

Home

About

Service

Contact

≡
Canva

Hot **PIZZA**

Presentation



www.reallygreatsite.com



PIZZA SALES CASE STUDY



In this project I have utilized SQL query to solve questions that were related to Pizza sales

1. RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
select count(quantity) as Total_orders  
from orders_details;
```

Result Grid	
	Total_orders
▶	23250

2. CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT round(sum(o.quantity * p.price),0) as total_revenue  
from orders_details as o join pizzas as p  
on o.pizza_id = p.pizza_id;
```

Result Grid	
	total_revenue
▶	391558

3. IDENTIFY THE HIGHEST-PRICED PIZZA.

```
select pt.name, p.price  
from pizza_types as pt join pizzas as p  
on pt.pizza_type_id = p.pizza_type_id  
order by p.price desc limit 1;
```

Result Grid

Filter Rows:

	name	price
▶	The Greek Pizza	35.95

4. IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select p.size, count(od.order_details_id) as order_count  
from pizzas as p join orders_details as od  
on p.pizza_id = od.pizza_id  
group by p.size  
order by order_count desc;
```

Result Grid | Filter

	size	order_count
▶	L	8896
	M	7344
	S	6715
	XL	279
	XXL	16

5. LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
select pt.name, count(od.quantity) as quantities  
from pizza_types as pt join pizzas as p  
on pt.pizza_type_id = p.pizza_type_id  
join orders_details as od  
on p.pizza_id = od.pizza_id  
group by pt.name  
order by quantities desc limit 5;
```

Result Grid | Filter Rows:

	name	quantities
▶	The Barbecue Chicken Pizza	1172
	The Pepperoni Pizza	1123
	The Hawaiian Pizza	1121
	The Classic Deluxe Pizza	1114
	The California Chicken Pizza	1094

6. JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```
select pt.category, sum(od.quantity) as quantity  
from pizza_types as pt join pizzas as p  
on pt.pizza_type_id = p.pizza_type_id  
join orders_details as od  
on p.pizza_id = od.pizza_id  
group by pt.category;
```

Result Grid

	category	quantity
▶	Classic	7040
	Veggie	5651
	Supreme	5764
	Chicken	5246

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;
```

Result Grid | Filter R

	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009
	20	1642

Result 22 ×

8. JOIN RELEVANT TABLES TO FIND THE

CATEGORY-WISE DISTRIBUTION OF PIZZAS.



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

Result Grid | Filter Row

	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(sum_quantity),0) from  
  (select o.order_date, sum(od.quantity) as sum_quantity  
   from orders as o join orders_details as od  
   on o.order_id = od.order_id  
   group by o.order_date) as order_quantity;
```

	round(avg(sum_quantity),0)
▶	138

10. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
select pt.name, sum(p.price * od.quantity) as revenue  
from pizza_types as pt join pizzas as p  
on pt.pizza_type_id = p.pizza_type_id  
join orders_details as od  
on od.pizza_id = p.pizza_id  
group by pt.name  
order by revenue desc limit 3;
```

Result Grid | Filter Rows:

	name	revenue
▶	The Barbecue Chicken Pizza	21193.5
	The Thai Chicken Pizza	19998
	The California Chicken Pizza	19702.25

11. CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
select pt.category, round(sum(p.price * od.quantity) /  
    (select sum(p.price * od.quantity) total_sales  
        from orders_details as od join pizzas as p on od.pizza_id = p.pizza_id)*100 ,2) as revenue  
from pizza_types as pt join pizzas as p  
on pt.pizza_type_id = p.pizza_type_id  
join orders_details as od  
on od.pizza_id = p.pizza_id  
group by pt.category  
order by revenue desc;
```

Result Grid | Filter

	category	revenue
▶	Classic	26.63
	Supreme	25.61
	Veggie	23.99
	Chicken	23.77

12. ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select order_date, sum_revenue,  
sum(sum_revenue) over(order by order_date) as cum_revenue  
from  
(select o.order_date, round(sum(p.price * od.quantity),0) as sum_revenue  
from pizzas as p join orders_details as od  
on od.pizza_id = p.pizza_id  
join orders as o  
on o.order_id = od.order_id  
group by o.order_date) as sales;
```

	order_date	sum_revenue	cum_revenue
▶	2015-01-01	2714	2714
	2015-01-02	2732	5446
	2015-01-03	2662	8108
	2015-01-04	1755	9863
	2015-01-05	2066	11929
	2015-01-06	2429	14358
	2015-01-07	2222	16580

13. 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 pt.category, pt.name, sum(p.price * od.quantity) as revenue
     from pizza_types as pt join pizzas as p
     on pt.pizza_type_id = p.pizza_type_id
     join orders_details as od
     on od.pizza_id = p.pizza_id
     group by pt.category, pt.name) as a) as b
where rn <=3;
```

	category	name	revenue
▶	Chicken	The Barbecue Chicken Pizza	21193.5
	Chicken	The Thai Chicken Pizza	19998
	Chicken	The California Chicken Pizza	19702.25
	Classic	The Classic Deluxe Pizza	17516.5
	Classic	The Hawaiian Pizza	15241
	Classic	The Pepperoni Pizza	14267
	Supreme	The Spicy Italian Pizza	16697.75
	Supreme	The Italian Supreme Pizza	16404
	Supreme	The Sicilian Pizza	14805
	Veggie	The Four Cheese Pizza	15836.05000000016
	Veggie	The Five Cheese Pizza	12894.5
	Veggie	The Mexicana Pizza	12523.25

THANK YOU

