

## PEMROGRAMAN BASIS DATA

### Join (Penggabungan Tabel)

**Kamarudin, M.Kom**

[kamarudin@amikom.ac.id](mailto:kamarudin@amikom.ac.id)

<http://coding4ever.net/>

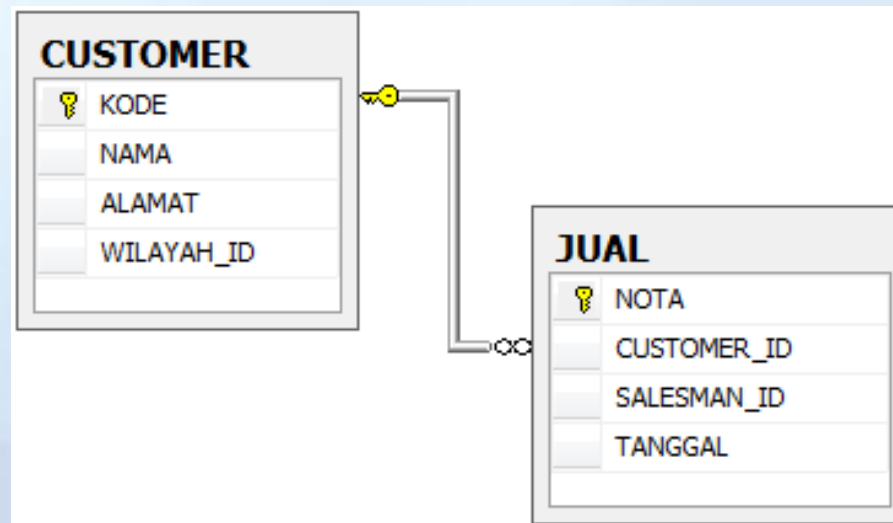
<https://github.com/rudi-krsoftware/open-retail>

# Review Pertemuan Sebelumnya

- ✓ Jelaskan beberapa kemampuan yang dimiliki perintah SELECT seperti projection, selection dan join.
- ✓ Sebutkan tahapan/urutan eksekusi perintah Query (SELECT)
- ✓ Apa perbedaan klausa WHERE dan HAVING?
- ✓ Apa yang dimaksud dengan subquery?
- ✓ Apa yang dimaksud dengan single-row subquery? Sebutkan operator yang digunakan.
- ✓ Apa yang dimaksud dengan multiple-row subquery? Sebutkan operator yang digunakan.

# Join

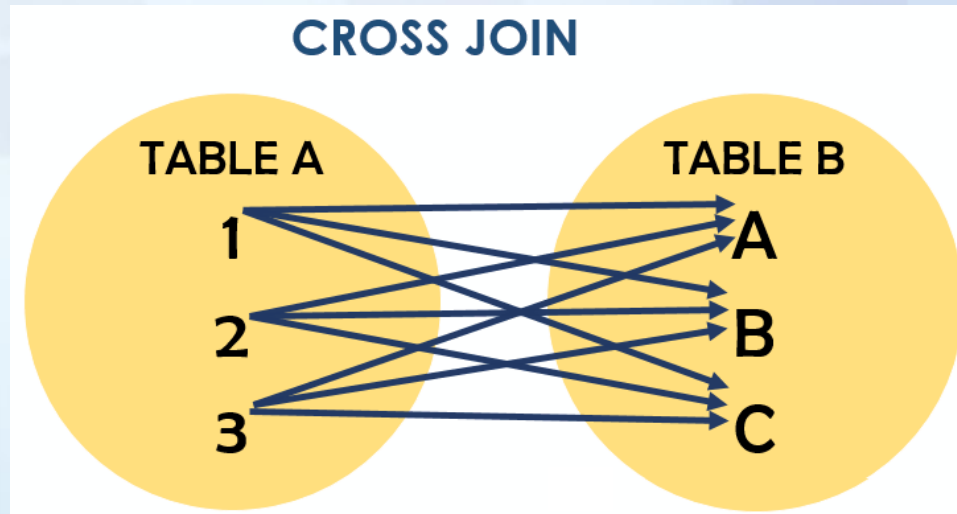
- ✓ Join merupakan sebuah operasi yang digunakan untuk mendapatkan data gabungan dari dua tabel atau lebih yang memiliki relasi.



# Jenis-jenis Operasi Join

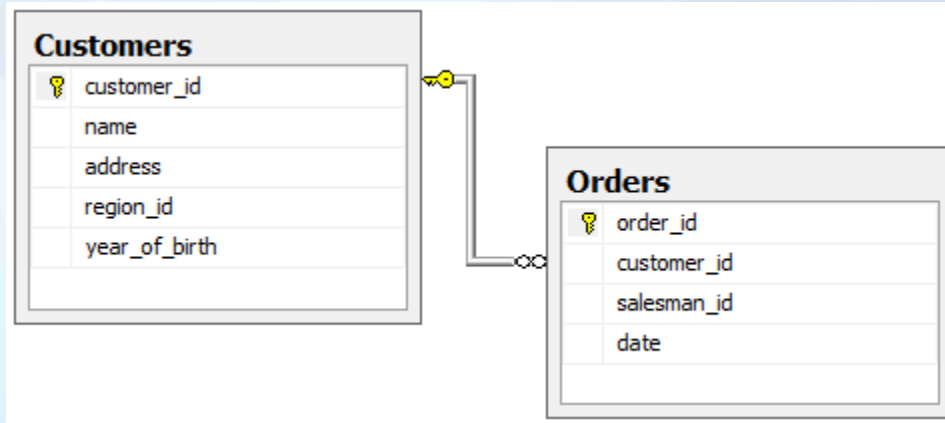
- ✓ Cross Join
- ✓ Inner Join
- ✓ Outer Join :
  - ☐ Left Outer Join
  - ☐ Right Outer Join
  - ☐ Full Outer Join

# Cross Join



- ✓ Menghasilkan kombinasi semua baris yang terdapat dalam tabel-tabel yang digabungkan, baik itu yang berpasangan maupun yang tidak berpasangan. Pada kenyataannya join jenis ini tidak akan pernah dipakai. Meskipun begitu join jenis inilah yang merupakan dasar dari join antar tabel.
- ✓ Banyaknya record dari cross join = jumlah record tabel pertama X jumlah record tabel kedua.

# Cross Join (Lanjutan)



Contoh :

✓ Cara 1

```
SELECT *  
FROM customers CROSS JOIN orders
```

✓ Cara 2

```
SELECT *  
FROM customers, orders
```

# Cross Join (Lanjutan)

**Customers**

CustomerId	Name
CUS-000001	Bagus Nur Hakim
CUS-000002	Putri Kharisma
CUS-000003	Aliany Milawaty

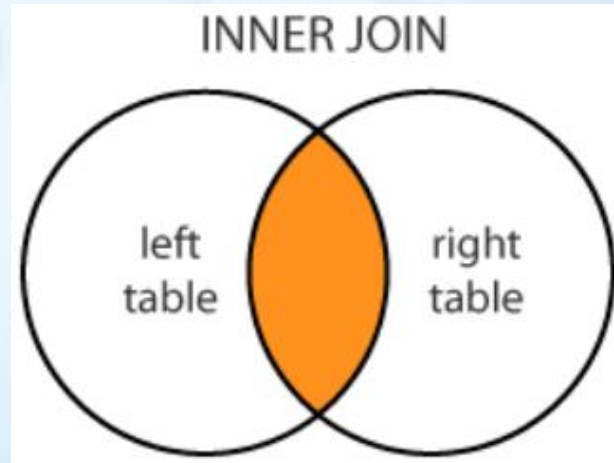
**Orders**

OrderId	CustomerId	Date
20160301-00001	CUS-000002	01-03-2019
20160301-00003	CUS-000001	01-03-2019
20160304-00001	CUS-000003	04-03-2019

**Hasil Cross Join**

CustomerId	Name	OrderId	CustomerId	Date
CUS-000001	Bagus Nur Hakim	20160301-00001	CUS-000002	01-03-2019
CUS-000001	Bagus Nur Hakim	20160301-00003	CUS-000001	01-03-2019
CUS-000001	Bagus Nur Hakim	20160304-00001	CUS-000003	04-03-2019
CUS-000002	Putri Kharisma	20160301-00001	CUS-000002	01-03-2019
CUS-000002	Putri Kharisma	20160301-00003	CUS-000001	01-03-2019
CUS-000002	Putri Kharisma	20160304-00001	CUS-000003	04-03-2019
CUS-000003	Aliany Milawaty	20160301-00001	CUS-000002	01-03-2019
CUS-000003	Aliany Milawaty	20160301-00003	CUS-000001	01-03-2019
CUS-000003	Aliany Milawaty	20160304-00001	CUS-000003	04-03-2019

# Inner Join

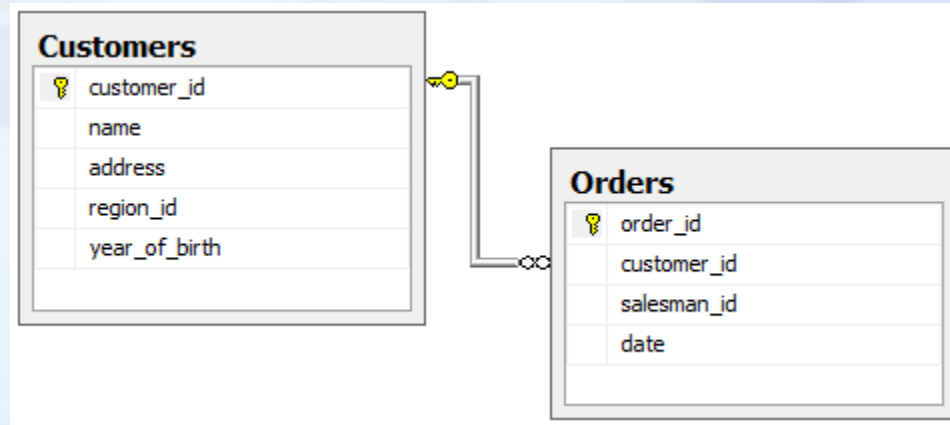


- ✓ Join jenis ini hanya akan menghasilkan output berupa kombinasi baris-baris yang berpasangan saja. Kombinasi baris yang bukan pasangan akan dieliminasi. Baris-baris yang tidak memiliki pasangan pada tabel lainnya juga tidak akan dimunculkan.



# Inner Join (Lanjutan)

Contoh :



✓ Cara 1

```
SELECT *  
FROM customers INNER JOIN orders  
ON customers.customer_id = orders.customer_id
```

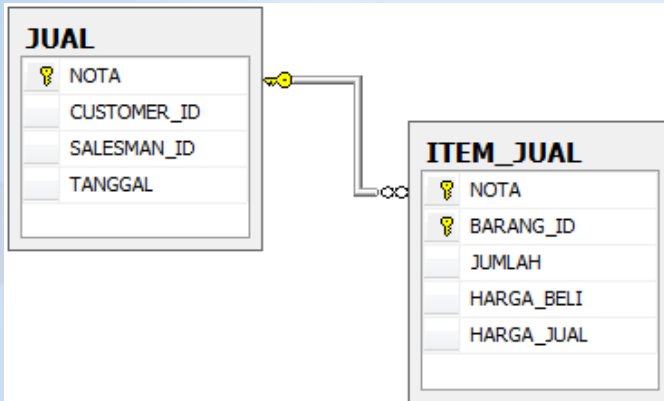
✓ Cara 2

```
SELECT *  
FROM customers, orders  
WHERE customers.customer_id = orders.customer_id
```

# Inner Join (Lanjutan)

## Catatan :

- ✓ Untuk INNER JOIN, Anda dapat menghilangkan kata 'INNER'. Jadi, cukup dengan kata 'JOIN' saja.
- ✓ Jika kolom yang ingin ditampilkan ada di dua tabel atau lebih, maka Anda harus menentukan tabel mana yang diinginkan untuk menghindari terjadinya error "ambiguous column name".



	nota	tanggal	barang_id
1	01100173	2010-12-01 00:00:00.000	20101015000002
2	01100173	2010-12-01 00:00:00.000	20101015000004
3	01100173	2010-12-01 00:00:00.000	20101020000017

```
SELECT jual.nota, tanggal, barang_id
FROM jual INNER JOIN item_jual
ON jual.nota = item_jual.nota
```

# Inner Join (Lanjutan)

- ✓ Gunakan alias jika nama tabel terlalu panjang, sehingga mudah di ingat, dan dari sisi penulisan syntak query lebih ringkas.

```
SELECT jual.nota, tanggal, barang_id  
FROM jual INNER JOIN item_jual ON jual.nota = item_jual.nota  
ORDER BY jual.nota
```

## Contoh menggunakan alias tabel

```
SELECT j.nota, tanggal, barang_id  
FROM jual j INNER JOIN item_jual ij ON j.nota = ij.nota  
ORDER BY j.nota
```

# Inner Join (Lanjutan)

Ilustrasi :

Tabel : Customer

NO	KODE	NAMA
1	0000000019	ARYA COMPUTER
2	0000000055	AMANAH COMPUTER
3	0000000016	SOFT INFO COMPUTER



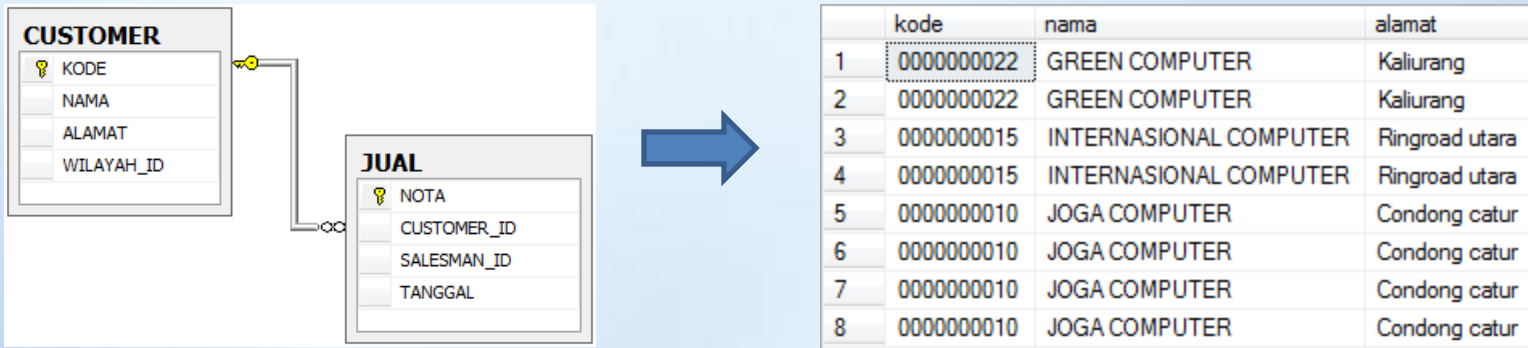
Tabel : Jual

NO	NOTA	CUSTOMER_ID	TANGGAL
1	170091263	0000000019	17-09-2010
2	200091299	0000000019	20-09-2010
3	10024584	0000000055	27-07-2011
4	10024645	0000000055	15-08-2011
5	10024665	0000000055	20-08-2011

NO	CUSTOMER_ID	NAMA	NOTA	TANGGAL
1	0000000019	ARYA COMPUTER	170091263	17-09-2010
2	0000000019	ARYA COMPUTER	200091299	20-09-2010
3	0000000055	AMANAH COMPUTER	10024584	27-07-2011
4	0000000055	AMANAH COMPUTER	10024645	15-08-2011
5	0000000055	AMANAH COMPUTER	10024665	20-08-2011

# Inner Join dan DISTINCT

- ✓ Hasil operasi join, kadang-kadang ada informasi yang berulang. Untuk menghilangkan pengulangan tersebut, gunakan DISTINCT.



```
SELECT customer.kode, customer.nama, customer.alamat
FROM customer INNER JOIN jual ON customer.kode = jual.customer_id
ORDER BY customer.nama
```

	kode	nama	alamat
1	0000000022	GREEN COMPUTER	Kaliurang
2	0000000015	INTERNASIONAL COMPUTER	Ringroad utara
3	0000000010	JOGA COMPUTER	Condong catur

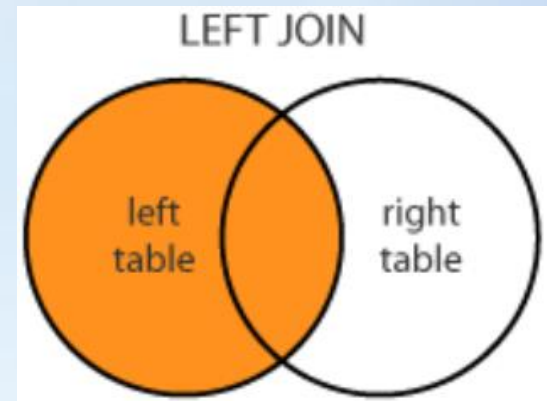
```
SELECT DISTINCT customer.kode, customer.nama, customer.alamat
FROM customer INNER JOIN jual ON customer.kode = jual.customer_id
ORDER BY customer.nama
```

# Outer Join

Hampir sama dengan Inner Join, hanya saja baris yang tidak memiliki pasangan tetap akan ditampilkan, sesuai dengan tipe dari outer join.

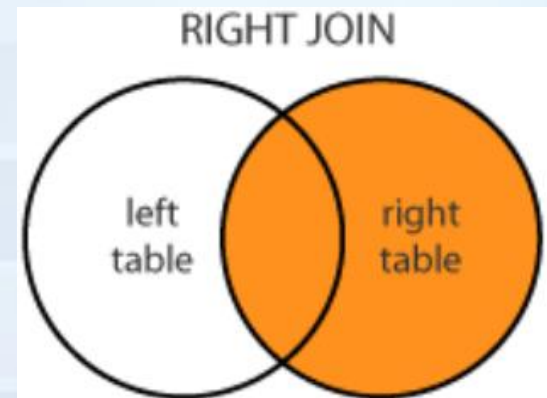
## ✓ LEFT JOIN

LEFT JOIN akan menampilkan semua baris dari tabel sebelah kiri dengan gabungan dari tabel sebelah kanan.



## ✓ RIGHT JOIN

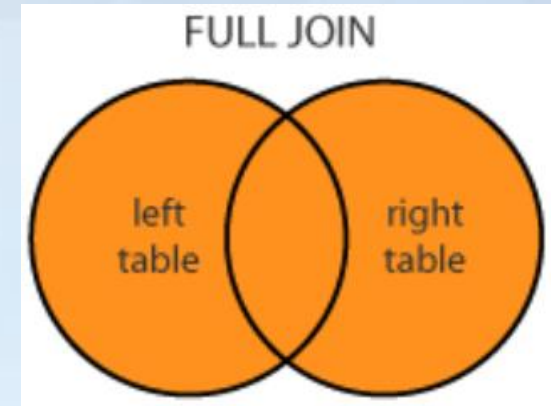
RIGHT JOIN kebalikan dari LEFT JOIN yaitu tabel yang menjadi acuan ada disebelah kanan.



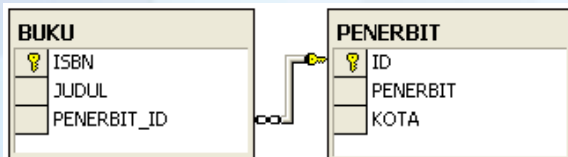
# Outer Join

## ✓ FULL JOIN

Menampilkan hasil join tabel pertama dengan tabel kedua, serta semua record di kedua tabel tersebut. Hasil FULL JOIN sama persis seperti LEFT/RIGHT JOIN tanpa terpengaruh terhadap posisi tabel.



# Contoh Inner, Left & Right Join



TABEL : BUKU

ISBN	JUDUL	PENERBIT_ID
12345	Basis Data	2
12346	Professional C# 2008	2
12347	Belajar Pemrograman C#	2
12348	Java Fundamental I	3
12349	Java Fundamental II	3
12350	Thinking in Java	4
12351	Pengenalan Sistem Operasi Linux	7
12352	Red Hat Linux 8	7
12353	C++ How to Program	2

TABEL : PENERBIT

ID	PENERBIT	KOTA
1	Pustaka Hidayah	Bandung
2	Elex Media Komputindo	Jakarta
3	Andi Offset	Yogyakarta
4	Widya Sarana Informatika	Yogyakarta
5	Amikom	Yogyakarta
6	Wahana Komputer	Jakarta
7	PT Infolinux Media Utama	Jakarta
8	Informatika	Bandung
9	Gava Media	NULL
10	Pustaka Pelajar	NULL

Hasil query menggunakan INNER JOIN

PENERBIT	KOTA	ISBN	JUDUL
Elex Media Komputindo	Jakarta	12345	Basis Data
Elex Media Komputindo	Jakarta	12346	Professional C# 2008
Elex Media Komputindo	Jakarta	12347	Belajar Pemrograman C#
Andi Offset	Yogyakarta	12348	Java Fundamental I
Andi Offset	Yogyakarta	12349	Java Fundamental II
Widya Sarana Informatika	Yogyakarta	12350	Thinking in Java
PT Infolinux Media Utama	Jakarta	12351	Pengenalan Sistem Operasi Linux
PT Infolinux Media Utama	Jakarta	12352	Red Hat Linux 8
Elex Media Komputindo	Jakarta	12353	C++ How to Program

Tuliskan querynya ?

Hasil query menggunakan LEFT/RIGHT JOIN

PENERBIT	KOTA	ISBN	JUDUL
Pustaka Hidayah	Bandung	NULL	NULL
Elex Media Komputindo	Jakarta	12345	Basis Data
Elex Media Komputindo	Jakarta	12346	Professional C# 2008
Elex Media Komputindo	Jakarta	12347	Belajar Pemrograman C#
Elex Media Komputindo	Jakarta	12353	C++ How to Program
Andi Offset	Yogyakarta	12348	Java Fundamental I
Andi Offset	Yogyakarta	12349	Java Fundamental II
Widya Sarana Informatika	Yogyakarta	12350	Thinking in Java
Amikom	Yogyakarta	NULL	NULL
Wahana Komputer	Jakarta	NULL	NULL
PT Infolinux Media Utama	Jakarta	12351	Pengenalan Sistem Operasi Linux
PT Infolinux Media Utama	Jakarta	12352	Red Hat Linux 8
Informatika	Bandung	NULL	NULL
Gava Media	NULL	NULL	NULL
Pustaka Pelajar	NULL	NULL	NULL

Tuliskan querynya ?