BDI

Name: Kartik Jolapara   **Branch:** Computer Engineering

SAP ID: 60004200107   **Batch:** B1

EXPERIMENT NO. 2
Installation of Hadoop on a single node cluster

AIM: Install Hadoop on a Single Node Cluster

THEORY:

Apache Hadoop is an open source framework that is used to efficiently store and process large datasets ranging in size from gigabytes to petabytes of data. Instead of using one large computer to store and process the data, Hadoop allows clustering multiple computers to analyze massive datasets in parallel more quickly.

Hadoop consists of four main modules:

- Hadoop Distributed File System (HDFS) – A distributed file system that runs on standard or low-end hardware. HDFS provides better data throughput than traditional file systems, in addition to high fault tolerance and native support of large datasets.

- Yet Another Resource Negotiator (YARN) – Manages and monitors cluster nodes and resource usage. It schedules jobs and tasks.

- MapReduce – A framework that helps programs do the parallel computation on data. The map task takes input data and converts it into a dataset that can be computed in key value pairs. The output of the map task is consumed by reduce tasks to aggregate output and provide the desired result.

- Hadoop Common – Provides common Java libraries that can be used across all modules.

How Hadoop Works

Hadoop makes it easier to use all the storage and processing capacity in cluster servers, and to execute distributed processes against huge amounts of data. Hadoop provides the building blocks on which other services and applications can be built.

Applications that collect data in various formats can place data into the Hadoop cluster by using an API operation to connect to the NameNode. The NameNode tracks the file directory structure and placement of "chunks" for each file, replicated across DataNodes. To run a job to query the data, provide a MapReduce job made up of many map and reduce tasks that run against the data in HDFS spread across the DataNodes. Map tasks run on each node against the input files supplied, and reducers run to aggregate and organize the final output.

The Hadoop ecosystem has grown significantly over the years due to its extensibility. Today, the Hadoop ecosystem includes many tools and applications to help collect, store, process, analyze, and manage big data. Some of the most popular applications are:

- Spark – An open source, distributed processing system commonly used for big data workloads. Apache Spark uses in-memory caching and optimized execution for fast performance, and it supports general batch processing, streaming analytics, machine learning, graph databases, and ad hoc queries.

- Presto – An open source, distributed SQL query engine optimized for low-latency, ad-hoc

analysis of data. It supports the ANSI SQL standard, including complex queries, aggregations, joins, and window functions. Presto can process data from multiple data sources including the Hadoop Distributed File System (HDFS) and Amazon S3.

- Hive – Allows users to leverage Hadoop MapReduce using a SQL interface, enabling analytics at a massive scale, in addition to distributed and fault-tolerant data warehousing.

- HBase – An open source, non-relational, versioned database that runs on top of Amazon S3 (using EMRFS) or the Hadoop Distributed File System (HDFS). HBase is a massively scalable, distributed big data store built for random, strictly consistent, real-time access for tables with billions of rows and millions of columns.

- Zeppelin – An interactive notebook that enables interactive data exploration.

Install Hadoop 2.9.1 on Windows 10

First download the **Hadoop 2.9.1** from the below link.

https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.9.1/hadoop-2.9.1.tar.gz

Create a folder path as below and copy the downloaded msi into this folder. Path:-

'**C:/BigData/hadoop-2.9.1**'
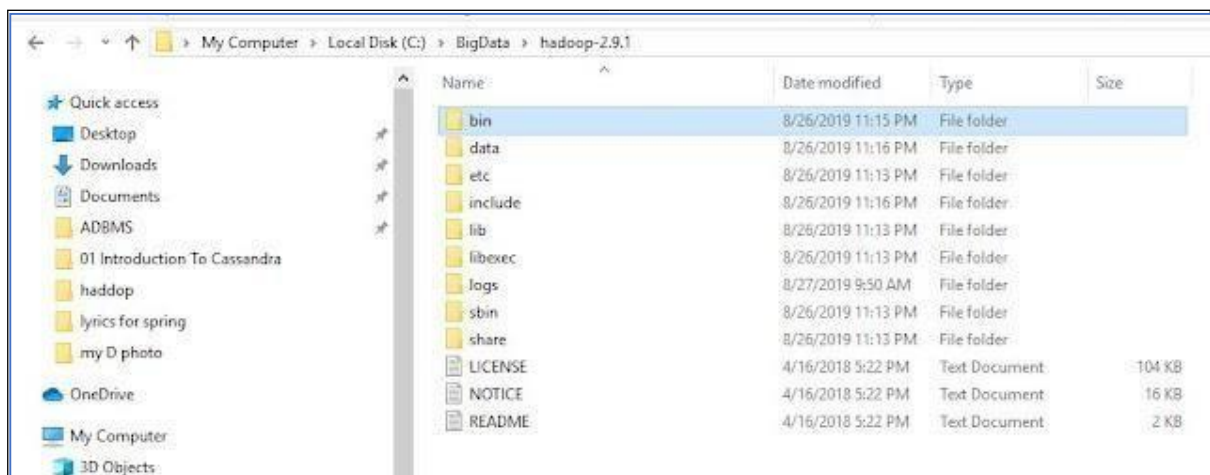
Then download the windows compatible binaries from the git hub repo.

Link:- https://github.com/ParixitOdedara/Hadoop
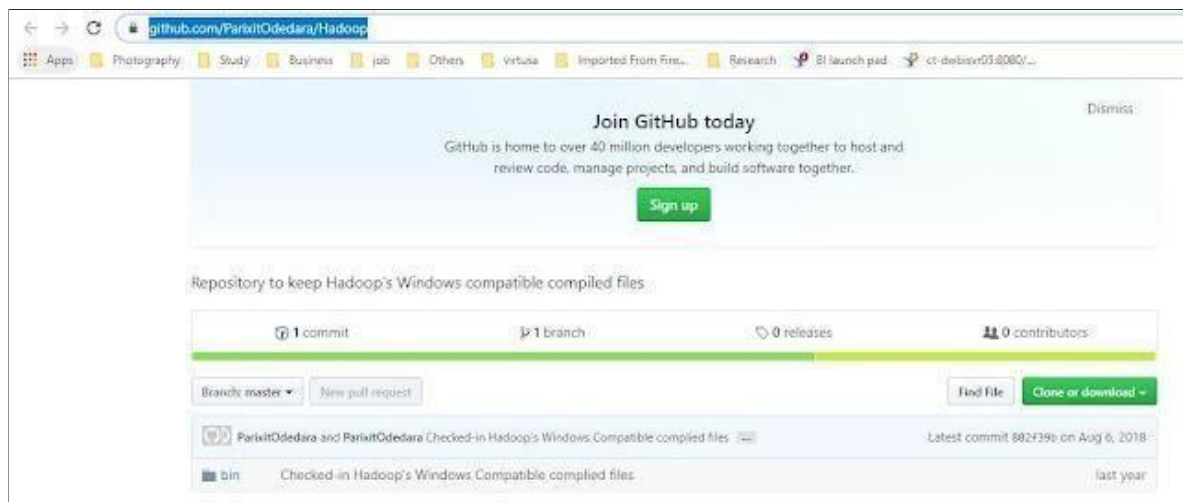
Extract the zip and copy all the files present under bin folder to C:\BigData\hadoop-2.9.1\bin. Replace the existing files as well.

Go to **C:/BigData/3adoop-2.9.1** and create a folder '**data**'. Inside the 'data' folder create two folders '**datanode**' and '**namenode**'.
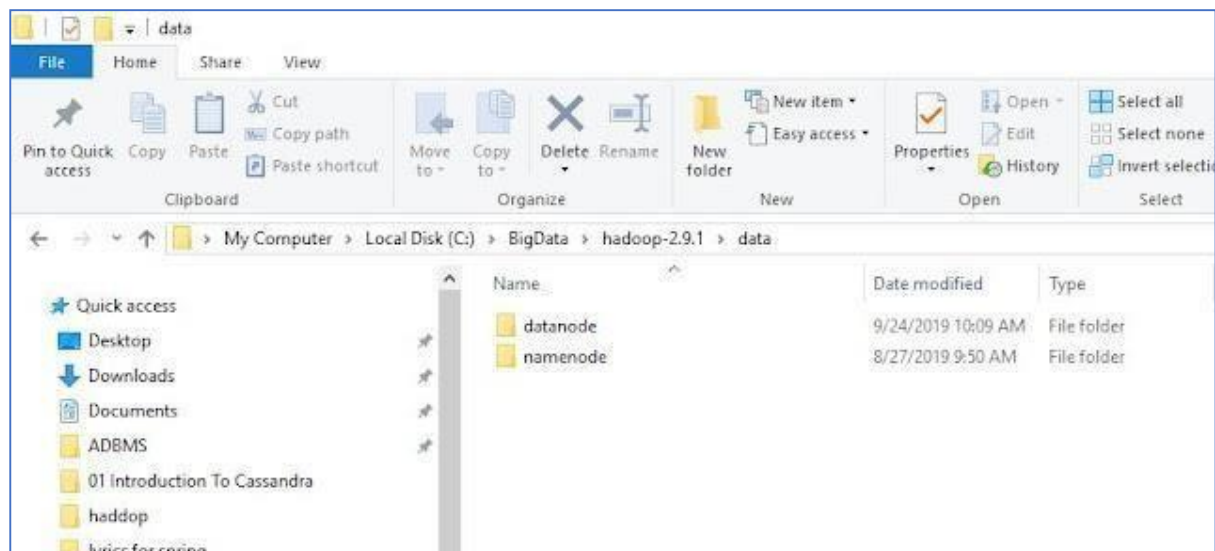
Then Set Hadoop Environment Variables

HADOOP_HOME="C:\BigData\hadoop-2.9.1"

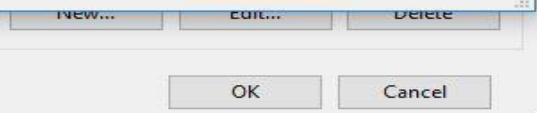*HADOOP_BIN="C:\BigData\hadoop-2.9.1\bin"*

JAVA_HOME=<JDK installation location>"

To set these variables, go to My Computer or This PC. Right click --> Properties --> Advanced

System settings --> Environment variables. Click New to create a new environment variables.
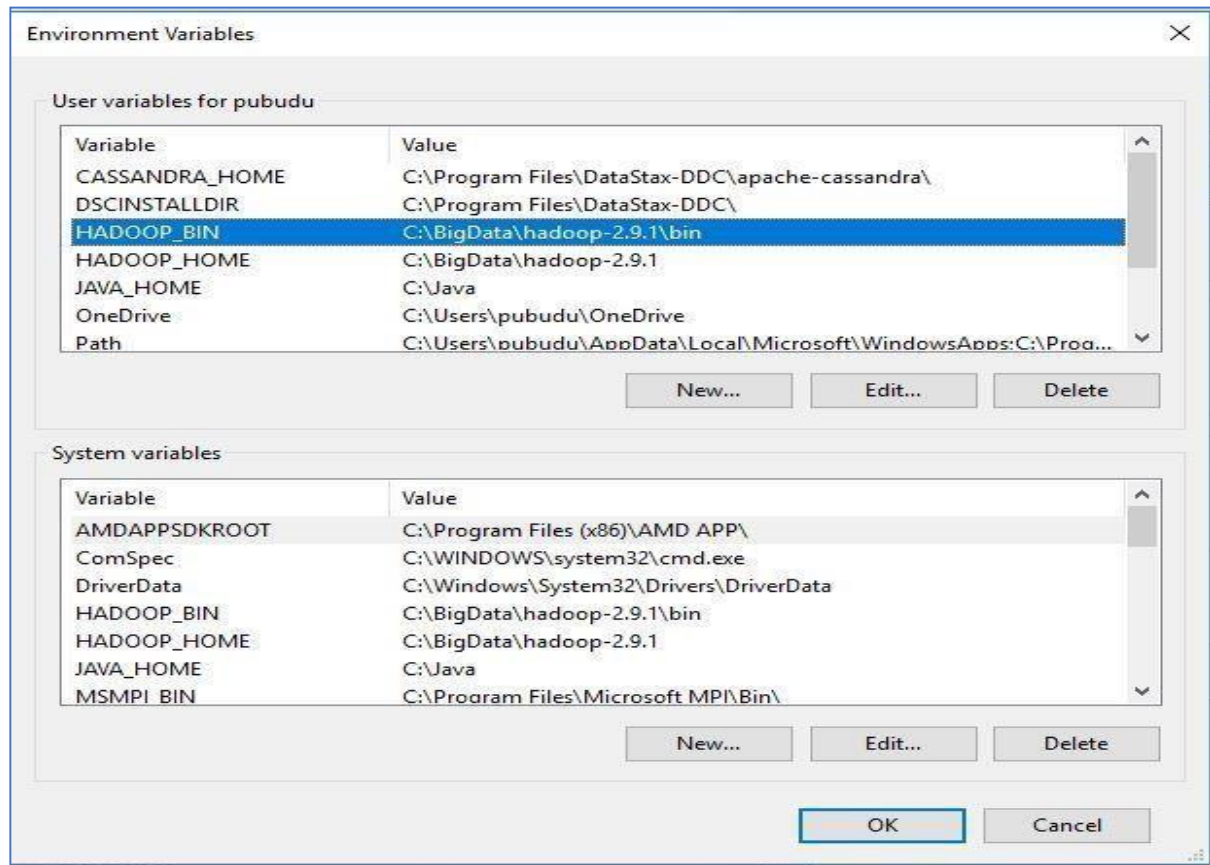
Then edit PATH Environment Variable

To validate the above setting, **open new cmd** and check the output.

echo %HADOOP_HOME%
echo %HADOOP_BIN%

echo %PATH%

To configure the Hadoop on windows we have to edit below mention files in the extracted location.

1. hadoop-env.cmd
2. core-site.xml
3. hdfs-site.xml
4. mapred-site.xml
5. yarn-site.xml

Edit hadoop-env.cmd

File location:-C:\BigData\hadoop-2.9.1\etc\hadoop\hadoop-env.cmd Need to add:-

```
set HADOOP_PREFIX=%HADOOP_HOME%
set HADOOP_CONF_DIR=%HADOOP_PREFIX%\etc\hadoop
set YARN_CONF_DIR=%HADOOP_CONF_DIR% set
PATH=%PATH%;%HADOOP_PREFIX%\bin
```

```
88  @rem        potential for a symlink attack.
89  set HADOOP_PID_DIR=%HADOOP_PID_DIR%
90  set HADOOP_SECURE_DN_PID_DIR=%HADOOP_PID_DIR%
91
92  @rem A string representing this instance of hadoop. %USERNAME% by default.
93  set HADOOP_IDENT_STRING=%USERNAME%
94  set HADOOP_PREFIX=%HADOOP_HOME%
95  set HADOOP_CONF_DIR=%HADOOP_PREFIX%\etc\hadoop
96  set YARN_CONF_DIR=%HADOOP_CONF_DIR%
97  set PATH=%PATH%;%HADOOP_PREFIX%\bin
```

Edit core-site.xml

File Location:- C:\BigData\hadoop-2.9.1\etc\hadoop\core-site.xml

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

Need to add:-content within <configuration> </configuration> tags.

```
14  |     limitations under the License. See accompanying LICENSE file.
15  -->
16
17  <!-- Put site-specific property overrides in this file. -->
18
19  <configuration>
20  <property>
21        <name>fs.default.name</name>
22        <value>hdfs://0.0.0.0:19000</value>
23     </property>
24  </configuration>
25
```

Edit hdfs-site.xml

File Location:- C:\BigData\hadoop-2.9.1\etc\hadoop\hdfs-site.xml. Need to
add;- below content within <configuration> </configuration> tags.

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>C:\BigData\hadoop-2.9.1\data\namenode</value>
  </property>
  <property>
```

```
      <name>dfs.datanode.data.dir</name>
      <value>C:\BigData\hadoop-2.9.1\data\datanode</value> </property>
</configuration>
```

Edit mapred-site.xml

File location:- Open C:\BigData\hadoop-2.9.1\etc\hadoop\mapred-site.xml

Need to add:- below content within <configuration> </configuration> tags. If you don't
see mapred-site.xml then open mapred-site.xml.template file and rename it to mapred-
site.xml

```
 <configuration>
   <property>
      <name>mapreduce.job.user.name</name>
      <value>%USERNAME%</value>
   </property>
   <property>
      <name>mapreduce.framework.name</name>
      <value>yarn</value>
   </property>
   <property>
      <name>yarn.apps.stagingDir</name>
      <value>/user/%USERNAME%/staging</value>
   </property>
   <property>
      <name>mapreduce.jobtracker.address</name>
      <value>local</value>
   </property>
</configuration>
```

Editing yarn-site.xml

Right click on the file, select edit and paste the following content withintags.

Note:- Below part already has the configuration tag, we need to copy only the part inside it.

```
<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>
<!-- Site specific YARN configuration properties --></configuration>
```
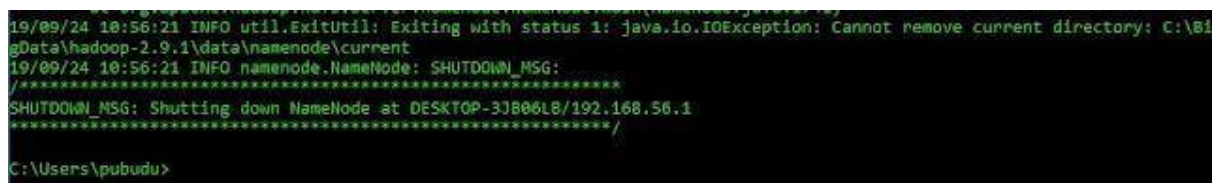
Additional Configuration:-

Check if C:\BigData\hadoop-2.9.1\etc\hadoop\slaves file is present, if that file not available create the file called slave and insert locahost as below.
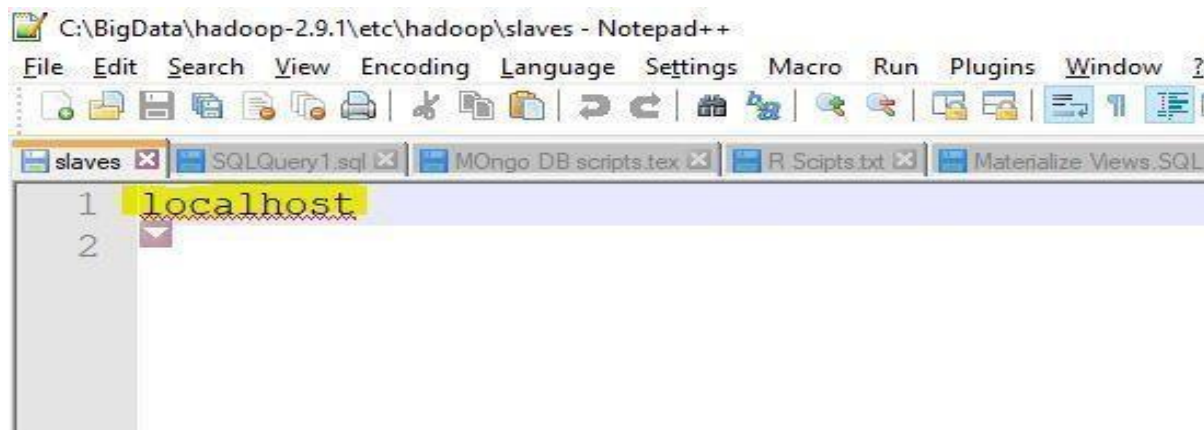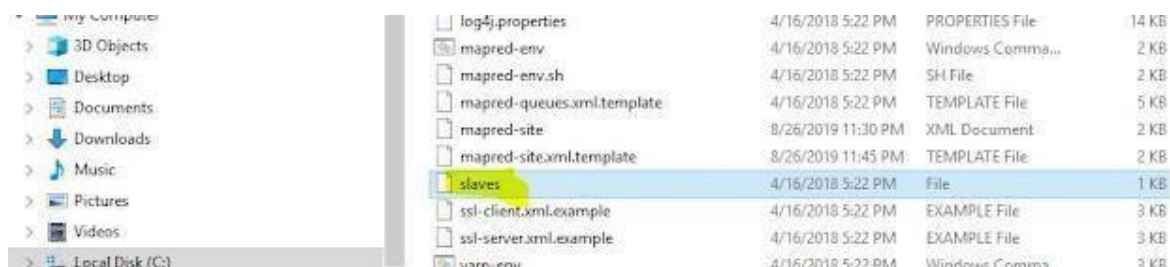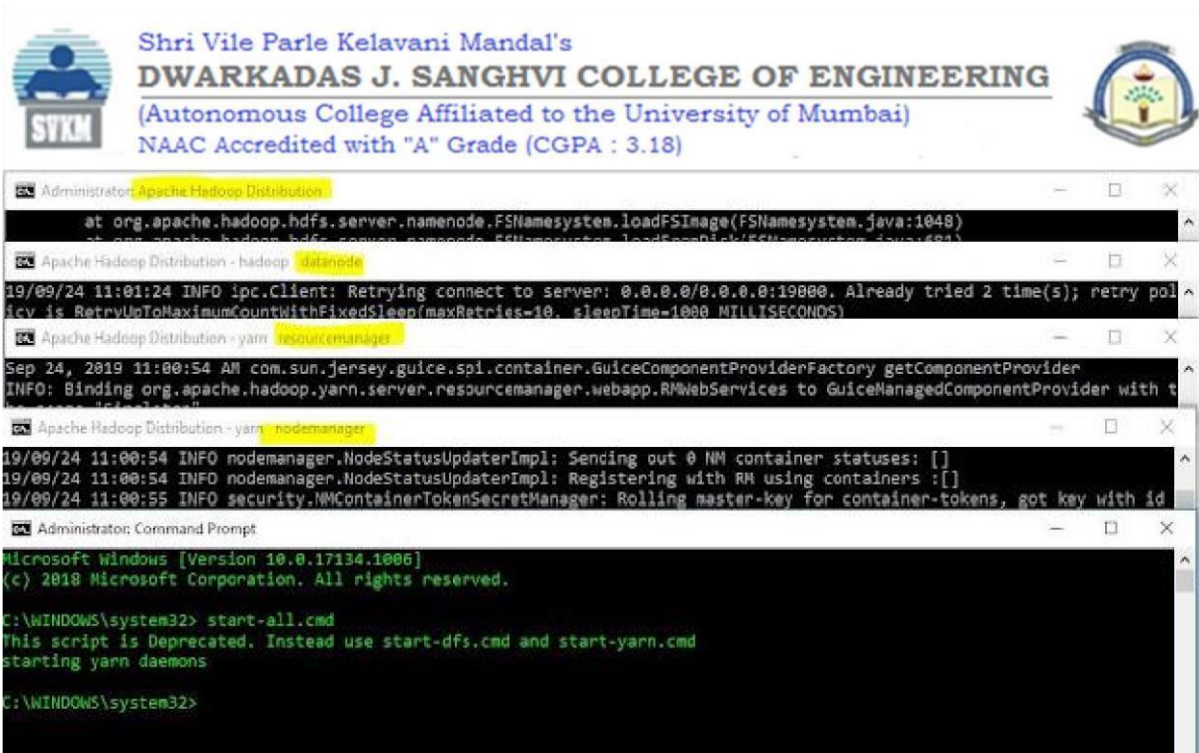

Node formatting

To format the node, **open the cmd** and execute the below command. *hadoop*

 namenode -format

To enable the Hadoop open the **CMD as Administrator** and type below command **start-all.cmd**
It will open 4 new windows cmd terminals for 4 daemon processes, namely namenode, datanode, nodemanager, and resourcemanager.
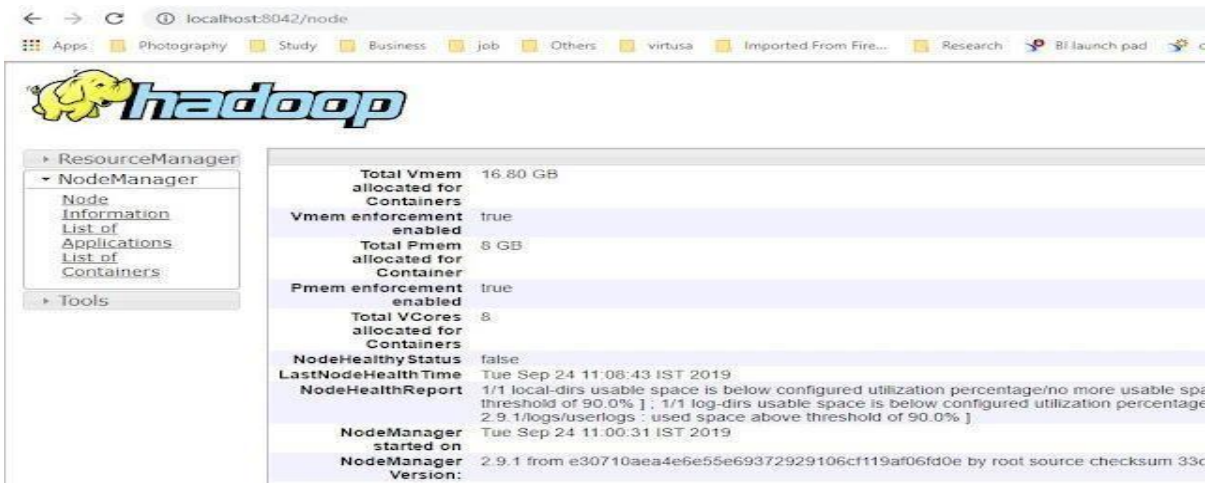
Then you have succesfully install the hadoop 2.9.1 on windows platform.

Now you can access all the Hadoop components via web urls.

To access Resource Manager go to http://localhost:8088 from your web browser.

To access Node Manager go to http://localhost:8042 from your web browser.

To access Name Node go to http://localhost:50070 from your web browser.

To access Data Node go to http://localhost:50075 from your web browser.

CONCLUSION: We have successfully installed Hadoop 2.9.1 on Windows 10.