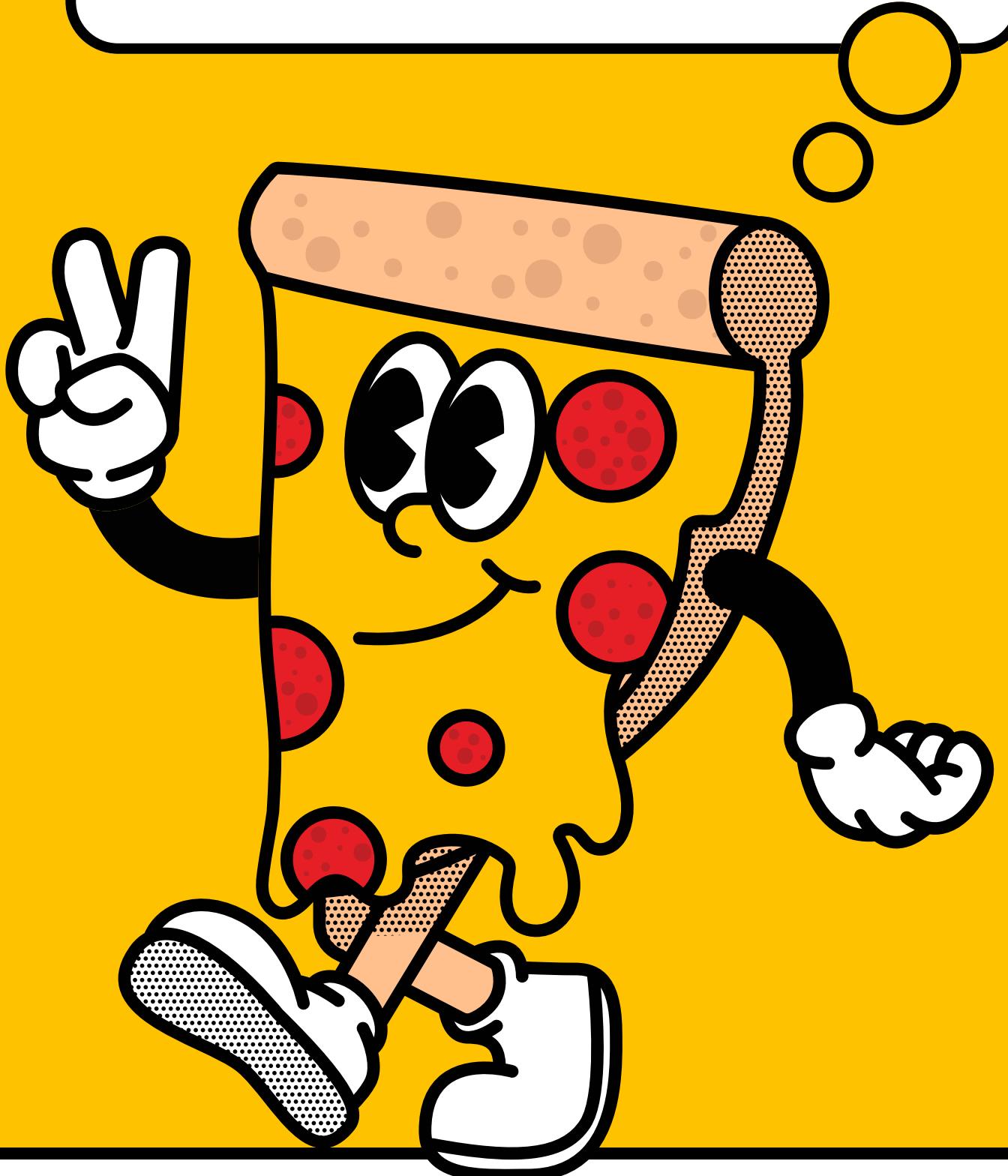


PIZZA SALES PROJECT

USING

SQL

21,000 ROWS OF DATA



WHAT'S THE TOTAL NUMBER OF ORDERS PLACED AND TOTAL REVENUE GENERATED?

-- 1. Total number of order placed

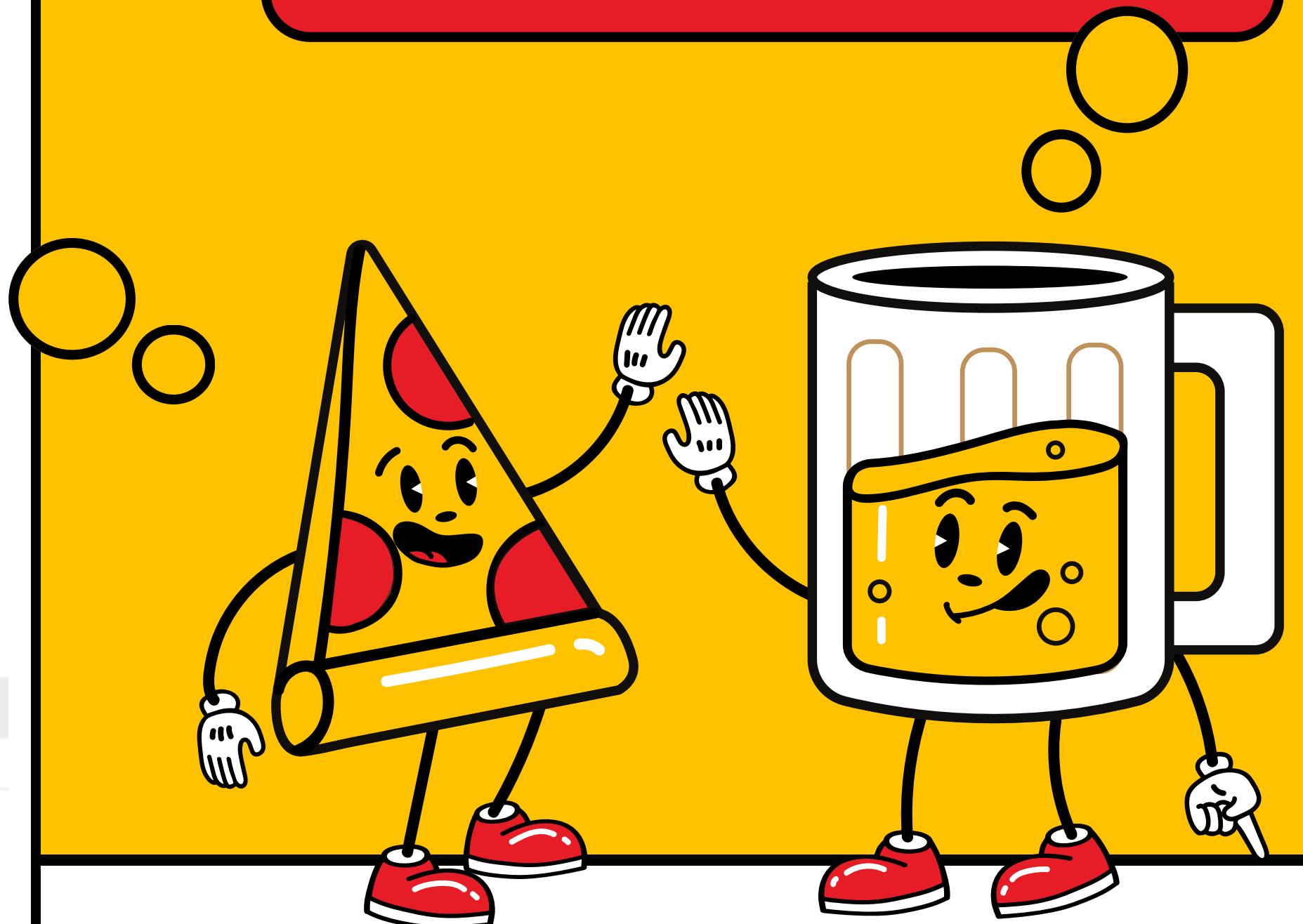
```
select count(order_id) as total_orders from orders;
```

Result Grid	
	total_orders
▶	21350

-- 2. The total revenue generated from pizza sales

```
select round(sum(a.quantity*b.price),2) as Total_Revenue  
from order_details as a  
join pizzas as b  
on a.pizza_id = b.pizza_id;
```

Result Grid	
	Total_Revenue
▶	817860.05



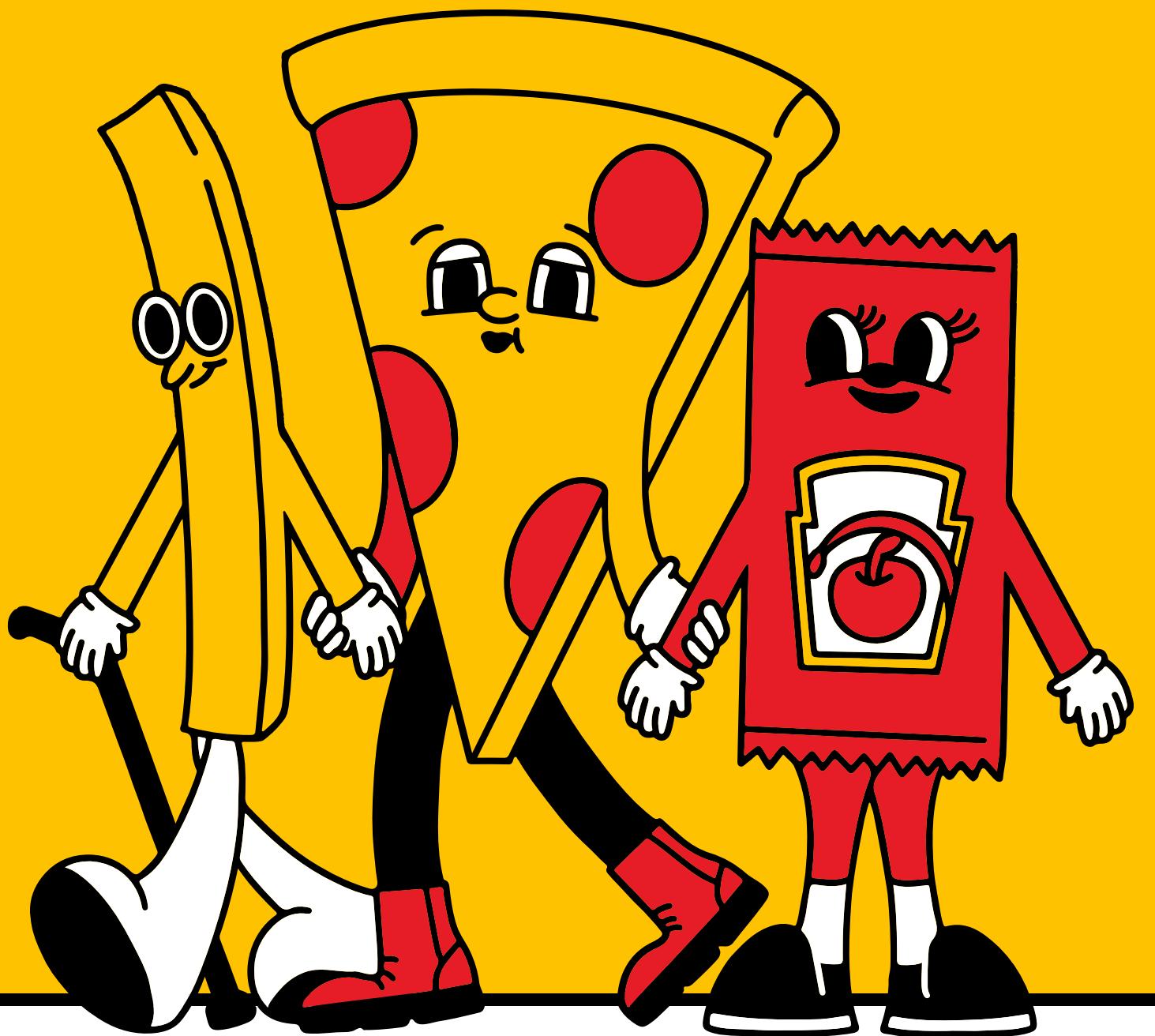
HIGHEST PRICE PIZZA

```
-- 3. The highest-priced pizza
select a.name, b.price
from pizza_types as a
join pizzas as b
  on a.pizza_type_id = b.pizza_type_id
order by b.price desc limit 1;
```

	name	price
▶	The Greek Pizza	35.95



THE MOST ORDERED PIZZA SIZE



```
-- 4. The most common pizza size ordered  
select a.size, count(b.order_details_id) as count  
from pizzas as a  
join order_details as b  
on a.pizza_id = b.pizza_id  
group by a.size  
order by count desc limit 1;
```

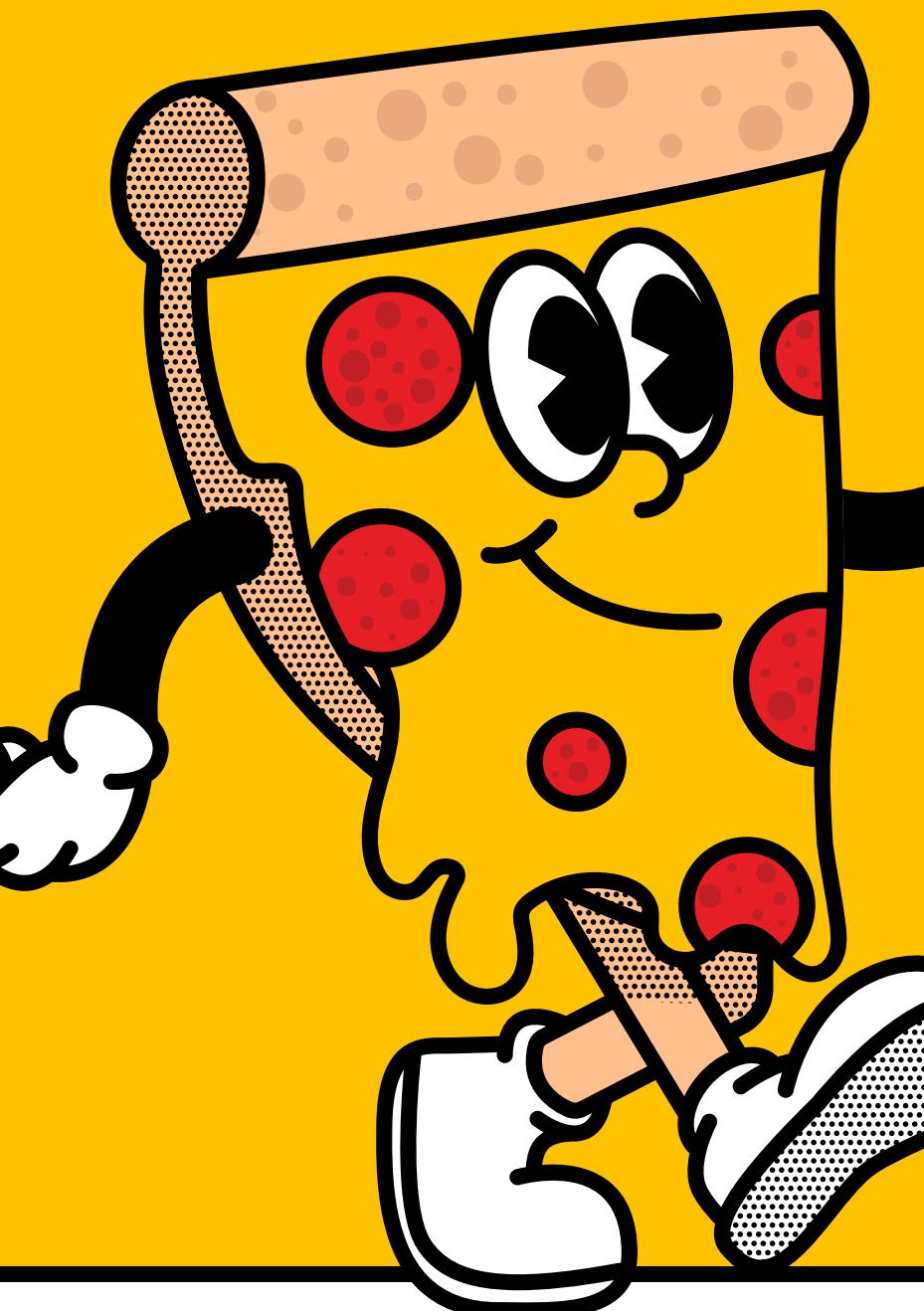
	size	count
▶	L	18526

MOST ORDERED PIZZA TYPES

```
-- 5. The top 5 most ordered pizza types along with their quantities
select pizza_types.name as most_ordered_pizza, sum(order_details.quantity) as quantity
from pizzas
join order_details
  on pizzas.pizza_id = order_details.pizza_id
join pizza_types
  on pizza_types.pizza_type_id = pizzas.pizza_type_id
group by most_ordered_pizza
order by quantity desc limit 5;
```

most_ordered_pizza	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED!



```
-- 6. The total quantity of each pizza category ordered
select pizza_types.category as category, sum(order_details.quantity) as total_quantity
from pizza_types
join pizzas
  on pizzas.pizza_type_id = pizza_types.pizza_type_id
join order_details
  on order_details.pizza_id = pizzas.pizza_id
group by category
order by total_quantity desc;
```

Result Grid | Filter Rows: _____

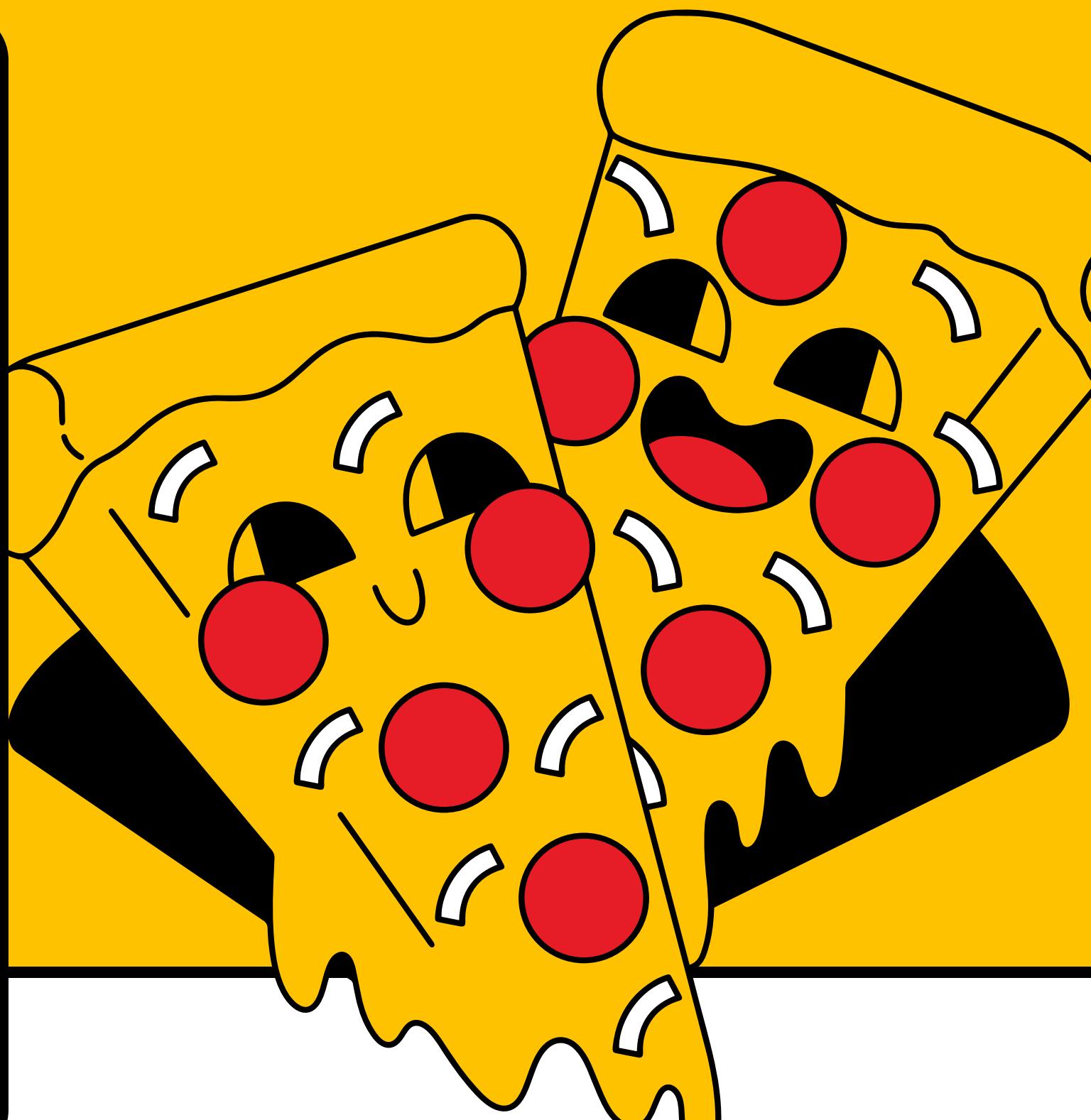
category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

ORDERS EACH HOURS IN A DAY

```
-- 7. The distribution of orders by hour of the day  
select hour(order_time) as hours, count(order_id) as orders  
from orders  
group by hours;
```

Result Grid | Filter Rows:

hours	orders
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198



WHAT'S THE QUANTITY OF ORDERS IN EACH CATEGORY



```
-- 6. The total quantity of each pizza category ordered
select pizza_types.category as category, sum(order_details.quantity) as total_quantity
from pizza_types
join pizzas
  on pizzas.pizza_type_id = pizza_types.pizza_type_id
join order_details
  on order_details.pizza_id = pizzas.pizza_id
group by category
order by total_quantity desc;
```

Result Grid | Filter Rows:

category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

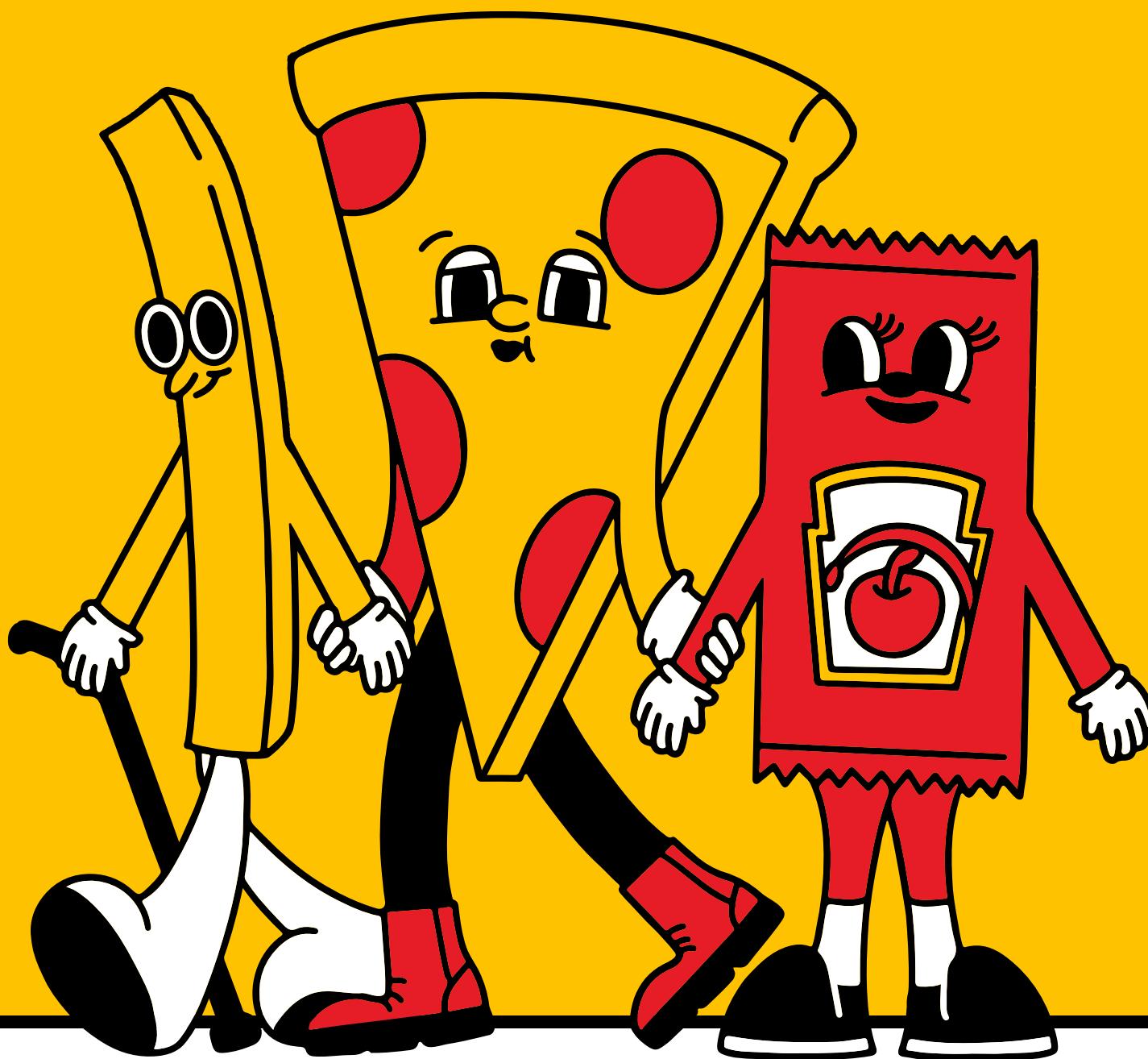
CATEGORY WISE PIZZA

```
-- 8. 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



AVERAGE ORDERS PER DAY



```
-- 10. The average orders per day
select avg(order_count) as orders_per_day
from (
    select order_date, count(order_id) as order_count
    from orders
    group by order_date
) as orders_per_date;
```

orders_per_day
59.6369

PIZZAS PER DAY



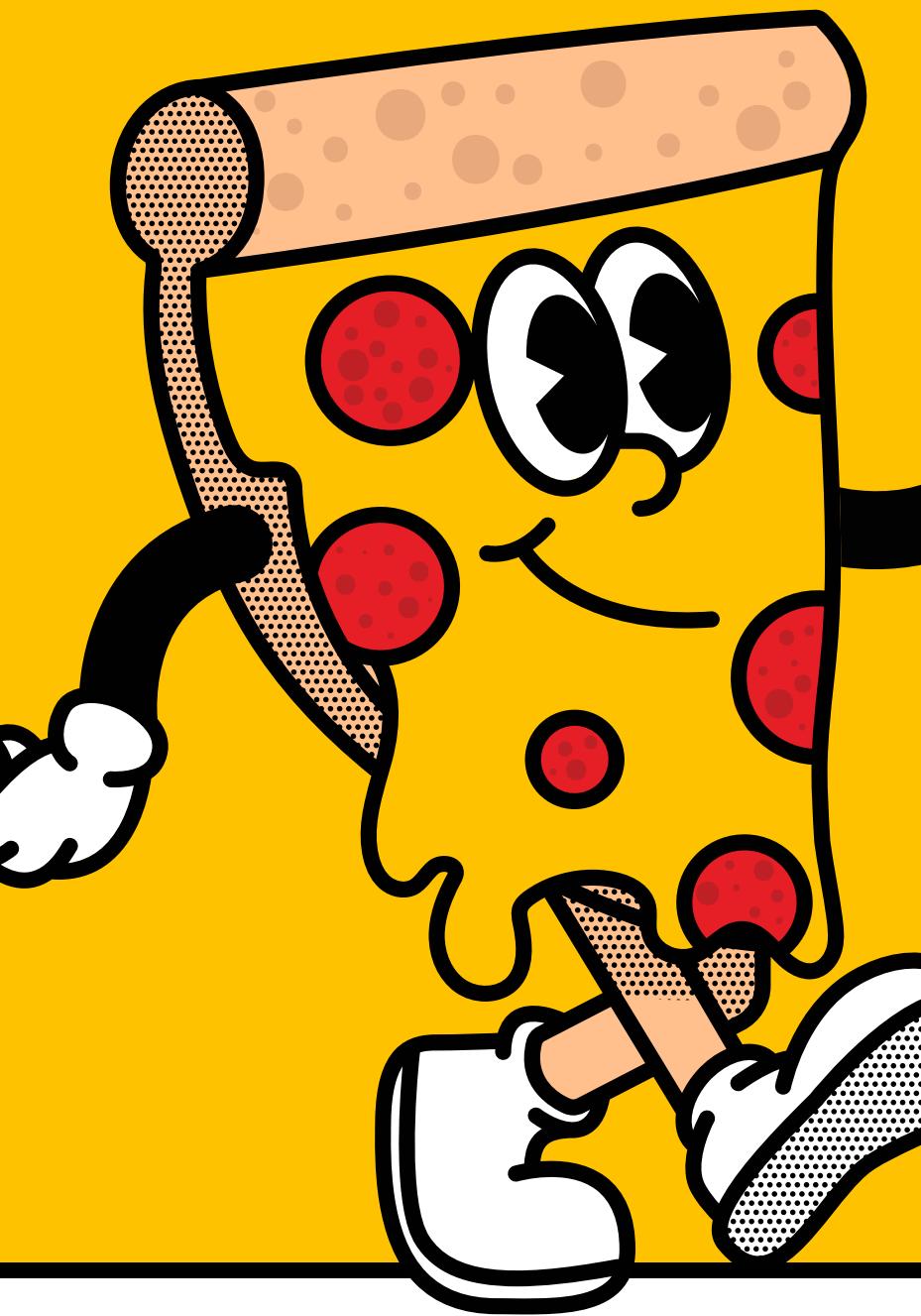
```
-- 11. Average number of pizzas per day
select round(avg(num_of_pizzas),0) as avg_num_of_pizzas_per_day
from (
    select orders.order_date, sum(order_details.quantity) as num_of_pizzas
    from orders
    join order_details
        on order_details.order_id = orders.order_id
    group by orders.order_date)
as num_of_pizzas_per_day;
```

Result Grid | Filter Rows:

	avg_num_of_pizzas_per_day
▶	138



MOST ORDERED PIZZA TYPE BASED ON REVENUE



```
-- 12. The top 3 most ordered pizza types based on revenue
select pizza_types.name as most_ordered_pizza, sum(pizzas.price*order_details.quantity) as revenue
from pizza_types
join pizzas
  on pizzas.pizza_type_id = pizza_types.pizza_type_id
join order_details
  on order_details.pizza_id = pizzas.pizza_id
group by most_ordered_pizza
order by revenue desc limit 3;
```

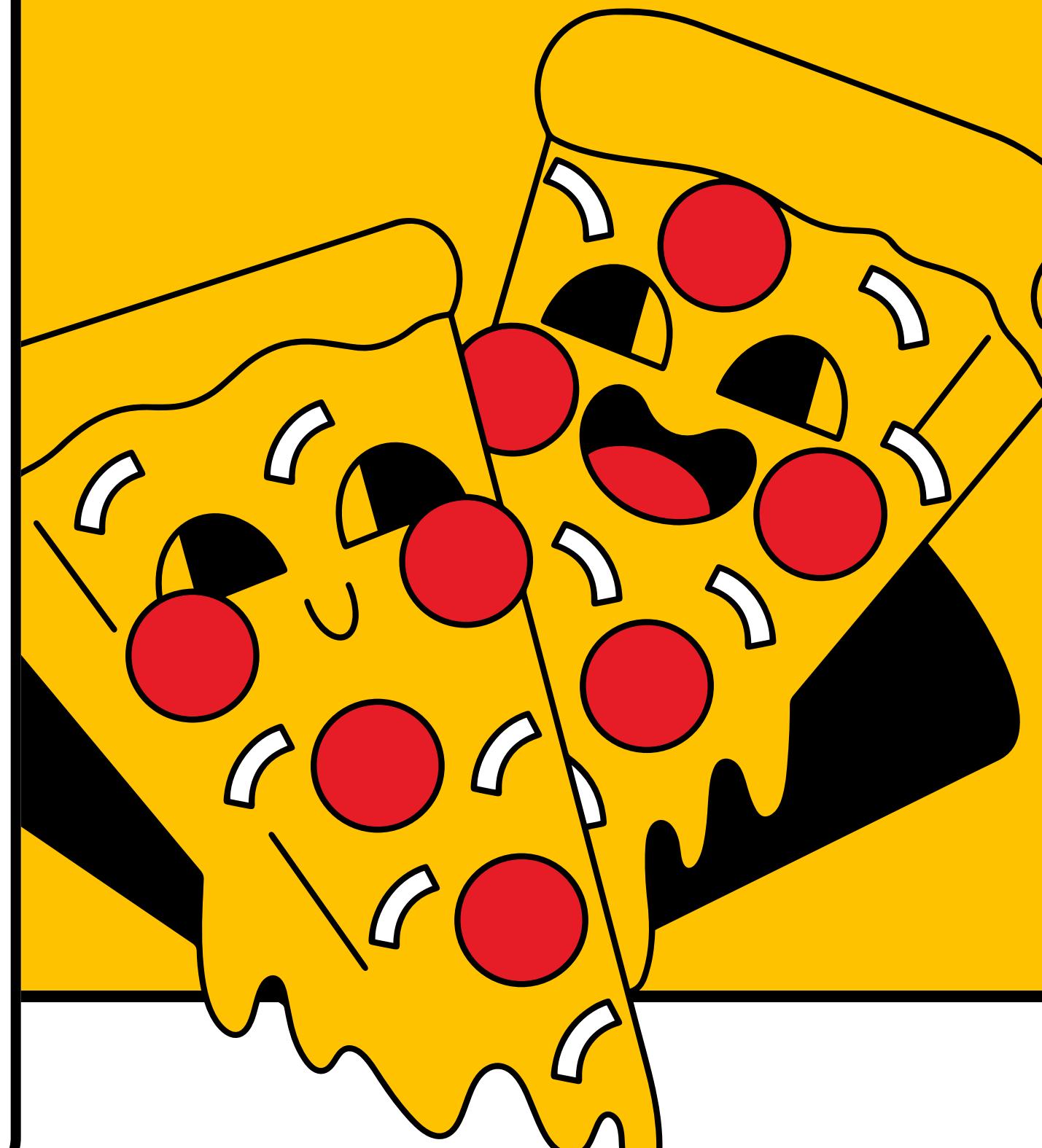
Result Grid | Filter Rows:

most_ordered_pizza	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

PERCENTAGE CONTRIBUTION

```
-- 13. The percentage contribution of each pizza category to total revenue  
  
select pizza_types.category as Pizza_Category,  
       round((sum(pizzas.price*order_details.quantity) /  
              (select round(sum(order_details.quantity*pizzas.price),2) as total_rev  
               from order_details  
              join pizzas  
                 on order_details.pizza_id = pizzas.pizza_id)*100),2) as percentage  
  
from pizza_types  
join pizzas  
   on pizzas.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
   on order_details.pizza_id = pizzas.pizza_id  
group by Pizza_Category;
```

Result Grid	
Pizza_Category	percentage
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96



WHAT'S THE CUMMULATIVE REVENUE BY DATE?



```
-- 14. The cumulative revenue generated over time
• select date, sum(day_revenue) over (order by date) as cummulative_revenue
  from (select orders.order_date as date,
              round(sum(order_details.quantity*pizzas.price),2) as day_revenue
        from orders
        join order_details
          on orders.order_id = order_details.order_id
        join pizzas
          on order_details.pizza_id = pizzas.pizza_id
       group by date) as sales;
```

Result Grid		
	date	cummulative_revenue
▶	2015-01-01	2713.85
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55
	2015-01-06	14358.5
	2015-01-07	16560.7
	2015-01-08	19399.05
	2015-01-09	21526.399999999998
	2015-01-10	23990.35
	2015-01-11	25047.410000000003

MOST ORDER PIZZA BASED ON REVENUE BY RANK

```
-- 15. The top 3 most ordered pizza types based on revenue for each pizza category
with
    pizza_rev_cte as ( select pizza_types.category as pizza_category,
                            pizza_types.name as name, sum(order_details.quantity*pizzas.price) as revenue
                        from pizza_types
                        join pizzas
                            on pizzas.pizza_type_id = pizza_types.pizza_type_id
                        join order_details
                            on order_details.pizza_id = pizzas.pizza_id
                        group by pizza_category, name),
    pizza_rank_cte as ( select pizza_category, name, revenue,
                            rank() over (partition by pizza_category order by revenue desc) as pizza_rank
                        from pizza_rev_cte)
```



MOST ORDER PIZZA BASED ON REVENUE BY RANK

Result Grid | Filter Rows: Export: Wrap Cell Content:

pizza_category	name	revenue	pizza_rank
Chicken	The Thai Chicken Pizza	43434.25	1
Chicken	The Barbecue Chicken Pizza	42768	2
Chicken	The California Chicken Pizza	41409.5	3
Classic	The Classic Deluxe Pizza	38180.5	1
Classic	The Hawaiian Pizza	32273.25	2
Classic	The Pepperoni Pizza	30161.75	3
Supreme	The Spicy Italian Pizza	34831.25	1
Supreme	The Italian Supreme Pizza	33476.75	2
Supreme	The Sicilian Pizza	30940.5	3
Veggie	The Four Cheese Pizza	32265.70000000065	1
Veggie	The Mexicana Pizza	26780.75	2
Veggie	The Five Cheese Pizza	26066.5	3



THANK YOU

MADE BY SHIVANI SINGH