

Nama:Dzakiyya Puteri Aulia

Kelas: XII RPL 1

## Query JOIN

### 1. INNER JOIN

Inner join adalah salah satu jenis operasi penggabungan tabel dalam SQL yang menggabungkan baris dari dua tabel berdasarkan kondisi yang ditentukan, dan hanya mengambil baris yang memiliki nilai yang cocok di kedua tabel tersebut.

Contoh:

```
SELECT
bb.nofak_beli,
bb.tgl_beli,
br.kode_barang,
br.nama_barang,
bb.jumlah_beli,
bb.harga_beli,
(bb.jumlah_beli * bb.harga_beli) AS total_bayar,
tj.jenis_barang
FROM tbl_beli_barang AS bb
JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang
JOIN tbl_jenis AS tj ON br.id_jenis = tj.id_jenis;
```

```
C:\Windows\system32\cmd.exe - mysql -u root
MariaDB [db_jual_xii_rpl1]> SELECT
-> bb.nofak_beli,
-> bb.tgl_beli,
-> br.kode_barang,
-> br.nama_barang,
-> bb.jumlah_beli,
-> bb.harga_beli,
-> (bb.jumlah_beli * bb.harga_beli) AS total_bayar,
-> tj.jenis_barang
-> FROM tbl_beli_barang AS bb
-> JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang
-> JOIN tbl_jenis AS tj ON br.id_jenis = tj.id_jenis;
```

nofak_beli	tgl_beli	kode_barang	nama_barang	jumlah_beli	harga_beli	total_bayar	jenis_barang
1	2023-08-10	1	Laptop	10	9500000	95000000	Elektronik
2	2023-08-11	2	T-Shirt	100	4500000	450000000	Pakaian
3	2023-08-12	3	Coklat	300	800000	240000000	Makanan
4	2023-08-13	4	Air Mineral	500	1500	750000	Minuman
5	2023-08-14	5	Pensil	200	2000	400000	Alat Tulis
6	2023-08-10	6	Cincin Emas	5	1800000	9000000	Perhiasan
7	2023-08-11	7	Meja Kayu	2	450000	900000	Furniture
8	2023-08-12	8	Lipstik	50	20000	1000000	Kosmetik
9	2023-08-13	9	Bola Sepak	10	90000	900000	Olahraga
10	2023-08-14	10	Boneka	30	70000	2100000	Mainan

```
10 rows in set (0.002 sec)

MariaDB [db_jual_xii_rpl1]>
```

## 2. LEFT OUTER JOIN

Left outer join adalah salah satu jenis operasi penggabungan (join) dalam SQL yang mengembalikan semua baris dari tabel di sebelah kiri (LEFT), dan baris yang sesuai dari tabel di sebelah kanan (RIGHT). Jika tidak ada baris yang sesuai dari tabel di sebelah kanan, kolom yang berasal dari tabel kanan akan memiliki nilai NULL dalam hasilnya.

Contoh:

```
SELECT
bb.nofak_beli,
bb.tgl_beli,
bb.kode_barang,
br.nama_barang,
bb.jumlah_beli,
bb.harga_beli,
(bb.jumlah_beli * bb.harga_beli) AS total_bayar
FROM tbl_beli_barang AS bb
LEFT JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang;
```

cmd C:\Windows\system32\cmd.exe - mysql -u root

```
MariaDB [db_jual_xii_rpl1]> SELECT
-> bb.nofak_beli,
-> bb.tgl_beli,
-> bb.kode_barang,
-> br.nama_barang,
-> bb.jumlah_beli,
-> bb.harga_beli,
-> (bb.jumlah_beli * bb.harga_beli) AS total_bayar
-> FROM tbl_beli_barang AS bb
-> LEFT JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang;
```

nofak_beli	tgl_beli	kode_barang	nama_barang	jumlah_beli	harga_beli	total_bayar
1	2023-08-10	1	Laptop	10	9500000	95000000
2	2023-08-11	2	T-Shirt	100	4500000	450000000
3	2023-08-12	3	Coklat	300	800000	240000000
4	2023-08-13	4	Air Mineral	500	1500	750000
5	2023-08-14	5	Pensil	200	2000	400000
6	2023-08-10	6	Cincin Emas	5	1800000	9000000
7	2023-08-11	7	Meja Kayu	2	450000	900000
8	2023-08-12	8	Lipstik	50	20000	1000000
9	2023-08-13	9	Bola Sepak	10	90000	900000
10	2023-08-14	10	Boneka	30	70000	2100000

```
10 rows in set (0.001 sec)

MariaDB [db_jual_xii_rpl1]>
```

### 3. RIGHT OUTER JOIN

Kebalikan dari left outer join right outer join adalah salah satu jenis operasi penggabungan (join) dalam SQL yang menggabungkan data dari dua tabel berdasarkan suatu kondisi yang diberikan, tetapi mengambil semua baris dari tabel yang berada di sisi kanan (right table) dan hanya baris yang sesuai dari tabel di sisi kiri (left table).

Contoh:

```
SELECT
```

```
bb.nofak_beli,
```

```
bb.tgl_beli,
```

```
bb.kode_barang,
```

```
br.nama_barang,
```

```
bb.jumlah_beli,
```

```
bb.harga_beli
```

```
FROM tbl_beli_barang AS bb
```

```
RIGHT JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang;
```

```
C:\Windows\system32\cmd.exe - mysql -u root
```

```
MariaDB [db_jual_xii_rpl1]> SELECT
```

```
-> bb.nofak_beli,
```

```
-> bb.tgl_beli,
```

```
-> bb.kode_barang,
```

```
-> br.nama_barang,
```

```
-> bb.jumlah_beli,
```

```
-> bb.harga_beli
```

```
-> FROM tbl_beli_barang AS bb
```

```
-> RIGHT JOIN tbl_barang AS br ON bb.kode_barang = br.kode_barang;
```

nofak_beli	tgl_beli	kode_barang	nama_barang	jumlah_beli	harga_beli
1	2023-08-10	1	Laptop	10	9500000
2	2023-08-11	2	T-Shirt	100	4500000
3	2023-08-12	3	Coklat	300	800000
4	2023-08-13	4	Air Mineral	500	1500
5	2023-08-14	5	Pensil	200	2000
6	2023-08-10	6	Cincin Emas	5	1800000
7	2023-08-11	7	Meja Kayu	2	450000
8	2023-08-12	8	Lipstik	50	20000
9	2023-08-13	9	Bola Sepak	10	90000
10	2023-08-14	10	Boneka	30	70000

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

```
MariaDB [db_jual_xii_rpl1]>
```

### 4. FULL OUTER JOIN

Full outer join adalah jenis operasi penggabungan dalam SQL yang menggabungkan data dari dua tabel berdasarkan kolom yang sesuai, dan juga mencakup baris yang tidak cocok di kedua tabel tersebut. Dengan kata lain, Full outer join mengambil semua baris dari kedua tabel dan menggabungkannya berdasarkan nilai yang sesuai dalam kolom yang ditentukan, bahkan jika ada baris yang tidak cocok di salah satu tabel.

Contoh:

```
SELECT
COALESCE(b.nofak_beli, j.nofak_jual) AS nofak,
COALESCE(b.tgl_beli, j.tgl_jual) AS tgl_transaksi,
br.nama_barang,
COALESCE(b.jumlah_beli, 0) AS jumlah_pembelian,
COALESCE(j.jumlah_jual, 0) AS jumlah_penjualan,
COALESCE(b.harga_beli, j.harga_jual) AS harga,
COALESCE((b.jumlah_beli * b.harga_beli), 0) AS total_pembelian,
COALESCE((j.jumlah_jual * j.harga_jual), 0) AS total_penjualan
FROM tbl_barang AS br
LEFT JOIN tbl_beli_barang AS b ON br.kode_barang = b.kode_barang
LEFT JOIN tbl_jual_barang AS j ON br.kode_barang = j.kode_barang
ORDER BY COALESCE(b.tgl_beli, j.tgl_jual) ASC;;
```

C:\Windows\system32\cmd.exe - mysql -u root

MariaDB [db\_jual\_xii\_rpl1]> SELECT

```
-> COALESCE(b.nofak_beli, j.nofak_jual) AS nofak,
-> COALESCE(b.tgl_beli, j.tgl_jual) AS tgl_transaksi,
-> br.nama_barang,
-> COALESCE(b.jumlah_beli, 0) AS jumlah_pembelian,
-> COALESCE(j.jumlah_jual, 0) AS jumlah_penjualan,
-> COALESCE(b.harga_beli, j.harga_jual) AS harga,
-> COALESCE((b.jumlah_beli * b.harga_beli), 0) AS total_pembelian,
-> COALESCE((j.jumlah_jual * j.harga_jual), 0) AS total_penjualan
-> FROM tbl_barang AS br
-> LEFT JOIN tbl_beli_barang AS b ON br.kode_barang = b.kode_barang
-> LEFT JOIN tbl_jual_barang AS j ON br.kode_barang = j.kode_barang
-> ORDER BY COALESCE(b.tgl_beli, j.tgl_jual) ASC;
```

nofak	tgl_transaksi	nama_barang	jumlah_pembelian	jumlah_penjualan	harga	total_pembelian	total_penjualan
1	2023-08-10	Laptop	10	5	9500000	95000000	100000000
6	2023-08-10	Cincin Emas	5	1	1800000	9000000	2200000
2	2023-08-11	T-Shirt	100	20	4500000	45000000	160000000
7	2023-08-11	Meja Kayu	2	3	450000	900000	1800000
8	2023-08-12	Lipstik	50	10	20000	1000000	250000
3	2023-08-12	Coklat	300	100	800000	24000000	150000000
9	2023-08-13	Bola Sepak	10	2	90000	900000	140000
4	2023-08-13	Air Mineral	500	200	1500	750000	600000
5	2023-08-14	Pensil	200	50	2000	400000	250000
10	2023-08-14	Boneka	30	5	70000	2100000	400000

10 rows in set (0.001 sec)

MariaDB [db\_jual\_xii\_rpl1]> \_

## 5. PERBEDAAN QUERY JOIN

**INNER JOIN:**

- Menggabungkan baris yang memiliki nilai yang cocok dalam kolom yang digunakan sebagai kondisi JOIN.
- Hanya baris yang memiliki nilai yang cocok dalam kedua tabel yang akan dimasukkan ke dalam hasil penggabungan.
- Baris yang tidak cocok diabaikan.
- Hanya kolom yang sesuai yang akan tergabung.

**LEFT JOIN:**

- Menggabungkan semua baris dari tabel kiri (tabel pertama dalam JOIN).
- Hanya baris dari tabel kanan yang memiliki nilai yang cocok akan dimasukkan.
- Baris yang tidak cocok pada tabel kanan akan diisi dengan nilai NULL.
- Hanya kolom yang sesuai yang akan tergabung.

**RIGHT JOIN:**

- Mirip dengan LEFT JOIN, tetapi sebaliknya.
- Menggabungkan semua baris dari tabel kanan (tabel kedua dalam JOIN).
- Hanya baris dari tabel kiri yang memiliki nilai yang cocok akan dimasukkan.
- Baris yang tidak cocok pada tabel kiri akan diisi dengan nilai NULL.
- Hanya kolom yang sesuai yang akan tergabung.

**FULL OUTER JOIN:**

- Menggabungkan semua baris dari kedua tabel.
- Termasuk semua baris dari tabel kiri dan tabel kanan, serta baris yang cocok.
- Jika tidak ada nilai cocok, kolom yang berasal dari tabel yang tidak memiliki nilai cocok akan diisi dengan nilai NULL.
- Menghasilkan hasil gabungan lengkap dari kedua tabel.