**database penjualan sederhana dengan tiga tabel** : `Produk`, `Pelanggan`, dan `Penjualan`.

1. \*\*Tabel Produk (`produk`)\*\*: Menyimpan data tentang produk yang dijual.

2. \*\*Tabel Pelanggan (`pelanggan`)\*\*: Menyimpan data pelanggan.

3. \*\*Tabel Penjualan (`penjualan`)\*\*: Menyimpan data transaksi penjualan.

### 1. Tabel Produk (`produk`)

| Nama Kolom | Tipe Data | Keterangan |

|--------------|-----------------|-----------------------------|

| id\_produk | INT | Primary Key, Auto Increment |

| nama\_produk | VARCHAR(50) | Nama produk |

| harga | DECIMAL(10, 2) | Harga produk per unit |

| stok | INT | Jumlah stok produk |

### 2. Tabel Pelanggan (`pelanggan`)

| Nama Kolom | Tipe Data | Keterangan |

|--------------|-----------------|-----------------------------|

| id\_pelanggan | INT | Primary Key, Auto Increment |

| nama | VARCHAR(50) | Nama pelanggan |

| alamat | VARCHAR(100) | Alamat pelanggan |

| telepon | VARCHAR(15) | Nomor telepon pelanggan |

### 3. Tabel Penjualan (`penjualan`)

| Nama Kolom | Tipe Data | Keterangan |

|--------------|-----------------|-------------------------------------|

| id\_penjualan | INT | Primary Key, Auto Increment |

| id\_pelanggan | INT | Foreign Key ke tabel `pelanggan` |

| id\_produk | INT | Foreign Key ke tabel `produk` |

| jumlah | INT | Jumlah produk yang dibeli |

| tanggal | DATE | Tanggal penjualan |

| total\_harga | DECIMAL(10, 2) | Total harga dari produk yang dibeli |

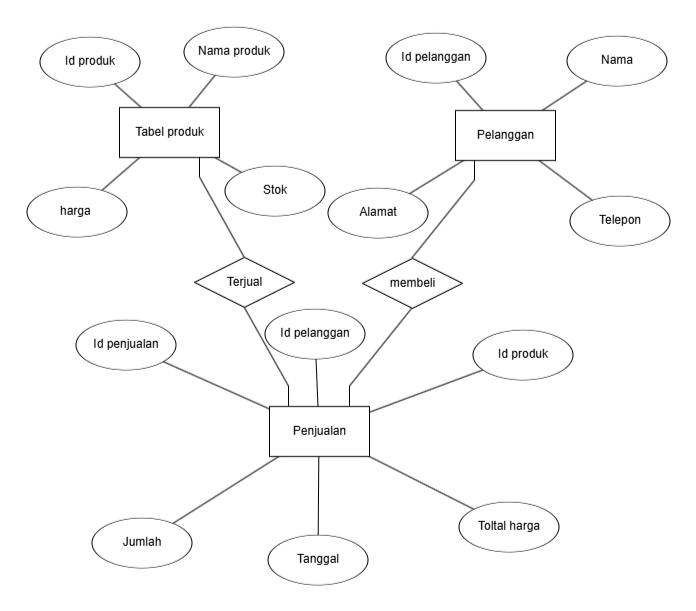
### Penjelasan

1. \*\*Tabel Produk\*\*: Menyimpan data produk, harga, dan stok.

2. \*\*Tabel Pelanggan\*\*: Menyimpan data pelanggan seperti nama, alamat, dan telepon.

3. \*\*Tabel Penjualan\*\*: Menyimpan transaksi penjualan, yang menghubungkan produk dengan pelanggan beserta informasi transaksi.

ERD Penjualan



Penjelasan :

1. **Entitas Produk (Tabel Produk)**:
   * **Atribut**: id\_produk, nama\_produk, harga, dan stok.
   * **Hubungan**: Tabel produk memiliki hubungan "terjual" dengan tabel penjualan, yang berarti produk dapat terlibat dalam banyak transaksi penjualan.
2. **Entitas Pelanggan**:
   * **Atribut**: id\_pelanggan, nama, alamat, dan telepon.
   * **Hubungan**: Pelanggan memiliki hubungan "membeli" dengan tabel penjualan, yang berarti satu pelanggan dapat melakukan banyak transaksi penjualan.
3. **Entitas Penjualan**:
   * **Atribut**: id\_penjualan, id\_pelanggan, id\_produk, jumlah, tanggal, dan total\_harga.
   * **Hubungan**:
     + **Ke Pelanggan**: Menggunakan atribut id\_pelanggan sebagai foreign key, menghubungkan penjualan dengan entitas pelanggan untuk menunjukkan siapa yang membeli produk.
     + **Ke Produk**: Menggunakan atribut id\_produk sebagai foreign key, menghubungkan penjualan dengan entitas produk untuk menunjukkan produk mana yang terjual.
     + Mencatat detail penjualan seperti jumlah produk yang dibeli, tanggal transaksi, dan total harga dari transaksi tersebut.

**Penjelasan Hubungan**

* **Pelanggan - Penjualan**: Hubungan "membeli" antara pelanggan dan penjualan menunjukkan bahwa satu pelanggan dapat melakukan banyak transaksi penjualan.
* **Produk - Penjualan**: Hubungan "terjual" antara produk dan penjualan menunjukkan bahwa satu produk dapat dijual dalam berbagai transaksi.

DATABASE PERPAJAKAN

Untuk membuat database tentang perpajakan, kita perlu merancang tabel-tabel yang sesuai dengan informasi yang akan dikelola. Berikut adalah struktur dasar untuk database perpajakan. Ini bisa dikembangkan lebih lanjut tergantung pada kebutuhan, seperti mencatat pembayaran pajak, jenis pajak, detail wajib pajak, dan sebagainya.

**Struktur Database Perpajakan**

**1. Tabel Wajib\_Pajak**  
Menyimpan informasi tentang individu atau perusahaan yang terdaftar sebagai wajib pajak.

| **Field** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| wp\_id | INT (Primary Key) | ID Wajib Pajak |
| nama | VARCHAR(100) | Nama wajib pajak |
| npwp | VARCHAR(20) | Nomor NPWP |
| alamat | TEXT | Alamat wajib pajak |
| nomor\_telepon | VARCHAR(15) | Nomor telepon wajib pajak |
| email | VARCHAR(100) | Email wajib pajak |
| jenis\_wp | ENUM('individu', 'perusahaan') | Jenis wajib pajak |

**2. Tabel Jenis\_Pajak**  
Menyimpan informasi tentang berbagai jenis pajak yang tersedia.

| **Field** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| jenis\_pajak\_id | INT (Primary Key) | ID Jenis Pajak |
| nama\_pajak | VARCHAR(100) | Nama pajak |
| deskripsi | TEXT | Deskripsi pajak |
| tarif | DECIMAL(5,2) | Tarif pajak dalam persentase |

**3. Tabel Transaksi\_Pajak**  
Menyimpan informasi tentang transaksi pembayaran pajak oleh wajib pajak.

| **Field** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| transaksi\_id | INT (Primary Key) | ID Transaksi Pajak |
| wp\_id | INT | ID Wajib Pajak (Foreign Key ke Wajib\_Pajak) |
| jenis\_pajak\_id | INT | ID Jenis Pajak (Foreign Key ke Jenis\_Pajak) |
| tanggal | DATE | Tanggal transaksi |
| jumlah\_pembayaran | DECIMAL(15,2) | Jumlah pembayaran pajak |
| status | ENUM('lunas', 'belum lunas') | Status pembayaran |

**4. Tabel Riwayat\_Pajak**  
Mencatat riwayat pajak tiap wajib pajak, berguna untuk melacak pelaporan dan pembayaran pajak.

| **Field** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| riwayat\_id | INT (Primary Key) | ID Riwayat Pajak |
| wp\_id | INT | ID Wajib Pajak (Foreign Key ke Wajib\_Pajak) |
| jenis\_pajak\_id | INT | ID Jenis Pajak (Foreign Key ke Jenis\_Pajak) |
| periode | DATE | Periode pajak (misal: 2023-12-01 untuk Desember 2023) |
| status\_lapor | ENUM('sudah', 'belum') | Status pelaporan |
| jumlah\_pajak | DECIMAL(15,2) | Jumlah pajak yang harus dibayar |

**5. Tabel Sanksi\_Pajak**  
Menyimpan informasi tentang sanksi pajak jika ada keterlambatan atau ketidakpatuhan.

| **Field** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| sanksi\_id | INT (Primary Key) | ID Sanksi Pajak |
| wp\_id | INT | ID Wajib Pajak (Foreign Key ke Wajib\_Pajak) |
| jenis\_pajak\_id | INT | ID Jenis Pajak (Foreign Key ke Jenis\_Pajak) |
| tanggal\_sanksi | DATE | Tanggal dikenakan sanksi |
| jumlah\_sanksi | DECIMAL(15,2) | Jumlah denda/sanksi |
| status\_sanksi | ENUM('dibayar', 'belum dibayar') | Status sanksi |

**Relasi Antar-Tabel**

* **Tabel Wajib\_Pajak** berelasi dengan **tabel Transaksi\_Pajak** dan **Riwayat\_Pajak** berdasarkan kolom wp\_id.
* **Tabel Jenis\_Pajak** berelasi dengan **tabel Transaksi\_Pajak**, **Riwayat\_Pajak**, dan **Sanksi\_Pajak** berdasarkan kolom jenis\_pajak\_id.

**Contoh Query**

1. **Menambah Wajib Pajak Baru**:
2. INSERT INTO Wajib\_Pajak (nama, npwp, alamat, nomor\_telepon, email, jenis\_wp)
3. VALUES ('John Doe', '123456789', 'Jl. Merdeka No.10', '08123456789', 'johndoe@email.com', 'individu');
4. **Menambah Jenis Pajak Baru**:
5. INSERT INTO Jenis\_Pajak (nama\_pajak, deskripsi, tarif)
6. VALUES ('PPN', 'Pajak Pertambahan Nilai', 10.00);
7. **Mencatat Pembayaran Pajak**:
8. INSERT INTO Transaksi\_Pajak (wp\_id, jenis\_pajak\_id, tanggal, jumlah\_pembayaran, status)
9. VALUES (1, 1, '2023-10-01', 500000.00, 'lunas');

Ini adalah struktur dasar yang bisa dikembangkan lebih jauh sesuai kebutuhan sistem perpajakan yang lebih kompleks.

