



INSTALLATION AND COMMANDS

WORD COUNTS EXAMPLE

PDC Project: Hadoop

Name: Muhammad Allah Rakha

Roll No: P19-0006 (BCS-6A)

Name: Muhammad Aamir Khan

Roll No: P18-0121 (BCS-6B)

Source Link

Website: <https://aaaastark.herokuapp.com>

GitHub: <https://github.com/aaaastark>

INTRODUCTION TO HADOOP

What is Hadoop?

Hadoop is an open source software programming framework for storing a large amount of data and performing the computation. Its framework is based on Java programming with some native code in C and shell scripts.

- Apache Hadoop is an open source framework intended to make interaction with Big Data.
- Hadoop has made its place in the industries and companies that need to work on large data sets which are sensitive and needs efficient handling.
- Hadoop is a framework that enables processing of large data sets which reside in the form of clusters.

History of Hadoop?

Apache Software Foundation is the developers of Hadoop, and its co-founders are Doug Cutting and Mike Cafarella. In October 2003 the first paper release was Google File System. In January 2006, MapReduce development started on the Apache Nutch which consisted of around 6000 lines coding for it and around 5000 lines coding for HDFS. In April 2006 Hadoop 0.1.0 was released.

Some common frameworks of Hadoop?

- ✓ **Hive**- It uses HiveQL for data structuring and for writing complicated MapReduce in HDFS.
- ✓ **Drill**- It consists of user-defined functions and is used for data exploration.
- ✓ **Storm**- It allows real-time processing and streaming of data.
- ✓ **Spark**- It contains a Machine Learning Library (MLlib) for providing enhanced machine learning and is widely used for data processing. It also supports Java, Python, and Scala.
- ✓ **Pig**- It has Pig Latin, a SQL-Like language and performs data transformation of unstructured data.
- ✓ **Tez**- It reduces the complexities of Hive and Pig and helps in the running of their codes faster.

Advantages and Disadvantages of Hadoop?

Advantages:

- Ability to store a large amount of data.
- High flexibility.
- Cost effective.
- High computational power.
- Tasks are independent.
- Linear scaling.

Disadvantages:

- Not very effective for small data.
- Hard cluster management.
- Has stability issues.
- Security concerns.

Hadoop Ecosystem?

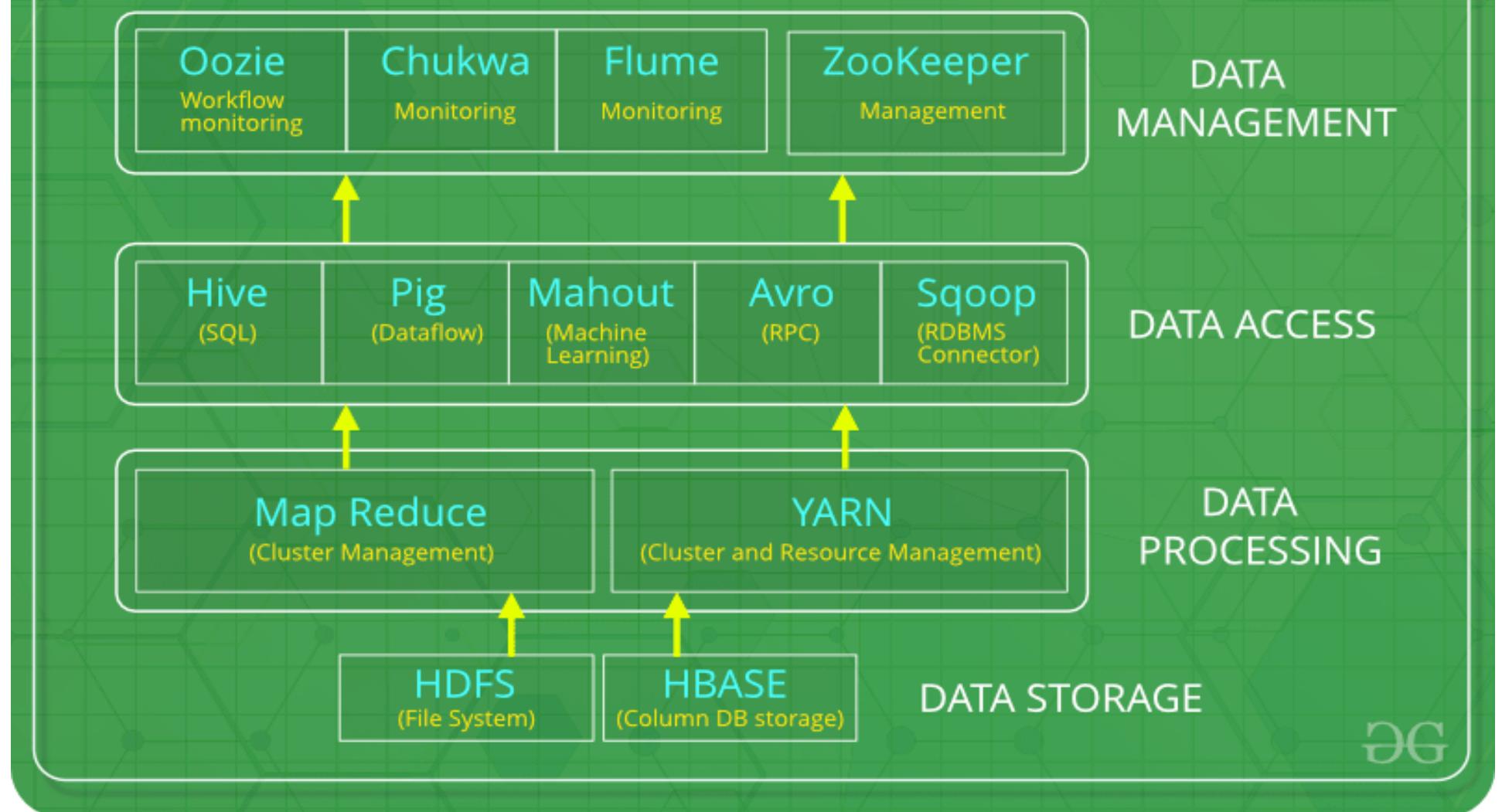
Hadoop Ecosystem is a platform or a suite which provides various services to solve the big data problems.

Following are the components that collectively form a Hadoop ecosystem:

- | | |
|--|---|
| ✓ HDFS : Hadoop Distributed File System | ✓ HBase : NoSQL Database |
| ✓ YARN : Yet Another Resource Negotiator | ✓ Mahout, Spark MLlib : Machine Learning algorithm libraries |
| ✓ MapReduce : Programming based Data Processing | ✓ Solar, Lucene : Searching and Indexing |
| ✓ Spark : In-Memory data processing | ✓ Zookeeper : Managing cluster |
| ✓ PIG, HIVE : Query based processing of data services | ✓ Oozie : Job Scheduling |

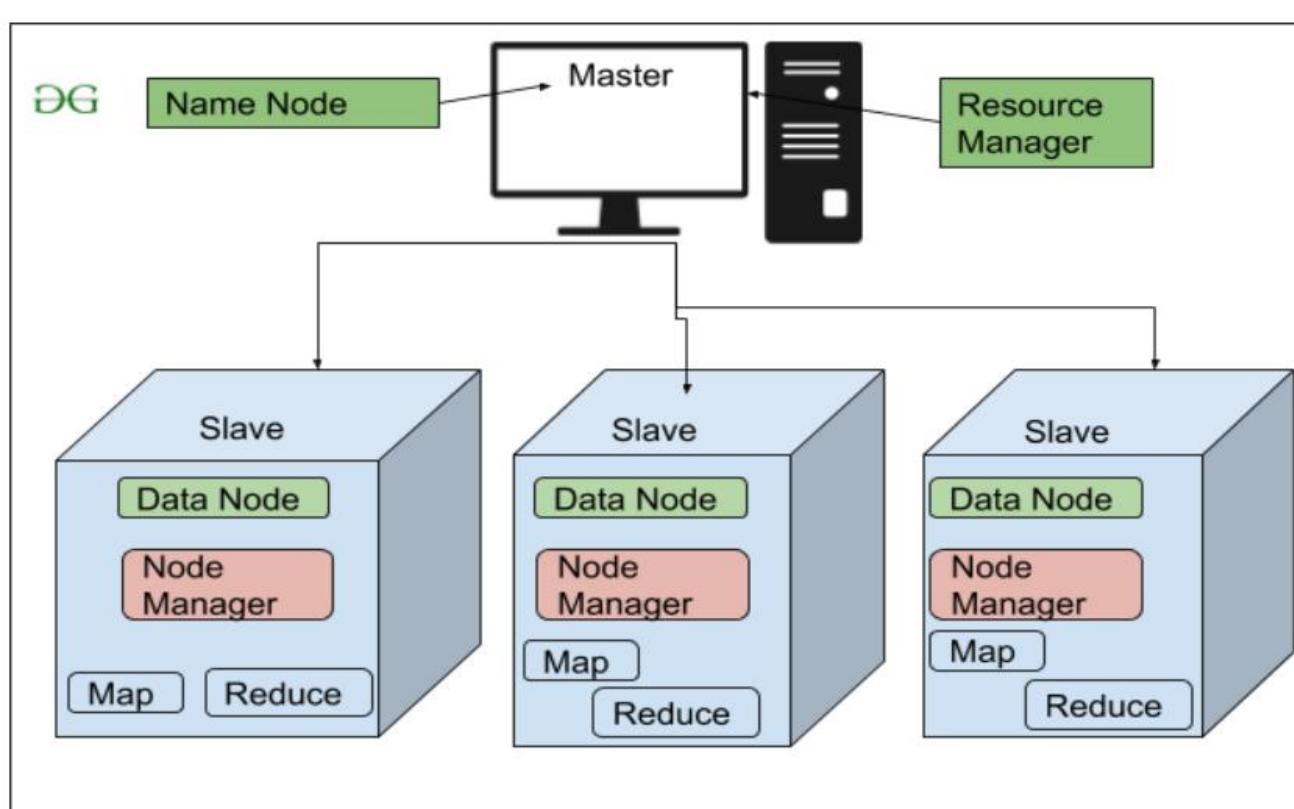
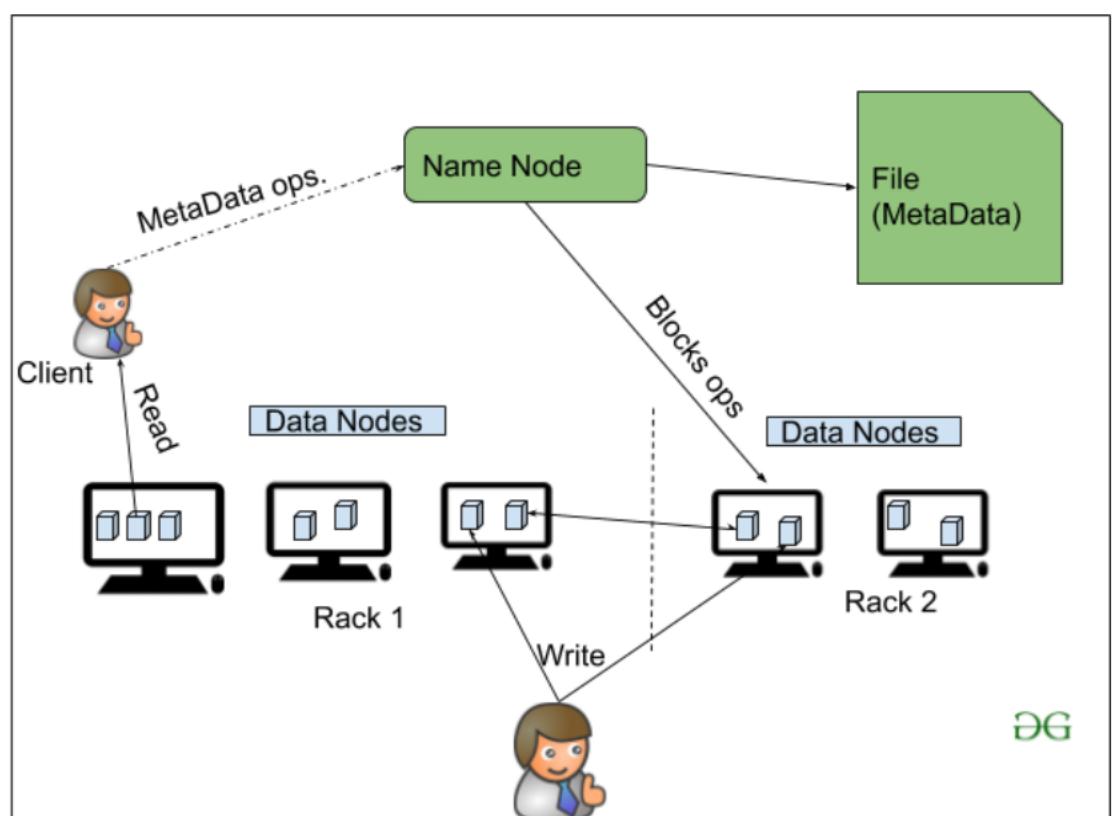
All these tools work collectively to provide services such as absorption, analysis, storage and maintenance of data etc.

Hadoop Ecosystem



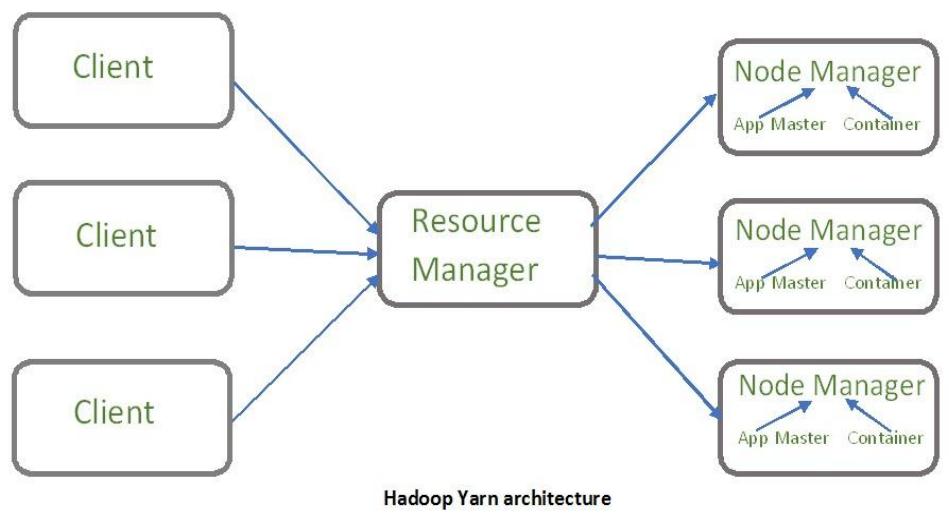
HDFS:

- HDFS is the primary or major component of Hadoop ecosystem and is responsible for storing large data sets of structured or unstructured data across various nodes and thereby maintaining the metadata in the form of log files.
- HDFS consists of two core components i.e.
 - Name node
 - Data Node
- Name Node is the prime node which contains metadata (data about data) requiring comparatively fewer resources than the data nodes that stores the actual data. These data nodes are commodity hardware in the distributed environment.



YARN:

- Yet Another Resource Negotiator, as the name implies, YARN is the one who helps to manage the resources across the clusters. In short, it performs scheduling and resource allocation for the Hadoop System.
- Consists of three major components i.e.
 - Resource Manager
 - Nodes Manager
 - Application Manager
- Resource manager has the privilege of allocating resources for the applications in a system whereas Node managers work on the allocation of resources such as CPU, memory, bandwidth per machine and later on acknowledges the resource manager. Application manager works as an interface between the resource manager and node manager and performs negotiations as per the requirement of the two.



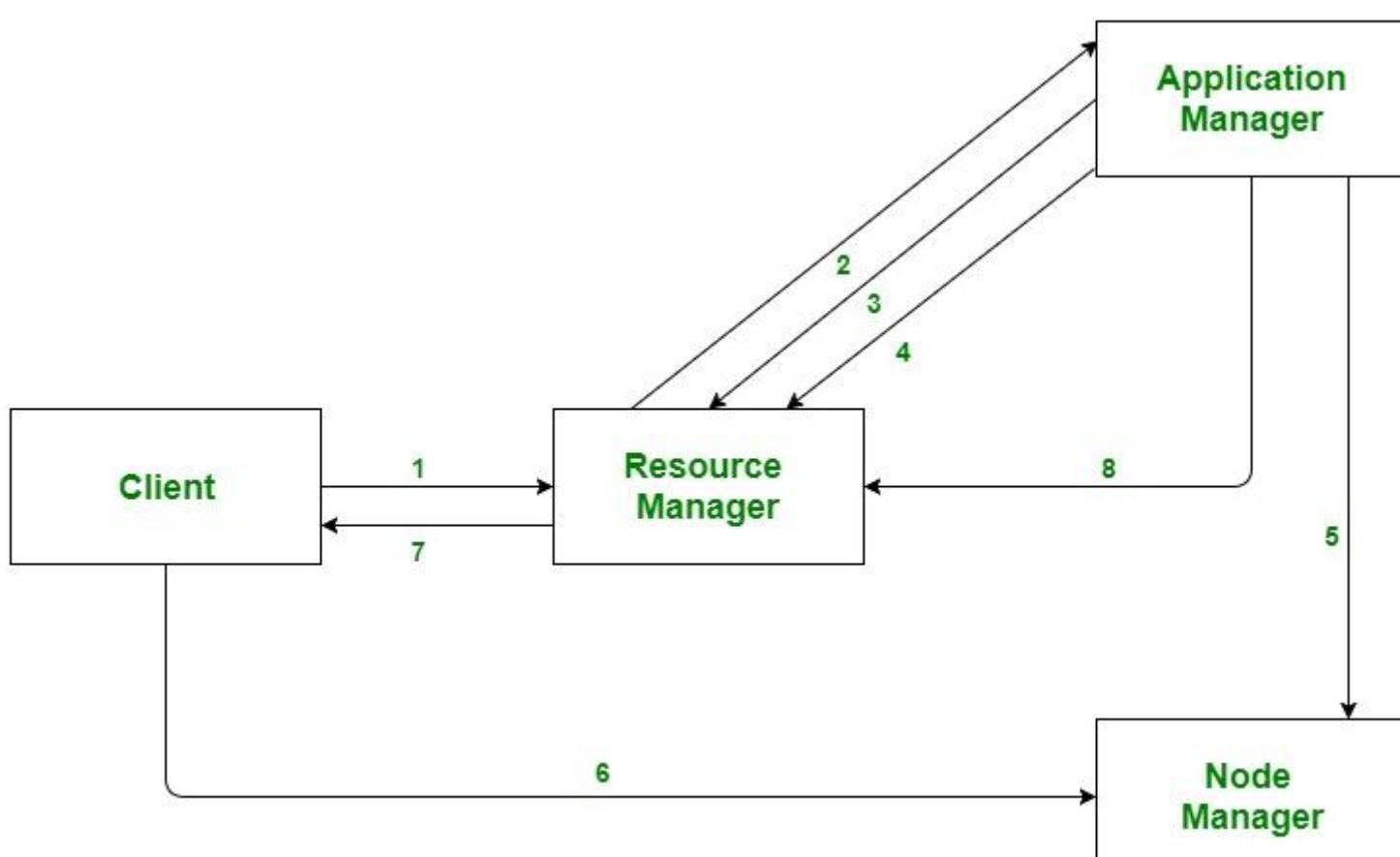
Hadoop Yarn architecture

Features of YARN:

- **Multi-Tenancy:** It allows multiple engine access thus giving organizations a benefit of multi-tenancy.
- **Scalability:** The scheduler in Resource manager of YARN architecture allows Hadoop to extend and manage thousands of nodes and clusters.
- **Cluster-Utilization:** YARN supports Dynamic utilization of cluster in Hadoop, which enables optimized Cluster Utilization.
- **Compatibility:** YARN supports the existing map-reduce applications without disruptions thus making it compatible with Hadoop 1.0 as well.

The main components of YARN architecture include:

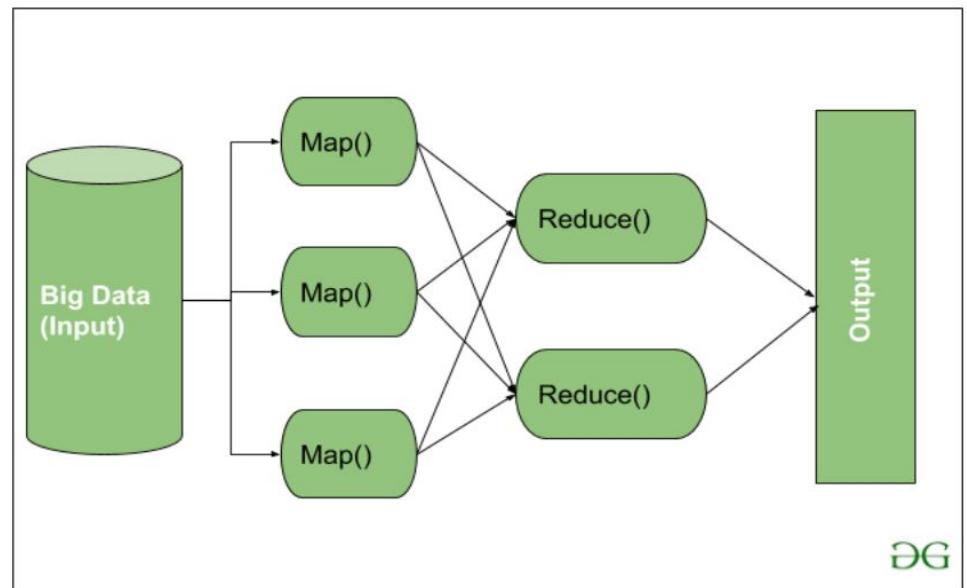
- **Client:** It submits map-reduce jobs.
- **Resource Manager:** Whenever it receives a processing request, it forwards it to the corresponding node manager and allocates resources for the completion of the request accordingly.
- **Node Manager:** Manages application and workflow and that particular node. It is also responsible for creating the container process and start it on the request of Application master. It registers with the Resource Manager and sends heartbeats with the health status of the node.
- **Application Master:** The application master is responsible for negotiating resources with the resource manager, tracking the status and monitoring progress of a single application. The application master requests the container from the node manager by sending a Container Launch Context(CLC) which includes everything an application needs to run. Once the application is started, it sends the health report to the resource manager from time-to-time.
- **Container:** It is a collection of physical resources such as RAM, CPU cores and disk on a single node. The containers are invoked by Container Launch Context(CLC) which is a record that contains information such as environment variables, security tokens, dependencies etc.



MapReduce:

MapReduce is a programming model used for efficient processing in parallel over large data-sets in a distributed manner. The data is first split and then combined to produce the final result.

- **Map()**: Performs sorting and filtering of data and thereby organizing them in the form of group. Map generates a key-value pair based result which is later on processed by the Reduce() method.
- **Reduce()**: Takes the output generated by Map() as input and combines those tuples into smaller set of tuples.



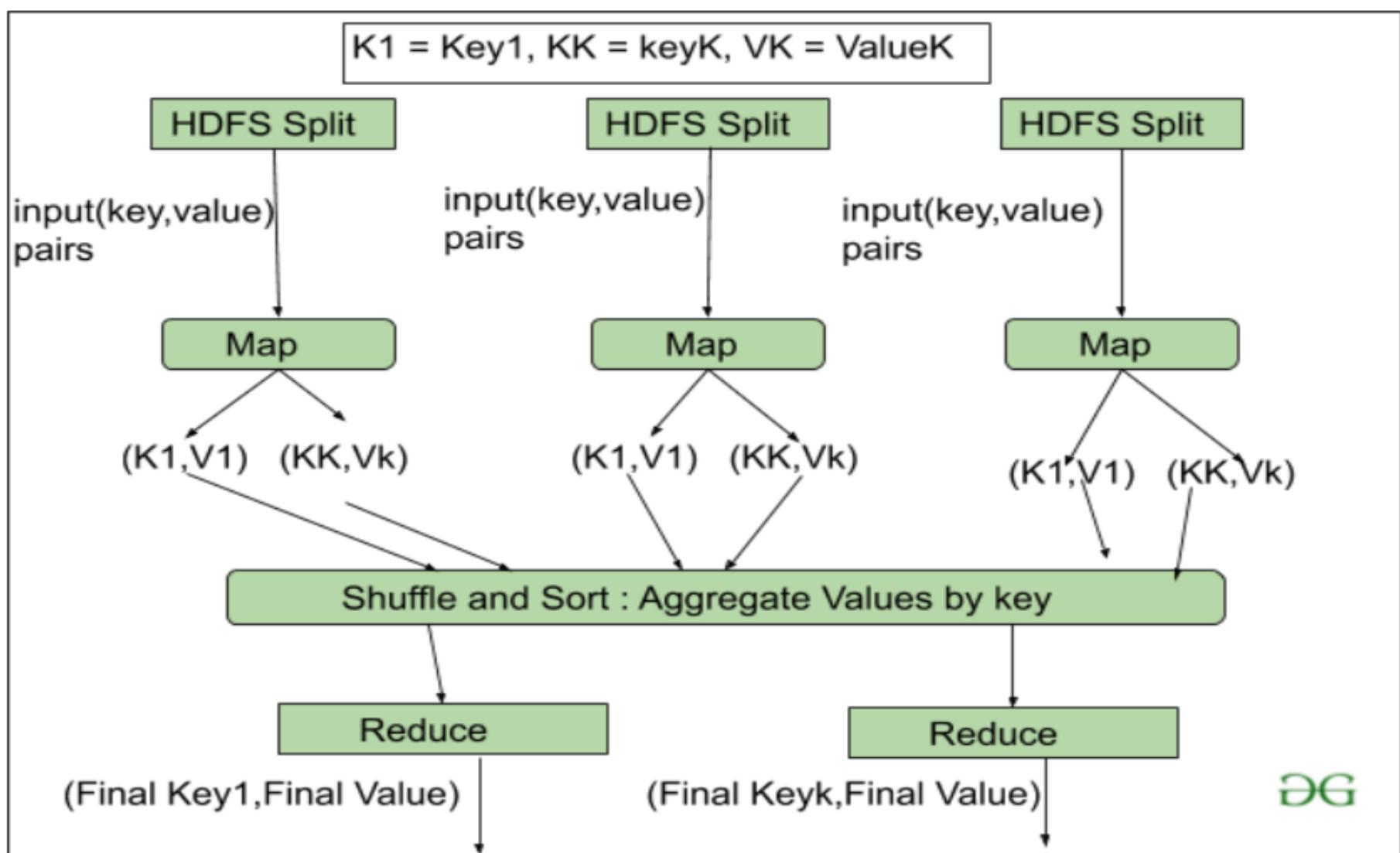
Let's understand the Map Task and Reduce Task in detail.

Map Task:

- **RecordReader**: The purpose of *recordreader* is to break the records. It is responsible for providing key-value pairs in a Map() function. The key is actually its locational information and value is the data associated with it.
- **Map**: Is to process the Tuples obtained from record reader. The Map() function either does not generate any key-value pair or generate multiple pairs of these tuples.
- **Combiner**: Combiner is used for grouping the data in the Map workflow.
- **Partitioner**: Partitioner is responsible for fetching key-value pairs generated in the Mapper Phases. Hashcode of each key is also fetched by this partitioner. Then partitioner performs its (Hashcode) modulus with the number of reducers ($\text{key.hashCode()} \% (\text{number of reducers})$).

Reduce Task:

- **Shuffle and Sort**: The Task of Reducer starts with this step, the process in which the Mapper generates the intermediate key-value and transfers them to the Reducer task is known as *Shuffling*. Using the Shuffling process the system can sort the data using its key value.
- **Reduce**: The main function or task of the Reduce is to gather the Tuple generated from Map and then perform some sorting and aggregation sort of process on those key-value depending on its key element.
- **OutputFormat**: Once all the operations are performed, the key-value pairs are written into the file with the help of record writer, each record in a new line, and the key and value in a space-separated manner.



PIG:

Pig was basically developed by Yahoo which works on a pig Latin language, which is **Query based language similar to SQL**. It is a platform for structuring the data flow, processing and analyzing huge data sets.

HIVE:

HIVE performs reading and writing of large data sets. However, its query language is called as **HQL (Hive Query Language)**. It is highly scalable as it allows real-time processing and batch processing both. HIVE too comes with two components: **JDBC Drivers** and **HIVE Command Line**.

Mahout:

Mahout, allows Machine Learnability to a system or application. It provides various libraries or functionalities such as collaborative filtering, clustering, and classification which are nothing but concepts of Machine learning.

Apache Spark:

Apache Spark is a lightning-fast cluster computing technology, designed for fast computation. The main feature of Spark is its **in-memory cluster computing** that increases the processing speed of an application. Spark is designed to cover a wide range of workloads such as **batch applications, iterative algorithms, interactive queries and streaming**.

Apache HBase:

It's a NoSQL database which supports all kinds of data and thus capable of handling anything of Hadoop Database. It provides capabilities of **Google's BigTable**, thus able to work on Big Data sets effectively.

Solr, Lucene:

These are the two services that **perform the task of searching and indexing** with the help of some java libraries, especially **Lucene** is based on Java which allows spell check mechanism, as well. However, Lucene is driven by Solr.

Zookeeper:

There was a **huge issue of management of coordination and synchronization** among the resources or the components of Hadoop which resulted in inconsistency, often.

Zookeeper overcame all the problems by performing **synchronization, inter-component based communication, grouping, and maintenance**.

Oozie:

Oozie simply performs the task of a scheduler, thus **scheduling jobs and binding them together as a single unit**. There are two kinds of jobs i.e. **Oozie workflow** and **Oozie coordinator jobs**.

- **Oozie workflow** is the jobs that need to be executed in a **sequentially ordered manner**.
- **Oozie Coordinator** jobs are those that are **triggered** when some data or external stimulus is given to it.

INSTALLATION OF HADOOP

Admin User: hassanraza

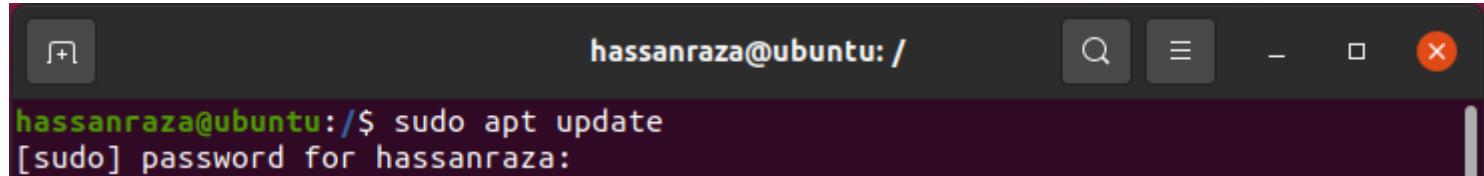
User Password: Hassan

Hadoop User: hdoop

User Password: hdoop

NOTE: Install the new Ubuntu using the Dual Boot or VMware.

COMMANDS: Following these commands for Hadoop Installation.

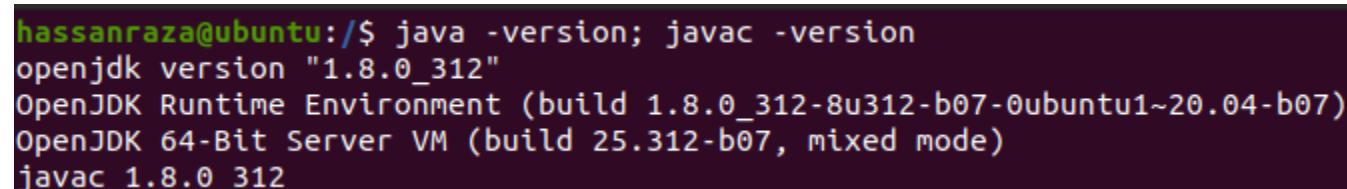


```
hassanraza@ubuntu:/$ sudo apt update  
[sudo] password for hassanraza:
```

Install the Java JDK 8. Required to Hadoop installation process.

```
hassanraza@ubuntu:/$ sudo apt install openjdk-8-jdk -y
```

Check the version of Java and Java Compiler.

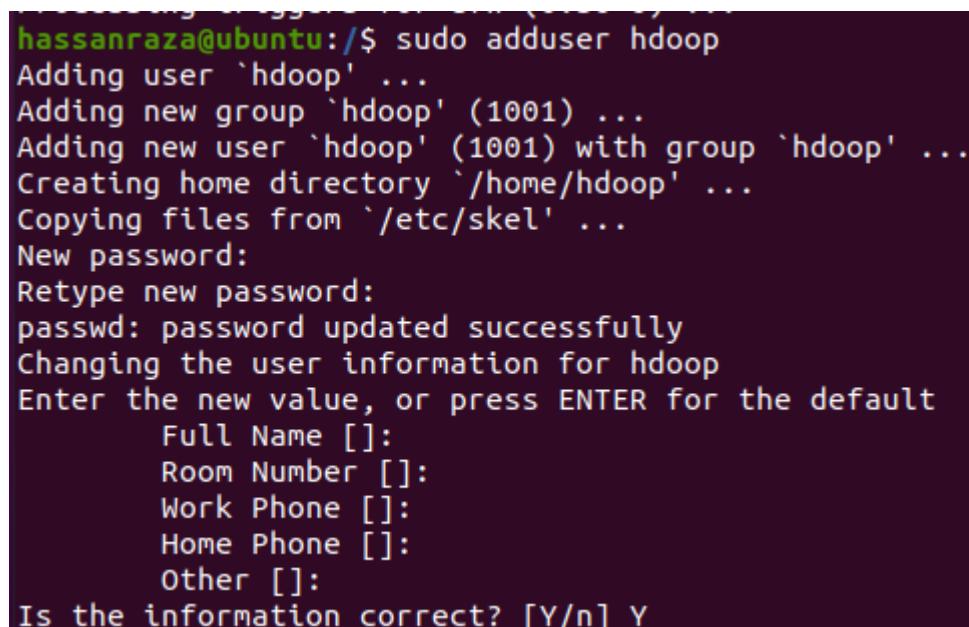


```
hassanraza@ubuntu:/$ java -version; javac -version  
openjdk version "1.8.0_312"  
OpenJDK Runtime Environment (build 1.8.0_312-8u312-b07-0ubuntu1~20.04-b07)  
OpenJDK 64-Bit Server VM (build 25.312-b07, mixed mode)  
javac 1.8.0_312
```

OpenSSH provides a large suite of secure tunneling capabilities, several authentication methods, and sophisticated configuration options. Secure Shell (SSH) protocol family of tools for **remotely controlling, or transferring files between, computers. Install for both Client and Server.**

```
hassanraza@ubuntu:/$ sudo apt install openssh-server openssh-client -y  
Reading package lists... Done
```

Add the new user to Hadoop Setup.



```
hassanraza@ubuntu:/$ sudo adduser hdoop  
Adding user `hdoop' ...  
Adding new group `hdoop' (1001) ...  
Adding new user `hdoop' (1001) with group `hdoop' ...  
Creating home directory `/home/hdoop' ...  
Copying files from `/etc/skel' ...  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for hdoop  
Enter the new value, or press ENTER for the default  
    Full Name []:  
    Room Number []:  
    Work Phone []:  
    Home Phone []:  
    Other []:  
Is the information correct? [Y/n] Y
```

Change the Root user to Hadoop user.

```
hassanraza@ubuntu:/$ su - hdoop  
Password:  
hdoop@ubuntu:~$
```

Generate the Key RSA then save into “.ssh” Directory. Two files (id_rsa and id_rsa.pub).

```
hadoop@ubuntu:~$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/home/hadoop/.ssh'.
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:H/hPHZx9nFRcMF4wO10w5Pdzlo2CxlUdlpEjHPyUDWY hadoop@ubuntu
The key's randomart image is:
+---[RSA 3072]---+
|          o.E@&|
|          @.&=|
|          . @.o|
|         o o ..X=|
|        S = . =+o|
|       + . o o+|
|       o . .|
|       o   |
|       .   |
+---[SHA256]---+
```

Create and Write the content of “id_rsa.pub” key file into “authorized_keys” file.

```
hadoop@ubuntu:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hadoop@ubuntu:~$ ls ~/.ssh/
authorized_keys  id_rsa  id_rsa.pub
hadoop@ubuntu:~$
```

Change the Mode of key file “authorized_keys”. Assign a “Read” and “Write” mode.

```
hadoop@ubuntu:~$ chmod 0600 ~/.ssh/authorized_keys
hadoop@ubuntu:~$ ls -lrt ~/.ssh/
total 12
-rw-r--r-- 1 hadoop hadoop 566 Mar 23 21:56 id_rsa.pub
-rw----- 1 hadoop hadoop 2602 Mar 23 21:56 id_rsa
-rw----- 1 hadoop hadoop 566 Mar 23 21:58 authorized_keys
```

Open the Secure Shell (SSH) into local-host port.

```
hadoop@ubuntu:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:aJ6WnievAC1oTqoGc5QCR7ntW4pAtv0DbrQBiEDNuso.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

246 updates can be applied immediately.
163 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

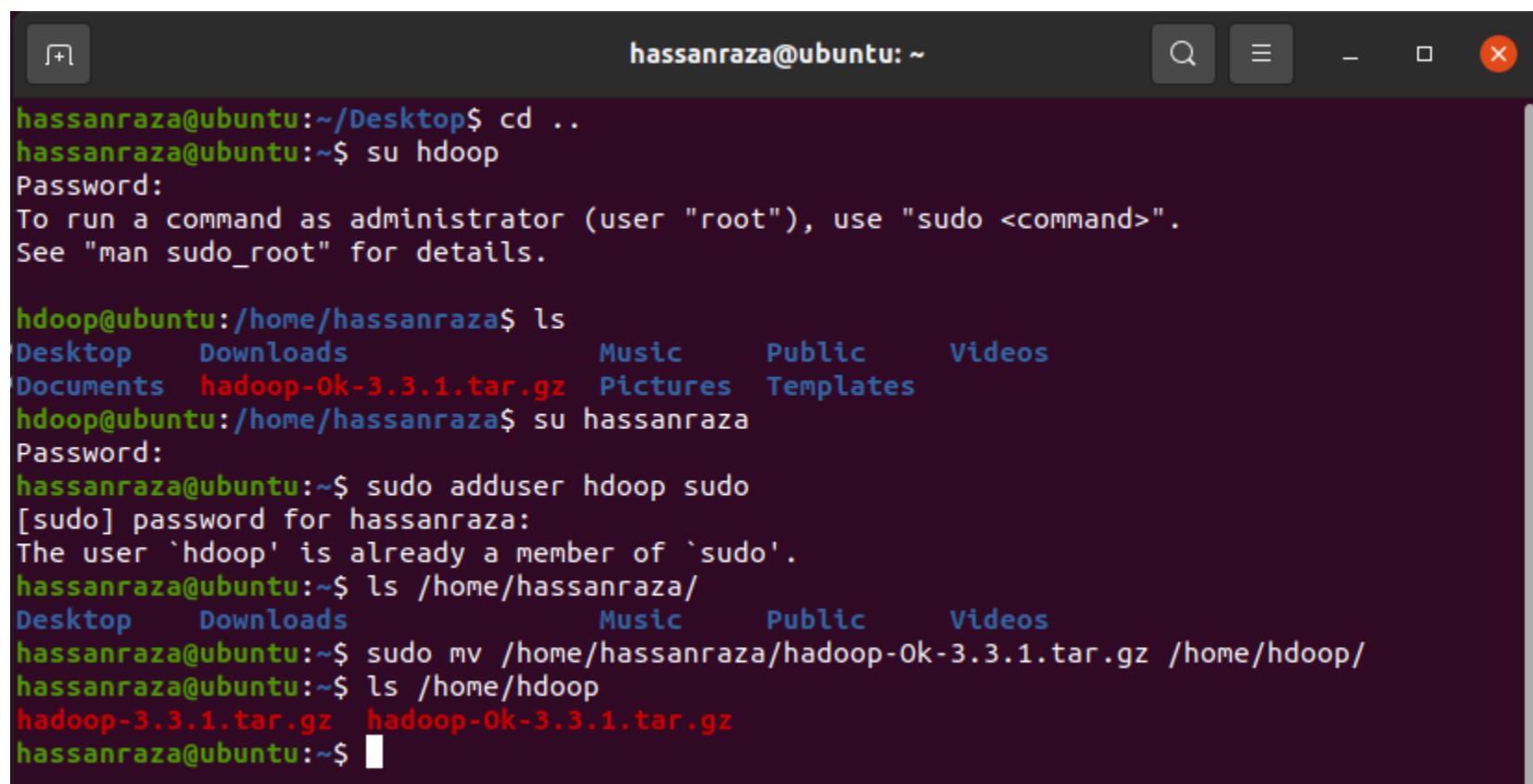
Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

First Way:

If you have already download Hadoop file. Want to install into Hadoop Home Root (hadoop). Then follow these process. The download file placed on Home Directory (hassanraza). Move Directory (hassanraza) to Directory (hadoop).



```
hassanraza@ubuntu:~/Desktop$ cd ..
hassanraza@ubuntu:~$ su hdoop
Password:
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

hdoop@ubuntu:/home/hassanraza$ ls
Desktop Downloads Music Public Videos
Documents hadoop-Ok-3.3.1.tar.gz Pictures Templates
hdoop@ubuntu:/home/hassanraza$ su hassanraza
Password:
hassanraza@ubuntu:~$ sudo adduser hdoop sudo
[sudo] password for hassanraza:
The user 'hdoop' is already a member of 'sudo'.
hassanraza@ubuntu:~$ ls /home/hassanraza/
Desktop Downloads Music Public Videos
hassanraza@ubuntu:~$ sudo mv /home/hassanraza/hadoop-Ok-3.3.1.tar.gz /home/hdoop/
hassanraza@ubuntu:~$ ls /home/hdoop
hadoop-3.3.1.tar.gz hadoop-Ok-3.3.1.tar.gz
hassanraza@ubuntu:~$
```

Second Way:

Then follow these process for Hadoop download Online. The download file placed on Home Directory (hdoop).

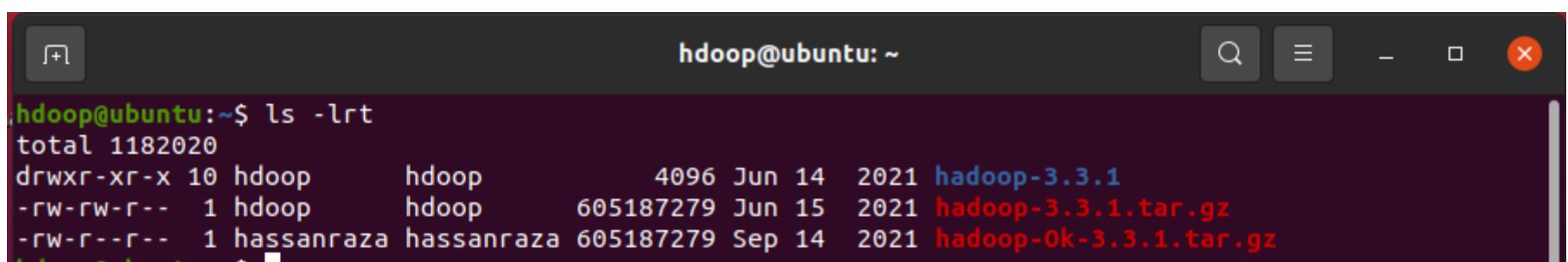


```
hdoop@ubuntu:~$ wget https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
--2022-03-23 22:22:19-- https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
Resolving downloads.apache.org (downloads.apache.org)... 135.181.214.104, 88.99.95.219, 2a01:4f9:3a:2c57::2, ...
Connecting to downloads.apache.org (downloads.apache.org)|135.181.214.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 605187279 (577M) [application/x-gzip]
Saving to: 'hadoop-3.3.1.tar.gz'

hadoop-3.3.1.tar.gz      100%[=====] 577.15M  1.20MB/s   in 15m 23s

2022-03-23 22:37:43 (640 KB/s) - 'hadoop-3.3.1.tar.gz' saved [605187279/605187279]

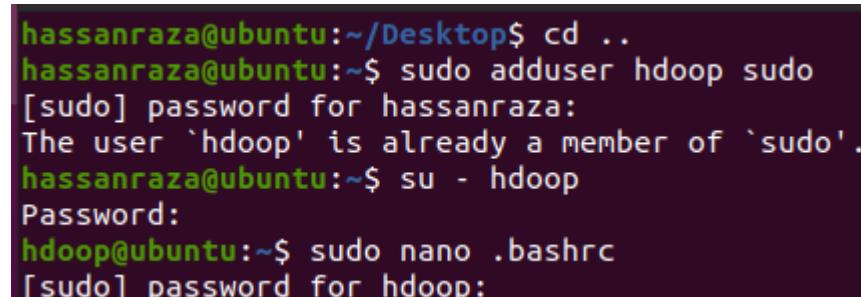
hdoop@ubuntu:~$ ls
hadoop-3.3.1.tar.gz  hadoop-Ok-3.3.1.tar.gz
```



```
hdoop@ubuntu:~$ ls -lrt
total 1182020
drwxr-xr-x 10 hdoop      hdoop          4096 Jun 14  2021 hadoop-3.3.1
-rw-rw-r--  1 hdoop      hdoop        605187279 Jun 15  2021 hadoop-3.3.1.tar.gz
-rw-r--r--  1 hassanraza hassanraza  605187279 Sep 14  2021 hadoop-Ok-3.3.1.tar.gz
```

Allow the “sudo” permission to Hadoop Home Root (hdoop). Then edit the “.bashrc” file for Global/Local paths variables. Here export Hadoop and Java paths (installation and access location etc). Assign the code at end of file.

Some changing into file, then these changing make available for execution to OS Kernel. Using “source ~/.bashrc”



```
hassanraza@ubuntu:~/Desktop$ cd ..
hassanraza@ubuntu:~$ sudo adduser hdoop sudo
[sudo] password for hassanraza:
The user 'hdoop' is already a member of 'sudo'.
hassanraza@ubuntu:~$ su - hdoop
Password:
hdoop@ubuntu:~$ sudo nano .bashrc
[sudo] password for hdoop:
```

```
# Hadoop Related Options
export HADOOP_HOME=/home/hadoop/hadoop-3.3.1
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
```

```
hadoop@ubuntu:~$ source ~/.bashrc
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
```

Add the path of Java JDK (JAVA_HOME) into Hadoop Environment shell file.

```
hadoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
GNU nano 4.8 /home/hadoop/hadoop-3.3.1/etc/hadoop/hadoop-env.sh
# It uses the format of (command)_(subcommand)_USER.
#
# For example, to limit who can execute the namenode command,
# export HDFS_NAMENODE_USER=hdfs

#####
# Registry DNS specific parameters
#####
# For privileged registry DNS, user to run as after dropping privileges
# This will replace the hadoop.id.str Java property in secure mode.
# export HADOOP_REGISTRYDNS_SECURE_USER=yarn

# Supplemental options for privileged registry DNS
# By default, Hadoop uses jsvc which needs to know to launch a
# server jvm.
# export HADOOP_REGISTRYDNS_SECURE_EXTRA_OPTS="-jvm server"

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

^G Get Help **^O** Write Out **^W** Where Is **^K** Cut Text **^J** Justify **^C** Cur Pos
^X Exit **^R** Read File **^V** Replace **^U** Paste Text **^T** To Spell **^** Go To Line

1: Core-Site.xml

```
hadoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml
GNU nano 4.8 /home/hadoop/hadoop-3.3.1/etc/hadoop/core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>/home/hadoop/tmpdata</value>
        <description>A base for other temporary directories.</description>
    </property>
    <property>
        <name>fs.default.name</name>
        <value>hdfs://localhost:9000</value>
        <description>The name of the default file system. A URI who is</description>
    </property>
</configuration>
```

^G Get Help **^O** Write Out **^W** Where Is **^K** Cut Text **^J** Justify **^C** Cur Pos
^X Exit **^R** Read File **^V** Replace **^U** Paste Text **^T** To Spell **^** Go To Line

2: hdfs-site.xml

```
hdoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml

GNU nano 4.8          /home/hdoop/hadoop-3.3.1/etc/hadoop/hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
    <property>
        <name>dfs.data.dir</name>
        <value>/home/hdoop/dfsdata/namenode</value>
    </property>

    <property>
        <name>dfs.data.dir</name>
        <value>/home/hdoop/dfsdata/datanode</value>
    </property>

    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
</configuration>

^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify      ^C Cur Pos
^X Exit         ^R Read File     ^\ Replace       ^U Paste Text    ^T To Spell     ^_ Go To Line
```

3: mapred-site.xml

```
hdoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml

GNU nano 4.8          /home/hdoop/hadoop-3.3.1/etc/hadoop/mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

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

^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify      ^C Cur Pos
^X Exit         ^R Read File     ^\ Replace       ^U Paste Text    ^T To Spell     ^_ Go To Line
```

4: yarn-site.xml

The screenshot shows a terminal window titled "hadoop@ubuntu: ~". The command "sudo nano \$HADOOP_HOME/etc/hadoop/yarn-site.xml" is run. The file content is an XML configuration for YARN. It includes a license notice, several property definitions for nodemanager aux-services, and a configuration section for environment variables. The nano editor's status bar at the bottom shows various keyboard shortcuts.

```
hadoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml
GNU nano 4.8          /home/hadoop/hadoop-3.3.1/etc/hadoop/yarn-site.xml
<?xml version="1.0"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
-->
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  </property>
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>127.0.0.1</value>
  </property>
  <property>
    <name>yarn.acl.enable</name>
    <value>0</value>
  </property>
  <property>
    <name>yarn.nodemanager.env-whitelist</name>
    <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,CLASSPATH,>
  </property>
  <!-- <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,
        CLASSPATH_PERPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_MAPRED_HOME</value>
  -->
</configuration>
```

Open the SSH Local-Host. Make the format of Hadoop (XML) files. That you have modified change according to yourself.

The screenshot shows a terminal window titled "hadoop@ubuntu: ~". The command "ssh localhost" is run, connecting to the local host. The user then runs "hdfs namenode -format". The output shows the NameNode starting up and performing a format operation, creating a new fsimage and fsmeta files. The log concludes with the shutdown message.

```
hadoop@ubuntu:~$ ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

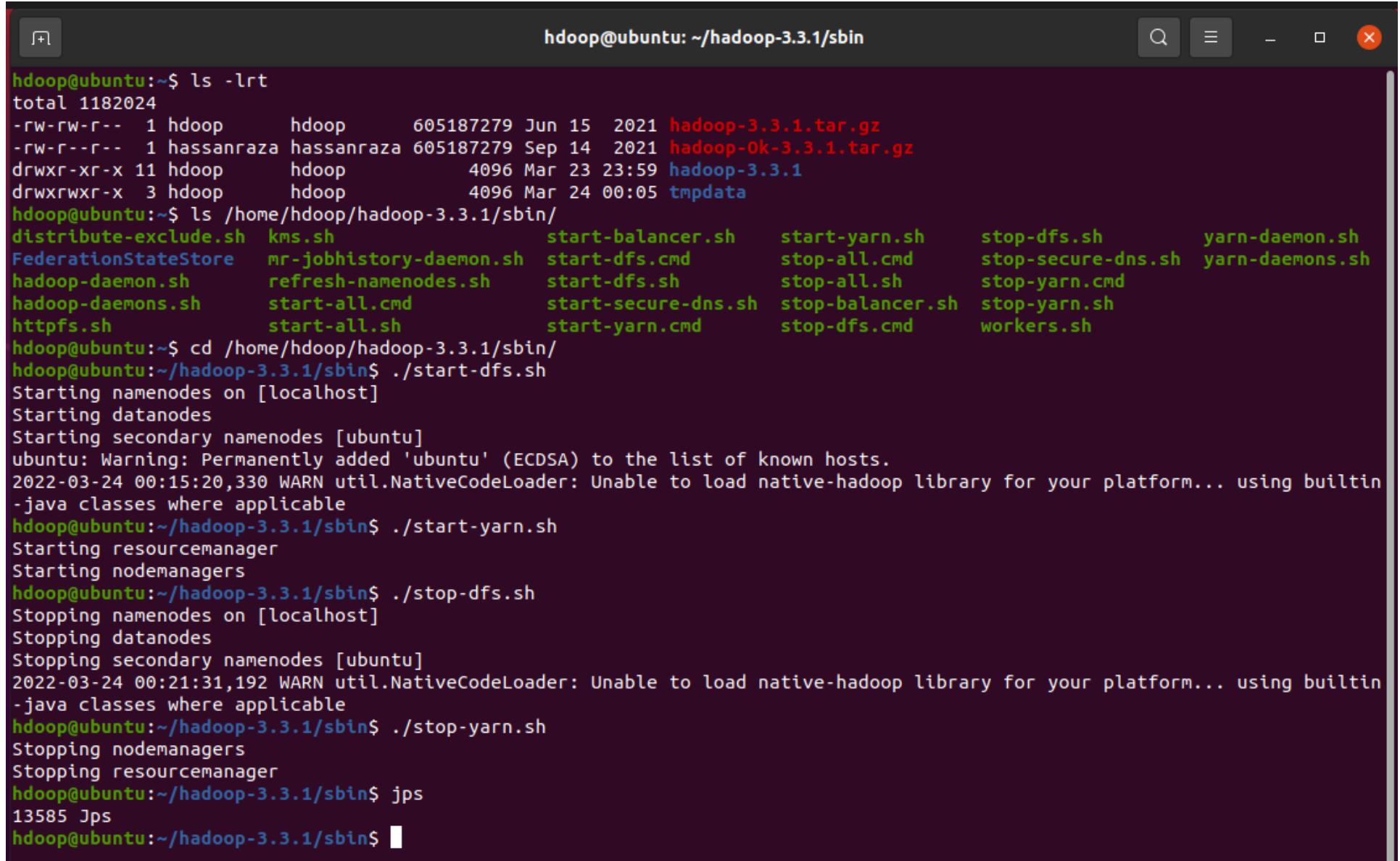
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

246 updates can be applied immediately.
163 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Thu Mar 24 00:00:24 2022 from 127.0.0.1
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~$ hdfs namenode -format
2022-03-24 00:04:54,667 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = ubuntu/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.3.1
STARTUP_MSG:   classpath = /home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/jetty-webap

2022-03-24 00:05:02,106 INFO namenode.FSImageFormatProtobuf: Saving image file /home/hadoop/tmpdata/dfs/name/current/fsimage.ckpt_00000000000000000000 using no compression
2022-03-24 00:05:02,917 INFO namenode.FSImageFormatProtobuf: Image file /home/hadoop/tmpdata/dfs/name/current/fsimage.ckpt_00000000000000000000 of size 400 bytes saved in 0 seconds .
2022-03-24 00:05:03,028 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2022-03-24 00:05:03,131 INFO namenode.FSNamesystem: Stopping services started for active state
2022-03-24 00:05:03,131 INFO namenode.FSNamesystem: Stopping services started for standby state
2022-03-24 00:05:03,195 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2022-03-24 00:05:03,201 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at ubuntu/127.0.1.1
```

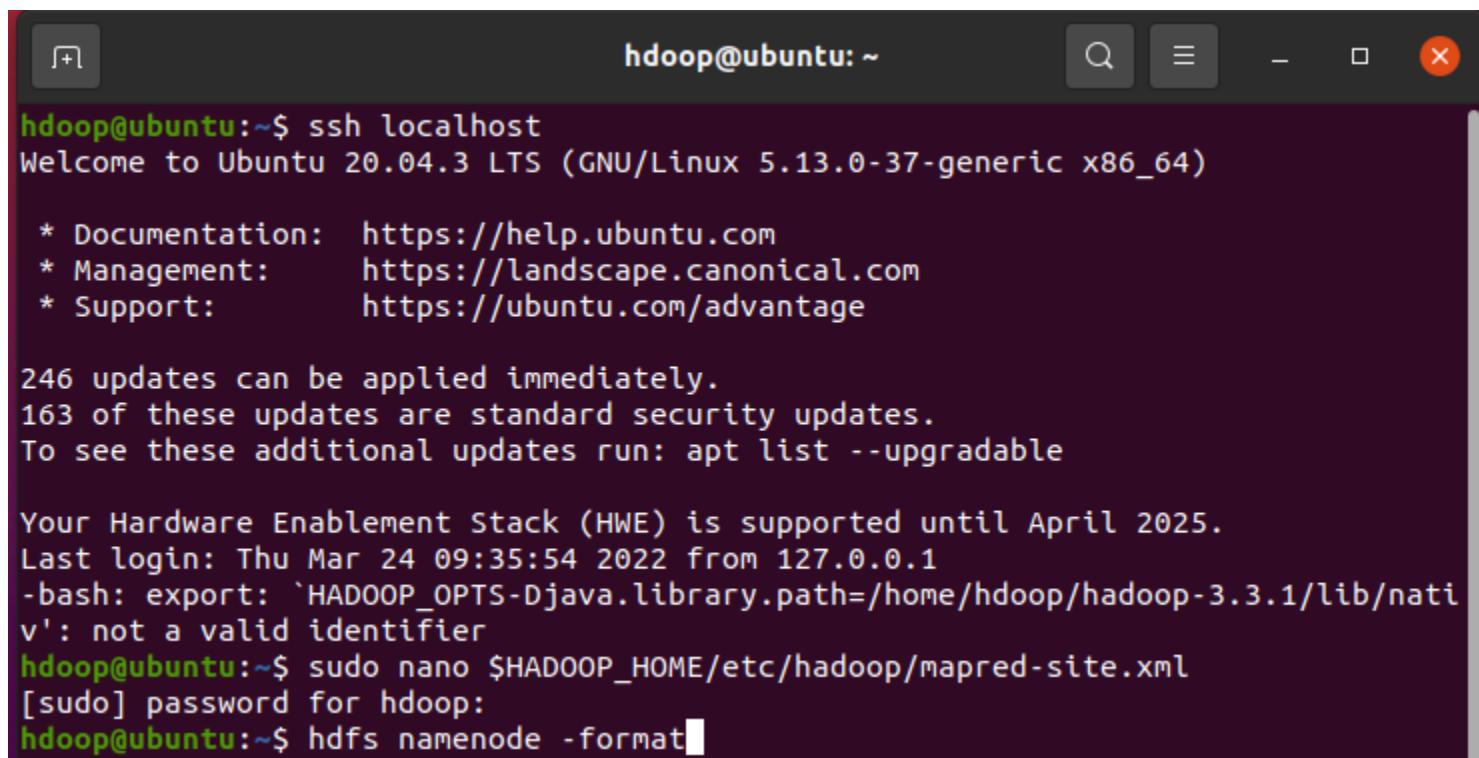
Change the root of Hadoop “hadoop-3.3.1/sbin”. Start/Stop the services of Hadoop Distributed Files and Yarn.



```
hadoop@ubuntu:~$ ls -lrt
total 1182024
-rw-rw-r-- 1 hadoop      hadoop      605187279 Jun 15 2021 hadoop-3.3.1.tar.gz
-rw-r--r-- 1 hassanraza hassanraza 605187279 Sep 14 2021 hadoop-Ok-3.3.1.tar.gz
drwxr-xr-x 11 hadoop      hadoop      4096 Mar 23 23:59 hadoop-3.3.1
drwxrwxr-x  3 hadoop      hadoop      4096 Mar 24 00:05 tmpdata
hadoop@ubuntu:~$ ls /home/hadoop/hadoop-3.3.1/sbin/
distribute-exclude.sh  kms.sh          start-balancer.sh   start-yarn.sh      stop-dfs.sh      yarn-daemon.sh
FederationStateStore   mr-jobhistory-daemon.sh  start-dfs.cmd    stop-all.cmd    stop-secure-dns.sh  yarn-daemons.sh
hadoop-daemon.sh       refresh-namenodes.sh  start-dfs.sh     stop-all.sh     stop-yarn.cmd
hadoop-daemons.sh     start-all.cmd      start-secure-dns.sh  stop-balancer.sh  stop-yarn.sh
httpfs.sh              start-all.sh      start-yarn.cmd    stop-dfs.cmd    workers.sh
hadoop@ubuntu:~$ cd /home/hadoop/hadoop-3.3.1/sbin/
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ ./start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu]
ubuntu: Warning: Permanently added 'ubuntu' (ECDSA) to the list of known hosts.
2022-03-24 00:15:20,330 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin
-java classes where applicable
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ ./start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ ./stop-dfs.sh
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu]
2022-03-24 00:21:31,192 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin
-java classes where applicable
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ ./stop-yarn.sh
Stopping nodemanagers
Stopping resourcemanager
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ jps
13585 Jps
hadoop@ubuntu:~/hadoop-3.3.1/sbin$
```

Local Host Web Server Access

Open the new terminal then run “su – hadoop” and follow these process.



```
hadoop@ubuntu:~$ ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

246 updates can be applied immediately.
163 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Thu Mar 24 09:35:54 2022 from 127.0.0.1
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/nati
v': not a valid identifier
hadoop@ubuntu:~$ sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml
[sudo] password for hadoop:
hadoop@ubuntu:~$ hdfs namenode -format
```

Just add the new “Property” in “mapred-site.xml” file.

```
<property>
  <name>mapred.job.tracker</name>
  <description>The root and host start of Mapreduce Job Tracker at.</description>
</property>
```

```

GNU nano 4.8 /home/hadoop/hadoop-3.3.1/etc/hadoop/mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
    <property>
        <name>mapred.job.tracker</name>
        <description>The host and port that the MapReduce job tracker runs on.</description>
    </property>
</configuration>

```

```

hadoop@ubuntu:~$ ls
dfsdata  hadoop-3.3.1  hadoop-3.3.1.tar.gz  hadoop-0k-3.3.1.tar.gz  tmpdata
hadoop@ubuntu:~$ ls hadoop-3.3.1/sbin/
distribute-exclude.sh      start-all.sh          stop-balancer.sh
FederationStateStore       start-balancer.sh    stop-dfs.cmd
hadoop-daemon.sh          start-dfs.cmd       stop-dfs.sh
hadoop-daemons.sh         start-dfs.sh        stop-secure-dns.sh
httpfs.sh                  start-secure-dns.sh stop-yarn.cmd
kms.sh                     start-yarn.cmd     stop-yarn.sh
mr-jobhistory-daemon.sh   start-yarn.sh      workers.sh
refresh-namenodes.sh      stop-all.cmd       yarn-daemon.sh
start-all.cmd              stop-all.sh        yarn-daemons.sh
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin/
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ ./start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu]
2022-03-24 09:53:43,460 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ jps
3089 SecondaryNameNode
2786 NameNode
3778 Jps
3465 NodeManager
3339 ResourceManager
hadoop@ubuntu:~/hadoop-3.3.1/sbin$
```

Overview 'localhost:9000' (✓active)

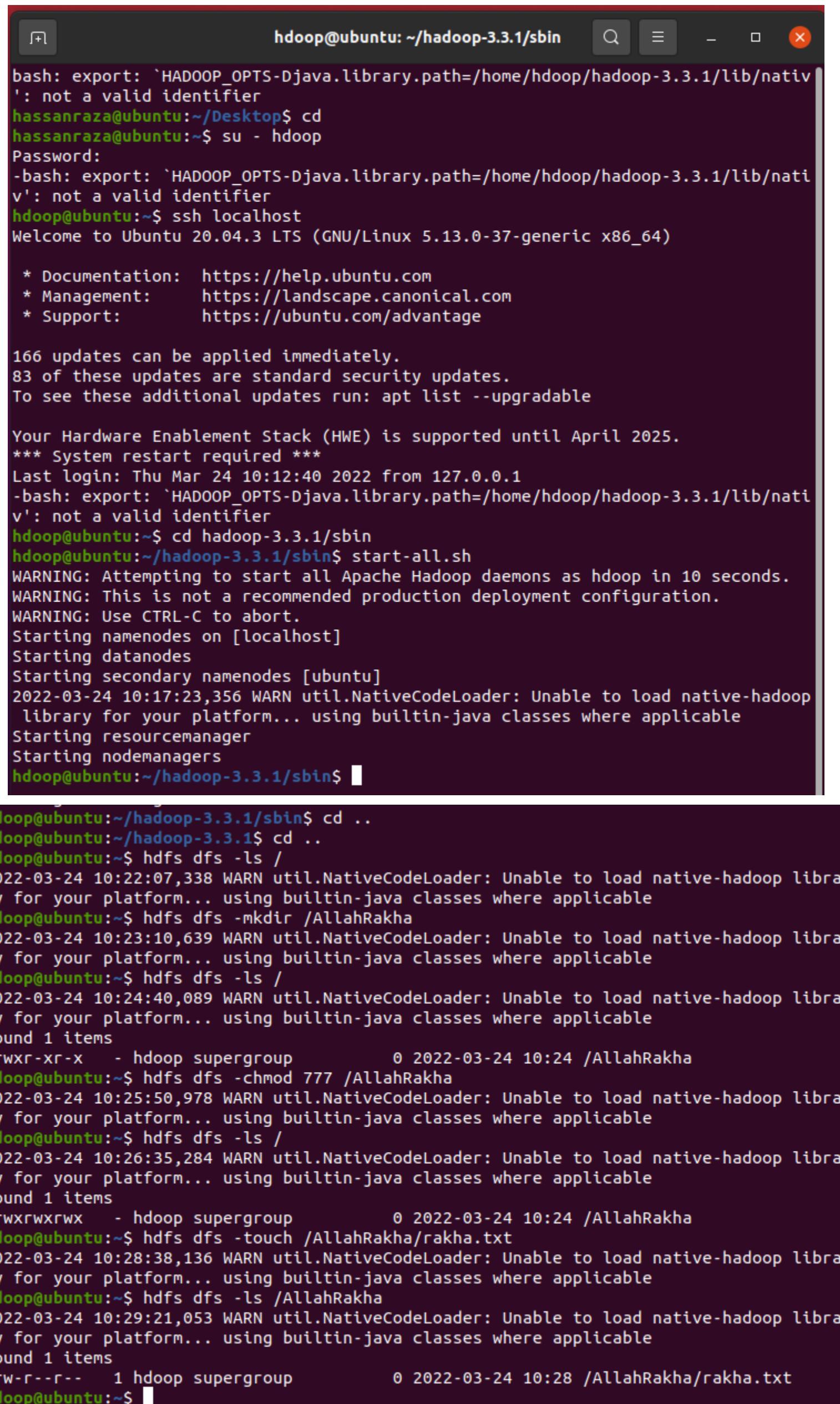
Started:	Thu Mar 24 09:53:29 -0700 2022
Version:	3.3.1, ra3b9c37a397ad4188041dd80621bdefc46885f2
Compiled:	Mon Jun 14 22:13:00 -0700 2021 by ubuntu from (HEAD detached at release-3.3.1-RC3)
Cluster ID:	CID-46018362-c532-46cc-9ab8-657309e6aa17
Block Pool ID:	BP-878120081-127.0.1.1-1648140375419

HADOOP COMMANDS

Open the new terminal and then follow these process of commands.

Here, the all services start of Hadoop. Such as HDFS, YARN and MapReduce etc.

Folder, File create in Hadoop then change permission of folder.



A screenshot of a terminal window titled "hadoop@ubuntu: ~/hadoop-3.3.1/sbin". The terminal displays a series of Linux shell commands and their outputs. It starts with some export errors, then logs into the system, shows system status, and runs the "start-all.sh" script to start Hadoop daemons. It then lists files in the HDFS root directory, creates a file named "rakha.txt", and lists it again. The terminal has a dark background with light-colored text and standard Linux-style icons at the top.

```
bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hassanraza@ubuntu:~/Desktop$ cd
hassanraza@ubuntu:~$ su - hdoop
Password:
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hdoop@ubuntu:~$ ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

166 updates can be applied immediately.
83 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
*** System restart required ***
Last login: Thu Mar 24 10:12:40 2022 from 127.0.0.1
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hdoop@ubuntu:~$ cd hadoop-3.3.1/sbin
hdoop@ubuntu:~/hadoop-3.3.1/sbin$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hdoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu]
2022-03-24 10:17:23,356 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
hdoop@ubuntu:~/hadoop-3.3.1/sbin$ 

hdoop@ubuntu:~/hadoop-3.3.1/sbin$ cd ..
hdoop@ubuntu:~/hadoop-3.3.1$ cd ..
hdoop@ubuntu:~$ hdfs dfs -ls /
2022-03-24 10:22:07,338 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hdoop@ubuntu:~$ hdfs dfs -mkdir /AllahRakha
2022-03-24 10:23:10,639 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hdoop@ubuntu:~$ hdfs dfs -ls /
2022-03-24 10:24:40,089 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x - hdoop supergroup 0 2022-03-24 10:24 /AllahRakha
hdoop@ubuntu:~$ hdfs dfs -chmod 777 /AllahRakha
2022-03-24 10:25:50,978 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hdoop@ubuntu:~$ hdfs dfs -ls /
2022-03-24 10:26:35,284 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
drwxrwxrwx - hdoop supergroup 0 2022-03-24 10:24 /AllahRakha
hdoop@ubuntu:~$ hdfs dfs -touch /AllahRakha/rakha.txt
2022-03-24 10:28:38,136 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hdoop@ubuntu:~$ hdfs dfs -ls /AllahRakha
2022-03-24 10:29:21,053 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r-- 1 hdoop supergroup 0 2022-03-24 10:28 /AllahRakha/rakha.txt
hdoop@ubuntu:~$ 
```

Get the Hadoop file from server “/AllahRakha” directory into your client local machine the root directory (hadoop).

Put your client local machine the root directory (hadoop) into Hadoop server “/AllahRakha” directory.

```

hadoop@ubuntu:~$ hdfs dfs -get /AllahRakha/rakha.txt /home/hadoop/
2022-03-24 10:40:49,953 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ ls -lrt
total 1182028
-rw-rw-r-- 1 hdoop      hdoop      605187279 Jun 15 2021 hadoop-3.3.1.tar.gz
-rw-r--r-- 1 hassanraza hassanraza 605187279 Sep 14 2021 hadoop-0k-3.3.1.tar.gz
drwxr-xr-x 11 hdoop      hdoop      4096 Mar 23 23:59 hadoop-3.3.1
drwxrwxr-x  3 hdoop      hdoop      4096 Mar 24 00:14 dfsdta
drwxrwxr-x  4 hdoop      hdoop      4096 Mar 24 00:19 tmpdata
-rw-r--r--  1 hdoop      hdoop      0 Mar 24 10:40 rakha.txt
hadoop@ubuntu:~$ sudo gedit allah.txt
[sudo] password for hadoop:
Unable to init server: Could not connect: Connection refused

(gedit:62980): Gtk-WARNING **: 10:41:22.990: cannot open display:
hadoop@ubuntu:~$ sudo nano allah.txt
hadoop@ubuntu:~$ cat allah.txt
Allah is the great.....

```

```

hadoop@ubuntu:~$ ls
allah.txt  hadoop-3.3.1      hadoop-0k-3.3.1.tar.gz  tmpdata
dfsdta    hadoop-3.3.1.tar.gz  rakha.txt
hadoop@ubuntu:~$ hdfs dfs -put rakha.txt /AllahRakh
2022-03-24 11:58:56,884 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -ls /AllahRakh
2022-03-24 11:59:12,128 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r--  1 hdoop  supergroup          0 2022-03-24 11:58 /AllahRakh/rakha.txt

```

Browse Directory

/									Go!				
		Show 25 entries								Search:			
<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name					
<input type="checkbox"/>	drwxr-xr-x	hdoop	supergroup	0 B	Mar 24 11:58	0	0 B	AllahRakh					
Showing 1 to 1 of 1 entries													
Previous 1 Next													

Browse Directory

/AllahRakh									Go!				
		Show 25 entries								Search:			
<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name					
<input type="checkbox"/>	-rw-r--r--	hdoop	supergroup	0 B	Mar 24 11:58	1	128 MB	rakha.txt					
Showing 1 to 1 of 1 entries													
Previous 1 Next													

```

hadoop@ubuntu:~$ ls
allah.txt  hadoop-3.3.1      hadoop-0k-3.3.1.tar.gz  tmpdata
dfsdta    hadoop-3.3.1.tar.gz  rakha.txt
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as hdoop in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu]
2022-03-24 12:07:59,235 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Stopping nodemanagers
Stopping resourcemanager
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ cd
hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-24 12:08:23,263 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
ls: Call From ubuntu/127.0.1.1 to localhost:9000 failed on connection exception:
java.net.ConnectException: Connection refused; For more details see: http://wiki.apache.org/hadoop/ConnectionRefused
hadoop@ubuntu:~$ 

```

PUT COMMAND ERROR AND RESOLVE (HADOOP FILE)

If any case this error appear at end of “put” command run in Hadoop. The Follow these process of commands to resolve error.

```
put: File /count_inputfile._COPYING_ could only be replicated to 0 nodes instead of minReplication (=1). There are 0 datanode(s) running and no node(s) are excluded in this operation.

hadoop@ubuntu:~$ cd
hadoop@ubuntu:~$ su - hdoop
Password:
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hdoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~/Desktop$ cd
hadoop@ubuntu:~$ ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

86 updates can be applied immediately.
3 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Fri Mar 25 02:23:00 2022 from 127.0.0.1
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hdoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~$ ls
allah.txt  hadoop-3.3.1      hadoop-Ok-3.3.1.tar.gz  rakha.txt
dfsdata    hadoop-3.3.1.tar.gz Input.txt            tmpdata
hadoop@ubuntu:~$ ls tmpdata/*
tmpdata/dfs:
name  namenode

tmpdata/nm-local-dir:
filecache  nmPrivate  usercache
hadoop@ubuntu:~$
```

Here we create “namenode” directory in “dfsdata” folder.

```
hadoop@ubuntu:~$ ls
allah.txt  hadoop-3.3.1      hadoop-Ok-3.3.1.tar.gz  rakha.txt
dfsdata    hadoop-3.3.1.tar.gz Input.txt            tmpdata
hadoop@ubuntu:~$ ls dfsdata
datanode
hadoop@ubuntu:~$ sudo mkdir dfsdata/namenode
hadoop@ubuntu:~$ ls dfsdata/
datanode  namenode
hadoop@ubuntu:~$
```

Main Import Part. Just remove all files and folder in “tmpdata” & “dfsdata/datanode”. Because new Hadoop installation in which “tmpdata” & “dfsdata/datanode” are store buffer record. The not accessible to “put” command into Hadoop Root Directory “/”.

```
hadoop@ubuntu:~$ sudo rm -R tmpdata/*
[sudo] password for hadoop:
hadoop@ubuntu:~$ ls tmpdata/*
ls: cannot access 'tmpdata/*': No such file or directory
hadoop@ubuntu:~$ ls dfsdata/*
dfsdata/datanode:
current

dfsdata/namenode:
hadoop@ubuntu:~$ ls dfsdata/datanode/*
BP-645937097-127.0.1.1-1648205614081  VERSION
hadoop@ubuntu:~$ ls dfsdata/datanode/
current
hadoop@ubuntu:~$ sudo rm -R dfsdata/datanode/*
hadoop@ubuntu:~$ ls dfsdata/datanode/
hadoop@ubuntu:~$ ls dfsdata/datanode/*
ls: cannot access 'dfsdata/datanode/*': No such file or directory
hadoop@ubuntu:~$
```

Point: The “namenode” & “datanode” format is most import after remove some directory. Because, previous “cache” record delete and then create new empty record.

```
hadoop@ubuntu:~$ hdfs namenode -format
2022-03-25 04:13:49,903 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG: host = ubuntu/127.0.1.1
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.3.1
STARTUP_MSG: classpath = /home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/iettv-webapp-9.4.40.v20210413.jar:/home/hadoop/hadoo
```

```
hadoop@ubuntu:~$ hdfs datanode -format
2022-03-25 04:14:27,434 INFO datanode.DataNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting DataNode
STARTUP_MSG: host = ubuntu/127.0.1.1
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.3.1
STARTUP_MSG: classpath = /home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/iettv-webapp-9.4.40.v20210413.jar:/home/hadoop/hadoo
```

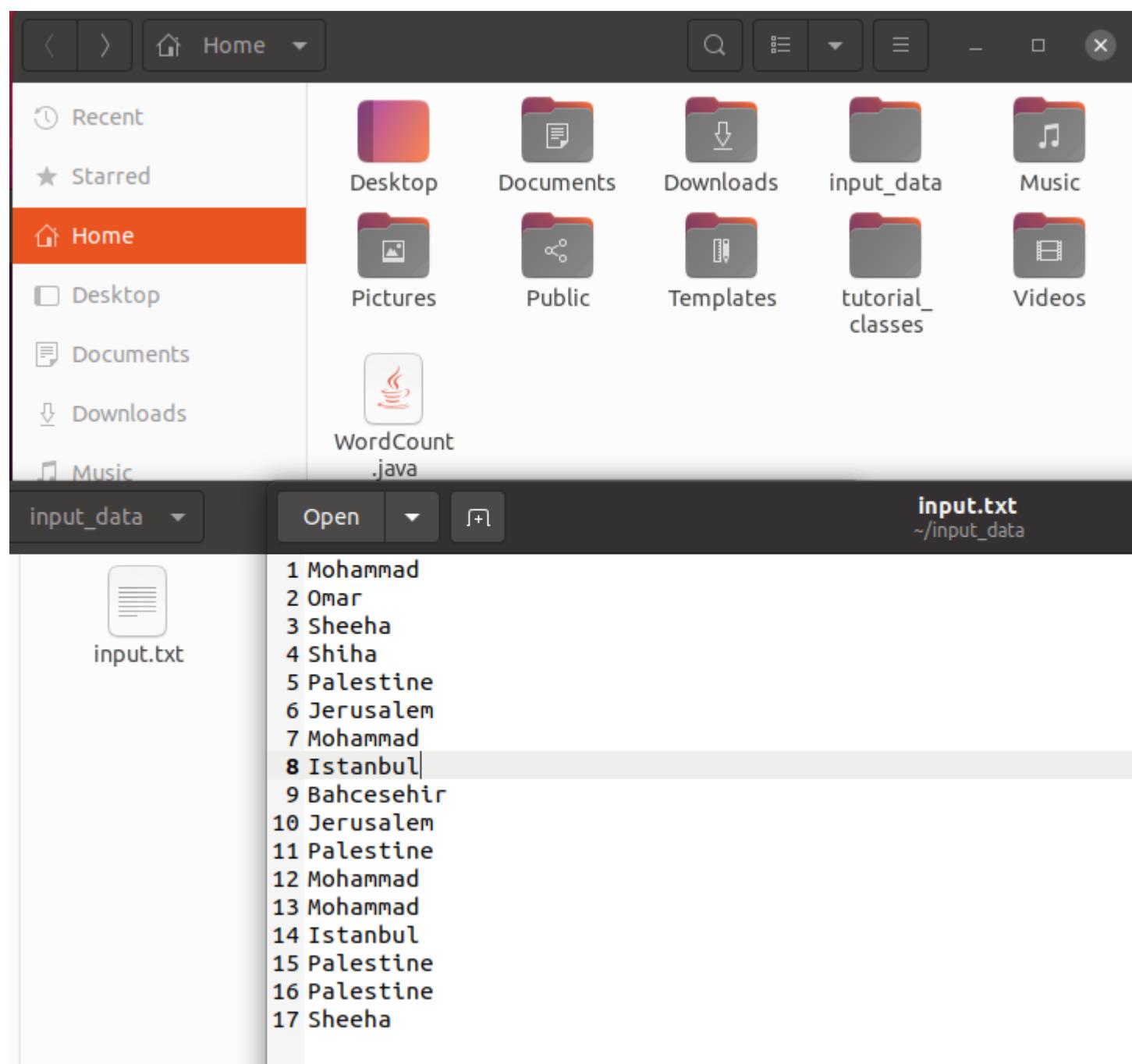
```
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu]
2022-03-25 04:15:49,972 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ cd
hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-25 04:16:17,229 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$
```

Remember: Here, we create a new file “allah.txt” in Hadoop Home Root Directory. Then put into Hadoop Server on Root Directory. Then create folder “/AllahRakhaRaza” and in which put “allah.txt” file.

```
hadoop@ubuntu:~$ ls -lrt
total 1182028
-rw-rw-r-- 1 hadoop      hadoop      605187279 Jun 15 2021 hadoop-3.3.1.tar.gz
-rw-r--r-- 1 hassanraza hassanraza 605187279 Sep 14 2021 hadoop-0k-3.3.1.tar.gz
drwxrwxrwx 11 hadoop      hadoop      4096 Mar 23 23:59 hadoop-3.3.1
drwxrwxr-x  4 hadoop      hadoop      4096 Mar 25 04:13 dfsdata
drwxrwxr-x  4 hadoop      hadoop      4096 Mar 25 04:15 tmpdata
-rw-r--r-- 1 root        root        0 Mar 25 04:17 allah.txt
hadoop@ubuntu:~$ hdfs dfs -put allah.txt /
2022-03-25 04:18:52,396 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-25 04:19:05,075 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r-- 1 hadoop supergroup      0 2022-03-25 04:18 /allah.txt
hadoop@ubuntu:~$ hdfs dfs -mkdir /AllahRakhaRaza
2022-03-25 04:19:31,649 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -put allah.txt /AllahRakhaRaza/
2022-03-25 04:19:54,606 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-25 04:20:05,278 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x - hadoop supergroup      0 2022-03-25 04:19 /AllahRakhaRaza
-rw-r--r--  1 hadoop supergroup      0 2022-03-25 04:18 /allah.txt
hadoop@ubuntu:~$ hdfs dfs -ls /AllahRakhaRaza
2022-03-25 04:20:25,419 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r--  1 hadoop supergroup      0 2022-03-25 04:19 /AllahRakhaRaza/allah.txt
hadoop@ubuntu:~$
```

WORD COUNTS EXAMPLE

Create the two folder “input_data” & “tutorial_classes”. Then paste “WordCount.java” file in “/home/hassanraza” directory. File create “Input.txt” in which folder of “input_data”.



Add “export” the hadoop classpath to global/local variable of HADOOP_CLASSPATH and then echo or cat for testing.

```
hadoop@ubuntu:~$ bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~/Desktop$ cd
hadoop@ubuntu:~$ su -
Password:
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~$ ls
allah.txt  hadoop-3.3.1          hadoop-0k-3.3.1.tar.gz  tmpdata
dfsdata    hadoop-3.3.1.tar.gz   rakha.txt
hadoop@ubuntu:~$ echo HADOOP_CLASSPATH
HADOOP_CLASSPATH
hadoop@ubuntu:~$ echo HADOOP_CLASSPATH=$(hadoop classpath)
HADOOP_CLASSPATH=/home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/common/*:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs/*:/home/hadoop/hadoop-3.3.1/share/hadoop/mapreduce/*:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn/*
hadoop@ubuntu:~$ export HADOOP_CLASSPATH=$(hadoop classpath)
hadoop@ubuntu:~$ echo $HADOOP_CLASSPATH
/home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/common/*:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs/*:/home/hadoop/hadoop-3.3.1/share/hadoop/mapreduce/*:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn/lib/*:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn/*
hadoop@ubuntu:~$
```

```

hadoop@ubuntu:~$ ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

83 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Thu Mar 24 11:48:46 2022 from 127.0.0.1
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/native': not a valid identifier
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu]
2022-03-24 15:36:04,281 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ cd
hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-24 15:36:42,645 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x  - hadoop supergroup          0 2022-03-24 11:58 /AllahRakh

```

Create “WordCounts/Input” directory in Hadoop. Then put file of “input.txt”.

```

input_data
input.txt

hadoop@ubuntu:~$ hdfs dfs -ls /
2022-03-25 11:36:36,074 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x  - hadoop supergroup          0 2022-03-25 04:19 /AllahRakhaRaza
-rw-r--r--  1 hadoop supergroup          0 2022-03-25 04:18 /allah.txt
hadoop@ubuntu:~$ hdfs dfs -mkdir /WordCounts
2022-03-25 11:37:04,699 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -mkdir /WordCounts/Input
2022-03-25 11:37:23,482 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -ls /WordCounts
2022-03-25 11:37:34,538 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x  - hadoop supergroup          0 2022-03-25 11:37 /WordCounts/Input
hadoop@ubuntu:~$ hdfs dfs -put '/home/hassanraza/input_data/input.txt' /WordCounts/Input
2022-03-25 11:38:48,566 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
hadoop@ubuntu:~$ hdfs dfs -ls /WordCounts/Input/
2022-03-25 11:39:03,347 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r--  1 hadoop supergroup         150 2022-03-25 11:38 /WordCounts/Input/input.txt
hadoop@ubuntu:~$ 

```

Change the Mode of file and folder (WordCount.java, tutorial_classes) with read_write_execute permission.

```

hadoop@ubuntu:~$ export HADOOP_CLASSPATH=$(hadoop classpath)
hadoop@ubuntu:~$ echo $HADOOP_CLASSPATH
/home/hadoop/hadoop-3.3.1/etc/hadoop:/home/hadoop/hadoop-3.3.1/share/hadoop/common/lib/*
:/home/hadoop/hadoop-3.3.1/share/hadoop/common/*:/home/hadoop/hadoop-3.3.1/share/hadoop/
hdfs:/home/hadoop/hadoop-3.3.1/share/hadoop/hdfs/lib/*:/home/hadoop/hadoop-3.3.1/share/h
adoop/hdfs/*:/home/hadoop/hadoop-3.3.1/share/hadoop/mapreduce/*:/home/hadoop/hadoop-3.3.
1/share/hadoop/yarn:/home/hadoop/hadoop-3.3.1/share/hadoop/yarn/lib/*:/home/hadoop/hadoo
p-3.3.1/share/hadoop/yarn/*
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/WordCount.java'
-rw-r--r-- 1 hassanraza hassanraza 2148 Mar 24 19:48 /home/hassanraza/WordCount.java
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/WordCount.java'
[sudo] password for hadoop:
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/WordCount.java'
-rwxrwxrwx 1 hassanraza hassanraza 2148 Mar 24 19:48 /home/hassanraza/WordCount.java
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/tutorial_classes'
total 0
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/tutorial_classes'

```

Add all privilege user specification permission to “hadoop” user.

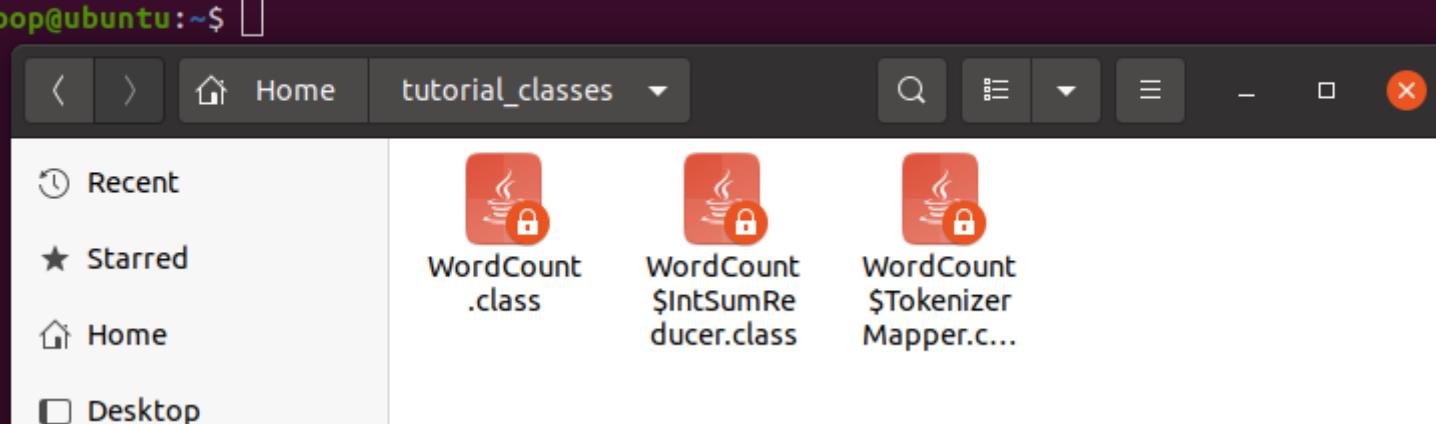
```
hadoop@ubuntu:~$ sudo nano /etc/sudoers
[REDACTED]
GNU nano 4.8                               /etc/sudoers
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
#
# Host alias specification
#
# User alias specification
#
# Cmnd alias specification
#
# User privilege specification
root    ALL=(ALL:ALL) ALL
hadoop  ALL=(ALL:ALL) ALL
# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
#includeif /etc/sudoers.d
```

Create “Word Counts” some files to counting a words in “tutorial_classes” folder. Here, HADOOP_CLASSPATH used for API access to create these files. First path use for Output files (tutorial_classes) and Second path use for Input file (WordCount.java). “-d” here directory path show. Here, “-classpath” to assign “HADOOP_CLASSPATH” all list of paths that help out for “javac” execution.

```
hadoop@ubuntu:~$ javac -classpath ${HADOOP_CLASSPATH} -d '/home/hassanraza/tutorial_classes'
  '/home/hassanraza/WordCount.java'
hadoop@ubuntu:~$
```



Then change permission of all files “tutorial_classes” folder. Assign read_write_execute permission to files.

Command: jar -cvf firstWordCount.jar -C “/home/hassanraza/tutorial_classes/” .

- Jar –cvf : used for compressed all files into single file. Location current directory root “/home/hadoop”.
- -C : use to access all files for compressing.
- “/home/hassanraza/tutorial_classes/” : Path (provide all files to compressing into single file).

Then copy file “/home/hadoop/” to “/home/hassanraza/” directory root.

Then change permission of “firstWordCount.jar” file. Assign read_write_execute permission.

Then open “ssh localhost” and start all services of Hadoop.

```
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/tutorial_classes/'  
total 12  
-rw-rw-r-- 1 hadoop hadoop 1736 Mar 25 12:13 'WordCount$TokenizerMapper.class'  
-rw-rw-r-- 1 hadoop hadoop 1739 Mar 25 12:13 'WordCount$IntSumReducer.class'  
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/tutorial_classes/WordCount$TokenizerMapper.class'  
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/tutorial_classes/WordCount$IntSumReducer.class'  
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/tutorial_classes/WordCount.class'  
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/tutorial_classes/'  
total 12  
-rwxrwxrwx 1 hadoop hadoop 1736 Mar 25 12:13 'WordCount$TokenizerMapper.class'  
-rwxrwxrwx 1 hadoop hadoop 1739 Mar 25 12:13 'WordCount$IntSumReducer.class'  
-rwxrwxrwx 1 hadoop hadoop 1491 Mar 25 12:13 WordCount.class  
hadoop@ubuntu:~$ jar -cvf /home/hassanraza/firstWordCount.jar -C '/home/hassanraza/tutorial_classes/' .  
java.io.FileNotFoundException: /home/hassanraza/firstWordCount.jar (Permission denied)  
    at java.io.FileOutputStream.open0(Native Method)  
    at java.io.FileOutputStream.open(FileOutputStream.java:270)  
    at java.io.FileOutputStream.<init>(FileOutputStream.java:213)  
    at java.io.FileOutputStream.<init>(FileOutputStream.java:101)  
    at sun.tools.jar.Main.run(Main.java:195)  
    at sun.tools.jar.Main.main(Main.java:1288)  
hadoop@ubuntu:~$ jar -cvf firstWordCount.jar -C '/home/hassanraza/tutorial_classes/' .  
added manifest  
adding: WordCount.class(in = 1491) (out= 814)(deflated 45%)  
adding: WordCount$TokenizerMapper.class(in = 1736) (out= 754)(deflated 56%)  
adding: WordCount$IntSumReducer.class(in = 1739) (out= 739)(deflated 57%)  
hadoop@ubuntu:~$ ls -lrt  
total 1182040  
-rw-rw-r-- 1 hadoop hadoop 605187279 Jun 15 2021 hadoop-3.3.1.tar.gz  
-rw-r--r-- 1 hassanraza hassanraza 605187279 Sep 14 2021 hadoop-0k-3.3.1.tar.gz  
drwxrwxrwx 11 hadoop hadoop 4096 Mar 23 23:59 hadoop-3.3.1  
drwxrwxr-x 4 hadoop hadoop 4096 Mar 25 04:13 dfsdata  
drwxrwxr-x 4 hadoop hadoop 4096 Mar 25 04:15 tmpdata  
-rw-r--r-- 1 root root 0 Mar 25 04:17 allah.txt  
-rwxrwxrwx 1 root root 2148 Mar 25 11:57 WordCount.java  
drwxrwxrwx 2 root root 4096 Mar 25 12:01 tutorial_classes  
-rw-rw-r-- 1 hadoop hadoop 3069 Mar 25 12:23 firstWordCount.jar  
hadoop@ubuntu:~$ sudo cp firstWordCount.jar /home/hassanraza/  
hadoop@ubuntu:~$ ls /home/hassanraza/  
Desktop Downloads input_data Pictures Templates Videos  
Documents firstWordCount.jar Music Public tutorial_classes WordCount.java  
hadoop@ubuntu:~$  
  
hadoop@ubuntu:~$ sudo chmod 777 '/home/hassanraza/firstWordCount.jar'  
hadoop@ubuntu:~$ ls -lrt '/home/hassanraza/firstWordCount.jar'  
-rwxrwxrwx 1 root root 3069 Mar 25 12:25 /home/hassanraza/firstWordCount.jar  
hadoop@ubuntu:~$ ssh localhost  
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-37-generic x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/advantage  
  
83 updates can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
Your Hardware Enablement Stack (HWE) is supported until April 2025.  
Last login: Fri Mar 25 11:33:58 2022 from 127.0.0.1  
-bash: export: `HADOOP_OPTS-Djava.library.path=/home/hadoop/hadoop-3.3.1/lib/nativ': not a valid identifier  
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin  
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ start-all.sh  
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.  
WARNING: This is not a recommended production deployment configuration.  
WARNING: Use CTRL-C to abort.  
Starting namenodes on [localhost]  
localhost: namenode is running as process 2570. Stop it first and ensure /tmp/hadoop-hadoop-namenode.pid file is empty before retry.  
Starting datanodes  
localhost: datanode is running as process 2697. Stop it first and ensure /tmp/hadoop-hadoop-datanode.pid file is empty before retry.  
Starting secondary namenodes [ubuntu]  
ubuntu: secondarynamenode is running as process 2863. Stop it first and ensure /tmp/hadoop-hadoop-secondarynamenode.pid file is empty before retry.  
2022-03-25 12:29:51,267 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
Starting resourcemanager  
resourcemanager is running as process 3076. Stop it first and ensure /tmp/hadoop-hadoop-resourcemanager.pid file is empty before retry.  
Starting nodemanagers  
localhost: nodemanager is running as process 3203. Stop it first and ensure /tmp/hadoop-hadoop-nodemanager.pid file is empty before retry.  
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ cd  
hadoop@ubuntu:~$ hdfs dfs -ls /  
2022-03-25 12:30:15,701 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
Found 3 items  
drwxr-xr-x - hdoop supergroup 0 2022-03-25 04:19 /AllahRakhaRaza
```

Command: hadoop jar '/home/hassanraza/firstWordCount.jar' WordCount /WordCounts/Input /WordCounts/Output1

- hadoop jar : used for upload “firstWordCount.jar” into Hadoop Server and counting all words from file.
- ‘/home/hassanraza/firstWordCount.jar’ : used path of file that upload into Hadoop server.
- WordCount : class name of file original “WordCount.java”.
- /WordCounts/Input : path used to access “input.txt” file.
- /WordCounts/Output1 : path used to store all output record of relative counts word.

```
hadoop@ubuntu:~$ hadoop jar '/home/hassanraza/firstWordCount.jar' WordCount /WordCounts/Input /WordCounts/Output1
2022-03-25 13:11:42,427 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using
builtin-java classes where applicable
2022-03-25 13:11:43,571 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:
8032
2022-03-25 13:11:44,665 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement
the Tool interface and execute your application with ToolRunner to remedy this.
2022-03-25 13:11:44,723 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/stag
ing/hadoop/.staging/job_1648238963605_0001
2022-03-25 13:11:47,131 INFO input.FileInputFormat: Total input files to process : 1
2022-03-25 13:11:47,328 INFO mapreduce.JobSubmitter: number of splits:1
2022-03-25 13:11:51,635 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1648238963605_0001
2022-03-25 13:11:51,636 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-03-25 13:11:53,882 INFO conf.Configuration: resource-types.xml not found
2022-03-25 13:11:53,885 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-03-25 13:11:54,827 INFO impl.YarnClientImpl: Submitted application application_1648238963605_0001
2022-03-25 13:11:55,181 INFO mapreduce.Job: The url to track the job: http://ubuntu:8088/proxy/application_164823896
3605_0001/
2022-03-25 13:11:55,185 INFO mapreduce.Job: Running job: job_1648238963605_0001
2022-03-25 13:12:23,834 INFO mapreduce.Job: Job job_1648238963605_0001 running in uber mode : false
2022-03-25 13:12:23,835 INFO mapreduce.Job: map 0% reduce 0%
2022-03-25 13:12:39,367 INFO mapreduce.Job: map 100% reduce 0%
2022-03-25 13:12:49,571 INFO mapreduce.Job: map 100% reduce 100%
2022-03-25 13:12:49,588 INFO mapreduce.Job: Job job_1648238963605_0001 completed successfully
2022-03-25 13:12:49,785 INFO mapreduce.Job: Counters: 54
    File System Counters
        FILE: Number of bytes read=121
        FILE: Number of bytes written=544669
        FILE: Number of read operations=0
        FILE: Number of large read operations=0
        FILE: Number of write operations=0
        HDFS: Number of bytes read=263
        HDFS: Number of bytes written=83
        HDFS: Number of read operations=8
        HDFS: Number of large read operations=0
        HDFS: Number of write operations=2
        HDFS: Number of bytes read erasure-coded=0
    Job Counters
        Launched map tasks=1
        Launched reduce tasks=1
```

```
Total vcore-milliseconds taken by all map tasks=12064
Total vcore-milliseconds taken by all reduce tasks=7085
Total megabyte-milliseconds taken by all map tasks=12353536
Total megabyte-milliseconds taken by all reduce tasks=7255040
Map-Reduce Framework
    Map input records=17
    Map output records=17
    Map output bytes=218
    Map output materialized bytes=121
    Input split bytes=113
    Combine input records=17
    Combine output records=8
    Reduce input groups=8
    Reduce shuffle bytes=121
    Reduce input records=8
    Reduce output records=8
    Spilled Records=16
    Shuffled Maps =1
    Failed Shuffles=0
    Merged Map outputs=1
    GC time elapsed (ms)=3045
    CPU time spent (ms)=9080
    Physical memory (bytes) snapshot=665219072
    Virtual memory (bytes) snapshot=5083590656
    Total committed heap usage (bytes)=560988160
    Peak Map Physical memory (bytes)=464707584
    Peak Map Virtual memory (bytes)=2538524672
    Peak Reduce Physical memory (bytes)=200511488
    Peak Reduce Virtual memory (bytes)=2545065984
Shuffle Errors
    BAD_ID=0
    CONNECTION=0
    IO_ERROR=0
    WRONG_LENGTH=0
    WRONG_MAP=0
    WRONG_REDUCE=0
File Input Format Counters
    Bytes Read=150
File Output Format Counters
    Bytes Written=83
```

Show the output of record from “/WordCounts/Output1/*” folder.

```
hadoop@ubuntu:~$ hadoop dfs -cat /WordCounts/Output1/*
WARNING: Use of this script to execute dfs is deprecated.
WARNING: Attempting to execute replacement "hdfs dfs" instead.

2022-03-25 13:25:24,820 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using
builtin-java classes where applicable
Bahcesehir      1
Istanbul        2
Jerusalem       2
Mohammad        4
Omar            1
Palestine       4
Sheeha          2
Shiha           1
hadoop@ubuntu:~$
```

Shut down all services of Hadoop.

```
hadoop@ubuntu:~$ cd hadoop-3.3.1/sbin
hadoop@ubuntu:~/hadoop-3.3.1/sbin$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu]
2022-03-25 13:28:52,403 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using
builtin-java classes where applicable
Stopping nodemanagers
localhost: WARNING: nodemanager did not stop gracefully after 5 seconds: Trying to kill with kill -9
Stopping resourcemanager
hadoop@ubuntu:~/hadoop-3.3.1/sbin$
```