

**LAPORAN PRAKTIKUM
BASIS DATA RD
MODUL 8**

Oleh :

Muhammad Fadhil Zurani (122140146)



**Program Studi Teknik Informatika
Fakultas Teknologi Industri
Institut Teknologi Sumatera
2024**

Daftar Isi

Daftar Isi	2
1. Dasar Teori.....	3
2. Ulasan	5
3. Hasil dan Jawaban.....	7
4. Kesimpulan dan Saran.....	12

1. Dasar Teori

A. Operator Aritmatika

Operator aritmatika digunakan untuk melakukan operasi matematika dasar pada nilai numerik. Berikut adalah operator aritmatika yang umum:

Penjumlahan (+): Menambahkan dua nilai.

SELECT 5 + 3; -- Hasilnya adalah 8

Pengurangan (-): Mengurangi satu nilai dari nilai lainnya.

SELECT 5 - 3; -- Hasilnya adalah 2

Perkalian (*): Mengalikan dua nilai.

SELECT 5 * 3; -- Hasilnya adalah 15

Pembagian (/): Membagi satu nilai dengan nilai lainnya.

SELECT 10 / 2; -- Hasilnya adalah 5

Modulus (%): Menghasilkan sisa dari pembagian dua nilai.

SELECT 10 % 3; -- Hasilnya adalah 1

B. Operator Pembandingan

Operator pembandingan digunakan untuk membandingkan dua nilai dan menghasilkan nilai boolean (TRUE atau FALSE). Berikut adalah operator pembandingan yang umum:

Sama dengan (=): Mengecek apakah dua nilai sama.

SELECT * FROM barang WHERE Harga_brg = 40000;

Tidak sama dengan (<> atau !=): Mengecek apakah dua nilai tidak sama.

SELECT * FROM barang WHERE Harga_brg <> 40000;

Lebih besar dari (>): Mengecek apakah nilai pertama lebih besar dari nilai kedua.

SELECT * FROM barang WHERE stok > 100;

Lebih kecil dari (<): Mengecek apakah nilai pertama lebih kecil dari nilai kedua.

SELECT * FROM barang WHERE stok < 100;

Lebih besar atau sama dengan (>=): Mengecek apakah nilai pertama lebih besar atau sama dengan nilai kedua.

SELECT * FROM barang WHERE stok >= 100;

Lebih kecil atau sama dengan (<=): Mengecek apakah nilai pertama lebih kecil atau sama dengan nilai kedua.

```
SELECT * FROM barang WHERE stok <= 100;
```

C. Operator Logika

Operator logika digunakan untuk menggabungkan dua atau lebih kondisi dan mengembalikan nilai boolean (TRUE atau FALSE). Berikut adalah operator logika yang umum:

AND: Menghasilkan TRUE jika kedua kondisi benar.

```
SELECT * FROM barang WHERE stok > 100 AND Harga_brg < 50000;
```

OR: Menghasilkan TRUE jika salah satu kondisi benar.

```
SELECT * FROM barang WHERE stok > 100 OR Harga_brg < 50000;
```

NOT: Membalikkan nilai kondisi, menghasilkan TRUE jika kondisi adalah FALSE dan sebaliknya.

```
SELECT * FROM barang WHERE NOT (stok > 100);
```

Penggunaan operator-operator ini memungkinkan kita untuk melakukan berbagai operasi dan kondisi dalam query SQL, memberikan fleksibilitas dan kekuatan dalam pengelolaan dan manipulasi data di dalam basis data.

2. Ulasan

LATIHAN OPERATOR

1. Buat sebuah database dengan nama “db_toko”, kemudian tampilkan seluruh daftar database yang ada.
2. Aktifkan database db_toko.
3. Buat tabel bernama “**brg**”, dengan ketentuan sebagai berikut:

Field Name	Type	Size	Keterangan
Kode_brg	Char	4	Primary Key
Nama_brg	Varchar	40	
Harga_brg	Int	10	
Thn_pembuatan	Year	-	
Stok	Int	3	

4. Lihat struktur tabel
5. Isikan data untuk tabel brg, dengan data sebagai berikut:

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	Stok
BR01	Clame Plate	40000	2005	100
BR02	CF Diafram	35000	2001	250
BR03	Press Cover	65000	2002	300
BR04	Terminal	15000	2000	57
BR05	Alumunium Solt	27000	2006	410

6. Tampilkan isi seluruh record pada tabel brg
7. Tampilkan isi record hanya field “kode_brg”, “nama_brg”, “stok”, kemudian urutkan data secara ascending berdasarkan nama_brg.
8. Tampilkan isi record tabel brg, hanya Nama Barang = “Terminal”.
9. Tampilkan isi record dengan Nama Barang depan huruf berawalan= “C”.
10. Tampilkan isi record hanya field kode_brg, nama_brg, harga_brg, stok dimana stok berada dibawah dari 200.
11. Tampilkan isi record barang dimana tahun pembuatan berada diantara tahun 2002-2006.
12. Tampilkan 3 isi record teratas dari tabel barang.

13. Tampilkan isi record barang dimana stok berada dibawah dari 200 dan tahun pembuatan tahun 2000.
14. Tampilkan isi record, hanya dari record ke-1 hingga record ke-3, kemudian urutkan berdasarkan kode_brg
15. Tampilkan isi record dengan Kode Barang kecuali BR05

3. Hasil dan Jawaban

1. Soal 1 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [(none)]> create database db_toko;  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [db_toko]> show databases;
```

```
+-----+  
| Database          |  
+-----+  
| akademik          |  
| bank_itera        |  
| db_latihan_dml    |  
| db_toko           |  
| galeri_itera      |  
| information_schema |  
| mysql             |  
| performance_schema |  
| phpmyadmin        |  
| test              |  
+-----+  
10 rows in set (0.001 sec)
```

Membuat database db_toko.

2. Soal 2 Screenshoot hasil dan jawaban dari pengujian

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

Menggunakan database db_toko

3. Soal 3 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> create table barang (
    → Kode_brg char(4) primary key not null,
    → Nama_brg varchar(40) not null,
    → Harga_brg int(10) not null,
    → Thn_pembuatan year not null,
    → stok int(3) not null
    → );
Query OK, 0 rows affected (0.008 sec)
```

Membuat tabel barang dengan ketentuan seperti pada ulasan.

4. Soal 4 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> desc barang;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Kode_brg       | char(4)       | NO   | PRI | NULL    |       |
| Nama_brg       | varchar(40)   | NO   |     | NULL    |       |
| Harga_brg      | int(10)       | NO   |     | NULL    |       |
| Thn_pembuatan  | year(4)       | NO   |     | NULL    |       |
| stok           | int(3)        | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.009 sec)
```

Melihat struktur tabel barang.

5. Soal 5 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> insert into barang values
    → ('BR01', 'Clame Plate', 40000, 2005, 100),
    → ('BR02', 'CF Diafram', 35000, 2001, 250),
    → ('BR03', 'Press Cover', 65000, 2002, 300),
    → ('BR04', 'Terminal', 15000, 2000, 57),
    → ('BR05', 'Alumunium Solt', 27000, 2006, 410);
Query OK, 5 rows affected (0.002 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

Menginputkan data ke tabel

6. Soal 6 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT * FROM barang;
```

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	stok
BR01	Clame Plate	40000	2005	100
BR02	CF Diafram	35000	2001	250
BR03	Press Cover	65000	2002	300
BR04	Terminal	15000	2000	57
BR05	Alumunium Solt	27000	2006	410

5 rows in set (0.000 sec)

Menampilkan isi seluruh record pada tabel barang

7. Soal 7 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT Kode_brg, Nama_brg, stok FROM barang ORDER BY Nama_brg ASC;
```

Kode_brg	Nama_brg	stok
BR05	Alumunium Solt	410
BR02	CF Diafram	250
BR01	Clame Plate	100
BR03	Press Cover	300
BR04	Terminal	57

5 rows in set (0.000 sec)

Menampilkan isi record hanya field "kode_brg", "nama_brg", "stok", kemudian urutkan data secara ascending berdasarkan nama_brg.

8. Soal 8 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT * FROM barang WHERE Nama_brg = 'Terminal';
```

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	stok
BR04	Terminal	15000	2000	57

1 row in set (0.000 sec)

Menampilkan isi record tabel barang, hanya nama barang = "Terminal"

9. Soal 9 Screenshoot hasil dan jawaban dari pengujian

```

MariaDB [db_toko]> SELECT * FROM barang WHERE Nama_brg LIKE 'C%';
+-----+-----+-----+-----+-----+
| Kode_brg | Nama_brg   | Harga_brg | Thn_pembuatan | stok |
+-----+-----+-----+-----+-----+
| BR01     | Clame Plate | 40000     | 2005          | 100  |
| BR02     | CF Diafram  | 35000     | 2001          | 250  |
+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)

```

Menampilkan isi record dengan Nama Barang depan huruf berawalan = "C"

10. Soal 10 Screenshoot hasil dan jawaban dari pengujian

```

MariaDB [db_toko]> SELECT Kode_brg, Nama_brg, Harga_brg, stok FROM barang WHERE stok
< 200;
+-----+-----+-----+-----+
| Kode_brg | Nama_brg   | Harga_brg | stok |
+-----+-----+-----+-----+
| BR01     | Clame Plate | 40000     | 100  |
| BR04     | Terminal   | 15000     | 57   |
+-----+-----+-----+-----+
2 rows in set (0.000 sec)

```

Menampilkan isi record hanya field kode_brg, nama_brg, harga_brg, stok, dimana stok berada di bawah dari 200

11. Soal 11 Screenshoot hasil dan jawaban dari pengujian

```

MariaDB [db_toko]> SELECT * FROM barang WHERE Thn_pembuatan BETWEEN 2002 AND 2006;
+-----+-----+-----+-----+-----+
| Kode_brg | Nama_brg   | Harga_brg | Thn_pembuatan | stok |
+-----+-----+-----+-----+-----+
| BR01     | Clame Plate | 40000     | 2005          | 100  |
| BR03     | Press Cover | 65000     | 2002          | 300  |
| BR05     | Alumunium Solt | 27000     | 2006          | 410  |
+-----+-----+-----+-----+-----+
3 rows in set (0.000 sec)

```

Menampilkan isi record barang dimana tahun pembuatan berada di antara tahun 2002-2006

12. Soal 12 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT * FROM barang LIMIT 3;
```

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	stok
BR01	Clame Plate	40000	2005	100
BR02	CF Diafram	35000	2001	250
BR03	Press Cover	65000	2002	300

3 rows in set (0.000 sec)

Menampilkan 3 isi record teratas dari tabel barang

13. Soal 13 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT * FROM barang WHERE stok < 200 AND Thn_pembuatan = 2000;
```

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	stok
BR04	Terminal	15000	2000	57

1 row in set (0.001 sec)

Menampilkan isi record barang dimana stok berada di bawah dari 200 dan tahun pembuatan tahun 2000

14. Soal 14 Screenshoot hasil dan jawaban dari pengujian

```
MariaDB [db_toko]> SELECT * FROM barang ORDER BY Kode_brg ASC LIMIT 3 OFFSET 0;
```

Kode_brg	Nama_brg	Harga_brg	Thn_pembuatan	stok
BR01	Clame Plate	40000	2005	100
BR02	CF Diafram	35000	2001	250
BR03	Press Cover	65000	2002	300

3 rows in set (0.000 sec)

Menampilkan isi record, hanya dari record ke-1 hingga record ke-3, kemudian urutkan berdasarkan kode_brg

15. Soal 15 Screenshoot hasil dan jawaban dari pengujian

```

MariaDB [db_toko]> SELECT * FROM barang WHERE Kode_brg ≠ 'BR05';
+-----+-----+-----+-----+-----+
| Kode_brg | Nama_brg   | Harga_brg | Thn_pembuatan | stok |
+-----+-----+-----+-----+-----+
| BR01     | Clame Plate | 40000     | 2005          | 100  |
| BR02     | CF Diafram  | 35000     | 2001          | 250  |
| BR03     | Press Cover | 65000     | 2002          | 300  |
| BR04     | Terminal    | 15000     | 2000          | 57   |
+-----+-----+-----+-----+-----+
4 rows in set (0.000 sec)

```

Menampilkan isi record dengan Kode Barang kecuali BR05

4. Kesimpulan dan Saran

Dalam praktikum basis data SQL ini, praktikan mempelajari dan mengaplikasikan berbagai jenis operator yang digunakan dalam query SQL. Pertama, praktikan mengeksplorasi operator aritmatika, yang memungkinkan praktikan untuk melakukan operasi matematika dasar seperti penjumlahan, pengurangan, perkalian, pembagian, dan modulus. Penggunaan operator aritmatika ini sangat penting dalam mengelola dan menghitung data numerik dalam database. Praktikan juga menggunakan operator perbandingan untuk membandingkan nilai-nilai dalam tabel, seperti mengecek kesamaan, perbedaan, dan perbandingan lebih besar atau lebih kecil dari nilai tertentu. Penggunaan operator perbandingan ini membantu dalam menyaring data dan mendapatkan informasi yang relevan sesuai dengan kondisi yang diinginkan.

Selanjutnya, praktikan menerapkan operator logika untuk menggabungkan beberapa kondisi dalam query SQL. Operator seperti AND, OR, dan NOT memungkinkan praktikan untuk membuat query yang lebih kompleks dan spesifik, yang sangat berguna dalam pengambilan keputusan berbasis data. Melalui praktikum ini, praktikan memahami bagaimana operator-operator ini berfungsi dan bagaimana menggunakannya secara efektif untuk memanipulasi dan mengakses data dalam database. Keseluruhan praktikum ini memberikan pemahaman mendalam tentang pentingnya penggunaan operator dalam SQL, yang merupakan dasar dalam analisis dan pengelolaan data dalam berbagai aplikasi database.