###DDL Statements###

CREATE TABLE `main`.`Restaurant` (

`restaurant\_name` VARCHAR(100) NULL,

`restaurant\_phone` VARCHAR(45) NULL,

`restaurant\_website` VARCHAR(200) NULL,

`hours` VARCHAR(200) NULL,

`price\_range` VARCHAR(45) NULL,

`price\_range\_num` INT NULL,

`restaurant\_id` BIGINT NOT NULL,

`address.city` VARCHAR(45) NULL,

`address.state` VARCHAR(45) NULL,

`address.postal\_code` INT NULL,

`address.street` VARCHAR(100) NULL,

`address.formatted` VARCHAR(200) NULL,

`geo.lat` FLOAT NULL,

`geo.lon` FLOAT NULL,

PRIMARY KEY (`restaurant\_id`));

CREATE TABLE `main`.`Cuisine` (

`cuisine` VARCHAR(100) NOT NULL,

`restaurant\_id` BIGINT NOT NULL,

PRIMARY KEY (`cuisine`, `restaurant\_id`),

CONSTRAINT `restaurant\_id`

FOREIGN KEY (`restaurant\_id`)

REFERENCES `main`.`Restaurant` (`restaurant\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

CREATE TABLE `main`.`FoodItem` (

`name` VARCHAR(200) NOT NULL,

`description` VARCHAR(200) NOT NULL,

`price` FLOAT NULL,

`rest\_id` BIGINT NOT NULL,

PRIMARY KEY (`name`, `description`, `rest\_id`),

CONSTRAINT `rest\_id`

FOREIGN KEY (`rest\_id`)

REFERENCES `main`.`Restaurant` (`restaurant\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

###Queries###

# Find restaurants of a certain cuisine

SELECT restaurant\_name, `address.formatted`

FROM Cuisine C

JOIN Restaurant R ON C.restaurant\_id = R.restaurant\_id

WHERE cuisine = 'Ice Cream';

# Dish recommendation off a search

SELECT restaurant\_name, name, description

FROM Restaurant R

JOIN FoodItem F ON F.rest\_id = R.restaurant\_id

WHERE LOWER(name) LIKE '%pizza%';

# Find restaurants and the dishes they serve that fall under a certain price range and serve a desired dish

SELECT r.restaurant\_name, f.name

FROM Restaurant r

INNER JOIN (SELECT \*

FROM FoodItem

WHERE name LIKE '%Hamburger%') f ON r.restaurant\_id = f.rest\_id

WHERE r.price\_range\_num < 2;

# Find the most expensive dish that a restaurant serves

SELECT DISTINCT f.name, f.price

FROM Restaurant r

INNER JOIN FoodItem f ON r.restaurant\_id = f.rest\_id

WHERE r.restaurant\_name = "Dunkin'"

AND f.price >= ALL(SELECT f1.price

FROM Restaurant r1

INNER JOIN FoodItem f1 ON r1.restaurant\_id = f1.rest\_id

WHERE r1.restaurant\_name = "Dunkin'");

# For a given cuisine and dish, get places where the dish is priced above average

WITH choose\_food\_cuisine AS(SELECT restaurant\_name, name, description, price

FROM FoodItem F

JOIN Cuisine C ON F.rest\_id = C.restaurant\_id

JOIN Restaurant R ON R.restaurant\_id = F.rest\_id

WHERE cuisine = 'Ice Cream' AND name LIKE '%sundae%' AND price <> 0)

SELECT restaurant\_name, name, description, price

FROM choose\_food\_cuisine

WHERE price > (SELECT AVG(price) FROM choose\_food\_cuisine);

# Find the location of a given restuarant for a chain that has less expensive dishes on average with regards to the equivalent dishes at the other restuarant locations

WITH avg\_prices\_rest AS ((SELECT r1.restaurant\_id, AVG(f1.price) as avg\_price

FROM Restaurant r1

INNER JOIN FoodItem f1 ON r1.restaurant\_id = f1.rest\_id

WHERE restaurant\_name = "McDonald's"

GROUP BY r1.restaurant\_id

HAVING avg\_price != 0))

SELECT DISTINCT r.restaurant\_name, r.`address.formatted`

FROM Restaurant r

INNER JOIN FoodItem f ON r.restaurant\_id = f.rest\_id

WHERE r.restaurant\_name = "McDonald's"

AND r.restaurant\_id = (SELECT av.restaurant\_id

FROM avg\_prices\_rest av

WHERE av.avg\_price <= ALL(SELECT avg\_price

FROM avg\_prices\_rest));

# Find restaurants within a certain number of miles from ourselves

SELECT restaurant\_name,

( 3959 \* acos( cos( radians(MY\_LAT) ) \* cos( radians( `geo.lat` ) )

\* cos( radians(`geo.lon`) - radians(MY\_LNG)) + sin(radians(MY\_LAT))

\* sin( radians(`geo.lat`)))) AS distance

FROM Restaurant

HAVING distance < MILE\_LIMIT

ORDER BY distance;

#Find all restaurants with contact information

SELECT \*

FROM Restaurant

WHERE restaurant\_phone is not null AND restaurant\_website is not null

AND LENGTH(restaurant\_phone) > 0 AND LENGTH(restaurant\_website) > 0

# Find all restaurants of a cuisine type serving a food item under a certain amount within a close distance from ourselves

WITH within\_dist AS (

SELECT restaurant\_id,

( 3959 \* acos( cos( radians(MY\_LAT) ) \* cos( radians( `geo.lat` ) )

\* cos( radians(`geo.lon`) - radians(MY\_LNG)) + sin(radians(MY\_LAT))

\* sin( radians(`geo.lat`)))) AS distance

FROM Restaurant

HAVING distance < MILE\_LIMIT

),

of\_cuisine AS (

SELECT restaurant\_id

FROM Cuisine

WHERE cuisine = CUISINE\_TYPE

)

has\_cheap\_food AS (

SELECT DISTINCT restaurant\_id

FROM FoodItem fi Join Restaurant r ON fi.rest\_id = r.restaurant\_id

WHERE fi.price < PRICE\_LIMIT

)

SELECT restaurant\_name

FROM within\_dist wd JOIN of\_cuisine oc ON wd.restaurant\_id = oc.restaurant\_id

JOIN has\_cheap\_food hcf ON wd.restaurant\_id = hcf.restaurant\_id

JOIN Restaurant r ON r.restaurant\_id = wd.restaurant\_id;

#Find the number of restaurants that serve a certain type of cuisine in a specific New York zipcode

SELECT COUNT(Restaurant.restaurant\_id) AS Number\_Restaurants, `address.postal\_code`

FROM Restaurant

JOIN Cuisine ON Restaurant.restaurant\_id = Cuisine.restaurant\_id

WHERE Cuisine.cuisine LIKE '%Italian%' AND `address.city` LIKE 'New York'

GROUP BY `address.postal\_code`

ORDER BY Number\_Restaurants DESC;

#Find the restaurant (less than or equal to a certain price range) name, website, number of dishes, and website that serves a certain cuisine in a certain city and has the most number of dishes (has the most selection)

WITH rests AS (

SELECT restaurant\_name, restaurant\_website, restaurant\_id

FROM Restaurant

WHERE price\_range\_num <= 2 AND `address.city` LIKE '%Staten Island%' AND LENGTH(restaurant\_website) >0

),

cuisines AS(

SELECT \* FROM Cuisine

WHERE Cuisine LIKE '%Chinese%'

),

joined AS (

SELECT rests.restaurant\_name,rests.restaurant\_website,rests.restaurant\_id, cuisines.cuisine

FROM rests

JOIN cuisines ON rests.restaurant\_id = cuisines.restaurant\_id

),

rest\_ids AS (

SELECT restaurant\_id FROM joined),

num\_dishes AS (

SELECT COUNT(name) AS num\_dishes, FoodItem.rest\_id

FROM FoodItem

JOIN rest\_ids ON FoodItem.rest\_id = rest\_ids.restaurant\_id

GROUP BY FoodItem.rest\_id

)

SELECT num\_dishes, restaurant\_name, restaurant\_website,cuisine FROM num\_dishes

JOIN joined ON num\_dishes.rest\_id = joined.restaurant\_id

ORDER BY num\_dishes DESC LIMIT 1;