Atelier 8

```
hadoop@maram-VirtualBox:~$ export HADOOP_CLASSPATH=$($HADOOP_HOME/bin/hadoop classp
ath)
hadoop@maram-VirtualBox:~$ echo $HADOOP_CLASSPATH
/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/*:/usr/local
/hadoop/share/hadoop/common/*:/usr/local/hadoop/share/hadoop/hdfs:/usr/local/hadoop
/share/hadoop/hdfs/lib/*:/usr/local/hadoop/share/hadoop/hdfs/*:/usr/local/hadoop/sh
are/hadoop/mapreduce/*:/usr/local/hadoop/share/hadoop/yarn:/usr/local/hadoop/share/
hadoop/yarn/lib/*:/usr/local/hadoop/share/hadoop/yarn/*:/usr/local/hadoop/lib/javax
.activation-api-1.2.0.jar
```

compilateur `javac` de savoir où trouver les classes et packages Hadoop
Et ensuite faire l'affiche le contenu de la variable `HADOOP_CLASSPATH` pour s'assurer qu'elle est bien initialisée.

```
hadoop@maram-VirtualBox:~$ javac -classpath $HADOOP_CLASSPATH -d WCClasses WordCoun
t.java
hadoop@maram-VirtualBox:~$ jar -cvf WordCount.jar -C WCClasses/ .
added manifest
adding: WordCount$IntSumReducer.class(in = 1793) (out= 777)(deflated 56%)
adding: WordCount$TokenizerMapper.class(in = 1790) (out= 792)(deflated 55%)
adding: WordCount.class(in = 1511) (out= 832)(deflated 44%)
```

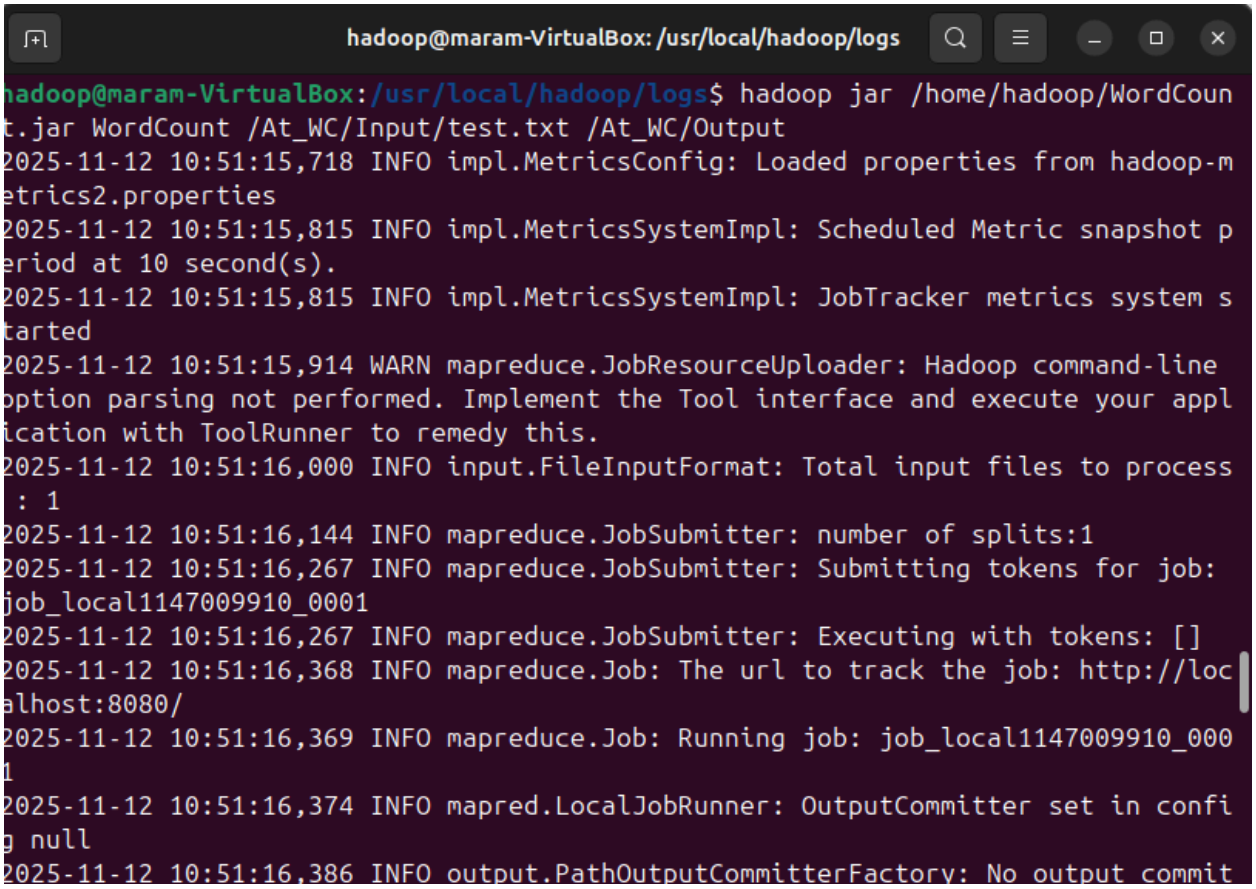
après avoir écrit le programme *WordCount.java*, j'ai procédé à sa compilation et à la création du fichier exécutable JAR.

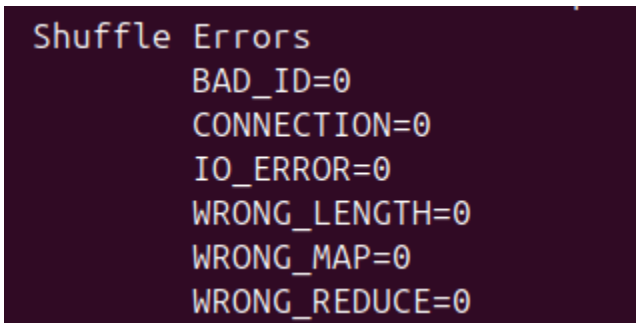
javac -classpath $HADOOP_CLASSPATH -d WCClasses WordCount.java

m'a permis de compiler le code source Java en utilisant les bibliothèques Hadoop spécifiées dans la variable d'environnement $HADOOP_CLASSPATH. Les fichiers .class générés ont été placés dans le dossier WCClasses, assurant une bonne organisation du projet.
Ensuite, la commande:

jar -cvf WordCount.jar -C WCClasses/ .

a servi à regrouper les fichiers compilés dans une archive JAR nommée `WordCount.jar`.
Ce fichier représente le programme MapReduce complet, prêt à être exécuté sur le cluster Hadoop.





Ici, le fichier `WordCount.jar` contient le code compilé du programme, et `WordCount` est le nom de la classe principale à exécuter. Le chemin `/At_WC/Input/test.txt` désigne le fichier d'entrée stocké dans le système de fichiers distribué HDFS, tandis que `/At_WC/Output` correspond au répertoire de sortie où les résultats du traitement seront enregistrés.

affichage de contenu des fichiers de sortie générés par le job MapReduce dans le système de fichiers HDFS.