SQL

SIMPLE QUERY IN SIMPLE CASES

CREATED BY : AL QARANI

Description

This database consists of four tables containing customer data, address data, order data, and merchant data for the DataFood business.

1. The management wants to understand the extent of dataFood's growth and expansion so far. Please display all the unique cities from which customers originate.

```
SELECT DISTINCT kota
FROM customer_address
```

Data Output | Messages | Notifications

	kota character varying
1	Jakarta Barat
2	Jakarta Pusat
3	Tangerang
4	Jakarta Utara
5	Jakarta Selatan
6	Bekasi
7	Jakarta Timur
8	Depok

To display the unique values in a column (eliminating duplicates), you can use the SELECT DISTINCT statement followed by the name of the column you want to display.

2. The Engineering team has just updated the schema of the "orders" table. Display the 10 most recent transaction rows from the "orders" table to see the format of this table.

```
SELECT *
FROM orders
ORDER BY tanggal_pembelian DESC
LIMIT 10
```

Data Output | Messages | Notifications

	id_order numeric	id_pelanggan numeric	id_merchant numeric	tanggal_pembelian date	kuantitas numeric	harga numeric	ppn numeric	bayar_cash numeric	metode_bayar character varying
1	57	45	6	2019-12-26	2	54526	0.1	1	cash
2	38	74	2	2019-12-24	4	41460	0.1	1	cash
3	93	72	3	2019-12-20	2	43041	0.1	0	ovo
4	98	64	6	2019-12-13	5	64724	0.1	0	gopay
5	1	75	3	2019-12-12	5	39452	0.1	1	cash
6	68	50	6	2019-12-06	5	41892	0.1	0	gopay
7	46	75	3	2019-12-01	2	28565	0.1	1	cash
8	44	73	3	2019-11-09	3	40323	0.1	1	cash
9	32	46	2	2019-10-24	4	34266	0.1	1	cash
10	54	10	2	2019-10-23	4	68779	0.1	0	ovo

The "tanggal_pembelian" column indicates when the data row was recorded in the database. Therefore, to sort this column from the most recent to the oldest, you can use "ORDER BY ... DESC." Then, to get only the top 10 rows, you use "LIMIT 10."

3. The Risk team wants to understand the performance of the fraud detection system that has been running. Show the customer data that has been detected as fraudsters in DataFood.

```
SELECT *
FROM customer
WHERE penipu = 1
```

Data Output | Messages | Notifications

	id_pelanggan numeric	nama character varying	email character varying	telepon bigint	umur numeric	bulan_lahir character varying	tanggal_registrasi date	konfirmasi_telepon numeric	penipu numeric	pengguna_aktif numeric
1	17	Rifqy Butar-butar	rifqybutar-butar@yahoo.com	6289625971	30	September	2012-01-14	1	1	1
2	18	Wahyu Hayati	wahyuhayati@gmail.com	6289627184	23	Desember	2012-09-15	1	1	1
3	58	Nadia Persadani	nadiapersadani@roketmail.com	6287833759	25	Juli	2012-10-26	1	1	1
4	73	Natasha Fadholi	natashafadholi@gmail.com	6285282083	49	April	2013-02-08	1	1	1

To obtain rows of data consisting of fraudsters only, you can use the filter WHERE penipu = 1. Then, you can retrieve all columns from the customer table.

4. The Business Development team is currently establishing a new partnership with ShopeePay. For this purpose, they would like to obtain order data from DataFood that used the ShopeePay payment method and also see the largest transaction amounts.

```
SELECT *
FROM orders
WHERE metode_bayar = 'shopeepay'
ORDER BY kuantitas DESC
```

Data Output | Messages | Notifications

	id_order numeric	id_pelanggan numeric	id_merchant numeric	tanggal_pembelian date	kuantitas numeric	harga numeric	ppn numeric	bayar_cash numeric	metode_bayar character varying
1	39	77	5	2016-06-30	5	48985	0.1	0	shopeepay
2	69	65	4	2016-07-17	4	30562	0.1	0	shopeepay
3	59	16	4	2016-01-17	2	48920	0.1	0	shopeepay
4	10	34	1	2016-02-14	1	31084	0.1	0	shopeepay

To retrieve transactions specifically using ShopeePay as the payment method, you can filter the column "metode_bayar" by using metode_bayar = 'shopeepay', and to sort them by the largest transaction, you can use ORDER BY harga DESC. You can select all columns from the "order" table using SELECT *

5. The Marketing Team is planning a campaign in Tangerang City for the next 2 months. For this, they need specific data of all DataFood customer addresses located in Tangerang. To streamline their analysis process, create a new table named "customer_address_tangerang" that contains data from the "customer_address" table specifically for addresses in Tangerang city.

```
CREATE TABLE customer_address_tangerang AS
SELECT * FROM customer_address
WHERE kota = 'Tangerang';

SELECT * FROM customer_address_tangerang;
```

Data Output | Messages | Notifications

	id_alamat integer	id_pelanggan integer	alamat character varying	kota character varying	provinsi character varying
1	10	10	Daan Mogot	Tangerang	Jawa Barat
2	37	37	Daan Mogot	Tangerang	Jawa Barat
3	54	54	Daan Mogot	Tangerang	Jawa Barat
4	62	62	Daan Mogot	Tangerang	Jawa Barat

To create a table from a query, you can use the syntax CREATE TABLE <table_name> AS <query>. To obtain all address data located in Tangerang, you can apply the filter kota = 'Tangerang' on the customer_address table.

6. The Marketing Team has identified some incorrect data points in the "customer_address_tangerang" table. The province column for Tangerang city should be "Banten" (not "Jawa Barat"), and for the customer with id_pelanggan 10, the address should be in "Karawaci" (not "Daan Mogot").

```
UPDATE customer_address_tangerang
SET provinsi = 'Banten'
WHERE kota = 'Tangerang';

SELECT * FROM customer_address_tangerang;

----------------------------
UPDATE customer_address_tangerang
SET alamat = 'Karawaci'
WHERE id_pelanggan = 10;

SELECT * FROM customer_address_tangerang;
```

Data Output | Messages | Notifications

id_alamat integer	id_pelanggan integer	alamat character varying	kota character varying	provinsi character varying
37	37	Daan Mogot	Tangerang	Banten
54	54	Daan Mogot	Tangerang	Banten
62	62	Daan Mogot	Tangerang	Banten
10	10	Karawaci	Tangerang	Banten

To make changes to data in a table, you can use the UPDATE statement. The UPDATE statement is followed by the table name you want to modify, then the SET keyword to specify which columns you want to change and the new values you want to assign. The WHERE clause is used to specify conditions if you want to update specific rows; if you want to update all rows, you can omit the WHERE clause.

7. The table customer_address_tangerang doesn't have the new customer data yet. Add one row of data with the following information to the table:

- **Address ID: 101**
- **Customer ID: 70**
- **Address: Ciledug**
- **The city and province should follow the existing rows.**

```
INSERT INTO customer_address_tangerang (id_alamat, id_pelanggan, alamat, kota, provinsi)
VALUES (101, 70, 'Ciledug', 'Tangerang', 'Banten');

SELECT * FROM customer_address_tangerang;
```

Data Output | Messages | Notifications

	id_alamat integer	id_pelanggan integer	alamat character varying	kota character varying	provinsi character varying
1	37	37	Daan Mogot	Tangerang	Banten
2	54	54	Daan Mogot	Tangerang	Banten
3	62	62	Daan Mogot	Tangerang	Banten
4	10	10	Karawaci	Tangerang	Banten
5	101	70	Ciledug	Tangerang	Banten

The addition of a data row can be done using the keyword INSERT INTO <table_name> followed by VALUES (<data points>).

8. An error has been found in the customer_address_tangerang table. It appears that the data with id_alamat = 54 is invalid and was entered incorrectly. To remove this data row so that the Marketing team can use the table more effectively,

```
DELETE FROM customer_address_tangerang
WHERE id_alamat = 54;

SELECT * FROM customer_address_tangerang;
```

Data Output | Messages | Notifications

	id_alamat integer	id_pelanggan integer	alamat character varying	kota character varying	provinsi character varying
1	37	37	Daan Mogot	Tangerang	Banten
2	62	62	Daan Mogot	Tangerang	Banten
3	10	10	Karawaci	Tangerang	Banten
4	101	70	Ciledug	Tangerang	Banten

Data rows can be deleted using the DELETE FROM <table_name> keyword followed by a WHERE condition if you want to delete specific rows.

Thank You

Linkedin : https://www.linkedin.com/in/alqarani/
Github : https://github.com/jalikarani/simple-query-analysis