A client application creates an hdfs file named foo.txt with a replication factor of 3. Identify what best describes the file access file rules in hdfs had a single block that is stored on the data nodes a ,b,c

A) Each node stores a copy of the file in the local file system with the same name as the hdfs

B) Each data node locks file to prohibit concurrent readers & writers of the file
C) The file can be accessed if at least one of the data nodes storing the file is available
D) The file will be marked as corrupted if the data node B fails during the creation of the file

ANS: C

You use the hadoop fs -put command to write a 300mb file using an hdfs block size of 64mb. Just after this command has finished writing 200mb of this file, what would another user see when trying to access this file?

A) They would see no content until the whole file is written and closed.

B) They would see the content of the file through the last completed block.

C) They would see the current state of the file, upto the last bit written by the command.

D) They would see hadoop throw an concurrent file access exception when they try to access this file.

ANS: B

Which describes how client reads file from hdfs?

A)The client contacts the namenode for the block location(s). The namenode contacts the datanode that holds the requested data block. Data is transferred from the datanode to the namenode and then from the namenode to the client.

B) The client contacts the namenode for the block location(s). The namenode then queries the datanode for block locations. the datanodes responds to the namenode and the namenode redirects the client to the datanode that holds the requested data block(s). The client then reads the data directly on the datanode.

C) The client queries all datanodes in parallel. the datanode that contains the requested data responds directly to the client. The client reads the data directly off the data node.

D)The client queries the name-node for the block location(s). The namenode returns the block location(s) to the client. The client reads the data directly off the datanode.

ANS: D

You need to copy a file titled "weblogs" into HDFS. When you try to copy the file, you can't. You know you have ample space on your datanodes.

A) increase the amount of memory for the namenode.

B) increase the number of disks (or size) for the namenode.

C) decrease the block size on your remaining files.
D) decrease the block size on all current files in HDFS.
E) increase the block size on your remaining files.

F) increase the block size on all current files in HDFS

.

Ans: A

A cluster has 12 datanodes, each of which has a single 1TB capacity for HDFS. With default settings, about how much user data can it hold?

1. 3TB
2. 12TB
3. 4TB
4. 6TB

Ans: 3

The nature of hardware for the namenode should be

A - Superior than commodity grade

B - Commodity grade

C - Does not matter

D - Just have more Ram than each of the data nodes

Ans: A

The purpose of checkpoint node in a Hadoop cluster is to

A - Check if the namenode is active

B - Check if the fsimage file is in sync between namenode and secondary namenode

C - Merges the fsimage and edit log and uploads it back to active namenode.

D - Check which data nodes are unreachable

ANS: C

Which of the below property gets configured on hdfs-site.xml ?

A - Replication factor

B - Directory names to store hdfs files.

C - Host and port where MapReduce task runs.

D - Java Environment variables

ANS: A

When you increase the number of files stored in HDFS, the memory required by namenode

A - Increases

B - Decreases

C - Remains unchanged

D - May increase or decrease

ANS: A

What are the steps involved in writing a file from local system to HDFS?

1. Client reads dfs.blockSize and create blocks
2. Client connects to Name Node using fs.defaultName from core-site.xml
3. Client sends File name and Block name to NN
4. Name Node writes the metadata and applies Replica placement Strategy using the replication factor
5. Name Node sends the IP addresses of data nodes to store blocks to Client.
6. Client copies the data to name nodes sequentially

Feature	Hadoop 2.x	Hadoop 3.x
Minimum Required Java Version	JDK 6 and above.	JDK 8 is the minimum runtime version of JAVA required to run Hadoop 3.x as many dependency library files have been used from JDK 8.
Fault Tolerance	Fault Tolerance is handled through replication leading to storage and network bandwidth overhead.	Support for Erasure Coding in HDFS improves fault tolerance
Storage Scheme	Follows a 3x Replication Scheme for data recovery leading to 200% storage overhead. For instance, if there are 8 data blocks then a total of 24 blocks will occupy the storage space because of the 3x replication scheme.	Storage overhead in Hadoop 3.0 is reduced to 50% with support for Erasure Coding. In this case, if here are 8 data blocks then a total of only 12 blocks will occupy the storage space.

Feature	Hadoop 2.x	Hadoop 3.x
Change in Port Numbers	Hadoop HDFS NameNode -8020 Hadoop HDFS DataNode - 50010 Secondary NameNode HTTP -50091	Hadoop HDFS NameNode -9820 Hadoop HDFS DataNode - 9866 Secondary NameNode HTTP -9869
Intra DataNode Balancing	HDFS Balancer in Hadoop 2.0 caused skew within a DataNode because of addition or replacement of disks.	Support for Erasure Coding in HDFS improves fault toleranceIntra DataNode Balancing has been introduced in Hadoop 3.0 to address the intra-DataNode skews which occur when disks are added or replaced.
Number of NameNodes	Hadoop 2.0 introduced a secondary namenode as standby.	Hadoop 3.0 supports 2 or more NameNodes.