

# Part

15

Java Programming Language  
Mr.Rungrote Phonkam  
rungrote@it.kmitl.ac.th



# Contents

1. Relational Database
2. SQL



# 1. Relational Database

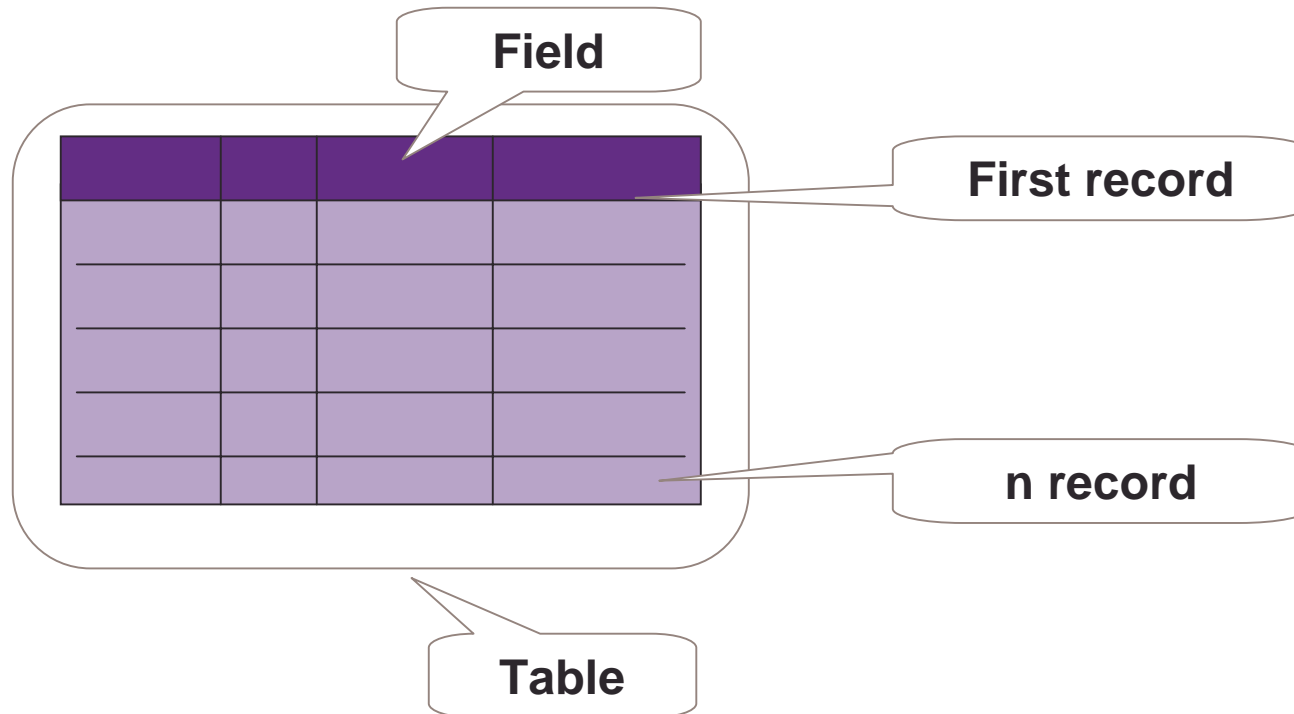
## Relational Database

- A Database consists of more tables
- A table consists of one or more records
- A record contains data in any fields
- Each table related by data in any fields
- Relation between tables is 1:1, 1:M, or M:M



# 1. Relational Database

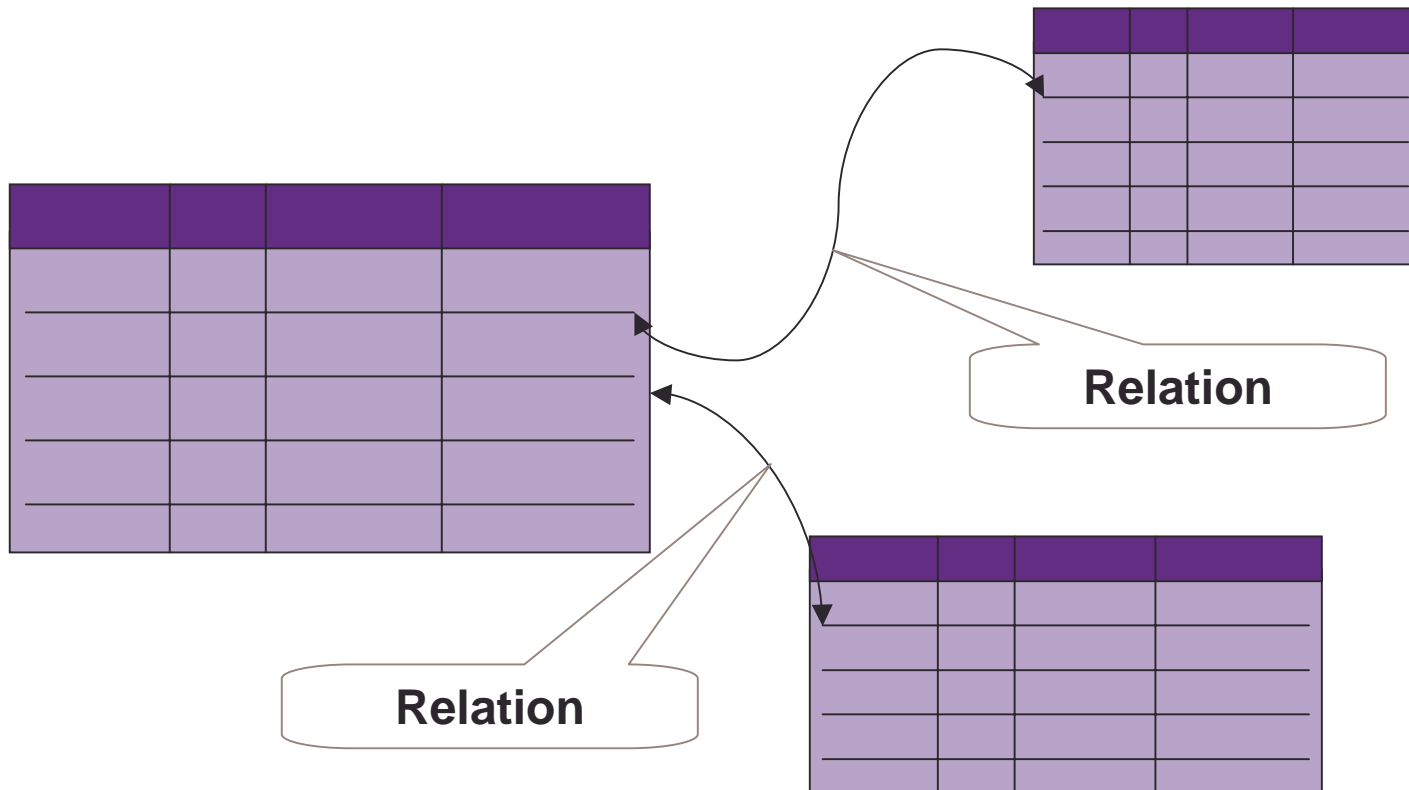
## Relational Database





# 1. Relational Database

## Relational Database





## 2. SQL

เทเบิล Products

ชื่อฟิลด์

ProductID	TEXT(5)	Primary Key
Name	TEXT(30)	NOT NULL
Price	FLOAT	
UnitID	TEXT(2)	Reference Key ของฟิลด์ UnitID ในเทเบิล Units

เทเบิล Units

ชื่อฟิลด์

UnitID	TEXT(2)	Primary Key
Description	TEXT(30)	NOT NULL



## 2. SQL

การสร้างเทเบิล

รูปแบบ

```
CREATE TABLE TableName {  
    FieldName1    DataType (Size)        Constraint,  
    FieldName2    DataType (Size)        Constraint  
};
```

เมื่อ

TableName คือชื่อเทเบิลที่ต้องการสร้าง

FieldName1, ... คือชื่อฟิลด์ที่ต้องการสร้างไว้ในเทเบิล

DataType คือชนิดข้อมูลที่กำหนดให้กับ

Size คือขนาดที่กำหนดให้กับข้อมูล

Constraint คือข้อกำหนดบังคับ ซึ่งจะมีผลในการใช้งานเมื่อนำข้อมูลใส่ลงในเทเบิล



## 2. SQL

### Data Type

CHAR(x)	ใช้สำหรับกำหนดข้อมูลตัวอักษร โดย x คือจำนวนตัวอักษร
INTEGER	ใช้สำหรับกำหนดข้อมูลตัวเลขจำนวนเต็ม
DECIMAL(x, y)	ใช้สำหรับกำหนดข้อมูลตัวเลขทศนิยม โดย x คือจำนวนหลักทั้งหมด y คือจำนวนหลักทศนิยม
DATE	ใช้สำหรับกำหนดข้อมูลประเภทวันที่
LOGICAL	ใช้สำหรับกำหนดข้อมูลตรรกะ เช่น TRUE หรือ FALSE





## 2. SQL

### Constraint

NULL หมายถึงยินยอมให้ข้อมูลในฟิลด์เป็นค่า null

NOT NULL หมายถึงไม่ยินยอมให้ข้อมูลในฟิลด์เป็นค่า null

PRIMARY KEY หมายถึงกำหนดให้ฟิลด์มีคุณสมบัติเป็นคีย์หลัก

CHECK หมายถึงให้มีการตรวจสอบข้อมูล

เช่น CHECK ( Price > 0 ) คือการยินยอมให้ใส่ข้อมูลที่มีค่ามากกว่า 0 ลงในฟิลด์ Price

DEFAULE หมายถึงการกำหนดค่าเริ่มต้นให้กับฟิลด์ในข้อมูลในกรณีที่ไม่ได้ระบุข้อมูลให้กับฟิลด์นั้นๆ

REFERENCE หมายถึงการระบบคุณสมบัติของฟิลด์ให้เป็นแบบคีย์อ้างอิง โดยมีรูปแบบคือ

REFERENCE TableName ( FieldName )



## 2. SQL

การสร้างเทเบิล Products

```
CREATE TABLE Products (
```

```
    ProductID      CHAR(5)      PRIMARY KEY,  
    Name           CHAR(30)     NOT NULL,  
    Price          NUMBER(8, 2) CHECK > 0,  
    UnitID         CHAR(2)     REFERENCE Units(UnitID)  
)
```

การสร้างเทเบิล Units

```
CREATE TABLE Units (
```

```
    UnitID         CHAR (2)    PRIMARY KEY,  
    Description    CHAR(30)   NOT NULL  
)
```



## 2. SQL

การลบเทเบิล

รูปแบบ

```
DELETE TABLE TableName ;
```

การเปลี่ยนแปลงเทเบิล

รูปแบบ

```
ALTER TABLE TableName ADD/DROP/MODIFY (  
    FieldName1 DataType (Size) Constraint,  
    FieldName2 DataType (Size) Constraint  
);
```



## การแทรกเรคอร์ดในเทเบิล

รูปแบบ

```
INSERT INTO TableName ( FieldName, ... )  
VALUES ( Value, ... ) ;
```

เมื่อ

TableName	คือชื่อเทเบิลที่ต้องการลบ
FieldName	คือชื่อฟิลด์ขอเทเบิล ต้องใช้ชื่อที่มีอยู่จริงในเทเบิล
Values	คือข้อมูลที่ตรงกับฟิลด์ในตำแหน่งเดียวกัน



## 2. SQL

### ตัวอย่าง

- 1) 

```
INSERT INTO Products(ProductID, Name, Price, UnitID)
VALUES('10002', 'Computer Pentium III 450 MHz', 30500,
'01');
```
- 2) 

```
INSERT INTO Products(ProductID, Name, UnitID)
VALUES('20002', 'Software Java Tutorial', 220, '02');
```
- 3) 

```
INSERT INTO Units(UnitID, Description)
VALUES('01', 'Set');
```



## การลบเรคอร์ดในเทเบิล

รูปแบบ

```
DELETE FROM TableName  
WHERE Condition ;
```

ตัวอย่าง

- 1) DELETE FROM Units ;
- 2) DELETE FROM Products WHERE ProductID = '10010' ;



## 2. SQL

การเปลี่ยนแปลงข้อมูลเรคอร์ดในเทเบิล

รูปแบบ

```
UPDATE TableName SET FieldName = Value , ...  
WHERE Condition ;
```

ตัวอย่าง

- 1) UPDATE Products SET Price = 10000 ;
- 2) UPDATE Products SET Price = 10000  
WHERE ProductID = '10010' ;



# SQL

การสืบค้นข้อมูล

รูปแบบ

```
SELECT *  
FROM TableName
```

ตัวอย่าง

```
SELECT * FROM Units ;
```





## 2. SQL

การสืบค้นข้อมูล

รูปแบบ

```
SELECT FieldName1, FieldName2  
FROM TableName
```

ตัวอย่าง

- 1) SELECT UnitID FROM Units ;
- 2) SELECT Name, Price FROM Products ;



# SQL

การสืบค้นข้อมูล

รูปแบบ

```
SELECT FieldName1, FieldName2
FROM TableName
WHERE Condition
```

ตัวอย่าง

1. SELECT Name, Price FROM Products  
WHERE Price > 5000 ;
2. SELECT ProductID FROM Products  
WHERE Name LIKE '%Computer%'
3. SELECT ProductID, Name FROM Products  
WHERE Price IN (3000, 5000)