SQL Questions for data analyst role 1 to 76

1. **What is SQL and what does it stand for?**
   - SQL stands for Structured Query Language. It is a standard language used to interact with relational databases. SQL allows users to perform tasks such as querying databases, updating data, and managing database structures.
2. **Differentiate between SQL and MySQL.**
   - SQL is a standard language used to interact with relational databases, while MySQL is a specific relational database management system (RDBMS) that uses SQL as its language for managing and manipulating data.
3. **What is a primary key in SQL?**
   - A primary key is a unique identifier for each record in a table. It ensures that each row in a table is uniquely identifiable. Primary keys cannot contain null values.
4. **Explain the concept of a foreign key.**
   - A foreign key is a column or a set of columns in a table that refers to the primary key of another table. It establishes a link between two tables, enforcing referential integrity between them.
5. **How do you retrieve all the records from a table?**
   - To retrieve all records from a table, you can use the `SELECT` statement without any conditions, like this: `SELECT * FROM table_name;`
6. **What is the difference between the WHERE and HAVING clauses in SQL?**
   - The `WHERE` clause is used to filter rows before they are grouped and sorted, while the `HAVING` clause is used to filter groups after they have been formed.
7. **How do you eliminate duplicate rows from a result set?**
   - To eliminate duplicate rows, you can use the `DISTINCT` keyword in the `SELECT` statement, like this: `SELECT DISTINCT column1, column2 FROM table_name;`
8. **Explain the purpose of the GROUP BY clause.**
   - The `GROUP BY` clause is used to group rows that have the same values into summary rows. It is often used with aggregate functions like `SUM`, `COUNT`, `AVG`, etc., to perform calculations on grouped data.
9. **How do you sort the result set in descending order?**
   - To sort the result set in descending order, you can use the `ORDER BY` clause followed by the column name and the keyword `DESC`, like this: `ORDER BY column_name DESC;`

10. **What are the different types of JOINs in SQL?**
    - The different types of JOINs in SQL are `INNER JOIN`, `LEFT JOIN` (or `LEFT OUTER JOIN`), `RIGHT JOIN` (or `RIGHT OUTER JOIN`), and `FULL JOIN` (or `FULL OUTER JOIN`).
11. **Explain the difference between INNER JOIN and OUTER JOIN.**
    - `INNER JOIN` returns only the rows that have matching values in both tables, while `OUTER JOIN` returns all rows from both tables, with matching rows from both sides where available, and `NULL` values where no match is found.
12. **What is a subquery in SQL?**
    - A subquery is a query nested within another query. It can be used to return values that help in filtering, sorting, or performing calculations in the main query.
13. **How do you perform a conditional search in SQL?**
    - Conditional searches in SQL are performed using the `WHERE` clause, where you specify conditions to filter the rows returned by the `SELECT` statement.
14. **Explain the difference between CHAR and VARCHAR data types.**
    - `CHAR` is a fixed-length character data type that stores blank spaces up to the defined length, while `VARCHAR` is a variable-length character data type that only stores the actual characters entered.
15. **What is the purpose of the LIKE operator in SQL?**
    - The `LIKE` operator is used to search for a specified pattern in a column. It is often used with wildcard characters (% and _) to match patterns.
16. **How do you count the number of rows in a table?**
    - To count the number of rows in a table, you can use the `COUNT` function, like this: `SELECT COUNT(*) FROM table_name;`
17. **What is the use of the DISTINCT keyword in SQL?**
    - The `DISTINCT` keyword is used to eliminate duplicate rows from the result set returned by a `SELECT` statement.
18. **Explain the purpose of the ORDER BY clause.**
    - The `ORDER BY` clause is used to sort the result set returned by a `SELECT` statement in either ascending or descending order based on one or more columns.
19. **How do you insert records into a table?**
    - To insert records into a table, you can use the `INSERT INTO` statement followed by the table name and the values to be inserted.
20. **What is a NULL value in SQL?**
    - NULL is a special marker used in SQL to indicate that a data value does not exist in the database. It represents missing, unknown, or undefined data.

21. **How do you update records in a table?**
    - To update records in a table, you can use the `UPDATE` statement followed by the table name, `SET` keyword, and the column-value pairs to be updated, along with an optional `WHERE` clause to specify which rows to update.
22. **Explain the concept of normalization in databases.**
    - Normalization is the process of organizing data in a database to reduce redundancy and dependency. It involves dividing large tables into smaller tables and defining relationships between them.
23. **What is a stored procedure and why is it used?**
    - A stored procedure is a set of SQL statements that are stored in the database and can be executed by calling the procedure. It is used to encapsulate and execute frequently used tasks or business logic on the database server.
24. **How do you delete records from a table?**
    - To delete records from a table, you can use the `DELETE FROM` statement followed by the table name and an optional `WHERE` clause to specify which rows to delete.
25. **Explain the concept of ACID properties in SQL.**
    - ACID (Atomicity, Consistency, Isolation, Durability) properties are a set of properties that guarantee the reliability of database transactions. Atomicity ensures that transactions are either fully completed or fully rolled back, Consistency ensures that the database remains in a valid state after transactions, Isolation ensures that transactions do not interfere with each other, and Durability ensures that the effects of committed transactions are permanent.
26. **What is the difference between UNION and UNION ALL?**
    - `UNION` is used to combine the result sets of two or more `SELECT` statements into a single result set, while `UNION ALL` also combines result sets but includes duplicate rows.
27. **How do you rename a table in SQL?**
    - To rename a table in SQL, you can use the `ALTER TABLE` statement followed by the `RENAME TO` clause and the new table name.
28. **What is a trigger in SQL?**
    - A trigger is a special type of stored procedure that is automatically executed in response to certain events (e.g., `INSERT`, `UPDATE`, `DELETE`) occurring in the database.
29. **How do you alter a table to add a new column?**

- To add a new column to an existing table, you can use the `ALTER TABLE` statement followed by the `ADD` keyword, the new column name, and its data type.

30. **Explain the purpose of the BETWEEN operator in SQL.**
    - The `BETWEEN` operator is used to filter the result set to include only rows that fall within a specified range of values.
31. **How do you find the second highest value in a column?**
    - To find the second highest value in a column, you can use the `MAX` function with a subquery to exclude the highest value.
32. **What is a self-join in SQL?**
    - A self-join is a join operation where a table is joined with itself. It is used to compare rows within the same table.
33. **How do you retrieve only a specific number of rows from a result set?**
    - To retrieve only a specific number of rows from a result set, you can use the `LIMIT` clause in MySQL or the `TOP` clause in SQL Server.
34. **What is a view in SQL and why is it used?**
    - A view is a virtual table created by a query. It is used to simplify complex queries, restrict access to sensitive data, and provide a layer of abstraction over the underlying tables.
35. **How do you calculate the average, sum, and maximum value of a column?**
    - To calculate the average, sum, and maximum value of a column, you can use the `AVG`, `SUM`, and `MAX` aggregate functions, respectively.
36. **Explain the concept of indexing in SQL.**
    - Indexing is a database optimization technique used to improve the speed of data retrieval operations by creating a data structure (index) that allows for quick lookups of rows based on one or more columns.
37. **What is the purpose of the COALESCE function in SQL?**
    - The `COALESCE` function is used to return the first non-null value in a list of expressions.
38. **How do you perform a case-insensitive search in SQL?**
    - To perform a case-insensitive search, you can use the `LOWER` or `UPPER` function to convert both the column value and the search string to the same case.
39. **What is a deadlock in SQL and how can it be resolved?**
    - A deadlock is a situation where two or more transactions are waiting indefinitely for each other to release locks on resources. Deadlocks can be resolved by implementing proper transaction management techniques, such as setting appropriate isolation levels or using deadlock detection and resolution mechanisms.

40. **How do you create a new database in SQL?**
    - To create a new database, you can use the `CREATE DATABASE` statement followed by the database name.
41. **Explain the concept of data integrity in SQL.**
    - Data integrity refers to the accuracy, consistency, and reliability of data stored in a database. It is maintained through various mechanisms such as constraints, relationships, and validation rules.
42. **What is the difference between a clustered and non-clustered index?**
    - A clustered index determines the physical order of rows in a table based on the indexed columns, while a non-clustered index does not affect the physical order of rows and stores the index separately from the data.
43. **How do you retrieve the current date and time in SQL?**
    - To retrieve the current date and time, you can use the `GETDATE()` function in SQL Server or the `CURRENT_TIMESTAMP` or `NOW()` function in MySQL.
44. **What is the purpose of the TRUNCATE statement in SQL?**
    - The `TRUNCATE` statement is used to quickly delete all rows from a table without logging individual row deletions.
45. **How do you retrieve the first N records from a table?**
    - To retrieve the first N records from a table, you can use the `LIMIT` clause in MySQL or the `TOP` clause in SQL Server.
46. **Explain the purpose of the ISNULL function in SQL.**
    - The `ISNULL` function is used to replace NULL values with a specified replacement value.
47. **How do you find the length of a string in SQL?**
    - To find the length of a string, you can use the `LEN` function in SQL Server or the `LENGTH` function in MySQL.
48. **What is the difference between the UPDATE and INSERT statements?**
    - The `INSERT` statement is used to add new rows to a table, while the `UPDATE` statement is used to modify existing rows in a table.
49. **How do you calculate a running total in SQL?**
    - To calculate a running total, you can use a window function such as `SUM` along with the `OVER` clause to specify the partitioning and ordering of rows.
50. **Explain the concept of a correlated subquery in SQL.**
    - A correlated subquery is a subquery that depends on the outer query for its values. It is executed once for each row processed by the outer query and can be used to filter or modify the results based on values from the outer query.

51. **What is a composite key in SQL?**
    - A composite key is a combination of two or more columns that uniquely identifies each row in a table.
52. **Explain the purpose of the IN operator in SQL.**
    - The `IN` operator is used to specify multiple values in a `WHERE` clause, allowing you to filter data based on a set of predefined values.
53. **What is the purpose of the ALL and ANY operators in SQL?**
    - The `ALL` operator is used to compare a value to all values returned by a subquery, while the `ANY` operator is used to compare a value to any value returned by a subquery.
54. **What is a constraint in SQL?**
    - A constraint is a rule enforced on data in a table to maintain data integrity. Common types of constraints include `PRIMARY KEY`, `FOREIGN KEY`, `UNIQUE`, `NOT NULL`, and `CHECK`.
55. **What is denormalization and when would you use it?**
    - Denormalization is the process of adding redundancy to a database design to improve read performance by reducing the need for joins. It is often used in data warehousing and reporting scenarios where read performance is critical.
56. **Explain the purpose of the UNION operator in SQL.**
    - The `UNION` operator is used to combine the result sets of two or more `SELECT` statements into a single result set, removing duplicate rows by default.
57. **What is a correlated subquery and how is it different from a regular subquery?**
    - A correlated subquery is a subquery that depends on the outer query for its values. It is executed once for each row processed by the outer query. In contrast, a regular subquery is executed independently of the outer query.
58. **Explain the purpose of the CASE statement in SQL.**
    - The `CASE` statement is used to perform conditional logic in SQL, allowing you to return different values based on specified conditions.
59. **What is a temporary table in SQL and when would you use it?**
    - A temporary table is a temporary storage structure that exists only for the duration of a session or transaction. It is often used to store intermediate results or perform complex data manipulations.

60. **What is the purpose of the EXISTS operator in SQL?**
    - The `EXISTS` operator is used to test whether a subquery returns any rows. It returns `TRUE` if the subquery returns at least one row and `FALSE` otherwise.

61. **What is the purpose of the ROW_NUMBER() function in SQL?**
    - The `ROW_NUMBER()` function assigns a unique sequential integer to each row in the result set based on the specified ordering.

62. **Explain the difference between UNION and UNION ALL in SQL.**
    - `UNION` combines the result sets of two or more `SELECT` statements, removing duplicate rows, while `UNION ALL` combines the result sets without removing duplicates.

63. **What is a correlated subquery and how is it used?**
    - A correlated subquery is a subquery that references columns from the outer query. It is used to filter the results based on values derived from the outer query.

64. **How do you use the CASE statement to categorize data in SQL?**
    - You can use the `CASE` statement to categorize data by specifying conditions and corresponding categories. For example:

```
SELECT
  CASE
    WHEN column_name < 10 THEN 'Category A'
    WHEN column_name >= 10 AND column_name < 20 THEN 'Category B'
    ELSE 'Other'
  END AS category
FROM
  table_name;
```

61. **What is a recursive CTE (Common Table Expression) in SQL?**
    - A recursive CTE is a type of CTE that references itself in its definition. It is commonly used to traverse hierarchical data structures, such as organizational charts or bill of materials.

62. **Explain the purpose of the OVER() clause in SQL.**
    - The `OVER()` clause is used in conjunction with window functions to define the window or set of rows over which the function operates.

63. **What is a correlated update in SQL?**
    - A correlated update is an `UPDATE` statement that references columns from a correlated subquery. It updates rows in the table based on the results of the subquery.

64. **How do you use the COALESCE function to handle NULL values in SQL?**
    - The `COALESCE` function returns the first non-NULL value from a list of expressions. It can be used to handle NULL values by replacing them with alternate values.
65. **Explain the purpose of the LEAD() and LAG() functions in SQL.**
    - The `LEAD()` function is used to access data from the next row in the result set, while the `LAG()` function is used to access data from the previous row. They are often used to perform calculations across adjacent rows.
66. **What is a window function in SQL and how is it used?**
    - A window function performs a calculation across a set of rows that are related to the current row. It is used in conjunction with the `OVER()` clause to define the window or set of rows over which the function operates.
67. **What is the purpose of the GROUP_CONCAT() function in SQL?**
    - The `GROUP_CONCAT()` function concatenates the values of a column from multiple rows into a single string, grouped by a specified column.
68. **Explain the difference between a subquery and a derived table in SQL.**
    - A subquery is a query nested within another query, while a derived table is a subquery that is treated as a temporary table in the outer query's context.
69. **What is the purpose of the MERGE statement in SQL?**
    - The `MERGE` statement is used to perform multiple DML operations (INSERT, UPDATE, DELETE) in a single statement based on a condition.
70. **How do you pivot data in SQL using the PIVOT clause?**
    - The `PIVOT` clause is used to rotate rows into columns to produce a summarized result. It requires an aggregate function and a list of values to pivot.
71. **Explain the purpose of the XMLAGG() function in SQL.**
    - The `XMLAGG()` function is used to aggregate values into an XML document. It concatenates values and generates an XML element for each row.
72. **What is the purpose of the ROLLUP and CUBE operators in SQL?**
    - The `ROLLUP` and `CUBE` operators are used in combination with the `GROUP BY` clause to generate multiple levels of subtotals and grand totals in a result set.
73. **How do you handle hierarchical data in SQL?**
    - Hierarchical data can be handled using recursive common table expressions (CTEs) or hierarchical queries (e.g., `CONNECT BY` clause in Oracle).

74. **Explain the purpose of the FOR XML clause in SQL Server.**
    - The `FOR XML` clause in SQL Server is used to return query results as XML. It allows you to customize the XML structure and format.
75. **What is the purpose of the UNPIVOT clause in SQL?**
    - The `UNPIVOT` clause is used to rotate columns into rows, essentially the opposite of the `PIVOT` operation.
76. **How do you handle missing values in SQL?**
    - Missing values can be handled using the `COALESCE()` function to replace NULL values with a specified alternative value, or by using conditional logic in queries.