

Slice Analytics

Portfolio project based on SQL



Description

The Slice Analytics project leverages SQL to analyze and visualizes sales data. Identifying the key trends and performances metrices. This project aims to optimize inventory management, enhance marketing strategies and improve overall sales efficiency for a pizza business.



Query 1- Retrieve the total number of orders placed.

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

Output

Result Grid	
	total_orders
▶	21350

Query 2- Calculate total revenue generate from pizza sales.

```
SELECT
    ROUND(SUM(order_details.qunatity * pizzas.price),
          2) AS total_revenue
FROM
    order_details
    JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```



Output

Result Grid	
	total_revenue
▶	817860.05

Query 3- 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;
```



Output



Result Grid			Filter Rows:
	name	price	
▶	The Greek Pizza	35.95	

Query 4- 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;
```



Output

Result Grid					F
	size	order_count			
▶	L	18526			

Query 5- 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;
```



Output

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	



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



Output

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

```
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;
```





Query 7- Determine the distribution of orders by hour of the day.



Output

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	

```
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```




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



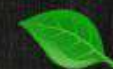
Output

Result Grid			Filter Rows:
	category	count(name)	
▶	Chicken	6	
	Classic	8	
	Supreme	9	
	Veggie	9	

```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```




Query 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



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



Output

Result Grid			Filter Rows:
	name	revenue	
▶	The Thai Chicken Pizza	43434.25	
	The Barbecue Chicken Pizza	42768	
	The California Chicken Pizza	41409.5	

```
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;
```


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

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



Output

category	revenue
Classic	26.90596025566967
Supreme	25.45631126009862
Chicken	23.955137556847287
Veggie	23.682590927384577



Query 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;
```



Output

order_date	cum_revenue
2015-01-01	2713.8500000000004
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	23990.350000000002
2015-01-11	25862.65



Query 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_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
```

Output

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

Conclusion

Slice Analytics project demonstrates the power of data analytics in driving informed decision-making and achieving business growth in the competitive pizza industry.

The slide features a dark, textured background. A large, rounded rectangular frame in a light yellowish-gold color is centered, containing the text "Thank You". The frame is surrounded by various food-related illustrations: a slice of pizza in the top left, a tomato slice in the top center, two red chili peppers and a green leaf in the top right, a whole pizza in the bottom right, and a green leaf in the bottom left. A bright yellow, brush-stroke-like shape is in the bottom left corner.

Thank You