

Pizzas Sale Sql project

Hello!!

My name is Vivek Kumar,
In this Project I am utilise some SQL
Queries to solve the question that were
releted to pizzas sales



Dataset use -

their is 4 csv files are used as data set

| pizzahut | |
|----------|---------------|
| ▼ | Tables |
| > | order_details |
| > | orders |
| > | pizzas |
| > | pizza_type |



Queries Question



- 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.



1. Retrieve the total number of orders placed.

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

output

| total_orders |
|--------------|
| 21350 |



2.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 |
| | 817860.05 |



3. Identify the highest-priced pizza.

```
select pizza_type.name,  
pizzas.price  
from pizza_type join pizzas  
on pizza_type.pizza_type_id=pizzas.pizza_type_id  
order by pizzas.price desc limit 1;
```

Result Grid

Filter

| name | price |
|-----------------|-------|
| The Greek Pizza | 35.95 |



4. Identify the most common pizza size ordered



```
-----  
select quantity, count(order_details_id)  
from order_details group by quantity;  
  
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

| | size | order_count |
|--|------|-------------|
| | L | 18526 |

5.List the top 5 most ordered pizza types along with their quantities.



```
select pizza_type.name,  
sum(order_details.quantity) as quantity  
from pizza_type  
join pizzas  
on pizza_type.pizza_type_id=pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id=pizzas.pizza_id  
group by pizza_type.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 |

6.Join the necessary tables to find the total quantity of each pizza category ordered.

```
select pizza_type.category,  
sum(order_details.quantity)as quantity  
from pizza_type join pizzas  
on pizza_type.pizza_type_id=pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id=pizzas.pizza_id  
group by pizza_type.category order by quantity desc;
```



Result Grid

| category | quantity |
|----------|----------|
| Classic | 14888 |
| Supreme | 11987 |
| Veggie | 11649 |
| Chicken | 11050 |

7.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 |

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



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

Result Grid Filter

| category | count(name) |
|----------|-------------|
| Chicken | 6 |
| Classic | 8 |
| Supreme | 9 |
| Veggie | 9 |

9. 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 orders.order_date) as order_quantity;
```

| Result Grid | Filter |
|-------------------------|--------|
| round(avg(quantity)...) | |
| 138 | |



10.Determine the top 3 most ordered pizza types based on revenue.

```
select pizza_type.name,  
sum(order_details.quantity*pizzas.price) as revenue  
from pizza_type join pizzas  
on pizzas.pizza_type_id=pizza_type.pizza_type_id  
join order_details  
on order_details.pizza_id=pizzas.pizza_id  
group by pizza_type.name order by revenue desc limit 3;
```

Result Grid

Filter Rows: S

| name | revenue |
|------------------------------|----------|
| The Thai Chicken Pizza | 43434.25 |
| The Barbecue Chicken Pizza | 42768 |
| The California Chicken Pizza | 41409.5 |



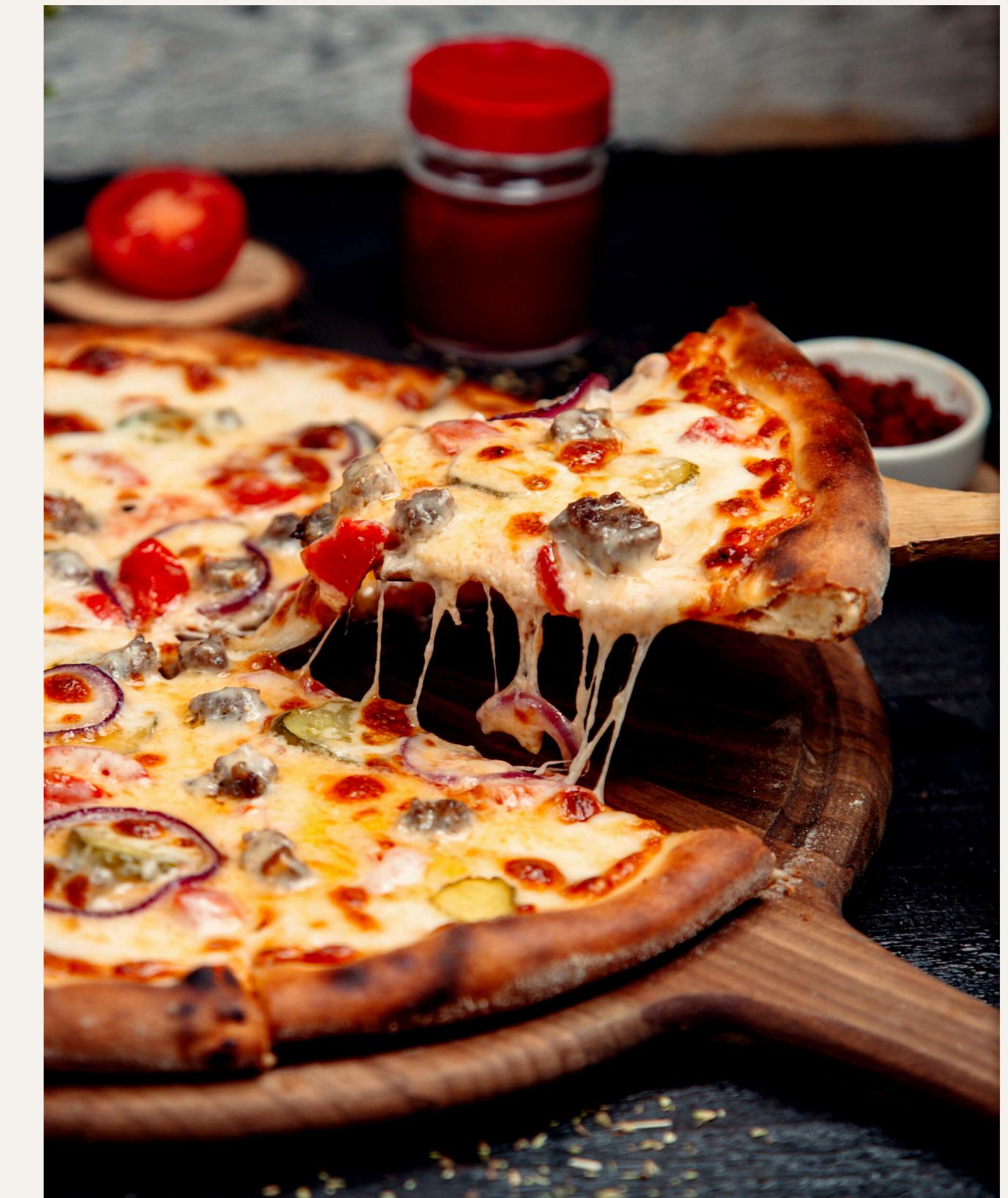
11. Calculate the percentage contribution of each pizza type to total revenue.

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

Result Grid

Filter

| category | revenue |
|----------|---------|
| Classic | 26.91 |
| Supreme | 25.46 |
| Chicken | 23.96 |
| Veggie | 23.68 |



12.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 Re

| order_date | cum_revenue |
|------------|--------------------|
| 2015-01-01 | 2713.8500000000003 |
| 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 | 23990.350000000003 |
| 2015-01-11 | 25862.65 |

13.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 rn
from(
select pizza_type.category,pizza_type.name,
sum((order_details.quantity)*pizzas.price) as revenue
from pizza_type join pizzas
on pizza_type
.pizza_type_id=pizzas.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_type.category,pizza_type.name) as a) as b
where rn<=3;
```

| Result Grid | | Filter Rows: | Search |
|-------------|------------------------------|--------------------|--------|
| | 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 | |
| | The Pepperoni Pizza | 30161.75 | |
| | The Spicy Italian Pizza | 34831.25 | |
| | The Italian Supreme Pizza | 33476.75 | |
| | The Sicilian Pizza | 30940.5 | |
| | The Four Cheese Pizza | 32265.700000000004 | |
| | The Mexicana Pizza | 26780.75 | |
| | The Five Cheese Pizza | 26066.5 | |



Thank you!