Title: DBMS HW2
Date: 9/30/2025

Question 1: Average food price at each restaurant

This query works by selecting name and food price. We are able to attain those values by joining the foodIDs from foods and serves, which would match columns only where the food IDs match. Finally, we join the restaurant IDs where they match the serve IDs, because we are only interested in restaurants that serve their own food. We also Group by the name.

```
18
19 •  SELECT
20        r.`name`, AVG(price)
21    FROM
22        serves s
23            JOIN
24        foods f ON s.foodID = f.foodID
25            JOIN
26        restaurants r ON r.restID = s.restID
27    GROUP BY r.`name`;
28
29
30        /*
31        Maximum food price at each restaurant:
```

Result Grid | Filter Rows: | Export: | Wrap Cell Cor

name	AVG(price)
La Trattoria	13.5
Sushi Haven	12
Taco Town	9.5
Bistro Paris	13.5
Thai Delight	12
Indian Spice	13.5

Question 2: Maximum food price at each restaurant:

We select name and max price. We get it from serves, and we want to join serves to the food associated with each restaurant. Additionally, we join restaurant as well, because we want the restaurant associated with their food. Since we used the MAX function, we will retrieve the max price, and it will be grouped by restaurant name.

```
40 •      SELECT
41            r.`name`, MAX(price)
42        FROM
43            serves s
44        JOIN
45            foods f ON s.foodID = f.foodID
46        JOIN
47            restaurants r ON r.restID = s.restID
48        GROUP BY r.`name`;
49
```

lesult Grid | Filter Rows: | Export: | Wrap Cell

name	MAX(price)
La Trattoria	15
Sushi Haven	14
Taco Town	11
Bistro Paris	18
Thai Delight	13
Indian Spice	15

Question 3: Count of different food types at each restaurant

Here we select restaurant name and the count of distinct food types. The

distinct keyword makes it so that it will only contribute to the count if it is unique type. We select it from serves, and we want to join foods with the restaurant associated with it again. We combine the tables by joining foods to the corresponding foodID in serves, and likewise with restaurants so there will only be foods and restaurants that are associated, then group by name.

```
59 •    SELECT
60          r.`name`, count( distinct f.`type`)
61      FROM
62          serves s
63      JOIN
64          foods f ON s.foodID = f.foodID
65      JOIN
66          restaurants r ON r.restID = s.restID
67      GROUP BY r.`name`;
68
69      -- Average price of foods by chef
70      /*
71      We select the chef name and average food prices for display. W
72      need to use joins to associate chefs with the restaurant at wh
73      and restID in works. Next, we need to know what foods are serv
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	count(distinct f.`type`)
Bistro Paris	1
Indian Spice	1
La Trattoria	1
Sushi Haven	2
Taco Town	1
Thai Delight	1

Question 4: Average price of foods by chef

We select the chef name and average food prices for display. We go to the chefs table first, and need to use joins to associate chefs with the restaurant at which they work using the corresponding chefID and restID in works. Next, we need to know what foods are served at each restaurant, so we combine restaurants and foods using the serves table, making sure the restID and foodID are matching. Next we want to join the foods with the serves table, and since we joined only matching restIDs and matching foodIDs throughout all of our joins, we are left only with each food that each chef serves. Finally, we group by name.

```
//
78 •  SELECT
79        c.`name`, AVG(f.price)
80    FROM
81        chefs c
82            JOIN
83        works w ON c.chefID = w.chefID
84            JOIN
85        restaurants r ON r.restID = w.restID
86            JOIN
87        serves s ON r.restID = s.restID
88            JOIN
89        foods f ON f.foodID = s.foodID
90    GROUP BY c.`name`;
91
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	AVG(f.price)
Jane Smith	12.75
John Doe	11.5
Robert Brown	12.75
Alice Johnson	11.5
Michael Wilson	12.75
Emily Davis	12.75

Question 5: Find the Restaurant with the Highest Average Food Price

We select the restaurant name and average food price, and we make avg_price a variable. We use restaurants first, and join it to the corresponding foods using the serves table, because we only want foods that the restaurant serves. Then we group by restaurant name, but since we care about the highest price, we want to order it by the avg_price. Since we want the highest average priced restaurant specifically, we limit the result to 1.

```
100 •  SELECT
101        r.`name`, AVG(f.price) as avg_price
102    FROM
103        restaurants r
104            JOIN
105        serves s ON s.restID = r.restID
106            JOIN
107        foods f ON s.foodID = f.foodID
108    GROUP BY r.`name`
109    ORDER BY avg_price desc
110    limit 1;
111
112    /*
```

Result Grid | Filter Rows: | Export: | Wrap Cell

name	avg_price
La Trattoria	13.5

Extra Credit: Question 6:

We select the average price, assign it to the variable avg_price, and each chef name. We need the last column to fill the requirement, but some chefs work at multiple restaurants and we can't cram two rows into one. Therefore instead of creating a whole new column we can just concatenate a column called restaurants onto the end, and use the distinct keyword to avoid repeats. From there, we start with the chefs table and want to join them to the restaurant that they work at using the works table, and we do this by making sure the chefID and restID match each other. next, we need to combine restaurants to the foods they serve using the serve table,and making sure the foodID and restID match. We will be left with only the average price of each chef, their name, and the restaurants they work at. Finally, we group by chef name and order it by avg price with the keyword desc to get the highest values on top.

```
134        GROUP_CONCAT(DISTINCT r.`name`) Restaurants
135    FROM
136        chefs c
137            JOIN
138        works w ON c.chefID = w.chefID
139            JOIN
140        restaurants r ON w.restID = r.restID
141            JOIN
142        serves s ON r.restID = s.restID
143            JOIN
144        foods f ON s.foodID = f.foodID
145    GROUP BY c.`name`
146    ORDER BY avg_price desc;
147
148
```

Result Grid | Filter Rows: | Export: | Wrap Cell

avg_price	name	Restaurants
12.75	Emily Davis	Indian Spice,Thai Delight
12.75	Jane Smith	La Trattoria,Sushi Haven
12.75	Michael Wilson	Indian Spice,Thai Delight
12.75	Robert Brown	Bistro Paris,Sushi Haven
11.5	Alice Johnson	Bistro Paris,Taco Town
11.5	John Doe	La Trattoria,Taco Town