Batch-19(IT-A)……
Installing Hadoop for Windows

Choosing a Hadoop distribution to install

In this chapter I will do an installation of Apache Hadoop for Windows. This is a version of Hadoop that is totally free, and is the basis of all Hadoop distributions. To test whether Hadoop can be installed on pretty much any Windows PC, I will do a local installation on Windows 8.
While the screenshots I show are from Windows 8, the process on Windows 10 and Windows Server is similar, and you shouldn't have any trouble making the necessary adjustments. It’s important that you don’t feel you need the latest shiny, new PC to run Hadoop, though later when we look at multi-node installations, we will use multiple Windows Server 2016 machines. The PC in this exercise meets the requirements shown in Table 1, with 8 GB of RAM, a Quad Core 2.4 GHz AMD processor, and solid-state drives.

Apache Hadoop installation prerequisites

- **JAVA 8 or later**: You can download the 64-bit Windows .jdk file **jdk-8u381-windows-x64.exe** from https://www.oracle.com/java/technologies/downloads/#java8-windows. It is important to state why a prerequisite is required, so the nature of the dependency on the prerequisite is understood. Hadoop is a Java-based application that creates various dependencies on Java. For example, in a single-node Hadoop installation, there is a single Java process running all Hadoop functions. Without Java, all those functions would be unavailable. It is essential to have the right version and architecture of Java, and a 64-bit JDK higher than 1.6 should always be chosen to install Hadoop for Windows.

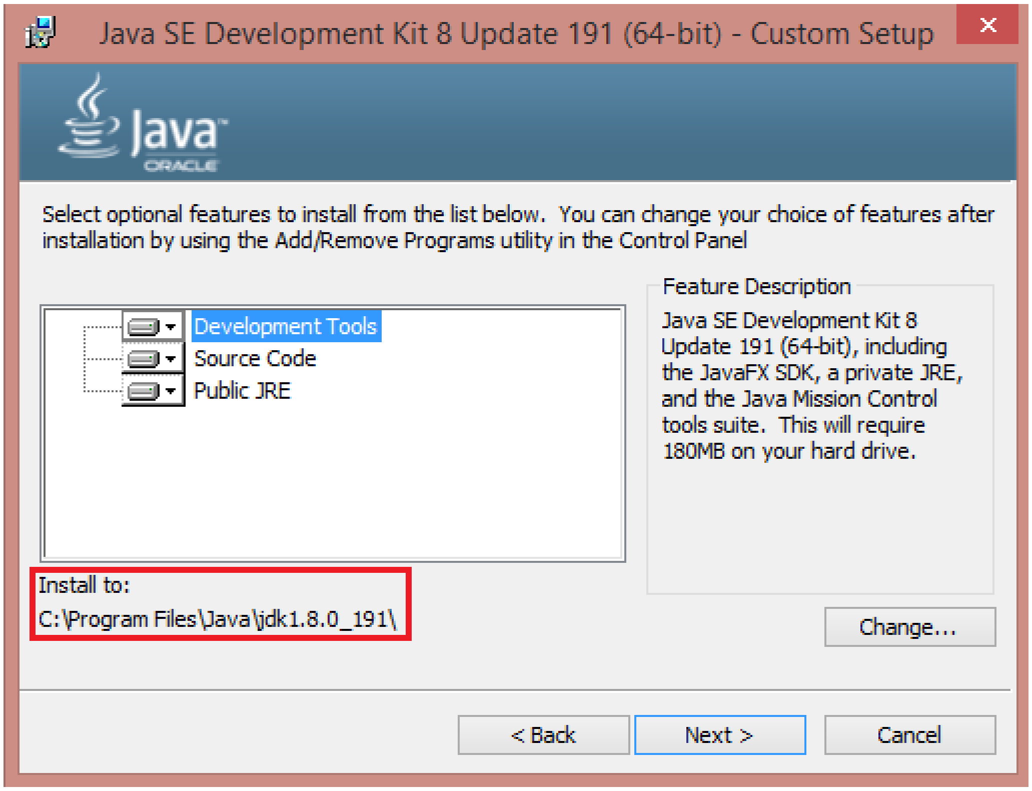
- **Hadoop 2.0 or later**: You can download the Hadoop binary file **hadoop-3.2.4.tar.gz** from https://hadoop.apache.org/releases.html.

- **Microsoft Windows**: Windows 7, 8, 10, and Windows Server 2008 and above.

- **Additional prerequisites**: You’ll also need a text editor, such as Notepad or Notepad ++, for writing short amounts of code, and Winutils 3.1, which you can download from https://github.com/killerangaswamy56/hadoop_config.git

Java installation for Hadoop for Windows

Run the downloaded Java installer, following the onscreen instructions to complete the installation. Ensure that you right-click on the Java installation file and choose **Run as administrator** from the menu. You will see a User Account Control message asking you to allow the application to make changes to your computer, to which you answer **Yes**. Follow the onscreen prompts to install Java, including the following screen, where you can accept the default installation path.


Java SE Development Kit 8 Update 191 (64-bit) - Custom Setup

Select optional features to install from the list below. You can change your choice of features after installation by using the Add/Remove Programs utility in the Control Panel

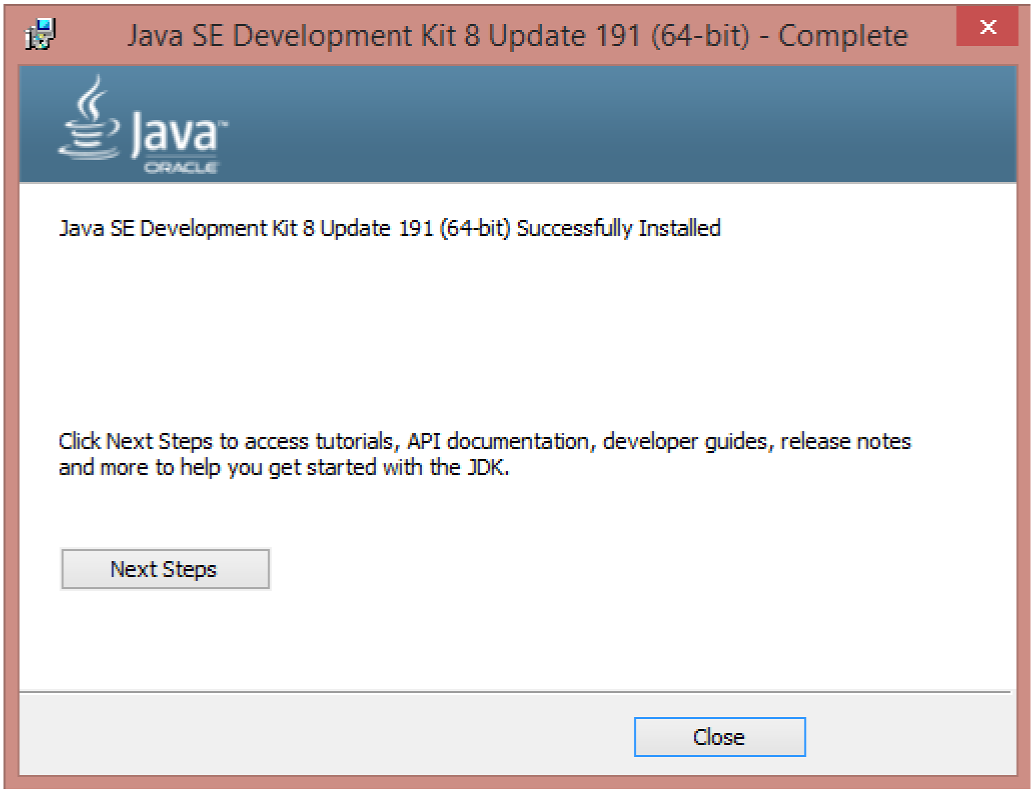
- Development Tools
- Source Code
- Public JRE

Feature Description

Java SE Development Kit 8 Update 191 (64-bit), including the JavaFX SDK, a private JRE, and the Java Mission Control tools suite. This will require 180MB on your hard drive.

Install to:
C:\Program Files\Java\jdk1.8.0_191\

Figure 9: Default Java installation path


Java SE Development Kit 8 Update 191 (64-bit) - Complete

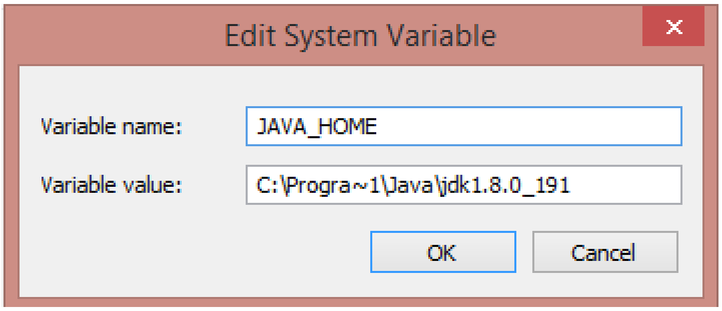
Java SE Development Kit 8 Update 191 (64-bit) Successfully Installed

Click Next Steps to access tutorials, API documentation, developer guides, release notes and more to help you get started with the JDK.

Figure 10: Successful Java installation

Ensure that you see the screen informing you that Java has been successfully installed.

Go to **Control Panel** > **System and Security** > **System** and click **Advanced System Settings**, and then click the **System Environment Variables** button. Whether creating a new environment variable for **JAVA_HOME** or editing an existing one, you must alter the Program Files text to text that Hadoop can interpret. On Windows 8, to create a Hadoop-compatible **JAVA_HOME** file instead of entering Program Files, insert **Progra~1** when entering the Java location in the **Variable value** field. On Windows 10 and Windows Server, avoid folder names with blank spaces.

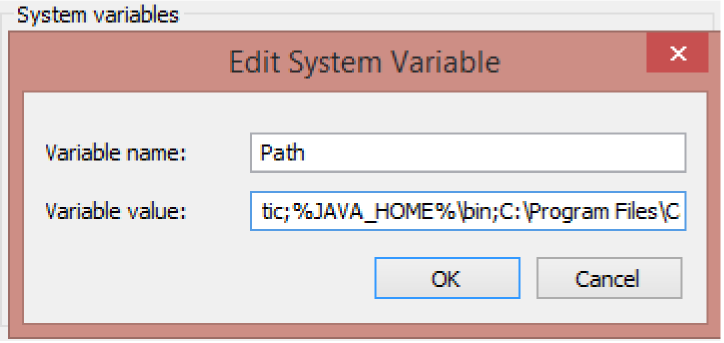


Variable name: JAVA_HOME

Variable value: C:\Progra~1\Java\jdk1.8.0_191

Figure 11: Hadoop compatible Java Home

Please ensure that you add the **JAVA_HOME** to the **Path** variable in System Variables. In this instance, it is done by adding **%JAVA_HOME%\bin** between semi colons in the Path **Variable value** field. Use the **java -version** command from a command prompt to ensure that Java is installed and running correctly.



Variable name: Path

Variable value: tic;%JAVA_HOME%\bin;C:\Program Files\C

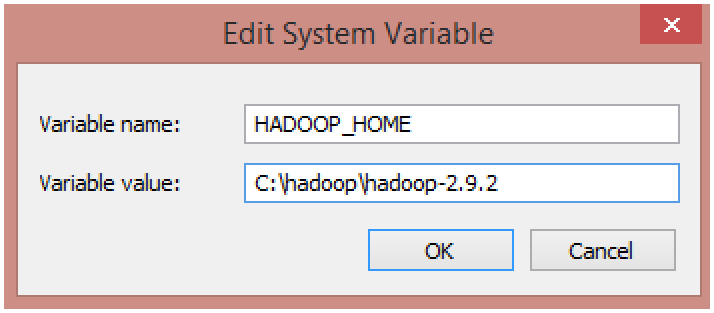
Figure 12: Adding Java Home to the Path Variable

Apache Hadoop installation

1. Create a folder called **C:\hadoop** on your hard drive.

2. Using an application such as 7-Zip File Manager, extract the Hadoop binary file **hadoop-3.2.4.tar.gz** from the Apche hadoop website to a directory of your choice, or directly to **C:\hadoop\hadoop-3.2.4**. If you choose to extract the files to a directory of your choice, then you first have to copy the extracted files to **C:\hadoop**. You may find it more convenient to extract them directly to **C:\hadoop**, which will then have an extracted folder in it called hadoop-3.2.4, so you end up with the C:\hadoop\hadoop-3.2.4 folders.

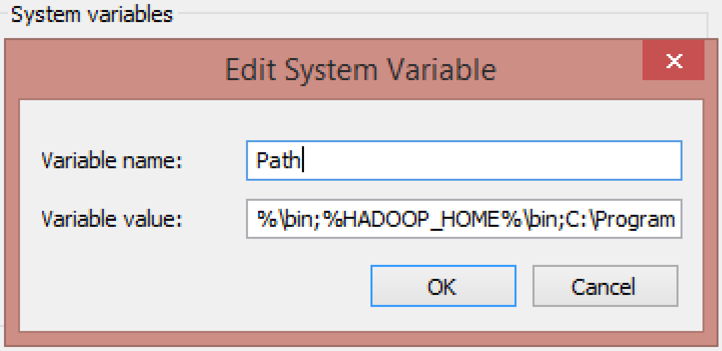
3. You can now create a **HADOOP_HOME** similar to how we created one previously, by going back to **Control Panel** > **System and Security** > **System**, clicking **Advanced System Settings**, and then clicking the **Environment Variables** button. Create the Hadoop home by adding the system variable name **HADOOP_HOME**, with the system variable value being the folder that we extracted the Hadoop binary to, which was **C:\hadoop\hadoop-3.2.4**.



Variable name: HADOOP_HOME

Variable value: C:\hadoop\hadoop-2.9.2

Figure 13: Creating a Hadoop Home



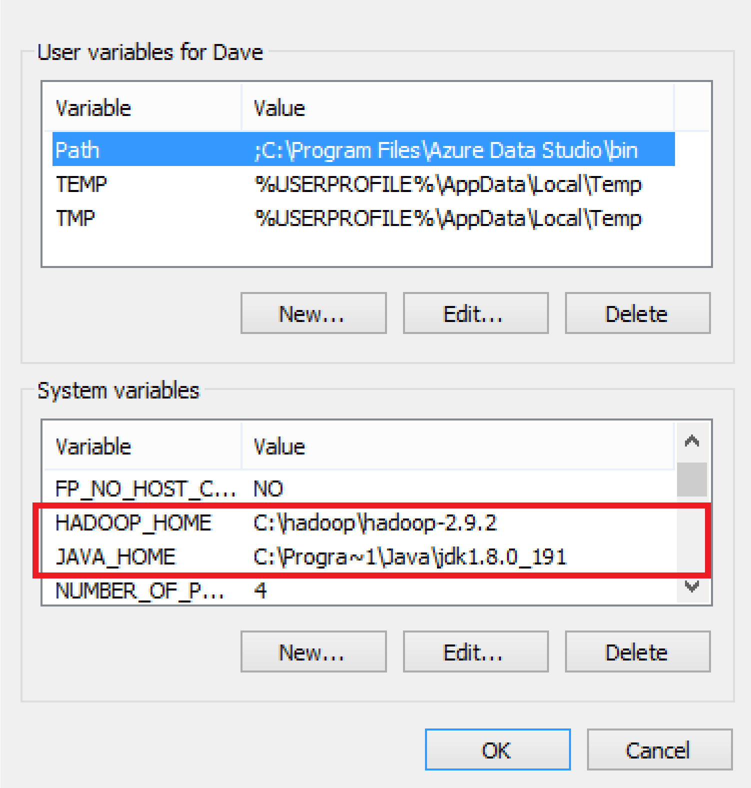
Variable name: Path

Variable value: %\bin;%HADOOP_HOME%\bin;C:\Program

Figure 14: Adding Hadoop Home to the Path variable

We must add the **HADOOP_HOME** file to the **Path** variable in System variables. In this instance, it is done by adding **%HADOOP_HOME%\bin** between semi colons in the **Variable value** field.

In addition, we must add a second **HADOOP_HOME** to the **Path** variable for the folder in Hadoop called **sbin**. This is done by adding **%HADOOP_HOME%\sbin** between semi colons in the **Variable value** field. You should now have Hadoop and Java homes, and two Hadoop path variables.



User variables for Dave

Variable	Value
Path	;C:\Program Files\Azure Data Studio\bin
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

System variables

Variable	Value
FP_NO_HOST_C...	NO
HADOOP_HOME	C:\hadoop\hadoop-2.9.2
JAVA_HOME	C:\Progra~1\Java\jdk1.8.0_191
NUMBER_OF_P...	4

Figure 15: Java and Hadoop homes

The resource page I mentioned previously is an official Apache resource that will assist us in finishing the installation. The area of the site we need first is “Section 3.1. Example HDFS Configuration,” which states:

> “Before you can start the Hadoop Daemons you will need to make a few edits to configuration files. The configuration file templates will all be found in c:\deploy\etc\hadoop, assuming your installation directory is c:\deploy.”

Since our installation is at C:\hadoop\hadoop-3.2.4, our configuration file templates will be located at C:\hadoop\hadoop-3.2.4\etc\hadoop\. The first file we need to edit is the **core-site.xml** file. The following code listing shows the format of the core-site.xml file, which is the style that we need to adapt. You will need your code editor at this point (I am using Notepad++).

Code Listing 1: The core-site.xml file format

```xml
<configuration>
<property>
<name>fs.default.name</name>
<value>hdfs://0.0.0.0:19000</value>
</property>
</configuration>
```

We need to substitute the name and value elements shown on the core-site.xml file on the Apache Wiki page for values in the installation we are carrying out. The values we require are contained in the following code listing and reflect our current Hadoop installation.

Code Listing 2: Editing the core-site.xml file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl"href="configuration.xsl"?>

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

We need to do the same for the **hdfs-site.xml** file template, and the new values we require are in the following code listing.

Code Listing 3: Editing the hdfs-site.xml template

```xml
<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:///C:/Hadoop/hadoop-3.2.4/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:///C:/Hadoop/hadoop-3.2.4/datanode</value>
</property>

</configuration>
```

Note: You must create two folders in the C:\Hadoop\hadoop-3.2.4\ folder in Windows Explorer to reflect the `namenode` and `datanode` directories mentioned in

25

Code Listing 3. Note that the Hadoop configuration files use forward slashes instead of backward slashes in file paths, even on Windows systems.

Next, we need to edit the **mapred-site.xml** configuration file**;** the values required are shown in the following code listing.

Code Listing 4: Editing the mapred-site.xml configuration file

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

</configuration>
```

We also need to edit the **yarn-site.xml** configuration file; the values required are provided in the following code listing.

Code Listing 5: Editing the yarn-site.xml configuration 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</value>
</property>

</configuration>
```

* *Now open etc/Hadoop.env file as shown below.*

```
set JAVA_HOME=C:\Java\jdk-1.8
```

Next, follow these steps:

1. Replace the bin folder at **C:\hadoop\hadoop-3.2.4\bin** with a bin folder extracted from https://github.com/killerangaswamy56/hadoop_config.git

2. Extract the bin folder from the **apache-hadoop-3.1.0-winutils-master** file downloaded from https://github.com/killerangaswamy56/hadoop_config.git.

3. Make a copy of the bin folder at **C:\hadoop\hadoop-3.2.4\bin**, and then delete the folder you made the copy from.

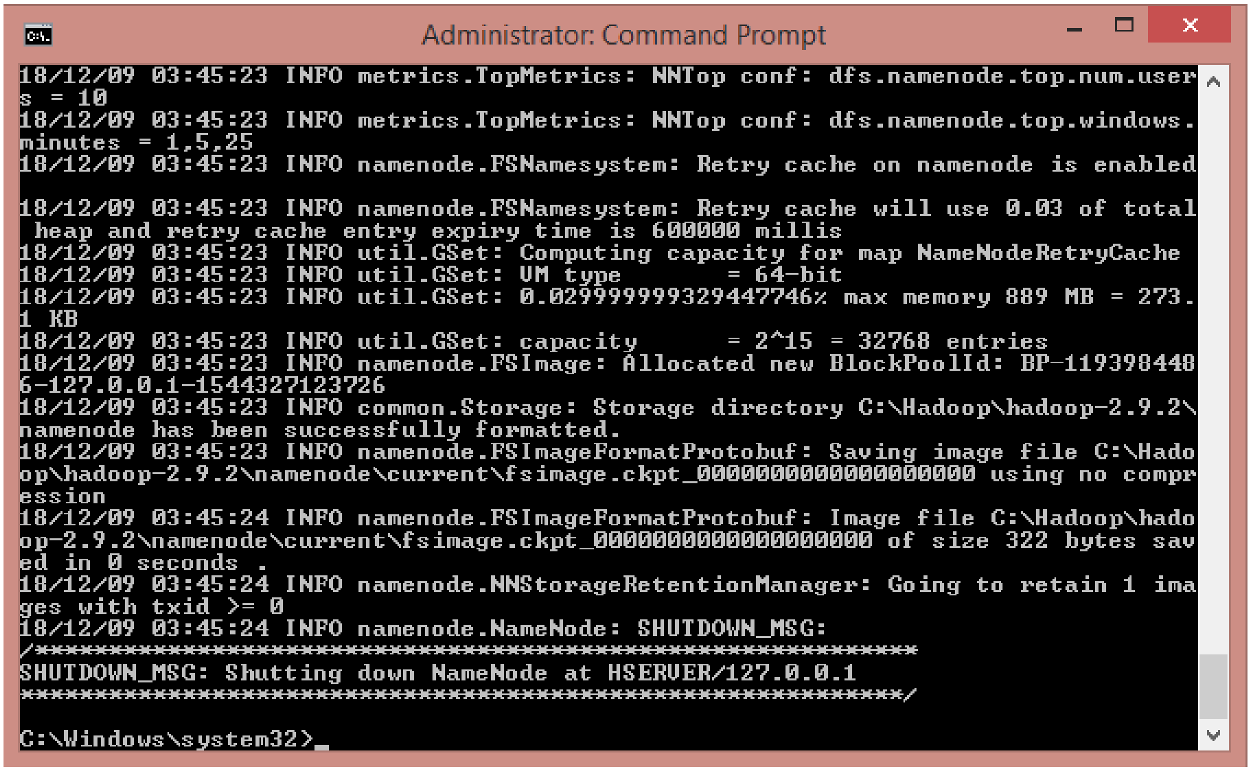
4. Copy the bin folder you extracted from the **apache-hadoop-3.1.0-winutils-master** file to **C:\hadoop\hadoop-3.2.4\**; it replaces the bin folder you deleted.

Now we must follow the instructions in section. Format the FileSystem.” This is done by executing the following command (with administrator privileges) from a command shell:

Code Listing 6: Format of the Filesystem

```
hdfs namenode -format
```

You should now see the following on your screen.



```
18/12/09 03:45:23 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.user
s = 10
18/12/09 03:45:23 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.
minutes = 1,5,25
18/12/09 03:45:23 INFO namenode.FSNamesystem: Retry cache on namenode is enabled

18/12/09 03:45:23 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total
 heap and retry cache entry expiry time is 600000 millis
18/12/09 03:45:23 INFO util.GSet: Computing capacity for map NameNodeRetryCache
18/12/09 03:45:23 INFO util.GSet: VM type       = 64-bit
18/12/09 03:45:23 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.
1 KB
18/12/09 03:45:23 INFO util.GSet: capacity      = 2^15 = 32768 entries
18/12/09 03:45:23 INFO namenode.FSImage: Allocated new BlockPoolId: BP-119398448
6-127.0.0.1-1544327123726
18/12/09 03:45:23 INFO common.Storage: Storage directory C:\Hadoop\hadoop-2.9.2\
namenode has been successfully formatted.
18/12/09 03:45:23 INFO namenode.FSImageFormatProtobuf: Saving image file C:\Hado
op\hadoop-2.9.2\namenode\current\fsimage.ckpt_0000000000000000000 using no compr
ession
18/12/09 03:45:24 INFO namenode.FSImageFormatProtobuf: Image file C:\Hadoop\hado
op-2.9.2\namenode\current\fsimage.ckpt_0000000000000000000 of size 322 bytes sav
ed in 0 seconds .
18/12/09 03:45:24 INFO namenode.NNStorageRetentionManager: Going to retain 1 ima
ges with txid >= 0
18/12/09 03:45:24 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at HSERVER/127.0.0.1
************************************************************/

C:\Windows\system32>_
```

Figure 16: Successful formatting of the Filesystem

You must now copy the **hadoop-yarn-server-timelineservice-3.2.4** file from **C:\hadoop\hadoop-3.2.4\share\hadoop\yarn\timelineservice** to the folder **C:\hadoop\hadoop-3.2.4\share\hadoop\yarn**. We can start Hadoop with the instructions in sections 3.5 and 3.6. of the Hadoop Wiki page, called “3.5. Start HDFS Daemons” and “3.6. Start YARN Daemons and run a YARN job.”

You start HDFS daemons by running the following code from the command prompt.

Code Listing 7: Start HDFS Daemons command

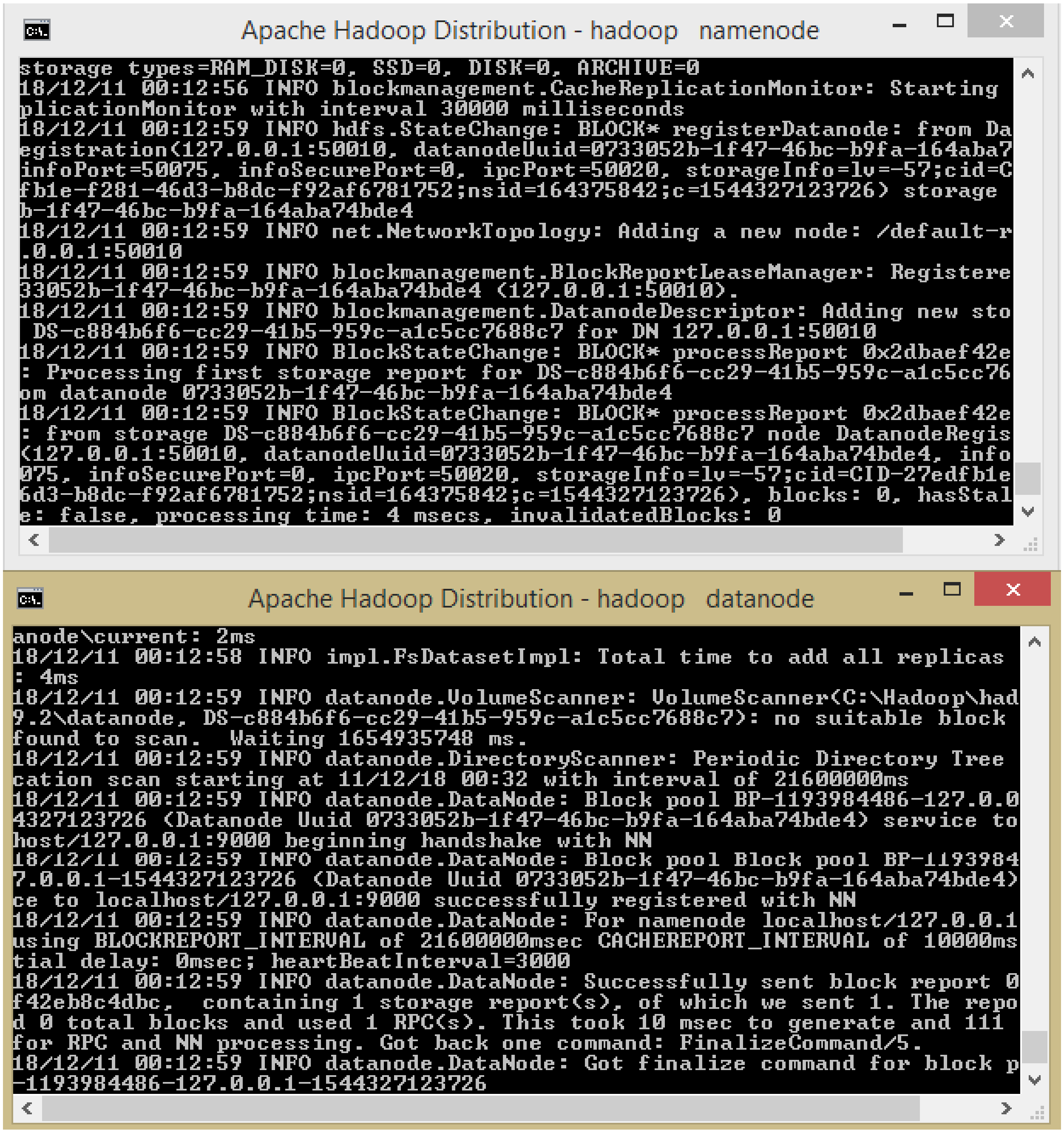
```
start-dfs
```

You start YARN daemons and run a YARN job by running the following code.

Code Listing 8: Start YARN daemons and run a YARN job command

```
start-yarn
```

You should now see the Hadoop **namenode** and **datanode** successfully started.

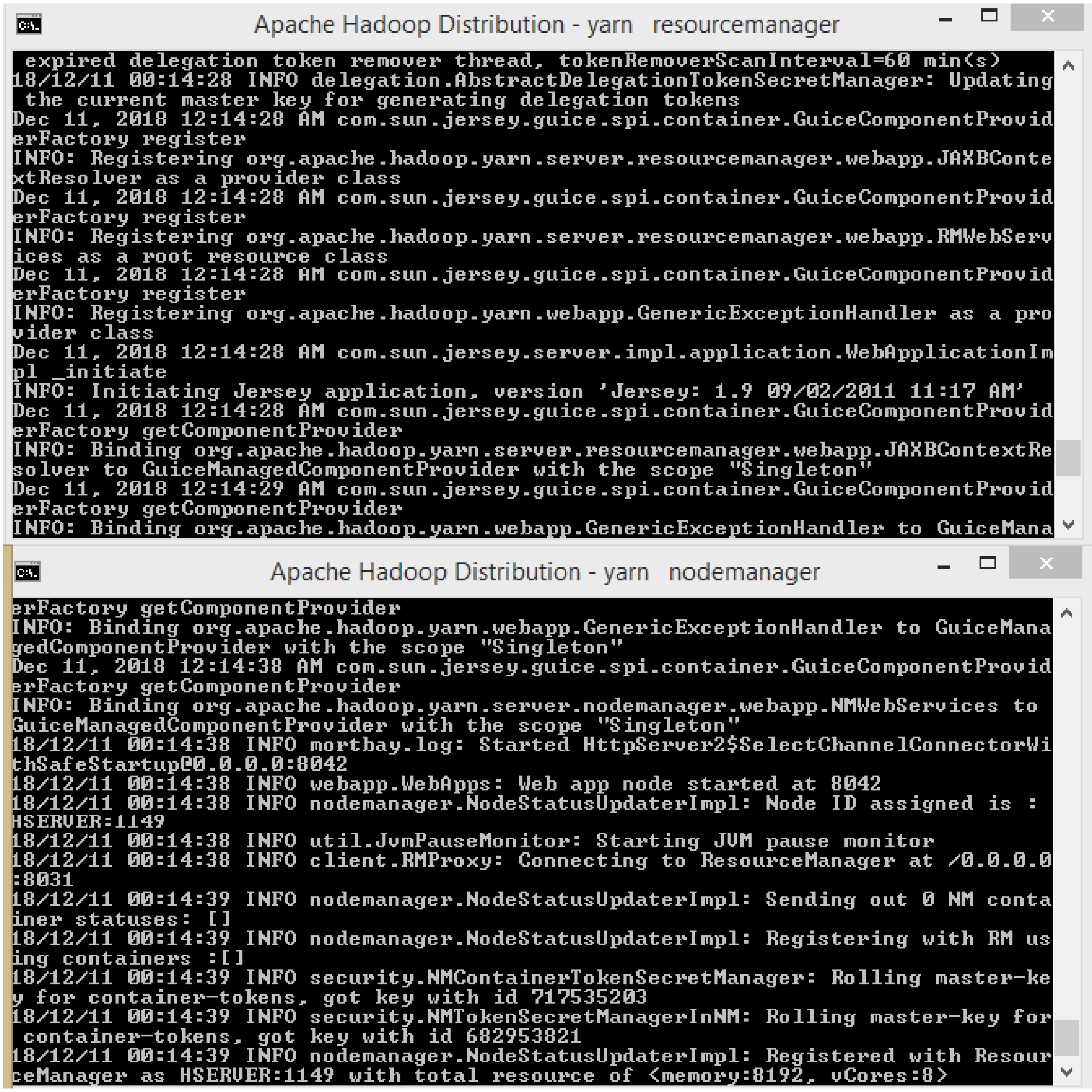


```
storage types=RAM_DISK=0, SSD=0, DISK=0, ARCHIVE=0
18/12/11 00:12:56 INFO blockmanagement.CacheReplicationMonitor: Starting
plicationMonitor with interval 30000 milliseconds
18/12/11 00:12:59 INFO hdfs.StateChange: BLOCK* registerDatanode: from Da
egistration(127.0.0.1:50010, datanodeUuid=0733052b-1f47-46bc-b9fa-164aba7
infoPort=50075, infoSecurePort=0, ipcPort=50020, storageInfo=lv=-57;cid=C
fb1e-f281-46d3-b8dc-f92af6781752;nsid=164375842;c=1544327123726) storage
b-1f47-46bc-b9fa-164aba74bde4
18/12/11 00:12:59 INFO net.NetworkTopology: Adding a new node: /default-r
.0.0.1:50010
18/12/11 00:12:59 INFO blockmanagement.BlockReportLeaseManager: Registere
33052b-1f47-46bc-b9fa-164aba74bde4 (127.0.0.1:50010).
18/12/11 00:12:59 INFO blockmanagement.DatanodeDescriptor: Adding new sto
 DS-c884b6f6-cc29-41b5-959c-a1c5cc7688c7 for DN 127.0.0.1:50010
18/12/11 00:12:59 INFO BlockStateChange: BLOCK* processReport 0x2dbaef42e
: Processing first storage report for DS-c884b6f6-cc29-41b5-959c-a1c5cc76
om datanode 0733052b-1f47-46bc-b9fa-164aba74bde4
18/12/11 00:12:59 INFO BlockStateChange: BLOCK* processReport 0x2dbaef42e
: from storage DS-c884b6f6-cc29-41b5-959c-a1c5cc7688c7 node DatanodeRegis
(127.0.0.1:50010, datanodeUuid=0733052b-1f47-46bc-b9fa-164aba74bde4, info
075, infoSecurePort=0, ipcPort=50020, storageInfo=lv=-57;cid=CID-27edfb1e
6d3-b8dc-f92af6781752;nsid=164375842;c=1544327123726), blocks: 0, hasStal
e: false, processing time: 4 msecs, invalidatedBlocks: 0
```

```
anode\current: 2ms
18/12/11 00:12:58 INFO impl.FsDatasetImpl: Total time to add all replicas
: 4ms
18/12/11 00:12:59 INFO datanode.VolumeScanner: VolumeScanner(C:\Hadoop\had
9.2\datanode, DS-c884b6f6-cc29-41b5-959c-a1c5cc7688c7): no suitable block
found to scan.  Waiting 1654935748 ms.
18/12/11 00:12:59 INFO datanode.DirectoryScanner: Periodic Directory Tree
cation scan starting at 11/12/18 00:32 with interval of 21600000ms
18/12/11 00:12:59 INFO datanode.DataNode: Block pool BP-1193984486-127.0.0
4327123726 (Datanode Uuid 0733052b-1f47-46bc-b9fa-164aba74bde4) service to
host/127.0.0.1:9000 beginning handshake with NN
18/12/11 00:12:59 INFO datanode.DataNode: Block pool Block pool BP-1193984
7.0.0.1-1544327123726 (Datanode Uuid 0733052b-1f47-46bc-b9fa-164aba74bde4)
ce to localhost/127.0.0.1:9000 successfully registered with NN
18/12/11 00:12:59 INFO datanode.DataNode: For namenode localhost/127.0.0.1
using BLOCKREPORT_INTERVAL of 21600000msec CACHEREPORT_INTERVAL of 10000ms
tial delay: 0msec; heartBeatInterval=3000
18/12/11 00:12:59 INFO datanode.DataNode: Successfully sent block report 0
f42eb8c4dbc,  containing 1 storage report(s), of which we sent 1. The repo
d 0 total blocks and used 1 RPC(s). This took 10 msec to generate and 111
for RPC and NN processing. Got back one command: FinalizeCommand/5.
18/12/11 00:12:59 INFO datanode.DataNode: Got finalize command for block p
-1193984486-127.0.0.1-1544327123726
```

Figure 17: Hadoop namenode and datanode started successfully

In addition, you will see the YARN **resourcemanager** and YARN **nodemanager** successfully started.

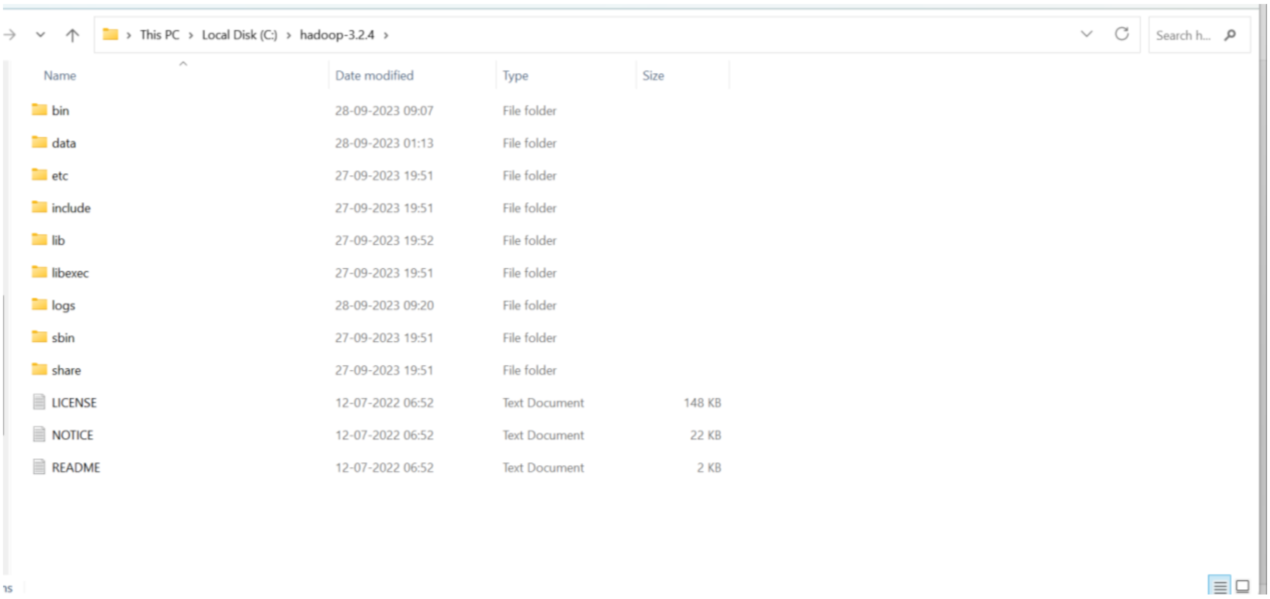

Apache Hadoop Distribution - yarn resourcemanager
Apache Hadoop Distribution - yarn nodemanager

```
 expired delegation token remover thread, tokenRemoverScanInterval=60 min(s)
18/12/11 00:14:28 INFO delegation.AbstractDelegationTokenSecretManager: Updating
 the current master key for generating delegation tokens
Dec 11, 2018 12:14:28 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory register
INFO: Registering org.apache.hadoop.yarn.server.resourcemanager.webapp.JAXBConte
xtResolver as a provider class
Dec 11, 2018 12:14:28 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory register
INFO: Registering org.apache.hadoop.yarn.server.resourcemanager.webapp.RMWebServ
ices as a root resource class
Dec 11, 2018 12:14:28 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory register
INFO: Registering org.apache.hadoop.yarn.webapp.GenericExceptionHandler as a pro
vider class
Dec 11, 2018 12:14:28 AM com.sun.jersey.server.impl.application.WebApplicationIm
pl _initiate
INFO: Initiating Jersey application, version 'Jersey: 1.9 09/02/2011 11:17 AM'
Dec 11, 2018 12:14:28 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.resourcemanager.webapp.JAXBContextRe
solver to GuiceManagedComponentProvider with the scope "Singleton"
Dec 11, 2018 12:14:29 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.webapp.GenericExceptionHandler to GuiceMana
```

```
erFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.webapp.GenericExceptionHandler to GuiceMana
gedComponentProvider with the scope "Singleton"
Dec 11, 2018 12:14:38 AM com.sun.jersey.guice.spi.container.GuiceComponentProvid
erFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.NMWebServices to
GuiceManagedComponentProvider with the scope "Singleton"
18/12/11 00:14:38 INFO mortbay.log: Started HttpServer2$SelectChannelConnectorWi
thSafeStartup@0.0.0.0:8042
18/12/11 00:14:38 INFO webapp.WebApps: Web app node started at 8042
18/12/11 00:14:38 INFO nodemanager.NodeStatusUpdaterImpl: Node ID assigned is :
HSERVER:1149
18/12/11 00:14:38 INFO util.JvmPauseMonitor: Starting JVM pause monitor
18/12/11 00:14:38 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0
:8031
18/12/11 00:14:39 INFO nodemanager.NodeStatusUpdaterImpl: Sending out 0 NM conta
iner statuses: []
18/12/11 00:14:39 INFO nodemanager.NodeStatusUpdaterImpl: Registering with RM us
ing containers :[]
18/12/11 00:14:39 INFO security.NMContainerTokenSecretManager: Rolling master-ke
y for container-tokens, got key with id 717535203
18/12/11 00:14:39 INFO security.NMTokenSecretManagerInNM: Rolling master-key for
 container-tokens, got key with id 682953821
18/12/11 00:14:39 INFO nodemanager.NodeStatusUpdaterImpl: Registered with Resour
ceManager as HSERVER:1149 with total resource of <memory:8192, vCores:8>
```

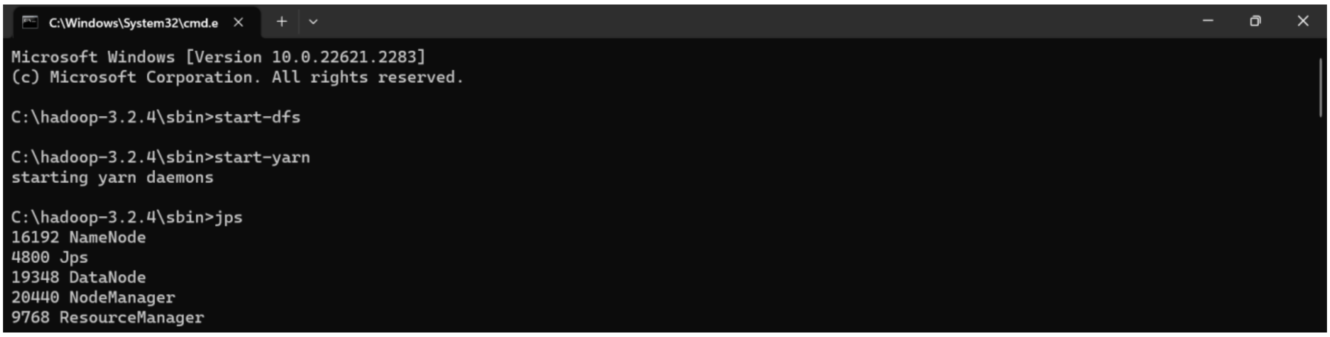
Figure 18: Yarn resourcemanager and Yarn nodemanager successfully started

The finished Hadoop installation directory is shown in the following image.

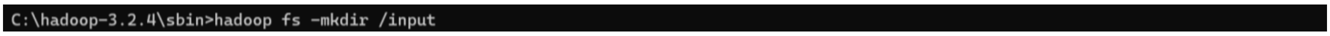


Name	Date modified	Type	Size
bin	28-09-2023 09:07	File folder	
data	28-09-2023 01:13	File folder	
etc	27-09-2023 19:51	File folder	
include	27-09-2023 19:51	File folder	
lib	27-09-2023 19:52	File folder	
libexec	27-09-2023 19:51	File folder	
logs	28-09-2023 09:20	File folder	
sbin	27-09-2023 19:51	File folder	
share	27-09-2023 19:51	File folder	
LICENSE	12-07-2022 06:52	Text Document	148 KB
NOTICE	12-07-2022 06:52	Text Document	22 KB
README	12-07-2022 06:52	Text Document	2 KB

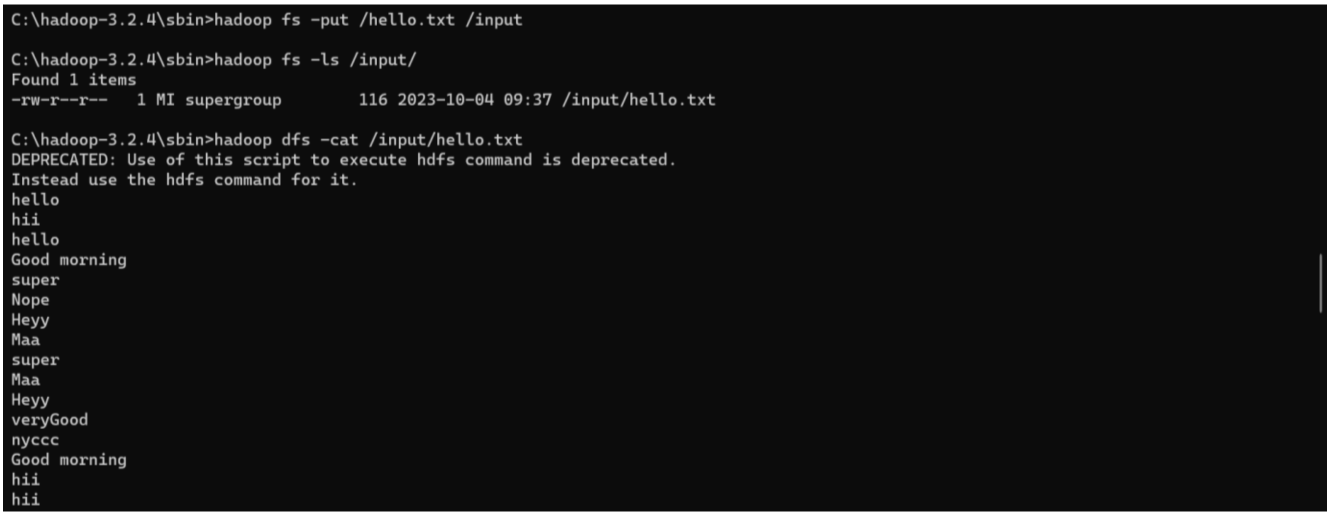
Figure 19: Final Hadoop installation directory



```
Microsoft Windows [Version 10.0.22621.2283]
(c) Microsoft Corporation. All rights reserved.

C:\hadoop-3.2.4\sbin>start-dfs

C:\hadoop-3.2.4\sbin>start-yarn
starting yarn daemons

C:\hadoop-3.2.4\sbin>jps
16192 NameNode
4800 Jps
19348 DataNode
20440 NodeManager
9768 ResourceManager
```



```
C:\hadoop-3.2.4\sbin>hadoop fs -mkdir /input
```

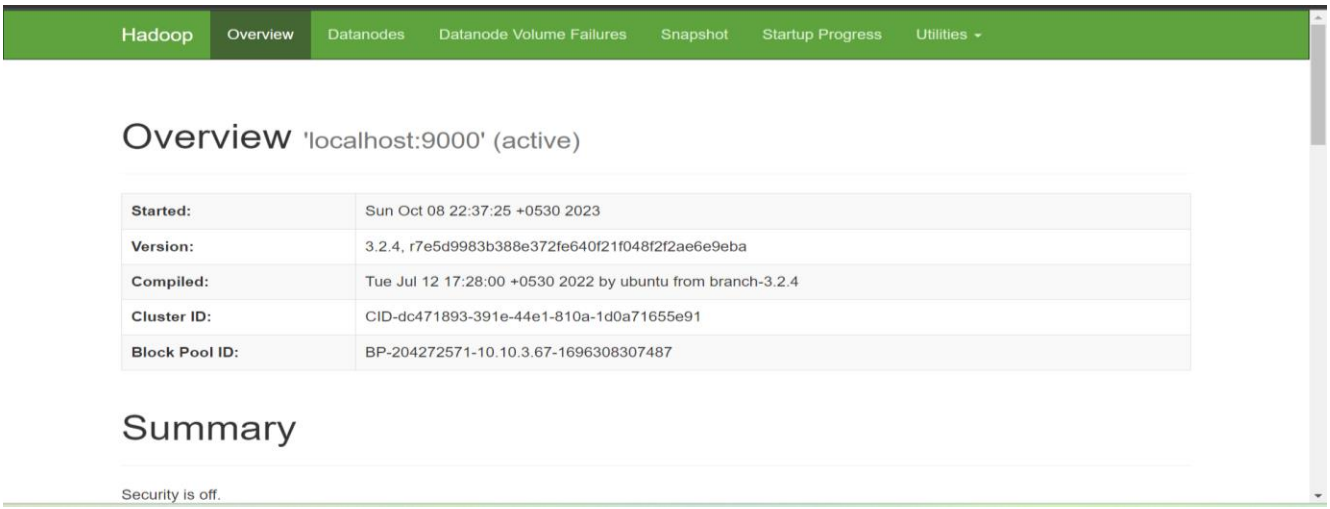


```
C:\hadoop-3.2.4\sbin>hadoop fs -put /hello.txt /input

C:\hadoop-3.2.4\sbin>hadoop fs -ls /input/
Found 1 items
-rw-r--r--   1 MI supergroup        116 2023-10-04 09:37 /input/hello.txt

C:\hadoop-3.2.4\sbin>hadoop dfs -cat /input/hello.txt
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
hello
hii
hello
Good morning
super
Nope
Heyy
Maa
super
Maa
Heyy
veryGood
nyccc
Good morning
hii
hii
```

Now Follow the below steps to view your data in server

Step:1 After data files are Loaded into the HDFS server then type the below url in your default browser. http://localhost:9870

Step:2 Then it will navigates to the below Hadoop page



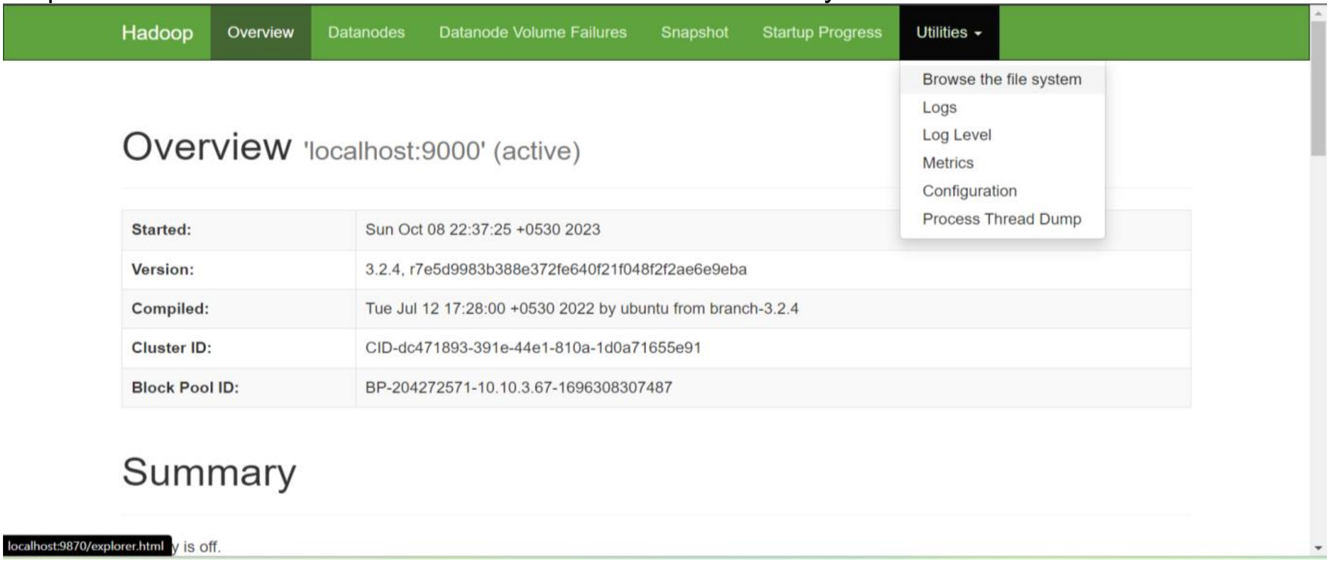
Overview 'localhost:9000' (active)

Started:	Sun Oct 08 22:37:25 +0530 2023
Version:	3.2.4, r7e5d9983b388e372fe640f21f048f2f2ae6e9eba
Compiled:	Tue Jul 12 17:28:00 +0530 2022 by ubuntu from branch-3.2.4
Cluster ID:	CID-dc471893-391e-44e1-810a-1d0a71655e91
Block Pool ID:	BP-204272571-10.10.3.67-1696308307487

Summary

Security is off.

Step:3 Now Click on the Utilities then click on “Browse the file system”



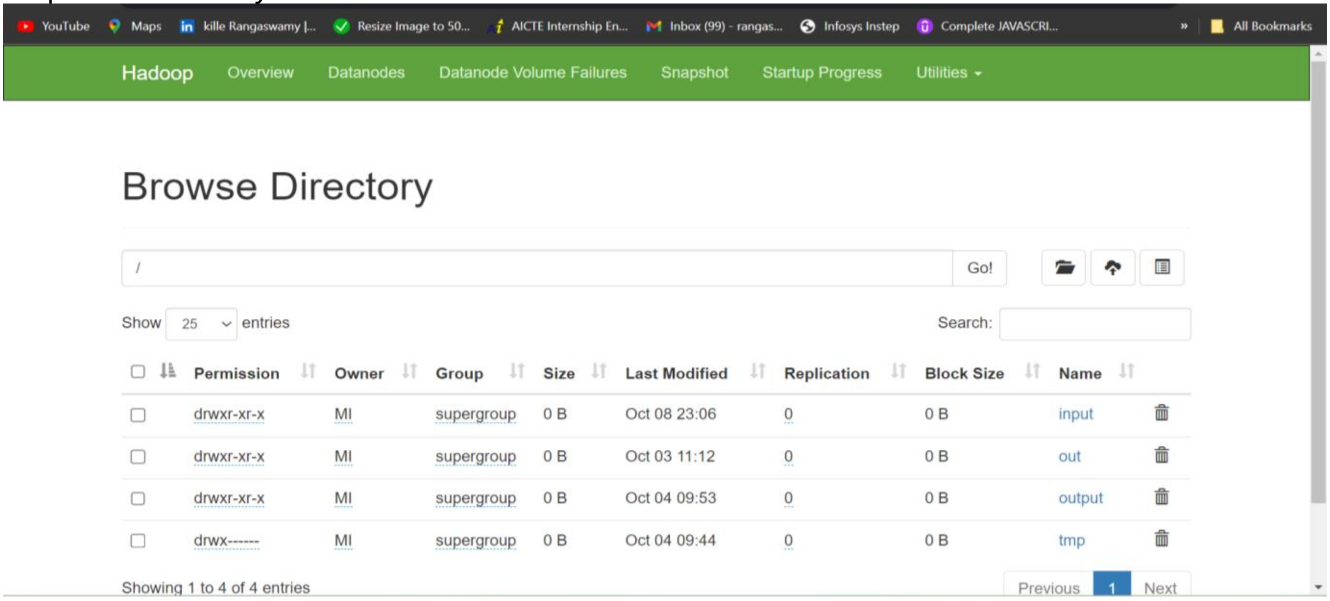
Overview 'localhost:9000' (active)

Started:	Sun Oct 08 22:37:25 +0530 2023
Version:	3.2.4, r7e5d9983b388e372fe640f21f048f2f2ae6e9eba
Compiled:	Tue Jul 12 17:28:00 +0530 2022 by ubuntu from branch-3.2.4
Cluster ID:	CID-dc471893-391e-44e1-810a-1d0a71655e91
Block Pool ID:	BP-204272571-10.10.3.67-1696308307487

Summary

y is off.

Step:4 Now Check your data file is loaded into the server or not as shown below



Browse Directory

/

Show 25 entries

Search:

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	MI	supergroup	0 B	Oct 08 23:06	0	0 B	input	🗑
□	drwxr-xr-x	MI	supergroup	0 B	Oct 03 11:12	0	0 B	out	🗑
□	drwxr-xr-x	MI	supergroup	0 B	Oct 04 09:53	0	0 B	output	🗑
□	drwx------	MI	supergroup	0 B	Oct 04 09:44	0	0 B	tmp	🗑

Showing 1 to 4 of 4 entries

Previous 1 Next