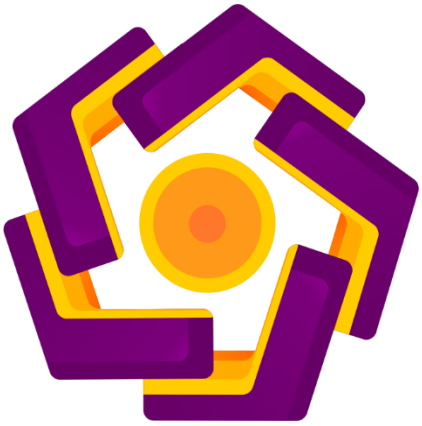
LAPORAN UJIAN AKHIR SEMESTER
PEMROGRAMAN BASIS DATA

Dosen: Novi Prisma Yunita, M.Kom



Muhammad Ali Arrayyan Bukhori	22.11.5108
Putra Qifadanu Tamher Sirua	22.11.5091
Patrias Dannis Pratama	22.11.4961

PROGRAM STUDI S1 INFORMATIKA, FAKULTAS ILMU KOMPUTER

UNIVERSITAS AMIKOM YOGYAKARTA

2024

Desain Basis Data Sistem Kost-Al

1.1 Entity Relationship Diagram (ERD)

Penjelasan ERD:

- Pengguna: Tabel ini menyimpan data pengguna (penyewa dan pemilik).
- Detail_Pengguna: Tabel ini menyimpan informasi tambahan mengenai pengguna dan memiliki relasi one-to-one dengan tabel Pengguna.
- Kost: Tabel ini menyimpan data kost yang dimiliki oleh pemilik.
- Kamar: Tabel ini menyimpan data kamar yang ada di kost dan memiliki relasi one-to-many dengan tabel Kost.
- Penyewaan: Tabel ini menyimpan data penyewaan kamar dan memiliki relasi many-to-many dengan tabel Pengguna dan Kamar.
- Fasilitas: Tabel ini menyimpan data fasilitas yang ada di kamar kost.
- Kamar_Fasilitas: Tabel ini menyimpan relasi many-to-many antara tabel Kamar dan Fasilitas.
- Transaksi: Tabel ini menyimpan data transaksi pembayaran penyewaan kamar.

1.2 Struktur Tabel dan Relasi

Pengguna

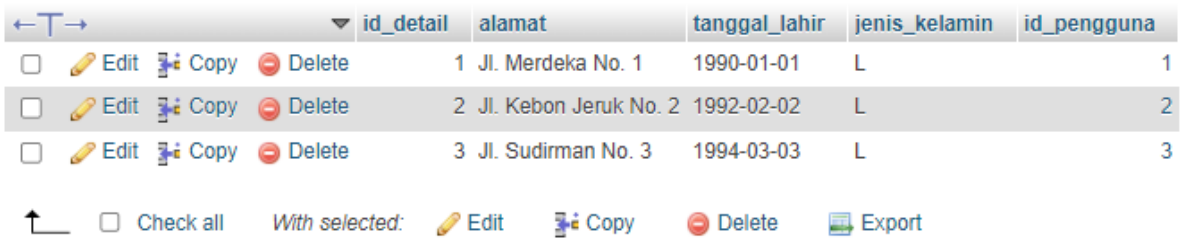
```
CREATE TABLE Pengguna (
    id_pengguna INT AUTO_INCREMENT PRIMARY KEY,
    nama VARCHAR(100),
    email VARCHAR(100),
    password VARCHAR(100),
    no_telepon VARCHAR(15)
);
```

	id_pengguna	nama	email	password	no_telepon
Edit Copy Delete	1	Muhammad Ali	muhammadali@gmail.com	password123	081234567890
Edit Copy Delete	2	Naufal Haidar	naufal@gmail.com	password123	082345678901
Edit Copy Delete	3	Bayu Setya Adjie	bayu@gmail.com	password123	083456789012

Check all With selected: Edit Copy Delete Export

Detail_Pengguna

```
CREATE TABLE Detail_Pengguna (
    id_detail_pengguna INT AUTO_INCREMENT PRIMARY KEY,
    alamat VARCHAR(255),
    tanggal_lahir DATE,
    jenis_kelamin CHAR(1),
    id_pengguna INT,
    FOREIGN KEY (id_pengguna) REFERENCES Pengguna(id_pengguna)
);
```



	id_detail	alamat	tanggal_lahir	jenis_kelamin	id_pengguna
Edit Copy Delete	1	Jl. Merdeka No. 1	1990-01-01	L	1
Edit Copy Delete	2	Jl. Kebon Jeruk No. 2	1992-02-02	L	2
Edit Copy Delete	3	Jl. Sudirman No. 3	1994-03-03	L	3

Check all With selected: Edit Copy Delete Export

Kost

```
CREATE TABLE Kost (
    id_kost INT AUTO_INCREMENT PRIMARY KEY,
    nama_kost VARCHAR(100),
    alamat VARCHAR(255),
    pemilik_id INT,
    FOREIGN KEY (pemilik_id) REFERENCES Pengguna(id_pengguna)
);
```

id_kost	nama_kost	alamat	pemilik_id
1	Kost Indah	Jl. Mawar No. 10	1
2	Kost Mewah	Jl. Melati No. 20	2

Edit Copy Delete | Check all | With selected: Edit Copy Delete Export

Kamar

```
CREATE TABLE Kamar (
    id_kamar INT AUTO_INCREMENT PRIMARY KEY,
    nomor_kamar VARCHAR(10),
    ukuran VARCHAR(10),
    harga DECIMAL(10, 2),
    status VARCHAR(20),
    id_kost INT,
    FOREIGN KEY (id_kost) REFERENCES Kost(id_kost)
);
```

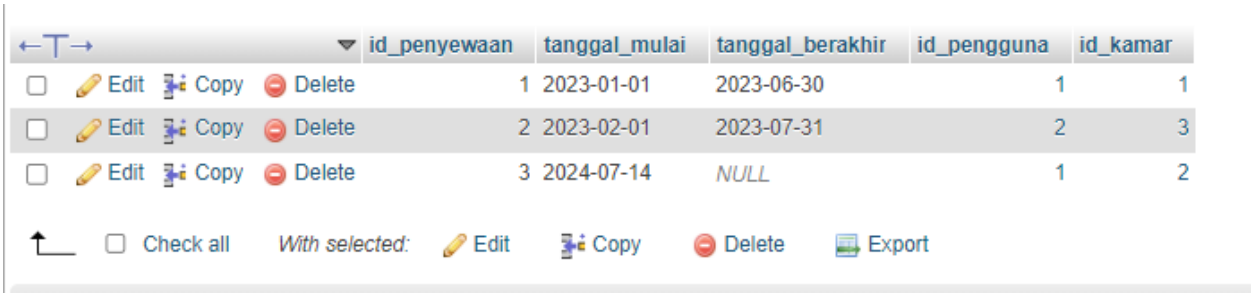
id_kamar	nomor_kamar	ukuran	harga	status	id_kost
1	101	3x4	1500000.00	tidak tersedia	1
2	102	3x4	1500000.00	tidak tersedia	1
3	201	3x4	2000000.00	tidak tersedia	2
4	202	3x4	2000000.00	tidak tersedia	2

Edit Copy Delete | Check all | With selected: Edit Copy Delete Export

Penyewaan

```
CREATE TABLE Penyewaan (
    id_penyewaan INT AUTO_INCREMENT PRIMARY KEY,
    tanggal_mulai DATE,
```

```
    tanggal_berakhir DATE,
    id_pengguna INT,
    id_kamar INT,
    FOREIGN KEY (id_pengguna) REFERENCES Pengguna(id_pengguna),
    FOREIGN KEY (id_kamar) REFERENCES Kamar(id_kamar)
);
```



	id_penyewaan	tanggal_mulai	tanggal_berakhir	id_pengguna	id_kamar
Edit Copy Delete	1	2023-01-01	2023-06-30	1	1
Edit Copy Delete	2	2023-02-01	2023-07-31	2	3
Edit Copy Delete	3	2024-07-14	NULL	1	2

Check all With selected: Edit Copy Delete Export

Fasilitas

```
CREATE TABLE Fasilitas (
    id_fasilitas INT AUTO_INCREMENT PRIMARY KEY,
    nama_fasilitas VARCHAR(100)
);
```

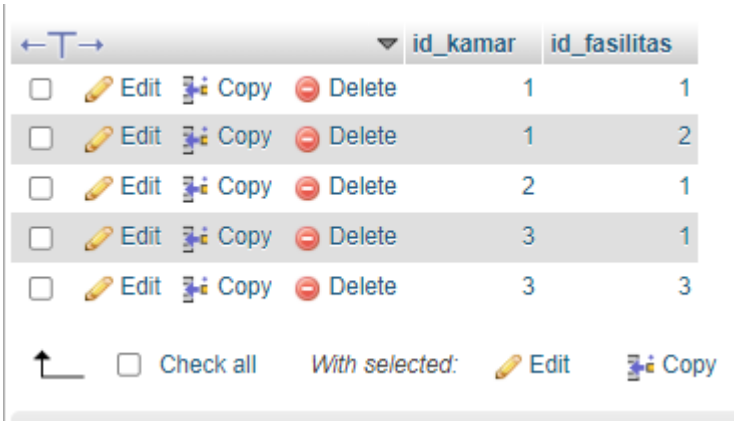
	id_fasilitas	nama_fasilitas
Edit Copy Delete	1	WiFi
Edit Copy Delete	2	AC
Edit Copy Delete	3	Kamar Mandi Dalam

Check all With selected: Edit Copy Delete Export

Kamar_Fasilitas

```
CREATE TABLE Kamar_Fasilitas (
    id_kamar INT,
```

```
    id_fasilitas INT,
    PRIMARY KEY (id_kamar, id_fasilitas),
    FOREIGN KEY (id_kamar) REFERENCES Kamar(id_kamar),
    FOREIGN KEY (id_fasilitas) REFERENCES
Fasilitas(id_fasilitas)
);
```



Transaksi

```
CREATE TABLE Transaksi (
    id_transaksi INT AUTO_INCREMENT PRIMARY KEY,
    tanggal_transaksi DATE,
    jumlah DECIMAL(10, 2),
    metode_pembayaran VARCHAR(50),
    id_penyewaan INT,
    FOREIGN KEY (id_penyewaan) REFERENCES
Penyewaan(id_penyewaan)
);
```

	id_transaksi	tanggal_transaksi	jumlah	metode_pembayaran	id_penyewaan
Edit Copy Delete	1	2023-01-01	1500000.00	Transfer	1
Edit Copy Delete	2	2023-02-01	2000000.00	Cash	2

1.3 Insert Data

Pengguna

```
INSERT INTO Pengguna (nama, email, password, no_telepon) VALUES
('Muhammad Ali', 'muhammadali@gmail.com', 'password123',
'081234567890'),
('Naufal Haidar', 'naufal@example.com', 'password123',
'082345678901'),
('Bayu Setya Adjie', 'bayu@example.com', 'password123',
'083456789012');
```

Detail Pengguna

```
INSERT INTO Detail_Pengguna (alamat, tanggal_lahir,
jenis_kelamin, id_pengguna) VALUES
('Jl. Merdeka No. 1', '1990-01-01', 'L', 1),
('Jl. Kebon Jeruk No. 2', '1992-02-02', 'L', 2),
('Jl. Sudirman No. 3', '1994-03-03', 'L', 3);
```

Kost

```
INSERT INTO Kost (nama_kost, alamat, pemilik_id) VALUES
('Kost Indah', 'Jl. Mawar No. 10', 1),
('Kost Mewah', 'Jl. Melati No. 20', 2);
```

Kamar

```
INSERT INTO Kamar (nomor_kamar, ukuran, harga, status, id_kost)
VALUES
```

```
('101', '3x4', 1500000, 'tersedia', 1),
('102', '3x4', 1500000, 'tersedia', 1),
('201', '3x4', 2000000, 'tidak tersedia', 2),
('202', '3x4', 2000000, 'tersedia', 2);
```

Penyewaan

```
INSERT INTO Penyewaan (tanggal_mulai, tanggal_berakhir,
id_pengguna, id_kamar) VALUES
('2023-01-01', '2023-06-30', 1, 1),
('2023-02-01', '2023-07-31', 2, 3);
```

Fasilitas

```
INSERT INTO Fasilitas (nama_fasilitas) VALUES
('WiFi'),
('AC'),
('Kamar Mandi Dalam');
```

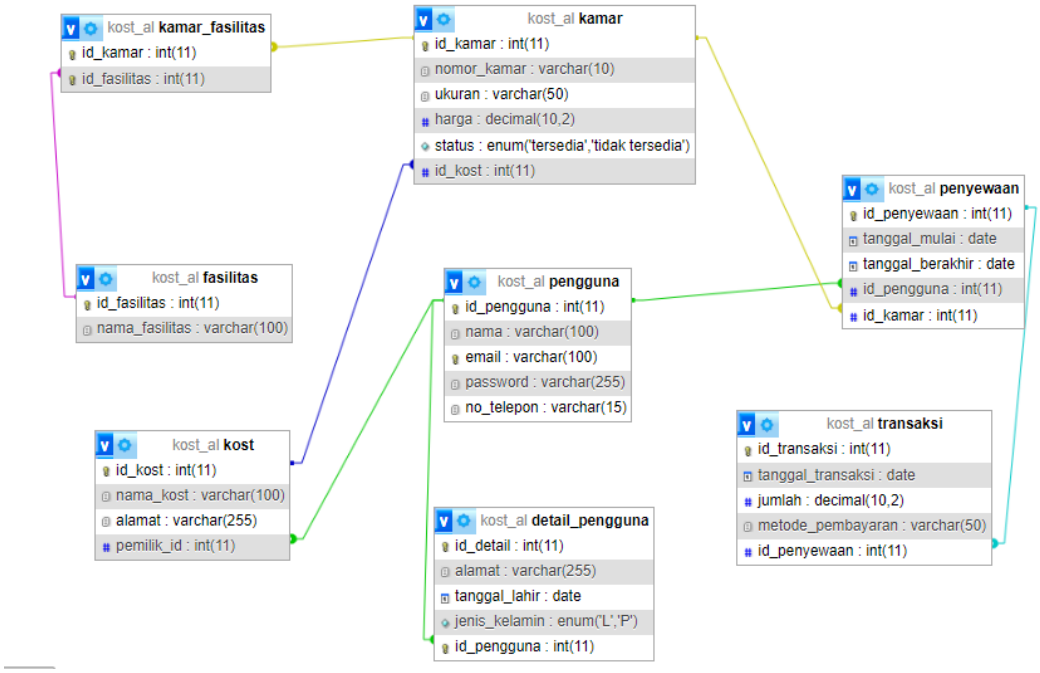
Kamar_Fasilitas

```
INSERT INTO Kamar_Fasilitas (id_kamar, id_fasilitas) VALUES
(1, 1),
(1, 2),
(2, 1),
(3, 1),
(3, 3);
```

Transaksi

```
INSERT INTO Transaksi (tanggal_transaksi, jumlah,
metode_pembayaran, id_penyewaan) VALUES
('2023-01-01', 1500000, 'Transfer', 1),
```

```
('2023-02-01', 2000000, 'Cash', 2);
```



Gambar Entity Relationship Diagram

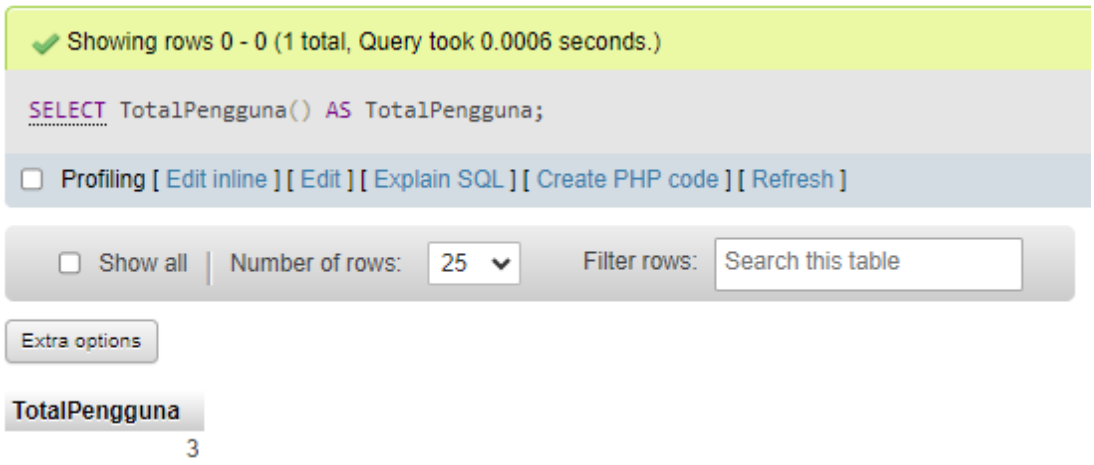
Implementasi Perintah-Perintah SQL

2.1 Function

Function dengan parameter kosong

```
CREATE FUNCTION TotalPengguna()
RETURNS INT
BEGIN
    DECLARE total INT;
    SELECT COUNT(*) INTO total FROM Pengguna;
    RETURN total;
END;
```

Hasil Eksekusi

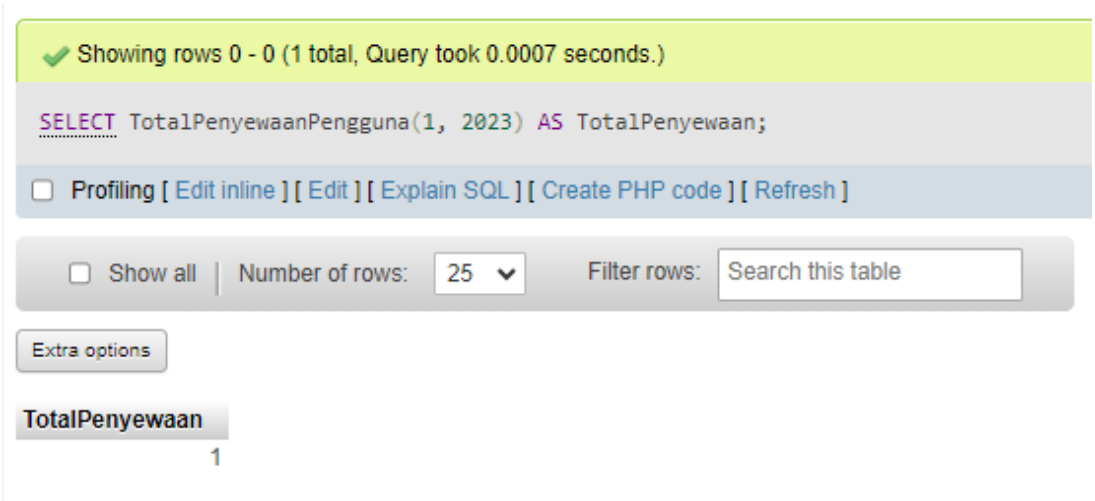


Function dengan dua parameter

```
CREATE FUNCTION TotalPenyewaanPengguna(id INT, tahun INT)
RETURNS INT
BEGIN
    DECLARE total INT;
    SELECT COUNT(*) INTO total
    FROM Penyewaan
    WHERE id_pengguna = id AND YEAR(tanggal_mulai) = tahun;
```

```
    RETURN total;
END;
```

Hasil Eksekusi



Daftrar Function

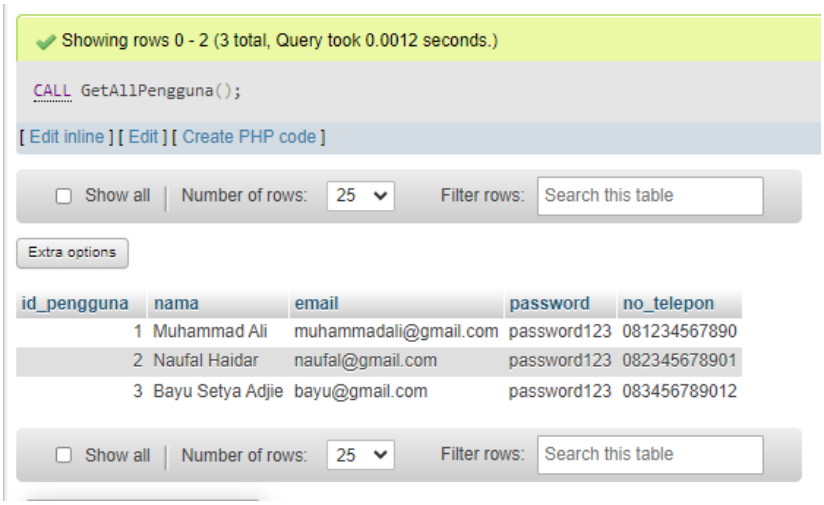
```
SHOW FUNCTION STATUS WHERE Db = 'kost_al';
```

2.2 Procedure

Procedure dengan parameter kosong

```
CREATE PROCEDURE GetAllPengguna()
BEGIN
    SELECT * FROM Pengguna;
END;
```

Hasil Eksekusi

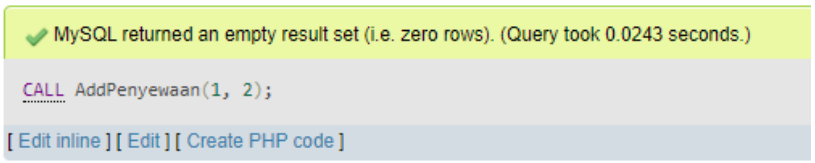


Function dengan dua parameter

```
CREATE PROCEDURE AddPenyewaan(id_pengguna INT, id_kamar INT)
BEGIN
    DECLARE kamar_status VARCHAR(20);
    SELECT status INTO kamar_status FROM Kamar WHERE id_kamar = id_kamar;

    IF kamar_status = 'tersedia' THEN
        INSERT INTO Penyewaan (tanggal_mulai, id_pengguna, id_kamar)
        VALUES (CURDATE(), id_pengguna, id_kamar);
        UPDATE Kamar SET status = 'tidak tersedia' WHERE id_kamar = id_kamar;
    ELSE
        SELECT 'Kamar tidak tersedia' AS pesan;
    END IF;
END;
```

Hasil Eksekusi



Daftrar Procedure

```
SHOW PROCEDURE STATUS WHERE Db = 'kost_al';
```

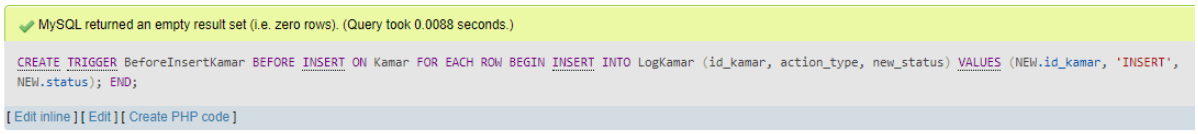
2.3 Trigger

Membuat Tabel Log

```
CREATE TABLE LogKamar (
    id_log INT AUTO_INCREMENT PRIMARY KEY,
    id_kamar INT,
    action_type VARCHAR(50),
    old_status VARCHAR(20),
    new_status VARCHAR(20),
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

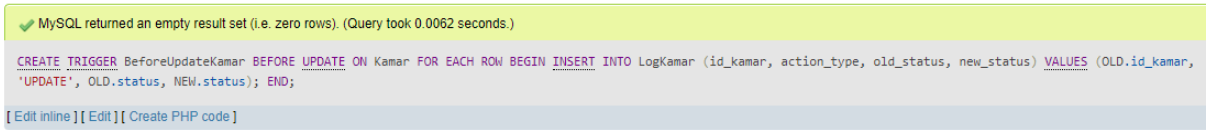
Trigger BEFORE INSERT

```
CREATE TRIGGER BeforeInsertKamar
BEFORE INSERT ON Kamar
FOR EACH ROW
BEGIN
    INSERT INTO LogKamar (id_kamar, action_type, new_status)
    VALUES (NEW.id_kamar, 'INSERT', NEW.status);
END;
```



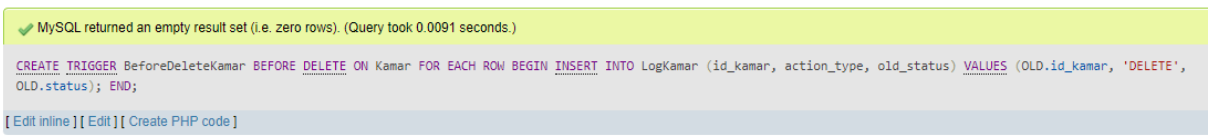
Trigger BEFORE UPDATE

```
CREATE TRIGGER BeforeUpdateKamar
BEFORE UPDATE ON Kamar
FOR EACH ROW
BEGIN
    INSERT INTO LogKamar (id_kamar, action_type, old_status, new_status)
    VALUES (OLD.id_kamar, 'UPDATE', OLD.status, NEW.status);
END;
```



Trigger BEFORE DELETE

```
CREATE TRIGGER BeforeDeleteKamar
BEFORE DELETE ON Kamar
FOR EACH ROW
BEGIN
    INSERT INTO LogKamar (id_kamar, action_type, old_status)
    VALUES (OLD.id_kamar, 'DELETE', OLD.status);
END;
```

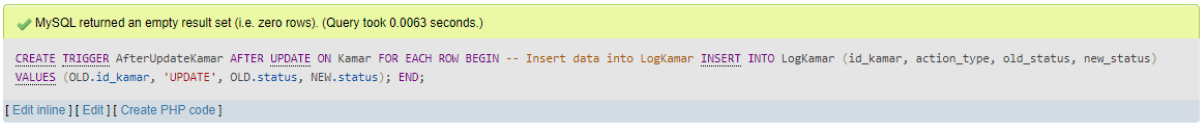


Trigger AFTER INSERT

```
CREATE TRIGGER AfterInsertKamar
AFTER INSERT ON Kamar
FOR EACH ROW
BEGIN
    INSERT INTO LogKamar (id_kamar, action_type, new_status)
    VALUES (NEW.id_kamar, 'INSERT', NEW.status);
END;
```

Trigger AFTER UPDATE

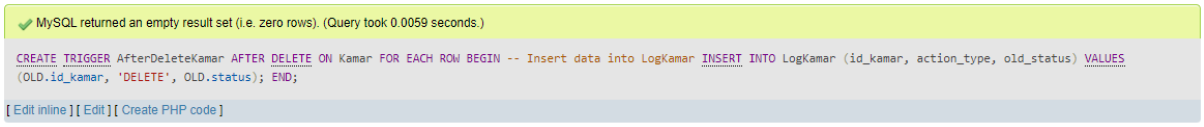
```
CREATE TRIGGER AfterUpdateKamar
AFTER UPDATE ON Kamar
FOR EACH ROW
BEGIN
    INSERT INTO LogKamar (id_kamar, action_type, old_status,
new_status)
    VALUES (OLD.id_kamar, 'UPDATE', OLD.status, NEW.status);
END;
```



Trigger AFTER DELETE

```
CREATE TRIGGER AfterDeleteKamar
```

```
AFTER DELETE ON Kamar
FOR EACH ROW
BEGIN
    -- Insert data into LogKamar
    INSERT INTO LogKamar (id_kamar, action_type, old_status)
    VALUES (OLD.id_kamar, 'DELETE', OLD.status);
END;
```



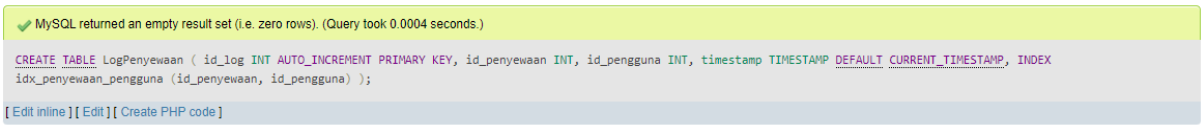
Daftar TRIGGER

```
SHOW TRIGGERS WHERE `Table` = 'Kamar';
```

2.4 Idex

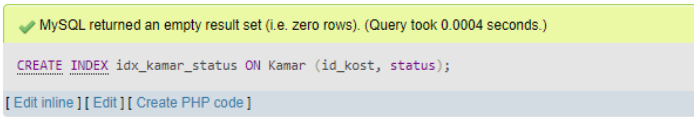
Membuat index saat membuat tabel baru

```
CREATE TABLE LogPenyewaan (
    id_log INT AUTO_INCREMENT PRIMARY KEY,
    id_penyewaan INT,
    id_pengguna INT,
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    INDEX idx_penyewaan_pengguna (id_penyewaan, id_pengguna)
);
```



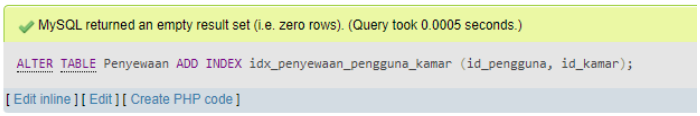
Membuat index dengan CREATE INDEX

```
CREATE INDEX idx_kamar_status ON Kamar (id_kost, status);
```



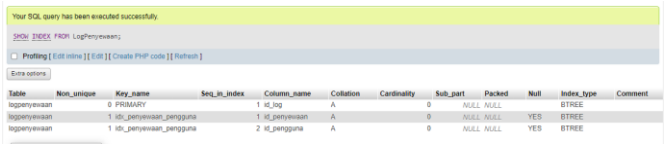
Membuat index dengan ALTER TABLE

```
ALTER TABLE Penyewaan
ADD INDEX idx_penyewaan_pengguna_kamar (id_pengguna, id_kamar);
```



Dafter Index

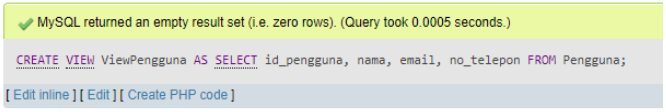
```
SHOW INDEX FROM LogPenyewaan;
SHOW INDEX FROM Kamar;
SHOW INDEX FROM Penyewaan;
```



2.5 View

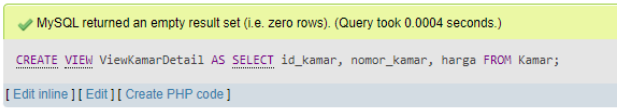
Horizontal View

```
CREATE VIEW ViewPengguna AS
SELECT id_pengguna, nama, email, no_telepon FROM Pengguna;
```



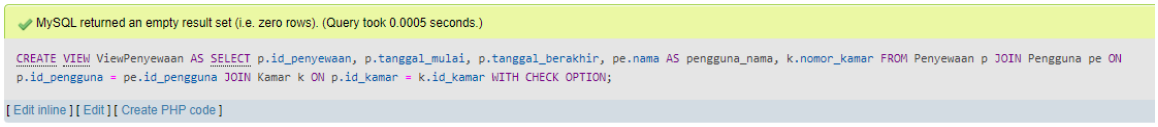
Vertikal View

```
CREATE VIEW ViewKamarDetail AS
SELECT id_kamar, nomor_kamar, harga FROM Kamar;
```



View inside View

```
CREATE VIEW ViewPenyewaan AS
SELECT p.id_penyewaan, p.tanggal_mulai, p.tanggal_berakhir,
pe.nama AS pengguna_nama, k.nomor_kamar
FROM Penyewaan p
JOIN Pengguna pe ON p.id_pengguna = pe.id_pengguna
JOIN Kamar k ON p.id_kamar = k.id_kamar
WITH CHECK OPTION;
```

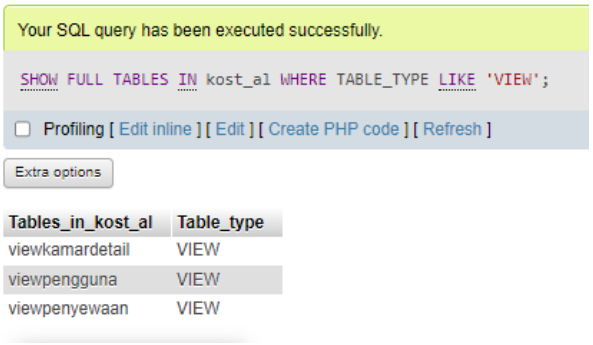


Eksekusi Perintah untuk Melakukan Update dan Insert Menggunakan View

```
INSERT INTO ViewPenyewaan (id_pengguna, id_kamar,
tanggal_mulai) VALUES (1, 2, '2023-07-01');
UPDATE ViewPenyewaan SET tanggal_berakhir = '2023-12-31' WHERE
id_penyewaan = 1;
```

Daftar View

```
SHOW FULL TABLES IN kost_al WHERE TABLE_TYPE LIKE 'VIEW';
```



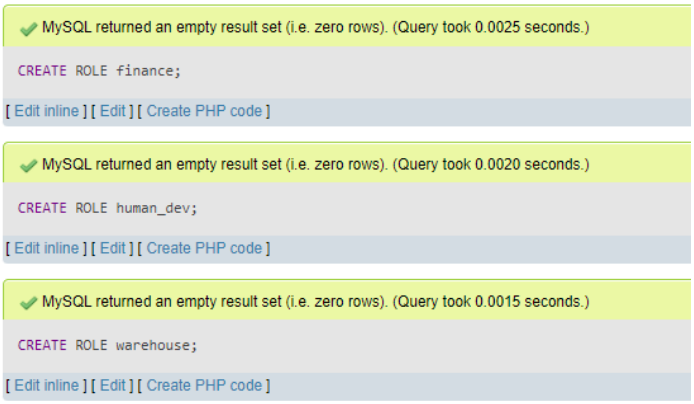
2.6 Database Security

Membuat User Baru

```
CREATE USER 'user1'@'localhost' IDENTIFIED BY 'password1';
CREATE USER 'user2'@'localhost' IDENTIFIED BY 'password2';
CREATE USER 'user3'@'localhost' IDENTIFIED BY 'password3';
```

Membuat Role Baru

```
CREATE ROLE finance;
CREATE ROLE human_dev;
CREATE ROLE warehouse;
```



Memberikan Privilege kepada User dan Role

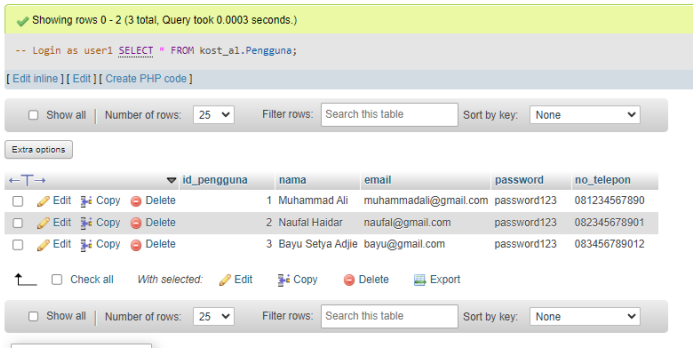
```
GRANT SELECT ON kost_al.Pengguna TO 'user1'@'localhost';
GRANT SELECT ON kost_al.ViewPengguna TO 'user2'@'localhost';
GRANT EXECUTE ON PROCEDURE kost_al.AddPenyewaan TO finance;
```

Eksekusi untuk Membuktikan Privilege

```
-- Login as user1
SELECT * FROM kost_al.Pengguna;


-- Login as user2
SELECT * FROM kost_al.ViewPengguna;
```

```
-- Login as a user with the finance role
CALL kost_al.AddPenyewaan(1, 2);
```



Daftar User dan Privilege

```
SELECT User, Host FROM mysql.user;
SHOW GRANTS FOR 'user1'@'localhost';
SHOW GRANTS FOR 'user2'@'localhost';
SHOW GRANTS FOR 'finance';
```