**SQL ASSIGNMENT 3**

**Dataset:** Jomato

**About the dataset:**

You work for a data analytics company, and your client is a food delivery platform similar to Jomato. They have provided you with a dataset containing information about various restaurants in a city. Your task is to analyze this dataset using SQL queries to extract valuable insights and generate reports for your client

-- Assignment 3

-- Tasks Performed:

-- 1. Create a stored procedure to display the restaurant name, type and cuisine where the table booking is not zero.

IF OBJECT\_ID('GetRestaurantsWithTableBooking', 'P') IS NOT NULL

DROP PROCEDURE GetRestaurantsWithTableBooking;

CREATE PROCEDURE GetRestaurantsWithTableBooking

AS

BEGIN

-- Display restaurant name, type, and cuisine where table booking is either 'Yes' or 'No'

SELECT

RestaurantName,

RestaurantType,

CuisinesType

FROM

Jomato$

WHERE

TableBooking IN ('Yes', 'No');

END;

CREATE PROCEDURE GetRestaurantsWithTableBooking

AS

BEGIN

-- Display restaurant name, type, and cuisine with converted TableBooking values

SELECT

RestaurantName,

RestaurantType,

CuisinesType,

CASE

WHEN TableBooking = 'Yes' THEN 1

WHEN TableBooking = 'No' THEN 0

ELSE NULL -- Handle any other cases if needed

END AS ConvertedTableBooking

FROM

Jomato$;

END;

EXEC GetRestaurantsWithTableBooking;

CREATE PROCEDURE GetRestaurantsWithTableBooking

AS

BEGIN

-- Display restaurant name, type, and cuisine where table booking is 'Yes' (1)

SELECT

RestaurantName,

RestaurantType,

CuisinesType

FROM

Jomato$

WHERE

CASE

WHEN TableBooking = 'Yes' THEN 1

ELSE 0

END <> 0;

END;

-- 2. Create a transaction and update the cuisine type ‘Cafe’ to ‘Cafeteria’. Check the result and rollback it.

-- Start the transaction

BEGIN TRANSACTION;

-- Update the cuisine type 'Cafe' to 'Cafeteria'

UPDATE Jomato$

SET CuisinesType = 'Cafeteria'

WHERE CuisinesType = 'Cafe';

-- Commit the transaction (or rollback to undo the changes)

-- Uncomment either COMMIT or ROLLBACK as needed

-- COMMIT;

-- ROLLBACK;

-- Check the updated data

SELECT \* FROM Jomato;

-- End the transaction

COMMIT;

-- 3. Generate a row number column and find the top 5 areas with the highest rating of restaurants.

WITH RankedRestaurants AS (

SELECT

Area,

RestaurantName,

Rating,

ROW\_NUMBER() OVER (PARTITION BY Area ORDER BY Rating DESC) AS RowNum

FROM

Jomato$

)

SELECT

Area,

RestaurantName,

Rating

FROM

RankedRestaurants

WHERE

RowNum <= 5;

-- 4. Use the while loop to display the 1 to 50.

DECLARE @Counter INT = 1;

WHILE @Counter <= 50

BEGIN

PRINT @Counter;

SET @Counter = @Counter + 1;

END;

-- 5. Write a query to Create a Top rating view to store the generated top 5 highest rating of restaurants.

CREATE VIEW TopRatingView AS

WITH RankedRestaurants AS (

SELECT

Area,

RestaurantName,

Rating,

ROW\_NUMBER() OVER (PARTITION BY Area ORDER BY Rating DESC) AS RowNum

FROM

Jomato$

)

SELECT

Area,

RestaurantName,

Rating

FROM

RankedRestaurants

WHERE

RowNum <= 5;

SELECT \* FROM TopRatingView;

-- 6. Write a trigger that sends an email notification to the restaurant owner whenever a new record is inserted.

CREATE TRIGGER trgAfterInsert

ON Jomato$

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

DECLARE @RestaurantName NVARCHAR(255);

-- Assuming the restaurant owner's email is stored in the 'EmailAddress' column

DECLARE @EmailAddress NVARCHAR(255);

SELECT

@RestaurantName = i.RestaurantName,

@EmailAddress = i.EmailAddress -- Adjust this column name based on your schema

FROM

inserted i;

-- Send email notification

EXEC msdb.dbo.sp\_send\_dbmail

@profile\_name = 'YourMailProfile', -- Replace with your mail profile name

@recipients = @EmailAddress,

@subject = 'New Restaurant Record Inserted',

@body = 'A new record has been inserted for the restaurant: ' + @RestaurantName;

END;

CREATE TRIGGER trgAfterInsert

ON Jomato$

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

DECLARE @RestaurantName NVARCHAR(255);

DECLARE @EmailAddress NVARCHAR(255);

SELECT

@RestaurantName = i.RestaurantName,

@EmailAddress = i.EmailAddress

FROM

inserted i;

-- Send email notification

DECLARE @Subject NVARCHAR(255) = 'New Restaurant Record Inserted';

DECLARE @Body NVARCHAR(MAX) = 'A new record has been inserted for the restaurant: ' + @RestaurantName;

EXEC msdb.dbo.sp\_send\_dbmail

@profile\_name = 'YourMailProfile', -- Replace with your mail profile name

@recipients = @EmailAddress,

@subject = @Subject,

@body = @Body;

END;

ALTER TABLE Jomato$ ADD EmailAddress NVARCHAR(255);