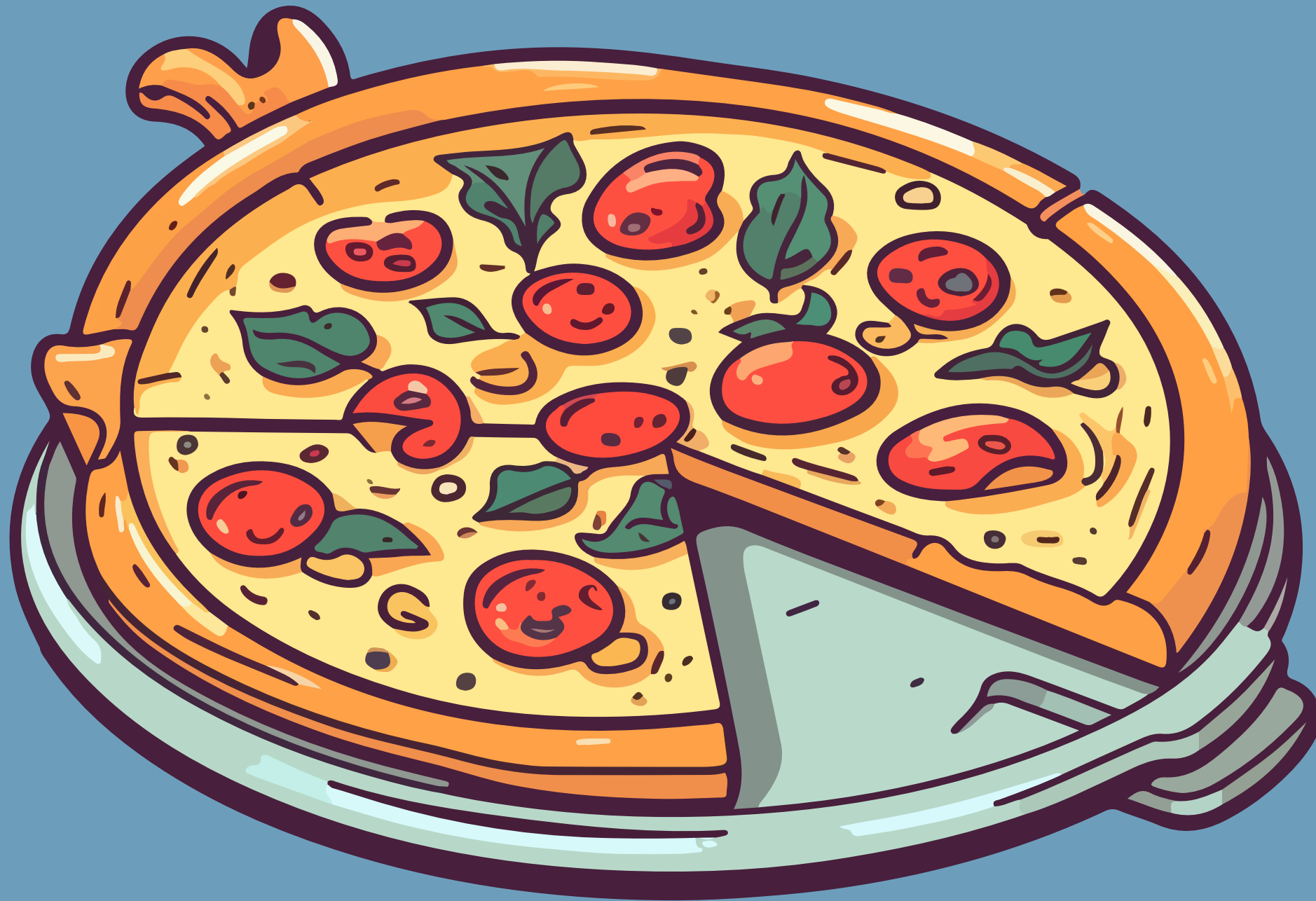
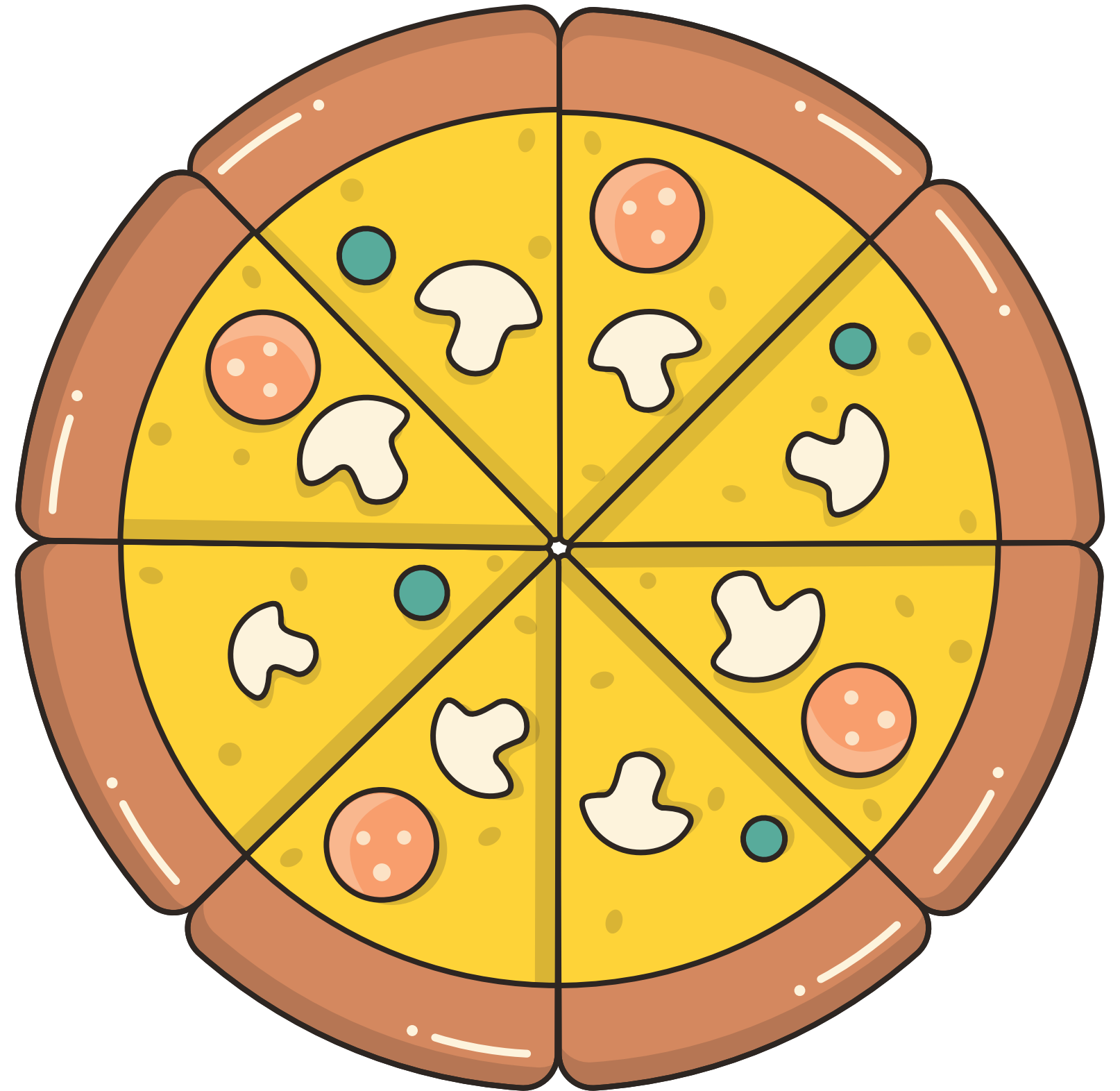


PIZZA SALES PROJECT



1

In this project, I have utilized SQL queries to solve questions related to pizza sales.



Retrieve the total number of orders placed.

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

Calculate the total revenue generated from pizza sales.

```
SELECT
    ROUND(SUM(orders_details.quantity * pizzas.price),
          2) AS total_revenue
FROM
    orders_details
    JOIN
    pizzas ON orders_details.pizza_id = pizzas.pizza_id
```

Identify the highest-priced pizza.

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

Identify the most common pizza size ordered.

```
SELECT
    pz.size, COUNT(od.quantity)
FROM
    pizzas pz
    JOIN
        orders_details AS od ON od.pizza_id = pz.pizza_id
GROUP BY pz.size
ORDER BY pz.size
```

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

```
SELECT
    pt.name, COUNT(od.quantity) AS quantity_pizza
FROM
    pizzas AS pz
    JOIN
        orders_details AS od ON od.pizza_id = pz.pizza_id
    JOIN
        pizza_types AS pt ON pt.pizza_type_id = pz.pizza_type_id
GROUP BY pt.name
ORDER BY quantity_pizza DESC
LIMIT 5
```

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

```
SELECT
    pt.category, COUNT(od.quantity) AS Total_quality
FROM
    pizza_types AS pt
    JOIN
    pizzas AS pz ON pz.pizza_type_id = pt.pizza_type_id
    JOIN
    orders_details AS od ON od.pizza_id = pz.pizza_id
GROUP BY pt.category
ORDER BY Total_quality
```


Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(order_time) AS Hour, COUNT(order_id) AS count_order
FROM
    orders
GROUP BY HOUR(order_time);
```

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

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

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

```
SELECT
    ROUND(AVG(Quantity), 0) AS avg_pizza_per_day
FROM
    (SELECT
        od.order_date, SUM(od.quantity) AS Quantity
    FROM
        orders AS od
    JOIN orders_details AS odi ON odi.order_id = od.order_id
    GROUP BY order_date) AS order_quantity;
```

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

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

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

```
SELECT
    pt.category,
    ROUND(SUM(od.quantity * pd.price) / (SELECT
        ROUND(SUM(orders_details.quantity * pizzas.price),
            2) AS total_sales
        FROM
            orders_details
            JOIN
                pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100,
        2) AS revenue
FROM
    pizza_types AS pt
    JOIN
        pizzas AS pd ON pt.pizza_type_id = pd.pizza_type_id
    JOIN
        orders_details AS od ON od.pizza_id = pd.pizza_id
GROUP BY category
ORDER BY revenue DESC;
```