```sql
-- Retrieve the total number of orders placed.
USE pizzahut;
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;

```

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

total_orders
21350

```sql
-- Calculate the total revenue generated from pizza sales.

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

```

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

total_sales
817860.05

```sql
-- Identify the highest-priced pizza

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

```

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

name	price
The Greek Pizza	35.95

```sql
-- Identify the most common pizza size ordered

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

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

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

```sql
-- List the top 5 most orderd pizza types along with their quantities.

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

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

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

```sql
-- JOIN the nacessary tables to find the
-- total quantity of each pizza category orderd.

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

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

category	Quantity
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

```sql
-- Detetmine the distrubuiton of orders by hour of the day .

SELECT HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM orders
group by hour ;
```

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

hour	order_count
11	1231
12	2520
13	2455

```sql
-- Join Relevant tables to find the categry - wise distrubution of piz

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

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

category	name
Chicken	6
Classic	8
Supreme	9
Veggie	9

```sql
-- Group the orders by date and calculate the average number of
-- pizzas orders per day.

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

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

avg_pizza_ordered_per_day
138

```sql
-- Detrmine the top 3 most ordered pizza types based on revenue.
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC  LIMIT 3;
```

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

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

```sql
-- Calculate the percentage contribution of each pizza type  to total revenue
SELECT pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
          ROUND (SUM(order_details.quantity * pizzas.price),2) AS total
        FROM order_details
              JOIN
            pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,2) AS revenue
FROM pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category;
```

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

category	revenue
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96

```sql
-- Analyze the cumulative revanue generated over time.

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 orders JOIN order_details
ON orders.order_id = order_details.order_id
JOIN pizzas
ON order_details.pizza_id = pizzas.pizza_id
GROUP BY orders.order_date) AS sales ;

```

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

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15

```sql
-- Determine the top 3 most ordered pizzas 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 rn FROM
(SELECT pizza_types.category,pizza_types.name,
SUM(order_details.quantity * pizzas.price) AS revenue
FROM pizza_types JOIN pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id
JOIN order_details
ON order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category,pizza_types.name) AS A)AS B
WHERE rn <= 3;
```

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

category	name	revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5
Classic	The Classic Deluxe Pizza	38180.5
Classic	The Hawaiian Pizza	32273.25