

# Soal & Template Jawaban

Task 5

Nama : Ivan Yudha Putranto H.

# Petunjuk

**Silahkan merujuk pada Data Source Task 5 yang telah disediakan untuk mengerjakan soal soal di bawah ini**

Pada bagian data analytics, terdiri dari 4 soal dengan use case & tabel yang sama. Bayangkan kamu memiliki database erp yang terdiri dari 3 tabel: penjualan, pelanggan, barang. Tabel tersebut akan dibuat menjadi sebuah datamart yang nantinya digunakan untuk visualisasi.

# Query

## Soal 1\*:

Dari 2 query ini, mana yang bekerja lebih baik? Jelaskan mengapa.

(a) `SELECT * FROM pelanggan WHERE SUBSTR(alamat, 1, 3) = Mat;`

(b) `SELECT * FROM pelanggan WHERE alamat LIKE 'Mat%'`

*\*disclaimer: soal ini tidak terkait dengan data source*

Jawaban : (b)

Alasan : Query kedua lebih efisien secara performa daripada query pertama. Menggunakan fungsi **SUBSTR()** dalam query pertama akan memaksa basis data untuk menghitung substring untuk setiap baris dalam tabel sebelum membandingkannya dengan 'Mat'. Ini dapat memperlambat kueri, terutama jika tabel memiliki banyak baris.

# Query

## Soal 2 \*:

Anggap kita memiliki tabel pelanggan dengan kolom: id, nama, tanggal\_lahir, alamat. Bagaimana cara yang lebih tepat dalam menulis query untuk mendapatkan data pelanggan yang tanggal\_lahir nya ada di antara 2000-01-01 sampai 2008-12-31? Pilihlah salah satu jawaban dan berikan alasannya.

- (a) `SELECT * FROM pelanggan WHERE tanggal_lahir >= '2000-01-01' AND tanggal_lahir <= '2008-12-31'`
- (b) `SELECT * FROM pelanggan WHERE tanggal_lahir BETWEEN '2000-01-01' AND '2008-12-31'`

*\*disclaimer: soal ini tidak terkait dengan data source*

Jawaban : (b)

Alasan : Dalam banyak sistem basis data, penggunaan **BETWEEN** dapat mengoptimalkan pencarian, terutama jika ada indeks pada kolom tanggal\_lahir. Ini dapat meningkatkan kinerja eksekusi query.

## Soal 3: Menentukan Primary Key

### A. Tugas

Tentukan primary key dari table penjualan. jelaskan alasannya

- B. Jawaban & Penjelasan : id\_invoice, Primary Key harus unik dan tidak boleh berubah sepanjang siklus hidup data dalam table. 'id\_invoice' adalah nomor invoice yang biasanya bersifat unik untuk setiap transaksi penjualan dan tidak akan berubah setelah dibuat. Ini memenuhi syarat untuk menjadi Primary Key.

# Soal 4: Design Datamart

## A. Tugas

Buatlah design datamart (Terdiri dari tabel base, dan tabel aggregate). Upload file query dalam gdrive mu (pastikan dapat diakses public). Lalu masukkan linknya di tabel di bawah, dan cantumkan juga screenshoot query nya (jika lebih dari 1 file, maka masing masing file di-screenshoot)  
Silahkan tambah halaman jika dibutuhkan

## B. Jawaban :

No	Nama File	Link
1	Saya membuat tablenya menggunakan export data dari dataset yang dipecah menjadi satu sheet satu tabel	<a href="https://drive.google.com/file/d/1GjJjyavtQfBFCqJKT34o58zNg-O_2pav/view?usp=share_link">https://drive.google.com/file/d/1GjJjyavtQfBFCqJKT34o58zNg-O_2pav/view?usp=share_link</a>
2	Design Datamart (Table Base)	<a href="https://drive.google.com/file/d/1lbChviXJD0Jlv_Q040pNZdJQ0PU7Paa3/view?usp=share_link">https://drive.google.com/file/d/1lbChviXJD0Jlv_Q040pNZdJQ0PU7Paa3/view?usp=share_link</a>
3	Design Datamart (Table Aggregate)	<a href="https://drive.google.com/file/d/1nrKAZerru5a03l-MgeJY_iZLrPWje6ba/view?usp=sharing">https://drive.google.com/file/d/1nrKAZerru5a03l-MgeJY_iZLrPWje6ba/view?usp=sharing</a>

# Table Base "<<base\_table>>"



```
1  -- Membuat Tabel Base
2  • CREATE TABLE base_table AS (
3      SELECT
4          pn.id_invoice, pn.tanggal, pn.id_customer, pl.nama, pn.id_distributor,
5          pn.id_cabang, pl.cabang_sales, pl.id_group, pl.group, pn.id_barang,
6          br.nama_barang, pn.brand_id, br.kode_lini, pn.lini, br.kemasan, pn.jumlah_barang,
7          pn.harga, pn.mata_uang
8      FROM Penjualan AS pn
9      LEFT JOIN Pelanggan AS pl ON pn.id_customer = pl.id_customer
10     LEFT JOIN Barang AS br ON pn.id_barang = br.kode_barang
11 );
12
13 • select * from base_table;
```

# Table Base "<<base\_table>>"

Column	Data Type	Description	Transformation
id_invoice	VARCHAR	ID setiap pembelian customer	-
tanggal	DATE	Tanggal transaksi	-
id_customer	VARCHAR	ID Pelanggan	-
nama	TEXT	Nama Pelanggan	-
id_distributor	VARCHAR	ID Distributor	-
id_cabang	VARCHAR	ID Cabang	-
cabang_sales	TEXT	Cabang Penjualan	-
id_group	TEXT	ID Group	-



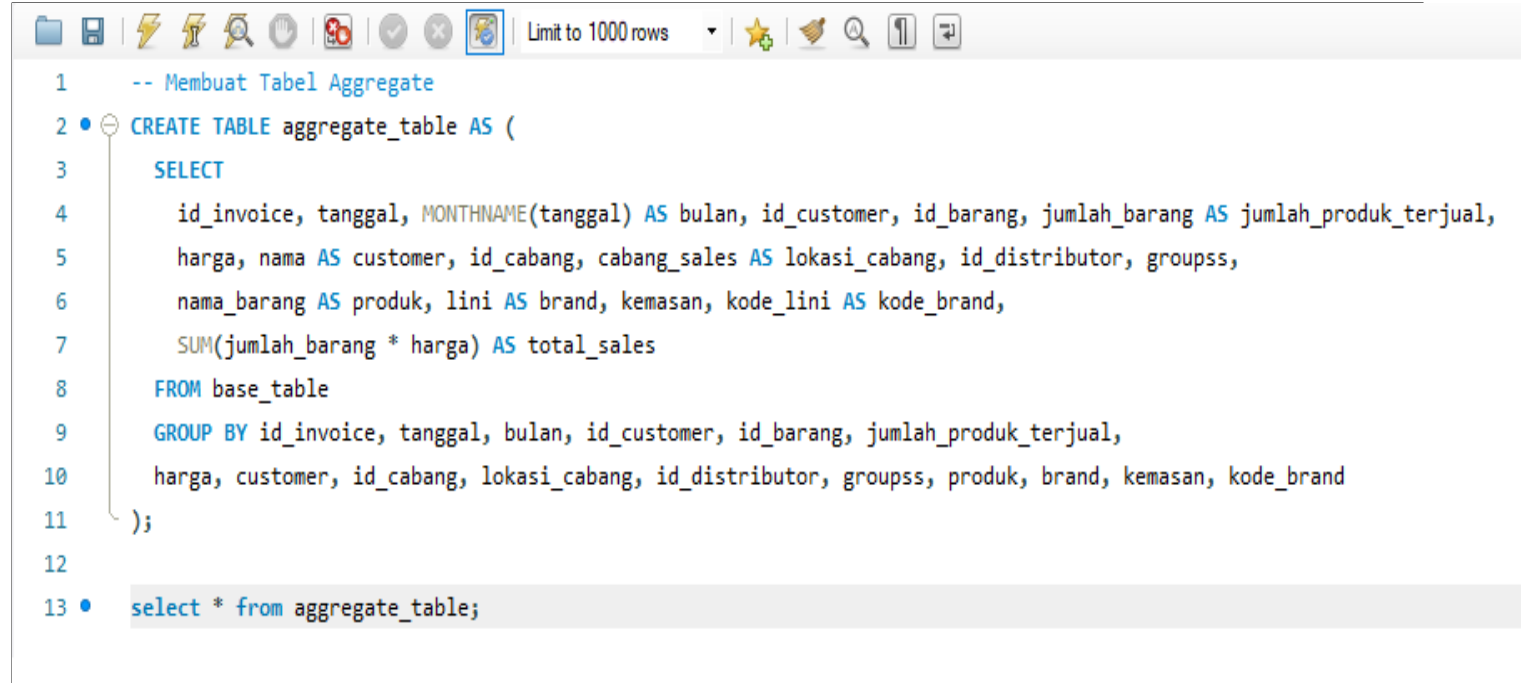
# Table Base "<<base\_table>>"

Column	Data Type	Description	Transformation
groupss	TEXT	Nama Group	-
id_barang	VARCHAR	ID Barang	-
nama_barang	TEXT	Nama Barang	-
brand_id	TEXT	ID Brand	-
kode_lini	INT	Kode Brand	-
lini	TEXT	Brand	-
kemasan	TEXT	Kemasan untuk setiap produk	-
jumlah_barang	INT	Jumlah produk yang terjual	-

# Table Base “<<base\_table>>”

Column	Data Type	Description	Transformation
harga	DECIMAL	Harga satuan pada produk	-
mata_uang	TEXT	Kode mata uang negara	-

# Table Aggregate "<<aggregate\_table>>"



The screenshot shows a SQL IDE interface with a toolbar at the top containing icons for file operations, execution, and search. Below the toolbar, a SQL script is displayed with line numbers 1 through 13. The script creates a table named 'aggregate\_table' and inserts data from a 'base\_table'.

```
1  -- Membuat Tabel Aggregate
2  • CREATE TABLE aggregate_table AS (
3      SELECT
4          id_invoice, tanggal, MONTHNAME(tanggal) AS bulan, id_customer, id_barang, jumlah_barang AS jumlah_produk_terjual,
5          harga, nama AS customer, id_cabang, cabang_sales AS lokasi_cabang, id_distributor, groupss,
6          nama_barang AS produk, lini AS brand, kemasan, kode_lini AS kode_brand,
7          SUM(jumlah_barang * harga) AS total_sales
8      FROM base_table
9      GROUP BY id_invoice, tanggal, bulan, id_customer, id_barang, jumlah_produk_terjual,
10         harga, customer, id_cabang, lokasi_cabang, id_distributor, groupss, produk, brand, kemasan, kode_brand
11  );
12
13  • select * from aggregate_table;
```

# Table Aggregate “<<aggregate\_table>>”

Column	Data Type	Description	Transformation
id_invoice	VARCHAR	ID setiap pembelian customer	-
tanggal	DATE	Tanggal transaksi	-
bulan	VARCHAR	Bulan transaksi	Penambahan column 'bulan' menggunakan MONTHNAME
id_customer	VARCHAR	ID Pelanggan	-
id_barang	VARCHAR	ID barang yang terjual	-
jumlah_produk_terjual	INT	Jumlah produk yang terjual	Perubahan nama column "jumlah_barang' menjadi 'jumlah_produk_terjual'
harga	DECIMAL	Harga satuan produk	-
customer	TEXT	Nama customer (store)	Perubahan nama column 'nama' menjadi 'customer'

# Table Aggregate “<<aggregate\_table>>”

Column	Data Type	Description	Transformation
id_barang	TEXT	ID barang yang terjual	-
lokasi_cabang	TEXT	Cabang penjualan	Perubahan nama column 'cabang_sales' menjadi lokasi_cabang'
id_distributor	VARCHAR	ID distributor	-
groupss	TEXT	Nama group	-
produk	TEXT	Nama produk	Perubahan nama column 'nama_barang' menjadi 'produk'
brand	TEXT	Nama brand	Perubahan nama column 'lini' menjadi 'brand'
kemasan	TEXT	Kemasan untuk setiap produk	

# Table Aggregate “<<aggregate\_table>>”

Column	Data Type	Description	Transformation
kode_brand	INT	Kode pada brand	Perubahan nama column 'kode_lini' menjadi 'kode_brand'
total_sales	DECIMAL	Nilai total pada setiap produk yang dikalikan dengan quantity	(jumlah_produk_terjual*price)

## Soal 5 : Data Visualization

### A. Tugas

buatlah data visualiasasi nya, dan cantumkan linknya di bawah (pastikan bisa diakses publik).  
Lalu cantumkan juga screenshot visualisasinya

Silahkan tambah halaman jika dibutuhkan

### A. Jawaban :

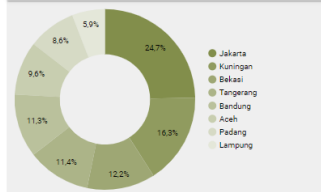
Link visualisasi <https://lookerstudio.google.com/reporting/65fc2f9a-afdb-43c8-824c-af2642b35cb1>

Branch Area

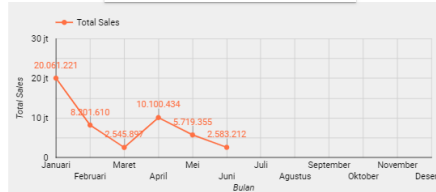
BIG DATA ANALYTICS

Product

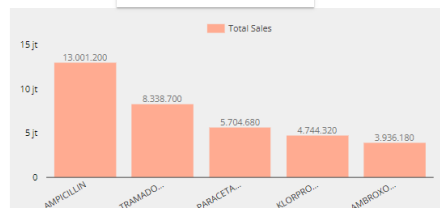
## SALES PERFORMANCE DISTRIBUTION AREA



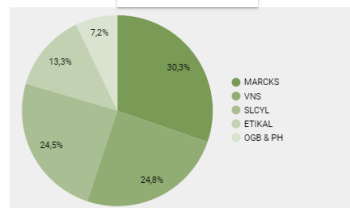
## TOTAL SALES MONTH TO MONTH



## TOP 5 SALES BY PRODUCT



## SALES BY BRAND



## TOTAL SALES AND QUANTITY (SOLD) BY CUSTOMER

ID Invoice	Customer Name	Quantity (Sold)	Total Sales
1. IN5263	APOTEK SINAR JAYA	130	1.389.700
2. IN5285	KLINIK SAHABAT	130	596.960
3. IN6181	KLINIK SAHABAT	130	902.200
4. IN5286	APOTEK SAHABAT	120	693.600
5. IN6302	APOTEK SINAR JAYA	120	338.280
6. IN6316	APOTEK SINAR JAYA	120	1.282.800
7. IN6097	APOTEK MAJA	100	578.000
8. IN6056	APOTEK MAJA	100	233.700
9. IN6203	KLINIK GM	100	564.800
10. IN6248	KLINIK GM	100	399.100

1 - 350 / 350



## Soal 6 : Additional Complementary Data

A. Tugas :

Dari data yang tersedia, menurut kamu untuk melengkapi analisis nya apakah diperlukan data lain juga? jika iya, sebutkan data apa yang kamu maksud dan mengapa memerlukan data tersebut

A. Jawaban : Data tambahan yang diperlukan meliputi informasi demografis untuk memungkinkan analisis yang lebih mendalam berdasarkan karakteristik demografis, seperti pemahaman tentang bagaimana segmentasi pasar. Selain itu, data pengiriman seperti tanggal pengiriman juga menjadi penting karena dapat digunakan untuk mengevaluasi sejauh mana proses transaksi berlangsung dan mengidentifikasi potensi keterlambatan dalam pengiriman.