Pizza Hut Sales Insights through SQL

PIZZAHUT

Introduction

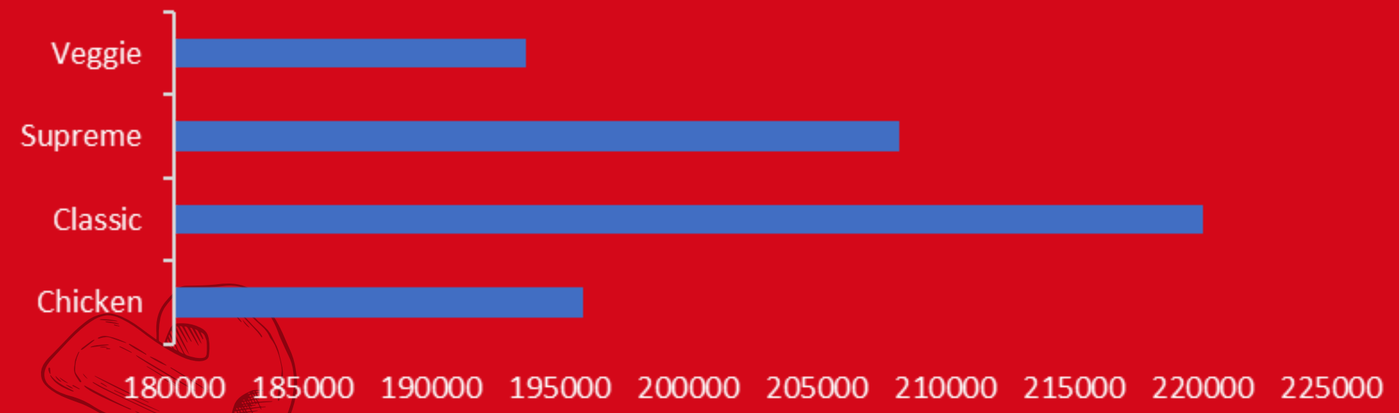
Hello, my name is Yash Bansode. In this project, I have utilized SQL queries to solve various questions related to Pizza Hut sales and gain useful insights.

PIZZAHUT

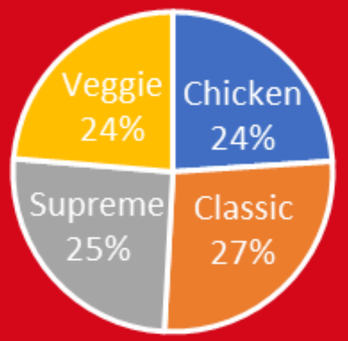
Market Analysis

Total Orders	Total Revenue	Avg Order Value
21350	₹ 8,17,860.05	₹ 38.31

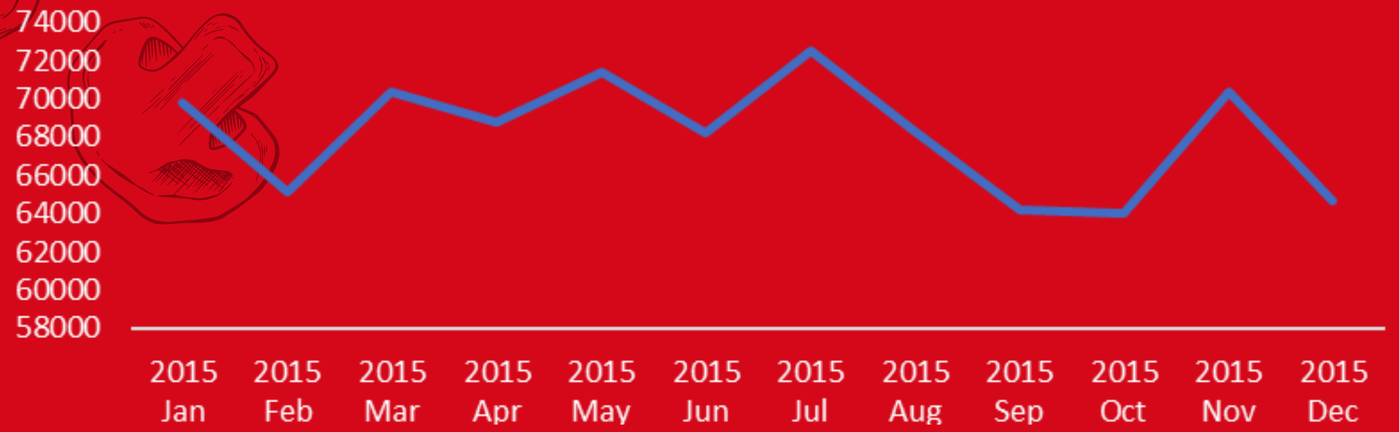
Top 5 Pizza Type By Quantity



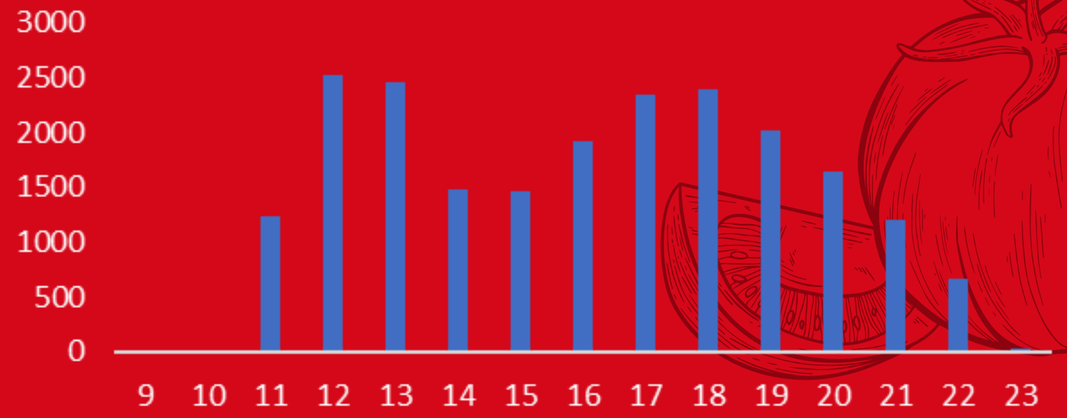
Revenue By Pizza Category



2015: Revenue Trend By Month



Orders by Hour of Day



PIZZAHUT

Retrieve the total number of orders placed

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

Result Grid

	total_orders
▶	21350

PIZZAHUT

Calculate the total revenue generated from pizza sales

```
SELECT SUM(od.quantity * p.price) AS total_revenue
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id;
```

Result Grid

total_revenue
817860.049999993

PIZZAHUT

Identify the highest-priced pizza

```
SELECT pizza_id, price
FROM pizzas
ORDER BY price DESC
LIMIT 1;
```

Result Grid | Filter Ro

	pizza_id	price
▶	the_greek_xxl	35.95

PIZZAHUT

Identify the most common pizza size ordered

```
SELECT p.size, SUM(od.quantity) AS total_ordered
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
GROUP BY p.size
ORDER BY total_ordered DESC
LIMIT 1;
```

Result Grid | Filter

size	total_ordered
L	18956

PIZZAHUT

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

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

Result Grid | Filter Rows:

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

PIZZAHUT

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

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

Result Grid | Filter R

category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

PIZZAHUT

Determine the distribution of orders by hour of the day

```
SELECT HOUR(o.time) AS order_hour, COUNT(o.order_id) AS total_orders
FROM orders o
GROUP BY HOUR(o.time)
ORDER BY order_hour;
```

Result Grid | Filter Rov

order_hour	total_orders
9	1
10	8
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

PIZZAHUT

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

```
SELECT pt.category, COUNT(p.pizza_id) AS total_pizzas
FROM pizzas p
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.category;
```

Result Grid | Filter Ro

category	total_pizzas
Chicken	18
Classic	26
Supreme	25
Veggie	27

PIZZAHUT

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

```
SELECT o.date, AVG(daily.total_quantity) AS avg_pizzas_per_day
FROM orders o
JOIN (
    SELECT order_id, SUM(quantity) AS total_quantity
    FROM order_details
    GROUP BY order_id
) daily ON o.order_id = daily.order_id
GROUP BY o.date;
```

Result Grid | Filter Rows:

date	avg_pizzas_per_day
2015-01-01	2.3478
2015-01-02	2.4627
2015-01-03	2.3939
2015-01-04	2.0385
2015-01-05	2.3148

PIZZAHUT

Determine the top 3 most ordered pizza types based on revenue

```
SELECT pt.name, SUM(od.quantity * p.price) AS total_revenue
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.name
ORDER BY total_revenue DESC
LIMIT 3;
```

Result Grid | Filter Rows:

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

PIZZAHUT

Calculate the percentage contribution of each pizza type to total revenue

```
SELECT pt.name,
       ROUND((SUM(od.quantity * p.price) /
              (SELECT SUM(od2.quantity * p2.price)
               FROM order_details od2
               JOIN pizzas p2 ON od2.pizza_id = p2.pizza_id) * 100), 2) AS revenue_percentage
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.name
ORDER BY revenue_percentage DESC;
```

Result Grid | Filter Rows:

name	revenue_percentage
The Thai Chicken Pizza	5.31
The Barbecue Chicken Pizza	5.23
The California Chicken Pizza	5.06
The Classic Deluxe Pizza	4.67

PIZZAHUT

Analyze the cumulative revenue generated over time

```
SELECT o.date,
       SUM(od.quantity * p.price) AS daily_revenue,
       SUM(SUM(od.quantity * p.price)) OVER (ORDER BY o.date) AS cumulative_revenue
FROM orders o
JOIN order_details od ON o.order_id = od.order_id
JOIN pizzas p ON od.pizza_id = p.pizza_id
GROUP BY o.date
ORDER BY o.date;
```

Result Grid | Filter Rows: | Export:

date	daily_revenue	cumulative_revenue
2015-01-01	2713.8500000000004	2713.8500000000004
2015-01-02	2731.8999999999996	5445.75
2015-01-03	2662.4	8108.15
2015-01-04	1755.4500000000003	9863.6
2015-01-05	2065.95	11929.55

PIZZAHUT

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

```sql
SELECT category, name, total_revenue
FROM (
    SELECT pt.category, pt.name,
           SUM(od.quantity * p.price) AS total_revenue,
           RANK() OVER (PARTITION BY pt.category ORDER BY SUM(od.quantity * p.price) DESC) AS rank_in_category
    FROM order_details od
    JOIN pizzas p ON od.pizza_id = p.pizza_id
    JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
    GROUP BY pt.category, pt.name
) ranked
WHERE rank_in_category <= 3
ORDER BY category, total_revenue DESC;
```

Result Grid | Filter Rows: | Export:

category	name	total_revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5

PIZZAHUT

Thank you!