Hadoop Distributed File System

Read Chapter 3 of the book "Hadoop – The Definitive Guide" and answer the following questions.

1. Understand the design considerations of HDFS and also when is it not a suitable solution.

HDFS Blocks

1. Why are block sizes in HDFS so large? If the total file size is 1000 MB and seek time for each block is 10 ms and transfer rate is 100 MB/s, calculate the total file transfer time assuming a block size of 128 MB? If you double the block size, what would happen to the total time?

2. What are some of the advantages of having a block abstraction for a multi-disk distributed file system?

Namenodes and Datanodes

1. Understand the functions of namenodes and datanodes.

2. What are the two mechanisms that Hadoop provides for making the namenode resilient to failure?

3. Understand the purpose of secondary namenode. Does it run on the same machine as the primary namenode?

Block Caching

1. If an application wanted to cache a file, it would instruct the _____ (namenode/datanode), and the file would be cached by the _____ (one/multiple) datanodes by default.

HDFS Federation

1. If a cluster became very large, what would be the limiting factor? How does HDFS federation solve this problem. Understand the concept of namespace volume.

HDFS High Availability

1. If the namenode fails, the whole cluster goes down and time to recover and restore it to the previous running state would be large. How does HDFS high availability solve this problem in Hadoop 2.x? Understand the various architectural changes needed.

HDFS Command Line Interface

1. Besides the commands mentioned in the text, understand common file system shell commands: (No written answer required)
https://hadoop.apache.org/docs/r2.7.2/hadoop-project-dist/hadoop-common/FileSystemShell.html

The Java Interface

1. Understand the Java interface for interacting with HDFS. Go through the examples and try to understand the various classes involved. No written answer required.