

# MySQL

Kelompok 1

Ketua : Mahisa Aghisni Fadhli

Anggota :

1. Ibnu Rusdianto
2. Eko Agung Prasetyo
3. Ahmad Saifullah
4. R.A. Nur'aini Hamzah

Ketua : Mahisa Aghisni Fadhli



ANGGOTA



Eko Agung Prasetyo



R.A. Nur'aini Hamzah



Ahmad Saifullah



Ibnu Rusdianto

# Selasa, 3 Oktober 2023

## QUERY INSERT

Tambahkan ke tabel pasien yang meliputi (id\_pasien, nama, sex, umur, alamat, no\_hp)

```
console_1      console x
Tx: Auto ✓ ⌂ | Playground v | ⌂
1   INSERT INTO pasien (id_pasien, nama, sex, umur, alamat, no_hp)
2     VALUES (NULL, "Eko Agung Prasetyo", "L", 50, "Semarang City", 08766423),
3             (NULL, "Rani", "P", 80, "Palembang", 08566423),
4             (NULL, "Mahisa Aghisni", "L", 100, "Mojokerto", 08565423),
5             (NULL, "AHMAD SAIFULLAH", "L", 40, "Jakarta", 08166423),
6             (NULL, "IB", "L", 20, "Subang", 08966424);
7 ;
8
9   DESCRIBE pasien;
```

```
apotek> INSERT INTO pasien (id_pasien, nama, sex, umur, alamat, no_hp)
VALUES (NULL, "Eko Agung Prasetyo", "L", 50, "Semarang City", 08766423), Mahisa Aghisni Fadhli
        (NULL, "Rani", "P", 80, "Palembang", 08566423),
        (NULL, "Mahisa Aghisni", "L", 100, "Mojokerto", 08565423),
        (NULL, "AHMAD SAIFULLAH", "L", 40, "Jakarta", 08166423),
        (NULL, "IB", "L", 20, "Subang", 08966424)
[2023-10-03 19:58:41] 5 rows affected in 21 ms
apotek> SELECT * FROM pasien
```

## QUERY UPDATE

Update tabel pasien kolom umur menjadi 22 dimana id\_pasien = 5;

```
UPDATE pasien SET umur = 22 WHERE id_pasien = 5;
UPDATE pasien SET umur = 22 WHERE id_pasien = 6;
UPDATE pasien SET umur = 22 WHERE id_pasien = 7;
UPDATE pasien SET umur = 22 WHERE id_pasien = 8;
UPDATE pasien SET umur = 22 WHERE id_pasien = 9;
```

```
apotek> UPDATE pasien SET umur = 22 WHERE id_pasien = 5
[2023-10-03 20:03:24] 1 row affected in 11 ms
apotek> UPDATE pasien SET umur = 22 WHERE id_pasien = 6
[2023-10-03 20:03:58] 1 row affected in 11 ms
apotek> UPDATE pasien SET umur = 22 WHERE id_pasien = 7
[2023-10-03 20:04:00] 1 row affected in 17 ms
apotek> UPDATE pasien SET umur = 22 WHERE id_pasien = 8
[2023-10-03 20:04:01] 1 row affected in 184 ms
apotek> UPDATE pasien SET umur = 22 WHERE id_pasien = 9
[2023-10-03 20:04:03] 1 row affected in 8 ms
```

## QUERY DELETE

Hapus dari tabel pasien dimana id\_pasien = 5;

```
DELETE FROM pasien WHERE id_pasien = 5;
DELETE FROM pasien WHERE id_pasien = 6;
DELETE FROM pasien WHERE id_pasien = 7;
DELETE FROM pasien WHERE id_pasien = 8;
DELETE FROM pasien WHERE id_pasien = 9;
# DELETE FROM pasien WHERE id_pasien = 5;
```

```
apotek> DELETE FROM pasien WHERE id_pasien = 5
[2023-10-03 20:05:52] 1 row affected in 49 ms
apotek> DELETE FROM pasien WHERE id_pasien = 6
[2023-10-03 20:05:53] 1 row affected in 15 ms
apotek> DELETE FROM pasien WHERE id_pasien = 7
[2023-10-03 20:05:56] 1 row affected in 15 ms
apotek> DELETE FROM pasien WHERE id_pasien = 8
[2023-10-03 20:05:57] 1 row affected in 5 ms
```



You are screen sharing



Stop Share

## SELECT

Pilih semua dari tabel pasien;

	id_pasien	nama	sex	umur	alamat	no_hp
1	1	Heru Rusdianto	L	25	Jatihandap, Mandalajati, Bandung City,...	8713
2	2	Ibnu Rusdianto	L	20	Subang, Pusakajaya, Cigugur Kidul	8513
3	3	Ravindra El-Barra	L	5	Kecamatan Coblong, Kota Bandung, Jawa ...	8213
4	4	Kireina	P	2	Subang, Pusakajaya, Cigugur Kaler	8213
5	10	Eko Agung Prasetyo	L	50	Semarang City	876
6	11	Rani	P	80	Palembang	856
7	12	Mahisa Aghisni	L	100	Mojokerto	856
8	13	AHMAD SAIFULLAH	L	40	Jakarta	816
9	14	IB	L	20	Subang	896

### # SINTAK SELECT

```
SELECT * FROM pasien;
```

Pilih id\_pasien, nama, no\_hp dari tabel pasien;

	id_pasien	nama	no_hp
1	1	Heru Rusdianto	87134887
2	2	Ibnu Rusdianto	85134887
3	3	Ravindra El-Barra	82134867
4	4	Kireina	82134887
5	10	Eko Agung Prasetyo	8766423
6	11	Rani	8566423
7	12	Mahisa Aghisni	8565423
8	13	AHMAD SAIFULLAH	8166423
9	14	IB	8966424

```
SELECT id_pasien, nama, no_hp FROM pasien;
```

Pilih no\_hp dari tabel pasien;

	no_hp
3	82134867
4	82134887
5	8766423
6	8566423
7	8565423
8	8166423
9	8966424

```
SELECT no_hp FROM pasien;
```

Pilih sex dari tabel pasien;

	sex
1	L
2	L
3	L
4	P
5	L
6	P
7	L
8	L
9	L

```
SELECT sex FROM pasien;
```

Pilih umur, alamat dari tabel pasien;

	umur	alamat
1	25	Jatihandap, Mandalajati, Bandung City, West Java 40195
2	20	Subang, Pusakajaya, Cigugur Kidul
3	5	Kecamatan Coblong, Kota Bandung, Jawa Barat, Indonesia
4	2	Subang, Pusakajaya, Cigugur Kaler
5	50	Semarang City
6	80	Palembang
7	100	Mojokerto
8	40	Jakarta
9	20	Subang

```
SELECT umur, alamat FROM pasien;
```

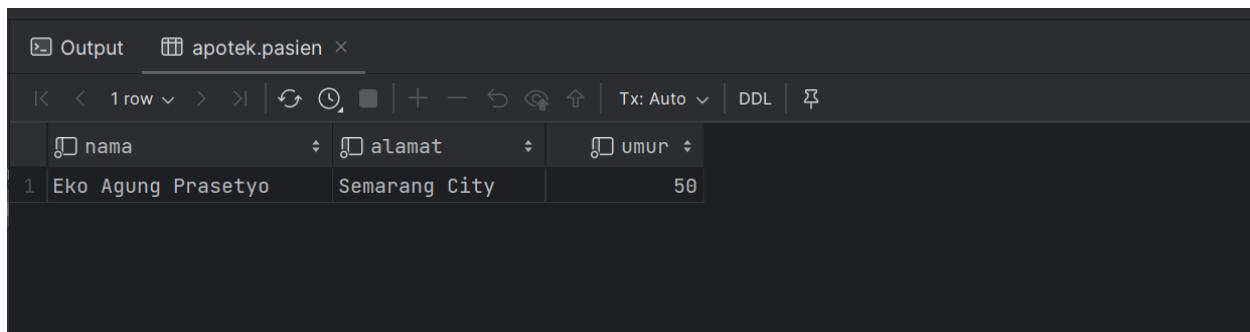
## WHERE

Pilih id\_pasien dari tabel pasien dimana pada kolom id\_pasien = 4;

	id_pasien
1	4

```
# SINTAK WHERE
SELECT id_pasien FROM pasien WHERE id_pasien = 4;
```

Pilih nama, alamat, umur dari tabel pasien dimana pada kolom id\_pasien = 10;

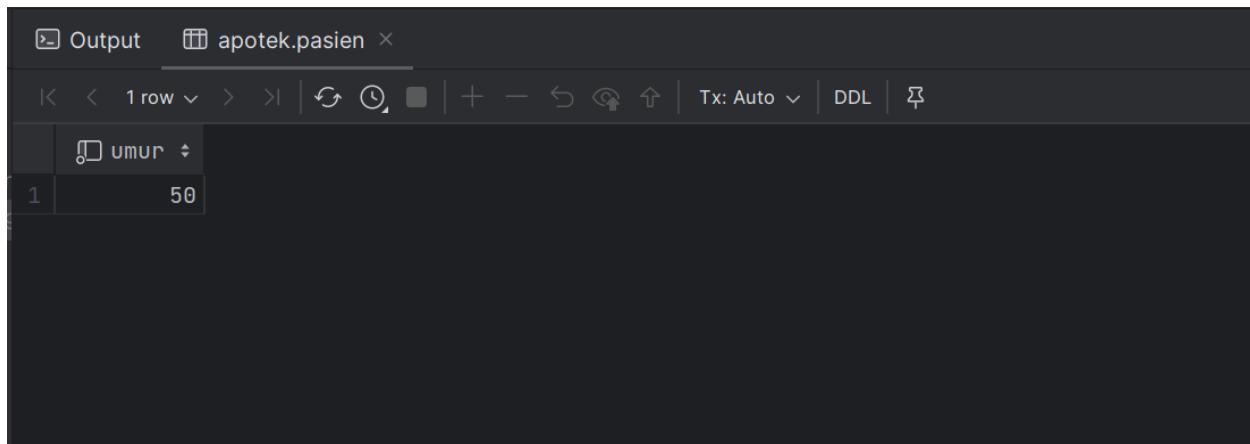


	nama	alamat	umur
1	Eko Agung Prasetyo	Semarang City	50

```
SELECT nama, alamat, umur FROM pasien WHERE id_pasien = 10;
```

Pilih umur dari tabel pasien dimana pada kolom nama = Eko Agung Prasetyo;

```
SELECT * FROM pasien;
SELECT umur FROM pasien WHERE nama = "Eko Agung Prasetyo";
```



	umur
1	50

Pilih umur dari tabel pasien dimana pada kolom nama = Mahisa Aghisni;

```
1 SELECT umur FROM pasien WHERE nama = "Eko Agung Prasetyo";
2 ✓ SELECT umur FROM pasien WHERE nama = "Mahisa Aghisni";
3
```

	umur
1	100

Pilih umur dari tabel pasien dimana pada kolom nama = AHMAD SAIFULLAH;

```
43 ✓ | SELECT umur FROM pasien WHERE nama = "AHMAD SAIFULLAH";  
44  
45
```

	umur
1	40

## AND

Pilih nama, sex dari tabel pasien yang mana umur = 20 dan alamat = "Subang";

```
SELECT nama, sex FROM pasien WHERE umur = 20 AND alamat = "Subang";
SELECT nama, sex FROM pasien WHERE umur = 100 AND alamat = "Mojokerto";
SELECT nama, sex FROM pasien WHERE umur = 40 AND alamat = "Jakarta";
SELECT nama, sex FROM pasien WHERE umur = 80 AND alamat = "Palembang";
SELECT nama, sex FROM pasien WHERE umur = 50 AND alamat = "Semarang City";
```

	Output	apotek.pasien	apotek.pasien 2	apotek.pasien 3	apotek.pasien 4	apotek.pasien 5
1	< 1 row >   ↻ ⏴   + - ⏵ ⏶ ⏷   Tx: Auto   DDL   ⚙	nama	sex			
1	IB	L				

## OR

Pilih nama, sex dari tabel pasien dimana umur = 20 atau alamat = "Subang";

```
53 ✓ SELECT nama, sex FROM pasien WHERE umur = 20 OR alamat = "Subang";
54 ✓ SELECT nama, sex FROM pasien WHERE umur = 100 OR alamat = "Mojokerto";
55 ✓ SELECT nama, sex FROM pasien WHERE umur = 40 OR alamat = "Jakarta";
56 ✓ SELECT nama, sex FROM pasien WHERE umur = 80 OR alamat = "Palembang";
57 ✓ SELECT nama, sex FROM pasien WHERE umur = 50 OR alamat = "Semarang City";
```

	Output	apotek.pasien	apotek.pasien 2	apotek.pasien 3	apotek.pasien 4	apotek.pasien 5
1	< 1 row >   ↻ ⏴   + - ⏵ ⏶ ⏷   Tx: Auto   DDL   ⚙	nama	sex			
1	Eko Agung Prasetyo	L				

## WHERE NOT

Pilih semua dari tabel pasien yang tidak memiliki alamat Subang;

```
58
59 SELECT * FROM pasien WHERE NOT alamat LIKE "%Subang%";
60 ✓ SELECT * FROM pasien WHERE NOT alamat LIKE "%Jawa Barat%";
61 SELECT * FROM pasien WHERE NOT alamat LIKE "%Jakarta%";
62 SELECT * FROM pasien WHERE NOT alamat LIKE "%Bandung%";
63 SELECT * FROM pasien WHERE NOT alamat LIKE "%Palembang%";
```

Output apotek.pasien apotek.pasien 2 apotek.pasien 3 apotek.pasien 4 apotek.pasien 5

Tx: Auto DDL CSV ↴ ↑ ↵ ⌂ ⌂

	id_pasien	nama	sex	umur	alamat	no_hp
1	Heru Rusdianto	L		25	Jatihandap, Mandalajati, Bandung City, West Java 40195	87134887
2	Ibnu Rusdianto	L		20	Subang, Pusakajaya, Cigugur Kidul	85134887
3	Ravindra El-Barra	L		5	Kecamatan Coblong, Kota Bandung, Jawa Barat, Indonesia	82134867
4	Kireina	P		2	Subang, Pusakajaya, Cigugur Kaler	82134887
5	Eko Agung Prasetyo	L		50	Semarang City	8766423
6	Mahisa Aghisni	L		100	Mojokerto	8565423
7	AHMAD SAIFULLAH	L		40	Jakarta	8166423
8	IB	L		20	Subang	8966424

You are screen sharing Stop Share

## ORDER BY

Pilih umur dari tabel pasien diurutkan dari umur yang terbesar;

console\_1 console

Tx: Auto ✓ ⌂ Playground

```

64
65 # ORDER BY
66 SELECT * FROM pasien;
67 SELECT umur FROM pasien ORDER BY umur DESC;
68 SELECT umur FROM pasien ORDER BY umur ASC;
69 SELECT nama FROM pasien ORDER BY nama ASC;
70 SELECT nama FROM pasien ORDER BY nama DESC ;
71 SELECT alamat FROM pasien ORDER BY alamat ASC ;
72 ✓ SELECT alamat FROM pasien ORDER BY alamat DESC ;
73

```

Output apotek.pasien

Tx: Auto DDL

alamat
Subang, Pusakajaya, Cigugur Kidul
Subang, Pusakajaya, Cigugur Kaler
Subang
Semarang City
Palembang
Mojokerto
Kecamatan Coblong, Kota Bandung, Jawa Barat, Indonesia
Jatihandap, Mandalajati, Bandung City, West Java 40195
Jakarta

## MIN

Pilih min(harga\_obat) dari tabel orders;

```
76    SELECT min(harga_obat) FROM orders;
77    SELECT min(id_order) FROM orders;
78    SELECT min(id_obat) FROM orders;
79    SELECT min(id_pasien) FROM orders;
80 ✓   SELECT min(tgl_order) FROM orders;
81
```

	Output	min(harga_obat):varchar	min(id_order):int(11)	min(id_obat):int(11)	min(id_pasien):int(11)	min(tgl_order):date
1	1 row	1	1	1	1	1
1	min(`min(tgl_order)`)					2023-09-01

## MAX

Pilih max(harga\_obat) dari tabel orders;

```
81
82    SELECT max(harga_obat) FROM orders;
83    SELECT max(id_order) FROM orders;
84    SELECT max(id_obat) FROM orders;
85    SELECT max(id_pasien) FROM orders;
86    SELECT max(tgl_order) FROM orders;
87
```

	Output	max(harga_obat):varchar	max(id_order):int(11)	max(id_obat):int(11)	max(id_pasien):int(11)	max(tgl_order):date
1	1 row	1	1	1	1	1
1	max(`max(tgl_order)`)					2023-09-28

## COUNT

Pilih jumlah semua dari tabel pasien;

```
87
88     SELECT COUNT(*) FROM pasien;
89     SELECT COUNT(*) FROM pasien where sex = "L";
90     SELECT COUNT(*) FROM pasien where sex = "P";
91     SELECT COUNT(*) FROM pasien where sex = "E";
92 ✓    SELECT COUNT(umur) FROM pasien where alamat LIKE "%Mojokerto%";
93     SELECT * FROM pasien;
```

	id_pasien	nama	sex	umur	alamat	no_hp
1	1	Heru Rusdianto	L	25	Jatihandap, Mandalajati, Bandung City, West Java 40195	87134887
2	2	Ibnu Rusdianto	L	20	Subang, Pusakajaya, Cigugur Kidul	85134887
3	3	Ravindra El-Barra	L	5	Kecamatan Coblong, Kota Bandung, Jawa Barat, Indonesia	82134887
4	4	Kireina	P	2	Subang, Pusakajaya, Cigugur Kaler	82134887
5	10	Eko Agung Prasetyo				8766423

## AVG

Pilih AVG(harga\_obat) dari tabel orders;

```
SELECT AVG(harga_obat) FROM orders;
SELECT AVG(id_obat) FROM orders;
SELECT AVG(id_pasien) FROM orders;
SELECT AVG(id_order) FROM orders;
SELECT AVG(tgl_order) FROM orders;
```

	AVG(tgl_order):decimal	AVG(harga_ob
1	20230914.0000	

## SUM

Pilih SUM(harga\_obat) dari tabel orders;

```
SELECT SUM(harga_obat) FROM orders;  
SELECT SUM(id_obat) FROM orders;  
SELECT SUM(id_pasien) FROM orders;  
SELECT SUM(id_order) FROM orders;  
SELECT SUM(tgl_order) FROM orders;
```

	Output	SUM(harga_obat):varchar(255)	SUM(id_obat):int(11)	SUM(id_pasien):int(11)	SUM(id_order):int(11)	SUM(tgl_order):date
1	□ `SUM(id_order)` ◆					60692742

# Kamis, 5 Oktober

## SQL JOIN

SQL Join adalah perintah dalam SQL yang digunakan untuk menggabungkan informasi dari dua atau lebih tabel berdasarkan kolom atau key tertentu yang memiliki nilai terkait untuk mendapatkan satu set data dengan informasi lengkap. Fungsi JOIN terdiri dari beberapa jenis perintah seperti INNER JOIN, LEFT JOIN, RIGHT JOIN, dan FULL OUTER JOIN

### INNER JOIN

Showing rows 0 - 2 (3 total, Query took 0.0109 seconds.)

```
SELECT orders.* , pasien.nama FROM orders INNER JOIN pasien ON orders.id_pasien = pasien.id_pasien;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 ▾ Filter rows: Search this table Sort by key: None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	nama
2	2	1	2023-09-13	50000	Heru Rusdianto
1	1	2	2023-09-01	20000	Ibnu Rusdianto
3	3	2	2023-09-28	10000	Ibnu Rusdianto

Showing rows 0 - 2 (3 total, Query took 0.0014 seconds.)

```
SELECT petugas.id_petugas, petugas.nama, pasien.nama FROM petugas INNER JOIN pasien ON petugas.id_pasien = pasien.id_pasien;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 ▾ Filter rows: Search this table Sort by key: None ▾

Extra options

id_petugas	nama	nama
1	Silebau	Heru Rusdianto
2	Joko	Ibnu Rusdianto
3	Nabila	Ravindra El-Barra

Showing rows 0 - 6 (7 total, Query took 0.0339 seconds.)

```
SELECT * FROM orders INNER JOIN obat ON orders.id_obat = obat.id_obat;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	id_obat	nama_obat	expired_obat	jenis_obat	dosis_obat
1	1	2	2023-09-01	20000	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
2	2	1	2023-09-13	50000	2	Bodrex Strip	2024-09-10	sakit kepala	1 atau 2 tablet setiap 4-6 jam
3	3	2	2023-09-28	10000	3	Bioderma Sensibio	2024-09-10	membersihkan wajah	2
4	4	5	2023-10-02	5000	4	Cendo Lyteers	2025-09-10	mengatasi iritasi mata disebabkan mata kering	1-2 tetes dimasukkan ke dalam mata
5	1	6	2023-10-01	2500	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
6	6	7	2023-10-04	12000	6	Arcoxia	2029-09-10	mengurangi rasa sakit dan peradangan	1 x 60 mg dalam sehari
7	5	8	2023-10-03	4000	5	Vaclo 75 mg Tablet	2029-09-10	mencegah terjadinya serangan iskemik pada pasien y...	1 dos isi 5 strip x 6 tablet

Showing rows 0 - 1 (2 total, Query took 0.0013 seconds.)

```
SELECT * FROM orders INNER JOIN order_obat ON orders.id_order = order_obat.id_order;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	id_order	id_obat
1	1	2	2023-09-01	20000	1	1
2	2	1	2023-09-13	50000	2	2

Showing rows 0 - 1 (2 total, Query took 0.0017 seconds.)

```
SELECT * FROM order_details INNER JOIN obat ON order_details.id_obat = obat.id_obat;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id_order_details	id_order	id_obat	qty	id_obat	nama_obat	expired_obat	jenis_obat	dosis_obat
1	1	1	10	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
2	2	2	6000	2	Bodrex Strip	2024-09-10	sakit kepala	1 atau 2 tablet setiap 4-6 jam

## LEFT JOIN

✓ Menampilkan baris 0 - 5 (total 6, Pencarian dilakukan dalam 0,0011 detik.)

```
1 SELECT obat.id_obat, nama_obat, harga_obat
2 FROM obat
3 LEFT JOIN orders
4 ON obat.id_obat = orders.id_order;
```

Perbolehkan cek foreign key

Profil [ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Buat kode PHP ] [ Segarkan ]

Tampilkan semua | Jumlah baris: 25  Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options

id_obat	nama_obat	harga_obat
1	Obat Baru	20000
2	Bodrex Strip	50000
3	Bioderma Sensibio	10000
4	Cendo Lyteers	NULL
5	Vaclo 75 mg Tablet	NULL
6	Arcoxia	NULL

✓ Menampilkan baris 0 - 3 (total 4, Pencarian dilakukan dalam 0,0010 detik.)

```
1 SELECT id_pasien, pasien.nama, nama_obat, jenis_obat
2 FROM pasien
3 LEFT JOIN obat
4 ON pasien.id_pasien = obat.id_obat;
```

Perbolehkan cek foreign key

Profil [ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Buat kode PHP ] [ Segarkan ]

Tampilkan semua | Jumlah baris: 25  Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options

id_pasien	nama	nama_obat	jenis_obat
1	Heru Rusdianto	Obat Baru	Jenis Baru
2	Ibnu Rusdianto	Bodrex Strip	sakit kepala
3	Ravindra El-Barra	Bioderma Sensibio	membersihkan wajah
4	Kireina	Cendo Lyteers	mengatasi iritasi mata disebabkan mata kering

✓ Menampilkan baris 0 - 3 (total 4, Pencarian dilakukan dalam 0,0011 detik.)

```

1 SELECT nama, umur, alamat, tgl_order
2 FROM pasien
3 LEFT JOIN orders
4 ON pasien.id_pasien = orders.id_order;

```

Perbolehkan cek foreign key

Kirim  Batal

Profil [ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Buat kode PHP ] [ Segarkan ]

Tampilkan semua | Jumlah baris: 25  Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options

nama	umur	alamat	tgl_order
Heru Rusdianto	25	Jatihandap, Mandalajati, Bandung City, West Java 4...	2023-09-01
Ibnu Rusdianto	20	Subang, Pusakajaya, Cigugur Kidul	2023-09-13
Ravindra El-Barra	5	Kecamatan Coblong, Kota Bandung, Jawa Barat, Indon...	2023-09-28
Kireina	2	Subang, Pusakajaya, Cigugur Kaler	NULL

✓ Menampilkan baris 0 - 5 (total 6, Pencarian dilakukan dalam 0,0013 detik.)

```

1 SELECT obat.id_obat,nama_obat, qty
2 FROM obat
3 LEFT JOIN order_details
4 ON obat.id_obat = order_details.id_order_details;

```

Perbolehkan cek foreign key

Kirim  Batal

Profil [ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Buat kode PHP ] [ Segarkan ]

Tampilkan semua | Jumlah baris: 25  Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options

id_obat	nama_obat	qty
1	Obat Baru	10
2	Bodrex Strip	6000
3	Bioderma Sensibio	NULL
4	Cendo Lyteers	NULL
5	Vaclo 75 mg Tablet	NULL
6	Arcoxia	NULL

✓ Menampilkan baris 0 - 2 (total 3, Pencarian dilakukan dalam 0,0019 detik.)

```

1 SELECT orders.*, order_details.qty
2 FROM orders
3 LEFT JOIN order_details
4 ON orders.id_order = order_details.id_order;

```

Perbolehkan cek foreign key

Profil [ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Buat kode PHP ] [ Segarkan ]

Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	qty
1	1	2	2023-09-01	20000	10
2	2	1	2023-09-13	50000	6000
3	3	2	2023-09-28	10000	NULL

Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

## RIGHT JOIN

✓ Showing rows 0 - 5 (6 total, Query took 0.0094 seconds.)

```
SELECT * FROM order_details RIGHT JOIN obat ON order_details.id_obat = obat.id_obat;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id_order_details	id_order	id_obat	qty	id_obat	nama_obat	expired_obat	jenis_obat	dosis_obat
1	1	1	10	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
2	2	2	6000	2	Bodrex Strip	2024-09-10	sakit kepala	1 atau 2 tablet setiap 4-6 jam
NULL	NULL	NULL	NULL	3	Bioderma Sensibio	2024-09-10	membersihkan wajah	2
NULL	NULL	NULL	NULL	4	Cendo Lyteers	2025-09-10	mengatasi iritasi mata disebabkan mata kering	1-2 tetes dimasukkan ke dalam mata
NULL	NULL	NULL	NULL	5	Vaclo 75 mg Tablet	2029-09-10	mencegah terjadinya serangan iskemik pada pasien y...	1 dos isi 5 strip x 6 tablet
NULL	NULL	NULL	NULL	6	Arcoxia	2029-09-10	mengurangi rasa sakit dan peradangan	1 x 60 mg dalam sehari

Showing rows 0 - 8 (9 total, Query took 0.0040 seconds.)

```
SELECT orders.*, pasien.nama FROM orders RIGHT JOIN pasien ON orders.id_pasien = pasien.id_pasien;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all

Number of rows: 25 ▾

Filter rows:

Search this table

Sort by key:

None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	nama
2	2	1	2023-09-13	50000	Heru Rusdianto
1	1	2	2023-09-01	20000	Ibnu Rusdianto
3	3	2	2023-09-28	10000	Ibnu Rusdianto
NULL	NULL	NULL	NULL	NULL	Ravindra El-Barra
NULL	NULL	NULL	NULL	NULL	Kireina
4	4	5	2023-10-02	5000	Mahisa Aghisni
5	1	6	2023-10-01	2500	R.A. Nur'aini Hamzah
6	6	7	2023-10-04	12000	Eko Agung
7	5	8	2023-10-03	4000	Ahmad Saifullah

Showing rows 0 - 7 (8 total, Query took 0.0030 seconds.)

```
SELECT petugas.id_petugas, petugas.nama, pasien.nama FROM petugas RIGHT JOIN pasien ON petugas.id_pasien = pasien.id_pasien;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all

Number of rows: 25 ▾

Filter rows:

Search this table

Sort by key:

None

▼

Extra options

id_petugas	nama	nama
1	Silebau	Heru Rusdianto
2	Joko	Ibnu Rusdianto
3	Nabila	Ravindra El-Barra
NULL	NULL	Kireina
NULL	NULL	Mahisa Aghisni
NULL	NULL	R.A. Nur'aini Hamzah
NULL	NULL	Eko Agung
NULL	NULL	Ahmad Saifullah

Showing rows 0 - 6 (7 total, Query took 0.0031 seconds.)

```
SELECT * FROM orders RIGHT JOIN obat ON orders.id_obat = obat.id_obat;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	id_obat	nama_obat	expired_obat	jenis_obat	dosis_obat
1	1	2	2023-09-01	20000	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
5	1	6	2023-10-01	2500	1	Obat Baru	2023-10-01	Jenis Baru	Dosis Baru
2	2	1	2023-09-13	50000	2	Bodrex Strip	2024-09-10	sakit kepala	1 atau 2 tablet setiap 4-6 jam
3	3	2	2023-09-28	10000	3	Bioderma Sensibio	2024-09-10	membersihkan wajah	2
4	4	5	2023-10-02	5000	4	Cendo Lyteers	2025-09-10	mengatasi iritasi mata disebabkan mata kering	1-2 tetes dimasukkan ke dalam mata
7	5	8	2023-10-03	4000	5	Vaclo 75 mg Tablet	2029-09-10	mencegah terjadinya serangan iskemik pada pasien y...	1 dos isi 5 strip x 6 tablet
6	6	7	2023-10-04	12000	6	Arcoxia	2029-09-10	mengurangi rasa sakit dan peradangan	1 x 60 mg dalam sehari

Showing rows 0 - 1 (2 total, Query took 0.0012 seconds.)

```
SELECT * FROM orders RIGHT JOIN order_obat ON orders.id_order = order_obat.id_order;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

id_order	id_obat	id_pasien	tgl_order	harga_obat	id_order	id_obat
1	1	2	2023-09-01	20000	1	1
2	2	1	2023-09-13	50000	2	2

**Selasa, 10 Oktober 2023**

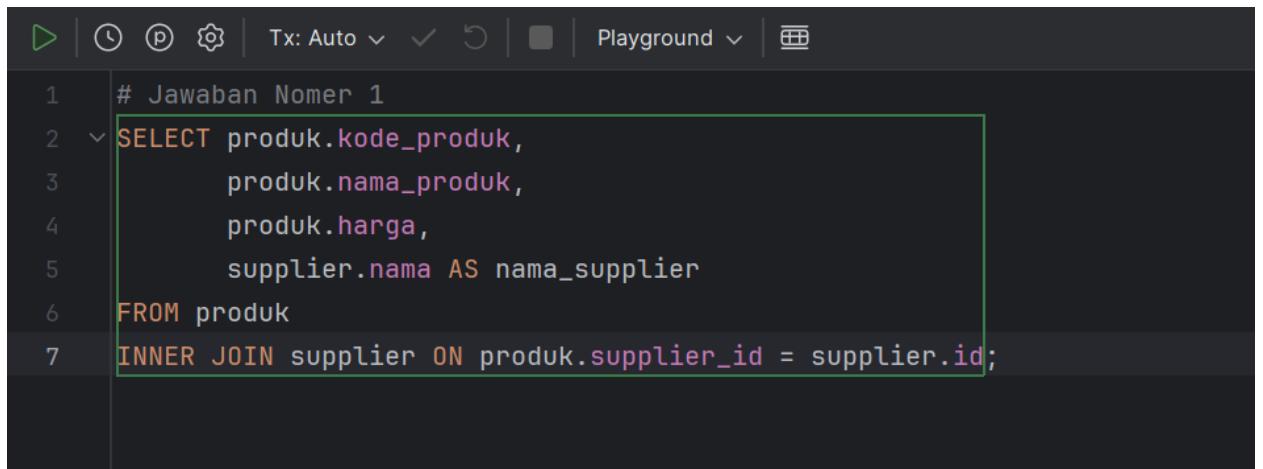
Latihan Studi Kasus



### **Pengertian MySQL**

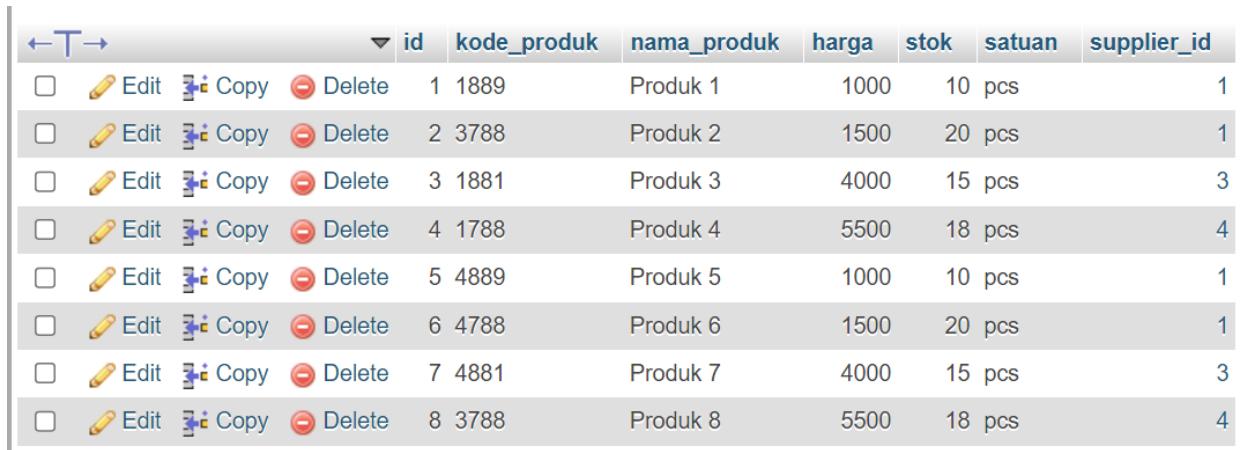
MySQL adalah suatu sistem manajemen database relasional atau Relational Database Management System (RDMS). Sistem ini bersifat open source yang dikembangkan berdasarkan pada Structured Query Language (SQL). Pada saat ini, MySQL menjadi salah satu pilihan database yang sangat populer untuk beragam tujuan, seperti untuk membuat serta mengelola database, mengelola transaksi e-commerce, penyimpanan data, pencatatan data, dan yang sangat populer yaitu sebagai database untuk sebuah website.

1. Tampilkan kode produk, nama produk, harga dan nama supplier



```
# Jawaban Nomer 1
SELECT produk.kode_produk,
       produk.nama_produk,
       produk.harga,
       supplier.nama AS nama_supplier
FROM produk
INNER JOIN supplier ON produk.supplier_id = supplier.id;
```

**Penjelasan :** Coba kita perhatikan dari soal nya, kita disuruh menampilkan kode, nama, harga dari produk dan juga nama suplier nya, okay, mari kita cek database dan lihat table produk terlebih dahulu



	<input type="checkbox"/> Edit		Delete	1 1889	Produk 1	1000	10	pcs	1	
	<input type="checkbox"/> Edit		Delete	2 3788	Produk 2	1500	20	pcs	1	
	<input type="checkbox"/> Edit		Delete	3 1881	Produk 3	4000	15	pcs	3	
	<input type="checkbox"/> Edit		Delete	4 1788	Produk 4	5500	18	pcs	4	
	<input type="checkbox"/> Edit		Delete	5 4889	Produk 5	1000	10	pcs	1	
	<input type="checkbox"/> Edit		Delete	6 4788	Produk 6	1500	20	pcs	1	
	<input type="checkbox"/> Edit		Delete	7 4881	Produk 7	4000	15	pcs	3	
	<input type="checkbox"/> Edit		Delete	8 3788	Produk 8	5500	18	pcs	4	

Coba lihat apakah di dalam table produk terdapat kolom nama supplier? (tidak ada) berarti nama supplier itu sendiri terpisah dari table produk, dia berada di table sendiri yaitu supplier, berikut menampakan dari table supplier

			<b>id</b>	<b>nama</b>	<b>telpon</b>	<b>alamat</b>
<input type="checkbox"/>	 Edit	 Copy	 Delete	1 Supplier 1	0829174111	Bandung
<input type="checkbox"/>	 Edit	 Copy	 Delete	2 Supplier 2	1937737302	Jakarta
<input type="checkbox"/>	 Edit	 Copy	 Delete	3 Supplier 3	72727199	Cimahi
<input type="checkbox"/>	 Edit	 Copy	 Delete	4 Supplier 4	447741010	Bandung

Nah ketemu, disini kita sudah terdapat nama supplier, berarti untuk menyelesaikan ini kita gunakan sebuah kondisi JOIN ke table supplier dan jangan lupa untuk kolom “supplier.nama” dikasih ALIAS (AS) nama supplier” biar sesuai dengan soal.

2. Tampilkan nama pelanggan, jenis kelamin, telpon, alamat, dengan kondisi yang alamatnya ada di Bandung

```

9 # Jawaban Nomer 2
10 < SELECT nama, jenis_kelamin, telpon, alamat
11 < FROM pelanggan
12 < WHERE alamat LIKE '%Bandung%';

```

**Penjelasan :** Coba perhatikan baik baik pada soalnya, yaitu kita disuruh menampilkan nama, jk, telpon, alamat dan kondisi pada table pelanggan, kenapa bisa tahu hanya satu table saja yaitu pelanggan? Coba perhatikan kembali apakah ada pemanggilan kolom dengan table lain contoh “supplier.telpon” tidak ada kan? Semuanya berjalan lurus ke FROM pelanggan, lalu untuk mencari alamat yang berada di bandung kita menggunakan Kondisi WHERE dan LIKE, kenapa? Karena WHERE yaitu mencari sedangkan LIKE adalah untuk mencari specified pattern di dalam kolom, pada umum nya terdapat 3 fungsi pada LIKE untuk specified pattern

di dalam kolom, baiklah mari kita lihat perbedaan nya, berikut adalah perbedaan dari LIKE operator value

```
9 # Jawaban Nomer 2
10 ✓ SELECT nama, jenis_kelamin, telpon, alamat
11 FROM pelanggan
12 WHERE alamat LIKE 'Bandung%';
```

Jika “Bandung%” berarti kita mencari huruf awalan bandung

Jika “%Bandung” berarti kita mencari huruf akhiran bandung.

Sedangkan jika “%Bandung%” yaitu kita mencari keseluruhan yg mengandung kata atau deskripsi bandung.

Jadi dalam tugas 2 itu kita disuruh semua menampilkan yang beralamat di bandung jadi kita gunakan “%Bandung%”

3. tampilkan tanggal penjualan, keterangan dan nama pelanggan

```
4 # Jawaban Nomer 3
5 ✓ SELECT penjualan.tanggal, penjualan.keterangan, pelanggan.nama
6 FROM penjualan
7 INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id;
```

**Penjelasan :** Seperti biasa ya kita harus perhatikan terlebih dahulu dari soal yang ingin kita selesaikan, nah untuk nomer 3 ini kita bisa baca dari soalnya ada 2 table yaitu pelanggan dan penjualan, tetapi disini yang dilakukan join yaitu table pelanggan, kenapa? Coba perhatikan kata kata “Tampilkan tanggal penjualan, keterangan dan nama pelanggan” berarti kita disuruh untuk menampilkan tanggal, keterangan dari penjualan lalu JOIN ke pelanggan atau menggabungkan table pelanggan tetapi hanya 1 kolom yaitu nama pelanggan.

4. tampilkan tanggal penjualan, keterangan dan nama pelanggan dengan kondisi yang melakukan transaksi pada bulan september

```
# Jawaban Nomer 4
✓  SELECT penjualan.tanggal, penjualan.keterangan, pelanggan.nama
    FROM penjualan
    INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id
    WHERE MONTH(penjualan.tanggal) = 9;
```

**Penjelasan :** Sepertinya ini sama aja seperti kasus pada nomer 3, tetapi yang membedakan yaitu disini kita disuruh mencari riwayat transaksi yang hanya pada bulan september = 9. Jadi untuk menyelesaikan ini sama seperti nomer 3 tetapi ditambahkan WHERE MONTH CLAUSE, seperti gambar diatas.

5. tampilkan id penjualan, tanggal penjualan, produk id, kuantitas dan nama produk

```
25      # Jawaban Nomer 5
26  ✓  SELECT penjualan_detail.id AS id_penjualan,
27          penjualan.tanggal,
28          penjualan_detail.produk_id,
29          penjualan_detail.kuantitas,
30          produk.nama_produk
31  FROM penjualan_detail
32  INNER JOIN penjualan ON penjualan_detail.penjualan_id = penjualan.id
33  INNER JOIN produk ON penjualan_detail.produk_id = produk.id;
```

**Penjelasan :** Untuk nomer 5 ini kita perlu JOIN 2 table yaitu penjualan dan produk, kenapa? Yuk kita lihat pada table masing masing, yg pertama kita lihat pada table detail penjualan

	<input type="button" value="←"/>	<input type="button" value="→"/>		id	penjualan_id	produk_id	kuantitas	total
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	1	4	1	200
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	1	1	1	200
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	2	5	1	1000
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	3	8	2	1000
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	4	6	1	2000

Bisa kita lihat diatas, di dalam table penjualan detail kita terdapat beberapa kolom yaitu penjualan\_id, produk\_id dan kuantitas yang masuk pada soal, sedangkan yang lainnya tidak ada, mari kita simak table penjualan

← T →		▼	id	tanggal	keterangan	pelanggan_id	total	
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	2021-08-11 -		2	1000
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	2021-08-18 -		5	1000
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	2021-09-07 -		1	1000
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	2021-09-01 -		7	1000

Di dalam table penjualan cuman hanya tanggal yang kita ingin tampilkan pada soal diatas, yg terakhir kita cek coba pada table produk

Produk				id	kode_produk	nama_produk	harga	stok	satuan	supplier_id
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	1889	Produk 1	1000	10	pcs	1
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	3788	Produk 2	1500	20	pcs	1
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	1881	Produk 3	4000	15	pcs	3
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	1788	Produk 4	5500	18	pcs	4
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	4889	Produk 5	1000	10	pcs	1
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	4788	Produk 6	1500	20	pcs	1
<input type="checkbox"/>	 Edit	 Copy	 Delete	7	4881	Produk 7	4000	15	pcs	3
<input type="checkbox"/>	 Edit	 Copy	 Delete	8	3788	Produk 8	5500	18	pcs	4

Kalo kalian perhatikan pada soal “**nama produk**” nah di dalam produk ini kita hanya perlu menampilkan nama produk , karena produk\_id akan ditampilkan pada penjualan details.

6. tampilkan id penjualan, tanggal penjualan, produk id, kuantitas dan nama produk, dengan kondisi yang melakukan transaksi di bulan agustus

```
34  
35 # Jawaban Nomer 6  
36 SELECT penjualan_detail.id AS id_penjualan,  
37     penjualan.tanggal,  
38     penjualan_detail.produk_id,  
39     penjualan_detail.kuantitas,  
40     produk.nama_produk  
41 FROM penjualan_detail  
42 INNER JOIN penjualan ON penjualan_detail.penjualan_id = penjualan.id  
43 INNER JOIN produk ON penjualan_detail.produk_id = produk.id  
44 WHERE MONTH(penjualan.tanggal) = 8;
```

**Penjelasan :** untuk soal nomer 6 ini sama aja seperti nomer 5 cuman yang membedakan yaitu untuk nomer 6 ini menggunakan WHERE MONTH CLAUSE untuk mencari yang melakukan transaksi pada bulan Agustus (bulan ke 8)

7. tampilkan tanggal penjualan, keterangan, nama pelanggan, total dan metode pembayaran

```
46 # Jawaban Nomer 7  
47 SELECT penjualan.tanggal,  
48     penjualan.keterangan, pelanggan.nama,  
49     SUM(penjualan_detail.total) AS total,  
50     pembayaran.metode  
51 FROM penjualan  
52 INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id  
53 LEFT JOIN penjualan_detail ON penjualan.id = penjualan_detail.penjualan_id  
54 LEFT JOIN pembayaran ON penjualan.id = pembayaran.penjualan_id  
55 GROUP BY penjualan.id;
```

**Penjelasan :** Kita lihat sebuah studi kasus pada nomer 7 yaitu disuruh tampilkan tanggal penjualan, keterangan nama pelanggan total metode pembayaran, berarti yang pertama kita harus menggabungkan antara table Penjualan dan pelanggan, dimana pada penjualan kita panggil kolom tanggal, keterangan dan pelanggan yaitu berupa kolom nama, setelah kita

melakukan INNER JOIN kita lakukan LEFT JOIN table penjualan\_details AS (alias) total dan pembayaran, lalu setelah melakukan LEFT JOIN maka kita kelompokan hasilnya tetapi berdasarkan penjualan id

8. Tampilkan tanggal penjualan, keterangan, nama pelanggan, total dan metode pembayaran dengan kondisi tanggal bayarnya di bulan september

```
57      # Jawaban Nomer 8
58  ✓  SELECT penjualan.tanggal,
59          penjualan.keterangan, pelanggan.nama,
60          SUM(penjualan_detail.total) AS total,
61          pembayaran.metode
62  FROM penjualan
63  INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id
64  LEFT JOIN penjualan_detail ON penjualan.id = penjualan_detail.penjualan_id
65  LEFT JOIN pembayaran ON penjualan.id = pembayaran.penjualan_id
66  WHERE MONTH(pembayaran.tanggal_bayar) = 9
67  GROUP BY penjualan.id;
68
```

**Penjelasan :** Untuk kasus pada nomer 8 ini sama seperti nomer 7 tetapi yang membedakan yaitu di nomer 8 ini menggunakan WHERE CLAUSE (MONTH) dimana kita mencari kondisi tanggal bayarnya pada bulan September.

9. Tampilkan metode pembayaran dan hitung total group by dari metode

```
68
69      # Jawaban Nomer 9
70  ✓  SELECT pembayaran.metode, SUM(pembayaran.total) AS total
71  FROM pembayaran
72  GROUP BY pembayaran.metode;
73
74
```

**Penjelasan :** Coba kita perhatikan pada soal nya kita hanya perlu menampilkan gruppung berdasarkan metode pembayaran tetapi kita totalin semua dari pembayaran tersebut (SUM), jadi kita hanya perlu SELEC pem.metode, SUM(pem.total) AS total FROM pembayaran, lalu gruppung berdasarkan pem.metode

10. tampilkan tanggal bayar dan total pada table pembayaran yang di kelompokan berdasarkan tanggal bayarnya

```
73  
74      # Jawaban Nomer 10  
75      ▼ SELECT tanggal_bayar, SUM(total) AS total  
76      FROM pembayaran  
77      GROUP BY tanggal_bayar;  
78
```

**Penjelasan :** Untuk nomer 10 ini sama saja seperti nomer 9 tetapi untuk studi kasus nomer 10 ini kita disuruh tampilkan tgl bayar, dan hitung total dari semua table pembayaran lalu gruppung berdasarkan tanggal bayarnya

11. tampilkan data penjualan yg belum melakukan pembayaran

```
79      # Jawaban Nomer 11  
80      ▼ SELECT penjualan.id,  
81          penjualan.tanggal,  
82          penjualan.keterangan, pelanggan.nama,  
83          SUM(penjualan_detail.total) AS total  
84      FROM penjualan  
85      INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id  
86      LEFT JOIN penjualan_detail ON penjualan.id = penjualan_detail.penjualan_id  
87      LEFT JOIN pembayaran ON penjualan.id = pembayaran.penjualan_id  
88      WHERE pembayaran.id IS NULL  
89      GROUP BY penjualan.id;
```

**Penjelasan :** Pada studi kasus ini kita disuruh melakukan tampil data penjualan yg belum melakukan pembayaran, berarti pada logika ini kita hanya perlu mengecek terlebih dahulu pada penjualan details, berikut kolom dan isi pada table penjualan details

	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Copy	Delete	<b>id</b>	<b>penjualan_id</b>	<b>produk_id</b>	<b>kuantitas</b>	<b>total</b>
						1	1	4	1	200
						2	1	1	1	200
						3	2	5	1	1000
						4	3	8	2	1000
						6	4	6	1	2000

Kita coba lihat penjualan\_id berarti itu yang melakukan penjualan dengan details seperti di gambar diatas 1,2,3,4 , setelah itu kita cek pada table penjualan

	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Copy	Delete	<b>id</b>	<b>tanggal</b>	<b>keterangan</b>	<b>pelanggan_id</b>	<b>total</b>
						1	2021-08-11	-	2	1000
						2	2021-08-18	-	5	1000
						3	2021-09-07	-	1	1000
						4	2021-09-01	-	7	1000

Bisa kita lihat di dalam table penjualan juga terdapat id yang penjualan 1,2,3,4 oke berarti disini sama ya , langkah selanjutnya coba kita buka table pembayaran

	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Copy	Delete	<b>id</b>	<b>tanggal_bayar</b>	<b>total</b>	<b>metode</b>	<b>penjualan_id</b>
						1	2021-09-01	1000	tunai	1
						2	2021-09-01	1000	tunai	2
						3	2021-09-01	1000	tunai	3

Nah ini problem dari nomer 11 ini , berarti yang belum melakukan pembayaran itu penjualan\_id dari nomer 4 , silahkan kalian buka lagi table penjualan maka terdapat nomer 4, dan langkah terakhir kita coba lihat pada penjualan details lihat baik baik pada kolom “**penjualan\_id**” dan cari angka 4 nah di penjualan, penjualan details nya masuk tetapi si penjual belum melakukan pembayaran jadi nanti outputnya yaitu id 4, tgl 2021-09-01, nama pelanggan G, total 2000 (karena ambil jumlah dari penjualan\_details) ,

13. Tampilkan pelanggan yang pernah melaukan transaksi

```
97 # Jawaban 13
98 ✓ SELECT pelanggan.nama
99 FROM pelanggan
100 INNER JOIN penjualan ON pelanggan.id = penjualan.pelanggan_id
101 GROUP BY pelanggan.id;
```

**Penjelasan :** Coba kita perhatikan logika soal nya tampilkan pelanggan yang pernah melakukan transaksi, langka pertama kita harus select nama pelanggan dulu (karena disini logika nya kita harus menampilkan pelanggan) lalu kita gabungkan (JOIN) pada table penjualan setelah itu kita lakukan pengkelompokan berdasarkan pelanggan.id.

14. tampilkan data penjualan yang melakukan pembayaran di bulan agustus

```
an 3 # Jawaban 14
104 SELECT penjualan.id,
105     penjualan.tanggal,
106     penjualan.keterangan, pelanggan.nama,
107     SUM(penjualan_detail.total) AS total
108 FROM penjualan
109 INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id
110 INNER JOIN penjualan_detail ON penjualan.id = penjualan_detail.penjualan_id
111 INNER JOIN produk ON penjualan_detail.produk_id = produk.id
112 INNER JOIN pembayaran ON penjualan.id = pembayaran.penjualan_id
113 WHERE MONTH(pembayaran.tanggal_bayar) = 8
114 GROUP BY penjualan.id;
```

**Penjelasan :** Seperti biasa kita perhatikan sebuah logika dari soal nomer 14 ini secara baik baik, disini kita disuruh menampilkan data penjualan yang melakukan pembayaran di bulan agustus , berarti dalam kondisi ini kita perlu melakukan INNER JOIN yang lumayan banyak supaya kompleks dan sesuai, diantraranya yaitu pelanggan, penjualan\_details, produk, pembayaran, dan melakukan WHERE MONTH (Sebuah kondisi untuk pengecekan bahwa pada tgl pembayaran tersebut pada bulan agustus = 8) , setelah itu kita kelompokan (pengelompokan) berdasarkan penjualan, kenapa ? karena **tampilkan data penjualan**

15. tampilkan pelanggan yang melakukan transaksi dimana produk tersebut adalah dari supplier 1

```
116 # Jawaban 15
117 ✓ SELECT pelanggan.nama
118 FROM penjualan
119 INNER JOIN penjualan_detail ON penjualan.id = penjualan_detail.penjualan_id
120 INNER JOIN produk ON penjualan_detail.produk_id = produk.id
121 INNER JOIN pelanggan ON penjualan.pelanggan_id = pelanggan.id
122 WHERE produk.supplier_id = 1;
```

**Kamis, 12 Oktober 2023**

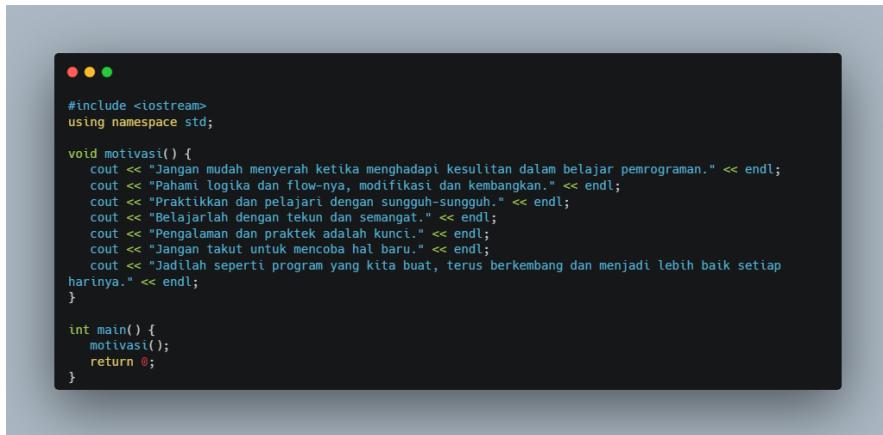
Praktek Bersama Kelompok

**Kelompok 1**



 痛 Eko Agung Prasetyo	 Mahisa Aghisni Fadhli	 R.A. Nur'aini Hamzah	 Ahmad Saifullah	 Ibnu Rusdianto
--	---	---	---	---

## Kata kata Motivasi



```
#include <iostream>
using namespace std;

void motivasi() {
    cout << "Jangan mudah menyerah ketika menghadapi kesulitan dalam belajar pemrograman." << endl;
    cout << "Pahami logika dan flow-nya, modifikasi dan kembangkan." << endl;
    cout << "Praktikkan dan pelajari dengan sungguh-sungguh." << endl;
    cout << "Belajarlah dengan tekun dan semangat." << endl;
    cout << "Pengalaman dan praktik adalah kunci." << endl;
    cout << "Jangan takut untuk mencoba hal baru." << endl;
    cout << "Jadilah seperti program yang kita buat, terus berkembang dan menjadi lebih baik setiap harinya." << endl;
}

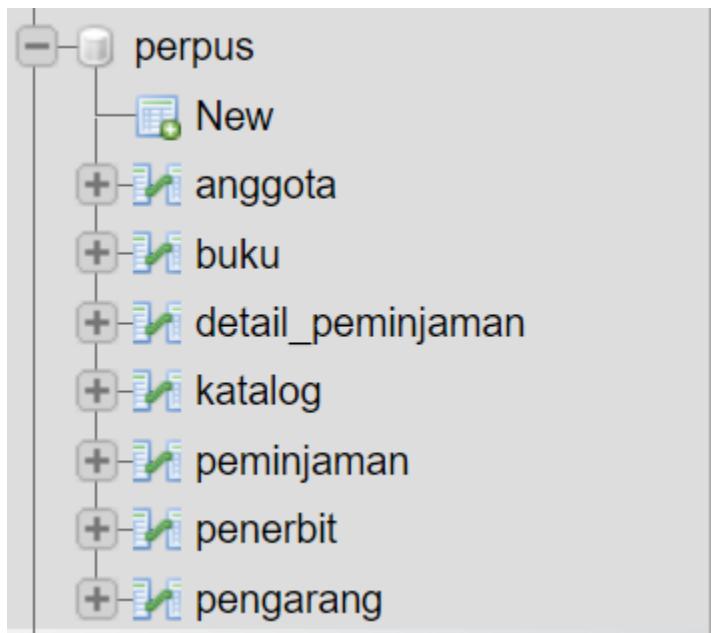
int main() {
    motivasi();
    return 0;
}
```

## Dokumentasi





## 1. Interface Database Perpus



Download perpus.sql : [perpus.sql](#)

## 2. Task Tugas Mysql Kuis

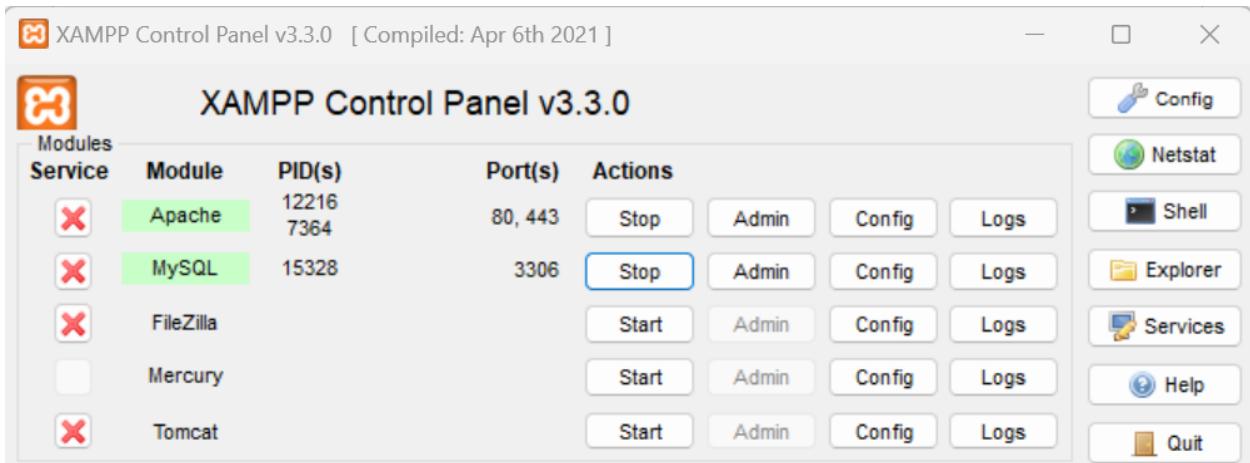
1. Tampilkan seluruh data anggota yg menjadi admin
  2. Tampilan seluruh data anggota yg tidak menjadi admin
  3. Tampilkan id dan nama anggota yg belum pernah melakukan peminjaman
  4. Tampilan id, nama, telp anggota yg pernah melakukan peminjaman
  5. Tampilkan id, nama, telp anggota yg pernah melakukan peminjaman lebih dari 1x
  6. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali
  7. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal kembalinya ada di bulan juni
  8. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal pinjamnya ada di bulan mei
  9. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal pinjam dan tanggal kembalinya ada di bulan juni
  10. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang anggotanya beralamat di Bandung
- 
11. Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang anggotanya beralamat di Bandung dan berjenis kelamin perempuan
  12. Tampilkan nama, telp, alamat, tanggal pinjam, tanggal kembali, isbn dan qty, dimana jumlah qty lebih dari 1
  13. Tampilkan nama, telp, alamat, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, harga pinjam dan total harga (qty di kali harga pinjam)
  14. Tampilkan nama anggota, telp anggota, alamat anggota, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, nama penerbit, nama pengarang dan nama katalog
  15. Tampilkan semua data katalog, judul buku, dimana semua data katalog mempunyai relasi ke data buku
  16. Tampilkan semua data buku dan nama penerbit. Beserta data buku yang tidak mempunyai relasi ke data penerbit
  17. Tampilkan ada berapa jumlah pengarang PG05 pada table buku
  18. Tampilkan data buku yang harganya lebih dari 10000
  19. Tampilkan seluruh data buku yang diterbitkan oleh Penerbit 01, dimana buku tersebut harus mempunyai qty lebih dari 10
  20. Tampilkan seluruh data anggota yang baru ditambahkan pada bulan juni

Atau bisa download di link berikut : [download](#)

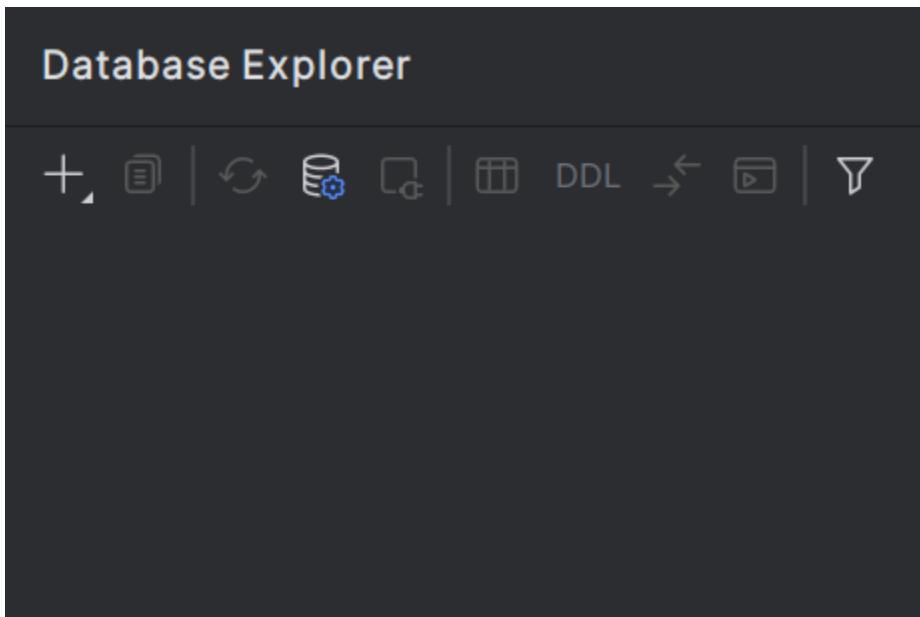
## Jawaban Task Kuis Sesi 23 - Kamis, 12 Okt 2023

1. Configurasi connect ke Mysql pada DataGrip terlebih dahulu

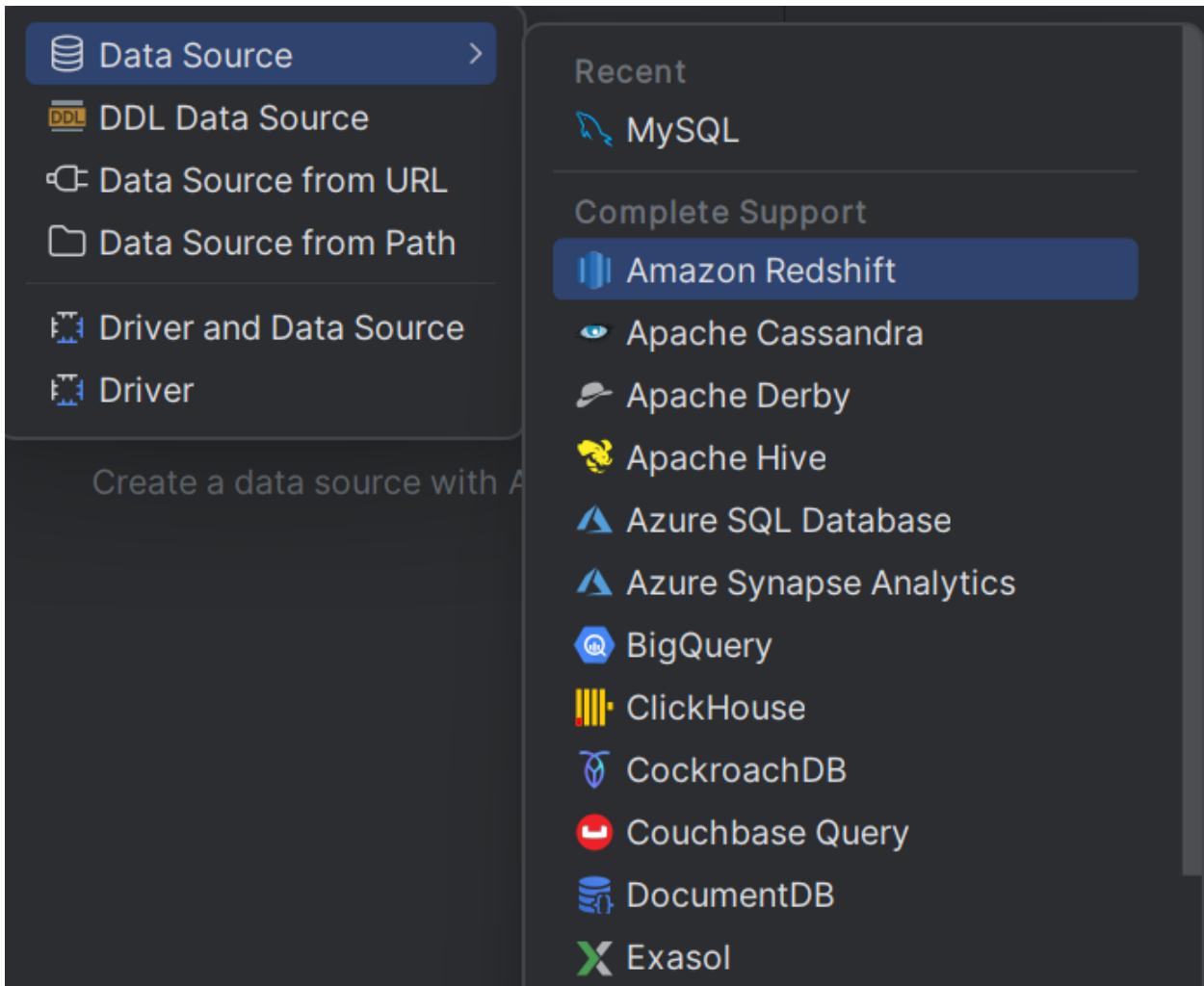
- Jalankan XAMPP dan start “Apache, Mysql”



- Setelah sudah menyalakan XAMPP dan start MySQL , Apache maka selanjutnya yaitu membuka DataGrip

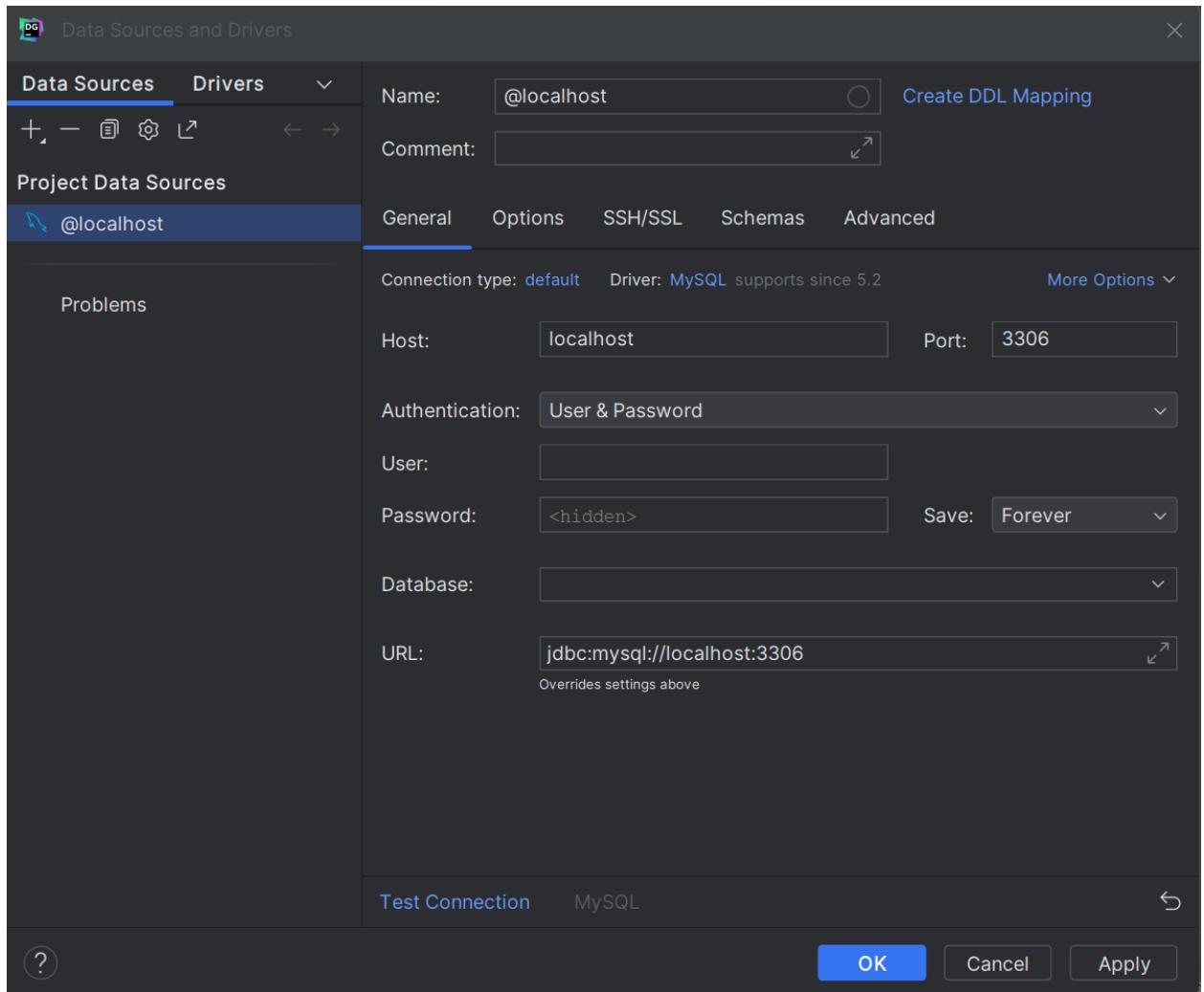


Setelah DataGrip terbuka maka di sebelah kiri ada yang namanya Database Explorer nah disitu ada tanda “+” Maka click sehingga akan memunculkan beberapa pilihan Database yang ingin kita connect, berikut contoh gambarnya



Maka akan muncul dimana ada pilihan (Data Source, DDL, etc) lalu pilih Data Source terus sebelah kanan banyak pilihan sebuah database diantaranya yaitu (Amazon RedShift, Big Query, etc) namun disini kita hanya membutuhkan database yaitu MySQL , jadi kita pilih yang Mysql

- Selanjutnya kita configurasi Data Source Dan Driver dari MySQL



Disini kita akan fokus ke configurasi Data Source

**Nama :** isi bebas namun disini kita default aja yaitu @localhost

**Comment :** kita abaikan saja

**Host :** biarkan saja localhost secara default

**Port :** 3306 (sesuai Mysql di xampp)

**User :** isi dengan “root”

**Password :** kosongin saja, dan jangan lupa save nya juga set ke “forever”

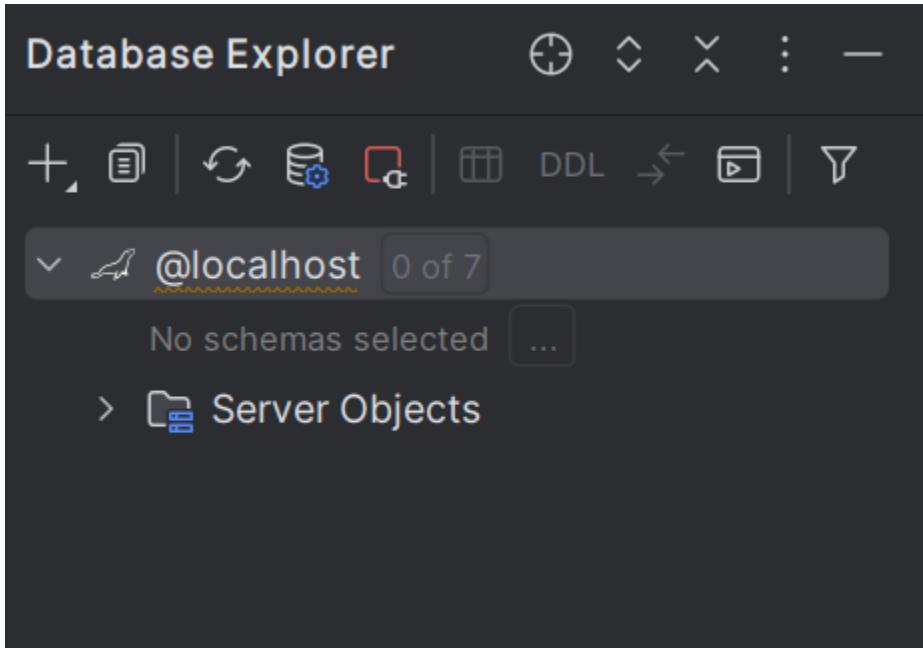
**Database :** kosongin saja karena kita sebelumnya sudah import database

perpus.sql

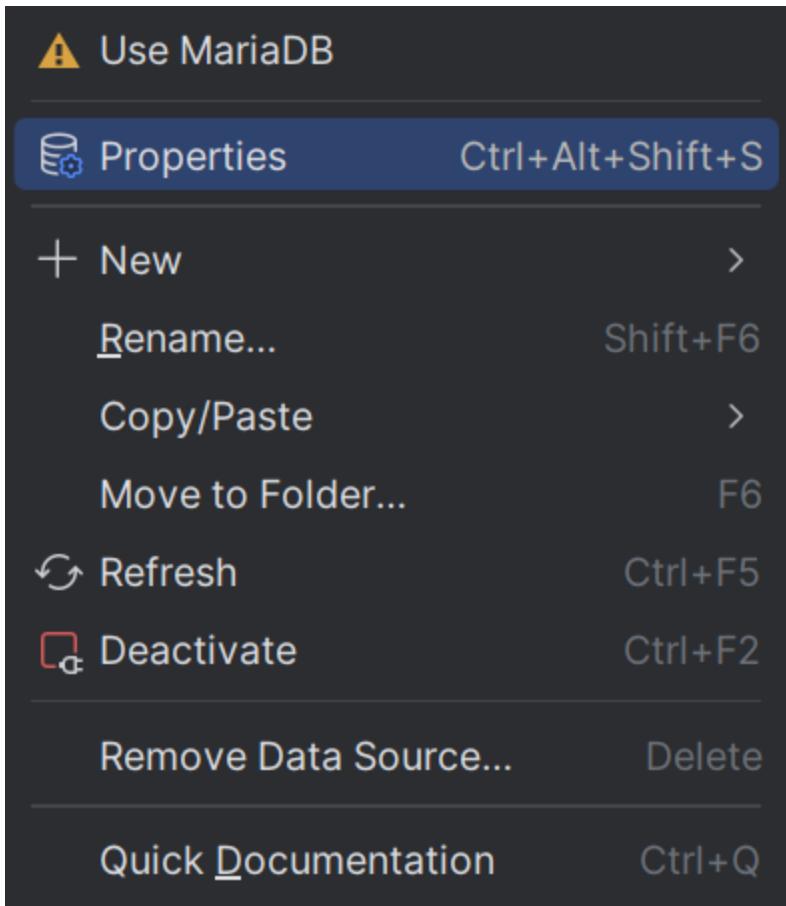
Setelah semuanya kita sudah configurasi, maka langsung saja kita **Apply** lalu

**OK**

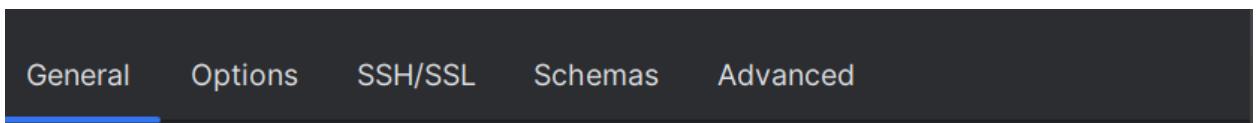
- Setelah semuanya selesai konfigurasi maka akan muncul tampilan seperti gambar berikut :



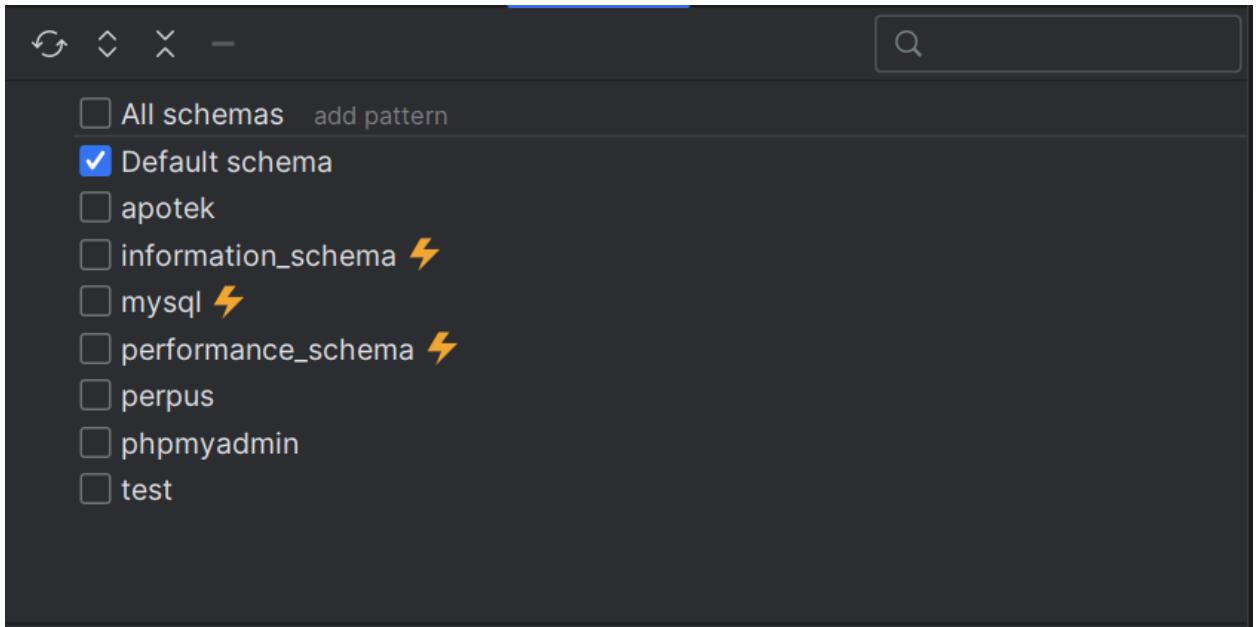
Lalu kita click kanan di @localhost sehingga akan memunculkan menu sebagai berikut :



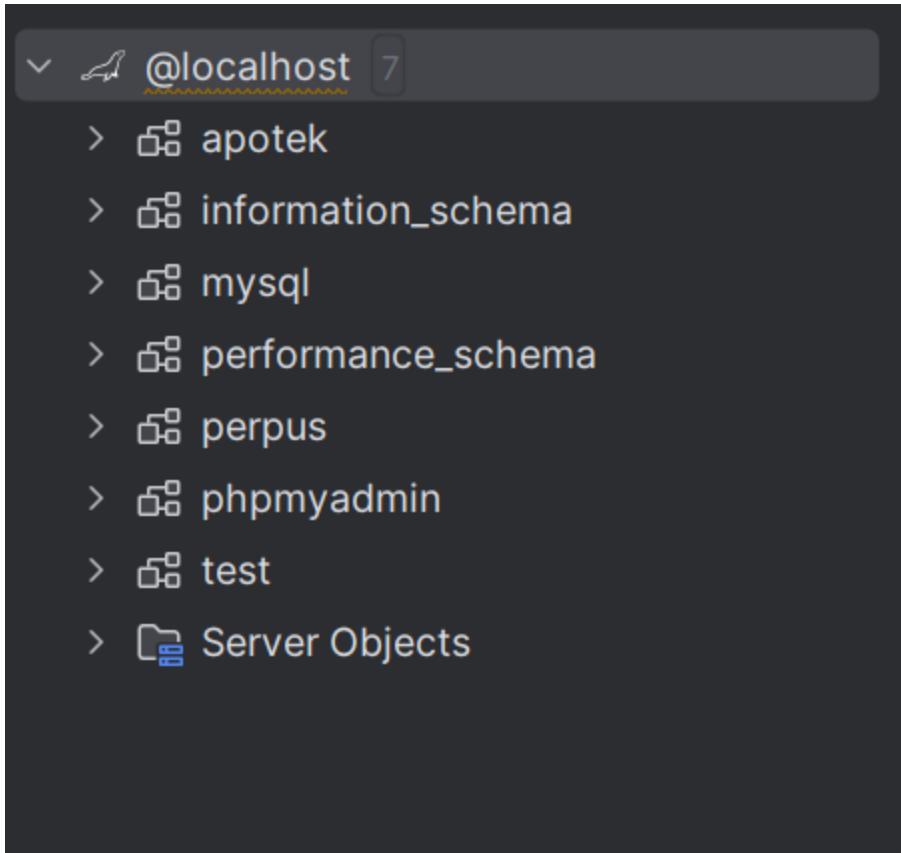
Maka Pilih Properties sehingga akan muncul seperti gambar berikut :



Lalu pilih Schemas



Bisa kita lihat , kita mempunya banyak schemas dalam database, namun kita belum melakukan panggilan semua schemas tersebut, cara agar schemas tersebut connect ke DataGrip semuanya yaitu tinggal kalian ceklis “**All Schemas**” lalu click **Apply** dan **OK**, Maka akan muncul seperti gambar dibawah ini :



Sekarang schemas kita sudah terpanggil semua, langkah selanjutnya tinggal kita melakukan console query untuk perpus dalam pengerjaan tugas KUIS dari studi kasus ini.

**Catatan :** jika kalian belum install DataGrip, bisa download di "[Many databases, one tool](#)", lalu install, dan cara untuk mendapatkan secara gratis setiap every month (bulan) kalian bisa menggunakan "[Temporary Email](#)" (gunakan setiap expired bulan habis, maka otomatis akan renew trial setiap bulan 100% tanpa verifikasi)

## 2. Mengerjakan Tugas Kuis

- 1) Tampilkan seluruh data anggota yg menjadi admin

Dari pertanyaan tersebut kita disuruh menampilkan seluruh anggota yang menjadi admin, jadi ini ada di table anggota, mari kita cek data query kolom di table anggota

console\_1 anggota

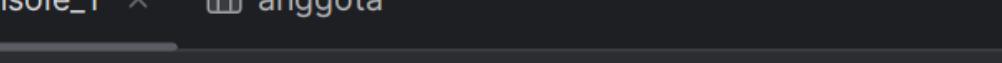
WHERE ORDER BY

	id_anggota	nama	username	password	sex	telp	alamat	e
1	1	Administrator	admin	admin	P	0891281111	Bandung	adm
2	2	Pelita	pelita	pelita	P	087821505412	Gunung Batu, Bandung	pel
3	3	Ayu	ayu	ayu	P	08112121222	Sukawarna, Bandung	ayu
4	4	Fadhli	fadhli	fadhli	L	08133613111	Cilandak, Jakarta	fad
5	5	Nur	nur	nur	P	08212221311	Sunter, Jakarta	nur
6	6	Bagus	bagus	bagus	L	0827379111	Sarijadi, Bandung	bag
7	7	Mahendra	mahendra	mahendra	P	08772191811	Sariwangi, Bandung	mah
8	8	Najmin	najmin	najmin	P	08712911991	Sukaraja, Bandung	naj
9	9	Putri	putri	putri	P	0827191811	Cimahi	put
10	10	Ridwan	ridwan	ridwan	L	0898188191	Baros, Cimahi	rid
11	11	Feby	feby	feby	P	08991717711	Sukajadi, Bandung	feb
12	12	Cindy	cindy	cindy	P	08272772791	Sentral, Cimahi	cin
13	13	Farid	farid	farid	P	0876637911	Buah Batu, Bandung	far
14	14	Bayu	bayu	bayu	L	0887639199	Sunter, Jakarta	bay
15	15	Deni	deni	deni	L	0876619111	Cikutra, Subang	den

Bisa kita lihat data dari semua kolom yang ada di table anggota yaitu role yang memiliki Admin itu cuman 1 yaitu “Administrator”

tgl_entry	role
2022-01-19 13:10:23	ADMIN
2022-01-19 13:10:23	USER

Bagaimana cara menyelesaikan logic untuk mencari seluruh data anggota mencari role “Admin”, berikut adalah sintak untuk mencari role admin dalam seluruh data kolom pada table anggota :



A screenshot of a MySQL command-line interface. The title bar says "console\_1 × anggota". The toolbar includes icons for play, pause, stop, transaction mode (Tx: Auto), commit, rollback, playground, and a grid. The main area shows a query: "SELECT \* FROM anggota WHERE role = 'ADMIN';" with a green checkmark indicating it was successful. Below the query are two empty lines.

“Sistem tolong tampilkan semua data dari table anggota dimana yang memiliki role admin” jika diterjemahkan dalam bahasa manusia, mungkin seperti itu, maka ketika kita run hasilnya akan menampilkan anggota yang memiliki role admin, berikut adalah gambar hasil injeksi querynya

	id_anggota	nama	username	password	sex	telp	alamat	email
1	1	Administrator	admin	admin	P	0891281111	Bandung	admin@gmail.com

- 2) Tampilan seluruh data anggota yg tidak menjadi admin

Dari Task ini kita disuruh untuk mencari dan menampilkan anggota yang bukan role admin, untuk menyelesaikan logic dari ini yaitu kita harus menggunakan sebuah kondisi NOT atau TIDAK SAMA DENGAN (!=), sehingga query untuk menyelesaikan masalah ini sebagai berikut :

The screenshot shows a MySQL command-line interface. The title bar has two tabs: 'console\_1' and 'anggota'. Below the title bar are several icons: a green play button, a clock, a circular arrow, a gear, a dropdown menu set to 'Tx: Auto', a checkmark, a refresh symbol, a square, another dropdown menu set to 'Playground', and a grid icon. The main area contains a code editor with the following text:

```
1 ✓ SELECT * FROM anggota WHERE role != 'ADMIN';  
2
```

“Sistem tolong tampilkan semua data pada kolom yang ada di table anggota tetapi tidak mempunya role admin (TIDAK SAMA DENGAN ADMIN !=)” maka outputnya yaitu data anggota yang bukan admin (atau tidak memiliki role admin), berikut hasilnya :

	id_anggota	nama	username	password	sex	telp	alamat	email
1	2	Pelita	pelita	pelita	P	087821505412	Gunung Batu, Bandung	pelita@gmail.com
2	3	Ayu	ayu	ayu	P	08112121222	Sukawarna, Bandung	ayu@gmail.com
3	4	Fadhli	fadhli	fadhli	L	08133613111	Cilandak, Jakarta	fadhli@gmail.com
4	5	Nur	nur	nur	P	08212221311	Sunter, Jakarta	nur@gmail.com
5	6	Bagus	bagus	bagus	L	0827379111	Sarijadi, Bandung	bagus@gmail.com
6	7	Mahendra	mahendra	mahendra	P	08772191811	Sariwangi, Bandung	mahendra@gmail.com
7	8	Najmin	najmin	najmin	P	08712911991	Sukaraja, Bandung	najmina@gmail.com
8	9	Putri	putri	putri	P	0827191811	Cimahi	putri@gmail.com
9	10	Ridwan	ridwan	ridwan	L	0898188191	Baros, Cimahi	ridwan@gmail.com

- 3) Tampilkan id dan nama anggota yg belum pernah melakukan peminjaman

untuk logic dari nomer 3 ini kita harus melakukan LEFT JOIN ke table peminjaman karena disini kita disuruh untuk mencari id\_anggota dan nama anggota yang belum pernah melakukan peminjaman, jadi secara singkat sistem kita harus melakukan pengecekan ke table peminjaman lalu cocokan dengan table anggota dimana jika di dalam table peminjaman tidak ada id\_anggota yang tertera maka itulah yang belum melakukan peminjaman. (karena dalam table peminjaman foreign key yang terhubung dengan anggota yaitu id\_anggota), berikut sql query untuk menyelesaikan logic nomer 3

```
console_1 ✘ anggota
Tx: Auto ✓ ⏪ Playground □
1 ✓ SELECT a.id_anggota, a.nama FROM anggota a
2 LEFT JOIN peminjaman p ON a.id_anggota = p.id_anggota
3 WHERE p.id_anggota IS NULL;
```

Dan Outputnya akan seperti berikut :

Output perpus.anggota

	id_anggota	nama
1	1	Administrator
2	6	Bagus
3	8	Najmin
4	9	Putri
5	12	Cindy
6	13	Farid
7	14	Bayu
8	15	Deni

- 4) Tampilan id, nama, telp anggota yg pernah melakukan peminjaman

Dalam studi kasus nomer 4 ini kebalikan dari nomer 3 kayaknya, di nomer 4 ini kita harus mencari id anggota, nama, dan telp dari anggota yang melakukan peminjaman, berarti kita harus melakukan INNER JOIN table peminjaman berdasarkan id\_anggota yang ada di table anggota dan id anggota yang ada di table peminjaman, lebih singkatnya kita akan menampilkan data kolom pada table peminjaman tetapi ditambahkan dengan nama, telp, berikut adalah sql query untuk menyelesaikan logic dari nomer 4

console\_1 x anggota

```

SELECT a.id_anggota, a.nama, a.telp
FROM anggota a
INNER JOIN peminjaman p ON a.id_anggota = p.id_anggota;

```

Maka hasilnya akan seperti berikut

	id_anggota	nama	telp
1	2	Pelita	087821505412
2	3	Ayu	08112121222
3	3	Ayu	08112121222
4	4	Fadhl	08133613111
5	4	Fadhl	08133613111
6	5	Nur	08212221311
7	5	Nur	08212221311
8	7	Mahendra	08772191811
9	10	Ridwan	0898188191
10	11	Eshy	08991717711

- 5) Tampilkan id, nama, telp anggota yg pernah melakukan peminjaman lebih dari 1x

Mari kita lihat, pada kasus ini kita disuruh untuk menampilkan hasil dari anggota yang melakukan peminjaman lebih dari 1x ya, berarti kondisi ini kita harus menampilkan 3 kolom dari anggota, dimana yaitu id,nama,telp lalu kita melakukan pengecekan atau pencocokan ke table peminjaman, mana yang  $>$  (lebih dari) 1 peminjaman, untuk menyelesaikan masalah logic ini kita membutuhkan beberapa sebuah kondisi yaitu JOIN, GROUP BY, DAN HAVING. Berikut adalah solved query untuk masalah logic nomer 5 ini

```
SELECT a.id_anggota, a.nama, a.telp
FROM anggota a
JOIN (
    SELECT id_anggota, COUNT(*) AS total_pinjaman
    FROM peminjaman
    GROUP BY id_anggota
    HAVING total_pinjaman > 1
) p ON a.id_anggota = p.id_anggota;
```

Kalo dalam bahasa manusia yaitu “Sistem tolong ambilkan semua data kolom pada table anggota dimana yaitu id, nama, telp, lalu tolong gabungkan dari table peminjaman yg memiliki foregin key ke anggota yaitu “id\_anggota” tolong hitung semua total dari AS (alias) total\_peminjaman, lalu tolong kelompokan berdasarkan id\_anggota yang mempunyai total\_peminjaman lebih dari 1 ( $> 1$ )”, maka outputnya akan seperti dibawah ini

Output perpus.anggota

	id_anggota	nama	telp
1	3	Ayu	08112121222
2	4	Fadhli	08133613111
3	5	Nur	08212221311

Kenapa bisa 3 output ? oke mari kita buka localhostnya dan kita lihat table pada peminjaman.

Extra options

	id_pinjam	id_anggota	tgl_pinjam	tgl_kembali
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	4	2021-05-26	2021-05-31
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	2021-05-27	2021-05-29
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	2021-05-10	2021-05-12
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	7	2021-05-27	2021-05-31
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	5	2021-06-01	2021-06-05
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	10	2021-06-01	2021-06-03
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	3	2021-05-04	2021-05-06
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	4	2021-06-03	2021-06-09
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	11	2021-06-02	2021-06-08
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	5	2021-05-25	2021-06-02

Kita lihat disitu ada id\_peminjam, id\_anggota, tgl\_pinjam, tgl\_kembali, kita fokus pada id\_anggota, coba perhatikan baik2, mana id\_anggota yang terdapat duplikat , jika ada yang duplikat berarti itu melakukan pinjam lebih dari satu x ( $> 1$ ),

Extra options

	<input type="button" value="←"/> <input type="button" value="→"/>		<input type="button" value="id_pinjam"/>	<input type="button" value="id_anggota"/>	<input type="button" value="tgl_pinjam"/>	<input type="button" value="tgl_kembali"/>	
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	1	4 2 3 7 5 10 1 4 11 5	2021-05-26 2021-05-27 2021-05-10 2021-05-27 2021-06-01 2021-06-01 2021-05-04 2021-06-03 2021-06-02 2021-05-25	2021-05-31 2021-05-29 2021-05-12 2021-05-31 2021-06-05 2021-06-03 2021-05-06 2021-06-09 2021-06-08 2021-06-02
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	2			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	3			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	4			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	5			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	6			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	7			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	8			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	9			
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	10			

Kita lihat yang terdapat duplikat id\_anggot yaitu, anggota dari 3,4,5

- 6) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali

Pada kasus ini kita disuruh menampilkan nama,telp,alamat,tanggal pinjam, dan tanggal\_kembali berarti kita masih berada di table anggota dan untuk menyelesaikan logic masalah ini kita masih menggunakan JOIN, berikut query untuk menyelesaikan kasus ini

```
console_1 ×
▶ | ⏴ | ⏵ | Tx: Auto ✓ | ↻ | ⏷ | Playground | ⏷
1 ✓   SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
2     FROM anggota a
3     JOIN peminjaman p ON a.id_anggota = p.id_anggota;
4
```

Kalo dalam bahasa manusia “Sistem tolong ambil nama, telp, alamat pada table anggota, lalu tolong gabungkan dengan table peminjaman dimana saya ingin ambil kolom tgl\_pinjam, tgl\_kembali” maka hasil outputnya seperti berikut :

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Fadhl	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31
2	Pelita	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29
3	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
4	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
5	Nur	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05
6	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
7	Ayu	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06
8	Fadhl	08133613111	Cilandak, Jakarta	2021-06-03	2021-06-09
9	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08
10	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02

- 7) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal kembalinya ada di bulan juni

Pada studi kasus ini kita disuruh menyelesaikan untuk menampilkan data anggota nama, telp, alamat, tgl pinjam, tgl kembali tetapi yang tgl kembalinya ada di bulan juni, nah kalian tau bulan juni itu bulan ke berapa? bagi yang belum tau bisa buka ini, nah setelah kita tau bulan juni bulan ke berapa, selanjutnya kita disini supaya menampilkan yang hanya pada bulan juni kita membutuhkan function query SQL WHERE MONTH , kalian bisa baca di link berikut [MySQL MONTH\(\) Function](#), OKAY Let's go kita selesaikan masalah ini, berikut adalah query untuk problem solving ini

```

1 ✓  SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
2   FROM anggota a
3   JOIN peminjaman p ON a.id_anggota = p.id_anggota
4 WHERE MONTH(p.tgl_kembali) = 6;
5

```

Kalo dalam bahasa manusia yaitu “Sistem tolong ambil nama, telp, alamat pada table anggota dan gabungkan ke table peminjaman dan saya ingin untuk menggabungkan data peminjaman yaitu tgl pinjam dan tgl kembali

tetapi saya ingin tgl kembali nya yg ada di bulan juni (bulan 6)" maka outputnya seperti berikut ini

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Nur	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05
2	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
3	Fadhli	08133613111	Cilandak, Jakarta	2021-06-03	2021-06-09
4	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08
5	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02

- 8) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal pinjamnya ada di bulan mei

Untuk menyelesaikan studi kasus ini, sama saja kayak nomer 7 tetapi yang membedakan hanya WHERE MONTH dimana dibagian ini adalah, angka dari WHERE MONTH untuk bulan mei ialah 5, berikut query untuk problem solving nomer 8

```
console_1 > SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
  FROM anggota a
  JOIN peminjaman p ON a.id_anggota = p.id_anggota
 WHERE MONTH(p.tgl_pinjam) = 5;
```

Sehingga outputnya sebagai berikut :

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Fadhli	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31
2	Pelita	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29
3	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
4	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
5	Ayu	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06
6	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02

- 9) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang tanggal pinjam dan tanggal kembalinya ada di bulan juni

Kita lanjut gais, di nomer 9 ini kita disuruh menampilkan tgl pinjam dan tgl kembali nya yaitu sama di bulan juni atau bulan keenam, seperti kita tahu kondisi dan itu bisa kita gunakan AND Operator [MySQL AND, OR and NOT Operators](#), okay berikut query untuk menyelesaikan problem solving dari nomer 9

```

1 ✓ | SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
2   | FROM anggota a
3   | JOIN peminjaman p ON a.id_anggota = p.id_anggota
4   | WHERE MONTH(p.tgl_pinjam) = 6 AND MONTH(p.tgl_kembali) = 6;
5

```

Sehingga outputnya sebagai berikut :

The screenshot shows a database interface with a results table titled "Result 9". The columns are labeled: nama, telp, alamat, tgl\_pinjam, and tgl\_kembali. The data consists of four rows:

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Nur	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05
2	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
3	Fadhl	08133613111	Cilandak, Jakarta	2021-06-03	2021-06-09
4	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08

- 10) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang anggotanya beralamat di Bandung

Untuk menyelesaikan nomer 10 ini kita membutuhkan function LIKE dalam database [MySQL LIKE Operator](#), dan kita juga masih menggunakan JOIN untuk table peminjaman, okay langsung saja, berikut adalah query untuk problem solving nomer 10

```

console_1 ×
SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
FROM anggota a
JOIN peminjaman p ON a.id_anggota = p.id_anggota
WHERE a.alamat LIKE '%Bandung%';
  
```

Jika kita ubah ke bahasa manusia seperti berikut “Sistem tolong ambil nama, telp, alamat pada table anggota lalu gabungkan menggunakan JOIN pada table peminjaman tetapi dalam table peminjaman ambil kolom tgl pinjam dan tgl kembali lalu carikan dimana data anggota tersebut berdomisili di bandung” sehingga outputnya seperti berikut :

The screenshot shows a database interface with a results table titled "Result 11". The table has columns: nama, telp, alamat, tgl\_pinjam, and tgl\_kembali. The data consists of 5 rows:

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Pelita	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29
2	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
3	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
4	Ayu	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06
5	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08

- 11) Tampilkan nama, telp, alamat, tanggal pinjam dan tanggal kembali yang anggotanya beralamat di Bandung dan berjenis kelamin perempuan
- Lanjuttt gaisssss, untuk menyelesaikan masalah nomer 11 ini kita hanya membutuhkan yaitu sebuah Operator Tambahan yaitu AND, mengapa? Karena kita membutuhkan untuk menampilkan alamat di bandung dan yang hanya memiliki jenis kelamin “p” perempuan. Berikut adalah query untuk menyelesaikan masalah nomer 11 ini

The screenshot shows a MySQL command-line interface with a query window titled "console\_1". The query is:

```

1 ✓ | SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali
2   | FROM anggota a
3   | JOIN peminjaman p ON a.id_anggota = p.id_anggota
4   | WHERE a.alamat LIKE '%Bandung%' AND a.sex = 'P';
5

```

The query is highlighted with a green box.

Mari kita ubah ke dalam bahasa manusia “Sistem tolong ambil data nama, telp, alamat pada table anggota lalu lakukan JOIN kepada table peminjaman dan ambil dari table peminjaman yaitu kolom tgl pinjam, tgl kembali dan saya ingin yang di tampilkan hanya berdomisili di bandung dan memiliki jenis kelamin perempuan” sehingga outputnya sebagai berikut :

	nama	telp	alamat	tgl_pinjam	tgl_kembali
1	Pelita	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29
2	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
3	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
4	Ayu	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06
5	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08

- 12) Tampilkan nama, telp, alamat, tanggal pinjam, tanggal kembali, isbn dan qty, dimana jumlah qty lebih dari 1

Okey lanjut gais, untuk nomer 12 ini kayak nya lumayan akan banyak eksekusi di query karena ini harus di lakukan JOIN 2X karena isbn dan qty ada di table details\_peminjaman sedangkan tgl pinjam dan tgl kembali ada di table peminjam, **tetapi masalah ini dominan pada details\_peminjaman**, okay langsung saja berikut untuk menyelesaikan problem solving dari nomer 12

```

console_1

SELECT a.nama, a.telp, a.alamat, p.tgl_pinjam, p.tgl_kembali, dp.isbn, dp.qty
FROM anggota a
JOIN peminjaman p ON a.id_anggota = p.id_anggota
JOIN detail_peminjaman dp ON p.id_pinjam = dp.id_pinjam
WHERE dp.qty > 1;

```

Jika kita artikan ke bahasa manusia “Sistem tolong ambilkan data nama, telp, alamat pada table anggota lalu JOIN ke table peminjaman dimana dalam table peminjaman ambil dari data tgl pinjam dan tgl kembali lalu tolong saya minta JOIN juga untuk table details\_peminjaman dimana saya ingin ambil kolom isbn dan qty tetapi saya ingin validasi bahwa qty yang ingin saya tampilkan yaitu qty yg lebih dari 1 (> 1)” Sehingga outputnya sebagai berikut :

Output Result 18

	nama	telp	alamat	tgl_pinjam	tgl_kembali	isbn	qty
1	Fadhli	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31	777-381	3
2	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12	777-381	2
3	Nur	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05	999-281	2
4	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03	377-482	2
5	Fadhli	08133613111	Cilandak, Jakarta	2021-06-03	2021-06-09	777-380	2
6	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02	977-381	2

- 13) Tampilkan nama, telp, alamat, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, harga pinjam dan total harga (qty di kali harga pinjam)

Untuk nomer 13 ini kita membutuhkan JOIN 3x table peminjaman, details\_peminjaman dan Buku, kenapa begitu ? kita lihat table peminjaman

Server: 127.0.0.1 » Database: perpus » Table: peminjaman

Browse Structure SQL Search Insert Export Import Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows: Search this table Sort by key:

Extra options

	id_pinjam	id_anggota	tgl_pinjam	tgl_kembali
<input type="checkbox"/>	1	4	2021-05-26	2021-05-31
<input type="checkbox"/>	2	2	2021-05-27	2021-05-29
<input type="checkbox"/>	3	3	2021-05-10	2021-05-12
<input type="checkbox"/>	4	7	2021-05-27	2021-05-31
<input type="checkbox"/>	5	5	2021-06-01	2021-06-05
<input type="checkbox"/>	6	10	2021-06-01	2021-06-03
<input type="checkbox"/>	7	3	2021-05-04	2021-05-06
<input type="checkbox"/>	8	4	2021-06-03	2021-06-09
<input type="checkbox"/>	9	11	2021-06-02	2021-06-08
<input type="checkbox"/>	10	5	2021-05-25	2021-06-02

Bisa kita lihat diatas ya , kita butuh peminjaman karena kita mau ambil kolom tgl pinjam dan tgl kembali, okey lanjut ke table kedua yaitu details peminjaman

The screenshot shows the phpMyAdmin interface for the 'detail\_peminjaman' table. The top navigation bar includes 'Server: 127.0.0.1', 'Database: perpus', and 'Table: detail\_peminjaman'. Below the navigation are tabs for 'Browse' (selected), 'Structure', 'SQL', 'Search', 'Insert', 'Export', and 'Import'. A green status bar at the top says 'Showing rows 0 - 24 (26 total, Query took 0.0007 seconds.)'. The SQL query shown is 'SELECT \* FROM `detail\_peminjaman`'. Below the query are options for Profiling, Edit inline, Edit, Explain SQL, Create PHP code, and Refresh. The results table has columns: id\_pinjam, isbn, and qty. The data is as follows:

	id_pinjam	isbn	qty
<input type="checkbox"/>	1	092-111	1
<input type="checkbox"/>	1	777-381	3
<input type="checkbox"/>	1	999-281	1
<input type="checkbox"/>	2	777-381	1
<input type="checkbox"/>	3	009-281	1
<input type="checkbox"/>	3	381-561	1

Below the table are buttons for 'Extra options' and 'Search the table'.

Kita lihat di table detail peminjaman kita disini membutuhkan isbn, dan qty , lanjut table buku kita butuh menampilkan semua pada kolom table buku

Server: 127.0.0.1 » Database: perpus » Table: buku

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking

Showing rows 0 - 14 (15 total, Query took 0.0004 seconds.)

SELECT \* FROM `buku`

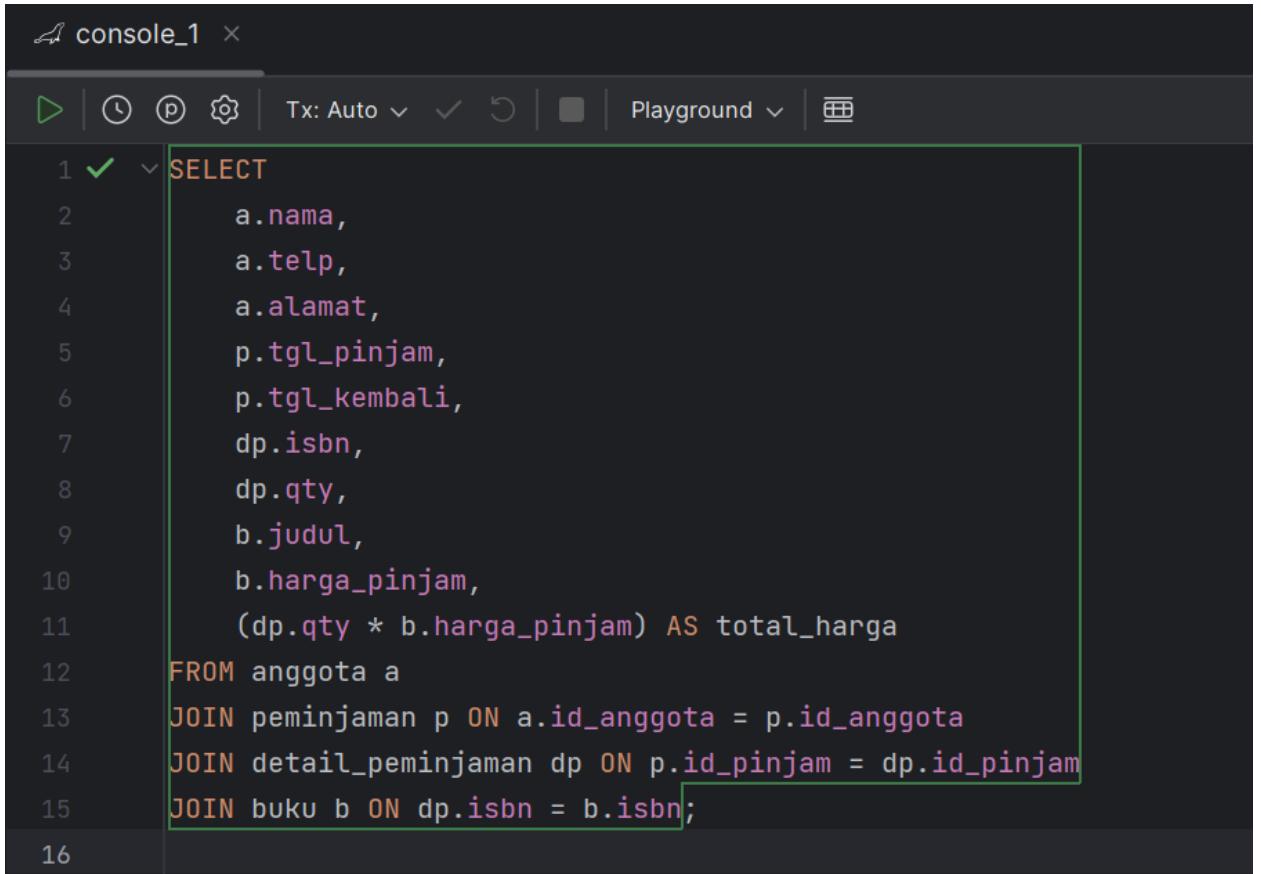
Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	isbn	judul	tahun	id_penerbit	id_pengarang	id_katalog	qty_stok	harga_pinjam
<input type="checkbox"/>	002-291	Lancar Javascript	2018	PN02	PG05	KG2	8	5000
<input type="checkbox"/>	009-281	Basic PHP	2021	PN04	PG01	KG1	19	7500
<input type="checkbox"/>	092-111	Belajar PHP	2010	PN01	PG01	KG0	12	12000
<input type="checkbox"/>	377-482	MySQL Dasar	2020	PN04	PG04	KG0	20	4000
<input type="checkbox"/>	381-561	Basic Vue.js	2014	PN03	PG01	KG2	5	5000
<input type="checkbox"/>	774-210	Laravel Master	2021	PN03	PG05	KG1	7	6500
<input type="checkbox"/>	774-211	Laravel Part 1	2018	PN03	PG05	KG1	5	4500
<input type="checkbox"/>	777-380	Mongo DB Lanjut	2020	PN01	PG03	KG2	7	10000

Itu contoh data dari semua kolom pada table buku, tetapi di studi kasus ini kita disuruh menampilkan nama, telp, alamat, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, harga pinjam dan total harga (qty di kali harga pinjam), dan untuk perkalian kalian gunakan tanda (\*) atau bisa lihat disini [Operator Aritmatika](#), okay langsung saja berikut adalah query untuk menyelesaikan logic dari studi kasus nomer 13

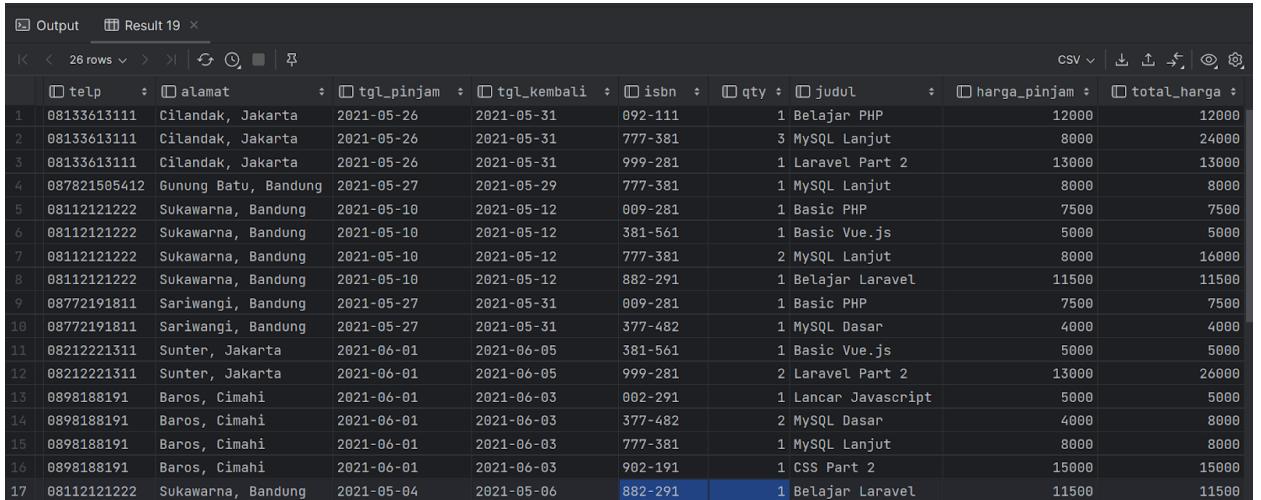


```

1 ✓  v SELECT
2      a.nama,
3      a.telp,
4      a.alamat,
5      p.tgl_pinjam,
6      p.tgl_kembali,
7      dp.isbn,
8      dp.qty,
9      b.judul,
10     b.harga_pinjam,
11     (dp.qty * b.harga_pinjam) AS total_harga
12    FROM anggota a
13    JOIN peminjaman p ON a.id_anggota = p.id_anggota
14    JOIN detail_peminjaman dp ON p.id_pinjam = dp.id_pinjam
15    JOIN buku b ON dp.isbn = b.isbn;
16

```

Jika kita ubah ke bahasa manusia maka seperti berikut : “sistem tolong ambil nama, telp, alamat pada table anggota dan ambilkan juga pada table peminjaman, detail peminjaman dan buku, dimana peminjaman yaitu kolom tgl pinjam, tgl kembali, detail peminjaman yaitu kolom qty, isbn dan table buku kolom judul, harga pinjam tetapi saya ingin dari qty x harga pinjam” maka outputnya sebagai berikut :



	telp	alamat	tgl_pinjam	tgl_kembali	isbn	qty	judul	harga_pinjam	total_harga
1	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31	092-111	1	Belajar PHP	12000	12000
2	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31	777-381	3	MySQL Lanjut	8000	24000
3	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31	999-281	1	Laravel Part 2	13000	13000
4	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29	777-381	1	MySQL Lanjut	8000	8000
5	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12	009-281	1	Basic PHP	7500	7500
6	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12	381-561	1	Basic Vue.js	5000	5000
7	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12	777-381	2	MySQL Lanjut	8000	16000
8	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12	882-291	1	Belajar Laravel	11500	11500
9	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31	009-281	1	Basic PHP	7500	7500
10	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31	377-482	1	MySQL Dasar	4000	4000
11	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05	381-561	1	Basic Vue.js	5000	5000
12	08212221311	Sunter, Jakarta	2021-06-01	2021-06-05	999-281	2	Laravel Part 2	13000	26000
13	0898188191	Baros, Cimahi	2021-06-01	2021-06-03	002-291	1	Lancar Javascript	5000	5000
14	0898188191	Baros, Cimahi	2021-06-01	2021-06-03	377-482	2	MySQL Dasar	4000	8000
15	0898188191	Baros, Cimahi	2021-06-01	2021-06-03	777-381	1	MySQL Lanjut	8000	8000
16	0898188191	Baros, Cimahi	2021-06-01	2021-06-03	902-191	1	CSS Part 2	15000	15000
17	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06	882-291	1	Belajar Laravel	11500	11500

- 14) Tampilkan nama anggota, telp anggota, alamat anggota, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, nama penerbit, nama pengarang dan nama katalog

Okay kita lanjut gais, untuk menyelesaikan studi kasus nomer 4 ini kita membutuhkan JOIN dari beberapa table diantarnya table peminjam, detail peminjaman, buku, penerbit, pengarang, dan katalog, **TETAP SEMANGAT YUK**

```
var user = new Programmer();
while( true ){
    user.wake();
    user.code();
    user.sleep();
    if( user.isDead() != false ) {
        user.erase();
        break;
    }
}
```

“Jika ingin menjadi programmer kita bisa simulasi menggunakan siklus nya perulangan while true (wake, code, sleep, repeat) tetapi kita tidak mungkin seterusnya menjadi programmer, karena manusia pada dasarnya akan mati jadi jika kita dead (mati) !=false (= true) maka user.erase dan break; selesai”  
Wkwkwk okey lanjut dengan pembahasan menyelesaikan logic dari nomer 14, sebelumnya mohon maaf bila garing :v, okay langsung saja berikut adalah query untuk menyelesaikan logic nomer 14

```
console_1 ×
▶ ⏴ Tx: Auto ✓ ⏵ ⏴ Playground ⏴

1 ✓ SELECT
2     a.nama AS nama_anggota,
3     a.telp AS telp_anggota,
4     a.alamat AS alamat_anggota,
5     p.tgl_pinjam,
6     p.tgl_kembali,
7     dp.isbn,
8     dp.qty,
9     b.judul,
10    pen.nama_penerbit,
11    peng.nama_pengarang,
12    kat.nama AS nama_katalog
13   FROM anggota a
14   JOIN peminjaman p ON a.id_anggota = p.id_anggota
15   JOIN detail_peminjaman dp ON p.id_pinjam = dp.id_pinjam
16   JOIN buku b ON dp.isbn = b.isbn
17   JOIN penerbit pen ON b.id_penerbit = pen.id_penerbit
18   JOIN pengarang peng ON b.id_pengarang = peng.id_pengarang
19   JOIN katalog kat ON b.id_katalog = kat.id_katalog;
20
```

Mari kita ubah ke dalam bahasa manusia “Sistem ambilkan dan tampilkan nama, tlp, alamat pada table anggota dan tolong saya minta gabungkan atau JOIN table peminjaman, details peminjaman, buku, penerbit, pengarang dan katalog tetapi hanya spesifikasi tertentu yaitu “nama anggota, telp anggota, alamat anggota, tanggal pinjam, tanggal kembali, isbn, qty, judul buku, nama penerbit, nama pengarang dan nama katalog”” sehingga outputnya sebagai berikut

Output Result 35

	nama_anggota	telp_anggota	alamat_anggota	tgl_pinjam	tgl_kembali
1	Fadhli	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31
2	Fadhli	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31
3	Pelita	087821505412	Gunung Batu, Bandung	2021-05-27	2021-05-29
4	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
5	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
6	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02
7	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
8	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
9	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08
10	Ridwan	0898188191	Baros, Cimahi	2021-06-01	2021-06-03
11	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02
12	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
13	Ayu	08112121222	Sukawarna, Bandung	2021-05-04	2021-05-06
14	Nur	08212221311	Sunter, Jakarta	2021-05-25	2021-06-02
15	Ayu	08112121222	Sukawarna, Bandung	2021-05-10	2021-05-12
16	Mahendra	08772191811	Sariwangi, Bandung	2021-05-27	2021-05-31
17	Feby	08991717711	Sukajadi, Bandung	2021-06-02	2021-06-08
18	Fadhli	08133613111	Cilandak, Jakarta	2021-05-26	2021-05-31

- 15) Tampilkan semua data katalog, judul buku, dimana semua data katalog mempunyai relasi ke data buku

Okey untuk logic nomer 15 ini kita tidak banyak lagi melakukan JOIN, hanya untuk satu join yaitu BUKU karena disini kita disuruh menampilkan semua data katalog, judul buku tetapi yang hanya mempunyai relasi ke data buku berarti foreign key ke buku, **Let's go kita selesaikan**

console\_1

```

SELECT bismilahkatalog.*, buku.judul
FROM katalog bismilahkatalog
JOIN buku ON bismilahkatalog.id_katalog = buku.id_katalog;

```

Mari kita ubah ke bahasa manusia “Sistem tolong ambil semua data pada katalog tetapi gabungkan dengan table buku menggunakan JOIN dan

tampilkan nama buku yang ada atau cocok dengan data di id\_katalog” sehingga outputnya sebagai berikut :

	id_katalog	nama	judul
1	KG2	Buku Belajar	Lancar Javascript
2	KG1	Buku Anak	Basic PHP
3	KG0	Buku Dewasa	Belajar PHP
4	KG0	Buku Dewasa	MySQL Dasar
5	KG2	Buku Belajar	Basic Vue.js
6	KG1	Buku Anak	Laravel Master
7	KG1	Buku Anak	Laravel Part 1
8	KG2	Buku Belajar	Mongo DB Lanjut
9	KG0	Buku Dewasa	MySQL Lanjut
10	KG0	Buku Dewasa	Belajar CSS
11	KG1	Buku Anak	Belajar Laravel
12	KG0	Buku Dewasa	CSS Part 2
13	KG0	Buku Dewasa	Basic JQuery
14	KG0	Buku Dewasa	CSS Part 1
15	KG1	Buku Anak	Laravel Part 2

- 16) Tampilkan semua data buku dan nama penerbit. Beserta data buku yang tidak mempunyai relasi ke data penerbit

Okay pada studi kasus nomer 16 ini kita hanya membutuhkan LEFT JOIN nama penerbit dari table penerbit, berikut sql query untuk problem solving pada nomer 16

```
console_1
SELECT buku.*, penerbit.nama_penerbit
FROM buku
LEFT JOIN penerbit ON buku.id_penerbit = penerbit.id_penerbit;
```

Sehingga outputnya sebagai berikut :

Output Result 28

	isbn	judul	tahun	id_penerbit	id_pengarang	id_katalog
1	002-291	Lancar Javascript	2018	PN02	PG05	KG2
2	009-281	Basic PHP	2021	PN04	PG01	KG1
3	092-111	Belajar PHP	2010	PN01	PG01	KG0
4	377-482	MySQL Dasar	2020	PN04	PG04	KG0
5	381-561	Basic Vue.js	2014	PN03	PG01	KG2
6	774-210	Laravel Master	2021	PN03	PG05	KG1
7	774-211	Laravel Part 1	2018	PN03	PG05	KG1
8	777-380	Mongo DB Lanjut	2020	PN01	PG03	KG2
9	777-381	MySQL Lanjut	2021	PN01	PG04	KG0
10	882-191	Belajar CSS	2020	PN03	PG05	KG0
11	882-291	Belajar Laravel	2020	PN03	PG05	KG1
12	902-191	CSS Part 2	2020	PN04	PG05	KG0
13	929-181	Basic JQuery	2019	<null>	PG05	KG0
14	977-381	CSS Part 1	2018	PN01	PG01	KG0
15	999-281	Laravel Part 2	2020	PN04	PG05	KG1

17) Tampilan ada berapa jumlah pengarang “PG05” pada table buku

Okay untuk logic ini kita hanya butuh function COUNT untuk menghitung total data pada table buku lalu gunakan WHERE dengan id\_pengarang = “PG05”, berikut contoh query untuk menyelesaikan problem solving nomer 17

console\_1

```

SELECT COUNT(*) AS jumlah_pengarang_PG05
FROM buku
WHERE id_pengarang = 'PG05';

```

Maka outputnya adalah :

jumlah_pengarang_PG05:int	
<	< 1 row > >
1	8

18) Tampilkan data buku yang harganya lebih dari 10000

Untuk nomer 18 kita hanya membutuhkan kondisi WHERE harga\_pinjam > 10000 , berikut adalah sql query untuk menyelesaikan logic dari nomer 18

```
console_1 x
SELECT *
FROM buku
WHERE harga_pinjam > 10000;
```

Sehingga outputnya sebagai berikut :

Output perpus.buku

	isbn	judul	tahun	id_penerbit	id_pengarang	id_katalog
1	092-111	Belajar PHP	2010	PN01	PG01	KG0
2	882-191	Belajar CSS	2020	PN03	PG05	KG0
3	882-291	Belajar Laravel	2020	PN03	PG05	KG1
4	902-191	CSS Part 2	2020	PN04	PG05	KG0
5	999-281	Laravel Part 2	2020	PN04	PG05	KG1

- 19) Tampilkan seluruh data buku yang diterbitkan oleh Penerbit 01, dimana buku tersebut harus mempunyai qty lebih dari 10

Nomer 19 ini kita disuruh menampilkan buku tetapi hanya yang di terbitkan oleh Penerbit 01 dan mempunya qty lebih dari 10, untuk menyelesaikan ini kita hanya perlu menggunakan logic WHERE + AND, berikut sql query untuk problem solving nomer 19

console\_1

```

SELECT *
FROM buku
WHERE id_penerbit = 'PN01' AND qty_stok > 10;

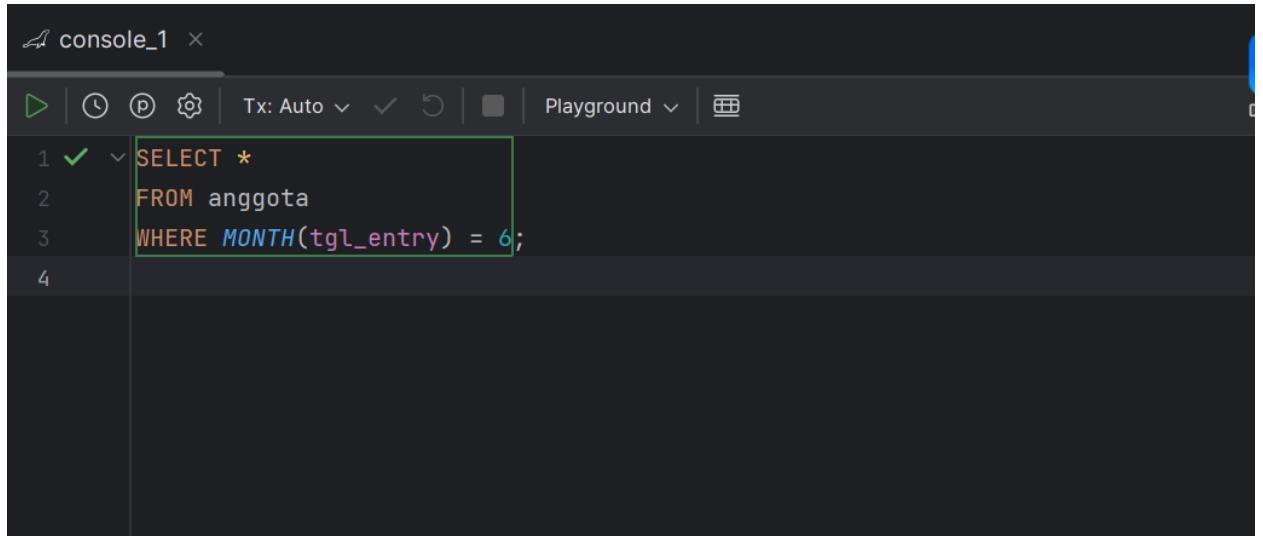
```

Sehingga Outputnya adalah :

Output perpus.buku

	isbn	judul	tahun	id_penerbit	id_pengarang	id_katalog
1	092-111	Belajar PHP	2010	PN01	PG01	KG0

- 20) Tampilkan seluruh data anggota yang baru ditambahkan pada bulan juni  
Untuk kasus akhir ini kita hanya perlu logic WHERE MONTH = 6, berikut adalah sql query untuk menyelesaikan logic dari nomer 20

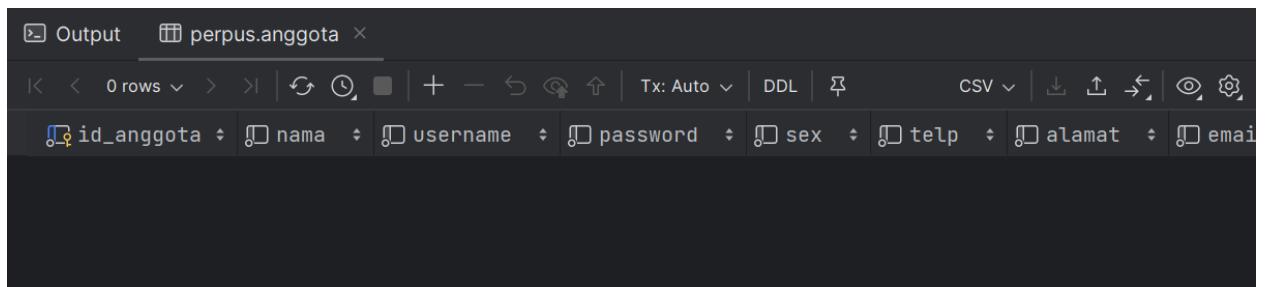


The screenshot shows a MySQL command-line interface window titled "console\_1". In the query editor, the following SQL code is written:

```
1 ✓ 1 SELECT *
2      FROM anggota
3     WHERE MONTH(tgl_entry) = 6;
```

The first line has a green checkmark icon and a dropdown arrow, indicating it is the current or selected query. The code is highlighted in green.

Sehingga outputnya adalah :



The screenshot shows the results of the SQL query in a table viewer. The title bar says "Output perpus.anggota". The table header includes columns: id\_anggota, nama, username, password, sex, telp, alamat, and email. Below the header, there is a message: "0 rows".