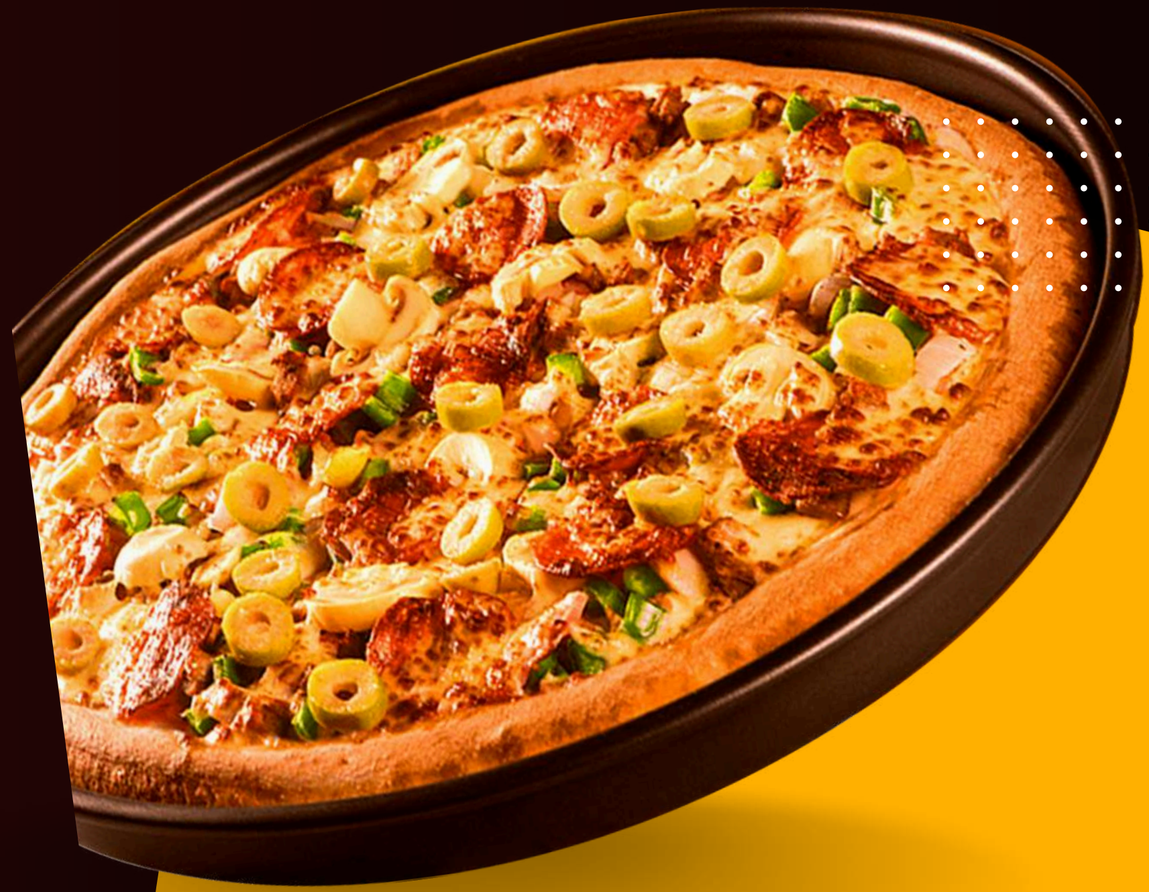
SQL PROJECT

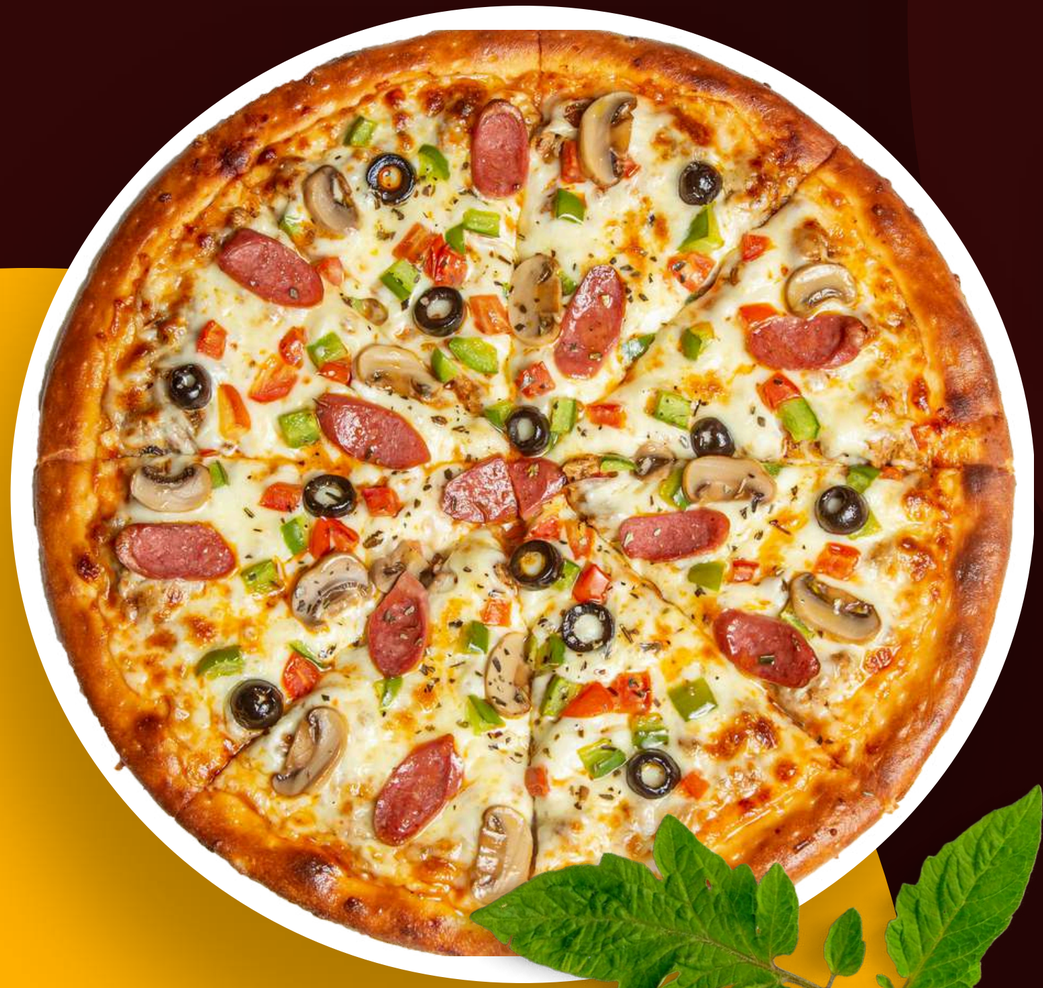
About Contact

# PIZZA SALES



SQL PROJECT

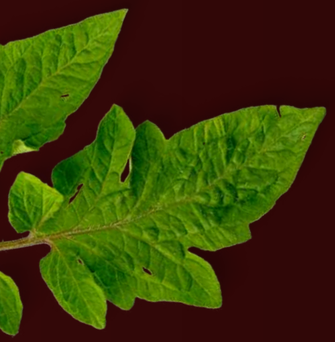
Home About Contact



# ABOUT PROJECT

Hi, My Name is Ashish, I have made a End to End SQL Project in which I have utilzed SQL Queries to solve SQL questions related to Pizza Sales.

# DATA SET INFO



I have used a sample data set downloaded from Kaggle which has four tables:

1-Order Details

2-Orders

3-Pizzas

4-Pizza Types

SQL PROJECT

# QUESTIONS FOR PROJECT

## Basic 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

## Intermediate Questions

1. Join the necessary tables to find the total quantity of each pizza category ordered.
2. Determine the distribution of orders by hour of the day.
3. Join relevant tables to find the category-wise distribution of pizzas.
4. Group the orders by date and calculate the average number of pizzas ordered per day.
5. Determine the top 3 most ordered pizza types based on revenue.

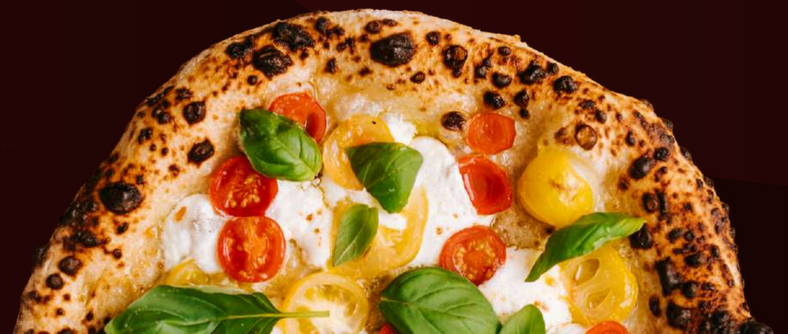
SQL PROJECT

Home About Contact

## Advanced Questions

1. Calculate the percentage contribution of each pizza type to total revenue.
2. Analyze the cumulative revenue generated over time.
3. Determine the top 3 most ordered pizza types based on revenue for each pizza category.

.



Retrieve the total number of orders placed.

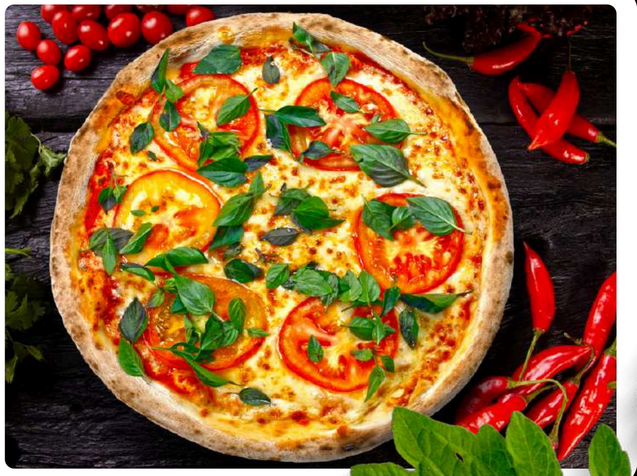
Select count(*) as num_of_orders from orders;

Result Grid

| num_of_orders |
| --- |
| 21350 |

SQL PROJECT

Home



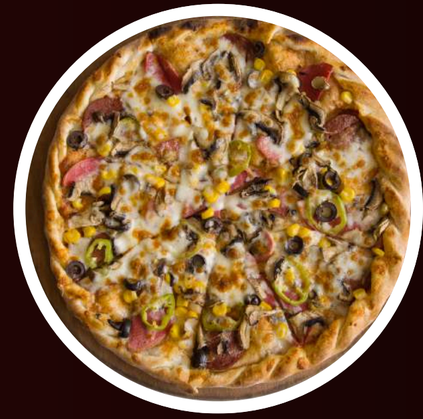
Calculate the total revenue generated from pizza sales.

```
select round(sum(price*quantity),2) as total_revenue from
pizzas inner join order_details on
order_details.pizza_id=pizzas.pizza_id;
```

Result Grid

| total_revenue |
| --- |
| 817860.05 |

SQL PROJECT



Identify the highest-priced pizza.

```
select name,price from pizzas inner join pizza_types on
pizza_types.pizza_type_id=pizzas.pizza_type_id order by
price desc limit 1;
```

| | name | price |
|---|---|---|
| ▶ | The Greek Pizza | 35.95 |

SQL PROJECT

Home About

Identify the most common pizza size ordered.

```
select size,count(size) as Size_ordered from pizzas inner join
order_details on order_details.pizza_id=pizzas.pizza_id
group by size order by Size_ordered desc limit 5;
```

| size | Size_ordered |
|---|---|
| L | 18526 |
| M | 15385 |
| S | 14137 |
| XL | 544 |
| XXL | 28 |

SQL PROJECT

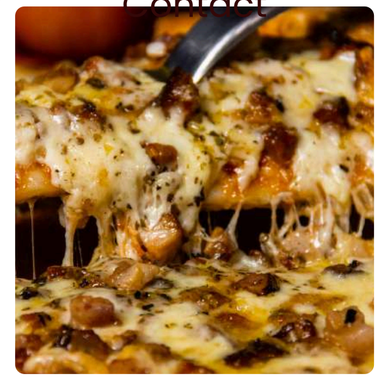
List the top 5 most ordered pizza types along with their quantities.

Select name, sum(quantity) as Quantity_count from order_details inner join pizzas on pizzas.pizza_id=order_details.pizza_id inner join pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id group by name order by Quantity_count desc limit 5;

| name | Quantity_count |
|---|---|
| The Classic Deluxe Pizza | 2453 |
| The Barbecue Chicken Pizza | 2432 |
| The Hawaiian Pizza | 2422 |
| The Pepperoni Pizza | 2418 |
| The Thai Chicken Pizza | 2371 |

SQL PROJECT

Home About Contact



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

```
select category,sum(quantity) as Tot_quantity from pizzas inner join
order_details on order_details.pizza_id=pizzas.pizza_id inner join
pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id group by
category;
```

| category | Tot_quantity |
|---|---|
| Classic | 14888 |
| Veggie | 11649 |
| Supreme | 11987 |
| Chicken | 11050 |

SQL PROJECT

Determine the distribution of orders by hour of the day.

```
select count(order_id) as No_of_order,date,hour(time)
as hourly from orders group by date,hourly;
```

| No_of_order | date | hourly |
|---|---|---|
| 2 | 2015-01-01 | 11 |
| 7 | 2015-01-01 | 12 |
| 10 | 2015-01-01 | 13 |
| 7 | 2015-01-01 | 14 |
| 7 | 2015-01-01 | 15 |
| 4 | 2015-01-01 | 16 |

SQL PROJECT

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

```
select category, count(order_details_id) as distribution from pizza_types inner join pizzas on pizzas.pizza_type_id=pizza_types.pizza_type_id inner join order_details on order_details.pizza_id=pizzas.pizza_id group by category;
```

| | category | distribution |
|---|---|---|
| ▶ | Classic | 14579 |
| | Veggie | 11449 |
| | Supreme | 11777 |
| | Chicken | 10815 |

SQL PROJECT

Group the orders by date and calcu the average number of pizzas ordered per day.

Select date,round(avg(order_details_id)) as avg_order from orders inner join order_details on order_details.order_id=orders.order_id group by date;

| date | avg_order |
|---|---|
| 2015-01-01 | 81 |
| 2015-01-02 | 242 |
| 2015-01-03 | 399 |
| 2015-01-04 | 529 |
| 2015-01-05 | 642 |
| 2015-01-06 | 775 |

SQL PROJECT

Determine the top 3 most ordered pizza types based on revenue.

```
select pizza_types.pizza_type_id,
sum(order_details.quantity*price) as tot_revenue from
pizza_types inner join pizzas on
pizzas.pizza_type_id=pizza_types.pizza_type_id inner join
order_details on order_details.pizza_id=pizzas.pizza_id group
by pizza_types.pizza_type_id order by tot_revenue desc limit 3;
```

| pizza_type_id | tot_revenue |
|---|---|
| thai_ckn | 43434.25 |
| bbq_ckn | 42768 |
| cali_ckn | 41409.5 |

SQL PROJECT

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

```
Select pizza_types.pizza_type_id,
round(sum(order_details.quantity*price)/(select
sum(price*order_details.quantity) from pizzas inner join
order_details on order_details.pizza_id=pizzas.pizza_id)*100,2) as
revenue from pizza_types inner join pizzas on
pizzas.pizza_type_id=pizza_types.pizza_type_id inner join order_details
on order_details.pizza_id=pizzas.pizza_id group by
pizza_types.pizza_type_id order by revenue desc;
```

| pizza_type_id | revenue |
|---|---|
| thai_ckn | 5.31 |
| bbq_ckn | 5.23 |
| cali_ckn | 5.06 |
| classic_dlx | 4.67 |
| spicy_ital | 4.2 |

SQL PROJECT

# Analyze the cumulative revenue generated over time.

```
Select date,round(sum(revenue)over(order by date),2) as
Cum_revenue from
(Select date,sum(order_details.quantity*price) as revenue from
pizzas inner join order_details on
order_details.pizza_id=pizzas.pizza_id
inner join orders on orders.order_id=order_details.order_id group
by date) as sales;
```

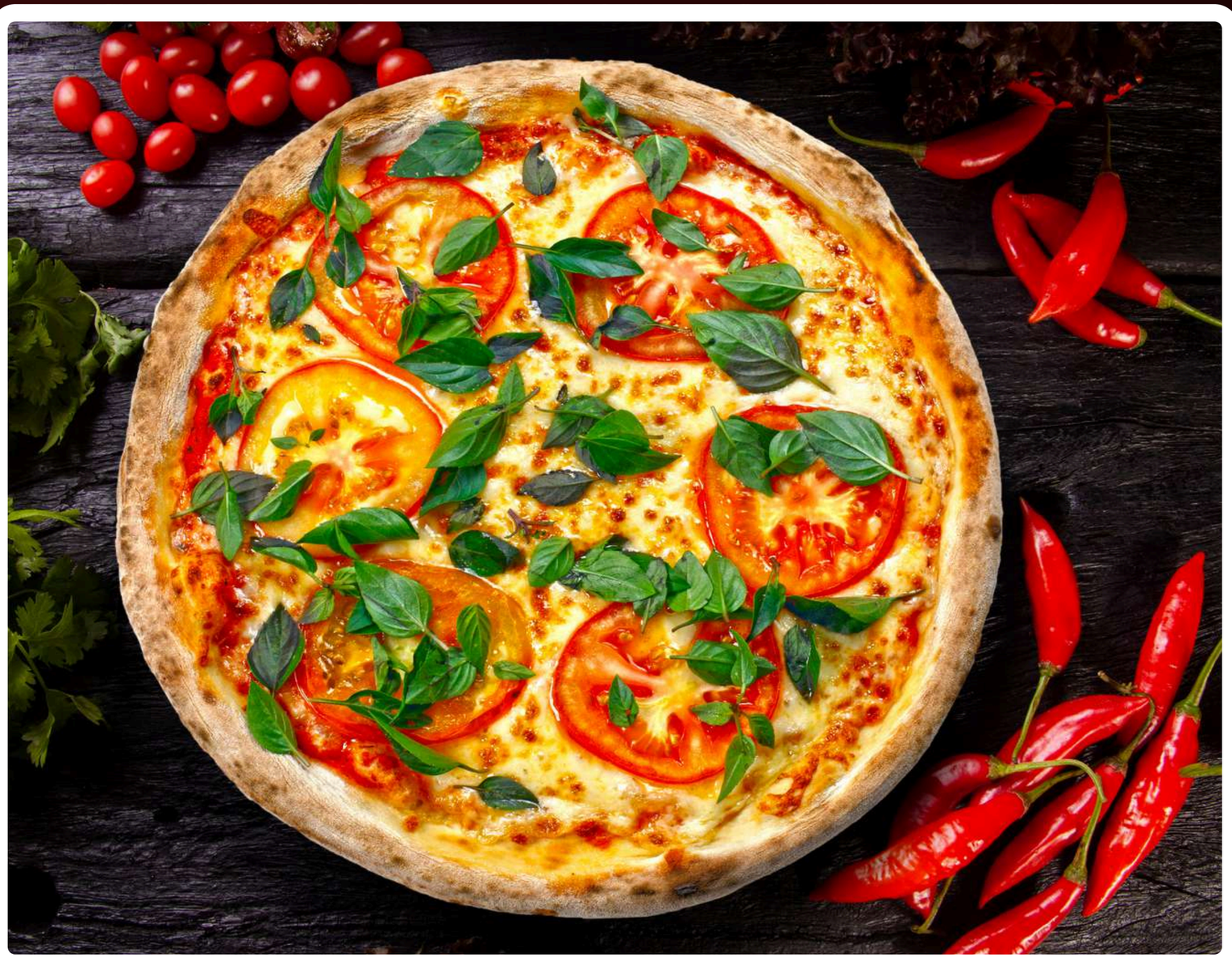
| date | Cum_revenue |
|---|---|
| 2015-01-01 | 2713.85 |
| 2015-01-02 | 5445.75 |
| 2015-01-03 | 8108.15 |
| 2015-01-04 | 9863.6 |

SQL PROJECT

Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
Select name,category,revenue from
(Select name,category,revenue, rank () over (partition by category order by revenue desc) as rev_by_category from
(select pizza_types.category,pizza_types.pizza_type_id,pizza_types.name, sum(order_details.quantity*price) as revenue from pizza_types
inner join pizzas on pizzas.pizza_type_id=pizza_types.pizza_type_id inner join order_details on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.category,pizza_types.pizza_type_id,pizza_types.name) as cate) as cate2 where rev_by_category<=3;
```

| name | category | revenue |
|---|---|---|
| The Thai Chicken Pizza | Chicken | 43434.25 |
| The Barbecue Chicken Pizza | Chicken | 42768 |
| The California Chicken Pizza | Chicken | 41409.5 |



# THANK YOU