Pizza Sales Queries

Total number of orders

```
SELECT
    COUNT(*) as total_orders
FROM
    orders;
```

total_orders
21350

Total revenue generated from pizza sales.

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

total_sales
484269

Average Cost of order

```
SELECT
    ROUND(SUM(pizzas.price * order_details.quantity) / (SELECT
                    COUNT(DISTINCT order_id)
                FROM
                    orders),
            2) AS avg_order_value
FROM
    pizzas
        INNER JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id;
```

avg_order_value
22.68

Highest Price Pizza

```
SELECT
    name, price
FROM
```

```
    pizza_types
        INNER JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
WHERE
    price IN (SELECT
            MAX(price)
        FROM
            pizzas);
```

name	price
The Greek Pizza	35.95

Top category - show percent orders of each category

```
SELECT
    category,
    ROUND(SUM(order_details.quantity) / (SELECT
                    COUNT(order_id)
                FROM
                    order_details) * 100,
            2) AS percent
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        INNER JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY category
ORDER BY percent DESC;
```

category	percent
Classic	30.40
Supreme	24.74
Veggie	24.24
Chicken	22.59

Top size - show percent orders of each size

```
SELECT
    size,
    ROUND(COUNT(order_details_id) / (SELECT
                    COUNT(*)
                FROM
                    order_details) * 100,
            2) as percent
FROM
    pizzas
        INNER JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY size
ORDER BY percent DESC;
```

size	percent
L	38.31
M	31.69
S	28.76
XL	1.19
XXL	0.06

Top 10 most ordered pizza types by count

```
SELECT
    pizza_types.name,
    ROUND(COUNT(order_details_id) / (SELECT
                    COUNT(*)
                FROM
                    order_details) * 100,
            2) AS percent,
      ROUND(SUM(price * quantity),2) as revenue
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        INNER JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.name
ORDER BY percent DESC
LIMIT 10;
```

name	percent	revenue
The Barbecue Chicken Pizza	5.04	26034.75
The Classic Deluxe Pizza	4.88	22111.5
The Pepperoni Pizza	4.86	17769.25
The Hawaiian Pizza	4.75	18630.5
The Thai Chicken Pizza	4.67	25224.75
The California Chicken Pizza	4.67	24183
The Spicy Italian Pizza	3.95	21202
The Four Cheese Pizza	3.84	19264.8
The Italian Supreme Pizza	3.83	19817.25
The Southwest Chicken Pizza	3.80	20136.5

Top 3 most ordered pizza types based on revenue for each pizza category

```
SELECT category, name, revenue
FROM
(SELECT category, name, revenue, RANK() OVER(PARTITION BY category
ORDER BY revenue DESC) as rnk
FROM
(SELECT
    category, name, SUM(price * quantity) 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 category , name) as a) as b
```

```
WHERE rnk <=3;
```

category	name	revenue
Chicken	The Barbecue Chicken Pizza	26034.75
Chicken	The Thai Chicken Pizza	25224.75
Chicken	The California Chicken Pizza	24183
Classic	The Classic Deluxe Pizza	22111.5
Classic	The Hawaiian Pizza	18630.5
Classic	The Pepperoni Pizza	17769.25
Supreme	The Spicy Italian Pizza	21202
Supreme	The Italian Supreme Pizza	19817.25
Supreme	The Sicilian Pizza	17925.25
Veggie	The Four Cheese Pizza	19264.8
Veggie	The Five Cheese Pizza	16021
Veggie	The Mexicana Pizza	15735.5

Average pizzas ordered per day

```
SELECT
    ROUND(AVG(quantity), 0) as average_order_quantity_per_day
FROM
    (SELECT
        o.order_date, SUM(quantity) AS quantity
    FROM
        orders o
    INNER JOIN order_details od ON o.order_id = od.order_id
    GROUP BY order_date) AS ordered_quantity;
```

average_order_quantity_per_day
138

Average order trend over the week

```
SELECT
    day_name, ROUND(AVG(count_of_orders),0) as avg_count_of_orders
FROM
    (SELECT
        order_date,
            DAYNAME(order_date) AS day_name,
            COUNT(order_id) AS count_of_orders
    FROM
        orders
    GROUP BY order_date , day_name) AS orders_by_day
GROUP BY day_name;
```

day_name	avg_count_of_orders
Thursday	62
Friday	71
Saturday	61
Sunday	50
Monday	58
Tuesday	57
Wednesday	58

Peak Order Hours

```sql
SELECT
    HOUR(order_time) AS order_hour,
    COUNT(order_id) AS order_count
FROM
    orders
GROUP BY order_hour
ORDER BY order_hour;
```

order_hour	order_count
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

Category Revenue Vs Volume

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

category	revenue	percent
Classic	129398.4	29.82
Supreme	123790.8	24.26
Veggie	115919.55	23.79
Chicken	115160.25	22.12

Monthly Sales Trend

```
SELECT
    MONTH(order_date) AS order_month,
    ROUND(SUM(p.price * od.quantity), 2) AS monthly_sales
FROM
    Orders
INNER JOIN
    order_details ON orders.order_id = order_details.order_id
INNER JOIN
    Pizzas ON pizzas.pizza_id = order_details.pizza_id
GROUP BY order_month
ORDER BY monthly_sales DESC;
```

order_month	order_count
7	1935
5	1853
1	1845
8	1841
3	1840
4	1799
11	1792
6	1773
2	1685
12	1680
9	1661
10	1646

Cumulative revenue generated over time.

```
SELECT order_date,
 ROUND(SUM(revenue) Over(ORDER BY order_date),2) as cumulative
FROM
(SELECT
    order_date, ROUND(SUM(price * quantity), 2) AS revenue
FROM
    orders o
        INNER JOIN
    order_details od ON o.order_id = od.order_id
        INNER JOIN
    pizzas p ON p.pizza_id = od.pizza_id
GROUP BY order_date) as sales;
```

order_date	cumulative
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
2015-01-09	21526.4
2015-01-10	23990.35
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.3
2015-01-14	32358.7
2015-01-15	34343.5
2015-01-16	36937.65
2015-01-17	39991.75