

MENU

ABOUT US

HOME

PIZZA PARADISE 

PIZZA PARADISE



SALES

NEXT >

INTRODUCTION



A COMPREHENSIVE STUDY ON PIZZA SALES DATA USING SQL QUERIES.

**THIS PROJECT EXPLORES THE USE OF SQL TO ANALYZE PIZZA SALES DATA.
BY LEVERAGING SQL QUERIES,
WE AIM TO UNCOVER INSIGHTS AND TRENDS IN PIZZA SALES.
HELPING BUSINESSES MAKE DATA-DRIVEN DECISIONS.**

**PRESENTED BY:
SOHAM SAWANT**

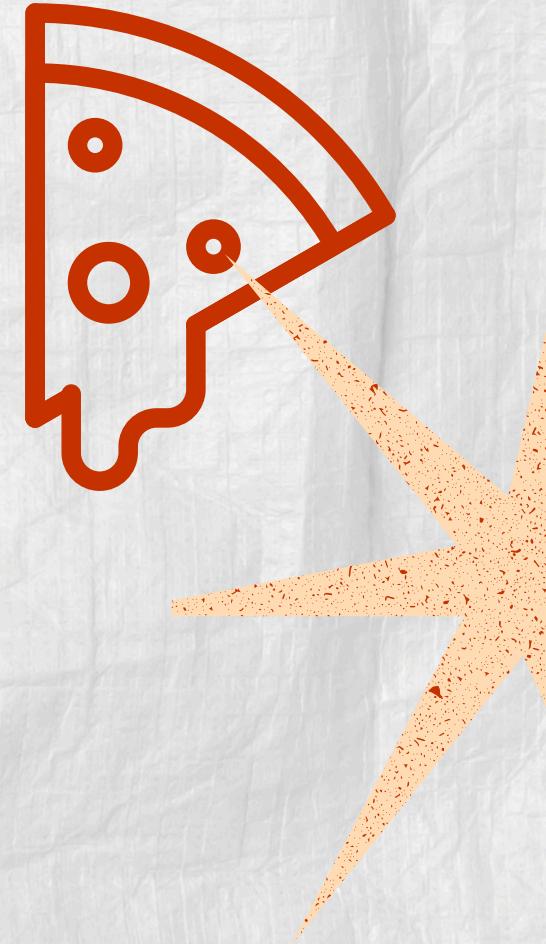
< BACK



RETRIVE THE TOTAL NUMBER OF ORDER REPLACE

```
use pizzahut;  
SELECT  
    COUNT(order_id) AS Total_order  
FROM  
    pizza_orders;
```

Result Grid	
	Total_order
▶	21350



< BACK

NEXT >

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
▶	593504.25

< BACK

NEXT >

IDENTIFY THE HIGHEST PRICE 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 1;
```

| Result Grid | Filter Rows

	name	price
▶	The Greek Pizza	35.95

< BACK

NEXT >

IDENTIFY THE MOST COMMON PIZZA ORDER



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;
```

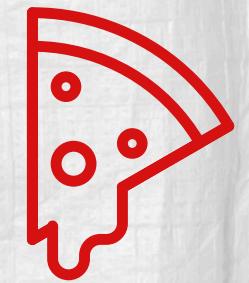
Result Grid |

	size	order_count
▶	L	13440
	M	11186
	S	10238
	XL	406
	XXL	19

< BACK

NEXT >

LIST THE MOST 5 MOST ORDERED PIZZA TYPES ALONG WITH THERE 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;
```

	name	quantity
▶	The Barbecue Chicken Pizza	1793
	The Pepperoni Pizza	1759
	The Classic Deluxe Pizza	1758
	The Hawaiian Pizza	1721
	The California Chicken Pizza	1716

< BACK

NEXT >

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 category
ORDER BY quantity DESC LIMIT 5;
```

	category	Quantity
▶	Classic	10773
	Supreme	8674
	Veggie	8531
	Chicken	7988

< BACK

NEXT >

DETERMINE THE DISTRIBUTION OF ORDER BY HOUR OF THE DAY.



SELECT

```
HOUR(order_time) AS hour, COUNT(order_id) AS order_count
```

FROM

```
pizza_orders
```

```
GROUP BY HOUR(order_time);
```

	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336

< BACK

NEXT >



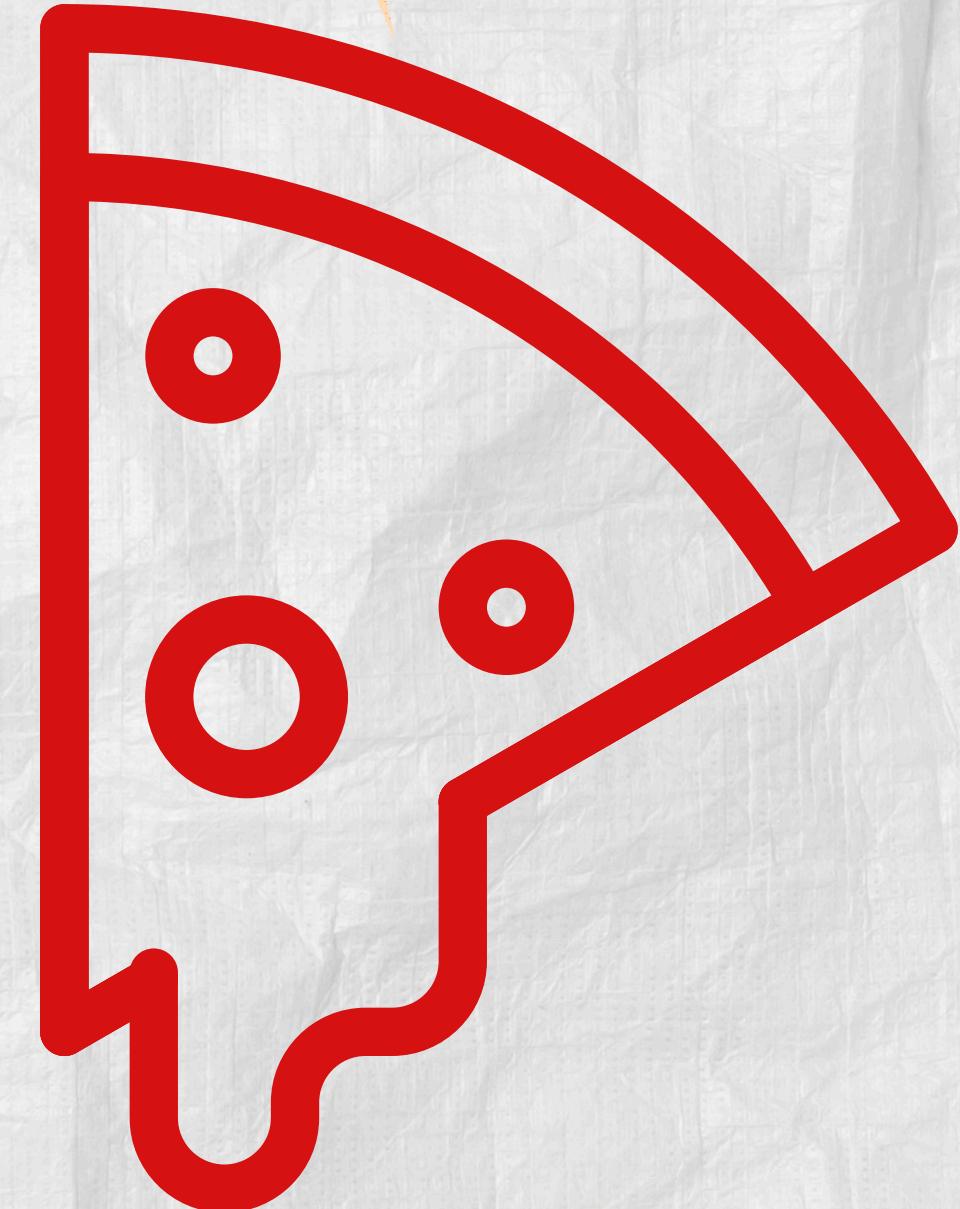
GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
SELECT  
    ROUND(AVG(quantity), 0) as pizza_ordered_per_day  
FROM  
(SELECT  
    pizza_orders.order_date,  
    SUM(order_details.quantity) AS quantity  
FROM  
    pizza_orders  
JOIN order_details ON pizza_orders.order_id = order_details.order_id  
GROUP BY pizza_orders.order_date) AS order_quantity;
```

pizza_ordered_per_day
138

< BACK

NEXT >



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

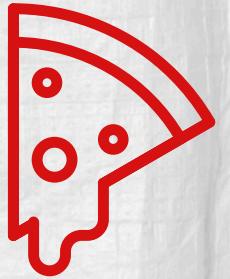
```
SELECT
    pizza_types.category,
    ROUND((SUM(order_details.quantity * pizzas.price) /
        (SELECT SUM(order_details.quantity * pizzas.price)
        FROM pizza_types
        JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN order_details ON order_details.pizza_id = pizzas.pizza_id)
        ) * 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	26.85
	Supreme	25.42
	Veggie	23.9
	Chicken	23.83

< BACK

NEXT >

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.



```
select order_date,  
sum(revenue) over (order by order_date) as cum_revenue  
from  
(select pizza_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 pizza_orders  
on pizza_orders.order_id = order_details.order_id  
group by pizza_orders.order_date) as sales;
```

	order_date	cum_revenue
▶	2015-01-01 00:00:00	2713.850000000004
	2015-01-02 00:00:00	5445.75
	2015-01-03 00:00:00	8108.15
	2015-01-04 00:00:00	9863.6
	2015-01-05 00:00:00	11929.55
	2015-01-06 00:00:00	14358.5
	2015-01-07 00:00:00	16560.7
	2015-01-08 00:00:00	19399.05
	2015-01-09 00:00:00	21526.4
	2015-01-10 00:00:00	23000.250000000002

< BACK

NEXT >

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

```
select name,revenue from  
(select category,name,revenue,  
rank() over(partition by category order by revenue desc) as rn  
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 rn <=3;
```

	name	revenue
►	The Barbecue Chicken Pizza	31480.75
	The Thai Chicken Pizza	30945
	The California Chicken Pizza	29975
	The Classic Deluxe Pizza	27315.5
	The Hawaiian Pizza	22970.25
	The Pepperoni Pizza	21960
	The Spicy Italian Pizza	25590.5

< BACK



THANK YOU!

< BACK