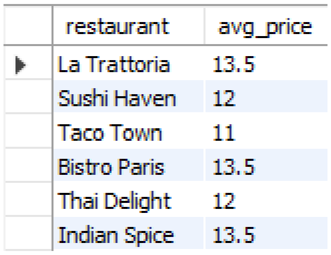
DB Assignment 2
9/29/2025

SQL

```sql
-- 1. Average Price of Foods at Each Restaurant
select restaurants.name as restaurant, avg(foods.price) AS avg_price
from restaurants
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by restaurants.name;
```

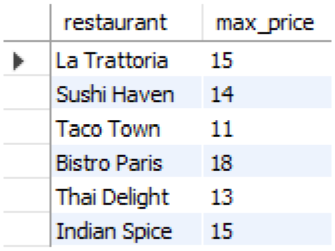
- 

restaurant	avg_price
La Trattoria	13.5
Sushi Haven	12
Taco Town	11
Bistro Paris	13.5
Thai Delight	12
Indian Spice	13.5

- This query joins restaurants and foods through serves, first through restaurant ID, then food ID in order to return the restaurant name and average food price at the restaurant.

SQL

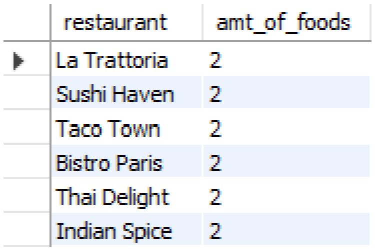
```sql
-- 2. Maximum Food Price at Each Restaurant
select restaurants.name as restaurant, max(foods.price) as max_price
from restaurants
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by restaurants.name;
```

- 

restaurant	max_price
La Trattoria	15
Sushi Haven	14
Taco Town	11
Bistro Paris	18
Thai Delight	13
Indian Spice	15

- This query joins restaurants and foods through serves in the same manner as the last query, but instead selecting for the max food price that can be returned to find the most expensive food at each restaurant.

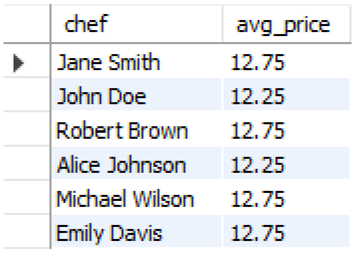
```SQL
-- 3. Count of Different Food Types Served at Each Restaurant
select restaurants.name as restaurant, count(foods.foodID) as amt_of_foods
from restaurants
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by restaurants.name;
```

- 

restaurant	amt_of_foods
La Trattoria	2
Sushi Haven	2
Taco Town	2
Bistro Paris	2
Thai Delight	2
Indian Spice	2

- This query join restaurants and foods the same as before, but instead selecting for the count of food items for each restaurant.

```SQL
-- 4. Average Price of Foods Served by Each Chef
select chefs.name as chef, avg(foods.price) as avg_price
from chefs
join works on chefs.chefID = works.chefID
join restaurants on works.restID = restaurants.restID
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by chefs.chefID, chefs.name;
```

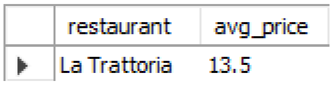
- 

chef	avg_price
Jane Smith	12.75
John Doe	12.25
Robert Brown	12.75
Alice Johnson	12.25
Michael Wilson	12.75
Emily Davis	12.75

- This query joins chefs, restaurants, and foods using works and serves respectively to correlate chefs to the foods that they serve at each restaurant they work at.

SQL

```
-- 5. Find the Restaurant with the Highest Average Food Price
select restaurants.name as restaurant, avg(foods.price) AS avg_price
from restaurants
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by restaurants.name
order by avg_price desc
limit 1;
```

- 

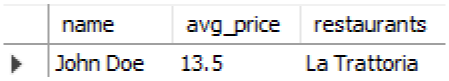
	restaurant	avg_price
▶	La Trattoria	13.5

- This query joins restaurants and foods again, but puts them in descending order by average price, limiting the query to one result to find the restaurant with the highest average food price.

SQL

```
-- 6 Extra Credit
-- Determine which chef has the highest average price of the foods served at
the
-- restaurants where they work. Include the chef's name, the average food
price,
-- and the names of the restaurants where the chef works.
-- Sort the  results by the average food price in descending order.

select chefs.name, avg(foods.price) as avg_price, restaurants.name,
group_concat(distinct restaurants.name order by restaurants.name) as
restaurants
from chefs
join works on chefs.chefID = works.chefID
join restaurants on works.restID = restaurants.restID
join serves on restaurants.restID = serves.restID
join foods on serves.foodID = foods.foodID
group by chefs.chefID, chefs.name, restaurants.name
order by avg(foods.price) desc
limit 1;
```

- 

	name	avg_price	restaurants
▶	John Doe	13.5	La Trattoria

- This query joins across every relationship in the model in order to find the chef with the highest average food price and where they work.