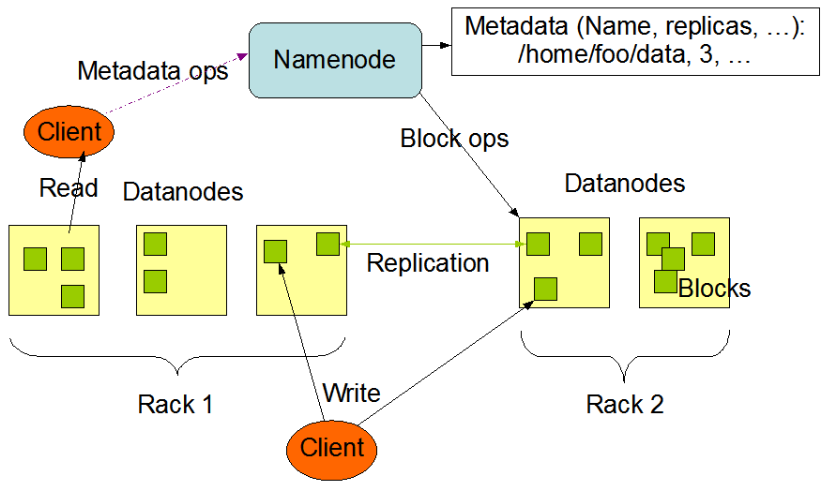
|  |  |
| --- | --- |
| Name: | Prerna Sunil Jadhav |
| Sap Id: | 60004220127 |
| Class: | T. Y. B. Tech (Computer Engineering) |
| Course: | Big Data Infrastructure Laboratory |
| Course Code: | DJ19CEEL6011 |
| Experiment No.: | 03 |

**AIM:** Execute different HDFS Commands.

**WHAT IS HDFS?**

The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS relaxes a few POSIX requirements to enable streaming access to file system data. HDFS was originally built as infrastructure for the Apache Nutch web search engine project.

**HDFS Architecture:**

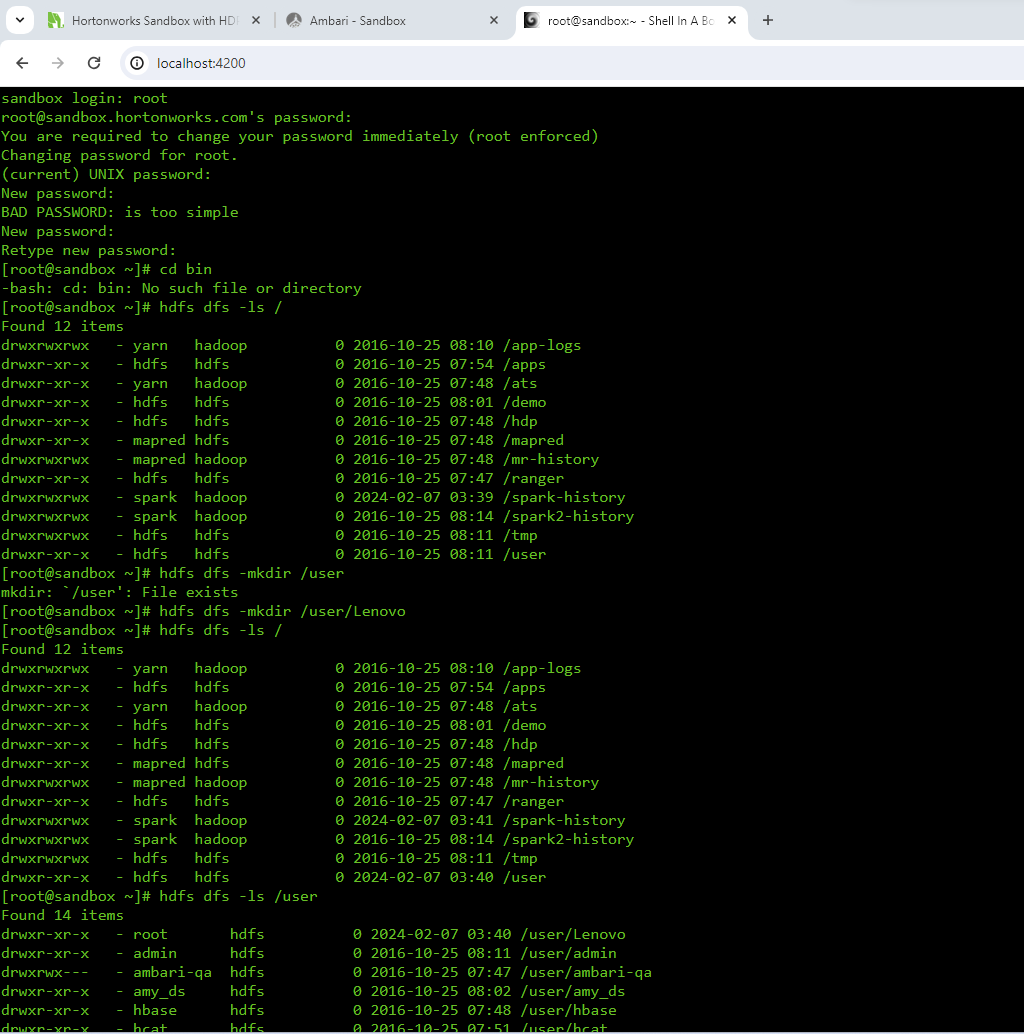
**Commands:**

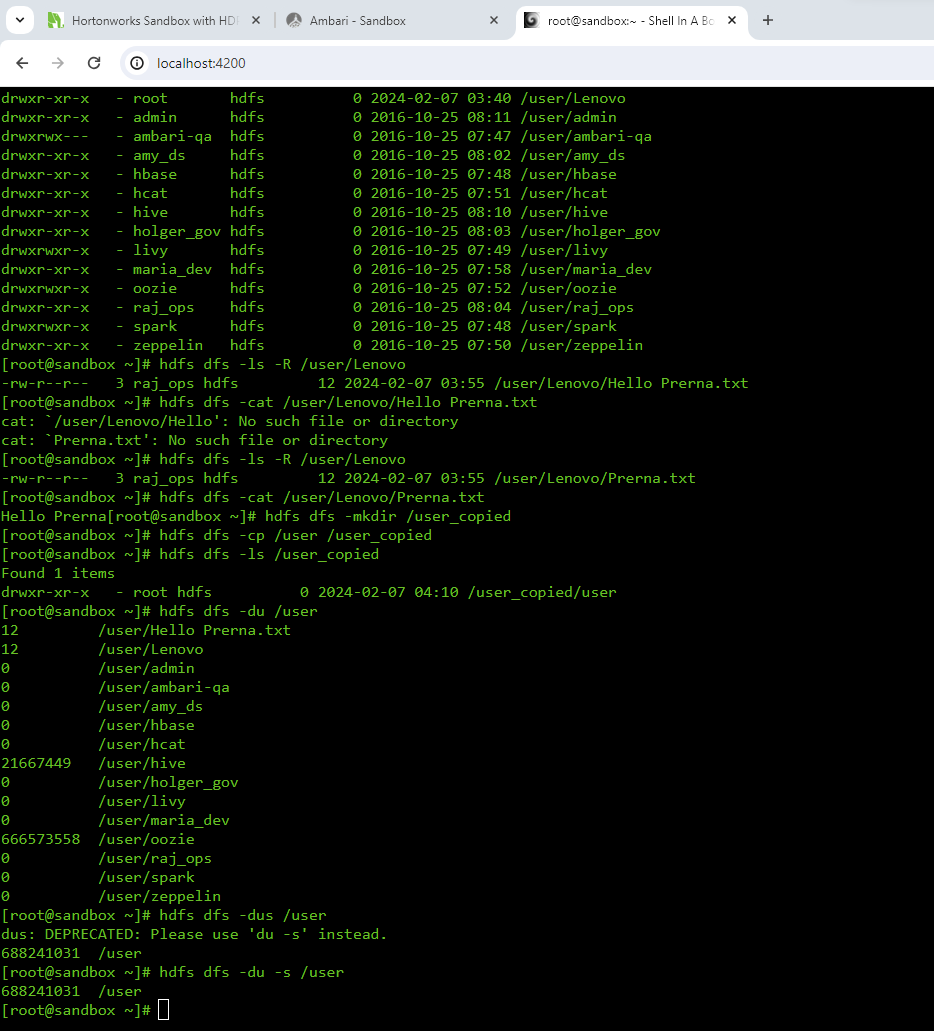
* ***hadoop version***: The Hadoop fs shell command version prints the Hadoop version.
* ***hadoop fs –mkdir /path/directory\_name:*** This command creates the directory in HDFS if it does not already exist. Use ***hadoop fs mkdir -p /path/directoryname***, so not to fail even if directory exists.

Note: If the directory already exists in HDFS, then we will get an error message that file already exists.

* ***hadoop fs -ls /path:*** The Hadoop fs shell command ls displays a list of the contents of a directory specified in the path provided by the user. It shows the name, permissions, owner, size, and modification date for each file or directories in the specified directory.
* ***hadoop fs –cat /path\_to\_file\_in\_hdfs:*** The cat command reads the file in HDFS and displays the content of the file on console or stdout.
* ***hadoop fs -mv <src> <dest>:*** The HDFS mv command moves the files or directories from the source to a destination within HDFS.
* ***hadoop fs -cp <src> <dest>:*** The cp command copies a file from one directory to another directory within the HDFS.

**Code & Output:**





**Advantages**:

* Distributed data storage, High fault tolerance, Blocks reduce seek time.
* The data is highly available as the same block is present at multiple data-nodes.
* Even if multiple data-nodes are down we can still do our work, thus making it highly reliable.

**Limitations**: Though HDFS provide many features there are some areas where it doesn’t work well.

* Low latency data access
* Small file problem