

HELLO



MY NAME IS ROHIT KUMAR THIS PROJECT I HAVE
UTILIZED SQL QUERIES TO SOLVE A QUESTION AS
REALTED TO PIZZA SALES





DESCRIPTION

The Pizza Hut Database Management System is designed to handle the operations of a Pizza Hut restaurant, focusing on the core functionalities of order management, inventory control, menu tracking, and employee management. The project utilizes MySQL to store and manipulate data, ensuring that the restaurant can efficiently handle day-to-day activities

< 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
```

< BACK

NEXT >

Result Grid	
	total_sales
▶	817860.05

List the top 5 most ordered pizza types along with their 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 Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

< back

next >

.Group the orders by date and calculate the average number of pizzas ordered per day.



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

Result Grid	
	AVG(quantity)
▶	60.7709

< back

next >

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 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 revenue DESC
LIMIT 3;
```

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

< back

next >

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(order_details.quantity * pizzas.price) 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	2713.850000000004
	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

< back

next >

Determine the distribution of orders by hour of the day.



SELECT

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

FROM

```
orders
```

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



Result Grid		
	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
	22	663
	23	28
	10	8
	9	1

< back

next >

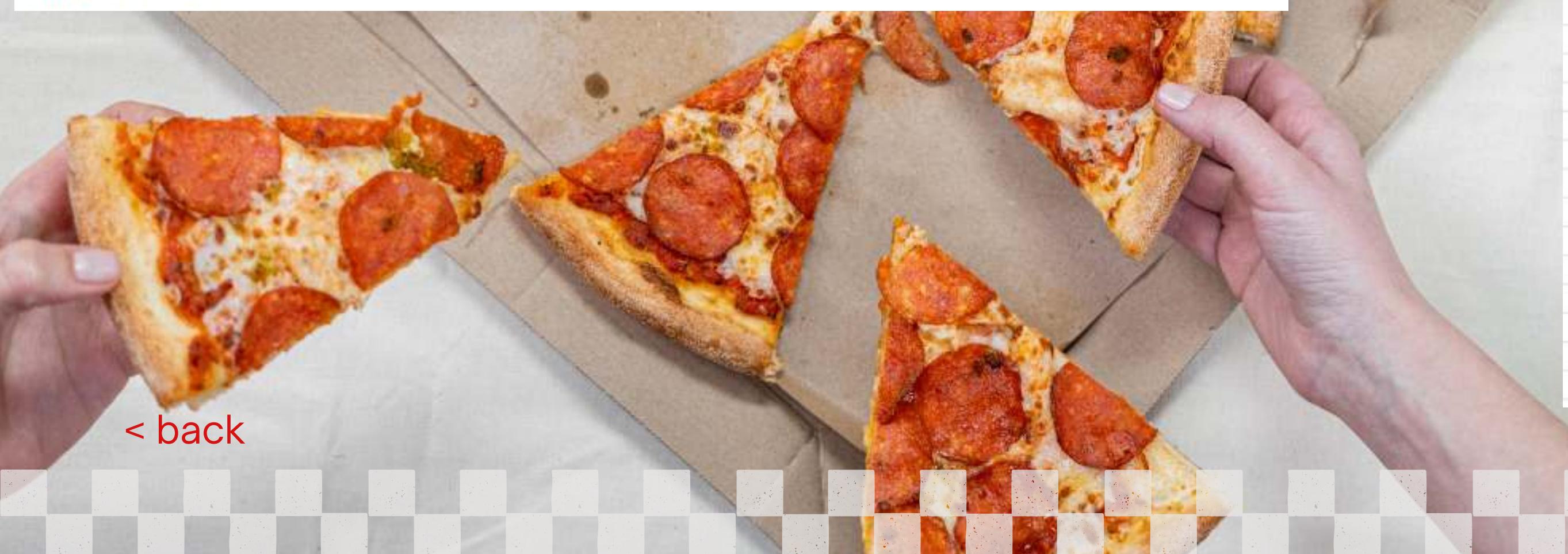
Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```

select category, 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 = pizza_types.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;

```



[< back](#)

	category	name	revenue
▶	Chicken	The Barbecue Chicken Pizza	817860.049999993
	Chicken	The California Chicken Pizza	817860.049999993
	Chicken	The Chicken Alfredo Pizza	817860.049999993
	Chicken	The Chicken Pesto Pizza	817860.049999993
	Chicken	The Southwest Chicken Pizza	817860.049999993
	Chicken	The Thai Chicken Pizza	817860.049999993
	Classic	The Big Meat Pizza	817860.049999993
	Classic	The Classic Deluxe Pizza	817860.049999993
	Classic	The Hawaiian Pizza	817860.049999993
	Classic	The Italian Capocollo Pizza	817860.049999993
	Classic	The Napolitana Pizza	817860.049999993
	Classic	The Pepperoni, Mushroom, ...	817860.049999993
	Classic	The Pepperoni Pizza	817860.049999993
	Classic	The Greek Pizza	817860.049999993
	Supreme	The Brie Carre Pizza	817860.049999993
	Supreme	The Calabrese Pizza	817860.049999993
	Supreme	The Italian Supreme Pizza	817860.049999993
	Supreme	The Pepper Salami Pizza	817860.049999993

[next >](#)



THANK YOU!

< back