Nama : SELKI DEBEI USYAMI

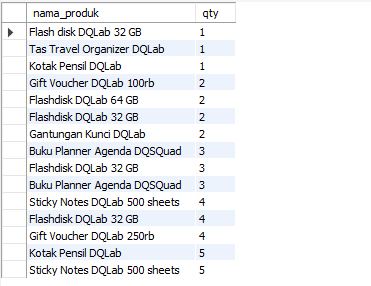
Nim : 23241074

Kelas : PTI 4/B

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



**HASIL**



**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**

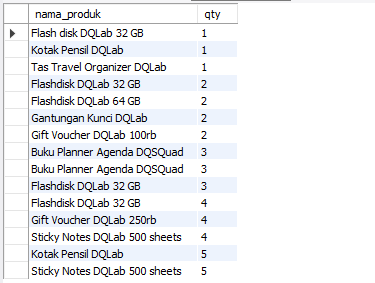
Artinya: data diambil dari tabel yang bernama **tr\_penjualan**, yang bisa diartikan sebagai "tabel transaksi penjualan".

* **ORDER BY qty**

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



**HASIL**



**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.

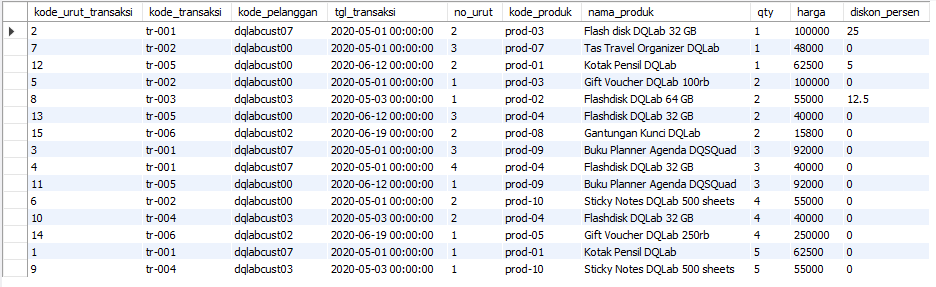
* **ORDER BY qty, nama\_produk**

→ Artinya:

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



**HASIL**



**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:

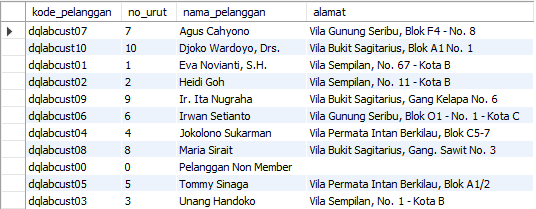
* + **Urutkan data berdasarkanqty** terlebih dahulu dari **terkecil ke terbesar** (default

ASC).

Jika ada baris-baris dengan nilai **qty** yang sama, maka diurutkan**berdasarkantgl\_transaksi** tanggal **terlama ke terbaru.**



**HASIL**



**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.

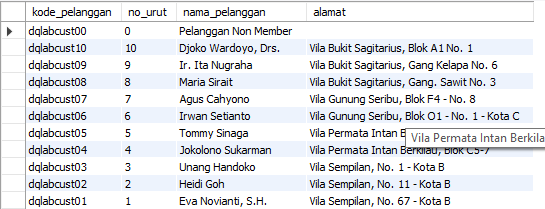
* **FROM pelanggan**

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

* **ORDER BY nama\_pelanggan**

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

 **HASIL**



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**.



**HASIL**



**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**

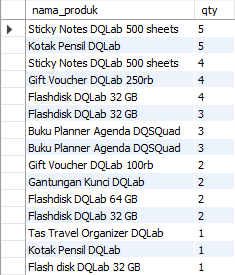
→ Artinya:

# Urutkan hasil berdasarkan kolom qty secara menurun (Descending).

* + Artinya, produk dengan **jumlah penjualan terbanyak** akan muncul **di atas**, dan yang paling sedikit di **bawah**.



**HASIL**



**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.

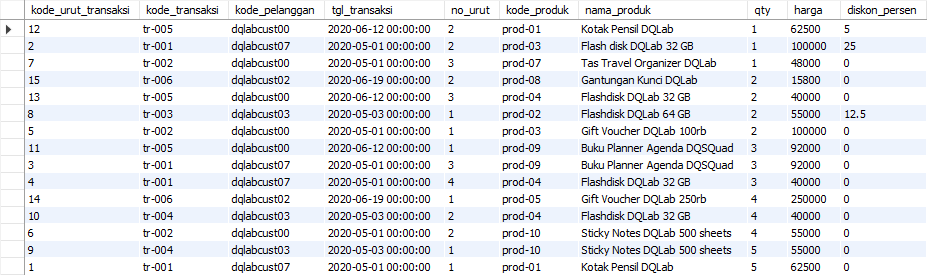
* **ORDER BY qty DESC, nama\_produk DESC**

→ 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**).



**HASIL**



**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.

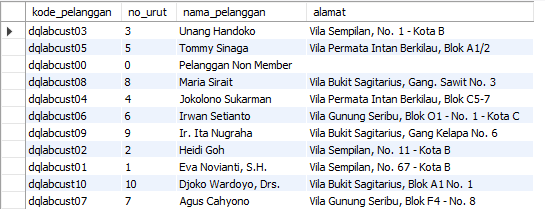
* **ORDER BY qty ASC, tgl\_transaksi DESC**

→ 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**.



HASIL



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**.



HASIL



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.

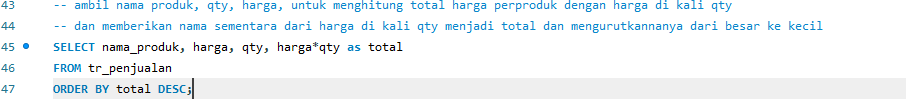
* **ORDER BY alamat DESC**

→ Artinya:

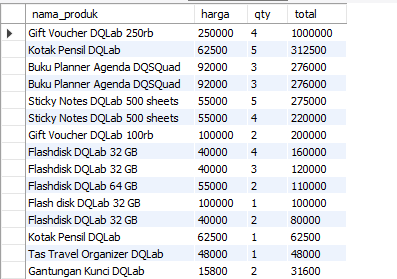
# Urutkan data berdasarkan kolom alamat secara menurun (descending).

* + Data dengan alamat yang dimulai huruf **terakhir di alfabet (Z, Y, X, dst)** akan muncul **lebih dulu**.

Urutan berdasarkan **abjad secara terbalik**.



**HASIL**

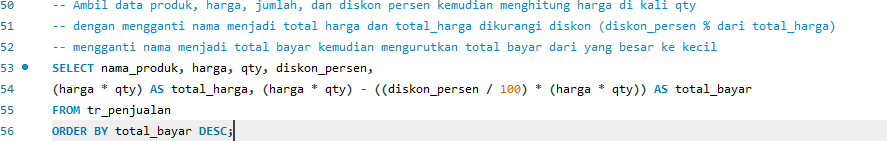


**PENJELASAN**

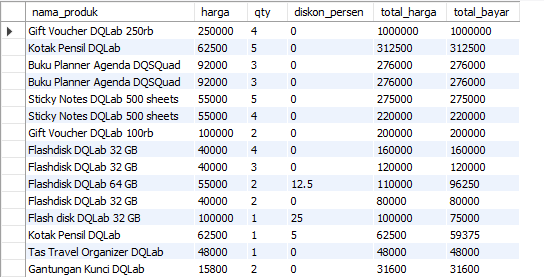
1. **SELECT nama\_produk, harga, qty, harga\*qty AS total**
   * Memilih kolom **nama\_produk** (nama produk).
   * Memilih kolom **harga** (harga per unit produk).
   * Memilih kolom **qty** (jumlah produk yang terjual).
   * Membuat kolom baru bernama **total** yang merupakan hasil perkalian antara

**harga** dan **qty** (total nilai penjualan per produk).

1. **FROM tr\_penjualan**
   * Mengambil data dari tabel **tr\_penjualan,** yaitu tabel berisi transaksi penjualan produk.
2. **ORDER BY total DESC**
   * Mengurutkan hasil berdasarkan kolom **total** dari nilai terbesar ke terkecil (**DESC** = descending).
   * Produk dengan nilai penjualan terbesar akan muncul paling atas.

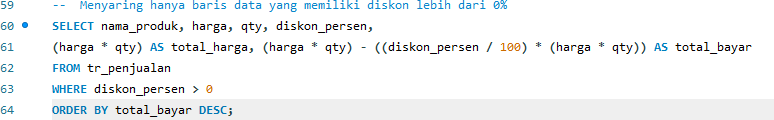


**HASIL**

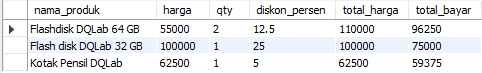


**PENJELASAN**

* **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.



**HASIL**

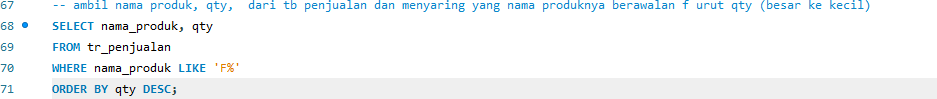


**PENJELASAN**

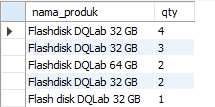
* **SELECT nama\_produk, harga, qty, diskon\_persen, ...**
  + Memilih kolom-kolom berikut:
    - **nama\_produk:** nama produk yang dijual.
    - **harga**: harga per satuan produk.
    - **qty**: jumlah produk yang dibeli.
    - **diskon\_persen**: persentase diskon yang diberikan.
* **(harga \* qty) 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 diikutkan dalam hasil.
* **ORDER BY total\_bayar DESC**
  + Mengurutkan hasil berdasarkan total\_bayar dari yang terbesar ke terkecil.

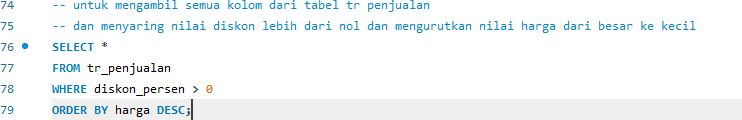


**HASIL**

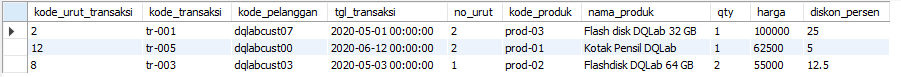


**PENJELASAN**

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

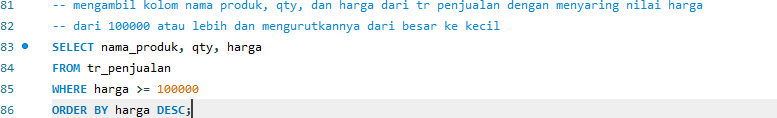


**HASIL**

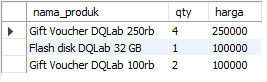


**PENJELASAN**

* **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.

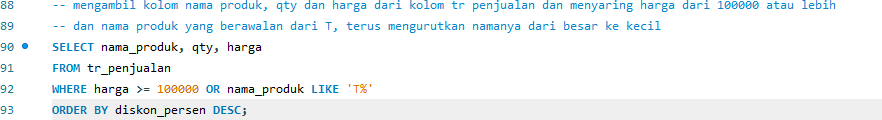


**HASIL**

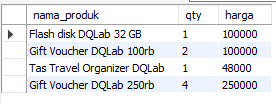


**PENJELASKAN**

* **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.

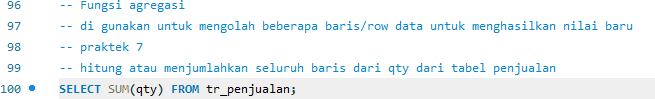


**HASIL**



**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.



**HASIL**



**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.



**HASIL**



**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.



**HASIL**



**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(\*)**
   * Fungsi agregat yang menghitung **jumlah total baris** dalam tabel.
   * Ini memberi tahu berapa banyak transaksi atau entri yang ada di tabel

**tr\_penjualan.**

1. **FROM tr\_penjualan**
   * Mengambil data dari tabel **tr\_penjualan**



**HASIL**



**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.**

** HASIL**



**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.**



**HASIL**



**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**.



**HASIL**



**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**.



**HASIL**

****

**PENJELASAN**

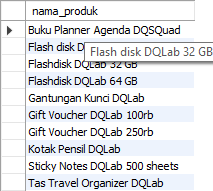
* **COUNT(nama\_produk):** Menghitung jumlah baris **yang memiliki nilai nama\_produktidak NULL** .
* Jadi, jika ada 100 baris tapi 5 di antaranya **nama\_produk** = NULL, maka hasilnya **95** .
* **COUNT(DISTINCT nama\_produk):** Menghitung jumlah **nilai unik (tidak duplikat dan tidak NULL)** pada 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**



HASIL



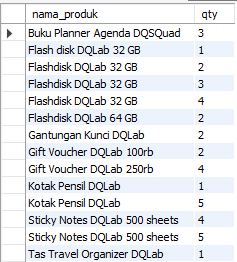
PENJELASAN

* + **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.**



**HASIL**



**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** .



**HASIL**

****

**PENJELASAN**

* + **SELECT nama\_produk, qty:** Menginstruksikan database untuk mengambil **dua kolom** , yaitu:

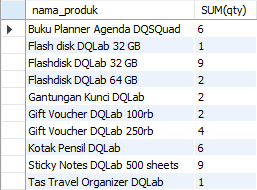
**nama\_produk** → Nama produk yang dijual

* **qty** → Jumlah (kuantitas) produk yang dijual
  + **FROM tr\_penjualan**: Menytakan bahwa data tersebut diambil dari tabel

tr\_penjualan.



**HASIL**

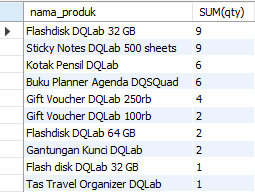
****

**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.



HASIL



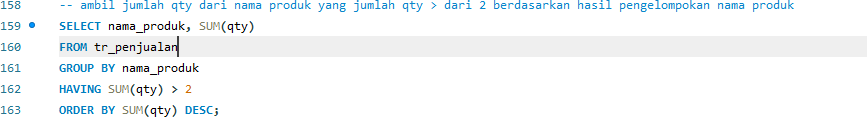
PENJELASAN

* **SELECT nama\_produk, SUM(qty)**Menampilkan dua kolom:

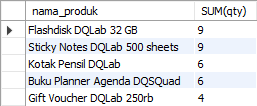
**nama\_produk**: Nama produk.

**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**).



**HASIL**

****

**PENJELASAN**

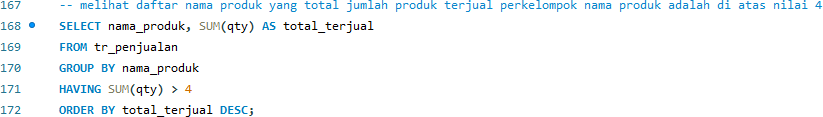
1. **SELECT nama\_produk, SUM(qty):** Menampilkan dua kolom:
   * + **nama\_produk**: Nama produk
     + **SUM(qty):** Total penjualan (jumlah/kuantitas) dari masing-masing produk
2. **FROM tr\_penjualan :** Mengambil data dari tabel **tr\_penjualan.**
3. **GROUP BY nama\_produk :** Mengelompokkan data 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.

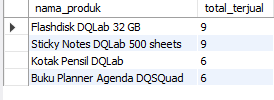
**HAVING** digunakan untuk **menyaring hasil setelah pengelompokan** (berbeda dari

**WHERE**, yang menyaring sebelum **GROUP BY**).

1. **ORDER BY SUM(qty) DESC :** Mengurutkan hasil berdasarkan total penjualan dari yang **terbanyak ke yang paling sedikit** .

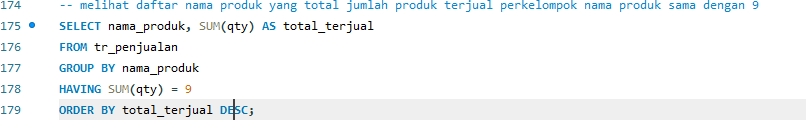


HASIL



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.

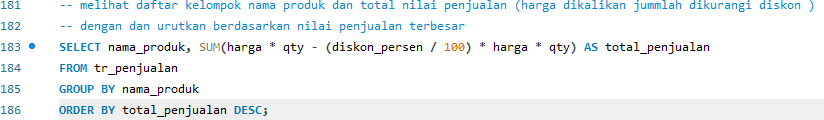


**HASIL**

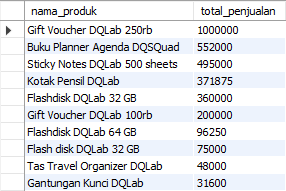
****

**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** .



**HASIL**



PENJELASAN

* **harga \* qty** → Total nilai penjualan kotor (tanpa diskon) per transaksi.
* (**diskon\_persen / 100) \* (harga \* qty**)→ Nilai diskon dalam rupiah.
* **harga \* qty – diskon** → Nilai penjualan bersih setelah diskon.
* **SUM(...)→** Menjumlahkan nilai penjualan bersih untuk setiap produk.
* **GROUP BY nama\_produk** → Dikelompokkan berdasarkan produk.
* **ORDER BY total\_penjualan DESC** → Diurutkan dari penjualan tertinggi ke terendah.