

Materi Pengenalan Basis Data

penggunaan awal MySQL

Query

```
<mysql -u root -p>
```

Hasil

```
Setting environment for using XAMPP for Windows.
ASUS@LAPTOP-I1PB5MGJ g:\File XAMP
# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
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.

MariaDB [(none)]>
```

membuat data base

- `CREATE DATABASE` adalah perintah untuk membuat database baru.
- `[XI_RPL_1]` adalah nama yang Anda pilih untuk database baru Anda. Tanda kurung siku `<(" ["]>` digunakan di sini untuk menghindari kesalahan jika nama database mengandung karakter spesial atau spasi. Namun, perlu dicatat bahwa tidak semua DBMS mengizinkan penggunaan tanda kurung siku dalam nama database, jadi pastikan untuk menyesuaikan sintaks dengan DBMS yang Anda gunakan.

Query

```
create database xi_rpl_1;
```

Hasil:

```
MariaDB [(none)]> create database xi_rpl_1;  
Query OK, 1 row affected (0.004 sec)
```

Tampilkan data base

`SHOW DATABASE` digunakan untuk menampilkan daftar database yang ada dalam sistem manajemen basis data (DBMS). Perintah ini dapat digunakan di beberapa DBMS seperti MYSQL, PostgreSQL, dan beberapa DBMS lainnya. Namun, perintahnya dapat sedikit berbeda tergantung

Query

```
show databases;
```

Hasil:

```
MariaDB [(none)]> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| latihan_rpl1 |  
| latihan_rpl_1 |  
| mysql |  
| nafa |  
| performance_schema |  
| phpmyadmin |  
| rental_nafa |  
| test |  
| x_rpl_1 |  
| xi_rpl_1 |  
+-----+  
11 rows in set (0.086 sec)
```

Hapus database

`<DROP DATABASE [nama_database]>` digunakan dalam sistem manajemen basis data (DBMS) untuk menghapus sebuah database beserta semua objek yang terkait dengan database tersebut, seperti tabel, indeks, tampilan, prosedur tersimpan, dan lain-lain.

Query

```
drop database xi_rpl_1;
```

Hasil:

```
MariaDB [(none)]> drop database xi_rpl_1;  
Query OK, 0 rows affected (0.044 sec)
```

gunakan data base

`USE [nama_database]` digunakan dalam sistem manajemen basis data (DBMS) untuk beralih atau memilih database yang akan digunakan. Ketika Anda menggunakan perintah `<USE>` diikuti dengan nama database, DBMS akan mengarahkan semua perintah dan operasi selanjutnya pada database yang ditentukan.

Query

```
use rental_nafan;
```

Hasil:

```
MariaDB [(none)]> use rental_nafan;  
Database changed  
MariaDB [rental_nafan]>
```

Membuat Tabel

- Membuat tabel bernama `nafan_tables`.
- Tabel memiliki tiga kolom: `nama`, `alamat`, dan `catatan`.
- `nama` menyimpan teks maksimal 50 karakter.
- `alamat` menyimpan teks maksimal 100 karakter.
- `catatan` menyimpan teks panjang.

Query

```
CREATE TABLE nafan_tables (
    nama CHAR(50),
    alamat VARCHAR(100),
    catatan TEXT,
);
```

Hasil :

```
MariaDB [table_nafan]> CREATE TABLE nafan_tables (
->     nama CHAR(50),
->     alamat VARCHAR(100),
->     catatan TEXT
-> );
Query OK, 0 rows affected (0.012 sec)
```

```
MariaDB [table_nafan]> desc nafan_tables;
```

Field	Type	Null	Key	Default	Extra
nama	char(50)	YES		NULL	
alamat	varchar(100)	YES		NULL	
catatan	text	YES		NULL	

3 rows in set (0.024 sec)

Materi Select Lanjutan

AND

untuk **AND** ini akan mengambil "data 1" dan "data 2", contoh kodenya adalah seperti berikut :

Query :

```
SELECT warna,pemilik FROM data_mobil WHERE warna="Hitam" AND pemilik="ibrahim";
```

Hasil

```
MariaDB [rental_nafan]> SELECT warna,pemilik FROM data_mobil WHERE warna="Hitam" AND pemilik="ibrahim";
+-----+-----+
| warna | pemilik |
+-----+-----+
| Hitam | Ibrahim |
+-----+-----+
1 row in set (0.020 sec)
```

NOT BETWEEN

Untuk **NOT BETWEEN** ini akan mengambil "data" yang *bukan antara* "data 1" dan "data 2", contoh kodenya ialah seperti berikut :

Query :

```
SELECT * FROM data_mobil WHERE harga_rental NOT BETWEEN 100000 AND 200000;
```

Hasil

```
MariaDB [rental_nafan]> SELECT * FROM data_mobil WHERE harga_rental NOT BETWEEN 100000 AND 200000;
+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | 50000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | 50000 |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)
```

>=

Untuk **>=** ini akan mengambil "data" lebih besar atau sama dengan "nilai data", contoh kodenya ialah seperti berikut :

Query :

```
MariaDB [rental_nafan]> SELECT * FROM data_mobil WHERE harga_rental >= 50000;
```

Hasil

```
MariaDB [rental_nafan]> SELECT * FROM data_mobil WHERE harga_rental >= 50000;
+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.002 sec)
```

<> atau !=

Untuk <> atau != ini akan mengambil "data" yang tidak sama dengan "nilai data", contoh kodenya ialah seperti berikut :

Query :

```
MariaDB [rental_nafan]> SELECT * FROM data_mobil WHERE harga_rental <> 50000;
```

Hasil

```
MariaDB [rental_nafan]> SELECT * FROM data_mobil WHERE harga_rental <> 50000;
+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | 100000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.002 sec)
```

IN & OR

Struktur Query

```
SELECT * FROM [nama_tabel]
WHERE [nama_kolom] IN ('nilai')
OR [nama_kolom] = nilai;
```

Contoh

```
SELECT * FROM mobil
WHERE warna IN('Merah','silver')
OR harga_rental = 150000;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

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



```
MariaDB [rental_nafan]> SELECT * FROM mobil
-> WHERE warna IN('Merah','silver')
-> OR harga_rental = 150000;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000

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

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.
- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `warna IN('Merah','silver') OR harga_rental = 150000;` : query tersebut mencari data pada kolom warna bernilai Merah dan silver atau kolom harga_rental yang bernilai 150000 yang akan ditampilkan.

Kesimpulan

Menggunakan operator logika OR memastikan bahwa salah satu dari 2 kondisi/kolom tersebut memenuhi syarat akan diambil.

IN + AND + OPERATOR

LEBIH BESAR

Struktur Query

```
SELECT * FROM [nama_tabel]
WHERE [nama_kolom] IN ('nilai1', 'nilai2')
AND [nama_kolom] > nilai;
```

Query :

```
SELECT * FROM mobil
WHERE warna IN('Merah', 'silver')
AND harga_rental > 50000;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nafan]> SELECT * FROM mobil
-> WHERE warna IN('Merah', 'silver')
-> AND harga_rental > 50000;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.
- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `warna IN('Merah','silver') AND harga_rental > 50000`; : query tersebut mencari data pada kolom warna yang bernilai Merah atau silver dengan kriteria harga_rental berada di atas 50000.

Kesimpulan

Query tersebut mencari data pada kolom warna yang bernilai Merah atau silver dan pada saat yang sama kolom harga_rental harus lebih besar dari 50000.

Like

MENCARI AWALAN & AKHIRAN

Struktur Query

```
SELECT * FROM [nama_tabel]
WHERE [nama_kolom] LIKE 'nilai_awal%nilai_akhir';
```

Query :

```
SELECT * FROM mobil
WHERE pemilik LIKE 'i%m';
```

Hasil

```

MariaDB [rental_nafan]> SELECT * FROM mobil
-> ;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nafan]> SELECT * FROM mobil
-> WHERE pemilik LIKE 'i%m';
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

```

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.
- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `pemilik LIKE 'T%q';` : Perintah query tersebut akan menampilkan sebuah data pada kolom pemilik yang berawalan huruf T dan akhiran katanya q.

Kesimpulan

`SELECT * FROM mobil WHERE pemilik LIKE 'T%q';` : Perintah query tersebut mencari data pada tabel dimana kolom pemilik yang awalan katanya huruf T dan akhiran katanya q yang akan diambil.

KOMBINASI

Struktur Query :

```

SELECT * FROM [nama_tabel]
    WHERE [nama_kolom] LIKE 'nilai%'

SELECT * FROM [nama_tabel]
    WHERE [nama_kolom] LIKE 'nilai%';

```

Query :

```
SELECT * FROM mobil
WHERE pemilik LIKE '_d%';

SELECT * FROM mobil
WHERE pemilik LIKE '__i%';
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

```
MariaDB [rental_nafan]> SELECT * FROM mobil
-> WHERE pemilik LIKE '_i%';
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.003 sec)
```

```
MariaDB [rental_nafan]> SELECT * FROM mobil
-> WHERE pemilik LIKE '__i%';
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.002 sec)
```

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.

- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `pemilik LIKE '_d%'` ; perintah ini akan mencari data pada kolom pemilik, dimana underscore nya sebagai pengganti karakter awal dan karakter keduanya bernilai d.
- `pemilik LIKE '__i%'` ; perintah ini akan mencari data pada kolom pemilik, yang dimana 2 underscore nya sebagai pengganti nilai asli dan karakter ketiganya bernilai i.

Kesimpulan

- `SELECT * FROM mobil WHERE pemilik LIKE '_d%'` ; Perintah query tersebut akan mencari data pada tabel mobil, dimana kolom pemilik yang bernilai d pada karakter keduanya akan diambil.
 - `SELECT * FROM mobil WHERE pemilik LIKE '__i%'` ; Perintah query tersebut akan mencari data pada tabel mobil, dimana kolom pemilik yang bernilai i pada karakter ketiga yang akan diambil.
-

NOT LIKE

Struktur Query

```
SELECT * FROM [nama_tabel]
WHERE [nama_kolom] NOT LIKE 'nilai%';
```

Query :

```
SELECT * FROM mobil
WHERE pemilik NOT LIKE 'nafan%';
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

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

```
MariaDB [rental_nafan]> SELECT * FROM mobil  
-> WHERE pemilik NOT LIKE 'nafan%';
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

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

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.
- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `pemilik NOT LIKE 'opi%'` : Perintah query ini tidak akan menampilkan yang bernilai Opi pada kolom pemilik

Kesimpulan

`SELECT * FROM mobil WHERE pemilik NOT LIKE 'opi%'`; : Jadi Perintah tersebut mengecualikan sebuah data pada tabel mobil dimana kolom pemilik yang bernilai Opi tidak akan ditampilkan.

NULL

Struktur Query

```
SELECT * FROM [nama_tabel] WHERE [nama_kolom] IS NULL;
```

Query :

```
SELECT * FROM mobil WHERE peminjam IS NULL;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

```
5 rows in set (0.002 sec)
```



```
MariaDB [rental_nafan]> SELECT * FROM mobil WHERE peminjam IS NULL;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

```
2 rows in set (0.003 sec)
```

Analisis

- `SELECT * FROM` : Merupakan query awal untuk menampilkan sebuah tabel yang didalamnya berisikan kolom, di mana kolom tersebut menampung sebuah data.
- `mobil` nama dari tabel yang ingin kita tampilkan data-data nya.
- `WHERE` : digunakan untuk menyaring data yang ada pada tabel.
- `peminjam IS NULL` pada kolom peminjam akan menampilkan data yang kosong.

Kesimpulan

Null digunakan ketika kita ingin menampilkan kolom yang kosong/belum ada isinya.

Kombinasi Distinct dan Order By

Struktur Query

```
SELECT DISTINCT(nama_kolom) FROM [nama_tabel] ORDER BY [nama_kolom] DESC;
```

Query :

```
SELECT DISTINCT(harga_rental) FROM mobil ORDER BY harga_rental DESC;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nafan]> SELECT DISTINCT(harga_rental) FROM mobil ORDER BY harga_rental DESC;
+-----+
| harga_rental |
+-----+
| 150000 |
| 100000 |
| 50000 |
+-----+
3 rows in set (0.004 sec)
```

Analisis

- `SELECT DISTINCT (harga_rental)` digunakan untuk mengambil nilai unik dari kolom `harga_rental` dalam tabel, dan menghapus data duplikat sehingga hanya menampilkan data-data yang berbeda.
- `FROM mobil` Query tersebut memerintahkan untuk mengambil data pada tabel mobil.
- `ORDER BY harga_rental DESC` Query tersebut Mengurutkan isi dalam kolom `harga_rental` dari yang tertinggi.

Kesimpulan

Kombinasi Distinct dan Order By digunakan ketika ingin mengetahui jumlah data asli yang ada pada kolom secara berurutan, baik dari yang tertinggi ataupun yang terendah.

CONCAT, CONCAT_WS, AS

Menggabungkan kolom tanpa pemisah

Struktur Query

```
SELECT CONCAT(kolom1,kolom2) FROM [nama_tabel];
```

Query :

```
SELECT CONCAT(pemilik,warna) FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.002 sec)

MariaDB [rental_nafan]> SELECT CONCAT(pemilik,warna) FROM mobil;
+-----+
| CONCAT(pemilik,warna) |
+-----+
| IbrahimHitam          |
| IbrahimMerah          |
| BaimSilver            |
| IbeHitam              |
| IbeHitam              |
+-----+
5 rows in set (0.004 sec)
```

Analisis

- `SELECT CONCAT` digunakan untuk mengambil dan menggabungkan 2 kolom atau lebih menjadi satu string tunggal.
- `(pemilik,warna)` Merupakan kolom yang akan di gabungkan menjadi satu.
- `FROM mobil` query tersebut akan mengambil data dari tabel mobil.

kesimpulan

`CONCAT` digunakan ketika kita ingin menggabungkan suatu kolom menjadi satu.

Memberikan nama kolom alias

Struktur Query

```
SELECT CONCAT_WS(Separator,kolom1,kolom2) AS [nama_alias] FROM mobil;
```

Query :

```
SELECT CONCAT_WS("+",pemilik,peminjam) AS COLLAB FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
```

	id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000	
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000	
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000	
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000	
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000	

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

```
MariaDB [rental_nafan]> SELECT CONCAT_WS("+",pemilik,peminjam) AS COLLAB FROM mobil;
```

COLLAB
Ibrahim+Afdal
Ibrahim+Elia
Baim+Anty
Ibe
Ibe

```
5 rows in set (0.004 sec)
```

Analisis

- `SELECT CONCAT_WS` digunakan untuk mengambil dan menggabungkan 2 kolom atau lebih menjadi satu string tunggal dengan pemisah.
- `("+",pemilik,peminjam)` tanda "+" sebagai pemisah `pemilik,peminjam` merupakan kolom yang akan digabungkan nilainya,setiap nilai dari kolom `pemilik` dan `peminjam` akan digabungkan menjadi satu string dengan tanda "+" di antara mereka.
- `AS COLLAB` `AS` merupakan alias sedangkan `collab` nama dari penggabungan 2 kolom yaitu `pemilik` dan `peminjam`.
- `FROM mobil` query tersebut akan mengambil data dari tabel `mobil`.

kesimpulan

Kita bisa menggunakan Query tersebut untuk menggabungkan sebuah kolom dengan memberi nama alias, agar lebih mudah di mengerti.

VIEW

Membuat tabel virtual

Struktur Query

```
CREATE VIEW [nama_tabel_virtual] AS SELECT id_mobil, no_plat, pemilik, peminjam FROM [nama_tabel] WHERE [nama_kolom] = "nilai";
```

Query :

```
CREATE VIEW info_no_plat AS SELECT id_mobil, no_plat, pemilik, peminjam FROM mobil WHERE pemilik = "Baim";
```

Hasil

```
MariaDB [rental_nafan]> CREATE VIEW info_no_plat AS SELECT id_mobil, no_plat, pemilik, peminjam FROM mobil WHERE pemilik = "Baim";
Query OK, 0 rows affected (0.013 sec)
```

Analisis

- `CREATE VIEW info_no_plat AS` Perintah query tersebut membuat sebuah tabel virtual dengan nama `info_no_plat`.
- `SELECT id_mobil, no_plat, pemilik, peminjam FROM mobil` digunakan untuk memilih kolom-kolom dari tabel `mobil` untuk di masukkan ke dalam tabel virtual yang dibuat.
- `WHERE pemilik = "Baim";` perintah tersebut akan menampilkan data-data yang pemiliknya Dani.

kesimpulan

Tabel virtual digunakan untuk mengelompokkan sebuah kolom tertentu dari tabel yang sudah ada.

AGREGASI

Menghitung total nilai numerik suatu kolom (Sum)

Struktur Query

```
SELECT SUM(nama_kolom) AS [nama_alias] FROM [nama_tabel]
```

Query :

```
SELECT SUM(harga_rental) AS total_harga FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | Farhan | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nafan]> SELECT SUM(harga_rental) AS total_harga FROM mobil;
+-----+
| total_harga |
+-----+
| 450000 |
+-----+
1 row in set (0.004 sec)
```

Analisis

- `SELECT` digunakan untuk mengambil/menampilkan data
- `SUM` digunakan untuk menjumlahkan nilai numerik dalam kolom.
- `harga_rental` nama kolom yang ingin di jumlahkan nilainya.
- `AS total_harga` sebagai nama alias kolom
- `FROM mobil` query tersebut akan mengambil data dari tabel mobil.

Kesimpulan

`SUM` digunakan untuk menjumlahkan nilai numerik dalam kolom

Menghitung jumlah baris/data, biasanya berdasarkan kriteria tertentu (Count)

Struktur Query

```
SELECT COUNT(nama_kolom) AS [nama_alias] FROM  
[nama_tabel];
```

Query :

```
SELECT COUNT(pemilik) AS total_pemilik FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	Farhan	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

```
5 rows in set (0.003 sec)
```

```
MariaDB [rental_nafan]> SELECT COUNT(pemilik) AS total_pemilik FROM mobil;
```

total_pemilik
5

```
1 row in set (0.006 sec)
```

Analisis

- `SELECT` digunakan untuk mengambil/menampilkan data.
- `COUNT` digunakan untuk menghitung jumlah baris(nilai non-NULL).
- `pemilik` nama kolom yang ingin di hitung jumlah barisnya.
- `AS total_pemilik` sebagai nama alias kolom
- `FROM mobil` query tersebut akan mengambil data dari tabel mobil.

Kesimpulan

COUNT digunakan ketika ingin menghitung jumlah baris dalam suatu kolom(atau jumlah non-NULL).

Menampilkan nilai terendah (Min)

Struktur Query

```
SELECT MIN(nama_kolom) AS [nama_alias] FROM [nama_tabel];
```

Query :

```
SELECT MIN(harga_rental) AS minimum FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | Farhan | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nafan]> SELECT MIN(harga_rental) AS minimum FROM mobil;
+-----+
| minimum |
+-----+
| 50000 |
+-----+
1 row in set (0.005 sec)
```

Analisis

- `SELECT` digunakan untuk mengambil/menampilkan data.
- `MIN` digunakan untuk menampilkan nilai numerik terendah dalam suatu kolom.
- `harga_rental` nama kolom yang ingin kita tampilkan nilai terendahnya.
- `AS minimum` sebagai nama alias kolom
- `FROM mobil` query tersebut akan mengambil data dari tabel mobil.

Kesimpulan

`MIN` digunakan untuk menampilkan nilai terendah dari suatu kolom.

Menampilkan nilai rata-rata (Avg)

Struktur Query

```
SELECT AVG(nama_kolom) AS [nama_alias] FROM [nama_tabel];
```

Query :

```
SELECT AVG(harga_rental) AS rerata FROM mobil;
```

Hasil

```
MariaDB [rental_nafan]> select * from mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | Farhan | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.009 sec)

MariaDB [rental_nafan]> SELECT AVG(harga_rental) AS rerata FROM mobil;
+-----+
| rerata |
+-----+
| 90000.0000 |
+-----+
1 row in set (0.001 sec)
```

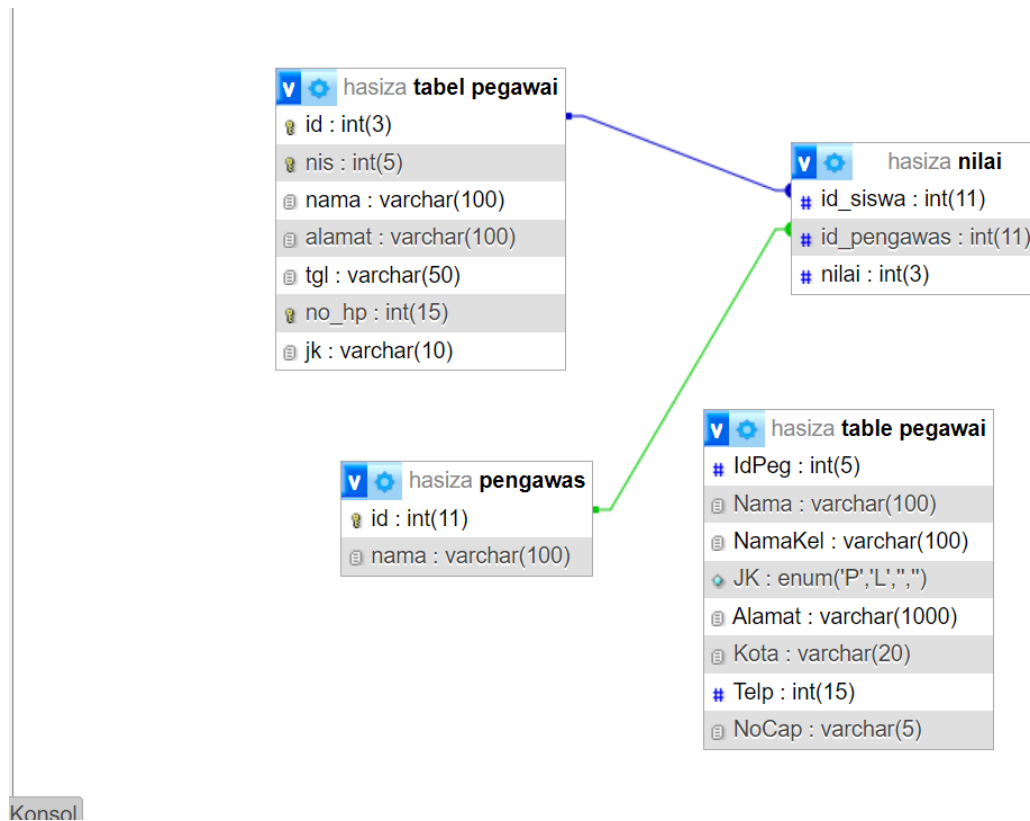
Analisis

- `SELECT` digunakan untuk mengambil/menampilkan data.
- `AVG` digunakan untuk menghitung rata-rata dari nilai numerik yang ada pada kolom
- `harga_rental` nama kolom yang ingin kita tampilkan nilai tertinggi nya.
- `AS rerata` sebagai nama alias kolom
- `FROM mobil` query tersebut akan mengambil data dari tabel mobil.

Kesimpulan

AVG digunakan untuk menghitung rata-rata dari nilai numerik pada suatu kolom.

Relasi Tabel



Struktur Pegawai

The screenshot shows the phpMyAdmin interface for the 'latihan_rpl_1' database. The 'Struktur tabel' (Table Structure) tab is selected for the 'pegawai' table. The table structure is as follows:

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	idpeg	int(7)			Tidak	Tidak ada			Ubah Hapus Lainnya
2	nama	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	namakel	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	jk	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	alamat	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	kota	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
7	telp	int(20)			Tidak	Tidak ada			Ubah Hapus Lainnya
8	nocab	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

Below the table structure, there are options to 'Pilih Semua' (Select All), 'Dengan pilihan:' (With selection:), and 'Tindakan' (Actions) including 'Ubah' (Edit), 'Hapus' (Delete), 'Utama' (Primary), 'Unik' (Unique), 'Indeks' (Index), and 'Spasial' (Spatial). There is also a 'Konsol' (Console) section at the bottom for running queries.

Data Pegawai

				idpeg	nama	namakel	jk	alamat	kota	telp	nocab
<input type="checkbox"/>				10107	Emya	Salsalina	P	Jl.Suci 78	Bandung	22555768	C101
<input type="checkbox"/>				10176	Diah	Wahyuni	P	Jl.Maluku 56	Bandung	22555934	C101
<input type="checkbox"/>				10246	Dian	Anggraini	P	Jl.Mawar 5	Semarang	24555102	C103
<input type="checkbox"/>				10252	Antoni	Irawan	L	Jl.A.Yani 15	Jakarta	21555888	C102
<input type="checkbox"/>				10307	Erik	Andrian	L	Jl.Manggis 155	Semarang	24555236	C103
<input type="checkbox"/>				10314	Ayu	Rahmadani	P	Jl.Malaka 342	Jakarta	21555098	C102
<input type="checkbox"/>				10324	Martin	Susanto	L	Jl.Bima 51	Jakarta	21555785	C102
<input type="checkbox"/>				10407	Rio	Gunawan	L	Jl.Melati 356	Surabaya	31555231	C104
<input type="checkbox"/>				10415	Susan	Sumantri	P	Jl.Pahlawan 24	Surabaya	31555120	C104

Before

phpMyAdmin

Terbaru Favorit

- Baru
- information_schema
- latihan
 - latihan_rpl1
 - latihan_rpl_1
 - Baru
 - nilai
 - pengawas11
 - siswa1
 - tabel cabang
 - tabel pegawai
- mysql
- nafan
- performance_schema
- phpmyadmin
- rental_nafan
- test
- x_rpl_1

Server: 127.0.0.1 » Database: latihan_rpl_1 » Tabel: tabel pegawai

[Jelajahi](#)
[Struktur](#)
[SQL](#)
[Cari](#)
[Tambahkan](#)
[Ekspor](#)
[Impor](#)
[Hak Akses](#)
[Operasi](#)
[Lainnya](#)

[Edit dikotak] [Ubah] [Buat kode PHP]

[Struktur tabel](#)
[Tampilan hubungan](#)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	idpeg	int(7)			Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 2	nama	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 3	namakel	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 4	jk	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 5	alamat	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 6	kota	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 7	telp	int(20)			Tidak	Tidak ada			Lainnya
<input type="checkbox"/> 8	nocab	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Lainnya

☐ Pilih Semua
 Dengan pilihan:
 [Jelajahi](#)

[Teks penuh](#)
[Add to central columns](#)
[Remove from central columns](#)

Konsol

Tekan Ctrl+Enter untuk menjalankan kueri

>

Bookmark Opsi Riwayat Bersihkan

After

phpMyAdmin

Server: 127.0.0.1 » Database: latihan_rpl_1 » Tabel: tabel pegawai

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	idpeg	int(7)			Tidak	Tidak ada			Ubah Hapus Lainnya
2	nama	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	namakel	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	jk	enum('L', 'P', '', '')	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	alamat	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	kota	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
7	telp	int(20)			Tidak	Tidak ada			Ubah Hapus Lainnya
8	nocab	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

Pilih Semua Dengan pilihan: Jelajahi Ubah Hapus Utama Unik Indeks Spasial

Teks penuh Add to central columns Remove from central columns

Cabang Before

phpMyAdmin

Server: 127.0.0.1 » Database: latihan_rpl_1 » Tabel: tabel cabang

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	Nocab	varchar(11)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
2	nama	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	alamat	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	Kabupaten	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	no_telp	int(100)			Tidak	Tidak ada			Ubah Hapus Lainnya
6	kota	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
7	email	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

Pilih Semua Dengan pilihan: Jelajahi Ubah Hapus Utama Unik Indeks Spasial

Teks penuh Add to central columns Remove from central columns

Cetak Usulkan struktur tabel Lacak tabel Move columns Normalisasi

Konsol

Tekan Ctrl+Enter untuk menjalankan kueri

```
> SELECT * FROM `tabel cabang`
>
```

localhost/phpmyadmin/index.php?route=/table/structure&db=latihan_rpl_1&table=tabel+cabang

After

phpMyAdmin

Server: 127.0.0.1 » Database: latihan_rpl_1 » Tabel: tabel cabang

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Lainnya

Terbaru Favorit

Baru
information_schema
latihan
latihan_rpl1
latihan_rpl_1
Baru
nilai
pengawas11
siswa1
tabel cabang
tabel pegawai
mysql
nafan
performance_schema
phpmyadmin
rental_nafan
test
x_rpl_1

✓ Tabel tabel cabang berhasil diubah.

```
ALTER TABLE `tabel cabang` CHANGE `email` `email` TEXT CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NOT NULL;
```

[Edit dikotak] [Ubah] [Buat kode PHP]

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	Nocab	varchar(11)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 2	nama	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 3	alamat	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 4	Kabupaten	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 5	no_telp	int(100)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 6	kota	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 7	email	text	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

Konsol

Tekan Ctrl+Enter untuk menjalankan kueri

Bookmark Opsi Riwayat Bersihkan

Materi Alter

Menambahkan kolom

INI SEBELUM MENAMBAHKAN ALTER

Struktur query

```
ALTER TABLE nama_tabel ADD nama_kolom varchar(10) AFTER nama_kolom;
```

Query :

```
ALTER TABLE data_mobil ADD batas_peminjam varchar(10) peminjam;
```

Hasil

```
MariaDB [rental_nafan]> ALTER TABLE data_mobil ADD batas_peminjam varchar(10) AFTER peminjam;
Query OK, 0 rows affected (0.014 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [rental_nafan]> desc data_mobil;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_mobil | int(2) | NO | PRI | NULL | |
| no_plat | varchar(10) | NO | UNI | NULL | |
| no_mesin | varchar(10) | NO | UNI | NULL | |
| warna | varchar(25) | NO | | NULL | |
| pemilik | varchar(25) | NO | | NULL | |
| peminjam | varchar(25) | YES | | NULL | |
| batas_peminjam | varchar(10) | YES | | NULL | |
| harga_rental | int(10) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.018 sec)
```

Inl Yang Sudah Ditambah

Query :

```
UPDATE mobil SET batas_peminjam="2024-04-24" WHERE peminjam IS NOT Null;
```

Hasil

```
MariaDB [rental_nafan]> select * from data_mobil;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_mobil | no_plat | no_mesin | warna | pemilik | peminjam | batas_peminjam | harga_rental |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | DD 2650 XY | ACX3568 | Hitam | Ibrahim | Afdal | NULL | 50000 |
| 2 | DD 2440 AX | BCS1120 | Merah | Ibrahim | Elia | NULL | 100000 |
| 3 | B 1611 QC | LSQ1112 | Silver | Baim | Anty | NULL | 50000 |
| 4 | DD 2901 JK | UQL1029 | Hitam | Ibe | NULL | NULL | 150000 |
| 5 | DD 2210 LS | CJH1011 | Hitam | Ibe | NULL | NULL | 100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.111 sec)
```

Analisis

- `ALTER TABLE` : Digunakan untuk mengubah struktur tabel yang sudah ada.
- `mobil` : Nama tabel yang akan diubah.
- `ADD batas_peminjam varchar(10)` : Menambahkan kolom baru bernama `batas_peminjam` dengan tipe data karakter (maksimal 10 karakter).
- `AFTER peminjam` : Menempatkan kolom baru `batas_peminjam` setelah kolom `peminjam` yang sudah ada.

Kesimpulan

Pernyataan SQL tersebut menambahkan kolom `batas_peminjam` ke tabel `mobil` dengan tipe data karakter (maksimal 10 karakter), dan menempatkannya setelah kolom `peminjam`.

 **Ini Yang Sudah Ditambah**

Query :

```
UPDATE mobil SET batas_peminjam="2024-04-24" WHERE peminjam IS NOT Null;
```

Hasil

```
MariaDB [rental_nafan]> select * from data_mobil;
```

id_mobil	no_plat	no_mesin	warna	pemilik	peminjam	batas_peminjam	harga_rental
1	DD 2650 XY	ACX3568	Hitam	Ibrahim	Afdal	NULL	50000
2	DD 2440 AX	BCS1120	Merah	Ibrahim	Elia	NULL	100000
3	B 1611 QC	LSQ1112	Silver	Baim	Anty	NULL	50000
4	DD 2901 JK	UQL1029	Hitam	Ibe	NULL	NULL	150000
5	DD 2210 LS	CJH1011	Hitam	Ibe	NULL	NULL	100000

```
5 rows in set (0.111 sec)
```

Mengubah tipe data kolom

Struktur query

```
ALTER TABLE nama_tabel MODIFY nama_kolom DATE;
```

Query

```
ALTER TABLE data_mobil MODIFY varchar(10) DATE;
```

Hasil

```
MariaDB [rental_nafan]> desc data_mobil;
```

Field	Type	Null	Key	Default	Extra
id_mobil	int(2)	NO	PRI	NULL	
no_plat	varchar(10)	NO	UNI	NULL	
no_mesin	varchar(10)	NO	UNI	NULL	
warna	varchar(25)	NO		NULL	
pemilik	varchar(25)	NO		NULL	
peminjam	varchar(25)	YES		NULL	
batas_peminjam	date	YES		NULL	
harga_rental	int(10)	YES		NULL	

8 rows in set (0.017 sec)

Analisis

- `ALTER TABLE` : Perintah SQL untuk mengubah struktur tabel.
- `mobil` : Nama tabel yang akan diubah.
- `MODIFY deadline DATE` : Mengubah kolom `deadline` dalam tabel `mobil` menjadi tipe data `DATE`.

Kesimpulan

Perintah `ALTER TABLE mobil MODIFY deadline DATE;` mengubah tipe data kolom `deadline` dalam tabel `mobil` menjadi `DATE`, memastikan kolom tersebut hanya menyimpan nilai tanggal yang valid.

Pastikan data kompatibel dan lakukan pencadangan sebelum menjalankan perintah ini.

Menambahkan constraint

Struktur query

```
ALTER TABLE nama_tabel ALTER nama_kolom SET DEFAULT 'READY';
```

Query

```
ALTER TABLE data_mobil ALTER deadline SET DEFAULT 'READY';
```

Hasil

Field	Type	Null	Key	Default	Extra
id_mobil	int(2)	NO	PRI	NULL	
no_plat	varchar(10)	NO	UNI	NULL	
no_mesin	varchar(10)	NO	UNI	NULL	
warna	varchar(25)	NO		NULL	
pemilik	varchar(25)	NO		NULL	
peminjam	varchar(25)	YES		NULL	
batas_peminjam	date	YES		NULL	
harga_rental	int(10)	YES		NULL	
deadline	varchar(10)	YES		READY	

9 rows in set (0.014 sec)

Analisis

Perintah `ALTER TABLE mobil ALTER deadline SET DEFAULT 'READY';` mengatur nilai default untuk kolom `deadline` dalam tabel `mobil` menjadi `READY`. Jika baris baru ditambahkan tanpa nilai untuk kolom `deadline`, nilainya otomatis akan diatur menjadi `READY`.

kesimpulan

Perintah `ALTER TABLE mobil ALTER deadline SET DEFAULT 'READY';` mengatur nilai default kolom `deadline` dalam tabel `mobil` menjadi `READY`. Ini berarti setiap baris baru tanpa nilai `deadline` akan otomatis diisi dengan `READY`. Pastikan ini sesuai dengan kebutuhan aplikasi Anda sebelum menjalankan perintah.

Referensi

<https://revou.co/panduan-teknis/sql-constraint>

Menghapus kolom

Struktur query

```
ALTER TABLE nama_tabel DROP COLUMN nama_kolom;
```

Query

```
ALTER TABLE data_mobil DROP COLUMN deadline;
```

Hasil

```
MariaDB [rental_nafan]> desc data_mobil;
```

Field	Type	Null	Key	Default	Extra
id_mobil	int(2)	NO	PRI	NULL	
no_plat	varchar(10)	NO	UNI	NULL	
no_mesin	varchar(10)	NO	UNI	NULL	
warna	varchar(25)	NO		NULL	
pemilik	varchar(25)	NO		NULL	
peminjam	varchar(25)	YES		NULL	
batas_peminjam	date	YES		NULL	
harga_rental	int(10)	YES		NULL	

```
8 rows in set (0.013 sec)
```

Analisis

- `ALTER TABLE` : Perintah SQL untuk mengubah struktur tabel.
- `data_mobil` : Nama tabel yang ingin diubah.
- `DROP COLUMN deadline` : Menghapus kolom `deadline` dari tabel `data_mobil`. Ini berarti semua data dalam kolom `deadline` akan dihapus.

kesimpulan

Perintah `ALTER TABLE data_mobil DROP COLUMN deadline;` menghapus kolom `deadline` dari tabel `data_mobil`. Pastikan untuk mencadangkan data sebelum menjalankan perintah ini dan konfirmasi bahwa penghapusan kolom sesuai dengan kebutuhan aplikasi Anda.

Mengganti nama tabel

Struktur query

```
ALTER TABLE nama_tabel RENAME TO nama_kolom;
```

Query

```
ALTER TABLE data_mobil RENAME TO ganteng;
```

Hasil

```
MariaDB [rental_nafan]> ALTER TABLE data_mobil RENAME TO mobil;  
Query OK, 0 rows affected (0.061 sec)
```

```
MariaDB [rental_nafan]> desc mobil;
```

Field	Type	Null	Key	Default	Extra
id_mobil	int(2)	NO	PRI	NULL	
no_plat	varchar(10)	NO	UNI	NULL	
no_mesin	varchar(10)	NO	UNI	NULL	
warna	varchar(25)	NO		NULL	
pemilik	varchar(25)	NO		NULL	
peminjam	varchar(25)	YES		NULL	
batas_peminjam	date	YES		NULL	
harga_rental	int(10)	YES		NULL	

```
8 rows in set (0.028 sec)
```

Analisis

1. `ALTER TABLE` : Ini adalah perintah SQL yang digunakan untuk mengubah struktur tabel.
2. `mobil` : Ini adalah nama tabel yang ingin Anda ubah.
3. `RENAME TO data_mobil` : Ini adalah bagian yang melakukan perubahan. Perintah ini mengubah nama tabel "`mobil`" menjadi "`data_mobil`".

kesimpulan

Perintah `ALTER TABLE mobil RENAME TO data_mobil;` mengubah nama tabel "`mobil`" menjadi "`data_mobil`". Pastikan untuk memperbarui kode atau query yang menggunakan nama tabel lama setelah menjalankan perintah ini.

Referensi pembahasan ALTER

https://www.techonthenet.com/mysql/tables/alter_table.php