Pizza Sales Analysis Using SQL

Project Objectives-

- Understand customer behavior by analyzing order patterns, pizza sizes, and popular types.
- Measure business performance through revenue calculations, category contributions, and cumulative growth.
- Apply SQL skills such as aggregation, grouping, joins, and analytical queries to solve real-world problems.
- Generate actionable insights that highlight trends in sales, peak ordering times, and top-performing pizzas.

DATABASE SCHEMA OVERVIEW

MENU HIGHLIGHTS

Table Name	Key Columns	Purpose
Orders	order_id, date, time	Stores each order's basic info (when it was placed).
Orders_details	order_details_id, order_id, pizza_id, quantity	Links orders to specific pizzas and records how many were ordered.
Pizzas	pizza_id, size, price	Contains details about each pizza's size and price.
Pizza_types	pizza_type_id, name, category, ingredients	Provides descriptive info about pizza varieties.

SHODWE
Pizza Resto

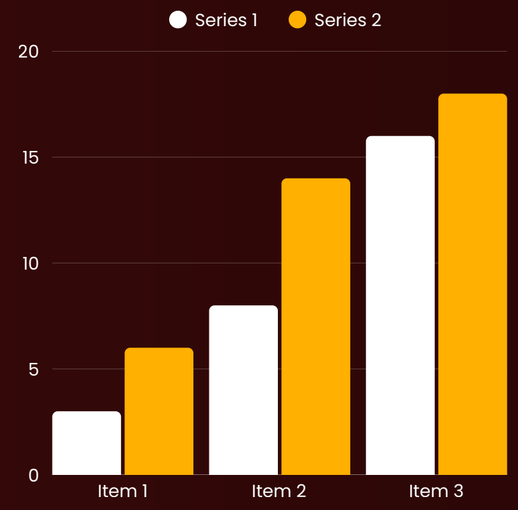
Contact

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
3 •   SELECT COUNT(order_id) AS total_orders FROM orders;
```

Result Grid

total_orders
21350



SHODWE
Pizza Resto

Contact

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
3 •   SELECT
4         ROUND(SUM(orders_details.quantity * pizzas.price),
5                 2) AS total_sales
6     FROM
7         orders_details
8             JOIN
9         pizzas ON pizzas.pizza_id = orders_details.pizza_id
```

Result Grid

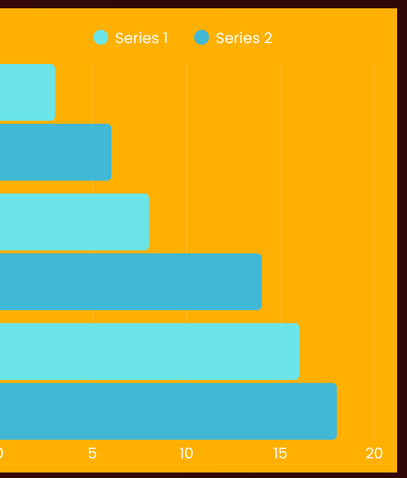
	total_sales
▶	817860.05

SHODWE
Pizza Resto

Contact

IDENTIFY THE HIGHEST-PRICED PIZZA.

```sql
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

SHODWE
Pizza Resto

Contact

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

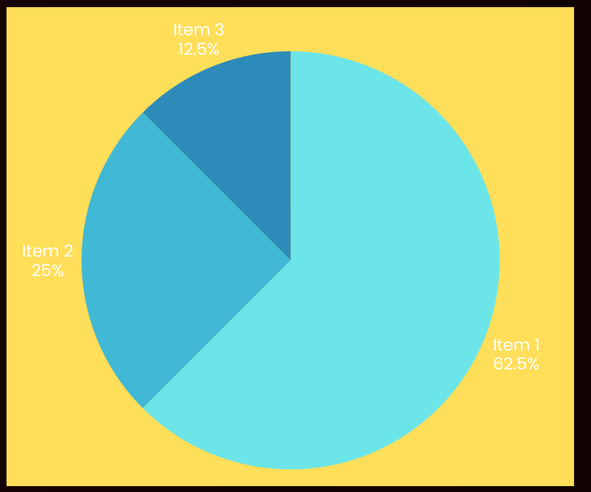
```sql
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid | Filter

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

SHODWE
Pizza Resto

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.




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

Result Grid | Filter

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

SHODWE
Pizza Resto

Contact

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```sql
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
```

Result Grid | Filter

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

SHODWE
Pizza Resto

Contact

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
3 •   SELECT
4         HOUR(order_time) AS hour, COUNT(order_id) AS order_count
5     FROM
6         orders
7     GROUP BY HOUR(order_time);
```

Result Grid | Filter Rc

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

SIGNATURE PIZZA

SHODWE
Pizza Resto

Contact

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

```
3 •   SELECT
4         category, COUNT(name)
5     FROM
6         pizza_types
7     GROUP BY category;
```

Result Grid | Filter Rows:

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

SHODWE
Pizza Resto

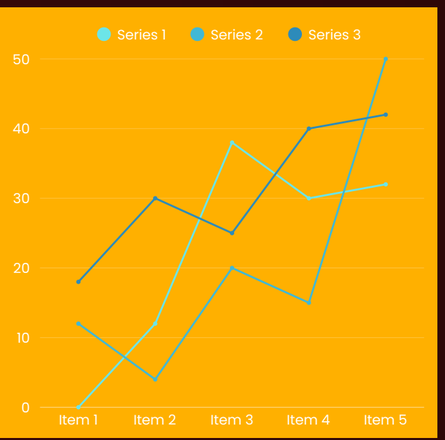
Contact

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

```sql
SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders_details.order_id = orders.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid | Filter Rows:

avg_pizza_ordered_per_day
138



SHODWE
Pizza Resto

Contact

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

```sql
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;
```

Result Grid | Filter Rows:

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

SHODWE
Pizza Resto

Contact

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

```sql
SELECT
    pizza_types.category,
    ROUND( (SUM(orders_details.quantity * pizzas.price) / (SELECT
            ROUND(SUM(orders_details.quantity * pizzas.price),
                    2) AS total_sales
        FROM
            orders_details
                JOIN
            pizzas ON pizzas.pizza_id = orders_details.pizza_id)) * 100, 2) AS revenue
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 revenue DESC;
```

Result Grid | Filter

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

SHODWE
Pizza Resto

Contact

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```sql
SELECT order_date,
ROUND(SUM(revenue) OVER (ORDER BY order_date), 2)AS cum_revenue
FROM
(SELECT orders.order_date,
ROUND(SUM(orders_details.quantity * pizzas.price), 2) AS revenue
FROM orders_details JOIN pizzas ON orders_details.pizza_id = pizzas.pizza_id
JOIN orders ON orders.order_id = orders_details.order_id
GROUP BY orders.order_date) AS sales;
```

Result Grid | Filter Rows:

order_date	cum_revenue
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
2015-01-07	16560.7
2015-01-08	19399.05

Result 4

SHODWE
Pizza Resto

Contact

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

```sql
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((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.category, pizza_types.name) AS a) AS b
WHERE rn <= 3;
```

Result Grid | Filter Rows:

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	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

41409.5

SHODWE
Pizza Resto

Home About Contact

THANK YOU! THROUGH THIS PROJECT, WE EXPLORED THE POWER OF SQL IN ANALYZING SALES DATA AND PRESENTING ACTIONABLE INSIGHTS.

THE PIZZAHUT SCHEMA AND QUERIES SHOWCASE THE JOURNEY FROM DATA TO DECISIONS.