



## โครงการ เรื่อง Car repair service

เสนอ

อาจารย์ ดร. นันทวุฒิ คะอังกู

อาจารย์ศรีัญญา กาญจนวัฒนา

จัดทำโดย

นายจิรศักดิ์ การกระสัง B5902644

นางสาวเปมิกา อินจันทร์ B5903320

นายดิกิ คาวาซากิ B5904105

นายอิสราพงษ์ รัตนศักดิ์ B5904846

นางสาววรารณ คอบุญทรง B5908974

นายปองพล ปั้นประเสริฐ B5925650

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา ระบบฐานข้อมูล

(Database System) 523211

สาขาวิชาวิศวกรรมคอมพิวเตอร์

ภาคเรียนที่ 3 ปีการศึกษา 2560

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



## โครงการ เรื่อง Car repair service

เสนอ

อาจารย์ ดร. นันทวุฒิ คะอังกู

อาจารย์ศรัณญา กาญจนวัฒนา

จัดทำโดย

นายจิรศักดิ์ การกระสัง B5902644

นางสาวเปมิกา อินจันทร์ B5903320

นายดิกิ คาวาซากิ B5904105

นายอิสราพงษ์ รัตนศักดิ์ B5904846

นางสาววรารณ คอบุญทรง B5908974

นายปองพล ปั้นประเสริฐ B5925650

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา ระบบฐานข้อมูล

(Database System) 523211

สาขาวิชาวิศวกรรมคอมพิวเตอร์

ภาคเรียนที่ 3 ปีการศึกษา 2560

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

## คำนำ

รายงานวิชานี้เป็นส่วนหนึ่งของวิชา 523211 Database System โดยมีจุดประสงค์เพื่อแก้ไขปัญหาการจัดเก็บข้อมูลที่มีความยุ่งยากและซับซ้อน และเข้าถึงข้อมูลได้ง่ายมากขึ้น เพิ่มความปลอดภัยในฐานข้อมูล เพราะการเก็บข้อมูลก่อนหน้านี้มีความยุ่งยากและซับซ้อน ไม่มีความปลอดภัย และข้อมูลอาจเกิดความเสียหายได้เนื่องจากการเก็บข้อมูลที่เป็นแฟ้มหรือกระดาษ เพื่อแก้ปัญหาดังกล่าว

ผู้จัดทำได้เลือกหัวข้อรายงานเรื่อง Car repair service นี้ขึ้นเพื่อเป็นการศึกษาและการประยุกต์ความรู้เกี่ยวกับระบบฐานข้อมูล การสร้างตารางเพื่อนำมาเก็บข้อมูลและประมวลผลข้อมูล นอกจากอำนวยความสะดวกในการทำงานได้รวดเร็วแล้ว ยังมีความถูกต้องแม่นยำในการประมวลผล การจัดการฐานข้อมูลให้บริการการเข้าถึงข้อมูลได้เป็นอย่างดีมีระบบรักษาความปลอดภัยรวมทั้งการจัดการข้อมูลที่ดี

สุดท้ายนี้ขอกราบขอบพระคุณ อาจารย์ ดร. นันทูม คะอังกุ และอาจารย์ ศรัญญา กาญจนวัฒนา อาจารย์ที่ปรึกษารายงานที่ได้ให้คำแนะนำ แนวคิดตลอดจนแก้ไขข้อบกพร่องต่างๆมาโดยตลอดในการจัดทำโครงการระบบฐานข้อมูล (Database System) เรื่อง Car repair service หวังว่ารายงานฉบับนี้จะเป็นประโยชน์ต่อผู้ใช้ทุกๆ ท่าน

คณะผู้จัดทำ

## สารบัญ

	หน้า
คำนำ	ก
สารบัญ	ข
ความเป็นมา	
ที่มาและความสำคัญ	1
วัตถุประสงค์	1
ขอบเขตการศึกษา	1
ศึกษาข้อมูล	2
ER – Diagram	18
Schema Diagrams	20
การใช้งานโปรแกรม	26
บรรณานุกรม	30

## ที่มาและความสำคัญ

จากที่ได้ศึกษาระบบฐานข้อมูล Database system ซึ่งได้รู้ เกี่ยวกับการจัดการฐานข้อมูล การสร้าง ER-Diagram และแปลง ER ให้เป็น Schema เราจึงนำมาประยุกต์ใช้เข้ากับโปรเจ็ค โดยที่โปรเจ็คเริ่มมาจากการที่ผู้ปกครองได้ทำกิจการ การซ่อม รถ เราจึงนำความรู้ที่ได้มานั้น เข้ามาจัดการกับระบบฐาน ข้อมูล การนำเอาระบบคอมพิวเตอร์มาใช้งานเพื่อประมวลผล ข้อมูล นอกจากอำนวยความสะดวกในการทำงานได้รวดเร็วแล้ว ยังมีความถูกต้องแม่นยำในการประมวลผลอีกด้วย การ จัดการฐานข้อมูลให้บริการการเข้าถึงข้อมูลได้เป็นอย่างดีมี ระบบรักษาความปลอดภัยรวมทั้งการจัดการข้อมูลที่ดี เพราะ ระบบการจัดการฐานข้อมูลมีฟังก์ชันในการให้สิทธิ์เข้าถึงข้อมูล โดยบุคคลภายนอกไม่สามารถเข้าถึงข้อมูลได้ ถ้าหากไม่ได้รับ อนุญาตจากผู้ดูแลระบบ คณะผู้จัดทำจึงได้จัดทำโปรเจ็คขึ้นมา เพื่อช่วยให้การ จัดการระบบฐานข้อมูลเป็นไปอย่างมีประสิทธิภาพ และลดความ ไม่สอดคล้อง หรือข้อมูลที่ขัดแย้งกันให้น้อยลงทำให้ข้อมูลมี ความสมบูรณ์มากขึ้น

## วัตถุประสงค์

- เพื่อแก้ไขปัญหาการจัดเก็บข้อมูลที่มีความยุ่งยากซับซ้อน
- เพื่อทำให้ข้อมูลที่จัดเก็บมีความปลอดภัยมากยิ่งขึ้นและ ลดความสูญหายต่างๆของข้อมูล

## ขอบเขตการศึกษา

### ขอบเขตด้านเนื้อหา

- ศึกษาการเขียน SQL
- ศึกษาการเขียนภาษา Java

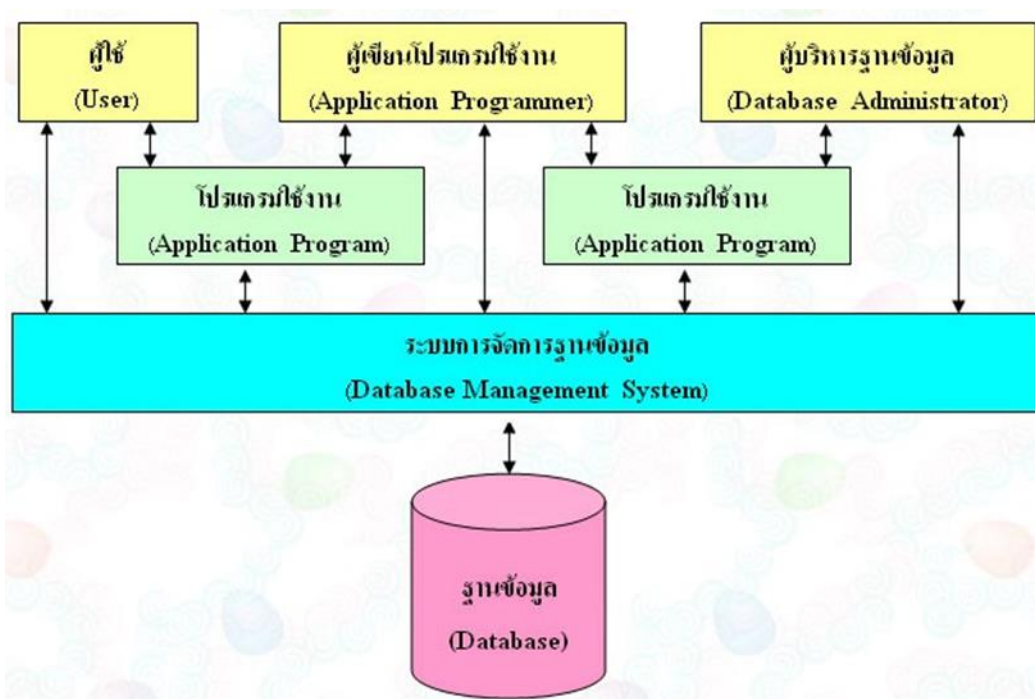
### โปรแกรมที่ใช้ในการดำเนินงาน

- โปรแกรม Oracle
- โปรแกรม NetBeans
- โปรแกรมNotepad ++

## ศึกษาข้อมูล

**ฐานข้อมูล (database)** คือ กลุ่ม ข้อมูลที่มีความสัมพันธ์กันและถูกนำมารวบรวมไว้ในที่ เดียวกัน อย่างเป็นระบบเพื่อนำไปใช้ในวัตถุประสงค์อย่างใดอย่างหนึ่ง โดยกลุ่มผู้ใช้ตั้งแต่หนึ่งกลุ่มขึ้นไป

**ข้อมูล (data)** คือ ข้อเท็จจริง (real facts) ต่างๆที่เกี่ยวข้องหรือแสดงคุณลักษณะของบุคคล สิ่งของ สถานที่ หรือ เหตุการณ์ใดๆ ที่อาจเป็นได้ทั้งตัวเลข (numeric) เช่น ราคา ปริมาณ จำนวนเงิน ส่วนสูง ระยะทาง รหัสวิชาเกรดเฉลี่ย หรือ ข้อเท็จจริงที่ไม่ใช่ตัวเลข (non-numeric) เช่น ชื่อ นามสกุล ที่อยู่ ชื่อสินค้า ข้อความ รูปภาพ หรือ อื่นๆ เป็นต้น



**ผู้ใช้งานฐานข้อมูล หรือ ผู้ที่เกี่ยวข้องกับฐานข้อมูลแบ่งได้เป็น**

1. **ผู้ใช้ (User)** หมายถึง ผู้ที่ต้องการใช้ฐานข้อมูลทั่วไป การใช้งานอาจทำได้โดยผ่านโปรแกรม ใช้งานหรือผ่านระบบการจัดการฐานข้อมูล ถ้ามีความรู้เกี่ยวกับ ระบบเพียงพอ
2. **ผู้เขียนโปรแกรมใช้งาน (Application Programmer)** หมายถึง ผู้ที่สร้างฐานข้อมูล และ พัฒนาโปรแกรมใช้งานสำหรับให้ผู้ใช้สามารถใช้งานฐานข้อมูล ได้ง่ายและให้ผู้บริหารฐานข้อมูล สามารถจัดการฐานข้อมูลได้สะดวกขึ้น
3. **ผู้บริการฐานข้อมูล (Database Administrator)** หมายถึง ผู้ที่ออกแบบฐานข้อมูลและดูแล รักษาและจัดการฐานข้อมูล ให้ปลอดภัย ทันสมัย และถูกต้องอยู่เสมอ

## ระบบการจัดการฐานข้อมูล (DatabaseManagementSystem:DBMS)

หมายถึงชุดโปรแกรมที่ใช้ในการจัดการฐานข้อมูล เช่น การสร้างฐานข้อมูล , การบันทึกข้อมูล, การปรับปรุงแก้ไขข้อมูล , การสืบค้น ข้อมูล , การวิเคราะห์ข้อมูล ,การจัดทำรายงาน และอื่น ๆ

ตัวอย่างโปรแกรมที่ใช้จัดการฐานข้อมูล ได้แก่

Microsoft Access / Microsoft SQL Server

MySQL / Oracle

## ซอฟต์แวร์ระบบฐานข้อมูล มีหลายโปรแกรม ดังนี้

1.1ซอฟต์แวร์ที่ใช้ในเครื่องคอมพิวเตอร์ขนาดเล็กที่เป็นที่รู้จักอย่างแพร่หลายคือ โปรแกรม Microsoft Access เป็นโปรแกรมหนึ่งในชุดไมโครซอฟต์ออฟฟิศ ของบริษัท ไมโครซอฟต์ จำกัด โปรแกรม FoxPro เป็นโปรแกรมที่พัฒนาจากโปรแกรม Fox Plus ปัจจุบันบริษัทไมโครซอฟต์ได้ซื้อ ลิขสิทธิ์รวมไว้ในโปรแกรมชุด Visual Studio โปรแกรม Paradox และโปรแกรม dBASE ซึ่งเป็นโปรแกรมฐานข้อมูล ยุคแรกที่ยอมรับใช้ ในเครื่องคอมพิวเตอร์ขนาดเล็ก

1.2 ซอฟต์แวร์ระบบฐานข้อมูล ขนาดใหญ่เป็นซอฟต์แวร์ที่มีสมรรถนะสูง ทำงานบน ระบบปฏิบัติการที่หลากหลายได้ เช่น โปรแกรม Oracle โปรแกรม IDMS โปรแกรม IMS/VS โปรแกรม Sybase โปรแกรม Informix เป็นต้น

**ซอฟต์แวร์ประยุกต์** นอกจากการใช้คำสั่งโดยตรงของโปรแกรมฐานข้อมูล แล้ว ยังมีภาษาที่ใช้ เฉพาะกับระบบฐานข้อมูลต่างๆ ได้แก่

2.1ภาษาSQLสามารถเขียนเป็นชุดคำสั่งไว้ในเครื่องมือของระบบฐานข้อมูลโดยตรง แล้วทำการแปล (Compile)ไปเป็นภาษาเครื่องเพื่อใช้งานได้ทันทีหรือใช้โปรแกรมเฉพาะของ ภาษา SQL ก็ได้

2.2 ภาษา MySQL เป็นภาษาสำหรับจัดการกับฐานข้อมูล ขนาดใหญ่ที่มีราคาถูก มี โปรแกรมต้นฉบับให้ผู้นำไปพัฒนาต่อได้เรียกว่าเป็น Opensource เช่นเดียวกับ ระบบปฏิบัติการลินุกซ์

## ภาษา Standard relational database Query Language (SQL)

ภาษาที่กลายเป็นภาษามาตรฐานสำหรับระบบฐานข้อมูลคือ ภาษา Standard relational database Query Language หรือเอสคิวแอล (SQL) หรือซีคิวอล (SE-QUEL) ซึ่งเป็นภาษาที่พัฒนาขึ้นมาโดยบริษัท ไอบเอ็ม ภาษา SQL (Standard Query Language) เป็นส่วนหนึ่งของระบบฐานข้อมูลแบบรีเลชันเนล (Relational Database) ที่ได้รับความนิยมมากเพราะง่ายต่อความเข้าใจ และอยู่ในรูปภาษาอังกฤษ ภาษา SQL แบ่งออกเป็น 2 ส่วน คือ

- ภาษาที่ใช้สำหรับนิยามข้อมูล (Data Definition Language-DDL)
- ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML)

ภาษาควบคุม (Control Language)

ภาษาในการเลือกข้อมูล (Data Query Language)

รูปแบบการใช้คำสั่ง SQL สามารถใช้ได้เป็น 2 รูปแบบ ดังนี้คือ

- คำสั่ง SQL ที่ใช้เรียกดูข้อมูลได้ทันที (Interactive SQL)

เป็นการเรียกใช้คำสั่ง SQL สั่งงานบนจอภาพ เพื่อเรียกดูข้อมูลในขณะที่ทำงานได้ทันทีเช่น

```
SELECT CITY  
FROM SUPPLIER  
WHERE SNO = 'SE';
```

- คำสั่ง SQL ที่ใช้เขียนร่วมกับโปรแกรมอื่น ๆ (Embedded SQL)

เป็นคำสั่ง SQL ที่ใช้ร่วมกับคำสั่งของโปรแกรมภาษาต่าง ๆ เช่น PL/1 PASCAL ฯลฯ หรือแม้แต่ว่ากับคำสั่งในโปรแกรมที่ระบบจัดการฐานข้อมูลนั้นมิใช่เฉพาะ เช่น ORACLE มี PL/SQL (Procedural Language /SQL) ที่สามารถเขียนโปรแกรมและนำคำสั่ง SQL มาเขียนร่วมด้วย เป็นต้น

ตัวอย่างการใช้คำสั่ง SQL ในภาษา PL/1

```
EXEC SQL SELECT CITY  
  
INTO :XCITY
```



FROM SUPPLIER

WHERE SNO = 'S4';

### ภาษาที่ใช้สำหรับนิยามข้อมูล (Data Definition Language-DDL )

Data Definition Language (DDL) เป็นภาษาที่ใช้นิยามโครงสร้างข้อมูล เพื่อเปลี่ยนแปลง หรือ ยกเลิกโครงสร้างฐานข้อมูลตามที่ต้องการแบบไว้โครงสร้างดังกล่าวคือ สคีมา (Schema) นั้นเอง ตัวอย่างเช่นการ กำหนดฐานข้อมูลประกอบด้วยตารางอะไรบ้าง ชื่ออะไร ประเภทใด มีอินเด็กซ์ (Index)

เอกสารประกอบการบรรยายวิชา 204204 การออกแบบและพัฒนาฐานข้อมูล 8 - 2

### ภาษา DDLประกอบด้วย 3 คำสั่งคือ

- คำสั่งการสร้าง (Create) ได้แก่ การสร้างตารางและอินเด็กซ์

CREATE TABLE <Table name>

( Attribute 1 Type 1,

Attribute 2 Type 2 ,);

CREATE Unique Index on X<Table name>

เช่น

CREATE TABLE S11

(SNO CHAR(5) Not NULL,

SNAME CHAR(10) ,

STATUS integer);

CREATE Unique Index XS11 on S11(SNO)

- คำสั่งเปลี่ยนแปลงโครงสร้าง

ALTER TABLE < ชื่อตารางที่ตั้งขึ้น >

<คำสั่งการเปลี่ยนแปลง> (<ชื่อคอลัมน์ประเภทข้อมูล>);

ตัวอย่างเช่น

```
ALTER TABLE SUPPLIER
```

```
ADD (LAST_SNAME Char(10));
```

- คำสั่งยกเลิก (Drop) ต่างๆ

การลบโครงสร้างตาราง

```
DROP TABLE < ชื่อตารางที่ตั้งขึ้น >
```

ภาษาดังกล่าวคือ ภาษาที่ใช้สร้างฐานข้อมูลลงในคอมพิวเตอร์หลังจากที่เราได้ออกแบบแล้วว่าฐานข้อมูลมีกี่รีเลชัน แต่ละรีเลชันมีความสัมพันธ์อย่างไร จากนั้นการใช้ภาษา DDL นี้แปลงรีเลชันต่างๆ ให้อยู่ในรูปภาษาสำหรับนิยามข้อมูล เพื่อป้อนเข้าสู่ระบบฐานข้อมูล เพื่อสร้างฐานข้อมูลที่แท้จริงให้เกิดขึ้นในคอมพิวเตอร์ ภาษา DDL สามารถสรุปคำสั่งต่างๆได้ดังต่อไปนี้

คำสั่ง	ความหมาย
CREATE TABLE	นิยามโครงสร้างข้อมูลในรูปตารางบนฐานข้อมูล
DROP TABLE	ลบโครงสร้างตารางข้อมูลออกจากระบบ
ALTER TABLE	แก้ไขปรับปรุงโครงสร้างตาราง
CREATE INDEX	สร้างดัชนีของตาราง
DROP INDEX	ลบดัชนีของตารางออกจากระบบ
CREATE VIEW	กำหนดโครงสร้างวิวของผู้ใช้
DROP VIEW	ลบโครงสร้างวิวออกจากระบบ

### คำสั่งนิยามโครงสร้างตาราง

การสร้างตารางใน ฐานข้อมูลแบบรีเลชันแนล โดยเฉพาะฐานข้อมูลขนาดใหญ่บนระบบ UNIX จะทำการป้อนคำสั่งในลักษณะกึ่งข้อความ (Text Mode) เข้าไปในระบบฐานข้อมูล ดังรูป แบบต่อไปนี้

```
CREATE TABLE <ชื่อตาราง >
```

```
(<ชื่อคอลัมน์ ประเภทของข้อมูล>[,<ชื่อคอลัมน์ ประเภทของข้อมูล>]....);
```

## ประเภทของข้อมูล

ประเภทของข้อมูลแบ่งเป็น 5 ประเภทใหญ่ๆขึ้นอยู่กับซอฟต์แวร์ระบบฐานข้อมูลที่ใ้ใช้คืออะไร ตัวอย่าง เช่น CHAR, INTEGER, DATE ฯลฯ

### คำสั่งการลบโครงสร้างตาราง

DROP TABLE <ชื่อตารางที่ต้องการลบ>

### คำสั่งการเปลี่ยนแปลงโครงสร้างตาราง

ในกรณีที่ต้องการเปลี่ยนแปลงโครงสร้างตารางที่เคยนิยามไว้สามารถใช้คำสั่งต่อไปนี้

ALTER TABLE <ชื่อตารางที่ต้องการเปลี่ยนแปลง>

<คำสั่งการเปลี่ยนแปลง>[,<ชื่อคอลัมน์ ประเภทของข้อมูล>]>

## คำสั่งดัชนี

ดัชนี ( INDEX ) มีความสำคัญมากต่อฐานข้อมูลเชิงสัมพันธ์ เนื่องจาก ระบบฐานข้อมูลแบบรีเลชันแนล (RDBMS) จะใช้ดัชนีในการค้นหาเงื่อนไขที่ต้องการได้อย่างรวดเร็ว โดยดัชนีที่ถูกสร้างขึ้น จะเก็บไว้แยกจากตารางในพื้นที่ต่างหากของคอมพิวเตอร์ โดยปกติ ถ้าไม่มีการประกาศดัชนีไว้การค้นหาข้อมูลในตาราง นั้น จะต้องทำแบบเรียงลำดับจากแถวที่หนึ่งจนถึงแถวสุดท้าย การสร้างดัชนีสำหรับตารางใดๆ จะทำได้โดยการเลือกคอลัมน์ใดคอลัมน์หนึ่งจากตารางมาเป็นดัชนีและตารางหนึ่งๆ สามารถมีได้หลายดัชนี

นอกจากเพิ่มความรวดเร็วในการดึงข้อมูลแล้ว ยังสามารถนำไปใช้ในการควบคุม คอลัมน์ที่นำมาสร้างเป็นดัชนีให้มีการเก็บข้อมูลที่ไม่ซ้ำกัน(Unique) อีกด้วย

การสร้างดัชนีจะใช้คำสั่ง CREATE INDEX แล้วตามด้วยชื่อดัชนีที่เราตั้งขึ้น ดังรูปแบบ

ต่อไปนี้

CREATE [UNIQUE] INDEX < ชื่อตารางที่ตั้งขึ้น >

ON (<ชื่อตารางที่สร้างดัชนี> (< ชื่อคอลัมน์\_1> [< ชื่อคอลัมน์\_2>]...);

## การลบดัชนี

เมื่อต้องการลบดัชนีที่สร้างขึ้น ก็สามารถทำได้ด้วยคำสั่ง DROP INDEX แล้วตามด้วยชื่อดัชนีที่ต้องการลบ ดังรูปแบบดังนี้

DROP INDEX <ชื่อดัชนี>

## ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language-DML)

หลังจากที่เราสร้างโครงสร้างฐานข้อมูลขึ้นแล้ว คำสั่งต่อไปในการป้อนข้อมูลลงในฐานข้อมูลและเปลี่ยนแปลงข้อมูลในฐานข้อมูล โดยการใช้ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language-DML) ใช้จัดการข้อมูลภายในตารางภายในฐานข้อมูล และภาษาแก้ไขเปลี่ยนแปลงตาราง แบ่งออกเป็น 4 Statement คือ

- Select Statement : การเรียกหา (Retrieve) ข้อมูลจาก ฐานข้อมูล
- Insert Statement : การเพิ่มเติมข้อมูลลงใน ตาราง (Table) จาก ฐานข้อมูล
- Delete Statement: การลบข้อมูลลงออกจาก ตาราง (Table) จาก ฐานข้อมูล
- Update Statement: การเปลี่ยนแปลงข้อมูลใน ตาราง (Table) จาก ฐานข้อมูล

คำสั่ง	ความหมาย
SELECT	เรียกค้นข้อมูลในตาราง
INSERT	เพิ่มแถวข้อมูลลงในตาราง

DELETE	ลบแถวข้อมูล
UPDATE	ปรับปรุงแถวข้อมูลในตาราง

### คำสั่งค้นหาข้อมูล (Query Statement)

คำสั่ง SELECT เป็นคำสั่งการเรียกดูข้อมูล หรือ ค้นหาข้อมูล ตามเงื่อนไขที่ระบุ เนื่องจากคำสั่ง SELECT เป็นคำสั่งที่มีรูปแบบการใช้งานที่ง่ายเพื่อช่วยในการค้นหาข้อมูลที่ซับซ้อน ดังมีรูปแบบดังนี้

SELECT <ชื่อคอลัมน์ที่ต้องการดูข้อมูล>

FROM <ชื่อตาราง>

WHERE <เงื่อนไขตามที่ระบุ>

SELECT --- เป็นคำสั่งให้ทำการเรียกดูข้อมูลในคอลัมน์ที่ระบุ ซึ่งอาจจะมากกว่า หนึ่งก็ได้และถ้ามีมากกว่าหนึ่งคอลัมน์ต้องคั่นด้วย คอมม่า (,) และนอกจากนี้ยังสามารถใช้เครื่องหมายดอกจัน (\*) เพื่อแสดงถึงการขอข้อมูลทั้งหมดได้อีกด้วย

FROM --- เป็นคำส่วนประกอบของคำสั่งที่บอกถึงตารางที่ต้องการดูซึ่งอาจจะมีมากกว่าหนึ่งตารางก็ได้ที่จะถูกเรียกใช้จากคำสั่ง SELECT

WHERE--- เป็นส่วนประกอบของคำสั่ง ที่ใช้บ่งบอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล ขึ้นมาจากตารางใด ๆ ที่อยู่หลัง FROM นี้

### การเรียกดูแบบซ้อนกัน (Nested SELECT Statement)

SELECT <ชื่อคอลัมน์>

FROM <ชื่อตาราง>

```
WHERE <ชื่อคอลัมน์> IN  
  
    ( SELECT <ชื่อคอลัมน์>  
  
    FROM <ชื่อตาราง>  
  
    WHERE <ชื่อคอลัมน์> )
```

### คำสั่งเติมข้อมูล (Insert Statement)

```
INSERT INTO < ชื่อตาราง >  
  
VALUES (< ชื่อคอลัมน์_1> [< ชื่อคอลัมน์_2>]...);
```

### คำสั่งแก้ไขและลบแถว (Update Statement )

```
UPDATE < ชื่อตาราง >  
  
SET <คำที่ต้องการ>  
  
WHERE <เงื่อนไข>
```

### ภาษาควบคุม (Control Language)

ใช้เป็นภาษาที่ใช้ควบคุมระบบรักษาความปลอดภัย ของฐานข้อมูล ประกอบด้วยคำสั่ง 2 คำสั่งคือ

- คำสั่ง GRANT เป็นคำสั่งที่ใช้กำหนดสิทธิให้กับผู้ใช้แต่ละคนให้มีสิทธิ์กระทำการใดกับข้อมูลเช่นการเพิ่มข้อมูล การแก้ไข หรือ การลบข้อมูลในตารางใดบ้าง

- คำสั่ง REVOKE เป็นคำสั่งให้มีการยกเลิกสิทธิ์นั้นหลังจากที่ได้ GRANT แล้ว

### ค่าบูลีน ( Boolean-Type Data)

- AND
- OR
- NOT

- =
- > หรือ >=
- < หรือ =<
- <>
- Built-In Function
- COUNT
- SUM
- AVG
- MAX
- MIN
- DISTINCT

### ชุดคำสั่ง GROUP BY, ORDER BY และ HAVING

เนื่องจากข้อมูลที่น่าสนใจมักจะเป็น ผลสรุป หรือ ข้อมูลที่ใช้ในการตัดสินใจ ภาษา SQL จึงได้ออก

แบบให้มีชุดคำสั่งพิเศษ ที่ทำหน้าที่แบ่งออกเป็นกลุ่มๆ ได้แก่

#### 1.1 ชุดคำสั่ง GROUP BY

GROUP BY เป็นคำสั่งให้มีการจัดกลุ่มแถวข้อมูลตามคอลัมน์

โดยข้อมูลที่เหมือนกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน ดังนั้นจากตัวอย่างตาราง S

```
SELECT S# , MAX( QTY)
```

```
FROM SPJ
```

GROUP BY S#

ผลลัพธ์

S#	QTY
S5	800
S2	800
S1	700
S3	500
S4	300

## 1.2 ชุดคำสั่ง ORDER BY

ORDER BY เป็นคำสั่งให้มีการเรียงลำดับข้อมูลในแถวข้อมูลตามคอลัมน์

โดยข้อมูลที่เหมือนกันจะถูกจัดเรียงจากน้อยไปมาก ถ้าไม่มีการระบุ แต่ถ้าระบุว่าเป็น DESC

จะเรียงจากมากไปน้อย ดังนั้นจากตัวอย่างต่อไปนี้

```
SELECT S# , MAX( QTY)
```

```
FROM SPJ
```

```
GROUP BY S#
```

```
ORDER BY 2 DESC
```



S#	QTY
S5	800
S2	800
S1	700
S3	500
S4	300

**หมายเหตุ** ตัวเลข 2 ที่อยู่หลัง ORDER BY จะเป็น การบอกระบบให้ทำการเรียงลำดับข้อมูลตามลำดับคอลัมน์ที่ 2 ของคำสั่ง SQL ซึ่งก็คือ MAX(QTY) นั่นเอง

### 1.3 ชุดคำสั่ง HAVING

HAVING เหมือนกับคำสั่ง WHERE ที่ต้องตามด้วยเงื่อนไข แต่ HAVING จะใช้ในกรณีที่มีการจัดการจัดกลุ่มหรือการใช้ GROUP BY

#### สรุปชุดคำสั่งมาตรฐานของ SQL

1. CREATE TABLE [[database.]owner.]table\_name

(column\_name datatype [not null | null] IDENTITY [(seed, increment)][constraint]

[, column\_name datatype [not null | null] IDENTITY [(seed, increment)]]).

[constraint] ...)

[ON segment name]

2. DELETE [FROM] table\_name

WHERE column\_name = "value"

### 3. ALTER TABLE [[<database.>]<owner.>]<table\_name>

ADD <column\_name><datatype>NULL [Canstsaint]

[WITH nocmeck]

{drop [ ]

### 4. UPDATE table\_name

SET column\_name= value

WHERE column\_name =operator\_value

### 5. CREATE VIEW View\_name AS

SELECT column

FROM table\_1 ....

WHERE table\_key\_1=table\_key\_2

## Oracle

Oracle ซอฟต์แวร์ระบบการจัดการฐานข้อมูลที่คุณควรรู้จัก ซอฟต์แวร์เป็นองค์ประกอบหนึ่งที่ช่วยขับเคลื่อนองค์กรอย่างมีประสิทธิภาพผู้ผลิตซอฟต์แวร์จึงจำเป็นต้องมีการพัฒนาเทคโนโลยีเพื่อให้องค์กรตัดสินใจเลือกซอฟต์แวร์ไปใช้งานภายในองค์กรซอฟต์แวร์ที่ได้รับความนิยมด้านการจัดการฐานข้อมูลอย่างเป็นระบบ รายงานล่าสุดจากออรากิล คอร์ปอเรชั่น ผู้ผลิตซอฟต์แวร์สำหรับองค์กรรายใหญ่สุดของโลกเปิดเผยว่าซอฟต์แวร์และบริการของออรากิลเป็นที่ยอมรับในตลาดเอเชียมากขึ้นอย่างต่อเนื่อง ทั้งระบบแอปพลิเคชันเพื่อธุรกิจ เทคโนโลยีด้านดาต้าเบส ระบบอัจฉริยะ ที่ช่วยลดค่าใช้จ่ายในการสร้างระบบกลางที่ช่วยให้เห็นการทำงานของระบบปฏิบัติการเดียวกันทั้งหมดไม่ว่าจะอยู่ส่วนไหนของโลก

ในไตรมาสแรกถูกมีผู้ประกอบการในภูมิภาคที่เลือกใช้ระบบ Oracle9i Database และ/หรือ Oracle9i Application Server และหรือ Oracle Collaboration Suite ได้แก่ Citibank, N.A. Asia Pacific Processing Center (สิงคโปร์), Compal Electronics Inc. (ไต้หวัน), กรมทางหลวง กระทรวงคมนาคม ไทย), Ministry of the Interior (ไต้หวัน), National Archives Administration (ไต้หวัน) และ Tenaga Nasional Berhad (มาเลเซีย) เนื่องจากระบบเว็บไซต์อี-บิสซิเนส และแอปพลิเคชันเพื่อสนับสนุนการทำธุรกรรม และการตัดสินใจ ต้องมีระบบแอปพลิเคชันเซิร์ฟเวอร์รองรับ และทั้ง Oracle9i Database กับ Oracle9i Application Server ก็เป็นอีกทางเลือกสำหรับองค์กรที่ต้องการยกระดับผลการดำเนินงานและเพื่อประโยชน์ในด้านค่าใช้จ่ายของระบบโครงสร้างที่สมบูรณ์แบบ

ฐานข้อมูล Oracle เป็นฐานข้อมูลที่ได้รับความนิยมและมีการใช้งานอย่างแพร่หลาย เนื่องจากเป็นฐานข้อมูลที่มีประสิทธิภาพในการทำงานสูง เหมาะกับการจัดการเก็บข้อมูลขนาดใหญ่ที่ต้องการความปลอดภัย และความมั่นคงในการใช้งาน การจัดการฐานข้อมูล Oracleให้เป็นไปอย่างถูกต้องจะช่วยให้การใช้งานฐานข้อมูลเป็นไปได้อย่างมีประสิทธิภาพมากยิ่งขึ้น หน้าที่ในการจัดการฐานข้อมูลเป็นหน้าที่หลักของผู้ดูแลระบบฐานข้อมูล หรือ DBA (Database Administrator)

ระบบจัดการฐานข้อมูล หรือ DBMS(Database Management System) ระบบฐานการจัดการฐานข้อมูล( DBMS) คือ ซอฟต์แวร์ที่ถูกสร้างขึ้น วัตถุประสงค์เพื่ออำนวยความสะดวกต่อผู้ใช้ฐานข้อมูล ทำหน้าที่สร้างฐานข้อมูล แก้ไขฐานข้อมูล การเข้าถึงข้อมูล การเรียกคืนข้อมูล เป็นต้น ปรับปรุง และใช้ฐานข้อมูล ซึ่งในปัจจุบันมีการพัฒนาซอฟต์แวร์มาเพื่อทำหน้าที่ต่างๆของ DBMS เช่น ,SQL Serverเวอร์ชัน 7.0, 2000 ,Oracel ,dBASE, FOXPRO, MICROSOFT ACCESS, INFORMIX, เป็นต้น

## ลักษณะงาน ซอฟต์แวร์ของ Oracle มี 2 แบบ คือ

1. Personal Oracle
2. Oracle Server

ทั้ง 2 แบบนี้มีลักษณะการใช้งานและคำสั่งเหมือนกัน แตกต่างกันที่ Personal Oracle คือฐานข้อมูลเมื่อติดตั้งบนเครื่องคอมพิวเตอร์แล้ว ผู้ใช้จะต้องนั่งทำงานกับ Oracle นี้ที่หน้าเครื่องเท่านั้น ส่วนของ Oracle Server คือ ฐานข้อมูลในลักษณะเซิร์ฟเวอร์ คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมของ Oracle Server ไว้ และยอมให้ผู้ใช้ฐานข้อมูล หรือจัดการกับข้อมูลจากเครื่องคอมพิวเตอร์อื่น ๆ ได้(เรียกเครื่องอื่นๆ เป็นไคลเอนต์) ดังนั้นถ้าต้องการให้มีผู้ใช้ข้อมูลจากฐานข้อมูลหลาย ๆ คนได้ ก็ควรต้องเลือกแบบที่เป็น Oracle Server

## การทำงานของ Oracle

ในการติดต่อใช้งานกับ Oracle Database นั้น เราต้องใช้ภาษา SQL (ซึ่งบางท่านเรียกว่า SEQUEL) ซึ่งเป็นภาษาที่ใช้ในการกำหนด และจัดการกับ Database (DDL,DML) การทำงานกับ Database ในแบบ Relational Database นั้นหมายความว่า Database Engine จะมีการจัดเก็บข้อมูล ในลักษณะที่เป็นกลุ่มของข้อมูลที่มีความสัมพันธ์กัน ใน 1 Database สามารถมีได้หลาย Column หลาย Row

**โดยปกติแล้ว Oracle Database Software จะแบ่งออกเป็น 4 ประเภทด้วยกัน คือ**

- Oracle Enterprise Edition จะเป็น Edition ที่มีความสามารถครบที่สุด มี Feature ตามที่ Oracle โฆษณาไว้ทุกอย่าง
- Oracle Standard Edition จะมีความสามารถน้อยกว่า Enterprise Edition ซึ่งถูกออกแบบเพื่อรองรับงานระดับ Department โดยความสามารถที่ขาดหายไปจาก Enterprise Edition ก็อย่างเช่น การทำ Parallel Processing และ Function ที่เกี่ยวกับ Data Warehousing เป็นต้น
- Oracle Personal Edition จะมีความสามารถใกล้เคียงกับ Enterprise Edition เกือบทุกอย่าง เพียงแต่ว่า Edition นี้ถูกออกแบบให้ใช้งาน Database ได้เพียงคนเดียว
- Oracle Lite ถูกออกแบบเพื่อให้ใช้งานกับอุปกรณ์ประเภท Mobile อย่าง Palm หรือ Windows CE เป็นต้น

## NetBeans IDE

เน็ตบีนส์ (NetBeans) เป็นเครื่องมือสำหรับนักโปรแกรมเมอร์ที่จะใช้พัฒนา Application ด้วยภาษาจาวา ในปี ค.ศ. 1998 ได้มีกลุ่มนักศึกษา "rock solid software" ได้พัฒนาซอฟต์แวร์ขึ้นมาตัวหนึ่ง ที่จะใช้ในการพัฒนา Application ด้วยภาษาจาวา เป็นโปรเจกต์นักศึกษา โดยตั้งชื่อว่า NetBeans และได้เผยแพร่ให้โปรแกรมเมอร์และบุคคลทั่วไปนำไปใช้งานได้ฟรีในรูปแบบ Opensource software ต่อมาในปี ค.ศ. 2000 บริษัทซัน ไมโครซิสเต็มส์ ผู้พัฒนาภาษาจาวา ได้เข้ามาเป็นผู้สนับสนุนหลักในการพัฒนา NetBeans และได้ทำออกมาในรูปแบบของ Opensource software โดยผู้ใช้งานไม่จำเป็นต้องเสียเงิน เพื่อซื้อมาใช้งาน และยังได้เปิดเผย Source code ให้ผู้สนใจและนักพัฒนานำไปดัดแปลง แก้ไข ตามกฎของ Opensource ปัจจุบันมีนักโปรแกรมเมอร์ทั่วโลกต่างช่วยกันพัฒนา NetBeans ให้มีความสามารถสูงขึ้น

ปัจจุบัน NetBeans IDE ได้รับความนิยมมากยิ่งขึ้น และได้รับการพัฒนาให้มีความสามารถสูงขึ้นเรื่อยๆ นอกจากจะใช้ในการพัฒนา Application ด้วยภาษาจาวาแล้ว ยังสามารถพัฒนาอื่นๆได้อีกหลากหลายโดย

ติดตั้งโปรแกรมเสริม (Add-on) ได้จากเว็บไซต์ หรือผ่านตัวอัปเดตเซนเตอร์ (Update Center) ของ NetBeans เช่น ภาษาซี/ซีพลัสพลัส (C/C++), Ruby, UML, SOA, Web Application, Java EE, Mobility (Java ME), Java FX, Java Script, PHP เป็นต้น ในเวอร์ชัน 6.0 เป็นต้นไปมีการรวมโปรแกรมเสริมต่างๆ ที่สำคัญไว้ในตัวติดตั้งของ NetBeans โดยสามารถเลือกติดตั้งได้ภายหลัง

## Entity Relationship diagram (ER – Diagram)

ER Diagram คือ แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพ การอธิบายโครงสร้างและความสัมพันธ์ของข้อมูล (Relationship) ประกอบด้วย

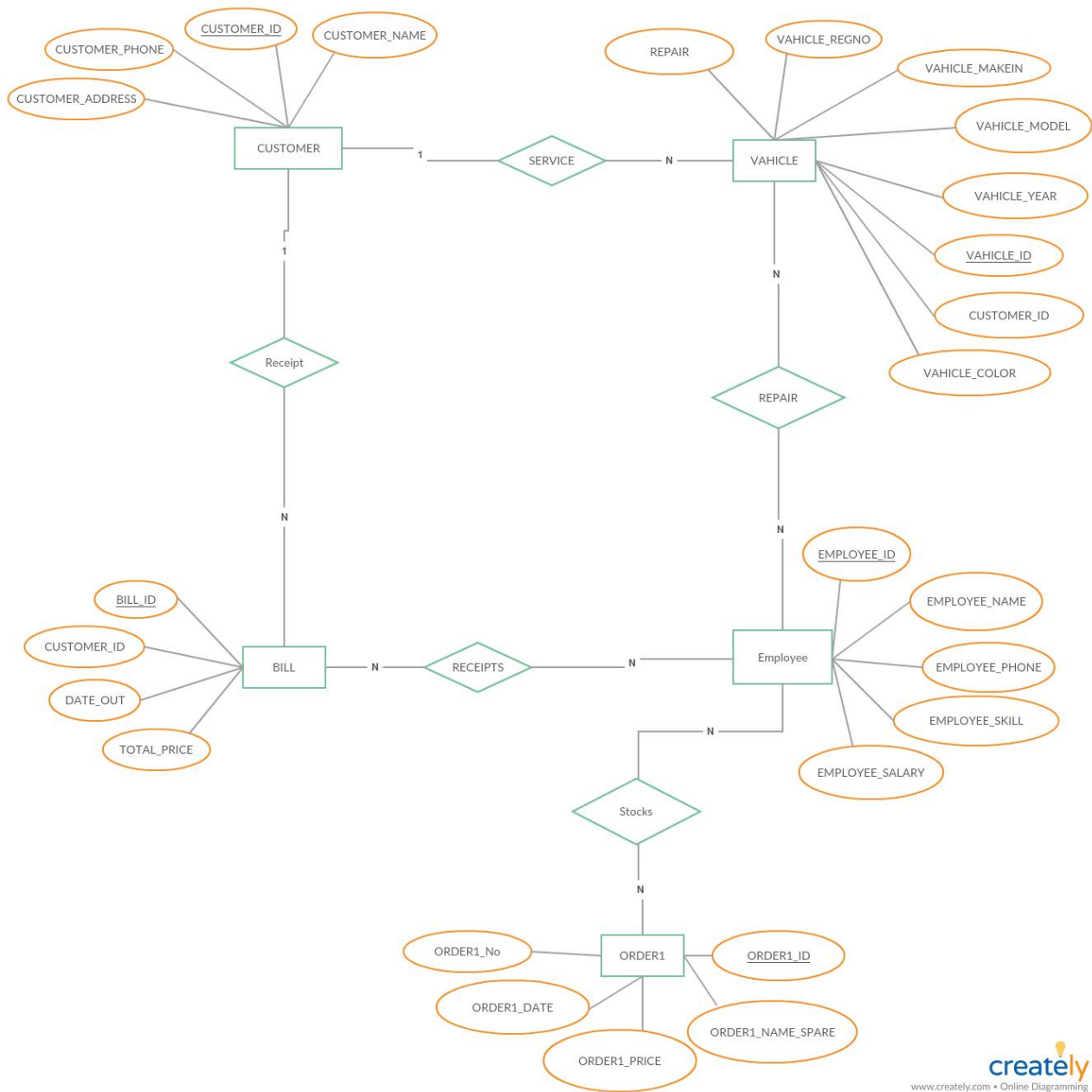
- เอนทิตี (Entity) เป็นวัตถุ หรือสิ่งของที่เราสงสัยในระบบงานนั้น ๆ
- แอททริบิว (Attribute) เป็นคุณสมบัติของวัตถุที่เราสงสัย
- ความสัมพันธ์ (Relationship) คือ ความสัมพันธ์ระหว่างเอนทิตี

ER Diagram มีความสำคัญต่อการพัฒนาระบบงานฐานข้อมูล Application ต่างๆ ที่ต้องการการเก็บข้อมูลอย่างมีระบบ มีโครงสร้าง ดังนั้น ER Diagram จึงใช้เพื่อเป็นเอกสารในการสื่อสารระหว่าง นักออกแบบระบบ และนักพัฒนาระบบ เพื่อให้สื่อสารอย่างตรงกัน และเป็นสากลอีกด้วย

### การเขียนโครงสร้าง ER – Diagram

1. หา entity จากผู้ใช้งานที่ต้องการเก็บข้อมูลอะไร
2. กำหนด attribute ให้ entity แต่ละตัวเท่าที่จำเป็น โดยกำหนด attribute หนึ่งตัวเป็น primary key โดยดูจากคุณสมบัติของ attribute
3. พิจารณาความสัมพันธ์ รูปแบบความสัมพันธ์ระหว่าง entity ที่มี

## ER – Diagram



## ความสัมพันธ์ (Relationship)

คือ ความสัมพันธ์ระหว่างเอนทิตี

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง ( One - to - One Relationship (1:1))

เป็นการแสดงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่า มีความสัมพันธ์กับข้อมูลอย่างมากหนึ่งข้อมูลกับอีกเอนทิตีหนึ่งในลักษณะที่เป็นหนึ่งต่อหนึ่ง เช่น เอนทิตีนักศึกษา กับเอนทิตีโครงการวิจัยมีความสัมพันธ์กันแบบหนึ่งต่อหนึ่ง คือ นักศึกษาแต่ละคนทำโครงการวิจัยได้ 1 โครงการเท่านั้น และแต่ละโครงการวิจัยมีนักศึกษารับผิดชอบได้ไม่เกิน 1 คน เป็นต้น

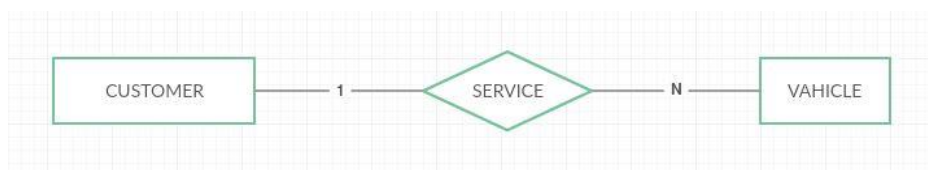
- **ความสัมพันธ์แบบหนึ่งต่อกลุ่ม ( One - to - Many Relationship (1:M) หรือ (1:N))**

เป็นการแสดงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่ามีความสัมพันธ์กับข้อมูลหลายข้อมูลกับอีกเอนทิตีหนึ่ง เช่น ความสัมพันธ์ของลูกค้าและคำสั่งซื้อเป็นแบบหนึ่งต่อกลุ่ม คือ ลูกค้าแต่ละคนสามารถสั่งซื้อได้หลายคำสั่งซื้อ แต่แต่ละคำสั่งซื้อมาจากลูกค้าเพียงคนเดียว เป็นต้น

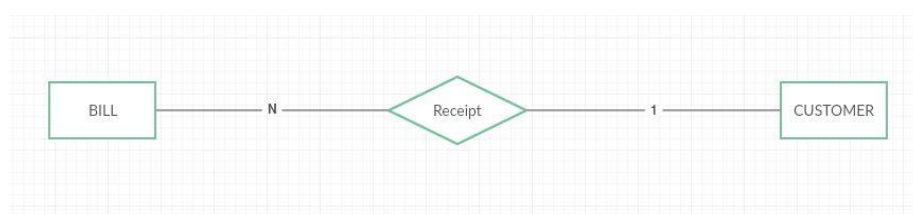
- **ความสัมพันธ์แบบกลุ่มต่อกลุ่ม ( Many - to - Many Relationship (M:M) หรือ (M:N))**

เป็นการแสดงความสัมพันธ์ของข้อมูลของสองเอนทิตีในลักษณะแบบกลุ่มต่อกลุ่ม เช่น ความสัมพันธ์ระหว่างคำสั่งซื้อกับสินค้าเป็นแบบกลุ่มต่อกลุ่ม คือ แต่ละคำสั่งซื้ออาจสั่งซื้อสินค้าได้มากกว่า 1 ชนิด และในสินค้าแต่ละชนิดอาจปรากฏอยู่ในคำสั่งซื้อได้มากกว่า 1 คำสั่งซื้อ

### ความสัมพันธ์ใน ER – Diagram

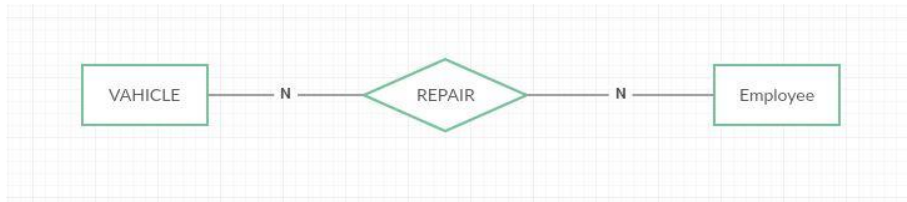


- entity customer มีความสัมพันธ์แบบหนึ่งต่อหลายกับ entity vehicle โดยระบุว่า ยานพาหนะเป็นของลูกค้าท่านไหน

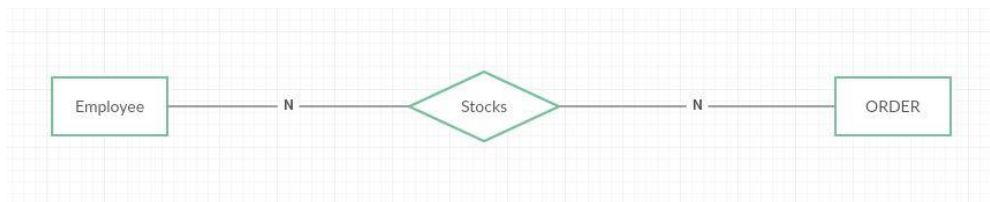




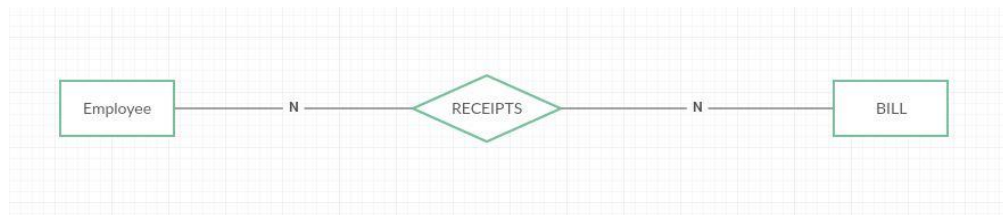
- entity bill มีความสัมพันธ์แบบหลายต่อหนึ่งกับ entity customer โดยระบุว่าลูกค้าสามารถมีบิลได้หลายบิล



- entity vehicle มีความสัมพันธ์แบบหลายต่อหลายกับ entity employee โดยระบุว่ายานพาหนะของลูกค้าที่นำมาใช้บริการอยู่ในความดูแลของพนักงานคนไหน



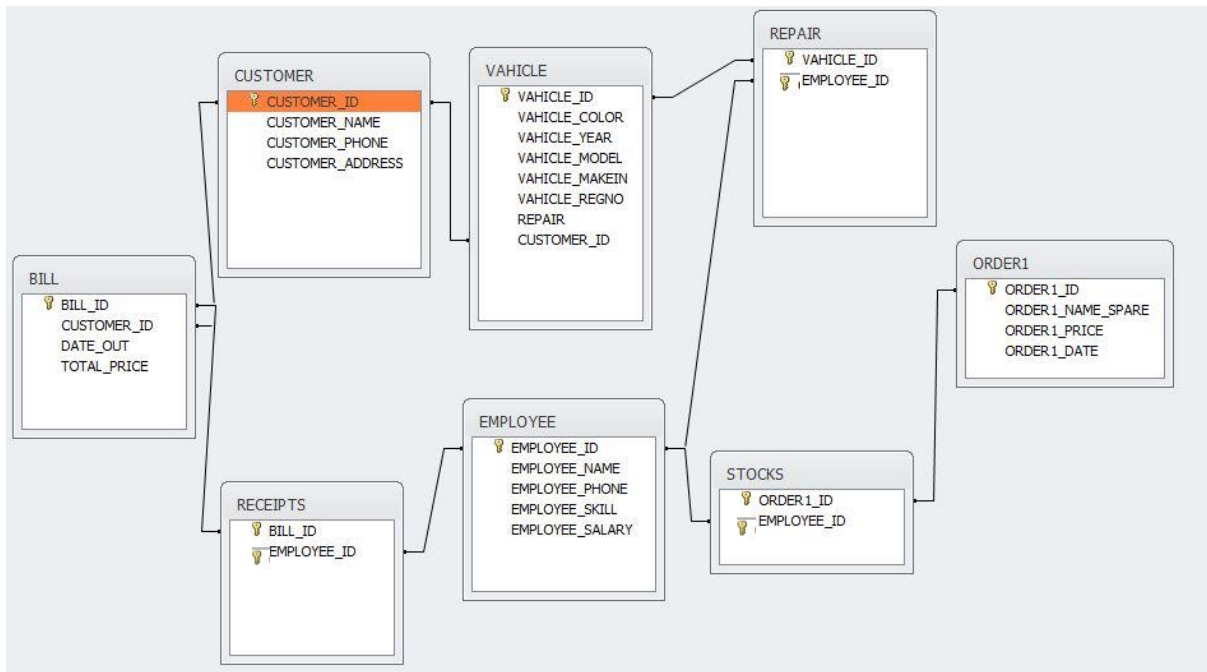
- entity employee มีความสัมพันธ์แบบหลายต่อหลายกับ entity order โดยระบุว่าพนักงานคนใดเป็นผู้สั่งซื้ออุปกรณ์การซ่อม



- entity employee มีความสัมพันธ์แบบหลายต่อหลายกับ entity bill โดยระบุว่าพนักงานออกบิลได้

## Schema Diagrams

นำ ER – Diagram มาแปลงเป็น Schema




### ตาราง Customer

Entity Customer เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute customer\_id เป็น Primary Key ของตาราง โดยใช้เก็บข้อมูลรหัสลูกค้า และระบุรายละเอียดของลูกค้าหรือ Attribute อื่นๆในตารางดังนี้ customer\_name ใช้เก็บชื่อ-นามสกุลของลูกค้า, customer\_phone ใช้เก็บหมายเลขโทรศัพท์ของลูกค้า, customer\_address ใช้เก็บที่อยู่ของลูกค้า

EDIT	CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_PHONE	CUSTOMER_ADDRESS
	B5904105	dikinakub	083-4318890	180 m.5 korat
	B5902644	jirasak	099-8720563	91 m.1 korat
	B5908974	waraphorn	063-3606456	3/2 m.3 saraburee
row(s) 1 - 3 of 3				

## ตาราง Vehicle

Entity Vehicle เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute vehicle\_id เป็น Primary Key ของตาราง โดยใช้เก็บ id ยานพาหนะ ของลูกค้าและยังสามารถระบุ attribute อื่นๆของตารางได้ดังนี้ vehicle\_color ที่ใช้เก็บสีของยานพาหนะ, vehicle\_year ที่ใช้เก็บปีที่ผลิตของรถ, vehicle\_Model ที่ใช้ระบุประเภทของยานพาหนะ, vehicle\_makein ประเทศที่ผลิตยานพาหนะ, vehicle\_regno เก็บหมายเลขเครื่องของยานพาหนะ, repair เก็บรายการการซ่อมว่ามีส่วนไหนที่ซ่อมบ้าง และยังมี Foreign Key จากตาราง customer ก็คือ customer\_id ที่ใช้ระบุว่ายานพาหนะเป็นของลูกค้าท่านใด

EDIT	CUSTOMER_ID	VEHICLE_ID	VEHICLE_COLOR	VEHICLE_YEAR	VEHICLE_MODEL	VEHICLE_MAKEIN	VEHICLE_REGNO	REPAIR
	B5904105	1	red	2018	car	thailand	ii2850	4 rubber tires
row(s) 1 - 1 of 1								

## ตาราง Employee

Entity Employee เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute employee\_id เป็น Primary Key ของตาราง โดยใช้เก็บ id ของพนักงานและยังสามารถระบุ attribute อื่นๆของตารางได้ดังนี้ employee\_name เก็บชื่อ-นามสกุล ของพนักงาน, employee\_phone เก็บหมายเลขโทรศัพท์ของพนักงาน, employee\_skill ที่ใช้ระบุความสามารถในการทำงานของพนักงาน และ employee\_salary ใช้

EDIT	EMPLOYEE_ID	EMPLOYEE_NAME	EMPLOYEE_PHONE	EMPLOYEE_SKILL	EMPLOYEE_SALARY
	B12345	dikiemployee	08123456789	employeeskill	20000
row(s) 1 - 1 of 1					

ระบุเงินเดือนของพนักงาน

## ตาราง Stocks

Entity Stocks เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute employee\_id และ Order1\_id เป็น Primary Key ของตาราง ซึ่ง attribute ทั้งสองก็ยังเป็น Foreign Key อีกด้วย

EDIT	ORDER1_ID	EMPLOYEE_ID
	001	B12345
	002	B12345
row(s) 1 - 2 of 2		

### ตาราง Order1

Entity Order1 เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute Order1\_id เป็น Primary Key ของตาราง โดยใช้เก็บ id ของรายการที่ส่งอุปกรณ์การซ่อมรถ และสามารถระบุ attribute อื่นๆของตารางได้ดังนี้ order1\_name\_spare เก็บชื่ออุปกรณ์ที่สั่งซื้อ , order1\_pricr เก็บราคาของอุปกรณ์ที่สั่งซื้อ, order1\_date ระบุวันที่สั่งซื้อ , order1\_No ระบุจำนวนอุปกรณ์ที่สั่งซื้อไปจำนวนกี่ชิ้น

EDIT	ORDER1_ID	ORDER1_NAME_SPARE	ORDER1_PRICE	ORDER1_DATE	ORDER1_NO
	001	brakes	1000	01-12-2018	0
	002	front brakes	2000	02-12-2018	0
	003	rear brakes	1500	03-12-2018	0
	004	ABS brakes/Anti-Lock brakes	10000	04-12-2018	0
	005	brake shoes	2000	05-12-2018	0
	006	disk brake pads	12000	05-12-2018	0
	007	gear box	10000	05-12-2018	0
	008	car glasss	11000	05-12-2018	0
	009	wing mirror	5000	06-12-2018	0
	0010	windscreen	5000	07-12-2018	0
	0011	car rear windscreen	4000	08-12-2018	0
	0012	rear view mirror	4500	10-12-2018	0
	0013	front door glass	8000	12-2-2018	0
	0014	rear door glass	900	01-2-2018	0
	0015	manual gear	8000	01-01-2018	0
row(s) 1 - 15 of 63 >					

### ตาราง Receipts



Entity Receipts เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute employee\_id และ bill\_id เป็น Primary Key ของตาราง และทั้งสอง attribute ยังเป็น Foreign Key อีกด้วยโดย จะใช้เก็บว่าพนักงาน

EDIT	BILL_ID	EMPLOYEE_ID
	Billid	B12345
row(s) 1 - 1 of 1		

เป็นผู้ออกบิลให้ลูกค้า

### ตาราง Bill

Entity Order1 เมื่อนำไปสร้างตารางการเก็บข้อมูลก็จะมี attribute bill\_id เป็น Primary Key ของตาราง โดยใช้เก็บ id ของรายการที่ซ่อมและราคาค่าใช้จ่ายที่ลูกค้าต้องจ่าย โดยสามารถระบุ attribute อื่นๆ ของตารางได้ดังนี้ date\_out ระบุวันที่ออกบิล , total\_price ระบุราคาการซ่อมทั้งหมด และยังมี Foreign Key จากตาราง customer คือ customer\_id เพื่อระบุว่าเป็นบิลของลูกค้าท่านใด

EDIT	BILL_ID	CUSTOMER_ID	DATE_OUT	TOTAL_PRICE
	Billid	B5904105	01/12/2555	12345
	GGGGG	B5904105	02/11/2556	12345
row(s) 1 - 2 of 2				

## การใช้งานโปรแกรม

## หน้า Customer

Customer Detail Create User Employee Vehicle Orders

## Customer Details

User ID ใส่ ID ลูกค้า ENTER CLEAR

User_ID	Name	telephone	Address

### Detail Repairs

Name	Price

แล้วจะแสดงข้อมูลของลูกค้าออกมา

Customer Detail Create User Employee Vehicle Orders

## Customer Details

User ID B590410 ENTER CLEAR

User_ID	Name	telephone	Address

Message X  
i INVALID USER  
OK

ถ้าไม่มีข้อมูลในระบบจะแสดง INVALID USER



Customer Detail **Create User** Employee Vehicle Orders

## Create User ID

User ID  check

Name

Telephone

Address

**CREATE** **UPDATE** **CLEAR**

สร้างไอดี แก้ไขข้อมูล ลบข้อมูลที่อยู่ในช่องทั้งหมด

Customer Detail **Create User** Employee Vehicle Orders

## Create User ID

User ID  check

Name

Telephone

Address

**CREATE** **UPDATE** **CLEAR**

ถ้าใส่ข้อมูลครบ แล้วกดสร้างไอดีสำเร็จ จะแสดง SUCCESS

Message

**i** SUCCESS !!

OK

## หน้า Employee

Customer Detail Create User Employee Vehicle Orders

## Employee ID

ใส่ ID ของพนักงาน แล้วกด ENTER จะแสดง  
ข้อมูลของพนักงานที่อยู่ในระบบออกมา

Employee ID

### CREATE EMPLOYEE ID

Employee ID

Name

Telephone

SKILL

Salary

สร้าง Employee พนักงาน ปุ่มแก้ไขข้อมูล ลบข้อมูลที่อยู่ในช่องทั้งหมด

Customer Detail Create User Employee Vehicle Orders

## Employee ID

Employee ID

### CREATE EMPLOYEE ID

Employee ID

Name

Telephone

SKILL

Salary

Message

**SUCCESS !!**

OK

ในขณะเดียวกันก็สามารถ  
สร้าง Employee ID ใน  
หน้าต่างนี้ได้เลย



หน้า Vehicle ใส่ข้อมูลยานพาหนะของลูกค้า

Customer Detail Create User Employee **Vehicle** Orders

**Vehicle** User ID ใส่ ID ลูกค้า check

Color Vehicle สีของรถ

Year ปีที่รถผลิตหรือปีที่เราซื้อรถมา

Model ประเภทรถ

Made in สถานที่รถผลิต

Reg No. ป้ายทะเบียนรถ

Repairs: รายละเอียดการซ่อม

Price ราคาในการซ่อม

ENTER UPDATE CLEAR

หน้า Order

Customer Detail Create User Employee Vehicle **Orders**

**ORDER SPARES** จำนวน

Name Spare ชื่อของอะไหล่สามารถกดเลือกได้ -10

price ราคา

Order Date 01 01 2018

เพิ่มอะไหล่เข้าระบบ แสดงข้อมูลในระบบ แก้ไขข้อมูล ปิดโปรแกรม

CLEAR SHOW UPDATE CANCEL

id	name	price	date	EA
001	brakes	2000	01-01-2018	-3
002	front brakes	2000	02-12-2018	5
003	rear brakes	154184	03-03-2021	6
004	ABS brakes...	10000	04-12-2018	5
005	brake shoes	2000	05-12-2018	15
006	disk brake ...	12000	05-12-2018	20
007	gear box	10000	05-12-2018	20
008	car glasss	11000	05-12-2018	10
009	wing mirror	5000	06-12-2018	11
0010	windscreen	5000	07-12-2018	12
0011	car rear wi...	4000	08-12-2018	2
0012	rear view ...	4500	10-12-2018	1
0013	front door ...	8000	12-2-2018	1
0014	rear door ...	900	01-2-2018	3
0015	manual gear	8000	01-01-2018	5

### บรรณานุกรม

- ❖ ศิริลักษณ์โรจน์กิจอานวย, ระบบฐานข้อมูล, พิมพ์ครั้งที่ 3, ดวงกลมสมัย : กรุงเทพฯ, 2542
- ❖ ชนวัฒน์ศรีสอาน, การออกแบบและพัฒนาฐานข้อมูล, มหาวิทยาลัยเทคโนโลยีสุรนารี : นครราชสีมา, 2542.
- ❖ ศิริภัทรา เหมือนมาลัย. ระบบฐานข้อมูล. กรุงเทพฯ: เอมพันธ์, 2547
- ❖ จิราพร เรืองทอง .ซอฟต์แวร์ Oracle . [ออนไลน์].เข้าถึงได้จาก  
:https://www.gotoknow.org/posts/387489. (วันที่สืบค้นข้อมูล : 14 มิถุนายน 2561)
- ❖ วิกิพีเดีย สารานุกรมเสรี. Oracle. [ออนไลน์].เข้าถึงได้จาก :  
https://th.wikipedia.org/wiki/Oracle. (วันที่สืบค้นข้อมูล : 14 มิถุนายน 2561)
- ❖ Suphakit Annopornchai. Database Oracle. [ออนไลน์].เข้าถึงได้จาก  
:https://saixiii.com/what-is-oracle/. (วันที่สืบค้นข้อมูล : 14 มิถุนายน 2561)
- ❖ วิกิพีเดีย สารานุกรมเสรี. เน็ตบีนส์. [ออนไลน์].เข้าถึงได้จาก  
:https://th.wikipedia.org/wiki/%E0%B9%80%E0%B8%99%E0%B9%87%E0%B8%99%E0%B8%9A%E0%B8%B5%E0%B8%99%E0%B8%AA%E0%B9%8C.(วันที่  
สืบค้นข้อมูล : 14 มิถุนายน 2561)