไป

3. ส่วนควบคุมการทำงานและดิจิทัลอินเตอร์เฟซ (Digital Interface)

โมเด็มจะสร้างสัญญาณอนาล็อกส่งไปยังชุมสายโทรศัพท์ จากนั้นสัญญาณอนาล็อกนี้จะถูกแปลงไปเป็นสัญญาณดิจิทัลแล้วส่งต่อไปยังเครื่องเซิร์ฟเวอร์ของผู้ให้บริการอินเทอร์เน็ต เพื่อเข้าไปใช้บริการต่อไปดังรูปที่ 2.4



รูปที่ 2.4 การเชื่อมต่อของ โมเด็มความเร็ว 56 Kbps

2.2.2 ประเภทของโมเด็ม

การแบ่งประเภทของโมเด็มตามรูปร่างและการติดตั้งใช้งานนั้นสามารถแบ่งออกได้ 2 ประเภท

ดังนี้

- โมเด็มแบบติดตั้งภายใน (Internal Modem) จะมีลักษณะเป็นแผงวงจรอิเล็กทรอนิกส์นำมาติดตั้งไว้ภายในตัวเครื่องคอมพิวเตอร์ โดยรูปร่างจะแตกต่างกันตามทีผู้ผลิตจะออกแบบมาสำหรับคอมพิวเตอร์ชนิดนั้นๆ โดยที่โมเด็มชนิดนี้จะใช้ไฟฟ้าจากสล็อต (slot) บนเมนบอร์ด (mainboard) ทำให้เราไม่ต้องต่อจากไฟหม้อแปลงแยกต่างหากจากภายนอก มีโมเด็มบางประเภทที่ทำออกมาเป็นแบบการ์ดเสียบในเครื่อง โดยตัดฟังก์ชัน (function) การควบคุมออกไปทำให้เหลือแต่ตัวแปลงสัญญาณแล้วจะใช้ซีพียูทำงานแทน มักจะเรียกกันทั่วไปว่า WinModem หรือ Modem Riser โมเด็มประเภทนี้มีข้อดีคือจะมีราคาถูกแต่จะต้องการซีพียูที่มีความเร็วสูงๆ จึงจะใช้งานได้ ข้อเสียคือจะต้องใช้ซีพียูในการทำงานด้วยส่วนหนึ่ง จึงเป็นการไปถ่วงการทำงานของซีพียู
- โมเด็มแบบติดตั้งภายนอก (External Modem) จะมีลักษณะเป็นกล่องสี่เหลี่ยมแบนๆ ภายในมีวงจรโมเด็ม ไฟแสดงสถานะและลำโพงเนื่องจากการเป็นการทำงานจึงต้องมีแอดAPTER (adapter) เพื่อแปลงสัญญาณ และจะมีสายต่อแบบ 25 ขา (DB25) เอาไว้ใช้เชื่อมต่อผ่านทางพอร์ต (port) อนุกรม RS - 232C ดังนั้นเครื่องคอมพิวเตอร์หรืออุปกรณ์ที่ใช้กับโมเด็มแบบติดตั้งภายนอกจะมีพอร์ตอนุกรม (serial port) RS - 232C ด้วยเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 ข้อเปรียบเทียบระหว่างโมเด็มแบบติดตั้งภายในและโมเด็มแบบติดตั้งภายนอก

เป็นดังต่อไปนี้คือ

โมเด็มแบบติดตั้งภายใน	โมเด็มแบบติดตั้งภายนอก
1. ราคาถูก	1. ราคาแพง
2. ติดตั้งยาก ต้องเปิดฝาเครื่องเพื่อติดตั้ง	2. ติดตั้งโดยใช้สายเคเบิลต่อผ่านพอร์ตอนุกรม
3. ไม่เปลืองพอร์ตอนุกรมของเครื่องคอมพิวเตอร์	3. จะเปลืองพอร์ตอนุกรมเพราะใช้ในการต่อกับโมเด็ม
4. เคลื่อนย้ายไปใช้กับคอมพิวเตอร์เครื่องอื่นได้ลำบาก	4. เคลื่อนย้ายไปใช้กับคอมพิวเตอร์เครื่องอื่นได้ง่าย
5. ไม่ต้องมีการต่อสายไฟหรือแหล่งจ่ายไฟเพิ่มเติม	5. ต้องมีแหล่งจ่ายไฟและต่อสายไฟต่างหาก
6. ต้องการ ซีพียู ความเร็วสูงหรือ MMX ขึ้นไป	6. ใช้กับ ซีพียู รุ่นเก่า ๆ ได้
7. พบปัญหาต่างๆ ได้บ่อย เช่นสายหลุดง่าย	7. ไม่ค่อยมีปัญหาในการใช้งาน

ตารางที่ 2.1 ข้อเปรียบเทียบระหว่างโมเด็มแบบติดตั้งภายในโมเด็มและแบบติดตั้งภายนอก

2.2.4 ประโยชน์ของโมเด็ม

- หมุนโทรศัพท์และตอบรับโดยอัตโนมัติ
- การเลือกระบบการหมุนของระบบโทรศัพท์แบบทัช โทน (Touch Tone) หรือพัลส์
- หมุนทวนหมายเลขเดิมได้เมื่อติดต่อกแล้วสายไม่ว่าง
- การหยุดหมุนโทรศัพท์ชั่วคราว
- เปลี่ยนการติดต่อโทรศัพท์ให้ติดต่อแบบหมุนหรือแบบกดปุ่มก็ได้
- ใช้กับระบบโทรศัพท์ชุมสาย (PBX หรือ PABX) ทางไกลและการติดต่อระหว่างประเทศก็ได้
- รับส่งโปรแกรม ข่าวสาร ข้อมูลต่างๆ ผ่านทางสายโทรศัพท์ธรรมดาได้
- มีการเลือกปรับค่าพารามิเตอร์ที่ใช้ในการสื่อสารข้อมูล
- มีระบบบริการความปลอดภัยด้วยรหัสผ่าน

2.2.5 มาตรฐานของโมเด็ม

เนื่องจากในปัจจุบันได้มีการผลิตโมเด็มจากหลายบริษัทดังนั้นจึงต้องมีหน่วยงานที่ทำกาหนดมาตรฐานของโมเด็มขึ้น ซึ่งมาตรฐานนี้จะกำหนดโดยสมาคม ITU (International Telecommunications Union) และสำหรับโมเด็มที่มีจำหน่ายในประเทศไทยในปัจจุบันนี้ จะใช้มาตรฐาน V.92 ซึ่งเป็นมาตรฐานสากลของการสื่อสารข้อมูลในระดับความเร็ว 56 Kbps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.6 โมเด็มเสียง (Voice Modem)

โมเด็มชนิดนี้มีความสามารถในการมอดูเลตสัญญาณข้อมูลในลักษณะรูปแบบเสียงไว้ด้วย ซึ่งเรียกว่า โมเด็มเสียง โดยทำการเพิ่มวงจรแปลงข้อมูลเสียงทั้งจากสัญญาณอนาล็อกเป็นดิจิทัล และจากสัญญาณดิจิทัลเป็นอนาล็อก

โมเด็มทั่วไปที่ใช้งานกันอยู่นั้นเรียกว่า Data/Fax คือ ส่งได้เฉพาะ Data และ Fax แต่มีโมเด็มอีกประเภทหนึ่งที่เพิ่มคุณสมบัติการประมวลผลเสียงได้ด้วย เรียกว่า Data/Fax/Voice โดยผู้ผลิตชิป (ship) สำหรับโมเด็มหลายๆ รายได้ผลิตชิปสำหรับโมเด็มเสียงออกมา อาทิเช่น Rockwell, Cirrus Logic, Lucent โดยหลักการแล้วโมเด็มเสียงจะเพิ่มในส่วนของตัวแปลงสัญญาณเสียงพูด ซึ่งเป็นสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัลที่ประมวลผลได้ ทำให้ต้องมีภาคแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ADC (Analog to Digital Converter) ด้วย เช่น แปลงเสียงพูดที่ได้รับจากหูฟังหรือไมโครโฟนเป็นสัญญาณดิจิทัลและส่งไปยังผู้รับอีกด้านหนึ่ง หรือทำการแปลงเสียงที่ได้รับจากผู้ติดต่อเข้ามาซึ่งเป็นเสียงแบบอนาล็อกให้เป็นสัญญาณดิจิทัลเพื่อเก็บไว้ในรูปของไฟล์คอมพิวเตอร์อีกที

2.2.6.1 ลักษณะการใช้งานของโมเด็มเสียง

- เป็นโมเด็มที่สามารถทำงานเป็นเครื่องรับ โทรศัพท์ (Answering Machine) หรือระบบตู้จดหมาย แบบเสียงหรือทำงานเป็น Speakerphone
- ทำงานแบบ Full Duplex Speakerphone คือ สามารถส่งข้อมูลตอบโต้ได้สองทิศทาง
- โมเด็มที่สามารถส่งข้อมูลหรือเสียงได้พร้อมกัน

การใช้ประโยชน์จากโมเด็มเสียง คือ การนำมาทำเป็นระบบตอบรับ โทรศัพท์อัตโนมัติโดยเมื่อมีโทรศัพท์เข้ามา แต่ไม่มีผู้รับสายจากนี้ยังใช้งาน Fax Back ได้อีกด้วยแต่ต้องมีซอฟต์แวร์ควบคุมการทำงานอีกที ปัจจุบันมีผู้ผลิตซอฟต์แวร์ที่ทำงานร่วมกับ Fax/data/Voice Modem หลายรายด้วยกันที่เป็นที่รู้จักกันดีได้แก่ Supper Voice, Cheyenne Bitware โดยซอฟต์แวร์ดังกล่าวจะประมวลผลได้ทั้ง Fax/data/Voice มีกล่องรับจดหมายประเภท Voice Mail มีระบบ Fax Back และทำงานร่วมกับ TAPI

2.2.6.2 โมเด็มความเร็ว 56 Kbps

ในปัจจุบันโมเด็มมีความเร็วถึง 56 Kbps ซึ่งสามารถใช้กับสายโทรศัพท์เดิมที่มีอยู่ และทำงานเหมือนกับโมเด็มเครื่องเก่าทุกอย่างเพียงแต่ว่าเร็วขึ้น โมเด็ม 56 Kbps นี้ต้องใช้การเชื่อมต่อแบบกึ่งดิจิทัลเพื่อให้ได้ความเร็วสูง ข้อมูลที่อยู่ในเครื่องคอมพิวเตอร์อยู่ในรูปแบบดิจิทัล นั่นคือ 0 หรือ 1 สายโทรศัพท์แบบปกติรับส่งข้อมูลแบบอนาล็อก ซึ่งเป็นปัจจัยหนึ่งที่จำกัดความเร็วในการรับส่งข้อมูล เนื่องจากการรบกวนที่เกิดขึ้นจากสายโทรศัพท์เอง ดังนั้นในทุกวันนี้สายโทรศัพท์จะเป็นดิจิทัลมากขึ้น โดยเฉพาะอย่างยิ่งส่วนที่ต่อกับชุมสายโทรศัพท์และบริษัทที่ให้บริการอินเทอร์เน็ต แต่สายโทรศัพท์ดิจิทัลก็ยังคงมีการรบกวนเกิดขึ้นแต่น้อยกว่าสายโทรศัพท์แบบอนาล็อก หลายบริษัทที่ให้บริการอินเทอร์เน็ตจะยอมให้มีการใช้โมเด็ม 56 Kbps ได้ และจะต้องมีการต่อสายโทรศัพท์แบบดิจิทัลไปยังชุมสายโทรศัพท์ ดังนั้นหากต้องการใช้โมเด็ม 56 Kbps จะต้องมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โมเด็มที่มีความเร็ว 56 Kbps
- บริษัทที่ให้บริการอินเทอร์เน็ตที่ขอมให้ใช้โมเด็ม 56 Kbps ซึ่งบริษัทที่ให้บริการในเมืองไทยขอมให้ใช้โมเด็ม 56 Kbps ได้ทั้งหมด
- การใช้สายโทรศัพท์ที่สามารถใช้กับโมเด็ม 56 Kbps เป็นปัญหาใหญ่พอสมควร หากไม่มีสายโทรศัพท์ที่สามารถรองรับความเร็วขนาด 56 Kbps ได้แล้วความเร็วในการรับส่งข้อมูลจะเท่ากับ 28.8 Kbps หรือ 33.6 Kbps เท่านั้น

นอกจากด้านความเร็วแล้วโมเด็ม 56 Kbps มีข้อดีเหนือกว่าโมเด็ม 28.8 Kbps หรือ 33.6 Kbps คือโมเด็ม 56 Kbps จะเป็นที่แพร่หลายมากกว่าคู่แข่งที่สำคัญของโมเด็ม 56 Kbps คือการต่อกับอินเทอร์เน็ตที่มีความเร็วสูงกว่าอย่างเช่น ISDN, Cable Modems และ xDSLs (ADSL, SDSL อื่นๆ) เทคโนโลยีเหล่านี้ให้ความเร็วในการรับส่งข้อมูลสูงกว่าโมเด็ม 56 Kbps แต่การใช้โมเด็ม 56 Kbps มีข้อดีเหนือกว่าเทคโนโลยีเหล่านี้ตรงที่ยังคงสามารถใช้กับสายโทรศัพท์ได้โดยไม่ต้องเสียค่าใช้จ่ายในการติดตั้งการเชื่อมต่อกับอินเทอร์เน็ตเพิ่มเติม

ส่วนข้อเสียของโมเด็ม 56 Kbps ก็คือ ความเร็วในการดาวน์โหลด (download) (การรับข้อมูลจากเครื่องคอมพิวเตอร์เครื่องอื่น ยกตัวอย่างเช่น การดูเว็บเพจหรือการรับจดหมายอิเล็กทรอนิกส์) สูงกว่าการอัปโหลด (upload) (การส่งข้อมูลไปยังเครื่องคอมพิวเตอร์เครื่องอื่น เช่น การส่งจดหมายอิเล็กทรอนิกส์หรือการส่งไฟล์ไปยังเว็บเพจ) คือความเร็วในการดาวน์โหลดนั้นจะสามารถใช้ความเร็วได้ตามความเร็วของโมเด็มนั่นคือ 56 Kbps แต่สำหรับการอัปโหลดจะสามารถใช้ความเร็วได้เพียง 28.8 Kbps หรือ 33.6 Kbps เท่านั้น

2.2.6.3 โมเด็ม ADSL (Asymmetric Digital Subscriber Line)

เป็นเทคโนโลยีโมเด็มอย่างหนึ่งของ DSL ที่ใช้คู่สายทองแดง โดยในขณะที่มีการรับส่งข้อมูลความเร็วสูงนั้นจะใช้วิธีส่งสัญญาณในย่านความถี่สูงกว่าการใช้งานโทรศัพท์ทั่วไป ดังนั้นจึงทำให้เราสามารถใช้งานอินเทอร์เน็ต (Internet) ความเร็วสูง (ADSL) พร้อมกับการใช้งานโทรศัพท์ธรรมดาได้ในเวลาเดียวกัน สามารถทำการล็อกอิน (log in) ได้ทันทีโดยไม่ต้องทำการต่อสายโทรศัพท์ จึงทำให้ไม่ต้องเสียค่าใช้จ่ายโทรศัพท์ในการเข้าใช้งาน และยังมีปัญหาสายหลุดระหว่างใช้งานอีกด้วยสำหรับความเร็วในการดาวน์โหลดของ ADSL จะอยู่ที่ 8 Mbps และการอัปโหลดจะอยู่ที่ 1 Mbps ซึ่งเหมาะสมที่จะใช้ในการส่งข้อมูล (file transfer), มัลติมีเดีย, การประชุมทางไกลและเกมออนไลน์

2.3 จาวา

เป็นภาษาสำหรับเขียนโปรแกรม ซึ่งเป็นลักษณะของโปรแกรมเชิงวัตถุ (OOP : Object Oriented Programming) ได้รับการพัฒนามาจาก programming language ที่เรียกว่า “Oak” โดยบริษัท Sun Microsystems โดยมุ่งหมายให้เกิดภาษาที่ใช้งานได้บนเว็บและใช้ได้กับคอมพิวเตอร์ทุกเครื่องโดยไม่ต้องทำการเขียนโปรแกรมขึ้นใหม่อีกไม่ว่าจะใช้ระบบปฏิบัติการ (Platform) ใดๆ

ในการพัฒนางานด้วย Java จะมีเครื่องมือที่ใช้พัฒนาโปรแกรม คือ Java 2 Platform Standard Development Kit หรือ J2SDK ซึ่งสามารถดาวน์โหลดฟรีได้ที่ www.java.sun.com เป็น java platform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ถูกพัฒนาจนปัจจุบันนี้เป็น J2SE 1.5.0.1 โดยบริษัท SUN ได้เพิ่มความสามารถต่างๆให้มากขึ้น และมีประสิทธิภาพเพิ่มขึ้น รวมทั้งเพิ่มในส่วนของ “Class Libraries” หรือที่เรียกกันว่า “Application Programming Interfaces” หรือ APIs โดย APIs นี้จะมีโปรแกรมสำเร็จรูปพร้อมใช้เรียกว่า “components” ให้เลือกนำมาใช้งานได้ทันที โดยไม่ต้องเสียเวลาในการพัฒนาโปรแกรมนั้นๆ ขึ้นมาใหม่

ภายใน SDK จะมีคอมไพเลอร์ (Compiler) ที่ใช้สำหรับแปลซอสโค้ด (Source Code) ให้เป็นไฟล์ไบนารีโค้ด (Byte Code) ที่สามารถทำงานได้ทั้งแบบที่เป็น Java Application และ Java Applet ในส่วนของ SDK นอกจากจะมีเครื่องมือที่ใช้ในการ Compile และ Interpret แล้ว ยังมี JRE และกลุ่มของ Libraries ต่างๆ ที่ใช้ในโปรแกรมด้วย เครื่องมือของ Java SDK จะมีทั้งหมด 23 ชุด แต่ในที่นี้จะกล่าวถึงเพียง 7 ชุด ได้แก่

เครื่องมือ	หน้าที่
javac	คำสั่งของคอมไพเลอร์ที่ใช้เปลี่ยนจากซอสโค้ดเป็นไบนารีโค้ด
java	คำสั่งของตัวแปลภาษาที่สั่งให้แอปพลิเคชันทำงาน โดยการอ่านจากไบนารีโค้ดสำหรับใช้ใน Java Application
appletviewer	ใช้สำหรับสั่งให้ Java Applet ทำงานเพื่อทดสอบก่อนการนำไปใช้งานจริงผ่านบราวเซอร์
javadoc	เป็นเครื่องมือที่ใช้อธิบายโปรแกรมจาวาโดยสร้างเอกสาร HTML จากโปรแกรมจาวา
jdb	เป็นตัว debugger ของจาวาที่ใช้สำหรับตรวจสอบการทำงานของโปรแกรม โดยสั่งให้โปรแกรมหยุดการทำงานตามขั้นตอนที่โปรแกรมเมอร์กำหนด เพื่อตรวจสอบตัวแปรและคู่ลำดับในการทำงานของชุดคำสั่ง
javap	เป็นตัว Java Disassembler ซึ่งใช้ทดสอบไบนารีโค้ดที่เกิดขึ้นแล้วแสดงข้อมูล (Method หรือ Function) ของไบนารีโค้ดนั้น ถูกเรียกใช้ในกรณีที่มีปัญหาในเรื่องของการเรียกประมวลผลโปรแกรม
jar	คือตัวจัดการเกี่ยวกับไฟล์ Jar Archive (JAR) โดยการรวมและบีบไฟล์หลายๆไฟล์ รวมถึงระดับไดเรกทอรีให้เป็นไฟล์เดียว เพื่อความสะดวกในการใช้งาน

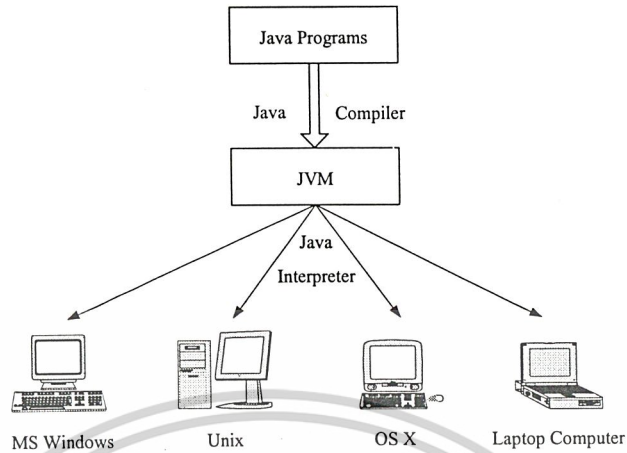
ตารางที่ 2.2 เครื่องมือของ Java SDK

2.3.1 Java Virtual Machine (JVM)

เป็นกลไกเสมือนซึ่งสร้างขึ้นโดยตัวแปลภาษา (Interpreter) ของจาวา มีขั้นตอนการทำงานคือนำข้อมูลที่อยู่ในรูปของโปรแกรมจาวา มาผ่านการคอมไพล์ด้วยตัวคอมไพเลอร์ที่อยู่ในเครื่องที่สร้างโปรแกรมของจาวา ซึ่งจะทำได้ไฟล์ไบนารีโค้ดที่ JVM สามารถเข้าใจได้ หลังจากนั้นจึงทำการเรียกใช้ตัวแปลภาษาเพื่อสั่งให้โปรแกรมทำงาน โดยในระหว่างแปลภาษานี้ ตัวแปลภาษาจะสร้าง JVM ขึ้นมาเพื่อนำโค้ดที่ได้จากการคอมไพล์มาเข้ากระบวนการที่สร้างขึ้น ซึ่งตัวแปลภาษานี้ไม่จำเป็นต้องอยู่เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

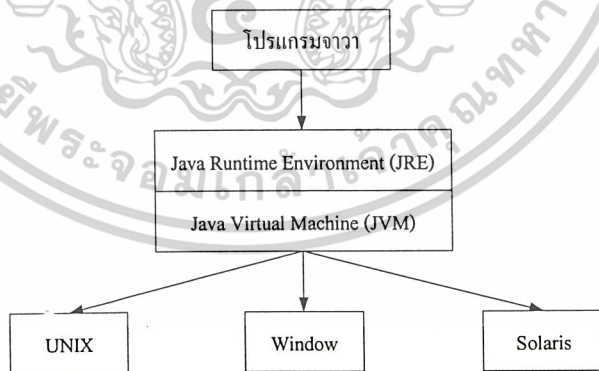
เดียวกับซอสโค้ดก็ได้ การใช้งานก็เพียงแค่นำตัวแปลภาษาไปติดตั้งที่เครื่องที่ต้องการไว้ ดังนั้นภาษาจาวาจึงทำงานได้เร็วและไม่ขึ้นกับระบบว่าเป็นระบบปฏิบัติการแบบใด



รูปที่ 2.5 Java Virtual Machine

2.3.2 Java Runtime Environment (JRE)

จาวาได้จัดเตรียมองค์ประกอบต่างๆที่สนับสนุนการทำงานของโปรแกรมที่สร้างขึ้นอยู่ใน “Java Runtime Environment (JRE)” ซึ่งจะประกอบด้วยไฟล์ที่จำเป็นต้องใช้ในการทำงานของจาวา เช่น ไฟล์ที่บรรจุแพ็คเกจต่างๆ ไฟล์ใช้สำหรับแสดงรูปแบบตัวอักษร เป็นต้น โดยที่ไฟล์ต่างๆเหล่านี้จะถูกติดตั้งพร้อมกับ J2SDK การทำงานของ JRE จะนำโปรแกรมที่สร้างขึ้นจากจาวาผ่านกระบวนการของ JRE คือการนำองค์ประกอบที่จำเป็นสำหรับเรียกโปรแกรมทำงานมาใช้ จากนั้นจึงเข้าไปทำงานในส่วนของ JVM



รูปที่ 2.6 ลำดับการทำงานของเครื่องรันโปรแกรมจาวา

2.3.3 การสร้างแอปพลิเคชันของจาวา

การสร้างโปรแกรมจาวาแบบแอปพลิเคชัน มีขั้นตอนดังนี้

1. จัดเตรียมเครื่องมือที่ใช้สำหรับการเขียนและรันโปรแกรม สิ่งที่ต้องเตรียม คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Editor Tools หรือเครื่องมือสำหรับเขียนโปรแกรม เช่น Notepad, Editplus, JBuilder
 - J2SDK เป็นเครื่องมือที่จำเป็นสำหรับการเขียนโปรแกรมจาวา เนื่องจากภายในบรรจุเครื่องมือที่จำเป็นสำหรับการเขียนโปรแกรมไว้ด้วย เช่น JVM JRE, คอมไพเลอร์, ตัวแปลภาษา
2. ติดตั้ง J2SDK ให้เรียบร้อย
 3. เริ่มเขียนโปรแกรมโดยใช้ Editor แล้วบันทึกไฟล์ให้มีนามสกุลเป็น .java ซึ่งไฟล์นี้เมื่อใช้คำสั่ง javadoc.exe จะทำให้ได้ไฟล์ HTML ที่อธิบายส่วนประกอบต่างๆ ของไฟล์จาวานั้น
 4. คอมไพล์โปรแกรมโดยใช้คำสั่ง javac.exe จะได้ไฟล์ที่มีนามสกุล .class
 5. เมื่อต้องการให้โปรแกรมทำงานจึงเรียกใช้ java.exe เพื่อรันโปรแกรม
 6. ในขณะที่โปรแกรมกำลังทำงานอาจเรียกใช้ jdb.exe เพื่อทำการแก้ไขโปรแกรม

2.3.4 แบบจำลองเชิงวัตถุ (Object Model)

ปัจจุบันนี้การพัฒนาซอฟต์แวร์ด้วยการเขียนโปรแกรมในแนวดั้งเดิม เริ่มจะมีความนิยมลดลง เนื่องจากแนวทางในการเขียนโปรแกรมไม่สามารถรองรับความซับซ้อนที่เพิ่มมากขึ้นได้ แนวทางการเขียนโปรแกรมเริ่มพัฒนามาจากการเขียนแบบ Subprogram และแบบ Procedure จนมาถึงลักษณะแบบโมดูล ได้มีการพัฒนาโปรแกรมเพื่อสนับสนุนการเขียนโปรแกรมในลักษณะเหล่านี้มากมาย เช่น COBOL, FORTRAN, Pascal และ C เป็นต้น จนกระทั่งมีการพัฒนาการเขียนโปรแกรมเชิงวัตถุที่มีลักษณะสำคัญคือ การมองข้อมูล เหตุการณ์ หรือปัญหาต่างๆ ให้เป็น ออบเจกต์ (Object) โดยใช้แบบจำลองที่เรียกว่า “แบบจำลองเชิงวัตถุ” จากนั้นจึงทำการแยกแยะออบเจกต์เหล่านั้นให้เป็นหมวดหมู่ที่เรียกว่า คลาส (Class) เพื่อพัฒนาเป็นโปรแกรมต่อไป

โปรแกรมภาษาที่พัฒนาขึ้นมา เพื่อสนับสนุนการเขียนโปรแกรมเชิงวัตถุเริ่มมีความนิยมเพิ่มมากขึ้นจนมาถึงโปรแกรมภาษาที่เป็นที่นิยมมากที่สุดในปัจจุบันคือ จาวา

2.3.4.1 แนวคิดแบบ Object Model Technology (OMT)

จะใช้การมองสิ่งต่างๆภายในระบบงานให้เป็นวัตถุหรือออบเจกต์และกำหนดลักษณะและพฤติกรรมให้แก่วัตถุเหล่านั้น การจำลองหรือมองสิ่งใดๆ ก็ตามให้เป็นวัตถุได้นั้นในทางออบเจกต์ Object Oriented จะใช้หลักเกณฑ์หรือแนวความคิดในการจำลองหรือมองสิ่งเหล่านั้นด้วยหลักเกณฑ์หรือแนวคิดดังกล่าวเรียกว่า “แบบจำลองเชิงวัตถุ”

- ออบเจกต์และคลาส

“ออบเจกต์” เป็นองค์ประกอบสำคัญของการเขียนโปรแกรมคือเป็นตัวแทนของสิ่งที่อยู่ในโลกความเป็นจริง ซึ่งอาจเป็นสิ่งที่จับต้องได้หรือไม่ได้จะต้องมีคุณลักษณะ (Attribute) ที่บ่งบอกว่า เป็นออบเจกต์ของอะไรและมีพฤติกรรม (Behavior) ที่บ่งบอกว่า ออบเจกต์นั้นทำอะไร ส่วน “คลาส” เป็นกลุ่มของออบเจกต์ที่มีคุณลักษณะและพฤติกรรมหรือการกระทำโดยทั่วไปเหมือนกัน

- **Abstraction และ Instantiation**

เราจะเรียกกระบวนการในการให้แนวความคิดกับออบเจกต์ จนเกิดขึ้นเป็นคลาสว่า “Abstraction” และเรียกกระบวนการของการเกิดออบเจกต์จากคลาสที่สร้างขึ้นว่า “Instantiation”

- **คุณลักษณะและเมธอด (Method)**

“คุณลักษณะ” คือ การบรรยายคุณสมบัติของออบเจกต์ต่างๆ ในโลกนี้ได้โดยคุณสมบัตินี้เป็นคุณสมบัติที่เราสนใจหรืออยู่ในโดเมนที่เราสนใจนั่นเอง ส่วน “เมธอด” เป็นเครื่องมือที่มีอยู่ในคลาสและออบเจกต์ที่กำหนดขึ้นมาเพื่อทำงานหรือทำหน้าที่อย่างใดอย่างหนึ่ง โดยที่เมธอดที่อยู่ต่างออบเจกต์กันจะสามารถเรียกใช้ซึ่งกันและกันได้ (ถ้าเมธอดนั้นอยู่ในสถานะเป็น Public)

2.3.5 ข้อดีของการเลือกใช้ภาษาจาวา

เนื่องจากภาษาจาวาเป็นภาษาที่ได้รับความนิยมสูงมาก เพราะโครงสร้างของจาวาสามารถรองรับการทำงานได้หลากหลาย รวมถึงคุณสมบัติอื่นๆ ที่สนับสนุน ได้แก่

- ผู้ที่เคยใช้โปรแกรมที่ได้รับความนิยมอื่นๆ มาก่อน เช่น C, C++ และ Smalltalk สามารถใช้โปรแกรมจาวาได้อย่างคุ้นเคย เนื่องจากจาวาได้ดึงเอาคุณสมบัติที่ดีของภาษาเหล่านี้มาปรับปรุงให้ใช้งานได้ง่ายและมีประสิทธิภาพมากขึ้น
- มีการทำงานที่ติดต่อกับเว็บเพจอย่างใกล้ชิด จึงทำให้ได้รับความนิยมเป็นอย่างมาก
- สามารถรองรับการทำงานกับภาษาโปรแกรมอื่นๆ ในอนาคตได้ทุกภาษา
- จาวาใช้หลักการของ Object Orientation ซึ่งเป็นรูปแบบภาษาการเขียนโปรแกรมแบบใหม่ที่มีข้อดีมากมาย
- เขียนโปรแกรมน้อยลงเนื่องจากจาวาสนับสนุนคุณสมบัติของการนำกลับมาใช้ใหม่ (reuse) ทำให้แอปพลิเคชันที่สร้างขึ้นมีขนาดเล็ก และสามารถนำไปใช้ร่วมกับแอปพลิเคชันอื่นๆ ได้อย่างเหมาะสม
- เขียนโปรแกรมครั้งเดียวสามารถนำไปใช้ได้ทุกระบบปฏิบัติการ ที่มี Java Runtime Environment (JRE)

2.4 จาวาเทเลโฟนี API (Java Telephony API : JTAPI)

JTAPI หรือ Java Telephony Application Programming Interface เป็น API พื้นฐานสำหรับแอปพลิเคชันที่เกี่ยวข้องกับระบบโทรศัพท์ที่เป็นภาษาจาวา ซึ่งถูกออกแบบให้การเขียนโปรแกรมนั้นทำได้ง่ายขึ้น ในขณะที่ยังคงความสามารถที่จำเป็นไว้สำหรับการสร้างแอปพลิเคชันในระดับที่สูงขึ้นต่อไป

โปรแกรมที่เขียนขึ้นโดยใช้ JTAPI จะสามารถทำการเชื่อมต่อระหว่างระบบโทรศัพท์กับเครื่องคอมพิวเตอร์รูปแบบต่างๆ ให้ใช้งานร่วมกันได้ โดยเครื่องมือของ JTAPI นี้สามารถใช้งานร่วมกับ computer-telephony integration platform ที่พบเห็นอยู่ในปัจจุบันได้ อาทิเช่น Sun Microsystem's SunXTL™, Microsoft's TAPI, Intel's TAPI, Novell' TSAPI, Avaya's TAPI และ IBM's CallPath และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนั้นอุปกรณ์ของบริษัทผู้ให้บริการโทรศัพท์นั้นอาจเลือกที่จะจัดเตรียมเครื่องมือของ JTAPI ไว้สำหรับอุปกรณ์ของตนเองได้เพื่อรองรับการใช้งานที่มีอยู่หลากหลายได้อีกด้วย

ภายใน JTAPI จะจัดเตรียมกลไกสำหรับเข้าถึงรูปแบบการบริการของผู้ให้บริการรายต่างๆ ไว้ภายในมี “core API” หรือ “core package” ซึ่งจัดเตรียม *call model* และความสามารถที่เป็นลักษณะพื้นฐานทั่วไปของโทรศัพท์ เช่น การตอบรับการเรียกเข้าของโทรศัพท์ “core API” นี้จะถูกล้อมรอบด้วย *standard extension package* ซึ่งจัดเตรียมรูปแบบการทำงานที่เพิ่มขึ้นจาก core API เช่น การทำงานแบบ *call center* และการเข้าถึงสื่อสัญญาณที่เข้ามา (*media stream access*)

ความสามารถและแนวทางของกระบวนการออกแบบของ JTAPI มีดังนี้

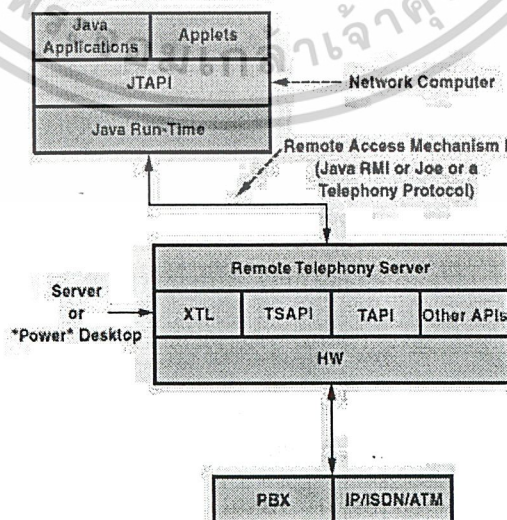
- สามารถสร้างแอปพลิเคชันพื้นฐานที่เกี่ยวข้องกับระบบโทรศัพท์ได้ง่าย
- มีเฟรมเวิร์กที่ทำให้แอปพลิเคชันในคอมพิวเตอร์นั้นทำงานเป็น *call center* ในระบบโทรศัพท์
- อินเทอร์เน็ตแอปพลิเคชันสามารถให้บริการกับผู้ใช้งานได้โดยตรง หรือทำหน้าที่เป็นอินเทอร์เน็ตของภาษาจาวาไปยัง *telephony API* อื่นๆ เช่น *TSAPI* และ *TAPI*
- Core package จะถูกขยายด้วย *standard extension packages*
- สามารถทำงานได้ทุกระบบปฏิบัติการ ที่สามารถใช้ *Java Runtime Environment (JRE)* ได้

2.4.1 รูปแบบการพัฒนาจาวาเทเลโฟนี API (Configuration)

2.4.1.1 รูปแบบการพัฒนาเพื่อใช้งานแบบเน็ตเวิร์ก (Network Computer (NC)

Configuration๑)

แอปพลิเคชันของ JTAPI จะรันอยู่ที่เครื่องคอมพิวเตอร์เครื่องอื่นที่ไม่ใช่ตัวเซิร์ฟเวอร์ซึ่งสามารถเข้าถึงทรัพยากรในโครงข่ายได้ แล้วทำการเรียกใช้ตัวเซิร์ฟเวอร์ที่จัดการกับทรัพยากรของระบบโทรศัพท์อีกที JTAPI จะติดต่อกับเซิร์ฟเวอร์เช่นเดียวกับ *Remote Method Invocation (RMI)* ของภาษาจาวาหรือโปรโตคอลในระบบโทรศัพท์ รูปแบบของ *Network Computer Configuration* จะเป็นดังรูปที่ 2.7



รูปที่ 2.7 Network computer configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.2 รูปแบบการพัฒนาเพื่อใช้งานบนเครื่องคอมพิวเตอร์ส่วนตัว

(Desktop Computer Configuration)

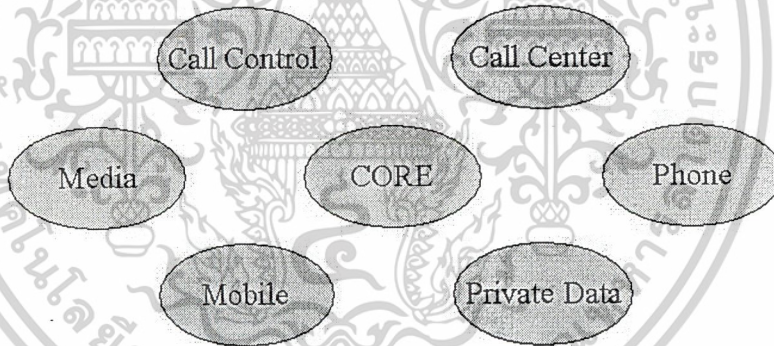
รูปแบบนี้ JTAPI จะรันอยู่บนเครื่องคอมพิวเตอร์ที่เชื่อมต่อกับโทรศัพท์ ซึ่งแสดงไว้ในรูปที่ 2.8

Java Applications		Applets	
Java Telephony API			
Java Run-Time			
XTL	TSAPI	TAPI	Other APIs
Telephony H/W (POTS Card/Fax Card)			

รูปที่ 2.8 Desktop configuration

2.4.2 โครงสร้างของแพ็คเกจ

JTAPI จะประกอบด้วยแพ็คเกจต่างๆ ซึ่งแต่ละแพ็คเกจจะประกอบด้วยส่วนของการทำงานที่เฉพาะเจาะจงสำหรับ computer telephony applications ดังรูปที่ 2.9



รูปที่ 2.9 Core Package and Extension Package

ภายใน JTAPI 1.4 ภายในแบ่งเป็น core package และ standard extension package

2.4.2.1 แพ็คเกจคอร์ (Core package) จะประกอบด้วย JTAPI interface และ class ต่างๆ มีเฟรมเวิร์คพื้นฐานต่างๆ ที่จะจำลอง “call” และความสามารถพื้นฐานทั่วไปของโทรศัพท์ ความสามารถนี้จะรวมถึงการวางเส้นทางการเรียกของโทรศัพท์ (placing a telephone call), การตอบรับการเรียกเข้าของโทรศัพท์ (answering a telephone call) และการตัดสายโทรศัพท์ (disconnecting a connection to a telephone call) ซึ่งแอปพลิเคชันที่เกี่ยวข้องกับโทรศัพท์พื้นฐานทั่วไปจะจำเป็นต้องใช้เพียง core package เท่านั้น ซึ่งจะสามารถเรียกใช้งานได้จากแพ็คเกจ javax.telephony

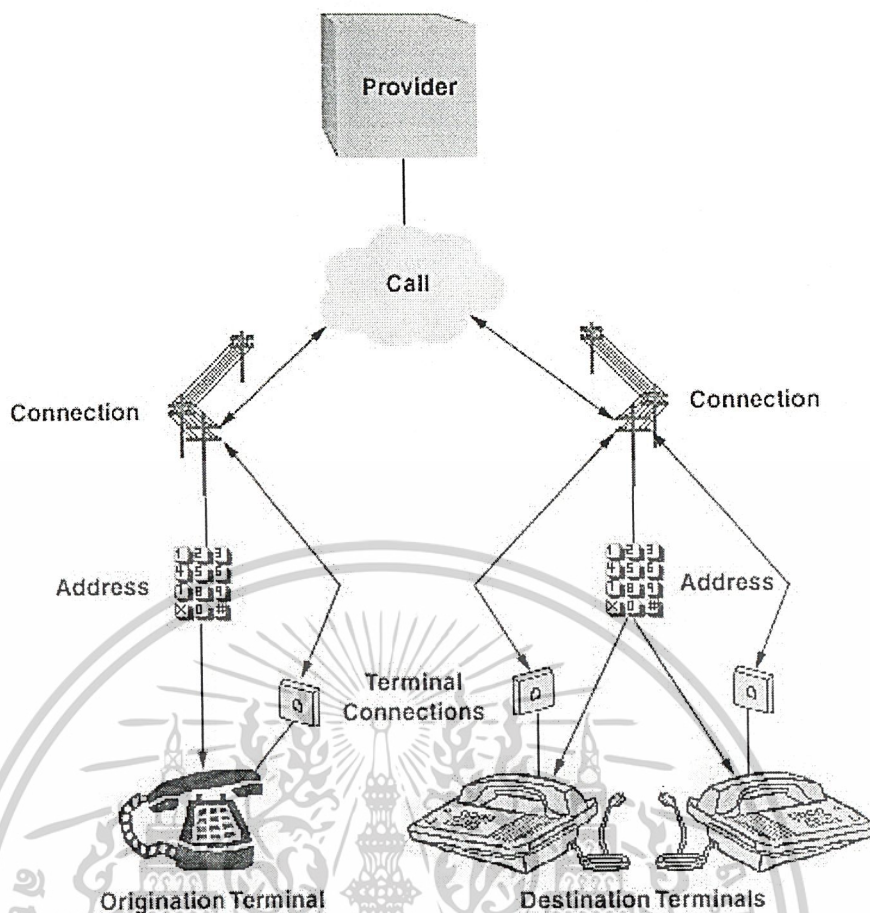
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.2 แพคเกจสแตนด์เอิร์ทเอ็นชัน (Standard extension package) แต่ละแพคเกจจะมีส่วนที่ขยายที่เฉพาะเจาะจงลงไปจาก core API แอปพลิเคชันจะเลือกใช้ extension package เมื่อมีความจำเป็นเพื่อทำให้งานสำเร็จลุล่วงตามที่ต้องการ ส่วนในการติดตั้งเครื่องมือของตัวเซิร์ฟเวอร์ของระบบโทรศัพท์ จะต้องเลือกว่า extension package ใดที่เหมาะสมกับการใช้งานเครื่องมืออื่นๆ ซึ่งจะต้องอาศัยความสามารถ (capabilities) ของอุปกรณ์พื้นฐานเหล่านั้นมาประกอบด้วย Standard extension package ของ JTAPI 1.4 มีอยู่ดังนี้

- **javax.telephony.callcontrol** เป็นแพคเกจที่ขยายจาก core package ซึ่งจะจัดการการพัฒนาแบบและรายละเอียดเกี่ยวกับ call model โดยมีความสามารถในการทำงานในเรื่องของการควบคุมการเรียก (call control features) ที่สูงขึ้น เช่น conferencing telephone call และมีรายละเอียดเกี่ยวกับสถานะที่เพิ่มขึ้นอีก
- **javax.telephony.callcenter** เป็นแพคเกจที่รองรับการทำงานในรูปแบบของ call center ซึ่งมีลักษณะการทำงานดังนี้ Automatic Call Distribution (ACD)เป็นการใช้เครื่องมือ(mechanism) เพื่อตอบรับการเรียกเข้าจากภายนอก , การหาเส้นทางไปยังปลายทางเพื่อตอบสนองต่อการเรียกเข้า (routing) โดยมี Agent คือ บุคคลหรือผู้ที่ทำหน้าที่ตอบรับการเรียกเข้าโทรศัพท์
- **javax.telephony.media** เป็นแพคเกจที่ทำให้แอปพลิเคชันนั้นสามารถจัดการกับสื่อสัญญาณที่ไหลอยู่ในสายโทรศัพท์ได้ สามารถสร้างและตรวจจับสัญญาณ DTMF และสัญญาณอื่นๆ, ทำการเล่นและบันทึกข้อความ, เทคโนโลยี Text-to-Speech
- **javax.telephony.phone**จะอนุญาตให้แอปพลิเคชันนั้นสามารถควบคุมส่วนประกอบทางกายภาพของชุดอุปกรณ์โทรศัพท์ได้ การใช้งานนั้นอาจจะต้องอธิบาย Terminal เป็นส่วนประกอบต่างๆ ซึ่งส่วนประกอบแต่ละชนิดนี้จะมีอินเตอร์เฟซอยู่ในแพคเกจนี้ด้วย
- **javax.telephony.capabilities** จะอนุญาตให้แอปพลิเคชันตามว่าความสามารถใดที่จะสามารถเข้าถึงแพคเกจนั้นๆ ได้ ซึ่งจะส่งค่ากลับคืนมาเป็นจริงหรือเท็จ แบ่งเป็น 2 แบบ คือ static capabilities และ dynamic capabilities
- **javax.telephony.privatedata** จะอนุญาตให้แอปพลิเคชันติดต่อกับข้อมูลได้โดยตรงด้วยอุปกรณ์สวีตซ์ ข้อมูลเหล่านี้จะส่งออกไปที่สวีตซ์เพื่อแสดงผลที่เกิดขึ้นของสวีตซ์ตัวนั้นๆ

2.4.3 แบบจำลองการเรียก (Call model)

call model ประกอบด้วยออบเจกต์ของจาวาที่ถูกกำหนดขึ้นโดยใช้อินเตอร์เฟซของจาวาใน core package แต่ละออบเจกต์ของ call model จะแทนทั้งส่วนที่มีอยู่จริงในระบบโทรศัพท์ทั้งทางกายภาพและในเชิงตรรกะเพื่อใช้อธิบายการเรียกของโทรศัพท์และจุดปลายทางที่เกี่ยวข้องกับการเรียก



รูปที่ 2.10 JTAPI Call model

จากรูปที่ 2.10 เป็นการแสดง JTAPI call model และออบเจกต์ต่างๆ ที่ประกอบกันเป็น call model ซึ่งมีดังนี้

- **Provider object**
เป็นออบเจกต์ที่แทนโปรแกรมของผู้ให้บริการโทรศัพท์ ซึ่งผู้ให้บริการจะปลุกปิดการทำงานของระบบภายในเอาไว้ และให้แอปพลิเคชันหรือแอปเพลต (applet) ของจาวาทำการติดต่อกับระบบภายใน
- **Call object**
Call object จะแทนการเรียกของโทรศัพท์ ข้อมูลที่ไหลอยู่ระหว่างผู้ให้บริการโทรศัพท์และผู้ใช้บริการโทรศัพท์ การเรียกของโทรศัพท์นี้จะประกอบด้วย Call object และ Connection ซึ่งอาจจะเป็นศูนย์หรือมากกว่าศูนย์ก็ได้
- **Address object**
Address object จะแทนหมายเลขโทรศัพท์ซึ่งเป็นจุดปลายในเชิงตรรกะของการเรียก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Connection object**

Connection object จะจำลองการเชื่อมต่อระหว่าง Call object และ Address object ซึ่งจะอยู่ในสถานะใดสถานะหนึ่ง เพื่อระบุว่าขณะนั้นสถานะที่เกิดขึ้นระหว่าง Call และ Address

- **Terminal object**

Terminal object จะแทนอุปกรณ์ทางกายภาพ เช่น โทรศัพท์และอุปกรณ์ที่เกี่ยวข้อง แต่ละ Terminal object อาจจะมี Address object ที่เกี่ยวข้องด้วยอยู่เพียงหนึ่งหรือมากกว่าหนึ่ง Address ก็ได้

- **TerminalConnection object**

TerminalConnection object จะจำลองความสัมพันธ์ระหว่าง Connection กับจุดปลายทางกายภาพของการเรียก ซึ่งก็คือ Terminal object โดยที่ TerminalConnection จะอธิบายสถานะปัจจุบันที่เกิดขึ้นระหว่าง Connection และ Terminal ที่เฉพาะเจาะจง

2.4.4 เมธอดในแพ็คเกจคอร์ (Core Package Methods)

แพ็คเกจคอร์มีการกำหนดเมธอด 3 เมธอดขึ้นมาเป็นความสามารถพื้นฐาน ซึ่งมีดังนี้

2.4.4.1 เมธอด Call.connect()

เมื่อทำการสร้าง Call object ผ่านเมธอด Provider.createCall() แอปพลิเคชันจะทำการเรียกของโทรศัพท์ด้วยเมธอด Call.connect() ซึ่งจำเป็นต้องระบุ Terminal และ Address ดันทาง และต้องจัดเตรียมเบอร์โทรศัพท์ปลายทางไว้ Connection object จะถูกส่งกลับมาจากเมธอดนี้

2.4.4.2 เมธอด TerminalConnection.answer()

แอปพลิเคชันจะตรวจตราการเรียกเข้าที่เกิดขึ้นที่ Terminal ด้วย Listener สายที่เรียกเข้ามาที่ Terminal นี้จะถูกแสดงด้วย TerminalConnection ในสถานะ RINGING และ ณ เวลานั้น แอปพลิเคชันเรียกใช้เมธอด TerminalConnection.answer() เพื่อที่ตอบรับการเรียกของโทรศัพท์ที่เข้ามา

2.4.4.3 เมธอด Connection.disconnect()

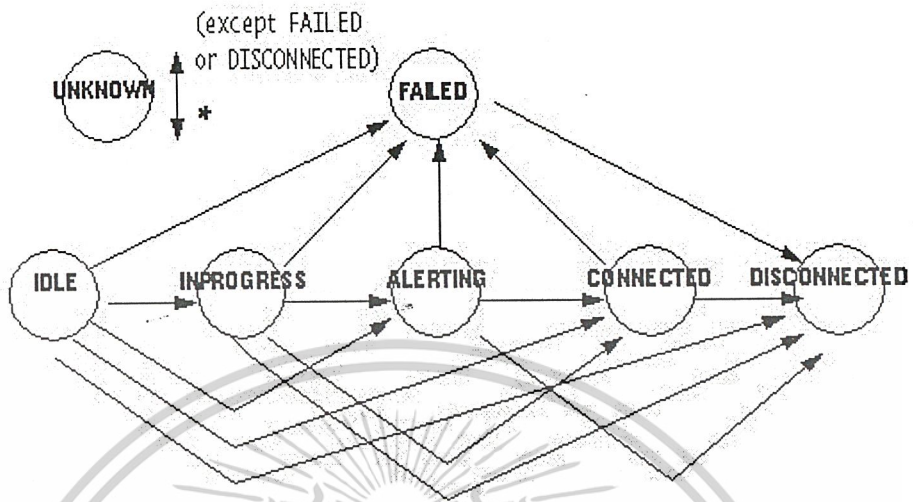
เมธอด Connection.disconnect() จะใช้เพื่อนำ Address ออกจากการเรียกของโทรศัพท์ Connection object จะแทนความสัมพันธ์ระหว่าง Address กับการเรียกของโทรศัพท์ โดยทั่วไปแอปพลิเคชันจะเรียกใช้เมธอดนี้เมื่อ Connection อยู่ในสถานะ CONNECTED และผลเมื่อมีการเปลี่ยนแปลงสถานะไปเป็น DISCONNECTED สำหรับแพ็คเกจคอร์แอปพลิเคชันอาจจะนำ Address ทั้งหมดออกจากการเรียกนั้น รวมทั้ง Terminal ทั้งหมดที่เกี่ยวข้องกับ Address ที่มีการเรียกนั้นเกิดขึ้นก็จะถูกนำออกไปเช่นกัน ใน call control extension package มีความสามารถสำหรับใช้กับแอปพลิเคชันเพื่อที่จะนำ Terminal เฉพาะบาง Terminal เท่านั้นออกจากการเรียกนั้น

2.4.5 สถานะของ Connection Object

Connection Object ปกติจะอยู่ในรูปของสถานะที่แสดงความสัมพันธ์ที่เกิดขึ้นระหว่าง Call และ Address สถานะของ Connection ที่ปรากฏอยู่นั้นจะเป็นตัวระบุว่าเมธอดและการกระทำใดที่จะถูกเรียกใช้ได้จาก Connection object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงของสถานะของ Connection object จะถูกควบคุมโดยกฎซึ่งแสดงเป็นไดอะแกรม Connection State Transitions ดังรูปที่ 2.11 โดยจะแสดงถึงการเปลี่ยนแปลงสถานะที่เป็นไปได้ของ Connection Object สถานะต่างๆ ที่เป็นไปได้นี้ ได้แก่



รูปที่ 2.11 Connection State Transitions

- **IDLE state**
เป็นสถานะเริ่มต้นสำหรับทุกๆ Connection object ใหม่ เพื่อใช้ระบุว่าผู้เช่าหรือผู้ใช้บริการโทรศัพท์ได้เริ่มเข้ามาพร้อมใช้งานแล้ว และไม่มี core method ใดที่ถูกประกาศใช้เมื่ออยู่ในสถานะนี้
- **INPROGRESS state**
สถานะ INPROGRESS จะระบุว่า การเรียกนั้นกำลังถูกวางไปยังปลายทางอยู่
- **ALERTING state**
สถานะ ALERTING จะระบุว่าผู้เช่าปลายทางนั้นถูกแจ้งให้ทราบแล้วว่า มีสายเรียกเข้า
- **CONNECTED state**
สถานะ CONNECTED จะระบุว่าคู่สายนั้นถูกใช้งานเป็นส่วนหนึ่งของการเรียกของโทรศัพท์ เมื่อ Connection อยู่ในสถานะ CONNECTED นั้นมีความหมายว่าผู้เช่ากำลังสนทนากับผู้เช่ารายอื่นอยู่หรืออาจจะถูกเชื่อมต่อสัญญาณอยู่ในขณะนั้น
- **DISCONNECTED state**
สถานะ DISCONNECTED จะระบุว่าผู้เช่านั้นไม่ได้เป็นส่วนหนึ่งของการเรียกของโทรศัพท์แล้วและไม่มีเมธอดใดๆ ที่ถูกประกาศไว้สำหรับสถานะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **FAILED state**

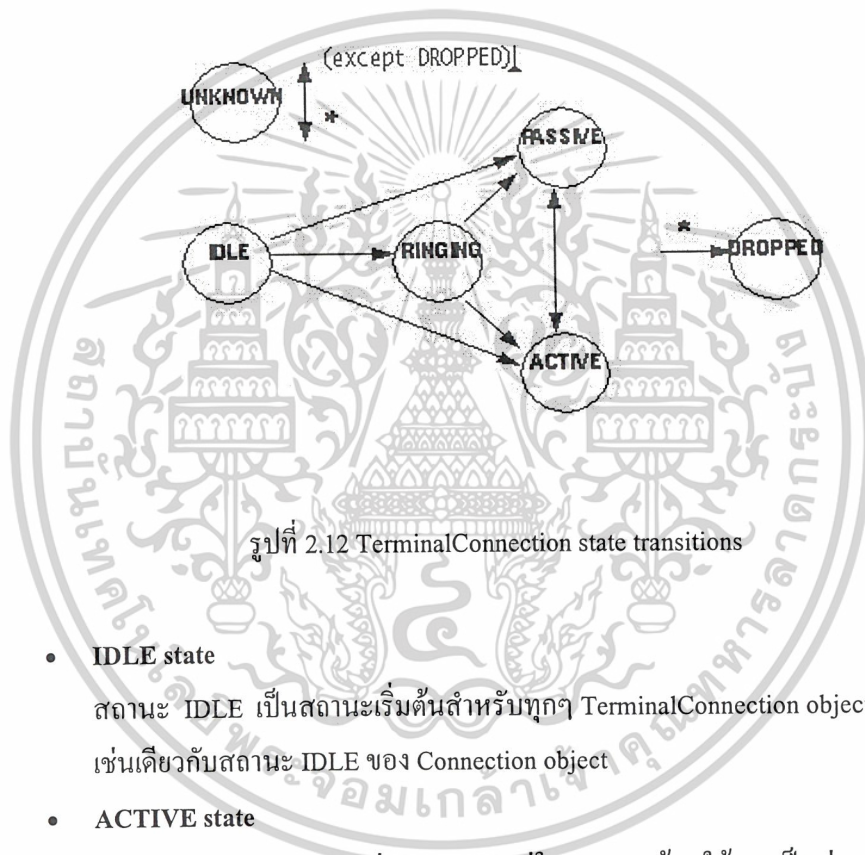
สถานะ FAILED จะระบุว่าการทำงานเรียกของโทรศัพท์นั้นล้มเหลว เช่น ถ้าแอปพลิเคชันเรียกใช้ Call.connect() เพื่อจัดวางเส้นทางยังคู่สายที่ไม่ว่าง Connection ที่เกี่ยวข้องกับคู่สายที่ทำการเรียกจะเปลี่ยนสถานะไปยังสถานะ FAILED

- **UNKNOWN state**

สถานะ UNKNOWN จะระบุว่าผู้ให้บริการนั้นไม่สามารถตัดสินได้ว่าสถานะของ Connection ในขณะนั้นควรเป็นเช่นไร

2.4.6 สถานะของ TerminalConnection Object

TerminalConnection object จะแทนความสัมพันธ์ระหว่าง Terminal และ Connection ซึ่งมีสถานะต่างๆ ตาม state transition diagram ดังรูปที่ 2.12



รูปที่ 2.12 TerminalConnection state transitions

- **IDLE state**

สถานะ IDLE เป็นสถานะเริ่มต้นสำหรับทุกๆ TerminalConnection object ซึ่งมีความหมายเช่นเดียวกับสถานะ IDLE ของ Connection object

- **ACTIVE state**

สถานะ ACTIVE จะระบุว่า Terminal อยู่ในสภาวะพร้อมใช้งานเป็นส่วนหนึ่งของการเรียกของโทรศัพท์ ซึ่งมีความหมายว่าโทรศัพท์นั้น off-hook อยู่

- **RINGING state**

สถานะ RINGING จะระบุว่า Terminal จะส่งสัญญาณไปยังผู้ใช้ที่มีสายเรียกเข้ามาในขณะนั้น

- **DROPPED state**

สถานะ DROPPED จะระบุว่า Terminal นั้นหลุดจากการเรียกของโทรศัพท์แล้ว ซึ่งสถานะนี้เป็นสถานะสุดท้ายสำหรับทุกๆ TerminalConnection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

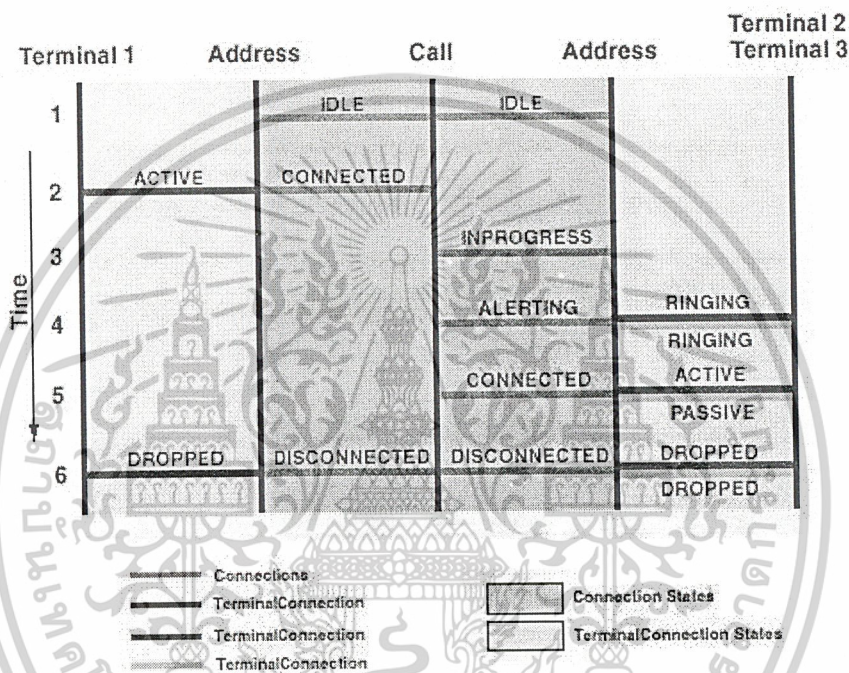
- **PASSIVE state**

สถานะ PASSIVE จะระบุว่า Terminal นั้นเป็นส่วนหนึ่งของการเรียกของโทรศัพท์ แต่ว่าไม่อยู่ในสภาวะที่พร้อมใช้งาน เป็นการระบุว่าทรัพยากรที่ Terminal นั้นถูกใช้งานโดยการเรียกนั้นๆ

- **UNKNOWN state**

สถานะ UNKNOWN จะระบุว่าผู้ให้บริการนั้นไม่สามารถที่จะตัดสินได้ว่าสถานะขณะนั้นเป็นเช่นไร ซึ่งคล้ายกับสถานะ UNKNOWN ของ Connection object

2.4.7 การวางเส้นทางการเรียกของโทรศัพท์ (Placing a Telephone Call)



รูปที่ 2.13 Call Model timing diagram

จากรูปที่ 2.13 แสดงการเปลี่ยนแปลงของสถานะที่เกิดขึ้นผ่าน call model ซึ่งบรรยายการเปลี่ยนแปลงของสถานะที่เกิดขึ้นหลังจากแอปพลิเคชันเรียกใช้เมธอด Call.connect() ซึ่งเป็นกรณีที่การเรียกเกิดขึ้นระหว่างผู้เช่า 2 ผู้เช่า โคดอะแกรมจะถูกแบ่งออกเป็น 2 ส่วน คือทางด้านขวาส่วนหนึ่ง และทางด้านซ้ายอีกส่วนหนึ่ง ส่วนทางด้านซ้ายจะแทนปลายสุดของต้นทางที่ทำการเรียกนั้น และส่วนทางด้านขวาจะแทนส่วนปลายสุดของปลายทางของการเรียกนั้น

ทางด้านซ้ายของโคดอะแกรม จะมีเส้นตรงในแนวตั้ง 2 เส้นแทน Terminal object และ Address object ของฝั่งต้นทาง ส่วนเส้นในแนวอนจะแทนทั้ง Connection object และ TerminalConnection object เช่นเดียวกับทางซ้าย ทางด้านขวาเส้นต่างๆจะแทนออบเจกต์ต่างๆ ของฝั่งปลายทาง

จากตัวอย่างการทำการเรียกของโทรศัพท์ เมื่อ Connection ทั้งสองฝั่งอยู่ในสถานะ IDLE Connection ของทางต้นทางจะเปลี่ยนไปอยู่ในสถานะ CONNECTED ขณะที่ Connection ปลายทางจะเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนไปอยู่ในสถานะ INPROGRESS ณ เวลานั้น TerminalConnection ของฝั่งต้นทางก็จะถูกสร้างขึ้น และมีสถานะเปลี่ยนไปเป็น ACTIVE เมื่อ Connection ปลายทางเปลี่ยนสถานะไปเป็น ALERTING TerminalConnection ปลายทางก็จะถูกสร้างให้อยู่ในสถานะ RINGING ซึ่งเมื่อถึงจุดนี้ ที่ปลายทางจะมีผู้รับโทรศัพท์ เมื่อเหตุการณ์นี้เกิดขึ้นเมื่อไหร่ TerminalConnection ที่เหลืออยู่จะเปลี่ยนไปอยู่ในสถานะ PASSIVE ซึ่งในเวลาเดียวกันนี้ ที่ Connection ปลายทางจะอยู่ในสถานะ CONNECTED และเมื่อทำการวางสาย ทุก Connection จะเปลี่ยนไปอยู่ในสถานะ DISCONNECTED และทุก TerminalConnection จะเปลี่ยนไปอยู่ที่สถานะ DROPPED

2.4.8 แบบจำลองการรายงานเหตุการณ์ของจาวาเทเลโฟนนี่ (The Java Telephony Event Delivery Model)

JTAPI จะแจ้งให้แอปพลิเคชันทราบถึงการเปลี่ยนแปลงที่เกิดขึ้นใน JTAPI call model ซึ่งจะรวมถึงการเปลี่ยนแปลงของสถานะของออบเจกต์หรือการสร้างออบเจกต์ขึ้นมา

- **Listeners และ Observers**

แบบจำลอง observer หรือ observable เป็นรูปแบบที่แอปพลิเคชันของ JTAPI สามารถติดตามรับ event ต่างๆจากการทำงานของ JTAPI โดยใช้ API ของ JTAPI และด้วยความสามารถของ JTAPI 1.4 แอปพลิเคชันจะมีการเปลี่ยนแปลงสถานะ โดยการนำแบบแผนการส่ง event มาใช้ และนอกจากนี้ JTAPI 1.4 ยังรวมเมธอดและ event ต่างๆของ Observer ไว้ แต่เมื่อเรียกมาใช้งานก็จะทำการ deprecate

- **Listener Interfaces และ Events**

แอปพลิเคชันจะอธิบายการเปลี่ยนแปลงที่เกิดขึ้นกับออบเจกต์ด้วย Listener Interface ซึ่ง Listener Interface นี้เป็นแบบจำลอง Listener ของ JTAPI ซึ่งจะระบุไปยังชุดของเมธอดที่แอปพลิเคชันต้องใช้งาน โดยที่แต่ละ event จะมีเมธอดอยู่หนึ่งเมธอดที่จะถูกระบุไปยังออบเจกต์ของแอปพลิเคชันนั้นๆ

- **Objects และ Listeners**

Provider, Call, Terminal และ Address Object จะมี Listener interface สำหรับออบเจกต์แต่ละตัวอยู่ ซึ่งได้แก่ ProviderListener, CallListener, TerminalListener และ AddressListener

- ProviderListener จะรายงานการเปลี่ยนแปลงสถานะทั้งหมดที่เกิดกับ Provider object ซึ่ง ภายในแพ็คเกจจอร์ การเปลี่ยนแปลงของสถานะของ Provider จะเริ่มจาก OUT_OF_SERVICE ไปเป็น IN_SERVICE หรือไปเป็น SHUTDOWN
- CallListener จะรายงานข้อมูลการเปลี่ยนแปลงสถานะของทุก Connection และ TerminalConnection ไปยัง Address listener และ Terminal listener ในกรณีที่มีสายเรียกเข้าแล้วแอปพลิเคชันต้องการจะแสดง Address object และ Terminal object บนหน้าจอ แอปพลิเคชันจะเรียกใช้ Address.addCallListener() หรือ Terminal.addCallListener() เพื่อทำการเพิ่ม CallListener ไปยัง call ที่เรียกเข้ามาที่ Address และ Terminal อย่างอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ CallListener เหล่านี้จะถูกนำออกไปเมื่อ call นั้นได้ออกจาก Address และ Terminal ไปแล้ว

- AddressListener และ TerminalListener จะรายงานการเปลี่ยนแปลงของสถานะที่เกิดขึ้นกับออบเจกต์ทั้งสองนี้ ภายในแพ็คเกจจอร์จะไม่มี event ใดๆสำหรับ Address object และ Terminal object

2.4.9 การระบุและการจัดหา Providers (Locating and obtaining Providers)

JTAPI จะอธิบายถึงระเบียบแบบแผนของการใช้งานของ JTAPI ของผู้ให้บริการโทรศัพท์ ซึ่งทำให้แอปพลิเคชันนั้นสามารถใช้งานร่วมกับบริการของผู้ให้บริการโทรศัพท์ได้ ซึ่งมีส่วนประกอบอยู่ 2 ส่วนที่ใช้เชื่อมต่อแอปพลิเคชันไปยังเซิร์ฟเวอร์ของผู้ให้บริการ คือ

- **JtapiPeerFactory**

JtapiPeerFactory class เป็นจุดแรกของการติดต่อสำหรับแอปพลิเคชันที่ต้องการใช้บริการโทรศัพท์ เป็น static class ที่มีเมธอดที่ประกาศเป็น public เพียงเมธอดเดียวคือ `getJtapiPeer()` ซึ่งจะนำชื่อ class ของ JTAPI server implementation ที่ต้องการไปเป็นพารามิเตอร์เพื่อส่งชื่อ JtapiPeer object นั้นกลับมา และถ้าหากไม่มีชื่อบอกไว้ก็จะส่ง JtapiPeer object ที่ติดต่อกันอยู่ขณะนั้น (default JtapiPeer object) กลับมา

- **JtapiPeer**

JtapiPeer interface เป็นเครื่องมือพื้นฐานของผู้ให้บริการโทรศัพท์ ผู้ให้บริการแต่ละรายจะต้องมีส่วนที่ทำงานเป็นอินเทอร์เฟซนี้ ในคลาสที่บรรจุโดย JtapiPeerFactory การใช้งานผ่านคลาสอินเทอร์เฟซนี้ จะทำให้แอปพลิเคชันนั้นสามารถรับทราบ Provider object, ถามถึงบริการที่มีอยู่ของ Provider และรับค่าที่เป็นชื่อของ JtapiPeer object นั้นได้ โดยการเรียกใช้เมธอดดังต่อไปนี้

- `JtapiPeer.getProvider()` เพื่อรับ Provider object ใหม่เข้ามา และแอปพลิเคชันยังอาจจะจัดส่งส่วนที่เพิ่มเติมผ่านเมธอดนี้ได้
- `JtapiPeer.getServices()` เพื่อเรียกดูบริการที่มีของ Provider
- `JtapiPeer.getName()` เพื่อเรียกดูชื่อของ JtapiPeer object ซึ่งจะเป็นชื่อเดียวกับที่ใช้เป็นตัวแปรในเมธอด `JtapiPeerFactory.getJtapiPeer()`

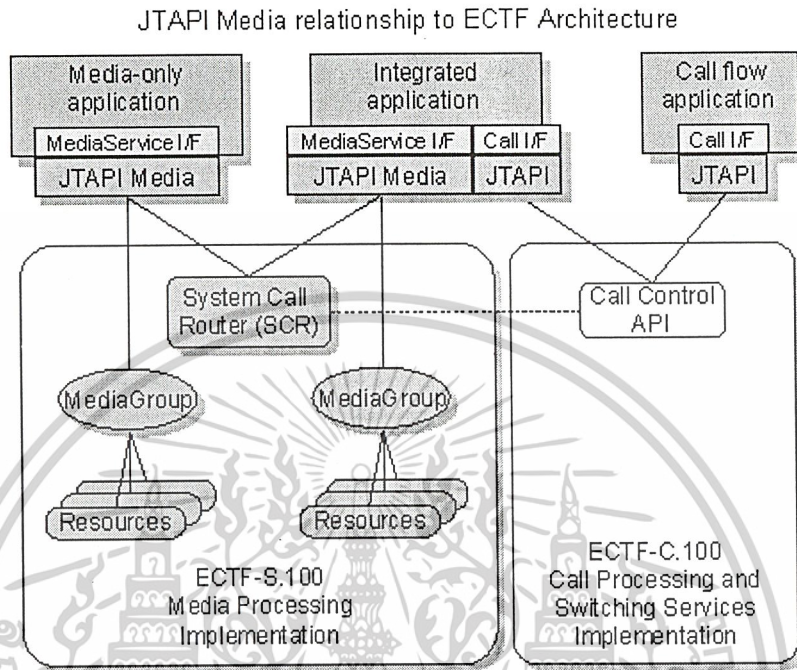
2.4.10 แพคเกจ javax.telephony.media

แพคเกจมีเดียของ JTAPI จะกำหนด “MediaService” และวิธีการในการเชื่อมต่อสำหรับการนำสื่อสัญญาณไปใช้ประโยชน์ แพคเกจนี้เหมาะสมสำหรับแอปพลิเคชันที่เกี่ยวกับสัญญาณและการจัดทางเส้นทาง เช่น การนำไปใช้ในระบบ interactive voice response (IVR), voice mail, speech recognition, text-to-speech และแฟกซ์ เป็นต้น

JTAPI Media เป็นการอธิบายถึงโครงสร้างและอินเทอร์เฟซของ ECTF media services ในลักษณะของภาษาจาวา ถูกออกแบบมาให้แอปพลิเคชันนั้นสามารถใช้งานสื่อสัญญาณในเฉพาะ high-level ในโครงสร้างของ ECTF (Enterprise Computer Telephony Forum) ได้กำหนดเฟรมเวิร์คสำหรับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริการต่างๆ ที่อาจจะใช้สื่อสัญญาณในแอปพลิเคชัน และยังรวม service สำหรับการจัดการกับสื่อสัญญาณและ call ได้ดีพอๆ กับอินเทอร์เน็ตที่ใช้ในการจัดการสิ่งเหล่านี้โดยตรง JTAPI Media นี้สามารถร่วมเข้ากับ ECTF ได้และมีโครงสร้างที่เข้ากันกับโครงสร้างของ ECTF อีกด้วย ซึ่งความสัมพันธ์ของทั้ง 2 โครงสร้างแสดงไว้ดังรูปที่ 2.14



รูปที่ 2.14 ความสัมพันธ์ทางโครงสร้างของ JTAPI Media และ ECTF

จุดมุ่งหมายของโครงสร้างนี้คือการให้บริการเกี่ยวกับสื่อสัญญาณต่างๆ อาทิเช่น การอัดเสียง (voice recording), การส่งแฟกซ์ (fax transmission) นั้นสามารถสร้างขึ้นได้อย่างอิสระและยังเข้ากับบริการอื่นๆ ได้ด้วย

MediaService เป็นอินเทอร์เน็ตเฟสที่ใช้กำหนดคอนโทรลเฟสระหว่างแอปพลิเคชันและการเรียกที่เกิดขึ้น และองค์ประกอบที่ใช้ในการจัดการข้อมูลหรือทรัพยากรที่จำเป็นของเฟรมเวิร์ค

ใน JTAPI Media มีคลาส BasicMediaService ซึ่งสามารถแบ่งเมธอดตามจุดมุ่งหมายในการใช้เมธอดนั้นๆ ได้ 3 กลุ่มด้วยกัน ดังนี้

- Bind and Release methods
- Generic media group methods
- Resource specific methods

BasicMediaService ถูกกำหนดให้เป็นคลาสคลาสหนึ่งที่แอปพลิเคชันนั้นสามารถที่จะสร้างขึ้นหรือทำการ extend class เพื่อกำหนดการใช้งาน service ด้วยผู้สร้างแอปพลิเคชันเอง BasicMediaService ใช้หลักการ *bind/release* เพื่อใช้ในการเข้าถึงสื่อสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.10.1 หลักการ Bind และ Release

MediaTerminal extend จาก *Terminal interface* เพื่อเป็นการระบุว่า *Terminal* นั้นมีความสามารถในการจัดเตรียมสื่อสัญญาณสำหรับการทำงานของคอมพิวเตอร์ หรือ DSP processing สื่อสัญญาณนั้นจะต้องสามารถติดต่อกับ *Resources* ซึ่งจะเป็นตัวกำเนิดและรับข้อมูลในสื่อสัญญาณได้อย่างหลากหลาย การจัดเก็บของ *Resources* จะเชื่อมต่อไปยัง *MediaTerminal* เพื่อสร้าง *MediaGroup* object

เมื่อมีการเรียกมาถึง *MediaTerminal* SCR ซึ่งทำหน้าที่จัดการ *Terminal* นั้นๆ จะตัดสินใจว่าแอปพลิเคชันใดจะควบคุม *MediaTerminal* นั้นก่อน สื่อสัญญาณจะเชื่อมต่อไปยัง *MediaGroup* และ *MediaGroup* จะ *bound* ไปยัง *MediaService* object ของแอปพลิเคชัน ในขณะที่ *MediaGroup* ยัง *bound* อยู่ เมล็ดที่เกี่ยวข้องบน *MediaService* จะถูกส่งต่อไปยัง *MediaGroup*

เมื่อการเรียกออกจาก *MediaTerminal* ไป อาจจะโดยการส่งต่อไปยัง *MediaTerminal* อื่นๆ หรือการเรียกนั้นสิ้นสุดลง แอปพลิเคชันยังคงมี *MediaService* object อยู่ แต่จะ *released* (unbound) จาก *MediaGroup* เฟรมเวิร์คจะเข้าใช้งาน *MediaTerminal* นั้นได้ทันที ส่วน *MediaGroup* และ *Resources* จะไม่เกิดการรบกวนจากการเชื่อมต่อกับแอปพลิเคชันก่อนหน้านี้

2.4.10.2 อินเทอร์เฟซ MediaService

อินเทอร์เฟซ *MediaService* ได้กำหนดเมธอดสำหรับการเข้าถึงสื่อสัญญาณของการเรียก (call's media stream) โดยการเชื่อมต่อ *MediaService* objects ไปยังชุดของ *media processing resource* ซึ่งจะเกี่ยวข้องกับ *Terminal* และ *resources* จะดำเนินการเกี่ยวกับสื่อสัญญาณของ *Connection* นั้นๆ ที่มีสถานะ *active* ที่ *Terminal* นั้นๆ วิธีการเชื่อมต่อมีดังต่อไปนี้

- **void bindToServiceName (ConfigSpec configSpec, String serviceName)**
เมธอดนี้จะรอการเรียก (Call) เพื่อตรงไปยัง *serviceName* ที่เฉพาะเจาะจง และจะทำการเชื่อมต่อ *MediaService* ไปที่ *MediaTerminal* ที่ติดต่อกับ Call นั้นๆ
- **void bindAndConnect (ConfigSpec configSpec, String origAddr, String dialDigits)**
เมธอดนี้ใช้ในกรณีทำการเรียกไปยังหมายเลขปลายทาง แล้วติดต่อกับ Call นี้ไปยัง *MediaTerminal* ปลายทางและทำการเชื่อมต่อ *MediaService* นี้ไปยัง *MediaTerminal* นั้น
- **Connection bindToCall (ConfigSpec configSpec, Call call)**
เมธอดนี้จะสร้าง *Connection* ใหม่ขึ้นมาจาก Call ที่ติดต่อกับบาง *MediaTerminal* และทำการเชื่อมต่อ *MediaService* ไปยัง *MediaTerminal* นั้น
- **void bindToTerminal (ConfigSpec configSpec, Terminal terminal)**
เป็นการเชื่อมต่อ *MediaService* ไปยัง *Terminal* ที่เฉพาะเจาะจง การจัดเตรียมการเข้าถึงสื่อสัญญาณไปยัง Call หรือ *Connection* ที่ *active* อยู่บน *Terminal* นั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **void bindToTerminalName (ConfigSpec configSpec, String terminalName)**
เป็นการเชื่อมต่อ MediaService ไปยัง terminal โดยการให้ implementation-specific terminalName string ซึ่งแตกต่างจาก bindToTerminal() ตรงที่เมธอดนี้จะใช้ในกรณี
ที่ underlying framework ไม่สนับสนุน JTAPI provider

เมธอดที่ใช้ในการ release

- **void release()**
การปลดการเชื่อมต่อระหว่าง MediaService และ MediaTerminal จะเป็นการแสดง
ว่า MediaService นี้จะไม่สนใจในการให้บริการ (servicing) กับ call แล้ว โดย
ปกติเมื่อไม่มี MediaService อื่นๆ เชื่อมต่อกับ MediaTerminal นี้แล้ว ก็จะทำให้การ
disconnects the call และจัดเตรียม MediaTerminal ที่ว่างนี้สำหรับ call ที่จะเข้ามา
ใหม่
- **void releaseToService(String disposition, int timeout)**
การปลดการเชื่อมต่อระหว่าง MediaService และ MediaTerminal โดยคาดว่า Call
และ Connection ที่อยู่บน MediaTerminal นั้นๆ จะถูกดำเนินการต่อโดย
MediaService อื่นๆ เมื่อไม่มีแอปพลิเคชันใดๆ อยู่บน MediaTerminal ในช่วงเวลา
timeout (milliseconds) Exception จะถูก thrown และ MediaService ปัจจุบันจะเป็น
เจ้าของ call นี้ต่อไป

2.4.10.3 ECTF Resource interfaces

ECTF Resource interfaces มีทั้งหมด 9 อินเตอร์เฟสแต่ในที่นี้ขอกล่าวถึงเฉพาะ
อินเตอร์เฟสที่จะนำมาใช้ในปฏิญานพนธ์เท่านั้น อันได้แก่

- **Player**
ทำการเล่น ไฟล์ media data object (MDO)
Basic method:
 - play(String streamID, int offset, RTC[] rtc, Dictionary optargs);
streamID : เป็น String ซึ่งใช้เป็นชื่อไฟล์ MDO ที่จะใช้ในการเล่นเสียง
offset : ข้อมูลชนิดตัวเลข (integer) แสดงเวลาในหน่วย millisecond เพื่อ
แสดงเวลาที่ตั้งไว้ก่อนทำการเล่นไฟล์ MDO
rtc : อาร์เรย์ของ RTC ที่มีผลต่อการเล่นเสียง
optargs : พจนานุกรมของ optional arguments
- **Recorder**
ทำการบันทึกไปยัง MDO file.
Basic method:
 - record(String streamID, RTC[] rtc, Dictionary optargs);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

streamID : เป็น String ซึ่งใช้เป็นชื่อไฟล์ MDO ที่จะใช้ในการเล่นเสียง
 rtcs : อาร์เรย์ของ RTC ที่มีผลต่อการเล่นเสียง
 optarg : พจนานุกรมของ optional arguments

- SignalGenerator

สร้างและส่งสัญญาณ DTMF (และสัญญาณอื่นๆ)

Basic methods:

- sendSignals(String signals, RTC[] rtcs, Dictionary optarg);
- sendSignals(Symbol[] signals, RTC[] rtcs, Dictionary optarg);

signals : อาร์เรย์ของ Symbol ที่ระบุถึงสัญญาณ

rtc : อาร์เรย์ของ RTC objects ซึ่งควบคุมการดำเนินการนี้

optarg : พจนานุกรมของ optional arguments

- SignalDetector

ตรวจจับสัญญาณ DTMF และรูปแบบของสัญญาณที่ถูกระบุจากแอปพลิเคชัน สัญญาณที่ตรวจจับได้ และรูปแบบของสัญญาณที่ถูกกำหนดโดยแอปพลิเคชันจะสามารถใช้ในการควบคุมกระบวนการ play และ record โดยการใช้ RTCs (run time controls).

Basic method:

- retrieveSignals(int numSignals, Symbol[] patterns, RTC[] rtcs, Dictionary optarg);

numSignals : ข้อมูลชนิดตัวเลขแสดงจำนวนของสัญญาณที่จะถูกลับมา

patterns : เป็นการระบุว่า Patterns ใดที่จะทำการหยุดการตรวจจับสัญญาณ

rtcs : อาร์เรย์ของ RTC (Run Time Control) objects การกระทำของ RTC อาจจะเป็น rtca_Stop หรือ rtca_FlushBuffer.

Optarg : พจนานุกรมของ optional arguments และ p_Pattern[i] อาจจะถูกกำหนดให้เป็น transient parameters

2.4.11 เจเนอริก JTAPI (Generic JTAPI : GJTAPI)

เจเนอริก JTAPI Framework เป็นเครื่องมือที่จะนำ JTAPI ไปใช้งานได้สะดวกยิ่งขึ้น ซึ่ง GJTAPI นี้ถูกออกแบบให้สนับสนุนการเชื่อมต่อในระดับ low-level หรือ “raw” interface จุดมุ่งหมายหนึ่งของการสร้างเฟรมเวิร์กนี้ขึ้นมา ก็เพื่อที่จะช่วยให้การพัฒนาแอปพลิเคชันที่ใช้ความสามารถของ JTAPI สามารถทำได้ง่ายยิ่งขึ้น โดยลดความซับซ้อนของการอินเตอร์เฟสที่ต้องนำไปใช้งาน โดย GJTAPI framework จะสามารถจัดการส่วนของ JTAPI ได้ดังนี้

- All core functionality
- Under “callcontrol” : hold, unhold, transfer, conference, release
- Media control (version 1.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Deprecated 1.2 Media Control

การกำหนดค่าเริ่มต้น (Initialization)

1. แอปพลิเคชันจะรับค่า GenericJtapiPeer ผ่านทางคำสั่ง getJtapiPeer(String jtapiPeerName)
2. จากนั้นก็ทำการรับค่า GenericProvider ผ่านทางคำสั่ง getProvider(String providerString) และสอบถามไปยัง TelephonyProvider สำหรับการเปลี่ยนแปลงค่าของเซตของ capabilities ใหม่ และใช้ในการสร้าง static Capabilities objects
3. GenericProvider จะร้องขอ TelephonyProvider รายชื่อทั้งหมดของ Address
4. ในแต่ละ address name GenericProvider จะร้องขอ TelephonyProvider ถึงชื่อของ Terminal ที่เกี่ยวข้องทั้งหมด
5. ในแต่ละ terminal name GenericProvider จะร้องขอ TelephonyProvider ว่าควรกำหนดเป็น MediaTerminal หรือเป็น Terminal ทั่วไป
6. GenericProvider จะสร้างชุดของ Address และ Terminal ที่ตายตัวขึ้นมา

2.5 SQL Server 2000

ระบบฐานข้อมูลถือเป็นหัวใจสำคัญของการทำแอปพลิเคชันทั่วไป เนื่องจากเป็นแหล่งที่เก็บรวบรวมข้อมูล ซึ่งจะถูกแปรเปลี่ยนให้สามารถนำไปวิเคราะห์ต่อการใช้งานได้ต่อไป โดย SQL Server 2000 เป็นโปรแกรมจัดการฐานข้อมูลระดับเซิร์ฟเวอร์ที่มีขีดความสามารถในการรองรับข้อมูลขนาดใหญ่ ช่วยให้การบริหารจัดการฐานข้อมูลมีประสิทธิภาพ รวดเร็ว สามารถตอบสนองต่อความต้องการขององค์กรขนาดใหญ่ได้ดี SQL Server 2000 แบ่งออกได้หลายเวอร์ชันตามลักษณะการใช้งานดังนี้

- **Personal Edition** เป็นเวอร์ชันที่ออกแบบมาเพื่อใช้กับงานฐานข้อมูลที่มีขนาดเล็ก โดยสามารถใช้งานได้เฉพาะภายในเครื่องโลคอล (Local) เท่านั้น หรืออาจกล่าวได้ว่าเป็นแบบ Stand Alone ส่วนใหญ่นิยมนำมาใช้ในการทดสอบการเขียนแอปพลิเคชัน

- **Standard Edition** เป็นเวอร์ชันมาตรฐานที่ใช้สำหรับฐานข้อมูลทั่วไป สามารถรองรับการใช้งานของเครื่องเซิร์ฟเวอร์ที่มีขนาดโปรเซสเซอร์ (processor) ได้สูงสุด 4 CPU หน่วยความจำสูงสุด 2 GB

- **Enterprise Edition** เป็นเวอร์ชันที่ขยายขีดความสามารถจากเวอร์ชันมาตรฐาน เพื่อให้สามารถรองรับการทำงานได้กับเครื่องเซิร์ฟเวอร์ที่มีขนาดโปรเซสเซอร์สูงสุดถึง 32 CPU หน่วยความจำสูงสุด 64 GB นิยมนำมาใช้งานกับองค์กรที่มีขนาดใหญ่และมีสาขามากมาย เพราะมีระบบสนับสนุนการทำงานมากมาย เช่น การทำ Data Mining, Data Warehouse เป็นต้น โดย SQL Server 2000 มีวิธีการจัดการระบบฐานข้อมูล ทั้งการสร้างตาราง เปลี่ยนแปลง หรือ ทำลาย รวมถึงการกำหนดคอปชั่นต่างๆ ของฐานข้อมูล โดยตารางจะเป็นส่วนสำคัญหลักของระบบฐานข้อมูล ดังนั้นจะอธิบายถึงส่วนประกอบต่างๆ ของตาราง ประเภทของข้อมูลที่จัดเก็บ ตลอดจน การใช้ภาษา SQL เพื่อทำคิวรี (Query) ข้อมูลที่อยู่ในตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 ส่วนประกอบของตารางฐานข้อมูล

Airport Code	Airport Name	Airport City	Airport Country	Airport ID
BKK	Don Muang Int'l	Bangkok	Thailand	1
HKG	International	Hong Kong	China	2
ICN	Incheon Int'l	Seoul	Republic of Korea	3
SIN	Changi	Singapore	Singapore	4

คีย์หลัก
คอลัมน์

รูปที่ 2.15 ส่วนประกอบของตารางฐานข้อมูล

จากรูปที่ 2.15 ตารางประกอบด้วย 2 ส่วนหลักคือ คอลัมน์ (Column) เมื่อมองในแนวนิ่ง และ แถว (Row) เมื่อมองในแนวนอน

2.5.1.1 คอลัมน์

คอลัมน์หรือฟิลด์ (Field) จะถูกกำหนดเป็นชื่อที่ไม่ซ้ำกันสำหรับในตารางหนึ่งๆ โดยมีการระบุประเภทของข้อมูลที่ถูกนำมาเก็บในฟิลด์นี้ รวมถึงการกำหนดขนาดความยาวของข้อมูลที่สามารถรองรับในแต่ละฟิลด์เหล่านี้ได้หรือความสามารถในการยอมรับข้อมูลที่ไม่ทราบค่า (Unknown) หรือ Null Value

2.5.1.2 แถว

แถว หรือ เร็คคอร์ด (Record) แทนลักษณะของสิ่งหนึ่งสิ่งใดที่อยู่ภายในบรรทัดเดียวกัน อธิบายง่าย ๆ คือ ข้อมูลที่เกิดขึ้นในแต่ละคอลัมน์โดยพิจารณาที่บรรทัดเดียวกันเท่านั้น

สำหรับความสามารถของ SQL Sever 2000 ภายใน 1 ฐานข้อมูลสามารถบรรจุตารางไว้ได้มากถึง 2 ล้านตาราง และภายใน 1 ตารางสามารถมีจำนวนคอลัมน์ได้มากที่สุด 1,024 คอลัมน์ โดยข้อมูลที่สามารถจัดเก็บได้ในแต่ละคอลัมน์มีขนาดสูงสุดเท่ากับ 8,000 ไบต์

2.5.1.3 คีย์ (Key)

นอกจากองค์ประกอบทางกายภาพที่เห็นเด่นชัดสำหรับแถวและคอลัมน์แล้ว ตารางยังประกอบไปด้วยคีย์หลายประเภทเพื่อใช้ทำหน้าที่ที่แตกต่างกันออกไป

1. คีย์คู่แข่ง (Candidate Key หรือ Alternate Key) โดยคอลัมน์ หรือกลุ่มของคอลัมน์ที่เป็นคีย์คู่แข่ง จะต้องประกอบไปด้วยคุณสมบัติ 2 ประการ

- Uniqueness คือต้องมีค่าไม่ซ้ำกันใน 2 แถวใดๆ ภายในตารางเดียวกัน
- Minimality คือประกอบไปด้วยจำนวนคอลัมน์น้อยสุด ที่ทำให้ค่า 2 แถวใดๆ ไม่ซ้ำกัน

โดยพิจารณาจากตารางที่ 2.3 จะพบว่าตาราง author มีคีย์คู่แข่งเป็นไปตามคุณสมบัติข้างต้น คือ คอลัมน์ authored

authorID	firstName	lastName
1	Harvey	Deitel
2	Paul	Deitel
3	Tem	Nieto
4	Sean	Santry

ตารางที่ 2.3 ตัวอย่างข้อมูลจากราย author

2. คีย์หลัก (Primary Key) เป็นคีย์คู่แข่งที่ถูกเลือกให้มาทำหน้าที่ บอกความแตกต่างประจำตารางนั้นๆ ซึ่งสามารถมีได้เพียงคีย์เดียวสำหรับตารางหนึ่งๆ เท่านั้น ซึ่งในบางครั้งอาจไม่สามารถหา คีย์หลักเดี่ยว (Simple Key) ในการค้นหาข้อมูลได้ ดังนั้นต้องใช้มากกว่า 1 คอลัมน์มาประกอบกันเพื่อทำหน้าที่บอกความแตกต่าง ยกตัวอย่างดังตารางที่ 2.4 ซึ่งจะใช้ทั้งคอลัมน์ OrderID และ ProductID มาร่วมกันทำหน้าที่เป็นคีย์หลักของตาราง โดยอาจเรียกคีย์ที่กล่าวถึงนี้ว่า คีย์ร่วม (Composite Key หรือ Combine Key) อาจกล่าวได้ว่าคีย์หลักจะใช้ประโยชน์ในการระบุ (identify) ข้อมูลที่ต้องการเข้าถึงและช่วยไม่ให้เกิดการผิดพลาดเนื่องจากป้อนข้อมูลที่ซ้ำกันลงในคอลัมน์ที่ไม่อนุญาตให้มีข้อมูลซ้ำ ถ้าใส่ข้อมูลซ้ำโปรแกรมจะแสดงข้อความเตือนและไม่ทำงานต่อจนกว่าจะแก้ไขข้อมูลให้ถูกต้อง

OrderID	ProductID	Quantity
1000	TVC100	10
1000	RFG100	5
1001	IRN110	3
1001	RFG100	15

ตารางที่ 2.4 ตัวอย่างข้อมูลจากราย Order Detail

3. คีย์นอก (Foreign Key) เป็นคอลัมน์ที่ไม่ใช่คีย์หลักของตารางหนึ่ง แต่กลายเป็นคีย์หลักให้อีกตารางหนึ่ง

4. น็อนคีย์ (Non Key) คือคอลัมน์ที่ไม่ใช่ส่วนใดส่วนหนึ่งของคีย์คู่แข่งหรือคีย์หลัก

2.5.2 การจัดการและทำงานกับข้อมูลในฐานข้อมูล

SQL ย่อมาจากคำว่า Structured Query Language หมายถึง ภาษามาตรฐานกลางที่ใช้สำหรับจัดการข้อมูลในฐานข้อมูล เราจะใช้ภาษา SQL เพื่อทำควิรี่ข้อมูลที่อยู่ในตารางต่างๆ โดยมีจุดประสงค์ต่างกัน เช่น การเรียกดูข้อมูลทั้งหมด, การเรียกดูข้อมูลแบบมีเงื่อนไขอาจจะมาจากตารางเดียวหรือหลายตารางก็ได้ แล้วนำมาแสดงร่วมกันในเวลาเดียวกัน, การแก้ไขข้อมูล, การลบข้อมูล เป็นต้น ดังนั้นการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำควิรี่จึงเปรียบเสมือนกับเป็นการสร้างตารางเสมือนขึ้นมา เพื่อนำข้อมูลที่อยู่ในตารางต่าง ๆ มาแสดงร่วมกัน

เพื่อให้่ายต่อความเข้าใจในการใช้คำสั่งต่างๆ จึงยกตัวอย่างฐานข้อมูลของระบบจองตั๋วเครื่องบิน ซึ่งประกอบด้วยตาราง Flight Instance, Flight schedule

Flight number	Date	Plane number	available seats
TG401	16/3/2005	777	100
TG402	16/3/2005	333	50
TG601	17/3/2005	773	25
TG603	17/3/2005	773	20

ตารางที่ 2.5 ตัวอย่างข้อมูลจากตาราง Flight Instance

Flight number	From airportcode	To airportcode	Departure time	Arrival time
TG401	BKK	SIN	19:15	22:35
TG402	SIN	BKK	08:20	09:40
TG409	BKK	SIN	15:15	18:35

ตารางที่ 2.6 ตัวอย่างข้อมูลจากตาราง Flight schedule

Airport Code	Airport Name	Airport City	Airport Country	Airport ID
BKK	Don Muang Int'l	Bangkok	Thailand	1
HKG	International	Hong Kong	China	2
ICN	Incheon Int'l	Seoul	Republic of Korea	3

ตารางที่ 2.7 ตัวอย่างข้อมูลจากตาราง Airport

2.5.2.1 ประโยคคำสั่ง SELECT

เป็นคำสั่งสำหรับเรียกดูข้อมูลจากตารางหรือแสดงเร็คคอร์ดจากตารางต่างๆ ที่ต้องการ มีรูปแบบประโยคคำสั่งการใช้งาน 2 กรณีคือ

1. การเรียกดูข้อมูลแบบไม่มีเงื่อนไข เป็นการแสดงข้อมูลทุกแถวที่อยู่ในแต่ละตารางมีรูปแบบประโยคคือ

```
SELECT column_name1, column_name2,..... FROM table_name
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อคีย์เวิร์ด **SELECT** จะตามด้วยชื่อคอลัมน์ที่ต้องการแสดงซึ่งจะคั่นด้วยเครื่องหมายจุลภาคไปเรื่อยๆ ซึ่งใช้ระบุคอลัมน์ที่จะถูกคืนค่ากลับมาจากการคิวรีและ **FROM** ตามด้วยชื่อตารางที่ต้องการ เช่น **SELECT [Flight number], Date, [available seats] FROM [Flight Instance]** เป็นการคืนค่าในคอลัมน์ **Flight number, Date** และ **available seats** สำหรับทุกแถว จากตาราง **Flight Instance** ซึ่งผลที่ได้จะแสดงดังตารางที่ 2.8 จะเห็นได้ว่าถ้าต้องการแสดงคอลัมน์ทั้งหมดภายในตารางสำหรับทุกๆ แถวจะใช้เครื่องหมาย (*) เป็นการย่อพิเศษสำหรับระบุคอลัมน์ทั้งหมดได้ ดังนั้นประโยคคำสั่งนี้มีรูปแบบดังนี้

SELECT * FROM table_name

ยกตัวอย่างเช่น **SELECT * FROM [Flight Instance]** เป็นการเรียกดูคอลัมน์ทั้งหมดจากตาราง **Flight Instance** ซึ่งได้ผลดังตารางที่ 2.9

Flight number	Date	available seats
TG401	16/3/2005	100
TG402	16/3/2005	50
TG601	17/3/2005	25
TG603	17/3/2005	20

ตารางที่ 2.8 ผลการคิวรีเมื่อระบุคอลัมน์ที่จะเรียกดูโดยใช้ประโยคคำสั่ง **SELECT**

Flight number	Date	Plane number	available seats
TG401	16/3/2005	777	100
TG402	16/3/2005	333	50
TG601	17/3/2005	773	25
TG603	17/3/2005	773	20

ตารางที่ 2.9 ผลการคิวรีเมื่อต้องการเรียกดูคอลัมน์ทั้งหมดโดยใช้ประโยคคำสั่ง **SELECT**

2. การเรียกดูข้อมูลแบบมีเงื่อนไข เป็นการจำกัดข้อมูลที่จะแสดงออกมาเพื่อให้ได้เฉพาะข้อมูลที่อยู่ในเงื่อนไขที่ต้องการออกมา โดยการใช้งานจะใช้ร่วมกับคำสั่ง **WHERE** มีรูปแบบประโยคคือ

SELECT column_name1, column_name2,..... FROM table_name WHERE condition

เมื่อ **WHERE condition** ใช้เป็นเงื่อนไขในการแสดงข้อมูล ยกตัวอย่างเช่น **SELECT [Flight number], Date, [available seats] FROM [Flight Instance] WHERE [available seats] > 50** จะคืนค่าในคอลัมน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flight number, Date และ available seats จากตาราง Flight Instance โดยมีเงื่อนไขว่าค่าในคอลัมน์ available seats มีค่ามากกว่า 50 เท่านั้น ผลที่ได้แสดงดังตาราง 2.10

Flight number	Date	available seats
TG401	16/3/2005	100
TG402	16/3/2005	50

ตารางที่ 2.10 ผลการคิวรีเมื่อเรียกดูข้อมูลแบบมีเงื่อนไขโดยใช้ประโยคคำสั่ง SELECT

2.5.2.2 ประโยคย่อย FROM

มักใช้ร่วมกับประโยคคำสั่ง SELECT และ ประโยคคำสั่ง DELETE เราจะใช้ประโยคย่อย FROM เพื่อแสดงถึงการเรียกดูมาจากตารางฐานข้อมูลตัวไหน หรือว่าต้องการจะลบข้อมูลจากตารางไหน ยกตัวอย่างเช่น DELETE FROM Flight Instance WHERE ([Flight number]='TG401') (DAY([Date]) = 16) AND (MONTH([Date]) =3) AND (YEAR([Date])=2005) AND ([available seats]=50) AND ([Plane number]=777) เป็นการลบแถวในตาราง Flight Instance เมื่อค่า Flight number = TG401, Date = 16/3/2005, Plane number = 777, available seats = 50

2.5.2.3 ประโยคย่อย WHERE

ในการใช้ประโยคย่อย WHERE นั้น สามารถระบุกลุ่มย่อยของแถวที่จะถูกคืนค่ากลับมาได้ดังตัวอย่างข้างต้น ดังนั้นรูปแบบของประโยคย่อย WHERE คือ

WHERE <column> <operator> <value>

จะเห็นว่าประโยคย่อย WHERE จะใช้ร่วมกับโอเปอเรเตอร์ (operator) ใน SQL ซึ่งใน SQL Server มีโอเปอเรเตอร์ในการเปรียบเทียบทั้งหมดดังแสดงในตารางที่ 2.11

โอเปอเรเตอร์	ความหมาย
=	เท่ากับ
>	มากกว่า
<	น้อยกว่า
>=	มากกว่าหรือเท่ากับ
<=	น้อยกว่าหรือเท่ากับ
<>	ไม่เท่ากับ

ตารางที่ 2.11 โอเปอเรเตอร์ใน SQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2.4 ประโยคย่อย ORDER BY

เป็นประโยคองค์ประกอบที่เป็นตัวเลือกของประโยคคำสั่ง SELECT ซึ่งยอมให้ระบุลำดับของแถวที่จะถูกเรียกขึ้นมาได้ โดยสามารถระบุคอลัมน์หลายคอลัมน์และเรียกดูแถวจากลำดับมากไปหาน้อยหรือน้อยไปหามากได้ รูปแบบที่ง่ายที่สุดของประโยคย่อย ORDER BY คือใส่ชื่อคอลัมน์ที่จะใช้ในการเรียงลำดับแถวที่จะถูกเรียกดูจากคิวรีเพียงคอลัมน์เดียว ยกตัวอย่างเช่น `SELECT * FROM [Flight schedule] ORDER BY [Departure time]` เป็นการคืนค่าทุกคอลัมน์จากตาราง Flight schedule โดยแสดงผลเรียงลำดับตามค่าเวลาในคอลัมน์ Departure time น้อยไปมาก ผลที่ได้ดังตารางที่ 2.12

Flight number	From airportcode	To airportcode	Departure time	Arrival time
TG402	SIN	BKK	08:20	09:40
TG401	BKK	SIN	19:15	22:35
TG409	BKK	SIN	15:15	18:35

ตารางที่ 2.12 ผลการคิวรีเมื่อต้องการเรียกดูคอลัมน์โดยใช้ร่วมกับประโยคย่อย ORDER BY

2.5.2.5 ประโยคคำสั่ง INSERT

ใช้สำหรับเพิ่มแถวให้ตารางมีรูปแบบประโยคคำสั่งการใช้งานดังนี้คือ

`INSERT INTO table_name (column_name1, column_name2,) VALUES (value1,value2,value3,.....)`

เช่น `INSERT INTO Airport([Airport Code], [Airport Name], [Airport City], [Airport Country], [Airport ID]) VALUES (SIN, Changi , Singapore, Singapore, 4)` เป็นการเพิ่มค่าในฐานข้อมูลของตาราง Airport ที่คอลัมน์ต่างๆ คือ Airport Code, Airport Name, Airport City, Airport Country, และ Airport ID โดยค่าที่เพิ่มเข้าไปคือ SIN , Changi, Singapore, Singapore และ 4 ตามลำดับของคอลัมน์ ผลที่ได้ดังตารางที่ 2.13 เมื่อเทียบกับตารางที่ 2.7

Airport Code	Airport Name	Airport City	Airport Country	Airport ID
BKK	Don Muang Int'l	Bangkok	Thailand	1
HKG	International	Hong Kong	China	2
ICN	Incheon Int'l	Seoul	Republic of Korea	3
SIN	Changi	Singapore	Singapore	4

ตารางที่ 2.13 ผลการคิวรีเมื่อใช้คำสั่ง INSERT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2.6 ประโยคคำสั่ง UPDATE

เมื่อต้องการเปลี่ยนแปลงหรือแก้ไขข้อมูลในคอลัมน์ของแต่ละแถวแล้ว สามารถทำได้โดยอาศัยคีย์เวิร์ด UPDATE และ SET ซึ่งรูปแบบการใช้งานเป็นดังนี้คือ

```
UPDATE table_name SET [ column_name1= value1, column_name2= value2, .....]
```

WHERE (condition)

ประโยคคำสั่ง UPDATE จะตามด้วยชื่อตารางที่ต้องการแก้ไขข้อมูลและคีย์เวิร์ด SET จะตามด้วยคอลัมน์และค่าที่ต้องการปรับปรุงใหม่ ประโยคย่อย WHERE เป็นตัวเลือก เมื่อใส่ประโยคย่อยนี้ลงไปจะเป็นการระบุแถวที่ต้องการแก้ไข ถ้าไม่มีประโยคเงื่อนไข WHERE อยู่ในประโยคคำสั่ง UPDATE จะทำให้แถวทั้งหมดในตารางได้รับการปรับปรุง เช่น UPDATE [Flight Instance] SET([available seats] = 95) เป็นการแก้ไขค่าในตาราง [Flight Instance] ในคอลัมน์ [available seats] ถ้าค่าเก่าในคอลัมน์เป็น 100 ให้มีค่าใหม่เป็น 95 สำหรับทุกแถว แต่ถ้าใช้ประโยคย่อย WHERE ร่วมด้วย เช่น UPDATE [Flight Instance] SET [available seats] = 95 WHERE [Flight number] = 'TG409' เป็นการแก้ไขค่าในตาราง [Flight Instance] ในคอลัมน์ [available seats] ให้มีค่าใหม่เป็น 95 เฉพาะแถวที่ค่าในคอลัมน์ [Flight number] เท่ากับ TG409 เท่านั้นส่วนค่าในคอลัมน์ [Flight number] แถวอื่นจะมีค่าเท่าเดิม

Flight number	Date	Plane number	available seats
TG401	16/3/2005	777	95
TG402	16/3/2005	333	95
TG409	17/3/2005	773	95
TG601	17/3/2005	773	95

ตารางที่ 2.14 ผลการคิวรีเมื่อใช้คำสั่ง UPDATE

Flight number	Date	Plane number	available seats
TG401	16/3/2005	777	100
TG402	16/3/2005	333	100
TG409	17/3/2005	773	95
TG601	17/3/2005	773	100

ตารางที่ 2.15 ผลการคิวรีเมื่อใช้คำสั่ง UPDATE ร่วมกับประโยคย่อย WHERE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2.7 ประโยคคำสั่ง DELETE

คือประโยคคำสั่งที่ใช้สำหรับการลบแถวจากตารางข้อมูล โดยมีรูปแบบประโยคคำสั่งดังนี้

DELETE FROM *table_name* **WHERE** *where_condition*

ประโยคคำสั่ง DELETE จะยอมให้ลบแถวที่เลือกไว้เท่านั้นโดยขึ้นอยู่กับประโยคย่อย WHERE แต่ถ้าไม่ใส่ประโยคย่อย WHERE จะทำให้แถวในตารางข้อมูลที่ถูกระบุไว้ถูกลบออกทั้งหมด แต่ถ้าใช้ประโยคย่อย WHERE ร่วมด้วย เช่น DELETE FROM [Flight Instance] WHERE Date = '3/1/2005' เป็นการลบค่าของแถวในตารางข้อมูล [Flight Instance] ที่ซึ่งค่าในคอลัมน์ Date มีค่าเท่ากับ 16/3/2005 เท่านั้น จากตารางที่ 2.14 จะได้ผลการควิรี่เมื่อใช้คำสั่ง DELETE ดังตารางที่ 2.16

Flight number	Date	Plane number	available seats
TG409	17/3/2005	773	100
TG601	17/3/2005	773	100

ตารางที่ 2.16 ผลการควิรี่เมื่อใช้คำสั่ง DELETE ร่วมกับประโยคย่อย WHERE

2.5.2.8 การรวมข้อมูลในแต่ละคอลัมน์จากหลายตาราง (Merging Data from Multiple Table)

ปกติผู้ออกแบบฐานข้อมูล จะออกแบบฐานข้อมูลโดยแยกความสัมพันธ์ของข้อมูลไว้ในแต่ละตาราง เพื่อให้แน่ใจว่าไม่มีการซ้ำซ้อนของข้อมูลซึ่งถูกจัดเก็บในแต่ละตาราง พิจารณาจากตารางที่ 2.5 และ 2.6 ซึ่งประกอบด้วยข้อมูลจากตาราง Flight Instance และ Flight schedule ตามลำดับ โดยทั้งสองตารางจะสัมพันธ์กันด้วยค่าในคอลัมน์ Flight number ในบางครั้งมีความจำเป็นที่จะต้องรวมข้อมูลจากหลายตารางแล้วทำการแสดงผลออกมาทีเดียว (Single result) ซึ่งกล่าวได้ว่าเป็นการ Join กันของแต่ละตาราง มีรูปแบบของคำสั่งดังนี้

SELECT *column_name1, column_name2,.....* **FROM** *table1* **INNER JOIN** *table2*
ON *table1.column_name = table2.column_name*

ในคำสั่ง SELECT ถ้าชื่อของคอลัมน์ในแต่ละตาราง มีข้อมูลเหมือนกันต้องกำหนดชื่อของตารางข้อมูลตามด้วยจุดหน้าชื่อคอลัมน์นั้นๆ เพื่อให้ทราบว่าเป็นคอลัมน์ของตารางข้อมูลใด โดย INNER JOIN เป็นคำสั่งที่ใช้รวมแถวจาก 2 ตาราง โดยทั้งสองตารางจะต้องมีค่าของข้อมูลในคอลัมน์ใดคอลัมน์หนึ่งเหมือนกัน สำหรับคีย์เวิร์ด ON จะตามด้วยชื่อของคอลัมน์จากแต่ละตารางเพื่อใช้คำนวณในรวมแถวระหว่างสองตาราง ยกตัวอย่างคำสั่งต่อไปนี้

SELECT [Flight Instance.Flight number], Date, [available seats], [From airportcode],
[To airportcode], [Departure time], [Arrival time] **FROM** [Flight Instance] **INNER JOIN** [Flight
schedule] **ON** [Flight Instance].[Flight number] = [Flight schedule].[Flight number]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

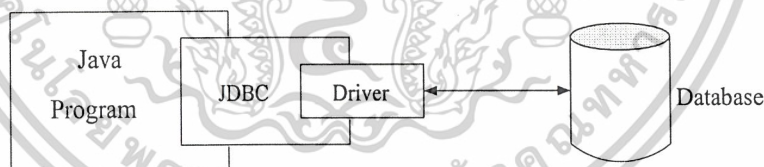
จากคำสั่งดังกล่าวเป็นการรวมข้อมูลในคอลัมน์ Flight number, Date, available seats จากตาราง Flight Instance กับข้อมูลในคอลัมน์ From airportcode, To airportcode, Departure time, Arrival time จากตาราง Flight schedule โดยใช้ค่าในคอลัมน์ Flight number เป็นเงื่อนไขในการรวมแถว ซึ่งแถวจากทั้งสองตารางจะรวมกันได้จะต้องมีค่าในคอลัมน์ Flight number เหมือนกัน ผลจากคำสั่งข้างต้นแสดงไว้ดังตารางที่ 2.14

Flight number	Date	available seats	From airport code	To air port code	Departure time	Arrival time
TG401	16/3/2005	100	BKK	SIN	19:15	22:35
TG402	16/3/2005	50	SIN	BKK	08:20	09:40

ตารางที่ 2.17 ผลการรวมข้อมูลระหว่างตาราง Flight Instance กับ Flight schedule

2.6 JDBC

ในการเขียนโปรแกรมโดยใช้ภาษาจาวาติดต่อกับฐานข้อมูลจะต้องอาศัยไดรเวอร์ (Driver) เป็นตัวกลางที่ใช้ติดต่อระหว่างภาษาจาวากับฐานข้อมูลนั้นๆ ไดรเวอร์ที่ใช้ชื่อว่า JDBC ภายใน JDBC API ประกอบไปด้วยคลาสต่างๆที่จำเป็นต้องใช้ในการติดต่อกับฐานข้อมูล เช่น มีคลาสที่สนับสนุนการให้สามารถใช้คำสั่ง SQL จัดการกับฐานข้อมูลผ่านทางภาษาจาวา ได้โดยตรง การติดต่อระหว่างโปรแกรมกับระบบฐานข้อมูลแสดงไว้ดังรูป 2.16



รูปที่ 2.16 การติดต่อระหว่างโปรแกรมกับระบบฐานข้อมูล

โปรแกรมภาษาจาวาจะส่งคำสั่งสำหรับติดต่อกับระบบฐานข้อมูลเป็น SQL ไปที่ JDBC แล้ว JDBC จะเปลี่ยนคำสั่งนั้นเป็นคำสั่งสำหรับไดรเวอร์ โดยที่ระบบฐานข้อมูลนั้นจะมีไดรเวอร์อยู่แล้วโดยไดรเวอร์จะแปลคำสั่งจาก JDBC เป็นคำสั่งสำหรับเข้าไปใช้งานระบบฐานข้อมูลต่อไป

2.6.1 การเขียนโปรแกรมติดต่อกับฐานข้อมูลโดยใช้ JDBC

ในการเขียนโปรแกรมติดต่อกับฐานข้อมูลนั้นสามารถแบ่งขั้นตอนการทำงานออกเป็นขั้นตอนหลักๆ ได้ 4 ขั้นตอนดังนี้ คือ

1. สร้างการติดต่อกับฐานข้อมูล
2. ดำเนินการกับฐานข้อมูล
3. แสดงผลลัพธ์ให้กับผู้ใช้หรือการสอบถามข้อมูล
4. ยกเลิกการติดต่อกับฐานข้อมูล

2.6.1.1 สร้างการติดต่อกับฐานข้อมูล

มี 2 ขั้นตอนคือ

1. โหลดไดรเวอร์ขึ้นมาทำงาน

ในขั้นตอนแรกนั้นเราต้องโหลดไดรเวอร์ที่ต้องการใช้ติดต่อกับฐานข้อมูลขึ้นมาให้ได้เสียก่อน ถ้าหากขั้นตอนนี้มีความผิดพลาดเกิดขึ้นการทำงานในขั้นต่อไปจะไม่สามารถดำเนินการต่อไปได้ สามารถทำได้โดยเรียกใช้ static Method “forName()” ซึ่งมีรูปแบบดังนี้

```
Class.forName(drivename)
```

โดยที่ drivename หมายถึงชื่อไดรเวอร์ที่ต้องการโหลดขึ้นมาใช้งานเพื่อใช้ติดต่อกับฐานข้อมูลและชื่อไดรเวอร์ที่ใช้ในปฏิญญาพันธคือ "sun.jdbc.odbc.JdbcOdbcDriver" ซึ่งเป็นไดรเวอร์ที่ใช้บนระบบปฏิบัติการวินโดวส์ (Windows)

2. เชื่อมต่อกับฐานข้อมูล

สามารถสร้างการเชื่อมต่อโดยใช้ Method “getConnection()” ซึ่งมีรูปแบบการเรียกใช้ดังนี้

```
connection = DriverManager.getConnection(url)
```

โดยที่ url คือข้อมูลที่ใช้ระบุเส้นทางไปยังฐานข้อมูลที่ต้องการจะติดต่อซึ่งในปฏิญญาพันธใช้

```
url = "jdbc:odbc:ReserveTicket"
```

เมื่อ jdbc หมายถึง ส่วนที่บ่งบอกว่าการติดต่อกับฐานข้อมูลเป็นแบบ JDBC

odbc ใช้สำหรับการสื่อสารผ่าน ODBC ของวินโดวส์

ReserveTicket หมายถึง ชื่อที่ใช้อ้างอิงกับฐานข้อมูลที่ต้องการติดต่อผ่าน odbc

แล้ว ReserveTicket ก็คือชื่อ Data Source

2.6.1.2 ดำเนินการกับฐานข้อมูล

เมื่อเราทำการเชื่อมต่อกับฐานข้อมูลแล้วเราอาจจะต้องการดำเนินการกับฐานข้อมูล อย่างไม่อย่างหนึ่งด้วยคำสั่ง SQL ถ้าเราเลือกใช้คำสั่ง CREATE,INSERT,UPDATE,DELETE การทำงานใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะนี้จะต้องทำงานผ่าน Method “executeUpdate()” ของ Statement Object รูปแบบการใช้งาน Method “executeUpdate()” ของ Class “Statement” ดังนี้คือ

```
Statement statement = connection.createStatement() // ใช้สร้าง Statement Object
Statement.executeUpdate(SQL) // ส่งคำสั่ง SQL ให้กับระบบฐานข้อมูล
```

โดยที่ connection คือ Object ของ Class “Connection” ที่สร้างขึ้นตอนติดต่อกับฐานข้อมูล createStatement() คือเมธอดที่ใช้สร้าง Statement Object ซึ่งเมธอดนี้จะคืนค่ากลับมาเป็น reference ของ Class “Statement”

executeUpdate คือเมธอดที่ใช้ส่งคำสั่ง SQL ที่ระบุไว้ในพารามิเตอร์ (parameter) ไปดำเนินการกับฐานข้อมูล

SQL คือ คำสั่ง SQL ที่ต้องการดำเนินการกับฐานข้อมูล

Statement คือ อินเตอร์เฟสสำหรับสร้างคลาสที่ทำหน้าที่ส่งประโยค SQL ไปสู่ระบบฐานข้อมูล

2.6.1.3 แสดงผลลัพธ์ให้กับผู้ใช้หรือการสอบถามข้อมูล

ในการแสดงผลลัพธ์ให้กับผู้ใช้หรือการสอบถามข้อมูลไปยังฐานข้อมูลจะเลือกใช้คำสั่ง SQL คือ SELECT กระทำผ่าน Method “executeQuery()” ซึ่งมีอยู่ใน Class “Statement” รูปแบบการใช้งาน Method “executeQuery()” ของ Class “Statement” มีดังนี้คือ

```
Statement statement = connection.createStatement() // ใช้สร้าง Statement Object
Statement.executeQuery(SQL) // ส่งคำสั่ง SQL ให้กับระบบฐานข้อมูล
```

จะเห็นว่ารูปแบบการใช้งานของการแสดงผลลัพธ์ให้กับผู้ใช้หรือการสอบถามข้อมูลจะคล้ายกับรูปแบบการใช้งานของดำเนินการกับฐานข้อมูลโดยที่เปลี่ยนรูปแบบคำสั่งจาก

```
Statement.executeUpdate(SQL) มาเป็น Statement.executeQuery(SQL)
```

โดยที่

Statement.executeQuery(SQL) คือเมธอดที่ใช้ส่งคำสั่ง SQL ที่ระบุไว้ในพารามิเตอร์ไปดำเนินการกับฐานข้อมูล

2.6.1.4 ยกเลิกการติดต่อกับฐานข้อมูล

การยกเลิกการติดต่อกับฐานข้อมูล ควรจะทำเมื่อไม่ต้องการใช้งานกับฐานข้อมูลนั้นอีกแล้ว เพราะถ้าเมื่อใส่คำสั่งนี้ไปแล้วจะเป็นการสั่งยกเลิกการติดต่อกับฐานข้อมูลทันทีแล้ว โปรแกรมจะไม่สามารถติดต่อกับฐานข้อมูลได้อีกจนกว่าจะได้รับการติดต่อใหม่อีกครั้งหนึ่ง สามารถทำได้โดยเรียกใช้ Method “close()” ซึ่งเมธอดนี้มีอยู่ใน Class “Connection”, “Statement” มีรูปแบบการใช้งานดังนี้ คือ

```
connection.close()
```

คือการยกเลิกการติดต่อกับ “connection” ของ Class “Connection”

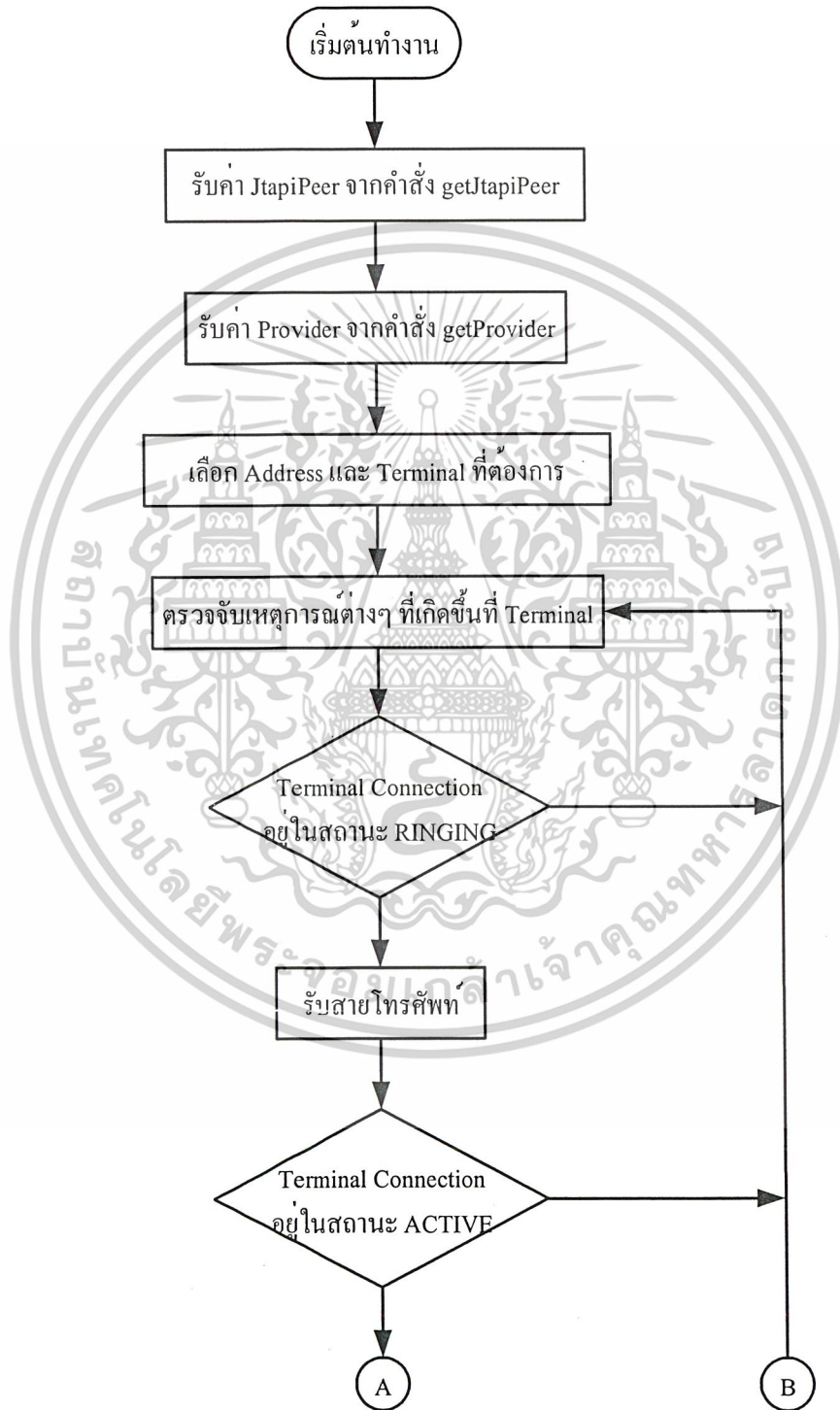


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

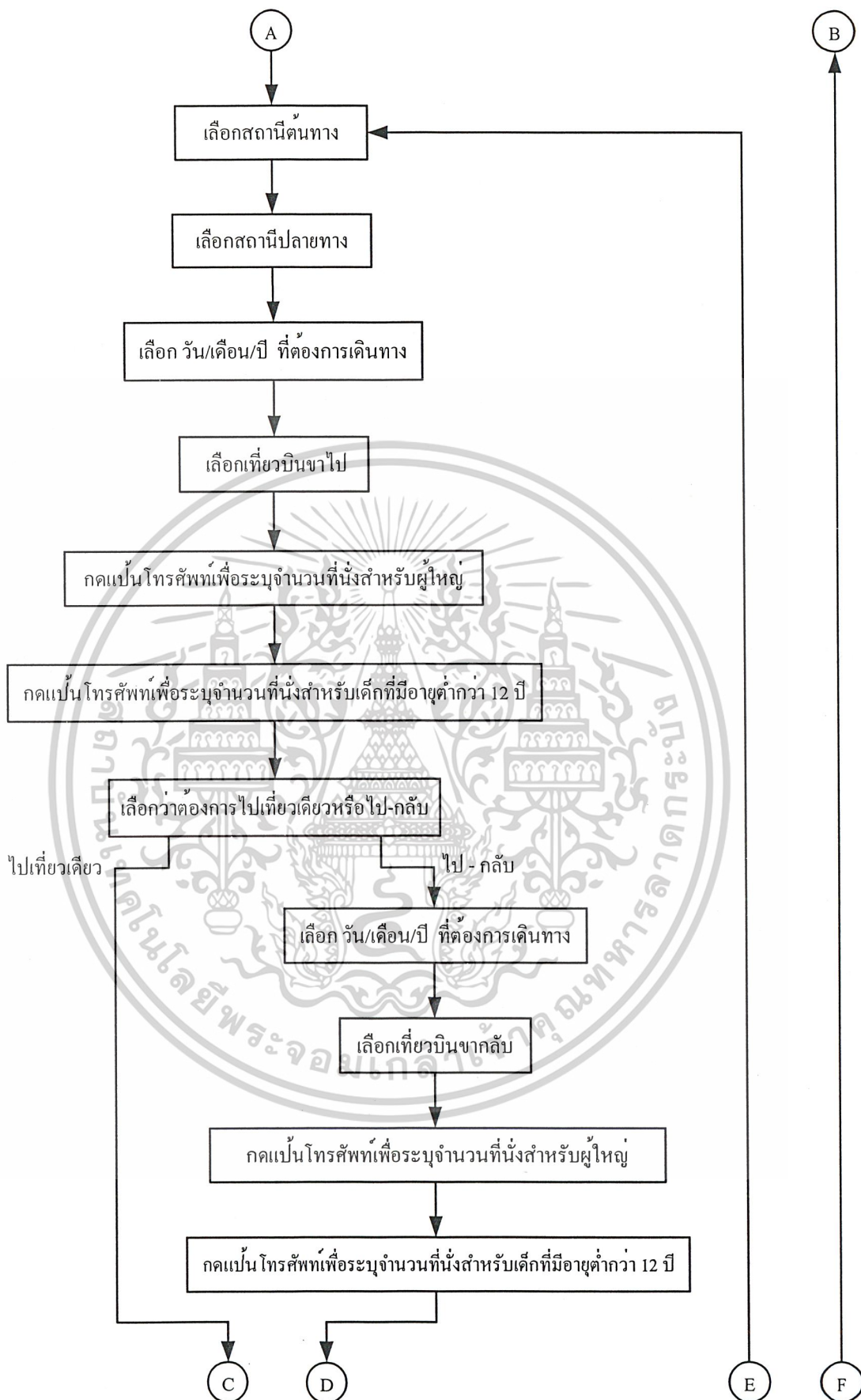
บทที่ 3
การออกแบบ

3.1 หลักการทำงานของระบบ

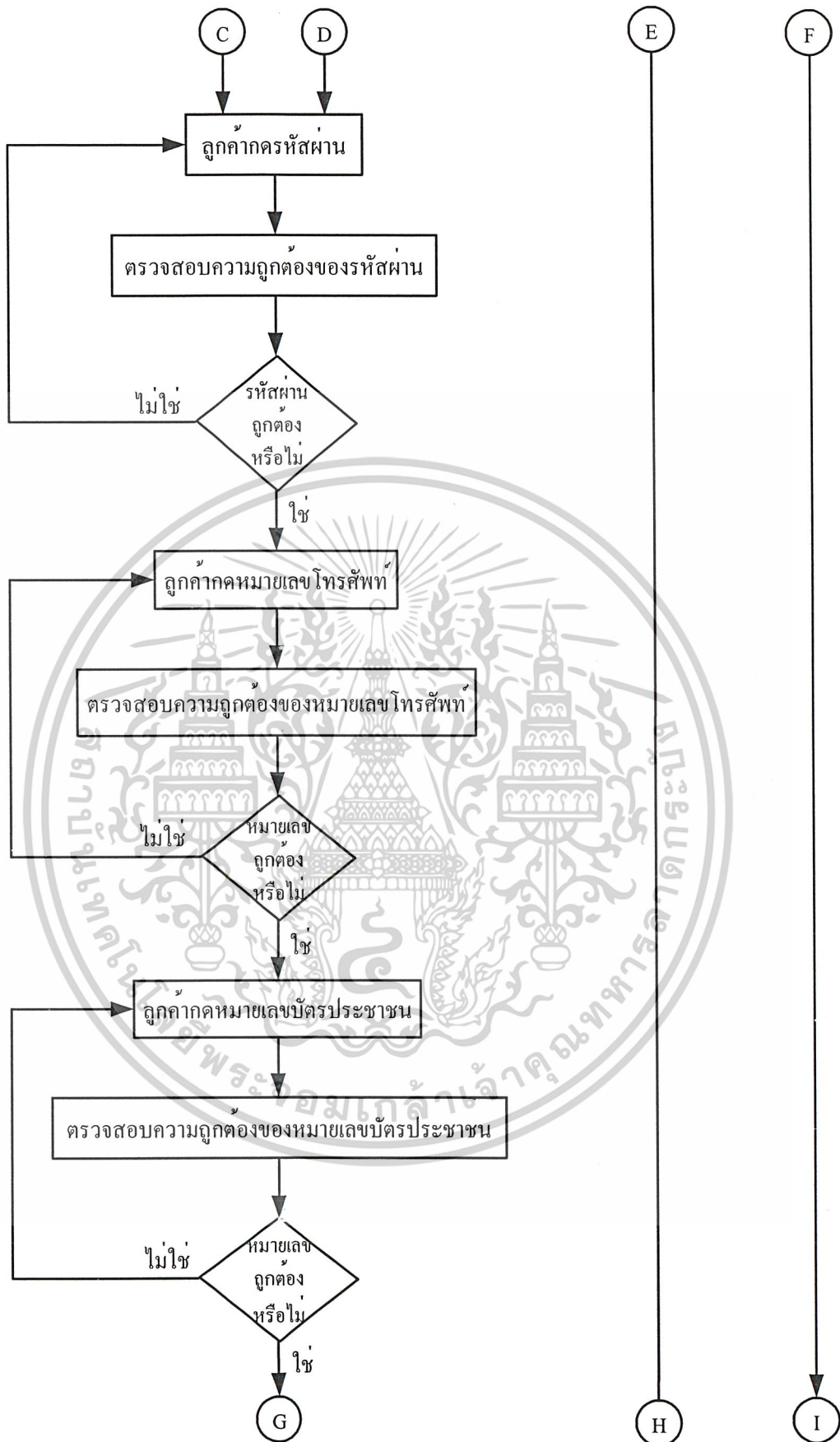
หลักการทำงานของระบบของตัวเครื่องบินมีขั้นตอนการทำงานดังนี้คือ



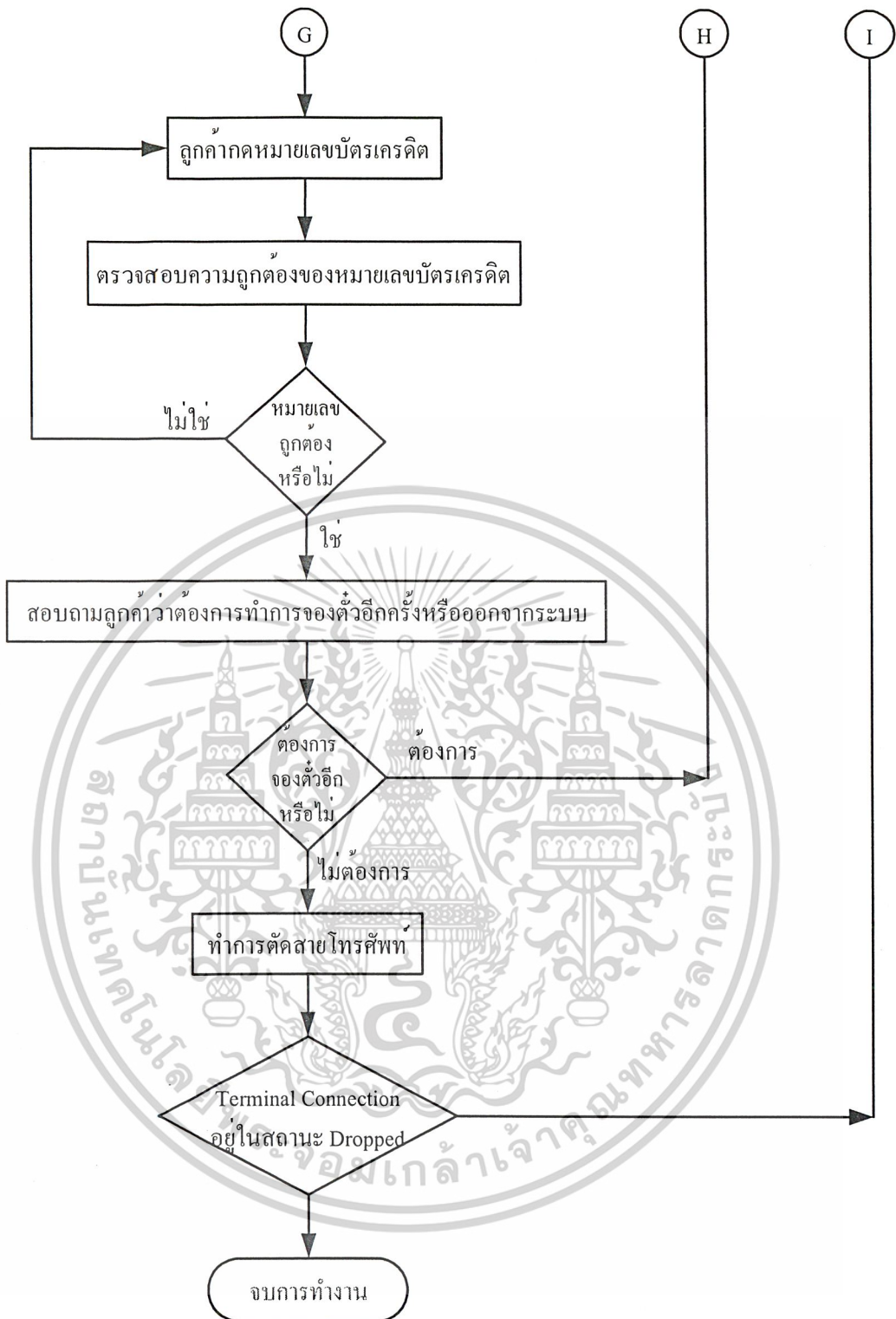
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 ขั้นตอนการทำงานของระบบจองตัวเครื่องบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายหลักการการทำงานของระบบ

จากรูป 3.1 เป็นระบบการจองตั๋วเครื่องบินของสายการบินหนึ่ง มีขั้นตอนการจองตั๋วเครื่องบินเป็นดังนี้

1. เลือกสถานีต้นทาง โดยระบบจะเล่นไฟล์เสียงอัตโนมัติซึ่งมีรายละเอียดในการเลือกสถานีต้นทาง ยกตัวอย่างเช่น กด “1” เพื่อเลือก กรุงเทพฯ กด “2” เพื่อเลือกฮ่องกง เป็นต้น
2. เลือกสถานีปลายทาง โดยระบบจะเล่นไฟล์เสียงอัตโนมัติซึ่งมีรายละเอียดในการเลือกสถานีปลายทาง ยกตัวอย่างเช่น กด “3” เพื่อเลือกเกาหลี กด “4” เพื่อเลือกสิงคโปร์ เป็นต้น
3. เลือก วัน/เดือน/ปี ที่ต้องการออกเดินทาง โดยระบบจะเล่นไฟล์เสียงอัตโนมัติเพื่อออกรายละเอียดต่างๆ โดยอันดับแรกจะให้ผู้ใช้บริการเลือกว่าวันที่ต้องการออกเดินทาง โดยระบุวันที่ต้องการเดินทางจากการกดหมายเลขที่อยู่บนแป้นโทรศัพท์ ต่อจากนั้นทำการเลือกเดือนระบบจะเล่นไฟล์เสียงอัตโนมัติ เพื่อให้ผู้ใช้บริการทำการระบุเดือนที่ต้องการเดินทางเป็นตัวเลขจากการกดแป้นโทรศัพท์และสุดท้ายเป็นการเลือกปีที่ต้องการเดินทาง ซึ่งในระบบนี้จะรับค่าเป็น ค.ศ เท่านั้น โดยจะใช้วิธีการกดหมายเลขจากแป้นโทรศัพท์เพื่อระบุรายละเอียดดังกล่าวเช่นกัน ยกตัวอย่าง ถ้าต้องการออกเดินทางในวันที่ 1/5/2005 จะต้องกดหมายเลขจากแป้นโทรศัพท์ดังนี้ เลือกวันที่ต้องการโดยการกด “1” เลือกเดือนที่ต้องการโดยการกด “5” เลือกปีที่ต้องการโดยการกด “2” “0” “0” “5” ตามลำดับ
4. เลือกเที่ยวบินขาไป โดยที่ระบบจะทำการเล่นไฟล์เสียงอัตโนมัติเพื่อออกรายละเอียดต่างๆ ยกตัวอย่างเช่น กด “1” เพื่อเลือก เที่ยวบิน TG403 กรุงเทพฯ - สิงคโปร์ เวลา 8.00 – 11.15 น. เป็นต้น
5. ระบุจำนวนที่นั่งซึ่งจะแยกเป็น 2 กรณี
 1. เลือกจำนวนที่นั่งสำหรับผู้ใหญ่ โดยระบุจำนวนจากการกดหมายเลขบนแป้นโทรศัพท์ เช่นกัน ยกตัวอย่าง มีผู้ใหญ่ 4 คน ให้กด “4” ถ้าไม่มีกด “0”
 2. เลือกจำนวนที่นั่งสำหรับเด็กอายุต่ำกว่า 12 ปี โดยระบบจะเล่นไฟล์เสียงอัตโนมัติเพื่อออกรายละเอียดต่างๆ โดยระบุจำนวนจากการกดหมายเลขบนแป้นโทรศัพท์ เช่น มีเด็กจำนวน 2 คนก็กด “2” ถ้าไม่มีกด “0”
6. เลือกจะไปเที่ยวเดียวหรือไปกลับโดยระบบจะเล่นไฟล์เสียงอัตโนมัติ เพื่อออกรายละเอียดต่างๆ ในกรณีที่เลือกไปกลับ จะกลับไปดำเนินการตามขั้นตอนในข้อ 3 - 5 หลังจากนั้นจะทำการเลือกเที่ยวบินขากลับ แล้วจึงเข้าสู่ขั้นตอนในข้อที่ 7 หากเลือกไปเพียงอย่างเดียวจะเข้าสู่ขั้นตอนในข้อที่ 7 ทันที
7. ให้ลูกค้ากรอกรหัสผ่าน 4 ตัว เพื่อใช้ในการยืนยันเมื่อมารับบัตรโดยสารรหัสผ่านหมายเลขที่อยู่บนแป้นโทรศัพท์ จากนั้นระบบจะทำการตรวจสอบว่าลูกค้ากรอกรหัสผ่านครบทั้ง 4 ตัวหรือไม่ ถ้าไม่ครบลูกค้าจะต้องป้อนให้ครบทั้ง 4 ตัว จากนั้นก็ให้ลูกค้ายืนยันรหัสผ่านอีกครั้งว่าถูกต้องหรือไม่ ถ้าใช่กด “1” เพื่อเข้าสู่ขั้นตอนต่อไป ไม่ใช่กด “2” เพื่อให้ลูกค้ากรอกรหัสผ่านใหม่อีกครั้ง
8. ให้ลูกค้ากดหมายเลขโทรศัพท์ 9 ตัว เพื่อใช้เป็นข้อมูลของลูกค้าโดยกดจากหมายเลขที่อยู่บนแป้นโทรศัพท์ จากนั้นระบบจะทำการตรวจสอบว่าลูกค้ากดหมายเลขโทรศัพท์ครบทั้ง 9 ตัวหรือไม่ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ครบลูกค้าจะต้องป้อนให้ครบทั้ง 9 ตัว จากนั้นก็ให้ลูกค้ายืนยันหมายเลขโทรศัพท์อีกครั้งว่าถูกต้องหรือไม่ ถ้าใช่กด “1” เพื่อเข้าสู่ขั้นตอนต่อไป ไม่ใช่กด “2” เพื่อให้ลูกค้ากดหมายเลขโทรศัพท์อีกครั้ง

9. ให้ลูกค้ากดเลขประจำตัวบัตรประชาชน 13 ตัว เพื่อใช้เป็นข้อมูลของลูกค้า โดยกดจากหมายเลขที่อยู่บนเป็นโทรศัพท์ จากนั้นระบบจะทำการตรวจสอบว่าลูกค้า กดเลขประจำตัวบัตรประชาชนครบทั้ง 13 ตัวหรือไม่ ถ้าไม่ครบลูกค้าจะต้องป้อนให้ครบทั้ง 13 ตัว จากนั้นก็ให้ลูกค้ายืนยันหมายเลขบัตรอีกครั้งว่าถูกต้องหรือไม่ ถ้าใช่กด “1” เพื่อเข้าสู่ขั้นตอนต่อไป ไม่ใช่กด “2” เพื่อให้ลูกค้ากดหมายเลขประจำตัวบัตรประชาชนอีกครั้ง

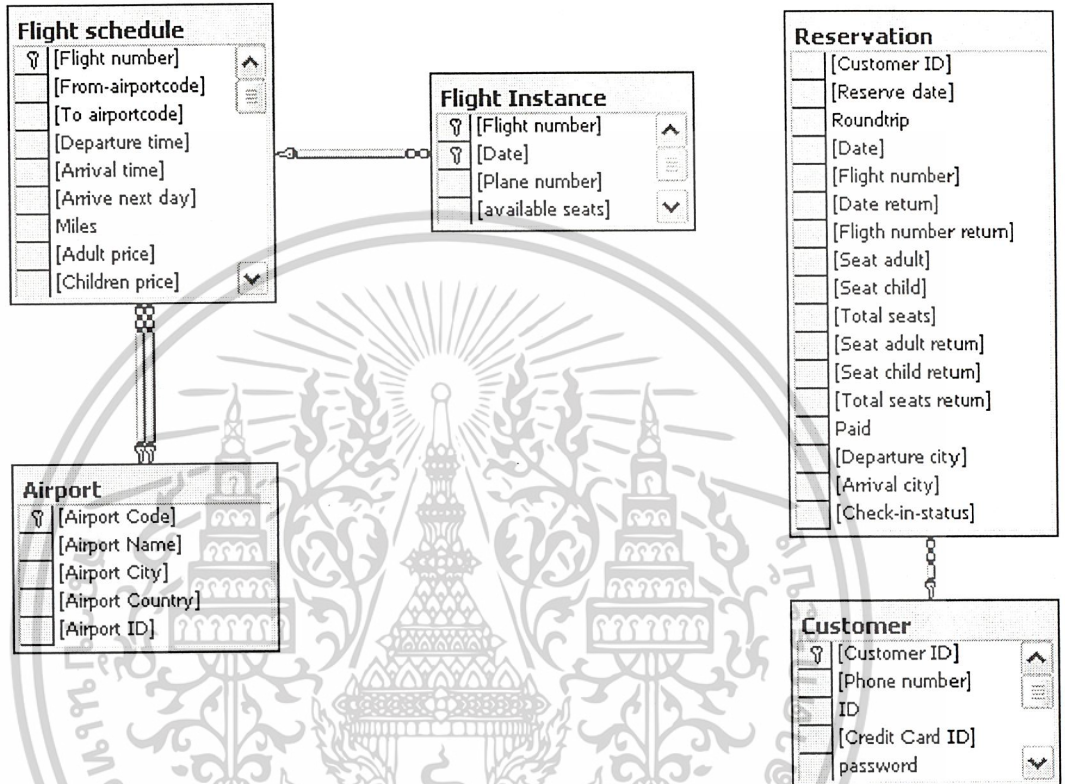
10. ให้ลูกค้ากดหมายเลขบัตรเครดิต 16 ตัว เพื่อใช้เป็นข้อมูลของลูกค้า โดยกดจากหมายเลขที่อยู่บนเป็นโทรศัพท์ จากนั้นระบบจะทำการตรวจสอบว่าลูกค้ากดหมายเลขบัตรเครดิตครบทั้ง 16 ตัวหรือไม่ ถ้าไม่ครบลูกค้าจะต้องป้อนให้ครบทั้ง 16 ตัว จากนั้นก็ให้ลูกค้ายืนยันหมายเลขบัตรอีกครั้งว่าถูกต้องหรือไม่ ถ้าใช่กด “1” เพื่อเข้าสู่ขั้นตอนต่อไป ไม่ใช่กด “2” เพื่อให้ลูกค้ากดหมายเลขบัตรเครดิตอีกครั้ง

11. ระบบจะเล่นเสียงอัตโนมัติซึ่งจะมีรายละเอียดดังนี้คือ กด “1” เพื่อทำการจองตัวเครื่องบินใหม่อีกครั้ง ดังนั้นระบบจะกลับไปทำในขั้นตอนที่ 1 - 11 ใหม่ แต่ถ้าลูกค้าไม่ต้องการจองตัวใหม่อีกครั้ง ให้ กด “2” เพื่อทำการออกจากระบบ

12. เมื่อลูกค้าทำการออกจากระบบแล้ว โปรแกรมจะทำการตัดสายโทรศัพท์เพื่อที่จะรอรับการเรียกเข้ามาของลูกค้าคนต่อไป

3.2 การออกแบบระบบฐานข้อมูล

รูปที่ 3.2 เป็นระบบฐานข้อมูลของระบบการจองตั๋วเครื่องบินที่ถูกจำลองขึ้น ซึ่งประกอบด้วยกลุ่มของตารางทั้งหมด 5 ตารางด้วยกัน ได้แก่ ตาราง Airport, ตาราง Flight schedule, ตาราง Flight instance, ตาราง Reservation และตาราง Customer โดยมีโครงสร้างของตารางและความสัมพันธ์ระหว่างตารางดังรูปที่ 3.2 และรายละเอียดของข้อมูลในฐานข้อมูลได้แสดงไว้ในตารางที่ 3.1 ตามลำดับ



รูปที่ 3.2 ฐานข้อมูลของระบบจองตั๋วเครื่องบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตาราง	ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย
Airport	Airport Code	char (5)	รหัสของสนามบิน
	Airport Name	varchar (18)	ชื่อของสนามบิน
	Airport City	varchar (18)	เมืองหลวงของสนามบิน
	Airport Country	varchar (18)	ประเทศของสนามบิน
	Airport ID	varchar (4)	ลำดับของสนามบิน
Flight schedule	Flight number	char (5)	ชื่อสายการบิน
	From-airportcode	Airport-codeChar(3)	รหัสสนามบินที่ออกเดินทาง
	To-airportcode	Airport-codeChar(3)	รหัสสนามบินที่เดินทางไปถึง
	Departure time	char (5)	เวลาที่ออกเดินทาง
	Arrival time	char (5)	เวลาที่เดินทางไปถึง
	Arrival next day	smallint (2)	แสดงค่าเป็น 1 เมื่อถึงปลายทางในวันถัดไป
	Miles	smallint (2)	ระยะทางที่ใช้เดินทาง
	Adult price	decimal 5(7,2)	ราคาตั๋วผู้ใหญ่
	Children price	decimal 5(7,2)	ราคาตั๋วเด็ก
Flight instance	Flight number	FlightnumberChar(5)	ชื่อสายการบิน
	Date	datetime (8)	วันที่เดินทาง
	Plane number	int (4)	หมายเลขประจำเครื่องบิน
	Available seats	int (4)	จำนวนที่นั่งของสายการบินนั้น
Reservation	Customer ID	int (4)	ลำดับการจองของลูกค้า
	Reserve date	datetime (8)	วันที่เข้ามาจองตั๋วของลูกค้า
	Roundtrip	nvarchar (50)	รูปแบบการจองเที่ยวเดียว-ไปกลับ
	Date	nvarchar (10)	วันที่เดินทางไป
	Flight number	FlightnumberChar(5)	เที่ยวบินที่เดินทางไป
	Date return	nvarchar (10)	วันที่เดินทางกลับ
	Flight number return	FlightnumberChar(5)	เที่ยวบินที่เดินทางกลับ
	Seat adult	int (4)	จำนวนที่นั่งผู้ใหญ่ขาไป
	Seat child	int (4)	จำนวนที่นั่งเด็กขาไป
	Total seats	int (4)	จำนวนที่นั่งขาไปทั้งหมด
	Seat adult return	int (4)	จำนวนที่นั่งผู้ใหญ่จากกลับ
	Seat child return	int (4)	จำนวนที่นั่งเด็กจากกลับ
	Total seats return	int (4)	จำนวนที่นั่งจากกลับทั้งหมด
	Paid	int (4)	จำนวนเงินทั้งหมดที่ลูกค้าต้องจ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตาราง	ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย
	Departure city	nchar (20)	สถานีต้นทาง
	Arrival city	nchar (20)	สถานีปลายทาง
	Check-in-status	varchar (20)	สถานะของลูกค้า
Customer	Customer ID	int (4)	ลำดับการจองของลูกค้า
	Phone number	char (9)	หมายเลขโทรศัพท์ของลูกค้า
	ID	char (13)	หมายเลขบัตรประชาชนของลูกค้า
	Credit Card ID	char (16)	หมายเลขบัตรเครดิตของลูกค้า
	Password	char (4)	รหัสผ่านของลูกค้า

ตารางที่ 3.1 รายละเอียดของข้อมูลในฐานข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ขั้นตอนการทำงานและผลการทดลองของระบบ

แอปพลิเคชันระบบของตัวเครื่องบินแบ่งออกเป็น 3 ส่วน โดยส่วนแรกเป็นแอปพลิเคชันที่ทำหน้าที่ในการติดต่อสื่อสารกับโทรศัพท์ของผู้ใช้บริการ และตรวจจับสัญญาณ DTMF เพื่อนำไปเป็นข้อมูลให้กับแอปพลิเคชันในส่วนที่สอง สำหรับแอปพลิเคชันในส่วนที่สองเมื่อรับข้อมูลของผู้ใช้บริการจากแอปพลิเคชันแรกแล้วก็จะทำการติดต่อกับฐานข้อมูลของระบบ และค้นหาข้อมูลที่ต้องการเพื่อส่งกลับไปยังผู้บริการ โดยผ่านแอปพลิเคชันแรก ส่วนแอปพลิเคชันสุดท้ายเป็นส่วนที่สร้างขึ้นสำหรับผู้ทำการดูแลระบบซึ่งผลที่ได้จากการทำงานในแต่ละแอปพลิเคชันแสดงดังนี้

4.1 ส่วนที่ใช้ในการติดต่อสื่อสารกับโทรศัพท์ของผู้ใช้บริการ และตรวจจับสัญญาณ DTMF

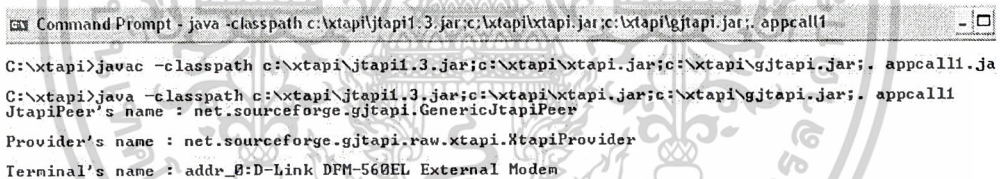
สามารถแสดงการทำงานของระบบ ได้ดังนี้

1. เปิดหน้าต่าง Command Prompt เพื่อทำการคอมไพล์โปรแกรม โดยใช้คำสั่ง javac ดังนี้

```
javac -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar; appcall1.java  
และรัน โปรแกรม โดยใช้คำสั่ง java
```

```
java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar; appcall1
```

เมื่อทำตามขั้นตอนแล้วจะได้ผลดังรูปที่ 4.1



```
Command Prompt - java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar; appcall1
C:\xtapi>javac -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1.java
C:\xtapi>java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1
JtapiPeer's name : net.sourceforge.gjtapi.GenericJtapiPeer
Provider's name : net.sourceforge.gjtapi.raw.xtapi.XtapiProvider
Terminal's name : addr_0:D-Link DFM-560EL External Modem
```

รูปที่ 4.1 ผลที่ได้การรันโปรแกรมและการรายงานผลการเชื่อมต่ออุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.1 แสดงผลการเชื่อมต่อไปยัง Terminal Object หรืออุปกรณ์ที่ใช้ติดต่อกับชุมสายโทรศัพท์ ซึ่งในปฏิญญาอินทรีอุปกรณ์ดังกล่าว คือ D-Link DFM-560EL External Modem

หลักการติดต่อไปยังอุปกรณ์ดังกล่าว แอปพลิเคชันจะทำการติดต่อไปยังอุปกรณ์ปลายทางนี้ โดยการร้องขอไปยัง TAPI ที่เชื่อมต่อกับคอมพิวเตอร์ จากนั้นจึงเลือกอุปกรณ์ปลายทางที่ต้องการใช้งานซึ่งจะมี D-Link DFM-560EL External Modem เป็นอุปกรณ์ที่เชื่อมต่ออยู่ด้วย

จากผลการทดลองในส่วนนี้ แสดงว่าขณะนี้แอปพลิเคชันสามารถติดต่อไปยังส่วนของอุปกรณ์ได้เรียบร้อยแล้ว

2. เมื่อมีการเรียกเข้ามายังหมายเลขที่ให้บริการเหตุการณ์ที่เกิดขึ้นจะถูกรายงานดังรูปที่ 4.2

```

Command Prompt - java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1
C:\xtapi>javac -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1.ja
C:\xtapi>java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1
jtapiPeer's name : net.sourceforge.gjtapi.GenericJtapiPeer
Provider's name : net.sourceforge.gjtapi.raw_xtapi.XtapiProvider
Terminal's name : addr_0:D-Link DFM-560EL External Modem
Terminal Connection is created!! : addr_0:D-Link DFM-560EL External Modem
Terminal Connection is Ringing!! : addr_0:D-Link DFM-560EL External Modem
Answering the phone
Connection Connected : addr_0
Terminal Connection is active : addr_0:D-Link DFM-560EL External Modem
Binding media service to this terminal is successful
Playing sound is successful
Playing sound is successful
Playing sound is successful
  
```

รูปที่ 4.2 การรายงานเหตุการณ์ต่างๆ ที่เกิดขึ้นจากการเรียกเข้า

จากรูปที่ 4.2 จะสามารถอธิบายได้ด้วย Call Model timing diagram ในรูปที่ 2.13 การเปลี่ยนแปลงของสถานะที่เกิดขึ้น เมื่อมีการเรียกเข้ามายังหมายเลขปลายทางที่ให้บริการ หรือ Address Object จะทำให้ TerminalConnection ของฝั่งปลายทางนี้ถูกสร้างขึ้น ซึ่งจะถูกรายงานผ่านทางแอปพลิเคชันว่า

“Terminal Connection is created!! : addr_0:D-Link DFM-560EL External Modem”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นเมื่อมีสัญญาณเรียกเข้ามาถึงอุปกรณ์ปลายทาง สถานะของ TerminalConnection จะอยู่ในสถานะ RINGING และรายงานมายังแอปพลิเคชันว่า

“Terminal Connection is Ringing!! : addr_0:D-Link DFM-560EL External Modem”

ในขณะที่ TerminalConnection อยู่ในสถานะ RINGING นี้ ระบบจะทำการรับสายและจะทำให้ Connection อยู่ในสถานะ CONNECTED กับ Address ปลายทาง และ TerminalConnection จะเปลี่ยนไปอยู่ในสถานะ ACTIVE สิ่งที่ถูกรายงานกลับมาจากเหตุการณ์ที่เกิดขึ้นนี้ คือ

“Answering the phone

Connection Connected : addr_0

Terminal Connection is active!! : addr_0:D-Link DFM-560EL External Modem”

ในสถานะ ACTIVE นี้ แสดงว่าปลายทางได้รับสายโทรศัพท์และอยู่ในสถานะที่พร้อมใช้งาน และจะทำการเชื่อมแอปพลิเคชันให้สามารถจัดการกับสื่อสัญญาณได้ในสถานะนี้ เมื่อทำการเชื่อมต่อสำเร็จ สิ่งที่ถูกรายงานมาคือ

“Binding media service to this terminal is successful”

หลังจากนั้นจึงทำการเล่นไฟล์เสียงไปยังต้นทางที่ทำการเรียกเข้ามา คู่สายที่อยู่ต้นทางจะได้ยินไฟล์เสียงที่ทำการเล่นนี้และจะรายงานกลับมายังแอปพลิเคชันว่า

“Playing sound is successful”

3. เมื่อสิ้นสุดการให้บริการ แอปพลิเคชันจะทำการตัดสาย ซึ่งจะทำให้ TerminalConnection อยู่ในสถานะ DROPPED และทุก Connection จะเปลี่ยนไปอยู่ในสถานะ DISCONNECTED และจะรายงานว่าขณะนี้เหตุการณ์ต่างๆ จะไม่ถูกรายงานกลับมายังแอปพลิเคชัน ซึ่งก็คือการเรียกนั้นสิ้นสุดลง สิ่งที่ถูกรายงานกลับมาคือ

“Terminal Connection is dropped : addr_0:D-Link DFM-560EL External Modem

no longer receive CallEvent event”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดั่งรูปที่ 4.3

```

Command Prompt - java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. appcall1
Do you want to reserve the flight anymore?
Yes : please press 1
No : please press 2
Please select the choice : 2
Playing sound is successful
Terminal Connection is dropped : addr_0:D-Link DFM-560EL External Modem
no longer receive CallEvent events

```



รูปที่ 4.3 การรายงานเหตุการณ์การสิ้นสุดการเรียกเข้า

5. เมื่อมีสายเรียกเข้ามายังหมายเลขปลายทางนี้อีกครั้ง แอปพลิเคชันก็ทำการรับสายใหม่อีกครั้ง และเหตุการณ์ต่างๆ ก็จะถูกรายงานกลับมาเช่นเดิม ดังเช่นผลการทดลองที่ได้จากข้อที่ 2-4
6. ผลการทดลองการตรวจจับสัญญาณ DTMF เมื่อผู้ใช้บริการทำการกดแป้นโทรศัพท์ เป็นตัวเลข “38569” แสดงไว้ในรูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Command Prompt - java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. TestMediaCall addr_0... -
C:\>cd\xtapi
C:\xtapi>javac -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. TestMediaCa
C:\xtapi>java -classpath c:\xtapi\jtapi1.3.jar;c:\xtapi\xtapi.jar;c:\xtapi\gjtapi.jar;. TestMediaCa
In Media Test Script...
1.1: Successfully loaded the JTapi Peer
1.2: Successfully loaded Provider
2.1: Attempting to get address... success.
2.3: Attempting to create call... success.
4.3: Attempting to get terminals for an address... success.
3.1: Attempting to connect call... success.
Call initiated...
3.2: Attempting to get call connections... success.
Hit return to continue...

Attempting to create a Media Service... success.
2.4: Attempting to get a Terminal by name... success.
13.1: Attempting to bind a Media Service to a terminal... success.
13.6: Attempting to generate DTMF on a Media Service... success.
13.7: Attempting to detect DTMF on a Media Service... success: 38569
13.4: Attempting to play on a Media Service... success.
13.8: Attempting to release a Media Service... success.

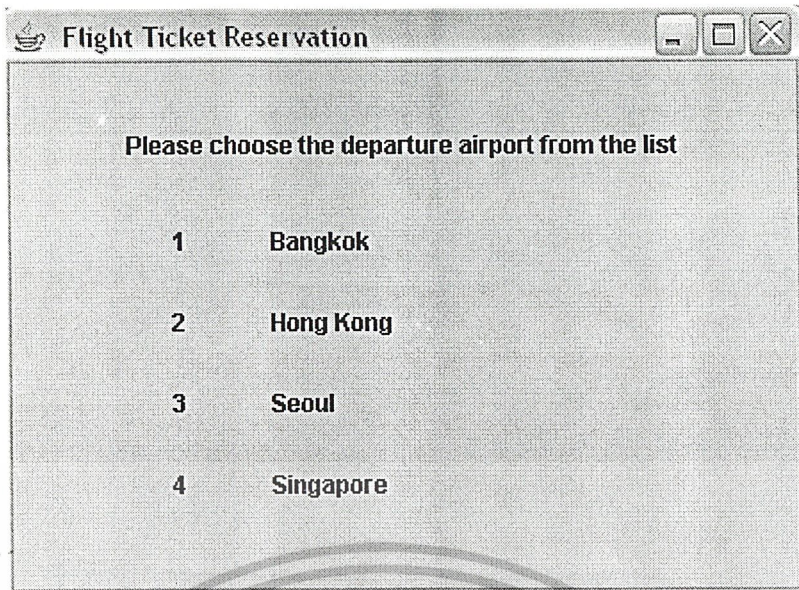
```

รูปที่ 4.4 ผลการตรวจจับสัญญาณ DTMF

4.2 ส่วนที่ทำการติดต่อกับฐานข้อมูลและค้นหาข้อมูลที่ต้องการ

ในส่วนนี้จะเป็นการรับข้อมูลของผู้ใช้บริการมาจากแอปพลิเคชันแรก ซึ่งได้ทำการเชื่อมต่อกับผู้ให้บริการแล้ว ตามขั้นตอนการทำงานของระบบหัวข้อที่ 4.1 ข้อ 2 ข้อมูลจะอยู่ในรูปแบบหมายเลข 1, 2, 3,...,9 โดยผู้ให้บริการจะทำการกดหมายเลขต่างๆ บนเป็นโทรศัพท์ตามขั้นตอนการใช้งานของระบบการจองตัวเครื่องบินในหัวข้อที่ 3.1 ข้อ 1 - 12 ตามลำดับ เมื่อผู้ให้บริการได้ทำการเลือกรายการแล้วข้อมูลที่รับเข้ามาในส่วนนี้ จะถูกนำไปให้แอปพลิเคชันในส่วนที่ติดต่อกับฐานข้อมูลได้ดำเนินการและจัดการข้อมูลในฐานข้อมูลต่อไป ซึ่งจะได้ผลการทดลองจากการที่มีผู้เข้ามาใช้บริการในระบบและผลการทดลองจากการเข้ามาเลือกรายการต่างๆ ตามขั้นตอนของระบบการจองตัวเครื่องบินดังนี้คือ

1. ตามขั้นตอนหัวข้อที่ 3.1 ข้อ 1 ของระบบการจองตัวเครื่องบิน จะเป็นการให้ผู้ให้บริการเลือกสถานีต้นทาง ผลที่ได้เป็นดังรูปที่ 4.5



รูปที่ 4.5 ผลในขั้นตอนการเลือกสถานีต้นทาง

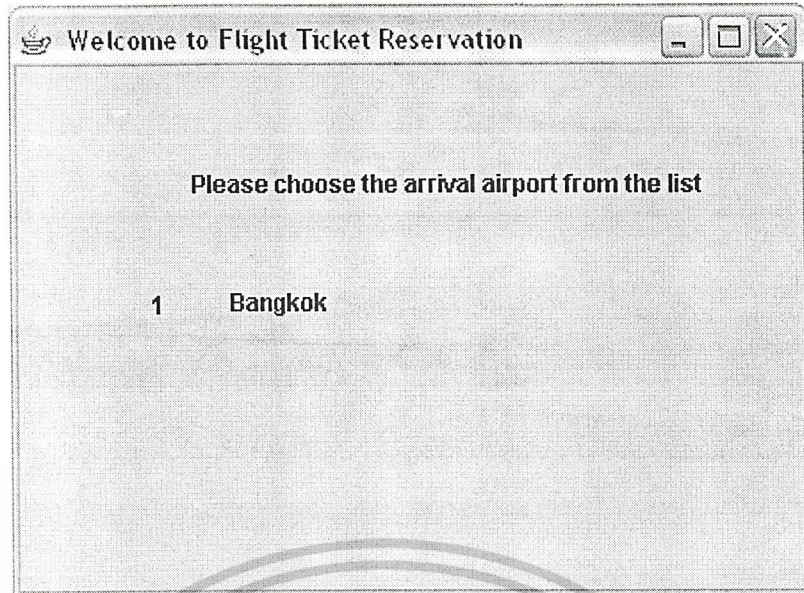
2. ตามขั้นตอนหัวข้อที่ 3.1 ข้อ 2 ของระบบการจองตั๋วเครื่องบิน จะเป็นการให้ผู้ใช้บริการเลือกสถานีปลายทางจะแบ่งเป็น 2 กรณี คือ
1. กรณีที่เลือกสถานีต้นทางเป็นกรุงเทพฯ ผลที่ได้ของสถานีปลายทางเป็นดังรูปที่ 4.6



รูปที่ 4.6 ผลในขั้นตอนการเลือกสถานีปลายทางเมื่อสถานีต้นทางเป็น Bangkok

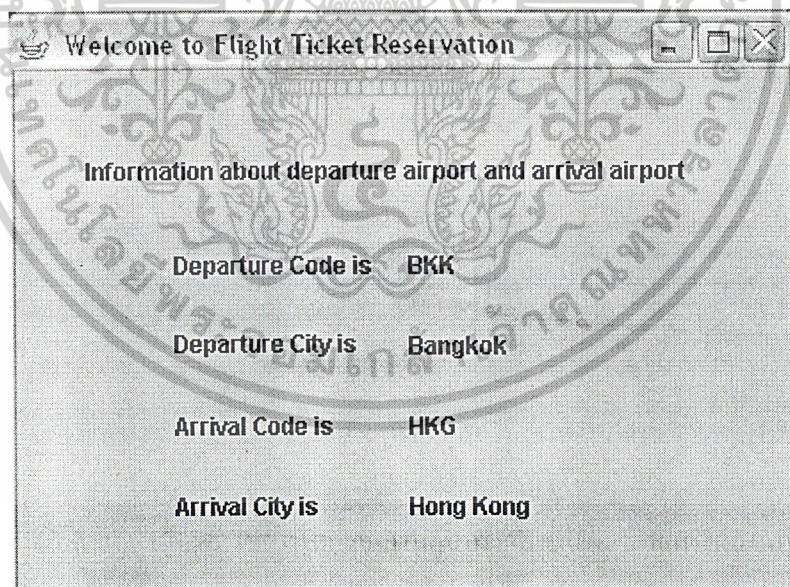
2. กรณีที่เลือกสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok ผลที่ได้ของสถานีปลายทางเป็นดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 ผลในขั้นตอนการเลือกสถานีปลายทางเมื่อสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok

3. ผลจากการที่ผู้ใช้บริการทำการเลือกสถานีต้นทางและสถานีปลายทางแบ่งเป็น 2 กรณี คือ
 1. เมื่อผู้ใช้บริการเลือกสถานีต้นทางเป็น Bangkok ผลที่ได้จากการเลือกสถานีต้นทางและปลายทางเป็นดังรูปที่ 4.8



รูปที่ 4.8 ผลจากการเลือกสถานีต้นทางและสถานีปลายทางเมื่อสถานีต้นทางเป็น Bangkok

2. เมื่อผู้ใช้บริการเลือกสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok ผลที่ได้จากการเลือกสถานีต้นทางและปลายทางเป็นดังรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Welcome to Flight Ticket Reservation

Information about departure airport and arrival airport

Departure Code is	HKG
Departure City is	Hong Kong
Arrival Code is	BKK
Arrival City is	Bangkok

รูปที่ 4.9 ผลจากการเลือกสถานีต้นทางและสถานีปลายทางเมื่อสถานีต้นทางเป็นที่อื่นๆ ยกเว้น Bangkok

4. ตามขั้นตอนหัวข้อที่ 3.1 ข้อ 3 ของระบบการจองตั๋วเครื่องบิน จะเป็นการให้ผู้ให้บริการเลือก วัน/เดือน/ปี ที่ต้องการออกเดินทางผลที่ได้เป็นดังรูปที่ 4.10

Welcome to Flight Ticket Reservation

you choose the trip on

Date	4
Month	4
Year	2005

รูปที่ 4.10 ผลที่ได้จากการเลือก วัน/เดือน/ปี ที่ต้องการออกเดินทาง

5. ผลจากการที่ผู้ให้บริการได้ทำการเลือก วัน/เดือน/ปี ที่ต้องการออกเดินทาง แล้วระบบจะทำการแสดงเที่ยวบินและรายละเอียดเกี่ยวกับสายการบินต่างๆ ได้ผลดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Welcome to Flight Ticket Reservation

Flight schedule is

No.	Flight number	D/M/Y	From	To	Depart time	Arrival time	No. seat	Arive next day	Adult price	children price
1	TG403	4/4/2548 0:00:00	BKK	SIN	11:15	08:00	98.00	.00	7,280.00	5,605.00
2	TG413	4/4/2548 0:00:00	BKK	SIN	14:35	11:15	100.00	.00	7,280.00	5,605.00
3	TG411	4/4/2548 0:00:00	BKK	SIN	16:30	13:10	100.00	.00	7,280.00	5,605.00
4	TG409	4/4/2548 0:00:00	BKK	SIN	18:35	15:15	100.00	.00	7,280.00	5,605.00
5	TG421	4/4/2548 0:00:00	BKK	SIN	21:20	18:00	100.00	.00	7,280.00	5,605.00

รูปที่ 4.11 ผลการแสดงผลเที่ยวบินและรายละเอียดเกี่ยวกับสายการบินต่างๆ

6. ตามขั้นตอนหัวข้อที่ 3.1 ข้อ 4 – 6 ของระบบการจองตั๋วเครื่องบิน จะเป็นการให้ผู้ใช้บริการเลือกเที่ยวบินขาไป เลือกระบุจำนวนที่นั่งสำหรับผู้ใหญ่และระบุที่นั่งสำหรับเด็ก และเลือกจะไปเที่ยวเดียวหรือไปกลับ ถ้าเลือกไปเที่ยวเดียวจะได้ผลจากการเลือกรายการต่างๆ ดังรูปที่ 4.12 แต่ถ้าเลือกไปกลับจะได้ผลจากการเลือกรายการต่างๆ รูปที่ 4.13 ตามลำดับ

Welcome to Flight Ticket Reservation

Information about your trip

Flight number : From : To :

D/M/Y : Type of trip :

Departure time : Arrival time :

Amount of adult :

Amount of children :

Total price :

รูปที่ 4.12 ผลจากการเลือกเที่ยวบินและการเลือกรายการต่างๆ เมื่อเดินทางไปเที่ยวเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Welcome to Flight Ticket Reservation

Information about your trip

Flight number : TG403 From : Bangkok (BKK) To : Singapore (SIN)

D/M/Y : 4/4/2005 Type of trip : one way trip

Departure time : 08:00 Arrival time : 11:15

Amount of adult : 2 Amount of children : 1 Total price : 20165

Return trip is : Flight number : TG403

From : Bangkok (BKK) To : Singapore (SIN) D/M/Y : 4/4/2005

Departure time : 08:00 Arrival time : 11:15

Amount of adult : 2 Amount of children : 1 Total price : 12885

รูปที่ 4.13 ผลจากการเลือกเที่ยวบินและการเลือกรายการต่างๆ เมื่อเดินทางไป – กลับ

7. ตามขั้นตอนหัวข้อที่ 3 ข้อ 7–10 ของระบบการจองตั๋วเครื่องบิน จะเป็นการให้ผู้ใช้บริการระบุข้อมูลของผู้ใช้บริการเองคือ รหัสผ่าน, หมายเลขโทรศัพท์, หมายเลขประจำตัวบัตรประชาชน และหมายเลขบัตรเครดิต แล้วระบบจะทำการแสดงผลจากการที่ผู้ใช้บริการทำการเลือกรายการต่างๆ ออกมา พร้อมทั้งบอกด้วยว่าผู้ใช้บริการที่เข้ามาในระบบเป็นลูกค้าลำดับที่เท่าไร ผลที่ได้ดังรูป 4.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Welcome to Flight Ticket Reservation

Information about customer

Customer ID : 17

Password : 0001

Phonenumber : 092390390

ID : 4150711110239

Credit crad : 1101110488888444

รูปที่ 4.14 ผลการแสดงผลข้อมูลและรายละเอียดของผู้ใช้บริการ

8. เมื่อผู้ให้บริการได้เลือกทำรายการต่างๆแล้ว ตามขั้นตอนของระบบการจองตั๋วเครื่องบิน ในหัวข้อที่ 3.1 ข้อ 1 - 12 จะทำให้เกิดการเปลี่ยนแปลงในระบบฐานข้อมูลขึ้นคือ จะมีการแทรกข้อมูลต่างๆ ที่เกี่ยวกับการจองของผู้ใช้บริการตารางข้อมูลเกี่ยวกับการจองตั๋วเครื่องบินของผู้ใช้บริการดังรูปที่ 4.15 และยังมีการเปลี่ยนแปลงจำนวนที่นั่งในแต่ละเที่ยวบินที่สามารถจองได้ด้วย ในตารางข้อมูลเกี่ยวกับผู้ใช้บริการดังรูปที่ 4.16 โดยค่าจำนวนที่นั่งจะถูกเปลี่ยนแปลงเมื่อมีการจองตั๋วเครื่องบินเสร็จสิ้นในแต่ละครั้ง จากรูปที่ 4.17 จะเห็นว่าจำนวนที่นั่งของเที่ยวบิน TG401 16/3/2548 (วัน/เดือน/ปี) ในคอลัมน์ available seats ถูกเปลี่ยนค่าจากเดิม 100 ที่นั่งเป็น 97 ที่นั่งที่สามารถจองได้เพราะผู้ใช้บริการได้ทำการจองไปแล้ว 3 ที่นั่ง

SQL Server Enterprise Manager - [Data in Table 'Reservation' in 'ReserveTicket' on 'JOYJA']

Customer ID	Reserve date	Round trip	Date	Flight number	Date return	Flight number retur	Seat adult	Seat child
1	10/3/2548	one way trip	16/3/2005	TG401	<NULL>	<NULL>	2	1
2	10/3/2548	return trip	21/3/2005	TG421	26/3/2005	TG414	3	2
3	10/3/2548	return trip	16/3/2005	TG403	20/3/2005	TG402	2	1
4	11/3/2548	one way trip	30/3/2005	TG411	<NULL>	<NULL>	1	1

SQL Server Enterprise Manager - [Data in Table 'Reservation' in 'ReserveTicket' on 'JOYJA']

Total seats	Seat adult return	Seat child return	Total seats return	Paid	Depart city	Arrival city	Check-in-status
3	<NULL>	<NULL>	<NULL>	12885	Bangkok	Singapore	<NULL>
5	3	2	5	33850	Bangkok	Singapore	<NULL>
3	2	1	3	20665	Bangkok	Singapore	<NULL>
2	<NULL>	<NULL>	<NULL>	12885	Bangkok	Singapore	<NULL>

รูปที่ 4.15 ตารางข้อมูลเกี่ยวกับการจองตั๋วเครื่องบินของผู้ใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQL Server Enterprise Manager - [Data in Table 'Customer' in 'ReserveTicket' on ']

File Window Help

Customer ID	Phone number	ID	Credit Card ID	password
1	090800799	4150611110229	1101456177718888	1234
2	019229101	4120700001249	0000124252213256	1103
3	090855484	5401500006237	1233032104525251	1104
4	060034022	9484955557894	1425125250021325	1105
5	060037705	1111223452123	1212103546020152	7894

รูปที่ 4.16 ตารางข้อมูลเกี่ยวกับผู้ใช้บริการ

SQL Server Enterprise Manager - [Data in Table 'Flight Instance' on ']

File Window Help

Flight number	Date	Plane number	available seats
TG401	12/3/2548	777	100
TG401	13/3/2548	777	100
TG401	14/3/2548	777	100
TG401	15/3/2548	777	100
TG401	16/3/2548	777	97
TG401	17/3/2548	777	100
TG401	18/3/2548	777	100
TG401	19/3/2548	777	100
TG401	20/3/2548	777	100
TG401	21/3/2548	777	100
TG401	22/3/2548	777	100
TG401	23/3/2548	777	100
TG401	24/3/2548	777	100
TG401	25/3/2548	777	100
TG401	26/3/2548	777	100

รูปที่ 4.17 ข้อมูลในตาราง Flight Instance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

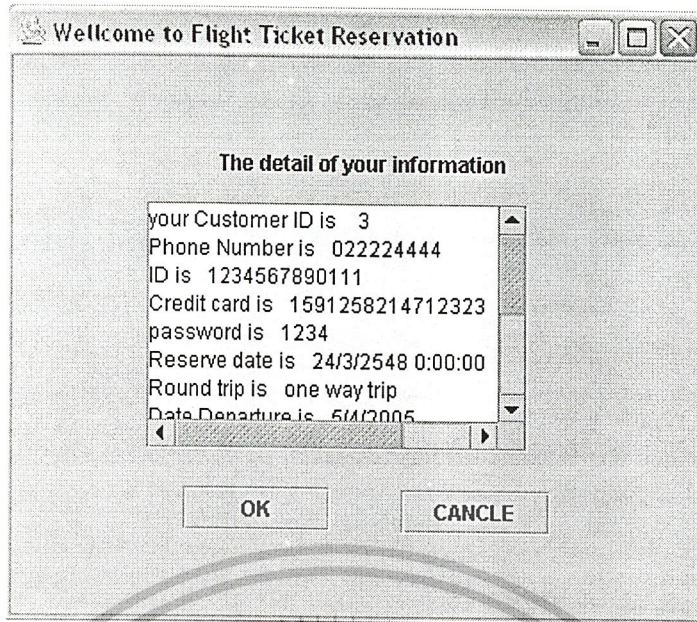
4.3 ส่วนที่ใช้สำหรับผู้ดูแลระบบซึ่งจะประกอบไปด้วยแอปพลิเคชันที่ใช้ในการรับตัว

ในส่วนสุดท้ายจะเป็นแอปพลิเคชันที่มีไว้สำหรับผู้ดูแลระบบ เพื่อใช้ในการเรียกดูข้อมูลและรายละเอียดของลูกค้า ผลที่ได้จากการทำงานในแต่ละแอปพลิเคชันแสดงดังรูปที่ 4.18– 4.20

รูปที่ 4.18 ผู้ดูแลระบบทำการล็อกอินเพื่อใช้แอปพลิเคชัน

รูปที่ 4.19 หน้าต่างที่ใช้กรอกรายละเอียดข้อมูลในส่วน password และ ID ของลูกค้าที่ต้องการรับตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 ข้อมูลต่างๆ ของลูกค้า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์

ในปฏิญานิพนธ์นี้ได้ทำการสร้างแอปพลิเคชันต่างๆ เพื่อนำมาใช้งานในระบบดังนี้

1. ส่วนที่เกี่ยวข้องกับ JTAPI คือส่วนที่ทำการติดต่อออกไปยังโครงข่ายภายนอก จัดการเกี่ยวกับกิจกรรมที่เกิดขึ้นในระหว่างการเรียกจากต้นทาง (ผู้เข้ามาใช้บริการ) ไปยังปลายทาง (ผู้ที่ให้บริการ) แอปพลิเคชันสามารถติดต่อไปถึงยังอุปกรณ์ที่ใช้ติดต่อกับโครงข่ายโทรศัพท์พื้นฐานได้ ตรวจสอบเหตุการณ์ต่างๆ ที่เกิดขึ้นกับการเรียกได้ จึงทำให้ทราบถึงสถานะต่างๆ ของออบเจกต์ที่เกี่ยวข้องของ JTAPI แล้วจึงจัดเตรียมแอปพลิเคชันในแต่ละเหตุการณ์ที่เกิดขึ้นได้อย่างถูกต้อง การรายงานผลการทำงานของระบบกลับมายังแอปพลิเคชันทำให้ง่ายแก่การเข้าใจมากขึ้น การสร้างแอปพลิเคชันที่ใช้ความสามารถของ JTAPI ในปฏิญานิพนธ์นี้สามารถทำการรับสายจากการเรียกเข้ามา และส่งสัญญาณ DTMF และสามารถเล่นเสียงตอบกลับไปยังผู้ใช้บริการที่ทำการเรียกเข้ามาได้อย่างถูกต้อง ซึ่งเป็นการชี้ให้เห็นว่า JTAPI นี้สามารถจัดการในเรื่องของสัญญาณได้

ปัญหาที่พบในการทำปฏิญานิพนธ์นี้คือ การศึกษา ทำความเข้าใจ และนำ JTAPI มาพัฒนาเป็น telephony application นั้นค่อนข้างยาก เนื่องจากแหล่งข้อมูลที่ทำให้ความรู้ในเรื่องนี้มีอยู่ไม่มากเท่าที่ควร ทั้งยังเป็นเรื่องที่ไม่แพร่หลายมากนัก การใช้งานอยู่ในวงที่จำกัดและอาจจะไม่เปิดเผยข้อมูลให้บุคคลภายนอกทราบ

2. ในปฏิญานิพนธ์นี้จะสร้างแอปพลิเคชันเพื่อรับข้อมูลที่อยู่ในรูปแบบหมายเลข 1, 2, 3, ..., 9 จากการกดเป็นโทรศัพท์เพื่อใช้เป็นข้อมูลในการเข้าสู่ขั้นตอนต่างๆ ของระบบการจองตั๋วเครื่องบิน แล้วนำข้อมูลที่ได้มาประมวลผลเพื่อให้ได้ข้อมูลตามต้องการ และในส่วนของ การติดต่อกับฐานข้อมูลที่สร้างขึ้นจะนำ JDBC มาใช้ในแอปพลิเคชัน ซึ่งทำให้ระบบสามารถทำการดึงข้อมูลจากฐานข้อมูลมาใช้งานในโปรแกรม หรือ การนำค่าที่ได้จากแอปพลิเคชันไปเก็บไว้ในฐานข้อมูล และจากการทดลองพบว่าแอปพลิเคชันสามารถทำงานได้อย่างถูกต้อง สามารถนำไปประยุกต์ใช้งานในรูปแบบการบริการอื่นๆ ต่อไปได้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class appcall1 (mian)

import javax.telephony.*;
import javax.telephony.events.*;
import javax.telephony.media.*;
import javax.telephony.callcontrol.*;
import javax.telephony.media.events.*;
import java.util.*;
import net.sourceforge.gjtapi.*;
import net.sourceforge.gjtapi.media.*;
import java.io.*;

public class appcall1 {
    static int ds;
    static ArrayList desarr;
    static String code1,code2,city1,city2;
    Display2 hhh= new Display2();
    int m1;

    public static void main(String args[]) {
        JtapiPeer peer = null;
        Provider myprovider = null;
        int num1;
        try {
            // Get a JTAPI Peer
            peer = JtapiPeerFactory.getJtapiPeer(
                "net.sourceforge.gjtapi.GenericJtapiPeer");
            System.out.println("JtapiPeer's name:peer.getName()+"\n");
            // Ask it for the Provider
            myprovider = peer.getProvider(
                "net.sourceforge.gjtapi.raw.xtapi.XtapiProvider;
                XtapiSp = TAPI");
            System.out.println("Provider'sname:"+myprovider.getName()+"\n");
        }
        catch (Exception excp) {
            System.out.println("Can't get Provider: "+excp.toString());
            System.exit(1);
        }
        // end try - catch

        try {
            Terminal[] ts = myprovider.getTerminals();
            Terminal terminal = ts[7];

            System.out.println("Terminal'sname:"+terminal.getName()+"\n");
            terminal.addCallListener(new MyNewCallListener());
        }
        catch (Exception excp) {
            System.out.println("Can't get Terminal:"+ excp.toString());
            System.exit(0);
        }
    }

    static class MyNewCallListener implements
    TerminalConnectionListener {
        CallControlCall c_call;
        Connection mycon;
        Call mycall;
        TerminalConnection tc;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Address addr;
CallControlTerminalConnection _tc;

public void callActive(CallEvent event) {
    System.out.println("The Call is now ACTIVE");
}

public void callEventTransmissionEnded(CallEvent event) {
    System.out.println("no longer receive CallEvent events\n");
}

public void callInvalid(CallEvent event) {
    System.out.println("The Call is now INVALID");
}

public void multiCallMetaMergeEnded(MetaEvent event) {}

public void multiCallMetaMergeStarted(MetaEvent event) {}

public void multiCallMetaTransferEnded(MetaEvent event) {}

public void multiCallMetaTransferStarted(MetaEvent event) {}

public void singleCallMetaProgressEnded(MetaEvent event) {}

public void singleCallMetaProgressStarted(MetaEvent event) {}

public void singleCallMetaSnapshotEnded(MetaEvent event) {}

public void singleCallMetaSnapshotStarted(MetaEvent event) {}

public void connectionAlerting(ConnectionEvent event) {
    mycon = event.getConnection();
    addr = mycon.getAddress();
    String addr_name = addr.getName();
    mycall = mycon.getCall();
    System.out.println("Connection Alerting:" + addr_name + "\n");
}

public void connectionConnected(ConnectionEvent event) {

    mycon = event.getConnection();
    addr = mycon.getAddress();
    String addr_name = addr.getName();
    System.out.println("Connection Connected:" + addr_name + "\n")
}

public void connectionCreated(ConnectionEvent event) {
    mycon = event.getConnection();
    addr = mycon.getAddress();
    String addr_name = addr.getName();
    System.out.println("Connection Created: " + addr_name + "\n")
}

public void connectionDisconnected(ConnectionEvent event) {
    mycon = event.getConnection();
    addr = mycon.getAddress();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        Display dis = new Display();
        dis.setVisible(true);

// receive origination
    getdata obj1 = new getdata();
    connectdata obj2 = new connectdata();
    int num1 = obj1.rundata1(os,ms);
    System.out.println("You choose the number : " + num1);
    code1 = obj2.connecttodata1(num1);
    System.out.println("Departure Code is : "+code1);
    city1 =obj2.getcity1();
    System.out.println("Departure City is : "+city1+"\n");
    dis.setVisible(false);

// receive destination
    ms.play("des.wav", 0, null, null);
    System.out.println("Playing sound is successful\n");
    ArrayList desarr = gl.getchoicedes(num1);
    if (num1==1){
    ms.play("deslist1.wav", 0, null, null);
    Display2 hhh =new Display2();
    hhh.setVisible(true);
    } else {
        ms.play("deslist2.wav", 0, null, null);
        Display3 jjjj= new Display3();
        jjjj.setVisible(true);
    }
    int ds = gl.getsize();
    for (int i = 1; i <= desarr.size(); i++)
        System.out.println(i + "\t\t\t" + desarr.get(i - 1));

    int num2 = obj1.rundata2(ds,ms);
    code2 = obj2.connecttodata2(num1,num2);
    System.out.println("Arrival Code is : "+code2);
    city2 =obj2.getcity1();
    System.out.println("Arrival City is : "+city2+"\n");
    Display4 try2 =new Display4();
    try2.setVisible(true);

// ทำการ เลือก วัน เดือน ปี และเลือก Flight
    flight chflight = new flight();
    d2= chflight.allflight (code1,code2,ms);
    daylog1 = chflight.DMYlogin();
    m2 =chflight.monthc();
    y2 =chflight.yearc();
    t =chflight.flightcode();
    d5 =Integer.parseInt (d2);
    m5 =Integer.parseInt (m2);
    y5 =Integer.parseInt (y2);

// JFrame Display D/M/Y
    try2.setVisible(true);
    Display5 ttt= new Display5();
    ttt.setVisible(true);
    ttt.jTextField1.setText (d2);
    ttt.jTextField2.setText (m2);
    ttt.jTextField3.setText (y2);

// แสดง และ เลือก Flight ที่ต้องการ
    String flightch= obj1.rundata3(d5,m5,y5,ms);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String Dptime1 =obj1.Deptime();
String Atime1 =obj1.Arrival();

// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
String flightch1 =""+flightch+"";
int seat1= obj2.tryseat (flightch1,y2,m2,d2);
String s_seat1 = Integer.toString(seat1);
playsound p = new playsound();
ms.play("limitseat.wav",0,null,null);
p.playnumber(s_seat1,ms);
ms.play("seat.wav",0,null,null);
System.out.println("\nFlight "+flightch+" have the number of
seats that can reserve = "+seat1);
// ระบุจำนวนที่นั่ง รวมทั้ง update จำนวนที่นั่งใน database
int Ad = obj2.AdPriceNew();
int Ch = obj2.CHPriceNew();
int m = obj1.numberseat (Ad,Ch,seat1,ms);
int a = obj1.numberseatA();
int b = obj1.numberseatC();
int checkseat = a+b;
if (checkseat < seat1){
    totalseat = seat1 - (a + b);
    String flightnum = obj2.getFlightnumber();
    String day1 = obj2.getdate1();
    obj2.toupdateseat (flightch1, y2, m2, d2, totalseat);
    System.out.print(" Display flight's information\n" +
flightnum +" " + day1.substring(0,9) +"\nThe number of seats " +
checkseat+"\n");
}
else // start else1 กรณีที่นั่งไม่เพียงพอจะทำการ เปลี่ยน , จำนวนที่นั่ง, flight , วันเดือนปี
{
    boolean check1 =true;
    do {
        while (true) {
            try{
                ms.play("seat is not enough.wav",0,null,null);
            } catch(Exception e){}
            System.out.println("You can not reserve this
flight\npress 1 choose new seat\npress 2 choose new flight\npress 3
choose new D/M/Y ");
            checkcon1 = obj1.newchoose();
            switch (checkcon1) {

                case 1: {
                    // ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
                    flightch1 = "" + flightch + "";
                    seat1 = obj2.tryseat (flightch1, y2, m2, d2);
                    s_seat1 = Integer.toString(seat1);
                    ms.play("limitseat.wav",0,null,null);
                    p.playnumber(s_seat1,ms);
                    ms.play("seat.wav",0,null,null);
                    System.out.println("Flight " +flightch+" have
the number of seats that can reserve = "
+ seat1);
                }
            }
        }
    }
}
// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
m = obj1.numberseat (Ad, Ch, seat1, ms);
a = obj1.numberseatA();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        b = obj1.numberseatC();
        checkseat = a + b;
        break;
    }
    case 2: {
        flightch = obj1.rundata3(d5, m5, y5, ms);
// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
        flightch1 = "" + flightch + "";
        seat1 = obj2.tryseat(flightch1, y2, m2, d2);
        s_seat1 = Integer.toString(seat1);
        ms.play("limitseat.wav", 0, null, null);
        p.playnumber(s_seat1, ms);
        ms.play("seat.wav", 0, null, null);
        System.out.println("Flight " + flightch +
            " have the number of seats that can reserve = "
            + seat1);
// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
        m = obj1.numberseat(Ad, Ch, seat1, ms);
        a = obj1.numberseatA();
        b = obj1.numberseatC();
        checkseat = a + b;
        break;
    }
// กรณีต้องการเลือกวัน เดือน ปี ใหม่
    case 3: {
        d2 = chflight.allflight(code1, code2, ms);
        m2 = chflight.monthc();
        y2 = chflight.yearc();
        t = chflight.flightcode();
// แสดงและเลือก Flight ที่ต้องการ
        flightch = obj1.rundata3(d5, m5, y5, ms);
// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
        seat1 = obj2.tryseat(flightch1, y2, m2, d2);
        s_seat1 = Integer.toString(seat1);
        ms.play("limitseat.wav", 0, null, null);
        p.playnumber(s_seat1, ms);
        ms.play("seat.wav", 0, null, null);
        System.out.println("Flight " + flightch +
            " have the number of seats that can reserve = "
            + seat1);
// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
        m = obj1.numberseat(Ad, Ch, seat1, ms);
        a = obj1.numberseatA();
        b = obj1.numberseatC();
        checkseat = a + b;
        break;
    } // end case3
} // end switch
try {
    if (checkseat > seat1) {
        continue;
    }
    totaleat = seat1 - (a + b);
    String flightnum = obj2.getFlightnumber();
    String day1 = obj2.getdate1();
    obj2.toupdateseat(flightch1, y2, m2, d2, totaleat);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        System.out.println("  Displayflight' s  information\n"+
flightnum+" " +day1.substring(0,9) + "\nThe number of seats "
+checkseat +"\n");
        check1 =false;
    } catch (NumberFormatException e) {
        System.out.println(
            "\nPlease choose the number from the list\n");
    }
        break;
    }
        } while(check1);
    }

// ตรวจสอบว่าไปเที่ยวเคียวหรือว่าไปกลับ
    boolean triptype = obj1.checktrip(ms);
    System.out.println(triptype);
    String trip = obj1.gettrip();
    System.out.println(trip);
    if (triptype == true) {
// one way
    Displaytest2 try6 = new Displaytest2();
    try6.setVisible(true);
    try6.jTextField1.setText(flightch);
    try6.jTextField3.setText(city1+" (" +code1+"));
    try6.jTextField4.setText(city2+" (" +code2+"));
    try6.jTextField2.setText(d2+"/"+m2+"/"+y2);
    try6.jTextField7.setText(trip);
    try6.jTextField5.setText(Dptime1);
    try6.jTextField6.setText(Atime1);
    String seatAf = Integer.toString(a);
    try6.jTextField8.setText(seatAf);
    String seatCf =Integer.toString(b);
    try6.jTextField9.setText( seatCf );
    String totalprf =Integer.toString(m);
    try6.jTextField10.setText( totalprf );
    ms.play("You reserve flight ticket from.wav",0,null,null);
//ท่านได้ทำการจองตั๋วเครื่องบินจาก
        ms.play(city1 + ".wav",0,null,null);
        ms.play("goto.wav",0,null,null);
//ไปยัง
        ms.play(city2+".wav", 0, null, null);
        ms.play("indate.wav",0,null,null);
//วันที่
        p.playnumber(d2,ms);
        ms.play("m"+m2+".wav",0,null,null);
//เดือน
        ms.play("century.wav",0,null,null);
// ค.ศ.
        p.playnumber(y2,ms);
        ms.play("flight.wav",0,null,null);
        ms.play(flightch+".wav",0,null,null);
//flight
        String cs = Integer.toString(checkseat);
        ms.play("all.wav",0,null,null);
        p.playnumber(cs,ms);
        ms.play("seat.wav",0,null,null);
//ที่นั่ง

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ms.play("totalprice.wav", 0, null, null);
//เป็นจำนวนเงินทั้งสิ้น
String mm = Integer.toString(m);
p.playnumber(mm, ms);
ms.play("baht.wav", 0, null, null);
System.out.println("You choose flight number
"+flightch+"in"+ d2+"/"+"m2+"/"+"y2+"\n
The number of seats for adult : "+a+"\n
The number of seats for children : " + b+ "\n
total price : " + m + " baht" );
} else {
if (triptype == false) {
// ทำการ เลือก วัน เดือน ปี และเลือก Flight
chflight = new flight();
d22= chflight.allflight(code2,code1,ms);
m22 =chflight.monthc();
y22 =chflight.yearc();
t =chflight.flightcode();
int d23 = Integer.parseInt(d22);
int m23 = Integer.parseInt(m22);
int y23 = Integer.parseInt(y22);
// แสดง และ เลือก Flight ที่ต้องการ
flightch22 = obj1.rundata3(d23,m23,y23,ms);
// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
String flightch2 = ""+flightch22+"";
Int seat2 = obj2.tryseat(flightch2,y22,m22,d22);
s_seat1 = Integer.toString(seat1);
ms.play("limitseat.wav", 0, null, null);
p.playnumber(s_seat1, ms);
ms.play("seat.wav", 0, null, null);
System.out.println("Flight"+flightch22+"have the
number of seats that can reserve = "+seat2);
// ระบุจำนวนที่นั่ง หรือทั้ง update จำนวนที่นั่งใน database
m1 = obj1.numberseat(Ad, Ch, seat1, ms);
a1 = obj1.numberseatA();
b1 = obj1.numberseatC();
checkseat1 = a1+b1;
if (checkseat1 < seat2){
totalseat1 = seat2 - (a1 + b1);
String flightnum3 = obj2.getFlightnumber();
String day3 = obj2.getdate1();
obj2.toupdateseat(flightch2, y22, m22, d22, totalseat1);
ms.play("You reserve flight ticket from.wav",0,null,null);
//ท่านได้ทำการจองตั๋วเครื่องบินจาก
ms.play(city1 + ".wav", 0, null, null);
ms.play("goto.wav", 0, null, null);
//ไปยัง
ms.play(city2+ ".wav", 0, null, null);
ms.play("indate.wav", 0, null, null);
//วันที่
p.playnumber(d2, ms);
ms.play("m"+m2+ ".wav", 0, null, null);
//เดือน
ms.play("century.wav", 0, null, null);
// ก.ศ.
p.playnumber(y2, ms);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ms.play("flight.wav", 0, null, null);
ms.play(flightch+".wav", 0, null, null);

//flight
String cs = Integer.toString(checkseat);
ms.play("all.wav", 0, null, null);
p.playnumber(cs, ms);
ms.play("seat.wav", 0, null, null);

//ที่นั่ง
ms.play("returnflight.wav", 0, null, null);
ms.play("from.wav", 0, null, null);
ms.play(city2+".wav", 0, null, null);
ms.play("come back.wav", 0, null, null);
ms.play(city1+".wav", 0, null, null);
ms.play("in date.wav", 0, null, null);

//วันที่
p.playnumber(d22, ms);
ms.play("m"+m22+".wav", 0, null, null);

//เดือน
ms.play("century.wav", 0, null, null);

// ค.ศ.
p.playnumber(y22, ms);
ms.play("flight.wav", 0, null, null);
ms.play(flightch22+".wav", 0, null, null);

//flight
String cs1 = Integer.toString(checkseat1);
ms.play("all.wav", 0, null, null);
p.playnumber(cs1, ms);
ms.play("seat.wav", 0, null, null);

//ที่นั่ง
ms.play("total price.wav", 0, null, null);

//เป็นจำนวนเงินทั้งสิ้น
String mm = Integer.toString((m+m1));
p.playnumber(mm, ms);
ms.play("baht.wav", 0, null, null);
System.out.println("You choose flight number"+flightch+
"in"+ d2 + "/" + m2+ "/" + y2+"\t" + "\nThe number of
seats for adult :"+a+
"seats\nThe number of seats for children:"+ b+
"seats\n and the return flight is "+ flightch22+
"in"+d22+"/"+m22+"/"+y22+"\nThe number of seats
for adult : " +a1+" seats\nThe number of seats
for children : " +b1 + " seats\ntotal price : "
+ (m + m1) + " baht");
}
else // start else1 กรณีที่นั่งไม่เพียงพอจะทำการ เปลี่ยน , จำนวนที่นั่ง, flight , วัน
เดือนปี
{
boolean check1 =true;
do {
while (true) {
try{
ms.play("seat is not enough.wav", 0, null, null);
} catch(Exception e){ }
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println("You can not reserve this flight\npress
1 choose new seat\npress 2 choose new flight\npress 3 choose new
D/M/Y");

```

```

        checkcon1 = obj1.newchoose();
        switch (checkcon1) {

// กรณี ต้องการ
                case 1: {
// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
                    flightch2 = "" + flightch22 + "";
                    seat2 = obj2.tryseat(flightch2, y22, m22, d22);
                    String s_seat2 = Integer.toString(seat2);
                    ms.play("limitseat.wav",0,null,null);
                    p.playnumber(s_seat2,ms);
                    ms.play("seat.wav",0,null,null);

                    System.out.println("Flight" + flightch22 + "have the number
of seats that can reserve = " +seat2);
                    String datetrip_oneway = d2.concat(m2 + y2);
// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
                    m1 = obj1.numberseat(Ad, Ch, seat1, ms);
                    a1 = obj1.numberseatA();
                    b1 = obj1.numberseatC();
                    checkseat1 = a1 + b1;
                    break;
                }

                case 2: {
                    flightch22 = obj1.rundata3(d5, m5, y5, ms);
// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด
                    flightch2 = "" + flightch22 + "";
                    seat2 = obj2.tryseat(flightch2, y22, m22, d22);
                    String s_seat2 = Integer.toString(seat2);
                    ms.play("limitseat.wav",0,null,null);
                    p.playnumber(s_seat2,ms);
                    ms.play("seat.wav",0,null,null);
                    System.out.println("Flight " + flightch22 +
" have the number of seats that can reserve ="
+seat2);
// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
                    m1 = obj1.numberseat(Ad, Ch, seat1, ms);
                    a1 = obj1.numberseatA();
                    b1 = obj1.numberseatC();
                    checkseat1 = a1 + b1;
                    break;
                }

// กรณีต้องการเลือกวัน เดือน ปี ใหม่
                case 3: {
// ทำการ เลือก วัน เดือน ปี และเลือก Flight
                    d22 = chflight.allflight(code2, code1,ms);
                    m22 = chflight.monthc();
                    y22 = chflight.yearc();
                    t = chflight.flightcode();

// แสดง และ เลือก Flight ที่ต้องการ
                    flightch22 = obj1.rundata3(d5, m5, y5, ms);
                    flightch2 = "" + flightch22 + "";

// ตรวจสอบว่ามีที่นั่งที่สามารถจองได้จำนวนเท่าใด

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

seat2 = obj2.tryseat(flightch2, y22, m22, d22);
String s_seat2 = Integer.toString(seat2);
ms.play("limitseat.wav", 0, null, null);
p.playnumber(s_seat2, ms);
ms.play("seat.wav", 0, null, null);
System.out.println("Flight " + flightch22 +
" have the number of seats that can reserve = "
+seat2);

// ระบุจำนวนที่นั่ง พร้อมทั้ง update จำนวนที่นั่งใน database
m1 = obj1.numberseat(Ad, Ch, seat1, ms);
a1 = obj1.numberseatA();
b = obj1.numberseatC();
checkseat1 = a1 + b1;
break;
} // end case3

} // end switch

try {
if (checkseat1 > seat2) {
continue;
}
totalseat1 = seat1 - (a1 + b1);
String flightnum3 = obj2.getFlightnumber();
String day3 = obj2.getdate1();
obj2.toupdateseat(flightch2, y22, m22, d22, totalseat1);
ms.play("You reserve flight ticket from.wav", 0, null, null);
// ท่านได้ทำการจองตั๋วเครื่องบินจาก
ms.play(city1 + ".wav", 0, null, null);
ms.play("goto.wav", 0, null, null);
// ไปยัง
ms.play(city2 + ".wav", 0, null, null);
ms.play("indate.wav", 0, null, null);
// วันที่
p.playnumber(d2, ms);
ms.play("m" + m22 + ".wav", 0, null, null);
// เดือน
ms.play("century.wav", 0, null, null);
// ค.ศ.
p.playnumber(y2, ms);
ms.play("flight.wav", 0, null, null);
ms.play(flightch + ".wav", 0, null, null);
// flight
String cs = Integer.toString(checkseat);
ms.play("all.wav", 0, null, null);
p.playnumber(cs, ms);
ms.play("seat.wav", 0, null, null);

// ที่นั่ง
ms.play("returnflight.wav", 0, null, null);
ms.play("from.wav", 0, null, null);
ms.play(city2 + ".wav", 0, null, null);
ms.play("come back.wav", 0, null, null);
ms.play(city1 + ".wav", 0, null, null);
ms.play("indate.wav", 0, null, null);

// วันที่
p.playnumber(d22, ms);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ms.play("m"+m22+".wav", 0, null, null);
//เดือน
ms.play("century.wav", 0, null, null);
// ก.ศ.
p.playnumber(y22, ms);
ms.play("flight.wav", 0, null, null);
ms.play(flightch22+".wav", 0, null, null);
//flight
String cs1 = Integer.toString(checkseat1);
ms.play("all.wav", 0, null, null);
p.playnumber(cs1, ms);
ms.play("seat.wav", 0, null, null);
//ที่นั่ง
ms.play("total price.wav", 0, null, null);
//เป็นจำนวนเงินทั้งสิ้น
String mm = Integer.toString((m+m1));
p.playnumber(mm, ms);
ms.play("baht.wav", 0, null, null);
System.out.println("Display flight's information\n" +
flightnum3 + " " + day3.substring(0,9) + "\nThe number of seats
"+checkseat1 + " seats");
System.out.println("You choose flight number" + flightch +
"in "+d2+"/" m2+"/"+y2+ trip + "\nThe number of seats for adult :
"+ a+ "seats\nThe number of seats for children:
"+b+flightch22+"in"+d22+ "/" + m22+"/"+" \n\nThe number of seats for
adult:"a1+"seats\nThe number of seat for children :"+b1"seats\ntotal
price: " + (m + m1)+" baht");
check1 = false;
}
catch (NumberFormatException e) {
System.out.println(
"\nPlease choose the number from the list\n");
}
break;
} // end while
} while(check1);
} // end else
}
}
String Dptime2 =obj1.Deptime();
String Atime2 =obj1.Arrival();
d22 = chflight.dayc();
m22 = chflight.monthc();
y22 = chflight.yearc();
flightch22 = obj1.flightna();
a1 = obj1.numberseatA();
b1 = obj1.numberseatC();
m1=(a1*200)+(b1*100);
checkseat1 = a1 + b1;
Displaytest3 try7=new Displaytest3();
try7.setVisible(true);
try7.jTextField1.setText(flightch);
try7.jTextField2.setText(city1+" (" +code1+"));
try7.jTextField3.setText(city2+" (" +code2+"));
try7.jTextField4.setText(d2+"/"+m22+"/"+y2);
try7.jTextField5.setText(trip);
try7.jTextField6.setText(Dptime1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

try7.jTextField7.setText(Atime1);
String seatAf = Integer.toString(a);
try7.jTextField8.setText(seatAf);
String seatCf =Integer.toString(b);
try7.jTextField9.setText( seatCf );
String totalprf =Integer.toString(m);
try7.jTextField10.setText( totalprf );
try7.jTextField11.setText(flightch22);
try7.jTextField12.setText(city1+" (" +code1+" "));
try7.jTextField13.setText(city2+" (" +code2+" "));
try7.jTextField14.setText(d22+"/" +m22+"/" +y22);
try7.jTextField15.setText(Dptime2);
try7.jTextField16.setText( Atime2);
String seatAd =Integer.toString(a1);
String seatCd = Integer.toString(b1);
try7.jTextField17.setText(seatAd);
try7.jTextField18.setText(seatCd);
String totalprd =Integer.toString(m1);
try7.jTextField19.setText(totalprd);

// ลำดับการจองตั๋ว
int id;
id = obj2.connectdata_id();
System.out.println("Your Customer ID " );
ms.play("your customer ID.wav",0,null,null);
String ID = Integer.toString(id);
p.playnumber(ID,ms);
System.out.println("Customer ID: " + id);
String cutumID =Integer.toString(id);
customer obj3 = new customer();
// รับค่าpassword มาจากลูกค้า
String pass = obj3.password(ms);
// รับเบอร์โทรศัพท์
// รับเลขบัตรประชาชน
String idc = obj3.idcustomer(ms);
// รับเลขบัตรเครดิต
String credit = obj3.creditcard(ms);
// การขึ้นเงินรายละเอียดว่าถูกสั่งหรือไม่
// con1= obj1.confirm(ms);
Displaytest4 try8=new Displaytest4();
try8.setVisible(true);
try8.jTextField1.setText(cutumID);
try8.jTextField2.setText(pass);
try8.jTextField3.setText(phone);
try8.jTextField4.setText( idc);
try8.jTextField5.setText(credit);

// Customer
obj2.customer(id,phone,idc,credit, pass);
String datetrip_oneway = d2+"/" +m2+"/" + y2;
String datetrip_return = d22+"/" +m22+"/" +y22;
int totat_paid = m + m1;

// Insert reservation oneway
if (triptype == true) {
obj2.reservation_oneway(id,daylog1,trip,datetrip_oneway,
flightch, a,b, checkseat, m,city1,city2,Dptime1,Atime1);
}

// Insert reservation return

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        else {
            if (triptype == false) {
                obj2.reservation_return(id, daylog1, trip, datetrip_oneway,
                flightch, datetrip_return, flightch22, a,b, checkseat, a1, b1,
                checkseat1, totat_paid, city1, city2, Dptime1, Atime1, Dptime2, Atime2);
            }
        }
        doing = obj1.doagain(ms);
    }
    ms.play("thank.wav", 0, null, null);
    System.out.println("Playing sound is successful");
    mycon.disconnect();

    } catch (Exception e) {
        System.exit(0);
    }
}

public void terminalConnectionCreated(TerminalConnectionEvent
event) {
    tc = event.getTerminalConnection();
    Terminal myterm = tc.getTerminal();

    System.out.println("Terminal Connection is created!! : " +
myterm.getName()+"\n");
}

public void terminalConnectionDropped(TerminalConnectionEvent
event) {
    tc = event.getTerminalConnection();
    Terminal myterm = tc.getTerminal();

    System.out.println("Terminal Connection is dropped : " +
myterm.getName()+"\n");
}

public void terminalConnectionPassive(TerminalConnectionEvent
event) {}

public void terminalConnectionRinging(TerminalConnectionEvent
event) {
    try {
        final TerminalConnection tc = event.getTerminalConnection();
        Terminal myterm = tc.getTerminal();
        System.out.println("Terminal Connection is Ringing!! : " +
myterm.getName()+"\n");

        _tc = (CallControlTerminalConnection)tc;

        Runnable r = new Runnable() {
        public void run() {
            try {
                tc.answer();
                System.out.println("Answering the phone\n");
            }
            catch (Exception excp) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        System.out.println("Failed to answer : " + excp);
        excp.printStackTrace();
        System.exit(1);
    }
};

    Thread T = new Thread(r);
    T.start();
} catch (Exception excp) {
System.out.println("Failedin terminalConnectionRinging");
System.exit(0);
}
} // terminalConnectionRinging

public void terminalConnectionUnknown(TerminalConnectionEvent
event) {}

} // end class MyNewCallListesner*/

static class playsound{
public void playnumber(String number, GenericMediaService ms){
    try{
        String dec_num = number;
        int count = dec_num.length();
        while (count > 0){
            switch (count) {
                // หลักหน่วย
                case 1:
                    if( dec_num.length() ==1){
                        ms.play(dec_num + ".wav",0,null,null);
                    } else{
                        String s_1 = "";
                        s_1=s_1+ dec_num.charAt(dec_num.length() -1);
                        int ns_1 = Integer.parseInt(s_1);
                        switch (ns_1) {
                            case 1:
                                ms.play("end1.wav",0,null,null);
                                break;
                            case 0:
                                break;
                            default:
                                ms.play(s_1 + ".wav",0,null,null);
                                break;
                        }
                    }
                }
            }
            break;

            // หลักสิบ
            case 2:
                String s_2 = "";
                s_ =s_2+ dec_num.charAt(dec_num.length() -2);
                int ns_2 = Integer.parseInt(s_2);
                switch (ns_2) {
                    case 0:
                        break;
                    // 10
                    case 1:
                        ms.play("10.wav",0,null,null);
                }
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
        // 20
        case 2:
            ms.play("20.wav",4,null,null);
            break;
        default:
            ms.play(s_2+".wav",4,null, null);
            ms.play("10.wav",4,null,null);
            break;
    } // ใน case 2
    break; // case หลักสิบ

// หลักร้อย
        case 3:
            String s_3="";
            s_3=s_3+dec_num.charAt(dec_num.length()-3);
            int ns_3= Integer.parseInt(s_3);
            switch (ns_3) {
                case 0:
                    break;
                default:
                    ms.play(s_3+".wav",0,null,null);
                    ms.play("100.wav",0,null,null);
                    break;
            }
            break; // หลักร้อย

// หลักพัน
        case 4:
            String s_4="";
            s_4=s_4+dec_num.charAt(dec_num.length()-4);
            int ns_4= Integer.parseInt(s_4);
            switch (ns_4) {
                case 0:
                    break;
                default:
                    ms.play(s_4+".wav",0,null, null);
                    ms.play("1000.wav",0,null, null);
                    break;
            }
            break; // หลักพัน

// หลักหมื่น
        case 5:
            String s_5="";
            s_5 = s_5+dec_num.charAt(dec_num.length()-5);
            int ns_5= Integer.parseInt(s_5);
            switch (ns_5) {
                case 0:
                    break;
                default:
                    ms.play(s_5 + ".wav", 0, null, null);
                    ms.play("10000.wav", 0, null, null);
                    break;
            }
            break; // หลักหมื่น

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// หลักแสน
```

```
        case 6:  
            String s_6="";  
s_6 = s_6+dec_num.charAt(dec_num.length()-6);  
int ns_6= Integer.parseInt(s_6);  
switch (ns_6) {  
    case 0:  
        break;  
    default:  
        ms.play(s_6 + ".wav",0,null,null);  
        ms.play("100000.wav",0,null,null);  
        break;  
    }  
break; // หลักแสน
```

```
    }  
    count = count-1;  
}  
  
}catch(Exception e){}  
} // end method  
} // end class playsound  
} // end class appcall
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class connectdata

import java.sql.*;
import java.util.*;
import javax.swing.*;
public class connectdata{
private Connection connection;
private Statement statement;
private String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
private String url = "jdbc:odbc:ReserveTicket";
private String city = "", country = "";
static int size=0;
static ArrayList arr1=null,arr2 = null,arr3 =null,arr4=null
static ArrayList arr10,arr5=null,arr6=null,arr7=null,arr8=null,
static ArrayList arr9=null,
String srr1=""; String srr2="";
String insrt3; int srr3, srr4,srr5; String g;
// ทดสอบ
public String connecttodatal(int num){
String code = "";
try {
// load database driver class
Class.forName(driver);
// establish connection to database
connection = DriverManager.getConnection(url);
// create Statement for querying database
statement = connection.createStatement();

String insrt = "SELECT [Airport Code],[Airport
City],[Airport Country]FROM Airport WHERE [Airport
ID] = " +num;
ResultSet rec1 = statement.executeQuery(insrt);
while (rec1.next()){
code = rec1.getString(1);
city = rec1.getString(2);
country = rec1.getString(3);
}
rec1.close();
statement.close();
connection.close();
}
catch (SQLException sqlException){

// detect problems loading database driver
}

catch (ClassNotFoundException classNotFound){
System.out.println("Driver Not Found");
}
return code;
}

public String getcity1(){
return city;
}

public String getcountry1(){
return country;
} //end connecttodatal()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ปลายทาง
public String connecttodata2(int num1, int num2){
    String code = "";
    String insrt = "";
    try{
// load database driver class
Class.forName(driver);
// establish connection to database
connection = DriverManager.getConnection(url);
// create Statement for querying database
statement = connection.createStatement();
        if (num1 != 1){
            insrt = "SELECT [Airport Code],[Airport City],[Airport
Country] FROM Airport WHERE [Airport ID] = 1";
        }
        else{
int number = num2 + 1;
insrt = "SELECT [Airport Code],[Airport City],[Airport
Country] FROM Airport WHERE [Airport ID] = " +
number;
        }
        ResultSet rec1 = statement.executeQuery(insrt);
        while (rec1.next()){
            code = rec1.getString(1);
            city = rec1.getString(2);
            country = rec1.getString(3);
        }
        rec1.close();
        statement.close();
        connection.close();
    }
    catch (SQLException sqlException){
// detect problems loading database driver
    }
    catch (ClassNotFoundException classNotFound){
        System.out.println("Driver Not Found");
    }
    return code;
}

public String getcity2(){
    return city;
}

public String getcountry2(){
    return country;
} // end connecttodata2()

// การหาFlight ทั้งหมด จากต้นทางไปปลายทางตาม วัน/เดือน/ปีที่กำหนด
public ArrayList connecttodata3(String org, String des,String
datef,String monthf, String year){
    arr1 = new ArrayList();
    arr2 = new ArrayList();
    arr3 = new ArrayList();
    arr4 = new ArrayList();
    arr5 = new ArrayList();
    arr6 = new ArrayList();
    arr7 = new ArrayList();
    arr8 = new ArrayList();
    arr9 = new ArrayList();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

arr10 =new ArrayList();
try {
// load data base driver class
Class.forName(driver);
// establish connect to database
connection =DriverManager.getConnection(url);
// create Statement for querying database
statement = connection.createStatement();
String aaa= "'BKK'";
String bbbb= "1";
String insrt3 ="SELECT [Flight schedule].[Flight number],
[Flight Instance].[Date], [Flight Instance].[available seats],
[Flight schedule].[From-airportcode], "+" [Flight schedule].[To
airportcode], [Flight schedule].[Arrival time], [Flight
schedule].[Departure time], [Flight schedule].[Arrive next day],
"+" [Flight schedule].[Adult price], [Flight schedule].[Children
price]" + " FROM [Flight Instance] INNER JOIN [Flight schedule]
ON [Flight Instance].[Flight number] = [Flight schedule].[Flight
number]AND [Flight schedule].[From-airportcode] = "+" + "'"+org+" "' + "
AND " + " [Flight schedule].[To airportcode] = "+des+" AND
DAY([Flight Instance].[Date]) = "+datef+" AND MONTH([Flight
Instance].[Date]) = "+monthf+" AND YEAR([Flight Instance].[Date]) = "+y
ear +"ORDER BY [Flight schedule].[Departure time]" ;
ResultSet rec3 = statement.executeQuery(insrt3);
ResultSetMetaData metadata = rec3.getMetaData();
int numberOfColumns = metadata.getColumnCount();
while (rec3.next()){
arr1.add(rec3.getString("Flight number"));
arr2.add(rec3.getString(2));
arr3.add(rec3.getString("available seats"));
arr4.add(rec3.getString("From-airportcode"));
arr5.add(rec3.getString("To airportcode"));
arr6.add(rec3.getString("Departure time"));
arr7.add(rec3.getString("Arrival time"));
arr8.add(rec3.getString("Arrive next day"));
arr9.add(rec3.getString("Adult price"));
arr10.add(rec3.getString("Children price"));
}
size = arr1.size();
rec3.close();
statement.close();
connection.close();
}
catch (SQLException sqlException) {
System.out.println("Database error");
// detect problems loading database driver
}
catch (ClassNotFoundException classNotFound) {
System.out.println("Driver Not Found");
}
catch (IndexOutOfBoundsException e) {
System.out.println(e);
}
return arr1;
}
public ArrayList getFlightarr(){
return arr1;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public int getsize(){
return size;
}
public ArrayList getdttimearr(){
return arr2;
}
public ArrayList numseat(){
return arr3;
}
public ArrayList getfromarr(){
return arr4;
}
public ArrayList gettoarr(){
return arr5;
}
public ArrayList getdepartarr(){
return arr6;
}
public ArrayList getarrivalarr(){
return arr7;
}
public ArrayList arrivnextday(){
return arr8;
}
public ArrayList Adultprice(){
return arr9;
}
public ArrayList Childrenprice(){
return arr10;
}
public String Adprice(){
return arr9.get(0).toString();
}

// ระบุจำนวนที่นั่งเด็กและผู้ใหญ่
public int tryseat(String flightnumber, String year1, String
month1,String datel){
try {
// load database driver class
Class.forName(driver);
// establish connection to database
connection = DriverManager.getConnection(url);
// create Statement for querying database
statement = connection.createStatement();
String insrt1= " SELECT [Flight Instance].[Flight
number], [Flight Instance].[Date], [Flight Instance].[available
seats], [Flight schedule].[Adult price], "+"[Flight
schedule].[Children price] FROM [Flight Instance] INNER JOIN
[Flight schedule] ON [Flight Instance].[Flight number] = [Flight
schedule].[Flight number] "+" WHERE (DAY([Flight Instance].[Date]
= "+datel+") AND (MONTH([Flight Instance].[Date]) = "+month1+ ")
AND ([Flight Instance].[Flight number] = "+flightnumber+")";
ResultSet rec2 = statement.executeQuery(insrt1);

while (rec2.next()) {
srr1 = rec2.getString(1);
srr2 = rec2.getString(2);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        srr3 = rec2.getInt("available seats");
        srr4 = rec2.getInt(4);
        srr5 = rec2.getInt(5);
    }
    rec2.close();
    statement.close();
    connection.close();

} //end try
    catch (SQLException sqlException) {

// detect problems loading database driver
    }
    catch (ClassNotFoundException classNotFound) {
        System.out.println("Driver Not Found");
    }
    return srr3;
}

public String getFlightnumber(){
return srr1;
}
public String getdate1(){
return srr2;
}
public int AdPriceNew(){
return srr4;
}
public int CHPriceNew(){
return srr5;
}
// up date seat
public String touupdateseat (String flightnumber,String year,
String month,String date ,int SCR){
    try {
// load database driver class
        Class.forName(driver);
// establish connection to database
        Connection connection = DriverManager.getConnection(url);
// create Statement for querying database
        Statement statement = connection.createStatement();
// System.out.print (totalSCR);
        String insrt3 ="UPDATE [Flight Instance] SET
[available seats]="+SCR+" WHERE ([Flight number]
="+flightnumber+")AND (DAY([Date]) = "+date+" ) AND
(MONTH([Date]) ="+month+ " )";
        statement.executeUpdate(insrt3);
        tatement.close();
        connection.close();
    } // end try
    catch (SQLException sqlException) {

// detect problems loading database driver
    }
    catch (ClassNotFoundException classNotFound) {
        System.out.println("Driver Not Found");
    }
    return g ="p";
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ลำดับการจองตั๋ว
public int connectdata_id () {
int id=0;
    try {
// load database driver class
        Class.forName(driver);
// establish connection to database
        connection = DriverManager.getConnection(url);
// create Statement for querying database

statement=connection.createStatement( ResultSet.TYPE_SCROLL_INSE
NSITIVE,ResultSet.CONCUR_READ_ONLY);
        String insrtl= "SELECT [Customer ID] FROM Customer ";
        ResultSet re1 = statement.executeQuery(insrtl);
        re1.first();
// Database null
        if ( re1.getRow() == 0)
            {
                return 1;
            }
// Database not null
        re1.last();
        id = re1.getInt(1)+1;
        re1.close();
        connection.close();
    }
    catch(SQLException sqlException) {
System.out.print("error");
JOptionPane.showMessageDialog(null,sqlException.getMessage(),"Da
tatabase Error",JOptionPane.ERROR_MESSAGE );
System.exit(1);
    }
    catch (ClassNotFoundException classNotFound) {
System.out.println("Driver Not Found");
    }
    return id;
} // end connectdata_id

// แทรกข้อมูลการจองตั๋วของลูกค้าในฐานข้อมูล

// Reservation oneway trip
public void reservation_oneway(int id,String daylog1,String
trip,String datetrip_oneway,String flightch,int a,int b,int
checkseat,int m,String city1,String city2,String Dptime1,String
Atime1) {
try {
// load database driver class
        Class.forName(driver);
// establish connection to database
        connection = DriverManager.getConnection(url);
// create Statement for querying database
        statement = connection.createStatement();
        String SQL="INSERT INTO ReservationVALUES
        ("+id+", '"+daylog1+"', '"+trip+"', '"+datetrip_oneway+"', '"+
flightch+"', null, null, "+a+", "+b+", "+checkseat+", null, null,
null, "+m+", '"+city1+"', '"+city2+"', null, '"+Dptime1+"', '"+A
time1+"', null, null)";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        statement.executeUpdate(SQL);
    }
    catch(SQLException sqlException ){
        System.out.print("error");

JOptionPane.showMessageDialog(null,sqlException.getMessage(),"Da
tabase Error",JOptionPane.ERROR_MESSAGE );
        System.exit(1);
    }
    catch (ClassNotFoundException classNotFound) {
        System.out.println("Driver Not Found");
    }
} // end reservation oneway trip
// Reservation return trip
public void reservation_return(int id, String daylog1, String
trip, String datetrip_oneway, String flightch,String
datetrip_return, String flightch22, int a,int b,int checkseat,
int a1,int b1, int checkseat1, int total_paid,String
city1,String city2,String Dptime1,String Atime1,String
Dptime2,String Atime2){
    try {
        // load database driver class
        Class.forName(driver);
        // establish connection to database
        connection = DriverManager.getConnection(url);
        // create Statement for querying database
        statement = connection.createStatement();
        String SQL="INSERT INTO Reservation VALUES
("+id+", '"+daylog1+"', '"+trip+"', '"+datetrip_oneway+"', '"+flight
ch+"', '"+datetrip_return+"', '"+flightch22+"', '"+a+", '"+b+", '"+che
ckseat+", '"+a1+"', '"+b1+", '"+checkseat1+", '"+total_paid+", '"+city1+"',
 '"+city2+"', null, '"+Dptime1+"', '"+Atime1+"', '"+Dptime2+"', '"+At
ime2+"')";
        statement.executeUpdate(SQL);
    }
    catch(SQLException sqlException ){
        System.out.print("error");

JOptionPane.showMessageDialog(null,sqlException.getMessage(),"Da
tabase Error",JOptionPane.ERROR_MESSAGE );
        System.exit(1);
    }
    catch (ClassNotFoundException classNotFound) {
        System.out.println("Driver Not Found");
    }
} // end reservation return trip

// แทรกข้อมูลของลูกค้าในฐานข้อมูลเพื่อใช้แสดงเวลารับตัว

// Customer
public void customer(int id,String phone,String idc,String
credit,String pass) {
    try {

// load database driver class
        Class.forName(driver);
        // establish connection to database
        connection = DriverManager.getConnection(url);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// create Statement for querying database
statement = connection.createStatement();
String SQL="INSERT INTO CustomerVALUES
("+id+", '"+phone+"', '"+idc+'', '"+credit+'', '"+pass
+'')";
statement.executeUpdate(SQL);
}
catch(SQLException sqlException ){
System.out.print("error");

JOptionPane.showMessageDialog(null,sqlException.getMessage(),"Da
tabase Error",JOptionPane.ERROR_MESSAGE );
System.exit(1);
}
catch (ClassNotFoundException classNotFound) {
System.out.println("Driver Not Found");
}

} // end customer
} //end class connectdata

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class getdata

import java.io.*;
import java.util.*;
import net.sourceforge.gjtapi.media.*;
public class getdata extends appcall1{
private String trip = "";
private String a ,f,c;
private int num,Discon;
private String ml;
private String dl,yl;
private int d,y,m,getipl,num1;
String contrip ="";
private int NSeatA,NSeatC,totalprice;
connectdata data = new connectdata();
boolean con;
ArrayList arr1,arr9,arr10,arr6,arr5;
String aaaa = ."";

//รับค่าเส้นทาง
public int rundatal(int o,GenericMediaService ms){
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
System.out.print( "Please choose the departure airport from the
list : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                try {
                    if (num <= 0 || num > o) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" +o + ")\n");
                            continue;
                        }
                    } catch (Exception e){}
                check = false;
                break; //ออกจากลูป
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
            } catch (IOException e){
System.out.println(e);
            }
        } while (check);
        return num;
    } //endrundatal()

//รับค่าปลายทาง
public int rundata2(int d,GenericMediaService ms){
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.print( "Please choose the arrival airport from the
list : ");

        try{
            a = stdin.readLine();
            num = Integer.parseInt(a);
            try{
                if (num <= 0 || num > d) {
                    ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" +d + ")\n");
                    continue;
                }
            }catch (Exception e){}
            check = false;
            break; //ออกจากลูป
        }catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        }catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
System.out.println("You choose the number : " + num);
    return num;
} // end rundata2()

//เลือกวันที่ออกเดินทาง
public String getD(GenericMediaService ms) {
    BufferedReader srдин = new BufferedReader(new
InputStreamReader(System.in));
    d1="";
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("date.wav", 0, null, null);
System.out.println("Please choose day for your trip from the
list (1,2,3,4,...,31) ");
            } catch (Exception e){}
            try{
                d1 = srдин.readLine();
                d = Integer.parseInt(d1);
                try{
                    if (d <= 0 || d > 31) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list
\n");
                            continue;
                        }
                    }catch(Exception e){}
                    check = false;
                    break; //ออกจากลูป
                } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
                } catch (IOException e){
System.out.println(e);
                }
            }
        } while (check);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        return d1;
    }
}
// เลือกเดือนที่ออกเดินทาง
public String getM(GenericMediaService ms) {
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    m1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("month.wav", 0, null, null);
                System.out.println("Please choose month for your trip from the
                list (1,2,3,4,5,...,12)");
            }catch(Exception e){}
            try{
                m1 = srdir.readLine();
                m = Integer.parseInt(m1);
            }try{
                if (m <= 0 || m > 12) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
            }catch(Exception e){}
            check = false;
            break; //ออกจากลูป
        } catch (NumberFormatException e) {
            System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
            System.out.println(e);
        }
    } while (check);
    return m1;
}

// เลือกปีที่เลือกเดือนทาง
public String getY(GenericMediaService ms){
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("year.wav", 0, null, null);
                System.out.println("Please choose year for your trip from the
                list (2005,2006,...) ");
            }catch (Exception e){}
            try{
                y1 = srdir.readLine();
                y = Integer.parseInt(y1);
            }try{
                if (y <= 2004 || y > 2010) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        } catch (Exception e) {}

        check = false;
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    }
    } while (check);
    return y1;
    } // end detY

public String getmonth1() {
    return m1;
}
public String getyear1(){
    return y1;
}

// สอบถามว่าเป็นการเดินทางเที่ยวเดียวหรือไปกลับ
public boolean checktrip(GenericMediaService ms){
    boolean triptype = true;
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("checktrip.wav",0,null,null);
            } catch (Exception e){}
System.out.print( "Do you want to reserve one way trip or return
trip?\nOne way trip : please press 1\nReturn trip : please press
2\nPlease select the type of your trip : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav",0,null,null);
                    } catch (Exception e) {}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
                    continue;
                }
                switch (num ) {
                    case 1:
                        triptype = true;
                        trip = "one way trip";
                        break;
                    case 2:
                        triptype = false;
                        trip = "return trip";
                        break;
                }
                check = false;
                break; //ออกจากลูป
            }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
return triptype;
} // end checktrip()

public String gettrip(){
return trip;
}

//เลือก Flight ที่ต้องการ
public String rundata3(int d6,int m6, int y6,GenericMediaService
ms){

    BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
    num = 0;
    arr1= data.getFlightarr();
    ArrayList arr2 =data.getdttimearr();
    ArrayList arr3 = data.getfromarr();
    ArrayList arr4 =data.gettoarr();
    ArrayList arr5 =data.getarrivalarr();
    ArrayList arr6 =data.getdepartarr();
    ArrayList arr7 =data.numseat();
    ArrayList arr8 = data.arrivenextday();
    ArrayList arr9 = data.Adultprice();
    ArrayList arr10 =data.Childrenprice();
    int size = data.getsize();
    Displaytest1 rrr =new Displaytest1();
    rrr.setVisible(true);
    rrr.jTextArea1.setText("1"+"\\n2"+"\\n3"+"\\n4"+"\\n5");
    String at1 =arr1.get(0).toString();
    String at2 =arr1.get(1).toString();
    String at3 =arr1.get(2).toString();
    String at4 =arr1.get(3).toString();
    String at5 =arr1.get(4).toString();
    rrr.jTextArea2.setText( at1+"\\n"+at2+"\\n"+
at3+"\\n"+ at4+"\\n"+ at5);
    String at11 =arr2.get(0).toString();
    String at21 =arr2.get(1).toString();
    String at31 =arr2.get(2).toString();
    String at41 =arr2.get(3).toString();
    String at51 =arr2.get(4).toString();
    rrr.jTextArea3.setText( at11+"\\n"+at21+"\\n"+
at31+"\\n"+ at41+ "\\n"+ at51);
    String bt11 =arr3.get(0).toString();
    String bt21 =arr3.get(1).toString();
    String bt31 =arr3.get(2).toString();
    String bt41 =arr3.get(3).toString();
    String bt51 =arr3.get(4).toString();
    rrr.jTextArea4.setText( bt11+ "\\n"+bt21+"\\n"+
bt31+ "\\n"+ bt41+"\\n"+ bt51);
    String ct11 =arr4.get(0).toString();
    String ct21 =arr4.get(1).toString();
    String ct31 =arr4.get(2).toString();
    String ct41 =arr4.get(3).toString();
    String ct51 =arr4.get(4).toString();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

rrr.jTextArea5.setText( ct11+"\n"+ct21+"\n"+
ct31+"\n"+ ct41+
String dt11 =arr5.get(0).toString();
String dt21 =arr5.get(1).toString();
String dt31 =arr5.get(2).toString();
String dt41 =arr5.get(3).toString();
String dt51 =arr5.get(4).toString();
rrr.jTextArea6.setText( dt11+"\n"+dt21+"\n"+ dt31+
"\n"+ dt41+"\n"+ dt51);
String et11 =arr6.get(0).toString();
String et21 =arr6.get(1).toString();
String et31 =arr6.get(2).toString();
String et41 =arr6.get(3).toString();
String et51 =arr6.get(4).toString();
rrr.jTextArea7.setText( et11+"\n"+et21+"\n"+
et31+"\n"+ et41+"\n"+ et51);
String ft11 =arr7.get(0).toString();
String ft21 =arr7.get(1).toString();
String ft31 =arr7.get(2).toString();
String ft41 =arr7.get(3).toString();
String ft51 =arr7.get(4).toString();
rrr.jTextArea8.setText( ft11+"\n"+ft21+"\n"+
ft31+"\n"+ ft41+ "\n"+ ft51);
String gt11 =arr8.get(0).toString();
String gt21 =arr8.get(1).toString();
String gt31 =arr8.get(2).toString();
String gt41 =arr8.get(3).toString();
String gt51 =arr8.get(4).toString();
rrr.jTextArea9.setText( gt11+"\n"+gt21+"\n"+
gt31+"\n"+ gt41+"\n"+ gt51);
String ht11 =arr9.get(0).toString();
String ht21 =arr9.get(1).toString();
String ht31 =arr9.get(2).toString();
String ht41 =arr9.get(3).toString();
String ht51 =arr9.get(4).toString();
rrr.jTextArea10.setText( ht11+"\n"+ht21+"\n"+
ht31+"\n"+ ht41+"\n"+ ht51);
String it11 =arr10.get(0).toString();
String it21 =arr10.get(1).toString();
String it31 =arr10.get(2).toString();
String it41 =arr10.get(3).toString();
String it51 =arr10.get(4).toString();
rrr.jTextAreall1.setText( it11+"\n"+it21+"\n"+
it31+"\n"+ it41+"\n"+ it51);
String test34 = "1";

System.out.println("\nThe flight schedule\n");
System.out.println("Number Flight number D/M/Y From
To Departure time Arrival time Number of seats Arrive
next day Adult price Children price ");
for (int i = 1; i <= size; i++) {
if (arr8.get(i -1) ==test34) {
// System.out.println(
// "Number Flight number D/M/Y
From To Departure time Arrival time
number seat Arive next day Adult price
children price " );
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
cal = new GregorianCalendar(y6, (m6 - 1), d6 + 1);
inticial = new MyDate(cal.getTimeInMillis());

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String arrivenext = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1) + "" + arr7.get(i - 1) +"" + arr8.get(i - 1)
+ "" + arr9.get(i - 1) + "" +arr10.get(i - 1));
    } else {
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
" " + arr3.get(i - 1)+ "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1)+"" + arr7.get(i - 1) + ""+ arr8.get(i - 1)
+ " " + arr9.get(i - 1) + " " + arr10.get(i - 1));
    }
    }

BufferedReader choose = new BufferedReader(new
InputStreamReader(System.in));
boolean check = true;
do{
while (true){
try{
ms.play("chooseflight.wav",0,null,null);
for (int i = 1; i <= size; i++) {
String ii = Integer.toString(i);
ms.play(ii+".wav",0,null,null);
ms.play(arr1.get(i - 1).toString()+".wav",0,null,null);
}
} catch(Exception e){}
try{
System.out.print("Please choose the flight for your trip : ");
GregorianCalendar cal=new GregorianCalendar(y6, (m6-1) ,d6);
MyDate inticial = new MyDate (cal.getTimeInMillis());
String dateflight = inticial.DMY2();
aaaa = stdin.readLine();
num = Integer.parseInt(aaaa);
if (num <= 0 || num > size){
try{
ms.play("again.wav",0,null,null);
} catch (Exception e){}
System.out.println("\nPlease choose flight from the list (1-
"+size+" )\n");
System.out.println("\nThe flight schedule\n");
System.out.println(
"Number Flight number D/M/Y From To Departure time
Arrival time Number of seats Arrive next day Adult price
Children price ");
for (int i = 1; i <= size; i++) {
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +""+ arr6.get(i - 1)
+ ""+ arr5.get(i - 1)+"" + arr7.get(i - 1) + "" + arr8.get(i - 1)
+"" + arr9.get(i - 1) + " " + arr10.get(i - 1));
}
continue;
}

System.out.println();
System.out.println
"Flight number D/M/Y From To Departure time Arrival
time Number of seats Arrive next day Adult price
Children price ");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println(
arr1.get(num-1)+ ""+dateflight+ ""+ arr3.get(num-1)+ "
"+arr4.get(num-1)+ ""+arr6.get(num-1)+" "+arr5.get(num-1) +"
" +arr7.get(num-1)+""+arr8.get(num-1)+ " " +arr9.get(num-1)+ "
"+arr10.get(num-1) );
        check = false;
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    }
    } while (check);
    return arr1.get(num-1).toString();
} // end checktrip()

public String flightna() {
return arr1.get(num-1).toString();
}
public String Deptime() {
ArrayList arr6 =data.getdepartarr();
num1 = Integer.parseInt(aaaa);
return arr6.get(num-1). toString ();
}
public String Arrival() {
ArrayList arr5 =data.getarrivalarr();
num1 = Integer.parseInt(aaaa);
return arr5.get(num-1). toString ();
}

//ระบุจำนวนที่นั่งเด็กและผู้ใหญ่
public int numberseat ( int priceA, int priceC ,int
w,GenericMediaService ms) {
    BufferedReader nof = new BufferedReader(new
InputStreamReader(System.in));
    String n1 =""; String n2="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("adultseat.wav",0,null,null);
            } catch(Exception e){}
System.out.println("Please choose the number of seats for adult :
");
            try{
                n1 = nof.readLine();
                NSeatA = Integer.parseInt(n1);
                int price1 = NSeatA*priceA;
            } catch(Exception e){}
            try{
                ms.play("childseat.wav",0,null,null);
            } catch(Exception e){}
System.out.println("Please choose the number of seats for
children : ");
            n2= nof.readLine();
            NSeatC =Integer.parseInt(n2);
            int price2 = NSeatC*priceC;
            int totalprice = (price1 +price2);
            return totalprice;
        } catch (NumberFormatException e) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
    }

public int totalprice01() {
return totalprice;
}
public int numberseatA () {
return NSeatA;
}
public int numberseatC () {
return NSeatC;
}

//กรณี que เลือกจำนวนที่นั่งเกิน
public int newchoose(){
String getip="";
try {
BufferedReader input = new BufferedReader(new
InputStreamReader(System.in));
getip = input.readLine();
getipl = Integer.parseInt(getip);
}
catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
} catch (IOException e){
System.out.println(e);
}
return getipl;
}
// ขึ้นชั้นข้อมูล
public boolean confirm(GenericMediaService ms) {
BufferedReader conf = new BufferedReader(new
InputStreamReader(System.in));
String c = "";
boolean check = true;
do{
while (true){
try{
ms.play("confirmdetail.wav",0,null,null);
}catch(Exception e){}
System.out.print( " If information are correct \nPlease press 1
to confirm \n press 2 to cancel : ");
try{
a = conf.readLine();
Discon = Integer.parseInt(a);
if (Discon <= 0 || Discon > 2){
try{
ms.play("again.wav",0,null,null);
}catch(Exception e){}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
continue;
}
switch (Discon) {
case 1:
con = true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        contrip = "I accept";
        check = false;
        break;
        case 2:
            con = false;
            contrip = "I do not accept";
            check = false;
            break;
        }
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
        System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
        System.out.println(e);
    }
}
} while (check);
return con;
} // end checktrip()

public String Disconfirm(){
    return contrip;
} // enn method
// สอบถามว่าต้องการจะทำการจองตั๋วเพิ่มอีกหรือไม่
public boolean doagain(GenericMediaService ms){
    boolean doing = true;
    BufferedReader stdin = new BufferedReader(new
        InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("do again.wav", 0, null, null);
            } catch (Exception e){}
            System.out.print( "\nDo you want to reserve the flight
                anymore?\nYes : please press 1\nNo : please press 2\nPlease
                select the choice : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav", 0, null, null);
                    } catch (Exception e) {}
                }
                System.out.println("\nPlease choose the number from the list
                    (lor 2)\n");
                continue;
            }
            switch (num ) {
                case 1:
                    doing = true;
                    break;
                case 2:
                    doing = false;
                    break;
            }
            check = false;
            break; //ออกจากลูป
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
return doing;
} // end checktrip()

} // end class
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class getdata

import java.io.*;
import java.util.*;
import net.sourceforge.gjtapi.media.*;
public class getdata extends appcall1{
private String trip = "";
private String a ,f,c;
private int num,Discon;
private String m1;
private String d1,y1;
private int d,y,m,getipl,num1;
String contrip ="";
private int NSeatA,NSeatC,totalprice;
connectdata data = new connectdata();
boolean con;
ArrayList arr1,arr9,arr10,arr6,arr5;
String aaaa = "";

//รับค่าต้นทาง
public int rundata1(int o,GenericMediaService ms){
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
System.out.print("Please choose the departure airport from the
list : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                try {
                    if (num <= 0 || num > o) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" +o + ")\n");
                            continue;
                        }
                    } catch(Exception e){}
                check = false;
                break; //ออกจากลูป
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
            } catch (IOException e){
System.out.println(e);
            }
        } while (check);
        return num;
    }//endrundata1()

//รับค่าปลายทาง
public int rundata2(int d,GenericMediaService ms){
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.print( "Please choose the arrival airport from the
list : ");

        try{
            a = stdin.readLine();
            num = Integer.parseInt(a);
            try{
                if (num <= 0 || num > d) {
                    ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" +d + ")\n");
                continue;
            }
            }catch (Exception e){}
            check = false;
            break; //ออกจากลูป
        }catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        }catch (IOException e){
System.out.println(e);
        }
    } while (check);
System.out.println("You choose the number : " + num);
    return num;
} // end rundata2()

//เลือกว่าวันที่ออกเดินทาง
public String getD(GenericMediaService ms) {
    BufferedReader srdir = new BufferedReader(new
InputStreamReader(System.in));
    d1="";
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("date.wav", 0, null, null);
System.out.println("Please choose day for your trip from the
list (1,2,3,4,...,31) ");
            } catch (Exception e){}
            try{
                d1 = srdir.readLine();
                d = Integer.parseInt(d1);
                try{
                    if (d <= 0 || d > 31) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list
\n");
                    continue;
                }
            }catch(Exception e){}
            check = false;
            break; //ออกจากลูป
        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        return dl;
    }
// เลือกเดือนที่ออกเดินทาง
public String getM(GenericMediaService ms) {
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    m1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("month.wav", 0, null, null);
                System.out.println("Please choose month for your trip from the
                list (1,2,3,4,5,...,12)");
            }catch(Exception e){}
            try{
                m1 = srdir.readLine();
                m = Integer.parseInt(m1);
            }try{
                if (m <= 0 || m > 12) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
            }catch(Exception e){}
            check = false;
            break; //ออกจากลูป
        } catch (NumberFormatException e) {
            System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
            System.out.println(e);
        }
    }
    } while (check);
    return m1;
}

// เลือกปีที่เลือกเดินทาง
public String getY(GenericMediaService ms){
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("year.wav", 0, null, null);
                System.out.println("Please choose year for your trip from the
                list (2005,2006,...) ");
            }catch (Exception e){}
            try{
                y1 = srdir.readLine();
                y = Integer.parseInt(y1);
            }try{
                if (y <= 2004 || y > 2010) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }catch(Exception e){}

    check = false;
    break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    } while (check);
    return y1;
    } // end detY

```

```

public String getmonth1() {
    return m1;
}
public String getyear1(){
    return y1;
}

```

// สอบถามว่าเป็นการเดินทางเที่ยวเดียวหรือไปกลับ

```

public boolean checktrip(GenericMediaService ms){
    boolean triptype = true;
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("checktrip.wav", 0, null, null);
            } catch(Exception e){}
System.out.print( "Do you want to reserve one way trip or return
trip?\nOne way trip : please press 1\nReturn trip : please press
2\nPlease select the type of your trip : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav", 0, null, null);
                    } catch(Exception e) {}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
                }
                continue;
            }
            switch (num ) {
                case 1:
                    triptype = true;
                    trip = "one way trip";
                    break;
                case 2:
                    triptype = false;
                    trip = "return trip";
                    break;
            }
            check = false;
            break; //ออกจากลูป

```

```

                continue;
            }
            switch (num ) {
                case 1:
                    triptype = true;
                    trip = "one way trip";
                    break;
                case 2:
                    triptype = false;
                    trip = "return trip";
                    break;
            }
            check = false;
            break; //ออกจากลูป

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
return triptype;
} // end checktrip()

public String gettrip(){
return trip;
}

//เลือก Flight ที่ต้องการ
public String rundata3(int d6,int m6, int y6,GenericMediaService
ms){
    BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
    num = 0;
    arr1= data.getFlightarr();
    ArrayList arr2 =data.getdttimearr();
    ArrayList arr3 = data.getfromarr();
    ArrayList arr4 =data.gettoarr();
    ArrayList arr5 =data.getarrivalarr();
    ArrayList arr6 =data.getdepartarr();
    ArrayList arr7 =data.numseat();
    ArrayList arr8 = data.arrivenextday();
    ArrayList arr9 = data.Adultprice();
    ArrayList arr10 =data.Childrenprice();
    int size = data.getsize();
    Displaytest1 rrr =new Displaytest1();
    rrr.setVisible(true);
    rrr.jTextArea1.setText("1"+" \n2"+" \n3"+" \n4"+" \n5");
    String at1 =arr1.get(0).toString();
    String at2 =arr1.get(1).toString();
    String at3 =arr1.get(2).toString();
    String at4 =arr1.get(3).toString();
    String at5 =arr1.get(4).toString();
    rrr.jTextArea2.setText( at1+" \n"+at2+" \n"+
at3+" \n"+ at4+" \n"+ at5);
    String at11 =arr2.get(0).toString();
    String at21 =arr2.get(1).toString();
    String at31 =arr2.get(2).toString();
    String at41 =arr2.get(3).toString();
    String at51 =arr2.get(4).toString();
    rrr.jTextArea3.setText( at11+" \n"+at21+" \n"+
at31+" \n"+ at41+ " \n"+ at51);
    String bt11 =arr3.get(0).toString();
    String bt21 =arr3.get(1).toString();
    String bt31 =arr3.get(2).toString();
    String bt41 =arr3.get(3).toString();
    String bt51 =arr3.get(4).toString();
    rrr.jTextArea4.setText( bt11+ " \n"+bt21+" \n"+
bt31+ " \n"+ bt41+" \n"+ bt51);
    String ct11 =arr4.get(0).toString();
    String ct21 =arr4.get(1).toString();
    String ct31 =arr4.get(2).toString();
    String ct41 =arr4.get(3).toString();
    String ct51 =arr4.get(4).toString();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

rrr.jTextArea5.setText( ct11+"\n"+ct21+"\n"+
ct31+"\n"+ ct41+
String dt11 =arr5.get(0).toString();
String dt21 =arr5.get(1).toString();
String dt31 =arr5.get(2).toString();
String dt41 =arr5.get(3).toString();
String dt51 =arr5.get(4).toString();
rrr.jTextArea6.setText( dt11+"\n"+dt21+"\n"+ dt31+
"\n"+ dt41+"\n"+ dt51);
String et11 =arr6.get(0).toString();
String et21 =arr6.get(1).toString();
String et31 =arr6.get(2).toString();
String et41 =arr6.get(3).toString();
String et51 =arr6.get(4).toString();
rrr.jTextArea7.setText( et11+"\n"+et21+"\n"+
et31+"\n"+ et41+"\n"+ et51);
String ft11 =arr7.get(0).toString();
String ft21 =arr7.get(1).toString();
String ft31 =arr7.get(2).toString();
String ft41 =arr7.get(3).toString();
String ft51 =arr7.get(4).toString();
rrr.jTextArea8.setText( ft11+"\n"+ft21+"\n"+
ft31+"\n"+ ft41+ "\n"+ ft51);
String gt11 =arr8.get(0).toString();
String gt21 =arr8.get(1).toString();
String gt31 =arr8.get(2).toString();
String gt41 =arr8.get(3).toString();
String gt51 =arr8.get(4).toString();
rrr.jTextArea9.setText( gt11+"\n"+gt21+"\n"+
gt31+"\n"+ gt41+"\n"+ gt51);
String ht11 =arr9.get(0).toString();
String ht21 =arr9.get(1).toString();
String ht31 =arr9.get(2).toString();
String ht41 =arr9.get(3).toString();
String ht51 =arr9.get(4).toString();
rrr.jTextArea10.setText( ht11+"\n"+ht21+"\n"+
ht31+"\n"+ ht41+"\n"+ ht51);
String it11 =arr10.get(0).toString();
String it21 =arr10.get(1).toString();
String it31 =arr10.get(2).toString();
String it41 =arr10.get(3).toString();
String it51 =arr10.get(4).toString();
rrr.jTextArea11.setText( it11+"\n"+it21+"\n"+
it31+"\n"+ it41+"\n"+ it51);
String test34 = "1";

System.out.println("\nThe flight schedule\n");
System.out.println("Number Flight number D/M/Y From
To Departure time Arrival time Number of seats Arrive
next day Adult price Children price ");
for (int i = 1; i <= size; i++) {
if (arr8.get(i -1) ==test34) {
// System.out.println(
// "Number Flight number D/M/Y
From To Departure time Arrival time
number seat Arive next day Adult price
children price " );
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
cal = new GregorianCalendar(y6, (m6 - 1), d6 + 1);
inticial = new MyDate(cal.getTimeInMillis());

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String arrivenext = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1) +"" + arr7.get(i - 1) +"" + arr8.get(i - 1)
+"" + arr9.get(i - 1) + "" +arr10.get(i - 1));
    } else {
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
" " + arr3.get(i - 1)+ "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1)+"" + arr7.get(i - 1) + ""+ arr8.get(i - 1)
+"" + arr9.get(i - 1) + " " + arr10.get(i - 1));
    }
    }

BufferedReader choose = new BufferedReader(new
InputStreamReader(System.in));
boolean check = true;
do{
while (true){
try{
ms.play("chooseflight.wav",0,null,null);
for (int i = 1; i <= size; i++) {
String ii = Integer.toString(i);
ms.play(ii+".wav",0,null,null);
ms.play(arr1.get(i - 1).toString()+".wav",0,null,null);
}
} catch(Exception e){}
try{
System.out.print( "Please choose the flight for your trip : ");
GregorianCalendar cal=new GregorianCalendar(y6, (m6-1) ,d6);
MyDate inticial = new MyDate (cal.getTimeInMillis());
String dateflight = inticial.DMY2();
aaaa = stdin.readLine();
num = Integer.parseInt(aaaa);
if (num <= 0 || num > size){
try{
ms.play("again.wav",0,null,null);
} catch (Exception e){}
System.out.println("\nPlease choose flight from the list (1-
"+size+" )\n");
System.out.println("\nThe flight schedule\n");
System.out.println(
"Number Flight number D/M/Y From To Departure time
Arrival time Number of seats Arrive next day Adult price
Children price ");
for (int i = 1; i <= size; i++) {
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ ""+ arr5.get(i - 1)+"" + arr7.get(i - 1) + "" + arr8.get(i - 1)
+"" + arr9.get(i - 1) + " " + arr10.get(i - 1));
}
continue;
}

System.out.println();
System.out.println
"Flight number D/M/Y From To Departure time Arrival
time Number of seats Arrive next day Adult price
Children price ";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println(
arr1.get(num-1)+ ""+dateflight+ ""+ arr3.get(num-1)+ "
"+arr4.get(num-1)+ ""+arr6.get(num-1)+" "+arr5.get(num-1) +"
" +arr7.get(num-1)+""+arr8.get(num-1)+ " " +arr9.get(num-1)+ "
"+arr10.get(num-1) );
        check = false;
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    }
    } while (check);
    return arr1.get(num-1).toString();
} // end checktrip()

public String flightna() {
return arr1.get(num-1).toString();
}
public String Deptime() {
ArrayList arr6 =data.getdepartarr();
num1 = Integer.parseInt(aaaa);
return arr6.get(num-1). toString ();
}
public String Arrival() {
ArrayList arr5 =data.getarrivalarr();
num1 = Integer.parseInt(aaaa);
return arr5.get(num-1). toString ();
}

//ระบุจำนวนที่นั่งเด็กและผู้ใหญ่
public int numberseat ( int priceA, int priceC ,int
w,GenericMediaService ms) {
    BufferedReader nof = new BufferedReader(new
InputStreamReader(System.in));
    String n1 =""; String n2="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("adultseat.wav",0,null,null);
            } catch(Exception e){}
        }
        System.out.println("Please choose the number of seats for adult :
");
        try{
            n1 = nof.readLine();
            NSeatA = Integer.parseInt(n1);
            int price1 = NSeatA*priceA;
            try{
                ms.play("childseat.wav",0,null,null);
            } catch(Exception e){}
        }
        System.out.println("Please choose the number of seats for
children : ");
        n2= nof.readLine();
        NSeatC =Integer.parseInt(n2);
        int price2 = NSeatC*priceC;
        int totalprice = (price1 +price2);
        return totalprice;
    } catch (NumberFormatException e) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
    }

public int totalprice01() {
return totalprice;
}
public int numberseatA () {
return NSeatA;
}
public int numberseatC () {
return NSeatC;
}

//กรณี que เลือกจำนวนที่นั่งเกิน
public int newchoose(){
String getip="";
try {
BufferedReader input = new BufferedReader(new
InputStreamReader(System.in));
getip = input.readLine();
getip1 = Integer.parseInt(getip);
}
catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
} catch (IOException e){
System.out.println(e);
}
}
return getip1;
}
// ยืนยันข้อมูล
public boolean confirm(GenericMediaService ms) {
BufferedReader conf = new BufferedReader(new
InputStreamReader(System.in));
String c = "";
boolean check = true;
do{
while (true){
try{
ms.play("confirmdetail.wav",0,null,null);
} catch (Exception e){}
System.out.print( " If information are correct \nPlease press 1
to confirm \n press 2 to cancel : ");
try{
a = conf.readLine();
Discon = Integer.parseInt(a);
if (Discon <= 0 || Discon > 2){
try{
ms.play("again.wav",0,null,null);
} catch (Exception e){}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
continue;
}
switch (Discon) {
case 1:
con = true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        contrip = "I accept";
        check = false;
        break;
        case 2:
            con = false;
            contrip = "I do not accept";
            check = false;
            break;
        }
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    } while (check);
    return con;
    } // end checktrip()

public String Disconfirm(){
    return contrip;
} // enn method
// สอบถามว่าต้องการจะทำการจองตัวเพิ่มอีกหรือไม่
public boolean doagain(GenericMediaService ms){
    boolean doing = true;
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("do again.wav", 0, null, null);
            } catch (Exception e){}
System.out.print( "\nDo you want to reserve the flight
anymore?\nYes : please press 1\nNo : please press 2\nPlease
select the choice : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav", 0, null, null);
                    } catch (Exception e) {}
System.out.println("\nPlease choose the number from the list
(lor 2)\n");
                    continue;
                }
                switch (num ) {
                    case 1:
                        doing = true;
                        break;
                    case 2:
                        doing = false;
                        break;
                }
                check = false;
                break; //ออกจากลูป
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

// class getdata

import java.io.*;
import java.util.*;
import net.sourceforge.gjtapi.media.*;
public class getdata extends appcall1{
private String trip = "";
private String a ,f,c;
private int num,Discon;
private String m1;
private String d1,y1;
private int d,y,m,getipl,num1;
String contrip ="";
private int NSeatA,NSeatC,totalprice;
connectdata data = new connectdata();
boolean con;
ArrayList arr1,arr9,arr10,arr6,arr5;
String aaaa = "";

//รับค่าค้นหา
public int rundatal(int o,GenericMediaService ms){
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
System.out.print("Please choose the departure airport from the
list : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                try {
                    if (num <= 0 || num > o) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" + o + ")\n");
                            continue;
                        }
                    } catch(Exception e){}
                check = false;
                break; //ออกจากลูป
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
            } catch (IOException e){
System.out.println(e);
            }
        } while (check);
        return num;
    }//endrundatal()

//รับค่าปลายทาง
public int rundata2(int d,GenericMediaService ms){
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.print( "Please choose the arrival airport from the
list : ");

        try{
            a = stdin.readLine();
            num = Integer.parseInt(a);
            try{
                if (num <= 0 || num > d) {
                    ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list (1-
" +d + ")\n");

                    continue;
                }
            }catch (Exception e){}
            check = false;
            break; //ออกจากลูป
        }catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        }catch (IOException e){
System.out.println(e);
        }
    }
} while (check);
System.out.println("You choose the number : " + num);
return num;
} // end rundata2()

//เลือกวันที่ออกเดินทาง
public String getD(GenericMediaService ms) {
    BufferedReader srdin = new BufferedReader(new
InputStreamReader(System.in));
    d1="";
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("date.wav", 0, null, null);
System.out.println("Please choose day for your trip from the
list (1,2,3,4,...,31) ");
            } catch (Exception e){}
            try{
                d1 = srdin.readLine();
                d = Integer.parseInt(d1);
                try{
                    if (d <= 0 || d > 31) {
                        ms.play("again.wav", 0, null, null);
System.out.println("\nPlease choose the number from the list
\n");

                        continue;
                    }
                }catch(Exception e){}
                check = false;
                break; //ออกจากลูป
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
            } catch (IOException e){
                System.out.println(e);
            }
        }
    } while (check);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        return dl;
    }
// เลือกเดือนที่ออกเดินทาง
public String getM(GenericMediaService ms) {
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    m1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("month.wav", 0, null, null);
                System.out.println("Please choose month for your trip from the
                list (1,2,3,4,5,...,12)");
            }catch(Exception e){}
            try{
                m1 = srdir.readLine();
                m = Integer.parseInt(m1);
            }try{
                if (m <= 0 || m > 12) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
            }catch(Exception e){}
            check = false;
            break; //ออกจากลูป
        } catch (NumberFormatException e) {
            System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
            System.out.println(e);
        }
    } while (check);
    return m1;
}

// เลือกปีที่เลือกเดินทาง
public String getY(GenericMediaService ms){
    BufferedReader srdir = new BufferedReader(new
    InputStreamReader(System.in));
    y1="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("year.wav", 0, null, null);
                System.out.println("Please choose year for your trip from the
                list (2005,2006,...) ");
            }catch (Exception e){}
            try{
                y1 = srdir.readLine();
                y = Integer.parseInt(y1);
            }try{
                if (y <= 2004 || y > 2010) {
                    ms.play("again.wav", 0, null, null);
                    System.out.println("\nPlease choose the number from the list
                    \n");
                }
                continue;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        } catch (Exception e) {}

        check = false;
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e) {
System.out.println(e);
    }
    }
    } while (check);
    return y1;
    } // end detY

public String getmonth1() {
    return m1;
}
public String getyear1(){
    return y1;
}

// สอบถามว่าเป็นการเดินทางเที่ยวหรือไปกลับ
public boolean checktrip(GenericMediaService ms) {
    boolean triptype = true;
    BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("checktrip.wav", 0, null, null);
            } catch (Exception e) {}
System.out.print( "Do you want to reserve one way trip or return
trip?\nOne way trip : please press 1\nReturn trip : please press
2\nPlease select the type of your trip : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav", 0, null, null);
                    } catch (Exception e) {}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
                    continue;
                }
                switch (num) {
                    case 1:
                        triptype = true;
                        trip = "one way trip";
                        break;
                    case 2:
                        triptype = false;
                        trip = "return trip";
                        break;
                }
                check = false;
                break; //ออกจากลูป
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
return triptype;
} // end checktrip()

public String gettrip(){
return trip;
}

//เลือกFlight ที่ต้องการ
public String rundata3(int d6,int m6, int y6,GenericMediaService
ms){

BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
num = 0;
arr1= data.getFlightarr();
ArrayList arr2 =data.getdttimearr();
ArrayList arr3 = data.getfromarr();
ArrayList arr4 =data.gettoarr();
ArrayList arr5 =data.getarrivalarr();
ArrayList arr6 =data.getdepartarr();
ArrayList arr7 =data.numseat();
ArrayList arr8 = data.arrivenextday();
ArrayList arr9 = data.Adultprice();
ArrayList arr10 =data.Childrenprice();
int size = data.getsize();
Displaytest1 rrr =new Displaytest1();
rrr.setVisible(true);
rrr.jTextArea1.setText("1"+ "\n2"+ "\n3"+ "\n4"+ "\n5");
String at1 =arr1.get(0).toString();
String at2 =arr1.get(1).toString();
String at3 =arr1.get(2).toString();
String at4 =arr1.get(3).toString();
String at5 =arr1.get(4).toString();
rrr.jTextArea2.setText( at1+ "\n"+at2+ "\n"+
at3+ "\n"+ at4+ "\n"+ at5);
String at11 =arr2.get(0).toString();
String at21 =arr2.get(1).toString();
String at31 =arr2.get(2).toString();
String at41 =arr2.get(3).toString();
String at51 =arr2.get(4).toString();
rrr.jTextArea3.setText( at11+ "\n"+at21+ "\n"+
at31+ "\n"+ at41+ "\n"+ at51);
String bt11 =arr3.get(0).toString();
String bt21 =arr3.get(1).toString();
String bt31 =arr3.get(2).toString();
String bt41 =arr3.get(3).toString();
String bt51 =arr3.get(4).toString();
rrr.jTextArea4.setText( bt11+ "\n"+bt21+ "\n"+
bt31+ "\n"+ bt41+ "\n"+ bt51);
String ct11 =arr4.get(0).toString();
String ct21 =arr4.get(1).toString();
String ct31 =arr4.get(2).toString();
String ct41 =arr4.get(3).toString();
String ct51 =arr4.get(4).toString();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

rrr.jTextArea5.setText( ct11+"\n"+ct21+"\n"+
ct31+"\n"+ ct41+
String dt11 =arr5.get(0).toString();
String dt21 =arr5.get(1).toString();
String dt31 =arr5.get(2).toString();
String dt41 =arr5.get(3).toString();
String dt51 =arr5.get(4).toString();
rrr.jTextArea6.setText( dt11+"\n"+dt21+"\n"+ dt31+
"\n"+ dt41+"\n"+ dt51);
String et11 =arr6.get(0).toString();
String et21 =arr6.get(1).toString();
String et31 =arr6.get(2).toString();
String et41 =arr6.get(3).toString();
String et51 =arr6.get(4).toString();
rrr.jTextArea7.setText( et11+"\n"+et21+"\n"+
et31+"\n"+ et41+"\n"+ et51);
String ft11 =arr7.get(0).toString();
String ft21 =arr7.get(1).toString();
String ft31 =arr7.get(2).toString();
String ft41 =arr7.get(3).toString();
String ft51 =arr7.get(4).toString();
rrr.jTextArea8.setText( ft11+"\n"+ft21+"\n"+
ft31+"\n"+ ft41+ "\n"+ ft51);
String gt11 =arr8.get(0).toString();
String gt21 =arr8.get(1).toString();
String gt31 =arr8.get(2).toString();
String gt41 =arr8.get(3).toString();
String gt51 =arr8.get(4).toString();
rrr.jTextArea9.setText( gt11+"\n"+gt21+"\n"+
gt31+"\n"+ gt41+"\n"+ gt51);
String ht11 =arr9.get(0).toString();
String ht21 =arr9.get(1).toString();
String ht31 =arr9.get(2).toString();
String ht41 =arr9.get(3).toString();
String ht51 =arr9.get(4).toString();
rrr.jTextArea10.setText( ht11+"\n"+ht21+"\n"+
ht31+"\n"+ ht41+"\n"+ ht51);
String it11 =arr10.get(0).toString();
String it21 =arr10.get(1).toString();
String it31 =arr10.get(2).toString();
String it41 =arr10.get(3).toString();
String it51 =arr10.get(4).toString();
rrr.jTextAreall.setText( it11+"\n"+it21+"\n"+
it31+"\n"+ it41+"\n"+ it51);
String test34 = "1";
System.out.println("\nThe flight schedule\n");
System.out.println("Number Flight number D/M/Y From
To Departure time Arrival time Number of seats Arrive
next day Adult price Children price ");
for (int i = 1; i <= size; i++) {
if (arr8.get(i -1) ==test34) {
// System.out.println(
// "Number Flight number D/M/Y
From To Departure time Arrival time
number seat Arive next day Adult price
children price " );
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
cal = new GregorianCalendar(y6, (m6 - 1), d6 + 1);
inticial = new MyDate(cal.getTimeInMillis());

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

String arrivenext = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1) +"" + arr7.get(i - 1) +"" + arr8.get(i - 1)
+"" + arr9.get(i - 1) + "" +arr10.get(i - 1));
    } else {
GregorianCalendar cal = new GregorianCalendar(y6, (m6 - 1), d6);
MyDate inticial = new MyDate(cal.getTimeInMillis());
String dateflight = inticial.DMY2();
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
" " + arr3.get(i - 1)+ "" + arr4.get(i - 1) +"" + arr6.get(i - 1)
+ "" + arr5.get(i - 1)+"" + arr7.get(i - 1) + ""+ arr8.get(i - 1)
+"" + arr9.get(i - 1) + " " + arr10.get(i - 1));
    }
    }

BufferedReader choose = new BufferedReader(new
InputStreamReader(System.in));
boolean check = true;
do{
    while (true){
        try{
            ms.play("chooseflight.wav",0,null,null);
            for (int i = 1; i <= size; i++) {
                String ii = Integer.toString(i);
                ms.play(ii+".wav",0,null,null);
                ms.play(arr1.get(i - 1).toString()+".wav",0,null,null);
            }
        } catch(Exception e){}
    }
}
try{
System.out.print("Please choose the flight for your trip : ");
GregorianCalendar cal=new GregorianCalendar(y6,(m6-1),d6);
MyDate inticial = new MyDate (cal.getTimeInMillis());
String dateflight = inticial.DMY2();
aaaa = stdin.readLine();
num = Integer.parseInt(aaaa);
if (num <= 0 || num > size){
    try{
        ms.play("again.wav",0,null,null);
    } catch (Exception e){}
}

System.out.println("\nPlease choose flight from the list (1-
"+size+" )\n");
System.out.println("\nThe flight schedule\n");
System.out.println(
"Number Flight number D/M/Y From To Departure time
Arrival time Number of seats Arrive next day Adult price
Children price ");
    for (int i = 1; i <= size; i++) {
System.out.println(i + "" + arr1.get(i - 1) + " " + dateflight +
"" + arr3.get(i - 1) + "" + arr4.get(i - 1) +""+ arr6.get(i - 1)
+ ""+ arr5.get(i - 1)+"" + arr7.get(i - 1) + "" + arr8.get(i - 1)
+"" + arr9.get(i - 1) + " " + arr10.get(i - 1));
    }
    continue;
}

System.out.println();
System.out.println
"Flight number D/M/Y From To Departure time Arrival
time Number of seats Arrive next day Adult price
Children price ";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println(
arr1.get(num-1)+ ""+dateflight+ ""+ arr3.get(num-1)+ "
"+arr4.get(num-1)+ ""+arr6.get(num-1)+"" +arr5.get(num-1)  +""
" +arr7.get(num-1)+""+arr8.get(num-1)+ " " +arr9.get(num-1)+ "
"+arr10.get(num-1) );
        check = false;
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    }
    } while (check);
    return arr1.get(num-1).toString();
} // end checktrip()

public String flightna() {
return arr1.get(num-1).toString();
}
public String Deptime() {
ArrayList arr6 =data.getdepartarr();
num1 = Integer.parseInt(aaaa);
return arr6.get(num-1). toString ();
}
public String Arrival() {
ArrayList arr5 =data.getarrivalarr();
num1 = Integer.parseInt(aaaa);
return arr5.get(num-1). toString ();
}

//ระบุจำนวนที่นั่งเด็กและผู้ใหญ่
public int numberseat ( int priceA, int priceC ,int
w,GenericMediaService ms) {
    BufferedReader nof = new BufferedReader(new
InputStreamReader(System.in));
    String n1 =""; String n2="";
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("adultseat.wav",0,null,null);
            } catch(Exception e){}
System.out.println("Please choose the number of seats for adult :
");
            try{
                n1 = nof.readLine();
                NSeatA = Integer.parseInt(n1);
                int price1 = NSeatA*priceA;
                try{
                    ms.play("childseat.wav",0,null,null);
                } catch(Exception e){}
System.out.println("Please choose the number of seats for
children : ");
                n2= nof.readLine();
                NSeatC =Integer.parseInt(n2);
                int price2 = NSeatC*priceC;
                int totalprice = (price1 +price2);
                return totalprice;
            } catch (NumberFormatException e) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
    }

```

```

public int totalprice01() {
return totalprice;
}
public int numberseatA () {
return NSeatA;
}
public int numberseatC () {
return NSeatC;
}

```

//กรณี que เลือกจำนวนที่นั่งเกิน

```

public int newchoose(){
String getip="";
try {
BufferedReader input = new BufferedReader(new
InputStreamReader(System.in));
getip = input.readLine();
getip1 = Integer.parseInt(getip);
}
catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
} catch (IOException e){
System.out.println(e);
}
return getip1;
}

```

// ยืนยันข้อมูล

```

public boolean confirm(GenericMediaService ms) {
BufferedReader conf = new BufferedReader(new
InputStreamReader(System.in));
String c = "";
boolean check = true;
do{
while (true){
try{
ms.play("comfirmdetail.wav",0,null,null);
} catch(Exception e){}
System.out.print( " If information are correct \nPlease press 1
to confirm \n press 2 to cancel : ");
try{
a = conf.readLine();
Discon = Integer.parseInt(a);
if (Discon <= 0 || Discon > 2){
try{
ms.play("again.wav",0,null,null);
} catch(Exception e){}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
continue;
}
switch (Discon) {
case 1:
con = true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        contrip = "I accept";
        check = false;
        break;
        case 2:
            con = false;
            contrip = "I do not accept";
            check = false;
            break;
        }
        break; //ออกจากลูป
    } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
    } catch (IOException e){
System.out.println(e);
    }
    }
    } while (check);
    return con;
    } // end checktrip()

public String Disconfirm(){
    return contrip;
}

} // enn method
// สอบถามว่าต้องการจะทำการจองตั๋วเพิ่มอีกหรือไม่
public boolean doagain(GenericMediaService ms){
    boolean doing = true;
    BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
    a = "";
    num = 0;
    boolean check = true;
    do{
        while (true){
            try{
                ms.play("do again.wav",0,null,null);
            } catch(Exception e){}
System.out.print( "\nDo you want to reserve the flight
anymore?\nYes : please press 1\nNo : please press 2\nPlease
select the choice : ");
            try{
                a = stdin.readLine();
                num = Integer.parseInt(a);
                if (num <= 0 || num > 2){
                    try{
                        ms.play("again.wav",0,null,null);
                    } catch(Exception e) {}
System.out.println("\nPlease choose the number from the list
(1or 2)\n");
                    continue;
                }
            }
            switch (num ) {
                case 1:
                    doing = true;
                    break;
                case 2:
                    doing = false;
                    break;
            }
            check = false;
            break; //ออกจากลูป
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        } catch (IOException e){
System.out.println(e);
        }
    }
    } while (check);
return doing;
} // end checktrip()

} // end class
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class flight

import java.util.*;
import java.io.*;
import net.sourceforge.gjtapi.media.*;
public class flight extends appcall1{
// ทำการ วัน เดือน ปี และเลือก Flight
    MyDate test1;
    getdata obj11 =new getdata();
    connectdata obj22 = new connectdata();
    String d3,m3,y3,t1,no,daylogin,daylogin3;
    ArrayList checkflight;
public String allflight (String code1, String
code2,GenericMediaService ms){
    boolean checkk1 = true;
    do {
        while (true){
// เลือกวันที่ต้องการเดินทาง
            System.out.print("Today is ");
            test1 = new MyDate(System.currentTimeMillis());
            daylogin= test1.DMY1();
            daylogin3 = test1.DMY2();
            System.out.println(daylogin);
            d3= obj11.getD(ms);
            m3= obj11.getM(ms);
            y3=obj11.getY(ms);
            System.out.println( "You choose (D/M/Y) :"+d3+"/"+m3+"/"+y3 );
            t1 =""+code2+"";
            try {
                checkflight = obj22.connecttodata3 (code1, t1, d3, m3, y3);
                if (checkflight.size() == 0) {
                    System.out.println("Don't have flight " + code1 +"\ttt" + code2 +
                    "on" + d3 + "/" + m3 + "/" + y3 + " Please choose new D/M/Y for
                    you trip ");
                    continue;
                }
            } else
                checkk1 = false;
                break;
            }
            catch (NumberFormatException e) {
                System.out.println("\nPlease choose the number from the list\n");
            }
        }
    } while (checkk1);
    return d3;
}

public String dayc() {
    return d3;
}
public String monthc() {
    return m3 ;
}
public String yearc() {
    return y3;
}
public String flightcode () {
    return t1;
}
public String DMYlogin(){
    return daylogin3;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class g

import java.sql.*;
import java.util.*;
public class g {
private Connection connection;
private Statement statement;
private String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
private String url = "jdbc:odbc:ReserveTicket";
private int dsize, osize;
public ArrayList getOrig(){
    ArrayList origarr = new ArrayList();
    try {
// load database driver class
        Class.forName(driver);
// establish connection to database
        connection = DriverManager.getConnection(url);
// create Statement for querying database
        statement = connection.createStatement();
        String insrt = "SELECT [Airport city] FROM Airport
ORDER BY [Airport ID] ";
        ResultSet recl = statement.executeQuery(insrt);
        ResultSetMetaData metadata = recl.getMetaData();
        int numberOfColumns = metadata.getColumnCount();
        while (recl.next()) {
            origarr.add(recl.getString("Airport city"));
        }
        osize = origarr.size();
        recl.close();
        statement.close();
        connection.close();
    } catch (SQLException sqlException){
System.out.println("Database error");
// detect problems loading database driver
    }
    catch (ClassNotFoundException classNotFound){
        System.out.println("Driver Not Found");
    }
    return origarr;
}
public int getsizel(){
    return osize;
}
public ArrayList getchoicedes(int id){
    String insrt="";
    ArrayList desarr = new ArrayList();
    try {
        Class.forName(driver);
        connection = DriverManager.getConnection(url);
        statement = connection.createStatement();
        if (id ==1){
            insrt = "SELECT [Airport city] FROM Airport WHERE
([Airport ID] <> " + id +")ORDER BY [Airport ID] ";
        } else{
            insrt = "SELECT [Airport city] FROM Airport WHERE
([Airport ID] =1) ";
        }
        ResultSet recl = statement.executeQuery(insrt);
        ResultSetMetaData metadata = recl.getMetaData();
        int numberOfColumns = metadata.getColumnCount();
        while (recl.next()) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        desarr.add(recl.getString("Airport City"));
    }
    dsize =desarr.size();
    recl.close();
    statement.close();
    connection.close();
} catch (SQLException sqlException) {
    System.out.println("Database error");
} catch (ClassNotFoundException classNotFound) {
    System.out.println("Driver Not Found");
}
}
return desarr;
}
public int getsize(){
    return dsize;
}
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class MyDate

import java.util.*;
public class MyDate {
    String Dmy1="";
    String Dmy2="";
    private String day;
    private String month;
    private String date;
    private String time;
    private String year;
    public String getDate() {
        return date;
    }
    public String getTime() {
        return time;
    }
    public String getYear() {
        return year;
    }
    public String getDay() {
        return day;
    }
    public String getMonth() {
        return month;
    }
    public MyDate(long msec){
        Date cur = new Date(msec);
        String result[] = cur.toString().split(" ");
        day=result[0];
        month=result[1];
        date=result[2];
        time=result[3];
        year=result[5];
    }
    public String daylogin() {
        return day;
    }
    public String monthlogin() {
        return month;
    }
    public String yearlogin() {
        return year;
    }
    public String DMY2() {
        return Dmy2= date+" "+month+" "+year;
    }

    public String DMY1() {
        return Dmy1= day+"\n"+date+" "+month+" "+year + " " +time;
    }

}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class customer
import java.io.*;
import java.util.*;
import net.sourceforge.gjtapi.media.*;
public class customer extends appcall1{
// รับpassword มาจากลูกค้า
public String password(GenericMediaService ms) {
    String pass = "";
    String pass1;
    System.out.println("Your password");
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    boolean check = true,checkpass = true;
    do{
        try{
            checkpass = true;
            try{
                ms.play("presspw.wav",0,null,null);
            } catch(Exception e){}
        } catch(Exception e){}
        System.out.print( "Please enter your password : ");
        pass = stdin.readLine();
        if (pass.length() == 4){
            try{
                ms.play("your pw is.wav", 0, null, null);
                for (int i = 0; i < pass.length(); i++) {
                    ms.play(pass.charAt(i) + ".wav", 0, null, null);
                }
            } catch(Exception e){}
        }
        System.out.println("Your password is : "+ pass);
        while (checkpass) {
            try{
                ms.play("confirmdetail.wav",0,null,null);
            }catch(Exception e){}
        }
        System.out.print( "Check your password 'Yes' or 'No' \n Yes :
        please press '1' \n No : please press '2' \n" );
        System.out.print("You choose : ");
        try{
            pass1 = stdin.readLine();
            if (pass1.equals("1") || pass1.equals("2"))
                {
                    if (pass1.equals("1"))
                        {checkpass = false;
                        check=false;}
                    else {
                        try{
                            ms.play("presspw.wav",0,null,null);
                        } catch(Exception e){}
                    }
                }
            System.out.println("You don't accept your password \nPlease
            enter your password again");
            checkpass= false;
            check=true; }
        }
        else{
            System.out.println("\nPlease choose the number from the list\n");
            System.out.println("Your password is : " + pass);
            checkpass = true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        check = true;}

        } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
        }
        catch (IOException e){
System.out.println(e) ; }
        } // end while
        } // end if pass <>4

    else{
System.out.println("your password !=4 \nPlease enter your
password again");
        check = true ; }
        } catch (NumberFormatException e) {
System.out.println("\nPlease input number\n");
        } catch (IOException e){
System.out.println(e);
        }
        } while (check);
        return pass;
    } // end password

// รับเบอร์โทรศัพท์มาจากลูกค้า
public String phonenumber(GenericMediaService ms) {
    String phone = "";
    String phonel;
    System.out.println("Your phone number");
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    boolean check = true,checkphone = true;
    do{
        try{
            checkphone = true;
            try{
                ms.play("pressphonenumber.wav",0,null,null);
            } catch(Exception e){}
            System.out.print( "Please enter your phone number : ");
            phone = stdin.readLine();
            if (phone.length() == 9){
                try{
                    ms.play("your phonenumber.wav",0,null,null);
                    for (int i = 0; i < phone.length(); i++) {
                        ms.play(phone.charAt(i) + ".wav", 0, null, null);
                    }
                } catch(Exception e){}
            }
        }
    } while (checkphone){
        try{
            ms.play("confirmdetail.wav",0,null,null);
        }catch(Exception e){}
    }
    System.out.print("Check your phone number 'Yes' or 'No' \n Yes :
please press '1' \n No : please press '2' \n" );
    System.out.print("You choose: ");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

try{
    phone1 = stdin.readLine();
    if (phone1.equals("1") ||
phone1.equals("2"))
        {
            if (phone1.equals("1"))
            {checkphone = false;
            check=false;}
            else {
System.out.println("You don't accept your phone number \nPlease
enter your phone number again");
                checkphone= false;
                check=true; }
            }
        else{
System.out.println("\nPlease choose the number from the list\n");
System.out.println("Your phone number is : " + phone);
                checkphone = true;
                check = true;}
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the
list\n");
            }
        } catch (IOException e){
System.out.println(e) ; } // end while
            } // end if phone <>9
        else{
System.out.println("Your phone number !=9 \nPlease enter your
phone number again");
                check = true ; }
            } catch (NumberFormatException e) {
System.out.println("\nPlease input number\n");
            } catch (IOException e){
                System.out.println(e);
            }
        } while (check);
        return phone;
    } // end phonenumber

```

```

// รับเลขบัตรประจำตัวประชาชนจากลูกค้า
public String idcustomer(GenericMediaService ms) {
    String idc = "";
    String idc1 ;
    System.out.println("Your ID card");
    BufferedReader stdin = new BufferedReader(new
InputStreamReader(System.in));
    boolean check = true,checkidcard = true;
    do{
        try{
            checkidcard = true;
            try{
                ms.play("pressID.wav",0,null,null);
            } catch(Exception e){}

            System.out.print( "Please enter your ID card : ");
            idc = stdin.readLine();
            if (idc.length() == 13){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

try{
    ms.play("your ID.wav",0,null,null);
    for (int i = 0; i < idc.length(); i++) {
        ms.play(idc.charAt(i) + ".wav", 0, null, null);
    }
} catch(Exception e){}
System.out.println("Your ID card is : " + idc);
while (checkidcard){
    try{
        ms.play("confirmdetail.wav",0,null,null);
    }catch(Exception e){}
System.out.print( "Check your ID card 'Yes' or 'No' \n Yes :
please press '1' \n No : please press '2' \n" );
System.out.print("You choose: ");
    try{
        idc1 = stdin.readLine();
        if (idc1.equals("1") || idc1.equals("2"))
            {
                if (idc1.equals("1"))
                    {checkidcard = false;
                    check=false;}
                else {
System.out.println("You don't accept your ID card \nPlease enter
your ID card again");
                    checkidcard= false;
                    check=true; }
            }
        else{
System.out.println("\nPlease choose the number from the list\n");
System.out.println("Your ID card is : " + idc);
                    checkidcard = true;
                    check = true;}
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
            }
            catch (IOException e){
                System.out.println(e) ; }
            } // end while
            } // end if idcard <>13
        else{
System.out.println("Your ID card !=13 \nPlease enter your ID
card again");
            check = true ; }
        } catch (NumberFormatException e) {
System.out.println("\nPlease input number\n");
        } catch (IOException e){
System.out.println(e);
        }
    } while (check);
    return idc;
} //end idcustomer

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// รับเลขบัตรเครดิต
public String creditcard(GenericMediaService ms) {
    String credit = "";
    String credit1 ;
    System.out.println("Your credit card ID");
    BufferedReader stdin = new BufferedReader(new
    InputStreamReader(System.in));
    boolean check = true,checkcredit = true;
do{
try{
    checkcredit = true;
    try{
        ms.play("presscredit.wav",0,null,null);
    } catch(Exception e){}
System.out.print( "Please enter your credit card ID : ");
    credit = stdin.readLine();
        if (credit.length() == 16){
            try{
                ms.play("your creditnum.wav",0,null,null);
                for (int i = 0; i < credit.length(); i++) {
                    ms.play(credit.charAt(i) + ".wav", 0, null, null);
                }
            } catch(Exception e){}
System.out.println("Your credit card ID is : " + credit);
            while (checkcredit){
                try{
                    ms.play("confirmdetail.wav",0,null,null);
                }catch(Exception e){}
System.out.print( "Check your credit card ID 'Yes' or 'No' \n
Yes : please press '1' \n No : please press '2' \n" );
System.out.print("You choose: ");
                try{
                    credit1 = stdin.readLine();
                    if (credit1.equals("1") ||
credit1.equals("2"))
                    {
                        if (credit1.equals("1"))
                            {checkcredit = false;
                            check=false;}
                        else {
System.out.println("You don't accept your credit card ID
\nPlease enter your credit card ID again");
                            checkcredit= false;
                            check=true; }
                    }
                } else{
System.out.println("\nPlease choose the number from the list\n");
System.out.println("Your credit card ID is : " + credit);
                    checkcredit = true;
                    check = true;}
            } catch (NumberFormatException e) {
System.out.println("\nPlease choose the number from the list\n");
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        catch (IOException e){
System.out.println(e) ; }
        } // end while
        } // end if credit card ID <>16
    else{
System.out.println("Your credit card ID !=16 \nPlease enter your
credit card ID again");
        check = true ; }
        } catch (NumberFormatException e) {
System.out.println("\nPlease input number\n");
        } catch (IOException e){
System.out.println(e);
        }
    } while (check);
    return credit;
} // end creditcard

} // end class customer

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class Display

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;

public class Display extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JLabel jLabel5 = new JLabel();
    JLabel jLabel6 = new JLabel();
    JLabel jLabel7 = new JLabel();
    JLabel jLabel8 = new JLabel();
    JLabel jLabel9 = new JLabel();
    public Display() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle(" Flight Ticket Reservation");
        ArrayList ori = new ArrayList();
        g_get = new g ();
        ori = get.getOrig();
        String c1 =ori.get(0).toString();
        String c2 =ori.get(1).toString();
        String c3 =ori.get(2).toString();
        String c4 =ori.get(3).toString();
        System.out.println(c1);
        jLabel1.setText("Please choose the departure airport from
the list");
        jLabel1.setBounds(new Rectangle(58, 33, 283, 15));
        this.setDefaultCloseOperation(HIDE_ON_CLOSE);
        this.getContentPane().setLayout(null);
        jLabel2.setText("1");
        jLabel2.setBounds(new Rectangle(80, 80, 34, 15));
        jLabel3.setText("2");
        jLabel3.setBounds(new Rectangle(80, 120, 34, 15));
        jLabel4.setText("3");
        jLabel4.setBounds(new Rectangle(80, 160, 34, 15));
        jLabel5.setText(c1);
        jLabel5.setBounds(new Rectangle(130, 80, 130, 15));
        jLabel6.setText(c2);
        jLabel6.setBounds(new Rectangle(130, 120, 98, 15));
        jLabel7.setText(c3);
        jLabel7.setBounds(new Rectangle(130, 160, 101, 15));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

jLabel8.setText("4");
jLabel8.setBounds(new Rectangle(80, 200, 34, 15));
jLabel9.setText(c4);
jLabel9.setBounds(new Rectangle(130, 200, 128, 15));
this.getContentPane().add(jLabel1, null);
this.getContentPane().add(jLabel2, null);
this.getContentPane().add(jLabel3, null);
this.getContentPane().add(jLabel4, null);
this.getContentPane().add(jLabel8, null);
this.getContentPane().add(jLabel5, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jLabel7, null);
this.getContentPane().add(jLabel9, null);
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class Display2

import javax.swing.*;
import java.awt.*;
import java.util.*;
public class Display2 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JLabel jLabel5 = new JLabel();
    JLabel jLabel6 = new JLabel();
    JLabel jLabel7 = new JLabel();
    public Display2() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("Please choose the arrival airport from the
list ");
        jLabel1.setBounds(new Rectangle(70, 50, 260, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("1");
        jLabel2.setBounds(new Rectangle(70, 100, 34, 15));
        jLabel3.setText("2");
        jLabel3.setBounds(new Rectangle(70, 150, 34, 15));
        jLabel4.setText("3");
        jLabel4.setBounds(new Rectangle(70, 200, 34, 15));
        jLabel5.setText("Hong Kong");
        jLabel5.setBounds(new Rectangle(140, 100, 78, 15));
        jLabel6.setText("Seoul");
        jLabel6.setBounds(new Rectangle(140, 150, 109, 15));
        jLabel7.setText("Singapore");
        jLabel7.setBounds(new Rectangle(140, 200, 124, 15));
        this.getContentPane().add(jLabel1, null);
        this.getContentPane().add(jLabel5, null);
        this.getContentPane().add(jLabel6, null);
        this.getContentPane().add(jLabel7, null);
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel3, null);
        this.getContentPane().add(jLabel4, null);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class Display3

import javax.swing.*;
import java.awt.*;
public class Display3 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    public Display3() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("Please choose the arrival airport from the
list");
        jLabel1.setBounds(new Rectangle(87, 51, 265, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("1");
        jLabel2.setBounds(new Rectangle(66, 110, 34, 15));
        jLabel3.setText("Bangkok");
        jLabel3.setBounds(new Rectangle(106, 109, 73, 15));
        this.getContentPane().add(jLabel1, null);
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel3, null);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class Display4

import javax.swing.*;
import java.awt.*;
import java.util.*;
public class Display4 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JLabel jLabel5 = new JLabel();
    JLabel jLabel6 = new JLabel();
    JLabel jLabel7 = new JLabel();
    JLabel jLabel8 = new JLabel();
    JLabel jLabel9 = new JLabel();
    public Display4() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        String cityorg,citydes,codedes,codeorg;
        appcall1 i1= new appcall1();
        cityorg =i1.city1;
        codeorg=i1.code1;
        citydes=i1.city2;
        codedes=i1.code2;
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("Information about departure airport and
arrival airport ");
        jLabel1.setBounds(new Rectangle(36, 43, 381, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("Departure Code is ");
        jLabel2.setBounds(new Rectangle(80, 90, 148, 15));
        jLabel3.setText("Departure City is ");
        jLabel3.setBounds(new Rectangle(80, 130, 127, 13));
        jLabel4.setText("Arrival Code is");
        jLabel4.setBounds(new Rectangle(80, 170, 118, 15));
        jLabel5.setText("Arrival City is");
        jLabel5.setBounds(new Rectangle(80, 210, 130, 15));
        jLabel6.setText(codeorg);
        jLabel6.setBounds(new Rectangle(200, 90, 95, 15));
        jLabel7.setText(cityorg);
        jLabel7.setBounds(new Rectangle(200, 130, 121, 15));
        jLabel8.setText( codedes);
        jLabel8.setBounds(new Rectangle(200, 170, 117, 15));
        jLabel9.setText(citydes);
        jLabel9.setBounds(new Rectangle(200, 210, 107, 15));
        this.getContentPane().add(jLabel3, null);
        this.getContentPane().add(jLabel4, null);
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel5, null);
        this.getContentPane().add(jLabel7, null);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
this.getContentPane().add(jLabel8, null);
this.getContentPane().add(jLabel9, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jLabel11, null);
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class Display5

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Display5 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JTextField jTextField2 = new JTextField();
    JTextField jTextField3 = new JTextField();
    public Display5() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("you choose the trip on");
        jLabel1.setBounds(new Rectangle(134, 38, 159, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("Date");
        jLabel2.setBounds(new Rectangle(105, 101, 46, 15));
        jLabel3.setText("Month");
        jLabel3.setBounds(new Rectangle(105, 154, 57, 15));
        jLabel4.setText("Year");
        jLabel4.setBounds(new Rectangle(105, 202, 46, 15));
        jTextField1.setBackground(Color.lightGray);
        jTextField1.setText("");
        jTextField1.setBounds(new Rectangle(165, 96, 103, 21));
        jTextField1.addActionListener(new
Display5_jTextField1_actionAdapter(this));
        jTextField2.setBackground(Color.lightGray);
        jTextField2.setBounds(new Rectangle(165, 149, 104, 21));
        jTextField3.setBackground(Color.lightGray);
        jTextField3.setBounds(new Rectangle(165, 200, 105, 21));
        jTextField3.addActionListener(new
Display5_jTextField3_actionAdapter(this));
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel3, null);
        this.getContentPane().add(jLabel4, null);
        this.getContentPane().add(jLabel1, null);
        this.getContentPane().add(jTextField1, null);
        this.getContentPane().add(jTextField2, null);
        this.getContentPane().add(jTextField3, null);
    }

    void jTextField3_actionPerformed(ActionEvent e) {

}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void jTextField1_actionPerformed(ActionEvent e) {
}
}
class Display5_jTextField3_actionAdapter implements
java.awt.event.ActionListener {
    Display5 adaptee;
    Display5_jTextField3_actionAdapter(Display5 adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField3_actionPerformed(e);
    }
}
class Display5_jTextField1_actionAdapter implements
java.awt.event.ActionListener {
    Display5 adaptee;
    Display5_jTextField1_actionAdapter(Display5 adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField1_actionPerformed(e);
    }
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class Displaytest1

import javax.swing.*;
import java.awt.*;

public class Displaytest1 extends JFrame {
    JScrollPane jScrollPane1 = new JScrollPane();
    JTextArea jTextArea1 = new JTextArea();
    JScrollPane jScrollPane2 = new JScrollPane();
    JTextArea jTextArea2 = new JTextArea();
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JScrollPane jScrollPane3 = new JScrollPane();
    JTextArea jTextArea3 = new JTextArea();
    JScrollPane jScrollPane4 = new JScrollPane();
    JTextArea jTextArea4 = new JTextArea();
    JLabel jLabel5 = new JLabel();
    JScrollPane jScrollPane5 = new JScrollPane();
    JLabel jLabel6 = new JLabel();
    JTextArea jTextArea5 = new JTextArea();
    JScrollPane jScrollPane6 = new JScrollPane();
    JTextArea jTextArea6 = new JTextArea();
    JLabel jLabel7 = new JLabel();
    JScrollPane jScrollPane7 = new JScrollPane();
    JTextArea jTextArea7 = new JTextArea();
    JLabel jLabel8 = new JLabel();
    JScrollPane jScrollPane8 = new JScrollPane();
    JTextArea jTextArea8 = new JTextArea();
    JLabel jLabel9 = new JLabel();
    JScrollPane jScrollPane9 = new JScrollPane();
    JTextArea jTextArea9 = new JTextArea();
    JLabel jLabel10 = new JLabel();
    JScrollPane jScrollPane10 = new JScrollPane();
    JLabel jLabel11 = new JLabel();
    JTextArea jTextArea10 = new JTextArea();
    JScrollPane jScrollPane11 = new JScrollPane();
    JTextArea jTextArea11 = new JTextArea();
    JLabel jLabel12 = new JLabel();

    public Displaytest1() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }

    private void jbInit() throws Exception {
        this.setSize(new Dimension(3500, 2000));
        this.setTitle(" Welcome to Flight Ticket Reservation");
        this.getContentPane().setLayout(null);
        jScrollPane1.setBounds(new Rectangle(12, 88, 29, 131));
        jTextArea1.setBackground(Color.lightGray);
        jTextArea1.setText("jTextArea1");
        jScrollPane2.setMinimumSize(new Dimension(25, 25));
        jScrollPane2.setBounds(new Rectangle(48, 88, 49, 131));
        jTextArea2.setBackground(Color.lightGray);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

jTextArea2.setText("jTextArea1");
jLabel1.setText("Flight schedule is");
jLabel1.setBounds(new Rectangle(318, 25, 189, 15));
jLabel2.setText("No.");
jLabel2.setBounds(new Rectangle(16, 60, 62, 15));
jLabel3.setText("Flight number");
jLabel3.setBounds(new Rectangle(46, 59, 85, 15));
jLabel4.setText(" D/M/Y ");
jLabel4.setBounds(new Rectangle(132, 60, 64, 15));
jScrollPane3.setPreferredSize(new Dimension(74, 25));
jScrollPane3.setBounds(new Rectangle(104, 88, 102, 131));
jTextArea3.setBackground(Color.lightGray);
jTextArea3.setText("jTextArea3");
jScrollPane4.setBounds(new Rectangle(213, 88, 39, 131));
jTextArea4.setBackground(Color.lightGray);
jTextArea4.setText("jTextArea4");
jLabel5.setText("From");
jLabel5.setBounds(new Rectangle(222, 60, 46, 15));
jScrollPane5.setBounds(new Rectangle(267, 87, 38, 131));
jLabel6.setText("To");
jLabel6.setBounds(new Rectangle(278, 60, 35, 15));
jTextArea5.setBackground(Color.lightGray);
jTextArea5.setText("jTextArea5");
jScrollPane6.setBounds(new Rectangle(314, 88, 49, 131));
jTextArea6.setBackground(Color.lightGray);
jTextArea6.setText("jTextArea6");
jLabel7.setText(" Depart time ");
jLabel7.setBounds(new Rectangle(300, 60, 98, 15));
jScrollPane7.setBounds(new Rectangle(375, 88, 50, 131));
jTextArea7.setBackground(Color.lightGray);
jTextArea7.setText("jTextArea7");
jLabel8.setText(" Arrival time ");
jLabel8.setBounds(new Rectangle(369, 56, 74, 19));
jScrollPane8.setBounds(new Rectangle(434, 88, 61, 132));
jTextArea8.setBackground(Color.lightGray);
jTextArea8.setText("jTextArea8");
jLabel9.setText(" No. seat ");
jLabel9.setBounds(new Rectangle(439, 59, 93, 15));
jScrollPane9.setBounds(new Rectangle(507, 88, 69, 131));
jTextArea9.setBackground(Color.lightGray);
jTextArea9.setText("jTextArea9");
jLabel10.setText("Arive next day ");
jLabel10.setBounds(new Rectangle(509, 60, 108, 15));
jScrollPane10.setBounds(new Rectangle(585, 88, 76, 131));
jLabel11.setText(" Adult price ");
jLabel11.setBounds(new Rectangle(594, 60, 86, 15));
jTextArea10.setBackground(Color.lightGray);
jTextArea10.setText("jTextArea10");
jScrollPane11.setBounds(new Rectangle(671, 88, 67, 131));
jTextArea11.setBackground(Color.lightGray);
jTextArea11.setText("jTextArea11");
jLabel12.setText(" children price ");
jLabel12.setBounds(new Rectangle(671, 60, 99, 15));
this.getContentPane().add(jScrollPane2, null);
this.getContentPane().add(jLabel4, null);
this.getContentPane().add(jScrollPane3, null);
this.getContentPane().add(jLabel5, null);
this.getContentPane().add(jScrollPane4, null);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
this.getContentPane().add(jScrollPane5, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jScrollPane7, null);
this.getContentPane().add(jScrollPane8, null);
this.getContentPane().add(jScrollPane9, null);
this.getContentPane().add(jLabel10, null);
this.getContentPane().add(jScrollPane10, null);
this.getContentPane().add(jLabel11, null);
this.getContentPane().add(jScrollPane6, null);
this.getContentPane().add(jScrollPane1, null);
this.getContentPane().add(jLabel7, null);
this.getContentPane().add(jLabel8, null);
this.getContentPane().add(jLabel12, null);
this.getContentPane().add(jLabel9, null);
this.getContentPane().add(jLabel13, null);
this.getContentPane().add(jLabel11, null);
this.getContentPane().add(jLabel12, null);
this.getContentPane().add(jScrollPane11, null);
jScrollPane11.getViewport().add(jTextArea11, null);
jScrollPane1.getViewport().add(jTextArea1, null);
jScrollPane6.getViewport().add(jTextArea6, null);
jScrollPane10.getViewport().add(jTextArea10, null);
jScrollPane9.getViewport().add(jTextArea9, null);
jScrollPane8.getViewport().add(jTextArea8, null);
jScrollPane7.getViewport().add(jTextArea7, null);
jScrollPane5.getViewport().add(jTextArea5, null);
jScrollPane4.getViewport().add(jTextArea4, null);
jScrollPane3.getViewport().add(jTextArea3, null);
jScrollPane2.getViewport().add(jTextArea2, null);
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//Class Displaytest2

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;

public class Displaytest2 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JLabel jLabel3 = new JLabel();
    JTextField jTextField2 = new JTextField();
    JLabel jLabel4 = new JLabel();
    JTextField jTextField3 = new JTextField();
    JLabel jLabel5 = new JLabel();
    JTextField jTextField4 = new JTextField();
    JLabel jLabel6 = new JLabel();
    JTextField jTextField5 = new JTextField();
    JLabel jLabel7 = new JLabel();
    JTextField jTextField6 = new JTextField();
    JLabel jLabel8 = new JLabel();
    JTextField jTextField7 = new JTextField();
    JLabel jLabel9 = new JLabel();
    JTextField jTextField8 = new JTextField();
    JLabel jLabel10 = new JLabel();
    JTextField jTextField9 = new JTextField();
    JLabel jLabel11 = new JLabel();
    JTextField jTextField10 = new JTextField();

    public Displaytest2() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }

    private void jbInit() throws Exception {
        this.setSize(new Dimension(500, 391));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("Information about your trip");
        jLabel1.setBounds(new Rectangle(197, 31, 184, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("Flight number :");
        jLabel2.setBounds(new Rectangle(36, 71, 96, 15));
        jTextField1.setBackground(Color.lightGray);
        jTextField1.setText("");
        jTextField1.setBounds(new Rectangle(122, 67, 79, 21));
        jLabel3.setText(" D/M/Y :");
        jLabel3.setBounds(new Rectangle(45, 128, 60, 15));
        jTextField2.setBackground(Color.lightGray);
        jTextField2.setText(" ");
        jTextField2.setBounds(new Rectangle(125, 125, 105, 21));
        jLabel4.setText("From :");
        jLabel4.setBounds(new Rectangle(211, 69, 45, 15));
        jTextField3.setBackground(Color.lightGray);
        jTextField3.setText(" ");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

jTextField3.setBounds(new Rectangle(243, 69, 96, 21));
jTextField3.addActionListener(new
Displaytest2_jTextField3_actionAdapter(this));
jLabel5.setText("To :");
jLabel5.setBounds(new Rectangle(354, 71, 34, 15));
jTextField4.setBackground(Color.lightGray);
jTextField4.setText("");
jTextField4.setBounds(new Rectangle(374, 65, 99, 21));
jTextField4.addActionListener(new
Displaytest2_jTextField4_actionAdapter(this));
jLabel6.setText("Departure time :");
jLabel6.setBounds(new Rectangle(48, 175, 97, 15));
jTextField5.setBackground(Color.lightGray);
jTextField5.setText("");
jTextField5.setBounds(new Rectangle(155, 172, 65, 21));
jLabel7.setText("Arrival time :");
jLabel7.setBounds(new Rectangle(237, 176, 99, 15));
jTextField6.setBackground(Color.lightGray);
jTextField6.setText("");
jTextField6.setBounds(new Rectangle(341, 170, 64, 21));
jLabel8.setText("Type of trip :");
jLabel8.setBounds(new Rectangle(238, 129, 82, 15));
jTextField7.setBackground(Color.lightGray);
jTextField7.setText("jTextField7");
jTextField7.setBounds(new Rectangle(325, 124, 134, 21));
jLabel9.setText("Amount of adult :");
jLabel9.setBounds(new Rectangle(128, 219, 108, 15));
jTextField8.setBackground(Color.lightGray);
jTextField8.setText("");
jTextField8.setBounds(new Rectangle(258, 214, 110, 24));
jLabel10.setText("Amount of children :");
jLabel10.setBounds(new Rectangle(125, 253, 130, 15));
jTextField9.setBackground(Color.lightGray);
jTextField9.setText("");
jTextField9.setBounds(new Rectangle(257, 252, 108, 23));
jLabel11.setText("Total price :");
jLabel11.setBounds(new Rectangle(126, 293, 73, 15));
jTextField10.setBackground(Color.lightGray);
jTextField10.setText("");
jTextField10.setBounds(new Rectangle(258, 293, 108, 23));
this.getContentPane().add(jLabel11, null);
this.getContentPane().add(jTextField7, null);
this.getContentPane().add(jLabel8, null);
this.getContentPane().add(jTextField2, null);
this.getContentPane().add(jLabel3, null);
this.getContentPane().add(jTextField4, null);
this.getContentPane().add(jTextField1, null);
this.getContentPane().add(jLabel2, null);
this.getContentPane().add(jTextField6, null);
this.getContentPane().add(jLabel7, null);
this.getContentPane().add(jTextField5, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jLabel5, null);
this.getContentPane().add(jLabel4, null);
this.getContentPane().add(jTextField8, null);
this.getContentPane().add(jTextField9, null);
this.getContentPane().add(jTextField10, null);
this.getContentPane().add(jLabel9, null);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        this.getContentPane().add(jLabel10, null);
        this.getContentPane().add(jLabel11, null);
        this.getContentPane().add(jTextField3, null);
    }

    void jTextField4_actionPerformed(ActionEvent e) {

    }

    void jTextField3_actionPerformed(ActionEvent e) {

    }
}

class Displaytest2_jTextField4_actionAdapter implements
java.awt.event.ActionListener {
    Displaytest2 adaptee;

    Displaytest2_jTextField4_actionAdapter(Displaytest2 adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField4_actionPerformed(e);
    }
}

class Displaytest2_jTextField3_actionAdapter implements
java.awt.event.ActionListener {
    Displaytest2 adaptee;

    Displaytest2_jTextField3_actionAdapter(Displaytest2 adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField3_actionPerformed(e);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Class Displaytest3

import javax.swing.*;
import java.awt.*;

public class Displaytest3 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JLabel jLabel3 = new JLabel();
    JTextField jTextField2 = new JTextField();
    JLabel jLabel4 = new JLabel();
    JTextField jTextField3 = new JTextField();
    JLabel jLabel5 = new JLabel();
    JTextField jTextField4 = new JTextField();
    JLabel jLabel6 = new JLabel();
    JTextField jTextField5 = new JTextField();
    JLabel jLabel7 = new JLabel();
    JLabel jLabel8 = new JLabel();
    JTextField jTextField6 = new JTextField();
    JTextField jTextField7 = new JTextField();
    JLabel jLabel9 = new JLabel();
    JLabel jLabel10 = new JLabel();
    JLabel jLabel11 = new JLabel();
    JTextField jTextField8 = new JTextField();
    JTextField jTextField9 = new JTextField();
    JTextField jTextField10 = new JTextField();
    JLabel jLabel12 = new JLabel();
    JLabel jLabel13 = new JLabel();
    JTextField jTextField11 = new JTextField();
    JLabel jLabel14 = new JLabel();
    JLabel jLabel15 = new JLabel();
    JLabel jLabel16 = new JLabel();
    JTextField jTextField12 = new JTextField();
    JTextField jTextField13 = new JTextField();
    JTextField jTextField14 = new JTextField();
    JLabel jLabel17 = new JLabel();
    JTextField jTextField15 = new JTextField();
    JLabel jLabel18 = new JLabel();
    JTextField jTextField16 = new JTextField();
    JLabel jLabel19 = new JLabel();
    JTextField jTextField17 = new JTextField();
    JLabel jLabel110 = new JLabel();
    JTextField jTextField18 = new JTextField();
    JLabel jLabel111 = new JLabel();
    JTextField jTextField19 = new JTextField();

    public Displaytest3() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }

    private void jbInit() throws Exception {
        this.getContentPane().setBackground(Color.lightGray);
        this.setSize(new Dimension(600, 472));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.setLocation(50,50);
this.setTitle(" Welcome to Flight Ticket Reservation");
jLabel1.setText("Information about your trip ");
jLabel1.setBounds(new Rectangle(232, 20, 162, 15));
this.getContentPane().setLayout(null);
jLabel2.setText("Flight number :");
jLabel2.setBounds(new Rectangle(29, 68, 93, 15));
jTextField1.setBackground(Color.lightGray);
jTextField1.setText("jTextField1");
jTextField1.setBounds(new Rectangle(135, 66, 76, 21));
jLabel3.setText("From :");
jLabel3.setBounds(new Rectangle(228, 69, 45, 15));
jTextField2.setBackground(Color.lightGray);
jTextField2.setText("jTextField2");
jTextField2.setBounds(new Rectangle(292, 63, 96, 21));
jLabel4.setText("To :");
jLabel4.setBounds(new Rectangle(411, 69, 34, 15));
jTextField3.setBackground(Color.lightGray);
jTextField3.setText("jTextField3");
jTextField3.setBounds(new Rectangle(460, 63, 103, 21));
jLabel5.setText("D/M/Y :");
jLabel5.setBounds(new Rectangle(97, 113, 56, 15));
jTextField4.setBackground(Color.lightGray);
jTextField4.setText("jTextField4");
jTextField4.setBounds(new Rectangle(180, 110, 92, 21));
jLabel6.setText("Type of trip :");
jLabel6.setBounds(new Rectangle(298, 113, 71, 15));
jTextField5.setBackground(Color.lightGray);
jTextField5.setText("jTextField5");
jTextField5.setBounds(new Rectangle(387, 110, 114, 21));
jLabel7.setText("Departure time :");
jLabel7.setBounds(new Rectangle(57, 155, 92, 15));
jLabel8.setText("Arrival time :");
jLabel8.setBounds(new Rectangle(282, 153, 108, 15));
jTextField6.setBackground(Color.lightGray);
jTextField6.setText("jTextField6");
jTextField6.setBounds(new Rectangle(160, 149, 84, 21));
jTextField7.setBackground(Color.lightGray);
jTextField7.setCaretColor(Color.black);
jTextField7.setText("jTextField7");
jTextField7.setBounds(new Rectangle(392, 151, 90, 21));
jLabel9.setText("Amount of adult :");
jLabel9.setBounds(new Rectangle(13, 206, 108, 15));
jLabel10.setText("Amount of children :");
jLabel10.setBounds(new Rectangle(207, 203, 127, 15));
jLabel11.setText("Total price :");
jLabel11.setBounds(new Rectangle(406, 205, 75, 15));
jTextField8.setBackground(Color.lightGray);
jTextField8.setText("jTextField8");
jTextField8.setBounds(new Rectangle(126, 202, 68, 21));
jTextField9.setBackground(Color.lightGray);
jTextField9.setText("jTextField9");
jTextField9.setBounds(new Rectangle(325, 200, 61, 21));
jTextField10.setBackground(Color.lightGray);
jTextField10.setText("jTextField10");
jTextField10.setBounds(new Rectangle(475, 199, 86, 21));
jLabel12.setText("Return trip is :");
jLabel12.setBounds(new Rectangle(128, 244, 107, 15));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

jLabel13.setBounds(new Rectangle(231, 245, 95, 15));
jLabel13.setText("Flight number :");
jTextField11.setBackground(Color.lightGray);
jTextField11.setText("jTextField11");
jTextField11.setBounds(new Rectangle(354, 240, 107, 21));
jLabel14.setBounds(new Rectangle(29, 286, 46, 15));
jLabel14.setText("From :");
jLabel15.setBounds(new Rectangle(200, 286, 34, 15));
jLabel15.setText("To :");
jLabel16.setBounds(new Rectangle(397, 285, 56, 15));
jLabel16.setText("D\\M\\Y :");
jTextField12.setBackground(Color.lightGray);
jTextField12.setText("jTextField12");
jTextField12.setBounds(new Rectangle(78, 284, 114, 21));
jTextField13.setBackground(Color.lightGray);
jTextField13.setText("jTextField13");
jTextField13.setBounds(new Rectangle(246, 282, 118, 22));
jTextField14.setBackground(Color.lightGray);
jTextField14.setAlignmentY((float) 0.5);
jTextField14.setText("jTextField14");
jTextField14.setBounds(new Rectangle(465, 280, 106, 21));
jLabel17.setBounds(new Rectangle(78, 326, 92, 15));
jLabel17.setText("Departure time :");
jTextField15.setBackground(Color.lightGray);
jTextField15.setText("jTextField15");
jTextField15.setBounds(new Rectangle(189, 321, 109, 21));
jLabel18.setBounds(new Rectangle(323, 325, 108, 15));
jLabel18.setText("Arrival time :");
jTextField16.setBackground(Color.lightGray);
jTextField16.setText("jTextField16");
jTextField16.setBounds(new Rectangle(413, 321, 93, 21));
jLabel19.setBounds(new Rectangle(30, 367, 93, 15));
jLabel19.setText("Amount of adult :");
jTextField17.setBackground(Color.lightGray);
jTextField17.setText("jTextField17");
jTextField17.setBounds(new Rectangle(132, 364, 52, 21));
jLabel110.setBounds(new Rectangle(210, 366, 131, 15));
jLabel110.setText("Amount of children :");
jTextField18.setBackground(Color.lightGray);
jTextField18.setText("jTextField18");
jTextField18.setBounds(new Rectangle(334, 361, 65, 21));
jLabel111.setBounds(new Rectangle(416, 363, 80, 15));
jLabel111.setText("Total price :");
jTextField19.setBackground(Color.lightGray);
jTextField19.setText("jTextField19");
jTextField19.setBounds(new Rectangle(494, 358, 73, 21));
this.getContentPane().add(jLabel10, null);
this.getContentPane().add(jTextField8, null);
this.getContentPane().add(jTextField9, null);
this.getContentPane().add(jLabel11, null);
this.getContentPane().add(jTextField10, null);
this.getContentPane().add(jTextField7, null);
this.getContentPane().add(jTextField6, null);
this.getContentPane().add(jLabel7, null);
this.getContentPane().add(jLabel8, null);
this.getContentPane().add(jTextField3, null);
this.getContentPane().add(jLabel4, null);
this.getContentPane().add(jLabel2, null);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
this.getContentPane().add(jLabel3, null);
this.getContentPane().add(jLabel12, null);
this.getContentPane().add(jLabel13, null);
this.getContentPane().add(jLabel14, null);
this.getContentPane().add(jTextField13, null);
this.getContentPane().add(jLabel15, null);
this.getContentPane().add(jTextField14, null);
this.getContentPane().add(jLabel16, null);
this.getContentPane().add(jLabel19, null);
this.getContentPane().add(jTextField17, null);
this.getContentPane().add(jTextField15, null);
this.getContentPane().add(jTextField16, null);
this.getContentPane().add(jLabel18, null);
this.getContentPane().add(jLabel17, null);
this.getContentPane().add(jTextField5, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jTextField4, null);
this.getContentPane().add(jLabel5, null);
this.getContentPane().add(jLabel9, null);
this.getContentPane().add(jLabel11, null);
this.getContentPane().add(jTextField19, null);
this.getContentPane().add(jLabel111, null);
this.getContentPane().add(jTextField18, null);
this.getContentPane().add(jLabel110, null);
this.getContentPane().add(jTextField1, null);
this.getContentPane().add(jTextField2, null);
this.getContentPane().add(jTextField11, null);
this.getContentPane().add(jTextField12, null);
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class Displaytest4

import javax.swing.*;
import java.awt.*;

public class Displaytest4 extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JLabel jLabel3 = new JLabel();
    JLabel jLabel4 = new JLabel();
    JLabel jLabel5 = new JLabel();
    JLabel jLabel6 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JTextField jTextField2 = new JTextField();
    JTextField jTextField3 = new JTextField();
    JTextField jTextField4 = new JTextField();
    JTextField jTextField5 = new JTextField();

    public Displaytest4() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }

    private void jbInit() throws Exception {
        this.setSize(new Dimension(400, 325));
        this.setLocation(200,180);
        this.setTitle(" Welcome to Flight Ticket Reservation");
        jLabel1.setText("Information about customer");
        jLabel1.setBounds(new Rectangle(122, 36, 186, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("Customer ID :");
        jLabel2.setBounds(new Rectangle(80, 80, 84, 15));
        jLabel3.setText("Password :");
        jLabel3.setBounds(new Rectangle(80, 125, 65, 15));
        jLabel4.setText("Phonenumber :");
        jLabel4.setBounds(new Rectangle(80, 170, 90, 15));
        jLabel5.setText("ID :");
        jLabel5.setBounds(new Rectangle(80, 215, 34, 15));
        jLabel6.setText("Credit crad :");
        jLabel6.setBounds(new Rectangle(80, 260, 76, 15));
        jTextField1.setBackground(Color.lightGray);
        jTextField1.setText("jTextField1");
        jTextField1.setBounds(new Rectangle(170, 74, 132, 21));
        jTextField2.setBackground(Color.lightGray);
        jTextField2.setText("jTextField2");
        jTextField2.setBounds(new Rectangle(170, 119, 131, 21));
        jTextField3.setBackground(Color.lightGray);
        jTextField3.setText("jTextField3");
        jTextField3.setBounds(new Rectangle(170, 164, 130, 21));
        jTextField4.setBackground(Color.lightGray);
        jTextField4.setText("jTextField4");
        jTextField4.setBounds(new Rectangle(118, 208, 179, 21));
        jTextField5.setBackground(Color.lightGray);
        jTextField5.setText("jTextField5");
        jTextField5.setBounds(new Rectangle(170, 256, 132, 21));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
this.getContentPane().add(jLabel1, null);
this.getContentPane().add(jLabel2, null);
this.getContentPane().add(jLabel3, null);
this.getContentPane().add(jLabel4, null);
this.getContentPane().add(jLabel6, null);
this.getContentPane().add(jLabel5, null);
this.getContentPane().add(jTextField2, null);
this.getContentPane().add(jTextField1, null);
this.getContentPane().add(jTextField3, null);
this.getContentPane().add(jTextField4, null);
this.getContentPane().add(jTextField5, null);
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// class monitor (main)

package displayinformation;

public class monitor {
    public monitor() {
    }
    public static void main(String[] args) {
        password monitor1 = new password();
        monitor1.setVisible(true);
    }
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class password

package displayinformation;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class password extends JFrame {
    JLabel jLabel1 = new JLabel();
    JLabel jLabel2 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JLabel jLabel3 = new JLabel();
    JTextField jTextField2 = new JTextField();
    JButton jButton1 = new JButton();
    JButton jButton2 = new JButton();
    CustomerPassword pass = new CustomerPassword();
    String um, pw;
    public String a = "p" ;
    public password() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        jLabel1.setText("Please insert ");
        jLabel1.setBounds(new Rectangle(56, 51, 159, 15));
        this.getContentPane().setLayout(null);
        jLabel2.setText("User name :");
        jLabel2.setBounds(new Rectangle(76, 96, 78, 15));
        jTextField1.setText("");
        jTextField1.setBounds(new Rectangle(177, 90, 78, 21));
        jLabel3.setText("Password :");
        jLabel3.setBounds(new Rectangle(74, 148, 74, 15));
        jTextField2.setBounds(new Rectangle(178, 141, 78, 21));
        jTextField2.addActionListener(new
            password_jTextField2_actionAdapter(this));
        jButton1.setBounds(new Rectangle(81, 217, 81, 25));
        jButton1.setText("OK");
        jButton1.addActionListener(new
            password_jButton1_actionAdapter(this));
        jButton2.setBounds(new Rectangle(212, 216, 86, 25));
        jButton2.setText("CANCEL");
        jButton2.addActionListener(new
            password_jButton2_actionAdapter(this));
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel3, null);
        this.getContentPane().add(jButton1, null);
        this.getContentPane().add(jButton2, null);
        this.getContentPane().add(jLabel1, null);
        this.getContentPane().add(jTextField1, null);
        this.getContentPane().add(jTextField2, null);
        this.setSize(new Dimension(400, 350));
        this.setLocation(200,180);
        this.setTitle("Wellcome to Flight Ticket Reservation");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void jTextField2_actionPerformed(ActionEvent e) {

}

void jButton1_actionPerformed(ActionEvent e) {

    um = jTextField1.getText();
    System.out.print(um);
    pw = jTextField2.getText();
    System.out.print(pw);
    String test1 = "may";
    String test2 = "mata";

    int try1 = Integer.parseInt( pw );
    switch (try1) {
        case 1101 :
            { if (um .equals(test2)) {
                System.out.print(um);
                pass.setVisible(true);
                this.setVisible(false); }
              else
                this.setVisible(true);
              break; }
        case 1104 :
            { if (um .equals(test2)) {
                System.out.print(um);
                pass.setVisible(true);
                this.setVisible(false); }
              else
                this.setVisible(true);
              break; }
    } // end jButton1

    void jButton2_actionPerformed(ActionEvent e) {
        this.setVisible(false);
    }
}

class password_jTextField2_actionAdapter implements
java.awt.event.ActionListener {
    password adaptee;

    password_jTextField2_actionAdapter(password adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField2_actionPerformed(e);
    }
}

class password_jButton1_actionAdapter implements
java.awt.event.ActionListener {
    password adaptee;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

password_jButton1_actionAdapter(password adaptee) {
    this.adaptee = adaptee;
}
public void actionPerformed(ActionEvent e) {
    adaptee.jButton1_actionPerformed(e);
}
}

class password_jButton2_actionAdapter implements
java.awt.event.ActionListener {
    password adaptee;

    password_jButton2_actionAdapter(password adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jButton2_actionPerformed(e);
    }
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class connectdata
package displayinformation;

import java.sql.*;
import java.util.*;
public class connectdata {
private Connection connection;
private Statement statement;
private String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
private String url = "jdbc:odbc:ReserveTicket";
private String city = "", country = "";
static int size=0;
String insrt3; int srr3; String g;
int try3;
String dis2, dis3, dis4, dis5, dis6, dis7, dis8, dis9, dis10, dis11;
String dis12, dis13, dis15, dis18, dis19, dis20, dis21;
int dis1, dis16, dis17, dis14;
String dis22, dis23, dis24, dis25;

public String Display1( int custom ,String password) {
try {
// load database driver class
Class.forName(driver);
// establish connection to database
connection = DriverManager.getConnection(url);
// create Statement for querying database
statement = connection.createStatement();
String insrt1= " SELECT
Reservation.[Customer ID], Customer.[Phone number],
Customer.ID, Customer.[Credit Card ID], Customer.password,
Reservation.[Reserve date],"+ "Reservation.Roundtrip,
Reservation.[Date], Reservation.[Flight number],
Reservation.[Date return],Reservation.[Flighth number return],
Reservation.[Seat adult],"+ " Reservation.[Seat child],
Reservation.[Total seats], Reservation.[Seat adult return],
Reservation.[Seat child return] , Reservation.[Total seats
return], Reservation.Paid, "+ "Reservation.[Arrival city],
Reservation.[Departure city],Reservation.[Check-in-status],
Reservation.[Departure time], Reservation.[Arrival time],
Reservation.[Departure time return], Reservation.[Arrival
time return] FROM Reservation INNER JOIN Customer ON
Reservation.[Customer ID] = Customer.[Customer ID]"+" WHERE
(Reservation.[Customer ID] = "+custom+") AND
(Customer.password = '"+password+"')" ;
ResultSet rec2 = statement.executeQuery(insrt1);
while (rec2.next()) {

dis1 = rec2.getInt(1);
dis2 = rec2.getString(2);
dis3 = rec2.getString(3);
dis4 = rec2.getString(4);
dis5 = rec2.getString(5);
dis6 = rec2.getString(6);
dis7 = rec2.getString(7);
dis8 = rec2.getString(8);
dis9 = rec2.getString(9);
dis10 = rec2.getString(10);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        dis11 = rec2.getString(11);
        dis12= rec2.getString(12);
        dis13 = rec2.getString(13);
        dis14 = rec2.getInt(14);
        dis15 = rec2.getString(15);
        dis16 = rec2.getInt(16);
        dis17 = rec2.getInt(17);
        dis18 = rec2.getString(18);
        dis19 = rec2.getString(19);
        dis20 = rec2.getString(20);
        dis21 =rec2.getString(21);
        dis22 = rec2.getString(22);
        dis23 = rec2.getString(23);
        dis24 = rec2.getString(24);
        dis25 = rec2.getString(25);
    }

    // size = srr1.size();
    // System.out.println( size );
    rec2.close();
    statement.close();
    connection.close();

} //end try

catch (SQLException sqlException) {
    // detect problems loading database driver
}
catch (ClassNotFoundException classNotFound)
    System.out.println("Driver Not Found");
}
return dis2;
}

public int testtry1() {
    return dis1;
}
public String testtry2 () {
    return dis2;
}
public String testtry3 () {
    return dis3;
}
public String testtry4 () {
    return dis4;
}
public String testtry5 () {
    return dis5;
}
public String testtry6 () {
    return dis6;
}
public String testtry7 () {
    return dis7;
}
public String testtry8 () {
    return dis8;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    public String testtry9 () {
        return dis9;
    }
    public String testtry10 () {
        return dis10;
    }
    public String testtry11 () {
        return dis11;
    }
    public String testtry12 () {
        return dis12;
    }
    public String testtry13 () {
        return dis13;
    }
    public int testtry14 () {
        return dis14;
    }
    public String testtry15 () {
        return dis15;
    }
    public int testtry16 () {
        return dis16;
    }
    public int testtry17 () {
        return dis17;
    }
    public String testtry18 () {
        return dis18;
    }
    public String testtry19 () {
        return dis19;
    }
    public String testtry20 () {
        return dis20;
    }
    public String testtry21 () {
        return dis21;
    }
    public String testtry22 () {
        return dis22;
    }
    public String testtry23 () {
        return dis23;
    }
    public String testtry24 () {
        return dis24;
    }
    public String testtry25 () {
        return dis25;
    }
} //end connectodata end class

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class CustomerPassword
package displayinformation;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class CustomerPassword extends JFrame {
    JButton jButton1 = new JButton();
    JLabel jLabel1 = new JLabel();
    JTextField jTextField1 = new JTextField();
    Imformation inform = new Imformation();
    connectdata info;
    String display2,display3,display4,display5,display6,display7,
    String display8,display9,display10,display11 ;
    String display ,display12,display13,display15,display18;
    String display19,display20,display21,display22,display23;
    String display24,display25 ;
    int display1 ,display16,display17 ,display14;
    public CustomerPassword() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        jButton1.setBounds(new Rectangle(164, 208, 55, 25));
        jButton1.setText("OK");
        jButton1.addActionListener(new
        CustomerPassword_jButton1_actionAdapter(this));
        this.getContentPane().setLayout(null);
        this.setSize(new Dimension(400, 292));
        this.setLocation(200,180);
        this.setTitle("Wellcome to Flight Ticket Reservation");
        jLabel1.setText("Please insert customer password");
        jLabel1.setBounds(new Rectangle(115, 68, 198, 15));
        jTextField1.setBounds(new Rectangle(151, 104, 84, 21));
        jTextField1.addActionListener(new
        CustomerPassword_jTextField1_actionAdapter(this));
        jLabel2.setText("customerID");
        jLabel2.setBounds(new Rectangle(161, 142, 76, 15));
        jTextField2.setBounds(new Rectangle(151, 165, 82, 21));
        this.getContentPane().add(jTextField1, null);
        this.getContentPane().add(jTextField2, null);
        this.getContentPane().add(jButton1, null);
        this.getContentPane().add(jLabel2, null);
        this.getContentPane().add(jLabel1, null);
    }

    void jButton1_actionPerformed(ActionEvent e) {
        String password1 = this.jTextField1.getText();
        String custom = this.jTextField2.getText();
        System.out.print(password1);
        int cus =Integer.parseInt( custom);
        info = new connectdata ();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

display = info.Display1(cus,password1);
display1 = info.testtry1();
display2 = info.testtry2();
display3 = info.testtry3();
display4 = info.testtry4();
display5 = info.testtry5();
display6 = info.testtry6();
display7 = info.testtry7();
display8 = info.testtry8();
display9 = info.testtry9();
display10 = info.testtry10();
display11 = info.testtry11();
display12 = info.testtry12();
display13 = info.testtry13();
display14 = info.testtry14();
display15 = info.testtry15();
display16 = info.testtry16();
display17 = info.testtry17();
display18 = info.testtry18();
display19 = info.testtry19();
display20 = info.testtry20();
display21 = info.testtry21();
display22 = info.testtry22();
display23 = info.testtry23();
display24 = info.testtry24();
display25 = info.testtry25();

System.out.println(display8);
if (( display16 ==0) && ( display17==0) ) {
    inform.setVisible(true);
    this.setVisible(false);
    inform.jTextArea1.setText("your Customer ID is "
+ display1 + "\nPhone Number is " +display2
+"\nID is "+display3+ "\nCredit card is " + display4
+ "\npassword is " +display5+"\nReserve date is " + display6
+"\nRound trip is " + display7+"\nDate Departure is " +display8
+ "\nFlight number is " +display9+"\nFrom - To " + display20
+ " to " + display19 + "\n Departure is" + display22 + "and
Arrival time is" + display23+"\nSeat adult is " + display12
+ "\nSeat child is " +display13 + "\nTotal seats " +
display14+"\nPaid = " +display18 );
}

else {

    inform.setVisible(true);
    this.setVisible(false);
    inform.jTextArea1.setText("your Customer ID is "
+display1+ "\nPhone Number is " +display2 + "\nID is
"+display3+"\nCredit card is " + display4 + "\npassword is
"+display5+"\nReserve date is " + display6+"\nRound trip is
" + display7+"\nDate Departure is " +display8+"\nFlight number
is " +display9+"\nFrom - To " + display20 + " to " +
display19+"\n Departure is" + display22 + "and Arrival time is" +
display23+"\nSeat adult is " + display12+"\nSeat child is "
+display13+"\nTotal seats " + display14+"\n Date Return is
" + display10+"\n Flight Return is " + display11+"\n From -To
" + display19 + " to " + display20

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

+"\\n Departure return is"+ display24 +"and Arrival time return
is"+ display25+"\\nSeat adult return is  "+ display15
+"\\nSeat child return is  "+display16+"\\nTotal seats  " +
(display17+display14) +"\\nPaid =  " +display18 );

        } // end else
    }

    void jTextField1_actionPerformed(ActionEvent e) {

    }
    JLabel jLabel2 = new JLabel();
    JTextField jTextField2 = new JTextField();

}

class CustomerPassword_jButton1_actionAdapter implements
java.awt.event.ActionListener {
    CustomerPassword adaptee;

    CustomerPassword_jButton1_actionAdapter(CustomerPassword
adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jButton1_actionPerformed(e);
    }
}

class CustomerPassword_jTextField1_actionAdapter implements
java.awt.event.ActionListener {
    CustomerPassword adaptee;

    CustomerPassword_jTextField1_actionAdapter (CustomerPassword
adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.jTextField1_actionPerformed(e);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//class Imformation
package displayinformation;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Imformation extends JFrame {
    JLabel jLabel1 = new JLabel();
    JButton jButton1 = new JButton();
    JButton jButton2 = new JButton();
    JScrollPane jScrollPane1 = new JScrollPane();
    JTextArea jTextArea1 = new JTextArea();
    public Imformation() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    private void jbInit() throws Exception {
        this.getContentPane().setLayout(null);
        jLabel1.setText("The detail of your information");
        jLabel1.setBounds(new Rectangle(118, 52, 201, 15));
        jButton1.setBounds(new Rectangle(97, 243, 86, 25));
        jButton1.setText("OK");
        jButton1.addActionListener(new
Information_jButton1_actionAdapter(this));
        jButton2.setBounds(new Rectangle(223, 244, 85, 25));
        jButton2.setText("CANCL");
        jScrollPane1.setBounds(new Rectangle(76, 82, 219, 140));
        jTextArea1.setText("jTextArea1");
        this.getContentPane().add(jLabel1, null);
        this.getContentPane().add(jButton1, null);
        this.getContentPane().add(jButton2, null);
        this.getContentPane().add(jScrollPane1, null);
        jScrollPane1.getViewport().add(jTextArea1, null);
        this.setSize(new Dimension(400, 350));
        this.setLocation(200,180);
        this.setTitle("Wellcome to Flight Ticket Reservation");
    }
    void jButton1_actionPerformed(ActionEvent e) {
    }
}

class Imformation_jButton1_actionAdapter implements
java.awt.event.ActionListener {
    Imformation adaptee;

    Imformation_jButton1_actionAdapter(Imformation adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee(jButton1_actionPerformed(e));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// class TestMediaCall

import javax.telephony.media.*;
import net.sourceforge.gjtapi.media.*;
import java.io.*;
import javax.telephony.callcontrol.CallControlTerminalConnection;
import net.sourceforge.gjtapi.*;
import javax.telephony.*;

public class TestMediaCall extends TestMakeCall {

public static void main(java.lang.String[] args) {
    // Report error if insufficient parameters passed in
    if (args.length < 2) {
        System.err.println(
            "Usage: java net.sourceforge.gjtapi.test.TestMediaCall from
to");
        System.exit(1);
    }
    test("net.sourceforge.gjtapi.raw.xtapi.XtapiProvider; XtapiSp =
TAPI",
        args[0], args[1]);
}

public static void test(String providerName, String fromAddr, String
toAddr) {
    java.io.PrintStream out = System.out;
    out.println("In Media Test Script...");

    // Get a JTAPI Peer
    JtapiPeer peer = null;
    try {
        peer = JtapiPeerFactory.getJtapiPeer(
            "net.sourceforge.gjtapi.GenericJtapiPeer");
        out.println("1.1: Successfully loaded the JTAPI Peer");
    }
    catch (JtapiPeerUnavailableException jpue) {
        out.println("1.1: Failed to locate Peer with the factory");
        jpue.printStackTrace();
        System.exit(1);
    }

    // Ask it for the TAPI Provider
    Provider prov = null;
    try {
        prov = peer.getProvider(providerName);
        out.println("1.2: Successfully loaded Provider");
    }
    catch (ProviderUnavailableException pue) {
        out.println("1.2: Failed to load Provider");
        pue.printStackTrace();
        System.exit(1);
    }

    try {
        // Load the Terminal
        out.print("2.1: Attempting to get address...");
        Address addr = prov.getAddress(fromAddr);
        out.println(" success.");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Make the Call
out.print("2.3: Attempting to create call...");
Call c = prov.createCall();
out.println(" success.");

out.print("4.3: Attempting to get terminals for an
address...");
Terminal[] ts = addr.getTerminals();
out.println(" success.");

out.print("3.1: Attempting to connect call...");
c.connect(ts[0], addr, toAddr);
out.println(" success.");

// Notify progress
System.out.println("Call initiated...");

// answer args[1]
out.print("3.2: Attempting to get call connections...");
Connection cons[] = c.getConnections();
out.println(" success.");

// force a wait
prompt();

// Create a media service
out.print("Attempting to create a Media Service...");
GenericMediaService ms = new GenericMediaService(
(MediaProvider) prov);
out.println(" success.");

// find the outgoing terminal
out.print("2.4: Attempting to get a Terminal by name...");
fromAddr = fromAddr + ":D-Link DFM-560EL External Modem";
Terminal t = prov.getTerminal(fromAddr);
out.println(" success.");

// bind the media service
out.print("13.1: Attempting to bind a Media Service to a
terminal...");
ms.bindToTerminal(null, t);
out.println(" success.");

// generate a signal
out.print("13.6: Attempting to generate DTMF on a Media
Service...");
ms.sendSignals("411", null, null);
out.println(" success.");

// detect a signal
out.print("13.7: Attempting to detect DTMF on a Media
Service...");
SignalDetectorEvent sde = ms.retrieveSignals(5, null, null,
null);
out.println(" success: " + sde.getSignalString());

// play a message
out.print("13.4: Attempting to play on a Media Service...");
ms.play("Greeting.wav", 0, null, null);
out.println(" success.");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// release the media service
out.print("13.8: Attempting to release a Media Service...");
ms.release();
out.println(" success.");

}
catch (Exception e) {
    out.println(" failure: " + e);
    e.printStackTrace();
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสือและเอกสารอ้างอิง

- [1] กิตติ ภัคดีวัฒนะกุล , “คัมภีร์ Java”, เคพีที คอมพ์ แอนด์ คอนซัลท์ จำกัด , 2546
- [2] SPENCER ROBERTS, “essential JTAPI JAVA TELEPHONY API : Design Telecom Projects with you”, Prentice Hall Inc, 1999
- [3] การใช้งาน Microsoft SQL Server 2000 Step by Step , กรุงเทพฯ, สามย่าน.com, 2544
- [4] JAVA How to Program, Fifth Edition, Prentice Hall Inc, 2003
- [5] <http://java.sun.com/products/jtapi/>
- [6] <http://www.javaolympus.com/J2SE/MEDIA/JTAPI/JTAPI.jsp>
- [7] www.sourceforge.net
- [8] www.brekeke.com/en/products/jtapi/jtapi_en.html



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้