

# SQL

## Complex Queries



# SQL Complex Queries

- ☐ Alias
- ☐ Subquery: FROM
- ☐ Subquery: WHERE
- ☐ JOIN Introduction
- ☐ JOIN Use Case
- ☐ LEFT JOIN vs INNER JOIN
- ☐ LEFT JOIN vs RIGHT JOIN
- ☐ Subquery & JOIN
- ☐ Other JOIN Use Cases
- ☐ UNION
- ☐ Common Table Expressions

# Hands-On Required :

## Database :

- [rakamin.tar / rakafood.zip](#) (raw CSVs)

Klik disini untuk mengakses  
folder Database

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# Alias - Definisi

Alias merupakan teknik untuk memberikan nama sementara pada tabel atau kolom.

## Perintah dalam SQL


```
SELECT [kolom_1] AS [a], [kolom_2]  
FROM <table_a> AS [b]
```



# Alias - Example

Contoh:

Bagaimana persebaran pelanggan aktif dan non-aktif untuk setiap kelompok umur.

 tutorial/postgres@local@postgres ▾

Query Editor   Query History

```

1  -- Alias
2  SELECT
3      CASE
4          WHEN umur BETWEEN 0 AND 12 THEN 'Anak-anak'
5          WHEN umur BETWEEN 13 AND 18 THEN 'Remaja'
6          WHEN umur BETWEEN 19 AND 50 THEN 'Dewasa'
7          ELSE 'Lansia'
8      END AS grup_umur,
9      SUM(CASE WHEN pengguna_aktif = 0 THEN 1 ELSE 0 END) AS total_pengguna_nonaktif,
10     SUM(CASE WHEN pengguna_aktif = 1 THEN 1 ELSE 0 END) AS total_pengguna_aktif
11 FROM rakamin_customer AS rc
12 GROUP BY 1;
13
14
    
```

Data Output   Explain   Messages   Notifications

	grup_umur text	total_pengguna_nonaktif bigint	total_pengguna_aktif bigint
1	Remaja	3	12
2	Dewasa	4	69

# Alias - Tips & Tricks

- Hindari penggunaan nama alias yang sama dengan **keyword** pada SQL.
- Untuk menghindari ambiguitas, hindari penggunaan nama alias kolom yang sama dengan **nama kolom lain**, **nama tabel** atau **nama subquery**. (Berlaku juga untuk alias nama tabel)
- Lebih baik menghindari penggunaan nama alias kolom yang **mengandung spasi**. Namun jika ingin memberikan nama alias kolom dengan spasi, dapat dilakukan dengan mengutip nama alias tersebut dengan double quote ("").

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions



# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# Subquery - Definisi


Kita dapat membuat sebuah query didalam sebuah query, yang dinamakan subquery. Biasanya digunakan untuk mendapatkan value dari tabel lain yang dibutuhkan untuk proses selanjutnya

## Perintah dalam SQL

```
SELECT [kolom_1], [kolom_2]  
FROM  
( SELECT [kolom_1], [kolom_2]  
  FROM <tabel_a>  
) ALIAS
```

# Subquery 1 - Tabel Virtual

Subquery bisa digunakan untuk membuat sebuah tabel 'virtual'. Setiap subquery harus memiliki **Alias**.

 tutorial/postgres@local@postgres ▼

Query Editor    Query History

```

1  SELECT nama, email
2  FROM (
3      SELECT nama, email, telepon, umur
4      FROM rakamin_customer
5      WHERE pengguna_aktif = 1
6  ) AS tmp
7  LIMIT 10;
    
```

Data Output    Explain    Messages    Notifications

	nama text	email text
1	Fakhrul Fandy Widyaningtias	fakhrulfandywidyaningtias@outlook.com
2	Ferdiansyah Amalina	ferdiansyahamalina@yahoo.com
3	Ficky Rizkylananta	fickyrizkylananta@hotmail.com
4	Singgih Kharisma	singgihkharisma@hotmail.com
5	Jeremiah Dinanti	jeremiahdinanti@rocketmail.com
6	Ridhwan Noerani	ridhwannoerani@hotmail.com
7	Okky Oktaviaman	okkyoktaviaman@rocketmail.com
8	Dikposa Yosua Krisnanto	dikposayosuakrisnanto@hotmail.com
9	Dhanu Rahman	dhanurahman@gmail.com
10	Bob Arisa	bobarisa@yahoo.com

# Subquery 1 - Tabel Virtual

Dari subquery tersebut, kita juga bisa melakukan filtering.

Contoh:

Bagaimana persebaran total customer pada masing-masing provider yang jumlah customernya lebih dari 20?

```

10 -- Subquery 2
11 SELECT *
12 FROM (
13     SELECT
14         CASE
15             WHEN telepon LIKE '62852%' THEN 'Telkomsel'
16             WHEN telepon LIKE '62878%' THEN 'XL'
17             WHEN telepon LIKE '62896%' THEN '3'
18             ELSE 'Lainnya'
19         END AS Provider,
20         COUNT(nama) AS Total_Pelanggan
21     FROM rakamin_customer
22     GROUP BY 1
23 ) AS rc
24 WHERE rc.total_pelanggan >= 20;

```

Data Output Explain Messages Notifications

	provider text	total_pelanggan bigint	
1	Lainnya	20	
2	XL	37	
3	Telkomsel	22	

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions



# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# Subquery 2 - Filtering

Hasil dari subquery digunakan sebagai filter query utama.

## Perintah dalam SQL

```
SELECT [kolom_1], [kolom_2]  
FROM <table_a>  
WHERE [kolom_1] IN  
( SELECT [kolom_1]  
  FROM <tabel_a>  
)
```

# Subquery 2 - Filtering

Hasil dari subquery digunakan sebagai filter query utama.

Contoh:

Ambil data customer yang pernah melakukan order di `KFC`

```

26  -- Subquery 3
27  SELECT
28      nama,
29      email,
30      telepon
31  FROM rakamin_customer AS rc
32  WHERE rc.id_pelanggan IN (
33      SELECT DISTINCT id_pelanggan
34      FROM rakamin_order AS ro
35      WHERE ro.id_merchant = 5
36  )
37  LIMIT 5;

```

	Data Output	Explain	Messages	Notifications
	nama text		email text	telepon text
1	Ficky Rizkyananta		fickyrizkyananta@hotmail.com	6285244922
2	Rifqy Butar-butur		rifqybutar-butur@yahoo.com	6289625971
3	Wahyu Hayati		wahyuhayati@gmail.com	6289627184
4	Yutama Riahdita		yutamariah dita@gmail.com	6289629610
5	Haikal Nurahmadhita		haikalnurahmadhita@rocketmail.com	6289630823

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions



# JOIN

Proses join menggunakan **kolom** dengan **nilai baris** yang **sama** antara 2 tabel.



# JOIN

Proses join menggunakan **kolom** dengan **nilai baris** yang **sama** antara 2 tabel.

## Perintah dalam SQL

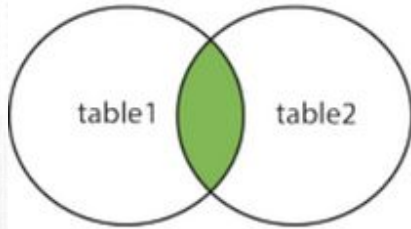
```
select [kolom_1], [kolom_2]
from <table_a> [tipe_join] <table_b>
on table_a.[kolom] = table_b.[kolom]
```

keyword **ON** mencocokkan value yang sama pada kondisi **JOIN**

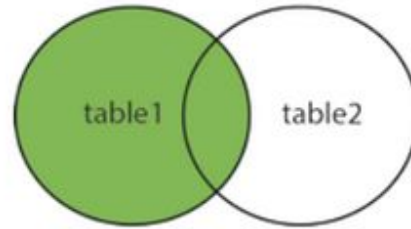
# Tipe-tipe Join Table

Terdapat berbagai macam tipe tergantung kebutuhan menggabungkan tabel

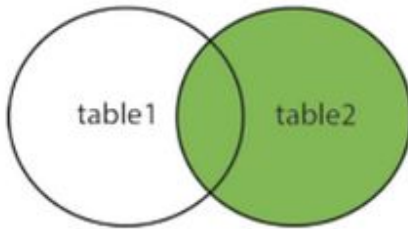
INNER JOIN



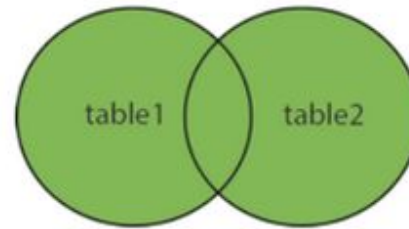
LEFT JOIN



RIGHT JOIN



FULL OUTER JOIN



# Tipe-tipe Join Table

Terdapat berbagai macam tipe tergantung kebutuhan menggabungkan tabel

Tipe	Definisi
INNER JOIN	Hanya mencocokkan dan menampilkan data yang sama antara 2 tabel
LEFT JOIN	Menampilkan semua data di tabel sebelah kiri, dan data yang cocok di tabel sebelah kanan
RIGHT JOIN	Menampilkan semua data di tabel sebelah kanan, dan data yang cocok di tabel sebelah kiri
FULL OUTER JOIN	Menampilkan semua data baik ketika ada yang cocok antara kedua tabel ataupun tidak

# Ilustrasi

**Tabel A**

Id	Usia
1	18
2	50
4	24
6	45

**Tabel B**

Id	Negara
1	ID
2	SG
3	JP
5	KR

**INNER JOIN**

Id	Usia	Negara
1	18	ID
2	50	SG

**FULL OUTER JOIN**

Id	Usia	Negara
1	18	ID
2	50	SG
4	24	NULL
6	45	NULL
3	NULL	JP
5	NULL	KR

**LEFT JOIN**

Id	Usia	Negara
1	18	ID
2	50	SG
4	24	NULL
6	45	NULL

**RIGHT JOIN**

Id	Usia	Negara
1	18	ID
2	50	SG
3	NULL	JP
5	NULL	KR



# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# Use Case



“

Guys, kita kasih service lebih yuk untuk pelanggan yang menggunakan provider X, kita bakal kirimin souvenir ke alamat mereka biar jadi lebih loyal ke kita.

Boleh tuh. berarti kita bakal dapetin user yang punya nomor hp dengan awalan `62852` ya

”

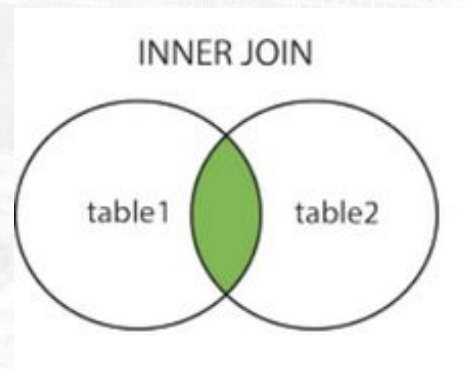
# INNER JOIN / JOIN

Contoh : Ambil alamat, nama dan nomor hp dengan awalan 62852 (untuk provider x)

```
53 -- Inner Join
54 SELECT
55     rc.nama,
56     rc.telepon,
57     rca.alamat
58 FROM rakamin_customer AS rc
59 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
60 WHERE rc.telepon LIKE '62852%'
61 LIMIT 5;
```

Data Output Explain Messages Notifications

	nama text	telepon text	alamat text
1	Fakhrul Fandy Widyaningtias	6285244454	Margonda
2	Ferdiansyah Amalina	6285244688	Bekasi Timur
3	Ficky Rizkyananta	6285244922	Margonda
4	Singgih Kharisma	6285245156	TMII
5	Jeremiah Dinanti	6285245390	Bekasi Barat



# Use Case



“

Dari kondisi yang sebelumnya, lebih bagus kalau kita tandai orang-orang yang pernah melakukan order di rakamin agar kita beri treatment yang lebih, gimana?

”



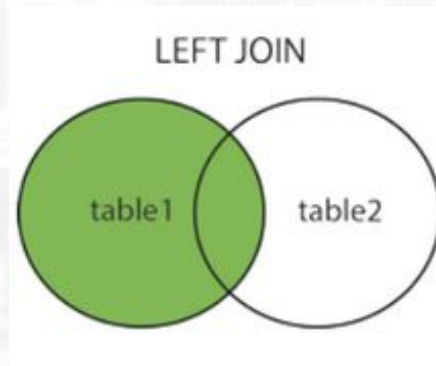
# LEFT JOIN

**Contoh : Ambil alamat, nama dan nomor hp dengan awalan 62852 (untuk provider x) yang melakukan order (di table rakamin\_orders)**

```
63 -- Left Join
64 SELECT
65     rc.nama,
66     rc.telepon,
67     rca.alamat
68 FROM rakamin_customer AS rc
69 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
70 LEFT JOIN rakamin_order AS ro ON ro.id_pelanggan = rc.id_pelanggan
71 WHERE rc.telepon LIKE '62852%'
72 LIMIT 5;
```

Data Output Explain Messages Notifications

	nama text	telepon text	alamat text	
1	Fakhrul Fandy Widyaningtias	6285244454	Margonda	
2	Ferdiansyah Amalina	6285244688	Bekasi Timur	
3	Ficky Rizkyananta	6285244922	Margonda	
4	Singgih Kharisma	6285245156	TMII	
5	Singgih Kharisma	6285245156	TMII	



LEFT JOIN disebut juga dengan LEFT OUTER JOIN

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



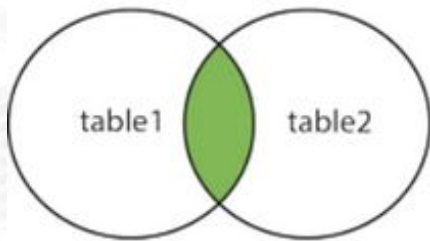
Common Table Expressions

# Balik lagi ke sebelumnya, Apa bedanya ya JOIN dan LEFT JOIN?

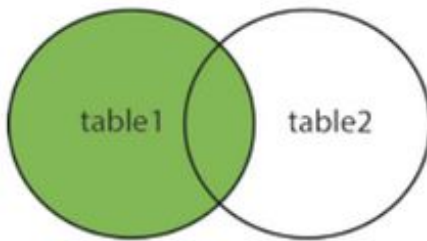
Untuk statement JOIN pada SQL, **hanya mencocokkan dan menampilkan data yang sama antara 2 tabel,**

sedangkan LEFT JOIN, mengambil semua data pada tabel sebelah kiri, dan **data yang cocok saja di sebelah kanan**

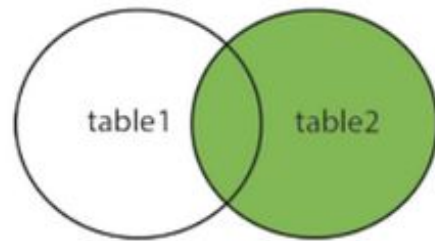
INNER JOIN



LEFT JOIN



RIGHT JOIN



```

63 -- Left Vs Inner Join
64 SELECT
65     rc.nama,
66     rc.telepon,
67     rca.alamat
68 FROM rakamin_customer AS rc
69 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
70 JOIN rakamin_order AS ro ON ro.id_pelanggan = rc.id_pelanggan
71 WHERE rc.telepon LIKE '62852%';

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 43 msec.  
27 rows affected.

```

63 -- Left Vs Inner Join
64 SELECT
65     rc.nama,
66     rc.telepon,
67     rca.alamat
68 FROM rakamin_customer AS rc
69 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
70 LEFT JOIN rakamin_order AS ro ON ro.id_pelanggan = rc.id_pelanggan
71 WHERE rc.telepon LIKE '62852%';

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 37 msec.  
33 rows affected.

# JOIN

# LEFT JOIN





# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

```

63 -- Left Vs Inner Join
64 SELECT
65     rc.nama,
66     rc.telepon,
67     rca.alamat
68 FROM rakamin_customer AS rc
69 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
70 LEFT JOIN rakamin_order AS ro ON ro.id_pelanggan = rc.id_pelanggan
71 WHERE rc.telepon LIKE '62852%';

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 37 msec.  
33 rows affected.

```

74 -- Left Vs Right Join
75 SELECT
76     rc.nama,
77     rc.telepon,
78     rca.alamat,
79     ro.tanggal_pembelian
80 FROM rakamin_customer AS rc
81 JOIN rakamin_customer_address AS rca ON rc.id_pelanggan = rca.id_pelanggan
82 RIGHT JOIN rakamin_order AS ro ON ro.id_pelanggan = rc.id_pelanggan
83 WHERE rc.telepon LIKE '62852%';

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 44 msec.  
27 rows affected.

# LEFT JOIN

# RIGHT JOIN



# Kenapa **RIGHT JOIN** kurang populer?

- RIGHT JOIN selalu bisa ditulis dalam bentuk LEFT JOIN.
- Kurang intuitif dalam melakukan RIGHT JOIN karena tabel utama (base table) ditulis paling terakhir.





# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions



# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases




UNION



Common Table Expressions

# Contoh - Menggunakan Subquery pada JOIN

Contoh:  
Ambil data customer yang pernah melakukan transaksi yang total harganya lebih dari 50000.

 tutorial/postgres@local@postgres

Query Editor
 Query History

```

85 -- Join With Subquery
86 SELECT *
87 FROM (
88     (SELECT
89         id_pelanggan,
90         SUM(harga) AS total_harga
91     FROM rakamin_order
92     GROUP BY 1
93     HAVING SUM(harga) > 50000
94     ) AS ro
95     LEFT JOIN rakamin_customer AS rc ON ro.id_pelanggan = rc.id_pelanggan
96 );
97
    
```

Data Output
 Explain
 Messages
 Notifications

	id_pelanggan bigint	total_harga numeric	index bigint	id_pelanggan bigint	nama text	email text
1	1	58642	0	1	Fakhrul Fandy Widyaningtias	fakhrulfandywi
2	3	58509	2	3	Ficky Rizkyananta	fickyrizkyanant
3	4	110349	3	4	Singgih Kharisma	singgihkharisr
4	8	103520	7	8	Dikposa Yosua Krisnanto	dikposayosuak
5	10	68779	9	10	Bob Arisa	bobarisa@yahc

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION

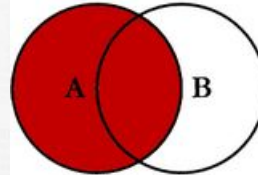


Common Table Expressions

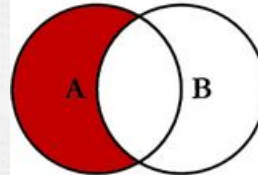
# Penggunaan join lainnya

Perintah *joins* bisa dimodifikasi untuk penggunaan yang lebih kompleks.

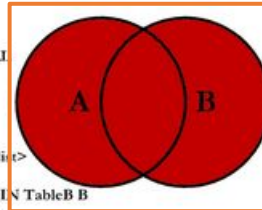
## LEFT JOIN ✓



```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```

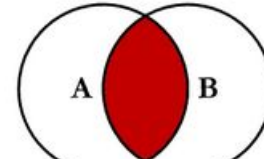


```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```

## OUTER JOIN ✓

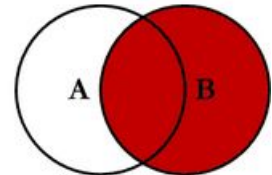
## SQL JOINS

### INNER JOIN ✓

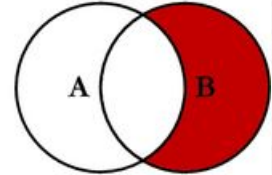


```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```

## RIGHT JOIN ✓



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```

```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```



# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases




UNION



Common Table Expressions

# SQL Complex Queries

- ☒ Alias
- ☒ Subquery: FROM
- ☒ Subquery: WHERE
- ☒ JOIN Introduction
- ☒ JOIN Use Case
- ☒ LEFT JOIN vs INNER JOIN
- ☒ LEFT JOIN vs RIGHT JOIN
- ☒ Subquery & JOIN
- ☒ Other JOIN Use Cases
-  ☐ UNION
- ☐ Common Table Expressions

# Union

Untuk menggabungkan dua atau lebih hasil query dari sebuah tabel, dapat digunakan UNION.

## Perintah dalam SQL

```
select [kolom_1], [kolom_2]  
from <table_a>  
[UNION]  
select [kolom_1], [kolom_2]  
from <table_b>
```

### Syarat:

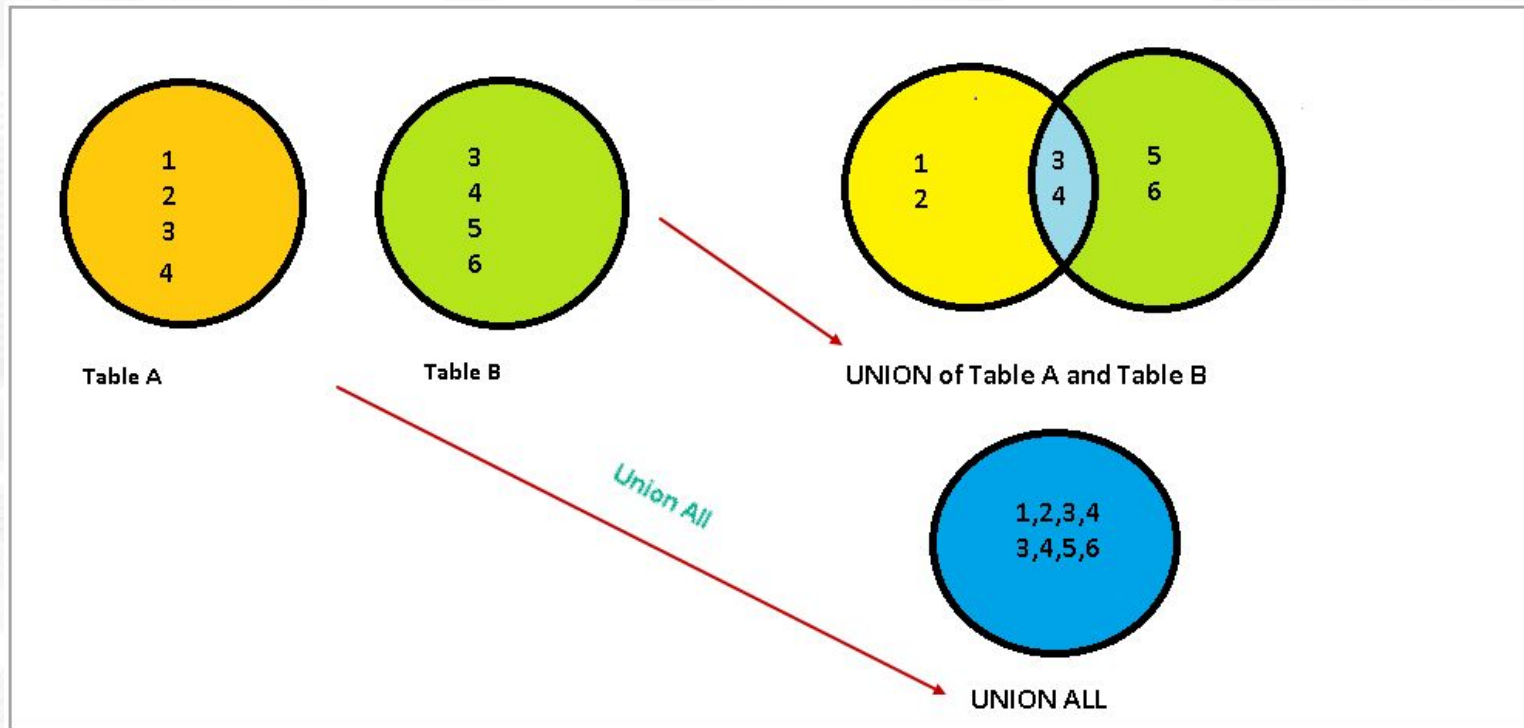
- Semua kolom yang digunakan antara tabel a dan tabel b harus sama
- Tipe data dari kolomnya juga harus sama
- Urutan dari kolom pun juga harus sama

# Jenis UNION

Ada beberapa jenis untuk menggabungkan sebuah data dengan UNION

Tipe	Definisi
UNION	Mengembalikan data secara <i>distinct</i>
UNION ALL	Mengembalikan data secara duplikat (jika ada)

# Ilustrasi





# Contoh - UNION & UNION ALL

Ambil data customer yang memiliki umur lebih dari 20. Ambil juga data customer yang lahir pada bulan Mei. (gunakan UNION & UNION ALL)

```

tutorial/postgres@local@postgres ▾
Query Editor  Query History

98  -- Union Vs Union All
99  SELECT
100     nama,
101     email,
102     bulan_lahir
103  FROM rakamin_customer AS rc
104  WHERE umur > 20
105  UNION
106  SELECT
107     nama,
108     email,
109     bulan_lahir
110  FROM rakamin_customer AS rc
111  WHERE bulan_lahir = 'Mei';
112

Data Output  Explain  Messages  Notifications
Successfully run. Total query runtime: 47 msec.
72 rows affected.
  
```

```

tutorial/postgres@local@postgres ▾
Query Editor  Query History

98  -- Union Vs Union All
99  SELECT
100     nama,
101     email,
102     bulan_lahir
103  FROM rakamin_customer AS rc
104  WHERE umur > 20
105  UNION ALL
106  SELECT
107     nama,
108     email,
109     bulan_lahir
110  FROM rakamin_customer AS rc
111  WHERE bulan_lahir = 'Mei';
112

Data Output  Explain  Messages  Notifications
Successfully run. Total query runtime: 39 msec.
83 rows affected.
  
```

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions

# WITH ... AS ()

Kita dapat juga membentuk sebuah temporary table dengan perintah SQL ini

## Perintah dalam SQL

```
WITH [nama_temporary_table] AS (  
    [put your query here...]  
)
```

```
SELECT *  
FROM [nama_temporary_table]
```

# Contoh - WITH ... AS

Dengan menggunakan contoh sebelumnya (Ambil data customer yang pernah melakukan transaksi yang total harganya lebih dari 50000)

tutorial/postgres@local@postgres ▾

Query Editor
Query History

```

113  -- CTE
114  WITH orders AS (
115      SELECT
116          id_pelanggan,
117          SUM(harga) AS total_harga
118      FROM rakamin_order
119      GROUP BY 1
120      HAVING SUM(harga) > 50000
121  )
122  SELECT *
123  FROM orders AS o
124  LEFT JOIN rakamin_customer AS rc ON rc.id_pelanggan = o.id_pelanggan;
125  
```

Data Output

Explain

Messages


Notifications

	id_pelanggan bigint	total_harga numeric	index bigint	id_pelanggan bigint	nama text	email text
1	1	58642	0	1	Fakhrul Fandy Widyaningtias	fakhrulfa
2	3	58509	2	3	Ficky Rizkyananta	fickyrizky
3	4	110349	3	4	Singgih Kharisma	singgihkh
4	8	103520	7	8	Dikposa Yosua Krisnanto	dikposay
5	10	68779	9	10	Bob Arisa	bobarisa(



# Contoh - WITH ... AS (2)

Variabel *with* bisa lebih dari satu.

 tutorial/postgres@local@postgres ▾

Query Editor
Query History

```

112
113 -- CTE
114 WITH orders AS (
115     SELECT
116         id_pelanggan,
117         SUM(harga) AS total_harga
118     FROM rakamin_order
119     GROUP BY 1
120     HAVING SUM(harga) > 50000
121 ),
122 customers AS (
123     SELECT *
124     FROM rakamin_customer
125     WHERE penipu = 0 AND pengguna_aktif = 1
126 )
127 SELECT * FROM orders AS o
128 JOIN customers AS c ON o.id_pelanggan = c.id_pelanggan;
129

```

Data Output
Explain
Messages
Notifications

	id_pelanggan bigint	total_harga numeric	index bigint	id_pelanggan bigint	nama text
1	68	79901	67	68	Reny Oktavianty
2	51	66413	50	51	Athirah Atsila
3	70	68504	69	70	Fathina Narimanda
4	69	57831	68	69	Salsabila Resa
5	60	50636	59	60	Hanifa Ivangkia

# SQL Complex Queries



Alias



Subquery: FROM



Subquery: WHERE



JOIN Introduction



JOIN Use Case



LEFT JOIN vs INNER JOIN



LEFT JOIN vs RIGHT JOIN



Subquery & JOIN



Other JOIN Use Cases



UNION



Common Table Expressions