**Penjelasan entitas**

**1. PRODUCT**

Entitas PRODUCT mewakili produk atau barang yang dijual dalam sistem e-commerce atau retail.

Atribut:

* product\_id: Primary key yang unik untuk setiap produk.
* warna: Warna dari produk.
* ukuran: Ukuran dari produk.
* model: Model dari produk.
* harga: Harga dari produk.
* kategori: Kategori dari produk.

**2. SALE\_ITEM**

Entitas SALE\_ITEM mewakili item individu dalam sebuah transaksi penjualan.

Atribut:

* sale\_item\_id: Primary key yang unik untuk setiap item penjualan.
* product\_id: Kunci asing yang menghubungkan ke entitas PRODUCT.
* sale\_id: Kunci asing yang menghubungkan ke entitas SALE.
* total\_pembayaran: Total pembayaran untuk item penjualan tersebut.
* jumlah: Jumlah produk yang terjual dalam item penjualan tersebut.

**3. SALE**

Entitas SALE mewakili transaksi penjualan secara keseluruhan.

Atribut:

* sale\_id: Primary key yang unik untuk setiap penjualan.
* customer\_id: Kunci asing yang menghubungkan ke entitas CUSTOMER.
* tanggal\_pembayaran: Tanggal pembayaran dilakukan.
* metode\_pembayaran: Metode pembayaran yang digunakan.
* total\_pembayaran: Total pembayaran untuk seluruh transaksi penjualan.

**4. CUSTOMER**

Entitas CUSTOMER mewakili pelanggan yang melakukan pembelian dalam sistem.

Atribut:

* customer\_id: Primary key yang unik untuk setiap pelanggan.
* nama: Nama pelanggan.
* no\_hp: Nomor telepon pelanggan.
* email: Alamat email pelanggan.
* jenis\_kelamin: Jenis kelamin pelanggan.

**5. STOCK**

Entitas STOCK mewakili stok atau persediaan produk yang tersedia.

Atribut:

* stock\_id: Primary key yang unik untuk setiap entri stok.
* product\_id: Kunci asing yang menghubungkan ke entitas PRODUCT.
* jumlah\_stock: Jumlah stok yang tersedia.
* tanggal\_masuk: Tanggal stok masuk ke dalam persediaan.

**6. PREFERENCE**

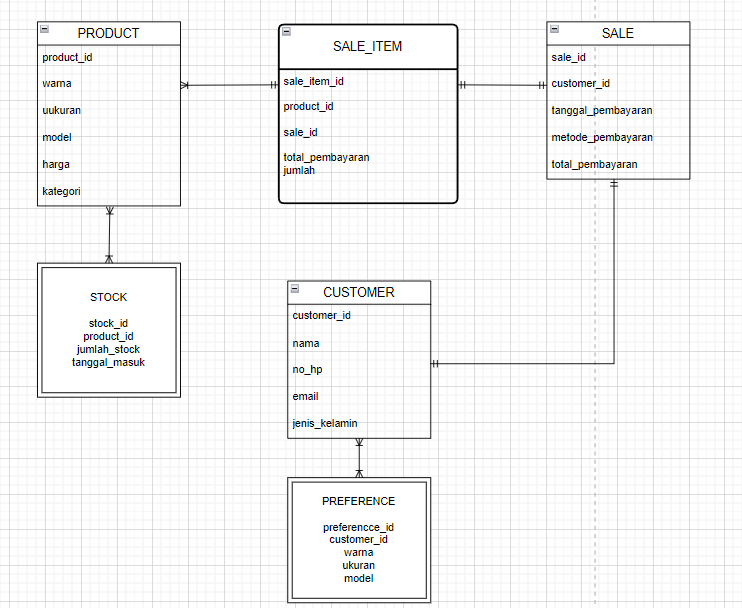
Entitas PREFERENCE mewakili preferensi atau kesukaan pelanggan terhadap produk tertentu.

Atribut:

* preference\_id: Primary key yang unik untuk setiap preferensi.
* customer\_id: Kunci asing yang menghubungkan ke entitas CUSTOMER.
* warna: Warna yang disukai oleh pelanggan.
* ukuran: Ukuran yang disukai oleh pelanggan.
* model: Model yang disukai oleh pelanggan.

**ERD**

Berikut ini adalah gambar ERD yang kami buat:



**Notasi crowfoot yang digunakan:**

- Many to many



- Many to one



- One to one



- One to many



**Penjelasan hubungan antar Entitas:**

**1. PRODUCT**

Hubungan:

* Satu produk dapat muncul di banyak item penjualan (hubungan one-to-many dengan SALE\_ITEM).
* Satu produk dapat memiliki satu atau lebih entri stok (hubungan one-to-many dengan STOCK).
* Satu produk dapat muncul di banyak preferensi pelanggan (hubungan one-to-many dengan PREFERENCE).

**2. SALE\_ITEM**

Hubungan:

* Banyak item penjualan dapat terkait dengan satu produk (hubungan many-to-one dengan PRODUCT).
* Banyak item penjualan dapat terkait dengan satu penjualan (hubungan many-to-one dengan SALE).

**3. SALE**

Hubungan:

* Satu penjualan dapat memiliki banyak item penjualan (hubungan one-to-many dengan SALE\_ITEM).
* Satu pelanggan dapat melakukan banyak penjualan (hubungan one-to-many dengan CUSTOMER).

**4. CUSTOMER**

Hubungan:

* Satu pelanggan dapat melakukan banyak penjualan (hubungan one-to-many dengan SALE).
* Satu pelanggan dapat memiliki banyak preferensi (hubungan one-to-many dengan PREFERENCE).

**5. STOCK**

Hubungan:

* Banyak entri stok dapat terkait dengan satu produk (hubungan many-to-one dengan PRODUCT).

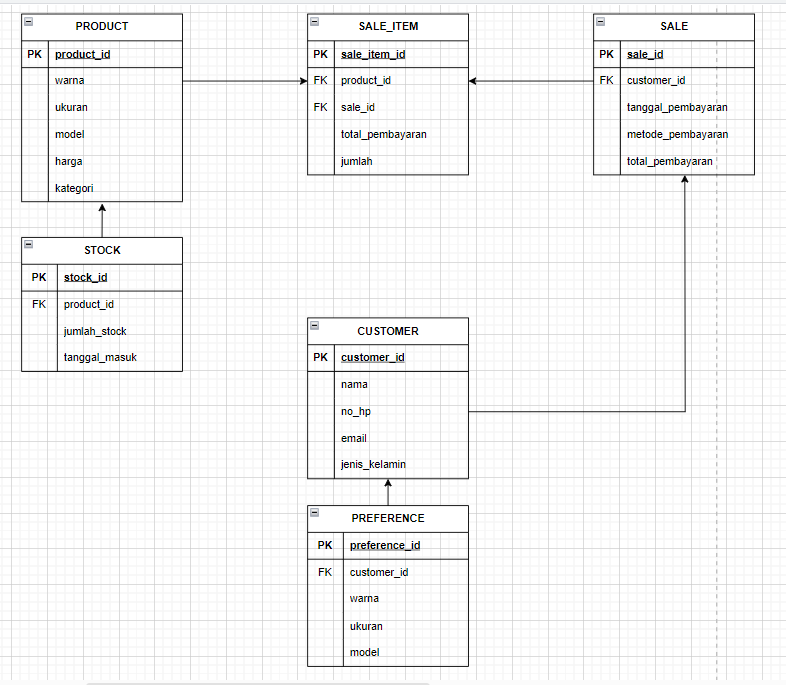
**6. PREFERENCE**

Hubungan:

* Banyak preferensi dapat terkait dengan satu pelanggan (hubungan many-to-one dengan CUSTOMER).
* Banyak preferensi dapat terkait dengan satu produk (hubungan many-to-one dengan PRODUCT).

**2.3 Skema relasi**

Berikut adalah hasil skema relasi berdsasarkan ERD yang sudah dibuat:



**Pembuatan tabel pada DBMS (phpMyAdmin)**

**1. Tabel product**

Berikut adalah query untuk membuat tabel product:

CREATE TABLE PRODUCT (

product\_id VARCHAR(100) PRIMARY KEY,

warna VARCHAR(50),

ukuran VARCHAR(50),

model VARCHAR(50),

harga INT,

kategori VARCHAR(50)

);

Penjelasan:

Tabel PRODUCT menyimpan informasi tentang produk yang dijual oleh bisnis.

Atribut:

* product\_id: Kunci utama yang unik untuk setiap produk. Tipe data VARCHAR dengan panjang 100 karakter.
* warna: Warna dari produk. Tipe data VARCHAR dengan panjang 50 karakter.
* ukuran: Ukuran atau dimensi dari produk. Tipe data VARCHAR dengan panjang 50 karakter.
* model: Model atau gaya dari produk. Tipe data VARCHAR dengan panjang 50 karakter.
* harga: Harga dari produk. Tipe data INT.
* kategori: Kategori atau jenis dari produk. Tipe data VARCHAR dengan panjang 50 karakter.

**2. Tabel stock**

Berikut adalah query untuk membuat tabel stock:

CREATE TABLE STOCK (

stock\_id VARCHAR(100) PRIMARY KEY,

product\_id VARCHAR(100),

jumlah\_stock INT,

tanggal\_masuk DATE,

FOREIGN KEY (product\_id) REFERENCES PRODUCT(product\_id)

);

Penjelasan:

Tabel STOCK menyimpan informasi tentang stok atau persediaan produk yang tersedia.

Atribut:

* stock\_id: Kunci utama yang unik untuk setiap entri stok. Tipe data VARCHAR dengan panjang 100 karakter.
* product\_id: Kunci asing yang menghubungkan stok ke produk yang sesuai. Tipe data VARCHAR dengan panjang 100 karakter.
* jumlah\_stock: Jumlah stok yang tersedia. Tipe data INT.
* tanggal\_masuk: Tanggal ketika produk ditambahkan ke stok. Tipe data DATE.

Hubungan:

* product\_id adalah kunci asing yang mengacu pada product\_id di tabel PRODUCT.

**3. Tabel customer**

Berikut adalah query untuk membuat tabel customer:

CREATE TABLE CUSTOMER (

customer\_id VARCHAR(100) PRIMARY KEY,

nama VARCHAR(100),

no\_hp VARCHAR(15),

email VARCHAR(100),

jenis\_kelamin VARCHAR(10)

);

Penjelasan:

Tabel CUSTOMER menyimpan informasi tentang pelanggan yang melakukan pembelian.

Atribut:

* customer\_id: Kunci utama yang unik untuk setiap pelanggan. Tipe data VARCHAR dengan panjang 100 karakter.
* nama: Nama pelanggan. Tipe data VARCHAR dengan panjang 100 karakter.
* no\_hp: Nomor telepon pelanggan. Tipe data VARCHAR dengan panjang 15 karakter.
* email: Alamat email pelanggan. Tipe data VARCHAR dengan panjang 100 karakter.
* jenis\_kelamin: Jenis kelamin pelanggan. Tipe data VARCHAR dengan panjang 10 karakter.

**4. Tabel sale**

Berikut adalah query untuk membuat tabel stock:

CREATE TABLE SALE (

sale\_id VARCHAR(100) PRIMARY KEY,

customer\_id VARCHAR(100),

tanggal\_pembayaran DATE,

metode\_pembayaran VARCHAR(50),

total\_pembayaran INT,

FOREIGN KEY (customer\_id) REFERENCES CUSTOMER(customer\_id)

);

Penjelasan:

Tabel SALE menyimpan informasi tentang transaksi penjualan secara keseluruhan.

Atribut:

* sale\_id: Kunci utama yang unik untuk setiap penjualan. Tipe data VARCHAR dengan panjang 100 karakter.
* customer\_id: Kunci asing yang menghubungkan penjualan ke pelanggan yang sesuai. Tipe data VARCHAR dengan panjang 100 karakter.
* tanggal\_pembayaran: Tanggal pembayaran dilakukan. Tipe data DATE.
* metode\_pembayaran: Metode pembayaran yang digunakan. Tipe data VARCHAR dengan panjang 50 karakter.
* total\_pembayaran: Total pembayaran untuk seluruh transaksi penjualan. Tipe data INT.

Hubungan:

* customer\_id adalah kunci asing yang mengacu pada customer\_id di tabel CUSTOMER.

**5. Tabel sale\_item**

Berikut adalah query untuk membuat tabel stock:

CREATE TABLE SALE\_ITEM (

sale\_item\_id VARCHAR(100) PRIMARY KEY,

product\_id VARCHAR(100),

sale\_id VARCHAR(100),

total\_pembayaran INT,

jumlah INT,

FOREIGN KEY (product\_id) REFERENCES PRODUCT(product\_id),

FOREIGN KEY (sale\_id) REFERENCES SALE(sale\_id)

);

Penjelasan:

Tabel SALE\_ITEM menyimpan detail dari setiap item yang terjual dalam sebuah transaksi penjualan.

Atribut:

* sale\_item\_id: Kunci utama yang unik untuk setiap item penjualan. Tipe data VARCHAR dengan panjang 100 karakter.
* product\_id: Kunci asing yang menghubungkan item penjualan ke produk yang sesuai. Tipe data VARCHAR dengan panjang 100 karakter.
* sale\_id: Kunci asing yang menghubungkan item penjualan ke penjualan yang sesuai. Tipe data VARCHAR dengan panjang 100 karakter.
* total\_pembayaran: Total pembayaran untuk item penjualan tersebut. Tipe data INT.
* jumlah: Jumlah produk yang terjual dalam item penjualan tersebut. Tipe data INT.

Hubungan:

* product\_id adalah kunci asing yang mengacu pada product\_id di tabel PRODUCT.
* sale\_id adalah kunci asing yang mengacu pada sale\_id di tabel SALE.

**6. Tabel preference**

Berikut adalah query untuk membuat tabel stock:

CREATE TABLE PREFERENCE (

preference\_id VARCHAR(100) PRIMARY KEY,

customer\_id VARCHAR(100),

warna VARCHAR(50),

ukuran VARCHAR(50),

model VARCHAR(50),

FOREIGN KEY (customer\_id) REFERENCES CUSTOMER(customer\_id)

);

Penjelasan:

Tabel PREFERENCE menyimpan preferensi atau kesukaan pelanggan terhadap produk tertentu.

Atribut:

* preference\_id: Kunci utama yang unik untuk setiap preferensi. Tipe data VARCHAR dengan panjang 100 karakter.
* customer\_id: Kunci asing yang menghubungkan preferensi ke pelanggan yang sesuai. Tipe data VARCHAR dengan panjang 100 karakter.
* warna: Warna yang disukai oleh pelanggan. Tipe data VARCHAR dengan panjang 50 karakter.
* ukuran: Ukuran yang disukai oleh pelanggan. Tipe data VARCHAR dengan panjang 50 karakter.
* model: Model yang disukai oleh pelanggan. Tipe data VARCHAR dengan panjang 50 karakter.

Hubungan:

* customer\_id adalah kunci asing yang mengacu pada customer\_id di tabel CUSTOMER.

**Insert data pada tabel di DBMS**

**1. Tabel product**

Berikut adalah query untuk insert data pada tabel product:

INSERT INTO PRODUCT (product\_id, warna, ukuran, model, harga, kategori) VALUES

('P001', 'Merah', '42', 'Sneakers', 500000, 'Sepatu Olahraga'),

('P002', 'Biru', '40', 'Boots', 750000, 'Sepatu Kasual'),

('P003', 'Hitam', '41', 'Loafers', 600000, 'Sepatu Formal'),

('P004', 'Putih', '39', 'Sneakers', 550000, 'Sepatu Olahraga'),

('P005', 'Coklat', '43', 'Boots', 800000, 'Sepatu Kasual'),

('P006', 'Abu-abu', '42', 'Loafers', 650000, 'Sepatu Formal'),

('P007', 'Hijau', '40', 'Sneakers', 520000, 'Sepatu Olahraga'),

('P008', 'Kuning', '41', 'Boots', 770000, 'Sepatu Kasual'),

('P009', 'Merah', '39', 'Loafers', 620000, 'Sepatu Formal'),

('P010', 'Biru', '43', 'Sneakers', 530000, 'Sepatu Olahraga');

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM product

**2. Tabel stock**

Berikut adalah query untuk insert data pada tabel stock:

INSERT INTO STOCK (stock\_id, product\_id, jumlah\_stock, tanggal\_masuk) VALUES

('S001', 'P001', 100, '2024-01-01'),

('S002', 'P002', 50, '2024-01-02'),

('S003', 'P003', 75, '2024-01-03'),

('S004', 'P004', 120, '2024-01-04'),

('S005', 'P005', 60, '2024-01-05'),

('S006', 'P006', 80, '2024-01-06'),

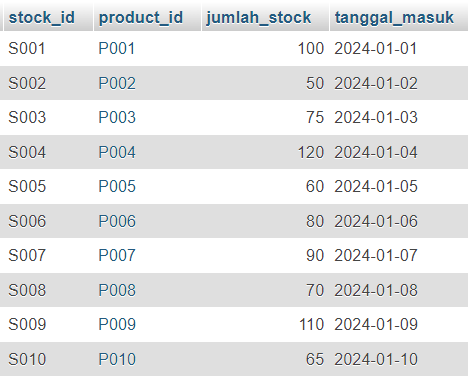
('S007', 'P007', 90, '2024-01-07'),

('S008', 'P008', 70, '2024-01-08'),

('S009', 'P009', 110, '2024-01-09'),

('S010', 'P010', 65, '2024-01-10');

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM stock

**3. Tabel customer**

Berikut adalah query untuk insert data pada tabel customer:

INSERT INTO CUSTOMER (customer\_id, nama, no\_hp, email, jenis\_kelamin) VALUES

('C001', 'Andhika', '081234567890', 'Andhika@example.com', 'Laki-laki'),

('C002', 'Rizqi', '081234567891', 'Rizqi@example.com', 'Perempuan'),

('C003', 'Raehan', '081234567892', 'Raehan@example.com', 'Perempuan'),

('C004', 'Rifqi', '081234567893', 'Rifqi@example.com', 'Laki-laki'),

('C005', 'Yobel', '081234567894', 'Yobel@example.com', 'Laki-laki'),

('C006', 'Alvino', '081234567895', 'Alvino@example.com', 'Perempuan'),

('C007', 'Kastanja', '081234567896', 'Kastanja@example.com', 'Perempuan'),

('C008', 'Emir', '081234567897', 'Emir@example.com', 'Laki-laki'),

('C009', 'Martin', '081234567898', 'Martin@example.com', 'Perempuan'),

('C010', 'Sabasti', '081234567899', 'Sabasti@example.com', 'Laki-laki');

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM customer

**4. Tabel sale**

Berikut adalah query untuk insert data pada tabel sale:

INSERT INTO SALE (sale\_id, customer\_id, tanggal\_pembayaran, metode\_pembayaran, total\_pembayaran) VALUES

('SA001', 'C001', '2024-02-01', 'Kartu Kredit', 1500000),

('SA002', 'C002', '2024-02-02', 'Tunai', 2000000),

('SA003', 'C003', '2024-02-03', 'Transfer Bank', 1750000),

('SA004', 'C004', '2024-02-04', 'Kartu Kredit', 1600000),

('SA005', 'C005', '2024-02-05', 'Tunai', 1800000),

('SA006', 'C006', '2024-02-06', 'Transfer Bank', 1900000),

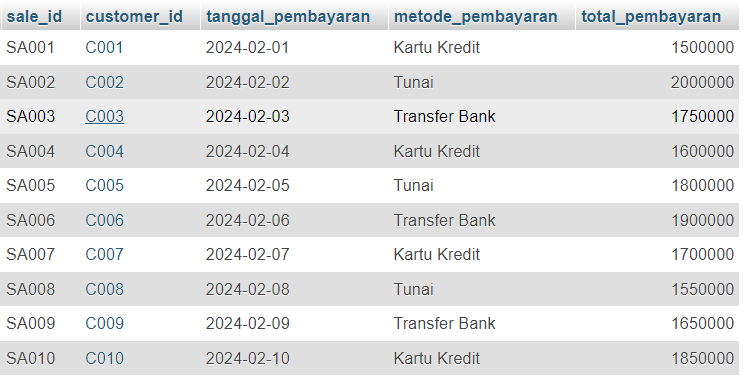
('SA007', 'C007', '2024-02-07', 'Kartu Kredit', 1700000),

('SA008', 'C008', '2024-02-08', 'Tunai', 1550000),

('SA009', 'C009', '2024-02-09', 'Transfer Bank', 1650000),

('SA010', 'C010', '2024-02-10', 'Kartu Kredit', 1850000);

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM sale

**5. Tabel sale\_item**

Berikut adalah query untuk insert data pada tabel sale\_item:

INSERT INTO SALE\_ITEM (sale\_item\_id, product\_id, sale\_id, total\_pembayaran, jumlah) VALUES

('SI001', 'P001', 'SA001', 500000, 1),

('SI002', 'P002', 'SA002', 750000, 1),

('SI003', 'P003', 'SA003', 600000, 1),

('SI004', 'P004', 'SA004', 550000, 1),

('SI005', 'P005', 'SA005', 800000, 1),

('SI006', 'P006', 'SA006', 650000, 1),

('SI007', 'P007', 'SA007', 520000, 1),

('SI008', 'P008', 'SA008', 770000, 1),

('SI009', 'P009', 'SA009', 620000, 1),

('SI010', 'P010', 'SA010', 530000, 1);

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM sale\_item

**6. Tabel preference**

Berikut adalah query untuk insert data pada tabel preference:

INSERT INTO PREFERENCE (preference\_id, customer\_id, warna, ukuran, model) VALUES

('PR001', 'C001', 'Merah', '42', 'Sneakers'),

('PR002', 'C002', 'Biru', '40', 'Boots'),

('PR003', 'C003', 'Hitam', '41', 'Loafers'),

('PR004', 'C004', 'Putih', '39', 'Sneakers'),

('PR005', 'C005', 'Coklat', '43', 'Boots'),

('PR006', 'C006', 'Abu-abu', '42', 'Loafers'),

('PR007', 'C007', 'Hijau', '40', 'Sneakers'),

('PR008', 'C008', 'Kuning', '41', 'Boots'),

('PR009', 'C009', 'Merah', '39', 'Loafers'),

('PR010', 'C010', 'Biru', '43', 'Sneakers');

Hasil:

****

Query untuk menampilkan tabel:

SELECT \* FROM preference