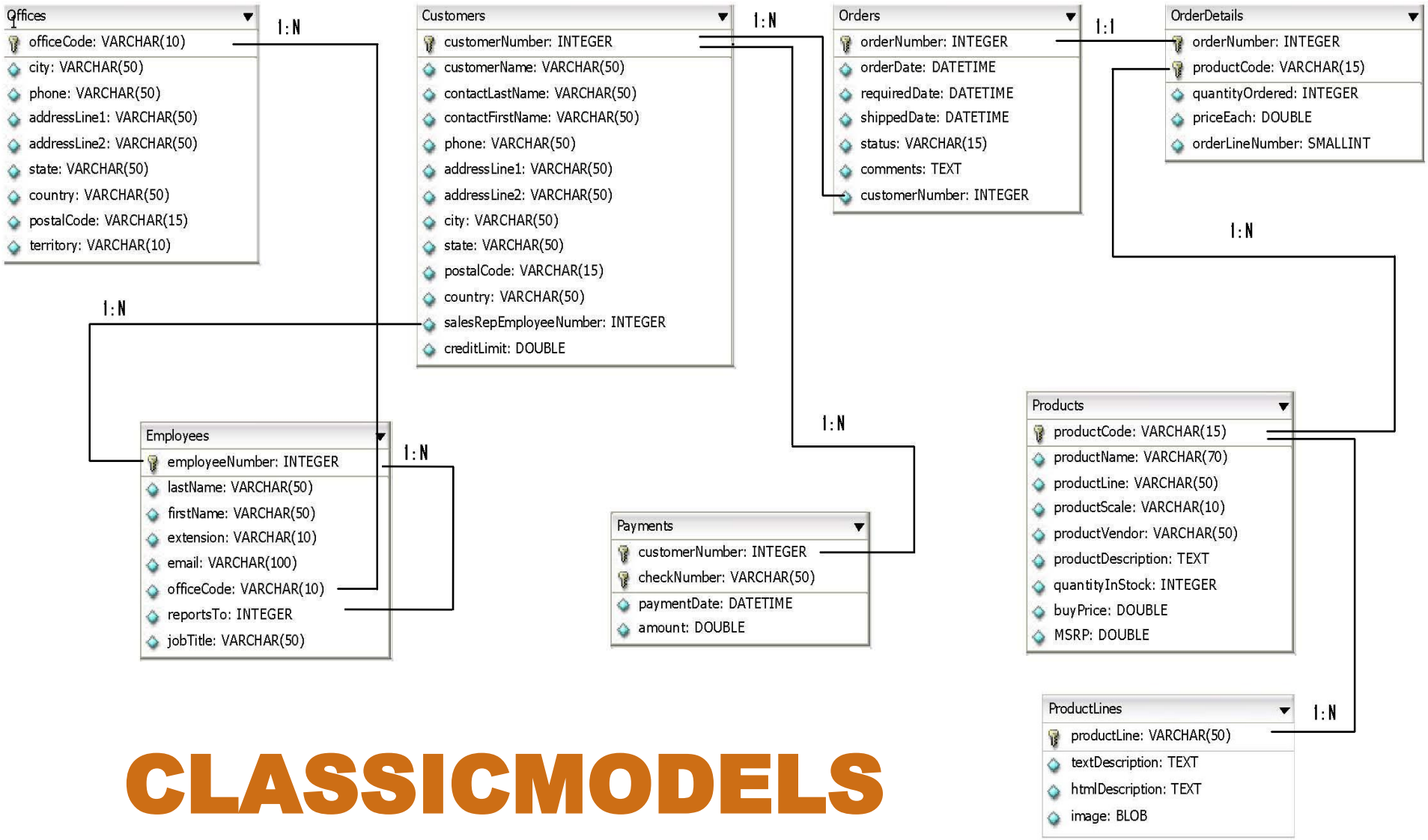
6

JOIN

SKEMA RELASI YANG DIGUNAKAN

CLASSICMODELS



CLASSICMODELS

Jenis Join

- JOIN menggunakan operator “,” (koma)
- CROSS JOIN
- INNER JOIN
- NATURAL JOIN
- SELF JOIN
- LEFT OUTER JOIN
- RIGHT OUTER JOIN
- FULL OUTER JOIN (MySQL tidak support)

“,”, CROSS JOIN & [INNER] JOIN (TANPA KONDISI)

- Perintah “,”, CROSS JOIN dan [INNER] JOIN yang tanpa menggunakan kondisi akan menghasilkan suatu himpunan Cartesian product yang akan meng**kombinasi**kan **semua baris di tabel pertama** dengan **semua baris di tabel kedua**.
- Jika anda menggunakan join menggunakan perintah tersebut, maka jika tabel pertama terdiri dari 5 baris dan tabel kedua terdiri dari 3 baris maka akan menghasilkan baris sebanyak 3.
- Di MySQL, ketiga jenis join tersebut akan menghasilkan hasil yang sama

“,”, CROSS JOIN & [INNER] JOIN (TANPA KONDISI)

```
SELECT EmployeeNumber,
       LastName, OfficeCode
FROM employees
```

EmployeeNumber	LastName	OfficeCode
1002	Murphy	1
1056	Patterson	1
1076	Firrelli	1
1088	Patterson	6
1102	Bondur	4
1143	Bow	1
	...	
1612	Marsh	6
1619	King	6
1621	Nishi	5
1625	Kato	5
1702	Gerard	4

23 Rows

```
SELECT OfficeCode,
       City, Country
FROM offices
```

OfficeCode	City	Country
1	San Francisco	USA
2	Boston	USA
3	NYC	USA
4	Paris	France
5	Tokyo	Japan
6	Sydney	Australia
7	London	UK

7 Rows

“,”, CROSS JOIN & [INNER] JOIN (TANPA KONDISI)

CONTOH 1:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e, offices o;
```

Perhatikan Alias Table:
Tabel Employees → e
Tabel Offices → o

CONTOH 2:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e CROSS JOIN offices o;
```

CONTOH 3:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e INNER JOIN offices o;
```

Hasil Eksekusi

“,”, CROSS JOIN & [INNER] JOIN (TANPA KONDISI)

EmployeeNumber	LastName	OfficeCode	OfficeCode	City	Country
1401	Castillo	4	3	NYC	USA
1401	Castillo	4	4	Paris	France
1401	Castillo	4	5	Tokyo	Japan
1401	Castillo	4	6	Sydney	Australia
1401	Castillo	4	7	London	UK
1501	Bott	7	1	San Francisco	USA
1501	Bott	7	2	Boston	USA
1501	Bott	7	3	NYC	USA
1501	Bott	7	4	Paris	France
...					
1625	Kato	5	6	Sydney	Australia
1625	Kato	5	7	London	UK
1702	Gerard	4	1	San Francisco	USA
1702	Gerard	4	2	Boston	USA
1702	Gerard	4	3	NYC	USA
1702	Gerard	4	4	Paris	France
1702	Gerard	4	5	Tokyo	Japan
1702	Gerard	4	6	Sydney	Australia
1702	Gerard	4	7	London	UK

161 Rows ← 23 (employees) x 7 (offices)

“,”, CROSS JOIN & [INNER] JOIN (DENGAN KONDISI)

- Biasanya suatu relasi dinyatakan dengan suatu kesamaan nilai yaitu kesamaan 1 atau banyak field di tabel pertama dengan 1 atau banyak field di tabel kedua.
- Dengan menggunakan kondisi relasi, maka baris yang dihasilkan bisa difilter berdasarkan kondisi relasi tersebut.
- Hal ini disebut sebagai EQUI-JOIN (Join yang didasarkan persamaan nilai).
- Untuk join yang menggunakan koma, gunakan clausa WHERE untuk mendefinisikan kondisi relasi
- Untuk join yang menggunakan CROSS JOIN atau [INNER] JOIN, gunakan clausa ON untuk mendefinisikan kondisi relasi.

",", CROSS JOIN & [INNER] JOIN (DENGAN KONDISI)

CONTOH 1:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e, offices o
WHERE e.OfficeCode = o.OfficeCode;
```

CONTOH 2:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e CROSS JOIN offices o
ON e.OfficeCode = o.OfficeCode;
```

CONTOH 3:

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e INNER JOIN offices o
ON e.OfficeCode = o.OfficeCode;
```

Hasil Eksekusi

“,”, CROSS JOIN & [INNER] JOIN (DENGAN KONDISI)

EmployeeNumber	LastName	OfficeCode	OfficeCode	City	Country
1002	Murphy	1	1	San Francisco	USA
1056	Patterson	1	1	San Francisco	USA
1076	Firrelli	1	1	San Francisco	USA
1088	Patterson	6	6	Sydney	Australia
1102	Bondur	4	4	Paris	France
1143	Bow	1	1	San Francisco	USA
1165	Jennings	1	1	San Francisco	USA
...					
1611	Fixter	6	6	Sydney	Australia
1612	Marsh	6	6	Sydney	Australia
1619	King	6	6	Sydney	Australia
1621	Nishi	5	5	Tokyo	Japan
1625	Kato	5	5	Tokyo	Japan
1702	Gerard	4	4	Paris	France

23 Rows

Field yang Sama

NATURAL JOIN

- NATURAL JOIN digunakan untuk menjoinkan 2 tabel atau lebih yang mempunyai 1 atau banyak kolom yang mempunyai nama yang sama dalam semua tabelnya.
- Jika tabel 1 mempunyai 2 kolom yang mempunyai nama field yang sama dengan di tabel 2, maka NATURAL JOIN akan melakukan kondisi relasi berdasarkan kedua field tersebut.
- Dengan NATURAL JOIN, anda tidak perlu mendefinisikan kondisi relasi.

NATURAL JOIN

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e NATURAL JOIN offices o;
```

EmployeeNumber	LastName	OfficeCode	OfficeCode	City	Country
1002	Murphy	1	1	San Francisco	USA
1056	Patterson	1	1	San Francisco	USA
1076	Firrelli	1	1	San Francisco	USA
1088	Patterson	6	6	Sydney	Australia
1102	Bondur	4	4	Paris	France
1143	Bow	1	1	San Francisco	USA
1165	Jennings	1	1	San Francisco	USA
...					
1611	Fixter	6	6	Sydney	Australia
1612	Marsh	6	6	Sydney	Australia
1619	King	6	6	Sydney	Australia
1621	Nishi	5	5	Tokyo	Japan
1625	Kato	5	5	Tokyo	Japan
1702	Gerard	4	4	Paris	France

23 Rows

CLAUSA USING()

- Clausa USING() digunakan untuk menyingkat kondisi relasi yang nama kolomnya identik.

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e JOIN offices o
ON e.OfficeCode = o.OfficeCode;
```

MENJADI :

```
SELECT e.EmployeeNumber, e.LastName, e.OfficeCode,
       o.OfficeCode, o.City, o.Country
FROM employees e JOIN offices o
USING(OfficeCode);
```

CLAUSA USING()

Dengan menggunakan USING() pada JOIN, maka NATURAL JOIN sebenarnya sama dengan JOIN yang menggunakan USING untuk semua kolom yang identik pada kedua tabelnya.

JOIN DENGAN BANYAK KONDISI

- Join dengan banyak kondisi bisa dilakukan dengan menggunakan operator boolean seperti AND. Operator OR bisa saja digunakan, tetapi umumnya menggunakan operator AND.

JOIN DENGAN BANYAK KONDISI

Kondisi 1
(T1.a = T2.a)

Tabel T1

a	b	c
1	1	1
1	2	2
2	1	3
2	2	4
2	3	5

Tabel T2

a	b	d
1	1	11
1	2	12
2	1	13
2	2	14
2	3	15

Kondisi 2
(T1.b = T2.b)

JOIN DENGAN BANYAK KONDISI

CONTOH 1:

```
SELECT *
FROM t1, t2
WHERE t1.a = t2.a and t1.b = t2.b;
```

Semua Kolom Ditampilkan

a	b	c	a	b	d
1	1	1	1	1	11
1	2	2	1	2	12
2	1	3	2	1	13
2	2	4	2	2	14
2	3	5	2	3	15

CONTOH 2:

```
SELECT *
FROM t1 JOIN t2
ON t1.a = t2.a and t1.b = t2.b;
```

Kolom duplikat tidak ditampilkan

a	b	c	d
1	1	1	11
1	2	2	12
2	1	3	13
2	2	4	14
2	3	5	15

CONTOH 3:

```
SELECT *
FROM t1 NATURAL JOIN t2;
```

CONTOH 3:

```
SELECT *
FROM t1 JOIN t2 USING (a,b)
```

JOIN LEBIH DARI 2 TABEL

Contoh Kasus :

Berdasarkan skema relasi yang ada di halaman 3, tampilkan data nomor pesanan(, nama pelanggan dan nama pegawai yang menangani pemesanan tersebut

ordernumber	customername	firstname
10100	Online Diecast Creations Co.	Steve
10101	Blauer See Auto, Co.	Barry
	...	
10423	Petit Auto	Pamela
10424	Euro+ Shopping Channel	Gerard
10425	La Rochelle Gifts	Gerard

326 Rows

JOIN LEBIH DARI 2 TABEL

Solusi :

- Query yang dibutuhkan membutuhkan 3 field yaitu OrderNumber dari tabel **Orders**, CustomerName dari tabel **Customers** dan FirstName dari tabel **Employees**.
- Berdasarkan skemar relasi, ketiga tabel tersebut dihubungkan dengan 2 relasi yaitu :
  - Orders.CustomerNumber ←→ Customers.CustomerNumber
  - Customer.salesRepEmployeeNumber ←→Employees.EmployeesNumber

JOIN LEBIH DARI 2 TABEL

SOLUSI 1 (MENGGUNAKAN KOMA DAN WHERE):

```
SELECT o.orderNumber,c.customerName,e.firstname
FROM orders o, customers c, employees e
WHERE o.customerNumber = c.customerNumber AND
      c.salesRepEmployeeNumber = e.employeeNumber;
```

SOLUSI 2 (MENGGUNAKAN JOIN ON):

```
SELECT o.ordernumber,c.customername,e.firstname
FROM orders o JOIN customers c ON o.customerNumber = c.customerNumber
JOIN employees e ON c.salesRepEmployeeNumber = e.employeeNumber;
```

SOLUSI 3 (MENGGUNAKAN NATURAL JOIN DAN JOIN):

```
SELECT o.ordernumber,c.customername,e.firstname
FROM orders o NATURAL JOIN customers c
JOIN employees e ON c.salesRepEmployeeNumber = e.employeeNumber
```

SELF JOIN

- Self join adalah suatu kondisi dimana suatu tabel berelasi tabel itu sendiri.
- Dalam self join, penggunaan alias tabel PASTI digunakan.
- Contoh kasus :
  - Tampilan data pegawai lengkap dengan data atasannya. Atasannya berasal dari tabel yang sama.
  - Perhatikan pada skema relasi ClassicModel, pada table Employees ada field reportsTo yang berelasi dengan field CustomerNumber pada tabel yang sama.
  - Bisa kita anggap bahwa kolom reportsTo merupakan kolom yang menyatakan siapa yang menjadi atasan seorang pegawai.

SELF JOIN

```
SOLUSI 1 (MENGGUNAKAN KOMA DAN WHERE):
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg, employees ats
WHERE peg.reportsTo = ats.employeeNumber
```

```
SOLUSI 2 (JOIN ON):
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
```

SELF JOIN

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
1056	Mary Patterson	1002	1002	Diane Murphy
1076	Jeff Firrelli	1002	1002	Diane Murphy
1088	William Patterson	1056	1056	Mary Patterson
1102	Gerard Bondur	1056	1056	Mary Patterson
...				
1621	Mami Nishi	1056	1056	Mary Patterson
1625	Yoshimi Kato	1621	1621	Mami Nishi
1702	Martin Gerard	1102	1102	Gerard Bondur

22 Rows

Pegawai 1002 Diane Murphi tidak ditampilkan karena nilai field reportsTo bernilai NULL, berarti dia adalah pegawai tertinggi (tidak mempunyai atasan).

OUTER JOIN

- OUTER JOIN adalah suatu join yang akan menampilkan baris hasil JOIN (equi-join) dilengkapi dengan baris yang tidak mempunyai relasi di kedua tabel.
- Ada tiga jenis OUTER JOIN, yaitu
  - LEFT [OUTER] JOIN : menampilkan semua hasil equi join, lengkap dengan baris dari tabel kiri (tabel 1) yang tidak mempunyai relasi/pasangan dari tabel kanan (tabel 2). Jika relasi tidak terpenuhi, maka kolom yang berasal dari tabel kanan akan diberi nilai NULL
  - RIGHT [OUTER] JOIN : menampilkan semua hasil equi join, lengkap dengan baris dari tabel kanan (tabel 2) yang tidak mempunyai relasi/pasangan dari tabel kiri(tabel 1). Jika relasi tidak terpenuhi, maka kolom yang berasal dari tabel kiri akan diberi nilai NULL
  - FULL [OUTER] JOIN : Baris Hasil JOIN + LEFT OUTER JOIN + RIGHT OUTER JOIN. MySQL tidak mendukung FULL OUTER JOIN, tetapi bisa diemulasikan.

LEFT [OUTER] JOIN

Perhatikan SQL Berikut :

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg LEFT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
```

SQL tersebut akan menampilkan semua data pegawai lengkap dengan nama atasan. Perbedaan dengan SELF JOIN adalah SEMUA baris di tabel kiri (Employees peg) akan ditampilkan walaupun tidak punya relasi ke tabel kanan (Employees ats)

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
1002	Diane Murphy	*NULL*	*NULL*	
1056	Mary Patterson	1002	1002	Diane Murphy
1076	Jeff Firrelli	1002	1002	Diane Murphy
1088	William Patterson	1056	1056	Mary Patterson
1102	Gerard Bondur	1056	1056	Mary Patterson
1143	Anthony Bow	1056	1056	Mary Patterson
1165	Leslie Jennings	1143	1143	Anthony Bow
1166	Leslie Thompson	1143	1143	Anthony Bow
1188	Julie Firrelli	1143	1143	Anthony Bow
1216	Steve Patterson	1143	1143	Anthony Bow
1286	Foon Yue Tseng	1143	1143	Anthony Bow
1323	George Vanauf	1143	1143	Anthony Bow
1337	Loui Bondur	1102	1102	Gerard Bondur
1370	Gerard Hernandez	1102	1102	Gerard Bondur
1401	Pamela Castillo	1102	1102	Gerard Bondur
1501	Larry Bott	1102	1102	Gerard Bondur
1504	Barry Jones	1102	1102	Gerard Bondur
1611	Andy Fixter	1088	1088	William Patterson
1612	Peter Marsh	1088	1088	William Patterson
1619	Tom King	1088	1088	William Patterson
1621	Mami Nishi	1056	1056	Mary Patterson
1625	Yoshimi Kato	1621	1621	Mami Nishi
1702	Martin Gerard	1102	1102	Gerard Bondur

Pegawai yang tidak mempunyai relasi ke tabel Atasan

Baris tabel Kiri yang tidak punya relasi di tabel Kanan

Pegawai yang mempunyai relasi ke tabel Atasan

Baris tabel kiri yang mempunyai relasi di tabel Kanan

LEFT [OUTER] JOIN

Jika anda ingin mengambil baris data dari tabel kiri yang tidak punya relasi di tabel kanan (data pegawai yang tidak mempunyai atasan), maka yang perlu ditambahkan adalah filter terhadap field yang bersumber dari tabel kanan (Employees ats) yang berisi NULL.

LEFT [OUTER] JOIN

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg LEFT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
WHERE ats.employeeNumber IS NULL
```

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
1002	Diane Murphy	*NULL*	*NULL*	

RIGHT [OUTER] JOIN

Perhatikan SQL Berikut :

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg RIGHT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
```

SQL tersebut akan menampilkan semua data pegawai lengkap dengan nama atasan. Perbedaan dengan SELF JOIN adalah SEMUA baris di tabel kanan (Employees ats) akan ditampilkan walaupun tidak punya relasi ke tabel kiri (Employees peg)

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
1056	Mary Patterson	1002	1002	Diane Murphy
1076	Jeff Firrelli	1002	1002	Diane Murphy
1088	William Patterson	1056	1056	Mary Patterson
1102	Gerard Bondur	1056	1056	Mary Patterson
1143	Anthony Bow	1056	1056	Mary Patterson
1621	Mami Nishi	1056	1056	Mary Patterson
NULL		*NULL*	1076	Jeff Firrelli
1611	Andy Fixter	1088	1088	William Patterson
		. . .		
1323	George Vanauf	1143	1143	Anthony Bow
NULL		*NULL*	1165	Leslie Jennings
NULL		*NULL*	1166	Leslie Thompson
NULL		*NULL*	1188	Julie Firrelli
NULL		*NULL*	1216	Steve Patterson
NULL		*NULL*	1286	Foon Yue Tseng
NULL		*NULL*	1323	George Vanauf
NULL		*NULL*	1337	Loui Bondur
NULL		*NULL*	1370	Gerard Hernandez
NULL		*NULL*	1401	Pamela Castillo
NULL		*NULL*	1501	Larry Bott
NULL		*NULL*	1504	Barry Jones
NULL		*NULL*	1611	Andy Fixter
NULL		*NULL*	1612	Peter Marsh
NULL		*NULL*	1619	Tom King
1625	Yoshimi Kato	1621	1621	Mami Nishi
NULL		*NULL*	1625	Yoshimi Kato
NULL		*NULL*	1702	Martin Gerard

Pegawai (Atasan) yang tidak mempunyai relasi ke tabel Pegawai (Bawahan)

Baris Tabel Kanan yang tidak mempunyai relasi di Tabel Kiri

RIGHT [OUTER] JOIN

Jika anda ingin baris dari tabel kanan yang tidak mempunyai relasi ke tabel kiri (data pegawai yang tidak mempunyai bawahan), maka yang perlu ditambahkan adalah filter terhadap field yang bersumber dari tabel kiri (Employees peg) yang berisi NULL.

RIGHT [OUTER] JOIN

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg RIGHT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
WHERE peg.employeeNumber IS NULL
```

RIGHT [OUTER] JOIN

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
NULL		*NULL*	1076	Jeff Firrelli
NULL		*NULL*	1165	Leslie Jennings
NULL		*NULL*	1166	Leslie Thompson
NULL		*NULL*	1188	Julie Firrelli
NULL		*NULL*	1216	Steve Patterson
NULL		*NULL*	1286	Foon Yue Tseng
NULL		*NULL*	1323	George Vanauf
NULL		*NULL*	1337	Loui Bondur
NULL		*NULL*	1370	Gerard Hernandez
NULL		*NULL*	1401	Pamela Castillo
NULL		*NULL*	1501	Larry Bott
NULL		*NULL*	1504	Barry Jones
NULL		*NULL*	1611	Andy Fixter
NULL		*NULL*	1612	Peter Marsh
NULL		*NULL*	1619	Tom King
NULL		*NULL*	1625	Yoshimi Kato
NULL		*NULL*	1702	Martin Gerard

FULL OUTER JOIN

- FULL OUTER JOIN tidak disupport oleh MySQL.
- Hasil suatu join FULL OUTER JOIN akan menghasilkan baris tabel kiri yang mempunyai relasi di tabel kanan + baris tabel kiri yang tidak mempunyai relasi di tabel kanan + baris tabel kanan yang tidak mempunyai relasi di tabel kiri
- FULL OUTER JOIN merupakan :
  - Gabungan antara LEFT OUTER JOIN dengan hasil RIGHT OUTER JOIN yang tabel kirinya bernilai NULL

  Atau
  - Gabungan antara RIGHT OUTER JOIN dengan hasil LEFT OUTER JOIN yang tabel kanannya bernilai NULL

FULL OUTER JOIN

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg LEFT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber

UNION

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg RIGHT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
WHERE peg.employeeNumber IS NULL
```

FULL OUTER JOIN

```
SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg RIGHT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber

UNION

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
FROM employees peg LEFT JOIN employees ats
ON peg.reportsTo = ats.employeeNumber
WHERE ats.employeeNumber IS NULL
```

FULL OUTER JOIN

NoPegawai	NamaPegawai	reportsTo	NoAtasan	NamaAtasan
1002	Diane Murphy	*NULL*	*NULL*	
1056	Mary Patterson	1002	1002	Diane Murphy
1076	Jeff Firrelli	1002	1002	Diane Murphy
1088	William Patterson	1056	1056	Mary Patterson
1102	Gerard Bondur	1056	1056	Mary Patterson
1143	Anthony Bow	1056	1056	Mary Patterson
1165	Leslie Jennings	1143	1143	Anthony Bow
1166	Leslie Thompson	1143	1143	Anthony Bow
1188	Julie Firrelli	1143	1143	Anthony Bow
1216	Steve Patterson	1143	1143	Anthony Bow
1286	Foon Yue Tseng	1143	1143	Anthony Bow
1323	George Vanauf	1143	1143	Anthony Bow
1337	Loui Bondur	1102	1102	Gerard Bondur
1370	Gerard Hernandez	1102	1102	Gerard Bondur
1401	Pamela Castillo	1102	1102	Gerard Bondur
1501	Larry Bott	1102	1102	Gerard Bondur
1504	Barry Jones	1102	1102	Gerard Bondur
1611	Andy Fixter	1088	1088	William Patterson
1612	Peter Marsh	1088	1088	William Patterson
1619	Tom King	1088	1088	William Patterson
1621	Mami Nishi	1056	1056	Mary Patterson
1625	Yoshimi Kato	1621	1621	Mami Nishi
1702	Martin Gerard	1102	1102	Gerard Bondur
NULL		*NULL*	1076	Jeff Firrelli
NULL		*NULL*	1165	Leslie Jennings
NULL		*NULL*	1166	Leslie Thompson
NULL		*NULL*	1188	Julie Firrelli
NULL		*NULL*	1216	Steve Patterson
NULL		*NULL*	1286	Foon Yue Tseng
NULL		*NULL*	1323	George Vanauf
NULL		*NULL*	1337	Loui Bondur
NULL		*NULL*	1370	Gerard Hernandez
NULL		*NULL*	1401	Pamela Castillo
NULL		*NULL*	1501	Larry Bott
NULL		*NULL*	1504	Barry Jones
NULL		*NULL*	1611	Andy Fixter
NULL		*NULL*	1612	Peter Marsh
NULL		*NULL*	1619	Tom King
NULL		*NULL*	1625	Yoshimi Kato
NULL		*NULL*	1702	Martin Gerard

LATIHAN-LATIHAN

- Cari data pemesanan yang dilakukan oleh customer yang berasal dari negara Austria

orderNumber	CustomerNumber	CUstomerName	country
10119	382	Salzburg Collectables	Austria
10164	452	Mini Auto Werke	Austria
10170	452	Mini Auto Werke	Austria
10269	382	Salzburg Collectables	Austria
10341	382	Salzburg Collectables	Austria
10392	452	Mini Auto Werke	Austria
10419	382	Salzburg Collectables	Austria

7 Rows

LATIHAN-LATIHAN

- Tampilkan data item pemesanan (OrderDetails) lengkap dengan nama produk (ProductName) dan deskripsi jenis produknya (textDescription)

OrderNumber	ProductCode	ProductName	textdescription
10100	S18_1749	1917 Grand Touring Sedan	Our Vintage Car models realistically portray autom...
10100	S18_2248	1911 Ford Town Car	Our Vintage Car models realistically portray autom...
10100	S18_4409	1932 Alfa Romeo 8C2300 Spider Sport	Our Vintage Car models realistically portray autom...
10100	S24_3969	1936 Mercedes Benz 500k Roadster	Our Vintage Car models realistically portray autom...
10101	S18_2325	1932 Model A Ford J-Coupe	Our Vintage Car models realistically portray autom...
10101	S18_2795	1928 Mercedes-Benz SSK	Our Vintage Car models realistically portray autom...
10101	S24_1937	1939 Chevrolet Deluxe Coupe	Our Vintage Car models realistically portray autom...
10101	S24_2022	1938 Cadillac V-16 Presidential Limousine	Our Vintage Car models realistically portray autom...
10102	S18_1342	1937 Lincoln Berline	Our Vintage Car models realistically portray autom...
10102	S18_1367	1936 Mercedes-Benz 500K Special Roadster	Our Vintage Car models realistically portray autom...
10103	S10_1949	1952 Alpine Renault 1300	Attention car enthusiasts: Make your wildest car o...

2,996 Rows

LATIHAN-LATIHAN

- Tampilkan data produk yang pernah dipesan. Tampilkan kode produk, nama produk dan deskripsinya.

ProductCode	ProductName	textdescription
S18_1749	1917 Grand Touring Sedan	Our Vintage Car models realistically portray autom...
S18_2248	1911 Ford Town Car	Our Vintage Car models realistically portray autom...
S18_4409	1932 Alfa Romeo 8C2300 Spider Sport	Our Vintage Car models realistically portray autom...
S24_3969	1936 Mercedes Benz 500k Roadster	Our Vintage Car models realistically portray autom...
S18_2325	1932 Model A Ford J-Coupe	Our Vintage Car models realistically portray autom...
S18_2795	1928 Mercedes-Benz SSK	Our Vintage Car models realistically portray autom...
S24_1937	1939 Chevrolet Deluxe Coupe	Our Vintage Car models realistically portray autom...
S24_2022	1938 Cadillac V-16 Presidential Limousine	Our Vintage Car models realistically portray autom...
S18_1342	1937 Lincoln Berline	Our Vintage Car models realistically portray autom...
S18_1367	1936 Mercedes-Benz 500K Special Roadster	Our Vintage Car models realistically portray autom...
S10_1949	1952 Alpine Renault 1300	Attention car enthusiasts: Make your wildest car o...
S10_4962	1962 LanciaA Delta 16V	Attention car enthusiasts: Make your wildest car o...

109 Rows

LATIHAN-LATIHAN

- Cara data customer yang belum pernah melakukan pembayaran.

customerNumber	CustomerName
125	Havel & Zbyszek Co
168	American Souvenirs Inc
169	Porto Imports Co.
206	Asian Shopping Network, Co
223	Natürlich Autos
237	ANG Resellers
247	Messner Shopping Network
273	Franken Gifts, Co
293	BG&E Collectables
303	Schuyler Imports
307	Der Hund Imports
335	Cramer Spezialitäten, Ltd
348	Asian Treasures, Inc.
356	SAR Distributors, Co
361	Kommission Auto
369	Lisboa Souveniers, Inc
376	Precious Collectables
409	Stuttgart Collectable Exchange
443	Feuer Online Stores, Inc
459	Warburg Exchange
465	Anton Designs, Ltd.
477	Mit Vergnügen & Co.
480	Kremlin Collectables, Co.
481	Raanan Stores, Inc

24 Rows

LATIHAN-LATIHAN

- Tampilkan data pemesanan barang lengkap dengan keuntungan yang didapat

OrderNumber	productcode	productname	HargaBeli	HargaJual	Keuntungan	Qty	TotalKeuntungan
10100	S18_1749	1917 Grand Touring Sedan	86.7	136	49.3	30	1479
10100	S18_2248	1911 Ford Town Car	33.3	55.09	21.79	50	1089.5
10100	S18_4409	1932 Alfa Romeo 8C2300 Spider Sport	43.26	75.46	32.2	22	708.4
10100	S24_3969	1936 Mercedes Benz 500k Roadster	21.75	35.29	13.54	49	663.46
10101	S18_2325	1932 Model A Ford J-Coupe	58.48	108.06	49.58	25	1239.5
10101	S18_2795	1928 Mercedes-Benz SSK	72.56	167.06	94.5	26	2457

2,996 Rows