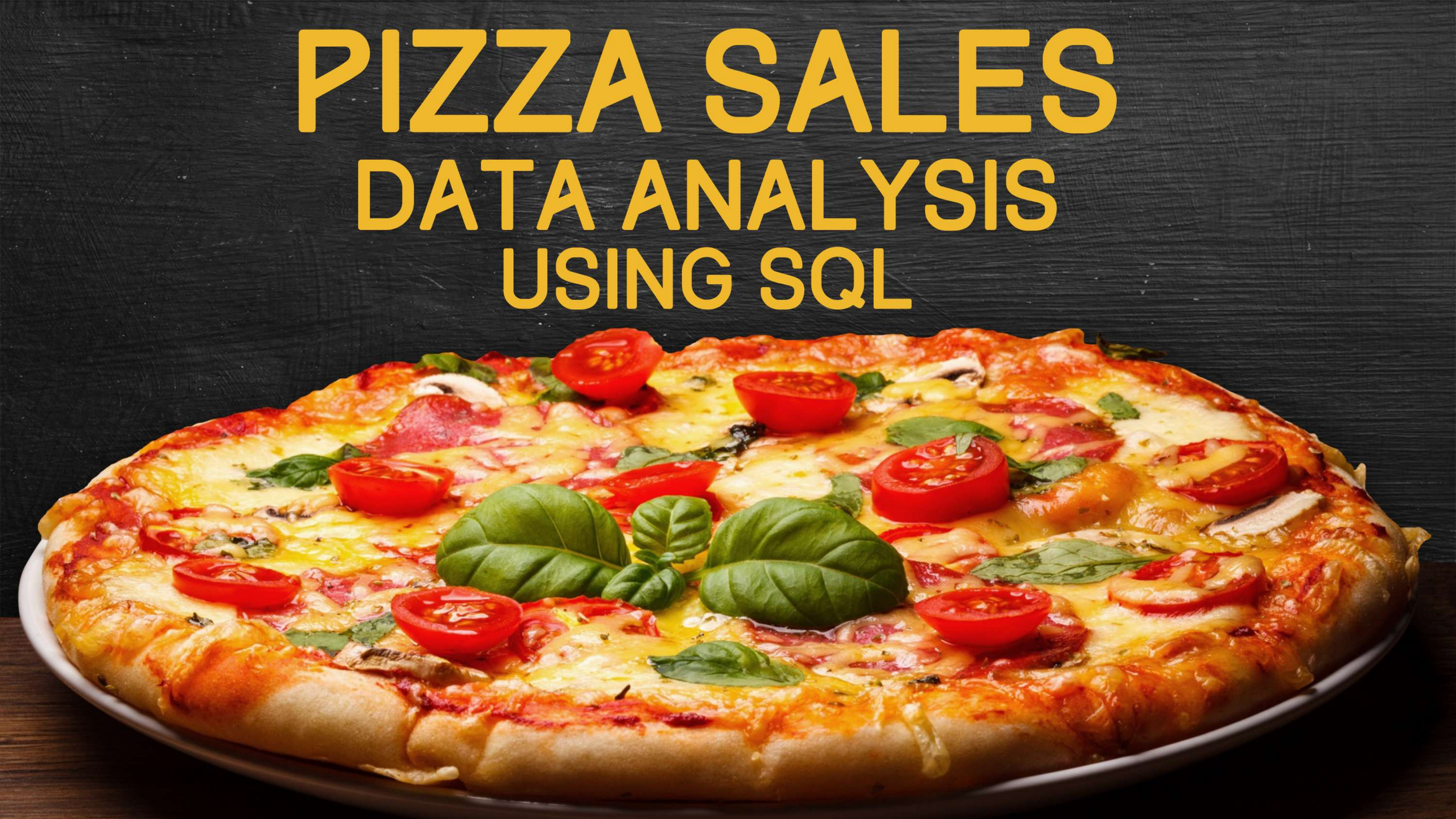
# PIZZA SALES
## DATA ANALYSIS
## USING SQL



# 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.

# 1 RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

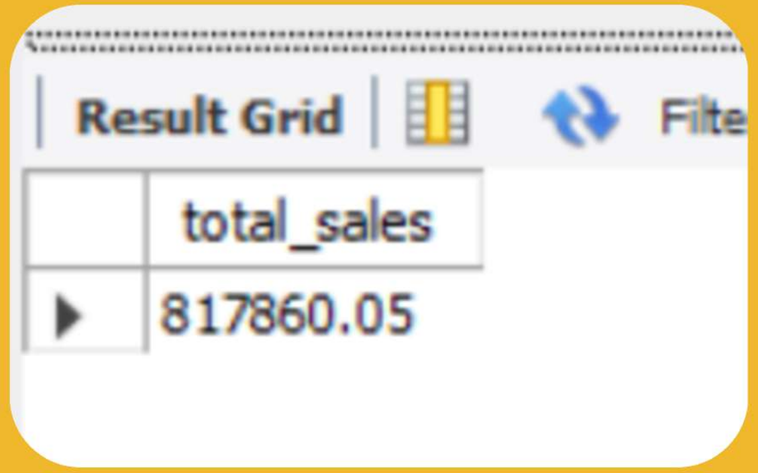
```sql
select count(order_id) as total_orders from orders;
```

Result Grid | Filter Rows:

| total_orders |
| --- |
| 42700 |

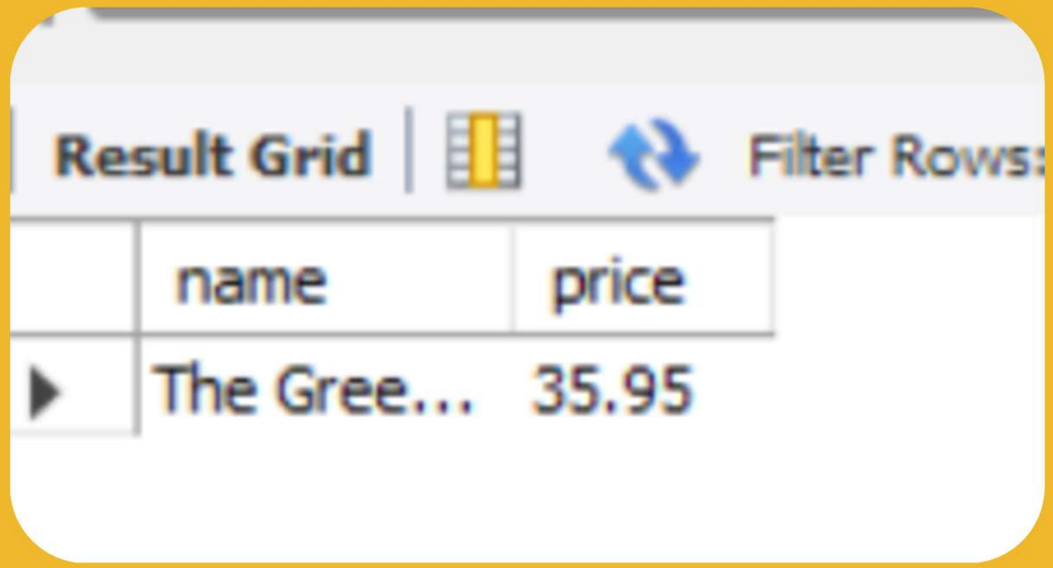
# 2 CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```sql
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
```



# 3 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;
```



# 4 IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```sql
select pizzas.size, count(orders_details.order_details_id) as order_count
from pizzas join orders_details
on pizzas.pizza_id = orders_details.pizza_id
group by pizzas.size order by order_count  desc;
```

Result Grid | Filter Rows:

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

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

```
select pizza_types.name,
sum(orders_details.quantity) as quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.name order by quantity desc limit 5;
```

Result Grid | Filter Rows:

| name | quantity |
|---|---|
| The Clas... | 2453 |
| The Barb... | 2432 |
| The Haw... | 2422 |
| The Pepp... | 2418 |
| The Thai ... | 2371 |

# 6 JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

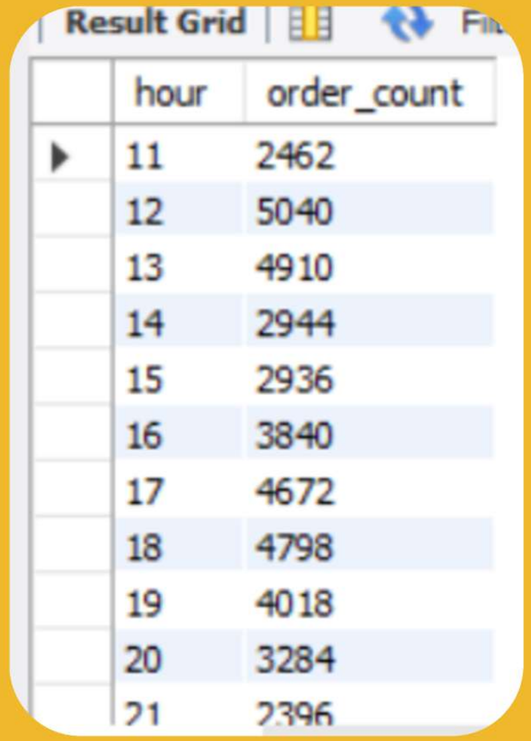
```sql
select pizza_types.category,
sum(orders_details.quantity) as quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.category order by quantity desc;
```

Result Grid | Filter Rows:

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

# 7 DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
select hour(order_time) as hour, count(order_id) as order_count
from orders
group by hour(order_time);
```



Result Grid

| hour | order_count |
|---|---|
| 11 | 2462 |
| 12 | 5040 |
| 13 | 4910 |
| 14 | 2944 |
| 15 | 2936 |
| 16 | 3840 |
| 17 | 4672 |
| 18 | 4798 |
| 19 | 4018 |
| 20 | 3284 |
| 21 | 2396 |

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

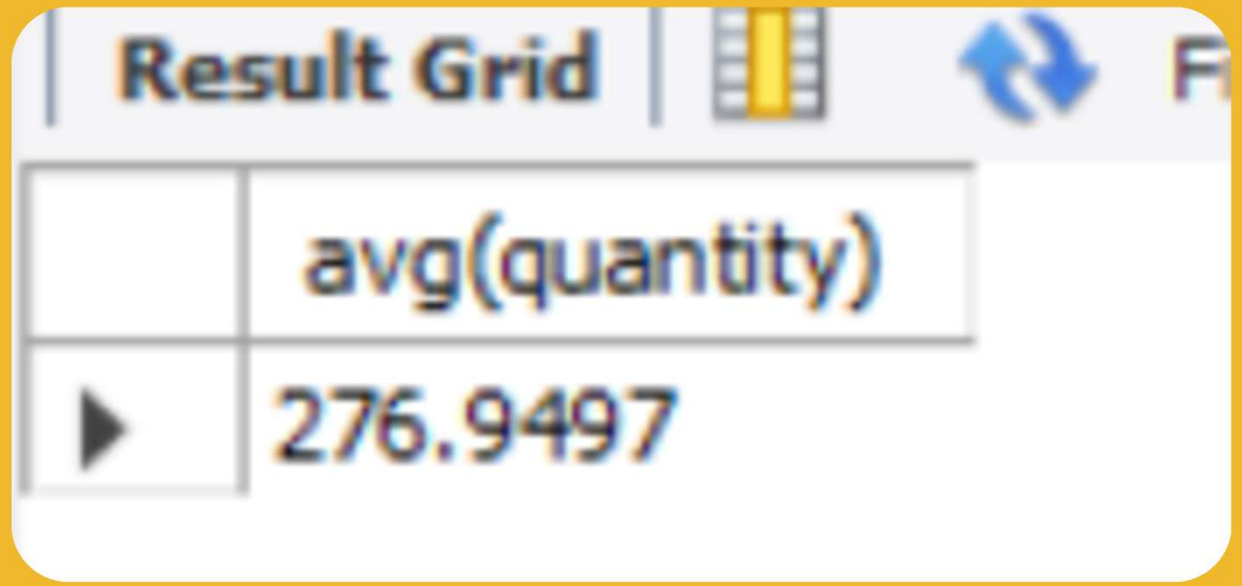
```
select category , count(name) from pizza_types
group by category ;
```

Result Grid | Filter Rows:

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

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

```
select avg(quantity) from
(select orders.order_date, sum(orders_details.quantity) as quantity
from orders join orders_details
on orders.order_id = orders_details.order_id
group by orders.order_date) as order_quantity;
```



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

```
select pizza_types.name,
sum(orders_details.quantity * pizzas.price) as revenue
from pizza_types join pizzas
on pizzas.pizza_type_id = pizza_types.pizza_type_id
join orders_details
on orders_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 ... | 43434.25 |
| The Barb... | 42768 |
| The Calif... | 41409.5 |

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

```
select pizza_types.category,
(sum(orders_details.quantity*pizzas.price) / (select
ROUND(SUM(orders_details.quantity*pizzas.price),
2) as total_sales
from
orders_details
join
pizzas on pizzas.pizza_type_id = orders_details.pizza_id) ) *100 as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.category order by revenue desc;
```

| category | revenue |
|---|---|
| Classic | NULL |
| Veggie | NULL |
| Supreme | NULL |
| Chicken | NULL |

# 12 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(orders_details.quantity*pizzas.price) as revenue
from orders_details join pizzas
on orders_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = orders_details.order_id
group by orders.order_date) as sales;
```

| order_date | cum_revenue |
|---|---|
| 2015-01-01 | 5427.7 |
| 2015-01-02 | 10891.5 |
| 2015-01-03 | 16216.3 |
| 2015-01-04 | 19727.2 |
| 2015-01-05 | 23859.1 |
| 2015-01-06 | 28717 |
| 2015-01-07 | 33121.4 |
| 2015-01-08 | 38798.1 |
| 2015-01-09 | 43052.799999999996 |
| 2015-01-10 | 47980.7 |

# 13 DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGO

```sql
select name,revenue from
(select category,name,revenue,
rank() over(partition by category order by revenue desc) as rn
from
(select pizza_types.category, pizza_types.name,
sum((orders_details.quantity) * pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where rn<=3;
```

| name | revenue |
|---|---|
| The Thai ... | 43434.25 |
| The Barb... | 42768 |
| The Calif... | 41409.5 |
| The Clas... | 38180.5 |
| The Haw... | 32273.25 |
| The Pepp... | 30161.75 |
| The Spic... | 34831.25 |
| The Italia... | 33476.75 |
| The Sicili... | 30940.5 |
| The Four... | 32265.70000000065 |
| The Mexi... | 26780.75 |

# CREATED BY :

# RISHAV RAJ

# THANK YOU!