Nomor Kelompok : *10*

Kelas : *K01*

NIM : *18222013*
Nama : *Aththariq Lisan Q. D. S.*

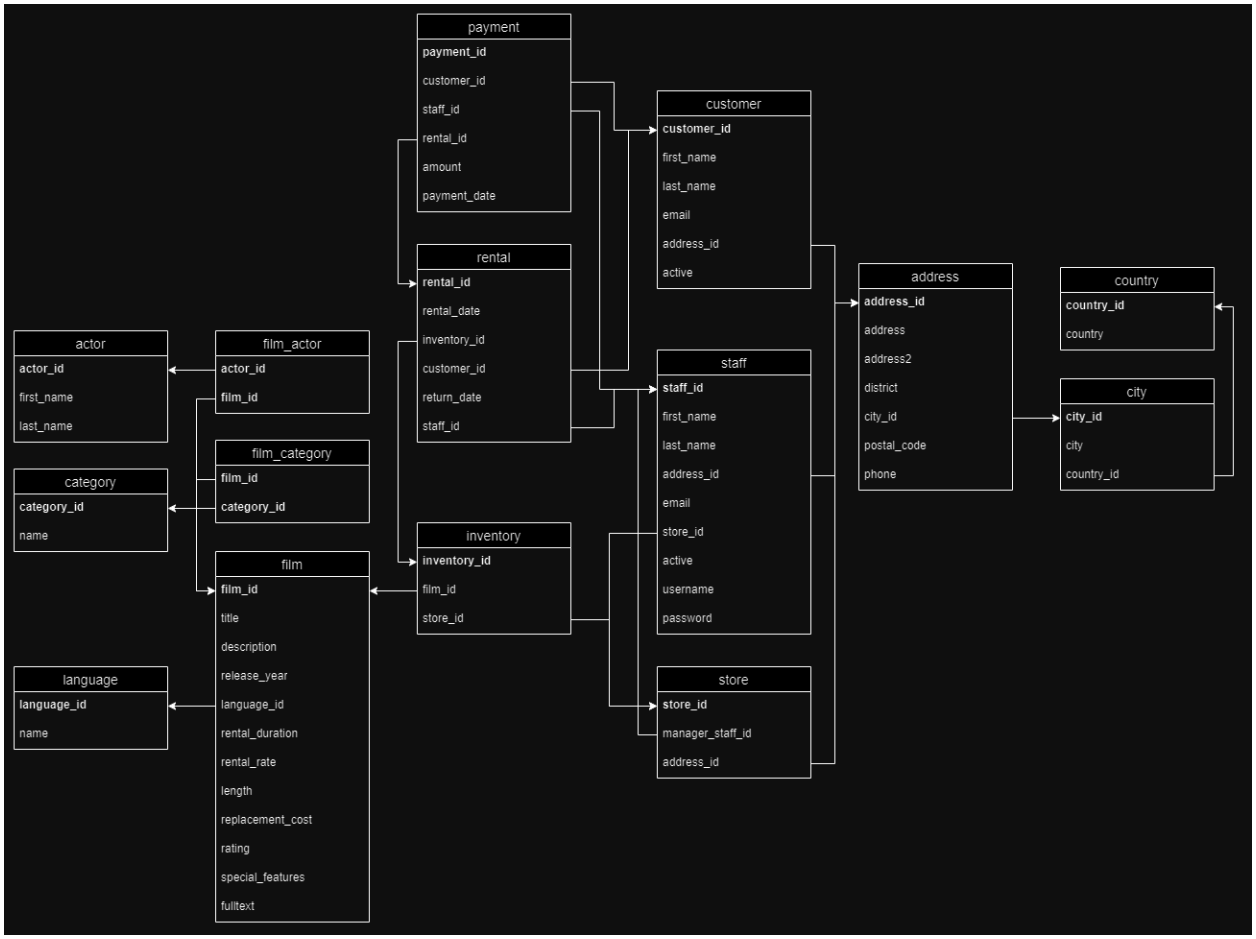
NIM : *<isi dengan NIM>*
Nama : *<isi dengan nama lengkap>*

Lembar Kerja Praktikum 2 II2250 Manajemen Basis Data STI

Materi: *Schema Tuning & Index Tuning*

I. Skema Basis Data

Diberikan skema basis data sebagai berikut yang tersimpan dalam database `pagila`.



II. Soal

(**Note**: Pastikan telah terdapat database bernama pagila di dalam komputer yang digunakan. Jika belum, buatlah sebuah database bernama pagila dan import pagila.sql ke dalam database tersebut!)

1. Function

Toko pagila mengatur harga rental film dengan ketentuan 50% dari harga rental tersebut akan digunakan sebagai biaya operasional (sewa tempat, gaji pegawai, dsb), dan sisanya sebagai keuntungan rental tiap kali dilakukan penyewaan. Rental baru bisa dikatakan menghasilkan keuntungan ketika biaya beli film (80% dari biaya penggantian film) sudah tercover dari keuntungan yang didapatkan ketika pelanggan melakukan penyewaan atau dapat disebut sebagai break even point (BEP).

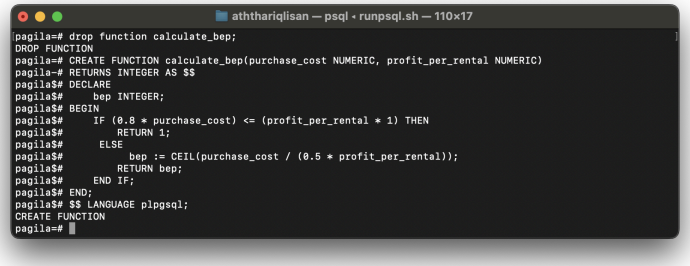
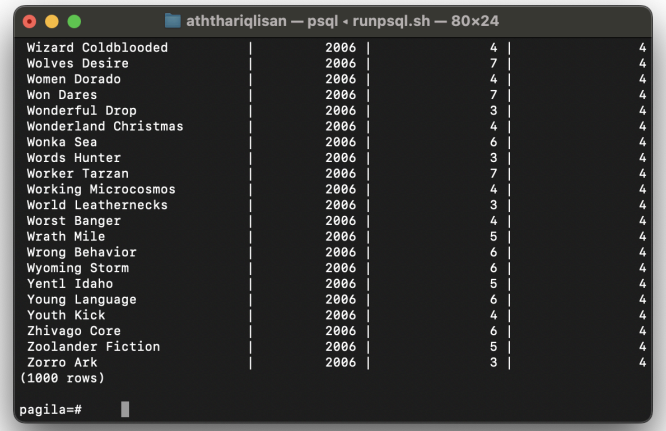
Buatlah sebuah function untuk menghitung break even point (berapa kali film disewa agar menutupi biaya pembelian film tersebut) dengan menerima argumen berupa harga beli dan keuntungan tiap melakukan rental film tersebut.

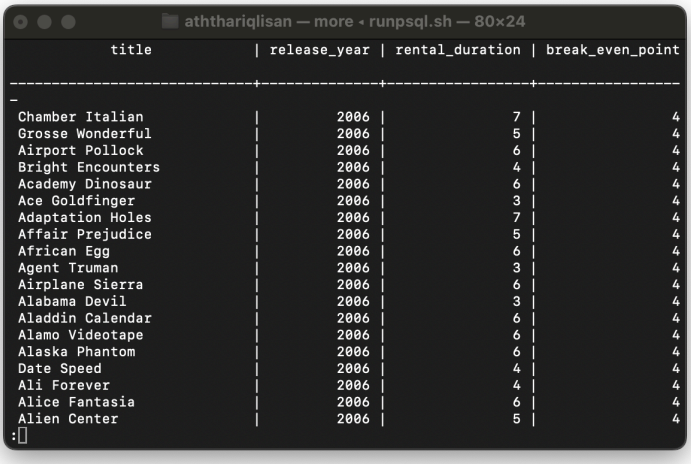
Lalu buatlah query untuk menampilkan judul film, tahun rilis, durasi rental, dan break even point (BEP) dari film tersebut dan urutkan berdasarkan BEP terkecil.

HINT : Bulatkan hasil fungsi BEP ke atas karena penyewaan hanya bisa dilakukan dalam bilangan bulat.
HINT TAMBAHAN : IF DALAM POSTGRESQL

Jawaban:

Query Pembuatan Function	CREATE FUNCTION calculate_bep(purchase_cost NUMERIC, profit_per_rental NUMERIC)
	RETURNS INTEGER AS $$
	DECLARE
	  bep INTEGER;
	BEGIN
	  IF (0.8 * purchase_cost) <= (profit_per_rental * 1) THEN
	  RETURN 1;
	  ELSE
	  bep := CEIL(purchase_cost / (0.5 * profit_per_rental));
	  RETURN bep;
	  END IF;
	END;
	$$ LANGUAGE plpgsql;

SS Query Pembuatan Function	
Query Penerapan Function	SELECT film.title, film.release_year, film.rental_duration, calculate_bep(film.replacement_cost * 0.8, film.replacement_cost * 0.5) AS break_even_point FROM film ORDER BY break_even_point;
SS Hasil Penerapan Function *boleh SS awal dan akhir row saja jika terlalu panjang	

2. Function

Toko Pagila berencana untuk menerapkan diskon pada biaya sewa film berdasarkan dua kriteria: popularitas kategori film dan nilai rating dari film tersebut. Diskon ditentukan dengan aturan berikut:

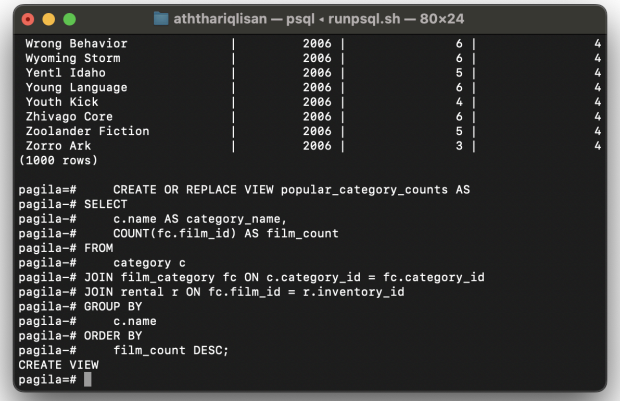
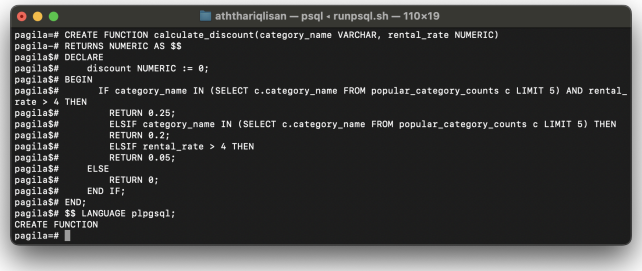
- Film yang termasuk dalam 5 besar kategori film terpopuler (kategori film yang paling banyak dipinjam) akan menerima diskon sebesar 20%.
- Film yang memiliki rental rate lebih dari 4 akan menerima diskon sebesar 5%.
- Film yang memenuhi kedua kriteria di atas akan menerima diskon sebesar 25%.

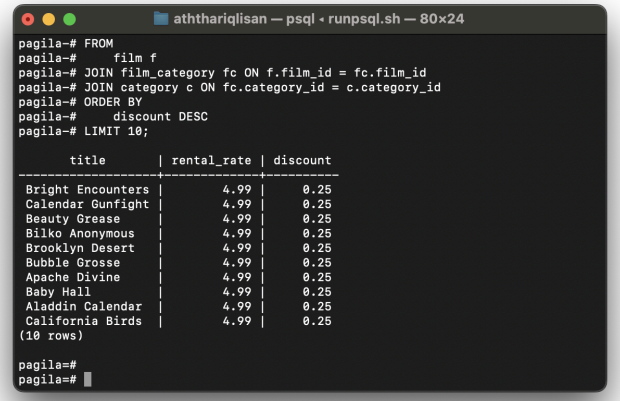
Tampilkan 10 film dengan diskon terbesar
Catatan : Diperbolehkan membuat view untuk membantu perhitungan

Jawaban:

Query Pembuatan Function	CREATE VIEW popular_category_counts AS
	SELECT
	  c.name AS category_name,
	  COUNT(fc.film_id) AS film_count
	FROM
	  category c
	JOIN film_category fc ON c.category_id = fc.category_id
	JOIN rental r ON fc.film_id = r.inventory_id
	GROUP BY

	c.name
	ORDER BY
	film_count DESC;
	
	CREATE FUNCTION
	calculate_discount(category_name VARCHAR,
	rental_rate NUMERIC)
	RETURNS NUMERIC AS $$
	DECLARE
	discount NUMERIC := 0;
	BEGIN
	IF category_name IN (SELECT c.category_name
	FROM popular_category_counts c LIMIT 5) AND
	rental_rate > 4 THEN
	RETURN 0.25;
	ELSIF category_name IN (SELECT
	c.category_name FROM popular_category_counts c
	LIMIT 5) THEN
	RETURN 0.2;
	ELSIF rental_rate > 4 THEN
	RETURN 0.05;
	ELSE
	RETURN 0;
	END IF;
	END;
	$$ LANGUAGE plpgsql;

	 
Query Penerapan Function	SELECT f.title, f.rental_rate, calculate_discount(c.name, f.rental_rate) AS discount FROM film f JOIN film_category fc ON f.film_id = fc.film_id JOIN category c ON fc.category_id = c.category_id ORDER BY discount DESC LIMIT 10;

SS Hasil Penerapan Function	

3. Function

Toko Pagila saat ini menjalin kerja sama dengan beberapa negara, diantaranya adalah India, Indonesia, dan Japan. Kerja sama ini akan berupaya untuk menyejahterakan staff Toko Pagila yang sering melayani rental dari customer yang berasal dari negara-negara tersebut dengan memberikan insentif tambahan dengan aturan sebagai berikut.

1. Setiap transaksi yang berasal dari negara India, maka staff akan diberikan insentif sebesar 1% dari amount transaksi.
2. Setiap transaksi yang berasal dari negara Indonesia, maka staff akan diberikan insentif sebesar 0,5% dari amount transaksi.
3. Setiap transaksi yang berasal dari negara Japan, maka staff akan diberikan insentif sebesar 2% dari amount transaksi.

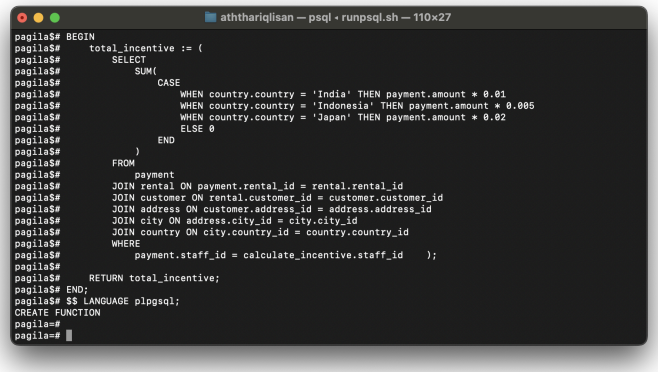
Buatlah sebuah function untuk menghitung total insentif yang didapatkan oleh setiap staff dengan menerima argumen berupa staff_id.

Lalu, buatlah query untuk menampilkan keseluruhan informasi dari tabel staff dan tampilkan juga total insentif yang mereka dapatkan.

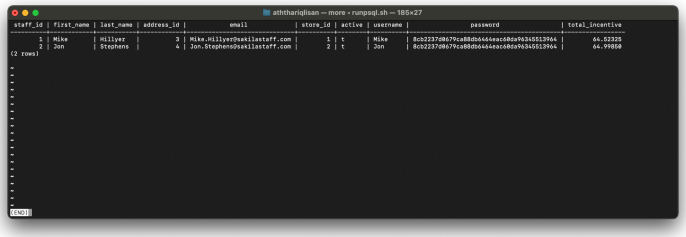
Note: Dapat menggunakan CASE WHEN

Jawaban:

Query Pembuatan Function	CREATE OR REPLACE FUNCTION calculate_incentive(staff_id INT) RETURNS NUMERIC AS $$ DECLARE

	total_incentive NUMERIC := 0;
	BEGIN
	total_incentive := (
	SELECT
	SUM(
	CASE
	WHEN country.country = 'India' THEN payment.amount * 0.01
	WHEN country.country = 'Indonesia' THEN payment.amount * 0.005
	WHEN country.country = 'Japan' THEN payment.amount * 0.02
	ELSE 0
	END
	)
	FROM
	payment
	JOIN rental ON payment.rental_id = rental.rental_id
	JOIN customer ON rental.customer_id = customer.customer_id
	JOIN address ON customer.address_id = address.address_id
	JOIN city ON address.city_id = city.city_id
	JOIN country ON city.country_id = country.country_id
	WHERE
	payment.staff_id = calculate_incentive.staff_id
	);
	
	RETURN total_incentive;
	END;
	$$ LANGUAGE plpgsql;
	
	 

Query Penerapan	SELECT

Function	staff.*, calculate_incentive(staff.staff_id) AS total_incentive FROM staff;
SS Hasil Penerapan Function	

4. **Procedure**

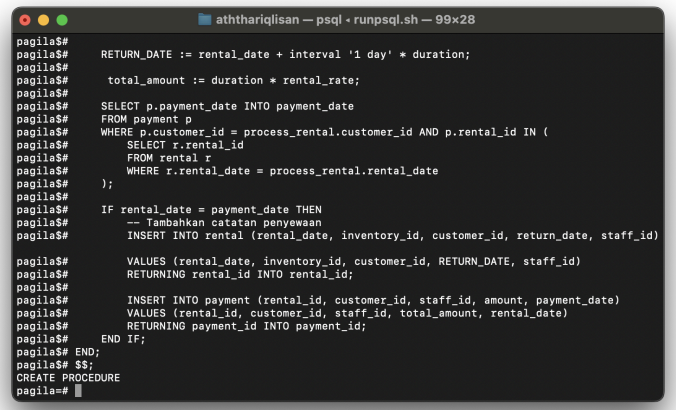
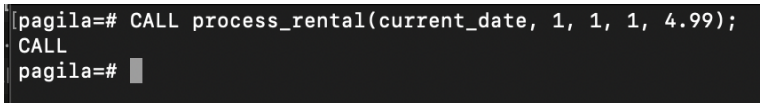
Toko Pagila ingin memperbaiki sistem penyewaan film mereka dengan menambahkan fitur untuk mengelola transaksi penyewaan film secara otomatis. Oleh karena itu, Toko Pagila ingin membuat prosedur untuk menangani transaksi penyewaan film, termasuk pembuatan catatan penyewaan dan pembayaran.

a. Buatlah prosedur untuk menangani transaksi penyewaan film. Prosedur ini harus menerima parameter seperti tanggal sewa, id inventory, id customer, id staff, durasi sewa, dan tarif sewa (rental rate). Prosedur ini harus secara otomatis menambahkan catatan penyewaan dan membuat catatan pembayaran apabila pembayaran dilakuan di depan (
b. . Jumlah yang dibayar didapatkan dari rental_duration * rental_rate
c. Lakukan panggilan pada prosedur tersebut dengan parameter yang sesuai. Gunakan tanggal hari ini sebagai rental date.
d. Buat query untuk menampilkan data dari tabel rental dan payment setelah prosedur berhasil dipanggil.

Jawaban:

Query Pembuatan Prosedur	CREATE PROCEDURE process_rental(rental_date DATE, inventory_id INT, customer_id INT, staff_id INT, rental_rate NUMERIC) LANGUAGE plpgsql AS $$ DECLARE rental_id INT; payment_id INT;

```
    total_amount NUMERIC;
    duration INT;
    RETURN_DATE DATE;
    payment_date DATE;
BEGIN
    SELECT rental_duration INTO duration
    FROM film
    WHERE film_id = inventory_id;

    RETURN_DATE := rental_date + interval '1 day' *
duration;

     total_amount := duration * rental_rate;

    SELECT p.payment_date INTO payment_date
    FROM payment p
    WHERE p.customer_id = process_rental.customer_id
AND p.rental_id IN (
        SELECT r.rental_id
        FROM rental r
        WHERE r.rental_date = process_rental.rental_date
    );

    IF rental_date = payment_date THEN
        INSERT INTO rental (rental_date, inventory_id,
customer_id, return_date, staff_id)
        VALUES (rental_date, inventory_id, customer_id,
RETURN_DATE, staff_id)
        RETURNING rental_id INTO rental_id;

        INSERT INTO payment (rental_id, customer_id,
staff_id, amount, payment_date)
        VALUES (rental_id, customer_id, staff_id,
total_amount, rental_date)
        RETURNING payment_id INTO payment_id;
    END IF;
END;
$$;
```

SS Query Pembuatan Prosedur	
Query Pemanggilan Prosedur	CALL process_rental(current_date, 1, 1, 1, 4.99); 
Query untuk Menampilkan Data & SS	SELECT * FROM rental r JOIN payment p ON r.rental_id = p.rental_id ORDER BY rental_date desc LIMIT 5;

5. Procedure

Berkurangnya attention span masyarakat masa kini membuat manajemen Toko Pagila berpikir untuk meningkatkan promosi film dengan durasi pendek. Oleh karena itu, Toko Pagila ingin menandai film-film mana yang memiliki durasi kurang dari 60 menit sebagai film pendek untuk menjadi bahan marketing.

a. Tambahkan atribut is_short_movie (boolean) pada tabel film. Dengan definisi durasi film pendek yang ditentukan manajemen, buatlah prosedur untuk memberikan value is_short_movie sesuai dengan apakah film tersebut termasuk film pendek (true) atau bukan (false).
b. Lakukan panggilan pada prosedur tersebut!
c. Buat query untuk menampilkan data dari tabel film setelah prosedur berhasil dipanggil.

Jawaban:

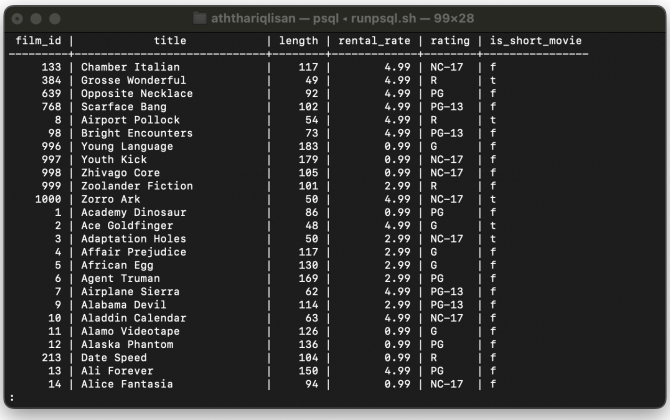
Query	ALTER TABLE film

Penambahan kolom	ADD COLUMN is_short_movie BOOLEAN;
	`pagila=# ALTER TABLE film`
	`pagila-# ADD COLUMN is_short_movie BOOLEAN;`
	`ALTER TABLE`

Query Pembuatan Prosedur	CREATE PROCEDURE short_movies()
	LANGUAGE plpgsql
	AS $$
	BEGIN
	UPDATE film
	SET is_short_movie = CASE
	WHEN length < 60 THEN true
	ELSE false
	END;
	END;
	$$;

SS Query Pembuatan Prosedur	`pagila=# CREATE PROCEDURE short_movies()`
	`pagila-# LANGUAGE plpgsql`
	`pagila-# AS $$`
	`pagila$# BEGIN`
	`pagila$#  UPDATE film`
	`pagila$#  SET is_short_movie = CASE`
	`pagila$#  WHEN length < 60 THEN true`
	`pagila$#  ELSE false`
	`pagila$#  END;`
	`pagila$# END;`
	`pagila$# $$;`
	`CREATE PROCEDURE`

Query Pemanggilan Prosedur	CALL short_movies();
	`pagila=# CALL short_movies();`
	`CALL`

Query untuk Menampilkan Data & SS	SELECT film_id, title, length, rental_rate, rating, is_short_movie FROM film; 

film_id	title	length	rental_rate	rating	is_short_movie
133	Chamber Italian	117	4.99	NC-17	f
384	Grosse Wonderful	49	4.99	R	t
639	Opposite Necklace	92	4.99	PG	f
768	Scarface Bang	102	4.99	PG-13	f
8	Airport Pollock	54	4.99	R	t
98	Bright Encounters	73	4.99	PG-13	f
996	Young Language	183	0.99	G	f
997	Youth Kick	179	0.99	NC-17	f
998	Zhivago Core	105	0.99	NC-17	f
999	Zoolander Fiction	101	2.99	R	f
1000	Zorro Ark	50	4.99	NC-17	t
1	Academy Dinosaur	86	0.99	PG	f
2	Ace Goldfinger	48	4.99	G	t
3	Adaptation Holes	50	2.99	NC-17	t
4	Affair Prejudice	117	2.99	G	f
5	African Egg	130	2.99	G	f
6	Agent Truman	169	2.99	PG	f
7	Airplane Sierra	62	4.99	PG-13	f
9	Alabama Devil	114	2.99	PG-13	f
10	Aladdin Calendar	63	4.99	NC-17	f
11	Alamo Videotape	126	0.99	G	f
12	Alaska Phantom	136	0.99	PG	f
213	Date Speed	104	0.99	R	f
13	Ali Forever	150	4.99	PG	f
14	Alice Fantasia	94	0.99	NC-17	f

:

6. Procedure

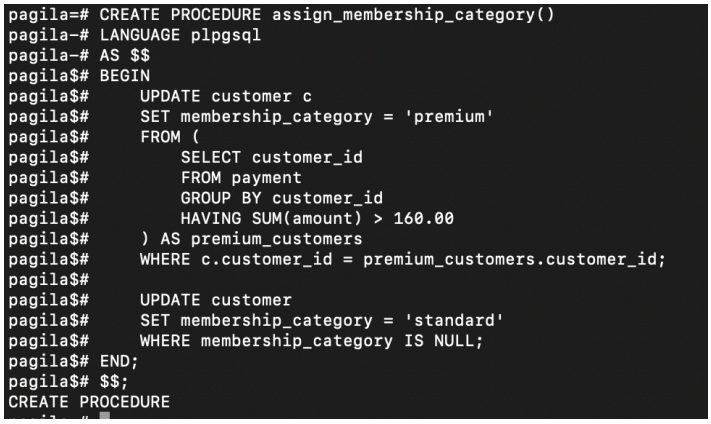
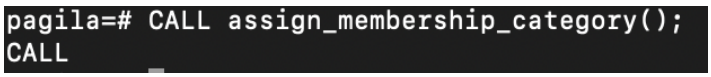
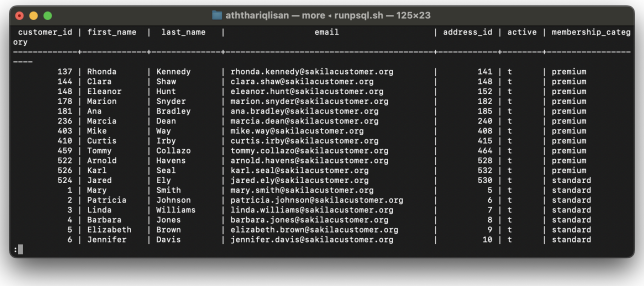
Toko Pagila ingin membuat strategi bisnis baru berupa membership untuk meningkatkan loyalitas customer mereka. Terdapat 2 kategori membership, yaitu premium dan standard. Membership premium memiliki benefit yang lebih banyak dibandingkan dengan membership standard. Untuk memudahkan pemberian benefit, toko pagila ingin melakukan penambahan atribut membership pada tabel customer. Customer yang termasuk dalam membership premium merupakan customer yang memiliki total amount payment di atas 160.00.

a. Tambahkan atribut membership_category (varchar (25)) pada tabel customer. Buatlah prosedur untuk memberikan value membership_category sesuai dengan kategori membership (premium / standard) dari customer terkait.
b. Lakukan panggilan pada prosedur tersebut.
c. Buat query untuk menampilkan data dari tabel customer setelah prosedur berhasil dipanggil. Tampilkan juga data customer dengan kategori membership premium.

Jawaban:

Query Penambahan kolom	ALTER TABLE customer
	ADD COLUMN membership_category VARCHAR(25);
	`pagila=# ALTER TABLE customer`
	`pagila-# ADD COLUMN membership_category VARCHAR(25);`
	`ALTER TABLE`

Query Pembuatan Prosedur	CREATE PROCEDURE assign_membership_category() LANGUAGE plpgsql AS $$ BEGIN UPDATE customer c SET membership_category = 'premium' FROM (SELECT customer_id FROM payment GROUP BY customer_id HAVING SUM(amount) > 160.00) AS premium_customers WHERE c.customer_id = premium_customers.customer_id; UPDATE customer SET membership_category = 'standard' WHERE membership_category IS NULL;

	END; $$;
SS Query Pembuatan Prosedur	
Query Pemanggilan Prosedur	CALL assign_membership_category(); 
Query untuk Menampilkan Data & SS	SELECT * FROM customer; 

III. Pembagian Tugas

NIM	Nama	Tugas
18222013	Aththariq Lisan Q. D. S.	