

SQL Project On Pizza Sales

BY
U m a i r N o o r

...

Hello !

My name is **Umair Noor** and
in this project, I have utilized
SQL queries to solve
questions that is related to
pizza sales



PROBLEMS

Retrieve the total number of orders placed.

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

	COUNT(order_id)
▶	21350

...

Calculate the total revenue generated from pizza sales.

```
SELECT  
    ROUND(SUM(orders_details.quantity * pizzas.price)) AS Total_Revenue  
FROM  
    orders_details  
    JOIN  
    pizzas ON orders_details.pizza_id = pizzas.pizza_id;
```

	AS Total_Revenue
▶	817860

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 5;
```

	name	price
▶	The Greek Pizza	35.95
	The Greek Pizza	25.5
	The Brie Carre Pizza	23.65
	The Italian Vegetables Pizza	21
	The Barbecue Chicken Pizza	20.75

Identify the most common pizza size ordered.

```
SELECT
    pizzas.size, COUNT(order_details_id) AS Most_Common_Pizza
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size;
```

	size	Most_Common_Pizza
▶	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.name, SUM(orders_details.quantity) AS Quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizza_types.name
ORDER BY Quantity DESC
LIMIT 5;
```

	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.

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

	name	total_quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371
	The California Chicken Pizza	2370
	The Sicilian Pizza	1938
	The Spicy Italian Pizza	1924
	The Southwest Chicken Pizza	1917
	The Big Meat Pizza	1914

Determine the distribution of orders by hour of the day.

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

	HOUR(order_time)	COUNT(order_id)
▶	9	1
	10	8
	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920

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

SELECT

```
    pizza_types.name,  
    SUM(orders_details.quantity) AS total_quantity  
FROM  
    pizza_types  
        JOIN  
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
        JOIN  
    orders_details ON orders_details.pizza_id = pizzas.pizza_id  
GROUP BY pizza_types.name  
ORDER BY total_quantity DESC;
```

...

	name	total_quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371
	The California Chicken Pizza	2370
	The Sicilian Pizza	1938

Determine the distribution of orders by hour of the day..

SELECT

```
HOUR(order_time), COUNT(order_id)  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

	HOUR(order_time)	COUNT(order_id)
▶	9	1
	10	8
	11	1231
	12	2520
	13	2455
	14	1472
	15	1468

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

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

	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  
    AVG(quantity)  
FROM  
    (SELECT  
        orders.order_date, SUM(orders_details.quantity) AS quantity  
    FROM  
        orders  
    JOIN orders_details ON orders.order_id = orders_details.order_id  
    GROUP BY orders.order_date) AS order_quantity;
```

	Avg(quantity)
▶	138.4749

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

```
SELECT  
    pizza_types.name,  
    SUM(orders_details.quantity * pizzas.price) AS revenue  
FROM  
    pizza_types  
        JOIN  
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
        JOIN  
    orders_details ON orders_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

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

...

```
SELECT
    pizza_types.category,
    ROUND(SUM(orders_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(orders_details.quantity * pizzas.price),
        2) AS total_sales
    )
    FROM
        orders_details
        JOIN
            pizzas ON orders_details.pizza_id = pizzas.pizza_id) * 100,
    2) AS revenue
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
        orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category;
```

	category	revenue
▶	Chicken	23.96
	Classic	26.91
	Supreme	25.46
	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(orders_details.quantity * pizzas.price) AS revenue
    FROM
      orders_details
      JOIN
      pizzas ON orders_details.pizza_id = pizzas.pizza_id
      JOIN
      orders ON orders.order_id = orders_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
	2015-01-07	16560.7
	2015-01-08	19399.05
	2015-01-09	21526.4
	2015-01-10	23990.35000000002
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.30000000003
	2015-01-14	32358.70000000004
	2015-01-15	34343.50000000001

Determine the 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 _rank
    FROM
  > (SELECT pizza_types.name , pizza_types.category , SUM(orders_details.quantity*pizzas.price) AS revenue
      FROM pizza_types JOIN pizzas ON pizza_types.pizza_type_id= pizzas.pizza_type_id
      JOIN orders_details ON orders_details.pizza_id= pizzas.pizza_id
      - GROUP BY pizza_types.name , pizza_types.category) as a) as b
  WHERE _rank<=3 ;
```

	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	32265.70000000065

...

THANK YOU