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 :

1.1 OBJECTIVES

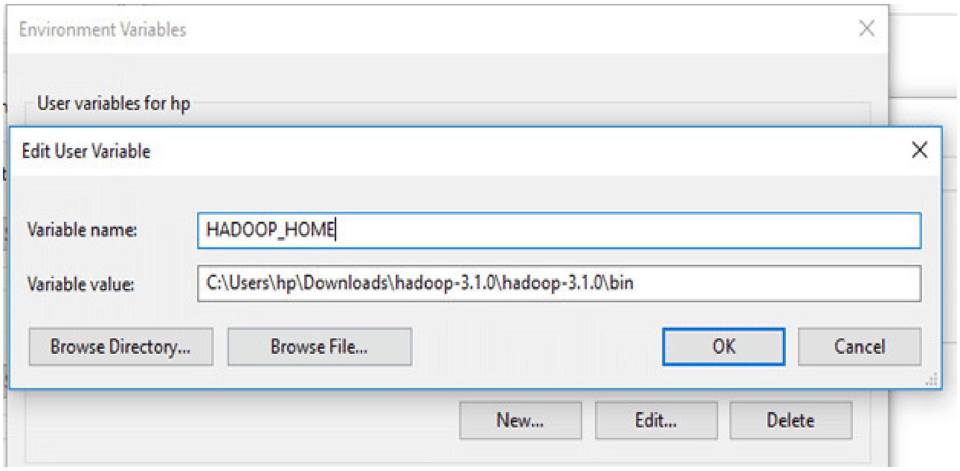
Hadoop file system stores the data in multiple copies. Also, it’s a cost-effective solution for any business to store their data efficiently. HDFS Operations acts as the key to open the vaults in which you store the data to be available from remote locations. This chapter describes how to set up and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

1.2 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

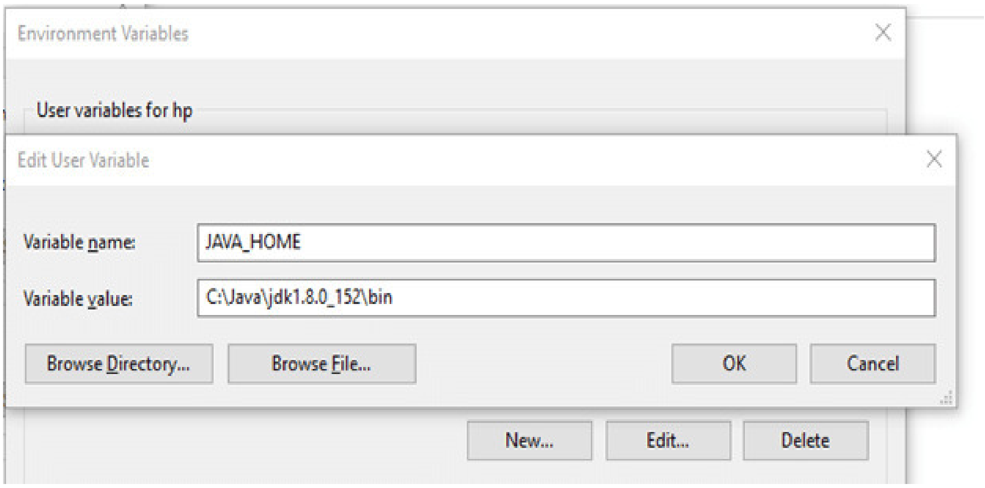
Create a new user variable. Put the Variable_name as HADOOP_HOME and Variable_value as the path of the bin folder where you extracted hadoop.



Variable name: HADOOP_HOME

Variable value: C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\bin

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.

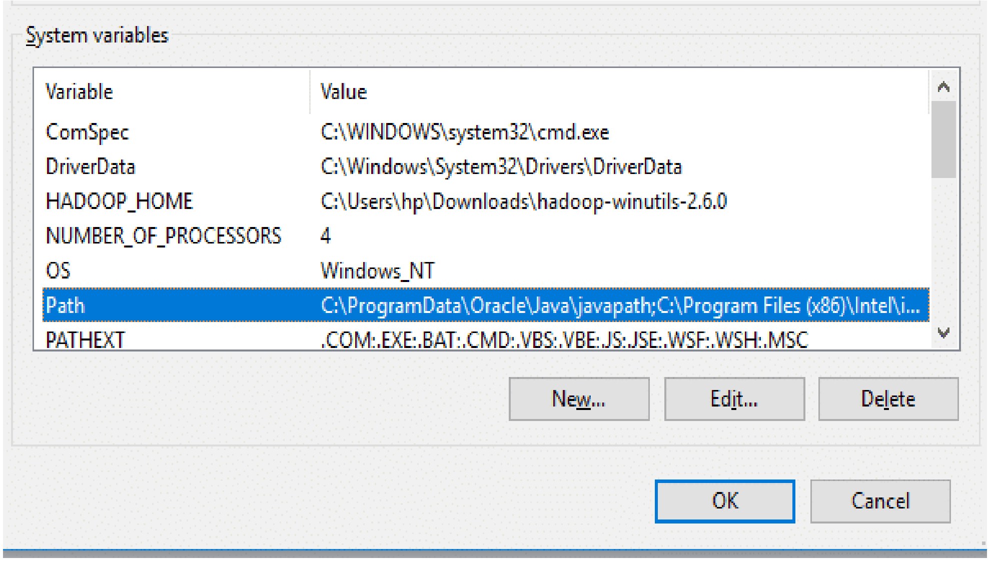


Variable name: JAVA_HOME

Variable value: C:\Java\jdk1.8.0_152\bin

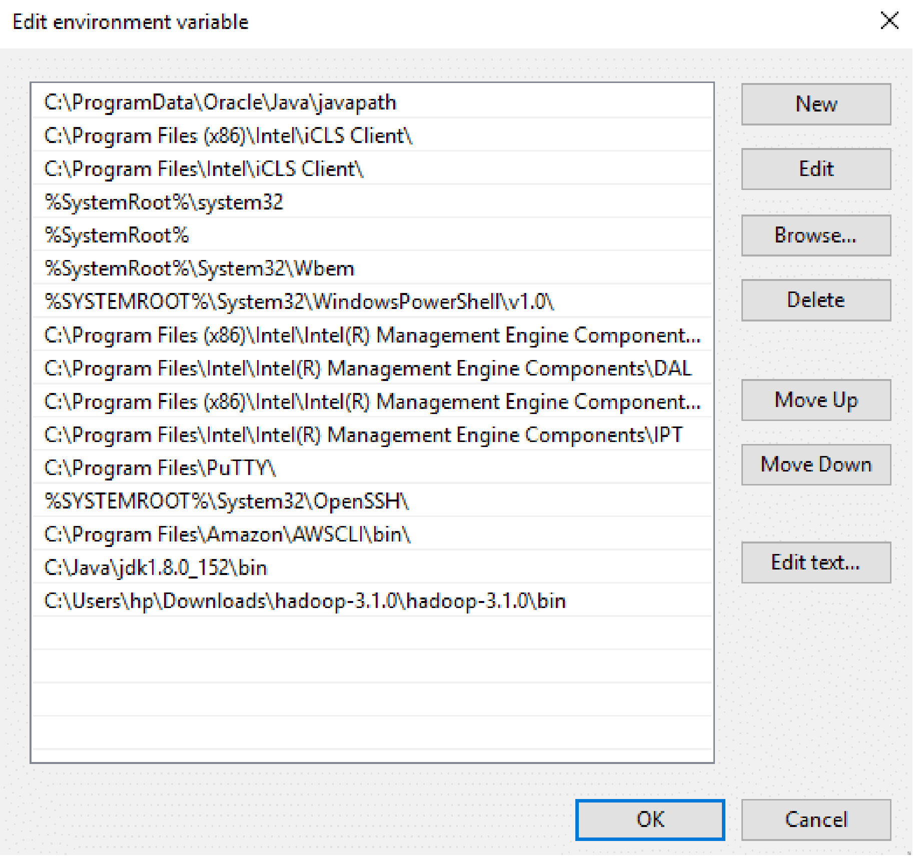
Now we need to set Hadoop bin directory and Java bin directory path in system variable path.

Edit Path in system variable



Variable	Value
ComSpec	C:\WINDOWS\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
HADOOP_HOME	C:\Users\hp\Downloads\hadoop-winutils-2.6.0
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\ProgramData\Oracle\Java\javapath;C:\Program Files (x86)\Intel\i...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

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

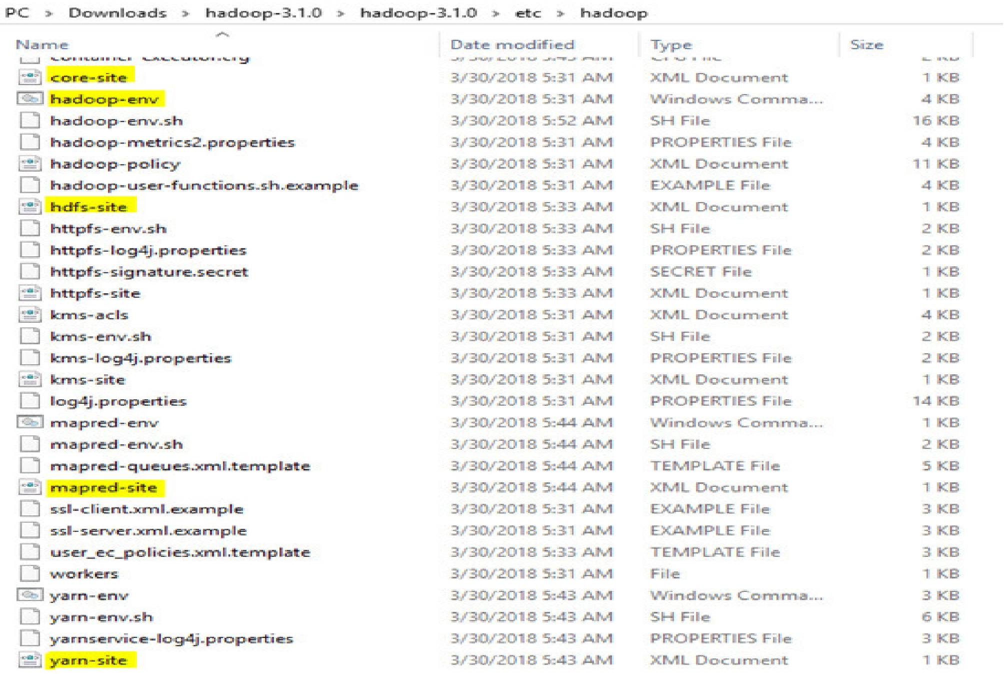


Edit environment variable

```
C:\ProgramData\Oracle\Java\javapath
C:\Program Files (x86)\Intel\iCLS Client\
C:\Program Files\Intel\iCLS Client\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Component...
C:\Program Files\Intel\Intel(R) Management Engine Components\DAL
C:\Program Files (x86)\Intel\Intel(R) Management Engine Component...
C:\Program Files\Intel\Intel(R) Management Engine Components\IPT
C:\Program Files\PuTTY\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\Amazon\AWSCLI\bin\
C:\Java\jdk1.8.0_152\bin
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\bin
```

1.3 GUI CONFIGURATIONS

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.



PC > Downloads > hadoop-3.1.0 > hadoop-3.1.0 > etc > hadoop

Name	Date modified	Type	Size
core-site	3/30/2018 5:31 AM	XML Document	1 KB
hadoop-env	3/30/2018 5:31 AM	Windows Comma...	4 KB
hadoop-env.sh	3/30/2018 5:52 AM	SH File	16 KB
hadoop-metrics2.properties	3/30/2018 5:31 AM	PROPERTIES File	4 KB
hadoop-policy	3/30/2018 5:31 AM	XML Document	11 KB
hadoop-user-functions.sh.example	3/30/2018 5:31 AM	EXAMPLE File	4 KB
hdfs-site	3/30/2018 5:33 AM	XML Document	1 KB
httpfs-env.sh	3/30/2018 5:33 AM	SH File	2 KB
httpfs-log4j.properties	3/30/2018 5:33 AM	PROPERTIES File	2 KB
httpfs-signature.secret	3/30/2018 5:33 AM	SECRET File	1 KB
httpfs-site	3/30/2018 5:33 AM	XML Document	1 KB
kms-acls	3/30/2018 5:31 AM	XML Document	4 KB
kms-env.sh	3/30/2018 5:31 AM	SH File	2 KB
kms-log4j.properties	3/30/2018 5:31 AM	PROPERTIES File	2 KB
kms-site	3/30/2018 5:31 AM	XML Document	1 KB
log4j.properties	3/30/2018 5:31 AM	PROPERTIES File	14 KB
mapred-env	3/30/2018 5:44 AM	Windows Comma...	1 KB
mapred-env.sh	3/30/2018 5:44 AM	SH File	2 KB
mapred-queues.xml.template	3/30/2018 5:44 AM	TEMPLATE File	5 KB
mapred-site	3/30/2018 5:44 AM	XML Document	1 KB
ssl-client.xml.example	3/30/2018 5:31 AM	EXAMPLE File	3 KB
ssl-server.xml.example	3/30/2018 5:31 AM	EXAMPLE File	3 KB
user_ec_policies.xml.template	3/30/2018 5:33 AM	TEMPLATE File	3 KB
workers	3/30/2018 5:31 AM	File	1 KB
yarn-env	3/30/2018 5:43 AM	Windows Comma...	3 KB
yarn-env.sh	3/30/2018 5:43 AM	SH File	6 KB
yarnservice-log4j.properties	3/30/2018 5:43 AM	PROPERTIES File	3 KB
yarn-site	3/30/2018 5:43 AM	XML Document	1 KB

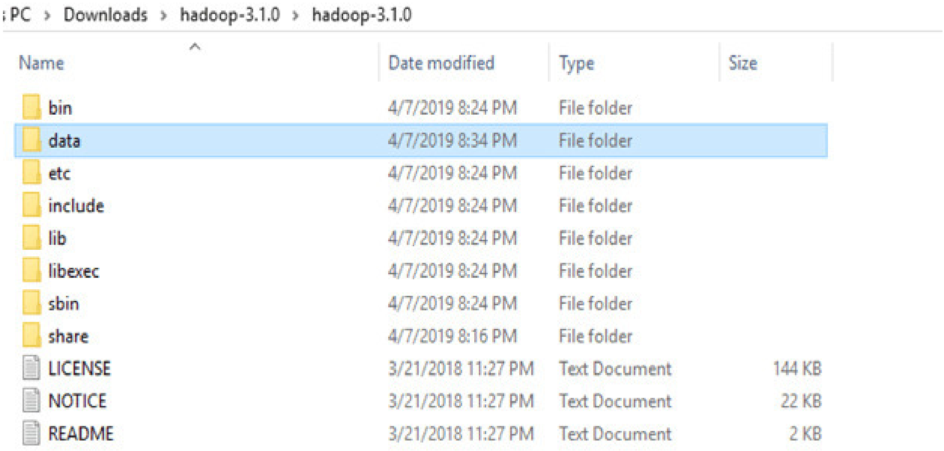
1. **Edit the file core-site.xml in the hadoop directory. Copy this xml property in the configuration in the file**

```
<configuration>
  <property>
     <name>fs.defaultFS</name>
     <value>hdfs://localhost:9000</value>
  </property>
</configuration>
```

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

```
<configuration>
  <property>
     <name>mapreduce.framework.name</name>
     <value>yarn</value>
  </property>
</configuration>
```

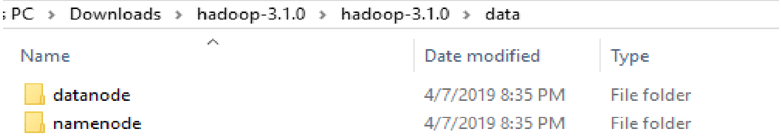
3. **Create a folder 'data' in the hadoop directory**



; PC › Downloads › hadoop-3.1.0 › hadoop-3.1.0

Name	Date modified	Type	Size
bin	4/7/2019 8:24 PM	File folder	
data	4/7/2019 8:34 PM	File folder	
etc	4/7/2019 8:24 PM	File folder	
include	4/7/2019 8:24 PM	File folder	
lib	4/7/2019 8:24 PM	File folder	
libexec	4/7/2019 8:24 PM	File folder	
sbin	4/7/2019 8:24 PM	File folder	
share	4/7/2019 8:16 PM	File folder	
LICENSE	3/21/2018 11:27 PM	Text Document	144 KB
NOTICE	3/21/2018 11:27 PM	Text Document	22 KB
README	3/21/2018 11:27 PM	Text Document	2 KB

4. **Create a folder with the name 'datanode' and a folder 'namenode' in this data directory**



; PC › Downloads › hadoop-3.1.0 › hadoop-3.1.0 › data

Name	Date modified	Type
datanode	4/7/2019 8:35 PM	File folder
namenode	4/7/2019 8:35 PM	File folder

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

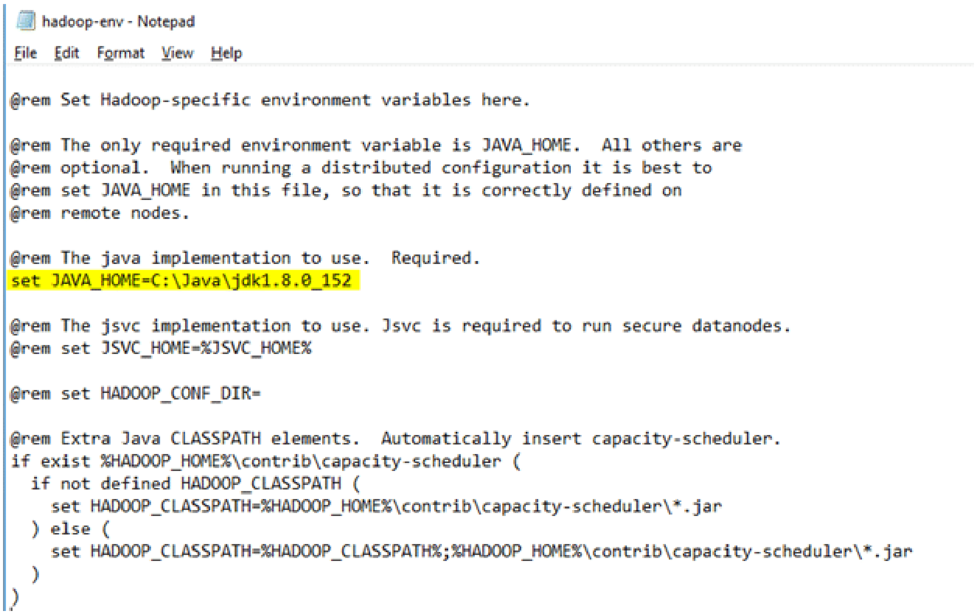
Note: The path of namenode and datanode across value would be the path of the datanode and namenode folders you just created.

```
<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>
```

6. 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>
```

7. **Edit hadoop-env.cmd and replace %JAVA_HOME% with the path of the java folder where your jdk 1.8 is installed**



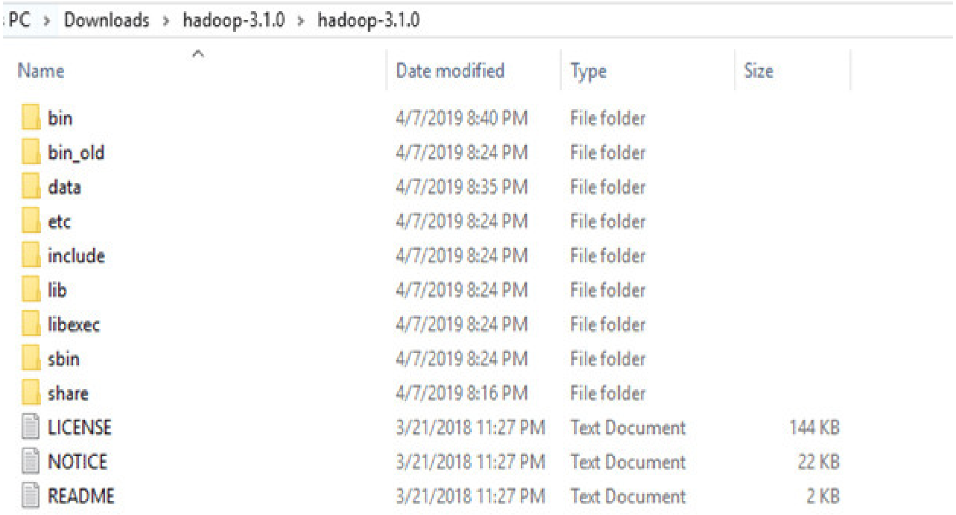
```
@rem Set Hadoop-specific environment variables here.

@rem The only required environment variable is JAVA_HOME.  All others are
@rem optional.  When running a distributed configuration it is best to
@rem set JAVA_HOME in this file, so that it is correctly defined on
@rem remote nodes.

@rem The java implementation to use.  Required.
set JAVA_HOME=C:\Java\jdk1.8.0_152

@rem The jsvc implementation to use. Jsvc is required to run secure datanodes.
@rem set JSVC_HOME=%JSVC_HOME%

@rem set HADOOP_CONF_DIR=

@rem Extra Java CLASSPATH elements.  Automatically insert capacity-scheduler.
if exist %HADOOP_HOME%\contrib\capacity-scheduler (
  if not defined HADOOP_CLASSPATH (
    set HADOOP_CLASSPATH=%HADOOP_HOME%\contrib\capacity-scheduler\*.jar
  ) else (
    set HADOOP_CLASSPATH=%HADOOP_CLASSPATH%;%HADOOP_HOME%\contrib\capacity-scheduler\*.jar
  )
)
```

8. **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.



Name	Date modified	Type	Size
bin	4/7/2019 8:40 PM	File folder	
bin_old	4/7/2019 8:24 PM	File folder	
data	4/7/2019 8:35 PM	File folder	
etc	4/7/2019 8:24 PM	File folder	
include	4/7/2019 8:24 PM	File folder	
lib	4/7/2019 8:24 PM	File folder	
libexec	4/7/2019 8:24 PM	File folder	
sbin	4/7/2019 8:24 PM	File folder	
share	4/7/2019 8:16 PM	File folder	
LICENSE	3/21/2018 11:27 PM	Text Document	144 KB
NOTICE	3/21/2018 11:27 PM	Text Document	22 KB
README	3/21/2018 11:27 PM	Text Document	2 KB

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 –

start-dfs.cmd

Two more cmd windows will open for NameNode and DataNode

Now start yarn through this command-

start-yarn.cmd

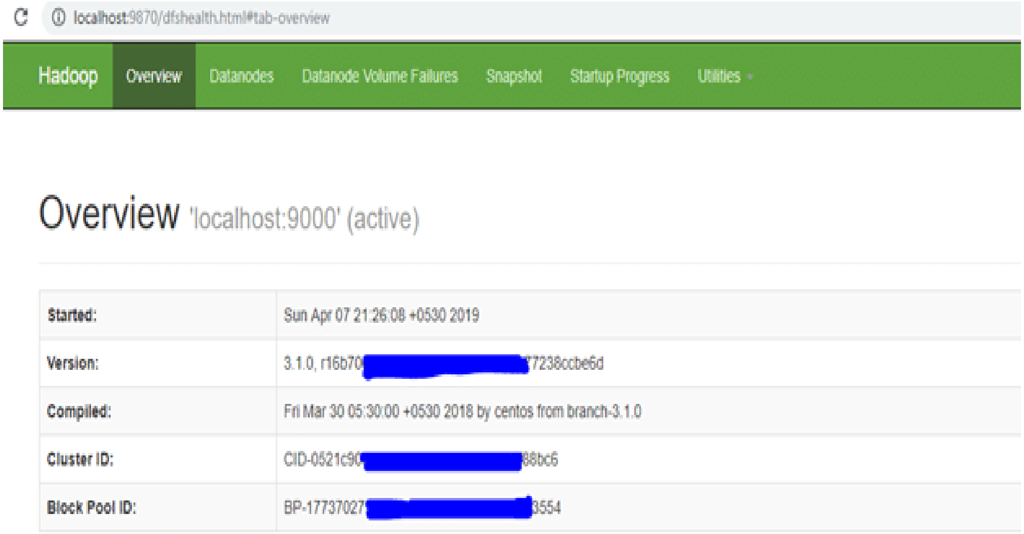
Note: Make sure all the 4 Apache Hadoop Distribution windows are up n running. If they are not running, you will see an error or a shutdown message. In that case, you need to debug the error.

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/



Overview 'localhost:9000' (active)

Started:	Sun Apr 07 21:26:08 +0530 2019
Version:	3.1.0, r16b70[redacted]7238ccbe6d
Compiled:	Fri Mar 30 05:30:00 +0530 2018 by centos from branch-3.1.0
Cluster ID:	CID-0521c90[redacted]88bc6
Block Pool ID:	BP-17737027[redacted]3554

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.

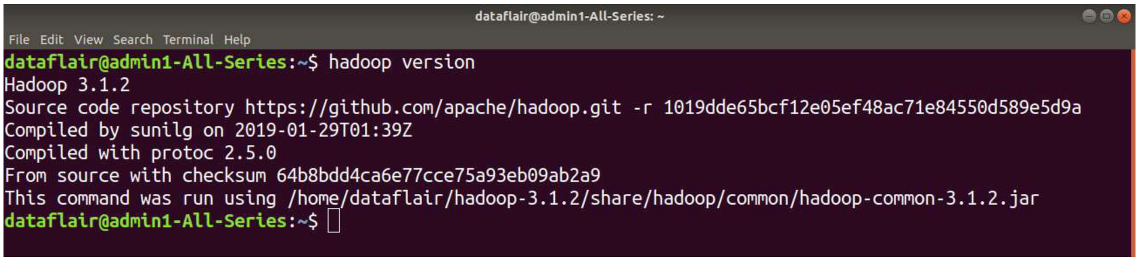
In this practical, we have mentioned the Hadoop HDFS commands with their usage, examples, and description.

1. version

Hadoop HDFS version Command Usage:

hadoop version

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


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

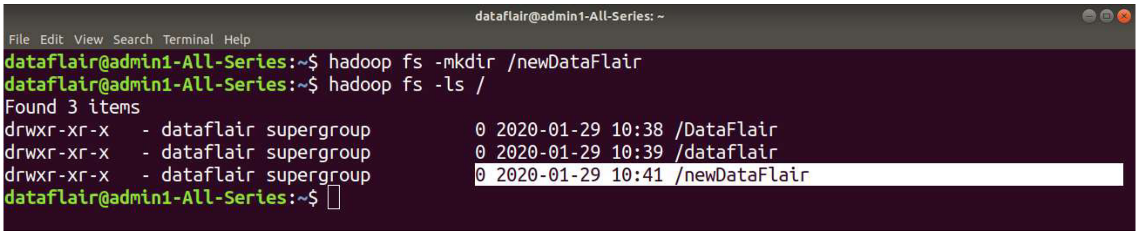
```
dataflair@admin1-All-Series:~$ hadoop version
Hadoop 3.1.2
Source code repository https://github.com/apache/hadoop.git -r 1019dde65bcf12e05ef48ac71e84550d589e5d9a
Compiled by sunilg on 2019-01-29T01:39Z
Compiled with protoc 2.5.0
From source with checksum 64b8bdd4ca6e77cce75a93eb09ab2a9
This command was run using /home/dataflair/hadoop-3.1.2/share/hadoop/common/hadoop-common-3.1.2.jar
dataflair@admin1-All-Series:~$ 
```

2. 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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -mkdir /newDataFlair
dataflair@admin1-All-Series:~$ hadoop fs -ls /
Found 3 items
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:38 /DataFlair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:39 /dataflair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:41 /newDataFlair
dataflair@admin1-All-Series:~$ 
```

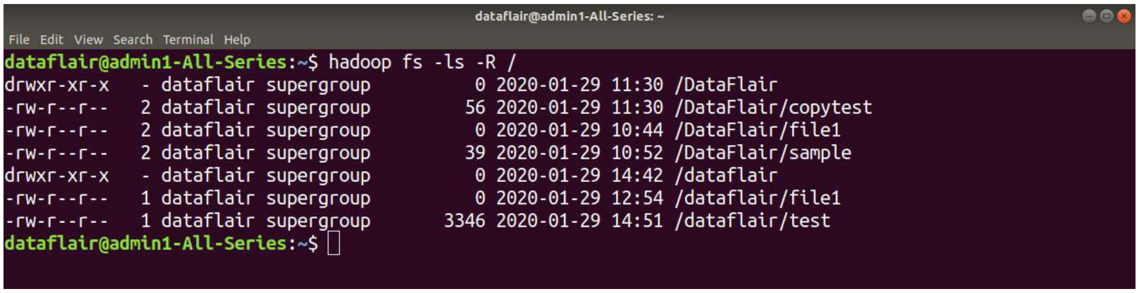
3. ls

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -ls -R /
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 11:30 /DataFlair
-rw-r--r--   2 dataflair supergroup         56 2020-01-29 11:30 /DataFlair/copytest
-rw-r--r--   2 dataflair supergroup          0 2020-01-29 10:44 /DataFlair/file1
-rw-r--r--   2 dataflair supergroup         39 2020-01-29 10:52 /DataFlair/sample
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 14:42 /dataflair
-rw-r--r--   1 dataflair supergroup          0 2020-01-29 12:54 /dataflair/file1
-rw-r--r--   1 dataflair supergroup       3346 2020-01-29 14:51 /dataflair/test
dataflair@admin1-All-Series:~$ 
```

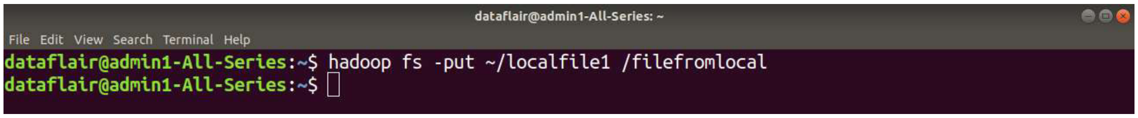
4. put

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -put ~/localfile1 /filefromlocal
dataflair@admin1-All-Series:~$ 
```

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.

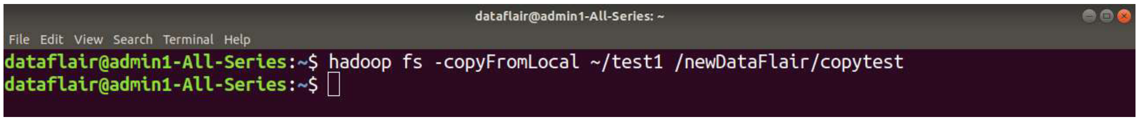
5. copyFromLocal

Hadoop HDFS copyFromLocal Command Usage:

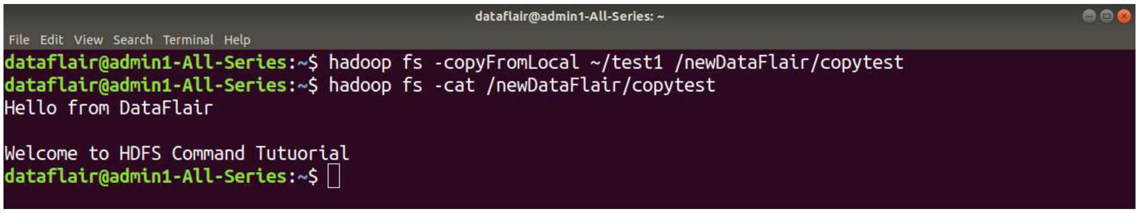
hadoop fs -copyFromLocal <localsrc> <hdfs destination>

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -copyFromLocal ~/test1 /newDataFlair/copytest
dataflair@admin1-All-Series:~$ 
```


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -copyFromLocal ~/test1 /newDataFlair/copytest
dataflair@admin1-All-Series:~$ hadoop fs -cat /newDataFlair/copytest
Hello from DataFlair

Welcome to HDFS Command Tutuorial
dataflair@admin1-All-Series:~$ 
```

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

6. get

Hadoop HDFS get Command Usage:

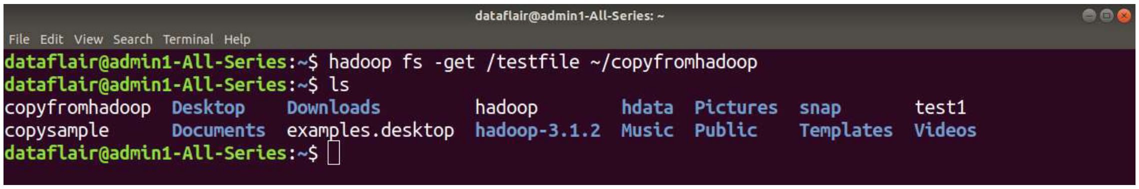
hadoop fs -get <src> <localdest>

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -get /testfile ~/copyfromhadoop
dataflair@admin1-All-Series:~$ ls
copyfromhadoop  Desktop    Downloads         hadoop        hdata  Pictures  snap       test1
copysample      Documents  examples.desktop  hadoop-3.1.2  Music  Public    Templates  Videos
dataflair@admin1-All-Series:~$
```

7. copyToLocal

Hadoop HDFS copyToLocal Command Usage:

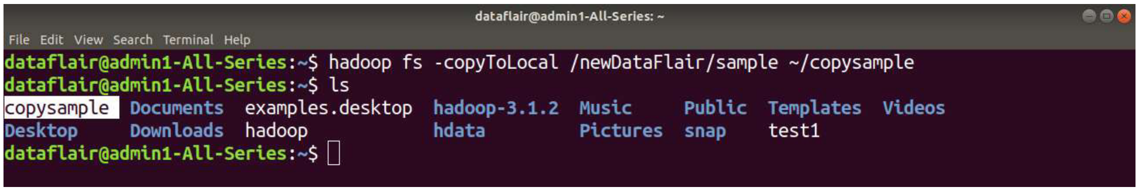
hadoop fs -copyToLocal <hdfs source> <localdst>

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -copyToLocal /newDataFlair/sample ~/copysample
dataflair@admin1-All-Series:~$ ls
copysample  Documents  examples.desktop  hadoop-3.1.2  Music     Public  Templates  Videos
Desktop     Downloads  hadoop            hdata         Pictures  snap    test1
dataflair@admin1-All-Series:~$
```

8. cat

Hadoop HDFS cat Command Usage:

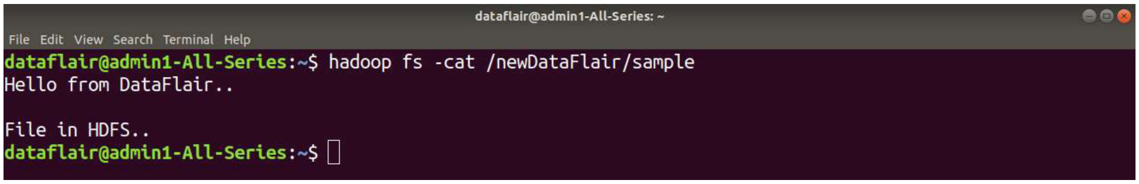
hadoop fs –cat /path_to_file_in_hdfs

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.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -cat /newDataFlair/sample
Hello from DataFlair..

File in HDFS..
dataflair@admin1-All-Series:~$ 
```

9. mv

Hadoop HDFS mv Command Usage:

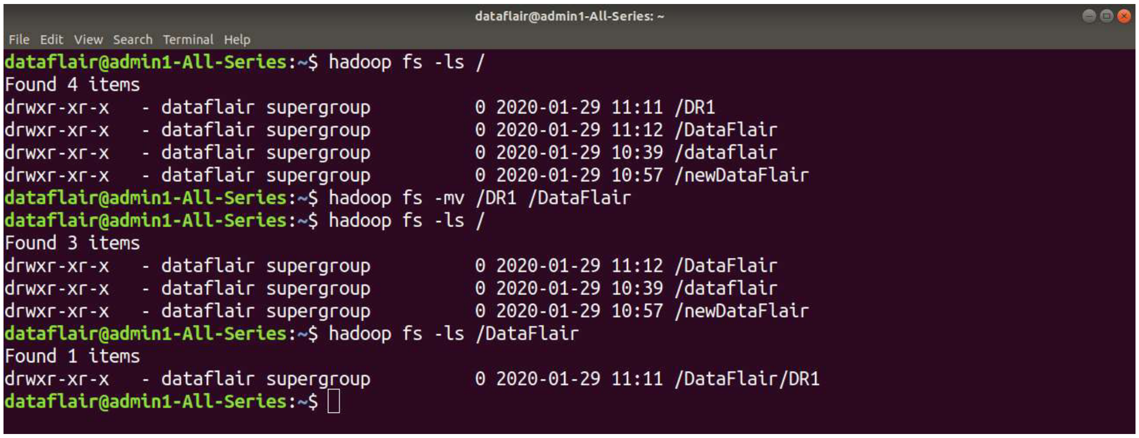
hadoop fs -mv <src> <dest>

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**.


dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -ls /
Found 4 items
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 11:11 /DR1
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 11:12 /DataFlair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:39 /dataflair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:57 /newDataFlair
dataflair@admin1-All-Series:~$ hadoop fs -mv /DR1 /DataFlair
dataflair@admin1-All-Series:~$ hadoop fs -ls /
Found 3 items
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 11:12 /DataFlair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:39 /dataflair
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 10:57 /newDataFlair
dataflair@admin1-All-Series:~$ hadoop fs -ls /DataFlair
Found 1 items
drwxr-xr-x   - dataflair supergroup          0 2020-01-29 11:11 /DataFlair/DR1
dataflair@admin1-All-Series:~$ 
```

10. cp

Hadoop HDFS cp Command Usage:

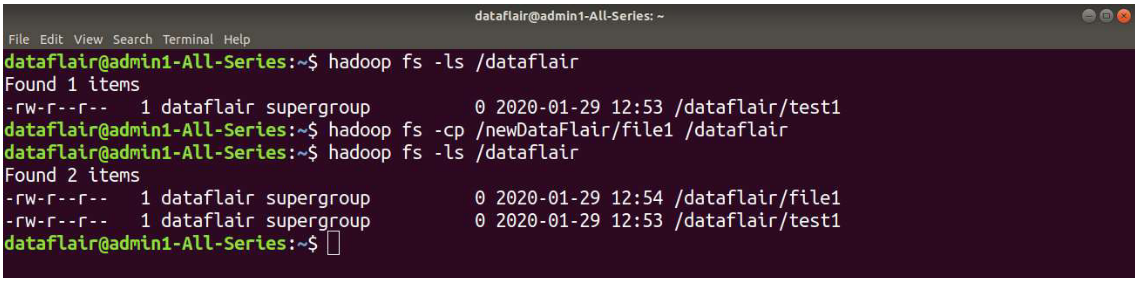
hadoop fs -cp <src> <dest>

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.



dataflair@admin1-All-Series: ~
File Edit View Search Terminal Help

```
dataflair@admin1-All-Series:~$ hadoop fs -ls /dataflair
Found 1 items
-rw-r--r--   1 dataflair supergroup          0 2020-01-29 12:53 /dataflair/test1
dataflair@admin1-All-Series:~$ hadoop fs -cp /newDataFlair/file1 /dataflair
dataflair@admin1-All-Series:~$ hadoop fs -ls /dataflair
Found 2 items
-rw-r--r--   1 dataflair supergroup          0 2020-01-29 12:54 /dataflair/file1
-rw-r--r--   1 dataflair supergroup          0 2020-01-29 12:53 /dataflair/test1
dataflair@admin1-All-Series:~$ 
```