



RELATIONSHIP

ToKu404



RESPONSE

5 Poin



table: books

id	book_name
1	tereliye
2	hujan
3	5 cm

table: genres

id_genre	genre	id_book
1	romance	1
2	fantasy	1
3	romance	2

```
SELECT books.book_name, genres.genre from books  
left join genres on books.id = genres.id_book;
```

Gambarkan hasil dari query diatas!



PENGANTAR RELATIONSHIP



RELATIONSHIP

Relasi biasanya digunakan untuk menggambarkan hubungan yang ada antara tabel dalam database relasional.

Relationship antara dua tabel database terjadi jika salah satu tabel memiliki **foreign key** yang mereferensikan **primary key** dari tabel lain.



Tabel yang berisi
primary key akan
disebut **Parent
Tabel** untuk tabel
referensi.

Table: Orders

order_id	product	total	customer_id
1	Paper	500	5
2	Pen	10	2
3	Marker	120	3
4	Books	1000	1
5	Erasers	20	4

Foreign Key

Tabel yang berisi
foreign key akan
disebut **Child
Table**

Table: Customers

id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE



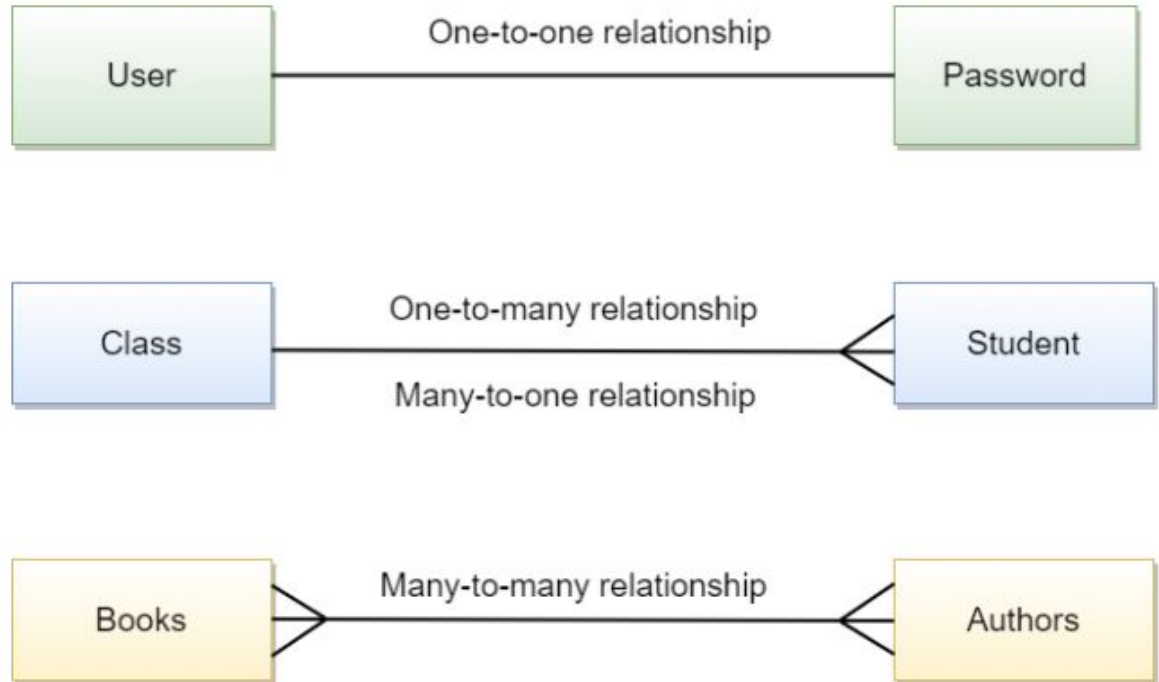
Entity Relational Diagram (ERD)

Entity Relational Diagram (ERD) atau ER Diagram, merupakan representasi grafis tentang bagaimana entitas berhubungan satu sama lain dalam database.

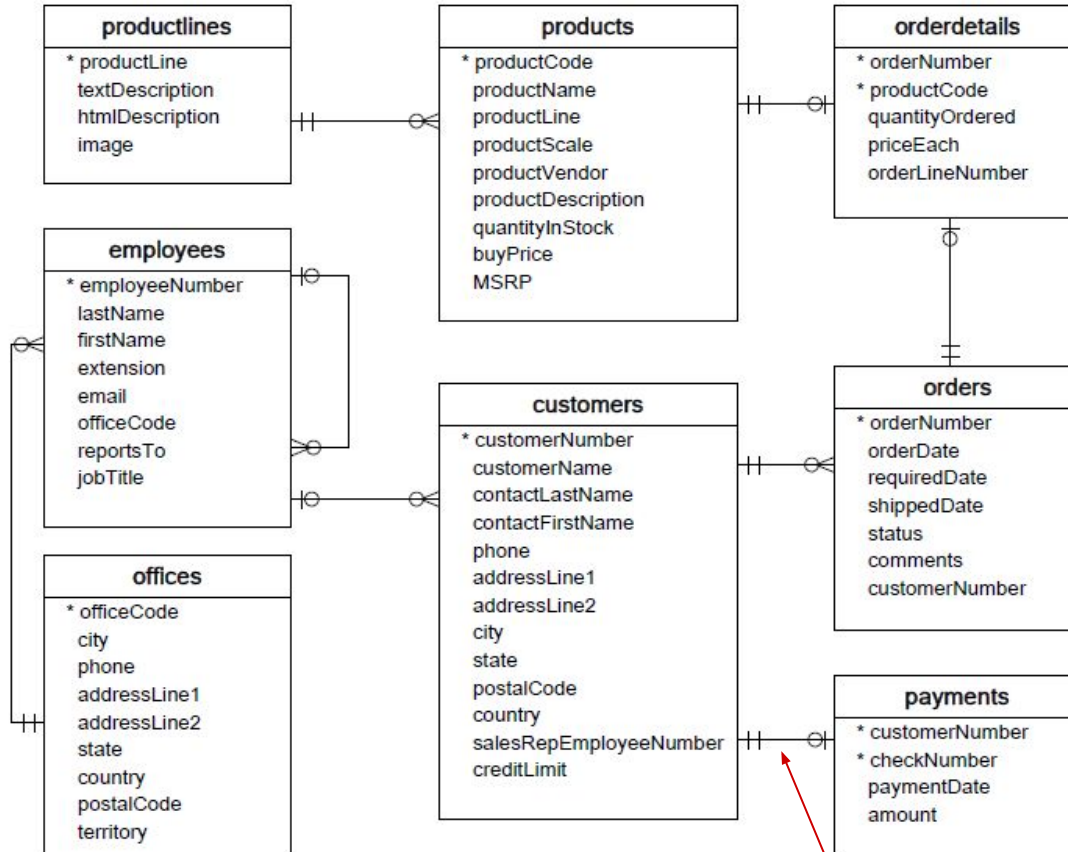
Tipe Relasi dalam Database

Terdapat 3 tipe relasi dalam database yaitu :

- One-to-one
- One-to-many
- Many-to-many



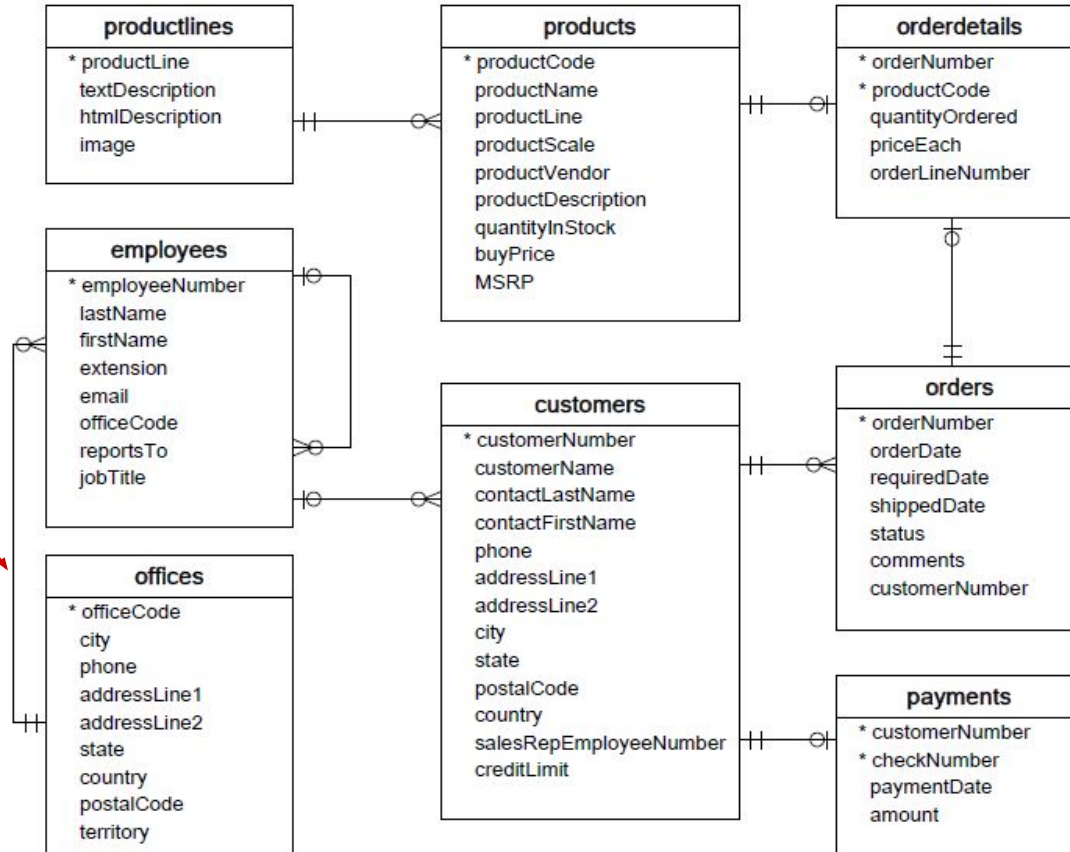
ERD Classicmodels



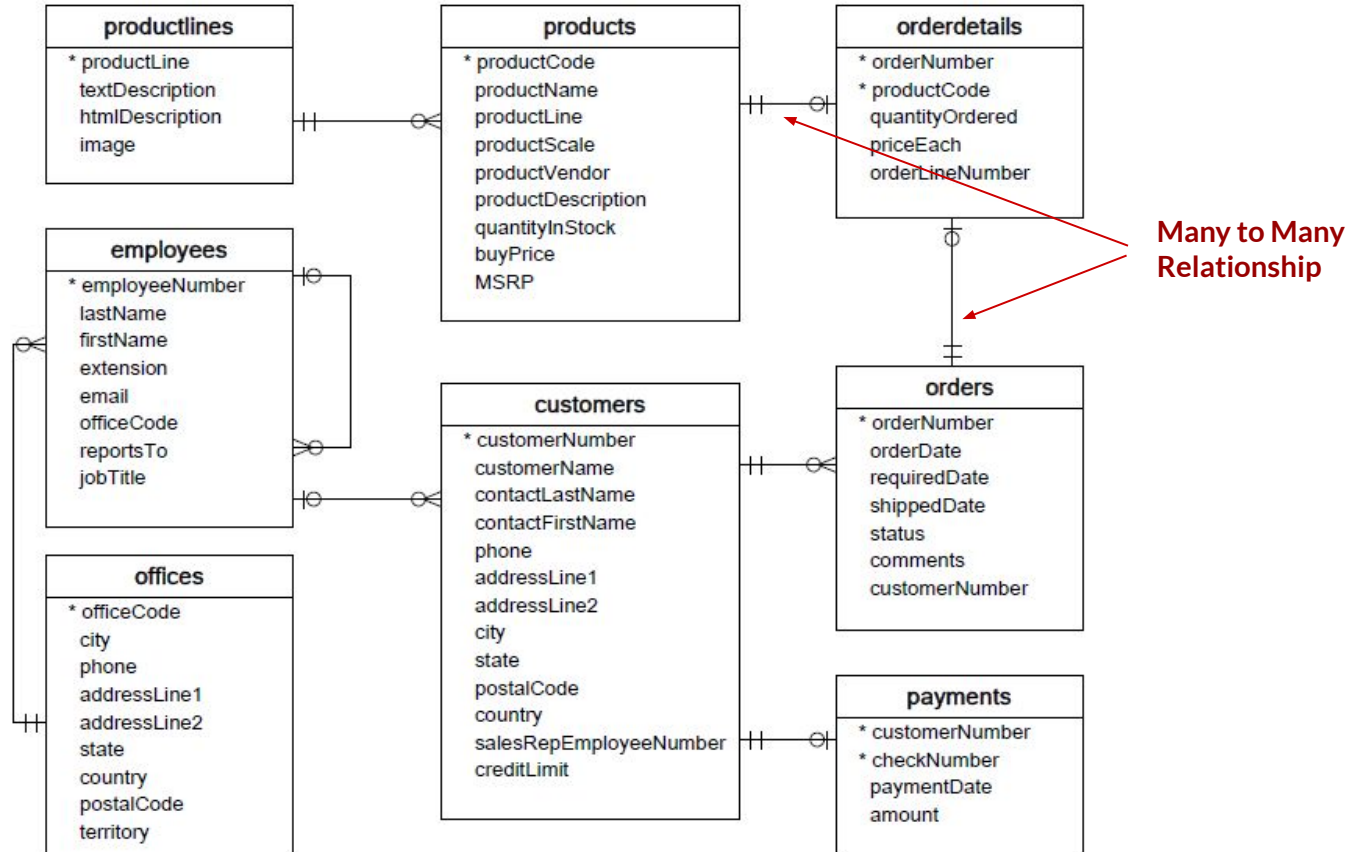
One to One Relationship

ERD Classicmodels

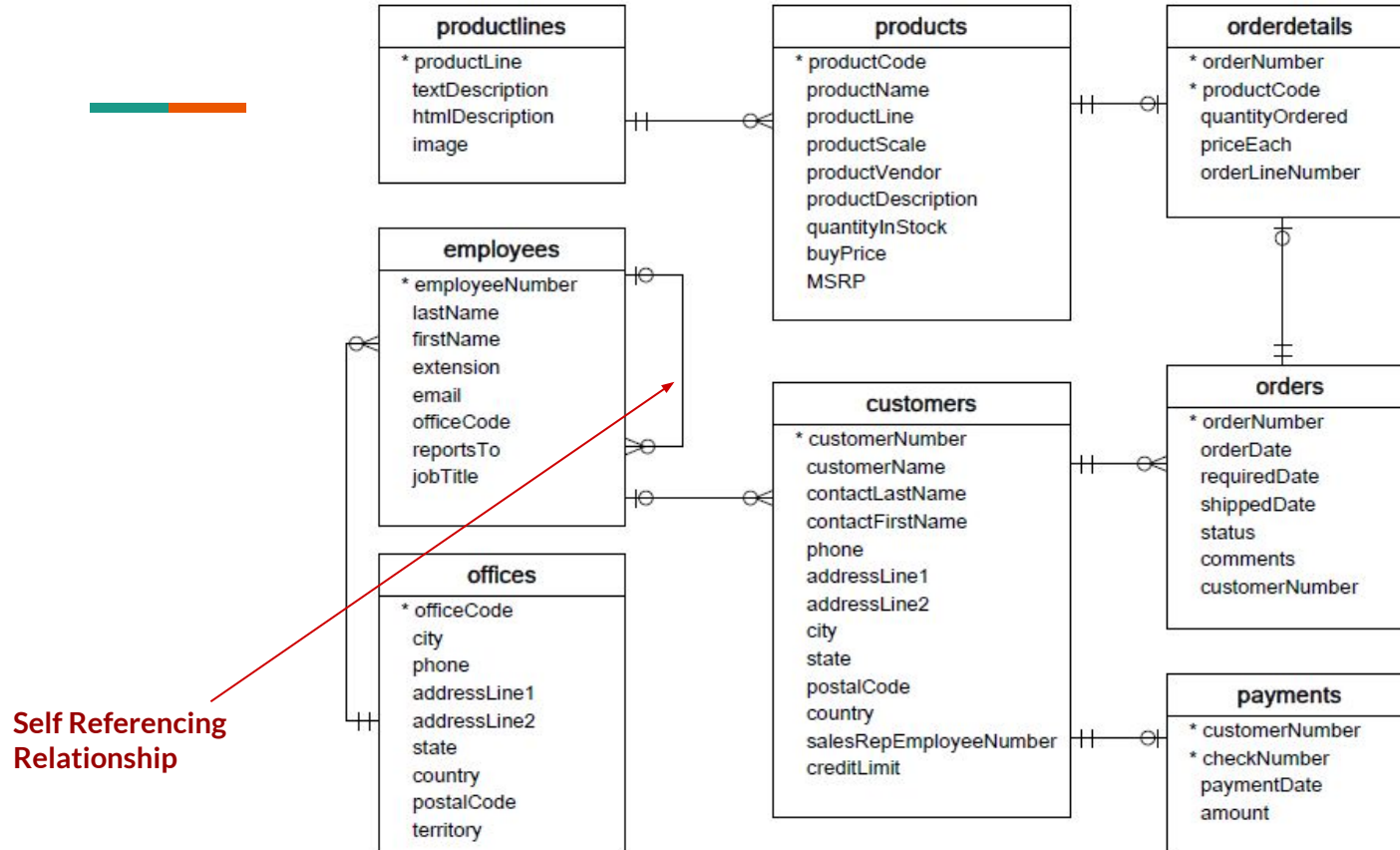
One to Many
Relationship



ERD Classicmodels



ERD Classicmodels





JOIN



JOIN

Dalam MySQL Join digunakan untuk mengakses atau menggabungkan dua tabel atau lebih berdasarkan kolom tertentu yang memiliki nilai atau key yang terikat satu sama lain.

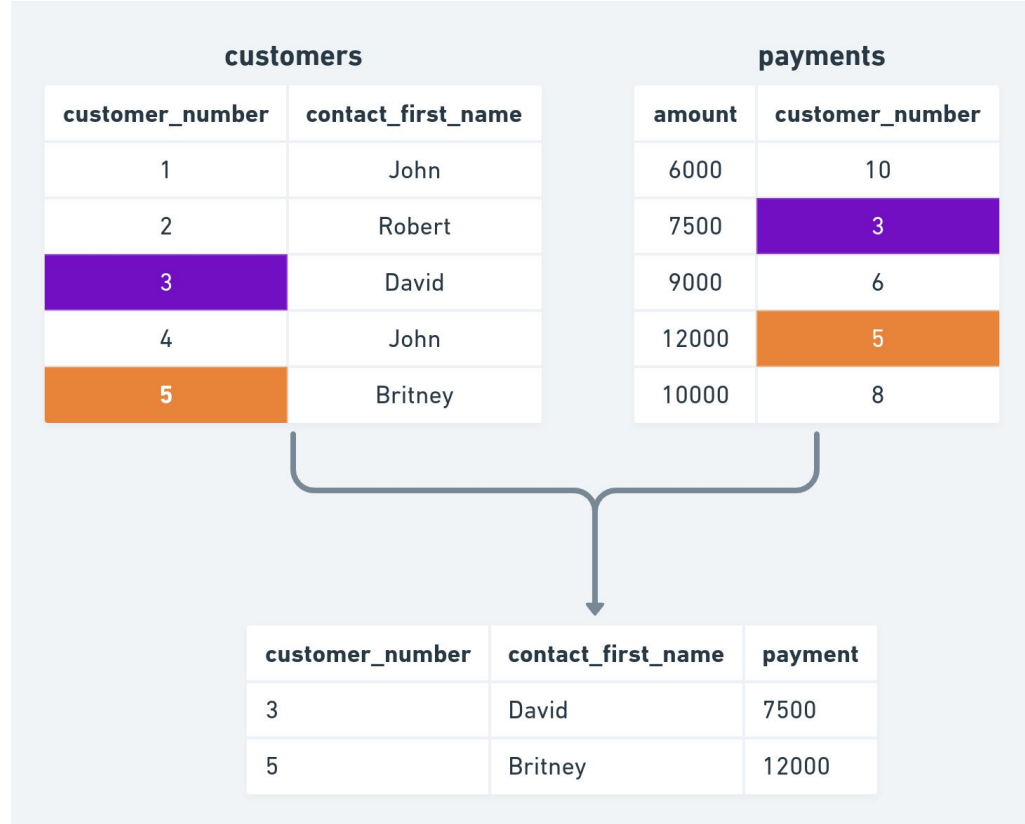
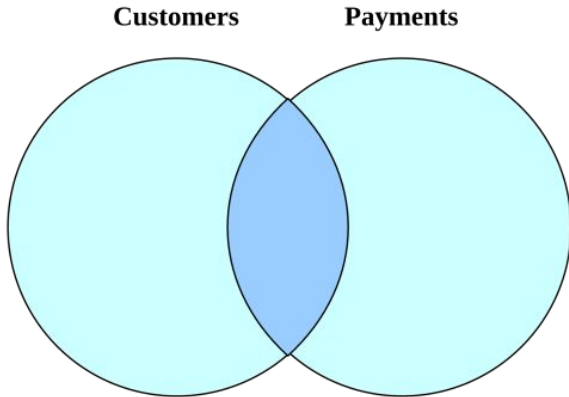
Secara umum terdapat dua tipe join dalam sql yaitu :

- INNER JOIN
- OUTER JOIN

INNER JOIN

Membandingkan record di setiap tabel untuk di cek apakah nilai sama atau tidak.

Jika nilai kedua tabel sama, maka akan terbentuk tabel baru yang hanya menampilkan record yang sama dari kedua tabel.



LATIHAN 1



1. Tampilkan semua data produk yang statusnya sedang dikirim!

LATIHAN 1



2. Tampilkan pelanggan yang tinggal di negara bagian yang sama dengan negara bagian tempat karyawan yang melayaninya bekerja!

LATIHAN 1




3. Tampilkan pelanggan yang pernah melakukan total pembayaran yang lebih besar dari 100.000



Quick Question 1

Apabila tidak terdapat foreign key yang menghubungkan dua tabel apakah tetap bisa melakukan perintah JOIN?

OUTER JOIN



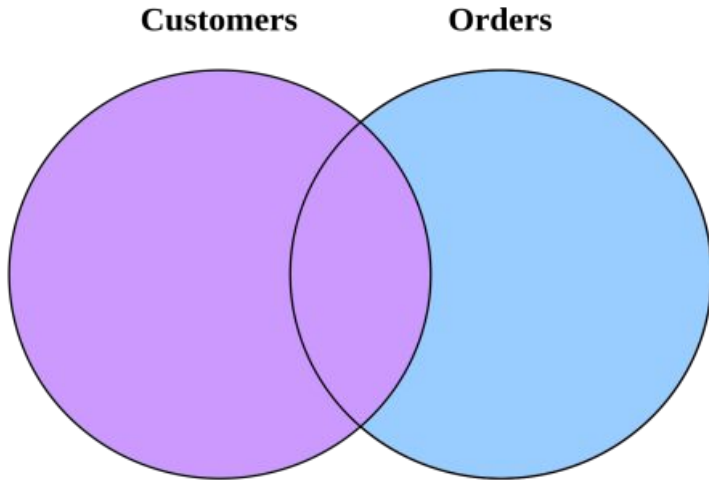
Pada outer join, tabel pertama yang ditentukan pada perintah SQL pada klausa FROM, ditandai sebagai tabel **KIRI** dan tabel selain itu ditandai sebagai tabel **KANAN**.

Outer join dibagi menjadi tiga tipe yaitu Left, Right, dan Full Outer Join.

LEFT JOIN




Pada perintah LEFT JOIN semua data pada tabel sebelah kiri akan ditampilkan, sedangkan data tabel sebelah kanan hanya akan ditampilkan jika data terkait pada tabel tersebut muncul di tabel sebelah kiri.



customers		orders	
customer_number	contact_first_name	order_number	customer_number
1	John	1023	10
2	Robert	1024	3
3	David	1025	6
4	John	1026	5
5	Britney	1027	8


customer_number	contact_first_name	order_number
1	John	null
2	Robert	null
3	David	1024
4	John	null
5	Britney	1026



Jika data tabelnya seperti berikut
bagaimana hasil left joinnya?

customer_number	contact_first_name
1	Lennon
2	Harrison

order_number	customer_number
1	1
2	2
3	2



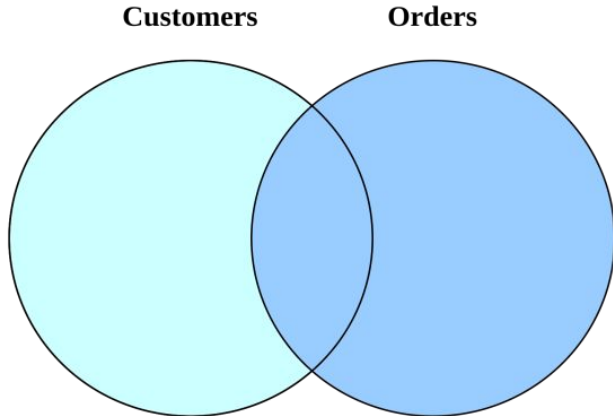
customer_number	name
1	Lennon
2	Harrison

order_number	customer_number
1	1
2	2
3	2

order_number	customer_number	name
1	1	Lenon
2	2	Harrison
3	2	Harrison

RIGHT JOIN

Berkebalikan dengan LEFT JOIN, pada perintah RIGHT JOIN semua data pada tabel sebelah kanan akan ditampilkan, sedangkan data tabel sebelah kiri hanya akan ditampilkan jika data yang terkait pada tabel tersebut muncul di tabel sebelah kanan.



customers		orders	
customer_number	contact_first_name	order_number	customer_number
1	John	1023	10
2	Robert	1024	3
3	David	1025	6
4	John	1026	5
5	Britney	1027	8

customer_number	contact_first_name	order_number
		1023
3	David	1024
		1025
5	Britney	1026
		1027



Latihan 2

Buat daftar semua pelanggan dan karyawan meskipun mereka tidak pernah terlibat kegiatan transaksi.



FULL OUTER JOIN

FULL OUTER JOIN atau FULL JOIN digunakan untuk menggabungkan dua tabel berdasarkan kolom tertentu. Perintah ini akan menampilkan data-data yang tidak berelasi. Namun pada tabel kanan dan tabel kiri data yang tidak berelasi akan bernilai NULL.

Namun MYSQL DBMS tidak mendukung sintaks FULL OUTER JOIN maka untuk memanipulasi hal ini dapat menggunakan LEFT JOIN dan RIGHT JOIN kemudian menggabungkannya menggunakan UNION

skip, akan dibahas di materi **set operation**



Equi Join

Melakukan join berdasarkan pencocokan nilai kolom dari tabel tabel terkait, tanda (=) digunakan sebagai operator perbandingan dalam klausa where untuk merujuk pada kesetaraan

```
SELECT col_name(s) FROM table_name1, table_name2  
WHERE table_name1.col_name = table_name2.col_name;
```



Non Equi Join dan Natural Join

Non Equi Join, menggabungkan tabel menggunakan operator perbandingan seperti \neq , $>$, $<$.

Natural Join, mirip equal join tapi kolom terkait hanya ditampilkan sekali.



SELF JOIN

Self join adalah join di mana sebuah tabel bergabung dengan dirinya sendiri, hal ini sering terjadi saat tabel tersebut memiliki foreign yang mereferensikan primary keynya sendiri.

```
SELECT alias1.col_name, alias2.col_name  
FROM tabel 'alias1' inner join tabel 'alias2'  
on alias1.key = alias2.key
```



ERROR KETIKA MENGHAPUS ATAU MENGUPDATE DATA KARENA MASALAH FOREIGN KEY



Solusi 1:

1. Menghapus tabel yang dijadikan key itu sebagai primary key
2. Kemudian baru menghapus baris yang memiliki primary key tabel tersebut

Masalahnya

1. Harus mengecek key satu-satu, **bagaimana jika data yang ingin dihapus banyak?**
2. **Bagaimana jika terdapat beberapa tabel yang menjadikan primary key dari tabel tersebut sebagai foreign key?**



Solusi 2:

Saat membuat tabel dengan foreign key dapat menambahkan

```
foreign key (reportsTo) references  
employees (employeeNumber)  
on delete cascade on update cascade
```

Sehingga saat data primary keynya dihapus maka otomatis semua data yang terkait primary key itu juga akan terhapus



Solusi 2:

Lalu bagaimana jika tabelnya sudah ada?

1. Drop Foreign key tabel lama

```
alter table nama_tabel drop  
constraint nama_constraint;
```

2. Buat foreign key baru dengan cascade

```
alter table nama_tabel add constraint nama_constraint  
foreign key (fk) references table_name(pk)  
on delete cascade on update cascade;
```

Bagaimana jika tidak mengetahui nama constraint?



Solusi 2:

Lalu bagaimana jika tabelnya sudah ada?

1. Drop Foreign key tabel lama

```
alter table nama_tabel drop  
constraint nama_constraint;
```

2. Buat foreign key baru dengan cascade

```
alter table nama_tabel add constraint nama_constraint  
foreign key (fk) references table_name(pk)  
on delete cascade on update cascade;
```



Solusi 2:

Saat membuat foreign key sebenarnya bisa menambahkan constraint name secara eksplisit

```
CONSTRAINT constraint_name FOREIGN KEY (fk)  
REFERENCES Persons(pk)
```

Namun jika terlanjur bisa menggunakan perintah berikut untuk mengecek nama defaultnya

```
select COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_COLUMN_NAME,  
REFERENCED TABLE NAME  
from information schema.KEY_COLUMN_USAGE  
where TABLE_NAME = 'nama_tabel';
```



Solusi 3:

1. Drop Foreign Key terkait pada child tabel :

```
ALTER TABLE table_name drop constraint constraint name;
```

2. Update atau delete table pada parent tabel,

3. Tambahkan kembali foreign key yang telah dihapus sebelumnya

```
ALTER TABLE table name ADD CONSTRAINT constraint name FOREIGN KEY  
(fk) REFERENCES productlines(pk);
```



Update dan Delete disertai join



Syntax Update tabel disertai JOIN

```
update table 1 inner join table_2 on  
table 1.key = table 2.key  
set table 1.col name = new value  
where [condition];
```




Syntax Delete tabel disertai JOIN

```
delete table_name from table_name1  
inner join table_name2 on  
table_name1.key=table_name2.key where  
[condition]
```



QUIZ



1 (Poin) (1.5 Menit)

Mengapa kita masih membutuhkan primary dan foreign key, ketika kita bisa menggabungkan tabel tanpa hal tersebut?

1 (Poin) (2 Menit)



1. Tabel manakah yang merupakan parent dari relasi diatas?
2. Kenapa saat ingin menghapus data pada tabel country muncul error berikut?

Cannot delete or update a parent row: a foreign key constraint fails (`db`.`capital`, CONSTRAINT `capital_ibfk_1` FOREIGN KEY (`country_id`) references `country` (`id`))

(3 Poin)
2.5 Menit

tb_buah

id	buah	id_warna
1	apel	1
2	anggur	2

tb_warna

id	warna
1	merah
2	ungu
3	kuning

1. Kira-kira apakah tipe relasi antara table tb_buah dengan tb_warna diatas?
2. Buatlah query untuk menampilkan hasil seperti dibawah!

buah	warna
apel	kuning
anggur	merah
<i>null</i>	ungu



SOAL PRAKTIKUM



TUGAS PRAKTIKUM

1. Tampilkan kolom tanggal pemesanan product dalam urutan menurun untuk pesanan Ford Pickup Truck 1940. (database classicmodels)
2. Tampilkan daftar nama produk yang dijual dengan harga kurang dari 80% dari MSRP (Harga Eceran Yang Disarankan). (database classicmodels)
3. Tampilkan Pembimbing Utama dari Mahasiswa bernama Sulaeman! (database appseminar)
4.
 - Buatlah kolom baru dengan nama status yang bertipe data varchar pada tabel customers
 - Kemudian isi dengan nilai 'VIP' untuk pelanggan yang jumlah pembayarannya pernah diatas 100.000 atau yang pernah mengorder barang dengan kuantitas 50 keatas
 - Sisanya isi dengan nilai 'Regular'.
5. Hapuslah semua data pelanggan yang pernah membatalkan pesanannya