

Pizza Sales Report

Using SQL Queries



Introduction

This pizza sales analysis report leverages SQL queries to uncover key insights. It examines sales trends, customer preferences, and performance metrics, providing a data-driven foundation for strategic decisions to boost revenue and enhance customer satisfaction.

We have four datasets to analyse the sales of the company in various ways to explain the trends of pizzas sold and to utilise it in expanding the business sales and profit.

Sample of Datasets used

1. **SELECT * FROM pizzago.pizzas;**

	pizza_id	pizza_type_id	size	price
▶	bbq_ckn_s	bbq_ckn	S	12.75
	bbq_ckn_m	bbq_ckn	M	16.75
	bbq_ckn_l	bbq_ckn	L	20.75
	cali_ckn_s	cali_ckn	S	12.75
	cali_ckn_m	cali_ckn	M	16.75
	cali_ckn_l	cali_ckn	L	20.75
	dkn_alfredo_s	dkn_alfredo	S	12.75
	dkn_alfredo_m	dkn_alfredo	M	16.75
	dkn_alfredo_l	dkn_alfredo	L	20.75
	dkn_pesto_s	dkn_pesto	S	12.75
	dkn_pesto_m	dkn_pesto	M	16.75
	dkn_pesto_l	dkn_pesto	L	20.75
	southw_ckn_s	southw_ckn	S	12.75

Sample of Datasets used

```
1 • SELECT * FROM pizzago.order_details;
```

	order_details_id	order_id	pizza_id	quantity
▶	1	1	hawaiian_m	1
	2	2	classic_dlx_m	1
	3	2	five_cheese_l	1
	4	2	ital_supr_l	1
	5	2	mexicana_m	1
	6	2	thai_ckn_l	1
	7	3	ital_supr_m	1
	8	3	prsc_argla_l	1
	9	4	ital_supr_m	1
	10	5	ital_supr_m	1
	11	6	bbq_ckn_s	1
	12	6	the_greek_s	1
	13	7	spinach_supr_s	1
	14	8	spinach_supr_s	1
	15	9	classic_dlx_s	1
	16	9	green_garde...	1

Sample of Datasets used

```
1 • SELECT * FROM pizzago.pizza_types;
```

	pizza_type_id	name	category	ingredients
▶	bbq_dkn	The Barbecue Chicken Pizza	Chicken	Barbecued Chicken, Red Peppers, Green Pepp...
	cali_dkn	The California Chicken Pizza	Chicken	Chicken, Artichoke, Spinach, Garlic, Jalapeno P...
	dkn_alfredo	The Chicken Alfredo Pizza	Chicken	Chicken, Red Onions, Red Peppers, Mushrooms...
	dkn_pesto	The Chicken Pesto Pizza	Chicken	Chicken, Tomatoes, Red Peppers, Spinach, Garl...
	southw_dkn	The Southwest Chicken Pizza	Chicken	Chicken, Tomatoes, Red Peppers, Red Onions, ...
	thai_dkn	The Thai Chicken Pizza	Chicken	Chicken, Pineapple, Tomatoes, Red Peppers, T...
	big_meat	The Big Meat Pizza	Classic	Bacon, Pepperoni, Italian Sausage, Chorizo Sau...
	classic_dlx	The Classic Deluxe Pizza	Classic	Pepperoni, Mushrooms, Red Onions, Red Peppe...
	hawaiian	The Hawaiian Pizza	Classic	Sliced Ham, Pineapple, Mozzarella Cheese
	ital_cpdlo	The Italian Capocollo Pizza	Classic	Capocollo, Red Peppers, Tomatoes, Goat Chee...
	napolitana	The Napolitana Pizza	Classic	Tomatoes, Anchovies, Green Olives, Red Onion...
	pep_msh_pep	The Pepperoni, Mushroom, ...	Classic	Pepperoni, Mushrooms, Green Peppers
	pepperoni	The Pepperoni Pizza	Classic	Mozzarella Cheese, Pepperoni
	the_greek	The Greek Pizza	Classic	Kalamata Olives, Feta Cheese, Tomatoes, Garli...
	brie_carre	The Brie Carre Pizza	Supreme	Brie Carre Cheese, Prosciutto, Caramelized Oni...
	calabrese	The Calabrese Pizza	Supreme	'Nduja Salami, Pancetta, Tomatoes, Red Onions...



Sample of Datasets used

```
1 • SELECT * FROM pizzago.orders;
```

	order_id	order_date	order_time
▶	1	2015-01-01	11:38:36
	2	2015-01-01	11:57:40
	3	2015-01-01	12:12:28
	4	2015-01-01	12:16:31
	5	2015-01-01	12:21:30
	6	2015-01-01	12:29:36
	7	2015-01-01	12:50:37
	8	2015-01-01	12:51:37
	9	2015-01-01	12:52:01
	10	2015-01-01	13:00:15
	11	2015-01-01	13:02:59
	12	2015-01-01	13:04:41
	13	2015-01-01	13:11:55
	14	2015-01-01	13:14:19
	15	2015-01-01	13:33:00

SQL Query

1. Retrieve the total number of orders placed.

```
1 -- Retrieve the total number of orders placed  
2 • SELECT COUNT(order_id) as Total_Orders FROM orders;
```

Total_Orders

21350



SQL Query

2. Calculate the total revenue generated from pizza sales.

```
1  -- Calculate the total revenue generated from pizza sales.  
2 •  SELECT  
3      round(sum(order_details.quantity*pizzas.price),1) as total_sales  
4  from order_details join pizzas  
5  where pizzas.pizza_id=order_details.pizza_id ;
```

total_sales
817860



SQL Query

3. Identify the highest-priced pizza.

```
1      -- Identify the highest-priced pizza.  
2 •  select pizza_types.name,pizzas.price  
3      from pizza_types join pizzas  
4      where pizza_types.pizza_type_id=pizzas.pizza_type_id  
5      order by pizzas.price desc limit 1;
```

	name	price
▶	The Greek Pizza	35.95



SQL Query

4. Identify the most common pizza size ordered.

```
1  -- Identify the most common pizza size ordered.  
2 • select pizzas.size, count(order_details.order_details_id) as count_of_order  
3   from pizzas join order_details  
4     on pizzas.pizza_id=order_details.pizza_id  
5   group by pizzas.size  
6   order by count_of_order desc limit 1;
```

	size	count_of_order
▶	L	18526

SQL Query

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

```
1 -- List the top 5 most ordered pizza types along with their quantities
2 • select pizza_types.name,sum(order_details.quantity) as total_order
3   from pizza_types join pizzas
4     on pizza_types.pizza_type_id=pizzas.pizza_type_id
5   join order_details
6     where order_details.pizza_id= pizzas.pizza_id
7   group by pizza_types.name
8   order by total_order desc limit 5;
```

	name	total_order
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

SQL Query

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

```
1 -- Join the necessary tables to find the total quantity of each pizza category ordered.  
2 • select pizza_types.category ,sum(order_details.quantity) as quantity  
3   from pizza_types join pizzas  
4   on pizza_types.pizza_type_id=pizzas.pizza_type_id  
5   join order_details  
6   on order_details.pizza_id=pizzas.pizza_id  
7   group by pizza_types.category  
8   order by quantity desc;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

SQL Query

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

```
1 -- Determine the distribution of orders by hour of the day.  
2 • select hour(order_time) as hour ,count(order_id) as count_order from orders  
3 group by hour(order_time);
```

	hour	count_order
▶	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

SQL Query

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

```
1 -- Join relevant tables to find the category-wise distribution of pizzas.  
2 • select category, count(name) from pizza_types  
3 group by category;
```

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



SQL Query

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

```
1  -- Group the orders by date and calculate the average number of pizzas ordered per day.  
2 • select round(avg(quantity),0) as avg_pizza_ordered_perday  
3   from  
4   (select orders.order_date, sum(order_details.quantity) as quantity  
5    from orders join order_details  
6    on order_details.order_id=orders.order_id  
7    group by orders.order_date)as order_quantity;
```

avg_pizza_ordered_perday
138

SQL Query

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

```
1  -- Determine the top 3 most ordered pizza types based on revenue.  
2 • select pizza_types.name,  
3   sum(order_details.quantity*pizzas.price) as revenue  
4   from pizza_types join pizzas  
5   on pizzas.pizza_type_id=pizza_types.pizza_type_id  
6   join order_details  
7   on order_details.pizza_id =pizzas.pizza_id  
8   group by pizza_types.name  
9   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

SQL Query

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

```
1  -- Calculate the percentage contribution of each pizza category to total revenue.  
2  
3 • select pizza_types.category,  
4   round(sum(order_details.quantity*pizzas.price)/(select sum(order_details.quantity*pizzas.price)  
5   from order_details join pizzas  
6   on pizzas.pizza_id=order_details.pizza_id)*100,2) as revenue  
7   from pizza_types join pizzas  
8   on pizzas.pizza_type_id=pizza_types.pizza_type_id  
9   join order_details  
10  on order_details.pizza_id =pizzas.pizza_id  
11  group by pizza_types.category  
12  order by revenue desc ;
```

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68



Thank you!



Ankan Kumar Paul



2024