**SQL Queries to fetch data from Pizza Sales datasets.**

**Q1. Retrieve the total number of orders placed.**

**Ans**. **select \* from orders;**

**select count(Order\_ID) as Total\_Orders from orders;**

**Q2**. **Calculate the total revenue generated from pizza sales.**

**Ans**. **select**

**round(sum(orders\_details.Quantity \* pizzas.price),2) as Total\_Revenue**

**from orders\_details join pizzas**

**on pizzas.pizza\_id = orders\_details.Pizza\_ID**

**Q3**. **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;**

**Q4. Identify the most common pizza size ordered.**

**Ans. select pizzas.size, count(orders\_details.Order\_details\_ID) as Order\_Count**

**from pizzas join orders\_details**

**on pizzas.pizza\_id = orders\_details.Pizza\_ID**

**group by pizzas.size order by Order\_Count desc;**

**Q5. List the top 5 most ordered pizza types along with their quantities.**

**Ans**. **select pizza\_types.name,**

**sum(orders\_details.Quantity) as Quantity**

**from pizza\_types join pizzas**

**on pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id**

**join orders\_details**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**group by pizza\_types.name order by Quantity desc limit 5;**

**Q6. Join the necessary tables to find the total quantity of each pizza category ordered.**

**Ans. select pizza\_types.category,**

**sum(orders\_details.Quantity) as Quantity**

**from pizza\_types join pizzas**

**on pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id**

**join orders\_details**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**group by pizza\_types.category order by Quantity desc;**

**#sum use kiya hai toh group by lgana is must.**

**Q7. Determine the distribution of orders by hour of the day.**

**Ans. Select hour(order\_time) as Hour, count(Order\_ID) as Order\_Count from orders**

**group by hour(order\_time);**

**Q8. Join relevant tables to find the category-wise distribution of pizzas.**

**Ans. select category , count(name) from pizza\_types**

**group by category;**

**Q9. Group the orders by date and calculate the average number of pizzas ordered per day.**

**Ans. Select round(avg(Quantity), 0) as average\_pizzas\_ordered from**

**(select orders.Order\_date, sum(orders\_details.Quantity) as Quantity**

**from orders join orders\_details**

**on orders.Order\_ID = orders\_details.Order\_ID**

**group by orders.Order\_date) as Orders\_Quanity ;**

**Q10. Determine the top 3 most ordered pizza types based on revenue.**

**Ans.** **select pizza\_types.name,**

**Sum(orders\_details.Quantity \* pizzas.price) as Revenue**

**from pizza\_types join pizzas**

**on pizzas.pizza\_type\_id = pizza\_types.pizza\_type\_id**

**join orders\_details**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**group by pizza\_types.name order by Revenue desc limit 4;**

**Q11. Calculate the percentage contribution of each pizza type to total revenue.**

**Ans. select pizza\_types.category,**

**Round((sum(orders\_details.quantity \* pizzas.price) / (SELECT**

**ROUND(SUM(orders\_details.Quantity \* pizzas.price),**

**2) AS Total\_Revenue**

**FROM**

**orders\_details**

**JOIN**

**pizzas ON pizzas.pizza\_id = orders\_details.Pizza\_ID)) \* 100 , 2) as Revenue**

**from pizza\_types join pizzas**

**on pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id**

**join orders\_details**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**group by pizza\_types.category order by Revenue desc;**

**Q12. Analyze the cumulative revenue generated over time.**

**Ans. select order\_date,**

**sum(revenue) over ( order by order\_date) as Cum\_Revenue**

**from**

**(select orders.Order\_date,**

**sum(orders\_details.Quantity \* pizzas.price ) as Revenue**

**from orders\_details join pizzas**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**join orders**

**on orders.Order\_ID = orders\_details.Order\_ID**

**group by orders.Order\_date) as sales;**

**Q13. 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((orders\_details.Quantity) \* pizzas.price) as Revenue**

**from pizza\_types join pizzas**

**on pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id**

**join orders\_details**

**on orders\_details.Pizza\_ID = pizzas.pizza\_id**

**group by pizza\_types.category, pizza\_types.name) as A) as B**

**where RN<=3 ;**

**#note: Pehle puri value nikali then unko ranking di since hum rank ko directly where me use nhi kar sakte to uska hi ek sub query bnaya or main table ke andar likha ki sirf name and revenue chahiye wo bhi wha se jha rn<=3.**

**Q14**. To upload datasets to our workbench?

**Ans**. **We have create database by giving database name.**

**Create Database Dominoz;**

**Q15.**To import tables in our database?

**Ans**. **Right click on tables—select table data import wizard—Select data.**

**Q16**. For large files having large data we have to create table? How? (becau takes time to import)

**Ans**. **We can create table with our specific data type.**

**Use Dominoz;**

**Create Table Orders (**

**Order\_ID int not null,**

**Order\_Date date not null,**

**Order\_Time time not null, primary key (Order\_ID));**

**Q17**. Define Pizza ID?

**Ans**. **Pizza ID differentiate between sizes of pizzas i.e., small, medium, large.**

**#note: To round off our numeric answer use round. # To beautify query CTRL+B.**