



Delicious Pizza for Everyone!

PIZZA BOXCAR

@reallygreatsite



hi

this is mansi,in this sql project i have utilized sql queries
to solve the questions were related to pizza sales

```
-- 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

```
    and(sum(order_details.quantity * pizzas.price),2) as total_sales  
from order_details join pizzas  
pizzas.pizza_id = order_details.pizza_id
```

Result Grid | Filter

total_sales
817860.05

```
1 -- Identify the highest-priced pizza.  
2  
3 • select pizza_types.name, pizzas.price  
4   from pizza_types join pizzas  
5     on pizza_types.pizza_type_id = pizzas.pizza_type_id  
6   order by pizzas.price desc limit 1;  
7
```

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;
```

Result Grid | Filter Rows

	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.  
  
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  
  
limit 5;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

```
1 -- Join the necessary tables to find the
2 -- total quantity of each pizza category ordered.
3
4• select pizza_types.category,
5   sum(order_details.quantity) as quantity
6   from pizza_types join pizzas
7   on pizza_types.pizza_type_id = pizzas.pizza_type_id
8   join order_details           I
9   on order_details.pizza_id = pizzas.pizza_id
10  group by pizza_types.category order by quantity desc;
```

Result Grid

	category	quantity
▶	Classic	14888
▶	Supreme	11987
▶	Veggie	11649
▶	Chicken	11050

```
1  -- Determine the distribution of orders by hour of the day.  
2 • SELECT  
3      HOUR(order_time) AS hour, COUNT(order_id) AS order_count  
4  FROM  
5    orders  
6 GROUP BY HOUR(order_time);
```

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

```
1 -- Join relevant tables to find the category-wise distribution of pizzas.  
2 • SELECT  
3     category, COUNT(name)  
4 FROM  
5     pizza_types  
6 GROUP BY category;
```

	category	COUNT(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Weggie	9

```
-- 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_quatity;
```

ROUND(AVG(quantity), 0)

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	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5