NAME:- DUBEY KARAN SANJEEV
CLASS:- B.E - 4
ROLL NO:- 04
BATCH:- A

EXPERIMENT NO-1

AIM: a: To study overview of Hadoop Ecosystem tools.
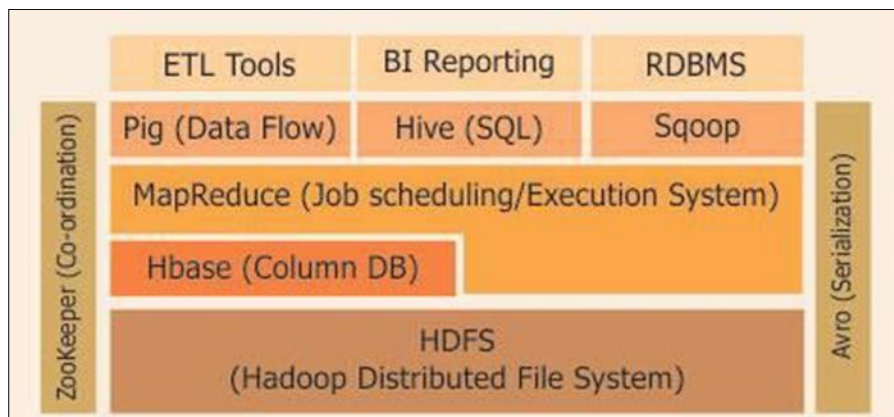
b: To execute basic commands of Hadoop eco system components like HDFS, Hive, Sqoop.

THEORY:

What is Hadoop?

Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs.

 Hadoop Ecosystem

Hadoop is distributed master slave architecture that consist of

1. Hadoop distributed file system (HDFS) for storage.
    - It is storage component of Hadoop.
    - It is distributed file system that is modeled after Google File System (GFS).
    - It is optimized for high throughput and works best while reading and writing large files.

2. Map Reduce for computational capabilities.
    - It is the heart of Hadoop.
    - It provides a mechanism for programmer to control distributed systems for processing data sets.

1. Pig:
   - A data flow language and execution environment for exploring very large datasets.
   - Pig is made up of two pieces:
       o The language used to express data flows, called Pig Latin.
       o The execution environment to run Pig Latin programs.

- There are currently two environments: local execution in a single JVM and distributed execution on a Hadoop cluster.
- A Pig Latin program is made up of a series of operations, or transformations, that are applied to the input data to produce output.
- Pig is a scripting language for exploring large datasets,Writing the mappers and reducers, compiling and packaging the code, submitting the job(s), and retrieving the results is a time consuming task.
- Pig has the ability to process terabytes of data simply by issuing half dozen times of pig latin from console.

2.Hive:

- A distributed data warehouse.
- Hive manages data stored in HDFS and provides a query language based on SQL (and which is translated by the runtime engine to MapReduce jobs) for querying the data.
- Hive was created to make it possible for analysts with strong SQL skills (but meager Java programming skills) to run queries on the huge volumes of data

3.Sqoop

- A tool for efficient bulk transfer of data between structured data stores such as RDBMS and HDFS.
- Hadoop platform has ability to work with data in several forms.
- Often, valuable data in an organization is stored in structured data stores such as RDBMS.
- Sqoop is an open-source tool that allows users to extract data from a relational database into Hadoop for further processing.

4. HBase

- A distributed, column-oriented database. HBase uses HDFS for its underlying storage, and supports both batch-style computations using MapReduce and point queries (random reads).
- HBase is not relational and does not support SQL, but given the proper problem space, it is able to do what an RDBMS cannot such as host very large, sparsely populated tables on clusters made from commodity hardware.

5.Zookeeper:

- A distributed, highly available coordination service. ZooKeeper provides primitives such as distributed locks that can be used for building distributed applications.
- Writing distributed applications is hard.
- For example

When a message is sent across the network between two nodes and the network fails,the sender does not know whether the receiver got the message. It may have gotten through before the network failed, or it may not have. Or perhaps the receiver's process died. The only way that the sender can find out what happened is to reconnect to the receiver and ask it. This is partial failure: when we don't even know if an operation failed. ZooKeeper can't make partial failures go away, since they are intrinsic to distributed systems. It certainly does not hide partial failures, either.1 But what ZooKeeper does do is give you a set of tools to build distributed applications that can safely handle partial failures.

6.Avro

- A serialization system for efficient, cross-language RPC, and persistent data storages.
- Apache Avro is a language-neutral data serialization system.
- Avro provides
  - Rich data structures.

- A compact, fast, binary data format.
- A container file, to store a persistent data.
- RPC.

CONLUSION:

Hadoop ecosystem consists of variety technologies which can be used for various applications and basic commands of HDFS are executed which are used for map reduce execution.

Program formation/ Execution/ ethical practices (06)	Timely Submission and Documentation (02)	Viva Answer (02)	Experimen t Marks (10)	Teacher Signature with date

1) help HDFS Shell Command

Syntax of help hdfs Command

```
$ hadoop fs -help
```
Help hdfs shell command helps hadoop developers figure out all the available hadoop commands and how to use them.

```
cloudera@localhost:~                                          _ □ ✗

File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -help
Usage: hadoop fs [generic options]
        [-cat [-ignoreCrc] <src> ...]
        [-chgrp [-R] GROUP PATH...]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-copyFromLocal <localsrc> ... <dst>]
        [-copyToLocal [-ignoreCrc] [-crc] <src> ... <localdst>]
        [-count [-q] <path> ...]
        [-cp <src> ... <dst>]
        [-df [-h] [<path> ...]]
        [-du [-s] [-h] <path> ...]
        [-expunge]
        [-get [-ignoreCrc] [-crc] <src> ... <localdst>]
        [-getmerge [-nl] <src> <localdst>]
        [-help [cmd ...]]
        [-ls [-d] [-h] [-R] [<path> ...]]
        [-mkdir [-p] <path> ...]
        [-moveFromLocal <localsrc> ... <dst>]
        [-moveToLocal <src> <localdst>]
        [-mv <src> ... <dst>]
        [-put <localsrc> ... <dst>]
        [-rm [-f] [-r|-R] [-skipTrash] <src> ...]
        [-rmdir [--ignore-fail-on-non-empty] <dir> ...]
        [-setrep [-R] [-w] <rep> <path/file> ...]
        [-stat [format] <path> ...]
        [-tail [-f] <file>]
        [-test -[ezd] <path>]
        [-text [-ignoreCrc] <src> ...]
        [-touchz <path> ...]
        [-usage [cmd ...]]

-cat [-ignoreCrc] <src> ...:    Fetch all files that match the file pattern <src
>
                and display their content on stdout.

-chgrp [-R] GROUP PATH...:      This is equivalent to -chown ... :GROUP ...
```

Variations of the Hadoop fs Help Command

$ hadoop fs –help ls

Using the help command with a specific command lists the usage information along with the options to use the command.

```
[cloudera@localhost ~]$ hadoop fs -help ls
-ls [-d] [-h] [-R] [<path> ...]:         List the contents that match the specifi
ed file pattern. If
                path is not specified, the contents of /user/<currentUser>
                will be listed. Directory entries are of the form
                        dirName (full path) <dir>
                and file entries are of the form
                        fileName(full path) <r n> size
                where n is the number of replicas specified for the file
                and size is the size of the file, in bytes.
                -d  Directories are listed as plain files.
                -h  Formats the sizes of files in a human-readable fashion
                    rather than a number of bytes.
                -R  Recursively list the contents of directories.
[cloudera@localhost ~]$
```

2) Usage HDFS Shell Command

`$ hadoop fs –usage ls`
Usage command gives all the options that can be used with a particular hdfs command.


```
[cloudera@localhost ~]$ hadoop fs -usage ls
Usage: hadoop fs [generic options] -ls [-d] [-h] [-R] [<path> ...]
[cloudera@localhost ~]$
```

3) ls HDFS Shell Command

Syntax for ls Hadoop Command –

`$ hadoop fs –ls`
This command will list all the available files and subdirectories under default directory.For instance, in our example the default directory for Cloudera VM is /user/cloudera


```
[cloudera@localhost ~]$ hadoop fs -ls
Found 4 items
drwx------   - cloudera cloudera          0 2016-09-06 19:00 .staging
drwxr-xr-x   - cloudera cloudera          0 2016-09-06 18:50 Input
drwxr-xr-x   - cloudera cloudera          0 2016-09-06 19:00 Output
drwxr-xr-x   - cloudera cloudera          0 2016-08-25 19:53 tweets
[cloudera@localhost ~]$
```

Variations of Hadoop ls Shell Command

$ hadoop fs –ls /

Returns all the available files and subdirectories present under the root directory.

$ hadoop fs –ls –R /user/cloudera

Returns all the available files and recursively lists all the subdirectories under /user/Cloudera

4) mkdir– Used to create a new directory in HDFS at a given location.

Example of HDFS mkdir Command –

$ hadoop fs –mkdir /user/cloudera/dezyre1

The above command will create a new directory named dezyre1 under the location /user/cloudera

```
cloudera@localhost:~                                          _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -mkdir /user/cloudera/dezyre1
[cloudera@localhost ~]$
```

Note : Cloudera and other hadoop distribution vendors provide /user/ directory with read/write permission to all users but other directories are available as read-only.Thus, to create a folder in the root directory, users require superuser permission as shown below –

$ sudo –u hdfs hadoop fs –mkdir /dezyre

This command will create a new directory named dezyre under the / (root directory).

```
cloudera@localhost:~                                          _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ sudo -u hdfs hadoop fs -mkdir /dezyre
[cloudera@localhost ~]$
```
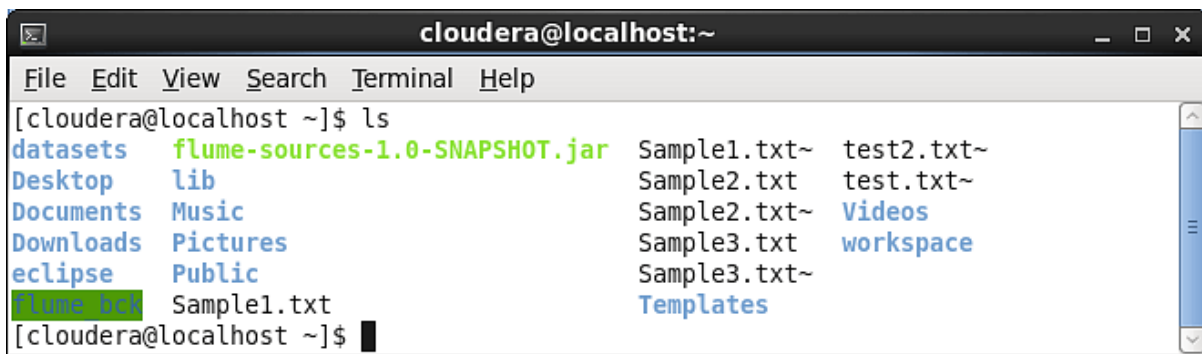
5) copyFromLocal

Copy a file from local filesytem to HDFS location.

For the following examples, we will use Sample.txt file available in the /home/Cloudera location.

```
cloudera@localhost:~                                          _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ ls
datasets     flume-sources-1.0-SNAPSHOT.jar  Sample1.txt~  test2.txt~
Desktop      lib                             Sample2.txt   test.txt~
Documents    Music                           Sample2.txt~  Videos
Downloads    Pictures                        Sample3.txt   workspace
eclipse      Public                          Sample3.txt~
flume_bck    Sample1.txt                     Templates
[cloudera@localhost ~]$
```

Example – $ hadoop fs –copyFromLocal Sample1.txt /user/cloudera/dezyre1

Copy/Upload Sample1.txt available in /home/cloudera (local default) to /user/cloudera/dezyre1 (hdfs path)

```
cloudera@localhost:~                                          _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -copyFromLocal Sample1.txt /user/cloudera/dezy
re1
[cloudera@localhost ~]$
```

6) put –

This hadoop command uploads a single file or multiple source files from local file system to hadoop distributed file system (HDFS).

Ex – $ hadoop fs –put Sample2.txt /user/cloudera/dezyre1

Copy/Upload Sample2.txt available in /home/cloudera (local default) to /user/cloudera/dezyre1 (hdfs path)

7) moveFromLocal

This hadoop command functions similar to the put command but the source file will be deleted after copying.

Example – $ hadoop fs –moveFromLocal Sample3.txt /user/cloudera/dezyre1

Move Sample3.txt available in /home/cloudera (local default) to /user/cloudera/dezyre1 (hdfs path). Source file will be deleted after moving.

8) du

Displays the disk usage for all the files available under a given directory.

Example – $ hadoop fs –du /user/cloudera/dezyre1

```
[cloudera@localhost ~]$ hadoop fs -du /user/cloudera/dezyre1
370  /user/cloudera/dezyre1/Sample1.txt
370  /user/cloudera/dezyre1/Sample2.txt
370  /user/cloudera/dezyre1/Sample3.txt
[cloudera@localhost ~]$
```

9) df

 Display disk usage of current hadoop distributed file system.

Example – $ hadoop fs –df

```
[cloudera@localhost ~]$ hadoop fs -df
Filesystem                                  Size      Used    Available  Use%
hdfs://localhost.localdomain:8020  28799380685  91496448  24875388928   0%
[cloudera@localhost ~]$
```

10) Expunge

This HDFS command empties the trash by deleting all the files and directories.

Example – $ hadoop fs –expunge

```
[cloudera@localhost ~]$ hadoop fs -expunge
[cloudera@localhost ~]$
```

11) Cat

This is similar to the cat command in Unix and displays the contents of a file.

Example – $ hadoop fs –cat /user/cloudera/dezyre1/Sample1.txt

```
┌─────────────────────────────────────────────────────────────────┐
│ ▣              cloudera@localhost:~                  _ □ ✕        │
├─────────────────────────────────────────────────────────────────┤
│ File  Edit  View  Search  Terminal  Help                         │
│ [cloudera@localhost ~]$ hadoop fs -cat /user/cloudera/dezyre1/Sample1.txt │
│         --------------------------                                │
│         / This is Sample-1 file. /                               │
│         --------------------------                                │
│                                                                   │
│                                                                   │
│ ================================================================  │
│ || Created to explain the functionality of Hadoop FsShell ||    │
│ ================================================================  │
│                                                                   │
│                                                                   │
│         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                          │
│         |   Thanks for choosing dezyre.|                         │
│         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                          │
│                                                                   │
│                                                                   │
│ [cloudera@localhost ~]$ █                                        │
└─────────────────────────────────────────────────────────────────┘
```

12) cp

Copy files from one HDFS location to another HDFS location.

Example – $ hadoop fs –cp /user/cloudera/dezyre/war_and_peace /user/cloudera/dezyre1/

```
┌─────────────────────────────────────────────────────────────────┐
│ ▣              cloudera@localhost:~                  _ □ ✕        │
├─────────────────────────────────────────────────────────────────┤
│ File  Edit  View  Search  Terminal  Help                         │
│ [cloudera@localhost ~]$ hadoop fs -cp /user/cloudera/dezyre/war_and_peace /user/│
│ cloudera/dezyre1/                                                 │
│ [cloudera@localhost ~]$ █                                        │
└─────────────────────────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────────────────────────┐
│ ▣              cloudera@localhost:~                  _ □ ✕        │
├─────────────────────────────────────────────────────────────────┤
│ File  Edit  View  Search  Terminal  Help                         │
│ [cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre       │
│ Found 1 items                                                     │
│ -rw-r--r--   3 cloudera cloudera    3291648 2016-09-29 14:19 /user/cloudera/dezy│
│ re/war_and_peace                                                  │
│ [cloudera@localhost ~]$ █                                        │
└─────────────────────────────────────────────────────────────────┘
```

13) mv

Move files from one HDFS location to another HDFS location.

Example – $ hadoop fs –mv /user/cloudera/dezyre1/Sample1.txt /user/cloudera/dezyre/

```
cloudera@localhost:~                                    _ □ ×

File  Edit  View  Search  Terminal  Help

[cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre1
Found 4 items
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:14 /user/cloudera/dezy
re1/Sample1.txt
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:16 /user/cloudera/dezy
re1/Sample2.txt
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:17 /user/cloudera/dezy
re1/Sample3.txt
-rw-r--r--   3 cloudera cloudera    3291648 2016-09-29 14:22 /user/cloudera/dezy
re1/war_and_peace
[cloudera@localhost ~]$
```

```
cloudera@localhost:~                                    _ □ ×

File  Edit  View  Search  Terminal  Help

[cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre
Found 1 items
-rw-r--r--   3 cloudera cloudera    3291648 2016-09-29 14:19 /user/cloudera/dezy
re/war_and_peace
[cloudera@localhost ~]$
```

```
cloudera@localhost:~                                    _ □ ×

File  Edit  View  Search  Terminal  Help

[cloudera@localhost ~]$ hadoop fs -mv /user/cloudera/dezyre1/Sample1.txt /user/c
loudera/dezyre/
[cloudera@localhost ~]$
```

```
cloudera@localhost:~                                    _ □ ×

File  Edit  View  Search  Terminal  Help

[cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre
Found 2 items
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:14 /user/cloudera/dezy
re/Sample1.txt
-rw-r--r--   3 cloudera cloudera    3291648 2016-09-29 14:19 /user/cloudera/dezy
re/war_and_peace
[cloudera@localhost ~]$
```

```
cloudera@localhost:~                                    _ □ ×

File  Edit  View  Search  Terminal  Help

[cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre1
Found 3 items
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:16 /user/cloudera/dezy
re1/Sample2.txt
-rw-r--r--   3 cloudera cloudera        370 2016-09-29 13:17 /user/cloudera/dezy
re1/Sample3.txt
-rw-r--r--   3 cloudera cloudera    3291648 2016-09-29 14:22 /user/cloudera/dezy
re1/war_and_peace
[cloudera@localhost ~]$
```

14) rm

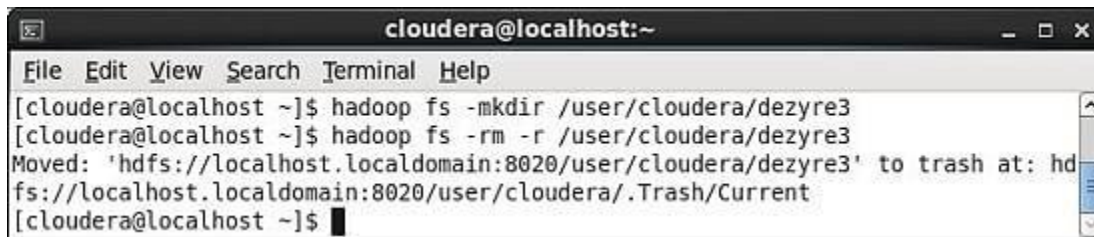Removes the file or directory from the mentioned HDFS location.

Example – $ hadoop fs –rm –r /user/cloudera/dezyre3

rm –r

Example – $ hadoop fs –rm –r /user/cloudera/dezyre3

Deletes or removes the directory and its content from HDFS location in a recursive manner.

Example – $ hadoop fs –rm /user/cloudera/dezyre3

Delete or remove the files from HDFS location.

15) tail

This hadoop command will show the last kilobyte of the file to stdout.

Example – $ hadoop fs –tail /user/cloudera/dezyre/war_and_peace

```
┌─ ⊠ ──────────────── cloudera@localhost:~ ────────── _ □ ✕ ─┐
│ File  Edit  View  Search  Terminal  Help                    │
│ [cloudera@localhost ~]$ hadoop fs -tail /user/cloudera/dezyre/war_and_peace │^│
│ p://pglaf.org/donate                                       │ │
│                                                             │ │
│                                                             │ │
│ Section 5.  General Information About Project Gutenberg-tm electronic │ │
│ works.                                                      │ │
│                                                             │ │
│ Professor Michael S. Hart is the originator of the Project Gutenberg-tm │ │
│ concept of a library of electronic works that could be freely shared │ │
│ with anyone.  For thirty years, he produced and distributed Project │ │
│ Gutenberg-tm eBooks with only a loose network of volunteer support. │ │
│                                                             │ │
│                                                             │ │
│ Project Gutenberg-tm eBooks are often created from several printed │ │
│ editions, all of which are confirmed as Public Domain in the U.S. unless │ │
│ a copyright notice is included.  Thus, we do not necessarily keep eBooks │ │
│ in compliance with any particular paper edition.           │ │
│                                                             │ │
│ Most people start at our Web site which has the main PG search facility: │ │
│                                                             │ │
│ http://www.gutenberg.org                                    │ │
│                                                             │ │
│ This Web site includes information about Project Gutenberg-tm, including │ │
│ how to make donations to the Project Gutenberg Literary Archive │ │
│ Foundation, how to help produce our new eBooks, and how to subscribe to │ │
│ our email newsletter to hear about new eBooks.             │≡│
│ [cloudera@localhost ~]$ █                                   │ │
└─────────────────────────────────────────────────────────────┘
```

Example – $ hadoop fs –tail –f /user/cloudera/dezyre/war_and_peace

Using the tail commands with –f option, shows the last kilobyte of the file from end in a page wise format.
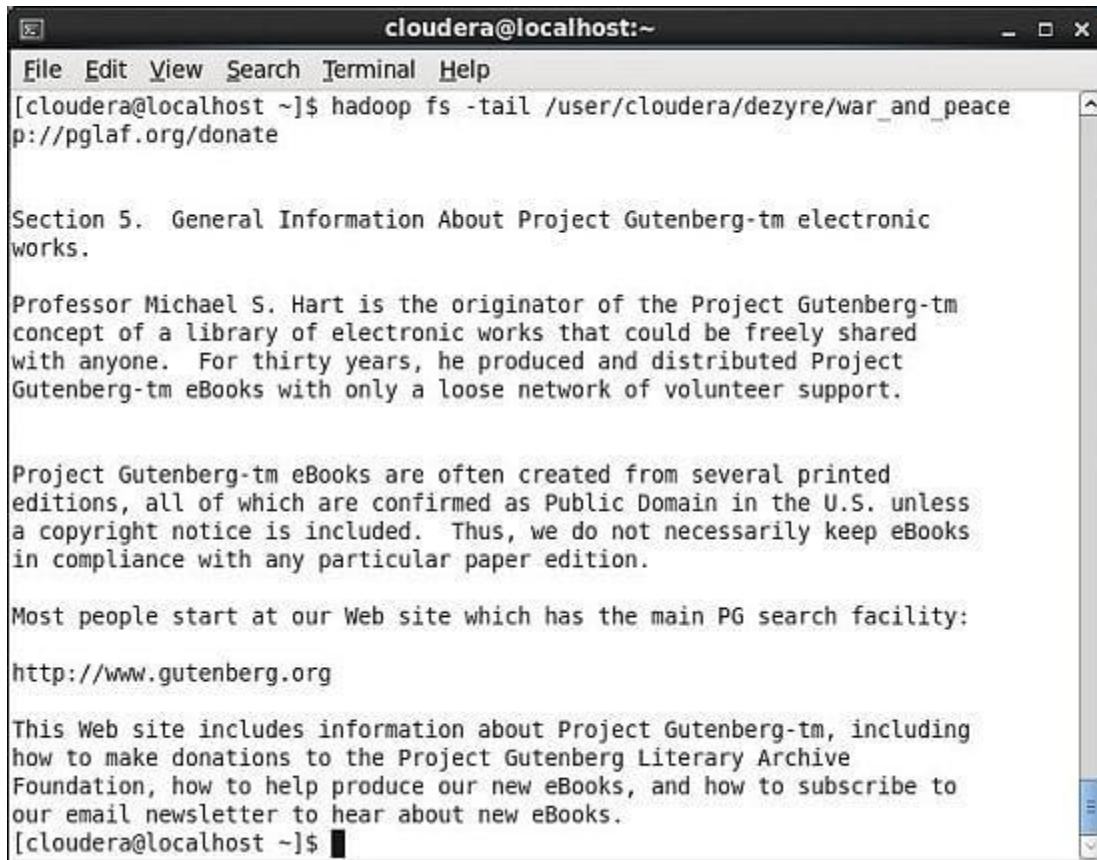
```
                    cloudera@localhost:~                    _ □ ✕
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -tail -f /user/cloudera/dezyre/war_and_peace ▲
p://pglaf.org/donate


Section 5.  General Information About Project Gutenberg-tm electronic
works.

Professor Michael S. Hart is the originator of the Project Gutenberg-tm
concept of a library of electronic works that could be freely shared
with anyone.  For thirty years, he produced and distributed Project
Gutenberg-tm eBooks with only a loose network of volunteer support.


Project Gutenberg-tm eBooks are often created from several printed
editions, all of which are confirmed as Public Domain in the U.S. unless
a copyright notice is included.  Thus, we do not necessarily keep eBooks
in compliance with any particular paper edition.

Most people start at our Web site which has the main PG search facility:

http://www.gutenberg.org

This Web site includes information about Project Gutenberg-tm, including
how to make donations to the Project Gutenberg Literary Archive
Foundation, how to help produce our new eBooks, and how to subscribe to
our email newsletter to hear about new eBooks.
█
```

16) copyToLocal

Copies the files to the local filesystem . This is similar to hadoop fs –get command but in this case the destination location msut be a local file reference

Example – $ hadoop fs –copyFromLocal /user/cloudera/dezyre1/Sample1.txt /home/cloudera/hdfs_bkp/

Copy/Download Sample1.txt available in /user/cloudera/dezyre1 (hdfs path) to /home/cloudera/hdfs_bkp/ (local path)


```
                    cloudera@localhost:~                    _ □ ✕
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -copyToLocal /user/cloudera/dezyre1/Sample1.tx ▲
t /home/cloudera/hdfs_bkp/
[cloudera@localhost ~]$ ls -l /home/cloudera/hdfs_bkp/
total 4
-rwxr-xr-x 1 cloudera cloudera 370 Oct 13 18:59 Sample1.txt
[cloudera@localhost ~]$ █
```

17) get

Downloads or Copies the files to the local filesystem.

Example – $ hadoop fs –get /user/cloudera/dezyre1/Sample2.txt /home/cloudera/hdfs_bkp/

Copy/Download Sample2.txt available in /user/cloudera/dezyre1 (hdfs path) to /home/cloudera/hdfs_bkp/ (local path)
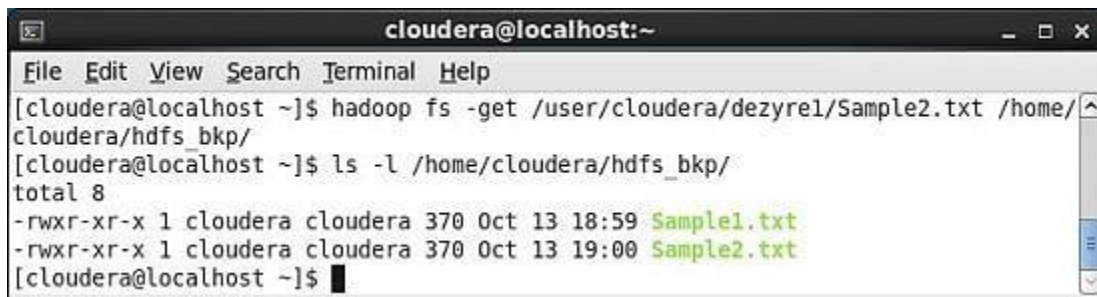
```
                         cloudera@localhost:~                    _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -get /user/cloudera/dezyre1/Sample2.txt /home/
cloudera/hdfs_bkp/
[cloudera@localhost ~]$ ls -l /home/cloudera/hdfs_bkp/
total 8
-rwxr-xr-x 1 cloudera cloudera 370 Oct 13 18:59 Sample1.txt
-rwxr-xr-x 1 cloudera cloudera 370 Oct 13 19:00 Sample2.txt
[cloudera@localhost ~]$
```

18) touchz

Used to create an emplty file at the specified location.

Example – $ hadoop fs –touchz /user/cloudera/dezyre1/Sample4.txt

It will create a new empty file Sample4.txt in /user/cloudera/dezyre1/ (hdfs path)

```
                         cloudera@localhost:~                    _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -touchz /user/cloudera/dezyre1/Sample4.txt
[cloudera@localhost ~]$ hadoop fs -ls /user/cloudera/dezyre1/
Found 4 items
-rw-r--r--   3 cloudera cloudera      370 2016-09-29 17:48 /user/cloudera/dezy
re1/Sample1.txt
-rw-r--r--   3 cloudera cloudera      370 2016-09-29 13:16 /user/cloudera/dezy
re1/Sample2.txt
-rw-r--r--   3 cloudera cloudera      370 2016-09-29 13:17 /user/cloudera/dezy
re1/Sample3.txt
-rw-r--r--   3 cloudera cloudera        0 2016-10-13 19:10 /user/cloudera/dezy
re1/Sample4.txt
[cloudera@localhost ~]$
```

19) setrep

This hadoop fs command is used to set the replication for a specific file.

Example – $ hadoop fs –setrep –w 1 /user/cloudera/dezyre1/Sample1.txt

It will set the replication factor of Sample1.txt to 1
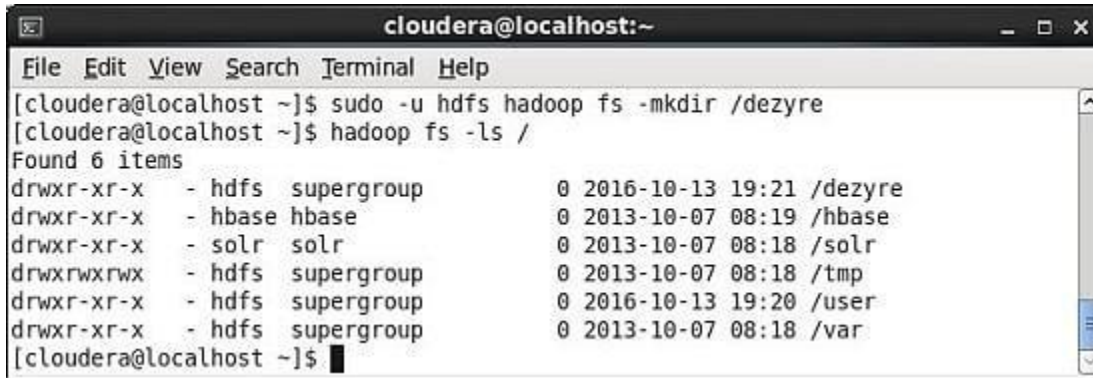
```
                         cloudera@localhost:~                    _ □ ×
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -setrep -w 1 /user/cloudera/dezyre1/Sample1.tx
t
Replication 1 set: /user/cloudera/dezyre1/Sample1.txt
Waiting for /user/cloudera/dezyre1/Sample1.txt ... done
[cloudera@localhost ~]$
```

20) chgrp

This hadoop command is basically used to change the group name.

Example – $ sudo –u hdfs hadoop fs –chgrp –R cloudera /dezyre

It will change the /dezyre directory group membership from supergroup to cloudera (To perform this operation superuser permission is required)


```
                          cloudera@localhost:~                        _ □ ✕
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ sudo -u hdfs hadoop fs -mkdir /dezyre
[cloudera@localhost ~]$ hadoop fs -ls /
Found 6 items
drwxr-xr-x   - hdfs   supergroup          0 2016-10-13 19:21 /dezyre
drwxr-xr-x   - hbase  hbase               0 2013-10-07 08:19 /hbase
drwxr-xr-x   - solr   solr                0 2013-10-07 08:18 /solr
drwxrwxrwx   - hdfs   supergroup          0 2013-10-07 08:18 /tmp
drwxr-xr-x   - hdfs   supergroup          0 2016-10-13 19:20 /user
drwxr-xr-x   - hdfs   supergroup          0 2013-10-07 08:18 /var
[cloudera@localhost ~]$
```



```
                          cloudera@localhost:~                        _ □ ✕
File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ sudo -u hdfs hadoop fs -chgrp -R cloudera /dezyre
[cloudera@localhost ~]$ hadoop fs -ls /dezyre
[cloudera@localhost ~]$ hadoop fs -ls /
Found 6 items
drwxr-xr-x   - hdfs   cloudera            0 2016-10-13 19:21 /dezyre
drwxr-xr-x   - hbase  hbase               0 2013-10-07 08:19 /hbase
drwxr-xr-x   - solr   solr                0 2013-10-07 08:18 /solr
drwxrwxrwx   - hdfs   supergroup          0 2013-10-07 08:18 /tmp
drwxr-xr-x   - hdfs   supergroup          0 2016-10-13 19:20 /user
drwxr-xr-x   - hdfs   supergroup          0 2013-10-07 08:18 /var
[cloudera@localhost ~]$
```

21) chown

This command lets you change both the owner and group name simulataneously.

Example – $ sudo –u hdfs hadoop fs –chown –R cloudera /dezyre

It will change the /dezyre directory ownership from hdfs user to cloudera user (To perform this operation superuser is permission required)

```
 ▣                    cloudera@localhost:~                    _ □ ✕

File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ sudo -u hdfs hadoop fs -chown -R cloudera /dezyre
[cloudera@localhost ~]$ hadoop fs -ls /
Found 6 items
drwxr-xr-x    -  cloudera cloudera       0 2016-10-13 19:21 /dezyre
drwxr-xr-x    -  hbase    hbase          0 2013-10-07 08:19 /hbase
drwxr-xr-x    -  solr     solr           0 2013-10-07 08:18 /solr
drwxrwxrwx    -  hdfs     supergroup     0 2013-10-07 08:18 /tmp
drwxr-xr-x    -  hdfs     supergroup     0 2016-10-13 19:20 /user
drwxr-xr-x    -  hdfs     supergroup     0 2013-10-07 08:18 /var
[cloudera@localhost ~]$ █
```

22) hadoop chmod

Used to change the permissions of a given file/dir.

Example – $ hadoop fs –chmod /dezyre

It will change the /dezyre directory permission to 700 (drwx------).

```
 ▣                    cloudera@localhost:~                    _ □ ✕

File  Edit  View  Search  Terminal  Help
[cloudera@localhost ~]$ hadoop fs -chmod 700 /dezyre
[cloudera@localhost ~]$ hadoop fs -ls /
Found 6 items
drwx------    -  cloudera cloudera       0 2016-10-13 19:21 /dezyre
drwxr-xr-x    -  hbase    hbase          0 2013-10-07 08:19 /hbase
drwxr-xr-x    -  solr     solr           0 2013-10-07 08:18 /solr
drwxrwxrwx    -  hdfs     supergroup     0 2013-10-07 08:18 /tmp
drwxr-xr-x    -  hdfs     supergroup     0 2016-10-13 19:20 /user
drwxr-xr-x    -  hdfs     supergroup     0 2013-10-07 08:18 /var
[cloudera@localhost ~]$ █
```

Note : hadoop chmod 777

To execute this , the user must be the owner of the file or must be a super user. On executing this command, all users will get read,write and execute permission on the file.