

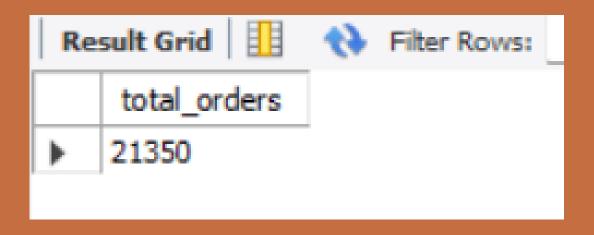
Project: Pizza Sale Analysis

Using SQL queries, the Pizza Sale Analysis project delves into data from four comprehensive datasets. The goal is to reveal insights on pizza sales, customer preferences, and ordering patterns. The datasets include:

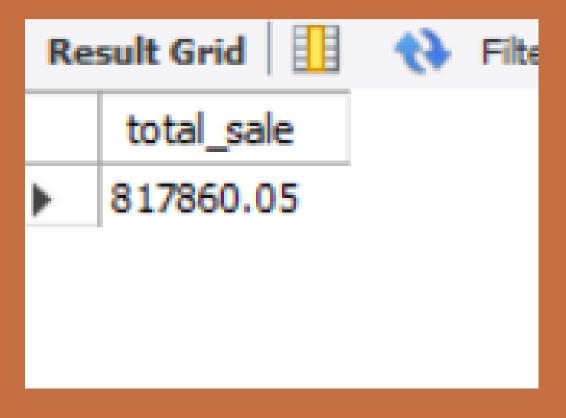
- Pizza: Detailed information on pizza id, size and price.
- Pizza Types: May consist of pizza id, category and ingredients.
- Orders: Records of orders placed, organized by date and time.
- Orders Details: Specifics on the quantities of each pizza type ordered in each transaction.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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



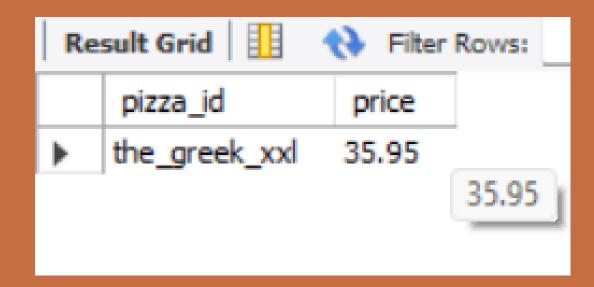
CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.



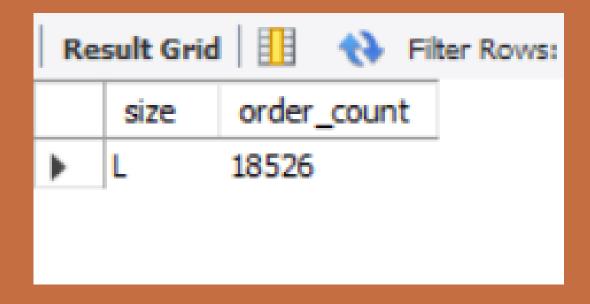
IDENTIFY THE HIGHEST-PRICED PIZZA.

```
pizza_id, price
FROM

pizzas
ORDER BY price DESC
LIMIT 1;
```



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.



LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
pizza_types.name, 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.name

ORDER BY quantity DESC

LIMIT 5;
```

| Result Grid | | | | |
|-------------|----------------------------|----------|--|--|
| | name | quantity | | |
| • | The Classic Deluxe Pizza | 2453 | | |
| | The Barbecue Chicken Pizza | 2432 | | |
| | The Hawaiian Pizza | 2422 | | |
| | The Pepperoni Pizza | 2418 | | |
| | The Thai Chicken Pizza | 2371 | | |
| | | | | |

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;
```

| 1 | | | |
|----------------|------|--|--|
| category quant | tity | | |
| Classic 14888 | | | |
| Supreme 11987 | | | |
| Veggie 11649 | | | |
| Chicken 11050 | | | |
| | | | |
| | | | |
| | | | |

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;
```

| | | 44 | Hilter | Rows |
|------|--|---|---|---|
| hour | order | _coun | t | |
| 11 | 1231 | | | |
| 12 | 2520 | | | |
| 13 | 2455 | | | |
| 14 | 1472 | | | |
| 15 | 1468 | | | |
| 16 | 1920 | | | |
| 17 | 2336 | | | |
| 18 | 2399 | | | |
| 19 | 2009 | | | |
| 20 | 1642 | | | |
| 21 | 1198 | | | |
| 22 | 663 | | | |
| 23 | 28 | | | |
| 10 | 8 | | | |
| 9 | 1 | | | |
| | | | | |
| | | | | |
| | hour 11 12 13 14 15 16 17 18 19 20 21 22 23 10 | hour order_ 11 1231 12 2520 13 2455 14 1472 15 1468 16 1920 17 2336 18 2399 19 2009 20 1642 21 1198 22 663 23 28 10 8 | hour order_coun 11 1231 12 2520 13 2455 14 1472 15 1468 16 1920 17 2336 18 2399 19 2009 20 1642 21 1198 22 663 23 28 10 8 | 12 2520 13 2455 14 1472 15 1468 16 1920 17 2336 18 2399 19 2009 20 1642 21 1198 22 663 23 28 10 8 |

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

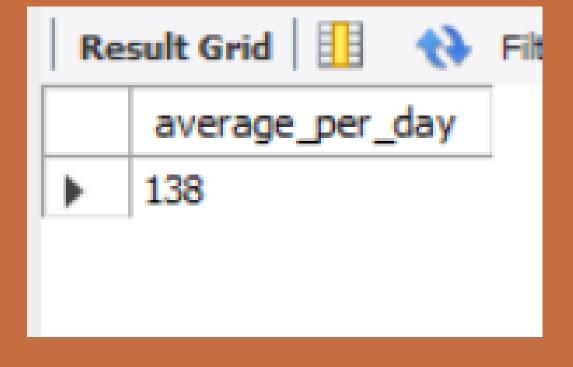
```
select category,count(pizza_type_id) as category_count
from pizza_types
group by category;
```

| Result Grid | | | | |
|-------------|----------|----------------|--|--|
| | category | category_count | | |
| • | Chicken | 6 | | |
| | Classic | 8 | | |
| | Supreme | 9 | | |
| | Veggie | 9 | | |
| | | | | |

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
SELECT
    ROUND(AVG(quantity), 0) AS average_per_day
FROM

(SELECT
    orders.order_date AS date,
        SUM(order_details.quantity) AS quantity
FROM
    orders
JOIN order_details ON orders.order_id = order_details.order_id
GROUP BY date) AS order_per_day;
```



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
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;
```

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

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
SELECT
    pizza_types.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,
            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.category
ORDER BY revenue DESC;
```

| | category | revenue |
|---|----------|---------|
| • | Classic | 26.91 |
| | Supreme | 25.46 |
| | Chicken | 23.96 |
| | Veggie | 23.68 |

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

| Result Grid | | 44 | Filter | Rows: |
|-------------|--|----|--------|-------|
|-------------|--|----|--------|-------|

| | order_date | cum_revenue |
|----------|------------|--------------------|
| • | 2015-01-01 | 2713.8500000000004 |
| | 2015-01-02 | 5445.75 |
| | 2015-01-03 | 8108.15 |
| | 2015-01-04 | 9863.6 |
| | 2015-01-05 | 11929.55 |
| | 2015-01-06 | 14358.5 |
| | 2015-01-07 | 16560.7 |
| | 2015-01-08 | 19399.05 |
| | 2015-01-09 | 21526.4 |
| | 2015-01-10 | 23990.350000000002 |
| | 2015-01-11 | 25862.65 |
| | 2015-01-12 | 27781.7 |
| | 2015-01-13 | 29831.300000000003 |
| | 2015-01-14 | 32358.700000000004 |
| | 2015-01-15 | 34343.50000000001 |

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
select name, revenue from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as rnk
from
(select pizza_types.category, pizza_types.name,
sum((order_details.quantity)*pizzas.price) 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.category, pizza_types.name) as a)as b
where rnk<=3;</pre>
```

| Re | sult Grid 🔢 🙌 Filter Row | /5: | Export: | |
|----|------------------------------|-------------------|---------|--|
| | name | revenue | | |
| • | The Thai Chicken Pizza | 43434.25 | | |
| | The Barbecue Chicken Pizza | 42768 | | |
| | The California Chicken Pizza | 41409.5 | | |
| | The Classic Deluxe Pizza | 38180.5 | | |
| | The Hawaiian Pizza | 32273.25 | | |
| | The Pepperoni Pizza | 30161.75 | | |
| | The Spicy Italian Pizza | 34831.25 | | |
| | The Italian Supreme Pizza | 33476.75 | | |
| | The Sicilian Pizza | 30940.5 | | |
| | The Four Cheese Pizza | 32265.70000000065 | | |
| | The Mexicana Pizza | 26780.75 | | |
| | The Five Cheese Pizza | 26066.5 | | |