**Steps to run WordCount Program on Hadoop:**

1. Make sure Hadoop and Java are installed properly

**hadoop version**

**javac -version**

1. 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/tutorial\_classes**

1. Add the file attached with this document “WordCount.java” in the directory Lab
2. Add the file attached with this document “input.txt” in the directory Lab/Input.
3. 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**

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

**hadoop fs -mkdir /WordCountTutorial**

**hadoop fs -mkdir /WordCountTutorial/Input**

**hadoop fs -put Lab/Input/input.txt /WordCountTutorial/Input**

1. 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.
2. Then, back to local machine where we will compile the WordCount.java file. Assuming we are currently in the Desktop directory.

**cd Lab**

**javac -classpath $HADOOP\_CLASSPATH -d tutorial\_classes WordCount.java**

1. Put the output files in one jar file (There is a dot at the end)

**jar -cvf WordCount.jar -C tutorial\_classes .**

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

**hadoop jar WordCount.jar WordCount /WordCountTutorial/Input /WordCountTutorial/Output**

The command you provided is attempting to run a Hadoop job using a JAR file named WordCount.jar with the main class WordCount. The job is supposed to process input data located in /WordCountTutorial/Input and output the results to /WordCountTutorial/Output.

1. Output the result:

**hadoop dfs -cat /WordCountTutorial/Output/\***

The cat command in Unix-like operating systems is used to concatenate and display the contents of files. It's one of the most frequently used commands in Unix-like systems and is quite versatile.

**Requirement:**

Vodafone Egypt is launching a marketing campaign in Ramadan to promote their sales and increase their profit from selling the prepaid recharge cards. These cards are worth 5, 10, 15, 50, and 100 EGP.

The data science team at Vodafone are analyzing the customers’ data which include the customer personal information, the prepaid card they purchased, the timestamp they registered the prepaid amount on their Vodafone accounts, among other information.

The details of the customers are omitted, and you are only provided with a file “in.csv” which includes two columns.

1. Customer ID. (Each ID maps to a certain customer, whose data is hidden for confidentiality).
2. Prepaid Card Amount.

Your task is to generate a report using MapReduce (similar to the WordCount program) showing the total amount of prepaid cards for each customer that they have purchased. For example, if a customer with ID 300 purchased 5 cards with 10, 15, 15, 10, 100, then the report should include that customer ID 300 bought cards with a total amount of 150.



*Disclaimer: Thanks to Vodafone DS team who provided us with this real customer data.*