

PIZZA SALES

U S I N G M Y S Q L

Data Analysis





ABOUT ME

I am an aspiring data analyst with passion for transforming data into meaningful insights. Through self-learning and guided projects, I have developed skills in SQL, Excel and Power BI. Recently, I completed a MySQL data analysis project, where I gained practical experience in data analysis and solved questions related to Pizza Sales.

WHY SQL?

SQL is essential for data analysts because it enables efficient retrieval, manipulation, and analysis of data from databases. It allows analysts to query large datasets, clean and prepare data, and perform complex operations, making it a fundamental tool for deriving insights and supporting data-driven decisions.





BASIC

1. Retrieve the total number of orders placed.

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

Result Grid | Filter Rows: Export: Wrap Cell Content:

total_orders
21350



BASIC

2. Calculate the total revenue generated from pizza sales.

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

Result Grid | Filter Rows: Export: Wrap Cell Content:

	total_sales
▶	506956.65

BASIC

3. Identify the highest-priced pizza.

```
3 •   SELECT
4     pizza_types.name, pizzas.price
5   FROM
6     pizza_types
7       JOIN
8     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9   ORDER BY pizzas.price DESC
10  LIMIT 1;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows

name	price
The Greek Pizza	35.95





BASIC

4. Identify the most common pizza size ordered.

```
3 • SELECT
4     pizzas.size,
5     COUNT(order_details.order_details_id) AS order_count
6 FROM
7     pizzas
8     JOIN
9     order_details ON pizzas.pizza_id = order_details.pizza_id
10    GROUP BY pizzas.size
11    ORDER BY order_count DESC;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	size	order_count
▶	L	11495
	M	9561
	S	8680
	XL	352
	XXL	17

BASIC

5. List the top 5 most ordered pizza types along with their quantities.

```
3 •   SELECT
4     pizza_types.name, SUM(order_details.quantity) AS quantity
5   FROM
6     pizza_types
7       JOIN
8       pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9       JOIN
10      order_details ON order_details.pizza_id = pizzas.pizza_id
11    GROUP BY pizza_types.name
12  ORDER BY quantity DESC
13  LIMIT 5;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows:

name	quantity
The Barbecue Chicken Pizza	1548
The Pepperoni Pizza	1487
The Classic Deluxe Pizza	1483
The California Chicken Pizza	1465
The Hawaiian Pizza	1463





INTERMEDIATE

1. Join the necessary tables to find the total quantity of each pizza category ordered.

```
2 •  SELECT
3      pizza_types.category,
4      SUM(order_details.quantity) AS quantity
5  FROM
6      pizza_types
7      JOIN
8          pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9      JOIN
10         order_details ON order_details.pizza_id = pizzas.pizza_id
11     GROUP BY pizza_types.category
12     ORDER BY quantity DESC;
13
```



category	quantity
Classic	9153
Supreme	7443
Veggie	7280
Chicken	6821

INTERMEDIATE

2. Determine the distribution of orders by hour of the day.

```
3 •  SELECT
4      HOUR(order_time), COUNT(order_id) AS order_count
5  FROM
6      orders
7  GROUP BY HOUR(order_time);
8
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

HOUR(order_time)	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



PIZZA SALES
DATA ANALYSIS

THANK YOU

