



PIZZA SALES

"PIZZA SALES ANALYSIS WITH MYSQL QUERIES" "A STRUCTURED APPROACH TO SQL FOR PIZZA SALES INSIGHTS"



CHANDAN



MYSQL



CHANDAN



THIS PROJECT PRESENTS A SERIES OF SQL QUERIES DESIGNED TO ANALYZE PIZZA SALES DATA, RANGING FROM BASIC ORDER RETRIEVAL TO ADVANCED REVENUE ANALYSIS. IT IS STRUCTURED INTO THREE LEVELS—BASIC, INTERMEDIATE, AND ADVANCED—TO PROGRESSIVELY EXPLORE SQL FUNCTIONALITIES."

Analyze pizza sales data using SQL to derive insights for improving business decisions.

Goals:

- Identify sales trends and performance metrics
- Analyze customer behavior patterns
- Support decision-making with data-driven insights

Pizza Sales Data Analysis Using SQL Unlocking Business Insights from Sales Data



DATASET DESCRIPTION

TABLES USED:

- orders: Contains order timestamps
- order_details: Items and quantities in each order
- pizzas: Prices and size information
- pizza_types: Category and type of pizza



CHANDAN



MYSQL

KEY QUESTIONS ANSWERED



- How many orders were placed?
- What's the total revenue generated?
- Which pizza is the highest-priced?
- Which pizza size is most popular?
- What are the top 5 most ordered pizza types?
- What's the revenue share by category?
- What time of day are orders most common?

MYSQL





◆ BASIC ANALYSIS

✓ Key Insights & Questions Solved

- Total number of orders placed
- Total revenue generated from pizza sales
- Highest-priced pizza
- Most common pizza size ordered
- Top 5 most ordered pizza types with quantities



CHANDAN





RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED



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

MYSQL



CHANDAN

Result Grid	
	total_orders
▶	21350

\$6







IDENTIFY THE HIGHEST-PRICED PIZZA.



MYSQL

```
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 1;
```

Result Grid   Filter Rows:		
	name	price
	The Greek Pizza	35.95



CHANDAN



MYSQL





CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
          1) AS total_sales
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
```

MYSQL

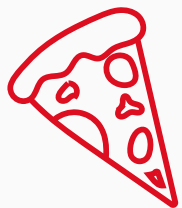


CHANDAN

Result Grid	
	total_sales
▶	817860

\$6





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;
```



MYSQL



CHANDAN

Result Grid				Filter
	size	order_count		
▶	L	18526		
	M	15385		
	S	14137		
	XL	544		
	XXL	28		




MYSQL





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

MYSQL

Result Grid  Filter Rows: <input type="text"/>		
	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

```
SELECT
    pizza_types.name,
    SUM(order_details.quantity) AS total_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 total_quantity DESC
LIMIT 5;
```



CHANDAN

\$6





◆ INTERMEDIATE ANALYSIS

- Total quantity ordered per pizza category
- Order distribution by hour of the day
- Category-wise distribution of pizzas
- Average number of pizzas ordered per day
- Top 3 most ordered pizza types by revenue



CHANDAN





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

MYSQL

Result Grid			Filter
	category	quantity	
	Classic	14579	
	Veggie	11449	
	Supreme	11777	
	Chicken	10815	

```
SELECT
    pizza_types.category,
    COUNT(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;
```



CHANDAN

\$6







DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT
    HOUR(order_time) AS hourly_time,
    COUNT(orders.order_id) AS order_count
FROM
    orders
GROUP BY HOUR(orders.order_time)
ORDER BY order_count DESC
LIMIT 5;
```



Result Grid   Filter Rows:		
	hourly_time	order_count
	12	2520
	13	2455
	18	2399
	17	2336
	19	2009

MYSQL



CHANDAN



MYSQL



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

```
select category , count(name)
from pizza_types
group by category;
```



CHANDAN

MYSQL

Result Grid Filter		
	category	quantity
	Classic	14579
	Veggie	11449
	Supreme	11777
	Chicken	10815

\$6







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 order_date) AS order_quantity;
```

Result Grid   Filter Rows:		
	hourly_time	order_count
	12	2520
	13	2455
	18	2399
	17	2336
	19	2009



CHANDAN



MYSQL





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



```
SELECT
    pizza_types.name,
    SUM(pizzas.price * order_details.quantity) 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;
```



MYSQL



CHANDAN

Result Grid   Filter Rows: <input data-bbox="2965 1210 3282 1322" type="text"/>		
	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5



 **MYSQL**

▼ **ADVANCED ANALYSIS**

- Percentage contribution of each pizza type to total revenue
- Cumulative revenue generated over time
- Top 3 revenue-generating pizza types per category

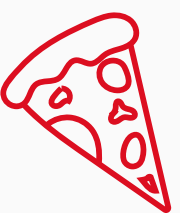


 **MYSQL**



CHANDAN





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



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



MYSQL

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68



CHANDAN







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, round(sum(order_details.quantity * pizzas.price),0) as revenue
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

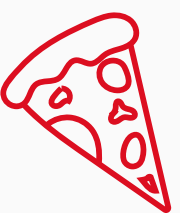


Result Grid   Filter Rows:		
	order_date	cum_revenue
▶	2015-01-01	2714
	2015-01-02	5446
	2015-01-03	8108
	2015-01-04	9863
	2015-01-05	11929
	2015-01-06	14358

MYSQL



CHANDAN



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 rnk
from
(select pizza_types.category, pizza_types.name, sum(order_details.quantity * pizzas.price) 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, pizza_types.name) as a) as b
where rnk <=3;
```



CHANDAN

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



MYSQL





CHANDAN

THROUGH THESE SQL QUERIES, WE GAIN VALUABLE INSIGHTS INTO PIZZA SALES TRENDS, REVENUE GENERATION, AND CUSTOMER PREFERENCES. UNDERSTANDING THESE DATA POINTS HELPS IN OPTIMIZING BUSINESS STRATEGIES AND IMPROVING SERVICE DELIVERY.







CHANDAN



MY DETAILS

 7042904626

 chandan7042904626.com

 Delhi



PROJECT PIZZA



CHANDAN

THANK YOU!

