

# Tabel Pegawai

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

idpeg	nama	namakel	jk	alamat	kota	Gaji	Jabatan	noCab
10107	Emya	Salsalina	P	Jl.suci 78	Bandung	5250000	Manager	C101
10176	Diah	Wahyuni	P	Jl.Maluku 56	Bandung	2750000	Sales	C103
10246	Dian	Anggraini	P	Jl.Mawar 5	Semarang	1750000	Staf	C102
10252	Antoni	Irawan	L	Jl.A.Yani 15	Jakarta	5750000	Manager	C102
10307	Erik	Andrian	L	Jl.Manggis 155	Semarang	2500000	Sales	C101
10314	Ayu	Rahmadani	P	Jl.Malaka 342	Jakarta	1950000	Staf	C102
10324	Martin	Susanto	L	Jl.Bima 51	Jakarta	6250000	Manager	C103
10407	Rio	Gunawan	L	Jl.Melati 356	Surabaya	2650000	Sales	C104
10415	Susan	Sumantri	P	Jl.Pahlawan 24	Surabaya	1725000	Staf	C104

```
9 rows in set (0.041 sec)
```

1

## Query

```
SELECT COUNT(NIP) AS JumlahPegawai, COUNT(jabatan) AS JumlahJabatan FROM pegawai2;
```

## Hasil

```
MariaDB [basis_data]> SELECT * FROM pegawai2;
```

NIP	NDep	NBIk	JK	Alamat	Telp	jabatan	gaji	NoCab
10107	Emya	Salsalina	P	Jl. Suci 78 Bandung	022-555768	Manajer	5250000	C101
10246	Dian	Anggraini	P	Jl. Mawar 5 Semarang	024-555102	Sales	2750000	C103
10324	Martin	Susanto	L	Jl. Bima 51 Jakarta	021-555785	Staf	1750000	C102
10252	Antoni	Irawan	L	Jl. A. Yani 15 Jakarta	021-555888	Manajer	5750000	C102
10176	Diah	Wahyuni	P	Jl. Maluku 56 Bandung	022-555934	Sales	2500000	C101
10314	Ayu	Rahmadani	P	Jl. Malaka 342 Jakarta	021-555098	Sales	1950000	C102
10307	Erik	Andrian	L	Jl. Manggis 5 Semarang	024-555236	Manajer	6250000	C103
10415	Susan	Sumantri	P	Jl. Pahlawan 24 Surabaya	031-555120	NULL	2650000	C104
10407	Rio	Gunawan	L	Jl. Melati 356 Surabaya	031-555231	Staf	1725000	C104

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

```
MariaDB [basis_data]> SELECT COUNT(NIP) AS JumlahPegawai, COUNT(jabatan) AS JumlahJabatan FROM pegawai2;
```

JumlahPegawai	JumlahJabatan
9	8

```
1 row in set (0.010 sec)
```

## Analisis

- `SELECT` : Untuk memilih kolom apa saja yang ingin dipilih (untuk dihitung)
- `COUNT(NIP)` : Untuk menghitung jumlah barisan data yang mempunyai isi data dari kolom yang dipilih. `NIP` adalah nama kolom yang dipilih untuk dihitung

- AS : Untuk mengubah nama dari suatu kolom untuk sementara
- "JumlahPegawai" : merupakan nama ubahan dari perintah AS yang digunakan merupakan nama sementara dari perintah COUNT(NIP)
- COUNT(jabatan) : untuk menghitung jumlah barisan data yang mempunyai isi data dari kolom yang dipilih jabatan adalah nama kolom yang dipilih untuk dihitung
- AS : Untuk mengubah nama dari suatu kolom untuk sementara
- "JumlahJabatan" : merupakan nama sementara dari perintah COUNT(jabatan)
- FROM "pegawai2" : merupakan dari tabel mana datanya yang digunakan "pegawai2" adalah nama tabel yang datanya ingin digunakan
- Hasilnya : karena ada 9 barisan data, yang ingin dihitung adalah kolom NIP , jumlah dari kolom NIP (isi datanya) ada 9, ditampilkan sebagai "JumlahPegawai". Kolom "jabatan" juga dihitung, akan tetapi ada satu data yang berisi NULL (kosong) oleh karena itu hanya ada 8 data ditampilkan sebagai "JumlahJabatan"

## 2

### Contoh query

```
select COUNT(NIP) AS jumlahpegawai
from pegawai
where NoCab = 'C102';
```

### Hasil

```
MariaDB [nitip]> select COUNT(NIP) AS jumlahpegawai
-> from pegawai
-> where NoCab = 'C102';
+-----+
| jumlahpegawai |
+-----+
|          3    |
+-----+
1 row in set (0.582 sec)
```

### Analisis

- Select = untuk memilih kolom mana saja yang ingin dipilih untuk dihitung.
- COUNT(NIP) = untuk menghitung jumlah barisan data yang mempunyai data dari kolom yang dipilih.  
Nip adalah nama kolom yang dipilih untuk dihitung.
- AS = untuk mengubah nama dari suatu kolom sementara.
- JumlahPegawai = nama sementara yang dipilih untuk kolom COUNT(NIP)

- `FROM Pegawai` = dari tabel mana datanya akan digunakan. Pegawai adalah nama tabel yang dipilih untuk digunakan.
- `WHERE` = merupakan kondisi yang harus dipenuhi agar datanya dapat dihitung dengan query `COUNT(NIP)`.
- `(Nocab = 'C102')` = adalah kondisi dari where yang harus dipenuhi, jadi hanya barisan data yang memiliki 'C102' di kolom "Nocab" yang bisa dihitung.
- Hasilnya = di 9 barisan data yang ada pada tabel pegawai, kita ingin menghitung jumlah barisan data yang memiliki nilai 'C102' pada kolom "Nocab"nya dengan menggunakan `COUNT`. Jadi yang muncul adalah 3 barisan data. Kita juga ingin mengubah nama dari kolom hasil perintah `COUNT` secara sementara dengan perintah `AS` namanya adalah `JumlahPegawai`.

### 3

## Contoh query

```
SELECT Nocab, COUNT(NIP) AS Jumlah_pegawai
  from pegawai
 GROUP BY NoCab;
```

## Hasil

```
MariaDB [nitip]> SELECT Nocab, COUNT(NIP) AS Jumlah_pegawai
-> from pegawai
-> GROUP BY NoCab;
```

Nocab	Jumlah_pegawai
C101	2
C102	3
C103	2
C104	2

```
4 rows in set (0.081 sec)
```

## Analisis

- `SELECT` = untuk memilih kolom mana saja yang ingin dihitung atau ditampilkan.
- `Nocab` = merupakan nama kolom yang ingin ditampilkan.
- `'COUNT(NIP)'` = untuk menghitung jumlah barisan data yang mempunyai isi data dari kolom yang dipilih.  
Nip adalah nama kolom yang ingin dipilih untuk dihitung.
- `AS` = untuk mengubah nama dari suatu kolom untuk sementara.
- `Jumlah Pegawai` = merupakan nama sementara dari kolom hasil `COUNT(NIP)`.

- `From Pegawai` = dari tabel mana yang data kodomnya ingin digunakan. Pegawai adalah nama tabel yang dipilih untuk digunakan.
- `GROUP BY` = untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan Pada kolom yang dipilih.
- `Nocab` = nama kolom Yang dipilih untuk datanya dikelompokkan.
- Hasilnya = Berdasarkan 9 barisan data, masing-masing nilai dalam kolom Nocab dikelompokkan berdasarkan nilainya sendiri. Jadi Nocab clol bersama Nocab yang nilainya sama Yaitu clol. Jadi Nocab Yang memiliki C101 ada 2, C102 ada 3, C103 ada 2, C104 ada 2. Total semuanya 9, sesuai densan Jumlah barisan data Yand ada. Adapun nama dari kolom hasil Yaitu Jumlah-Pegawai dari Perintah AS.

## 4

### Contoh query

```
SELECT Nocab, COUNT(NIP) AS Jumlah_pegawai
-> from pegawai
-> GROUP BY NoCab HAVING COUNT(NIP) >= 3;
```

### Hasil

```
MariaDB [nitip]> SELECT Nocab, COUNT(NIP) AS Jumlah_pegawai
-> from pegawai
-> GROUP BY NoCab HAVING COUNT(NIP) >= 3;
+-----+-----+
| Nocab | Jumlah_pegawai |
+-----+-----+
| C102  | 3              |
+-----+-----+
1 row in set (0.067 sec)

MariaDB [nitip]> _
```

### Analisis

- `SELECT` = untuk memilih kolom mana saja yang ingin dihitung atau ditampilkan.
- `Nocab` = merupakan nama kolom yang ingin ditampilkan.
- `COUNT (NIP)` = untuk menghitung Jumlah barisan data yang mempunyai isi data dari kolom Yang dipilih.  
NIP adalah nama kolom Yang dipilih untuk dihitung.
- `AS` = untuk mengubah nama dari suatu kolom untuk sementara.
- `Jumlah_Pegawai` = nama sementara dati kolom hasil COUNT (NIP).
- `From Pegawai` = untuk memilih dari tabel mana Yand duta kolomnya ingin digunakan. Pegawai adalah nama tabel Xang dipilih untuk digunakan.

- **GROUP BY** = untuk menjelompokkan data berdasarkan nilai data Yang telah ditentukan Pada kolom Yang dipilih.
- **Nocab** - nama kolom Yang dipilih untuk dikelompokkan datanya.
- **HAVING** = untuk menentukan kondisi (Yand hans dipenuhi) oleh suatu kelompok data agar bisa ditampilkan.
- **(COUNT (NIP) >= 3)** = merupakan kondisi Yand harus dipenuhi oleh suatu kelompok data. Jadi hanya kelompok data Yang hasil hitungannya lebih atau Sama dengan 3.
- Hasilnya seperti sebelumnya, ada 9 barisan data dibagi sesuai Nocab nya masing-masing. Namun Yand ingin ditampilkan adalah hasil hitungan yang lebih dari atau sama dengan 3. Yaitu Nocab C102 Yang ada 3. Yand Jain c101 ada 2, c103 ada 2, C104 ada 2.

## 5

### Query

```
Select SUM(Gaji) AS Total_Gaji
->FROM pegawai;
```

### Hasil

```
MariaDB [perusahaan]> select sum(Gaji) AS Total_Gaji
-> FROM pegawai;
+-----+
| Total_Gaji |
+-----+
|      30.575 |
+-----+
1 row in set, 9 warnings (0.863 sec)
```

### Analisis

- **SELECT** = Untuk memilih kolom mana saja yang dipilih untuk dijumlahkan.
- **SUM(Gaji)** = Untuk menghitung jumlah data (khusus angka) pada kolom yang harus dipilih. Gaji merupakan nama kolom yang dipilih untuk dihitung jumlah isi datanya.
- **AS** = Untuk mengganti nama dari kolom hasil SUM(Gaji) untuk sementara.
- **Total\_Gaji** = Merupakan nama sementara dari perintah AS.
- **FROM pegawai** = Untuk memilih dari table mana yang kolom datanya akan digunakan. Pegawai adalah nama dari table yang dipilih.

- *\*Hasilnya* = kolom Gaji yang isi datanya berupa angka-angka, semuanya dijumlahkan menjadi satu seperti ditotalkan (sama seperti matematika pada umumnya). Dan hasilnya adalah 30575000. Adapun nama kolom dari hasil jumlah tersebut diubah dari SUM(Gaji) menjadi Total\_Gaji.

## 6

### Query

```
Select SUM(Gaji) AS Gaji_Manager
->FROM pegawai
->WHERE Jabatan = 'Manager';
```

### Hasil

```
MariaDB [perusahaan]> select sum(Gaji) AS Gaji_Manager
-> FROM pegawai
-> WHERE Jabatan = 'Manager';
+-----+
| Gaji_Manager |
+-----+
|          17.25 |
+-----+
1 row in set, 3 warnings (0.001 sec)
```

### Analisis

- Select= untuk memiliki kolom mana saja yang dipilih untuk digunakan.
- SUM= untuk menghitung jumlah isi data (khusus angka) pada kolom yang dipilih. gaji yang dipilih untuk dijumlahkan isi datanya.
- AS= mengganti nama dari kolom hasil SUM(Gaji) secara sementara.
- Gaji\_Manager= mengubah nama sementara dari perintah AS.
- FROM pegawai= untuk memilih dari tabel mana yang kolom datanya akan digunakan pegawai adalah nama dari tabel yang dipilih.
- WHERE= kondisi yang harus dipenuhi oleh suatu kolom agar datanya bisa dijumlah
- (Jabatan="Manajer")= merupakan kondisi dari WHERE. hanya barisan data yang kolom Gaji-nya bisa dijumlahkan.
- Hasilnya = barisan data yang kolom jumlahnya berisi manager akan dijumlah kolom Gaji-nya menjadi 17.250.000. jadi hanya beberapa kolom saja yang dijumlah.

## 7

### Query

```
Select NoCab, SUM(Gaji) TotalGaji  
->FROM pegawai  
->GROUP BY NoCab;
```

## Hasil

```
MariaDB [perusahaan]> select NoCab, sum(Gaji) TotalGaji  
-> FROM pegawai  
-> GROUP BY NoCab;  
+-----+  
| NoCab | TotalGaji |  
+-----+  
| C101  | 7.75      |  
| C102  | 9.45      |  
| C103  | 9         |  
| C104  | 4.375     |  
+-----+  
4 rows in set, 9 warnings (0.049 sec)
```

## Analisis

- SELECT= untuk memilih kolom mana saja yang dipilih untuk ditampilkan atau dijumlahkan.
- SUM(Gaji)= untuk menghitung jumlah data (khusus angka) pada kolom yang dipilih. Gaji adalah nama kolom yang dipilih untuk dijumlahkan isi datanya.
- AS= untuk mengganti nama dari kolom hasil SUM(Gaji) untuk sementara.
- TOTALGaji= merupakan nama sementara dari perintah AS.
- FROM pegawai= untuk memilih dari tabel mana yang data kolomnya akan digunakan. pegawai adalah nama tabel yang dipilih.
- GROUP BY= untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan pada kolo yang dipilih.
- NoCab= nama kolom yang datanya dipilih untuk dikelompokkan.
- Hasilnya= jadi berdasarkan kolom NoCab, barisan data yang kolom NoCab-nya berisi C102 maka kolom gaji dari barisan data itu digunakan bersama barisan data yang memiliki NoCab C101 juga. maka kolom gaji dijumlahkan sesuai dengan kolom NOCab masing-masing, mulai dari C101 memiliki 2 kolom gaji yang bisa dijumlahkan. sama denfan C103 dand C104. adapun C102 memiliki 3 kolom Gaji yanf dapat dijumlahkan. TotalGaji merupakan hasil perintah dari AS untuk mengubah nama kolom gasil dari SUM(Gaji).

## 8

## Query

```
SELECT noCab, SUM(Gaji) AS Total_Gaji from pegawai GROUP BY noCab HAVING  
SUM(Gaji) >= 8000000;
```

## Hasil

```
MariaDB [perusahaan]> SELECT AVG(Gaji) AS GajiRataManager FROM pegawai WHERE Jabatan = "Manager";  
+-----+  
| GajiRataManager |  
+-----+  
| 5750000.0000 |  
+-----+  
1 row in set (0.016 sec)
```

## Analisis

**select** = untuk memilih kolom mana saja yang dipilih untuk di tampilkan atau di jumlah.

**noCab** = nama kolom yang dipilih untuk ditampilkan.

**SUM(Gaji)** = untuk menghitung jumlah data (Khusus Angka) pada kolom yang dipilih. **Gaji** adalah nama kolom yang dipilih untuk dijumlahkan isi datanya.

**AS** = untuk mengganti nama dari SUM(Gaji) untuk sementara.

**Total Gaji** = adalah nama sementara dari perintah **AS**.

**From Pegawai** = untuk memilih dari tabel mana yang data kolomnya akan di gunakan, **Pegawai** Adalah Nama Tabel Yang akan di pilih.

**GROUP BY** = untuk mengelompokkan data berdasarkan nilai data Yang telah ditentukan Pada kolom yang dipilih.

**Nocab** = nama kolom Yang dipilih untuk datanya dikelompokkan.

**Having** = kondisi yang harus dipenuhi oleh suatu kelompok data agar bisa ditampilkan.

**(Sum (Gaji) >= 8000000)** = kondisi dari HAVING, Hasil dari Penjumlahan Gai yang hanya bisa ditampilkan adalah Hasil yang lebih dari atau sama dengan 8000000.

**Hasilnya** = Sama seperti sebelumnya, tetapi Nocab Yang memenuhi kondisi tersebut hanyalah C102 dan C103 karena hasil Jumlah kolom Gaji nya lebih dari atau sama dengan 8000000. Adapun hasil kolom Sum(Gaji) diganti Jadi Total-Gaji.

## 9

## Query

```
SELECT AVG(Gaji) AS rata_rata from pegawai;
```

## Hasil



```
MariaDB [perusahaan]> select AVG(Gaji) AS rata_rata from pegawai;
+-----+
| rata_rata |
+-----+
| 3397222.2222 |
+-----+
1 row in set (0.007 sec)
```

## Analisis

**Select** = untuk memilih kolom mana data Yang dipilih untuk ditampilkan.

**AVG (Gaji)** = untuk menghitung rata-rata dari data yang ada Pada kolom Yang dipilih.

Gaji adalah nama kolom Yang dipilih untuk dihitung rata-ratanya.

**As** = untuk mengganti nama dari kolom hasil AVG (Gaji) untuk sementara.

**Rata-rata** =nama sementara dari Perintah AS.

**From Pegawai** = untuk memilih dari tabel mana Yang data kolomnya ingin digunakan

**Pegawai** = adalah nama dari tabel yang dipilih.

**Hasilnya** = 3397222.2222 merupakan hasil rata-rata dari semua 9 barisan data Pada kolom Gaji. Adapun nama kolom hasil dari AVG (Gaji) Yaitu Rata-rata

## 10

## Query

```
SELECT AVG(Gaji) AS GajiRataManager FROM pegawai WHERE Jabatan = "Manager";
```

## Hasil

```
MariaDB [perusahaan]> select noCab, SUM(Gaji) AS Total_Gaji from pegawai GROUP BY noCab HAVING SUM(Gaji) >= 8000000;;
+-----+-----+
| noCab | Total_Gaji |
+-----+-----+
| C102 | 9450000 |
| C103 | 9000000 |
+-----+-----+
2 rows in set (0.032 sec)
```

## Analisis

**SELECT** = untuk memilih kolom mana saja yang dipilih untuk ditampilkan.

**AVG (Gaji)** = untuk menghitung rata-rata dari data yang ada pada kolom Yang dipilih. Gaji adalah nama kolom Yang dipilih untuk dihitung rata-ratanya.

**AS** = untuk mengganti nama dari kolom hasil AVG (Gaji) untuk sementara.

**GajiRataManager** = nama sementara dari Perintah AS.

**FROM pegawai** = untuk memilih dari tabel mana Yang data kolomnya ingin Pegawai adalah nama dari tabel Yang diinginkan digunakan.

**WHERE** = Kondisi yang harus dipenuhi oleh suatu kolom agar datanya bisa dihitung rata-

ratanya

(Jabatan = 'Manajer') = kondisi dari WHERE. Barisan data yang kolom Jabatannya Manajer akan dihitung rata-rata kolom Gaji nya.

Hasilnya = 5750000.0000 merupakan hasil hitung rata-rata dari barisan data yang memiliki manajer di kolom Jabatan nya, dari situ kolom Gajinya di hitung.

## Tabel Pegawai

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

idpeg	nama	namakel	jk	alamat	kota	Gaji	Jabatan	noCab
10107	Emya	Salsalina	P	Jl.suci 78	Bandung	5250000	Manager	C101
10176	Diah	Wahyuni	P	Jl.Maluku 56	Bandung	2750000	Sales	C103
10246	Dian	Anggraini	P	Jl.Mawar 5	Semarang	1750000	Staf	C102
10252	Antoni	Irawan	L	Jl.A.Yani 15	Jakarta	5750000	Manager	C102
10307	Erik	Andrian	L	Jl.Manggis 155	Semarang	2500000	Sales	C101
10314	Ayu	Rahmadani	P	Jl.Malaka 342	Jakarta	1950000	Staf	C102
10324	Martin	Susanto	L	Jl.Bima 51	Jakarta	6250000	Manager	C103
10407	Rio	Gunawan	L	Jl.Melati 356	Surabaya	2650000	Sales	C104
10415	Susan	Sumantri	P	Jl.Pahlawan 24	Surabaya	1725000	Staf	C104

9 rows in set (0.041 sec)

## 11

## Query

```
SELECT nocab, AVG(gaji) AS ratagaji FROM pegawai GROUP BY nocab;
```

## Hasil

```
MariaDB [tugas]> SELECT nocab, AVG(gaji) AS ratagaji FROM pegawai GROUP BY nocab;
```

nocab	ratagaji
C101	3875000.0000
C102	3150000.0000
C103	4500000.0000
C104	2187500.0000

4 rows in set (0.005 sec)

## Analisis

- `SELECT` untuk memilih kolom mana saja yang dipilih untuk ditampilkan/dihitung.
- `nocab` kolom yang dipilih untuk ditampilkan.
- `AVG(gaji)` untuk menghitung rata-rata dari data yang ada pada kolom yang dipilih. `gaji` adalah nama kolom yang dipilih untuk dihitung rata-ratanya.
- `AS` untuk mengganti nama dari kolom hasil `AVG(gaji)` untuk sementara.

- `RataGaji` adalah nama sementara dari perintah `AS`.
- `FROM pegawai` untuk memilih dari tabel mana yang data kolomnya ingin digunakan. `Pegawai` adalah nama dari tabel yang dipilih.
- `GROUP BY` untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan pada kolom yang dipilih.
- `nocab` nama kolom yang dipilih untuk datanya dikelompokkan.
- hasilnya hampir sama seperti no.7 masing-masing kolom `nocab` dihitung rata-ratanya. Sesuai dengan isi `nocab`, jadi yang `C101` dihitung dengan `C101` yang lainnya juga (yang sama) adapun `ratagaji` merupakan nama sementara dari kolom hasil `AVG(gaji)`.

## 12

### Query

```
SELECT nocab, AVG(gaji) AS Ratagaji FROM pegawai GROUP BY nocab HAVING nocab = 'C101' OR nocab = 'C102';
```

### Hasil

```
MariaDB [tugas]> SELECT nocab, AVG(gaji) AS Ratagaji FROM pegawai GROUP BY nocab HAVING nocab = 'C101' OR nocab = 'C102';
```

nocab	Ratagaji
C101	3875000.0000
C102	3150000.0000

```
2 rows in set (0.008 sec)
```

### Analisis

- `SELECT` untuk memilih kolom mana saja yang dipilih untuk ditampilkan/dihitung.
- `nocab` kolom yang dipilih untuk ditampilkan.
- `AVG(gaji)` untuk menghitung rata-rata dari data yang ada pada kolom yang dipilih. `gaji` adalah nama kolom yang dipilih untuk dihitung rata-ratanya.
- `AS` untuk mengganti nama dari kolom hasil `AVG(gaji)` untuk sementara.
- `RataGaji` adalah nama sementara dari perintah `AS`.
- `FROM pegawai` untuk memilih dari tabel mana yang data kolomnya ingin digunakan. `Pegawai` adalah nama dari tabel yang dipilih.
- `GROUP BY` untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan pada kolom yang dipilih.
- `nocab` nama kolom yang dipilih untuk datanya dikelompokkan.
- `HAVING` kondisi yang harus dipenuhi oleh suatu kelompok data.

- `nocab = 'C101'` OR `nocab = 'C102'` merupakan kondisi dari HAVING. Jadi kolom `nocab` yang memiliki C101 atau C102 yang hanya akan ditampilkan OR adalah kondisi yang hanya salah satu datanya yang harus dipenuhi.

---

## 13

### Query

```
SELECT MAX(gaji) AS gajiterbesar, MIN(gaji) AS gajiterkecil FROM pegawai;
```

### Hasil

```
MariaDB [tugas]> SELECT MAX(gaji) AS gajiterbesar, MIN(gaji) AS gajiterkecil FROM pegawai;
+-----+-----+
| gajiterbesar | gajiterkecil |
+-----+-----+
|      6250000 |      1725000 |
+-----+-----+
1 row in set (0.009 sec)
```

### Analisis

- `SELECT` untuk memilih kolom mana saja yang dipilih untuk ditampilkan/dihitung.
- `MAX(gaji)` untuk menampilkan nilai maksimum atau terbesar/tertinggi dari suatu data dalam kolom yang dipilih. Gaji adalah nama kolom yang dipilih.
- `AS gajiterbesar` untuk mengganti nama dari kolom hasil `MAX(gaji)` menjadi nama sementara yaitu `gajiterbesar`.
- `MIN(gaji)` untuk menampilkan nilai minimum atau terkecil/terendah dari suatu data dalam kolom yang dipilih. gaji adalah nama kolom yang dipilih.
- `AS gajiterkecil` untuk mengganti nama dari kolom hasil `MIN(gaji)` menjadi `gajiterkecil` untuk sementara.
- `FROM pegawai` untuk memilih dari tabel mana yang data kolomnya ingin digunakan. Pegawai adalah nama dari tabel yang dipilih.
- `hasilnya` jadi dari 9 nilai yang ada di kolom gaji, gaji maksimumnya adalah `6250000` dan namanya diubah menjadi `gajiterbesar`, bagi minimumnya adalah `1725000` dan namanya diubah menjadi `gajiterkecil`.

---

## 14

### Query

```
SELECT MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
FROM pegawai
WHERE Jabatan = "Manajer";
```

## Hasil

```
MariaDB [Nitip]> SELECT MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
-> FROM pegawai
-> WHERE Jabatan = "Manajer";
+-----+-----+
| GajiTerbesar | GajiTerkecil |
+-----+-----+
|      6250000 |      5250000 |
+-----+-----+
1 row in set (0.013 sec)
```

## Analisis

- `SELECT` = Untuk memilih kolom nama mana saja yang dipilih untuk ditampilkan I.
- `MAX(Gaji)` = Untuk menampilkan nilai terbesar dari suatu data dalam kolom yang dipilih. Gaji adalah nama kolom yang dipilih.
- `AS GajiTerbesar` = Untuk Menganti nama dari kolom hasil `MAX(Gaji)` menjadi `GajiTerbesar` untuk sementara.
- `MIN(Gaji)` = Untuk menampilkan nilai terkecil dari suatu data dalam kolom yang dipilih. Gaji adalah nama kolom yang dipilih.
- `AS GajiTerkecil` = Untuk Menganti nama dari kolom hasil `MIN(Gaji)` menjadi `GajiTerkecil` untuk sementara.
- `FROM pegawai` = Untuk memilih dari tabel mana yang data kolomnya ingin ditampilkan.
- `WHERE` = Kondisi yang harus dipenuhi oleh suatu kolom data agar bisa ditampilkan.
- `(Jabatan = "Manajer")` = Kondisi dari `WHERE` yang harus dipenuhi. Barisan data yang kolom jabatannya berisi manajer akan ditampilkan kolom Gajinya.
- **Hasilnya** = Jabatan Manajer yang memiliki nilai maksimum adalah 6250000 kolom hasil `MAX` nya diubah jadi `GajiTerbesar`.  
Sedangkan nilai minimumnya adalah 5250000 kolom hasil `MIN` nya diubah jadi `GajiTerkecil`

---

## 15

## Query

```
SELECT NoCab, MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
FROM pegawai
```

```
GROUP BY NoCab;
```

## Hasil

```
MariaDB [Nitip]> SELECT NoCab, MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
-> FROM pegawai
-> GROUP BY NoCab;
```

NoCab	GajiTerbesar	GajiTerkecil
C101	5250000	250000
C102	5750000	1750000
C103	6250000	2750000
C104	2650000	1725000

```
4 rows in set (0.084 sec)
```

## Analisis

- `SELECT` = Untuk memilih kolom nama mana saja yang dipilih untuk ditampilkan.
- `NoCab` = Nama kolom yang ingin ditampilkan.
- `MAX(Gaji)` = Untuk menampilkan nilai terbesar dari suatu data dalam kolom yang dipilih. `Gaji` adalah nama kolom yang dipilih.
- `AS GajiTerbesar` = Untuk Menganti nama dari kolom hasil `MAX(Gaji)` menjadi `GajiTerbesar` untuk sementara.
- `MIN(Gaji)` = Untuk menampilkan nilai terkecil dari suatu data dalam kolom yang dipilih. `Gaji` adalah nama kolom yang dipilih.
- `AS GajiTerkecil` = Untuk Menganti nama dari kolom hasil `MIN(Gaji)` menjadi `GajiTerkecil` untuk sementara.
- `FROM pegawai` = Untuk memilih dari tabel mana yang data kolomnya ingin ditampilkan.
- `GROUP BY` = Untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan pada kolom yang dipilih.
- `NoCab` = Nama kolom yang ingin dikelompokkan.
- **Hasilnya** = Masing-masing `NoCab` dicari nilai maksimum dan minimumnya. Mulai dari C101,C102,,C103,C104 dan nama hasil kolomnya diubah menjadi `GajiTerbesar` dan `GajiTerkecil`.

## 16

## Query

```
SELECT NoCab, MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
FROM pegawai
```

```
GROUP BY NoCab HAVING COUNT(NIP) >= 3;
```

## Hasil

```
Database changed
MariaDB [Nitip]> SELECT NoCab, MAX(Gaji) AS GajiTerbesar, MIN(Gaji) AS GajiTerkecil
-> FROM pegawai
-> GROUP BY NoCab HAVING COUNT(NIP) >= 3;
+-----+-----+-----+
| NoCab | GajiTerbesar | GajiTerkecil |
+-----+-----+-----+
| C102  |      5750000 |      1750000 |
+-----+-----+-----+
1 row in set (0.011 sec)
```

## Analisis

- `SELECT` = Untuk memilih kolom nama mana saja yang dipilih untuk ditampilkan.
- `NoCab` = Nama kolom yang ingin ditampilkan.
- `MAX(Gaji)` = Untuk menampilkan nilai terbesar dari suatu data dalam kolom yang dipilih. `Gaji` adalah nama kolom yang dipilih.
- `AS GajiTerbesar` = Untuk Menganti nama dari kolom hasil `MAX(Gaji)` menjadi `GajiTerbesar` untuk sementara.
- `MIN(Gaji)` = Untuk menampilkan nilai terkecil dari suatu data dalam kolom yang dipilih. `Gaji` adalah nama kolom yang dipilih.
- `AS GajiTerkecil` = Untuk Menganti nama dari kolom hasil `MIN(Gaji)` menjadi `GajiTerkecil` untuk sementara.
- `FROM pegawai` = Untuk memilih dari tabel mana yang data kolomnya ingin ditampilkan.
- `GROUP BY` = Untuk mengelompokkan data berdasarkan nilai data yang telah ditentukan pada kolom yang dipilih.
- `NoCab` = Nama kolom yang ingin dikelompokkan.
- `HAVING` = Kondisi yang harus dipenuhi oleh suatu kelompok data.
- `(COUNT(NIP) >= 3)` = Kondisi dari `HAVING` hanya hasil hitung kolom `NIP` yang lebih dari atau sama dengan 3 yang muncul.
- **Hasilnya** = Seperti no.4, yang mempunyai hasil hitung lebih dari atau sama dengan 3 adalah `NoCab C102` . Jadi hanya itu yang dicari nilai maksimum dan minimumnya pada kolom `Gaji` .

---

## PRAKTIKUM 6

### 17

### Program

```
Select COUNT(NIP) AS JumlahPegawai,SUM(Gaji) AS totalGaji  
-> AVG (Gaji) AS RataGaji, Max(Gaji), AS GajiMaks, MIN(Gaji), AS GajiMin  
-> From Pegawai
```

## Penjelasan

- `Select` = untuk memilih kolom mana saja yang dipilih untuk di tampilkan.
- `COUNT(NIP)` = untuk menghitung jumlah barisan data yang ada pada kolom yang dipilih.
- `AS JumlahPegawai` = untuk mengganti nama kolom hasil `COUNT(NIP)` menjadi Jumlah Pegawai .
- `SUM(Gaji)` = untuk Menjumlah data yang ada pada kolom yang dipilih. `Gaji` adalah kolom yang dipilih.
- `AS TotalGaji` = untuk mengganti nama kolom hasil `sum(gaji)` menjadi total `Gaji` .
- `AVG(Gaji)` = untuk menghitung rata-ratanya suatu data dalam kolom yang dipilih. `Gaji` adalah nama kolom yang dipilih untuk dihitung.
- `AS RataGaji` = untuk mengganti nama kolom hasil `AVG(Gaji)` menjadi `RataGaji` .
- `MAX(Gaji)` = untuk menampilkan nilai terbesar dari suatu data dalam kolom yang dipilih `Gaji` adalah nama kolom yang dipilih.
- `AS Gajimaks` = untuk menampilkan nama dari kolom hasil `MAX(Gaji)` menjadi `Gajimaks` untuk sementara.
- `MIN(Gaji)` = untuk menampilkan nilai terkecil dari suatu kolom yang dipilih `Gaji` nama kolom yang dipilih.
- `AS Gaji min` = untuk mengganti nama dari kolom hasil `min(gaji)` menjadi `Gajimin` untuk sementara.
- `From Pegawai` = untuk memilih tabel mana yang dipilih untuk ditampilkan pegawai adalah nama tabel yang dipilih.
- Hasil Dihitung berapa `NIP` , dijumlahkan semua data pada kolom `Gaji` , Dihitung Rata-Rata dari kolom `Gaji` , Ditampilkan Nilai terbesar pada kolom `Gaji` , dan nilai terkecil dalam kolom `Gaji` .

## Hasil

```
MariaDB [nitip]> SELECT COUNT(NIP) AS jumlahpegawai, SUM(Gaji) AS TotalGaji,  
-> AVG(Gaji) AS RataGaji, MAX(Gaji) AS GajiMaks,MIN(Gaji) AS GajiMin  
-> From pegawai;  
+-----+-----+-----+-----+-----+  
| jumlahpegawai | TotalGaji | RataGaji | GajiMaks | GajiMin |  
+-----+-----+-----+-----+-----+  
| 9 | 30575000.00 | 3397222.22222 | 6250000.00 | 1725000.00 |  
+-----+-----+-----+-----+-----+  
1 row in set (0.422 sec)
```



```
Select COUNT(NIP) AS JumlahPegawai,SUM(Gaji) AS totalGaji
-> ,AVG (Gaji) AS RataGaji, Max(Gaji) AS GajiMaks, MIN(Gaji) AS GajiMin
-> From pegawai
-> WHERE Jabatan = 'Staf' OR Jabatan = 'Sales'
-> GROUP BY NoCab HAVING SUM(Gaji) <= 2600000;
```

## Penjelasan

- `Select` = untuk memilih kolom mana saja yang ingin digunakan.
- `COUNT(NIP)` = untuk menghitung barisan data yang ada pada kolom yang dipilih.
- `AS JumlahPegawai` = untuk mengganti nama dari kolom hasil `COUNT(NIP)` menjadi jumlah pegawai untuk sementara
- `SUM(Gaji)` = untuk Menjumlah data yang ada pada kolom yang dipilih. Gaji adalah nama kolom yang dipilih
- `AS TotalGaji` = untuk mengganti nama dari kolom hasil `Sum(Gaji)` menjadi `totalGaji` untuk sementara.
- `AVG(Gaji)` = untuk menghitung Rata-Rata dari kolom yang di pilih. Gaji adalah nama kolom yang di pilih.
- `AS Rata_gaji` = untuk mengganti nama dari kolom `AVG(Gaji)` menjadi `ratagaji` untuk sementara.
- `MAX(Gaji)` = untuk menampilkan nilai terbesar dari suatu data dalam kolom yang dipilih. Gaji adalah nama kolom yang dipilih.
- `AS Gajimaks` = untuk mengganti nama dari kolom `Max(Gaji)` menjadi `Gajimaks` sementara.
- `MIN(Gaji)` = untuk menampilkan nilai minimum dari suatu data dalam kolom yang dipilih. Gaji adalah nama kolom yang dipilih.
- `AS Gajimin` = untuk mengganti nama dari kolom `Min(Gaji)` menjadi `Gajimin` untuk sementara
- `From pegawai` = untuk memilih dari tabel mana yang datanya ingin digunakan Pegawai adalah nama tabel yang dipilih.
- `Where` = kondisi yang harus dipenuhi oleh suatu kolom
- `(jabatan = staf OR jabatan = sales)` = Kondisi dari `where` .
- jadi hanya jabatan staf atau jabatan sales, hanya salah satunya saja yang harus dipenuhi agar bisa tampil. OR (hanya salah satu kondisi yang harus dipenuhi)
- `GRUP BY` = untuk mengelompokkan data sesuai dengan kolom yang ingin di pilih
- `Nocab` = nama kolom yang di pilih untuk dikelompokkan
- `HAVING` = kondisi yang harus dipenuhi oleh suatu kelompok data agar bisa di ditampilkan
- `(Sum(Gaji)) <= 2.600.000` = kondisi dari `Having` , hanya data yang hasil jumlah gajinya kurang dari atau sama dengan `2600000` yang bisa tampil
- `Hasilnya` = ada 2 barisan data yang memenuhi kondisi `<= 2600000` Adapun kondisi `where` yang juga dipenuhi oleh barisan data tersebut. Masing-masing namanya diubah

sesuai Perintah AS .

## Hasil

```
MariaDB [nitip]> Select COUNT(NIP) AS JumlahPegawai,SUM(Gaji) AS totalGaji
-> ,AVG (Gaji) AS RataGaji, Max(Gaji) AS GajiMaks, MIN(Gaji) AS GajiMin
-> From pegawai
-> WHERE Jabatan = 'Staf' OR Jabatan = 'Sales'
-> GROUP BY NoCab HAVING SUM(Gaji) <= 2600000;
+-----+-----+-----+-----+-----+
| JumlahPegawai | totalGaji | RataGaji          | GajiMaks  | GajiMin    |
+-----+-----+-----+-----+-----+
|              1 | 2500000.00 | 2500000.000000    | 2500000.00 | 2500000.00 |
+-----+-----+-----+-----+-----+
1 row in set (0.439 sec)

MariaDB [nitip]>
```