

PRAKTEK DBMS LANJUT

M6-MYSQL TRIGGER

Gunakan database **db_sp_noabsen_nama** dan **tabel** yang anda buat dipertemuan minggu ke-4, praktekan materi di bawah ini.

TRIGGER adalah kumpulan kode SQL yang berjalan secara otomatis untuk mengeksekusi perintah INSERT, UPDATE, DELETE.

Biasanya TRIGGER akan dijalankan sebelum atau sesudah proses INSERT, UPDATE, DELETE (Perintah DML)

Cara penulisan TRIGGER

```
DELIMITER $$
CREATE TRIGGER nama_trigger
{BEFORE | AFTER} {INSERT | UPDATE | DELETE }
    ON nama_table
    FOR EACH ROW
BEGIN
    KODE SQL
END$$
DELIMITER ;
```

Untuk memulai menggunakan TRIGGER kita gunakan **CREATE TRIGGER** dilanjutkan nama TRIGGER yang ingin dibuat

{BEFORE | AFTER} adalah waktu TRIGGER akan dijalankan, apakah sebelum atau sesudah database dimodifikasi oleh perintah DML

{INSERT | UPDATE | DELETE} adalah perintah DML yang mengaktifkan TRIGGER

Lebih detail waktu TRIGGER akan dijelaskan di tabel berikut

Waktu TRIGGER	Keterangan TRIGGER
BEFORE INSERT	TRIGGER dijalankan sebelum record dimasukkan ke database
AFTER INSERT	TRIGGER dijalankan sesudah record dimasukkan ke database
BEFORE UPDATE	TRIGGER dijalankan sebelum record dirubah di database
AFTER UPDATE	TRIGGER dijalankan sesudah record dirubah database

BEFORE DELETE	TRIGGER dijalankan sebelum record dihapus di database
AFTER DELETE	TRIGGER dijalankan sesudah record dihapus di database

ON mendefinisikan table yang mengaktifkan TRIGGER

BEGIN END adalah pernyataan yang membungkus kode TRIGGER

Pastikan diawal gunakan **DELIMITER \$\$** dan diakhir dikembalikan ke **DELIMITER ;** seperti dalam membuat Stored Procedure

Hands On

Pada Hands-On TRIGGER akan dibuat 2 table yaitu **table mahasiswa** dan **table log_mahasiswa**

Table mahasiswa -> menyimpan data mahasiswa

Table log_mahasiswa -> menyimpan perubahan data mahasiswa

Jadi setiap ada perubahan data (UPDATE) alamat mahasiswa pada **table mahasiswa** maka akan disimpan di **table log_mahasiswa** tentang histori perubahan data alamat tersebut.

Dengan adanya log perubahan data mahasiswa maka akan memudahkan dalam melihat histori data mahasiswa yang pernah berubah dalam sistem.

Table mahasiswa

```
1 CREATE TABLE mahasiswa
2 (
3     nim INT(10),
4     nama VARCHAR(100),
5     alamat VARCHAR(100),
6     PRIMARY KEY(nim)
7 );
8
9 INSERT INTO mahasiswa
10 VALUES
11 (21400200,"faqih","bandung"),
12 (21400201,"ina","jakarta"),
13 (21400202,"anto","semarang"),
14 (21400203,"dani","padang"),
15 (21400204,"jaka","bandung"),
16 (21400205,"nara","bandung"),
17 (21400206,"senta","semarang");
```

Table log_mahasiswa

```
1 CREATE TABLE log_mahasiswa
2 (
3     id_log INT(10) AUTO_INCREMENT,
4     nim INT(10),
5     alamat_lama VARCHAR(100),
6     alamat_baru VARCHAR(100),
7     waktu DATE,
8     PRIMARY KEY(id_log)
9 );
```

Membuat TRIGGER

Kita akan menyimpan data perubahan alamat sebelum perintah UPDATE dijalankan

```

1 DELIMITER $$
2 CREATE TRIGGER update_alamat_mahasiswa
3 BEFORE UPDATE
4 ON mahasiswa
5 FOR EACH ROW
6 BEGIN
7     INSERT INTO log_mahasiswa
8     set nim = OLD.nim,
9     alamat_lama=old.alamat,
10    alamat_baru=new.alamat,
11    waktu = NOW();
12 END$$
13 DELIMITER ;

```

Keyword **OLD** digunakan untuk mengambil data kolom di table yang lama sedangkan keyword **NEW** digunakan untuk mengambil data kolom di table yang baru

Sekarang kita akan coba update alamat mahasiswa dengan NIM 21400200. Sebelum diupdate alamat mahasiswa dengan NIM 21400200 adalah “bandung”

Kita ganti alamat “bandung” menjadi “surabaya”

```

1 UPDATE mahasiswa
2 SET alamat = 'surabaya'
3 WHERE nim = 21400200;

```

Sekarang coba lakukan perintah SELECT untuk melihat **table log_mahasiswa**

```

1 > SELECT * FROM log_mahasiswa;
2 +-----+-----+-----+-----+-----+
3 | id_log | nim      | alamat_lama | alamat_baru | waktu      |
4 +-----+-----+-----+-----+-----+
5 |      1 | 21400200 | bandung     | surabaya    | 2019-11-02 |
6 +-----+-----+-----+-----+-----+
7 1 row in set (0.00 sec)
8

```

Oke, record baru secara otomatis telah ditambahkan ke **table log_mahasiswa** untuk mahasiswa dengan NIM 21400200 yang telah diubah alamat awal "bandung" menjadi "surabaya"

Sedangkan pada table mahasiswa alamat yang tercantum adalah alamat yang baru

```
1 > SELECT * FROM mahasiswa WHERE nim=21400200;
2 +-----+-----+-----+
3 | nim      | nama  | alamat |
4 +-----+-----+-----+
5 | 21400200 | faqih | surabaya |
6 +-----+-----+-----+
7 1 row in set (0.00 sec)
```

TUGAS MINGGU 6:

Screenshot setiap tahapan dan hasilnya, upload file dokumentasi ke elearning, nama file: **DBMSL-20211-REG-KELAS-M6-NOABSEN-NAMA.docx**