

## DAFTAR ISI

Link Github: <https://github.com/miya3333/BasisDataP13>

<b>Laporan Materi Pertemuan 13.....</b>	<b>2</b>
<i>COMMIT</i> .....	2
<i>ROLLBACK</i> .....	3
<i>STORED PROCEDURE</i> .....	4
<i>TRIGGER</i> .....	6
<b>Tugas Rumah 1.....</b>	<b>8</b>
<i>Membuat database toko_buku</i> .....	8
TABEL buku .....	9
TABEL penulis.....	10
TABEL detail_buku .....	11
1. Definisikan Store prosedure .....	12
2. Definisikan suatu Stored procedure .....	13
3. Definisikan Stored Procedure.....	15
4. Berikan Presentase Pemahaman kalian .....	15
<b>Tugas Rumah 2.....</b>	<b>16</b>
<i>Membuat database perkuliahan</i> .....	16
1. Buat tabel siswa dengan field nis, nama, dan status serta tabel daftarnilai dengan field nis dan nilai.....	16
Tabel siswa & Tabel daftarnilai.....	16
Trigger INSERT & UPDATE .....	17
Menambahkan data.....	18
2. Buatlah dua tabel dosen_pa (field : id_pa, nama_pa, jml_mhs) dan mahasiswa (field: nim, nama, id_pa).....	19
Tabel dosen_pa & Tabel mahasiswa .....	19
Trigger INSERT, UPDATE & DELETE .....	20
Menggunakan INSERT .....	21
Menggunakan UPDATE .....	21
Menggunakan DELETE .....	22
3. Buatlah Trigger untuk tabel nilaimhs agar jika terdapat INSERT, UPDATE dan DELETE pada tabel nilaimhs,.....	23
Tabel mhs, Tabel mk & Tabel nilaimhs .....	23
Menghitung ip.....	24
Trigger INSERT, UPDATE & DELETE .....	25
Menggunakan INSERT .....	26
Menggunakan UPDATE .....	26
Menggunakan DELETE .....	27

## Laporan Materi Pertemuan 13


**COMMIT**

```
CMD - mysql -u root -p
MariaDB [p13]> START TRANSACTION;
Query OK, 0 rows affected (0.000 sec)

MariaDB [p13]> CREATE TABLE mahasiswa(
  -> nim INT(10),
  -> nama VARCHAR(100),
  -> alamat VARCHAR(100));
Query OK, 0 rows affected (0.065 sec)

MariaDB [p13]> INSERT INTO mahasiswa VALUES
  -> (412410200, "faqih", "bandung"),
  -> (412410201, "ina", "jakatra"),
  -> (412410202, "anto", "semarang"),
  -> (412410203, "dani", "padang"),
  -> (412410210, "jaka", "kalimantan");
Query OK, 5 rows affected (0.320 sec)
Records: 5  Duplicates: 0  Warnings: 0

MariaDB [p13]> COMMIT;
Query OK, 0 rows affected (0.000 sec)
```

**ROLLBACK** CMD - mysql -u root -p

MariaDB [p13]&gt; START TRANSACTION;

Query OK, 0 rows affected (0.000 sec)

MariaDB [p13]&gt; INSERT INTO mahasiswa VALUES (412410211, "fitri", "surabaya");

Query OK, 1 row affected (0.071 sec)

MariaDB [p13]&gt; SELECT \* FROM mahasiswa;

nim	nama	alamat
412410200	faqih	bandung
412410201	ina	jakatra
412410202	anto	semarang
412410203	dani	padang
412410210	jaka	kalimantan
412410211	fitri	surabaya

6 rows in set (0.001 sec)

MariaDB [p13]&gt; ROLLBACK;

Query OK, 0 rows affected (0.049 sec)

MariaDB [p13]&gt; SELECT \* FROM mahasiswa;

nim	nama	alamat
412410200	faqih	bandung
412410201	ina	jakatra
412410202	anto	semarang
412410203	dani	padang
412410210	jaka	kalimantan

5 rows in set (0.000 sec)

**STORED PROCEDURE**

```
CA: CMD - mysql -u root -p
```

```
MariaDB [p13]> DELIMITER //
MariaDB [p13]> CREATE PROCEDURE hello()
  -> BEGIN
  ->   SELECT "Hello World!";
  -> END//
Query OK, 0 rows affected (0.074 sec)
```

```
MariaDB [p13]> DELIMITER ;
MariaDB [p13]> CALL hello();
```

```
+-----+
| Hello World! |
+-----+
| Hello World! |
+-----+
1 row in set (0.001 sec)
```

```
Query OK, 0 rows affected (0.014 sec)
```

```
CA: CMD - mysql -u root -p
```

```
MariaDB [p13]> DELIMITER //
MariaDB [p13]> CREATE PROCEDURE selectMahasiswa()
  -> BEGIN
  ->   SELECT nim, nama FROM mahasiswa;
  -> END//
Query OK, 0 rows affected (0.144 sec)
```

```
MariaDB [p13]> DELIMITER ;
MariaDB [p13]> CALL selectMahasiswa();
```

```
+-----+-----+
| nim      | nama  |
+-----+-----+
| 412410200 | faqih |
| 412410201 | ina   |
| 412410202 | anto  |
| 412410203 | dani  |
| 412410210 | jaka  |
| 412410205 | nara  |
| 412410206 | senta |
+-----+-----+
7 rows in set (0.001 sec)
```

```
Query OK, 0 rows affected (0.035 sec)
```

```
CMD - mysql -u root -p
MariaDB [p13]> DELIMITER //
MariaDB [p13]> CREATE PROCEDURE alamatMahasiswa
-> (
->   alamatMhs VARCHAR(100)
-> )
-> BEGIN
->   SELECT * FROM mahasiswa
->   WHERE alamat = alamatMhs;
-> END//
Query OK, 0 rows affected (0.166 sec)

MariaDB [p13]> DELIMITER ;
MariaDB [p13]> CALL alamatMahasiswa("bandung");
+-----+-----+-----+
| nim      | nama  | alamat |
+-----+-----+-----+
| 412410200 | faqih | bandung |
| 412410205 | nara  | bandung |
+-----+-----+-----+
2 rows in set (0.081 sec)

Query OK, 0 rows affected (0.101 sec)
```

**TRIGGER**

```
CMD - mysql -u root -p
```

```
MariaDB [p13]> CREATE TABLE mahasiswa  
-> (  
-> nim INT(10),  
-> nama VARCHAR(100),  
-> alamat VARCHAR(100),  
-> PRIMARY KEY(nim)  
-> );
```

Query OK, 0 rows affected (0.133 sec)

```
CMD - mysql -u root -p
```

```
MariaDB [p13]> INSERT INTO mahasiswa VALUES  
-> (21400200, "faqih", "bandung"),  
-> (21400201, "ina", "jakarta"),  
-> (21400202, "anto", "semarang"),  
-> (21400203, "dani", "semarang"),  
-> (21400204, "jaka", "bandung"),  
-> (21400205, "nara", "bandung"),  
-> (21400206, "senta", "semarang");
```

Query OK, 7 rows affected (0.110 sec)

Records: 7 Duplicates: 0 Warnings: 0

```
MariaDB [p13]> CREATE TABLE log_mahasiswa  
-> (  
-> id_log INT(10) AUTO_INCREMENT,  
-> nim INT(10),  
-> alamat_lama VARCHAR(100),  
-> alamat_baru VARCHAR(100),  
-> waktu DATE,  
-> PRIMARY KEY(id_log)  
-> );
```

Query OK, 0 rows affected (0.142 sec)

```
MariaDB [p13]> DELIMITER //
```

```
MariaDB [p13]> CREATE TRIGGER update_alamat_mahasiswa  
-> BEFORE UPDATE  
-> ON mahasiswa  
-> FOR EACH ROW  
-> BEGIN  
-> INSERT INTO log_mahasiswa  
-> set nim = OLD.nim,  
-> alamat_lama=old.alamat,  
-> alamat_baru=new.alamat,  
-> waktu = NOW();  
-> END//
```

Query OK, 0 rows affected (0.128 sec)

```
CMD - mysql -u root -p
```

```
MariaDB [p13]> DELIMITER ;
```

```
MariaDB [p13]> UPDATE mahasiswa
```

```
-> SET alamat = 'surabaya'
```

```
-> WHERE nim = 21400200;
```

```
Query OK, 1 row affected (0.212 sec)
```

```
Rows matched: 1 Changed: 1 Warnings: 0
```

```
MariaDB [p13]> SELECT * FROM log_mahasiswa;
```

id_log	nim	alamat_lama	alamat_baru	waktu
1	21400200	bandung	surabaya	2024-06-15

```
1 row in set (0.001 sec)
```

```
MariaDB [p13]> SELECT * FROM mahasiswa;
```

nim	nama	alamat
21400200	faqih	surabaya
21400201	ina	jakarta
21400202	anto	semarang
21400203	dani	semarang
21400204	jaka	bandung
21400205	nara	bandung
21400206	senta	semarang

```
7 rows in set (0.001 sec)
```

## Tugas Rumah 1

## Membuat database toko\_buku

```
CMD - mysql -u root -p
```

```
MariaDB [(none)]> CREATE DATABASE toko_buku;  
Query OK, 1 row affected (0.020 sec)
```

```
MariaDB [(none)]> show databases;
```

Database
aldi312310200
aldihermansyah_312310200
basis_data
information_schema
mysql
p13
performance_schema
phpmyadmin
toko_buku
uts
uts_312310200



**TABEL buku**

```

C:\> CMD - mysql -u root -p
MariaDB [(none)]> use toko_buku;
Database changed
MariaDB [toko_buku]> CREATE TABLE `buku` (
  -> `id_buku` INT NOT NULL,
  -> `id_penulis` INT NOT NULL,
  -> `nama_buku` VARCHAR(255),
  -> `genre` VARCHAR(255),
  -> `tahun` INT,
  -> PRIMARY KEY(`id_buku`, `id_penulis`)
  -> );
Query OK, 0 rows affected (0.116 sec)

```

```

C:\> CMD - mysql -u root -p
MariaDB [toko_buku]> INSERT INTO buku VALUES
  -> (102, 190, 'Spring In London', 'Romance', 2007),
  -> (111, 129, 'God Fifht', 'Action', 2010),
  -> (142, 178, 'Kata Hati', 'Romance', 2008),
  -> (153, 128, 'Marmut Merah Jambu', 'Comedy', 2012),
  -> (188, 128, 'Koala Kumal', 'Comedy', 2015),
  -> (196, 129, 'Pillow Talk', 'Romance', 2008),
  -> (322, 190, 'In Blue Moon', 'Action', 2007);
Query OK, 7 rows affected (0.060 sec)
Records: 7  Duplicates: 0  Warnings: 0

```

```

MariaDB [toko_buku]> SELECT * FROM buku;

```

id_buku	id_penulis	nama_buku	genre	tahun
102	190	Spring In London	Romance	2007
111	129	God Fifht	Action	2010
142	178	Kata Hati	Romance	2008
153	128	Marmut Merah Jambu	Comedy	2012
188	128	Koala Kumal	Comedy	2015
196	129	Pillow Talk	Romance	2008
322	190	In Blue Moon	Action	2007

```

7 rows in set (0.001 sec)

```

**TABEL penulis**

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> CREATE TABLE `penulis` (  
  -> `id_penulis` INT NOT NULL,  
  -> `nama_penulis` VARCHAR(255),  
  -> PRIMARY KEY(`id_penulis`)  
  -> );
```

```
Query OK, 0 rows affected (0.112 sec)
```

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> INSERT INTO penulis VALUE  
  -> (128, 'Raditya Dika'),  
  -> (129, 'Bernard Batubara'),  
  -> (178, 'Cristian Simamora'),  
  -> (190, 'Ilana Tan');
```

```
Query OK, 4 rows affected (0.055 sec)
```

```
Records: 4 Duplicates: 0 Warnings: 0
```

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> SELECT * FROM penulis;
```

id_penulis	nama_penulis
128	Raditya Dika
129	Bernard Batubara
178	Cristian Simamora
190	Ilana Tan

```
4 rows in set (0.001 sec)
```

**TABEL detail\_buku**

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> CREATE TABLE `detail_buku` (
  -> `id_buku` INT NOT NULL,
  -> `harga` VARCHAR(255),
  -> `stok` VARCHAR(255),
  -> PRIMARY KEY(`id_buku`)
  -> );
```

```
Query OK, 0 rows affected (0.088 sec)
```

```
MariaDB [toko_buku]> ALTER TABLE `buku`
  -> ADD FOREIGN KEY(`id_buku`) REFERENCES `detail_buku`(`id_buku`)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
```

```
Query OK, 0 rows affected (0.475 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [toko_buku]> ALTER TABLE `buku`
  -> ADD FOREIGN KEY(`id_penulis`) REFERENCES `penulis`(`id_penulis`)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
```

```
Query OK, 0 rows affected (0.485 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> INSERT INTO detail_buku VALUES
  -> (102, 75000, 20),
  -> (142, 92000, 12),
  -> (196, 84000, 27),
  -> (111, 63000, 21),
  -> (322, 129000, 7),
  -> (153, 89000, 14),
  -> (188, 93000, 26);
```

```
Query OK, 7 rows affected (0.058 sec)
```

```
Records: 7 Duplicates: 0 Warnings: 0
```

```
CAL CMD - mysql -u root -p
```

```
MariaDB [toko_buku]> SELECT * FROM detail_buku;
```

id_buku	harga	stok
102	75000	20
111	63000	21
142	92000	12
153	89000	14
188	93000	26
196	84000	27
322	129000	7

```
7 rows in set (0.000 sec)
```

### 1. Definisikan Store prosedur

Untuk mengetahui apakah suatu buku tersedia atau tidak. jika Tersedia, set status "BUKU TERSEDIA". jika tidak Tersedia, set status "BUKU SEDANG KOSONG".

(saya menggunakan >20 untuk melihat "BUKU TERSEDIA" dan <20 untuk "BUKU SEDANG KOSONG")

```
C:\> CMD - mysql -u root -p
MariaDB [toko_buku]> DELIMITER //
MariaDB [toko_buku]> CREATE PROCEDURE cek_stok()
  -> BEGIN
  ->   SELECT id_buku, stok,
  ->     CASE
  ->       WHEN stok > 20 THEN 'BUKU TERSEDIA'
  ->       ELSE 'BUKU SEDANG KOSONG'
  ->     END AS sisa_buku
  ->   FROM detail_buku;
  -> END//
Query OK, 0 rows affected (0.083 sec)

MariaDB [toko_buku]> DELIMITER ;
MariaDB [toko_buku]> CALL cek_stok();
+-----+-----+-----+
| id_buku | stok | sisa_buku |
+-----+-----+-----+
|      102 |   20 | BUKU SEDANG KOSONG |
|      111 |   21 | BUKU TERSEDIA |
|      142 |   12 | BUKU SEDANG KOSONG |
|      153 |   14 | BUKU SEDANG KOSONG |
|      188 |   26 | BUKU TERSEDIA |
|      196 |   27 | BUKU TERSEDIA |
|      291 |    1 | BUKU SEDANG KOSONG |
|      322 |    7 | BUKU SEDANG KOSONG |
+-----+-----+-----+
8 rows in set (0.001 sec)

Query OK, 0 rows affected (0.043 sec)
```

## 2. Definisikan suatu Stored procedure

Yang berfungsi untuk menambahkan data pada tabel penulis.

```
CMD - mysql -u root -p
MariaDB [toko_buku]> DELIMITER //
MariaDB [toko_buku]> CREATE PROCEDURE tambah_penulis(
->     IN id_penulis INT,
->     IN nama_penulis VARCHAR(255)
-> )
-> BEGIN
->     INSERT INTO penulis (id_penulis, nama_penulis)
->     VALUES (id_penulis, nama_penulis);
-> END //
ERROR 1304 (42000): PROCEDURE tambah_penulis already exists
MariaDB [toko_buku]> DELIMITER ;
MariaDB [toko_buku]>
MariaDB [toko_buku]> CALL tambah_penulis(191, 'John Doe');
Query OK, 1 row affected (0.032 sec)

MariaDB [toko_buku]> INSERT INTO detail_buku VALUES(291, 57000, 1);
Query OK, 1 row affected (0.035 sec)

MariaDB [toko_buku]> INSERT INTO buku VALUES( 291, 191, 'Sea In The Dark', 'Horror', 2019);
Query OK, 1 row affected (0.037 sec)
```

```
CMD - mysql -u root -p
MariaDB [toko_buku]> SELECT * FROM penulis;
+-----+-----+
| id_penulis | nama_penulis |
+-----+-----+
| 128 | Raditya Dika |
| 129 | Bernard Batubara |
| 178 | Cristian Simamora |
| 190 | Ilana Tan |
| 191 | John Doe |
+-----+-----+
5 rows in set (0.001 sec)

MariaDB [toko_buku]> SELECT * FROM detail_buku;
+-----+-----+-----+
| id_buku | harga | stok |
+-----+-----+-----+
| 102 | 75000 | 20 |
| 111 | 63000 | 21 |
| 142 | 92000 | 12 |
| 153 | 89000 | 14 |
| 188 | 93000 | 26 |
| 196 | 84000 | 27 |
| 291 | 57000 | 1 |
| 322 | 129000 | 7 |
+-----+-----+-----+
8 rows in set (0.001 sec)

MariaDB [toko_buku]> SELECT * FROM buku;
+-----+-----+-----+-----+-----+
| id_buku | id_penulis | nama_buku | genre | tahun |
+-----+-----+-----+-----+-----+
| 102 | 190 | Spring In London | Romance | 2007 |
| 111 | 129 | God Fift | Action | 2010 |
| 142 | 178 | Kata Hati | Romance | 2008 |
| 153 | 128 | Marmut Merah Jambu | Comedy | 2012 |
| 188 | 128 | Koala Kumal | Comedy | 2015 |
| 196 | 129 | Pillow Talk | Romance | 2008 |
| 291 | 191 | Sea In The Dark | Horror | 2019 |
| 322 | 190 | In Blue Moon | Action | 2007 |
+-----+-----+-----+-----+-----+
8 rows in set (0.001 sec)
```

### 3. Definisikan Stored Procedure

Untuk Mengetahui Jumlah buku yang Bergenre Romance.

```
CMD - mysql -u root -p
MariaDB [toko_buku]> DELIMITER //
MariaDB [toko_buku]> CREATE PROCEDURE hitung_buku_romance()
  -> BEGIN
  ->     SELECT COUNT(*) AS jumlah_buku_romance
  ->     FROM buku
  ->     WHERE genre = 'Romance';
  -> END //
Query OK, 0 rows affected (0.112 sec)

MariaDB [toko_buku]> DELIMITER ;
MariaDB [toko_buku]> CALL hitung_buku_romance();
+-----+
| jumlah_buku_romance |
+-----+
|                    3 |
+-----+
1 row in set (0.002 sec)

Query OK, 0 rows affected (0.008 sec)
```

### 4. Berikan Presentase Pemahaman kalian

Pada Modul Kali ini (0-100%) dan Berikan penjelasam pada bagian yang kalian belum pahami (apabila ada)

*70% saya paham dengan membaca modul, tetapi tidak bisa menghafal semua peintah yang ada. 30% saya mengambil referensi dari chatGPT dan youtube untuk memahami modul ini.*

## Tugas Rumah 2

## Membuat database perkuliahan

```

C:\> CMD - mysql -u root -p
MariaDB [(none)]> CREATE DATABASE perkuliahan;
Query OK, 1 row affected (0.005 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| aldi312310200 |
| aldihermansyah_312310200 |
| information_schema |
| mysql |
| p13 |
| performance_schema |
| perkuliahan |
| phpdasar |
| phpmyadmin |
| toko_buku |
| uts |
| uts_312310200 |
+-----+
12 rows in set (0.002 sec)

```

1. Buat tabel siswa dengan *field* ***nis***, ***nama***, dan ***status*** serta tabel daftarnilai dengan *field* ***nis*** dan ***nilai***.

Selanjutnya, definisikan trigger pada tabel daftarnilai, sehingga saat melakukan **INSERT** dan **UPDATE** pada tabel daftarnilai, *field* ***status*** akan otomatis terisi “**LULUS**” atau “**TIDAK LULUS**” dengan ketentuan apabila **nilai < 75** dinyatakan tidak lulus.

Tabel *siswa* & Tabel *daftarnilai*

```

C:\> CMD - mysql -u root -p
MariaDB [perkuliahan]> CREATE TABLE siswa(
  -> nis INT NOT NULL,
  -> nilai VARCHAR(255),
  -> status VARCHAR(255),
  -> PRIMARY KEY(nis));
Query OK, 0 rows affected (0.216 sec)

MariaDB [perkuliahan]> CREATE TABLE daftarnilai(
  -> nis INT NOT NULL,
  -> nilai VARCHAR(255),
  -> PRIMARY KEY(nis));
Query OK, 0 rows affected (0.122 sec)

MariaDB [perkuliahan]> ALTER TABLE daftarnilai
  -> ADD FOREIGN KEY(nis) REFERENCES siswa(nis)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
Query OK, 0 rows affected (1.039 sec)
Records: 0 Duplicates: 0 Warnings: 0

```



**Trigger *INSERT & UPDATE***

```
CMD - mysql -u root -p
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER after_insert_nilai
  -> AFTER INSERT ON daftarnilai
  -> FOR EACH ROW
  -> BEGIN
  ->     IF NEW.nilai < 75 THEN
  ->         UPDATE siswa
  ->         SET status = 'TIDAK LULUS'
  ->         WHERE nis = NEW.nis;
  ->     ELSE
  ->         UPDATE siswa
  ->         SET status = 'LULUS'
  ->         WHERE nis = NEW.nis;
  ->     END IF;
  -> END //
Query OK, 0 rows affected (0.083 sec)

MariaDB [perkuliahan]> DELIMITER ;
MariaDB [perkuliahan]>
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER after_update_nilai
  -> AFTER UPDATE ON daftarnilai
  -> FOR EACH ROW
  -> BEGIN
  ->     IF NEW.nilai < 75 THEN
  ->         UPDATE siswa
  ->         SET status = 'TIDAK LULUS'
  ->         WHERE nis = NEW.nis;
  ->     ELSE
  ->         UPDATE siswa
  ->         SET status = 'LULUS'
  ->         WHERE nis = NEW.nis;
  ->     END IF;
  -> END //
Query OK, 0 rows affected (0.077 sec)
```

**Menambahkan data**

```
CMD - mysql -u root -p
MariaDB [perkuliahan]> INSERT INTO siswa (nis, nama) VALUES
  -> (1001, 'Ali'),
  -> (1005, 'Dita'),
  -> (1013, 'Jinha'),
  -> (1015, 'Aldi'),
  -> (1017, 'Bella');
Query OK, 5 rows affected (0.066 sec)
Records: 5  Duplicates: 0  Warnings: 0

MariaDB [perkuliahan]> INSERT INTO daftarnilai (nis, nilai) VALUES
  -> (1001, 80),
  -> (1005, 70),
  -> (1013, 75),
  -> (1015, 76),
  -> (1017, 74);
Query OK, 5 rows affected (0.046 sec)
Records: 5  Duplicates: 0  Warnings: 0

MariaDB [perkuliahan]> SELECT * FROM siswa;
+-----+-----+-----+
| nis | nama | status |
+-----+-----+-----+
| 1001 | Ali  | LULUS  |
| 1005 | Dita | TIDAK LULUS |
| 1013 | Jinha | LULUS  |
| 1015 | Aldi | LULUS  |
| 1017 | Bella | TIDAK LULUS |
+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [perkuliahan]> SELECT * FROM daftarnilai;
+-----+-----+
| nis | nilai |
+-----+-----+
| 1001 | 80    |
| 1005 | 70    |
| 1013 | 75    |
| 1015 | 76    |
| 1017 | 74    |
+-----+-----+
5 rows in set (0.001 sec)
```

2. Buatlah dua tabel **dosen\_pa** (field : **id\_pa**, **nama\_pa**, **jml\_mhs**) dan **mahasiswa** (field: **nim**, **nama**, **id\_pa**).

Kemudian buatlah trigger pada tabel mahasiswa untuk menghitung jumlah mahasiswa yang dibimbing oleh masing-masing **dosen\_pa**. Skenarionya, ketika terjadi **INSERT**, **UPDATE** dan **DELETE** pada tabel **mahasiswa**, field **jml\_mhs** pada tabel **dosen\_pa** akan terupdate secara otomatis.

#### Tabel **dosen\_pa** & Tabel **mahasiswa**

```
CAL CMD - mysql -u root -p
MariaDB [perkuliahan]> CREATE TABLE dosen_pa(
  -> id_pa INT NOT NULL,
  -> nama_pa VARCHAR(255),
  -> jml_mhs INT NOT NULL,
  -> PRIMARY KEY(id_pa));
Query OK, 0 rows affected (0.463 sec)

MariaDB [perkuliahan]> CREATE TABLE mahasiswa(
  -> nim INT NOT NULL,
  -> nama VARCHAR(255),
  -> id_pa INT NOT NULL,
  -> PRIMARY KEY(nim, id_pa));
Query OK, 0 rows affected (0.249 sec)

MariaDB [perkuliahan]> ALTER TABLE mahasiswa
  -> ADD FOREIGN KEY(id_pa) REFERENCES dosen_pa(id_pa)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
Query OK, 0 rows affected (1.079 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

**Trigger *INSERT, UPDATE & DELETE***

```
CMD - mysql -u root -p
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER insert_mahasiswa
  -> AFTER INSERT ON mahasiswa
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE dosen_pa
  ->     SET jml_mhs = jml_mhs + 1
  ->     WHERE id_pa = NEW.id_pa;
  -> END //
Query OK, 0 rows affected (0.450 sec)

MariaDB [perkuliahan]> DELIMITER ;
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER update_mahasiswa
  -> AFTER UPDATE ON mahasiswa
  -> FOR EACH ROW
  -> BEGIN
  ->     IF OLD.id_pa != NEW.id_pa THEN
  ->         UPDATE dosen_pa
  ->         SET jml_mhs = jml_mhs - 1
  ->         WHERE id_pa = OLD.id_pa;
  ->
  ->         UPDATE dosen_pa
  ->         SET jml_mhs = jml_mhs + 1
  ->         WHERE id_pa = NEW.id_pa;
  ->     END IF;
  -> END //
Query OK, 0 rows affected (0.222 sec)

MariaDB [perkuliahan]> DELIMITER ;
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER delete_mahasiswa
  -> AFTER DELETE ON mahasiswa
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE dosen_pa
  ->     SET jml_mhs = jml_mhs - 1
  ->     WHERE id_pa = OLD.id_pa;
  -> END //
Query OK, 0 rows affected (0.091 sec)
```

### Menggunakan *INSERT*

```

C:\> CMD - mysql -u root -p

MariaDB [perkuliahan]> INSERT INTO dosen_pa (id_pa, nama_pa) VALUES
  -> (001, 'Triyanna Widiyaningtyas, S.T., M.T.'),
  -> (002, 'Utomo Pujiyanto, S.Kom., M.Kom.'),
  -> (003, 'M. Zainal Arifin, S.Si., M.Kom.'),
  -> (004, 'Didik Dwi Prasetya, S.T., M.T.'),
  -> (005, 'Drs. Wahyu Sakti G. I., M.Kom.');
```

Query OK, 5 rows affected, 1 warning (0.101 sec)  
Records: 5 Duplicates: 0 Warnings: 1

```

MariaDB [perkuliahan]> INSERT INTO mahasiswa (nim, nama, id_pa) VALUES
  -> (150571, 'Mudafiq', 001),
  -> (150572, 'Fitrika', 003),
  -> (150573, 'Charisma', 002),
  -> (150574, 'Ivan', 001),
  -> (150575, 'Adam', 002),
  -> (150576, 'Irma', 001);
```

Query OK, 6 rows affected (0.042 sec)  
Records: 6 Duplicates: 0 Warnings: 0

### Menggunakan *UPDATE*

```

C:\> CMD - mysql -u root -p

MariaDB [perkuliahan]> SELECT * FROM mahasiswa;
```

nim	nama	id_pa
150571	Mudafiq	1
150572	Fitrika	3
150573	Charisma	2
150574	Ivan	1
150575	Adam	2
150576	Irma	1

6 rows in set (0.002 sec)

```

MariaDB [perkuliahan]> UPDATE mahasiswa SET id_pa = 3 WHERE nim = 150575;
```

Query OK, 1 row affected (0.030 sec)  
Rows matched: 1 Changed: 1 Warnings: 0

```

MariaDB [perkuliahan]> SELECT * FROM mahasiswa;
```

nim	nama	id_pa
150571	Mudafiq	1
150572	Fitrika	3
150573	Charisma	2
150574	Ivan	1
150575	Adam	3
150576	Irma	1

6 rows in set (0.001 sec)

**Menggunakan *DELETE***

```
CAL CMD - mysql -u root -p
MariaDB [perkuliahan]> DELETE FROM mahasiswa WHERE nim = 150576;
Query OK, 1 row affected (0.079 sec)

MariaDB [perkuliahan]> DELETE FROM mahasiswa WHERE nim = 150572;
Query OK, 1 row affected (0.081 sec)

MariaDB [perkuliahan]> SELECT * FROM mahasiswa;
+-----+-----+-----+
| nim    | nama      | id_pa |
+-----+-----+-----+
| 150571 | Mudafiq   | 1     |
| 150573 | Charisma  | 2     |
| 150574 | Ivan      | 1     |
| 150575 | Adam      | 3     |
+-----+-----+-----+
4 rows in set (0.001 sec)

MariaDB [perkuliahan]> SELECT * FROM dosen_pa;
+-----+-----+-----+
| id_pa | nama_pa                                     | jml_mhs |
+-----+-----+-----+
| 1     | Triyanna Widiyaningtyas, S.T., M.T.      | 2       |
| 2     | Utomo Pujiyanto, S.Kom., M.Kom.          | 1       |
| 3     | M. Zainal Arifin, S.Si., M.Kom.          | 1       |
| 4     | Didik Dwi Prasetya, S.T., M.T.           | 0       |
| 5     | Drs. Wahyu Sakti G. I., M.Kom.           | 0       |
+-----+-----+-----+
5 rows in set (0.001 sec)
```

3. Buatlah Trigger untuk tabel *nilaimhs* agar jika terdapat INSERT, UPDATE dan DELETE pada tabel *nilaimhs*, maka data pada *field ip* di tabel *mhs* secara otomatis terupdate.

Rumus menghitung Ip (Indeks Prestasi):

$$Ip = \frac{\text{jumlah Bobot mk yang diambil}}{\text{jumlah SKS mk yang diambil}}$$

Dimana, **bobot** =

Jika Nilai Abjad “A” = sks x 4

Jika Nilai Abjad “C” = sks x 2

Jika Nilai Abjad “B” = sks x 3

Jika Nilai Abjad “D” = sks x 1

Tabel *mhs*, Tabel *mk* & Tabel *nilaimhs*

```

C:\ CMD - mysql -u root -p
MariaDB [perkuliahan]> CREATE TABLE mhs(
  -> nim INT NOT NULL,
  -> nama VARCHAR(255),
  -> ip FLOAT NOT NULL,
  -> PRIMARY KEY(nim));
Query OK, 0 rows affected (0.335 sec)

MariaDB [perkuliahan]> CREATE TABLE mk(
  -> kode_mk VARCHAR(255) NOT NULL,
  -> nama_mk VARCHAR(255),
  -> sks INT NOT NULL,
  -> PRIMARY KEY(kode_mk));
Query OK, 0 rows affected (0.111 sec)

MariaDB [perkuliahan]> CREATE TABLE nilaimhs(
  -> nim INT NOT NULL,
  -> kode_mk VARCHAR(255) NOT NULL,
  -> nilai_abjad VARCHAR(255) NOT NULL,
  -> PRIMARY KEY(nim, kode_mk));
Query OK, 0 rows affected (0.229 sec)

MariaDB [perkuliahan]> ALTER TABLE nilaimhs
  -> ADD FOREIGN KEY(nim) REFERENCES mhs(nim)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
Query OK, 0 rows affected (0.562 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [perkuliahan]> ALTER TABLE nilaimhs
  -> ADD FOREIGN KEY(kode_mk) REFERENCES mk(kode_mk)
  -> ON UPDATE CASCADE ON DELETE CASCADE;
Query OK, 0 rows affected (0.484 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

**Menghitung *ip***

```
CMD - mysql -u root -p
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE FUNCTION hitung_ip(nim INT) RETURNS FLOAT
-> BEGIN
->     DECLARE total_bobot FLOAT;
->     DECLARE total_sks INT;
->
->     -- Menghitung total bobot
->     SELECT SUM(
->         CASE
->             WHEN nilai_abjad = 'A' THEN mk.sks * 4
->             WHEN nilai_abjad = 'B' THEN mk.sks * 3
->             WHEN nilai_abjad = 'C' THEN mk.sks * 2
->             WHEN nilai_abjad = 'D' THEN mk.sks * 1
->             ELSE 0
->         END
->     ) INTO total_bobot
->     FROM nilaimhs
->     JOIN mk ON nilaimhs.kode_mk = mk.kode_mk
->     WHERE nim = nim;
->
->     -- Menghitung total sks
->     SELECT SUM(mk.sks) INTO total_sks
->     FROM nilaimhs
->     JOIN mk ON nilaimhs.kode_mk = mk.kode_mk
->     WHERE nim = nim;
->
->     -- Menghitung IP
->     IF total_sks > 0 THEN
->         RETURN total_bobot / total_sks;
->     ELSE
->         RETURN 0;
->     END IF;
-> END //
Query OK, 0 rows affected (0.080 sec)
```



**Trigger *INSERT, UPDATE & DELETE***

```
CMD - mysql -u root -p
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER insert_nilaimhs
  -> AFTER INSERT ON nilaimhs
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE mhs
  ->     SET ip = hitung_ip(NEW.nim)
  ->     WHERE nim = NEW.nim;
  -> END //
Query OK, 0 rows affected (0.058 sec)

MariaDB [perkuliahan]> DELIMITER ;
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER update_nilaimhs
  -> AFTER UPDATE ON nilaimhs
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE mhs
  ->     SET ip = hitung_ip(NEW.nim)
  ->     WHERE nim = NEW.nim;
  -> END //
Query OK, 0 rows affected (0.109 sec)

MariaDB [perkuliahan]> DELIMITER ;
MariaDB [perkuliahan]> DELIMITER //
MariaDB [perkuliahan]> CREATE TRIGGER delete_nilaimhs
  -> AFTER DELETE ON nilaimhs
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE mhs
  ->     SET ip = hitung_ip(OLD.nim)
  ->     WHERE nim = OLD.nim;
  -> END //
Query OK, 0 rows affected (0.111 sec)
```

## Menggunakan *INSERT*

```

C:\ CMD - mysql -u root -p
MariaDB [perkuliahan]> INSERT INTO mhs (nim, nama) VALUES
  -> (101, 'Andi'),
  -> (102, 'Rahmat'),
  -> (103, 'Bayu');
Query OK, 3 rows affected, 1 warning (0.067 sec)
Records: 3  Duplicates: 0  Warnings: 1

MariaDB [perkuliahan]> INSERT INTO mk (kode_mk, nama_mk, sks) VALUES
  -> ('PTI447', 'Praktikum Basis Data', 1),
  -> ('TIK342', 'Praktikum Basis Data', 1),
  -> ('PTI333', 'Basis Data Terdistribusi', 3),
  -> ('TIK123', 'Jaringan Komputer', 2),
  -> ('TIK333', 'Sistem Operasi', 3),
  -> ('PTI123', 'Grafika Multimedia', 3),
  -> ('PTI777', 'Sistem Informasi', 2);
Query OK, 7 rows affected (0.050 sec)
Records: 7  Duplicates: 0  Warnings: 0

MariaDB [perkuliahan]> INSERT INTO nilaimhs (nim, kode_mk, nilai_abjad) VALUES
  -> (101, 'PTI447', 'A'),
  -> (101, 'PTI123', 'B');
Query OK, 2 rows affected (0.061 sec)
Records: 2  Duplicates: 0  Warnings: 0

MariaDB [perkuliahan]> SELECT * FROM mhs;
+-----+-----+-----+
| nim | nama  | ip    |
+-----+-----+-----+
| 101 | Andi  | 3.25  |
| 102 | Rahmat | 0     |
| 103 | Bayu  | 0     |
+-----+-----+-----+
3 rows in set (0.002 sec)

```

## Menggunakan *UPDATE*

```

C:\ CMD - mysql -u root -p
MariaDB [perkuliahan]> UPDATE nilaimhs SET nilai_abjad = 'A' WHERE nim = 101 AND kode_mk = 'PTI123';
Query OK, 1 row affected (0.070 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [perkuliahan]> SELECT * FROM mhs;
+-----+-----+-----+
| nim | nama  | ip    |
+-----+-----+-----+
| 101 | Andi  | 4     |
| 102 | Rahmat | 0     |
| 103 | Bayu  | 0     |
+-----+-----+-----+
3 rows in set (0.001 sec)

```

**Menggunakan *DELETE***

```
CAL CMD - mysql -u root -p
MariaDB [perkuliahan]> DELETE FROM nilaimhs
  -> WHERE nim = 101
  -> AND kode_mk = 'PTI123';
Query OK, 1 row affected (0.145 sec)

MariaDB [perkuliahan]> SELECT * FROM mhs;
+-----+-----+-----+
| nim | nama  | ip |
+-----+-----+-----+
| 101 | Andi  | 4  |
| 102 | Rahmat | 0  |
| 103 | Bayu  | 0  |
+-----+-----+-----+
3 rows in set (0.001 sec)

MariaDB [perkuliahan]> DELETE FROM nilaimhs
  -> WHERE nim = 101
  -> AND kode_mk = 'PTI447';
Query OK, 1 row affected (0.038 sec)

MariaDB [perkuliahan]> SELECT * FROM mhs;
+-----+-----+-----+
| nim | nama  | ip |
+-----+-----+-----+
| 101 | Andi  | 0  |
| 102 | Rahmat | 0  |
| 103 | Bayu  | 0  |
+-----+-----+-----+
3 rows in set (0.001 sec)
```