Stored Procedures, Stored Functions dan Trigger

Worksheet 6

Nama : Muhammad Rizky Fauzi Nurrohman

SOAL 6.1

1. Buatlah Procedure untuk mengupdate harga\_jual berdasarkan jenis produk tertentu (jenis\_produk\_id), beri nama procedure **pro\_naikan\_harga** memiliki parameter yang akan menerima argumen: Jenis Produk ID dan Persentase kenaikan harga.

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> CREATE PROCEDURE pro\_naik(

-> IN jenis\_produk INT,

-> IN persentasi\_kenaikan INT)

-> BEGIN

-> UPDATE produk SET harga\_jual = harga\_jual + (harga\_jual \* persentasi\_kenaikan /100)

-> WHERE jenis\_produk\_id = jenis\_produk;

-> END $$

Query OK, 0 rows affected (0.004 sec)

MariaDB [dbkoperasi]> DELIMITER ;

MariaDB [dbkoperasi]> SELECT nama,harga\_jual,jenis\_produk\_id FROM produk;

+------------------+------------+-----------------+

| nama | harga\_jual | jenis\_produk\_id |

+------------------+------------+-----------------+

| Televisi 21 inch | 5040000 | 1 |

| Televisi 40 inch | 7440000 | 1 |

| Kulkas 2 pintu | 4680000 | 1 |

| Meja Makan | 600000 | 2 |

| Teh Kotak | 3500 | 4 |

| PC Desktop HP | 9600000 | 5 |

| Teh Botol | 2500 | 4 |

| Notebook Acer | 10800000 | 5 |

| Notebook Lenovo | 12000000 | 5 |

| Laptop HP | 13000000 | 5 |

+------------------+------------+-----------------+

10 rows in set (0.001 sec)

MariaDB [dbkoperasi]> CALL pro\_naik(5, 4);

Query OK, 4 rows affected (0.004 sec)

MariaDB [dbkoperasi]> SELECT nama,harga\_jual,jenis\_produk\_id FROM produk;

+------------------+------------+-----------------+

| nama | harga\_jual | jenis\_produk\_id |

+------------------+------------+-----------------+

| Televisi 21 inch | 5040000 | 1 |

| Televisi 40 inch | 7440000 | 1 |

| Kulkas 2 pintu | 4680000 | 1 |

| Meja Makan | 600000 | 2 |

| Teh Kotak | 3500 | 4 |

| PC Desktop HP | 9984000 | 5 |

| Teh Botol | 2500 | 4 |

| Notebook Acer | 11232000 | 5 |

| Notebook Lenovo | 12480000 | 5 |

| Laptop HP | 13520000 | 5 |

+------------------+------------+-----------------+

10 rows in set (0.000 sec)

1. Buat fungsi **umur** dengan parameter yang menerima inputan argumen tipe data date dan mengembalikan hasil perhitungan umur (tahun sekarang dikurang tahun inputan) dengan tipe data bilangan bulat (integer) positif.

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> CREATE FUNCTION umur(tgl\_lahir DATE)

-> RETURNS INT

-> BEGIN

-> DECLARE umur INT;

-> SET umur = YEAR(CURDATE()) - YEAR(tgl\_lahir);

-> RETURN umur;

-> END $$

Query OK, 0 rows affected (0.005 sec)

MariaDB [dbkoperasi]> DELIMITER ;

MariaDB [dbkoperasi]> SELECT kode, nama, jk, umur(tgl\_lahir) AS umur FROM pelanggan;

+------+---------------+------+------+

| kode | nama | jk | umur |

+------+---------------+------+------+

| C001 | Agung Sedayu | L | 13 |

| C002 | Pandan Wangi | P | 73 |

| C003 | Sekar Mirah | P | 40 |

| C004 | Swandaru Geni | L | 42 |

| C005 | Pradabashu | L | 38 |

| C006 | Gayatri Dwi | P | 36 |

| C007 | Dewi Gyat | P | 35 |

| C008 | Andre Haru | L | 33 |

| C009 | Ahmad Hasan | L | 31 |

| C010 | Cassanndra | P | 33 |

+------+---------------+------+------+

10 rows in set (0.002 sec)

1. Buat fungsi **kategori\_harga** dengan parameter yang menerima inputan argument tipe data double dan mengembalikan tipe data string kategori harga berdasarkan:

* 0 – 500rb : murah
* 500rb – 3 juta : sedang
* 3jt – 10 juta : mahal
* > 10 juta : sangat mahal

*CREATE FUNCTION ...*

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> CREATE FUNCTION kategori\_harga(harga DOUBLE) RETURNS VARCHAR(20)

-> BEGIN

-> DECLARE kategori VARCHAR(20);

-> IF harga <= 500000 THEN

-> SET kategori = 'murah';

-> ELSEIF harga > 500000 AND harga <= 3000000 THEN

-> SET kategori = 'sedang';

-> ELSEIF harga > 3000000 AND harga <= 10000000 THEN

-> SET kategori = 'mahal';

-> ELSE

-> SET kategori = 'sangat mahal';

-> END IF;

-> RETURN kategori;

-> END $$

Query OK, 0 rows affected (0.004 sec)

MariaDB [dbkoperasi]> DELIMITER ;

MariaDB [dbkoperasi]> SELECT kategori\_harga(400000);

+------------------------+

| kategori\_harga(400000) |

+------------------------+

| murah |

+------------------------+

1 row in set (0.000 sec)

MariaDB [dbkoperasi]> SELECT kategori\_harga (4000000);

+--------------------------+

| kategori\_harga (4000000) |

+--------------------------+

| mahal |

+--------------------------+

1 row in set (0.003 sec)

MariaDB [dbkoperasi]> SELECT kategori\_harga(40000000);

+--------------------------+

| kategori\_harga(40000000) |

+--------------------------+

| sangat mahal |

+--------------------------+

1 row in set (0.000 sec)

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

MariaDB [dbkoperasi]> SELECT \* FROM pesanan;

+----+------------+---------+--------------+

| id | tanggal | total | pelanggan\_id |

+----+------------+---------+--------------+

| 1 | 2015-11-04 | 9720000 | 1 |

| 2 | 2015-11-04 | 17500 | 3 |

| 3 | 2015-11-04 | 0 | 6 |

| 4 | 2015-11-04 | 0 | 7 |

| 5 | 2015-11-04 | 0 | 10 |

| 6 | 2015-11-04 | 0 | 2 |

| 7 | 2015-11-04 | 0 | 5 |

| 8 | 2015-11-04 | 0 | 4 |

| 9 | 2015-11-04 | 0 | 8 |

| 10 | 2015-11-04 | 0 | 9 |

| 11 | 2015-11-04 | 30000 | 9 |

+----+------------+---------+--------------+

11 rows in set (0.000 sec)

Didalam table pembayaran tambahkan kolom status\_pembayaran

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

1. Dilanjutkan dengan proses pembayaran di table pembayaran

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> 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;

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

-> SET NEW.status\_pembayaran = 'Lunas';

-> END IF;

-> END $$

Query OK, 0 rows affected (0.009 sec)

// Menambahkan data pada tabel pembayaran

MariaDB [dbkoperasi]> INSERT INTO pembayaran (nokuitansi, tanggal, jumlah, ke, pesanan\_id,

-> status\_pembayaran) VALUES ('MD001','2023-10-10',30000,1,11, '');

Query OK, 1 row affected (0.008 sec)

MariaDB [dbkoperasi]> select \* from pembayaran;

+----+------------+------------+--------+------+------------+-------------------+

| id | nokuitansi | tanggal | jumlah | ke | pesanan\_id | status\_pembayaran |

+----+------------+------------+--------+------+------------+-------------------+

| 1 | MD001 | 2023-10-10 | 30000 | 1 | 11 | Lunas |

| 2 | MD002 | 0000-00-00 | 30000 | 2 | 11 | Lunas |

| 3 | MD003 | 2023-10-10 | 30000 | 1 | 11 | Lunas |

| 4 | MD004 | 2023-10-10 | 15000 | 1 | 2 | |

| 5 | MD005 | 2023-10-10 | 18000 | 2 | 2 | Lunas |

| 6 | MD006 | 2023-10-18 | 15000 | 2 | 2 | Lunas |

+----+------------+------------+--------+------+------------+-------------------+

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

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> CREATE PROCEDURE kurangi\_stok(IN produk\_id INT, IN jumlah\_pesanan INT)

-> BEGIN

-> DECLARE stok\_produk INT;

-> SELECT stok INTO stok\_produk WHERE id = produk\_id;

-> SET stok\_produk = stok\_produk - jumlah\_pesanan;

-> IF stok\_produk < 0 THEN

-> SET stok\_produk = 0;

-> END IF;

-> UPDATE produk SET stok = stok\_produk WHERE id = produk\_id;

-> END $$

Query OK, 0 rows affected (0.005 sec)

MariaDB [dbkoperasi]> DELIMITER ;

MariaDB [dbkoperasi]> SELECT \* FROM produk;

+----+------+------------------+------------+------------+------+----------+-----------------+

| id | kode | nama | harga\_beli | harga\_jual | stok | min\_stok | jenis\_produk\_id |

+----+------+------------------+------------+------------+------+----------+-----------------+

| 1 | TV01 | Televisi 21 inch | 3500000 | 5040000 | 5 | 2 | 1 |

| 2 | TV02 | Televisi 40 inch | 5500000 | 7440000 | 4 | 2 | 1 |

| 3 | K001 | Kulkas 2 pintu | 3500000 | 4680000 | 6 | 2 | 1 |

| 4 | M001 | Meja Makan | 500000 | 600000 | 4 | 3 | 2 |

| 5 | TK01 | Teh Kotak | 3000 | 3500 | 6 | 10 | 4 |

| 6 | PC01 | PC Desktop HP | 7000000 | 9984000 | 9 | 2 | 5 |

| 7 | TB01 | Teh Botol | 2000 | 2500 | 53 | 10 | 4 |

| 8 | AC01 | Notebook Acer | 8000000 | 11232000 | 7 | 2 | 5 |

| 9 | LN01 | Notebook Lenovo | 9000000 | 12480000 | 9 | 2 | 5 |

| 10 | L004 | Laptop HP | 12000000 | 13520000 | 20 | 5 | 5 |

+----+------+------------------+------------+------------+------+----------+-----------------+

10 rows in set (0.000 sec)

MariaDB [dbkoperasi]> CALL kurangi\_stok(2, 2);

Query OK, 2 rows affected (0.003 sec)

MariaDB [dbkoperasi]> SELECT \* FROM produk;

+----+------+------------------+------------+------------+------+----------+-----------------+

| id | kode | nama | harga\_beli | harga\_jual | stok | min\_stok | jenis\_produk\_id |

+----+------+------------------+------------+------------+------+----------+-----------------+

| 1 | TV01 | Televisi 21 inch | 3500000 | 5040000 | 5 | 2 | 1 |

| 2 | TV02 | Televisi 40 inch | 5500000 | 7440000 | 2 | 2 | 1 |

| 3 | K001 | Kulkas 2 pintu | 3500000 | 4680000 | 6 | 2 | 1 |

| 4 | M001 | Meja Makan | 500000 | 600000 | 4 | 3 | 2 |

| 5 | TK01 | Teh Kotak | 3000 | 3500 | 6 | 10 | 4 |

| 6 | PC01 | PC Desktop HP | 7000000 | 9984000 | 9 | 2 | 5 |

| 7 | TB01 | Teh Botol | 2000 | 2500 | 53 | 10 | 4 |

| 8 | AC01 | Notebook Acer | 8000000 | 11232000 | 7 | 2 | 5 |

| 9 | LN01 | Notebook Lenovo | 9000000 | 12480000 | 9 | 2 | 5 |

| 10 | L004 | Laptop HP | 12000000 | 13520000 | 20 | 5 | 5 |

+----+------+------------------+------------+------------+------+----------+-----------------+

10 rows in set (0.000 sec)

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 1).

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

MariaDB [dbkoperasi]> DELIMITER $$

MariaDB [dbkoperasi]> CREATE TRIGGER trig\_kurangi\_stok

-> AFTER INSERT ON pesanan\_items FOR EACH ROW

-> BEGIN

-> DECLARE produk\_id INT;

-> DECLARE jumlah\_pesanan INT;

-> SELECT NEW.produk\_id, NEW.qty INTO produk\_id, jumlah\_pesanan;

-> CALL kurangi\_stok(produk\_id, jumlah\_pesanan);

-> END $$

Query OK, 0 rows affected (0.007 sec)

MariaDB [dbkoperasi]> DELIMITER ;

MariaDB [dbkoperasi]> SELECT \* FROM produk;

+----+------+------------------+------------+------------+------+----------+-----------------+

| id | kode | nama | harga\_beli | harga\_jual | stok | min\_stok | jenis\_produk\_id |

+----+------+------------------+------------+------------+------+----------+-----------------+

| 1 | TV01 | Televisi 21 inch | 3500000 | 5040000 | 5 | 2 | 1 |

| 2 | TV02 | Televisi 40 inch | 5500000 | 7440000 | 2 | 2 | 1 |

| 3 | K001 | Kulkas 2 pintu | 3500000 | 4680000 | 6 | 2 | 1 |

| 4 | M001 | Meja Makan | 500000 | 600000 | 4 | 3 | 2 |

| 5 | TK01 | Teh Kotak | 3000 | 3500 | 6 | 10 | 4 |

| 6 | PC01 | PC Desktop HP | 7000000 | 9984000 | 9 | 2 | 5 |

| 7 | TB01 | Teh Botol | 2000 | 2500 | 53 | 10 | 4 |

| 8 | AC01 | Notebook Acer | 8000000 | 11232000 | 7 | 2 | 5 |

| 9 | LN01 | Notebook Lenovo | 9000000 | 12480000 | 9 | 2 | 5 |

| 10 | L004 | Laptop HP | 12000000 | 13520000 | 20 | 5 | 5 |

+----+------+------------------+------------+------------+------+----------+-----------------+

10 rows in set (0.000 sec)

MariaDB [dbkoperasi]> SELECT \* FROM pesanan\_items;

+----+-----------+------------+------+---------+

| id | produk\_id | pesanan\_id | qty | harga |

+----+-----------+------------+------+---------+

| 1 | 1 | 1 | 1 | 5040000 |

| 2 | 3 | 1 | 1 | 4680000 |

| 3 | 5 | 2 | 5 | 3500 |

| 6 | 5 | 3 | 10 | 3500 |

| 7 | 1 | 3 | 1 | 5040000 |

| 9 | 5 | 5 | 10 | 3500 |

| 10 | 5 | 6 | 20 | 3500 |

+----+-----------+------------+------+---------+

7 rows in set (0.004 sec)

MariaDB [dbkoperasi]> INSERT INTO pesanan\_items (produk\_id, pesanan\_id, qty, harga) VALUES

-> (1, 2, 1, 19000);

Query OK, 1 row affected (0.003 sec)

MariaDB [dbkoperasi]> SELECT \* FROM pesanan\_items;

+----+-----------+------------+------+---------+

| id | produk\_id | pesanan\_id | qty | harga |

+----+-----------+------------+------+---------+

| 1 | 1 | 1 | 1 | 5040000 |

| 2 | 3 | 1 | 1 | 4680000 |

| 3 | 5 | 2 | 5 | 3500 |

| 6 | 5 | 3 | 10 | 3500 |

| 7 | 1 | 3 | 1 | 5040000 |

| 9 | 5 | 5 | 10 | 3500 |

| 10 | 5 | 6 | 20 | 3500 |

| 11 | 1 | 2 | 1 | 19000 |

+----+-----------+------------+------+---------+

8 rows in set (0.000 sec)

MariaDB [dbkoperasi]> SELECT \* FROM produk;

+----+------+------------------+------------+------------+------+----------+-----------------+

| id | kode | nama | harga\_beli | harga\_jual | stok | min\_stok | jenis\_produk\_id |

+----+------+------------------+------------+------------+------+----------+-----------------+

| 1 | TV01 | Televisi 21 inch | 3500000 | 5040000 | 4 | 2 | 1 |

| 2 | TV02 | Televisi 40 inch | 5500000 | 7440000 | 2 | 2 | 1 |

| 3 | K001 | Kulkas 2 pintu | 3500000 | 4680000 | 6 | 2 | 1 |

| 4 | M001 | Meja Makan | 500000 | 600000 | 4 | 3 | 2 |

| 5 | TK01 | Teh Kotak | 3000 | 3500 | 6 | 10 | 4 |

| 6 | PC01 | PC Desktop HP | 7000000 | 9984000 | 9 | 2 | 5 |

| 7 | TB01 | Teh Botol | 2000 | 2500 | 53 | 10 | 4 |

| 8 | AC01 | Notebook Acer | 8000000 | 11232000 | 7 | 2 | 5 |

| 9 | LN01 | Notebook Lenovo | 9000000 | 12480000 | 9 | 2 | 5 |

| 10 | L004 | Laptop HP | 12000000 | 13520000 | 20 | 5 | 5 |

+----+------+------------------+------------+------------+------+----------+-----------------+

10 rows in set (0.000 sec)