

## **LAPORAN PROGRES TUGAS BESAR BASIS DATA**

**“VTwoLaundry”**

*Laporan ini disusun untuk memenuhi Tugas Besar mata kuliah Basis Data*

**Dosen Pengampu : Adi Wahyu Pribadi, S.Si., M.Kom.**



Disusun Oleh :

Kelompok 10

Selsa Trikartika 4523210102

Chaerul Cahyadi 4523210120

Avryzel Alifian Hakim 4523210121

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

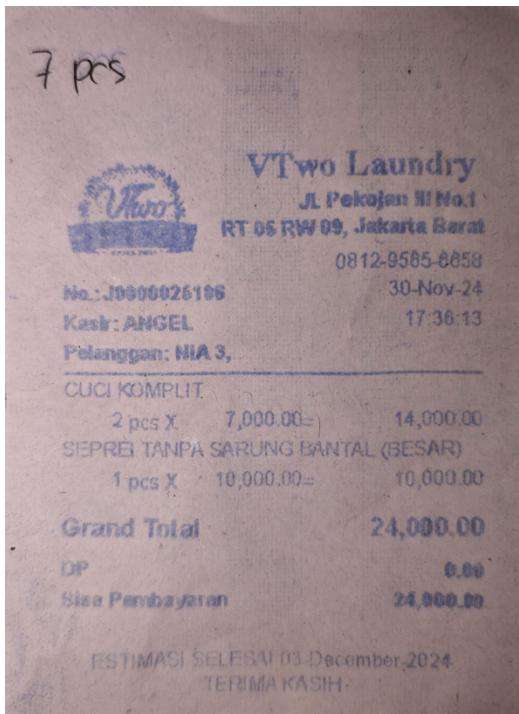
**UNIVERSITAS PANCASILA**

**2024/2025**

## 1. Latar Belakang

VTwoLaundry adalah sebuah usaha laundry yang berlokasi di Jl. Pekojan III No.1, RT.10/RW.1, Pekojan, Kec. Tambora, Kota Jakarta Barat, Daerah Khusus Ibukota Jakarta 14450. VTwoLaundry menyediakan berbagai layanan laundry berkualitas, seperti cuci kering, setrika, dry cleaning, dan layanan lainnya untuk memenuhi kebutuhan sehari-hari masyarakat yang memang memiliki banyak kegiatan dan tidak memiliki waktu untuk membersihkan pakaian maupun seprai yang biasanya digunakan untuk tidur. Dengan ini kelompok kami mencoba memberanikan diri untuk menanyakan hal apa saja yang biasa dilakukan VTwoLaundry dan mengubahnya menjadi basis data.

## 2. Struk Transaksi Laundry



## 3. Rancangan Database

### a. Entitas Utama

#### i. Tabel Master

- Ms\_Pelanggan:** Menyimpan informasi pelanggan dengan atribut utama seperti ID pelanggan dan nama pelanggan.
- Ms\_Kasir:** Mencatat data kasir yang melakukan transaksi, termasuk ID kasir dan nama kasir.
- Ms\_Toko:** Mendokumentasikan rincian toko laundry, seperti ID toko, nama toko, alamat, dan nomor telepon.
- Ms\_Parfum:** Mencatat jenis-jenis parfum yang tersedia, termasuk ID parfum, nama parfum, dan jumlah stok yang ada.

5. **Ms\_Layanan:** Berisi daftar layanan yang ditawarkan, dengan informasi mengenai nama layanan, satuan, dan harga.

## ii. Tabel Relasi

1. **Trx\_Laundry:** Mencatat informasi penting terkait transaksi laundry, termasuk nomor struk, nama pelanggan, kasir, nama toko, jenis parfum yang digunakan, total biaya, uang muka, sisa pembayaran, serta tanggal transaksi.
2. **Trx\_Layanan:** Mencatat detail layanan yang digunakan dalam suatu transaksi laundry, yang mencakup nomor struk yang relevan, jenis layanan yang diberikan, jumlah layanan, serta total biaya dari layanan tersebut.

## b. Penjelasan Atribut

### i. Ms\_Pelanggan

1. **idPelanggan:** Primary key untuk mengidentifikasi kasir dengan tipe data char(6).
2. **nama\_pelanggan:** Nama kasir yang bertugas dengan tipe data varchar.

### ii. Ms\_Kasir

1. **idKasir:** Primary key untuk mengidentifikasi toko dengan tipe data char(6).
2. **nama\_kasir:** Nama kasir yang bertugas dengan tipe data varchar.

### iii. Ms\_Toko

1. **idToko:** Primary key untuk mengidentifikasi toko dengan tipe data char(6).
2. **nama\_toko:** Nama toko laundry dengan tipe data varchar.
3. **alamat:** Alamat lengkap toko dengan tipe data varchar.
4. **no\_tlp:** Nomor telepon toko untuk keperluan kontak dengan tipe data bigint.

### iv. Ms\_Parfum

1. **idParfum:** Primary key untuk mengidentifikasi parfum dengan tipe data char(6).
2. **nama\_parfum:** Nama parfum yang tersedia dengan tipe data varchar.
3. **stok\_tersedia:** Jumlah stok parfum yang dapat digunakan dengan tipe data int.

##### v. Ms\_Layanan

1. **idLayanan:** Primary key untuk mengidentifikasi layanan dengan tipe data char(6).
2. **nama\_layanan:** Nama layanan yang disediakan dengan tipe data varchar.
3. **satuan:** Satuan layanan dengan tipe data varchar.
4. **harga** Biaya layanan berdasarkan satuan dengan tipe data int.

##### vi. Trx\_Laundry

1. **no\_struk:** Primary key untuk mencatat transaksi laundry dengan tipe data char(6).
2. **idParfum:** Foreign key parfum dari table MS\_Parfum yang digunakan dalam transaksi dengan tipe data char(6).
3. **idPelanggan:** Foreign key parfum dari table MS\_Pelanggan yang melakukan transaksi dengan tipe data char(6).
4. **idKasir:** Foreign key kasir dari table MS\_Kasir sebagai relasi yang menyimpan ID kasir yang melayani transaksi dengan tipe data char(6).
5. **idToko:** Foreign key toko dari table MS\_Toko tempat menyimpan transaksi yang dilakukan dengan tipe data char(6).
6. **grand\_total:** Total biaya transaksi dengan tipe data int.
7. **dp:** Uang yang dibayarkan secara langsung oleh pelanggan dengan tipe data int.
8. **sisa:** Sisa uang dari hasil pembayaran dengan tipe data int.
9. **tgl\_transaksi:** Tanggal transaksi dilakukan dengan tipe data timestamp.

##### vii. Trx\_Layanan

1. **idTrxLayanan:** Primary key untuk mencatat layanan dalam transaksi dengan tipe data int.
2. **no\_struk:** Nomor struk transaksi dengan tipe data char(6).
3. **idLayanan:** Foreign key layanan dari table MS\_Layanan yang digunakan dengan tipe data char(6).
4. **kuantitas:** Jumlah layanan yang digunakan dengan tipe data int.
5. **total\_harga:** Total harga untuk layanan berdasarkan kuantitas dengan tipe data int.

**c. Penjelasan Relasi**

**i. Ms\_Pelanggan - Trx\_Laundry (One to Many)**

Ms\_Pelanggan ini berisi informasi pelanggan yang terdaftar, dan setiap pelanggan dapat melakukan banyak transaksi yang ada di Trx\_Laundry itu sendiri. Lalu karena itulah Ms\_Pelanggan (pelanggan) ini entitasnya adalah One atau satu, dan Trx\_Laundry (transaksi laundry) adalah entitas Many atau banyak.

**ii. Ms\_Kasir - Trx\_Laundry (One to Many)**

Ms\_Kasir ini berisi informasi tentang kasir yang bertugas, dan setiap kasir dapat melayani banyak transaksi yang ada di Trx\_Laundry itu sendiri. Lalu karena itulah Ms\_Kasir (kasir) ini entitasnya adalah One atau satu, dan Trx\_Laundry (transaksi laundry) adalah entitas Many atau banyak.

**iii. Ms\_Toko - Trx\_Laundry (One to Many)**

Ms\_Toko ini berisi informasi tentang toko laundry yang terdaftar, dan setiap toko dapat mencatat banyak transaksi yang ada di Trx\_Laundry itu sendiri. Lalu karena itulah Ms\_Toko (toko) ini entitasnya adalah One atau satu, dan Trx\_Laundry (transaksi laundry) adalah entitas Many atau banyak.

**iv. Ms\_Parfum - Trx\_Laundry (One to Many)**

Ms\_Parfum ini dapat digunakan dalam banyaknya transaksi yang ada di Trx\_Laundry itu sendiri, lalu karena itulah Ms\_Parfum (parfum) ini entitasnya adalah One atau satu, dan Trx\_Laundry (transaksi laundry) adalah entitas Many atau banyak.

**v. Ms\_Layanan - Trx\_Layanan (One to Many)**

Ms\_Layanan ini berisi informasi tentang jenis layanan yang tersedia, layanan tersebut dapat digunakan dalam banyak transaksi yang ada di Trx\_Layanan itu sendiri. Lalu karena itulah Ms\_Layanan (layanan) ini entitasnya adalah One atau satu, dan Trx\_Layanan (transaksi layanan) adalah entitas Many atau banyak.

**vi. Trx\_Laundry - Trx\_Layanan (One to Many)**

Trx\_Laundry ini berisi transaksi utama pelanggan, setiap transaksi laundry dapat mencakup banyak layanan yang ada di Trx\_Layanan itu sendiri, karena itulah Trx\_Laundry (transaksi laundry) ini entitasnya adalah One atau satu, dan Trx\_Layanan (transaksi layanan) adalah entitas Many atau banyak.

#### d. ERD

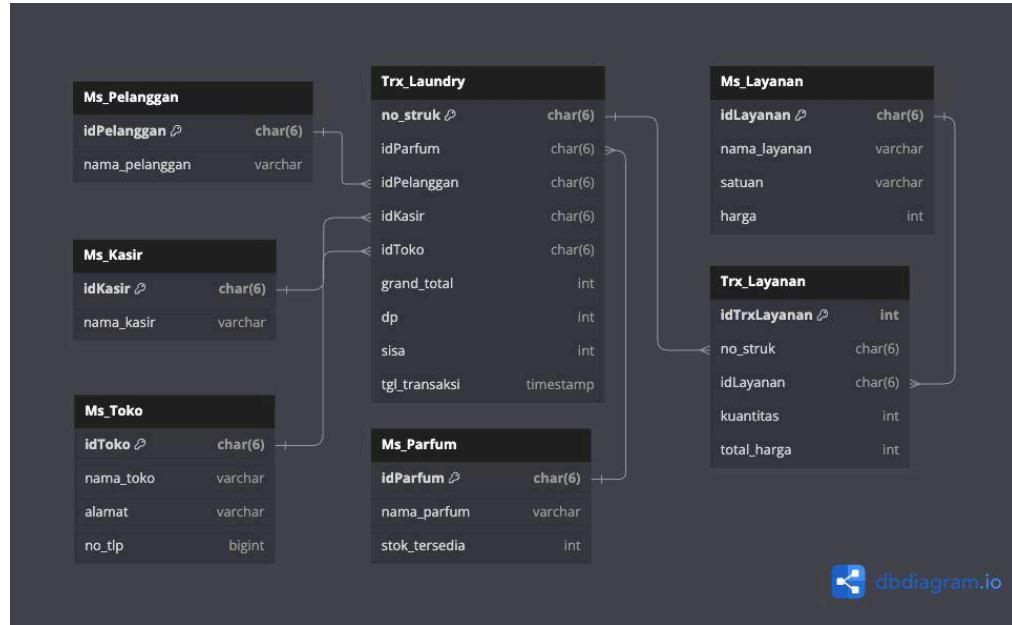


Diagram ERD ini menggambarkan sistem manajemen transaksi laundry yang terdiri dari beberapa tabel utama, yaitu Ms\_Pelanggan, Ms\_Kasir, Ms\_Toko, Ms\_Perfum, Ms\_Layanan, Trx\_Laundry, dan Trx\_Layanan.

Tabel-tabel tersebut terhubung satu sama lain melalui atribut kunci seperti idPelanggan, idKasir, idToko, idParfum, dan idLayanan, yang berfungsi sebagai kunci asing dalam tabel Trx\_Laundry atau Trx\_Layanan. Sistem ini memfasilitasi pencatatan data mengenai pelanggan, kasir, toko, parfum, layanan, serta rincian transaksi laundry, termasuk jumlah layanan, total harga, pembayaran uang muka, dan sisa pembayaran.

## 4. Implementasi Database

### a. Create Database

#### Query:

```
CREATE DATABASE VTwoLaundryDB CHARACTER SET utf8 COLLATE utf8_general_ci;
```

#### Output:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0093 seconds.)

```
CREATE DATABASE VTwoLaundryDB CHARACTER SET utf8 COLLATE utf8_general_ci;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

**Penjelasan:**

Query CREATE DATABASE berfungsi untuk menciptakan sebuah database baru yang dinamakan VTTwoLaundyDB.

Query CHARACTER SET utf8 menunjukkan bahwa database tersebut menggunakan pengkodean karakter UTF-8, sedangkan COLLATE utf8\_general\_ci menetapkan aturan perbandingan string yang tidak membedakan antara huruf kapital dan huruf kecil.

**b. Create Table Entitas atau Master****i. Create Table Ms\_Pelanggan****Query:**

```
CREATE TABLE Ms_Pelanggan ( idPelanggan CHAR(6) PRIMARY KEY, nama_pelanggan VARCHAR(255) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

**Output:**

```
✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0187 seconds.)  
CREATE TABLE Ms_Pelanggan ( idPelanggan CHAR(6) PRIMARY KEY, nama_pelanggan VARCHAR(255) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

**Penjelasan:**

Membuat tabel Ms\_Pelanggan, yang merupakan singkatan dari master pelanggan, bertujuan untuk menyimpan informasi mengenai pelanggan.

Kunci utama dari ID Pelanggan memiliki nilai yang unik dan tidak boleh kosong, dengan tipe data char(6).

Nama\_pelanggan digunakan untuk menyimpan nama pelanggan, dengan kapasitas 255 karakter, menggunakan tipe data varchar(255), dan ditetapkan sebagai not null, yang berarti nama\_pelanggan tidak boleh kosong dan harus diisi dengan nama pelanggan yang sesuai. Idlayanan memiliki tipe data yang sama dengan no\_struk, yaitu char(6), yang terdiri dari 6 karakter, dan idlayanan ini menyimpan identitas layanan yang diambil dari tabel Ms\_Layanan.

Foreign Key dari Ms\_Layanan(idLayanan) memastikan bahwa data layanan yang digunakan adalah valid. Kuantitas memiliki tipe data integer(int), yang dapat menyimpan jumlah layanan yang dipilih dalam transaksi serta mencatat banyaknya item atau unit layanan yang digunakan.

## ii. Create Table Ms\_Kasir

### Query:

```
CREATE TABLE Ms_Kasir ( idKasir CHAR(6) PRIMARY KEY, nama_kasir VARCHAR(255) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### Output:

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0180 seconds.)  
CREATE TABLE Ms_Kasir ( idKasir CHAR(6) PRIMARY KEY, nama_kasir VARCHAR(255) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
[Edit inline] [Edit] [Create PHP code]
```

### Penjelasan:

Tabel Ms\_Kasir dibuat sebagai representasi dari master kasir, yang berfungsi untuk menyimpan data utama (master data) terkait kasir.

Primary Key yang digunakan adalah idKasir, yang memiliki nilai unik dan tidak boleh kosong, dengan tipe data char(6).

Kolom nama\_kasir digunakan untuk menyimpan nama kasir, menggunakan tipe data varchar(255), dan ditandai dengan not null untuk menunjukkan bahwa kolom ini wajib diisi dan tidak boleh kosong.

## iii. Create Table Ms\_Layanan

### Query:

```
CREATE TABLE Ms_Layanan ( idLayanan CHAR(6) PRIMARY KEY, nama_layanan VARCHAR(255) NOT NULL, satuan VARCHAR(50) NOT NULL, harga INT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### Output:

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0119 seconds.)  
CREATE TABLE Ms_Layanan ( idLayanan CHAR(6) PRIMARY KEY, nama_layanan VARCHAR(255) NOT NULL, satuan VARCHAR(50) NOT NULL, harga INT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
[Edit inline] [Edit] [Create PHP code]
```

### Penjelasan:

Membuat tabel Ms\_Layanan, yang merupakan singkatan dari master layanan, menunjukkan bahwa tabel ini menyimpan data utama (master data) terkait layanan. Kunci utama dari idLayanan memiliki nilai yang unik dan tidak boleh kosong, dengan tipe data char(6).

Kolom nama\_layanan menyimpan nama layanan itu sendiri dengan tipe data varchar(255), di mana not null menandakan bahwa nama layanan tidak boleh kosong. Kolom satuan menyimpan satuan dari layanan dengan tipe data varchar(50), juga dengan not null yang menunjukkan bahwa satuan tidak boleh kosong.

Kolom harga menyimpan biaya layanan berdasarkan satuan dengan tipe data int, di mana not null menandakan bahwa harga tidak boleh kosong.

#### iv. Create Table Ms\_Toko

##### Query:

```
CREATE TABLE Ms_Toko ( idToko CHAR(6) PRIMARY KEY, nama_toko VARCHAR(255) NOT NULL, alamat VARCHAR(255) NOT NULL, no_tlp BIGINT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

##### Output:

The screenshot shows a MySQL command-line interface. A green status bar at the top indicates: "MySQL returned an empty result set (i.e. zero rows). (Query took 0.0126 seconds.)". Below it, the SQL query is displayed in a grey text area: "CREATE TABLE Ms\_Toko ( idToko CHAR(6) PRIMARY KEY, nama\_toko VARCHAR(255) NOT NULL, alamat VARCHAR(255) NOT NULL, no\_tlp BIGINT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;". At the bottom of the interface, there are three buttons: "[Edit inline]", "[Edit]", and "[Create PHP code]".

##### Penjelasan:

Membuat tabel dari Ms\_Toko yang berfungsi untuk menyimpan informasi mengenai toko.

Kunci utama dari IdToko memiliki tipe data char(6), yang berarti terdiri dari 6 karakter, di mana kunci utama ini harus bersifat unik dan tidak boleh bernilai null, sehingga dapat mengidentifikasi setiap record secara unik.

Nama\_toko memiliki tipe data varchar(255), yang menunjukkan bahwa kolom ini dapat menampung maksimal 255 karakter, dan not null menunjukkan bahwa kolom ini wajib diisi, berfungsi untuk menyimpan alamat toko.

Kolom no\_tlp bertipe bigint, digunakan untuk menyimpan nomor telepon toko, di mana not null menunjukkan bahwa kolom ini juga wajib diisi. Penggunaan tipe bigint memungkinkan penyimpanan angka dengan panjang besar, sehingga cocok untuk nomor telepon.

## v. Create Table Ms\_Parfum

### Query:

```
CREATE TABLE Ms_Parfum ( idParfum CHAR(6) PRIMARY KEY, nama_parfum VARCHAR(255) NOT NULL, stok_tersedia INT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### Output:

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0207 seconds.)  
CREATE TABLE Ms_Parfum ( idParfum CHAR(6) PRIMARY KEY, nama_parfum VARCHAR(255) NOT NULL, stok_tersedia INT NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

### Penjelasan:

Membuat tabel dari Ms\_Parfum yang berfungsi untuk menyimpan informasi mengenai parfum yang tersedia.

Kunci utama dari IdParfum memiliki tipe data char(6), di mana kunci utama tersebut harus bersifat unik dan tidak boleh bernilai null, karena digunakan untuk mengidentifikasi parfum secara eksklusif.

Kolom Nama\_parfum memiliki tipe data varchar(255), yang berarti dapat menyimpan hingga 255 karakter, dan ditandai sebagai not null, yang menunjukkan bahwa kolom ini wajib diisi untuk menyimpan nama parfum.

## c. Create Table Relasi

### i. Create Table Trx\_Laundry

### Query:

```
CREATE TABLE Trx_Laundry ( no_struk CHAR(6) PRIMARY KEY, idParfum CHAR(6), idPelanggan CHAR(6), idKasir CHAR(6), idToko CHAR(6), grand_total INT, dp INT, sisa INT, tgl_transaksi TIMESTAMP DEFAULT CURRENT_TIMESTAMP, FOREIGN KEY (idParfum) REFERENCES Ms_Parfum(idParfum) ON DELETE CASCADE, FOREIGN KEY (idPelanggan) REFERENCES Ms_Pelanggan(idPelanggan) ON DELETE CASCADE, FOREIGN KEY (idKasir) REFERENCES Ms_Kasir(idKasir) ON DELETE CASCADE, FOREIGN KEY (idToko) REFERENCES Ms_Toko(idToko) ON DELETE CASCADE ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### Output:

```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0285 seconds)

CREATE TABLE Trx_Laundry (
    no_struk CHAR(6) PRIMARY KEY,
    idParfum CHAR(6),
    idPelanggan CHAR(6),
    idKasir CHAR(6),
    idToko CHAR(6),
    grand_total INT,
    dp INT,
    sisa INT,
    tgl_transaksi TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (idParfum) REFERENCES Ms_Parfum(idParfum) ON DELETE CASCADE,
    FOREIGN KEY (idPelanggan) REFERENCES Ms_Pelanggan(idPelanggan) ON DELETE CASCADE,
    FOREIGN KEY (idKasir) REFERENCES Ms_Kasir(idKasir) ON DELETE CASCADE,
    FOREIGN KEY (idToko) REFERENCES Ms_Toko(idToko) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

[Edit inline] [Edit] [Create PHP code]

```

### Penjelasan:

Membuat tabel dari Trx\_Laundry bertujuan untuk menyimpan data transaksi laundry. Primary key yang digunakan adalah no\_struk dengan tipe data char(6), yang berarti terdiri dari 6 karakter.

Primary key ini harus unik dan tidak boleh bernilai null, sehingga setiap transaksi dapat diidentifikasi dengan jelas.

Id\_parfum memiliki tipe data char(6) yang berfungsi untuk menyimpan kode parfum yang digunakan dalam transaksi. Terdapat relasi foreign key ke tabel Ms\_Parfum (idParfum), yang mengharuskan data parfum sudah ada di tabel tersebut. Fitur on delete cascade akan menghapus transaksi terkait jika data parfum dihapus dari tabel Ms\_Parfum.

Idpelanggan juga menggunakan tipe data char(6) untuk menyimpan identitas pelanggan yang melakukan transaksi. Foreign key berasal dari tabel Ms\_Pelanggan (idPelanggan), yang memastikan bahwa hanya pelanggan terdaftar yang dapat melakukan transaksi.

IdToko memiliki tipe data char(6) yang mencatat lokasi toko tempat transaksi berlangsung. Foreign key dari tabel Ms\_Toko (idToko) memastikan bahwa hanya toko yang terdaftar dan valid yang dapat digunakan.

Grand\_total menggunakan tipe data int untuk menyimpan total biaya transaksi laundry dalam bentuk bilangan bulat, dan setiap nilai ini harus dicatat untuk setiap transaksi.

Dp memiliki tipe data yang sama dengan grand\_total, yaitu int, yang menyimpan jumlah uang muka yang telah dibayarkan oleh pelanggan. Ini penting untuk mencatat transaksi yang dilakukan secara cicilan.

Sisa juga menggunakan tipe data yang sama seperti grand\_total dan dp, dan menyimpan sisa pembayaran yang harus dilunasi oleh pelanggan. Nilai sisa dihitung sebagai selisih antara grand\_total dan dp.

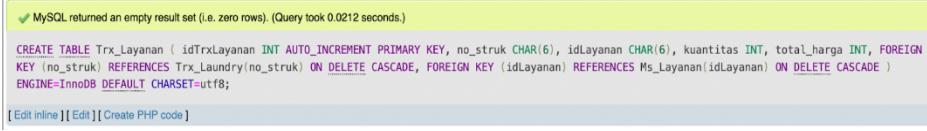
Tgl\_transaksi memiliki tipe data timestamp, yang dapat mencatat waktu transaksi secara otomatis dengan nilai default current\_timestamp. Timestamp pada tgl\_transaksi juga berfungsi untuk mencatat kapan transaksi dilakukan.

## ii. Create Table Trx\_Layanan

### Query:

```
CREATE TABLE Trx_Layanan ( idTrxLayanan INT AUTO_INCREMENT PRIMARY KEY, no_struk CHAR(6), idLayanan CHAR(6), kuantitas INT, total_harga INT, FOREIGN KEY (no_struk) REFERENCES Trx_Laundry(no_struk) ON DELETE CASCADE, FOREIGN KEY (idLayanan) REFERENCES Ms_Layanan(idLayanan) ON DELETE CASCADE ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### Output:



MySQL returned an empty result set (i.e. zero rows). (Query took 0.0212 seconds.)

```
CREATE TABLE Trx_Layanan ( idTrxLayanan INT AUTO_INCREMENT PRIMARY KEY, no_struk CHAR(6), idLayanan CHAR(6), kuantitas INT, total_harga INT, FOREIGN KEY (no_struk) REFERENCES Trx_Laundry(no_struk) ON DELETE CASCADE, FOREIGN KEY (idLayanan) REFERENCES Ms_Layanan(idLayanan) ON DELETE CASCADE ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

[Edit inline] [Edit] [Create PHP code]

### Penjelasan:

Membuat tabel Trx\_Layanan bertujuan untuk menyimpan rincian layanan yang dipilih dalam transaksi laundry. Kunci utama, idTrxLayanan, memiliki tipe data int dengan auto\_increment, yang berfungsi untuk menyimpan data dari id transaksi layanan.

Fitur auto\_increment akan secara otomatis menambah nilai setiap kali ada record baru, sedangkan kunci utama memastikan bahwa setiap transaksi layanan dapat diidentifikasi dengan jelas.

Kolom no\_struk memiliki tipe data char(6) dan berfungsi untuk menyimpan nomor struk transaksi yang merujuk pada tabel Trx\_Laundry.

Foreign key dari Trx\_Laundry(no\_struk) memastikan bahwa layanan ini hanya terhubung dengan transaksi yang sudah ada atau yang tersedia. Selain itu, fitur on delete cascade berfungsi untuk menghapus data layanan terkait jika data transaksi dihapus dari tabel Trx\_Laundry.

## d. Input 20 Rows ke Table Master

### i. Table Ms\_Pelanggan

### Query:

```
INSERT INTO Ms_Pelanggan (idPelanggan, nama_pelanggan)
VALUES ('PLG001', 'Ahmad Setiawan'), ('PLG002', 'Dewi Lestari'),
('PLG003', 'Budi Santoso'), ('PLG004', 'Citra Maharani'), ('PLG005', 'Eka Pratama'),
('PLG006', 'Farah Indah'), ('PLG007', 'Gilang Permana'),
('PLG008', 'Hadi Saputra'), ('PLG009', 'Irma Dwi'), ('PLG010', 'Joko')
```

Rahmat'), ('PLG011', 'Kurniawati Ayu'), ('PLG012', 'Lia Santika'), ('PLG013', 'Mario Budiman'), ('PLG014', 'Nina Maharani'), ('PLG015', 'Oki Kurniawan'), ('PLG016', 'Putri Anjani'), ('PLG017', 'Rina Setiani'), ('PLG018', 'Satria Bima'), ('PLG019', 'Tina Handayani'), ('PLG020', 'Utami Lestari');

### Output:

```
✓ 20 rows inserted. (Query took 0.0112 seconds.)  
INSERT INTO Ms_Pelanggan (idPelanggan, nama_pelanggan) VALUES ('PLG001', 'Ahmad Setiawan'), ('PLG002', 'Dewi Lestari'), ('PLG003', 'Budi Santoso'), ('PLG004', 'Citra Maharani'), ('PLG005', 'Eka Pratama'), ('PLG006', 'Farah Indah'), ('PLG007', 'Gilang Permana'), ('PLG008', 'Hadi Saputra'), ('PLG009', 'Irma Dwit'), ('PLG010', 'Joko Rahmat'), ('PLG011', 'Kurniawati Ayu'), ('PLG012', 'Lia Santika'), ('PLG013', 'Mario Budiman'), ('PLG014', 'Nina Maharani'), ('PLG015', 'Oki Kurniawan'), ('PLG016', 'Putri Anjani'), ('PLG017', 'Rina Setiani'), ('PLG018', 'Satria Bima'), ('PLG019', 'Tina Handayani'), ('PLG020', 'Utami Lestari');
```

### Penjelasan:

Menambahkan data baru ke dalam tabel Ms\_Pelanggan, perintah insert into\_pelanggan ini menunjukkan data baru yang akan dimasukkan ke dalam tabel tersebut.

Kolom idpelanggan memiliki tipe data char(6), yang berarti terdiri dari 6 karakter, sedangkan kolom nama pelanggan menggunakan tipe data varchar, yang diisi dengan nama lengkap pelanggan.

Proses ini akan dilakukan dengan menambahkan 20 entri baru dari data pelanggan ke dalam Ms\_Pelanggan.

Untuk nilai-nilai yang dimasukkan, akan mengikuti format (idPelanggan, nama\_pelanggan).

## ii. Table Ms\_Kasir

### Query:

```
INSERT INTO Ms_Kasir (idKasir, nama_kasir) VALUES ('KSR001', 'Dian Suryani'), ('KSR002', 'Rizky Permata'), ('KSR003', 'Edi Santoso'), ('KSR004', 'Faisal Anwar'), ('KSR005', 'Ika Nurul'), ('KSR006', 'Beni Setiawan'), ('KSR007', 'Lutfi Ramadhan'), ('KSR008', 'Vina Lestari'), ('KSR009', 'Wawan Gunawan'), ('KSR010', 'Yuni Indah'), ('KSR011', 'Nani Kurnia'), ('KSR012', 'Arif Widodo'), ('KSR013', 'Dewi Sartika'), ('KSR014', 'Fajar Maulana'), ('KSR015', 'Hendra Jaya'), ('KSR016', 'Irfan Saputra'), ('KSR017', 'Maya Kusuma'), ('KSR018', 'Rina Nuraini'), ('KSR019', 'Taufik Rahman'), ('KSR020', 'Zahra Fadillah');
```

### Output:

```
✓ 20 rows inserted. (Query took 0.0029 seconds.)  
  
INSERT INTO Ms_Kasir (idKasir, nama_kasir) VALUES ('KSR001', 'Dian Suryani'), ('KSR002', 'Rizky Per mata'), ('KSR003', 'Edi Santoso'), ('KSR004', 'Faisal Amwar'), ('KSR005', 'Ika Nurul'), ('KSR006', 'Beni Setiawan'), ('KSR007', 'Lutfi Ramdhani'), ('KSR008', 'Vina Lestari'), ('KSR009', 'Wawan Gunawan'), ('KSR010', 'Yuni Indah'), ('KSR011', 'Nani Kurnia'), ('KSR012', 'Arif Widodo'), ('KSR013', 'Dewi Sartika'), ('KSR014', 'Fajair Maulana'), ('KSR015', 'Hendra Jaya'), ('KSR016', 'Irwan Saputra'), ('KSR017', 'Maya Kusuma'), ('KSR018', 'Rina Nuraini'), ('KSR019', 'Taufik Rahman'), ('KSR020', 'Zahra Fadillah');  
  
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

### Penjelasan:

Menambahkan informasi baru ke dalam tabel Ms\_Kasir bertujuan untuk menyimpan data mengenai kasir. Perintah insert into Ms\_Kasir menunjukkan bahwa data baru akan ditambahkan ke tabel tersebut.

Kolom idKasir memiliki tipe data char(6), yang berarti dapat menyimpan hingga 6 karakter. Sementara itu, kolom nama\_kasir menggunakan tipe data varchar, yang berisi nama lengkap kasir.

Nilai yang dimasukkan mengikuti format (idKasir, nama\_kasir) yang mencerminkan daftar data kasir.

### iii. Table Ms\_Layanan

#### Query:

```
INSERT INTO Ms_Layanan (idLayanan, nama_layanan, satuan, harga)  
VALUES ('LYN001', 'Cuci Kering', 'Kg', 15000), ('LYN002', 'Cuci Basah', 'Kg', 10000),  
('LYN003', 'Cuci Kering Lipat', 'Kg', 20000), ('LYN004', 'Setrika Saja', 'Kg', 8000),  
('LYN005', 'Dry Cleaning', 'Pcs', 25000), ('LYN006', 'Cuci Boneka', 'Pcs', 30000),  
('LYN007', 'Cuci Karpet', 'Pcs', 50000), ('LYN008', 'Cuci Selimut', 'Pcs', 40000),  
('LYN009', 'Cuci Gordyn', 'Pcs', 60000), ('LYN010', 'Laundry Ekspres', 'Kg', 30000),  
('LYN011', 'Laundry Regular', 'Kg', 12000), ('LYN012', 'Cuci Jaket', 'Pcs', 20000),  
('LYN013', 'Cuci Sepatu', 'Pcs', 15000), ('LYN014', 'Cuci Helm', 'Pcs', 12000), ('LYN015', 'Cuci Topi', 'Pcs', 10000),  
('LYN016', 'Cuci Tas', 'Pcs', 25000), ('LYN017', 'Cuci Baju Delicate', 'Kg', 40000), ('LYN018', 'Laundry Jas', 'Pcs', 35000),  
('LYN019', 'Laundry Hotel', 'Kg', 50000), ('LYN020', 'Laundry Industrial', 'Kg', 80000);
```

#### Output:

```
✓ 20 rows inserted. (Query took 0.0034 seconds.)  
  
INSERT INTO Ms_Layanan (idLayanan, nama_layanan, satuan, harga) VALUES ('LYN001', 'Cuci Kering', 'Kg', 15000), ('LYN002', 'Cuci Basah', 'Kg', 10000),  
('LYN003', 'Cuci Kering Lipat', 'Kg', 20000), ('LYN004', 'Setrika Saja', 'Kg', 8000), ('LYN005', 'Dry Cleaning', 'Pcs', 25000), ('LYN006', 'Cuci Boneka', 'Pcs', 30000),  
('LYN007', 'Cuci Karpet', 'Pcs', 50000), ('LYN008', 'Cuci Selimut', 'Pcs', 40000), ('LYN009', 'Cuci Gordyn', 'Pcs', 60000),  
('LYN010', 'Laundry Ekspres', 'Kg', 30000), ('LYN011', 'Laundry Regular', 'Kg', 12000), ('LYN012', 'Cuci Jaket', 'Pcs', 20000), ('LYN013', 'Cuci Sepatu', 'Pcs', 15000),  
('LYN014', 'Cuci Helm', 'Pcs', 12000), ('LYN015', 'Cuci Topi', 'Pcs', 10000), ('LYN016', 'Cuci Tas', 'Pcs', 25000), ('LYN017', 'Cuci Baju Delicate', 'Kg', 40000), ('LYN018', 'Laundry Jas', 'Pcs', 35000), ('LYN019', 'Laundry Hotel', 'Kg', 50000), ('LYN020', 'Laundry Industrial', 'Kg', 80000);  
  
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

**Penjelasan:**

Menambahkan data baru ke dalam tabel Ms\_Layanan yang berisi informasi mengenai layanan laundry.

Proses ini dilakukan dengan perintah insert into Ms\_Layanan, yang menunjukkan bahwa data baru akan dimasukkan ke dalam tabel tersebut. Idlayanan memiliki tipe data char(6), yang berarti hanya terdiri dari 6 karakter.

Nama\_layanan berisi jenis layanan laundry dan memiliki tipe data varchar. Satuan dalam query menunjukkan unit layanan, seperti kg atau pcs, yang merujuk pada berat per item.

Selanjutnya, harga merupakan biaya untuk setiap layanan yang dinyatakan dalam bentuk angka integer (int).

Nilai-nilai yang dimasukkan mencakup idLayanan, nama\_layanan, satuan, dan harga.

**iv. Table Ms\_Toko****Query:**

```
INSERT INTO Ms_Toko (idToko, nama_toko, alamat, no_tlp)
VALUES ('TOK001', 'Laundry Center 1', 'Jl. Mawar No.1', 6281234567890), ('TOK002', 'Laundry Center 2', 'Jl. Melati No.2', 6281234567891), ('TOK003', 'Laundry Center 3', 'Jl. Anggrek No.3', 6281234567892), ('TOK004', 'Laundry Center 4', 'Jl. Kenanga No.4', 6281234567893), ('TOK005', 'Laundry Center 5', 'Jl. Dahlia No.5', 6281234567894), ('TOK006', 'Laundry Center 6', 'Jl. Teratai No.6', 6281234567895), ('TOK007', 'Laundry Center 7', 'Jl. Kemuning No.7', 6281234567896), ('TOK008', 'Laundry Center 8', 'Jl. Cempaka No.8', 6281234567897), ('TOK009', 'Laundry Center 9', 'Jl. Bougenville No.9', 6281234567898), ('TOK010', 'Laundry Center 10', 'Jl. Kamboja No.10', 6281234567899), ('TOK011', 'Laundry Center 11', 'Jl. Mawar No.11', 6281234567800), ('TOK012', 'Laundry Center 12', 'Jl. Melati No.12', 6281234567801), ('TOK013', 'Laundry Center 13', 'Jl. Anggrek No.13', 6281234567802), ('TOK014', 'Laundry Center 14', 'Jl. Kenanga No.14', 6281234567803), ('TOK015', 'Laundry Center 15', 'Jl. Dahlia No.15', 6281234567804), ('TOK016', 'Laundry Center 16', 'Jl. Teratai No.16', 6281234567805), ('TOK017', 'Laundry Center 17', 'Jl. Kemuning No.17', 6281234567806), ('TOK018', 'Laundry Center 18', 'Jl. Cempaka No.18', 6281234567807), ('TOK019', 'Laundry Center 19', 'Jl. Bougenville No.19', 6281234567808), ('TOK020', 'Laundry Center 20', 'Jl. Kamboja No.20', 6281234567809);
```

### Output:

```
✓ 20 rows inserted. (Query took 0.0024 seconds.)  
  
INSERT INTO Ms_Toko (idToko, nama_toko, alamat, no_tlp) VALUES ('TOK001', 'Laundry Center 1', 'Jl. Mawar No.1', 6281234567890), ('TOK002', 'Laundry Center 2', 'Jl. Melati No.2', 6281234567891), ('TOK003', 'Laundry Center 3', 'Jl. Angrek No.3', 6281234567892), ('TOK004', 'Laundry Center 4', 'Jl. Kenanga No.4', 6281234567893), ('TOK005', 'Laundry Center 5', 'Jl. Dahlia No.5', 6281234567894), ('TOK006', 'Laundry Center 6', 'Jl. Teratai No.6', 6281234567895), ('TOK007', 'Laundry Center 7', 'Jl. Kemuning No.7', 6281234567896), ('TOK008', 'Laundry Center 8', 'Jl. Cempaka No.8', 6281234567897), ('TOK009', 'Laundry Center 9', 'Jl. Bougenville No.9', 6281234567898), ('TOK010', 'Laundry Center 10', 'Jl. Kamboja No.10', 6281234567899), ('TOK011', 'Laundry Center 11', 'Jl. Mawar No.11', 6281234567800), ('TOK012', 'Laundry Center 12', 'Jl. Melati No.12', 6281234567801), ('TOK013', 'Laundry Center [Edit]
```

### Penjelasan:

Menambahkan informasi baru ke dalam tabel Ms\_Toko yang berisi data mengenai toko laundry, perintah insert into Ms\_Toko menunjukkan bahwa data baru akan dimasukkan ke dalam tabel tersebut.

Nilai-nilai yang dimasukkan mencakup idToko, nama\_toko, alamat, dan no\_telp.

## v. Table Ms\_Parfum

### Query:

```
INSERT INTO Ms_Parfum (idParfum, nama_parfum, stok_tersedia)  
VALUES ('PRF001', 'Lavender', 100), ('PRF002', 'Rose', 80), ('PRF003',  
'Jasmine', 120), ('PRF004', 'Ocean', 150), ('PRF005', 'Lemon', 90),  
(('PRF006', 'Vanilla', 70), ('PRF007', 'Musk', 110), ('PRF008',  
'Sandalwood', 50), ('PRF009', 'Mint', 130), ('PRF010', 'Citrus', 140),  
(('PRF011', 'Pine', 60), ('PRF012', 'Amber', 85), ('PRF013', 'Cedarwood',  
75), ('PRF014', 'Peach', 65), ('PRF015', 'Apple', 95), ('PRF016',  
'Strawberry', 115), ('PRF017', 'Grapefruit', 125), ('PRF018', 'Coconut',  
55), ('PRF019', 'Cherry', 105), ('PRF020', 'Blueberry', 135);
```

### Output:

```
✓ 20 rows inserted. (Query took 0.0031 seconds.)  
  
INSERT INTO Ms_Parfum (idParfum, nama_parfum, stok_tersedia) VALUES ('PRF001', 'Lavender', 100), ('PRF002', 'Rose', 80), ('PRF003', 'Jasmine', 120),  
(('PRF004', 'Ocean', 150), ('PRF005', 'Lemon', 90), ('PRF006', 'Vanilla', 70), ('PRF007', 'Musk', 110), ('PRF008', 'Sandalwood', 50), ('PRF009',  
'Mint', 130), ('PRF010', 'Citrus', 140), ('PRF011', 'Pine', 60), ('PRF012', 'Amber', 85), ('PRF013', 'Cedarwood', 75), ('PRF014', 'Peach', 65),  
(('PRF015', 'Apple', 95), ('PRF016', 'Strawberry', 115), ('PRF017', 'Grapefruit', 125), ('PRF018', 'Coconut', 55), ('PRF019', 'Cherry', 105),  
(('PRF020', 'Blueberry', 135);
```

### Penjelasan:

Menambahkan informasi baru ke dalam tabel Ms\_Parfum yang berisi rincian parfum yang tersedia di laundry, perintah insert into Ms\_Parfum menunjukkan bahwa data baru akan dimasukkan ke dalam tabel tersebut.

Untuk nilai-nilai ini, datanya berasal dari idParfum, nama\_parfum, dan stok\_tersedia.

#### e. Implementasi Konsep Transaction

**Query:**

```
DELIMITER $$ CREATE PROCEDURE InsertTrxLaundry(IN p_no_struk
CHAR(6), IN p_idParfum CHAR(6), IN p_idPelanggan CHAR(6), IN
p_idKasir CHAR(6), IN p_idToko CHAR(6), IN p_layanan1 CHAR(6), IN
p_kuantitas_layanan1 INT, IN p_layanan2 CHAR(6), IN p_kuantitas_layanan2
INT, IN p_layanan3 CHAR(6), IN p_kuantitas_layanan3 INT, IN p_dp INT)
BEGIN DECLARE p_stok_tersedia INT; DECLARE p_harga_layanan1,
p_harga_layanan2, p_harga_layanan3 INT DEFAULT 0; DECLARE
p_total_harga_layanan1, p_total_harga_layanan2, p_total_harga_layanan3 INT
DEFAULT 0; DECLARE p_grand_total INT DEFAULT 0; DECLARE
p_total_kuantitas INT DEFAULT 0; transaksi_block: BEGIN IF p_layanan2 IS
NULL THEN SET p_layanan2 = ""; END IF; IF p_kuantitas_layanan2 IS NULL
THEN SET p_kuantitas_layanan2 = 0; END IF; IF p_layanan3 IS NULL
THEN SET p_layanan3 = ""; END IF; IF p_kuantitas_layanan3 IS NULL THEN
SET p_kuantitas_layanan3 = 0; END IF; START TRANSACTION; IF
p_layanan1 IS NOT NULL AND p_kuantitas_layanan1 > 0 THEN SELECT
harga INTO p_harga_layanan1 FROM Ms_Layanan WHERE idLayanan =
p_layanan1; SET p_total_harga_layanan1 = p_harga_layanan1 *
p_kuantitas_layanan1; SET p_total_kuantitas = p_total_kuantitas +
p_kuantitas_layanan1; END IF; IF p_layanan2 IS NOT NULL AND
p_kuantitas_layanan2 > 0 THEN SELECT harga INTO p_harga_layanan2
FROM Ms_Layanan WHERE idLayanan = p_layanan2; SET
p_total_harga_layanan2 = p_harga_layanan2 * p_kuantitas_layanan2; SET
p_total_kuantitas = p_total_kuantitas + p_kuantitas_layanan2; END IF; IF
p_layanan3 IS NOT NULL AND p_kuantitas_layanan3 > 0 THEN SELECT
harga INTO p_harga_layanan3 FROM Ms_Layanan WHERE idLayanan =
p_layanan3; SET p_total_harga_layanan3 = p_harga_layanan3 *
p_kuantitas_layanan3; SET p_total_kuantitas = p_total_kuantitas +
p_kuantitas_layanan3; END IF; SET p_grand_total = p_total_harga_layanan1 +
p_total_harga_layanan2 + p_total_harga_layanan3; INSERT INTO
Trx_Laundry (no_struk, idParfum, idPelanggan, idKasir, idToko, grand_total,
dp, sisa, tgl_transaksi) VALUES (p_no_struk, p_idParfum, p_idPelanggan,
p_idKasir, p_idToko, p_grand_total, p_dp, p_grand_total - p_dp,
CURRENT_TIMESTAMP); IF p_layanan1 IS NOT NULL AND
p_kuantitas_layanan1 > 0 THEN INSERT INTO Trx_Layanan (no_struk,
idLayanan, kuantitas, total_harga) VALUES (p_no_struk, p_layanan1,
p_kuantitas_layanan1, p_total_harga_layanan1); END IF; IF p_layanan2 IS
NOT NULL AND p_kuantitas_layanan2 > 0 THEN INSERT INTO
Trx_Layanan (no_struk, idLayanan, kuantitas, total_harga) VALUES
(p_no_struk, p_layanan2, p_kuantitas_layanan2, p_total_harga_layanan2);
END IF; IF p_layanan3 IS NOT NULL AND p_kuantitas_layanan3 > 0 THEN
INSERT INTO Trx_Layanan (no_struk, idLayanan, kuantitas, total_harga)
VALUES (p_no_struk, p_layanan3, p_kuantitas_layanan3,
p_total_harga_layanan3); END IF; SELECT stok_tersedia INTO
p_stok_tersedia FROM Ms_Parfum WHERE idParfum = p_idParfum; IF
```

```

p_stok_tersedia < p_total_kuantitas THEN ROLLBACK; SELECT 'Transaksi Gagal: Stok Parfum Tidak Mencukupi.' AS message; LEAVE transaksi_block;
END IF; UPDATE Ms_Parfum SET stok_tersedia = stok_tersedia - p_total_kuantitas WHERE idParfum = p_idParfum; COMMIT; SELECT 'Transaksi berhasil' AS Status, p_no_struk AS No_Struk; END transaksi_block;
END$$ DELIMITER ;

```

### Output:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0276 seconds.)

```

CREATE PROCEDURE InsertTrxLaundry( IN p_no_struk CHAR(6), IN p_idParfum CHAR(6), IN p_idPelanggan CHAR(6), IN p_idKasir CHAR(6), IN p_idToko CHAR(6),
IN p_layanan1 CHAR(6), IN p_kuantitas_layanan1 INT, IN p_layanan2 CHAR(6), IN p_kuantitas_layanan2 INT, IN p_layanan3 CHAR(6), IN p_kuantitas_layanan3
INT, IN p_dp INT ) BEGIN -- Deklarasi variabel lokal DECLARE p_stok_tersedia INT; DECLARE p_harga_layanan1, p_harga_layanan2, p_harga_layanan3 INT
DEFAULT 0; DECLARE p_total_harga_layanan1, p_total_harga_layanan2, p_total_harga_layanan3 INT DEFAULT 0; DECLARE p_grand_total INT DEFAULT 0; DECLARE
p_total_kuantitas INT DEFAULT 0; -- Label untuk prosedur transaksi_block: BEGIN -- Nilai default jika parameter NULL IF p_layanan2 IS NULL THEN SET
p_layanan2 = ''; END IF; IF p_kuantitas_layanan2 IS NULL THEN SET p_kuantitas_layana[...]

```

[Edit]

### Penjelasan:

Menyusun prosedur yang disebut InsertTrxLaundry untuk mengelola proses transaksi laundry secara otomatis.

Prosedur ini memiliki tanggung jawab untuk menghitung total biaya layanan, memeriksa ketersediaan stok parfum, menyimpan informasi transaksi ke dalam tabel Trx\_Laundry dan Trx\_Layanan, serta memperbarui stok parfum di tabel Ms\_Parfum, dengan melakukan rollback apabila stok tidak mencukupi.

### f. Input 20 Rows ke Table Relasi Menggunakan Transaction

#### Query:

```

START TRANSACTION; CALL InsertTrxLaundry('TRX001', 'PRF001', 'PLG001', 'KSR001', 'TOK001', 'LYN001', 10, NULL, 0, NULL, 0, 20000);
CALL InsertTrxLaundry('TRX002', 'PRF002', 'PLG002', 'KSR002', 'TOK002', 'LYN002', 5, 'LYN003', 3, NULL, 0, 15000); CALL
InsertTrxLaundry('TRX003', 'PRF003', 'PLG003', 'KSR003', 'TOK003', 'LYN001', 8, 'LYN004', 2, NULL, 0, 18000); CALL
InsertTrxLaundry('TRX004', 'PRF004', 'PLG004', 'KSR004', 'TOK001', 'LYN003', 7, NULL, 0, 14000); CALL InsertTrxLaundry('TRX005',
'PRF005', 'PLG005', 'KSR005', 'TOK002', 'LYN004', 4, 'LYN002', 6, 'LYN001', 3, 22000); CALL InsertTrxLaundry('TRX006', 'PRF001', 'PLG006', 'KSR006',
'TOK003', 'LYN001', 6, NULL, 0, NULL, 0, 16000); CALL InsertTrxLaundry('TRX007', 'PRF002', 'PLG007', 'KSR007', 'TOK001',
'LYN003', 3, 'LYN002', 4, NULL, 0, 14000); CALL InsertTrxLaundry('TRX008', 'PRF003', 'PLG008', 'KSR008', 'TOK002',
'LYN002', 9, 'LYN004', 2, NULL, 0, 25000); CALL InsertTrxLaundry('TRX009', 'PRF004', 'PLG009', 'KSR009', 'TOK003',
'LYN001', 5, 'LYN003', 4, 'LYN002', 2, 30000); CALL InsertTrxLaundry('TRX010', 'PRF005', 'PLG010', 'KSR010', 'TOK001',
'LYN004', 2, NULL, 0, NULL, 0, 10000); CALL InsertTrxLaundry('TRX011', 'PRF001', 'PLG011', 'KSR011', 'TOK002', 'LYN001', 7,
NULL, 0, NULL, 0, 17000); CALL InsertTrxLaundry('TRX012', 'PRF002', 'PLG012', 'KSR012', 'TOK003', 'LYN002', 4, 'LYN003', 3,
NULL, 0, 15000); CALL

```

```

InsertTrxLaundry('TRX013', 'PRF003', 'PLG013', 'KSR013', 'TOK001',
'LYN003', 8, 'LYN004', 1, NULL, 0, 20000); CALL
InsertTrxLaundry('TRX014', 'PRF004', 'PLG014', 'KSR014', 'TOK002',
'LYN001', 6, 'LYN003', 2, NULL, 0, 19000); CALL
InsertTrxLaundry('TRX015', 'PRF005', 'PLG015', 'KSR015', 'TOK003',
'LYN004', 3, 'LYN002', 5, 'LYN001', 2, 22000); CALL
InsertTrxLaundry('TRX016', 'PRF001', 'PLG016', 'KSR016', 'TOK001',
'LYN001', 4, NULL, 0, NULL, 0, 12000); CALL InsertTrxLaundry('TRX017',
'PRF002', 'PLG017', 'KSR017', 'TOK002', 'LYN002', 10, NULL, 0, NULL, 0,
24000); CALL InsertTrxLaundry('TRX018', 'PRF003', 'PLG018', 'KSR018',
'TOK003', 'LYN003', 2, 'LYN001', 3, NULL, 0, 13000); CALL
InsertTrxLaundry('TRX019', 'PRF004', 'PLG019', 'KSR019', 'TOK001',
'LYN004', 5, NULL, 0, NULL, 0, 15000); CALL InsertTrxLaundry('TRX020',
'PRF005', 'PLG020', 'KSR020', 'TOK002', 'LYN001', 6, 'LYN002', 3, 'LYN003',
1, 18000); COMMIT;

```

### Output:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0001 seconds.)

START TRANSACTION;

[ Edit inline ] [ Edit ] [ Create PHP code ]

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 0 (1 total, Query took 0.0271 seconds.)

CALL InsertTrxLaundry ('TRX001', 'PRF001', 'PLG001', 'KSR001', 'TOK001', 'LYN001', 10, NULL, 0, NULL, 0, 20000);

Show all | Number of rows: 25 Filter rows: Search this table

Status No\_Sruk

Status	No_Sruk
Transaksi berhasil	TRX001

### Penjelasan:

Membuat data dummy yang terdiri dari 20 baris ke dalam tabel relasi melalui transaksi.

Setiap transaksi mencakup data seperti nomor struk, ID parfum, identitas pelanggan, kasir, lokasi toko, jenis layanan, jumlah layanan, dan uang muka. Setelah seluruh transaksi berhasil diproses, perintah diakhiri dengan COMMIT untuk menyimpan perubahan secara permanen dalam basis data.

### g. Skenario Kesalahan pada Transaction

#### Query:

```
CALL InsertTrxLaundry('TRX021', 'PRF002', 'PLG019', 'KSR019', 'TOK001',
'LYN004', 55, NULL, 0, NULL, 0, 20000);
```

#### Output:

Showing rows 0 - 0 (1 total, Query took 0.0072 seconds.)

```
CALL InsertTrxLaundry('TRX021', 'PRF002', 'PLG019', 'KSR019', 'TOK001', 'LYN004', 55, NULL, 0, NULL, 0, 20000);
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

Show all | Number of rows: 25 Filter rows: Search this table

Extra options

**message**

Transaksi Gagal: Stok Parfum Tidak Mencukupi.

### Penjelasan:

Menyusun skenario kesalahan dalam transaksi di mana jumlah layanan LYN004 yang dipesan (55 unit) melebihi jumlah stok parfum PRF002 yang ada, prosedur akan mengidentifikasi adanya kekurangan stok.

Transaksi tersebut akan dibatalkan secara otomatis melalui ROLLBACK, dan akan muncul pesan kesalahan "Transaksi Gagal: Stok Parfum Tidak Mencukupi."

### h. Update Data Ms\_Parfum

#### Query:

```
UPDATE Ms_Parfum SET stok_tersedia = 200 WHERE idParfum = 'PRF002';
```

#### Output:

1 row affected. (Query took 0.0045 seconds.)

```
UPDATE Ms_Parfum SET stok_tersedia = 200 WHERE idParfum = 'PRF002';
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

	idParfum	nama_parfum	stok_tersedia
<input type="checkbox"/>	PRF001	Lavender	73
<input type="checkbox"/>	PRF002	Rose	200
<input type="checkbox"/>	PRF003	Jasmine	85
<input type="checkbox"/>	PRF004	Ocean	119
<input type="checkbox"/>	PRF005	Lemon	55

### Penjelasan:

Pembaruan pada Ms\_Parfum ini bertujuan untuk memperbarui informasi dalam tabel Ms\_Parfum itu sendiri.

Set stok\_tersedia merupakan query yang akan diperbarui, dengan nilai yang diubah menjadi 200.

Di mana idparfum berfungsi untuk menentukan kolom mana yang akan dipilih untuk diperbarui dalam query tersebut, yaitu nilai PRF2002 yang akan mengalami pembaruan.

### i. Delete Data Trx\_Laundry

**Query:**

```
DELETE FROM Trx_Laundry WHERE no_struk = 'TRX020';
```

**Output:**

<input type="checkbox"/>				TRX015	PRF005	PLG015	KSR015	TOK003	104000	22000	82000	2024-12-27 23:3				
<input type="checkbox"/>				TRX016	PRF001	PLG016	KSR016	TOK001	60000	12000	48000	2024-12-27 23:3				
<input type="checkbox"/>				TRX017	PRF002	PLG017	KSR017	TOK002	100000	24000	76000	2024-12-27 23:3				
<input type="checkbox"/>				TRX018	PRF003	PLG018	KSR018	TOK003	85000	13000	72000	2024-12-27 23:3				
<input type="checkbox"/>				TRX019	PRF004	PLG019	KSR019	TOK001	40000	15000	25000	2024-12-27 23:3				

**Penjelasan:**

Penghapusan itu sendiri bertujuan untuk menghilangkan pernyataan di atas menunjukkan bahwa data dari trx\_laundry akan dihapus, dengan syarat no\_struk yang dapat menentukan baris mana yang akan dihapus, di mana baris dengan no\_struk yang bernilai 'TRX020' akan dihapus.

## 5. Link Github

<https://github.com/xnoname2003/vtwolaundry.git>

## 6. Dokumentasi Pengerjaan

