iNeuron

Python

Interview Questions

1. Explain tables

Some rows and columns make up a table. It provides the ability to store and display data in a structured fashion. Similar to spreadsheet worksheets, it is a type of document. The tuples are represented by rows, while columns represent the attributes of the data items in a specific row. It's possible to think of rows as horizontal and columns as vertical.

2. State the difference between primary and unique keys

To uniquely identify records in a database, a primary key might be one or more table fields. A unique key, on the other hand, stops two rows from having identical items in a column.

In a relational database, a table can have numerous unique keys, but it can only have one primary key.

A unique key can have NULL values, however only one NULL is permitted in a table, whereas a main key column cannot have NULL values.

Although uniqueness is preferred for the main key, it's not a requirement that it serves as the primary key.

3. What is a foreign key?

A foreign key is an attribute or feature group that refers to a table's primary key in a different database. A foreign key is a data structure that connects two databases.

4. What are operators in SQL?

In SQL queries, specific operations can be performed using SQL operators, keywords, or characters. The WHERE clause of SQL

statements can employ these operators. SQL operators filter data based on the condition.

5. Explain data integrity

It guarantees data accuracy and consistency throughout its lifecycle. Systems that store, process, or retrieve data necessitate this feature in their design, implementation, and use.

6. Is SQL a programming language?

The Standard Query Language (SQL) is a query language, not a programming language. SQL does not have a loop, conditional statements, or logical operations, so it can only be used to manipulate information. It is used to navigate databases in the same way as commanding (Query) language is used. The primary goal of SQL is to acquire, manipulate, alter, and execute complex operations on database records, such as joins.

7. What is a Data Definition Language, and how does it work?

Data definition language (DDL) is a database subset that specifies the database's data structure in the initial stages of development. It's made up of the following instructions: CREATE, ALTER, and DELETE database objects, including schemas, tables, views, and sequences, among others.

8. What are the applications of SQL?

SQL is in charge of keeping the database's relational records and data models up to date.

- To submit queries against a database
- To get details out of a database
- To add information to a database
- To make changes to the records in a folder
- To exclude data from a database
- To create new databases
- Adding new tables to a database
- To populate a database of views
- To operate on the servers in a complex manner

9. What do you mean by a database management system? What are the various types?

A database management system (DBMS) is a software application that communicates with the individual, other programs, and the database to collect and interpret data. A database is a list of ordered records.

A database management system (DBMS) enables users to communicate with the database. The database's data can be changed, restored, and erased, and it can be of any kind, including strings, numbers, and illustrations.

There are two kinds of database management systems:

● Data is maintained in connections in a relational database management system (tables). MySQL is a good example.

● There is no definition of links, tuples, or attributes in a non-relational database management system. MongoDB as an illustration.

10. What is the difference between the SQL data types CHAR and VARCHAR2?

Both Char and Varchar2 are utilized for character data types, but Varchar2 is used for variable-length character strings, while Char is used for fixed-length character strings. For instance, Char(10) can only store ten characters and cannot store strings of any other length, while varchar2(10) can store strings of any length, e.g. 6,8,2.

11. In SQL, what is a foreign key?

By imposing a relation between the data in any two tables, a foreign key preserves referential integrity. The child table's international key refers to the parent table's primary key.

The international key restriction avoids behaviour that might cause the child and parent tables to lose their links.

12. What are SQL joins?

A JOIN clause is used to join rows from two or more tables together cantered on a common column. It is used to join two tables together or to extract data from one of them. There are four distinct forms of joins, as detailed below:

● Inner join:
The most often used form of join in SQL is the inner join. It is used to return all rows from multiple tables that satisfy the join condition.

● Left Join:
In SQL, a left join returns all rows from the left table but only those that satisfy the join condition in the right table.

● Right Join:

In SQL, a right join returns all rows from the right table but only those that satisfy the join condition in the left table.

● Full Join:
When there is a similarity in either of the columns, a full join recovers all of the data. As a result, it recovers both rows from both the left-hand side table and the right-hand side table.

13. How can a user distinguish between clustered and non-clustered indexes?

The following are the distinctions between clustered and non-clustered indexes in SQL:

● Clustered indexes are used to facilitate data extraction from the database and are quicker while reading from a non-clustered index is slower.

● A clustered index modifies the way records are stored in a database by sorting them by the clustered index column, while a non-clustered index does not change the way records are stored but also generates a different object inside a table that leads back to the initial table rows when searching.

● A table may have just one clustered index but may have several non-clustered indexes.

14. What do you mean when you say "query optimization"?

The process in which a strategy for evaluating a database is defined that has the lowest expected cost is referred to as query optimization.

The below are the benefits of query optimization:

- The product is delivered more quickly.
- One can execute a greater amount of queries in a shorter period.
- It reduces the uncertainty of time and space.

15. Describe the various forms of indexes available in SQL.

In SQL, there are three kinds of indexes:

A unique index:

If the column is uniquely indexed, this index prevents the sector from having repeated values. If a primary key is established, one can create a unique automated index.

A clustered Index that:

This index reorders the table's physical columns and performs queries dependent on key values. Each table can have a maximum of one clustered index.

The non-clustered index does not affect the spatial order of the table and preserves the data's conceptual order. Each table can contain a large number of nonclustered indexes.

16. What are you referring to when you say "Denormalization"?

Denormalization is a method for accessing data from a database's higher to lower forms. It enables database administrators to boost the overall efficiency of the infrastructure by introducing redundancy into a table. It populates a table with redundant data by adding database queries that consolidate data from several tables into a single table.

17. What exactly is an index?

An index is a performance optimization technique that enables the quicker retrieval of records from a table. An index provides an entry for each value, making data retrieval easier. SQL indexes are a way of lowering the cost of a query since a high cost of a query would result in a drop in query results. An index is used to improve efficiency and enable quicker retrieval of records from the table. The amount of data pages we need to access to locate a certain data page is reduced

thanks to indexing. Indexing, therefore, provides a one-of-a-kind significance, which ensures that one can't duplicate the index. An index provides an entry for each value, making data retrieval easier. E.g., suppose you have a book with nation details, and you want to learn more about India, instead of going through every page of the book. In that case, you could go straight to the index, and then from the index, you could go to the specific page where all the information about India is provided.

18. What is the Difference Between Entities and Relationships?

● Entity: In the physical world, an entity, location, or thing about which data can be stored in a database. Tables are used to store data that pertains to a certain class of object. For instance, a bank database contains a customer table that stores customer data. This information is stored in the customer table as a set of attributes (columns inside the table) for each customer.

● Relationships: Relationships or connections is when two entities share something in common. For instance, a customer's name is associated with the customer's account number and contact details, stored in the same table. Additionally, associations between different tables are possible (for example, customer to accounts).

19. What is a relational database management system (RDBMS)?

It's a database management framework using a relational database architecture. RDBMS is a database management system that stores

data in a series of tables and links them together with relational operators when required. Using relational operators, you can modify the data in the tables more quickly. Microsoft Access, MySQL, SQL Server, Oracle client, among other relational database management systems, are examples.

20. What exactly is normalization, and what are the benefits of doing so?

The method of arranging data in SQL to prevent repetition and replication is known as normalization. The below are some of the benefits:

- Improved Database Management
- Tables with narrower rows
- Data access that is quick and easy
- Greater Queries Flexibility
- Locate the details quickly.
- Security is easier to enforce.
- Allows for simple customization
- Data duplication and redundancy are reduced.
- More compact database
- Ensure the data remains consistent since it has been modified.