Experiment 2 : Implement word count program using Map Reduce.

Steps to run WordCount Program on Hadoop:

1. Make sure Hadoop and Java are installed properly

```
hadoop version
```

```
javac -version
```

2. Create a directory on the Desktop named Lab and inside it create two folders; one called “Input” and the other called “tutorial_classes”.

[You can do this step using GUI normally or through terminal commands]

```
cd Desktop
```

```
mkdir Lab
```

```
mkdir Lab/Input
```

```
mkdir Lab/BDA
```

3. Add the file attached with this document “WordCount.java” in the directory Lab

4. Add the file attached with this document “input.txt” in the directory Lab/Input.

5. Type the following command to export the hadoop classpath into bash.

```
export HADOOP_CLASSPATH=$(hadoop classpath)
```

Make sure it is now exported.

```
echo $HADOOP_CLASSPATH
```

7. Go to **localhost:9870** from the browser, Open “Utilities → Browse File System” and you should see the directories and files we placed in the file system.

8. Then, back to local machine where we will compile the WordCount.java file. Assuming we are currently in the Desktop directory.

9. Start the HDFS System using the command.

```
start-dfs.sh
```

10. Start the YARN using the command

```
start-yarn.sh
```

11. Type the following command. You should see an output similar to the one in the following figure.

```
jps
```

12. Make sure these nodes are listed: (ResourceManager, NameNode, NodeManager, SecondaryNameNode, Jps and DataNode).

13. It is time to create these directories on HDFS rather than locally. Type the following commands.

```
hadoop fs -mkdir /WordCountProgram
```

```
hadoop fs -mkdir /WordCountProgram/Input
```

```
hadoop fs -put Lab/Input/input.txt /WordCountProgram/Input
```

14. Go to localhost:9870 from the browser. You should expect the following

```
cd Lab
```

```
javac -classpath $HADOOP_CLASSPATH -d BDA WordCount.java
```
Put the output files in one jar file (There is a dot at the end)

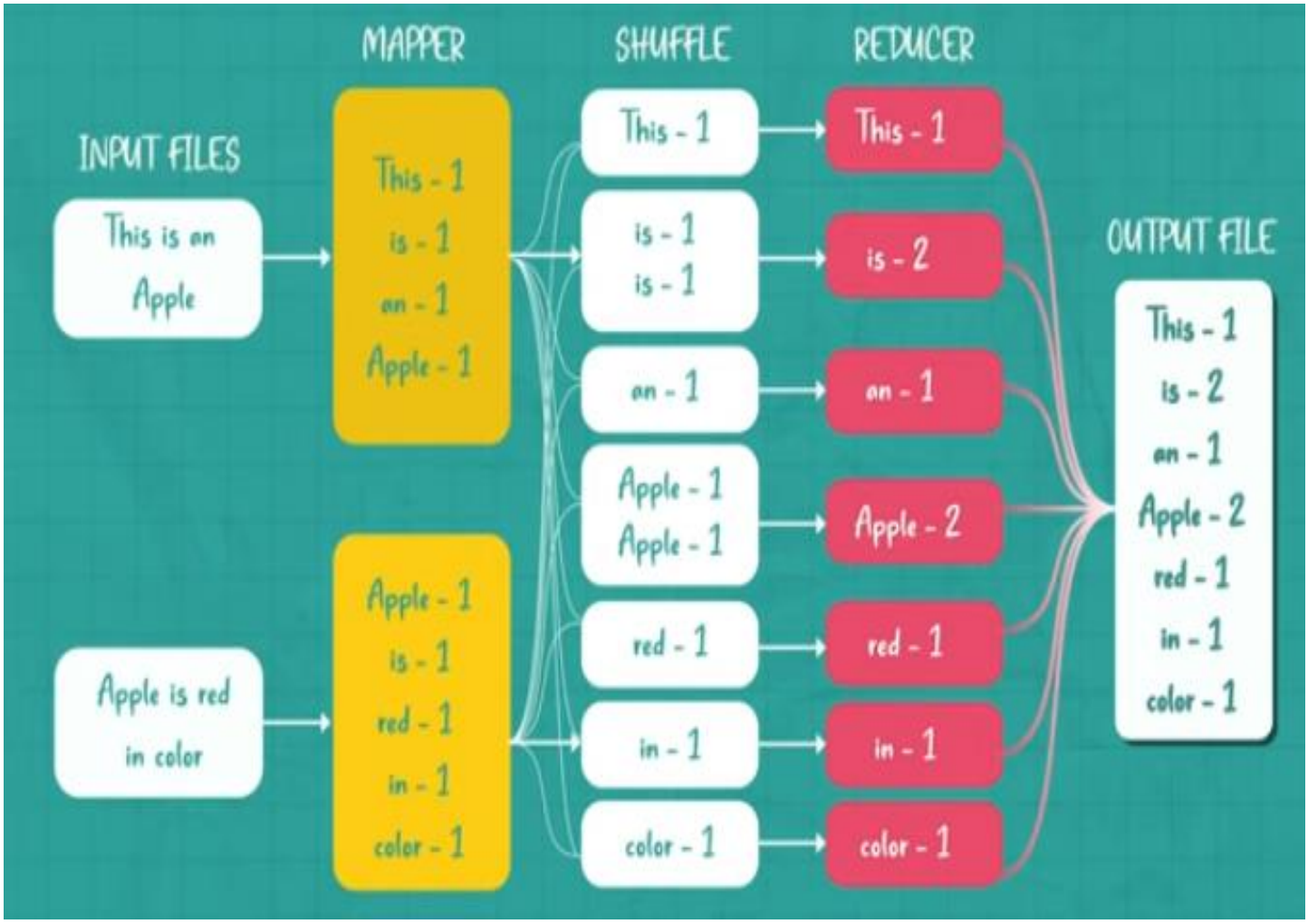
```
jar -cvf WordCount.jar -C BDA .
```

15. Now, we run the jar file on Hadoop.

```
hadoop jar WordCount.jar WordCount /WordCountProgram/Input
/WordCountProgram/Output
```

16. Output the result:

```
hadoop dfs -cat /WordCountProgram/Output/*
```


INPUT FILES
This is an Apple
Apple is red in color
MAPPER
This - 1
is - 1
an - 1
Apple - 1
Apple - 1
is - 1
red - 1
in - 1
color - 1
SHUFFLE
This - 1
is - 1
is - 1
an - 1
Apple - 1
Apple - 1
red - 1
in - 1
color - 1
REDUCER
This - 1
is - 2
an - 1
Apple - 2
red - 1
in - 1
color - 1
OUTPUT FILE
This - 1
is - 2
an - 1
Apple - 2
red - 1
in - 1
color - 1