



# Python

## Interview Questions

Q1. What are the different subsets of SQL?

Ans-

- Data Definition Language (DDL) – It allows you to perform various operations on the database such as CREATE, ALTER, and DELETE objects.
- Data Manipulation Language (DML) – It allows you to access and manipulate data. It helps you to insert, update, delete and retrieve data from the database.
- Data Control Language (DCL) – It allows you to control access to the database. Example – Grant, Revoke access permissions.

Q2. What do you mean by DBMS? What are its different types?

Ans- A Database Management System (DBMS) is a software application that interacts with the user, applications, and the database itself to capture and analyze data. A database is a structured collection of data.

A DBMS allows a user to interact with the database. The data stored in the database can be modified, retrieved and deleted and can be of any type like strings, numbers, images, etc.

There are two types of DBMS:

- Relational Database Management System: The data is stored in relations (tables). Example – MySQL.
- Non-Relational Database Management System: There is no concept of relations, tuples and attributes. Example – MongoDB

Q3. What is a Self-Join?

Ans- A self-join is a type of join that can be used to connect two tables. As a result, it is a unary relationship. Each row of the table is attached to itself and all other rows of the same table in a self-join. As a result, a self-join is mostly used to combine and compare rows from the same database table.

Q4. What is the SELECT statement?

Ans- A SELECT command gets zero or more rows from one or more database tables or views. The most frequent data manipulation language (DML) command is SELECT in most applications. SELECT queries define a result set, but not how to calculate it, because SQL is a declarative programming language.

Q5. What are some common clauses used with SELECT query in SQL?

Ans- The following are some frequent SQL clauses used in conjunction with a SELECT query:

**WHERE** clause: In SQL, the WHERE clause is used to filter records that are required depending on certain criteria.

**ORDER BY** clause: The ORDER BY clause in SQL is used to sort data in ascending (ASC) or descending (DESC) order depending on specified field(s) (DESC).

**GROUP BY** clause: GROUP BY clause in SQL is used to group entries with identical data and may be used with aggregation methods to obtain summarised database results.

**HAVING** clause in SQL is used to filter records in combination with

the GROUP BY clause. It is different from WHERE, since the WHERE clause cannot filter aggregated records.

Q6. What is UNION, MINUS and INTERSECT commands?

Ans- The UNION operator is used to combine the results of two tables while also removing duplicate entries.

The MINUS operator is used to return rows from the first query but not from the second query.

The INTERSECT operator is used to combine the results of both queries into a single row.

Before running either of the above SQL statements, certain requirements must be satisfied –

Within the clause, each SELECT query must have the same amount of columns.

The data types in the columns must also be comparable.

In each SELECT statement, the columns must be in the same order.

Q7. What is Cursor? How to use a Cursor?

Ans- After any variable declaration, DECLARE a cursor. A SELECT Statement must always be coupled with the cursor definition.

To start the result set, move the cursor over it. Before obtaining rows from the result set, the OPEN statement must be executed.

To retrieve and go to the next row in the result set, use the FETCH command.

To disable the cursor, use the CLOSE command.

Finally, use the DEALLOCATE command to remove the cursor definition and free up the resources connected with it.

Q8. List the different types of relationships in SQL.

Ans- There are different types of relations in the database:

**One-to-One** – This is a connection between two tables in which each record in one table corresponds to the maximum of one record in the other.

**One-to-Many and Many-to-One** – This is the most frequent connection, in which a record in one table is linked to several records in another.

**Many-to-Many** – This is used when defining a relationship that requires several instances on each sides.

**Self-Referencing Relationships** – When a table has to declare a connection with itself, this is the method to employ.

Q9. What is OLTP?

Ans- OLTP, or online transactional processing, allows huge groups of people to execute massive amounts of database transactions in real time, usually via the internet. A database transaction occurs when data in a database is changed, inserted, deleted, or queried.

Q10. What are the differences between OLTP and OLAP?

Ans- OLTP stands for online transaction processing, whereas OLAP stands for online analytical processing. OLTP is an online database modification system, whereas OLAP is an online database query response system.

Q11. How to create empty tables with the same structure as another table?

Ans- To create empty tables:

Using the INTO operator to fetch the records of one table into a new table while setting a WHERE clause to false for all entries, it is possible to create empty tables with the same structure. As a result, SQL creates a new table with a duplicate structure to accept the fetched entries, but nothing is stored into the new table since the WHERE clause is active.

Q12. What is PostgreSQL?

Ans- In 1986, a team lead by Computer Science Professor Michael Stonebraker created PostgreSQL under the name Postgres. It was created to aid developers in the development of enterprise-level applications by ensuring data integrity and fault tolerance in systems. PostgreSQL is an enterprise-level, versatile, resilient, open-source, object-relational database management system that supports variable workloads and concurrent users.

Q13. What are SQL comments?

Ans- SQL Comments are used to clarify portions of SQL statements and to prevent SQL statements from being executed. Comments are quite important in many programming languages. The comments are not supported by a Microsoft Access database. As a result, the Microsoft Access database is used in the examples in Mozilla Firefox and Microsoft Edge.

Single Line Comments: It starts with two consecutive hyphens (—).

Multi-line Comments: It starts with /\* and ends with \*/.

Q14. What is the usage of the NVL () function?

Ans- You may use the NVL function to replace null values with a default value. The function returns the value of the second parameter if the first parameter is null. If the first parameter is anything other than null, it is left alone.

This function is used in Oracle, not in SQL and MySQL. Instead of NVL() function, MySQL have IFNULL() and SQL Server have ISNULL() function.

Q15. Explain character-manipulation functions? Explains its different types in SQL.

Ans- Change, extract, and edit the character string using character manipulation routines. The function will do its action on the input strings and return the result when one or more characters and words are supplied into it.

The character manipulation functions in SQL are as follows:

A) CONCAT (joining two or more values): This function is used to join two or more values together. The second string is always appended to the end of the first string.

B) SUBSTR: This function returns a segment of a string from a given start point to a given endpoint.

C) LENGTH: This function returns the length of the string in numerical form, including blank spaces.

D) INSTR: This function calculates the precise numeric location of a character or word in a string.

E) LPAD: For right-justified values, it returns the padding of the left-side character value.

F) RPAD: For a left-justified value, it returns the padding of the right-side character value.



G) TRIM: This function removes all defined characters from the beginning, end, or both ends of a string. It also reduced the amount of wasted space.

H) REPLACE: This function replaces all instances of a word or a section of a string (substring) with the other string value specified.

Q16. What is the difference between the RANK() and DENSE\_RANK() functions?

Ans- The RANK () function in the result set defines the rank of each row within your ordered partition. If both rows have the same rank, the next number in the ranking will be the previous rank plus a number of duplicates. If we have three records at rank 4, for example, the next level indicated is 7.

The DENSE\_RANK () function assigns a distinct rank to each row within a partition based on the provided column value, with no gaps. It always indicates a ranking in order of precedence. This function will assign the same rank to the two rows if they have the same rank, with the next rank being the next consecutive number. If we have three records at rank 4, for example, the next level indicated is 5.

Q17. What is a UNIQUE constraint?

The UNIQUE Constraint prevents identical values in a column from appearing in two records. The UNIQUE constraint guarantees that every value in a column is unique.

Q18. What is a Self-Join?

A self-join is a type of join that can be used to connect two tables. As a result, it is a unary relationship. Each row of the table is attached to itself and all other rows of the same table in a self-join. As a result, a self-join is mostly used to combine and compare rows from the same database table.

Q19. What is UNION, MINUS and INTERSECT commands?

Ans- The UNION operator is used to combine the results of two tables while also removing duplicate entries.

The MINUS operator is used to return rows from the first query but not from the second query.

The INTERSECT operator is used to combine the results of both queries into a single row.

Before running either of the above SQL statements, certain requirements must be satisfied –

Within the clause, each SELECT query must have the same number of columns.

The data types in the columns must also be comparable.

In each SELECT statement, the columns must be in the same order.

Q20. List the different types of relationships in SQL.

Ans- There are different types of relations in the database:

- a) One-to-One – This is a connection between two tables in which each record in one table corresponds to the maximum of one record in the other.
- b) One-to-Many and Many-to-One – This is the most frequent connection, in which a record in one table is linked to several records in another.
- c) Many-to-Many – This is used when defining a relationship that requires several instances on each sides.
- d) Self-Referencing Relationships – When a table has to declare a connection with itself, this is the method to employ.