

# DATA ANALYTICS

## ASSIGNMENT 1

Naresh kumar V  
210701323

## HADOOP

### INTRODUCTION:

Hadoop is an open-source software framework that is used for storing and processing large amounts of data in a distributed computing environment. It is designed to handle big data and is based on the MapReduce programming model, which allows for the parallel processing of large datasets.

### HISTORY:

2003	Google released the paper, Google File System (GFS).
2004	Google released a white paper on Map Reduce.
2006	<ul style="list-style-type: none"><li>○ Hadoop introduced.</li><li>○ Hadoop 0.1.0 released.</li><li>○ Yahoo deploys 300 machines and within this year reaches 600 machines.</li></ul>
2007	<ul style="list-style-type: none"><li>○ Yahoo runs 2 clusters of 1000 machines.</li><li>○ Hadoop includes HBase.</li></ul>
2008	<ul style="list-style-type: none"><li>○ YARN JIRA opened</li><li>○ Hadoop becomes the fastest system to sort 1 terabyte of data on a 900 node cluster within 209 seconds.</li><li>○ Yahoo clusters loaded with 10 terabytes per day.</li><li>○ Cloudera was founded as a Hadoop distributor.</li></ul>
2009	<ul style="list-style-type: none"><li>○ Yahoo runs 17 clusters of 24,000 machines.</li><li>○ Hadoop becomes capable enough to sort a petabyte.</li><li>○ MapReduce and HDFS become separate subproject.</li></ul>

### Hardware Requirements

2010	<ul style="list-style-type: none"> <li>○ Hadoop added the support for Kerberos.</li> <li>○ Hadoop operates 4,000 nodes with 40 petabytes.</li> </ul>
	<ul style="list-style-type: none"> <li>○ Apache Hive and Pig released.</li> </ul>
2011	<ul style="list-style-type: none"> <li>○ Apache Zookeeper released.</li> <li>○ Yahoo has 42,000 Hadoop nodes and hundreds of petabytes of storage.</li> </ul>
2012	Apache Hadoop 1.0 version released.
2013	Apache Hadoop 2.2 version released.
2014	Apache Hadoop 2.6 version released.
2015	Apache Hadoop 2.7 version released.
2017	Apache Hadoop 3.0 version released.
2018	Apache Hadoop 3.1 version released.

1. **Memory:**

- At least 8 GB of RAM per machine (16 GB or more is recommended for production environments).

2. **Storage:**

- At least 500 GB of disk space per machine.
- Use high-speed disks (SSD) for better performance.

3. **CPU:** ○ Multi-core processors are recommended. At least 4 cores per machine.

4. **Network:** ○ High bandwidth (1 Gbps or higher) network connection between nodes.

## Software Requirements

### 1. Operating System:

- Linux-based OS (e.g., CentOS, Ubuntu, Debian). ○ Some versions of Hadoop support Windows, but Linux is preferred for production.

### 2. Java:

- Oracle JDK 8 or OpenJDK 8 (Java 8).
- Some versions of Hadoop may support Java 11, but it's crucial to verify compatibility.

### 3. SSH:

- Password-less SSH (Secure Shell) setup for communication between nodes.

### 4. Hadoop Distribution: ○ Latest stable version of Hadoop. You can download it from the [Apache Hadoop website](#).

### 5. Additional Software:

- Python (optional, but recommended for certain Hadoop ecosystem tools).
- Various Hadoop ecosystem components (e.g., HDFS, YARN, MapReduce, Hive, HBase, etc.) as required by your specific use case.

## Configuration Considerations

### 1. Cluster Management:

- Use tools like Apache Ambari, Cloudera Manager, or other cluster management tools for easier setup and maintenance.

### 2. Resource Management:

- Properly configure YARN for resource allocation.
- Set appropriate heap sizes for NameNode and DataNode based on available memory.

### 3. Replication Factor: ○ Set the HDFS replication factor based on data redundancy needs (default is 3).

### 4. Network Configuration:

- Ensure proper network configuration and DNS settings.
- Optimize network settings for Hadoop traffic.

## INSTALLATION STEPS

```
C:\Windows\System32>java -version
java version "1.8.0_421"
Java(TM) SE Runtime Environment (build 1.8.0_421-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.421-b09, mixed mode)
```

Check for Hadoop version

```
C:\Windows\System32>hadoop
Usage: hadoop [--config confdir] [--loglevel loglevel] COMMAND
where COMMAND is one of:
    fs                run a generic filesystem user client
    version            print the version
    jar <jar>         run a jar file
                     note: please use "yarn jar" to launch
                           YARN applications, not this command.
    checknative [-a|-h] check native hadoop and compression libraries availability
    conftest          validate configuration XML files
    distch path:owner:group:permission distributed metadata changer
    distcp <srcurl> <desturl> copy file or directories recursively
    archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
    classpath          prints the class path needed to get the
                     Hadoop jar and the required libraries
    credential         interact with credential providers
    jnipath            prints the java.library.path
    kerbname           show auth_to_local principal conversion
    kdiag             diagnose kerberos problems
    key               manage keys via the KeyProvider
    trace             view and modify Hadoop tracing settings
    daemonlog         get/set the log level for each daemon
or
    CLASSNAME          run the class named CLASSNAME

Most commands print help when invoked w/o parameters.
```

add path variables for java and Hadoop

HADOOP_HOME	C:\hadoop-3.4.0\hadoop-3.4.0\bin
JAVA_HOME	C:\java\jdk-1.8

```
C:\hadoop-3.4.0\hadoop-3.4.0\bin
C:\hadoop-3.4.0\hadoop-3.4.0\sbin
```

Run start -dfs.cmd

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

C:\hadoop-3.4.0\hadoop-3.4.0>start-dfs.cmd

C:\hadoop-3.4.0\hadoop-3.4.0>|
```

```
C:\Windows\System32\cmd.e x + v
Apache Hadoop Distribution x + v
C:\hadoop-3.4.0\hadoop-3.4.0>start-dfs.cmd
C:\hadoop-3.4.0\hadoop-3.4.0>|
```

Run start-yarn.cmd

```
C:\Windows\System32\cmd.e  X + v

Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

C:\hadoop-3.4.0\hadoop-3.4.0>start-dfs.cmd

C:\hadoop-3.4.0\hadoop-3.4.0>start-yarn.cmd
starting yarn daemons

C:\hadoop-3.4.0\hadoop-3.4.0>
```

```
Apache Hadoop Distribution  X + v

at org.
at org.
at org.
Caused by: 5: A
2024-08-08 08:59:17,482 INFO handler.ContextHandler: Started o.e.j.w.WebAppContext@1fe8f5e8{node,/file:///C:/Users/lath
i/AppData/Local/Temp/jetty-0_0_0-8042-hadoop-yarn-common-3_4_0_jar_-_any-8580262993751253720/webapp/,AVAILABLE}{jar:fi
le:/C:/hadoop-3.4.0/hadoop-3.4.0/share/hadoop/yarn/hadoop-yarn-common-3.4.0.jar!/webapps/node}
2024-08-08 08:59:17,521 INFO server.AbstractConnector: Started ServerConnector@7e2c64{HTTP/1.1, (http/1.1)}{0.0.0.0:8042
}
at org.
at org.
2024-08-08 08:59:17,522 INFO server.Server: Started @6357ms
at org.
2024-08-08 08:59:17,527 INFO webapp.WebApps: Web app node started at 8042
at org.
2024-08-08 08:59:17,529 INFO nodemanager.NodeStatusUpdaterImpl: Node ID assigned is : LAPTOP-3P2JP0HQ:63036.
at org.
2024-08-08 08:59:17,532 INFO util.JvmPauseMonitor: Starting JVM pause monitor
at org.
2024-08-08 08:59:17,546 INFO client.DefaultNoHARMAFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8031
at org.
2024-08-08 08:59:17,626 INFO nodemanager.NodeStatusUpdaterImpl: Running Applications Size : 0.
at org.
2024-08-08 08:59:20,720 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 0 time(s); retr
y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
at org.
2024-08-08 08:59:23,756 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 1 time(s); retr
ttributeStore.j y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
at org.
2024-08-08 08:59:26,795 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 2 time(s); retr
eAttributesMana y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
at org.
2024-08-08 08:59:29,843 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 3 time(s); retr
sManagerImpl.ja y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
at org.
2024-08-08 08:59:32,891 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 4 time(s); retr
... 13 y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-08-08 08:59:35,954 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 5 time(s); retr
/***** SHUTDOWN_MSG: S 2024-08-08 08:59:39,006 INFO ipc.Client: Retrying connect to server: 0.0.0.0/0.0.0.0:8031. Already tried 6 time(s); retr
***** y policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=10, sleepTime=1000 MILLISECONDS)
```

Run in the localhost using localhost:9870



## Run using localhost:8088