MOTO PIZZA

Sales Analysis by
Sundram Tiwari

SOLVE NOW

Mysql

QUESTION - QUERY

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
select count(order_id) from orders;
```

Result Grid

	count(order_id)
▶	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
    ROUND(SUM(order_details.QUANTITY * pizzas.PRICE),
            2) AS total_revenue
FROM
    ORDER_DETAILS
        JOIN
    PIZZAS ON order_details.PIZZA_ID = pizzas.PIZZA_ID;
```

Result Grid

	total_revenue
▶	817860.05

IDENTIFY THE HIGHEST PRICED PIZZA.

```sql
SELECT pizza_types.name, pizzas.price
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

name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```sql
SELECT pizzas.size, COUNT(order_details.order_details_id) AS order_count
FROM pizzas
JOIN order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid | Filter Rc

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

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

```
3
4 •  select pizza_types.name,
5    sum(order_details.quantity) as quantity
6    from pizza_types join pizzas
7    on pizza_types.pizza_type_id = pizzas.pizza_type_id
8    join order_details
9    on order_details.PIZZA_ID = pizzas.pizza_id
10   group by pizza_types.name
11   order by quantity desc limit 5;
```

Result Grid | Filter Rows:

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.

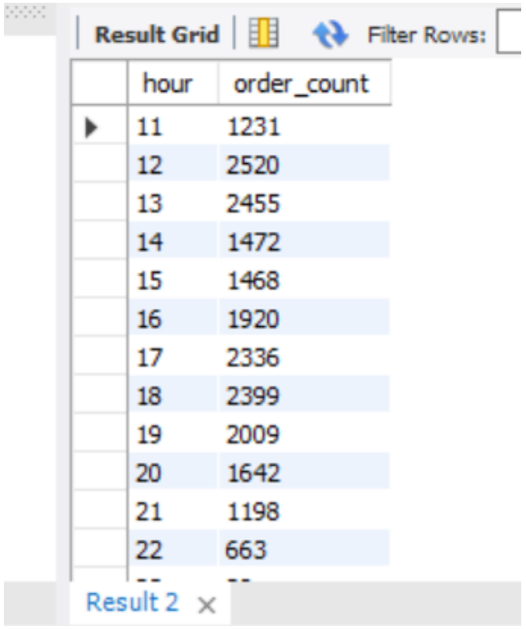
```sql
SELECT pizza_types.category,
       SUM(order_details.quantity) AS quantity
FROM pizza_types
JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
JOIN order_details ON order_details.PIZZA_ID = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

Result Grid | Filter Row

category	quantity
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(order_time);
  ```



Result Grid | Filter Rows:

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663

Result 2
  ```

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

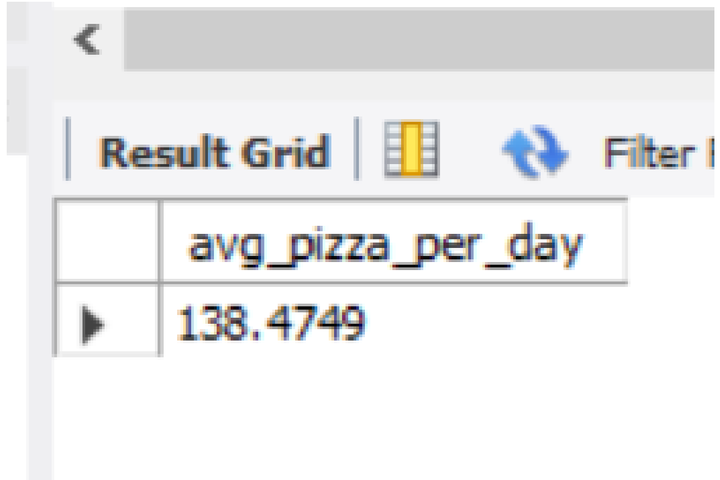
```
select category , count(name) from pizza_types
group by category;
```

Result Grid | Filter Rows:

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

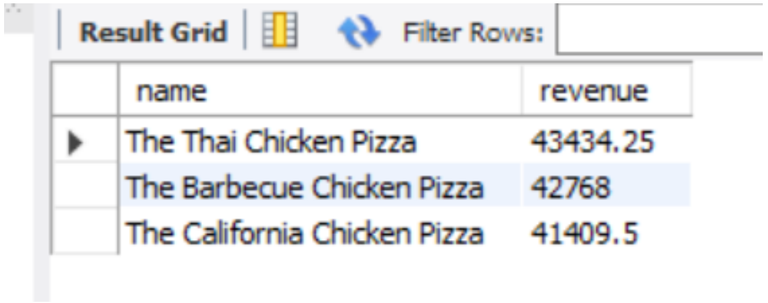
GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
select avg(quantity) as avg_pizza_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;
```



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

```sql
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;
```

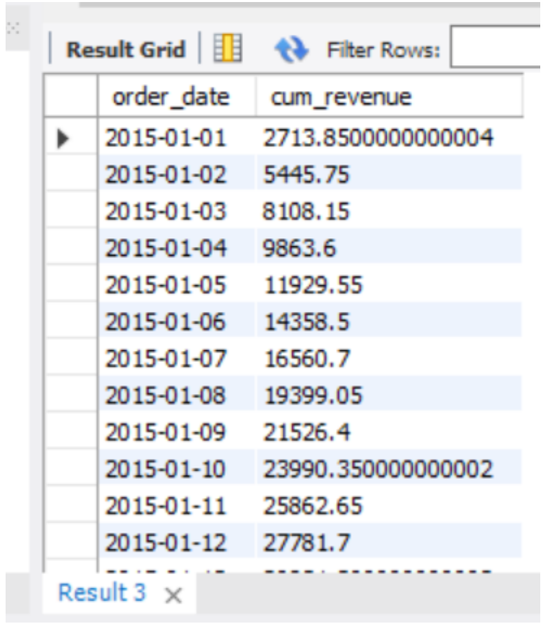


Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```sql
SELECT
  order_date,
  SUM(revenue) OVER (ORDER BY order_date) 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 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

Result 3