

MODUL IV

FUNCTION

TUJUAN

Setelah menyelesaikan modul ini, anda diharapkan dapat :

1. Mengenalkan mahasiswa tentang konsep function/fungsi
2. Mengenalkan mahasiswa tentang penggunaan fungsi dalam basis data

DASAR TEORI

Store Routine

Store routine (function dan procedure) merupakan sekumpulan statement SQL yang dapat disimpan dalam server. Setelah routine disimpan, client tidak perlu memanggil statement individual terus menerus, namun cukup dengan memanggil stored routine.

Stored routine dapat bermanfaat pada situasi berikut :

- Ketika beberapa aplikasi client ditulis menggunakan bahasa yang berbeda-beda atau bekerja pada platform yang berbeda, namun memerlukan operasi database yang sama.
- Ketika keamanan diutamakan, function dan stored procedure dapat digunakan untuk operasi-operasi database yang umum digunakan. Routine dapat memastikan setiap operasi dicatat lognya. Aplikasi dan user dapat diatur tidak memiliki akses langsung ke database, tapi hanya dapat mengakses routine tertentu.

Stored routine dapat meningkatkan performa, karena dapat mengurangi informasi yang perlu dikirimkan antara server dan client. Kelemahan yang ditimbulkan oleh stored routine adalah peningkatan kinerja server, karena server melakukan lebih banyak pekerjaan.

Stored routine dalam MySQL dapat didefinisikan menjadi dua jenis, function menjadi dua jenis, yaitu function dan procedure. Sebuah function dapat digunakan dalam statement apapun layaknya function yang telah disediakan oleh MySQL. Berbeda dengan function, sebuah stored procedure dapat dipanggil menggunakan statement CALL. Stored function tidak dapat rekursif, sedangkan stored procedure memperbolehkan penggunaan rekursif namun secara default dinonaktifkan (disable).

Function

Sebuah function dapat digunakan secara langsung dalam statement SELECT, UPDATE, dan DELETE. Hasil dari function dapat dikembalikan sebagai output. Sebuah function hanya dapat mengembalikan sebuah nilai saja.

Membuat Function.

Sintaks yang digunakan untuk membuat function adalah sebagai berikut :

```
CREATE FUNCTION function_name ([func_parameter[,...]])
RETURNS type
Routine_body
```

Function menerima parameter (func_parameter) yang didefinisikan dengan nama parameter beserta tipe datanya. Pembuatan function juga perlu pendefinisian tipe data nilai yang dikembalikan oleh fungsi tersebut menggunakan statement RETURNS. Definisi apa yang dilakukan oleh sebuah fungsi diletakkan pada routine_body diapit oleh BEGIN dan END.

Berikut adalah contoh pembuatan fungsi.

1. Menggunakan table beli. Data yang terdapat pada table beli :

```
MariaDB [test]> select * from beli;
+-----+-----+-----+-----+
| id_beli | kd_barang | nama_barang | jumlah |
+-----+-----+-----+-----+
|      1 | K001     | PENSIL      |      30 |
|      2 | K001     | PENSIL      |      23 |
|      3 | K002     | BUKU        |      30 |
|      4 | K002     | BUKU        |      15 |
+-----+-----+-----+-----+
4 rows in set, 3 warnings (0.68 sec)
```

Buat fungsi tampil_jml_brg :

```
MariaDB [test]> DELIMITER $$
MariaDB [test]> CREATE FUNCTION tampil_jml_brg (p_jumlah INT) RETURNS INT
-> DETERMINISTIC
->
-> BEGIN
-> DECLARE jml INT;
-> SELECT COUNT(*) AS jml_brg INTO jml FROM beli WHERE
-> jumlah=p_jumlah;
-> RETURN jml;
-> END$$
Query OK, 0 rows affected (0.34 sec)
```

Panggil fungsi tampil_jml_brg :

```

MariaDB [test]> SELECT tampil_jml_brg (30);
+-----+
| tampil_jml_brg (30) |
+-----+
|                    2 |
+-----+
1 row in set (0.00 sec)

MariaDB [test]>

```

Pada perintah sql di atas, kita akan menampilkan berapa barang yang memiliki jumlah barang sebanyak 30. Dengan memanggil fungsi tampil_jml_brg, maka terdapat 2 barang yang memiliki jumlah 30.

Keterangan :

1. DELIMITER : berfungsi untuk menginfokan kepada mysql tentang delimite yang digunakan secara default adalah menggunakan (;) jadi bila ada tanda (;) mysql akan mengartikan akhir statement, pada contoh dia atas, DELIMITER yang digunakan (\$\$), jadi akhir statement adalah (\$\$).
2. CREATE FUNCTION : header untuk membuat function.
3. RETURN : untuk menentukan tipe data yang di return-kan oleh function.
4. DETERMINISTIC/NOT DETERMINISTIC : untuk menentukan yang dapat menggunakan function ini adalah user pembuatnya saja (deterministic) atau user siapa saja (not deterministic).

Melihat Function

Untuk melihat fungsi apasaja yang telah dibuat, maka kita dapat menggunakan perintah SHOW FUNCTION STATUS.

LATIHAN

1. Tambahkan field harga pada tabel beli, isi harga pensil 5.000, buku 20.000
2. Buatlah fungsi untuk menampilkan harga total pembelian pensil. Diskon 10% untuk pensil.
3. Buatlah fungsi untuk menampilkan harga total pembelian buku. Diskon 5% untuk buku dibawah atau sama dengan 5, diskon 15% untuk buku diatas 5.