**Answer 1:**

HDFS is a distributed file system and has the following properties:  
1. It is optimized for streaming access of large files.   
2. HDFS files are write once files.

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.

HBase provides the following features:  
1. Low latency access to small amounts of data from within a large data set. We 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.

**Answer2:**

1. Region Server
2. Hbase Master Server
3. ZooKeeper

**Answer3:**

No

**Answer4:**

1. Hbase is used when consistency of data needs to be provided.
2. Hbase provide high write throughput.
3. Hbase provide low latency read.
4. Hbase is used when high availability is required.

**Answer5:**

1. Standalone : In standalone mode , hbase uses the local filesystem. It runs all HBase daemons and a local ZooKeeper all up in the same JVM.
2. Distributed :

Distributed mode can be subdivided into pseudo distributed and fully distributes.

pseudo-distributed: all daemons run on a single node

 fully-distributed: the daemons are spread across all nodes in the cluster.

Distributed modes require an instance of the Hadoop Distributed File System (HDFS).

**Answer6:**

HBase uses ZooKeeper as a distributed coordination service to maintain server state in the cluster. Zookeeper maintains which servers are alive and available, and provides server failure notification

**Answer 8:**

Input data is not required to have fixed column schema to load to a table.

**Answer 9:**

Hbase should have minimum of 1 column family.

**Answer 10:**

1. Need to make single connection to Zookeeper to fetch meta table information and shared across all the clients.
2. Single fetch of table location information from meta table and shared across all the clients thus reducing the network round-trips.
3. Any update in meta information will be shared across all the clients sharing the same connection.