



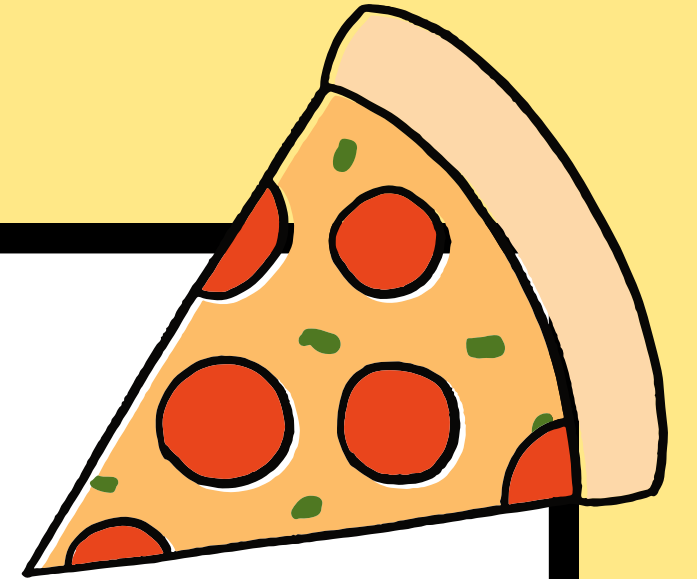
Chirag Gaba

Sales Report

SQL Project



Summary



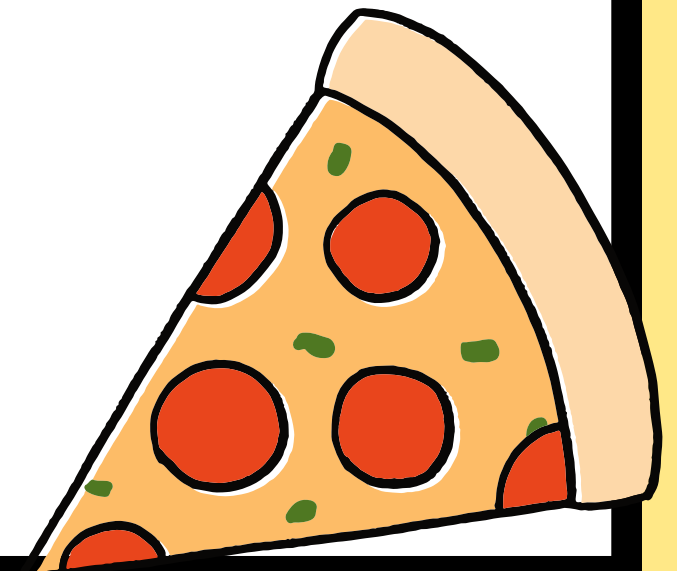
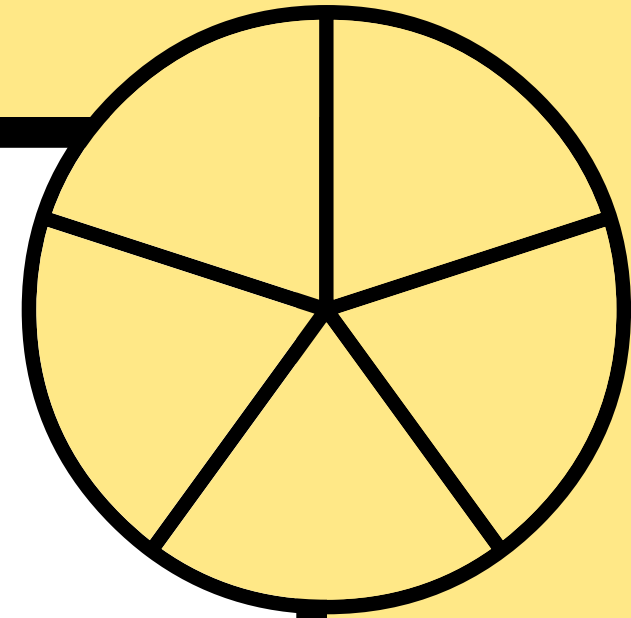
This Data Analysis project focuses on analyzing pizza sales data to derive valuable business insights, by leveraging SQL Queries, the project aims to answer key questions, related to sales performance, customer preferences and order patterns.

Question #1

Retrieve the total number of orders placed.

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

	count(order_id)
▶	21350

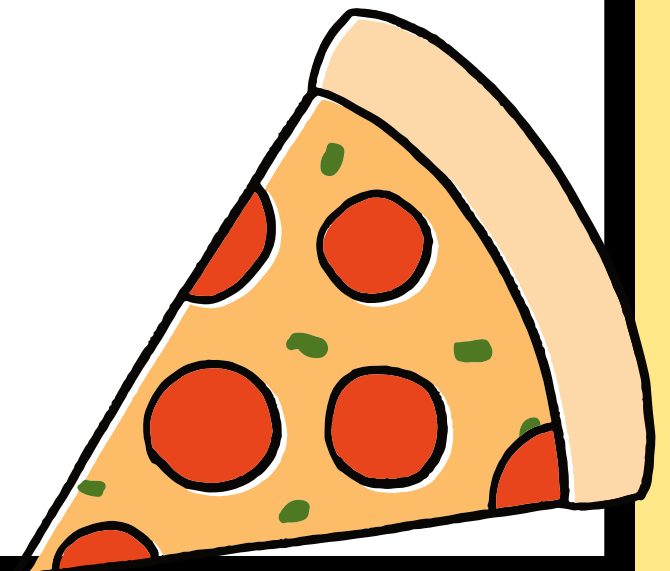
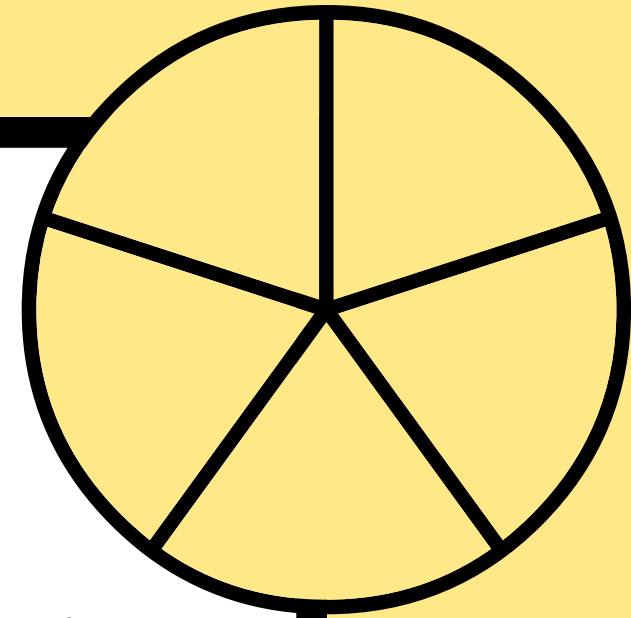


Question #2

Calculate the total revenue generated from pizza sales.

```
SELECT
    round(SUM(pizzas.price * order_details.quantity),2) AS total_revenue
FROM
    pizzas
    JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id;
```

	total_revenue
▶	817860.05

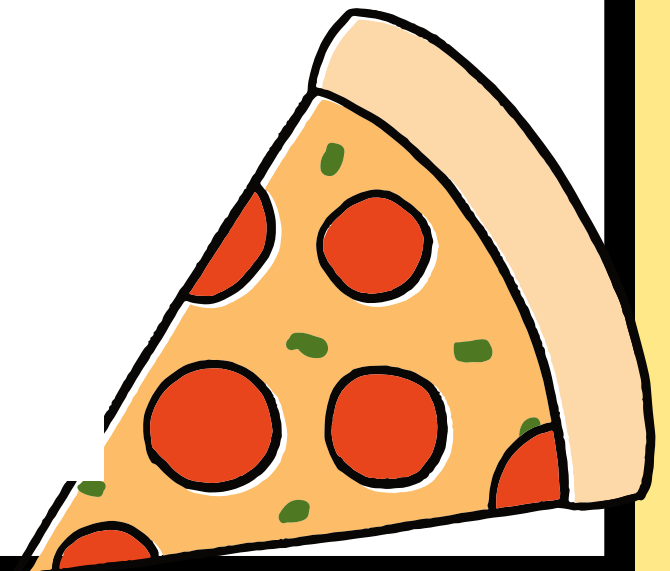
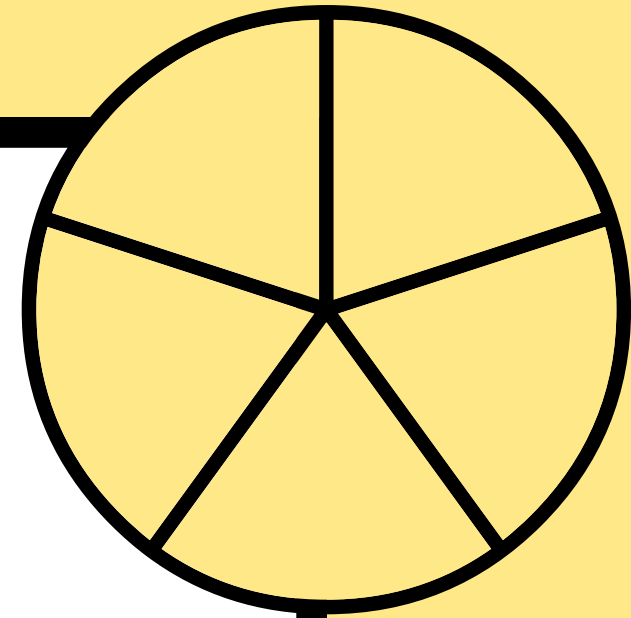


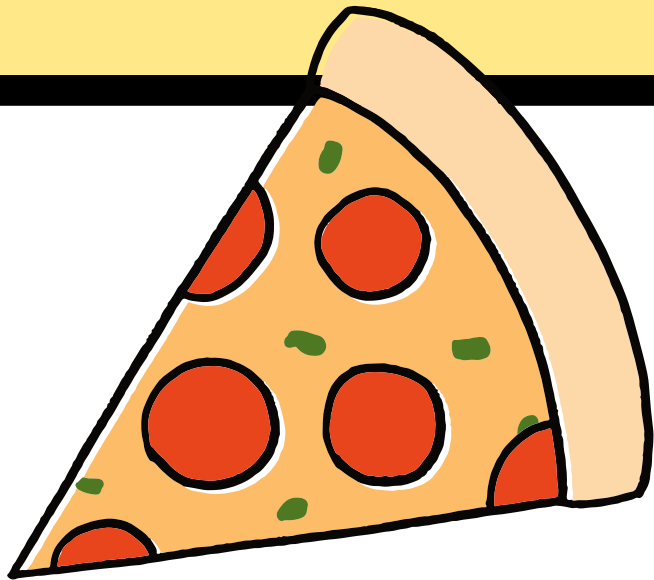
Question #3

Identify the highest-priced pizza.

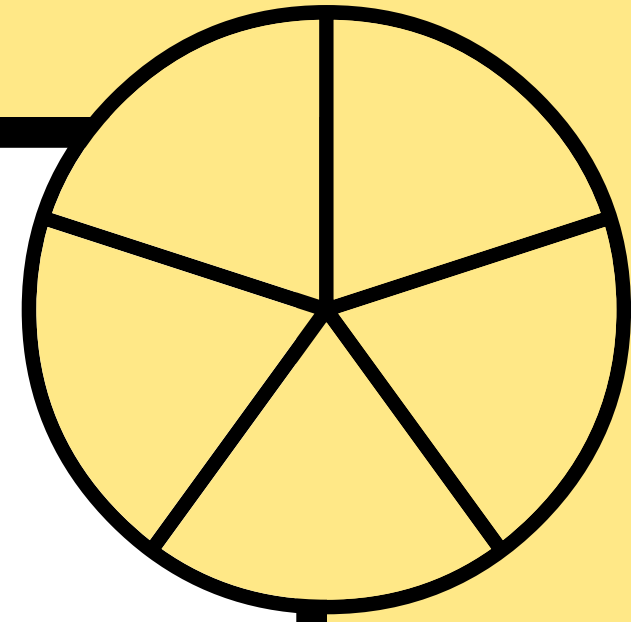
```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

	name	price
▶	The Greek Pizza	35.95





Question #4



Identify the most common pizza size ordered.

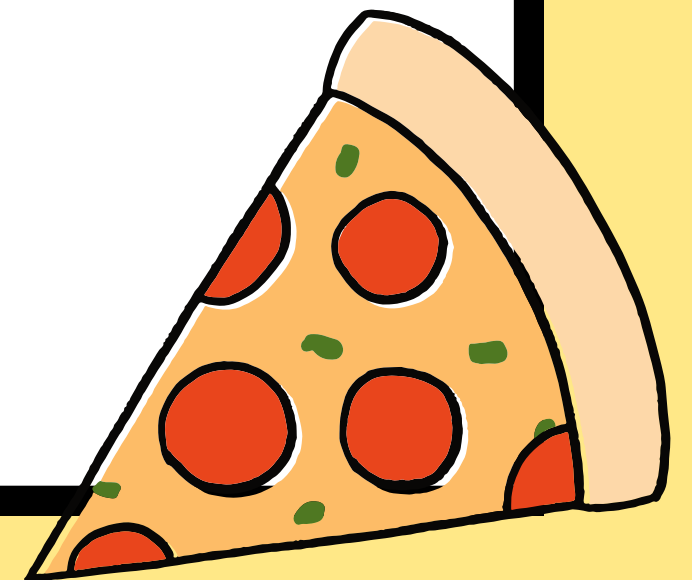
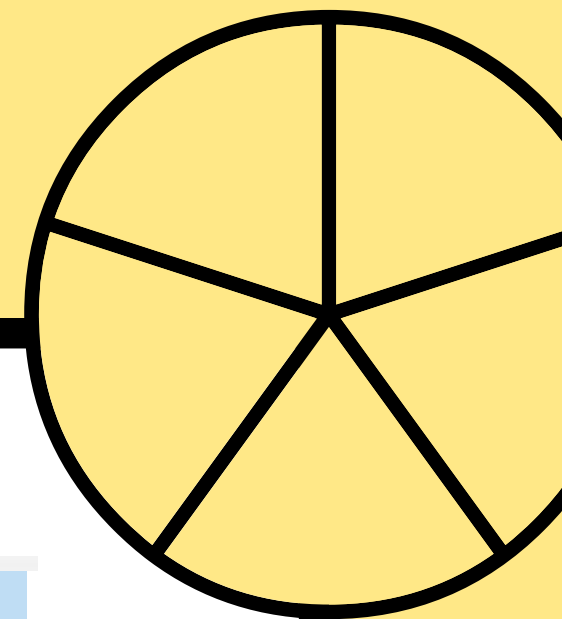
```
SELECT
  pizzas.size, COUNT(order_details.order_details_id) AS count
FROM
  pizzas
  JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY count DESC
LIMIT 1;
```

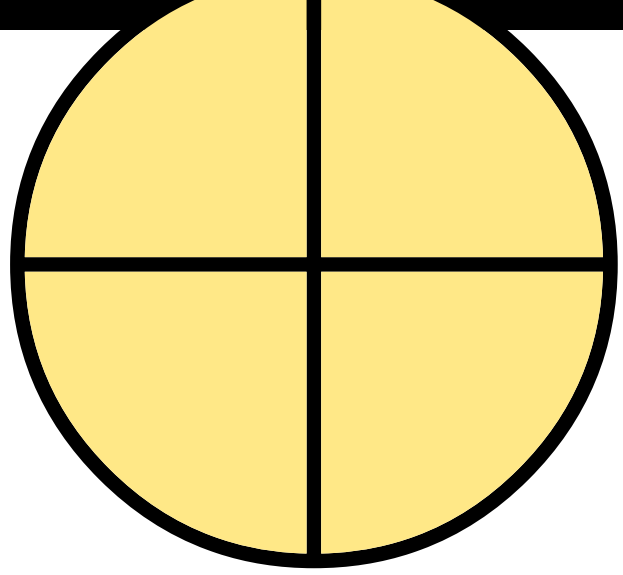
size	count
L	18526

Question #5

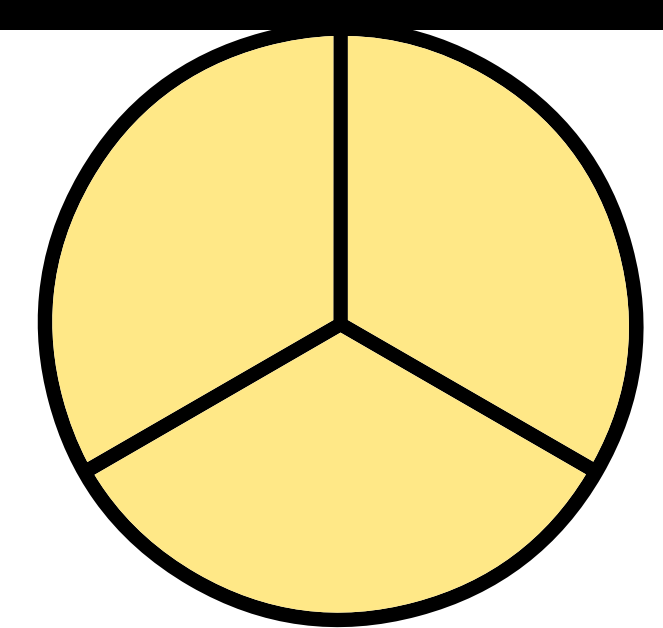
List the top 5 most ordered pizza types along with their quantities.

```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity_sum
FROM
    pizza_types
    JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity_sum DESC limit 5;
```



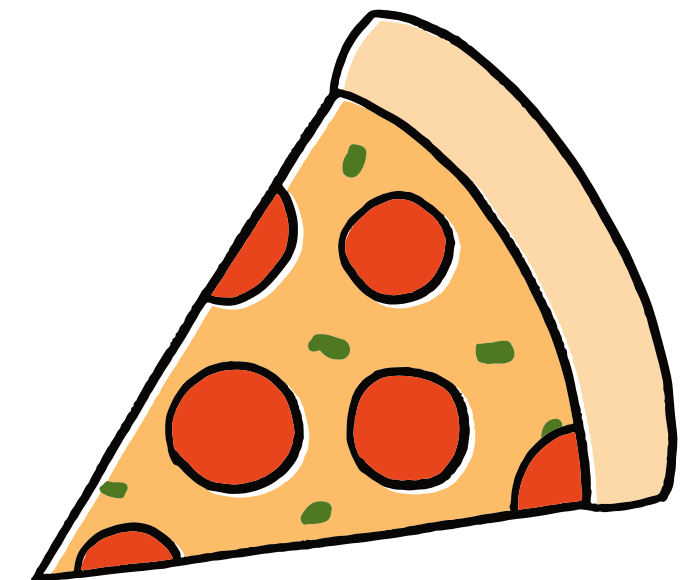


Question #6



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

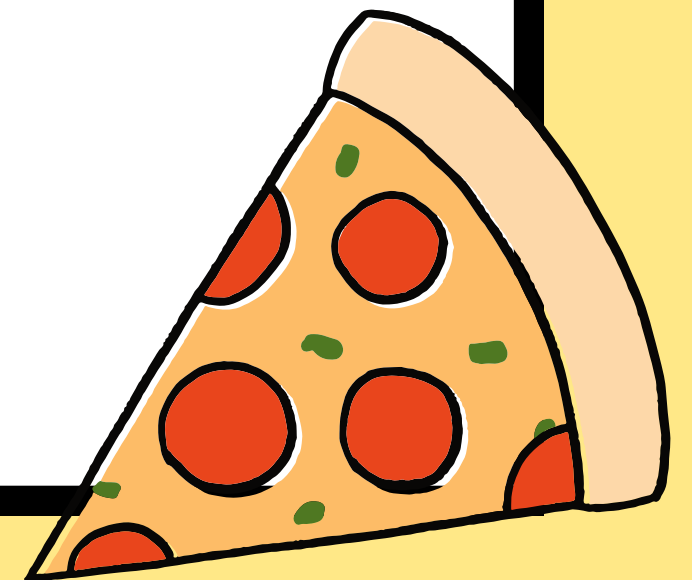
```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity_total
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity_total DESC;
```



Question #7

Determine the distribution of orders by hour of the day

```
SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY HOUR(order_time)
ORDER BY COUNT(order_id) desc;
```

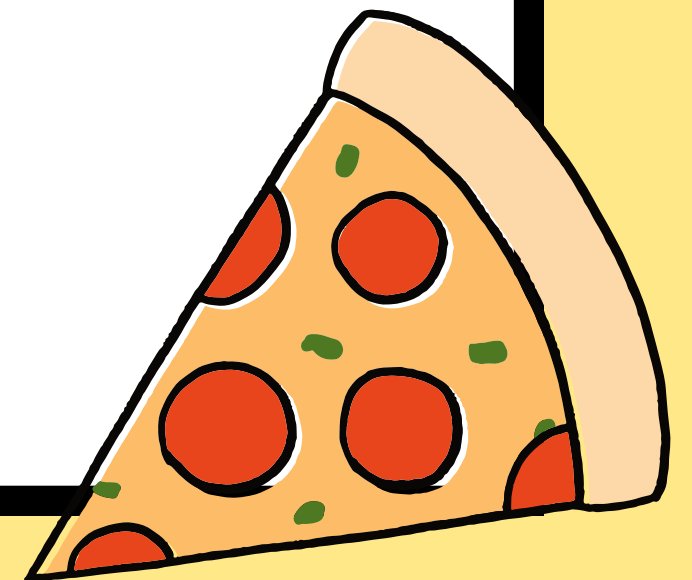


Question #8

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

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

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



Question #9

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

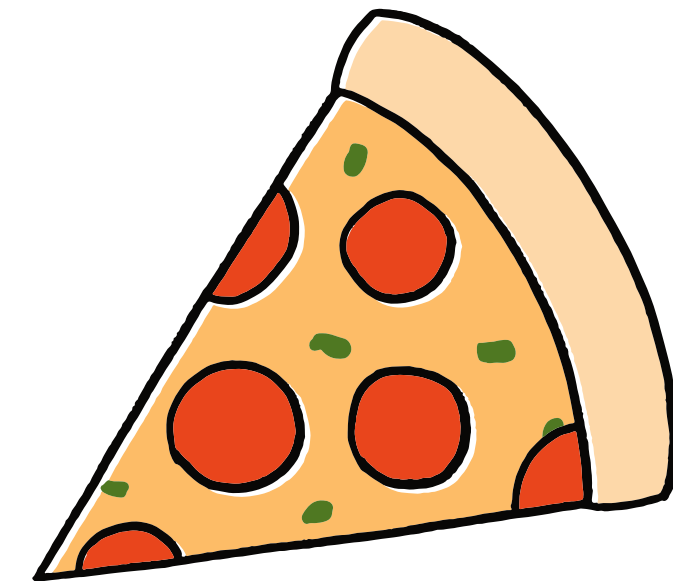
```
SELECT
    round(AVG(quantity),0) as average_order_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

	average_order_per_day
▶	61

Question #10

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

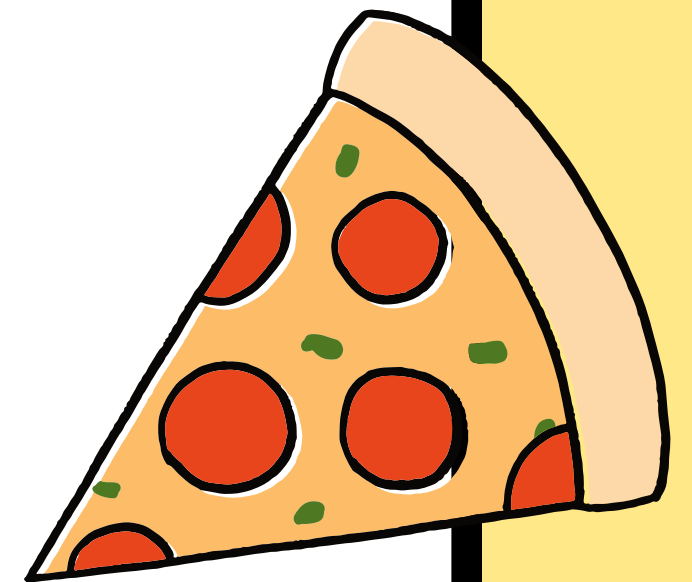
```
SELECT
    pizza_types.name,
    sum(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
group by pizza_types.name order by revenue desc limit 3 ;
```



Question #11

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

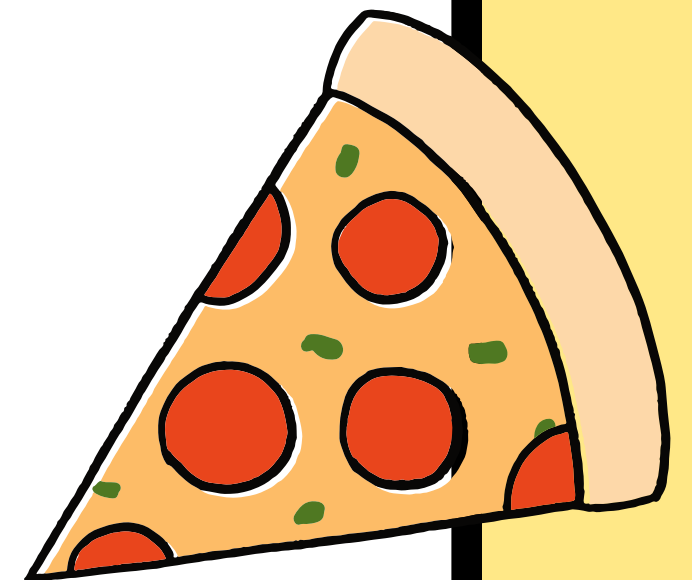
```
SELECT
    pizza_types.category,
    (SUM(pizzas.price * order_details.quantity) / (SELECT
        ROUND(SUM(pizzas.price * order_details.quantity),
            2) AS total_revenue
    FROM
        pizzas
        JOIN
        order_details ON pizzas.pizza_id = order_details.pizza_id)) * 100 AS revenue
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 pizza_types.category;
```



Question #12

Analyze the cumulative revenue generated over time.

```
select order_date,  
sum(revenue) over (order by order_date) as cumulative_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;
```





Thanks