

Single-row Function dan Group Function

Rosa Ariani Sukamto
Email: rosa_if_itb_01@yahoo.com
Blog: <http://udinrosa.wordpress.com>
Website: <http://www.gangsir.com>

Single-row Function

- Memanipulasi Data
- Proses dilakukan per baris dan menghasilkan satu nilai per baris
- Bisa digunakan untuk memodifikasi tipe data (konversi)
- Dapat digunakan bersarang (*nested*)
- Argumen dapat berupa baris atau ekspresi

Pengelompokan Single-row Function

- Character Function
- Number Function
- Date Function
- Conversion
- General Function

Character Function: Case Manipulation

LOWER = mengubah karakter menjadi huruf kecil

UPPER = mengubah karakter menjadi huruf besar

INITCAP = setiap kata diawali dengan huruf besar

```
SELECT LOWER(nama), UPPER(nama),  
       INITCAP(nama) FROM pacar;
```

Lower	Upper	Initcap
sami	SAMI	Sami

Character Function

LENGTH(x) = mengembalikan jumlah karakter pada string masukan atau isi dari kolom masukan

```
SELECT LENGTH(nama) FROM pacar WHERE  
UPPER(nama) LIKE 'SAMI%';
```

Character Function

SUBSTR(x, y [, z]) = menampilkan substring dari string atau kolom **x** mulai posisi karakter **y** sampai akhir atau sebanyak **z**

```
SUBSTR('ABCDEFGH', 3) = 'CDEFGH'
```

```
SUBSTR('ABCDEFGH', 3, 4) = 'CDEF'
```

```
SUBSTR('ABCDEFGH', -1, 1) = 'H'
```

Jika **y** bernilai minus (-) maka penghitungan karakter dimulai dari belakang

Character Function

INSTR(X, Y) = mencari posisi string atau substring **Y** dalam string **X**

INSTR('ABCDEFGH', 'CD') = 3

```
SQL> select instr('ABCDEFGH', 'CD') from dual;
```

```
INSTR('ABCDEFGH','CD')
```

```
-----  
3
```

Character Function

LPAD(X, Y [, Z]) = rata kanan (untuk string dengan alokasi tempat sebanyak Y dan kosong diisi dengan Z)

```
SQL> select lpad('Jon Bon Jovi', 20) from dual;
```

```
LPAD('JONBONJOVI',20
```

```
-----  
Jon Bon Jovi
```

```
SQL> select lpad('Jon Bon Jovi', 20, '$') from dual;
```

```
LPAD('JONBONJOVI',20
```

```
-----  
$$$$$$$$$Jon Bon Jovi
```

Character Function

RPAD(X, Y [, Z]) = rata kiri (untuk string dengan alokasi tempat sebanyak Y dan kosong diisi dengan Z)

```
SQL> select rpad('thank you for break my heart', 30) from dual;
```

```
RPAD('THANKYOUFORBREAKMYHEART'
```

```
-----  
thank you for break my heart
```

```
SQL> select rpad('i get better now :p', 30, '$') from dual;
```

```
RPAD('IGETBETTERNOW:P',30,'$')
```

```
-----  
i get better now :p$$$$$$$$$$$$
```

Character Function: TRIM

TRIM([A] B FROM C) = menghilangkan karakter di kanan dan kiri string C yang merupakan karakter B

A dapat diganti dengan:

LEADING = dihilangkan di awal

TRAILING = dihilangkan di akhir

BOTH = dihilangkan di awal dan di akhir


```

SQL> select trim(leading 'i' from 'i love you') from dual;

TRIM(LEAD
-----
love you

SQL> select trim(trailing 'u' from 'miss u') from dual;

TRIM(
-----
miss

SQL> select trim(both 'u' from 'u miss u') from dual;

TRIM(B
-----
miss

SQL> select trim('$' from '$duit$') from dual;

TRIM
----
duit

SQL> select trim('$' from '$duit$duit$') from dual;

TRIM('$'F
-----
duit$duit

SQL> select trim('$' from '$$$$duit$$$') from dual;

TRIM
----
duit

```

Character Function

REPLACE(X, Y, [, Z]) = mengganti string atau substring **Y** yang ada di string **X** dengan string atau substring **Z**

```

SQL> select replace('i hate you', 'hate', 'love') from dual;

REPLACE('I
-----
i love you

```

Number Function

ROUND(X, Y) = membulatkan **X** dengan presisi desimal **Y**

```
SQL> select round(99.9999, 2),  
2 round(99.9999, 0),  
3 round(99.9999, -1),  
4 round(99.9999, -2)  
5 from dual;
```

```
ROUND(99.9999,2) ROUND(99.9999,0) ROUND(99.9999,-1) ROUND(99.9999,-2)  
-----  
100 100 100 100
```

Number Function: Round

```
SQL> select round(45.9999, 2),  
2 round(45.9999, 0),  
3 round(45.9999, -1),  
4 round(45.9999, -2)  
5 from dual;
```

```
ROUND(45.9999,2) ROUND(45.9999,0) ROUND(45.9999,-1) ROUND(45.9999,-2)  
-----  
46 46 50 0
```

Number Function

TRUNC(X, Y) = memotong **X** dengan presisi desimal **Y**

```
SQL> select trunc(45.9999, 2),  
2 trunc(45.9999, 0),  
3 trunc(45.9999, -1)  
4 from dual;
```

```
TRUNC(45.9999,2) TRUNC(45.9999,0) TRUNC(45.9999,-1)  
-----  
45.99 45 40
```

Number Function

MOD(X, Y) = sisa pembagian **X** dengan **Y**

```
SQL> select mod(9, 4) from dual;
```

```
MOD(9,4)  
-----  
1
```


Oracle Date Format

`TO_CHAR(date, format)` = mengkonversi tipe date di oracle menjadi format yang kita inginkan

```
SQL> select to_char(sysdate, 'dd-month-yyyy') from dual;  
  
TO_CHAR(SYSDATE, '  
-----  
26-february -2009
```

Oracle Date Format

YYYY = 4 digit angka untuk tahun

YY = 2 digit untuk tahun

YEAR = tahun ditampilkan dengan ejaan

MM = dua digit untuk bulan

MONTH = nama bulan dalam ejaan penuh

MON = nama bulan ditampilkan 3 huruf depan

DY = nama hari ditampilkan 3 huruf depannya

DAY = nama hari ditampilkan penuh

DD = 2 digit untuk tanggal

HH = 2 digit untuk jam

HH12 = 2 digit untuk jam (0-12)

HH24 = 2 digit untuk jam (0-24)

MI = menit (0-59)

SS = detik (0-59)

General Function

NVL(x, y) = jika **x** null maka akan ditampilkan sebagai **y**

NVL2(x, y, z) = jika nilai **x** tidak null maka akan ditampilkan sebagai **y** jika nilai **x** null maka akan ditampilkan sebagai **z**

NULLIF(x, y) = jika nilai **x** dan **y** sama maka akan menghasilkan null, jika tidak sama maka akan menampilkan **x**

General Function

COALESCE(x, y, z) = jika **X** ada tampilkan **X**, jika **X** tidak ada maka tampilkan **Y**, jika **X** dan **Y** tidak ada maka tampilkan **Z**

Conditional Expression

```
CASE nama_kolom  
  WHEN nilai1 THEN statement  
  WHEN nilai2 THEN statement  
  .....  
ELSE  
  statement  
END
```

Conditional Expression

```
SELECT jumlah_kartu_kredit,  
  CASE pekerjaan_babe  
    WHEN 'dokter' THEN jumlah_kartu_kredit+5  
    WHEN 'anggota dpr' THEN jumlah_kartu_kredit+99  
  ELSE  
    jumlah_kartu_kredit  
  END  
FROM pacar;
```

Conditional Expression

```
SELECT DECODE(pekerjaan_babe,  
  'dokter', jumlah_kartu_kredit+5,  
  'anggota dpr',  
  jumlah_kartu_kredit+99) FROM pacar;
```

Group Function

AVG (rata-rata), **SUM** (jumlah nilai), **STDDEV**,
VARIANCE => (hanya untuk nilai numerik)
MAX (nilai maksimal), **MIN** (nilai minimum) =>
(bisa untuk numerik dan karakter)
COUNT (menghitung jumlah baris)

harus diikuti dengan **GROUP BY** jika perintah
seleksi diikuti dengan kolom lain

Group Function

```
SELECT SUM(jumlah_kartu_kredit),  
       pekerjaan_babe FROM pacar GROUP BY  
       nama;
```

```
SELECT SUM(jumlah_kartu_kredit),  
       pekerjaan_babe FROM pacar GROUP BY  
       nama HAVING jumlah_kartu_kredit=1;
```

GROUP BY tidak bisa menggunakan alias