

DATA ANALYST - SQL PROJECT

SQL PROJECT ON PIZZA SALES

Start Your Slide

ORDER
NOW





ABOUT OUR PROJECT

FULL ANALYSIS OF PIZZA SALES

Pizza sales have shown consistent growth, driven by an increase in consumer demand for both traditional and innovative options. Revenue trends indicate strong performance, particularly in delivery and online orders.

Sales can be classified based on revenue generated by different pizza types, sizes, and services. Premium pizzas and large-sized orders contribute the most, while discounts and promotions boost customer volume.

BASIC QUESTION



- 1. Retrieve the total number of orders placed.**
- 2. Calculate the total revenue generated from pizza sales.**
- 3. Identify the highest-priced pizza.**

EASY QUESTION RELATED
TO SALES



- 1. Identify the most common pizza size ordered**
- 2. List the top 5 most ordered pizza types along with their quantities.**



QUESTION-1



-- Retrieve the total number of orders placed.--

INPUT

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

OUTPUT

Result Grid	
	total_orders
▶	21350

QUESTION-2



-- Calculate the total revenue generated from pizza sales.

INPUT

```
select round(sum(o.quantity * p.price), 2) as revenue  
from order_details as o  
join pizzas as p  
on p.pizza_id = o.pizza_id
```

OUTPUT

Result Grid	
	revenue
→	817860.05

QUESTION-3



-- Identify the highest-priced pizza.

INPUT

```
select pt.name as Costly_pizza_name, p.price as price  
from pizza_types as pt join pizzas as p  
on p.pizza_type_id = pt.pizza_type_id  
order by p.price desc limit 1;
```

OUTPUT

| Result Grid | Filter Rows:

	Costly_pizza_name	price
▶	The Greek Pizza	35.95

QUESTION-4



-- Identify the most common pizza size ordered.

INPUT

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

OUTPUT

	size	sum
▶	M	15385
	L	18526
	S	14137
	XL	544
	XXL	28

QUESTION-5



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

INPUT

```
SELECT p.pizza_type_id AS ordered_pizza,  
       SUM(o.quantity) AS quantities  
    FROM pizzas AS p JOIN  
order_details AS o ON p.pizza_id = o.pizza_id  
   GROUP BY p.pizza_type_id  
ORDER BY quantities DESC LIMIT 5;
```

OUTPUT

	ordered_pizza	quantities
1	dassic_dlx	2453
2	bbq_ckn	2432
3	hawaiian	2422
4	pepperoni	2418
5	thai_ckn	2371

INTERMEDIATE QUESTION



1. Join the necessary tables to find the total quantity of each pizza category ordered.
2. Determine the distribution of orders by hour of the day.
3. join relevant tables to find the category-wise distribution of pizzas.

MEDIUM QUESTION
RELATED TO SALES



1. Identify the most common pizza size ordered
2. List the top 5 most ordered pizza types along with their quantities.

QUESTION-6



-- Join the necessary tables to find the total quantity of each pizza category ordered.

INPUT

```
SELECT pizza_types.category AS cat,  
       SUM(order_details.quantity) AS quant  
    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.category ORDER BY quant  
        DESC;
```

OUTPUT

Result Grid				Filter Row
	cat	quant		
▶	Classic	14888		
	Supreme	11987		
	Veggie	11649		
	Chicken	11050		

QUESTION-7



Determine the distribution of orders by hour of the day.

INPUT

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

OUTPUT

hour	count(order_id)
9	1
10	8
11	1231
12	2520
13	2455

hour	count(order_id)
14	1472
15	1468
16	1920
17	2336
18	2399

hour	count(order_id)
16	1920
17	2336
18	2399
19	2009
20	1642

QUESTION-8



Determine the distribution of orders by hour of the day.

INPUT

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

OUTPUT

hour	count(order_id)
9	1
10	8
11	1231
12	2520
13	2455

hour	count(order_id)
14	1472
15	1468
16	1920
17	2336
18	2399

hour	count(order_id)
16	1920
17	2336
18	2399
19	2009
20	1642

QUESTION-9



-- Group the orders by date and calculate the average number of pizzas ordered per day.

INPUT

```
select round(avg(quantity),0) as avg_pizzas_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
```

OUTPUT

Result Grid | Filter Rows:

	avg_pizzas_per_day
138	138

QUESTION-10



Determine the top 3 most ordered pizza types based on revenue.

INPUT

```
SELECT pizza_types.name,  
SUM(pizzas.price * order_details.quantity) AS total_revenue  
FROM pizzas JOIN pizza_types ON pizzas.pizza_type_id =  
pizza_types.pizza_type_id  
JOIN order_details ON pizzas.pizza_id = order_details.pizza_id  
GROUP BY pizza_types.name ORDER BY total_revenue DESC  
LIMIT 3;
```

OUTPUT

Result Grid | Filter Rows:

	name	total_revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5

ADVANCE QUESTION



- 1. Calculate the percentage contribution of each pizza type to total revenue.**
- 2. Analyze the cumulative revenue generated over time.**



HARD QUESTION RELATED
TO SALES

- 3 Determine the top 3 most ordered pizza types based on revenue for each pizza category.**



QUESTION-11



- Calculate the percentage contribution of each pizza type to total revenue.

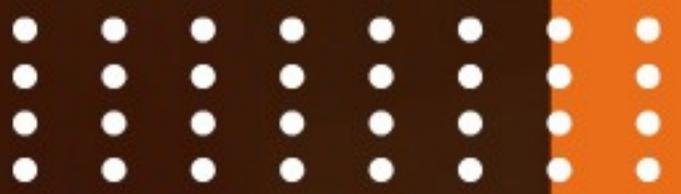
INPUT

```
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 category  
order by revenue desc;
```

OUTPUT

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

QUESTION-12



Analyze the cumulative revenue generated over time.

INPUT

```
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 pizzas join order_details on pizzas.pizza_id =  
                   order_details.pizza_id  
      join orders on order_details.order_id = orders.order_id  
      group by orders.order_date) as sales;
```

OUTPUT

Result Grid		Filter Rows:
	order_date	cum_revenue
1	2015-01-01	2713.8500000000004
2	2015-01-02	5445.75
3	2015-01-03	8108.15
4	2015-01-04	9863.6
5	2015-01-05	11626.55

QUESTION-13



Determine the top 3 most ordered pizza types based on revenue for each pizza category

INPUT

```
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;
```

OUTPUT

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

Pizza Resto Poject

THANK YOU
FOR ATTENTION

See You Next

www.reallygreatsite.com