

## 2.

# คำสั่ง SQL ในการเรียกดูข้อมูล

DB304+DB404 การจัดการฐานข้อมูลทางธุรกิจดิจิทัล/ปฏิบัติฯ

อาจารย์เฉลิมพล ศิริกายน และ ดร.ปาณิตา ชูสรานนท์

# การเรียกค้นข้อมูล

## The SQL SELECT Statement

The SELECT statement is used to select data from a database.

The data returned is stored in a result table, called the result-set.

### SELECT Syntax

```
SELECT column1, column2, ...  
FROM table_name;
```

```
SELECT ProductName, UnitPrice  
FROM products
```

Here, column1, column2, ... are the field names of the table you want to select data from. If you want to select all the fields available in the table, use the following syntax:

```
SELECT * FROM table_name;
```

```
SELECT * FROM products
```

Let's try

```
SELECT ProductName, UnitPrice*1.1 FROM products
```

## ตัวอย่าง 0

1. เรียกดูชื่อสินค้าและจำนวนสินค้าคงเหลือค้างอยู่ใน Stock

```
SELECT ProductName, UnitsInStock  
FROM products
```

2. เรียกดูชื่อสินค้า เฉพาะสินค้าที่มีจำนวนสินค้าอยู่ใน Stock มากกว่า 50

```
SELECT ProductName, UnitsInStock  
FROM products  
WHERE UnitInStock > 50;
```

3. เรียกดูชื่อสินค้าที่เลิกขายแล้ว (Discontinued)

```
SELECT ProductName, UnitsInStock  
FROM products  
WHERE Discontinued=1;
```

## ตัวอย่าง 1

เรียกดูซื้อสินค้าที่เลิกผลิตแล้ว แต่ยังมีจำนวนสินค้าคงเหลือค้างอยู่ใน Stock

# Alias

## SQL Aliases

SQL aliases are used to give a table, or a column in a table, a temporary name.

Aliases are often used to make column names more readable.

An alias only exists for the duration of the query.

### Alias Column Syntax

```
SELECT column_name AS alias_name  
FROM table_name;
```

## ตัวอย่าง 2

เรียกดูซื้อสินค้าที่มียอดสั่งซื้อมูลค่าเกิน 500

# Distinct

## The SQL SELECT DISTINCT Statement

The SELECT DISTINCT statement is used to return only distinct (different) values.

Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.


### SELECT DISTINCT Syntax

```
SELECT DISTINCT column1, column2, ...  
FROM table_name;
```

### ตัวอย่าง 3

ลูกค้าจากประเทศ France มาจากเมือง (city) อะไรบ้าง

```
SELECT  
FROM customers  
WHERE country = 'France';
```



ต้องสะกดให้เหมือน  
ข้อมูลในตาราง



# LIKE operator

## The SQL LIKE Operator

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards often used in conjunction with the LIKE operator:

- % - The percent sign represents zero, one, or multiple characters
- \_ - The underscore represents a single character

**Note:** MS Access uses an asterisk (\*) instead of the percent sign (%), and a question mark (?) instead of the underscore (\_).

The percent sign and the underscore can also be used in combinations!

## LIKE Syntax

```
SELECT column1, column2, ...  
FROM table_name  
WHERE columnN LIKE pattern;
```

## ตัวอย่าง 4

เรียกดุรายชื่อผู้ติดต่อ (ContactName) และชื่อบริษัท (CompanyName) ของลูกค้า เฉพาะบริษัทที่มีชื่อขึ้นต้นด้วย a

## ตัวอย่าง 5

เรียกดูชื่อผู้ติดต่อ (ContactName) และชื่อบริษัท (CompanyName) ของลูกค้า  
เฉพาะบริษัทที่ซื้อลงท้ายว่า markets

## ตัวอย่าง 6

เรียกดูชื่อผู้ติดต่อ (ContactName) และชื่อบริษัท (CompanyName) ของลูกค้า  
เฉพาะบริษัทที่มีตัวอักษร et อยู่ในชื่อบริษัท

## ตัวอย่าง 7

เรียกดูชื่อผู้ติดต่อ (ContactName) และชื่อบริษัท (CompanyName) ของลูกค้า เฉพาะชื่อบริษัทที่มีตัวอักษร e และ t โดยมีตัวอักษร 1 ตัว คั่นกลางระหว่าง e และ t เช่น ect, ent, est

## ตัวอย่าง 8

เรียกดูชื่อผู้ติดต่อ (ContactName) และชื่อบริษัท (CompanyName) ของลูกค้า  
เฉพาะบริษัทที่มีชื่อขึ้นต้นด้วยตัวอักษร b และลงท้ายด้วย s

# BETWEEN operator

## The SQL BETWEEN Operator

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.

The BETWEEN operator is inclusive: begin and end values are included.

### BETWEEN Syntax

```
SELECT column_name(s)
FROM table_name
WHERE column_name BETWEEN value1 AND value2;
```

## ตัวอย่าง 9

รายชื่อสินค้าและราคาต่อหน่วย เฉพาะสินค้าที่มีราคาต่อหน่วยตั้งแต่ 20 ถึง 50



## ตัวอย่าง 10.1

จากตารางข้อมูลลูกค้า เรียกดูชื่อผู้ติดต่อ ( ContactName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France

# IN operator

## The SQL IN Operator

The IN operator allows you to specify multiple values in a WHERE clause.

The IN operator is a shorthand for multiple OR conditions.

### IN Syntax

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

or:

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (SELECT STATEMENT);
```

## ตัวอย่าง 10.2

จากตารางข้อมูลลูกค้า เรียกดูชื่อผู้ติดต่อ ( ContactName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France

## การเรียงลำดับและการจำกัดการแสดงผล

```
SELECT column1, column2, ...  
FROM table_name  
ORDER BY column1, column2, ... ASC|DESC;
```

```
SELECT column_name(s)  
FROM table_name  
WHERE condition  
LIMIT number;
```

## ตัวอย่าง 11.1

จากตารางข้อมูลลูกค้า เรียกดูชื่อบริษัท (CompanyName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France เรียงลำดับตามชื่อบริษัท  
จาก A-Z

## ตัวอย่าง 11.2

จากตารางข้อมูลลูกค้า เรียกดูชื่อบริษัท (CompanyName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France เรียงลำดับตามชื่อบริษัท  
จาก Z-A

## ตัวอย่าง 11.3

จากตารางข้อมูลลูกค้า เรียกดูชื่อบริษัท (CompanyName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France เรียงลำดับตามชื่อบริษัท โดยแสดงข้อมูลแค่ 2 รายการแรก

## ตัวอย่าง 11.4

จากตารางข้อมูลลูกค้า เรียกดูชื่อบริษัท (CompanyName), ตำแหน่งผู้ติดต่อ (ContactTitle), ประเทศ (Country) โดยเรียกดูเฉพาะลูกค้าที่มีตำแหน่งเป็น Owner และอยู่ในประเทศ Mexico, Spain, France เรียงลำดับตามชื่อบริษัท โดยแสดงข้อมูลรายการที่ 5-6



## ตัวอย่าง 12

แสดงชื่อสินค้า ราคาต่อหน่วย และจำนวนสินค้าต่อหน่วย โดยแสดงเฉพาะสินค้าที่มีหน่วยเป็นกล่อง (box) 5 รายการที่มีราคาต่อหน่วยสูงสุด

# Group and Aggregate functions

## The SQL GROUP BY Statement

The GROUP BY statement groups rows that have the same values into summary rows, like "find the number of customers in each country".

The GROUP BY statement is often used with aggregate functions (COUNT, MAX, MIN, SUM, AVG) to group the result-set by one or more columns.

### GROUP BY Syntax

```
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
ORDER BY column_name(s);
```

## ตัวอย่าง 13.1

นับว่ามีลูกค้าทั้งหมดกี่คน

นับว่ามีลูกค้าทั้งหมดกี่คน ในแต่ละประเทศ

```
SELECT country, COUNT(*) AS member  
FROM customers  
GROUP BY country;
```

ลองเพิ่ม ContactName

## ตัวอย่าง 13.1

นับว่ามีลูกค้าทั้งหมดกี่คน

## ตัวอย่าง 13.2

นับว่ามีลูกค้าทั้งหมดกี่คน ในแต่ละประเทศ

## ตัวอย่าง 13.3

นับจำนวนลูกค้าที่อยู่ในแต่ละประเทศ UK, France หรือ Spain

## ตัวอย่าง 13.4

นับจำนวนลูกค้าที่อยู่ในแต่ละประเทศ UK, France หรือ Spain เรียงลำดับตามจำนวนลูกค้า

# HAVING condition

## The SQL HAVING Clause

The HAVING clause was added to SQL because the WHERE keyword could not be used with aggregate functions.

### HAVING Syntax

```
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
HAVING condition
ORDER BY column_name(s);
```



## ตัวอย่าง 14

นับจำนวนลูกค้าจากประเทศ UK, France หรือ Spain โดยแสดงเฉพาะประเทศที่มีลูกค้ามากกว่า 5 ราย และแสดงผลเรียงลำดับตามจำนวนลูกค้าจากมากไปน้อย

## Group and Aggregate functions (ต่อ)

```
SELECT MAX(UnitPrice) max_price,  
       MIN(UnitPrice) min_price  
FROM products;
```

```
SELECT Discontinued, MAX(UnitPrice) max_price,  
       MIN(UnitPrice) min_price  
FROM products  
GROUP BY Discontinued;
```

## ตัวอย่าง 15.1

ราคาเฉลี่ยของสินค้าในแต่ละหมวด (CategoryID)

```
SELECT CategoryID, AVG(UnitPrice) avg_price  
FROM products  
GROUP BY CategoryID;
```

CategoryID	avg_price
1	37.97916667
2	23.06250000
3	25.16000000
4	28.73000000
5	20.25000000
6	54.00666667
7	32.37000000
8	20.68250000

## ตัวอย่าง 15.2

ราคาเฉลี่ยของสินค้าที่มีค้างอยู่ใน Stock จำแนกตามหมวดหมู่ (CategoryID)

CategoryID	avg_price
1	37.97916667
2	23.21818182
3	25.16000000
4	30.53333333
5	20.25000000
6	42.81666667
7	32.37000000
8	20.68250000

# Built-in functions

- **String**

เช่น concat, char\_length, lower, upper, repeat

- **Date & Time**

เช่น now, curdate, curtime, timestamp, datediff, dayofweek, month, year

- **Numeric**

เช่น least, greatest, ceiling, floor, round, rand, sqrt, power

# Built-in functions

## ○ String

เช่น concat, char\_length, lower, upper, repeat

### ตัวอย่าง

```
SELECT CONCAT('Hello', ' ', 'world');  
SELECT REPEAT('OX', 5);
```

# Built-in functions

## ○ Date & Time

เช่น now, curdate, curtime, timestamp, datediff, dayofweek, month, year

### ตัวอย่าง

```
SELECT NOW();
```

```
SELECT CURRENT_DATE();
```

```
SELECT DAYOFWEEK(NOW());
```

```
SELECT DATEDIFF(CURDATE(), TIMESTAMP('1974-08-13'));
```

# Built-in functions

## ○ Numeric

เช่น least, greatest, ceiling, floor, round, rand, sqrt, power

### ตัวอย่าง

```
SELECT LEAST (5, 7, 1, -3, 0) ;
```

```
SELECT CEILING (3.42) ;
```

```
SELECT FLOOR (3.42) ;
```

```
SELECT ROUND (3.42, 1) ;
```

```
SELECT POWER (9, 2) ;
```

```
SELECT SQRT (16) ;
```