**CODE SQL**

**16\_Join - Union - Agregasi - Subquery - Function (DBMS)**

**DML**

**INSERT : Insert 5 operators pada table operators.**

**INSERT INTO operators (name) values ('Toko A');**

**INSERT INTO operators (name) values ('Toko B');**

**INSERT INTO operators (name) values ('Toko C');**

**INSERT INTO operators (name) values ('Toko D');**

**INSERT INTO operators (name) values ('Toko E');**

**INSERT : Insert 3 product type.**

**INSERT INTO product\_type (name) values ('Toner');**

**INSERT INTO product\_type (name) values ('Krim');**

**INSERT INTO product\_type (name) values ('Serum');**

**INSERT 3 :**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product A', 100, 1, 3);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product B', 200, 1, 3);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product C', 150, 2, 1);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product D', 250, 2, 1);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product E', 250, 2, 1);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product F', 175, 3, 4);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product G', 225, 3, 4);**

**INSERT INTO products (name, price, product\_type\_id, operator\_id) VALUES ('Product H', 275, 3, 4);**

**INSERT 4 :**

**INSERT INTO products (product\_name, description)**

**VALUES ('Product A', 'This is the description for Product A.'),**

**('Product B', 'This is the description for Product B.'),**

**('Product C', 'This is the description for Product C.')**

**INSERT 5 :**

**INSERT INTO payment\_methods (id, name, description)**

**VALUES**

**(1, 'Credit Card', 'Payment using credit card'),**

**(2, 'Bank Transfer', 'Payment using bank transfer'),**

**(3, 'PayPal', 'Payment using PayPal');**

**INSERT 6 :**

**INSERT INTO user (username, email, password)**

**VALUES**

**('user1', 'user1@example.com', 'password1'),**

**('user2', 'user2@example.com', 'password2'),**

**('user3', 'user3@example.com', 'password3'),**

**('user4', 'user4@example.com', 'password4'),**

**('user5', 'user5@example.com', 'password5');**

**INSERT 7 :**

**INSERT INTO tabel\_transaksi (user\_id, tanggal, jumlah)**

**VALUES**

**(1, '2023-03-17', 100),**

**(1, '2023-03-16', 50),**

**(1, '2023-03-15', 75),**

**(2, '2023-03-17', 200),**

**(2, '2023-03-16', 150),**

**(2, '2023-03-15', 175),**

**(3, '2023-03-17', 300),**

**(3, '2023-03-16', 250),**

**(3, '2023-03-15', 275);**

**INSERT 8 :**

**INSERT INTO Transaksi (id\_transaksi, tanggal\_transaksi) VALUES**

**(1, '2022-01-01'),**

**(2, '2022-01-02'),**

**(3, '2022-01-03');**

**INSERT INTO Produk (id\_produk, nama\_produk, harga\_produk) VALUES**

**(1, 'Produk A', 1000),**

**(2, 'Produk B', 2000),**

**(3, 'Produk C', 3000);**

**INSERT INTO Detail\_Transaksi (id\_transaksi, id\_produk, jumlah\_produk) VALUES**

**(1, 1, 2),**

**(1, 2, 1),**

**(1, 3, 3),**

**(2, 1, 1),**

**(2, 2, 2),**

**(2, 3, 1),**

**(3, 1, 3),**

**(3, 2, 2),**

**(3, 3, 2);**

**SELECT :**

**SELECT nama\_pelanggan**

**FROM tabel\_pelanggan**

**WHERE gender = 'Laki-laki' OR gender = 'M';**

**SELECT \***

**FROM tabel\_product**

**WHERE id = 3;**

**SELECT \***

**FROM tabel\_pelanggan**

**WHERE created\_at BETWEEN DATE\_SUB(NOW(), INTERVAL 7 DAY) AND NOW() AND nama\_pelanggan LIKE '%a%';**

**SELECT COUNT(\*)**

**FROM tabel\_pelanggan**

**WHERE gender = 'Perempuan';**

**SELECT \***

**FROM tabel\_pelanggan**

**ORDER BY nama\_pelanggan ASC;**

**SELECT \***

**FROM tabel\_product**

**LIMIT 5;**

**UPDATE :**

**UPDATE products**

**SET product\_name = 'product dummy'**

**WHERE product\_id = 1;**

**UPDATE transaction\_detail**

**SET qty = 3**

**WHERE product\_id = 1;**

**DELETE :**

**DELETE FROM product WHERE id = 1;**

**DELETE FROM product WHERE product\_type\_id = 1;**

**Join - Union - Agregasi - Subquery - Function (DBMS)**

**SELECT \* FROM transactions**

**WHERE user\_id IN (1, 2);**

**SELECT SUM(total\_price) AS total\_harga FROM transactions**

**WHERE user\_id = 1;**

**SELECT COUNT(\*) AS total\_transaksi FROM transactions**

**JOIN transaction\_details ON transactions.id = transaction\_details.transaction\_id**

**JOIN products ON transaction\_details.product\_id = products.id**

**WHERE products.product\_type\_id = 2;**

**SELECT \* FROM products**

**JOIN product\_types ON products.product\_type\_id = product\_types.id;**

**SELECT \* FROM transactions**

**JOIN transaction\_details ON transactions.id = transaction\_details.transaction\_id**

**JOIN products ON transaction\_details.product\_id = products.id**

**JOIN users ON transactions.user\_id = users.id;**

**CREATE FUNCTION delete\_transaction(id INT)**

**RETURNS VOID AS**

**$$**

**BEGIN**

**DELETE FROM transaction\_details WHERE transaction\_id = id;**

**DELETE FROM transactions WHERE id = id;**

**END;**

**$$**

**LANGUAGE plpgsql;**

**CREATE FUNCTION update\_total\_qty(id INT)**

**RETURNS VOID AS**

**$$**

**BEGIN**

**UPDATE products**

**SET total\_qty = total\_qty - (**

**SELECT SUM(qty) FROM transaction\_details**

**WHERE transaction\_id = id AND product\_id = products.id**

**)**

**WHERE EXISTS (**

**SELECT \* FROM transaction\_details**

**WHERE transaction\_id = id AND product\_id = products.id**

**);**

**END;**

**$$**

**LANGUAGE plpgsql;**

**SELECT \* FROM products**

**WHERE id NOT IN (**

**SELECT DISTINCT product\_id FROM transaction\_details**

**);**