

BASIS DATA KELOMPOK 4

NAYLA PUTRI CAHYA RAMADANI - 2024071020

KEREN SANDRA SUDARTA – 2024071031

DDL (Data Definition Language)

```
-- Membuat database dan menggunakan database
CREATE DATABASE IF NOT EXISTS toko_kue;
USE toko_kue;

-- Tabel 1: Pelanggan
CREATE TABLE pelanggan (
    id_pelanggan INT PRIMARY KEY AUTO_INCREMENT,
    nama_pelanggan VARCHAR(100) NOT NULL,
    nomor_telepon VARCHAR(15) NOT NULL,
    email VARCHAR(100),
    tanggal_daftar DATETIME DEFAULT CURRENT_TIMESTAMP
);

-- Tabel 2: Alamat Pengiriman
CREATE TABLE alamat_pengiriman (
    id_alamat INT PRIMARY KEY AUTO_INCREMENT,
    id_pelanggan INT NOT NULL,
    alamat_lengkap TEXT NOT NULL,
    kecamatan VARCHAR(50),
    kota VARCHAR(50) NOT NULL,
    kode_pos VARCHAR(10),
    catatan_alamat TEXT,
    FOREIGN KEY (id_pelanggan) REFERENCES
    pelanggan(id_pelanggan)
);

-- Tabel 3: Produk Kue
CREATE TABLE produk_kue (
    id_produk INT PRIMARY KEY AUTO_INCREMENT,
    nama_kue VARCHAR(100) NOT NULL,
    kategori VARCHAR(50),
    harga DECIMAL(10,2) NOT NULL,
```

```

        deskripsi TEXT,
        stok INT DEFAULT 0,
        status_tersedia BOOLEAN DEFAULT TRUE
    );

-- Tabel 4: Pesanan
CREATE TABLE pesanan (
    id_pesanan INT PRIMARY KEY AUTO_INCREMENT,
    id_pelanggan INT NOT NULL,
    id_alamat INT NOT NULL,
    tanggal_pesan DATETIME DEFAULT CURRENT_TIMESTAMP,
    tanggal_pengiriman DATETIME,
    total_harga DECIMAL(10,2) NOT NULL,
    status_pesanan VARCHAR(20) DEFAULT 'Pending',
    metode_pembayaran VARCHAR(30),
    catatan_pesanan TEXT,
    FOREIGN KEY (id_pelanggan) REFERENCES
    pelanggan(id_pelanggan),
    FOREIGN KEY (id_alamat) REFERENCES
    alamat_pengiriman(id_alamat)
);

-- Tabel 5: Detail Pesanan
CREATE TABLE detail_pesanan (
    id_detail INT PRIMARY KEY AUTO_INCREMENT,
    id_pesanan INT NOT NULL,
    id_produk INT NOT NULL,
    jumlah INT NOT NULL,
    harga_satuan DECIMAL(10,2) NOT NULL,
    subtotal DECIMAL(10,2) NOT NULL,
    catatan_item TEXT,
    FOREIGN KEY (id_pesanan) REFERENCES pesanan(id_pesanan),
    FOREIGN KEY (id_produk) REFERENCES produk_kue(id_produk)
);

-- Membuat View
CREATE OR REPLACE VIEW view_pesanan_lengkap AS
SELECT p.id_pesanan, p.tanggal_pesan, p.tanggal_pengiriman,
       pel.nama_pelanggan, pel.nomor_telepon,
       a.alamat_lengkap, a.kota, p.total_harga,
       p.status_pesanan, p.metode_pembayaran
FROM pesanan p
JOIN pelanggan pel ON p.id_pelanggan = pel.id_pelanggan
JOIN alamat_pengiriman a ON p.id_alamat = a.id_alamat;

CREATE OR REPLACE VIEW view_detail_pesanan_lengkap AS

```

```
SELECT d.id_pesanan, k.nama_kue, k.kategori,  
       d.jumlah, d.harga_satuan, d.subtotal, d.catatan_item  
FROM detail_pesanan d  
JOIN produk_kue k ON d.id_produk = k.id_produk;
```

DML (Data Manipulation Language)

```
-- Menambahkan data pelanggan  
INSERT INTO pelanggan (nama_pelanggan, nomor_telepon, email)  
VALUES ('Budi Santoso', '081234567890', 'budi@email.com');  
  
-- Menambahkan produk  
INSERT INTO produk_kue (nama_kue, kategori, harga, deskripsi,  
stok)  
VALUES ('Kue Ulang Tahun Coklat', 'Birthday Cake', 250000, 'Kue  
coklat 3 lapis', 5);  
  
-- Mengupdate stok  
UPDATE produk_kue SET stok = 10 WHERE id_produk = 1;  
  
-- Menghapus pelanggan tertentu  
DELETE FROM pelanggan WHERE id_pelanggan = 5;
```

DQL (Data Query Language)

```
-- Menampilkan semua pelanggan  
SELECT * FROM pelanggan;  
  
-- Menampilkan semua pesanan  
SELECT * FROM view_pesanan_lengkap ORDER BY tanggal_pesan DESC;  
  
-- Menampilkan laporan penjualan per kategori  
SELECT k.kategori, COUNT(d.id_detail) AS total_item_terjual,  
       SUM(d.subtotal) AS total_pendapatan  
FROM detail_pesanan d  
JOIN produk_kue k ON d.id_produk = k.id_produk  
JOIN pesanan p ON d.id_pesanan = p.id_pesanan  
WHERE p.status_pesanan != 'Dibatalkan'  
GROUP BY k.kategori
```

```
ORDER BY total_pendapatan DESC;

-- Menampilkan stok yang hampir habis
SELECT nama_kue, kategori, stok FROM produk_kue WHERE stok <
10;
```

STORED PROCEDURE

| No. | Nama Procedure | Fungsi |
|-----|--------------------------|------------------------------------------------|
| 1. | sp_tambah_pelanggan | Menambah data pelanggan baru |
| 2. | sp_tambah_produk | Menambah produk baru |
| 3. | sp_tambah_alamat | Menambah alamat pengiriman baru |
| 4. | sp_buat_pesanan | Membuat pesanan baru (header) |
| 5. | sp_tambah_item_pesanan | Menambah item ke dalam pesanan |
| 6. | sp_update_status_pesanan | Mengubah status pesanan |
| 7. | sp_update_stok | Memperbarui stok produk |
| 8. | sp_laporan_haria | Membuat laporan penjualan harian |
| 9.. | sp_cek_stok_menipis | Menampilkan produk dengan stok rendah |
| 10. | sp_riwayat_pelanggan_v2 | Menampilkan riwayat pesanan pelanggan tertentu |