PROJECT NAME : “Pizza Sales”

By

AMRESH KUMAR YADAV

INTRODUCTION

- **I am amresh kumar yadav .In this project. I have utilize SQL Query to solve Questions related to pizza sales.**

- SQL (Structured Query Language) is a programming language designed for managing data in a relational database. It's been around since the 1970s and is the most common method of accessing data in databases today. SQL has a variety of functions that allow its users to read, manipulate, and change data.

Q1). RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
select count(order_id ) as total_orders from orders
```

Q2). CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
select round(sum(order_details.quantity*pizzas.price),2) as
total_salesfrom order_details join pizzas
on pizzas.pizza_id=order_details.pizza_id
```

Q3). IDENTIFY THE HIGHEST-PRICED PIZZA WITH NAME.

```
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
```

Q4). 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_idgroup by pizzas.size

order by order_count desc;
```

Q5). LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
select pizza_types.name , sum(order_details.quantity) as
quantity
from pizza_types join.pizza_type_id
group by pizza_types.name
order by quantity
limit 5;
```

Q6). DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
select hour(order_time) as hour, count(order_id) as order_count

from orders

group by hour(order_time);
```

Q7). JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
select category , count(name)

from pizza_types

group by category;
```

Q8). GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
select round(avg(quantity),0)
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;
```

Q9). ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

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


THANK
YOU !