1. What does the SQL acronym stand for?

SQL (pronounced "ess-que-el") stands for Structured Query Language.

2. The SQL language is divided into three parts: DDL, DML and DCL. What is the purpose of each of them?

DDL – Data Definition Language - Used to create, modify, and delete database objects like tables, views, and indexes.

DML – Data Manipulation Language - Used to insert, update, delete, and query data within tables.

DCL – Data Control Language - Used to manage user access and permissions to database objects.

3. What is the difference between VARCHAR and CHAR data types?

VARCHAR is flexible in the storage data space it takes, while in CHAR that space it's fixed.

4. What is the difference between the TEXT and BLOB/BYTEA/LOBS data types?

TEXT is used for storing large amounts of character-based data, while BLOB/BYTEA/LOBS are used for storing large amounts of binary data, such as images or audio files.

TEXT -> stores a large number of characters

BLOB/BYTEA/LOBS -> Store a large quantity of binary data

5. What is the difference between the DATETIME and TIMESTAMP data types?

Both DATETIME and TIMESTAMP are used for storing date and time data in SQL.

The main difference between the two is the range of values they can store. DATETIME can store dates from January 1, 1753, to December 31, 9999, and TIMESTAMP can store dates from January 1, 1970, to 2038-01-19 03:14:07 UTC.

Datetime -> contains both time and date information

TimeStamp -> For time only + time zone (1970-2038) UNIX TIME

6. What data type would you use if you needed to store an exact real number without rounding?

In SQL, if you need to store an exact real number without rounding, you would use the "DECIMAL" or "NUMERIC" data type. These data types allow you to specify the exact precision and scale of the number you want to store.

7. Explain what the ON DELETE and ON UPDATE clauses are for.

ON DELETE -> Triggers an action when rows are deleted on a target table

ON UPDATE -> ditto(IDEM), but for update rows

8. Explain what are the following options that we can use together with the ON DELETE and ON UPDATE clauses:

   CASCADE: Deletes or updates child records automatically when the parent record is deleted or updated.

   SET NULL: Sets the foreign key values in child records to NULL when the parent record is deleted or updated.

   RESTRICT: Prevents the deletion or update of parent records that would result in the violation of the foreign key constraint.

   SET DEFAULT: Sets the foreign key values in child records to a default value when the parent record is deleted or updated..

9. What is the difference between TRUNCATE and DELETE?

   DELETE statement removes specific rows from a table based on a WHERE clause, or all rows from a table if no WHERE clause is specified. It is slower than TRUNCATE but more flexible, as it allows the removal of specific rows.

   TRUNCATE statement removes all rows from a table without logging individual row deletions. It is faster than DELETE and resets auto-increment values but it cannot be used to remove specific rows.

10. What is the UNIQUE attribute used for?

In SQL, the UNIQUE attribute is used to define a column or a set of columns that must have unique values in a table. It ensures that no two rows in the table have the same values in the specified column or columns.

11. What is the difference between INDEX and KEY? What are they used for?
    An INDEX is used to speed up the retrieval of data from a table by creating a separate data structure that allows for faster access to specific data.
    A KEY is used to ensure that the values in a column or set of columns are unique, and it can be used as a PRIMARY KEY to uniquely identify each row in a table, or as a FOREIGN KEY to reference a primary key in another table.
    In summary, INDEX helps with faster data retrieval, while KEY helps with data integrity and uniqueness.

12. What do the braces and square brackets mean in the following notation description?
    The curly braces {} indicate a choice between the enclosed options.
    The square brackets [] indicate an optional part of the statement.

13. What SQL statement would you use to know the structure of a table already created?
    \d table_name;

14. What SQL statement would you use to erase a table named 'exams'?
    DROP TABLE exams;

15. What SQL statement would you use to list of all the databases? (investigate)

    SELECT datname FROM pg_database;

16. What SQL statement would you use to add a new column called 'name' to the 'students' table?
    ALTER TABLE students
    ADD COLUMN name VARCHAR(50);

17. What SQL sentence allows us to rename and modify the data type of a column and its attributes? Give an example.

    To rename and modify the data type of a column and its attributes in SQL, you can use the ALTER TABLE statement with the ALTER COLUMN clause.

    ALTER TABLE students
    ALTER COLUMN age TYPE INTEGER | RENAME COLUMN age TO student_age;

18. What data type could store date and time values greater than 2038-01-09 03:14:07?
    DATE TIME

19. What data type would you use if you needed to store exact real numbers without rounding?
    Float

20. What do the values 7 and 4 represent when we declare a column as FLOAT(7,4)?

    When we declare a column as FLOAT(7,4), the values 7 and 4 represent the precision and scale of the FLOAT data type, respectively.

21. What does the value 11 represent when we declare a column as INT(11)?

    When we declare a column as INT(11), the value 11 represents the display width of the INT data type, It means that the column can show up to 11 digits, but it can store any integer value that fits in 4 bytes12.

22. What are all the clauses that we can use in the SELECT statement and in what order is each of them executed?

    In SQL, the SELECT statement is used to retrieve data from one or more tables in a database. The SELECT statement can include various clauses to filter, sort, and manipulate the data.
    The clauses that can be used in a SELECT statement are as follows:

    SELECT: The columns and expressions are selected and computed.
    FROM: The data sources are specified and joined.
    WHERE: The records that do not match the conditions are filtered out.
    GROUP BY: The records are grouped by one or more columns.
    HAVING: The groups that do not match the conditions are filtered out.
    WINDOW:
    DISTINCT:
    UNION / INTERSET/ EXCEPT:
    ORDER BY: The records are sorted by one or more columns.
    LIMIT / OFFSET / FETCH: The number of records to return is limited.
    SELECT:

23. When we make queries with the SELECT statement we can use the ALL modifiers and DISTINCT. What is the purpose of each of them?
    The purpose of the ALL modifier is to retrieve all matching rows, including duplicates(default).
    The purpose of the DISTINCT modifier is to retrieve only the unique values from the selected columns. It removes duplicates from the result set.

24. What are the following operators for? Give an example of use for each of them:

    1. BETWEEN - You can use this to see if a value is in a range.

    2. LIKE - You can use this to see if a value matches a pattern.

    3. IS - You can use this to see if a value is NULL.

    4. IS NOT - You can use this to see if a value is not NULL.

25. What does the COUNT function do in each of these cases? Is there a difference between them?
    The COUNT function is used to get the number of rows returned by a query1. There is a difference between the cases you mentioned:

    COUNT(*): This counts all the rows in a table or a group, regardless of NULL values or duplicates.
    COUNT(column): This counts all the rows in a table or a group that have a non-NULL value in the specified column.
    COUNT(DISTINCT column): This counts all the distinct (unique) values in a column.

26. What are the most commonly used aggregation functions which allow you to perform specific operations on a group of rows?
    AVG: This function returns the average value of a numeric column.
    COUNT: This function returns the number of rows that match a specified condition.
    MAX: This function returns the maximum value of a column.
    MIN: This function returns the minimum value of a column.
    SUM: This function returns the total sum of a numeric column.

27. Explain the difference between the WHERE and HAVING clauses. Give an example of use for each case.
    The main difference between the WHERE and HAVING clauses is that WHERE applies the condition to individual rows before they are grouped by the GROUP BY clause, while HAVING applies the condition to the groups after they are formed by the GROUP BY clause.

    Suppose we have a table called students with columns name, grade, and score. If we want to find the names and grades of the students who scored above 80 in any subject, we can use the WHERE clause like this:

    SELECT name, grade FROM students WHERE score > 80;

    This will filter out the rows where the score is less than or equal to 80 before selecting the name and grade columns.

    However, if we want to find the grades that have an average score above 70 across all subjects, we need to use the HAVING clause like this:

    SELECT grade, AVG(score) AS average_score FROM students GROUP BY grade HAVING AVG(score) > 70;

28. Define what the LIMIT clause is used for. Given the syntax of the LIMIT clause in SQL, explain what the words offset and ROW_COUNT. Give an example of use. [LIMIT {[offset,]row_COUNT | ROW_COUNT OFFSET offset}]

29. What are the INNER JOIN and NATURAL JOIN operations used for? What is the difference between them? Give a query example that affects two tables pointing out the difference.
    INNER : We need to point out the matching fields → ON(...)
    NATURAL : No need to point out matching fields the DBMS fetches all the fields with similar names to join both tables

30. What result does the CROSS JOIN operation return? Give a simple example. Is there another way to get the same result from the CROSS JOIN operation? Give an example. Each row from each table will ve desplayed avng with a very other row in the other table

31. What are the LEFT OUTER JOIN and RIGHT OUTER JOIN operations used for? What is the difference between them? Give an example of a query affecting two tables pointing out the difference.

32. Point out whether the following queries are correct or not. If they are incorrect, explain the reason and how they could be resolved correctly:
    a) SELECT * FROM student WHERE phone = NULL;
       The correct way to check for null values is by using the "IS NULL" or "IS NOT NULL" operator.

    b) SELECT * FROM student WHERE lastname1 = 'S%';

The query is correct. USE LIKE table where the value of the "lastname1" column starts with "S".

    c) SELECT * FROM student WHERE birth_date >= '1999/01/01' AND '1999/12/31'

When using the "AND" operator, you need to specify the column name and the comparison operator for each condition.

    d) SELECT * FROM product, manufacturer WHERE manufacturer.name = 'Lenovo';

    e) SELECT * FROM product INNER JOIN manufacturer WHERE manufacturer.name = 'Lenovo';

    f) SELECT * FROM product INNER JOIN manufacturer ON product.code = manufacturer.code;

    g) SELECT * FROM product WHERE price > AVG(price);