A Beginner’s Journey into Data Exploration

PIZZAHUT

Didarul Islam
August 1, 2024

This presentation explores basic SQL queries applied to a pizza ordering database. It covers counting total orders, calculating revenue, identifying popular pizza types, and analyzing customer preferences, providing a solid foundation for data analysis beginners.

Dataset Overview

This database is designed to manage and analyze pizza orders efficiently. It includes the following tables:

- Orders: Stores information about each order, such as order_id, date, and time.

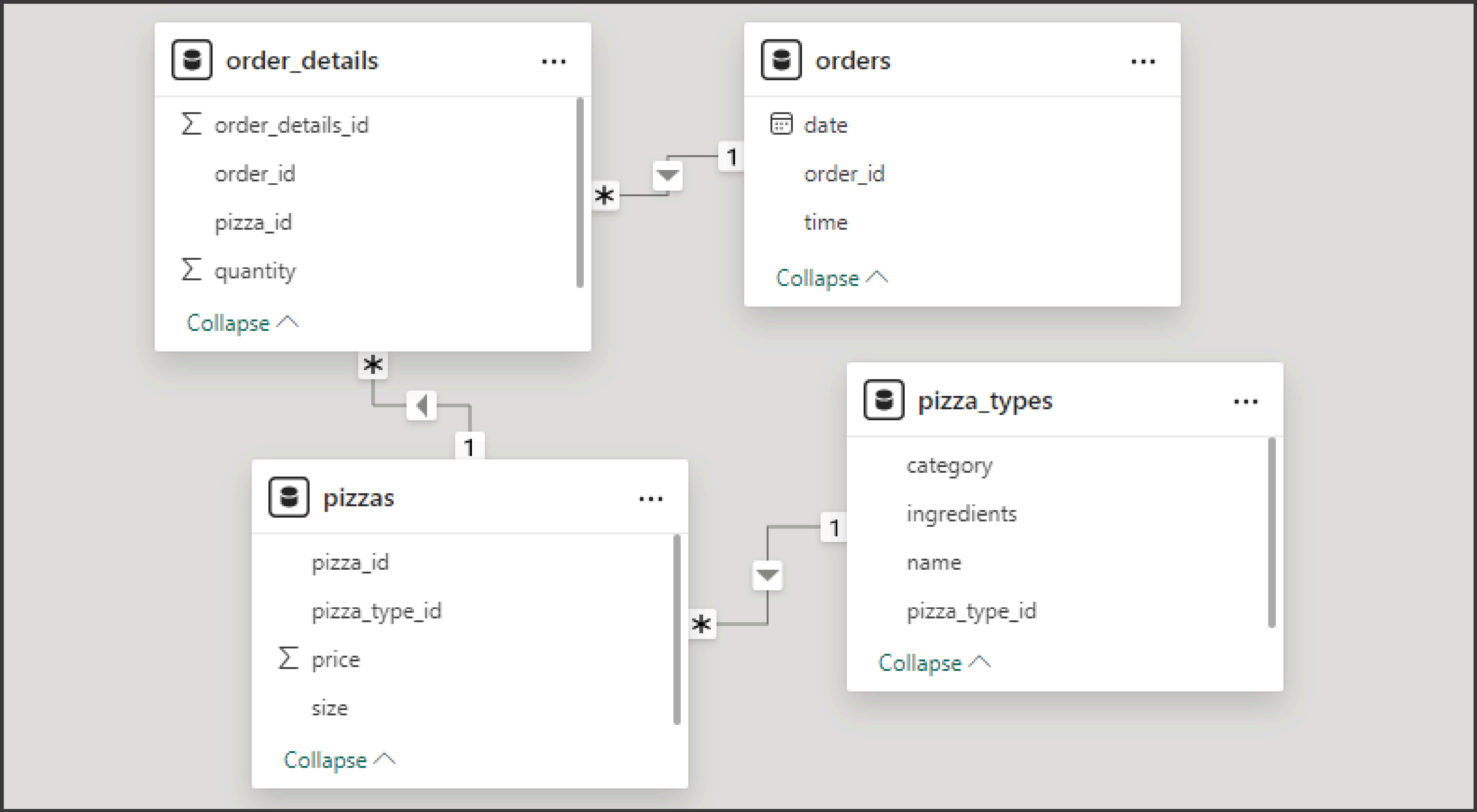
- Order_Details: Contains details of each item within an order, including order_details_id, order_id, pizza_id and quantity.

- Pizzas: Information about pizzas, with attributes like pizza_id, pizza_type_id, size and price.

- Pizza_Types: Categorizes pizzas into different types, such as Chicken, Classic, etc., with attributes including pizza_type_id, name, category and type ingredients.

This structure allows for comprehensive tracking and analysis of pizza sales and customer preferences.

Model View



order_details ...

∑ order_details_id

order_id

pizza_id

∑ quantity

Collapse ^

orders ...

date

order_id

time

Collapse ^

pizza_types ...

category

ingredients

name

pizza_type_id

Collapse ^

pizzas ...

pizza_id

pizza_type_id

∑ price

size

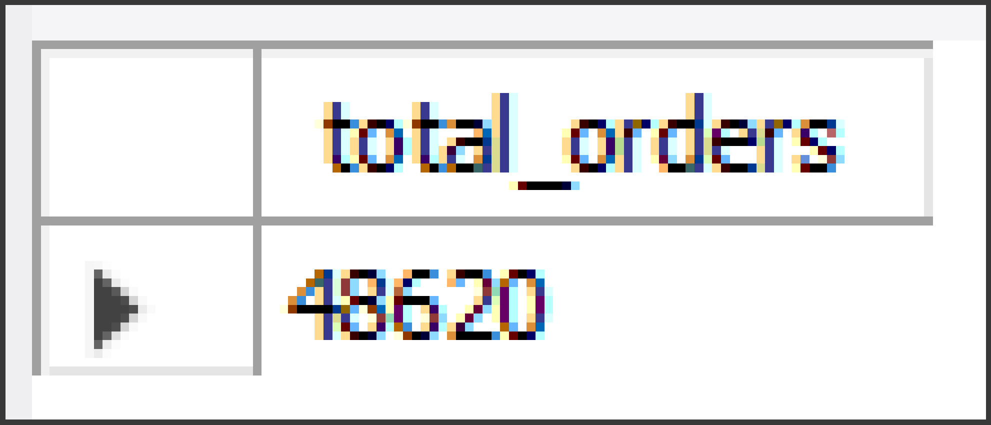
Collapse ^

SQL Queries and Outputs

- Total Orders
- Total Revenue
- Highest-Priced Pizza
- Most Common Pizza Size
- Top 5 Pizza Types
- Category Distribution
- Order Timing by Hour
- Average Pizzas per Day
- Top 3 Pizza Types by Revenue

Total Orders

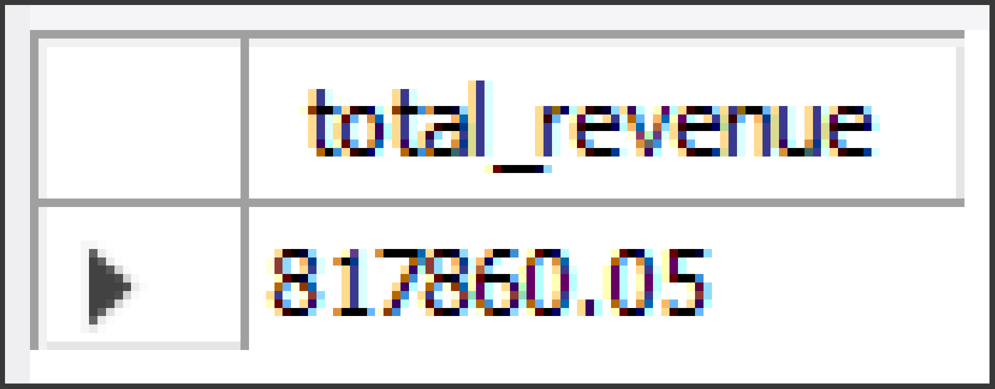
```
SELECT
   COUNT(order_id) AS total_orders
FROM
   order_details;
```



	total_orders
▶	48620

Total Revenue

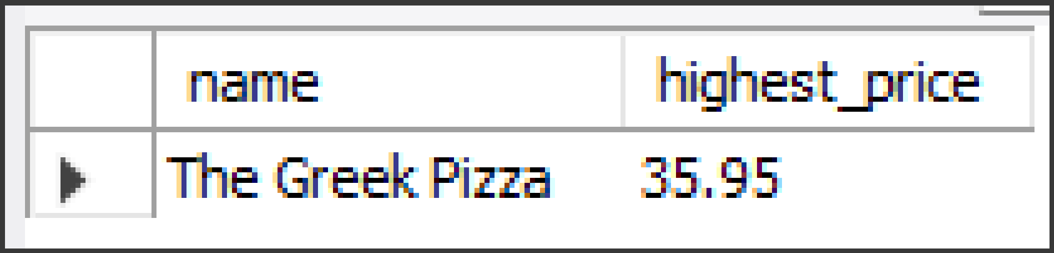
```
SELECT
    ROUND(SUM(quantity * price), 2) AS total_revenue
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```



	total_revenue
▶	817860.05

Highest-Priced Pizza

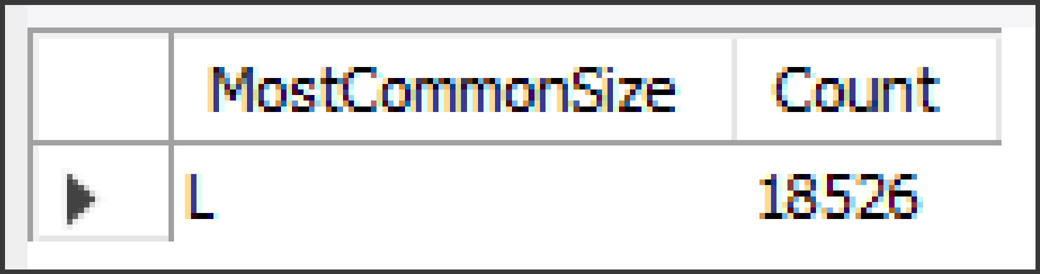
```
SELECT
    pizza_types.name, pizzas.price AS highest_price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```



	name	highest_price
▶	The Greek Pizza	35.95

Most Common Pizza Size

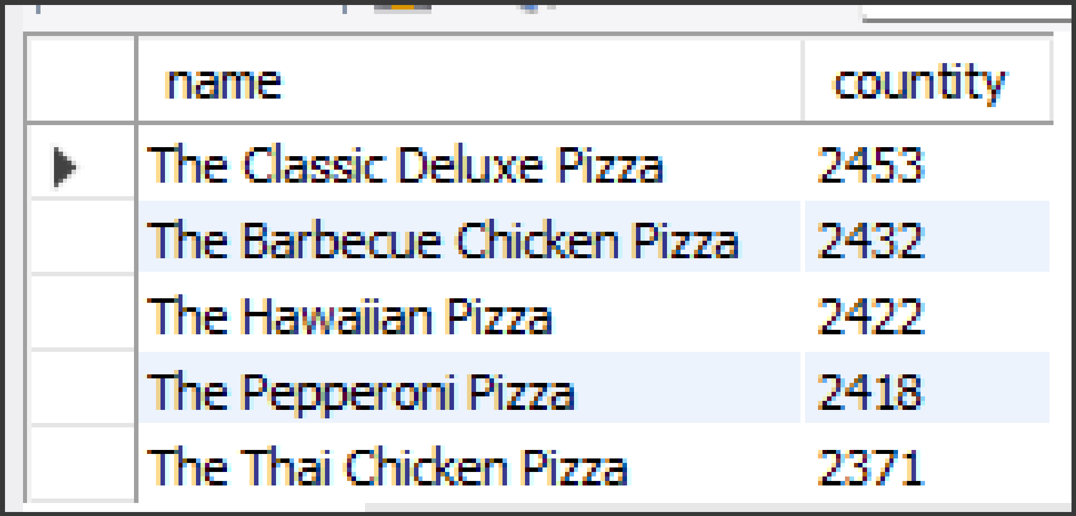
```
SELECT
    pizzas.size AS MostCommonSize,
    COUNT(order_details.order_details_id) AS Count
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY size
ORDER BY count DESC
LIMIT 1;
```



	MostCommonSize	Count
▶	L	18526

Top 5 Pizza Types

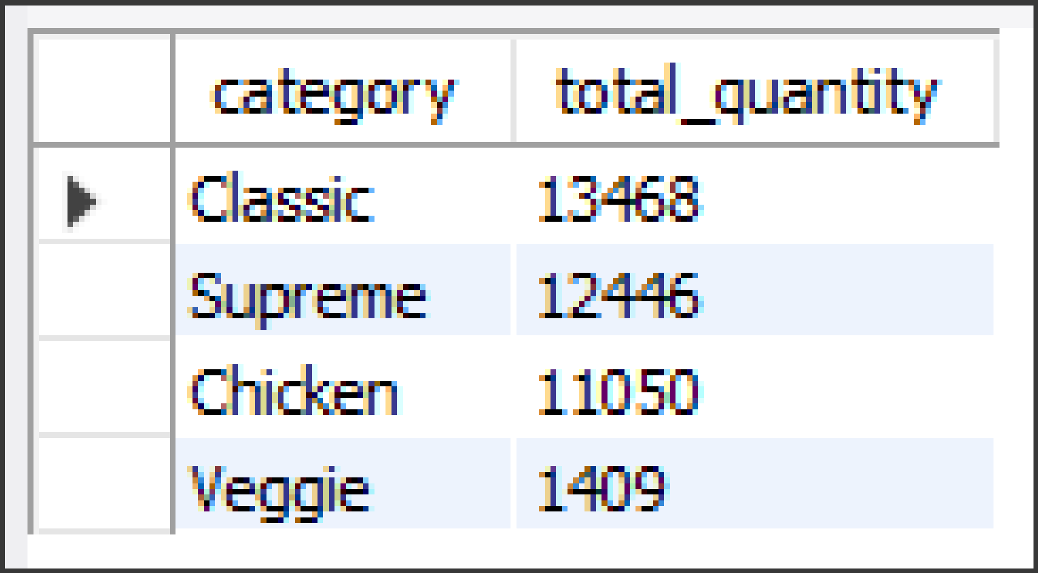
```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS
countity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id =
order_details.pizza_id
GROUP BY name
ORDER BY countity DESC
LIMIT 5;
```



name	countity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Category Distribution

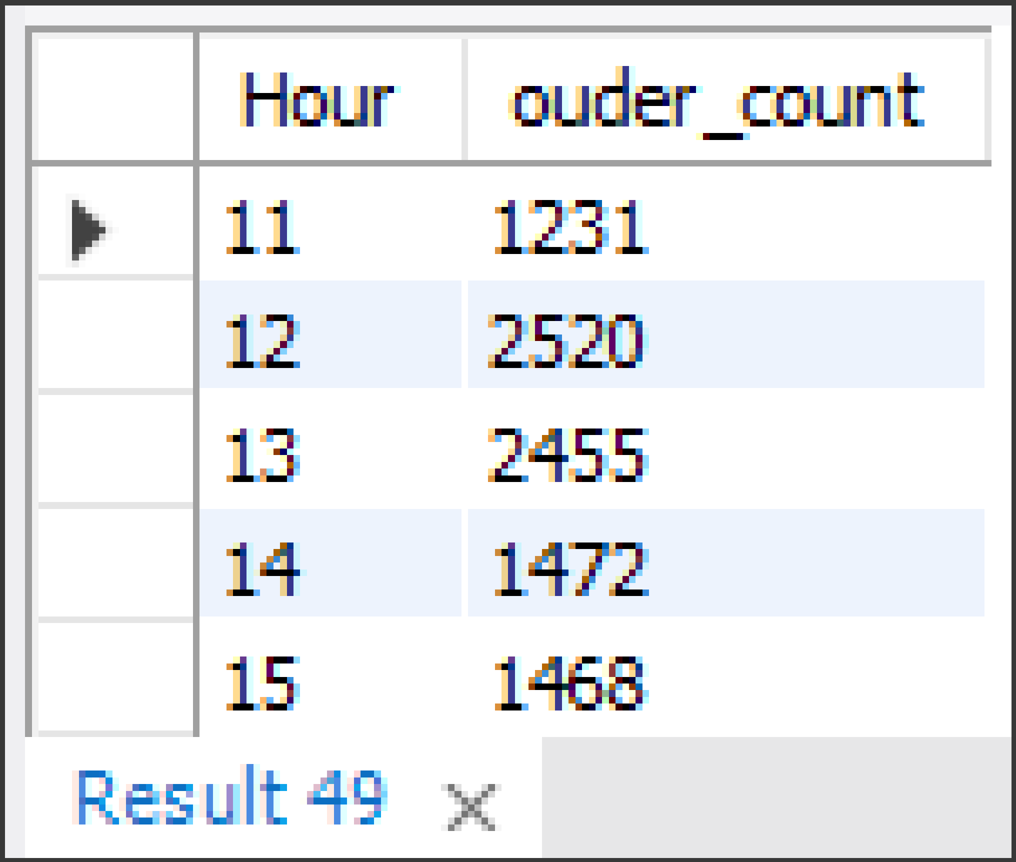
```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id =
order_details.pizza_id
GROUP BY pizza_types.category
ORDER BY total_quantity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



category	total_quantity
Classic	13468
Supreme	12446
Chicken	11050
Veggie	1409

Order Timing by Hour

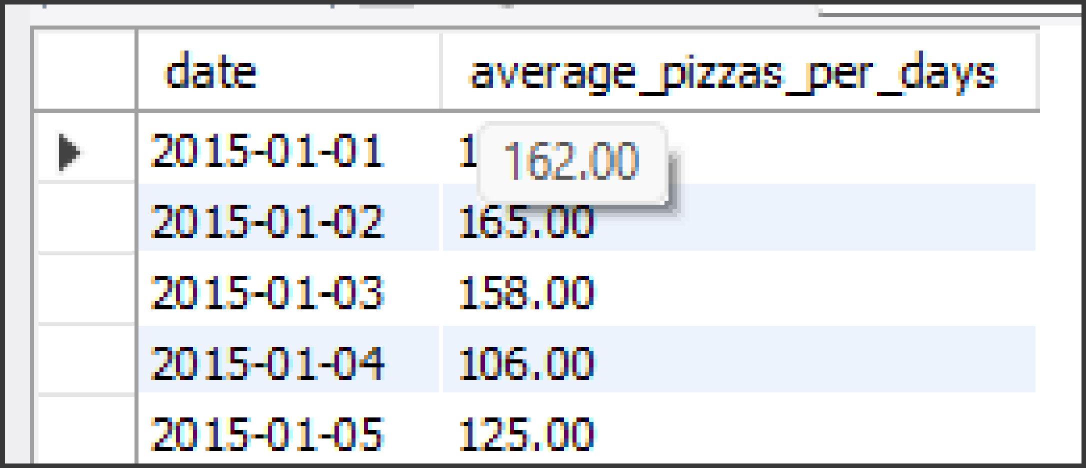
```
SELECT
    HOUR(time) AS Hour, COUNT(order_id) AS ouder_count
FROM
    orders
GROUP BY Hour;
```



Hour	ouder_count
11	1231
12	2520
13	2455
14	1472
15	1468

Average Pizzas per Day

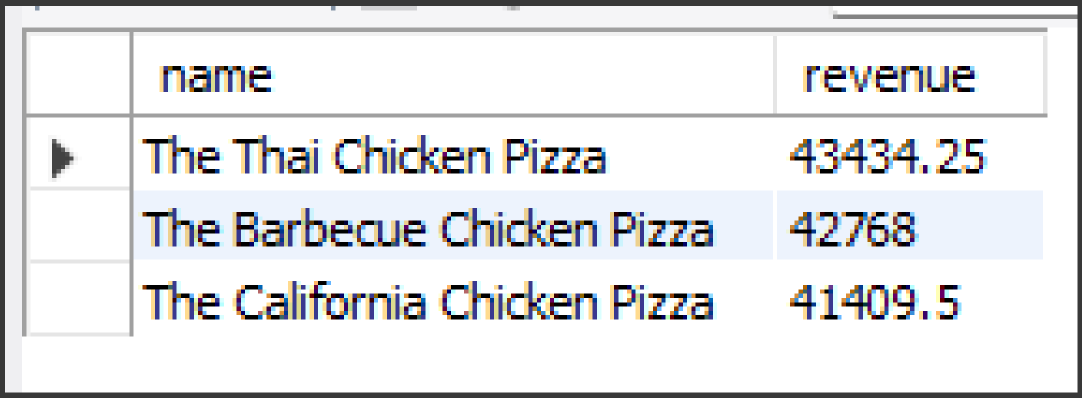
```
SELECT
    daily_totals.date,
    ROUND(AVG(total_pizzas), 2) AS
average_pizzas_per_days
FROM
    (SELECT
        orders.date, SUM(order_details.quantity) AS
total_pizzas
    FROM
        orders
    JOIN order_details ON orders.order_id =
order_details.order_id
    GROUP BY orders.date) AS daily_totals
GROUP BY daily_totals.date
LIMIT 5;
```



date	average_pizzas_per_days
2015-01-01	162.00
2015-01-02	165.00
2015-01-03	158.00
2015-01-04	106.00
2015-01-05	125.00

Top 3 Pizza Types by Revenue

```
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizzas.pizza_type_id =
pizza_types.pizza_type_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

Conclusion

The pizza ordering database efficiently tracks orders, details, and pizza types, offering insights into sales trends and customer preferences.

References

End-to-End SQL Project | SQL For Data Analysis Full Portfolio Project with Practical. Retrieved from https://www.youtube.com/@wscubetech