

Nama : Martiano Alesandri Jedeot

Nim : 23241070

Kelas : B

Modul : (4) part 2-(ORDER BY,FUNGSI AGREGASI, GROUP BY, HAVING)

```
9      -- Ambil nama produk dan qty dari tabel penjualan urutkan qty nya (1,2,3...)
10 •   SELECT nama_produk, qty FROM tr_penjualan ORDER BY qty;
```

HASIL

	nama_produk	qty
▶	Flash disk DQLab 32 GB	1
	Tas Travel Organizer DQLab	1
	Kotak Pensil DQLab	1
	Gift Voucher DQLab 100rb	2
	Flashdisk DQLab 64 GB	2
	Flashdisk DQLab 32 GB	2
	Gantungan Kunci DQLab	2
	Buku Planner Agenda DQSquad	3
	Flashdisk DQLab 32 GB	3
	Buku Planner Agenda DQSquad	3
	Sticky Notes DQLab 500 sheets	4
	Flashdisk DQLab 32 GB	4
	Gift Voucher DQLab 250rb	4
	Kotak Pensil DQLab	5
	Sticky Notes DQLab 500 sheets	5

PENJELASAN

- **SELECT nama_produk, qty**

→ Artinya: ambil (tampilkan) dua kolom, yaitu:

- nama_produk(nama produk yang dijual)
- qty(jumlah produk yang terjual)

- **FROM tr_penjualan**

→ Artinya: data diambil dari tabel yang bernama **tr_penjualan**, yang bisa diartikan sebagai "tabel transaksi penjualan".

- **ORDER BY qty**

→ Artinya: urutkan has**qty**(jumlah produk) secara default **menaik (ascending)** , dari jumlah terkecil ke terbesar.

```
12      -- ambil nama produk dan qty dari tb penjualan urutkan qty dan nama produk (a-z)
13 •   SELECT nama_produk, qty FROM tr_penjualan ORDER BY qty, nama_produk;
```

HASIL

	nama_produk	qty
▶	Flash disk DQLab 32 GB	1
	Kotak Pensil DQLab	1
	Tas Travel Organizer DQLab	1
	Flashdisk DQLab 32 GB	2
	Flashdisk DQLab 64 GB	2
	Gantungan Kunci DQLab	2
	Gift Voucher DQLab 100rb	2
	Buku Planner Agenda DQSquad	3
	Buku Planner Agenda DQSquad	3
	Flashdisk DQLab 32 GB	3
	Flashdisk DQLab 32 GB	4
	Gift Voucher DQLab 250rb	4
	Sticky Notes DQLab 500 sheets	4
	Kotak Pensil DQLab	5
	Sticky Notes DQLab 500 sheets	5

PENJELASAN

- **SELECT nama_produk, qty**

→ Artinya: ambil dan tampilkan dua kolom, yaitu:

- nama_produk: nama produk yang dijual
- qty: jumlah produk yang terjual

- **FROM tr_penjualan**

→ Artinya: data diambil dari tabel bernama **tr_penjualan**, yang biasanya menyimpan data transaksi penjualan.

- **ORDER BY qty, nama_produk**

→ Artinya:

- Pertama, **urutkan berdasarkan qty(jumlah produk)** dari yang paling kecil ke paling besar (default: **ASC / ascending**).
- Jika ada beberapa baris yang memiliki nilai **qty** yang sama, maka **urutkan lagi berdasarkan nama_produk secara alfabetis** .

```
16 -- tampilkan semua kolom dari tabel tr_penjualan dengan mengurutkan qty dan tgl_transaksi 1,2,3....
17 • SELECT * FROM tr_penjualan ORDER BY qty, tgl_transaksi;
```

HASIL

	kode_urut_transaksi	kode_transaksi	kode_pelanggan	tgl_transaksi	no_urut	kode_produk	nama_produk	qty	harga	diskon_persen
▶	2	tr-001	dqlabcust07	2020-05-01 00:00:00	2	prod-03	Flash disk DQLab 32 GB	1	100000	25
	7	tr-002	dqlabcust00	2020-05-01 00:00:00	3	prod-07	Tas Travel Organizer DQLab	1	48000	0
	12	tr-005	dqlabcust00	2020-06-12 00:00:00	2	prod-01	Kotak Pensil DQLab	1	62500	5
	5	tr-002	dqlabcust00	2020-05-01 00:00:00	1	prod-03	Gift Voucher DQLab 100rb	2	100000	0
	8	tr-003	dqlabcust03	2020-05-03 00:00:00	1	prod-02	Flashdisk DQLab 64 GB	2	55000	12.5
	13	tr-005	dqlabcust00	2020-06-12 00:00:00	3	prod-04	Flashdisk DQLab 32 GB	2	40000	0
	15	tr-006	dqlabcust02	2020-06-19 00:00:00	2	prod-08	Gantungan Kunci DQLab	2	15800	0
	3	tr-001	dqlabcust07	2020-05-01 00:00:00	3	prod-09	Buku Planner Agenda DQSquad	3	92000	0
	4	tr-001	dqlabcust07	2020-05-01 00:00:00	4	prod-04	Flashdisk DQLab 32 GB	3	40000	0
	11	tr-005	dqlabcust00	2020-06-12 00:00:00	1	prod-09	Buku Planner Agenda DQSquad	3	92000	0
	6	tr-002	dqlabcust00	2020-05-01 00:00:00	2	prod-10	Sticky Notes DQLab 500 sheets	4	55000	0
	10	tr-004	dqlabcust03	2020-05-03 00:00:00	2	prod-04	Flashdisk DQLab 32 GB	4	40000	0
	14	tr-006	dqlabcust02	2020-06-19 00:00:00	1	prod-05	Gift Voucher DQLab 250rb	4	250000	0
	1	tr-001	dqlabcust07	2020-05-01 00:00:00	1	prod-01	Kotak Pensil DQLab	5	62500	0
	9	tr-004	dqlabcust03	2020-05-03 00:00:00	1	prod-10	Sticky Notes DQLab 500 sheets	5	55000	0

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel **tr_penjualan**.

Ini bisa mencakup kolom seperti: **id_transaksi**, **nama_produk**, **qty**, **harga**, **tgl_transaksi**, dan lain-lain (tergantung struktur tabel).

- **FROM tr_penjualan**

→ Artinya: ambil data dari tabel bernama **tr_penjualan**, yaitu tabel yang menyimpan data transaksi penjualan.

- **ORDER BY qty, tgl_transaksi**

→ Artinya:

- **Urutkan data berdasarkan qty** terlebih dahulu dari **terkecil ke terbesar** (default ASC).
- Jika ada baris-baris dengan nilai **qty** yang sama, maka diurutkan **berdasarkan tgl_transaksi** tanggal **terlama ke terbaru**.

```
19 -- tampilkan semua kolom dari tabel ms_pelanggan dengan mengurutkan nama pelanggan dengan awalan huruf A ke Z
20 • SELECT * FROM pelanggan ORDER BY nama_pelanggan;
```

HASIL

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

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel pelanggan.

Kolom-kolom tersebut bisa saja meliputi: id_pelanggan, nama_pelanggan, alamat, no_telepon, email, dll — tergantung struktur tabelnya.

- **FROM pelanggan**

→ Artinya: data diambil dari tabel **pelanggan**, yang menyimpan informasi semua pelanggan.

- **ORDER BY nama_pelanggan**

→ Artinya: **urutkan data berdasarkan kolom nama_pelanggan** secara **alfabetis (A ke Z)**. Ini adalah urutan **menaik (ascending)** secara default.

```
22 -- tampilkan semua kolom dari tabel ms_pelanggan dengan mengurutkan alamat dengan awalan huruf A ke Z
23 • SELECT * FROM pelanggan ORDER BY alamat;
```

HASIL

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

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel pelanggan.

Biasanya mencakup: id_pelanggan, nama_pelanggan, alamat, no_telepon, email, dll.

- **FROM pelanggan**

→ Artinya: data diambil dari tabel bernama **pelanggan**, yang menyimpan informasi pelanggan.

- **ORDER BY alamat**

→ Artinya: **urutkan hasil data berdasarkan kolom alamat** secara **menaik (ascending)** — yaitu **berdasarkan urutan alfabetis nama alamat**.

```
26 -- ambil nama produk, qty dari tb penjualan urutkan berdasarkan qty besar ke kecil 9,8,7...
27 • SELECT nama_produk, qty FROM tr_penjualan ORDER BY qty DESC;
```

HASIL

	nama_produk	qty
►	Kotak Pensil DQLab	5
	Sticky Notes DQLab 500 sheets	5
	Sticky Notes DQLab 500 sheets	4
	Flashdisk DQLab 32 GB	4
	Gift Voucher DQLab 250rb	4
	Buku Planner Agenda DQSQuad	3
	Flashdisk DQLab 32 GB	3
	Buku Planner Agenda DQSQuad	3
	Gift Voucher DQLab 100rb	2
	Flashdisk DQLab 64 GB	2
	Flashdisk DQLab 32 GB	2
	Gantungan Kunci DQLab	2
	Flash disk DQLab 32 GB	1
	Tas Travel Organizer DQLab	1
	Kotak Pensil DQLab	1

PENJELASAN

- **SELECT nama_produk, qty**

→ Artinya: ambil dan tampilkan dua kolom dari tabel:

- nama_produk: nama dari produk yang dijual.
- qty: jumlah unit produk yang terjual.

- **FROM tr_penjualan**

→ Artinya: data diambil dari tabel **tr_penjualan**, yaitu tabel yang menyimpan data transaksi penjualan.

- **ORDER BY qty DESC**

→ Artinya:

- **Urutkan hasil berdasarkan kolom qty secara menurun (Descending).**
- Artinya, produk dengan **jumlah penjualan terbanyak** akan muncul **di atas**, dan yang paling sedikit di **bawah**.

```
29  -- ambil nama produk, qty dari tb penjualan urutkan berdasarkan qty besar ke kecil (huruf z,y,x...) dan (5.4.3...)
30  •  SELECT nama_produk, qty FROM tr_penjualan ORDER BY qty DESC, nama_produk DESC;
--
```

HASIL

	nama_produk	qty
►	Sticky Notes DQLab 500 sheets	5
	Kotak Pensil DQLab	5
	Sticky Notes DQLab 500 sheets	4
	Gift Voucher DQLab 250rb	4
	Flashdisk DQLab 32 GB	4
	Flashdisk DQLab 32 GB	3
	Buku Planner Agenda DQSQuad	3
	Buku Planner Agenda DQSQuad	3
	Gift Voucher DQLab 100rb	2
	Gantungan Kunci DQLab	2
	Flashdisk DQLab 64 GB	2
	Flashdisk DQLab 32 GB	2
	Tas Travel Organizer DQLab	1
	Kotak Pensil DQLab	1
	Flash disk DQLab 32 GB	1

PENJELASAN

- **SELECT nama_produk, qty**

→ Artinya: ambil dan tampilkan kolom:

- nama_produk: nama produk yang dijual.
- qty: jumlah produk yang terjual.

- **FROM tr_penjualan**

→ Artinya: data diambil dari tabel **tr_penjualan**, yang biasanya berisi catatan penjualan produk.

- **ORDER BY qty DESC, nama_produk DESC**

→ Artinya:

- **Urutkan berdasarkan qty dari terbesar ke terkecil** (DESC = descending).
- Jika ada beberapa produk dengan jumlah qty yang **sama**, maka urutkan lagi berdasarkan **nama_produk dari Z ke A** (juga DESC).

```
33 -- tampilkan semua kolom dari tabel tr_penjualan dengan mengurutkan qty secara ASC dan tgl_transaksi secara DESC
34 • SELECT * FROM tr_penjualan ORDER BY qty ASC, tgl_transaksi DESC;
35
```

HASIL

	kode_urut_transaksi	kode_transaksi	kode_pelanggan	tgl_transaksi	no_urut	kode_produk	nama_produk	qty	harga	diskon_persen
►	12	tr-005	dqlabcust00	2020-06-12 00:00:00	2	prod-01	Kotak Pensil DQLab	1	62500	5
	2	tr-001	dqlabcust07	2020-05-01 00:00:00	2	prod-03	Flash disk DQLab 32 GB	1	100000	25
	7	tr-002	dqlabcust00	2020-05-01 00:00:00	3	prod-07	Tas Travel Organizer DQLab	1	48000	0
	15	tr-006	dqlabcust02	2020-06-19 00:00:00	2	prod-08	Gantungan Kunci DQLab	2	15800	0
	13	tr-005	dqlabcust00	2020-06-12 00:00:00	3	prod-04	Flashdisk DQLab 32 GB	2	40000	0
	8	tr-003	dqlabcust03	2020-05-03 00:00:00	1	prod-02	Flashdisk DQLab 64 GB	2	55000	12.5
	5	tr-002	dqlabcust00	2020-05-01 00:00:00	1	prod-03	Gift Voucher DQLab 100rb	2	100000	0
	11	tr-005	dqlabcust00	2020-06-12 00:00:00	1	prod-09	Buku Planner Agenda DQSQuad	3	92000	0
	3	tr-001	dqlabcust07	2020-05-01 00:00:00	3	prod-09	Buku Planner Agenda DQSQuad	3	92000	0
	4	tr-001	dqlabcust07	2020-05-01 00:00:00	4	prod-04	Flashdisk DQLab 32 GB	3	40000	0
	14	tr-006	dqlabcust02	2020-06-19 00:00:00	1	prod-05	Gift Voucher DQLab 250rb	4	250000	0
	10	tr-004	dqlabcust03	2020-05-03 00:00:00	2	prod-04	Flashdisk DQLab 32 GB	4	40000	0
	6	tr-002	dqlabcust00	2020-05-01 00:00:00	2	prod-10	Sticky Notes DQLab 500 sheets	4	55000	0
	9	tr-004	dqlabcust03	2020-05-03 00:00:00	1	prod-10	Sticky Notes DQLab 500 sheets	5	55000	0
	1	tr-001	dqlabcust07	2020-05-01 00:00:00	1	prod-01	Kotak Pensil DQLab	5	62500	0

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel `tr_penjualan`.

Kolom-kolom itu bisa mencakup: `id_transaksi`, `nama_produk`, `qty`, `harga`, `tgl_transaksi`, dll.

- **FROM tr_penjualan**

→ Artinya: data diambil dari tabel `tr_penjualan`, yang berisi catatan transaksi penjualan.

- **ORDER BY qty ASC, tgl_transaksi DESC**

→ Artinya:

- Urutkan data berdasarkan kolom `qty` secara **menaik (ASC)** — dari jumlah penjualan terkecil ke terbesar.
- Jika ada beberapa baris dengan `qty` yang **sama**, maka urutkan baris-baris itu berdasarkan `tgl_transaksi` secara **menurun (DESC)** — dari **tanggal terbaru ke terlama**.

```
36 -- menampilkan semua kolom pelanggan dengan mengurutkan nama dari besar ke kecil (Z ke A)
37 • SELECT * FROM pelanggan ORDER BY nama_pelanggan DESC;
```

HASIL

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

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel `pelanggan`.

Kolom ini biasanya mencakup:

- `id_pelanggan`
- `nama_pelanggan`
- `alamat`
- `no_telepon`
- dan kolom lain yang berkaitan dengan data pelanggan.

- **FROM pelanggan**

→ Artinya: ambil data dari tabel **pelanggan**, yaitu tabel yang berisi daftar pelanggan.

- **ORDER BY nama_pelanggan DESC**

→ Artinya:

- Urutkan hasil berdasarkan kolom **nama_pelanggan** secara **menurun (descending)**.
- Dengan kata lain, **nama pelanggan akan ditampilkan dari Z ke A**.

```
39 -- menampilkan semua kolom ms pelanggan dengan mengurutkan alamat dari huruf besar ke kecil (Z ke A)
40 • SELECT * FROM pelanggan ORDER BY alamat DESC;
```

HASIL

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

PENJELASAN

- **SELECT ***

→ Artinya: ambil **semua kolom** dari tabel **pelanggan**.

Kolom-kolom ini bisa mencakup:

- id_pelanggan
- nama_pelanggan
- alamat
- no_telepon
- dan lainnya, tergantung struktur tabel.

- **FROM pelanggan**

→ Artinya: data diambil dari tabel **pelanggan**, yaitu daftar semua pelanggan.

- **ORDER BY alamat DESC**

→ Artinya:

- **Urutkan data berdasarkan kolom alamat secara menurun (descending)**.
- Data dengan alamat yang dimulai huruf **terakhir di alfabet (Z, Y, X, dst)** akan muncul **lebih dulu**.
- Urutan berdasarkan **abjad secara terbalik**.


```

43 -- ambil nama produk, qty, harga, untuk menghitung total harga perproduk dengan harga di kali qty
44 -- dan memberikan nama sementara dari harga di kali qty menjadi total dan mengurutkannya dari besar ke kecil
45 • SELECT nama_produk, harga, qty, harga*qty as total
46 FROM tr_penjualan
47 ORDER BY total DESC;

```

HASIL

	nama_produk	harga	qty	total
▶	Gift Voucher DQLab 250rb	250000	4	1000000
	Kotak Pensil DQLab	62500	5	312500
	Buku Planner Agenda DQSquad	92000	3	276000
	Buku Planner Agenda DQSquad	92000	3	276000
	Sticky Notes DQLab 500 sheets	55000	5	275000
	Sticky Notes DQLab 500 sheets	55000	4	220000
	Gift Voucher DQLab 100rb	100000	2	200000
	Flashdisk DQLab 32 GB	40000	4	160000
	Flashdisk DQLab 32 GB	40000	3	120000
	Flashdisk DQLab 64 GB	55000	2	110000
	Flash disk DQLab 32 GB	100000	1	100000
	Flashdisk DQLab 32 GB	40000	2	80000
	Kotak Pensil DQLab	62500	1	62500
	Tas Travel Organizer DQLab	48000	1	48000
	Gantungan Kunci DQLab	15800	2	31600

PENJELASAN

1. **SELECT nama_produk, harga, qty, harga*qty AS total**
 - o Memilih kolom nama_produk (nama produk).
 - o Memilih kolom harga (harga per unit produk).
 - o Memilih kolom qty (jumlah produk yang terjual).
 - o Membuat kolom baru bernama total yang merupakan hasil perkalian antara harga dan qty (total nilai penjualan per produk).
2. **FROM tr_penjualan**
 - o Mengambil data dari tabel tr_penjualan, yaitu tabel berisi transaksi penjualan produk.
3. **ORDER BY total DESC**
 - o Mengurutkan hasil berdasarkan kolom total dari nilai terbesar ke terkecil (DESC = descending).
 - o Produk dengan nilai penjualan terbesar akan muncul paling atas.

```

50 -- Ambil data produk, harga, jumlah, dan diskon persen kemudian menghitung harga di kali qty
51 -- dengan mengganti nama menjadi total harga dan total_harga dikurangi diskon (diskon_persen % dari total_harga)
52 -- mengganti nama menjadi total bayar kemudian mengurutkan total bayar dari yang besar ke kecil
53 • SELECT nama_produk, harga, qty, diskon_persen,
54 (harga * qty) AS total_harga, (harga * qty) - ((diskon_persen / 100) * (harga * qty)) AS total_bayar
55 FROM tr_penjualan
56 ORDER BY total_bayar DESC;

```

HASIL

	nama_produk	harga	qty	diskon_persen	total_harga	total_bayar
▶	Gift Voucher DQLab 250rb	250000	4	0	1000000	1000000
	Kotak Pensil DQLab	62500	5	0	312500	312500
	Buku Planner Agenda DQSQuad	92000	3	0	276000	276000
	Buku Planner Agenda DQSQuad	92000	3	0	276000	276000
	Sticky Notes DQLab 500 sheets	55000	5	0	275000	275000
	Sticky Notes DQLab 500 sheets	55000	4	0	220000	220000
	Gift Voucher DQLab 100rb	100000	2	0	200000	200000
	Flashdisk DQLab 32 GB	40000	4	0	160000	160000
	Flashdisk DQLab 32 GB	40000	3	0	120000	120000
	Flashdisk DQLab 64 GB	55000	2	12.5	110000	96250
	Flashdisk DQLab 32 GB	40000	2	0	80000	80000
	Flash disk DQLab 32 GB	100000	1	25	100000	75000
	Kotak Pensil DQLab	62500	1	5	62500	59375
	Tas Travel Organizer DQLab	48000	1	0	48000	48000
	Gantungan Kunci DQLab	15800	2	0	31600	31600

PENJELASAN

- **SELECT nama_produk, harga, qty, diskon_persen, ...**
 - Memilih kolom-kolom:
 - nama_produk: nama produk yang dijual.
 - harga: harga satuan produk.
 - qty: jumlah produk yang dibeli.
 - diskon_persen: persentase diskon yang diberikan pada produk tersebut.
- **(harga * qty) AS total_harga**
 - Menghitung total harga sebelum diskon, yaitu harga satuan dikalikan jumlah produk.
 - Memberi nama kolom hasil perhitungan ini dengan alias total_harga.
- **(harga * qty) - ((diskon_persen / 100) * (harga * qty)) AS total_bayar**
 - Menghitung total yang harus dibayar setelah diskon.
 - Cara hitung: total harga dikurangi jumlah diskon (diskon_persen dalam persen dibagi 100, dikalikan total harga).
 - Hasilnya diberi nama alias total_bayar.
- **FROM tr_penjualan**
 - Mengambil data dari tabel transaksi penjualan tr_penjualan.
- **ORDER BY total_bayar DESC**
 - Mengurutkan hasil berdasarkan nilai total_bayar dari yang terbesar ke terkecil (descending).
 - Produk dengan total bayar terbesar akan tampil paling atas.

```

59 -- Menyaring hanya baris data yang memiliki diskon lebih dari 0%
60 • SELECT nama_produk, harga, qty, diskon_persen,
61 (harga * qty) AS total_harga, (harga * qty) - ((diskon_persen / 100) * (harga * qty)) AS total_bayar
62 FROM tr_penjualan
63 WHERE diskon_persen > 0
64 ORDER BY total_bayar DESC;

```

HASIL

	nama_produk	harga	qty	diskon_persen	total_harga	total_bayar
▶	Flashdisk DQLab 64 GB	55000	2	12.5	110000	96250
	Flash disk DQLab 32 GB	100000	1	25	100000	75000
	Kotak Pensil DQLab	62500	1	5	62500	59375

PENJELASAN

- **SELECT nama_produk, harga, qty, diskon_persen, ...**
 - Memilih kolom-kolom berikut:
 - nama_produk: nama produk yang dijual.
 - harga: harga per satuan produk.
 - qty: jumlah produk yang dibeli.
 - diskon_persen: persentase diskon yang diberikan.
- **(harga * qty) AS total_harga**
 - Menghitung total harga sebelum diskon, yaitu harga dikalikan jumlah.
- **(harga * qty) - ((diskon_persen / 100) * (harga * qty)) AS total_bayar**
 - Menghitung jumlah yang harus dibayar setelah diskon (total harga dikurangi diskon).
- **FROM tr_penjualan**
 - Mengambil data dari tabel tr_penjualan.
- **WHERE diskon_persen > 0**
 - **Filter data hanya menampilkan transaksi yang memiliki diskon lebih dari 0%.**
 - Artinya, hanya produk yang mendapat diskon yang diikuti dalam hasil.
- **ORDER BY total_bayar DESC**
 - Mengurutkan hasil berdasarkan total_bayar dari yang terbesar ke terkecil.

```

67 -- ambil nama produk, qty, dari tb penjualan dan menyaring yang nama produknya berawalan f urut qty (besar ke kecil)
68 • SELECT nama_produk, qty
69 FROM tr_penjualan
70 WHERE nama_produk LIKE 'F%'
71 ORDER BY qty DESC;

```

HASIL

	nama_produk	qty
▶	Flashdisk DQLab 32 GB	4
	Flashdisk DQLab 32 GB	3
	Flashdisk DQLab 64 GB	2
	Flashdisk DQLab 32 GB	2
	Flash disk DQLab 32 GB	1

PENJELASAN

1. **SELECT nama_produk, qty**
 - o Memilih kolom nama_produk dan qty dari tabel tr_penjualan.
 - o nama_produk: nama produk yang terjual.
 - o qty: jumlah produk yang terjual.
2. **FROM tr_penjualan**
 - o Data diambil dari tabel tr_penjualan, tabel berisi transaksi penjualan.
3. **WHERE nama_produk LIKE 'F%'**
 - o **Filter** hanya mengambil baris data dimana nama_produk dimulai dengan huruf F.
 - o 'F%' artinya:
 - F adalah huruf pertama,
 - % artinya "diikuti oleh karakter apa saja sebanyak nol atau lebih".
 - o Contoh produk yang sesuai: "Flashdisk", "Fridge", "Fan", dll.
4. **ORDER BY qty DESC**
 - o Mengurutkan hasil berdasarkan kolom qty secara **menurun** (dari jumlah terbesar ke terkecil).
 - o Produk dengan penjualan terbanyak yang namanya mulai dengan "F" akan muncul di atas.

```

74 -- untuk mengambil semua kolom dari tabel tr penjualan
75 -- dan menyaring nilai diskon lebih dari nol dan mengurutkan nilai harga dari besar ke kecil
76 • SELECT *
77 FROM tr_penjualan
78 WHERE diskon_persen > 0
79 ORDER BY harga DESC;
```

HASIL

	kode_urut_transaksi	kode_transaksi	kode_pelanggan	tgl_transaksi	no_urut	kode_produk	nama_produk	qty	harga	diskon_persen
▶	2	tr-001	dqlabcust07	2020-05-01 00:00:00	2	prod-03	Flash disk DQLab 32 GB	1	100000	25
	12	tr-005	dqlabcust00	2020-06-12 00:00:00	2	prod-01	Kotak Pensil DQLab	1	62500	5
	8	tr-003	dqlabcust03	2020-05-03 00:00:00	1	prod-02	Flashdisk DQLab 64 GB	2	55000	12.5

PENJELASAN

- **SELECT ***
 - Memilih **semua kolom** dari tabel tr_penjualan.
 - Artinya, data lengkap setiap transaksi penjualan akan ditampilkan.
- **FROM tr_penjualan**
 - Data diambil dari tabel tr_penjualan yang berisi catatan transaksi penjualan produk.

- `WHERE diskon_persen > 0`
 - Filter data hanya menampilkan transaksi yang memiliki diskon lebih dari 0%.
 - Jadi, hanya transaksi dengan diskon yang muncul, transaksi tanpa diskon tidak ditampilkan.
- `ORDER BY harga DESC`
 - Mengurutkan hasil berdasarkan kolom `harga` secara menurun (descending).
 - Transaksi dengan harga satuan produk tertinggi akan muncul paling atas.

```

81 -- mengambil kolom nama produk, qty, dan harga dari tr penjualan dengan menyaring nilai harga
82 -- dari 100000 atau lebih dan mengurutkannya dari besar ke kecil
83 • SELECT nama_produk, qty, harga
84 FROM tr_penjualan
85 WHERE harga >= 100000
86 ORDER BY harga DESC;

```

HASIL

	nama_produk	qty	harga
▶	Gift Voucher DQLab 250rb	4	250000
	Flash disk DQLab 32 GB	1	100000
	Gift Voucher DQLab 100rb	2	100000

PENJELASKAN

- `SELECT nama_produk, qty, harga`
 - Memilih kolom:
 - `nama_produk`: nama produk yang dijual.
 - `qty`: jumlah produk yang terjual.
 - `harga`: harga per satuan produk.
- `FROM tr_penjualan`
 - Mengambil data dari tabel `tr_penjualan` yang berisi transaksi penjualan.
- `WHERE harga >= 100000`
 - Filter hanya menampilkan baris data di mana harga produk **lebih besar atau sama dengan 100.000**.
 - Produk dengan harga di bawah 100.000 tidak akan ditampilkan.
- `ORDER BY harga DESC`
 - Mengurutkan hasil berdasarkan kolom `harga` secara **menurun (descending)**.
 - Produk dengan harga terbesar akan muncul paling atas.

```

88 -- mengambil kolom nama produk, qty dan harga dari kolom tr penjualan dan menyaring harga dari 100000 atau lebih
89 -- dan nama produk yang berawalan dari T, terus mengurutkan namanya dari besar ke kecil
90 • SELECT nama_produk, qty, harga
91 FROM tr_penjualan
92 WHERE harga >= 100000 OR nama_produk LIKE 'T%'
93 ORDER BY diskon_persen DESC;

```

HASIL

	nama_produk	qty	harga
▶	Flash disk DQLab 32 GB	1	100000
	Gift Voucher DQLab 100rb	2	100000
	Tas Travel Organizer DQLab	1	48000
	Gift Voucher DQLab 250rb	4	250000

PENJELASAN

- **SELECT nama_produk, qty, harga**
 - Memilih kolom:
 - nama_produk: nama produk.
 - qty: jumlah produk yang terjual.
 - harga: harga satuan produk.
- **FROM tr_penjualan**
 - Data diambil dari tabel tr_penjualan yang berisi transaksi penjualan.
- **WHERE harga >= 100000 OR nama_produk LIKE 'T%'**
 - Filter data dengan **dua kondisi digabungkan menggunakan OR**:
 - Produk yang **harganya minimal 100.000**
ATAU
 - Produk yang **nama produknya dimulai dengan huruf "T"**.
 - Jika salah satu kondisi terpenuhi, data produk akan diikutkan dalam hasil.
- **ORDER BY diskon_persen DESC**
 - Mengurutkan hasil berdasarkan kolom diskon_persen secara menurun (dari diskon terbesar ke terkecil).
 - Produk dengan diskon terbesar muncul paling atas.

```

96 -- Fungsi agregasi
97 -- di gunakan untuk mengolah beberapa baris/row data untuk menghasilkan nilai baru
98 -- praktek 7
99 -- hitung atau menjumlahkan seluruh baris dari qty dari tabel penjualan
100 • SELECT SUM(qty) FROM tr_penjualan;

```

HASIL

	SUM(qty)
▶	42

PENJELASAN

- **SELECT SUM(qty)**
 - Fungsi SUM() adalah fungsi agregat yang menjumlahkan semua nilai dari kolom qty.
 - Jadi, query ini menghitung **total jumlah produk yang terjual** dengan menjumlahkan semua nilai di kolom qty dari seluruh baris.
- **FROM tr_penjualan**
 - Data diambil dari tabel tr_penjualan, yaitu tabel yang berisi data transaksi penjualan produk.

```

103      -- hitung seluruh baris pada tabel penjualan
104 •    SELECT * FROM tr_penjualan;
105 •    SELECT COUNT(*) FROM tr_penjualan;

```

HASIL

	COUNT(*)
▶	15

PENJELASAN

- COUNT(*) adalah fungsi agregat yang menghitung **jumlah total baris** dalam tabel.
- FROM tr_penjualan berarti menghitung jumlah baris dari tabel tr_penjualan.
- Jadi, query ini mengembalikan **berapa banyak baris data** yang ada dalam tabel tersebut.
- Contohnya, jika ada 100 transaksi, hasil query ini adalah angka 100.

```

108      -- menggunakan 2 fungsi agregasi dalam satu query sql
109 •    SELECT SUM(qty), COUNT(*) FROM tr_penjualan;

```

HASIL

	SUM(qty)	COUNT(*)
▶	42	15

PENJELASAN

1. **SUM(qty)**
 - Fungsi agregat yang menjumlahkan semua nilai di kolom qty.
 - Artinya, menghitung **total jumlah unit produk yang terjual** dari seluruh baris di tabel tr_penjualan.

2. COUNT (*)

- o Fungsi agregat yang menghitung **jumlah total baris** dalam tabel.
- o Ini memberi tahu berapa banyak transaksi atau entri yang ada di tabel tr_penjualan.

3. FROM tr_penjualan

- o Mengambil data dari tabel tr_penjualan.

```
112 -- hitung rata" penjualan, penjualan tertinggi dan terendah dari sekali transaksi
113 • SELECT AVG(qty), MAX(qty), MIN(qty) FROM tr_penjualan;
```

HASIL

	AVG(qty)	MAX(qty)	MIN(qty)
▶	2.8000	5	1

PENJELASAN

• VG(qty)

- Fungsi agregat yang menghitung **rata-rata** nilai dari kolom qty.
- Artinya, berapa rata-rata jumlah produk yang terjual per transaksi.

• MAX(qty)

- Fungsi agregat yang mengambil nilai **terbesar** dari kolom qty.
- Menunjukkan transaksi dengan jumlah produk terjual terbanyak.

• MIN(qty)

- Fungsi agregat yang mengambil nilai **terkecil** dari kolom qty.
- Menunjukkan transaksi dengan jumlah produk terjual paling sedikit.

• FROM tr_penjualan

- Data diambil dari tabel tr_penjualan.

```
116 -- mengkombinasikan fungsi agregasi dengan kolom tertentu (nama produk, kolom tanggal transaksi)
117 • SELECT tgl_transaksi, nama_produk, MAX(qty) FROM tr_penjualan;
```

HASIL

	tgl_transaksi	nama_produk	MAX(qty)
▶	2020-05-01 00:00:00	Kotak Pensil DQLab	5

PENJELASAN

- SELECT digunakan untuk memilih kolom-kolom data dari tabel.
- tgl_transaksi dan nama_produk adalah dua kolom yang ingin diambil nilainya dari tabel tr_penjualan.

- MAX (qty) adalah fungsi agregat yang mengambil nilai maksimum dari kolom qty(quantity/jumlah).
- FROM tr_penjualan berarti data diambil dari tabel bernama tr_penjualan.

```
123      -- tanpa distinct
124 •    SELECT COUNT(nama_produk) FROM tr_penjualan;
```

HASIL

	COUNT(nama_produk)
▶	15

PENJELASAN

- SELECT berarti memilih data dari tabel.
- COUNT (nama_produk) adalah fungsi agregat yang menghitung jumlah baris di mana kolom nama_produk **tidak bernilai NULL**.
- FROM tr_penjualan berarti data diambil dari tabel tr_penjualan.

```
125      -- pakai distinct
126 •    SELECT COUNT(DISTINCT nama_produk) FROM tr_penjualan;
```

HASIL

	COUNT(DISTINCT nama_produk)
▶	10

PENJELASAN

- SELECT digunakan untuk mengambil data dari tabel.
- COUNT (DISTINCT nama_produk) adalah fungsi agregat yang menghitung **jumlah nilai unik (tidak duplikat)** dari kolom nama_produk.
- FROM tr_penjualan artinya data diambil dari tabel tr_penjualan.

```
127      -- pakai distinct dan tanpa distinct
128 •    SELECT COUNT(nama_produk), COUNT(DISTINCT nama_produk) FROM tr_penjualan;
```

HASIL

	COUNT(nama_produk)	COUNT(DISTINCT nama_produk)
▶	15	10

PENJELASAN

- COUNT(nama_produk)
- Menghitung jumlah baris yang memiliki nilai nama_produk tidak NULL.
- Jadi, jika ada 100 baris tapi 5 di antaranya nama_produk= NULL, maka hasilnya 95.
- COUNT(DISTINCT nama_produk)
- Menghitung jumlah nilai unik (tidak duplikat dan tidak NULL) pada kolom nama_produk.
- Jika ada nilai 95 baris, tapi hanya ada 10 produk berbeda (unik), maka hasilnya 10.
- FROM tr_penjualan
- Artinya data tersebut diambil dari tabel tr_penjualan

```
136 -- mengelompokkan nilai data produk
137 • SELECT nama_produk FROM tr_penjualan GROUP BY nama_produk;
```

HASIL

nama_produk
Buku Planner Agenda DQSQuad
Flash disk DQLab 32 GB
Flashdisk DQLab 32 GB
Flashdisk DQLab 64 GB
Gantungan Kunci DQLab
Gift Voucher DQLab 100rb
Gift Voucher DQLab 250rb
Kotak Pensil DQLab
Sticky Notes DQLab 500 sheets
Tas Travel Organizer DQLab

PENJELASAN

- SELECT nama_produk → Meminta kolom nama_produk untuk ditampilkan.
- FROM tr_penjualan → Data diambil dari tabel tr_penjualan.
- GROUP BY nama_produk → Mengelompokkan data berdasarkan nilai unik dari nama_produk.

```
140 -- mengelompokkan nilai data produk dan qty pada tabel penjualan
141 -- pakai
142 • SELECT nama_produk, qty FROM tr_penjualan GROUP BY nama_produk, qty;
```

HASIL

	nama_produk	qty
▶	Buku Planner Agenda DQSQuad	3
	Flash disk DQLab 32 GB	1
	Flashdisk DQLab 32 GB	2
	Flashdisk DQLab 32 GB	3
	Flashdisk DQLab 32 GB	4
	Flashdisk DQLab 64 GB	2
	Gantungan Kunci DQLab	2
	Gift Voucher DQLab 100rb	2
	Gift Voucher DQLab 250rb	4
	Kotak Pensil DQLab	1
	Kotak Pensil DQLab	5
	Sticky Notes DQLab 500 sheets	4
	Sticky Notes DQLab 500 sheets	5
	Tas Travel Organizer DQLab	1

PENJELASAN

- `SELECT nama_produk, qty`: Meminta dua kolom, yaitu `nama_produk` dan `qty(jumlah)`.
- `FROM tr_penjualan`: Data diambil dari tabel `tr_penjualan`.
- `GROUP BY nama_produk, qty`: Mengelompokkan data berdasarkan kombinasi **nama produk** dan **jumlah**.

```
143      -- tidak pakai
144 •     SELECT nama_produk, qty FROM tr_penjualan;
```

HASIL

	nama_produk	qty
▶	Kotak Pensil DQLab	5
	Flash disk DQLab 32 GB	1
	Buku Planner Agenda DQSQuad	3
	Flashdisk DQLab 32 GB	3
	Gift Voucher DQLab 100rb	2
	Sticky Notes DQLab 500 sheets	4
	Tas Travel Organizer DQLab	1
	Flashdisk DQLab 64 GB	2
	Sticky Notes DQLab 500 sheets	5
	Flashdisk DQLab 32 GB	4
	Buku Planner Agenda DQSQuad	3
	Kotak Pensil DQLab	1
	Flashdisk DQLab 32 GB	2
	Gift Voucher DQLab 250rb	4
	Gantungan Kunci DQLab	2

PENJELASAN

- `SELECT nama_produk, qty`: Menginstruksikan database untuk mengambil **dua kolom**, yaitu:
 - `nama_produk` → Nama produk yang dijual

- o qty→ Jumlah (kuantitas) produk yang dijual
- FROM tr_penjualan: Menyatakan bahwa data tersebut diambil dari tabel tr_penjualan.

```

147 -- ambil jumlah qty dari nama produk berdasarkan hasil pengelompokkan nama produk
148 • SELECT nama_produk, SUM(qty) FROM tr_penjualan GROUP BY nama_produk;

```

HASIL

	nama_produk	SUM(qty)
▶	Buku Planner Agenda DQSQuad	6
	Flash disk DQLab 32 GB	1
	Flashdisk DQLab 32 GB	9
	Flashdisk DQLab 64 GB	2
	Gantungan Kunci DQLab	2
	Gift Voucher DQLab 100rb	2
	Gift Voucher DQLab 250rb	4
	Kotak Pensil DQLab	6
	Sticky Notes DQLab 500 sheets	9
	Tas Travel Organizer DQLab	1

PENJELASAN

- SELECT nama_produk, SUM(qty)
 - Pilih dua kolom:
 - nama_produk: Nama produk
 - SUM(qty): Menjumlahkan seluruh nilai kuantitas (qty) dari setiap produk yang sama.
- FROM tr_penjualan
 - Data diambil dari tabel tr_penjualan.
- GROUP BY nama_produk
 - Mengelompokkan data berdasarkan nama_produk, agar fungsi agregat SUM(qty) dapat diterapkan ke setiap kelompok.

```

151 -- ambil jumlah qty dari nama produk berdasarkan hasil pengelompokkan nama produk urutkan dari besar ke kecil
152 • SELECT nama_produk, SUM(qty) FROM tr_penjualan GROUP BY nama_produk ORDER BY SUM(qty) DESC;

```

HASIL

	nama_produk	SUM(qty)
▶	Flashdisk DQLab 32 GB	9
	Sticky Notes DQLab 500 sheets	9
	Kotak Pensil DQLab	6
	Buku Planner Agenda DQSQuad	6
	Gift Voucher DQLab 250rb	4
	Gift Voucher DQLab 100rb	2
	Flashdisk DQLab 64 GB	2
	Gantungan Kunci DQLab	2
	Flash disk DQLab 32 GB	1
	Tas Travel Organizer DQLab	1

PENJE • `SELECT nama_produk, SUM(qty)`

→ Menampilkan dua kolom:

- `nama_produk`: Nama produk.
- `SUM(qty)`: Total jumlah (kuantitas) produk tersebut yang dijual.

• `FROM tr_penjualan`

→ Data diambil dari tabel `tr_penjualan`.

• `GROUP BY nama_produk`

→ Mengelompokkan data berdasarkan `nama_produk`, sehingga fungsi agregat `SUM(qty)` dapat dihitung untuk masing-masing produk.

• `ORDER BY SUM(qty) DESC`

→ Mengurutkan hasil berdasarkan total penjualan (`SUM(qty)`) dari yang terbesar ke yang terkecil (`DESC` = descending).

```
158 -- ambil jumlah qty dari nama produk yang jumlah qty > dari 2 berdasarkan hasil pengelompokan nama produk
159 • SELECT nama_produk, SUM(qty)
160 FROM tr_penjualan
161 GROUP BY nama_produk
162 HAVING SUM(qty) > 2
163 ORDER BY SUM(qty) DESC;
```

HASIL

	nama_produk	SUM(qty)
▶	Flashdisk DQLab 32 GB	9
	Sticky Notes DQLab 500 sheets	9
	Kotak Pensil DQLab	6
	Buku Planner Agenda DQSQuad	6
	Gift Voucher DQLab 250rb	4

PENJELASAN

1. `SELECT nama_produk, SUM(qty)`

→ Menampilkan dua kolom:

- `nama_produk`: Nama produk
- `SUM(qty)`: Total penjualan (jumlah/kuantitas) dari masing-masing produk

2. `FROM tr_penjualan`

→ Mengambil data dari tabel `tr_penjualan`.

3. `GROUP BY nama_produk`

→ Mengelompokkan data berdasarkan nama produk agar `SUM(qty)` bisa dihitung untuk masing-masing produk.

4. `HAVING SUM(qty) > 2`

→ Menyaring hasil kelompok: hanya produk dengan total penjualan **lebih dari 2 unit** yang ditampilkan.

`HAVING` digunakan untuk **menyaring hasil setelah pengelompokan** (berbeda dari `WHERE`, yang menyaring sebelum `GROUP BY`).

5. ORDER BY SUM(qty) DESC

→ Mengurutkan hasil berdasarkan total penjualan dari yang **terbanyak ke yang paling sedikit** .

```
167 -- melihat daftar nama produk yang total jumlah produk terjual perkelompok nama produk adalah di atas nilai 4
168 • SELECT nama_produk, SUM(qty) AS total_terjual
169 FROM tr_penjualan
170 GROUP BY nama_produk
171 HAVING SUM(qty) > 4
172 ORDER BY total_terjual DESC;
```

HASIL

	nama_produk	total_terjual
▶	Flashdisk DQLab 32 GB	9
	Sticky Notes DQLab 500 sheets	9
	Kotak Pensil DQLab	6
	Buku Planner Agenda DQSQuad	6

PENJELASAN

- SELECT nama_produk, SUM(qty) → Memilih nama produk dan jumlah kuantitas (qty) yang dijumlahkan.
- FROM tr_penjualan_dqlab → Dari tabel transaksi penjualan.
- GROUP BY nama_produk → Mengelompokkan data berdasarkan nama produk.
- HAVING SUM(qty) > 4 → Menyaring hanya produk yang total jumlah terjual lebih dari 4 unit.

```
174 -- melihat daftar nama produk yang total jumlah produk terjual perkelompok nama produk sama dengan 9
175 • SELECT nama_produk, SUM(qty) AS total_terjual
176 FROM tr_penjualan
177 GROUP BY nama_produk
178 HAVING SUM(qty) = 9
179 ORDER BY total_terjual DESC;
```

HASIL

	nama_produk	total_terjual
▶	Flashdisk DQLab 32 GB	9
	Sticky Notes DQLab 500 sheets	9

PENJELASAN

- Hampir sama dengan query pertama, hanya berbeda di filter HAVING.
- HAVING SUM(qty) = 9 → Hanya menampilkan produk yang jumlah total terjualnya **tepat 9 unit** .

```

181 -- melihat daftar kelompok nama produk dan total nilai penjualan (harga dikalikan jumlah dikurangi diskon )
182 -- dengan dan urutkan berdasarkan nilai penjualan terbesar
183 • SELECT nama_produk, SUM(harga * qty - (diskon_persen / 100) * harga * qty) AS total_penjualan
184 FROM tr_penjualan
185 GROUP BY nama_produk
186 ORDER BY total_penjualan DESC;

```

HASIL

	nama_produk	total_penjualan
►	Gift Voucher DQLab 250rb	1000000
	Buku Planner Agenda DQSquad	552000
	Sticky Notes DQLab 500 sheets	495000
	Kotak Pensil DQLab	371875
	Flashdisk DQLab 32 GB	360000
	Gift Voucher DQLab 100rb	200000
	Flashdisk DQLab 64 GB	96250
	Flash disk DQLab 32 GB	75000
	Tas Travel Organizer DQLab	48000
	Gantungan Kunci DQLab	31600

- $\text{harga} * \text{qty}$ → Total nilai penjualan kotor (tanpa diskon) per transaksi.
- $(\text{diskon_persen} / 100) * (\text{harga} * \text{qty})$ → Nilai diskon dalam rupiah.
- $\text{harga} * \text{qty} - \text{diskon}$ → Nilai penjualan bersih setelah diskon.
- $\text{SUM}(\dots)$ → Menjumlahkan nilai penjualan bersih untuk setiap produk.
- $\text{GROUP BY nama_produk}$ → Dikelompokkan berdasarkan produk.
- $\text{ORDER BY total_penjualan DESC}$ → Diurutkan dari penjualan tertinggi ke terendah.