Bagian 1 : Membuat Entity Relationship Diagram

```
SELECT * FROM order_payments_dataset;

ALTER TABLE sellers_dataset

ADD PRIMARY KEY (seller_id);

ALTER TABLE order_items_dataset

ADD CONSTRAINT seller_id FOREIGN KEY (seller_id) REFERENCES sellers_dataset(seller_id);


ALTER TABLE orders_dataset

ADD PRIMARY KEY (order_id);

ALTER TABLE order_items_dataset

ADD CONSTRAINT order_id FOREIGN KEY (order_id) REFERENCES orders_dataset;


ALTER TABLE order_reviews_dataset

ADD CONSTRAINT order_id FOREIGN KEY (order_id) REFERENCES orders_dataset;


ALTER TABLE product_dataset

ADD PRIMARY KEY (product_id);

ALTER TABLE order_items_dataset

ADD CONSTRAINT product_id FOREIGN KEY (product_id) REFERENCES product_dataset;


ALTER TABLE order_payments_dataset

ADD CONSTRAINT order_id FOREIGN KEY (order_id) REFERENCES orders_dataset;


ALTER TABLE customer_dataset

ADD PRIMARY KEY (customer_id);

ALTER TABLE orders_dataset

ADD CONSTRAINT customer_id FOREIGN KEY (customer_id) REFERENCES customer_dataset;


ALTER TABLE customer_dataset
```

ADD CONSTRAINT zip_code_prefix **FOREIGN KEY** (customer_zip_code_prefix) **REFERENCES** geolocation_dataset

Bagian 2 : Analisis Pertumbuhan Aktifitas Pelanggan Tahunan

Bagian 2.1 : Menampilkan Jumlah Pelanggan Aktif Bulanan (Monthly Active User)

WITH tmp **AS** (

SELECT EXTRACT(**YEAR FROM** order_purchase_timestamp) **AS** tahun,

EXTRACT(**MONTH FROM** order_purchase_timestamp) **AS** bulan,

COUNT(**DISTINCT** customer_id) **AS** jumlah_pelanggan_aktif_bulanan

FROM orders_dataset

GROUP BY EXTRACT(**YEAR FROM** order_purchase_timestamp),

EXTRACT(**MONTH FROM** order_purchase_timestamp)

ORDER BY tahun, bulan),

Bagian 2.2 : Menampilkan Jumlah Pelanggan Baru (New User)

tmp2 **AS** (

SELECT EXTRACT(**YEAR FROM** first_order_date) **AS** tahun,

COUNT(*) **AS** jumlah_pelanggan_baru

FROM (

SELECT customer_id, **MIN**(order_purchase_timestamp) **AS** first_order_date

FROM orders_dataset

GROUP BY customer_id

) **AS** first_orders

GROUP BY EXTRACT(**YEAR FROM** first_order_date)

ORDER BY tahun),

Bagian 2.3 : Menampilkan Rata-Rata Order Customer (User Average Order)

tmp3 **AS** (

SELECT tahun,

AVG(order_count) AS rata_rata_order_per_customer

```
FROM (
SELECT customer_id, EXTRACT(YEAR FROM order_purchase_timestamp) AS tahun,
COUNT(*) AS order_count
FROM orders_dataset
GROUP BY customer_id,EXTRACT(YEAR FROM order_purchase_timestamp)
) AS order_counts_per_customer
GROUP BY tahun
ORDER BY tahun)
```

Bagian 2.4 : Menggabungkan Temporary Table yang Sudah Dibuat (CTE)

```
SELECT tmp2.tahun, jumlah_pelanggan_aktif_bulanan, jumlah_pelanggan_baru FROM
tmp JOIN orders_dataset AS od ON tmp.tahun = tmp.tahun
JOIN tmp2 ON tmp2.tahun = tmp.tahun
JOIN tmp3 ON tmp3.tahun = tmp2.tahun
GROUP BY 1,2,3
ORDER BY tahun
```

Bagian 3 : Analisis Kualitas Kategori Produk Tahunan

Bagian 3.1 : Menampilkan Jumlah Pendapatan (Total Revenue)

```
WITH tr AS (
SELECT EXTRACT(YEAR FROM o.order_purchase_timestamp) AS tahun,
SUM(od.price + od.freight_value) AS pendapatan_total
FROM orders_dataset o
INNER JOIN order_items_dataset AS od ON o.order_id = od.order_id
WHERE o.order_status = 'delivered'
GROUP BY tahun
ORDER BY tahun),
```

Bagian 3.2 : Menampilkan Jumlah Cancel Order (Annual Total Cancel Order)

```
tco AS (
SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS tahun,
COUNT(*) AS jumlah_cancel_order
FROM orders_dataset
WHERE order_status = 'canceled'
GROUP BY tahun
ORDER BY tahun),
```

Bagian 3.3 : Menampilkan Pendapatan Tiap Kategori (Revenue per Category)

```
rpc AS (
SELECT EXTRACT(YEAR FROM o.order_purchase_timestamp) AS tahun,
p.product_category_name,
SUM(oi.price) AS total_revenue,
ROW_NUMBER() OVER (PARTITION BY EXTRACT(YEAR FROM o.order_purchase_timestamp) ORDER BY SUM(oi.price) DESC) AS rn
FROM orders_dataset o
JOIN order_items_dataset oi ON o.order_id = oi.order_id
JOIN product_dataset p ON oi.product_id = p.product_id
WHERE o.order_status = 'delivered'
GROUP BY EXTRACT(YEAR FROM o.order_purchase_timestamp), p.product_category_name),
```

Bagian 3.4 : Menampilkan Pesanan Dibatalkan Tiap Kategori (Cancel Order per Category)

```
copc AS (
SELECT EXTRACT(YEAR FROM o.order_purchase_timestamp) AS tahun,
p.product_category_name,
COUNT(*) AS total_cancel_orders,
ROW_NUMBER() OVER (PARTITION BY EXTRACT(YEAR FROM o.order_purchase_timestamp) ORDER BY COUNT(*) DESC) AS rn
```

FROM orders_dataset o

JOIN order_items_dataset oi **ON** o.order_id = oi.order_id

JOIN product_dataset p **ON** oi.product_id = p.product_id

WHERE o.order_status = 'canceled'

GROUP BY EXTRACT(**YEAR FROM** o.order_purchase_timestamp), p.product_category_name)

Bagian 3.5 : Menggabungkan Temporary Table (CTE)

SELECT tr.tahun, copc.total_cancel_orders, rpc.total_revenue **FROM**

tr **JOIN** tco **ON** tr.tahun = tco.tahun

JOIN rpc **ON** rpc.tahun = tr.tahun

JOIN copc **ON** copc.tahun = tr.tahun

GROUP BY 1,2,3

ORDER BY 1

Bagian 4 : Analisis Metode Pembayaran Tahunan

Bagian 4.1 : Menampilkan Jumlah Masing-masing Metode Pembayaran

SELECT payment_type, **COUNT**(*) **AS** total_usage

FROM order_payments_dataset

GROUP BY payment_type

ORDER BY total_usage DESC;

Bagian 4.2 : Menampilkan Jumlah Metode Pembayaran Tahunan

SELECT EXTRACT(**YEAR FROM** order_purchase_timestamp) **AS** tahun, payment_type, **COUNT**(*) **AS** total_usage

FROM orders_dataset as o

JOIN order_payments_dataset opd **ON** o.order_id = opd.order_id

GROUP BY tahun, payment_type

ORDER BY tahun desc, total_usage DESC;