

# LAPORAN PROJEK AKHIR

## MATA KULIAH ADMINISTRASI BASIS DATA

Kelompok ke- 8

Paralel C

Anggota Kelompok:

| NO | NPM/NIM/NRP | NAMA MAHASISWA          | KEGIATAN YANG DIKERJAKAN       |
|----|-------------|-------------------------|--------------------------------|
| 1. | 21082010092 | Byanca Rebecca Sitompul | Skema CDM dan PDM              |
| 2. | 21082010092 | Byanca Rebecca Sitompul | Create Database & Table        |
| 3. | 21082010104 | Windy Fadhilah Susanti  | Create User & Create Role User |
| 4. | 21082010111 | Afif Ghani Zahran       | Export & Import database       |

### 1. Skema CDM dan PDM

Pada pembuatan *database* penjualan dan pembelian pada toko, diasumsikan bahwa pada toko roti terdiri dari 2 aktivitas bisnis yakni pembelian bahan baku yang melibatkan *supplier*, dan proses transaksi penjualan roti yang melibatkan pelanggan. Pembuatan *database* dilakukan menggunakan *software* power designer dengan mendesain CDM terlebih dahulu, kemudian diagram CDM di *generate* ke dalam PDM. Berikut merupakan tabel-tabel yang akan kami rancang untuk pembuatan database **dbroti** :

- **Tabel Roti**

Pada tabel Roti terdiri dari atribut id roti (varchar(10)) sebagai *primary key*, id transaksi (varchar(10)), id kategori (varchar(10)), nama roti (varchar(20)), dan harga roti (int).

- **Tabel Memasok**

Pada tabel Memasok terdapat 2 atribut yaitu id supplier (varchar(10)) dan id pembelian (varchar(10)). Kedua atribut tersebut berperan sebagai FK dan PK.

- **Tabel Supplier**

Pada tabel supplier terdapat 4 atribut yaitu, id supplier, nama supplier, alamat supplier yang bertipe data varchar, dan no telepon supplier yang bertipe data integer. Tabel supplier memiliki *primary key* yaitu id supplier.

- **Tabel Kategori**

Pada tabel kategori terdapat 2 atribut yaitu, id kategori dan jenis roti yang bertipe data varchar. Tabel kategori memiliki *primary key* yaitu id kategori.

- **Tabel Stok**

Pada tabel stok terdapat 2 atribut yaitu, id stok yang bertipe data varchar, jumlah stok yang bertipe data integer, dan 1 atribut tambahan yaitu, id roti yang bertipe data varchar. Tabel stok memiliki *primary key* yaitu id stok dan *foreign key* yaitu id roti. *Foreign key* yang didapatkan merupakan hasil relasi **One to Many** antara tabel roti dengan tabel stok. Berikut gambar dari relasinya :

- **Tabel Pembelian Bahan Header**

Pada tabel pembelian bahan header terdapat 2 atribut, yaitu id pembelian yang bertipe data *varchar* dan tanggal pembelian yang bertipe data *date & time*. Tabel pembelian bahan header memiliki *primary key* yaitu id pembelian.

- **Tabel Transaksi Header**

Pada tabel transaksi header terdapat 3 atribut, yaitu id transaksi yang bertipe data *varchar* sebagai *primary key*, id pelanggan yang bertipe data *varchar* sebagai *foreign key* dan id roti yang bertipe data *varchar* sebagai *foreign key*. Tabel transaksi header diasumsikan mempunyai relasi **One to Many** dengan tabel pelanggan dikarenakan satu pelanggan dapat melakukan transaksi, akan tetapi setiap transaksi hanya dapat dilakukan oleh pelanggan. Maka dari itu, tabel transaksi header memiliki *foreign key* pada id pelanggan untuk menunjukkan relasi antara kedua tabel tersebut. Selain itu, tabel transaksi header juga diasumsikan mempunyai relasi **Many to One** dengan tabel roti. Maka dari itu, tabel transaksi header memiliki *foreign key* pada id pelanggan untuk menunjukkan relasi antara kedua tabel tersebut. Berikut gambar dari relasinya :

- **Tabel Pembelian Bahan Detail**

Pada tabel pembelian header terdapat 6 atribut, yaitu id detail pembelian yang bertipe data *varchar* sebagai *primary key*, id pembelian yang bertipe data *varchar* sebagai *foreign key*, nama bahan yang bertipe data *varchar*, jumlah bahan yang bertipe data *integer*, harga pembelian yang bertipe data *integer*, dan total harga pembelian yang bertipe data *integer*. Tabel pembelian bahan detail diasumsikan mempunyai relasi **Many to One** dengan tabel pembelian bahan header dikarenakan setiap pembelian yang disimpan pada tabel pembelian bahan header akan terdapat beberapa baris pada tabel pembelian bahan detail karena pada pembelian bahan baku, toko roti dapat membeli 1 bahan atau lebih. Maka dari itu, tabel pembelian bahan detail memiliki *foreign key* pada id pembelian untuk menunjukkan relasi antara kedua tabel tersebut. Berikut gambar dari relasinya :

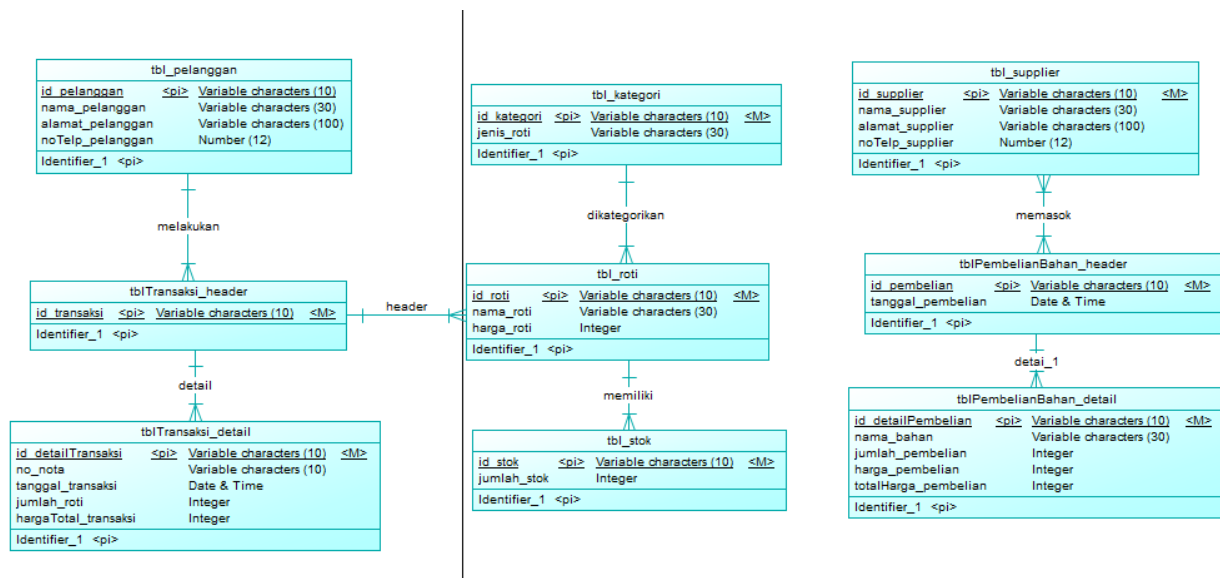
- **Tabel Pelanggan**

Pada tabel pelanggan terdapat 4 atribut, yaitu id\_pelanggan yang bertipe data *varchar* sebagai *primary key*, nama pelanggan yang bertipe data *varchar*, alamat pelanggan yang bertipe data *varchar*, dan nomor telepon pelanggan yang bertipe data *integer*.

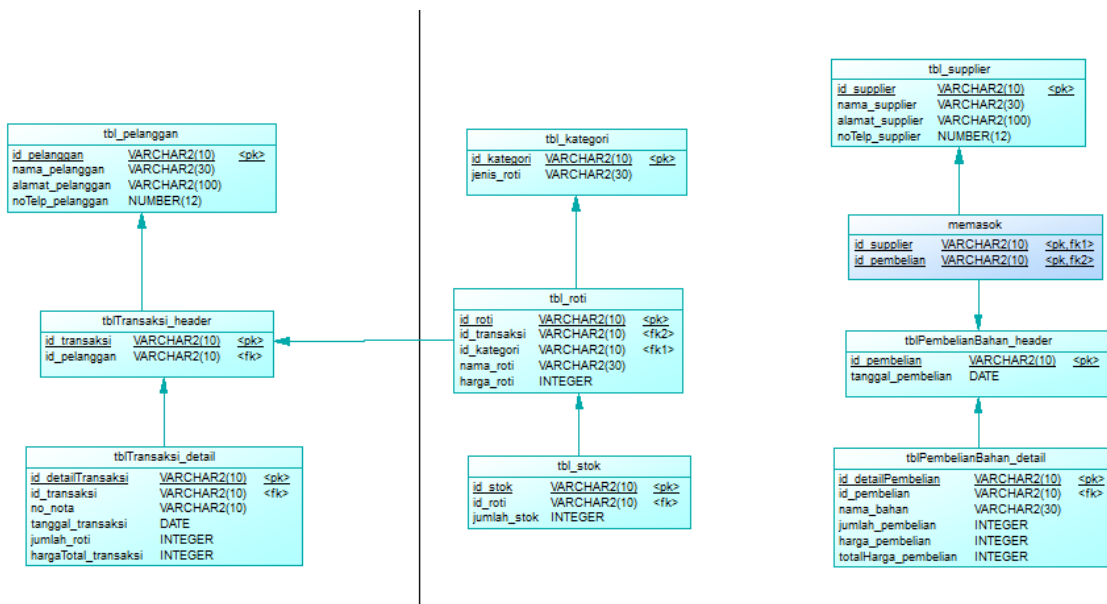
- **Tabel Transaksi Detail**

Pada tabel transaksi detail terdapat 6 atribut, yaitu id detail transaksi yang bertipe data *varchar* sebagai *primary key*, id\_transaksi yang bertipe data *varchar* sebagai *foreign key*, nomor nota yang bertipe data *varchar*, tanggal transaksi yang bertipe data *date & time*, jumlah roti yang bertipe data *integer*, dan harga total transaksi yang bertipe data *integer*. Tabel transaksi detail diasumsikan mempunyai relasi **Many to One** dengan tabel transaksi header dikarenakan setiap transaksi yang disimpan pada tabel transaksi header akan terdapat beberapa baris pada tabel detail transaksi karena pada satu kali transaksi seorang pelanggan dapat membeli 1 roti atau lebih. Maka dari itu, tabel transaksi detail memiliki *foreign key* pada id transaksi untuk menunjukkan relasi antara kedua tabel tersebut.

Berikut merupakan hasil desain **CDM** yang telah kami rancang :

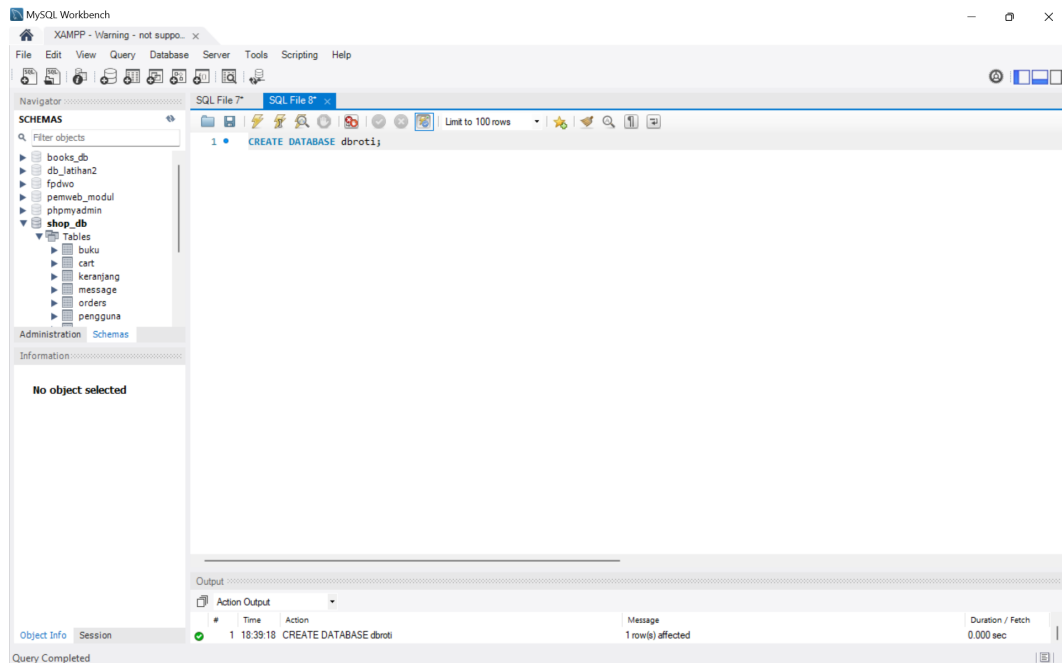


Berikut merupakan hasil desain PDM yang telah kami rancang :



## 2. Penjelasan Create Database dan Create Table (Min. 3 table)

Pada tahap ini yang pertama dilakukan yaitu pembuatan schema database dengan query dan hasil output yang dipaparkan pada gambar dibawah ini:



Setelah berhasil create database, jadikan schema database **dbroti** menjadi **set as default schema** untuk membuat tabel yang dibutuhkan pada database **dbroti** yang telah ditentukan melalui pembuatan CDM dan PDM. Berikut merupakan query untuk create table pada database:

```
CREATE TABLE `memasok` (
  `ID_SUPPLIER` varchar(10) NOT NULL,
  `ID_PEMBELIAN` varchar(10) NOT NULL
);

CREATE TABLE `tblpembelianbahan_detail` (
  `ID_DETAILPEMBELIAN` varchar(10) NOT NULL,
  `ID_PEMBELIAN` varchar(10) NOT NULL,
  `NAMA_BAHAN` varchar(30) DEFAULT NULL,
  `JUMLAH_PEMBELIAN` int(11) DEFAULT NULL,
  `HARGA_PEMBELIAN` int(11) DEFAULT NULL,
  `TOTALHARGA_PEMBELIAN` int(11) DEFAULT NULL
);

CREATE TABLE `tblpembelianbahan_header` (
  `ID_PEMBELIAN` varchar(10) NOT NULL,
  `TANGGAL_PEMBELIAN` datetime DEFAULT NULL
);

CREATE TABLE `tbltransaksi_detail` (
  `ID_DETAILTRANSAKSI` varchar(10) NOT NULL,
  `ID_TRANSAKSI` varchar(10) NOT NULL,
  `NO_NOTA` varchar(10) DEFAULT NULL,
  `TANGGAL_TRANSAKSI` datetime DEFAULT NULL,
  `JUMLAH_ROTI` int(11) DEFAULT NULL,
  `HARGATOTAL_TRANSAKSI` int(11) DEFAULT NULL
);
```

```

);
CREATE TABLE `tbltransaksi_header` (
  `ID_TRANSAKSI` varchar(10) NOT NULL,
  `ID_ROTI` varchar(10) NOT NULL,
  `ID_PELANGGAN` varchar(10) NOT NULL
);

CREATE TABLE `tbl_kategori` (
  `ID_KATEGORI` varchar(10) NOT NULL,
  `JENIS_ROTI` varchar(30) DEFAULT NULL
);

CREATE TABLE `tbl_pelanggan` (
  `ID_PELANGGAN` varchar(10) NOT NULL,
  `NAMA_PELANGGAN` varchar(30) DEFAULT NULL,
  `ALAMAT_PELANGGAN` varchar(100) DEFAULT NULL,
  `NOTELP_PELANGGAN` decimal(12,0) DEFAULT NULL
);

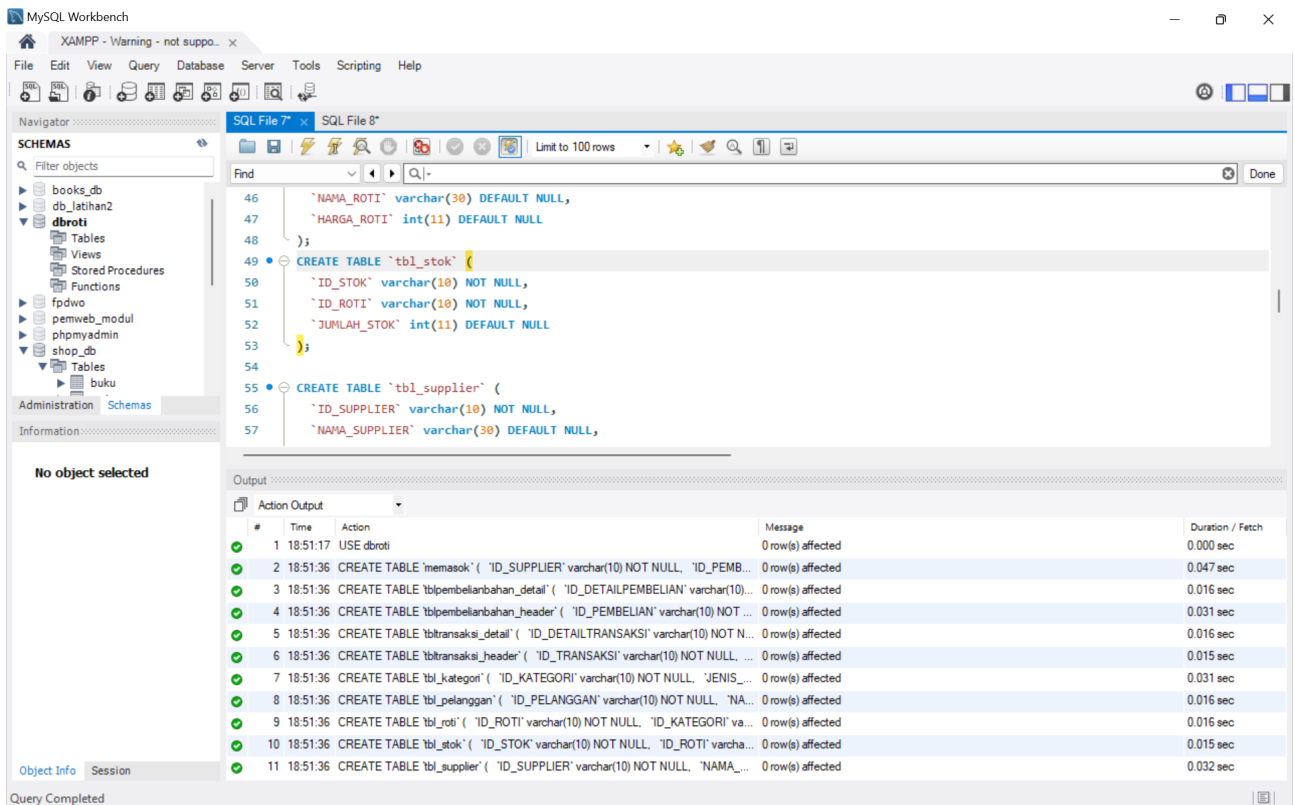
CREATE TABLE `tbl_roti` (
  `ID_ROTI` varchar(10) NOT NULL,
  `ID_KATEGORI` varchar(10) NOT NULL,
  `NAMA_ROTI` varchar(30) DEFAULT NULL,
  `HARGA_ROTI` int(11) DEFAULT NULL
);

CREATE TABLE `tbl_stok` (
  `ID_STOK` varchar(10) NOT NULL,
  `ID_ROTI` varchar(10) NOT NULL,
  `JUMLAH_STOK` int(11) DEFAULT NULL
);

CREATE TABLE `tbl_supplier` (
  `ID_SUPPLIER` varchar(10) NOT NULL,
  `NAMA_SUPPLIER` varchar(30) DEFAULT NULL,
  `ALAMAT_SUPPLIER` varchar(100) DEFAULT NULL,
  `NOTELP_SUPPLIER` decimal(12,0) DEFAULT NULL
);

```

Berikut merupakan gambar hasil **output** dari **create table** pada database **dbroti**:



Pada tahap selanjutnya yaitu dengan menentukan primary key dan foreign key pada setiap tabel yang telah ditentukan pada tahap pembuatan CDM dan PDM. Tujuan mendefinisikan primary key pada setiap tabel pada database **dbroti** ini yaitu untuk memberikan identitas unik dan pengidentifikasi untuk setiap baris data yang terdapat pada tabel. Selain itu, primary key juga bertujuan untuk dijadikan kunci utama agar dapat menggabungkan data yang relevan di table lain yang terdapat pada database ini. Dengan menentukan primary key dapat memastikan integritas data, mencegah terjadinya duplikasi dan memungkinkan pencarian serta pengambilan data secara efisien. Selain itu, menentukan foreign key digunakan untuk mengidentifikasi hubungan antar tabel. Maka dari itu, menentukan primary key dan foreign key dilakukan agar dapat membuat struktur database yang efisien dan dapat mengoptimalkan query serta meningkatkan efisiensi dan efektivitas dalam operasi database. Berikut merupakan query untuk menentukan **primary key** dan **foreign key** pada setiap tabel yang terdapat pada database **dbroti**:

```
ALTER TABLE `memasok`
  ADD PRIMARY KEY (`ID_SUPPLIER`,`ID_PEMBELIAN`),
  ADD KEY `FK_MEMASOK2` (`ID_PEMBELIAN`);

ALTER TABLE `tblpembelianbahan_detail`
  ADD PRIMARY KEY (`ID_DETAILPEMBELIAN`),
  ADD KEY `FK_DETAI_1` (`ID_PEMBELIAN`);

ALTER TABLE `tblpembelianbahan_header`
  ADD PRIMARY KEY (`ID_PEMBELIAN`);

ALTER TABLE `tbltransaksi_detail`
  ADD PRIMARY KEY (`ID_DETAILTRANSAKSI`),
  ADD KEY `FK_DETAIL` (`ID_TRANSAKSI`);

ALTER TABLE `tbltransaksi_header`
  ADD PRIMARY KEY (`ID_TRANSAKSI`),
```

```

ADD KEY `FK_HEADER` (`ID_ROTI`),
ADD KEY `FK_MELAKUKAN` (`ID_PELANGGAN`);

ALTER TABLE `tbl_kategori`
ADD PRIMARY KEY (`ID_KATEGORI`);

ALTER TABLE `tbl_pelanggan`
ADD PRIMARY KEY (`ID_PELANGGAN`);

ALTER TABLE `tbl_roti`
ADD PRIMARY KEY (`ID_ROTI`),
ADD KEY `FK_DIKATEGORIKAN` (`ID_KATEGORI`);

ALTER TABLE `tbl_stok`
ADD PRIMARY KEY (`ID_STOK`),
ADD KEY `FK_MEMILIKI` (`ID_ROTI`);

ALTER TABLE `tbl_supplier`
ADD PRIMARY KEY (`ID_SUPPLIER`);

ALTER TABLE `memasok`
ADD CONSTRAINT `FK_MEMASOK` FOREIGN KEY (`ID_SUPPLIER`) REFERENCES
`tbl_supplier` (`ID_SUPPLIER`),
ADD CONSTRAINT `FK_MEMASOK2` FOREIGN KEY (`ID_PEMBELIAN`) REFERENCES
`tblpembelianbahan_header` (`ID_PEMBELIAN`);

ALTER TABLE `tblpembelianbahan_detail`
ADD CONSTRAINT `FK_DETAI_1` FOREIGN KEY (`ID_PEMBELIAN`) REFERENCES
`tblpembelianbahan_header` (`ID_PEMBELIAN`);

ALTER TABLE `tbltransaksi_detail`
ADD CONSTRAINT `FK_DETAIL` FOREIGN KEY (`ID_TRANSAKSI`) REFERENCES
`tbltransaksi_header` (`ID_TRANSAKSI`);

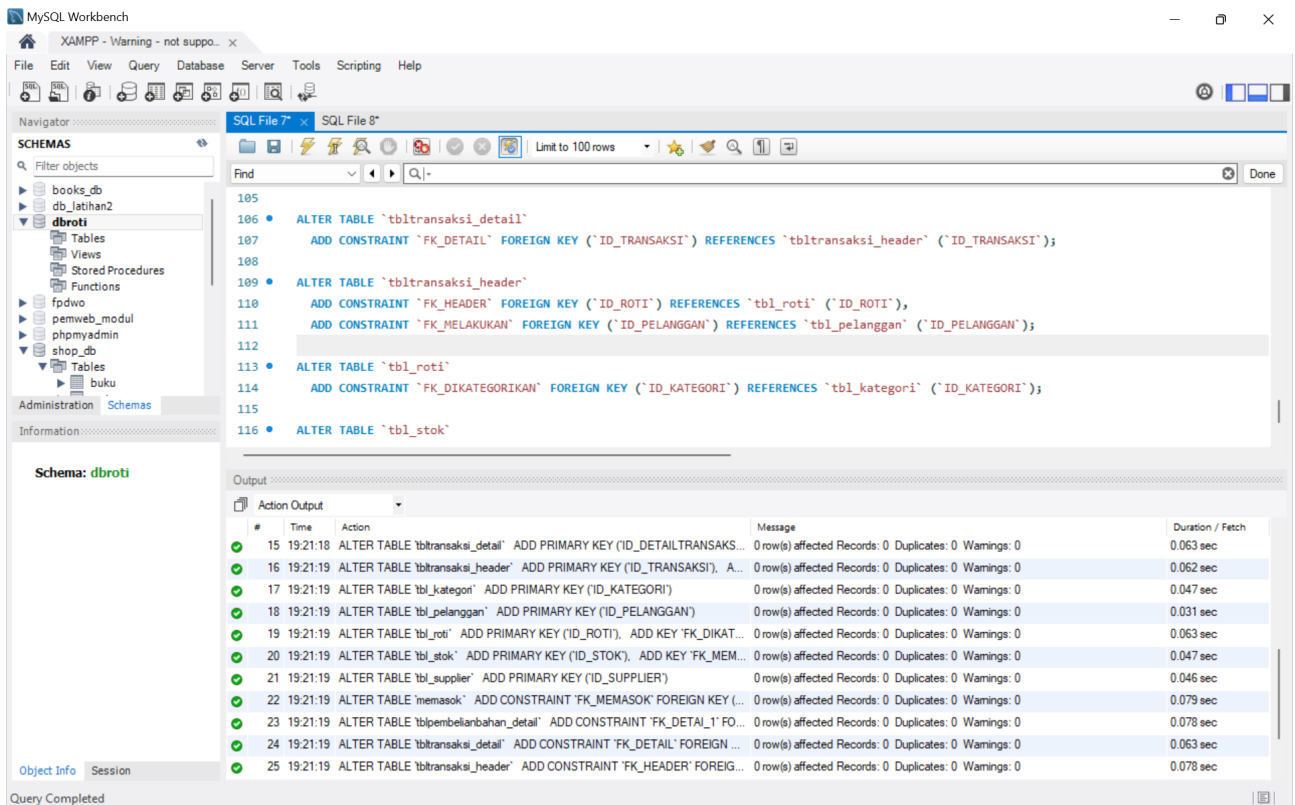
ALTER TABLE `tbltransaksi_header`
ADD CONSTRAINT `FK_HEADER` FOREIGN KEY (`ID_ROTI`) REFERENCES `tbl_roti`
(`ID_ROTI`),
ADD CONSTRAINT `FK_MELAKUKAN` FOREIGN KEY (`ID_PELANGGAN`) REFERENCES
`tbl_pelanggan` (`ID_PELANGGAN`);

ALTER TABLE `tbl_roti`
ADD CONSTRAINT `FK_DIKATEGORIKAN` FOREIGN KEY (`ID_KATEGORI`) REFERENCES
`tbl_kategori` (`ID_KATEGORI`);

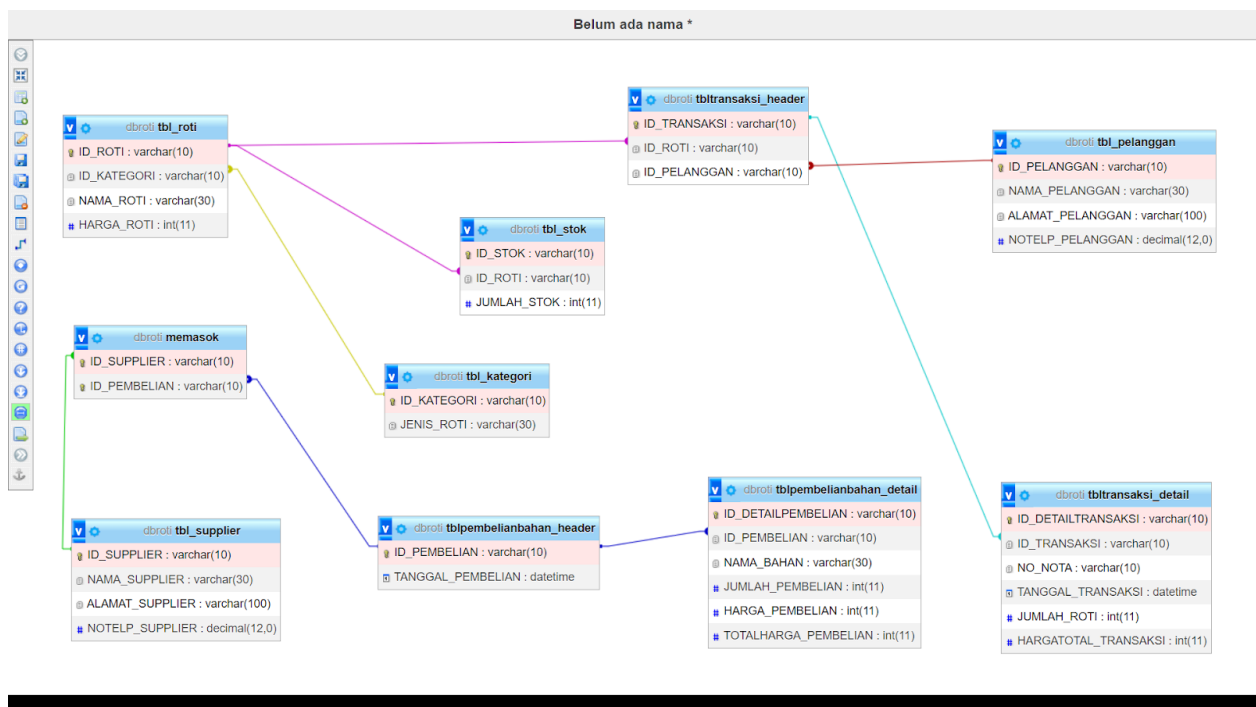
ALTER TABLE `tbl_stok`
ADD CONSTRAINT `FK_MEMILIKI` FOREIGN KEY (`ID_ROTI`) REFERENCES `tbl_roti`
(`ID_ROTI`);
COMMIT;

```

Berikut merupakan hasil output pada penentuan primary key dan foreign key dalam database **dbroti**:



Berikut merupakan gambar hasil dari struktur database **dbroti** yang telah kami rancang :



### 3. Penjelasan Create User (Min. 3 User)

Pada tahap ini dilakukan pembuatan create user pada cmd, 3 user yang dibuat antara lain :

- user\_21082010092 dengan password usabdb
- user\_21082010104 dengan password usabdb
- user\_21082010111 dengan password usabdb



Dari ketiga user yang sudah dibuat, masing masing nantinya akan diberikan role (peran) beserta hak aksesnya pada tahap selanjutnya.

```
Command Prompt - mysql -u x + v
C:\>cd xampp\mysql\bin

C:\xampp\mysql\bin>mysql -u root -p ;
Enter password:
ERROR 1049 (42000): Unknown database 'dbroti';

C:\xampp\mysql\bin>mysql -u root -p;
'mysql-u' is not recognized as an internal or external command,
operable program or batch file.

C:\xampp\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 834
Server version: 10.4.27-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.

MariaDB [(none)]> use dbroti;
Database changed
MariaDB [dbroti]> create user 'user_21082010092' identified by 'uasabd';
Query OK, 0 rows affected (0.038 sec)

MariaDB [dbroti]> create user 'user_21082010104' identified by 'uasabd';
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> create user 'user_21082010111' identified by 'uasabd';
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> |
```

#### 4. Penjelasan hak akses user / Role User

Pada tahap ini, setiap user yang dibuat akan diberikan role (peran) beserta hak aksesnya. Sebelum itu, harus membuat create role terlebih dahulu. Dalam studi kasus database roti di kelompok kami, role yang dibuat ada 3 dengan asumsi tugas utamanya, antara lain :

- Role Kasir : sebagai pencatat transaksi penjualan
- Role Admin : sebagai pengelola data
- Role Owner : sebagai pemilik dan administrator tingkat tinggi

```
Command Prompt - mysql -u x + v
C:\xampp\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 834
Server version: 10.4.27-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.

MariaDB [(none)]> use dbroti;
Database changed
MariaDB [dbroti]> create user 'user_21082010092' identified by 'uasabd';
Query OK, 0 rows affected (0.038 sec)

MariaDB [dbroti]> create user 'user_21082010104' identified by 'uasabd';
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> create user 'user_21082010111' identified by 'uasabd';
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> create role admin;
Query OK, 0 rows affected (0.023 sec)

MariaDB [dbroti]> create role kasir;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> create role owner;
Query OK, 0 rows affected (0.014 sec)

MariaDB [dbroti]> |
```

Selanjutnya, disini tahap memberikan hak akses role (peran) kepada masing-masing user yang telah dibuat beserta hak akses kepada masing-masing role (peran), antara lain :

- Table **tbltransaksi\_header, tbltransaksi\_detal, dan tbl\_pelanggan** □ untuk **user\_21082010092** yang memiliki hak akses role sebagai **kasir** dan memiliki hak akses berupa (Insert, Update) karena,
  1. **Insert** : untuk menginputkan data transaksi karena seorang kasir perlu mencatat transaksi penjualan serta data yang berkaitan dengan customer.
  2. **Update** : untuk memperbarui informasi data yang berkaitan dengan customer apabila terdapat kesalahan dalam penginputan
- Table **tbl\_roti, tbl\_kategori, dan tbl\_stok** □ untuk **user\_21082010104** yang memiliki hak akses role sebagai **admin** dan memiliki hak akses berupa (Insert, Delete) karena,
  1. **Insert** : digunakan untuk admin dalam menginputkan data roti beserta stoknya
  2. **Delete** : untuk melakukan pengelolaan data dengan menghapus data yang salah atau membersihkan data yang sudah tidak diperlukan
- Table **memasok, tbl\_supplier, tblpembelianbahan\_header, tblpembelianbahan\_detail, tbl\_roti, tbl\_kategori, dan tbl\_stok** □ untuk **user\_21082010111** yang memiliki hak akses role sebagai **owner** dan memiliki hak akses berupa (Delete, Update) karena,
  1. **Delete** : untuk melakukan penghapusan data yang mungkin berada pada wewenang administrator tingkat tinggi
  2. **Update** : untuk melakukan pembaruan data yang juga mungkin berada pada wewenang administrator tingkat tinggi

```
MariaDB [dbroti]> grant kasir to 'user_21082010092'@'localhost'
identified by 'uasabd';
Query OK, 0 rows affected (0.028 sec)

MariaDB [dbroti]> grant admin to 'user_21082010104'@'localhost'
identified by 'uasabd';
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant owner to 'user_21082010111'@'localhost'
identified by 'uasabd';
Query OK, 0 rows affected (0.002 sec)

MariaDB [dbroti]>
```

```
Command Prompt - mysql -u x + v
MariaDB [dbroti]> grant insert, update on dbroti.tbltransaksi_header to kasir;
Query OK, 0 rows affected (0.014 sec)

MariaDB [dbroti]> grant insert, update on dbroti.tbltransaksi_detail to kasir;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant insert, update on dbroti.tbl_pelanggan to kasir;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_roti to admin;
Query OK, 0 rows affected (0.073 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_kategori to admin;
Query OK, 0 rows affected (0.006 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_stok to admin;
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> grant delete, update on dbroti.memasok to owner;
Query OK, 0 rows affected (0.009 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tbl_supplier to owner;
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tblpembelianbahan_header to owner;
Query OK, 0 rows affected (0.005 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tblpembelianbahan_detail to owner;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tbl_roti to owner;
Query OK, 0 rows affected (0.003 sec)
```

```
Command Prompt - mysql -u x + v
MariaDB [dbroti]> grant insert, update on dbroti.tbl_pelanggan to kasir;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_roti to admin;
Query OK, 0 rows affected (0.073 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_kategori to admin;
Query OK, 0 rows affected (0.006 sec)

MariaDB [dbroti]> grant insert, delete on dbroti.tbl_stok to admin;
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> grant delete, update on dbroti.memasok to owner;
Query OK, 0 rows affected (0.009 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tbl_supplier to owner;
Query OK, 0 rows affected (0.004 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tblpembelianbahan_header to owner;
Query OK, 0 rows affected (0.005 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tblpembelianbahan_detail to owner;
Query OK, 0 rows affected (0.003 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tbl_roti to owner;
Query OK, 0 rows affected (0.003 sec)

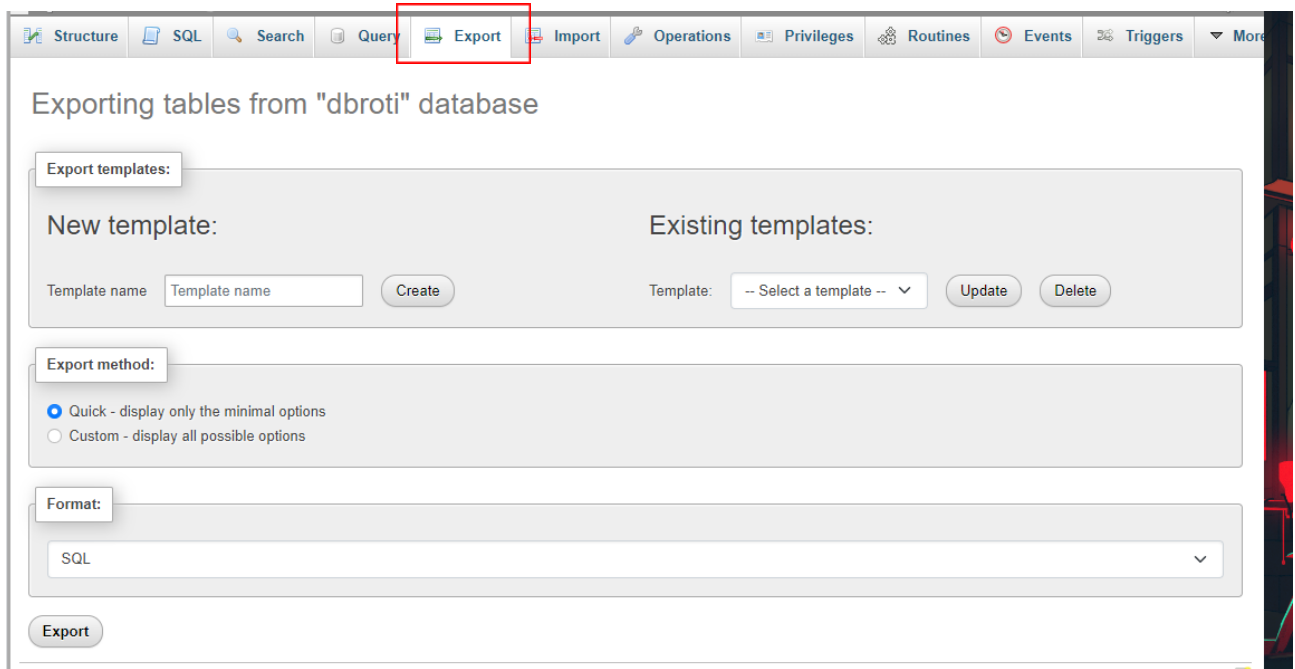
MariaDB [dbroti]> grant delete, update on dbroti.tbl_kategori to owner;
Query OK, 0 rows affected (0.002 sec)

MariaDB [dbroti]> grant delete, update on dbroti.tbl_stok to owner;
Query OK, 0 rows affected (0.003 sec)
```

## 5. Export dan Import database

- **Export dan Import Database pada phpmyadmin**

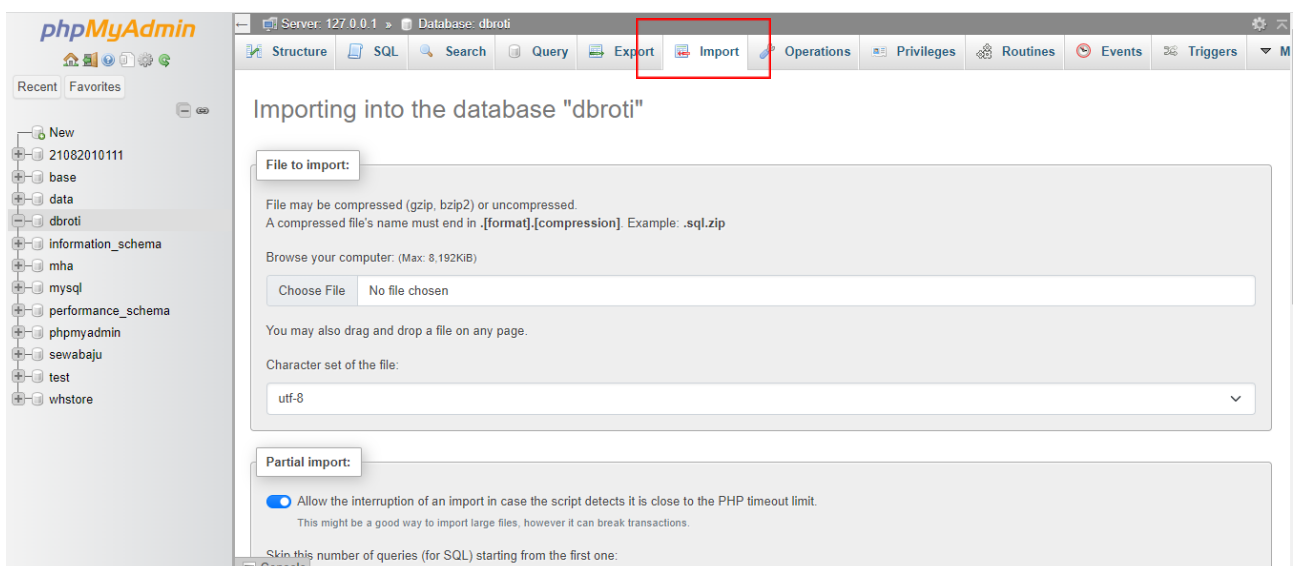
Cara dalam mengexport database cukup mudah, yakni dengan cara masuk ke menu export, lalu muncul tampilan pada gambar dibawah ini, anda bisa membenahi beberapa pilihan sesuai dengan keinginan anda, misalkan anda ingin format export selain dari sql (bisa xml, csv, dan sebagainya).



Setelah anda sudah mengatur opsi sesuai dengan pilihan, maka anda dapat mengklik button export, lalu file akan otomatis terdownload masuk ke dalam penyimpanan anda.



Berikutnya merupakan langkah untuk mengimport database, sebelum import pastikan anda sudah memiliki file yang akan diimport dengan format sesuai yg tersedia (sql, csv, dan sebagainya). Setelah itu, anda dapat langsung masuk pada bagian import sehingga muncul tampilan dibawah ini.



Anda bisa langsung memasukkan file yang sudah disiapkan dengan klik bagian choose file, lalu tekan OK. Jangan lupa menyesuaikan dengan format file yang anda miliki, dalam praktik ini menggunakan format SQL. Setelah semua opsi sudah diatur, maka anda dapat mengklik tombol import.

- **Export Database dengan CMD**

Pastikan anda sudah mempunyai database yang ingin di export. Setelah itu, anda dapat menentukan lokasi penyimpanan database, dalam kelompok kami menggunakan disk D sebagai tempat penyimpanan, membuat folder berupa backup. Input syntax berikut satu demi satu.

**Syntax :**

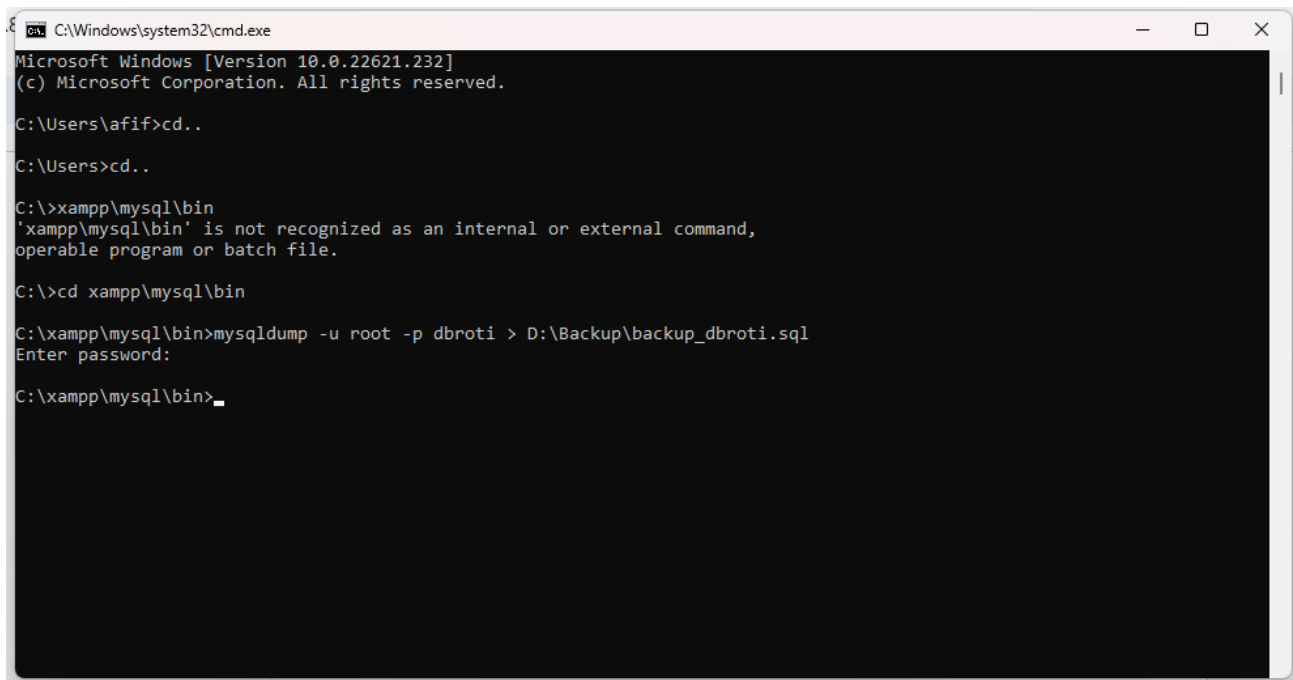
```
C:\Users\afif>cd..
```

```
C:\Users>cd..
```

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

```
C:\xampp\mysql\bin>mysqldump -u root -p dbroti > D:\Backup\backup_dbroti.sql
```

Enter password:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22621.232]
(c) Microsoft Corporation. All rights reserved.

C:\Users\afif>cd..
C:\Users>cd..
C:\>cd xampp\mysql\bin
'xampp\mysql\bin' is not recognized as an internal or external command,
operable program or batch file.
C:\>cd xampp\mysql\bin
C:\xampp\mysql\bin>mysqldump -u root -p dbroti > D:\Backup\backup_dbroti.sql
Enter password:
C:\xampp\mysql\bin>_
```

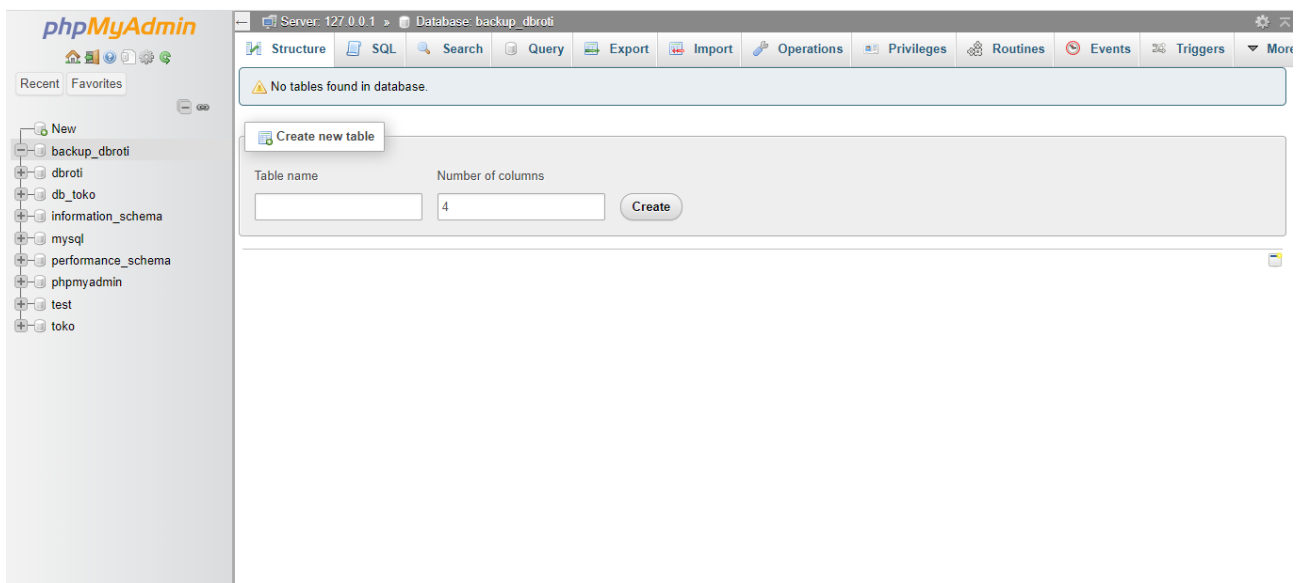
Anda dapat memeriksa lokasi tempat penyimpanan saat export database. Pada gambar dibawah ini menunjukan export berhasil dilakukan.

📁 > This PC > Data (D:) > backup

| Name            | Date modified     | Type            | Size  |
|-----------------|-------------------|-----------------|-------|
| 📁 backup        | 6/13/2023 2:57 PM | SQL Source File | 0 KB  |
| 📁 backup_dbroti | 6/13/2023 7:20 PM | SQL Source File | 10 KB |
| 📁 db_toko       | 6/13/2023 4:37 PM | SQL Source File | 3 KB  |
| 📁 toko          | 6/13/2023 3:03 PM | SQL Source File | 3 KB  |

- **Import Database dengan CMD**

Sebelum import database, pastikan anda telah membuat database baru kosong. Kelompok kami membuat database kosong dengan nama backup\_dbroti di phpmyadmin. Setelah itu, silakan buka CMD, dan mengetik satu persatu syntax dibawah ini,



### Syntax:

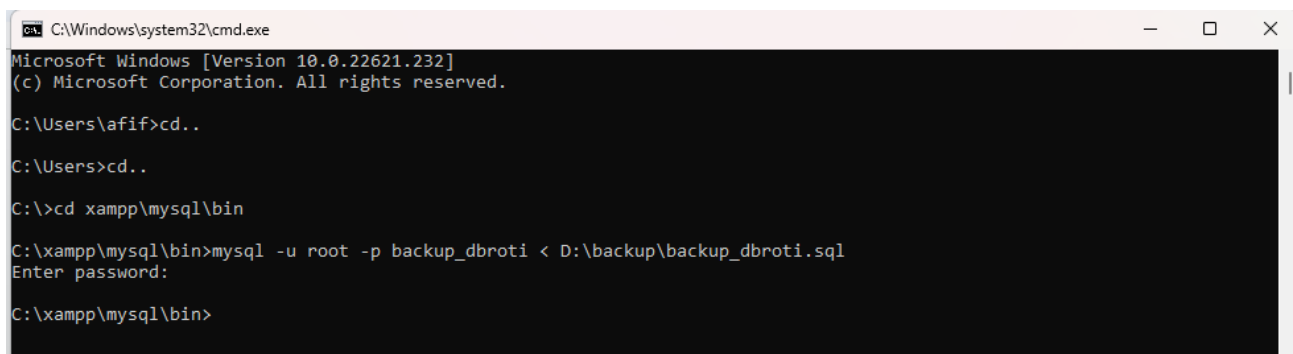
```
C:\Users\afif>cd..
```

```
C:\Users>cd..
```

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

```
C:\xampp\mysql\bin>mysql -u root -p backup_dbroti < D:\backup\backup_dbroti.sql
```

Enter password:



Setelah enter, lalu cek keberhasilan import database di phpmyadmin. Jika berhasil, maka data yang akan diimpor masuk ke dalam database baru tersebut.

phpMyAdmin

Recent Favorites

New

backup\_dbroti

dbroti

db\_toko

information\_schema

mysql

performance\_schema

phpmyadmin

test

toko

Server: 127.0.0.1 » Database: backup\_dbroti

StructureSQLSearchQueryExportImportOperationsPrivilegesRoutinesEventsTriggers

Filters

Containing the word:

| Table   | Action   | Rows | Type   | Collation          | Size      | Overhead |
|---|--|------|--------|--------------------|-----------|----------|
| <input type="checkbox"/> memasok                  | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 32.0 K18  | -        |
| <input type="checkbox"/> tblpembelianbahan_detail | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 32.0 K18  | -        |
| <input type="checkbox"/> tblpembelianbahan_header | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 K18  | -        |
| <input type="checkbox"/> tbltransaksi_detail      | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 32.0 K18  | -        |
| <input type="checkbox"/> tbltransaksi_header      | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 48.0 K18  | -        |
| <input type="checkbox"/> tbl_kategori             | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 K18  | -        |
| <input type="checkbox"/> tbl_pelanggan            | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 K18  | -        |
| <input type="checkbox"/> tbl_roti                 | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 32.0 K18  | -        |
| <input type="checkbox"/> tbl_stok                 | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 32.0 K18  | -        |
| <input type="checkbox"/> tbl_supplier             | Browse  Structure  Search  Insert  Empty  Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 K18  | -        |
| 10 tables   | Sum  | 0    | InnoDB | utf8mb4_general_ci | 272.0 K18 | 0 B      |

⬆

☐ Check all

With selected:

Print

Data dictionary