

DΦLab BOOTCAMP
DATA ANALYST
BATCH 4

Recap

Basic SQL for Data Analysis



PS461 - Fitri Dwi Hastuti



Daftar Isi:

Pengenalan DQLabMartBasic	3
Select Statement	5
Prefix dan Alias	6
Beberapa Operator pada MySQL	8
Fungsi	10
Filtering dengan Where	13
Operator Perbandingan Like	14
Operator Logika	15
Order By	16
Fungsi Agregasi	17
Group By dan Having	18
Join	19
Union	23
Limit	24

Pengenalan DQLabMartBasic

- Aplikasi database manajemen yang digunakan yaitu DBeaver.
- **DQLabMartBasic** adalah database dummy dari DQLab, yang terdiri dari data-data transaksi sebuah mini market fiktif bernama DQLabMart.
- Isi dari database ini bukan riil, namun dirancang khusus dengan ukuran data cukup kecil sehingga mudah dipahami namun berkualitas baik untuk digunakan sebagai bahan mempelajari konsep database dan SQL dasar.
- DQLabMartBasic terdiri dari tiga tabel sebagai berikut:

1

ms_pelanggan_dqlab

	kode_pelanggan	no_urut	nama_pelanggan	alamat
1	dqlabcust00	0	Pelanggan Non Member	[NULL]
2	dqlabcust01	1	Eva Novianti, S.H.	Vila Sempilan, No. 67 - Kota B
3	dqlabcust02	2	Heidi Goh	Vila Sempilan, No. 11 - Kota B
4	dqlabcust03	3	Unang Handoko	Vila Sempilan, No. 1 - Kota B
5	dqlabcust04	4	Jokolono Sukarman	Vila Permata Intan Berkilau, Blok C5-7
6	dqlabcust05	5	Tommy Sinaga	Vila Permata Intan Berkilau, Blok A1/2
7	dqlabcust06	6	Irwan Setianto	Vila Gunung Seribu, Blok O1 - No. 1 - Kota C
8	dqlabcust07	7	Agus Cahyono	Vila Gunung Seribu, Blok F4 - No. 8
9	dqlabcust08	8	Maria Sirait	Vila Bukit Sagitarius, Gang. Sawit No. 3
10	dqlabcust09	9	Ir. Ita Nugraha	Vila Bukit Sagitarius, Gang Kelapa No. 6
11	dqlabcust10	10	Djoko Wardoyo, Drs.	Vila Bukit Sagitarius, Blok A1 No. 1

2

ms_produk_dqlab

	kode_produk	no_urut	kategori_produk	nama_produk	harga
1	prod-01	1	Alat Tulis Kantor	Kotak Pensil DQLab	62,500
2	prod-02	2	Aksesoris Komputer	Flashdisk DQLab 64 GB	55,000
3	prod-03	3	Gift & Voucher	Gift Voucher DQLab 100rb	100,000
4	prod-04	4	Aksesoris Komputer	Flashdisk DQLab 32 GB	40,000
5	prod-05	5	Gift & Voucher	Gift Voucher DQLab 250rb	250,000
6	prod-06	6	Alat Tulis Kantor	Pulpen Multifunction + Laser DQLab	92,500
7	prod-07	7	Gift & Voucher	Tas Travel Organizer DigiSkills.id	48,000
8	prod-08	8	Gift & Voucher	Gantungan Kunci DQLab	15,800
9	prod-09	9	Alat Tulis Kantor	Buku Planner Agenda DQLab	92,000
10	prod-10	10	Alat Tulis Kantor	Sticky Notes DQLab 500 sheets	55,000

3

tr_penjualan_dqlab

- kode_urut_transaksi
- kode_transaksi
- kode_pelanggan
- tgl_transaksi
- no_urut
- kode_produk
- nama_produk
- qty
- harga
- diskon_persen

	ABC	ABC kode_	ABC kode_pe	🕒 tgl_transaksi	123 no_urut	ABC kode_produk	ABC nama_produk	123 qty	123 harga	123 diskon
1	1	tr-001	dqlabcust07	2020-05-01 00:00:00	1	prod-01	Kotak Pensil DQLab	5	62,500	0
2	2	tr-001	dqlabcust07	2020-05-01 00:00:00	2	prod-03	Flash disk DQLab 32 GB	1	100,000	25
3	3	tr-001	dqlabcust07	2020-05-01 00:00:00	3	prod-09	Buku Planner Agenda D	3	92,000	0
4	4	tr-001	dqlabcust07	2020-05-01 00:00:00	4	prod-04	Flashdisk DQLab 32 GB	3	40,000	0
5	5	tr-002	dqlabcust00	2020-05-01 00:00:00	1	prod-03	Gift Voucher DQLab 100	2	100,000	0
6	6	tr-002	dqlabcust00	2020-05-01 00:00:00	2	prod-10	Sticky Notes DQLab 500	4	55,000	0
7	7	tr-002	dqlabcust00	2020-05-01 00:00:00	3	prod-07	Tas Travel Organizer DQ	1	48,000	0
8	8	tr-003	dqlabcust03	2020-05-03 00:00:00	1	prod-02	Flashdisk DQLab 64 GB	2	55,000	12.5
9	9	tr-004	dqlabcust03	2020-05-03 00:00:00	1	prod-10	Sticky Notes DQLab 500	5	55,000	0
10	10	tr-004	dqlabcust03	2020-05-03 00:00:00	2	prod-04	Flashdisk DQLab 32 GB	4	40,000	0
11	11	tr-005	dqlabcust00	2020-06-12 00:00:00	1	prod-09	Buku Planner Agenda D	3	92,000	0
12	12	tr-005	dqlabcust00	2020-06-12 00:00:00	2	prod-01	Kotak Pensil DQLab	1	62,500	5
13	13	tr-005	dqlabcust00	2020-06-12 00:00:00	3	prod-04	Flashdisk DQLab 32 GB	2	40,000	0
14	14	tr-006	dqlabcust02	2020-06-19 00:00:00	1	prod-05	Gift Voucher DQLab 250	4	250,000	0
15	15	tr-006	dqlabcust02	2020-06-19 00:00:00	2	prod-08	Gantungan Kunci DQLab	2	15,800	0

Select Statement

SELECT adalah bagian perintah SQL yang digunakan untuk mengambil data dari satu atau beberapa tabel RDBMS.

mengambil 1 kolom tabel

```
SELECT nama_kolom FROM nama_tabel;
```

cth. `select` nama_produk `from` ms_produk_dqlab;

output :

	nama_produk
1	Kotak Pensil DQLab
2	Flashdisk DQLab 64 GB
3	Gift Voucher DQLab 100rb
4	Flashdisk DQLab 32 GB

mengambil seluruh kolom tabel

```
SELECT * FROM nama_tabel;
```

cth. `select` * `from` ms_produk_dqlab;

output :

	kode_produk	no_urut	kategori_produk	nama_produk	harga
1	prod-01	1	Alat Tulis Kantor	Kotak Pensil DQLab	62,500
2	prod-02	2	Aksesoris Komputer	Flashdisk DQLab 64 GB	55,000
3	prod-03	3	Gift & Voucher	Gift Voucher DQLab 100rb	100,000
4	prod-04	4	Aksesoris Komputer	Flashdisk DQLab 32 GB	40,000

mengambil beberapa kolom tabel

```
SELECT nama_kolom1, nama_kolom2  
FROM nama_tabel;
```

cth. `select` kategori_produk, nama_produk
`from` ms_produk_dqlab;

output :

	kategori_produk	nama_produk
1	Alat Tulis Kantor	Kotak Pensil DQLab
2	Aksesoris Komputer	Flashdisk DQLab 64 GB
3	Gift & Voucher	Gift Voucher DQLab 100rb
4	Aksesoris Komputer	Flashdisk DQLab 32 GB

Prefix

Prefix adalah bagian nama objek database dengan hirarki yang lebih tinggi. Prefix akan sangat berguna ketika kita sudah mempelajari query yang lebih kompleks seperti JOIN.

penulisan prefix

[prefix].nama_objek_database

prefix berupa nama tabel

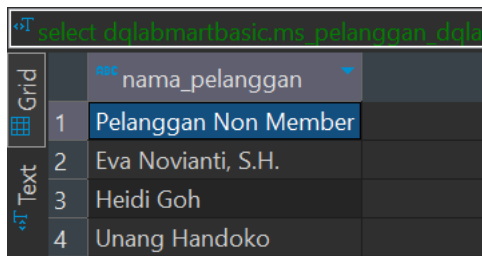
```
select ms_pelanggan_dqlab.nama_pelanggan
from ms_pelanggan_dqlab;
```

prefix berupa nama database dan tabel

```
select dqlabmartbasic.ms_pelanggan_dqlab.nama_pelanggan
from ms_pelanggan_dqlab;
```

di mana **dqlabmartbasic** merupakan database dan **ms_pelanggan_dqlab** merupakan tabel.

output :



	nama_pelanggan
1	Pelanggan Non Member
2	Eva Novianti, S.H.
3	Heidi Goh
4	Unang Handoko

Alias

Alias adalah nama sementara pengganti nama objek database asli.

Alias hanya dikenali pada query SQL yang memiliki alias tersebut.

penulisan alias

nama_objek_database as [nama_alias]

atau

nama_objek_database [nama_alias]

menggunakan alias pada kolom

```
select nama_pelanggan as np from ms_pelanggan_dqlab;
atau
```

```
select nama_pelanggan np from ms_pelanggan_dqlab;
```

menggunakan alias pada tabel

```
select nama_pelanggan from ms_pelanggan_dqlab as mpd;
atau
```

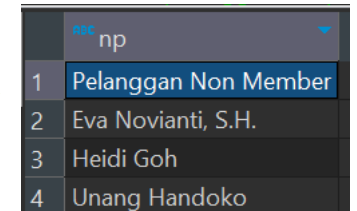
```
select nama_pelanggan from ms_pelanggan_dqlab mpd;
```

menggunakan alias pada kolom dan tabel

```
select nama_pelanggan as np from ms_pelanggan_dqlab as mpd;
atau
```

```
select nama_pelanggan np from ms_pelanggan_dqlab mpd;
```

output :



	np
1	Pelanggan Non Member
2	Eva Novianti, S.H.
3	Heidi Goh
4	Unang Handoko

Praktek 1:

Tim sales ini mencari list nama customer dari database, berupa nama customernya serta alamatnya dimana marketing ingin mencoba melakukan follow up penjualan produk dengan melakukan kunjungan ke alamat customer tersebut.

```
Select    nama_pelanggan,  
          alamat  
from ms_pelanggan_dqlab;
```

output :

	nama_pelanggan	alamat
1	Pelanggan Non Member	[NULL]
2	Eva Novianti, S.H.	Vila Sempilan, No. 67 - Kota B
3	Heidi Goh	Vila Sempilan, No. 11 - Kota B
4	Unang Handoko	Vila Sempilan, No. 1 - Kota B
5	Jokolono Sukarman	Vila Permata Intan Berkilau, Blok C5-7
6	Tommy Sinaga	Vila Permata Intan Berkilau, Blok A1/2
7	Irwan Setianto	Vila Gunung Seribu, Blok O1 - No. 1 -
8	Agus Cahyono	Vila Gunung Seribu, Blok F4 - No. 8
9	Maria Sirait	Vila Bukit Sagitarius, Gang. Sawit No. 3
10	Ir. Ita Nugraha	Vila Bukit Sagitarius, Gang Kelapa No.
11	Djoko Wardoyo, Drs.	Vila Bukit Sagitarius, Blok A1 No. 1

Praktek 2:

Sales dan Marketing ingin melihat nama produk dan harga dari setiap produk tersebut untuk membuat strategi penjualan ke konsumen berdasarkan harga produk tersebut.

```
select    nama_produk,  
          harga  
from ms_produk_dqlab;
```

output :

	nama_produk	harga
1	Kotak Pensil DQLab	62,500
2	Flashdisk DQLab 64 GB	55,000
3	Gift Voucher DQLab 100rb	100,000
4	Flashdisk DQLab 32 GB	40,000
5	Gift Voucher DQLab 250rb	250,000
6	Pulpen Multifunction + Laser DQLab	92,500
7	Tas Travel Organizer DigiSkills.id	48,000
8	Gantungan Kunci DQLab	15,800
9	Buku Planner Agenda DQLab	92,000
10	Sticky Notes DQLab 500 sheets	55,000





Beberapa Operator pada MySQL

Operator Matematika

Operator	Deskripsi	Jumlah Angka	Contoh Ekspresi Matematika dan hasilnya
<code>%</code> , <code>MOD</code>	Operator modulo (sisa bagi)	2	7 % 2 akan menghasilkan 1 7 MOD 2 akan menghasilkan 1
<code>*</code>	Operator pengali (multiplication)	2	7 * 2 akan menghasilkan 14
<code>+</code>	Operator penambah (addition)	2	7 + 2 akan menghasilkan 9
<code>-</code>	Operator pengurang (minus)	2	7 - 2 akan menghasilkan 5
<code>-</code>	Penanda nilai negatif	1	-7 adalah angka minus 7
<code>/</code>	Operator pembagi (division)	2	7 / 2 akan menghasilkan 3.5
<code>DIV</code>	Operator pembagi (division) Integer	2	7 DIV 2 akan menghasilkan 3



Operator Perbandingan

Operator	Deskripsi	Contoh Ekspresi Perbandingan dan hasilnya
<code>=</code>	Operator sama dengan (equal)	3 = 3 akan menghasilkan true 'DQLab'='UMN' akan menghasilkan false
<code><></code> atau <code>!=</code>	Operator tidak sama dengan (not equal)	4 != 3 akan menghasilkan true 4 <> 3 akan menghasilkan true
<code>></code>	Operator lebih besar (greater than)	5 > 3 akan menghasilkan true
<code><</code>	Operator lebih kecil (less than)	5 < 3 akan menghasilkan false
<code>>=</code>	Operator lebih besar sama dengan (greater than or equal)	5 >= 3 akan menghasilkan true
<code><=</code>	Operator lebih kecil sama dengan (less than or equal)	3 <= 5 akan menghasilkan true

Praktek 3:

```
Select 4 % 2 ,  
        (4 * 8) % 7,  
        (4 * 8) mod 7;
```

output :

	123 4 % 2	123 (4 * 8) % 7	123 (4 * 8) mod 7
1	0	4	4

Penjelasan :

$4 \% 2 = 0$, karena 4 habis dibagi 2 sehingga sisanya 0

$(4 * 8) \% 7 = 4$, karena $4 * 8 = 32$ kemudian $32 \% 7 = 4$ sisa 4 di mana $7 * 4 = 28$ kelebihan 4.

$28 + 4 = 32$

sehingga sisa baginya = 4

Hasilnya sama dengan $(4 * 8) \bmod 7$

Praktek 4:

```
Select 1 = true, # nilai true itu 1 maka 1  
= 1 hasilnya true/1  
        1 = false, # nilai false itu 0 maka  
1 = 0 hasilnya false/0  
        5 >= 5,  
        5.2 = 5.20000,  
        NULL = 1,  
        NULL = NULL;
```

output :

	123 1 = true	123 1 = false	123 5 >= 5	123 5.2 = 5.20000	123 NULL = 1	123 NULL = NULL
1	1	0	1	1	[NULL]	[NULL]



Fungsi

Referensi fungsi lainnya:
<https://dev.mysql.com/doc/refman/8.0/en/functions.html>



Nama Fungsi	Deskripsi	Contoh Penggunaan dan hasilnya
POW (X, D)	Fungsi pangkat dengan dua argumen, yaitu X untuk nilai basis dan D untuk nilai pangkatnya	POW(3, 2) akan menghasilkan nilai 9
ROUND (X)	Fungsi untuk membulatkan angka pecahan X	ROUND(3.14) akan menghasilkan nilai 3 ROUND(3.54) akan menghasilkan nilai 4
ROUND (X, D)	Fungsi untuk membulatkan angka pecahan X sebanyak D desimal	ROUND(3.155, 1) akan menghasilkan nilai 3.2 ROUND(3.155, 2) akan menghasilkan nilai 3.16
FLOOR (X)	Fungsi untuk membulatkan angka pecahan ke bawah	FLOOR(4.28) akan menghasilkan nilai 4 FLOOR(4.78) akan menghasilkan nilai 4
CEILING (X)	Fungsi untuk membulatkan angka pecahan ke atas	CEILING(4.39) akan menghasilkan nilai 5 CEILING(4.55) akan menghasilkan nilai 5
NOW ()	Fungsi untuk menghasilkan jam dan tanggal saat ini	NOW() akan menghasilkan tanggal dan jam saat ini
YEAR (X)	Fungsi untuk mengambil elemen tahun dari tanggal X	YEAR('2022-05-03') akan menghasilkan 2022
MONTH (X)	Fungsi untuk mengambil elemen bulan dari tanggal X	MONTH('2022-05-03') akan menghasilkan 5
DAY (X)	Fungsi untuk mengambil elemen hari dari tanggal X	DAY('2022-05-03') akan menghasilkan 3
DATEDIFF (X1, X2)	Fungsi untuk menghitung perbedaan hari dari tanggal X1 dan X2	DATEDIFF('2022-05-03', '2022-07-22') akan menghasilkan -80 DATEDIFF('2022-07-22', '2022-05-03') akan menghasilkan 80

Fungsi pada Text

Referensi fungsi
lainnya:
<https://www.postgresql.org/docs/9.1/functions-string.html>



Fungsi	Deskripsi	Contoh	Hasil
<code>concat()</code>	Menggabungkan semua argumen/input, NULL value akan diabaikan	<code>concat('abcde', 2, NULL, 22)</code>	abcde222
<code>split_part()</code>	Membagi string/text berdasarkan delimiter/pemisah yang ditentukan, kemudian mengembalikan value berdasarkan posisi string yang diinginkan (hitungan posisi mulai dari 1)	<code>split_part('abc@def@ghi', '@', 2)</code>	def
<code>substr()</code>	Mengekstrak karakter/string yang diinginkan	<code>substr('alphabet', 3, 2)</code>	ph
<code>length()</code>	Menghitung jumlah karakter dalam string/text	<code>length('jose')</code>	4
<code>replace()</code>	Mengganti karakter dalam suatu string/text	<code>replace('abcdefabcdef', 'cd', 'XX')</code>	abXXefabXXef
<code>trim()</code>	Menghapus karakter dalam suatu string	<code>trim(both 'x' from 'xTomxx');</code> <code>trim(leading 'x' from 'xTomxx');</code> <code>trim(trailing 'x' from 'xTomxx');</code>	Tom; Tomxx; xTom;
<code>upper()</code>	Mengubah huruf kecil ke huruf besar	<code>upper('tom')</code>	TOM
<code>lower()</code>	Mengubah huruf besar ke huruf kecil	<code>lower('TOM')</code>	tom

Praktek 5:

```
Select    datediff('2023-03-20', now()),
          year('2022-07-23'),
          month('2022-07-23'),
          day('2022-07-23'),
          year(now());
```

output :

Enter a SQL expression to filter results (use Ctrl+Space)					
	datediff('2023-03-20', now())	year('2022-07-23')	month('2022-07-23')	day('2022-07-23')	year(now())
1	-4	2,022	7	23	2,023

Praktek 6:

```
Select    datediff(now(), tgl_transaksi)
          jarak_hari_transaksi,
          year(tgl_transaksi) tahun_transaksi
from tr_penjualan_dqlab;
```

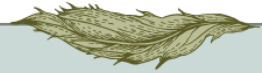
output :

	jarak_hari_transaksi	tahun_transaksi
1	1,057	2,020
2	1,057	2,020
3	1,057	2,020
4	1,057	2,020
5	1,057	2,020
6	1,057	2,020
7	1,057	2,020
8	1,055	2,020
9	1,055	2,020
10	1,055	2,020
11	1,015	2,020
12	1,015	2,020
13	1,015	2,020
14	1,008	2,020



Filtering dengan Where

Penyaringan atau **filtering** pada SQL adalah suatu proses untuk membatasi output data yang dihasilkan berdasarkan kondisi atau kriteria tertentu.



Sintaks : `select ... from nama_tabel where [ekspresi_perbandingan];`

Contoh :

```
Select      nama_produk,  
            harga  
from ms_produk_dqlab  
where harga > 50000;
```

output :

	nama_produk	harga
1	Kotak Pensil DQLab	62,500
2	Flashdisk DQLab 64 GB	55,000
3	Gift Voucher DQLab 100rb	100,000
4	Gift Voucher DQLab 250rb	250,000
5	Pulpen Multifunction + Laser DQLab	92,500
6	Buku Planner Agenda DQLab	92,000
7	Sticky Notes DQLab 500 sheets	55,000

```
Select      nama_produk,  
            qty  
from tr_penjualan_dqlab  
where qty > 3;
```

output :

	nama_produk	qty
1	Kotak Pensil DQLab	5
2	Sticky Notes DQLab 500 sheets	4
3	Sticky Notes DQLab 500 sheets	5
4	Flashdisk DQLab 32 GB	4
5	Gift Voucher DQLab 250rb	4

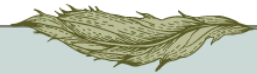
```
Select      nama_produk,  
            qty  
from tr_penjualan_dqlab  
where qty > 3 and  
month(tgl_transaksi) = 5;
```

output :

	nama_produk	qty
1	Kotak Pensil DQLab	5
2	Sticky Notes DQLab 500 sheets	4
3	Sticky Notes DQLab 500 sheets	5
4	Flashdisk DQLab 32 GB	4

Operator Perbandingan Like

LIKE pada SQL adalah suatu operator pada SQL untuk filter literal atau field teks dengan pola tertentu.



Sintaks : `select ... from nama_tabel where nama_field like [POLA];`

Pola Teks	Keterangan	Contoh Teks yang Dikenali oleh Pola Ini
'F%'	Suatu teks yang diawali karakter F, dan bisa atau tidak diikuti oleh karakter apapun setelah F ini	'F' 'Fera' 'flask Disk'
'F_'	Seluruh teks terdiri dari dua karakter, diawali karakter F dan diikuti satu karakter apapun	'F1' 'fa' 'f@'
'%f%'	Suatu teks yang bisa diawali dan diakhiri karakter apapun, dengan minimal mengandung satu huruf f	'F' 'Fera' 'afrika'
'_Q%'	Suatu teks yang karakter keduanya adalah Q	'!Q' 'DQLab' '!Q'

Contoh:

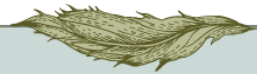
```
select nama_produk
from tr_penjualan_dqlab
where nama_produk like 'g%';
# menampilkan kolom nama_produk yang diawali
dengan huruf 'g'
```

output :

	nama_produk
1	Gift Voucher DQLab 100rb
2	Gift Voucher DQLab 250rb
3	Gantungan Kunci DQLab

Operator Logika

Operator Logika pada SQL adalah suatu operator untuk menggabungkan dua ekspresi perbandingan atau literal boolean. Hasil dari operator ini adalah nilai boolean.



Operator	Keterangan
AND atau &&	Operator logika untuk kondisi DAN
OR atau 	Operator logika untuk kondisi ATAU
NOT atau !	Operator logika untuk kondisi negasi
XOR	Operator logika Exclusive OR

X	NOT X
TRUE	FALSE
FALSE	TRUE

Contoh operator and:

```
Select    nama_produk,  
          qty  
from tr_penjualan_dqlab  
where nama_produk like 'g%' and  
      qty > 3;
```

output :

	nama_produk	qty
1	Gift Voucher DQLab 250rb	4

Contoh operator or:

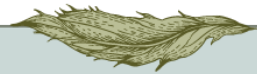
```
Select    nama_produk,  
          qty  
from tr_penjualan_dqlab  
where nama_produk like 'g%'  
or qty > 3;
```

output :

	nama_produk	qty
1	Kotak Pensil DQLab	5
2	Gift Voucher DQLab 100rb	2
3	Sticky Notes DQLab 500 sh	4
4	Sticky Notes DQLab 500 sh	5
5	Flashdisk DQLab 32 GB	4
6	Gift Voucher DQLab 250rb	4
7	Gantungan Kunci DQLab	2

Order By

ORDER BY adalah keyword SQL untuk menyusun urutan data sesuai nilai field atau nilai pengolahan field.



Sintaks : **select** ... **from** nama_tabel **where** [ekspresi perbandingan] **order by** nama_field [asc/desc];

Catatan :

- Secara default order by adalah **asc (ascending)** yaitu mengurutkan data dari nilai terkecil ke terbesar. Secara natural, asc tidak perlu ditulis.
- **Descending** atau **desc** untuk mengurutkan data dari nilai terbesar ke terkecil.
- Order by juga bisa menggunakan hasil perhitungan dari nama field, baik dengan operator maupun fungsi.
- Order by bisa digabungkan dengan keyword **where** untuk melakukan pengurutan dari hasil filtering data. Keyword **where** diletakkan sebelum order by.

menampilkan kolom nama_produk, harga, dan diskon dari transaksi penjualan yang memiliki diskon dan diurutkan berdasarkan harga tertinggi

```
select nama_produk, harga, diskon_persen
from tr_penjualan_dqlab
where diskon_persen != 0
order by harga desc;
```

	nama_produk	harga	diskon_persen
1	Flash disk DQLab 32 GB	100,000	25
2	Kotak Pensil DQLab	62,500	5
3	Flashdisk DQLab 64 GB	55,000	12.5

menampilkan kolom nama produk, kuantitas pembelian dan harga dari transaksi penjualan yang memiliki harga minimal seratus ribu rupiah dan diurutkan berdasarkan harga tertinggi

```
select nama_produk, qty, harga
from tr_penjualan_dqlab
where harga >= 100000
order by harga desc;
```

	nama_produk	qty	harga
1	Gift Voucher DQLab 250rb	4	250,000
2	Flash disk DQLab 32 GB	1	100,000
3	Gift Voucher DQLab 100rb	2	100,000

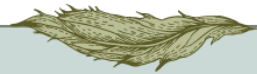
menampilkan kolom nama produk, kuantitas pembelian dan harga dari transaksi penjualan yang memiliki harga minimal seratus ribu rupiah atau nama produk berawalan karakter 'T' dan diurutkan berdasarkan diskon tertinggi

```
select nama_produk, qty, harga
from tr_penjualan_dqlab
where harga >= 100000 or nama_produk like 'T%'
order by diskon_persen desc;
```

	nama_produk	qty	harga
1	Flash disk DQLab 32 GB	1	100,000
2	Gift Voucher DQLab 100rb	2	100,000
3	Tas Travel Organizer DQLab	1	48,000
4	Gift Voucher DQLab 250rb	4	250,000

Fungsi Agregasi

FUNGSI AGREGASI adalah fungsi yang digunakan mengolah beberapa row data (bukan satu row data saja) untuk menghasilkan suatu nilai.



Fungsi Agregasi	Deskripsi
SUM	Menghasilkan nilai penjumlahan dari seluruh row
AVG	Menghitung nilai rata-rata dari nilai field dari seluruh row yang dihasilkan dari query
COUNT	Menghitung jumlah dari seluruh row yang dihasilkan dari query
COUNT (DISTINCT)	Menghitung jumlah nilai unik dari field tertentu dari seluruh row yang dihasilkan dari query
MAX	Mengambil nilai terbesar suatu field dari seluruh row yang dihasilkan dari query
MIN	Mengambil nilai terkecil suatu field dari seluruh row yang dihasilkan dari query

```
# menampilkan nilai total, rata-rata, max, dan min dari kolom harga, count semua kolom dan  
count(distinct) kolom nama_produk dari transaksi penjualan  
select sum(harga), avg(harga), max(harga), min(harga), count(*), count(distinct  
nama_produk)  
from tr_penjualan_dqlab;
```

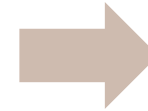
output :

	sum(harga)	avg(harga)	max(harga)	min(harga)	count(*)	count(distinct nama_produk)
1	1,107,800	73,853.3333333333	250,000	15,800	15	10

FUNGSI COUNT(DISTINCT) adalah fungsi agregasi yang digunakan untuk menghitung nilai unik yang terdapat pada seluruh row dari suatu query.

Contoh:

nama
Charlie
Budi
Amir
Budi
Amir
Budi



COUNT(DISTINCT nama)

nama
Amir
Budi
Charlie

Group By

GROUP BY adalah pengelompokan isi data dari satu atau beberapa field pada SQL, dan biasanya digabungkan dengan penggunaan fungsi agregasi.

Sintaks :

```
select field1, field2, fungsi(fields)
from nama_tabel
group by field1, field2;
```

Melihat daftar nama produk yang total jumlah produk terjual per kelompok nama produk adalah di atas nilai 4

```
select nama_produk, sum(qty) jumlah_produk
from tr_penjualan_dqlab
group by nama_produk
having sum(qty) > 4;
```

output :

	nama_produk	jumlah_produk
1	Kotak Pensil DQLab	6
2	Buku Planner Agenda DQSQuad	6
3	Flashdisk DQLab 32 GB	9
4	Sticky Notes DQLab 500 sheets	9

Melihat daftar nama produk yang total jumlah produk terjual per kelompok nama produk sama dengan 9.

```
select nama_produk, sum(qty) jumlah_produk
from tr_penjualan_dqlab
group by nama_produk
having sum(qty) = 9;
```

output :

	nama_produk	jumlah_produk
1	Flashdisk DQLab 32 GB	9
2	Sticky Notes DQLab 500 sheets	9

Having

HAVING adalah keyword SQL untuk melakukan filtering pada hasil fungsi agregasi dengan GROUP BY, dimana WHERE tidak bisa digunakan.

Sintaks :

```
select ...
from nama_tabel
group by ...
having [ekspresi_perbandingan];
```

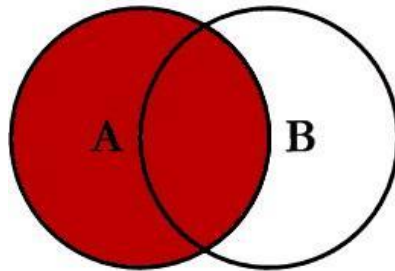
Melihat daftar kelompok nama produk dan total nilai penjualan (harga dikalikan jumlah dikurangi diskon) dengan dan urutan berdasarkan nilai penjualan terbesar

```
select nama_produk, sum((harga * qty) - diskon_persen)
total_penjualan
from tr_penjualan_dqlab
group by nama_produk
order by total_penjualan desc;
```

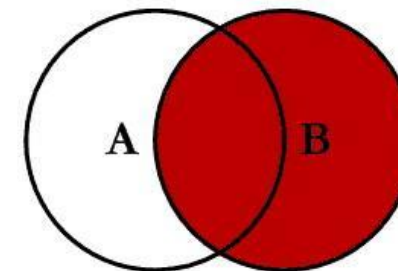
output :

	nama_produk	total_penjualan
1	Gift Voucher DQLab 250rb	1,000,000
2	Buku Planner Agenda DQSQuad	552,000
3	Sticky Notes DQLab 500 sheets	495,000
4	Kotak Pensil DQLab	374,995
5	Flashdisk DQLab 32 GB	360,000
6	Gift Voucher DQLab 100rb	200,000
7	Flashdisk DQLab 64 GB	109,987.5
8	Flash disk DQLab 32 GB	99,975
9	Tas Travel Organizer DQLab	48,000
10	Gantungan Kunci DQLab	31,600

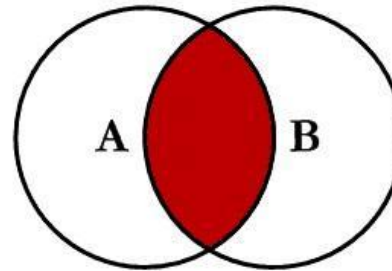
SQL Joins



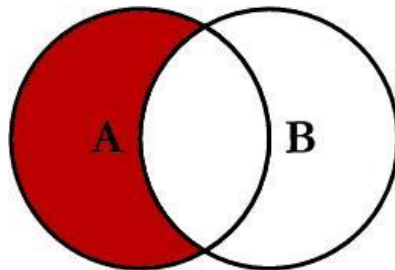
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



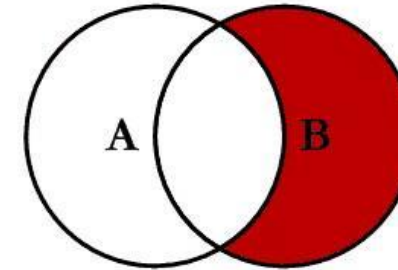
```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



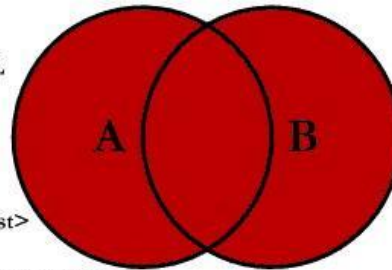
```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



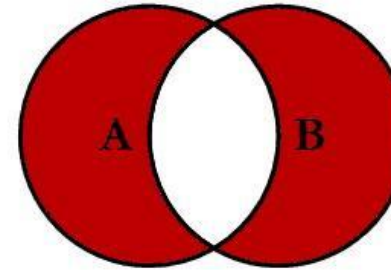
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
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
```



```
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
```

© C.L. Moffatt, 2008

JOINS

JOIN adalah keyword klausa SQL yang digunakan untuk menghubungkan dua tabel berdasarkan satu atau beberapa kolom tertentu.

Sintaks : `select ... from nama_tabel1 join nama_tabel2 on nama_tabel1.kolom = nama_tabel2.kolom ;`

Cross Join

Jika **JOIN** terjadi tanpa ekspresi filtering yang menggunakan relationship keys atau columns, maka setiap row data dari tiap tabel akan direlasikan satu sama lain. Kondisi ini sering disebut sebagai **CROSS JOIN**.

```
select tp.kode_pelanggan, mp.nama_pelanggan, tp.qty
from tr_penjualan_dqlab tp
join ms_pelanggan_dqlab mp
on true;
```

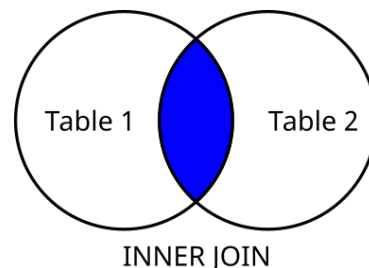
output dari query di atas akan menghasilkan 165 baris karena setiap baris pada tabel tr_penjualan_dqlab akan direlasikan ke semua baris yang ada di tabel ms_penjualan_dqlab :

	ABC kode_pelanggan	ABC nama_pelanggan	123 qty
157	dqlabcust02	Maria Sirait	2
158	dqlabcust02	Agus Cahyono	2
159	dqlabcust02	Irwan Setianto	2
160	dqlabcust02	Tommy Sinaga	2
161	dqlabcust02	Jokolono Sukarman	2
162	dqlabcust02	Unang Handoko	2
163	dqlabcust02	Heidi Goh	2
164	dqlabcust02	Eva Novianti, S.H.	2
165	dqlabcust02	Pelanggan Non Member	2

Inner Join atau Join

JOIN atau **INNER JOIN** adalah tipe join yang mengambil data dari kedua sisi tabel hanya yang berhubungan dari key columns.

Kalau digambarkan dengan diagram Venn maka bentuknya seperti berikut:



```
select
tp.kode_pelanggan,
mp.nama_pelanggan,
tp.qty
from tr_penjualan_dqlab
tp
join ms_pelanggan_dqlab
mp
on tp.kode_pelanggan =
mp.kode_pelanggan;
```

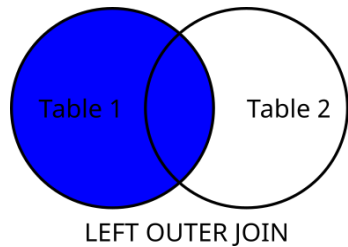
output dari query di samping menghasilkan 15 row data.

	ABC kode_pelanggan	ABC nama_pelanggan	123 qty
1	dqlabcust07	Agus Cahyono	5
2	dqlabcust07	Agus Cahyono	1
3	dqlabcust07	Agus Cahyono	3
4	dqlabcust07	Agus Cahyono	3
5	dqlabcust00	Pelanggan Non Member	2
6	dqlabcust00	Pelanggan Non Member	4
7	dqlabcust00	Pelanggan Non Member	1
8	dqlabcust03	Unang Handoko	2
9	dqlabcust03	Unang Handoko	5
10	dqlabcust03	Unang Handoko	4
11	dqlabcust00	Pelanggan Non Member	3
12	dqlabcust00	Pelanggan Non Member	1
13	dqlabcust00	Pelanggan Non Member	2
14	dqlabcust02	Heidi Goh	4
15	dqlabcust02	Heidi Goh	2

Left Outer Join atau Left Join

LEFT OUTER JOIN adalah tipe join yang mengambil seluruh data dari sisi tabel sebelah kiri, dan untuk tabel sebelah kanan hanya row-row yang memiliki relasi key columns. Bagian relasi yang tidak mendapatkan data, akan menampilkan NULL.

Kalau digambarkan dengan diagram Venn maka bentuknya seperti berikut:



output dari query di samping menghasilkan 15 row data.

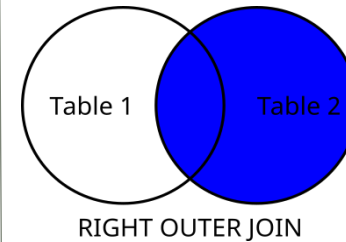
	ABC kode_pelanggan	ABC nama_pelanggan	123 qty
1	dqlabcust07	Agus Cahyono	5
2	dqlabcust07	Agus Cahyono	1
3	dqlabcust07	Agus Cahyono	3
4	dqlabcust07	Agus Cahyono	3
5	dqlabcust00	Pelanggan Non Member	2
6	dqlabcust00	Pelanggan Non Member	4
7	dqlabcust00	Pelanggan Non Member	1
8	dqlabcust03	Unang Handoko	2
9	dqlabcust03	Unang Handoko	5
10	dqlabcust03	Unang Handoko	4
11	dqlabcust00	Pelanggan Non Member	3
12	dqlabcust00	Pelanggan Non Member	1
13	dqlabcust00	Pelanggan Non Member	2
14	dqlabcust02	Heidi Goh	4
15	dqlabcust02	Heidi Goh	2

```
select
tp.kode_pelanggan,
mp.nama_pelanggan,
tp.qty
from tr_penjualan_dqlab
tp
left join
ms_pelanggan_dqlab mp
on tp.kode_pelanggan =
mp.kode_pelanggan;
```

Right Outer Join atau Right Join

RIGHT OUTER JOIN adalah tipe join yang mengambil seluruh data dari sisi tabel sebelah kanan, dan untuk tabel sebelah kiri hanya row-row yang memiliki relasi key columns. Bagian tabel yang tidak mendapatkan relasi, akan menampilkan NULL.

Kalau digambarkan dengan diagram Venn maka bentuknya seperti berikut:



output dari query di samping menghasilkan 22 row data.

	ABC kode_pelanggan	ABC nama_pelanggan	123 qty
1	dqlabcust00	Pelanggan Non Member	2
2	dqlabcust00	Pelanggan Non Member	1
3	dqlabcust00	Pelanggan Non Member	3
4	dqlabcust00	Pelanggan Non Member	1
5	dqlabcust00	Pelanggan Non Member	4
6	dqlabcust00	Pelanggan Non Member	2
7	[NULL]	Eva Novianti, S.H.	[NULL]
8	dqlabcust02	Heidi Goh	2
9	dqlabcust02	Heidi Goh	4
10	dqlabcust03	Unang Handoko	4
11	dqlabcust03	Unang Handoko	5
12	dqlabcust03	Unang Handoko	2
13	[NULL]	Jokolono Sukarman	[NULL]
14	[NULL]	Tommy Sinaga	[NULL]
15	[NULL]	Irwan Setianto	[NULL]
16	dqlabcust07	Agus Cahyono	3
17	dqlabcust07	Agus Cahyono	3
18	dqlabcust07	Agus Cahyono	1
19	dqlabcust07	Agus Cahyono	5
20	[NULL]	Maria Sirait	[NULL]
21	[NULL]	Ir. Ita Nugraha	[NULL]
22	[NULL]	Djoko Wardoyo, Drs.	[NULL]

```
select
tp.kode_pelanggan,
mp.nama_pelanggan,
tp.qty
from tr_penjualan_dqlab
tp
right join
ms_pelanggan_dqlab mp
on tp.kode_pelanggan =
mp.kode_pelanggan;
```


Order By pada Join

Sintaks :

```
select ... from nama_tabel1 [tipe join] nama_tabel2 on
[ekspresi_filtering]
order by [fields] ;
```

Select

```
tp.kode_pelanggan, mp.nama_pelanggan,
mpd.kategori_produk, mpd.nama_produk,
tp.qty
from ms_pelanggan_dqlab mp
left join tr_penjualan_dqlab tp
on mp.kode_pelanggan = tp.kode_pelanggan
left join ms_produk_dqlab mpd
on mpd.kode_produk = tp.kode_produk
order by qty desc;
```

output:

	asc kode_pelanggan	asc nama_pelanggan	asc kategori_produk	asc nama_produk	123 qty
13	dqlabcust00	Pelanggan Non Member	Alat Tulis Kantor	Kotak Pensil DQLab	1
14	dqlabcust00	Pelanggan Non Member	Gift & Voucher	Tas Travel Organizer DigiSkills.id	1
15	dqlabcust07	Agus Cahyono	Gift & Voucher	Gift Voucher DQLab 100rb	1
16	[NULL]	Eva Novianti, S.H.	[NULL]	[NULL]	[NULL]
17	[NULL]	Jokolono Sukarman	[NULL]	[NULL]	[NULL]
18	[NULL]	Tommy Sinaga	[NULL]	[NULL]	[NULL]
19	[NULL]	Irwan Setianto	[NULL]	[NULL]	[NULL]
20	[NULL]	Maria Sirait	[NULL]	[NULL]	[NULL]
21	[NULL]	Ir. Ita Nugraha	[NULL]	[NULL]	[NULL]
22	[NULL]	Djoko Wardoyo, Drs.	[NULL]	[NULL]	[NULL]

Group By pada Join

Sintaks :

```
select [fields], fungsi(...) from nama_tabel1 [tipe join]
nama_tabel2 on [ekspresi_filtering]
group by [fields] ;
```

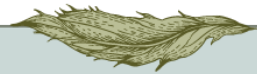
```
select mpd.kategori_produk , mpd.nama_produk , sum(tp.qty) as
qty
from ms_pelanggan_dqlab mp
left join tr_penjualan_dqlab tp
on mp.kode_pelanggan = tp.kode_pelanggan
left join ms_produk_dqlab mpd
on mpd.kode_produk = tp.kode_produk
group by mpd.kategori_produk , mpd.nama_produk
order by qty desc;
```

output:

	asc kategori_produk	asc nama_produk	123 qty
1	Aksesoris Komputer	Flashdisk DQLab 32 GB	9
2	Alat Tulis Kantor	Sticky Notes DQLab 500 sheets	9
3	Alat Tulis Kantor	Kotak Pensil DQLab	6
4	Alat Tulis Kantor	Buku Planner Agenda DQLab	6
5	Gift & Voucher	Gift Voucher DQLab 250rb	4
6	Gift & Voucher	Gift Voucher DQLab 100rb	3
7	Gift & Voucher	Gantungan Kunci DQLab	2
8	Aksesoris Komputer	Flashdisk DQLab 64 GB	2
9	Gift & Voucher	Tas Travel Organizer DigiSkills.id	1
10	[NULL]	[NULL]	[NULL]

Union

UNION adalah keyword klausa SQL yang digunakan untuk menggabungkan dua tabel secara vertikal. Di mana jumlah kolom dari query masing-masing harus sama.



Sintaks :

```
select [fields] from nama_tabel1
union
select [fields] from nama_tabel2;
```

Catatan:

- Jumlah field yang ditampilkan pada union harus sama untuk tiap query.
- nama-nama field yang ditampilkan adalah mengikuti query yang pertama dari konstruksi UNION
- Union pada tabel yang sama maupun berbeda tabel akan menghasilkan row data yang unik (tidak duplikat)

query:

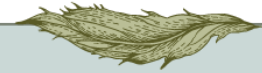
```
select kode_produk, nama_produk
from ms_produk_dqlab
union
select kode_produk, nama_produk
from tr_penjualan_dqlab
order by kode_produk;
```

output:

	kode_produk	nama_produk
1	prod-01	Kotak Pensil DQLab
2	prod-02	Flashdisk DQLab 64 GB
3	prod-03	Gift Voucher DQLab 100rb
4	prod-03	Flash disk DQLab 32 GB
5	prod-04	Flashdisk DQLab 32 GB
6	prod-05	Gift Voucher DQLab 250rb
7	prod-06	Pulpen Multifunction + Laser DQLab
8	prod-07	Tas Travel Organizer DigiSkills.id
9	prod-07	Tas Travel Organizer DQLab
10	prod-08	Gantungan Kunci DQLab
11	prod-09	Buku Planner Agenda DQLab
12	prod-09	Buku Planner Agenda DQSquad
13	prod-10	Sticky Notes DQLab 500 sheets

Union All

UNION ALL adalah tipe union yang akan menampilkan hasil gabungan apa adanya, dan tidak coba menjadikan row menjadi unik atau menghilangkan duplikat.



query:

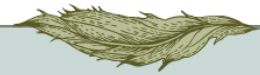
```
select kode_produk, nama_produk
from ms_produk_dqlab
union all
select kode_produk, nama_produk
from tr_penjualan_dqlab
order by kode_produk;
```

output:

	kode_produk	nama_produk
6	prod-03	Gift Voucher DQLab 100rb
7	prod-03	Flash disk DQLab 32 GB
8	prod-03	Gift Voucher DQLab 100rb
9	prod-04	Flashdisk DQLab 32 GB
10	prod-04	Flashdisk DQLab 32 GB
11	prod-04	Flashdisk DQLab 32 GB
12	prod-04	Flashdisk DQLab 32 GB
13	prod-05	Gift Voucher DQLab 250rb
14	prod-05	Gift Voucher DQLab 250rb
15	prod-06	Pulpen Multifunction + Laser DQLab
16	prod-07	Tas Travel Organizer DigiSkills.id
17	prod-07	Tas Travel Organizer DQLab
18	prod-08	Gantungan Kunci DQLab
19	prod-08	Gantungan Kunci DQLab
20	prod-09	Buku Planner Agenda DQLab
21	prod-09	Buku Planner Agenda DQSQuad
22	prod-09	Buku Planner Agenda DQSQuad
23	prod-10	Sticky Notes DQLab 500 sheets
24	prod-10	Sticky Notes DQLab 500 sheets
25	prod-10	Sticky Notes DQLab 500 sheets

Limit

LIMIT adalah keyword khusus dari MySQL untuk membatasi hasil dari query.



Sintaks :

```
select [fields] from nama_tabel1 ...  
limit [offset], [rowcount];
```

Di mana:

- Offset : nomor urut baris mulai diambilnya hasil oleh limit, nomor urut baris pertama adalah 0, nomor urut baris kedua adalah 1, dstnya
- Rowcount : jumlah baris yang diambil

menampilkan kolom produk dan jumlah produk dari tabel transaksi penjualan kemudian diurutkan dari jumlah produk yang paling banyak dan diambil 5 produk teratas

```
select nama_produk, sum(qty) jumlah_produk  
from tr_penjualan_dqlab  
group by nama_produk  
order by jumlah_produk desc  
limit 5;
```

output:

	nama_produk	jumlah_produk
1	Flashdisk DQLab 32 GB	9
2	Sticky Notes DQLab 500 sheets	9
3	Kotak Pensil DQLab	6
4	Buku Planner Agenda DQSQuad	6
5	Gift Voucher DQLab 250rb	4



Thank you



Fitri Dwi Hastuti
[LinkedIn](#) | [Medium](#) | [Github](#)