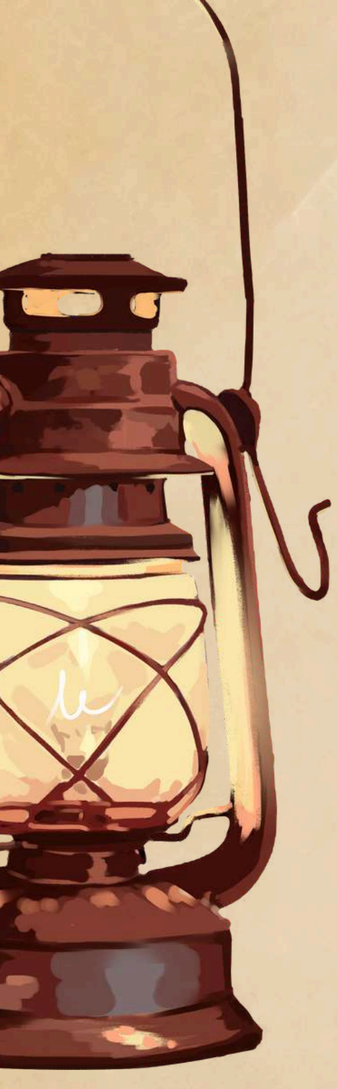
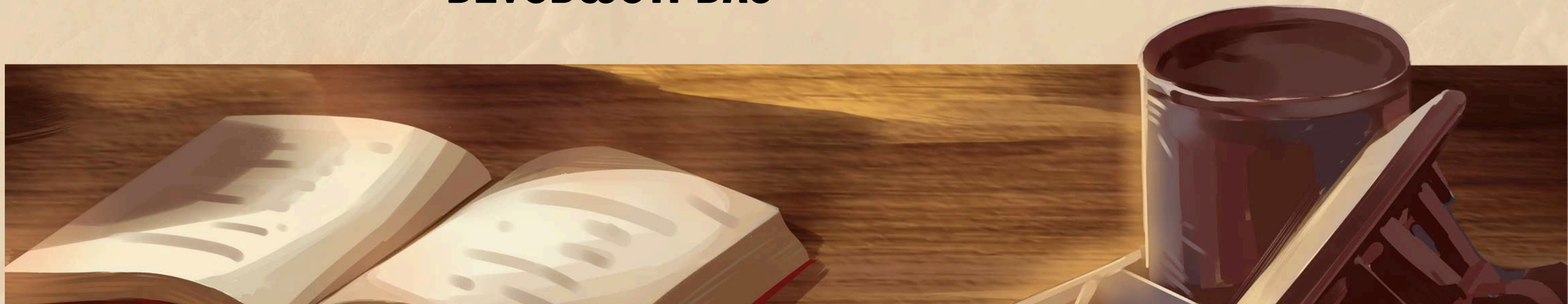


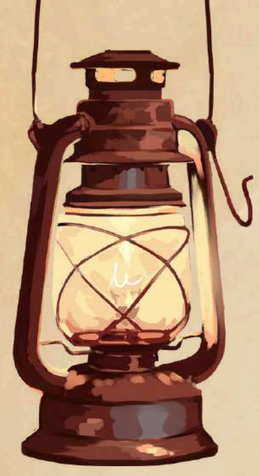
SALES REPORT

SQL Portfolio Analysis Project

by

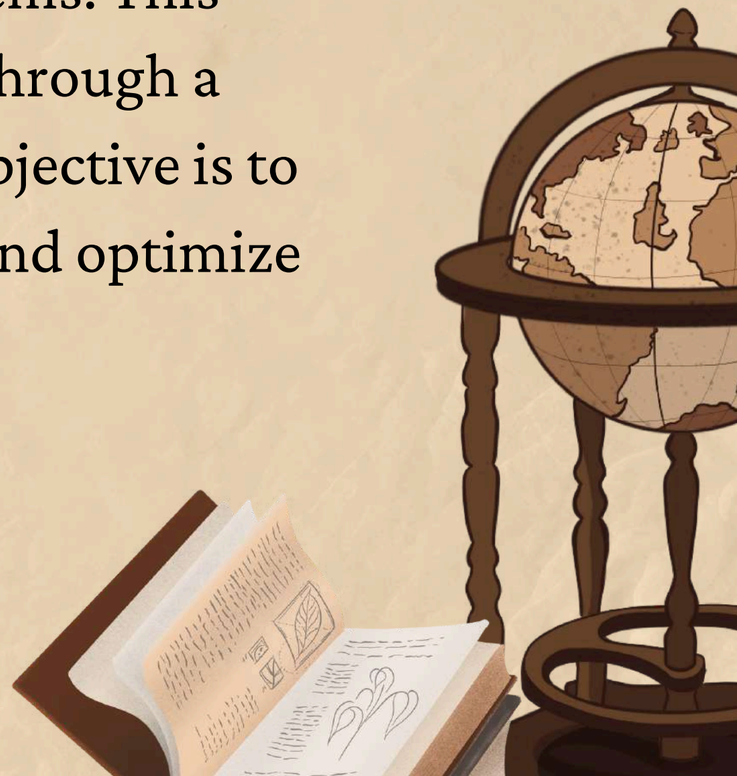
DEVODWOTI DAS





ABOUT PROJECT

Structured Query Language (SQL) remains the cornerstone of data management and manipulation in relational database systems. This SQL portfolio project demonstrates my proficiency in SQL through a comprehensive analysis of a chosen dataset. The primary objective is to analyze monthly sales trends to identify key growth areas and optimize inventory management.



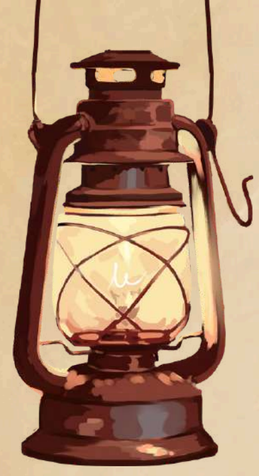


TABLE - 1

```
SELECT * FROM pizzahut.order_details;
```

Result Grid | Filter Rows: | Edi

order_detail_id	order_id	pizza_id	quantity
7	3	ital_supr_m	1
9	4	ital_supr_m	1
10	5	ital_supr_m	1
11	6	bbq_ckn_s	1
13	7	spinach_supr_s	1
14	8	spinach_supr_s	1
15	9	classic_dlx_s	1
24	10	mexicana_l	1
26	11	bbq_ckn_l	1
30	12	cali_ckn_l	1
34	13	mexicana_l	1
35	14	the_greek_s	1

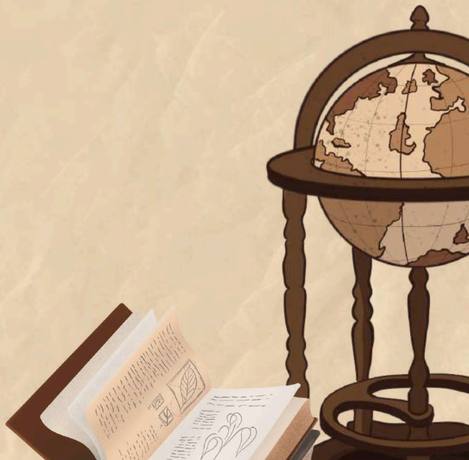
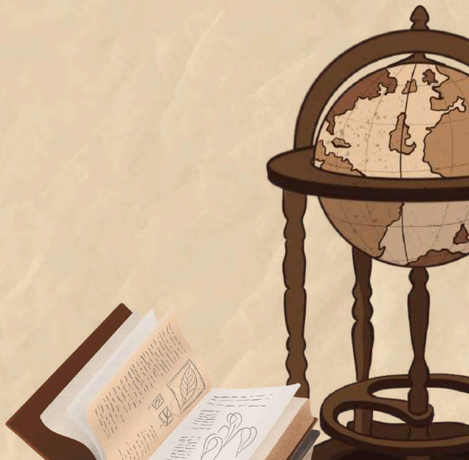


TABLE - 2

```
SELECT * FROM pizzahut.orders;
```

Result Grid | Filter Rows:

order_id	order_date	order_time
1	2015-01-01	11:38:36
2	2015-01-01	11:57:40
3	2015-01-01	12:12:28
4	2015-01-01	12:16:31
5	2015-01-01	12:21:30
6	2015-01-01	12:29:36
7	2015-01-01	12:50:37
8	2015-01-01	12:51:37
9	2015-01-01	12:52:01
10	2015-01-01	13:00:15



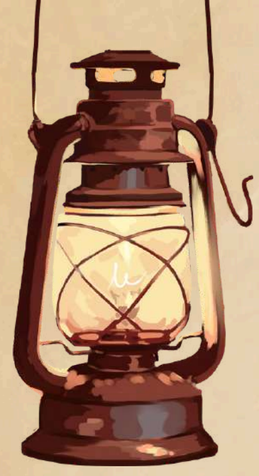
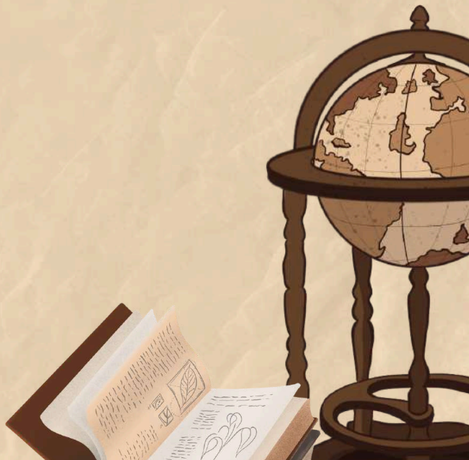


TABLE - 3

```
SELECT * FROM pizzahut.pizza_types;
```

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

pizza_type_id	name	category	ingredients
bbq_ckn	The Barbecue Chicken Pizza	Chicken	Barbecued Chicken, Red Peppers, Green Peppe...
cali_ckn	The California Chicken Pizza	Chicken	Chicken, Artichoke, Spinach, Garlic, Jalapeno P...
ckn_alfredo	The Chicken Alfredo Pizza	Chicken	Chicken, Red Onions, Red Peppers, Mushrooms...
ckn_pesto	The Chicken Pesto Pizza	Chicken	Chicken, Tomatoes, Red Peppers, Spinach, Garl...
southw_ckn	The Southwest Chicken Pizza	Chicken	Chicken, Tomatoes, Red Peppers, Red Onions, ...
thai_ckn	The Thai Chicken Pizza	Chicken	Chicken, Pineapple, Tomatoes, Red Peppers, T...
big_meat	The Big Meat Pizza	Classic	Bacon, Pepperoni, Italian Sausage, Chorizo Sau...
classic_dlx	The Classic Deluxe Pizza	Classic	Pepperoni, Mushrooms, Red Onions, Red Peppe...
hawaiian	The Hawaiian Pizza	Classic	Sliced Ham, Pineapple, Mozzarella Cheese
ital_cpclo	The Italian Capocollo Pizza	Classic	Capocollo, Red Peppers, Tomatoes, Goat Chee...
napolitana	The Napolitana Pizza	Classic	Tomatoes, Anchovies, Green Olives, Red Onion...



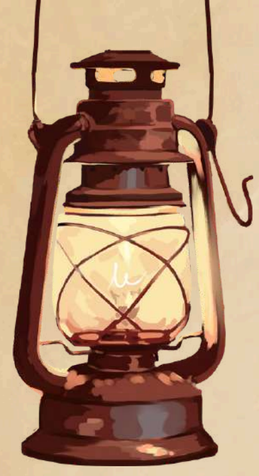
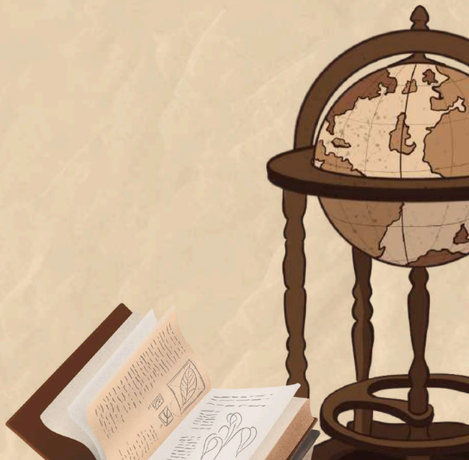


TABLE - 4

```
SELECT * FROM pizzahut.pizzas;
```

Result Grid | Filter Rows:

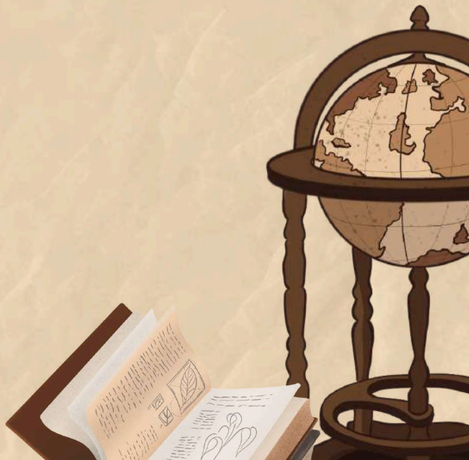
pizza_id	pizza_type_id	size	price
bbq_ckn_s	bbq_ckn	S	12.75
bbq_ckn_m	bbq_ckn	M	16.75
bbq_ckn_l	bbq_ckn	L	20.75
cali_ckn_s	cali_ckn	S	12.75
cali_ckn_m	cali_ckn	M	16.75
cali_ckn_l	cali_ckn	L	20.75
ckn_alfredo_s	ckn_alfredo	S	12.75
ckn_alfredo_m	ckn_alfredo	M	16.75
ckn_alfredo_l	ckn_alfredo	L	20.75
ckn_pesto_s	ckn_pesto	S	12.75

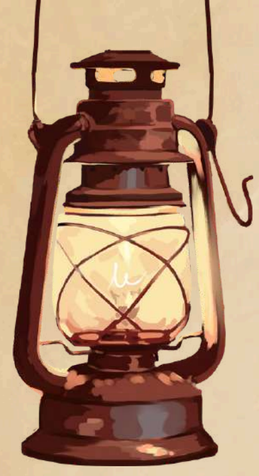


QUESTIONS

Basic:

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.

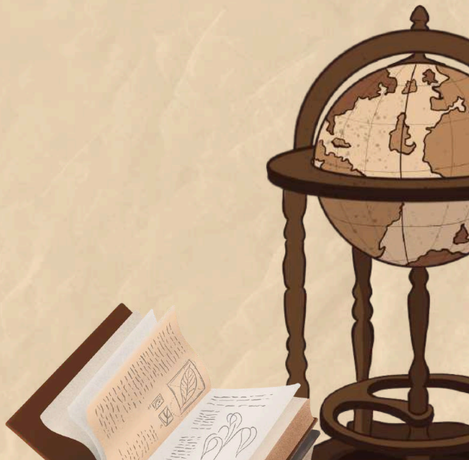


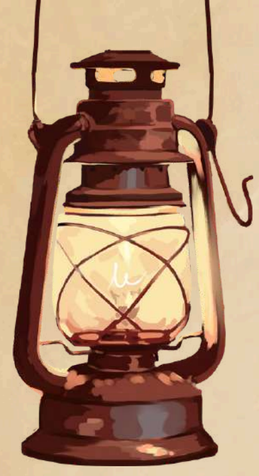


QUESTIONS

Intermediate:

- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.

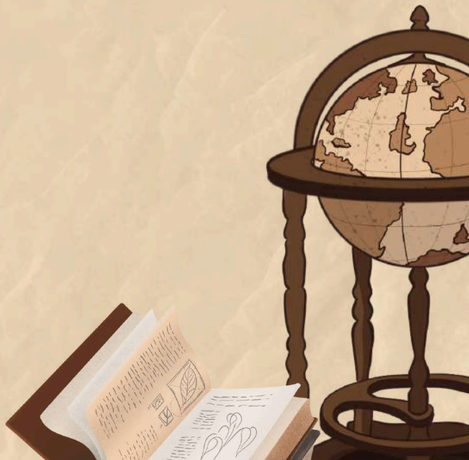




QUESTIONS

Advanced:

- Calculate the percentage contribution of each pizza type to total revenue.
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.



Retrieve the total number of orders placed

Query :

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

Result Grid

total_orders
21350

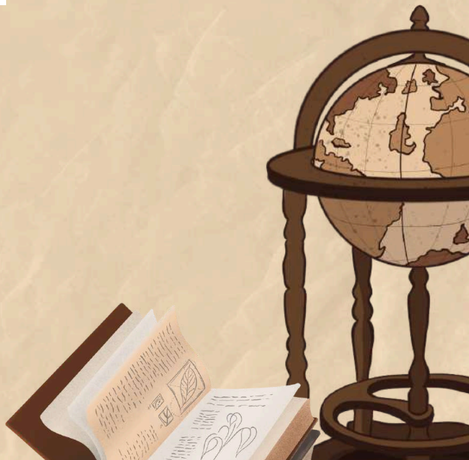
Calculate the total revenue generated from pizza sales

Query :

```
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
356723.85



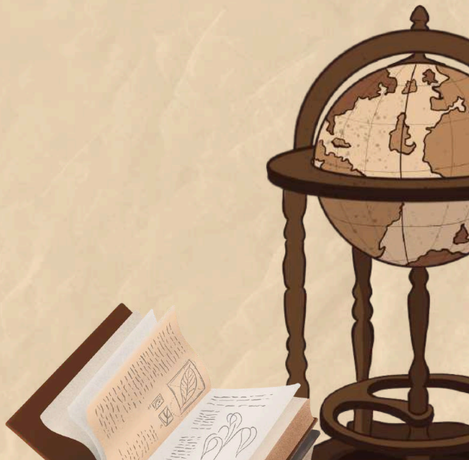
Identify the highest-priced pizza

Query :

```
SELECT
    pizza_types.name AS Gourmet_pizza, ROUND(pizzas.price,2)
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:

Gourmet_pizza	ROUND(pizzas.price,2)
The Greek Pizza	35.95



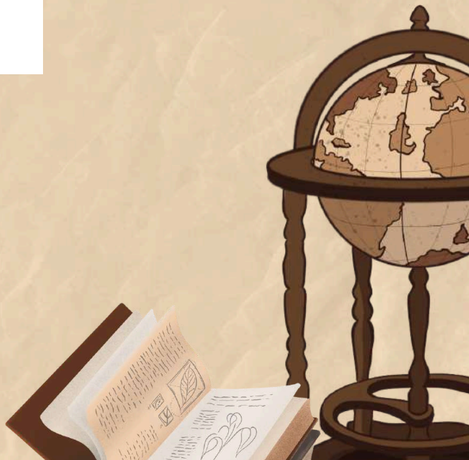
Identify the most common pizza size ordered

Query :

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

Result Grid | Filter

Size	order_count
L	7893



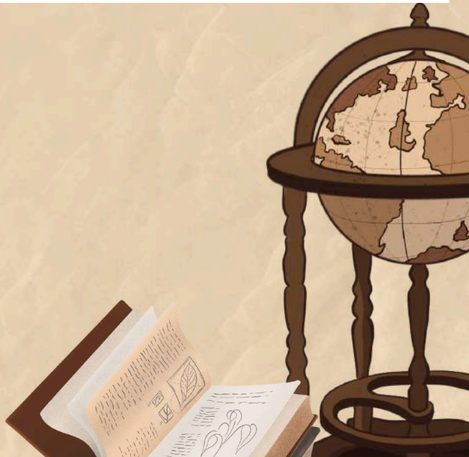
List the top 5 most ordered pizza types along with their quantities

Query :

```
SELECT
    pizza_types.name, SUM(order_details.quantity)
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

esult Grid | Filter Rows: | Exp

name	SUM(order_details.quantity
The Barbecue Chicken Pizza	2329
The California Chicken Pizza	1721
The Big Meat Pizza	1695
The Classic Deluxe Pizza	1506
The Hawaiian Pizza	1166



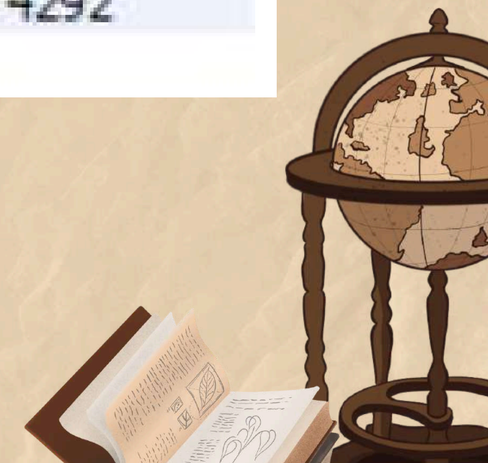
Join the necessary tables to find the total quantity of each pizza category ordered

Query :

```
SELECT
    pizza_types.category AS Category,
    COUNT(pizza_types.category) AS Quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1;
```

Result Grid | Filter

Category	Quantity
Classic	6841
Supreme	4084
Chicken	6133
Veggie	4292



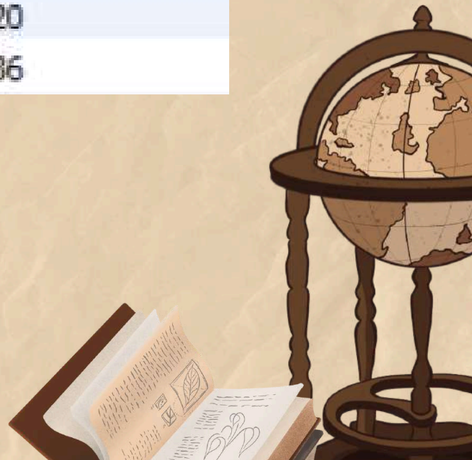
Determine the distribution of orders by hour of the day

Query :

```
SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY 1;
```

Result Grid | Filter Rows:

HOUR(order_time)	COUNT(order_id)
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336



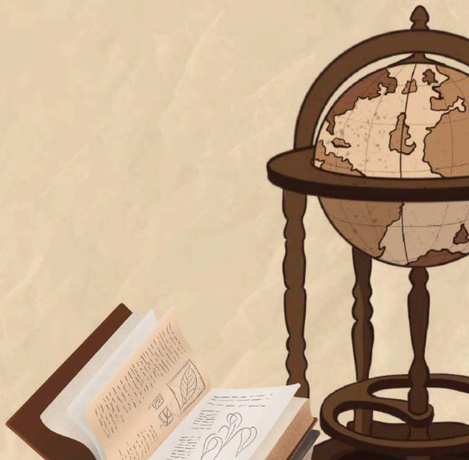
Join relevant tables to find the category-wise distribution of pizzas

Query :

```
SELECT
    category AS Category , COUNT(name)
FROM
    pizza_types
GROUP BY 1;
```

Result Grid | Filter Ro

Category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9



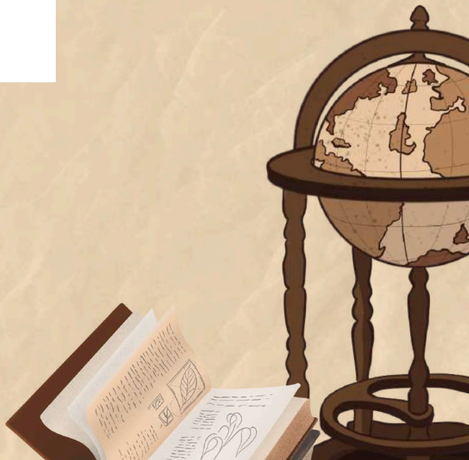
Group the orders by date and calculate the average number of pizzas ordered per day

Query :

```
SELECT
    ROUND(AVG(2),0) AS avg_no_ordered
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity)
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY 1) AS quantity;
```

Result Grid

avg_no_ordered
2



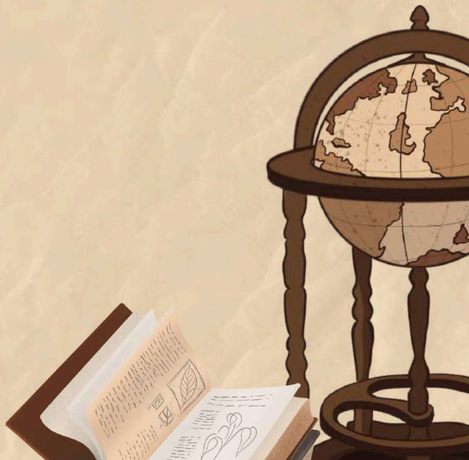
Determine the top 3 most ordered pizza types based on revenue

Query :

```sql
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),0) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 3;
```

Result Grid | Filter Rows:

name	revenue
The Barbecue Chicken Pizza	41231
The California Chicken Pizza	30103
The Classic Deluxe Pizza	23548



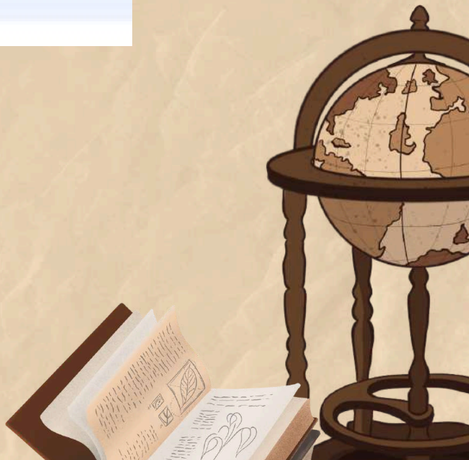
Calculate the percentage contribution of each pizza type to total revenue

Query :

```
SELECT
    pizza_types.category AS 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),0) 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 1
ORDER BY 2 DESC;
```

Result Grid | Filt

Category	Revenue
Chicken	31
Classic	28
Supreme	21
Veggie	21



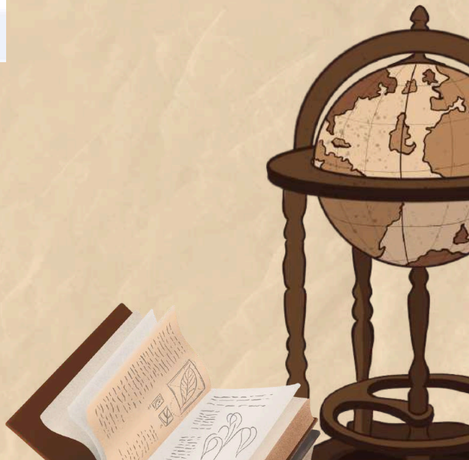
Analyze the cumulative revenue generated over time

Query :

```
SELECT
    order_date,
    ROUND(SUM(Revenue) OVER (ORDER BY order_date),2) 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) AS sales;
```

Result Grid | Filter Rov

order_date	cum_revenue
2015-01-01	1136.35
2015-01-02	2245.85
2015-01-03	3374.85
2015-01-04	4260.15
2015-01-05	5159.1
2015-01-06	6232.3



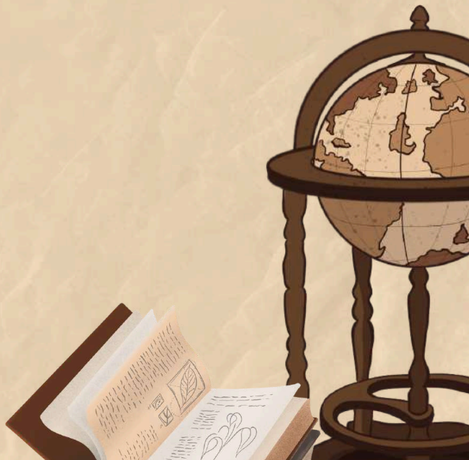
Determine the top 3 most ordered pizza types based on revenue for each pizza category

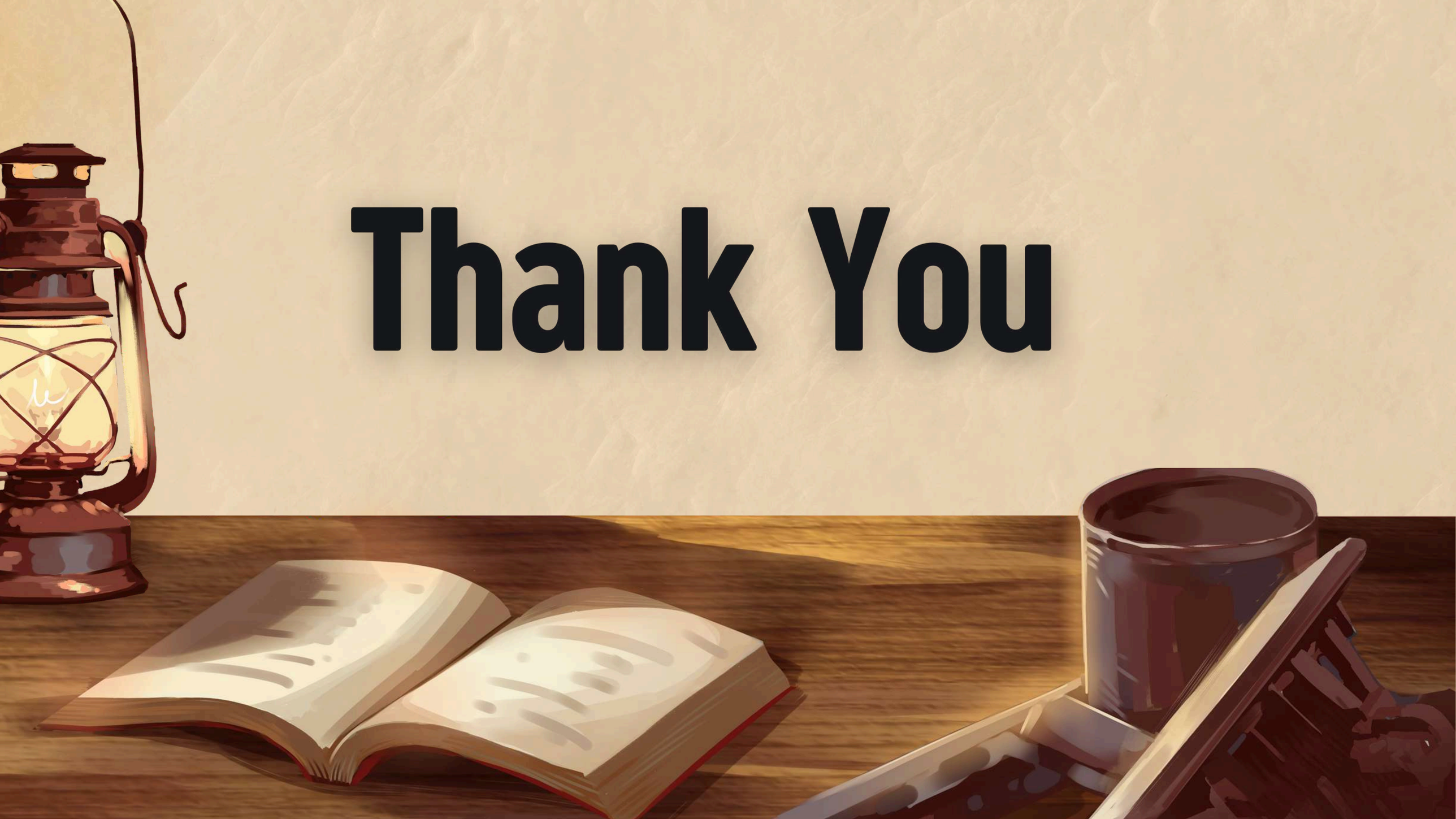
Query :

```
SELECT Name , Revenue, Category
FROM
(SELECT Category , Name , Revenue, Rank() OVER(PARTITION BY category ORDER BY Revenue DESC) AS RN
FROM
    (SELECT
        pizza_types.name,
        pizza_types.category,
        ROUND(SUM(order_details.quantity * pizzas.price),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.name , pizza_types.category) AS T1) AS T2
    WHERE RN <= 3;
```

Result Grid | Filter Rows:

Name	Revenue	Category
The Barbecue Chicken Pizza	41230.75	Chicken
The California Chicken Pizza	30102.75	Chicken
The Chicken Alfredo Pizza	11606	Chicken
The Classic Deluxe Pizza	23548	Classic
The Big Meat Pizza	20340	Classic
The Hawaiian Pizza	15546.5	Classic
The Italian Supreme Pizza	14359.75	Supreme




Thank You