

PIZZA SALES ANALYSIS USING SQL



```
1 -- Retrieve the total number of orders placed  
2 • select count(order_id) as total_orders from orders;  
3
```

	total_orders
▶	21350

```
1 -- Calculate the total revenue generated from pizza sales  
2 • select  
3 round(sum(order_details.quantity * pizzas.price),2)  
4 as total_revenue  
5 from order_details join pizzas  
6 on pizzas.pizza_id = order_details.pizza_id  
7
```

	total_revenue
▶	817860.05

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

	name	price
▶	The Greek Pizza	35.95





```
1 -- identify the most common pizza size ordered
2 • SELECT
3     pizzas.size,
4     COUNT(order_details.order_details_id) AS order_count
5 FROM
6     pizzas
7     JOIN
8         order_details ON pizzas.pizza_id = order_details.pizza_id
9 GROUP BY pizzas.size
10 ORDER BY order_count DESC
11 LIMIT 1;
```

	size	order_count
▶	L	18526

```
-- list the 5 most ordered pizza types along with their quantities
select pizza_types.name, sum(order_details.quantity) as quant
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 quant desc limit 5;
```

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

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

	category	quant
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

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

	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920

```
1  -- Join relevant tables to find the category-wise
2  -- distribution of pizzas.
3  • select category, count(name) from pizza_types
4  group by category
5
```

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

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

	round(avg(quant))
▶	138

```
1  -- Determine the top 3 most ordered pizza types based on revenue.
2
3 • select pizza_types.name,
4   sum(order_details.quantity*pizzas.price) as revenue
5   from pizza_types
6   join pizzas
7   on pizzas.pizza_type_id = pizza_types.pizza_type_id
8   join order_details
9   on order_details.pizza_id = pizzas.pizza_id
10  group by pizza_types.name order by revenue desc limit 3;
11
```

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

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

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

```
1  -- Analyze the cumulative revenue generated over time.  
2  
3 • select orders.date,  
4     sum(revenue) over(order by order_date) as cum_revenue  
5     from  
6     (select orders.date,  
7         sum(order_details.quantity * pizzas.price) as revenue  
8         from order_details  
9         join pizzas  
10        on order_details.pizza_id = pizzas.pizza_id  
11        join orders  
12        on orders.order_id = order_details.order_id  
13     group by orders.date) as sales;
```





```
1   -- Determine the top 3 most ordered pizza types based
2   -- on revenue for each pizza category.
3 • select name, revenue from
4   (select category, name, revenue,
5    rank() over(partition by category order by revenue desc) as rn
6    from
7   (select pizza_types.category, pizza_types.name,
8    sum((order_details.quantity)*pizzas.price) as revenue
9    from pizza_types
10   join pizzas
11   on pizza_types.pizza_type_id = pizzas.pizza_type_id
12   join order_details
13   on order_details.pizza_id = pizzas.pizza_id
14   group by pizza_types.category, pizza_types.name) as a) as b
15   where rn <=3;
16
```

	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	32265.70000000065
	The Mexicana Pizza	26780.75

THANK YOU!



