

Ex. No: 5

Date:

Installation of Single Node Hadoop Cluster

AIM

To find procedure to set up the one node Hadoop cluster.

SCENARIO

You are a data engineering intern working for a startup that handles large datasets. Your team is exploring Apache Hadoop for big data processing, and you have been assigned the task of setting up a one-node Hadoop cluster on a Linux machine.

DESCRIPTION

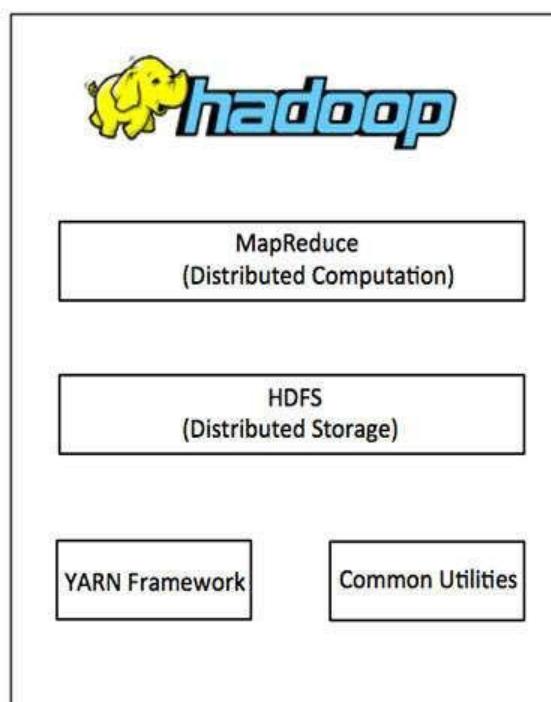
Hadoop is an Apache open source framework written in java that allows distributed

processing of large datasets across clusters of computers using simple programming models. The Hadoop framework application works in an environment that provides distributed *storage* and *computation* across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each

Hadoop Architecture

Hadoop has two major layers namely

- Processing/Computation layer (MapReduce),
- Storage layer (Hadoop Distributed File



MapReduce

MapReduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multi-terabyte datasets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner.

The MapReduce program runs on Hadoop which is an Apache open-source framework.

Hadoop Distributed File System

The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets.

Apart from the above-mentioned two core components, Hadoop framework also includes the following two modules –

- **Hadoop Common** – These are Java libraries and utilities required by other Hadoop modules.
 - **Hadoop YARN** – This is a framework for job scheduling and cluster resource management.
-

Advantages of Hadoop

- Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatically distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
- Hadoop does not rely on hardware to provide fault-tolerance and high availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer.
- Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.
- Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based.

PROCEDURE STEPS

Step 1: Download the following Packages

1. hadoop-3.2.2.tar.gz

Link: <https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-3.2.2/hadoop-3.2.2.tar.gz>

2. 7zip to unzip the tar.gz file

Link: <https://www.7-zip.org/download.html>

3. Java 8 (JDK-8U333 & JRE-8U333)

Link: <https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html>

4. Hadoop dll Files & hadoop-hdfs-3.2.2.jar

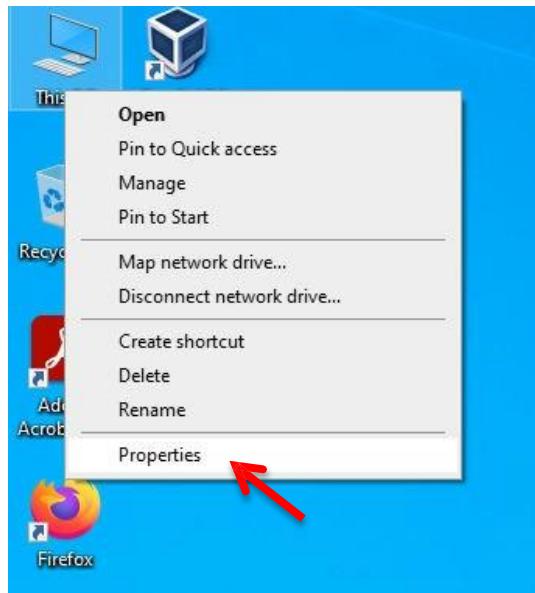
Link: <https://github.com/paranikumar/Hadoop-Dll-JAR-Files.git>

Step 2: Install / Extract the following Packages:

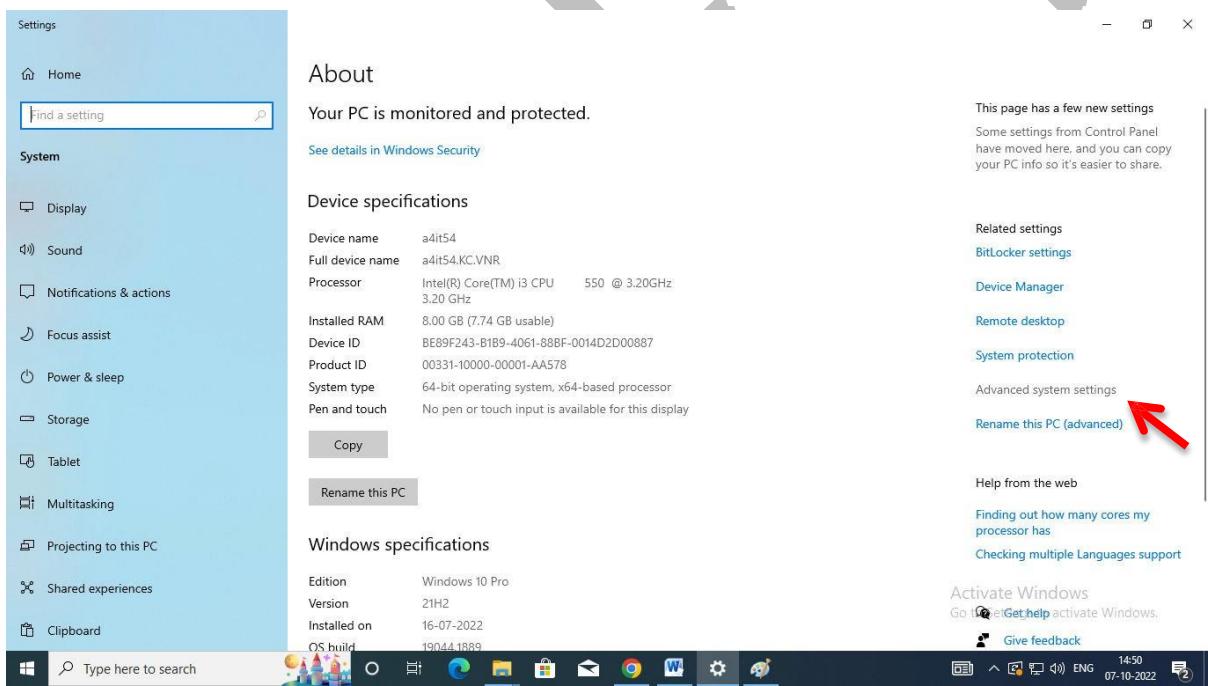
1. Uninstall the existing java and Install Java 8 in the location **C:\JAVA**. Before installing create two folders namely “**jdk1.8.0_333**” and “**jre1.8.0_333**” inside the “**C:\JAVA**” Folder.
2. Create a folder “**hadoop-env**” in **D:**
3. Paste the downloaded “hadoop-3.2.2.tar.gz” into “**d:\ hadoop-env**”
4. Unzip the “hadoop-3.2.2.tar.gz” using 7zip (do the unzip function two times)
5. Copy the downloaded Hadoop dll files to the location “**D:\hadoop-env\hadoop-3.2.2\bin**”
6. Copy the downloaded hadoop-hdfs-3.2.2.jar to “**D:\hadoop-env\hadoop-3.2.2\share\hadoop\hdfs**”
7. Edit the **JAVA_HOME = C:\JAVA\jdk1.8.0_333** in the path “**D:\hadoop-env\hadoop-3.2.2\etc\hadoop\hadoop-env.cmd**

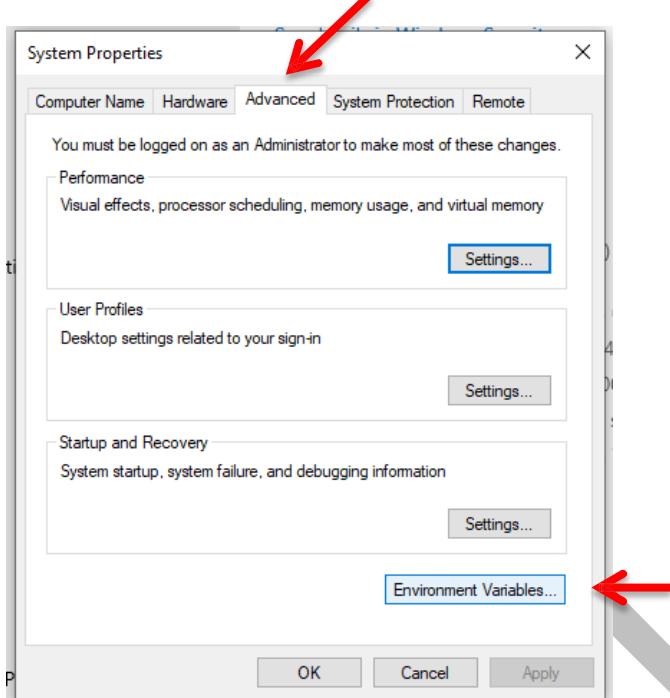
Step 3: Set the path for Java and Hadoop in system environment variables

To edit environment variables, go to Control Panel → System and Security → System (or) right-click “This PC” → Properties (My Computer icon) and click on the “Advanced system settings” link.

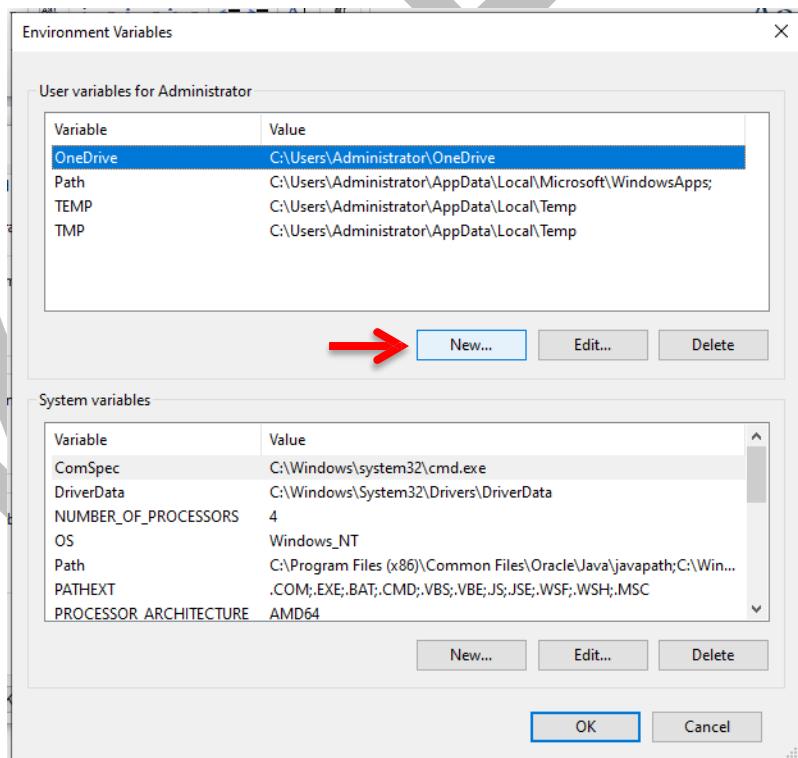


When the “Advanced system settings” dialog appears, go to the “Advanced” tab and click on the “Environment variables” button located on the bottom of the dialog.



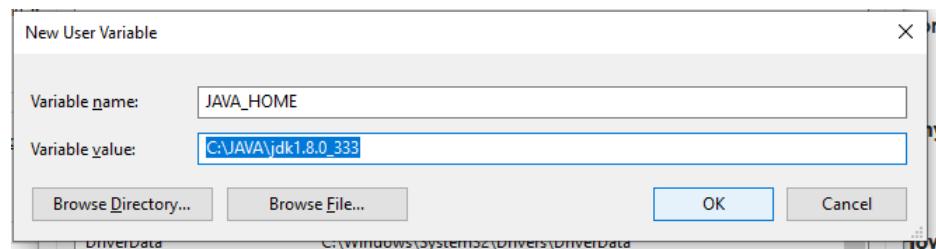


In the “Environment Variables” dialog, press the “New” button to add a new variable.

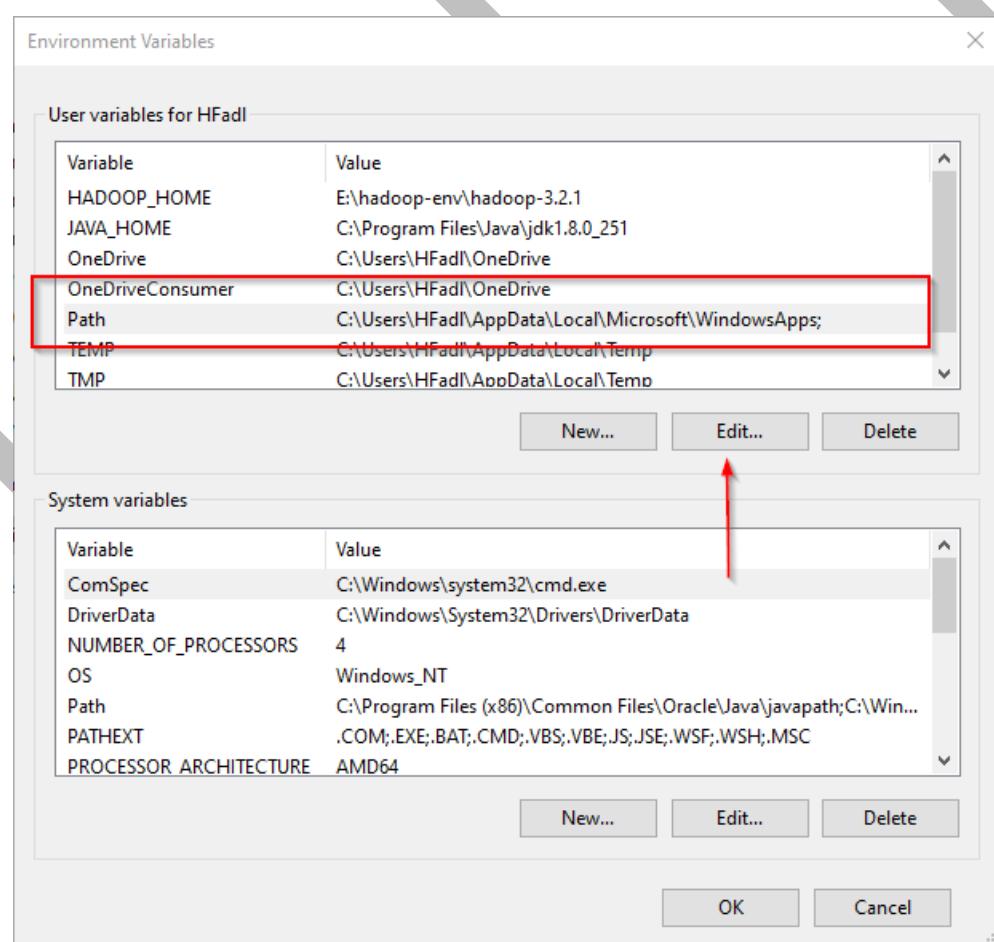
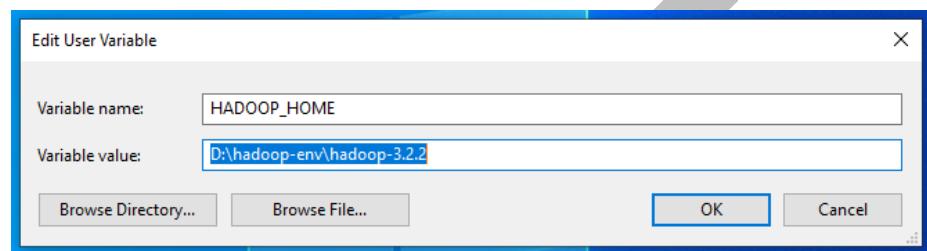


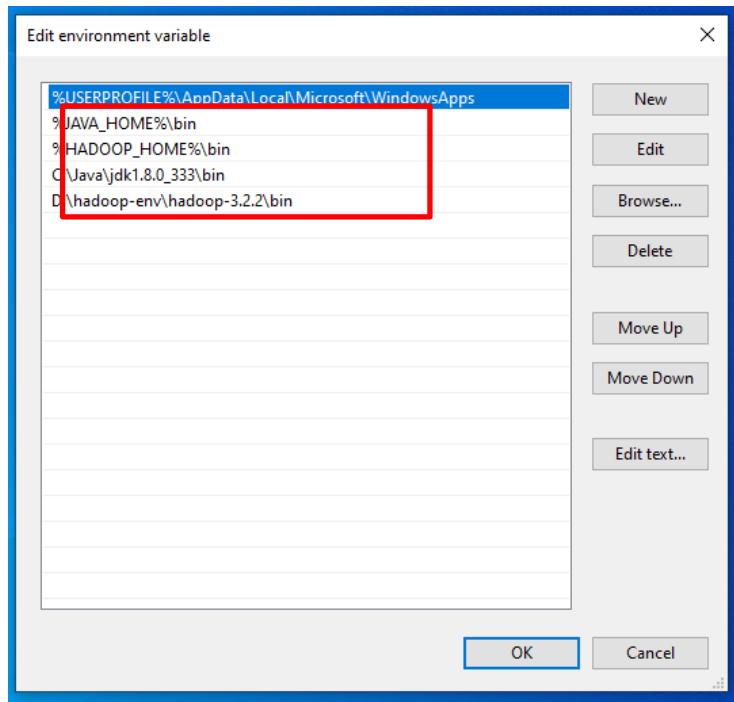
Now, Edit the PATH variable to add the Java and Hadoop binaries paths as shown in the following screenshots.

JAVA_HOME: JDK installation folder path



HADOOP_HOME: Hadoop installation folder path





Step 4: Open Command Prompt as Administrator and run the following command:

hadoop –version

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

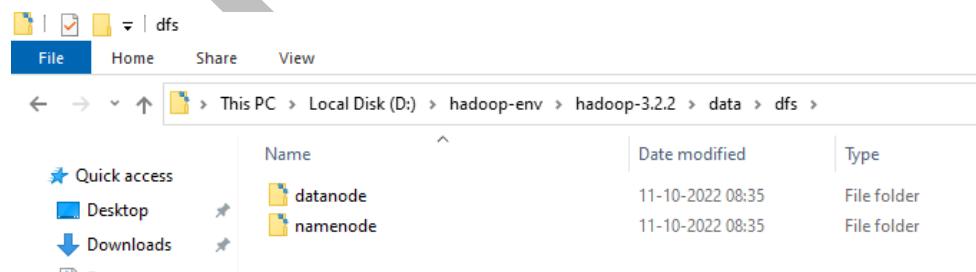
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator> hadoop -version
java version "1.8.0_333"
Java(TM) SE Runtime Environment (build 1.8.0_333-b02)
Java HotSpot(TM) 64-Bit Server VM (build 25.333-b02, mixed mode)
PS C:\Users\Administrator>
```

Step 5: Create the following folders

D:\hadoop-env\hadoop-3.2.2\data\dfs\namenode

D:\hadoop-env\hadoop-3.2.2\data\dfs\datanode



Step 6: Configuring Hadoop cluster

There are **four files** to configure Hadoop cluster:

Location of the File: “**D:\hadoop-env\hadoop-3.2.2\etc\hadoop**”

File Name: “**hdfs-site.xml**”

Add the following properties within the `<configuration></configuration>` element:

```
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:///D:/hadoop-env/hadoop-3.2.2/data/dfs/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:///D:/hadoop-env/hadoop-3.2.2/data/dfs/datanode</value>
</property>
```

File Name: “**core-site.xml**”

Add the following properties within the `<configuration></configuration>` element:

```
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9820</value>
</property>
```

File Name: “**mapred-site.xml**”

Add the following properties within the `<configuration></configuration>` element:

```
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
<description>MapReduce framework name</description>
</property>
```

File Name: “yarn-site.xml”

Add the following properties within the <configuration></configuration> element:

```
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
<description>Yarn Node Manager Aux Service</description>
</property>
```

Step 7: Format the namenode

After finishing the configuration, format the name using the following command

hdfs namenode –format

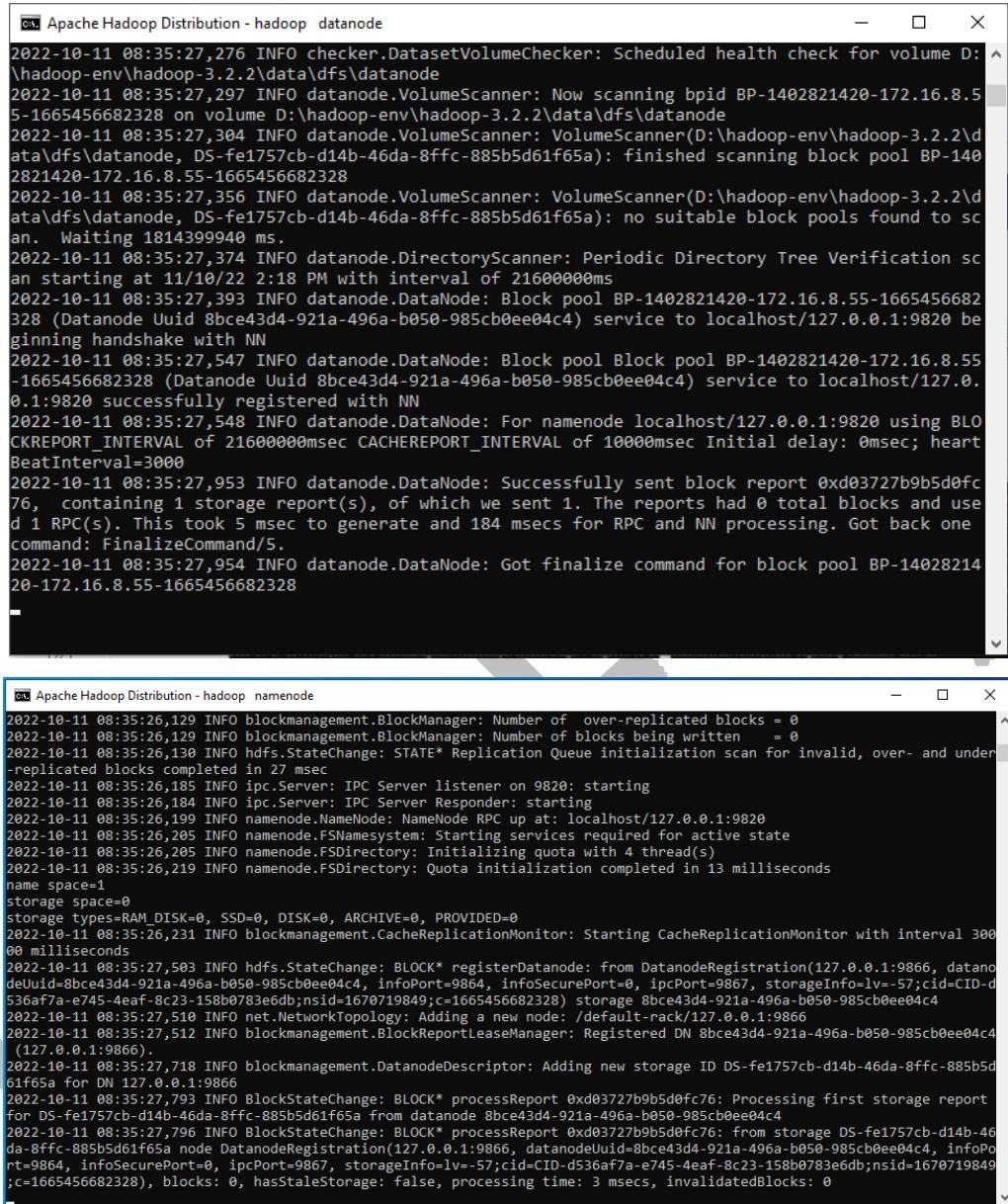
Step 8: Start the Hadoop nodes

Now navigate to the location “D:\hadoop-env\hadoop-3.2.2\sbin” in powershell and then run the following command to start the Hadoop nodes:

.\start-dfs.cmd

```
Administrator: Windows PowerShell
PS D:\hadoop-env\hadoop-3.2.2\sbin> .\start-dfs.cmd
PS D:\hadoop-env\hadoop-3.2.2\sbin>
```

Two command prompt windows will open (one for the **namenode** and one for the **datanode**) as follows:



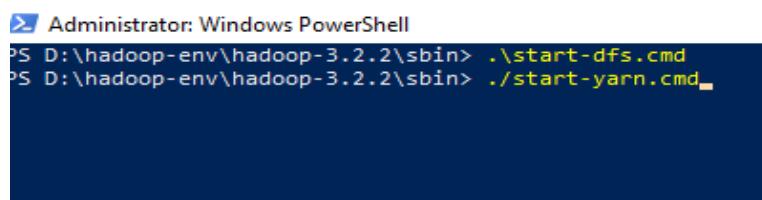
The image shows two command prompt windows side-by-side. The top window is titled "Apache Hadoop Distribution - hadoop datanode" and displays log entries from a DataNode. The bottom window is titled "Apache Hadoop Distribution - hadoop namenode" and displays log entries from a NameNode. Both windows show standard Hadoop startup and operational logs.

```
Apache Hadoop Distribution - hadoop datanode
2022-10-11 08:35:27,276 INFO checker.DatasetVolumeChecker: Scheduled health check for volume D:\hadoop-env\hadoop-3.2.2\data\dfs\datanode
2022-10-11 08:35:27,297 INFO datanode.VolumeScanner: Now scanning bpid BP-1402821420-172.16.8.55-1665456682328 on volume D:\hadoop-env\hadoop-3.2.2\data\dfs\datanode
2022-10-11 08:35:27,304 INFO datanode.VolumeScanner: VolumeScanner(D:\hadoop-env\hadoop-3.2.2\data\dfs\datanode, DS-fe1757cb-d14b-46da-8ffc-885b5d61f65a): finished scanning block pool BP-1402821420-172.16.8.55-1665456682328
2022-10-11 08:35:27,356 INFO datanode.VolumeScanner: VolumeScanner(D:\hadoop-env\hadoop-3.2.2\data\dfs\datanode, DS-fe1757cb-d14b-46da-8ffc-885b5d61f65a): no suitable block pools found to scan. Waiting 1814399940 ms.
2022-10-11 08:35:27,374 INFO datanode.DirectoryScanner: Periodic Directory Tree Verification scan starting at 11/10/22 2:18 PM with interval of 21600000ms
2022-10-11 08:35:27,393 INFO datanode.DataNode: Block pool BP-1402821420-172.16.8.55-1665456682328 (Datanode Uuid 8bce43d4-921a-496a-b050-985cb0ee04c4) service to localhost/127.0.0.1:9820 beginning handshake with NN
2022-10-11 08:35:27,547 INFO datanode.DataNode: Block pool Block pool BP-1402821420-172.16.8.55-1665456682328 (Datanode Uuid 8bce43d4-921a-496a-b050-985cb0ee04c4) service to localhost/127.0.0.1:9820 successfully registered with NN
2022-10-11 08:35:27,548 INFO datanode.DataNode: For namenode localhost/127.0.0.1:9820 using BLOCKREPORT_INTERVAL of 21600000msec CACHEREPORT_INTERVAL of 10000msec Initial delay: 0msec; heartBeatInterval=3000
2022-10-11 08:35:27,953 INFO datanode.DataNode: Successfully sent block report 0xd03727b9b5d0fc76, containing 1 storage report(s), of which we sent 1. The reports had 0 total blocks and used 1 RPC(s). This took 5 msec to generate and 184 msecs for RPC and NN processing. Got back one command: FinalizeCommand/5.
2022-10-11 08:35:27,954 INFO datanode.DataNode: Got finalize command for block pool BP-1402821420-172.16.8.55-1665456682328

Apache Hadoop Distribution - hadoop namenode
2022-10-11 08:35:26,129 INFO blockmanagement.BlockManager: Number of over-replicated blocks = 0
2022-10-11 08:35:26,129 INFO blockmanagement.BlockManager: Number of blocks being written = 0
2022-10-11 08:35:26,130 INFO hdfs.StateChange: STATE* Replication Queue initialization scan for invalid, over- and under-replicated blocks completed in 27 msec
2022-10-11 08:35:26,185 INFO ipc.Server: IPC Server listener on 9820: starting
2022-10-11 08:35:26,184 INFO ipc.Server: IPC Server Responder: starting
2022-10-11 08:35:26,199 INFO namenode.NameNode: NameNode RPC up at: localhost/127.0.0.1:9820
2022-10-11 08:35:26,205 INFO namenode.FSNamesystem: Starting services required for active state
2022-10-11 08:35:26,205 INFO namenode.FSDirectory: Initializing quota with 4 threads(s)
2022-10-11 08:35:26,219 INFO namenode.FSDirectory: Quota initialization completed in 13 milliseconds
name space=1
storage space=0
storage types=RAM_DISK=0, SSD=0, DISK=0, ARCHIVE=0, PROVIDED=0
2022-10-11 08:35:26,231 INFO blockmanagement.CacheReplicationMonitor: Starting CacheReplicationMonitor with interval 30000 milliseconds
2022-10-11 08:35:27,503 INFO hdfs.StateChange: BLOCK* registerDataNode: from DatanodeRegistration(127.0.0.1:9866, datanodeUuid=8bce43d4-921a-496a-b050-985cb0ee04c4, infoPort=9864, ipcPort=9867, storageInfo=lv=-57;cid=d536af7a-e745-4eaf-8c23-158b0783e6db;nsid=1670719849;c=1665456682328) storage 8bce43d4-921a-496a-b050-985cb0ee04c4
2022-10-11 08:35:27,510 INFO net.NetworkTopology: Adding a new node: /default-rack/127.0.0.1:9866
2022-10-11 08:35:27,512 INFO blockmanagement.BlockReportLeaseManager: Registered DN 8bce43d4-921a-496a-b050-985cb0ee04c4 (127.0.0.1:9866).
2022-10-11 08:35:27,718 INFO blockmanagement.DatanodeDescriptor: Adding new storage ID DS-fe1757cb-d14b-46da-8ffc-885b5d61f65a for DN 127.0.0.1:9866
2022-10-11 08:35:27,793 INFO BlockStateChange: BLOCK* processReport 0xd03727b9b5d0fc76: Processing first storage report for DS-fe1757cb-d14b-46da-8ffc-885b5d61f65a from datanode 8bce43d4-921a-496a-b050-985cb0ee04c4
2022-10-11 08:35:27,796 INFO BlockStateChange: BLOCK* processReport 0xd03727b9b5d0fc76: from storage DS-fe1757cb-d14b-46da-8ffc-885b5d61f65a node DatanodeRegistration(127.0.0.1:9866, datanodeUuid=8bce43d4-921a-496a-b050-985cb0ee04c4, infoPort=9864, ipcPort=9867, storageInfo=lv=-57;cid=d536af7a-e745-4eaf-8c23-158b0783e6db;nsid=1670719849;c=1665456682328), blocks: 0, hasStaleStorage: false, processing time: 3 msecs, invalidatedBlocks: 0
```

Next, Start the Hadoop Yarn service using the following command:

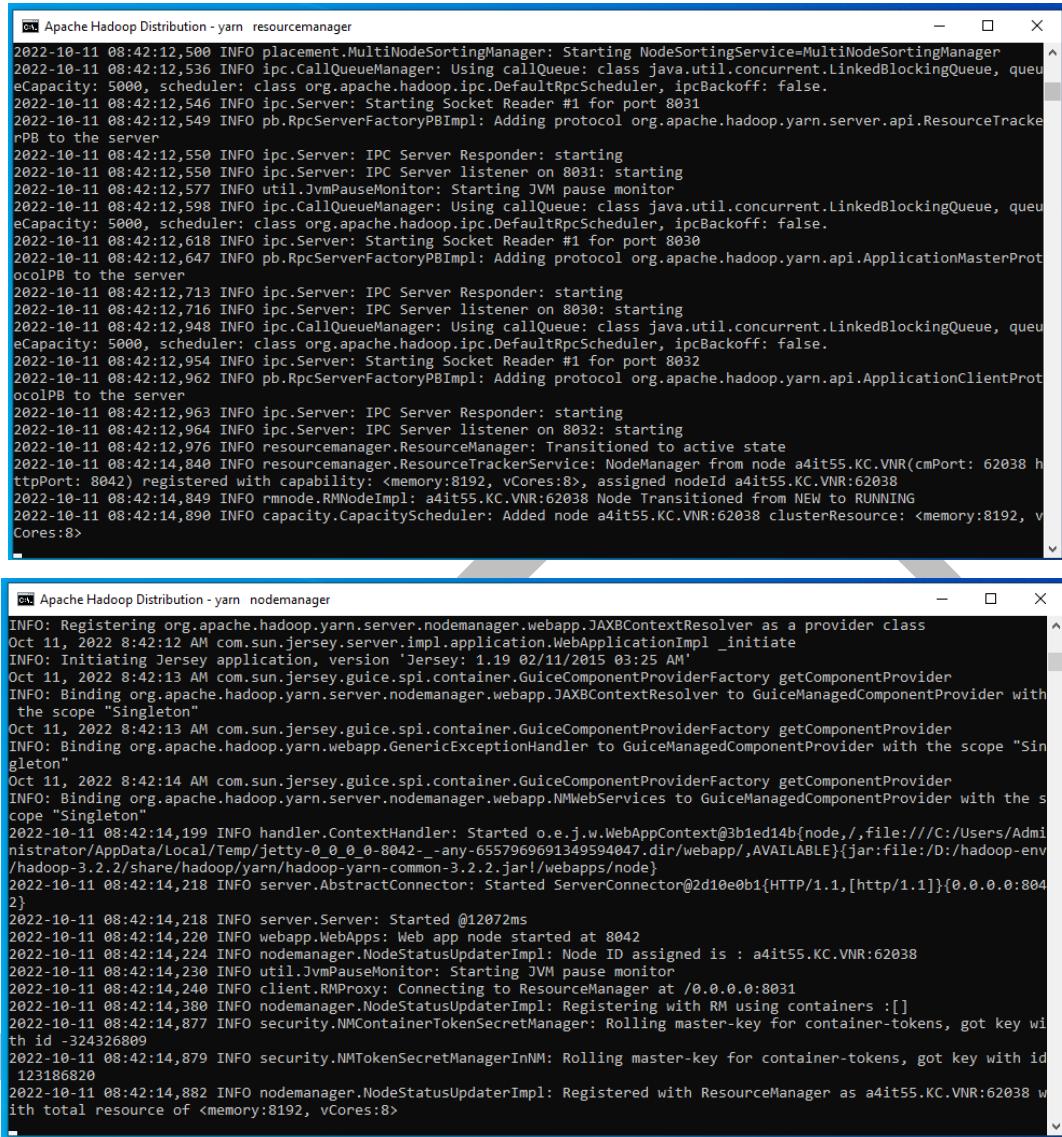
/start-yarn.cmd



The image shows an Administrator PowerShell window. It contains two commands: ".\start-dfs.cmd" and ".\start-yarn.cmd". Both commands are shown in yellow text, indicating they were successfully run.

```
Administrator: Windows PowerShell
PS D:\hadoop-env\hadoop-3.2.2\sbin> .\start-dfs.cmd
PS D:\hadoop-env\hadoop-3.2.2\sbin> ./start-yarn.cmd
```

Two command prompt windows will open (one for the **resource manager** and one for the **node manager**) as follows:



```
Apache Hadoop Distribution - yarn resourcemanager
2022-10-11 08:42:12,500 INFO placement.MultiNodeSortingManager: Starting NodeSortingService=MultiNodeSortingManager
2022-10-11 08:42:12,536 INFO ipc.CallQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.hadoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2022-10-11 08:42:12,546 INFO ipc.Server: Starting Socket Reader #0 for port 8031
2022-10-11 08:42:12,549 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.server.api.ResourceTrackerPB to the server
2022-10-11 08:42:12,550 INFO ipc.Server: IPC Server Responder: starting
2022-10-11 08:42:12,550 INFO ipc.Server: IPC Server listener on 8031: starting
2022-10-11 08:42:12,577 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2022-10-11 08:42:12,598 INFO ipc.CallQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.hadoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2022-10-11 08:42:12,618 INFO ipc.Server: Starting Socket Reader #1 for port 8030
2022-10-11 08:42:12,647 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.api.ApplicationMasterProtocolPB to the server
2022-10-11 08:42:12,713 INFO ipc.Server: IPC Server Responder: starting
2022-10-11 08:42:12,716 INFO ipc.Server: IPC Server listener on 8030: starting
2022-10-11 08:42:12,948 INFO ipc.CallQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.hadoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2022-10-11 08:42:12,954 INFO ipc.Server: Starting Socket Reader #1 for port 8032
2022-10-11 08:42:12,962 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.api.ApplicationClientProtocolPB to the server
2022-10-11 08:42:12,963 INFO ipc.Server: IPC Server Responder: starting
2022-10-11 08:42:12,964 INFO ipc.Server: IPC Server listener on 8032: starting
2022-10-11 08:42:12,976 INFO resourcemanager.ResourceManager: Transitioned to active state
2022-10-11 08:42:14,840 INFO resourcemanager.ResourceTrackerService: NodeManager from node a4it55.KC.VNR(cmPort: 62038 httpPort: 8042) registered with capability: <memory:8192, vcores:8>, assigned nodeid a4it55.KC.VNR:62038
2022-10-11 08:42:14,849 INFO rmnode.RMNodeImpl: a4it55.KC.VNR:62038 Node Transitioned from NEW to RUNNING
2022-10-11 08:42:14,890 INFO capacity.CapacityScheduler: Added node a4it55.KC.VNR:62038 clusterResource: <memory:8192, vcores:8>
```



```
Apache Hadoop Distribution - yarn nodemanager
INFO: Registering org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResolver as a provider class
Oct 11, 2022 8:42:12 AM com.sun.jersey.server.impl.application.WebApplicationImpl _initiate
INFO: Initiating Jersey application, version 'Jersey: 1.19 02/11/2015 03:25 AM'
Oct 11, 2022 8:42:13 AM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResolver to GuiceManagedComponentProvider with the scope "Singleton"
Oct 11, 2022 8:42:13 AM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.webapp.GenericExceptionHandler to GuiceManagedComponentProvider with the scope "Singleton"
Oct 11, 2022 8:42:14 AM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.NMWebServices to GuiceManagedComponentProvider with the scope "Singleton"
2022-10-11 08:42:14,199 INFO handler.ContextHandler: Started o.e.j.w.WebApplicationContext@3b1ed14b{node/,file:///C:/Users/Administrator/AppData/Local/Temp/jetty-0_0_0-8042_-any-6557969691349594047.dir/webapp/,AVAILABLE}{jar:file:/D:/hadoop-env/hadoop-3.2.2/share/hadoop/yarn/hadoop-yarn-common-3.2.2.jar!/webapps/node}
2022-10-11 08:42:14,218 INFO server.AbstractConnector: Started ServerConnector@2d10e0b1{HTTP/1.1,[http/1.1]}{0.0.0.0:8042}
2022-10-11 08:42:14,218 INFO server.Server: Started @12072ms
2022-10-11 08:42:14,220 INFO webapp.WebApps: Web app node started at 8042
2022-10-11 08:42:14,224 INFO nodemanager.NodeStatusUpdaterImpl: Node ID assigned is : a4it55.KC.VNR:62038
2022-10-11 08:42:14,230 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2022-10-11 08:42:14,240 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8031
2022-10-11 08:42:14,380 INFO nodemanager.NodeStatusUpdaterImpl: Registering with RM using containers :[]
2022-10-11 08:42:14,877 INFO security.NMContainerTokenSecretManager: Rolling master-key for container-tokens, got key with id -324326809
2022-10-11 08:42:14,879 INFO security.NMTokenSecretManagerInNM: Rolling master-key for container-tokens, got key with id 123186820
2022-10-11 08:42:14,882 INFO nodemanager.NodeStatusUpdaterImpl: Registered with ResourceManager as a4it55.KC.VNR:62038 with total resource of <memory:8192, vcores:8>
```

To make sure that all services started successfully, Run the following command:

jps

It should display the following services:

14560 DataNode

4960 ResourceManager

5936 NameNode

768 NodeManager

14636 Jps

```

PS D:\hadoop-env\hadoop-3.2.2\sbin> jps
4288 Jps
4896 ResourceManager
4708 NameNode
5852 NodeManager
6860 DataNode
PS D:\hadoop-env\hadoop-3.2.2\sbin>

```

Output

Hadoop Web UI

<http://localhost:9870/dfshealth.html>

Overview 'localhost:9820' (active)

Started:	Tue Oct 11 08:35:24 +0530 2022
Version:	3.2.2, r7a3bc90b05f257c8ace2f76d74264906f0f7a932
Compiled:	Sun Jan 03 14:56:00 +0530 2021 by hexiaoqiao from branch-3.2.2
Cluster ID:	CID-d536af7a-e745-4eaf-8c23-158b0783e6db
Block Pool ID:	BP-1402821420-172.16.8.55-1665456682328

Summary

Security is off.
Safemode is off.

1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).

Heap Memory used 64.55 MB of 207.5 MB Heap Memory. Max Heap Memory is 889 MB.

Non Heap Memory used 61.01 MB of 62.52 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Activate Windows
Go to Settings to activate Windows.

<http://localhost:9864/datanode.html>

DataNode on a4it55.KC.VNR:9866

Cluster ID:	CID-d536af7a-e745-4eaf-8c23-158b0783e6db
Version:	3.2.2, r7a3bc90b05f257c8ace2f76d74264906f0f7a932

Block Pools

Namenode Address	Block Pool ID	Actor State	Last Heartbeat	Last Block Report	Last Block Report Size (Max Size)
localhost:9820	BP-1402821420-172.16.8.55-1665456682328	RUNNING	2s	19 minutes	0 B (64 MB)

Volume Information

Activate Windows
Go to Settings to activate Windows.

<http://localhost:8088/cluster>

The screenshot shows the Hadoop Cluster Management UI at <http://localhost:8088/cluster>. The main title is "All Applications". On the left, there's a sidebar with "Cluster Metrics" and "Node Labels Applications" sections. The "Node Labels Applications" section lists states: NEW, NEW_SAVING, SUBMITTED, ACCEPTED, RUNNING, FINISHED, FAILED, KILLED. Below it is a "Scheduler Metrics" section. The main area displays "Cluster Metrics" with values: Apps Submitted: 0, Apps Pending: 0, Apps Running: 0, Apps Completed: 0, Containers Running: 0, Used Resources: <memory:0, vCores:0>, Total Resources: <memory:8192, vCores:8>, Reserved Resources: <memory:0, vCores:0>. It also shows "Cluster Nodes Metrics" with Active Nodes: 1, Decommissioning Nodes: 0, Decommissioned Nodes: 0, Lost Nodes: 0, Unhealthy Nodes: 0. Under "Scheduler Metrics", it shows Scheduler Type: Capacity Scheduler, Scheduling Resource Type: [memory-mb (unit=Mi), vcores], Minimum Allocation: <memory:1024, vCores:1>, Maximum Allocation: <memory:8192, vCores:4>. A table header for "Applications" is shown with columns: ID, User, Name, Application Type, Queue, Application Priority, StartTime, LaunchTime, FinishTime, State, FinalStatus, Running Containers, Allocated CPU Vcores, Allocated Memory MB, Allocated GPUs, Reserved CPU Vcores, Reserved Memory MB. The message "Showing 0 to 0 of 0 entries" is displayed. At the bottom right, there's a Windows taskbar with icons for Start, Search, Task View, File Explorer, Mail, Photos, OneDrive, Edge, Google Chrome, Word, Excel, Settings, Paint, File Explorer, and a few others. The date and time are 11-10-2022 09:20.

VIVA QUESTIONS

1. What is Hadoop?

2. State the types of data.

3. Name three components of Hadoop.

4. In Hadoop, _____ is used for processing / computing the task.

5. In Hadoop, _____ provides a distributed file system that is designed to run on commodity hardware.