1. What is Default replication factor and how will you change it at file level?

Ans, Default Replication factor is 3, we can define it in hdfs-site.xml file <value> no of replication</value>

```
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
```

2. Why do we need replication factor > 1 in production Hadoop cluster?

Ans : We need replication factor for security/backup, fault taulerant purpose, in case one node get crashed we can go for second copy.

3. How will you combine the 4 part-r files of a mapreduce job?

Ans : hdfs dfs -cat <source-dir-on-hdfs/part-r*> >> CombinerResult.txt

4. What are the Compression techniques in HDFS and which is the best one and why?

Ans : gzip is naturally supported by Hadoop. gzip is based on the DEFLATE algorithm, which is a combination of LZ77 and Huffman Coding.

5. How will you view the compressed files via HDFS command?

Ans : we can use “hdfs dfs -tar compressedfile” command

6. What is Secondary Namenode and its Functionalities? why do we need it?

Ans : Namenode holds the metadata for HDFS like Block information, size etc. This Information is stored in main memory as well as disk for persistence storage .

The information is stored in 2 different files .They are

Editlogs- It keeps track of each and every changes to HDFS.

Fsimage- It stores the snapshot of the file system.

Any changes done to HDFS gets noted in the edit logos the file size grows where as the size of fsimage remains same. This not have any impact until we restart the server. When we restart the server the edit file logs are written into fsimage file and loaded into main memory which takes some

time. If we restart the cluster after a long time there will be a vast down time since the edit log file would have grown. Secondary namenode would come into picture in rescue of this problem.

Secondary Namenode simply gets edit logs from name node periodically and copies to fsimage. This new fsimage is copied back to namenode.Namenode now, this uses this new fsimage for next restart which reduces the startup time. It is a helper node to Namenode and to precise Secondary Namenode whole purpose is to have checkpoint in HDFS, which helps namenode to function effectively. Hence, It is also called as Checkpoint node.

7. What is Backup node and how is it different from Secondary namenode?

Ans : Backup Node in hadoop is an extended checkpoint node that performs checkpointing and also supports online streaming of file system edits. **Backup node does not need to download fsimage and edits files from the active NameNode** to create a checkpoint, as it already has an up-to-date state of the namespace in it's own main memory. So, creating checkpoint in backup node is just saving a copy of file system meta-data (namespace) from main-memory to its local files system.

8. What is FSimage and editlogs and how they are related?

Ans :

Fsimage- It stores the snapshot of the file system.

Editlogs- It keeps track of each and every changes to HDFS.

9. what is default block size in HDFS? and why is it so large?

Ans. 128 MB. Because of huge volume of data

10. How will you copy a large file of 50GB into HDFS in parallel

Ans: We can use hdfs dfs –put sourceFile /Destination

11. what is Balancing in HDFS?

Ans : The HDFS Balancer is a tool for balancing the data across the storage devices of a HDFS cluster. We can also specify the source DataNodes, to free up the spaces in particular DataNodes. We can use a block distribution application to pin its block replicas to particular DataNodes so that the pinned replicas are not moved for cluster balancing.

- Factors such as addition of DataNodes, block allocation in HDFS, and behavior of the client application can lead to the data stored in HDFS clusters becoming unbalanced.

12. What is expunge in HDFS ?

Ans : This command is used to empty the trash available in an HDFS system.

Syntax:

$ hadoop fs –expunge