

Soal & Template Jawaban

Task 5

Nama : Rifaldi Tajrial

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 : `SELECT * FROM pelanggan WHERE alamat LIKE 'Mat%'`

Alasan : Query yang menggunakan operator LIKE memberikan tingkat pemahaman yang lebih baik oleh mesin, sehingga dapat memberikan fleksibilitas yang lebih besar dalam pencarian pola yang dimulai dengan kata "Mat" dari seluruh baris dalam tabel.

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 : `SELECT * FROM pelanggan WHERE tanggal_lahir BETWEEN '2000-01-01' AND '2008-12-31'`

Alasan : Penggunaan Query BETWEEN lebih mudah dipahami dan sederhana dalam menyatakan rentang tanggal. Sintaks ini umumnya dipergunakan untuk melakukan filter terhadap tipe data Angka maupun Tanggal.

Soal 3: Menentukan Primary Key

A. Tugas

Tentukan primary key dari table penjualan. jelaskan alasannya

B. Jawaban & Penjelasan : id_invoice, id_customer, dan kode_barang

Alasan:

Karena, nilai-nilai dalam tersebut unik tidak tercantum dalam tabel lain yang dapat diidentifikasi dengan cara yang jelas dan tidak ambigu.

Soal 4: Design Datamart

A. Tugas

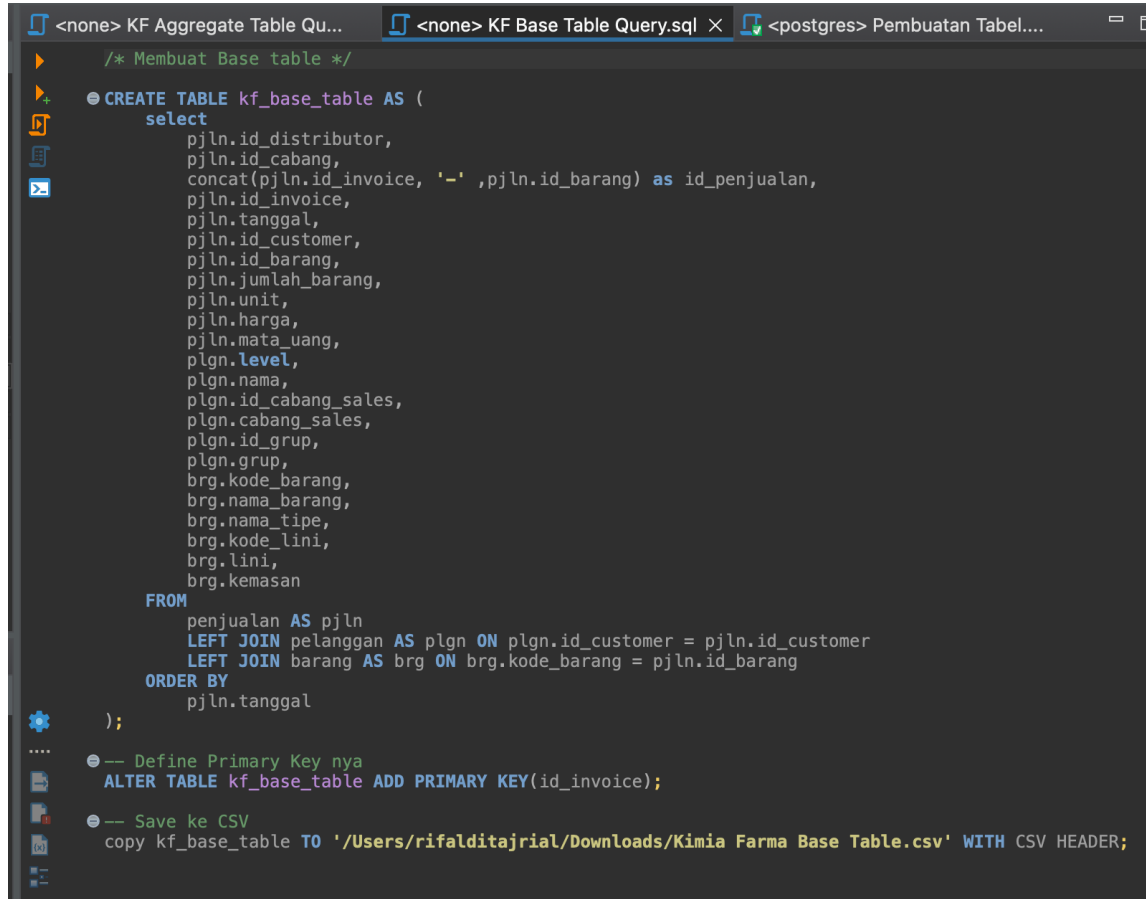
Buatlah design datamart (Terdiri dari tabel base, dan tabel aggregate). Upload file query dalam gdrive mu (pastikan dapat diakses public). Lalu masukkan linknya di tabel di bawah, dan cantumkan juga screenshoot query nya (jika lebih dari 1 file, maka masing masing file di-screenshoot)

Silahkan tambah halaman jika dibutuhkan

A. Jawaban :

No	Nama File	Link
1.	Pembuatan Tabel.sql	https://drive.google.com/file/d/1GE42bJsXkbTfotmLj1zzlkUmNPuQy-mS/view?usp=sharing
2.	KF Base Table Query.sql	https://drive.google.com/file/d/12glG_8PLAfltvrakvQhP61LRlqF3RHUa/view?usp=sharing
3.	KF Aggregate Table Query.sql	https://drive.google.com/file/d/1Mu_nNLNdfzoG-CCUY1pMWC9CEhFhkO9I/view?usp=sharing

Table Base “kf_base_table”

A screenshot of a SQL IDE window with three tabs: '<none> KF Aggregate Table Qu...', '<none> KF Base Table Query.sql' (active), and '<postgres> Pembuatan Tabel...'. The active tab contains SQL code for creating a table. The code starts with a comment '/* Membuat Base table */', followed by a 'CREATE TABLE' statement for 'kf_base_table'. The table is created as a SELECT query from several tables: 'penjualan', 'pelanggan', and 'barang'. The SELECT clause lists various columns including distributor ID, branch ID, invoice ID concatenated with product ID, invoice ID, date, customer ID, product ID, quantity, unit, price, currency, product level, product name, sales branch ID, sales branch name, product group, product name, product type, product code, product line, and packaging. The FROM clause uses LEFT JOINs to combine the tables. The ORDER BY clause orders by 'penjualan.tanggal'. The statement ends with a semicolon. Below this, there are two additional comments: '-- Define Primary Key nya' followed by an 'ALTER TABLE' statement to add a primary key on 'id_invoice', and '-- Save ke CSV' followed by a 'copy' command to save the table data to a CSV file with a header.

```
/* Membuat Base table */

CREATE TABLE kf_base_table AS (
    select
        pjln.id_distributor,
        pjln.id_cabang,
        concat(pjln.id_invoice, '-', pjln.id_barang) as id_penjualan,
        pjln.id_invoice,
        pjln.tanggal,
        pjln.id_customer,
        pjln.id_barang,
        pjln.jumlah_barang,
        pjln.unit,
        pjln.harga,
        pjln.mata_uang,
        plgn.level,
        plgn.nama,
        plgn.id_cabang_sales,
        plgn.cabang_sales,
        plgn.id_grup,
        plgn.grup,
        brg.kode_barang,
        brg.nama_barang,
        brg.nama_tipe,
        brg.kode_lini,
        brg.lini,
        brg.kemasan
    FROM
        penjualan AS pjln
        LEFT JOIN pelanggan AS plgn ON plgn.id_customer = pjln.id_customer
        LEFT JOIN barang AS brg ON brg.kode_barang = pjln.id_barang
    ORDER BY
        pjln.tanggal
);

....

-- Define Primary Key nya
ALTER TABLE kf_base_table ADD PRIMARY KEY(id_invoice);

-- Save ke CSV
copy kf_base_table TO '/Users/rifalditajrial/Downloads/Kimia Farma Base Table.csv' WITH CSV HEADER;
```

Table Base “kf_base_table.sql”

column	data type	description	transformation
id_distributor	varchar	Unique Value for Distributor	-
id_cabang	varchar	Unique Value for Branch	-
id_penjualan	varchar	Primary Key (PK)	concat(pjln.id_invoice, '-', pjln.id_barang) as id_penjualan
id_invoice	varchar	Unique Value for Invoice	-
tanggal	date	Date of the Transaction	-
id_customer	varchar	Unique Value for Customer	-
id_barang	varchar	Unique Value for Items	-
jumlah_barang	decimal	Number of Items Sold	-

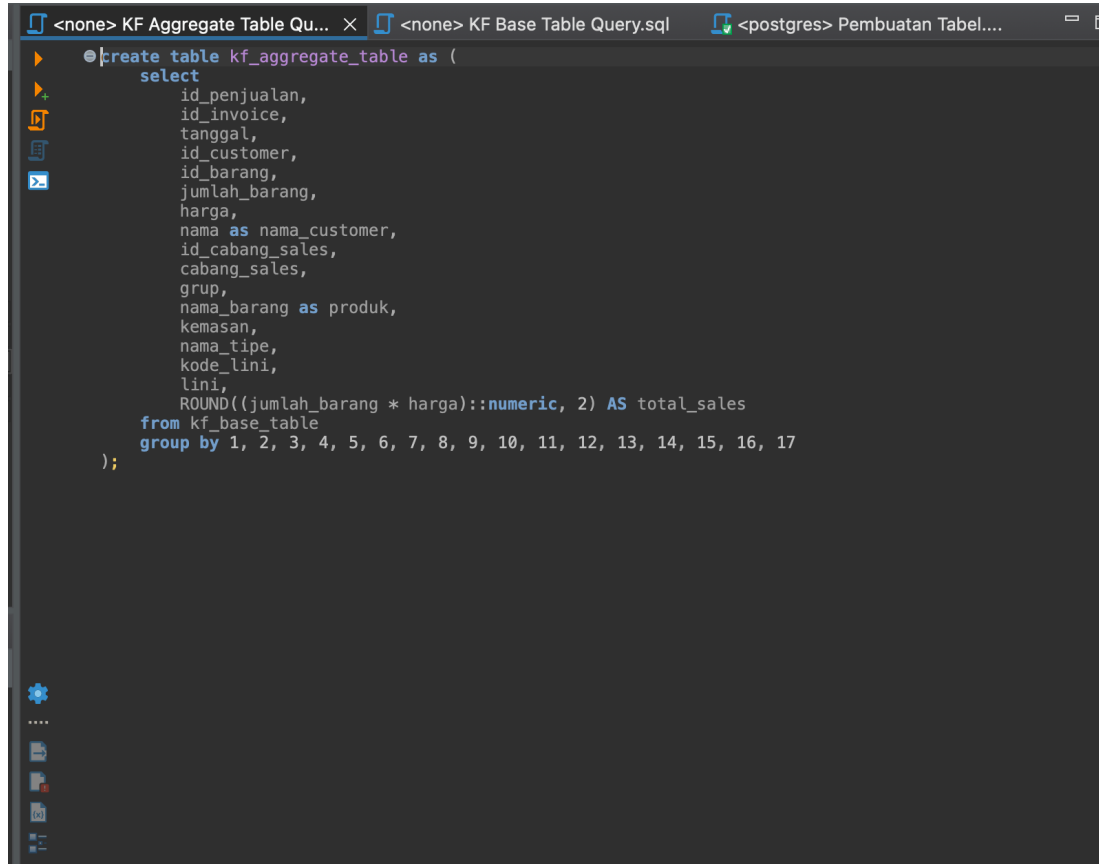
Table Base “kf_base_table.sql”

column	data type	description	transformation
unit	varchar	Measurement for Items Sold	-
harga	decimal	Price per Item	-
mata_uang	varchar	Transaction Currency (using IDR)	-
level	varchar	Level of Product Line	-
nama	varchar	Name of the Products	-
id_cabang_sales	varchar	Unique Value for Sales Branch	-
cabang_sales	varchar	City of Sales Branch	-
id_grup	varchar	Unique Value for Group of Health Services	-

Table Base “kf_base_table.sql”

column	data type	description	transformation
grup	varchar	Group of Health Services Facilities	-
kode_barang	varchar	Code for Item Sold	-
nama_barang	varchar	Name of the Item Sold	-
nama_tipe	varchar	Name of Type Item Sold	-
kode_lini	integer	Code of the Product Line	-
lini	varchar	Name of Category Product Line	-
kemasan	varchar	Packaging type for the Item	-

Table Aggregate "kf_aggregate_table"



```
<none> KF Aggregate Table Qu... X <none> KF Base Table Query.sql <postgres> Pembuatan Tabel...
● create table kf_aggregate_table as (
  select
    id_penjualan,
    id_invoice,
    tanggal,
    id_customer,
    id_barang,
    jumlah_barang,
    harga,
    nama as nama_customer,
    id_cabang_sales,
    cabang_sales,
    grup,
    nama_barang as produk,
    kemasan,
    nama_tipe,
    kode_lini,
    lini,
    ROUND((jumlah_barang * harga)::numeric, 2) AS total_sales
  from kf_base_table
  group by 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
);
```

Table Aggregate "kf_aggregate_table.sql"

column	data type	description	transformation
id_penjualan	varchar	Primary Key (PK)	-
id_invoice	varchar	Unique Value for Invoice	-
tanggal	date	Date of the Transaction	-
id_customer	varchar	Unique Value for Customer	-
id_barang	varchar	Unique Value for Items	-
jumlah_barang	decimal	Number of Items Sold	-
harga	decimal	Price per Item	-
nama_customer	varchar	Name of the Customer	nama AS nama_csutomer

Table Aggregate "kf_aggregate_table.sql"

column	data type	description	transformation
id_cabang_sales	varchar	Unique Value for Sales Branch	-
cabang_sales	varchar	City of Sales Branch	-
grup	varchar	Group of Health Services Facilities	-
nama_barang	varchar	Name of the Item Sold	nama_barang AS produk
kemasan	varchar	Packaging type for the Item	-
nama_tipe	varchar	Name of Type Item Sold	-
kode_lini	integer	Code of the Product Line	-
lini	varchar	Name of Category Product Line	nama AS nama_csutomer
total_sales	integer	the quantity of stuffs multiplied by unit price of the stuffs	ROUND ((jumlah_barang * harga) : : numeric, 2) AS total_sales

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 :

https://public.tableau.com/views/KimiaFarmaSalesPerformanceDashboard/Dashboard1?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link



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 :

Menurut pendapat saya, data yang sebaiknya dimasukkan dalam analisis ini mencakup penggolongan obat berdasarkan tingkat keamanan penggunaannya, seperti Obat Bebas, Obat Bebas Terbatas, dan Obat Terlarang. Selain itu, penting juga untuk memasukkan informasi mengenai bentuk fisik obat, apakah dalam bentuk tablet, kapsul, sirup, injeksi, dan sebagainya.