**1. What is Database?**

A database is an organized collection of data, stored and retrieved digitally from a remote or local computer system.

### 2. What is DBMS?

DBMS stands for Database Management System. DBMS is a system software responsible for the creation, retrieval, updation, and management of the database. It ensures that our data is consistent, organized, and is easily accessible by serving as an interface between the database and its end-users or application software.

### 3. What is RDBMS? How is it different from DBMS?

**The number of users who are permitted to utilise the system**  
A DBMS can only handle one user at a time, whereas an RDBMS can handle numerous users.  
**Hardware and software specifications**  
In comparison to an RDBMS, a DBMS requires fewer software and hardware.  
**Amount of information**  
RDBMSes can handle any quantity of data, from tiny to enormous, whereas DBMSes are limited to small amounts.  
**The structure of the database**  
Data is stored in a hierarchical format in a DBMS, whereas an RDBMS uses a table with headers that serve as column names and rows that hold the associated values.  
**Implementation of the ACID principle**  
The atomicity, consistency, isolation, and durability (ACID) concept is not used by DBMSs for data storage. RDBMSes, on the other hand, use the ACID model to organize their data and assure consistency.  
**Databases that are distributed**  
A DBMS will not provide complete support for distributed databases, whereas an RDBMS will.  
**Programs that are managed**  
A DBMS focuses on keeping databases that are present within the computer network and system hard discs, whereas an RDBMS helps manage relationships between its incorporated tables of data.  
**Normalization of databases is supported**  
A RDBMS can be normalized , but a DBMS cannot be normalized

### 4.List the different types of relationships in SQL.

**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.

EG. A student has only one ID card and an ID card is given to one person.

HAS

1 1

STUDENT

ID CARD

**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.

EG. A customer can place many orders, but an order cannot be placed by many customers.

PLACE

1. M

ORDER

CUSTOMER

EG. Student enrolls for only one course , but a course can have many students.

ENROLL

STUDENTS

**M 1**

COURSE

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

EG. Employee can have many projects and projects can have many employees.

ASSIGN

EMPLOYEE

### M M

PROJECT

**Self-Referencing** - This is used when a table needs to define a relationship with itself.

### 5.What is the usage of the NVL() function?

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.

**6.What is the ACID property in a database?**

ACID stands for Atomicity, Consistency, Isolation, Durability. It is used to ensure that the data transactions are processed reliably in a database system.

* **Atomicity:** Atomicity refers to the transactions that are completely done or failed where transaction refers to a single logical operation of a data. It means if one part of any transaction fails, the entire transaction fails and the database state is left unchanged.
* **Consistency:** Consistency ensures that the data must meet all the validation rules. In simple words,  you can say that your transaction never leaves the database without completing its state.
* **Isolation:** The main goal of isolation is concurrency control.
* **Durability:** Durability means that if a transaction has been committed, it will occur whatever may come in between such as power loss, crash or any sort of error.

### ****7.Are NULL values same as that of zero or a blank space?****

A NULL value is not at all same as that of zero or a blank space. NULL value represents a value which is unavailable, unknown, assigned or not applicable whereas a zero is a number and blank space is a character.

### ****8. What are the different types of a subquery?****

There are two types of subquery namely, Correlated and Non-Correlated.

**Correlated subquery**: These are queries which select the data from a table referenced in the outer query. It is not considered as an independent query as it refers to another table and refers the column in a table.

**Non-Correlated subquery**: This query is an independent query where the output of subquery is substituted in the main query.

### ****9.What is the need for MERGE statement?****

This statement allows conditional update or insertion of data into a table. It performs an UPDATE if a row exists, or an INSERT if the row does not exist.

### 10.What are the various levels of constraints?

* column level constraint
* table level constraint

### 11. What is the main difference between SQL and PL/SQL?

SQL is a query language that allows you to issue a single query or execute a single insert/update/delete whereas PL/SQL is Oracle’s “Procedural Language” SQL, which allows you to write a full program (loops, variables, etc.) to accomplish multiple operations such as selects/inserts/updates/deletes.

### 12.What is Auto Increment in SQL?

Autoincrement keyword allows the user to create a unique number to get generated whenever a new record is inserted into the table.  
This keyword is usually required whenever PRIMARY KEY in SQL is used.

[AUTO INCREMENT keyword](https://www.edureka.co/blog/sql-auto-increment/) can be used in Oracle and IDENTITY keyword can be used in SQL SERVER.

### 13.What is a Datawarehouse?

Datawarehouse refers to a central repository of data where the data is assembled from multiple sources of information. Those data are consolidated, transformed and made available for the mining as well as online processing. Warehouse data also have a subset of data called Data Marts.

### 14. What are Constraints in SQL?

Constraints are used to specify the rules concerning data in the table. It can be applied for single or multiple fields in an SQL table during the creation of the table.

* **NOT NULL** - Restricts NULL value from being inserted into a column.
* **CHECK** - Verifies that all values in a field satisfy a condition.
* **DEFAULT** - Automatically assigns a default value if no value has been specified for the field.
* **UNIQUE** - Ensures unique values to be inserted into the field.
* **PRIMARY KEY** - Uniquely identifies each record in a table.
* **FOREIGN KEY** - Ensures referential integrity for a record in another table.

### 15.What is a View?

A view in SQL is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.

**16.The difference between NVL and NVL2 functions?**

These functions work with any data type and pertain to the use of null values in the expression list. These are all single-rowfunctions i.e. provide one result per row.

**NVL(expr1, expr2):** In SQL, NVL() converts a null value to an actual value. Data types that can be used are date, character, and number. Data types must match with each other. i.e. expr1 and expr2 must be of the same data type.

**Syntax:**

**NVL (expr1, expr2)**

**NVL2(expr1, expr2, expr3):** The NVL2 function examines the first expression. If the first expression is not null, then the NVL2 function returns the second expression. If the first expression is null, then the third expression is returned i.e. If expr1 is not null, NVL2 returns expr2. If expr1 is null, NVL2 returns expr3. The argument expr1 can have any data type.

**Syntax:**

**NVL2 (expr1, expr2, expr3)**

### 17.What is the difference between a primary key and a unique key?



|  |  |
| --- | --- |
| **Primary Key** | **Unique Key** |
| The primary key act as a unique identifier for each record in the table. | The unique key is also a unique identifier for records when the primary key is not present in the table. |
| We cannot store NULL values in the primary key column. | We can store NULL value in the unique key column, but only one NULL is allowed. |
| We cannot change or delete the primary key column values. | We can modify the unique key column values. |

18.What are the different types of database management systems?

The database management systems can be categorized into several types. Some of the important lists are given below:

* Hierarchical databases (DBMS)
* Network databases (IDMS)
* Relational databases (RDBMS
* Object-oriented databases
* Document databases (Document DB)
* Graph databases
* ER model databases
* NoSQL databases

19.What is Normalization in a Database?

Normalization is used to minimize redundancy and dependency by organizing fields and table of a database.

There are some rules of database normalization, which is commonly known as Normal From, and they are:

* First normal form(1NF)



* Second normal form(2NF)

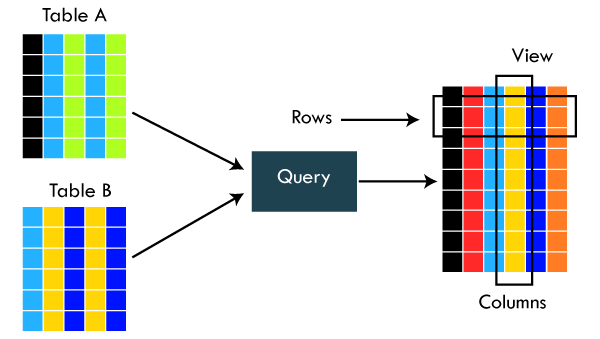


* Third normal form(3NF)
* Boyce-Codd normal form(BCNF)

Using these steps, the redundancy, anomalies, inconsistency of the data in the database can be removed.

20.What is a view in SQL?

A view is a database object that has no values. It is a virtual table that contains a subset of data within a table. It looks like an actual table containing rows and columns, but it takes less space because it is not present physically. It is operated similarly to the base table but does not contain any data of its own. Its name is always unique. A view can have data from one or more tables. If any changes occur in the underlying table, the same changes reflected in the views also.





The primary use of a view is to implement the security mechanism. It is the searchable object where we can use a query to search the view as we use for the table. It only shows the data returned by the query that was declared when the view was created.

We can create a view by using the following syntax:

1. CREATE VIEW view\_name AS
2. SELECT column\_lists FROM table\_name
3. WHERE condition;

### 21.What is an Index in SQL?

An index is a disc structure associated with a table or view that speeds up row retrieval. It reduces the cost of the query because the query's high cost will lead to a fall in its performance. It is used to increase the performance and allow faster retrieval of records from the table. Indexing reduces the number of data pages we need to visit to find a particular data page. It also has a unique value meaning that the index cannot be duplicated. An index creates an entry for each value which makes it faster to retrieve data.

**For example:** Suppose we have a book which carries the details of the countries. If you want to find out information about India, why will you go through every page of that book? You could directly go to the index. Then from the index, you can go to that particular page where all the information about India is given.

22.What are the differences between SQL, MySQL, and SQL Server?

The following comparison chart explains their main differences:

|  |  |  |
| --- | --- | --- |
| **SQL** | **MySQL** | **SQL Server** |
| SQL or Structured Query Language is useful for managing our relational databases. It is used to query and operate the database. | MySQL is the popular database management system used for managing the relational database. It is a fast, scalable, and easy-to-use database. | SQL Server is an RDBMS database system mainly developed for the Windows system to store, retrieve, and access data requested by the developer. |
| SQL first appeared in 1974. | MySQL first appeared on May 23, 1995. | SQL Server first appeared on April 24, 1989. |
| SQL was developed by IBM Corporation. | MySQL was developed by Oracle Corporation. | SQL Server was developed by Microsoft Company. |
| SQL is a query language for managing databases. | MySQL is database software that uses SQL language to conduct with the database. | SQL Server is also a software that uses SQL language to conduct with the database. |
| SQL has no variables. | MySQL can use variables constraints and data types. | SQL Server can use variables constraints and data types. |
| SQL is a programming language, so that it does not get any updates. Its commands are always fixed and remain the same. | MySQL is software, so it gets frequent updation. | SQL Server is also software, so it gets frequent updation. |

23.What is the difference between SQL and PL/SQL?

The following comparison chart explains their main differences:

|  |  |
| --- | --- |
| **SQL** | **PL/SQL** |
| SQL is a database structured query language used to communicate with relational databases. It was developed by IBM Corporations and first appeared in 1974. | PL/SQL or Procedural Language/Structured Query Language is a dialect of SQL used to enhance the capabilities of SQL. Oracle Corporation developed it in the early 90's. It uses SQL as its database language. |
| SQL is a declarative and data-oriented language. | PL/SQL is a procedural and application-oriented language. |
| SQL has no variables. | PL/SQL can use variables constraints and data types. |
| SQL can execute only a single query at a time. | PL/SQL can execute a whole block of code at once. |
| SQL query can be embedded in PL/SQL. | PL/SQL cannot be embedded in SQL as SQL does not support any programming language and keywords. |
| SQL can directly interact with the database server. | PL/SQL cannot directly interact with the database server. |
| SQL is like the source of data that we need to display. | PL/SQL provides a platform where SQL data will be shown. |

### 24.Is it possible to sort a column using a column alias?

Yes. We can use the alias method in the ORDER BY instead of the WHERE clause for sorting a column.

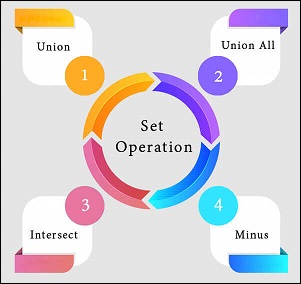
25.What are the different types of joins in SQL?

Joins are used to merge two tables or retrieve data from tables. It depends on the relationship between tables. According to the ANSI standard, the following are the different types of joins used in SQL:

* INNER JOIN
* SELF JOIN
* LEFT OUTER JOIN
* RIGHT OUTER JOIN
* FULL OUTER JOIN
* CROSS JOIN
* NATURAL JOIN

26.What are the set operators in SQL?

We use the set operators to merge data from one or more tables of the same kind. Although the set operators are like SQL joins, there is a significant distinction. SQL joins combine columns from separate tables, whereas SQL set operators combine rows from different queries. SQL queries that contain set operations are called compound queries. The set operators in SQL are categories into four different types:





**A. UNION:** It combines two or more results from multiple SELECT queries into a single result set. It has a default feature to remove the duplicate rows from the tables. The following syntax illustrates the Union operator:

1. SELECT columns FROM table1
2. UNION
3. SELECT columns FROM table2;

**B. UNION ALL:** This operator is similar to the Union operator, but it does not remove the duplicate rows from the output of the SELECT statements. The following syntax illustrates the UNION ALL operator:

1. SELECT columns FROM table1
2. UNION  ALL
3. SELECT columns FROM table2;

**C. INTERSECT:** This operator returns the common records from two or more SELECT statements. It always retrieves unique records and arranges them in ascending order by default. Here, the number of columns and data types should be the same. The following syntax illustrates the INTERSECT operator:

1. SELECT columns FROM table1
2. INTERSECT
3. SELECT columns FROM table2;

**D. MINUS:** This operator returns the records from the first query, which is not found in the second query. It does not return duplicate values. The following syntax illustrates the MINUS operator:

1. SELECT columns FROM table1
2. MINUS
3. SELECT columns FROM table2;

27.What is the difference between DELETE and TRUNCATE statements in SQL?

The main difference between them is that the delete statement deletes data without resetting a table's identity, whereas the truncate command resets a particular table's identity. The following comparison chart explains it more clearly:

|  |  |  |
| --- | --- | --- |
| **No.** | **DELETE** | **TRUNCATE** |
| 1) | The delete statement removes single or multiple rows from an existing table depending on the specified condition. | The truncate command deletes the whole contents of an existing table without the table itself. It preserves the table structure or schema. |
| 2) | DELETE is a **DML command.** | TRUNCATE is a **DML command.** |
| 3) | We **can use the WHERE** clause in the DELETE command. | We **cannot use the WHERE** clause with TRUNCATE. |
| 4) | DELETE statement is used **to delete a row** from a table. | TRUNCATE statement is used **to remove all the rows** from a table. |
| 5) | DELETE is **slower** because it maintained the log. | TRUNCATE statement is **faster** than DELETE statement as it deletes entire data at a time without maintaining transaction logs. |
| 6) | You **can roll back** data after using the DELETE statement. | It is **not possible to roll back** after using the TRUNCATE statement. |
| 7) | DELETE query **takes more space.** | TRUNCATE query **occupies less space.** |

28.What is the difference between the WHERE and HAVING clauses?

The main difference is that the WHERE clause is used to filter records before any groupings are established, whereas the HAVING clause is used to filter values from a group. The below comparison chart explains the most common differences:

|  |  |
| --- | --- |
| **WHERE** | **HAVING** |
| This clause is implemented in row operations. | This clause is implemented in column operations. |
| It does not allow to work with aggregate functions. | It can work with aggregate functions. |
| This clause can be used with the SELECT, UPDATE, and DELETE statements. | This clause can only be used with the SELECT statement. |

29.What are SQL comments?

Comments are explanations or annotations in SQL queries that are readable by programmers. It's used to make SQL statements easier to understand for humans. During the parsing of SQL code, it will be ignored. Comments can be written on a single line or across several lines.

* **Single Line Comments:** It starts with two consecutive hyphens (--).
* **Multi-line Comments:** It starts with /\* and ends with \*/.

### ****30.What is DBMS and RDBMS? Explain the difference between them.****

A database management system or [DBMS](https://intellipaat.com/blog/what-is-dbms/) is system software that can create, retrieve, update, and manage a database. It ensures the consistency of data and sees to it that it is organized and easily accessible by acting as an interface between the database and its end-users or application software. DBMS can be classified into four types:

* **Hierarchical Database:** It has a treelike structure with the data being stored in a hierarchical format. The parent in a database can have multiple children, but a child can have only a single parent.
* **Network Database:** This type of database is presented as a graph that can have many-to-many relationships allowing children to have multiple children.
* **Relational Database:** It is the most widely used and easy-to-use database. It is represented as a table and the values in the columns and rows are related to each other.
* **Object-oriented Database:** The data values and operations are stored as objects in this type of database, and these objects have multiple relationships among them.

RDBMS stores data in the form of a collection of [tables](https://intellipaat.com/blog/tutorial/sql-tutorial/tables-in-sql/). The relations are defined between the common fields of these tables. MS SQL Server, MySQL, IBM DB2, Oracle, and [Amazon Redshift](https://intellipaat.com/blog/what-is-amazon-redshift-in-aws/) are all based on RDBMS.

**DBMS vs RDBMS**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **DBMS** | **RDBMS** |
| **Access** | Data elements need to be accessed separately | Multiple data elements can be accessed at the same time |
| **Relationship Between Data** | No relationship between data | Data in tables is related to each other |
| **Normalization** | It is not present | It is present |
| **Distributed Database** | It does not support distributed database | It supports distributed database |
| **Data Storage Format** | Data is stored in either a navigational or hierarchical form | Data is stored in a tabular structure with headers being the column names and the rows containing corresponding values |
| **Amount of Data** | It deals with a small quantity of data | It deals with a larger amount of data |
| **Data Redundancy** | It is prevalent | Keys and indexes do not allow data redundancy |
| **Number of Users** | It supports a single user | It supports multiple users |
| **Data Fetching** | It is slower for large amounts of data | It is speedy due to the relational approach |
| **Data Security** | Low-security levels when it comes to data manipulation | Multiple levels of data security exist |
| **Software and Hardware Requirements** | Low | High |
| **Examples** | XML, Window Registry, etc. | MySQL, SQL Server, Oracle, Microsoft Access, PostgreSQL, etc. |
|  |  |  |

**31.DIFFERENCE BETWEEN SQL AND NOSQL ?**

|  |  |  |
| --- | --- | --- |
| **Index** | **SQL** | **NoSQL** |
| 1) | Databases are categorized as Relational Database Management System (RDBMS). | NoSQL databases are categorized as Non-relational or distributed database system. |
| 2) | SQL databases have fixed or static or predefined schema. | NoSQL databases have dynamic schema. |
| 3) | SQL databases display data in form of tables so it is known as table-based database. | NoSQL databases display data as collection of key-value pair, documents, graph databases or wide-column stores. |
| 4) | SQL databases are vertically scalable. | NoSQL databases are horizontally scalable. |
| 5) | SQL databases use a powerful language "Structured Query Language" to define and manipulate the data. | In NoSQL databases, collection of documents are used to query the data. It is also called unstructured query language. It varies from database to database. |
| 6) | SQL databases are best suited for complex queries. | NoSQL databases are not so good for complex queries because these are not as powerful as SQL queries. |
| 7) | SQL databases are not best suited for hierarchical data storage. | NoSQL databases are best suited for hierarchical data storage. |
| 8) | MySQL, Oracle, Sqlite, PostgreSQL and MS-SQL etc. are the example of SQL database. | MongoDB, BigTable, Redis, RavenDB, Cassandra, Hbase, Neo4j, CouchDB etc. are the example of nosql database |