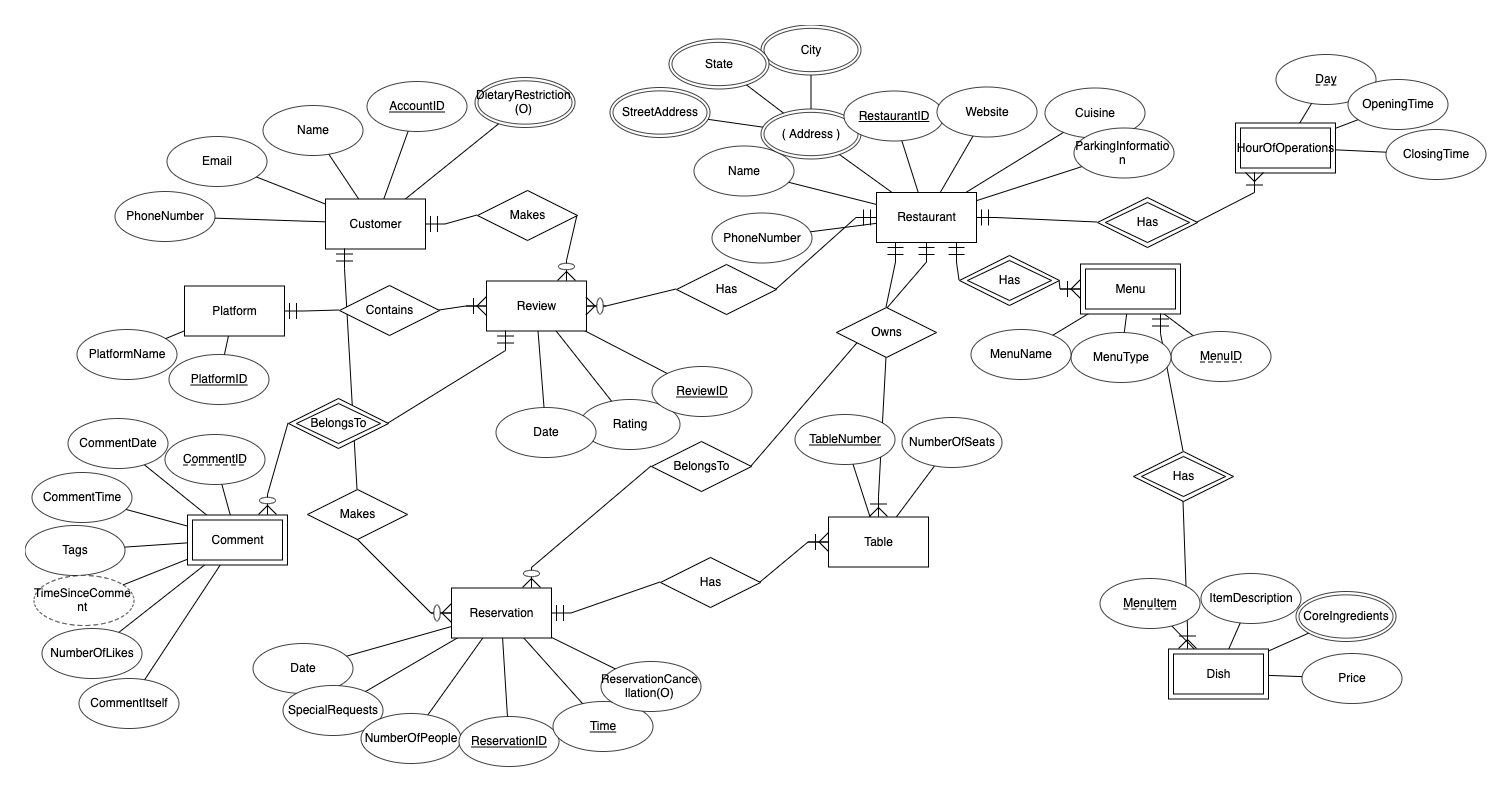
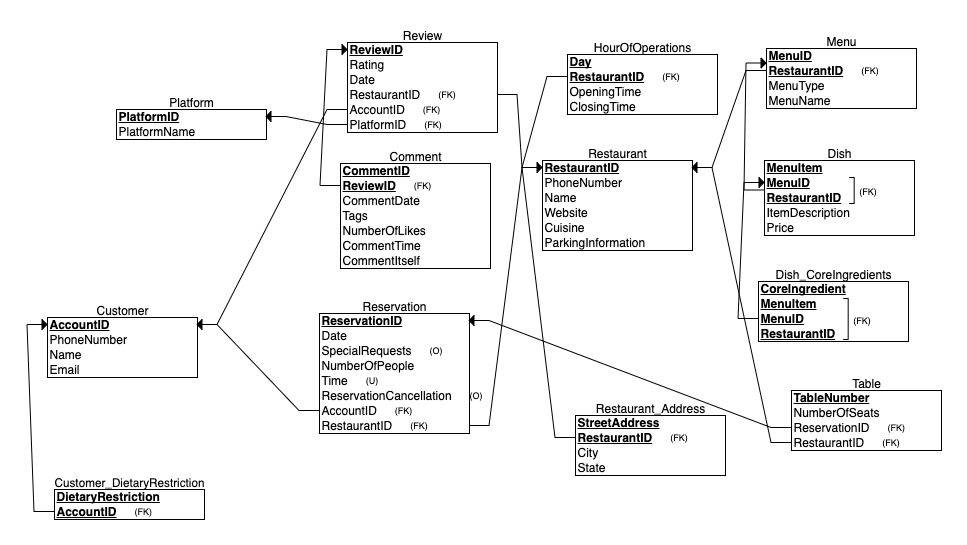
**Tech Tables**

**Part 1)**

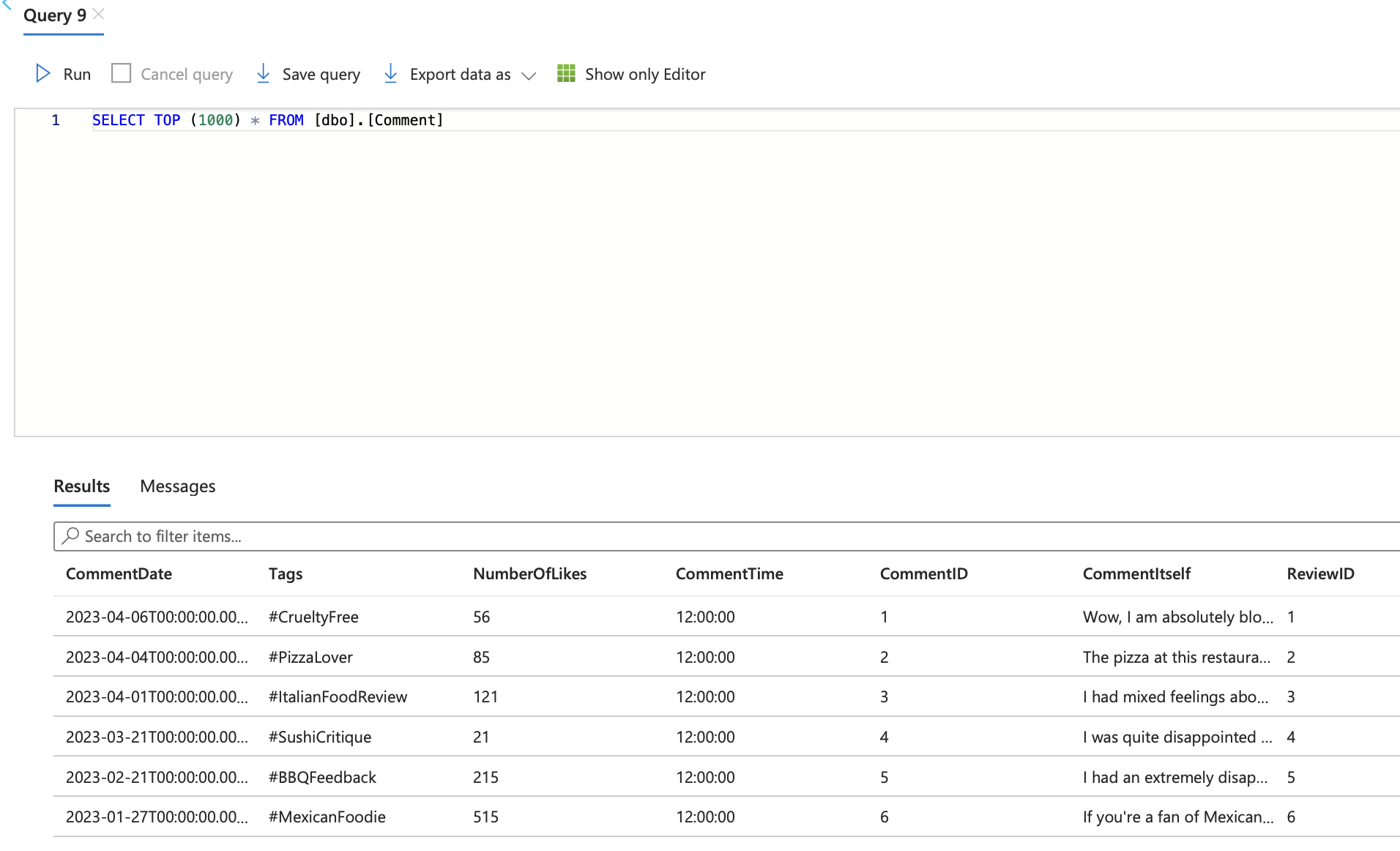
1.

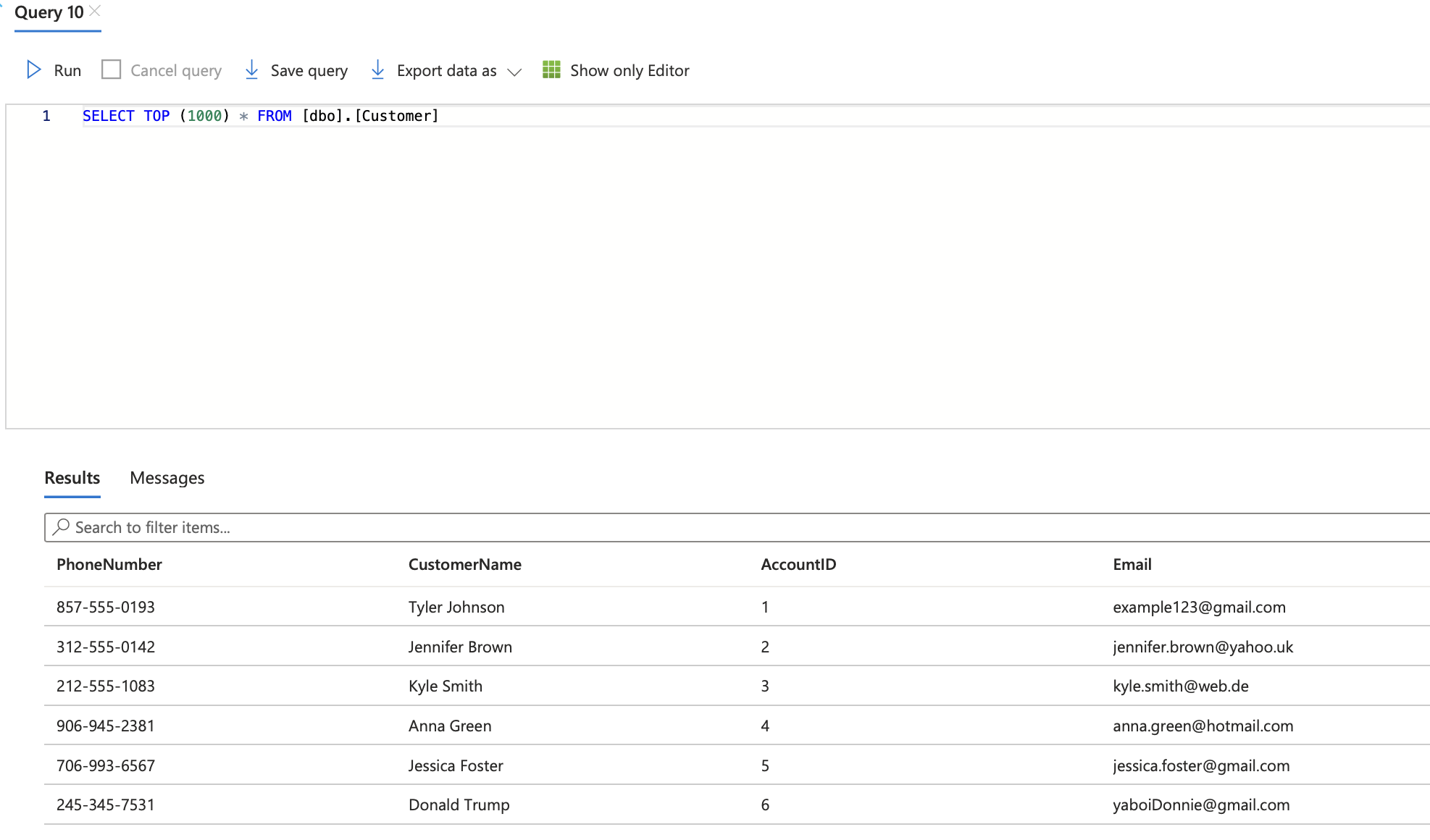


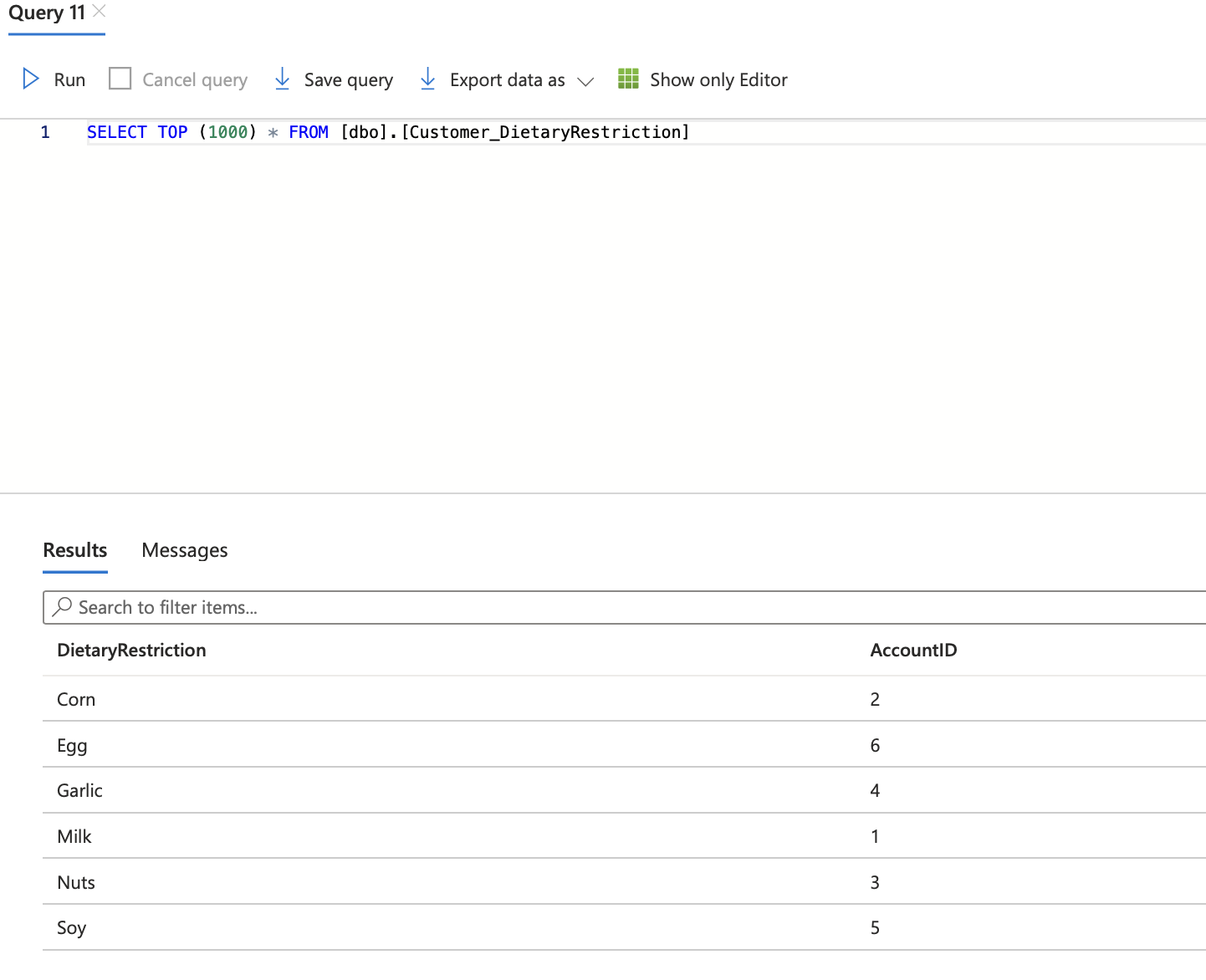


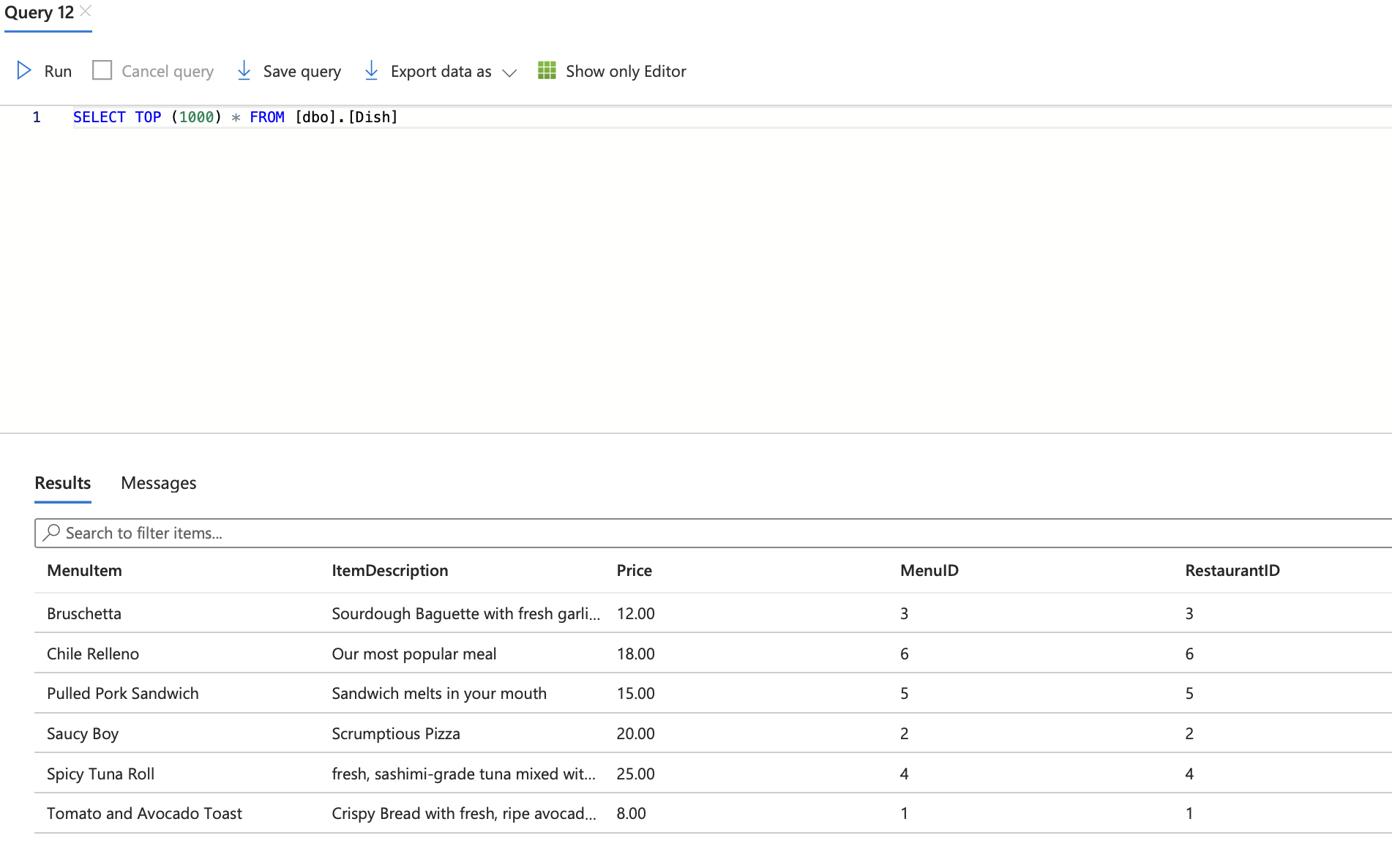
|  |
| --- |
| CREATE TABLE Customer  (  PhoneNumber VARCHAR(15),  CustomerName VARCHAR(50),  AccountID INT NOT NULL,  Email VARCHAR(255),  PRIMARY KEY (AccountID)  );  CREATE TABLE Restaurant  (  PhoneNumber VARCHAR(15),  RestaurantName VARCHAR(50),  RestaurantID INT NOT NULL,  Website VARCHAR(255),  Cuisine VARCHAR(50),  ParkingInformation VARCHAR(1000),  PRIMARY KEY (RestaurantID)  );  CREATE TABLE Platform  (  PlatformID INT NOT NULL,  PlatformName VARCHAR(50),  PRIMARY KEY (PlatformID)  );  CREATE TABLE HourOfOperations  (  Specificday VARCHAR(255) NOT NULL,  OpeningTime Time,  ClosingTime Time,  RestaurantID INT NOT NULL,  PRIMARY KEY (Specificday, RestaurantID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)  );  CREATE TABLE Customer\_DietaryRestriction  (  DietaryRestriction VARCHAR(50) NOT NULL,  AccountID INT NOT NULL,  PRIMARY KEY (DietaryRestriction, AccountID),  FOREIGN KEY (AccountID) REFERENCES Customer(AccountID)  );  CREATE TABLE Restaurant\_Address  (  StreetAddress VARCHAR(255) NOT NULL,  City VARCHAR(100),  Thestate VARCHAR(50),  RestaurantID INT NOT NULL,  PRIMARY KEY (StreetAddress, RestaurantID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)  );  CREATE TABLE Review  (  Rating INT,  ReviewID INT NOT NULL,  Reviewdate smalldatetime,  RestaurantID INT NOT NULL,  AccountID INT NOT NULL,  PlatformID INT NOT NULL,  PRIMARY KEY (ReviewID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID),  FOREIGN KEY (AccountID) REFERENCES Customer(AccountID),  FOREIGN KEY (PlatformID) REFERENCES Platform(PlatformID)  );  CREATE TABLE Menu  (  MenuID INT NOT NULL,  MenuType VARCHAR(50),  MenuName VARCHAR(50),  RestaurantID INT NOT NULL,  PRIMARY KEY (MenuID, RestaurantID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)  );  CREATE TABLE Reservation  (  Reservationdate date,  SpecialRequests VARCHAR(255),  NumberOfPeople INT,  ReservationID INT NOT NULL,  Reservationtime time NOT NULL,  ReservationCancellation CHAR(1),  AccountID INT NOT NULL,  RestaurantID INT NOT NULL,  PRIMARY KEY (ReservationID),  FOREIGN KEY (AccountID) REFERENCES Customer(AccountID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID),  UNIQUE (Reservationtime)  );  CREATE TABLE Dish  (  MenuItem VARCHAR(100) NOT NULL,  ItemDescription VARCHAR(255),  Price NUMERIC(8,2),  MenuID INT NOT NULL,  RestaurantID INT NOT NULL,  PRIMARY KEY (MenuItem, MenuID, RestaurantID),  FOREIGN KEY (MenuID, RestaurantID) REFERENCES Menu(MenuID, RestaurantID)  );  CREATE TABLE Dish\_CoreIngredients  (  CoreIngredient VARCHAR(50) NOT NULL,  MenuItem VARCHAR(100) NOT NULL,  MenuID INT NOT NULL,  RestaurantID INT NOT NULL,  PRIMARY KEY (CoreIngredient, MenuItem, MenuID, RestaurantID),  FOREIGN KEY (MenuItem, MenuID, RestaurantID) REFERENCES Dish(MenuItem, MenuID, RestaurantID)  );  CREATE TABLE Comment  (  CommentDate date,  Tags VARCHAR(100),  NumberOfLikes INT,  CommentTime time,  CommentID INT NOT NULL,  CommentItself VARCHAR(1000),  ReviewID INT NOT NULL,  PRIMARY KEY (CommentID, ReviewID),  FOREIGN KEY (ReviewID) REFERENCES Review(ReviewID)  );  CREATE TABLE Tablefacts  (  TableNumber INT NOT NULL,  NumberOfSeats INT,  ReservationID INT NOT NULL,  RestaurantID INT NOT NULL,  PRIMARY KEY (TableNumber),  FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID),  FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)  ); |

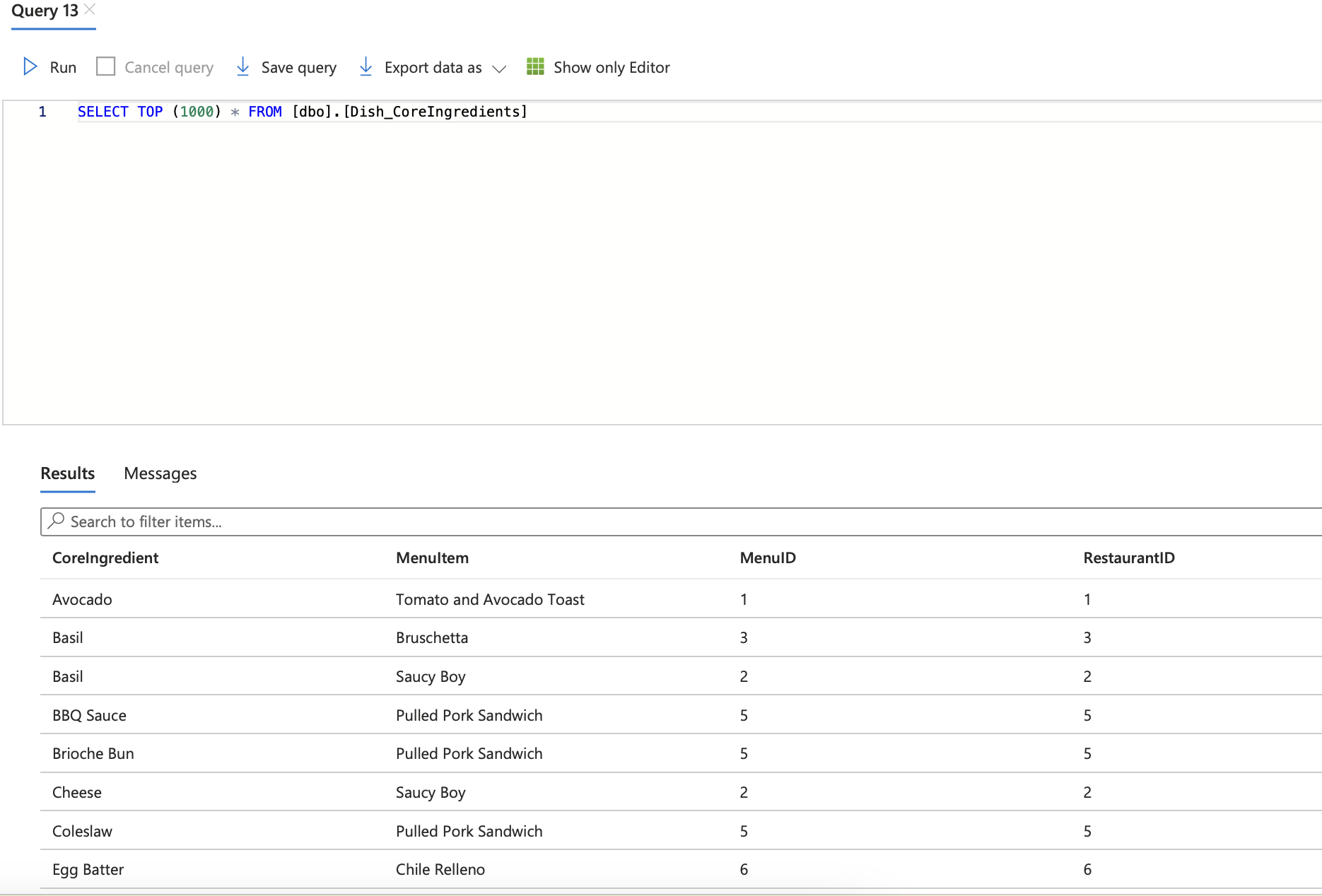
|  |
| --- |
| INSERT INTO CUSTOMER VALUES ('857-555-0193', 'Tyler Johnson', 0001, 'example123@gmail.com')  INSERT INTO CUSTOMER VALUES ('312-555-0142', 'Jennifer Brown', 0002, 'jennifer.brown@yahoo.uk')  INSERT INTO CUSTOMER VALUES ('212-555-1083', 'Kyle Smith', 0003, 'kyle.smith@web.de')  INSERT INTO CUSTOMER VALUES ('906-945-2381', 'Anna Green', 0004, 'anna.green@hotmail.com')  INSERT INTO CUSTOMER VALUES ('706-993-6567', 'Jessica Foster', 0005, 'jessica.foster@gmail.com')  INSERT INTO CUSTOMER VALUES ('245-345-7531', 'Donald Trump', 0006, 'yaboiDonnie@gmail.com')  INSERT INTO Restaurant VALUES ('617-555-0135', 'The Green Elephant', 1, 'www.greenelephant.com', 'Vegan', 'Street')  INSERT INTO Restaurant VALUES ('312-567-0142', 'Big Tony''s Pizza', 2, 'www.bigtonyspizza.com', 'Italian', 'Private Lot')  INSERT INTO Restaurant VALUES ('212-521-0183', 'Mama Mia''s Trattoria', 3, 'www.mamamiastrattoria.com', 'Italian', 'Valet')  INSERT INTO Restaurant VALUES ('317-489-9820', 'Sushi House', 4, 'www.sushihouse.com', 'Japanese', 'Garage')  INSERT INTO Restaurant VALUES ('713-995-0125', 'Georgia BBQ Company', 5, 'www.georgiabbqco.com', 'Barbecue', 'Parking Lot')  INSERT INTO Restaurant VALUES ('606-534-0198', 'El Gran Burrito', 6, 'www.elgranburrito.com', 'Mexican', 'Street Parking Available')  INSERT INTO Platform VALUES (1, 'Yelp')  INSERT INTO Platform VALUES (2, 'Google Reviews')  INSERT INTO Platform VALUES (3, 'Trip Advisor')  INSERT INTO Platform VALUES (4, 'Zomato')  INSERT INTO Platform VALUES (5, 'OpenTable')  INSERT INTO Platform VALUES (6, 'Facebook')  INSERT INTO HourOfOperations VALUES ('Normal', '12:00:00.00', '15:30:00.00', 1)  INSERT INTO HourOfOperations VALUES ('Normal', '19:00:00.00', '23:30:00.00', 2)  INSERT INTO HourOfOperations VALUES ('Normal', '18:00:00.00', '22:00:00.00', 3)  INSERT INTO HourOfOperations VALUES ('Normal', '13:00:00.00', '15:00:00.00', 4)  INSERT INTO HourOfOperations VALUES ('Normal', '11:00:00.00', '14:00:00.00', 5)  INSERT INTO HourOfOperations VALUES ('Normal', '17:30:00.00', '22:00:00.00', 6)  INSERT INTO Customer\_DietaryRestriction VALUES ('Milk', 1)  INSERT INTO Customer\_DietaryRestriction VALUES ('Corn', 2)  INSERT INTO Customer\_DietaryRestriction VALUES ('Nuts', 3)  INSERT INTO Customer\_DietaryRestriction VALUES ('Garlic', 4)  INSERT INTO Customer\_DietaryRestriction VALUES ('Soy', 5)  INSERT INTO Customer\_DietaryRestriction VALUES ('Egg', 6)  INSERT INTO Restaurant\_Address VALUES ('123 Peachtree St NE', 'Atlanta', 'Georgia', 1)  INSERT INTO Restaurant\_Address VALUES ('456 Edgewood Avenue SE', 'Atlanta', 'Georgia', 2)  INSERT INTO Restaurant\_Address VALUES ('789 West Peachtree St NW', 'Atlanta', 'Georgia', 3)  INSERT INTO Restaurant\_Address VALUES ('101 Marietta St NW', 'Atlanta', 'Georgia', 4)  INSERT INTO Restaurant\_Address VALUES ('222 Mitchell St SW', 'Atlanta', 'Georgia', 5)  INSERT INTO Restaurant\_Address VALUES ('333 Auburn Avenue NE', 'Atlanta', 'Georgia', 6)  INSERT INTO Review VALUES (5, 1, '04-06-2023 12:00:00.00', 1, 1, 1)  INSERT INTO Review VALUES (4, 2, '04-04-2023 12:00:00.00', 2, 2, 2)  INSERT INTO Review VALUES (3, 3, '04-01-2023 12:00:00.00', 3, 3, 3)  INSERT INTO Review VALUES (2, 4, '03-21-2023 12:00:00.00', 4, 4, 4)  INSERT INTO Review VALUES (1, 5, '02-21-2023 12:00:00.00', 5, 5, 5)  INSERT INTO Review VALUES (5, 6, '01-27-2023 12:00:00.00', 6, 6, 6)  INSERT INTO Menu VALUES (1, 'Lunch', 'The Green Elephant', 1)  INSERT INTO Menu VALUES (2, 'Dinner', 'Big Tony''s Pizza', 2)  INSERT INTO Menu VALUES (3, 'Dinner', 'Mama Mia''s Trattoria', 3)  INSERT INTO Menu VALUES (4, 'Lunch', 'Sushi House', 4)  INSERT INTO Menu VALUES (5, 'Lunch', 'Georgia BBQ Company', 5)  INSERT INTO Menu VALUES (6, 'Dinner', 'El Gran Burrito', 6)  INSERT INTO Reservation VALUES ('2023-04-06','We are having a birthday', 4, 1, '13:00:00.00', 'N', 1, 1)  INSERT INTO Reservation VALUES ('2023-04-06','We would like a booth', 5, 2, '20:00:00.00', 'N', 2, 2)  INSERT INTO Reservation VALUES ('2023-04-06','We would like a table', 6, 3, '18:00:00.00', 'N', 3, 3)  INSERT INTO Reservation VALUES ('2023-04-06','We would like to have Mark as our waiter', 7, 4, '14:00:00.00', 'N', 4, 4)  INSERT INTO Reservation VALUES ('2023-04-06','We would like a table with a view', 8, 5, '11:00:00.00', 'N', 5, 5)  INSERT INTO Reservation VALUES ('2023-04-06','It is our anniversary', 2, 6, '21:00:00.00', 'Y', 6, 6)  INSERT INTO Dish VALUES ('Tomato and Avocado Toast', 'Crispy Bread with fresh, ripe avocados and tomatoes', 8, 1, 1)  INSERT INTO Dish VALUES ('Saucy Boy', 'Scrumptious Pizza', 20, 2, 2)  INSERT INTO Dish VALUES ('Bruschetta', 'Sourdough Baguette with fresh garlic, tomatoes and basil', 12, 3, 3)  INSERT INTO Dish VALUES ('Spicy Tuna Roll', 'fresh, sashimi-grade tuna mixed with spicy mayonnaise and rolled inside a sheet of nori seaweed with sushi rice', 25, 4, 4)  INSERT INTO Dish VALUES ('Pulled Pork Sandwich', 'Sandwich melts in your mouth', 15, 5, 5)  INSERT INTO Dish VALUES ('Chile Relleno', 'Our most popular meal', 18, 6, 6)  INSERT INTO Dish\_CoreIngredients VALUES ('Tomato', 'Tomato and Avocado Toast', 1, 1)  INSERT INTO Dish\_CoreIngredients VALUES ('Avocado', 'Tomato and Avocado Toast', 1, 1)  INSERT INTO Dish\_CoreIngredients VALUES ('Vegan Bread', 'Tomato and Avocado Toast', 1, 1)  INSERT INTO Dish\_CoreIngredients VALUES ('Pizza Dough', 'Saucy Boy', 2, 2)  INSERT INTO Dish\_CoreIngredients VALUES ('Tomato Sauce', 'Saucy Boy', 2, 2)  INSERT INTO Dish\_CoreIngredients VALUES ('Cheese', 'Saucy Boy', 2, 2)  INSERT INTO Dish\_CoreIngredients VALUES ('Basil', 'Saucy Boy', 2, 2)  INSERT INTO Dish\_CoreIngredients VALUES ('SourDough Baguette', 'Bruschetta', 3, 3)  INSERT INTO Dish\_CoreIngredients VALUES ('Garlic', 'Bruschetta', 3, 3)  INSERT INTO Dish\_CoreIngredients VALUES ('Tomato', 'Bruschetta', 3, 3)  INSERT INTO Dish\_CoreIngredients VALUES ('Basil', 'Bruschetta', 3, 3)  INSERT INTO Dish\_CoreIngredients VALUES ('Sushi Rice', 'Spicy Tuna Roll', 4, 4)  INSERT INTO Dish\_CoreIngredients VALUES ('Seaweed Sheets', 'Spicy Tuna Roll', 4, 4)  INSERT INTO Dish\_CoreIngredients VALUES ('Tuna', 'Spicy Tuna Roll', 4, 4)  INSERT INTO Dish\_CoreIngredients VALUES ('Spicy Mayonnaise', 'Spicy Tuna Roll', 4, 4)  INSERT INTO Dish\_CoreIngredients VALUES ('Pulled Pork', 'Pulled Pork Sandwich', 5, 5)  INSERT INTO Dish\_CoreIngredients VALUES ('BBQ Sauce', 'Pulled Pork Sandwich', 5, 5)  INSERT INTO Dish\_CoreIngredients VALUES ('Brioche Bun', 'Pulled Pork Sandwich', 5, 5)  INSERT INTO Dish\_CoreIngredients VALUES ('Coleslaw', 'Pulled Pork Sandwich', 5, 5)  INSERT INTO Dish\_CoreIngredients VALUES ('Poblano Pepper', 'Chile Relleno', 6, 6)  INSERT INTO Dish\_CoreIngredients VALUES ('Ground Beef', 'Chile Relleno', 6, 6)  INSERT INTO Dish\_CoreIngredients VALUES ('Egg Batter', 'Chile Relleno', 6, 6)  INSERT INTO Dish\_CoreIngredients VALUES ('Tomato Sauce', 'Chile Relleno', 6, 6)  INSERT INTO Comment VALUES ('2023-04-06', '#CrueltyFree', 56, '12:00:00.00', 1, 'Wow, I am absolutely blown away by the  delicious flavors and creative dishes at this vegan restaurant!  From the mouth-watering appetizers to the satisfying entrees and decadent desserts, every bite was a culinary delight.  I especially appreciate the commitment to using fresh, plant-based ingredients and the attention to detail in presentation.  A must-visit for any foodie, vegan or not!', 1)  INSERT INTO Comment VALUES ('2023-04-04', '#PizzaLover', 85, '12:00:00.00', 2, 'The pizza at this restaurant is simply  outstanding! From the perfectly crispy crust to the generous toppings and gooey cheese, every slice is a flavor explosion.  I love that they offer a variety of creative options, as well as classic favorites, and everything is made with high-quality  ingredients. The atmosphere is cozy and inviting, making it the perfect spot for a casual dinner with friends or family.  I highly recommend this pizza joint!', 2)  INSERT INTO Comment VALUES ('2023-04-01', '#ItalianFoodReview', 121, '12:00:00.00', 3, 'I had mixed feelings about this Italian restaurant.  While the ambiance was lovely and the service was friendly, the food was just average. The pasta dishes were a bit underwhelming,  lacking the depth of flavor and authenticity that I was expecting. The prices were a bit high for the quality of the dishes.  Overall, a decent option if you''re in the area, but not a standout in terms of Italian cuisine.', 3)  INSERT INTO Comment VALUES ('2023-03-21', '#SushiCritique', 21, '12:00:00.00', 4, 'I was quite disappointed with my experience at this sushi  restaurant. The sushi rolls were poorly made and fell apart easily, and the fish tasted a bit off. The service was slow and inattentive,  and the ambiance was uninviting. The prices were high for the quality of the food. Overall, I would not recommend this sushi restaurant.', 4)  INSERT INTO Comment VALUES ('2023-02-21', '#BBQFeedback', 215, '12:00:00.00', 5, 'I had an extremely disappointing experience at this BBQ  restaurant. The meat was dry, tough, and lacked flavor, and the sides were lackluster at best. The service was slow and inattentive, and the  restaurant was not clean. The prices were high for the quality of the food. Overall, a terrible dining experience that I would not recommend  to anyone.', 5)  INSERT INTO Comment VALUES ('2023-01-27', '#MexicanFoodie', 515, '12:00:00.00', 6, 'If you''re a fan of Mexican food, you simply have to try this  restaurant! I was blown away by the delicious flavors and vibrant atmosphere. The menu offers a fantastic variety of dishes, from classic favorites  like tacos and burritos to unique specialties that you won''t find anywhere else. The quality of the food is simply outstanding, with each dish perfectly  seasoned and bursting with authentic Mexican flavors. The service was friendly and attentive, and the prices were very reasonable. Overall, a fantastic  dining experience that I would highly recommend!', 6)  INSERT INTO Tablefacts VALUES (1, 4, 1, 1)  INSERT INTO Tablefacts VALUES (2, 5, 2, 2)  INSERT INTO Tablefacts VALUES (3, 6, 3, 3)  INSERT INTO Tablefacts VALUES (4, 7, 4, 4)  INSERT INTO Tablefacts VALUES (5, 8, 5, 5)  INSERT INTO Tablefacts VALUES (6, 2, 6, 6) |

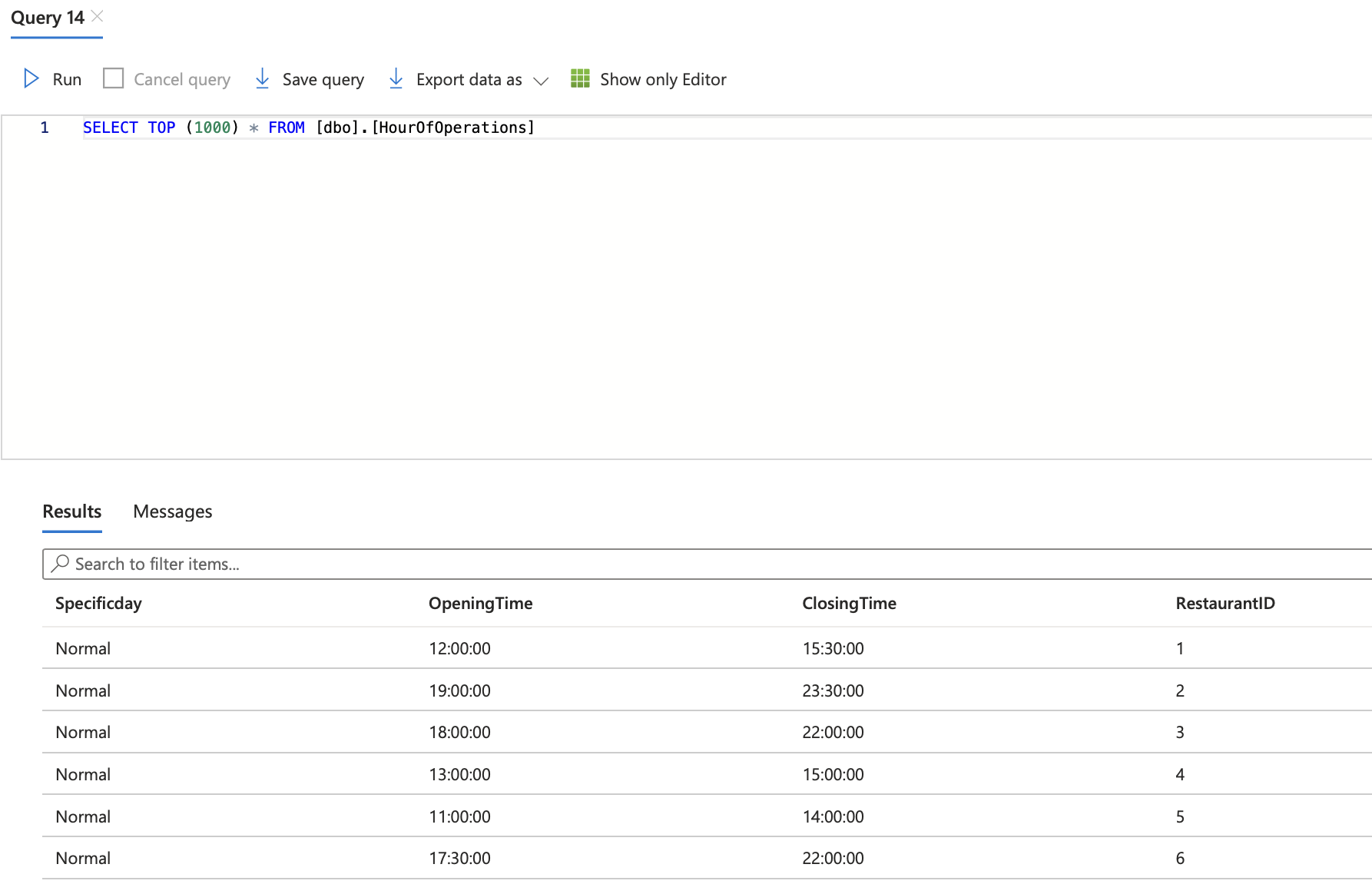
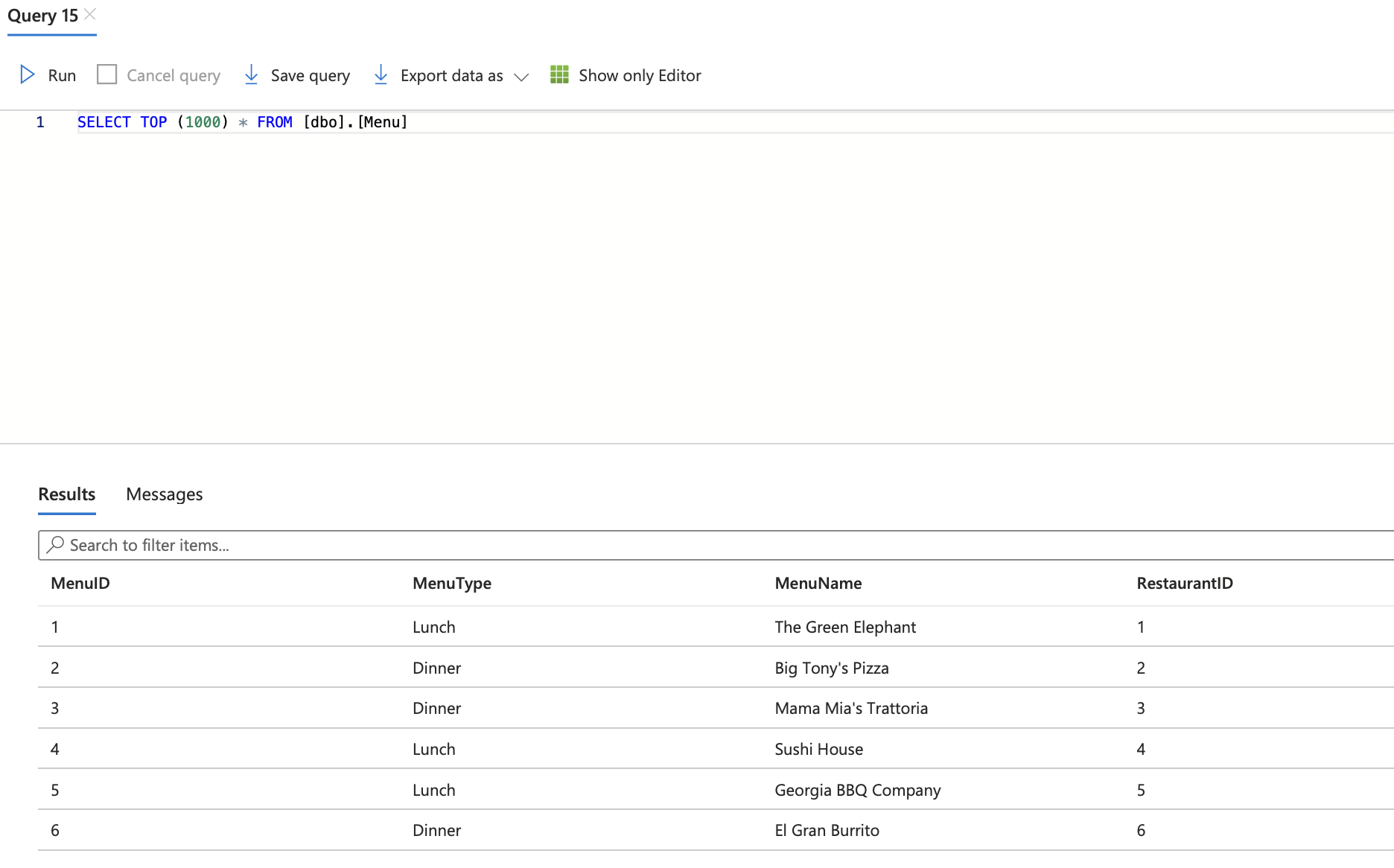


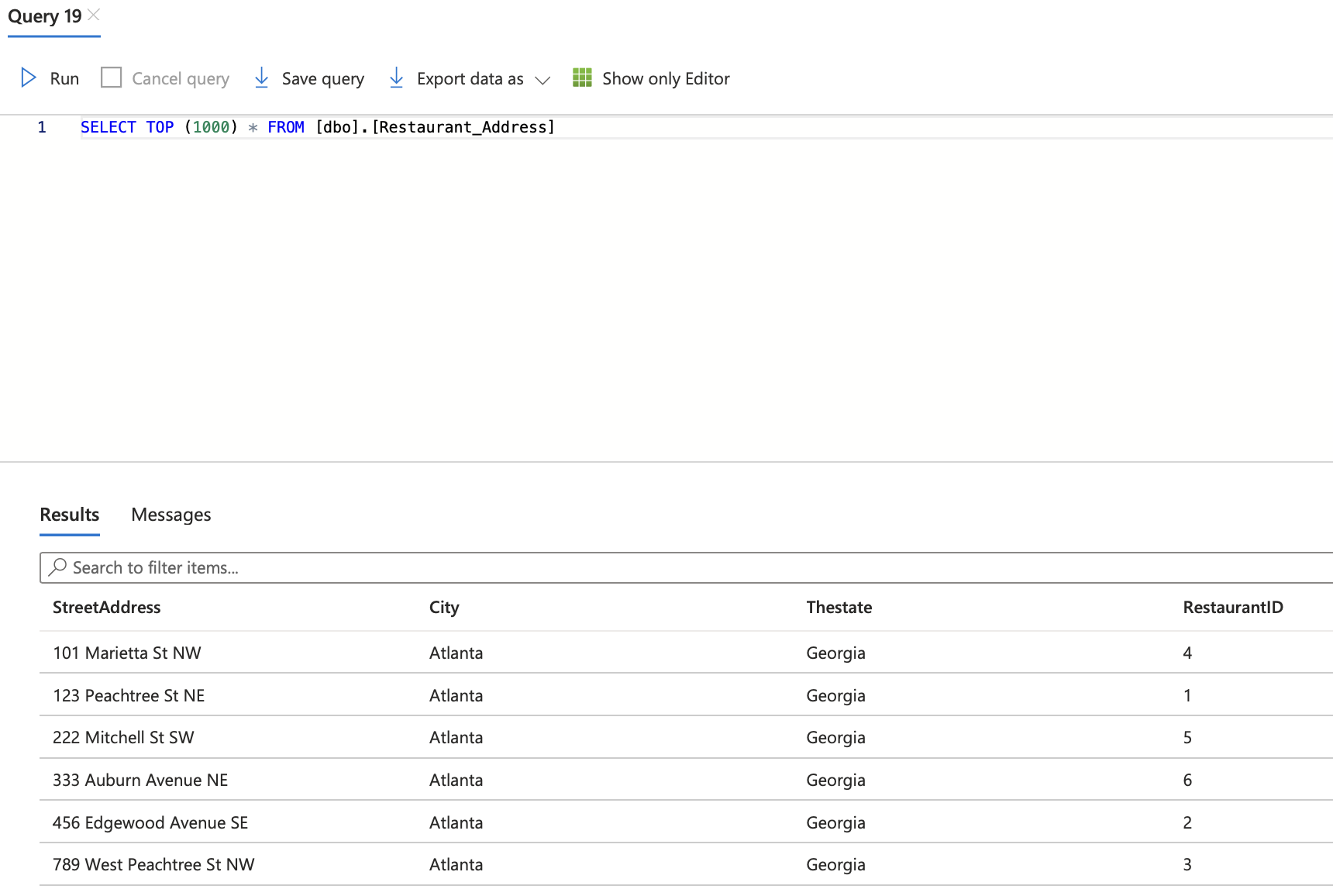
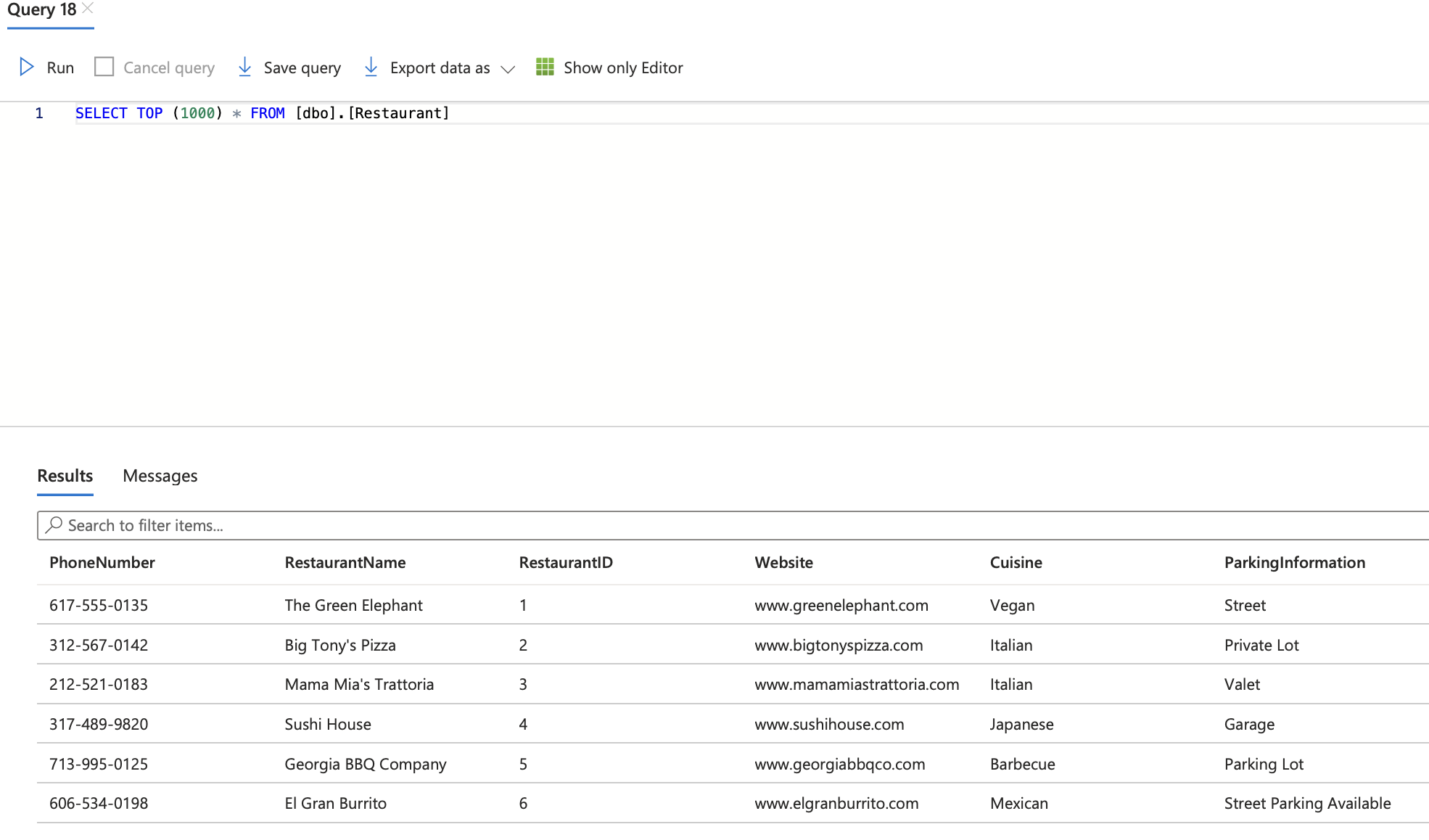
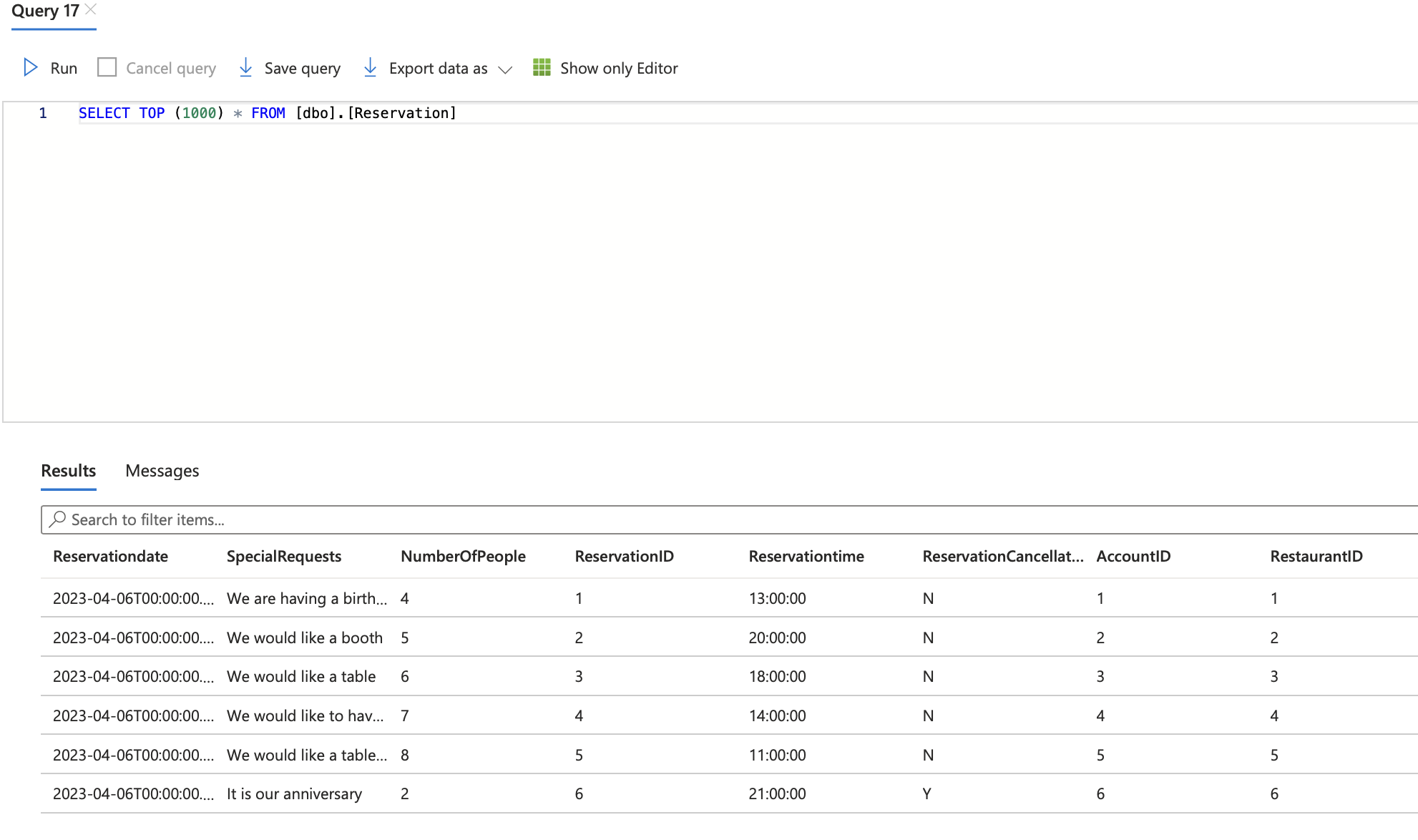
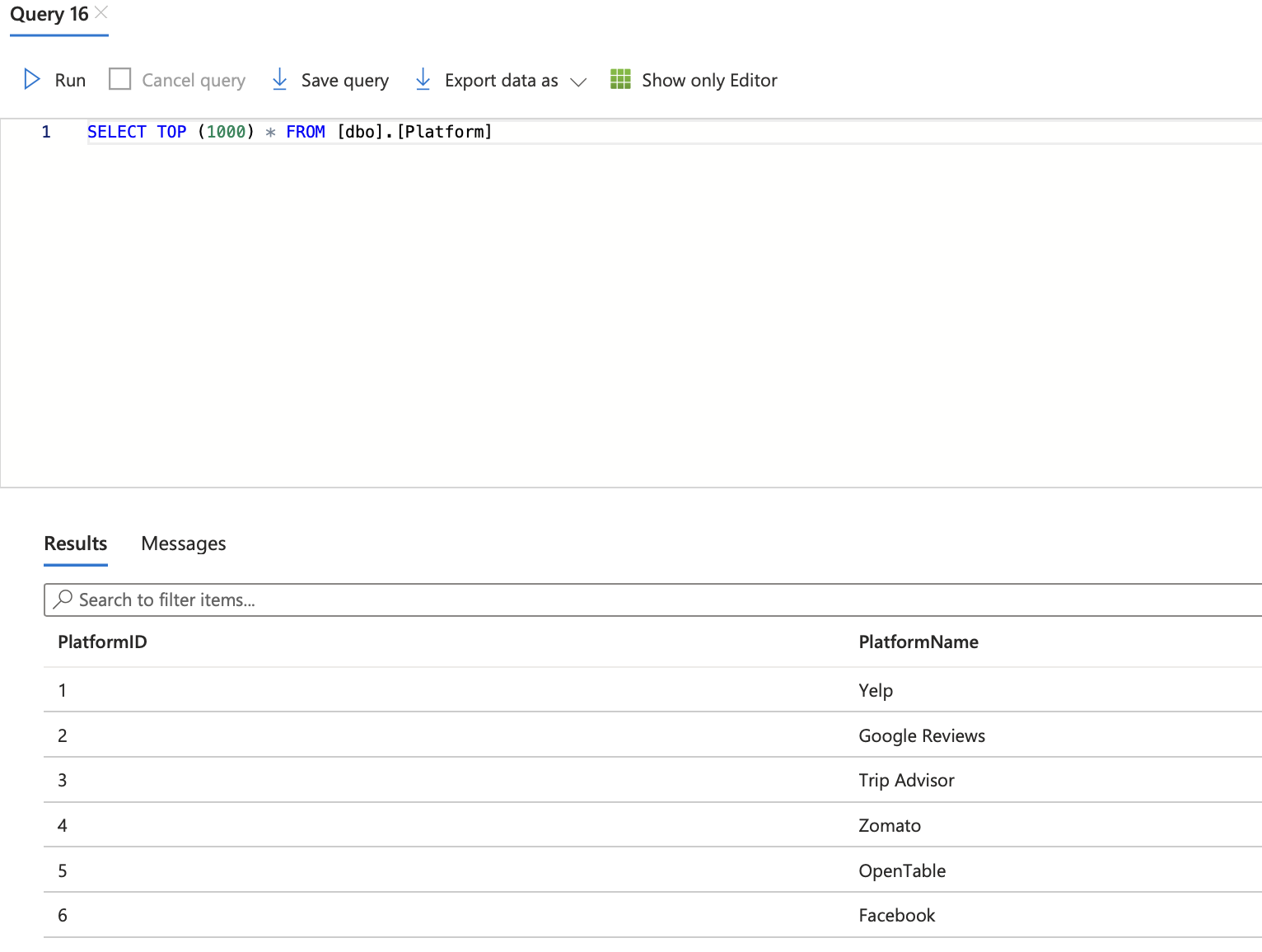


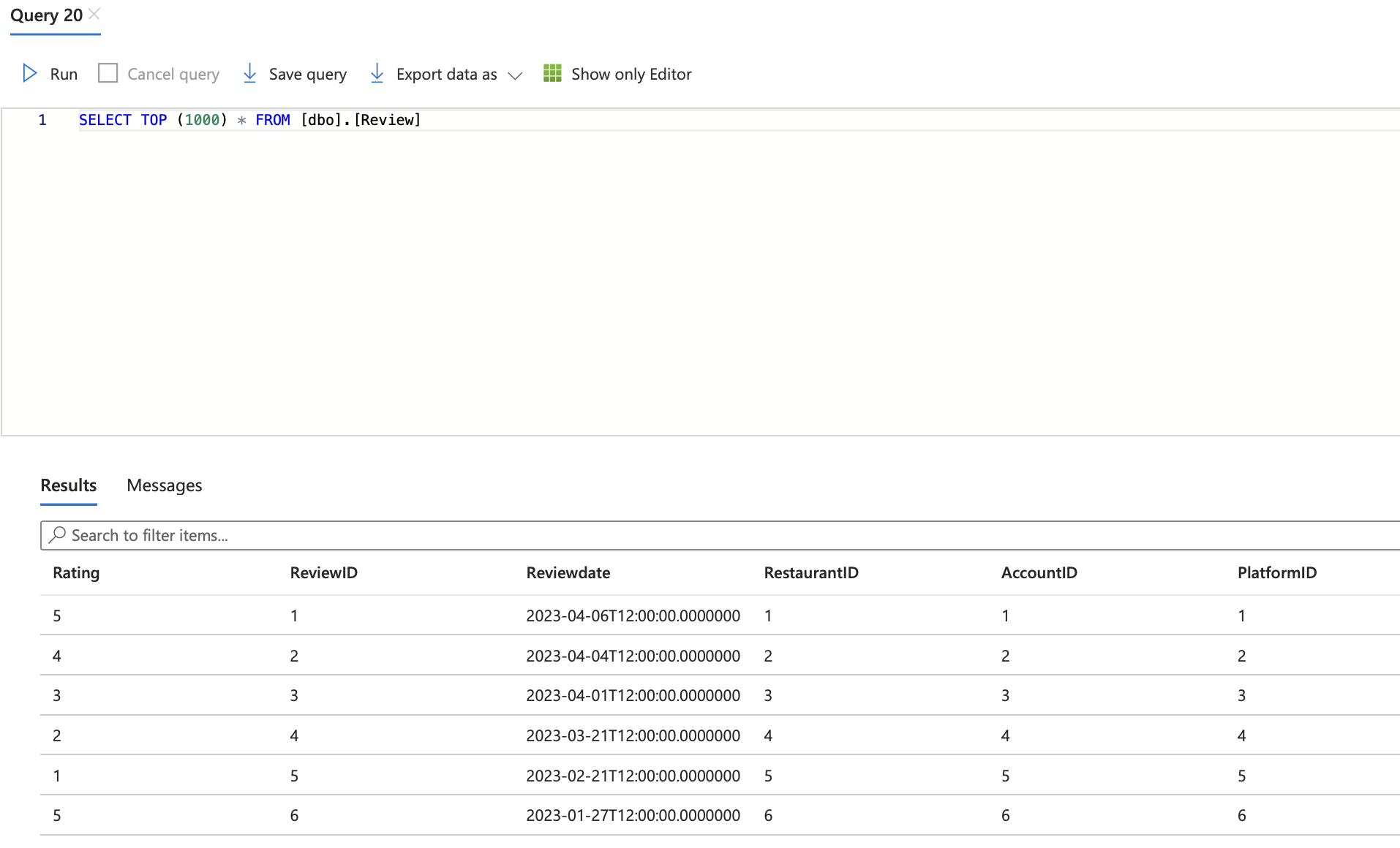


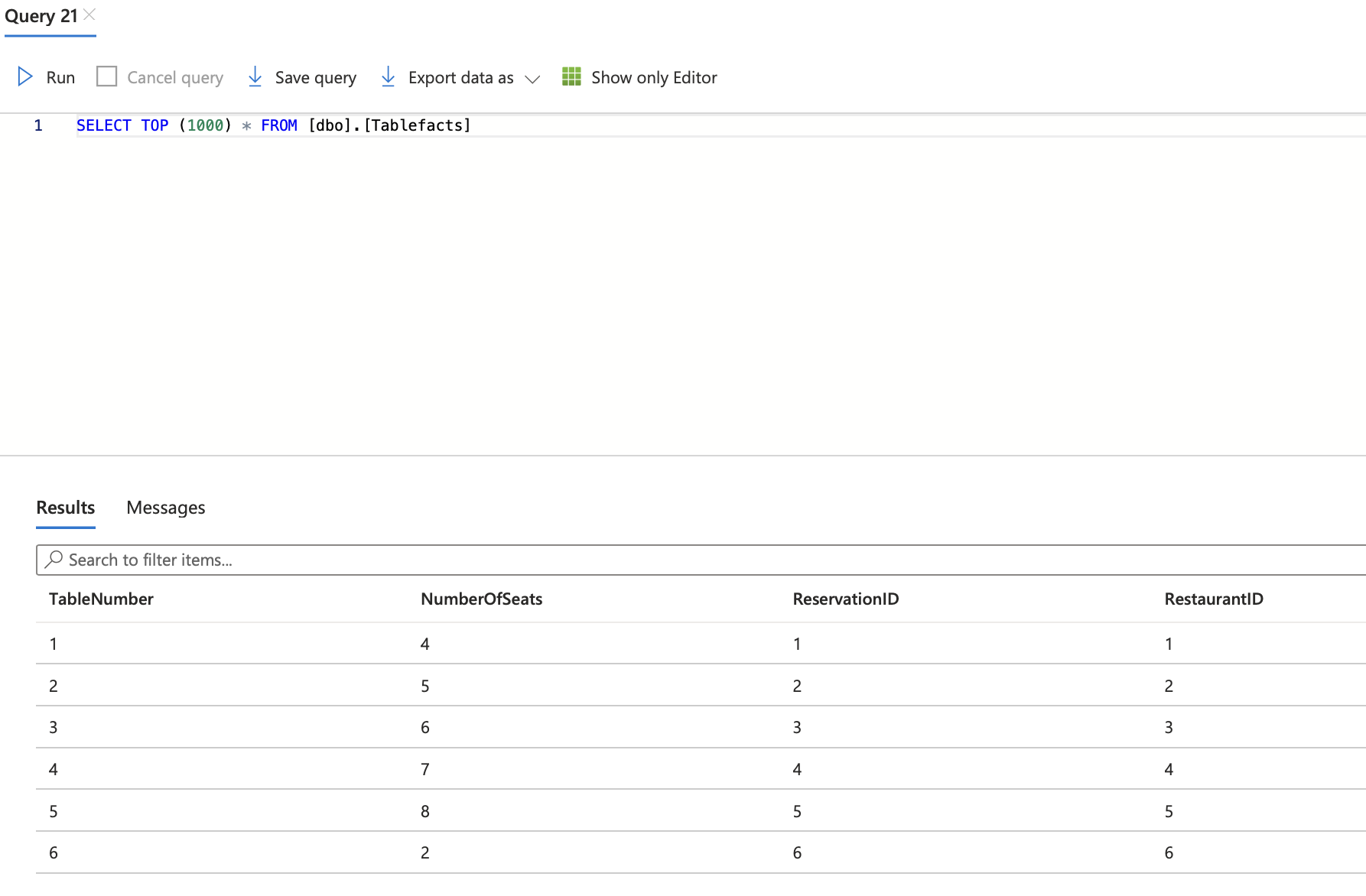












Query 1:

Business Justification: This query is designed for platform users to quickly access a list of restaurants that are open late night and have a 4 star rating or higher. This is useful for students who are up late at night studying and want to see what is good and available. The user will also see the address and phone number so they can decide to order at the closest restaurant that meets their criteria.

|  |
| --- |
| SELECT RestaurantName, AVG(Rating) AS Rating, ClosingTime, Cuisine, StreetAddress, City, ParkingInformation, PhoneNumber  FROM Restaurant, Restaurant\_Address, HourOfOperations, Review  WHERE Restaurant.RestaurantID = HourOfOperations.RestaurantID AND Restaurant\_Address.RestaurantID = Restaurant.RestaurantID AND Review.RestaurantID = Restaurant.RestaurantID AND ClosingTime >= '22:00:00' AND Rating >= 4  GROUP BY RestaurantName, ClosingTime, Cuisine, StreetAddress, City, ParkingInformation, PhoneNumber |

Query 2: **This query does NOT have an output based on our current data and it’s more of a hypothetical example of what could be possible.**

Business Justification: This query can be useful for helping identify the dietary restrictions of customers who have recently made a reservation at a particular restaurant (in this case “The Green Elephant”). This information will help the restaurant prepare for future customer needs in advance and ensure they have appropriate menu items available and help order the necessary ingredients.

|  |
| --- |
| SELECT DISTINCT Customer\_DietaryRestriction.DietaryRestriction  FROM Customer\_DietaryRestriction  JOIN Reservation ON Customer\_DietaryRestriction.AccountID = Reservation.AccountID  JOIN Restaurant ON Reservation.RestaurantID = Restaurant.RestaurantID  WHERE Restaurant.RestaurantName = 'The Green Elephant' AND Reservation.Reservationdate >= DATEADD(WEEK, -1, GETDATE()); |

Query 3:

Business Justification: **This query does NOT have an output based on our current data and it’s more of a hypothetical example of what could be possible.**

This query will find the top-rated restaurants in a specific city, helping a restaurant aggregator platform to find the top-rated restaurants in a specific city. The platform can use this information to recommend these restaurants to its users, thereby enhancing their user experience. For instance, when visiting a new town a user may want to pick a restaurant that has the best reviews (highly rated) within the city.

|  |
| --- |
| SELECT RestaurantName, AVG(Rating) as Avg\_Rating  FROM Restaurant r  INNER JOIN Review rv ON r.RestaurantID = rv.RestaurantID  INNER JOIN Restaurant\_Address ra ON r.RestaurantID = ra.RestaurantID  WHERE ra.City = 'New York'  GROUP BY RestaurantName  ORDER BY Avg\_Rating DESC; |

Query 4:

Business Justification:

This query will help a restaurant to determine the busiest days of the week. The restaurant can use this information to allocate resources effectively, manage staff schedules, and improve the customer experience. A business may want to offer discounts for when days it isn’t the busiest.

|  |
| --- |
| SELECT Specificday, COUNT(\*) as Reservation\_Count  FROM HourOfOperations ho  INNER JOIN Reservation r ON ho.RestaurantID = r.RestaurantID  WHERE r.ReservationCancellation != 'Y'  GROUP BY Specificday  ORDER BY Reservation\_Count DESC; |

Query 5:

Business Justification:

This query could help users find the top rated restaurants, filtering only the restaurants that have a higher rating than the average rating of all restaurants. After getting the results, the user can then look through them and make a choice, knowing that they’re only looking at highly rated restaurants.

|  |
| --- |
| select Restaurant.RestaurantID, Restaurant.RestaurantName, Review.Rating  from Restaurant inner join Review  on Restaurant.RestaurantID = Review.RestaurantID  where Review.Rating > (select avg(Rating) from Review)  order by Review.Rating desc; |

Query 6:

Business Justification:

This query serves as an example for a potential customer’s search. In this example, the user is looking for a restaurant that serves a specific cuisine and has enough space for their own party. This query could be customized in many ways depending on what the individual user specifically wants.

|  |
| --- |
| select Restaurant.RestaurantID, Restaurant.Cuisine, Tablefacts.NumberOfSeats  from Restaurant inner join Tablefacts  on Restaurant.RestaurantID = Tablefacts.RestaurantID  where Restaurant.Cuisine = 'Barbecue' and Tablefacts.NumberOfSeats >= 7; |

Query 7:

Business Justification:

This query simply shows the number of reviews on each platform for each individual restaurant. This could be useful information for restaurants to know as they would be able to see on what platform their customers are most active on as well as having the ability to develop various marketing strategies to keep pushing on the platforms they’re already strong on.

|  |
| --- |
| select Restaurant.RestaurantName, Platform.PlatformName, count(Review.ReviewID)  from Platform inner join Review  on Platform.PlatformID = Review.PlatformID  inner join Restaurant  on Review.RestaurantID = Restaurant.RestaurantID  group by Restaurant.RestaurantName, Platform.PlatformName  order by count(Review.ReviewID) desc; |

Query 8:

Business Justification:

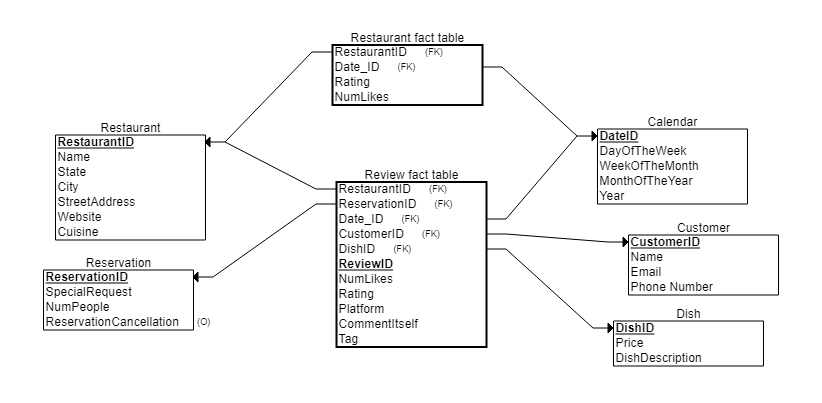
This query is useful to get the number of reservations for each restaurant. This information can be used to understand the popularity of each restaurant and can help in staffing and resource allocation decisions for the restaurant or help customers understand which restaurants tend to get the most busy.

|  |
| --- |
| SELECT RestaurantName, COUNT(\*) FROM Reservation  INNER JOIN Restaurant ON Reservation.RestaurantID = Restaurant.RestaurantID  GROUP BY RestaurantName; |

**Part 2)**

2. There are two fact tables. The subject of analysis for the review fact table are the reviews themselves. The subject of analysis for the other fact table is each restaurant.

3.



4.

The Review fact table includes very detailed data. The restaurant, the reservation, the date, the customer who made the reservation and the dish associated with the review will be correlated with each review. Also, the number of likes, the platform, the comment (textual data), the rating, and the tag will be correlated with each review. So, it is a very granular fact table that will allow for aggregations for further analysis. The restaurant fact table is aggregated on the restaurant level. So, the rating for each restaurant will be very clear along with the amount of likes associated with each restaurant. For example, if a restaurant has a rating of four out of five stars, the data will show that the reviews have 896 likes. On the contrary, a restaurant could get five out of five stars, but have 5 likes on those reviews leading to a less credible review. Also, the date of the aggregation is noted, so an investigation could be made if a restaurant’s rating goes significantly down or up. This is to prevent fake negative reviews or fake positive reviews.

5.

* For an example of Extract, Transform, and Load, we will look at constructing the Restaurant Fact Table
* For extraction from the operational database, the reviews in a certain time period need to be grouped by RestaurantID. The number of likes and rating will be other fields that will be aggregated on the restaurant level. The date of the aggregation will be the DateID.
* For Transformation, any records with missing values of number of likes should be dropped. The DateID needs to be transformed to the appropriate format for analysis. There should be quality checks making sure all the data was transferred properly. All of the data should be in the correct format to fit the data warehouse.
* For Loading, the restaurant dimension should have an initial load of the data from the operational database. After this, the data should be updated every day. This is to highlight a sudden drop in a restaurant’s ratings for any particular reason including false reviews.