-- Query yang digunakan pada Analyze eCommerce Business Performance with SQL

-- Database: Ecommerce

-- DROP DATABASE IF EXISTS "Ecommerce";

**CREATE DATABASE** "Ecommerce"

**WITH**

**OWNER** = postgres

**ENCODING** = 'UTF8'

**LC\_COLLATE** = 'English\_Indonesia.1252'

**LC\_CTYPE** = 'English\_Indonesia.1252'

**TABLESPACE** = pg\_default

**CONNECTION LIMIT** = -1

**IS\_TEMPLATE** = False;

-- Membuat table customers dataset

**CREATE TABLE IF NOT EXISTS** public.customers

(

customer\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

customer\_unique\_id character varying **COLLATE** pg\_catalog."default",

customer\_zip\_code\_prefix integer,

city character varying **COLLATE** pg\_catalog."default",

state character varying **COLLATE** pg\_catalog."default",

**CONSTRAINT** customers\_pkey **PRIMARY KEY** (customer\_id)

);

-- Membuat table items dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.items

(

order\_id character varying **COLLATE** pg\_catalog."default",

order\_item\_id integer,

product\_id character varying **COLLATE** pg\_catalog."default",

seller\_id character varying **COLLATE** pg\_catalog."default",

shipping\_limit\_date timestamp without time zone,

price double precision,

fright\_value double precision

);

-- Membuat table orders dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.orders

(

order\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

customer\_id character varying **COLLATE** pg\_catalog."default",

order\_status character varying **COLLATE** pg\_catalog."default",

order\_purchase\_timestamp timestamp without time zone,

order\_approved\_at timestamp without time zone,

order\_delivered\_carrier\_date timestamp without time zone,

order\_delivered\_customer\_date timestamp without time zone,

order\_estimated\_delivery\_date timestamp without time zone,

**CONSTRAINT** orders\_pkey **PRIMARY KEY** (order\_id)

);

-- Membuat table payment dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.payment

(

order\_id character varying **COLLATE** pg\_catalog."default",

payment\_sequential integer,

payment\_type character varying **COLLATE** pg\_catalog."default",

payment\_installments integer,

payment\_value double precision

);

-- Membuat table product dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.product

(

index integer,

product\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

product\_category\_name character varying **COLLATE** pg\_catalog."default",

product\_name\_lenght double precision,

product\_description\_lenght double precision,

product\_photos\_qty double precision,

product\_weight\_g double precision,

product\_height\_cm double precision,

product\_widht\_cm double precision,

product\_lenght\_cm double precision,

**CONSTRAINT** product\_pkey **PRIMARY KEY** (product\_id)

);

-- Membuat table reviews dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.reviews

(

review\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

order\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

review\_score integer,

review\_comment\_title character varying **COLLATE** pg\_catalog."default",

review\_comment\_message character varying **COLLATE** pg\_catalog."default",

review\_creation\_date timestamp without time zone,

review\_answer\_timestamp timestamp without time zone,

**CONSTRAINT** reviews\_pkey **PRIMARY KEY** (review\_id, order\_id)

);

-- Membuat table seller dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.seller

(

seller\_id character varying **COLLATE** pg\_catalog."default" **NOT** **NULL**,

zip\_code\_prefix integer,

seller\_city character varying **COLLATE** pg\_catalog."default",

seller\_state character varying **COLLATE** pg\_catalog."default",

**CONSTRAINT** seller\_pkey **PRIMARY KEY** (seller\_id)

);

-- Membuat table geolocation dataset

**CREATE TABLE IF** **NOT** **EXISTS** public.geolocation

(

zip\_code\_prefix integer **NOT** **NULL**,

geolocation\_lat double precision,

geolocation\_lng double precision,

city character varying **COLLATE** pg\_catalog."default",

state character varying **COLLATE** pg\_catalog."default",

);

-- Menghapus data duplikat pada table geolocation kolom zip\_code\_prefix

**DELETE FROM** public.geolocation

**USING** geolocation as geo

**WHERE** geolocation.zip\_code\_prefix = geo.zip\_code\_prefix **AND** geolocation.ctid < geo.ctid;

-- Menambahkan constraint primary key pada table geolocation

**ALTER TABLE IF** **EXISTS** public.geolocation

**CONSTRAINT** geolocation\_pkey **PRIMARY KEY** (zip\_code\_prefix),

**CONSTRAINT** geolocation\_zip\_code\_prefix\_key **UNIQUE** (zip\_code\_prefix);

-- Menambahkan foreign key customer\_id dari table orders

**ALTER TABLE IF** **EXISTS** public.customers

**ADD FOREIGN KEY** (customer\_id)

**REFERENCES** public.orders (customer\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key zip\_code\_prefix dari table customers

**ALTER TABLE IF** **EXISTS** public.geolocation

**ADD FOREIGN KEY** (zip\_code\_prefix)

**REFERENCES** public.customers (customer\_zip\_code\_prefix) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key zip\_code\_prefix dari table seller

**ALTER TABLE IF** **EXISTS** public.geolocation

**ADD FOREIGN KEY** (zip\_code\_prefix)

**REFERENCES** public.seller (zip\_code\_prefix) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key seller\_id pada table items

**ALTER TABLE IF** **EXISTS** public.items

**ADD FOREIGN KEY** (seller\_id)

**REFERENCES** public.seller (seller\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key product\_id pada table items

**ALTER TABLE IF** **EXISTS** public.items

**ADD FOREIGN KEY** (product\_id)

**REFERENCES** public.product (product\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key order\_id pada table items

**ALTER TABLE IF** **EXISTS** public.items

**ADD FOREIGN KEY** (order\_id)

**REFERENCES** public.orders (order\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key customer\_id pada table orders

**ALTER TABLE IF** **EXISTS** public.orders

**ADD FOREIGN KEY** (customer\_id)

**REFERENCES** public.customers (customer\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key order\_id pada table payment

**ALTER TABLE IF** **EXISTS** public.payment

**ADD FOREIGN KEY** (order\_id)

**REFERENCES** public.orders (order\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;

-- Menambahkan foreign key order\_id pada table reviews

**ALTER TABLE IF** **EXISTS** public.reviews

**ADD FOREIGN KEY** (order\_id)

**REFERENCES** public.orders (order\_id) **MATCH SIMPLE**

**ON** **UPDATE** **NO** **ACTION**

**ON** **DELETE** **NO** **ACTION**

**NOT** **VALID**;