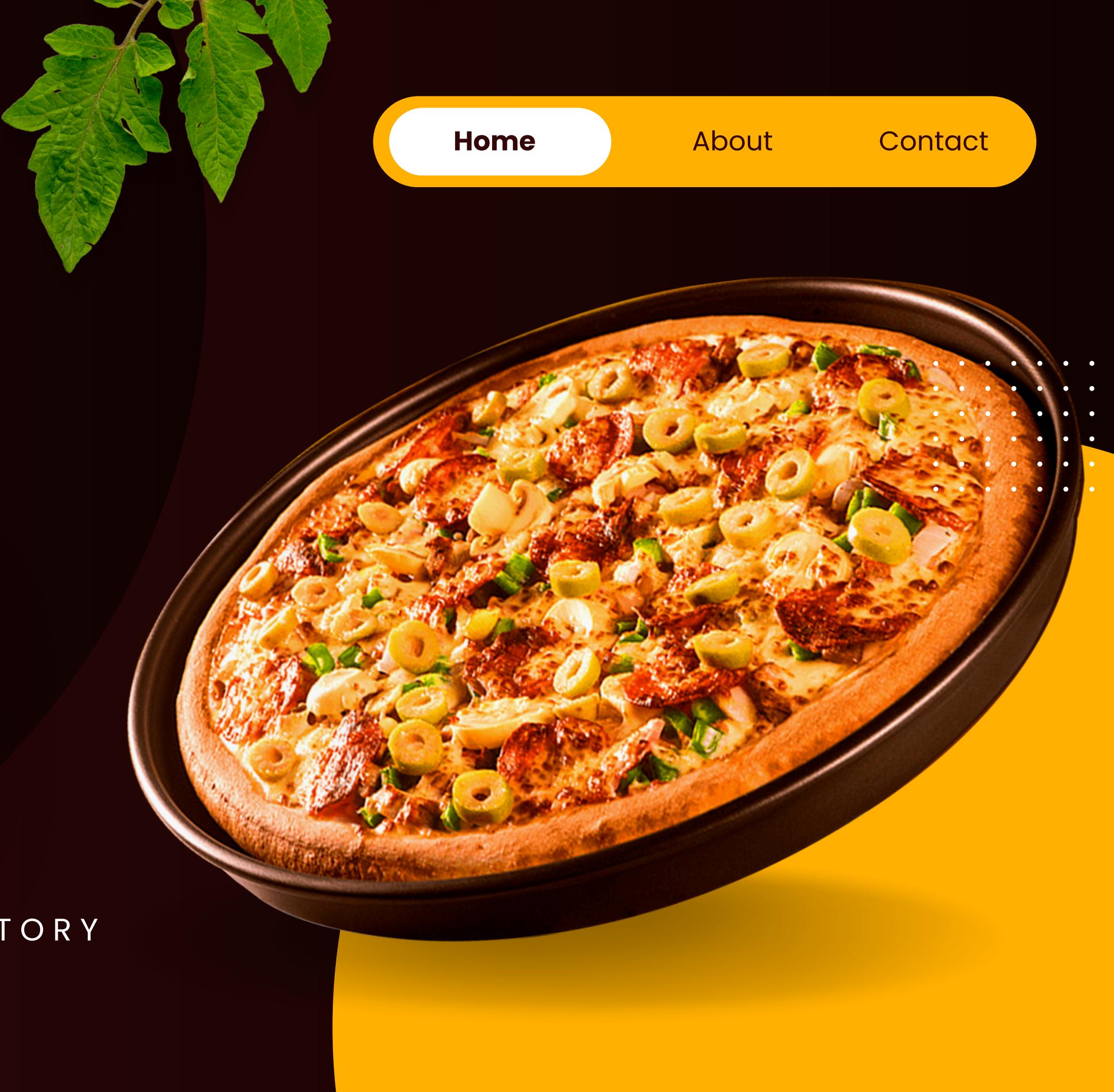




SHODWE
Pizza Resto

PIZZA RESTO

● WHERE EVERY SLICE TELLS A STORY



Home

About

Contact



ABOUT US

HELLO

My name is Mayank Kapoor and In this project I have utilized SQL queries to solve questions that were related to Pizza Sales.

QUESTIONS I AM SOLVING?

About

Contact

- 
- 1 Basic:
 - 2 Retrieve the total number of orders placed.
 - 3 Calculate the total revenue generated from pizza sales.
 - 4 Identify the highest-priced pizza.
 - 5 Identify the most common pizza size ordered.
 - 6 List the top 5 most ordered pizza types along with their quantities.
 - 7
 - 8
 - 9 Intermediate:
 - 10 Join the necessary tables to find the total quantity of each pizza category ordered.
 - 11 Determine the distribution of orders by hour of the day.
 - 12 Join relevant tables to find the category-wise distribution of pizzas.
 - 13 Group the orders by date and calculate the average number of pizzas ordered per day.
 - 14 Determine the top 3 most ordered pizza types based on revenue.
 - 15
 - 16 Advanced:
 - 17 Calculate the percentage contribution of each pizza type to total revenue.
 - 18 Analyze the cumulative revenue generated over time.
 - 19 Determine the top 3 most ordered pizza types based on revenue for each pizza category.



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.

[Home](#)

[About](#)

[Contact](#)

SELECT

```
ROUND(SUM(od.quantity * p.price), 2) AS total_revenue
```

FROM

```
order_details AS od
```

JOIN

```
pizzas p ON od.pizza_id = p.pizza_id;
```

Result Grid



	total_revenue
▶	817860.05



IDENTIFY THE HIGHEST-PRICED PIZZA.

[Home](#)[About](#)[Contact](#)

```
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 Row

	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
LIMIT 1;
```

Result Grid | Filter

	size	order_count
▶	L	18526



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

SELECT

```
 pizza_types.name,  
 SUM(order_details.quantity) AS order_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 order_quantity **DESC**

LIMIT 5;

	name	order_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 order_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 order_quantity **DESC**;

Result Grid | Filter Rows:

	category	order_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(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) AS distribution
FROM
    pizza_types
GROUP BY category;
```

Result Grid | Filter Row

	category	distribution
▶	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_order_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 order_date) AS order_quantity;
```

Result Grid		Filter Rows
avg_pizza_order_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  
          ROUND(SUM(od.quantity * p.price), 2) AS total_revenue  
       FROM  
          order_details AS od  
       JOIN  
          pizzas p ON od.pizza_id = p.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 | Filter

	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



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) 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;
```

	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
	The Pepperoni Pizza	30161.75
	The Spicy Italian Pizza	34831.25
	The Italian Supreme Pizza	33476.75
	The Sicilian Pizza	30940.5
	The Four Cheese Pizza	32265.7000000065
	The Mexicana Pizza	26780.75
	The Five Cheese Pizza	26066.5



SHODWE
Pizza Resto

[Home](#)

[About](#)

[Contact](#)

THANK YOU

FOR ATTENTION

- 2025 PIZZA RESTO PRESENTATION