

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ภาษาเอสคิวแอลเชิงวัตถุสัมพันธ์สำหรับออราเคิล 10g

ORACLE 10G OBJECT RELATIONAL SQL



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา ๒๕๕๐

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ภาษาเอสคิวแอลเชิงวัตถุสัมพันธ์สำหรับออบราเคิล 10g

Oracle 10g Object Relational SQL

ผู้จัดทำ

1. นางสาวพรทิพย์ ชูวิเศษวณิชย์ รหัสนักศึกษา 47010483
2. นางสาวพุทธมาศ สุวรรณแสง รหัสนักศึกษา 47010539



(Handwritten signature)
(รศ. ดร. ศุภมิตร จิตตะยโสธร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาเอสคิวแอลเชิงวัตถุสัมพันธ์สำหรับออรากิล 10g

นางสาวพรทิพย์ ชูวิเศษวณิชช์ 47010483

นางสาวพุทธมาศ สุวรรณแสง 47010539

รศ. ดร. ศุภมิตร จิตตะขโศธร อาจารย์ที่ปรึกษา

ปีการศึกษา 2550

บทคัดย่อ

ในอดีตที่ผ่านมาได้มีการค้นคว้าและพัฒนาระบบฐานข้อมูลขึ้นหลากหลายรูปแบบ และหลากหลายแนวความคิด ซึ่งระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database System) นั้นได้รับความนิยมและแพร่หลายสูงสุดในการใช้งาน แต่ถึงอย่างไรก็ตามเนื่องจากปัจจุบัน ในการออกแบบและพัฒนาระบบ ได้หันมาใช้แนวคิดเชิงวัตถุกันอย่างแพร่หลาย ทั้งในด้านการเขียนโปรแกรม และในด้านการจัดการฐานข้อมูล จึงได้มีการนำข้อดีของแนวคิดเชิงวัตถุเข้ามาผสมผสานกับระบบฐานข้อมูลเชิงสัมพันธ์ ทำให้เกิดแนวคิดของระบบฐานข้อมูลที่เรียกว่าระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database System) ซึ่งสามารถรองรับการใช้งานข้อมูลที่มีความซับซ้อนได้มีประสิทธิภาพมากกว่า

โครงการนี้จึงได้ทำการศึกษาแนวความคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์ โดยได้เลือกใช้ระบบการจัดการฐานข้อมูลออรากิล 10g (Oracle 10g) เพื่อนำมาทดสอบแนวความคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์ และได้ทำการออกแบบฐานข้อมูล President ซึ่งเป็นฐานข้อมูลของประธานาธิบดีในสหรัฐอเมริกาที่บริษัทไอบีเอ็มใช้ โดยได้ทำการออกแบบฐานข้อมูลใหม่จากเดิมที่เป็นฐานข้อมูลเชิงสัมพันธ์ให้อยู่ในรูปของฐานข้อมูลเชิงวัตถุสัมพันธ์

นอกจากนี้ยังได้ทำการออกแบบฐานข้อมูลรถซึ่งเก็บข้อมูลละเอียดของรถ 3 ประเภทคือ รถยนต์ รถกระบะ และรถจักรยานยนต์ การออกแบบฐานข้อมูลของรถนี้ได้ออกแบบตามหลักการของฐานข้อมูลเชิงสัมพันธ์ (Relational Database) และฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database) โดยในการออกแบบตามหลักการของฐานข้อมูลเชิงวัตถุสัมพันธ์นั้นได้ใช้แนวความคิดเชิงวัตถุที่ฐานข้อมูลเชิงวัตถุสัมพันธ์สนับสนุนเข้ามาช่วยในการออกแบบ จากนั้นทำการเรียกค้นข้อมูลบนฐานข้อมูลทั้งสองแบบเพื่อแสดงให้เห็นถึงข้อดีของฐานข้อมูลเชิงวัตถุสัมพันธ์

Oracle 10g Object Relational SQL

Ms. Pornthip Choowisetwanitch 47010483

Ms. Pootthamas Suwannasaeng 47010539

Assoc. Prof. Dr. Suphamit Chittayasothorn Advisor

Academic Year 2007

ABSTRACT

In the past, there have been research and development database systems in many data models. Relational database system is one of the most popular data models. But nowadays, the use of object-oriented concepts is widely used in design and development both of programming and DBMS. So the advantage concepts of object oriented are applied to relational database which produce a new concept called object relational database that supports complex data better than relational database.

In this project, we have studied a concept of object relational database by using Oracle 10g to support and testing. We have designed a new presidential database, a database of presidents in the USA that is widely used by IBM, which is a relational database into object relational database and test it using an object relational database language.

Moreover, we have designed a vehicle database that includes the parts of car, pickup and motorcycle. Designing a vehicle database, we have used concept of relational database and object relational database. In case of object relational database concept we have used object-oriented concept which supported by object relational database. Finally, we have queried data on both databases to show the advantage of object relational database.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้เป็นอย่างดี ด้วยคำแนะนำ คำปรึกษา และความช่วยเหลือจาก อาจารย์ศุภมิตร จิตตะยโสธร ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการที่เอาใจใส่และเป็นห่วงเสมอมา ขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา และครอบครัวอันเป็นที่เคารพรักยิ่ง ที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้สามารถทำปริญญานิพนธ์ฉบับนี้สำเร็จลงด้วยดี คุณค่าและประโยชน์อันพึงมาจากปริญญานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกท่านและผู้ที่เกี่ยวข้องศึกษา

พรทิพย์ ชูวิเศษวิชช์
พุทธมาศ สุวรรณแสง

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	X
สารบัญภาพ	XII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	2
1.4 ขอบเขตของโครงการ	2
1.5 วิธีการดำเนินงาน	2
1.6 ส่วนประกอบของรายงาน	3
บทที่ 2 ฐานข้อมูลเชิงวัตถุสัมพันธ์และภาษาเอสคิวแอลสำหรับฐานข้อมูลเชิงวัตถุสัมพันธ์	4
2.1 ฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database)	4
2.1.1 โครงสร้างข้อมูล	4
2.1.2 ภาษาที่ใช้ในการจัดการฐานข้อมูล	5
2.1.3 กลไกที่ใช้ในการเข้าถึงข้อมูล	6
2.1.4 ข้อดีของฐานข้อมูลเชิงวัตถุสัมพันธ์	6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.2 แนวคิดเชิงวัตถุและรูปแบบของเอสคิวแอล 3 (SQL3 Features).....	8
2.2.1 องค์ประกอบของออปเจกต์.....	8
2.2.1.1 ออปเจกต์ไอดี (OID - Object Identity).....	8
2.2.1.2 คุณสมบัติ (Property).....	8
2.2.1.3 เมธอด (Method).....	8
2.2.2 แนวคิดเชิงวัตถุ.....	8
2.2.2.1 คุณสมบัติการห่อหุ้ม (Encapsulation).....	8
2.2.2.1.1 ไพรเวท (Private).....	8
2.2.2.1.2 โพรเทค (Protected).....	9
2.2.2.1.3 พับลิก (Public).....	9
2.2.2.2 คุณสมบัติการสืบทอด (Inheritance).....	9
2.2.2.2.1 การสืบทอดเพียงคลาสเดียว (Single Inheritance).....	9
2.2.2.2.2 การสืบทอดมากกว่าหนึ่งคลาส (Multiple Inheritance).....	9
2.2.2.3 คุณสมบัติโพลิมอร์ฟิซึม (Polymorphism).....	10
2.2.2.3.1 การโอเวอร์โหลด (Overloading).....	10
2.2.2.3.2 การโอเวอร์ไรด์ (Overriding).....	10
2.2.3 รูปแบบของเอสคิวแอล 3.....	10
2.2.3.1 มาตรฐานเอสคิวแอล 3.....	10
2.2.3.2 ชนิดข้อมูลพื้นฐานที่เพิ่มเข้าไป (Extended Base Types).....	11
2.2.3.3 โรว์ไทป์ (Row Types).....	12
2.2.3.4 ชนิดข้อมูลที่ใช้สร้างขึ้นเอง (User-Defined Types).....	12
2.2.3.5 ฟังก์ชันที่ใช้สร้างขึ้นเอง (User-Defined Functions).....	13
2.2.3.6 ซับไทป์และซูเปอร์ไทป์ (Sub-Types and Super-Types).....	14
2.2.3.7 ซับเทเบิลและซูเปอร์เทเบิล (Sub-Tables and Super-Tables).....	15
2.2.3.8 ชนิดข้อมูลอ้างอิงและออปเจกต์ไอดี (Reference Types and Object Identity).....	16
2.2.3.9 ชนิดข้อมูลที่เป็นคอลเลกชัน (Collection Types).....	16

สารบัญ (ต่อ)

หน้า

2.2.3.10 การทำคิวรีในเอสคิวแอล 3	16
บทที่ 3 ภาษาเอสคิวแอลสำหรับฐานข้อมูลเชิงวัตถุสัมพันธ์บนออราเคิล10g	
(Object Relational SQL on Oracle 10g).....	18
3.1 ศัพท์เฉพาะและคุณสมบัติของออราเคิล 10g ที่มีในฐานข้อมูลเชิงวัตถุสัมพันธ์	18
3.1.1 ชนิดข้อมูลออบเจกต์ (Object Type)	18
3.1.2 ออบเจกต์หรืออินสแตนซ์ (Objects or Instances)	19
3.1.3 เมธอดของออบเจกต์ (Object Methods)	19
3.1.3.1 เมมเบอร์เมธอด (Member Methods)	19
3.1.3.2 เมธอดที่ใช้ในการเปรียบเทียบ (Methods for Comparing Objects)	19
3.1.3.2.1 แมพเมธอด (Map Methods)	19
3.1.3.2.2 ออเดอร์เมธอด (Order Methods)	20
3.1.3.3 สแตติกเมธอด (Static Methods)	20
3.1.3.4 คอนสตรัคเตอร์เมธอด (Constructor Methods)	20
3.1.4 การสืบทอดของชนิดข้อมูลที่เป็นออบเจกต์	20
3.1.5 การเปลี่ยนแปลงชนิดข้อมูลออบเจกต์	21
3.1.6 ออบเจกต์เทเบิล (Object Tables)	21
3.1.7 ไร้วออบเจกต์ (Row Objects)	21
3.1.8 คอลัมน์ออบเจกต์ (Column Objects)	21
3.1.9 ข้อมูลชนิดอ้างอิง (References)	21
3.1.10 คอลเลกชันไทป์ (Collection Type)	22
3.1.11 PL/SQL	22
3.2 การใช้คำสั่งภาษาเอสคิวแอลในการจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์	22

สารบัญ (ต่อ)

	หน้า
3.2.1 การใช้คำสั่งภาษาเอสคิวแอลกับชนิดข้อมูลออบเจกต์ และการใช้ข้อมูลชนิดอ้างอิง	22
3.2.1.1 การสร้างชนิดข้อมูลออบเจกต์	22
3.2.1.2 การสร้างออบเจกต์เทเบิล (Object Table).....	23
3.2.1.3 การเพิ่มข้อมูลชนิดที่เป็นออบเจกต์.....	23
3.2.1.4 การใช้คอนสเตรนท (Constraint).....	24
3.2.1.5 การสร้างอินเด็กซ์	25
3.2.1.6 Reference การอ้างอิง.....	25
3.2.2 การประกาศและสร้างเมธอดในชนิดข้อมูลออบเจกต์	26
3.2.2.1 เมมเบอร์เมธอด (Member Methods).....	26
3.2.2.2 เมธอดที่ใช้ในการเปรียบเทียบ (Methods for Comparing Objects).....	27
3.2.2.2.1 แมพเมธอด (Map Methods).....	27
3.2.2.2.2 ออเดอร์เมธอด (Order Methods).....	28
3.2.2.3 สแตติกเมธอด (Static Methods).....	28
3.2.3 การสืบทอด (Inheritance)	29
3.2.4 โพลีมอร์ฟิซึม (Polymorphism).....	30
3.2.5 คอลเลกชัน (Collection Type)	32
3.2.5.1 วารีเอเบิลอาร์เรย์ (VARRAY)	32
3.2.5.2 เนสต์เต็ดเทเบิล (Nested Table).....	33
 บทที่ 4 การทดสอบ	 35
4.1 การออกแบบฐานข้อมูลประธานาธิบดีในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์.....	35
4.1.1 คำสั่งต่างๆ ที่ใช้ในการสร้างฐานข้อมูล Presdb ในรูปแบบของ ฐานข้อมูลเชิงวัตถุสัมพันธ์	 38

สารบัญ (ต่อ)

	หน้า
4.1.1.1 คำสั่งทั้งหมดที่ใช้ในการสร้างชนิดข้อมูลออบเจกต์ (Object Type).....	38
4.1.2 คำสั่งทั้งหมดที่ใช้ในการสร้างตาราง.....	40
4.1.3 ตัวอย่างคำสั่งเพิ่มข้อมูลลงตาราง.....	41
4.1.4 ตัวอย่างการคิวรีข้อมูล.....	43
4.2 การออกแบบฐานข้อมูลในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์	44
4.2.1 รายละเอียดของฐานข้อมูล.....	46
4.2.1.1 ฐานข้อมูล.....	46
4.2.1.2 ชิ้นส่วนเครื่องยนต์ (Engine Components)	50
4.2.1.3 อุปกรณ์ส่วนควบคุมของเครื่องยนต์ (Engine Peripheral Components).....	52
4.2.1.4 ระบบไฟฟ้า (Electrical System).....	53
4.2.1.5 ระบบน้ำมันเชื้อเพลิง (Fuel System).....	55
4.2.1.6 ระบบหล่อเย็น (Cooling System)	56
4.2.1.7 ระบบไอเสีย (Exhaust System)	57
4.2.1.8 ระบบส่งกำลัง (Transmission System).....	58
4.2.1.9 ระบบเบรก (Brake System).....	60
4.2.1.10 ระบบกันสะเทือน (Suspension System)	62
4.2.1.11 ระบบบังคับเลี้ยว (Steering System).....	63
4.2.1.12 ชิ้นส่วนตัวถัง (Bodywork)	65
4.2.1.13 ชุดกระจก (Mirror)	66
4.2.1.14 ชิ้นส่วนภายในและอุปกรณ์ตกแต่ง (Accessories).....	67
4.2.1.15 อุปกรณ์อื่นๆ (Others).....	69
4.2.2 ตัวอย่างคำสั่งที่ใช้ในการสร้างชนิดข้อมูลในฐานข้อมูล	70
4.2.2.1 ตัวอย่างคำสั่งการสร้างชนิดข้อมูล	70
4.2.2.2 ตัวอย่างคำสั่งการสร้างชนิดข้อมูลที่มีคุณสมบัติการสืบทอด และ โพลีมอร์ฟิซึม.....	73
4.2.2.3 ตัวอย่างคำสั่งการสร้างชนิดข้อมูลที่เป็นเน็สเต็ดเทเบิล.....	75

สารบัญ (ต่อ)

	หน้า
4.2.3 ตัวอย่างคำสั่งที่ใช้ในการสร้างตารางในฐานข้อมูล.....	76
4.2.3.1 ตัวอย่างคำสั่งที่ใช้ในการสร้างตาราง.....	76
4.2.4 ตัวอย่างคำสั่งที่ใช้ในการเพิ่ม (INSERT) ข้อมูลในฐานข้อมูล	77
4.2.5 ตัวอย่างคำสั่งที่ใช้ในการคิวรีข้อมูลในฐานข้อมูล.....	78
บทที่ 5 บทสรุปและวิจารณ์.....	82
5.1 สรุปผลการดำเนินงาน.....	82
5.2 วิจารณ์สิ่งที่ได้จากโครงการ.....	82
5.3 ปัญหาและอุปสรรค.....	82
5.3.1 ปัญหา.....	82
5.3.2 อุปสรรค.....	83
5.4 แนวทางแก้ไข.....	83
5.5 แนวทางในการพัฒนาต่อ.....	83
บรรณานุกรม.....	84
ภาคผนวก ก การติดตั้งฐานข้อมูลออรากิล 10g.....	85
ก.1 การติดตั้งฐานข้อมูลออรากิล 10g.....	85
ก.2 ขั้นตอนการสร้างฐานข้อมูลจาก Database Configuration Assistant ของออรากิล	89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางเปรียบเทียบฐานข้อมูลแบบต่างๆ กับฐานข้อมูลเชิงวัตถุสัมพันธ์	7
2.2 ตารางเปรียบเทียบแนวคิดฐานข้อมูลแบบต่างๆ กับฐานข้อมูลเชิงวัตถุสัมพันธ์	7
4.1 แสดงรายละเอียดของชนิดข้อมูล VEHICLE.....	46
4.2 แสดงรายละเอียดของชนิดข้อมูล MOTORCYCLE	47
4.3 แสดงรายละเอียดของชนิดข้อมูล AUTOMOBILE	48
4.4 แสดงรายละเอียดของชนิดข้อมูล CARBODYWORK_TYPE	49
4.5 แสดงรายละเอียดของชนิดข้อมูล PICKUPBODYWORK_TYPE	49
4.6 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE	50
4.7 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE_CAR.....	51
4.8 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE_MBIKE.....	51
4.9 แสดงรายละเอียดของชนิดข้อมูล ENGINEPERIPHERAL_TYPE_MBIKE.....	52
4.10 แสดงรายละเอียดของชนิดข้อมูล ENGINEPERIPHERAL_TYPE_CAR	53
4.11 แสดงรายละเอียดของชนิดข้อมูล ELECTRICALSYSTEM_TYPE_MBIKE.....	54
4.12 แสดงรายละเอียดของชนิดข้อมูล ELECTRICALSYSTEM_TYPE_CAR	54
4.13 แสดงรายละเอียดของชนิดข้อมูล FUELSYSTEM_TYPE_MBIKE.....	55
4.14 แสดงรายละเอียดของชนิดข้อมูล FUELSYSTEM_TYPE_CAR.....	56
4.15 แสดงรายละเอียดของชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR.....	56
4.16 แสดงรายละเอียดของชนิดข้อมูล EXHAUSTSYSTEM_TYPE_MBIKE	57
4.17 แสดงรายละเอียดของชนิดข้อมูล EXHAUSTSYSTEM_TYPE_CAR	58
4.18 แสดงรายละเอียดของชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.19 แสดงรายละเอียดของชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE_CAR	59
4.20 แสดงรายละเอียดของชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE_MBIKE.....	60
4.21 แสดงรายละเอียดของชนิดข้อมูล BRAKESYSTEM_TYPE_MBIKE.....	61
4.22 แสดงรายละเอียดของชนิดข้อมูล BRAKESYSTEM_TYPE_CAR	61
4.23 แสดงรายละเอียดของชนิดข้อมูล SUSPENSYSYSTEM_TYPE	62
4.24 แสดงรายละเอียดของชนิดข้อมูล SUSPENSIONSYSYSTEM_TYPE_CAR.....	63
4.25 แสดงรายละเอียดของชนิดข้อมูล SUSPENSYSYSTEM_TYPE_MBIKE	63
4.26 แสดงรายละเอียดของชนิดข้อมูล STEERINGSYSTEM_TYPE_CAR	64
4.27 แสดงรายละเอียดของชนิดข้อมูล CARBODYWORK_TYPE	65
4.28 แสดงรายละเอียดของชนิดข้อมูล MIRROR_TYPE_MBIKE	66
4.29 แสดงรายละเอียดของชนิดข้อมูล MIRROR_TYPE_CAR.....	67
4.30 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE	68
4.31 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE_CAR.....	68
4.32 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE_MBIKE	69
4.33 แสดงรายละเอียดของชนิดข้อมูล OTHERS_TYPE_CAR.....	70

สารบัญภาพ

รูปที่	หน้า
2.1 การสืบทอดเพียงคลาสเดียว (Single Inheritance).....	9
2.2 การสืบทอดมากกว่าหนึ่งคลาส (Multiple Inheritance)	10
2.3 ตาราง Branch ที่มี Address เป็นคอลัมน์โรว์ไทป์ (Row Type)	12
2.4 คำสั่งในการสร้างโรว์ไทป์ (Row Type)	12
2.5 คำสั่งในการสร้าง User-Defined Type และ User-Defined Function.....	13
2.6 Sub-Type และ Super-Type.....	14
2.7 คำสั่งในการสร้าง Sub-Type และ Super-Type.....	15
2.8 คำสั่งในการสร้าง Reference Type.....	16
2.9 ตัวอย่างใช้คำสั่ง SELECT ในฐานข้อมูลเชิงวัตถุสัมพันธ์.....	17
2.10 ตัวอย่างใช้คำสั่ง INSERT ในฐานข้อมูลเชิงวัตถุสัมพันธ์.....	17
2.11 ตัวอย่างใช้คำสั่ง UPDATE ในฐานข้อมูลเชิงวัตถุสัมพันธ์.....	17
3.1 แสดงชนิดข้อมูลออปเจกต์ (Object Type) และออปเจกต์อินสแตนซ์.....	18
3.2 คำสั่งการสร้างข้อมูลชนิดออปเจกต์.....	22
3.3 คำสั่งการสร้างออปเจกต์เทเบิล (Object Table).....	23
3.4 คำสั่งการเพิ่มข้อมูลชนิดออปเจกต์.....	23
3.5 คำสั่งการใช้คอนสเตรนท์กับ ออปเจกต์เทเบิล.....	24
3.6 คำสั่งการใช้คอนสเตรนท์กับตารางที่มีแอตทริบิวต์ เป็นชนิดข้อมูลออปเจกต์.....	24
3.7 คำสั่งการสร้างอินเด็กซ์บนชนิดข้อมูลออปเจกต์ ในตาราง.....	25
3.8 คำสั่งการจัดการเกี่ยวกับชนิดข้อมูลอ้างอิง.....	25

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
3.9 การใช้ขอบเขตกับชนิดข้อมูลแบบอ้างอิง (Reference)	26
3.10 คำสั่งการประกาศเมมเบอร์เมธอด	26
3.11 คำสั่งการสร้างเมมเบอร์เมธอด	27
3.12 คำสั่งการประกาศเมพเมธอด	27
3.13 คำสั่งการสร้างเมพเมธอด	27
3.14 คำสั่งการประกาศออดเดอร์เมธอด	28
3.15 คำสั่งการสร้างออดเดอร์เมธอด	28
3.16 คำสั่งการประกาศสแตติกเมธอด	28
3.17 คำสั่งการสร้างสแตติกเมธอด	29
3.18 ตัวอย่างโครงสร้างข้อมูลที่ทำให้การสืบทอด	29
3.19 คำสั่งการสร้างซูเปอร์ไทป์ (Supertype)	29
3.20 คำสั่งการสร้างซับไทป์ (Subtype)	30
3.21 ตัวอย่างโครงสร้างข้อมูลที่ใช้โพลิมอर्फิซึม	31
3.22 คำสั่งการสร้างชนิดข้อมูลออปเจกต์ (Object Type) ที่มีคุณสมบัติ NOT INSTANTIABLE	31
3.23 คำสั่งการสร้างซับไทป์ที่สืบทอดมาจากชนิดข้อมูลออปเจกต์ ที่เป็น NOT INSTANTIABLE	31
3.24 คำสั่งการสร้างซับไทป์ที่สืบทอดมาจากชนิดข้อมูลออปเจกต์ ที่เป็น NOT INSTANTIABLE 2	32
3.25 การสร้าง VARRAY จากชนิดข้อมูลแบบ ออปเจกต์	33
3.26 การเพิ่มข้อมูลชนิดที่เป็น VARRAY	33
3.27 การสร้างเนสต์เตดเทเบิล (Nested Table) จากชนิดข้อมูลแบบ ออปเจกต์	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
3.28 การสร้างตารางที่ภายในประกอบด้วยข้อมูลชนิดเน็สเต็ดเทเบิล (Nested Table).....	34
4.1 โครงสร้างโดยรวมของฐานข้อมูล Presdb ในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์	36
4.2 คำสั่งการสร้าง state_type	38
4.3 คำสั่งการสร้าง spouse_type และ marriage_type ที่เป็นเน็สเต็ดเทเบิล (Nested Table)	38
4.4 คำสั่งการสร้าง Hobby Type ที่เป็นเน็สเต็ดเทเบิล (Nested Table).....	38
4.5 คำสั่งการสร้าง admin_pr_vp_type.....	38
4.6 คำสั่งการสร้าง administration_type	38
4.7 คำสั่งการสร้าง president_type.....	39
4.8 คำสั่งการสร้าง president_type ในส่วนของ Body.....	39
4.9 คำสั่งการสร้าง election_type.....	40
4.10 คำสั่งทั้งหมดที่ใช้ในการสร้างตาราง	40
4.11 คำสั่งการเพิ่มข้อมูลลงตาราง state_table	41
4.12 คำสั่งการเพิ่มข้อมูลลงตาราง president_table.....	41
4.13 คำสั่งการเพิ่มข้อมูลลงตาราง admin_pr_vp_table	42
4.14 คำสั่งการเพิ่มข้อมูลลงตาราง administration_table	42
4.15 คำสั่งการคิวรี	43
4.16 ผลลัพธ์จากการคิวรี	43
4.17 คำสั่งการคิวรี	43
4.18 ผลลัพธ์จากการคิวรี	43
4.19 โครงสร้างโดยรวมของฐานข้อมูลในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์.....	45

สารบัญภาพ (ต่อ)

รูปที่	หน้า
4.20 ฐานข้อมูลรถ	46
4.21 แสดงความสัมพันธ์ระหว่าง CARBODYWORK_TYPE กับ CAR และ PICKUP	49
4.22 ชิ้นส่วนเครื่องยนต์ (Engine Components)	50
4.23 อุปกรณ์ส่วนควบคุมของเครื่องยนต์ (Engine Peripheral Components)	52
4.24 ระบบไฟฟ้า (Electrical System).....	53
4.25 ระบบน้ำมันเชื้อเพลิง (Fuel System)	55
4.26 ระบบหล่อเย็น (Cooling System).....	56
4.27 ระบบไอเสีย (Exhaust System)	57
4.28 ระบบส่งกำลัง (Transmission System).....	58
4.29 ระบบเบรก (Brake System).....	60
4.30 ระบบกันสะเทือน (Suspension System).....	62
4.31 ระบบบังคับเลี้ยว (Steering System).....	63
4.32 ชิ้นส่วนตัวถัง (Bodywork).....	65
4.33 ชุดกระจก (Mirror).....	66
4.34 ชิ้นส่วนภายในและอุปกรณ์ตกแต่ง (Accessories).....	67
4.35 อุปกรณ์อื่นๆ (Others).....	69
4.36 คำสั่งในการสร้างชนิดข้อมูล PARTNER.....	70
4.37 คำสั่งในการสร้างชนิดข้อมูล DESCRIPTION	71
4.38 แสดงความสัมพันธ์ระหว่างชนิดข้อมูล DESCRIPTION และ PARTNER	71
4.39 คำสั่งในการสร้างชนิดข้อมูล RADIATOR_TYPE	72

สารบัญภาพ (ต่อ)

รูปที่	หน้า
4.40 COOLINGSYSTEM_TYPE_CAR.....	72
4.41 คำสั่งในการสร้างชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR.....	73
4.42 คำสั่งการสร้างชนิดข้อมูล VEHICLE.....	73
4.43 คำสั่งการสร้างข้อมูล AUTOMOBILE.....	74
4.44 คำสั่งการสร้างชนิดข้อมูล CAR.....	74
4.45 คำสั่งการสร้างชนิดข้อมูล SUSPENSIONSYSTEM_TYPE.....	75
4.46 คำสั่งการสร้างชนิดข้อมูล CAR.....	75
4.47 แสดงโครงสร้างของข้อมูลชนิดเน็สเต็ดเทเบิล.....	76
4.48 คำสั่งการสร้างตารางขึ้นส่วนทั้งหมดในระบบเครื่องยนต์.....	76
4.49 คำสั่งการสร้างตารางชนิดที่มีคอตัมน์เป็นเน็สเต็ดเทเบิล.....	77
4.50 คำสั่งการเพิ่มข้อมูลลงในตาราง PARTNER_TAB.....	77
4.51 คำสั่งการเพิ่มข้อมูลลงในตาราง BRAKEPAD_TAB.....	78
4.52 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์.....	78
4.53 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงสัมพันธ์.....	79
4.54 คำสั่งแสดงผลพัทธ์การคิวรี.....	79
4.55 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์.....	80
4.56 ผลลัพธ์การคิวรี.....	80
4.57 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์.....	80
4.58 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงสัมพันธ์.....	81
4.59 ผลลัพธ์การคิวรี.....	81

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
ก.1 หน้าจอ Welcome เป็นหน้าจอแรกของการติดตั้ง.....	85
ก.2 หน้าจอ Specify File Locations ตัวโปรแกรมติดตั้งจะให้เลือก Source file ที่ใช้ติดตั้ง และให้เลือก Oracle Home และ Path ที่เก็บด้วย ซึ่งสามารถ เลือกใช้ค่า default ที่มีให้อยู่แล้วได้.....	85
ก.3 หน้าจอ Select Installation Type ให้เลือก Product สำหรับการติดตั้ง.....	86
ก.4 หน้าจอ Analyzing dependencies	86
ก.5 หน้าจอ Select Database Configuration ตัวโปรแกรมติดตั้งจะให้เลือกว่าจะสร้าง database ในระหว่างการติดตั้งหรือไม่ ในที่นี้เลือกไม่สร้าง database.....	87
ก.6 หน้าจอ Summary ตัวโปรแกรมติดตั้งจะสรุปข้อมูลการติดตั้งให้อีกครั้ง.....	87
ก.7 หน้าจอ Install.....	88
ก.8 หน้าจอ Configuration Assistants หลังจากติดตั้ง Oracle Database Software เรียบร้อยแล้วตัว โปรแกรมติดตั้งจะเข้าสู่หน้าจอนี้.....	88
ก.9 หน้าจอ Database Configuration Assistant : Welcome.....	89
ก.10 หน้าจอ Operations เลือก Create a Database.....	89
ก.11 ตั้งชื่อให้กับ database	90
ก.12 หน้าจอ Management Option เลือก Configure the Database with Enterprise Manager และเลือก Use Database Control for Database Management	90
ก.13 หน้าจอ Database Credentials เลือกใช้ password เดียวกันกับทุก Account จากนั้นกำหนด password ลงไป	91
ก.14 หน้าจอ Storage Options เลือก File System	91
ก.15 หน้าจอ Database File Locations เลือก Drive, Path ที่จะกำหนดให้เก็บไฟล์ฐานข้อมูล	92
ก.16 หน้าจอ Recovery Configuration ให้เลือก Enable Flash Recovery Area.....	92

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
ก.17 หน้าจอ Database Content เลือก Sample Schemas เพื่อสร้าง Schema สำหรับในทดสอบ.....	93
ก.18 หน้าจอ Initialization Parameters ให้เลือก tab Character Sets เลือก Character Set เป็น TH8TISASCII เพื่อให้สามารถใช้ภาษาไทย กับฐานข้อมูลได้อย่างไม่มีปัญหา.....	93
ก.19 หน้าจอ Database Storage มีสรุป Path ที่ใช้เก็บไฟล์.....	94
ก.20 หน้าจอ Confirmation สรุปสิ่งที่เลือกไปทั้งหมดให้ดู.....	94



บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของโครงการ

เป็นที่ทราบกันดีอยู่แล้วว่าการพัฒนาโปรแกรมในยุคปัจจุบัน มีแนวโน้มไปทางการโปรแกรมเชิงวัตถุมากขึ้น ทำให้เกิดความต้องการในการใช้ข้อมูลที่มีความซับซ้อนมากขึ้นตามไปด้วย ซึ่งระบบฐานข้อมูลเชิงสัมพันธ์แบบเดิมไม่สามารถรองรับได้อย่างเต็มที่ จึงได้เกิดแนวคิดที่จะมีการเพิ่มความสามารถในการจัดการข้อมูลในรูปแบบออบเจกต์ (object) ให้แก่ระบบฐานข้อมูลเชิงสัมพันธ์ ระบบฐานข้อมูลใหม่นี้เรียกว่า ระบบฐานข้อมูลเชิงวัตถุสัมพันธ์

ในโครงการนี้จึงได้ทำการศึกษาแนวคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์ แล้วทำการทดสอบแนวคิดต่างๆ ของฐานข้อมูลเชิงวัตถุสัมพันธ์ โดยใช้ระบบการจัดการฐานข้อมูลออรากิล 10g (Oracle 10g) ทำการทดสอบกับฐานข้อมูล presdb ซึ่งเป็นฐานข้อมูลของประธานาธิบดีในสหรัฐอเมริกา โดยทำการออกแบบใหม่ให้เป็นฐานข้อมูลเชิงวัตถุสัมพันธ์จากฐานข้อมูล presdb แบบเดิมที่เป็นฐานข้อมูลเชิงสัมพันธ์

นอกจากทำการทดสอบกับฐานข้อมูลประธานาธิบดีแล้ว ในโครงการนี้ยังได้ทำการออกแบบฐานข้อมูลตามแนวคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์และฐานข้อมูลเชิงสัมพันธ์ ในฐานข้อมูลที่ได้เก็บข้อมูลเกี่ยวกับอะไหล่รถ 3 ประเภทคือ รถยนต์ รถกระบะ และรถจักรยานยนต์ จากนั้นก็ทำการเรียกค้นข้อมูลรถจากฐานข้อมูลทั้งสองแบบ เพื่อแสดงถึงข้อดีของฐานข้อมูลเชิงวัตถุสัมพันธ์เปรียบเทียบกับฐานข้อมูลเชิงสัมพันธ์

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อศึกษาแนวความคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object-Relational Database)
- 1.2.2 เพื่อศึกษาการใช้งานออรากิล 10g ในส่วนที่ใช้จัดการกับฐานข้อมูลเชิงวัตถุสัมพันธ์
- 1.2.3 ทดสอบโครงสร้างข้อมูลเชิงวัตถุสัมพันธ์กับออรากิล 10g
- 1.2.4 ออกแบบฐานข้อมูลโดยใช้แนวคิดเชิงวัตถุที่ฐานข้อมูลเชิงวัตถุสัมพันธ์สนับสนุน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ได้รับความรู้เกี่ยวกับแนวความคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์
- 1.3.2 ได้รับความรู้เกี่ยวกับการใช้งานออรากิล 10g ทั้งสถาปัตยกรรมของออรากิล 10g และภาษาที่ใช้จัดการกับฐานข้อมูล เช่น เอสคิวแอล (SQL), พีแอลเอสคิวแอล (PL/SQL)
- 1.3.3 การออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์ให้สามารถตอบโจทย์ที่ผู้ใช้องการได้มากที่สุด
- 1.3.4 การทดสอบใช้ภาษาฐานข้อมูลกับโครงสร้างข้อมูลเชิงวัตถุสัมพันธ์

1.4 ขอบเขตของโครงการ

ศึกษาแนวคิดเชิงวัตถุสัมพันธ์ จากนั้นทำการทดสอบแนวคิดที่ได้ศึกษามา โดยทำการสร้างฐานข้อมูลเชิงวัตถุสัมพันธ์และใช้ภาษาฐานข้อมูลร่วมกับออรากิล 10g ทดสอบคำสั่งต่างๆ ในการสืบค้นข้อมูล นอกจากนั้นก็จะนำความรู้ทั้งหมดที่ได้ศึกษามามาใช้ในการออกแบบและพัฒนาแอปพลิเคชันที่เหมาะสมกับโครงสร้างข้อมูลเชิงวัตถุสัมพันธ์

1.5 วิธีการดำเนินงาน

- 1.5.1 ศึกษาหลักการและแนวคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์
- 1.5.2 ศึกษาการออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์
- 1.5.3 ศึกษาการใช้งานออรากิล 10g ในส่วนที่เกี่ยวข้องฐานข้อมูลเชิงวัตถุสัมพันธ์
- 1.5.4 ทดสอบการใช้งานออรากิล 10g ในส่วนที่เป็นฐานข้อมูลเชิงวัตถุสัมพันธ์ โดยทำการสร้างฐานข้อมูลเชิงวัตถุสัมพันธ์และทำการทดสอบการเรียกค้นข้อมูล ตามทฤษฎีที่ได้ศึกษา
- 1.5.5 ออกแบบฐานข้อมูลประธานาธิบดีที่จากเดิมเป็นฐานข้อมูลเชิงสัมพันธ์ให้เป็นฐานข้อมูลเชิงวัตถุสัมพันธ์ แล้วทดสอบการคิวรีข้อมูลและการเรียกใช้เมธอด (method)
- 1.5.6 ออกแบบฐานข้อมูลรถโดยใช้แนวความคิดเชิงวัตถุซึ่งสนับสนุนในฐานข้อมูลเชิงวัตถุสัมพันธ์ จากนั้นทำการเรียกค้นข้อมูลเพื่อแสดงให้เห็นถึงข้อดีของแนวคิดเชิงวัตถุที่มีในฐานข้อมูลเชิงวัตถุสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ส่วนประกอบของรายงาน

รายงานฉบับนี้แบ่งเป็น 5 บท ได้แก่

เนื้อหาในบทที่ 1 กล่าวถึง ความเป็นมาของปัญหา วัตถุประสงค์ของโครงการ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงการ วิธีการดำเนินงาน และส่วนประกอบของรายงานฉบับนี้

เนื้อหาในบทที่ 2 กล่าวถึงฐานข้อมูลเชิงวัตถุสัมพันธ์และภาษาเอสคิวแอลสำหรับฐานข้อมูลเชิงวัตถุสัมพันธ์ และกล่าวถึงมาตรฐานเอสคิวแอล 3 ซึ่งเป็นมาตรฐานเอสคิวแอลที่ใช้ในฐานข้อมูลเชิงวัตถุสัมพันธ์

เนื้อหาในบทที่ 3 กล่าวถึงภาษาเอสคิวแอลเชิงวัตถุสัมพันธ์ใน Oracle 10g ประกอบด้วยการใช้งานภาษาเอสคิวแอลในแบบต่างๆ ตามทฤษฎีแนวคิดเชิงวัตถุที่ได้ศึกษา

เนื้อหาในบทที่ 4 กล่าวถึงการทดสอบและผลการทดสอบการใช้งานฐานข้อมูลเชิงวัตถุสัมพันธ์จากฐานข้อมูลประธานาธิบดีของประเทศสหรัฐอเมริกาและฐานข้อมูลรถที่ออกแบบโดยใช้หลักการของฐานข้อมูลเชิงวัตถุสัมพันธ์

เนื้อหาในบทที่ 5 เป็นบทสรุปและวิจารณ์ ซึ่งกล่าวถึงบทสรุปของโครงการ วิจารณ์สิ่งที่ได้รับจากโครงการ ปัญหาอุปสรรคและแนวทางแก้ไข และข้อเสนอแนะสำหรับเป็นแนวทางในการพัฒนาต่อไป

บทที่ 2

ฐานข้อมูลเชิงวัตถุสัมพันธ์และภาษาเอสคิวแอล สำหรับฐานข้อมูลเชิงวัตถุสัมพันธ์

2.1 ฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database)

ในปัจจุบัน เทคโนโลยีสารสนเทศได้เข้ามามีบทบาทสำคัญในการดำเนินกิจการ และระบบต่างๆ ซึ่งการที่จะทำให้ระบบสารสนเทศนั้นดำเนินการอย่างมีประสิทธิภาพได้ จำเป็นที่จะต้องมีการจัดการข้อมูลอย่างมีประสิทธิภาพ

ซึ่งในอดีตที่ผ่านมา ได้มีการค้นคว้า และพัฒนาระบบฐานข้อมูลขึ้นมามากมาย หลากหลายรูปแบบและหลายแนวความคิด แต่ระบบฐานข้อมูลที่ได้รับความนิยมและแพร่หลายมากที่สุดในการใช้งานคือ ระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database System) แต่ในปัจจุบัน การพัฒนาโปรแกรมส่วนใหญ่ที่มีการใช้หลักการเชิงวัตถุมากขึ้น จนทำให้หลักการดังกล่าวเข้ามามีบทบาทสำคัญในการพัฒนาระบบฐานข้อมูลเชิงวัตถุ (ออปเจกต์ Oriented Database System) อย่างไรก็ตาม เนื่องจากว่าระบบงานเก่าขององค์กรต่างๆ ได้พัฒนาโดยใช้ระบบฐานข้อมูลเชิงสัมพันธ์เป็นหลัก ทำให้การพัฒนาบบงานด้วยหลักการเชิงวัตถุ นั้นประสบปัญหา

ต่อมาได้มีแนวคิดที่จะประยุกต์นำเอาข้อดีของหลักการแนวคิดเชิงวัตถุ เข้ามาผสมผสานกับระบบฐานข้อมูลเชิงสัมพันธ์ ทำให้เกิดแนวคิดของระบบฐานข้อมูลที่เรียกว่า ระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database)

ในภายหลังสถาบันมาตรฐานแห่งชาติของสหรัฐอเมริกาหรือเอเอ็นเอสไอ(ANSI)ได้มีการเพิ่มมาตรฐานด้านออปเจกต์-โอเรียนเต็ด(Object-Oriented) ให้กับมาตรฐานเอสคิวแอลที่มีอยู่ โดยทำการเพิ่มคุณสมบัติของออปเจกต์ (Object Extension) ให้กับระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ โดยตรง ซึ่งเป็นการพัฒนามาตรฐานจากเอสคิวแอล 2 (SQL2) ไปสู่เอสคิวแอล 3 (SQL3)

2.1.1 โครงสร้างข้อมูล

ดังที่กล่าวมาแล้วว่าการพัฒนามาตรฐานเอสคิวแอล 3 (SQL3) เพื่อรองรับการใช้งานข้อมูลประเภทออปเจกต์ ของระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ ใช้วิธีการเพิ่มคุณสมบัติของออปเจกต์ (Object Extension) ให้แก่ระบบฐานข้อมูลเชิงสัมพันธ์ ซึ่งการเพิ่มคุณสมบัติของออปเจกต์ (Object

Extension) นี้เป็นการเพิ่มชนิดข้อมูลแบบใหม่ให้กับระบบฐานข้อมูลเชิงสัมพันธ์ โดยที่โครงสร้างการจัดเก็บข้อมูลยังถูกแสดงอยู่ในรูปแบบของตารางเช่นเดิม ชนิดข้อมูลแบบใหม่นี้เรียกว่า ชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type (UDT)) ซึ่งจะมีลักษณะคล้ายกับคลาสในหลักการโปรแกรมเชิงวัตถุ กล่าวคือ ชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) จะถูกใช้เพื่อรวมกลุ่มของข้อมูลที่มีความสัมพันธ์กันทั้งในส่วนของคุณสมบัติของข้อมูลและสถานการณ์ที่จะเกิดขึ้นกับข้อมูลเหล่านั้น โดยสนับสนุนคุณสมบัติการห่อหุ้ม (Encapsulation), การมีความสัมพันธ์ ชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) อื่นๆ ในรูปแบบของการสืบทอด (Inheritance) ซึ่งเป็นการถ่ายทอดคุณสมบัติของชนิดข้อมูล (Data Type) จากชนิดข้อมูลที่เป็นแม่ของคณ นอกจากนี้แล้วยังสามารถกำหนดโอเปอเรชัน (Operation) และฟังก์ชัน (Function) ที่มีความสัมพันธ์กับข้อมูลได้อีกด้วย

สำหรับชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) มีทั้งชนิดข้อมูลของแถวในตาราง (Row Type), การสร้างชนิดข้อมูลจากชนิดข้อมูลที่ระบบมีอยู่ (Distinct Type) และการสร้างชนิดข้อมูลในรูปแบบของคอลเลกชัน (Collection Type) เช่น ลิสต์ (List), เซต (Set), อาร์เรย์ (Array) อีกด้วย ซึ่งจะเห็นได้ว่าชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) จะสามารถรองรับการใช้งานข้อมูลที่มีโครงสร้างซับซ้อนมากให้ดีกว่าเดิมได้

2.1.2 ภาษาที่ใช้ในการจัดการฐานข้อมูล

เพื่อสนับสนุนการใช้งานชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) ในระบบฐานข้อมูล เชิงวัตถุสัมพันธ์ จึงได้มีการเพิ่มเติมความสามารถในการเข้าถึงข้อมูลดังกล่าวให้แก่ภาษาเอสคิวแอล ซึ่งอาจเรียกส่วนที่เพิ่มเติมใหม่นี้ว่าออปเจกต์เอสคิวแอล (Object SQL) ประเด็นสำคัญของส่วนที่เพิ่มเติมใหม่นี้คือ การสนับสนุนการใช้งานฐานข้อมูลในรูปแบบออปเจกต์ที่รวมการจัดการข้อมูลเนสต์คอปเจกต์ (Nested Object), เซต-แวลิว แอตทริบิวต์ (Set-Value Attribute), ความสามารถในการใช้โอเปอเรชันหรือฟังก์ชันของออปเจกต์ซึ่งในการทำคิวรี (Query) กับชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) ยังสามารถทำได้โดยยึดเอาภาษาเอสคิวแอลเป็นภาษาหลักในการทำคำจำกัดความของข้อมูล (Data Definition), การจัดการข้อมูล (Data Manipulation) และการทำคิวรี (Query) สำหรับผลของการทำคิวรีนั้น ข้อมูลยังจะถูกแสดงในรูปแบบของตารางอยู่

นอกจากนี้มาตรฐานเอสคิวแอล 3 (SQL3) ยังมีส่วนของการเพิ่มโพรซีเจอร์ (Procedure Extension) ซึ่งการดำเนินการดังกล่าวได้ครอบคลุมไปถึงการพัฒนาารูปแบบคำสั่งเอสคิวแอลที่ใช้ในการทำ สตอร์โพรซีเจอร์ (Store Procedure) และฟังก์ชันที่ผู้ใช้กำหนดขึ้นเอง (User Defined Function) เพื่อใช้งานร่วมกับฐานข้อมูลด้วย กล่าวคือโพรซีเจอร์และฟังก์ชันเหล่านี้จะสามารถสร้างขึ้นได้จากภาษาเอสคิวแอล เอง ซึ่งมีการพัฒนารูปแบบคำสั่งใหม่ที่ใกล้เคียงกับภาษาสำหรับการเขียนโปรแกรมทั่วไปในยุคปัจจุบัน ได้แก่ ความสามารถในการใช้คำสั่งเงื่อนไขประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If/Then/Else การใช้ลูป (Loop) แบบ While และ Do/Until หรือการใช้ Case เป็นต้น ทั้งยังให้โปรแกรมเมอร์สามารถสร้างฟังก์ชันไลบรารี (Function Library) ด้วยภาษาอื่นๆ เช่น ภาษาซี, ภาษาฟอร์เทน, ภาษาปาสคาล, ภาษาจาวา เป็นต้น เพื่อใช้งานร่วมกับภาษา เอสคิวแอลที่มีอยู่ได้ ซึ่งจะเพิ่มขีดความสามารถในการทำควิรี่ได้ดียิ่งขึ้น

2.1.3 กลไกที่ใช้ในการเข้าถึงข้อมูล

เนื่องจากภาษาหลักที่ใช้ในการจัดการฐานข้อมูลยังคงเป็นภาษาเอสคิวแอล และระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ได้ถูกพัฒนาขึ้นมาจากระบบฐานข้อมูลเชิงสัมพันธ์ ดังนั้นกลไกในการเข้าถึงข้อมูลส่วนใหญ่จะยังคงเป็นเช่นเดียวกับระบบฐานข้อมูลเชิงสัมพันธ์อยู่ ส่วนการเพิ่มความความสามารถในการเข้าถึงชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) หรือการเข้าถึงข้อมูลในรูปแบบออปเจกต์นั้นยังคงมีปัญหาเมื่อต้องการเข้าถึงโดยตรงจากภาษาการโปรแกรมเชิงวัตถุ ทำให้ยังจำเป็นที่จะต้องมีการแปลงรูปให้กลับมาอยู่ในลักษณะตารางเสียก่อน

2.1.4 ข้อดีของฐานข้อมูลเชิงวัตถุสัมพันธ์

- เป็นการเพิ่มคุณสมบัติของออปเจกต์เข้าไปในฐานข้อมูลเชิงสัมพันธ์ เช่น ชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) ทำให้ผู้ใช้สามารถสร้างชนิดข้อมูลที่ต้องการขึ้นมาใช้เองได้
- สามารถมีข้อมูลที่เป็นรีพีตติ้งกรุป (Repeating Group) อยู่ในคอลัมน์ (Column) ได้ ซึ่งฐานข้อมูลเชิงสัมพันธ์จะไม่ยอมให้มีข้อมูลในลักษณะนี้ ซึ่งการที่ยอมให้ คอลัมน์ (Column) สามารถมีชนิดข้อมูลเป็นคอลเลกชัน (Collection Type) ได้ จะช่วยให้จัดการข้อมูลบางลักษณะได้ง่ายขึ้น โดยไม่ต้องทำการแยกตารางใหม่
- ช่นความยุ่งยากในการเขียน โปรแกรมแอปพลิเคชัน ซึ่งผู้ใช้สามารถเรียกใช้เมธอดที่ผู้ออกแบบระบบฐานข้อมูลได้จัดเตรียมไว้ให้ โดยที่ไม่ต้องทราบว่าจะข้างในนั้นมีการทำงานเป็นอย่างไร
- ไม่ต้องทำการรวม (Join) ตารางหลายๆ ตารางในการควิรี่ข้อมูล เพราะในการออกแบบระบบฐานข้อมูล สามารถทำได้โดยสร้างคอลัมน์ (Column) ที่มีชนิดข้อมูลอ้างอิงไปยังตารางอื่นได้
- สามารถใช้คุณสมบัติการนำกลับมาใช้ใหม่ได้ เช่น สามารถเพิ่มคุณสมบัติชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง (User Defined Type) ที่มีอยู่เดิม โดยใช้คุณสมบัติของการสืบทอด ทำให้ไม่ต้องออกแบบส่วนที่ทำหน้าที่เหมือนกันหลายๆ รอบ และช่วยลดเวลาการทดสอบการทำงานอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อต้องการเปลี่ยนแปลงโครงสร้างข้อมูล ก็สามารถแก้ไขที่ชนิดข้อมูลที่ผู้ใช้กำหนดขึ้นเอง(User Defined Type) ที่สร้างเองได้เลย โดยไม่ต้องไปทำการเปลี่ยนแปลงที่ตาราง

ตาราง 2.1 ตารางเปรียบเทียบฐานข้อมูลแบบต่างๆ กับฐานข้อมูลเชิงวัตถุสัมพันธ์

ฐานข้อมูลเชิงวัตถุสัมพันธ์	ฐานข้อมูลเชิงวัตถุ	ฐานข้อมูลเชิงสัมพันธ์
1. ใช้ภาษา SQL3	1. ใช้ภาษา OQL หรือภาษาทางด้าน ออปเจกต์	1. ใช้ภาษา SQL2
2. เป็นการเพิ่มคุณสมบัติทางออปเจกต์ ให้กับภาษา SQL2	2. เป็นการนำความสามารถของ DBMS ไปผูกกับภาษาทางด้าน ออปเจกต์	2. ไม่รองรับความสามารถทาง ออปเจกต์
3. แสดงผลและเก็บข้อมูลในรูปแบบของตาราง	3. เก็บข้อมูลในรูปของ Abstract Data Type	3. แสดงผลและเก็บข้อมูลในรูปแบบของตาราง
4. สามารถสร้างโครงสร้างข้อมูลประเภทใหม่ได้	4. สามารถสร้างโครงสร้างข้อมูลประเภทใหม่ได้	4. ไม่สามารถสร้างโครงสร้างข้อมูลประเภทใหม่ได้
5. เข้าถึงข้อมูลได้โดยตรงหรือเข้าถึงข้อมูลผ่านทางเมธอด	5. เข้าถึงข้อมูลผ่านทางเมธอดเท่านั้น	5. เข้าถึงข้อมูลได้โดยตรง
6. รองรับ Complex Datatype	6. รองรับ Complex Datatype	6. ไม่รองรับ Complex Datatype

ตาราง 2.2 ตารางเปรียบเทียบแนวคิดฐานข้อมูลแบบต่างๆ กับฐานข้อมูลเชิงวัตถุสัมพันธ์

ฐานข้อมูลเชิงวัตถุสัมพันธ์	ฐานข้อมูลเชิงวัตถุ	ฐานข้อมูลเชิงสัมพันธ์
User Defined Type	Class	Entity Set
Instance	Instance	Entity
Attribute	Instance Variable	Attribute
OID, Primary	OID	Primary
Method	Method	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับก... เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 แนวคิดเชิงวัตถุและรูปแบบของเอสคิวแอล 3 (SQL3 Features)

2.2.1 องค์ประกอบของออบเจกต์

2.2.1.1 ออบเจกต์ไอดี (OID - Object Identity)

ออบเจกต์นั้นมีเอกลักษณ์ที่เรียกว่า OID หรือ Object Identity ซึ่งเป็นสิ่งที่ระบุ หรือใช้อ้างอิงถึงตัวออบเจกต์เอง ซึ่งค่า OID จะมีค่าไม่ซ้ำกันในแต่ละออบเจกต์ ออบเจกต์นั้นจะยังคงรักษาความเป็น เอกลักษณ์แม้ว่าข้อมูลของตัวแปรหรือเมธอดของออบเจกต์มีการเปลี่ยนแปลง ซึ่ง OID นี้ จะไม่สามารถมองเห็นได้โดยผู้ใช้ทั่วไป

2.2.1.2 คุณสมบัติ (Property)

คุณสมบัติ (Property) คือข้อมูลของออบเจกต์ เป็นไปได้ทั้งข้อมูลชนิดพื้นฐานทั่วไป เช่น ตัวเลข, สตริง (String) เป็นต้น หรืออาจเป็นชนิดข้อมูลที่ซับซ้อนคือการนำเอาข้อมูลชนิดพื้นฐานมารวมเป็นโครงสร้าง เช่น อาร์เรย์ (Array) หรืออาจจะเป็นข้อมูลที่เป็นออบเจกต์ก็ได้

2.2.1.3 เมธอด (Method)

ออบเจกต์จะถูกเรียกใช้งานได้โดยผ่านทางเมธอดเท่านั้น ซึ่งเป็นไปตามคุณสมบัติการห่อหุ้ม (Encapsulation) หรือการปกปิดข้อมูล ซึ่งจะเรียกใช้เมธอดเมื่อต้องการเรียกดูหรือเปลี่ยนแปลงค่าคุณสมบัติ (Property) ของออบเจกต์

2.2.2 แนวคิดเชิงวัตถุ

2.2.2.1 คุณสมบัติการห่อหุ้ม (Encapsulation)

หลักการหนึ่งของการ โปรแกรมเชิงวัตถุก็คือ ความสามารถในการซ่อนข้อมูล เป็นการซ่อนรายละเอียด คุณสมบัติ และเมธอดของคลาสไว้ ข้อมูลภายในออบเจกต์จะถูกปิดกั้นไว้ไม่ให้มองเห็นและแก้ไขเปลี่ยนแปลงจากภายนอกโดยตรง ถ้าต้องการแก้ไขหรือเปลี่ยนแปลงข้อมูลจะทำได้โดยผ่านทาง เมธอดที่คลาสนั้นมีให้เท่านั้น

การกำหนดการเข้าถึง (Access Modifier) มีผลทำให้สามารถควบคุมระดับการเข้าถึงข้อมูลได้ ผู้ออกแบบระบบสามารถกำหนดระดับการเข้าถึงข้อมูลได้หลายระดับ ซึ่งการกำหนดการเข้าถึงที่นิยมใช้มีดังนี้

2.2.2.1.1 ไพรเวท (Private)

ข้อมูลภายในของคลาสจะไม่ถูกเปิดเผยแก่ภายนอก จะเรียกใช้และประมวลผลได้เมื่ออยู่ภายในคลาสนั้นๆ เท่านั้น แม้แต่ซับคลาส (Subclass) ก็ไม่สามารถเรียกใช้งานได้โดยตรง การเรียกใช้จากภายนอกจะทำได้โดยผ่านทางเมธอดเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.1.2 โพรเทค (Protected)

คลาสภายนอกและคลาสอื่นที่ไม่มีความสัมพันธ์เป็นซับคลาส (Subclass) กับคลาสนั้น จะไม่สามารถเข้าถึงตัวแปรสมาชิกโพรเทค (Protected) โดยตรงได้ จะต้องเรียกใช้ผ่านทางเมธอด

2.1.2.1.3 พับลิค (Public)

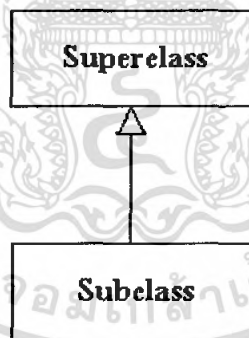
ข้อมูลภายในคลาสจะถูกเข้าถึงได้โดยตรงและอิสระจากภายนอก ไม่ว่าจะเป็นการเรียกดูหรือเปลี่ยนแปลงค่า นั่นก็คือไม่มีการปกปิดข้อมูลใดๆ เลย

2.2.2.2 คุณสมบัติการสืบทอด (Inheritance)

หลักการสืบทอด เป็นการนำคลาสที่เคยสร้างไว้แล้ว นำกลับมาใช้ใหม่ โดยใช้หลักการการมีคุณสมบัติบางอย่างร่วมกัน โดยออปเจกต์จะสืบทอดคุณสมบัติ (Property) และเมธอดมาจากคลาสที่มีอยู่แล้ว (Superclass) โดยสามารถเพิ่มคุณสมบัติหรือเมธอดบางอย่างเพิ่มขึ้นไปเพื่อให้เกิดคลาสใหม่ที่พิเศษกว่าคลาสเดิม (Subclass) ซึ่งสามารถที่จะแบ่งลักษณะของการสืบทอดได้ 2 ประเภทดังนี้

2.2.2.2.1 การสืบทอดเพียงคลาสเดียว (Single Inheritance)

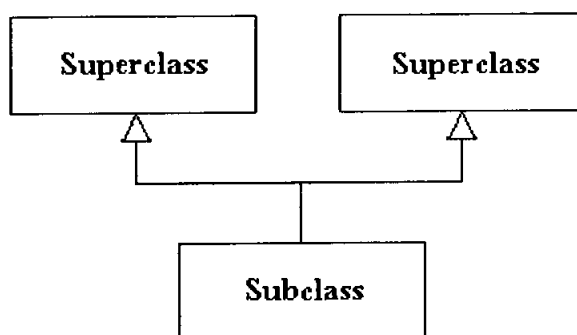
เป็นการสืบทอดมาจากคลาสแม่ (Superclass) เพียงแค่คลาสเดียว ดังรูป



รูปที่ 2.1 การสืบทอดเพียงคลาสเดียว (Single Inheritance)

2.2.2.2.2 การสืบทอดมากกว่าหนึ่งคลาส (Multiple Inheritance)

เป็นการสืบทอดมาจากคลาสแม่ (Superclass) มากกว่าหนึ่งคลาส ดังรูป



รูปที่ 2.2 การสืบทอดมากกว่าหนึ่งคลาส (Multiple Inheritance)

2.2.2.3 คุณสมบัติโพลิมอร์ฟิซึม (Polymorphism)

คุณสมบัติโพลิมอร์ฟิซึม (Polymorphism) คือการที่คลาสแตกต่างกันสามารถกระทำโอเปอเรชัน (Operation) เดียวกัน แต่ให้ผลลัพธ์ที่แตกต่างกันได้ โดยทำการส่งข้อความ (Message) เดียวไปยังออบเจกต์หลายตัว และออบเจกต์แต่ละตัวทำการตอบสนองข้อความ (Message) ต่างกัน แบ่งออกได้ 2 ลักษณะใหญ่ดังนี้

2.2.2.3.1 การโอเวอร์โหลด (Overloading)

คือการที่เมธอดภายใต้คลาสเดียวกัน มีชื่อเหมือนกัน แต่มีรูปแบบการใช้งานที่แตกต่างกัน โดยต้องทำการประกาศเมธอดที่ต่างกันด้วย ซึ่งการประกาศเมธอดจะเป็นการประกอบกันของ ชื่อเมธอด ชนิดของตัวแปร และจำนวนของตัวแปรที่ส่งผ่านไปยังเมธอดนั้น

2.2.2.3.2 การโอเวอร์ไรต์ (Overriding)

คือการที่ซับคลาส (Subclass) ทำการอิมพลีเมนต์เมธอดที่ได้รับการสืบทอดมาจากคลาสแม่ (Superclass) ใหม่ ซึ่งเป็นการแก้ไขการทำงานภายในเมธอดให้มีความเจาะจงมากขึ้น เมธอดนี้จะถูกเรียกใช้แทนเมธอดที่ได้รับสืบทอดมาจากคลาสแม่ (Superclass)

2.2.3 รูปแบบของเอสคิวแอล 3

2.2.3.1 มาตรฐานเอสคิวแอล 3

ก่อนจะกล่าวถึงมาตรฐานเอสคิวแอล 3 จะได้กล่าวถึงภาษาเอสคิวแอลซึ่งเป็นภาษาที่ใช้เรียกค้นข้อมูลจากฐานข้อมูล ซึ่งคำว่า SQL (ย่อมาจาก Structured Query Language) เป็นภาษาที่ใช้สำหรับการจัดการ และการดึงข้อมูลจากฐานข้อมูล ในการดึงข้อมูลจะใช้งานคำสั่งในภาษาเอสคิวแอลไปยังระบบการจัดการฐานข้อมูล หรือ DBMS(Database Management System) ซึ่งจะคอยทำหน้าที่จัดการ และอำนวยความสะดวกในการดูแลข้อมูล รวมถึงดึงข้อมูลจากฐานข้อมูลแล้วส่งกลับมา ผลลัพธ์ที่ได้เรียกว่า การคิวรีฐานข้อมูล (Query Database)

ภาษาเอสคิวแอลแบ่งได้เป็น 3 กลุ่มคือ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Data Manipulation Language (DML) เป็นกลุ่มคำสั่งในภาษาเอสคิวแอลที่ใช้สำหรับการเข้าถึงข้อมูลและการแก้ไขข้อมูล เช่น SELECT ใช้เพื่อคิวรีหาข้อมูล, INSERT ใช้เพื่อเพิ่มข้อมูล, DELETE ใช้เพื่อลบ และ UPDATE ใช้เพื่ออัปเดตลงในตารางฐานข้อมูล

2. Data Definition Language (DDL) เป็นกลุ่มคำสั่งในภาษาเอสคิวแอลที่ใช้กำหนดคอปเจกต์ฐานข้อมูลเช่น CREATE ใช้เพื่อเพิ่มตารางลงในฐานข้อมูล, DROP ใช้เพื่อลบตารางออก และ ALTER ใช้เพื่อเปลี่ยนแปลงโครงสร้างตารางในฐานข้อมูล

3. Data Control Language (DCL) เป็นกลุ่มคำสั่งในภาษาเอสคิวแอลที่ควบคุมความปลอดภัยกับข้อมูล เช่น GRANT ใช้เพื่อให้สิทธิในการใช้ฐานข้อมูล และ REVOKE ใช้เพื่อยกเลิกสิทธิในการใช้ฐานข้อมูล

สำหรับมาตรฐานเอสคิวแอล 3(SQL-3) หรือเอสคิวแอล 99 (SQL-99) เป็นมาตรฐานที่ใช้บนฐานข้อมูลเชิงวัตถุสัมพันธ์ มาตรฐานเอสคิวแอล 3 นี้พัฒนามาจากมาตรฐานเอสคิวแอล 92 โดยได้รวมโมเดลเชิงวัตถุเข้าไว้เป็นพื้นฐานหลักในมาตรฐาน สิ่งที่มีการเพิ่มเข้าไปในมาตรฐานเอสคิวแอล 3 ได้แก่

2.2.3.2 ชนิดข้อมูลพื้นฐานที่เพิ่มเข้าไป (Extended Base Types)

ชนิดข้อมูลพื้นฐานที่มีในเอสคิวแอล 92 ได้แก่

- Integer
- Floating-point number
- Character string, fixed or variable length
- Date, time, datetime, interval
- Numeric and decimal

ในเอสคิวแอล 3 นอกจากจากชนิดข้อมูลพื้นฐานดังกล่าวข้างต้นแล้ว ยังได้มีการเพิ่มชนิดข้อมูลพื้นฐานเข้ามาได้แก่ วิดีโอ(Video), รูปภาพ(Image), ออดิโอ(Audio), ข้อความ (Text), เว็บ-เพจ (Web-pages), กราฟฟิค (Geographic) เป็นต้น เหตุที่มีการเพิ่มชนิดข้อมูลพื้นฐานเข้ามานั้น เพราะการทำงานในเอสคิวแอล 92 ออกแบบให้ใช้กับชนิดข้อมูลพื้นฐาน (primitive datatype) เท่านั้น ทำให้ไม่สามารถรองรับกับแอปพลิเคชันยุคใหม่ที่มีข้อมูลซับซ้อนและยุ่งยากได้อย่างมีประสิทธิภาพ

2.2.3.3 ราวไทป์ (Row Types)

ในเอสคิวแอล 92 ค่าของแอตทริบิวต์ในคอลัมน์จะต้องไม่สามารถแบ่งแยกต่อไปได้อีก หรือเรียกว่า Atomic แต่ในเอสคิวแอล 3 นี้ค่าของแอตทริบิวต์ในคอลัมน์ไม่จำเป็นต้อง Atomic ดัง ตัวอย่าง

BRANCHNO	ADDRESS		
	STREET	AREA	CITY
10	Main ST.	Barking	London
20	High ST.	Leyton	London
30	High ST.	Marsden	Oxford

รูปที่ 2.3 ตาราง Branch ที่มี Address เป็นคอลัมน์ ราวไทป์ (Row Type)

```
CREATE TABLE Branch (
  BranchNo    VARCHAR2(3),
  Address     ROW( Street  VARCHAR2(25),
                  Area     VARCHAR2(15),
                  City     VARCHAR2(15) )
)
```

รูปที่ 2.4 คำสั่งในการสร้าง ราวไทป์ (Row Type)

จากรูปที่ 2.3 คอลัมน์ Address มีชนิดข้อมูลเป็น Row Type ซึ่งใน Address จะประกอบไปด้วยค่าของแอตทริบิวต์อีก 3 ค่า ได้แก่ Street, Area และ City โดยที่ในการสร้าง ราวไทป์นั้นใช้คีย์เวิร์ดคำว่า ROW

2.2.3.4 ชนิดข้อมูลที่ผู้ใช้สร้างขึ้นเอง (User-Defined Types)

ในเอสคิวแอล 3 นอกจากชนิดข้อมูลพื้นฐานแล้ว ยังยอมให้ผู้ใช้สร้างชนิดข้อมูลขึ้นมาใช้ได้เอง เรียกว่า User-Defined Type ซึ่งในการสร้าง User-Defined Type แสดงได้ดังตัวอย่าง

```

CREATE TYPE person_type AS (
    PRIVATE
        date_of_birth DATE
        CHECK(date_of_birth > DATE '19-01-1900')
    PUBLIC
        name          VARCHAR2(15) NOT NULL,
        address        VARCHAR2(50) NOT NULL,
        tel no         VARCHAR2(13) NOT NULL,
        FUNCTION get_age (P person_type)
            RETURNS INTERGER
            /* code to calculate age from date_of_birth */
            RETURN
        END;
)NOT FINAL;

```

รูปที่ 2.5 คำสั่งในการสร้าง User-Defined Type และ User-Defined Function

ในการสร้าง User-Defined Type ใช้คีย์เวิร์ดคำว่า CREATE TYPE ตามด้วยชื่อของ User-Defined Type นั้น และภายในบอดีของ User-Defined Type จะมีการประกาศคล้ายๆ กับการสร้างคลาสในภาษาเชิงโปรแกรมมิ่ง คือ มีส่วน PRIVATE และ PUBLIC เพื่อกำหนดสิทธิในการเข้าถึงข้อมูล

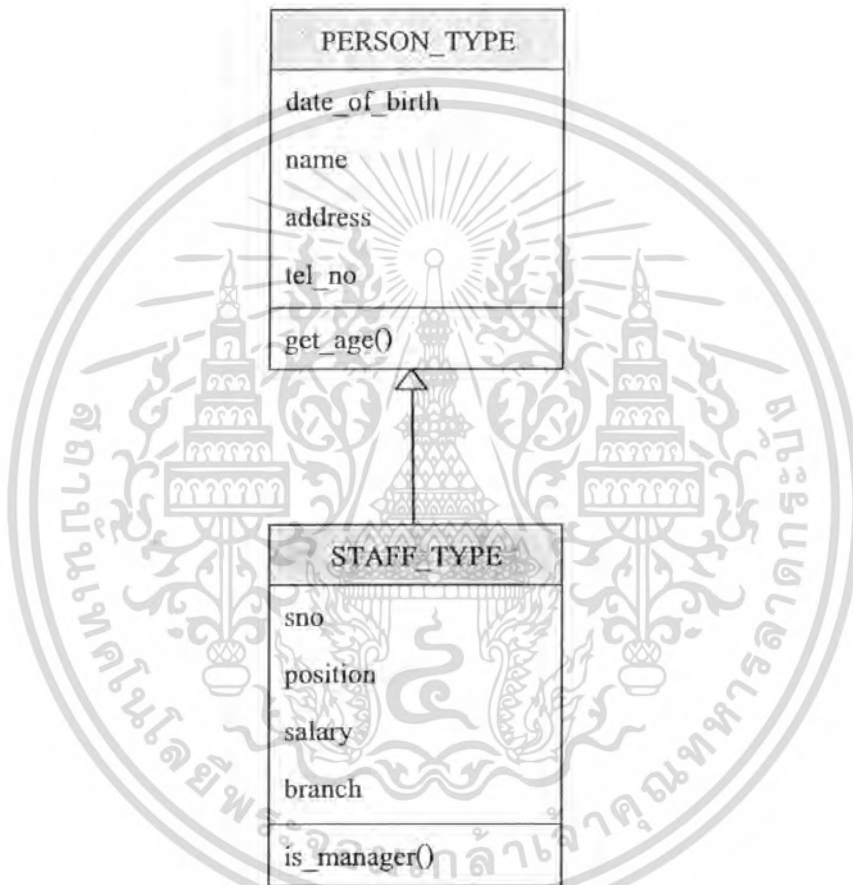
2.2.3.5 ฟังก์ชันที่ผู้ใช้สร้างขึ้นเอง (User-Defined Functions)

นอกจาก User-Defined Type แล้ว ยังสามารถสร้างฟังก์ชันการทำงานต่างๆ ให้กับชนิดข้อมูลที่สร้างขึ้นได้ โดยการสร้างฟังก์ชันให้กับชนิดข้อมูลนั้นๆ ให้ประกาศฟังก์ชันไว้ภายในบอดีของชนิดข้อมูลที่สร้างขึ้นมา ดังแสดงในรูป 2.5 ฟังก์ชัน get_age ทำการรีเทิร์นค่าอายุของ person_type ที่รับเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.6 ซับไทป์และซูเปอร์ไทป์ (Sub-Types and Super-Types)

เอสคิวแอล 3 สนับสนุนการสืบทอด (Inheritance) ซึ่งยอมให้ซับไทป์สืบทอดจากซูเปอร์ไทป์เพียงไทป์เดียว (Single Inheritance) หรือสืบทอดจากหลายไทป์ก็ได้ (Multiple Inheritance)



รูปที่ 2.6 Sub-Type และ Super-Type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE TYPE staff_type UNDER person_type AS (
    sno          VARCHAR2(5) NOT NULL UNIQUE,
    position     VARCHAR2(10) NOT NULL,
    salary       NUMBER (7,2),

    CREATE FUNCTION is_manager (s STAFF_TYPE)
        RETURNS BOOLEAN
        BEGIN
            IF s.position = 'Manager' THEN
                RETURN TRUE;
            ELSE
                RETURN FALSE;
            END IF;
        END;
) NOT FINAL;

```

รูปที่ 2.7 คำสั่งในการสร้าง Sub-Type และ Super-Type

จากรูป 2.6 แสดงการสืบทอด ชั้นไทยคือ STAFF TYPE และซูเปอร์ไทยคือ PERSON_TYPE โดยที่ชั้นไทยจะมีแอตทริบิวต์และฟังก์ชันของทั้งตัวเองและของซูเปอร์ไทยด้วย ส่วนในการสร้างชั้นไทยนั้นให้ใช้คีย์เวิร์ด UNDER ตามด้วยชื่อซูเปอร์ไทย

2.2.3.7 ชั้นเทเบิลและซูเปอร์เทเบิล (Sub-Tables and Super-Tables)

นอกจากชั้นไทยและซูเปอร์ไทยแล้ว เอสคิวแอล 3 ยังสนับสนุนการทำชั้นเทเบิลและซูเปอร์เทเบิลอีกด้วย ซึ่งสามารถแก้ปัญหาที่ว่าหากสร้างตารางตั้งแต่ 2 ตารางขึ้นไปที่มีชนิดข้อมูลเดียวกัน ถ้าต้องการหาอินสแตนซ์ของชนิดข้อมูลนั้นทั้งหมดจะหาได้อย่างไร ซึ่งทางแก้ปัญหานี้ก็คือทำเป็นชั้นเทเบิลและซูเปอร์เทเบิล จะทำให้สามารถหาอินสแตนซ์ของชนิดข้อมูลนั้นๆ ได้โดยง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.8 ชนิดข้อมูลอ้างอิงและออปเจกต์ไอดี (Reference Types and Object Identity)

ชนิดข้อมูลอ้างอิง (Reference Type) ในเอสคิวแอล 3 เหมือนกันกับชนิดข้อมูลอ้างอิงในฐานข้อมูลเชิงวัตถุ (Object Oriented Database) ชนิดข้อมูลอ้างอิงนี้จะยอมให้โรว์ใดๆ ในตารางถูกใช้ร่วมกันจากหลายๆ ตารางได้ โดยที่ค่าของชนิดข้อมูลอ้างอิงก็คือ ออปเจกต์ไอดี (OID)

```
CREATE TYPE staff_type UNDER person_type AS(
    sno          VARCHAR2(5) NOT NULL UNIQUE,
    position     VARCHAR2(10) NOT NULL,
    salary       NUMBER (7,2),
    next_of_kin  REF(person_type)
    bno          VARCHAR2(3) NOT NULL
) NOT FINAL;

CREATE TABLE person OF person_type(
    oid          REF(person_type) VALUES ARE SYSTEM GENERATED);
```

รูปที่ 2.8 คำสั่งในการสร้าง Reference Type

2.2.3.9 ชนิดข้อมูลที่เป็นคอลเลกชัน (Collection Types)

ชนิดข้อมูลที่เป็นคอลเลกชัน (Collection Types) ในเอสคิวแอล 3 ประกอบด้วย

- List ใช้เก็บข้อมูลที่สนใจลำดับ โดยค่าในลิสต์ซ้ำกันได้
- Set ใช้เก็บข้อมูลที่ไม่สนใจลำดับ โดยค่าในเซตจะต้องไม่ซ้ำกัน
- Multiset ใช้เก็บข้อมูลที่ไม่สนใจลำดับ โดยค่าในมัลติเซตซ้ำกันได้
- Array ใช้เก็บข้อมูลที่เป็นชนิดข้อมูลเดียวกัน ใช้หมายเลขลำดับในการเข้าถึงข้อมูล

2.2.3.10 การทำคิวรีในเอสคิวแอล 3

การทำคิวรีในฐานข้อมูลเชิงวัตถุสัมพันธ์จะเหมือนกับการทำคิวรีในระบบฐานข้อมูลเชิงสัมพันธ์ โดยที่สามารถอ้างถึงแอตทริบิวต์ย่อยในรายการโรว์ไทป์ (Row Type) ผ่าน Path expressions โดยใช้ .(dot-notation) เช่น

ตัวอย่าง หารายชื่อผู้ที่มีอยู่ในเขตรหัสไปรษณีย์ 10800

```
SELECT      P.Name
FROM        Person P
WHERE       P.Address.Zip = '10800';
```

รูปที่ 2.9 ตัวอย่างใช้คำสั่ง SELECT ในฐานข้อมูลเชิงวัตถุสัมพันธ์

การเพิ่มข้อมูลลงในตาราง ทำได้โดยคำสั่ง INSERT และสำหรับข้อมูลที่เป็นโรว์ไทป์ (Row type) เพิ่มคีย์เวิร์ด ROW เพื่อสร้างโครงสร้างให้กับข้อมูลชุดนั้น เช่นเพิ่มคนที่ชื่อนิดน้อย ที่อยู่ 1518 ถนนพิบูลสงคราม บางซื่อ 10800 มีเงินเดือน 20000 บาท

```
INSERT INTO Person
VALUES ('NidNoi',
Row('1518 Pibulsongkram Rd', 'Bangsue', '10800'),
20000 )
```

รูปที่ 2.10 ตัวอย่างใช้คำสั่ง INSERT ในฐานข้อมูลเชิงวัตถุสัมพันธ์

การปรับปรุงที่อยู่ในตาราง ทำได้โดยคำสั่ง UPDATE เช่น

```
UPDATE Person
SET Address = Row('111 Ruthada Rd', 'Jatujak', '10900')
WHERE Name = 'Nidnoi' AND
Address = Row('1518 Pibulsongkram Rd', 'Bangsue', '10800')
```

รูปที่ 2.11 ตัวอย่างใช้คำสั่ง UPDATE ในฐานข้อมูลเชิงวัตถุสัมพันธ์

83020

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ภาษาเอสคิวแอลสำหรับฐานข้อมูลเชิงวัตถุสัมพันธ์

บนออราเคิล10g

(Object Relational SQL on Oracle 10g)

3.1 ศัพท์เฉพาะและคุณสมบัติของออราเคิล 10g ที่มีในฐานข้อมูลเชิงวัตถุสัมพันธ์

ในส่วนนี้จะอธิบายถึงคุณสมบัติต่างๆ ของฐานข้อมูลเชิงวัตถุสัมพันธ์ที่ออราเคิล 10g สนับสนุน และกล่าวถึงศัพท์เฉพาะของฐานข้อมูลเชิงวัตถุสัมพันธ์ที่ออราเคิล 10g ใช้ในการอ้างอิง

3.1.1 ชนิดข้อมูลออบเจกต์ (Object Type)

ชนิดข้อมูลออบเจกต์ (Object Type) ในออราเคิล 10g เป็นชนิดข้อมูลแบบหนึ่ง ที่เป็นชนิดข้อมูลที่ใช้สร้างขึ้นมาเอง (User Defined Type) ซึ่งสามารถถูกใช้งานได้เหมือนกับข้อมูลชนิดพื้นฐานทั่วไป เช่น ตัวเลข (Number), ข้อความ (Varchar2), ค่าตรรกะ (Boolean), วันที่ เป็นต้น

ชนิดข้อมูลออบเจกต์ มีความแตกต่างกับชนิดข้อมูลพื้นฐานดังนี้

1. ชนิดข้อมูลออบเจกต์ไม่ได้มีมาให้พร้อมกับฐานข้อมูลในตอนแรก ผู้ใช้จะต้องเป็นผู้สร้างชนิดข้อมูลตามที่ต้องการ ใช้งานขึ้นมาเอง
2. ชนิดข้อมูลออบเจกต์ ประกอบขึ้นมาจากชนิดข้อมูลหลายๆ แบบ รวมไปถึงเมธอดที่ใช้ในการจัดการข้อมูลด้วย

Object Type <i>person_typ</i>	
Attributes	Methods
idno	get_idno
first_name	display_details
last_name	
email	
phone	

Object	
idno:	65
first_name:	Vema
last_name:	Mills
email:	vmills@oracle.com
phone:	1-800-555-4412

Object	
idno:	101
first_name:	John
last_name:	Smith
email:	jsmith@oracle.com
phone:	1-800-555-1212

รูปที่ 3.1 แสดงชนิดข้อมูลออบเจกต์ (Object Type) และออบเจกต์อินสแตนซ์ (Object Instances)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ออปเจกต์หรืออินสแตนซ์ (Objects or Instances)

เมื่อมีการสร้างชนิดข้อมูลออปเจกต์ (Object Type) ขึ้นมา ชนิดข้อมูลออปเจกต์นั้นสามารถที่จะมี อินสแตนซ์ (Instance) หรือออปเจกต์ (Object) เป็นของตัวเองได้ ซึ่งอินสแตนซ์จะมีแอตทริบิวต์ (Attribute) และเมธอดตามชนิดข้อมูลออปเจกต์นั้นๆ ซึ่งอินสแตนซ์นี้จะสามารถถูกเปลี่ยนแปลงค่าแอตทริบิวต์ภายในได้โดยตรง หรือผ่านทางเมธอดก็ได้

3.1.3 เมธอดของออปเจกต์ (Object Methods)

เมธอดเป็นฟังก์ชันการทำงาน ที่มีการประกาศไว้ในชนิดข้อมูลออปเจกต์ (Object Type) และมีการเขียนฟังก์ชันการทำงานไว้ในส่วนบอดี้ (Body) ของชนิดข้อมูลออปเจกต์ ซึ่งสามารถแบ่งประเภทของเมธอดได้ 4 ประเภทคือ เมมเบอร์เมธอด, เมธอดที่ใช้ในการเปรียบเทียบ, สเตติกเมธอด และคอนสตรัคเตอร์เมธอด

3.1.3.1 เมมเบอร์เมธอด (Member Methods)

เมมเบอร์เมธอด เป็นเมธอดที่ใช้ในการคำนวณค่าหรือเพื่อเข้าถึงค่าภายในอินสแตนซ์ ซึ่งเหมือนกับฟังก์ชันในภาษาโปรแกรมมิ่งทั่วไป ภายในเมมเบอร์เมธอดนี้จะมีตัวแปรแบบบิวท์อินชื่อว่า SELF ซึ่งเป็นการใช้เพื่ออ้างถึงอินสแตนซ์ของตัวเอง หรืออ้างถึงอินสแตนซ์ ตัวที่เรียกใช้เมธอดนี้ ซึ่งสามารถใช้ SELF ตามด้วยชื่อของแอตทริบิวต์ตัวที่ต้องการได้ ตัวแปร SELF นี้สามารถละไว้โดยไม่ต้องเขียนก็ได้

การเรียกใช้เมมเบอร์เมธอดนี้ทำได้โดยใช้ตัวแปรของออปเจกต์ของชนิดข้อมูลนั้นตามด้วยจุดแล้วเรียกใช้ชื่อเมธอด โดยมีเครื่องหมาย () ตามท้าย ดังนี้ `Object_variable.method ()`

3.1.3.2 เมธอดที่ใช้ในการเปรียบเทียบ (Methods for Comparing Objects)

โดยปกติแล้วการเปรียบเทียบข้อมูลที่มีชนิดเป็นสเกลาร์นั้นสามารถทำได้ แต่กรณีที่ต้องการทำการเปรียบเทียบชนิดข้อมูลที่เป็นออปเจกต์นั้นจะทำได้ เพราะไม่รู้ว่าจะต้องใช้ค่าไหนในการเปรียบเทียบ ดังนั้นจึงต้องมีการสร้างเมธอดที่ใช้ในการทำการเปรียบเทียบขึ้นมา ซึ่งเมธอดในการเปรียบเทียบที่ออราเคิล ได้กำหนดไว้มีสองแบบดังนี้

3.1.3.2.1 แมพเมธอด (Map Methods)

แมพเมธอด เป็นเมธอดที่ใช้ในการเปรียบเทียบระหว่างออปเจกต์ด้วยกัน จะทำการเปรียบเทียบออปเจกต์โดยทำการแปลงค่าของออปเจกต์ให้เป็นสเกลาร์

แมพเมธอดเป็นเมธอดที่ไม่มีพารามิเตอร์ และจะมีการใช้คีย์เวิร์ด MAP นำหน้า ซึ่งต้องมีการส่งคืนค่าเป็นชนิดข้อมูลแบบสเกลาร์ เมื่อมีการเปรียบเทียบระหว่างออปเจกต์ด้วยกัน แมพเมธอดนี้จะถูกเรียกอัตโนมัติ โดยที่ไม่ต้องทำการเรียกชื่อเมธอดเอง เมื่อมีการสร้างแมพเมธอดขึ้นมาแล้ว จะสามารถใช้ Distinct, Group by, Union, Order by, Min และ Max เพื่อจัดการกับชนิดข้อมูลแบบออปเจกต์ได้

3.1.3.2.2 ออเดอร์เมธอด (Order Methods)

ออเดอร์เมธอด เป็นเมธอดที่ใช้ในการเปรียบเทียบระหว่างออปเจกต์ด้วยกันอีกแบบหนึ่ง ซึ่งเป็นการเปรียบเทียบระหว่างออปเจกต์โดยตรง

ออเดอร์เมธอดเป็นเมธอดที่มีการรับพารามิเตอร์หนึ่งตัว โดยพารามิเตอร์นั้นจะมีชนิดข้อมูลเป็นออปเจกต์ของชนิดข้อมูลนั้น และจะมีการใช้คีย์เวิร์ด ORDER นำหน้า ซึ่งต้องมีการส่งคืนค่าเป็นค่าบวก ค่าลบ และค่าศูนย์ เป็นการแสดงถึงการเปรียบเทียบที่ได้ผลมากกว่า น้อยกว่า และเท่ากัน เมื่อมีการเปรียบเทียบกันระหว่างออปเจกต์เช่นเดียวกับแมพเมธอด ออเดอร์เมธอดจะถูกเรียกอัตโนมัติ และสามารถทำงานได้เหมือนกันกับแมพเมธอดทุกประการ

ในชนิดข้อมูลออปเจกต์ (Object Type) ใดๆจะสามารถมีแมพเมธอดหรือออเดอร์เมธอดได้แค่อย่างใดอย่างหนึ่งเท่านั้น

3.1.3.3 สเตติกเมธอด (Static Methods)

สเตติกเมธอดนั้นเป็นการเรียกใช้จากชนิดข้อมูลออปเจกต์ (Object Type) โดยตรง ซึ่งไม่ต้องทำการเรียกผ่านทางอินสแตนซ์เหมือนกับเมมเบอร์เมธอด แมพเมธอด และออเดอร์เมธอด ภายในสเตติกเมธอดนี้จะไม่มีความแปร SELF ดังนั้นจึงไม่สามารถอ้างถึงข้อมูลภายในชนิดข้อมูลนั้นๆ ได้ ซึ่งการสร้างสเตติกเมธอดทำได้โดยใช้คีย์เวิร์ด STATIC นำหน้า

3.1.3.4 คอนสตรัคเตอร์เมธอด (Constructor Methods)

คอนสตรัคเตอร์เมธอด เป็นเมธอดที่ถูกสร้างขึ้นให้โดยอัตโนมัติโดยระบบเมื่อมีการสร้าง ชนิดข้อมูลออปเจกต์ (Object Type)

เมื่อมีการสร้างอินสแตนซ์ (Instance) ของชนิดข้อมูลออปเจกต์ (Object Type) นั้น คอนสตรัคเตอร์เมธอดจะถูกเรียกใช้งาน และจะมีการคืนค่าเป็นอินสแตนซ์ใหม่ ซึ่งชื่อของคอนสตรัคเตอร์เมธอดจะมีชื่อเดียวกันกับชื่อของชนิดข้อมูลออปเจกต์ (Object Type) นั้นๆ

3.1.4 การสืบทอดของชนิดข้อมูลที่เป็นออปเจกต์

ในฐานะข้อมูลของออรากิล 10g ในส่วนการทำงานกับฐานข้อมูลเชิงวัตถุสัมพันธ์นั้น สนับสนุนการสืบทอดชนิดข้อมูล ซึ่งสามารถสร้างลำดับชั้นของชนิดข้อมูลโดยการสืบทอดได้ โดยซับไทป์ (Subtype) จะทำการสืบทอดมาจากซูเปอร์ไทป์ (Supertype) เรียกว่า Specialization เพราะนอกจากจะสืบทอดคุณสมบัติและเมธอดจากซูเปอร์ไทป์ (Supertype) มาแล้ว ยังสามารถที่จะเพิ่มคุณสมบัติหรือเมธอดอื่นๆ เพิ่มเติมได้อีก

ซึ่งสามารถทำ Specialization กับ Subtype ได้โดยใช้วิธีการดังนี้

- เพิ่มแอตทริบิวต์ใหม่
- เพิ่มเมธอดใหม่
- เปลี่ยนแปลงการทำงานของเมธอดเดิม (Override)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 การเปลี่ยนแปลงชนิดข้อมูลออปเจกต์

สามารถที่จะทำการเปลี่ยนแปลงโครงสร้างของชนิดข้อมูลออปเจกต์ที่ได้สร้างขึ้น โดยใช้คำสั่ง ALTER TYPE ซึ่งคำสั่งนี้จะสามารถใช้งานได้หลากหลายรูปแบบดังนี้

- เพิ่ม หรือ ลบแอตทริบิวต์
- เพิ่ม หรือ ลบเมธอด
- เปลี่ยนแปลงขนาดของชนิดข้อมูล
- เปลี่ยนแปลงคุณสมบัติอื่นๆ ของชนิดข้อมูลออปเจกต์

3.1.6 ออปเจกต์เทเบิล (Object Tables)

ออปเจกต์เทเบิล เป็นตารางที่มี 1 คอลัมน์ เท่านั้น และคอลัมน์นั้นมีชนิดข้อมูลที่เป็นออปเจกต์

ซึ่งสามารถแสดงค่าที่อยู่ภายในออปเจกต์เทเบิลได้สองแบบดังนี้

1. แสดงค่าในรูปแบบของคอลัมน์เดี่ยว หรือ Single Column แต่ละแถวของข้อมูลจะแสดงเป็น ชนิดข้อมูลออปเจกต์ซึ่งสามารถใช้คำสั่งทางออปเจกต์จัดการได้
2. แสดงค่าในรูปแบบของหลายคอลัมน์ หรือ Multi-Column แต่ละแถวของข้อมูล จะแสดงเป็นค่าของแอตทริบิวต์ที่อยู่ในชนิดข้อมูลออปเจกต์ (Object Type) นั้น สามารถใช้คำสั่งภาษาเอสคิวแอลที่ใช้ในฐานะข้อมูลเชิงสัมพันธ์จัดการได้

3.1.7 ไร้วออปเจกต์ (Row Objects)

คือ ออปเจกต์ที่ถูกเก็บเป็นข้อมูล 1 เรคอร์ดในออปเจกต์เทเบิล (Object Table)

3.1.8 คอลัมน์ออปเจกต์ (Column Objects)

คือ ออปเจกต์ที่ถูกเก็บอยู่ในคอลัมน์ในตารางใดๆ

3.1.9 ข้อมูลชนิดอ้างอิง (References)

เป็นลอจิคอลพอยเตอร์ (Logical Pointer) ที่ชี้ไปยังไร้วออปเจกต์ (Row Object) ใดๆ โดยข้อมูลชนิดอ้างอิง (Reference) จะเก็บออปเจกต์ไอดี (OID) ของไร้วออปเจกต์นั้นๆ ข้อมูลชนิดอ้างอิง (Reference) ในออรากิล 10g จะเป็นชนิดข้อมูลพื้นฐาน สามารถนำแนวคิดของการอ้างอิงนี้มาใช้แทนแนวคิดคีย์นอก (Foreign Key) ได้

เนื่องจากการใช้ข้อมูลชนิดอ้างอิง (Reference) เป็นการอ้างอิง Row Type ที่มีชนิดข้อมูลเป็น ออปเจกต์ใดๆ ดังนั้นอาจจะมีมากกว่าหนึ่งตารางที่มีการใช้ชนิดข้อมูลออปเจกต์นี้ ถ้าต้องการที่จะอ้างอิง Row Type ในตารางใดตารางหนึ่งก็สามารถทำได้โดยทำการกำหนดขอบเขต (Scope) ของการอ้างอิง

ในกรณีที่เราใช้ ข้อมูลชนิดอ้างอิง (Reference) เพื่ออ้างอิง Row Type แต่ว่า Row Type นั้นอาจจะไม่มีอยู่จริง หรืออาจโดนลบไปแล้ว ในกรณีนี้จะเกิดเหตุการณ์ที่เรียกว่า Dangling REF ซึ่งก็สามารถป้องกันได้โดยการใช้ คีย์นอก (Foreign Key) เข้ามาช่วยอีกทีหนึ่ง

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.10 คอลเลกชันไทป์ (Collection Type)

เป็นชนิดข้อมูลที่สามารมีค่าได้หลายค่าหรือเรียกว่า Multi-Valued ซึ่งออราเคิล 10g ก็สนับสนุน Collection Type อยู่สองชนิดคือวาริเอเบิลอาร์เรย์ (VARRAY) และเน็สต์เตเบิล (Nested Table)

3.1.11 PL/SQL

PL/SQL (Procedure Language SQL) คือ การนำคำสั่งต่างๆ ของภาษาเอสคิวแอลมา รวมกันเป็นโปรแกรมทำให้คำสั่งเหล่านั้นมีประสิทธิภาพ และทำงานได้ตรงตามความต้องการของผู้ใช้

คุณสมบัติของ PL/SQL มีข้อที่น่าสนใจดังนี้

- รูปแบบการเขียนโปรแกรมคล้ายกับภาษาปาสคาล (Pascal) เช่น มีการเริ่มต้นและจบโปรแกรมด้วย BEGIN....END, มีการจบแต่ละคำสั่งด้วยเครื่องหมายเซมิโคลอน (Semi colon ;)) เป็นต้น
- การใช้ตัวอักษรพิมพ์ใหญ่ หรือพิมพ์เล็กไม่มีผลต่อการเขียนคำสั่ง ลักษณะข้อนี้จะแตกต่างจากภาษาปาสคาล โดยที่ PL/SQL เป็น Non-Case sensitive ตัวอย่างเช่น SELECT...FROM หรือ SELECT...FROM สามารถคอมไพล์ผ่าน เรียกใช้งานได้เหมือนกัน

3.2 การใช้คำสั่งภาษาเอสคิวแอลในการจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์

3.2.1 การใช้คำสั่งภาษาเอสคิวแอลกับชนิดข้อมูลออปเจกต์และการใช้ข้อมูลชนิดอ้างอิง

3.2.1.1 การสร้างชนิดข้อมูลออปเจกต์

รูปแบบคำสั่งการสร้างข้อมูลชนิดออปเจกต์

```
CREATE OR REPLACE TYPE person_typ AS OBJECT (
  idno          NUMBER,
  first_name    VARCHAR2 (20) ,
  last_name     VARCHAR2 (25) ,
  email         VARCHAR2 (25) ,
  phone        VARCHAR2 (20)
);
```

รูปที่ 3.2 คำสั่งการสร้างข้อมูลชนิดออปเจกต์

ในการสร้างข้อมูลชนิดออปเจกต์ในออราเคิล 10g ให้ใช้คีย์เวิร์ดคำสั่ง CREATE TYPE ตามด้วยชื่อของข้อมูลชนิดออปเจกต์ (Object Type) ที่ต้องการสร้าง ในส่วนของ OR REPLACE นี้ อาจจะมีหรือไม่มีก็ได้ ซึ่งถ้ามี OR REPLACE อยู่ จะสามารถรันสคริปต์เพื่อทำการเปลี่ยนแปลงโครงสร้างข้อมูลได้เลยโดยไม่ต้องใช้คำสั่ง ALTER TYPE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำว่า AS OBJECT นั้น จริงๆ แล้วออปเจกต์ที่ออราเคิลเรียกใช้นั้นไม่ได้มีคุณสมบัติของออปเจกต์จริงๆ ครบทั้ง 3 ประการคือ การปกปิดข้อมูล การสืบทอดข้อมูล และ โพลีมอร์ฟิซึม แต่ออปเจกต์ของออราเคิลนี้มีคุณลักษณะของออปเจกต์แค่ 2 ประการคือ การสืบทอดข้อมูล และ โพลีมอร์ฟิซึม โดยที่ไม่มีคุณสมบัติของการปกปิดข้อมูล และผู้ใช้สามารถเข้าถึงข้อมูลภายในข้อมูลชนิดออปเจกต์ (Object Type) ได้โดยตรง โดยไม่ต้องเรียกใช้ผ่านเมธอด

ภายในขอบเขตของเครื่องหมาย () ให้ประกาศแอตทริบิวต์ไว้ภายใน โดยขึ้นด้วย ชื่อของแอตทริบิวต์ตามด้วยชนิดของข้อมูลของแอตทริบิวต์นั้น และให้คั่นแอตทริบิวต์แต่ละตัวด้วยเครื่องหมาย คอมมา (,)

3.2.1.2 การสร้างออปเจกต์เทเบิล (Object Table)

รูปแบบคำสั่งการสร้างออปเจกต์เทเบิล (Object Table)

```
CREATE TABLE person_obj_table OF person_typ;
```

รูปที่ 3.3 คำสั่งการสร้างออปเจกต์เทเบิล (Object Table)

ใช้คำสั่งการสร้างตารางทั่วไป แต่ให้ใช้คีย์เวิร์ด OF ตามด้วยชื่อชนิดข้อมูลออปเจกต์นั้นๆ

3.2.1.3 การเพิ่มข้อมูลชนิดที่เป็นออปเจกต์

สำหรับในการเพิ่มข้อมูลชนิดที่เป็นออปเจกต์ไม่ว่าจะเป็นการเพิ่มข้อมูลลงไป ในตารางธรรมดาหรือตารางที่เป็นออปเจกต์ (Object Table) นั้นก็ทำได้เหมือนกัน โดยใช้คอนสตรัคเตอร์ของชนิดข้อมูลที่เป็นออปเจกต์นั้นมาช่วย ดังตัวอย่างด้านล่างนี้

รูปแบบคำสั่งเพิ่มข้อมูลชนิดออปเจกต์

```
INSERT INTO person_obj_table
VALUES (person_typ(1, 'FName', 'LName', 'Mail', '02543'));
```

รูปที่ 3.4 คำสั่งการเพิ่มข้อมูลชนิดออปเจกต์

จากตัวอย่างคำสั่งด้านบนจะเห็นว่าในการเพิ่มข้อมูลที่เป็นออปเจกต์จะต้องมีการนำคอนสตรัคเตอร์ของชนิดข้อมูลที่เป็นออปเจกต์นั้นเข้ามาช่วย ในที่นี้เป็นการเพิ่มข้อมูลลงออปเจกต์เทเบิล (Object Table) ของ person_typ ก็มีการใช้คอนสตรัคเตอร์ของ person_typ เข้ามาช่วยในการเพิ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.4 การใช้คอนสเตรนซ์ (Constraint)

รูปแบบการใช้คอนสเตรนซ์กับออปเจกต์เทเบิลและตารางที่มีแอตทริบิวต์เป็นชนิดข้อมูล ออปเจกต์

```
CREATE TYPE location_typ AS OBJECT (
    building_no NUMBER,
    city          VARCHAR2(40) );

CREATE TYPE office_typ AS OBJECT (
    office_id   VARCHAR(10),
    office_loc  location_typ,
    occupant    person_typ );

CREATE TABLE office_tab OF office_typ (
    office_id PRIMARY KEY );
```

รูปที่ 3.5 คำสั่งการใช้คอนสเตรนซ์กับ ออปเจกต์เทเบิล

```
CREATE TABLE department_mgrs (
    dept_no NUMBER PRIMARY KEY,
    dept_name CHAR(20),
    dept_mgr person_typ,
    dept_loc location_typ,
    CONSTRAINT dept_loc_cons1
        UNIQUE (dept_loc.building_no, dept_loc.city),
    CONSTRAINT dept_loc_cons2
        CHECK (dept_loc.city IS NOT NULL) );
```

รูปที่ 3.6 คำสั่งการใช้คอนสเตรนซ์กับตารางที่มีแอตทริบิวต์ เป็นชนิดข้อมูลออปเจกต์

การสร้างคอนสเตรนซ์บนออปเจกต์เทเบิลหรือบนตารางที่มีแอตทริบิวต์ เป็นชนิดข้อมูลออปเจกต์สามารถทำได้เหมือนกับการสร้างคอนสเตรนซ์บนตารางทั่วไป ซึ่งในกรณีที่จะสร้างคอนสเตรนซ์บน คอลัมน์ที่เป็นชนิดข้อมูลออปเจกต์นั้น จะต้องกำหนดชนิดข้อมูลที่เป็นชนิดข้อมูลพื้นฐานแยกย่อยที่สุดในชนิดข้อมูลออปเจกต์นั้น เพราะคอนสเตรนซ์ไม่สามารถสร้างบนชนิดของข้อมูลที่เป็นออปเจกต์ได้ นอกจากนี้แล้ว ยังไม่สามารถใช้คอนสเตรนซ์กับข้อมูลชนิดอ้างอิงที่ไม่มีขอบเขตอีกด้วย

3.2.1.5 การสร้างอินเด็กซ์

รูปแบบคำสั่งการสร้างอินเด็กซ์บนชนิดข้อมูลออปเจกต์ในตาราง

```
CREATE TABLE department_loc (
  dept_no      NUMBER PRIMARY KEY,
  dept_name    CHAR(20),
  dept_addr    location_typ );

CREATE INDEX   i_dept_addr1
              ON department_loc (dept_addr.city);
```

รูปที่ 3.7 คำสั่งการสร้างอินเด็กซ์บนชนิดข้อมูลออปเจกต์ ในตาราง

เช่นเดียวกับกับการใช้คอนสเตรนทอินเด็กซ์จะต้องสร้างบนส่วนที่แยกย่อยที่สุดของชนิดข้อมูลออปเจกต์คือ แอตทริบิวต์ ที่มีชนิดข้อมูลพื้นฐานเท่านั้น และสามารถใช้อินเด็กซ์กับชนิดข้อมูลอ้างอิง (Reference) ได้ก็ต่อเมื่อชนิดข้อมูลอ้างอิงนั้นมีขอบเขต

3.2.1.6 Reference การอ้างอิง

ชนิดข้อมูลอ้างอิง (Reference) จะเก็บค่าของออปเจกต์ไอดี (OID) ซึ่งสามารถอ้างอิงไปยังโรว์ไทป์ (Row Type) เฉพาะที่อยู่ในออปเจกต์เทเบิลเท่านั้น

รูปแบบคำสั่งใช้ชนิดข้อมูลอ้างอิง

```
CREATE TYPE emp_person_typ AS OBJECT (
  name      VARCHAR2(30),
  manager   REF emp_person_typ );

CREATE TABLE emp_person_obj_table OF emp_person_typ;

INSERT INTO emp_person_obj_table
  SELECT emp_person_typ ('Bob Jones', REF(e))
  FROM emp_person_obj_table e
  WHERE e.name = 'John Smith';
```

รูปที่ 3.8 คำสั่งการจัดการเกี่ยวกับชนิดข้อมูลอ้างอิง

จากตัวอย่างคำสั่งด้านบนเป็นการสร้างชนิดข้อมูลออปเจกต์ชื่อ emp_person_typ ซึ่งมีแอตทริบิวต์ ตัวหนึ่งมีชนิดข้อมูลเป็นแบบอ้างอิง (Reference) ที่อ้างอิงถึงชนิดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

emp_person_typ หรือทำการอ้างอิงถึงชนิดข้อมูลของตัวเอง เช่นความสัมพันธ์ของนายจ้างกับลูกน้อง ซึ่งนายจ้างก็มีชนิดข้อมูลเป็น ออปเจกต์เหมือนกัน

สำหรับในการเพิ่มข้อมูลลงไปในตารางนั้น ต้องใช้คำสั่ง INSERT ร่วมกับ SELECT โดยใช้คีย์เวิร์ด REF เข้ามาช่วยเพื่อดึงค่าออปเจกต์ไอดี (OID) โดยค่าที่ได้นี้จะต้องมีเพียงแค่หนึ่งค่าเท่านั้น ถ้าในคำสั่ง SELECT ส่งค่าคืนกลับมามากกว่าหนึ่งค่าจะเกิดการผิดพลาดขึ้น

อย่างที่เคยได้กล่าวแล้วว่าอาจจะมียากกว่าหนึ่งตารางที่สร้างจากชนิดข้อมูลออปเจกต์ ถ้าต้องการที่จะอ้างอิงโรว์ไทป์ (Row Type) เพียงแค่ในตารางใดตารางหนึ่งก็สามารถทำได้โดยการกำหนดขอบเขตของการอ้างอิง ดังตัวอย่างคำสั่งด้านล่างต่อไปนี้

```
CREATE TABLE sale_dept_table OF emp_person_typ;
CREATE TABLE account_dept_table OF emp_person_typ;

CREATE TABLE account_ref(
    start_date      VARCHAR2(10),
    account_staff   emp_person_typ SCOPE account_dept_table
);
```

รูปที่ 3.9 การใช้ขอบเขตกับชนิดข้อมูลแบบอ้างอิง (Reference)

3.2.2 การประกาศและสร้างเมธอดในชนิดข้อมูลออปเจกต์

ในส่วนของการประกาศเมธอดนั้น ทำได้ในส่วนของการสร้างชนิดข้อมูลออปเจกต์ โดยทำการระบุประเภทของเมธอดที่ต้องการสร้าง ชื่อเมธอด และชนิดข้อมูลที่จะส่งคืน แต่สำหรับในส่วนของการสร้างเมธอด ซึ่งเป็นส่วนที่ต้องเขียนว่าเมธอดนี้มีการทำงานอย่างไร ทำได้โดยเขียนไว้ในส่วนของบอดี้ของชนิดข้อมูลออปเจกต์ ในที่นี้จะเป็นการสร้างส่วนของเมธอดโดยใช้ภาษา PL/SQL

3.2.2.1 เมมเบอร์เมธอด (Member Methods)

รูปแบบคำสั่งการประกาศเมมเบอร์เมธอด

```
CREATE TYPE solid_typ AS OBJECT (
    len      INTEGER,
    wth      INTEGER,
    hgt      INTEGER,
    MEMBER FUNCTION surface RETURN INTEGER,
    MEMBER FUNCTION volume RETURN INTEGER
);
```

รูปที่ 3.10 คำสั่งการประกาศเมมเบอร์เมธอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำสั่งการสร้างเมมเบอร์เมทอด

```
CREATE TYPE BODY solid_typ AS
  MEMBER FUNCTION volume RETURN INTEGER IS
  BEGIN
    RETURN SELF.len * SELF.wth * SELF.hgt;
  END;

  MEMBER FUNCTION surface RETURN INTEGER IS
  BEGIN
    RETURN 2 * (len * wth + len * hgt + wth * hgt);
  END;
END;
```

รูปที่ 3.11 คำสั่งการสร้างเมมเบอร์เมทอด

3.2.2.2 เมทอดที่ใช้ในการเปรียบเทียบ (Methods for Comparing Objects)

3.2.2.2.1 แมพเมทอด (Map Methods)

รูปแบบคำสั่งการประกาศแมพเมทอด

```
CREATE TYPE rectangle_typ AS OBJECT (
  len NUMBER,
  wid NUMBER,
  MAP MEMBER FUNCTION area RETURN NUMBER
);
```

รูปที่ 3.12 คำสั่งการประกาศแมพเมทอด

รูปแบบคำสั่งการสร้างแมพเมทอด

```
CREATE TYPE BODY rectangle_typ AS
  MAP MEMBER FUNCTION area RETURN NUMBER IS
  BEGIN
    RETURN len*wid;
  END area;
End;
```

รูปที่ 3.13 คำสั่งการสร้างแมพเมทอด

3.2.2.2 ออเดอร์เมธอด (Order Methods)

รูปแบบคำสั่งการประกาศออเดอร์เมธอด

```
CREATE TYPE location_typ AS OBJECT (
  building_no NUMBER,
  city          VARCHAR2 (40),
  ORDER MEMBER FUNCTION match (1 location_typ) RETURN INTEGER
);
```

รูปที่ 3.14 คำสั่งการประกาศออเดอร์เมธอด

รูปแบบคำสั่งการสร้างออเดอร์เมธอด

```
CREATE TYPE BODY location_typ AS
  ORDER MEMBER FUNCTION match (1 location_typ) RETURN INTEGER IS
  BEGIN
    IF building_no < 1.building_no THEN RETURN -1;
    ELSIF building_no > 1.building_no THEN RETURN 1;
    ELSE RETURN 0;
    END IF;
  END;
END;
```

รูปที่ 3.15 คำสั่งการสร้างออเดอร์เมธอด

3.2.2.3 สเตติกเมธอด (Static Methods)

รูปแบบคำสั่งการประกาศสเตติกเมธอด

```
CREATE TYPE person_typ AS OBJECT (
  idno          NUMBER,
  name          VARCHAR2 (30),
  phone        VARCHAR2 (20),
  STATIC FUNCTION show_greeting (str VARCHAR2) RETURN VARCHAR2
);
```

รูปที่ 3.16 คำสั่งการประกาศสเตติกเมธอด

รูปแบบคำสั่งการสร้างสแตติกเมธอด

```
CREATE TYPE BODY person_typ AS
  STATIC FUNCTION show_greeting (str VARCHAR2) RETURN VARCHAR2 IS
  BEGIN
    RETURN 'Hello ' || str;
  END;
END;
```

รูปที่ 3.17 คำสั่งการสร้างสแตติกเมธอด

3.2.3 การสืบทอด (Inheritance)

อย่างที่ได้อธิบายไว้แล้วในบทที่ 2 ว่าการสืบทอดเป็นหลักการนำคลาสที่ได้สร้างไว้แล้ว นำกลับมาใช้ใหม่ โดยเพิ่มคุณสมบัติบางประการเข้าไป ซึ่งการสืบทอดนั้นมีอยู่ 2 ชนิดคือ Single Inheritance และ Multiple Inheritance แต่ในออราเคิล 10g นั้นสนับสนุนให้มีการใช้งานแค่ Single Inheritance เท่านั้น ซึ่งมีการใช้งานดังนี้

รูปแบบคำสั่งการสืบทอด



รูปที่ 3.18 ตัวอย่างโครงสร้างข้อมูลที่ทำให้การสืบทอด

```
CREATE TYPE person_type AS OBJECT (
  idno NUMBER,
  name VARCHAR2 (30),
  phone VARCHAR2 (20),
  FINAL MEMBER FUNCTION get_idno RETURN NUMBER,
  MEMBER FUNCTION show RETURN VARCHAR2
) NOT FINAL;
```

รูปที่ 3.19 คำสั่งการสร้างซูเปอร์ไทป์ (Supertype)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE TYPE student_type UNDER person_type (
    dept_id NUMBER,
    major    VARCHAR2 (30)
    OVERRIDING MEMBER FUNCTION show RETURN VARCHAR2
);

CREATE TYPE employee_type UNDER person_type (
    emp_id NUMBER,
    mgr     VARCHAR2 (30)
);

```

รูปที่ 3.20 คำสั่งการสร้างซับไทป์ (Subtype)

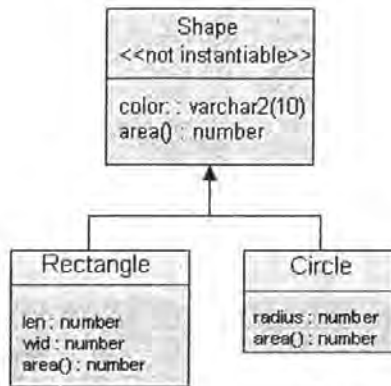
สำหรับการสร้างซูเปอร์ไทป์ (Supertype) จะต้องทำการประกาศคีย์เวิร์ด NOT FINAL ไว้ที่ชนิดข้อมูลออบเจกต์ที่ทำการสร้าง เพื่อบอกว่าให้ชนิดข้อมูลออบเจกต์นี้สามารถมีชนิดข้อมูลสืบทอดได้ และเมื่อต้องการสร้างชนิดข้อมูลสืบทอดก็จะต้องใช้คีย์เวิร์ด UNDER ตามด้วยชื่อชนิดข้อมูลออบเจกต์ที่ทำการสืบทอด เมื่อทำการสืบทอดแล้วซับไทป์ (Subtype) จะมีแอตทริบิวต์และเมธอดเหมือนกับซูเปอร์ไทป์ (Supertype) ทุกประการ และในซับไทป์ (Subtype) สามารถที่จะเพิ่ม แอตทริบิวต์ หรือเมธอดอื่นเพิ่มเติมได้

โดยปกติแล้วชนิดข้อมูลออบเจกต์จะมีคุณสมบัติเป็น FINAL คือไม่ยอมให้ชนิดข้อมูลอื่นมา สืบทอดไปจากตัวมันเอง ถ้าเราต้องการให้มันสืบทอดได้จะต้องใช้คีย์เวิร์ด NOT FINAL ซึ่งต่างกับเมธอด โดยปกติแล้วเมธอดจะมีคุณสมบัติเป็น NOT FINAL คือยอมให้ซับคลาส (Subclass) นั้นสามารถโอเวอร์ไรด์ (Override) เมธอดนั้นใหม่ได้ และเช่นเดียวกันถ้าไม่ต้องการให้เมธอดนั้นถูกโอเวอร์ไรด์ (Override) ก็จะต้องทำการประกาศ FINAL ให้กับเมธอดนั่นเอง

3.2.4 โพลีมอร์ฟิซึม (Polymorphism)

ในออราเคิล 10g สนับสนุนการทำโพลีมอร์ฟิซึม (Polymorphism) โดยสนับสนุนการทำ NOT INSTANTIABLE ให้กับชนิดข้อมูลออบเจกต์ซึ่งชนิดข้อมูลออบเจกต์ที่มีคุณสมบัติเป็น NOT INSTANTIABLE นี้ จะไม่สามารถมีอินสแตนซ์ (Instance) ที่สร้างโดยคอนสตรัคเตอร์ของตัวเองได้ โดยจะมีการประกาศแอตทริบิวต์และเมธอดที่ไม่มีการกำหนดการทำงานไว้ (Not Instantiable Method) และจะทำตัวเป็นซูเปอร์ไทป์ (Supertype) เพื่อให้ซับไทป์ (Subtype) อื่นมาสืบทอดแล้วทำการสร้างเมธอดที่ได้ทำการประกาศไว้ ดังตัวอย่างการทำงานด้านล่างนี้

รูปแบบคำสั่งการทำ NOT INSTANTIABLE ให้กับชนิดข้อมูลออบเจกต์



รูปที่ 3.21 ตัวอย่างโครงสร้างข้อมูลที่ใช้โพลีมอร์ฟิซึม

```

CREATE TYPE shape AS OBJECT (
  color VARCHAR2(10),
  NOT INSTANTIABLE MEMBER FUNCTION area RETURN NUMBER)
NOT INSTANTIABLE NOT FINAL;
  
```

รูปที่ 3.22 คำสั่งการสร้างชนิดข้อมูลออบเจกต์ (Object Type) ที่มีคุณสมบัติ NOT INSTANTIABLE

```

CREATE TYPE rectangle UNDER shape (
  len NUMBER,
  wid NUMBER,
  OVERRIDING MEMBER FUNCTION area RETURN NUMBER
);

CREATE TYPE BODY rectangle AS
  OVERRIDING MEMBER FUNCTION area RETURN NUMBER IS
  BEGIN
    RETURN wid*len;
  END;
END;
  
```

รูปที่ 3.23 คำสั่งการสร้างซับไทป์ที่สืบทอดมาจากชนิดข้อมูลออบเจกต์ที่เป็น NOT INSTANTIABLE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE TYPE circle UNDER shape(
  radius      NUMBER,
  OVERRIDING MEMBER FUNCTION area RETURN NUMBER
);

CREATE TYPE BODY circle AS
  OVERRIDING MEMBER FUNCTION area RETURN NUMBER IS
  BEGIN
    RETURN 3.14*radius*radius;
  END;
END;

```

รูปที่ 3.24 คำสั่งการสร้างซัพ ไทป์ที่สืบทอดมาจากชนิดข้อมูลออบเจกต์ที่เป็น NOT INSTANTIABLE 2

จากตัวอย่างคำสั่งด้านบน จะเห็นได้ว่าการสร้างชนิดข้อมูลออบเจกต์ที่มีคุณสมบัติเป็น NOT INSTANTIABLE นั้นจะต้องทำการเพิ่มคุณสมบัติ NOT FINAL เข้าไปด้วย เพื่อให้ชนิดข้อมูลอื่นๆ สามารถสืบทอดข้อมูลชนิดนี้และทำการสร้างเมธอดที่เป็น NOT INSTANTIABLE ได้

ชนิดข้อมูลออบเจกต์ (Object Type) ใด ที่ถูกประกาศให้เป็น NOT INSTANTIABLE แล้ว เมธอดภายในชนิดข้อมูลนั้นก็จะมีคุณสมบัติเป็น NOT INSTANTIABLE ตามไปด้วย ซึ่ง NOT INSTANTIABLE Method คือ เมธอดที่มีแค่การประกาศชื่อเมธอดเท่านั้น โดยที่ไม่มีการสร้างขั้นตอนการทำงานภายใน และเช่นเดียวกัน ถ้าชนิดข้อมูลออบเจกต์ใดมีเมธอดที่เป็น NOT INSTANTIABLE แล้ว ชนิดข้อมูลออบเจกต์นั้นก็จะต้องประกาศเป็น NOT INSTANTIABLE ตามไปด้วย

ชนิดข้อมูลออบเจกต์ (Object Type) ใดที่ทำการสืบทอดมาจากชนิดข้อมูลออบเจกต์ (Object Type) ที่เป็น NOT INSTANTIABLE จะต้องทำการโอเวอร์ไรด์(Override) เมธอดที่ได้จากซูเปอร์ไทป์ (Supertype) ให้หมด มิฉะนั้นแล้วชนิดข้อมูลออบเจกต์ (Object Type) นั้นก็จะต้องทำตัวเป็น NOT INSTANTIABLE ต่อไป

3.2.5 คอลเลกชัน (Collection Type)

ในออรากิล 10g มีการสนับสนุนการทำคอลเลกชันไทป์ (Collection Type) คือการยอมให้ 1 คอลัมน์มีข้อมูลมากกว่า 1 ชุดได้ สำหรับออรากิล 10g มีคอลเลกชันไทป์ (Collection Type) ให้ใช้ 2 ประเภทดังนี้

3.2.5.1 วาริเอเบิลอาร์เรย์ (VARRAY)

VARRAY เป็นเซตของข้อมูลที่มีลำดับ ซึ่งข้อมูลที่ถูกเก็บอยู่ใน VARRAY จะเป็นข้อมูลชนิดเดียวกัน ในการสร้าง VARRAY นั้นจำเป็นที่จะต้องกำหนดขนาดสูงสุดของข้อมูลที่จะสามารถเก็บได้ ซึ่งเลขที่ใช้ระบุขนาดของ VARRAY นี้ สามารถที่จะเปลี่ยนแปลงภายหลังได้ จึงเรียกว่า Variable Length Array

รูปแบบคำสั่งการสร้าง VARRAY

```
CREATE TYPE phone_typ AS OBJECT (
    country_code  VARCHAR2(2),
    area_code     VARCHAR2(3),
    ph_number     VARCHAR2(7));

CREATE TYPE phone_varray_typ AS VARRAY(5) OF phone_typ;
```

รูปที่ 3.25 การสร้าง VARRAY จากชนิดข้อมูลแบบ ออปเจกต์

```
CREATE TABLE dept_phone_list (
    dept_no NUMBER(5),
    phone_list phone_varray_typ);

INSERT INTO dept_phone_list VALUES (
    100,
    phone_varray_typ( phone_typ ('01', '650', '5061111'),
                     phone_typ ('01', '650', '5062222'),
                     phone_typ ('01', '650', '5062525')));
```

รูปที่ 3.26 การเพิ่มข้อมูลชนิดที่เป็น VARRAY

จากตัวอย่างคำสั่งด้านบนเป็นการสร้างชนิดข้อมูลแบบออปเจกต์ ชื่อว่า phone_typ จากนั้นทำการสร้างชนิดข้อมูลที่เป็น VARRAY จากชนิดข้อมูล phone_typ มีการจองพื้นที่ไว้สำหรับข้อมูล 5 ชุด ซึ่งการสร้างชนิดข้อมูลที่เป็น VARRAY นั้นจะสามารถสร้างได้จากชนิดข้อมูลพื้นฐานต่างๆ ไปหรืออาจจะสร้างจากชนิดข้อมูลแบบออปเจกต์ ก็ได้

เมื่อต้องการทำการเพิ่มข้อมูลเข้าไปในชนิดข้อมูลที่เป็น VARRAY ทำได้เหมือนการเพิ่มข้อมูลชนิดออปเจกต์ ทั่วไป และในกรณีที่ต้องการเพิ่มข้อมูลมากกว่า 1 ชุด ก็ให้ใส่เครื่องหมายคอมมา (,) เพื่อคั่นข้อมูลแต่ละชุด

สำหรับข้อมูลที่ถูกเก็บใน VARRAY นั้น ไม่สามารถที่จะถูกควิรี่เข้าไปถึงแต่ละอีลีเมนต์ข้างในได้ ทำได้แค่เพียงดึงข้อมูลทั้งชุดมาแสดงหรือเปลี่ยนแปลงแก้ไขเท่านั้น

3.2.5.2 เนสต์เตดเทเบิล (Nested Table)

เนสต์เตดเทเบิล (Nested Table) เป็นเซตของข้อมูลที่ไม่มีลำดับ ข้อมูลที่ถูกเก็บอยู่ในเนสต์เตดเทเบิล (Nested Table) จะต้องเป็นข้อมูลชนิดเดียวกัน และไม่จำเป็นต้องมีการระบุขนาดสูงสุดของข้อมูลที่จะเก็บ โดยชนิดข้อมูลที่เป็นเนสต์เตดเทเบิล (Nested Table) นี้จะถูกเก็บแยกคนละที่กับตารางหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำสั่งการสร้างเนสต์เตเบิล (Nested Table)

```
CREATE TYPE person_typ AS OBJECT (
    idno        NUMBER,
    name        VARCHAR2 (30) ,
    phone       VARCHAR2 (20)
);

CREATE TYPE people_typ AS TABLE OF person_typ;
```

รูปที่ 3.27 การสร้างเนสต์เตเบิล (Nested Table) จากชนิดข้อมูลแบบ ออปเจกต์

```
CREATE TABLE students (
    graduation    DATE,
    math_majors   people_typ,
    chem_majors   people_typ,
    physics_majors people_typ)
NESTED TABLE math_majors STORE AS math_majors_nt
NESTED TABLE chem_majors STORE AS chem_majors_nt
NESTED TABLE physics_majors STORE AS physics_majors_nt;
```

รูปที่ 3.28 การสร้างตารางที่ภายในประกอบด้วยข้อมูลชนิดเนสต์เตเบิล (Nested Table)

จากตัวอย่างคำสั่งด้านบน เป็นการสร้างชนิดข้อมูลออปเจกต์ (Object Type) ชื่อว่า person_typ จากนั้นทำการสร้างข้อมูลชนิดเนสต์เตเบิล (Nested Table) ชื่อว่า people_typ จาก person_typ ซึ่งในการสร้างชนิดข้อมูลแบบเนสต์เตเบิล (Nested Table) นี้ไม่จำเป็นที่จะต้องกำหนดขนาดการของข้อมูลไว้ตั้งแต่แรก เช่นเดียวกับ VARRAY การสร้างชนิดข้อมูลแบบเนสต์เตเบิล (Nested Table) นี้อาจจะสร้างจากชนิดข้อมูลพื้นฐาน หรือสร้างจากชนิดข้อมูลแบบ ออปเจกต์ ก็ได้

สำหรับการสร้างตาราง ที่ภายในประกอบด้วยชนิดข้อมูลแบบเนสต์เตเบิล (Nested Table) นั้น จำเป็นที่จะต้องทำการกำหนดชื่อตารางย่อยด้วย เพราะออรากิลจะทำการเก็บข้อมูลชนิดที่เป็นเนสต์เตเบิล (Nested Table) แยกไว้คนละที่กับตารางหลัก โดยใช้คีย์เวิร์ด STORE AS แล้วตามด้วยชื่อตารางของ เนสต์เตเบิล (Nested Table) นั้น

ข้อมูลชนิดแบบเนสต์เตเบิล (Nested Table) นั้นต่างจากข้อมูลชนิด VARRAY ตรงที่ เราสามารถที่จะควิรี่เข้าไปถึงแต่ละอีลิเมนต์ภายในเนสต์เตเบิล (Nested Table) ได้ ซึ่งจะสร้างชนิดข้อมูลแบบ VARRAY หรือเนสต์เตเบิล (Nested Table) นั้นก็ขึ้นอยู่กับรูปแบบการใช้งาน ถ้าต้องการเก็บไว้เฉยๆ แล้วนำขึ้นมาแสดงก็ให้ใช้ VARRAY หรือถ้าต้องการทำการค้นหาข้อมูลในคอลเลกชัน (Collection) นั้นด้วยก็ให้ใช้เนสต์เตเบิล (Nested Table)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

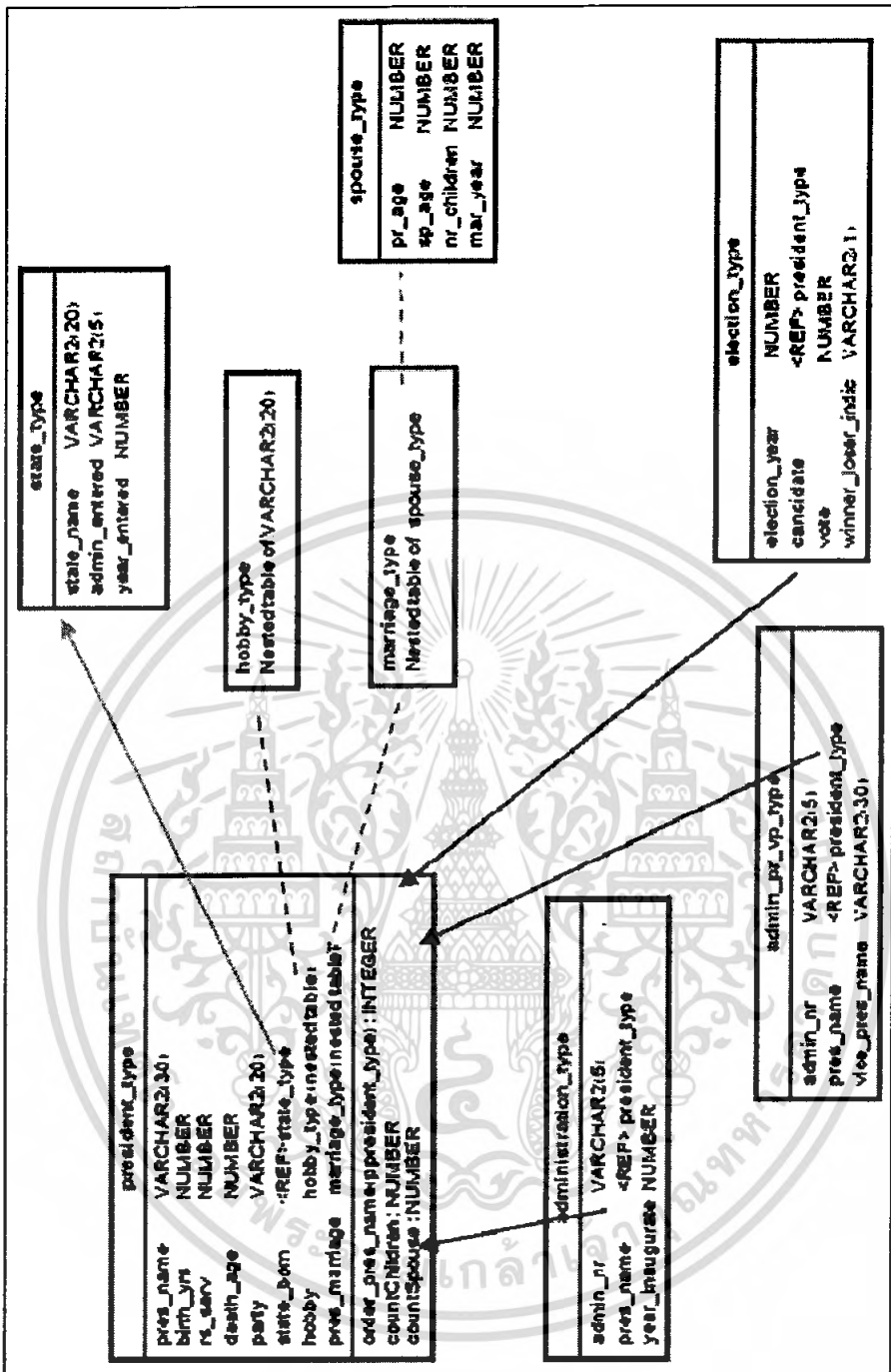
การทดสอบ

ในบทนี้จะเป็นการออกแบบฐานข้อมูลโดยใช้หลักการของฐานข้อมูลเชิงวัตถุสัมพันธ์ ซึ่งฐานข้อมูลที่นำมาใช้ในการออกแบบได้แก่

1. ฐานข้อมูลประธานาธิบดีของประเทศสหรัฐอเมริกา
2. ฐานข้อมูลรถ

4.1 การออกแบบฐานข้อมูลประธานาธิบดีในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์

ฐานข้อมูลประธานาธิบดี (Presdb) แต่เดิมเป็นฐานข้อมูลเชิงสัมพันธ์ นำมาออกแบบใหม่ให้เป็นฐานข้อมูลเชิงวัตถุสัมพันธ์ ซึ่งในขั้นตอนการออกแบบนั้น ตารางใดๆ ที่มีความสัมพันธ์ โดยใช้คีย์นอก (Foreign Key) อ้างถึงต่อกัน นั้นสามารถใช้คุณสมบัติของชนิดข้อมูลอ้างอิง (Reference) มาแทนได้ และตารางใดที่มีข้อมูลในลักษณะที่เป็นรีพีตติงกรุป (Repeating Group) อยู่ ก็ไม่จำเป็นที่จะต้องแยกตารางเก็บ สามารถใช้ชนิดข้อมูลแบบคอลเลกชัน (Collection) เช่น VARRAY หรือเน็สเต็ดเทเบิล (Nested Table) เข้ามาช่วยจัดการได้อีกด้วย ดังตัวอย่างการออกแบบด้านล่างนี้



รูปที่ 4.1 โครงสร้างโดยรวมของฐานข้อมูล Presdb ในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESIDENT_TYPE	เป็นชนิดข้อมูลที่ประกอบไปด้วยชื่อประธานาธิบดี ปีเกิด จำนวนปีทั้งหมดที่อยู่ในตำแหน่งประธานาธิบดี อายุขัยเมื่อเสียชีวิต พรรคการเมืองที่สังกัด รัฐที่เกิด รวมทั้งอ้างอิงไปยังชนิดข้อมูล STATE_TYPE, HOBBY_TYPE และ MARRIAGE_TYPE
ADMINISTRATION_TYPE	เป็นชนิดข้อมูลที่เก็บข้อมูล ลำดับที่ของสมัยการปกครอง และปี ค.ศ. ที่ประธานาธิบดีสาบานตนเพื่อเข้ารับตำแหน่งในสมัยการปกครอง รวมทั้งอ้างอิงไปยังชนิดข้อมูล PRESIDENT_TYPE
ADMIN_PR_VP_TYPE	เป็นชนิดข้อมูลที่เก็บข้อมูลของรองประธานาธิบดี (Vice President) ในแต่ละสมัยการปกครอง
STATE_TYPE	เป็นชนิดข้อมูลที่เก็บข้อมูลของรัฐต่าง ๆ โดยจะเก็บ ชื่อรัฐ สมัยการปกครอง และปี ค.ศ. ที่รัฐแต่ละรัฐขึ้นต่อสัญญาเข้าร่วมเป็นรัฐหนึ่งของประเทศสหรัฐ (รัฐที่มี Admin_nr = 0 หมายความว่าป็นรัฐแรก ๆ ที่เริ่มจับมือกันจัดตั้งเป็นประเทศสหรัฐอเมริกา ก่อนที่จะมีการจัดการเลือกตั้งครั้งแรกขึ้น)
ELECTION_TYPE	เป็นชนิดข้อมูลที่เก็บข้อมูลการเลือกตั้งของประเทศสหรัฐอเมริกา โดยจะจัดเก็บ ปี ค.ศ. ที่จัดการเลือกตั้ง ชื่อผู้สมัครเข้ารับเลือกตั้งเป็นประธานาธิบดี คะแนนเสียงที่ได้รับ และสถานะที่แสดงว่าผู้ใดเป็นผู้ได้รับการเลือกตั้ง
MARRIAGE_TYPE	เก็บเนื้สเค็ดเทเบิลของชนิดข้อมูล SPOUSE_TYPE
SPOUSE_TYPE	เป็นชนิดข้อมูลที่เก็บข้อมูลการสมรสของประธานาธิบดี โดยเก็บอายุของประธานาธิบดี และคู่สมรสในขณะที่ทำการสมรส จำนวนบุตรที่แต่ละคู่สมรสมีด้วยกัน และปี ค.ศ. ที่ทำการสมรส
HOBBY_TYPE	ชนิดข้อมูลที่เก็บข้อมูลงานอดิเรกของประธานาธิบดีแต่ละท่าน

4.1.1 คำสั่งต่างๆ ที่ใช้ในการสร้างฐานข้อมูล Presdb ในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์

4.1.1.1 คำสั่งทั้งหมดที่ใช้ในการสร้างชนิดข้อมูลออบเจกต์ (Object Type)

```
CREATE OR REPLACE TYPE state_type AS OBJECT(
    state_name      VARCHAR(20),
    admin_entered   VARCHAR(5),
    year_entered    NUMBER
);
```

รูปที่ 4.2 คำสั่งการสร้าง state_type

```
CREATE OR REPLACE TYPE spouse_type AS OBJECT(
    spouse_name     VARCHAR(30),
    pr_age          NUMBER,
    sp_age          NUMBER,
    nr_children     NUMBER,
    mar_year        NUMBER
);

CREATE OR REPLACE TYPE marriage_type AS TABLE OF spouse_type;
```

รูปที่ 4.3 คำสั่งการสร้าง spouse_type และ marriage_type ที่เป็นเนสต์เตดเทเบิล (Nested Table)

```
CREATE OR REPLACE TYPE hobby_type AS TABLE OF VARCHAR(20);
```

รูปที่ 4.4 คำสั่งการสร้าง Hobby Type ที่เป็นเนสต์เตดเทเบิล (Nested Table)

```
CREATE OR REPLACE TYPE admin_pr_vp_type AS OBJECT(
    admin_nr        VARCHAR(5),
    pres_name       REF president_type,
    vice_pres_name  VARCHAR(30)
);
```

รูปที่ 4.5 คำสั่งการสร้าง admin_pr_vp_type

```
CREATE OR REPLACE TYPE administration_type AS OBJECT(
    admin_nr        VARCHAR(5),
    pres_name       REF president_type,
    year_inaugurate NUMBER
);
```

รูปที่ 4.6 คำสั่งการสร้าง administration_type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE OR REPLACE TYPE president_type AS OBJECT(
  pres_name      VARCHAR(30),
  birth_yr       NUMBER,
  yrs_serv       NUMBER,
  death_age      NUMBER,
  party          VARCHAR(20),
  state_born     REF state_type,
  hobby          hobby_type,
  pres_marriage  marriage_type,
  ORDER MEMBER FUNCTION order_pres_name(p president_type)
    RETURN INTEGER,
  MEMBER FUNCTION countChildren RETURN NUMBER,
  MEMBER FUNCTION countSpouse RETURN NUMBER
);

```

รูปที่ 4.7 คำสั่งการสร้าง president_type

```

CREATE OR REPLACE TYPE BODY president_type AS
  ORDER MEMBER FUNCTION order_pres_name (p president_type)
  RETURN INTEGER IS
  BEGIN
    IF pres_name < p.pres_name THEN
      RETURN -1;
    ELSIF pres_name > p.pres_name THEN
      RETURN 1;
    ELSE
      RETURN 0;
    END IF;
  END;

  MEMBER FUNCTION countChildren RETURN NUMBER IS
    v_numChildren NUMBER;
  BEGIN
    SELECT sum(mar.nr_children) INTO v_numChildren
    FROM president_table p , TABLE(p.PRES_MARRIAGE) mar
    WHERE p.pres_name = SELF.pres_name;
    RETURN v_numChildren;
  END;

  MEMBER FUNCTION countSpouse RETURN NUMBER IS
    v_numSpouse NUMBER;
  BEGIN
    SELECT count(mar.spouse_name) INTO v_numSpouse
    FROM president_table p , TABLE(p.pres_marriage) mar
    WHERE p.pres_name = SELF.pres_name;
    RETURN v_numSpouse;
  END;
END;

```

รูปที่ 4.8 คำสั่งการสร้าง president_type ในส่วนของ Body

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CREATE OR REPLACE TYPE election_type AS OBJECT(
    election_year          NUMBER,
    candidate              REF president_type,
    vote                  NUMBER,
    WINNER_LOSER_INDIC    VARCHAR(1)
);

```

รูปที่ 4.9 คำสั่งการสร้าง election_type

4.1.2 คำสั่งทั้งหมดที่ใช้ในการสร้างตาราง

```

CREATE TABLE state_table OF state_type;

CREATE TABLE president_table OF president_type
NESTED TABLE hobby STORE AS hobby_table
NESTED TABLE pres_marriage STORE AS marriage_table;

CREATE TABLE administration_table OF administration_type;

CREATE TABLE admin_pr_vp_table OF admin_pr_vp_type;

CREATE TABLE election_table OF election_type;

```

รูปที่ 4.10 คำสั่งทั้งหมดที่ใช้ในการสร้างตาราง

4.1.3 ตัวอย่างคำสั่งเพิ่มข้อมูลลงตาราง

```

INSERT INTO state_table
VALUES(state_type('Massachusetts' , null , 1776));
INSERT INTO state_table
VALUES(state_type('Pennsylvania' , null , 1776));
INSERT INTO state_table
VALUES(state_type('Virginia' , null , 1776));
INSERT INTO state_table
VALUES(state_type('Vermont' , '1' , 1791));
INSERT INTO state_table
VALUES(state_type('Tennessee' , '2' , 1796));
INSERT INTO state_table
VALUES(state_type('Ohio' , '4' , 1803));
INSERT INTO state_table
VALUES(state_type('Louisiana' , '6' , 1812));
INSERT INTO state_table
VALUES(state_type('Florida' , '14' , 1845));
INSERT INTO state_table
VALUES(state_type('Colorado' , '22' , 1876));
INSERT INTO state_table
VALUES(state_type('Alaska' , '43' , 1959));

```

รูปที่ 4.11 คำสั่งการเพิ่มข้อมูลลงตาราง state_table

```

INSERT INTO president_table
SELECT president_type('Washington G' , 1732 , 7 , 67 ,
'Federalist' , REF(s) ,
hobby_type('Fishing' , 'Riding') ,
marriage_type(
spouse_type('Custis M D' , 26 , 27 , 0 , 1759)) )
FROM state_table s
WHERE s.state_name = 'Virginia';

INSERT INTO president_table
SELECT president_type('Adams J' , 1735 , 4 , 90 ,
'Federalist' , REF(s) ,
null ,
marriage_type(
spouse_type('Smith A' , 28 , 19 , 5 , 1764)) )
FROM state_table s
WHERE s.state_name = 'Massachusetts';

```

รูปที่ 4.12 คำสั่งการเพิ่มข้อมูลลงตาราง president_table

```

INSERT INTO admin_pr_vp_table
  SELECT  admin_pr_vp_type('1' , REF(p) , 'Adams J')
  FROM    president_table p
  WHERE   p.pres_name = 'Washington G';

INSERT INTO admin_pr_vp_table
  SELECT  admin_pr_vp_type('2' , REF(p) , 'Adams J')
  FROM    president_table p
  WHERE   p.pres_name = 'Washington G';

INSERT INTO admin_pr_vp_table
  SELECT  admin_pr_vp_type('3' , REF(p) , 'Jefferson T')
  FROM    president_table p
  WHERE   p.pres_name = 'Adams J';

```

รูปที่ 4.13 คำสั่งการเพิ่มข้อมูลลงตาราง admin_pr_vp_table

```

INSERT INTO administration_table
  SELECT  administration_type('1' , REF(t) , 1789)
  FROM    president_table t
  WHERE   t.pres_name = 'Washington G';

INSERT INTO administration_table
  SELECT  administration_type('2' , REF(t) , 1839)
  FROM    president_table t
  WHERE   t.pres_name = 'Washington G';

INSERT INTO administration_table
  SELECT  administration_type('3' , REF(t) , 1797)
  FROM    president_table t
  WHERE   t.pres_name = 'Adams J';

```

รูปที่ 4.14 คำสั่งการเพิ่มข้อมูลลงตาราง administration_table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 ตัวอย่างการคิวรีข้อมูล

ตัวอย่าง 1 ให้แสดงชื่อประธานาธิบดี และเมืองที่เกิด

```
SELECT p.pres_name, p.state_born.state_name
FROM president_table p
```

รูปที่ 4.15 คำสั่งการคิวรี

PRES_NAME	STATE_BORN.STATE_NAME
Washington G	Virginia
Adams J	Massachusetts
Jefferson T	Virginia
Madison J	Virginia
Adams J Q	Massachusetts
Buchanan J	Pennsylvania
Hayes R B	Ohio
Arthur C A	Vermont
Harrison B	Ohio
McKinley W	Ohio

รูปที่ 4.16 ผลลัพธ์จากการคิวรี

ตัวอย่าง 2 ให้แสดงชื่อประธานาธิบดี ชื่อภรรยา จำนวนบุตร และจำนวนภรรยาทั้งหมด

```
select p.pres_name, mar.spouse_name, p.countChildren(), p.countSpouse()
from president_table p , TABLE(p.PRES_MARRIAGE) mar
order by value(p) DESC
```

รูปที่ 4.17 คำสั่งการคิวรี

PRES_NAME	SPOUSE_NAME	P.COUNTCHILDREN()	P.COUNTSPOUSE()
Washington G	Custis M D	0	1
McKinley W	Sexton I	2	1
Madison J	Todd D D P	3	1
Jefferson T	Skehon M W	6	1
Hayes R B	Webb L W	8	1
Harrison B	Scott C L	3	2
Harrison B	Dimmick M S L	3	2
Arthur C A	Herndon E L	3	1
Adams J Q	Johnson L C	4	1
Adams J	Smith A	5	1

รูปที่ 4.18 ผลลัพธ์จากการคิวรี

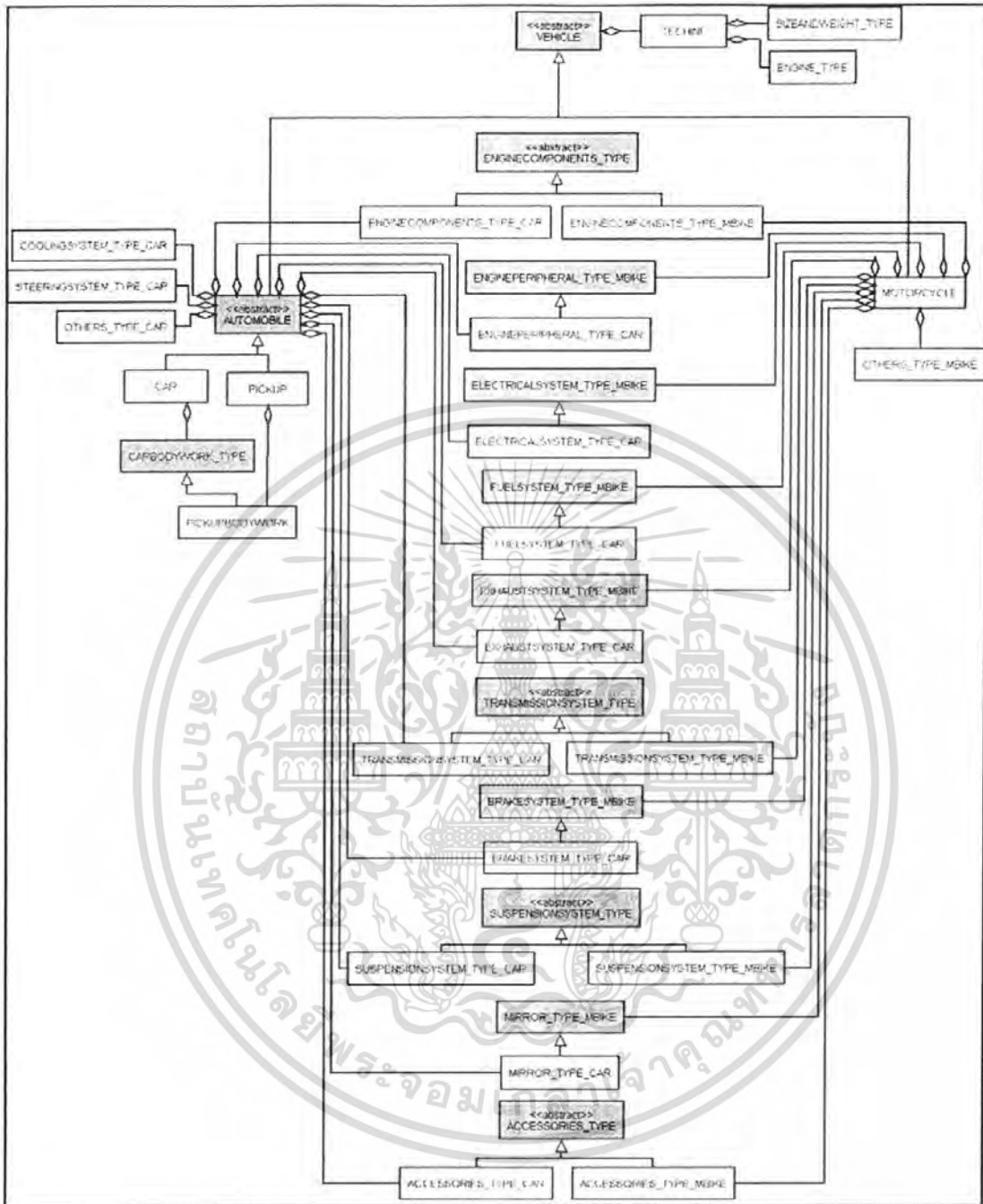
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำมาใช้

4.2 การออกแบบฐานข้อมูลรถในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์

ฐานข้อมูลรถ เก็บข้อมูลเกี่ยวกับอะไหล่ของรถ 3 ประเภทคือ รถยนต์ รถกระบะ และรถจักรยานยนต์ โดยที่อะไหล่ของรถทั้งสามประเภทจำแนกออกเป็นกลุ่มต่างๆ ได้ตามหน้าที่การทำงานได้ 14 กลุ่มดังนี้คือ

1. ชิ้นส่วนเครื่องยนต์ (Engine Components)
2. อุปกรณ์ส่วนควบคุมของเครื่องยนต์ (Engine Peripheral Components)
3. ระบบไฟฟ้า (Electrical System)
4. ระบบน้ำมันเชื้อเพลิง (Fuel System)
5. ระบบหล่อเย็น (Cooling System)
6. ระบบไอเสีย (Exhaust System)
7. ระบบส่งกำลัง (Transmission System)
8. ระบบเบรก (Brake System)
9. ระบบกันสะเทือน (Suspension System)
10. ระบบบังคับเลี้ยว (Steering System)
11. ชิ้นส่วนตัวถัง (Bodywork)
12. ชุดกระจก (Mirror)
13. ชิ้นส่วนภายในและอุปกรณ์ตกแต่ง (Accessories)
14. อุปกรณ์อื่นๆ (Others)

แต่ละกลุ่มจะประกอบไปด้วยชิ้นส่วนอะไหล่ย่อยๆ ลงไปอีก ซึ่งข้อมูลชิ้นส่วนอะไหล่รถที่ใช้ในโครงการนี้ ในส่วนของราคาอะไหล่แต่ละชิ้นและข้อมูลทางเทคนิคของรถ ได้อ้างอิงข้อมูลโดยประมาณจากเว็บไซต์ของบริษัทโตโยต้า (Toyota) บริษัทเอพิฮอนดา (Aphonda) และออโต้แอ็ดวัน (Autoatone)

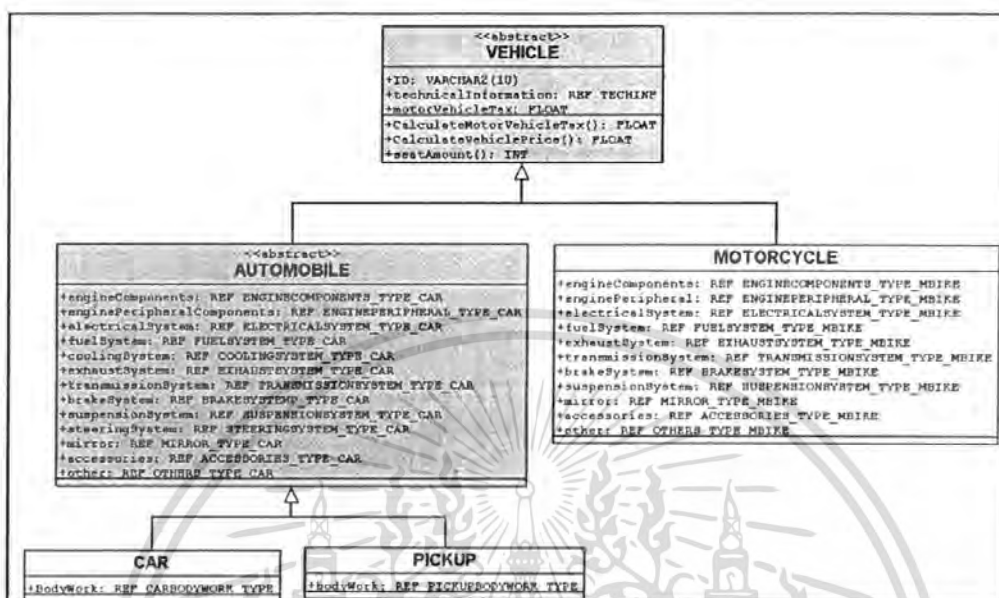


รูปที่ 4.19 โครงสร้างโดยรวมของฐานข้อมูลรถในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 รายละเอียดของฐานข้อมูลรถ

4.2.1.1 ฐานข้อมูลรถ



รูปที่ 4.20 ฐานข้อมูลรถ

รูปที่ 4.20 แสดงโครงสร้างโดยรวมอย่างคร่าวๆ ของฐานข้อมูล ประกอบไปด้วย VEHICLE ซึ่งเป็นชนิดข้อมูลแอ็บสแตร็ก (abstract) และมี AUTOMOBILE และ MOTORCYCLE เป็นชนิดข้อมูลที่สืบทอดมาจากชนิดข้อมูล VEHICLE ขณะเดียวกัน AUTOMOBILE ก็สร้างให้เป็นชนิดข้อมูลแอ็บสแตร็ก โดยมีชนิดข้อมูล CAR และ PICKUP ที่สืบทอดมาจาก AUTOMOBILE อีกทีหนึ่ง

สำหรับในชนิดข้อมูล VEHICLE ได้ทำการประกาศเมธอดขึ้นมา 3 ตัวคือ เมธอดที่ใช้ในการคำนวณภาษีรถ, เมธอดที่ใช้ในการคำนวณราคาคราด และเมธอดที่ใช้คำนวณหาจำนวนที่นั่งภายในรถ ซึ่งเมธอดทั้งสามตัวนี้ได้ประกาศเป็นแอ็บสแตร็กเมธอด ดังนั้นจึงต้องมีการอิมพลีเมนต์ (implement) ใน CAR, PICKUP และ MOTORCYCLE

ตาราง 4.1 แสดงรายละเอียดของชนิดข้อมูล VEHICLE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
technicalInformation	REF TECHINF	TECHINF เป็นชนิดข้อมูลที่เก็บรายละเอียดทางเทคนิคของรถ
motorVehicleTax	FLOAT	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.2 แสดงรายละเอียดของชนิดข้อมูล MOTORCYCLE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
EngineComponents	REF ENGINECOMPONENTS_TYPE_MBIKE	ENGINECOMPONENTS_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนเครื่องยนต์ของรถจักรยานยนต์
enginePeripheral	REF ENGINEPERIPHERAL_TYPE_MBIKE	ENGINEPERIPHERAL_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดอุปกรณ์ส่วนควบคุมของเครื่องยนต์ของรถจักรยานยนต์
electricalSystem	REF ELECTRICALSYSTEM_TYPE_MBIKE	ELECTRICALSYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบไฟฟ้าของรถจักรยานยนต์
fuelSystem	REF FUELSYSTEM_TYPE_MBIKE	FUELSYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบน้ำมันเชื้อเพลิงของรถจักรยานยนต์
exhaustSystem	REF EXHAUSTSYSTEM_TYPE_MBIKE	EXHAUSTSYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบไอเสียของรถจักรยานยนต์
TransmissionSystem	REF TRANSMISSIONSYSTEM_TYPE_MBIKE	TRANSMISSIONSYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบส่งกำลังของรถจักรยานยนต์
brakeSystem	REF BRAKESYSTEM_TYPE_MBIKE	BRAKESYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบเบรกของรถจักรยานยนต์
SuspensionSystem	REF SUSPENSIONSYSTEM_TYPE_MBIKE	SUSPENSIONSYSTEM_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบกันสะเทือนของรถจักรยานยนต์
mirror	REF MIRROR_TYPE_MBIKE	MIRROR_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดชุดกระจกของรถจักรยานยนต์
accessories	REF ACCESSORIES_TYPE_MBIKE	ACCESSORIES_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนภายในและอุปกรณ์ตกแต่งของรถจักรยานยนต์
other	REF OTHERS_TYPE_MBIKE	OTHERS_TYPE_MBIKE เป็นชนิดข้อมูลที่เก็บรายละเอียดอุปกรณ์อื่นๆ ของรถจักรยานยนต์

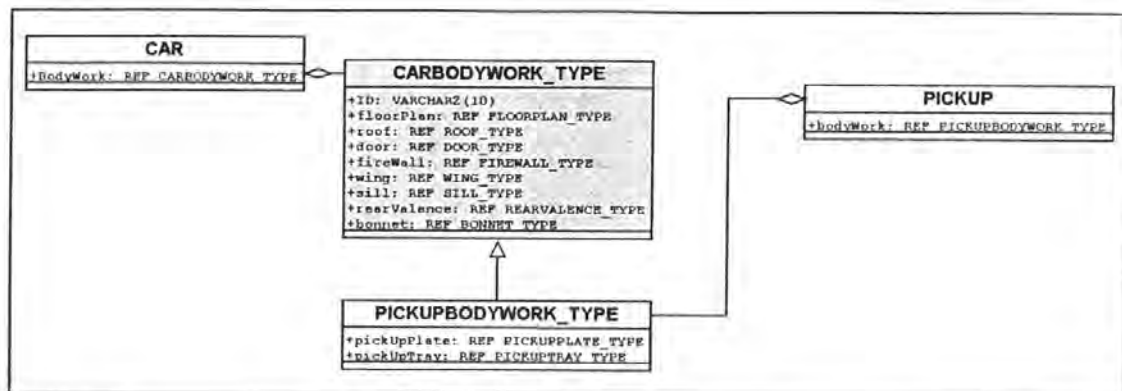
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.3 แสดงรายละเอียดของชนิดข้อมูล AUTOMOBILE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
EngineComponents	REF ENGINECOMPONENTS_TYPE_CAR	ENGINECOMPONENTS_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนเครื่องยนต์ของรถยนต์
enginePeripheral	REF ENGINEPERIPHERAL_TYPE_CAR	ENGINEPERIPHERAL_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดอุปกรณ์ส่วนควบคุมของเครื่องยนต์ของรถยนต์
electricalSystem	REF ELECTRICALSYSTEM_TYPE_CAR	ELECTRICALSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบไฟฟ้าของรถยนต์
fuelSystem	REF FUELSYSTEM_TYPE_CAR	FUELSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบน้ำมันเชื้อเพลิงของรถยนต์
coolingSystem	REF COOLINGSYSTEM_TYPE_CAR	EXHAUSTSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบหล่อเย็นของรถยนต์
exhaustSystem	REF EXHAUSTSYSTEM_TYPE_CAR	EXHAUSTSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบไอเสียของรถยนต์
TransmissionSystem	REF TRANSMISSIONSYSTEM_TYPE_CAR	TRANSMISSIONSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบส่งกำลังของรถยนต์
brakeSystem	REF BRAKESYSTEM_TYPE_CAR	BRAKESYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบเบรกของรถยนต์
SuspensionSystem	REF SUSPENSIONSYSTEM_TYPE_CAR	SUSPENSIONSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบกันสะเทือนของรถยนต์
steeringSystem	REF STEERINGSYSTEM_TYPE_CAR	EXHAUSTSYSTEM_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดระบบบังคับเลี้ยวของรถยนต์
mirror	REF MIRROR_TYPE_CAR	MIRROR_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดชุดกระจกของรถยนต์
accessories	REF ACCESSORIES_TYPE_CAR	ACCESSORIES_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนภายในและอุปกรณ์ตกแต่งของรถยนต์
other	REF OTHERS_TYPE_CAR	OTHERS_TYPE_CAR เป็นชนิดข้อมูลที่เก็บรายละเอียดอุปกรณ์อื่นๆ ของรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำข้อมูลไปเผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 แสดงความสัมพันธ์ระหว่าง CARBODYWORK_TYPE กับ CAR และ PICKUP

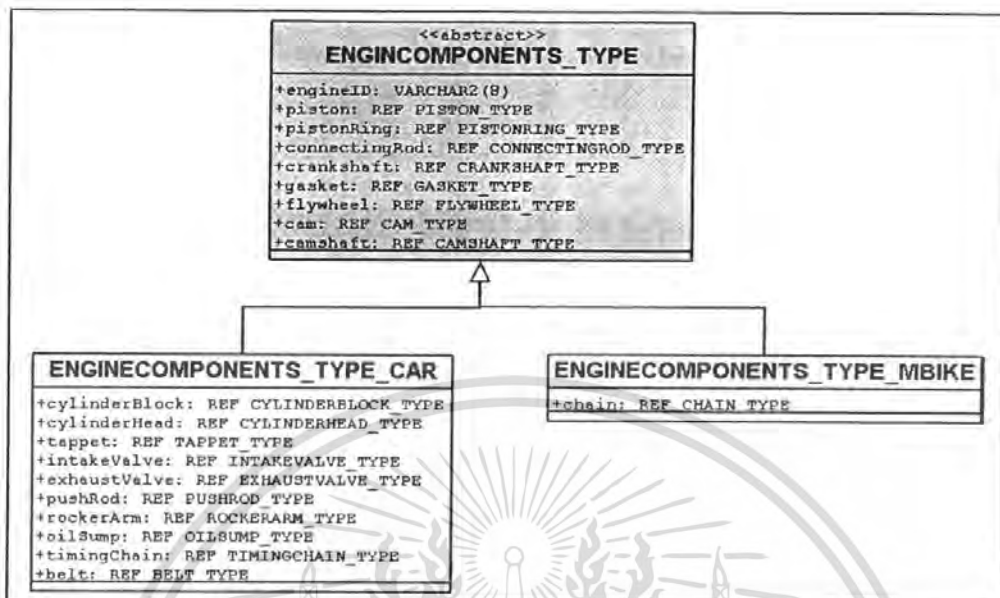
ตาราง 4.4 แสดงรายละเอียดของชนิดข้อมูล CARBODYWORK_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	
floorPlan	REF FLOORPLAN_TYPE	FLOORPLAN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดพื้นของรถยนต์
roof	REF ROOF_TYPE	ROOF_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหลังคาของรถยนต์
door	REF DOOR_TYPE	DOOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดประตูของรถยนต์
firewall	REF FIREWALL_TYPE	FIREWALL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดห้องเครื่องยนต์ของรถยนต์
wing	REF WING_TYPE	WING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงตัวถังหน้าของรถยนต์
sill	REF SILL_TYPE	SILL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงชายล่างด้านข้างของรถยนต์
rearValence	REF REARVALENCE_TYPE	REARVALENCE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงชายล่างหลังของรถยนต์
bonnet	REF BONNET_TYPE	BONNET_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดฝากระโปรงของรถยนต์

ตาราง 4.5 แสดงรายละเอียดของชนิดข้อมูล PICKUPBODYWORK_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
pickupPlate	REF PICKUPPLATE_TYPE	PICKUPPLATE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียด ของรถกระบะ
pickupTray	REF PICKUPTRAY_TYPE	PICKUPTRAY_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียด ของรถกระบะ

4.2.1.2 ชั้นส่วนเครื่องยนต์ (Engine Components)



รูปที่ 4.22 ชั้นส่วนเครื่องยนต์ (Engine Components)

รูปที่ 4.22 แสดงชนิดข้อมูล ENGINECOMPONENTS_TYPE ซึ่งเป็นชนิดข้อมูลที่เก็บชั้นส่วนเครื่องยนต์ของรถทั้งหมด โดยชนิดข้อมูล ENGINECOMPONENTS_TYPE_CAR และชนิดข้อมูล ENGINECOMPONENTS_TYPE_MBIKE เป็นชนิดข้อมูลที่ทำการสืบทอดมาจากชนิดข้อมูล ENGINECOMPONENTS_TYPE

ตาราง 4.6 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
engineID	VARCHAR2(8)	-
piston	REF PISTON_TYPE	PISTON_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดลูกสูบของรถ
pistonRing	REF PISTONRING_TYPE	PISTONRING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแหวนลูกสูบของรถ
connectingRod	REF CONNECTINGROD_TYPE	CONNECTINGROD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดก้านสูบของรถ
crankshaft	REF CRANKSHAFT_TYPE	CRANKSHAFT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเพลาค้อเหวี่ยงของรถ
gasket	REF GASKET_TYPE	GASKET_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดปะเก็นของรถ
flywheel	REF FLYWHEEL_TYPE	FLYWHEEL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดฟลายวีลของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่โดยไม่ขออนุญาตจากศูนย์ฯ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.6 (ต่อ)

cam	REF CAM_TYPE	CAM_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคู่มือของรถ
camshaft	REF CAMSHAFT_TYPE	CAMSHAFT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเพลาลูกเบี้ยวของรถ

ตาราง 4.7 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE_CAR

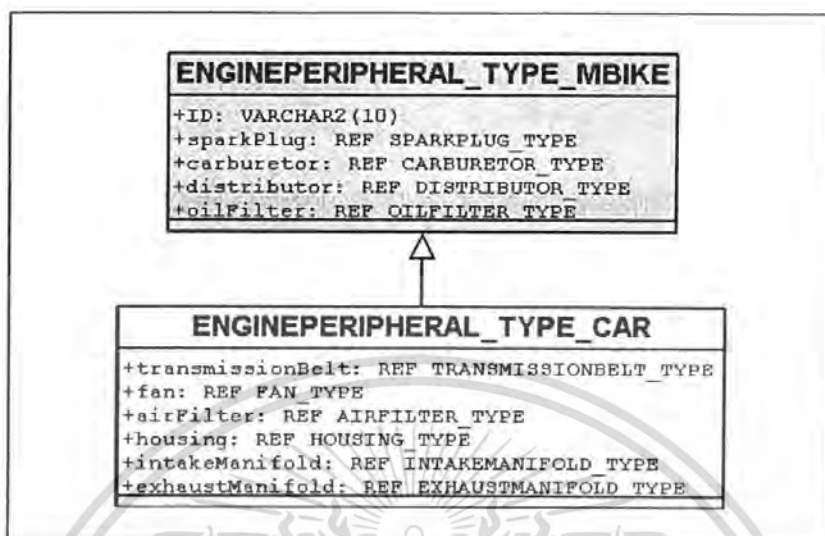
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
cylinderBlock	REF CYLINDERBLOCK_TYPE	CYLINDERBLOCK_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเสื้อสูบรถยนต์
cylinderHead	REF CYLINDERHEAD_TYPE	CYLINDERHEAD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดฝาสูบรถยนต์
tappet	REF TAPPET_TYPE	TAPPET_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดข้อเหวี่ยงรถยนต์
intakeValve	REF INTAKEVALVE_TYPE	INTAKEVALVE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดลิ้นไอดีรถยนต์
exhaustValve	REF EXHAUSTVALVE_TYPE	EXHAUSTVALVE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไอเสียรถยนต์
pushrod	REF PUSHROD_TYPE	PUSHROD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดก้านกระทุ้งลิ้นรถยนต์
rockerArm	REF ROCKERARM_TYPE	ROCKERARM_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดชูกดลิ้นรถยนต์
oilSump	REF OILSUMP_TYPE	OILSUMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดอ่างน้ำมันเครื่องรถ
timingChain	REF TIMINGCHAIN_TYPE	TIMINGCHAIN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดโซ่รถยนต์
belt	REF BELT_TYPE	BELT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายพานขับเคลื่อน

ตาราง 4.8 แสดงรายละเอียดของชนิดข้อมูล ENGINECOMPONENTS_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
chain	REF CHAIN_TYPE	CHAIN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดโซ่รถจักรยานยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.3 อุปกรณ์ส่วนควบคุมของเครื่องยนต์ (Engine Peripheral Components)



รูปที่ 4.23 อุปกรณ์ส่วนควบคุมของเครื่องยนต์ (Engine Peripheral Components)

รูปที่ 4.23 แสดงชนิดข้อมูล ENGINEPERIPHERALCOMPONENTS_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เก็บอุปกรณ์ส่วนควบคุมของเครื่องยนต์ ซึ่งชนิดข้อมูลนี้ถูกสืบทอดโดยชนิดข้อมูล ENGINEPERIPHERALCOMPONENTS_TYPE_CAR

ตาราง 4.9 แสดงรายละเอียดของชนิดข้อมูล ENGINEPERIPHERAL_TYPE_MBIKE

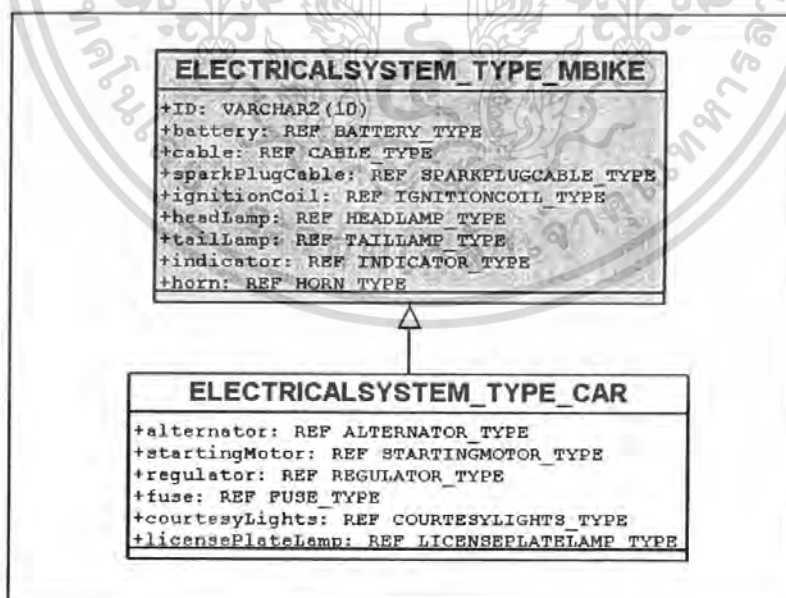
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
sparkplug	REF SPARKPLUG_TYPE	SPARKPLUG_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหัวเทียนรถจักรยานยนต์
carburetor	REF CARBURETOR_TYPE	CARBURETOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหัวฉีดน้ำมันเชื้อเพลิงรถจักรยานยนต์
distributor	REF DISTRIBUTOR_TYPE	DISTRIBUTOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดจานจ่ายรถจักรยานยนต์
oilFilter	REF OILFILTER_TYPE	OILFILTER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไส้กรองน้ำมันเครื่องรถจักรยานยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.10 แสดงรายละเอียดของชนิดข้อมูล ENGINEPERIPHERAL_TYPE_CAR

แอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
transmissionBelt	REF TRANSMISSIONBELT_TYPE	TRANSMISSIONBELT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายพานรถยนต์
fan	REF FAN_TYPE	FAN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดพัดลมรถยนต์
airFilter	REF AIRFILTER_TYPE	AIRFILTER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไส้กรองอากาศรถยนต์
housing	REF HOUSING_TYPE	HOUSING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหม้อกรองอากาศรถยนต์
intakeManifold	REF INTAKEMANIFOLD_TYPE	INTAKEMANIFOLD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อร่วมไอดีรถยนต์
exhaustManifold	REF EXHAUSTMANIFOLD_TYPE	EXHAUSTMANIFOLD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อร่วมไอเสียรถยนต์

4.2.1.4 ระบบไฟฟ้า (Electrical System)



รูปที่ 4.24 ระบบไฟฟ้า (Electrical System)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.24 แสดงชนิดข้อมูล ELECTIRCALSYSTEM_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เก็บระบบไฟฟ้าของรถจักรยานยนต์ มีชนิดข้อมูล ELECTRICALSYSTEM_TYPE_CAR ทำการสืบทอดมา

ตาราง 4.11 แสดงรายละเอียดของชนิดข้อมูล ELECTRICALSYSTEM_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
battery	REF BATTERY_TYPE	BATTERY_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแบตเตอรี่รถจักรยานยนต์
cable	REF CABLE_TYPE	CABLE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายไฟรถจักรยานยนต์
sparkPlugCable	REF SPARKPLUG_TYPE	SPARKPLUG_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายหัวเทียนรถจักรยานยนต์
ignitionCoil	REF IGNITIONCOIL_TYPE	IGNITIONCOIL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคอยล์จุดระเบิดรถจักรยานยนต์
headLamp	REF HEADLAMP_TYPE	HEADLAMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไฟหน้ารถจักรยานยนต์
tailLamp	REF TAILLAMP_TYPE	TAILLAMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไฟท้ายรถจักรยานยนต์
indicator	REF INDICATOR_TYPE	INDICATOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไฟเลี้ยวรถจักรยานยนต์
horn	REF HORN_TYPE	HORN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแตรรถจักรยานยนต์

ตาราง 4.12 แสดงรายละเอียดของชนิดข้อมูล ELECTRICALSYSTEM_TYPE_CAR

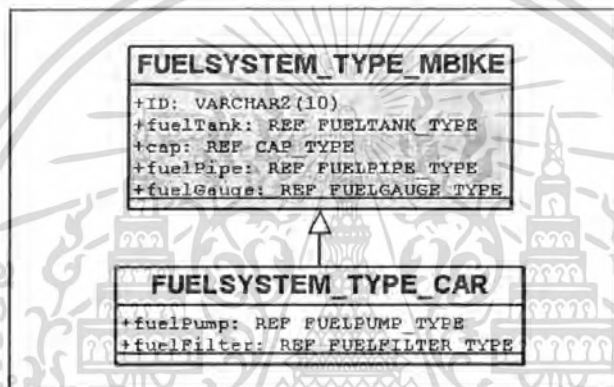
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
alternator	REF ALTERNATOR_TYPE	ALTERNATOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดโซ่รถยนต์
startingMotor	REF STARTINGMOTOR_TYPE	STARTINGMOTOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดมอเตอร์สตาร์ทรถยนต์
regulator	REF REGULATOR_TYPE	REGULATOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเรกูเลเตอร์รถยนต์
Fuse	REF FUSE_TYPE	FUSE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดชุดฟิวส์รถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.12 (ต่อ)

courtesyLights	REF COURTESYLIGHTS_TYPE	COURTESYLIGHTS_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไฟส่องสว่างภายในรถยนต์
licensePlateLamp	REF LICENSEPLATE_TYPE	LICENSEPLATE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไฟส่องป้ายรถยนต์

4.2.1.5 ระบบน้ำมันเชื้อเพลิง (Fuel System)



รูปที่ 4.25 ระบบน้ำมันเชื้อเพลิง (Fuel System)

รูปที่ 4.25 แสดงชนิดข้อมูล FUELSYSTEM_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบน้ำมันเชื้อเพลิง โดยมีชนิดข้อมูล FUELSYSTEM_TYPE_CAR ทำการสืบทอด

ตาราง 4.13 แสดงรายละเอียดของชนิดข้อมูล FUELSYSTEM_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
fuelTank	REF FUELTANK_TYPE	FUELTANK_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดถังน้ำมันรถจักรยานยนต์
cap	REF CAP_TYPE	CAP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดฝาปิดรถจักรยานยนต์
fuelPipe	REF FUELPIPE_TYPE	FUELPIPE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อน้ำมันรถจักรยานยนต์
fuelGauge	REF FUELGAUGE_TYPE	FUELGAUGE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเครื่องวัดระดับน้ำมันเชื้อเพลิงรถจักรยานยนต์

ตาราง 4.14 แสดงรายละเอียดของชนิดข้อมูล FUELSYSTEM_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
fuelPump	REF FUELPUMP_TYPE	FUELPUMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดปั๊มน้ำมันรถยนต์
fuelFilter	REF FUELFILTER_TYPE	FUELFILTER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดไส้กรองน้ำมันเชื้อเพลิงรถยนต์

4.2.1.6 ระบบหล่อเย็น (Cooling System)

COOLINGSYSTEM_TYPE_CAR
+ID: VARCHAR2(10)
+radiator: RADIATOR_TYPE
+oilCooler: OILCOOLER_TYPE
+waterPump: WATERPUMP_TYPE
+oilHose: OILHOSE_TYPE
+thermostat: THERMOSTAT_TYPE
+waterReservoir: WATERRESERVOIR_TYPE
+radiatorCap: RADIATORCAP_TYPE

รูปที่ 4.26 ระบบหล่อเย็น (Cooling System)

รูปที่ 4.26 แสดงชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR ซึ่งเป็นชนิดข้อมูลที่เก็บชิ้นส่วนระบบหล่อเย็นของรถยนต์ทั้งหมด

ตาราง 4.15 แสดงรายละเอียดของชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR

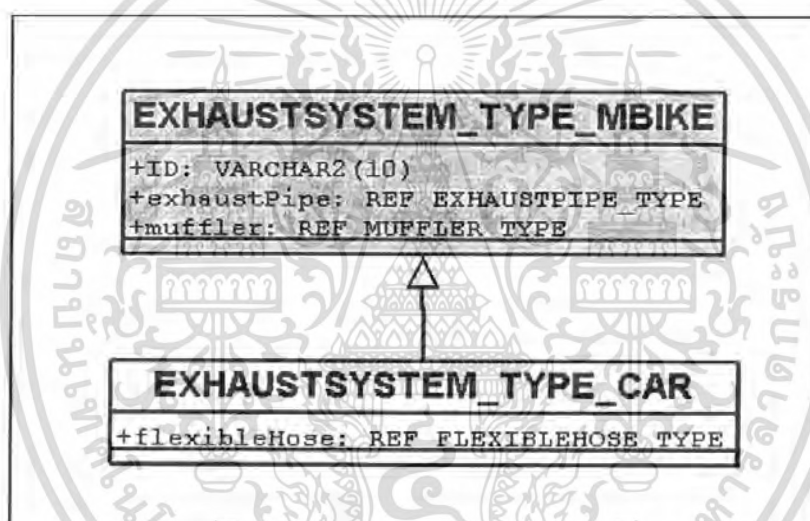
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
radiator	REF RADIATOR_TYPE	RADIATOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหม้อน้ำรด
oilCooler	REF OILCOOLER_TYPE	OILCOOLER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคอยล์คูลเลอร์รด
waterPump	REF WATERPUMP_TYPE	WATERPUMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดปั๊มน้ำรด
oilHose	REF OILHOSE_TYPE	OILHOSE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อน้ำมันรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.15 (ต่อ)

thermostat	REF THERMOSTAT_TYPE	THERMOSTAT_TYPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดเทอร์โมสแตทรถ
waterReservoir	REF WATERRESERVOIR_TYPE	WATERRESERVOIR_TYPE เป็นชนิด ข้อมูลที่เก็บรายละเอียดถังพักน้ำสำรอง รถ
radiatorCap	REF RADIATORCAP_TYPE	RADIATORCAP_TYPE เป็นชนิด ข้อมูลที่เก็บรายละเอียดฝาปิดหม้อน้ำรถ

4.2.1.7 ระบบไอเสีย (Exhaust System)



รูปที่ 4.27 ระบบ ไอเสีย (Exhaust System)

รูปที่ 4.27 แสดงชนิดข้อมูล EXHAUSTSYSTEM_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบไอเสีย โดยมีชนิดข้อมูล EXHAUSTSYSTEM_TYPE_CAR ทำการสืบทอด

ตาราง 4.16 แสดงรายละเอียดของชนิดข้อมูล EXHAUSTSYSTEM_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
exhaustPipe	REF EXHAUSTPIPE_TYPE	EXHAUSTPIPE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อไอเสียรถ
muffler	REF MUFFLER_TYPE	MUFFLER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหม้อเก็บเสียงรถ

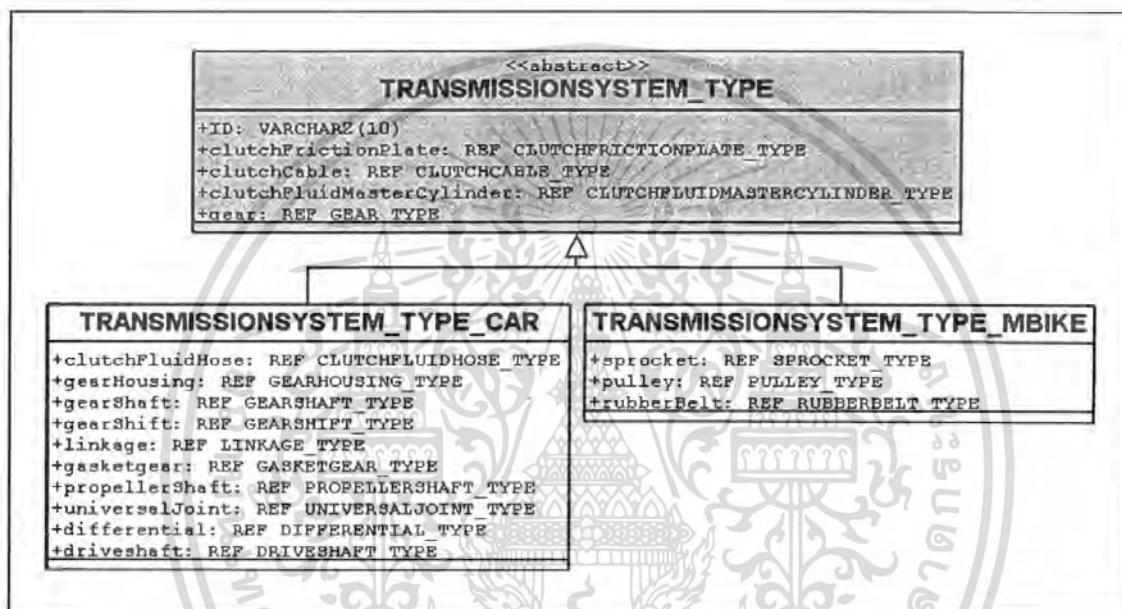
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่โดยไม่ได้รับอนุญาตจากหน่วยงานที่เกี่ยวข้อง

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.17 แสดงรายละเอียดของชนิดข้อมูล EXHAUSTSYSTEM_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
	REF FLEXIBLEHOSE_TYPE	FLEXIBLEHOSE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่ออ่อนเหล็กดัดรถ

4.2.1.8 ระบบส่งกำลัง (Transmission System)



รูปที่ 4.28 ระบบส่งกำลัง (Transmission System)

รูปที่ 4.28 แสดงชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบส่งกำลัง โดยมีชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE_CAR และชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE_MBIKE ทำการสืบทอด

ตาราง 4.18 แสดงรายละเอียดของชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
Id	VARCHAR2(10)	-
clutchFrictionPlate	REF CLUTCHFRICTIONPLATE_TYPE	CLUTCHFRICTIONPLATE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผ่นคลัทช์ชุด
clutchCable	REF CLUTCHCABLE_TYPE	CLUTCHCABLE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายคลัทช์ชุด

ตาราง 4.18 (ต่อ)

clutchFluidMasterCylinder	REF CLUTCHFLUIDMASTERCYLINDER_ TYPE	CLUTCHFLUIDMASTERCYLINDER_T YPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดแม่ ปั้มน้ำมัน คลัทช์รถ
gear	REF GEAR_TYPE	GEAR_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดเฟืองเกียร์รถ

ตาราง 4.19 แสดงรายละเอียดของชนิดข้อมูล TRANSMISSIONSYSTEM_TYPE_CAR

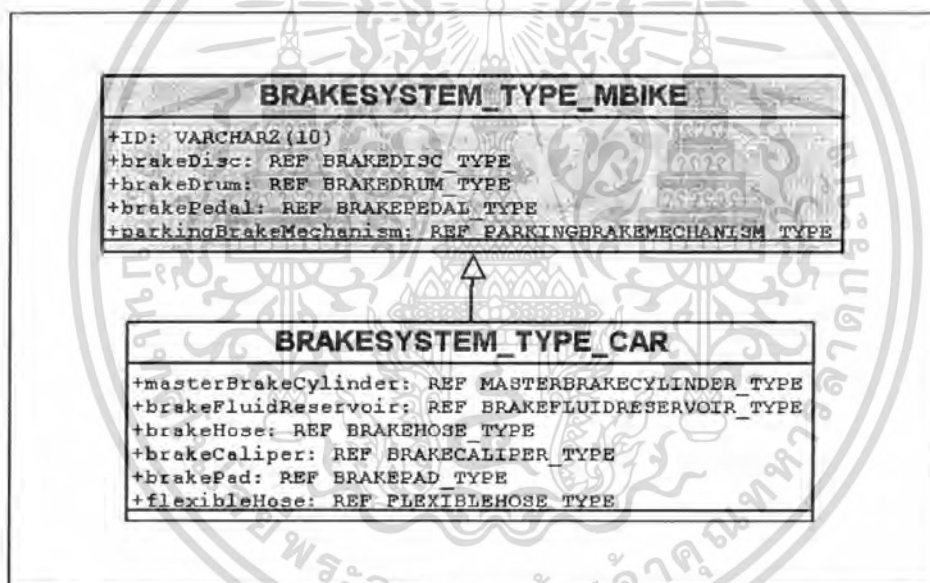
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
clutchFluidHose	REF CLUTCHFLUIDHOSE_TYPE	CLUTCHFLUIDHOSE_TYPE เป็นชนิด ข้อมูลที่เก็บรายละเอียดท่อน้ำมันคลัทช์ รถยนต์
gearHousing	REF GEARHOUSING_TYPE	GEARHOUSING_TYPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดเสื้อเกียร์รถยนต์
gearShaft	REF GEARSHAFT_TYPE	GEARSHAFT_TYPE เป็นชนิดข้อมูลที่ เก็บรายละเอียดเพลาเกียร์รถยนต์
gearShift	REF GEARSHIFT_TYPE	GEARSHIFT_TYPE เป็นชนิดข้อมูลที่ เก็บรายละเอียดก้านเกียร์รถยนต์
linkage	REF LINKAGE_TYPE	LINKAGE_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดกลไกควบคุมรถยนต์
gasketgear	REF GASKETGEAR_TYPE	GASKETGEAR_TYPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดปะเก็นชุดเกียร์รถยนต์
propellerShaft	REF PROPELLERSHAFT_TYPE	PROPELLERSHAFT_TYPE เป็นชนิด ข้อมูลที่เก็บรายละเอียดเพลาส่งกำลัง รถยนต์
universalJoint	REF UNIVERSALJOINT_TYPE	UNIVERSALJOINT_TYPE เป็นชนิด ข้อมูลที่เก็บรายละเอียดจอยท์เพลากลาง รถยนต์
differential	REF DIFFERENTIAL_TYPE	DIFFERENTIAL_TYPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดเฟืองท้ายรถยนต์
drivershaft	REF DRIVERSHAFT_TYPE	DRIVERSHAFT_TYPE เป็นชนิดข้อมูล ที่เก็บรายละเอียดเพลาขับรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.20 แสดงรายละเอียดของชนิดข้อมูล TRANMISSIONSYSTEM_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
sprocket	REF SPROCKET_TYPE	SPROCKET_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดโซ่ฟันเฟืองรถจักรยานยนต์
pulley	REF PULLEY_TYPE	PULLEY_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดรอกรถจักรยานยนต์
rubberBelt	REF RUBBERBELT_TYPE	RUBBERBELT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสายพานรถจักรยานยนต์

4.2.1.9 ระบบเบรก (Brake System)



รูปที่ 4.29 ระบบเบรก (Brake System)

รูปที่ 4.29 แสดงชนิดข้อมูล BRAKESYSTEM_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบน้ำมันเชื้อเพลิง โดยมีชนิดข้อมูล BRAKESYSTEM_TYPE_CAR ทำการสืบทอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.21 แสดงรายละเอียดของชนิดข้อมูล BRAKESYSTEM_TYPE_MBIKE

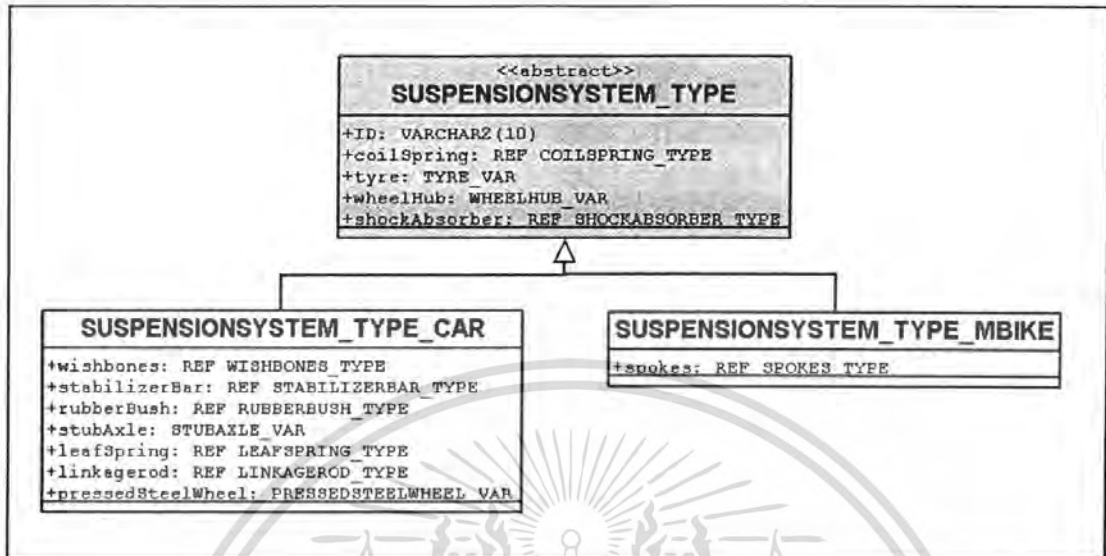
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
brakeDisc	REF BRAKEDISC_TYPE	BRAKEDISC_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดจานเบรกรด
brakeDrum	REF BRAKEDRUM_TYPE	BRAKEDRUM_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคุมเบรกรด
brakePedal	REF BRAKEPEDAL_TYPE	BRAKEPEDAL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดที่เหยียบเบรกรด
parkingBrakeMechanism	REF PARKINGBRAKEMECHANISM_TYPE	PARKINGBRAKEMECHANISM_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดชุดเบรกรมมือรด

ตาราง 4.22 แสดงรายละเอียดของชนิดข้อมูล BRAKESYSTEM_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
masterBrakeCylinder	REF MASTERBRAKECYLINDER_TYPE	MASTERBRAKECYLINDER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแม่ปั๊มเบรกรด
brakeFluidReservoir	REF BRAKEFLUIDRESERVOIR_TYPE	BRAKEFLUIDRESERVOIR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดถังเก็บน้ำมันเบรกรด
brakeHose	REF BRAKEHOSE_TYPE	BRAKEHOSE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่อเบรกรด
brakeCaliper	REF BRAKECALIPER_TYPE	BRAKECALIPER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคองห้ามล้อรด
brakePad	REF BRAKEPAD_TYPE	BRAKEPAD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดผ้าเบรกรด
flexibleHose	REF FLEXIBLEHOSE_TYPE	FLEXIBLEHOSE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดท่ออ่อนน้ำมันเบรกรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.10 ระบบกันสะเทือน (Suspension System)



รูปที่ 4.30 ระบบกันสะเทือน (Suspension System)

รูปที่ 4.30 แสดงชนิดข้อมูล SUSPENSIONSYSTEM_TYPE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบกันสะเทือน โดยมีชนิดข้อมูล SUSPENSIONSYSTEM_TYPE_CAR และ SUSPENSIONSYSTEM_TYPE_MBIKE ทำการสืบทอด

ตาราง 4.23 แสดงรายละเอียดของชนิดข้อมูล SUSPENSIONSYSTEM_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
coilSpring	REF COILSPRING_TYPE	COILSPRING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดสปริงขดรถ
tyre	TYRE_VAR	TYRE_VAR เป็นเนื้อสตั๊ดเทเบิลของ TYRE_TYPE ซึ่ง TYRE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดยางรถ
wheelHub	WHEELHUB_VAR	WHEELHUB_VAR เป็นเนื้อสตั๊ดเทเบิลของ WHEELHUB_TYPE ซึ่ง WHEELHUB_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคัมถ์รถ
shockAbsorber	REF SHOCKABSORBER_TYPE	SHOCKABSORBER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดชอคแอบซอร์บเบอร์รถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

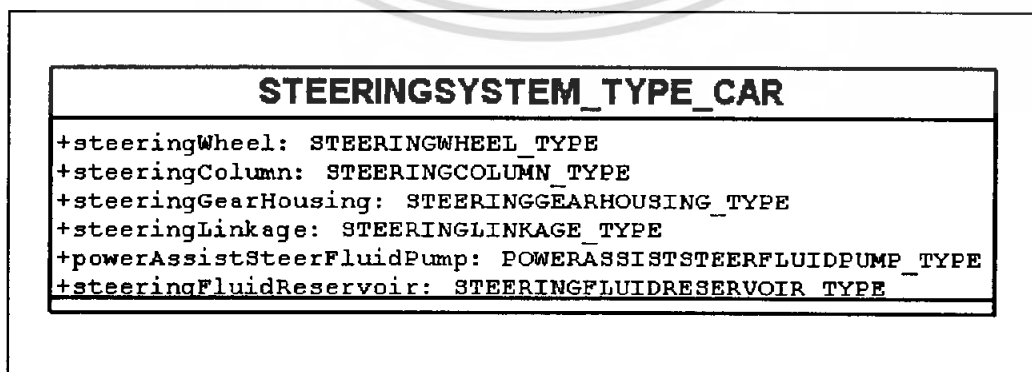
ตาราง 4.24 แสดงรายละเอียดของชนิดข้อมูล SUSPENSIONSYSTEM_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
wishbones	REF WISHBONES_TYPE	WISHBONES_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดปีกนกรถยนต์
stabilizerBar	REF STABILIZERBAR_TYPE	STABILIZERBAR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเหล็กกันโคลงรถยนต์
rubberBush	REF RUBBERBUSH_TYPE	RUBBERBUSH_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคานของรถยนต์
stubAxle	STUBAXLE_VAR	STUBAXLE_VAR เป็นชนิดข้อมูลที่เก็บรายละเอียดเพลารถยนต์
leafSpring	REF LEAFSPRING_TYPE	LEAFSPRING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแหนบรถยนต์
linkageRod	REF LINKAGEROD_TYPE	LINKAGEROD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดคานเชื่อมรถยนต์
pressedSteelWheel	PRESSEDSTEELWHEEL_VAR	PRESSEDSTEELWHEEL_VAR เป็นชนิดข้อมูลที่เก็บรายละเอียดล้อเหล็กรถยนต์

ตาราง 4.25 แสดงรายละเอียดของชนิดข้อมูล SUSPENSIONSYSTEM_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
spokes	REF SPOKES_TYPE	SPOKES_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดซี่ล้อจักรยานยนต์

4.2.1.11 ระบบบังคับเลี้ยว (Steering System)



รูปที่ 4.31 ระบบบังคับเลี้ยว (Steering System)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

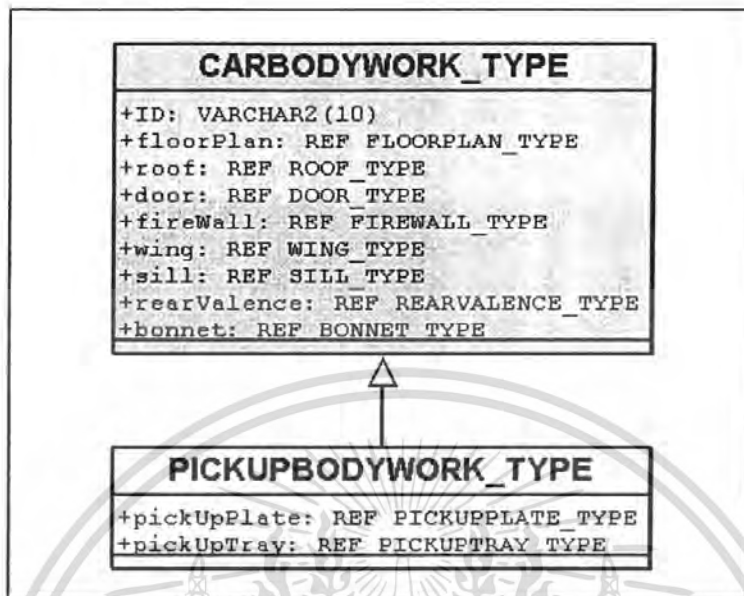
รูปที่ 4.31 แสดงชนิดข้อมูล STEERINGSYSTEM_TYPE_CAR ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดระบบบังคับเลี้ยว

ตาราง 4.26 แสดงรายละเอียดของชนิดข้อมูล STEERINGSYSTEM_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
steeringWheel	REF STEERINGWHEEL_TYPE	STEERINGWHEEL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดพวงมาลัยรถยนต์
steeringColumn	REF STEERINGCOLUMN_TYPE	STEERINGCOLUMN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดก้านพวงมาลัยรถยนต์
steeringGearHousing	REF STEERINGGEARHOUSING_TYPE	STEERINGGEARHOUSING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดชุดเฟืองบังคับเลี้ยวรถยนต์
steeringLinkage	REF STEERINGLINKAGE_TYPE	STEERINGLINKAGE_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดก้านยึดชุดบังคับเลี้ยวรถยนต์
powerAssistSteerFluidPump	REF POWERASSISTSTEERFLUIDPUMP_TYPE	POWERASSISTSTEERFLUIDPUMP_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดปั๊มน้ำมันพวงมาลัยพาวเวอร์รถยนต์
steeringFluidReservoir	REF STEERINGFLUIDRESERVOIR_TYPE	STEERINGFLUIDRESERVOIR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดถังเก็บน้ำมันพวงมาลัยพาวเวอร์รถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.12 ชิ้นส่วนตัวถัง (Bodywork)



รูปที่ 4.32 ชิ้นส่วนตัวถัง (Bodywork)

รูปที่ 4.32 แสดงชนิดข้อมูล CARBODYWORK_TYPE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนตัวถัง โดยมีชนิดข้อมูล PICKUPBODYWORK_TYPE ทำการสืบทอด

ตาราง 4.27 แสดงรายละเอียดของชนิดข้อมูล CARBODYWORK_TYPE

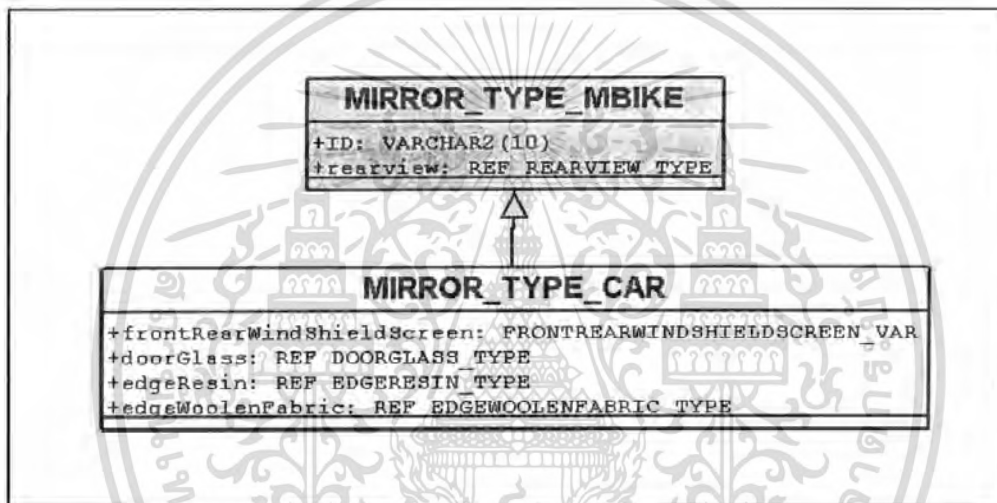
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
floorPlan	REF FLOORPLAN_TYPE	FLOORPLAN_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดพื้นรถยนต์
roof	REF ROOF_TYPE	ROOF_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดหลังคารรถยนต์
door	REF DOOR_TYPE	DOOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดประตูรถยนต์
fireWall	REF FIREWALL_TYPE	FIREWALL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดห้องเครื่องรถยนต์
wing	REF WING_TYPE	WING_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงตัวถังหน้ารถยนต์
sill	REF SILL_TYPE	SILL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงชายล่างด้านข้างรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.27 (ต่อ)

rearValence	REF REARVALENCE_TYPE	REARVALENCE_TYPE เป็นชนิดข้อมูลที่เกี่ยวข้องกับรายละเอียดแผงขายล่างหลังรถยนต์
bonnet	REF BONNET_TYPE	BONNET_TYPE เป็นชนิดข้อมูลที่เกี่ยวข้องกับรายละเอียดฝากระโปรงรถยนต์

4.2.1.13 ชุดกระจก (Mirror)



รูปที่ 4.33 ชุดกระจก (Mirror)

รูปที่ 4.33 แสดงชนิดข้อมูล MIRROR_TYPE_MBIKE ซึ่งเป็นชนิดข้อมูลที่เกี่ยวข้องกับรายละเอียดชุดกระจก โดยมีชนิดข้อมูล MIRROR_TYPE_CAR ทำการสืบทอด

ตาราง 4.28 แสดงรายละเอียดของชนิดข้อมูล MIRROR_TYPE_MBIKE

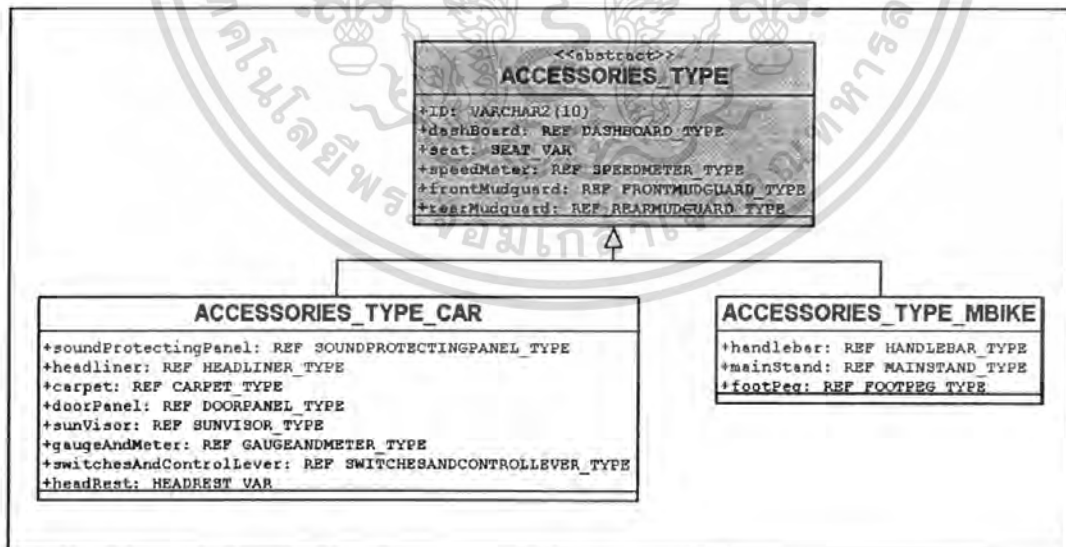
ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
rearview	REF REARVIEW_TYPE	REARVIEW_TYPE เป็นชนิดข้อมูลที่เกี่ยวข้องกับรายละเอียดกระจกมองหลังรถจักรยานยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.29 แสดงรายละเอียดของชนิดข้อมูล MIRROR_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
frontRearWindShieldScreen	FRONTREARWINDSHIELDSCREEN_ VAR	FRONTREARWINDSHIELDSCREEN_ VAR เป็นเน็ตต์เค็ดเทเบิลของ FRONTREARWINDSHIELDSCREEN_ TYPE ซึ่ง FRONTREARWINDSHIELDSCREEN_ TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียด กระจกบังลมหน้าหลังรถยนต์
doorGlass	REF DOORGLASS_TYPE	DOORGLASS_TYPE เป็นชนิดข้อมูลที่ เก็บรายละเอียดกระจกประตูรถยนต์
edgeResion	REF EDGERESIN_TYPE	EDGERESIN_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดยางขอบกระจกรถยนต์
edgeWoolenFabric	REF EDGEWOOLENFABRIC_TYPE	EDGEWOOLENFABRIC_TYPE เป็น ชนิดข้อมูลที่เก็บรายละเอียดสั๊กหลาดขอบ กระจกรถยนต์

4.2.1.14 ชิ้นส่วนภายในและอุปกรณ์ตกแต่ง (Accessories)



รูปที่ 4.34 ชิ้นส่วนภายในและอุปกรณ์ตกแต่ง (Accessories)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.34 แสดงชนิดข้อมูล ACCESSORIES_TYPE ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียดชิ้นส่วนภายในและอุปกรณ์ตกแต่ง โดยมีชนิดข้อมูล ACCESSORIES_TYPE_CAR และ ACCESSORIES_TYPE_MBIKE ทำการสืบทอด

ตาราง 4.30 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
ID	VARCHAR2(10)	-
dashboard	REF DASHBOARD_TYPE	DASHBOARD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงหน้าปัดมรด์
Seat	SEAT_VAR	SEAT_VAR เป็นเน้สเด็คเทเบิลของชนิดข้อมูล SEAT_TYPE ซึ่ง SEAT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดที่นั่งของรด์
speedMeter	REF SPEEDMETER_TYPE	SPEEDMETER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดมาตรวัดความเร็วรด์
frontMudguard	REF FRONTMUDGUARD_TYPE	FRONTMUDGUARD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดบังโคลนหน้ารด์
rearMudguard	REF REARMUDGUARD_TYPE	REARMUDGUARD_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดบังโคลนหลังรด์

ตาราง 4.31 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
soundProtectingPanel	REF SOUNDPROTECTINGPANEL_TYPE	SOUNDPROTECTINGPANEL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดวัสดุกันเสียงรด์ยนต์
headliner	REF HEADLINER_TYPE	HEADLINER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดผ้าปิดหลังคารด์ยนต์
carpet	REF CARPET_TYPE	CARPET_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดพรมปูพื้นรด์ยนต์
doorpanel	REF DOORPANEL_TYPE	DOORPANEL_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงประตูรด์ยนต์
sunvisor	REF SUNVISOR_TYPE	SUNVISOR_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดแผงบังแดดรด์ยนต์
gaugeAndMeter	REF GAUGEANDMETER_TYPE	GAUGEANDMETER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเครื่องวัดประกอบการขับรด์ยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

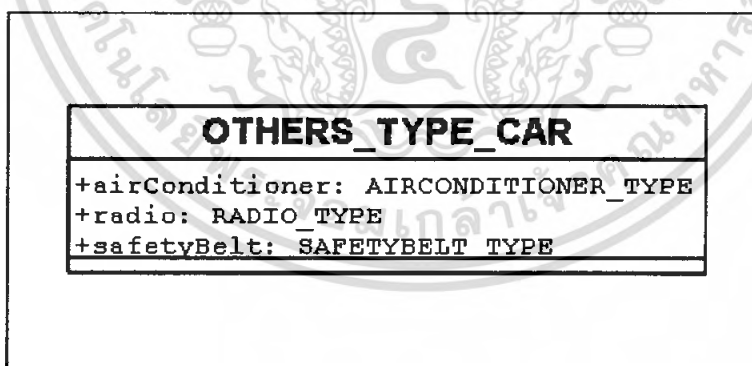
ตาราง 4.31 (ต่อ)

switchesAndControlLever	REF SWITCHESANDCONTROLLVER_T PE	SWITCHESANDCONTROLLVER_T YPE เป็นชนิดข้อมูลที่เก็บรายละเอียด สวิตช์และก้านควบคุมรถยนต์
headrest	HEADREST_VAR	HEADREST_VAR เป็นเน็ตเวิร์คเทเบิล ของHEADREST_TYPE ซึ่งเป็นชนิด ข้อมูลที่เก็บรายละเอียดหมอนพิงรถยนต์

ตาราง 4.32 แสดงรายละเอียดของชนิดข้อมูล ACCESSORIES_TYPE_MBIKE

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
handlebar	REF HANDLEBAR_TYPE	HANDLEBAR_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดที่สำหรับมือจับรถจักรยานยนต์
mainStand	REF MAINSTAND_TYPE	MAINSTAND_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดขาตั้งรถจักรยานยนต์
footPeg	REF FOOTPEG_TYPE	FOOTPEG_TYPE เป็นชนิดข้อมูลที่เก็บ รายละเอียดที่สำหรับขาเหยียบ รถจักรยานยนต์

4.2.1.15 อุปกรณ์อื่นๆ (Others)



รูปที่ 4.35 อุปกรณ์อื่นๆ (Others)

รูปที่ 4.35 แสดงชนิดข้อมูล OTHERS_TYPE_CAR ซึ่งเป็นชนิดข้อมูลที่เก็บรายละเอียด
อุปกรณ์อื่นๆ ของรถยนต์

ตาราง 4.33 แสดงรายละเอียดของชนิดข้อมูล OTHERS_TYPE_CAR

ชื่อแอตทริบิวต์	ชนิดข้อมูล	คำอธิบายเพิ่มเติม
airConditioner	REF AIRCONDITIONER_TYPE	AIRCONDITIONER_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเครื่องปรับอากาศรถยนต์
Radio	REF RADIO_TYPE	RADIO_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดวิทยุรถยนต์
safetyBelt	REF SAFETYBELT_TYPE	SAFETYBELT_TYPE เป็นชนิดข้อมูลที่เก็บรายละเอียดเข็มขัดนิรภัยรถยนต์

4.2.2 ตัวอย่างคำสั่งที่ใช้ในการสร้างชนิดข้อมูลในฐานข้อมูลรถ

4.2.2.1 ตัวอย่างคำสั่งการสร้างชนิดข้อมูล

ในฐานข้อมูลรถที่ออกแบบตามแนวคิดของฐานข้อมูลเชิงวัตถุ จะเก็บข้อมูลในรูปแบบของตาราง เช่นเดียวกับฐานข้อมูลเชิงสัมพันธ์ แต่ว่าจะมีคุณสมบัติของออปเจกต์เพิ่มเข้าไปในฐานข้อมูลด้วย ดังนั้นเพื่อให้ง่ายต่อการอธิบายจึงขอยกตัวอย่างขั้นตอนการสร้างชนิดข้อมูลในฐานข้อมูลรถ ดังนี้

1. สร้างชนิดข้อมูล PARTNER ซึ่งเป็นชนิดข้อมูลที่แสดงรายละเอียดของผู้ผลิตชิ้นส่วนอะไหล่ PARTNER เป็นชนิดข้อมูลที่จะประกอบอยู่ในทุกๆ ชนิดข้อมูล หรือกล่าวอีกนัยหนึ่งก็คือ อะไหล่รถทุกชิ้นต้องมีชื่อผู้ผลิตเสมอ โดยใน PARTNER ประกอบไปด้วยแอตทริบิวต์ 3 ค่า ได้แก่

name	ชื่อบริษัทผู้ผลิตชิ้นส่วนอะไหล่รถ
address	ที่อยู่ของบริษัทผู้ผลิตชิ้นส่วนอะไหล่รถ
phone	หมายเลขโทรศัพท์ของบริษัทผู้ผลิตชิ้นส่วนอะไหล่รถ

```
CREATE OR REPLACE TYPE partner AS OBJECT(
  name      VARCHAR2 (50),
  address   VARCHAR2 (256),
  phone     VARCHAR2 (20),
  MEMBER FUNCTION getName RETURN VARCHAR2,
  MEMBER FUNCTION getAddress RETURN VARCHAR2,
  MEMBER FUNCTION getPhone RETURN VARCHAR2,
  MEMBER PROCEDURE setName (n VARCHAR2),
  MEMBER PROCEDURE setAddress (a VARCHAR2),
  MEMBER PROCEDURE setPhone (p VARCHAR2),
  STATIC PROCEDURE doInsert (n VARCHAR2, a VARCHAR2, p VARCHAR2)
);
```

รูปที่ 4.36 คำสั่งในการสร้างชนิดข้อมูล PARTNER

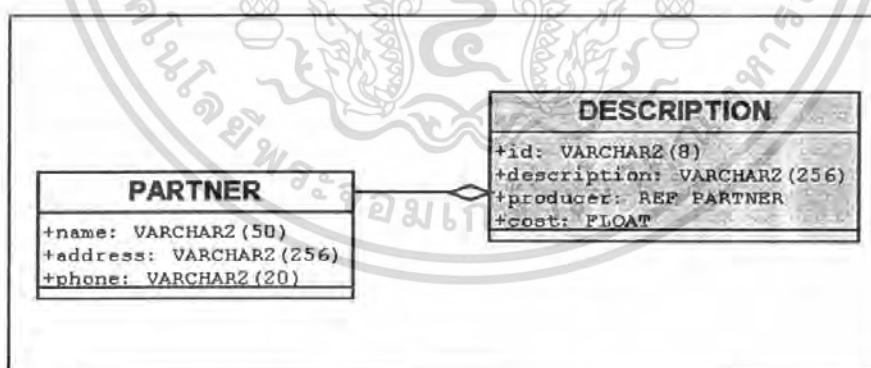
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สร้างชนิดข้อมูล DESCRIPTION ซึ่งเป็นชนิดข้อมูลที่แสดงรายละเอียดชิ้นส่วนอะไหล่ โดยใน DESCRIPTION ประกอบไปด้วยแอตทริบิวต์ 4 ค่าได้แก่

ID	รหัสของชนิดข้อมูล DESCRIPTION
descrip	รายละเอียดของอะไหล่รถ
producer	เป็นชนิดข้อมูล PARTNER
cost	ราคาอะไหล่รถ

```
CREATE OR REPLACE TYPE description AS OBJECT(
  ID          VARCHAR2 (8) ,
  descrip    VARCHAR2 (256) ,
  producer   REF partner,
  cost       FLOAT,
  MEMBER FUNCTION getId RETURN VARCHAR2,
  MEMBER FUNCTION getDescrip RETURN VARCHAR2,
  MEMBER FUNCTION getProducer RETURN REF partner,
  MEMBER FUNCTION getCost RETURN FLOAT,
  MEMBER PROCEDURE setID (i VARCHAR2),
  MEMBER PROCEDURE setDescrip(d VARCHAR2),
  MEMBER PROCEDURE setProducer(p VARCHAR2),
  MEMBER PROCEDURE setCost (c FLOAT)
)NOT FINAL;
```

รูปที่ 4.37 คำสั่งในการสร้างชนิดข้อมูล DESCRIPTION



รูปที่ 4.38 แสดงความสัมพันธ์ระหว่างชนิดข้อมูล DESCRIPTION และ PARTNER

3. สร้างชนิดข้อมูล RADIATOR_TYPE ซึ่งเป็นชนิดข้อมูลที่แสดงรายละเอียดหม้อน้ำของรถ โดยใช้คำสั่ง CREATE TYPE RADIATOR_TYEP UNDER DESCRIPTION ดังนั้นชนิดข้อมูล RADIATOR_TYPE ก็จะมีแอตทริบิวต์เหมือนกับชนิดข้อมูล DESCRIPTION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CREATE OR REPLACE TYPE radiator_type UNDER description(
  STATIC PROCEDURE doInsert(a VARCHAR2, b VARCHAR2, c VARCHAR2, d FLOAT)
);
```

รูปที่ 4.39 คำสั่งในการสร้างชนิดข้อมูล RADIATOR_TYPE

ในกรณีที่จะสร้างชนิดข้อมูลของชิ้นส่วนอะไหล่อื่นๆก็ใช้คำสั่งในการสร้างเช่นเดียวกับชนิดข้อมูล RADIATOR_TYPE

4. สร้างชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR (ระบบหล่อเย็นของรถยนต์) ซึ่งจะเป็นชนิดข้อมูลที่ประกอบไปด้วยอะไหล่รถที่อยู่ในระบบหล่อเย็น โดยในชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR ประกอบไปด้วยแอตทริบิวต์ดังนี้

radiator	เป็นชนิดข้อมูล RADIATOR_TYPE
oilCooler	เป็นชนิดข้อมูล OILCOOLER_TYPE
waterPump	เป็นชนิดข้อมูล WATERPUMP_TYPE
oilHose	เป็นชนิดข้อมูล OILHOSE_TYPE
thermostat	เป็นชนิดข้อมูล THERMOSTAT_TYPE
waterReservoir	เป็นชนิดข้อมูล WATERRESERVOIR_TYPE
radiatorCap	เป็นชนิดข้อมูล RADIATORCAP_TYPE

```

COOLINGSYSTEM_TYPE_CAR
+ radiator: RADIATOR_TYPE
+ oilCooler: OILCOOLER_TYPE
+ waterPump: WATERPUMP_TYPE
+ oilHose: OILHOSE_TYPE
+ thermostat: THERMOSTAT_TYPE
+ waterReservoir: WATERRESERVOIR_TYPE
+ radiatorCap: RADIATORCAP_TYPE

```

รูปที่ 4.40 COOLINGSYSTEM_TYPE_CAR

```

CREATE OR REPLACE TYPE coolingsystem_type_car AS OBJECT(
  id          VARCHAR2(10),
  radiator    REF radiator_type,
  oilCooler   REF oilcooler_type,
  waterPump   REF waterpump_type,
  oilHose     REF oilhose_type,
  thermostat  REF thermostat_type,
  waterReservoir REF waterreservoir_type,
  radiatorCap REF radiatorcap_type,
  MEMBER FUNCTION getID RETURN VARCHAR2,
  MEMBER FUNCTION getRadiator RETURN REF radiator_type,
  MEMBER FUNCTION getOilCooler RETURN REF oilcooler_type,
  MEMBER FUNCTION getWaterPump RETURN REF waterpump_type,
  MEMBER FUNCTION getOilHose RETURN REF oilhose_type,
  MEMBER FUNCTION getThermostat RETURN REF thermostat_type,
  MEMBER FUNCTION getWaterReservoir RETURN REF waterreservoir_type,
  MEMBER FUNCTION getRadiatorCap RETURN REF radiatorcap_type,
  MEMBER PROCEDURE setID (b VARCHAR2),
  MEMBER PROCEDURE setRadiator (b VARCHAR2),
  MEMBER PROCEDURE setOilCooler (b VARCHAR2),
  MEMBER PROCEDURE setWaterPump (b VARCHAR2),
  MEMBER PROCEDURE setOilHose (b VARCHAR2),
  MEMBER PROCEDURE setThermostat (b VARCHAR2),
  MEMBER PROCEDURE setWaterReservoir (b VARCHAR2),
  MEMBER PROCEDURE setRadiatorCap (b VARCHAR2),
  STATIC PROCEDURE doInsert(a VARCHAR2, b VARCHAR2, c VARCHAR2, d VARCHAR2,
                             e VARCHAR2, f VARCHAR2, g VARCHAR2, h VARCHAR2)
);

```

รูปที่ 4.41 คำสั่งในการสร้างชนิดข้อมูล COOLINGSYSTEM_TYPE_CAR

4.2.2.2 ตัวอย่างคำสั่งการสร้างชนิดข้อมูลที่มีคุณสมบัติการสืบทอดและโพลีมอร์ฟิซึม

```

CREATE OR REPLACE TYPE vehicle AS OBJECT(
  id          VARCHAR2(10),
  technicalInformation REF techinf,
  motorVehicleTax    FLOAT,
  | -- NOT INSTANTIABLE MEMBER FUNCTION CalculateMotorVehicleTax RETURN FLOAT,
  | NOT INSTANTIABLE MEMBER FUNCTION CalculateVehiclePrice RETURN FLOAT,
  | -- NOT INSTANTIABLE MEMBER FUNCTION SeatAmount RETURN INT,
  MEMBER FUNCTION getID RETURN VARCHAR2,
  MEMBER FUNCTION getTechnicalInformation RETURN REF techinf,
  MEMBER FUNCTION getMotorVehicleTax RETURN FLOAT,
  MEMBER PROCEDURE setID(b VARCHAR2),
  MEMBER PROCEDURE setTechnicalInformation(b VARCHAR2),
  MEMBER PROCEDURE setMotorVehicleTax(b FLOAT)
) NOT INSTANTIABLE
NOT FINAL;

```

รูปที่ 4.42 คำสั่งการสร้างชนิดข้อมูล VEHICLE

```

CREATE OR REPLACE TYPE automobile UNDER vehicle(
  engineComponents REF enginecomponents_type_car,
  enginePeripheralComponents REF engineperipheral_type_car,
  electricalSystem REF electricalsystem_type_car,
  fuelSystem REF fuelsystem_type_car,
  coolingSystem REF coolingsystem_type_car,
  exhaustSystem REF exhaustsystem_type_car,
  transmissionSystem REF transmissionsystem_type_car,
  brakeSystem REF brakesystem_type_car,
  suspensionSystem REF suspensionsystem_type_car,
  steeringSystem REF steeringssystem_type_car,
  mirror REF mirror_type_car,
  accessories REF accessories_type_car,
  other REF others_type_car,
  MEMBER FUNCTION getEngineComponents RETURN REF enginecomponents_type_car,
  MEMBER FUNCTION getEnginePeripheralComponents RETURN REF engineperipheral_type_car,
  MEMBER FUNCTION getElectricalSystem RETURN REF electricalsystem_type_car,
  MEMBER FUNCTION getFuelSystem RETURN REF fuelsystem_type_car,
  MEMBER FUNCTION getCoolingSystem RETURN REF coolingsystem_type_car,
  MEMBER FUNCTION getExhaustSystem RETURN REF exhaustsystem_type_car,
  MEMBER FUNCTION getTransmissionSystem RETURN REF transmissionsystem_type_car,
  MEMBER FUNCTION getBrakeSystem RETURN REF brakesystem_type_car,
  MEMBER FUNCTION getSuspensionSystem RETURN REF suspensionsystem_type_car,
  MEMBER FUNCTION getSteeringSystem RETURN REF steeringssystem_type_car,
  MEMBER FUNCTION getMirror RETURN REF mirror_type_car,
  MEMBER FUNCTION getAccessories RETURN REF accessories_type_car,
  MEMBER FUNCTION getOther RETURN REF others_type_car,
  MEMBER PROCEDURE setEngineComponents (b VARCHAR2),
  MEMBER PROCEDURE setEnginePeripheralComponents (b VARCHAR2),
  MEMBER PROCEDURE setElectricalSystem (b VARCHAR2),
  MEMBER PROCEDURE setFuelSystem (b VARCHAR2),
  MEMBER PROCEDURE setCoolingSystem (b VARCHAR2),
  MEMBER PROCEDURE setExhaustSystem (b VARCHAR2),
  MEMBER PROCEDURE setTransmissionSystem (b VARCHAR2),
  MEMBER PROCEDURE setBrakeSystem (b VARCHAR2),
  MEMBER PROCEDURE setSuspensionSystem (b VARCHAR2),
  MEMBER PROCEDURE setSteeringSystem (b VARCHAR2),
  MEMBER PROCEDURE setMirror (b VARCHAR2),
  MEMBER PROCEDURE setAccessories (b VARCHAR2),
  MEMBER PROCEDURE setOther (b VARCHAR2)
);
NOT INSTANTIABLE
NOT FINAL;

```

รูปที่ 4.43 คำสั่งการสร้างข้อมูล AUTOMOBILE

```

CREATE OR REPLACE TYPE car UNDER automobile(
  bodyWork REF carbodywork_type,
  OVERRIDING MEMBER FUNCTION CalculateMotorVehicleTax RETURN FLOAT,
  OVERRIDING MEMBER FUNCTION CalculateVehiclePrice RETURN FLOAT,
  OVERRIDING MEMBER FUNCTION SeatAmount RETURN INT,
  MEMBER FUNCTION getBodywork RETURN REF carbodywork_type,
  MEMBER PROCEDURE setBodywork(b VARCHAR2)
);

```

รูปที่ 4.44 คำสั่งการสร้างชนิดข้อมูล CAR

จากรูปที่ 4.42 เป็นการสร้างชนิดข้อมูล VEHICLE จะเห็นได้ว่าเมื่อทำการสร้างชนิดข้อมูลขึ้นมาแล้ว ยังมีการประกาศเพิ่มเติมว่า NOT FINAL ซึ่งมีความหมายว่า ชนิดข้อมูลนี้อ่อนุญาตให้ชนิดข้อมูลอื่นมาสืบทอดได้ และมีการประกาศ NOT INSTANTIABLE ไว้ที่ชนิดข้อมูล ซึ่งมีความหมายว่าชนิดข้อมูลนี้จะไม่สามารถสร้างออปเจกต์เองได้ ส่วนการประกาศ NOT

INSTANTIABLE ไว้ที่เมธอดนั้น มีความหมายว่าเมธอดนี้ไม่ต้องทำการอิมพลิเมนต์ ให้ไปอิมพลิเมนต์ในส่วนของชนิดข้อมูลที่มาสืบทอดแทน

จากรูปที่ 4.43 เป็นการสร้างชนิดข้อมูล AUTOMOBILE ที่ทำการสืบทอดมาจากชนิดข้อมูล VEHICLE และได้ทำการสร้างชนิดข้อมูล AUTOMOBILE ให้เป็นชนิดข้อมูลแบบ NOT FINAL และ NOT INSTANTIABLE อีกทีหนึ่ง

จากรูปที่ 4.44 เป็นการสร้างชนิดข้อมูล CAR ที่ทำการสืบทอดมาจากชนิดข้อมูล AUTOMOBILE และในชนิดข้อมูล CAR นี้ไม่ได้มีการประกาศให้เป็น NOT INSTANTIABLE แล้ว จึงต้องทำการอิมพลิเมนต์ในส่วนของเมธอดที่ได้รับการสืบทอดมาด้วย โดยใช้คีย์เวิร์ดคำว่า OVERRIDING อยู่หน้าเมธอดที่ต้องทำการอิมพลิเมนต์

4.2.2.3 ตัวอย่างคำสั่งการสร้างชนิดข้อมูลที่เป็นเน็สเต็ดเทเบิล

```
CREATE OR REPLACE TYPE suspensionsystem_type AS OBJECT(
    id          VARCHAR2(10),
    coilSpring REF coilspring_type,
    tyre        tyre_var,
    wheelHub   wheelhub_var,
    shockAbsorber REF shockabsorber_type,
    MEMBER FUNCTION getID RETURN VARCHAR2,
    MEMBER FUNCTION getCoilSpring RETURN REF coilspring_type,
    MEMBER FUNCTION getTyre RETURN tyre_var,
    MEMBER FUNCTION getWheelHub RETURN wheelhub_var,
    MEMBER FUNCTION getShockAbsorber RETURN REF shockabsorber_type,
    MEMBER PROCEDURE setID (b VARCHAR2),
    MEMBER PROCEDURE setCoilSpring (b VARCHAR2),
    MEMBER PROCEDURE setTyre (b tyre_var),
    MEMBER PROCEDURE setWheelHub (b wheelhub var),
    MEMBER PROCEDURE setShockAbsorber (b VARCHAR2)
) NOT FINAL
NOT INSTANTIABLE;
```

รูปที่ 4.45 คำสั่งการสร้างชนิดข้อมูล SUSPENSIONSYSTEM_TYPE

```
CREATE OR REPLACE TYPE tyre_type UNDER description(
    STATIC PROCEDURE doInsert(a VARCHAR2, b VARCHAR2,
                             c VARCHAR2, d FLOAT)
);

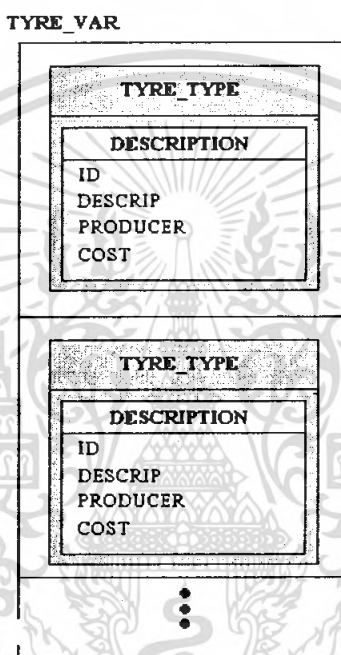
CREATE OR REPLACE TYPE tyre_var AS TABLE OF tyre_type;
```

รูปที่ 4.46 คำสั่งการสร้างชนิดข้อมูล CAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.45 แอดทริบิวต์ tyre และ wheelHub เป็นตัวแปรชนิดข้อมูลแบบเนสต์เตดเทเบิลทำให้การเก็บข้อมูลของล้อและคุมล้อสามารถเก็บเป็นเซตของข้อมูลได้ ทำให้การคิวรีข้อมูลนั้นง่ายขึ้นเพราะสามารถดึงข้อมูลเป็นเซตขึ้นมาแสดงผลได้ โดยไม่ต้องมีการ JOIN ตาราง (join)

จากรูปที่ 4.46 แสดงการสร้างชนิดข้อมูล TYRE_TYPE โดยมีแอดทริบิวต์ที่สืบทอดมาจากชนิดข้อมูล DESCRIPTION หลังจากนั้นก็ทำการสร้างชนิดข้อมูลแบบเนสต์เตดเทเบิลชื่อว่า TYRE_VAR ซึ่งมีการสร้างเหมือนการสร้างชนิดข้อมูลทั่วไป แต่ต่างกันที่มีการใช้คีย์เวิร์ดคำว่า AS TABLE OF แทน



รูปที่ 4.47 แสดง โครงสร้างของข้อมูลชนิดเนสต์เตดเทเบิล

4.2.3 ตัวอย่างคำสั่งที่ใช้ในการสร้างตารางในฐานข้อมูลรถ

4.2.3.1 ตัวอย่างคำสั่งที่ใช้ในการสร้างตาราง

```

4.  --ENGINECOMPONENTS_TYPE
CREATE TABLE piston_tab OF piston_type;
CREATE TABLE pistonring_tab OF pistonring_type;
CREATE TABLE connectingrod_tab OF connectingrod_type;
CREATE TABLE crankshaft_tab OF crankshaft_type;
CREATE TABLE gasket_tab OF gasket_type;
CREATE TABLE flywheel_tab OF flywheel_type;
CREATE TABLE cam_tab OF cam_type;
CREATE TABLE camshaft_tab OF camshaft_type;

```

รูปที่ 4.48 คำสั่งการสร้างตารางขึ้นส่วนทั้งหมดในระบบเครื่องยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CREATE TABLE suspensionssystem_tab_car OF suspensionssystem_type_car
NESTED TABLE tyre STORE AS tyre_nt
NESTED TABLE wheelhub STORE AS wheelhub_car_nt
NESTED TABLE stubaxle STORE AS stubaxle_car_nt
NESTED TABLE pressedsteelwheel STORE AS pressedsteelwheel_nt;
```

รูปที่ 4.49 คำสั่งการสร้างตารางชนิดที่มีคอลัมน์เป็นเนสต์เตดเทเบิล

จากรูป 4.48 เป็นการสร้างออปเจกต์เทเบิล ซึ่งเป็นเทเบิลที่สร้างขึ้นมาจากชนิดข้อมูลที่ผู้ใช้สร้างขึ้นเอง มีรูปแบบการใช้งานคือ CREATE TABLE table_name OF type_name;

จากรูป 4.49 เป็นการสร้างออปเจกต์เทเบิล ที่แอตทริบิวต์ภายในชนิดข้อมูลที่จะนำมาสร้างตารางเป็นเนสต์เตดเทเบิล จะต้องมรูปแบบการสร้างที่ต่างออกไป เพราะในออราเคิลจะมีการเก็บข้อมูลที่เป็นเนสต์เตดเทเบิลแยกไว้ต่างหากอีกตารางหนึ่ง เราจึงจำเป็นที่จะต้องสร้างชื่อสำหรับตารางเหล่านั้น ซึ่งรูปแบบการสร้างจะต้องมีการระบุ NESTED TABLE column_name STORE AS table_name ด้วย

4.2.4 ตัวอย่างคำสั่งที่ใช้ในการเพิ่ม (INSERT) ข้อมูลในฐานข้อมูล

```
INSERT INTO partner_tab
VALUES (partner('Car Spares',
'53/8 Moo 2 Bangphu Muang Pathumthani 12000',
'029580003'));
INSERT INTO partner_tab
VALUES (partner('Thai Cars',
'657/23 Romkloa Meenburi Bangkok 10510',
'029197773'));
```

รูปที่ 4.50 คำสั่งการเพิ่มข้อมูลลงในตาราง PARTNER_TAB

```

INSERT INTO brakepad_tab
SELECT brakepad_type('BPO0001' , 'description1' , REF(p) , 6070)
FROM partner_tab p
WHERE p.NAME = 'Car Spares';

```

```

INSERT INTO brakepad_tab
SELECT brakepad_type('BPO0002' , 'description2' , REF(p) , 6120)
FROM partner_tab p
WHERE p.NAME = 'Car Spares';

```

```

INSERT INTO brakepad_tab
SELECT brakepad_type('BPO0003' , 'description2' , REF(p) , 8320)
FROM partner_tab p
WHERE p.NAME = 'Discount Autoparts';

```

รูปที่ 4.51 คำสั่งการเพิ่มข้อมูลลงในตาราง BRAKEPAD_TAB

4.2.5 ตัวอย่างคำสั่งที่ใช้ในการคิวรีข้อมูลในฐานข้อมูลรถ

ในหัวข้อนี้จะแสดงตัวอย่างการคิวรีข้อมูลในฐานข้อมูลทั้งบนฐานข้อมูลเชิงวัตถุ สัมพันธ์และฐานข้อมูลเชิงสัมพันธ์ เพื่อแสดงให้เห็นถึงข้อดีของฐานข้อมูลที่พัฒนาตามหลักการของฐานข้อมูลเชิงวัตถุสัมพันธ์

ตัวอย่าง 1 ให้แสดงรหัสรถยนต์ (ID), รุ่นรถยนต์ (MODEL) และชื่อบริษัทที่ผลิตสินค้า (PRODUCER_NAME) ที่ใช้ประกอบชิ้นส่วนรถยนต์ในระบบน้ำมันเชื้อเพลิง

```

SELECT c.ID, c.TECHNICALINFORMATION.ENGINE.ENGINEMODEL MODEL,
c.FUELSYSTEM.ID,
c.FUELSYSTEM.FUELTANK.PRODUCER.NAME FUELTANK,
c.FUELSYSTEM.CAP.PRODUCER.NAME CAP,
c.FUELSYSTEM.FUELPIPE.PRODUCER.NAME FUELPIPE,
c.FUELSYSTEM.FUELGAUGE.PRODUCER.NAME FUELGAUGE,
c.FUELSYSTEM.FUELPUMP.PRODUCER.NAME FUELPUMP,
c.FUELSYSTEM.FUELFILTER.PRODUCER.NAME FUELFILTER
FROM car_tab c

```

รูปที่ 4.52 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์

```

SELECT c.CAR_ID CAR_ID,
       tec.ENGINEMODEL MODEL,
       fsys.FUELSYSTEMCAR_ID FUEL_ID,
       p1.NAME FULETANK,
       p2.NAME CAP,
       p3.NAME FUELPIPE,
       p4.NAME FUELGAUGE,
       p5.NAME FUELPUMP,
       p6.NAME FUELFILTER
FROM   car_tab c, techni_tab tec,
       fuelsystemcar_tab fsys, fueltank_tab ft,
       cap_tab cp, fuelpipe_tab fpi,
       fuelgauge_tab fg, fuelpump_tab fpu, fuelfilter_tab ff,
       partner p1, partner p2, partner p3,
       partner p4, partner p5, partner p6
WHERE  c.FUELSYSTEMCAR_ID = fsys.FUELSYSTEMCAR_ID and
       c.TECHNINF_ID = tec.TECHINF_ID and
       fsys.FUELTANK_ID = ft.FUELTANK_ID and
       fsys.CAP_ID = cp.CAP_ID and
       fsys.FUELPIPE_ID = fpi.FUELPIPE_ID and
       fsys.FUELGAUGE_ID = fg.FUELGAUGE_ID and
       fsys.FUELPUMP_ID = fpu.FUELPUMP_ID and
       fsys.FUELFILTER_ID = ff.FUELFILTER_ID and
       ft.PRODUCER_ID = p1.PRODUCER_ID and
       cp.PRODUCER_ID = p2.PRODUCER_ID and
       fpi.PRODUCER_ID = p3.PRODUCER_ID and
       fg.PRODUCER_ID = p4.PRODUCER_ID and
       fpu.PRODUCER_ID = p5.PRODUCER_ID and
       ff.PRODUCER_ID = p6.PRODUCER_ID

```

รูปที่ 4.53 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงสัมพันธ์

CAR_ID	MODEL	FUEL_ID	FUELTANK	CAP
CAR000001	TOYOTA YARIS 1NZ-FE	FUELC00001	Car Spares	JAFCO ENTERPRISE
CAR000002	TOYOTA CAMRY 2GR-FEV	FUELC00002	Cambridge (Auto Bulbs)	JAFCO ENTERPRISE
CAR000003	TOYOTA VIOS 1NZ-FE	FUELC00003	Car Spares	JAFCO ENTERPRISE
CAR000004	COROLLA ALTI 3ZZ-FE	FUELC00004	The Spare Place	JAFCO ENTERPRISE

รูปที่ 4.54 คำสั่งแสดงผลลัพธ์การคิวรี

จากตัวอย่างข้างต้น รูปที่ 4.52 แสดงคำสั่งการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์ ส่วนรูปที่ 4.53 แสดงคำสั่งคิวรีบนฐานข้อมูลเชิงสัมพันธ์ จะเห็นว่า คำถามเดียวกัน ผลลัพธ์เหมือนกันแต่คำสั่งที่ใช้แตกต่างกัน การคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์จะทำได้ง่ายกว่าการคิวรีบนฐานข้อมูลเชิงสัมพันธ์ เนื่องจากแอตทริบิวต์สามารถกำหนดเป็นชนิดข้อมูลอ้างอิง (reference) ได้ การเข้าถึงข้อมูลจึงใช้เพียง dot notation (.) ในการอ้างถึงแอตทริบิวต์ที่อยู่คนละตาราง ในส่วนการคิวรีบนฐานข้อมูลเชิงสัมพันธ์จะต้องทำการ JOIN ตารางก่อนจึงจะสามารถเรียกค้นข้อมูลได้อย่างถูกต้อง

ตัวอย่าง 2 ให้แสดงรหัส (ID) รถยนต์, รุ่นรถยนต์ (MODEL), ราคากำหนด และราคาของรถแต่ละคัน

```
SELECT v.ID,
       v.TECHNICALINFORMATION.ENGINE.ENGINEMODEL MODEL,
       v.CALCULATEMOTORVEHICLETAX() TAX,
       v.CALCULATEVEHICLEPRICE() PRICE
FROM   vehicle_tab v
```

รูปที่ 4.55 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์

ID	MODEL	TAX	PRICE
CAR0000001	TOYOTA YARIS 1NZ-FE	1645.5	1172865
CAR0000002	TOYOTA CAMRY 2GR-FEV	8724	1199386
CAR0000003	TOYOTA VIOS 1NZ-FE	1645.5	1393281
CAR0000004	COROLLA ALTIS 3ZZ-FE	1797	1531053
PICK0000001	TOYOTA WISH 1AZ-FE	900	1616446
PICK0000002	TOYOTA WISH 1AZ-FE	1050	1446904
PICK0000003	TOYOTA VIOS 1NZ-FE	900	1668984
MOTOR000001	TOYOTA YARIS 1NZ-FE	100	277195

รูปที่ 4.56 ผลลัพธ์การคิวรี

จากตัวอย่างการคิวรีในรูปที่ 4.55 เป็นการเรียกใช้เมธอดจากชนิดข้อมูล ซึ่งภายในเมธอดนี้ จะมีการใช้การตัดสินใจ (If-Condition) และการคำนวณร่วมด้วย เนื่องจากมีการใช้การตัดสินใจ ทำให้การเขียนคำสั่งคิวรีแบบนี้ในฐานข้อมูลเชิงสัมพันธ์ทำไม่ได้ ต้องมีการเลือกขึ้นมาทีละค่า แล้วนำมาทำการตัดสินใจและการคำนวณในส่วนของโปรแกรมแทน

ตัวอย่าง 3 ให้แสดงรุ่นรถยนต์, สีรถยนต์, ราคาของกระจกบังลม และรายละเอียดกระจกบังลม

```
SELECT ca.TECHNICALINFORMATION.ENGINE.ENGINEMODEL MODEL,
       ca.TECHNICALINFORMATION.SIZEANDWEIGHT.COLOR COLOR,
       fr.COST WINDSHIELDSCREEN_COST,
       fr.DESCRIP DESCRIPTION
FROM   car_tab ca, TABLE(ca.MIRROR.FRONTREARWINDSHIELDSCREEN) fr
```

รูปที่ 4.57 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงวัตถุสัมพันธ์

```

SELECT  t.enginemodel      MOTORCYCLE_MODEL,
        t.color            COLOR,
        frch.cost         COST,
        frch.description  DESCRIPTION
FROM    techinf_tab      t,
        car_tab         c,
        mirrorcar_tab   mc,
        frearwindshieldscreensuper_tab frsp,
        frearwindshieldscreenchild_tab frch
WHERE   c.techninf_id = t.techninf_id AND
        c.mirrorcar_id = mc.mirrorcar_id AND
        mc.frearwindshieldscreensuper_id = frsp.frearwindshieldscreensuper_id AND
        frsp.frearwindshieldscreenchild_id = frch.frearwindshieldscreenchild_id

```

รูปที่ 4.58 คำสั่งแสดงการคิวรีบนฐานข้อมูลเชิงสัมพันธ์

MODEL	COLOR	WINDSHIELDSCREEN_COST	DESCRIPTION
TOYOTA YARIS 1NZ-FE	Platinum Silver	5900	Car1 front windshield screen
TOYOTA YARIS 1NZ-FE	Platinum Silver	4777	Car1 rear windshield screen
TOYOTA CAMRY 2GR-FEV	White Pearl Crystal	9005	Car2 front windshield screen
TOYOTA CAMRY 2GR-FEV	White Pearl Crystal	5800	Car2 rear windshield screen
TOYOTA MIO 1NZ-FE	Greyish Blue Metallic	12345	Car3 front windshield screen
TOYOTA MIO 1NZ-FE	Greyish Blue Metallic	11200	Car3 rear windshield screen
COROLLA ALTIS 3ZZ-FE	Blue Mica	20000	Car4 front windshield screen
COROLLA ALTIS 3ZZ-FE	Blue Mica	13000	Car4 rear windshield screen

รูปที่ 4.59 ผลลัพธ์การคิวรี

ตัวอย่างนี้แสดงการคิวรีข้อมูลที่เป็นเน็สเต็ดเทเบิลบนฐานข้อมูลเชิงวัตถุสัมพันธ์ ซึ่งในที่นี้ชนิดข้อมูลกระจกบังลมของรถยนต์ เก็บเป็นเน็สเต็ดเทเบิล การคิวรีข้อมูลสามารถอ้างอิงผ่านตารางหลัก (CAR_TAB) โดยที่ไม่ต้องทำการจอยตารางเหมือนกับการคิวรีในฐานข้อมูลเชิงสัมพันธ์

บทที่ 5

บทสรุปและวิจารณ์

5.1 สรุปผลการดำเนินงาน

โครงการนี้ ในขั้นแรกจะทำการศึกษาและค้นคว้าเกี่ยวกับทฤษฎีของฐานข้อมูลเชิงวัตถุสัมพันธ์ ทั้งในเชิงวิชาการและเชิงพาณิชย์ จากนั้นก็ศึกษาการออกแบบและการสร้างฐานข้อมูลเชิงวัตถุสัมพันธ์ รวมทั้งการใช้งานออรากิล 10g จากคู่มือของออรากิล และจากเอกสารอื่นที่เกี่ยวข้อง พร้อมทั้งทำการทดสอบใช้งานออรากิล 10g ไปในขณะเดียวกัน

ในส่วนของการทดสอบนั้น ได้ทำการออกแบบข้อมูลในตาราง President ให้เป็นฐานข้อมูลเชิงวัตถุสัมพันธ์แล้วนำมาทดสอบการเรียกค้นข้อมูล โดยใช้ภาษาเอสคิวแอลร่วมกับออรากิล 10g

นอกจากจากนั้นยังได้ทำการออกแบบฐานข้อมูลรถ ซึ่งเก็บอะไหล่ของรถยนต์ รถกระบะ และรถจักรยานยนต์ ในรูปแบบของฐานข้อมูลเชิงวัตถุสัมพันธ์และฐานข้อมูลเชิงสัมพันธ์ พร้อมทั้งทำการคิวรีข้อมูลจากฐานข้อมูลที่ได้ออกแบบไว้ โดยแสดงการคิวรีในส่วนที่ออกแบบตามแนวคิดเชิงวัตถุ เพื่อแสดงให้เห็นถึงข้อดีของแนวความคิดเชิงวัตถุที่มีในฐานข้อมูลเชิงวัตถุสัมพันธ์

5.2 วิจารณ์สิ่งที่ได้จากโครงการ

1. ทฤษฎีของโครงสร้างข้อมูลเชิงวัตถุสัมพันธ์และคุณสมบัติของเอสคิวแอล 3
2. การใช้งานออรากิล 10g และสถาปัตยกรรมต่างๆ ของออรากิล 10g
3. การออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์ โดยใช้แนวคิดเชิงวัตถุ
4. การใช้งานภาษาเอสคิวแอลและภาษาพีแอลเอสคิวแอล (PL/SQL)

5.3 ปัญหาและอุปสรรค

5.3.1 ปัญหา

1. ในการศึกษาทฤษฎีของฐานข้อมูลเชิงสัมพันธ์จะเกิดปัญหาเกี่ยวกับศัพท์เทคนิคในเชิงวิชาการกับศัพท์เทคนิคในเชิงพาณิชย์ ซึ่งจะมีความแตกต่างกัน รวมทั้งรายละเอียดของทฤษฎี ถ้าไปศึกษาในเชิงพาณิชย์ เจ้าของผลิตภัณฑ์ก็จะกล่าวถึงเฉพาะในเชิงของผลิตภัณฑ์ของตัวเองเท่านั้น ทำให้ได้ข้อเท็จจริงที่ไม่ครอบคลุมกับทุกผลิตภัณฑ์
 2. การหาข้อมูลอะไหล่รถยนต์ ไม่ว่าจะเป็นอะไหล่ทั้งหมดที่ใช้ประกอบเป็นรถ หนึ่งคัน, ราคา, ชื่อบริษัทผู้ผลิต, การคำนวณภาษี ค่อนข้างจะหาข้อมูลได้ยากมาก เนื่องจากไม่ทราบว่าแต่ละระบบของรถคันหนึ่งประกอบจากอะไรว่าง
- เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.2 อุปสรรค

1. เอกสารที่อ้างอิงในการทำโครงการ ในส่วนของทฤษฎีนี้ ไม่ว่าจะเป็นจากหนังสือ หรือ งานวิจัย รายละเอียดข้างในเอกสารที่ใช้อ้างอิงยังมีที่ขัดแย้งกันเอง ไม่ครอบคลุมพอ หรือแม้กระทั่ง ไม่ถูกต้องตามทฤษฎีเลขก็มี
2. ราคาอะไหล่รถถึงแม้จะเป็นอะไหล่แบบเดียวกันก็จะมีราคาแตกต่างกันไป ขึ้นอยู่กับบริษัทที่เป็นตัวแทนจำหน่าย ยี่ห้อของผลิตภัณฑ์นั้นๆ

5.4 แนวทางแก้ไข

จากปัญหาและอุปสรรคที่ได้กล่าวมาข้างต้น มีแนวทางในการแก้ปัญหาคือ

1. ต้องศึกษาจากเอกสารอ้างอิงหลายๆ แหล่ง และเอกสารเหล่านั้นต้องน่าเชื่อถือ ซึ่งต้องทำการตรวจสอบประวัติหรือหาข้อมูลเพิ่มเติมเกี่ยวกับผู้เขียนด้วย หลังจากนั้นก็ทำการวิเคราะห์ข้อมูลที่ได้อีกครั้ง แล้วจึงสรุปออกมา ซึ่งในบางครั้งอาจต้องทำการทดลองสร้างฐานข้อมูลแล้วเรียกค้นข้อมูล เพื่อประกอบการพิจารณาด้วย

2. สำหรับปัญหาเกี่ยวกับข้อมูลของอะไหล่รถ เนื่องจากว่าอะไหล่รถแต่ละชิ้นมีหลายราคาแตกต่างกัน ในโครงการนี้จึงได้อ้างอิงข้อมูลจากแหล่งข้อมูลเพียงแหล่งเดียว ดังนี้คือ

- 2.1 ข้อมูลอะไหล่รถยนต์และรถกระบะ ในส่วนของราคานั้น โครงการนี้ได้อ้างอิงข้อมูลจากเว็บไซต์ www.autoatone.com ซึ่งในเว็บไซต์นี้จะมีบริการถามตอบราคาของอะไหล่รถยนต์ ส่วนข้อมูลทางเทคนิคของรถยนต์ได้อ้างอิงจากเว็บไซต์ของบริษัทโตโยต้า (www.toyota.co.th)

- 2.2 ข้อมูลอะไหล่รถจักรยานยนต์ โครงการนี้ได้อ้างอิงจากเว็บไซต์เอพียอนด์ไทยแลนด์ www.aphonda.co.th

5.5 แนวทางในการพัฒนาต่อ

สำหรับแนวทางในการพัฒนาต่อของโครงการนี้คือ

1. สร้างแอปพลิเคชันมาเรียกใช้ฐานข้อมูลรถที่ได้ออกแบบไว้
2. ข้อมูลเกี่ยวกับอะไหล่ของรถที่ใช้ควรจะเป็นข้อมูลจริง เพื่อที่จะให้สามารถนำฐานข้อมูลรถไปใช้ได้จริง โดยต้องออกแบบให้ตรงกับความต้องการในการใช้งานด้วย ดังนั้นจึงต้องมีการไปเก็บความต้องการของผู้ใช้ (Requirement) และศึกษาข้อมูลเกี่ยวกับส่วนประกอบของรถเพิ่มเติมไม่ว่าจะเป็น การคิดคำนวณราคา รถ ภาษีรถ เป็นต้น
3. ศึกษาสถาปัตยกรรมและการใช้งานออร์เคิล 10g เพิ่มเติม เพื่อที่จะสามารถใช้ผลิตภัณฑ์นี้ได้อย่างมีประสิทธิภาพที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

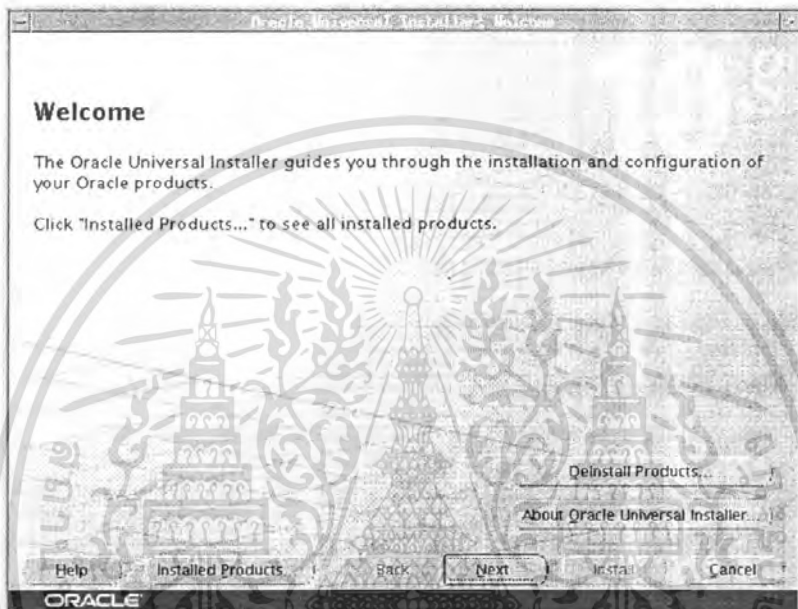
บรรณานุกรม

- ศุภชัย จิระรังสี และ ขจรศักดิ์ สังข์เจริญ. 2549. ระบบฐานข้อมูล Oracle Database 10g Express Edition. กรุงเทพมหานคร : TrainLyst.
- ยุทธนา ลีลาศวัฒนกุล และ อมรพันธุ์ คำอรรถ. 2545. สร้างระบบงานฐานข้อมูลด้วย PL/SQL และ Oracle Developer. นนทบุรี : อินโฟเพรส.
- Aphonda. Aphonda [Online].
Available : <http://www.aphonda.co.th/honda2006/index.asp>.
- AutoatOne. กระดานตอบราคาอะไหล่. [Online].
Available : http://www.autoatone.com/inquiry_answer.php?page=answer.
- Chen, Y.S. 2007. Object-Relational Concepts. [Online].
Available : <http://www.cs.nthu.edu.tw/~yishin/Courses/ISA6120/Handout/Session6%20-20Handout.pdf>.
- Elmasri, R. and Navathe, S.B. 2007. Fundamentals of Database Systems. 5th ed. Boston. Addison Wesley.
- Memon, N. 2007. Object-Relational Database Systems. [Online].
Available : http://cs.aau.dk/~nasrullah/F7S/db/Fall06/lecture_notes/week41_2b.pdf.
- Oracle. 2007. Oracle Database Application Developer's Guide – Object-Relational Feature 10g Release 2 (10.2). [Online].
Available : http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14260/toc.htm.
- Stonebraker, M. and Brown, P. 1999. Object-Relational DBMSs Tracking the Next Great Wave. San Francisco : Morgan Kaufmann.
- Thaiautoparts. ชิ้นส่วนต่างๆ ที่ประกอบขึ้นเป็นรถยนต์ 1 คัน. [Online].
Available : www.thaiautoparts.or.th/fileupload/autoparts14.doc.
- Toyota. Toyota. [Online].
Available : http://www.toyota.co.th/th/toyota2007_home.htm.

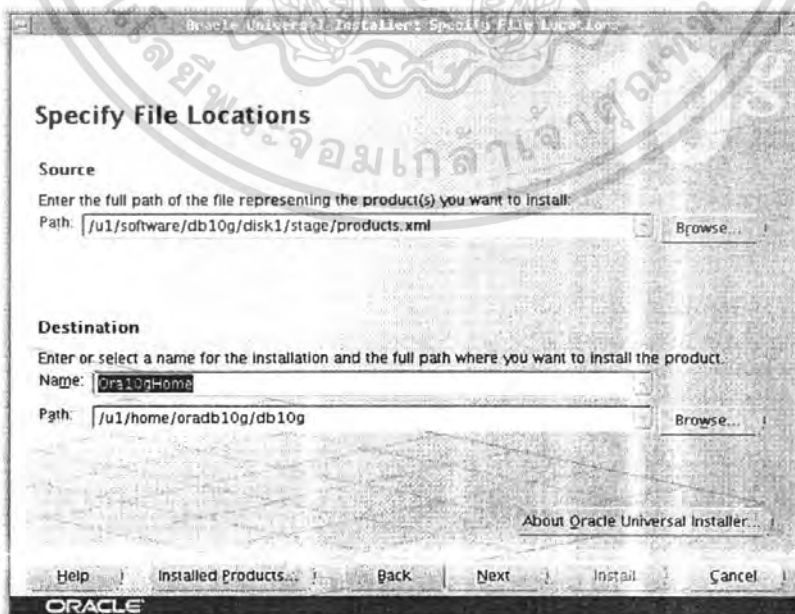
ภาคผนวก ก.

การติดตั้งฐานข้อมูลออราเคิล 10g

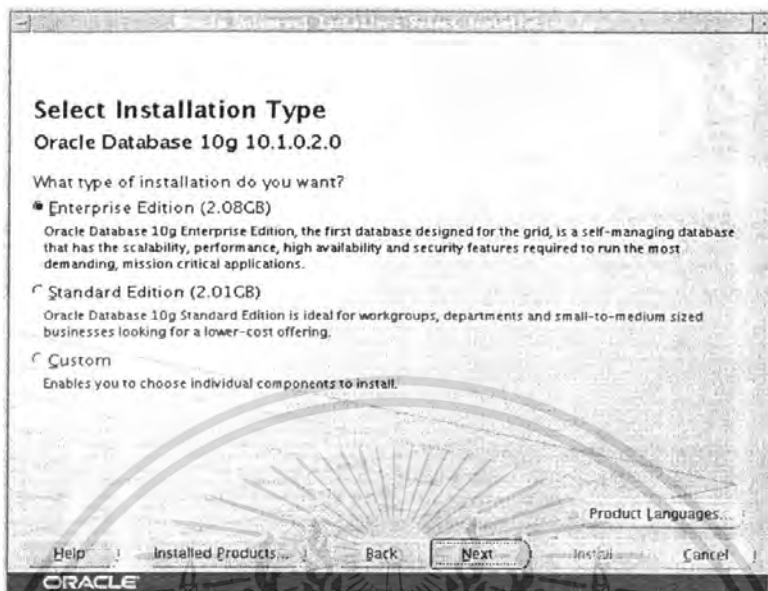
ก.1 การติดตั้งฐานข้อมูลออราเคิล 10g



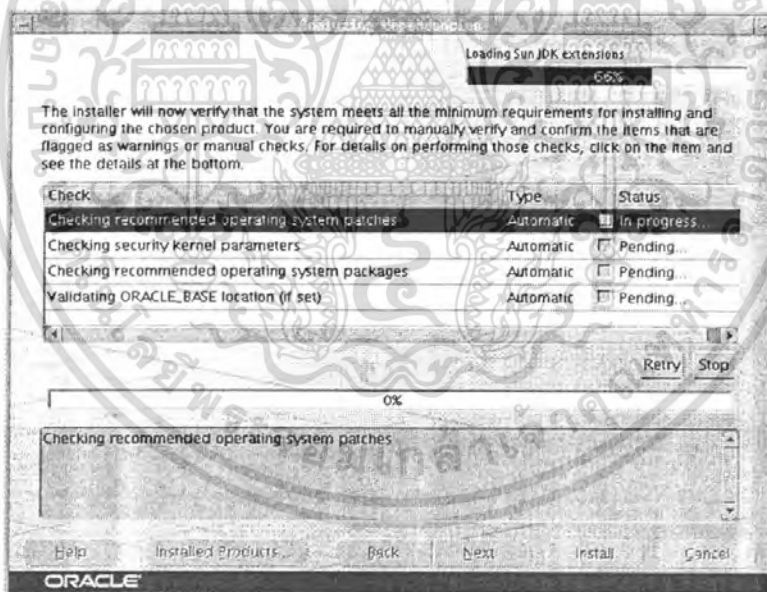
รูปที่ ก.1 หน้าจอ Welcome เป็นหน้าจอแรกของการติดตั้ง



รูปที่ ก.2 หน้าจอ Specify File Locations ตัวโปรแกรมติดตั้งจะให้เลือก Source file ที่ใช้ติดตั้ง และให้เลือก Oracle Home และ Path ที่เก็บด้วย ซึ่งสามารถเลือกใช้ค่า default ที่มีให้อยู่แล้วได้ ยกเว้นกรณีการตั้งค่าอื่น ๆ นอกเหนือจากนี้ อย่างไรก็ตามถ้าหากต้องการแก้ไขค่า default ที่มีให้อยู่แล้วได้ ยกเว้นกรณีการตั้งค่าอื่น ๆ นอกเหนือจากนี้ อย่างไรก็ตามถ้าหากต้องการแก้ไขค่า default ที่มีให้อยู่แล้วได้ ยกเว้นกรณีการตั้งค่าอื่น ๆ นอกเหนือจากนี้

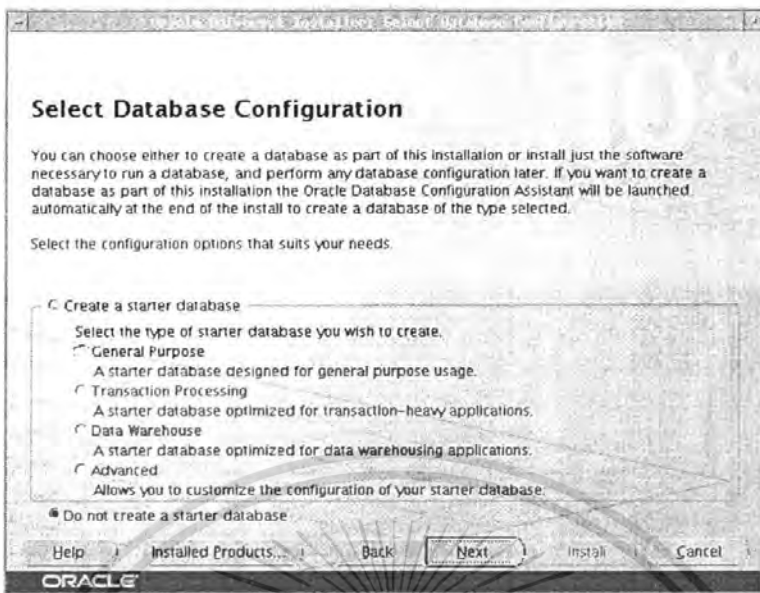


รูปที่ ก.3 หน้าจอ Select Installation Type ให้เลือก Product สำหรับการติดตั้ง

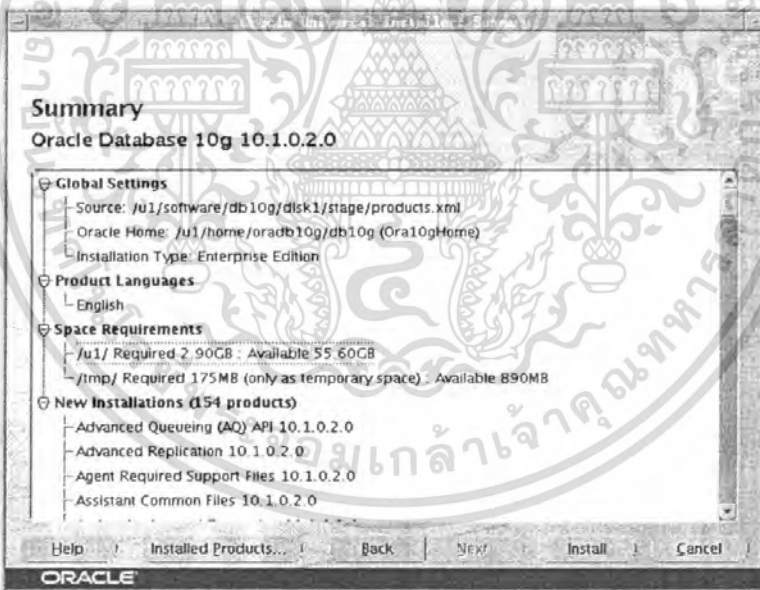


รูปที่ ก.4 หน้าจอ Analyzing dependencies

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

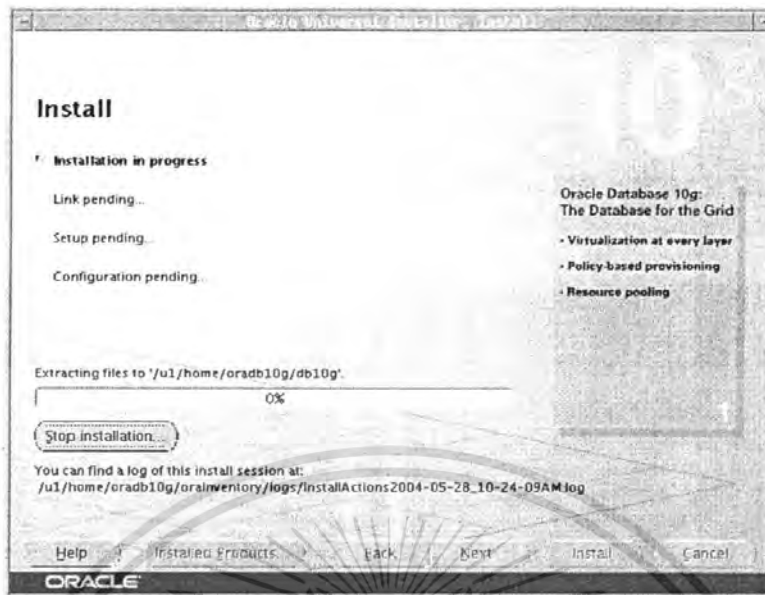


รูปที่ ก.5 หน้าจอ Select Database Configuration ตัวโปรแกรมติดตั้งจะให้เลือกว่าจะสร้าง database ในระหว่างการติดตั้งหรือไม่ ในที่นี้เลือกไม่สร้าง database

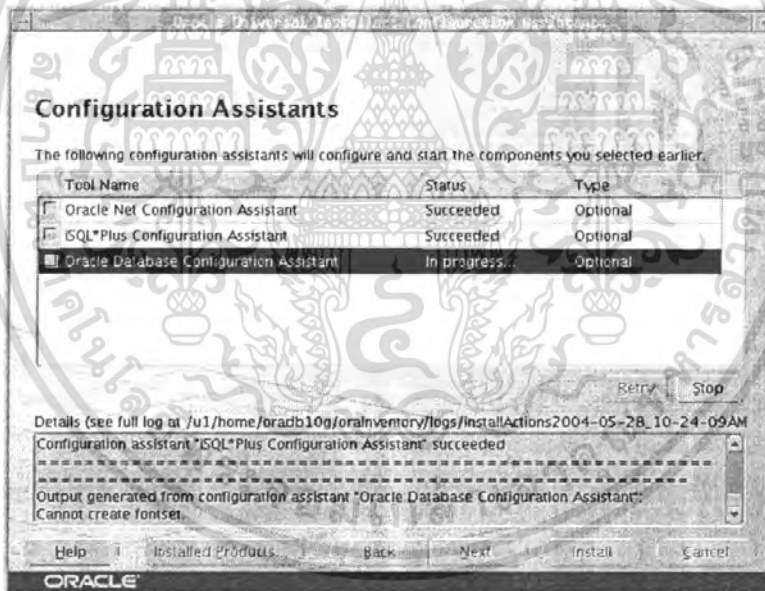


รูปที่ ก.6 หน้าจอ Summary ตัวโปรแกรมติดตั้งจะสรุปข้อมูลการติดตั้งให้อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



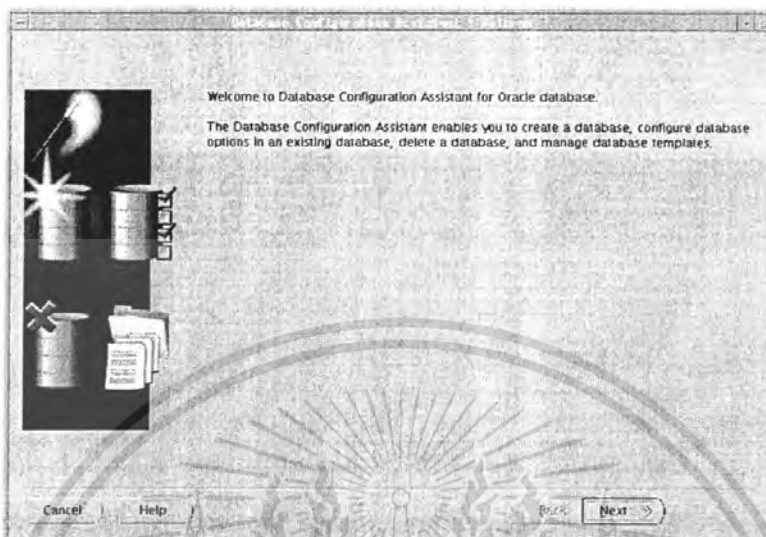
รูปที่ ก.7 หน้าจอ Install



รูปที่ ก.8 หน้าจอ Configuration Assistants หลังจากติดตั้ง Oracle Database Software เรียบร้อยแล้ว ตัวโปรแกรมติดตั้งจะเข้าสู่หน้าจอนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.2 ขั้นตอนการสร้างฐานข้อมูลจาก Database Configuration Assistant ของออราเคิล

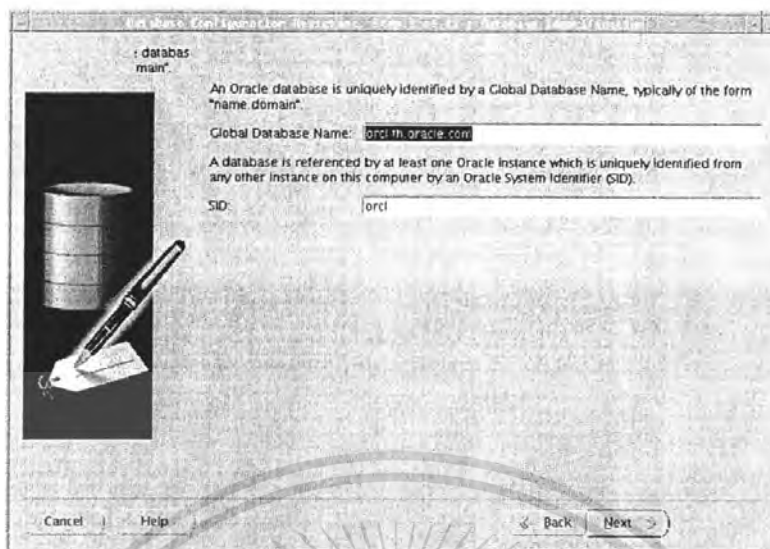


รูปที่ ก.9 หน้าจอ Database Configuration Assistant : Welcome



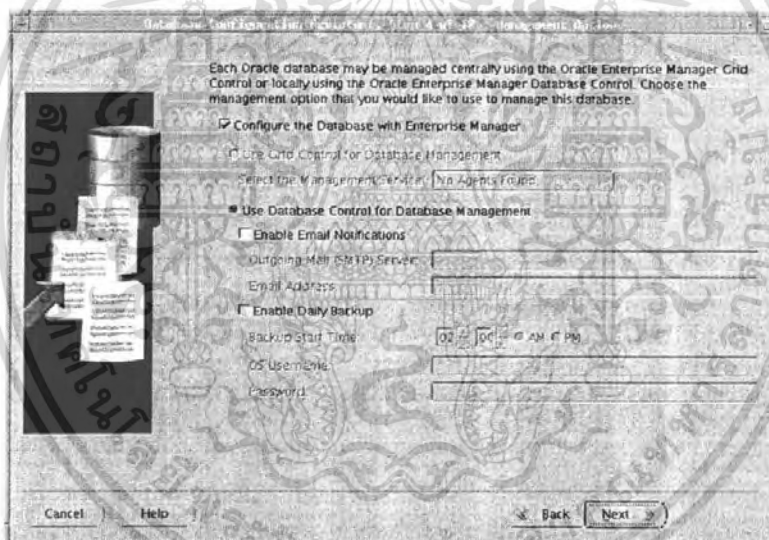
รูปที่ ก.10 หน้าจอ Operations เลือก Create a Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



An Oracle database is uniquely identified by a Global Database Name, typically of the form "name.domain".
 Global Database Name:
 A database is referenced by at least one Oracle Instance which is uniquely identified from any other instance on this computer by an Oracle System Identifier (SID).
 SID:

รูปที่ ก.11 ตั้งชื่อให้กับ database



Each Oracle database may be managed centrally using the Oracle Enterprise Manager Grid Control or locally using the Oracle Enterprise Manager Database Control. Choose the management option that you would like to use to manage this database.

Configure the Database with Enterprise Manager
 Use Grid Control for Database Management
 Select the Management Option:

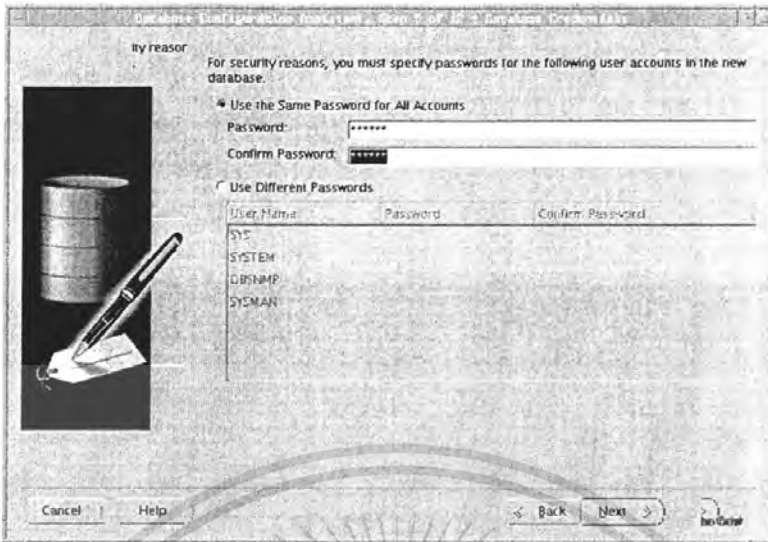
Use Database Control for Database Management
 Enable Email Notifications
 Outgoing Mail (SMTP) Server:
 Email Address:

Enable Daily Backup
 Backup Start Time:

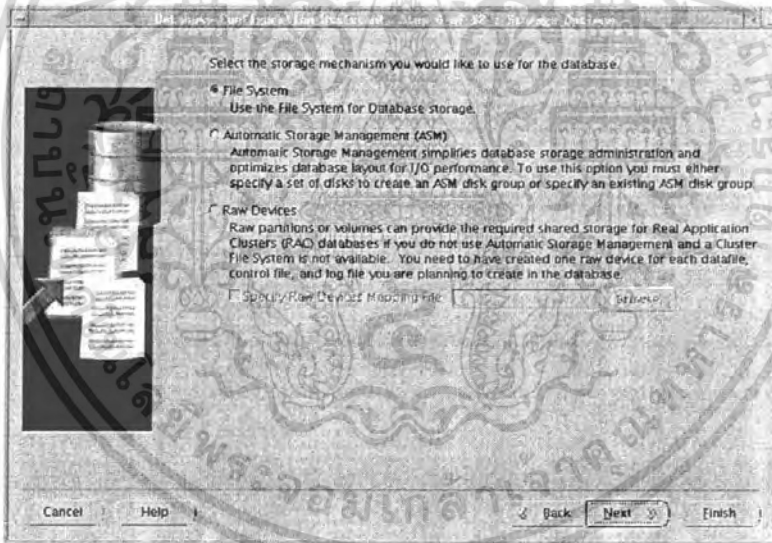
OS Username:
 Password:

รูปที่ ก.12 หน้าจอ Management Option เลือก Configure the Database with Enterprise Manager และเลือก Use Database Control for Database Management

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

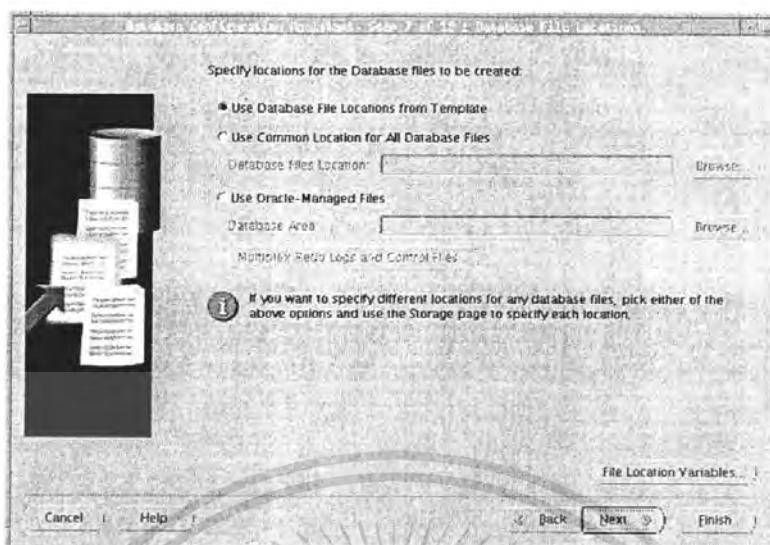


รูปที่ ก.13 หน้าจอ Database Credentials เลือกใช้ password เดียวกันกับทุก Account จากนั้นกำหนด password ลงไป

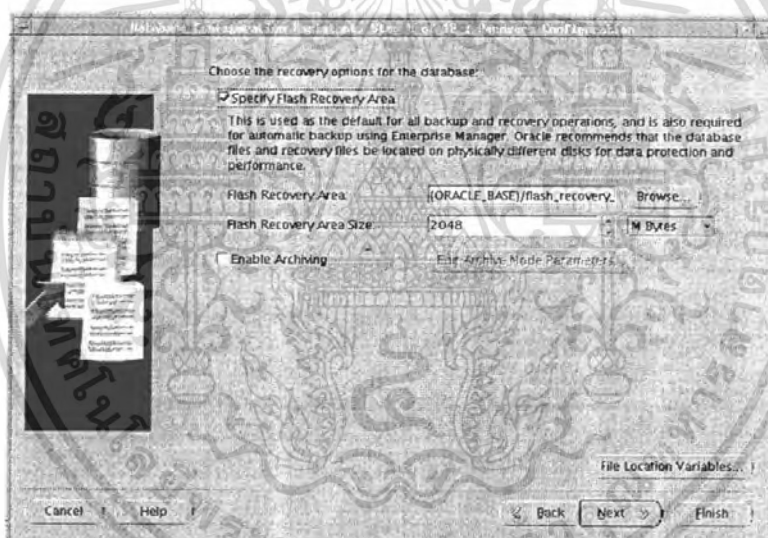


รูปที่ ก.14 หน้าจอ Storage Options เลือก File System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

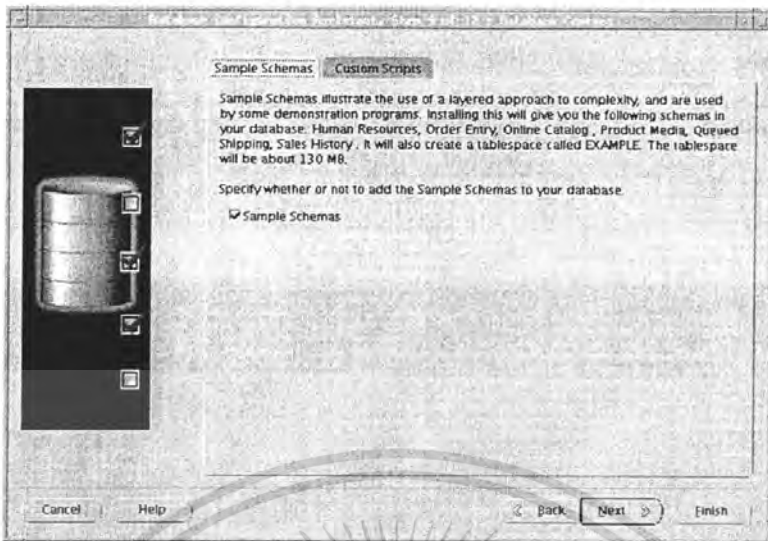


รูปที่ ก.15 หน้าจอ Database File Locations เลือก Drive, Path ที่จะกำหนดให้เก็บไฟล์ฐานข้อมูล



รูปที่ ก.16 หน้าจอ Recovery Configuration ให้เลือก Enable Flash Recovery Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

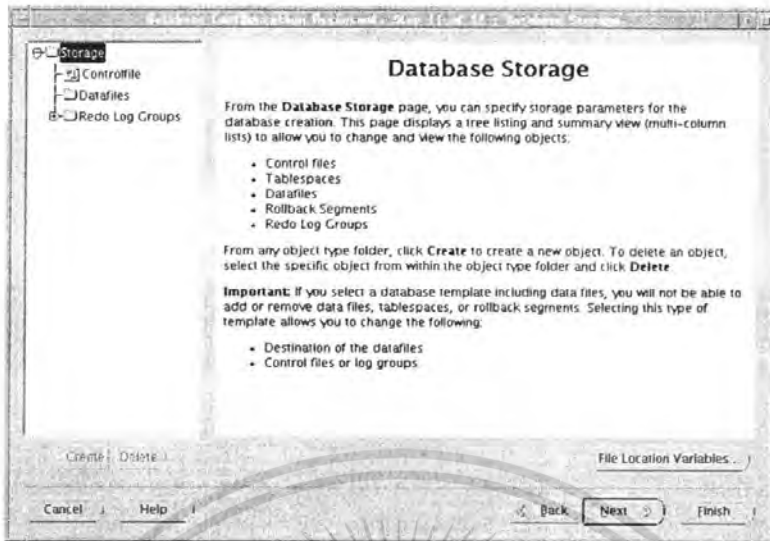


รูปที่ ก.17 หน้าจอ Database Content เลือก Sample Schemas เพื่อสร้าง Schema สำหรับในทดสอบ

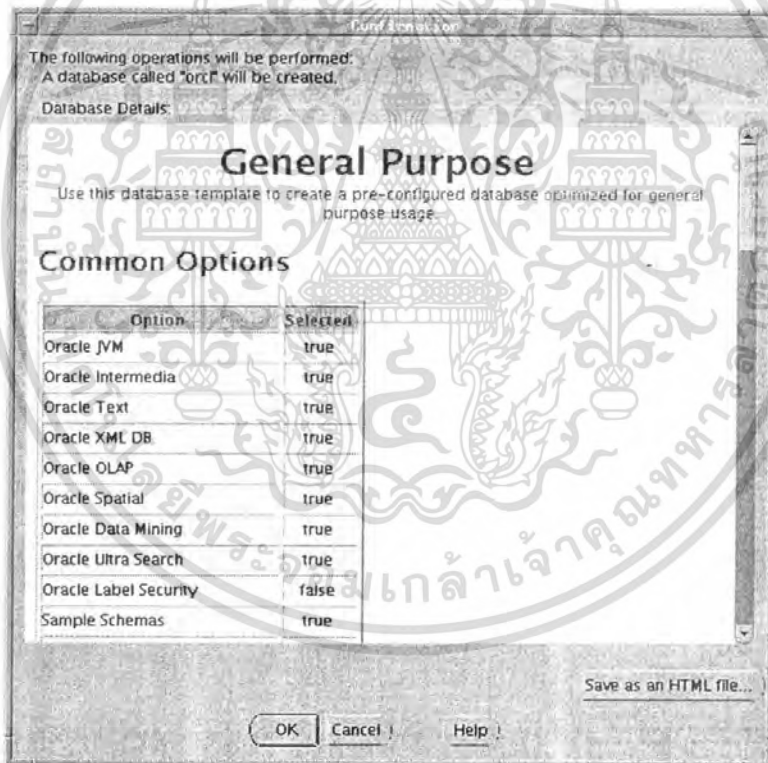


รูปที่ ก.18 หน้าจอ Initialization Parameters ให้เลือก tab Character Sets เลือก Character Set เป็น TH8TISASCII เพื่อให้สามารถใช้ภาษาไทยกับฐานข้อมูลได้อย่างไม่มีปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.19 หน้าจอ Database Storage มีสรุป Path ที่ใช้เก็บไฟล์



รูปที่ ก.20 หน้าจอ Confirmation สรุปสิ่งที่เลือกไปทั้งหมดให้ดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้