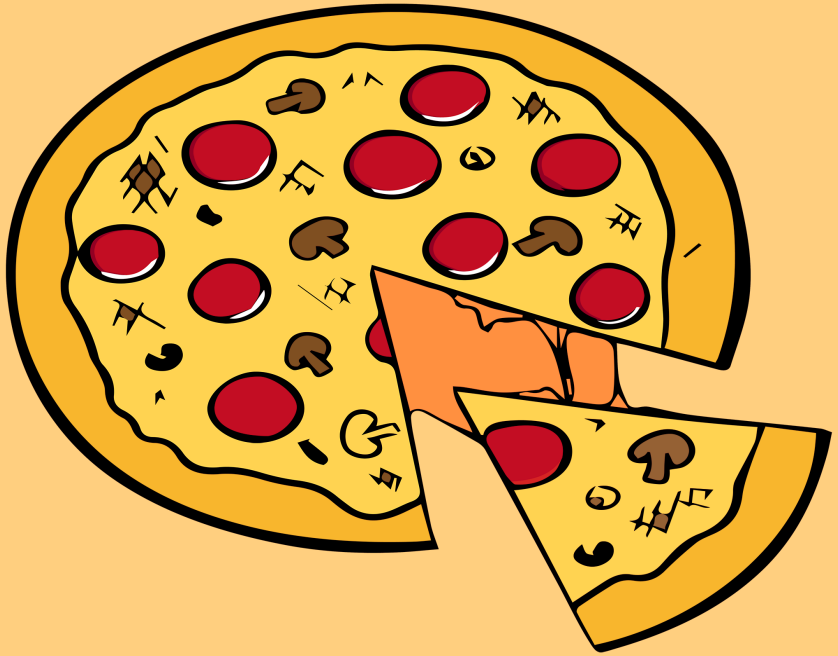
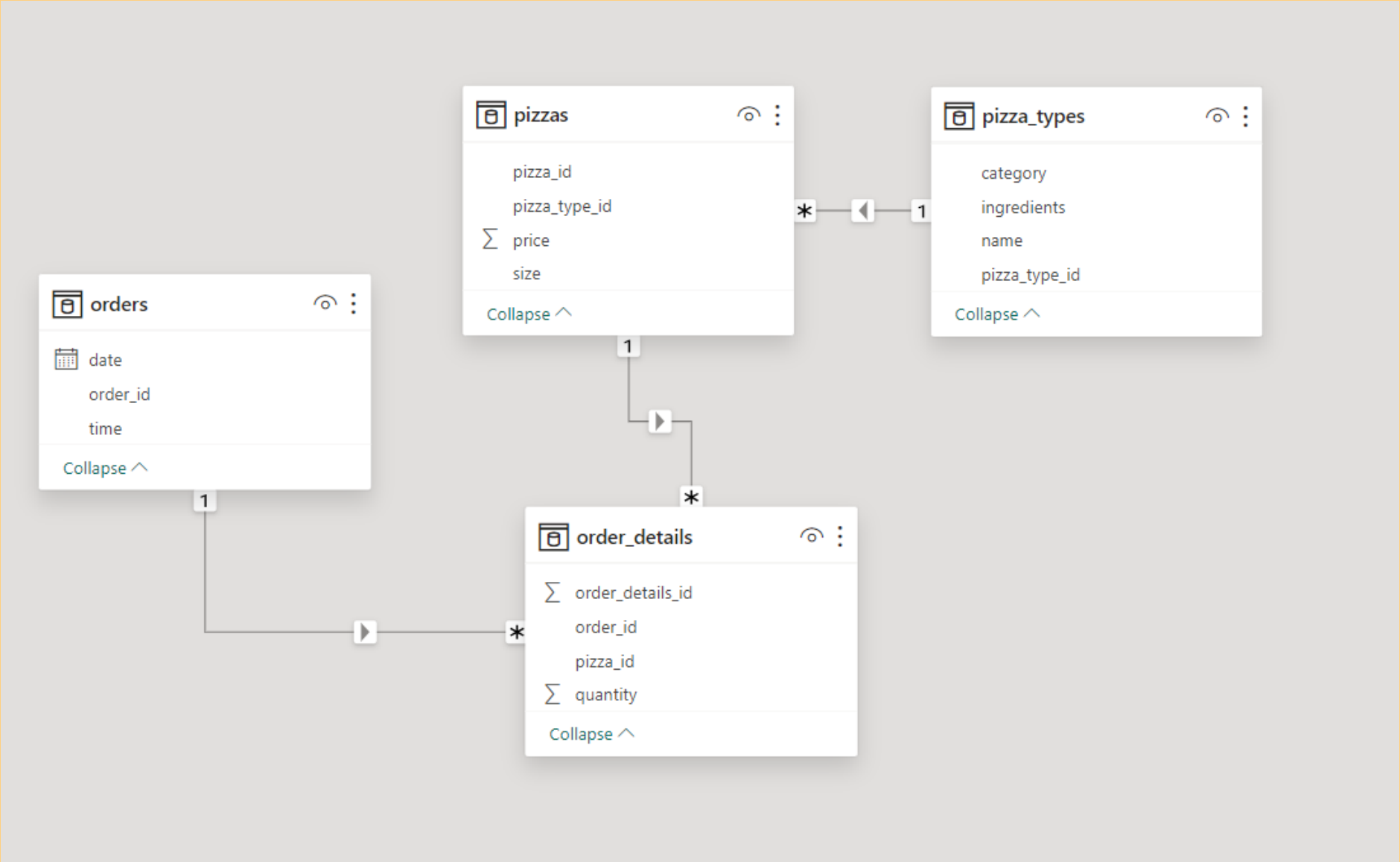


SQL Project on Pizza Sales

HELLO!

I am Prasad, Master of Technology Management graduate from UCSB and I have used SQL queries to solve questions related to sales data of pizza

Schema of the data set



Questions:

1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.
3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.
5. List the top 5 most ordered pizza types along with their quantities.
6. Join the necessary tables to find the total quantity of each pizza category ordered.
7. Determine the distribution of orders by hour of the day.
8. Join relevant tables to find the category-wise distribution of pizzas.
9. Group the orders by date and calculate the average number of pizzas ordered per day.
10. Determine the top 3 most ordered pizza types based on revenue.
11. Calculate the percentage contribution of each pizza type to total revenue.

Retrieve the total number of orders placed.

```sql
-- Retrieve the total number of orders placed.
SELECT COUNT(order_id) AS total_orders FROM orders;
```

Result Grid

total_orders
21350

Calculate the total revenue generated from pizza sales.

```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS Revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

Result Grid

Revenue
817860.05

Identify the highest-priced pizza.

```sql
-- Identify the highest-priced pizza.
SELECT
    pizza_type.name, pizzas.price
FROM
    pizzas
        JOIN
    pizza_type ON pizza_type.pizza_type_id = pizzas.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

Result Grid   Filter

name	price
The Greek Pizza	35.95

Identify the most common pizza size ordered.

```
41    -- Identify the most common pizza size ordered.
42 •  SELECT
43        pizzas.size, COUNT(order_details.quantity) AS order_count
44    FROM
45        pizzas
46            JOIN
47        order_details ON order_details.pizza_id = pizzas.pizza_id
48    GROUP BY pizzas.size
49    ORDER BY order_count DESC;
```

Result Grid

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

List the top 5 most ordered pizza types along with their quantities.

```
8   -- List the top 5 most ordered pizza types along with their quantities.
9   SELECT
10      pizza_type.name, ROUND(SUM(order_details.quantity), 2) as Quantity
11  FROM
12      pizza_type
13          JOIN
14      pizzas ON pizzas.pizza_type_id = pizza_type.pizza_type_id
15          JOIN
16      order_details ON order_details.pizza_id = pizzas.pizza_id
17  GROUP BY pizza_type.name
18  ORDER BY Quantity DESC
19  LIMIT 5;
```

Result Grid | Filter Rows:

name	Quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Join the necessary tables to find the total quantity of each pizza category ordered.

```sql
-- Join the necessary tables to find the total quantity of each pizza category ordered.
SELECT
    pizza_type.category, SUM(order_details.quantity) AS quantity
FROM
    pizza_type
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_type.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_type.category
ORDER BY quantity DESC;
```

Result Grid

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.

```sql
-- Determine the distribution of orders by hour of the day.
SELECT
    HOUR(order_time) AS hr_of_the_day,
    COUNT(order_id) AS num_of_orders
FROM
    orders
GROUP BY HOUR(order_time)
ORDER BY hr_of_the_day;
```

Result Grid Filter Rows:

hr_of_the_day	num_of_orde...
9	1
10	8
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28

Join relevant tables to find the category-wise distribution of pizzas.

```sql
-- Join relevant tables to find the category-wise distribution of pizzas.
SELECT
    category, COUNT(pizza_type.category) AS TYPE
FROM
    pizza_type
GROUP BY pizza_type.category
ORDER BY pizza_type.category DESC;
```

Result Grid

category	TYPE
Veggie	9
Supreme	9
Classic	8
Chicken	6

Group the orders by date and calculate the average number of pizzas ordered

```sql
-- Group the orders by date and calculate the average number of pizzas ordered per day.
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity_per_day;
```

59:61

Result Grid Filter

Round(Avg(quantity),0)
138

Retrieve the total number of orders placed.

```sql
-- Determine the top 3 most ordered pizza types based on revenue.
SELECT
    pizza_type.name,
    ROUND(SUM(pizzas.price * order_details.quantity),
            2) AS revenue
FROM
    pizza_type
        JOIN
    pizzas ON pizza_type.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_type.name
ORDER BY revenue DESC
LIMIT 3;
```

Result Grid Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Calculate the percentage contribution of each pizza type to total revenue.

```sql
-- Calculate the percentage contribution of each pizza type to total revenue.
SELECT
    pizza_type.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),
                                2) AS total_sales
                FROM
                    order_details
                        JOIN
                    pizzas ON pizzas.pizza_id = order_details.pizza_id),
            2) * 100 AS percent_share
FROM
    pizza_type
        JOIN
    pizzas ON pizza_type.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_type.category;
```

Result Grid | Filter

category	percent_share
Classic	27
Veggie	24
Supreme	25
Chicken	24