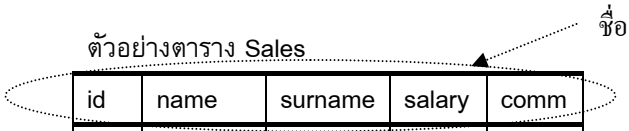
A typical SQL query

```
select <ชื่อคอลัมน์1, ชื่อคอลัมน์2, …>
from <ชื่อตาราง1, ชื่อตาราง2>
where <เงื่อนไข>;
```



id	name	surname	salary	comm
101	Somchai	Jaidee	10000	40000
123	Somsri	DeeDen	25000	21000
204	Sakchai	Boonma	18000	22000

```
Sales (
    id          integer,
    name        char(20),
    surname     char(20),
    salary      real,
    comm        real,
    primary key(id));
```

เช่น

□ แสดงเงินเดือนพนักงานขายคนที่มี id = 123

```
select   salary
from     Sales
where id=123;
```

□ แสดงข้อมูลทั้งหมดของพนังงานที่มีเงินเดือนรวมกับค่าคอมมิชชั่นมากกว่า 20000

```
select   *
from     Sales
where    salary+comm > 20000;
```

WHERE Clause

หากมีมากกว่า 1 เงื่อนไข สามารถใช้ตัวเชื่อม and, or และ not ได้

เช่น

□ แสดงข้อมูลทั้งหมดของพนักงานที่มีเงินเดือนมากกว่า 10000 และค่าคอมมิชชั่น (comm) มากกว่าเงินเดือน

```
select   *
from     Sales
where    salary > 10000
and      comm > salary;
```

□ แสดงข้อมูลทั้งหมดของพนักงานซึ่งมีเงินเดือนอยู่ในช่วง [8000, 9000]

```
select   *
from     Sales
where    salary between 8000 and 9000;
```

Pattern Matching in the WHERE Clause

★ ใช้ % สำหรับแทนตัวอักษรใดๆ กี่ตัวก็ได้

★ ใช้ _ สำหรับตัวอักษรใดๆเพียงหนึ่งตัวเท่านั้น

เช่น

□ แสดงข้อมูลทั้งหมดของพนักงานที่มีชื่อเริ่มต้นด้วยคำว่า ‘Som’

```
select  *
from    Sales
where   name like ‘Som%’;
```

□ แสดงข้อมูลทั้งหมดของพนักงานที่ชื่อมี 3 ตัวอักษร

```
select  *
from    Sales
where   name like ‘_ _ _’;
```

Aggregate Function

เป็นฟังก์ชันที่คำนวณจำนวนหรือค่าที่ต้องการเป็นกลุ่มๆได้ เช่น

★ count นับค่าที่ต้องการว่ามีจำนวนเท่าไร

★ sum บวกค่าที่ต้องการ

★ avg หาค่าเฉลี่ย

★ min หาค่าที่น้อยที่สุด

★ max หาค่าที่มากที่สุด

เช่น

□ แสดงเงินเดือนที่น้อยที่สุดและมากที่สุดของพนักงานทั้งหมด

```
select  min(salary) as low,
        max(salary) as high
from    Sales;
```

□ แสดงค่าเฉลี่ยคอมมิชชั่นของพนักงานทั้งหมด

```
select  avg(comm)
from    Sales;
```

□ แสดงจำนวนของพนักงานที่มีเงินเดือนมากกว่า 10000

```
select  count(*)
from    Sales
where   salary > 10000;
```

Order by

ใช้สำหรับเรียงลำดับแถวที่เป็นผลลัพธ์จากการ Query โดยมักจะวางไว้หลัง **WHERE** เช่น

- แสดงข้อมูลทั้งหมดโดยเรียงลำดับตามชื่อพนักงานจาก A -> Z (น้อยไปมาก)

```
select  *
from    Sales
order by name asc;
```

โดยปกติ asc จะเป็นค่า default ของ order by ดังนั้นอาจจะไม่ต้องเขียนก็จะได้ผลลัพธ์ที่เหมือนกับเขียน asc เช่น

```
select  *
from    Sales
order by name;
```

- แสดงชื่อพนักงานที่มีเงินเดือนน้อยกว่า 50000 โดยเรียงลำดับตามรหัสพนักงาน (id) จากมากไปน้อย

```
select  name
from    Sales
where   salary < 50000
order by id desc;
```

Data Modification

◎ Insertion Statements

• แทรกแถวใหม่ในตาราง :

```
insert into product values (10, 'PDA', 1, 129.99);
insert into product values (11,'Laptop',1,1000.00);
```

• ในกรณีที่ไม่ทราบค่าของข้อมูลบางตัวให้ใส่ null :

```
insert into product values (10, 'PDA', null, 129.99);
```

หรือ

```
insert into product (name, code, price) values ('PDA', 10, 129.99);
```

◎ Deletion Statements

• ลบแถวออกจากตาราง

```
delete from product
where type = 1;

delete from category
where name = 'PDA';
```

◎ Update Statements

• เปลี่ยนแปลงค่าแต่ละฟิลด์ในตาราง ซึ่งจะเปลี่ยนได้ทีละแถว(เรคคอร์ด)

```
update product
set     name = 'PocketPC',
        type = 2,
        price = price * 1.1
where code = 10;
```