

BAB V

Operator Precedence, LIKE, NOT LIKE, REGEXP

Tujuan Pembelajaran

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman database MySQL dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Materi

5.1 Operator Precedence

Operator precedence adalah tingkatan hirarki dalam memproses serangkaian operator.

Tingkatan hirarki	Jenis Operator
Paling Tinggi	BINARY
	NOT!
	- (unary minus)
	*%
	+
	<<>>
	&
	<< = <> != >> IN IS LIKE
	REGEXP RLIKE
	BETWEEN
	AND &&
Paling Rendah	OR

Semakin keatas posisi operator, maka semakin tinggi tingkat hirarki operator tersebut. Begitu pula sebaliknya, semakin rendah posisinya maka akan semakin lemah hirarkinya. Untuk operator yang sama kuat, misal + dan - digabung dengan operator * / %, maka akan ditentukan hirarkinya tergantung dari posisi mana yang paling kiri/paling awal ditemukan. Dan untungnya posisi hirarki ini dapat diubah dengan

bantuan tanda kurung "(" dan ")". Sekarang kita lihat penerapannya.

```
mysql> select 10+15-11*2, (10+15-11)*2,  
-> 2*6-5, 2*(6-5) ;  
+-----+-----+-----+-----+ |  
10+15-11*2 | (10+15-11)*2 | 2*6-5 | 2*(6-5) |  
+-----+-----+-----+-----+  
|          3|          28 |       7|       2|  
+-----+-----+-----+-----+  
1 row in set (0.00 sec)
```

Anda bisa perhatikan bahwa walaupun angka dan operatornya sama, tapi hasilnya bisa berbeda. Dan itu karena adanya peranan dari tanda kurung "(" dan ")" yang akan mengubah peta posisi hirarki operator.

5.2 Operator LIKE, NOT LIKE, REGEXP

Operator LIKE, NOT LIKE, REGEXP akan banyak kita gunakan terutama dalam operasi karakter.

A. Operator LIKE

Sekarang kita akan coba menggunakan operator LIKE. Operator LIKE ini digunakan untuk mencari data yang "menyerupai" atau "hampir sama" dengan kriteria tertentu. Biasanya untuk mencari data string/teks. Simbol "%" digunakan untuk membantu pelaksanaan operator LIKE. Posisi "%" sangat berpengaruh dalam menentukan kriteria. Kita langsung dengan contoh-contohnya saja ya biar lebih jelas penggunaannya. Tampilkan data karyawan yang namanya berawalan huruf "a": (perhatikan posisi simbol persennya "%")

```
mysql> select noid, nama  
-> from karyawan  
-> where nama LIKE "a%" ;  
  
+-----+-----+  
| noid | nama          |  
+-----+-----+  
|    1 | Ahmad Zobari  |  
|   31 | Anwar         |  
|   37 | Andika        |  
+-----+-----+  
3 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berawalan huruf "d":

```
mysql> select noid, nama
-> from karyawan
-> where nama LIKE "d%" ;
```

noid	nama
13	Dadan
18	Dian
19	Donno
22	Dadang

```
4 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berakhiran huruf "i". Perhatikan posisi penulisan tanda "%":

```
mysql> select noid, nama
-> from karyawan
-> where nama LIKE "%i" ;
```

noid	nama
1	Ahmad Zobari
2	Sundariwati
5	Yuliawati
7	Zobari
23	Yuliawati
26	Banowati
28	Gunadi
32	Susilowati
38	Siti
41	Yanti
42	Miranti

```
11 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berakhiran "wati":

```
mysql> select noid, nama
-> from karyawan
-> where nama LIKE "%wati" ;
```

noid	nama
2	Sundariwati
5	Yuliawati
23	Yuliawati
26	Banowati
32	Susilowati

```
5 rows in set (0.00 sec)
```

Bagaimana caranya agar operator LIKE dapat membedakan huruf besar dan kecil... Sederhana saja, cukup dengan menambahkan kata BINARY saja setelah perintah LIKE (sehingga perintahnya menjadi LIKE BINARY). Kita lihat contohnya...

```
mysql> select noid, nama
      -> from karyawan
      -> where nama LIKE BINARY "a%" ;
Empty set (0.34 sec)
```

Kenapa hasilnya menjadi "Empty set"? Kita coba dengan mengubah perintah tadi menjadi:

```
mysql> select noid, nama
      -> from karyawan
      -> where nama LIKE BINARY "A%" ;
+-----+-----+
| noid | nama          |
+-----+-----+
|    1 | Ahmad Zobari  |
|   31 | Anwar         |
|   37 | Andika        |
+-----+-----+
3 rows in set (0.00 sec)
```

Ya dengan menggunakan LIKE BINARY, penulisan huruf "a" akan dibedakan artinya dengan "A". Jelas kan?

Sekarang bagaimana kalau kita ingin menampilkan data, dengan kriteria bukan diawal atau diakhir kalimat, tapi berada diantara sebuah kata/kalimat? Misal ada berapa nama karyawan yang memiliki kata "lia"? . Perhatikan posisi penulisan tanda "%":

```
mysql> select noid, nama
      -> from karyawan
      -> where nama LIKE BINARY "%lia%" ;
+-----+-----+
| noid | nama          |
+-----+-----+
|    5 | Yuliawati     |
|    8 | Melia         |
|   23 | Yuliawati     |
|   30 | Melia         |
+-----+-----+
4 rows in set (0.00 sec)
```

Atau memiliki kata "Di" pada namanya?

```
mysql> select noid, nama
      -> from karyawan
      -> where nama LIKE BINARY "%Di%" ;
+-----+-----+
| noid | nama |
+-----+-----+
| 18 | Dian |
+-----+-----+
1 row in set (0.00 sec)
```

B. Operator REGEXP

Operator REGEXP (singkatan dari REGular EXPressions) merupakan bentuk lain dari operator LIKE, dengan fungsi yang lebih disempurnakan. Operator REGEXP biasanya ditemani juga dengan simbol-simbol tertentu dalam melaksanakan tugasnya, seperti:

Simbol	Keterangan
.	Satu tanda titik (.) untuk mewakili satu karakter
[?]	Untuk mewakili beberapa karakter atau range yang ditentukan
^	Untuk posisi awal dari sebuah kriteria yang ditentukan
\$	Untuk posisi akhir dari sebuah kriteria yang ditentukan

Kita langsung saja pada contohnya. Tampilkan nama karyawan yang berawalan huruf 'a':

```
mysql> select noid, nama
      -> from karyawan
      -> where nama REGEXP "^a" ;
+-----+-----+
| noid | nama          |
+-----+-----+
| 1 | Ahmad Zobari |
| 31 | Anwar         |
| 37 | Andika        |
+-----+-----+
3 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berawalan huruf "d".

```
mysql> select noid, nama
      -> from karyawan
      -> where nama REGEXP "^d" ;
+-----+-----+
| noid | nama          |
```

```
+-----+-----+
| 13 | Dadan |
| 18 | Dian  |
| 19 | Donno |
| 22 | Dadang |
+-----+-----+
4 rows in set (0.00 sec)
```

Tampilkan nama karyawan yang berawalan huruf 'a' sampai dengan huruf 'd':

```
mysql> select noid, nama
-> from karyawan
-> where nama REGEXP "[a-d]"
-> order by nama ;

+-----+-----+
| noid | nama          |
+-----+-----+
| 1    | Ahmad Zobari  |
| 37   | Andika        |
| 31   | Anwar         |
| 21   | Bambang       |
| 26   | Banowati      |
| 40   | Beno          |
| 13   | Dadan         |
| 22   | Dadang        |
| 18   | Dian          |
| 19   | Donno         |
+-----+-----+
10 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berakhiran huruf "i":

```
mysql> select noid, nama
-> from karyawan
-> where nama REGEXP "i$"
-> order by nama ;

+-----+-----+
| noid | nama          |
+-----+-----+
| 1    | Ahmad Zobari  |
| 26   | Banowati      |
| 28   | Gunadi        |
| 42   | Miranti       |
| 38   | Siti          |
| 7    | Zobari        |
| 2    | Sundariwati   |
| 32   | Susilowati    |
| 41   | Yanti         |
| 5    | Yuliawati     |
| 23   | Yuliawati     |
+-----+-----+
11 rows in set (0.00 sec)
```

Tampilkan data karyawan yang namanya berakhiran "wati":

```
mysql> select noid, nama
-> from karyawan
-> where nama REGEXP "wati$"
-> order by nama ;
```

noid	nama
26	Banowati
2	Sundariwati
32	Susilowati
5	Yuliawati
23	Yuliawati

```
5 rows in set (0.00 sec)
```

Tampilkan nama karyawan yang panjangnya 10 karakter:

```
mysql> select noid, nama
-> from karyawan
-> where nama REGEXP "^.....$" ;
```

noid	nama
3	Ryan Cakep
9	Zanda Cute

```
2 rows in set (0.00 sec)
```

Atau perintah di atas bisa juga ditulis dengan:

```
mysql> select noid, nama
-> from karyawan
-> where nama REGEXP "^.{10}$" ;
```

noid	nama
3	Ryan Cakep
9	Zanda Cute
32	Susilowati

```
3 rows in set (0.00 sec)
```

Nah, kurang lebih itulah dasar-dasar MySQL. Ini baru tutorial pengenalan. Kita akan bertemu lagi dengan tutorial berikutnya, dasar-dasar database relasi. Kemudian dilanjutkan dengan tutorial penerapan sederhana "database relasi dengan menggunakan 2 buah tabel".

Soal Latihan

1. Buka tabel peserta.
2. Tampilkan nama peserta yang diawali dengan huruf A
3. Tampilkan nama peserta yang diawali dengan huruf A dan diakhiri dengan huruf i
4. Tampilkan nama peserta yang berawalan d - g dan panjang karakternya 10

