Database:  
a database is an organized collection of data stored and accessed electronically from a computer system.  
  
DBMS:  
The database management system (DBMS) is the software that interacts with end users, applications, and the database itself to capture and analyze the data.   
  
Because of the close relationship between them, the term "database" is often used casually to refer to both a database and the DBMS used to manipulate it.  
  
A relational database is a digital database based on the relational model of data, as proposed by E. F. Codd in 1970. A system used to maintain relational databases is a relational database management system (RDBMS).   
  
--------------------------------------------------------------------------------------------------------------------------  
< Codd's 12 Rules >  
1.Information Rule:  
All data ( user or meta) must be a cell of a table (tabular format)   
  
2.Guaranteed Access Rule:  
Every single data element (value) is guaranteed to be accessible logically with a combination of table-name, primary-key (row value), and attribute-name (column value). not by other means like pointers  
  
3. Systematic and Uniform Treatment of NULL Values :  
a NULL can be interpreted as one the following − data is missing, data is not known, or data is not applicable.  
  
4. Active Online Catalog :  
The structure description of the entire database must be stored in an online catalog, known as data dictionary, which can be accessed by authorized users. Users can use the same query language to access the catalog which they use to access the database itself.  
  
5. Comprehensive Data Sub-Language Rule :  
A database can only be accessed using a language having linear syntax that supports data definition, data manipulation, and transaction management operations.  
  
6. View Updating Rule :  
All the views of a database, which can theoretically be updated, must also be updatable by the system.  
  
7. High-Level Insert, Update, and Delete Rule :  
it must also support union, intersection and minus operations to yield sets of data records.  
  
8. Physical Data Independence :  
Any change in the physical structure of a database must not have any impact on how the data is being accessed by external applications.  
  
  
10. Integrity Independence  
A database must be independent of the application that uses it. All its integrity constraints can be independently modified without the need of any change in the application. This rule makes a database independent of the front-end application and its interface.  
  
11. Distribution Independence  
The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only. This rule has been regarded as the foundation of distributed database systems.  
  
12. Non-Subversion Rule  
If a system has an interface that provides access to low-level records, then the interface must not be able to subvert the system and bypass security and integrity constraints.  
--------------------------------------------------------------------------------------------------------------------------  
Normalization :  
There are a few rules for database normalization. Each rule is called a "normal form."  
  
First normal form  
• Eliminate repeating groups in individual tables.  
• Create a separate table for each set of related data.  
• Identify each set of related data with a primary key.  
  
Second normal form  
\* Create separate tables for sets of values that apply to multiple records.  
\* Relate these tables with a foreign key.

Third normal form

\*Eliminate fields that do not depend on the key.

--------------------------------------------------------------------------------------------------------------------------

Types of databases :  
1. Centralized :   
A centralized database is one that operates entirely within a single location.   
  
2. Cloud database :  
A cloud database is one that runs over the Internet. The data is stored on a local hard drive or server, but the information is available online. Encryption is an essential part of any cloud database, as all information needs to be protected as it is transmitted online.   
  
3. Commercial database  
A commercial database is any that is designed by a commercial business. Businesses develop feature-rich databases, which they then sell to their customers.   
  
4. Distributed database :   
A distributed database is one that is spread out over multiple devices.   
distributed databases will operate across multiple machines, such as different computers within the same location or across a network. The benefits of a distributed database include increased speed, better reliability and ease of expansion.  
  
5. End-user database :  
a database that is primarily used by a single person. A good example of this type of database is a spreadsheet stored on your local computer.  
  
6. Graph database :  
databases that focus equally on the data and the connections between them. In this database, data is not constricted to predefined models. Most other databases can find connections between data when you run a search. With a graph database, these connections are stored inside the database right alongside the original data. This makes for a more efficient and faster database when your primary goal is to manage the connections between your data.  
  
7. NoSQL database :  
A NoSQL database has a hierarchy similar to a file folder system and the data within it is unstructured, or non-relational.   
  
8. Object-oriented database :  
Object-oriented databases are a type of relational database in which data is represented as objects and classes.  
  
9. Open-source database :  
An open-source database is designed for the public to use for free.  
  
10. Operational database :  
The purpose of an operational database is to allow users to modify data in real time. Operational databases are critical in business analytics and data warehousing. They can be set up either as relational databases or NoSQL, depending on needs. Conventional databases rely on batch processing, where commands are carried out in groups. Operational databases, on the other hand, allow you to add, edit and remove data at any moment.  
  
11. Personal databse :  
A personal database is one that is designed for a single person. It is typically stored on a personal computer and has a very simple design, consisting of only a few tables  
  
12. Relational :  
Relational databases are the other major type of database, opposite of NoSQL. With a relational database, information is stored structured about other data.  
  
--------------------------------------------------------------------------------------------------------------------------  
Different types of NoSql :  
1- Document databases  
2- Graph databases  
3- Key-value stores  
4- Wide column stores : It uses tables, rows, and columns, but unlike a relational database, the names and format of the columns can vary from row to row in the same table.  
--------------------------------------------------------------------------------------------------------------------------  
Benefits of SQL :  
1. update, organize and query the data quickly  
2. access specific information easily  
3. a universal language for SQL users, driving consistency in processes and problem-solving methods across many technical applications.  
4. Ease of access ( without having special programming skill)  
5. The interactive and straightforward design of the SQL language makes it easy to collect data and receive answers to complex queries.   
  
Benefits of NonSQL :  
1. Easier to expand  
2. store massive quantities of data without a prefixed technical structure  
3. large degree of scalability and flexibility  
4. ability to store and access information without having to define it makes it easy for NoSQL to search for and return information at high speeds  
  
--------------------------------------------------------------------------------------------------------------------------

NoSQL vs. SQL databases :  
1. Types and structures :  
SQL and NoSQL databases use different structures to perform processes. Typically, SQL databases feature tables while NoSQL databases are often document-based.  
  
2. Data storage models  
In an SQL database, you store data in a schema-based model where you must define data types before you can input information, NoSQL instead uses a schema-less model.  
SQL grows up vertically and building on previous files  
NoSQL branches out horizontally to store data within additional networks, servers and devices.  
  
3. Replication capabilities :  
Companies that use NoSQL often use it for the automatic replication of important data in the network.SQL makes automatic replication challenging  
  
4. Scaling possibilities:  
SQL databases are vertically scalable which means users can increase the load on a single instance server by adding additional RAM, CPU, or solid-state drives. NoSQL databases scale horizontally, allowing users to create database shards. Database shards partition data on separate database instances to spread the workload across multiple instances to increase the speed and availability of data.  
--------------------------------------------------------------------------------------------------------------------------  
create database in sql server :  
CREATE DATABASE <database name> ;  
Or   
USE master;   
GO   
CREATE DATABASE Sales   
ON   
( NAME = Sales\_dat,   
FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\saledat.mdf',   
SIZE = 10,   
MAXSIZE = 50,   
FILEGROWTH = 5 )   
LOG ON   
( NAME = Sales\_log,   
FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\salelog.ldf',   
SIZE = 5MB,   
MAXSIZE = 25MB,   
FILEGROWTH = 5MB );   
GO   
  
Master database :  
The master database records all the system-level information for a SQL Server system. This includes instance-wide metadata such as logon accounts, endpoints, linked servers, and system configuration settings. In SQL Server, system objects are no longer stored in the master database; instead, they are stored in the Resource database. Also, master is the database that records the existence of all other databases and the location of those database files and records the initialization information for SQL Server. Therefore, SQL Server cannot start if the master database is unavailable.  
  
--------------------------------------------------------------------------------------------------------------------------  
Back-up :  
back up [verb]  
Copies the data or log records from a SQL Server database or its transaction log to a backup device, such as a disk, to create a data backup or log backup.  
  
backup [noun]  
A copy of SQL Server data that can be used to restore and recover the data after a failure. A backup of SQL Server data is created at the level of a database or one or more of its files or filegroups. Table-level backups cannot be created. In addition to data backups, the full recovery model requires creating backups of the transaction log.  
  
recovery model  
A database property that controls transaction log maintenance on a database. Three recovery models exist: simple, full, and bulk-logged. The recovery model of database determines its backup and restore requirements.  
  
restore  
A multi-phase process that copies all the data and log pages from a specified SQL Server backup to a specified database, and then rolls forward all the transactions that are logged in the backup by applying logged changes to bring the data forward in time.  
--------------------------------------------------------------------------------------------------------------------------  
  
Types of backups  
copy-only backup  
A special-use backup that is independent of the regular sequence of SQL Server backups.  
  
data backup  
A backup of data in a complete database (a database backup), a partial database (a partial backup), or a set of data files or filegroups (a file backup).  
  
database backup  
A backup of a database. Full database backups represent the whole database at the time the backup finished. Differential database backups contain only changes made to the database since its most recent full database backup.  
  
differential backup  
A data backup that is based on the latest full backup of a complete or partial database or a set of data files or filegroups (the differential base) and that contains only the data extents that have changed since the differential base.  
  
A differential partial backup records only the data extents that have changed in the filegroups since the previous partial backup, known as the base for the differential.  
  
full backup  
A data backup that contains all the data in a specific database or set of filegroups or files, and also enough log to allow for recovering that data.  
  
log backup  
A backup of transaction logs that includes all log records that were not backed up in a previous log backup. (full recovery model)  
  
file backup  
A backup of one or more database files or filegroups.  
  
partial backup  
Contains data from only some of the filegroups in a database, including the data in the primary filegroup, every read/write filegroup, and any optionally-specified read-only files.  
--------------------------------------------------------------------------------------------------------------------------  
A restore scenario in SQL Server is the process of restoring data from one or more backups and then recovering the database. The supported restore scenarios depend on the recovery model of the database and the edition of SQL Server.  
--------------------------------------------------------------------------------------------------------------------------  
Attach and detach :  
to move a detached database to another location and re-attach it to the same or a different server instance in SQL Server.  
--------------------------------------------------------------------------------------------------------------------------  
Transcation :   
A transaction basically represents a change in the database. It can contain a single or multiple operations. A transaction is treated as a whole even if it contains multiple operations. After a transaction is completed, the state of the database changes.  
  
database properties:  
  
ACID:  
Atomicity :   
An atomic transaction either happens or does not happen at all.  
  
Consistency :  
Consistency means that a transaction must be compatible with the constraints on data integrity. If a transaction violates a constraint, consistency ensures that the transaction is not executed and the database goes back to the previous stable state.  
  
Isolation :  
Isolation is used to handle concurrent transactions.   
  
Durability :  
Durability ensures that once a transaction is completed, the change in the database due to this transaction is permanent. It will not be affected by the system failures such as power outages.  
The completed transactions are stored in non-volatile memory.   
--------------------------------------------------------------------------------------------------------------------------  
mdf and ldf :  
A Microsoft SQL database consists a primary data file (mdf) a secondary data file (ndf) and a transaction log file (ldf).  
MDF stands for Main Database File and contains all the information in a database. LDF records all the transactions and changes to the database. The ldf is critical for disaster recovery.  
Setting the Initial size and Auto growth for these files, can have a significant impact on the performance or even the ability to use the database.  
Auto growth is even more important.   
--------------------------------------------------------------------------------------------------------------------------  
Schema :  
A database schema is a blueprint or architecture of how our data will look. It doesn’t hold data itself, but instead describes the shape of the data and how it might relate to other tables or models. An entry in our database will be an instance of the database schema. It will contain all of the properties described in the schema.