SQL PORTFOLIO Project

Domino Pizza Sales

Prepared by Haris Fayyaz

5 June, 2024

Introduction

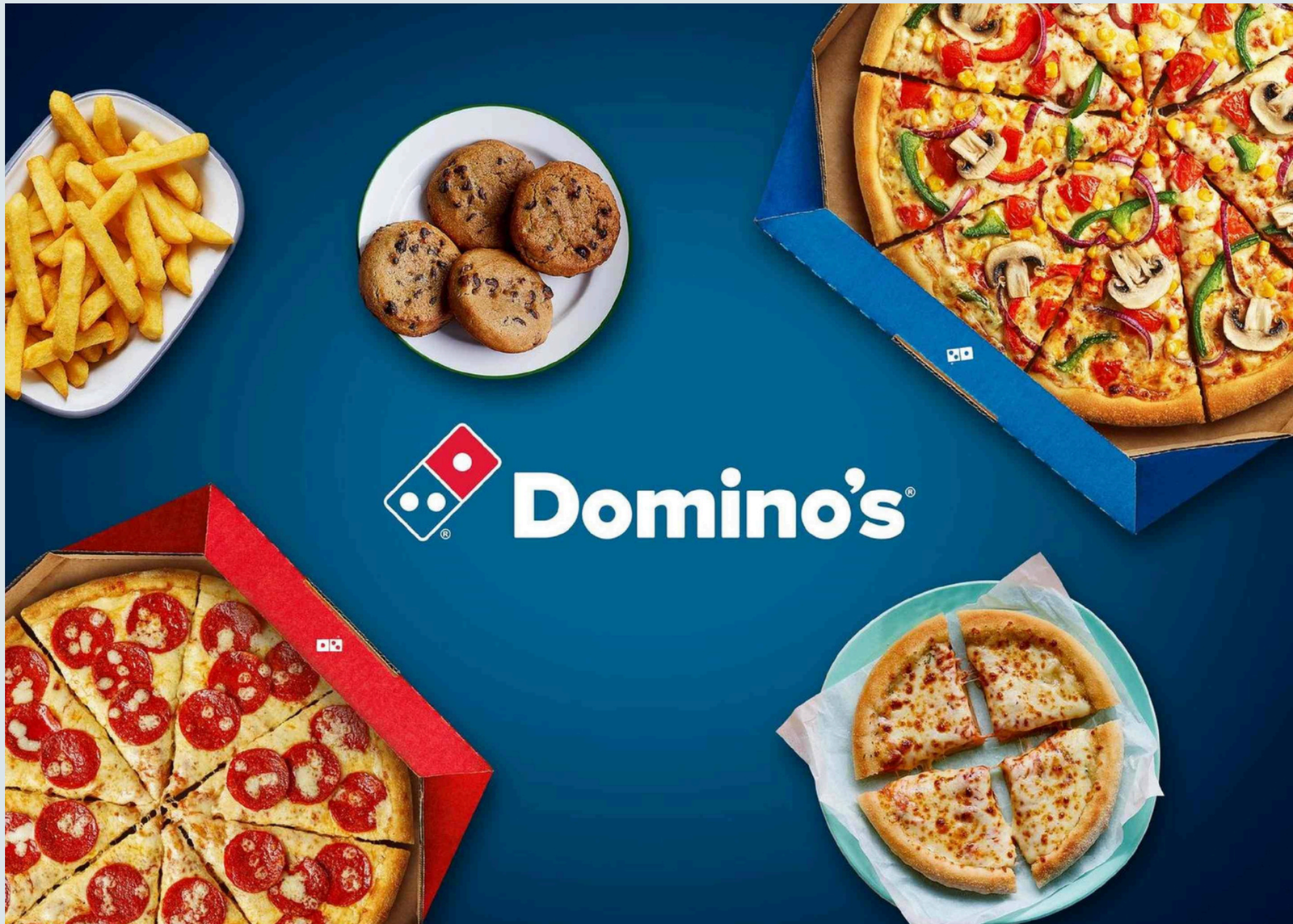
In this project, I will demonstrate my SQL skills by analyzing a pizza delivery dataset composed of four CSV files: pizzas, pizza types, orders, and order details. This dataset provides detailed information on various pizzas, their types, pricing, and order specifics. Through this analysis, I will extract key business insights, such as popular pizza choices, order trends, and revenue calculations. This project highlights my ability to handle real-world data, perform complex SQL queries, and derive actionable insights.

Creating a Schema..

```sql
CREATE DATABASE pizzahut;
SELECT *
FROM pizzahut.pizzas;
```

```sql
CREATE TABLE orders (
order_id INT NOT NULL,
order_date DATE NOT NULL,
order_time TIME NOT NULL,
PRIMARY KEY (order_id));
```

```sql
CREATE TABLE order_details (
order_details_id INT NOT NULL,
order_id INT NOT NULL,
pizza_id TEXT NOT NULL,
quantity INT NOT NULL,
PRIMARY KEY(order_details_id));
```



Retrieve the total number of orders placed?

```sql
SELECT
    COUNT(order_id) AS Total_Orders
FROM
    orders;
```

	Total_Orders
▶	21350

From the above we can observe that the total orders placed amounted to 21,350

Calculate the total revenue generated from pizza sales?

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

	Total_Sales
▶	817860.05

From the above we can observe that the total revenue generated amounted to 817,860.

Identify the highest priced pizza?

```sql
SELECT
    pizza_types.name AS Name, pizzas.price AS Price
FROM
    pizza_types
        INNER 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

From the above we can observe that the highest priced pizza is their ‘Greek Pizza’, priced at an amount of 35.95/pizza.

Identify the most common pizza size ordered?

```
SELECT
    pizzas.size AS Size,
    COUNT(order_details.order_details_id) AS Order_Count
FROM
    pizzas
        INNER JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY Order_count DESC;
```

	Size	Order_Count
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

From the above we can observe that the most common pizza from Dominoes is their large pizza followed by their medium pizza, then their small, followed by their extra large, and concluding to their extra extra large pizza.

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

```sql
SELECT
    pizza_types.name, SUM(order_details.quantity) AS Quantity
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        INNER JOIN
    order_details ON order_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

From the above we can observe that the most ordered pizza type is their ‘Classic Deluxe’, followed by their ‘Barbecue Pizza’, then their ‘Hawaiiian Pizza’, then their ‘Pepperoni Pizza’ concluding to their ‘Thai Chicken Pizza’.

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

```
SELECT
    pizza_types.category AS Category,
    SUM(order_details.quantity) AS Quantity
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        INNER JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
```

	Category	Quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

From the above we can observe that the most category observed was their ‘Classic’,then ‘Supreme’,then ‘Veggie’ and lastly ‘Chicken’.

Determine the distribution of orders by hour of the day?

```
SELECT
    HOUR(order_time) AS Hours, COUNT(order_id) AS Order_Count
FROM
    orders
GROUP BY Hours
ORDER BY Order_count DESC;
```

	Hours	Order_Count
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	21	1198
	22	663
	23	28
	10	8
	9	1

From the above we can observe that the most orders that were placed with in Dominoes was in the evening after 12 pm, till 11 pm at night. However the least orders are to be observed in the morning around 9-10 am.

Join relevant tables to find the category-wise distribution of pizzas?

```
SELECT
    category AS Category, COUNT(name) AS Count_Of_Name
FROM
    pizza_types
GROUP BY category
```

	Category	Count_Of_Name
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

From the above we can observe that the least category could be observed within Chicken, followed by ‘Classic’, then the most number to be within ‘Supreme’ and the ‘Veggie’.

Group the orders by date and calculate the average number of pizzas ordered per day?

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

	Average_Quantity_Pizza_Per_Day
▶	138

From the above we can observe that 138 pizzas were ordered on an average every day.

Determine the top 3 most ordered pizza types based on revenue?

```sql
SELECT
    pizza_types.name AS Name,
    SUM(order_details.quantity * pizzas.price) AS Revenue
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        INNER 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

From the above we can observe that based on revenues the most ordered pizza was their ‘Thai Chicken’, followed by their ‘Barbecue Chicken’ and lastly their ‘California Chicken Pizza’.

Calculate the percentage contribution of each pizza type to total revenue?

```sql
WITH TotalRevenue AS (
    SELECT ROUND(SUM(order_details.quantity * pizzas.price), 2) AS Total_Sales
    FROM order_details
    INNER JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id
)
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / TotalRevenue.Total_Sales * 100, 2) AS Percentage_Contribution
FROM pizza_types
INNER JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
INNER JOIN order_details ON order_details.pizza_id = pizzas.pizza_id
CROSS JOIN TotalRevenue
GROUP BY pizza_types.category, TotalRevenue.Total_Sales
ORDER BY Percentage_Contribution DESC;
```

	category	Percentage_Contribution
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

From the above we can observe that based on each pizza type to total revenue , ‘Classic’ had the most percentage share amounting to 26.91%, followed by their ‘Supreme’ category amounting to 25.46%, then their ‘Chicken’ category amounting to 23.96% , and lastly their ‘Veggie’ category amounting to 23.68%.

Analyze the cumulative revenue generated over time?

```sql
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
INNER JOIN pizzas
ON order_details.pizza_id=pizzas.pizza_id
INNER 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	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006
2015-12-14	785389.55
2015-12-15	787777
2015-12-16	790011.8
2015-12-17	791892.55
2015-12-18	794778.8500000001
2015-12-19	797083.05
2015-12-20	799187.9500000001
2015-12-21	801288.65
2015-12-22	803171.6
2015-12-23	805415.9
2015-12-24	807553.75
2015-12-26	809196.8
2015-12-27	810615.8
2015-12-28	812253
2015-12-29	813606.25
2015-12-30	814944.05
2015-12-31	817860.05

From the above we can observe that the cummulative revenues for Dominoes, summing to around 817,860 at the end.

Determine the top 3 most ordered pizza types based on revenue for each pizza category?

```sql
    SELECT
        pizza_types.category,
        pizza_types.name,
        SUM(order_details.quantity * pizzas.price) AS Revenue
    FROM pizza_types
    INNER JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    INNER JOIN order_details ON order_details.pizza_id = pizzas.pizza_id
    GROUP BY pizza_types.category, pizza_types.name
),
RankedPizzaRevenue AS (
    SELECT
        category,
        name,
        Revenue,
        RANK() OVER (PARTITION BY category ORDER BY Revenue DESC) AS `rank`
    FROM PizzaRevenue
)
SELECT
    category,
    name,
    Revenue,
    `rank`
FROM RankedPizzaRevenue
WHERE `rank` <= 3
ORDER BY category, `rank`;
```

	category	name	Revenue	rank
▶	Chicken	The Thai Chicken Pizza	43434.25	1
	Chicken	The Barbecue Chicken Pizza	42768	2
	Chicken	The California Chicken Pizza	41409.5	3
	Classic	The Classic Deluxe Pizza	38180.5	1
	Classic	The Hawaiian Pizza	32273.25	2
	Classic	The Pepperoni Pizza	30161.75	3
	Supreme	The Spicy Italian Pizza	34831.25	1
	Supreme	The Italian Supreme Pizza	33476.75	2
	Supreme	The Sicilian Pizza	30940.5	3
	Veggie	The Four Cheese Pizza	32265.70000000065	1
	Veggie	The Mexicana Pizza	26780.75	2
	Veggie	The Five Cheese Pizza	26066.5	3

From the above category we observe the ‘Thai Chicken Pizza’ to be the most ordered based on chicken category, ‘Classic Deluxe Pizza’ ordered the most in classic category ,”Spicy Italian ordered the most in supreme category and lastly the ‘Four Cheese’ pizza orderd in the veggie category.

Conclusion

This project delves into the power of data analysis within Domino's Pizza operations, revealing critical insights: a total revenue of $817,860 from 21,350 orders, with the highest-priced item being the ‘Greek Pizza’ at $35.95. Large pizzas emerged as the most popular size, while the ‘Classic Deluxe’ pizza led sales. Peak ordering times were between 12 PM and 11 PM, with an average of 138 pizzas ordered daily. Revenue was notably driven by pizzas like ‘Thai Chicken’, ‘Barbecue Chicken’, and ‘California Chicken’. These findings guide strategic decisions in product offerings, pricing, and marketing to enhance customer satisfaction and fuel business growth.

Thank you