



LARANA PIZZA

LARANA PIZZA



WELCOME TO SQL PORTFOLIO PROJECT





LARANA PIZZA

HELLO!

MY NAME IS SWETHA.M.
IN THIS PROJECT I AM
UTILIZING SQL QUERIES TO
SOLVE QUESTIONS THAT ARE
RELATED  TO PIZZA SALES.



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
SELECT COUNT(order_id) as total_orders FROM orders;
```

Result Grid	
	total_orders
	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
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;
```

Result Grid	
	total_sales
▶	817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA.

```
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;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

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

Result Grid		
	size	order_count
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

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

```
SELECT pizza_types.name,SUM(order_details.quantity) AS quantity
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.name ORDER BY quantity DESC LIMIT 5;
```

Result Grid		
	name	quantity
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

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 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 ORDER BY quantity DESC;
```

Result Grid | Filter

	category	quantity
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT HOUR(order_time) AS hour, COUNT(order_id) AS order_count  
FROM orders GROUP BY HOUR(order_time);
```

	hour	order_count
11	1231	
12	2520	
13	2455	
14	1472	
15	1468	
16	1920	
17	2336	
18	2399	
19	2009	
20	1642	
21	1198	
22	663	
23	28	
10	8	
9	1	

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

```
SELECT category, count(name) FROM pizza_types  
GROUP BY category;
```

Result Grid		
	category	count(name)
>	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9
	11:38:36	2
	11:57:40	1
	12:12:28	1
	12:16:31	3
	12:21:30	1
	12:29:36	1
	12:50:37	1
	12:51:37	1
	12:52:01	1
	13:00:15	2
	13:02:59	1
	13:04:41	2
	13:11:55	1
	13:14:19	2
	13:33:00	2
	13:34:07	3
	13:53:00	1
	13:57:08	1
	13:59:09	1
	14:03:08	3
	14:14:29	1

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

```
SELECT ROUND(avg(quantity),0) AS avg_pizza_ordered_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;
```

Result Grid	
	avg_pizza_ordered_per_day
▶	138

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

```
SELECT pizza_types.name,  
       sum(order_details.quantity * pizzas.price) as revenue  
  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.name ORDER BY revenue desc limit 3;
```

Result Grid		
	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

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

```
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 pizza_types.category order by revenue desc;
```

Result Grid		
	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
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 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) AS sales;
```

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
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.35000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.30000000003
2015-01-14	32358.70000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.60000000006
2015-01-19	43365.75000000001
2015-01-20	45763.65000000001
2015-01-21	47804.20000000001
2015-01-22	50300.90000000001
2015-01-23	52724.60000000006
2015-01-24	55013.85000000006
2015-01-25	56631.40000000001

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)
   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	
	name
	revenue
1	The Thai Chicken Pizza
2	43434.25
3	The Barbecue Chicken Pizza
4	42768
5	The California Chicken Pizza
6	41409.5
7	The Classic Deluxe Pizza
8	38180.5
9	The Hawaiian Pizza
10	32273.25
11	The Pepperoni Pizza
12	30161.75
13	The Spicy Italian Pizza
14	34831.25
15	The Italian Supreme Pizza
16	33476.75
17	The Sicilian Pizza
18	30940.5
19	The Four Cheese Pizza
20	32265.70000000065
21	The Mexicana Pizza
22	26780.75
23	The Five Cheese Pizza
24	26066.5



FREE DELIVERY SERVICE

Ut enim ad minim veniam.
Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.





LARANA PIZZA

THANK YOU!

