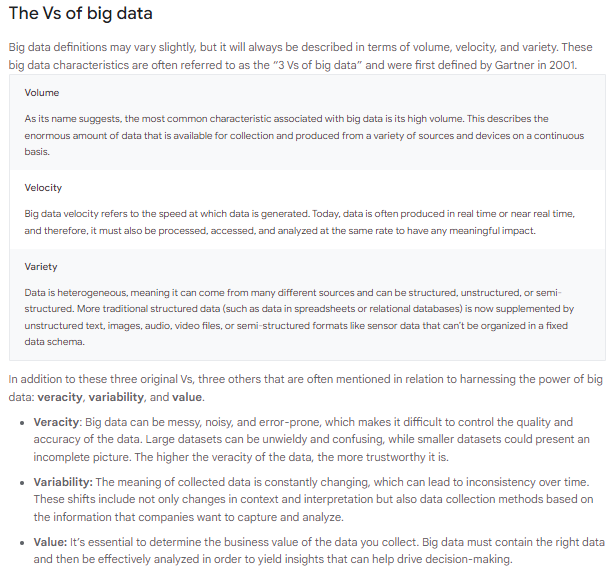
Big data

Big data refers to extremely large and diverse collections of structured, unstructured, and semi-structured data that continues to grow exponentially over time. These datasets are so huge and complex in volume, velocity, and variety, that traditional data management systems cannot store, process, and analyze them.

Link to article I have read:: <https://cloud.google.com/learn/what-is-big-data#:~:text=Big%20data%20describes%20large%20and,problems%20and%20make%20informed%20decisions>.



HADOOP::

What is Hadoop?

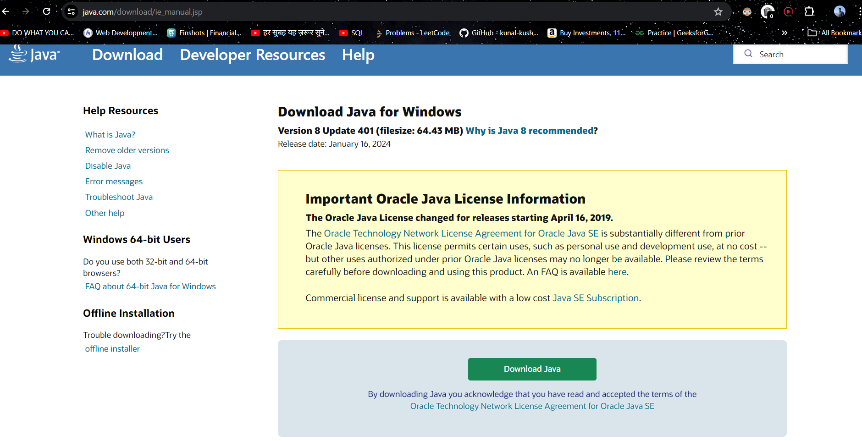
Apache Hadoop is an open source framework that is used to efficiently store and process large datasets ranging in size from gigabytes to petabytes of data. Instead of using one large computer to store and process the data, Hadoop allows clustering multiple computers to analyze massive datasets in parallel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

Basically Hadoop uses distributed computing model

**Installation and setup Hadoop:**

**Hadoop Is java based to for that we need to install JDK and JRE**

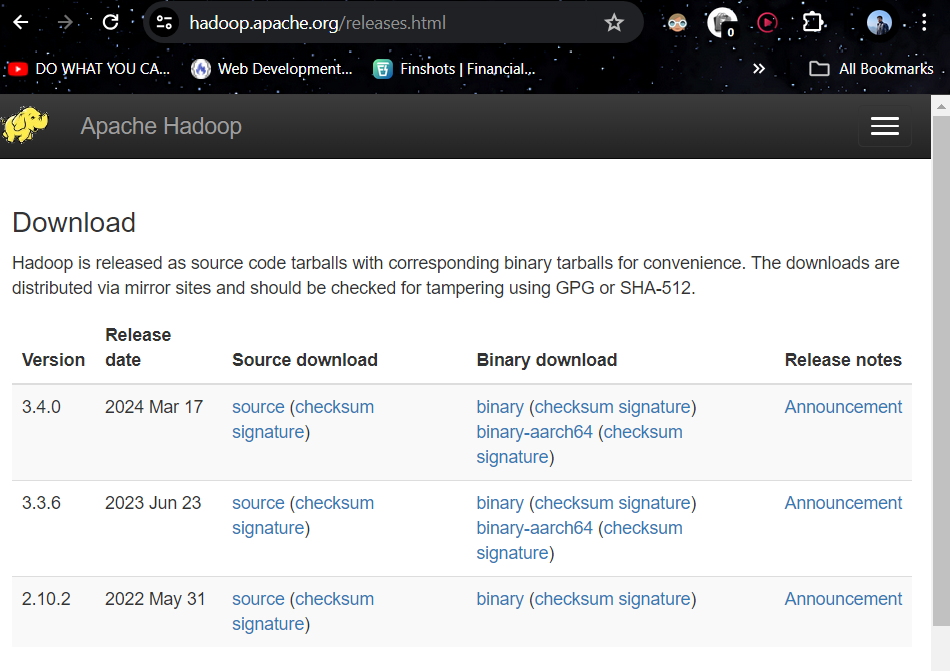
1. JRE 1.8 — Offline installer for JRE
2. [Java Development Kit — 1.8](https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html#license-lightbox)
3. A Software for Un-Zipping like [7Zip](https://www.7-zip.org/download.html) or [Win Rar](https://www.win-rar.com/download.html?L=0)  
   \* I will be using a 64-bit windows for the process, please check and download the version supported by your system x86 or x64 for all the software.



Now Hadoop file to download:

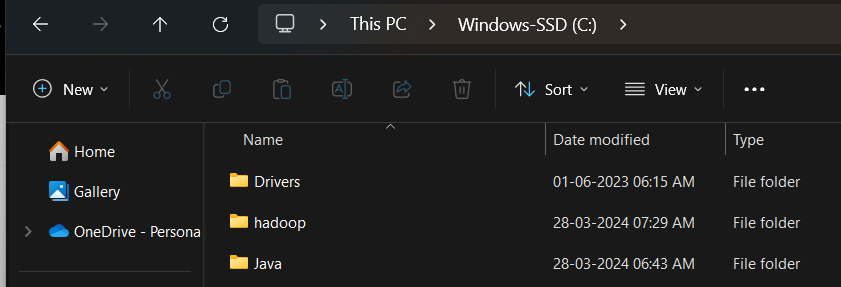
I am downloading 3.3.6 version of Hadoop from

<https://hadoop.apache.org/releases.html>

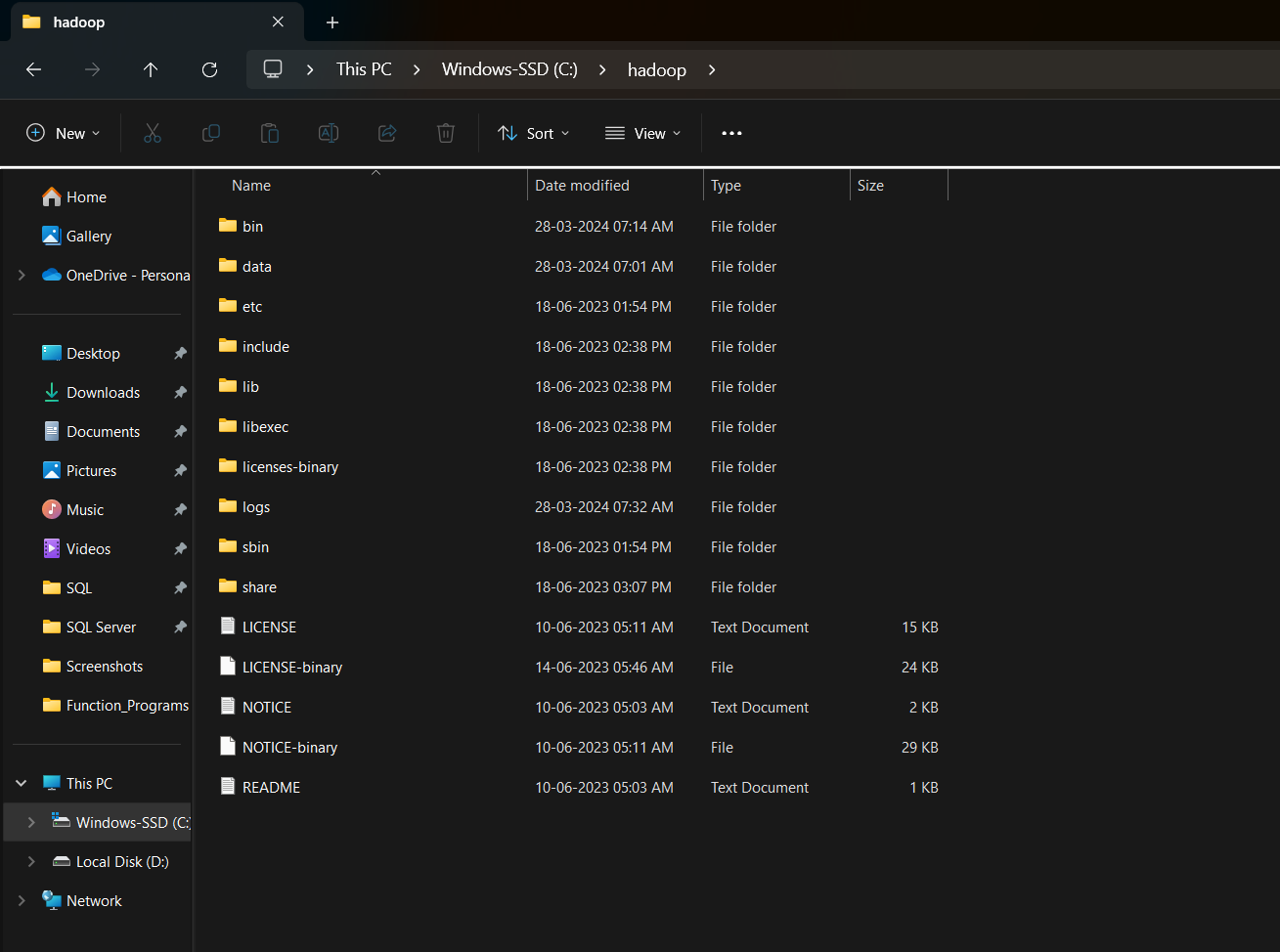


**2. Unzip and Install Hadoop**

After Downloading the Hadoop, we need to Unzip the hadoop-3.3.6.tar.gz file.

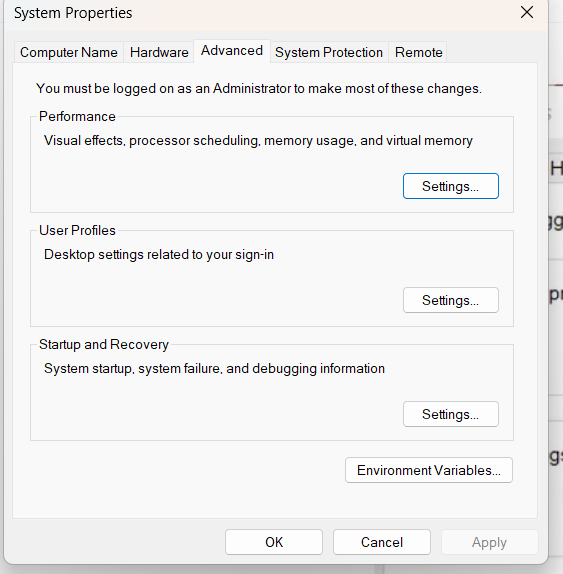


Once extracted, we would get a new file hadoop-2.9.2.tar.  
Now, once again we need to extract this tar file.



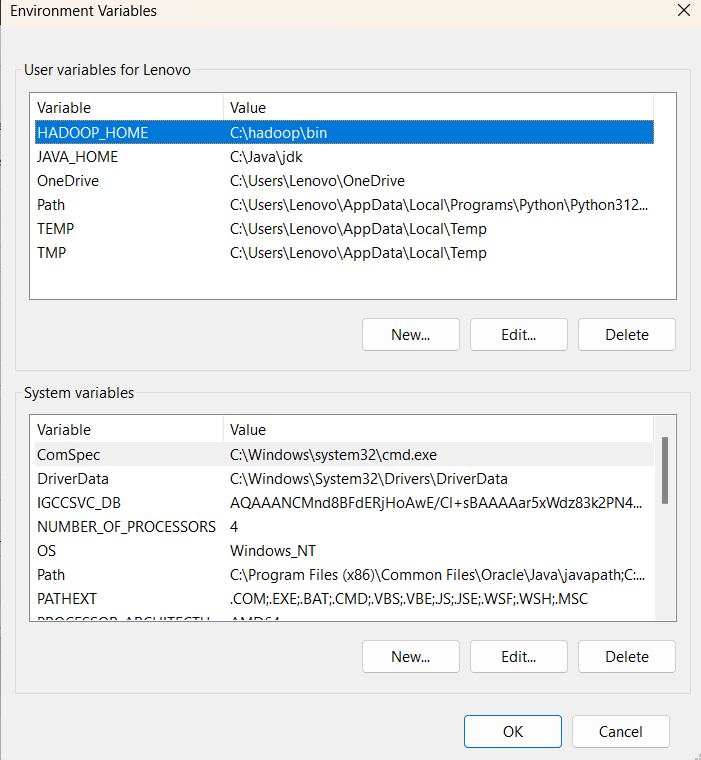
* Please note while creating folders, DO NOT ADD SPACES IN BETWEEN THE FOLDER NAME.(it can cause issues later)
* I have placed my Hadoop in C: drive

# 3. Setting Up Environment Variables

Another important step in setting up a work environment is to set your Systems environment variable.To edit environment variables, go to Control Panel > System > click on the “Advanced system settings” link

**Setting JAVA\_HOME & HADOOP\_HOME**

* Open environment Variable and click on “New” in “User Variable”



* Now as shown, add JAVA\_HOME in variable name and path of Java(jdk) in Variable Value.
* Click OK and we are half done with setting JAVA\_HOME.

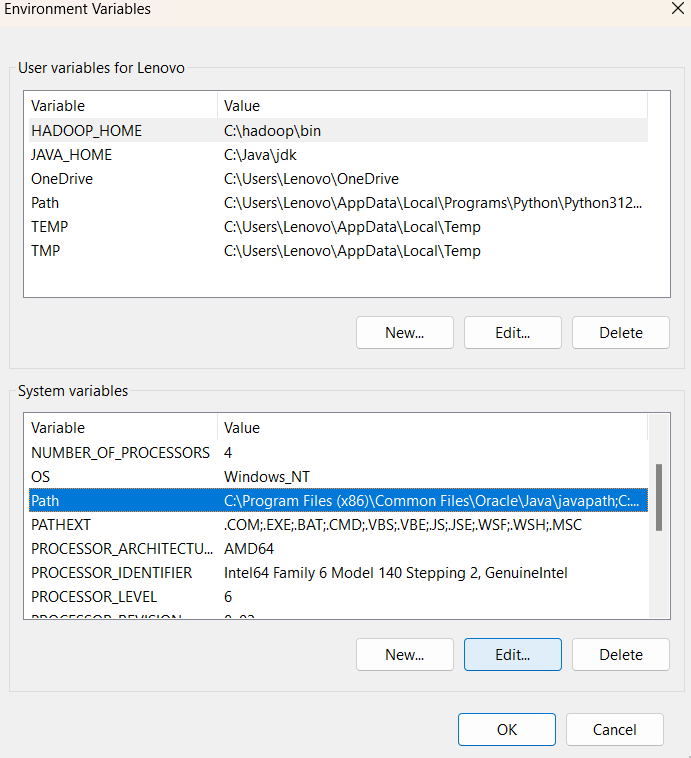
**FOR HADOOP**

* Now as shown, add HADOOP\_HOME in variable name and path of Hadoop folder in Variable Value.
* Click OK and we are half done with setting HADOOP\_HOME.

**Setting Path Variable**

* Last step in setting Environment variable is setting Path in System Variable.

Select Path variable in the system variables and click on Edit



* Now we need to add these paths to Path Variable one by one:-  
  \* %JAVA\_HOME%\bin  
  \* %HADOOP\_HOME%\bin  
  \* %HADOOP\_HOME%\sbin
* Click OK and OK. & we are done with Setting Environment Variables.

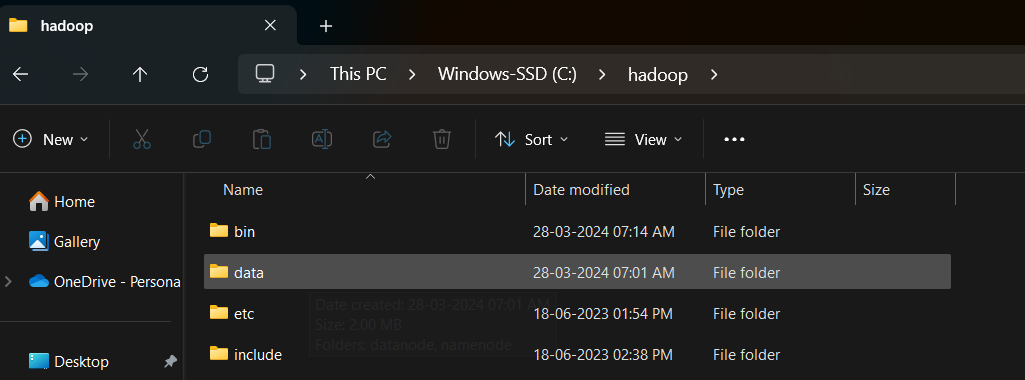
# Editing Hadoop files

Once we have configured the environment variables next step is to configure Hadoop. It has 3 parts:-

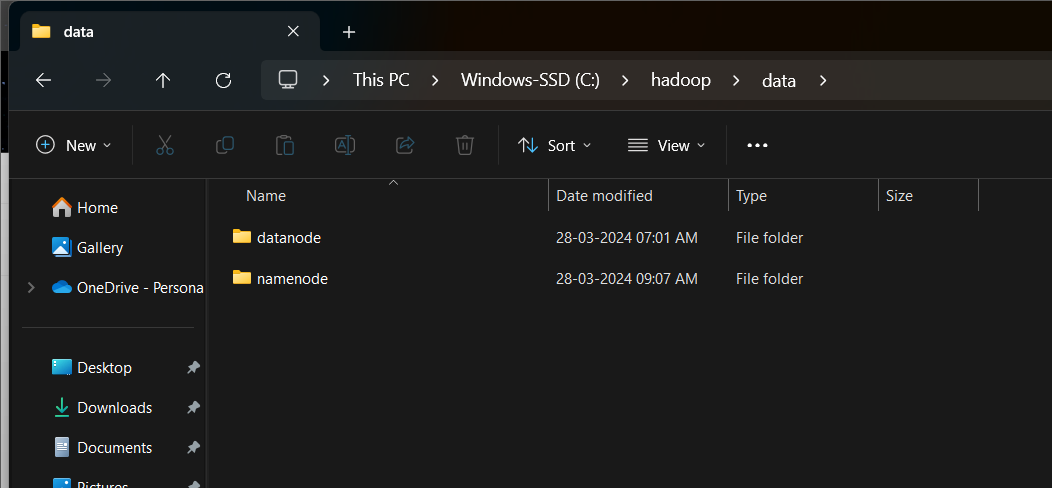
**4.1 Creating Folders**

We need to create a folder data in the hadoop directory, and 2 sub folders namenode and datanode

* Create **DATA folder**in the Hadoop directory



Once DATA folder is created, we need to create 2 new folders namely, namenode and datanode inside the data folder

These folders are important because files on HDFS resides inside the datanode.

**Editing Configuration Files:**

Now we need to edit the following config files in hadoop for configuring it :-

(We can find these files in Hadoop -> etc -> hadoop)

\* core-site.xml

\* hdfs-site.xml

\* mapred-site.xml

\* yarn-site.xml

\* hadoop-env.cmd

4.2.1 Editing core-site.xml

Right click on the file, select edit and paste the following content within <configuration> </configuration> tags.

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

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>

4.2.2 Editing hdfs-site.xml

Right click on the file, select edit and paste the following content within <configuration></configuration>tags.

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

Also replace PATH~1 and PATH~2 with the path of namenode and datanode folder that we created recently(step 4.1).

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>PATH~1\namenode</value>

<final>true</final>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>PATH~2\datanode</value>

<final>true</final>

</property>

</configuration>

4.2.3 Editing mapred-site.xml

Right click on the file, select edit and paste the following content within <configuration> </configuration> tags.

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

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

</configuration>

4.2.4 Editing yarn-site.xml

Right click on the file, select edit and paste the following content within <configuration> </configuration> tags.

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>

4.2.5 Verifying hadoop-env.cmd

Right click on the file, select edit and check if the JAVA\_HOME is set correctly or not.

We can replace the JAVA\_HOME variable in the file with your actual JAVA\_HOME that we configured in the System Variable.

set JAVA\_HOME=%JAVA\_HOME%

OR

set JAVA\_HOME="C:\Program Files\Java\jdk1.8.0\_221"

4.3 Replacing bin

Last step in configuring the hadoop is to download and replace the bin folder.

\* Go to this GitHub Repo and download the bin folder as a zip.

\* Extract the zip and copy all the files present under bin folder to %HADOOP\_HOME%\bin

**Testing Setup**

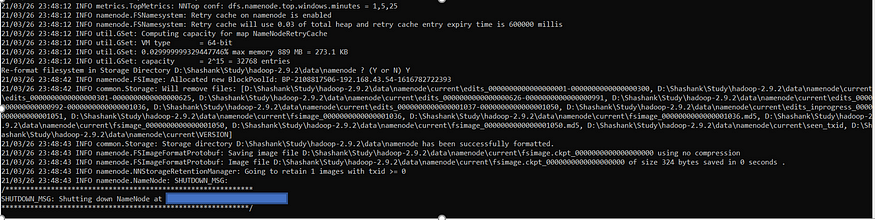
Congratulation..!!!!!  
We are done with the setting up the Hadoop in our System.

Now we need to check if everything works smoothly…

**5.1 Formatting Namenode**

Before starting hadoop we need to format the namenode for this we need to start a NEW Command Prompt and run below command

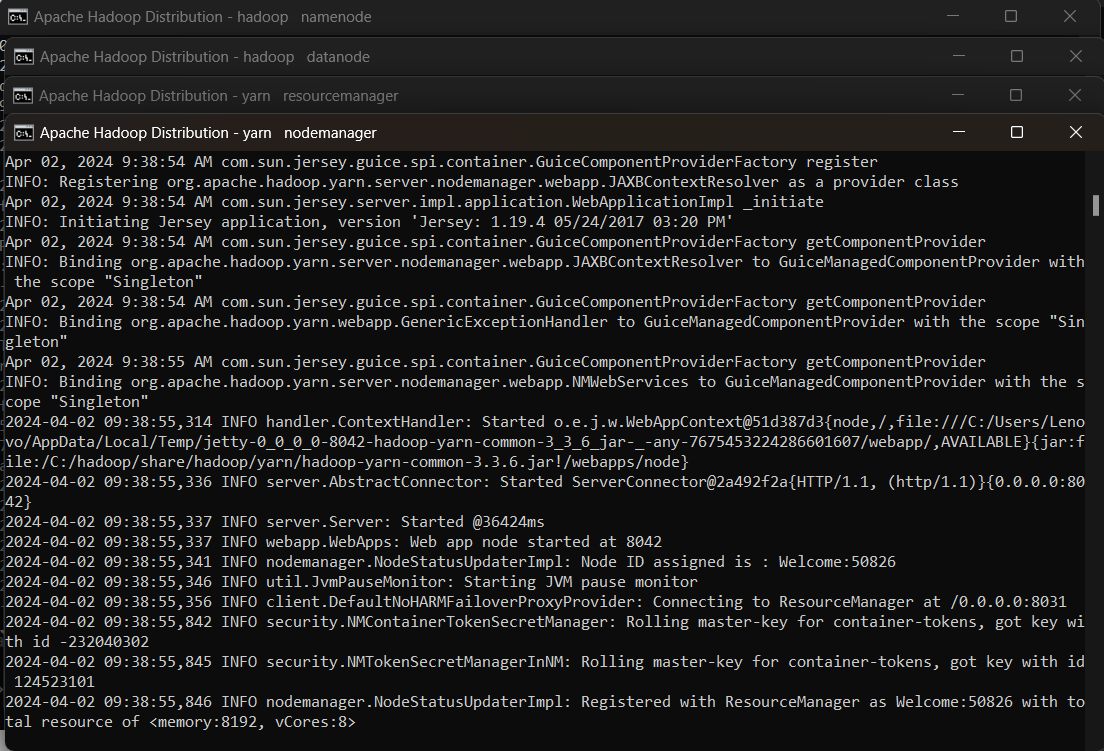
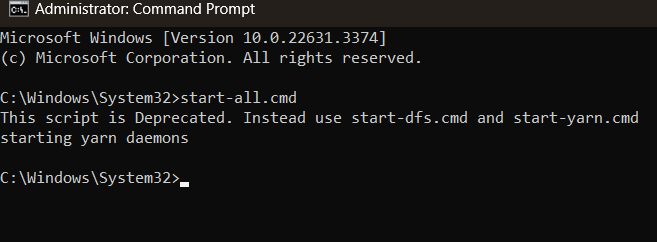
hadoop namenode -format



**Launching Hadoop**

Now we need to start a new Command Prompt remember to run it as administrator to avoid permission issues and execute below commands

**start-all.cmd**



**RUNNING HADOOP ON WEB:**

**Use localhost.9000**

**And localhost.8080**

**DONE!!!!!!!**

**Hadoop architecture::**

**Sor for that I refer to   
this medium article where I learned about how hadpoo got intorduce and how it updated with time**

**HADOOP 1X, 2X, 3X.**

[**https://medium.com/@sujathamudadla1213/explain-hadoop-1-2-3-architechture-in-depth-95fd71e13bc6**](https://medium.com/@sujathamudadla1213/explain-hadoop-1-2-3-architechture-in-depth-95fd71e13bc6)

**WORDCOUNT PROBLEM:**

* **First create a text file in local machine**
* **Then** Create a directory in HDFS, where to kept text file.  
  $ hdfs dfs -mkdir /test
* Upload the data.txt file on HDFS in the specific directory.  
  $ hdfs dfs -put /C/data.txt /test
* Then reat that file and open it into browser using Hadoop
* Then apply mapreduce on it and peform tasks
* Then see how map reduce work on that file and display it on to browser