

Nama : Fezria Mahardika

Kelas : TI.22.A.5

NIM : 312210585

LAPORAN PRAKTIKUM 4

Langkah-Langkah Praktikum Beserta Outputnya

- Membuat database

```
create database Praktikum4;
```

- Masuk ke database

```
use Praktikum4;
```

- BUAT TABLE DATA_PEGAWAI

```
CREATE TABLE data_pegawai (idpegawai VARCHAR(4) NOT NULL,nama_depan TEXT(10) NOT NULL,nama_belakang TEXT(10) NOT NULL,email TEXT(20) NOT NULL,telepon VARCHAR(12) NOT NULL,tgl_kontrak VARCHAR( 10) NOT NULL,id_job VARCHAR(5) NOT NULL,gaji VARCHAR(7) NOT NULL,tunjangan VARCHAR(6) NOT NULL);
```

OUTPUTNYA :

```
Database changed
mysql> CREATE TABLE data_pegawai (idpegawai VARCHAR(4) NOT NULL,nama_depan TEXT(10) NOT NULL,nama_belakang TEXT(10) NOT NULL,email TEXT(20) NOT NULL,telepon VARCHAR(12) NOT NULL,tgl_kontrak VARCHAR( 10) NOT NULL,id_job VARCHAR(5) NOT NULL,gaji VARCHAR(7) NOT NULL,tunjangan VARCHAR(6) NOT NULL);
Query OK, 0 rows affected (0.40 sec)

mysql> desc data_pegawai;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idpegawai | varchar(4) | NO | | NULL | |
| nama_depan | tinytext | NO | | NULL | |
| nama_belakang | tinytext | NO | | NULL | |
| email | tinytext | NO | | NULL | |
| telepon | varchar(12) | NO | | NULL | |
| tgl_kontrak | varchar(10) | NO | | NULL | |
| id_job | varchar(5) | NO | | NULL | |
| gaji | varchar(7) | NO | | NULL | |
| tunjangan | varchar(6) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.12 sec)
```

- BUAT TABLE PEGAWAI DAN ISI DATANYA SEBAGAI BERIKUT:

```
INSERT INTO data_pegawai VALUES ('E001', 'Ferry', 'gustiawan', 'ferry@yahoo.com',
'07117059004', '2005-09-01', 'L0001', '2000000', '500000'); INSERT INTO data_pegawai VALUES
('E002', 'aris', 'ginardi', 'aris@yahoo.com', '081312345678', '2006-09-01', 'L0002', '2000000',
'200000'); INSERT INTO data_pegawai VALUES ('E003', 'faiz', 'ahnad', 'faiz@gmail.com',
'081367384322', '2006-10-01', 'L0003', '1500000', 'NULL'); INSERT INTO data_pegawai VALUES
('E004', 'emna', 'bunton', 'emna@gmail.com', '081363484342', '2006-10-01', 'L0004', '1500000',
'9'); INSERT INTO data_pegawai VALUES ('E005', 'mike', 'scolf', 'mike@plasa.com', '08163454555',
'2007-09-01', 'L0005', '1250000', '9'); INSERT INTO data_pegawai VALUES ('E006', 'lincoln',
'borrows', 'linc@yahoo.com', '08527388432', '2008-09-01', 'L0006', '1750000', 'NULL');
```

OUTPUTNYA:

```
MySQL 8.0 Command Line Client - Unicode
Query OK, 1 row affected (0.11 sec)
Query OK, 1 row affected (0.09 sec)
mysql> select *from data_pegawai;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| idpegawai | nama_depan | nama_belakang | email | telepon | tgl_kontrak | id_job | gaji | tunjangan |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| E001 | Ferry | gustiawan | ferry@yahoo.com | 07117059004 | 2005-09-01 | L0001 | 2000000 | 500000 |
| E002 | aris | ginardi | aris@yahoo.com | 081312345678 | 2006-09-01 | L0002 | 2000000 | 200000 |
| E003 | faiz | ahnad | faiz@gmail.com | 081367384322 | 2006-10-01 | L0003 | 1500000 | NULL |
| E004 | emna | bunton | emna@gmail.com | 081363484342 | 2006-10-01 | L0004 | 1500000 | 9 |
| E005 | mike | scolf | mike@plasa.com | 08163454555 | 2007-09-01 | L0005 | 1250000 | 9 |
| E006 | lincoln | borrows | linc@yahoo.com | 08527388432 | 2008-09-01 | L0006 | 1750000 | NULL |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> _
```

TUGAS PRAKTIKUM :

1. Tampilkan pegawai yang gajinya bukan 2.000.000 dan 1.250.000

```
SELECT * FROM data_pegawai WHERE gaji<>'2000000' AND gaji<>'1250000';
```

OUTPUTNYA:

```
6 rows in set (0.00 sec)
mysql> SELECT * FROM data_pegawai WHERE gaji<>'2000000' AND gaji<>'1250000';
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| idpegawai | nama_depan | nama_belakang | email | telepon | tgl_kontrak | id_job | gaji | tunjangan |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| E003 | faiz | ahnad | faiz@gmail.com | 081367384322 | 2006-10-01 | L0003 | 1500000 | NULL |
| E004 | emna | bunton | emna@gmail.com | 081363484342 | 2006-10-01 | L0004 | 1500000 | 9 |
| E006 | lincoln | borrows | linc@yahoo.com | 08527388432 | 2008-09-01 | L0006 | 1750000 | NULL |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> _
```

2. Tampilkan pegawai yang tunjangannya NULL

```
SELECT * FROM data_pegawai WHERE tunjangan=NULL;
```

OUTPUTNYA :

```
mysql> SELECT * FROM data_pegawai WHERE tunjangan=NULL;
```

idpegawai	nama_depan	nama_belakang	email	telepon	tgl_kontrak	id_job	gaji	tunjangan
E003	faiz	ahnad	faiz@gmail.com	081367384322	2006-10-01	L0003	1500000	NULL
E006	lincoln	borrows	linc@yahoo.com	08527388432	2008-09-01	L0006	1750000	NULL

2 rows in set (0.00 sec)

mysql>

3. Tampilkan pegawai yang tunjangannya tidak NULL

```
SELECT * FROM data_pegawai WHERE tunjangan<>'NULL';
```

OUTPUTNYA :

```
mysql> SELECT * FROM data_pegawai WHERE tunjangan<>'NULL';
```

idpegawai	nama_depan	nama_belakang	email	telepon	tgl_kontrak	id_job	gaji	tunjangan
E001	Ferry	gustiawan	ferry@yahoo.com	07117059004	2005-09-01	L0001	2000000	500000
E002	aris	ginardi	aris@yahoo.com	081312345678	2006-09-01	L0002	2000000	200000
E004	emma	bunton	enna@gmail.com	081363484342	2006-10-01	L0004	1500000	9
E005	mike	scolf	mike@plasa.com	08163454555	2007-09-01	L0005	1250000	9

4 rows in set (0.14 sec)

mysql>

4. Tampilkan/menghitung jumlah baris/record table pegawai

```
SELECT COUNT(idpegawai) FROM data_pegawai;
```

OUTPUTNYA :

```
mysql> SELECT COUNT(idpegawai) FROM data_pegawai;
```

COUNT(idpegawai)
6

1 row in set (1.42 sec)

mysql>

5. Tampilkan/hitung jumlah total gaji di tabel pegawai

```
SELECT SUM(gaji) as JUMLAH FROM data_pegawai;
```

OUTPUTNYA :

```
mysql> SELECT SUM(gaji) as JUMLAH FROM data_pegawai;
+-----+
| JUMLAH |
+-----+
| 10000000 |
+-----+
1 row in set (0.14 sec)

mysql>
```

6. Tampilkan/hitung rata-rata gaji pegawai

```
SELECT AVG(gaji) AS RATARATA FROM data_pegawai;
```

OUTPUTNYA:

```
mysql> SELECT AVG(gaji) AS RATARATA FROM data_pegawai;
+-----+
| RATARATA |
+-----+
| 1666666.666666667 |
+-----+
1 row in set (0.18 sec)

mysql>
```

7. Tampilkan gaji terkecil

```
SELECT MIN (gaji) AS Terkecil FROM data_pegawai;
```

OUTPUTNYA :

```
mysql> SELECT MIN(gaji) AS Terkecil FROM data_pegawai;
+-----+
| Terkecil |
+-----+
| 1250000 |
+-----+
1 row in set (0.20 sec)
```

8. Tampilkan gaji terbesar

```
SELECT MAX(gaji) AS Terbesar FROM data_pegawai;
```

OUTPUTNYA :

```
mysql> SELECT MAX(gaji) AS Terbesar FROM data_pegawai;
+-----+
| Terbesar |
+-----+
| 2000000 |
+-----+
1 row in set (0.00 sec)

mysql>
```

TUGAS PRAKTIKUM

LAPORAN PRAKTIKUM 4.1

Langkah-Langkah Praktikum Beserta Outputnya

- Membuat database

```
create database Praktikum4.1;
```

- Masuk ke database

```
Use Praktikum4.1;
```

- BUAT TABLE DAFTAR_HEWAN

```
CREATE TABLE daftar_hewan (id VARCHAR(5) NOT NULL,nama TEXT(8) NOT NULL,owner  
VARCHAR(7) NOT NULL,species TEXT(7) NOT NULL,sex TEXT(4) NOT NULL);
```

```
ALTER TABLE daftar_hewan ADD PRIMARY KEY(id);
```

Outputnya:

```
mysql> DESC daftar_hewan;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| id    | varchar(5)    | NO   | PRI | NULL    |       |  
| nama  | tinytext      | NO   |     | NULL    |       |  
| owner | varchar(7)    | NO   |     | NULL    |       |  
| species | tinytext    | NO   |     | NULL    |       |  
| sex   | tinytext      | NO   |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
5 rows in set (0.01 sec)  
mysql> _
```

- BUAT TABLE DAFTAR HEWAN DAN ISI DATANYA SEBAGAI BERIKUT:

```
9 OUTPUTNYA  
mysql> INSERT INTO daftar_hewan VALUES ('p1', 'Puffball', 'Diane', 'Hamster', 'f'); INSERT INTO daftar_hewan VALUES  
('p2', 'Claws', 'Gwen', 'cat', 'm'); INSERT INTO daftar_hewan VALUES ('p3', 'Fluffy', 'Haro 1d', 'cat', 'f'); INSERT  
INTO daftar_hewan VALUES ('p4', 'Buffy', 'Haro 1d', 'dog', 'f'); INSERT INTO daftar_hewan VALUES ('p5', 'Fang',  
'Benny', 'dog', 'm'); INSERT INTO daftar_hewan VALUES ('p6', 'Bowser', 'Diane', 'dog', 'm'); INSERT INTO  
daftar_hewan VALUES ('p7', 'Chirpy', 'Gwen', 'bird', 'f'); INSERT INTO daftar_hewan VALUES ('p8', 'Whistier',  
'Gwen', 'bird', 'NULL'); INSERT INTO daftar_hewan VALUES ('p9', 'Slim', 'Benny', 'snake', 'm');  
mysql> _
```

TUGAS PRAKTIKUM 4.1

1. Tampilkan jumlah hewan yang dimiliki setiap pemilik

```
Select owner, COUNT(id) AS JUMLAH FROM daftar_hewan GROUP BY owner;
```

OUPUTNYA:

```
mysql> Select owner, COUNT(id) AS JUMLAH FROM daftar_hewan GROUP BY owner;
+-----+-----+
| owner | JUMLAH |
+-----+-----+
| Diane |      2 |
| Gwen  |      3 |
| Haro Id |      2 |
| Benny |      2 |
+-----+-----+
4 rows in set (0.02 sec)

mysql>
```

2. Tampilkan jumlah hewan berdasarkan spesies

```
SELECT species, COUNT(id) AS JUMLAH FROM daftar_hewan GROUP BY species;
```

Outputnya :

```
mysql> SELECT species, COUNT(id) AS JUMLAH FROM daftar_hewan GROUP BY species;
+-----+-----+
| species | JUMLAH |
+-----+-----+
| Hamster |      1 |
| cat     |      2 |
| dog     |      3 |
| bird    |      2 |
| snake   |      1 |
+-----+-----+
5 rows in set (0.02 sec)

mysql>
```

3. Tampilkan jumlah hewan berdasarkan jenis kelamin

```
SELECT SEX, COUNT(sex) AS JUMLAH FROM daftar_hewan GROUP BY sex;
```

Outputnya :

```
Select MySQL 8.0 Command Line Client - Unicode
mysql> SELECT SEX, COUNT(sex) AS JUMLAH FROM daftar_hewan GROUP BY sex;
+-----+-----+
| SEX | JUMLAH |
+-----+-----+
| f   |      4 |
| m   |      4 |
| NULL |      1 |
+-----+-----+
3 rows in set (0.00 sec)
```

4. Tampilkan jumlah hewan berdasarkan spesies dan jenis kelamin

```
SELECT species, sex, COUNT(*) AS JUMLAH FROM daftar_hewan GROUP BY species, sex;
```

Outputnya :

```
mysql> SELECT species, sex, COUNT(*) AS JUMLAH FROM daftar_hewan GROUP BY species, sex;
+-----+-----+-----+
| species | sex | JUMLAH |
+-----+-----+-----+
| Hamster | f   |      1 |
| cat     | m   |      1 |
| cat     | f   |      1 |
| dog     | f   |      1 |
| dog     | m   |      2 |
| bird    | f   |      1 |
| bird    | NULL |      1 |
| snake   | m   |      1 |
+-----+-----+-----+
8 rows in set (0.02 sec)
```

5. Tampilkan jumlah hewan berdasarkan spesies (kucing dan anjing) dan jenis kelamin

```
SELECT species, sex, COUNT(*) AS JUMLAH FROM daftar_hewan WHERE species='cat' OR
```

Outputnya :

```
mysql> SELECT species, sex, COUNT(*) AS JUMLAH FROM daftar_hewan WHERE species='cat' OR species='dog' GROUP BY species, sex;
```

species	sex	JUMLAH
cat	m	1
cat	f	1
dog	f	1
dog	m	2

4 rows in set (0.00 sec)

6. Tampilkan jumlah hewan berdasarkan jenis kelamin yang sudah diketahui

```
SELECT * FROM daftar_hewan WHERE sex<>'NULL';
```

Outputnya :

```
mysql> SELECT * FROM daftar_hewan WHERE sex<>'NULL';
```

id	nama	owner	species	sex
p1	Puffball	Diane	Hamster	f
p2	Claws	Gwen	cat	m
p3	Fluffy	Haro 1d	cat	f
p4	Buffy	Haro 1d	dog	f
p5	Fang	Benny	dog	m
p6	Bowser	Diane	dog	m
p7	Chirpy	Gwen	bird	f
p9	Slim	Benny	snake	m

8 rows in set (0.00 sec)

```
mysql>
```

EVALUASI DAN PERTANYAAN

● KESIMPULAN

Operator relasional adalah elemen sintaksis yang dapat menerima satu atau beberapa parameter input bernama atau tidak bernama dan mengembalikan tatahan hasil. Operator relasional digunakan sebagai sumber tabel dalam pertanyaan DML.

Didalam basis data query filtering sangat membantu dalam memfilter atau membandingkan dua buah nilai atau menentukan relasi atau hubungan dari dua buah operand dengan menggunakan perintah operator sebagai berikut :

= (sama dengan),
> (lebih besar),
< (kurang dari),
>= (lebih besar sama dengan),
<= (kurang dari sama dengan), dan
<> (tidak sama dengan)

Agregasi data adalah proses membawa atau mengumpulkan data dari berbagai sumber dan meringkasnya dalam bentuk yang terpadu.

Ini adalah langkah penting yang mendahului analisis data atau analisis statistik. Setelah data teragregasi dianalisis, data tersebut dapat digunakan untuk membuat intelijen bisnis yang dapat ditindaklanjuti atau memandu proses pengambilan keputusan. Beberapa perintah yang digunakan dalam melakukan proses agregasi adalah :

COUNT berfungsi untuk menghitung jumlah baris suatu kolom pada tabel,

SUM berfungsi untuk menghitung jumlah nilai suatu kolom pada table,

AVG berfungsi untuk menghitung nilai rata-rata dari nilai suatu kolom pada tabel,

MIN berfungsi untuk menghitung nilai minimal atau terkecil dari suatu kolom pada table,
MAX berfungsi untuk menghitung nilai maksimal atau terbesar dari suatu