

## LAPORAN UJIAN AKHIR SEMESTER DATABASE

M.S ARDIANSYAH YUDHONO

ID_Transaksi	Tanggal	Nama_Pelanggan	Produk	Kuantitas	Harga
1	2024-12-25	Ardi	Paku,Kayu	10,20	100,200
2	2024-12-26	Siti	Semen	5	500

1NF

ID_Transaksi	Tanggal	Nama_Pelanggan	Produk	Kuantitas	Harga
1	2024-12-25	Ardi	Paku	10	100
2	2024-12-25	Ardi	Kayu	20	200
3	2024-12-26	Siti	Semen	5	500

2nf

Tbl Transaksi

ID_Transaksi	Tanggal	ID_Pelanggan
1	2024-12-25	101
2	2024-12-26	102

Tbl Pelanggan

ID_Pelanggan	Nama_Pelanggan
101	Ardi
102	Siti

Tbl Detail\_transaksi

ID_Transaksi	Produk	Kuantitas	Harga
1	Paku	10	100
1	Kayu	20	200
2	Semen	5	300

3NF

Tbl Produk

ID_Produk	Nama_Produk	Harga
201	Paku	100
202	Kayu	200
203	Semen	500

Tbl Detail\_Transaksi

ID_Transaksi	ID_Produk	Kuantitas
1	201	10
1	202	20
2	203	5

```

1 CREATE TABLE Produk (
2     id_produk INT PRIMARY KEY,
3     nama_produk VARCHAR(100),
4     kategori VARCHAR(50),
5     harga DECIMAL(10, 2),
6     stok INT
7 );

```

```

1 CREATE TABLE Pelanggan (
2     id_pelanggan INT PRIMARY KEY,
3     nama_pelanggan VARCHAR(100),
4     alamat VARCHAR(255),
5     email VARCHAR(100),
6     nomor_telepon VARCHAR(15)
7 );

```

```

1 CREATE TABLE Transaksi (
2     id_transaksi INT PRIMARY KEY,
3     tanggal DATE,
4     id_pelanggan INT,
5     FOREIGN KEY (id_pelanggan) REFERENCES Pelanggan(id_pelanggan)
6 );

```

```

1 CREATE TABLE Detail_Transaksi (
2     id_detail_transaksi INT PRIMARY KEY,
3     id_transaksi INT,
4     id_produk INT,
5     kuantitas INT,
6     subtotal DECIMAL(10, 2),
7     FOREIGN KEY (id_transaksi) REFERENCES Transaksi(id_transaksi),
8     FOREIGN KEY (id_produk) REFERENCES Produk(id_produk)
9 );

```

```

1 CREATE TABLE Pemasok (
2     id_pemasok INT PRIMARY KEY,
3     nama_pemasok VARCHAR(100),
4     alamat_pemasok VARCHAR(255),
5     nomor_telepon VARCHAR(15)
6 );

```

```

1 CREATE TABLE Pemasok (
2     id_pemasok INT PRIMARY KEY,
3     nama_pemasok VARCHAR(100),
4     alamat_pemasok VARCHAR(255),
5     nomor_telepon VARCHAR(15)
6 );

```

## SOAL NOMOR 2 JOIN DAN SUBQUERY

```

1 SELECT
2     Transaksi.id_transaksi,
3     Transaksi.tanggal,
4     Pelanggan.nama_pelanggan,
5     Produk.nama_produk,
6     Detail_Transaksi.kuantitas,
7     Detail_Transaksi.subtotal
8 FROM
9     Transaksi
10 JOIN
11     Pelanggan ON Transaksi.id_pelanggan = Pelanggan.id_pelanggan
12 JOIN
13     Detail_Transaksi ON Transaksi.id_transaksi = Detail_Transaksi.id_transaksi
14 JOIN
15     Produk ON Detail_Transaksi.id_produk = Produk.id_produk
16 ORDER BY
17     Transaksi.tanggal DESC;
18

```

Transaksi (0r x 6c)					
id_transaksi	tanggal	nama_pelanggan	nama_produk	kuantitas	subtotal

```

1 SELECT
2     nama_pelanggan,
3     total_belanja
4 FROM
5     (SELECT
6         Pelanggan.nama_pelanggan,
7         SUM(Detail_Transaksi.subtotal) AS total_belanja
8     FROM
9         Pelanggan
10    JOIN
11        Transaksi ON Pelanggan.id_pelanggan = Transaksi.id_pelanggan
12    JOIN
13        Detail_Transaksi ON Transaksi.id_transaksi = Detail_Transaksi.id_transaksi
14    GROUP BY
15        Pelanggan.id_pelanggan, Pelanggan.nama_pelanggan
16    ) AS Total_Pelanggan
17 WHERE
18     total_belanja > 1000000;
19

```

Result #1 (0r x 2c)

nama_pelanggan	total_belanja
----------------	---------------

### SOAL NOMOR 3 IMPLEMENTASI 3 JENIS DATABASE OBJECK

- VIEW

```

1 CREATE VIEW Laporan_Transaksi AS
2 SELECT
3     Transaksi.id_transaksi,
4     Transaksi.tanggal,
5     Pelanggan.nama_pelanggan,
6     Produk.nama_produk,
7     Detail_Transaksi.kuantitas,
8     Detail_Transaksi.subtotal
9 FROM
10    Transaksi
11 JOIN
12    Pelanggan ON Transaksi.id_pelanggan = Pelanggan.id_pelanggan
13 JOIN
14    Detail_Transaksi ON Transaksi.id_transaksi = Detail_Transaksi.id_transaksi
15 JOIN
16    Produk ON Detail_Transaksi.id_produk = Produk.id_produk;
17

```

Kasus : Membuat laporan transaksi yang menampilkan ID Transaksi, Tanggal Transaksi, Nama pelanggan, Nama produk, Kuantitas, dan Subtotal

- PROCEDURE

```
1 DELIMITER //
2 CREATE PROCEDURE Insert_Transaksi(
3     IN p_id_transaksi INT,
4     IN p_tanggal DATE,
5     IN p_id_pelanggan INT,
6     IN p_id_produk INT,
7     IN p_kuantitas INT
8 )
9 BEGIN
10     DECLARE v_harga DECIMAL(10, 2);
11     DECLARE v_subtotal DECIMAL(10, 2);
12
13     -- Ambil harga produk
14     SELECT harga INTO v_harga FROM Produk WHERE id_produk = p_id_produk;
15
16     -- Hitung subtotal
17     SET v_subtotal = v_harga * p_kuantitas;
18
19     -- Insert ke tabel Transaksi
20     INSERT INTO Transaksi (id_transaksi, tanggal, id_pelanggan)
21     VALUES (p_id_transaksi, p_tanggal, p_id_pelanggan);
22
23     -- Insert ke tabel Detail Transaksi
24     INSERT INTO Detail_Transaksi (id_detail_transaksi, id_transaksi, id_produk, kuantitas, subtotal)
25     VALUES (NULL, p_id_transaksi, p_id_produk, p_kuantitas, v_subtotal);
26 END //
27 DELIMITER ;
28
```

Kasus : Memasukan Data transaksi baru Bersama detailnya sekaligus, untuk menghindari kesalahan input.

- Function

```
1 DELIMITER //
2 CREATE FUNCTION Total_Pengeluaran_Pelanggan(p_id_pelanggan INT)
3 RETURNS DECIMAL(10, 2)
4 DETERMINISTIC
5 BEGIN
6     DECLARE total DECIMAL(10, 2);
7
8     SELECT SUM(subtotal) INTO total
9     FROM Transaksi
10    JOIN Detail_Transaksi ON Transaksi.id_transaksi = Detail_Transaksi.id_transaksi
11    WHERE Transaksi.id_pelanggan = p_id_pelanggan;
12
13     RETURN total;
14 END //
15 DELIMITER ;
16 |
```

Kasus : Menghitung total pengeluaran pelanggan berdasarkan id pelanggan