

# SQL for Data Analysis

Rizki Teguh Kurniawan

Data Scientist, eFishery

Youtube: Solving with Akoyo

# Agenda



- *Mengenal fungsi agregat: COUNT, MIN, MAX, SUM, AVG*
- *Mengelompokkan data dengan GROUP BY*
- *Memfilter data dengan HAVING dan perbedaannya dengan WHERE*
- *SQL Join*



# Fungsi Agregat

Fungsi agregat melakukan operasi pada sekumpulan nilai dan mengembalikan satu nilai  
Fungsi-fungsi agregat di SQL antara lain: COUNT, SUM, MIN, MAX, AVG

## SELECT

```
COUNT(*) AS "Banyak baris",  
COUNT DISTINCT "Order ID" AS "Banyak order",  
MIN("Order Date") AS "Tanggal awal",  
MAX("Order Date") AS "Tanggal akhir",  
SUM("Profit") AS "Total profit",  
AVG("Sales") AS "Rerata sales"
```

## FROM

```
transactions
```

	Banyak baris bigint	Tanggal awal order date	Tanggal akhir order date	Total profit double precision	Rata rata sales double precision
1	9994	2014-01-03	2017-12-30	286397.02170000126	229.85800083049386

- **COUNT**  
Menghitung banyaknya data.  
Menggabungkan **COUNT** dengan **DISTINCT** digunakan untuk menghitung banyaknya data yang berbeda
- **MIN**  
Menghitung nilai minimum
- **MAX**  
Menghitung nilai maksimum
- **SUM**  
Menghitung jumlah
- **AVG**  
Menghitung rata-rata



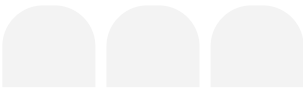
# Mengelompokkan data menggunakan **GROUP BY**

Fungsi-fungsi agregat biasanya digabungkan dengan **GROUP BY** untuk mengelompokkan data berdasarkan satu atau lebih kolom

```
// GROUP BY menggunakan kolom categorical

SELECT
  "Segment",
  COUNT(DISTINCT "Order ID") AS "Banyak order",
FROM
  transactions
GROUP BY
  "Segment"
```

	Segment character varying	Banyak Order bigint
1	Consumer	2586
2	Corporate	1514
3	Home Office	909





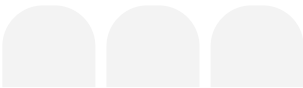
## (Cont...) Mengelompokkan data menggunakan **GROUP BY**

Fungsi-fungsi agregat biasanya digabungkan dengan **GROUP BY** untuk mengelompokkan data berdasarkan satu atau lebih kolom

```
// GROUP BY menggunakan kolom numerical

SELECT
    "Order Date",
    COUNT(DISTINCT "Order ID") AS "Banyak order",
FROM
    transactions
GROUP BY
    "Order Date"
```

	Order Date date	Banyak Order bigint
43	2014-03-01	5
44	2014-03-02	2
45	2014-03-03	5
46	2014-03-04	2
47	2014-03-05	1
48	2014-03-07	3



## Menyaring data menggunakan **HAVING**

- **HAVING** dan **WHERE** memiliki fungsi yang sama
- Perbedaan di antara keduanya adalah **WHERE** tidak bisa digabungkan dengan fungsi agregat sedangkan **HAVING** bisa.
- **WHERE** dan **HAVING** bisa ada dalam satu query dengan **WHERE** diposisikan lebih awal daripada **HAVING**

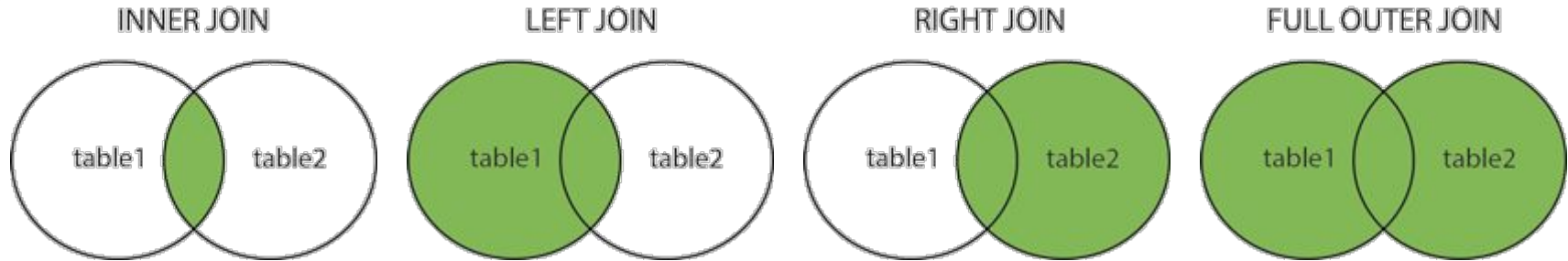
```
SELECT
    order_id AS "Order ID",
    SUM(sales) AS "Nilai order"
FROM
    transactions
WHERE
    segment = 'Consumer'
GROUP BY
    order_id
HAVING
    SUM(sales) > 100
```

	Order ID character varying	Nilai Order double precision
1	CA-2016-146143	133.38
2	CA-2015-131856	769.024
3	CA-2017-142909	182.55
4	CA-2016-153598	1761.8500000000001
5	CA-2017-162481	278.23
6	US-2017-169320	171.43

# SQL Join

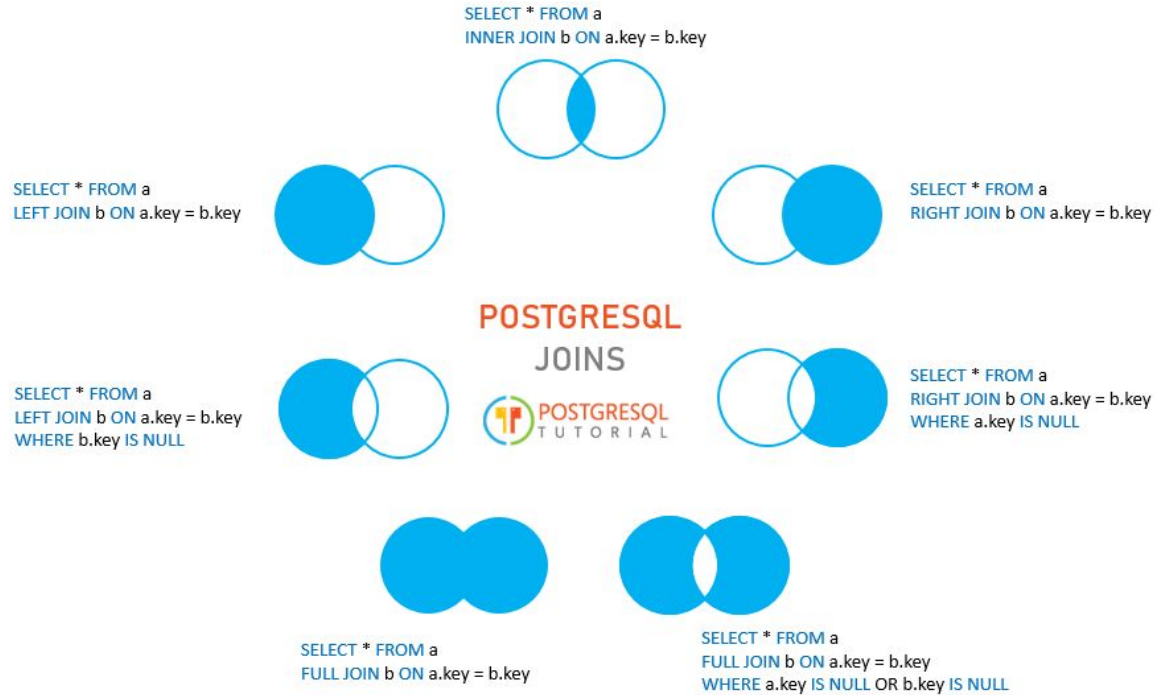
Misalkan kita memiliki dua tabel A dan B

- **INNER JOIN** : Returns records that have matching values in both tables
- **LEFT JOIN** : Returns all records from the left table, and the matched records from the right table
- **RIGHT JOIN** : Returns all records from the right table, and the matched records from the left table
- **FULL JOIN** : Returns all records when there is a match in either left or right table





# SQL Join







# Porto

Report -> slides -> ngangkat suatu isu

Dashboard