

## แบบฝึกปฏิบัติการที่ 5

### คำสั่ง ฟังก์ชันที่ใช้ในการจัดการข้อมูลที่เป็นตัวอักษรและตัวเลข

วัตถุประสงค์

1. ฝึกปฏิบัติการทำงานกับ String Functions
2. ฝึกปฏิบัติการทำงานกับ Number Functions

#### 5.1 String Functions

คือการจัดดำเนินการข้อมูลที่เป็นตัวอักษร หากคุ้นเคยกับการเขียนโปรแกรมก็คือการประกาศให้ข้อมูลเหล่านี้เป็นประเภท String หรือ text นั่นเอง

##### 5.1.1 ฟังก์ชัน Lower และ Upper

โดย Lower ใช้ในการแปลงข้อความเป็นตัวเล็กทั้งหมด Upper ตัวในการแปลงข้อความให้เป็นตัวใหญ่ทั้งหมด

Lower(text)

Uper(text)

text :: ข้อความหรือประยุดต์ให้มองเป็นชื่อ column\_nameก็ได้

ตัวอย่าง

```
select lower("XXX"),upper("xxx") ;
```

จะได้

lower("XXX")	upper("xxx")
xxx	XXX

ตัวอย่าง SELECT custName FROM dbsuj.custtbl;

จะได้

Teddy Danders
Ammy Jenkins
Betty Thomson
Harry Hayes
Colin White

หากเพิ่ม function Upper จะได้

upper(custName)
TEDDY DANDERS
AMMY JENKINS
BETTY THOMSON
HARRY HAYES
COLIN WHITE

### 5.1.2 ฟังก์ชันการตัดคำ Rtrim , Ltrim , Trim

เป็นฟังก์ชันที่ใช้ในการตัดคำ ที่ระบุตำแหน่ง มีรูปแบบดังนี้

```
Rtrim(text);
Ltrim(text);
Trim(text);
Trim(text);
```

ตัวอย่าง drop table student ;

table\_name :: ชื่อตารางที่ต้องการแก้ไขรายการข้อมูล

column\_name\_1, column\_name\_2, ... , column\_name\_n :: ชื่อคอลัมน์ที่ต้องการเป็นหัวตาราง

value1, value2,..., valuen :: เป็นเซตของข้อมูล

ตัวอย่าง

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
▶	7369	SMITH	CLERK	7902	1981-12-07	800.00	NULL	20
	7788	SCOTT	ANALYST	7566	1981-11-03	3000.00	NULL	20
	7876	ADAMS	CLERK	7788	1981-12-03	1100.00	NULL	20
	7900	JAMES	CLERK	7698	1981-12-03	950.00	NULL	30
	7902	FORD	ANALYST	7566	1981-12-03	3000.00	NULL	20
	7934	MILLER	CLERK	7782	1982-01-23	1300.00	NULL	10

หมายเหตุสามารถเขียนอีกแบบได้เป็น

select \* from emp where (job = 'ANALYST' or job = 'CLERK');

ตัวอย่าง การค้นหาค่าในตาราง emp ที่มี Job ไม่เป็น 'ANALYST','CLERK'

select \* from emp where job not in ('ANALYST','CLERK');

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
▶	7499	ALLEN	SALESMAN	7698	1981-02-20	1600.00	300.00	30
	7521	WARD	SALESMAN	7698	1981-12-02	1250.00	500.00	30
	7566	JONES	MANAGER	7839	1981-04-02	2975.00	NULL	20
	7654	MARTIN	SALESMAN	7698	1981-09-08	1250.00	1400.00	30
	7698	BLAKE	MANAGER	7839	1981-01-05	2850.00	NULL	30
	7782	CLARK	MANAGER	7839	1981-09-06	2450.00	NULL	10
	7839	KING	PRESIDENT	NULL	1981-11-07	5000.00	NULL	10
	7844	TURNER	SALESMAN	7698	1981-09-08	1500.00	0.00	30

5.1.2 ฟังก์ชัน SUBSTR เป็นฟังก์ชันที่ใช้ในการแยกสตริงที่ระบุ โดยที่เราสามารถระบุได้ว่าการแยกสตริงตั้งแต่อักขระตัวที่ เท่าไรของสตริงที่ระบุ หรือต้องการแยกสตริงเป็นจำนวนกี่อักขระ ฯลฯ

SUBSTR(str, pos);

SUBSTR(str FROM pos);

SUBSTR(str, pos, len);

SUBSTR(str FROM pos FOR len);

- ฟังก์ชัน substr(str,pos) เป็นฟังก์ชันที่ทำการแยกสตริง str ตั้งแต่ตำแหน่ง pos z ฟังก์ชัน
- substr(str FROM pos) เป็นฟังก์ชันที่ทำการแยกสตริง str ตั้งแต่ตำแหน่ง pos ภาคผนวก 85
- ฟังก์ชัน substr (str,pos,len) เป็นฟังก์ชันที่ทำการแยกสตริง str ตั้งแต่ตำแหน่ง pos เป็น จำนวน len
- ฟังก์ชัน substr (str FROM pos FOR len) เป็นฟังก์ชันที่ทำการแยกสตริง str ตั้งแต่ตำแหน่ง pos เป็น จำนวน len

ตัวอย่าง กำหนดสตริง 'COMPUTER' ให้เขียนคำสั่ง SQL เพื่อแสดงข้อความตั้งแต่อักขระ ตัวที่ 4

```
mysql> select substring('COMPUTER',4);
+-----+
| substring('COMPUTER',4) |
+-----+
| PUTER                    |
+-----+
1 row in set (0.03 sec)

mysql> select substring('COMPUTER'from 4);
+-----+
| substring('COMPUTER'from 4) |
+-----+
| PUTER                        |
+-----+
1 row in set (0.00 sec)
```

กำหนดสตริง 'COMPUTER' ให้เขียนคำสั่ง SQL เพื่อแสดงข้อความตั้งแต่อักขระ ตัวที่ 4 เป็นจำนวน 3 อักขระ

Select substring(ชื่อฟิลด์,3) from table\_name;

```
mysql> select substring('COMPUTER',4, 3);
+-----+
| substring('COMPUTER',4, 3) |
+-----+
| PUT                        |
+-----+
1 row in set (0.00 sec)

mysql> select substring('COMPUTER' from 4 for 3);
+-----+
| substring('COMPUTER' from 4 for 3) |
+-----+
| PUT                                |
+-----+
1 row in set (0.00 sec)
```

### 5.1.3 ฟังก์ชัน Length

ฟังก์ชัน LENGTH เป็นฟังก์ชันที่ใช้ในการหาค่าความยาวของสตริงที่ระบุ โดยให้ผลลัพธ์เป็นจำนวนอักขระของสตริง

LENGTH(str1)

Select length(ชื่อฟิลด์) from table\_name;

#### 5.1.4 Concat เชื่อม string

กำหนดสตริง 'COMPUTER' ให้เขียนคำสั่ง SQL เพื่อแสดงความยาวของสตริง

Select concat(ชื่อฟิล1,' ', ชื่อฟิลที่2) from table\_name;

concat(str1, str2,...)

ตัวอย่าง กำหนดสตริง 'CS', '&' และ สตริง 'IT' ให้เขียนคำสั่ง SQL เพื่อเชื่อมสตริงทั้งหมด ให้เป็นข้อความเดียวกัน

```
mysql> select concat('CS',' & ','IT');
+-----+
| concat('CS',' & ','IT') |
+-----+
| CS & IT                  |
+-----+
1 row in set (0.00 sec)
```

5.1.5 ฟังก์ชัน REPLACE เป็นฟังก์ชันที่ใช้ในการแทนที่สตริง str ด้วยการนำเอาสตริง fromStr ไปแทนที่สตริง toStr

REPLACE(str, fromStr, toStr);

ตัวอย่าง กำหนดสตริง 'www.mysql.com' ให้เขียนคำสั่ง SQL เพื่อแทนที่อักขระ 'w' ในสตริงที่กำหนด ด้วย 'Ww'

```
mysql> select replace('www.mysql.com','w','Ww');
+-----+
| replace('www.mysql.com','w','Ww') |
+-----+
| WwWwWw.mysql.com                  |
+-----+
1 row in set (0.00 sec)
```

## 5.2 Numeric Functions

Numeric Functions หมายถึง ฟังก์ชันที่ใช้ในการจัดการกับข้อมูลที่เป็นตัวเลข หัวข้อนี้จะอธิบายและ ยกตัวอย่าง ฟังก์ชันสำคัญๆ ที่ใช้ในการจัดการข้อมูลที่เป็นตัวเลข

### 5.2.1 ฟังก์ชัน ABS เป็นฟังก์ชันที่ใช้ในการหาค่าสัมบูรณ์ (Absolute Number) ของตัวเลข

```
ABS(value);
```

### 5.2.2 ฟังก์ชัน DIV

```
value DIV divisor;
```

### 5.2.3 Mod

```
MOD(value, divisor);
```

### 5.2.4 ฟังก์ชัน ROUND

```
ROUND (value [, precision]);
```

## 5.3 count

```
count( * | ชื่อฟิลด์);
```

## 5.4 sum, avg ,mean ,max

```
sum (value [, precision]);
```

\*\*\* value เป็นชื่อ ฟิลด์ได้

## แบบฝึกปฏิบัติ

สร้าง table ให้ empNo เป็น primary key

- empNo          7 char          (หมายเลขประจำตัวพนักงาน)
- empName       30 char        (ชื่อพนักงาน)
- empAddr        20 char        (ที่อยู่พนักงาน)
- empEmail       20 char        (อีเมลพนักงาน)
- empSalary      integer        (เงินเดือนพนักงาน)

มีข้อมูล

AA01122	Landi Santos	Denvor	landi@hotmail.com	16000
BB02233	Joe Jenkins	Seattle	joe@yahoo.com	15000
CC03344	Cherry Gordon	Seattle	cherry@hotmail.com	20000
CC04455	Colin White	Denvor	colin@gmail.com	25000
AA05566	Thomas Johnson	Lynnwood	thomas@yahoo.com	27000
BB06677	Candy Kendall	Denvor	candy@gmail.com	18000
AA07788	Smith Gordon	Denvor	smith@gmail.com	22000
CC08899	Cherry Gordon	Lynnwood	jessy@yahoo.com	17000
CC12233	Susan Santos	Denvor	susan@gmail.com	19000
AA23344	John Sanders	Lynnwood	john@gmail.com	23000

1. แสดงรายการข้อมูลพนักงานที่ใช้ gmail โดยเรียงตามลำดับตัวอักษรชื่อพนักงานจากมากไปน้อย
2. ให้แสดงชื่อพนักงานเป็นอักษรตัวใหญ่ ตั้งชื่อคอลัมน์ว่า empName
3. ให้แสดงหมายเลขประจำตัวพนักงานที่ได้ตัดอักษร 2 ตัวแรกออกไป เช่น 01122, 02233 ตั้งชื่อ คอลัมน์ว่า empNo
4. ให้แสดงชื่อพนักงาน และจำนวนอักขระของชื่อพนักงาน ตั้งชื่อคอลัมน์ว่า Name Length

5. แสดงรายการข้อมูลพนักงานแผนก AA (รหัสพนักงานขึ้นต้นด้วย AAxxxxx) โดยเรียงตามลำดับ รหัสพนักงานจากน้อยไปมาก
6. ให้แสดงรายชื่อเมือง (empAddr) ที่แตกต่างกัน โดยแสดงตัวอักษร 3 ตัวสุดท้ายของชื่อ เช่น vor, tle, ood
7. ให้แสดงชื่อเมือง 3 ตัวอักษร และชื่อพนักงาน 3 ตัวอักษร โดยมี “-” คั่นกลาง เช่น Den-Lan ตั้งชื่อคอลัมน์ว่า City-Name โดยเรียงลำดับตามชื่อเมืองจากมากไปน้อย และเรียงลำดับตามชื่อ พนักงานจากน้อยไปมาก
8. แสดงหมายเลขพนักงาน และหมายเลขพนักงานที่มี “-” คั่นระหว่าง แผนก และเลขลำดับ เช่น AA-01122, BB-02233
9. ให้แสดงนามสกุลพนักงานทั้งหมด เช่น Santos, Jankins ตั้งชื่อคอลัมน์ว่า Last Name โดยเรียง ลำดับตามอักขระของนามสกุลจากน้อยไปมาก (a, b, c,..., z) และต้องไม่มีนามสกุลซ้ำ
10. แสดงหมายเลขประจำตัวพนักงาน ชื่อพนักงาน เงินเดือนพนักงาน (บาท) และเงินเดือนพนักงาน (K) หมายถึง  $1K = 1,000$  บาท
11. แสดงพนักงานที่มีเงินเดือนเฉลี่ย
12. แสดงจำนวนพนักงานทั้งหมด