

# PIZZA POINT



# ABOUT US

- Delicious Variety: Choose from classic Margherita to loaded meat-lovers or veggie-packed options.
- Fresh Ingredients: Hand-tossed dough, savory sauces, and premium toppings.
- Fast & Hot Delivery: Enjoy your pizza fresh and piping hot, right at your door.



# OUR MENU

- THE CLASSIC DELUXE PIZZA
- THE BARBECUE CHICKEN PIZZA
- THE HAWAIIAN PIZZA
- THE PEPPERONI PIZZA
- THE THAI CHICKEN PIZZA
- THE CALIFORNIA CHICKEN PIZZA
- THE SICILIAN PIZZA
- THE SPICY ITALIAN PIZZA



- THE SOUTHWEST CHICKEN PIZZA
- THE BIG MEAT PIZZA
- THE FOUR CHEESE PIZZA
- THE ITALIAN SUPREME PIZZA
- THE VEGETABLES + VEGETABLES PIZZA
- THE MEXICANA PIZZA
- THE NAPOLITANA PIZZA
- THE PROSCIUTTO AND ARUGULA PIZZA
- THE PEPPER SALAMI PIZZA
- THE SPINACH AND FETA PIZZA
- THE ITALIAN CAPOCOLLO PIZZA
- THE GREEK PIZZA
- THE FIVE CHEESE PIZZA
- THE PEPPERONI, MUSHROOM, AND PEPPERS PIZZA
- THE GREEN GARDEN PIZZA
- THE CHICKEN ALFREDO PIZZA
- THE ITALIAN VEGETABLES PIZZA
- THE CHICKEN PESTO PIZZA
- THE SPINACH PESTO PIZZA
- THE SOPPRESSATA PIZZA
- THE SPINACH SUPREME PIZZA
- THE CALABRESE PIZZA
- THE MEDITERRANEAN PIZZA

# 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 order_details.pizza_id = pizzas.pizza_id
```

Result Grid	
	total_sales
▶	817860.05

# 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 | Filter Rows:

	name	quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	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 |

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	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
```

	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 | Filter Row

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

# 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  
    order_details  
JOIN orders ON order_details.order_id = orders.order_id  
GROUP BY orders.order_date) AS order_quantity
```

	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 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 revenue DESC  
 LIMIT 3
```

	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  
 SUM(order_details.quantity * pizzas.price)  
 FROM order_details  
 JOIN pizzas ON order_details.pizza_id = pizzas.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
```

Result Grid | Filter

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

# 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 limit 1

Result Grid |

	size	order_count
▶	L	18526

# DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
select category, 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
```

	category	name	revenue
1	Chicken	The Thai Chicken Pizza	43434.
2	Chicken	The Barbecue Chicken Pizza	42768
3	Chicken	The California Chicken Pizza	41409.
4	Classic	The Classic Deluxe Pizza	38180.
5	Classic	The Hawaiian Pizza	32273.
6	Classic	The Pepperoni Pizza	30161.
7	Supreme	The Spicy Italian Pizza	34831.
8	Supreme	The Italian Supreme Pizza	33476.
9	Supreme	The Sicilian Pizza	30940.
10	Veggie	The Four Cheese Pizza	32265.
11	Veggie	The Mexicana Pizza	26780.

# 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	2713.850000000004
	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

order_date	cum_revenue	order_date	cum_revenue	order_date	cum_revenue	order_date	cum_revenue
2015-01-17	39001.75000000001	2015-02-01	72982.50000000001	2015-02-17	109334.45000000001	2015-03-05	146013.34999999998
2015-01-18	40978.60000000006	2015-02-02	75311.10000000002	2015-02-18	111977.30000000002	2015-03-06	148527.3
2015-01-19	43365.75000000001	2015-02-03	77925.90000000002	2015-02-19	114007.55000000002	2015-03-07	150927.75
2015-01-20	45763.65000000001	2015-02-04	80159.80000000002	2015-02-20	116898.70000000001	2015-03-08	153115.9
2015-01-21	47804.20000000001	2015-02-05	82375.60000000002	2015-02-21	119009.70000000001	2015-03-09	155450.44999999998
2015-01-22	50300.90000000001	2015-02-06	84885.55000000002	2015-02-22	120589.65000000001	2015-03-10	157839.15
2015-01-23	52724.60000000006	2015-02-07	87123.20000000001	2015-02-23	122758.20000000001	2015-03-11	160046.85
2015-01-24	55013.85000000006	2015-02-08	89158.20000000001	2015-02-24	124952.75000000001	2015-03-12	162041.75
2015-01-25	56631.40000000001	2015-02-09	91353.55000000002	2015-02-25	127294.05000000002	2015-03-13	164828.4
2015-01-26	58515.80000000001	2015-02-10	93410.05000000002	2015-02-26	129555.35000000002	2015-03-14	166867.85
2015-01-27	61043.85000000001	2015-02-11	95870.05000000002	2015-02-27	132413.3000000002	2015-03-15	168936.45
2015-01-28	63059.85000000001	2015-02-12	98028.85000000002	2015-02-28	134952.9000000002	2015-03-16	171231.5
2015-01-29	65105.150000000016	2015-02-13	100783.35000000002	2015-03-01	136551.45	2015-03-17	174196.8
2015-01-30	67375.45000000001	2015-02-14	103102.50000000001	2015-03-02	138930.5	2015-03-18	176272.19999999998
2015-01-31	69793.30000000002	2015-02-15	105243.75000000001	2015-03-03	141218.4	2015-03-19	178660.8
2015-02-01	72982.50000000001	2015-02-16	107212.55000000002	2015-03-04	143662.69999999998	2015-03-20	181122.05



LARANA PIZZA

# THANK YOU!

