Intermediate SQL

DigitalSkola
Uncover The World Of Digital Skills With Us

Today's Quote

Googling to learn and get the correct answer (or syntax) is one of the most underrated skills

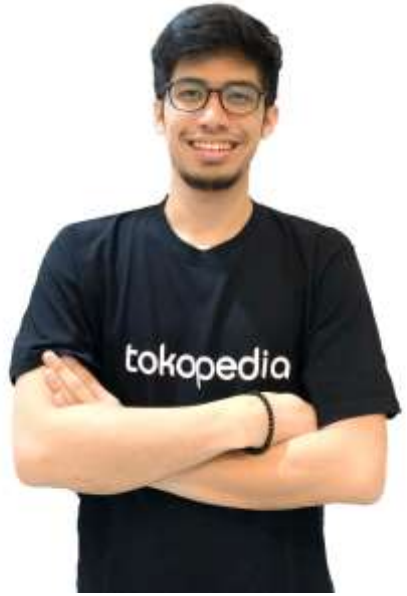
- anonymous

Profile

Mathematics

Data Scientist –
Stream Intelligence

Data Analyst –
Tokopedia



Farhan Reza Gumay

Review

Yesterday, we learn how to :

- Create
- Insert
- Update
- Delete
- Select

Now, imagine we already have table, what can we do to manipulate the table?

DigitalSkola
Uncover The World Of Digital Skills With Us

Table of Content

What will We Learn Today?

1. Limit
2. Distinct
3. Where
4. Order by
5. Aggregate Functions
6. Group by
7. Other SQL most used functions

Notes : The syntax might be slightly different between PostgreSQL, MySQL, BigQuery, etc

DigitalSkola
Uncover The World Of Digital Skills With Us

Limit

What if you just want to show like 5 top rows?

	employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary
1	100	Steven	King	SKING	515.123.4567	1987-06-17	AD_PRES	24,000
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	1987-06-18	AD_VP	17,000
3	102	Lex	De Haan	LDEHAAN	515.123.4569	1987-06-19	AD_VP	17,000
4	103	Alexander	Hunold	AHUNOLD	590.423.4567	1987-06-20	IT_PROG	9,000
5	104	Bruce	Ernst	BERNST	590.423.4568	1987-06-21	IT_PROG	6,000
6	105	David	Austin	DAUSTIN	590.423.4569	1987-06-22	IT_PROG	4,800
7	106	Valli	Pataballa	VPATABAL	590.423.4560	1987-06-23	IT_PROG	4,800
8	107	Diana	Lorentz	DLORENTZ	590.423.5567	1987-06-24	IT_PROG	4,200
9	108	Nancy	Greenberg	NGREENBE	515.124.4569	1987-06-25	FI_MGR	12,000
10	109	Daniel	Faviet	DFAVIET	515.124.4169	1987-06-26	FI_ACCOUNT	9,000

```
select *
from table
limit 5
```

Distinct

How can we filter the unique **location** possible?

transaction_detail

seller_id	buyer_id	location
123	756	Jakarta
124	768	Jakarta
125	798	Bandung

```
select distinct
    location
from transaction_detail
```

location
Jakarta
Bandung

Where

What if our business partners just want to analyze transactions that happen in Jakarta?

transaction_detail

seller_id	buyer_id	location
123	756	Jakarta
124	768	Jakarta
125	798	Bandung

```
select *
from transaction_detail
where location = 'Jakarta'
```

seller_id	buyer_id	location
123	756	Jakarta
124	768	Jakarta

Order by

What if we want to sort from the smallest to largest transaction

transaction_detail

seller_id	buyer_id	location	total_transaction
123	756	Jakarta	125,000
124	768	Jakarta	300,000
125	798	Bandung	200,000

```
select *
from transaction_detail
order by total_transaction
```

seller_id	buyer_id	location	total_transaction
123	756	Jakarta	125,000
125	798	Bandung	200,000
124	768	Jakarta	300,000

Aggregate Functions

Some of the most-used function in SQL :

- Sum
- Average
- Median / Percentile
- Count
- Count Distinct
- Lead & Lag

Remember : don't forget to googling anything that you want to find out.
For example, if you want to calculate moving average in SQL, just type in google "how to calculate moving average in PostgreSQL

Aggregate Functions

In general, the syntax will be like this :

function(expression)

For Median & Percentile, the function will be like this :

percentile_cont(fraction) within group (order by sort_expression)

If you in the future you work with BigQuery on Google Cloud Platform, there is another functions to calculate median (not as precise as percentile_cont() functions, but much faster and will be very useful in very large dataset

Aggregate Functions

Example :
How if we want to know what is the total transaction that happens?

transaction_detail

seller_id	buyer_id	location	total_transaction
123	756	Jakarta	125,000
124	768	Jakarta	300,000
125	798	Bandung	200,000

```
select sum(total_transaction) as total_all_transaction
from transaction_detail
```

total_all_transaction	625,000

Aggregate Functions

Example :
How if we want to know what is the average per transaction?

transaction_detail

seller_id	buyer_id	location	total_transaction
123	756	Jakarta	125,000
124	768	Jakarta	300,000
125	798	Bandung	200,000

```
select avg(total_transaction) as avg_per_trx
from transaction_detail
```

avg_per_trx	208,333

Group by

Then…
Comes the question, how if we want to know the total transaction but per location?

Group by

Don’t worry, we have group by function to help!

transaction_detail

seller_id	buyer_id	location	total_transaction
123	756	Jakarta	125,000
124	768	Jakarta	300,000
125	798	Bandung	200,000

```
select location,
sum(total_transaction) as total_trx_per_location
from transaction_detail
group by location
```

location	total_transaction
Jakarta	425,000
Bandung	200,000

Other Most Used Function in SQL

Consider below example, how if we want to know the difference of sales with next year sales?
How is the difference between sales in 2018 & 2019?

	year smallint	sum numeric
1	2018	5021.00
2	2019	4944.00
3	2020	5137.00

Lead() Function

First, we need to transform the initial table into table like this first

	year smallint	sum numeric
1	2018	5021.00
2	2019	4944.00
3	2020	5137.00

Step 1

	year smallint	amount numeric	next_year_sales numeric
1	2018	5021.00	4944.00
2	2019	4944.00	5137.00
3	2020	5137.00	[null]

Step 2

	year smallint	amount numeric	next_year_sales numeric	variance numeric
1	2018	5021.00	4944.00	-77.00
2	2019	4944.00	5137.00	193.00
3	2020	5137.00	[null]	[null]

Step 3

Lead() Function

lead(expression, offset) over(partition by partition_expression order by order_expression)

Here comes functions lead() to help!

Step 2

```
select
    year,
    amount,
    lead(amount, 1) over(partition by year order by year) as next_year_sales
from table
```

Step 3

```
select
    year,
    amount,
    next_year_sales,
    (amount – next_year_sales) as diff_sales
from prev_table
```

Lead() Function

	year smallint	amount numeric (10,2)	group_id integer	next_year_sales numeric
1	2018	1474.00	1	1915.00
2	2019	1915.00	1	1646.00
3	2020	1646.00	1	[null]
4	2018	1787.00	2	1911.00
5	2019	1911.00	2	1975.00
6	2020	1975.00	2	[null]
7	2018	1760.00	3	1118.00
8	2019	1118.00	3	1516.00
9	2020	1516.00	3	[null]

```
select
   year,
   amount,
   lead(amount, 1) over(partition by group_id order by year asc) as next_year_sales
from table
```

Lag() Function in SQL

There is an opposite function of **lead(),** called **lag()**

lag(expression, offset) over(partition by partition_expression order by order_expression)

	year smallint	sum numeric
1	2018	5021.00
2	2019	4944.00
3	2020	5137.00

Initial Table

	year smallint	amount numeric	previous_year_sales numeric
1	2018	5021.00	[null]
2	2019	4944.00	5021.00
3	2020	5137.00	4944.00

Transformed Table

Case When Function

Similar to If else function in general programming

```
case
   when condition_1 then result_1
   when condition_2 then result_2
   [when …]
  else other_result
end
```

DigitalSkola
Uncover The World Of Digital Skills With Us

Case When Function

	title character varying (255)	length smallint
1	Academy Dinosaur	86
2	Ace Goldfinger	48
3	Adaptation Holes	50
4	Affair Prejudice	117
5	African Egg	130
6	Agent Truman	169
7	Airplane Sierra	62
8	Airport Pollock	54
9	Alabama Devil	114
10	Aladdin Calendar	63
11	Alamo Videotape	126
12	Alaska Phantom	136
13	Ali Forever	150
14	Alice Fantasia	94

```
select
    title,
    length,
    case
        when length > 0 and length <= 50 then 'Short'
        when length > 50 and length <= 120 then 'Medium'
        when length > 120 then 'Long'
    end as duration
from film
order by title
```

Case When Function

Output :

	title character varying (255)	length smallint	duration text
1	Academy Dinosaur	86	Medium
2	Ace Goldfinger	48	Short
3	Adaptation Holes	50	Short
4	Affair Prejudice	117	Medium
5	African Egg	130	Long
6	Agent Truman	169	Long
7	Airplane Sierra	62	Medium
8	Airport Pollock	54	Medium
9	Alabama Devil	114	Medium
10	Aladdin Calendar	63	Medium
11	Alamo Videotape	126	Long
12	Alaska Phantom	136	Long
13	Ali Forever	150	Long
14	Alice Fantasia	94	Medium

Other Most Used Function

- *and* : To get the value that fulfill both condition
- *or* : To get the value that fulfill one of the condition
- *between* : Include both lower & upper threshold

DigitalSkola
Uncover The World Of Digital Skills With Us

Let's Practice!

DigitalSkola
Uncover The World Of Digital Skills With Us

1. Get the total sales

DigitalSkola
Uncover The World Of Digital Skills With Us

2. Get total profit per category

DigitalSkola
Uncover The World Of Digital Skills With Us

3. What are the unique categories and sub categories in the table?

DigitalSkola
Uncover The World Of Digital Skills With Us

4. How many order are there?

5. Sort category from the highest profit to the lowest along with their total profit

DigitalSkola
Uncover The World Of Digital Skills With Us

6. What is the average and median profit per order?

DigitalSkola
Uncover The World Of Digital Skills With Us

7. What are the top 5 sub category with most orders?

DigitalSkola
Uncover The World Of Digital Skills With Us

8. What is the total quantity sold for Category “Furniture” ?

DigitalSkola
Uncover The World Of Digital Skills With Us

9. Assume total profit for category Furniture is in US Dollar (1 USD = IDR 14k), but for category Office Supplies is in Pound Sterling (1 Pound Sterling = IDR 19k).
What is the total profit category furniture & office supplies in Rupiah?

DigitalSkola
Uncover The World Of Digital Skills With Us

10. What is the average and median profit per 1 product in 1 order?

Review on what we learn today

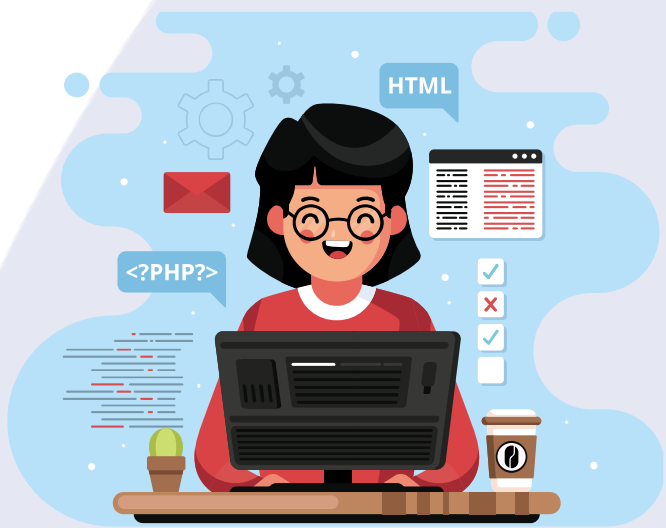
Today we learn about many function that often used in SQL

1. Limit
2. Distinct
3. Where
4. Order by
5. Aggregate Function
6. Group by
7. Lead, Lag
8. Case when

References

https://www.postgresqltutorial.com/

https://cloud.google.com/bigquery/docs/



DigitalSkola

Uncover The World Of Digital Skills With Us

Thank YOU