# Query For Pizza Sales Report

Q **Retrieve the total number order placed**
Ans:

```
select count(order_id) from orders;
```

Q **Retrieve a total revenue genrated from pizza sales**
Ans:

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

Q **Identify the highest-priced pizza.**

Ans:

```
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;
```

Q **Identify the most common pizza size ordered.**

Ans:

```
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;
```

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

Ans:

```
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;
```

Q **Join the necessary tables to find the total quantit**

Ans:

```
SELECT
    pizza_types.category,
    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.category;
```

Q **Determine the distribution of orders by hour of the day.**

Ans:

```
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS
order_id
FROM
    orders
GROUP BY HOUR(order_time);
```

Q **Join relevant tables to find the category-wise distribution of pizzas.**
Ans:

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

Q **Group the orders by date and calculate the average number of pizzas ordered per day.**
Ans:

```
SELECT
   ROUND(AVG(quantity), 0) as avd_pizza_order_perDay
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;
```

Q **Determine the top 3 most ordered pizza types based on revenue.**

Ans:

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

Q **Calculate the percentage contribution of each pizza type to total revenue.**

Ans:

```
SELECT
   pizza_types.category,
   ROUND(SUM(pizzas.price * order_details.quantity) /
(SELECT
               SUM(order_details.quantity * pizzas.price)
            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
ORDER BY revenue DESC;
```

Q **Analyze the cumulative revenue generated over time.**
Ans:

```
select order_date,
sum(revenue) over (order by order_date) as
cm_revenue
from
(select orders.order_date,
round(sum(order_details.quantity * pizzas.price),2) as
revenue
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

Q **Determine the top 3 most ordered pizza types based on revenue for each pizza category.**
Ans:

```
select 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;
```