



PIZZA

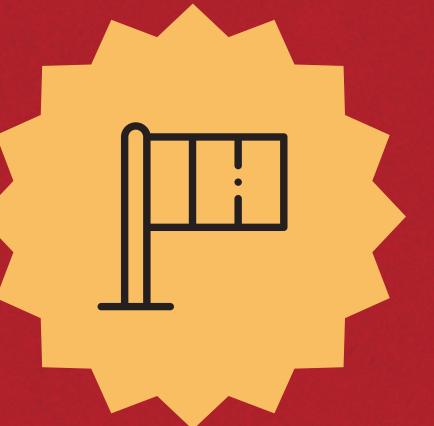
Pizza sales Project



Hello !

My Self Pavan Kumar I Have Done and utilize SQL queries to solve Questions that were related to Pizza Sales





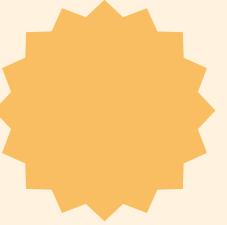
Pizza Sales

We Love Pizzas

Project



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.

```
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
▶	49874.55

Identify the highest-priced pizza.



```
Workbench will automatically  
refresh when you make changes.  
  
SELECT pizza_types.name, pizzas.price  
FROM pizzas  
JOIN pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
ORDER BY pizzas.price DESC  
LIMIT 1
```



Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

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
```



	size	order_count
▶	L	1127
	M	934
	S	869
	XL	36
	XXL	2

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 Hawaiian Pizza	168
	The Thai Chicken Pizza	159
	The California Chicken Pizza	139
	The Classic Deluxe Pizza	137
	The Pepperoni Pizza	136

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

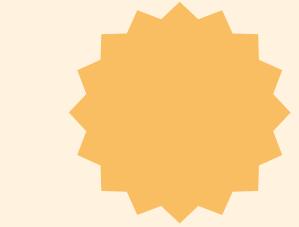
```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC
```



Result Grid | Filter R

	category	quantity
▶	Classic	925
	Veggie	733
	Supreme	715
	Chicken	651

Determine the distribution 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



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

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

```
SELECT  
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day  
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_quantity
```



Result Grid	
	avg_pizza_ordered_per_day
▶	138

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

```
SELECT  
    pizza_types.name,  
    SUM(order_details.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
```



	name	revenue
▶	The Thai Chicken Pizza	2863.25
	The California Chicken Pizza	2384.25
	The Hawaiian Pizza	2288.5

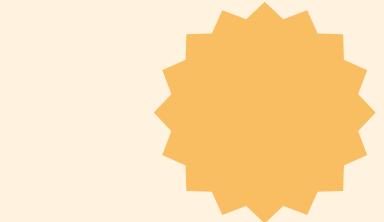
Calculate the percentage contribution of each pizza type to total revenue.

```
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
```



	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

Analyze 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  
    orders.order_id = order_details.order_id  
group by  
  orders.order_date) as sales
```



	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



A PICTURE IS WORTH A THOUSAND WORDS



THANKS!

Do you have any questions?

pavankumarmuthyala39@gmail.com

+9381773234

<https://www.linkedin.com/in/pavan-kumar-muthyala>

