

A cartoon character with blue hair styled in two puffs, wearing a white chef's hat and a red and white striped apron, is holding a large pizza with various toppings like pepperoni and basil leaves. The character is standing on a red and white checkered floor.

Pizza Sales Analysis

using SQL

Retrieve the total number of orders placed

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

total_orders	bigint
21350	🔒

Calculate the total revenue generated from pizza sales.

```
select
    round(
        cast(
            sum(order_details.quantity * pizzas.prize) as numeric
        ),
        2
    ) as total_revenue
from
    order_details
left join pizzas on order_details.pizz_id = pizzas.pizza_id;
```

total_revenue
numeric

817860.05

Identify the highest-priced pizza

```
select
    pizza_types.name,
    pizzas.prize
from
    pizza_types
    left join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id
order by
    pizzas.prize desc
limit
    1;
```

name character varying (100)	prize double precision
The Greek Pizza	35.95

Identify the most common pizza size ordered.

```
select
    pizzas.size,
    count(order_details.order_id)
from
    order_details
    left join pizzas on order_details.pizz_id = pizzas.pizza_id
group by
    pizzas.size
order by
    count desc
limit
    1;
```

size	count
character varying (20)	bigint
L	18526

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

```
with my_table as (
  select
    t1.name,
    t2.pizza_id
  from
    pizza_types as t1
    left join pizzas as t2 on t1.pizza_type_id = t2.pizza_type_id
)
select
  my_table.name,
  count(order_details.quantity) as quantity
from
  my_table
  left join order_details on my_table.pizza_id = order_details.pizza_id
group by
  my_table.name
order by
  quantity desc
limit 5;
```

name	quantity
character varying (100)	bigint
The Classic Deluxe Pizza	2416
The Barbecue Chicken Pizza	2372
The Hawaiian Pizza	2370
The Pepperoni Pizza	2369
The Thai Chicken Pizza	2315

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

```
with my_table as (
  select
    t1.category,
    t2.pizza_id
  from
    pizza_types as t1
    left join pizzas as t2 on t1.pizza_type_id = t2.pizza_type_id
)
select
  my_table.category,
  sum(order_details.quantity) as total_quantity
from
  my_table
  left join order_details on my_table.pizza_id = order_details.pizza_id
group by
  my_table.category
order by
  total_quantity desc;
```

category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day

```
select  
    extract(hour from time) as hours,  
    count(order_id)  
from  
    orders  
group by  
    hours  
order by  
    hours;
```

hours numeric	count bigint
9	1
10	8
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

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

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

category	count
character varying (30)	bigint
Supreme	9
Classic	8
Veggie	9
Chicken	6

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

```
with
    my_pizza as (
        select
            date,
            sum(quantity)
        from
            order_details as t1
        left join orders as t2 on t1.order_id = t2.order_id
    group by
        date
    order by
        date
)
select
    avg(sum)
from
    my_pizza
```

avg	numeric
138.4748603351955307	🔒

Determine the top 3 most ordered pizza types based on revenue

```
with
my_table as (
  select
    t2.pizza_type_id,
    (t2.prize * t1.quantity) as revenue
  from
    order_details as t1
    left join pizzas as t2 on t1.pizz_id = t2.pizza_id
)
select
  pizza_types.name,
  sum(my_table.revenue) as total_revenue
from
  my_table
  left join pizza_types on my_table.pizza_type_id = pizza_types.pizza_type_id
group by
  pizza_types.name
order by
  total_revenue desc
limit
  3;
```

name	total_revenue
character varying (100)	double precision
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

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

```
with
my_table as (
  select
    t2.pizza_type_id,
    (t1.quantity * t2.prize) as total_revenue
  from
    order_details as t1
    left join pizzas as t2 on t1.pizz_id = t2.pizza_id
)
select
category,
(
  sum(t1.total_revenue) / (
    select
      round(
        cast(
          sum(order_details.quantity * pizzas.prize) as numeric
        ),
        2
      ) as total_sales
    from
      order_details
      left join pizzas on order_details.pizz_id = pizzas.pizza_id
  )
) * 100 as percent_revenue
from
my_table as t1
left join pizza_types as t2 on t1.pizza_type_id = t2.pizza_type_id
group by
category;
```

category	percent_revenue
Supreme	25.45631126009862
Classic	26.90596025566967
Veggie	23.682590927384577
Chicken	23.955137556847287

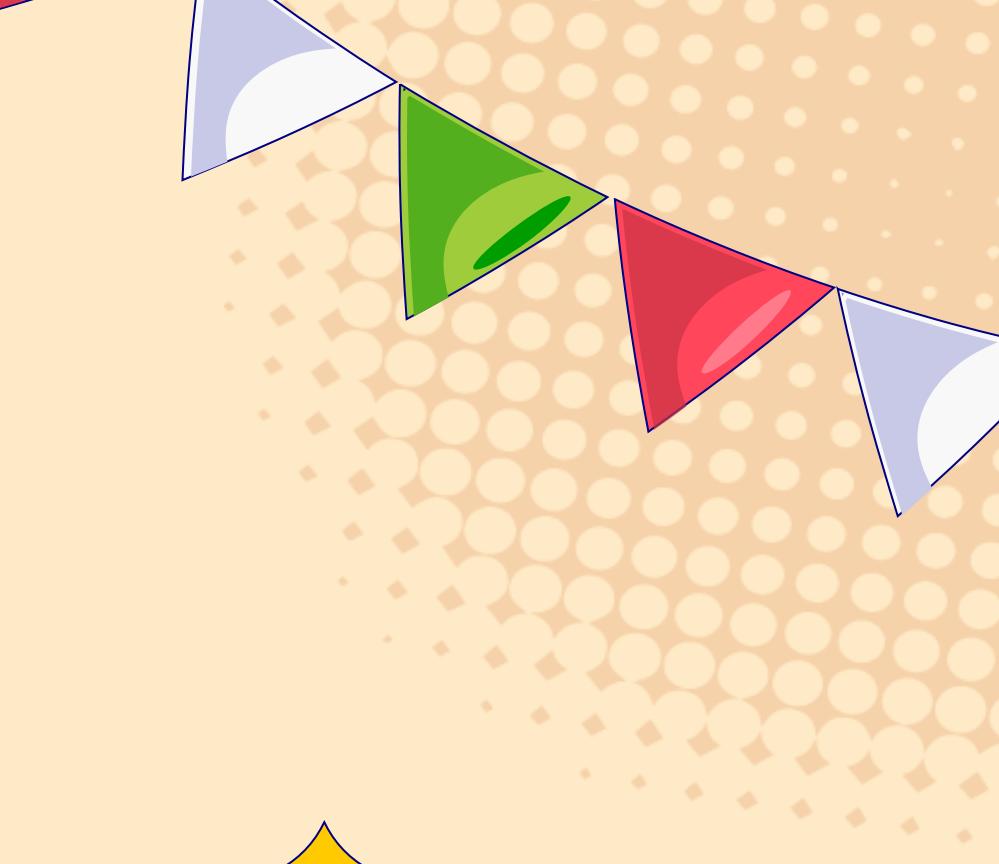
Analyze the cumulative revenue generated over time

```
with
pizza_table as (
  with
    my_table as (
      select
        order_details.order_id,
        (pizzas.prize * order_details.quantity) as revenue
      from
        pizzas
        left join order_details on pizzas.pizza_id = order_details.pizz_id
    )
  select
    orders.date,
    round(cast(sum(my_table.revenue) as numeric), 2) as total_revenue
  from
    my_table
    left join orders on my_table.order_id = orders.order_id
  group by
    date
  order by
    date
)
select
  date,
  sum(total_revenue) over (
    order by
    date
)
from
  pizza_table
```

date	sum
date	numeric
2015-01-01	2713.85
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.60
2015-01-05	11929.55
2015-01-06	14358.50
2015-01-07	16560.70
2015-01-08	19399.05
2015-01-09	21526.40
2015-01-10	23990.35
2015-01-11	25862.65
2015-01-12	27781.70
2015-01-13	29831.30
2015-01-14	32358.70
2015-01-15	34343.50
2015-01-16	36937.65
2015-01-17	39001.75
2015-01-18	40978.60
2015-01-19	43365.75
2015-01-20	45763.65
2015-01-21	47804.20
2015-01-22	50300.90

Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
with
main_table as (
  with
  pizza_table as (
    with
    my_table as (
      select
        pizza_types.name,
        pizzas.prize,
        pizzas.pizza_id,
        pizza_types.category
      from
        pizza_types
      left join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id
    )
    select
      my_table.name,
      round(
        cast(
          sum(my_table.prize * order_details.quantity) as numeric
        ),
        2
      ) as total_revenue,
      my_table.category
    from
      my_table
    left join order_details on my_table.pizza_id = order_details.pizz_id
  )
  group by
    category,
    name
  )
  select
    name,
    category,
    total_revenue,
    rank() over (
      partition by
        category
      order by
        total_revenue desc
    ) as "rank"
  from
    pizza_table
)
select
  name,
  category,
  total_revenue
from
  main_table
where
  rank <= 3;
```



name	category	total_revenue
character varying (100)	character varying (30)	numeric
The Thai Chicken Pizza	Chicken	43434.25
The Barbecue Chicken Pizza	Chicken	42768.00
The California Chicken Pizza	Chicken	41409.50
The Classic Deluxe Pizza	Classic	38180.50
The Hawaiian Pizza	Classic	32273.25
The Pepperoni Pizza	Classic	30161.75
The Spicy Italian Pizza	Supreme	34831.25
The Italian Supreme Pizza	Supreme	33476.75
The Sicilian Pizza	Supreme	30940.50
The Four Cheese Pizza	Veggie	32265.70
The Mexicana Pizza	Veggie	26780.75
The Five Cheese Pizza	Veggie	26066.50



Thank
You!