**TUGAS 9 NF ACADEMY**

**FULLSTACK WEB DEVELOPER**

**‘Database MySQL/MariaDB’**

Nama : Agnes Yosefina Sura

Soal 6.2

*Trigger*

1. Buatlah bisnis proses pembayaran dengan menggunakan trigers, dengan skenario sebagai berikut :

- pelanggan memesan didalam table pesanan

- dilanjutkan dengan proses pembayaran di table pembayaran

- didalam table pembayaran tambahkan kolom status\_pembayaran

- jika pesanan sudah dibayar maka status pembayaran akan berubah menjadi lunas

1. Pelanggan memesan didalam table pesanan

*SELECT \* FROM pesanan;*

*Didalam table pembayaran tambahkan kolom status\_pembayaran*

*ALTER TABLE pembayaran ADD status\_pembayaran varchar(25);*

2. Dilanjutkan dengan proses pembayaran di table pembayaran

*DELIMITER $$*

*CREATE TRIGGER cek\_pembayaran BEFORE INSERT ON pembayaran*

*FOR EACH ROW*

*BEGIN*

*DECLARE total\_bayar DECIMAL(10, 2);*

*DECLARE total\_pesanan DECIMAL(10, 2);*

*SELECT SUM(jumlah) INTO total\_bayar FROM pembayaran WHERE pesanan\_id = NEW.pesanan\_id;*

*SELECT total INTO total\_pesanan FROM pesanan WHERE id = NEW.pesanan\_id;*

3. Jika pesanan sudah dibayar maka status pembayaran akan berubah menjadi lunas

*IF total\_bayar + NEW.jumlah >= total\_pesanan THEN*

*SET NEW.status\_pembayaran = 'Lunas';*

*END IF;*

*END $$*

*DELIMITER ;*

4. Menambahkan data pada tabel pembayaran

*INSERT INTO pembayaran (no\_kuitansi, tanggal, jumlah, ke, pesanan\_id, status\_pembayaran)*

*VALUES ('KWI001', '2023-03-03', 200000, 1, 1);*

*ALTER TABLE pembayaran ADD COLUMN status\_pembayaran VARCHAR(25);*

*DELIMITER $$*

*CREATE TRIGGER cek\_pembayaran*

*BEFORE INSERT ON pembayaran*

*FOR EACH ROW*

*BEGIN*

*DECLARE total\_bayar DECIMAL(10, 2);*

*DECLARE total\_pesanan DECIMAL(10, 2);*

*SELECT COALESCE(SUM(jumlah), 0) INTO total\_bayar*

*FROM pembayaran*

*WHERE pesanan\_id = NEW.pesanan\_id;*

*SELECT total INTO total\_pesanan*

*FROM pesanan*

*WHERE id = NEW.pesanan\_id;*

*IF total\_bayar + NEW.jumlah >= total\_pesanan THEN*

*SET NEW.status\_pembayaran = 'Lunas';*

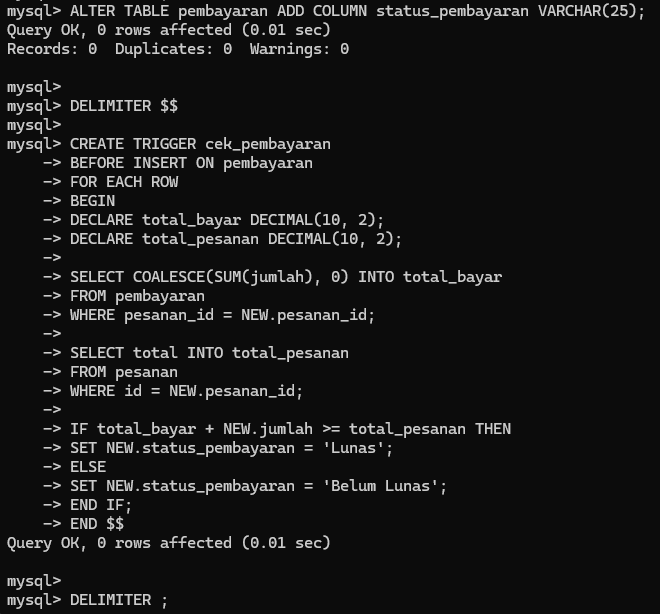
*ELSE*

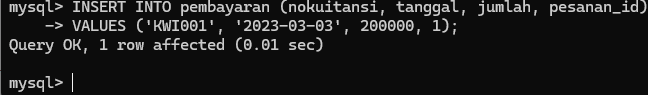
*SET NEW.status\_pembayaran = 'Belum Lunas';*

*END IF;*

*END $$*

*DELIMITER ;*





1. Buatlah Stored Procedure dengan nama **kurangi\_stok** untuk mengurangi stok produk. Stok berkurang sesuai dengan jumlah pesanan produk.

*DELIMITER $$*

*CREATE PROCEDURE kurangi\_stok(*

*IN p\_produk\_id INT,*

*IN p\_jumlah INT*

*)*

*BEGIN*

*DECLARE current\_stok INT;*

*SELECT stok INTO current\_stok*

*FROM produk*

*WHERE id = p\_produk\_id;*

*IF current\_stok >= p\_jumlah THEN*

*UPDATE produk*

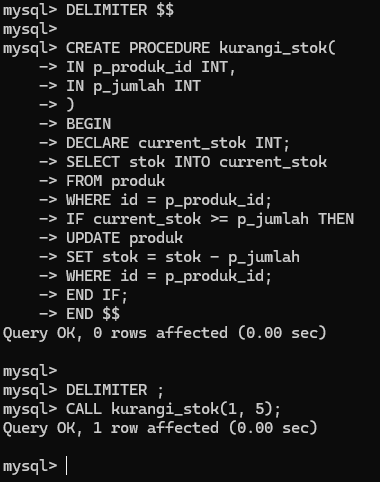
*SET stok = stok - p\_jumlah*

*WHERE id = p\_produk\_id;*

*END IF;*

*END $$*

*DELIMITER ;*

**

1. Buatlah Trigger dengan nama **trig\_kurangi\_stok** yang akan mengurangi stok produk jika terjadi transaksi pesanan oleh pelanggan (memanggil stored procedure kurangi\_stok soal no 2).

Trigger ini aktif setelah trigger **after\_pesanan\_items\_insert** (trigger pada contoh 3).

*DELIMITER $$*

*CREATE TRIGGER trig\_kurangi\_stok*

*AFTER INSERT ON pesanan\_items*

*FOR EACH ROW*

*BEGIN*

*CALL kurangi\_stok(NEW.produk\_id, NEW.qty);*

*END $$*

*DELIMITER ;*

**