Pizza Hut Sales Analysis Using SQL

This presentation outlines a data-driven approach to analyzing Pizza Hut's sales data using SQL, providing key insights into order patterns, popular items, and revenue trends.

Project Overview & Objectives

Project Goal

To analyze Pizza Hut's sales and order data to uncover actionable business insights using SQL.

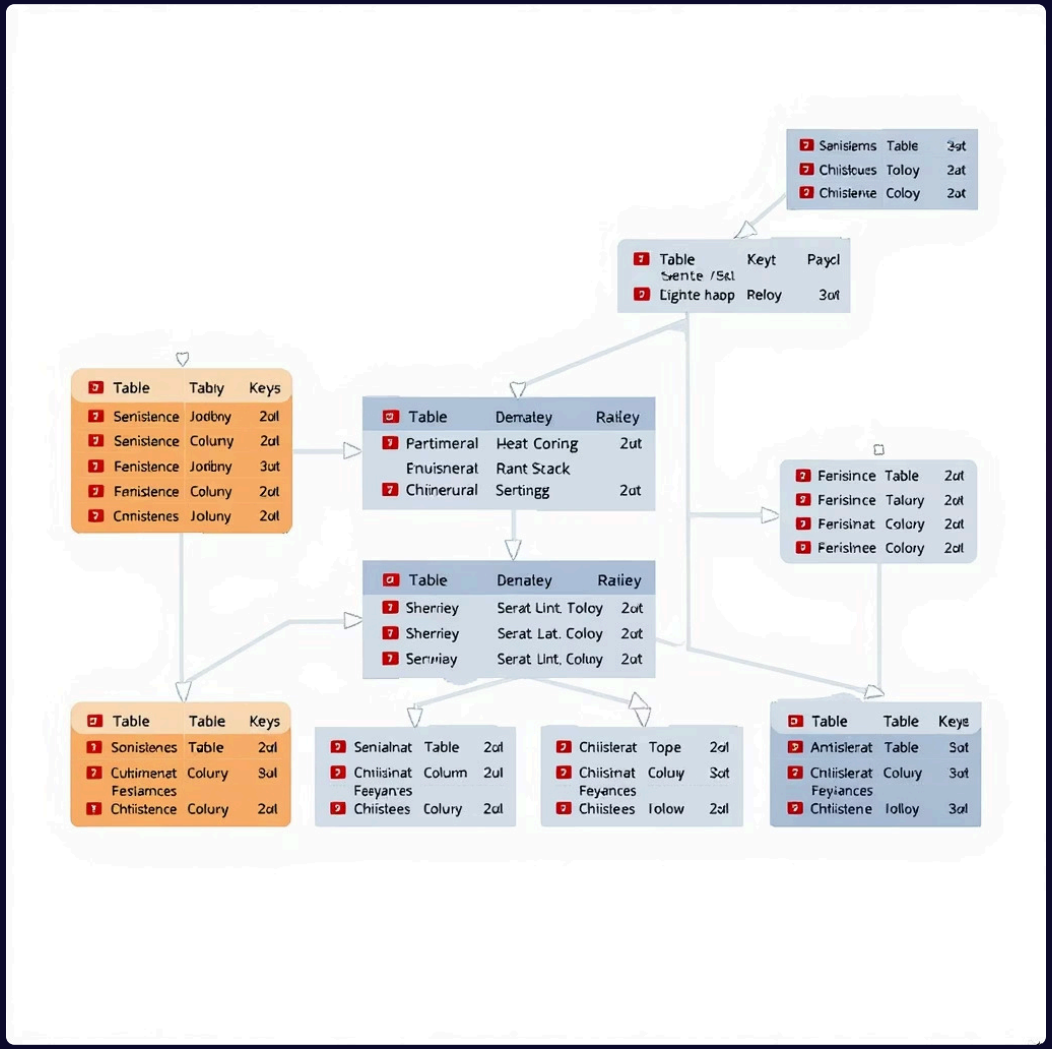
Data-Driven Insights

Focus on understanding customer behavior, popular products, and sales performance over time.

SQL Application

Demonstrate practical SQL skills for data extraction, transformation, and analysis.

Database Schema Overview



The analysis utilizes a well-structured database with four core tables, each playing a crucial role in capturing comprehensive sales data.

- **orders:** Contains unique order IDs and timestamps.
- **order_details:** Links orders to specific pizzas, including quantities.
- **pizzas:** Stores details about each pizza variant, including size and price.
- **pizza_types:** Provides high-level information on pizza types like names and ingredients.

Key Business Questions Addressed

Our SQL analysis provides answers to critical business questions, offering a 360-degree view of Pizza Hut's operations and sales performance.

Revenue Insights

What is the total revenue generated from pizza sales?

Popular Products

Which is the most common pizza size ordered?

What are the top 3 most ordered pizza types by revenue?

What are the top 5 most ordered pizza types (by quantity)?

Order Trends

What is the average number of pizzas ordered per day?

What is the distribution of orders by hour of the day?

What is the cumulative revenue over time?

What is the total number of orders placed?

Sales Distribution

What is the total quantity of pizzas ordered by category?

Which is the lowest-priced pizza?

Which is the second highest-priced pizza?

What is the pizza stock by size?

Which is the highest-priced pizza?

What is the revenue percentage contributed by each pizza category?

What are the top 3 most ordered pizza types based on revenue per category?

How many pizza types are there in each category?

Which is the least common pizza size ordered?

Learnings & Skills Gained

Advanced SQL Techniques

Proficiency in complex joins, aggregations, and subqueries for meaningful data extraction.

Data Storytelling

Ability to translate raw data into compelling visual narratives and actionable insights.

Business Acumen

Deeper understanding of how data analysis directly impacts business decisions and strategies.

Conclusion & Next Steps

> This analysis provides a strong foundation for optimizing Pizza Hut's operations and maximizing sales. By leveraging SQL, we've transformed raw data into strategic assets.

Impact of Analysis:

- Informed menu adjustments based on popular items.
- Optimized staffing and resource allocation during peak hours.
- Targeted marketing campaigns for specific pizza categories and sizes.

Future Work:

- Predictive modeling for future sales forecasting.
- Customer segmentation based on ordering patterns.
- Geospatial analysis of sales performance across different locations.

What is the total revenue generated from pizza sales?

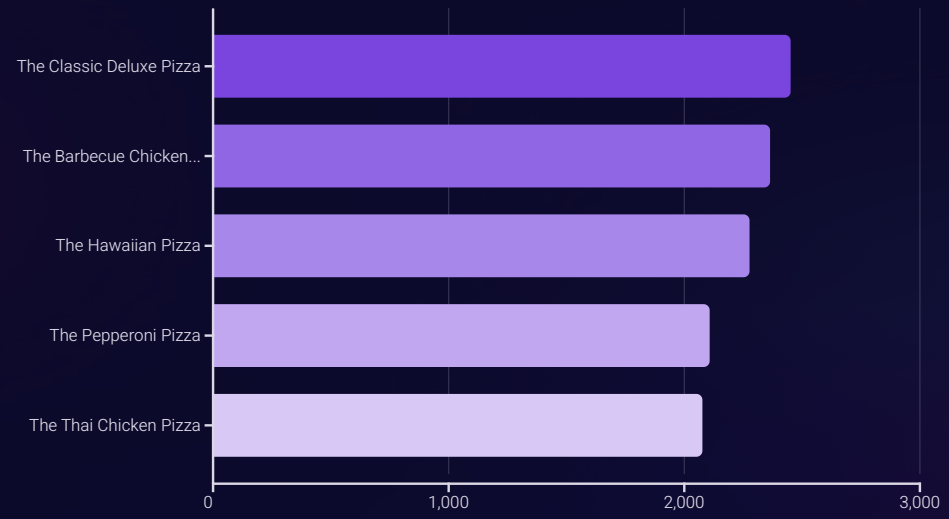
```
SELECT
    SUM(od.quantity * p.price) AS total_Revenue
FROM
    order_details od
        JOIN
    pizzas p ON p.pizza_id = od.pizza_id;

-- Identify the highest-priced pizza.

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

What are the top 5 most ordered pizza types (by quantity)?

```
SELECT
    pt.name AS pizza_type, SUM(od.quantity) AS total_quantity
FROM
    order_details od
        JOIN
    pizzas p ON od.pizza_id = p.pizza_id
        JOIN
    pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.name
ORDER BY total_quantity DESC
LIMIT 5;
```



What is the average number of pizzas ordered per day?

```
SELECT
    ROUND(AVG(Quantity), 0) as Avg_Pizza_Order_Per_Day
FROM
    (SELECT
        o.order_date AS Date, SUM(od.quantity) AS Quantity
    FROM
        orders o
    JOIN order_details od ON o.order_id = od.order_id
    GROUP BY o.order_date) AS Order_Quantity;
```

138

Pizzas ordered on average each day

501

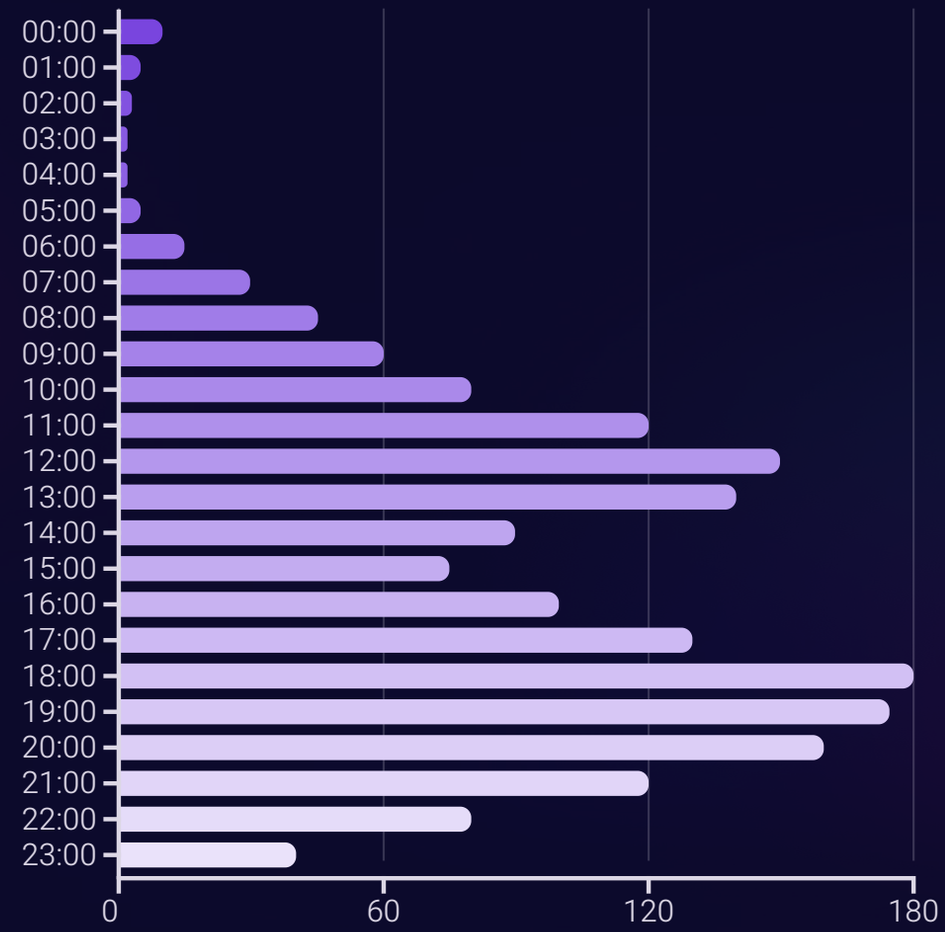
Highest daily pizza orders recorded

7

Lowest daily pizza orders recorded

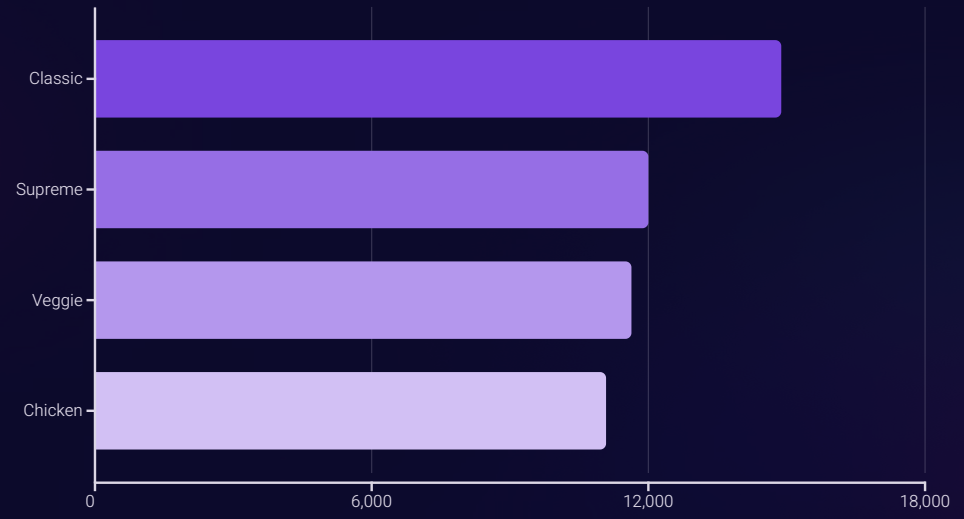
What is the distribution of orders by hour of the day?

```
SELECT
    HOUR(o.time) AS order_hour,
    COUNT(DISTINCT o.order_id) AS total_orders
FROM
    orders o
GROUP BY
    order_hour
ORDER BY
    order_hour;
```



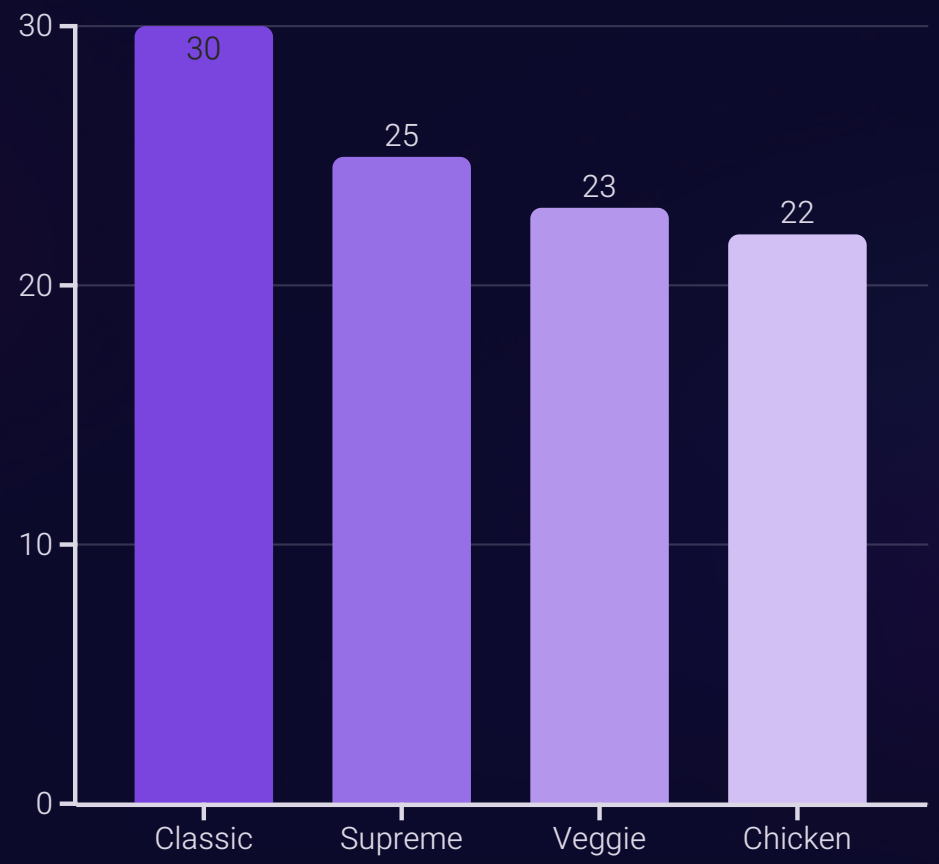
What is the total quantity of pizzas ordered by category?

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



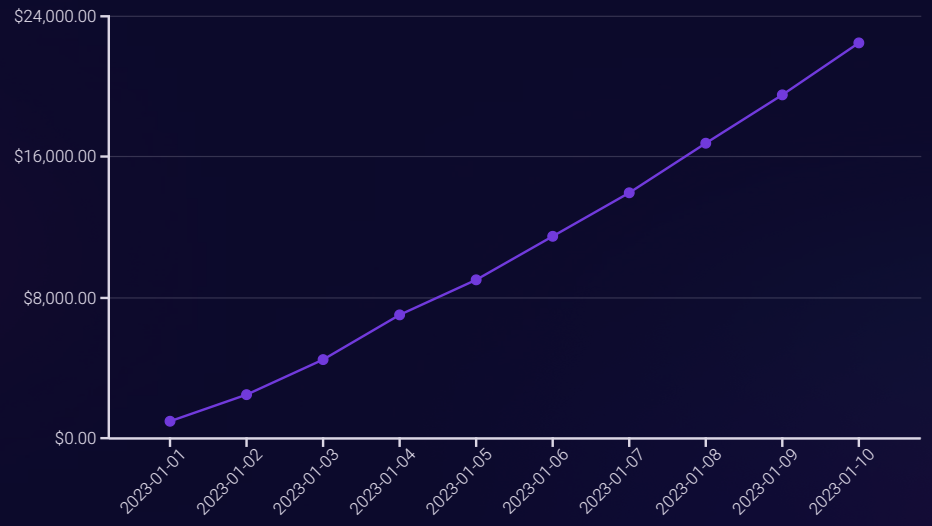
What is the revenue percentage contributed by each pizza category?

```
SELECT
pt.category,
CONCAT(
    ROUND(
        (SUM(od.quantity * p.price) /
         (SELECT SUM(od2.quantity * p2.price)
          FROM order_details od2
          JOIN pizzas p2 ON p2.pizza_id = od2.pizza_id)
        ) * 100, 2
    ), '%') AS revenue_percentage
FROM pizza_types pt
JOIN pizzas p ON pt.pizza_type_id = p.pizza_type_id
JOIN order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.category
ORDER BY revenue_percentage DESC;
```



What is the cumulative revenue over time?

```
    select order_date,round(sum(revenue) over (order by
order_date),2) as cumulative_revenue from
(SELECT
    o.order_date, SUM(od.quantity * p.price) AS revenue
FROM
    order_details od
        JOIN
    pizzas p ON od.pizza_id = p.pizza_id
        JOIN
    orders o ON o.order_id = od.order_id
GROUP BY o.order_date) as sales;
```

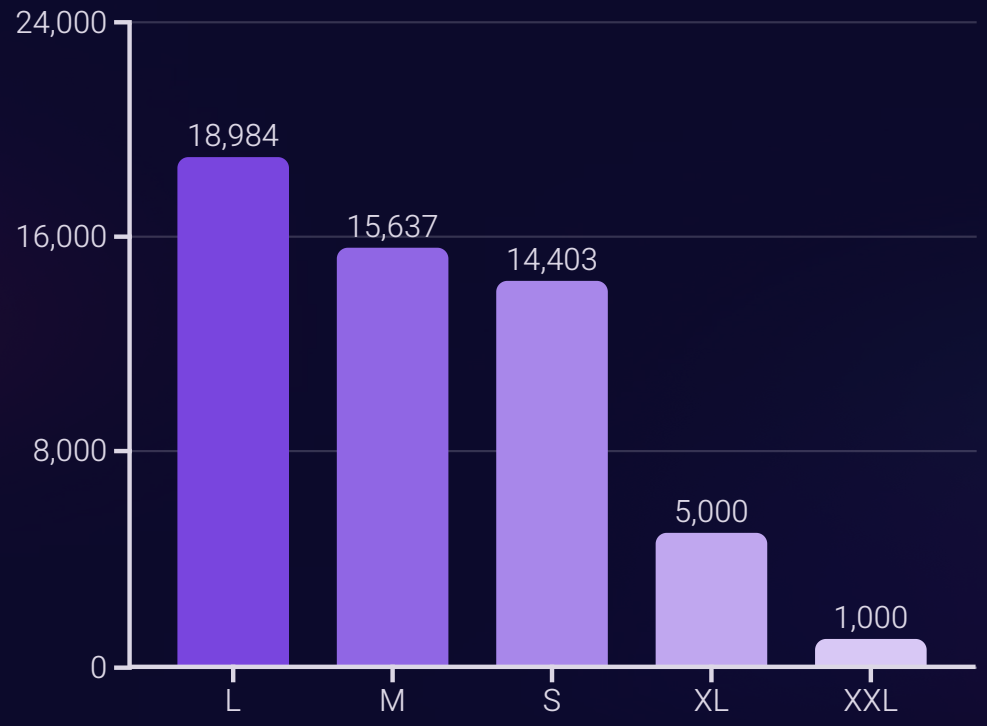


What is the total number of orders placed?

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

Which is the most common pizza size ordered?

```
SELECT
    SUBSTRING_INDEX(pizza_id, '_', - 1) AS pizza_size,
    SUM(quantity) AS total_quantity
FROM
    order_details
GROUP BY pizza_size
ORDER BY total_quantity DESC
LIMIT 1;
```

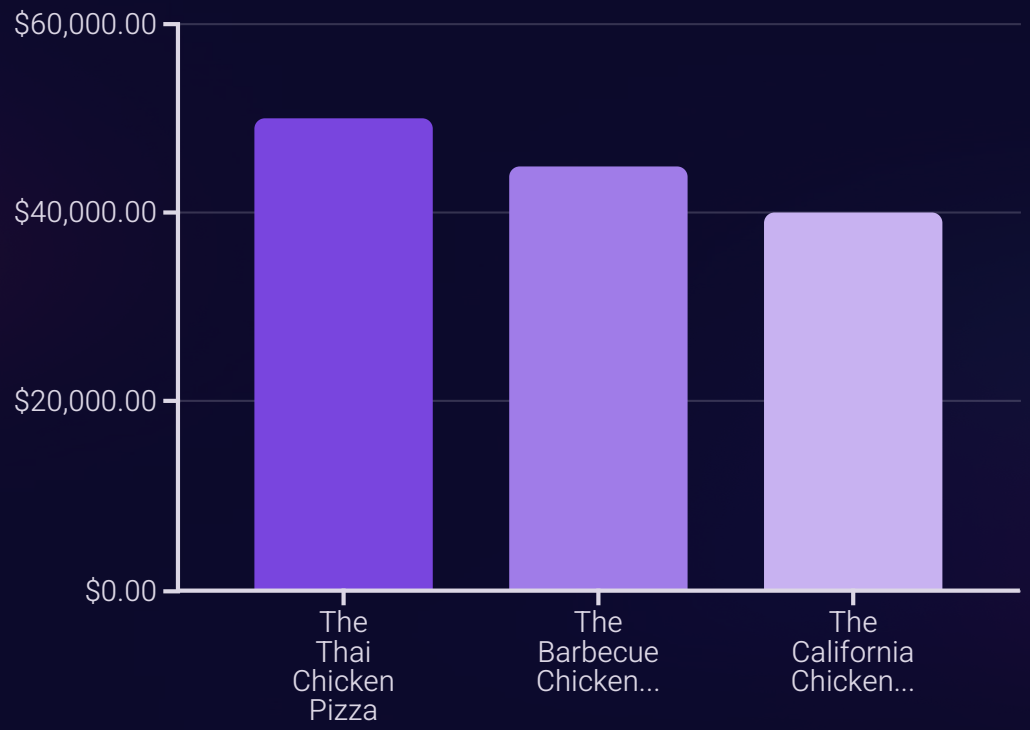


Which is the least common pizza size ordered?

```
SELECT
    SUBSTRING_INDEX(pizza_id, '_', - 1) AS Pizza_size,
    SUM(quantity) AS total_quantity
FROM
    order_details
GROUP BY pizza_size
ORDER BY total_quantity
LIMIT 1;
```

What are the top 3 most ordered pizza types by revenue?

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



What are the top 3 most ordered pizza types based on revenue per category?

```
select name, revenue from
(Select category,name,revenue,rank() over (partition by category order by revenue desc) as
rn from (SELECT
    pt.category, pt.name, SUM(od.quantity * p.price) AS Revenue
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.category , pt.name) as A) as B where rn<=3;
```

This query identifies the top 3 pizza types by revenue within each category. The results are presented below, showcasing the highest-earning pizzas across different categories.

Chicken	The Thai Chicken Pizza	50000
Chicken	The Barbecue Chicken Pizza	45000
Chicken	The California Chicken Pizza	40000
Classic	The Classic Deluxe Pizza	45000
Classic	The Pepperoni Pizza	42000
Classic	The Italian Supreme Pizza	38000
Veggie	The Mediterranean Pizza	35000
Veggie	The Garden Veggie Pizza	32000
Veggie	The Spinach and Feta Pizza	30000
Supreme	The Big Meat Pizza	48000
Supreme	The Spicy Italian Pizza	41000
Supreme	The Southwest Chicken Pizza	39000

Which is the highest-priced pizza?

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

Which is the second highest-priced pizza?

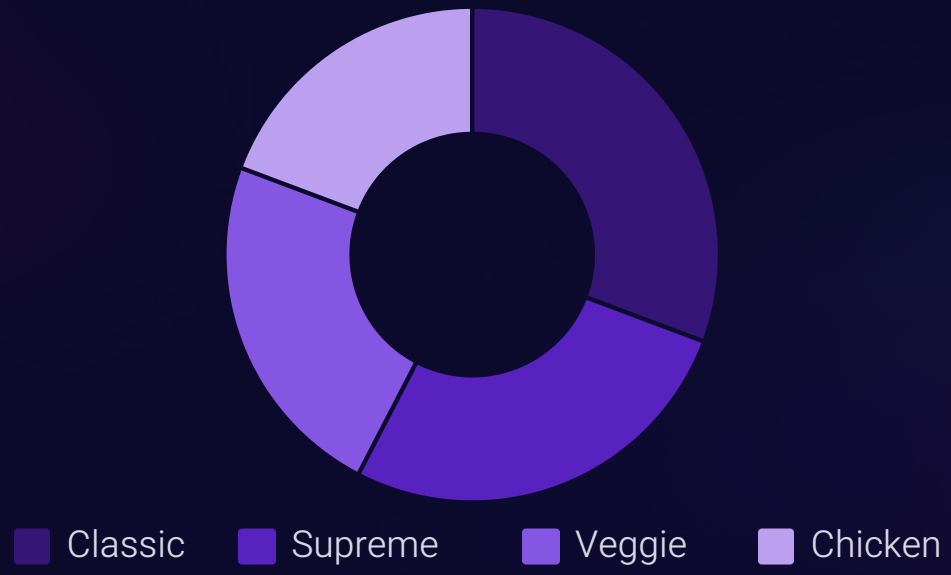
```
SELECT
    pt.name, p.price
FROM
    pizzas p
        JOIN
    pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
ORDER BY price DESC
LIMIT 1 OFFSET 1;
```

Which is the lowest-priced pizza?

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

How many pizza types are there in each category?

```
SELECT
    category,
    COUNT(pizza_type_id) AS number_of_types
FROM
    pizzahut.pizza_types
GROUP BY
    category
ORDER BY
    number_of_types DESC;
```



Which is the least common pizza size ordered?

```
SELECT
    SUBSTRING_INDEX(pizza_id, '_', - 1) AS Pizza_size,
    SUM(quantity) AS total_quantity
FROM
    order_details
GROUP BY pizza_size
ORDER BY total_quantity
LIMIT 1;
```