

LAPORAN

PRAKTIKUM BASIS DATA

FINAL PROJECT KOPERASI KEMITRAAN SUKSES BERSAMA

22 JUNI 2023



DISUSUN OLEH:

1. Bagas Yudha Aditya(2231730096)
2. Faris Maulana Hilmi (2231730032)
3. Hety Rachmawati (2231730070)

DIII Manajemen Informatika

Program Studi di Luar Kampus Utama

Politeknik Negeri Malang Kampus Kediri

2023

BUAT DATABASE UASKOPERASI

```
DROP DATABASE IF EXISTS uaskoperasi;  
CREATE DATABASE uaskoperasi;  
USE uaskoperasi;
```

Penjelasan: query ini digunakan untuk membuat database yang bernama uaskoperasi database ini digunakan untuk menyimpan sebuah tabel2 yang di buat.

1. Buat table peternak

```
6 DROP TABLE IF EXISTS peternak;  
7  
8 CREATE TABLE peternak (  
9 kd_peternak VARCHAR(10) NOT NULL PRIMARY KEY,  
10 nama VARCHAR(20) NOT NULL,  
11 hp CHAR(10) NOT NULL,  
12 ternak_daging VARCHAR(10) NOT NULL,  
13 ternak_telur VARCHAR(10) NOT NULL);  
14
```

Penjelasan : Query ini digunakan untuk membuat tabel baru yang di beri nama peternak dengan ditambahkan 5 kolom dan setiap kolom memiliki tipe data tersendiri yang sesuai dengan kebutuhan, ada juga not null untuk untuk memberikan nilai kosong dan kode_peternak sebagai primarykey.

➤ DROP DATABASE IF EXISTS peternak;

Query ini bertujuan untuk menghapus tabel yang tersedia. Dengan menggunakan IF EXISTS, tabel akan dihapus hanya jika sudah ada. Ini dilakukan untuk memastikan bahwa kita memulai dengan tabel yang kosong.

- Tambahkan data kedalam table

```

INSERT INTO peternak (kd_peternak, nama, hp, ternak_daging, ternak_telur) VALUES
('LL001', 'Hety', '081234765', 'Ya', 'Tidak'),
('LL002', 'Yudha', '085223531', 'Ya', 'Tidak'),
('LL003', 'Hilmi', '081515855', 'Tidak', 'Ya'),
('LL004', 'Faris', '085349271', 'Ya', 'Ya'),
('LL005', 'Maulana', '085856328', 'Tidak', 'Ya'),
('LL006', 'Bagas', '081767857', 'Ya', 'Tidak'),
('LL007', 'Aditya', '085676456', 'Tidak', 'Ya'),
('LL008', 'Rahma', '085785865', 'Ya', 'Ya'),
('LL009', 'Wati', '085475405', 'Ya', 'Ya'),
('LL010', 'Siti', '081753431', 'Ya', 'Tidak');

SELECT *FROM peternak;

```

Penjelasan : Query ini bertujuan untuk mengisi data data yang ada ditabel peternak, setiap baris memiliki nilai untuk setiap kolom yang sesuai .

➤ SELECT * FROM peternak;

Query ini digunakan untuk menampilkan semua data yang ada dalam tabel 'peternak'. Dengan menggunakan tanda bintang (*) setelah SELECT, kita memilih semua kolom dalam tabel.

- Output

(Read Only)

</

2. Buat table dengan nama produk

```

29      -- 2. buat table dengan nama produk --
30      DROP TABLE IF EXISTS produk;
31
32      CREATE TABLE produk (
33          kd_produk VARCHAR(10) NOT NULL PRIMARY KEY,
34          nama VARCHAR(30) NOT NULL,
35          satuan VARCHAR(30) NOT NULL,
36          harga INT(20) NOT NULL
37      )
38

```

Penjelasan : Pernyataan ini digunakan untuk membuat tabel baru dengan nama "produk". Tabel ini memiliki empat kolom yaitu "kd_produk", "nama", "satuan", dan "harga".

- Kolom "kd_produk" memiliki tipe data VARCHAR(10), yang berarti dapat menampung string dengan panjang maksimal 10 karakter. Kolom ini diberi batasan NOT NULL, yang berarti harus selalu memiliki nilai.
- Kolom "nama" memiliki tipe data VARCHAR(30), yang berarti dapat menampung string dengan panjang maksimal 30 karakter. Kolom ini juga diberi batasan NOT NULL.
- Kolom "satuan" memiliki tipe data VARCHAR(30) dan juga diberi batasan NOT NULL.
- Kolom "harga" memiliki tipe data INT(20), yang berarti dapat menampung nilai integer dengan panjang maksimal 20 digit. Kolom ini juga diberi batasan NOT NULL.

Primary key pada tabel ditentukan di kolom "kd_produk", yang berarti menjadi kunci utama tabel.

- DROP TABLE IF EXISTS produk;

Query ini digunakan untuk menghapus tabel "produk" jika tabel tersebut sudah ada sebelumnya. Jika tabel tidak ada, pernyataan ini tidak akan menghasilkan kesalahan.

- Insert kan data kedalam table produk

```

39
40 INSERT INTO produk (kd_produk, nama, satuan, harga) VALUES
41 ('P011', 'DOC ayam kampung', 'ekor', 9000),
42 ('P012', 'AyamVita Plus', 'pcs', 120000),
43 ('P013', 'ChickenGuardian', 'pcs', 15000),
44 ('P014', 'PoultryVit', 'pcs', 10000),
45 ('P015', 'RoosterBoost', 'pcs', 9500),
46 ('P016', 'FeatherFortify', 'pcs', 8000),
47 ('P017', 'CluckyVitamins', 'pcs', 20000),
48 ('P018', 'AvianVita', 'pcs', 18000),
49 ('P019', 'Daging Ayam Kampung', 'kg', 35000),
50 ('P020', 'Telur Ayam Kampung', 'kg', 20000);
51
52
53 SELECT *FROM produk
54 |


```


Penjelasan : Query ini digunakan untuk memasukkan beberapa baris data ke dalam tabel "produk". Setiap baris data memasukkan nilai ke kolom-kolom yang sesuai dalam urutan yang ditentukan. Data diwakili oleh nilai-nilai dalam tanda kurung.

➤ SELECT FROM produk;


Query ini digunakan untuk mengambil semua data yang ada dalam tabel "produk". "*" digunakan untuk memilih semua kolom dalam tabel. Hasilnya adalah seluruh data yang ada dalam tabel "produk" akan ditampilkan.


- Output








(Read Only)




















<input type="checkbox"/>	kd_produk	nama	satuan	harga
<input type="checkbox"/>	P011	DOC ayam kampung	ekor	9000
<input type="checkbox"/>	P012	AyamVita Plus	pcs	120000
<input type="checkbox"/>	P013	ChickenGuardian	pcs	15000
<input type="checkbox"/>	P014	PoultryVit	pcs	10000
<input type="checkbox"/>	P015	RoosterBoost	pcs	9500
<input type="checkbox"/>	P016	FeatherFortify	pcs	8000
<input type="checkbox"/>	P017	CluckyVitamins	pcs	20000
<input type="checkbox"/>	P018	AvianVita	pcs	18000
<input type="checkbox"/>	P019	Daging Ayam Kampung	kg	35000
<input type="checkbox"/>	P020	Telur Ayam Kampung	kg	20000

3. Buat table dengan nama ms_limit_hutang

```

55      -- 3. buat table dengan nama ms_limit_hutang
56      DROP TABLE IF EXISTS ms_limit_hutang
57
58      CREATE TABLE ms_limit_hutang(
59      | kd_utang VARCHAR(10) NOT NULL PRIMARY KEY,
60      | batas_bawah INT(20),
61      | batas_atas INT(20),
62      | dt_limit VARCHAR(20) NOT NULL)
63

```

Penjelasan : Query ini digunakan untuk membuat tabel baru dengan nama "ms_limit_hutang". Tabel ini memiliki empat kolom yaitu "kd_utang", "batas_bawah", "batas_atas", dan "dt_limit".

- Kolom "kd_utang" memiliki tipe data VARCHAR(10), yang berarti dapat menampung string dengan panjang maksimal 10 karakter. Kolom ini diberi batasan NOT NULL, yang berarti harus selalu memiliki nilai.
- Kolom "batas_bawah" dan "batas_atas" memiliki tipe data INT(20), yang berarti dapat menampung nilai integer dengan panjang maksimal 20 digit.

- Kolom "dt_limit" memiliki tipe data VARCHAR(20) dan juga diberi batasan NOT NULL.

Primary key pada tabel ini berada di kolom "kd_utang", yang berarti nilai di kolom ini harus unik dan menjadi kunci utama tabel.

- DROP TABLE IF EXISTS ms_limit_hutang; Pernyataan ini digunakan untuk menghapus tabel "ms_limit_hutang" jika tabel tersebut sudah ada sebelumnya. Jika tabel tidak ada, pernyataan ini tidak akan menghasilkan kesalahan.
- Tambahkan data ke dalam tabel ms_limit_hutang

```
INSERT INTO ms_limit_hutang (kd_utang, batas_bawah, batas_atas, dt_limit) VALUES
('LHG01', '500', '999', '25JT');
('LHG02', '1000', '1999', '50JT');
('LHG03', '2000', NULL, '100JT');
('LHG04', '3000', NULL, '100JT');
('LHG05', '250', '499', '20JT');
('LHG06', '100', '249', '15JT');
('LHG07', '50', '100', '10JT');
('LHG08', '600', '900', '25JT');
('LHG09', '1500', '1900', '50JT');
('LHG10', '550', '700', '25JT');

SELECT * FROM ms_limit_hutang
```

Penjelasan : Query ini digunakan untuk memasukkan beberapa baris data ke dalam tabel "ms_limit_hutang". Setiap baris data memasukkan nilai ke kolom-kolom yang sesuai dalam urutan yang ditentukan.

- SELECT FROM ms_limit_hutang; query ini digunakan untuk mengambil semua data yang ada dalam tabel "ms_limit_hutang". "" digunakan untuk menampilkan seluruh data yang ada dalam tabel "ms_limit_hutang".

- Output

<input type="checkbox"/> kd_utang	batas_bawah	batas_atas	dt_limit
<input type="checkbox"/> LHG01	500	999	25JT
<input type="checkbox"/> LHG02	1000	1999	50JT
<input type="checkbox"/> LHG03	2000	(NULL)	100JT
<input type="checkbox"/> LHG04	3000	(NULL)	100JT
<input type="checkbox"/> LHG05	250	499	20JT
<input type="checkbox"/> LHG06	100	249	15JT
<input type="checkbox"/> LHG07	50	100	10JT
<input type="checkbox"/> LHG08	600	900	25JT
<input type="checkbox"/> LHG09	1500	1900	50JT
<input type="checkbox"/> LHG10	550	700	25JT

4. Buat table dengan nama trx_penjualan

```

DROP TABLE IF EXISTS trx_penjualan

CREATE TABLE trx_penjualan(
  kd_trx VARCHAR(10) NOT NULL PRIMARY KEY,
  tgl_trx DATE NOT NULL,
  kd_peternak VARCHAR(10) NOT NULL,
  total_trx INT(10) NOT NULL,
  stat_payment VARCHAR (10) NOT NULL,
  stat_lunas VARCHAR (15) NOT NULL)

ALTER TABLE trx_penjualan ADD FOREIGN KEY (kd_peternak) REFERENCES peternak (kd_peternak);

```

Penjelasan : Pernyataan ini digunakan untuk membuat tabel baru dengan nama "trx_penjualan". Tabel ini memiliki enam kolom yaitu "kd_trx", "tgl_trx", "kd_peternak", "total_trx", "stat_payment", dan "stat_lunas".

- Kolom "kd_trx" memiliki tipe data VARCHAR(10), yang berarti dapat menampung string dengan panjang maksimal 10 karakter. Kolom ini diberi batasan NOT NULL dan menjadi primary key tabel.
- Kolom "tgl_trx" memiliki tipe data DATE, yang berarti menampung tanggal. Kolom ini diberi batasan NOT NULL, yang berarti harus selalu memiliki nilai.
- Kolom "kd_peternak" memiliki tipe data VARCHAR(10) dan diberi batasan NOT NULL.

- Kolom "total_trx" memiliki tipe data INT(10), yang berarti dapat menampung nilai integer dengan panjang maksimal 10 digit. Kolom ini diberi batasan NOT NULL.
- Kolom "stat_payment" memiliki tipe data VARCHAR(10) dan diberi batasan NOT NULL.
- Kolom "stat_lunas" memiliki tipe data VARCHAR(15) dan diberi batasan NOT NULL.
- ALTER TABLE trx_penjualan ADD FOREIGN KEY (kd_peternak) REFERENCES peternak (kd_peternak); Query ini digunakan untuk menambahkan foreign key pada tabel "trx_penjualan". Foreign key ini menghubungkan kolom "kd_peternak" dalam tabel "trx_penjualan" dengan kolom "kd_peternak" dalam tabel "peternak".

- Tambahkan data ke table trx_penjualan

```
INSERT INTO trx_penjualan (kd_trx, tgl_trx, kd_peternak, total_trx, stat_payment, stat_lunas) VALUES
('JJ021', '2022-01-01', 'LL001', '45000', 'Cash', 'FULL PAYMENT'),
('JJ022', '2022-01-01', 'LL002', '36000', 'Cash', 'FULL PAYMENT'),
('JJ023', '2022-02-03', 'LL003', '60000', 'Kredit', 'NYICIL'),
('JJ024', '2022-03-01', 'LL004', '20000', 'Cash', 'FULL PAYMENT'),
('JJ025', '2022-03-17', 'LL005', '95000', 'Cash', 'NYICIL'),
('JJ026', '2022-03-28', 'LL006', '120000', 'Cash', 'FULL PAYMENT'),
('JJ027', '2022-04-14', 'LL007', '320000', 'Cash', 'FULL PAYMENT'),
('JJ028', '2022-04-21', 'LL008', '234000', 'Kredit', 'NYICIL'),
('JJ029', '2022-04-31', 'LL009', '420000', 'Cash', 'NYICIL'),
('JJ030', '2022-05-09', 'LL010', '280000', 'Kredit', 'FULL PAYMENT'),
('JJ031', '2022-05-18', 'LL001', '198000', 'Cash', 'FULL PAYMENT'),
('JJ032', '2022-06-21', 'LL002', '132000', 'Kredit', 'FULL PAYMENT'),
('JJ033', '2022-07-03', 'LL003', '315000', 'Cash', 'FULL PAYMENT'),
('JJ034', '2022-08-15', 'LL004', '60000', 'Kredit', 'NYICIL'),
('JJ035', '2022-09-02', 'LL005', '47500', 'Cash', 'FULL PAYMENT'),
('JJ036', '2022-09-30', 'LL006', '80000', 'Cash', 'FULL PAYMENT'),
('JJ037', '2022-10-06', 'LL007', '160000', 'Cash', 'NYICIL'),
('JJ038', '2022-11-13', 'LL008', '486000', 'Cash', 'FULL PAYMENT'),
('JJ039', '2022-12-08', 'LL009', '805000', 'Kredit', 'FULL PAYMENT'),
('JJ040', '2022-12-31', 'LL0010', '500000', 'Cash', 'FULL PAYMENT');

SELECT *FROM trx_penjualan;
```

Penjelasan : Query ini digunakan untuk memasukkan beberapa data ke dalam tabel "trx_penjualan". Setiap baris data memasukkan nilai ke kolom-kolom yang sesuai dalam urutan yang ditentukan.

- SELECT FROM trx_penjualan; query ini digunakan untuk mengambil semua data yang ada dalam tabel "trx_penjualan". "" digunakan untuk menampilkan seluruh data yang ada dalam tabel "trx_penjualan".

- Output

<input type="checkbox"/>	kd_trx	tgl_trx	kd_peternak	total_trx	stat_payment	stat_lunas
<input type="checkbox"/>	JJ021	2022-01-01	LL001	45000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ022	2022-01-01	LL002	36000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ023	2022-02-03	LL003	60000	Kredit	NYICIL
<input type="checkbox"/>	JJ024	2022-03-01	LL004	20000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ025	2022-03-17	LL005	95000	Cash	NYICIL
<input type="checkbox"/>	JJ026	2022-03-28	LL006	120000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ027	2022-04-14	LL007	320000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ028	2022-04-21	LL008	234000	Kredit	NYICIL
<input type="checkbox"/>	JJ029	0000-00-00	LL009	420000	Cash	NYICIL
<input type="checkbox"/>	JJ030	2022-05-09	LL010	280000	Kredit	FULL PAYMENT
<input type="checkbox"/>	JJ031	2022-05-18	LL001	198000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ032	2022-06-21	LL002	132000	Kredit	FULL PAYMENT
<input type="checkbox"/>	JJ033	2022-07-03	LL003	315000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ034	2022-08-15	LL004	60000	Kredit	NYICIL
<input type="checkbox"/>	JJ035	2022-09-02	LL005	47500	CASH	FULL PAYMENT
<input type="checkbox"/>	JJ036	2022-09-30	LL006	80000	CASH	FULL PAYMENT
<input type="checkbox"/>	JJ037	2022-10-06	LL007	160000	Cash	NYICIL
<input type="checkbox"/>	JJ038	2022-11-13	LL008	486000	Cash	FULL PAYMENT
<input type="checkbox"/>	JJ039	2022-12-08	LL009	805000	Kredit	FULL PAYMENT
<input type="checkbox"/>	JJ040	2022-12-31	LL0010	500000	Cash	FULL PAYMENT

5. Buat table dengan nama dtl_trx_penjualan

```

DROP TABLE IF EXISTS dtl_trx_penjualan;

CREATE TABLE dtl_trx_penjualan(
  kd_dt_trx VARCHAR(10) NOT NULL PRIMARY KEY,
  kd_trx VARCHAR(10) NOT NULL,
  kd_produk VARCHAR(10) NOT NULL,
  qty INT(10) NOT NULL,
  satuan VARCHAR(15) NOT NULL,
  jumlah INT(20) NOT NULL)

ALTER TABLE Dtl_trx_penjualan ADD FOREIGN KEY (kd_trx) REFERENCES trx_penjualan (kd_trx);
ALTER TABLE Dtl_trx_penjualan ADD FOREIGN KEY (kd_produk) REFERENCES produk (kd_produk);

```

Penjelasan : . Query ini digunakan untuk membuat tabel baru dengan nama "dtl_trx_penjualan". Tabel ini memiliki enam kolom yaitu "kd_dt_trx", "kd_trx", "kd_produk", "qty", "satuan", dan "jumlah".

- Kolom "kd_dt_trx" memiliki tipe data VARCHAR(10), yang berarti dapat menampung string dengan panjang maksimal 10 karakter. Kolom ini diberi batasan NOT NULL dan menjadi primary key tabel.
- Kolom "kd_trx" memiliki tipe data VARCHAR(10) dan diberi batasan NOT NULL.
- Kolom "kd_produk" memiliki tipe data VARCHAR(10) dan diberi batasan NOT NULL.
- Kolom "qty" memiliki tipe data INT(10), yang berarti dapat menampung nilai integer dengan panjang maksimal 10 digit. Kolom ini diberi batasan NOT NULL.
- Kolom "satuan" memiliki tipe data VARCHAR(15) dan diberi batasan NOT NULL.
- Kolom "jumlah" memiliki tipe data INT(20), yang berarti dapat menampung nilai integer dengan panjang maksimal 20 digit. Kolom ini diberi batasan NOT NULL.
- ALTER TABLE dtl_trx_penjualan ADD FOREIGN KEY (kd_trx) REFERENCES trx_penjualan (kd_trx); Query ini digunakan untuk menambahkan foreign key pada tabel "dtl_trx_penjualan". Foreign key ini menghubungkan kolom "kd_trx" dalam tabel "dtl_trx_penjualan" dengan kolom "kd_trx" dalam tabel "trx_penjualan".
- ALTER TABLE dtl_trx_penjualan ADD FOREIGN KEY (kd_produk) REFERENCES produk (kd_produk);
- Pernyataan ini digunakan untuk menambahkan foreign key pada tabel "dtl_trx_penjualan". Foreign key ini menghubungkan kolom "kd_produk" dalam tabel "dtl_trx_penjualan" dengan kolom "kd_produk" dalam tabel "produk".

- Tambahkan data ke dalam table trx_penjualan

```

125
126 INSERT INTO dtl_trx_penjualan (kd_dt_trx, kd_trx, kd_produk, qty, satuan, jumlah) VALUES
127 ('PJ01', 'JJ021', 'P011', 5, 'ekor', 45000),
128 ('PJ02', 'JJ022', 'P012', 3, 'pcs', 36000),
129 ('PJ03', 'JJ023', 'P013', 4, 'pcs', 60000),
130 ('PJ04', 'JJ024', 'P014', 2, 'pcs', 20000),
131 ('PJ05', 'JJ025', 'P015', 10, 'pcs', 95000),
132 ('PJ06', 'JJ026', 'P016', 15, 'pcs', 120000),
133 ('PJ07', 'JJ027', 'P017', 16, 'pcs', 320000),
134 ('PJ08', 'JJ028', 'P018', 13, 'pcs', 234000),
135 ('PJ09', 'JJ029', 'P019', 12, 'kg', 420000),
136 ('PJ10', 'JJ030', 'P020', 14, 'kg', 280000),
137 ('PJ11', 'JJ031', 'P011', 22, 'ekor', 198000),
138 ('PJ12', 'JJ032', 'P012', 11, 'pcs', 132000),
139 ('PJ13', 'JJ033', 'P013', 21, 'pcs', 315000),
140 ('PJ14', 'JJ034', 'P014', 6, 'pcs', 60000),
141 ('PJ15', 'JJ035', 'P015', 5, 'pcs', 47500),
142 ('PJ16', 'JJ036', 'P016', 10, 'pcs', 80000),
143 ('PJ17', 'JJ037', 'P017', 8, 'pcs', 160000),
144 ('PJ18', 'JJ038', 'P018', 27, 'PCS', 486000),
145 ('PJ19', 'JJ039', 'P019', 23, 'kg', 805000),
146 ('PJ20', 'JJ040', 'P020', 25, 'kg', 500000);
147
148 SELECT *FROM dtl_trx_penjualan
149

```

Penjelasan : Query ini digunakan untuk memasukkan beberapa baris data ke dalam tabel "dtl_trx_penjualan". Setiap baris data memasukkan nilai ke kolom-kolom yang sesuai dalam urutan yang ditentukan. Data diwakili oleh nilai-nilai dalam tanda kurung. Terakhir, yaitu query SELECT digunakan untuk menampilkan semua data yang ada dalam tabel "dtl_trx_penjualan", sehingga kita dapat melihat rincian transaksi penjualan yang telah dimasukkan sebelumnya.

- Output

<input type="checkbox"/> kd_dt_trx	kd_trx	kd_produk	qty	satuan	jumlah
<input type="checkbox"/> PJ01	JJ021	P011	5	ekor	45000
<input type="checkbox"/> PJ02	JJ022	P012	3	pcs	36000
<input type="checkbox"/> PJ03	JJ023	P013	4	pcs	60000
<input type="checkbox"/> PJ04	JJ024	P014	2	pcs	20000
<input type="checkbox"/> PJ05	JJ025	P015	10	pcs	95000
<input type="checkbox"/> PJ06	JJ026	P016	15	pcs	120000
<input type="checkbox"/> PJ07	JJ027	P017	16	pcs	320000
<input type="checkbox"/> PJ08	JJ028	P018	13	pcs	234000
<input type="checkbox"/> PJ09	JJ029	P019	12	kg	420000
<input type="checkbox"/> PJ10	JJ030	P020	14	kg	280000
<input type="checkbox"/> PJ11	JJ031	P011	22	ekor	198000
<input type="checkbox"/> PJ12	JJ032	P012	11	pcs	132000
<input type="checkbox"/> PJ13	JJ033	P013	21	pcs	315000
<input type="checkbox"/> PJ14	JJ034	P014	6	pcs	60000
<input type="checkbox"/> PJ15	JJ035	P015	5	pcs	47500
<input type="checkbox"/> PJ16	JJ036	P016	10	pcs	80000
<input type="checkbox"/> PJ17	JJ037	P017	8	pcs	160000
<input type="checkbox"/> PJ18	JJ038	P018	27	PCS	486000
<input type="checkbox"/> PJ19	JJ039	P019	23	kg	805000
<input type="checkbox"/> PJ20	JJ040	P020	25	kg	500000

Acti
Go to

6. Buat table dengan nama trx_pembayaran

```
DROP TABLE IF EXISTS trx_pembayaran
```

```
CREATE TABLE trx_pembayaran(  
  kd_bayar VARCHAR(10) NOT NULL PRIMARY KEY,  
  tgl_bayar DATE,  
  kd_trx VARCHAR(10) NOT NULL,  
  total_tagihan INT(10) NOT NULL,  
  jumlah_bayar INT(10) NOT NULL,  
  pembayaran_ke INT(10) NOT NULL)
```

```
ALTER TABLE trx_pembayaran ADD FOREIGN KEY (kd_trx) REFERENCES trx_penjualan (kd_trx);
```

Penjelasan : Pada query ini, digunakan untuk membuat tabel "trx_pembayaran" yang merupakan tabel untuk mencatat pembayaran atas transaksi penjualan. Tabel ini memiliki kolom-kolom seperti kode pembayaran (kd_bayar), tanggal pembayaran (tgl_bayar), kode transaksi (kd_trx), total tagihan (total_tagihan), jumlah pembayaran (jumlah_bayar), dan pembayaran ke (pembayaran_ke).

Selanjutnya, query ALTER TABLE digunakan untuk menambahkan foreign key yang menghubungkan kolom kd_trx pada tabel "trx_pembayaran" dengan kolom kd_trx pada tabel "trx_penjualan". Karena query ini bertujuan untuk menghubungkan data pembayaran dengan transaksi penjualan yang terkait.

- Tambahkan data kedalam table trx_pembayaran

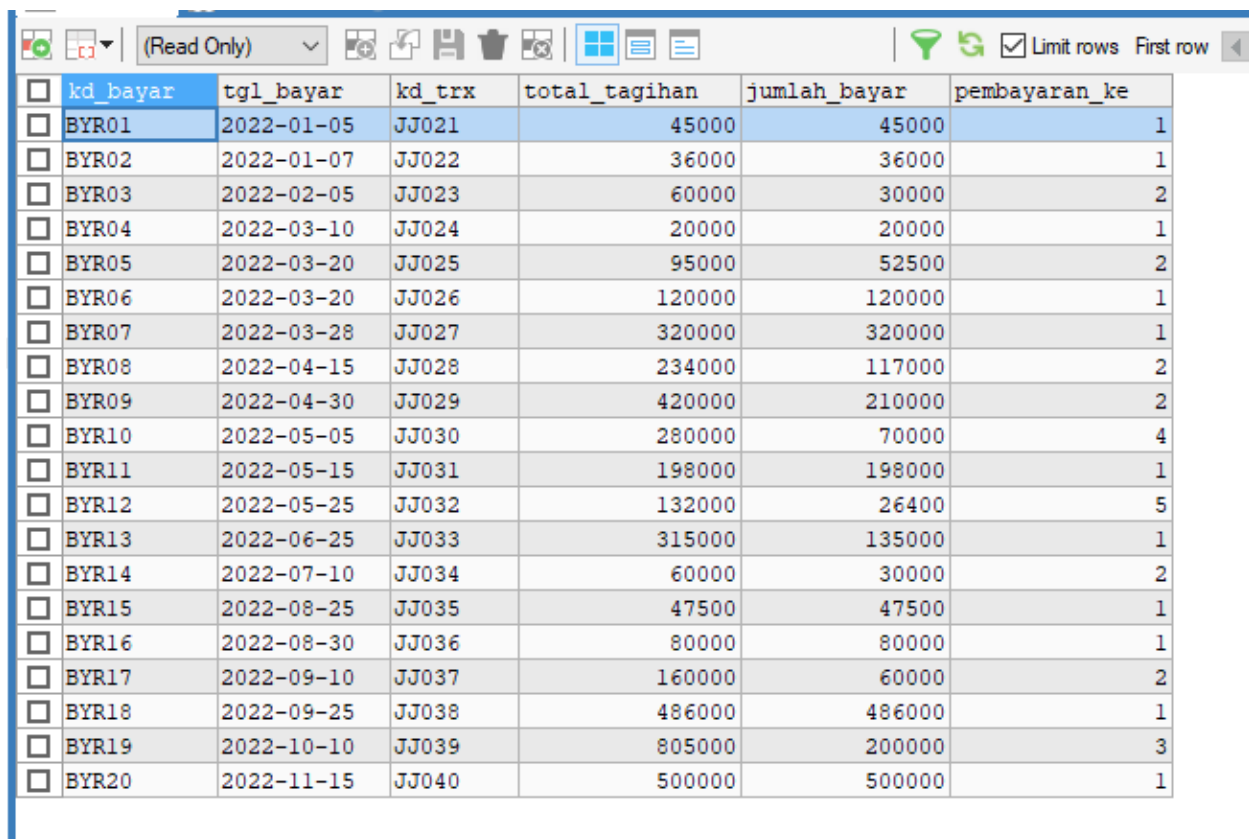
```
165 INSERT INTO trx_pembayaran(kd_bayar, tgl_bayar, kd_trx, total_tagihan, jumlah_bayar, pembayaran_ke) VALUES  
166 ('BYR01', '2022-01-05', 'JJ021', 45000, 45000, 1),  
167 ('BYR02', '2022-01-07', 'JJ022', 36000, 36000, 1),  
168 ('BYR03', '2022-02-05', 'JJ023', 60000, 30000, 2),  
169 ('BYR04', '2022-03-10', 'JJ024', 20000, 20000, 1),  
170 ('BYR05', '2022-03-20', 'JJ025', 95000, 52500, 2),  
171 ('BYR06', '2022-03-20', 'JJ026', 120000, 120000, 1),  
172 ('BYR07', '2022-03-28', 'JJ027', 320000, 320000, 1),  
173 ('BYR08', '2022-04-15', 'JJ028', 234000, 117000, 2),  
174 ('BYR09', '2022-04-30', 'JJ029', 420000, 210000, 2),  
175 ('BYR10', '2022-05-05', 'JJ030', 280000, 70000, 4),  
176 ('BYR11', '2022-05-15', 'JJ031', 198000, 198000, 1),  
177 ('BYR12', '2022-05-25', 'JJ032', 132000, 26400, 5),  
178 ('BYR13', '2022-06-25', 'JJ033', 315000, 135000, 1),  
179 ('BYR14', '2022-07-10', 'JJ034', 60000, 30000, 2),  
180 ('BYR15', '2022-08-25', 'JJ035', 47500, 47500, 1),  
181 ('BYR16', '2022-08-30', 'JJ036', 80000, 80000, 1),  
182 ('BYR17', '2022-09-10', 'JJ037', 160000, 60000, 2),  
183 ('BYR18', '2022-09-25', 'JJ038', 486000, 486000, 1),  
184 ('BYR19', '2022-10-10', 'JJ039', 805000, 200000, 3),  
185 ('BYR20', '2022-11-15', 'JJ040', 500000, 500000, 1);  
186  
187  
188 SELECT *FROM trx_pembayaran;  
189 |
```

Penjelasan : Kemudian, query INSERT digunakan untuk memasukkan beberapa data pembayaran ke dalam tabel "trx_pembayaran". Setiap baris data mencakup nilai-nilai untuk

setiap kolom yang sesuai. Misalnya, kd_bayar, tgl_bayar, kd_trx, total_tagihan, jumlah_bayar, dan pembayaran_ke.

Terakhir, ada query SELECT digunakan untuk mengambil dan menampilkan semua data yang ada dalam tabel "trx_pembayaran". Dengan pernyataan SELECT * FROM trx_pembayaran, kita dapat melihat seluruh data yang telah dimasukkan sebelumnya ke dalam tabel "trx_pembayaran".

- Output



The screenshot shows a database application interface. At the top, there is a toolbar with various icons for file operations and a status bar indicating "(Read Only)". Below the toolbar, a table is displayed with the following columns: kd_bayar, tgl_bayar, kd_trx, total_tagihan, jumlah_bayar, and pembayaran_ke. The table contains 20 rows of data, each representing a payment transaction. The first row is highlighted in blue.

kd_bayar	tgl_bayar	kd_trx	total_tagihan	jumlah_bayar	pembayaran_ke
BYR01	2022-01-05	JJ021	45000	45000	1
BYR02	2022-01-07	JJ022	36000	36000	1
BYR03	2022-02-05	JJ023	60000	30000	2
BYR04	2022-03-10	JJ024	20000	20000	1
BYR05	2022-03-20	JJ025	95000	52500	2
BYR06	2022-03-20	JJ026	120000	120000	1
BYR07	2022-03-28	JJ027	320000	320000	1
BYR08	2022-04-15	JJ028	234000	117000	2
BYR09	2022-04-30	JJ029	420000	210000	2
BYR10	2022-05-05	JJ030	280000	70000	4
BYR11	2022-05-15	JJ031	198000	198000	1
BYR12	2022-05-25	JJ032	132000	26400	5
BYR13	2022-06-25	JJ033	315000	135000	1
BYR14	2022-07-10	JJ034	60000	30000	2
BYR15	2022-08-25	JJ035	47500	47500	1
BYR16	2022-08-30	JJ036	80000	80000	1
BYR17	2022-09-10	JJ037	160000	60000	2
BYR18	2022-09-25	JJ038	486000	486000	1
BYR19	2022-10-10	JJ039	805000	200000	3
BYR20	2022-11-15	JJ040	500000	500000	1

7. Buat table dengan nama trx_pembelian

```
DROP TABLE IF EXISTS trx_pembelian
```

```
CREATE TABLE trx_pembelian(  
  kd_beli VARCHAR(10) NOT NULL PRIMARY KEY,  
  tgl_beli DATE,  
  kd_peternak VARCHAR(10) NOT NULL,  
  satuan VARCHAR(15) NOT NULL,  
  total_trx INT(10) NOT NULL,  
  beli_daging VARCHAR(10) NOT NULL,  
  beli_telur VARCHAR(10) NOT NULL,  
  stat_payment_beli VARCHAR(20) NOT NULL)
```

```
ALTER TABLE trx_pembelian ADD FOREIGN KEY (kd_peternak) REFERENCES peternak (kd_peternak);
```

- Tambahkan data kedalam table trx_pembelian

```
205 INSERT INTO trx_pembelian  
206 (kd_beli, tgl_beli, kd_peternak, satuan, total_trx, beli_daging, beli_telur, stat_payment_beli) VALUES  
207 ('BLI01', '2022-02-25', 'LL001', 'KG', 500000, 'tidak', 'ya', 'KREDIT'),  
208 ('BLI02', '2022-03-10', 'LL002', 'KG', 125000, 'ya', 'tidak', 'KREDIT'),  
209 ('BLI03', '2022-04-25', 'LL003', 'KG', 750000, 'ya', 'ya', 'KREDIT'),  
210 ('BLI04', '2022-05-13', 'LL004', 'KG', 1750000, 'ya', 'tidak', 'KREDIT'),  
211 ('BLI05', '2022-07-14', 'LL005', 'KG', 500000, 'tidak', 'ya', 'KREDIT'),  
212 ('BLI06', '2022-08-31', 'LL006', 'KG', 360000, 'ya', 'tidak', 'KREDIT'),  
213 ('BLI07', '2022-09-20', 'LL007', 'KG', 900000, 'tidak', 'ya', 'KREDIT'),  
214 ('BLI08', '2022-10-30', 'LL008', 'KG', 1000000, 'tidak', 'ya', 'KREDIT'),  
215 ('BLI09', '2022-04-31', 'LL009', 'KG', 600000, 'ya', 'ya', 'KREDIT'),  
216 ('BLI10', '2022-03-31', 'LL010', 'KG', 400000, 'ya', 'ya', 'KREDIT'),  
217 ('BLI11', '2022-11-31', 'LL001', 'KG', 500000, 'ya', 'ya', 'KREDIT'),  
218 ('BLI12', '2022-10-31', 'LL002', 'KG', 200000, 'ya', 'ya', 'KREDIT'),  
219 ('BLI13', '2022-09-31', 'LL003', 'KG', 450000, 'ya', 'ya', 'KREDIT'),  
220 ('BLI14', '2022-08-31', 'LL004', 'KG', 40000, 'tidak', 'ya', 'KREDIT'),  
221 ('BLI15', '2022-07-31', 'LL005', 'KG', 60000, 'tidak', 'ya', 'KREDIT'),  
222 ('BLI16', '2022-06-31', 'LL006', 'KG', 56000, 'tidak', 'ya', 'KREDIT'),  
223 ('BLI17', '2022-05-31', 'LL007', 'KG', 80000, 'tidak', 'ya', 'KREDIT'),  
224 ('BLI18', '2022-04-31', 'LL008', 'KG', 600000, 'ya', 'ya', 'KREDIT'),  
225 ('BLI19', '2022-03-31', 'LL009', 'KG', 100000, 'ya', 'ya', 'KREDIT'),  
226 ('BLI20', '2022-03-31', 'LL010', 'KG', 100000, 'ya', 'ya', 'KREDIT');  
227  
228 SELECT * FROM trx_pembelian  
229
```

Activate Windows

- Output

									Limit rows	First row	<input type="text" value="0"/>			# of rows	1000
<input type="checkbox"/>	kd_beli	tgl_beli	kd_peternak	satuan	total_trx	beli_daging	beli_telur	stat_payment_beli							
<input type="checkbox"/>	BLI01	2022-02-25	LL001	KG	500000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI02	2022-03-10	LL002	KG	125000	ya	tidak	KREDIT							
<input type="checkbox"/>	BLI03	2022-04-25	LL003	KG	750000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI04	2022-05-13	LL004	KG	1750000	ya	tidak	KREDIT							
<input type="checkbox"/>	BLI05	2022-07-14	LL005	KG	500000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI06	2022-08-31	LL006	KG	360000	ya	tidak	KREDIT							
<input type="checkbox"/>	BLI07	2022-09-20	LL007	KG	900000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI08	2022-10-30	LL008	KG	1000000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI09	0000-00-00	LL009	KG	600000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI10	2022-03-31	LL010	KG	400000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI11	0000-00-00	LL001	KG	500000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI12	2022-10-31	LL002	KG	200000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI13	0000-00-00	LL003	KG	450000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI14	2022-08-31	LL004	KG	40000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI15	2022-07-31	LL005	KG	60000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI16	0000-00-00	LL006	KG	56000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI17	2022-05-31	LL007	KG	80000	tidak	ya	KREDIT							
<input type="checkbox"/>	BLI18	0000-00-00	LL008	KG	600000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI19	2022-03-31	LL009	KG	100000	ya	ya	KREDIT							
<input type="checkbox"/>	BLI20	2022-03-31	LL010	KG	100000	ya	ya	KREDIT							

TUGAS

- a. Simulasikan data yang di transaksikan mulai dari 1 januari 2022 s.d 31 Desember 2022.

```
SELECT *  
FROM trx_pembayaran  
WHERE tgl_bayar BETWEEN '2022-01-01' AND '2022-12-31';  
|
```

Penjelasan : Query ini untuk menampilkan data transaksi dari 1 januari sampai dengan 31 Desember

- Output

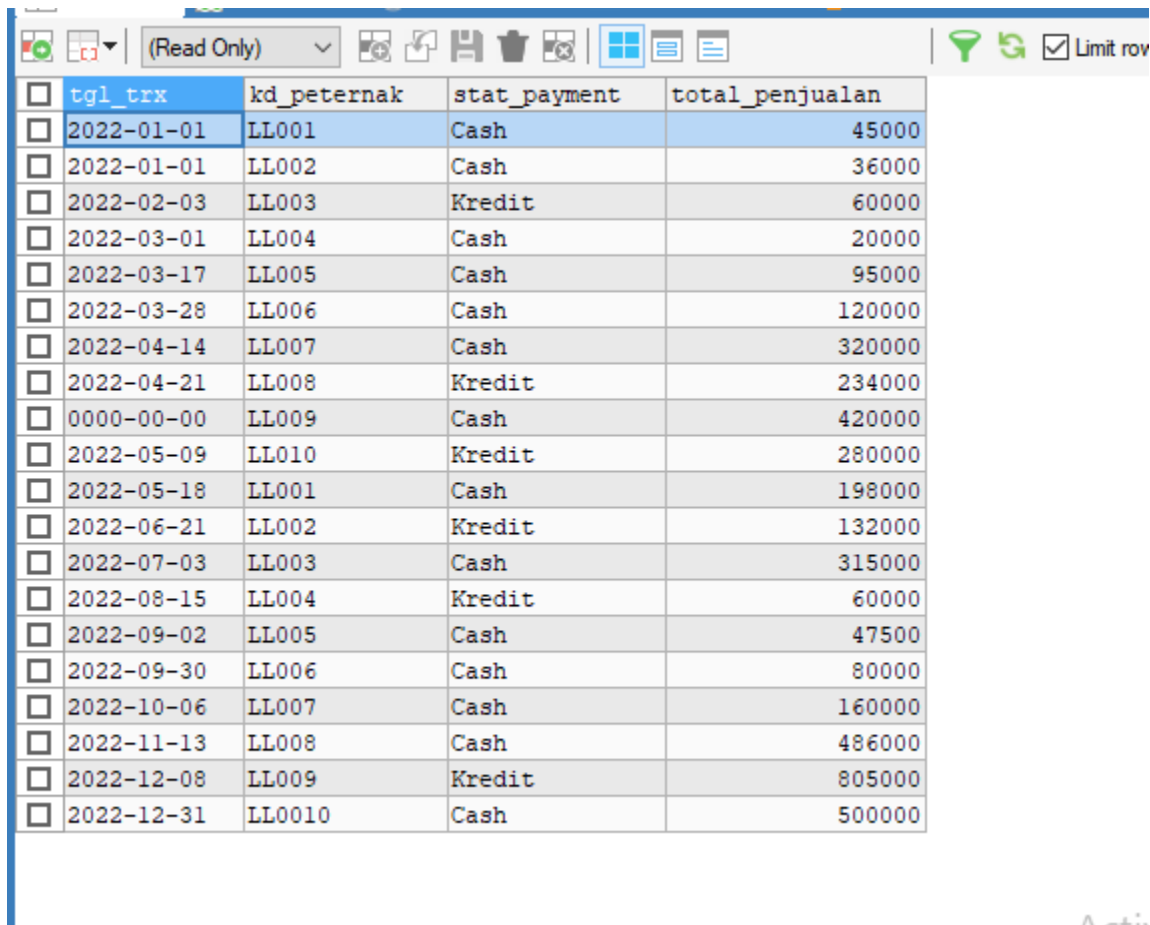
(Read Only)						Limit rows	First row	0
<input type="checkbox"/> kd_bayar	tgl_bayar	kd_trx	total_tagihan	jumlah_bayar	pembayaran_ke			
<input type="checkbox"/> BYR01	2022-01-05	JJ021	45000	45000	1			
<input type="checkbox"/> BYR02	2022-01-07	JJ022	36000	36000	1			
<input type="checkbox"/> BYR03	2022-02-05	JJ023	60000	30000	2			
<input type="checkbox"/> BYR04	2022-03-10	JJ024	20000	20000	1			
<input type="checkbox"/> BYR05	2022-03-20	JJ025	95000	52500	2			
<input type="checkbox"/> BYR06	2022-03-20	JJ026	120000	120000	1			
<input type="checkbox"/> BYR07	2022-03-28	JJ027	320000	320000	1			
<input type="checkbox"/> BYR08	2022-04-15	JJ028	234000	117000	2			
<input type="checkbox"/> BYR09	2022-04-30	JJ029	420000	210000	2			
<input type="checkbox"/> BYR10	2022-05-05	JJ030	280000	70000	4			
<input type="checkbox"/> BYR11	2022-05-15	JJ031	198000	198000	1			
<input type="checkbox"/> BYR12	2022-05-25	JJ032	132000	26400	5			
<input type="checkbox"/> BYR13	2022-06-25	JJ033	315000	135000	1			
<input type="checkbox"/> BYR14	2022-07-10	JJ034	60000	30000	2			
<input type="checkbox"/> BYR15	2022-08-25	JJ035	47500	47500	1			
<input type="checkbox"/> BYR16	2022-08-30	JJ036	80000	80000	1			
<input type="checkbox"/> BYR17	2022-09-10	JJ037	160000	60000	2			
<input type="checkbox"/> BYR18	2022-09-25	JJ038	486000	486000	1			
<input type="checkbox"/> BYR19	2022-10-10	JJ039	805000	200000	3			
<input type="checkbox"/> BYR20	2022-11-15	JJ040	500000	500000	1			

- b. Laporan penjualan harian baik secara cash atau kredit

```
SELECT tgl_trx,kd_peternak,stat_payment, total_trx AS total_penjualan  
FROM trx_penjualan  
GROUP BY kd_trx;
```

Penjelasan : Pada query ini, dilakukan pemilihan kolom tgl_trx (tanggal transaksi), kd_peternak, stat_payment (status pembayaran), dan total_trx (total penjualan) dari tabel Trx_penjualan. Kemudian, hasilnya dikelompokkan berdasarkan kd_trx. Query ini menghasilkan laporan penjualan harian yang mencakup transaksi baik dengan metode pembayaran cash maupun kredit.

- Output



tgl_trx	kd_peternak	stat_payment	total_penjualan
2022-01-01	LL001	Cash	45000
2022-01-01	LL002	Cash	36000
2022-02-03	LL003	Kredit	60000
2022-03-01	LL004	Cash	20000
2022-03-17	LL005	Cash	95000
2022-03-28	LL006	Cash	120000
2022-04-14	LL007	Cash	320000
2022-04-21	LL008	Kredit	234000
0000-00-00	LL009	Cash	420000
2022-05-09	LL010	Kredit	280000
2022-05-18	LL001	Cash	198000
2022-06-21	LL002	Kredit	132000
2022-07-03	LL003	Cash	315000
2022-08-15	LL004	Kredit	60000
2022-09-02	LL005	Cash	47500
2022-09-30	LL006	Cash	80000
2022-10-06	LL007	Cash	160000
2022-11-13	LL008	Cash	486000
2022-12-08	LL009	Kredit	805000
2022-12-31	LL0010	Cash	500000

c. Laporan pembelian harian baik secara cash atau kredit

```
SELECT tgl_beli, stat_payment_beli, SUM(total_trx) AS total_pembelian
FROM trx_pembelian
GROUP BY tgl_beli, stat_payment_beli;
```

Penjelasan: Query ini menampilkan laporan pembelian harian yang mencakup transaksi baik dengan metode pembayaran cash maupun kredit. Kolom yang diambil adalah tgl_trx (tanggal transaksi), stat_payment (status pembayaran), dan total_pembelian (total pembelian). Data

diambil dengan melakukan join antara tabel Dtl_trx_penjualan dan Trx_penjualan berdasarkan kd_trx. Hasilnya dikelompokkan berdasarkan tgl_trx dan stat_payment, dan total jumlah pembelian dihitung dengan menggunakan fungsi SUM().

- Output

<input type="checkbox"/>	tgl_beli	stat_payment_beli	total_pembelian
<input type="checkbox"/>	0000-00-00	KREDIT	2206000
<input type="checkbox"/>	2022-02-25	KREDIT	500000
<input type="checkbox"/>	2022-03-10	KREDIT	125000
<input type="checkbox"/>	2022-03-31	KREDIT	600000
<input type="checkbox"/>	2022-04-25	KREDIT	750000
<input type="checkbox"/>	2022-05-13	KREDIT	1750000
<input type="checkbox"/>	2022-05-31	KREDIT	80000
<input type="checkbox"/>	2022-07-14	KREDIT	500000
<input type="checkbox"/>	2022-07-31	KREDIT	60000
<input type="checkbox"/>	2022-08-31	KREDIT	400000
<input type="checkbox"/>	2022-09-20	KREDIT	900000
<input type="checkbox"/>	2022-10-30	KREDIT	1000000
<input type="checkbox"/>	2022-10-31	KREDIT	200000

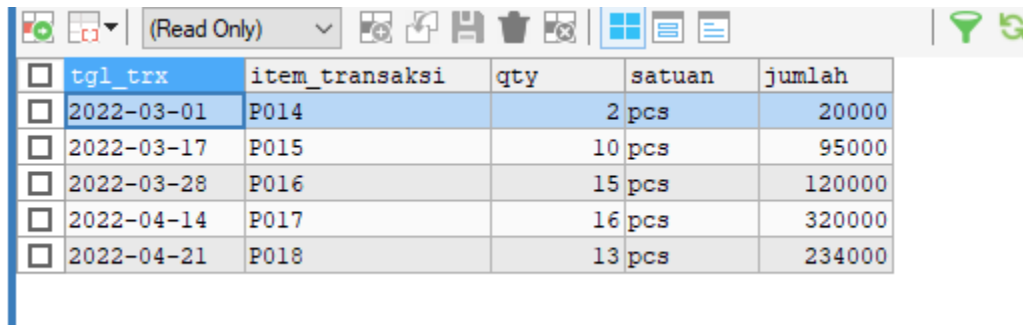
- d. Laporan rekap transaksi per peternak dari tanggal 1 maret s.d 31 april dengan kolom: tgl_trx, item_transaksi, qty, satuan, jumlah

```
SELECT T.tgl_trx, P.kd_produk AS item_transaksi, DT.qty, DT.satuan, DT.jumlah
FROM trx_penjualan T
JOIN Dtl_trx_penjualan DT ON T.kd_trx = DT.kd_trx
JOIN produk P ON DT.kd_produk = P.kd_produk
WHERE T.tgl_trx BETWEEN '2022-03-01' AND '2022-04-30';
```

Penjelasan : Query ini akan menampilkan laporan rekap transaksi per peternak dalam rentang tanggal 1 Maret hingga 31 April. Kolom yang diambil adalah tgl_trx (tanggal transaksi), item_transaksi (kd_produk), qty (kuantitas), satuan, dan jumlah (total harga). Data diambil

dengan melakukan join antara tabel `trx_penjualan`, `dtl_trx_penjualan`, dan `produk` berdasarkan `kd_trx` dan `kd_produk`. Data yang ditampilkan hanya yang memiliki `tgl_trx` antara 1 Maret hingga 30 April.

- Output



<input type="checkbox"/>	tgl_trx	item_transaksi	qty	satuan	jumlah
<input type="checkbox"/>	2022-03-01	P014	2	pcs	20000
<input type="checkbox"/>	2022-03-17	P015	10	pcs	95000
<input type="checkbox"/>	2022-03-28	P016	15	pcs	120000
<input type="checkbox"/>	2022-04-14	P017	16	pcs	320000
<input type="checkbox"/>	2022-04-21	P018	13	pcs	234000

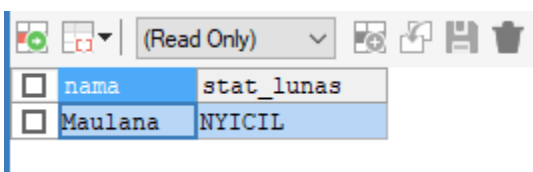
e. Menampilkan data peternak yang masih memiliki tanggungan pada bulan Maret

```
SELECT T.tgl_trx, P.kd_produk AS item_transaksi, DT.qty, DT.satuan, DT.jumlah
FROM trx_penjualan T
JOIN Dtl_trx_penjualan DT ON T.kd_trx = DT.kd_trx
JOIN produk P ON DT.kd_produk = P.kd_produk
WHERE T.tgl_trx BETWEEN '2022-03-01' AND '2022-04-30';
```

Penjelasan :

Query ini akan menampilkan data peternak yang masih memiliki tanggungan (status pembayaran 'NYICIL') pada bulan Maret. Kolom yang ditampilkan adalah nama peternak (`p.nama`) dan `stat_lunas` (status pelunasan) dari tabel peternak (`p`) dan `trx_penjualan` (`tp`). Data diambil dengan melakukan join antara tabel peternak dan `trx_penjualan` berdasarkan `kd_peternak`. Hanya data dengan `tgl_trx` antara 1 Maret hingga 31 Maret dan `stat_lunas` 'NYICIL' yang akan ditampilkan.

- Output



<input type="checkbox"/>	nama	stat_lunas
<input type="checkbox"/>	Maulana	NYICIL

- f. Menampilkan data transaksi yang sudah dilakukan oleh seorang peternak selama kurun waktu bulan Mei

```
|  
SELECT tp.kd_trx, tp.tgl_trx, p.nama, dtp.qty, dtp.satuan, dtp.jumlah  
FROM Trx_penjualan tp  
JOIN Peternak p ON tp.kd_peternak = p.kd_peternak  
JOIN Dtl_trx_penjualan dtp ON tp.kd_trx = dtp.kd_trx  
WHERE tp.tgl_trx BETWEEN '2022-05-01' AND '2022-05-31';
```

Penjelasan :

Query ini akan menampilkan data transaksi yang telah dilakukan oleh seorang peternak selama bulan Mei. Kolom yang diambil adalah kd_trx, tgl_trx, nama peternak (p.nama), qty (kuantitas), satuan, dan jumlah dari tabel trx_penjualan (tp), peternak (p), dan dtl_trx_penjualan (dtp). Data diambil dengan melakukan join antara ketiga tabel tersebut berdasarkan kd_peternak dan kd_trx. Hanya data dengan tgl_trx antara 1 Mei hingga 31 Mei yang akan ditampilkan.

- Output

<input type="checkbox"/>	kd_trx	tgl_trx	nama	qty	satuan	jumlah
<input type="checkbox"/>	JJ030	2022-05-09	Siti	14	kg	280000
<input type="checkbox"/>	JJ031	2022-05-18	Hety	22	ekor	198000

PEMBERIAN HAK AKSES KEPADA ANGGOTA KELOMPOK

MASUK KE ROOT UNTUK MEMBERIKAN HAK AKSES

```
C:\xampp\mysql\bin>mysql -u root
```

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 17

Server version: 10.4.25-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MEMBERIKAN HAK AKSES HILMI

```
MariaDB [(none)]> create user 'hilmi';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> GRANT ALL ON uaskoperasi.Trx_penjualan TO 'hilmi'@'localhost';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> GRANT ALL ON uaskoperasi.Dtl_Trx_penjualan TO 'hilmi'@'localhost';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> GRANT ALL ON uaskoperasi.Trx_pembayaran TO 'hilmi'@'localhost';
```

Query OK, 0 rows affected (0.001 sec)

```
=====
=====
```

➤ PEMBERIAN HAK AKSES PADA HETY

```
MariaDB [(none)]> create user 'hety';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> GRANT INSERT, DELETE ON uaskoperasi.trx_pembelian TO  
'hety'@'localhost';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> GRANT INSERT, DELETE ON uaskoperasi.Dtl_trx_penjualan TO 'hety'@'localhost';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
MariaDB [(none)]> GRANT INSERT, DELETE ON uaskoperasi.trx_penjualan TO 'hety'@'localhost';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
=====
```

MEMBERIKAN HAK AKSES KE YUDHA

```
MariaDB [(none)]> create user 'yudha';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
MariaDB [(none)]> GRANT INSERT, SELECT ON uaskoperasi.trx_penjualan TO 'yudha'@'localhost';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
MariaDB [(none)]> GRANT INSERT, SELECT ON uaskoperasi.Dtl_trx_penjualan TO 'yudha'@'localhost';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
MariaDB [(none)]> GRANT INSERT, SELECT ON uaskoperasi.trx_pembelian TO 'yudha'@'localhost';
```

```
Query OK, 0 rows affected (0.001 sec)
```

BACKUP SQL

Backup dilakukan dengan memasukkan perintah backup yang dimasukkan ke backup.bat yang nantinya akan di trigger menggunakan task scheduler agar otomatis terbackup pada pukul 12.00 dan 16.00 setiap harinya. Berikut isi perintah pada backup.bat

SCRIPT UNTUK MEMBACKUP DATA BASE

```
cd C:\xampp\mysql\bin
```

```
mysqldump -u root uaskoperasi > C:\backup\uasbackup_%date:~10,2%%date:~7,2%%date:~4,4%%time:~0,2%-%time:~3,2%-%time:~6,2%.sql
```

BACKUPAN DATA SQL JAM 12.00 DAN JAM 18.00

