Queries Questions on whole database

Easy Level Queries (1-10)

1. Find all customers in the system.

```
SELECT * FROM Customer;
```

2. Get the list of products with their prices.

```
SELECT Name, Price FROM Product;
```

3. Find all orders placed by a customer with CustomerID = 10.

```
SELECT * FROM Orders WHERE CustomerID = 10;
```

4. List all employees in the system.

```
SELECT * FROM Employee;
```

5. Get the details of products that have a price greater than 50.

```
SELECT * FROM Product WHERE Price > 50;
```

6. Find all activities of type 'Sale'.

```
SELECT * FROM Activity WHERE Type = 'Sale';
```

7. Get the total amount of the order with OrderID = 15.

```
SELECT TotalAmount FROM Orders WHERE OrderID = 15;
```

8. List all employees who are in the position of 'Manager'.

```
SELECT * FROM Employee WHERE Position = 'Manager';
```

9. Find all order details where the quantity is greater than 2.

```
SELECT * FROM OrderDetail WHERE Quantity > 2;
```

10. Get all activities that occurred on '2024-01-01'.

```sql
SELECT * FROM Activity WHERE Date = '2024-01-01';
```

Intermediate Level Queries (11-20)

11. Find the total amount spent by each customer.

```sql
SELECT CustomerID, SUM(TotalAmount) AS TotalSpent

FROM Orders

GROUP BY CustomerID;
```

12. Get the details of orders with the highest total amount.

```sql
SELECT * FROM Orders WHERE TotalAmount = (SELECT MAX(TotalAmount) FROM Orders);
```

13. Find all products that are part of orders with a total amount greater than 100.

```sql
SELECT DISTINCT p.Name

FROM Product p

JOIN OrderDetail od ON p.ProductID = od.ProductID

JOIN Orders o ON od.OrderID = o.OrderID

WHERE o.TotalAmount > 100;
```

14. List the employees who have activities scheduled after '2024-02-01'.

```sql
SELECT DISTINCT e.Name

FROM Employee e

JOIN Activity a ON a.EmployeeID = e.EmployeeID

WHERE a.Date > '2024-02-01';
```

15. Get the total quantity of each product sold.

```sql
SELECT p.Name, SUM(od.Quantity) AS TotalQuantity
```

```
FROM Product p
JOIN OrderDetail od ON p.ProductID = od.ProductID
GROUP BY p.Name;
```

16. Find all customers who have made an order with a total amount greater than 500.

```
SELECT DISTINCT c.Name
FROM Customer c
JOIN Orders o ON c.CustomerID = o.CustomerID
WHERE o.TotalAmount > 500;
```

17. List all employees along with the activities they have participated in.

```
SELECT e.Name AS EmployeeName, a.Type AS ActivityType
FROM Employee e
JOIN Activity a ON e.EmployeeID = a.EmployeeID;
```

18. Find the most expensive product in the product list.

```
SELECT * FROM Product WHERE Price = (SELECT MAX(Price) FROM Product);
```

19. Get the number of orders placed in the year 2024.

```
SELECT COUNT(*) AS OrdersIn2024
FROM Orders
WHERE YEAR(OrderDate) = 2024;
```

20. Find the total number of products sold for each order.

```
SELECT od.OrderID, SUM(od.Quantity) AS TotalProductsSold
FROM OrderDetail od
GROUP BY od.OrderID;
```

Hard Level Queries (21-30)

21. Get the total sales for each product type (e.g., based on the description).

```
SELECT p.Description, SUM(od.Quantity * od.UnitPrice) AS TotalSales
FROM Product p
JOIN OrderDetail od ON p.ProductID = od.ProductID
GROUP BY p.Description;
```

22. Find the customers who have ordered more than three times in the past 30 days.

```
SELECT c.Name
FROM Customer c
JOIN Orders o ON c.CustomerID = o.CustomerID
WHERE o.OrderDate > CURDATE() - INTERVAL 30 DAY
GROUP BY c.CustomerID
HAVING COUNT(o.OrderID) > 3;
```

23. List the employees who have worked on more than 5 different activities.

```
SELECT e.Name
FROM Employee e
JOIN Activity a ON e.EmployeeID = a.EmployeeID
GROUP BY e.EmployeeID
HAVING COUNT(DISTINCT a.ActivityID) > 5;
```

24. Get the total amount spent by each customer, including their email.

```
SELECT c.Name, c.Email, SUM(o.TotalAmount) AS TotalSpent
```

```
FROM Customer c
JOIN Orders o ON c.CustomerID = o.CustomerID
GROUP BY c.CustomerID;
```

25. Find all employees who have activities scheduled at the same time.

```
SELECT a.Time, GROUP_CONCAT(e.Name) AS EmployeeNames
FROM Activity a
JOIN Employee e ON e.EmployeeID = a.EmployeeID
GROUP BY a.Time
HAVING COUNT(e.EmployeeID) > 1;
```

26. Get the average price of all products ordered by each customer.

```
SELECT c.Name AS CustomerName, AVG(p.Price) AS AvgProductPrice
FROM Customer c
JOIN Orders o ON c.CustomerID = o.CustomerID
JOIN OrderDetail od ON o.OrderID = od.OrderID
JOIN Product p ON od.ProductID = p.ProductID
GROUP BY c.CustomerID;
```

27. Find the product that was ordered the most in terms of quantity.

```
SELECT p.Name, SUM(od.Quantity) AS TotalOrdered
FROM Product p
JOIN OrderDetail od ON p.ProductID = od.ProductID
GROUP BY p.ProductID
ORDER BY TotalOrdered DESC
```

```
LIMIT 1;
```

28. Get the total sales for the year 2024 broken down by month.

```
SELECT MONTH(o.OrderDate) AS Month, SUM(o.TotalAmount) AS TotalSales

FROM Orders o

WHERE YEAR(o.OrderDate) = 2024

GROUP BY MONTH(o.OrderDate);
```

29. List all orders and their details where the total amount exceeds 1000 and the order contains the product with ProductID = 5.

```
SELECT o.OrderID, o.TotalAmount, p.Name AS ProductName, od.Quantity, od.UnitPrice

FROM Orders o

JOIN OrderDetail od ON o.OrderID = od.OrderID

JOIN Product p ON od.ProductID = p.ProductID

WHERE o.TotalAmount > 1000 AND od.ProductID = 5;
```

30. Find the employee who participated in the most activities related to 'Sale'.

```
SELECT e.Name, COUNT(a.ActivityID) AS ActivityCount

FROM Employee e

JOIN Activity a ON e.EmployeeID = a.EmployeeID

WHERE a.Type = 'Sale'

GROUP BY e.EmployeeID

ORDER BY ActivityCount DESC

LIMIT 1;
```

CRM query questions On tables

1. Customer Table(105 records)

1. How many customers are there in each city?

```
SELECT city, COUNT(*) AS customer_count
FROM Customer
GROUP BY city;
```

2. Which email domains (e.g., example.com) are most used by customers?

```
SELECT SUBSTRING_INDEX(email, '@', -1) AS email_domain, COUNT(*) AS
domain_count
FROM Customer
GROUP BY email_domain
ORDER BY domain_count DESC;
```

3. Are there any customers with duplicate phone numbers?

```
SELECT phone, COUNT(*) AS phone_count
FROM Customer
GROUP BY phone
HAVING phone_count > 1;
```

4. Who has the longest name among all customers?

```
SELECT name
FROM Customer
ORDER BY LENGTH(name) DESC
LIMIT 1;
```

5. How many customers have names starting with 'S'?

```
SELECT COUNT(*) AS customers_starting_with_S
FROM Customer
WHERE name LIKE 'S%';
```

6. How many customers are from "MG Road"?

```
SELECT COUNT(*) AS customers_from_MG_Road
FROM Customer
WHERE address LIKE '%MG Road%';
```

7. How many unique phone numbers are there in the table?

```
SELECT COUNT(DISTINCT phone) AS unique_phone_numbers
```

```
FROM Customer;
```

8. What is the most common first name among customers?

```
SELECT first_name, COUNT(*) AS name_count
FROM Customer
GROUP BY first_name
ORDER BY name_count DESC
LIMIT 1;
```

9. How many customers have names longer than 10 characters?

```
SELECT COUNT(*) AS customers_with_long_names
FROM Customer
WHERE LENGTH(name) > 10;
```

10. What are the names of customers in alphabetical order?

```
SELECT name
FROM Customer
ORDER BY name;
```

11. How many customers live in addresses containing the word "Road"?

```
SELECT COUNT(*) AS customers_with_Road_in_address
FROM Customer
WHERE address LIKE '%Road%';
```

12. Are there any customers without an email address?

```
SELECT COUNT(*) AS customers_without_email
FROM Customer
WHERE email IS NULL OR email = '';
```

13. What is the distribution of customers by name length?

```
SELECT LENGTH(name) AS name_length, COUNT(*) AS customer_count
FROM Customer
GROUP BY name_length;
```

14. Which city has the highest number of customers?

```
SELECT city, COUNT(*) AS customer_count
FROM Customer
GROUP BY city
ORDER BY customer_count DESC
LIMIT 1;
```

15. How many customers share the same first name?

```
SELECT first_name, COUNT(*) AS first_name_count
FROM Customer
GROUP BY first_name
HAVING first_name_count > 1;
```

16. Are there any phone numbers with invalid lengths?

```
SELECT phone
FROM Customer
WHERE LENGTH(phone) != 10;  -- Assuming valid phone number length is 10
```

17. How many customers have Gmail addresses?

```
SELECT COUNT(*) AS customers_with_gmail
FROM Customer
WHERE email LIKE '%@gmail.com';
```

18. What percentage of customers are from each city?

```
SELECT city,
      (COUNT(*) / (SELECT COUNT(*) FROM Customer)) * 100 AS city_percentage
FROM Customer
GROUP BY city;
```

19. How many customers live in cities starting with the letter 'B'?

```
SELECT COUNT(*) AS customers_in_cities_starting_with_B
FROM Customer
WHERE city LIKE 'B%';
```

20. Are there customers with names that contain only one word?

```
SELECT COUNT(*) AS customers_with_one_word_name
FROM Customer
WHERE name NOT LIKE '% %';  -- Names without spaces
```

2. Product Table(30 records)

1. What is the average price of all products in the inventory?

```
SELECT AVG(price) AS average_price
FROM Product;
```

2. Which products have the highest price, and what is their price?

```
SELECT name, price
FROM Product
WHERE price = (SELECT MAX(price) FROM Product);
```

3. What are the total sales (i.e., sum of the prices) of all products in the inventory?

```
SELECT SUM(price) AS total_sales
FROM Product;
```

4. How many products have a price greater than ₹5000?

```
SELECT COUNT(*) AS products_above_5000
FROM Product
WHERE price > 5000;
```

5. Which products have a price less than ₹1000, and what are their names?

```
SELECT name, price
FROM Product
WHERE price < 1000;
```

6. What is the price range of products available in the inventory?

```
SELECT MIN(price) AS lowest_price, MAX(price) AS highest_price
FROM Product;
```

7. How many products have no description provided?

```
SELECT COUNT(*) AS products_without_description
FROM Product
WHERE description IS NULL OR description = '';
```

8. What is the most common price for the products in the inventory?

```
SELECT price, COUNT(*) AS price_count
FROM Product
GROUP BY price
ORDER BY price_count DESC
LIMIT 1;
```

9. How many products are priced within a specific range, say ₹1000 to ₹3000?

```
SELECT COUNT(*) AS products_in_price_range
FROM Product
WHERE price BETWEEN 1000 AND 3000;
```

10. Which products have a price that is greater than the average price of all products?

```
SELECT name, price
FROM Product
WHERE price > (SELECT AVG(price) FROM Product);
```

11. List products with a description containing a specific keyword (e.g., "luxury").

```
SELECT name, description
FROM Product
WHERE description LIKE '%luxury%';
```

12. What is the total number of unique product names in the inventory?

```
SELECT COUNT(DISTINCT name) AS unique_products
FROM Product;
```

13. Which product has the longest description?

```
SELECT name, description
FROM Product
ORDER BY LENGTH(description) DESC
LIMIT 1;
```

14. Which products have a description length greater than a specific number of characters (e.g., 200 characters)?

```
SELECT name, description
FROM Product
WHERE LENGTH(description) > 200;
```

15. How many products fall into each price range (e.g., ₹0-₹1000, ₹1001-₹3000, ₹3001+)?

```
SELECT
   CASE
      WHEN price BETWEEN 0 AND 1000 THEN '₹0-₹1000'
      WHEN price BETWEEN 1001 AND 3000 THEN '₹1001-₹3000'
      ELSE '₹3001+'
   END AS price_range,
   COUNT(*) AS product_count
FROM Product
GROUP BY price_range;
```

16. What are the top 5 most expensive products in the inventory?

```
SELECT name, price
FROM Product
ORDER BY price DESC
LIMIT 5;
```

17. How many products have a price within a specific budget (e.g., ₹500-₹1500)?

```
SELECT COUNT(*) AS products_in_budget
FROM Product
WHERE price BETWEEN 500 AND 1500;
```

18. Which product has the lowest price, and what is its description?

```
SELECT name, price, description
FROM Product
WHERE price = (SELECT MIN(price) FROM Product);
```

19. What is the total number of products priced above ₹2000?

```
SELECT COUNT(*) AS products_above_2000
FROM Product
WHERE price > 2000;
```

20. What is the distribution of product prices (e.g., how many products are in each price category)?

```
SELECT
   CASE
      WHEN price BETWEEN 0 AND 1000 THEN '₹0-₹1000'
      WHEN price BETWEEN 1001 AND 3000 THEN '₹1001-₹3000'
      ELSE '₹3001+'
   END AS price_category,
   COUNT(*) AS product_count
FROM Product
GROUP BY price_category;
```

3. Orders Table(300 records)

1. What is the total number of orders placed in the system?

```
SELECT COUNT(*) AS total_orders
FROM Orders;
```

2. What is the total sales amount from all orders?

```
SELECT SUM(TotalAmount) AS total_sales
FROM Orders;
```

3. Which customer has placed the most orders?

```
SELECT customer_id, COUNT(*) AS order_count
```

```
FROM Orders
GROUP BY customer_id
ORDER BY order_count DESC
LIMIT 1;
```

4. What is the average order value (TotalAmount) for all orders?

```
SELECT AVG(TotalAmount) AS average_order_value
FROM Orders;
```

5. Which month had the highest total sales?

```
SELECT EXTRACT(MONTH FROM order_date) AS month, SUM(TotalAmount) AS
total_sales
FROM Orders
GROUP BY month
ORDER BY total_sales DESC
LIMIT 1;
```

6. How many orders were placed in each month?

```
SELECT EXTRACT(MONTH FROM order_date) AS month, COUNT(*) AS orders_count
FROM Orders
GROUP BY month
ORDER BY month;
```

7. What is the total number of orders placed by each customer?

```
SELECT customer_id, COUNT(*) AS orders_count
FROM Orders
GROUP BY customer_id;
```

8. Which customer has spent the most money in total on all their orders?

```
SELECT customer_id, SUM(TotalAmount) AS total_spent
FROM Orders
GROUP BY customer_id
ORDER BY total_spent DESC
LIMIT 1;
```

9. What is the average number of orders placed per customer?

```
SELECT AVG(order_count) AS avg_orders_per_customer
FROM (SELECT customer_id, COUNT(*) AS order_count FROM Orders GROUP BY
customer_id) AS customer_orders;
```

10. What is the total amount spent in orders for each customer?

```
SELECT customer_id, SUM(TotalAmount) AS total_spent
FROM Orders
GROUP BY customer_id;
```

11. Which order has the highest total amount?

```
SELECT order_id, TotalAmount
FROM Orders
ORDER BY TotalAmount DESC
LIMIT 1;
```

12. Which month had the lowest total sales?

```
SELECT EXTRACT(MONTH FROM order_date) AS month, SUM(TotalAmount) AS
total_sales
FROM Orders
GROUP BY month
ORDER BY total_sales ASC
LIMIT 1;
```

13. What is the total sales amount for orders placed after a specific date (e.g., after June 1st, 2024)?

```
SELECT SUM(TotalAmount) AS total_sales_after_date
FROM Orders
WHERE order_date > '2024-06-01';
```

14. How many orders have a total amount greater than ₹5000?

```
SELECT COUNT(*) AS orders_above_5000
FROM Orders
WHERE TotalAmount > 5000;
```

15. What is the distribution of order amounts (e.g., how many orders are below ₹1000, ₹1001-₹3000, above ₹3000)?

```
SELECT
  CASE
    WHEN TotalAmount < 1000 THEN 'Below ₹1000'
    WHEN TotalAmount BETWEEN 1001 AND 3000 THEN '₹1001-₹3000'
    ELSE 'Above ₹3000'
  END AS order_range,
  COUNT(*) AS order_count
FROM Orders
GROUP BY order_range;
```

16. How many orders were placed on each specific day?

```
SELECT order_date, COUNT(*) AS orders_count
FROM Orders
GROUP BY order_date
ORDER BY order_date;
```

17. What is the average order value for orders placed in each month?

```
SELECT EXTRACT(MONTH FROM order_date) AS month, AVG(TotalAmount) AS
avg_order_value
FROM Orders
GROUP BY month
ORDER BY month;
```

18. Which customer placed an order with the highest total amount in a single transaction?

```
SELECT customer_id, order_id, TotalAmount
FROM Orders
ORDER BY TotalAmount DESC
LIMIT 1;
```

19. What is the average time between orders placed by the same customer?

```
SELECT customer_id, AVG(DATEDIFF(order_date, LAG(order_date) OVER (PARTITION
BY customer_id ORDER BY order_date))) AS avg_time_between_orders
FROM Orders
GROUP BY customer_id;
```

20. How many orders were placed within a specific date range (e.g., from January 1st, 2024 to March 31st, 2024)?

```
SELECT COUNT(*) AS orders_in_date_range
FROM Orders
WHERE order_date BETWEEN '2024-01-01' AND '2024-03-31';
```

4. OrderDetail Table(500 records)

1. What is the total sales amount for each product (Quantity * UnitPrice)?

```
SELECT product_id, SUM(quantity * unit_price) AS total_sales
FROM OrderDetail
GROUP BY product_id;
```

2. How many units of each product were sold in total?

```
SELECT product_id, SUM(quantity) AS total_units_sold
FROM OrderDetail
```

```
GROUP BY product_id;
```

3. What is the average quantity ordered for each product?

```
SELECT product_id, AVG(quantity) AS avg_quantity_ordered
FROM OrderDetail
GROUP BY product_id;
```

4. What is the maximum quantity ordered for any single product?

```
SELECT product_id, MAX(quantity) AS max_quantity_ordered
FROM OrderDetail
GROUP BY product_id;
```

5. What is the total quantity sold for each product?

```
SELECT product_id, SUM(quantity) AS total_quantity_sold
FROM OrderDetail
GROUP BY product_id;
```

6. What is the total revenue generated by each product?

```
SELECT product_id, SUM(quantity * unit_price) AS total_revenue
FROM OrderDetail
GROUP BY product_id;
```

7. Which product has the highest total sales volume?

```
SELECT product_id, SUM(quantity * unit_price) AS total_sales
FROM OrderDetail
GROUP BY product_id
ORDER BY total_sales DESC
LIMIT 1;
```

8. Which product has the lowest total sales volume?

```
SELECT product_id, SUM(quantity * unit_price) AS total_sales
FROM OrderDetail
GROUP BY product_id
ORDER BY total_sales ASC
LIMIT 1;
```

9. What is the average unit price of products across all records?

```
SELECT AVG(unit_price) AS avg_unit_price
FROM OrderDetail;
```

10. Which product has the highest unit price?

```
SELECT product_id, MAX(unit_price) AS highest_unit_price
FROM OrderDetail
GROUP BY product_id
ORDER BY highest_unit_price DESC
LIMIT 1;
```

11. Which product has the lowest unit price?

```
SELECT product_id, MIN(unit_price) AS lowest_unit_price
FROM OrderDetail
GROUP BY product_id
ORDER BY lowest_unit_price ASC
LIMIT 1;
```

12. How many products have been ordered more than 100 times?

```
SELECT product_id, COUNT(*) AS order_count
FROM OrderDetail
GROUP BY product_id
HAVING COUNT(*) > 100;
```

13. What is the total revenue from products with a unit price greater than 500?

```
SELECT product_id, SUM(quantity * unit_price) AS total_revenue
FROM OrderDetail
WHERE unit_price > 500
GROUP BY product_id;
```

14. What is the total quantity sold for products with a unit price below 200?

```
SELECT product_id, SUM(quantity) AS total_quantity_sold
FROM OrderDetail
WHERE unit_price < 200
GROUP BY product_id;
```

15. Which product has the highest revenue per unit sold?

```
SELECT product_id, SUM(quantity * unit_price) / SUM(quantity) AS revenue_per_unit
FROM OrderDetail
GROUP BY product_id
ORDER BY revenue_per_unit DESC
LIMIT 1;
```

16. What is the average unit price for products ordered more than 50 times?

```
SELECT product_id, AVG(unit_price) AS avg_unit_price
FROM OrderDetail
```

```
GROUP BY product_id
HAVING COUNT(*) > 50;
```

17. What is the total number of product details (order items) in the system?

```
SELECT COUNT(*) AS total_order_items
FROM OrderDetail;
```

18. Which product is ordered the most in terms of quantity?

```
SELECT product_id, SUM(quantity) AS total_quantity_ordered
FROM OrderDetail
GROUP BY product_id
ORDER BY total_quantity_ordered DESC
LIMIT 1;
```

19. Which product has been ordered the least in terms of quantity?

```
SELECT product_id, SUM(quantity) AS total_quantity_ordered
FROM OrderDetail
GROUP BY product_id
ORDER BY total_quantity_ordered ASC
LIMIT 1;
```

20. What is the total quantity of products with a price greater than 300?

```
SELECT product_id, SUM(quantity) AS total_quantity
FROM OrderDetail
WHERE unit_price > 300
GROUP BY product_id;
```

5. Employee Table(25 records)

1. How many employees hold each unique position?

```
SELECT position, COUNT(*) AS employee_count
FROM Employee
GROUP BY position;
```

2. What is the total count of employees in the table?

```
SELECT COUNT(*) AS total_employees
FROM Employee;
```

3. Are there any duplicate employee names in the table?

```
SELECT first_name, last_name, COUNT(*) AS name_count
FROM Employee
GROUP BY first_name, last_name
HAVING COUNT(*) > 1;
```

4. How many employees have a phone number starting with a specific digit or prefix?

```
SELECT COUNT(*) AS employees_with_prefix
FROM Employee
WHERE phone LIKE '123%';  -- Change '123%' to the desired prefix
```

5. What are the most common positions held by employees?

```
SELECT position, COUNT(*) AS position_count
FROM Employee
GROUP BY position
ORDER BY position_count DESC
LIMIT 1;
```

6. What percentage of employees have a specific email domain (e.g., @gmail.com)?

```
SELECT
   (COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Employee)) AS percentage
FROM Employee
WHERE email LIKE '%@gmail.com';
```

7. How many employees' names start with a particular letter or pattern?

```
SELECT COUNT(*) AS employees_with_pattern
FROM Employee
WHERE first_name LIKE 'A%';  -- Change 'A%' to desired pattern
```

8. What is the distribution of employees based on the length of their names?

```
SELECT LENGTH(first_name) AS name_length, COUNT(*) AS employee_count
FROM Employee
GROUP BY LENGTH(first_name)
ORDER BY name_length;
```

9. How many employees have phone numbers with a length greater than 10 digits?

```
SELECT COUNT(*) AS employees_with_long_phone_numbers
FROM Employee
WHERE LENGTH(phone) > 10;
```

10. Are there any employees with missing or null email or phone values?

```
SELECT COUNT(*) AS employees_with_missing_data
```

```
FROM Employee
WHERE email IS NULL OR phone IS NULL;
```

11. What are the unique positions available in the table?

```
SELECT DISTINCT position
FROM Employee;
```

12. How many employees have names longer than 15 characters?

```
SELECT COUNT(*) AS employees_with_long_names
FROM Employee
WHERE LENGTH(first_name) + LENGTH(last_name) > 15;
```

13. What is the frequency of employees whose position titles contain the word "Manager"?

```
SELECT position, COUNT(*) AS position_count
FROM Employee
WHERE position LIKE '%Manager%'
GROUP BY position;
```

14. How many employees have names containing specific characters or substrings (e.g., "an")?

```
SELECT COUNT(*) AS employees_with_substring
FROM Employee
WHERE first_name LIKE '%an%' OR last_name LIKE '%an%';
```

15. What is the average length of employee names?

```
SELECT AVG(LENGTH(first_name) + LENGTH(last_name)) AS avg_name_length
FROM Employee;
```

16. How many employees have phone numbers ending in specific digits?

```
SELECT COUNT(*) AS employees_with_end_digits
FROM Employee
WHERE phone LIKE '%123';  -- Change '%123' to the desired ending
```

17. How many unique email domains are used by employees?

```
SELECT COUNT(DISTINCT SUBSTRING_INDEX(email, '@', -1)) AS
unique_email_domains
FROM Employee;
```

18. Which positions are held by only one employee?

```
SELECT position
```

```
FROM Employee
GROUP BY position
HAVING COUNT(*) = 1;
```

19. Are there any patterns in employee phone numbers?

```
SELECT phone, COUNT(*) AS phone_count
FROM Employee
GROUP BY phone
HAVING COUNT(*) > 1;
```

20. How many employees have emails with more than 20 characters?

```
SELECT COUNT(*) AS employees_with_long_emails
FROM Employee
WHERE LENGTH(email) > 20;
```

21. Are there any positions without any employees?

```
SELECT position
FROM (SELECT DISTINCT position FROM Employee) AS unique_positions
WHERE position NOT IN (SELECT DISTINCT position FROM Employee);
```

22. What is the proportion of employees in managerial vs. non-managerial roles?

```
SELECT
   SUM(CASE WHEN position LIKE '%Manager%' THEN 1 ELSE 0 END) AS
manager_count,
   SUM(CASE WHEN position NOT LIKE '%Manager%' THEN 1 ELSE 0 END) AS
non_manager_count
FROM Employee;
```

23. Are there any anomalies in phone numbers, such as non-numeric characters?

```
SELECT phone
FROM Employee
WHERE phone NOT REGEXP '^[0-9]+$';
```

24. What percentage of employees have generic position titles (e.g., "Staff")?

```
SELECT
   (COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Employee)) AS percentage
FROM Employee
WHERE position = 'Staff';  -- Change 'Staff' to the generic title
```

25. How many employees have names with special characters or spaces?

```
SELECT COUNT(*) AS employees_with_special_chars
```

```
FROM Employee
WHERE first_name REGEXP '[^a-zA-Z0-9]' OR last_name REGEXP '[^a-zA-Z0-9]';
```

6. Activity Table(105 records)

1. How many activities are recorded for each type?

```
SELECT activity_type, COUNT(*) AS activity_count
FROM Activity
GROUP BY activity_type;
```

2. What is the total number of activities in the table?

```
SELECT COUNT(*) AS total_activities
FROM Activity;
```

3. How many activities took place on weekends?

```
SELECT COUNT(*) AS weekend_activities
FROM Activity
WHERE DAYOFWEEK(activity_date) IN (1, 7);  -- 1 = Sunday, 7 = Saturday
```

4. What is the distribution of activities by time of day (morning, afternoon, evening)?

```
SELECT
   CASE
      WHEN HOUR(activity_time) BETWEEN 6 AND 11 THEN 'Morning'
      WHEN HOUR(activity_time) BETWEEN 12 AND 17 THEN 'Afternoon'
      WHEN HOUR(activity_time) BETWEEN 18 AND 23 THEN 'Evening'
   END AS time_of_day,
   COUNT(*) AS activity_count
FROM Activity
GROUP BY time_of_day;
```

5. Are there any duplicate activity descriptions?

```
SELECT description, COUNT(*) AS description_count
FROM Activity
GROUP BY description
HAVING COUNT(*) > 1;
```

6. How many activities occurred in a specific month?

```
SELECT COUNT(*) AS activities_in_month
FROM Activity
WHERE MONTH(activity_date) = 5;  -- Change '5' to the desired month number
```

7. What are the most frequently occurring activity types?

```
SELECT activity_type, COUNT(*) AS type_count
FROM Activity
GROUP BY activity_type
ORDER BY type_count DESC
LIMIT 1;
```

8. How many activities were scheduled on a specific date?

```
SELECT COUNT(*) AS activities_on_date
FROM Activity
WHERE activity_date = '2024-11-01';  -- Change the date as needed
```

9. What is the earliest and latest time of activities in the table?

```
SELECT MIN(activity_time) AS earliest_time, MAX(activity_time) AS latest_time
FROM Activity;
```

10. How many activities are scheduled for each day of the week?

```
SELECT
   DAYOFWEEK(activity_date) AS day_of_week,
   COUNT(*) AS activity_count
FROM Activity
GROUP BY day_of_week;
```

11. Are there any activities with missing or null descriptions?

```
SELECT COUNT(*) AS activities_with_missing_descriptions
FROM Activity
WHERE description IS NULL OR description = '';
```

12. What percentage of activities have descriptions longer than 50 characters?

```
SELECT
   (COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Activity)) AS percentage
FROM Activity
WHERE LENGTH(description) > 50;
```

13. How many activities occurred within a specific date range?

```
SELECT COUNT(*) AS activities_in_date_range
FROM Activity
WHERE activity_date BETWEEN '2024-01-01' AND '2024-12-31';  -- Adjust the range as needed
```

14. Are there any patterns in activity scheduling times?

```
SELECT activity_time, COUNT(*) AS activity_count
FROM Activity
GROUP BY activity_time
HAVING COUNT(*) > 1;
```

15. How many unique activity types are listed in the table?

```
SELECT COUNT(DISTINCT activity_type) AS unique_activity_types
FROM Activity;
```

16. How many activities are scheduled before 12:00 PM?

```
SELECT COUNT(*) AS activities_before_noon
FROM Activity
WHERE HOUR(activity_time) < 12;
```

17. What is the frequency of activities with descriptions containing specific keywords (e.g., "training")?

```
SELECT COUNT(*) AS training_activities
FROM Activity
WHERE description LIKE '%training%';
```

18. How many activities are scheduled on the same date and time?

```
SELECT activity_date, activity_time, COUNT(*) AS same_time_activities
FROM Activity
GROUP BY activity_date, activity_time
HAVING COUNT(*) > 1;
```

19. What is the average number of activities per day?

```
SELECT AVG(daily_activity_count) AS avg_activities_per_day
FROM (
   SELECT activity_date, COUNT(*) AS daily_activity_count
   FROM Activity
   GROUP BY activity_date
) AS daily_counts;
```

20. Are there any days with no activities scheduled?

```
SELECT activity_date
FROM (SELECT DISTINCT activity_date FROM Activity) AS unique_dates
WHERE activity_date NOT IN (SELECT DISTINCT activity_date FROM Activity);
```

21. How many activities took place during working hours (e.g., 9 AM to 5 PM)?

```
SELECT COUNT(*) AS working_hours_activities
FROM Activity
WHERE HOUR(activity_time) BETWEEN 9 AND 17;
```

22. What is the distribution of activities across different months?

```
SELECT MONTH(activity_date) AS month, COUNT(*) AS activity_count
FROM Activity
GROUP BY month;
```

23. Are there any activities scheduled in the past or future?

```
SELECT COUNT(*) AS past_activities
FROM Activity
WHERE activity_date < CURDATE();  -- Past activities

SELECT COUNT(*) AS future_activities
FROM Activity
WHERE activity_date > CURDATE();  -- Future activities
```

24. How many activities have unique descriptions?

```
SELECT COUNT(DISTINCT description) AS unique_descriptions
FROM Activity;
```

25. What is the proportion of activities that occur in the morning versus the afternoon?

```
SELECT
   SUM(CASE WHEN HOUR(activity_time) BETWEEN 6 AND 11 THEN 1 ELSE 0 END) AS morning_activities,
   SUM(CASE WHEN HOUR(activity_time) BETWEEN 12 AND 17 THEN 1 ELSE 0 END) AS afternoon_activities
FROM Activity;
```

26. How many activities have the same type but different descriptions?

```
SELECT activity_type, COUNT(DISTINCT description) AS unique_descriptions
FROM Activity
GROUP BY activity_type
HAVING COUNT(DISTINCT description) > 1;
```

27. What is the average length of activity descriptions?

```
SELECT AVG(LENGTH(description)) AS avg_description_length
FROM Activity;
```

28. How many activities are scheduled at the same time across different dates?

```
SELECT activity_time, COUNT(DISTINCT activity_date) AS same_time_activity_count
FROM Activity
GROUP BY activity_time
HAVING COUNT(DISTINCT activity_date) > 1;
```

29. How many activities have "Meeting" or "Training" in their type?

```
SELECT COUNT(*) AS meeting_or_training_activities
FROM Activity
WHERE activity_type LIKE '%Meeting%' OR activity_type LIKE '%Training%';
```

30. How many activities occurred in the last 30 days?

```
SELECT COUNT(*) AS activities_in_last_30_days
FROM Activity
WHERE activity_date >= CURDATE() - INTERVAL 30 DAY;
```