

**LAPORAN PRAKTIKUM WORKSHOP MANAJEMEN BASISDATA**  
**“MINGGU 11: ACARA 29, 30, 31, 32 DAN 33”**



**DOSEN PENGAMPU:**  
**HUSIN, S.KOM, M.MT**  
**NIP : 19880702 201903 1 010**

**DISUSUN OLEH:**  
Dilas Sholeh Masysyuhur  
E31240474

**PROGRAM STUDI MANAJEMEN INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI JEMBER**  
**2025**

## ACARA 29

Tugas:

1. Implementasikan join tabel dengan relasi 2 tabel dari studi kasus
2. Implementasikan Join dengan Kondisi dan Join tanpa Kondisi pada studi kasus
3. Implementasikan Natural Join pada studi kasus
4. Implementasikan Join dengan clausa yang lain pada studi kasus
5. Lakukan dokumentasi disetiap implementasi join pada studi kasus

Jawaban:

1.

Query SQL

```
1 SELECT
2     penyewaan.id_sewa,
3     pelanggan.nama_pelanggan,
4     penyewaan.tgl_sewa,
5     penyewaan.tgl_kembali
6 FROM
7     penyewaan
8 JOIN
9     pelanggan ON penyewaan.id_pelanggan = pelanggan.id_pelanggan;
10
```

Output

Extra options			
id_sewa	nama_pelanggan	tgl_sewa	tgl_rencana_kembali
PN001	Bintang	2025-04-10	2025-04-11

2. Join kondisi dan tanpa kondisi

a) Join dengan kondisi

Query SQL

```
1 SELECT
2     detail_sewa.id_sewa,
3     barang.nama_barang,
4     barang.status
5 FROM
6     detail_sewa
7 JOIN
8     barang ON detail_sewa.id_barang = barang.id_barang
9 WHERE
10    barang.status = 'Tersedia';
11
```

Output

☐ Tampilkan semua | Jumlah baris: 25

Extra options

id_sewa	nama_barang	status
PN001	Tenda	Tersedia

☐ Tampilkan semua | Jumlah baris: 25

b) Join tanpa kondisi

### Query SQL

```
Run SQL query/queries on table brobets.pengembalian:
1 SELECT
2   detail_sewa.id_sewa,
3   barang.nama_barang,
4   barang.status
5 FROM
6   detail_sewa
7 JOIN
8   barang ON detail_sewa.id_barang = barang.id_barang;
9
```

### Output

Extra options

id_sewa	nama_barang	status
PN001	Tenda	Tersedia

3.

### Query SQL

```
1 SELECT *
2 FROM
3   detail_sewa
4 NATURAL JOIN barang;
5
```

### Output

id_barang	id_detail	id_sewa	qty	sub_total	nama_barang	harga_sewa	kategori	stok	status
BRG001	DT001	PN001	1	20000	Tenda	20000	Peralatan	10	Tersedia

4.

### Query SQL

```
1 SELECT
2   penyewaan.id_sewa,
3   pengembalian.tgl_kembali,
4   pengembalian.status
5 FROM
6   penyewaan
7 LEFT JOIN
8   pengembalian ON penyewaan.id_sewa = pengembalian.id_sewa;
9
```

### Output

id_sewa	tgl_kembali	status
PN001	2025-04-11	NULL

## ACARA 30

Tugas:

1. Implementasikan join tiga tabel pada studi kasus kalian
2. Implementasikan join tiga tabel dengan kondisi pada studi kasus kalian
3. Lakukan dokumentasi di setiap implementasi join tiga tabel pada studi kasus.

Jawaban:

1.

### Query SQL

```
1 SELECT
2     sewa.id_sewa,
3     pelanggan.nama_pelanggan,
4     pengguna.nama_lengkap AS petugas,
5     sewa.tgl_sewa,
6     sewa.tgl_rencana_kembali
7 FROM penyewaan AS sewa
8 JOIN pelanggan ON sewa.id_pelanggan = pelanggan.id_pelanggan
9 JOIN pengguna ON sewa.id_pengguna = pengguna.id_pengguna;
```

### Output

d_sewa	nama_pelanggan	petugas	tgl_sewa	tgl_rencana_kembali
P0001	Bintang	doan	2025-04-10	2025-04-11

2.

### Query SQL

```
1 SELECT
2     penyewaan.id_sewa,
3     pelanggan.nama_pelanggan,
4     barang.nama_barang,
5     barang.status,
6     penyewaan.tgl_sewa
7 FROM
8     penyewaan
9 JOIN
10    detail_sewa ON penyewaan.id_sewa = detail_sewa.id_sewa
11 JOIN
12    barang ON detail_sewa.id_barang = barang.id_barang
13 JOIN
14    pelanggan ON penyewaan.id_pelanggan = pelanggan.id_pelanggan
15 WHERE
16    barang.status = 'Disewa';
```

### Output

id_sewa	nama_pelanggan	nama_barang	status	tgl_sewa
Operasi hasil kueri				
Buat tampilan				

## ACARA 31

Tugas:

1. Implementasikan COUNT, SUM, AVG, MIN, dan MAX pada studi kasus
2. Implementasikan LIKE dan WILDCARDS pada studi kasus
3. Implementasikan IN pada studi kasus
4. Lakukan dokumentasi disetiap implementasi fungsi agregate pada studi kasus

Jawaban:

1. COUNT, SUM, AVG, MIN, dan MAX

a) Count

### Query SQL

```
1 SELECT COUNT(*) AS total_transaksi
2 FROM penyewaan;
```

### Output

total_transaksi
1

b) Sum

### Query SQL

```
1 SELECT SUM(total_harga) AS total_pendapatan
2 FROM penyewaan;
```

### Output

total_pendapatan
20000

c) Avg

### Query SQL

```
SELECT AVG(total_harga) AS rata_rata_penyewaan FROM penyewaan;
```

### Output

rata_rata_penyewaan
20000.0000

d) Min

### Query SQL

```
SELECT MIN(total_harga) AS penyewaan_termurah FROM penyewaan;
```

### Output

penyewaan_termurah
20000

e) Max

### Query SQL

```
SELECT MAX(total_harga) AS penyewaan_termahal FROM penyewaan;
```

### Output

penyewaan_termahal
20000

## 2. COUNT, SUM, AVG, MIN, dan MAX

a) Like

### Query SQL

```
SELECT * FROM pelanggan WHERE nama_pelanggan LIKE '%Andi%';
```

### Output

id_pelanggan	nama_pelanggan	no_hp	poin

b) Wildcards

### Query SQL

```
SELECT * FROM barang WHERE nama_barang LIKE 'T%';
```

### Output

id_barang	nama_barang	harga_sewa	kategori	stok	status
Hapus BRG001	Tenda	20000	Peralatan	10	Tersedia

Dengan pilihan: Ubah Salin Hapus Ekspor

3.

### Query SQL

```
SELECT * FROM barang WHERE status IN ('Tersedia', 'Rusak');
```

### Output

id_barang	nama_barang	harga_sewa	kategori	stok	status
Rusak BRG001	Tenda	20000	Peralatan	10	Tersedia

## ACARA 32

Tugas:

1. Implementasikan salah satu penggunaan Operator Aritmatika pada studi kasus
2. Implementasikan salah satu penggunaan Operator Bitwise pada studi kasus
3. Implementasikan salah satu penggunaan Operator Perbandingan studi kasus
4. Implementasikan salah satu penggunaan Operator Logika
5. Implementasikan penggunaan Operator UNION dengan kondisi
6. Implementasikan penggunaan Operator EXISTS
7. Lakukan dokumentasi disetiap implementasi operator pada studi kasus

Jawaban:

1.

### Query SQL

```
1 SELECT
2   nama_barang,
3   harga_sewa,
4   harga_sewa * 0.1 AS pajak,
5   harga_sewa * (harga_sewa * 0.1) AS total_dengan_pajak
6 FROM barang;
```

### Output

nama_barang	harga_sewa	pajak	total_dengan_pajak
s Tenda	20000	2000.0	22000.0

2.

### Query SQL

```
SELECT 5 & 3 AS hasil_bitwise;
```

### Output

hasil_bitwise
1

3.

### Query SQL

```
SELECT * FROM penyewaan WHERE total_harga > 5000;
```

### Output

id_sewa	id_pelanggan	id_pengguna	tgl_sewa	tgl_rencana_kembali	total_harga	jaminan	Status
PN001	PL001	UR001	2025-04-10	2025-04-11	20000	IKTM	Sudah Kembali

pilihan: Udah Ralin Hanus Fksnor

4.

### Query SQL

✓ Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0,0007 detik.)

```
SELECT * FROM barang WHERE status = 'Tersedia' AND stok > 0;
```

### Output

	id_barang	nama_barang	harga_sewa	kategori	stok	status
us	BRG001	Tenda	20000	Peralatan	10	Tersedia

5.

### Query SQL

```
1 SELECT id_barang
2 FROM detail_sewa
3 WHERE qty > 1
4
5 UNION
6
7 SELECT id_barang
8 FROM detail_pengembalian
9 WHERE jumlah_dikembalikan > 0;
```

### Output

id_barang
BRG001

6.

### Query SQL

```
SELECT *
FROM pelanggan p
WHERE EXISTS (
  SELECT 1
  FROM penyewaan s
  WHERE s.id_pelanggan = p.id_pelanggan
);
```

### Output

	id_pelanggan	nama_pelanggan	no_hp	poin
ipus	PL001	Bintang	9876543	10



## ACARA 33

Tugas:

1. Implementasikan salah satu penggunaan Statement Case pada studi kasus
2. Implementasikan salah satu penggunaan Klausa Limit pada studi kasus
3. Implementasikan salah satu penggunaan Klausa Having pada studi kasus
4. Lakukan dokumentasi disetiap implementasi operator pada studi kasus

Jawaban:

1.

Query SQL

```
1 SELECT
2   id_barang,
3   nama_barang,
4   stok,
5   status,
6   CASE
7     WHEN stok = 0 THEN 'Perlu Restok'
8     WHEN stok BETWEEN 1 AND 5 THEN 'Stok Menipis'
9     ELSE 'Stok Aman'
10  END AS kondisi_stok
11 FROM
12  barang;
```

Output

	id_barang	nama_barang	stok	status	kondisi_stok
<input type="checkbox"/>	BR001	Tenda Kapasitas 4 Orang	9	Tersedia	Stok Aman
<input type="checkbox"/>	BR002	Kompor Portable	5	Tersedia	Stok Menipis

2.

Query SQL

```
SELECT id_barang, nama_barang, harga_sewa FROM barang ORDER BY harga_sewa DESC LIMIT 5;
```

Output

	id_barang	nama_barang	harga_sewa
<input type="checkbox"/>	BR001	Tenda Kapasitas 4 Orang	50000
<input type="checkbox"/>	BR002	Kompor Portable	25000

3.

Query SQL

```
SELECT p.id_pelanggan, p1.nama_pelanggan, SUM(total_harga) AS total_sewa FROM penyewaan p JOIN pelanggan p1 ON p.id_pelanggan = p1.id_pelanggan GROUP BY p.id_pelanggan, p1.nama_pelanggan HAVING SUM(total_harga) > 500000;
```

Output

