



**PROGRAM STUDI
TEKNIK INFORMATIKA**

MATA KULIAH

Pemrograman Basisdata



FUNGSI SQL

Capaian Pembelajaran

- Mahasiswa mampu menjelaskan fungsi SQL

Kemampuan Akhir yang Diharapkan

- Mahasiswa mampu menerapkan fungsi-fungsi SQL pada sebuah tabel

tabel mhs

```
MariaDB [mahasiswa]> select * from mhs;
```

nim	nama	alamat	kota	sex
A12.2001.00050	Awalina	Jl. Maluku 23	Surabaya	W
A12.2001.00100	Dwisari	Jl. Makmur 256	Semarang	W
A12.2001.00123	Cinanty	Jl. kawi I 2	Semarang	W
A12.2001.00255	Priyadi	Jl. Semeru 230	Kendal	P
A12.2001.00750	Zaendy	Jl. Flamboyan 307	Bawen	P

5 rows in set (0.000 sec)

Mengurutkan Data dg ORDER BY

Untuk mengurutkan data, maka kita dapat menggunakan ORDER BY.

Pengurutan dapat dilakukan secara ascending (urut naik) maupun urut menurun (descending).

Contoh 1 :

Mengurutkan berdasarkan sebuah field (field yang dijadikan kunci adalah field nama dengan urutan naik).

Perintahnya :

MySQL>Select * From mhs Order by nama;

```
MariaDB [mahasiswa]> select * from mhs order by nama;
```

nim	nama	alamat	kota	sex
A12.2001.00050	Awalina	Jl. Maluku 23	Surabaya	W
A12.2001.00123	Cinanty	Jl. kawi I 2	Semarang	W
A12.2001.00100	Dwisari	Jl. Makmur 256	Semarang	W
A12.2001.00255	Priyadi	Jl. Semeru 230	Kendal	P
A12.2001.00750	Zaendy	Jl. Flamboyan 307	Bawen	P

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

Contoh 2 :

Mengurutkan berdasarkan sebuah field (field yang dijadikan field kunci adalah field nama berdasarkan urutan menurun (descending)).

```
MySQL>Select * From mhs Order by nama Desc;
```

```
MariaDB [mahasiswa]> select * from mhs order by nama desc;
```

nim	nama	alamat	kota	sex
A12.2001.00750	Zaendy	Jl. Flamboyan 307	Bawen	P
A12.2001.00255	Priyadi	Jl. Semeru 230	Kendal	P
A12.2001.00100	Dwisari	Jl. Makmur 256	Semarang	W
A12.2001.00123	Cinanty	Jl. kawi I 2	Semarang	W
A12.2001.00050	Awalina	Jl. Maluku 23	Surabaya	W

5 rows in set (0.000 sec)

Contoh 3:

Mengurutkan berdasarkan beberapa field (field yang dijadikan field kunci adalah field sex dan nama).

Perintahnya :

MySQL>Select * From mhs Order by sex, nama;

```
MariaDB [mahasiswa]> select * from mhs order by sex, nama;
```

nim	nama	alamat	kota	sex
A12.2001.00255	Priyadi	Jl. Semeru 230	Kendal	P
A12.2001.00750	Zaendy	Jl. Flamboyan 307	Bawen	P
A12.2001.00050	Awalina	Jl. Maluku 23	Surabaya	W
A12.2001.00123	Cinanty	Jl. kawi I 2	Semarang	W
A12.2001.00100	Dwisari	Jl. Makmur 256	Semarang	W

5 rows in set (0.001 sec)

Contoh 4:

Mengurutkan berdasarkan beberapa field (field yang dijadikan field kunci adalah field sex secara ascending dan nama secara descending).

Perintahnya :

MySQL>Select * From mhs Order by sex Asc, nama Desc;

```
MariaDB [mahasiswa]> select * from mhs order by sex asc, nama desc;
+-----+-----+-----+-----+-----+
| nim          | nama    | alamat                | kota    | sex |
+-----+-----+-----+-----+-----+
| A12.2001.00750 | Zaendy  | Jl. Flamboyan 307     | Bawen   | P   |
| A12.2001.00255 | Priyadi | Jl. Semeru 230        | Kendal  | P   |
| A12.2001.00100 | Dwisari | Jl. Makmur 256        | Semarang | W   |
| A12.2001.00123 | Cinanty | Jl. kawi I 2          | Semarang | W   |
| A12.2001.00050 | Awalina | Jl. Maluku 23         | Surabaya | W   |
+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

Contoh 5:

Mengurutkan berdasarkan nomor kolom tertentu (akan diurutkan data berdasarkan kolom kedua dari field yang disebutkan dalam Select).

Perintahnya :

MySQL>Select nim, nama From mhs Order by 2 Desc;

```
MariaDB [mahasiswa]> select nim, nama from mhs order by 2 desc;
+-----+-----+
| nim           | nama    |
+-----+-----+
| A12.2001.00750 | Zaendy  |
| A12.2001.00255 | Priyadi |
| A12.2001.00100 | Dwisari |
| A12.2001.00123 | Cinanty |
| A12.2001.00050 | Awalina |
+-----+-----+
5 rows in set (0.001 sec)
```


Mengelompokkan Data dengan GROUP BY

Untuk mengelompokkan data, maka dapat digunakan perintah GROUP BY.

Contoh :

Akan dikelompokkan data mahasiswa berdasarkan sex.

Perintahnya :

MySQL>Select nama, sex From mhs Group by sex;

```
MariaDB [mahasiswa]> select nama, sex from mhs group by sex;
```

```
+-----+-----+
| nama   | sex   |
+-----+-----+
| Priyadi | P     |
| Awalina | W     |
+-----+-----+
2 rows in set (0.001 sec)
```

nim	nama	alamat	kota	sex
A12.2001.00050	Awalina	Jl. Maluku 23	Surabaya	W
A12.2001.00100	Dwisari	Jl. Makmur 256	Semarang	W
A12.2001.00123	Cinanty	Jl. kawi I 2	Semarang	W
A12.2001.00255	Priyadi	Jl. Semeru 230	Kendal	P
A12.2001.00750	Zaendy	Jl. Flamboyan 307	Bawen	P

Menentukan Kondisi pada GROUP BY

Untuk memberikan kondisi pada saat mengelompokkan data, maka kita dapat menggunakan HAVING.

Contoh 1 :

Akan dikelompokkan data mahasiswa berdasarkan agama yang agamanya adalah selain Budha (kode=5)

Perintahnya :

MySQL>Select agama From mhs Group by agama Having agama <> '5';

Contoh 2 :

Mencari daftar kota dari data mahasiswa yang kotanya selain Kendal

```
MariaDB [mahasiswa]> select kota from mhs group by kota having kota <> 'Kendal';
+-----+
| kota  |
+-----+
| Bawen |
| Semarang |
| Surabaya |
+-----+
3 rows in set (0.001 sec)
```

Menggunakan Fungsi Agregasi

Fungsi agregasi adalah fungsi yang disediakan oleh MySQL untuk menghasilkan sebuah nilai berdasarkan sejumlah data.

Adapun beberapa fungsi agregasi pada MySQL adalah :

Fungsi Agregasi	Keterangan
AVG	Memperoleh nilai rata – rata
Count	Menghitung jumlah data
Max	Menghasilkan nilai terbesar
Min	Menghasilkan nilai terkecil
Sum	Menjumlahkan suatu nilai

Tabel pegawai

```
MariaDB [mahasiswa]> desc pegawai;
```

Field	Type	Null	Key	Default	Extra
nip	char(5)	YES		NULL	
nama	varchar(30)	YES		NULL	
depart	varchar(15)	YES		NULL	
gaji	int(11)	YES		NULL	

```
4 rows in set (0.034 sec)
```

```
MariaDB [mahasiswa]> select * from pegawai;
```

nip	nama	depart	gaji
11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Fungsi AVG

Digunakan untuk memperoleh nilai rata-rata.

Contoh 1 :

Menghitung nilai rata gaji pegawai dari tabel pegawai.

MySQL>Select AVG(gaji) From pegawai;

```
MariaDB [mahasiswa]> select avg(gaji) from pegawai;
+-----+
| avg(gaji) |
+-----+
| 5428.5714 |
+-----+
1 row in set (0.001 sec)
```

Fungsi AVG

Digunakan untuk memperoleh nilai rata-rata.

Contoh 2 :

Menghitung nilai rata-rata gaji pegawai per departemen.

MySQL>Select Depart, AVG(gaji) From pegawai Group by depart;

```
MariaDB [mahasiswa]> select depart, avg(gaji) from pegawai group by depart;
```

depart	avg(gaji)
Akunting	4625.0000
Pemasaran	4500.0000
Produksi	7500.0000

3 rows in set (0.001 sec)

depart	gaji
Akunting	5000
Akunting	5500
Pemasaran	4500
Produksi	7500
Akunting	3500
Produksi	7500
Akunting	4500

Contoh 3 :

Menghitung gaji rata pegawai di departemen Akunting dari tabel pegawai.

MySQL>Select avg(gaji) From Pegawai Group by depart Having depart = 'Akunting';

```
MariaDB [mahasiswa]> select avg(gaji) from pegawai group by depart having depart = 'Akunting';
+-----+
| avg(gaji) |
+-----+
| 4625.0000 |
+-----+
1 row in set (0.000 sec)
```

Fungsi COUNT

Digunakan untuk melakukan pencacahan terhadap data.

Contoh 1 :

Menghitung jumlah pegawai dari tabel pegawai.

Perintahnya :

```
MySQL>Select Count(*) From pegawai;
```

```
MariaDB [mahasiswa]> select count(*) from pegawai;
```

```
+-----+  
| count(*) |  
+-----+  
|          7 |  
+-----+
```

```
1 row in set (0.000 sec)
```

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Contoh 2 :

Menghitung jumlah pegawai per departemen dari tabel pegawai.

Perintahnya :

MySQL>Select depart, count(*) From Pegawai Group by depart;

```
MariaDB [mahasiswa]> select depart, count(*) from pegawai group by depart;
```

depart	count(*)
Akunting	4
Pemasaran	1
Produksi	2

3 rows in set (0.001 sec)

id	nama	depart	gaji
11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

depart	count(*)
Akunting	4
Pemasaran	1
Produksi	2

Contoh 3 :

Menghitung jumlah pegawai per departemen, khusus untuk yang jumlah lebih dari 1.

Perintahnya :

MySQL>Select depart, count(*) From Pegawai Group by depart Having Count(depart) > 1;

```
MariaDB [mahasiswa]> select depart, count(*) from pegawai group by depart having count(depart) > 1;
+-----+-----+
| depart | count(*) |
+-----+-----+
| Akunting | 4 |
| Produksi | 2 |
+-----+-----+
2 rows in set (0.000 sec)
```

Fungsi MAX

Digunakan untuk memperoleh nilai yang terbesar.

Contoh 1 :

Memperoleh gaji terbesar dari tabel pegawai.

Perintahnya :

MySQL>Select Max(Gaji) From Pegawai;

```
MariaDB [mahasiswa]> select max(gaji) from pegawai;
```

```
+-----+  
| max(gaji) |  
+-----+  
|      7500 |  
+-----+
```

```
1 row in set (0.001 sec)
```

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Contoh 2 :

Menghitung gaji terbesar per departemen dari tabel pegawai

Perintahnya :

MySQL>Select Max(Gaji) From Pegawai Group by depart;

```
MariaDB [mahasiswa]> select max(gaji) from pegawai group by depart;
```

max(gaji)
5500
4500
7500

3 rows in set (0.001 sec)

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Fungsi MIN

Digunakan untuk memperoleh nilai terkecil.

Contoh 1:

Memperoleh nilai terkecil dari tabel pegawai

Perintahnya :

MySQL>Select Min(Gaji) From Pegawai;

```
MariaDB [mahasiswa]> select min(gaji) from pegawai;
+-----+
| min(gaji) |
+-----+
|      3500 |
+-----+
1 row in set (0.000 sec)
```

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Fungsi MIN

Digunakan untuk memperoleh nilai terkecil.

Contoh 2 :

Menghitung gaji terkecil per departemen

Perintahnya :

MySQL>Select Min(Gaji) From Pegawai Group by depart;

```
MariaDB [mahasiswa]> select min(gaji) from pegawai group by depart;
```

```
+-----+  
| min(gaji) |  
+-----+  
|      3500 |  
|      4500 |  
|      7500 |  
+-----+
```

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

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Fungsi SUM

Digunakan untuk melakukan penjumlahan data.

Contoh 1 : Menghitung total gaji dari tabel pegawai

Perintahnya :

MySQL>Select sum(gaji) From Pegawai;

```
MariaDB [mahasiswa]> select sum(gaji) from pegawai;
```

```
+-----+  
| sum(gaji) |  
+-----+  
|      38000 |  
+-----+
```

```
1 row in set (0.000 sec)
```

11111	Mia	Akunting	5000
11112	Mila	Akunting	5500
11113	Karin	Pemasaran	4500
11114	Jamal	Produksi	7500
11115	Lidia	Akunting	3500
22222	Dita	Produksi	7500
22225	Wakso	Akunting	4500

Fungsi SUM

Digunakan untuk melakukan penjumlahan data.

Contoh 2: Menghitung total gaji pegawai per departemen

Perintahnya :

MySQL>Select depart, sum(gaji) From Pegawai Group by depart;

```
MariaDB [mahasiswa]> select depart, sum(gaji) from pegawai group by depart;
```

depart	sum(gaji)
Akunting	18500
Pemasaran	4500
Produksi	15000

3 rows in set (0.001 sec)

depart	gaji
Akunting	5000
Akunting	5500
Pemasaran	4500
Produksi	7500
Akunting	3500
Produksi	7500
Akunting	4500

Praktikum Mandiri 5

FUNGSI SQL

Latihan

1. Buatlah sebuah tabel pada database yang sudah anda miliki dengan nama tabel mkuliah dan struktur tabelnya adalah sebagai berikut :

Nama Field	Type Data	Keterangan
Kode_kul	Char(7)	Primary key, tidak boleh, unik
Nama_kul	Char(20)	Tidak boleh kosong
SKS	Decimal(2)	Tidak boleh kosong
Smt_tawar	Char(1)	Diisikan 1,2,3,4 dst
Klp_kul	Char(1)	Dapat diisikan angka 1-3 : 1 = MPK 2 = MKK 3 = MKB default adalah 1
Progdi	Char(1)	Dapat diisikan angka 1-2 : 1 = MI-D3 2 = SI-S1

2. Isikan data – data berikut :

'A21-101','Pancasila', 2, '1', '1','1'

'A21-102','Logika Algoritma', 4, '1', '2','1'

'A21-103','Pemrograman Basic', 2, '1', '2','1'

'A21-201','Sistem Basis Data I', 2, '2', '2','1'

'A21-202','Pemrograman Pascal', 4, '2', '2','1'

'A21-203','Ilmu Sosial Dasar', 2, '2', '1','1'

'A21-501','Bimbingan Karier', 2, '5', '3','1'

'A21-601','Proyek Akhir', 4, '6', '3','1'

3. Tampilkan seluruh data dari tabel mkuliah.

4. Tampilkan Nama mata kuliah yang sksnya 2 dan ditawarkan disemester 6

5. Tampilkan nama mata kuliah yang sksnya lebih besar dari 2 sks

6. Tampilkan nama mata kuliah yang huruf depannya dimulai dengan 'P'

7. Gantilah nama mata kuliah 'Pancasila' menjadi 'Pendidikan Pancasila'

8. Gantilah nama mata kuliah 'Pemrograman Basic' menjadi 'Pemrograman (Basic)'
9. Tambahkan record-record berikut :
 - 'A12-202', 'Pemrograman Pascal', 4, '2', '2', '2'
 - 'A12-203', 'Kewiraan', 2, '2', '1', '2'
 - 'A12-501', 'Bimbingan Karier', 2, '5', '3', '2'
 - 'A12-601', 'Tugas Akhir', 6, '6', '3', '2'
10. Gantilah nama mata kuliah 'Kewiraan' menjadi 'Pendidikan Kewarganegaraan'
11. Hitunglah seluruh data yang telah tersimpan pada tabel mkuliah tersebut.
12. Tampilkan nama mata kuliah dan sks yang telah diurutkan berdasarkan kunci nama mata kuliah secara ascending
13. Tampilkan nama mata kuliah, sks, semester tawar dan program studi yang telah diurutkan berdasarkan field kunci semester tawar dengan urutan ascending dan field kunci program studi dengan urutan descending
14. Kelompokkan data mata kuliah berdasarkan kelompok mata kuliah.

15. Tampilkan kelompok mata kuliah dan hitung jumlah sks berdasarkan kelompok mata kuliah
16. Tampilkan kode program studi dan hitung jumlah sks berdasarkan kelompok program studi
17. Tampilkan jumlah sks terkecil untuk program studi MI-D3
18. Tampilkan jumlah sks terbesar untuk program studi MI-D3
19. Tampilkan jumlah sks terkecil untuk program studi SI-S1
20. Tampilkan sks rata-rata untuk mata kuliah kelompok MKP
21. Tampilkan sks rata-rata untuk mata kuliah kelompok MKB
22. Hitung jumlah sks untuk mata kuliah kelompok MPK
23. Hitung jumlah sks untuk mata kuliah kelompok MKK
24. Berapa jumlah mata kuliah yang termasuk dalam kelompok MPK