1.Difference between Hbase and HDFS.

HDFS is a distributed file system and has the following properties:  
1. It is optimized for streaming access of large files. You would typically store files that are in the 100s of MB upwards on HDFS and access them through MapReduce to process them in batch mode.   
2. HDFS files are write once files. You can append to files in some of the recent versions but that is not a feature that is very commonly used. Consider HDFS files as write-once and read-many files. There is no concept of random writes.  
3. HDFS doesn't do random reads very well.  
  
HBase on the other hand is a database that stores it's data in a distributed filesystem. The filesystem of choice typically is HDFS owing to the tight integration between HBase and HDFS. Having said that, it doesn't mean that HBase can't work on any other filesystem. It's just not proven in production and at scale to work with anything except HDFS.  
HBase provides you with the following:  
1. Low latency access to small amounts of data from within a large data set. You can access single rows quickly from a billion row table.  
2. Flexible data model to work with and data is indexed by the row key.  
3. Fast scans across tables.  
4. Scale in terms of writes as well as total volume of data.

2. List and explain the main components of HBASE.

HBase has three major components:

* The client library
* A master server
* Region servers.

MasterServer

The master server -

* Assigns regions to the region servers and takes the help of Apache ZooKeeper for this task.
* Handles load balancing of the regions across region servers. It unloads the busy servers and shifts the regions to less occupied servers.
* Maintains the state of the cluster by negotiating the load balancing.
* Is responsible for schema changes and other metadata operations such as creation of tables and column families.

## Regions

Regions are nothing but tables that are split up and spread across the region servers.

### Region server

The region servers have regions that -

* Communicate with the client and handle data-related operations.
* Handle read and write requests for all the regions under it.
* Decide the size of the region by following the region size thresholds.

## Zookeeper

* Zookeeper is an open-source project that provides services like maintaining configuration information, naming, providing distributed synchronization, etc.
* Zookeeper has ephemeral nodes representing different region servers. Master servers use these nodes to discover available servers.
* In addition to availability, the nodes are also used to track server failures or network partitions.
* Clients communicate with region servers via zookeeper.
* In pseudo and standalone modes, HBase itself will take care of zookeeper.

3. Does Hbase support sql?

HBase is a column-oriented database management system that runs on top of [HDFS](http://www.ibm.com/software/data/infosphere/hadoop/hdfs/). It is well suited for sparse data sets, which are common in many big data use cases. Unlike relational database systems, HBase does not support a structured query language like SQL; in fact, HBase isn’t a relational data store at all. HBase applications are written in Java much like a typical [MapReduce](http://www.ibm.com/software/data/infosphere/hadoop/mapreduce/) application. HBase does support writing applications in Avro, REST, and Thrift.