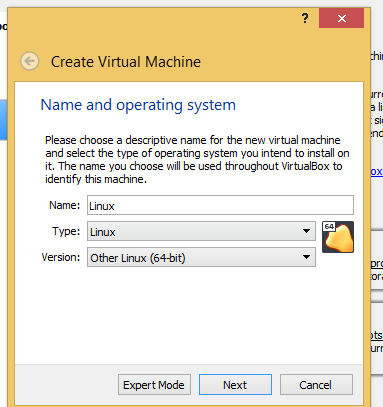
**A. Installation Cloudera QuickStartVM**

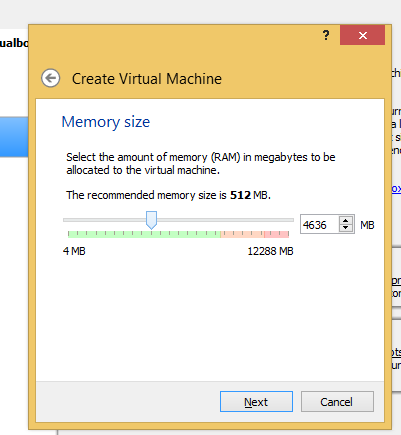
1. Download **Cloudera QuickStartVM from the below link**

<https://drive.google.com/file/d/1kgMDGeOKhDJ21EEqYOMaUzhndU71GuOL/view>

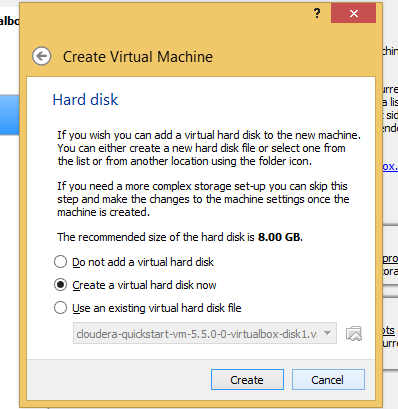
**2. Choose Linux machine 64 bit and Ram according to your machine capacity. If you have 8 GB ram**

**choose 4 GB RAM.**

**b**

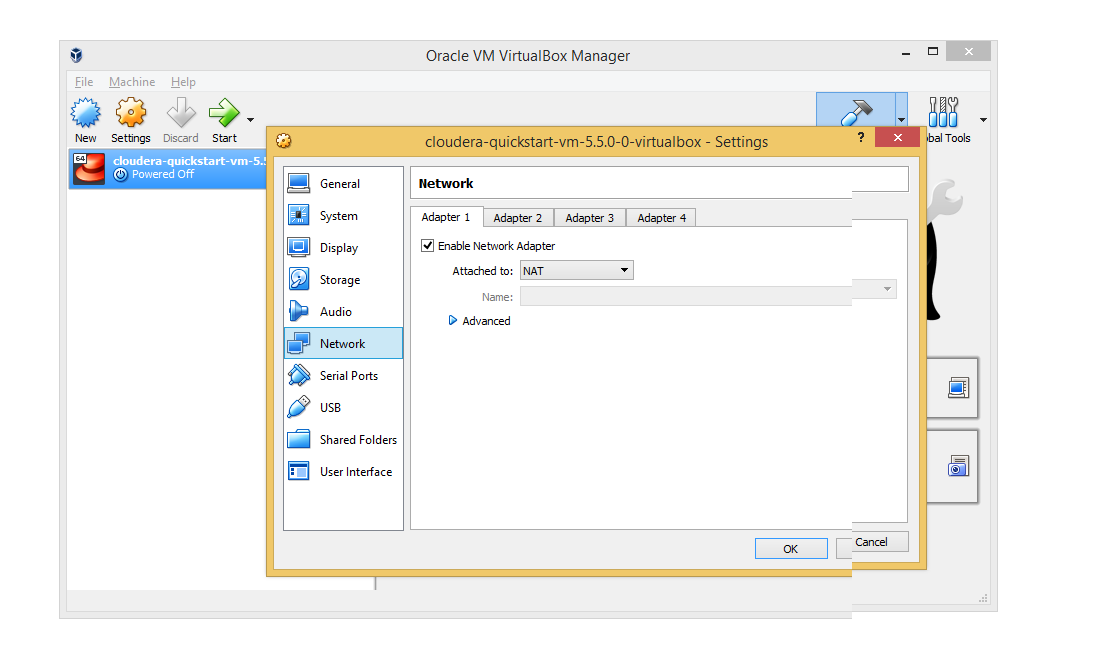
****

**Choose the Cloudera image you have downloaded as per below**

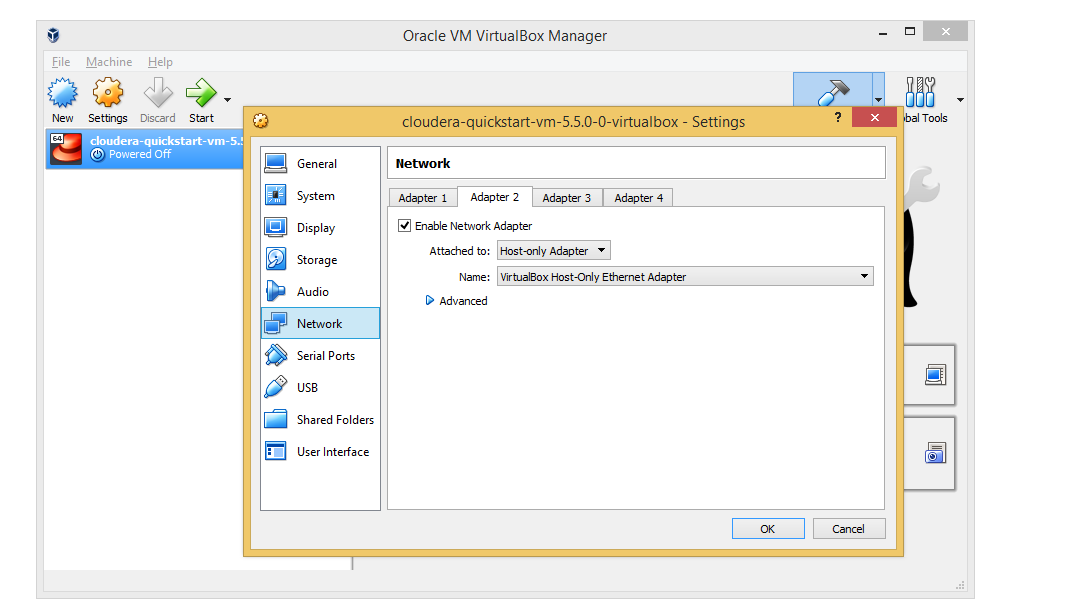
****

**3. Configuring the settings of Virtual Machine to transfer files from your windows machine and access internet**

**a). Add NAT adapter in your network settings for internet**

****

**b). Add host only adapter in your network settings to access your windows machine**

****

**4. Transferring the MR JAR file to Virtual Machine**

To transfer the file from your windows machine to a virtual machine follow the steps from 7 to 14

[**https://data-flair.training/blogs/transfer-data-from-windows-to-ubuntu/**](https://data-flair.training/blogs/transfer-data-from-windows-to-ubuntu/)

**5. Connecting to hadoop Cloudera vm / Any other lab instance available in the net using Putty**

*Watch the below video from timeline 0:14*

<https://www.youtube.com/watch?v=h_2txcRETHo>

Note:

Putty is helpful when you have low RAM in your windows machine using Cloudera quick start vm will become your machine slow.

**6.Accessing Hue from your windows machine.**

To check the successful execution of the Job you can use the Hue interface with the application id.

In earlier links i have explained how to find the ipaddress of your Cloudera quick start vm using ifconfig.

Use that ip and form the below url

[http://<ipaddress>:8888/](http://192.168.56.102:8888/about/)

Default user id / password: Cloudera /Cloudera

You will be able to access Hue from windows.

7. Resource Manager

[http://<ipaddress>:8088/](http://192.168.56.102:8888/about/)

8. Job history Server

[http://<ipaddress>:19888/](http://192.168.56.102:8888/about/)

B. Development of Jar

Refer documentation - Setting up\_Hadoop\_Project\_Eclipse document

C. Deployment and Execution of Jar

Execute the Jar with the below command

yarn jar MR\_WordCnt.jar WCDriver webin/ out/ -Dlog4j.configuration=log4j.properties -D mapred.map.tasks = 1

or

hadoop jar MR\_WordCnt.jar WCDriver webin/ out/ -Dlog4j.configuration=log4j.properties -D mapred.map.tasks = 1

jarName: MR\_WordCnt.jar

className: WCDriver

in inputfilePathInhdfs

out outputfilePathInhdfs

-Dlog4j.configuration=log4j.properties - This file should be their where you are running the jar, so it overrides other log4j.property file available in hadoop by default.

-D mapred.map.tasks = 1 - Map task is one so it is easy to debug. Otherwise there would be multiple map task if your file is greater than 128MB(default block size).

7. The input files needs to be available in hdfs. so the local file input.txt needs to be moved to hdfs using the below command

# create the in directory in hdfs

**hdfs fs -mkdir webin**

# move the file from local file system to hdfs

**hdfs fs -put input.txt in/**

hdfs fs -ls /user/cloudera/in

9. Check the job status and the log files for errors in Hue using the job browser with the application ID created on submission. You and can get that application ID from the console itself after starting your MapReduce job.

It will be similar to as shown below.

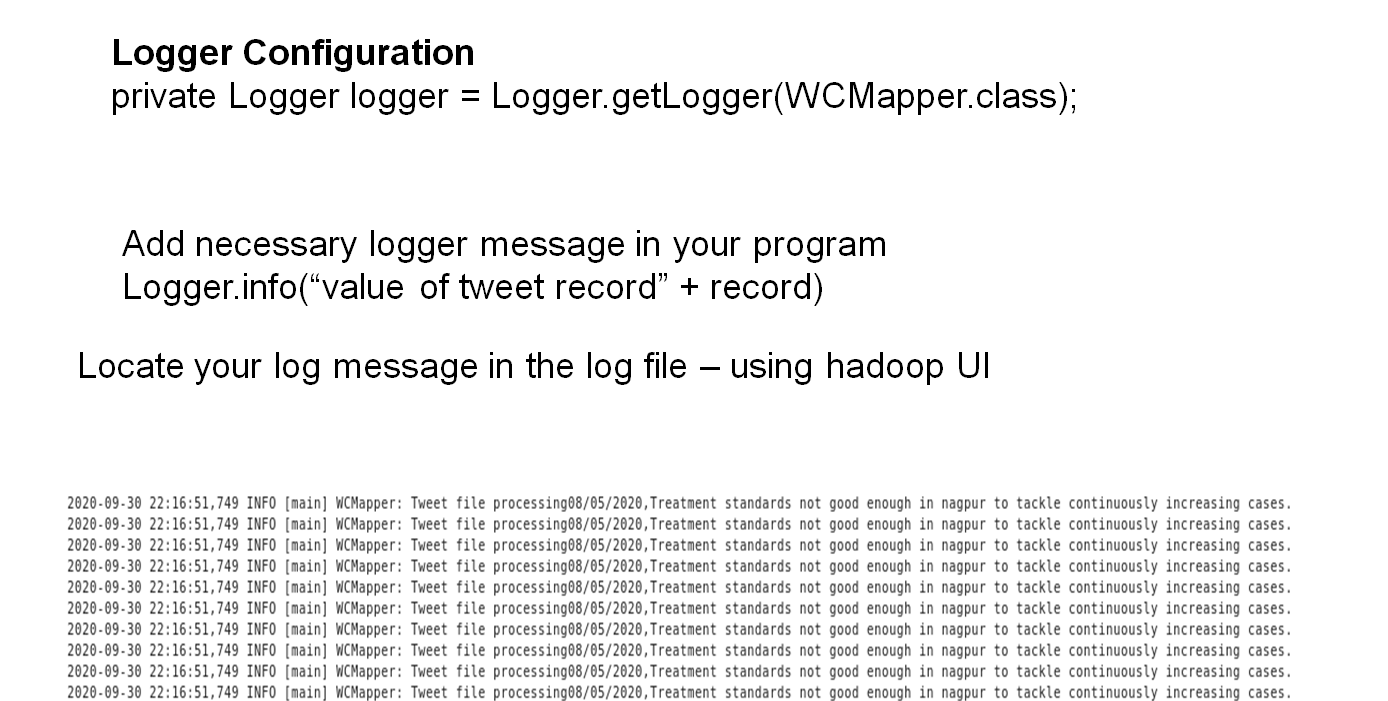
18/07/11 14:39:23 INFO impl.YarnClientImpl: Submitted application application\_1531299441901\_0001

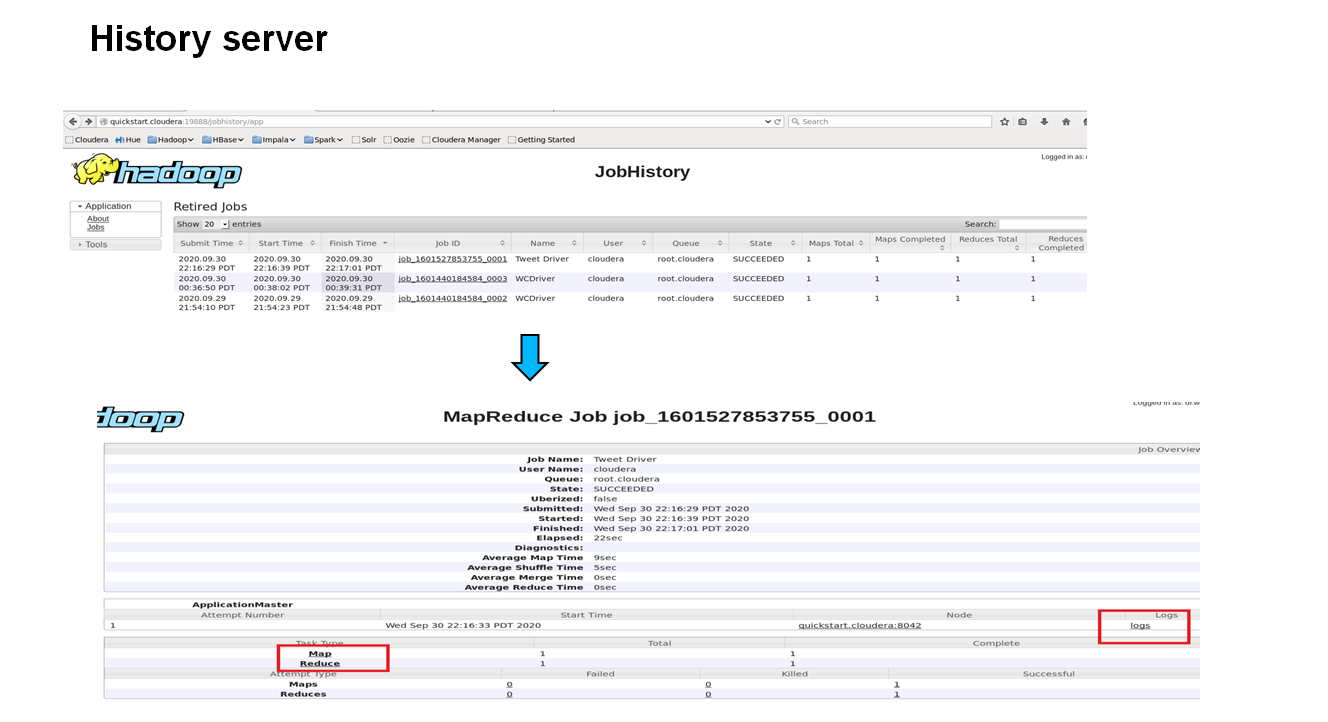
9. On successful execution check the output file in the hdfs out directory mentioned while running the MR job

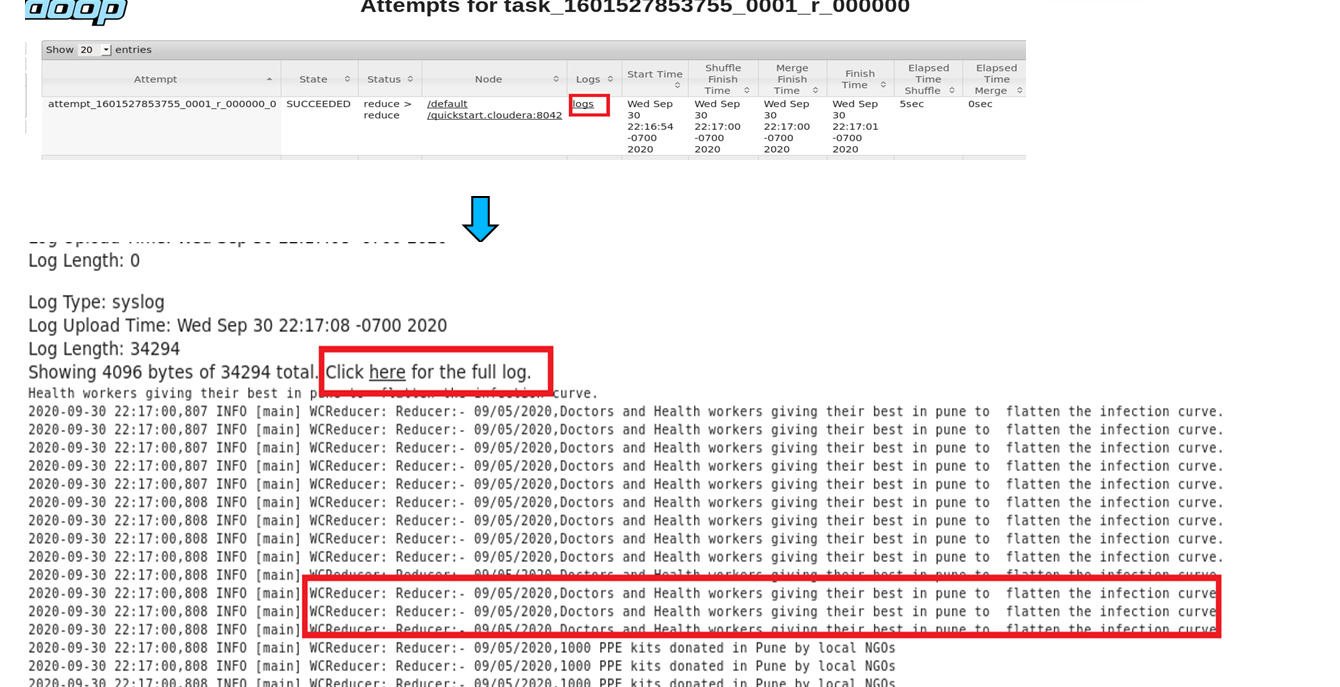
**hdfs dfs -cat /out/part-r-00000**

**Sample Output file: part-r-00000**

D. Debugging your mapreduce code - webUI







E. Debugging your code from console

A ll log files will be stored in 7

/var/log/hadoop-yarn/apps/cloudera/logs/application\_1590089433422\_0049/quickstart.cloudera\_48277