**LAPORAN PRAKTIKUM BASIS DATA**

**TRIGGER**



**DISUSUN OLEH :**

Restu Wibisono

2340506061

**PROGRAM STUDI S1 TEKNOLOGI INFORMASI**

**JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK**

**UNIVERSITAS TIDAR**

**2024**

**LAPORAN**

**PRAKTIKUM BASIS DATA**



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Diisi Mahsiswa Praktikan** | | | | | | | | |
| Nama Praktikan | Restu Wibisono | | | | | | | |
| NPM | 2340506061 | | | | | | | |
| Rombel | 03 | | | | | | | |
| Judul Praktikum | Trigger | | | | | | | |
| Tanggal Praktikum | 30 Mei 2024 | | | | | | | |
| **Diisi Asisten Praktikum** | | | | | | | | |
| Tanggal Pengumpulan |  |  |  |  |  |  |  |  |
| Catatan |  | | | | | | | |

|  |  |  |
| --- | --- | --- |
| PENGESAHAN | | NILAI |
| Diperiksa oleh : | Disahkan oleh : |  |
| Asisten Praktikum | Dosen Pengampu |
|  |  |
| Nanda Cahya Septiawan | Imam Adi Nata, S.Kom., M.Kom. |

**PROGRAM STUDI S1 TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS TIDAR**

**Genap 2023/2024**

## Tujuan Praktikum

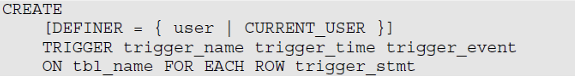
1. Mahasiswa mampu menjelaskan trigger dalam SQL
2. Mahasiswa mampu mengimplementasikan trigger dalam SQL

## Dasar Teori

1. TRIGGER Trigger dapat didefinisikan sebagai himpunan kode (prosedural) yang dieksekusi secara otomatis sebagai respon atas suatu kejadian berkaitan dengan tabel basis data. Kejadian (event) yang dapat membangkitkan trigger umumnya berupa pernyataan INSERT, UPDATE, dan DELETE. Berdasarkan ruang lingkupnya, trigger diklasifikasikan menjadi dua jenis: row trigger dan statement trigger. Trigger baris (row) mendefinisikan aksi untuk setiap baris tabel; trigger pernyataan hanya berlaku untuk setiap pernyataan INSERT, UPDATE, atau DELETE. Dari sisi perilaku (behavior) eksekusi, trigger dapat dibedakan menjadi beberapa jenis; namun umumnya ada dua jenis: trigger BEFORE dan AFTER. Sesuai penamaannya, jenis-jenis ini merepresentasikan waktu eksekusi trigger—misalnya sebelum ataukah sesudah pernyataan- pernyataan yang berkorespondensi.

Adakalanya trigger dipandang sebagai bentuk spesifik dari stored procedure (terkait pendefinisian body). Bagaimanapun, trigger akan dipanggil (secara otomatis) ketika event terjadi, sedangkan stored procedure harus dipanggil secara eksplisit.

1. Trigger dalam MySQL MySQL mendukung fitur trigger termasuk juga stored procedure dan view sejak versi 5.0.2 Sebagaimana objek-objek lainnya, trigger diciptakan menggunakan pernyataan CREATE. Sintaks pendefinisian trigger diperlihatkan sebagai berikut:



MySQL tidak mengizinkan multiple trigger dengan waktu aksi dan event sama per tabel. Misalkan di tabel A sudah didefinisikan trigger AFTER INSERT, maka kita tidak boleh mendefinisikan trigger AFTER INSERT lagi; namun AFTER EDIT, AFTER DELETE, atau BEFORE (INSERT, EDIT, dan DELETE) bisa diterima.

## Metode Praktikum

## Alat dan bahan

Alat :

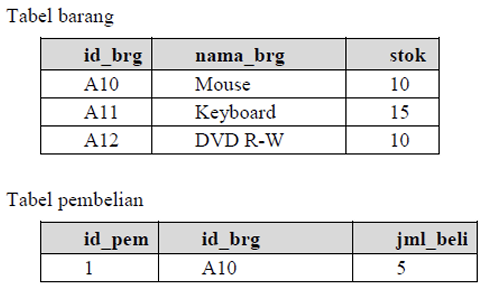
1. PC (Komputer)
2. Keyboard
3. Mouse

Bahan :

1. Operating System Windows 10
2. File Materi Praktikum
3. Aplikasi Paket Web server XAMPP
4. Aplikasi Kantor (Microsoft Office/Libre Office/WPS Office/etc)

## Langkah kerja (diisi seperti dengan modul)

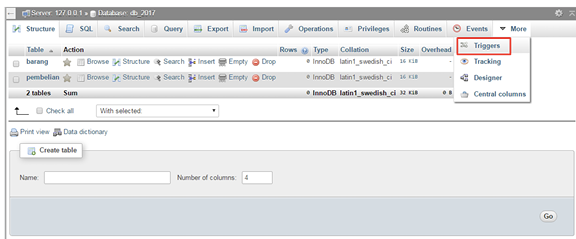
Dalam praktikum ini digunakan dua buah tabel bernama barang dan pembelian dengan data sebagai seperti berikut:



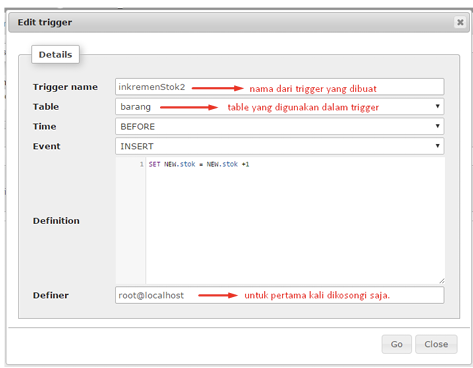
1. Menggunakan Trigger

Operasi-operasi berkenaan dengan pendefinisian trigger tidak berbeda dengan objek-objek database lainnya.

1. Masuk kedalam database yang digunakan untuk menciptakan tabel tadi, kemudian masuk ke menu Trigger.



1. Pilih Add Trigger pada bagian New. Dan isikan sesuai dengan gambar berikut.



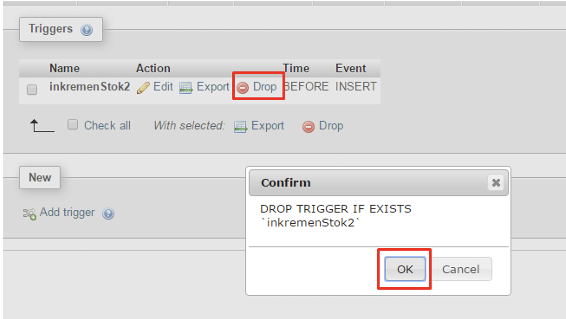
1. Setelah berhasil membuat trigger. Kembali ke SQL database dan tuliskan sintak berikut:



1. Jika berhasil maka akan menampilkan data seperti pada gambar berikut

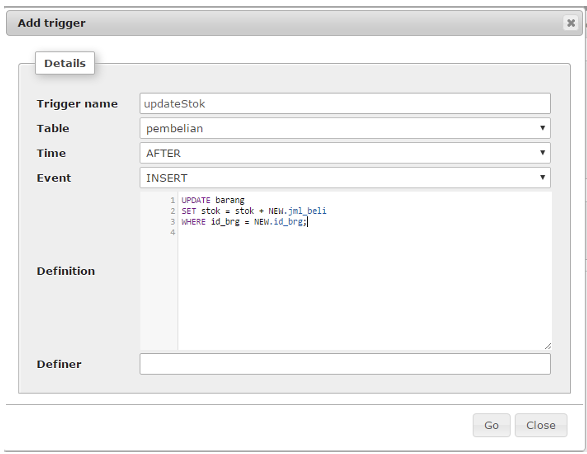


1. Sebagaimana objek-objek database lainnya, kita menghapus trigger dengan menggunakan perintah DROP dengan ketentuan DROP TRIGGER nama\_trigger. Atau bisa langsung ke menu trigger dan menghapusnya dengan cara berikut.



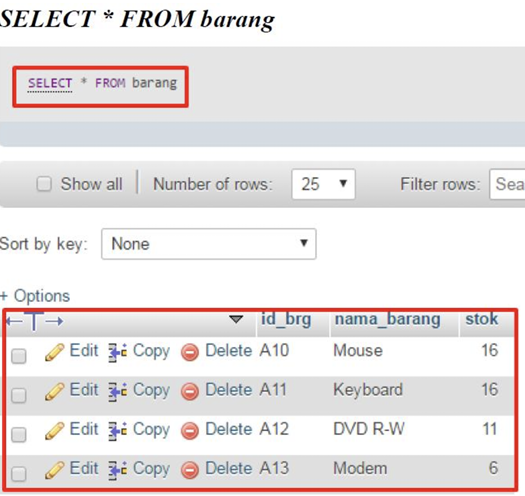
1. Keyword OLD dan NEW

Untuk merujuk ke kolom-kolom tabel yang diasosiasikan dengan trigger, kita menggunakan keyword OLD dan NEW. Keyword OLD mengacu pada nilai lama, sedangkan NEW merepresentasikan nilai baru. Di trigger INSERT, kita hanya dapat menggunakan keyword NEW karena tidak ada data lama.

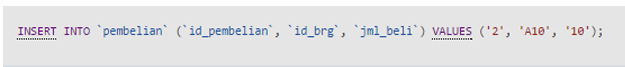


Pada contoh di atas, penambahan data pembelian akan mengakibatkan nilai stok barang berubah menyesuaikan banyaknya nilai jumlah pembelian.

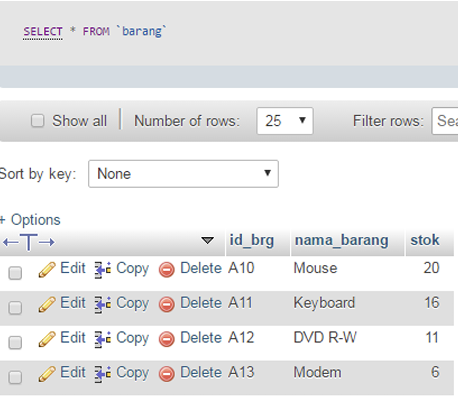
1. Cek tabel barang



1. Kemudian masukkan data kedalam table pembelian

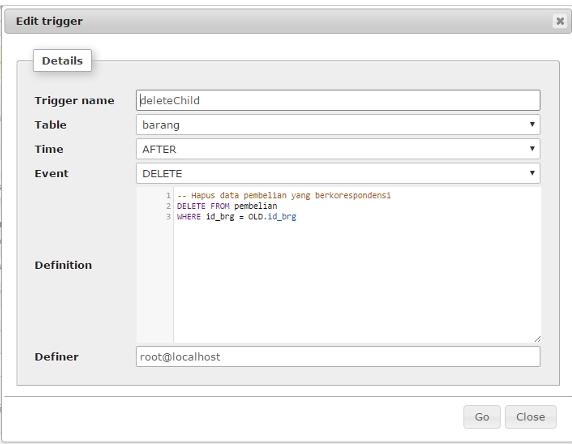


1. Cek kembali data dalam table barang



Untuk kasus trigger DELETE, keyword yang bisa digunakan hanya OLD. Misalkan kita ingin mendefinisikan trigger untuk menghapus semua data pembelian manakala data barang yang sesuai—diindikasikan melalui primary key dan foreign key—dihapus.

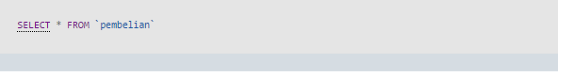
1. Membuat trigger baru seperti langkah sebelumnya



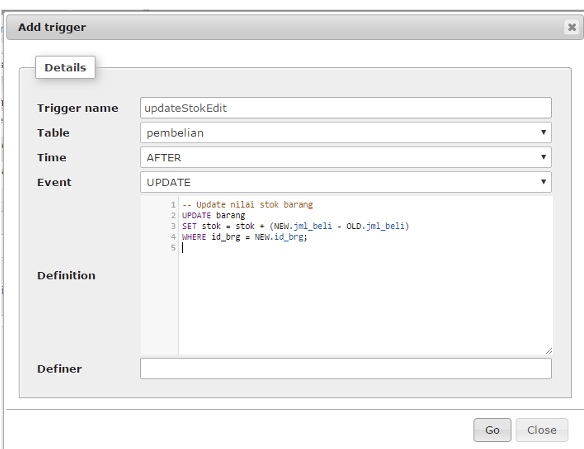
1. Hapus data pada tabel barang



1. Cek apakah trigger berjalan dengan baik dengan menampilkan tabel pembelian



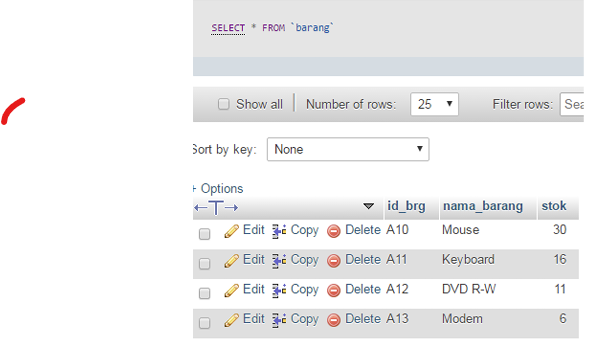
Khusus untuk operasi UPDATE, kita bisa memanfaatkan keyword NEW maupun OLD.



1. Lakukan update pada tabel pembelian

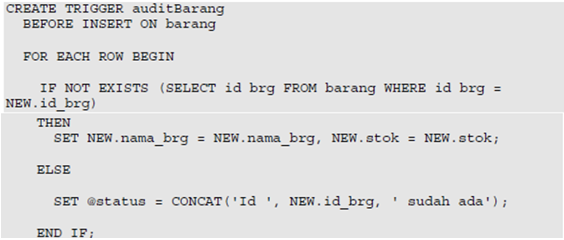


1. Cek apakah trigger berjalan dengan baik dengan cara tampilkan pada tabel barang



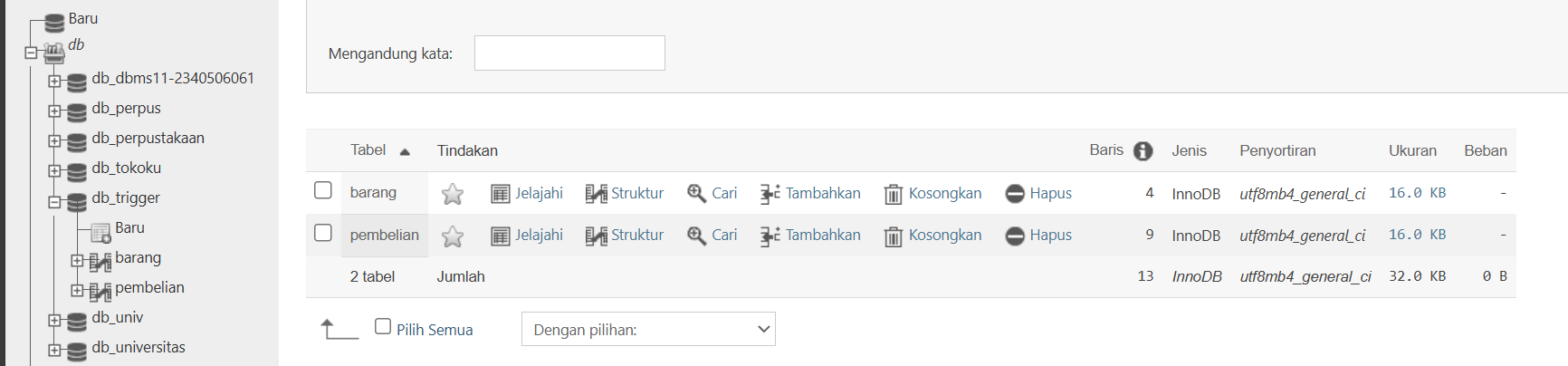
1. Trigger Kompleks

Keberadaan trigger secara nyata mampu mengatasi berbagai persoalan sulit, misalnya berkaitan dengan integritas atau audit data. Ini tentu sangat masuk akal, karena trigger juga bisa mengandung pernyataan-pernyataan yang kompleks termasuk pencabangan, pengulangan, fungsi-fungsi agregat, bahkan kode pemanggilan prosedur. Sebagai ilustrasi sederhana, kita bisa mendefinisikan trigger untuk memeriksa operasi penambahan data barang. Skenarionya, jika data sudah ada, berikan status eksistensi barang; sebaliknya, data bisa langsung dimasukkan.



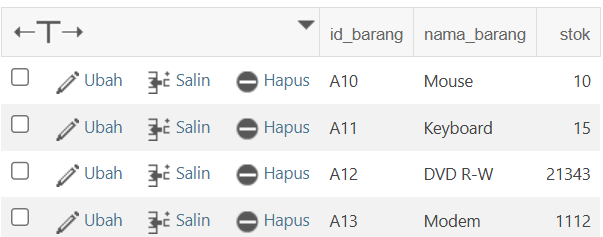
## Soal

1. Membuat database



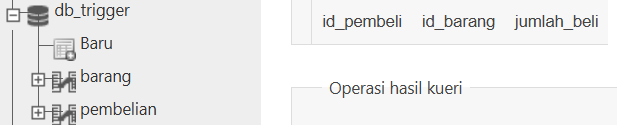
(Gambar 4.1.1)

Membuat database ‘Trigger’ yang berisikan 2 tabel yaitu table ‘barang’ dan table ‘pembelian’.



(Gambar 4.1.2 TABEL BARANG)

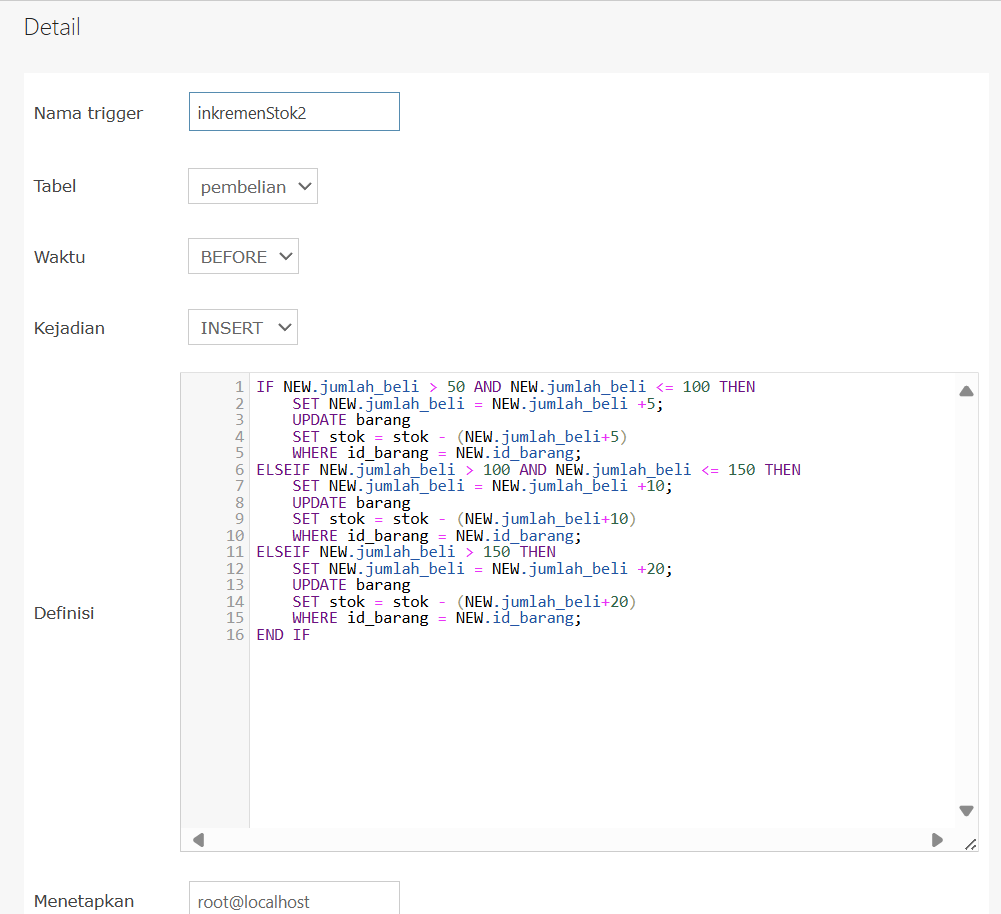
Tabel barang berisikan 3 kolom, dan kolom tersebut diisi dengan barang barang yang memiliki id dan jumlah stok yang ada.



(Gambar 4.1.3 TABEL PEMBELIAN)

Tabel pembelian nantinya akan direlasi dengan table barang, dan jika ada barang yang dibeli akan tercatat dalam table pembelian.

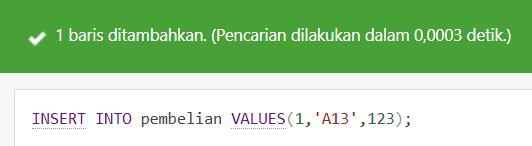
1. Membuat TRIGGER



(Gambar 4.2.1)

Membuat sebuah trigger INSERT pembelian untuk menambahkan fitur bonus di dalamnya. Aturannya adalah jika pembelian > 50 dan < 100, maka bonus = 5; jika pembelian > 100 dan < 150, maka bonus = 10; jika pembelian > 150, maka bonus = 20. Ingat, aturan penyesuaian stok barang masih berlaku, setiap ada pembelian maka stok akan berkurang.

1. Membuat perintah SQL



(Gambar 4.3.1)

Pada VALUES terdapat 3 bagian, pertama adalah id prmbeli, selanjutnya id barang yang dibeli dan terkahir adalah jumlah barang yang dibeli.

## Kesimpulan

Dengan memanfaatkan trigger, mahasiswa dapat melihat bagaimana perubahan data dapat dikelola secara efisien dan otomatis sesuai dengan aturan yang telah ditetapkan. Namun, penting juga untuk memahami bahwa penggunaan trigger harus dilakukan dengan hati-hati untuk menghindari dampak negatif pada kinerja basis data. Praktikum ini memberikan pemahaman praktis tentang bagaimana merancang dan mengimplementasikan trigger dengan tepat untuk meningkatkan efisiensi dan keandalan sistem basis data.

## Referensi

Van Rossum, G. (2003). An introduction to Python (p. 115). F. L. Drake (Ed.).

Bristol: Network Theory Ltd.. [*http://atk.fam.free.fr/fichiers/stage/Python/JF/site/pytut.pdf*](http://atk.fam.free.fr/fichiers/stage/Python/JF/site/pytut.pdf)

Kuhlman, D. (2009). A python book: Beginning python, advanced python, and python exercises (pp. 1-227). Lutz: Dave Kuhlman.

[*https://www.davekuhlman.org/python\_book\_01.pdf*](https://www.davekuhlman.org/python_book_01.pdf)

Python, W. (2021). Python. Python Releases for Windows, 24. [*https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1f2ee3831eebfc97 bfafd514ca2abb7e2c5c86bb*](https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1f2ee3831eebfc97bfafd514ca2abb7e2c5c86bb)