HOT & SPICY

PIZZA

50% OFF

ORDER NOW!

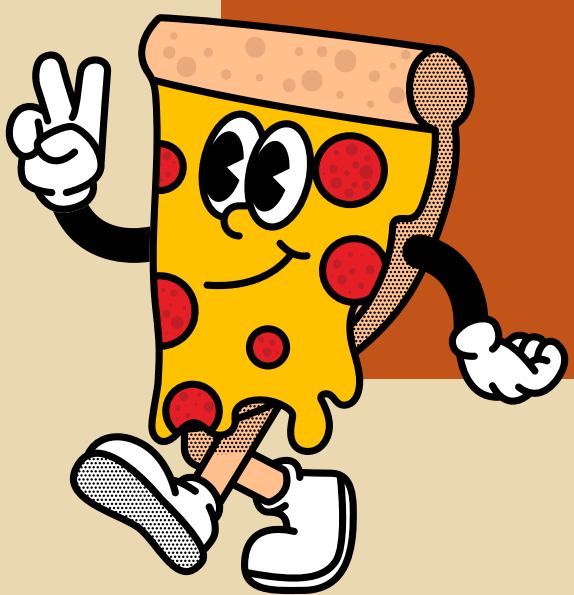
WWW.REALLYGREATSITE.COM

PIZZA SALES REPORT

In this project, we analyze the sales data of a fictional pizza shop using SQL to gain insights into business performance. The data set includes information on pizza types, orders, order details, and customer demographics, allowing for a comprehensive examination of sales trends, customer behavior, and product popularity.

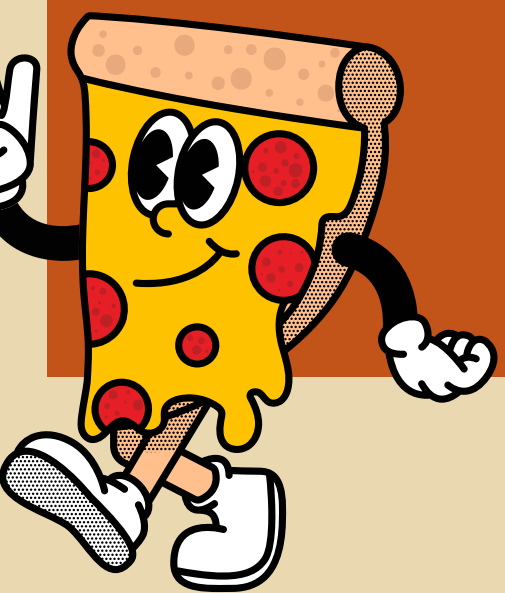
1)Retrieve the total number of orders placed.

```
CREATE DATABASE pizzhut
USE pizzhut
SELECT COUNT(order_id) FROM orders;
```



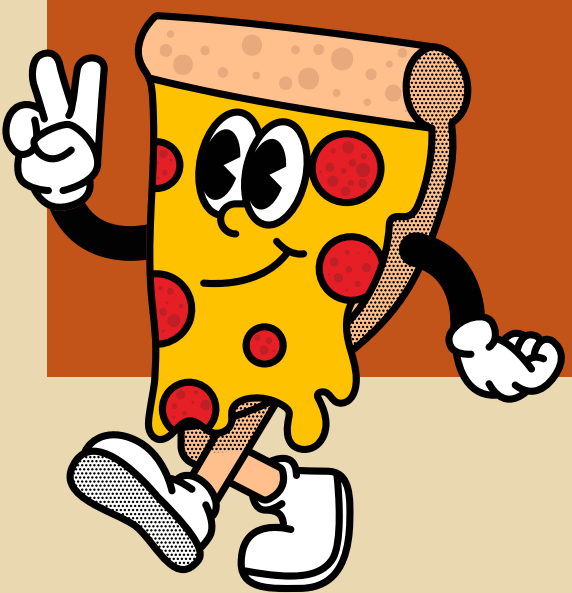
2)Calculate the total revenue generated from pizza sales.

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



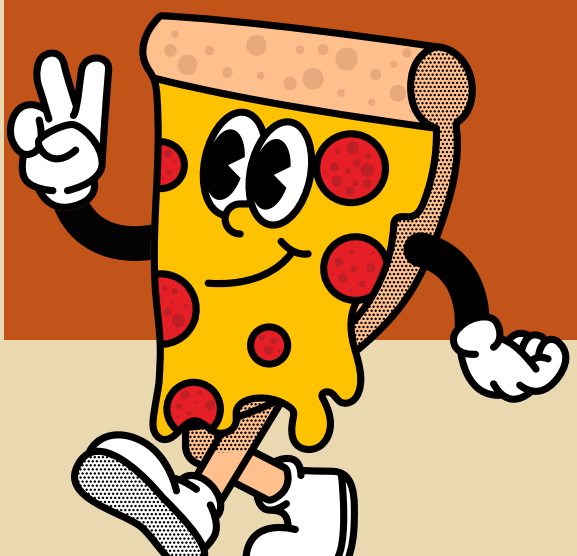
Identify the highest-priced pizza.

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



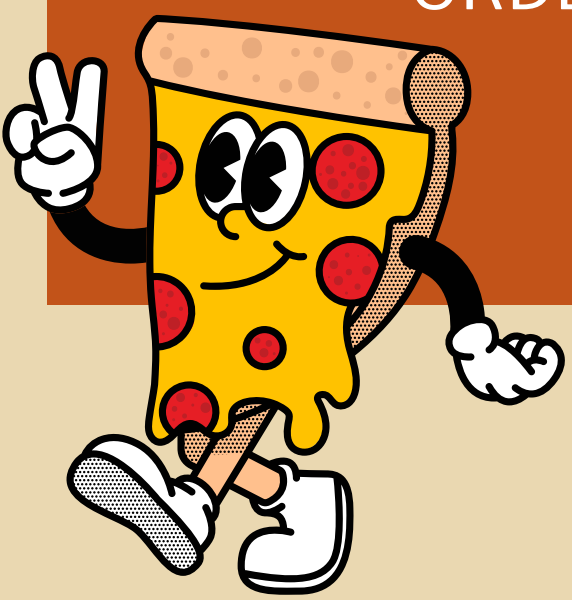
4) Identify the most common pizza size ordered.

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



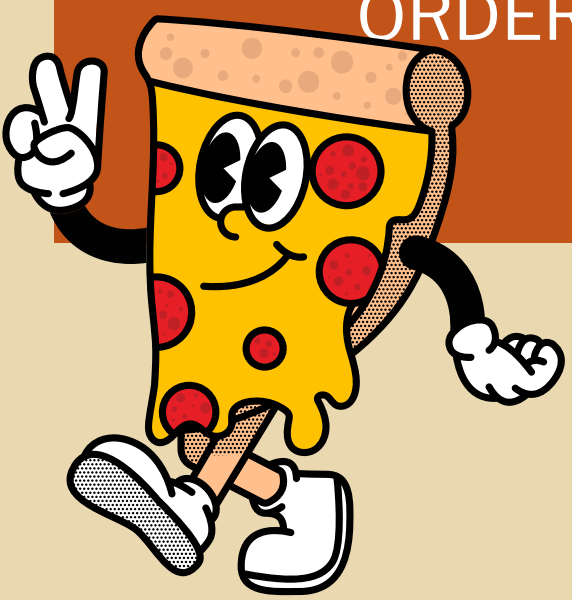
5) List the top 5 most ordered pizza types along with their quantities.

```
SELECT SUM(order_details.quantity) , pizza_types.name
FROM (( pizzas
INNER JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id)
INNER JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id)
GROUP BY pizza_types.name
ORDER BY COUNT(order_details.quantity)
LIMIT 5 ;
```



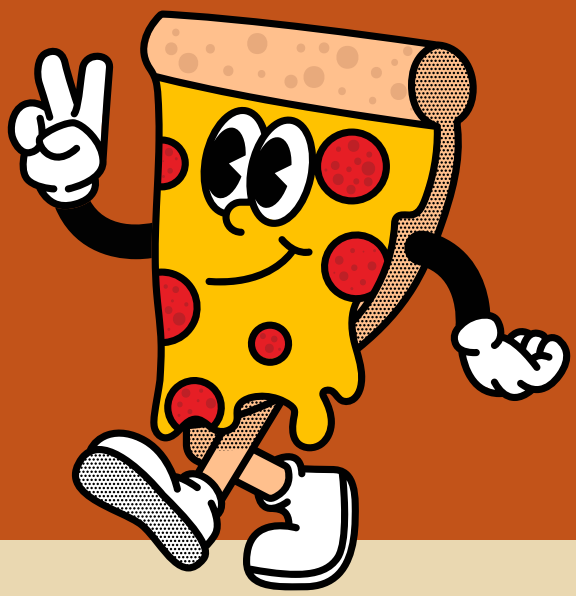
6) Join the necessary tables to find the total quantity of each pizza category ordered.

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



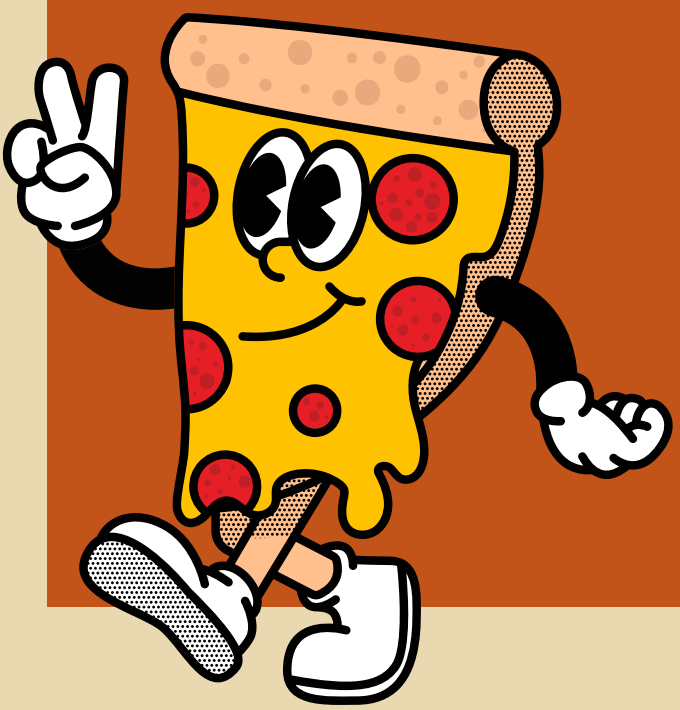
7) Determine the distribution of orders by hour of the day.

```
SELECT hour(time), COUNT(order_id)
FROM orders
GROUP BY hour(time) ;
```



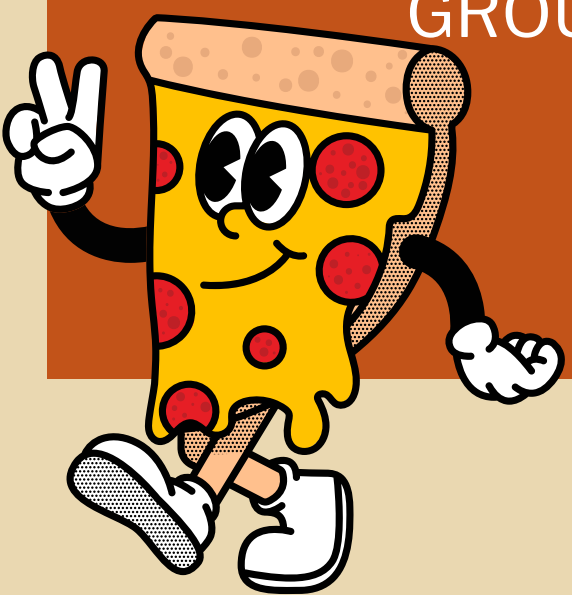
8) Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT category ,COUNT(name)
FROM pizza_types
GROUP BY category;
```



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

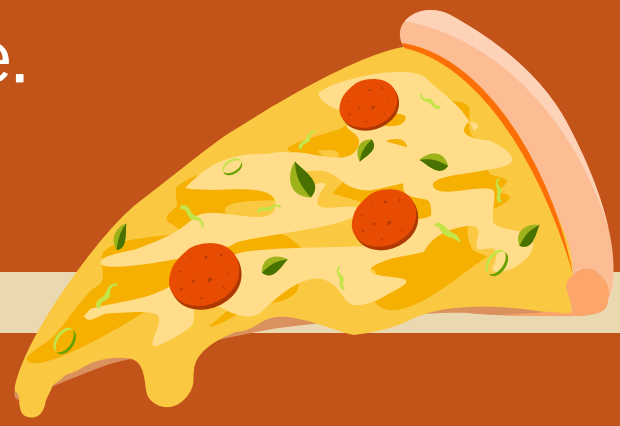
```
SELECT ROUND(AVG(quantity),0) FROM(
SELECT orders.date , SUM(order_details.quantity) AS quantity
FROM orders
LEFT JOIN order_details
ON orders.order_id = order_details.order_id
GROUP BY orders.date) AS order_quantity;
```



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

```
SELECT pizza_types.name ,
SUM(pizzas.price*order_details.quantity) AS Total_Revenue
FROM ((pizza_types
LEFT JOIN pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id)
LEFT JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id)
GROUP BY pizza_types.name
ORDER BY Total_Revenue DESC LIMIT 3;
```

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



```
SELECT pizza_types.category ,
ROUND(SUM(pizzas.price*order_details.quantity) / ( SELECT
(SUM(pizzas.price*order_details.quantity) )
FROM pizzas
LEFT JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id)*100,2) AS Revenue
FROM ((pizza_types
LEFT JOIN pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id)
LEFT JOIN order_details
 pizzas.pizza_id = order_details.pizza_id)
GROUP BY pizza_types.category
ORDER BY Revenue DESC;
```

12) Analyze the cumulative revenue generated over time.

```
SELECT pizza_types.category ,
ROUND(SUM(pizzas.price*order_details.quantity) / ( SELECT
(SUM(pizzas.price*order_details.quantity) )
FROM pizzas
LEFT JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id)*100,2) AS Revenue
FROM ((pizza_types
LEFT JOIN pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id)
LEFT JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id)
GROUP BY pizza_types.category
ORDER BY Revenue DESC;
```