

LAPORAN

“Desain dan Pengembangan Database pada Kost Poetra Sultan: Aplikasi Praktis Diagram ER dan SQL”

Diajukan untuk memenuhi salah satu praktikum Mata Kuliah Basis Data yang di ampu oleh:

Dewi Soyusiawaty, S.T., M.T. Teknik Informatika



Disusun Oleh:

Mohammad Farid Hendianto 2200018401 A

PROGRAM STUDI INFORMATIKA
UNIVERSITAS AHMAD DAHLAN
FAKULTAS TEKNOLOGI INDUSTRI
TAHUN 2024

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan dengan judul “Desain dan Pengembangan Database pada Kost Poetra Sultan: Aplikasi Praktis Diagram ER dan SQL” dengan baik dan tepat waktu.

Laporan ini disusun untuk memenuhi salah satu tugas mata kuliah Basis Data yang diampu oleh Ibu Dewi Soyusiawaty, S.T., M.T. Penulis menyadari bahwa laporan ini masih jauh dari sempurna karena keterbatasan pengetahuan dan pengalaman yang penulis miliki. Oleh karena itu, penulis sangat mengharapkan kritik dan saran dari pembaca untuk perbaikan di masa mendatang.

Dalam kesempatan ini, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah membantu dan membimbing penulis dalam menyelesaikan laporan ini, terutama kepada:

1. Ibu Dewi Soyusiawaty, S.T., M.T. selaku dosen pengampu mata kuliah Basis Data yang telah memberikan ilmu dan bimbingan kepada penulis.
2. Segenap dosen dan staf Program Studi Informatika Universitas Ahmad Dahlan yang telah membekali penulis dengan berbagai ilmu pengetahuan.
3. Teman-teman mahasiswa Program Studi Informatika Angkatan 2022 yang telah memberikan dukungan dan semangat kepada penulis.
4. Serta semua pihak yang tidak dapat penulis sebutkan satu-persatu yang telah membantu penulis dalam menyelesaikan laporan ini.

Penulis berharap semoga laporan ini dapat bermanfaat bagi penulis khususnya dan pembaca pada umumnya. Akhir kata, penulis mohon maaf apabila terdapat kesalahan dalam penulisan laporan ini.

Yogyakarta, 12 Januari 2024



Penulis

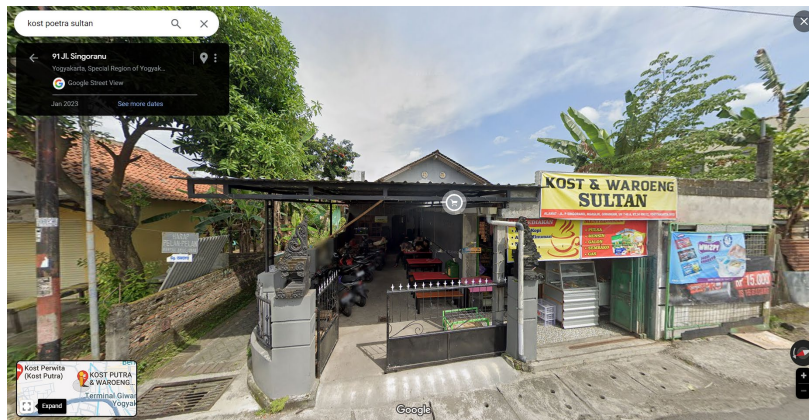
DAFTAR ISI

KATA PENGANTAR.....	2
DAFTAR ISI.....	3
BAB I PENDAHULUAN.....	4
1.1. Latar Belakang Masalah	4
1.2. Rumusan Masalah	6
1.3. Tujuan Penulisan	6
1.4. Manfaat Penulisan	6
BAB II PEMBAHASAN.....	7
2.1. Proses Bisnis	7
2.2. Entitas dan Relasi	8
2.2.1. Entitas	8
2.2.2. Relasi	11
2.3. Diagram ER dan Tabel yang dihasilkan.....	14
2.3.1. Diagram ER.....	14
2.3.2. Tabel.....	17
2.4. Penerapan SQL	18
2.4.1. Perencanaan Tipe Data SQL	18
2.4.2. Pembuatan Database	21
2.4.3. Tampilan Tabel Designer	31
2.4.4. Query.....	32
2.4.5. Trigger dan Procedure	49
BAB III PENUTUP.....	65
3.1. Kesimpulan	65
3.2. Kritik dan Saran	67
DAFTAR GAMBAR.....	68

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah



Gambar 1 Lokasi Kost Poetra Sultan.

Saat ini banyak mahasiswa perantauan yang memilih untuk tinggal di rumah kost selama menempuh pendidikan di Yogyakarta. Salah satu kost yang banyak diminati adalah Kost Poetra Sultan yang beralamat di Jl. P. Singoranu, Ngaglik Giwangan, UH 7/40A, RT.34, RW.12, Yogyakarta. 55163.

Kost Poetra Sultan memiliki 12 kamar dengan daya tampung 1 penghuni per kamar, memiliki 2 jenis kamar, yaitu 8 kamar berjenis tanpa kamar mandi dan 4 kamar dengan kamar mandi dalam. Sebagian besar penghuninya adalah mahasiswa Universitas Ahmad Dahlan. Kost ini berlokasi strategis dekat dengan kampus dan fasilitas umum seperti minimarket dan rumah makan.

Sebagai penghuni kost, penulis menghadapi beberapa masalah yang cukup mengganggu. Pertama, kurangnya transparansi dalam hal keuangan seperti biaya listrik, air, dan kebersihan seringkali menjadi sumber konflik antara penghuni dan pemilik kost. Setiap bulan, penghuni seharusnya membayar biaya tagihan listrik/air. Namun pemilik kost tidak pernah menginformasikan secara detail penggunaan uang iuran tersebut.

Hal ini menimbulkan kecurigaan di kalangan penghuni bahwa uang iuran tidak digunakan sepenuhnya untuk kepentingan perawatan kost. Beberapa penghuni pernah memprotes pemilik kost soal transparansi penggunaan uang iuran, namun pemilik kost bersikukuh uang tersebut digunakan untuk kebutuhan perawatan seperti untuk menggaji petugas kebersihan. Namun karena tidak ada bukti laporan penggunaan dana, penghuni masih merasa kurang yakin.

Kedua, kualitas wifi di Kost Poetra Sultan seringkali tidak konsisten dan lambat. Padahal sebagian besar penghuni membutuhkan koneksi internet yang stabil untuk keperluan

belajar dan bekerja. Jaringan wifi sering tiba-tiba terputus atau sulit tersambung jika digunakan beberapa orang secara bersamaan.

Penghuni sudah beberapa kali menyampaikan keluhan soal wifi ke pemilik kost. Namun hingga saat ini, kualitas wifi tidak juga membaik. Hal ini sangat mengganggu aktivitas penghuni kost yang bergantung pada koneksi internet.

Ketiga, tidak adanya fitur yang dapat menunjukkan berapa lama penghuni telah tinggal di kost. Informasi ini penting agar penghuni dapat merencanakan dan mengatur keuangannya dengan baik selama masa tinggal di kost. Misalnya, berapa bulan lagi kontrak kost akan habis sehingga penghuni bisa mempersiapkan diri untuk memperpanjang atau mencari kost yang baru.

Saat ini, pemilik kost mencatat manual di buku tentang tanggal masuk penghuni baru. Namun informasi itu tidak pernah diberitahukan ke penghuni secara berkala. Akibatnya, terkadang penghuni lupa sudah berapa lama tinggal di kost. Ketika hendak memperpanjang sewa, mereka baru menyadari bahwa masa kontraknya sudah mau habis.

Keempat, informasi tentang kost di website Kost Poetra Sultan sangat minim. Calon penghuni kesulitan mendapatkan info penting seperti berapa biaya sewa per bulan, fasilitas apa saja yang disediakan, berapa kamar yang masih kosong, dan bagaimana cara pendaftaran. Informasi yang tersedia di website hanya berupa alamat dan nomor telepon kost.

Akibatnya, calon penghuni harus bertanya langsung ke pemilik untuk mendapatkan informasi detail tentang kost. Padahal saat ini, semakin banyak calon penghuni kost yang lebih memilih mencari informasi secara online sebelum datang langsung ke lokasi kost.

Selanjutnya, masalah keamanan dan kenyamanan menjadi perhatian utama bagi penghuni kost. Saat ini, pintu pagar dan pintu kamar tidak dilengkapi kunci ganda sehingga risiko kehilangan kunci dan pencurian cukup tinggi. Juga tidak ada rekaman CCTV sehingga sulit melacak jika terjadi tindak kejahatan di kost.

Beberapa kali terjadi barang hilang di kamar mandi dan dapur umum. Diduga pelakunya adalah orang luar yang secara bebas keluar masuk kost karena pagar kost yang sering dibiarkan terbuka. Namun karena tidak ada bukti rekaman CCTV, sulit mengidentifikasi pelakunya.

Masalah lainnya adalah beberapa penghuni sering tidak bertanggung jawab atas kerusakan fasilitas kost atau terlambat membayar sewa bulanan. Hal ini karena tidak adanya sistem pengenaan denda yang jelas. Akibatnya, kerusakan yang terjadi tidak segera diperbaiki dan utang sewa menumpuk. Pemilik kost tentu dirugikan karena harus mengeluarkan biaya tambahan untuk perbaikan.

Berdasarkan berbagai permasalahan di atas, dibutuhkan sebuah sistem informasi terkomputerisasi yang dapat meningkatkan efisiensi dan efektivitas pengelolaan Kost Poetra Sultan. Dengan menerapkan sistem yang terintegrasi, diharapkan permasalahan yang dialami pemilik dan penghuni kost dapat diminimalisir.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah, dapat dirumuskan beberapa permasalahan yang dihadapi dalam pengelolaan Kost Poetra Sultan, yaitu:

1. Kurangnya transparansi informasi terkait keuangan kost seperti pemakaian listrik, air, dan iuran kebersihan.
2. Lemahnya jaringan wifi yang berdampak pada terganggunya aktivitas penghuni kost.
3. Tidak adanya fitur yang menginformasikan lama waktu menghuni kost oleh setiap penghuni.
4. Terbatasnya informasi terkait ketersediaan kamar, biaya sewa, dan data pemilik yang dapat diakses calon penghuni.
5. Kurangnya sistem keamanan di kost.
6. Tidak adanya sistem denda atas keterlambatan pembayaran dan kerusakan properti kost.

1.3. Tujuan Penulisan

Tujuan dari penelitian ini adalah:

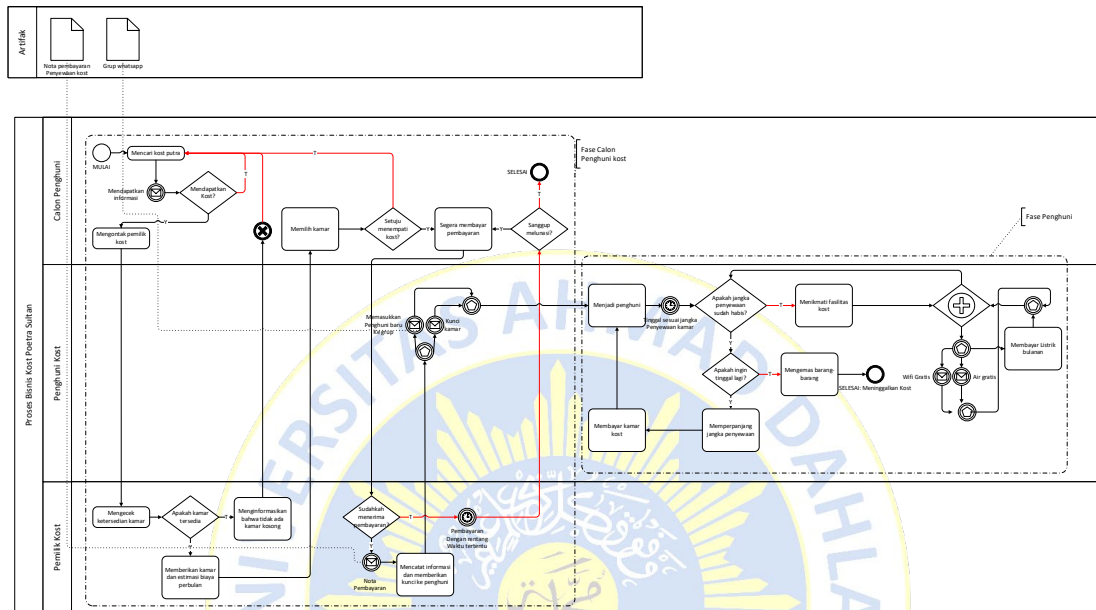
1. Merancang sistem informasi terkomputerisasi untuk Kost Poetra Sultan.
2. Memberikan informasi transparan terkait keuangan kost kepada penghuni.
3. Menyediakan informasi ketersediaan kamar dan data kost yang dapat diakses calon penghuni.
4. Mengimplementasikan fitur yang menginformasikan lama waktu menghuni kost oleh setiap penghuni.
5. Merancang sistem keamanan di kost.
6. Membuat sistem denda atas keterlambatan pembayaran dan kerusakan properti kost.
7. Secara umum, meningkatkan efisiensi dan efektivitas pengelolaan Kost Poetra Sultan.

1.4. Manfaat Penulisan

Manfaat yang diharapkan dari penelitian ini antara lain:

1. Memberikan kemudahan bagi pemilik dan penghuni kost dalam mengelola data dan kegiatan sehari-hari di kost.
2. Meningkatkan efisiensi dan efektivitas pengelolaan keuangan kost.
3. Memberikan kemudahan bagi calon penghuni kost untuk mendapatkan informasi.
4. Meningkatkan keamanan dan kenyamanan penghuni kost.
5. Memudahkan pemantauan kondisi kost oleh pemilik.
6. Sebagai referensi bagi pengembangan sistem serupa di tempat lain.

2.1. Proses Bisnis



Sebelum merancang diagram ER (Entity Relationship) untuk sistem basis data kost Poetra Sultan, langkah pertama yang harus dilakukan adalah memahami proses bisnis yang berjalan saat ini. Pemahaman mengenai proses bisnis yang komprehensif akan memudahkan dalam menentukan entitas dan relasi yang diperlukan dalam diagram ER.

Proses bisnis diawali dengan tahap penyebaran informasi mengenai ketersediaan kamar kosong di Kost Poetra Sultan. Informasi disebarluaskan melalui media sosial, spanduk, brosur, dan website resmi kost. Calon penghuni yang tertarik kemudian menghubungi petugas kost melalui telepon maupun datang langsung untuk menanyakan ketersediaan kamar serta informasi terkait fasilitas dan biaya sewa per bulan.

Kost Poetra Sultan menyediakan dua tipe kamar yaitu kamar standar tanpa kamar mandi dalam dan kamar premium dengan fasilitas kamar mandi dalam. Calon penghuni dapat memilih salah satu tipe kamar berdasarkan preferensi dan budget yang dimiliki.

Setelah tipe kamar ditentukan, calon penghuni akan melakukan pembayaran biaya sewa dan uang jaminan sesuai ketentuan yang berlaku. Biaya sewa minimum adalah untuk jangka waktu 6 bulan atau satu semester. Setelah itu, calon penghuni resmi terdaftar sebagai penghuni baru Kost Poetra Sultan.

Perencanaannya, penghuni baru akan diberikan akses ke sistem informasi kost yang berbasis website dan aplikasi. Melalui sistem tersebut, penghuni dapat melihat informasi tagihan bulanan, melakukan pembayaran secara online, mengecek durasi tinggal, dan mengakses fasilitas wifi serta layanan lainnya.

Selama menjadi penghuni, setiap bulan penghuni diwajibkan membayar tagihan sewa kamar dan utilitas seperti listrik dan air tepat waktu. Apabila terjadi keterlambatan pembayaran, sanksi denda akan diberlakukan. Pelanggaran terhadap peraturan kost lainnya juga akan dikenai sanksi yang dapat berujung pada pemutusan kontrak sewa secara sepihak oleh pihak kost.

Proses bisnis di kost berakhir saat penghuni telah menyelesaikan masa sewa dan melakukan prosedur check-out. Penghuni diharuskan membayar lunas semua tagihan dan kewajiban lainnya sebelum dinyatakan resmi check-out dari Kost Poetra Sultan.

2.2. Entitas dan Relasi

Untuk merancang basis data yang baik, langkah pertama adalah menentukan entitas dan relasi antar entitas berdasarkan proses bisnis yang telah dijelaskan sebelumnya. Kemudian menentukan atribut-atribut untuk masing-masing entitas beserta tipe datanya. Berikut adalah penjelasan mengenai entitas, relasi untuk sistem basis data Kost Poetra Sultan.

2.2.1. Entitas

Entitas merupakan objek nyata atau abstrak yang dapat dibedakan dalam lingkup sistem basis data. Entitas biasanya merepresentasikan objek dunia nyata seperti orang, tempat, benda atau peristiwa yang data-datanya akan disimpan dalam basis data.

Berdasarkan analisis proses bisnis Kost Poetra Sultan, terdapat 4 entitas utama yang terlibat, yaitu:

1. Penghuni

Penghuni merupakan entitas utama dalam sistem basis data kost ini. Entitas penghuni berisi data diri dan informasi terkait penghuni kost. Atribut pada entitas penghuni antara lain:

- idPenghuni (Primary Key)
- namaPenghuni
 - namaDepan
 - namaBelakang
- alamat
 - jalan
 - RT
 - RW
 - kelurahan

- kecamatan
- kota
- provinsi
- kodePos
- tanggalLahir
- umur() (Derived Attribute)
- email (Multivalued Attribute)
- noTelp (Multivalued Attribute)



penghuni	
PK	idPenghuni
	namaPenghuni
	namaDepan
	namaBelakang
	alamat
	jalan
	RT
	RW
	kelurahan
	kecamatan
	kota
	provinsi
	kodePos
	tanggalLahir
	umur()
	{email}
	{noTelp}

Gambar 3 menunjukkan entitas penghuni beserta atribut-atributnya.

Jenis Atribut:

Nama Atribut	Jenis Atribut
idPenghuni	Primary Key
namaPenghuni	Composite Attribute
namaDepan	Component Attribute of namaPenghuni
namaBelakang	Component Attribute of namaPenghuni
alamat	Composite Attribute
jalan	Component Attribute of alamat
RT	Component Attribute of alamat
RW	Component Attribute of alamat
kelurahan	Component Attribute of alamat
kecamatan	Component Attribute of alamat
kota	Component Attribute of alamat
provinsi	Component Attribute of alamat
kodePos	Component Attribute of alamat
tanggalLahir	Single Valued and Simple Attribute
umur()	Derived Attributes from tanggalLahir
{email}	Multivalued Attribute

{noTelp}	Multivalued Attribute
----------	-----------------------

2. kamar

Entitas kamar digunakan untuk menyimpan informasi detail setiap kamar pada Kost Poetra Sultan. Atribut entitas kamar meliputi:

- noKamar (Primary Key)
- idTipeKamar (Foreign Key)
- idPenghuni (Foreign Key)

kamar	
PK	<u>noKamar</u>
	idTipeKamar
	idPenghuni

Gambar 4 menampilkan entitas kamar beserta atribut-atributnya.

Jenis Atribut:

Nama Atribut	Jenis Atribut
noKamar	Primary Key
idTipeKamar	Foreign Key
idPenghuni	Foreign Key

3. tipeKamar

Entitas tipeKamar berisi informasi mengenai macam-macam tipe kamar yang tersedia di Kost Poetra Sultan beserta data spesifikasinya. Atribut pada entitas tipeKamar yaitu:

- idTipeKamar (Primary Key)
- tipeKamar
- hargaKamar
- luasKamar

tipeKamar	
PK	<u>idTipeKamar</u>
	tipeKamar
	hargaKamar
	luasKamar

Gambar 5 Entitas tipeKamar.

Nama Atribut	Jenis Atribut
noKamar	Primary Key
idTipeKamar	Foreign Key
idPenghuni	Foreign Key

4. Transaksi

Entitas transaksi digunakan untuk mencatat semua detail transaksi keuangan yang terjadi di Kost Poetra Sultan, seperti transaksi pembayaran sewa kamar, tagihan listrik, air, denda, dan lain-lain. Atribut pada entitas transaksi:

- idTransaksi (Primary Key)
- jenisTransaksi
- keterangan

transaksi	
PK	<u>idTransaksi</u>
	jenisTransaksi
	jumlahBiaya
	keterangan

Gambar 6 entitas transaksi.

Jenis atribut:

Nama Atribut	Jenis Atribut
idTransaksi	Primary Key
jenisTransaksi	Simple Attributes
Keterangan	Simple Attributes

2.2.2. Relasi

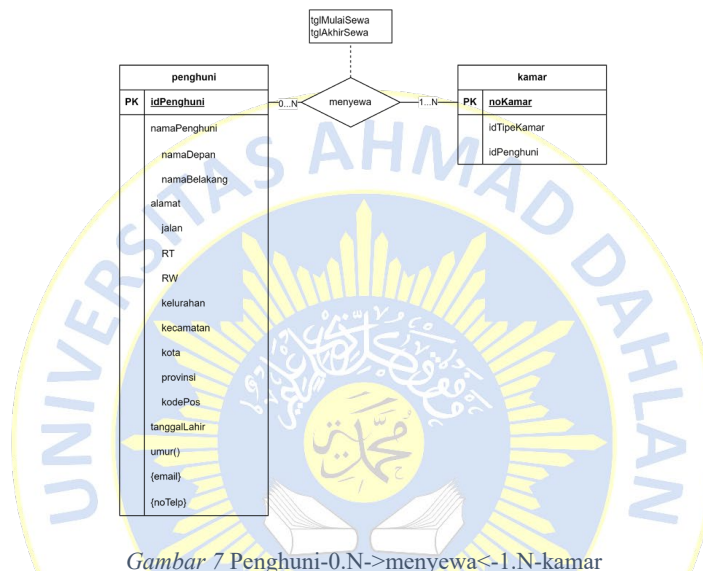
Setelah menentukan entitas, langkah selanjutnya adalah mendefinisikan relasi atau hubungan antar entitas tersebut. Relasi diperlukan agar entitas dapat saling terhubung dan terintegrasi membentuk basis data yang utuh.

Entitas I	KardinalitasEntitasI		Relasi	atribut _{relasi}	KardinalitasEntitasII		Entitas II
	Minimum	Maksimum			Minimum	Maksimum	

penghuni	0	N	menyewa	tglMulaiSewa tglAkhirSewa	1	N	kamar
kamar	1	1	memiliki		1	N	tipeKamar
Agregasi (penghuni, kamar, tipeKamar)	1	1	membayar	tglTransaksi metodePembayaran	1	N	transaksi

Berdasarkan proses bisnis Kost Poetra Sultan, terdapat 3 relasi utama, yaitu:

1. Relasi antara entitas Penghuni dan Kamar

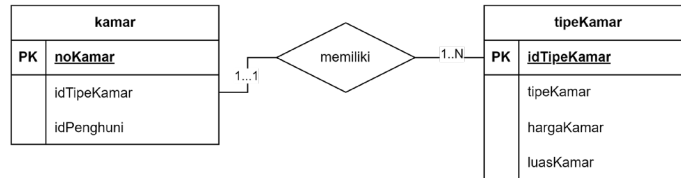


Gambar 7 Penghuni-0.N->menyewa<-1.N-kamar

Relasi ini dinamakan “menyewa” dengan kardinalitas penghuni 0..N menyewa 1..N kamar. Artinya, seorang penghuni dapat menyewa 1 atau lebih kamar, dan satu kamar dapat disewa oleh 1 atau lebih penghuni.

Relasi menyewa memiliki atribut tglMulaiSewa dan tglAkhirSewa yang menunjukkan tanggal mulai dan berakhirnya masa sewa suatu kamar oleh penghuni.

2. Relasi antara entitas Kamar dan TipeKamar

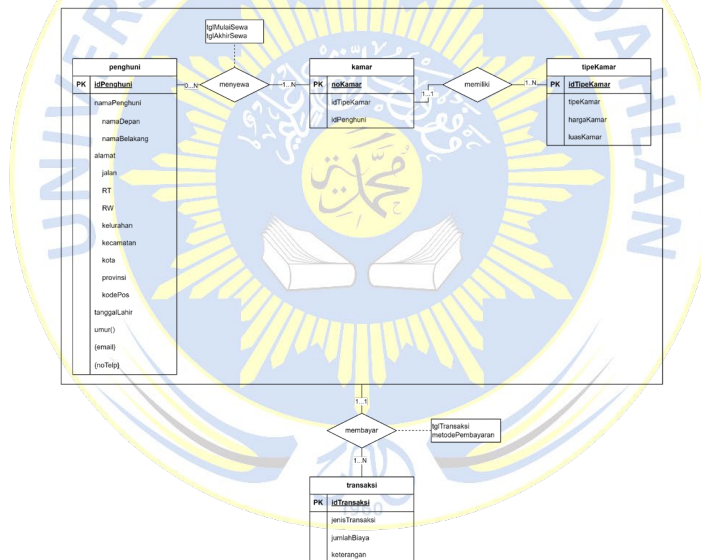


Gambar 8 kamar-1.1->memiliki<-1.N-tipekamar

Relasi antara kamar dan tipeKamar diberi nama “memiliki” dengan kardinalitas kamar 1..1 memiliki 1..N tipeKamar.

Ini berarti bahwa satu kamar hanya bisa memiliki satu tipe kamar, sedangkan satu tipe kamar bisa dimiliki oleh banyak kamar. Misalnya tipe kamar superior bisa dimiliki oleh kamar 101, 102, dan 103.

3. Agregasi antara Penghuni, Kamar, dan TipeKamar ke Transaksi



Gambar 9 Agregasi (penghuni, kamar, tipeKamar)-1.1->membayar<-1.N-transaksi

Entitas penghuni, kamar, dan tipeKamar membentuk agregasi dengan entitas transaksi. Agregasi ini dinamakan “membayar” dengan kardinalitas 1..1 membayar 1..N.

Maksudnya, satu record transaksi hanya dapat melibatkan satu record penghuni, kamar, dan tipe kamar. Sebaliknya, satu record penghuni, kamar, dan tipe kamar dapat terlibat dalam banyak record transaksi.

2.3. Diagram ER dan Tabel yang dihasilkan

Diagram ER (Entity Relationship) digunakan untuk memodelkan struktur data dan hubungan antar data dalam suatu sistem basis data berdasarkan objek-objek nyata yang terlibat. Diagram ER bermanfaat untuk memvisualisasikan entitas beserta atributnya dan juga relasi antar entitas.

Dari diagram ER kemudian dapat dihasilkan rancangan struktur tabel basis data yang mengimplementasikan entitas dan relasi yang telah dimodelkan sebelumnya. Berikut ini adalah penjelasan mengenai diagram ER dan hasil perancangan tabel basis data untuk sistem informasi Kost Poetra Sultan.

2.3.1. Diagram ER

Berdasarkan analisis kebutuhan dan proses bisnis pada Kost Poetra Sultan, telah diidentifikasi 4 entitas yang akan dimodelkan pada diagram ER, yaitu:

1. Penghuni
2. Kamar
3. TipeKamar
4. Transaksi

Keempat entitas tersebut memiliki atribut dan relasi antar entitas sebagai berikut:

Entitas Penghuni

- idPenghuni (Primary Key)
- namaPenghuni
 - namaDepan
 - namaBelakang
- alamat
 - jalan
 - RT
 - RW
 - kelurahan
 - kecamatan
 - kota
 - provinsi
 - kodePos
- tanggalLahir
- umur() (Derived Attribute)
- email (Multivalued Attribute)
- noTelp (Multivalued Attribute)

Entitas Kamar

- noKamar (Primary Key)
- idTipeKamar (Foreign Key)
- idPenghuni (Foreign Key)

Entitas TipeKamar

- idTipeKamar (Primary Key)
- tipeKamar
- hargaKamar
- luasKamar

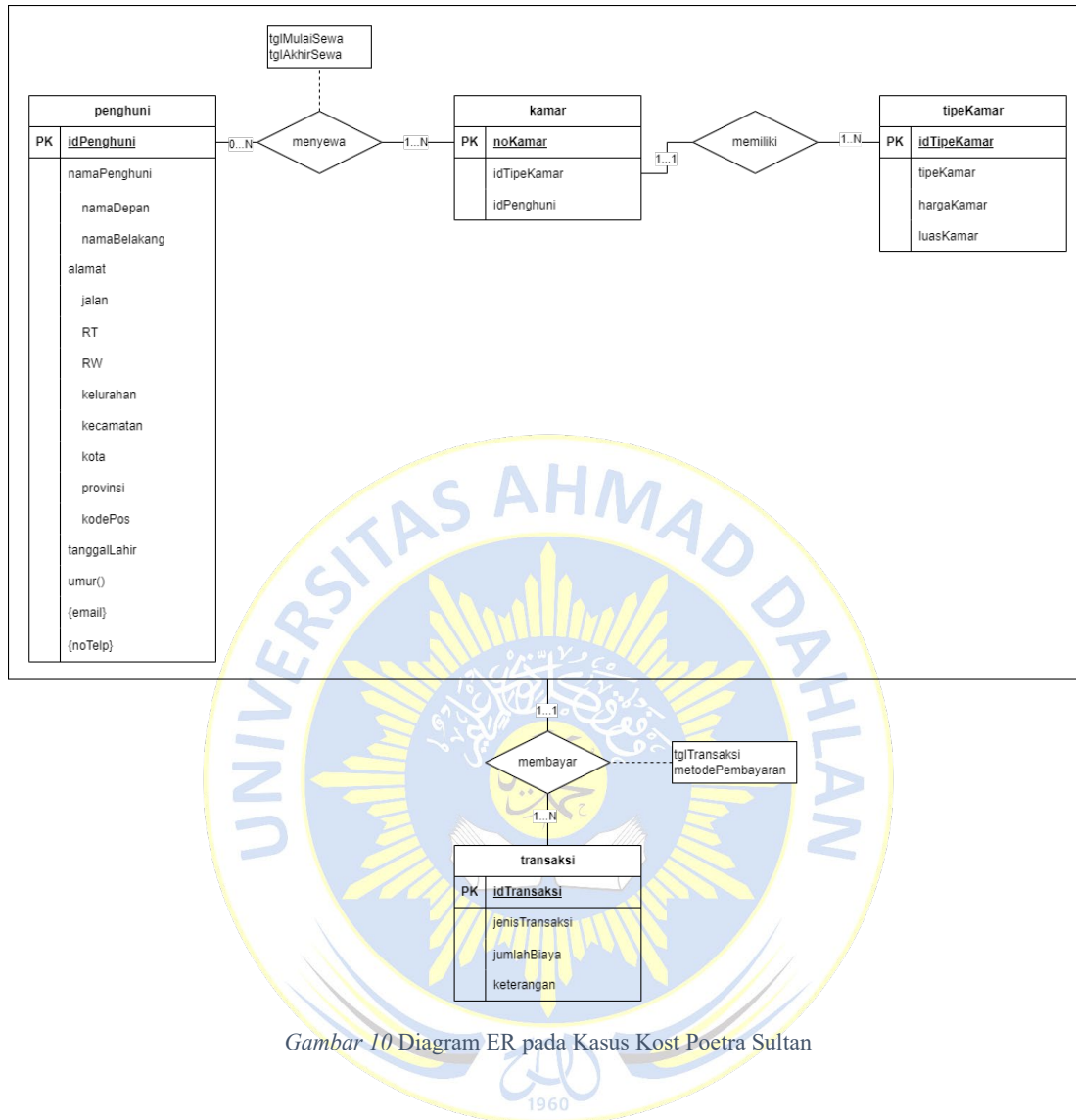
Entitas Transaksi

- idTransaksi (Primary Key)
- jenisTransaksi
- keterangan

Relasi:

- Penghuni (0,N) - menyewa (1,N) -> Kamar
- Kamar (1,1) - memiliki (1,N) -> TipeKamar
- Penghuni (1,1), Kamar (1,1), TipeKamar (1,1) - membayar (1,N) -> Transaksi

Berdasarkan entitas dan relasi tersebut, dapat dibentuk diagram ER untuk sistem basis data Kost Poetra Sultan seperti pada Gambar 10 berikut:

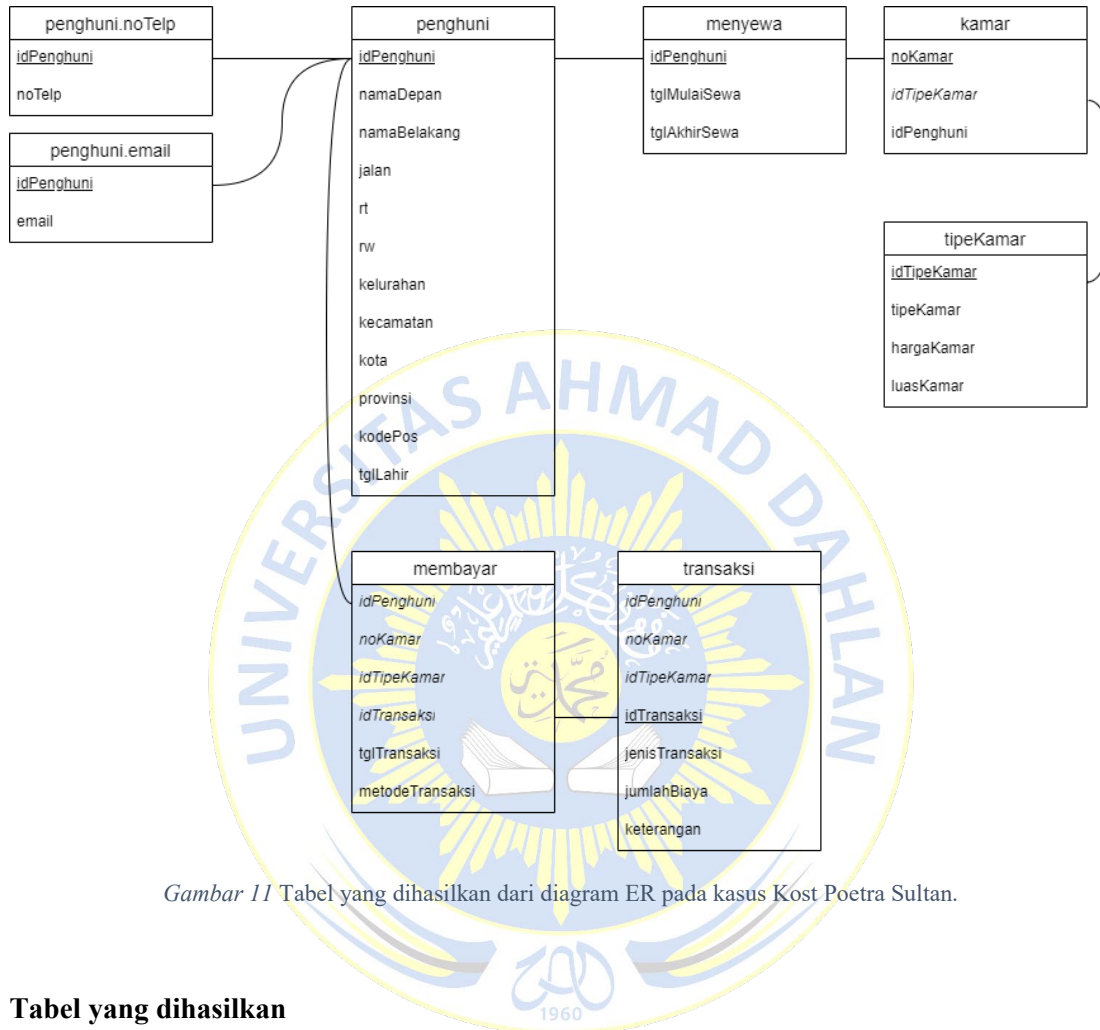


Gambar 10 Diagram ER pada Kasus Kost Poetra Sultan

Pada diagram ER di atas terlihat bahwa entitas Penghuni memiliki relasi banyak-ke-banyak (many to many) dengan entitas Kamar yang dinamakan menyewa. Relasi ini perlu diubah menjadi relasi satu-ke-banyak (one to many) dengan cara menambahkan entitas associative Menyewa di tengah. Entitas Kamar memiliki relasi satu-ke-banyak (one to many) dengan TipeKamar bernama memiliki. Sedangkan entitas Penghuni, Kamar, dan TipeKamar membentuk agregasi bernama membayar dengan entitas Transaksi. Dengan diagram ER, model data pada sistem basis data Kost Poetra Sultan menjadi lebih mudah dipahami. Diagram ER memastikan bahwa basis data yang dibangun nanti memiliki struktur data yang akurat dan relasi yang terdefinisi dengan baik.

2.3.2. Tabel

Tabel yang dihasilkan



Gambar 11 Tabel yang dihasilkan dari diagram ER pada kasus Kost Poetra Sultan.

Tabel yang dihasilkan

1. penghuni(idPenghuni,namaDepan, namaBelakang, jalan, RT, RW, kelurahan, kecamatan,kota,provinsi,kodePos,tanggalLahir)

Tabel penghuni digunakan untuk menyimpan data profil lengkap para penghuni kost.

2. penghuni.email(idPenghuni,email)

Tabel penghuni_email berelasi dengan tabel penghuni untuk menyimpan kumpulan alamat email milik masing-masing penghuni.

3. penghuni.noTelp(idPenghuni,noTelp)

Tabel penghuni_noTelp berelasi dengan tabel penghuni untuk mencatat semua nomor telepon yang dimiliki oleh masing-masing penghuni.

4. menyewa(idPenghuni,tglMulaiSewa,tglAkhirSewa)

Tabel menyewa digunakan untuk mencatat tanggal mulai dan tanggal berakhir sewa kamar oleh masing-masing penghuni.

5. kamar(noKamar,idTipeKamar,idPenghuni)

Tabel kamar berisi detail masing-masing kamar kost beserta tipe kamar dan data penghuninya.

6. tipeKamar(idTipeKamar,tipeKamar,hargaKamar,luasKamar)

Tabel tipeKamar digunakan untuk mencatat daftar tipe kamar yang tersedia beserta spesifikasinya.

7. membayar(idPenghuni, noKamar, idTipeKamar, idTransaksi, tglTransaksi, metodePembayaran)

Tabel membayar merupakan tabel perantara antara tabel penghuni, kamar, tipeKamar, dan transaksi untuk mencatat detail pembayaran transaksi oleh penghuni.

8. transaksi(idPenghuni, noKamar, idTipeKamar, idTransaksi, jenisTransaksi, jumlahBiaya, tglPembayaran, keterangan)

Tabel transaksi untuk mencatat semua jenis transaksi keuangan yang terjadi di kost.

Dengan merancang struktur tabel basis data seperti di atas, maka model pada diagram ER dapat diimplementasikan secara fisik menjadi basis data yang siap digunakan aplikasi. Proses normalisasi mungkin masih perlu dilakukan untuk mengoptimalkan struktur tabel yang dihasilkan dari diagram ER. Normalisasi bertujuan untuk mengeliminasi redundansi data dan meningkatkan integritas data pada basis data.

2.4. Penerapan SQL

2.4.1. Perencanaan Tipe Data SQL

Tahap selanjutnya dalam perancangan basis data adalah menentukan tipe data untuk setiap atribut pada setiap entitas. Penentuan tipe data yang tepat akan memastikan bahwa data yang disimpan sesuai dengan karakteristik masing-masing atribut.

Berikut ini perencanaan tipe data untuk entitas pada sistem basis data Kost Poetra Sultan:

1. Tabel Penghuni

```
CREATE TABLE penghuni (
  idPenghuni INT PRIMARY KEY AUTO_INCREMENT,
  namaDepan VARCHAR(255),
  namaBelakang VARCHAR(255),
  jalan VARCHAR(255),
  RT INT,
  RW INT,
  kelurahan VARCHAR(255),
  kecamatan VARCHAR(255),
  kota VARCHAR(255),
  provinsi VARCHAR(255),
  kodePos INT,
  tanggalLahir DATE
);
```

Pada tabel penghuni, idPenghuni sebagai primary key diberi tipe data INT AUTO_INCREMENT agar nilainya unik dan increment secara otomatis.

Nama depan dan belakang sebagai string diberi tipe VARCHAR 255 agar dapat menampung nama panjang. Alamat dibagi menjadi beberapa kolom seperti jalan, RT/RW, kelurahan, kecamatan, kota, dan provinsi bertipe VARCHAR 255 dan kode pos bertipe INT.

Tanggal lahir diberi tipe DATE yang sesuai dengan format tanggal.

2. Tabel penghuni_email

```
CREATE TABLE penghuni_email (
  idPenghuni INT,
  email VARCHAR(255),
  FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE CASCADE
);
```

Tabel penghuni_email digunakan untuk menyimpan data email milik penghuni. Email diberi tipe VARCHAR 255. idPenghuni sebagai foreign key terhubung dengan primary key pada tabel penghuni.

3. Tabel penghuni_notelp

```
CREATE TABLE penghuni_noTelp (
  idPenghuni INT,
  noTelp VARCHAR(255),
  FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE CASCADE
);
```

Tabel penghuni_noTelp untuk menyimpan data nomor telepon penghuni dengan tipe VARCHAR 255. Juga terhubung ke tabel penghuni melalui foreign key idPenghuni.

4. Tabel Menyewa

```
CREATE TABLE menyewa (
    idPenghuni INT,
    tglMulaiSewa DATE,
    tglAkhirSewa DATE,
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE SET CASCADE
);
```

Tabel menyewa untuk mencatat tanggal mulai dan berakhir sewa kamar oleh penghuni. Kedua kolom tanggal diberi tipe DATE. idPenghuni sebagai foreign key terhubung dengan tabel penghuni.

5. Tabel tipekamar

```
CREATE TABLE tipeKamar (
    idTipeKamar INT PRIMARY KEY AUTO_INCREMENT,
    tipeKamar VARCHAR(255),
    hargaKamar DECIMAL(10,2),
    luasKamar DECIMAL(10,2)
);
```

6. Tabel kamar

```
CREATE TABLE kamar (
    noKamar INT PRIMARY KEY AUTO_INCREMENT,
    idTipeKamar INT,
    idPenghuni INT,

    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON DELETE SET NULL,
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE SET NULL
);
```

Pada tabel kamar, noKamar sebagai primary key bertipe INT AUTO_INCREMENT. idTipeKamar dan idPenghuni sebagai foreign key terhubung ke tabel tipeKamar dan penghuni.

7. Tabel transaksi

```
CREATE TABLE transaksi (
    idPenghuni INT,
    noKamar INT,
    idTipeKamar INT,

    idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
    jenisTransaksi ENUM('sewa kamar', 'air', 'listrik', 'denda'),

    jumlahBiaya DECIMAL(10,2),
    keterangan VARCHAR(255),

    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE SET NULL,
    FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET NULL,
    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar)
);
```

8. Tabel membayar

```
CREATE TABLE membayar (
    idPenghuni INT,
    noKamar INT,
    idTipeKamar INT,

    idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
    tglTransaksi DATE,

    metodePembayaran ENUM('tunai', 'non-tunai'),

    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON DELETE SET NULL,
    FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET NULL,
    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON DELETE SET NULL,

    FOREIGN KEY (idTransaksi) REFERENCES transaksi(idTransaksi)
);
```

Tabel membayar untuk mencatat detail transaksi pembayaran oleh penghuni.

idTransaksi primary key AUTO_INCREMENT, tglTransaksi bertipe DATE. Metode pembayaran menggunakan ENUM tunai atau non-tunai.

Foreign key terhubung dengan tabel terkait termasuk transaksi.

Demikian rancangan tabel beserta tipe data yang direncanakan untuk basis data Kost Poetra Sultan. Pemilihan tipe data yang tepat sangat penting agar data yang disimpan konsisten dan efisien. Tipe data juga berpengaruh pada kinerja operasi basis data seperti proses insert, update, search dan lainnya.

Selanjutnya rancangan basis data ini akan digunakan sebagai acuan untuk tahap implementasi pembuatan basis data fisik menggunakan perintah SQL. Rancangan basis data yang matang di awal akan memudahkan proses implementasi dan menghasilkan basis data yang optimal untuk menunjang sistem informasi Kost Poetra Sultan.

2.4.2. Pembuatan Database

Masuk ke dalam database

```
mysql -u root
```

```
iredragonicy@NDIK-PC c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 17
```

```
Server version: 10.4.32-MariaDB mariadb.org binary distribution
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

Jika sudah ada databasenya bisa menghapus

```
DROP DATABASE kostpoetrasultan;
```

Output

```
MariaDB [(none)]> DROP DATABASE kostpoetrasultan;
Query OK, 8 rows affected (0.038 sec)
```

membuat database dan menggunakan database tersebut

```
CREATE DATABASE kostpoetrasultan;
USE kostpoetrasultan
```

```
MariaDB [(none)]> CREATE DATABASE kostpoetrasultan;
Query OK, 1 row affected (0.002 sec)
```

```
MariaDB [(none)]> USE kostpoetrasultan
Database changed
```

Melihat apakah database sudah terbuat atau belum

```
MariaDB [kostpoetrasultan]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| kostpoetrasultan |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
6 rows in set (0.001 sec)
```

Menerapkan ke dalam MySQL, ada beberapa sedikit revisi apabila sudah dihapus maka akan dibuat NULL agar bisa digunakan lagi

```
CREATE TABLE penghuni (  
    idPenghuni INT PRIMARY KEY AUTO_INCREMENT,  
    namaDepan VARCHAR(255),  
    namaBelakang VARCHAR(255),  
    jalan VARCHAR(255),  
    RT INT,  
    RW INT,  
    kelurahan VARCHAR(255),  
    kecamatan VARCHAR(255),  
    kota VARCHAR(255),  
    provinsi VARCHAR(255),  
    kodePos INT,  
    tanggalLahir DATE  
);  
  
CREATE TABLE penghuni_email (  
    idPenghuni INT,  
    email VARCHAR(255),  
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON  
DELETE CASCADE  
);  
  
CREATE TABLE penghuni_noTelp (  
    idPenghuni INT,  
    noTelp VARCHAR(255),  
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON  
DELETE CASCADE  
);  
  
CREATE TABLE menyewa (  
    idPenghuni INT,  
    tglMulaiSewa DATE,  
    tglAkhirSewa DATE,  
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON  
DELETE SET NULL  
);  
  
CREATE TABLE tipeKamar (  
    idTipeKamar INT PRIMARY KEY AUTO_INCREMENT,  
    tipeKamar VARCHAR(255),  
    hargaKamar DECIMAL(10,2),  
    luasKamar DECIMAL(10,2)  
);  
  
CREATE TABLE kamar (  
    noKamar INT PRIMARY KEY AUTO_INCREMENT,
```



```

    idTipeKamar INT,
    idPenghuni INT,
    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON
DELETE SET NULL,
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL
);

CREATE TABLE transaksi (
    idPenghuni INT,
    noKamar INT,
    idTipeKamar INT,
    idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
    jenisTransaksi ENUM('sewa kamar', 'air', 'listrik', 'denda'),
    jumlahBiaya DECIMAL(10,2),
    keterangan VARCHAR(255),
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL,
    FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET
NULL,
    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar)
);

CREATE TABLE membayar (
    idPenghuni INT,
    noKamar INT,
    idTipeKamar INT,
    idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
    tglTransaksi DATE,
    metodePembayaran ENUM('tunai', 'non-tunai'),
    FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL,
    FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET
NULL,
    FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON
DELETE SET NULL,
    FOREIGN KEY (idTransaksi) REFERENCES transaksi(idTransaksi)
);

```

output

```

MariaDB [kostpoetrasultan]> CREATE TABLE penghuni (
->   idPenghuni INT PRIMARY KEY AUTO_INCREMENT,
->   namaDepan VARCHAR(255),

```

```
-> namaBelakang VARCHAR(255),
-> jalan VARCHAR(255),
-> RT INT,
-> RW INT,
-> kelurahan VARCHAR(255),
-> kecamatan VARCHAR(255),
-> kota VARCHAR(255),
-> provinsi VARCHAR(255),
-> kodePos INT,
-> tanggalLahir DATE
-> );
Query OK, 0 rows affected (0.005 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE penghuni_email (
-> idPenghuni INT,
-> email VARCHAR(255),
-> FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL
-> );
Query OK, 0 rows affected (0.004 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE penghuni_noTelp (
-> idPenghuni INT,
-> noTelp VARCHAR(255),
-> FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL
-> );
Query OK, 0 rows affected (0.004 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE menyewa (
-> idPenghuni INT,
-> tglMulaiSewa DATE,
-> tglAkhirSewa DATE,
-> FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL
-> );
Query OK, 0 rows affected (0.003 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE tipeKamar (
-> idTipeKamar INT PRIMARY KEY AUTO_INCREMENT,
-> tipeKamar VARCHAR(255),
-> hargaKamar DECIMAL(10,2),
-> luasKamar DECIMAL(10,2)
-> );
Query OK, 0 rows affected (0.003 sec)
```

```
MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE kamar (
  ->   noKamar INT PRIMARY KEY AUTO_INCREMENT,
  ->   idTipeKamar INT,
  ->   idPenghuni INT,
  ->   FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON
DELETE SET NULL,
  ->   FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL
  -> );
Query OK, 0 rows affected (0.003 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE transaksi (
  ->   idPenghuni INT,
  ->   noKamar INT,
  ->   idTipeKamar INT,
  ->   idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
  ->   jenisTransaksi ENUM('sewa kamar', 'air', 'listrik', 'denda'),
  ->   jumlahBiaya DECIMAL(10,2),
  ->   keterangan VARCHAR(255),
  ->   FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL,
  ->   FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET
NULL,
  ->   FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar)
  -> );
Query OK, 0 rows affected (0.004 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CREATE TABLE membayar (
  ->   idPenghuni INT,
  ->   noKamar INT,
  ->   idTipeKamar INT,
  ->   idTransaksi INT PRIMARY KEY AUTO_INCREMENT,
  ->   tglTransaksi DATE,
  ->   metodePembayaran ENUM('tunai', 'non-tunai'),
  ->   FOREIGN KEY (idPenghuni) REFERENCES penghuni(idPenghuni) ON
DELETE SET NULL,
  ->   FOREIGN KEY (noKamar) REFERENCES kamar(noKamar) ON DELETE SET
NULL,
  ->   FOREIGN KEY (idTipeKamar) REFERENCES tipeKamar(idTipeKamar) ON
DELETE SET NULL,
  ->   FOREIGN KEY (idTransaksi) REFERENCES transaksi(idTransaksi)
  -> );
Query OK, 0 rows affected (0.004 sec)
```

Melihat daftar tabel yang ada di database

```
SHOW TABLES;
```

Output:

```
MariaDB [kostpoetrasultan]> SHOW TABLES;
```

```
+-----+
| Tables_in_kostpoetrasultan |
+-----+
| kamar                      |
| membayar                  |
| menyewa                   |
| penghuni                  |
| penghuni_email            |
| penghuni_notelp           |
| tipekamar                 |
| transaksi                 |
+-----+
8 rows in set (0.000 sec)
```

Berikut adalah contoh data-datanya yang ada di SQL.

```
-- Penghuni
INSERT INTO penghuni (namaDepan, namaBelakang, jalan, RT, RW, kelurahan, kecamatan, kota, provinsi, kodePos, tanggalahir)
VALUES
('Budi', 'Santoso', 'Jl. Melati 123', 6, 12, 'Cipedes', 'Bandung', 'Jawa Barat', 'Jawa Barat', 40135, '1995-08-20'),
('Ahmad', 'Wibowo', 'Jl. Kencana 456', 10, 15, 'Demangan', 'Yogyakarta', 'Yogyakarta', 'DI Yogyakarta', 55241, '1993-03-12'),
('Rudi', 'Susilo', 'Jl. Angrek 789', 14, 18, 'Sukolilo', 'Surabaya', 'Jawa Timur', 'Jawa Timur', 60223, '1994-12-05'),
('Dwi', 'Sulistyo', 'Jl. Dahlia 567', 8, 22, 'Tirta', 'Semarang', 'Jawa Tengah', 'Jawa Tengah', 50148, '1992-06-18'),
('Eko', 'Prasetyo', 'Jl. Mawar 890', 7, 11, 'Ciwidey', 'Bandung', 'Jawa Barat', 'Jawa Barat', 40354, '1996-10-25'),
('Hendra', 'Kusuma', 'Jl. Srikaya 123', 5, 17, 'Kasih', 'Bantul', 'Yogyakarta', 'DI Yogyakarta', 55183, '1994-02-15'),
('Indra', 'Setiawan', 'Jl. Flamboyan 456', 12, 16, 'Kenjeran', 'Surabaya', 'Jawa Timur', 'Jawa Timur', 60122, '1993-08-08'),
('Ade', 'Wijaya', 'Jl. Angrek 789', 9, 23, 'Genuk', 'Semarang', 'Jawa Tengah', 'Jawa Tengah', 50245, '1995-05-20'),
('Bayu', 'Nugroho', 'Jl. Kencur 234', 11, 14, 'Kalasan', 'Sleman', 'Yogyakarta', 'DI Yogyakarta', 55571, '1992-12-12'),
('Dicky', 'Saputra', 'Jl. Mawar 567', 13, 19, 'Mulyorejo', 'Surabaya', 'Jawa Timur', 'Jawa Timur', 60223, '1994-07-17'),
('Fandi', 'Surya', 'Jl. Kamboja 890', 6, 20, 'Banyumanik', 'Semarang', 'Jawa Tengah', 'Jawa Tengah', 50192, '1996-04-05'),
('Galih', 'Purnama', 'Jl. Raya 123', 8, 21, 'Cangkringan', 'Sleman', 'Yogyakarta', 'DI Yogyakarta', 55582, '1993-01-22');

-- Menyewa Kamar
INSERT INTO menyewa (idPenghuni, tglMulaiSewa, tglAkhirSewa)
VALUES
(1, '2022-01-01', '2023-01-01'),
(2, '2022-01-01', '2023-01-01'),
(3, '2022-01-01', '2023-01-01'),
(4, '2022-01-01', '2023-01-01'),
(5, '2022-01-01', '2023-01-01'),
(6, '2022-01-01', '2023-01-01'),
(7, '2022-01-01', '2023-01-01'),
(8, '2022-01-01', '2023-01-01'),
(9, '2022-01-01', '2023-01-01'),
(10, '2022-01-01', '2023-01-01'),
(11, '2022-01-01', '2023-01-01'),
(12, '2022-01-01', '2023-01-01');

-- Penghuni Email
INSERT INTO penghuni_email (idPenghuni, email)
VALUES
(1, 'budi.santoso@gmail.com'),
(1, 'budi.santoso@yahoo.com'),
(1, 'budi.santoso@outlook.com'),
(2, 'ahmad.wibowo@gmail.com'),
(2, 'ahmad.wibowo@yahoo.com'),
(3, 'rudi.susilo@gmail.com'),
(3, 'rudi.susilo@yahoo.com'),
(3, 'rudi.susilo@outlook.com'),
(4, 'dwi.sulisty@gmail.com'),
(4, 'dwi.sulisty@yahoo.com'),
(5, 'eko.prasetyo@gmail.com'),
(5, 'eko.prasetyo@yahoo.com'),
(5, 'eko.prasetyo@outlook.com'),
(6, 'hendra.kusuma@gmail.com'),
(6, 'hendra.kusuma@yahoo.com'),
(7, 'indra.setiawan@gmail.com'),
(8, 'ade.wijaya@gmail.com'),
(9, 'bayu.nugroho@gmail.com');
```

```

(9, 'bayu.nugroho@yahoo.com'),
(9, 'bayu.nugroho@outlook.com'),
(10, 'dicky.saputra@gmail.com'),
(11, 'fandi.surya@gmail.com'),
(12, 'galih.purnama@gmail.com');

-- Penghuni NoTelp
INSERT INTO penghuni_noTelp (idPenghuni, noTelp)
VALUES
(1, '081234567890'),
(2, '081234567890'),
(2, '081112233445'),
(3, '081234567890'),
(4, '081112233445'),
(5, '081234567890'),
(5, '081112233445'),
(6, '081234567890'),
(7, '081112233445'),
(8, '081234567890'),
(9, '081112233445'),
(10, '081234567890'),
(10, '081112233445'),
(11, '081234567890'),
(11, '081112233445'),
(12, '081234567890'),
(12, '081112233445');

-- Tipe Kamar
INSERT INTO tipeKamar (tipeKamar, hargaKamar, luasKamar)
VALUES
('Tanpa Kamar Mandi', 800000.00, 12.5),
('Dengan Kamar Mandi', 1200000.00, 15.0);

-- Kamar
INSERT INTO kamar (idTipeKamar, idPenghuni) VALUES
(1,1),(1,2),(1,3),(1,4),(2,5),(1,6),(1,7),(1,8),(2,9),(1,10),(2,11),(2,12);

-- Transaksi Januari
INSERT INTO transaksi (idPenghuni, noKamar, idTipeKamar, jenisTransaksi, jumlahBiaya, keterangan)
VALUES
(1, 1, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(1, 1, 1, 'listrik', 200000, 'Listrik bulan Januari'),
(1, 1, 1, 'air', 150000, 'Air bulan Januari'),
(2, 2, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(2, 2, 1, 'listrik', 250000, 'Listrik bulan Januari'),
(2, 2, 1, 'air', 180000, 'Air bulan Januari'),
(3, 3, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(3, 3, 1, 'listrik', 180000, 'Listrik bulan Januari'),
(3, 3, 1, 'air', 130000, 'Air bulan Januari'),
(4, 4, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(4, 4, 1, 'listrik', 220000, 'Listrik bulan Januari'),
(4, 4, 1, 'air', 160000, 'Air bulan Januari'),
(5, 5, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(5, 5, 2, 'listrik', 190000, 'Listrik bulan Januari'),
(5, 5, 2, 'air', 140000, 'Air bulan Januari'),
(6, 6, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(6, 6, 1, 'listrik', 240000, 'Listrik bulan Januari'),
(6, 6, 1, 'air', 170000, 'Air bulan Januari'),
(7, 7, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(7, 7, 1, 'listrik', 170000, 'Listrik bulan Januari'),
(7, 7, 1, 'air', 120000, 'Air bulan Januari'),
(8, 8, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(8, 8, 1, 'listrik', 210000, 'Listrik bulan Januari'),
(8, 8, 1, 'air', 150000, 'Air bulan Januari'),
(9, 9, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(9, 9, 2, 'listrik', 200000, 'Listrik bulan Januari'),
(9, 9, 2, 'air', 140000, 'Air bulan Januari'),
(10, 10, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(10, 10, 1, 'listrik', 250000, 'Listrik bulan Januari'),
(10, 10, 1, 'air', 180000, 'Air bulan Januari'),
(11, 11, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Januari'),
(11, 11, 2, 'listrik', 180000, 'Listrik bulan Januari'),
(11, 11, 2, 'air', 130000, 'Air bulan Januari'),
(12, 12, 2, 'sewa kamar', 1200000, 'Sewa kamar bulan Januari'),
(12, 12, 2, 'listrik', 220000, 'Listrik bulan Januari'),
(12, 12, 2, 'air', 160000, 'Air bulan Januari');

-- Transaksi Februari
INSERT INTO transaksi (idPenghuni, noKamar, idTipeKamar, jenisTransaksi, jumlahBiaya, keterangan)
VALUES
(1, 1, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),
(1, 1, 1, 'listrik', 200000, 'Listrik bulan Februari'),
(1, 1, 1, 'air', 150000, 'Air bulan Februari'),
(2, 2, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(2, 2, 1, 'listrik', 250000, 'Listrik bulan Februari'),
(2, 2, 1, 'air', 180000, 'Air bulan Februari'),
(3, 3, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),
(3, 3, 1, 'listrik', 180000, 'Listrik bulan Februari'),
(3, 3, 1, 'air', 130000, 'Air bulan Februari'),
(4, 4, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(4, 4, 1, 'listrik', 220000, 'Listrik bulan Februari'),
(4, 4, 1, 'air', 160000, 'Air bulan Februari'),
(5, 5, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),
(5, 5, 2, 'listrik', 190000, 'Listrik bulan Februari'),
(5, 5, 2, 'air', 140000, 'Air bulan Februari'),
(6, 6, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(6, 6, 1, 'listrik', 240000, 'Listrik bulan Februari'),
(6, 6, 1, 'air', 170000, 'Air bulan Februari'),
(7, 7, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),

```



```

(7, 7, 1, 'listrik', 170000, 'Listrik bulan Februari'),
(7, 7, 1, 'air', 120000, 'Air bulan Februari'),
(8, 8, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(8, 8, 1, 'listrik', 210000, 'Listrik bulan Februari'),
(8, 8, 1, 'air', 150000, 'Air bulan Februari'),
(9, 9, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),
(9, 9, 2, 'listrik', 200000, 'Listrik bulan Februari'),
(9, 9, 2, 'air', 140000, 'Air bulan Februari'),
(10, 10, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(10, 10, 1, 'listrik', 250000, 'Listrik bulan Februari'),
(10, 10, 1, 'air', 180000, 'Air bulan Februari'),
(11, 11, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Februari'),
(11, 11, 2, 'listrik', 180000, 'Listrik bulan Februari'),
(11, 11, 2, 'air', 130000, 'Air bulan Februari'),
(12, 12, 2, 'sewa kamar', 1200000, 'Sewa kamar bulan Februari'),
(12, 12, 2, 'listrik', 220000, 'Listrik bulan Februari'),
(12, 12, 2, 'air', 160000, 'Air bulan Februari');

-- Transaksi Maret
INSERT INTO transaksi (idPenghuni, noKamar, idTipeKamar, jenisTransaksi, jumlahBiaya, keterangan)
VALUES
(1, 1, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(1, 1, 1, 'listrik', 200000, 'Listrik bulan Maret'),
(1, 1, 1, 'air', 150000, 'Air bulan Maret'),
(2, 2, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(2, 2, 1, 'listrik', 250000, 'Listrik bulan Maret'),
(2, 2, 1, 'air', 180000, 'Air bulan Maret'),
(3, 3, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(3, 3, 1, 'listrik', 180000, 'Listrik bulan Maret'),
(3, 3, 1, 'air', 130000, 'Air bulan Maret'),
(4, 4, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(4, 4, 1, 'listrik', 220000, 'Listrik bulan Maret'),
(4, 4, 1, 'air', 160000, 'Air bulan Maret'),
(5, 5, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(5, 5, 2, 'listrik', 190000, 'Listrik bulan Maret'),
(5, 5, 2, 'air', 140000, 'Air bulan Maret'),
(6, 6, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(6, 6, 1, 'listrik', 240000, 'Listrik bulan Maret'),
(6, 6, 1, 'air', 170000, 'Air bulan Maret'),
(7, 7, 1, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(7, 7, 1, 'listrik', 170000, 'Listrik bulan Maret'),
(7, 7, 1, 'air', 120000, 'Air bulan Maret'),
(8, 8, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(8, 8, 1, 'listrik', 210000, 'Listrik bulan Maret'),
(8, 8, 1, 'air', 150000, 'Air bulan Maret'),
(9, 9, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(9, 9, 2, 'listrik', 200000, 'Listrik bulan Maret'),
(9, 9, 2, 'air', 140000, 'Air bulan Maret'),
(10, 10, 1, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(10, 10, 1, 'listrik', 250000, 'Listrik bulan Maret'),
(10, 10, 1, 'air', 180000, 'Air bulan Maret'),
(11, 11, 2, 'sewa kamar', 800000, 'Sewa kamar bulan Maret'),
(11, 11, 2, 'listrik', 180000, 'Listrik bulan Maret'),
(11, 11, 2, 'air', 130000, 'Air bulan Maret'),
(12, 12, 2, 'sewa kamar', 1200000, 'Sewa kamar bulan Maret'),
(12, 12, 2, 'listrik', 220000, 'Listrik bulan Maret'),
(12, 12, 2, 'air', 160000, 'Air bulan Maret'),
(1, 1, 1, 'denda', 50000, 'Keterlambatan pembayaran sewa kamar bulan Maret'),
(4, 4, 1, 'denda', 60000, 'Keterlambatan pembayaran sewa kamar bulan Maret');

--membayar Januari
INSERT INTO membayar (idPenghuni, noKamar, idTipeKamar, idTransaksi, tglTransaksi, metodePembayaran)
VALUES
(1, 1, 1, 1, '2023-01-01', 'tunai'),
(1, 1, 1, 2, '2023-01-05', 'non-tunai'),
(1, 1, 1, 3, '2023-01-10', 'non-tunai'),
(2, 2, 1, 4, '2023-01-01', 'tunai'),
(2, 2, 1, 5, '2023-01-07', 'tunai'),
(2, 2, 1, 6, '2023-01-12', 'non-tunai'),
(3, 3, 1, 7, '2023-01-01', 'non-tunai'),
(3, 3, 1, 8, '2023-01-06', 'tunai'),
(3, 3, 1, 9, '2023-01-11', 'non-tunai'),
(4, 4, 2, 10, '2023-01-01', 'tunai'),
(4, 4, 2, 11, '2023-01-08', 'non-tunai'),
(4, 4, 2, 12, '2023-01-13', 'tunai'),
(5, 5, 1, 13, '2023-01-01', 'non-tunai'),
(5, 5, 1, 14, '2023-01-09', 'tunai'),
(5, 5, 1, 15, '2023-01-14', 'tunai'),
(6, 6, 2, 16, '2023-01-01', 'non-tunai'),
(6, 6, 2, 17, '2023-01-10', 'non-tunai'),
(6, 6, 2, 18, '2023-01-15', 'tunai'),
(7, 7, 1, 19, '2023-01-01', 'non-tunai'),
(7, 7, 1, 20, '2023-01-11', 'tunai'),
(7, 7, 1, 21, '2023-01-16', 'tunai'),
(8, 8, 1, 22, '2023-01-01', 'non-tunai'),
(8, 8, 1, 23, '2023-01-12', 'non-tunai'),
(8, 8, 1, 24, '2023-01-17', 'tunai'),
(9, 9, 2, 25, '2023-01-01', 'non-tunai'),
(9, 9, 2, 26, '2023-01-13', 'non-tunai'),
(9, 9, 2, 27, '2023-01-18', 'tunai'),
(10, 10, 1, 28, '2023-01-01', 'tunai'),
(10, 10, 1, 29, '2023-01-14', 'non-tunai'),
(10, 10, 1, 30, '2023-01-19', 'tunai'),
(11, 11, 2, 31, '2023-01-01', 'non-tunai'),
(11, 11, 2, 32, '2023-01-15', 'tunai'),
(11, 11, 2, 33, '2023-01-20', 'non-tunai'),
(12, 12, 2, 34, '2023-01-01', 'tunai'),
(12, 12, 2, 35, '2023-01-16', 'non-tunai'),
(12, 12, 2, 36, '2023-01-21', 'tunai');

```

```
-- membayar Februari
INSERT INTO membayar (idPenghuni, noKamar, idTipeKamar, idTransaksi, tglTransaksi, metodePembayaran)
VALUES
(1, 1, 1, 37, '2023-02-01', 'tunai'),
(1, 1, 1, 38, '2023-02-05', 'non-tunai'),
(1, 1, 1, 39, '2023-02-10', 'non-tunai'),
(2, 2, 1, 40, '2023-02-01', 'tunai'),
(2, 2, 1, 41, '2023-02-07', 'tunai'),
(2, 2, 1, 42, '2023-02-12', 'non-tunai'),
(3, 3, 1, 43, '2023-02-01', 'non-tunai'),
(3, 3, 1, 44, '2023-02-06', 'tunai'),
(3, 3, 1, 45, '2023-02-11', 'non-tunai'),
(4, 4, 2, 46, '2023-02-01', 'tunai'),
(4, 4, 2, 47, '2023-02-08', 'non-tunai'),
(4, 4, 2, 48, '2023-02-13', 'tunai'),
(5, 5, 1, 49, '2023-02-01', 'non-tunai'),
(5, 5, 1, 50, '2023-02-09', 'tunai'),
(5, 5, 1, 51, '2023-02-14', 'tunai'),
(6, 6, 2, 52, '2023-02-01', 'non-tunai'),
(6, 6, 2, 53, '2023-02-10', 'non-tunai'),
(6, 6, 2, 54, '2023-02-15', 'tunai'),
(7, 7, 1, 55, '2023-02-01', 'non-tunai'),
(7, 7, 1, 56, '2023-02-11', 'tunai'),
(7, 7, 1, 57, '2023-02-16', 'tunai'),
(8, 8, 1, 58, '2023-02-01', 'non-tunai'),
(8, 8, 1, 59, '2023-02-12', 'non-tunai'),
(8, 8, 1, 60, '2023-02-17', 'tunai'),
(9, 9, 2, 61, '2023-02-01', 'non-tunai'),
(9, 9, 2, 62, '2023-02-13', 'non-tunai'),
(9, 9, 2, 63, '2023-02-18', 'tunai'),
(10, 10, 1, 64, '2023-02-01', 'tunai'),
(10, 10, 1, 65, '2023-02-14', 'non-tunai'),
(10, 10, 1, 66, '2023-02-19', 'tunai'),
(11, 11, 2, 67, '2023-02-01', 'non-tunai'),
(11, 11, 2, 68, '2023-02-15', 'tunai'),
(11, 11, 2, 69, '2023-02-20', 'non-tunai'),
(12, 12, 2, 70, '2023-02-01', 'tunai'),
(12, 12, 2, 71, '2023-02-16', 'non-tunai'),
(12, 12, 2, 72, '2023-02-21', 'tunai');

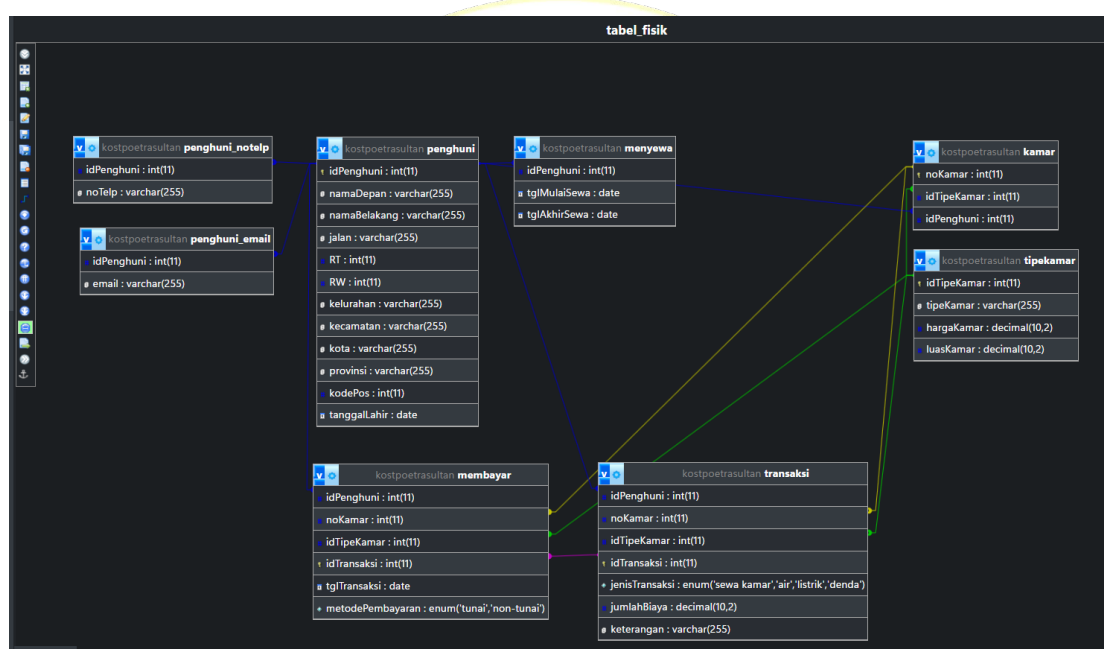
-- membayar Maret
INSERT INTO membayar (idPenghuni, noKamar, idTipeKamar, idTransaksi, tglTransaksi, metodePembayaran)
VALUES
(1, 1, 1, 73, '2023-03-01', 'tunai'),
(1, 1, 1, 74, '2023-03-05', 'non-tunai'),
(1, 1, 1, 75, '2023-03-10', 'non-tunai'),
(2, 2, 1, 76, '2023-03-01', 'tunai'),
(2, 2, 1, 77, '2023-03-07', 'tunai'),
(2, 2, 1, 78, '2023-03-12', 'non-tunai'),
(3, 3, 1, 79, '2023-03-01', 'non-tunai'),
(3, 3, 1, 80, '2023-03-06', 'tunai'),
(3, 3, 1, 81, '2023-03-11', 'non-tunai'),
(4, 4, 2, 82, '2023-03-01', 'tunai'),
(4, 4, 2, 83, '2023-03-08', 'non-tunai'),
(4, 4, 2, 84, '2023-03-13', 'tunai'),
(5, 5, 1, 85, '2023-03-01', 'non-tunai'),
(5, 5, 1, 86, '2023-03-09', 'tunai'),
(5, 5, 1, 87, '2023-03-14', 'tunai'),
(6, 6, 2, 88, '2023-03-01', 'non-tunai'),
(6, 6, 2, 89, '2023-03-10', 'non-tunai'),
(6, 6, 2, 90, '2023-03-15', 'tunai'),
(7, 7, 1, 91, '2023-03-01', 'non-tunai'),
(7, 7, 1, 92, '2023-03-11', 'tunai'),
(7, 7, 1, 93, '2023-03-16', 'tunai'),
(8, 8, 1, 94, '2023-03-01', 'non-tunai'),
(8, 8, 1, 95, '2023-03-12', 'non-tunai'),
(8, 8, 1, 96, '2023-03-17', 'tunai'),
(9, 9, 2, 97, '2023-03-01', 'non-tunai'),
(9, 9, 2, 98, '2023-03-13', 'non-tunai'),
(9, 9, 2, 99, '2023-03-18', 'tunai'),
(10, 10, 1, 100, '2023-03-01', 'tunai'),
(10, 10, 1, 101, '2023-03-14', 'non-tunai'),
(10, 10, 1, 102, '2023-03-19', 'tunai'),
(11, 11, 2, 103, '2023-03-01', 'non-tunai'),
(11, 11, 2, 104, '2023-03-15', 'tunai'),
(11, 11, 2, 105, '2023-03-20', 'non-tunai'),
(12, 12, 2, 106, '2023-03-01', 'tunai'),
(12, 12, 2, 107, '2023-03-16', 'non-tunai'),
(12, 12, 2, 108, '2023-03-21', 'tunai'),
(1, 1, 1, 109, '2023-03-25', 'tunai'),
(4, 4, 1, 110, '2023-03-26', 'tunai');
```

2.4.3. Tampilan Tabel Designer

Setelah merancang struktur tabel pada basis data, langkah selanjutnya adalah membuat tabel fisik menggunakan perintah SQL di phpMyAdmin.

phpMyAdmin merupakan aplikasi open source yang digunakan untuk mengelola basis data MySQL melalui antarmuka web. Di phpMyAdmin, pembuatan dan pengelolaan basis data menjadi lebih mudah karena dilakukan secara visual. Salah satu fitur phpMyAdmin yang sangat berguna adalah database designer. Fitur ini memungkinkan pembuatan tabel basis data secara visual dengan men-drag dan mengatur kolom serta relasi antar tabel.

Berikut ini adalah tampilan tabel designer untuk basis data Kost Poetra Sultan di phpMyAdmin:



Gambar 12 Tampilan Database Designer pada kasus Kost Poetra Sultan

Pada tampilan designer di atas, terlihat tabel-tabel yang dibutuhkan beserta kolom dan tipe datanya masing-masing. Relasi antar tabel juga disimbolkan dengan garis dan kardinalitas one atau many.

Proses pembuatan tabel diawali dengan menuliskan nama tabel di bagian tengah, lalu menambahkan kolom yang dibutuhkan ke tabel tersebut. Penentuan nama, tipe data, dan kunci kolom dilakukan pada form edit kolom.

Setelah semua tabel dan kolom selesai dibuat, relasi antar tabel didefinisikan dengan menghubungkan primary key dan foreign key antar tabel. Kardinalitas juga diatur agar sesuai dengan definisi relasi pada diagram ER.

Keuntungan menggunakan fitur designer adalah proses pembuatan tabel menjadi lebih cepat dan mudah dilakukan secara visual. Selain itu, desain basis data juga tersimpan sehingga memudahkan jika suatu saat diperlukan restrukturisasi tabel.

Fitur phpMyAdmin designer sangat direkomendasikan untuk membangun basis data, terutama basis data yang kompleks dengan banyak tabel dan relasi. Dengan begitu, resiko kesalahan pembuatan tabel secara manual dengan query SQL bisa diminimalisir.

2.4.4. Query

Untuk menampilkan nama lengkap penghuni, nomor kamar, jenis kamar, dan harga kamar,

```
SELECT p.namaDepan, p.namaBelakang, k.noKamar, tk.tipeKamar, tk.hargaKamar
FROM penghuni AS p
JOIN kamar AS k ON p.idPenghuni = k.idPenghuni
JOIN tipeKamar AS tk ON k.idTipeKamar = tk.idTipeKamar;
```

output

MariaDB [kostpoetrasultan]> SELECT p.namaDepan, p.namaBelakang, k.noKamar, tk.tipeKamar, tk.hargaKamar
-> FROM penghuni AS p
-> JOIN kamar AS k ON p.idPenghuni = k.idPenghuni
-> JOIN tipeKamar AS tk ON k.idTipeKamar = tk.idTipeKamar;

namaDepan	namaBelakang	noKamar	tipeKamar	hargaKamar
Budi	Santoso	1	Standard	800000.00
Ahmad	Wibowo	2	Standard	800000.00
Rudi	Susilo	3	Standard	800000.00
Dwi	Sulistyo	4	Standard	800000.00
Hendra	Kusuma	6	Standard	800000.00
Indra	Setiawan	7	Standard	800000.00
Ade	Wijaya	8	Standard	800000.00
Dicky	Saputra	10	Standard	800000.00
Eko	Prasetyo	5	Deluxe	1200000.00
Bayu	Nugroho	9	Deluxe	1200000.00
Fandi	Surya	11	Deluxe	1200000.00
Galih	Purnama	12	Deluxe	1200000.00

12 rows in set (0.004 sec)

Menghitung umur masing-masing penghuni

```
SELECT idPenghuni, CONCAT(namaDepan, ' ', namaBelakang) as 'Nama Lengkap',
TIMESTAMPDIFF(YEAR, tanggalLahir, CURDATE()) AS Umur
FROM penghuni
ORDER BY Umur DESC;
```

Output

```
MariaDB [kostpoetrasultan]> SELECT idPenghuni, CONCAT(namaDepan, ' ', namaBelakang) as 'Nama Lengkap',
```

```

-> TIMESTAMPDIFF(YEAR, tanggalLahir, CURDATE()) AS Umur
-> FROM penghuni
-> ORDER BY Umur DESC;

```

idPenghuni	Nama Lengkap	Umur
4	Dwi Sulisty	31
9	Bayu Nugroho	31
2	Ahmad Wibowo	30
7	Indra Setiawan	30
12	Galih Purnama	30
3	Rudi Susilo	29
6	Hendra Kusuma	29
10	Dicky Saputra	29
1	Budi Santoso	28
8	Ade Wijaya	28
5	Eko Prasetyo	27
11	Fandi Surya	27

12 rows in set (0.000 sec)

Menampilkan jumlah penghuni dan rata-rata umur dari keseluruhan penghuni

```

SELECT
    COUNT(*) as 'Jumlah Penghuni',
    AVG(TIMESTAMPDIFF(YEAR, tanggalLahir, CURDATE())) AS 'Rata-
rata Umur'
FROM
    penghuni;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
-> COUNT(*) as 'Jumlah Penghuni',
-> AVG(TIMESTAMPDIFF(YEAR, tanggalLahir, CURDATE())) AS 'Rata-rata Umur'
-> FROM
-> penghuni;

```

Jumlah Penghuni	Rata-rata Umur
12	29.0833

1 row in set (0.001 sec)

Melihat email dan nomor telepon masing masing penghuni

```

SELECT
    CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
    E.email,
    T.noTelp
FROM
    penghuni P
INNER JOIN
    penghuni_email E ON P.idPenghuni = E.idPenghuni
INNER JOIN
    penghuni_noTelp T ON P.idPenghuni = T.idPenghuni;

```

Output:


```

MariaDB [kostpoetrasultan]> SELECT
->     CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
->     E.email,
->     T.noTelp
-> FROM
->     penghuni P
-> INNER JOIN
->     penghuni_email E ON P.idPenghuni = E.idPenghuni
-> INNER JOIN
->     penghuni_noTelp T ON P.idPenghuni = T.idPenghuni;

```

Nama Lengkap	email	noTelp
Budi Santoso	budi.santoso@gmail.com	081234567890
Budi Santoso	budi.santoso@yahoo.com	081234567890
Budi Santoso	budi.santoso@outlook.com	081234567890
Ahmad Wibowo	ahmad.wibowo@gmail.com	081234567890
Ahmad Wibowo	ahmad.wibowo@yahoo.com	081234567890
Ahmad Wibowo	ahmad.wibowo@gmail.com	081112233445
Ahmad Wibowo	ahmad.wibowo@yahoo.com	081112233445
Rudi Susilo	rudi.susilo@gmail.com	081234567890
Rudi Susilo	rudi.susilo@yahoo.com	081234567890
Rudi Susilo	rudi.susilo@outlook.com	081234567890
Dwi Sulisty	dwi.sulisty@gmail.com	081112233445
Dwi Sulisty	dwi.sulisty@yahoo.com	081112233445
Eko Prasetyo	eko.prasetyo@gmail.com	081234567890
Eko Prasetyo	eko.prasetyo@yahoo.com	081234567890
Eko Prasetyo	eko.prasetyo@outlook.com	081234567890
Eko Prasetyo	eko.prasetyo@gmail.com	081112233445
Eko Prasetyo	eko.prasetyo@yahoo.com	081112233445
Eko Prasetyo	eko.prasetyo@outlook.com	081112233445
Hendra Kusuma	hendra.kusuma@gmail.com	081234567890
Hendra Kusuma	hendra.kusuma@yahoo.com	081234567890
Indra Setiawan	indra.setiawan@gmail.com	081112233445
Ade Wijaya	ade.wijaya@gmail.com	081234567890
Bayu Nugroho	bayu.nugroho@gmail.com	081112233445
Bayu Nugroho	bayu.nugroho@yahoo.com	081112233445
Bayu Nugroho	bayu.nugroho@outlook.com	081112233445
Dicky Saputra	dicky.saputra@gmail.com	081234567890
Dicky Saputra	dicky.saputra@gmail.com	081112233445
Fandi Surya	fandi.surya@gmail.com	081234567890
Fandi Surya	fandi.surya@gmail.com	081112233445
Galih Purnama	galih.purnama@gmail.com	081234567890
Galih Purnama	galih.purnama@gmail.com	081112233445

```

31 rows in set (0.000 sec)

```

Melihat lama penghuni menyewa (hari)

```

SELECT
    CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
    DATEDIFF(M.tglAkhirSewa, M.tglMulaiSewa) as 'Lama Sewa (hari)'
FROM
    penghuni P
INNER JOIN
    menyewa M ON P.idPenghuni = M.idPenghuni;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
->     CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
->     DATEDIFF(M.tglAkhirSewa, M.tglMulaiSewa) as 'Lama Sewa (hari)'
-> FROM
->     penghuni P
-> INNER JOIN
->     menyewa M ON P.idPenghuni = M.idPenghuni;

```

Nama Lengkap	Lama Sewa (hari)
Budi Santoso	365
Ahmad Wibowo	365
Rudi Susilo	365
Dwi Sulisty	365
Eko Prasetyo	365
Hendra Kusuma	365
Indra Setiawan	365
Ade Wijaya	365
Bayu Nugroho	365
Dicky Saputra	365
Fandi Surya	365
Galih Purnama	365

12 rows in set (0.004 sec)

Melihat lama penghuni menyewa (bulan)

```

SELECT
    CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
    ROUND(DATEDIFF(M.tglAkhirSewa, M.tglMulaiSewa) / 30) as 'Lama Sewa (bulan)'
FROM
    penghuni P
INNER JOIN
    menyewa M ON P.idPenghuni = M.idPenghuni;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
->     CONCAT(P.namaDepan, ' ', P.namaBelakang) as 'Nama Lengkap',
->     ROUND(DATEDIFF(M.tglAkhirSewa, M.tglMulaiSewa) / 30) as 'Lama Sewa (bulan)'
-> FROM
->     penghuni P
-> INNER JOIN
->     menyewa M ON P.idPenghuni = M.idPenghuni;

```

Nama Lengkap	Lama Sewa (bulan)
Budi Santoso	12
Ahmad Wibowo	12
Rudi Susilo	12
Dwi Sulisty	12
Eko Prasetyo	12
Hendra Kusuma	12
Indra Setiawan	12
Ade Wijaya	12
Bayu Nugroho	12

Dicky Saputra	12
Fandi Surya	12
Galih Purnama	12
+-----+	
12 rows in set (0.003 sec)	

Melihat jumlah masing-masing transaksi berdasarkan metode pembayaran

```
SELECT
    metodePembayaran,
    COUNT(*) as 'Jumlah Transaksi'
FROM
    membayar
GROUP BY
    metodePembayaran;
```

Output

```
MariaDB [kostpoetrasultan]> SELECT
->     metodePembayaran,
->     COUNT(*) as 'Jumlah Transaksi'
-> FROM
->     membayar
-> GROUP BY
->     metodePembayaran;
```

metodePembayaran	Jumlah Transaksi
tunai	56
non-tunai	54

2 rows in set (0.001 sec)

Perintah untuk mengetahui penghuni yang memiliki total biaya transaksi terbesar untuk setiap jenis transaksi.

```
SELECT
    jenisTransaksi,
    (SELECT CONCAT(namaDepan, ' ', namaBelakang)
     FROM penghuni
     WHERE idPenghuni = T.idPenghuni) as 'Nama Penghuni',
    MAX(jumlahBiaya) as 'Total Biaya'
FROM
    transaksi T
GROUP BY
    jenisTransaksi;
```

Output

```
MariaDB [kostpoetrasultan]> SELECT
->     jenisTransaksi,
->     (SELECT CONCAT(namaDepan, ' ', namaBelakang)
->     FROM penghuni
->     WHERE idPenghuni = T.idPenghuni) as 'Nama Penghuni',
->     MAX(jumlahBiaya) as 'Total Biaya'
-> FROM
->     transaksi T
-> GROUP BY
->     jenisTransaksi;
```

jenisTransaksi	Nama Penghuni	Total Biaya
sewa kamar	Budi Santoso	1200000.00
air	Budi Santoso	180000.00
listrik	Budi Santoso	250000.00
denda	Budi Santoso	60000.00

Untuk menampilkan nama lengkap, tipe kamar, dan total akumulasi biaya keseluruhannya untuk setiap penghuni

```
SELECT
    CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
    tk.tipeKamar,
    SUM(t.jumlahBiaya) as Total_Biaya
FROM
    penghuni p
JOIN
    kamar k ON p.idPenghuni = k.idPenghuni
JOIN
    tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN
    transaksi t ON p.idPenghuni = t.idPenghuni
GROUP BY
    p.idPenghuni, tk.tipeKamar
ORDER BY
    Total_Biaya DESC;
```

Output

```
MariaDB [kostpoetrasultan]> SELECT
->     CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
->     tk.tipeKamar,
->     SUM(t.jumlahBiaya) as Total_Biaya
-> FROM
```

```

-> penghuni p
-> JOIN
-> kamar k ON p.idPenghuni = k.idPenghuni
-> JOIN
-> tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
-> JOIN
-> transaksi t ON p.idPenghuni = t.idPenghuni
-> GROUP BY
-> p.idPenghuni, tk.tipeKamar
-> ORDER BY
-> Total_Biaya DESC;

```

Nama_Lengkap	tipeKamar	Total_Biaya
Ahmad Wibowo	Tanpa Kamar Mandi	4890000.00
Dicky Saputra	Tanpa Kamar Mandi	4890000.00
Hendra Kusuma	Tanpa Kamar Mandi	4830000.00
Dwi Sulisty	Tanpa Kamar Mandi	4800000.00
Galih Purnama	Dengan Kamar Mandi	4740000.00
Ade Wijaya	Tanpa Kamar Mandi	4680000.00
Budi Santoso	Tanpa Kamar Mandi	3500000.00
Bayu Nugroho	Dengan Kamar Mandi	3420000.00
Eko Prasetyo	Dengan Kamar Mandi	3390000.00
Fandi Surya	Dengan Kamar Mandi	3330000.00
Rudi Susilo	Tanpa Kamar Mandi	3330000.00
Indra Setiawan	Tanpa Kamar Mandi	3270000.00

12 rows in set (0.001 sec)

Seperti sebelumnya, tanpa diurutkan secara desc.

```

SELECT
  CONCAT(p.namaDepan, ' ', p.namaBelakang) AS 'Nama Lengkap',
  tk.tipeKamar,
  SUM(t.jumlahBiaya) as 'Total Biaya'
FROM
  penghuni p
JOIN
  kamar k ON p.idPenghuni = k.idPenghuni
JOIN
  tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN
  transaksi t ON p.idPenghuni = t.idPenghuni
GROUP BY
  p.idPenghuni, tk.tipeKamar;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
->  CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
->  tk.tipeKamar,
->  SUM(t.jumlahBiaya) as Total_Biaya
-> FROM
->  penghuni p

```



```

-> JOIN
-> kamar k ON p.idPenghuni = k.idPenghuni
-> JOIN
-> tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
-> JOIN
-> transaksi t ON p.idPenghuni = t.idPenghuni
-> GROUP BY
-> p.idPenghuni, tk.tipeKamar
-> ORDER BY
-> Total_Biaya DESC;

```

Nama_Lengkap	tipeKamar	Total_Biaya
Dicky Saputra	Tanpa Kamar Mandi	4890000.00
Ahmad Wibowo	Tanpa Kamar Mandi	4890000.00
Hendra Kusuma	Tanpa Kamar Mandi	4830000.00
Dwi Sulistyo	Tanpa Kamar Mandi	4800000.00
Galih Purnama	Dengan Kamar Mandi	4740000.00
Ade Wijaya	Tanpa Kamar Mandi	4680000.00
Budi Santoso	Tanpa Kamar Mandi	3500000.00
Bayu Nugroho	Dengan Kamar Mandi	3420000.00
Eko Prasetyo	Dengan Kamar Mandi	3390000.00
Fandi Surya	Dengan Kamar Mandi	3330000.00
Rudi Susilo	Tanpa Kamar Mandi	3330000.00
Indra Setiawan	Tanpa Kamar Mandi	3270000.00

12 rows in set (0.001 sec)

ganti nama tipeKamarnya menjadi Tanpa Kamar Mandi menjadi Standard, dan Dengan Kamar Mandi menjadi Deluxe

```

UPDATE tipeKamar
SET tipeKamar = CASE
    WHEN tipeKamar = 'Tanpa Kamar Mandi' THEN 'Standard'
    WHEN tipeKamar = 'Dengan Kamar Mandi' THEN 'Deluxe'
    ELSE tipeKamar
END;

```

Sebelum diganti

```

MariaDB [kostpoetrasultan]> select * from tipeKamar;
+-----+-----+-----+-----+
| idTipeKamar | tipeKamar | hargaKamar | luasKamar |
+-----+-----+-----+-----+
| 1 | Tanpa Kamar Mandi | 800000.00 | 12.50 |
| 2 | Dengan Kamar Mandi | 1200000.00 | 15.00 |
+-----+-----+-----+-----+
2 rows in set (0.001 sec)

```

Setelah diganti

```

MariaDB [kostpoetrasultan]> UPDATE tipeKamar
  -> SET tipeKamar = CASE
  ->     WHEN tipeKamar = 'Tanpa Kamar Mandi' THEN 'Standard'
  ->     WHEN tipeKamar = 'Dengan Kamar Mandi' THEN 'Deluxe'
  ->     ELSE tipeKamar
  -> END;
Query OK, 2 rows affected (0.006 sec)
Rows matched: 2  Changed: 2  Warnings: 0

MariaDB [kostpoetrasultan]> select * from tipeKamar;
+-----+-----+-----+-----+
| idTipeKamar | tipeKamar | hargaKamar | luasKamar |
+-----+-----+-----+-----+
|          1 | Standard | 800000.00 |    12.50 |
|          2 | Deluxe   | 1200000.00 |    15.00 |
+-----+-----+-----+-----+
2 rows in set (0.001 sec)

```

Untuk menampilkan akumulasi pembayaran tiap bulan, mengelompokkan data berdasarkan bulan dan tahun dari tglTransaksi

```

SELECT
  DATE_FORMAT(m.tglTransaksi, '%Y-%M') AS Bulan,
  SUM(t.jumlahBiaya) as Total_Biaya
FROM
  membayar m
JOIN
  transaksi t ON m.idTransaksi = t.idTransaksi
GROUP BY
  YEAR(m.tglTransaksi), MONTH(m.tglTransaksi)
ORDER BY
  Bulan ASC;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
  -> DATE_FORMAT(m.tglTransaksi, '%Y-%M') AS Bulan,
  -> SUM(t.jumlahBiaya) as Total_Biaya
  -> FROM
  -> membayar m
  -> JOIN
  -> transaksi t ON m.idTransaksi = t.idTransaksi
  -> GROUP BY
  -> YEAR(m.tglTransaksi), MONTH(m.tglTransaksi)
  -> ORDER BY
  -> m.tglTransaksi ASC;
+-----+-----+
| Bulan          | Total_Biaya |
+-----+-----+

```

```

+-----+-----+
| 2023-January | 16320000.00 |
| 2023-February | 16320000.00 |
| 2023-March    | 16430000.00 |
+-----+-----+
3 rows in set (0.001 sec)

```

Untuk mengambil data dari beberapa tabel yang terhubung dan menghasilkan informasi tentang transaksi yang dilakukan oleh penghuni

```

SELECT
  CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
  tk.tipeKamar,
  t.jenisTransaksi,
  DATE_FORMAT(m.tglTransaksi, '%Y-%M-%D') AS Bulan
FROM
  membayar m
JOIN
  penghuni p ON m.idPenghuni = p.idPenghuni
JOIN
  kamar k ON m.noKamar = k.noKamar
JOIN
  tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN
  transaksi t ON m.idTransaksi = t.idTransaksi
WHERE
  DAY(m.tglTransaksi) <= 10
ORDER BY
  m.tglTransaksi ASC;

```

Output

```

MariaDB [kostpoetrasultan]> SELECT
  ->  CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
  ->  tk.tipeKamar,
  ->  t.jenisTransaksi,
  ->  DATE_FORMAT(m.tglTransaksi, '%Y-%M-%D') AS Bulan
  -> FROM
  ->  membayar m
  -> JOIN
  ->  penghuni p ON m.idPenghuni = p.idPenghuni
  -> JOIN
  ->  kamar k ON m.noKamar = k.noKamar
  -> JOIN
  ->  tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
  -> JOIN
  ->  transaksi t ON m.idTransaksi = t.idTransaksi
  -> WHERE
  ->  DAY(m.tglTransaksi) <= 10
  -> ORDER BY
  ->  m.tglTransaksi ASC;

```

```

+-----+-----+-----+-----+
| Nama_Lengkap | tipeKamar | jenisTransaksi | Bulan |
+-----+-----+-----+-----+
| Dicky Saputra | Tanpa Kamar Mandi | sewa kamar | 2023-January-1st |
| Fandi Surya   | Dengan Kamar Mandi | sewa kamar | 2023-January-1st |
+-----+-----+-----+-----+

```

Ahmad Wibowo	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Ade Wijaya	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Rudi Susilo	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Eko Prasetyo	Dengan Kamar Mandi	sewa kamar	2023-January-1st
Dwi Sulisty	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Budi Santoso	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Bayu Nugroho	Dengan Kamar Mandi	sewa kamar	2023-January-1st
Hendra Kusuma	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Galih Purnama	Dengan Kamar Mandi	sewa kamar	2023-January-1st
Indra Setiawan	Tanpa Kamar Mandi	sewa kamar	2023-January-1st
Budi Santoso	Tanpa Kamar Mandi	listrik	2023-January-5th
Rudi Susilo	Tanpa Kamar Mandi	listrik	2023-January-6th
Ahmad Wibowo	Tanpa Kamar Mandi	listrik	2023-January-7th
Dwi Sulisty	Tanpa Kamar Mandi	listrik	2023-January-8th
Eko Prasetyo	Dengan Kamar Mandi	listrik	2023-January-9th
Hendra Kusuma	Tanpa Kamar Mandi	listrik	2023-January-10th
Budi Santoso	Tanpa Kamar Mandi	air	2023-January-10th
Eko Prasetyo	Dengan Kamar Mandi	sewa kamar	2023-February-1st
Galih Purnama	Dengan Kamar Mandi	sewa kamar	2023-February-1st
Dwi Sulisty	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Indra Setiawan	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Hendra Kusuma	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Dicky Saputra	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Budi Santoso	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Fandi Surya	Dengan Kamar Mandi	sewa kamar	2023-February-1st
Ade Wijaya	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Ahmad Wibowo	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Bayu Nugroho	Dengan Kamar Mandi	sewa kamar	2023-February-1st
Rudi Susilo	Tanpa Kamar Mandi	sewa kamar	2023-February-1st
Budi Santoso	Tanpa Kamar Mandi	listrik	2023-February-5th
Rudi Susilo	Tanpa Kamar Mandi	listrik	2023-February-6th
Ahmad Wibowo	Tanpa Kamar Mandi	listrik	2023-February-7th
Dwi Sulisty	Tanpa Kamar Mandi	listrik	2023-February-8th
Eko Prasetyo	Dengan Kamar Mandi	listrik	2023-February-9th
Hendra Kusuma	Tanpa Kamar Mandi	listrik	2023-February-10th
Budi Santoso	Tanpa Kamar Mandi	air	2023-February-10th
Bayu Nugroho	Dengan Kamar Mandi	sewa kamar	2023-March-1st
Ahmad Wibowo	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Galih Purnama	Dengan Kamar Mandi	sewa kamar	2023-March-1st
Indra Setiawan	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Rudi Susilo	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Eko Prasetyo	Dengan Kamar Mandi	sewa kamar	2023-March-1st
Dicky Saputra	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Dwi Sulisty	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Hendra Kusuma	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Fandi Surya	Dengan Kamar Mandi	sewa kamar	2023-March-1st
Ade Wijaya	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Budi Santoso	Tanpa Kamar Mandi	sewa kamar	2023-March-1st
Budi Santoso	Tanpa Kamar Mandi	listrik	2023-March-5th
Rudi Susilo	Tanpa Kamar Mandi	listrik	2023-March-6th
Ahmad Wibowo	Tanpa Kamar Mandi	listrik	2023-March-7th
Dwi Sulisty	Tanpa Kamar Mandi	listrik	2023-March-8th
Eko Prasetyo	Dengan Kamar Mandi	listrik	2023-March-9th
Budi Santoso	Tanpa Kamar Mandi	air	2023-March-10th
Hendra Kusuma	Tanpa Kamar Mandi	listrik	2023-March-10th

57 rows in set (0.001 sec)

Perintah SQL dibawah ini mengambil data nama lengkap, umur, nomor kamar, tipe kamar, dan lama menyewa dari tabel 'penghuni', 'kamar', 'tipeKamar', dan 'menyewa'. Selanjutnya,

menghitung umur dan lama menyewa menggunakan fungsi TIMESTAMPDIFF. Fungsi WHERE digunakan untuk memfilter data penghuni yang umurnya di atas rata-rata umur semua penghuni. Terakhir, data diurutkan berdasarkan umur dari yang tertua.

```
SELECT
    CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
    TIMESTAMPDIFF(YEAR, p.tanggalLahir, '2023-01-01') AS Umur,
    k.noKamar,
    tk.tipeKamar,
    TIMESTAMPDIFF(MONTH, m.tglMulaiSewa, m.tglAkhirSewa) AS
    Lama_Menyewa
FROM
    penghuni p
JOIN
    kamar k ON p.idPenghuni = k.idPenghuni
JOIN
    tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN
    menyewa m ON p.idPenghuni = m.idPenghuni
WHERE
    TIMESTAMPDIFF(YEAR, p.tanggalLahir, '2023-01-01') >
    (SELECT AVG(TIMESTAMPDIFF(YEAR, tanggalLahir, '2023-01-01'))
    FROM penghuni)
ORDER BY
    Umur DESC;
```

Output

```
MariaDB [kostpoetrasultan]> SELECT
-> CONCAT(p.namaDepan, ' ', p.namaBelakang) AS Nama_Lengkap,
-> TIMESTAMPDIFF(YEAR, p.tanggalLahir, '2023-01-01') AS Umur,
-> k.noKamar,
-> tk.tipeKamar,
-> TIMESTAMPDIFF(MONTH, m.tglMulaiSewa, m.tglAkhirSewa) AS Lama_Menyewa
-> FROM
-> penghuni p
-> JOIN
-> kamar k ON p.idPenghuni = k.idPenghuni
-> JOIN
-> tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
-> JOIN
-> menyewa m ON p.idPenghuni = m.idPenghuni
-> WHERE
-> TIMESTAMPDIFF(YEAR, p.tanggalLahir, '2023-01-01') >
-> (SELECT AVG(TIMESTAMPDIFF(YEAR, tanggalLahir, '2023-01-01')) FROM penghuni)
-> ORDER BY
-> Umur DESC;
```

Nama_Lengkap	Umur	noKamar	tipeKamar	Lama_Menyewa
Dwi Sulistyio	30	4	Tanpa Kamar Mandi	12
Bayu Nugroho	30	9	Dengan Kamar Mandi	12
Ahmad Wibowo	29	2	Tanpa Kamar Mandi	12
Indra Setiawan	29	7	Tanpa Kamar Mandi	12

Galih Purnama	29	12	Dengan Kamar Mandi	12
+-----+-----+-----+-----+-----+				
5 rows in set (0.001 sec)				

Perintah SQL di bawah ini mengambil data unik nama lengkap penghuni, nomor kamar, nomor dan jenis kamar, serta jenis transaksi dari tabel 'penghuni', 'kamar', 'tipeKamar', 'transaksi', dan 'membayar'. Fungsi WHERE digunakan untuk memfilter data transaksi yang jarak waktunya dari tanggal transaksi hingga saat ini lebih dari rata-rata jarak waktu semua transaksi.

```
SELECT DISTINCT
  CONCAT(penghuni.namaDepan, ' ', penghuni.namaBelakang) AS 'Nama Lengkap',
  kamar.noKamar,
  CONCAT(kamar.noKamar, ' (', tipeKamar.tipeKamar, ')') AS 'No Kamar dan Jenis Kamar',
  transaksi.jenisTransaksi
FROM
  penghuni
  JOIN kamar ON penghuni.idPenghuni = kamar.idPenghuni
  JOIN tipeKamar ON kamar.idTipeKamar = tipeKamar.idTipeKamar
  JOIN transaksi ON penghuni.idPenghuni = transaksi.idPenghuni
  JOIN membayar ON penghuni.idPenghuni = membayar.idPenghuni
WHERE
  DATEDIFF(CURDATE(), membayar.tglTransaksi) > (SELECT AVG(DATEDIFF(CURDATE(),
  tglTransaksi)) FROM membayar);
```

Output:

MariaDB [kostpoetrasultan]> SELECT DISTINCT

```
-> CONCAT(penghuni.namaDepan, ' ', penghuni.namaBelakang) AS 'Nama Lengkap',
-> kamar.noKamar,
-> CONCAT(kamar.noKamar, ' (', tipeKamar.tipeKamar, ')') AS 'No Kamar dan Jenis Kamar',
-> transaksi.jenisTransaksi
-> FROM
-> penghuni
-> JOIN kamar ON penghuni.idPenghuni = kamar.idPenghuni
-> JOIN tipeKamar ON kamar.idTipeKamar = tipeKamar.idTipeKamar
-> JOIN transaksi ON penghuni.idPenghuni = transaksi.idPenghuni
-> JOIN membayar ON penghuni.idPenghuni = membayar.idPenghuni
-> WHERE
-> DATEDIFF(CURDATE(), membayar.tglTransaksi) > (SELECT AVG(DATEDIFF(CURDATE(), tglTransaksi)) FROM membayar);
```

Nama Lengkap	noKamar	No Kamar dan Jenis Kamar	jenisTransaksi
Budi Santoso	1	1 (Tanpa Kamar Mandi)	sewa kamar
Budi Santoso	1	1 (Tanpa Kamar Mandi)	listrik
Budi Santoso	1	1 (Tanpa Kamar Mandi)	air
Budi Santoso	1	1 (Tanpa Kamar Mandi)	denda
Ahmad Wibowo	2	2 (Tanpa Kamar Mandi)	sewa kamar
Ahmad Wibowo	2	2 (Tanpa Kamar Mandi)	listrik
Ahmad Wibowo	2	2 (Tanpa Kamar Mandi)	air
Rudi Susilo	3	3 (Tanpa Kamar Mandi)	sewa kamar
Rudi Susilo	3	3 (Tanpa Kamar Mandi)	listrik
Rudi Susilo	3	3 (Tanpa Kamar Mandi)	air
Dwi Sulistyio	4	4 (Tanpa Kamar Mandi)	sewa kamar
Dwi Sulistyio	4	4 (Tanpa Kamar Mandi)	listrik
Dwi Sulistyio	4	4 (Tanpa Kamar Mandi)	air
Dwi Sulistyio	4	4 (Tanpa Kamar Mandi)	denda
Hendra Kusuma	6	6 (Tanpa Kamar Mandi)	sewa kamar
Hendra Kusuma	6	6 (Tanpa Kamar Mandi)	listrik
Hendra Kusuma	6	6 (Tanpa Kamar Mandi)	air
Indra Setiawan	7	7 (Tanpa Kamar Mandi)	sewa kamar
Indra Setiawan	7	7 (Tanpa Kamar Mandi)	listrik
Indra Setiawan	7	7 (Tanpa Kamar Mandi)	air
Ade Wijaya	8	8 (Tanpa Kamar Mandi)	sewa kamar
Ade Wijaya	8	8 (Tanpa Kamar Mandi)	listrik
Ade Wijaya	8	8 (Tanpa Kamar Mandi)	air
Dicky Saputra	10	10 (Tanpa Kamar Mandi)	sewa kamar
Dicky Saputra	10	10 (Tanpa Kamar Mandi)	listrik
Dicky Saputra	10	10 (Tanpa Kamar Mandi)	air
Eko Prasetyo	5	5 (Dengan Kamar Mandi)	sewa kamar
Eko Prasetyo	5	5 (Dengan Kamar Mandi)	listrik
Eko Prasetyo	5	5 (Dengan Kamar Mandi)	air
Bayu Nugroho	9	9 (Dengan Kamar Mandi)	sewa kamar

Bayu Nugroho	9	9 (Dengan Kamar Mandi)	listrik
Bayu Nugroho	9	9 (Dengan Kamar Mandi)	air
Fandi Surya	11	11 (Dengan Kamar Mandi)	sewa kamar
Fandi Surya	11	11 (Dengan Kamar Mandi)	listrik
Fandi Surya	11	11 (Dengan Kamar Mandi)	air
Galih Purnama	12	12 (Dengan Kamar Mandi)	sewa kamar
Galih Purnama	12	12 (Dengan Kamar Mandi)	listrik
Galih Purnama	12	12 (Dengan Kamar Mandi)	air

38 rows in set (0.001 sec)

Perintah Query di bawah ini menggabungkan dua query yang berbeda. Query pertama mengambil daftar penghuni yang menyewa kamar di tahun 2023 dan biaya sewa kamar mereka lebih tinggi dari rata-rata biaya sewa kamar. Query kedua mengambil daftar penghuni yang memiliki email Gmail dan biaya sewa kamar mereka lebih tinggi dari rata-rata biaya sewa kamar. Kedua query ini digabungkan menggunakan UNION.

```

SELECT CONCAT(p.namaDepan, ' ', p.namaBelakang) AS namaLengkap, k.noKamar, tk.tipeKamar, t.jenisTransaksi, t.jumlahBiaya
FROM penghuni p
JOIN kamar k ON p.idPenghuni = k.idPenghuni
JOIN tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN transaksi t ON p.idPenghuni = t.idPenghuni AND k.noKamar = t.noKamar AND tk.idTipeKamar = t.idTipeKamar
WHERE p.idPenghuni IN (
    SELECT idPenghuni
    FROM menyewa
    WHERE tglMulaiSewa > '2023-01-01'
    AND tglAkhirSewa < '2023-12-31'
)
AND t.jumlahBiaya > (
    SELECT AVG(jumlahBiaya)
    FROM transaksi
    WHERE jenisTransaksi = 'sewa kamar'
)
UNION
SELECT CONCAT(p.namaDepan, ' ', p.namaBelakang) AS namaLengkap, k.noKamar, tk.tipeKamar, m.metodePembayaran,
t.jumlahBiaya
FROM penghuni p
JOIN kamar k ON p.idPenghuni = k.idPenghuni
JOIN tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
JOIN transaksi t ON p.idPenghuni = t.idPenghuni AND k.noKamar = t.noKamar AND tk.idTipeKamar = t.idTipeKamar
JOIN membayar m ON t.idTransaksi = m.idTransaksi
WHERE p.idPenghuni IN (
    SELECT idPenghuni
    FROM penghuni_email
    WHERE email LIKE '%@gmail.com%'
)
AND t.jumlahBiaya > (
    SELECT AVG(jumlahBiaya)
    FROM transaksi
    WHERE jenisTransaksi = 'sewa kamar'
);

```

Output:

```

MariaDB [kostpoetrasultan]> SELECT p.namaDepan, p.namaBelakang, k.noKamar, tk.tipeKamar, t.jenisTransaksi, t.jumlahBiaya
-> FROM penghuni p
-> JOIN kamar k ON p.idPenghuni = k.idPenghuni
-> JOIN tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
-> JOIN transaksi t ON p.idPenghuni = t.idPenghuni AND k.noKamar = t.noKamar AND tk.idTipeKamar = t.idTipeKamar
-> WHERE p.idPenghuni IN (
-> SELECT idPenghuni
-> FROM menyewa
-> WHERE tglMulaiSewa > '2023-01-01'
-> AND tglAkhirSewa < '2023-12-31'
-> )
-> AND t.jumlahBiaya > (
-> SELECT AVG(jumlahBiaya)
-> FROM transaksi
-> WHERE jenisTransaksi = 'sewa kamar'
-> )
-> UNION
-> SELECT p.namaDepan, p.namaBelakang, k.noKamar, tk.tipeKamar, m.metodePembayaran, t.jumlahBiaya
-> FROM penghuni p

```

```

-> JOIN kamar k ON p.idPenghuni = k.idPenghuni
-> JOIN tipeKamar tk ON k.idTipeKamar = tk.idTipeKamar
-> JOIN transaksi t ON p.idPenghuni = t.idPenghuni AND k.noKamar = t.noKamar AND tk.idTipeKamar = t.idTipeKamar
-> JOIN membayar m ON t.idTransaksi = m.idTransaksi
-> WHERE p.idPenghuni IN (
->   SELECT idPenghuni
->   FROM penghuni_email
->   WHERE email LIKE '%@gmail.com%'
-> )
-> AND t.jumlahBiaya > (
->   SELECT AVG(jumlahBiaya)
->   FROM transaksi
->   WHERE jenisTransaksi = 'sewa kamar'
-> );

```

namaDepan	namaBelakang	noKamar	tipeKamar	jenisTransaksi	jumlahBiaya
Ahmad	Wibowo	2	Tanpa Kamar Mandi	tunai	1200000.00
Dwi	Sulistyo	4	Tanpa Kamar Mandi	tunai	1200000.00
Hendra	Kusuma	6	Tanpa Kamar Mandi	non-tunai	1200000.00
Ade	Wijaya	8	Tanpa Kamar Mandi	non-tunai	1200000.00
Dicky	Saputra	10	Tanpa Kamar Mandi	tunai	1200000.00
Galih	Purnama	12	Dengan Kamar Mandi	tunai	1200000.00

6 rows in set (0.043 sec)

Perintah SQL ini melakukan penghapusan semua data dari tabel 'membayar', 'transaksi', 'kamar', 'tipeKamar', 'menyewa', 'penghuni_email', 'penghuni_noTelp', dan 'penghuni'. Kemudian, melakukan reset pada nilai increment otomatis (AUTO_INCREMENT) pada masing-masing tabel tersebut menjadi 1. Ini berarti bahwa entri baru yang ditambahkan ke setiap tabel tersebut akan dimulai dengan ID 1.

```

DELETE FROM membayar;
DELETE FROM transaksi;
DELETE FROM kamar;
DELETE FROM tipeKamar;
DELETE FROM menyewa;
DELETE FROM penghuni_email;
DELETE FROM penghuni_noTelp;
DELETE FROM penghuni;
ALTER TABLE membayar AUTO_INCREMENT = 1;
ALTER TABLE transaksi AUTO_INCREMENT = 1;
ALTER TABLE kamar AUTO_INCREMENT = 1;
ALTER TABLE tipeKamar AUTO_INCREMENT = 1;
ALTER TABLE menyewa AUTO_INCREMENT = 1;
ALTER TABLE penghuni_email AUTO_INCREMENT = 1;
ALTER TABLE penghuni_noTelp AUTO_INCREMENT = 1;
ALTER TABLE penghuni AUTO_INCREMENT = 1;

```

Output

```

MariaDB [kostpoetrasultan]> DELETE FROM membayar;
Query OK, 108 rows affected (0.005 sec)

MariaDB [kostpoetrasultan]> DELETE FROM transaksi;
Query OK, 108 rows affected (0.002 sec)

MariaDB [kostpoetrasultan]> DELETE FROM kamar;
Query OK, 12 rows affected (0.001 sec)

```

```

MariaDB [kostpoetrasultan]> DELETE FROM tipeKamar;
Query OK, 2 rows affected (0.001 sec)

MariaDB [kostpoetrasultan]> DELETE FROM menyewa;
Query OK, 12 rows affected (0.001 sec)

MariaDB [kostpoetrasultan]> DELETE FROM penghuni_email;
Query OK, 23 rows affected (0.003 sec)

MariaDB [kostpoetrasultan]> DELETE FROM penghuni_noTelp;
Query OK, 17 rows affected (0.002 sec)

MariaDB [kostpoetrasultan]> DELETE FROM penghuni;
Query OK, 12 rows affected (0.002 sec)

MariaDB [kostpoetrasultan]> ALTER TABLE membayar AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE transaksi AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE kamar AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.004 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE tipeKamar AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.003 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE menyewa AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE penghuni_email AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE penghuni_noTelp AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [kostpoetrasultan]> ALTER TABLE penghuni AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.002 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

Saat kita lihat ternyata sudah kosong

```

-- Menampilkan data dari tabel Penghuni
SELECT * FROM penghuni;

-- Menampilkan data dari tabel Penghuni Email

```

```
SELECT * FROM penghuni_email;

-- Menampilkan data dari tabel Penghuni NoTelp
SELECT * FROM penghuni_noTelp;

-- Menampilkan data dari tabel Menyewa
SELECT * FROM menyewa;

-- Menampilkan data dari tabel Tipe Kamar
SELECT * FROM tipeKamar;

-- Menampilkan data dari tabel Kamar
SELECT * FROM kamar;

-- Menampilkan data dari tabel Transaksi
SELECT * FROM transaksi;

-- Menampilkan data dari tabel Membayar
SELECT * FROM membayar;
```

```
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel
Penghuni
MariaDB [kostpoetrasultan]> SELECT * FROM penghuni;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel
Penghuni Email
MariaDB [kostpoetrasultan]> SELECT * FROM penghuni_email;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel
Penghuni NoTelp
MariaDB [kostpoetrasultan]> SELECT * FROM penghuni_noTelp;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel Menyewa
MariaDB [kostpoetrasultan]> SELECT * FROM menyewa;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel Tipe
Kamar
```



```

MariaDB [kostpoetrasultan]> SELECT * FROM tipeKamar;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel Kamar
MariaDB [kostpoetrasultan]> SELECT * FROM kamar;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel
Transaksi
MariaDB [kostpoetrasultan]> SELECT * FROM transaksi;
Empty set (0.000 sec)

MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> -- Menampilkan data dari tabel
Membayar
MariaDB [kostpoetrasultan]> SELECT * FROM membayar;
Empty set (0.000 sec)

```

2.4.5. Trigger dan Procedure

1. Store Procedure

Perintah DESCRIBE dalam SQL digunakan untuk mendapatkan detail struktur tabel. Dalam hal ini, perintah DESCRIBE digunakan untuk mendapatkan detail struktur tabel 'kamar', 'membayar', 'menyewa', 'penghuni', 'penghuni_email', 'penghuni_notelp', 'tipekamar', dan 'transaksi'. Detail yang didapatkan meliputi nama kolom, tipe data, apakah kolom bisa bernilai NULL atau tidak, kunci (jika ada), nilai default (jika ada), dan ekstra (seperti AUTO_INCREMENT).

```

DESCRIBE kamar;
DESCRIBE membayar;
DESCRIBE menyewa;
DESCRIBE penghuni;
DESCRIBE penghuni_email;
DESCRIBE penghuni_notelp;
DESCRIBE tipekamar;
DESCRIBE transaksi;

```

Melihat seluruh isi atribut pada masing masing, untuk lebih mudah daripada menggunakan describe <nama_tabel> dapat menggunakan fungsi berikut

```

DELIMITER $$
CREATE PROCEDURE DescribeAllTables()
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE a CHAR(64);
    DECLARE cur CURSOR FOR SELECT table_name FROM
information_schema.tables WHERE table_schema = 'kostpoetrasultan';
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

    OPEN cur;

    read_loop: LOOP
        FETCH cur INTO a;
        IF done THEN
            LEAVE read_loop;
        END IF;
        SET @s = CONCAT('DESCRIBE ', a);
        PREPARE stmt FROM @s;
        EXECUTE stmt;
    END LOOP;

    CLOSE cur;
END;
$$
DELIMITER ;

```

Untuk memanggilnya hanya

```
CALL DescribeAllTables();
```

Output:

```

MariaDB [kostpoetrasultan]> CALL DescribeAllTables();
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| noKamar | int(11) | NO | PRI | NULL | auto_increment |
| idTipeKamar | int(11) | YES | MUL | NULL | |
| idPenghuni | int(11) | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.006 sec)

+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idPenghuni | int(11) | YES | MUL | NULL | |
| noKamar | int(11) | YES | MUL | NULL | |
| idTipeKamar | int(11) | YES | MUL | NULL | |
| idTransaksi | int(11) | NO | PRI | NULL | auto_increment |
| tglTransaksi | date | YES | | NULL | |
| metodePembayaran | enum('tunai','non-tunai') | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+

```

6 rows in set (0.010 sec)

Field	Type	Null	Key	Default	Extra
idPenghuni	int(11)	YES	MUL	NULL	
tglMulaiSewa	date	YES		NULL	
tglAkhirSewa	date	YES		NULL	

3 rows in set (0.022 sec)

Field	Type	Null	Key	Default	Extra
idPenghuni	int(11)	NO	PRI	NULL	auto_increment
namaDepan	varchar(255)	YES		NULL	
namaBelakang	varchar(255)	YES		NULL	
jalan	varchar(255)	YES		NULL	
RT	int(11)	YES		NULL	
RW	int(11)	YES		NULL	
kelurahan	varchar(255)	YES		NULL	
kecamatan	varchar(255)	YES		NULL	
kota	varchar(255)	YES		NULL	
provinsi	varchar(255)	YES		NULL	
kodePos	int(11)	YES		NULL	
tanggalLahir	date	YES		NULL	

12 rows in set (0.027 sec)

Field	Type	Null	Key	Default	Extra
idPenghuni	int(11)	YES	MUL	NULL	
email	varchar(255)	YES		NULL	

2 rows in set (0.046 sec)

Field	Type	Null	Key	Default	Extra
idPenghuni	int(11)	YES	MUL	NULL	
noTelp	varchar(255)	YES		NULL	

2 rows in set (0.049 sec)

Field	Type	Null	Key	Default	Extra
idTipeKamar	int(11)	NO	PRI	NULL	auto_increment
tipeKamar	varchar(255)	YES		NULL	
hargaKamar	decimal(10,2)	YES		NULL	
luasKamar	decimal(10,2)	YES		NULL	

4 rows in set (0.052 sec)

Field	Type	Null	Key	Default	Extra
idPenghuni	int(11)	YES	MUL	NULL	
noKamar	int(11)	YES	MUL	NULL	
idTipeKamar	int(11)	YES	MUL	NULL	
idTransaksi	int(11)	NO	PRI	NULL	auto_increment
jenisTransaksi	enum('sewa kamar','air','listrik','denda')	YES		NULL	
jumlahBiaya	decimal(10,2)	YES		NULL	
keterangan	varchar(255)	YES		NULL	

7 rows in set (0.059 sec)

Query OK, 0 rows affected (0.076 sec)

Melihat isi data semua data pada masing-masing tabel dengan

```
DELIMITER //
CREATE PROCEDURE ShowAllTables()
BEGIN
    DECLARE done INT DEFAULT 0;
    DECLARE tableName VARCHAR(255);
    DECLARE cur CURSOR FOR
        SELECT table_name
        FROM information_schema.tables
        WHERE table_schema = DATABASE();
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

    OPEN cur;
    read_loop: LOOP
        FETCH cur INTO tableName;
        IF done THEN
            LEAVE read_loop;
        END IF;
        SET @s = CONCAT('SELECT * FROM ', tableName);
        PREPARE stmt FROM @s;
        EXECUTE stmt;
        DEALLOCATE PREPARE stmt;
    END LOOP;
    CLOSE cur;
END //
DELIMITER ;
```

Untuk memanggilnya

```
CALL ShowAllTables();
```

Hasil output

```
MariaDB [kostpoetrasultan]> DELIMITER //
MariaDB [kostpoetrasultan]> CREATE PROCEDURE ShowAllTables()
-> BEGIN
->   DECLARE done INT DEFAULT 0;
->   DECLARE tableName VARCHAR(255);
->   DECLARE cur CURSOR FOR
->     SELECT table_name
->     FROM information_schema.tables
->     WHERE table_schema = DATABASE();
->   DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
->
->   OPEN cur;
->   read_loop: LOOP
->     FETCH cur INTO tableName;
->     IF done THEN
->       LEAVE read_loop;
->     END IF;
->     SET @s = CONCAT('SELECT * FROM ', tableName);
->     PREPARE stmt FROM @s;
->     EXECUTE stmt;
->     DEALLOCATE PREPARE stmt;
->   END LOOP;
->   CLOSE cur;
-> END //
Query OK, 0 rows affected (0.006 sec)
```

```
MariaDB [kostpoetrasultan]> DELIMITER ;
MariaDB [kostpoetrasultan]> CALL ShowAllTables();
```

nokamar	idTipeKamar	idPenghuni
1	1	1
2	1	2
3	1	3
4	1	4
5	2	5
6	1	6
7	1	7
8	1	8
9	2	9
10	1	10
11	2	11

12 2 12					
12 rows in set (0.004 sec)					
idPenghuni	noKamar	idTipeKamar	idTransaksi	tglTransaksi	metodePembayaran
1	1	1	1	2023-01-01	tunai
1	1	1	2	2023-01-05	non-tunai
1	1	1	3	2023-01-10	non-tunai
2	2	1	4	2023-01-01	tunai
2	2	1	5	2023-01-07	tunai
2	2	1	6	2023-01-12	non-tunai
3	3	1	7	2023-01-01	non-tunai
3	3	1	8	2023-01-06	tunai
3	3	1	9	2023-01-11	non-tunai
4	4	2	10	2023-01-01	tunai
4	4	2	11	2023-01-08	non-tunai
4	4	2	12	2023-01-13	tunai
5	5	1	13	2023-01-01	non-tunai
5	5	1	14	2023-01-09	tunai
5	5	1	15	2023-01-14	tunai
6	6	2	16	2023-01-01	non-tunai
6	6	2	17	2023-01-10	non-tunai
6	6	2	18	2023-01-15	tunai
7	7	1	19	2023-01-01	non-tunai
7	7	1	20	2023-01-11	tunai
7	7	1	21	2023-01-16	tunai
8	8	1	22	2023-01-01	non-tunai
8	8	1	23	2023-01-12	non-tunai
8	8	1	24	2023-01-17	tunai
9	9	2	25	2023-01-01	non-tunai
9	9	2	26	2023-01-13	non-tunai
9	9	2	27	2023-01-18	tunai
10	10	1	28	2023-01-01	tunai
10	10	1	29	2023-01-14	non-tunai
10	10	1	30	2023-01-19	tunai
11	11	2	31	2023-01-01	non-tunai
11	11	2	32	2023-01-15	tunai
11	11	2	33	2023-01-20	non-tunai
12	12	2	34	2023-01-01	tunai
12	12	2	35	2023-01-16	non-tunai
12	12	2	36	2023-01-21	tunai
1	1	1	37	2023-02-01	tunai
1	1	1	38	2023-02-05	non-tunai
1	1	1	39	2023-02-10	non-tunai
2	2	1	40	2023-02-01	tunai
2	2	1	41	2023-02-07	tunai
2	2	1	42	2023-02-12	non-tunai
3	3	1	43	2023-02-01	non-tunai
3	3	1	44	2023-02-06	tunai
3	3	1	45	2023-02-11	non-tunai
4	4	2	46	2023-02-01	tunai
4	4	2	47	2023-02-08	non-tunai
4	4	2	48	2023-02-13	tunai
5	5	1	49	2023-02-01	non-tunai
5	5	1	50	2023-02-09	tunai
5	5	1	51	2023-02-14	tunai
6	6	2	52	2023-02-01	non-tunai
6	6	2	53	2023-02-10	non-tunai
6	6	2	54	2023-02-15	tunai
7	7	1	55	2023-02-01	non-tunai
7	7	1	56	2023-02-11	tunai
7	7	1	57	2023-02-16	tunai
8	8	1	58	2023-02-01	non-tunai
8	8	1	59	2023-02-12	non-tunai
8	8	1	60	2023-02-17	tunai
9	9	2	61	2023-02-01	non-tunai
9	9	2	62	2023-02-13	non-tunai
9	9	2	63	2023-02-18	tunai
10	10	1	64	2023-02-01	tunai
10	10	1	65	2023-02-14	non-tunai
10	10	1	66	2023-02-19	tunai
11	11	2	67	2023-02-01	non-tunai
11	11	2	68	2023-02-15	tunai
11	11	2	69	2023-02-20	non-tunai
12	12	2	70	2023-02-01	tunai
12	12	2	71	2023-02-16	non-tunai
12	12	2	72	2023-02-21	tunai
1	1	1	73	2023-03-01	tunai
1	1	1	74	2023-03-05	non-tunai
1	1	1	75	2023-03-10	non-tunai
2	2	1	76	2023-03-01	tunai
2	2	1	77	2023-03-07	tunai
2	2	1	78	2023-03-12	non-tunai
3	3	1	79	2023-03-01	non-tunai
3	3	1	80	2023-03-06	tunai
3	3	1	81	2023-03-11	non-tunai
4	4	2	82	2023-03-01	tunai
4	4	2	83	2023-03-08	non-tunai
4	4	2	84	2023-03-13	tunai
5	5	1	85	2023-03-01	non-tunai
5	5	1	86	2023-03-09	tunai
5	5	1	87	2023-03-14	tunai
6	6	2	88	2023-03-01	non-tunai
6	6	2	89	2023-03-10	non-tunai
6	6	2	90	2023-03-15	tunai
7	7	1	91	2023-03-01	non-tunai
7	7	1	92	2023-03-11	tunai
7	7	1	93	2023-03-16	tunai
8	8	1	94	2023-03-01	non-tunai

8	8	1	95	2023-03-12	non-tunai
8	8	1	96	2023-03-17	tunai
9	9	2	97	2023-03-01	non-tunai
9	9	2	98	2023-03-13	non-tunai
9	9	2	99	2023-03-18	tunai
10	10	1	100	2023-03-01	tunai
10	10	1	101	2023-03-14	non-tunai
10	10	1	102	2023-03-19	tunai
11	11	2	103	2023-03-01	non-tunai
11	11	2	104	2023-03-15	tunai
11	11	2	105	2023-03-20	non-tunai
12	12	2	106	2023-03-01	tunai
12	12	2	107	2023-03-16	non-tunai
12	12	2	108	2023-03-21	tunai
1	1	1	109	2023-03-25	tunai
4	4	1	110	2023-03-26	tunai

110 rows in set (0.016 sec)

idPenghuni	tglMulaiSewa	tglAkhirSewa
1	2022-01-01	2023-01-01
2	2022-01-01	2023-01-01
3	2022-01-01	2023-01-01
4	2022-01-01	2023-01-01
5	2022-01-01	2023-01-01
6	2022-01-01	2023-01-01
7	2022-01-01	2023-01-01
8	2022-01-01	2023-01-01
9	2022-01-01	2023-01-01
10	2022-01-01	2023-01-01
11	2022-01-01	2023-01-01
12	2022-01-01	2023-01-01

12 rows in set (0.259 sec)

idPenghuni	namaDepan	namaBelakang	jalan	RT	RW	kelurahan	kecamatan	kota	provinsi	kodePos
1	Budi	Santoso	Jl. Melati 123	6	12	Cipedes	Bandung	Jawa Barat	Jawa Barat	40135
2	Ahmad	Wibowo	Jl. Kencana 456	10	15	Demangan	Yogyakarta	Yogyakarta	DI Yogyakarta	55241
3	Rudi	Susilo	Jl. Anggrek 789	14	18	Sukolilo	Surabaya	Jawa Timur	Jawa Timur	60223
4	Dwi	Sulistyo	Jl. Dahlia 567	8	22	Tirto	Semarang	Jawa Tengah	Jawa Tengah	50148
5	Eko	Prasetyo	Jl. Mawar 890	7	11	Ciwidey	Bandung	Jawa Barat	Jawa Barat	40354
6	Hendra	Kusuma	Jl. Srikaya 123	5	17	Kasihani	Bantul	Yogyakarta	DI Yogyakarta	55183
7	Indra	Setiawan	Jl. Flamboyan 456	12	16	Kenjeran	Surabaya	Jawa Timur	Jawa Timur	60122
8	Ade	Wijaya	Jl. Anggrek 789	9	23	Genuk	Semarang	Jawa Tengah	Jawa Tengah	50245
9	Bayu	Nugroho	Jl. Kencur 234	11	14	Kalasan	Sleman	Yogyakarta	DI Yogyakarta	55571
10	Dicky	Saputra	Jl. Mawar 567	13	19	Mulyorejo	Surabaya	Jawa Timur	Jawa Timur	60223
11	Fandi	Surya	Jl. Kamboja 890	6	20	Banyumanik	Semarang	Jawa Tengah	Jawa Tengah	50192
12	Galih	Purnama	Jl. Raya 123	8	21	Cangkringan	Sleman	Yogyakarta	DI Yogyakarta	55582

12 rows in set (0.274 sec)

idPenghuni	email
1	budi.santoso@gmail.com
1	budi.santoso@yahoo.com
1	budi.santoso@outlook.com
2	ahmad.wibowo@gmail.com
2	ahmad.wibowo@yahoo.com
3	rudi.susilo@gmail.com
3	rudi.susilo@yahoo.com
3	rudi.susilo@outlook.com
4	dwi.sulistyo@gmail.com
4	dwi.sulistyo@yahoo.com
5	eko.prasetyo@gmail.com
5	eko.prasetyo@yahoo.com
5	eko.prasetyo@outlook.com
6	hendra.kusuma@gmail.com
6	hendra.kusuma@yahoo.com
7	indra.setiawan@gmail.com
8	ade.wijaya@gmail.com
9	bayu.nugroho@gmail.com
9	bayu.nugroho@yahoo.com
9	bayu.nugroho@outlook.com
10	dicky.saputra@gmail.com
11	fandi.surya@gmail.com
12	galih.purnama@gmail.com

23 rows in set (0.323 sec)

idPenghuni	noTelp
1	081234567890
2	081234567890
2	081112233445
3	081234567890
4	081112233445
5	081234567890
5	081112233445
6	081234567890
7	081112233445
8	081234567890
9	081112233445
10	081234567890
10	081112233445
11	081234567890
11	081112233445
12	081234567890
12	081112233445

17 rows in set (0.349 sec)

idTipeKamar	tipeKamar	hargaKamar	luasKamar
1	Tanpa Kamar Mandi	800000.00	12.50
2	Dengan Kamar Mandi	1200000.00	15.00

2 rows in set (0.362 sec)

idPenghuni	noKamar	idTipeKamar	idTransaksi	jenisTransaksi	jumlahBiaya	keterangan
1	1	1	1	sewa kamar	800000.00	Sewa kamar bulan Januari
1	1	1	2	listrik	200000.00	Listrik bulan Januari
1	1	1	3	air	150000.00	Air bulan Januari
2	2	1	4	sewa kamar	1200000.00	Sewa kamar bulan Januari
2	2	1	5	listrik	250000.00	Listrik bulan Januari
2	2	1	6	air	180000.00	Air bulan Januari
3	3	1	7	sewa kamar	800000.00	Sewa kamar bulan Januari
3	3	1	8	listrik	180000.00	Listrik bulan Januari
3	3	1	9	air	130000.00	Air bulan Januari
4	4	1	10	sewa kamar	1200000.00	Sewa kamar bulan Januari
4	4	1	11	listrik	220000.00	Listrik bulan Januari
4	4	1	12	air	160000.00	Air bulan Januari
5	5	2	13	sewa kamar	800000.00	Sewa kamar bulan Januari
5	5	2	14	listrik	190000.00	Listrik bulan Januari
5	5	2	15	air	140000.00	Air bulan Januari
6	6	1	16	sewa kamar	1200000.00	Sewa kamar bulan Januari
6	6	1	17	listrik	240000.00	Listrik bulan Januari
6	6	1	18	air	170000.00	Air bulan Januari
7	7	1	19	sewa kamar	800000.00	Sewa kamar bulan Januari
7	7	1	20	listrik	170000.00	Listrik bulan Januari
7	7	1	21	air	120000.00	Air bulan Januari
8	8	1	22	sewa kamar	1200000.00	Sewa kamar bulan Januari
8	8	1	23	listrik	210000.00	Listrik bulan Januari
8	8	1	24	air	150000.00	Air bulan Januari
9	9	2	25	sewa kamar	800000.00	Sewa kamar bulan Januari
9	9	2	26	listrik	200000.00	Listrik bulan Januari
9	9	2	27	air	140000.00	Air bulan Januari
10	10	1	28	sewa kamar	1200000.00	Sewa kamar bulan Januari
10	10	1	29	listrik	250000.00	Listrik bulan Januari
10	10	1	30	air	180000.00	Air bulan Januari
11	11	2	31	sewa kamar	800000.00	Sewa kamar bulan Januari
11	11	2	32	listrik	180000.00	Listrik bulan Januari
11	11	2	33	air	130000.00	Air bulan Januari
12	12	2	34	sewa kamar	1200000.00	Sewa kamar bulan Januari
12	12	2	35	listrik	220000.00	Listrik bulan Januari
12	12	2	36	air	160000.00	Air bulan Januari
1	1	1	37	sewa kamar	800000.00	Sewa kamar bulan Februari
1	1	1	38	listrik	200000.00	Listrik bulan Februari
1	1	1	39	air	150000.00	Air bulan Februari
2	2	1	40	sewa kamar	1200000.00	Sewa kamar bulan Februari
2	2	1	41	listrik	250000.00	Listrik bulan Februari
2	2	1	42	air	180000.00	Air bulan Februari
3	3	1	43	sewa kamar	800000.00	Sewa kamar bulan Februari
3	3	1	44	listrik	180000.00	Listrik bulan Februari
3	3	1	45	air	130000.00	Air bulan Februari
4	4	1	46	sewa kamar	1200000.00	Sewa kamar bulan Februari
4	4	1	47	listrik	220000.00	Listrik bulan Februari
4	4	1	48	air	160000.00	Air bulan Februari
5	5	2	49	sewa kamar	800000.00	Sewa kamar bulan Februari
5	5	2	50	listrik	190000.00	Listrik bulan Februari
5	5	2	51	air	140000.00	Air bulan Februari
6	6	1	52	sewa kamar	1200000.00	Sewa kamar bulan Februari
6	6	1	53	listrik	240000.00	Listrik bulan Februari
6	6	1	54	air	170000.00	Air bulan Februari
7	7	1	55	sewa kamar	800000.00	Sewa kamar bulan Februari
7	7	1	56	listrik	170000.00	Listrik bulan Februari
7	7	1	57	air	120000.00	Air bulan Februari
8	8	1	58	sewa kamar	1200000.00	Sewa kamar bulan Februari
8	8	1	59	listrik	210000.00	Listrik bulan Februari
8	8	1	60	air	150000.00	Air bulan Februari
9	9	2	61	sewa kamar	800000.00	Sewa kamar bulan Februari
9	9	2	62	listrik	200000.00	Listrik bulan Februari
9	9	2	63	air	140000.00	Air bulan Februari
10	10	1	64	sewa kamar	1200000.00	Sewa kamar bulan Februari
10	10	1	65	listrik	250000.00	Listrik bulan Februari
10	10	1	66	air	180000.00	Air bulan Februari
11	11	2	67	sewa kamar	800000.00	Sewa kamar bulan Februari
11	11	2	68	listrik	180000.00	Listrik bulan Februari

11	11	2	69	air	130000.00	Air bulan Februari
12	12	2	70	sewa kamar	1200000.00	Sewa kamar bulan Februari
12	12	2	71	listrik	220000.00	Listrik bulan Februari
12	12	2	72	air	160000.00	Air bulan Februari
1	1	1	73	sewa kamar	800000.00	Sewa kamar bulan Maret
1	1	1	74	listrik	200000.00	Listrik bulan Maret
1	1	1	75	air	150000.00	Air bulan Maret
2	2	1	76	sewa kamar	1200000.00	Sewa kamar bulan Maret
2	2	1	77	listrik	250000.00	Listrik bulan Maret
2	2	1	78	air	180000.00	Air bulan Maret
3	3	1	79	sewa kamar	800000.00	Sewa kamar bulan Maret
3	3	1	80	listrik	180000.00	Listrik bulan Maret
3	3	1	81	air	130000.00	Air bulan Maret
4	4	1	82	sewa kamar	1200000.00	Sewa kamar bulan Maret
4	4	1	83	listrik	220000.00	Listrik bulan Maret
4	4	1	84	air	160000.00	Air bulan Maret
5	5	2	85	sewa kamar	800000.00	Sewa kamar bulan Maret
5	5	2	86	listrik	190000.00	Listrik bulan Maret
5	5	2	87	air	140000.00	Air bulan Maret
6	6	1	88	sewa kamar	1200000.00	Sewa kamar bulan Maret
6	6	1	89	listrik	240000.00	Listrik bulan Maret
6	6	1	90	air	170000.00	Air bulan Maret
7	7	1	91	sewa kamar	800000.00	Sewa kamar bulan Maret
7	7	1	92	listrik	170000.00	Listrik bulan Maret
7	7	1	93	air	120000.00	Air bulan Maret
8	8	1	94	sewa kamar	1200000.00	Sewa kamar bulan Maret
8	8	1	95	listrik	210000.00	Listrik bulan Maret
8	8	1	96	air	150000.00	Air bulan Maret
9	9	2	97	sewa kamar	800000.00	Sewa kamar bulan Maret
9	9	2	98	listrik	200000.00	Listrik bulan Maret
9	9	2	99	air	140000.00	Air bulan Maret
10	10	1	100	sewa kamar	1200000.00	Sewa kamar bulan Maret
10	10	1	101	listrik	250000.00	Listrik bulan Maret
10	10	1	102	air	180000.00	Air bulan Maret
11	11	2	103	sewa kamar	800000.00	Sewa kamar bulan Maret
11	11	2	104	listrik	180000.00	Listrik bulan Maret
11	11	2	105	air	130000.00	Air bulan Maret
12	12	2	106	sewa kamar	1200000.00	Sewa kamar bulan Maret
12	12	2	107	listrik	220000.00	Listrik bulan Maret
12	12	2	108	air	160000.00	Air bulan Maret
1	1	1	109	denda	50000.00	Keterlambatan pembayaran sewa kamar bulan Maret
4	4	1	110	denda	60000.00	Keterlambatan pembayaran sewa kamar bulan Maret

110 rows in set (0.366 sec)

Query OK, 0 rows affected (0.709 sec)

Mengecek apakah kamarnya kosong atau tidak

```

DELIMITER //
CREATE PROCEDURE CekStatusKamar(IN noKamarInput INT)
BEGIN
    DECLARE idPenghuniDiKamar INT;

    SELECT idPenghuni INTO idPenghuniDiKamar
    FROM kamar
    WHERE noKamar = noKamarInput;

    IF idPenghuniDiKamar IS NOT NULL THEN
        SELECT CONCAT('Kamar No ', noKamarInput, ' sedang ditempati.') as status;
    ELSE
        SELECT CONCAT('Kamar No ', noKamarInput, ' kosong.') as status;
    END IF;
END //
DELIMITER ;

```

Cara memanggilnya

```
CALL CekStatusKamar(<no_kamar>);
```

Contoh output

```

MariaDB [kostpoetrasultan]> DELIMITER //
MariaDB [kostpoetrasultan]> CREATE PROCEDURE CekStatusKamar(IN noKamarInput INT)
-> BEGIN

```

```

-> DECLARE idPenghuniDiKamar INT;
->
-> SELECT idPenghuni INTO idPenghuniDiKamar
-> FROM kamar
-> WHERE noKamar = noKamarInput;
->
-> IF idPenghuniDiKamar IS NOT NULL THEN
->     SELECT CONCAT('Kamar No ', noKamarInput, ' sedang ditempati.') as
status;
-> ELSE
->     SELECT CONCAT('Kamar No ', noKamarInput, ' kosong.') as status;
-> END IF;
-> END //
Query OK, 0 rows affected (0.004 sec)

MariaDB [kostpoetrasultan]> DELIMITER ;
MariaDB [kostpoetrasultan]>
MariaDB [kostpoetrasultan]> CALL CekStatusKamar(1);
+-----+
| status |
+-----+
| Kamar No 1 sedang ditempati. |
+-----+
1 row in set (0.001 sec)

Query OK, 1 row affected (0.003 sec)

```

Soft reset untuk menghapus data yang ada di masing-masing tabel di database

```

DELIMITER //
CREATE PROCEDURE TruncateAllTablesInDatabase()
BEGIN
    DECLARE _done INT DEFAULT FALSE;
    DECLARE _tableName VARCHAR(255);
    DECLARE _cursor CURSOR FOR
        SELECT table_name
        FROM information_schema.tables
        WHERE table_schema = 'kostpoetrasultan';
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET _done = TRUE;

    SET FOREIGN_KEY_CHECKS = 0; -- menonaktifkan cek kunci asing

    OPEN _cursor;

    read_loop: LOOP
        FETCH _cursor INTO _tableName;

        IF _done THEN

```

```

        LEAVE read_loop;
    END IF;

    SET @sql = CONCAT('TRUNCATE TABLE kostpoetrasultan.',
        _tableName);
    PREPARE stmt FROM @sql;
    EXECUTE stmt;
    DEALLOCATE PREPARE stmt;
END LOOP;

CLOSE _cursor;

SET FOREIGN_KEY_CHECKS = 1;
END //
DELIMITER ;

```

Untuk memanggilnya

```
CALL TruncateAllTablesInDatabase();
```

Hasil output

```

MariaDB [kostpoetrasultan]> DELIMITER //
MariaDB [kostpoetrasultan]> CREATE PROCEDURE TruncateAllTablesInDatabase()
-> BEGIN
-> DECLARE _done INT DEFAULT FALSE;
-> DECLARE _tableName VARCHAR(255);
-> DECLARE _cursor CURSOR FOR
-> SELECT table_name
-> FROM information_schema.tables
-> WHERE table_schema = 'kostpoetrasultan';
-> DECLARE CONTINUE HANDLER FOR NOT FOUND SET _done = TRUE;
->
-> SET FOREIGN_KEY_CHECKS = 0; -- menonaktifkan cek kunci asing
->
-> OPEN _cursor;
->
-> read_loop: LOOP
->   FETCH _cursor INTO _tableName;
->
->   IF _done THEN
->     LEAVE read_loop;
->   END IF;
->
->   SET @sql = CONCAT('TRUNCATE TABLE kostpoetrasultan.', _tableName);
->   PREPARE stmt FROM @sql;
->   EXECUTE stmt;
->   DEALLOCATE PREPARE stmt;
-> END LOOP;
->
-> CLOSE _cursor;
->
-> SET FOREIGN_KEY_CHECKS = 1;
-> END //

```



```
Query OK, 0 rows affected (0.006 sec)
```

```
MariaDB [kostpoetrasultan]> DELIMITER ;
MariaDB [kostpoetrasultan]> CALL TruncateAllTablesInDatabase();
Query OK, 0 rows affected (0.054 sec)
```

```
MariaDB [kostpoetrasultan]> select * from transaksi;
Empty set (0.000 sec)
```

Setelah kita cek, ternyata tabel transaksinya sudah kosong.

Mencoba membuat store procedure, input penghuni dengan prosedur.

```
DELIMITER $$
CREATE PROCEDURE `tambahPenghuni` (
  IN _namaDepan VARCHAR(255),
  IN _namaBelakang VARCHAR(255),
  IN _jalan VARCHAR(255),
  IN _RT INT,
  IN _RW INT,
  IN _kelurahan VARCHAR(255),
  IN _kecamatan VARCHAR(255),
  IN _kota VARCHAR(255),
  IN _provinsi VARCHAR(255),
  IN _kodePos INT,
  IN _tanggalLahir DATE,
  IN _emails TEXT,
  IN _noTelp TEXT,
  IN _tglMulaiSewa DATE,
  IN _tglAkhirSewa DATE,
  IN _idTipeKamar INT
)
BEGIN
  DECLARE _idPenghuni INT;
  DECLARE _noKamar INT;
  DECLARE _email VARCHAR(255);
  DECLARE _telp VARCHAR(255);

  -- Menambahkan data penghuni
  INSERT INTO penghuni (namaDepan, namaBelakang, jalan, RT, RW, kelurahan, kecamatan, kota, provinsi,
    kodePos, tanggalLahir)
  VALUES (_namaDepan, _namaBelakang, _jalan, _RT, _RW, _kelurahan, _kecamatan, _kota, _provinsi,
    _kodePos, _tanggalLahir);
  SET _idPenghuni = LAST_INSERT_ID();

  -- Menambahkan data email penghuni
  SET @emails = _emails;
  WHILE LOCATE(',', @emails) > 0 DO
    SET _email = SUBSTRING(@emails, 1, LOCATE(',', @emails) - 1);
    SET @emails = SUBSTRING(@emails, LOCATE(',', @emails) + 1);
    INSERT INTO penghuni_email (idPenghuni, email) VALUES (_idPenghuni, _email);
  END WHILE;
  INSERT INTO penghuni_email (idPenghuni, email) VALUES (_idPenghuni, @emails);

  -- Menambahkan data no telp penghuni
  SET @telps = _noTelp;
  WHILE LOCATE(',', @telps) > 0 DO
    SET _telp = SUBSTRING(@telps, 1, LOCATE(',', @telps) - 1);
    SET @telps = SUBSTRING(@telps, LOCATE(',', @telps) + 1);
    INSERT INTO penghuni_noTelp (idPenghuni, noTelp) VALUES (_idPenghuni, _telp);
  END WHILE;
  INSERT INTO penghuni_noTelp (idPenghuni, noTelp) VALUES (_idPenghuni, @telps);

  -- Menambahkan data sewa penghuni
  INSERT INTO menyewa (idPenghuni, tglMulaiSewa, tglAkhirSewa) VALUES (_idPenghuni, _tglMulaiSewa,
    _tglAkhirSewa);

  -- Menambahkan data kamar
  INSERT INTO kamar (idTipeKamar, idPenghuni) VALUES (_idTipeKamar, _idPenghuni);
END
```

```

SET _noKamar = LAST_INSERT_ID();

-- Menambahkan data transaksi
INSERT INTO transaksi (idPenghuni, noKamar, idTipeKamar, jenisTransaksi, jumlahBiaya, keterangan)
VALUES (_idPenghuni, _noKamar, _idTipeKamar, 'sewa kamar', (SELECT hargaKamar FROM tipeKamar WHERE
idTipeKamar = _idTipeKamar), 'Transaksi sewa kamar otomatis');
INSERT INTO membayar (idPenghuni, noKamar, idTipeKamar, idTransaksi, tglTransaksi, metodePembayaran)
VALUES (_idPenghuni, _noKamar, _idTipeKamar, LAST_INSERT_ID(), CURDATE(), 'tunai');
END$$
DELIMITER ;

```

Cara pemanggilan, dengan contoh isi parameternya

```

CALL tambahPenghuni(
    'Ahmad',
    'Budi',
    'Jl. Sudirman No. 45',
    1,
    2,
    'Kebayoran Baru',
    'Jakarta Selatan',
    'Jakarta',
    'DKI Jakarta',
    12110,
    '1990-05-10',
    'ahmadbudi@gmail.com,ahmad.budi@yahoo.com',
    '081234567890,085612345678',
    '2023-01-01',
    '2023-12-31',
    1
);

```

Output

```

MariaDB [kostpoetrasultan]> CALL tambahPenghuni(
-> 'Ahmad',
-> 'Budi',
-> 'Jl. Sudirman No. 45',
-> 1,
-> 2,
-> 'Kebayoran Baru',
-> 'Jakarta Selatan',
-> 'Jakarta',
-> 'DKI Jakarta',
-> 12110,
-> '1990-05-10',
-> 'ahmadbudi@gmail.com,ahmad.budi@yahoo.com',
-> '081234567890,085612345678',
-> '2023-01-01',
-> '2023-12-31',
-> 1
-> );
Query OK, 9 rows affected (0.009 sec)

MariaDB [kostpoetrasultan]> select * from penghuni;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| idPenghuni | namaDepan | namaBelakang | jalan | RT | RW | kelurahan | kecamatan | kota | provinsi | kodePos | tanggalahir |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Ahmad | Budi | Jl. Sudirman No. 45 | 1 | 2 | Kebayoran Baru | Jakarta Selatan | Jakarta | DKI Jakarta | 12110 | 1990-05-10 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [kostpoetrasultan]> select * from penghuni_email;
+-----+-----+
| idPenghuni | email |
+-----+-----+
| 1 | ahmadbudi@gmail.com |
| 1 | ahmad.budi@yahoo.com |
+-----+-----+
2 rows in set (0.000 sec)

MariaDB [kostpoetrasultan]> select * from penghuni_notelp;
+-----+-----+
| idPenghuni | noTelp |
+-----+-----+
| 1 | 081234567890 |
| 1 | 085612345678 |
+-----+-----+
2 rows in set (0.000 sec)

MariaDB [kostpoetrasultan]> select * from kamar;
+-----+-----+-----+
| noKamar | idTipeKamar | idPenghuni |
+-----+-----+-----+

```

```

+-----+-----+-----+
| 1 | 1 | 1 |
+-----+-----+-----+
1 row in set (0.000 sec)

```

Store Procedure Event Scheduler apabila sudah awal bulan yaitu setiap tanggal 1, maka semua penghuni akan ditetapkan biaya kost (sesuai tipe kamar yang dipilih) dan ditaruh di tabel transaksi.

```

CREATE PROCEDURE generateBiayaKost()
BEGIN
    INSERT INTO transaksi (idPenghuni, noKamar, idTipeKamar,
jenisTransaksi, jumlahBiaya, keterangan)
    SELECT k.idPenghuni, k.noKamar, k.idTipeKamar, 'sewa kamar',
t.hargaKamar, 'Biaya kost bulanan'
    FROM kamar k
    JOIN tipeKamar t ON k.idTipeKamar = t.idTipeKamar
    WHERE k.idPenghuni IS NOT NULL;
END;

CREATE EVENT generateBiayaKostEvent
ON SCHEDULE EVERY 1 MONTH STARTS '2023-01-01'
DO CALL generateBiayaKost();

```

Jangan lupa menyalakan

```
SET GLOBAL event_scheduler = ON;
```

Prosedur analisis transaksi oleh pemilik kost

```

DELIMITER //
CREATE PROCEDURE analyzeTransactions()
BEGIN
    DECLARE v_finished INTEGER DEFAULT 0;
    DECLARE v_jenisTransaksi ENUM('sewa kamar', 'air', 'listrik', 'denda');
    DECLARE v_totalOmset DECIMAL(10,2);
    DECLARE v_rataRata DECIMAL(10,2);
    DECLARE v_max DECIMAL(10,2);
    DECLARE v_min DECIMAL(10,2);
    DECLARE v_maxMonth INT;
    DECLARE v_minMonth INT;
    DECLARE v_totalTransaksi INT;
    DECLARE v_rataTransaksiPerBulan DECIMAL(10,2);

    DECLARE v_totalOmsetAll DECIMAL(10,2);
    DECLARE v_totalTransaksiAll INT;
    DECLARE v_rataRataAll DECIMAL(10,2);
    DECLARE v_maxAll DECIMAL(10,2);
    DECLARE v_minAll DECIMAL(10,2);
    DECLARE v_maxMonthAll INT;
    DECLARE v_minMonthAll INT;
    DECLARE v_rataTransaksiPerBulanAll DECIMAL(10,2);

    -- cursor untuk iterasi melalui jenis transaksi
    DECLARE transaksi_cursor CURSOR FOR SELECT DISTINCT jenisTransaksi FROM transaksi;

    -- handler untuk deteksi akhir cursor
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET v_finished = 1;

```

```

OPEN transaksi_cursor;

get_transaksi: LOOP
    FETCH transaksi_cursor INTO v_jenisTransaksi;
    IF v_finished = 1 THEN
        LEAVE get_transaksi;
    END IF;

    -- mendapatkan total omset, rata-rata, max, min dan bulan dari transaksi max dan min
    SELECT COUNT(*) INTO v_totalTransaksi FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi;
    SELECT SUM(jumlahBiaya) INTO v_totalOmset FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi;
    SELECT AVG(jumlahBiaya) INTO v_rataRata FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi;
    SELECT MAX(jumlahBiaya) INTO v_max FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi;
    SELECT MIN(jumlahBiaya) INTO v_min FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi;

    SELECT MONTH((SELECT tglTransaksi FROM membayar WHERE idTransaksi = (SELECT idTransaksi FROM
    transaksi WHERE jenisTransaksi = v_jenisTransaksi AND jumlahBiaya = v_max LIMIT 1))) INTO v_maxMonth;
    SELECT MONTH((SELECT tglTransaksi FROM membayar WHERE idTransaksi = (SELECT idTransaksi FROM
    transaksi WHERE jenisTransaksi = v_jenisTransaksi AND jumlahBiaya = v_min LIMIT 1))) INTO v_minMonth;

    -- mendapatkan rata-rata transaksi per bulan
    SELECT COUNT(*) / 12 INTO v_rataTransaksiPerBulan FROM membayar WHERE idTransaksi IN (SELECT
    idTransaksi FROM transaksi WHERE jenisTransaksi = v_jenisTransaksi) AND YEAR(tglTransaksi) =
    YEAR(CURDATE());

    -- output hasil
    SELECT v_jenisTransaksi AS 'Jenis Transaksi', v_totalOmset AS 'Total Omset', v_totalTransaksi AS
    'Jumlah Transaksi', v_rataRata AS 'Rata-Rata Transaksi', v_max AS 'Transaksi Tertinggi', v_maxMonth AS
    'Bulan Transaksi Tertinggi', v_min AS 'Transaksi Terendah', v_minMonth AS 'Bulan Transaksi Terendah',
    v_rataTransaksiPerBulan AS 'Rata-Rata Transaksi Per Bulan';

END LOOP get_transaksi;

SELECT COUNT(*) INTO v_totalTransaksiAll FROM transaksi;
SELECT SUM(jumlahBiaya) INTO v_totalOmsetAll FROM transaksi;
SELECT AVG(jumlahBiaya) INTO v_rataRataAll FROM transaksi;
SELECT MAX(jumlahBiaya) INTO v_maxAll FROM transaksi;
SELECT MIN(jumlahBiaya) INTO v_minAll FROM transaksi;

SELECT MONTH((SELECT tglTransaksi FROM membayar WHERE idTransaksi = (SELECT idTransaksi FROM transaksi
WHERE jumlahBiaya = v_maxAll LIMIT 1))) INTO v_maxMonthAll;
SELECT MONTH((SELECT tglTransaksi FROM membayar WHERE idTransaksi = (SELECT idTransaksi FROM transaksi
WHERE jumlahBiaya = v_minAll LIMIT 1))) INTO v_minMonthAll;

SELECT COUNT(*) / 12 INTO v_rataTransaksiPerBulanAll FROM membayar WHERE idTransaksi IN (SELECT
idTransaksi FROM transaksi) AND YEAR(tglTransaksi) = YEAR(CURDATE());

SELECT 'TOTAL' AS 'Jenis Transaksi', v_totalOmsetAll AS 'Total Omset', v_totalTransaksiAll AS 'Jumlah
Transaksi', v_rataRataAll AS 'Rata-Rata Transaksi', v_maxAll AS 'Transaksi Tertinggi', v_maxMonthAll AS
'Bulan Transaksi Tertinggi', v_minAll AS 'Transaksi Terendah', v_minMonthAll AS 'Bulan Transaksi
Terendah', v_rataTransaksiPerBulanAll AS 'Rata-Rata Transaksi Per Bulan';

CLOSE transaksi_cursor;

END//
DELIMITER ;

```

Output;

```

MariaDB [kostpoetra] > CALL analyzeTransactions();
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Jenis Transaksi | Total Omset | Jumlah Transaksi | Rata-Rata Transaksi | Transaksi Tertinggi | Bulan Transaksi Tertinggi | Transaksi Terendah | Bulan Transaksi Terendah | Rata-Rata Transaksi Per Bulan |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| sewa kamar      | 2.00      | 36 | 100000.00 | 120000.00 | NULL | 80000.00 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Jenis Transaksi | Total Omset | Jumlah Transaksi | Rata-Rata Transaksi | Transaksi Tertinggi | Bulan Transaksi Tertinggi | Transaksi Terendah | Bulan Transaksi Terendah | Rata-Rata Transaksi Per Bulan |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| listrik         | 2.00      | 36 | 209166.67 | 250000.00 | NULL | 170000.00 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.004 sec)

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Jenis Transaksi | Total Omset | Jumlah Transaksi | Rata-Rata Transaksi | Transaksi Tertinggi | Bulan Transaksi Tertinggi | Transaksi Terendah | Bulan Transaksi Terendah | Rata-Rata Transaksi Per Bulan |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| air            | 2.00      | 36 | 158833.33 | 180000.00 | NULL | 120000.00 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.007 sec)

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Jenis Transaksi | Total Omset | Jumlah Transaksi | Rata-Rata Transaksi | Transaksi Tertinggi | Bulan Transaksi Tertinggi | Transaksi Terendah | Bulan Transaksi Terendah | Rata-Rata Transaksi Per Bulan |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| denda          | 0.00      | 2 | 55000.00 | 60000.00 | NULL | 50000.00 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

1 row in set (0.009 sec)

Jenis Transaksi	Total Debet	Jumlah Transaksi	Rata-Rata Transaksi	Transaksi Tertinggi	Bulan Transaksi Tertinggi	Transaksi Terendah	Bulan Transaksi Terendah	Rata-Rata Transaksi Per Bulan
TOTAL	6.17	49070000.00	110	446000.91	12000000.00	NULL	50000.00	NULL

1 row in set (0.012 sec)

Query OK, 40 rows affected, 1 warning (0.015 sec)

2. Trigger

Pencegahan, apabila kita Insert itu harus kamar yang kosong

```
DELIMITER //
CREATE TRIGGER check_availability
BEFORE INSERT ON kamar
FOR EACH ROW
BEGIN
    DECLARE roomOccupied INT;
    SELECT COUNT(*) INTO roomOccupied FROM kamar WHERE noKamar =
    NEW.noKamar AND idPenghuni IS NOT NULL;
    IF roomOccupied > 0 THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Gagal input: Kamar sudah
ditempati.';
    END IF;
END;
//
DELIMITER ;
```

Output jika sudah ditempati, dengan custom error.

```
MariaDB [kostpoetrasultan]> INSERT INTO kamar(noKamar,
idTipeKamar, idPenghuni) VALUES (1, 1, 1);
ERROR 1644 (45000): Gagal input: Kamar sudah ditempati.
```

Membuat trigger akan berjalan setelah melakukan operasi INSERT pada tabel penghuni. Trigger akan otomatis membuat transaksi sewa kamar dengan harga kamar sesuai tipeKamar dari penghuni baru. Setelah itu, trigger juga akan membuat entri pada tabel membayar dengan metode pembayaran default sebagai 'tunai' dan tanggal transaksi adalah tanggal saat ini.

```
DELIMITER //
CREATE TRIGGER BiayaSewaKamarOtomatis
AFTER INSERT ON kamar
FOR EACH ROW
BEGIN
    DECLARE hargaKamar DECIMAL(10,2);
    DECLARE idTransaksi INT;

    SELECT hargaKamar INTO hargaKamar FROM tipeKamar WHERE
idTipeKamar = NEW.idTipeKamar;
```



```

INSERT INTO transaksi(idPenghuni, noKamar, idTipeKamar,
jenisTransaksi, jumlahBiaya, keterangan)
VALUES (NEW.idPenghuni, NEW.noKamar, NEW.idTipeKamar, 'sewa
kamar', hargaKamar, 'Transaksi sewa kamar otomatis');

SET idTransaksi = LAST_INSERT_ID();

IF hargaKamar IS NULL THEN
DELETE FROM transaksi WHERE idTransaksi = idTransaksi;
END IF;
END;
//
DELIMITER ;

```

Output

```

MariaDB [kostpoetrasultan]> CALL tambahPenghuni(
-> 'Ahmad',
-> 'Budi',
-> 'Jl. Sudirman No. 45',
-> 1,
-> 2,
-> 'Kebayoran Baru',
-> 'Jakarta Selatan',
-> 'Jakarta',
-> 'DKI Jakarta',
-> 12110,
-> '1990-05-10',
-> 'ahmadbudi@gmail.com,ahmad.budi@yahoo.com',
-> '081234567890,085612345678',
-> '2023-01-01',
-> '2023-12-31',
-> 1
-> );
Query OK, 12 rows affected (0.015 sec)

MariaDB [kostpoetrasultan]> select * from transaksi;
+-----+-----+-----+-----+-----+-----+-----+
| idPenghuni | noKamar | idTipeKamar | idTransaksi | jenisTransaksi | jumlahBiaya | keterangan |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 2 | sewa kamar | 800000.00 | Transaksi sewa kamar otomatis |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

```

BAB III

PENUTUP

3.1. Kesimpulan

Berdasarkan pembahasan pada laporan ini, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Telah berhasil dirancang basis data untuk sistem informasi Kost Poetra Sultan dengan memodelkan proses bisnis yang berlangsung saat ini ke dalam diagram ER yang terdiri dari 4 entitas utama yaitu Penghuni, Kamar, TipeKamar, dan Transaksi.
2. Entitas pada diagram ER telah diimplementasikan menjadi 8 buah tabel fisik basis data, yaitu:
 - Tabel Penghuni
 - Tabel Penghuni_Email
 - Tabel Penghuni_NoTelp
 - Tabel Menyewa
 - Tabel Kamar
 - Tabel TipeKamar
 - Tabel Transaksi
 - Tabel Membayar
3. Relasi antar entitas pada diagram ER telah diimplementasikan dengan membuat foreign key pada tabel yang terkait. Sehingga data pada masing-masing tabel saling terhubung dan konsisten.
4. Setiap atribut pada setiap entitas telah ditentukan tipe data SQL yang sesuai, seperti INTEGER, VARCHAR, DATE, DECIMAL, dan ENUM. Hal ini akan memastikan bahwa data yang disimpan sesuai dengan karakteristik masing-masing atribut.
5. Pemilihan tipe data yang tepat juga berpengaruh terhadap efisiensi operasi basis data seperti proses insert, update, search, join table, dan query lainnya. Dengan tipe data yang optimal diharapkan performa basis data menjadi lebih baik.
6. Selain merancang struktur tabel basis data, juga telah dibuat berbagai store procedure, trigger, dan query SQL untuk:
 - Melihat detail struktur tabel
 - Menampilkan isi seluruh tabel
 - Mengecek status kamar

- Melakukan CRUD data penghuni
- Reset data pada tabel
- Menganalisis transaksi
- dan lainnya

Basis data yang dirancang dapat menyimpan seluruh data yang dibutuhkan sistem informasi Kost Poetra Sultan seperti:

- Data profil penghuni
 - Data kamar
 - Data tipe kamar
 - Tanggal mulai dan selesai sewa
 - Transaksi keuangan
 - dan lainnya
7. Dengan basis data yang terkomputerisasi, berbagai permasalahan dalam pengelolaan Kost Poetra Sultan dapat diminimalisir seperti:
- Kurangnya transparansi informasi keuangan
 - Buruknya kualitas jaringan wifi
 - Tidak adanya informasi durasi tinggal penghuni
 - Terbatasnya informasi untuk calon penghuni
 - Lemahnya sistem keamanan
 - Tidak ada sistem denda
 - dan lainnya
8. Secara keseluruhan, basis data yang dirancang dalam laporan ini dapat menjadi acuan implementasi basis data untuk mendukung sistem informasi Kost Poetra Sultan agar lebih efisien dan efektif dalam operasional sehari-hari.

3.2. Kritik dan Saran

Berdasarkan analisis dan pembahasan yang telah dilakukan, berikut adalah beberapa kritik dan saran yang dapat diberikan terkait perancangan basis data pada laporan ini:

1. Basis data yang dirancang baru sebatas struktur tabel beserta hubungan antar tabelnya. Belum sampai pada tahap implementasi pembuatan basis data secara fisik dan pengembangan front end sistem informasinya.
2. Proses normalisasi mungkin masih perlu dilakukan lebih lanjut untuk mengoptimalkan struktur tabel yang dihasilkan dari diagram ER agar lebih efisien.
3. Pengujian performa basis data secara lebih komprehensif diperlukan terutama saat sudah diimplementasikan secara fisik dengan jumlah data yang besar.
4. Perlu dilakukan analisis risiko keamanan dan backup basis data secara berkala untuk menghindari hilangnya data.
5. Pembuatan interface sistem informasi yang user friendly agar pengguna awam seperti penghuni kost dapat dengan mudah mengakses dan memanfaatkan data.
6. Perlu adanya pelatihan dan pendampingan kepada user dalam mengoperasikan sistem informasi berbasis komputerisasi agar tidak gagap teknologi.
7. Sistem informasi perlu dikembangkan ke arah mobile sehingga dapat diakses melalui perangkat mobile pengguna.
8. Backup basis data secara berkala dan off-site sangat disarankan untuk menghindari hilangnya data akibat kerusakan sistem.
9. Pemantauan kinerja dan integritas basis data perlu dilakukan secara rutin untuk memastikan tidak ada masalah.
10. Diperlukan dokumentasi basis data yang lengkap agar memudahkan pengembangan ke depannya oleh tim IT lain.

Demikian beberapa kritik dan saran yang dapat diberikan terkait perancangan basis data pada laporan ini. Semua masukan tersebut diharapkan dapat menjadi bahan pertimbangan untuk pengembangan lebih lanjut sistem informasi Kost Poetra Sultan agar dapat memberikan manfaat yang optimal bagi seluruh pemangku kepentingan.

DAFTAR GAMBAR

Gambar 1 Lokasi Kost Poetra Sultan.....	4
Gambar 2 Proses Bisnis Kost Poetra Sultan.	7
Gambar 3 menunjukkan entitas penghuni beserta atribut-atributnya.	9
Gambar 4 menampilkan entitas kamar beserta atribut-atributnya.	10
Gambar 5 Entitas tipeKamar.....	10
Gambar 6 entitas transaksi.	11
Gambar 7 Penghuni-0.N->menyewa<-1.N-kamar.....	12
Gambar 8 kamar-1.1->memiliki<-1.N-tipekamar.....	13
Gambar 9 Agregasi (penghuni, kamar, tipeKamar)-1.1->membayar<-1.N-transaksi	13
Gambar 10 Diagram ER pada Kasus Kost Poetra Sultan	16
Gambar 11 Tabel yang dihasilkan dari diagram ER pada kasus Kost Poetra Sultan.....	17
Gambar 12 Tampilan Database Designer pada kasus Kost Poetra Sultan	31

