**Question No-01: What is Data?**

In general, **data** is any set of [characters](https://www.computerhope.com/jargon/c/charact.htm) that is gathered and translated for some purpose, usually analysis. If data is not put into context, it doesn't do anything to a human or computer.

There are multiple types of data. Some of the more common types of data include the following:

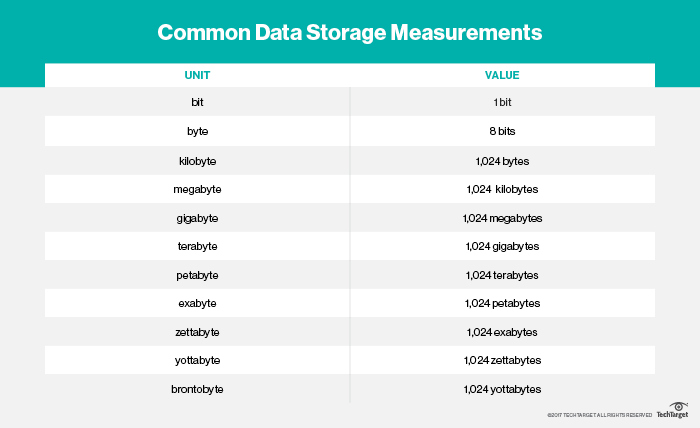
* Single character
* [Boolean](https://www.computerhope.com/jargon/b/boolean.htm) (true or false)
* Text ([string](https://www.computerhope.com/jargon/s/string.htm))
* Number ([integer](https://www.computerhope.com/jargon/i/integer.htm) or [floating-point](https://www.computerhope.com/jargon/f/floapoin.htm))
* [Picture](https://www.computerhope.com/jargon/p/picture.htm)
* [Sound](https://www.computerhope.com/jargon/s/sound.htm)
* [Video](https://www.computerhope.com/jargon/v/video.htm)

In a computer's storage, digital data is a sequence of [bits](https://www.computerhope.com/jargon/b/bit.htm) ([binary digits](https://www.computerhope.com/jargon/b/binary.htm)) that have the value one or zero. Data is processed by the [CPU](https://www.computerhope.com/jargon/c/cpu.htm), which uses [logical](https://www.computerhope.com/jargon/l/logic.htm) operations to produce new data ([output](https://www.computerhope.com/jargon/o/output.htm)) from source data ([input](https://www.computerhope.com/jargon/i/input.htm)).

**How data is stored:**

Computers represent data, including video, images, sounds and text, as binary values using patterns of just two numbers: 1 and 0. A [bit](https://www.techtarget.com/whatis/definition/bit-binary-digit) is the smallest unit of data, and represents just a single value. A byte is eight binary digits long. Storage and memory is measured in [megabytes](https://www.techtarget.com/searchstorage/definition/megabyte) and [gigabytes](https://www.techtarget.com/searchstorage/definition/gigabyte).

The units of data measurement continue to grow as the amount of data collected and stored grows. The relatively new term "[brontobyte](https://www.techtarget.com/searchstorage/definition/brontobyte)," for example, is [data storage](https://www.techtarget.com/searchstorage/definition/storage) that is equal to 10 to the 27th power of [bytes](https://www.techtarget.com/searchstorage/definition/byte).



**Question – 02:**

**What is Database?**

A database is a systematic collection of data. They support electronic storage and manipulation of data. Databases make data management easy.

Let us discuss a database example: An online telephone directory uses a database to store data of people, phone numbers, and other contact details. Your electricity service provider uses a database to manage billing, client-related issues, handle fault data, etc.

Let us also consider Facebook. It needs to store, manipulate, and present data related to members, their friends, member activities, messages, advertisements, and a lot more. We can provide a countless number of examples for the usage of databases.

## Types of Databases

Here are some popular types of databases.

* Distributed Database
* Relational Database
* Object-Oriented Database
* Centralized Database
* Open-Source Database
* Cloud Database
* Data Warehouse
* NoSql Database
* Graph Database
* OLTP Database
* Personal Database
* Multimodal Database
* Document/JSON Database
* Hierarchical Database

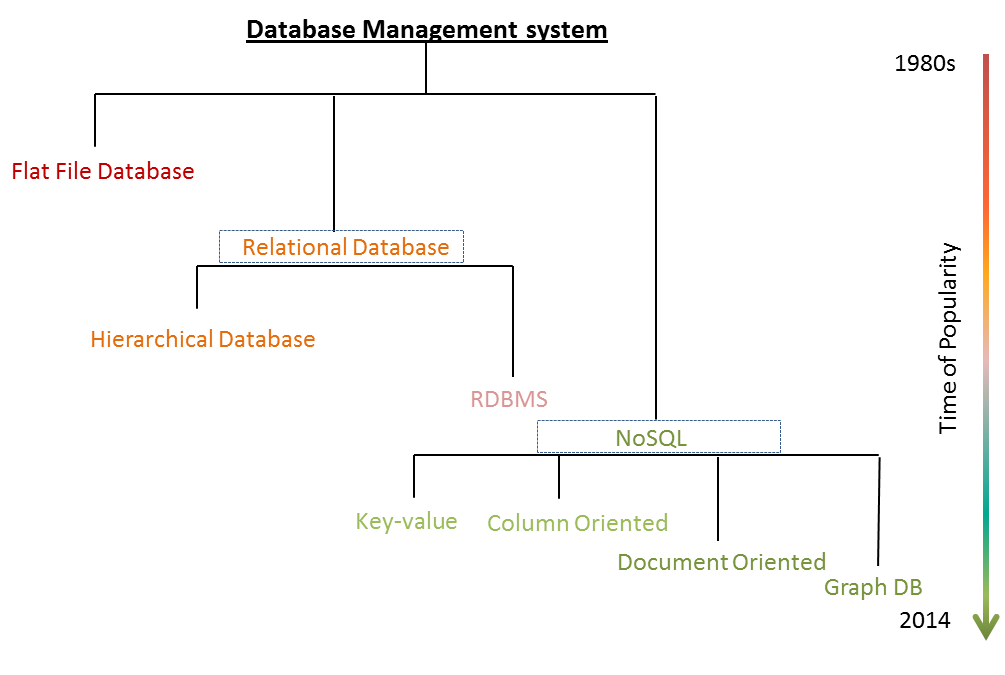
**Question-03:**

**Evolution of Database:**

A database management system is a set of programs that allows users to create and maintain a data base and interact with various users, including data base administrators, online users, application programmers, and naive users, as well as other users.

Hence we can say that basically database is a compilation of database files and each database file is further a collection of records.

**Database models:**

****

**1) Flat files (1960s – 1980s)**

**Implementation of a flat file database**

Flat file database is implemented in:

* Berkeley DB
* SQLite
* Mimesis
* TheIntegrationEngineer etc.

**2) Hierarchical database (1970s – 1990s)**

**3)** **Network database (1970s – 1990s)**

**Implementation of network database**

Network database is implemented in:

* Digital Equipment Corporation DBMS-10
* Digital Equipment Corporation DBMS-20
* RDM Embedded
* Turbo IMAGE
* Univac DMS-1100 etc.

**4)** **Relational database (1980s – present)**

Implementation of Relational Database:

* Oracle
* Microsoft
* IBM
* My SQL
* PostgreSQL
* SQLite

**5)** **Object-oriented database (1990s – present)**

**Question No-04:**

**NoSQL Database:**

**NoSQL database** technology stores information in JSON documents instead of columns and rows used by relational databases. To be clear, NoSQL stands for “not only SQL” rather than “no SQL” at all. This means a NoSQL JSON database can store and retrieve data using literally “no SQL.” Or you can combine the flexibility of JSON with the power of SQL for the best of both worlds. Consequently, NoSQL databases are built to be flexible, scalable, and capable of rapidly responding to the data management demands of modern businesses. The following defines the four most-popular types of NoSQL database:

* **Document databases** are primarily built for storing information as documents, including, but not limited to, JSON documents. These systems can also be used for storing XML documents, for example.
* **Key-value stores** group associated data in collections with records that are identified with unique keys for easy retrieval. Key-value stores have just enough structure to mirror the value of relational databases while still preserving the benefits of NoSQL.
* **Wide-column databases** use the tabular format of relational databases yet allow a wide variance in how data is named and formatted in each row, even in the same table. Like key-value stores, wide-column databases have some basic structure while also preserving a lot of flexibility
* **Graph databases** use graph structures to define the relationships between stored data points. Graph databases are useful for identifying patterns in unstructured and semi-structured information.

## Why use NoSQL?

 Customer experience has quickly become the most important competitive differentiator and ushered the business world into an era of monumental change. As part of this revolution, enterprises are interacting digitally – not only with their customers, but also with their employees, partners, vendors, and even their products – at an unprecedented scale. This interaction is powered by the internet and other 21st century technologies – and at the heart of the revolution are a company’s cloud, mobile, social media, big data, and IoT applications.

How are these applications different from legacy enterprise applications like ERP, HR, and financial accounting? Today’s web, mobile, and IoT applications share one or more (if not all) of the following characteristics. They need to:

 Support large numbers of concurrent users (tens of thousands, perhaps millions)

* Deliver highly responsive experiences to a globally distributed base of users
* Be always available – no downtime
* Handle semi- and unstructured data
* Rapidly adapt to changing requirements with frequent updates and new features