

SPECIAL PIZZA

Life is short, eat the pizza!

Project !



Hello Everyone!

Hello!

**Hi , my name is sakil middya.
In this project I utilized the query.
That are related to pizza sales.**

-retrieve the total number of orders placed.

**select count (order_id)
as total_orders
from orders**

--analyze the cumulative revenue generated over time.

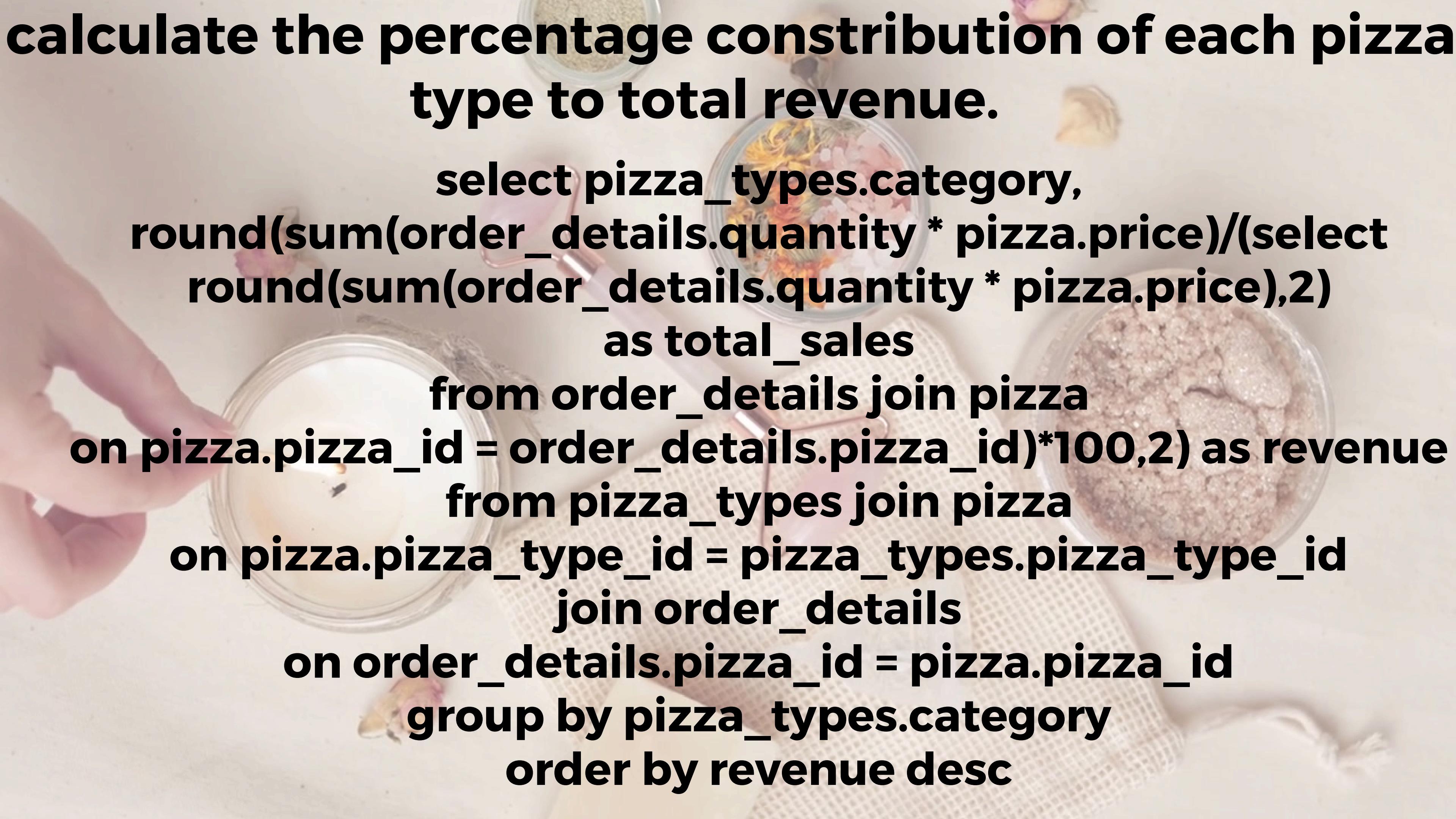
```
select order_date ,  
sum(revenue)over (order by order_date) as cumulative_revenue  
from  
(select orders.order_date ,  
sum(order_details.quantity * pizza.price) as revenue  
from order_details join pizza  
on order_details.pizza_id = pizza.pizza_id  
join orders  
on orders.order_id = order_details.order_id  
group by orders.order_date) as sales
```

--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 rn
from
(select pizza_types.category , pizza_types.name,
sum(order_details.quantity * pizza.price) as revenue
from pizza_types join pizza
on pizza_types.pizza_type_id = pizza.pizza_type_id
join order_details
on order_details.pizza_id = pizza.pizza_id
group by pizza_types.category , pizza_types.name)as a)as b
where rn <= 3
```

-- Calculate the total Revenue generated from pizza sales.

```
select  
round(sum(order_details.quantity *  
pizza.price),2) as total_sales  
from order_details join pizza  
on pizza.pizza_id = order_details.pizza_id
```



calculate the percentage contribution of each pizza type to total revenue.

```
select pizza_types.category,  
round(sum(order_details.quantity * pizza.price)/(select  
round(sum(order_details.quantity * pizza.price),2)  
as total_sales  
from order_details join pizza  
on pizza.pizza_id = order_details.pizza_id)*100,2) as revenue  
from pizza_types join pizza  
on pizza.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
on order_details.pizza_id = pizza.pizza_id  
group by pizza_types.category  
order by revenue desc
```

--determine top 3 most ordered pizza types based on revenue.

```
select top 3 pizza_types.name,  
sum(order_details.quantity * pizza.price) as revenue  
from pizza_types join pizza  
on pizza.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
on order_details.pizza_id = pizza.pizza_id  
group by pizza_types.name  
order by revenue desc
```

-group the orders by date and calculate the average
-- number of pizza ordered per day

```
select avg(quantity)as average_quantity 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
```

-- join relevant tables to find the
-- category_wise distribution of pizza.

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

-- 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 pizza  
on pizza_types.pizza_type_id = pizza.pizza_type_id  
join order_details  
on order_details.pizza_id = pizza.pizza_id  
group by pizza_types.category  
order by quantity desc
```

-- list the top 5 most ordered pizza types
-- along with their quantities

```
select top 5 pizza_types.name,  
sum(order_details.quantity) as quantity  
from pizza_types join pizza  
on pizza_types.pizza_type_id = pizza.pizza_type_id  
join order_details  
on order_details.pizza_id = pizza.pizza_id  
group by pizza_types.name  
order by quantity desc
```

--Identify the most common pizza size ordered .

```
select pizza.size ,  
count(order_details.order_details_id)as  
order_counts  
from pizza join order_details  
on pizza.pizza_id = order_details.pizza_id  
group by pizza.size  
order by order_counts desc
```

--Identify the highest price pizza.

```
select pizza_types.name ,  
      pizza.price  
from pizza_types join pizza  
on pizza_types.pizza_type_id =  
      pizza.pizza_type_id  
order by pizza.price desc
```

Thank You