



***Delicious Pizzas For
Everyone!***

Specialist in Italian Food





BORCELLE
R E S T A U R A N T

HELLO!

My name is Avantika Rajpurohit .

In this project i have utilized the SQL queries to solve and analyse the sales of pizzas



BORCELLE
RESTAURANT

THE QUESTIONS WE GOING TO COVER IN THIS PROJECT :-

- 1.RETRIEVEV THE TOTAL NUMBERS OF ORDERS PLACED.
- 2.CALCULATE TOTAL REVENUE GENERATED FROM PIZZA SALES.
- 3.IDENTIFY THE HIGHEST-PRICED PIZZA.
- 4.IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED
- 5.LIST THE TOP 5 ORDERED PIZZA TYPE ALONG WITH THEIR QUANTITY
- 6.IDENTIFY THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED
- 7.DETERMINE THE DISTRUBUTION OF ORDERS BY HOUR OF THE DAY
- 8.DETERMINE THE CATEGORY-WISE DISTRUBUTION OF PIZZAS
- 9.GROUP THE NUMBER OF DAY AND CALCULATE THE AVERAGE NUMBERS OF PIZZAS ORDERED PER DAY
- 10.DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE





RETRIEVEV THE TOTAL NUMBERS OF ORDERS PLACED :

```
SELECT  
    COUNT(order_id) AS total_orders  
FROM  
    orders1 ;
```

	total_orders
▶	21350



CALCULATE TOTAL REVENUE GENERATED FROM PIZZA SALES:

```
SELECT  
    ROUND(SUM(orders_details.quantity * pizzas.price),  
        2) AS total_revenue  
  
FROM  
    orders_details  
    JOIN  
    pizzas ON pizzas.pizza_id = orders_details.pizza_id;
```

	total_revenue
▶	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;
```

	name	price
▶	The Greek Pizza	35.95



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

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

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





LIST THE TOP 5 ORDERED PIZZA TYPE ALONG WITH THEIR QUANTITY:

```
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

	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



IDENTIFY THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED:

```
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050





DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY:

```
SELECT  
    HOUR(time) AS Hours, COUNT(order_id) AS order_count  
FROM  
    orders1  
GROUP BY Hours;
```

	Hours	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



DETERMINE THE CATEGORY-WISE DISTRUBUTION OF PIZZAS:

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

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

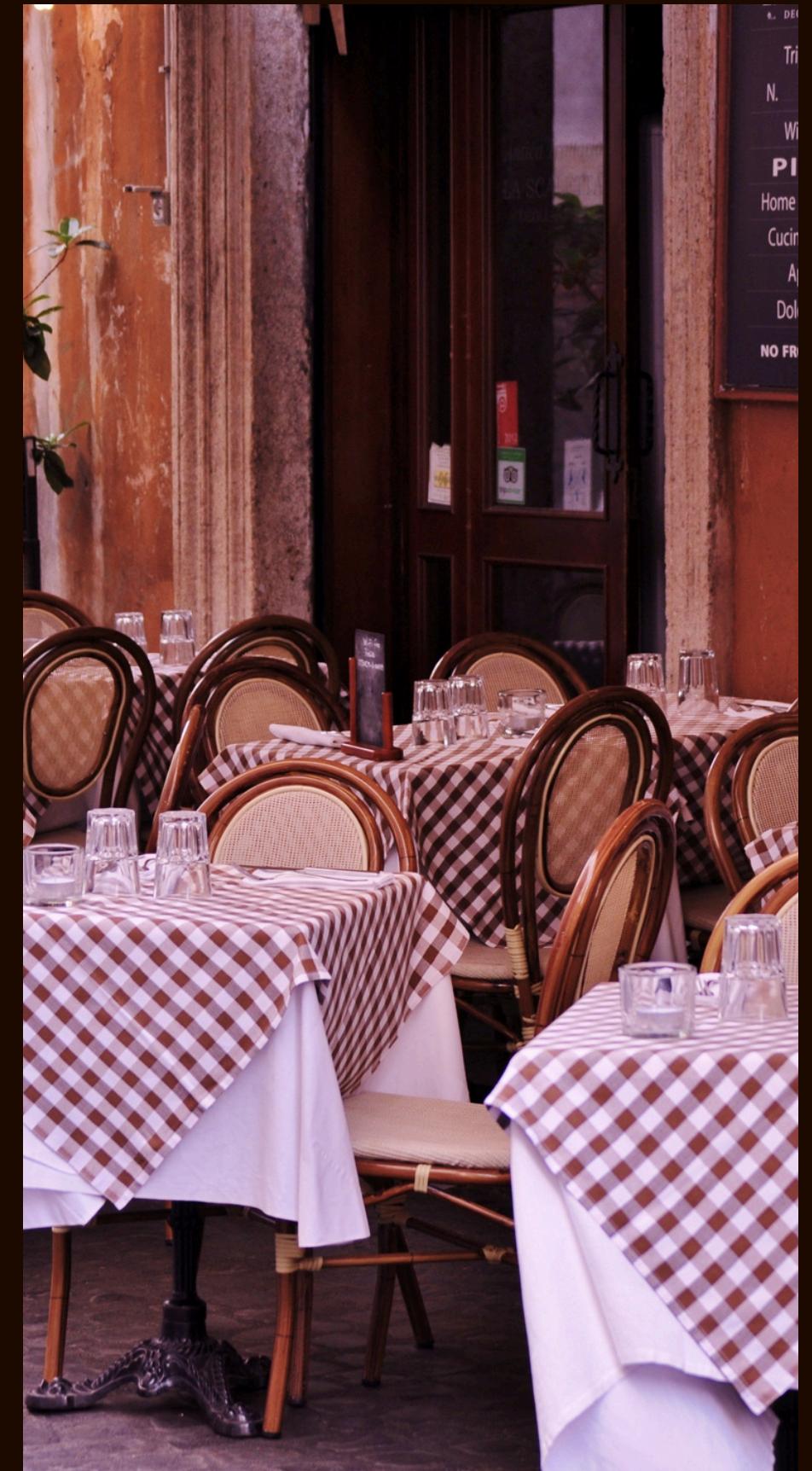




GROUP THE NUMBER OF DAY AND CALCULATE THE AVERAGE NUMBERS OF PIZZAS ORDERED PER DAY:

```
SELECT
    ROUND(AVG(quantity)) AS avg_pizza_ordered_per_day
FROM
    (SELECT
        date, SUM(orders_details.quantity) AS quantity
    FROM
        orders1
    JOIN orders_details ON orders1.order_id = orders_details.order_id
    GROUP BY orders1.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(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN
    orders_details ON orders_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



BORCELLE
R E S T A U R A N T

THANK YOU

avantikaraj7360@gmail.comt

