

Nama : Rafli

NIM : 3332190067

Jurusan : Teknik Elektro

UAS Perancangan Basis Data + Tutorial

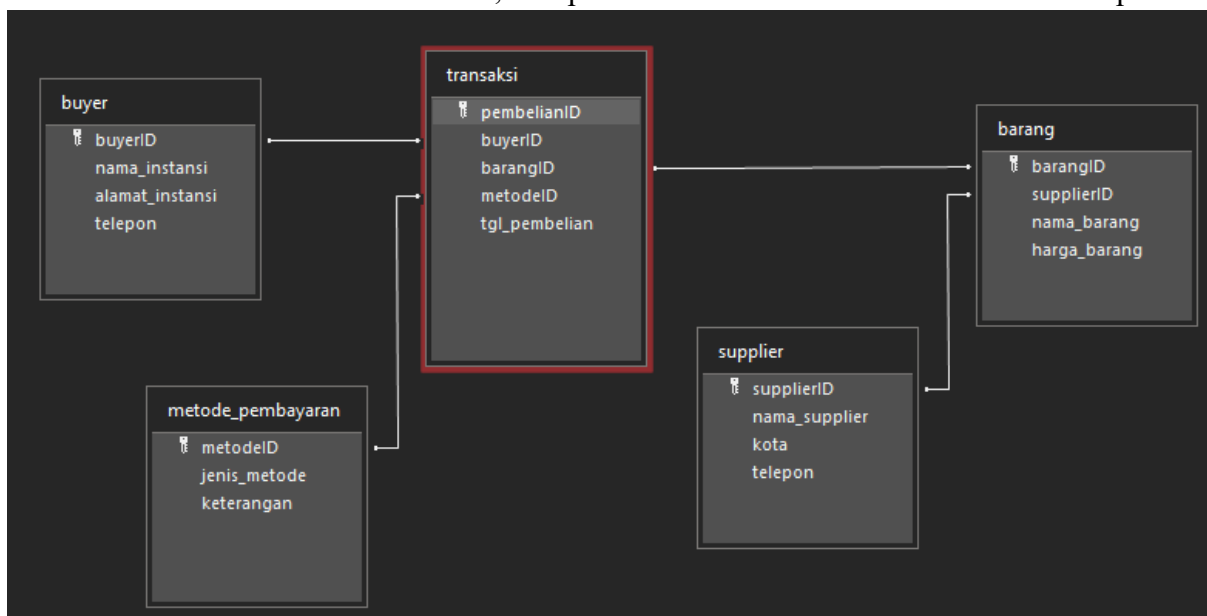
Universitas Sultan Ageng Tirtayasa

Database Supplier Industri

Deskripsi singkat mengenai basis data yang penulis rancang

A. Perancangan Database Menggunakan Microsoft Access

1. Buka Microsoft Access
2. Klik “Table” pada toolbar “Create”
3. Klik kanan pada tabel yang telah dibuat pada window “Table”
4. Klik “Design View”
5. Masukkan nama tabel
6. Isi field dan tipe data sesuai dengan kebutuhan
7. Ulangi Langkah 3-6 untuk tabel yang lain
8. Setelah semua table dibuat, klik pada toolbar “Table” dan klik “Relationship”

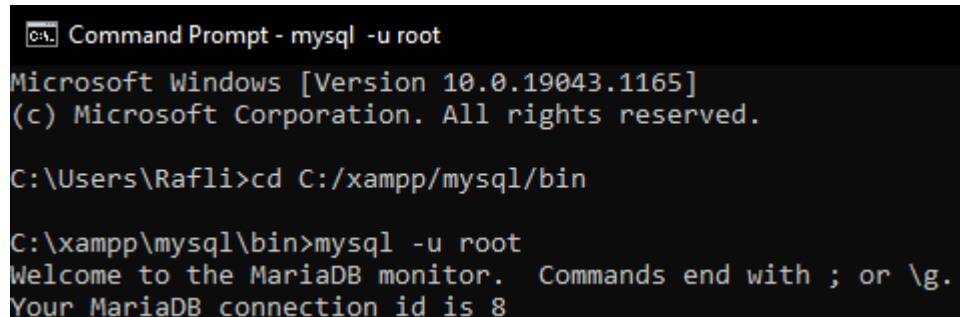


Gambar 1 Perancangan Database

B. Membuat Database

1. Jalankan Command Prompt
2. Ketikkan perintah berikut ini untuk terhubung ke mysql server:

```
mysql -u root
```



```
C:\Users\Rafli>cd C:/xampp/mysql/bin  
C:\xampp\mysql\bin>mysql -u root  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 8
```

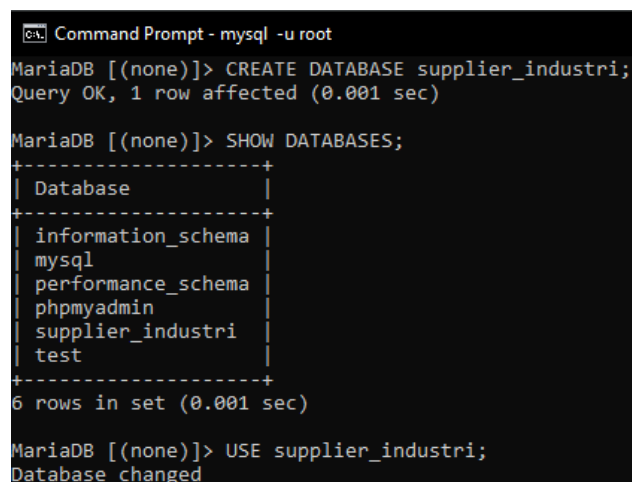
Gambar 2 Menghubungkan dengan mysql server

3. Setelah berhasil terhubung ke mysql server, selanjutnya yaitu kita akan membuat database baru dengan perintah sebagai berikut:

```
CREATE DATABASE supplier_industri;
```

4. Setelah berhasil dibuat databasenya, jika ingin melihat database yang sudah dibuat bisa menggunakan perintah sebagai berikut:

```
SHOW DATABASES;
```



```
MariaDB [(none)]> CREATE DATABASE supplier_industri;  
Query OK, 1 row affected (0.001 sec)  
  
MariaDB [(none)]> SHOW DATABASES;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| phpmyadmin |  
| supplier_industri |  
| test |  
+-----+  
6 rows in set (0.001 sec)  
  
MariaDB [(none)]> USE supplier_industri;  
Database changed
```

Gambar 3 Membuat Database

5. Selanjutnya yaitu kita akan menggunakan database tersebut dengan perintah sebagai berikut:

```
USE supplier_industri;
```

C. Membuat Tabel/Entitas

1. Setelah database berhasil digunakan, berikutnya semua tabel/entitas dibuat dengan perintah sebagai berikut:

```
CREATE TABLE buyer(buyerID int AUTO_INCREMENT NOT NULL PRIMARY KEY,  
nama_instansi varchar(50) NOT NULL, alamat varchar(50) NOT NULL,  
telepon varchar(15) NOT NULL);  
  
CREATE TABLE supplier(supplierID int AUTO_INCREMENT NOT NULL  
PRIMARY KEY, nama_supplier varchar(50) NOT NULL, kota varchar(50)  
NOT NULL, telepon_supplier varchar(15) NOT NULL);  
  
CREATE TABLE metode_pembayaran(metodeID int AUTO_INCREMENT NOT  
NULL PRIMARY KEY, jenis_metode varchar(50) NOT NULL, keterangan  
varchar(50) NOT NULL);  
  
CREATE TABLE barang(barangID int AUTO_INCREMENT NOT NULL PRIMARY  
KEY, supplierID int(50) NOT NULL, nama_barang varchar(50) NOT NULL,  
harga_barang int(13) NOT NULL);  
  
CREATE TABLE transaksi(pembelianID int AUTO_INCREMENT NOT NULL  
PRIMARY KEY, buyerID int(50) NOT NULL, barangID int(50) NOT NULL,  
metodeID int(50) NOT NULL, tgl_pembelian timestamp NOT NULL);
```

2. Setelah tabel dibuat, diatur initial value dari constraint beberapa tabel. Hal ini kita lakukan agar primary key masing-masing tabel memiliki keunikan, sehingga tidak terlihat redundansi.

```
ALTER TABLE metode_pembayaran AUTO_INCREMENT = 1001;  
  
ALTER TABLE supplier AUTO_INCREMENT = 2001;  
  
ALTER TABLE barang AUTO_INCREMENT = 3001;  
  
ALTER TABLE transaksi AUTO_INCREMENT = 4001;
```

3. Setelah semua tabel berhasil dibuat, kita bisa melihat list tabel yang telah dibuat beserta constraint dari masing-masing tabel dengan perintah sebagai berikut:

```
SHOW TABLES;  
  
DESC barang;  
  
DESC buyer;  
  
DESC metode_pembayaran;
```

```
DESC supplier;  
DESC transaksi;
```

Dengan cara ini, kita dapat melakukan cross-check terhadap tabel sebelum diinputkan data sehingga field yang akan diinputkan sesuai.

```
Command Prompt - mysql -u root  
MariaDB [supplier_industri]> SHOW TABLES;  
+-----+  
| Tables_in_supplier_industri |  
+-----+  
| barang  
| buyer  
| metode_pembayaran  
| supplier  
| transaksi  
+-----+  
5 rows in set (0.001 sec)  
  
MariaDB [supplier_industri]> DESC barang;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type      | Null | Key | Default | Extra      |  
+-----+-----+-----+-----+-----+-----+  
| barangID   | int(11)   | NO   | PRI | NULL    | auto_increment  
| supplierID | int(50)   | NO   |     | NULL    |  
| nama_barang | varchar(50) | NO   |     | NULL    |  
| harga_barang | int(13)   | NO   |     | NULL    |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.041 sec)  
  
MariaDB [supplier_industri]> DESC buyer;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type      | Null | Key | Default | Extra      |  
+-----+-----+-----+-----+-----+-----+  
| buyerID    | int(11)   | NO   | PRI | NULL    | auto_increment  
| nama_instansi | varchar(50) | NO   |     | NULL    |  
| alamat      | varchar(50) | NO   |     | NULL    |  
| telepon     | varchar(15) | NO   |     | NULL    |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.041 sec)  
  
MariaDB [supplier_industri]> DESC metode_pembayaran;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type      | Null | Key | Default | Extra      |  
+-----+-----+-----+-----+-----+-----+  
| metodeID   | int(11)   | NO   | PRI | NULL    | auto_increment  
| jenis_metode | varchar(50) | NO   |     | NULL    |  
| keterangan  | varchar(50) | NO   |     | NULL    |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.160 sec)
```

Gambar 4 Semua Tabel Yang Telah Dibuat

```
Command Prompt - mysql -u root
MariaDB [supplier_industri]> DESC supplier;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| supplierID     | int(11)       | NO   | PRI | NULL    | auto_increment |
| nama_supplier  | varchar(50)   | NO   |     | NULL    |                |
| kota           | varchar(50)   | NO   |     | NULL    |                |
| telepon_supplier | varchar(15)  | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.100 sec)

MariaDB [supplier_industri]> DESC transaksi;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default          | Extra          |
+-----+-----+-----+-----+-----+-----+
| pembelianID    | int(11)       | NO   | PRI | NULL            | auto_increment |
| buyerID       | int(50)       | NO   |     | NULL            |                |
| barangID       | int(50)       | NO   |     | NULL            |                |
| metodeID       | int(50)       | NO   |     | NULL            |                |
| tgl_pembelian  | timestamp     | NO   |     | current_timestamp() | on update current_timestamp() |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.155 sec)
```

Gambar 5 Semua Tabel Yang Telah Dibuat

D. Penginputan Data Pada Semua Tabel

1. Setelah dipastikan semua tabel yang diperlukan telah dibuat memiliki tipe data dan constraint yang sesuai, maka dapat dilakukan penginputan data
2. Ketikkan perintah sebagai berikut untuk menginputkan data kepada tabel. Berikut ini adalah contoh menginputkan data pada tabel “buyer”. Lakukan hal yang sama untuk tabel lain.

```
INSERT INTO buyer values (NULL, 'PT ABC', 'Jababeka Blok H01', '083762839971');
```

```

C:\ Command Prompt - mysql -u root
MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT ABC', 'Jababeka H01', '083762839971');
Query OK, 1 row affected (0.108 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT DEF', 'Jababeka H02', '083762839970');
Query OK, 1 row affected (0.105 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT GHI', 'Jababeka H03', '083762839973');
Query OK, 1 row affected (0.112 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT JKL', 'Jababeka H04', '083762839974');
Query OK, 1 row affected (0.180 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT MNO', 'Jababeka H05', '083762839975');
Query OK, 1 row affected (0.108 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT PQR', 'Jababeka H06', '083762839976');
Query OK, 1 row affected (0.139 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT STU', 'Jababeka H07', '083762839977');
Query OK, 1 row affected (0.113 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT VWX', 'Jababeka H08', '083762839978');
Query OK, 1 row affected (0.117 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT XYZ', 'Jababeka H09', '083762839979');
Query OK, 1 row affected (0.111 sec)

MariaDB [supplier_industri]> INSERT INTO buyer values(NULL, 'PT BCA', 'Jababeka H10', '083762839980');

```

Gambar 6 Penginputan Data Tabel “buyer”

4. Tabel “metode_pembayaran”

```

C:\ Command Prompt - mysql -u root
MariaDB [supplier_industri]> INSERT INTO metode_pembayaran values(NULL, 'Tunai', 'Bilyet Giro');
Query OK, 1 row affected (0.137 sec)

MariaDB [supplier_industri]> INSERT INTO metode_pembayaran values(NULL, 'Non-Tunai', 'Uang Elektronik');
Query OK, 1 row affected (0.118 sec)

```

Gambar 7 Penginputan Data Tabel “metode_pembayaran”

5. Tabel “supplier”

```

C:\ Command Prompt - mysql -u root
MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Samsung', 'Bekasi', '081635321201')
-> ;
Query OK, 1 row affected (0.109 sec)

MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Omron', 'Tangerang', '081635321202'):
-> \c
MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Omron', 'Tangerang', '081635321202');
Query OK, 1 row affected (0.096 sec)

MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Cisco', 'Jakarta', '081635321203');
Query OK, 1 row affected (0.100 sec)

MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Microsoft', 'Jakarta', '081635321204');
Query OK, 1 row affected (0.133 sec)

MariaDB [supplier_industri]> INSERT INTO supplier values(NULL, 'Nvidia', 'Jakarta', '081635321205');
Query OK, 1 row affected (0.103 sec)

```

Gambar 8 Penginputan Data Tabel “supplier”

6. Tabel “barang”

```
Command Prompt - mysql -u root
MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2001', 'AC 1 PK', '2000000');
Query OK, 1 row affected (0.131 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2001', 'Laserjet Printer', '3500000');
Query OK, 1 row affected (0.072 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2002', 'PLC Compact', '3000000');
Query OK, 1 row affected (0.050 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2002', 'PLC Modular', '5000000');
Query OK, 1 row affected (0.075 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2003', 'Gigabit Router', '1500000');
Query OK, 1 row affected (0.079 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2003', 'Wi-Fi Range Extender', '750000');
Query OK, 1 row affected (0.090 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2004', 'Windows 10 Enterprise License', '5000000');
Query OK, 1 row affected (0.074 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2004', 'Microsoft Office 2019 License', '2000000');
Query OK, 1 row affected (0.148 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2005', 'GeForce GTX 1050', '2000000');
Query OK, 1 row affected (0.095 sec)

MariaDB [supplier_industri]> INSERT INTO barang values(NULL, '2005', 'Jetson Nano', '1200000');
Query OK, 1 row affected (0.133 sec)
```

Gambar 9 Penginputan Data Tabel “barang”

7. Ketikkan perintah sebagai berikut untuk menampilkan data yang telah diinputkan pada tabel untuk melihat apakah data yang terinput sudah sesuai. Lakukan hal yang sama untuk tabel lain.

```
SELECT * FROM nama_tabel;
```

```
Command Prompt - mysql -u root
MariaDB [supplier_industri]> SELECT * FROM supplier;
+-----+-----+-----+-----+
| supplierID | nama_supplier | kota      | telepon_supplier |
+-----+-----+-----+-----+
| 2001       | Samsung       | Bekasi    | 081635321201     |
| 2002       | Omron         | Tangerang | 081635321202     |
| 2003       | Cisco         | Jakarta   | 081635321203     |
| 2004       | Microsoft     | Jakarta   | 081635321204     |
| 2005       | Nvidia        | Jakarta   | 081635321205     |
+-----+-----+-----+-----+
```

Gambar 10 Penampilan Isi Data Tabel “barang”

```
Command Prompt - mysql -u root

MariaDB [supplier_industri]> SELECT * FROM barang;
+-----+-----+-----+-----+
| barangID | supplierID | nama_barang | harga_barang |
+-----+-----+-----+-----+
| 3001 | 2001 | AC 1 PK | 2000000 |
| 3002 | 2001 | Laserjet Printer | 3500000 |
| 3003 | 2002 | PLC Compact | 3000000 |
| 3004 | 2002 | PLC Modular | 5000000 |
| 3005 | 2003 | Gigabit Router | 1500000 |
| 3006 | 2003 | Wi-Fi Range Extender | 750000 |
| 3007 | 2004 | Windows 10 Enterprise License | 5000000 |
| 3008 | 2004 | Microsoft Office 2019 License | 2000000 |
| 3009 | 2005 | GeForce GTX 1050 | 2000000 |
| 3010 | 2005 | Jetson Nano | 1200000 |
+-----+-----+-----+-----+
10 rows in set (0.000 sec)

MariaDB [supplier_industri]> SELECT * FROM buyer;
+-----+-----+-----+-----+
| buyerID | nama_instansi | alamat | telepon |
+-----+-----+-----+-----+
| 1 | PT ABC | Jababeka H01 | 083762839971 |
| 2 | PT DEF | Jababeka H02 | 083762839970 |
| 3 | PT GHI | Jababeka H03 | 083762839973 |
| 4 | PT JKL | Jababeka H04 | 083762839974 |
| 5 | PT MNO | Jababeka H05 | 083762839975 |
| 6 | PT PQR | Jababeka H06 | 083762839976 |
| 7 | PT STU | Jababeka H07 | 083762839977 |
| 8 | PT VWX | Jababeka H08 | 083762839978 |
| 9 | PT XYZ | Jababeka H09 | 083762839979 |
| 10 | PT BCA | Jababeka H10 | 083762839980 |
+-----+-----+-----+-----+
10 rows in set (0.001 sec)

MariaDB [supplier_industri]> SELECT * FROM metode_pembayaran;
+-----+-----+-----+
| metodeID | jenis_metode | keterangan |
+-----+-----+-----+
| 1001 | Tunai | Bilyet Giro |
| 1002 | Non-Tunai | Uang Elektronik |
+-----+-----+-----+
2 rows in set (0.000 sec)
```

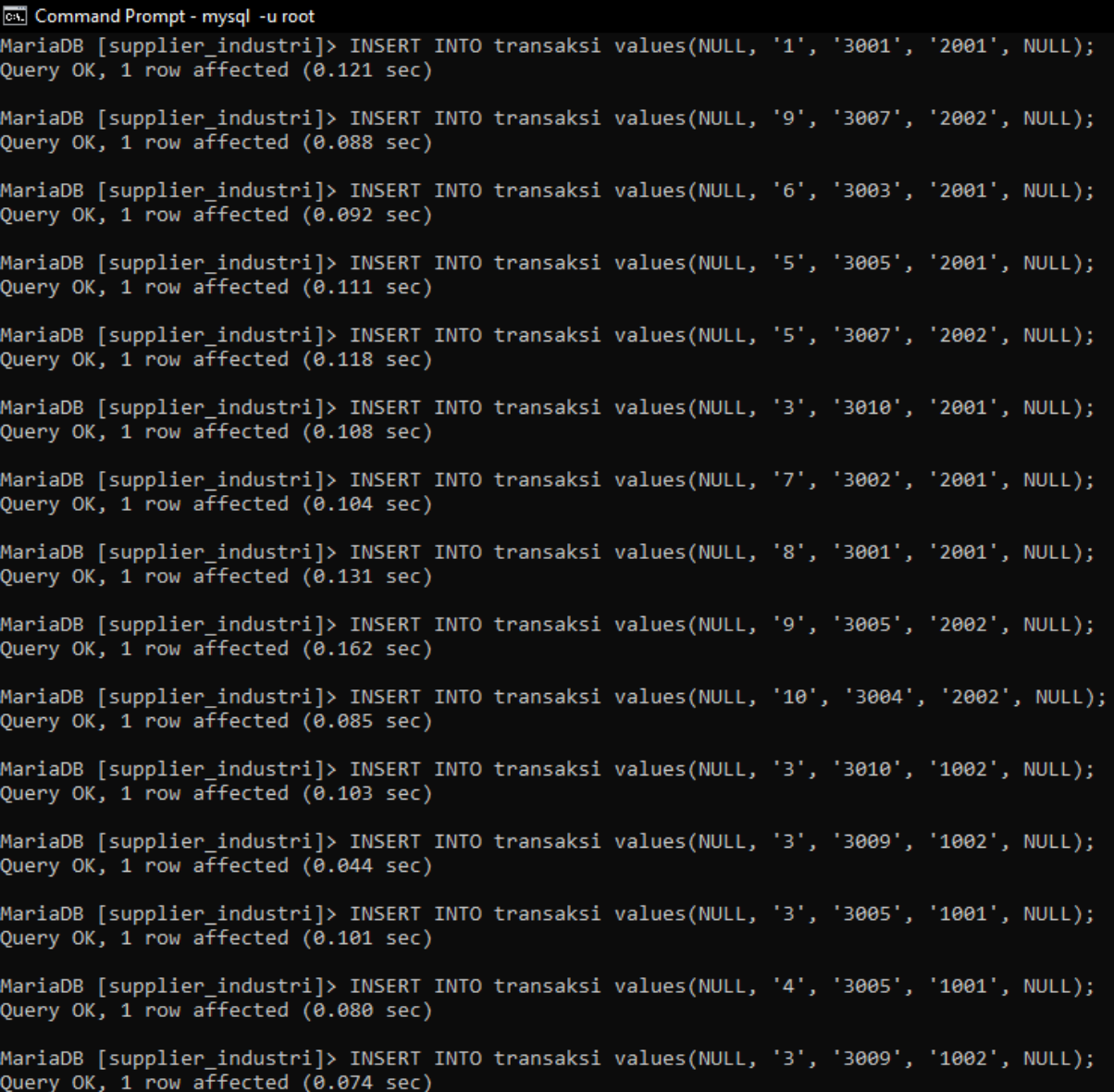
Gambar 11 Penampilan Isi Data Tabel

E. Simulasi Transaksi Pembelian

1. Ketikkan perintah berikut untuk mensimulasikan sebuah transaksi pembelian

```
INSERT INTO transaksi values(NULL, '1', '3001', '1001', NULL);
```

2. Lakukan sebanyak 100 transaksi atau lebih



```
CA\ Command Prompt - mysql -u root
MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '1', '3001', '2001', NULL);
Query OK, 1 row affected (0.121 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '9', '3007', '2002', NULL);
Query OK, 1 row affected (0.088 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '6', '3003', '2001', NULL);
Query OK, 1 row affected (0.092 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '5', '3005', '2001', NULL);
Query OK, 1 row affected (0.111 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '5', '3007', '2002', NULL);
Query OK, 1 row affected (0.118 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '3', '3010', '2001', NULL);
Query OK, 1 row affected (0.108 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '7', '3002', '2001', NULL);
Query OK, 1 row affected (0.104 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '8', '3001', '2001', NULL);
Query OK, 1 row affected (0.131 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '9', '3005', '2002', NULL);
Query OK, 1 row affected (0.162 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '10', '3004', '2002', NULL);
Query OK, 1 row affected (0.085 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '3', '3010', '1002', NULL);
Query OK, 1 row affected (0.103 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '3', '3009', '1002', NULL);
Query OK, 1 row affected (0.044 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '3', '3005', '1001', NULL);
Query OK, 1 row affected (0.101 sec)

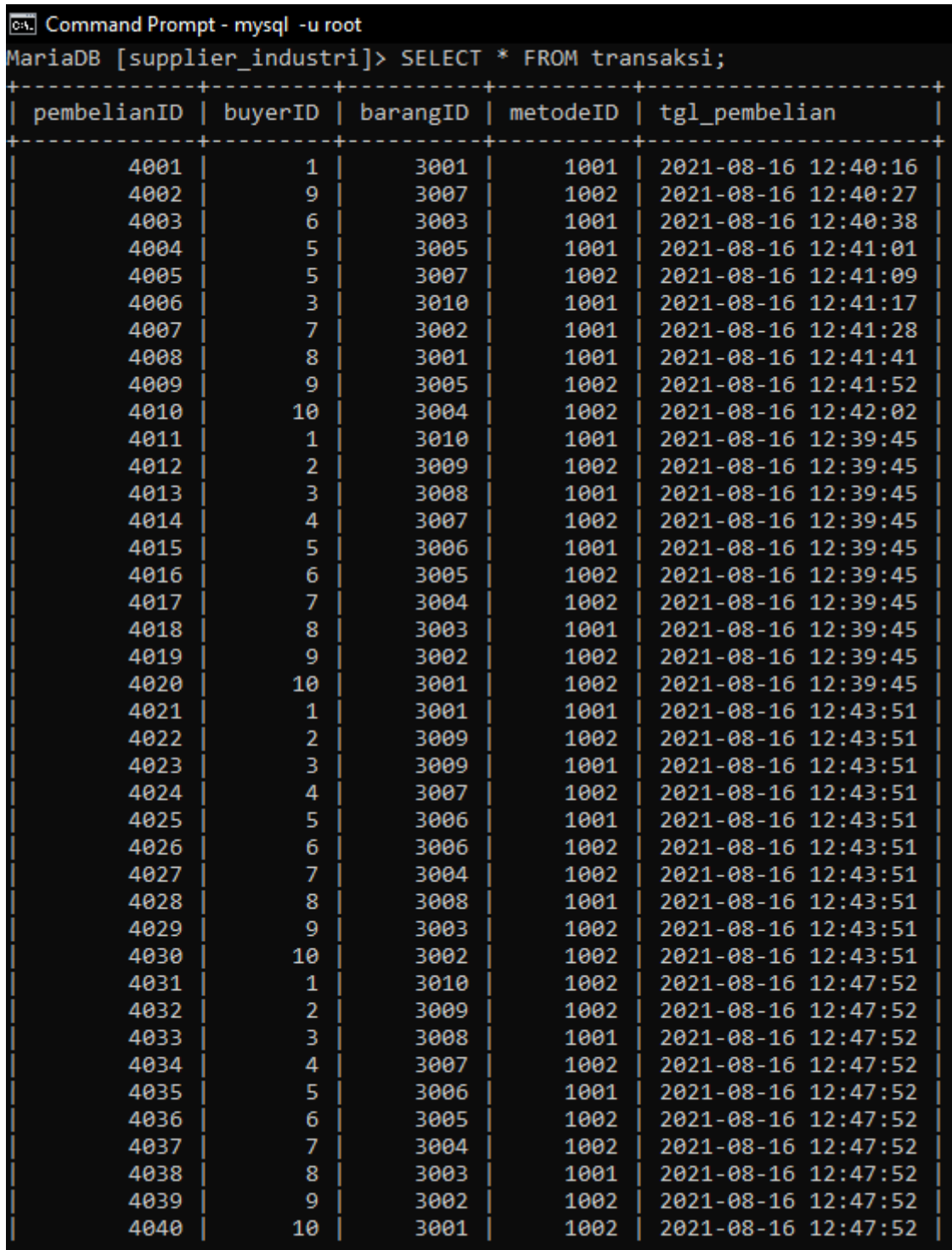
MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '4', '3005', '1001', NULL);
Query OK, 1 row affected (0.080 sec)

MariaDB [supplier_industri]> INSERT INTO transaksi values(NULL, '3', '3009', '1002', NULL);
Query OK, 1 row affected (0.074 sec)
```

Gambar 12 Pengisian Data Tabel “transaksi”

3. Tampilkan hasil transaksi pembelian dengan mengetikan perintah berikut

```
SELECT * FROM transaksi;
```



```
Command Prompt - mysql -u root
MariaDB [supplier_industri]> SELECT * FROM transaksi;
```

pembelianID	buyerID	barangID	metodeID	tgl_pembelian
4001	1	3001	1001	2021-08-16 12:40:16
4002	9	3007	1002	2021-08-16 12:40:27
4003	6	3003	1001	2021-08-16 12:40:38
4004	5	3005	1001	2021-08-16 12:41:01
4005	5	3007	1002	2021-08-16 12:41:09
4006	3	3010	1001	2021-08-16 12:41:17
4007	7	3002	1001	2021-08-16 12:41:28
4008	8	3001	1001	2021-08-16 12:41:41
4009	9	3005	1002	2021-08-16 12:41:52
4010	10	3004	1002	2021-08-16 12:42:02
4011	1	3010	1001	2021-08-16 12:39:45
4012	2	3009	1002	2021-08-16 12:39:45
4013	3	3008	1001	2021-08-16 12:39:45
4014	4	3007	1002	2021-08-16 12:39:45
4015	5	3006	1001	2021-08-16 12:39:45
4016	6	3005	1002	2021-08-16 12:39:45
4017	7	3004	1002	2021-08-16 12:39:45
4018	8	3003	1001	2021-08-16 12:39:45
4019	9	3002	1002	2021-08-16 12:39:45
4020	10	3001	1002	2021-08-16 12:39:45
4021	1	3001	1001	2021-08-16 12:43:51
4022	2	3009	1002	2021-08-16 12:43:51
4023	3	3009	1001	2021-08-16 12:43:51
4024	4	3007	1002	2021-08-16 12:43:51
4025	5	3006	1001	2021-08-16 12:43:51
4026	6	3006	1002	2021-08-16 12:43:51
4027	7	3004	1002	2021-08-16 12:43:51
4028	8	3008	1001	2021-08-16 12:43:51
4029	9	3003	1002	2021-08-16 12:43:51
4030	10	3002	1002	2021-08-16 12:43:51
4031	1	3010	1002	2021-08-16 12:47:52
4032	2	3009	1002	2021-08-16 12:47:52
4033	3	3008	1001	2021-08-16 12:47:52
4034	4	3007	1002	2021-08-16 12:47:52
4035	5	3006	1001	2021-08-16 12:47:52
4036	6	3005	1002	2021-08-16 12:47:52
4037	7	3004	1002	2021-08-16 12:47:52
4038	8	3003	1001	2021-08-16 12:47:52
4039	9	3002	1002	2021-08-16 12:47:52
4040	10	3001	1002	2021-08-16 12:47:52

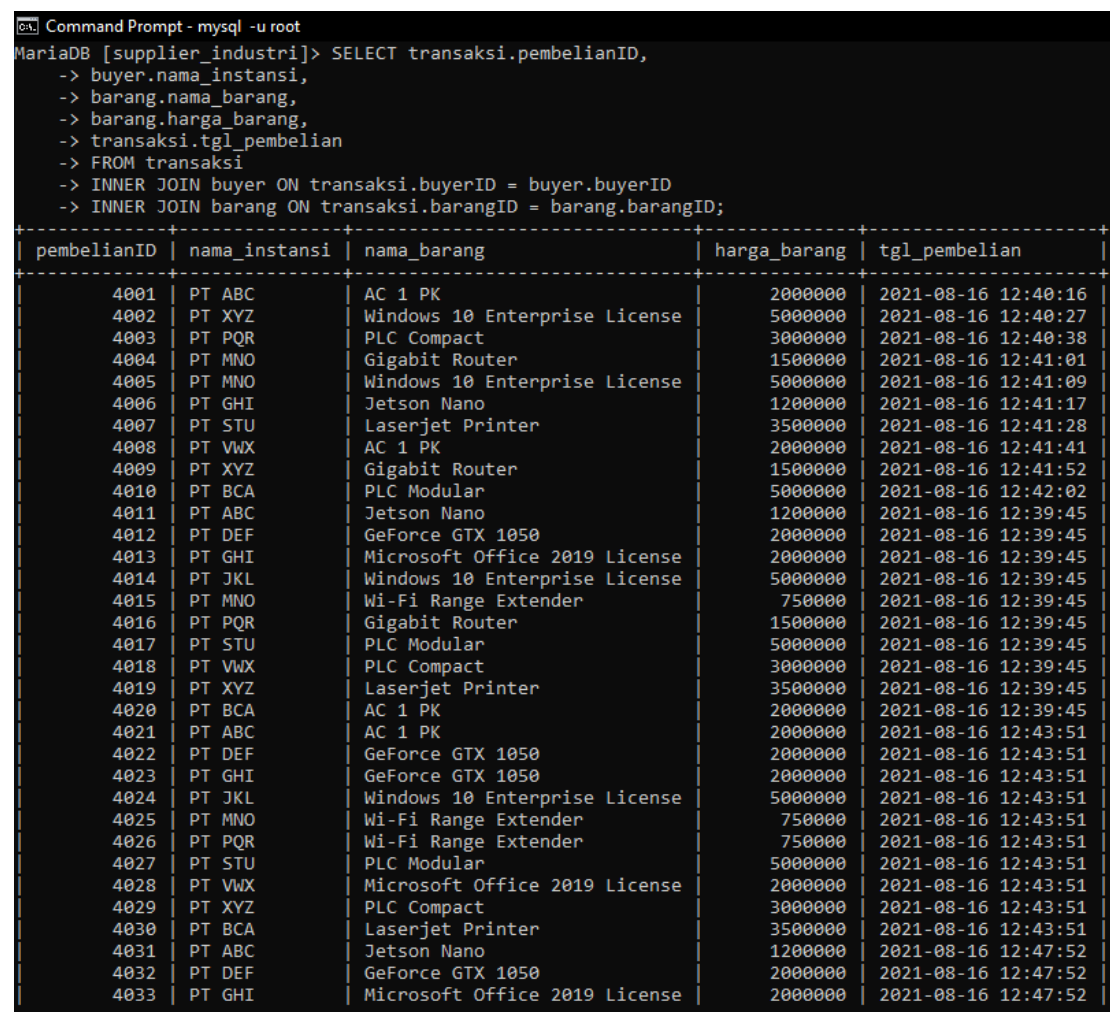
Gambar 13 Penampilan Isi Data Tabel “transaksi”

F. Menyajikan Data dengan Perintah INNER JOIN

1. Untuk dapat menyajikan data hasil transaksi lebih informatif, kita dapat menggunakan perintah INNER JOIN dengan mengetikkan sebagai berikut

```
SELECT transaksi.pembelianID, buyer.nama_instansi, barang.nama_barang,  
barang.harga_barang, transaksi.tgl_pembelian, buyer.alamat,  
metode_pembayaran.jenis_metode FROM transaksi INNER JOIN buyer ON  
transaksi.buyerID = buyer.buyerID INNER JOIN barang ON transaksi.barangID =  
barang.barangID INNER JOIN metode_pembayaran ON transaksi.metodeID =  
metode_pembayaran.metodeID;
```

2. Didapatkan hasil penyajian data sebagai berikut



Command Prompt - mysql -u root

MariaDB [supplier_industri]> SELECT transaksi.pembelianID,
-> buyer.nama_instansi,
-> barang.nama_barang,
-> barang.harga_barang,
-> transaksi.tgl_pembelian
-> FROM transaksi
-> INNER JOIN buyer ON transaksi.buyerID = buyer.buyerID
-> INNER JOIN barang ON transaksi.barangID = barang.barangID;

pembelianID	nama_instansi	nama_barang	harga_barang	tgl_pembelian
4001	PT ABC	AC 1 PK	2000000	2021-08-16 12:40:16
4002	PT XYZ	Windows 10 Enterprise License	5000000	2021-08-16 12:40:27
4003	PT PQR	PLC Compact	3000000	2021-08-16 12:40:38
4004	PT MNO	Gigabit Router	1500000	2021-08-16 12:41:01
4005	PT MNO	Windows 10 Enterprise License	5000000	2021-08-16 12:41:09
4006	PT GHI	Jetson Nano	1200000	2021-08-16 12:41:17
4007	PT STU	Laserjet Printer	3500000	2021-08-16 12:41:28
4008	PT VWX	AC 1 PK	2000000	2021-08-16 12:41:41
4009	PT XYZ	Gigabit Router	1500000	2021-08-16 12:41:52
4010	PT BCA	PLC Modular	5000000	2021-08-16 12:42:02
4011	PT ABC	Jetson Nano	1200000	2021-08-16 12:39:45
4012	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:39:45
4013	PT GHI	Microsoft Office 2019 License	2000000	2021-08-16 12:39:45
4014	PT JKL	Windows 10 Enterprise License	5000000	2021-08-16 12:39:45
4015	PT MNO	Wi-Fi Range Extender	750000	2021-08-16 12:39:45
4016	PT PQR	Gigabit Router	1500000	2021-08-16 12:39:45
4017	PT STU	PLC Modular	5000000	2021-08-16 12:39:45
4018	PT VWX	PLC Compact	3000000	2021-08-16 12:39:45
4019	PT XYZ	Laserjet Printer	3500000	2021-08-16 12:39:45
4020	PT BCA	AC 1 PK	2000000	2021-08-16 12:39:45
4021	PT ABC	AC 1 PK	2000000	2021-08-16 12:43:51
4022	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:43:51
4023	PT GHI	GeForce GTX 1050	2000000	2021-08-16 12:43:51
4024	PT JKL	Windows 10 Enterprise License	5000000	2021-08-16 12:43:51
4025	PT MNO	Wi-Fi Range Extender	750000	2021-08-16 12:43:51
4026	PT PQR	Wi-Fi Range Extender	750000	2021-08-16 12:43:51
4027	PT STU	PLC Modular	5000000	2021-08-16 12:43:51
4028	PT VWX	Microsoft Office 2019 License	2000000	2021-08-16 12:43:51
4029	PT XYZ	PLC Compact	3000000	2021-08-16 12:43:51
4030	PT BCA	Laserjet Printer	3500000	2021-08-16 12:43:51
4031	PT ABC	Jetson Nano	1200000	2021-08-16 12:47:52
4032	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:47:52
4033	PT GHI	Microsoft Office 2019 License	2000000	2021-08-16 12:47:52

3. Kita dapat menyajikan data dengan variasi perintah lain, misalnya mengurutkan data berdasarkan abjad pada field nama instansi, maka kita bisa ketikkan perintah sebagai berikut

```
ORDER BY buyer.nama_instansi;
```

- Didapatkan penyajian data sebagai berikut, terlihat bahwa data disajikan berdasarkan abjad nama instansi

Command Prompt - mysql -u root

```

MariaDB [supplier_industri]> SELECT transaksi.pembelianID,
-> buyer.nama_instansi,
-> barang.nama_barang,
-> barang.harga_barang,
-> transaksi.tgl_pembelian
-> FROM transaksi
-> INNER JOIN buyer ON transaksi.buyerID = buyer.buyerID
-> INNER JOIN barang ON transaksi.barangID = barang.barangID
-> ORDER BY buyer.nama_instansi;

```

pembelianID	nama_instansi	nama_barang	harga_barang	tgl_pembelian
4041	PT ABC	AC 1 PK	2000000	2021-08-16 12:49:53
4081	PT ABC	Microsoft Office 2019 License	2000000	2021-08-16 12:56:32
4001	PT ABC	AC 1 PK	2000000	2021-08-16 12:40:16
4051	PT ABC	Wi-Fi Range Extender	750000	2021-08-16 12:51:07
4091	PT ABC	PLC Compact	3000000	2021-08-16 12:56:32
4011	PT ABC	Jetson Nano	1200000	2021-08-16 12:39:45
4061	PT ABC	Laserjet Printer	3500000	2021-08-16 12:56:32
4021	PT ABC	AC 1 PK	2000000	2021-08-16 12:43:51
4071	PT ABC	GeForce GTX 1050	2000000	2021-08-16 12:56:32
4031	PT ABC	Jetson Nano	1200000	2021-08-16 12:47:52
4060	PT BCA	Jetson Nano	1200000	2021-08-16 12:51:07
4100	PT BCA	AC 1 PK	2000000	2021-08-16 12:56:32
4020	PT BCA	AC 1 PK	2000000	2021-08-16 12:39:45
4070	PT BCA	Windows 10 Enterprise License	5000000	2021-08-16 12:56:32
4030	PT BCA	Laserjet Printer	3500000	2021-08-16 12:43:51
4040	PT BCA	AC 1 PK	2000000	2021-08-16 12:47:52
4080	PT BCA	AC 1 PK	2000000	2021-08-16 12:56:32
4050	PT BCA	Gigabit Router	1500000	2021-08-16 12:49:53
4090	PT BCA	Wi-Fi Range Extender	750000	2021-08-16 12:56:32
4010	PT BCA	PLC Modular	5000000	2021-08-16 12:42:02
4052	PT DEF	Wi-Fi Range Extender	750000	2021-08-16 12:51:07
4092	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:56:32
4012	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:39:45
4062	PT DEF	Laserjet Printer	3500000	2021-08-16 12:56:32
4022	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:43:51
4072	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:56:32
4032	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:47:52
4042	PT DEF	AC 1 PK	2000000	2021-08-16 12:49:53
4082	PT DEF	GeForce GTX 1050	2000000	2021-08-16 12:56:32
4073	PT GHI	AC 1 PK	2000000	2021-08-16 12:56:32
4105	PT GHI	GeForce GTX 1050	2000000	2021-08-16 13:01:46
4033	PT GHI	Microsoft Office 2019 License	2000000	2021-08-16 12:47:52

Gambar 14 Penyajian Isi Data Tabel “transaksi” Berdasarkan Abjad

G. Ekspor File Database

Berikut ini adalah langkah-langkah untuk mengekspor database dalam ekstensi .sql

- Masuk ke phpMyAdmin
- Klik pada database yang ingin kita ekspor
- Klik tab “Export”
- Klik “Go”
- File database dengan ekstensi .sql akan otomatis terdownload

H. Push File .sql ke GitHub menggunakan Gitbash

Berikut ini adalah langkah-langkah untuk push file database ke dalam GitHub dengan menggunakan aplikasi GitBash.

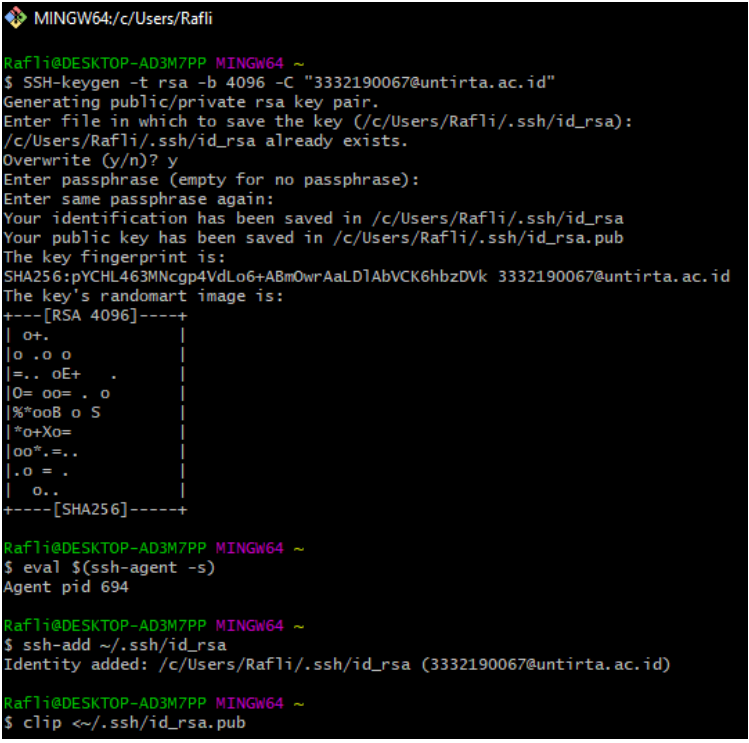
1. Langkah pertama yaitu membuat public key agar dapat terhubung antara Git Bash dengan Github. Ketikkan perintah berikut pada Git Bash

```
hm$ SSH-keygen -t rsa -b 4096 -C "3332190067@untirta.ac.id"

$ eval $(ssh-agent -s)

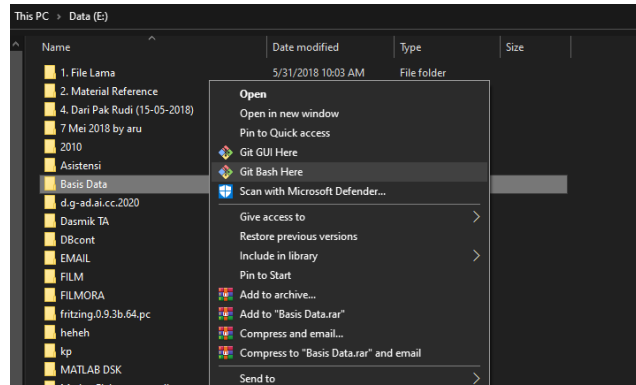
$ ssh-add ~/.ssh/id_rsa

$ clip <~/.ssh/id_rsa.pub
```

A screenshot of a Windows terminal window titled 'MINGW64: c/Users/Rafli'. The user 'Rafli' is at the prompt. They enter the command 'SSH-keygen -t rsa -b 4096 -C "3332190067@untirta.ac.id"'. The terminal shows the process of generating a public/private RSA key pair. It prompts for a file name (defaulting to /c/Users/Rafli/.ssh/id_rsa), asks if it exists to overwrite (y/n), and prompts for a passphrase (empty for no passphrase). It then shows the key fingerprint (SHA256) and a randomart image. After pressing 'y' for overwrite and leaving the passphrase empty, it shows the key fingerprint again. Then, the user enters 'eval \$(ssh-agent -s)' and the terminal shows 'Agent pid 694'. Next, the user enters 'ssh-add ~/.ssh/id_rsa' and the terminal shows 'Identity added: /c/Users/Rafli/.ssh/id_rsa (3332190067@untirta.ac.id)'. Finally, the user enters 'clip <~/.ssh/id_rsa.pub'.

Gambar 15 Generate SSH Public Key

2. Salin public key yang telah degenerate pada menu “SSH and GPG Key” akun Github.
3. Setelah memiliki SSH key, kita dapat melakukan push dan remote file databse kita. Klik kanan pada folder tempan menyimpan database, dan klik “Git Bash Here”



Gambar 16 Push File Database

4. Akan muncul window terminal Git Bash baru. Ketikkan perintah berikut

```
$ git config --global user.name "itsjusttrafli"
$ git config --global user.email "3332190067@untirta.ac.id"
$ git config --list
```

Username dan user email diisi sesuai dengan akun GitHub kita

5. Lalu ketikkan perintah berikut untuk file yang akan kita push

```
$ git init
$ git add db_perpustakaan.sql
$ git commit -m "Berikut ini adalah file SQL Database yang telah dibuat"
$ git remote add origin git@github.com:itsjusttrafli/basis_data.git
$ git push -u origin master
```

```
MINGW64/c/Users/Rafli
Rafli@DESKTOP-AD3M7PP MINGW64 ~
$ git config --global user.name "itsjusttrafli"
Rafli@DESKTOP-AD3M7PP MINGW64 ~
$ git config --global user.email "3332190067@untirta.ac.id"
Rafli@DESKTOP-AD3M7PP MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=itsjusttrafli
user.email=3332190067@untirta.ac.id
Rafli@DESKTOP-AD3M7PP MINGW64 ~
$ sds
bash: sds: command not found
```

```

MINGW64:/e/Basis Data

Rafli@DESKTOP-AD3M7PP MINGW64 /e/Basis Data
$ git init
Initialized empty Git repository in E:/Basis Data/.git/

Rafli@DESKTOP-AD3M7PP MINGW64 /e/Basis Data (master)
$ git add supplier_industri.sql
warning: LF will be replaced by CRLF in supplier_industri.sql.
The file will have its original line endings in your working directory

Rafli@DESKTOP-AD3M7PP MINGW64 /e/Basis Data (master)
$ git commit -m "Ini adalah file SQL Database yang telah dibuat"
[master (root-commit) 7a0cec3] Ini adalah file SQL Database yang telah dibuat
1 file changed, 327 insertions(+)
create mode 100644 supplier_industri.sql

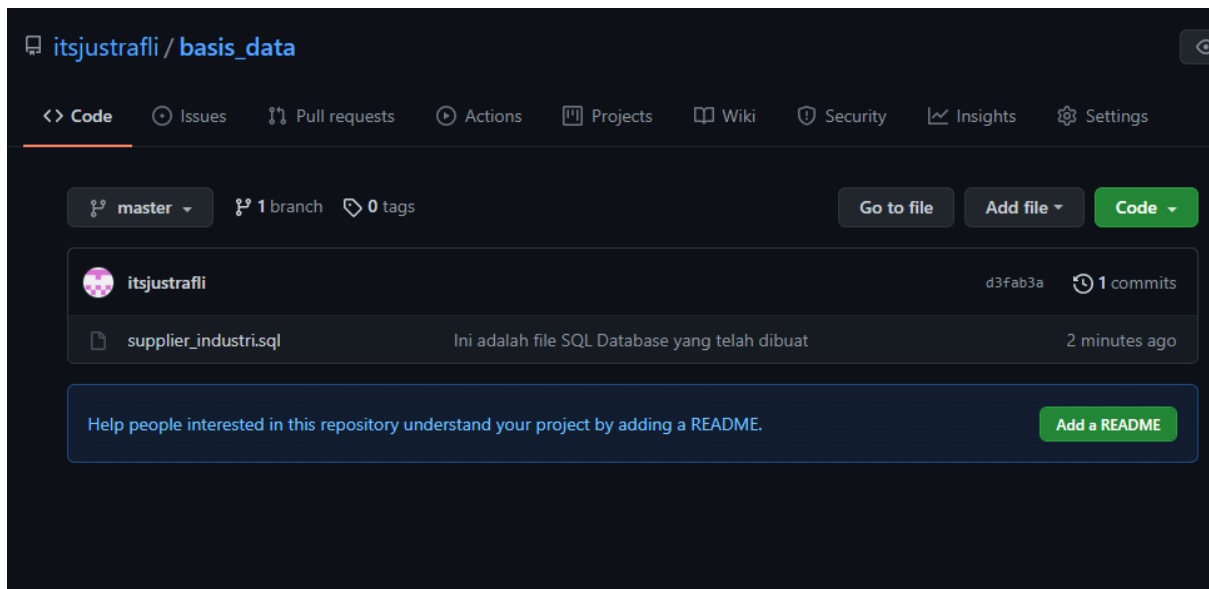
Rafli@DESKTOP-AD3M7PP MINGW64 /e/Basis Data (master)
$ git remote add origin git@github.com:itsjustrafli/basis_data.git

Rafli@DESKTOP-AD3M7PP MINGW64 /e/Basis Data (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 2.36 KiB | 2.36 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:itsjustrafli/basis_data.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

```

Gambar 18 Log Berhasil Push File

6. Berikut ini adalah file database yang sudah dipush pada GitHub.



Gambar 19 Tampilan File Database Sudah dipush Ke GitHub

Video Penjelasan YouTube : <https://youtu.be/YkjRAuWhH4s>

Repository GitHub : [itsjusttrafli/basis_data](https://github.com/itsjusttrafli/basis_data): Repo untuk menaruh file tugas akhir matakuliah Basis Data (github.com)