1. What is NoSQL database?

A NoSQL database supports processing semi-structured or non-structured data. NoSQL databases are different from the typical relational database management system (RDBMS). One difference is NoSQL databases do not require joins to link multiple tables for the desired result whereas a RDBMS database requires the data to be structured in a table format (rows and columns). Tables in a RDBMS database are joined as needed to produce an output that meets the business object. Another difference is horizontal scaling. Horizontal scaling is a problem with RDBMS databases because it requires the data to spread across a number of serves to obtain the breath of scaling. NoSQL databases support horizontal scaling with ease because it is designed for clustering on commodity servers. Last, as data becomes less structured and larger in size, RDBMS databases become less efficient. NoSQL overcomes the data size problem b6 using a clustering design pattern and the non-structured-data problem because of how data is stored within the database. NoSQL databases have the advantage of supporting large volumes of information that might be less structured than RDBMS. This advantage enables data architects to leverage data in its native form to perform data processing, analysis, and queries.

2. How does data get stored in NoSQL database?

Data in NoSQL databases are stored as a single table. This table can be altered dynamically to support additional types of data. NoSQL data is stored using a columnar format.

3. What is a column family in HBase?

A column family in HBase is the technique for categorizing columns of data within the database. HBase has no strict recommendations to the number of columns defined within the column family. However, HBase does recommend having as few column families as possible; the fewer the better. The recommendation for fewer columns families is for performance reasons.

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

In HBase, there is no maximum number of columns that case be added to the HBase table.

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

Columns are not defined at the time of table creation because HBase is designed to support a varying number of columns within the column family. In other words, each row within a column family does not have to support the same number of columns as adjoining rows. Supporting a varying number of columns allow the designer to create new columns on the fly as needed to support the use-case. Therefore, predefining the columns at table creation is unnecessary in HBase.

6. How does data get managed in HBase?

In HBase, data is managed within Hfiles using a key/value mapping strategy. When data is updated, it is first committed to the write-ahead logs (WAL) prior to writing to disk.

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

All table updates are first written to the WAL. Afterwards, the data is written to the memory store. After the data in the memory store exceeds a predefined value, then the data is flushed to the HFile on the disk and the commit logs are discarded.