

LAPORAN PERTEMUAN 9 TEORI

SISTEM BASIS DATA



Ditulis oleh :

NAMA : Ilham Romadhani

NIM : 23.230.0030

KELAS : 2P52

Dosen Pengampu :

Mosses Aidjilli, M.Kom

PROGRAM STUDI SISTEM INFORMASI

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

(STMIK) WIDYA PRATAMA PEKALONGAN

2024

➤ DATA DEFINITION LANGUAGE (DDL)

• Menampilkan record (SELECT lebih dari satu tabel / JOIN)

1. Menampilkan semua supplier dan part yang keduanya bertempat tinggal pada kota yang sama

```
MariaDB [ilham]> SELECT Sn, Sname, Status, S_ILHAM.City, Pn, Pname, Warna, Weight FROM S_ILHAM, P_ILHAM WHERE S_ILHAM.City = P_ILHAM.City;
```

Sn	Sname	Status	City	Pn	Pname	Warna	Weight
S1	Smith	40	London	P1	Nut	Red	12
S4	Clark	40	London	P1	Nut	Red	12
S2	Jones	10	Paris	P2	Bolt	Yellow	22
S3	Blake	30	Paris	P2	Bolt	Yellow	22
S1	Smith	40	London	P4	Screw	Red	14
S4	Clark	40	London	P4	Screw	Red	14
S2	Jones	10	Paris	P5	Cam	Blue	12
S3	Blake	30	Paris	P5	Cam	Blue	12
S1	Smith	40	London	P6	Cog	Red	19
S4	Clark	40	London	P6	Cog	Red	19

10 rows in set (0.000 sec)

Select sn,sname,status,S_ilham.city,pn,pname,warna,weight : memilih kolom-kolom dari tabel.

from s_ilham, p_ilham : dari 2 tabel.

where s_ilham.city = p_ilham.city : kondisi dimana nilai field city dari kedua tabel tersebut harus sama. (namatabel.namafield).

2. Menampilkan nama supplier yang memasok barang dengan nomor part P2

```
MariaDB [ilham]> SELECT Sname FROM S_ILHAM, SP_ILHAM WHERE S_ILHAM.Sn = SP_ILHAM.Sn AND SP_ILHAM.Pn = 'P2';
```

Sname
Smith
Jones
Blake
Clark

4 rows in set (0.001 sec)

Select Sname : memilih field/kolom Sname.

from s_ilham, sp_ilham : dari tabel s_ilham dan sp_ilham.

where s_ilham.sn = sp_ilham.sn and sp_ilham.pn = 'P2' : kondisi dimana nilai kolom sn dari tabel s_ilham harus sama dengan nilai kolom sn pada tabel sp_ilham. Serta ditambah dengan logika and dengan kondisi nilai kolom pn harus sama dengan 'p2' (string p2). Jadi karena ada logika AND, kedua kondisi tersebut harus benar.

3. Menampilkan nama supplier yang memasok part berwarna merah

```
MariaDB [ilham]> SELECT Sname FROM S_ILHAM, SP_ILHAM, P_ILHAM WHERE S_ILHAM.Sn = SP_ILHAM.Sn AND SP_ILHAM.Pn = P_ILHAM.Pn AND P_ILHAM.COLOR = 'RED';  
ERROR 1054 (42S22): Unknown column 'P_ILHAM.COLOR' in 'where clause'  
MariaDB [ilham]>
```

Error karena field sudah di ubah ke warna bukan color

```

MariaDB [ilham]> SELECT Sname FROM S_ILHAM, SP_ILHAM, P_ILHAM WHERE S_ILHAM.Sn = SP_ILHAM.Sn AND SP_ILHAM.Pn
= P_ILHAM.Pn AND P_ILHAM.warna = 'RED';
+-----+
| Sname |
+-----+
| Smith |
| Smith |
| Smith |
| Jones |
| Clark |
+-----+
5 rows in set (0.001 sec)

```

Select Sname : Memilih field Sname

FROM s_ilham, sp_ilham, p_ilham : Dari 3 tabel tersebut.

Where s_ilham.sn = sp_ilham.sn: kondisi dimana nilai pada kolom sn pada tabel s_ilham sama dengan nilai pada kolom sn pada tabel sp_ilham.

AND sp_ilham.pn = p_ilham.pn : ada logika AND dimana kondisi diantara keduanya harus benar = true. Kondisi selanjutnya dimana nilai kolom pn pada tabel sp_ilham harus sama dengan nilai kolom pn pada tabel p_ilham.

AND p_ilham.warna = 'RED' : ada logika AND lagi. Kondisi selanjutnya dimana kolom warna pada tabel p_ilham harus sama dengan nilai = 'RED'.

- **Menampilkan record (SELECT lebih dari satu tabel / SELECT Bertingkat)**

1. Menampilkan nama supplier yang memasok barang dengan nomor part P2

```

MariaDB [ilham]> SELECT Sname FROM S_ILHAM WHERE Sn IN (SELECT Sn FROM SP_ILHAM WHERE Pn = 'P2');
+-----+
| Sname |
+-----+
| Smith |
| Jones |
| Blake |
| Clark |
+-----+
4 rows in set (0.001 sec)

```

Select Sname : memilih kolom sname.

FROM S_Ilham : dari tabel s_Ilham.

Operator IN adalah cara untuk memeriksa apakah nilai kolom ada dalam daftar nilai tertentu.

WHERE Sn IN (Select Sn FROM sp_ilham WHERE Pn = 'P2') : kondisi dimana nilai kolom Sn dari tabel S_Ilham dicek apakah ada dalam daftar nilai Sn yang dihasilkan dari subquery SELECT Sn FROM sp_ilham WHERE Pn = 'P2'. Dengan kata lain, kita memilih baris-baris dari tabel S_Ilham di mana nilai Sn-nya ada di dalam daftar nilai yang kita dapatkan dari subquery.

2. Menampilkan nama supplier yang memasok part berwarna merah

```

MariaDB [ilham]> SELECT Sname FROM S_ILHAM WHERE Sn IN (SELECT Sn FROM SP_ILHAM WHERE Pn IN (SELECT Pn FROM P_ILHAM WHERE Warna = 'Red'));
+-----+
| Sname |
+-----+
| Smith |
| Jones |
| Clark |
+-----+
3 rows in set (0.000 sec)

```

SELECT Sname FROM S_ILHAM : Memilih kolom Sname dari tabel s_ilham.
WHERE Sn IN (SELECT Sn FROM SP_ILHAM WHERE Pn IN (SELECT Pn FROM P_ILHAM WHERE Warna = 'Red')) : kondisi untuk membatasi hasil yang kita inginkan. Di sini, kita menggunakan operator **IN** untuk memeriksa apakah nilai **Sn** (nomor supplier) ada dalam hasil subquery di dalam tanda kurung.

3. Menampilkan no.supplier dengan nilai status lebih kecil daripada nilai maksimum status yang ada pada tabel S_ilham

```
MariaDB [ilham]> SELECT Sn FROM S_ILHAM WHERE Status < (SELECT MAX(Status) FROM S_ILHAM);
+-----+
| Sn |
+-----+
| S2 |
| S3 |
+-----+
2 rows in set (0.001 sec)
```

SELECT Sn FROM S_ILHAM : Memilih kolom Sn dari tabel s_ilham.
WHERE status < (SELECT MAX(Status) FROM S_ILHAM) : kondisi dimana kolom nilai status pada tabel s_ilham lebih kecil dari nilai maksimum status.
(SELECT MAX(Status) FROM S_ILHAM) : subquery ini menghitung nilai maksimum dari kolom status di tabel s_ilham.
MAX(Status) : Mengambil nilai terbesar dari kolom status.

4. Menampilkan nama supplier yang tidak memasok barang dengan nomor part P2

```
MariaDB [ilham]> SELECT Sname FROM S_ILHAM WHERE Sn NOT IN (SELECT Sn FROM SP_ILHAM WHERE Pn = 'P2');
Empty set (0.001 sec)
```

SELECT Sname FROM s_ilham : Memilih kolom Sname dari tabel s_ilham.
WHERE Sn NOT IN : kondisi dimana mencari Sn yang tidak ada dalam daftar nilai yang di tentukan atau di hasilkan dari subquery di bawah ini.
(SELECT Sn FROM SP_ILHAM WHERE Pn = 'P2') : Mencari nilai Sn dengan kondisi dimana kolom Pn pada tabel SP_ILHAM nilainya sama dengan 'P2'.

5. Menampilkan semua nomor supplier yang sama lokasinya dengan S1

```
MariaDB [ilham]> SELECT Sn FROM S_ILHAM WHERE CITY =(SELECT CITY FROM S_ILHAM WHERE Sn = 'S1');
+-----+
| Sn |
+-----+
| S1 |
| S4 |
+-----+
2 rows in set (0.000 sec)
```

SELECT Sn FROM S_ILHAM : Memilih kolom Sn dari tabel s_ilham.
WHERE CITY = (SELECT CITY FROM S_ILHAM WHERE Sn = 'S1') : Kondisi dimana field city pada tabel s_ilham sama dengan subquery().
(SELECT CITY FROM S_ILHAM WHERE Sn = 'S1') : Mencari nilai city dari tabel s_ilham dimana jika nilai Sn pada tabel s_ilham sama dengan 'S1'.

- **Fungsi Perhitungan**

1. Menghitung jumlah supplier

```
MariaDB [ilham]> SELECT COUNT(*) FROM S_ILHAM;
+-----+
| COUNT(*) |
+-----+
|         4 |
+-----+
1 row in set (0.000 sec)
```

SELECT COUNT(*) FROM S_ILHAM : Menghitung jumlah record pada tabel.

2. Menampilkan nomor part dan total kuantitas pengiriman dari setiap part

```
MariaDB [ilham]> SELECT Pn, SUM(QTY) FROM SP_ILHAM GROUP BY Pn;
+----+-----+
| Pn | SUM(QTY) |
+----+-----+
| P  |         40 |
| P1 |        600 |
| P2 |       1000 |
| P4 |        500 |
| P5 |        500 |
| P6 |        100 |
+----+-----+
6 rows in set (0.000 sec)
```

SELECT Pn, SUM(QTY) : Memilih kolom Pn dan fungsi SUM untuk menjumlahkan nilai di kolom QTY.

FROM SP_ILHAM : Dari tabel sp_ilham.

GROUP BY Pn: Mengelompokkan baris yang memiliki nilai yang sama di kolom Pn. Setelah pengelompokan, fungsi agregat seperti SUM diterapkan ke setiap grup. Dalam hal ini, kita menjumlahkan nilai QTY untuk setiap grup yang diidentifikasi oleh Pn.

3. Menghitung jumlah kuantitas dari P2 yang telah disupply

```
MariaDB [ilham]> SELECT SUM(QTY) FROM SP_ILHAM WHERE Pn = 'P2';
+-----+
| SUM(QTY) |
+-----+
|       1000 |
+-----+
1 row in set (0.000 sec)
```

SELECT SUM(QTY) FROM SP_ILHAM : Memilih field qty dengan fungsi SUM untuk menjumlahkan nilai pada kolom qty.

WHERE Pn = 'P2' : kondisi dimana field Pn = 'P2' (string P2). Akan menjumlahkan semua nilai qty dari kolom yang sama dengan P2.

4. Menampilkan jumlah pengiriman barang dengan nomor P4 dan dipasok oleh nomor supplier S1

```
MariaDB [ilham]> SELECT COUNT(*) FROM SP_ILHAM WHERE Pn = 'P4' AND Sn = 'S1';
+-----+
| COUNT(*) |
+-----+
|          1 |
+-----+
1 row in set (0.001 sec)
```

SELECT COUNT(*) FROM SP_ILHAM : Menjumlahkan total record atau baris dari tabel sp_ilham, dengan kondisi di bawah ini.

WHERE Pn = 'P4' AND Sn = 'S1' : Kondisi dimana record akan terhitung/true jika field Pn = 'P4' DAN Sn = 'S1', karena ada logika AND jadi kedua kondisi tersebut harus benar maka data akan di hitung oleh fungsi count.

5. Menampilkan nomor part dan total kuantitas dari masing-masing part

```
MariaDB [ilham]> Select pn, sum(qty) from sp_ilham group by p3;
ERROR 1054 (42S22): Unknown column 'p3' in 'group statement'
```

P3 menghasilkan kesalahan karena bukan nama kolom yang valid dalam tabel.

Maka dari itu untuk menampilkan total kuantitas dari masing-masing part ganti dengan Pn.

```
MariaDB [ilham]> Select pn, sum(qty) from sp_ilham group by pn;
+----+-----+
| pn | sum(qty) |
+----+-----+
| P1 |        600 |
| P2 |       1000 |
| P4 |        500 |
| P5 |        500 |
| P6 |        100 |
+----+-----+
5 rows in set (0.000 sec)
```

SELECT Pn, SUM(QTY) FROM SP_ILHAM : Memilih kolom Pn dan fungsi SUM untuk menjumlahkan nilai di kolom QTY.

GROUP BY Pn : Mengelompokkan baris yang memiliki nilai yang sama di kolom Pn. Setelah pengelompokan, fungsi agregat seperti SUM diterapkan ke setiap grup. Dalam hal ini, kita menjumlahkan nilai QTY untuk setiap grup yang diidentifikasi oleh Pn.

TEORI YANG ADA DI PAPAN TULIS

MASTER = Primary Key

TRANSAKSI = FOREIGN KEY

MASTER TRANSAKSI = PRIMARY KEY

1. BUATLAH TABEL BARANG

```
MariaDB [ilham]> create table barang (  
  -> kd_brg char(10),  
  -> namabrg char(35),  
  -> harga int(8),  
  -> satuan char(25),  
  -> stok int(10),  
  -> primary key (kd_brg)  
  -> );
```

Query OK, 0 rows affected (0.205 sec)

```
MariaDB [ilham]> desc barang;
```

Field	Type	Null	Key	Default	Extra
kd_brg	char(10)	NO	PRI	NULL	
namabrg	char(35)	YES		NULL	
harga	int(8)	YES		NULL	
satuan	char(25)	YES		NULL	
stok	int(10)	YES		NULL	

5 rows in set (0.011 sec)

2. BUATLAH TABEL KASIR

```
MariaDB [ilham]> create table kasir (  
  -> kd_kasir char(10),  
  -> namakasir char(35),  
  -> alamat char(10),  
  -> jenis_kelamin char(1),  
  -> telepon char(15),  
  -> primary key (kd_kasir));
```

Query OK, 0 rows affected (0.183 sec)

```
MariaDB [ilham]> desc kasir;
```

Field	Type	Null	Key	Default	Extra
kd_kasir	char(10)	NO	PRI	NULL	
namakasir	char(35)	YES		NULL	
alamat	char(10)	YES		NULL	
jenis_kelamin	char(1)	YES		NULL	
telepon	char(15)	YES		NULL	

5 rows in set (0.028 sec)

3. BUATLAH TABEL MASTER TRANSAKSI

```
MariaDB [ilham]> create table master_transaksi (
  -> no_trans char(25),
  -> tgl_trans date,
  -> kd_kasir char(10),
  -> total_bayar int(10),
  -> foreign key (kd_kasir) references kasir(kd_kasir),
  -> primary key (no_trans));
Query OK, 0 rows affected (0.183 sec)

MariaDB [ilham]> desc master_transaksi;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| no_trans   | char(25)  | NO   | PRI | NULL    |       |
| tgl_trans  | date      | YES  |     | NULL    |       |
| kd_kasir   | char(10)  | YES  | MUL | NULL    |       |
| total_bayar | int(10)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.011 sec)
```

TABEL MASTER_TRANSAKSI BERELASI DENGAN TABEL KASIR

4. BUATLAH TABEL TRANSAKSI

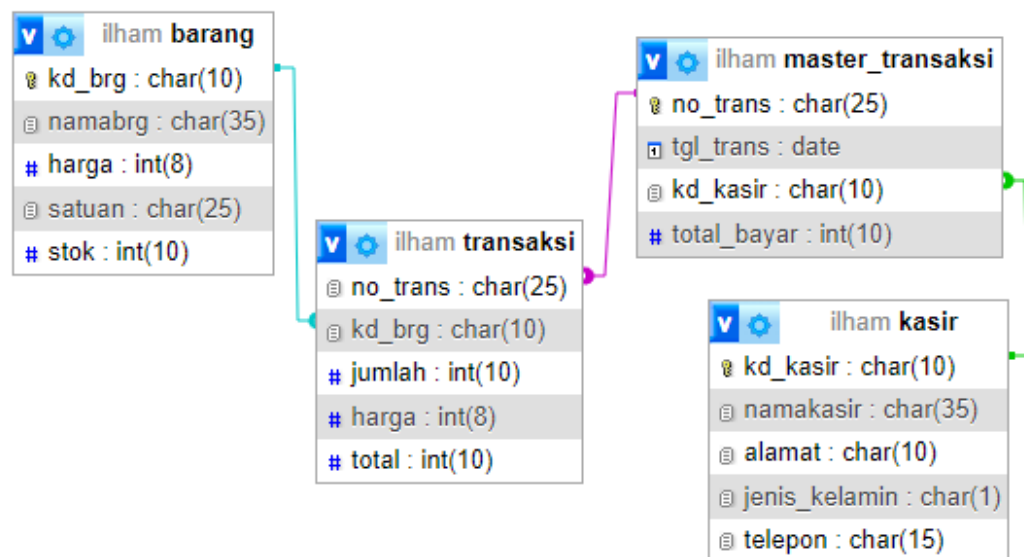
GENERATED ALWAYS AS (jumlah * harga) STORED adalah kolom yang nilainya otomatis dihitung dari jumlah x harga dan disimpan di tabel untuk kolom total di bawah ini.

```
MariaDB [ilham]> create table transaksi (
  -> no_trans char(25),
  -> kd_brg char(10),
  -> jumlah int(10),
  -> harga int(8),
  -> total int(10) GENERATED ALWAYS AS (jumlah * harga) STORED,
  -> foreign key (no_trans) references master_transaksi (no_trans),
  -> foreign key (kd_brg) references barang (kd_brg)
  -> );
Query OK, 0 rows affected (0.206 sec)

MariaDB [ilham]> desc transaksi;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| no_trans   | char(25)  | YES  | MUL | NULL    |       |
| kd_brg     | char(10)  | YES  | MUL | NULL    |       |
| jumlah     | int(10)   | YES  |     | NULL    |       |
| harga      | int(8)    | YES  |     | NULL    |       |
| total      | int(10)   | YES  |     | NULL    | STORED GENERATED |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.011 sec)
```

TABEL TRANSAKSI BERELASI DENGAN TABEL BARANG DAN
MASTER_TRANSAKSI

5. TAMPILAN RELASINYA



6. INSERT DATA PADA TABEL BARANG

```

MariaDB [ilham]> insert into barang values
  -> ('D001','SABUN',50,'BIJI',500),
  -> ('D002','CUKA',25,'BOTOL',1500)
  -> ;

```

```

Query OK, 2 rows affected (0.036 sec)
Records: 2  Duplicates: 0  Warnings: 0

```

```

MariaDB [ilham]> select * from barang;

```

kd_brg	namabrg	harga	satuan	stok
D001	SABUN	50	BIJI	500
D002	CUKA	25	BOTOL	1500

2 rows in set (0.000 sec)

7. INSERT DATA PADA TABEL KASIR

```

MariaDB [ilham]> insert into kasir values
  -> ('M-01','TUTI','JL.M','W','08509112145'),
  -> ('M-02','ALI','JL.T','L','08744124541');

```

```

Query OK, 2 rows affected (0.039 sec)
Records: 2  Duplicates: 0  Warnings: 0

```

```

MariaDB [ilham]> select * from kasir;

```

kd_kasir	namakasir	alamat	jenis_kelamin	telepon
M-01	TUTI	JL.M	W	08509112145
M-02	ALI	JL.T	L	08744124541

2 rows in set (0.001 sec)

8. INSERT DATA PADA TABEL MASTER_TRANSAKSI

Insert data akan tetapi total_bayar tidak ada nilai, karena akan di tambah nilai menggunakan rumus, di mana nilai di dapat dari tabel transaksi kolom total transaksi.

```
MariaDB [ilham]> insert into master_transaksi
-> (no_trans,tgl_trans,kd_kasir,total_bayar) values
-> ('tr-08-08-24-001','20240808','M-01',null);
Query OK, 1 row affected (0.043 sec)
```

```
MariaDB [ilham]> select * from master_transaksi;
+-----+-----+-----+-----+
| no_trans      | tgl_trans | kd_kasir | total_bayar |
+-----+-----+-----+-----+
| tr-08-08-24-001 | 2024-08-08 | M-01     | NULL        |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

9. INSERT DATA PADA TABEL TRANSAKSI

Tidak perlu input nilai total, karena akan di isi otomatis dengan mengkalikan jumlah * harga.

```
MariaDB [ilham]> INSERT INTO transaksi (no_trans, kd_brg, jumlah, harga) VALUES
-> ('tr-08-08-24-001', 'D001', 4, 50),
-> ('tr-08-08-24-001', 'D002', 10, 25);
Query OK, 2 rows affected (0.055 sec)
Records: 2 Duplicates: 0 Warnings: 0
```

```
MariaDB [ilham]> select * from transaksi;
+-----+-----+-----+-----+-----+
| no_trans      | kd_brg | jumlah | harga | total |
+-----+-----+-----+-----+-----+
| tr-08-08-24-001 | D001   | 4       | 50    | 200   |
| tr-08-08-24-001 | D002   | 10      | 25    | 250   |
+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)
```

10. MENGHITUNG/UPDATE TOTAL BAYAR TABEL MASTER_TRANSAKSI DARI TABEL TRANSAKSI(TOTAL) DENGAN RUMUS SUM.

```
MariaDB [ilham]> UPDATE master_transaksi
-> SET total_bayar = ( SELECT SUM(total) FROM transaksi
-> WHERE no_trans = 'tr-08-08-24-001')
-> WHERE no_trans = 'tr-08-08-24-001';
Query OK, 1 row affected (0.058 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
MariaDB [ilham]> SELECT * FROM MASTER_TRANSAKSI;
+-----+-----+-----+-----+
| no_trans      | tgl_trans | kd_kasir | total_bayar |
+-----+-----+-----+-----+
| tr-08-08-24-001 | 2024-08-08 | M-01     | 450         |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

Berikut penjelasan per bagian untuk pernyataan SQL diatas:

1. UPDATE master_transaksi: memperbarui tabel master_transaksi.

2. SET total_bayar = (SELECT SUM(total) FROM transaksi WHERE no_trans = 'tr-08-08-24-001') : menetapkan nilai baru untuk kolom total_bayar.

menghitung jumlah (SUM) dari kolom total di tabel transaksi untuk baris-baris yang memiliki no_trans sama dengan 'tr-08-08-24-001'. Nilai hasil dari subquery ini digunakan untuk memperbarui total_bayar.

4. WHERE no_trans = 'tr-08-08-24-001' : Memastikan bahwa hanya baris di tabel master_transaksi dengan no_trans yang sama dengan 'tr-08-08-24-001' yang diperbarui.

11. MENAMPILKAN record kolom

no_trans -> master_transaksi

namabrg -> barang

tgl_trans -> transaksi

namakasir -> kasir

total_bayar -> master_transaksi

```
MariaDB [ilham]> SELECT master_transaksi.no_trans, barang.namabrg,  
-> transaksi.harga, kasir.namakasir, master_transaksi.total_bayar  
-> FROM barang, master_transaksi, transaksi, kasir  
-> WHERE barang.kd_brg = transaksi.kd_brg AND  
-> master_transaksi.no_trans = transaksi.no_trans AND  
-> master_transaksi.kd_kasir = kasir.kd_kasir;
```

no_trans	namabrg	harga	namakasir	total_bayar
tr-08-08-24-001	SABUN	50	TUTI	450
tr-08-08-24-001	CUKA	25	TUTI	450

2 rows in set (0.001 sec)