

# Pizza Hut Sales

Saurabh Kumar Singh



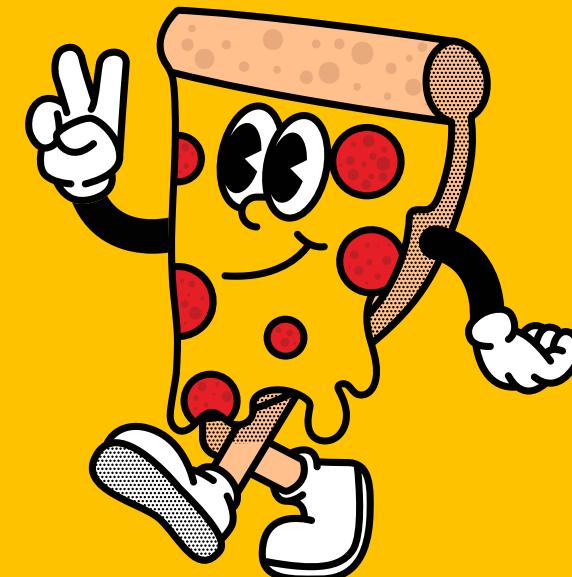
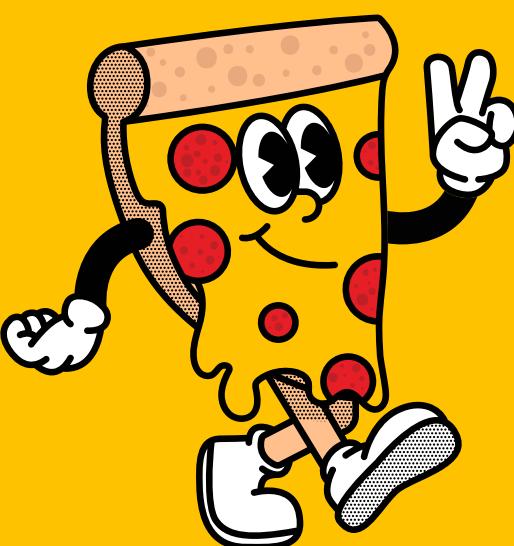
SQL PROJECT

# Retrieve the total number of orders placed.

```
SELECT  
    COUNT(order_id) AS TOTAL_NUMBER_OF_Orders  
FROM  
    orders;
```

Result Grid | Filter Rows:

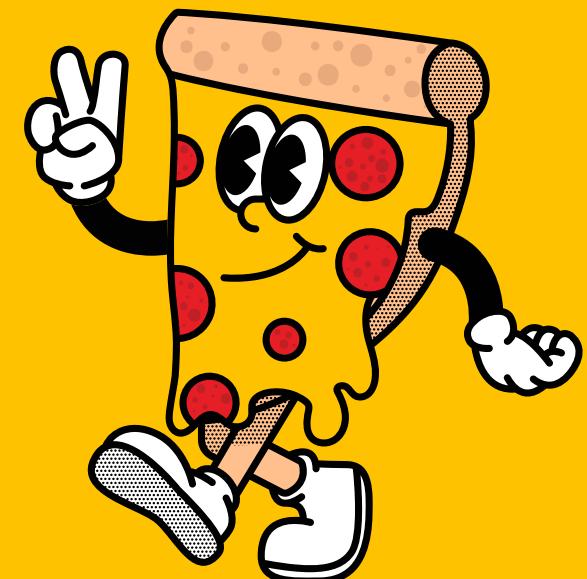
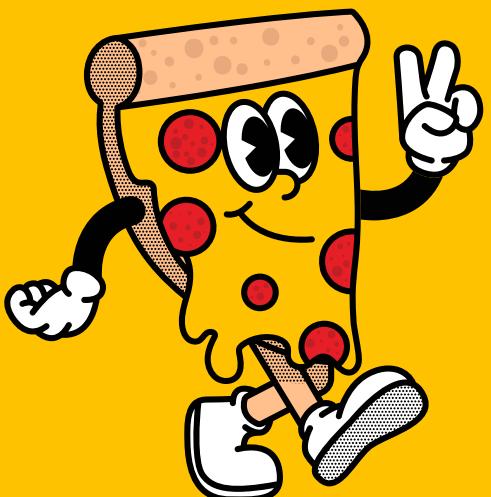
	TOTAL_NUMBER_OF_Orders
▶	21350



# Calculate the total revenue generated from pizza sales.

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

Result Grid	
	TOTAL_SALES
▶	817860.05

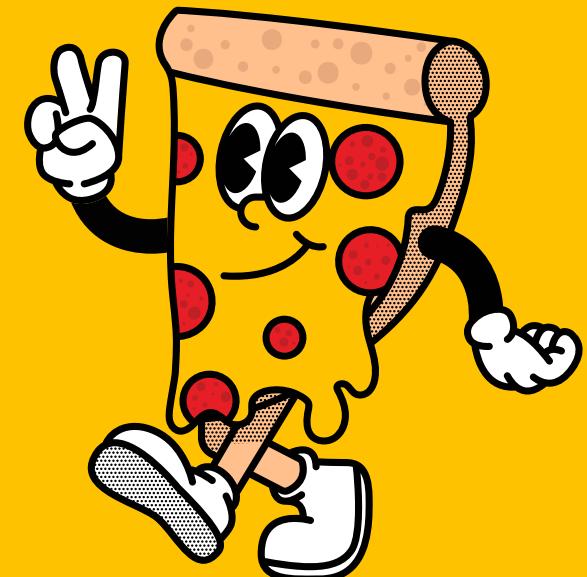
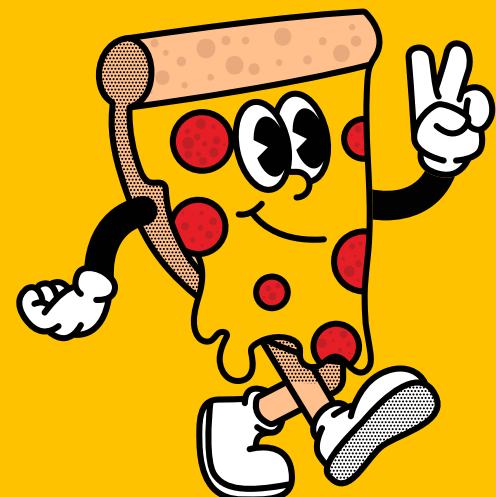


# Identify the highest-priced pizza.

```
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;
```

Result Grid | Filter Rows: [ ]

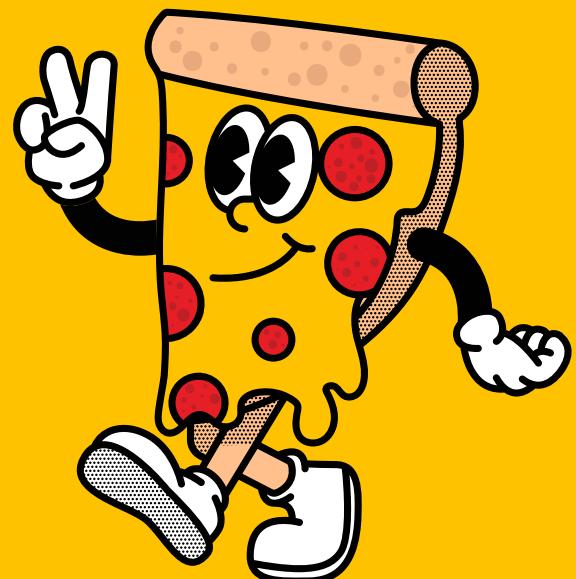
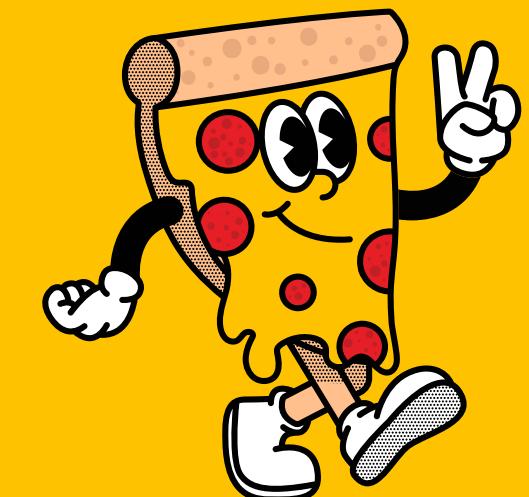
	name	price
▶	The Greek Pizza	35.95



# Identify the most common pizza size ordered

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

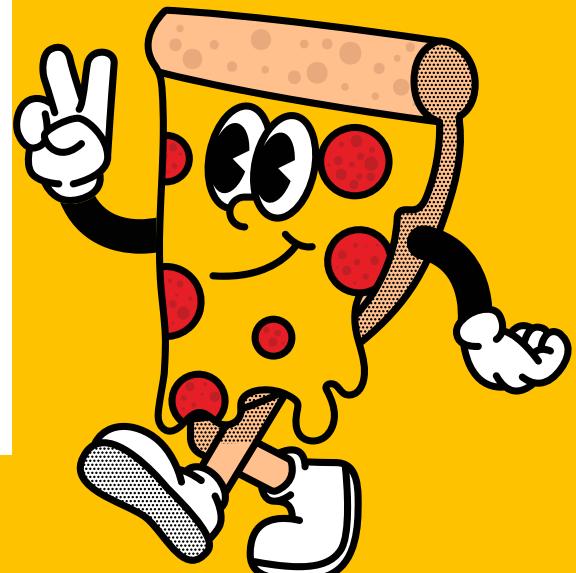
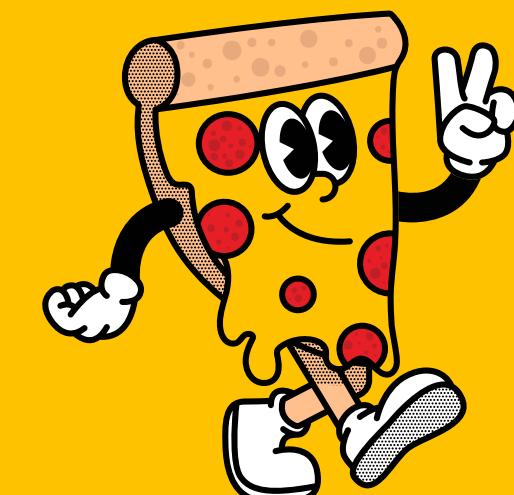
	size	order_count
▶	L	18526



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

```
SELECT  
    pizza_types.name,  
    SUM(order_details.quantity) AS total_quantity  
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 total_quantity DESC  
LIMIT 5;
```

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

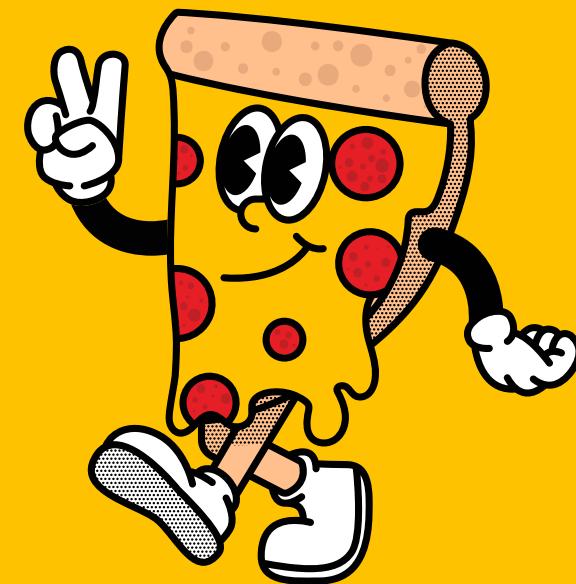
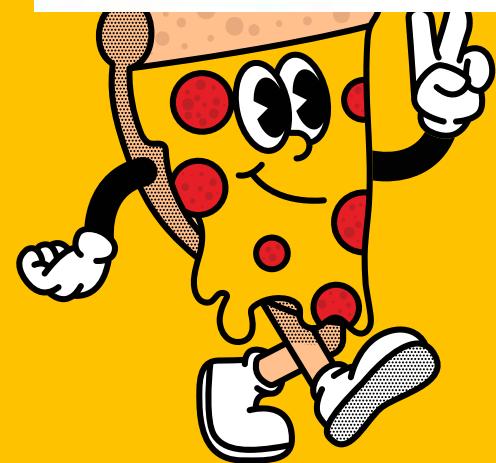


# Join the necessary tables to find the total quantity of each pizza category ordered.

SELECT

```
    pizza_types.category,  
    SUM(order_details.quantity) AS TOTAL_QUANTITY  
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 TOTAL_QUANTITY desc;
```

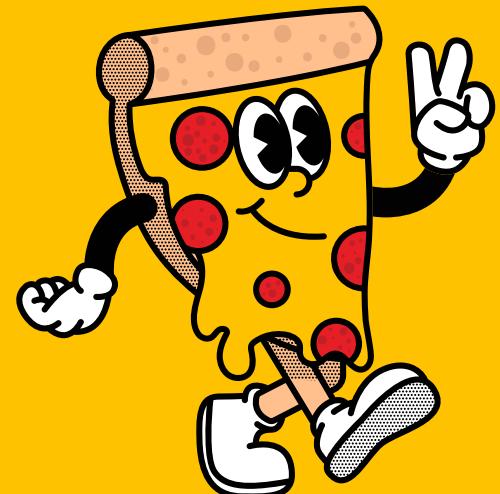
	category	TOTAL_QUANTITY
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



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

```
SELECT  
    HOUR(order_time) AS Hour_NO, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

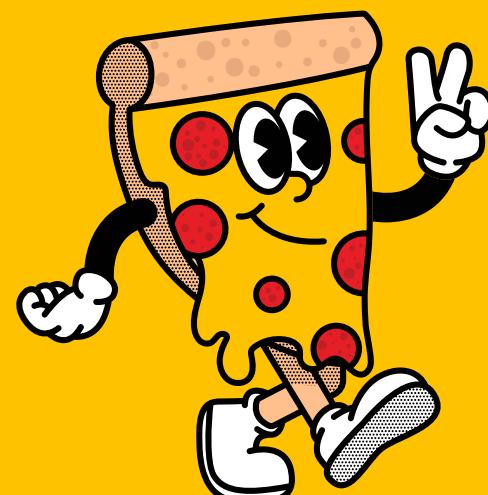
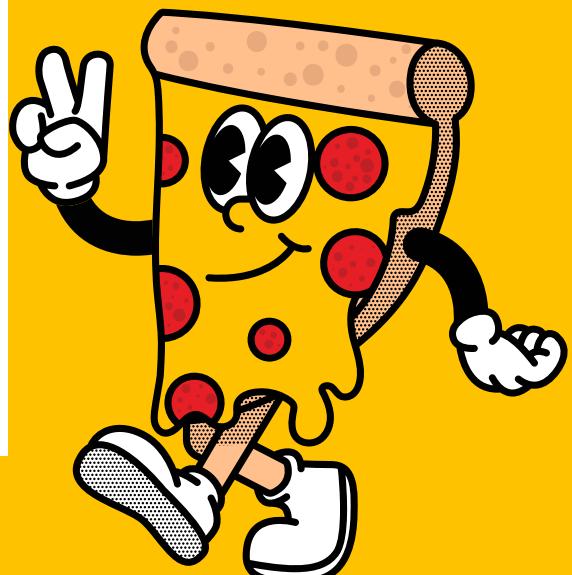
Hour_NO	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1



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

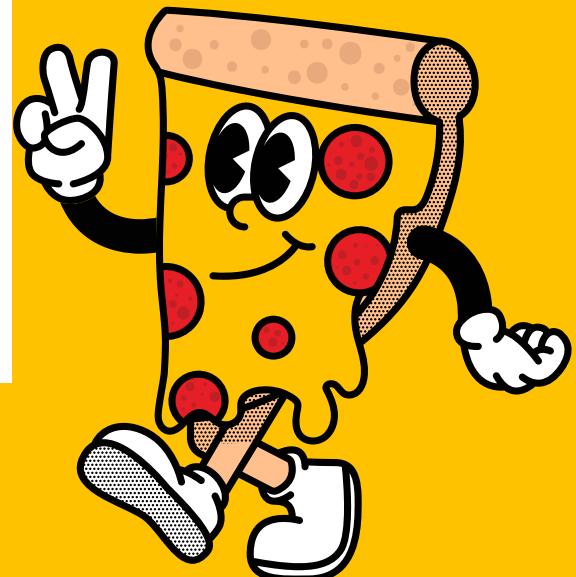
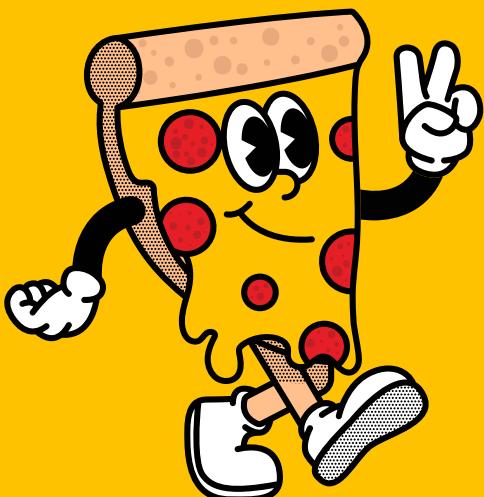


# Group the orders by date and calculate the average number of pizzas ordered per day

SELECT

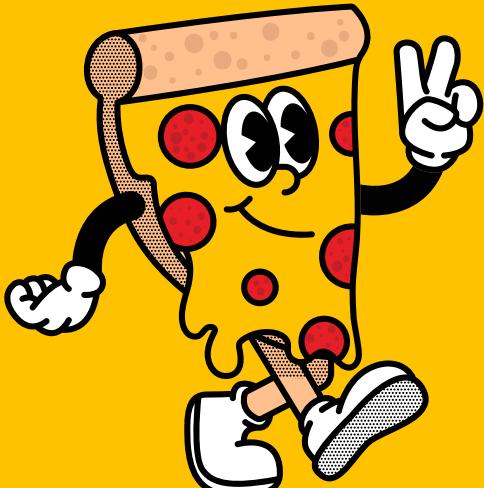
```
ROUND(AVG(quantity1), 0) AS AVG_Pizza_day  
FROM  
(SELECT  
    orders.order_date, SUM(order_details.quantity) AS quantity1  
FROM  
    orders  
JOIN order_details ON orders.order_id = order_details.order_id  
GROUP BY orders.order_date) AS ORDER_QUANTITY;
```

	AVG_Pizza_day
▶	138



# 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;
```



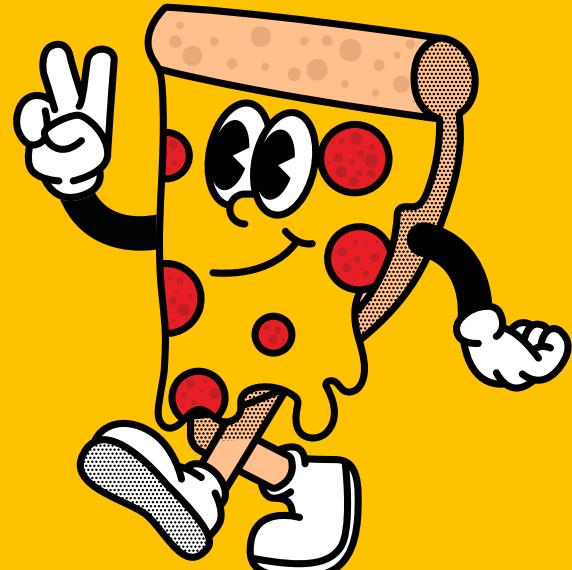
	name	Revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

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

SELECT

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

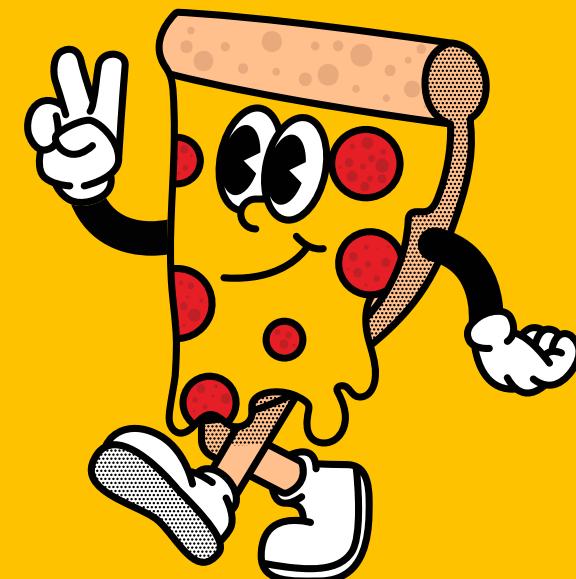
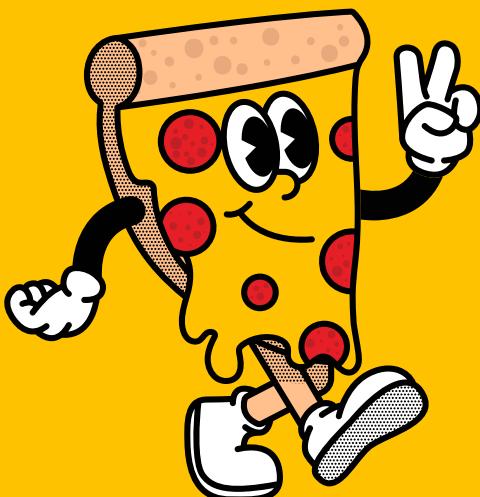
	category	Revenue
▶	Veggie	23.68
	Chicken	23.96
	Supreme	25.46
	Classic	26.91



# Analyze the cumulative revenue generated over time

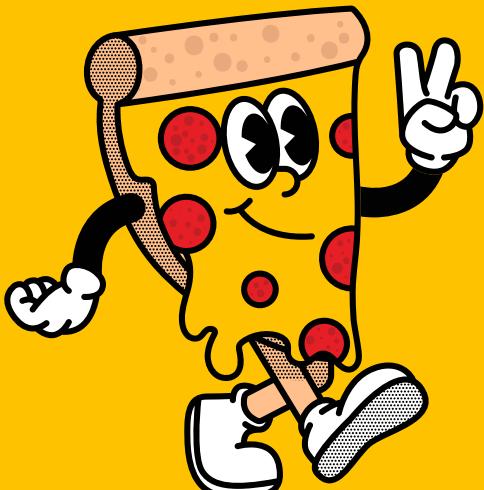
```
select order_date, sum(revenue) over (order by order_date) as Cum_Revenue  
from  
    (Select orders.order_date, sum(order_details.quantity*pizzas.price) as revenue  
     from order_details join pizzas  
     on order_details.pizza_id=pizzas.pizza_id  
     join orders on orders.order_id=order_details.order_id  
     group by orders.order_date) Sales ;
```

	order_date	Cum_Revenue
▶	2015-01-01 00:00:00	2713.850000000004
	2015-01-02 00:00:00	5445.75
	2015-01-03 00:00:00	8108.15
	2015-01-04 00:00:00	9863.6
	2015-01-05 00:00:00	11929.55
	2015-01-06 00:00:00	14358.5
	2015-01-07 00:00:00	16560.7
	2015-01-08 00:00:00	19399.05
	2015-01-09 00:00:00	21526.4



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

```
• select name, revenue from
  (Select category, name, revenue, rank() over (partition by category order by revenue desc) as RN from
  (
    SELECT pizza_types.category,
           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.category, pizza_types.name) as A) as B
  where RN <=3;
```



	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

Thank  
you!