1. Explain HDFS federation and High availability

* HDFS federation:
* In hdfs federation, there are multiple namenodes, each storing metadata and block mapping of files and directories contained in particular sub-directories.
* It is often used to improve the existing hdfsarchitecture with the help of a clear separation of namespace and storage, enabling generic block storage layer.
* The list of sub-directories managed by a namenode is referred to as a namespace volume .
* Blocks of files belonging to a Namespace is called a block pool.
* It consists of 2 main layers:
* Namespace
* Block storage
* High availability:
* It was mainly introduced to overcome the single point failure problem(SPOF) of hadoop 1.0.
* It overcomes this SPOF shortcoming by providing support for multiple namenodes.
* The main motive of the hadoop 2.0 High Availability project is to render availability to big data applications 24/7 by deploying 2  hadoop namenodes.
* One in active configuration and the other is the Standby Node in passive configuration.
* With this feature the hadoop will be equipped to overcome the namenode failure problem.

1. How HDFS handles failures while writing data

* BLOCK RECOVERY:

If the last block of the file being written is not propagated to all DataNodes in the pipeline, then the amount of data written to different nodes may be different when lease recovery happens. Before lease recovery causes the file to be closed, it’s necessary to ensure that all replicas of the last block have the same length; this process is known as block recovery.

* PIPELINE RECOVERY:

During write pipeline operations, some DataNodes in the pipeline may fail. When this happens, the underlying write operations can’t just fail. Instead, HDFS will try to recover from the error to allow the pipeline to keep going and the client to continue to write to the file. The mechanism to recover from the pipeline error is called pipeline recovery.

* The pipeline is closed and any packets in the acknowledgement queue are added to the front of the data queue .
* The current block on the good DataNodes is given a new identity, which is communicated to the NameNode .
* The failed DataNode is removed from the pipeline, and a new pipeline is constructed from the two good DataNodes .
* The remainder of the block’s data is written to the good DataNodes in the pipeline .
* The NameNode notices that the block is under-replicated, and it arranges for a further replica to be created on another node .
* As long as dfs.namenode.replication.min replicas are written, the write will succeed .
* The block will be asynchronously replicated across the cluster until its target replication factor is reached (dfs.replication, which defaults to 3).