



Calculate the percentage contribution of each Pizza type to total revenue

```
> (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) * 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 rn <= 3;
```

Result Grid	
category	revenue
Classic	220053.1000000001
Supreme	208196.99999999822
Chicken	195919.5
Veggie	193690.45000000298



Home

About

Contact





Retrieve the total sales

[Home](#)[About](#)[Contact](#)**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
```

RETURNS

Result Grid	
	total_sales
▶	817860.05

Analyze the cumulative revenue generated over time

```
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;
```

	order_date	revenue
1	2015-01-01	2713.8500000000004
	2015-01-02	2731.8999999999996
	2015-01-03	2662.3999999999996
	2015-01-04	1755.4500000000003
	2015-01-05	2065.95
	2015-01-06	2428.95





Retrieve the Total Numbers of Orders Placed

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

total_orders
▶ 21350



[Home](#)[About](#)[Contact](#)

Determine the top 3 most ordered pizza type based on revenue on pizza type

```
• 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) * 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 rn <= 3;
```

Result Grid | Filter

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96





Analyze the cumulative revenue generated over Time

[Home](#)[About](#)[Contact](#)

```
3 • select order_date,  
4 sum(revenue) over(order by order_date) as cum_revenue  
5 from  
6 (select orders.order_date,  
7 sum(order_details.quantity * pizzas.price) as revenue  
8 from order_details join pizzas  
9 on order_details.pizza_id = pizzas.pizza_id  
10 join orders  
11 on orders.order_id = order_details.order_id  
12 group by orders.order_date) as sales;
```

Result Grid		
	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