



SQL PROJECT ON PIZZA SALES



HELLO EVERYONE,
IN THIS PROJECT, I
HAVE UTILIZED SQL
QUERIES TO SOLVE
THE QUESTIONS
THAT WERE
RELATED TO PIZZA
SALES

```
create database pizzahut;
```

- ```
create table orders (
 order_id int not null,
 order_date date not null,
 order_time time not null,
 primary key (order_id)
);
```

- ```
create table orders_details (
    order_details_id int not null,
    order_id int not null,
    pizza_id text not null,
    quantity int not null,
    primary key (order_details_id)
);
```

```
1 -- Retrieve the total number of orders placed.  
2  
3 • SELECT  
4     COUNT(order_id) AS total_pizzas  
5 FROM  
6     orders;
```

total_pizzas
21350

```
1 -- Calculate the total revenue generated from pizza sales.  
2  
3 • SELECT  
4     ROUND(SUM(orders_details.quantity * pizzas.price),  
5             2) as total_revenue  
6 FROM  
7     orders_details  
8     JOIN  
9         pizzas ON orders_details.pizza_id = pizzas.pizza_id;
```

total_revenue
817860.05

```
1 -- Identify the highest-priced pizza.  
2  
3  
4 • SELECT  
5     pizzas.price, pizza_types.name  
6 FROM  
7     pizzas  
8         JOIN  
9     pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
10 ORDER BY pizzas.price DESC  
11 LIMIT 1;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

price	name
35.95	The Greek Pizza

```
1 -- Identify the most common pizza size ordered.  
2  
3  
4 • SELECT  
5     pizzas.size, COUNT(orders_details.quantity) as common_size  
6 FROM  
7     orders_details  
8     JOIN  
9     pizzas ON orders_details.pizza_id = pizzas.pizza_id  
10 GROUP BY pizzas.size order by common_size desc;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

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

```
1 -- List the top 5 most ordered pizza types along with their quantities.  
2  
3 • SELECT  
4     pizza_types.name,  
5     COUNT(orders_details.quantity) AS type_count  
6 FROM  
7     orders_details  
8     JOIN  
9     pizzas ON orders_details.pizza_id = pizzas.pizza_id  
10    JOIN  
11        pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
12    GROUP BY pizza_types.name  
13    ORDER BY type_count DESC  
14    LIMIT 5;  
15
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

name	type_count
The Classic Deluxe Pizza	2416
The Barbecue Chicken Pizza	2372
The Hawaiian Pizza	2370
The Pepperoni Pizza	2369
The Thai Chicken Pizza	2315

```
1 -- Join the necessary tables to find the total quantity of each pizza category ordered.  
2  
3 • SELECT  
4     pizza_types.category,  
5     COUNT(orders_details.quantity) AS category_count  
6 FROM  
7     orders_details  
8     JOIN  
9     pizzas ON orders_details.pizza_id = pizzas.pizza_id  
10    JOIN  
11    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
12 GROUP BY pizza_types.category  
13 ORDER BY category_count DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	category_count
Classic	14579
Supreme	11777
Veggie	11449
Chicken	10815

```
1 -- Group the orders by date and calculate the average number of pizzas ordered per day.  
2  
3 • SELECT  
4     ROUND(AVG(od), 0) as average  
5 FROM  
6     (SELECT  
7         orders.order_date, SUM(orders_details.quantity) AS od  
8     FROM  
9         orders_details  
10    JOIN orders ON orders_details.order_id = orders.order_id  
11    GROUP BY orders.order_date) AS a;
```

Result Grid	
average	
138	

```
1 -- Determine the top 3 most ordered pizza types based on revenue.  
2  
3 • SELECT  
4     pizza_types.name,  
5     SUM(pizzas.price * orders_details.quantity) AS revenue  
6 FROM  
7     orders_details  
8     JOIN  
9     pizzas ON orders_details.pizza_id = pizzas.pizza_id  
10    JOIN  
11     pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
12 GROUP BY pizza_types.name  
13 ORDER BY revenue DESC  
14 LIMIT 3;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

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

```
1      -- Calculate the percentage contribution of each pizza type to total revenue.  
2  
3 •  SELECT  
4      pizza_types.category,  
5      ROUND(SUM(pizzas.price * orders_details.quantity) / (SELECT  
6                  SUM(pizzas.price * orders_details.quantity)  
7                  FROM  
8                  pizzas  
9                  JOIN  
10                 orders_details ON pizzas.pizza_id = orders_details.pizza_id) * 100,  
11             2) AS revenue_percentage  
12  FROM  
13      orders_details  
14      JOIN  
15      pizzas ON orders_details.pizza_id = pizzas.pizza_id  
16      JOIN  
17      pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
18  GROUP BY pizza_types.category  
19  ORDER BY revenue_percentage DESC;  
20
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	category	revenue_percentage
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

```
1 -- Analyze the cumulative revenue generated over time.  
2  
3 • select order_date,  
4 round(sum(revenue) over(order by order_date),2) as cumu_rev  
5 from  
6     (select orders.order_date,  
7      sum(pizzas.price * orders_details.quantity) as revenue  
8      from orders_details  
9      join pizzas on orders_details.pizza_id = pizzas.pizza_id  
10     join orders on orders_details.order_id = orders.order_id  
11   group by orders.order_date) as sales;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

order_date	cumu_rev
2015-01-01	2713.85
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

Result 11 X

```
1 -- Determine the top 3 most ordered pizza types based on revenue for each pizza category.  
2  
3 • select name, revenue, category from  
4     (select name, revenue, category,  
5      rank() over(partition by category order by revenue desc) as rn from  
6      (select pizza_types.category,  
7       sum(pizzas.price * orders_details.quantity) as revenue,  
8       pizza_types.name  
9      from orders_details  
10     join pizzas on orders_details.pizza_id = pizzas.pizza_id  
11     join pizza_types on pizza_types.pizza_type_id = pizzas.pizza_type_id  
12     group by pizza_types.category, pizza_types.name  
13     order by pizza_types.category) as a) as b  
14 where rn <= 3;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

name	revenue	category
The Thai Chicken Pizza	43434.25	Chicken
The Barbecue Chicken Pizza	42768	Chicken
The California Chicken Pizza	41409.5	Chicken
The Classic Deluxe Pizza	38180.5	Classic
The Hawaiian Pizza	32273.25	Classic
The Pepperoni Pizza	30161.75	Classic

Result 11 x



THANK YOU.