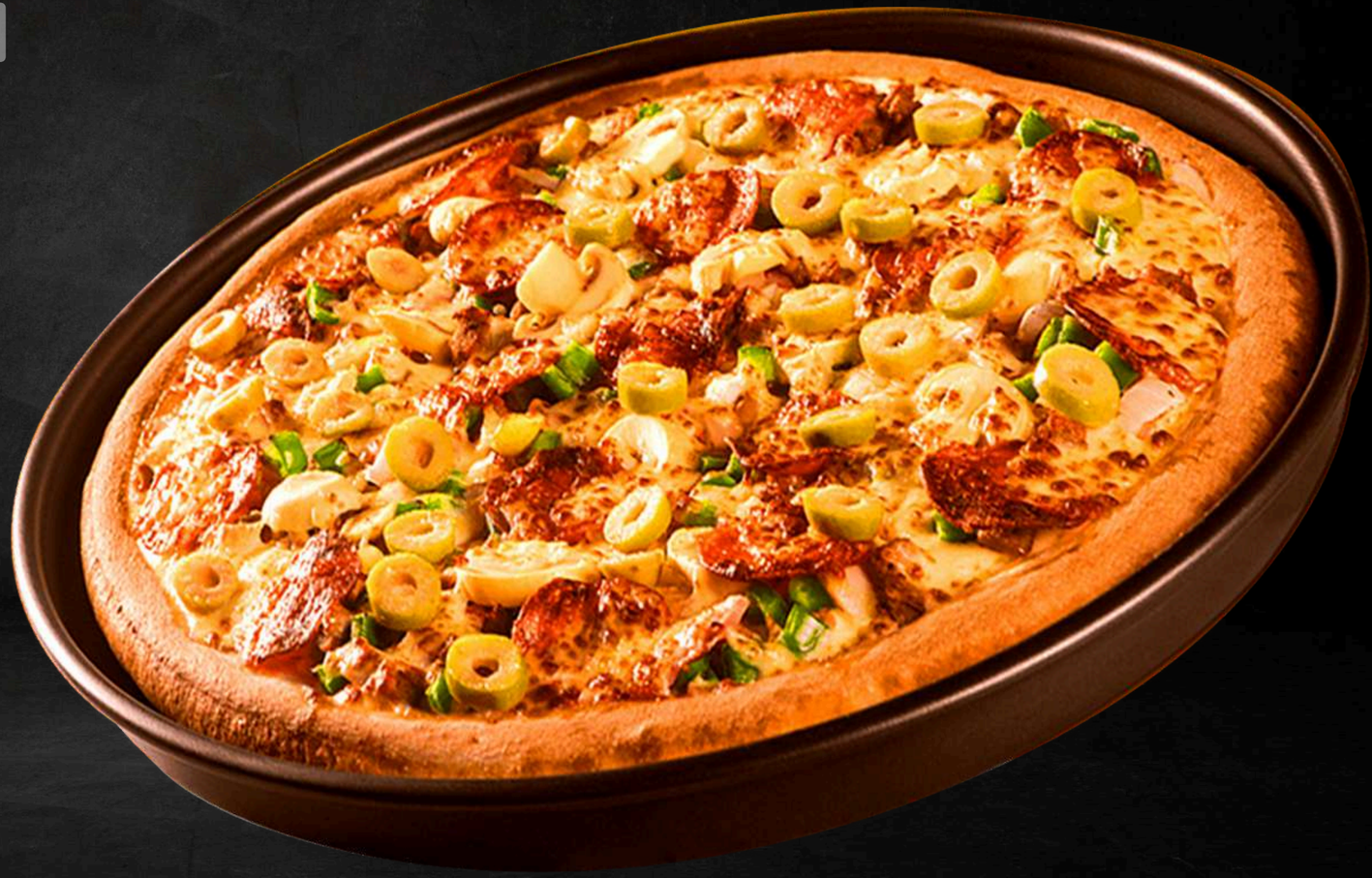
PIZZA SALES REPORT



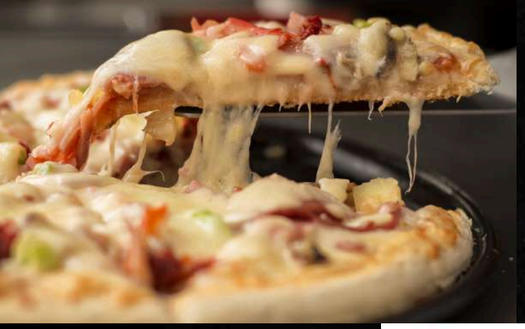
Utilizing MySQL for Data Analysis
An In-Depth Look at Sales Performance

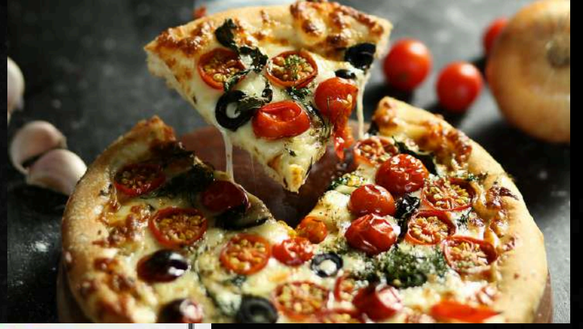
*Project Details
**Project by: Yogesh Gavhane
**Tools Used: MySQL, Excel, Canva

- **Project Objective**

  To analyze the pizza sales data using MySQL in order to

- Identify the best-selling pizza types
- Understand sales performance across different categories
- Find total revenue, average order value, and peak sales times
- Support data-driven business decisions.

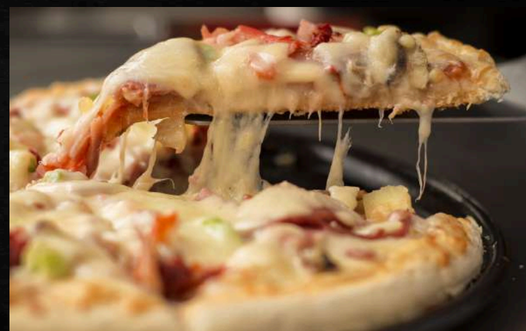


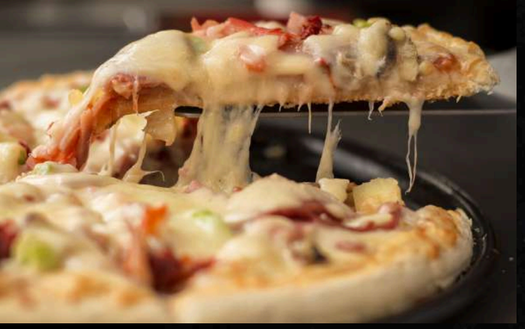


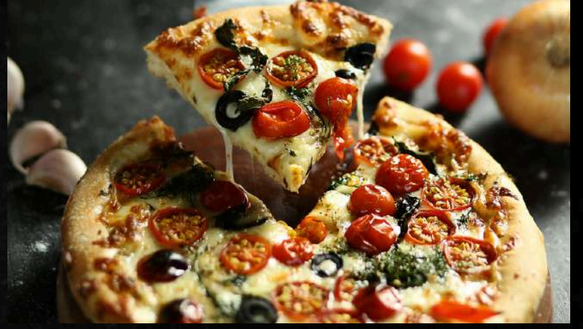
```sql
-- --Retrieve the total number of orders placed.

SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

Result Grid

	total_orders
▶	21350



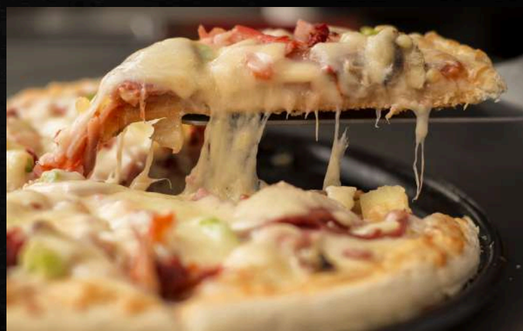


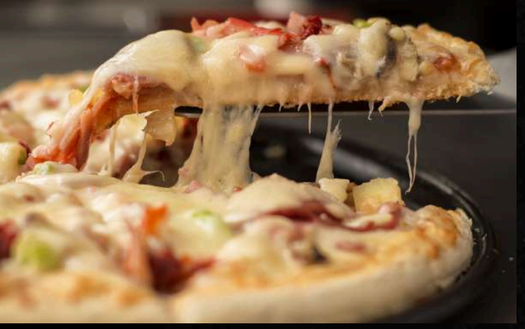


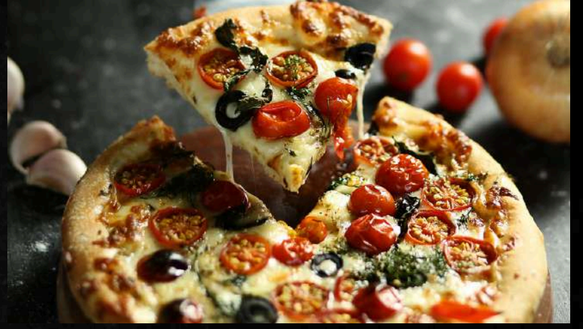
```sql
-- calculate the total revenue generated from pizza sales.

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
```

Result Grid

total_sales
817860.05



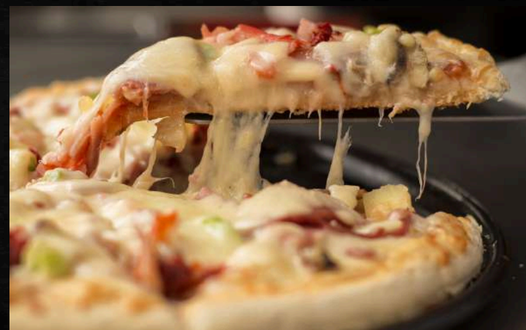


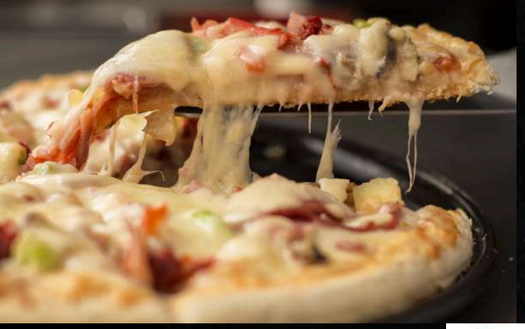


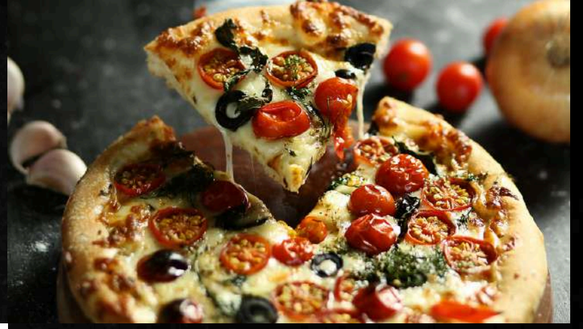
```sql
-- identify the highest -priced pizza.

SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95



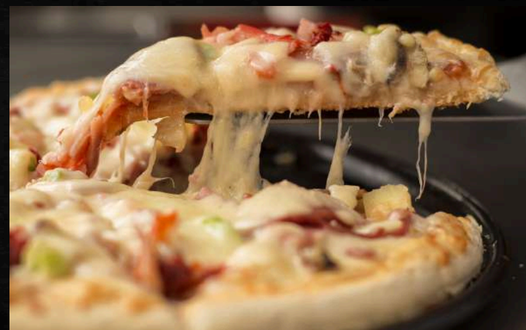


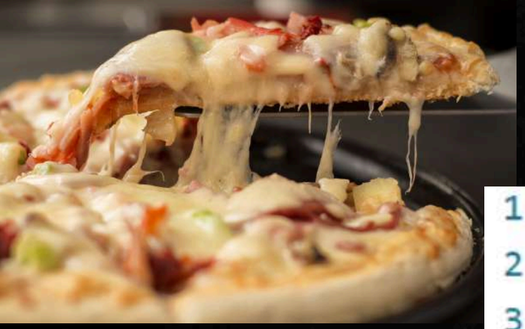


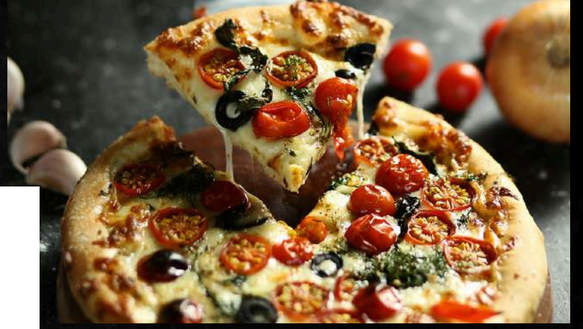
```sql
-- Identify the most common pizza size ordered.

SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid | Filter Rows:

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



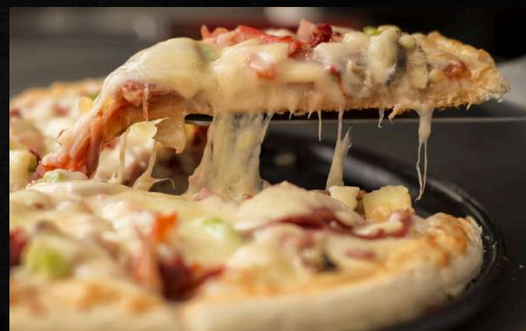


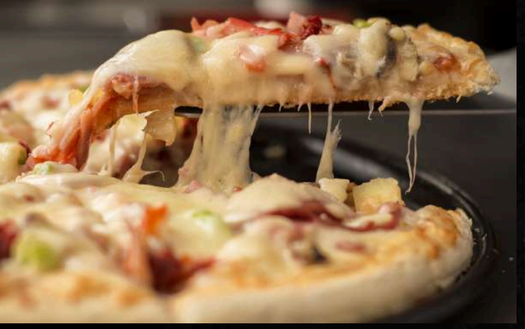


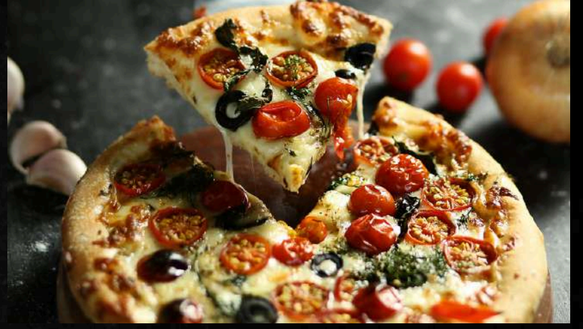
```sql
-- List the top 5 most ordered pizza types
-- along with their quantities.

SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
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



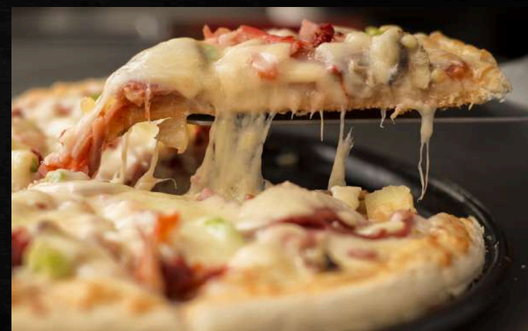


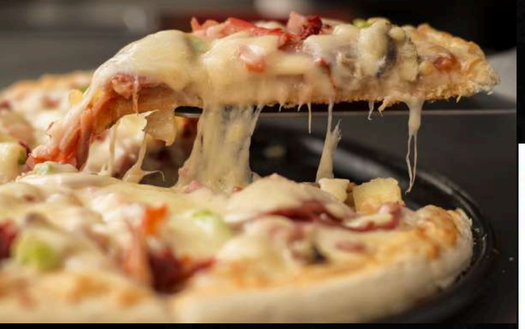


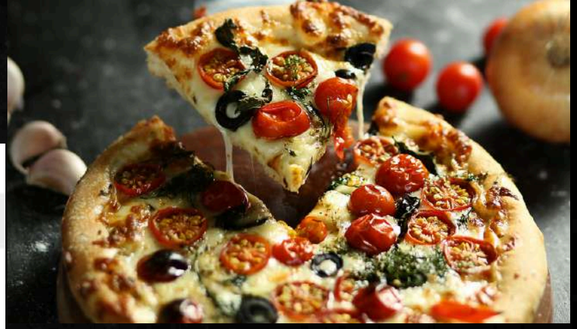
```sql
-- Join the necessary tables to find the
-- total quantity of each pizza category ordered.

SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

Result Grid | Filter

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050



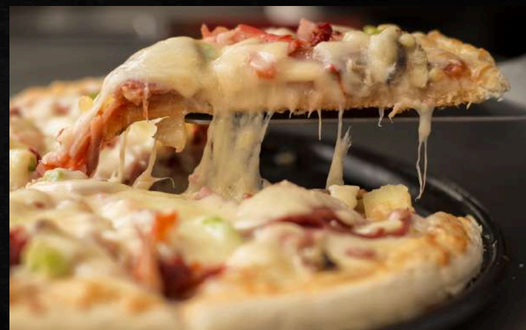


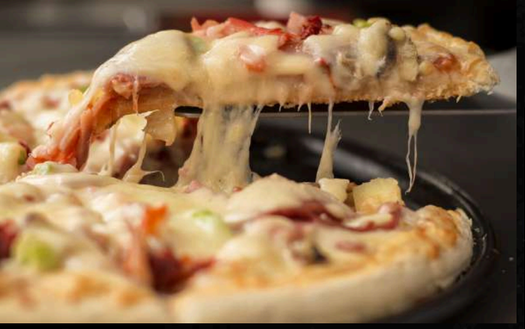


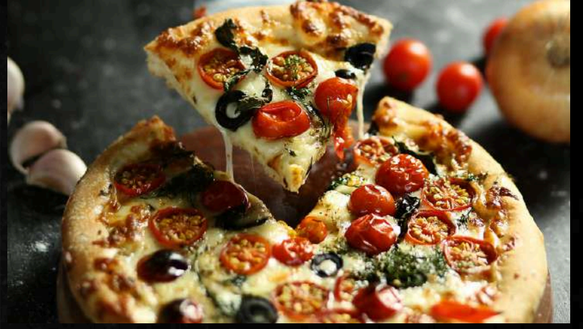
```sql
-- Determine the disribution of orders by hour of the day

SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid

hour	order_count
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
10	8
9	1



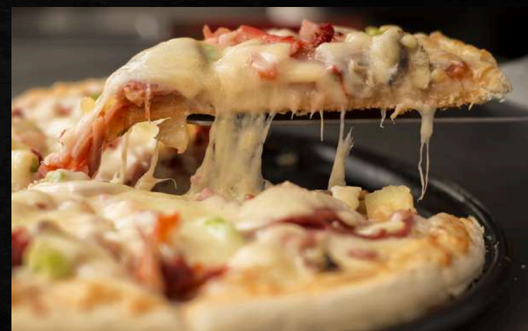


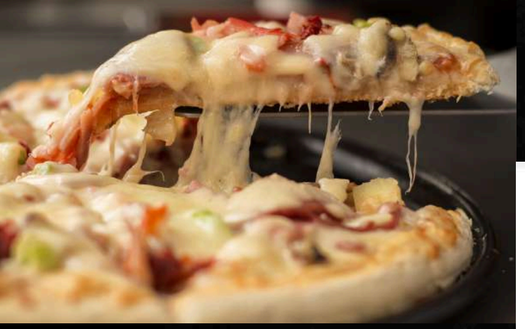


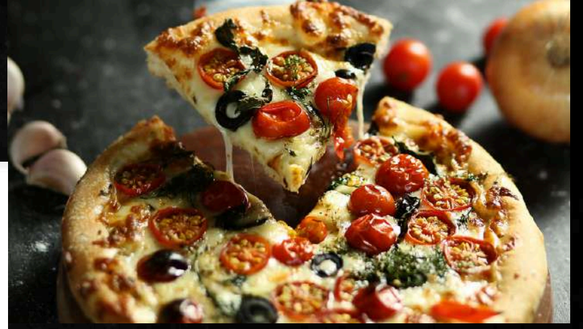
```sql
-- Join relevant tables to find the
-- category-wise distribution of pizzas.

SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

Result Grid | Filter Rows:

category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9



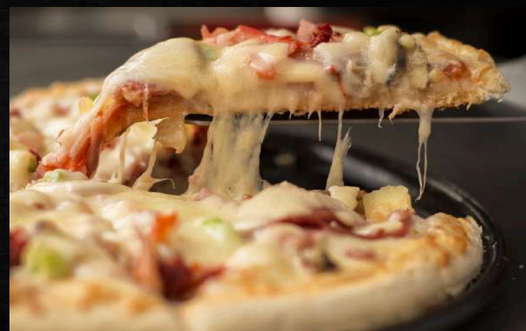


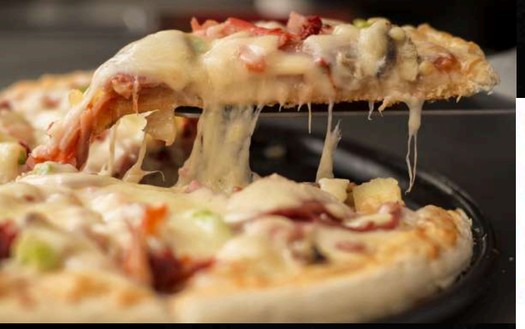


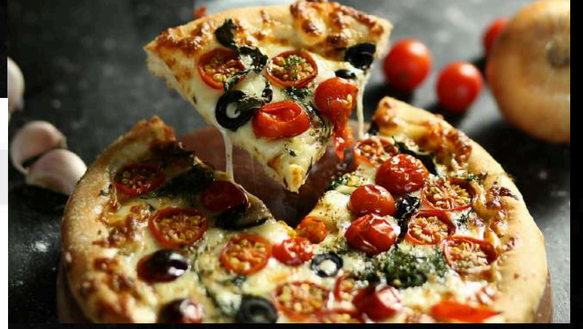
```sql
-- Group the orders by date and calculate the average
-- number of pizzas ordered per day.
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id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

Result Grid | Filter Ro

	ROUND(AVG(quantity), 0)
▶	138



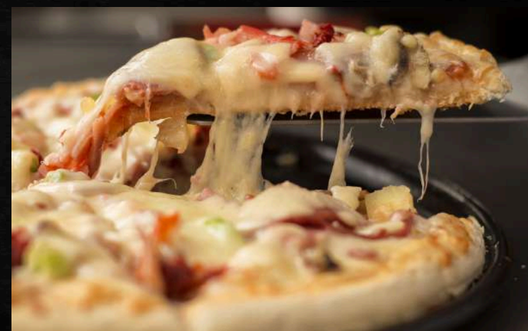


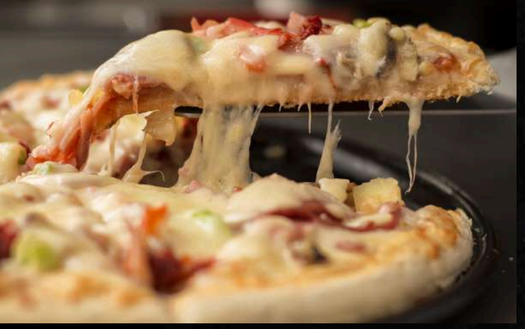


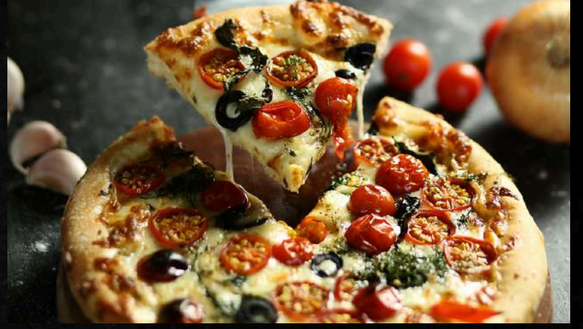
```sql
-- Determine the top 3 most ordered pizza types based on revenue.
SELECT
    pizza_types.name, SUM(quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id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



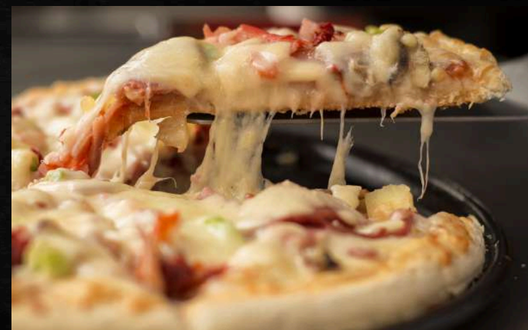


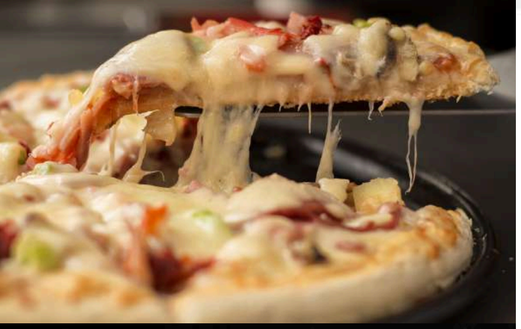


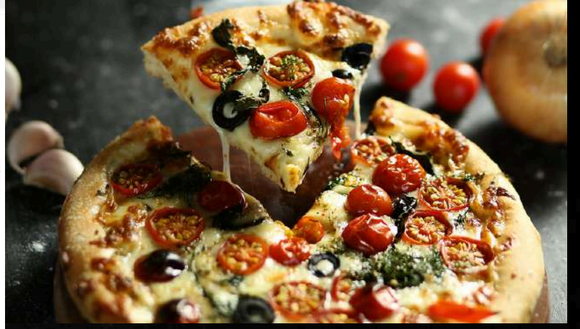
```sql
-- Calculate the percwntage contribution of each pizza
-- type to total revenue.

SELECT
    pizza_types.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) * 100,
            2) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

Result Grid | Filter Rc

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68



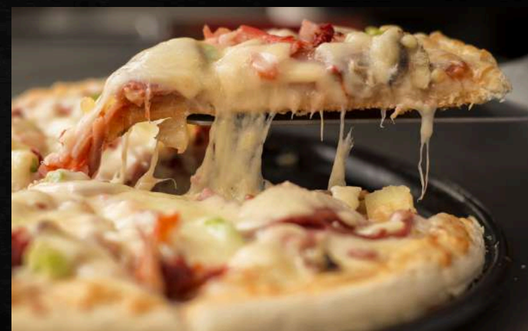


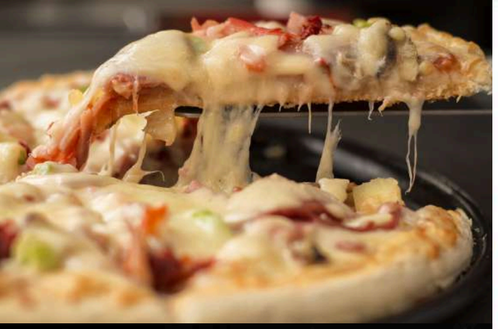


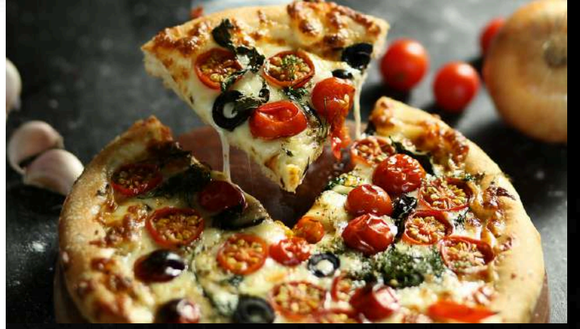
```sql
-- Analyse the cumulative revenue generated over time.
select order_date,
sum(revenue) over (order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity*pizzas.price) as revenue
from order_details join pizzas
on order_details.pizza_id=pizzas.pizza_id
join orders
on orders.order_id= order_details.order_id
group by orders.order_date) as sales;
```

Result Grid | Filter Rows:

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003



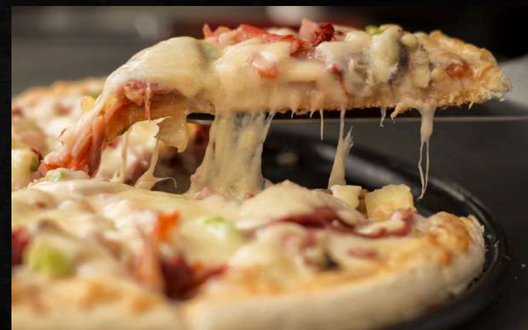


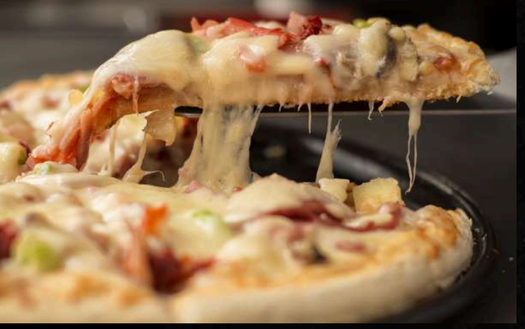


```sql
-- Determine thr top 3 most ordered pizza types
-- based on revenue for each pizza category.
select name, revenue from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as rn
from
(select pizza_types.category,pizza_types.name,
sum((order_details.quantity)*pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.category,pizza_types.name) as a) as b
 where rn<=3;
```

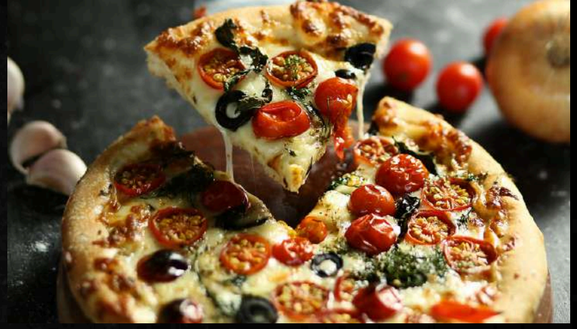
Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

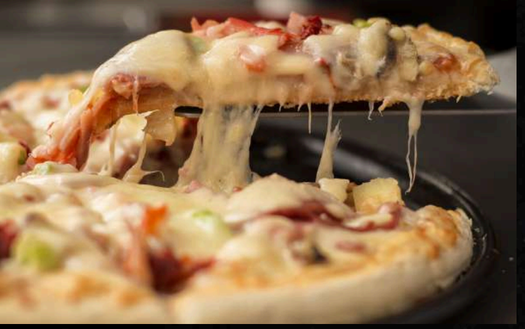




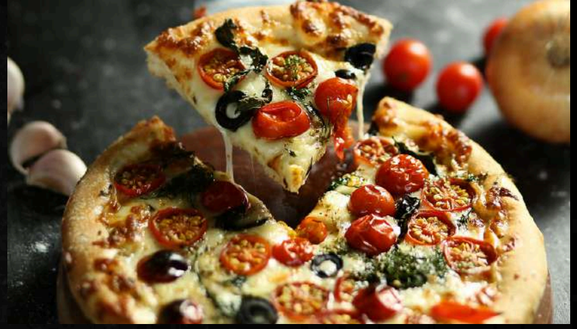
Conclusion & Key Takeaways



- 1. Total Orders Placed
  Result: 21,350 orders have been placed.
- 2. Total Revenue Generated.
  Result: ₹817,860.05 in revenue has been generated.
- 3. Pizza Category Popularity (Based on Quantity Ordered)
  Top-Selling Categories : Classic: 14,888 ,Supreme: 11,987, Veggie: 11,649 ,Chicken: 11,050
- Insight: Classic pizzas are the most popular category.
- 4. Order Distribution by Hour of the Day
- Peak Order Hours:
  - 12 PM: 2,520 orders
  - 1 PM: 2,455 orders
  - 6 PM: 2,336 orders
  - 5 PM: 2,399 orders
- Low Order Hours:
  - 9 AM to 10 AM: < 10 orders
  - After 10 PM: Drastically decreases



Conclusion & Key Takeaways



🍕 1. Top Revenue Generating Pizzas:

- The Thai Chicken Pizza: ₹43,434.25
- The Barbecue Chicken Pizza: ₹42,768
- The California Chicken Pizza: ₹41,409.5

Chicken pizzas dominate in terms of revenue generation.

📊 2. Pizza Category Distribution (Count-wise):

- Supreme & Veggie: 9 types each
- Classic: 8 types
- Chicken: 6 types

Supreme and Veggie offer the highest variety.

💰 3. Category-wise Revenue Contribution (%):

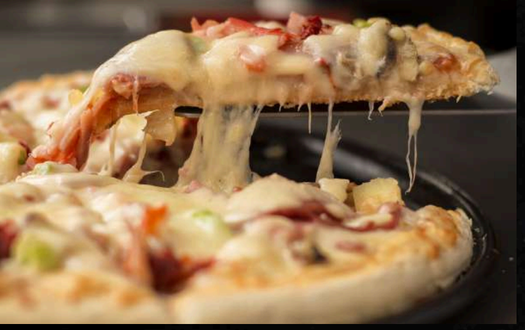
- Classic: 26.91% (Highest)
- Supreme: 25.46%
- Chicken: 23.96%
- Veggie: 23.68%

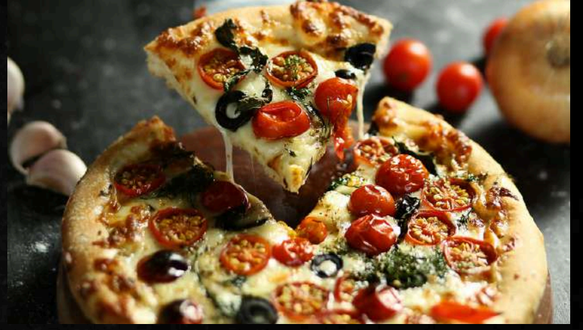
Classic pizzas lead in overall sales revenue.

🕒 4. Average Pizzas Ordered Per Day:

- 138 pizzas/day

Indicates strong and consistent customer demand.





🙏 Thank You!

For any questions or further analysis, contact:

Yogesh Gavhane – Data Analyst