

# Pizza Hut sale's Data Analysis

using

MySQL

RISHABH PATEL

# Retrieve the total number of orders placed.

```
select count(order_id) as total_orders  
from orders;
```

Result Grid	
	total_orders
▶	21350



# Calculate the total revenue generated from pizza sales.

```
select  
round(sum(order_details.quantity * pizzas.price), 4) as total_sales  
from order_details join pizzas  
on pizzas.pizza_id = order_details.pizza_id;
```

Result Grid	
	total_sales
▶	817860.05

# 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 1;
```

Result Grid				 Filter Rows
	name	price		
▶	The Greek Pizza	35.95		

# 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 limit 1;
```

Result Grid			Filter
	size	order_count	
▶	L	18526	

# 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;
```

Result Grid			Filter Rows:
	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.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 pizza_types.category order by quantity desc;
```

Result Grid			Filter
	category	quantity	
▶	Classic	14888	
	Supreme	11987	
	Veggie	11649	
	Chicken	11050	

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

```
select hour(order_time) as hour, count(order_id)
from orders as order_count
group by hour(order_time);
```

Result Grid			Filter Rows
	hour	count(order_id)	
▶	11	1231	
	12	2520	
	13	2455	
	14	1472	
	15	1468	
	16	1920	
	17	2336	
	18	2399	



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

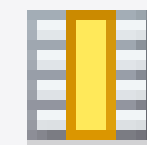
```
select category, count(name)
from pizza_types
group by category;
```

Result Grid			Filter R
	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 round(avg(quantity), 0) as avg_pizza_ordered_per_day 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 orders.order_date) as order_quantity;
```

Result Grid



Filter Rows:

	avg_pizza_ordered_per_day
▶	138

# 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 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;
```

Result Grid



Filter Rows:

	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(order_details.quantity * pizzas.price) /  
(select sum(order_details.quantity * pizzas.price) as total_sales  
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;
```

Result Grid			Filter
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

# Thank you

RISHABH PATEL