1. What is NoSQL data base?

Ans: NoSQL is an approach to databases that represents a shift away from traditional relational database management systems (RDBMS). NoSQL can mean “not SQL” or “not only SQL.” As RDBMS have increasingly failed to meet the performance, scalability, and flexibility needs that next-generation, data-intensive applications require, NoSQL databases have been adopted by mainstream enterprises. NoSQL is particularly useful for storing unstructured data, which is growing far more rapidly than structured data and does not fit the relational schemas of RDBMS. A **NoSQL** (originally referring to "non SQL", "non relational" or "not only SQL")[[1]](https://en.wikipedia.org/wiki/NoSQL#cite_note-1) database provides a mechanism for [storage](https://en.wikipedia.org/wiki/Computer_data_storage) and [retrieval](https://en.wikipedia.org/wiki/Data_retrieval) of data which is modeled in means other than the tabular relations used in [relational databases](https://en.wikipedia.org/wiki/Relational_database).

1. How does data get stored in NoSQl database?

## Ans: Their exists various types of NoSQL Database Types and way data are stored also differs in each of them.

* Document databases  pair each key with a complex data structure known as a document. Documents can contain many different key-value pairs, or key-array pairs, or even nested documents.
* Graph stores store information about networks of data, such as social connections and uses grapgh data structures. Graph stores include Neo4J and Giraph.
* Key-value stores are the simplest NoSQL databases. Every single item in the database is stored as an attribute name (or 'key'), together with its value. Examples of key-value stores are Riak and Berkeley DB. Some key-value stores, such as Redis, allow each value to have a type, such as 'integer', which adds functionality.
* Wide-column stores such as Cassandra and HBase are optimized for queries over large datasets, and store columns of data together, instead of rows. Here we have rowkey and corresponding column family which further has column qualifiers

1. What is a column family in HBase?

Ans : Columns in Apache HBase are grouped into column families. All column members of a column family have the same prefix. For example, the columns courses:history and courses:math are both members of the courses column family. The colon character (:) delimits the column family from the . The column family prefix must be composed of printable characters. The qualifying tail, the column family qualifier, can be made of any arbitrary bytes. Column families must be declared up front at schema definition time whereas columns do not need to be defined at schema time but can be conjured on the fly while the table is up an running.

1. How many maximum number of columns can be added to HBase table?

Ans:

1. Why columns are not defined at the time of table creation in HBase?

Ans:

1. How does data get managed in HBase?

Ans: Data in Hbase is organized into tables. Any characters that are legal in file paths are used to name tables. Tables are further organized into rows that store data. Each row is identified by a unique row key which does not belong to any data type but is stored as a bytearray. Column families are further used to group data in rows. Column families define the physical structure of data so they are defined upfront and their modification is difficult. Each row in a table has same column families. Data in a column family is addressed using a column qualifier. It is not necessary to specify column qualifiers in advance and there is no consistency requirement between rows. No data types are specified for column qualifiers, as such they are just stored as bytearrays. A unique combination of row key, column family and column qualifier forms a cell. Data contained in a cell is referred to as cell value. There is no concept of data type when referring to cell values and they are stored as bytearrays. Versioning happens to cell values using a timestamp of when the cell was written.

1. What happens internally when new data gets inserted into HBase table?

Ans:

1. When the client issues a Put request, the first step is to write the data to the write-ahead log, the WAL
2. Once the data is written to the WAL, it is placed in the MemStore. Then, the put request acknowledgement returns to the client.
3. The MemStorestores updates in memory as sorted KeyValues, the same as it would be stored in an HFile.