

Laporan Proyek Basis Data
MySQL Database Event
TI04



Anggota Kelompok :

- | | |
|--|-------------------|
| 1. Muhamad Wildan Ramadhan | 0110223132 |
| 2. Raihan Putra Syarif Hidayatulloh | 0110223133 |
| 3. Ahmad Fauzi Ramadhan | 0110223134 |
| 4. Dzakiah Nur Aini | 0110223140 |

Sekolah Tinggi Teknologi Terpadu Nurul Fikri
2024

Kata Pengantar

Pertama-tama kami ucapkan puji dan syukur kepada Allah S.W.T karena dengan karunia dan rahmat-Nya kami dapat menyelesaikan Laporan Proyek Basis Data ini dengan baik dan tepat waktu.

Penyusunan laporan ini ditujukan untuk memenuhi tugas akhir pada mata kuliah Basis Data di Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Ucapan terima kasih kami sampaikan kepada Bapak Nasrul, S.Pdi., S.Kom., M.Kom. selaku dosen dan saudara Daden Dharmawan selaku asisten dosen Mata Kuliah Basis Data yang telah membimbing dalam penyusunan laporan ini.

Semoga laporan ini dapat bermanfaat bagi semua pembaca dan memberikan kontribusi positif bagi perkembangan ilmu pengetahuan di bidang basis data. Terima kasih atas perhatian dan waktu yang telah diluangkan untuk membaca laporan ini. Semoga Allah SWT senantiasa melimpahkan rahmat dan hidayah-Nya kepada kita semua.

Bogor, 7 Juli 2024

Penyusun

Daftar Isi

Kata Pengantar	2
Daftar Isi	3
BAB I	4
Pendahuluan	4
1.1 Latar Belakang	4
1.2 Tujuan	4
BAB II	6
Pembahasan	6
2.1 Desain Database	6
2.2 Nama kolom dan Tipe Data	7
2.2.1 Tabel Jenis Event	8
2.2.2 Tabel Event	8
2.2.3 Tabel Peserta	9
2.2.4 Tabel Staff	9
2.2.5 Tabel Pembayaran	10
2.2.6 Tabel Registrasi	10
2.3 Input Data	10
2.3.1 Input Data Jenis Event	11
2.3.2 Input Data Event	11
2.3.3 Input Data Peserta	11
2.3.4 Input Data Staff	11
2.3.5 Input Data Pembayaran	11
2.3.6 Input Data Registrasi	11
2.4 SQL Inner Join dan Outer Join	11
2.4.1 Inner Join	11
2.4.2 Outer Join	12
2.4.2.1 Left Outer Join	12
2.4.2.2 Right Outer Join	13
2.5 SQL Index dan View	14
2.5.1 Index	14
2.6 SQL Transaction	14
2.6.1 Commit	14
2.6.2 Rollback	15
2.6.3 Savepoint	16
2.7 SQL Procedure	16
2.7.1 Procedure	16
2.8 SQL Trigger	17
2.9 SQL DCL	18
2.10 SQL Backup dan Restore	20
2.10.1 Backup	20
2.10.2 Restore	21
BAB III	23
Penutup	23
3.1 Kesimpulan	23

BAB I

Pendahuluan

1.1 Latar Belakang

Dalam era globalisasi dan digitalisasi yang semakin maju, manajemen data menjadi salah satu aspek yang paling krusial dalam berbagai bidang, termasuk dalam penyelenggaraan event. Event, baik itu seminar, konferensi, konser, atau festival, melibatkan banyak sekali data yang perlu dikelola secara efisien dan akurat. Data tersebut meliputi pendaftaran peserta, jadwal acara, informasi pembicara, alokasi tempat duduk, logistik, dan masih banyak lagi. Oleh karena itu, sebuah sistem database yang handal dan terstruktur menjadi kebutuhan utama dalam penyelenggaraan event yang sukses.

Database merupakan sistem yang memungkinkan penyimpanan, pengorganisasian, dan pengambilan data secara efisien. Penggunaan database dalam manajemen event memudahkan penyelenggara untuk mengakses informasi dengan cepat, melakukan analisis data, dan memastikan bahwa semua aspek dari acara tersebut berjalan dengan lancar. Misalnya, dengan adanya database, penyelenggara dapat dengan mudah memonitor jumlah peserta yang sudah mendaftar, mengirimkan pengingat melalui email, dan memastikan semua kebutuhan logistik tersedia tepat waktu.

Salah satu tools yang sangat berguna dalam mendesain dan mengelola database adalah MySQL Workbench. MySQL Workbench merupakan alat visual yang membantu dalam desain, pengembangan, dan administrasi database. Dengan MySQL Workbench, penyelenggara event dapat membuat desain database yang sesuai dengan kebutuhan mereka, memvisualisasikan hubungan antar tabel, dan melakukan query untuk mendapatkan data yang dibutuhkan.

1.2 Tujuan

Tujuan dari laporan ini adalah untuk memberikan pemahaman mendalam mengenai desain dan implementasi database menggunakan MySQL Workbench.

Laporan ini juga bertujuan untuk menjelaskan berbagai konsep dan perintah SQL yang digunakan dalam pengelolaan database, mulai dari JOIN, INDEX, VIEW, TRANSACTION, PROCEDURE, TRIGGER, DCL, hingga BACKUP dan RESTORE.

BAB II

Pembahasan

2.1 Desain Database

Desain Database adalah proses merancang struktur data dalam sebuah sistem untuk memastikan data tersebut dapat tersimpan, terakses, dan terkelola dengan efisien. MySQL Workbench adalah alat visual yang membantu dalam desain, pengembangan, dan administrasi database MySQL. Dengan MySQL Workbench, pengguna dapat membuat model ER diagram dengan antarmuka drag-and-drop, menetapkan hubungan antar tabel, dan menambahkan constraints. Alat ini juga memungkinkan forward engineering, yaitu menerjemahkan model ER ke perintah SQL untuk menciptakan struktur database dalam MySQL Server, serta reverse engineering untuk mengimpor dan menampilkan struktur database yang sudah ada. Selain itu, MySQL Workbench menyediakan editor SQL untuk menulis dan menjalankan skrip SQL serta alat administrasi untuk mengelola pengguna dan hak akses, backup dan restore, serta monitoring kinerja server. Dengan fitur-fitur tersebut, MySQL Workbench mempermudah proses desain database yang efisien dan terstruktur.

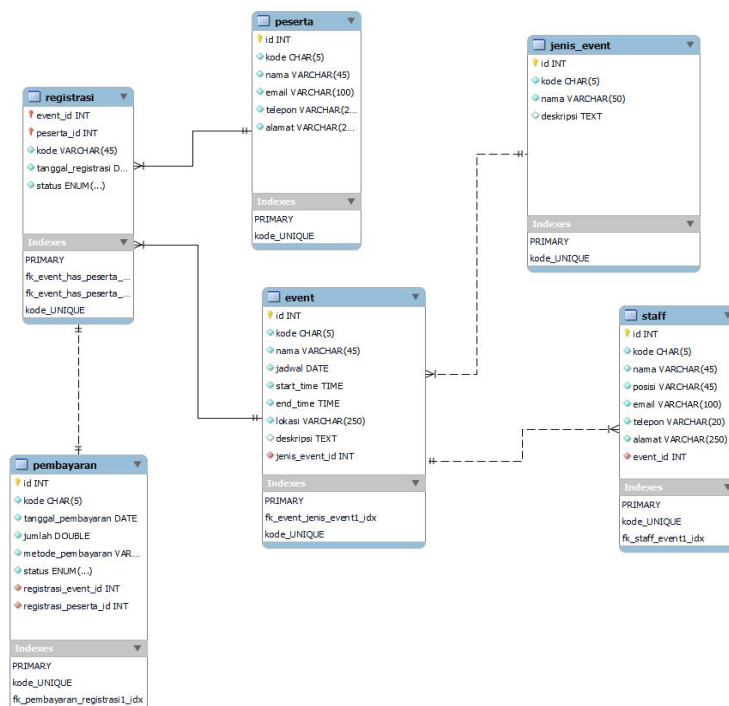
Desain database di MySQL Workbench melibatkan proses pembuatan dan pengelolaan tabel beserta relasi antar tabel untuk memastikan data terstruktur dan terhubung dengan baik. Salah satu aspek penting dalam desain ini adalah penentuan jenis relasi, seperti one-to-many dan many-to-many.

Relasi One-to-Many Relasi one-to-many terjadi ketika satu entitas dapat berhubungan dengan banyak entitas lainnya, tetapi setiap entitas lainnya hanya dapat berhubungan dengan satu entitas tersebut. Misalnya, dalam sebuah sistem akademik, satu dosen dapat mengajar banyak mata kuliah, tetapi setiap mata kuliah hanya diajarkan oleh satu dosen.

Relasi many-to-many terjadi ketika banyak entitas dari satu tabel dapat berhubungan dengan banyak entitas dari tabel lainnya. Misalnya, dalam sebuah sistem perpustakaan, satu buku bisa dipinjam oleh banyak anggota, dan satu anggota bisa meminjam banyak buku. Untuk menangani relasi ini di MySQL Workbench, kita

perlu membuat tabel ketiga (junction table) untuk menyimpan hubungan antar entitas. Jenis relasi yang kami gunakan untuk desain database event adalah sebagai berikut :

1. Tabel event memiliki relasi one-to-many dengan tabel jenis_event. Relasi ini menunjukkan bahwa satu jenis event dapat mencakup banyak event yang berbeda
2. Tabel event juga memiliki relasi one-to-many dengan tabel staff. Relasi ini menggambarkan bahwa dalam satu event, dapat terdapat banyak staff yang terlibat.
3. Tabel event memiliki relasi many-to-many dengan tabel peserta, dan tabel penghubungnya adalah tabel registrasi. Relasi ini mencerminkan bahwa seorang peserta dapat mengikuti beberapa event, dan sebaliknya, sebuah event dapat diikuti oleh banyak peserta. Tabel registrasi berperan sebagai tabel penghubung yang mencatat setiap pendaftaran peserta untuk setiap event yang diikuti.
4. Tabel registrasi memiliki relasi one-to-one dengan tabel pembayaran. Relasi ini menunjukkan bahwa setiap pembayaran hanya dapat dilakukan untuk satu registrasi saja. Artinya, satu transaksi pembayaran berkaitan langsung dengan satu pendaftaran event tertentu.



2.2 Nama kolom dan Tipe Data

2.2.1 Tabel Jenis Event

```
MariaDB [db_event]> desc jenis_event;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode	char(5)	NO	UNI	NULL	
nama	varchar(50)	NO		NULL	
deskripsi	text	YES		NULL	

4 rows in set (0.019 sec)

Tabel Jenis Event berisi daftar jenis-jenis event yang ada, di tabel jenis event terdapat beberapa kolom yaitu :

- id sebagai id jenis event
- kod yang berisikan kode jenis event
- nama yang berisi nama-nama jenis event
- deskripsi untuk menambahkan deskripsi terkait jenis event yang ada.

2.2.2 Tabel Event

```
MariaDB [db_event]> desc event;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode	char(5)	NO	UNI	NULL	
nama	varchar(45)	NO		NULL	
jadwal	date	NO		NULL	
start_time	time	NO		NULL	
end_time	time	NO		NULL	
lokasi	varchar(250)	NO		NULL	
deskripsi	text	YES		NULL	
jenis_event_id	int(11)	NO	MUL	NULL	

9 rows in set (0.019 sec)

Tabel Event berisi detail mengenai event yang ada disini terdapat beberapa kolom yaitu :

- Id berisikan id event yang ada
- Kode berisikan kode event
- Nama berisikan nama event yang akan diselenggarakan
- Jadwal berisikan tanggal jadwal event berlangsung
- Start time berisikan jam waktu event akan dimulai
- End time berisikan jam waktu event selesai
- Lokasi berisikan lokasi tempat diselenggarakannya event
- Deskripsi berisikan informasi tambahan mengenai event

2.2.3 Tabel Peserta

```
MariaDB [db_event]> desc peserta;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode	char(5)	NO	UNI	NULL	
nama	varchar(45)	NO		NULL	
email	varchar(100)	NO		NULL	
telepon	varchar(20)	NO		NULL	
alamat	varchar(250)	NO		NULL	

6 rows in set (0.033 sec)

Tabel Peserta berisikan detail mengenai peserta yang mengikuti, event di tabel ini terdiri dari beberapa baris yaitu:

- Id berisikan id peserta
- Kode berisikan kode peserta yang mengikuti event
- Nama berisikan nama peserta event
- Email berisikan email peserta
- Telepon berisikan telepon peserta
- Alamat berisikan alamat peserta

2.2.4 Tabel Staff

```
MariaDB [db_event]> desc staff;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode	char(5)	NO	UNI	NULL	
nama	varchar(45)	NO		NULL	
posisi	varchar(45)	NO		NULL	
email	varchar(100)	NO		NULL	
telepon	varchar(20)	NO		NULL	
alamat	varchar(250)	NO		NULL	
event_id	int(11)	NO	MUL	NULL	

8 rows in set (0.017 sec)

Tabel Staff berisikan detail mengenai staff yang mengikuti event, di tabel ini terdiri dari beberapa baris yaitu:

- Id berisikan id staff
- Kode berisikan kode staff event
- Nama berisikan nama staff event
- Posisi berisikan posisi staff event
- Email berisikan email staff
- Telepon berisikan telepon staff
- Alamat berisikan alamat staff
- Event id berisikan id event yang dipegang oleh staff

2.2.5 Tabel Pembayaran

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

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode	char(5)	NO	UNI	NULL	
tanggal_pembayaran	date	NO		NULL	
jumlah	double	NO		NULL	
metode_pembayaran	varchar(100)	NO		NULL	
status	enum('Berhasil','Belum Berhasil')	NO		NULL	
registrasi_event_id	int(11)	NO	MUL	NULL	
registrasi_peserta_id	int(11)	NO		NULL	

8 rows in set (0.024 sec)

Tabel Pembayaran berisikan detail mengenai pembayaran event, tabel pembayaran terdiri dari beberapa kolom yaitu :

- Id berisikan id pembayaran
- Kode berisi kode pembayaran
- Tanggal Pembayaran berisikan tanggal melakukan pembayaran
- Jumlah berisikan nominal jumlah pembayaran
- Metode Pembayaran berisikan metode pembayaran yang dilakukan
- Status berisikan status pembayaran 'Berhasil' atau 'Belum Berhasil'
- Registrasi Event Id berelasi ke tabel registrasi kolom event id
- Registrasi Peserta Id berelasi ke tabel registrasi kolom peserta id

2.2.6 Tabel Registrasi

```
MariaDB [db_event]> desc registrasi;
```

Field	Type	Null	Key	Default	Extra
event_id	int(11)	NO	PRI	NULL	
peserta_id	int(11)	NO	PRI	NULL	
kode	varchar(45)	NO	UNI	NULL	
tanggal_registrasi	date	NO		NULL	
status	enum('Pending','Confirmed','Cancelled')	NO		NULL	

5 rows in set (0.023 sec)

Tabel Registrasi berisikan data registrasi peserta yang terdiri dari beberapa kolom yaitu :

- Event Id berisikan id event yang ada
- Peserta Id berisikan id peserta
- Kode berisikan kode registrasi
- Tanggal Registrasi berisikan tanggal registrasi
- Status berisikan status registrasi yaitu 'Pending','Confirmed','Cancelled'.

2.3 Input Data

2.3.1 Input Data Jenis Event

```
MariaDB [db_event]> insert into jenis_event (id,kode,nama,deskripsi) values
-> (6, 'E0006', 'Event Turnamen', '');
Query OK, 1 row affected (0.004 sec)
```

2.3.2 Input Data Event

```
MariaDB [db_event]> insert into event (id,kode,nama,jadwal,start_time,end_time,lokasi,deskripsi,jenis_event_id) values
-> (6, 'EVE006', 'Marathon Lari Puncak', '2024-06-23', '06:00:00', '12:00:00', 'Bogor', '', 6);
Query OK, 1 row affected (0.004 sec)
```

2.3.3 Input Data Peserta

```
MariaDB [db_event]> insert into peserta (id,kode,nama,email,telepon,alamat) values
-> (16, 'PST16', 'Tukul Sugeng', 'tuku@gmail.com', '085873493958', 'Citayam'),
-> (17, 'PST17', 'Bambang Sutrisno', 'bambang@gmail.com', '089454244900', 'Cilebut'),
-> (18, 'PST18', 'Joko Priyanto', 'joko@gmail.com', '086529462863', 'Bojonggede'),
-> (19, 'PST19', 'Om Yono', 'yono@gmail.com', '08926592648', 'Depok'),
-> (20, 'PST20', 'Cang Uding', 'udin@gmail.com', '08942854628', 'Lenteng Agung'),
-> (21, 'PST21', 'Mimin Suhaimin', 'mimin@gmail.com', '+62876938593', 'Parung'),
-> (22, 'PST22', 'cici marcici', 'cici@gmail.com', '+62856037295', 'Citayam');
Query OK, 7 rows affected (0.006 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

2.3.4 Input Data Staff

```
MariaDB [db_event]> insert into staff (id,kode,nama,posisi,email,telepon,alamat,event_id) values
-> (26, 'STF26', 'Asep Galon', 'Leader', 'asep@gmail.com', '0857678910', 'Bojonggede', 6),
-> (27, 'STF27', 'Sutejoe', 'Teknisi', 'sutejoe@gmail.com', '0898073574', 'Cibinong', 6),
-> (28, 'STF28', 'Iyoh Wonderer', 'MC', 'iyoh@gmail.com', '08645365834', 'Tajurhalang', 6),
-> (29, 'STF29', 'Ipeh Maulida', 'Dokumentasi', 'ipeh@gmail.com', '+6287563459', 'Sasak Panjang', 6),
-> (30, 'STF30', 'Ujang Dkaprio', 'Cordinator', 'ujang@gmail.com', '+6286548242', 'Pasir Putih', 6);
Query OK, 5 rows affected (0.007 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

2.3.5 Input Data Pembayaran

```
MariaDB [db_event]> insert into pembayaran (id,kode,tanggal_pembayaran,jumlah,metode_pembayaran,status,registrasi_event_id,registrasi_peserta_id) values
-> (26, 'PYT26', '2024-06-02', 10000, 'Transfer', 'Berhasil', 6, 16),
-> (27, 'PYT27', '2024-06-03', 10000, 'Transfer', 'Berhasil', 6, 17),
-> (28, 'PYT28', '2024-06-03', 10000, 'Transfer', 'Belum Berhasil', 6, 18),
-> (29, 'PYT29', '2024-06-05', 10000, 'Tunai', 'Berhasil', 6, 19),
-> (30, 'PYT30', '2024-06-04', 10000, 'E-Wallet', 'Belum Berhasil', 6, 20),
-> (31, 'PYT31', '2024-06-05', 10000, 'Tunai', 'Belum Berhasil', 6, 21),
-> (32, 'PYT32', '2024-06-07', 10000, 'Transfer', 'Berhasil', 6, 22);
Query OK, 7 rows affected (0.008 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

2.3.6 Input Data Registrasi

```
MariaDB [db_event]> insert into registrasi (event_id,peserta_id,kode,tanggal_registrasi,status) values
-> (6, 16, 'RGR26', '2024-06-01', 'Confirmed'),
-> (6, 17, 'RGR27', '2024-06-02', 'Confirmed'),
-> (6, 18, 'RGR28', '2024-06-03', 'Pending'),
-> (6, 19, 'RGR29', '2024-06-04', 'Confirmed'),
-> (6, 20, 'RGR30', '2024-06-04', 'Cancelled'),
-> (6, 21, 'RGR31', '2024-06-05', 'Pending'),
-> (6, 22, 'RGR32', '2024-06-06', 'Confirmed');
Query OK, 7 rows affected (0.008 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

2.4 SQL Inner Join dan Outer Join

2.4.1 Inner Join

```

MariaDB [db_event]> select
-> event.kode as event_kode,
-> event.nama as event_nama,
-> event.jadwal,
-> event.start_time,
-> event.end_time,
-> event.lokasi,
-> jenis_event.kode as jenis_event_kode,
-> jenis_event.nama as jenis_event_nama
-> from event inner join jenis_event
-> on event.jenis_event_id = jenis_event.id;

```

event_kode	event_nama	jadwal	start_time	end_time	lokasi	jenis_event_kode	jenis_event_nama
EVE02	Technology in Teaching Conference	2024-07-11	08:00:00	13:00:00	Bogor	E0003	Event Pendidikan
EVE03	Bali Trade Exhibition	2024-01-05	08:00:00	13:00:00	Bali	E0001	Event Komersial
EVE04	Hearts for Humanity	2024-06-01	08:00:00	13:00:00	Bekasi	E0002	Event Sosial
EVE05	FutureTech Seminar	2024-01-12	09:00:00	12:00:00	Depok	E0005	Event Seminar
EVE06	Marathon Lari Puncak	2024-06-23	06:00:00	12:00:00	Bogor	E0006	Event Turnamen
EVE01	Sound of The City Festival	2024-02-01	20:00:00	23:00:00	Jakarta	E0004	Event Hiburan

6 rows in set (0.001 sec)

Pengertian: Inner Join adalah jenis join dalam SQL yang mengembalikan baris yang memiliki nilai yang cocok di kedua tabel yang bergabung.

Tujuan: Untuk menggabungkan data dari dua tabel berdasarkan kondisi yang sesuai, hanya menampilkan data yang memiliki hubungan di kedua tabel.

Penjelasan: Inner join disini berfungsi untuk menggabungkan/menjoinkan kolom kode dan nama dari table jenis event ke table event. sehingga kita dapat melihat detail event, mulai dari nama event, jadwal, waktu pelaksanaannya, lokasi dan juga jenis eventnya.

2.4.2 Outer Join

2.4.2.1 Left Outer Join

```

MariaDB [db_event]> select
-> event.kode as event_kode,
-> event.nama as event_nama,
-> event.jadwal,
-> event.lokasi,
-> staff.kode as staff_kode,
-> staff.nama as staff_nama,
-> staff.posisi,
-> staff.email,
-> staff.telepon
-> from event left outer join staff
-> on event.id = staff.event_id;

```

event_kode	event_nama	jadwal	lokasi	staff_kode	staff_nama	posisi	email	telepon
EVE02	Technology in Teaching Conference	2024-07-11	Bogor	STF01	Dimas Gepenk	Cordinator	Dimas@gmail.com	089548658
EVE02	Technology in Teaching Conference	2024-07-11	Bogor	STF02	Hasan Anas	MC	Hasan@gmail.com	0897354278
EVE02	Technology in Teaching Conference	2024-07-11	Bogor	STF03	Asep Puty	Teknisi	Asep@gmail.com	08964757294
EVE02	Technology in Teaching Conference	2024-07-11	Bogor	STF04	Amara Puput	Dokumentasi	Puput@gmail.com	0897275643
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF05	Voga Kurniawan	Leader	Voga@gmail.com	0894528614
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF06	Abdul Wafa	Teknisi	Wafa@gmail.com	0896512843
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF07	Putri	MC	Putri@gmail.com	0897491628
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF08	Doni	Leader	Doni@gmail.com	0893881923
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF09	Zaki	Dokumentasi	Zaki@gmail.com	0892147246
EVE03	Bali Trade Exhibition	2024-01-05	Bali	STF10	Nanda	Teknisi	Nanda@gmail.com	0892315623
EVE04	Hearts for Humanity	2024-06-01	Bekasi	STF11	Farhan	Cordinator	Farhan@gmail.com	0891362283
EVE04	Hearts for Humanity	2024-06-01	Bekasi	STF12	Ahmad Raja	Teknisi	Raja@gmail.com	0892231456
EVE04	Hearts for Humanity	2024-06-01	Bekasi	STF13	Resti	MC	Resti@gmail.com	089113425
EVE04	Hearts for Humanity	2024-06-01	Bekasi	STF14	Fatur	Dokumentasi	Fatur@gmail.com	0899918245
EVE04	Hearts for Humanity	2024-06-01	Bekasi	STF15	Jojon	Leader	Jojon@gmail.com	0897426671
EVE05	FutureTech Seminar	2024-01-12	Depok	STF16	Gojo	Cordinator	Gojo@gmail.com	0892447182
EVE05	FutureTech Seminar	2024-01-12	Depok	STF17	Milim	MC	Milim@gmail.com	0897728193
EVE05	FutureTech Seminar	2024-01-12	Depok	STF18	Dilan	Leader	Dilan@gmail.com	0892148261
EVE05	FutureTech Seminar	2024-01-12	Depok	STF19	Toni	Teknisi	Toni@gmail.com	0892135102
EVE05	FutureTech Seminar	2024-01-12	Depok	STF20	Bili	Dokumentasi	Bili@gmail.com	0894452819
EVE06	Marathon Lari Puncak	2024-06-23	Bogor	STF26	Asep Galon	Leader	asep@gmail.com	0895678910
EVE06	Marathon Lari Puncak	2024-06-23	Bogor	STF27	Sutejoe	Teknisi	sutejoe@gmail.com	0898073574
EVE06	Marathon Lari Puncak	2024-06-23	Bogor	STF28	Iyoh Wonderer	MC	iyoh@gmail.com	08645365834
EVE06	Marathon Lari Puncak	2024-06-23	Bogor	STF29	Ipeh Maulida	Dokumentasi	ipeh@gmail.com	+6287563459
EVE06	Marathon Lari Puncak	2024-06-23	Bogor	STF30	Ujang Dkaprio	Cordinator	ujang@gmail.com	+6286548242
EVE01	Sound of The City Festival	2024-02-01	Jakarta	STF21	Fadil	Teknisi	Fadil@gmail.com	089997482
EVE01	Sound of The City Festival	2024-02-01	Jakarta	STF22	Additya	Dokumentasi	Addit@gmail.com	088821461
EVE01	Sound of The City Festival	2024-02-01	Jakarta	STF23	Indri	MC	Indri@gmail.com	0892177432
EVE01	Sound of The City Festival	2024-02-01	Jakarta	STF24	Alucard	Leader	Alucard@gmail.com	0891156382
EVE01	Sound of The City Festival	2024-02-01	Jakarta	STF25	Fauzan	Cordinator	Fauzan@gmail.com	0891987336

30 rows in set (0.003 sec)

Pengertian: Left Outer Join mengembalikan semua baris dari tabel kiri (left table) dan baris yang cocok dari tabel kanan. Jika tidak ada kecocokan, hasilnya tetap

menampilkan semua baris dari tabel kiri dengan nilai NULL untuk kolom dari tabel kanan.

Tujuan: Untuk menampilkan semua data dari tabel kiri dan data yang cocok dari tabel kanan, termasuk baris dari tabel kiri yang tidak memiliki kecocokan.

Penjelasan: Jenis join dalam SQL yang menggabungkan baris dari dua tabel berdasarkan kondisi tertentu. Join ini akan mengembalikan semua baris dari tabel sebelah kiri, serta baris yang cocok dari tabel sebelah kanan. Jika tidak ada kecocokan, kolom dari tabel kanan akan berisi nilai NULL.

Penjelasan: disini tabel pertama yaitu event dan table kedua yaitu jenis_event

2.4.2.2 Right Outer Join

```

MariaDB [db_event]> select
-> staff.kode as staff_kode,
-> staff.nama as staff_nama,
-> staff.posisi,
-> staff.email,
-> staff.telepon,
-> event.kode as event_kode,
-> event.nama as event_nama,
-> event.jadwal,
-> event.lokasi
-> from staff right outer join event
-> on staff.event_id = event.id;

```

staff_kode	staff_nama	posisi	email	telepon	event_kode	event_nama	jadwal	lokasi
STF01	Dimas Gepenk	Cordinator	Dimas@gmail.com	089548658	EVE02	Technology in Teaching Conference	2024-07-11	Bogor
STF02	Hasan Anas	MC	Hasan@gmail.com	0897354278	EVE02	Technology in Teaching Conference	2024-07-11	Bogor
STF03	Asep Puty	Teknisi	Asep@gmail.com	08964757294	EVE02	Technology in Teaching Conference	2024-07-11	Bogor
STF04	Amara Puput	Dokumentasi	Puput@gmail.com	0892725643	EVE02	Technology in Teaching Conference	2024-07-11	Bogor
STF05	Yoga Kurniawan	Leader	Yoga@gmail.com	0894528614	EVE02	Technology in Teaching Conference	2024-07-11	Bogor
STF06	Abdul Wafa	Teknisi	Wafa@gmail.com	0896512843	EVE03	Bali Trade Exhibition	2024-01-05	Bali
STF07	Putri	MC	Putri@gmail.com	0897491628	EVE03	Bali Trade Exhibition	2024-01-05	Bali
STF08	Doni	Leader	Doni@gmail.com	0893881923	EVE03	Bali Trade Exhibition	2024-01-05	Bali
STF09	Zaki	Dokumentasi	Zaki@gmail.com	0892147246	EVE03	Bali Trade Exhibition	2024-01-05	Bali
STF10	Nanda	Teknisi	Nanda@gmail.com	0892315623	EVE03	Bali Trade Exhibition	2024-01-05	Bali
STF11	Farhan	Cordinator	Farhan@gmail.com	0891362283	EVE04	Hearts for Humanity	2024-06-01	Bekasi
STF12	Ahmad Raja	Teknisi	Raja@gmail.com	0892231456	EVE04	Hearts for Humanity	2024-06-01	Bekasi
STF13	Resti	MC	Resti@gmail.com	089113425	EVE04	Hearts for Humanity	2024-06-01	Bekasi
STF14	Fatur	Dokumentasi	Fatur@gmail.com	0899910245	EVE04	Hearts for Humanity	2024-06-01	Bekasi
STF15	Jojon	Leader	Leader@gmail.com	0897426671	EVE04	Hearts for Humanity	2024-06-01	Bekasi
STF16	Gojo	Cordinator	Gojo@gmail.com	0892447182	EVE05	FutureTech Seminar	2024-01-12	Depok
STF17	Milim	MC	Milim@gmail.com	0897728193	EVE05	FutureTech Seminar	2024-01-12	Depok
STF18	Dilan	Leader	Dilan@gmail.com	0892148261	EVE05	FutureTech Seminar	2024-01-12	Depok
STF19	Toni	Teknisi	Toni@gmail.com	0892135102	EVE05	FutureTech Seminar	2024-01-12	Depok
STF20	Bili	Dokumentasi	Bili@gmail.com	0894452819	EVE05	FutureTech Seminar	2024-01-12	Depok
STF26	Asep Galon	Leader	asep@gmail.com	0857678910	EVE06	Marathon Lari Puncak	2024-06-23	Bogor
STF27	Sutejoe	Teknisi	sutejoe@gmail.com	0898073574	EVE06	Marathon Lari Puncak	2024-06-23	Bogor
STF28	Iyoh Wonderer	MC	iyoh@gmail.com	08645365834	EVE06	Marathon Lari Puncak	2024-06-23	Bogor
STF29	Ipeh Maulida	Dokumentasi	ipeh@gmail.com	+6287563459	EVE06	Marathon Lari Puncak	2024-06-23	Bogor
STF30	Ujang Dkaprio	Cordinator	ujang@gmail.com	+6286548242	EVE06	Marathon Lari Puncak	2024-06-23	Bogor
STF21	Fadil	Teknisi	Fadil@gmail.com	089997482	EVE01	Sound of The City Festival	2024-02-01	Jakarta
STF22	Aditya	Dokumentasi	Adit@gmail.com	088821461	EVE01	Sound of The City Festival	2024-02-01	Jakarta
STF23	Indri	MC	Indri@gmail.com	0892177432	EVE01	Sound of The City Festival	2024-02-01	Jakarta
STF24	Alucard	Leader	Alucard@gmail.com	0891156382	EVE01	Sound of The City Festival	2024-02-01	Jakarta
STF25	Fauzan	Cordinator	Fauzan@gmail.com	0891987336	EVE01	Sound of The City Festival	2024-02-01	Jakarta

30 rows in set (0.001 sec)

Pengertian: Right Outer Join mengembalikan semua baris dari tabel kanan (right table) dan baris yang cocok dari tabel kiri. Jika tidak ada kecocokan, hasilnya tetap menampilkan semua baris dari tabel kanan dengan nilai NULL untuk kolom dari tabel kiri.

Tujuan: Untuk menampilkan semua data dari tabel kanan dan data yang cocok dari tabel kiri, termasuk baris dari tabel kanan yang tidak memiliki kecocokan.

Penjelasan: mengambil data dari tabel staff yaitu data kode ditampilkan dengan nama staff_kode, data nama ditampilkan dengan nama staff_nama, lalu ada data posisi, email dan telepon. Mengambil data di tabel event yaitu kode ditampilkan dengan nama event_kode, data nama ditampilkan dengan nama event_nama, data jadwal dan lokasi. Data saling berelasi di tabel staff kolom event_id ke tabel event kolom id.

2.5 SQL Index dan View

2.5.1 Index

```
MariaDB [db_event]> create index idx_nama_event on event (nama);
Query OK, 0 rows affected (0.024 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Pengertian: Index adalah struktur data dalam basis data yang meningkatkan kecepatan operasi pengambilan data.

Tujuan: Untuk mempercepat proses pencarian dan pengambilan data dalam tabel.

Membuat index untuk kolom nama event yang ada di tabel event.

Penjelasan: Membuat index untuk kolom nama event di tabel event.

2.5.2 View

```
MariaDB [db_event]> create view detail_event as select
-> event.kode as event_kode,
-> event.nama as event_nama,
-> event.jadwal,
-> event.start_time,
-> event.end_time,
-> event.lokasi,
-> jenis_event.kode as jenis_event_kode,
-> jenis_event.nama as jenis_event_nama
-> from event inner join jenis_event
-> on event.jenis_event_id = jenis_event.id;
Query OK, 0 rows affected (0.006 sec)
```

```
MariaDB [db_event]> select * from detail_event;
```

event_kode	event_nama	jadwal	start_time	end_time	lokasi	jenis_event_kode	jenis_event_nama
EVE02	Technology in Teaching Conference	2024-07-11	08:00:00	13:00:00	Bogor	E0003	Event Pendidikan
EVE03	Bali Trade Exhibition	2024-01-05	08:00:00	13:00:00	Bali	E0001	Event Komersial
EVE04	Hearts for Humanity	2024-06-01	08:00:00	13:00:00	Bekasi	E0002	Event Sosial
EVE05	FutureTech Seminar	2024-01-12	09:00:00	12:00:00	Depok	E0005	Event Seminar
EVE06	Marathon Lari Puncak	2024-06-23	06:00:00	12:00:00	Bogor	E0006	Event Turnamen
EVE07	Lomba Balap Karung	2024-06-23	08:00:00	11:00:00	Cibinong	E0006	Event Turnamen
EVE01	Sound of The City Festival	2024-02-01	20:00:00	23:00:00	Jakarta	E0004	Event Hiburan
EVE10	Turnamen PES mobile	2024-06-24	09:00:00	12:00:00	Depok	E0006	Event Turnamen
EVE11	Turnamen PUBG Mobile	2024-06-24	09:00:00	12:00:00	Depok	E0006	Event Turnamen

9 rows in set (0.001 sec)

Pengertian: View adalah tabel virtual yang didasarkan pada hasil query SQL. View tidak menyimpan data fisik.

Tujuan: Untuk menyederhanakan akses data, menyembunyikan kompleksitas query, dan meningkatkan keamanan data.

Penjelasan: membuat view untuk menampilkan detail_event dengan mengambil beberapa kolom event yaitu kolom kode,nama,jadwal,start_time,end_time, dan lokasi. dan mengambil beberapa kolom jenis_event yaitu kode dan nama, lalu kita dapat melihat hasilnya dengan menjalankan query select*from detail_event.

2.6 SQL Transaction

2.6.1 Commit

```

MariaDB [db_event]> start transaction;
Query OK, 0 rows affected (0.001 sec)

MariaDB [db_event]> insert into event (id,kode,nama,jadwal,start_time,end_time,lokasi,deskripsi,jenis_event_id) values
-> (7, 'EVE07', 'Lomba Balap Karung', '2024-06-23', '08:00:00', '11:00:00', 'Cibinong', '', 6);
Query OK, 1 row affected (0.002 sec)

MariaDB [db_event]> commit;
Query OK, 0 rows affected (0.005 sec)

```

Pengertian: Commit adalah perintah SQL yang digunakan untuk menyimpan semua perubahan yang telah dilakukan dalam transaksi.

Tujuan: Untuk memastikan bahwa semua perubahan dalam transaksi disimpan secara permanen dalam basis data.

Penjelasan: Membuat commit untuk menambahkan data pada table event. Pernyataan START TRANSACTION digunakan untuk memulai transaksi baru. Hal ini memastikan bahwa semua operasi dalam transaksi diselesaikan dengan sukses sebelum perubahan apa pun diterapkan ke basis data. Jika ada bagian dari transaksi yang gagal, semua perubahan yang dilakukan selama transaksi dapat dibatalkan, sehingga basis data kembali ke keadaan sebelumnya.

2.6.2 Rollback

```

MariaDB [db_event]> start transaction;
Query OK, 0 rows affected (0.000 sec)

MariaDB [db_event]> DELETE FROM staff WHERE id = 4;
Query OK, 1 row affected (0.003 sec)

MariaDB [db_event]> rollback;
Query OK, 0 rows affected (0.004 sec)

```

Pengertian: Rollback adalah perintah SQL yang digunakan untuk membatalkan semua perubahan yang telah dilakukan dalam transaksi.

Tujuan: Untuk mengembalikan basis data ke kondisi sebelum transaksi dimulai, dalam kasus kesalahan atau kebutuhan untuk membatalkan perubahan.

Penjelasan: Menghapus salah satu data tabel staff lalu rollback dilakukan agar data bisa kembali seperti semula. Pernyataan START TRANSACTION digunakan untuk memulai transaksi baru. Hal ini memastikan bahwa semua operasi dalam transaksi diselesaikan dengan sukses sebelum perubahan apa pun diterapkan ke basis data. Jika ada bagian dari transaksi yang gagal, semua perubahan yang dilakukan selama transaksi dapat dibatalkan, sehingga basis data kembali ke keadaan sebelumnya.

2.6.3 Savepoint

```
MariaDB [db_event]> start transaction;
Query OK, 0 rows affected (0.000 sec)

MariaDB [db_event]> savepoint update_staff;
Query OK, 0 rows affected (0.001 sec)

MariaDB [db_event]> update staff set posisi = 'Dokumentasi' where id = 2;
Query OK, 1 row affected (0.003 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [db_event]> rollback to update_staff;
Query OK, 0 rows affected (0.002 sec)
```

Pengertian: Savepoint adalah perintah SQL yang digunakan untuk menetapkan titik penyimpanan sementara dalam transaksi.

Tujuan: Untuk memungkinkan rollback sebagian dari transaksi ke titik penyimpanan tertentu tanpa mempengaruhi seluruh transaksi.

Penjelasan: Membuat savepoint update_staff di tabel staff lalu melakukan update pada salah satu staff, dan rollback ke savepoint update_staff.

2.7 SQL Procedure

2.7.1 Procedure


```

MariaDB [db_event]> delimiter //
MariaDB [db_event]> create procedure showEvent()
-> begin
-> select kode, nama, jadwal, start_time, end_time, lokasi
-> from event;
-> end
-> //
Query OK, 0 rows affected (0.024 sec)

MariaDB [db_event]> delimiter ;
MariaDB [db_event]> call showEvent();
+-----+-----+-----+-----+-----+-----+
| kode | nama | jadwal | start_time | end_time | lokasi |
+-----+-----+-----+-----+-----+-----+
| EVE02 | Technology in Teaching Conference | 2024-07-11 | 08:00:00 | 13:00:00 | Bogor |
| EVE03 | Bali Trade Exhibition | 2024-01-05 | 08:00:00 | 13:00:00 | Bali |
| EVE04 | Hearts for Humanity | 2024-06-01 | 08:00:00 | 13:00:00 | Bekasi |
| EVE05 | FutureTech Seminar | 2024-01-12 | 09:00:00 | 12:00:00 | Depok |
| EVE06 | Marathon Lari Puncak | 2024-06-23 | 06:00:00 | 12:00:00 | Bogor |
| EVE07 | Lomba Balap Karung | 2024-06-23 | 08:00:00 | 11:00:00 | Cibinong |
| EVE01 | Sound of The City Festival | 2024-02-01 | 20:00:00 | 23:00:00 | Jakarta |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

Query OK, 0 rows affected (0.062 sec)

```

Pengertian: Procedure adalah kumpulan pernyataan SQL yang disimpan dan dapat dijalankan sebagai unit tunggal.

Tujuan: Untuk mengotomatisasi tugas rutin, meningkatkan efisiensi, dan memastikan konsistensi operasi dalam basis data.

Penjelasan: Query delimiter digunakan untuk mengubah pembatas perintah yaitu ; karena akan membuat sebuah procedure, kita akan mengganti delimiter menjadi //, Membuat procedure dengan menggunakan query create procedure, Disini kita membuat procedure untuk melihat kolom kode, nama, jadwal, start_time, end_time Lokasi dari table event.

2.8 SQL Trigger

```

MariaDB [db_event]> delimiter //
MariaDB [db_event]> create or replace trigger hitungDurasi
-> before insert on event
-> for each row
-> begin
-> set new.durasi = timediff(new.end_time, new.start_time);
-> end//
Query OK, 0 rows affected (0.008 sec)

```

```

MariaDB [db_event]> alter table event add column durasi TIME;
Query OK, 0 rows affected (0.038 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

```

MariaDB [db_event]> insert into event (id,kode,nama,jadwal,start_time,end_time,lokasi,deskripsi,jenis_event_id) values
-> (11, 'EVE11', 'Turnamen PUBG Mobile', '2024-06-24', '09:00:00', '12:00:00', 'Depok', '', 6);
Query OK, 1 row affected (0.004 sec)

MariaDB [db_event]> select id, kode, nama, start_time, end_time, durasi
-> from event where id = 11;
+-----+-----+-----+-----+-----+-----+
| id | kode | nama                | start_time | end_time | durasi |
+-----+-----+-----+-----+-----+-----+
| 11 | EVE11 | Turnamen PUBG Mobile | 09:00:00   | 12:00:00 | 03:00:00 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

```

Pengertian: Trigger adalah prosedur yang dijalankan secara otomatis sebagai respons terhadap peristiwa tertentu dalam tabel atau view, seperti insert, update, atau delete.

Tujuan: Untuk memastikan integritas data, otomatisasi tugas, dan menegakkan aturan bisnis dalam basis data.

Penjelasan: Menambahkan kolom durasi untuk membuat trigger durasi pada table event dengan menggunakan query alter table, Query delimiter digunakan untuk mengubah pembatas perintah yaitu ; karena akan membuat sebuah trigger, kita akan mengganti delimiter menjadi //, Membuat trigger untuk durasi yg data baru masuk akan dapat terlihat durasi sedangkan yang data sebelum adanya trigger akan null karena trigger-nya menggunakan query before insert, Timediff digunakan untuk mengurangi waktu selesai event dengan waktu memulai event sehingga terlihat durasi waktu yang ada pada event, Hasilnya bisa liat menggunakan query select

2.9 SQL DCL

```

MariaDB [(none)]> GRANT ALL PRIVILEGES ON db_event .* TO ean@localhost IDENTIFIED BY '123';
Query OK, 0 rows affected (0.004 sec)

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

```

```

MariaDB [(none)]> GRANT SELECT,DELETE ON db_event.staff TO putra@localhost IDENTIFIED BY '123';
Query OK, 0 rows affected (0.005 sec)

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

```

```

MariaDB [(none)]> revoke delete on db_event.staff from putra@localhost;
Query OK, 0 rows affected (0.005 sec)

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

```

```

Setting environment for using XAMPP for Windows.
raiha@MSI c:\xampp
# mysql -u ean db_event -p
Enter password: ***
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 23
Server version: 10.4.28-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 [db_event]> show tables;
+-----+
| Tables_in_db_event |
+-----+
| detail_event       |
| event              |
| jenis_event        |
| pembayaran        |
| peserta            |
| registrasi         |
| staff              |
+-----+
7 rows in set (0.001 sec)

```

```

MariaDB [db_event]> update staff set posisi = 'Leader' where id = 1;
ERROR 1142 (42000): UPDATE command denied to user 'putra'@'localhost' for table 'db_event`.`staff`
MariaDB [db_event]> |

```

```

# mysql -u putra db_event -p
Enter password: ***
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 26
Server version: 10.4.28-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 [db_event]> show tables;
+-----+
| Tables_in_db_event |
+-----+
| staff              |
+-----+
1 row in set (0.001 sec)

MariaDB [db_event]> select * from staff limit 5;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id | kode | nama          | posisi    | email          | telepon    | alamat    | event_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  | STF01 | Dimas Gepenk  | Cordinator | Dimas@gmail.com | 089548658  | Bogor     | 2         |
| 2  | STF02 | Hasan Anas    | MC         | Hasan@gmail.com | 0897354278 | Bogor     | 2         |
| 3  | STF03 | Asep Puty     | Dokumentasi | Asep@gmail.com  | 08964757294 | Bogor     | 2         |
| 4  | STF04 | Amara Puput   | Dokumentasi | Puput@gmail.com | 0892725643 | Bogor     | 2         |
| 5  | STF05 | Yoga Kurniawan | Leader     | Yoga@gmail.com  | 0894528614 | Bogor     | 2         |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [db_event]> delete from staff where id = 4;
Query OK, 1 row affected (0.008 sec)

```

```
MariaDB [db_event]> select * from staff limit 5;
```

id	kode	nama	posisi	email	telepon	alamat	event_id
1	STF01	Dimas Gepenk	Cordinator	Dimas@gmail.com	089548658	Bogor	2
2	STF02	Hasan Anas	MC	Hasan@gmail.com	0897354278	Bogor	2
3	STF03	Asep Puty	Dokumentasi	Asep@gmail.com	08964757294	Bogor	2
5	STF05	Yoga Kurniawan	Leader	Yoga@gmail.com	0894528614	Bogor	2
6	STF06	Abdul Wafa	Teknisi	Wafa@gmail.com	0896512843	Bali	3

```
5 rows in set (0.001 sec)

MariaDB [db_event]> delete from staff where id = 6;
ERROR 1142 (42000): DELETE command denied to user 'putra'@'localhost' for table 'db_event`.`staff`
MariaDB [db_event]>
```

Pengertian: DCL adalah bagian dari SQL yang berhubungan dengan pengendalian akses dan izin pengguna dalam basis data. Contoh perintah DCL adalah GRANT dan REVOKE.

Tujuan: Untuk mengatur hak akses pengguna dan mengendalikan siapa yang dapat melakukan operasi tertentu pada basis data.

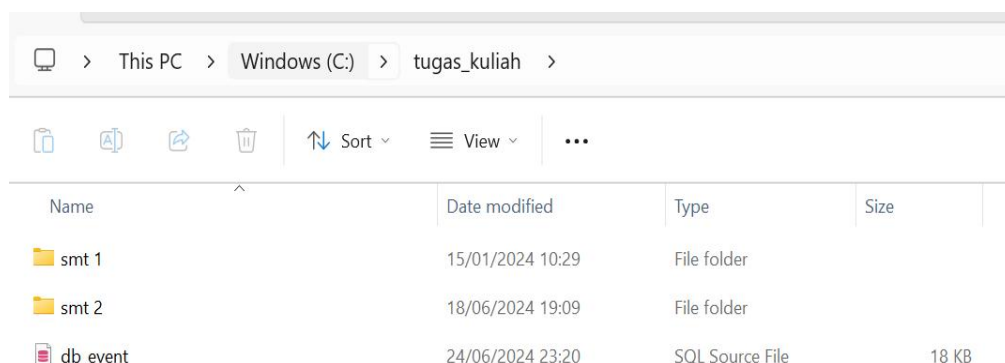
Penjelasan: Membuat akses penuh dengan menggunakan grant all privileges on nama_database yang berarti memberi akses penuh dalam database tersebut ke ean@localhost dengan identify(password) '123', lalu membuat akses hanya Sebagian dengan menggunakan grant select, delete on nama_database yang berarti hanya bisa melihat dan menghapus pada database tersebut. Query revoke digunakan untuk ntuk mencabut atau menghapus hak akses atau izin yang telah diberikan kepada pengguna atau peran disini kita mencabut untuk akses delete. Query flush privileges digunakan untuk memuat ulang tabel hak akses di MySQL yang disimpan dalam database

2.10 SQL Backup dan Restore

2.10.1 Backup

```
raiha@MSI c:\xampp
# mysqldump -u root -p db_event > C:/tugas_kuliah/db_event.sql
Enter password:

raiha@MSI c:\xampp
#
```



Name	Date modified	Type	Size
smt 1	15/01/2024 10:29	File folder	
smt 2	18/06/2024 19:09	File folder	
db_event	24/06/2024 23:20	SQL Source File	18 KB

Pengertian: Backup adalah proses membuat salinan data dari database untuk melindungi data dari kehilangan atau kerusakan.

Tujuan: Melindungi data dari kehilangan dan membantu dalam memindahkan data antar sistem.

Penjelasan: backup database dilakukan dengan memasukkan query mysqldump -u root -p db_event (database yang ingin kita backup) > C:/tugas_kuliah/db_event.sql (tempat untuk menyimpan backup database), lalu kita dapat melihat backupan di file yang ada di query.

2.10.2 Restore

```
MariaDB [(none)]> create database db_event_restored;  
Query OK, 1 row affected (0.005 sec)
```

```
MariaDB [(none)]> Bye
```

```
raiha@MSI c:\xampp  
# mysql -u root -p db_event_restored < C:/tugas_kuliah/db_event.sql  
Enter password:
```

```
raiha@MSI c:\xampp  
# mysql -u root db_event_restored  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 33  
Server version: 10.4.28-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 [db_event_restored]> show tables;
```

```
+-----+  
| Tables_in_db_event_restored |  
+-----+  
| detail_event  
| event  
| jenis_event  
| pembayaran  
| peserta  
| registrasi  
| staff  
+-----+  
7 rows in set (0.001 sec)
```

Pengertian: Restore adalah proses mengembalikan data dari backup ke database untuk memulihkan data yang hilang atau rusak.

Tujuan: Mengembalikan data yang hilang, memulihkan operasi sistem setelah kegagalan, memastikan backup dapat digunakan.

Penjelasan: Restore database dapat dilakukan dengan memasukan query create database db_event_restored, lalu mysql -u root -p db_event_restored (database yang ingin kita backup) > C:/tugas_kuliah/db_event.sql . dan kita dapat melihat table database yang sudah di restore dengan menjalankan query show tables.

BAB III

Penutup

3.1 Kesimpulan

Laporan ini merangkum berbagai aspek penting dalam desain dan pengelolaan database menggunakan MySQL Workbench, serta implementasi konsep-konsep SQL seperti join, indeks, view, transaksi, prosedur, trigger, dan DCL. MySQL Workbench adalah alat yang sangat berguna untuk merancang database secara visual. Desain database yang baik memastikan bahwa data tersimpan, terakses, dan terkelola dengan efisien, dengan relasi one-to-many dan many-to-many yang diterapkan sesuai kebutuhan aplikasi. Struktur tabel yang jelas dengan kolom dan tipe data yang tepat sangat penting untuk menyimpan data dengan cara yang terorganisir. Proses input data harus mengikuti desain tabel yang sudah dibuat untuk memastikan konsistensi dan integritas data.

Penggunaan join seperti inner join dan outer join (left dan right) memungkinkan penggabungan data dari beberapa tabel berdasarkan kondisi tertentu, sehingga pengambilan data menjadi lebih komprehensif. Indeks membantu meningkatkan kecepatan pengambilan data dalam tabel, sementara view menyederhanakan akses data dan meningkatkan keamanan dengan menyembunyikan kompleksitas query. Transaksi SQL, termasuk commit, rollback, dan savepoint, memastikan bahwa operasi data dilakukan secara atomik dan konsisten. Commit menyimpan perubahan secara permanen, rollback membatalkan perubahan, dan savepoint memungkinkan rollback parsial. Prosedur menyimpan kumpulan pernyataan SQL yang dapat dijalankan sebagai satu unit, membantu dalam otomatisasi tugas dan memastikan konsistensi operasi. Trigger menjalankan prosedur secara otomatis sebagai respons terhadap peristiwa tertentu dalam tabel, menjaga integritas data dan menegakkan aturan bisnis.

Perintah DCL seperti GRANT dan REVOKE digunakan untuk mengatur hak akses pengguna dalam basis data, memastikan keamanan dan kontrol akses yang baik. Backup membuat salinan data untuk melindungi dari kehilangan atau kerusakan, sementara restore mengembalikan data dari backup untuk memulihkan operasi sistem. Backup dan restore adalah bagian penting dari strategi pemulihan bencana dan migrasi data. Dengan memahami dan menerapkan konsep-konsep ini, kita dapat

merancang dan mengelola database dengan lebih efektif, memastikan data yang aman, konsisten, dan dapat diakses dengan cepat. MySQL Workbench menyediakan alat yang komprehensif untuk mendukung semua aspek ini, mempermudah pekerjaan pengembang dan administrator database.