**Assignment 9-1**

**1. What is NoSQL data base?**

NoSQL is an approach to database design that can accomodate a wide variety of data models, including key-value, document, columnar and graph formats. NoSQL, which stand for "not only SQL," is an alternative to traditional relational databases in which data is placed in tables and data schema is carefully designed before the database is built. NoSQL databases are especially useful for working with large sets of distributed data

**2. How does data get stored in NoSQl database?**

**There are various NoSQL Databases. Each one uses a different method to store data. Some might use column store, some document, some graph, etc., Each database has its own unique characteristics.**

* **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 are used to store information about networks of data, such as social connections. 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.

**3. What is a column family in HBase?**

In the HBase data model columns are grouped into column families, which must be defined up front during table creation. Column families are stored together on disk, which is why HBase is referred to as a column-oriented data store.

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.

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

There is no limit for coulmns.

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

Column families are part of the schema of the table. It can be added at runtime with an online schema change.

The reason column families are part of the schema and would require a schema change is that they profoundly impact the way the data is stored, both on disk and in memory. Each column family has its own set of HFiles, and its own set of data structures in memory of the RegionServer. It would be pretty expensive to dynamically create or start using new column families.

Column families are only needed when you need to configure differently various parts of a table (for instance you want some columns to have a TTL and others to not expire. So, again, because of those specialized reasons, it doesn't make sense to dynamically add new column families at runtime the way you would add regular "columns" within a family.

**6. How does data get managed in HBase?**

HBase contains two primary services:

**Master server**

The master server co-ordinates the cluster and performs administrative operations, such as assigning regions and balancing the loads.

**Region server**

The region servers do the real work. A subset of the data of each table is handled by each region server. Clients talk to region servers to access data in HBase.

**Regions**

Region servers manage a set of regions.

An HBase table is made up of a set of regions. Regions are the basic unit of work in HBase. It is what is used as a split by the map reduce framework. The region contains store objects that correspond to column families. There is one store instance for each column family. Store objects create one or more StoreFiles, which are wrappers around the actual storage file called the HFile.

The region also contains a MemStore, which is in-memory storage and is used as a write cache. Rows are written to the MemStore. The data in the MemStore is ordered. If the MemStore becomes full, it is persisted to an HFile on disk

To improve performance, it is important to get an even distribution of data among regions, which ensures the best parallelism in map tasks.

**HFiles**

HFiles are the physical representation of data in HBase. Clients do not read HFiles directly but go through region servers to get to the data.

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

HBase internally puts the data in indexed StoreFiles that exist on HDFS for high-speed lookups.Everything in HBase is stored as bytes and there are no types. There is no schema since each row in HBase can have a different set of columns.