1

**SET UP AND CONFIGURATION HADOOP USING CLOUDERA CREATING A HDFS SYSTEM WITH MINIMUM 1 NAME NODE AND 1 DATA NODES HDFS COMMANDS**

## Unit Structure :

* 1. Objectives
  2. Prerequisite
  3. GUI Configuration
  4. Command Line Configuration
  5. Summary
  6. Sample Questions
  7. References

# OBJECTIVES

Hadoop file system stores the data in multiple copies. Also, it9s a cost- effective solution for any business to store their data efficiently. HDFS Operations acts as the key to open the vaults in which you store the data to be available from remote locations. This chapter describes how to set up and edit the deployment configuration files for HDFS

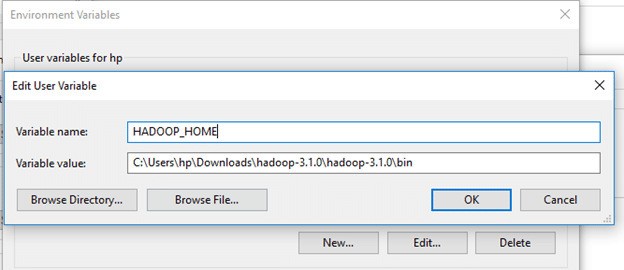
# PREREQUISITE: TO INSTALL HADOOP, YOU SHOULD HAVE JAVA VERSION 1.8 IN YOUR SYSTEM.

Check your java version through this command on command prompt

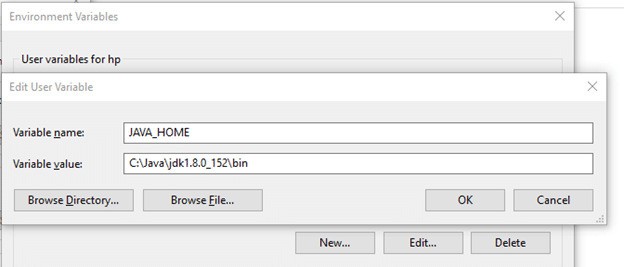
## Java -version

Create a new user variable. Put the Variable\_name as HADOOP\_HOME and Variable\_value as the path of the bin folder where you extracted hadoop.

Big Data Analytics and Visualization Lab

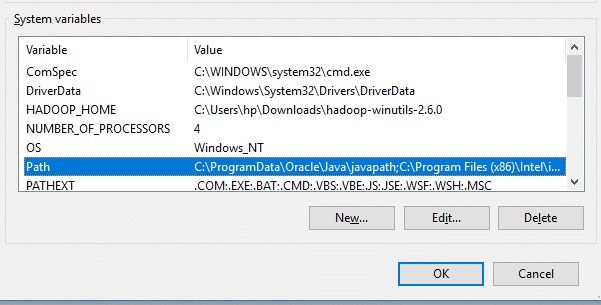


Likewise, create a new user variable with variable name as JAVA\_HOME and variable value as the path of the bin folder in the Java directory.

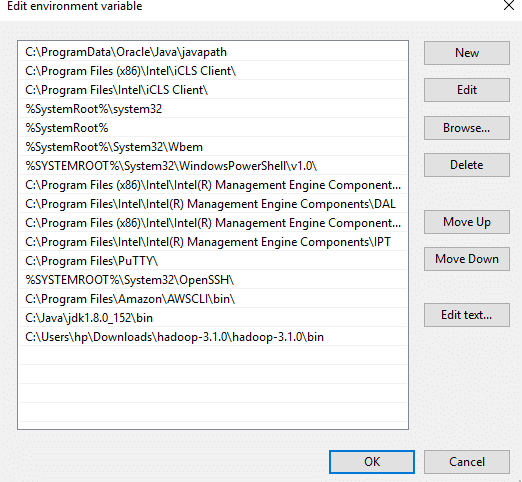


Now we need to set Hadoop bin directory and Java bin directory path in system variable path.

Edit Path in system variable



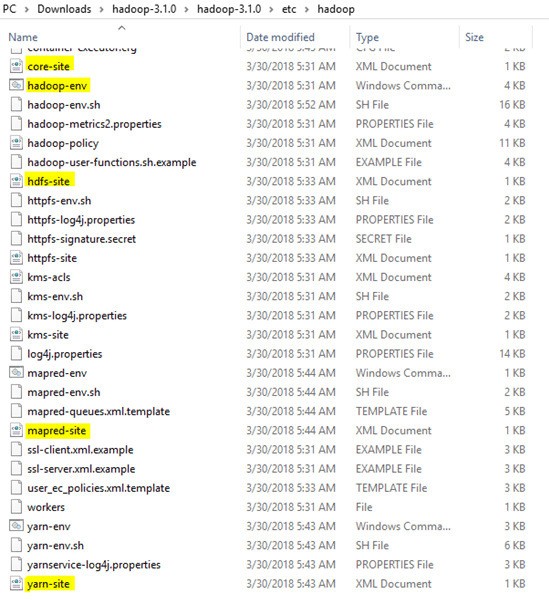
Click on New and add the bin directory path of Hadoop and Java in it.



# GUI CONFIGURATIONS

Set up and Configuration Hadoop using Cloudera creating a HDFS System with Minimum 1 Name Node

and 1 Data Nodes HDFS Commands

Now we need to edit some files located in the hadoop directory of the etc folder where we installed hadoop. The files that need to be edited have been highlighted.

Big Data Analytics and Visualization Lab

## Edit the file core-site.xml in the hadoop directory. Copy this xml property in the configuration in the file

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>

## Edit mapred-site.xml and copy this property in the configuration

<configuration>

<property>

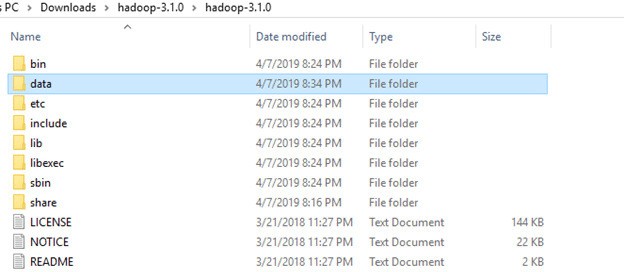
<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

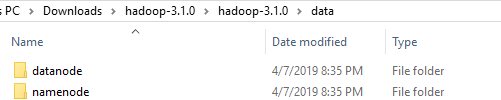
</configuration>

## Create a folder 8data9 in the hadoop directory



1. **Create a folder with the name 8datanode9 and a folder 8namenode9 in**

## this data directory



1. **Edit the file hdfs-site.xml and add below property in the configuration**

Note: The path of namenode and datanode across value would be the path of the datanode and namenode folders you just created.

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>C:\Users\hp\Downloads\hadoop-3.1.0\hadoop- 3.1.0\data\namenode</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value> C:\Users\hp\Downloads\hadoop-3.1.0\hadoop- 3.1.0\data\datanode</value>

</property>

</configuration>

## Edit the file yarn-site.xml and add below property in the configuration

<configuration>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.auxservices.mapreduce.shuffle.clas s</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

</configuration>

Set up and Configuration Hadoop using Cloudera creating a HDFS System with Minimum 1 Name Node

and 1 Data Nodes HDFS Commands

Big Data Analytics and Visualization Lab

## Edit hadoop-env.cmd and replace %JAVA\_HOME% with the path of the java folder where your jdk 1.8 is installed

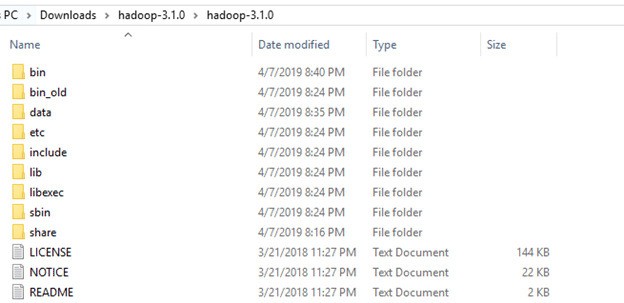


1. **Hadoop needs windows OS specific files which does not come with default download of hadoop.**

To include those files, replace the bin folder in hadoop directory with the bin folder provided in this github link.

https://github.com/s911415/apache-hadoop-3.1.0-winutils

Download it as zip file. Extract it and copy the bin folder in it. If you want to save the old bin folder, rename it like bin\_old and paste the copied bin folder in that directory.



Check whether hadoop is successfully installed by running this command on cmd-

## hadoop –version Format the NameNode

Formatting the NameNode is done once when hadoop is installed and not for running hadoop filesystem, else it will delete all the data inside HDFS. Run this command-

## hdfs namenode –format

Now change the directory in cmd to sbin folder of hadoop directory with this command,

Start namenode and datanode with this command 3

## start-dfs.cmd

Two more cmd windows will open for NameNode and DataNode Now start yarn through this command-

## start-yarn.cmd

Note: Make sure all the 4 Apache Hadoop Distribution windows are up n running. If they are not running, you will see an error or a shutdown message. In that case, you need to debug the error.

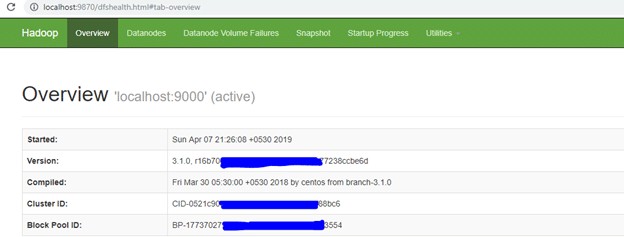
To access information about resource manager current jobs, successful and failed jobs, go to this link in browser-

http://localhost:8088/cluster

To check the details about the hdfs (namenode and datanode), http://localhost:9870/

Set up and Configuration Hadoop using Cloudera creating a HDFS System with Minimum 1 Name Node

and 1 Data Nodes HDFS Commands



**Hadoop HDFS Commands**

With the help of the HDFS commands, we can perform Hadoop HDFS file operations like changing the file permissions, viewing the file contents, creating files or directories, copying file/directory from the local file system to HDFS or vice-versa, etc.

Before starting with the HDFS command, we have to start the Hadoop services.

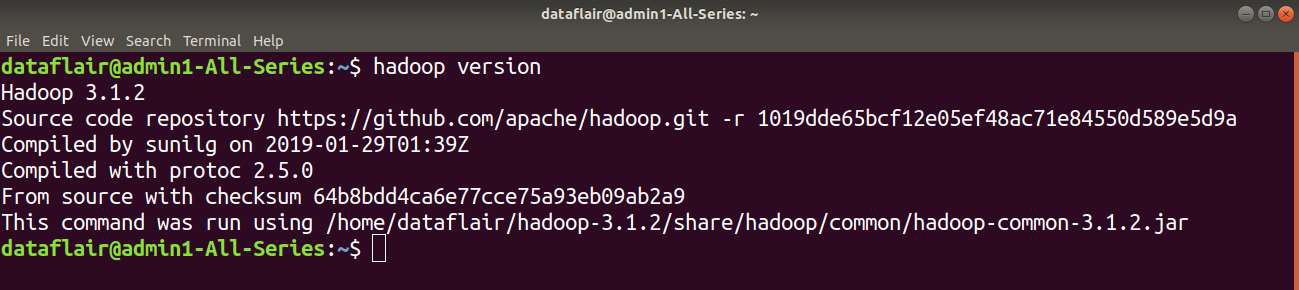
In this practical, we have mentioned the Hadoop HDFS commands with their usage, examples, and description.

* 1. **version**

**Hadoop HDFS version Command Usage:**

hadoop version

The Hadoop fs shell command **version** prints the Hadoop version.

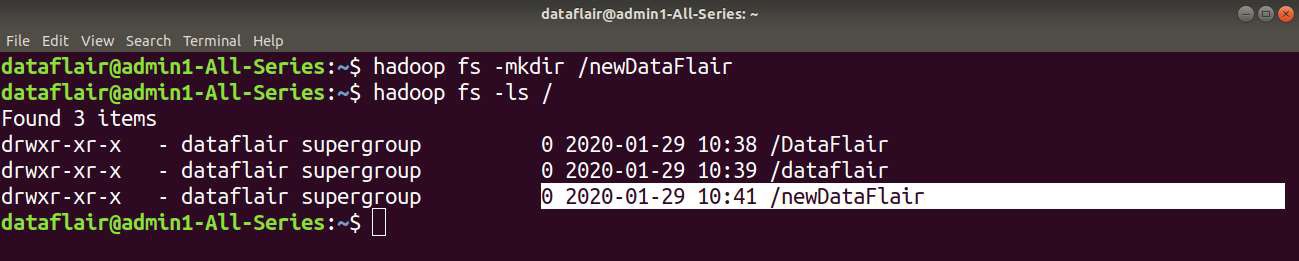


* 1. **mkdir**

**Hadoop HDFS mkdir Command Usage:**

hadoop fs –mkdir /path/directory\_name

we create a new directory named directory\_name in HDFS using the **mkdir** command.



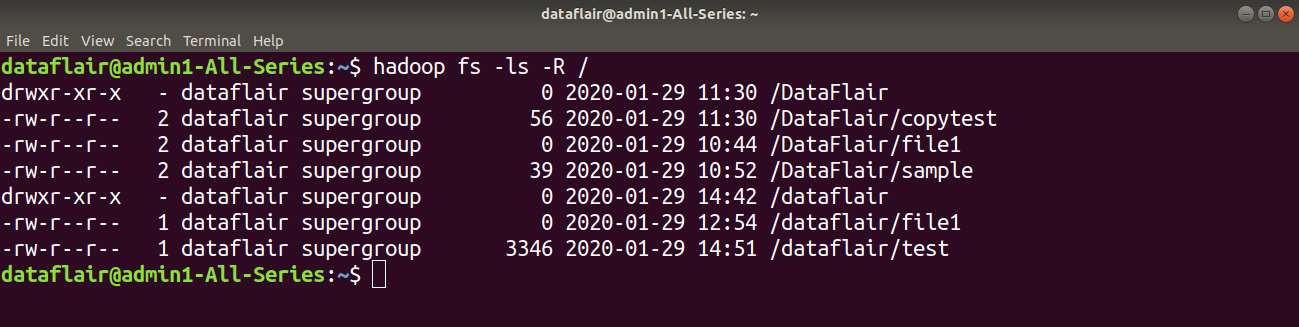
* 1. **ls**

**Hadoop HDFS ls Command Usage:**

hadoop fs -ls /path

**Hadoop HDFS ls Command Description:**

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.



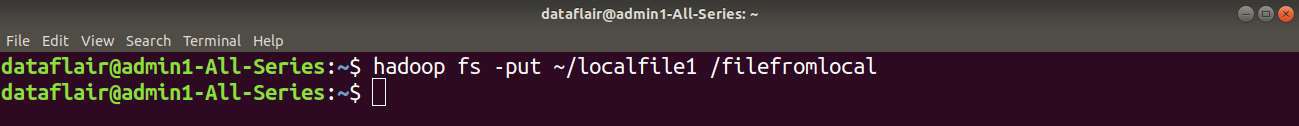
* 1. **put**

**Hadoop HDFS put Command Usage:**

haoop fs -put <localsrc> <dest>

**Hadoop HDFS put Command Example:**

Here in this example, we are trying to copy localfile1 of the local file system to the Hadoop filesystem.



**Hadoop HDFS put Command Description:**

The Hadoop fs shell command **put** is similar to the **copyFromLocal**, which copies files or directory from the local filesystem to the destination in the Hadoop filesystem.

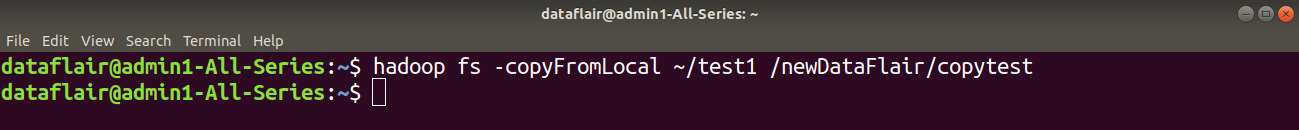
* 1. **copyFromLocal**

**Hadoop HDFS copyFromLocal Command Usage:**

hadoop fs -copyFromLocal <localsrc> <hdfs destination>

**Hadoop HDFS copyFromLocal Command Example:**

Here in the below example, we are trying to copy the ‘test1’ file present in the local file system to the newDataFlair directory of Hadoop.



This command copies the file from the local file system to HDFS.

* 1. **get**

**Hadoop HDFS get Command Usage:**

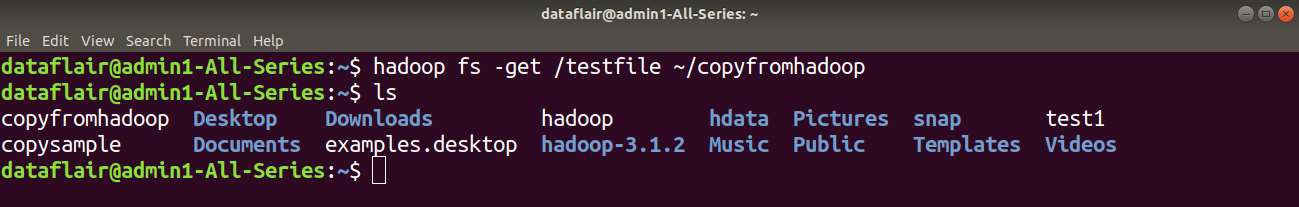
hadoop fs -get <src> <localdest>

**Hadoop HDFS get Command Example:**

In this example, we are trying to copy the ‘testfile’ of the hadoop filesystem to the local file system.

**Hadoop HDFS get Command Description:**

The Hadoop fs shell command get copies the file or directory from the Hadoop file system to the local file system.



* 1. **copyToLocal**

**Hadoop HDFS copyToLocal Command Usage:**

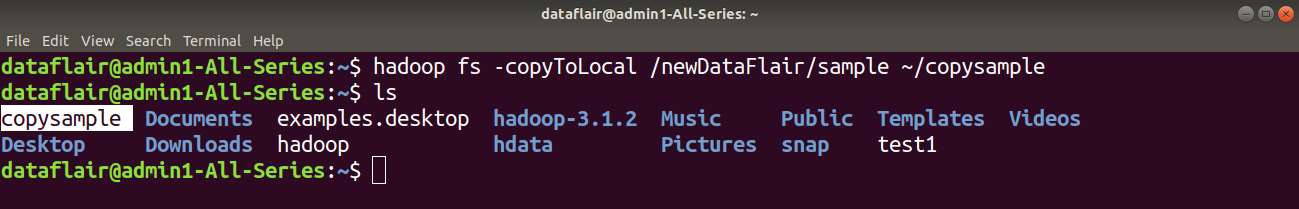
hadoop fs -copyToLocal <hdfs source> <localdst>

**Hadoop HDFS copyToLocal Command Example:**

Here in this example, we are trying to copy the ‘sample’ file present in the newDataFlair directory of HDFS to the local file system.

**adoop HDFS copyToLocal Description:**

**copyToLocal** command copies the file from HDFS to the local file system.



* 1. **cat**

**Hadoop HDFS cat Command Usage:**

hadoop fs –cat /path\_to\_file\_in\_hdfs

**Hadoop HDFS cat Command Example:**

Here in this example, we are using the cat command to display the content of the ‘sample’ file present in newDataFlair directory of HDFS.

**Hadoop HDFS cat Command Description:**

The **cat** command reads the file in HDFS and displays the content of the file on console or stdout.



* 1. **mv**

**Hadoop HDFS mv Command Usage:**

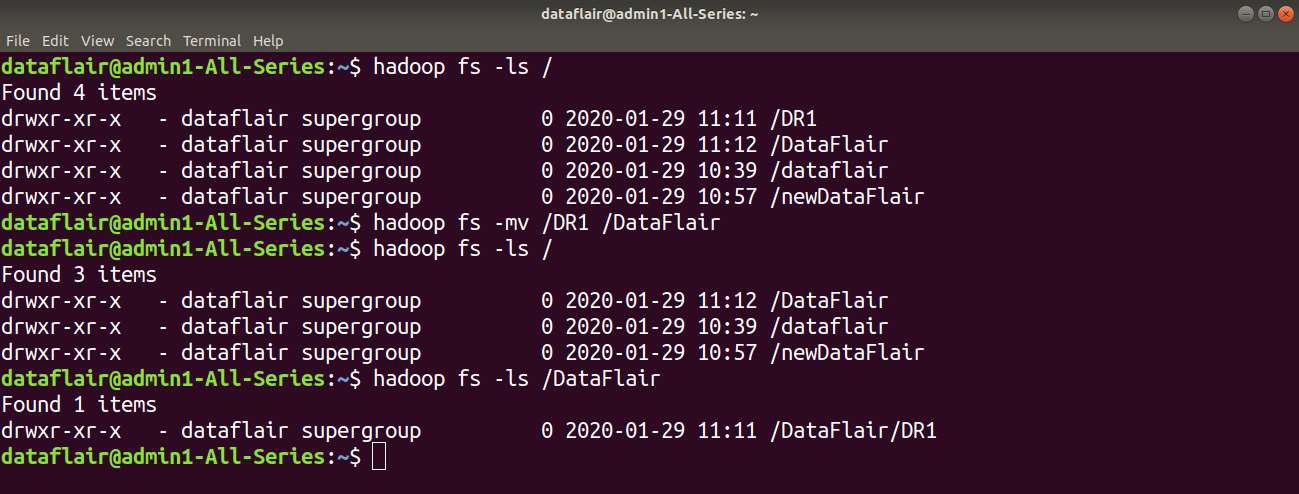
hadoop fs -mv <src> <dest>

**Hadoop HDFS mv Command Example:**

In this example, we have a directory ‘DR1’ in HDFS. We are using **mv** command to move the DR1 directory to the DataFlair directory in HDFS.

**Hadoop HDFS mv Command Description:**

The HDFS mv command moves the files or directories from the source to a destination within **HDFS.**



* 1. **cp**

**Hadoop HDFS cp Command Usage:**

hadoop fs -cp <src> <dest>

**Hadoop HDFS cp Command Example:**

In the below example we are copying the ‘file1’ present in newDataFlair directory in HDFS to the dataflair directory of HDFS.

**Hadoop HDFS cp Command Description:**

The **cp** command copies a file from one directory to another directory within the HDFS.

