Lab 2: HDFS

## Solution 1

Text

Description automatically generated

Text

Description automatically generated

## Solution 2

1. ls – List information about files and directories in the current directory in HDFS
2. mkdir – Create a new directory in HDFS
3. cat – Concatenate files from HDFS to standard output
4. put – Copy files or directories from the local file system to the destination file system
5. get – Copy files or directories from HDFS to the local file system
6. rm – Remove/Delete file from HDFS
7. rmdir – Remove/Delete directory from HDFS only if empty

## Solution 3

Commands used to:

1. Format Namenode – **hdfs namenode -format** OR **bin/hdfs namenode -format**
2. Start HDFS server – **start-dfs.sh** OR **sbin/start-dfs.sh**
3. Stop HDFS server – **stop-dfs.sh** OR **sbin/stop-dfs.sh**

\*We can directly use these commands once the **hadoop-3.3.4/sbin** & **hadoop-3.3.4/bin** is added to the $PATH environment variable; otherwise, we need to change the current working directory to hadoop-3.3.4 and execute the commands with appending the parent directory before the actual command.

## Solution 4

Files modified while setting up a single-node cluster on a pseudo-distributed mode

1. etc/hadoop/hadoop-env.sh – This script sets up the environment for Hadoop and its daemons. Here we have the environment variable setting the root of the Java installation, i.e., points the JAVA\_HOME environment variable to the file system path where Java JDK or JRE is installed.
2. etc/hadoop/core-site.xml – To define global configuration properties and settings for core services in the Hadoop cluster: HDFS namenode, datanode, YARN resource manager, and map-reduce job history server. Configure the default file system for namenode and datanode. The value for the site configuration property denotes where the namenode will be hosted on cluster & binded on which port, additionally the post where datanode will send its heartbeat.
3. etc/hadoop/hdfs-site.xml – Configure properties for HDFS components that control the behavior and performance and help to ensure the reliability and efficiency of data storage in a Hadoop cluster. Here, we set the datanode file system replicas to 1.