



SQL PROJECT

Data analysis

PIZZA SALES ANALYSIS





SQL PROJECT

Data analysis



HELLO !

Hi, I'm Deepanshu a Data Analyst passionate about turning raw data into clear, meaningful insights. I am currently building strong skills in SQL, Excel, Python, and Power BI. This Pizza Sales Analysis project reflects my ability to work with relational databases, write efficient SQL queries, and solve real business problems through data. I enjoy exploring patterns, optimizing queries, and using data to support decision-making





SQL PROJECT

Data analysis

ALL ABOUT DATASET

I used four table :

- order
- order_details
- pizzas
- pizza_types

- **Table : orders**

Columns:

1.order_id	int PK
2.order_date	date
3.order_time	time

- **Table : order_details**

Columns:

1. order_details_id	int PK
2. order_id	int
3.pizza_id	text
4.quantity	text

- **Table : pizzas**

Columns:

1.pizza_id	text
2.pizza_type_id	text
3.size	text
4.price	double

- **Table : pizza_types**

Columns:

1.pizza_type_id	text
2.name	text
3.category	text
4.ingredients	text





SQL PROJECT

Data analysis



OBJECTIVES:

BASIC:

- RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.
- CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.
- IDENTIFY THE HIGHEST-PRICED PIZZA.
- IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.
- LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

INTERMEDIATE:

- JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.
- DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.
- JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.
- GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.
- DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

ADVANCED:

- CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.
- ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.
- DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

Q1. RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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

Result Grid |   Fit

	total_orders
▶	21350



Q2. CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```
SELECT  
    ROUND(SUM(o.quantity * p.price), 2) AS Total_revenue  
FROM  
    order_details o  
    JOIN  
    pizzas p ON p.pizza_id = o.pizza_id;
```

Result Grid	
	Total_revenue
▶	817860.05



Q3. IDENTIFY THE HIGHEST-PRICED PIZZA

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

Result Grid		Filter Row
	name	price
▶	The Greek Pizza	35.95



Q4. IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

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

	size	Total_orderBy_size
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

Q5. LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```
SELECT pt.name, SUM(od.quantity) AS quantity
FROM pizza_types pt
JOIN pizzas p ON pt.pizza_type_id = p.pizza_type_id
JOIN order_details od ON p.pizza_id = od.pizza_id
GROUP BY pt.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



SQL PROJECT

Data analysis

Q6. JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED

```
SELECT pt.category, SUM(od.quantity) AS total_quantity
FROM pizza_types pt
JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id
JOIN order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.category;
```

	category	total_quantity
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050



SQL PROJECT

Data analysis

Q7. DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT  
    HOUR(order_time) AS Hours, COUNT(order_id) AS total_orders  
FROM  
    orders  
GROUP BY Hours  
ORDER BY total_orders DESC;
```

	Hours	total_orders
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	21	1198
	22	663
	23	28
	10	8
	9	1

Q8. JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```
SELECT category, COUNT(name) AS total_pizzas
FROM pizza_types
GROUP BY category
ORDER BY total_pizzas DESC;
```

	category	total_pizzas
▶	Supreme	9
▶	Veggie	9
▶	Classic	8
▶	Chicken	6



SQL PROJECT

Data analysis

Q9. GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

```
with Temp as
(
SELECT
    order_date, SUM(quantity) AS sum_of_Orders
FROM
    orders o
    JOIN
    order_details od ON o.order_id = od.order_id
GROUP BY order_date)

SELECT
    ROUND(AVG(sum_of_Orders), 2) AS Avg_ordered_per_day
FROM
    Temp;
```



Result Grid		Filter Rows:
		Avg_ordered_per_day
▶		138.47



SQL PROJECT

Data analysis

Q10. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
SELECT pt.name, SUM(od.quantity * p.price) AS revenue
FROM pizza_types pt
JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id
JOIN order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.name
ORDER BY revenue DESC
LIMIT 3;
```



Result Grid		
	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5



SQL PROJECT

Data analysis

Q11. CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```
WITH temp AS (
    SELECT pt.category AS category, SUM(od.quantity * p.price) AS revenue
    FROM pizza_types pt
    JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id
    JOIN order_details od ON od.pizza_id = p.pizza_id
    GROUP BY pt.category
),
total AS ( SELECT SUM(revenue) AS total_revenue
            FROM temp
        )
SELECT
    t.category,
    ROUND(t.revenue * 100.0 / tot.total_revenue, 2) AS per_of_revenue
FROM
    temp t
    CROSS JOIN
    total tot;
```

Result Grid		Filter Rows:
	category	per_of_revenue
▶	Classic	26.91
	Veggie	23.68
	Supreme	25.46
	Chicken	23.96



SQL PROJECT

Data analysis

Q12. ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
Select order_date, round(sum(revenue) over( order by order_date),2) as cum_revenue  
from  
(SELECT o.order_date, SUM(od.quantity * p.price) as revenue  
FROM order_details od  
join pizzas p on p.pizza_id = od.pizza_id  
join orders o on o.order_id = od.order_id  
group by o.order_date) as sales;
```

	order_date	cum_revenue
▶	2015-01-01	2713.85
	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	23000.55





SQL PROJECT

Data analysis

Q13. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```
Select category, name, revenue, Ranks  
from  
(Select category, name, revenue,  
rank() over(partition by category order by revenue desc) as Ranks  
from  
(Select pt.category, pt.name, sum(od.quantity * p.price) as revenue  
FROM pizza_types pt  
join pizzas p on p.pizza_type_id = pt.pizza_type_id  
join order_details od on od.pizza_id = p.pizza_id  
group by pt.category, pt.name) as a) as b  
where Ranks <= 3;
```

	category	name	revenue	Ranks
▶	Chicken	The Thai Chicken Pizza	43434.25	1
	Chicken	The Barbecue Chicken Pizza	42768	2
	Chicken	The California Chicken Pizza	41409.5	3
	Classic	The Classic Deluxe Pizza	38180.5	1
	Classic	The Hawaiian Pizza	32273.25	2
	Classic	The Pepperoni Pizza	30161.75	3
	Supreme	The Spicy Italian Pizza	34831.25	1
	Supreme	The Italian Supreme Pizza	33476.75	2
	Supreme	The Sicilian Pizza	30940.5	3
	Veggie	The Four Cheese Pizza	32265.70000000065	1
	Veggie	The Mexicana Pizza	26780.75	2
	Veggie	The Five Cheese Pizza	26066.5	3



Data analysis

THANK YOU FOR ATTENTION

dahiya2131@gmail.com

[LinkedIn.com](#)

[GitHub.com](#)