

Soal & Template Jawaban

Task 5

Nama : Marselius Agus Dhion.

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 :

Kedua query akan menghasilkan hasil yang sama, dimana keduanya akan retrieve data alamat yang berawalan 'Mat'. Namun query (b) merupakan query yang lebih baik digunakan. Karena, untuk menjelaskan query ini lebih mudah dibandingkan query (a). Dimana, kita hanya menjelaskan bahwa 'Mat%' mengartikan bahwa kita sedang mencari alamat berawalan 'Mat', misal 'Mataram'. Sedangkan, query (a), kita harus menjelaskan apa itu fungsi SUBSTR(), belum lagi kita harus jelaskan paramaternya, index yang mau dipilih, dll. Jadi, query (b) ini lebih baik digunakan, karena lebih mudah untuk dipahami oleh orang yang awam.

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 : Kedua query menghasilkan hasil yang sama. Namun, opsi (b) lebih baik digunakan. Karena, query ini lebih mudah dipahami bagi orang yang pemula sekalipun, misalnya kita mau presentasi ke stakeholder atau digunakan oleh seorang data analyst junior. Maka query ini lebih mudah dipahami/dicerna oleh orang awam dibandingkan menggunakan operator seperti '>=' atau '<='

Soal 3: Menentukan Primary Key

A. Tugas

Tentukan primary key dari table penjualan. jelaskan alasannya

B. Jawaban : id_invoice.

Penjelasan :

=> Karena, id_invoice merupakan satu-satunya field yang unique pada table ini, dimana setiap data penjualan memiliki id_invoice yang berbeda. Sedangkan field-field lainnya, pada setiap baris datanya memiliki nilai yang redudansi atau tidak unique. Oleh karena itu, id_invoice merupakan field yang cocok menjadi primary key pada table ini.

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 : Tabel base -> **table_base.sql & table_dim_fact** || Tabel Aggregate -> **table_aggregate.sql**
Link GDrive : [Link GDrive - File & Screenshoot Query](#)

No	Nama File	Link
1.	table_base.sql	Table Base Link GDrive
2.	table_dim_fact.sql	Table Dimension-Fact Link GDrive
3.	table_aggregate.sql	Table Aggregate Link GDrive

Design Snowflake Schema

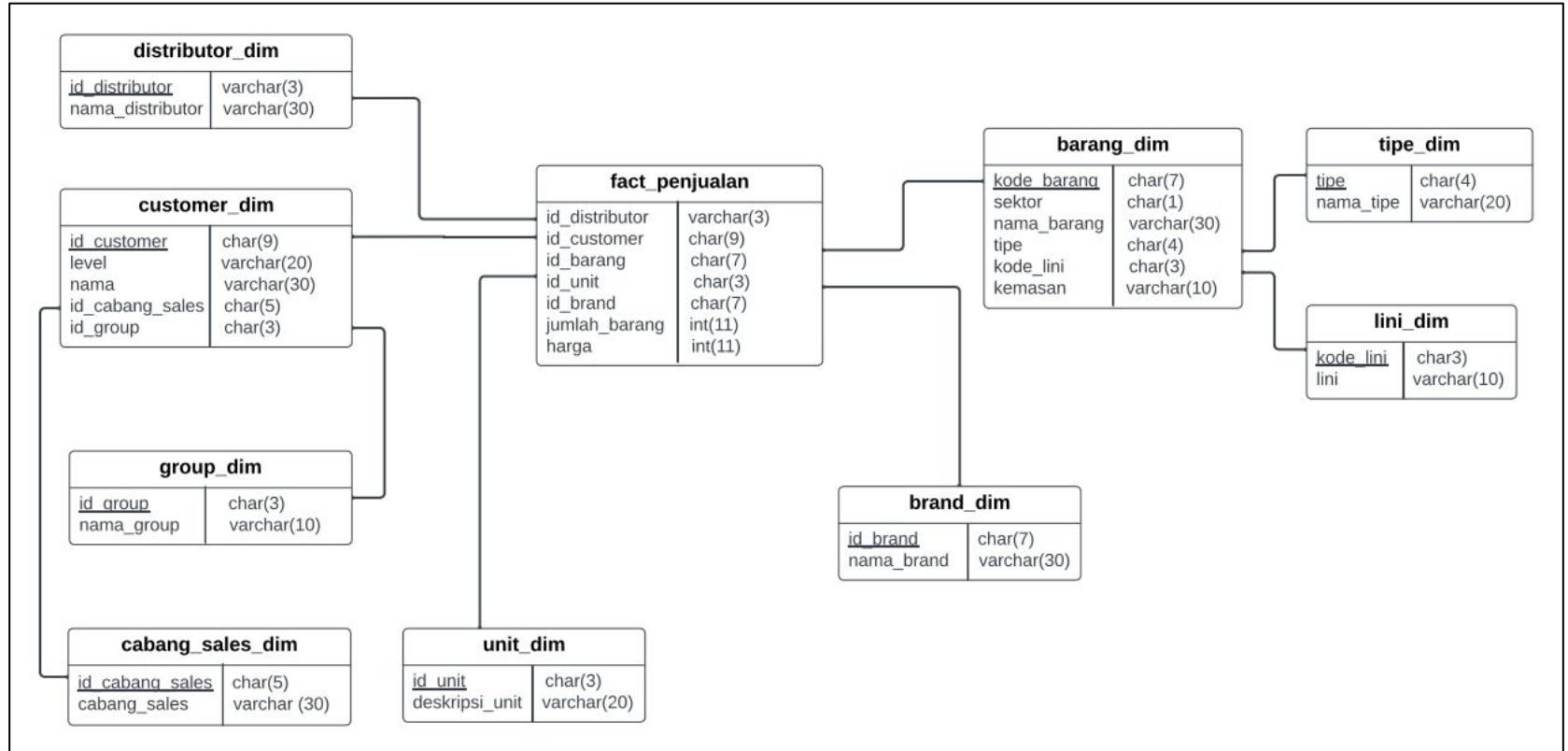


Table Base **distributor_dim**

```
1 • ○ CREATE TABLE distributor_dim (  
2     id_distributor VARCHAR(3) PRIMARY KEY,  
3     nama_distributor VARCHAR(30)  
4 );
```


Table Base distributor_dim

[illegible]

Table Base **group_dim**

```
5 • ○ CREATE TABLE group_dim (  
6     id_group CHAR(3) PRIMARY KEY,  
7     nama_group VARCHAR(10)  
8 );
```

Table Base group_dim

[illegible]

Table Base cabang_sales_dim

```
9 • CREATE TABLE cabang_sales_dim (  
10     id_cabang_sales CHAR(5) PRIMARY KEY,  
11     cabang_sales VARCHAR(30)  
12 );
```

Table Base cabang_sales_dim

[illegible]

Table Base **unit_dim**

```
13 • ○ CREATE TABLE unit_dim (  
14     id_unit CHAR(3) PRIMARY KEY,  
15     deskripsi_unit VARCHAR(20)  
16 );
```

Table Base unit_dim

[illegible]

Table Base **brand_dim**

```
17 • ○ CREATE TABLE brand_dim (  
18     id_brand CHAR(7) PRIMARY KEY,  
19     nama_brand VARCHAR(30)  
20 );
```


Table Base brand_dim

[illegible]

Table Base tipe_dim

```
21 • CREATE TABLE tipe_dim (  
22     tipe CHAR(4) PRIMARY KEY,  
23     nama_tipe VARCHAR(20)  
24 );
```

Table Base tipe_dim

[illegible]

Table Base lini_dim

```
25 • ○ CREATE TABLE lini_dim (  
26     kode_lini CHAR(3) PRIMARY KEY,  
27     lini VARCHAR(10)  
28 );
```

Table Base lini_dim

[illegible]

Table Base **customer_dim**

```
29 • ○ CREATE TABLE customer_dim(  
30     id_customer CHAR(9) PRIMARY KEY,  
31     level VARCHAR(20),  
32     nama VARCHAR(30),  
33     id_cabang_sales CHAR(5),  
34     id_group CHAR(3),  
35     FOREIGN KEY(id_cabang_sales) REFERENCES cabang_sales_dim(id_cabang_sales),  
36     FOREIGN KEY(id_group) REFERENCES group_dim(id_group)  
37 );  
38
```

Table Base **customer_dim**

column	data type	description	transformation
id_customer	char(9)	Primary Key	
level	varchar(20)	Level customer	
nama	varchar(30)	Nama customer	
id_cabang_sales	char(5)	Foreign Key dari table cabang_sales_dim	
id_group	char(3)	Foreign Key dari table group_dim	

Table Base **barang_dim**

```
41 • ○ CREATE TABLE barang_dim(  
42     kode_barang CHAR(7) PRIMARY KEY,  
43     sektor CHAR(1),  
44     nama_barang VARCHAR(30),  
45     tipe CHAR(4),  
46     kode_lini CHAR(3),  
47     kemasan VARCHAR(10),  
48     FOREIGN KEY(tipe) REFERENCES tipe_dim(tipe),  
49     FOREIGN KEY(kode_lini) REFERENCES lini_dm(kode_lini)  
50 );  
51 |
```


Table Base **barang_dim**

column	data type	description	transformation
kode_barang	char(7)	Primary Key	
sektor	char(1)	Sektor barang	
nama_barang	varchar(30)	Nama barang	
tipe	char(4)	Foreign Key dari table tipe_dim	
kode lini	char(3)	Foreign Key dari table lini_dim	
kemasan	varchar(10)	Tipe kemasan barang	

Table Base **fact_penjualan**

```
50 • CREATE TABLE fact_penjualan(  
51     id_distributor VARCHAR(3),  
52     id_customer CHAR(9),  
53     id_barang CHAR(7),  
54     id_unit CHAR(3),  
55     id_brand CHAR(7),  
56     jumlah_barang INT,  
57     harga INT,  
58     FOREIGN KEY(id_distributor) REFERENCES distributor_dim(id_distributor),  
59     FOREIGN KEY(id_customer) REFERENCES customer_dim(id_customer),  
60     FOREIGN KEY(id_barang) REFERENCES barang_dim(kode_barang),  
61     FOREIGN KEY(id_unit) REFERENCES unit_dim(id_unit),  
62     FOREIGN KEY(id_brand) REFERENCES brand_dim(id_brand)  
63 )  
64
```

Table Base **fact_penjualan**

column	data type	description	transformation
id_distributor	varchar(3)	Foreign Key dari table distributor_dim	
id_customer	char(9)	Foreign Key dari table customer_dim	
id_barang	char(7)	Foreign Key dari table barang_dim	
id_unit	char(3)	Foreign Key dari table unit_dim	
id_brand	char(7)	Foreign Key dari table brand_dim	Mengubah nama field dari "brand_id" menjadi "id_brand"
jumlah_barang	int	Jumlah barang yang dijual	
harga	int	Harga barang	Mengubah tipe data dari float ke int

Table Aggregate **penjualan_agg**

```
1 • CREATE TABLE penjualan_agg (  
2     id_invoice CHAR(6),  
3     id_distributor VARCHAR(3),  
4     id_customer CHAR(9),  
5     id_barang CHAR(7),  
6     id_unit CHAR(3),  
7     id_brand CHAR(7),  
8     tanggal DATE,  
9     total_harga INT  
10 );
```

Table Aggregate **penjualan_agg**

column	data type	description	transformation
id_invoice	char(6)	-	
id_distributor	varchar(3)	-	
id_customer	char(9)	-	
id_barang	char(7)	-	
id_unit	char(3)	-	
id_brand	char(7)	-	
tanggal	date	Tanggal Penjualan	
total_harga	int	Perkalian jumlah_barang dengan harga pada table fact_penjualan	

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

B. Jawaban :

Link visualisasi (ex link Google Data Studio) : [Link Google Data Studio](#)

Sales Report of Kimia Farma

Customer ID

Select date range

Branch ID

Total Branches

8

Total Transactions

350

Total Revenue

Rp49.22M

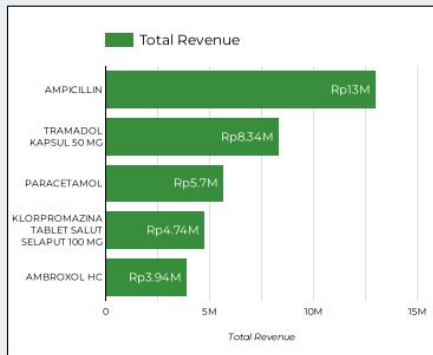
Number of Items Sold

9,176

Total Items

10

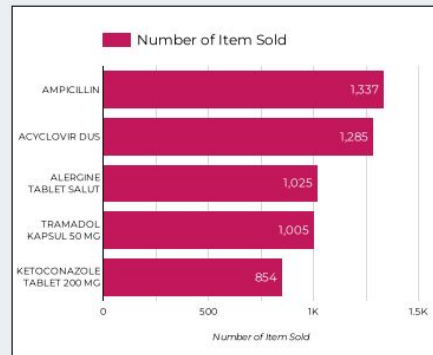
Total Sales per Items



Total Revenue & Quantity Items (Sold) Trend per Month

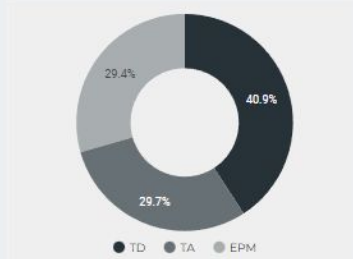


Number of Items Sold

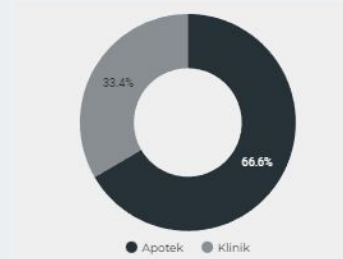


	Date	ID Invoice	Customers Name	Items Name	Number of Items	Items Price	Total Revenue
1.	31 Jan 2022	IN6263	APOTEK SINAR JAYA	AMPICILLIN	130	Rp10,690.6	Rp1,389,778
2.	19 Apr 2022	IN6316	APOTEK SINAR JAYA	AMPICILLIN	120	Rp10,690.6	Rp1,282,872
3.	27 Jan 2022	IN6074	KLINIK GM	AMPICILLIN	98	Rp10,690.6	Rp1,047,678.8
4.	27 Jan 2022	IN6124	KLINIK GM	AMPICILLIN	95	Rp10,690.6	Rp1,015,607
5.	7 Feb 2022	IN6181	KLINIK SAHABAT	PARACETAMOL	130	Rp6,940.3	Rp902,239
Grand total						Rp1,843,492.35	Rp49,216,465.14

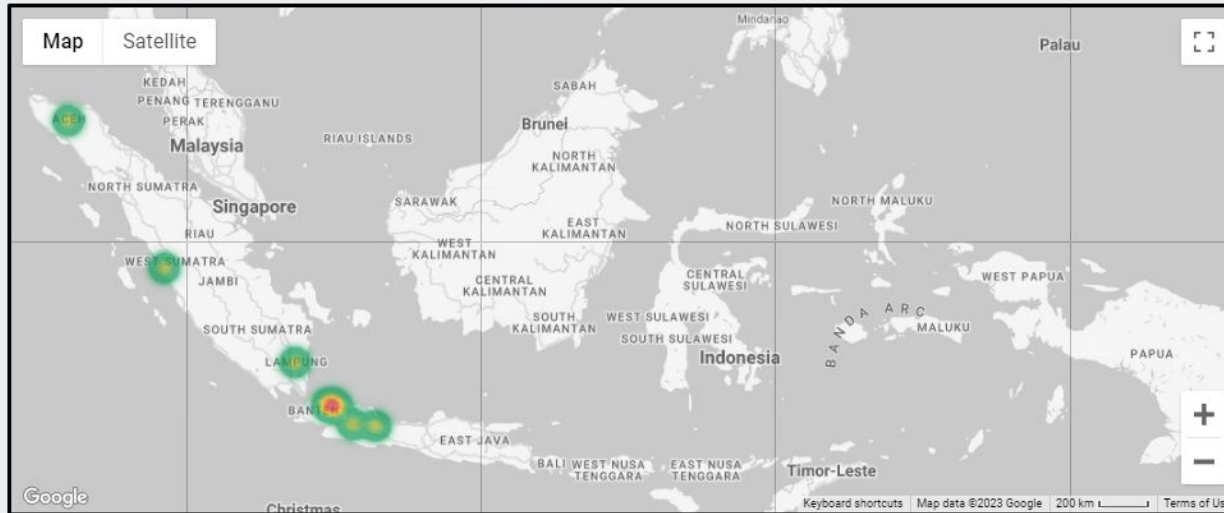
Distributor Percentage Comparison



Group Percentage Comparison



City Distribution Map



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

B. Jawaban :

- Pertama, data pengeluaran Kimia Farma. Dengan adanya data ini, nantinya bisa dibandingkan tiap periodenya (minggu, bulan, tahun, dsb), Kimia Farma ini bisa mendapatkan keuntungan/pemasukan yang bersih berapa. Yaitu dengan cara mengurangi pemasukan dengan pengeluaran tiap periode.
- Kedua, data mengenai distributor dan brand. Karena pada data yang tersedia, hanya terdapat id_distributor dan id_brand. Namun, tidak ada nama/deskripsi yang spesifik, bahwa id sekian merupakan distributor/brand apa.
- Terakhir, data mengenai perilaku customer. Dengan adanya data ini, nantinya customer A dengan customer B bisa diberi treatment yang berbeda dan berpatokan terhadap perilaku mereka setiap melakukan pembelian.