

## Important Command related to Hadoop :

1. `hdfs dfs -ls <directory Name>`: List the files and directories in the current directory in HDFS.

Note - We can also use 'hadoop fs' command instead of 'hdfs dfs' command to interact with the Hadoop Distributed File System. Both 'hdfs dfs' and 'hadoop fs' provide similar functionality for working with files and directories in HDFS.

2. `hdfs dfs -mkdir <hdfs directory>`: Create a new directory in HDFS.

3. `hdfs dfs -touchz <hdfs file>`: Create a new empty file in HDFS.

4. `hdfs dfs -cat <hdfs file>`: Display the contents of a file in HDFS.

5. `hdfs dfs -rm <hdfs file>`: Delete a file from HDFS.

6. `hdfs dfs -put <local file> <hdfs directory>`: Copy a file from or, -Copy From Local the local file system to specific directory in HDFS.

7. `hdfs dfs -get <hdfs file> <local directory>`: Copy a file or, -copyToLocal from HDFS to the specified directory in the local file system.

8. `hdfs dfs -chmod <Permission> <hdfs file>`: Change the permission of a file in HDFS.

9. `hdfs dfs -mv <source_path> <destination path>` : used to move or rename files and directories in HDFS. It works in a similar way to the Linux 'MV' command.

Ex - To move a file from one directory to another in HDFS:

→ hdfs dfs -mv /user/mynname/file.txt /user/mynname/new\_directory/file.txt

• To rename a file in HDFS:

⇒ hdfs dfs -mv /user/mynname/old\_name.txt /user/mynname/new\_name.txt.

• To move a directory and all its contents to a new location in HDFS:

⇒ hdfs dfs -mv /user/mynname/old\_directory /user/mynname/new\_directory

10. hdfs dfs -moveFromLocal <local file> <hdfs file>

Move a file from the local file system to HDFS.

11. hdfs dfs -moveToLocal <hdfs file> <hdfs file>:

Move a file from HDFS to the local file system.

12. hdfs dfs -appendToFile <local file> <hdfs file>:

Append data from a file in the local file system to a file in HDFS.

13. hdfs dfs -count <hdfs file> : Displays the number of files and directories in the specified directory in HDFS.

14. `hdfs dfs -cp <source-path> <destination-path>`:

Used to copy file from one location to another in HDFS.

15. `hdfs dfs -du <path of file>`: Used to estimate the size of a file or directory in HDFS.

16. `hdfs dfs -chown <new-owner> <path>`: Used to change the owner of a file or directory in HDFS.

17. `hdfs dfs -chgrp <new-group> <path>`: Used to change the group of a file or directory in HDFS.

⑪ `jps` → Display the status of Java virtual machines running on the system.

18. `hdfs fsck <file-name>`: Check the health of a file in HDFS.

19. `hdfs dfs -merge` : Merge multiple files in HDFS into a single file.

Ex → `hdfs dfs -merge /data/file1.txt /data/file2.txt /data/merged-file.txt`

Hive :- Hive is a data warehouse system that is used to query and analyze the large database datasets stored in the HDFS. Hive uses a query language called HiveQL, which is similar to SQL.

We can run Hive commands in cloudera quick-start in two ways :-

(i) Using terminal - Open the terminal in Cloudera Quickstart and then type 'hive' and hit enter. This will start the shell.

Note :- While entering in to the Hive shell if it shows some error like Name node is in safe mode then you can turn off the safe mode by typing below command in terminal.

- For Root user  $\Rightarrow$  hdfs dfsadmin -safemode leave
- Normal user  $\Rightarrow$  sudo -u hdfs hdfs dfsadmin -safemode leave.
- You can check the status of safe mode by using this command :-  
sudo -u hdfs hdfs dfsadmin -safemode get
- After typing the above command if it is still in safemode, then one of the reason would be not enough space in your node, you can check your node disk usages using :-

`df -h`

Namenode enters into safemode when there is shortage of memory. As result the HDFS becomes read only.

(ii) Using Browser :- Open the browser in cloudera quickstart and then click on 'Hue' which is added in bookmark tab. Type the username and password 'cloudera' and login.

You can use all HiveQL command in it. It will give you a blank space where you can write all the command and then click on play button (▶).

### Hive Commands :-

(i) Create :- Used to create a table or data-base.

Ex - Create database mydata;

Create table mytable (Id int, Name string, city string) row format delimited fields terminated by ',';

2. Show : Used to show Database , Table , Properties, etc.

Ex - show databases; | show tables;

The database creates in a default location of the Hive warehouse . In cloudera , Hive database store in a /user/hive/warehouse .

3. Use : The command is used to use database .

Ex - use mydata;

4. Describe : It describes the table columns .

Ex - describe mytable ;

Id	int
Name	String
city	String

5. To insert the record into already created table :-

• insert into table mytable values (1,'Satish','Patna')

→ Provide the value <sup>in sequence</sup> according to the table

Created. Write the string value in single quote ('').

• Insert the multiple record at a time .

insert into table mytable values (2,'sonpal','Ranchi'),  
(3,'sonu','Paris'),(4,'Ram','delhi');

• Display <sup>all</sup> the record of the table -

select \* from mytable;

1	Satish	Patna
2	Sonpal	Ranchi
3	Sonu	Paris
4	Ram	delhi

7. Display the particular field from table:

select Id, Name, city from mytable;

7. Load data:

Hive provides us the functionality to load pre-created table entities either from our local file system or from HDFS.

The load data statement is used to load data into the hive table.

→ load data input 'luser/cloudera/people' over-write into table employee ;  
table name.

→ For using the above command first create the a file into local file system and write some data into it - (comma Separated value).

cat > people

1, Ahu, 12000, software

2, Suraj, 35000, Software

3, Kunal, 45000, Data scientist

4, Deepu, 35000, Web developer

→ Save it.

- Now copy this file to hadoop or you can  
 → hdfs dfs -put people /user/cloudera
- Now get in to Hive and create a database and inside it create a table.

hive> Create database mydata;

hive> use database mydata;

hive> create table employee (Id int, Name string, Salary int, Field string) row format  
 delimited fields terminated by ',';

→ View the table

Select \* from employee;

Output:-	1	Anu	12000	Software
	2	Suraj	35000	Software
	3	Kunal	45000	Data scientist
	4	Deepu	35000	Web developer

→ You can perform multiple operation table like :-

→ Hive Aggregate function list :-

(i) COUNT () : Return the count of all rows in a table including rows containing NULL values. Ignore duplicates by using DISTINCT  
 Return: BIGINT

(ii) SUM() : Return the sum of all values in column  
 Return: DOUBLE

(iii) AVG() : Return the average of all values in a column.

Return: DOUBLE

(iv) MIN() : Return the minimum value of the column from all rows.

Return: DOUBLE

(v) MAX() ; Return the maximum value of the column from all rows.

## Hive Aggregate function Examples:-

- (i) ~~Select~~ select count(\*) from employee;
- (ii) Select count(salary) from employee;
- (iii) Select sum(salary) from employee;
- (iv) Select avg(salary) from employee;
- (v) Select min(salary) from employee;
- (vi) Select max(salary) from employee;