

Intermediate Problem Queries



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

- `SELECT`

```
    pizza_types.category,  
    SUM(order_details.quantity) AS quantity  
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.category  
ORDER BY quantity DESC;
```

Output:

| Result Grid | | |
|-------------|----------|----------|
| | category | quantity |
| ▶ | Classic | 14888 |
| | Supreme | 11987 |
| | Veggie | 11649 |
| | Chicken | 11050 |

Question 7: Determine the distribution of orders by hour of the day.

- `SELECT`

```
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

Output:

| Result Grid | | |
|-------------|------|-------------|
| | hour | order_count |
| ▶ | 11 | 1231 |
| | 12 | 2520 |
| | 13 | 2455 |
| | 14 | 1472 |
| | 15 | 1468 |
| | 16 | 1920 |

Intermediate Problem Queries



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

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

Output:

| Result Grid | | |
|-------------|----------|-------------|
| | category | count(name) |
| ▶ | Chicken | 6 |
| | Classic | 8 |
| | Supreme | 9 |
| | Veggie | 9 |

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

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

Output:

| Result Grid | |
|-------------|---------------------------|
| | avg_pizza_ordered_per_day |
| ▶ | 138 |

Intermediate Problem Queries



Question 10: Join relevant tables to find the category-wise distribution of pizzas.

- **SELECT**

```
    pizza_types.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_types.name  
ORDER BY revenue DESC  
LIMIT 3;
```

Output:

| Result Grid | | |
|-------------|------------------------------|----------|
| | name | revenue |
| ▶ | The Thai Chicken Pizza | 43434.25 |
| | The Barbecue Chicken Pizza | 42768 |
| | The California Chicken Pizza | 41409.5 |