

Makalah Basis Data (MySQL) DATABASE HOTEL



Anggota Kelompok :

1.Ryandra Athaya Saleh	0110223013
2.Satria Tri Ferdiansyah	0110223039
3.Firenze Higa Putra	0110223014
4.Shafhan Farizi	0110223019
5.M. Yuking Niqobal Haqqi	0110223026
6.Fadlan Zaki Ahmadani	0110223038

Sekolah Tinggi Teknik Terpadu Nurul Fikri

Program Studi Teknik Informatika

2023

KATA PENGANTAR

Puji Syukur kami panjatkan ke hadirat Allah Subhanahuwata'ala, karena hanya dengan hidayah dan TaufikNya-lah, sehingga penulis dapat menyelesaikan makalah ini dengan baik sesuai jadwal yang telah ditetapkan.

Penyusunan makalah ini dalam rangka memenuhi tugas Mata Kuliah Basis Data. Selain itu, makalah ini bertujuan menambah wawasan bagi para pembaca dan juga bagi penulis. Penulis mengucapkan terima kasih kepada Bapak Nasrul, S.Pdi., S.Kom., M.Kom. selaku dosen dan Bapak Sigit, S.Kom. selaku Asisten Dosen Mata Kuliah Basis Data.

Akhirnya penulis berharap semoga Allah memberikan imbalan yang setimpal pada mereka yang memberikan bantuan dan dapat menjadikan semua bantuan ini sebagai ibadah, Amiin Yaa Robbal ' Alamiin

Depok, 1 Juni 2024

Penyusun,

DAFTAR ISI

Makalah Basis Data (MySQL) DATABASE HOTEL i

KATA PENGANTAR..... ii

DAFTAR ISI..... 3

DAFTAR GAMBAR..... 4

BAB I PENDAHULUAN 5

 1.1. Pengertian MySQL 5

 1.2. Pengertian MySQL Workbench 5

BAB II PEMBAHASAN 6

 2.1. Database 6

 2.2.1. Nama kolom dan Tipe Data : 6

 2.2.2. Input Data Tabel 7

 2.2. Join Table dan Alias..... 9

 2.3. Query View 10

 2.4. Query Index 10

 2.5. Query Transaction 11

 2.6. Query Commit..... 11

 2.7. Query Rollback..... 11

 2.8. Query Savepoint 11

 2.9. Query Procedure 12

 2.10. Query Call..... 12

 2.11. Query Delimiter..... 13

 2.12. Query Trigger..... 13

 2.13. Query Grant..... 13

 2.14. Query Revoke 15

 2.15. Query Backup & Restore 15

BAB III KESIMPULAN DAN SARAN 16

DAFTAR GAMBAR

Gambar 2. 1 Database..... 6

Gambar 2. 2 Tabel jasa 6

Gambar 2. 3 Tabel Kamar 6

Gambar 2. 4 Tabel Pembayaran 6

Gambar 2. 5 Tabel Pemesanan 7

Gambar 2. 6 Tabel Penggunaan Jasa 7

Gambar 2. 7 Tabel Tamu 7

Gambar 2. 8 Table Tipe Kamar 7

Gambar 2. 9 Input Tabel Jasa 7

Gambar 2. 10 Input Tabel Kamar 8

Gambar 2. 11 Input Tabel Pembayaran 8

Gambar 2. 12 Input Tabel Pemesanan..... 8

Gambar 2. 13 Input Penggunaan Jasa..... 8

Gambar 2. 14 Input Tabel Tamu 8

Gambar 2. 15 Input Tipe Kamar..... 8

Gambar 2. 16 Join 2 Table dan 4 Table..... 9

Gambar 2. 17 Alias 2 table dan 4 table..... 10

Gambar 2. 18 Query View..... 10

Gambar 2. 19 Memanggil View 10

Gambar 2. 20 Query Index 10

Gambar 2. 21 Query Transaction 11

Gambar 2. 22 Query Commit 11

Gambar 2. 23 Query Rollback..... 11

Gambar 2. 24 Query Savepoint 11

Gambar 2. 25 Query Procedure 12

Gambar 2. 26 Query Call..... 12

Gambar 2. 27 Query Delimiter 13

Gambar 2. 28 Query Trigger 13

Gambar 2. 29 Grant All Privileges 13

Gambar 2. 30 Bukti user memiliki privileges..... 14

Gambar 2. 31 Grant SELECT dan DELETE Privileges..... 14

Gambar 2. 32 Bukti user hanya memiliki privileges database hotel pada table pemesanan . 14

Gambar 2. 33 Revoke user 15

Gambar 2. 34 Bukti User Tidak Memiliki Privileges..... 15

Gambar 2. 35 Backup database 15

Gambar 2. 36 Restore Database 15

Gambar 2. 37 Bukti database direstore..... 15

BAB I PENDAHULUAN

1.1.Pengertian MySQL

MySQL adalah Sebuah program database server yang mampu menerima dan mengirimkan datanya sangat cepat, multi user serta menggunakan perintah dasar SQL (Structured Query Language). MySQL merupakan dua bentuk lisensi, yaitu FreeSoftware dan Shareware. MySQL yang biasa kita gunakan adalah MySQL FreeSoftware yang berada dibawah Lisensi GNU/GPL (General Public License). MySQL Merupakan sebuah database server yang free, artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya.

MySQL pertama kali dirintis oleh seorang programmer database bernama Michael Widenius . Selain database server, MySQL juga merupakan program yang dapat mengakses suatu database MySQL yang berposisi sebagai Server, yang berarti program kita berposisi sebagai Client. Jadi MySQL adalah sebuah database yang dapat digunakan sebagai Client maupun server. Database MySQL merupakan suatu perangkat lunak database yang berbentuk database relasional atau disebut Relational Database Management System (RDBMS) yang menggunakan suatu bahasa permintaan yang bernama SQL (Structured Query Language).

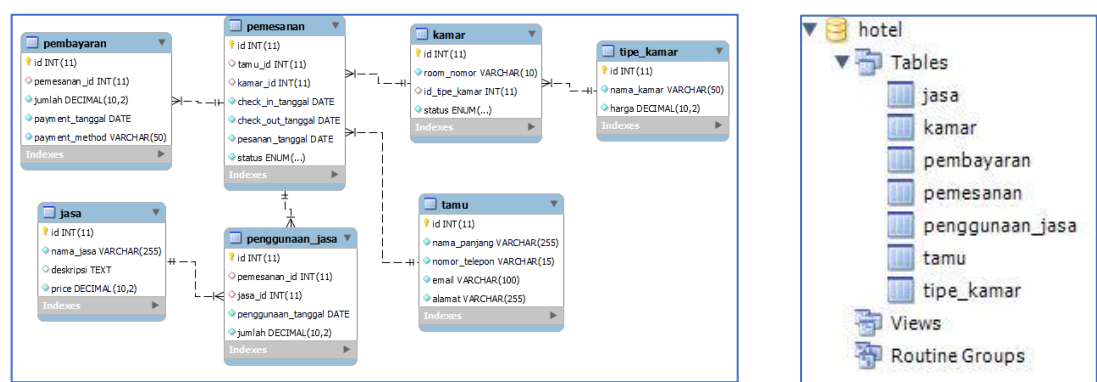
1.2.Pengertian MySQL Workbench

MySQL Workbench adalah perangkat lunak yang powerful dan mudah digunakan untuk mengelola basis data MySQL. Perangkat ini menyediakan berbagai fitur untuk merancang, mengembangkan, dan mengelola basis data MySQL.

BAB II PEMBAHASAN

2.1.Database

Database adalah sebuah media utama yang harus dibuat dalam membangun sebuah basis data agar nantinya dapat kita letakkan beberapa tabel dengan field-fieldnya.:



Gambar 2. 1 Database

2.2.1. Nama kolom dan Tipe Data :

1. Tabel Jasa

```
MariaDB [hotel]> desc jasa;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nama_jasa	varchar(255)	NO		NULL	
deskripsi	text	YES		NULL	
price	decimal(10,2)	NO		NULL	

4 rows in set (0.011 sec)

```
MariaDB [hotel]>
```

Gambar 2. 2 Tabel jasa

2. Tabel Kamar

```
MariaDB [hotel]> desc kamar;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
room_nomor	varchar(10)	NO		NULL	
id_tipe_kamar	int(11)	YES	MUL	NULL	
status	enum('Tersedia','Tidak Tersedia','Perbaikan')	NO		NULL	

4 rows in set (0.012 sec)

Gambar 2. 3 Tabel Kamar

3. Tabel Pembayaran

```
MariaDB [hotel]> desc pembayaran;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
pemesanan_id	int(11)	YES	MUL	NULL	
jumlah	decimal(10,2)	NO		NULL	
payment_tanggal	date	NO		NULL	
payment_method	varchar(50)	NO		NULL	

5 rows in set (0.014 sec)

Gambar 2. 4 Tabel Pembayaran

4. Tabel Pemesanan

```
MariaDB [hotel]> desc pemesanan;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
tamu_id	int(11)	YES	MUL	NULL	
kamar_id	int(11)	YES	MUL	NULL	
check_in_tanggal	date	NO		NULL	
check_out_tanggal	date	NO		NULL	
pesanan_tanggal	date	NO		NULL	
status	enum('Dikonfirmasi','Dibatalkan','Selesai')	NO		NULL	

7 rows in set (0.022 sec)

Gambar 2. 5 Tabel Pemesanan

5. Tabel Penggunaan Jasa

```
MariaDB [hotel]> desc penggunaan_jasa;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
pemesanan_id	int(11)	YES	MUL	NULL	
jasa_id	int(11)	YES	MUL	NULL	
penggunaan_tanggal	date	NO		NULL	
jumlah	decimal(10,2)	NO		NULL	

5 rows in set (0.011 sec)

Gambar 2. 6 Tabel Penggunaan Jasa

6. Tabel Tamu

```
MariaDB [hotel]> desc tamu;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nama_panjang	varchar(255)	NO		NULL	
nomor_telepon	varchar(15)	NO		NULL	
email	varchar(100)	NO		NULL	
alamat	varchar(255)	NO		NULL	

5 rows in set (0.010 sec)

Gambar 2. 7 Tabel Tamu

7. Tabel Tipe Kamar

```
MariaDB [hotel]> desc tipe_kamar;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nama_kamar	varchar(50)	NO		NULL	
harga	decimal(10,2)	NO		NULL	

3 rows in set (0.022 sec)

Gambar 2. 8 Table Tipe Kamar

2.2.2. Input Data Tabel

1. Input Tabel Jasa

```
INSERT INTO jasa (nama_jasa, deskripsi, price) VALUES
('Room Service', '24/7 Room Service', 20.00),
('Laundry', 'Laundry Service', 15.00),
('Spa', 'Spa Service', 50.00);
```

Gambar 2. 9 Input Tabel Jasa

2. Input Tabel Kamar

```
INSERT INTO kamar (room_nomor, id_tipe_kamar, status) VALUES
('101', 1, 'Tersedia'),
('102', 2, 'Tidak Tersedia'),
('103', 1, 'Perbaikan'),
('104', 3, 'Tersedia');
```

Gambar 2. 10 Input Tabel Kamar

3. Input Tabel Pembayaran

```
INSERT INTO pembayaran (pemesanan_id, jumlah, payment_tanggal, payment_method) VALUES
(1, 400.00, '2024-05-30', 'credit card'),
(2, 750.00, '2024-06-05', 'bank transfer'),
(3, 1500.00, '2024-06-10', 'cash'),
(4, 1000.00, '2024-06-25', 'credit card');
```

Gambar 2. 11 Input Tabel Pembayaran

4. Input Tabel Pemesanan

```
INSERT INTO pemesanan (tamu_id, kamar_id, check_in_tanggal, check_out_tanggal, pesanan_tanggal, status) VALUES
(1, 1, '2024-06-01', '2024-06-05', '2024-05-30', 'Dikonfirmasi'),
(2, 2, '2024-06-10', '2024-06-15', '2024-06-05', 'Dibatalkan'),
(3, 3, '2024-06-20', '2024-06-25', '2024-06-10', 'Selesai'),
(4, 4, '2024-07-01', '2024-07-05', '2024-06-25', 'Dikonfirmasi');
```

Gambar 2. 12 Input Tabel Pemesanan

5. Input Tabel Penggunaan Jasa

```
INSERT INTO penggunaan_jasa (pemesanan_id, jasa_id, penggunaan_tanggal, jumlah) VALUES
(1, 1, '2024-06-02', 20.00),
(2, 2, '2024-06-11', 15.00),
(3, 3, '2024-06-21', 50.00),
(4, 1, '2024-07-02', 20.00);
```

Gambar 2. 13 Input Penggunaan Jasa

6. Input Tabel Tamu

```
INSERT INTO tamu (nama_panjang, nomor_telepon, email, alamat) VALUES
('John Doe', '555-1234', 'john.doe@example.com', '123 Main St'),
('Jane Smith', '555-5678', 'jane.smith@example.com', '456 Elm St'),
('Jim Beam', '555-8765', 'jim.beam@example.com', '789 Oak St'),
('Jessica Jones', '555-4321', 'jessica.jones@example.com', '321 Pine St');
```

Gambar 2. 14 Input Tabel Tamu

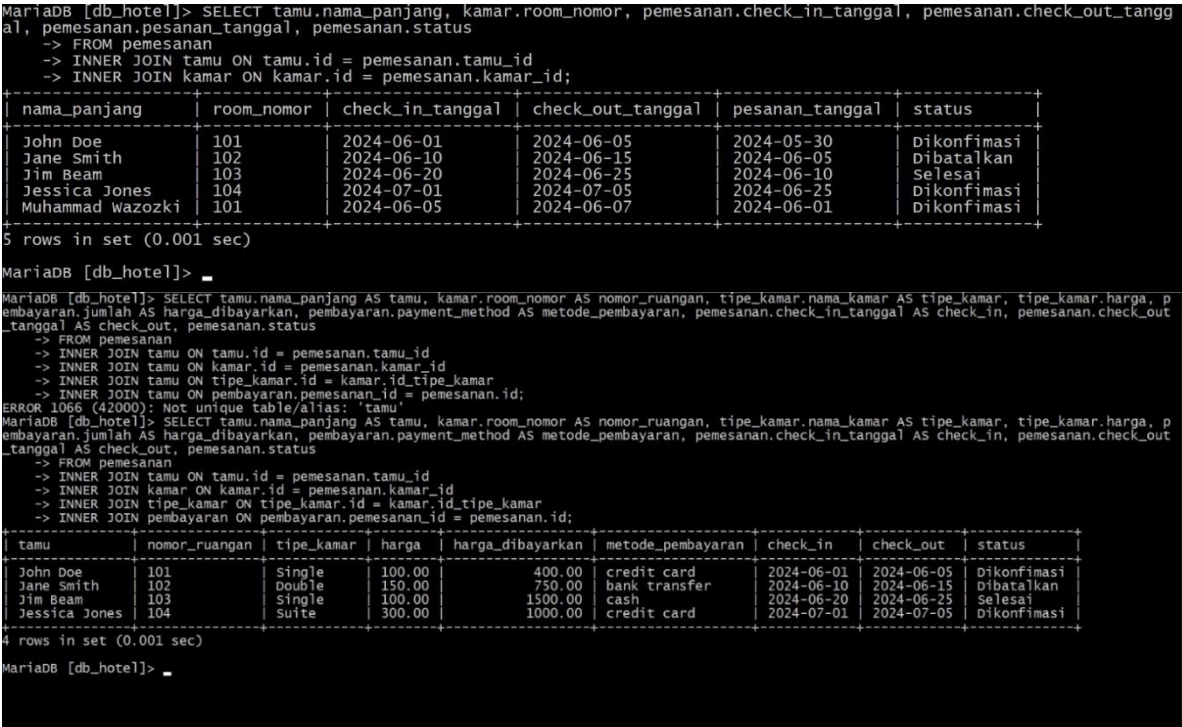
7. Input Tipe Kamar

```
INSERT INTO tipe_kamar (nama_kamar, harga) VALUES
('Single', 100.00),
('Double', 150.00),
('Suite', 300.00);
```

Gambar 2. 15 Input Tipe Kamar

2.2.Join Table dan Alias

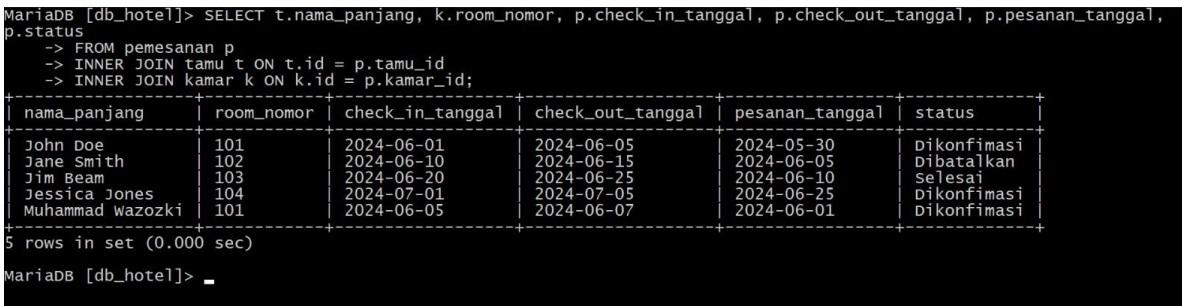
JOIN digunakan untuk menggabungkan baris dari dua atau lebih tabel berdasarkan kolom terkait di antara mereka.

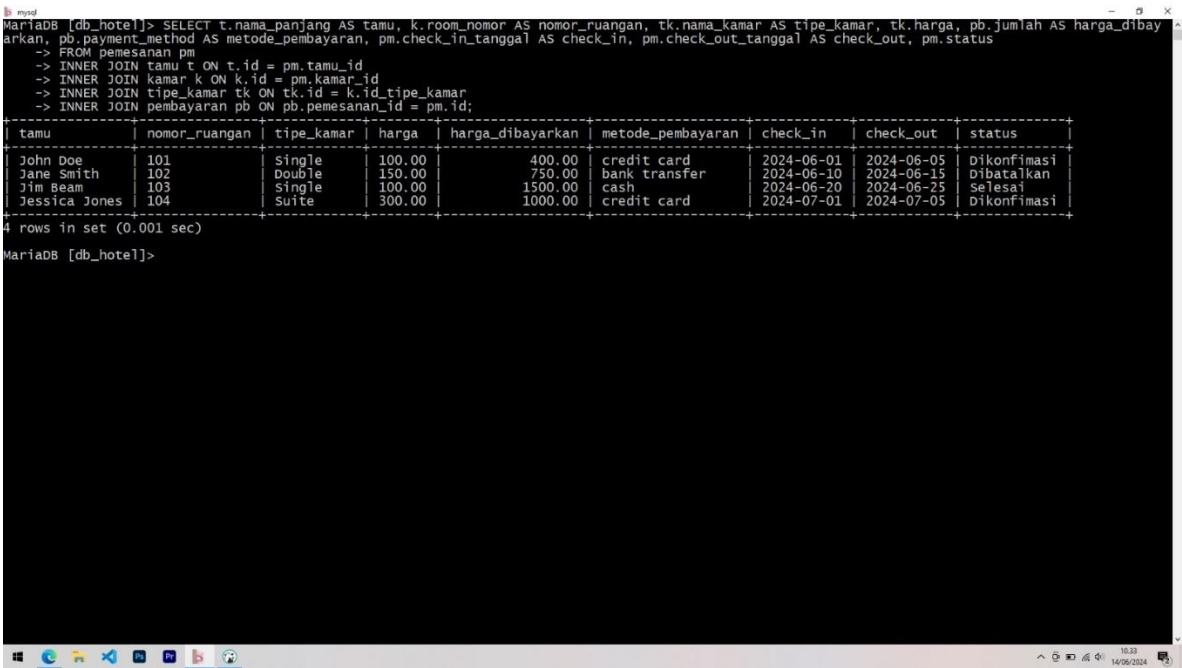


Gambar 2. 16 Join 2 Table dan 4 Table

Alias

Alias digunakan untuk memberikan nama sementara untuk tabel atau kolom, yang dapat mempermudah penulisan query dan meningkatkan keterbacaan

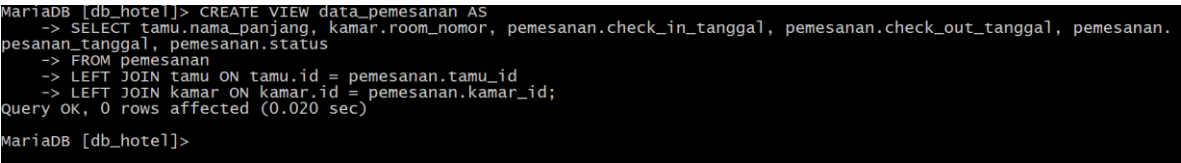




Gambar 2. 17 Alias 2 table dan 4 table

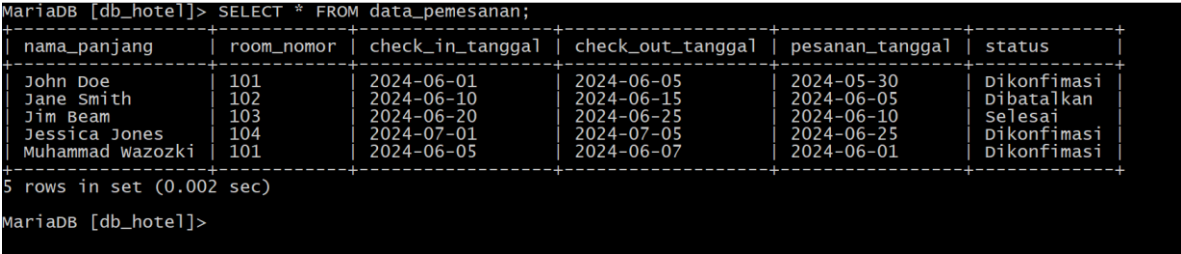
2.3.Query View

View digunakan untuk menyederhanakan akses data yang kompleks dan meningkatkan keamanan dengan membatasi akses langsung ke tabel.



Gambar 2. 18 Query View

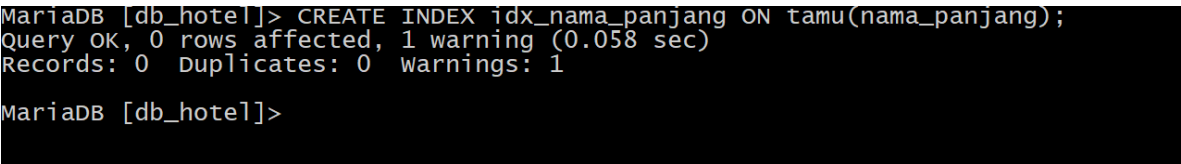
Panggil view berarti mengakses atau menggunakan view yang telah didefinisikan.



Gambar 2. 19 Memanggil View

2.4.Query Index

Index adalah struktur data yang meningkatkan kecepatan pengambilan data dari tabel dengan membuat akses lebih efisien.



Gambar 2. 20 Query Index

2.5.Query Transaction

Transaction adalah satu set operasi basis data yang dilakukan sebagai satu unit kerja yang tidak bisa dipisahkan. Transaksi harus dijalankan sepenuhnya atau tidak sama sekali

```
MariaDB [db_hotel]> START TRANSACTION;
Query OK, 0 rows affected (0.000 sec)

MariaDB [db_hotel]> INSERT INTO tamu(nama_panjang, nomor_telepon, email, alamat) VALUES (
-> 'Ambatukam Zeist', '666-6969', 'ambatukan_ze@gmail.com', 'Homeless');
Query OK, 1 row affected (0.001 sec)
```

Gambar 2. 21 Query Transaction

2.6.Query Commit

Commit adalah perintah untuk menyimpan semua perubahan yang dilakukan dalam transaksi ke basis data secara permanen.

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

MariaDB [db_hotel]> _
```

Gambar 2. 22 Query Commit

2.7.Query Rollback

Rollback adalah perintah untuk membatalkan semua perubahan yang dilakukan dalam transaksi dan mengembalikan basis data ke keadaan sebelum transaksi dimulai.

```
MariaDB [db_hotel]> ROLLBACK TO ambatukam_created;
Query OK, 0 rows affected (0.001 sec)

MariaDB [db_hotel]> ROLLBACK;
Query OK, 0 rows affected (0.007 sec)
```

Gambar 2. 23 Query Rollback

2.8.Query Savepoint

Dengan savepoint, kita bisa mengembalikan transaksi ke kondisi tertentu tanpa harus membatalkan seluruh transaksi.

```
MariaDB [db_hotel]> SAVEPOINT ambatukam_created;
Query OK, 0 rows affected (0.000 sec)

MariaDB [db_hotel]> DELETE FROM tamu WHERE id = 10;
Query OK, 1 row affected (0.001 sec)
```

Gambar 2. 24 Query Savepoint

2.9.Query Procedure

Procedure atau prosedur tersimpan adalah sekumpulan pernyataan SQL yang disimpan dalam basis data dan dapat dijalankan berulang kali. Prosedur digunakan untuk otomatisasi tugas rutin.

```
MariaDB [db_hotel]> CREATE PROCEDURE show_all_table()
-> BEGIN
-> SELECT * FROM pemesanan;
-> SELECT * FROM tamu;
-> SELECT * FROM kamar;
-> SELECT * FROM tipe_kamar;
-> SELECT * FROM pembayaran;
-> SELECT * FROM jasa;
-> SELECT * FROM penggunaan_jasa;
-> END $
Query OK, 0 rows affected (0.019 sec)

MariaDB [db_hotel]>
```

Gambar 2. 25 Query Procedure

2.10.Query Call

Call adalah perintah untuk menjalankan prosedur tersimpan yang telah didefinisikan dalam basis data.

```
MariaDB [db_hotel]> CALL show_all_table();
```

id	tamu_id	kamar_id	check_in_tanggal	check_out_tanggal	pesanan_tanggal	status
1	1	1	2024-06-01	2024-06-05	2024-05-30	Dikonfirmasi
2	2	2	2024-06-10	2024-06-15	2024-06-05	Dibatalkan
3	3	3	2024-06-20	2024-06-25	2024-06-10	Selesai
4	4	4	2024-07-01	2024-07-05	2024-06-25	Dikonfirmasi
5	8	5	2024-06-05	2024-06-07	2024-06-01	Dikonfirmasi

5 rows in set (0.000 sec)

id	nama_panjang	nomor_telepon	email	alamat
1	John Doe	555-1234	john.doe@example.com	123 Main St
2	Jane Smith	555-5678	jane.smith@example.com	456 Elm St
3	Jim Beam	555-8765	jim.beam@example.com	789 Oak St
4	Jessica Jones	555-4321	jessica.jones@example.com	321 Pine St
8	Muhammad Wazozki	555-2323	wazozki@gmail.com	Ohio

5 rows in set (0.010 sec)

id	room_nomor	id_tipe_kamar	status
1	101	1	Tersedia
2	102	2	
3	103	1	Perbaikan
4	104	3	Tersedia
5	101	1	Tidak Tersedia
6	102	2	Tidak Tersedia
7	103	1	Perbaikan
8	104	3	Tersedia

8 rows in set (0.019 sec)

id	nama_kamar	harga
1	Single	100.00
2	Double	150.00
3	Suite	300.00

3 rows in set (0.025 sec)

id	pemesanan_id	jumlah	payment_tanggal	payment_method
13	1	400.00	2024-05-30	credit card
14	2	750.00	2024-06-05	bank transfer
15	3	1500.00	2024-06-10	cash
16	4	1000.00	2024-06-25	credit card

4 rows in set (0.041 sec)

id	nama_jasa	deskripsi	price
----	-----------	-----------	-------

Gambar 2. 26 Query Call

2.11. Query Delimiter

Delimiter adalah karakter atau rangkaian karakter yang digunakan untuk menandai akhir pernyataan SQL. Dalam beberapa sistem basis data, delimiter bisa diubah sementara untuk memungkinkan skrip yang kompleks.

```
MariaDB [db_hotel]> DELIMITER $
MariaDB [db_hotel]> _
```

Gambar 2. 27 Query Delimiter

2.12. Query Trigger

Trigger adalah prosedur yang secara otomatis dijalankan atau oleh suatu peristiwa tertentu dalam tabel, seperti operasi insert, update, atau delete.

```
MariaDB [db_hotel]> CREATE TRIGGER update_status_kamar
-> AFTER INSERT ON pemesanan
-> FOR EACH ROW
-> BEGIN
-> UPDATE kamar SET status = 'Tidak Tersedia' WHERE id = new.kamar_id;
-> END $
Query OK, 0 rows affected (0.047 sec)
MariaDB [db_hotel]>
```

Gambar 2. 28 Query Trigger

2.13. Query Grant

Grant digunakan untuk memberikan izin kepada pengguna untuk melakukan tindakan tertentu.

2.13.1. Grant All Privileges

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON db_hotel.* TO wazozki@localhost
-> IDENTIFIED BY 'wazozki';
Query OK, 0 rows affected (0.009 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]>
```

Gambar 2. 29 Grant All Privileges

```
[~] ~ mysql -u wazozki -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 21
Server version: 11.3.2-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.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| crud_db  |
| db_hotel |
| db_hotel_restored |
| dbpegawai |
| dbpos    |
| dbpost   |
| dbpuskesmas |
| dbtoko   |
| dbtokoti01 |
| elearning |
| information_schema |
| mysql    |
| performance_schema |
| sys      |
+-----+
14 rows in set (0.006 sec)

MariaDB [(none)]>
```

Gambar 2. 30 Bukti user memiliki privileges

2.13.2. Grant SELECT dan DELETE Privileges

```
MariaDB [(none)]> GRANT SELECT, DELETE ON db_hotel.pemesanan TO mahmud@localhost
-> IDENTIFIED BY 'mahmud';
Query OK, 0 rows affected (0.007 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

Gambar 2. 31 Grant SELECT dan DELETE Privileges

```
[~] ~ mysql -u mahmud -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 23
Server version: 11.3.2-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.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| db_hotel |
| information_schema |
+-----+
2 rows in set (0.001 sec)

MariaDB [(none)]> use db_hotel;
Database changed
MariaDB [db_hotel]> show tables;
+-----+
| Tables_in_db_hotel |
+-----+
| pemesanan          |
+-----+
1 row in set (0.001 sec)

MariaDB [db_hotel]> SELECT * FROM pemesanan;
+----+-----+-----+-----+-----+-----+-----+
| id | tamu_id | kamar_id | check_in_tanggal | check_out_tanggal | pesanan_tanggal | status |
+----+-----+-----+-----+-----+-----+-----+
| 1  | 1       | 1        | 2024-06-01       | 2024-06-05       | 2024-05-30      | Dikonfirmasi |
| 2  | 2       | 2        | 2024-06-10       | 2024-06-15       | 2024-06-05      | Dibatalkan |
| 3  | 3       | 3        | 2024-06-20       | 2024-06-25       | 2024-06-10      | Selesai |
| 4  | 4       | 4        | 2024-07-01       | 2024-07-05       | 2024-06-25      | Dikonfirmasi |
| 5  | 8       | 5        | 2024-06-05       | 2024-06-07       | 2024-06-01      | Dikonfirmasi |
+----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

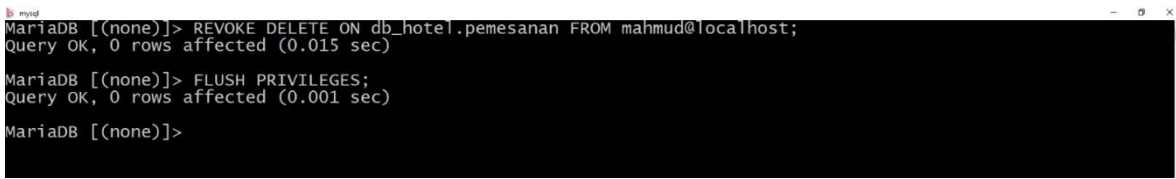
MariaDB [db_hotel]> DELETE FROM pemesanan WHERE id = 5;
Query OK, 1 row affected (0.065 sec)

MariaDB [db_hotel]> UPDATE pemesanan SET tamu_id = 5 WHERE id = 1;
ERROR 1142 (42000): UPDATE command denied to user 'mahmud'@'localhost' for table 'db_hotel`.`pemesanan`
MariaDB [db_hotel]>
```

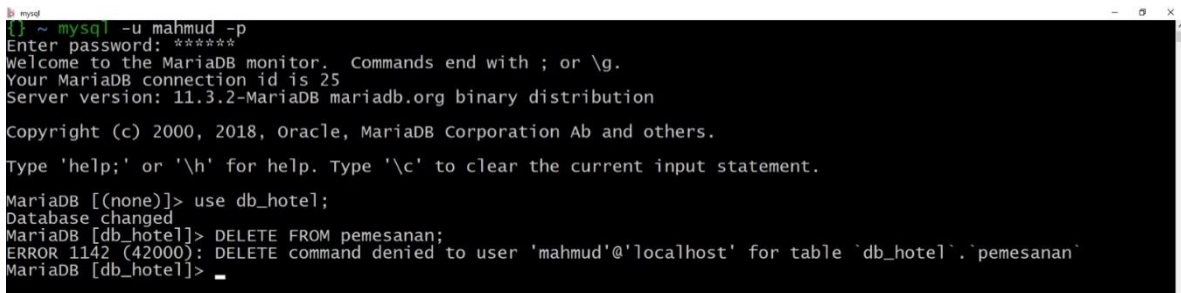
Gambar 2. 32 Bukti user hanya memiliki privileges database hotel pada table pemesanan

2.14. Query Revoke

Revoke digunakan untuk mencabut izin yang telah diberikan sebelumnya.



Gambar 2. 33 Revoke user



Gambar 2. 34 Bukti User Tidak Memiliki Privileges

2.15. Query Backup & Restore

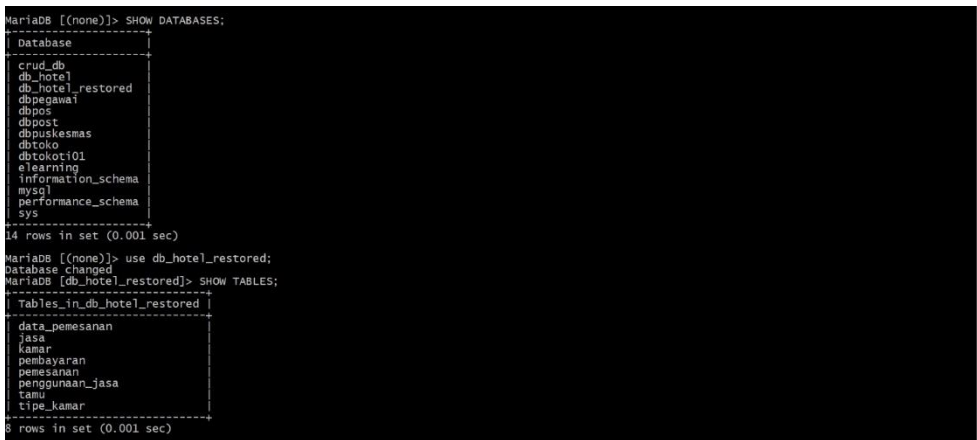
Backup dan restore adalah proses penting dalam manajemen basis data backup untuk tujuan pemulihan jika terjadi kehilangan data atau kerusakan dan restore adalah tindakan mengembalikan data dari backup ke sistem asli atau ke sistem baru.



Gambar 2. 35 Backup database



Gambar 2. 36 Restore Database



Gambar 2. 37 Bukti database direstore

BAB III KESIMPULAN DAN SARAN

3.1. Kesimpulan

Pada penelitian ini, penggunaan query MySQL dan MySQL Workbench telah terbukti sangat efektif dalam pengelolaan dan analisis data. Query MySQL memungkinkan pengguna untuk melakukan operasi data yang kompleks dengan cepat dan efisien. Melalui penggunaan berbagai jenis query. Pengguna dapat mengelola data dalam database dengan lebih fleksibel dan terstruktur. MySQL Workbench, di sisi lain, menawarkan antarmuka grafis yang memudahkan pengguna dalam merancang, mengelola, dan mengadministrasikan database. Fitur-fitur visualisasi dan alat pemodelan yang disediakan oleh MySQL Workbench sangat membantu dalam memahami struktur database dan aliran data, yang pada akhirnya meningkatkan produktivitas dan kualitas hasil kerja.

3.2. Saran

Untuk meningkatkan efisiensi dan keefektifan penggunaan MySQL, disarankan bagi para pengguna untuk terus memperdalam pemahaman mereka mengenai berbagai jenis query dan teknik optimasi yang tersedia. Pelatihan dan praktik yang konsisten akan sangat bermanfaat dalam menguasai berbagai fitur yang ditawarkan oleh MySQL. Selain itu, memanfaatkan MySQL Workbench secara maksimal juga sangat disarankan, terutama dalam hal pemodelan dan visualisasi data. Penggunaan fitur-fitur canggih seperti ERD di MySQL Workbench dapat membantu dalam merancang database yang lebih baik dan memahami hubungan antar tabel dengan lebih jelas. Dengan demikian, kombinasi antara pengetahuan yang mendalam tentang query MySQL dan pemanfaatan penuh MySQL Workbench akan memberikan hasil yang optimal dalam pengelolaan database.