**Assignment 1**  Kamaldeep Kaur

Data Technology Solutions(1024) c0908139

**Answer 1:**

**Creating a Directory and a File in Local System:**

1. $ -ls /
2. $ mkdir learning
3. $ cd learning
4. $ vi BeginnersTutorial.txt
5. To begin writing use: Hit i Button
6. To get out of the file Hit Escape : wq!
7. Press Enter
8. To display file contents use: $ cat Beginners.txt
9. After executing this command I received an error message

“Cannot create BeginnersTutorial.txt.Copying. Name node is in safe mode”.

1. To overcome this error remove the safemode using the following commands
2. To leave safe mode use: $ hdfs dfsadmin -safemode leave
3. After this command Safe mode is OFF
4. To verify use: $ hdfs dfsadmin -safemode get

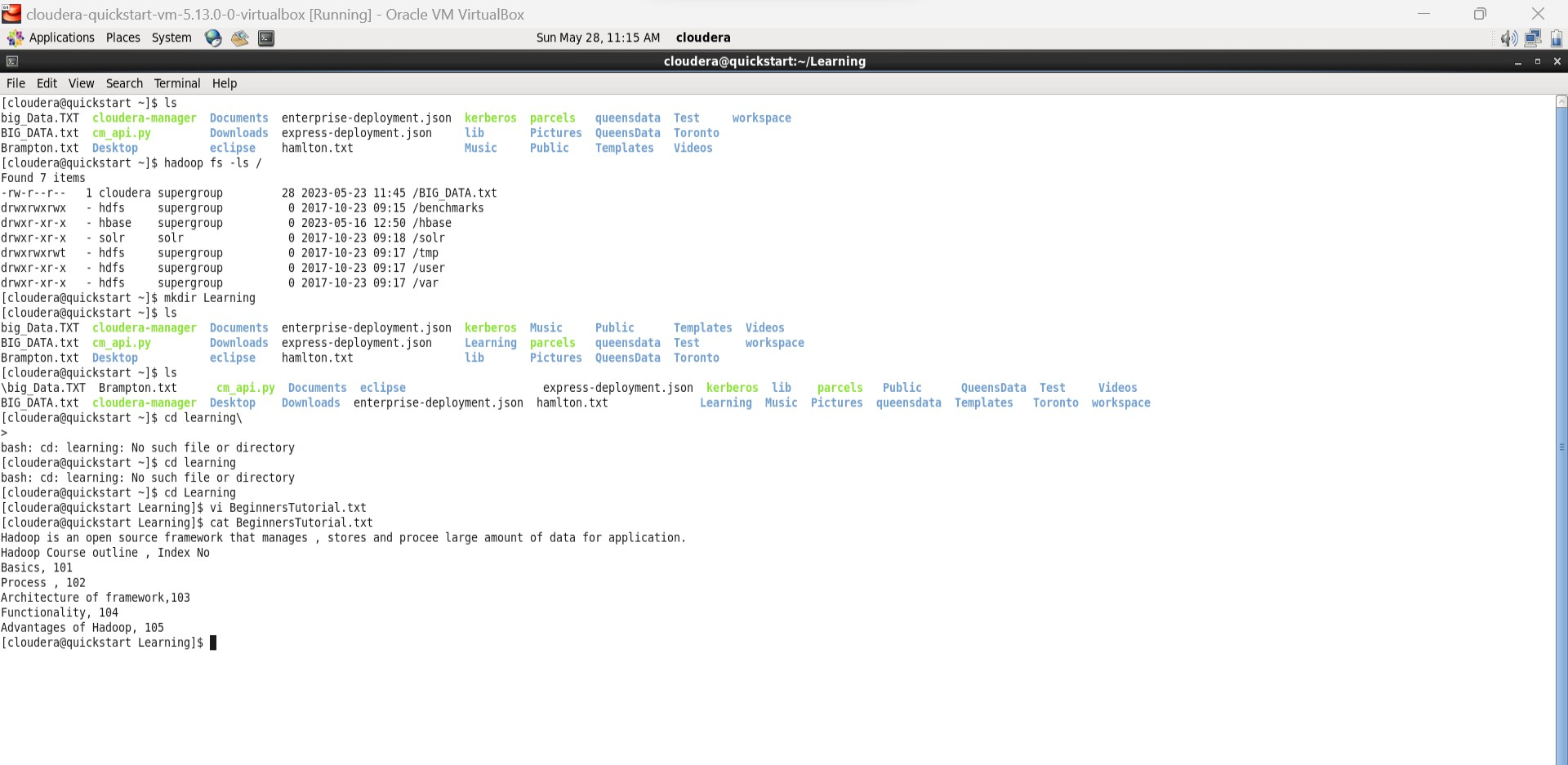


Fig 1

**Task1:**

**Moving a File from Local to the Hadoop:**

1. To transfer text file from local to Hadoop use: $ hadoop fs -put BeginnersTutorial.txt
2. To verify whether the file is transferred to Hadoop use: $ hadoop fs -ls /
3. To open the contents of file in hadoop use: $ hadoop fs -cat / BeginnerTutorial.txt
4. To check the present directory use: pwd

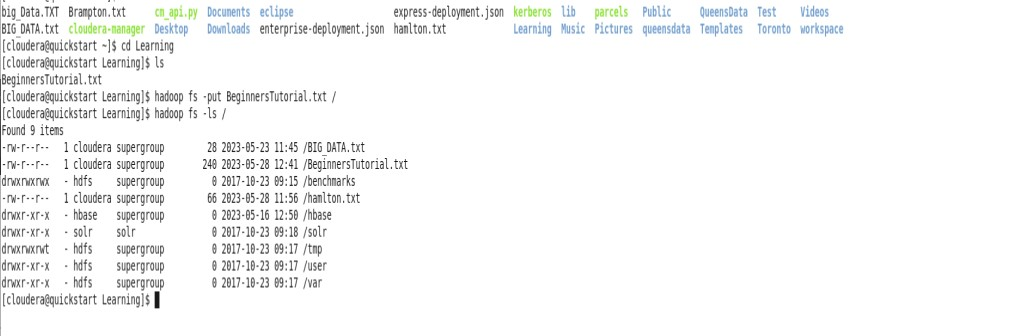


Fig 1.1

Fig 1.1 shows the commands use to move text file from local to Hadoop

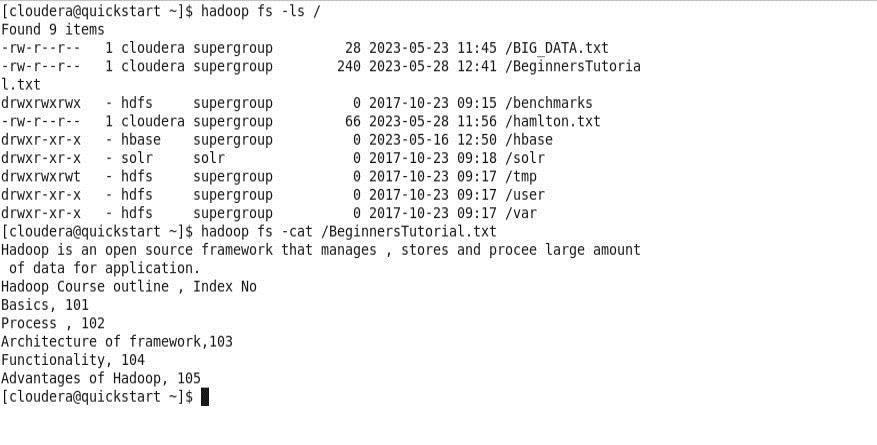


Fig 1.2

Fig 1.2 verifies whether the file is successfully moved or not and checking the format of the file using

“$ hadoop fs -ls /” command.

We can open the contents of file using command:

$ hadoop fs -cat /BeginnersTutorial.txt

**Task 2:**

**Creating & Importing the CSV file into hadoop:**

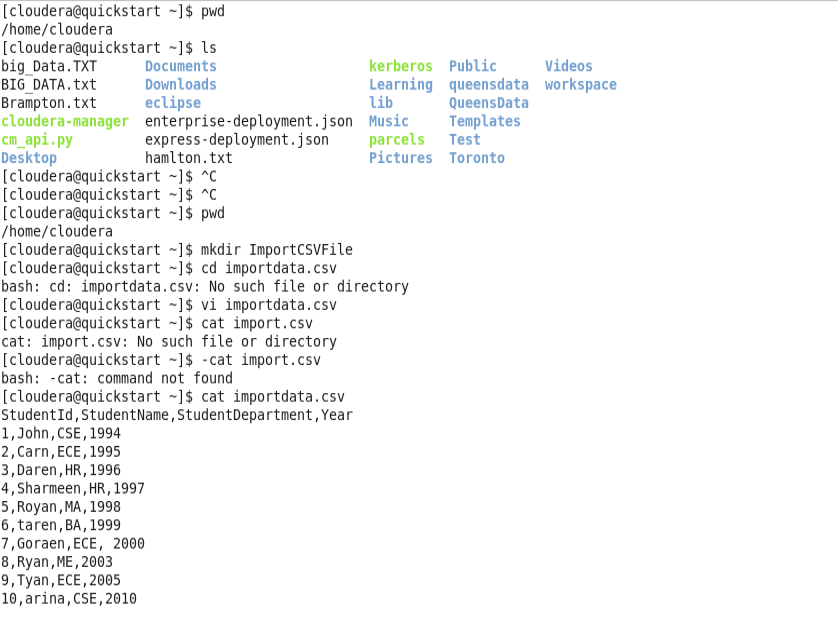
****

Fig 1.3

Fig 1.3 shows the steps to create the csv file in local.

**Importing the CSV file from local to Hadoop**

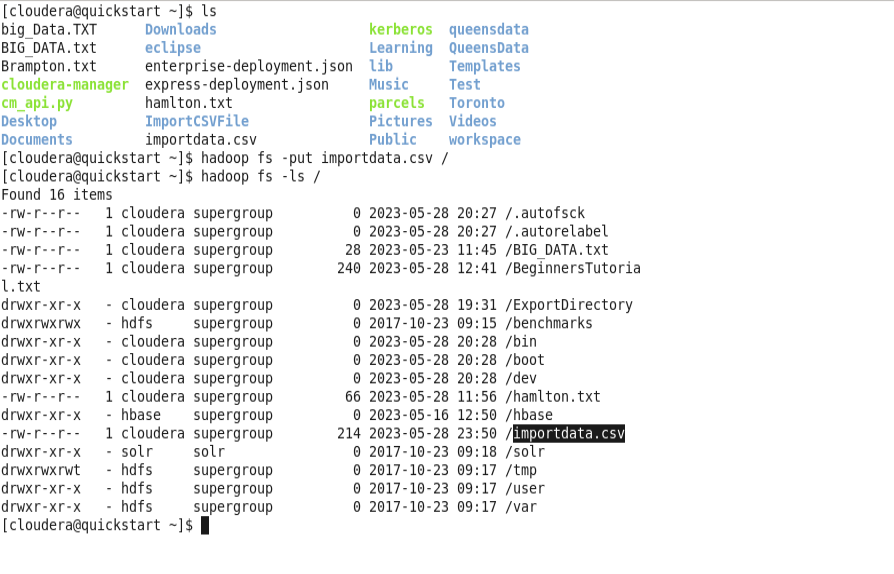
****

Fig 1.4

Fig1.4 shows the steps to move the local file to hadoop using “$ hadoop fs -put filename.csv /” and verifying the successful file transfer is done by using “$ hadoop fs -ls /”

**Answer 2:**

**Exporting a text file from hadoop to local:**

1. To get file from hadoop to local system use: $ hadoop fs -get /BeginnersTutorial.txt /home/cloudera
2. To verify use: $ ls
3. The BeginnersTutorial.txt file is now available on local.

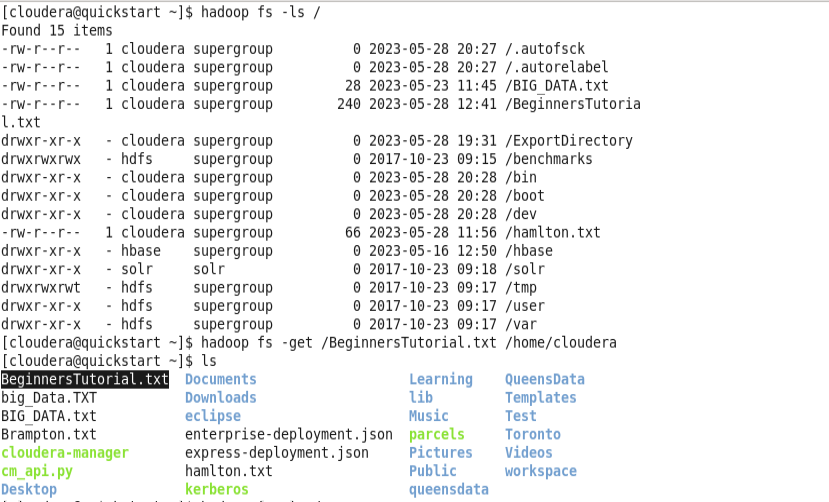
****

Fig 2

Fig 2 shows the file transferred from the Hadoop system to local using get command.

**Task 2:**

**Exporting CVS file from Hadoop to local:**

1. Use Command $ hadoop fs -ls / to list the files and directory names.
2. Use command $ hadoop fs -get /import.csv file to transfer.
3. Verify the file using $ ls command.

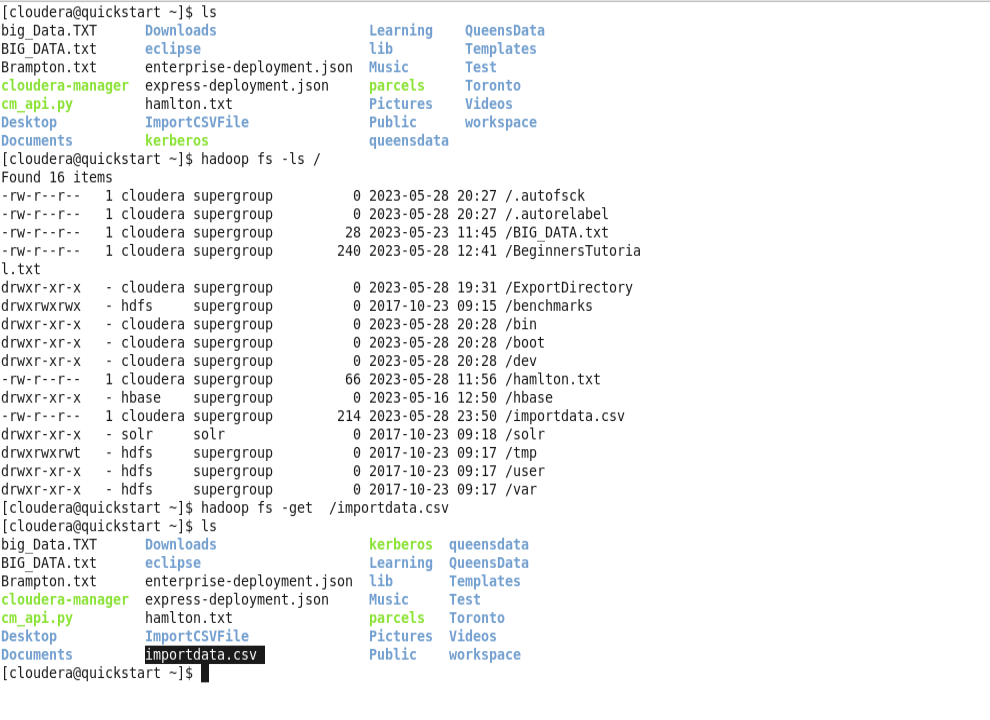
****

Fig 2.1

Fig 2.1 Shows the csv file transferred from hadoop to local.

**Parque File :**

1. It isalight very lightweight for saving data as compared to other files.
2. Parque file is 2 times faster than csv file as Csv needs to be zipped.
3. It uses data compression scheme for fast transfer.
4. It is columnar storage format.

**Answer 3:**

**Importing a Directory from Local to Hadoop**

1. Create a directory using $ mkdir MultipleFiles
2. Create multiple files in that directory using $ vi FirstFile.txt
3. Use $ ls command to list all the files in directory.

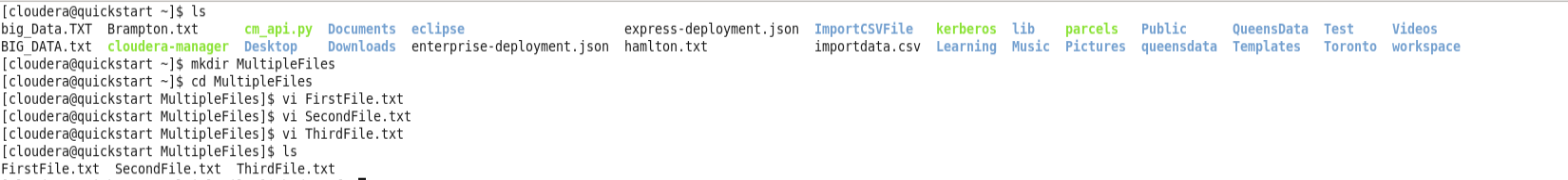
****

Fig 3

Fig 3 shows a directory with multiple files in it.

****

Fig 3.1

Fig 3.1 shows the steps to move the whole directory to Hadoop

1. Use command cd .. to come out of the current directory.
2. Use $ -put MultipleFiles / to move the directory.
3. Use $ hadoop fs -ls / command to check all the file and directories present at home.
4. To go inside the Multiple Files directory use $ hadoop fs -ls /MultipleFiles
5. Command 4 display all the files present in the directory.

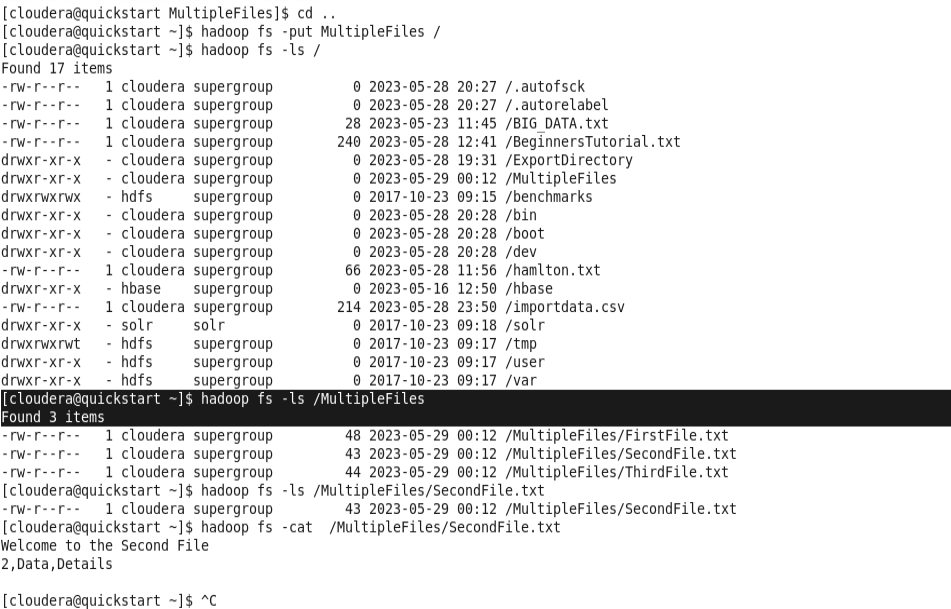
****

Fig 3.2

**Importing the directory from Hadoop to Local:**

1. Use $ hadoop fs -ls / to check all the directories and file present at hadoop.
2. Use command $ hadoop fs -get /MultipleFiles /home/cloudera/Desktop/Local.

$ hadoop fs -get Currentpath Destinationpath.

1. Use $ cd /home/clouder/Desktop/Local to check the directory on local

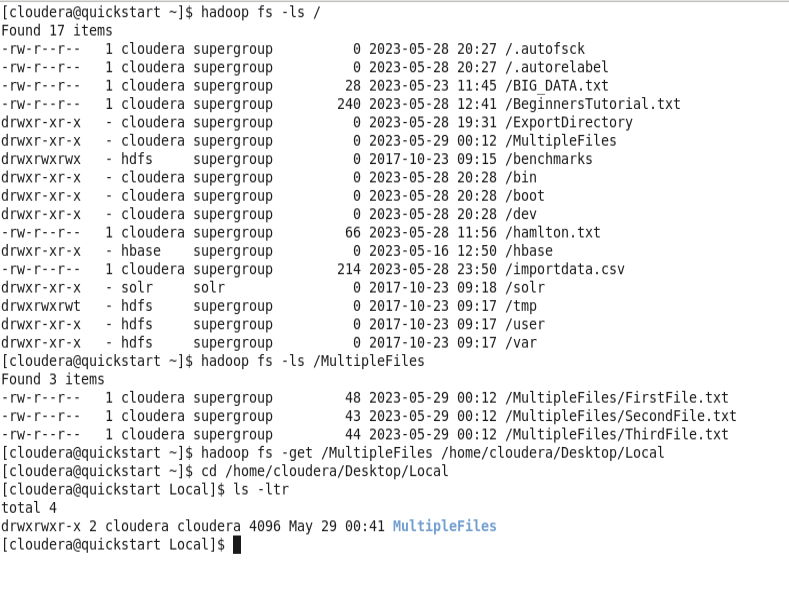
****

Fig 3.3

Fig 3.3 shows the output after transferring directory from hadoop to local.

**Answer 4:**

**Hadoop:**

1. Hadoop is a distributed structured file system where the large files are divided into blocks of pre-determined size and is stored a stored across a cluster of one or multiple machines.
2. The data nodes are spread across various machines but we can run several data nodes on a single machine.
3. To maintain , store the large data files we use hadoop.
4. The architecture mainly consists of 3 components:

Map Reduce

HDFS(Hadoop Distributed File system).

YARN(Yet Another Resource Negotiator).

1. Hadoop work on Master-slave Architecture as Namenode and DataNode.

**Map Reduce :**ISa programming model that is used to access big data stored in hadoop facilities processing the data by splitting into smaller chunks and processing them in parallel. It combine the data from multiple server to return the consolidated output.

The Map() reduce function takes input from the data and produce the key, value pairs as output

**Mappers and Reducers** are the servers which run the Map and Reduce the function.

For example: We have 100 records in a file, 50 or 100 mappers are assigned to run the process. The framework decide the mapper based on size and memory block available on mapper sever.

After processing the framework shuffles and sort the data. A reducer only starts after mappers complete the process.

All the output having the same key are assigned to single Reducer.

**YARN:** Yarn is the Management layer that manages the workloads, implement security controls etc.

YARN provides resources for running an application.

It act as Resource Manager, Application Manager and Node manager.

**Workflow of Application in YARN:**

1. Client submit the application.
2. Container allocation is performed for starting the application Manager.
3. After this Registering the Manager with Resource Manager.
4. Manager notifies the node manager to launch containers.
5. Code gets executed in the container.
6. Stauts of the application is monitored.
7. Application manager gets disconnected with Resource .

**Cluster:** Collection of nodes is known as cluster.

**MasterNode :**

1. MasterNode maintain, manage and assign tasks to slave nodes.It runs on high configuration hardware.
2. MaterNode file system performs operations like opening, closing the directories and files etc.
3. Master node can have many slaves nodes but slave node can only have one master node.
4. Masternode records the metadata of all the files stored in the cluster such as size, location , permissions .

**Data Node:**

1. Data nodes are slave nodes and is no-expensive system which is not of high quality.
2. Data nodes stores the actual data.
3. Data nodes perform the read and write requests according to the requirement.
4. Hadoop is also configured to make the copy of these files that ensures the availability of data/
5. By default, the Replication factor is set to 3 and we can change it manually to 4 or any number according to the requirement.
6. This is because Hadoop is using commodity hardware which has the chances of crashing any time.

**Blocks:**

1. Blocks are the smallest continuous location where data is stored.
2. The default size of the block is 128MB.
3. For example : we have text file of 514Mb Size which is converted into blocks as 128MB,128MB,128MB,128MB,2MB.

**Advantages of Hadoop:**

1. Hadoop is open source and cost-effective hardware unlike the traditional Databases that need expensive hardware to deal with data. That means Hadoop is open source and inexpensive.
2. Hadoop is highly scalable as it can manage large amount of data by dividing into multiple machines in form of clusters and can be processed parallelly. The number of machine can vary according to the requirements of data. Whereas in Traditional RDBMS system can not be scaled to manage large amount of data.
3. Hadoop is designed in such a way it can handle any kind of dataset which can be structured, unstructured and semi-structured.
4. Hadoop use HDFS in which large files are broken to small size blocks and distributed among nodes available in hadoop cluster which gives High- level Performance as compare to other Systems.
5. Hadoop has an amazing feature of Fault Tolerance as it uses commodity software so it has higher chances of crash. In hadoop data is replicated on various data Nodes in cluster which ensure the availability in case of crash. So, In case of crash data is not lost it can be restored from other data nodes. Default, Fault Tolerance is 3, which means 3 copies of each data file is placed on different nodes.
6. Each data node process a small amount of data which results in low traffic and high performance.

**Basic Commands:**

1. Pwd : Present Working Directory
2. Cd : Change directory.
3. Cd ~ : goes to home directory
4. ls -a: shows all the hidden file in addition to visible one.
5. mkdir : Make Directory
6. rmdir: to remove directory
7. head: to view the first ten lines of the text file
8. tail: to view the last 10 lines of the file.
9. ls : list
10. -get : To get file from hadoop to local (Implementation shown in above examples).
11. -put : To copy file from local to hadoop.
12. -cat : display the contents of file.
13. -vi : To perform operations in file such as insertion
14. mv : To move a file from one directory to another
15. -rm : To remove the file
16. -touchz : to create an empty file in hadoop.
17. -count: use to count the number of directories
18. -chmod: use to change the permissions of a file
19. dfsadmin: to run HDFS as admin
20. $ hdfs dfsadmin -safemode leave: To leave safemode
21. $ hdfs dfsadmin -safemode get: To verify.

The detailed implementation of commands is shown in above examples.