Laporan Praktikum

Sistem Basis Data Relasional

Periode I

Disusun Oleh :

Adam Rahmat Ilahi (13.2021.1.01030)

JURUSAN SISTEM INFORMASI

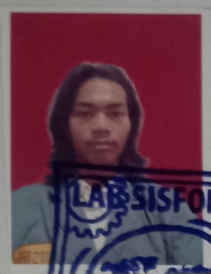
FAKULTAS TEKNIK ELEKTRO DAN TEKNOLOGI INFORMASI

INSTITUT TEKNOLOGI ADHI TAMA SURABAYA

2022/2023

HASIL SCAN KARTU

KARTU PRAKTIKUM
SISTEM BASIS DATA RELASIONAL
TAHUN AJARAN 2022/2023



NPM : 13.2021.1.01030
Nama : ADAM RAHMAT ILAHI
Kelas : P1 / Pagi
No. Telp : +62 856 4364 5402
Dosen Pengampu : SULISTYOWATI
Asisten Laboratorium : Azzah

*Stempel pada foto, bukti bahwa praktikan telah melunasi pembayaran praktikum

Pertemuan	Tanggal	Asisten Laboratorium	Dosen Pengampu
1	6 - 4 - 2023		
2	6 - 4 - 2023		
3	6 - 4 - 2023		
4	20 - 4 - 2023		
5	20 - 4 - 2023		
6	20 - 4 - 2023		
7	20 - 4 - 2023		
8	UTS		
9	8 - 6 - 2023		
10	8 - 6 - 2023		
11	8 - 6 - 2023		
12	15 - 6 - 2023		
13	15 - 6 - 2023		
14	15 - 6 - 2023		
15	15 - 6 - 2023		
16	UAS		

Mengetahui,
Ketua Lab. Algoritma Pemrograman dan Basis Data

Budanis Dwi Meilani, S.T., M.Kom.
NIP. 981090

MODUL 1

PRAKTIKUM MODUL 1

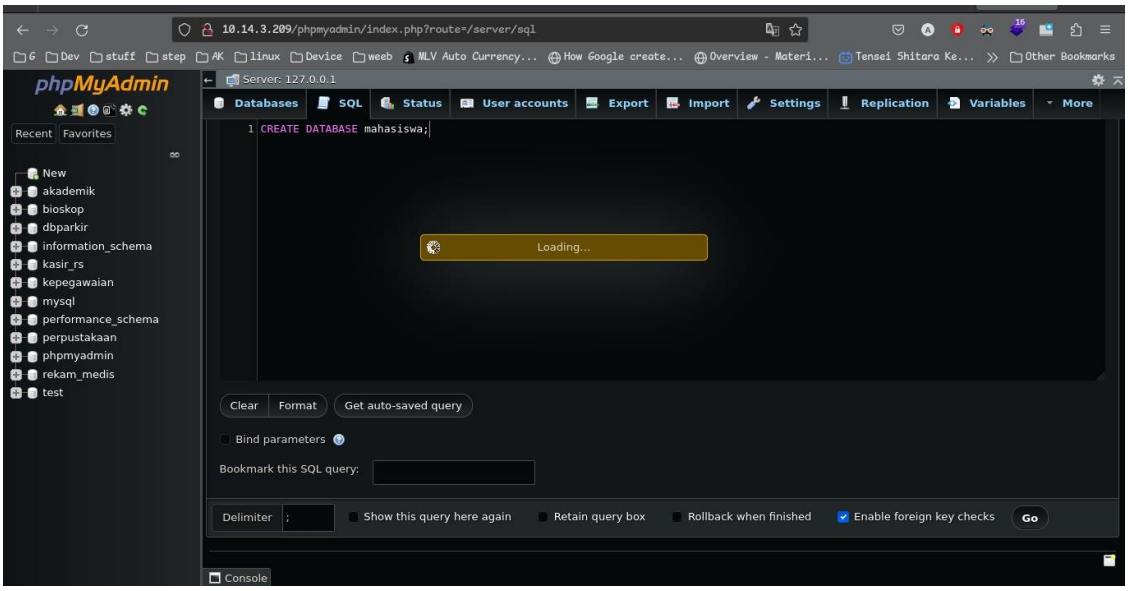
Pastikan configurasi xampp berada pada Require all granted.

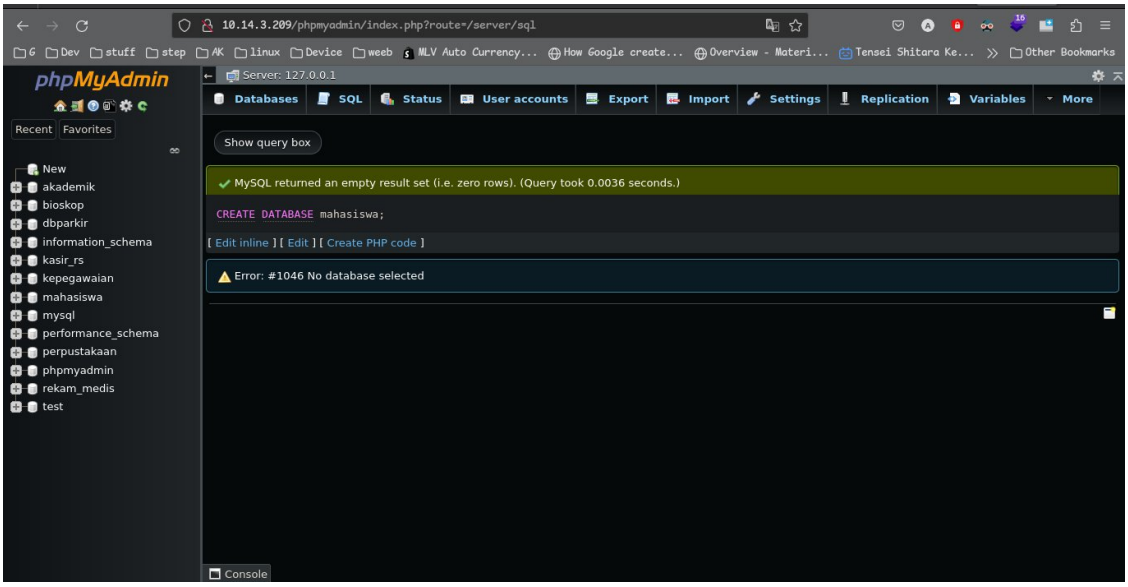
```
kitty

#<IfDefine PHP4>
#LoadModule php4_module        modules/libphp4.so
#</IfDefine>
#<IfDefine PHP7>
#LoadModule php7_module        modules/libphp7.so
#</IfDefine>

# We will enable it by default
#<IfDefine PHP>
   LoadModule php7_module        modules/libphp7.so
#</IfDefine>

LoadModule perl_module        modules/mod_perl.so

Alias /phpmyadmin "/opt/lampp/phpmyadmin"

# since XAMPP 1.4.3
<Directory "/opt/lampp/phpmyadmin">
    AllowOverride AuthConfig Limit
    #Require local
    Require all granted
    ErrorDocument 403 /error/XAMPP_FORBIDDEN.html.var
</Directory>

# since LAMPP 1.0RC1
AddType application/x-httpd-php .php .php3 .php4

XBitHack on

# since 0.9.8 we've mod_perl
<IfModule mod_perl.c>
        AddHandler perl-script .pl
    PerlHandler ModPerl::PerlRunPrefork
    PerlOptions +ParseHeaders
        PerlSendHeader On
</IfModule>
/opt/lampp/etc/extra/httpd-xampp.conf [RO]
```

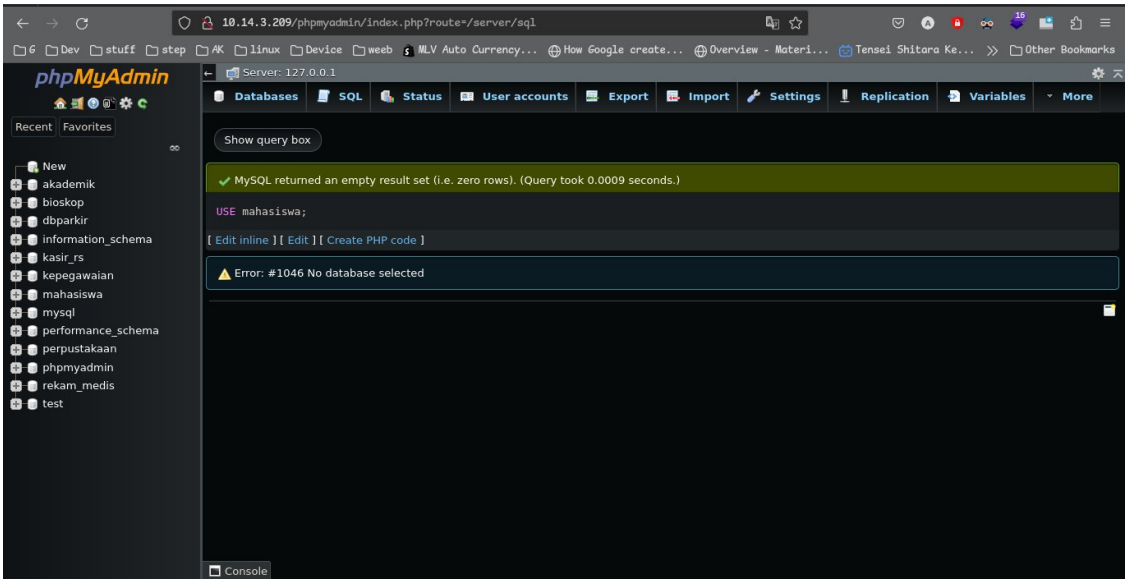
Pengecekan ip device untuk di bagikan kepada teman yang sekelompok, pastikan koneksi device terhubung pada 1 jaringan yang sama.

```
kitty

└> sudo /opt/lampp/xampp start
[sudo] password for adam:
Starting XAMPP for Linux 7.4.29-1...
XAMPP: Starting Apache...ok.
XAMPP: Starting MySQL...ok.
XAMPP: Starting ProFTPD...ok.
┌adam@adam in ~ took 2s
└> ifconfig
enp1s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether b0:5a:da:3e:72:41  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 164  bytes 13732 (13.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 164  bytes 13732 (13.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.14.3.48  netmask 255.255.255.0  broadcast 10.14.3.255
        inet6 fe80::2326:3d9d:1bdc:e26b  prefixlen 64  scopeid 0x20<link>
        ether 34:f3:9a:ed:b9:21  txqueuelen 1000  (Ethernet)
        RX packets 1429  bytes 1145656 (1.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 694  bytes 79563 (77.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

┌adam@adam in ~ took 14ms
└>
```

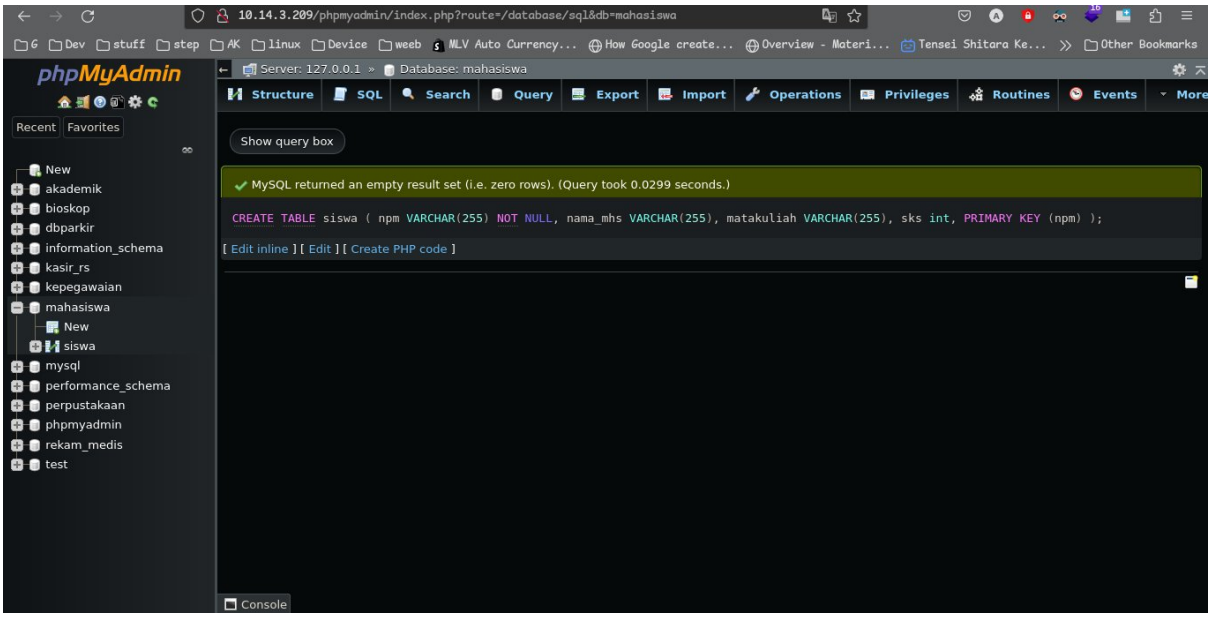
Buka ip device teman sekelompok, dan buat database bernama mahasiswa.



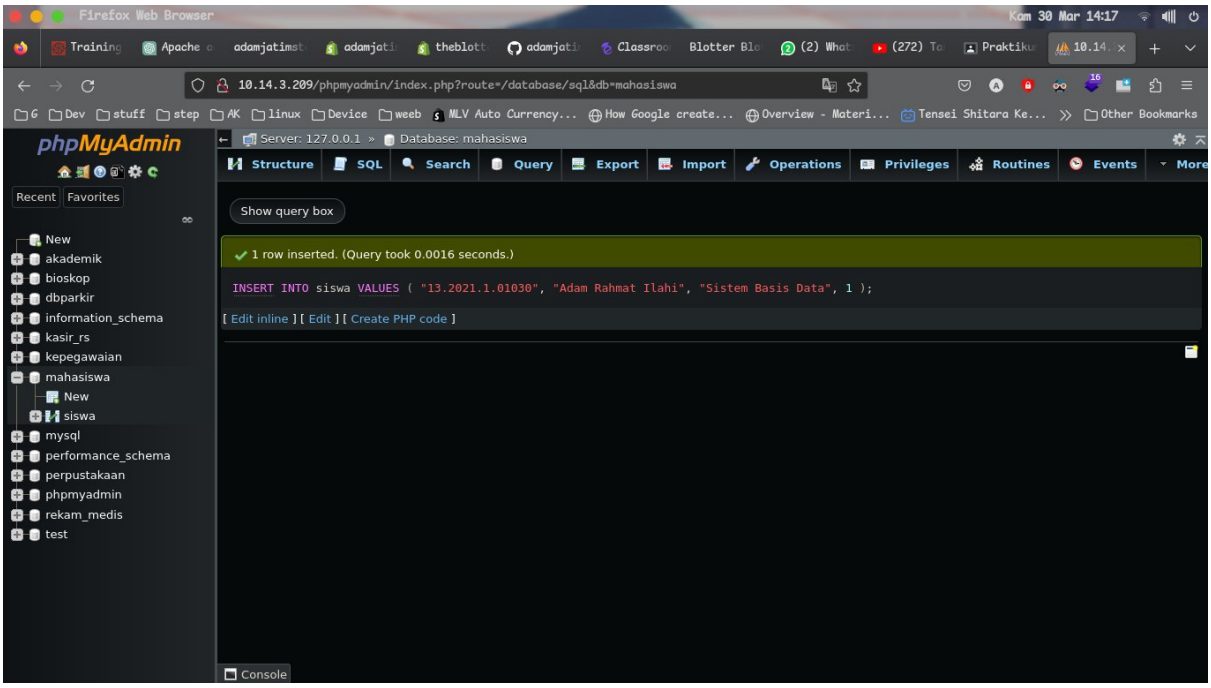




Buat table siswa dengan atribut nama, npm, matakuliah, sks.



Melakukan input data mandiri kedalam table tersebut.



MODUL 2

PRAKTIKUM MODUL 2

Pertama-tama download file.sql pada link yang sudah disediakan oleh ASLAB
https://drive.google.com/file/d/1vrGVTS4KicyVLb7PQdSjW_QI-
Lalu tambahkan perintah:
CREATE DATABASE IF NOT EXISTS bimbel;
USE bimbel;

```
24 CREATE DATABASE IF NOT EXISTS bimbel;
25 USE bimbel;
```

Setelah itu Save file.sql dan import file ke phpmyadmin.

```
MariaDB [restoran_reservasi]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| MAHASISWA          |
| auth_db            |
| bimbel             |
| crud_db            |
| information_schema |
| kepegawaian        |
| mysql              |
| performance_schema |
| perpustakaan       |
| phpmyadmin         |
| rawat_inap         |
| restoran_reservasi |
| test               |
+--------------------+
13 rows in set (0.001 sec)

MariaDB [restoran_reservasi]> USE bimbel;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [bimbel]>
```

```
Database changed
MariaDB [bimbel]> SHOW TABLES;
+-------------------+
| Tables_in_bimbel |
+-------------------+
| mapel             |
| mengambil         |
| paket             |
| pembayaran        |
| pengajar          |
| siswa             |
| terdapat          |
+-------------------+
7 rows in set (0.001 sec)

MariaDB [bimbel]>
```

SOAL PRAKTIKUM MODUL 2

1. Tampilkan nama pengajar yang mengambil mapel bahasa indonesia

   SELECT nama_pengajar, nama_mapel FROM pengajar p, mapel m WHERE p.kode_mapel = m.kode_mapel AND m.nama_mapel = ‘bahasa indonesia’

```
MariaDB [bimbel]> SELECT nama_pengajar,
+---------------+------------------+
| nama_pengajar | nama_mapel       |
+---------------+------------------+
| emi           | bahasa indonesia |
| ani           | bahasa indonesia |
+---------------+------------------+
2 rows in set (0.001 sec)
```

2. Tambahkan data pada tabel siswa (isi terserah)

   INSERT INTO siswa VALUES (121, ‘fuad’, 081234567890, ‘bulak banteng’, ‘sma 05’);

```
MariaDB [bimbel]> SELECT * FROM siswa;
+----------+------------+--------------+-----------+--------------+
| id_siswa | nama_siswa | no_hp        | alamat    | asal_Sekolah |
+----------+------------+--------------+-----------+--------------+
| 112      | slamet     | 89143276433  | diryorejo | sma 05       |
| 113      | birsa      | 089143276423 | ampel     | sma 03       |
| 114      | brend      | 089143276424 | ambengan  | sma 04       |
| 115      | frendi     | 089143276425 | perak     | sma 03       |
| 116      | ferdi      | 089143276426 | ampel     | sma 02       |
| 117      | nisa       | 089143276427 | ambengan  | sma 03       |
| 118      | narzo      | 089143276428 | sidotopo  | sma 01       |
| 119      | sakura     | 089143276429 | perak     | sma 02       |
| 120      | sasuning   | 089143276430 | sidotopo  | sma 01       |
+----------+------------+--------------+-----------+--------------+
9 rows in set (0.001 sec)

MariaDB [bimbel]> INSERT INTO siswa VALUES (121, 'fuad', 081234567890, 'bulak banteng', 'sma 05');
Query OK, 1 row affected (0.009 sec)

MariaDB [bimbel]> SELECT * FROM siswa;
+----------+------------+--------------+---------------+--------------+
| id_siswa | nama_siswa | no_hp        | alamat        | asal_Sekolah |
+----------+------------+--------------+---------------+--------------+
| 112      | slamet     | 89143276433  | diryorejo     | sma 05       |
| 113      | birsa      | 089143276423 | ampel         | sma 03       |
| 114      | brend      | 089143276424 | ambengan      | sma 04       |
| 115      | frendi     | 089143276425 | perak         | sma 03       |
| 116      | ferdi      | 089143276426 | ampel         | sma 02       |
| 117      | nisa       | 089143276427 | ambengan      | sma 03       |
| 118      | narzo      | 089143276428 | sidotopo      | sma 01       |
| 119      | sakura     | 089143276429 | perak         | sma 02       |
| 120      | sasuning   | 089143276430 | sidotopo      | sma 01       |
| 121      | fuad       | 81234567890  | bulak banteng | sma 05       |
+----------+------------+--------------+---------------+--------------+
10 rows in set (0.001 sec)

MariaDB [bimbel]>
```

3. Hapus salah satu data pada tabel siswa (data terserah)

   DELETE FROM siswa WHERE id_siswa=121;

```
MariaDB [bimbel]> delete from siswa where id_siswa=121;
Query OK, 1 row affected (0.008 sec)

MariaDB [bimbel]> select * from siswa
    -> ;
+----------+------------+--------------+-----------+--------------+
| id_siswa | nama_siswa | no_hp        | alamat    | asal_Sekolah |
+----------+------------+--------------+-----------+--------------+
| 112      | slamet     | 89143276433  | diryorejo | sma 05       |
| 113      | birsa      | 089143276423 | ampel     | sma 03       |
| 114      | brend      | 089143276424 | ambengan  | sma 04       |
| 115      | frendi     | 089143276425 | perak     | sma 03       |
| 116      | ferdi      | 089143276426 | ampel     | sma 02       |
| 117      | nisa       | 089143276427 | ambengan  | sma 03       |
| 118      | narzo      | 089143276428 | sidotopo  | sma 01       |
| 119      | sakura     | 089143276429 | perak     | sma 02       |
| 120      | sasuning   | 089143276430 | sidotopo  | sma 01       |
+----------+------------+--------------+-----------+--------------+
9 rows in set (0.000 sec)

MariaDB [bimbel]>
```

4. Update atau rubah salah satu data pada tabel pengajar (data terserah)

   UPDATE pengajar SET lulusan='itats' WHERE id_pengajar='p1';

```
' at line 1
MariaDB [bimbel]> SELECT * FROM pengajar;
+-------------+---------------+--------------+-------------+----------+------------+
| id_pengajar | nama_pengajar | no_hp        | alamat      | lulusan  | kode_mapel |
+-------------+---------------+--------------+-------------+----------+------------+
| b1          | emi           | 089143276450 | pucung      | ubaya    | b01        |
| b2          | ani           | 089143276451 | kupang      | itats    | b02        |
| e1          | putut         | 089143276453 | kenjeran    | untag    | e01        |
| e2          | pitut         | 089143276454 | sidotopo    | stikom   | e02        |
| i1          | firli         | 089143276452 | gunung sari | narotama | i01        |
| m1          | dikun         | 089143276458 | ambengan    | its      | m01        |
| m2          | nurkan        | 089143276459 | kenjeran    | unesa    | m02        |
| p1          | irma          | 089143276444 | ampel       | narotama | p01        |
+-------------+---------------+--------------+-------------+----------+------------+
8 rows in set (0.000 sec)

MariaDB [bimbel]> UPDATE pengajar SET lulusan='itats' WHERE id_pengajar='p1';
Query OK, 1 row affected (0.009 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [bimbel]> SELECT * FROM pengajar;
+-------------+---------------+--------------+-------------+----------+------------+
| id_pengajar | nama_pengajar | no_hp        | alamat      | lulusan  | kode_mapel |
+-------------+---------------+--------------+-------------+----------+------------+
| b1          | emi           | 089143276450 | pucung      | ubaya    | b01        |
| b2          | ani           | 089143276451 | kupang      | itats    | b02        |
| e1          | putut         | 089143276453 | kenjeran    | untag    | e01        |
| e2          | pitut         | 089143276454 | sidotopo    | stikom   | e02        |
| i1          | firli         | 089143276452 | gunung sari | narotama | i01        |
| m1          | dikun         | 089143276458 | ambengan    | its      | m01        |
| m2          | nurkan        | 089143276459 | kenjeran    | unesa    | m02        |
| p1          | irma          | 089143276444 | ampel       | itats    | p01        |
+-------------+---------------+--------------+-------------+----------+------------+
8 rows in set (0.000 sec)

MariaDB [bimbel]>
```

MODUL 3

PRAKTIKUM MODUL 3

Pertama membuat Database bernama apotek

```sql
CREATE DATABASE IF NOT EXISTS apotek;
USE apotek;

-- CREATE TABLE

CREATE TABLE defecta (
  id_defecta INT NOT NULL,
  tanggal DATE,
  keterangan VARCHAR(255),
  PRIMARY KEY (id_defecta)
);

CREATE TABLE log_aktivitas (
  id_log INT,
  tanggal DATE,
  aktivitas VARCHAR(255),
  PRIMARY KEY (id_log)
);

CREATE TABLE sisa_stok_barang (
  id_obat INT,
  nama_obat VARCHAR(255),
  stok_obat INT,
  PRIMARY KEY (id_obat)
);

CREATE TABLE obat_keluar (
  id_obat_keluar INT,
  tanggal DATE,
  id_obat INT,
  jumlah INT,
  PRIMARY KEY (id_obat_keluar),
  FOREIGN KEY (id_obat) REFERENCES sisa_stok_barang (id_obat)
);

CREATE TABLE penerimaan_barang (
  id_penerimaan_barang INT,
  tanggal DATE,
  id_obat INT,
```

```sql
  jumlah INT,
  keterangan VARCHAR(255),
  PRIMARY KEY (id_penerimaan_barang),
  FOREIGN KEY (id_obat) REFERENCES sisa_stok_barang (id_obat)
);

CREATE TABLE penerimaan_resep (
  id_penerimaan_resep INT,
  tanggal DATE,
  resep VARCHAR(255),
  id_obat INT,
  jumlah INT,
  nama_pasien VARCHAR(255),
  nama_dokter VARCHAR(255),
  PRIMARY KEY (id_penerimaan_resep),
  FOREIGN KEY (id_obat) REFERENCES sisa_stok_barang (id_obat)
);

-- INPUT DATA

INSERT INTO sisa_stok_barang VALUES
(1, 'Paracetamol', 50),
(2, 'Amoxicillin', 20),
(3, 'Loratadine', 30),
(4, 'Omeprazole', 10),
(5, 'Aspirin', 40),
(6, 'Metformin', 25),
(7, 'Atorvastatin', 15),
(8, 'Simvastatin', 30),
(9, 'Ciprofloxacin', 20),
(10, 'Metronidazole', 35),
(11, 'Azithromycin', 25),
(12, 'Diazepam', 15),
(13, 'Furosemide', 20),
(14, 'Ibuprofen', 40),
(15, 'Losartan', 10),
(16, 'Naproxen', 30),
(17, 'Prednisone', 25),
(18, 'Propranolol', 20),
(19, 'Sildenafil', 10),
(20, 'Tramadol', 15);
```

```sql
INSERT INTO defecta (id_defecta, tanggal, keterangan)
VALUES
(1, '2023-05-01', 'Obat cacat'),
(2, '2023-05-03', 'Kemasan rusak'),
(3, '2023-05-05', 'Obat kadaluarsa'),
(4, '2023-05-08', 'Obat pecah'),
(5, '2023-05-10', 'Kadaluarsa'),
(6, '2023-05-11', 'Cacat'),
(7, '2023-05-12', 'Kemasan rusak'),
(8, '2023-05-15', 'Obat pecah'),
(9, '2023-05-17', 'Kemasan rusak'),
(10, '2023-05-20', 'Obat kadaluarsa'),
(11, '2023-05-21', 'Obat cacat'),
(12, '2023-05-23', 'Kemasan rusak'),
(13, '2023-05-26', 'Obat kadaluarsa'),
(14, '2023-05-29', 'Obat cacat'),
(15, '2023-05-30', 'Kemasan rusak');

INSERT INTO log_aktivitas (id_log, tanggal, aktivitas)
VALUES
(1, '2023-05-01', 'Masuk aplikasi'),
(2, '2023-05-02', 'Pencarian obat'),
(3, '2023-05-03', 'Penerimaan barang'),
(4, '2023-05-05', 'Penjualan obat'),
(5, '2023-05-07', 'Penerimaan resep'),
(6, '2023-05-09', 'Pencarian obat'),
(7, '2023-05-12', 'Penerimaan barang'),
(8, '2023-05-14', 'Penjualan obat'),
(9, '2023-05-17', 'Penerimaan resep'),
(10, '2023-05-19', 'Masuk aplikasi'),
(11, '2023-05-22', 'Pencarian obat'),
(12, '2023-05-24', 'Penerimaan barang'),
(13, '2023-05-27', 'Penjualan obat'),
(14, '2023-05-29', 'Penerimaan resep'),
(15, '2023-05-30', 'Pencarian obat');
```

```sql
INSERT INTO obat_keluar (id_obat_keluar, tanggal, id_obat, jumlah)
VALUES
(1, '2023-05-02', 1, 10),
(2, '2023-05-04', 3, 5),
(3, '2023-05-06', 2, 7),
(4, '2023-05-08', 4, 3),
(5, '2023-05-10', 5, 2),
(6, '2023-05-12', 7, 4),
(7, '2023-05-15', 10, 1),
(8, '2023-05-16', 6, 2),
(9, '2023-05-18', 11, 3),
(10, '2023-05-20', 9, 5),
(11, '2023-05-22', 12, 2),
(12, '2023-05-24', 13, 1),
(13, '2023-05-26', 15, 4),
(14, '2023-05-28', 14, 6),
(15, '2023-05-30', 8, 3);

INSERT INTO penerimaan_barang (id_penerimaan_barang, tanggal, id_obat, jumlah, keterangan)
VALUES
(1, '2023-05-03', 1, 20, 'Penerimaan dari supplier A'),
(2, '2023-05-05', 2, 15, 'Penerimaan dari supplier B'),
(3, '2023-05-07', 3, 12, 'Penerimaan dari supplier C'),
(4, '2023-05-10', 4, 5, 'Penerimaan dari supplier D'),
(5, '2023-05-12', 5, 10, 'Penerimaan dari supplier E'),
(6, '2023-05-15', 6, 8, 'Penerimaan dari supplier F'),
(7, '2023-05-18', 7, 6, 'Penerimaan dari supplier G'),
(8, '2023-05-21', 8, 11, 'Penerimaan dari supplier H'),
(9, '2023-05-23', 9, 9, 'Penerimaan dari supplier I'),
(10, '2023-05-26', 10, 14, 'Penerimaan dari supplier J'),
(11, '2023-05-29', 11, 7, 'Penerimaan dari supplier K'),
(12, '2023-05-31', 12, 3, 'Penerimaan dari supplier L');

INSERT INTO penerimaan_resep (id_penerimaan_resep, tanggal, id_obat, jumlah, nama_pasien, nama_dokter)
VALUES
(1, '2023-05-01', 1, 30, 'Siti', 'Dr. Andi'),
(2, '2023-05-03', 2, 20, 'Agus', 'Dr. Budi'),
(3, '2023-05-05', 3, 15, 'Rina', 'Dr. Cici'),
(4, '2023-05-07', 4, 10, 'Joko', 'Dr. Didi'),
(5, '2023-05-09', 5, 5, 'Ayu', 'Dr. Eko'),
(6, '2023-05-11', 6, 12, 'Diana', 'Dr. Fifi'),
(7, '2023-05-13', 7, 8, 'Hendra', 'Dr. Gigi'),
(8, '2023-05-15', 8, 7, 'Rafi', 'Dr. Wiwik'),
(9, '2023-05-17', 9, 15, 'Budi', 'Dr. Hadi'),
(10, '2023-05-19', 10, 25, 'Ani', 'Dr. Indra'),
(11, '2023-05-21', 11, 18, 'Rudi', 'Dr. Joko'),
(12, '2023-05-23', 12, 6, 'Sari', 'Dr. Kiki')
```

1. Buatlah procedure tanpa menggunakan paraameter

```sql
DELIMITER //
CREATE PROCEDURE show_stok()
BEGIN
    SELECT * FROM sisa_stok_barang;
END //
DELIMITER ;
```

```
MariaDB [apotek]> CALL show_stok();
+---------+---------------+-----------+
| id_obat | nama_obat     | stok_obat |
+---------+---------------+-----------+
|       1 | Paracetamol   |        50 |
|       2 | Amoxicillin   |        21 |
|       3 | Loratadine    |        30 |
|       4 | Omeprazole    |        10 |
|       5 | Aspirin       |        40 |
|       6 | Metformin     |        25 |
|       7 | Atorvastatin  |        15 |
|       8 | Simvastatin   |        30 |
|       9 | Ciprofloxacin |        20 |
|      10 | Metronidazole |        35 |
|      11 | Azithromycin  |        25 |
|      12 | Diazepam      |        15 |
|      13 | Furosemide    |        20 |
|      14 | Ibuprofen     |        40 |
|      15 | Losartan      |        10 |
|      16 | Naproxen      |        30 |
|      17 | Prednisone    |        25 |
|      18 | Propranolol   |        20 |
|      19 | Sildenafil    |        10 |
|      20 | Tramadol      |        15 |
+---------+---------------+-----------+
20 rows in set (0.009 sec)

Query OK, 0 rows affected (0.010 sec)
```

2. Buatlah sebuah procedure menggunakan parameter IN

```
DELIMITER //
CREATE PROCEDURE add_stok_obat(IN input_id INT, IN input_nama
VARCHAR(255), IN input_stok INT)
BEGIN
    INSERT INTO sisa_stok_barang (id_obat, nama_obat, stok_obat)
    VALUES (input_id, input_nama, input_stok);
END //
DELIMITER ;
```

```
MariaDB [apotek]> CALL add_stok_obat(21, 'Bondrex', 7);
Query OK, 1 row affected (0.010 sec)

MariaDB [apotek]> CALL show_stok();
+---------+--------------+-----------+
| id_obat | nama_obat    | stok_obat |
+---------+--------------+-----------+
|       1 | Paracetamol  |        50 |
|       2 | Amoxicillin  |        21 |
|       3 | Loratadine   |        30 |
|       4 | Omeprazole   |        10 |
|       5 | Aspirin      |        40 |
|       6 | Metformin    |        25 |
|       7 | Atorvastatin |        15 |
|       8 | Simvastatin  |        30 |
|       9 | Ciprofloxacin |       20 |
|      10 | Metronidazole |       35 |
|      11 | Azithromycin |        25 |
|      12 | Diazepam     |        15 |
|      13 | Furosemide   |        20 |
|      14 | Ibuprofen    |        40 |
|      15 | Losartan     |        10 |
|      16 | Naproxen     |        30 |
|      17 | Prednisone   |        25 |
|      18 | Propranolol  |        20 |
|      19 | Sildenafil   |        10 |
|      20 | Tramadol     |        15 |
|      21 | Bondrex      |         7 |
+---------+--------------+-----------+
21 rows in set (0.001 sec)

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

3. Buatlah sebuah procedure menggunakan parameter OUT

```
DELIMITER //
CREATE PROCEDURE show_jumlah_penerimaan_barang(IN input_tanggal DATE,
OUT out_jumlah INT)
BEGIN
    SELECT SUM(jumlah) INTO out_jumlah FROM penerimaan_barang WHERE
tanggal = input_tanggal;
END //
DELIMITER ;
```

```
MariaDB [apotek]> SET @jumlah_penerimaan = 0;
Query OK, 0 rows affected (0.000 sec)

MariaDB [apotek]>
MariaDB [apotek]> CALL show_jumlah_penerimaan_barang('2023-05-07', @jumlah_penerimaan);
Query OK, 1 row affected (0.000 sec)

MariaDB [apotek]>
MariaDB [apotek]> SELECT @jumlah_penerimaan;
+--------------------+
| @jumlah_penerimaan |
+--------------------+
|                 12 |
+--------------------+
1 row in set (0.000 sec)

MariaDB [apotek]>
```

4. Buatlah sebuah procedure menggunakan parameter INOUT

```
DELIMITER //
CREATE PROCEDURE kali(INOUT angka INT)
BEGIN
    SELECT (angka * angka) INTO angka;
END //

DELIMITER ;
```

```
MariaDB [apotek]> SET @angka = 5;
Query OK, 0 rows affected (0.000 sec)

MariaDB [apotek]> CALL kali(@angka);
Query OK, 1 row affected (0.000 sec)

MariaDB [apotek]> SELECT @angka;
+--------+
| @angka |
+--------+
|     25 |
+--------+
1 row in set (0.000 sec)

MariaDB [apotek]>
```

1. Buatlah progam function dengan menggunakan count

```
SELECT COUNT(nama_obat) FROM sisa_stok_barang;
```

```
MariaDB [apotek]> SELECT COUNT(nama_obat) FROM sisa_stok_barang;
+------------------+
| COUNT(nama_obat) |
+------------------+
|               21 |
+------------------+
1 row in set (0.007 sec)

MariaDB [apotek]>
```

2. Buatlah progam function dengan menggunakan max

```
SELECT nama_obat, MAX(stok_obat) FROM sisa_stok_barang;
```

```
MariaDB [apotek]> SELECT nama_obat, MAX(stok_obat) FROM sisa_stok_barang;
+------------+----------------+
| nama_obat  | MAX(stok_obat) |
+------------+----------------+
| Paracetamol |             50 |
+------------+----------------+
1 row in set (0.007 sec)

MariaDB [apotek]>
```

3. Buatlah progam function dengan menggunakan min

```
SELECT nama_obat, MIN(stok_obat) FROM sisa_stok_barang;
```

```
MariaDB [apotek]> SELECT nama_obat, MIN(stok_obat) FROM sisa_stok_barang;
+------------+----------------+
| nama_obat  | MIN(stok_obat) |
+------------+----------------+
| Paracetamol |              7 |
+------------+----------------+
1 row in set (0.000 sec)

MariaDB [apotek]>
```

4. Buatlah progam function dengan menggunakan average

```
SELECT AVG(stok_obat) FROM sisa_stok_barang;
```

```
MariaDB [apotek]> SELECT AVG(stok_obat) FROM sisa_stok_barang;
+----------------+
| AVG(stok_obat) |
+----------------+
|        23.4762 |
+----------------+
1 row in set (0.000 sec)
```

5. Buatlah progam function dengan menggunakan sum

```
SELECT SUM(stok_obat) FROM sisa_stok_barang;
```

```
MariaDB [apotek]> SELECT SUM(stok_obat) FROM sisa_stok_barang;
+----------------+
| SUM(stok_obat) |
+----------------+
|            493 |
+----------------+
1 row in set (0.000 sec)
```

MODUL 4

PRAKTIKUM MODUL 4

Pertama kita membuat sebuah database bernama *toko_kain* sebagai berikut :

```
  id_produk INT,
  jumlah INT,
  total_harga DECIMAL(10, 2),
  FOREIGN KEY (id_produk) REFERENCES produk(id_produk)
);

CREATE TABLE pembelian (
  id_pembelian INT PRIMARY KEY,
  tanggal_pembelian DATE,
  id_supplier INT,
  id_produk INT,
  jumlah INT,
  total_harga DECIMAL(10, 2),
  FOREIGN KEY (id_supplier) REFERENCES supplier(id_supplier),
  FOREIGN KEY (id_produk) REFERENCES produk(id_produk)
);

CREATE TABLE pelanggan (
  id_pelanggan INT PRIMARY KEY,
  nama_pelanggan VARCHAR(50),
  alamat VARCHAR(100)
);

INSERT INTO supplier (id_supplier, nama_supplier, alamat_supplier)
VALUES (1, 'Supplier A', 'Jl. Supplier A No. 123'),
       (2, 'Supplier B', 'Jl. Supplier B No. 456'),
       (3, 'Supplier C', 'Jl. Supplier C No. 789'),
       (4, 'Supplier D', 'Jl. Supplier D No. 012'),
       (5, 'Supplier E', 'Jl. Supplier E No. 345'),
       (6, 'Supplier F', 'Jl. Supplier F No. 678'),
       (7, 'Supplier G', 'Jl. Supplier G No. 901'),
       (8, 'Supplier H', 'Jl. Supplier H No. 234'),
       (9, 'Supplier I', 'Jl. Supplier I No. 567'),
       (10, 'Supplier J', 'Jl. Supplier J No. 890');

```

```sql
CREATE DATABASE toko_kain;
USE toko_kain;

CREATE TABLE supplier (
  id_supplier INT PRIMARY KEY,
  nama_supplier VARCHAR(50),
  alamat_supplier VARCHAR(100)
);

CREATE TABLE produk (
  id_produk INT PRIMARY KEY,
  nama_produk VARCHAR(50),
  harga_produk DECIMAL(10, 2)
);

CREATE TABLE kategori_produk (
  id_kategori INT PRIMARY KEY,
  nama_kategori VARCHAR(50)
);

CREATE TABLE stok_produk (
  id_produk INT,
  stok_masuk INT,
  stok_keluar INT,
  FOREIGN KEY (id_produk) REFERENCES produk(id_produk)
);

CREATE TABLE penjualan (
  id_penjualan INT PRIMARY KEY,
  tanggal_penjualan DATE,
```

```sql
INSERT INTO produk (id_produk, nama_produk, harga_produk)
VALUES (1, 'Kain Katun', 50.00),
       (2, 'Kain Sutra', 100.00),
       (3, 'Kain Denim', 75.00),
       (4, 'Kain Linen', 90.00),
       (5, 'Kain Wol', 120.00),
       (6, 'Kain Satin', 80.00),
       (7, 'Kain Flanel', 60.00),
       (8, 'Kain Batik', 70.00),
       (9, 'Kain Tenun', 110.00),
       (10, 'Kain Rayon', 65.00);

INSERT INTO kategori_produk (id_kategori, nama_kategori)
VALUES (1, 'Kategori 1'),
       (2, 'Kategori 2'),
       (3, 'Kategori 3'),
       (4, 'Kategori 4'),
       (5, 'Kategori 5'),
       (6, 'Kategori 6'),
       (7, 'Kategori 7'),
       (8, 'Kategori 8'),
       (9, 'Kategori 9'),
       (10, 'Kategori 10');

INSERT INTO stok_produk (id_produk, stok_masuk, stok_keluar)
VALUES (1, 100, 50),
       (2, 200, 100),
       (3, 150, 80),
       (4, 120, 70),
       (5, 180, 90),
       (6, 140, 60),
       (7, 110, 40),
       (8, 130, 55),
       (9, 160, 75),
       (10, 90, 30);
```

```sql
INSERT INTO penjualan (id_penjualan, tanggal_penjualan, id_produk, jumlah, total_harga)
VALUES (1, '2023-06-01', 1, 10, 500.00),
       (2, '2023-06-02', 2, 5, 500.00),
       (3, '2023-06-03', 3, 8, 600.00),
       (4, '2023-06-04', 4, 7, 630.00),
       (5, '2023-06-05', 5, 12, 1440.00),
       (6, '2023-06-06', 6, 6, 480.00),
       (7, '2023-06-07', 7, 9, 540.00),
       (8, '2023-06-08', 8, 4, 280.00),
       (9, '2023-06-09', 9, 11, 1210.00),
       (10, '2023-06-10', 10, 3, 195.00);

INSERT INTO pembelian (id_pembelian, tanggal_pembelian, id_supplier, id_produk, jumlah, total_harga)
VALUES (1, '2023-06-01', 1, 1, 20, 1000.00),
       (2, '2023-06-02', 2, 2, 10, 1000.00),
       (3, '2023-06-03', 3, 3, 15, 1125.00),
       (4, '2023-06-04', 4, 4, 8, 720.00),
       (5, '2023-06-05', 5, 5, 25, 3000.00),
       (6, '2023-06-06', 6, 6, 12, 960.00),
       (7, '2023-06-07', 7, 7, 18, 1080.00),
       (8, '2023-06-08', 8, 8, 6, 420.00),
       (9, '2023-06-09', 9, 9, 22, 2420.00),
       (10, '2023-06-10', 10, 10, 5, 325.00);

INSERT INTO pelanggan (id_pelanggan, nama_pelanggan, alamat)
VALUES (1, 'Pelanggan A', 'Jl. Pelanggan A No. 123'),
       (2, 'Pelanggan B', 'Jl. Pelanggan B No. 456'),
       (3, 'Pelanggan C', 'Jl. Pelanggan C No. 789'),
       (4, 'Pelanggan D', 'Jl. Pelanggan D No. 012'),
       (5, 'Pelanggan E', 'Jl. Pelanggan E No. 345'),
       (6, 'Pelanggan F', 'Jl. Pelanggan F No. 678'),
       (7, 'Pelanggan G', 'Jl. Pelanggan G No. 901'),
       (8, 'Pelanggan H', 'Jl. Pelanggan H No. 234'),
       (9, 'Pelanggan I', 'Jl. Pelanggan I No. 567'),
       (10, 'Pelanggan J', 'Jl. Pelanggan J No. 890');
```

Soal dengan npm (genap) :

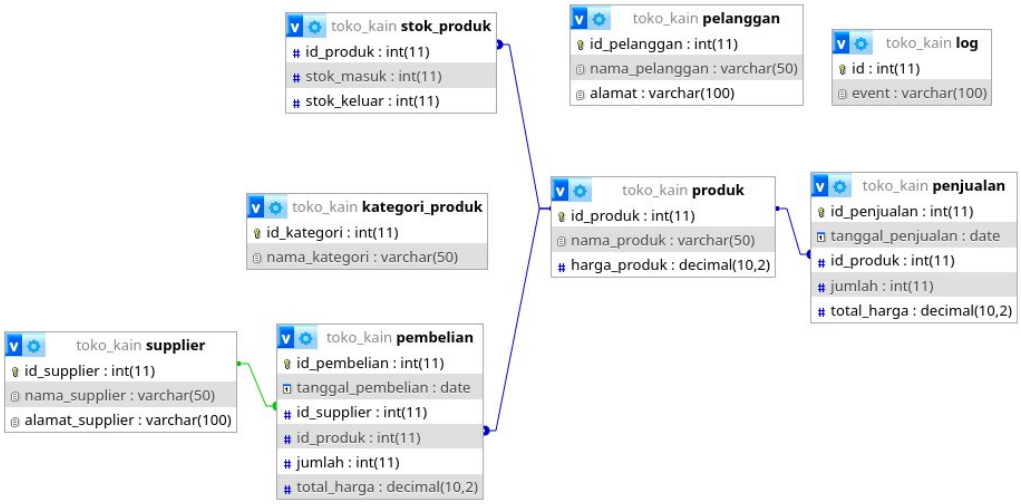
1. TRIGGER AFTER DELETE

```sql
-- Membuat tabel log untuk menyimpan entri log
CREATE TABLE log (
  id INT AUTO_INCREMENT PRIMARY KEY,
  event VARCHAR(100)
);

-- Membuat TRIGGER AFTER DELETE
DELIMITER //
CREATE TRIGGER produk_after_delete
AFTER DELETE ON produk
FOR EACH ROW
BEGIN
    INSERT INTO log (event) VALUES (CONCAT('Produk dihapus: ', OLD.nama_produk));
END //
DELIMITER ;
```

Pada saat melakukan *DELETE* terjadi sebuah error

```
sudo /opt/lampp/bin/mysql -u root

MariaDB [toko_kain]> DELETE FROM produk WHERE id_produk = 1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key con
straint fails (`toko_kain`.`pembelian`, CONSTRAINT `pembelian_ibfk_2` FOREI
GN KEY (`id_produk`) REFERENCES `produk` (`id_produk`))
MariaDB [toko_kain]>
```

Hal itu terjadi karena pada *TABLE* produk terdapat item yang menjadi *FOREIGN KEY* pada tabel lain



Cara untuk menangani hal tersebut yaitu menghapus data yang menjadi *FOREIGN KEY* pada setiap tabel-nya

```
sudo /opt/lampp/bin/mysql -u root

MariaDB [toko_kain]> DELETE FROM pembelian WHERE id_produk = 1;
Query OK, 1 row affected (0,010 sec)

MariaDB [toko_kain]> DELETE FROM stok_produk WHERE id_produk = 1;
Query OK, 1 row affected (0,010 sec)

MariaDB [toko_kain]> DELETE FROM penjualan WHERE id_produk = 1;
Query OK, 1 row affected (0,008 sec)

MariaDB [toko_kain]> DELETE FROM produk WHERE id_produk = 1;
Query OK, 1 row affected (0,009 sec)

MariaDB [toko_kain]>
```

2. TRIGGER BEFORE UPDATE

```
-- membuat TRIGGER BEFORE UPDATE:
DELIMITER //
CREATE TRIGGER produk_before_update
BEFORE UPDATE ON produk
FOR EACH ROW
BEGIN
    IF NEW.harga_produk = 0 THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Harga produk tidak dapat diubah menjadi 0.';
    END IF;
END //
DELIMITER ;
```

Pada *TRIGGER* diatas menggunakan *if statment* untuk menguji apakah nilai harga_produk == 0, jika iya maka SQLSTATE akan memberi SIGNAL berupa message yang bertuliskan "Harga produk tidak dapat diubah menjadi 0".

Pesan kesalahan ERROR 1644 (45000) pada MySQL menunjukkan adanya kesalahan atau pengecualian yang dikendalikan oleh pengguna (user-defined error). Kode status '45000' pada pesan kesalahan tersebut menunjukkan bahwa ini adalah jenis pengecualian umum.

Output :

```
sudo /opt/lampp/bin/mysql -u root

MariaDB [toko_kain]> UPDATE produk SET harga_produk = 0 WHERE id_produk = 2;

ERROR 1644 (45000): Harga produk tidak dapat diubah menjadi 0.
MariaDB [toko_kain]>
```

KRITIK

DAN

SARAN

- Kritik untuk (Nama Asisten Lab)

numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi

- Saran untuk (Nama Asisten Lab)

numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi

- Kritik untuk Laboratorium Sistem Informasi

numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi

- Saran untuk Laboratorium Sistem Informasi

numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi