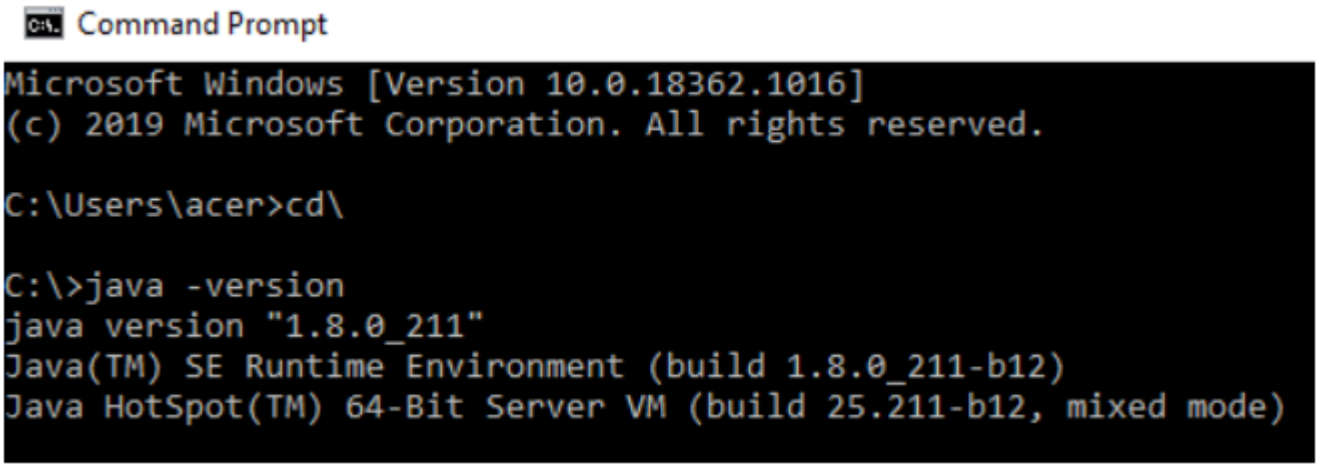
Practical-8

TASK : **Hadoop installation on Windows 10 (Single Node/Cluster)**

Prerequisite :

To install Hadoop you should have Java version 1.8 or greater in your system.



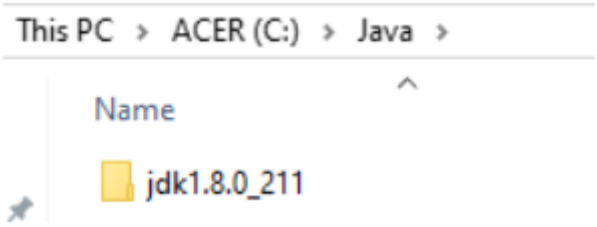
If java is not installed in your system, then go to the following link:

https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

Keep the java folder directly under the local disk directory

(C:\Java\jdk1.8.0_211) rather than in Program Files (C:\Program

Files\Java\jdk1.8.0_211) as it can create errors afterwards



Accept the license and Download the file according to your operating system.

Download Hadoop:

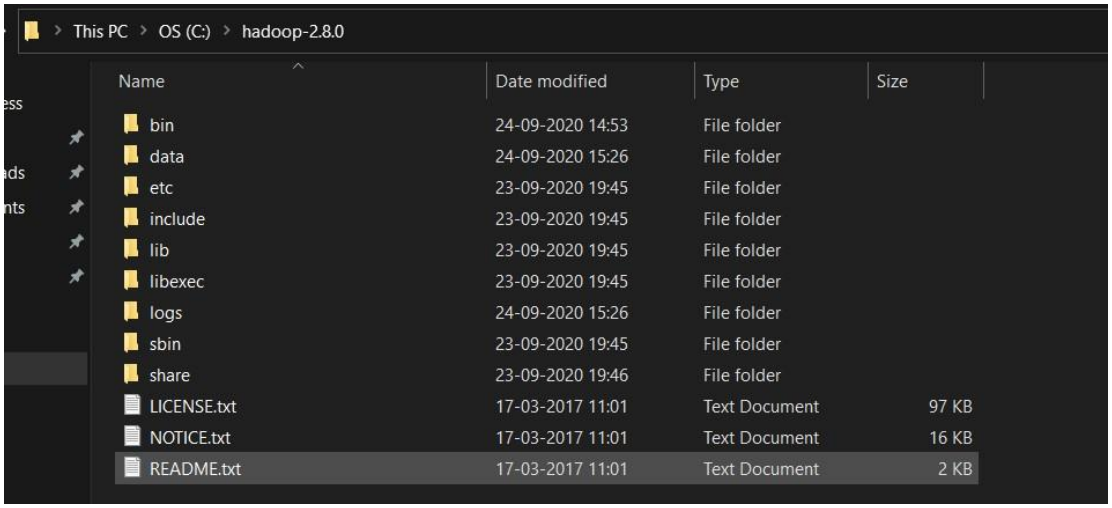
- After downloading java version 1.8, download Hadoop-2.8.0 from this link:

  http://archive.apache.org/dist/hadoop/core//hadoop-2.8.0/hadoop-2.8.0.tar.gz .

  OR

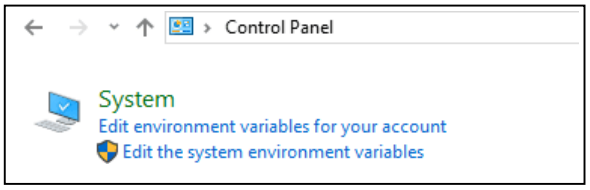
  http://archive.apache.org/dist/hadoop/core/hadoop-2.8.0/hadoop-2.8.0.tar.gz

Extract it to a folder.

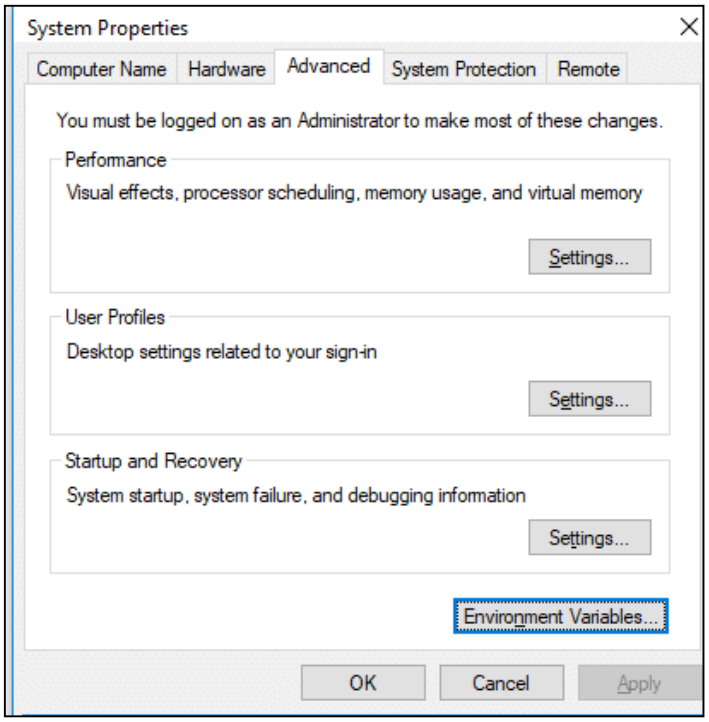


Setup System Environment Variables

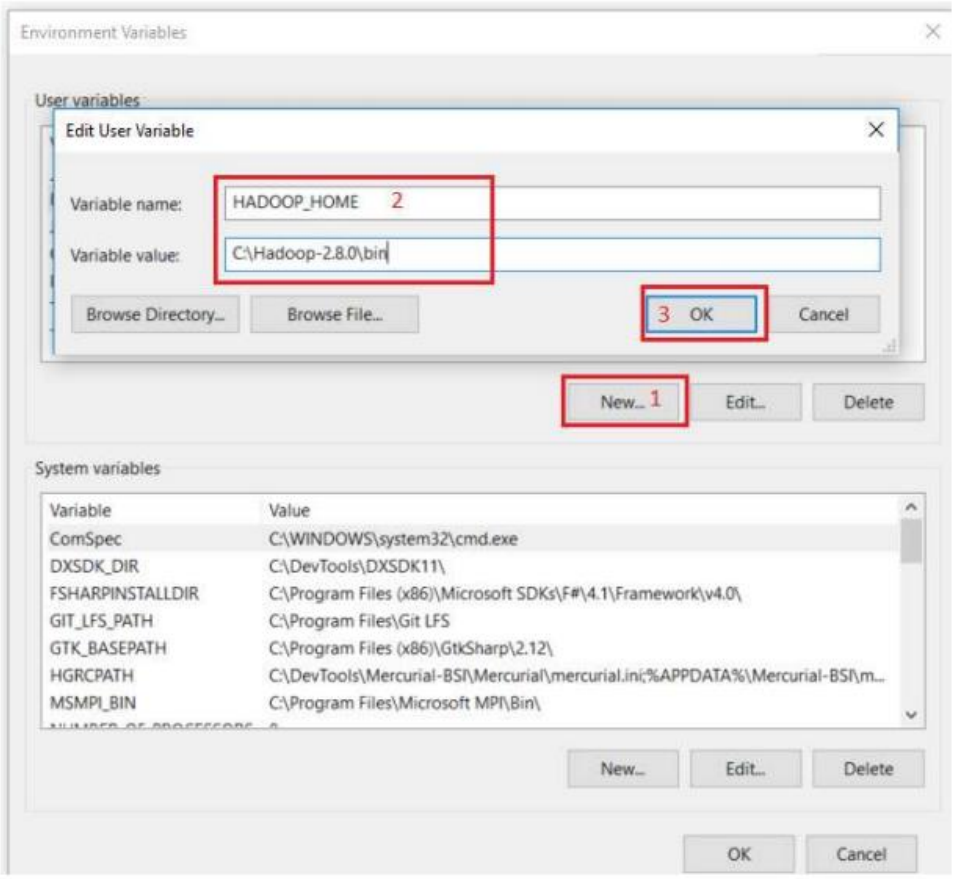
- Open control panel to edit the system environment variable.



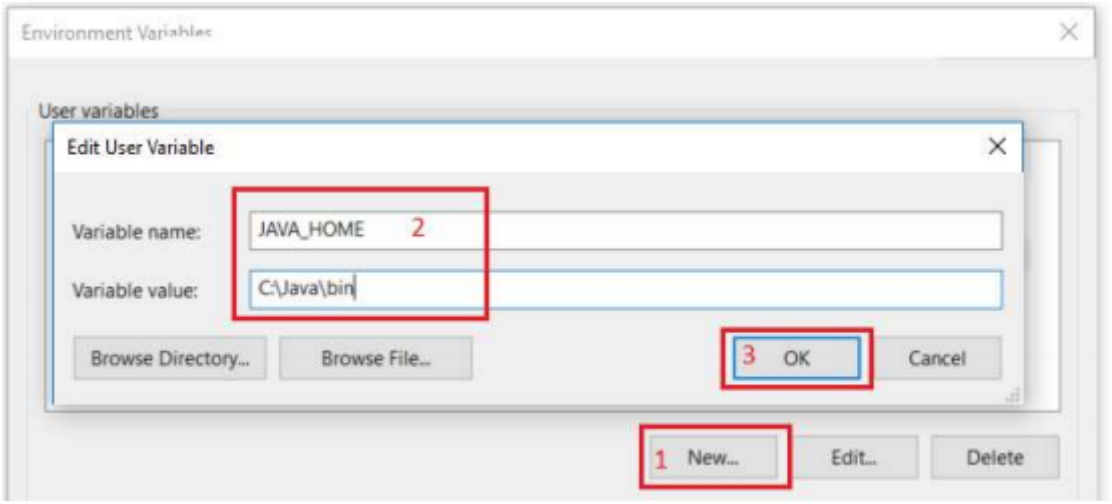
- Go to environment variable in system properties.



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. (Set the path HADOOP_HOME Environment variable on windows 10(see Step 1,2,3 and 4 below).



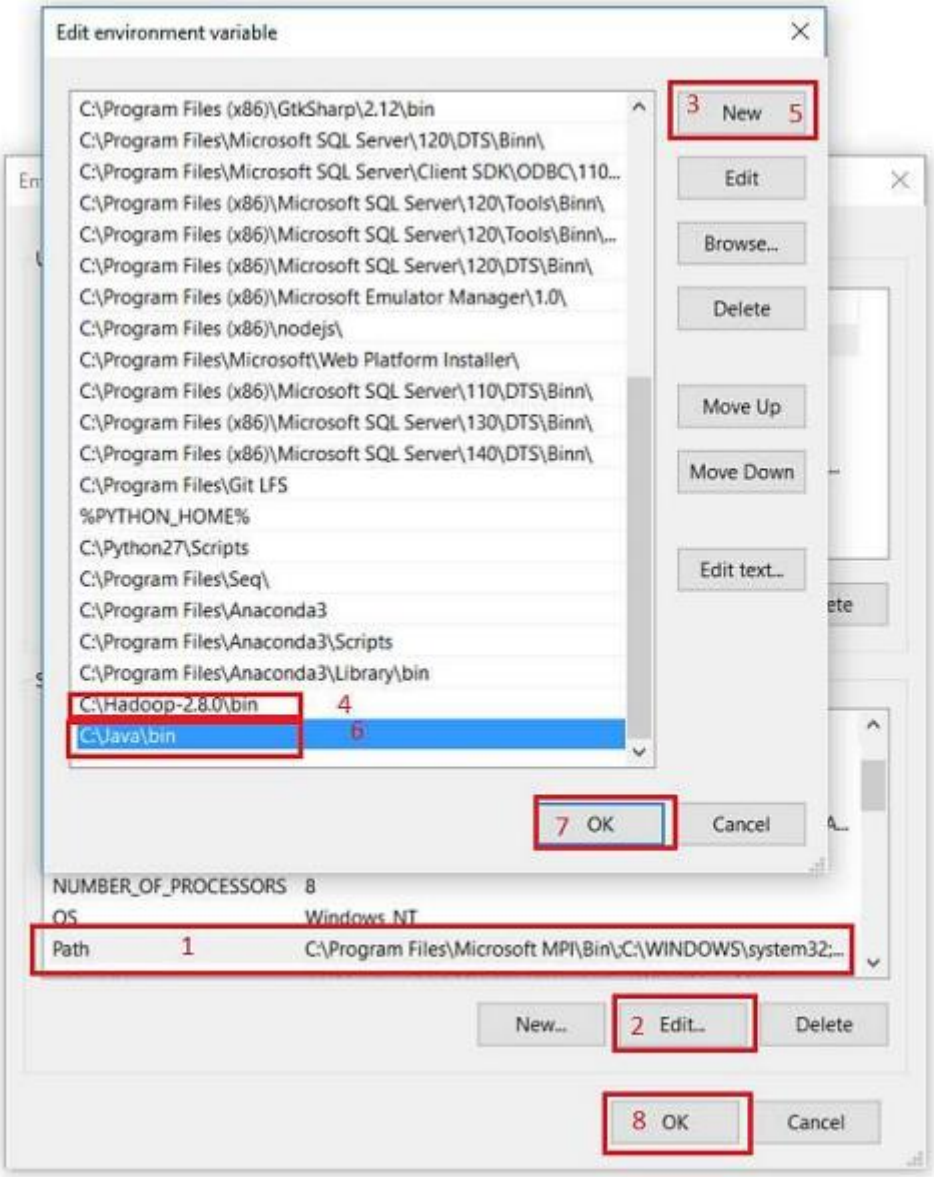
Likewise, create a new user variable with variable name as JAVA_HOME and variable value as the path of the bin folder in the Java directory. (Set the path JAVA_HOME Environment variable on windows 10(see Step 1, 2, 3 and 4).



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

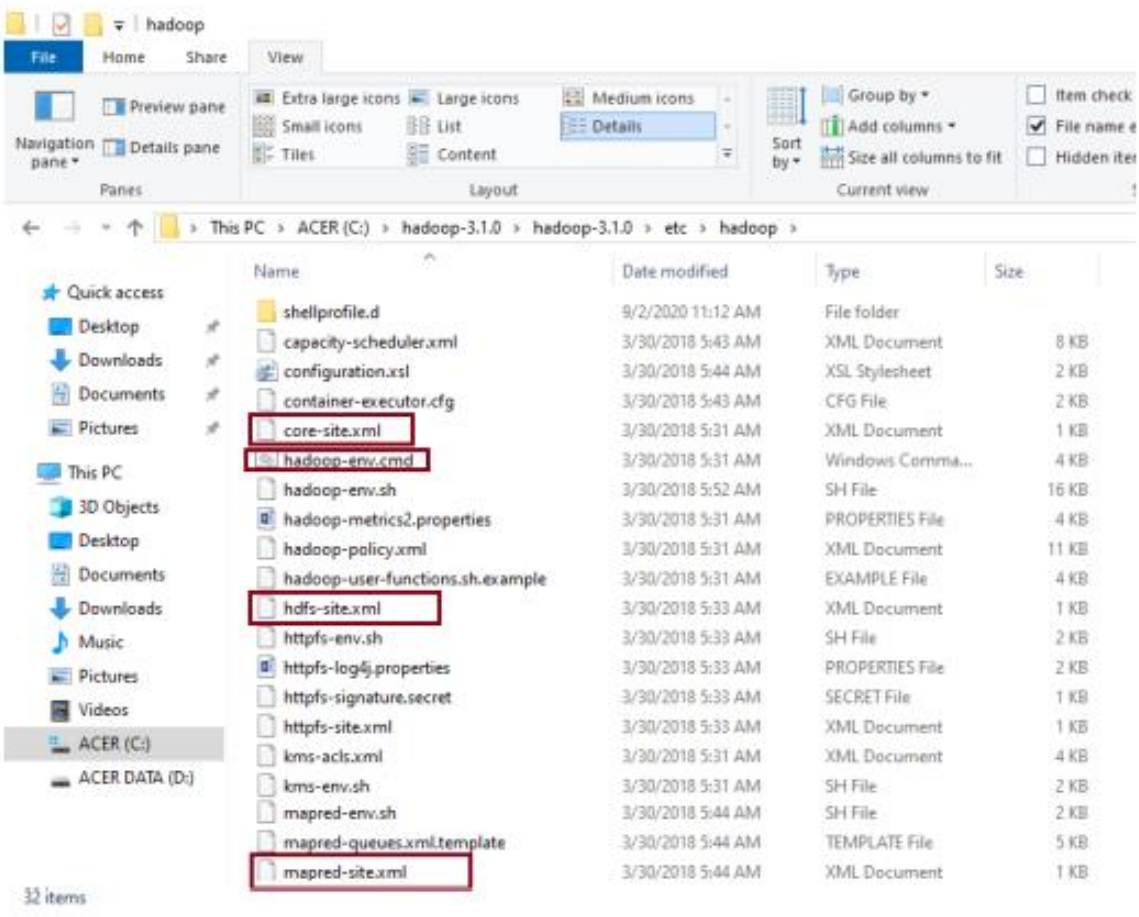
Edit Path in user variable.

(If you wish to configure multi-clusters node then add path in system variable)



Hadoop Configuration

- 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.



- Edit file C:\hadoop-2.8.0\etc\hadoop\core-site.xml, paste below xml paragraph and save this file4

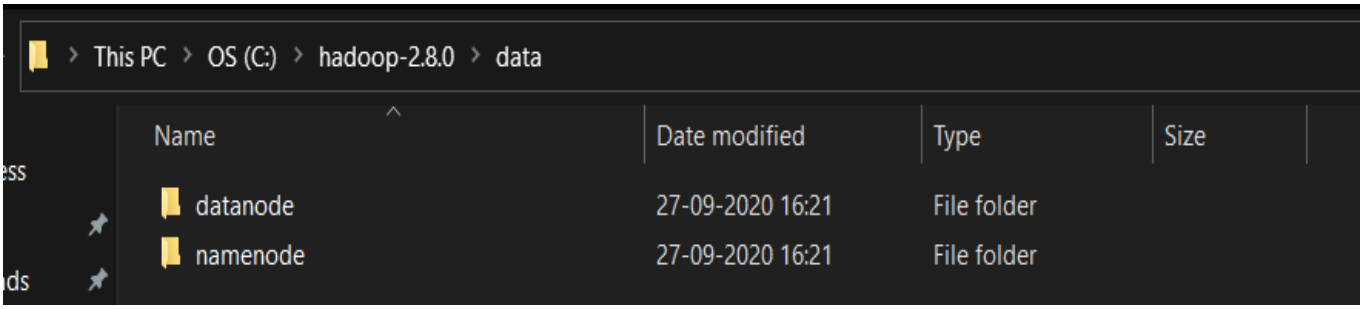
```
<configuration>
     <property>
          <name>fs.defaultFS</name>
          <value>hdfs://localhost:9000</value>
     </property>
</configuration>
```

- Rename “mapred-site.xml.template” to “mapred-site.xml” and edit this file **C:/Hadoop-2.8.0/etc/hadoop/mapred-site.xml**, paste below xml paragraph and save this file.

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

Create folder **"data"** under **"C:\Users\User\hadoop-2.8.0.tar\hadoop-2.8.0\data"**

- Create folder **"datanode"** under **"C:\Users\User\hadoop-2.8.0.tar\hadoop-2.8.0\data\datanode"**
- Create folder **"namenode"** under **"C:\Users\User\hadoop-2.8.0.tar\hadoop-2.8.0\data\namenode"**



- Edit file **C:\Users\User\hadoop-2.8.0.tar\hadoop-2.8.0\etc\hadoop\hdfs-site.xml**, paste below xml paragraph and save this file.

```
<configuration>
     <property>
          <name>dfs.replication</name>
          <value>1</value>
     </property>
     <property>
          <name>dfs.namenode.name.dir</name>
          <value>/hadoop-2.8.0/data/namenode</value>
     </property>
     <property>
          <name>dfs.namenode.data.dir</name>
          <value>/hadoop-2.8.0/data/datanode</value>
     </property>
</configuration>
```

- Edit file **C:\Users\User\hadoop-2.8.0.tar\hadoop-2.8.0\etc\hadoop\yarn-site.xml**, paste below xml paragraph and save this file.

```
<configuration>
     <property>
          <name>yarn.nodemanager.aux-services</name>
          <value>mapreduce_shuffle</value>
     </property>
     <property>
     <name>yarn.nodemanager.auxservices.mapreduce.shuffle
.class</name>

     <value>org.apache.hadoop.mapred.ShuffleHandler</valu
e>
     </property>
</configuration>
```

- Edit file **C:/Hadoop-2.8.0/etc/Hadoop/Hadoop-env.** cmd by closing the command line **"JAVA_HOME=%JAVA_HOME%"** instead of set **"JAVA_HOME=C:\Java\jdk14.0.2"**

```
@echo off
@rem Licensed to the Apache Software Foundation (ASF) under one or more
@rem contributor license agreements.  See the NOTICE file distributed w
@rem this work for additional information regarding copyright ownership
@rem The ASF licenses this file to You under the Apache License, Versio
@rem (the "License"); you may not use this file except in compliance wi
@rem the License.  You may obtain a copy of the License at
@rem
@rem     http://www.apache.org/licenses/LICENSE-2.0
@rem
@rem Unless required by applicable law or agreed to in writing, softwar
@rem distributed under the License is distributed on an "AS IS" BASIS,
@rem WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or im
@rem See the License for the specific language governing permissions an
@rem limitations under the License.

@rem Set Hadoop-specific environment variables here.

@rem The only required environment variable is JAVA_HOME.  All others a
@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\jdk-14.0.2

```

- Hadoop needs windows OS specific files which does not come with default download of Hadoop. To include this files, replace the bin folder in Hadoop directory with the bin provided in this link.

- Download file: https://github.com/MuhammadBilalYar/Hadoop-On-Window/blob/master/Hadoop%20Configuration.zip

- Delete file bin on C:\Hadoop-2.8.0\bin, replaced by file on file just downloaded (from Hadoop Configuration.zip). Check whether Hadoop is successfully installed by running this command on cmd

C:\>hadoop version

```
C:\>hadoop version
Hadoop 2.8.0
Subversion https://git-wip-us.apache.org/repos/asf/hadoop.git -r 91f2b7a13d1e97be65db92ddabc627cc29ac0009
Compiled by jdu on 2017-03-17T04:12Z
Compiled with protoc 2.5.0
From source with checksum 60125541c2b3e266cbf3becc5bda666
This command was run using /C:/hadoop-2.8.0/hadoop-2.8.0/share/hadoop/common/hadoop-common-2.8.0.jar
```

- Since it doesn't throw error and successfully shows the Hadoop version means Hadoop is successfully installed in the system.
- Open cmd and type command **"hdfs namenode -format".** You will see .

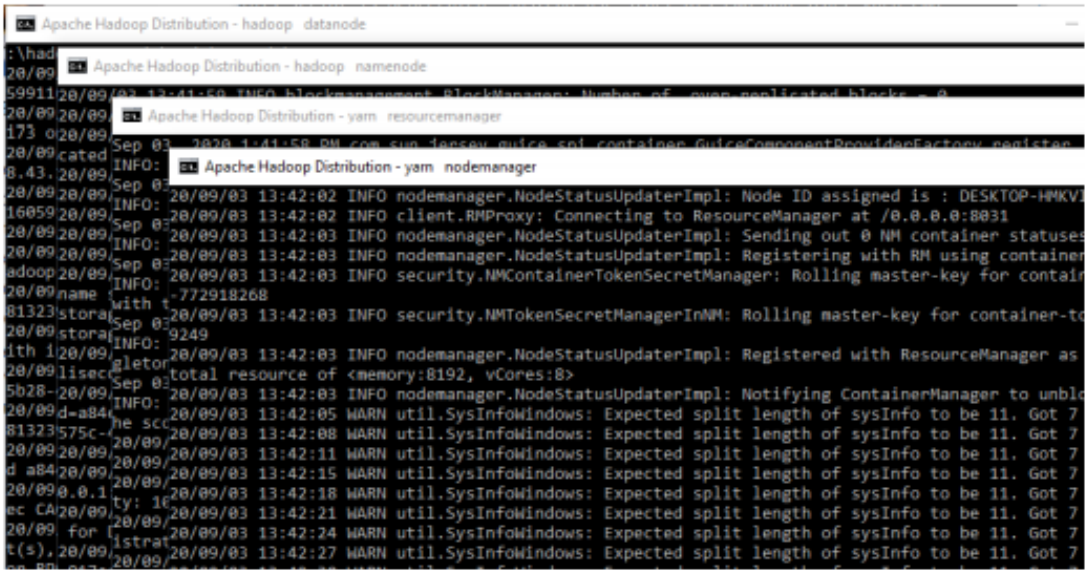
```
C:\>hdfs namenode -format
20/09/24 15:19:10 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   user = Asus
STARTUP_MSG:   host = LAPTOP-A5PKB0A0/192.168.42.10
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.8.0
STARTUP_MSG:   classpath = C:\hadoop-2.8.0\hadoop-2.8.0\etc\hadoop;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\com
tivation-1.1.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\apacheds-i18n-2.0.0-M15.jar;C:\hadoop-2.8.
.8.0\share\hadoop\common\lib\apacheds-kerberos-codec-2.0.0-M15.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\com
i-asn1-api-1.0.0-M20.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\api-util-1.0.0-M20.jar;C:\hadoop-2
p-2.8.0\share\hadoop\common\lib\asm-3.2.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\avro-1.7.4.jar;
2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-beanutils-1.7.0.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop
b\commons-beanutils-core-1.8.0.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-cli-1.2.jar;C:\h
0\hadoop-2.8.0\share\hadoop\common\lib\commons-codec-1.4.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\li
collections-3.2.2.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-compress-1.4.1.jar;C:\hadoop-
op-2.8.0\share\hadoop\common\lib\commons-configuration-1.6.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\
s-digester-1.8.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-io-2.4.jar;C:\hadoop-2.8.0\hadoo
are\hadoop\common\lib\commons-lang-2.6.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-logging-
C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\commons-math3-3.1.1.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\
mon\lib\commons-net-3.1.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\curator-client-2.7.1.jar;C:\had
hadoop-2.8.0\share\hadoop\common\lib\curator-framework-2.7.1.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\commo
tor-recipes-2.7.1.jar;C:\hadoop-2.8.0\hadoop-2.8.0\share\hadoop\common\lib\gson-2.2.4.jar;C:\hadoop-2.8.0\hadoop
```

```
20/09/24 15:19:11 INFO util.GSet: Computing capacity for map cachedBlocks
20/09/24 15:19:11 INFO util.GSet: VM type       = 64-bit
20/09/24 15:19:11 INFO util.GSet: 0.25% max memory 1000 MB = 2.5 MB
20/09/24 15:19:11 INFO util.GSet: capacity      = 2^18 = 262144 entries
20/09/24 15:19:11 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.99900
20/09/24 15:19:11 INFO namenode.FSNamesystem: dfs.namenode.safemode.min.datanodes = 0
20/09/24 15:19:11 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension     = 30000
20/09/24 15:19:11 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets
20/09/24 15:19:11 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
20/09/24 15:19:11 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1
20/09/24 15:19:11 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
20/09/24 15:19:11 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and r
 600000 millis
20/09/24 15:19:11 INFO util.GSet: Computing capacity for map NameNodeRetryCache
20/09/24 15:19:11 INFO util.GSet: VM type       = 64-bit
20/09/24 15:19:11 INFO util.GSet: 0.029999999329447746% max memory 1000 MB = 307.2 KB
20/09/24 15:19:11 INFO util.GSet: capacity      = 2^15 = 32768 entries
20/09/24 15:19:11 INFO namenode.FSImage: Allocated new BlockPoolId: BP-480738929-192.168.42
20/09/24 15:19:11 INFO common.Storage: Storage directory C:\hadoop-2.8.0\data\namenode has
20/09/24 15:19:11 INFO namenode.FSImageFormatProtobuf: Saving image file C:\hadoop-2.8.0\da
t_000000000000000000 using no compression
20/09/24 15:19:11 INFO namenode.FSImageFormatProtobuf: Image file C:\hadoop-2.8.0\data\name
00000000000000 of size 321 bytes saved in 0 seconds.
20/09/24 15:19:11 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with tx
20/09/24 15:19:11 INFO util.ExitUtil: Exiting with status 0
20/09/24 15:19:11 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at LAPTOP-A5PKB0A0/192.168.42.10
************************************************************/

C:\>
```

Testing

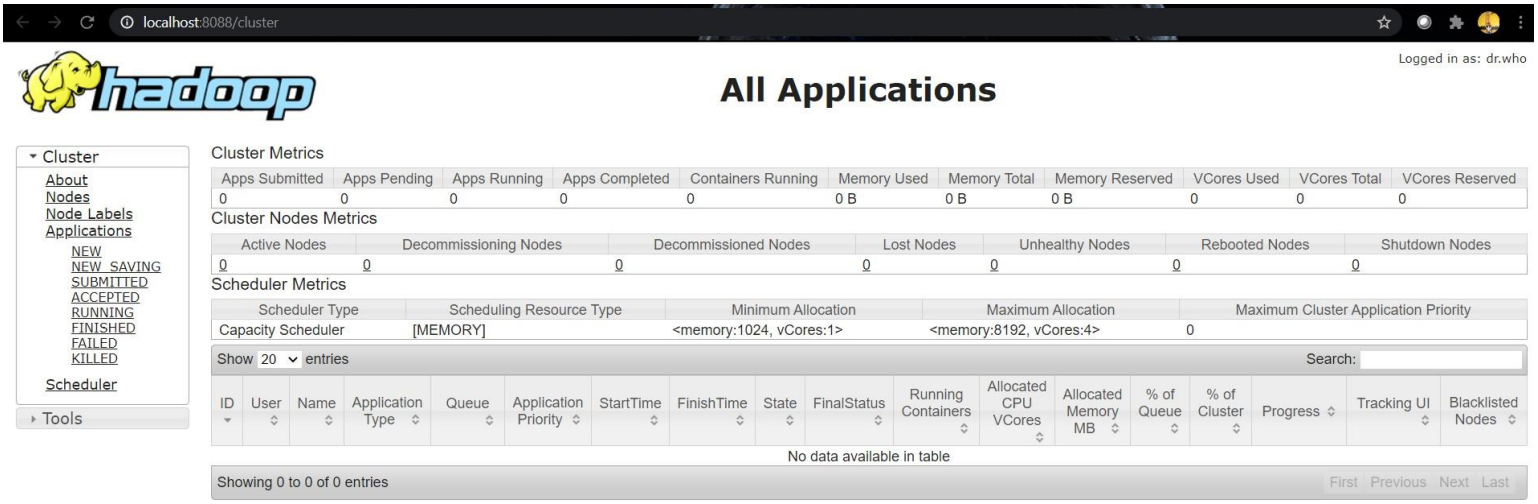
Open cmd and change directory to **"C:\Hadoop-2.8.0\sbin"** and type **"start-all.cmd"** to start apache.



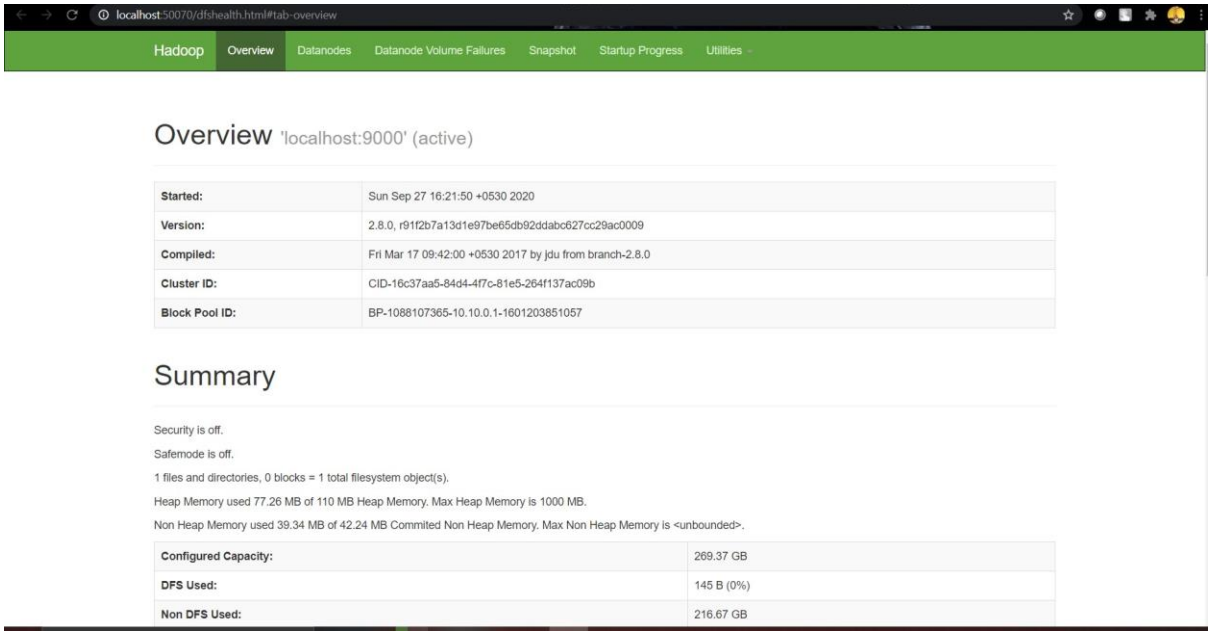
Make sure these apps are running

- Hadoop Namenode
- Hadoop Datanode
- YARN Resource Manager
- YARN Node Manager

Open: http://localhost:8080



Open: http://localhost:50070



Congratulations, Hadoop installed.

Working with HDFS

- Open command prompt and change the directory **"C:\hadoop-2.8.0\sbin".**
- Create a directory named "**sample**" in my Hadoop directory using the following command.
  hdfs dfs –mkdir /sample
- To verify if the directory is created in hdfs, we will use '**ls'** command which will list the files present in hdfs-
  hdfs dfs –ls /
- Copy a text file named "**hello.txt**" from my local file system to this folder that we have just created in hdfs using copyFromLocal command.

  hdfs dfs –copyFromLocal C:\demo.txt /sample
- To verify if the file is copied to the folder, we will use 'ls' command by specifying the folder name which will read the list of files in that folder
  hdfs dfs –ls /sample

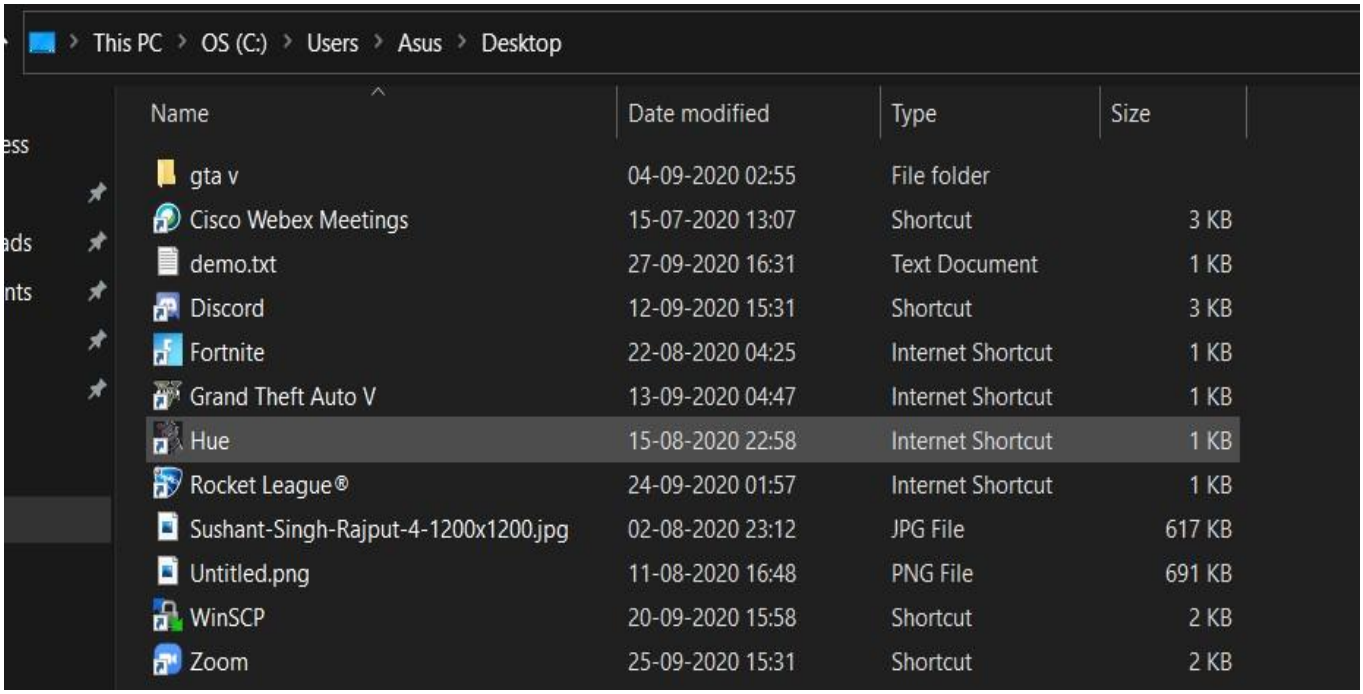
```
C:\hadoop-2.8.0\sbin>hdfs dfs -mkdir /sample
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/C:/hadoop-2.8.0
/share/hadoop/common/lib/hadoop-auth-2.8.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUti
l
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
```

```
C:\hadoop-2.8.0\sbin>hdfs dfs -ls /
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/C:/hadoop-2.8.0
/share/hadoop/common/lib/hadoop-auth-2.8.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUti
l
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Found 1 items
drwxr-xr-x   - Asus supergroup          0 2020-09-27 16:24 /sample
```

```
C:\hadoop-2.8.0\sbin>hdfs dfs -cat /sample/demo.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/C:/hadoop-2.8.0
/share/hadoop/common/lib/hadoop-auth-2.8.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUti
l
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
World's no 1
 level 3 vest
   top pro
     conquerer
    all-asia India
          no 1
       gaming skills
```

- To view the contents of the file we copied, we will use cat command
  hdfs dfs –cat /sample/demo.txt

- To copy file from hdfs to local directory, we will use get command
  hdfs dfs –get /sample/demo.txt C:\Users\Asus\Desktop



- If you wish to stop Hadoop then command
  Stop-all.cmd

```
C:\hadoop-2.8.0\sbin>Stop-all.cmd
This script is Deprecated. Instead use stop-dfs.cmd and stop-yarn.cmd
SUCCESS: Sent termination signal to the process with PID 5880.
SUCCESS: Sent termination signal to the process with PID 12984.
stopping yarn daemons
SUCCESS: Sent termination signal to the process with PID 8716.
SUCCESS: Sent termination signal to the process with PID 7564.

INFO: No tasks running with the specified criteria.

C:\hadoop-2.8.0\sbin>
```

TASK :- Perform MapReduce Program in CloudX-Lab.

- Mappers and Reducers in Hadoop are written to get their inputs from stdin, and output their tuples to stdout. (Word Count Program)

Mapper.py:

```
#!/usr/bin/env python
import sys
for line in sys.stdin:
 # remove leading and trailing whitespace
   line = line.strip()
   # split the line into words
   words = line.split()
   # increase counters
   for word in words:
      # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e. the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        print '%s\t%s' % (word, 1)
```

Reducer.py:

```
#!/usr/bin/env python
from operator import itemgetter
import sys
current_word = None
current_count = 0
word = None
# input comes from STDIN
for line in sys.stdin:
   # remove leading and trailing whitespace
   line = line.strip()
   # parse the input we got from mapper.py
   word, count = line.split('\t', 1)
```

```
    # Convert count (currently a string) to int
    try:
        count = int(count)
    except ValueError:
        # count was not a number, so silently
        # ignore/discard this line
        continue
    # this IF-switch only works because Hadoop sorts map output
    # by key (here: word) before it is passed to the reducer
    if current_word == word:
        current_count += count
    else:
        if current_word:
            # write result to STDOUT
            print '%s\t%s' % (current_word, current_count)
        current_count = count
        current_word = word
# do not forget to output the last word if needed!
if current_word == word:
        print '%s\t%s' % (current_word, current_count)
```

word.txt:

Cat mouse lion deer Tiger lion Elephant lion deer

STEPS :

1. Copy the manner.py and reducer.py in local webconsole(cloudXLab) folder .

```
[surejadharmay4494@cxln5 ~]$ ls -l mapreduce/
total 12
-rw-rw-r-- 1 surejadharmay4494 surejadharmay4494 481 Nov 12 18:57 mapper.py
-rw-rw-r-- 1 surejadharmay4494 surejadharmay4494 982 Nov 12 18:57 reducer.py
-rw-r--r-- 1 surejadharmay4494 surejadharmay4494  50 Nov 12 19:02 word.txt
[surejadharmay4494@cxln5 ~]$
```

2. Change the permission of both files using :

```
[surejadharmay4494@cxln5 ~]$ chmod a+x mapreduce/reducer.py
[surejadharmay4494@cxln5 ~]$ chmod a+x mapreduce/mapper.py
[surejadharmay4494@cxln5 ~]$
```

3. Perform following command to run map reducer program .

First :

```
[surejadharmay4494@cxln5 mapreduce]$ cat word.txt | python mapper.py
cat     1
mouse   1
lion    1
deer    1
Tiger   1
lion    1
Elephant        1
lion    1
deer    1
[surejadharmay4494@cxln5 mapreduce]$
```

Second :

```
[surejadharmay4494@cxln5 mapreduce]$ cat word.txt | python mapper.py |sort
cat     1
deer    1
deer    1
Elephant        1
lion    1
lion    1
lion    1
mouse   1
Tiger   1
[surejadharmay4494@cxln5 mapreduce]$
```

Third :

```
[surejadharmay4494@cxln5 mapreduce]$ cat word.txt | python mapper.py |sort | python reduce
r.py
cat     1
deer    2
Elephant        1
lion    3
mouse   1
Tiger   1
[surejadharmay4494@cxln5 mapreduce]$
```