Hadoop, Spark and HIVE on a AWS Hadoop cluster

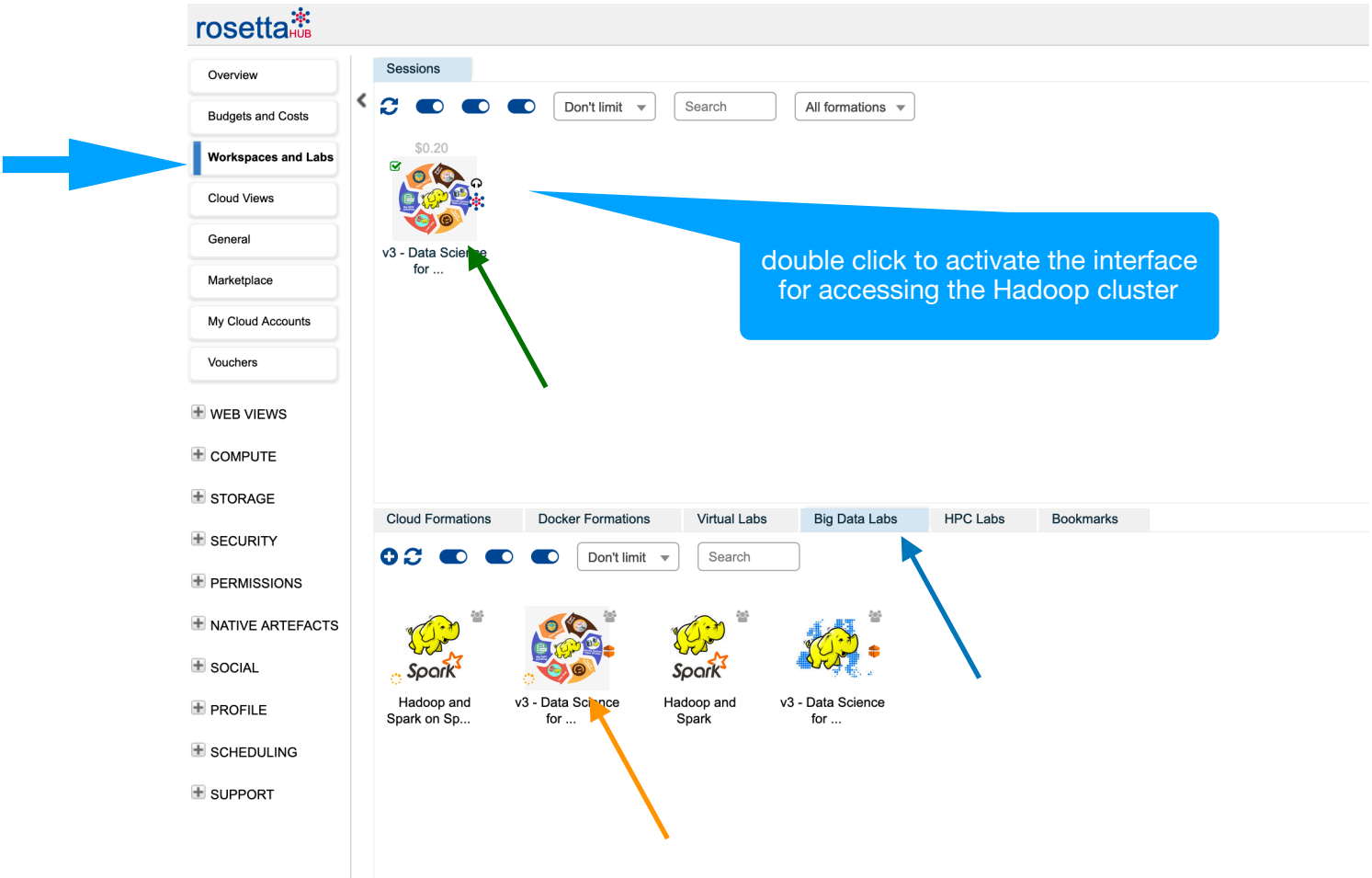
Dario Colazzo

First step. You should have received an email from RosettaHUB inviting you to set your password. Please follow the instructions.

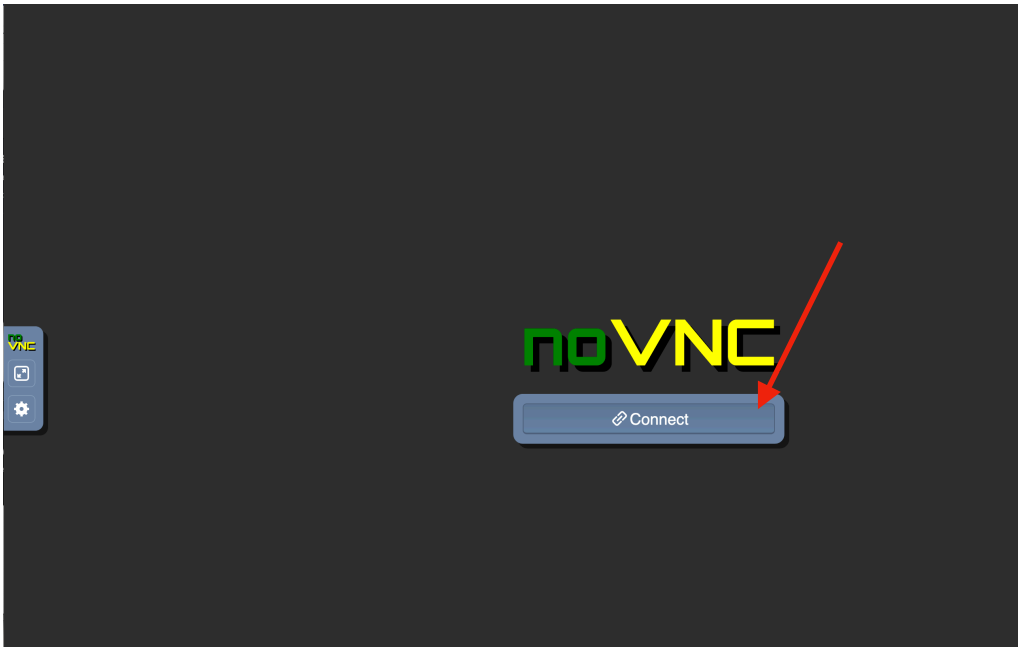
In case you have subscribed directly, just use login and pw to connect.

Using your AWS cluster. Go to the RosettaHUB page (preferably by means of Chrome or Firefox)
(**https://www.rosettahub.com/welcome**) and click on Sign in to connect, by using thw chosen password.

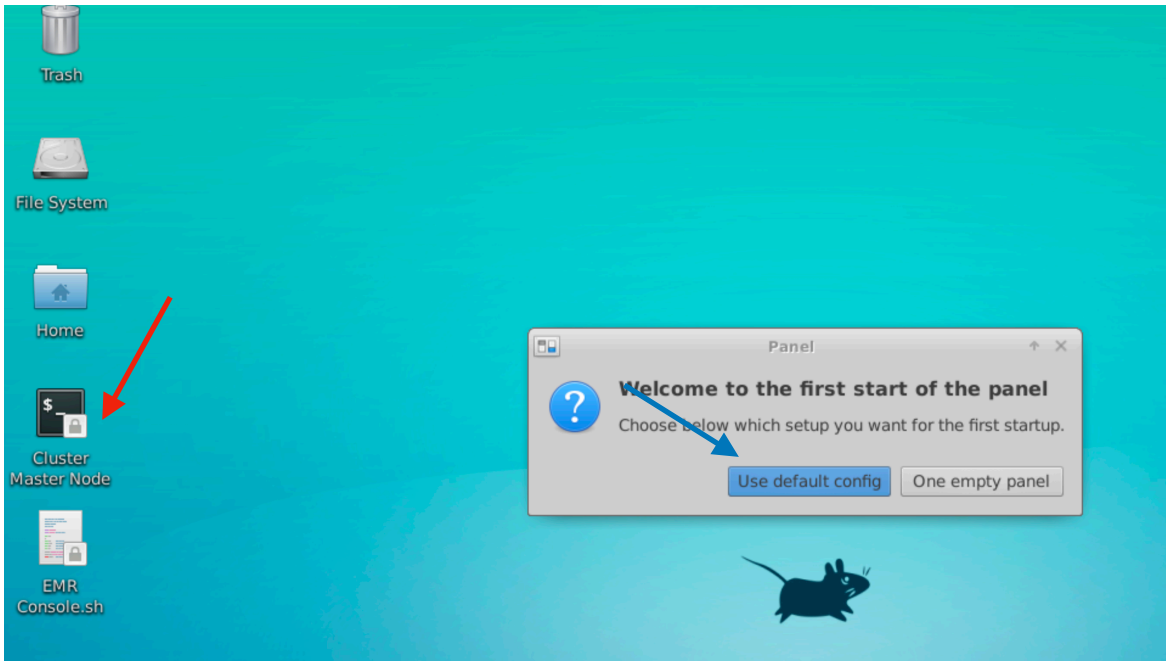
First, bouble click on the icon indicated by the orange arrow. After 10 minutes or so, you will have the icon indicated by the green arrow, this indicates that the cluster is running and ready. Now, click on the indicated icon in order to access a web interface.

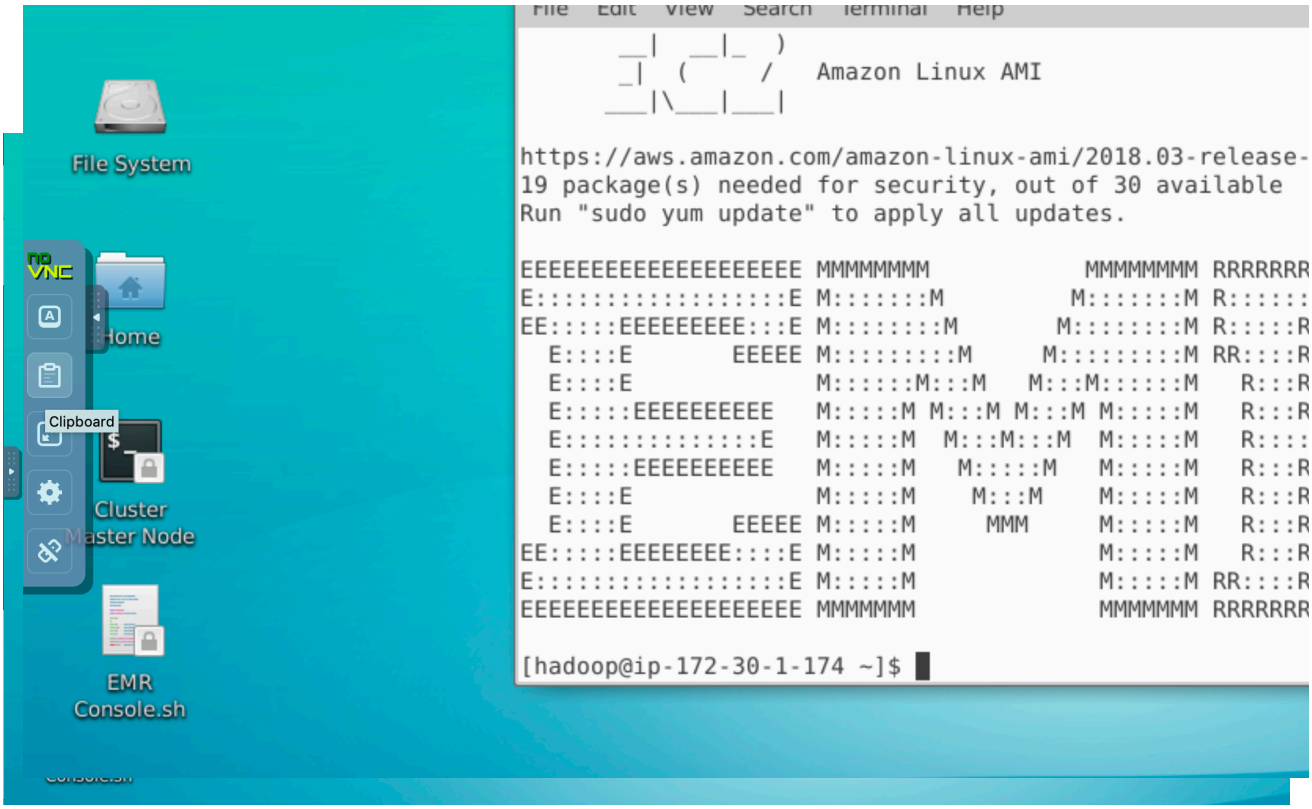


Double-click on the Cluster icon, you will see a new Tab as below:



Click on the Connect button (red arrow above), and you will see a Linux desktop, as follows

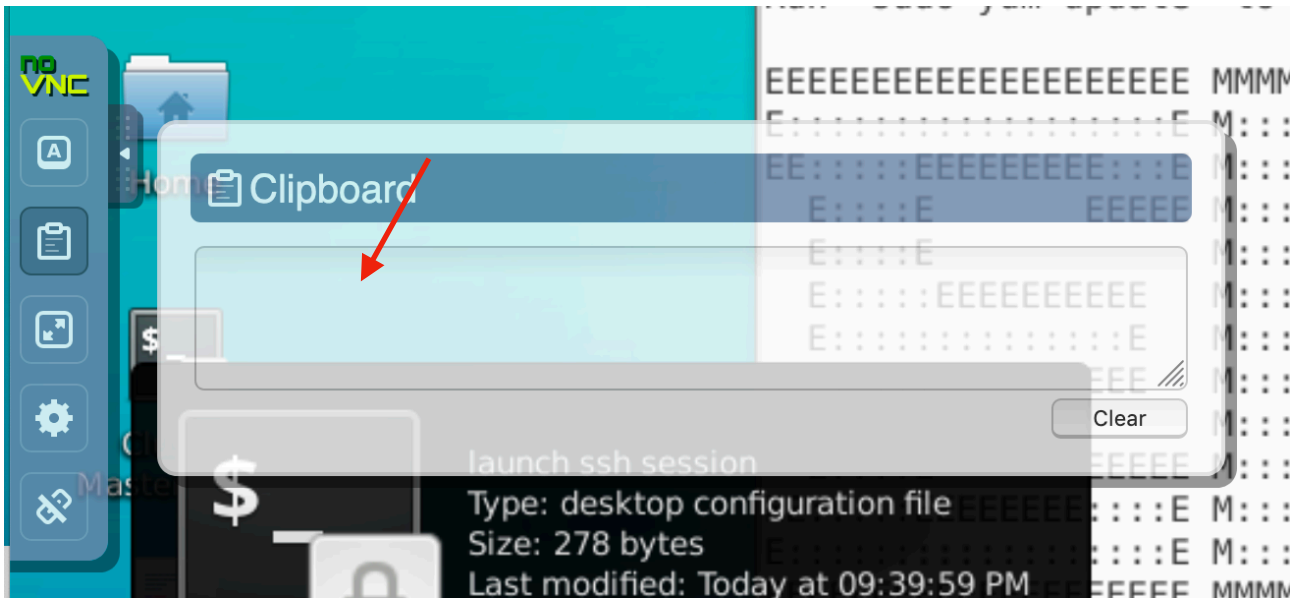


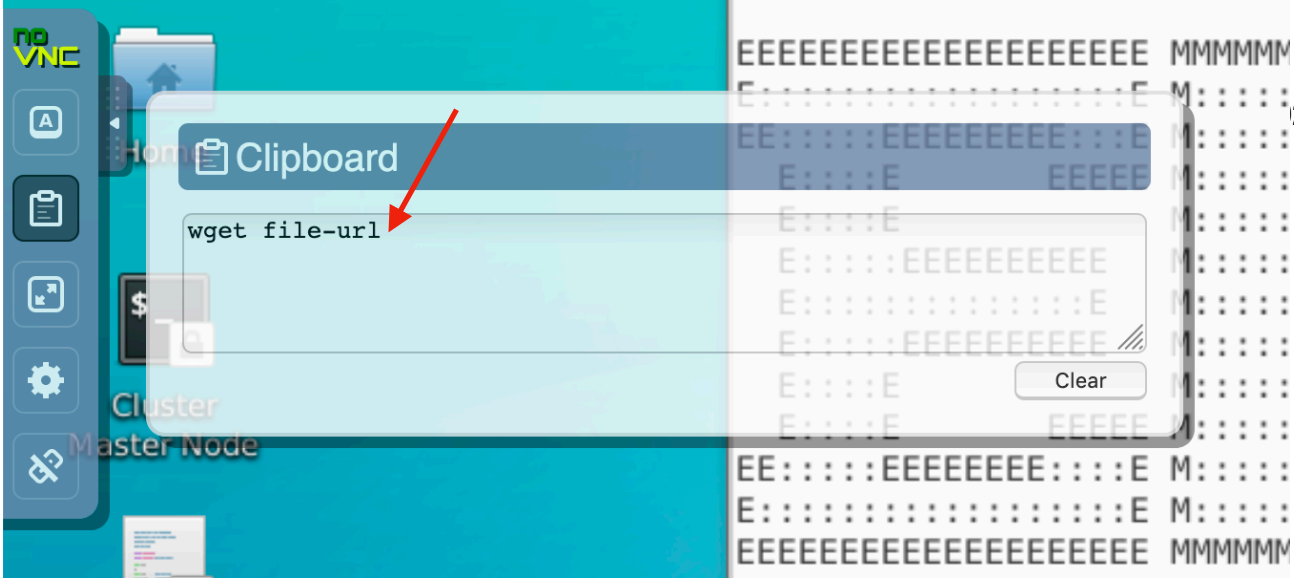


Click on 'Use default config.', then double click on the Cluster Muster Node (red arrow).

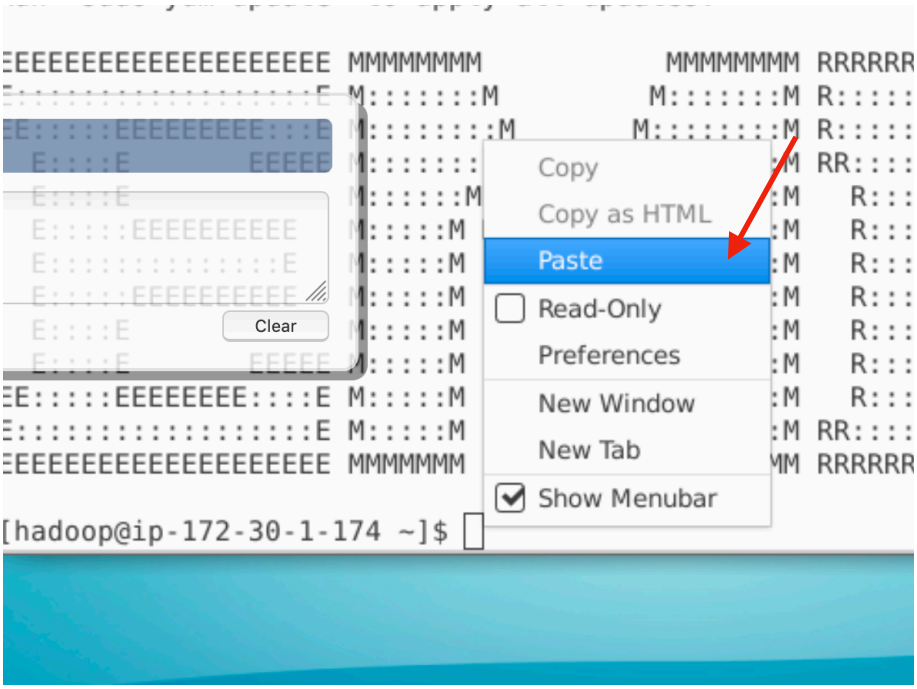
You have now access to an AWS Elastic Map Reduce cluster, and you can execute HDFS and MapReduce-related commands, as you will see.

Attention : in our lab-sessions we will need to copy-paste commands provided in the pdf sheet into the Linux command line (for instance wget command)
Since this EMR interface is not directly open to copy-paste of your operating system on you laptop, you have first to copy-paste into a clipboard space see below

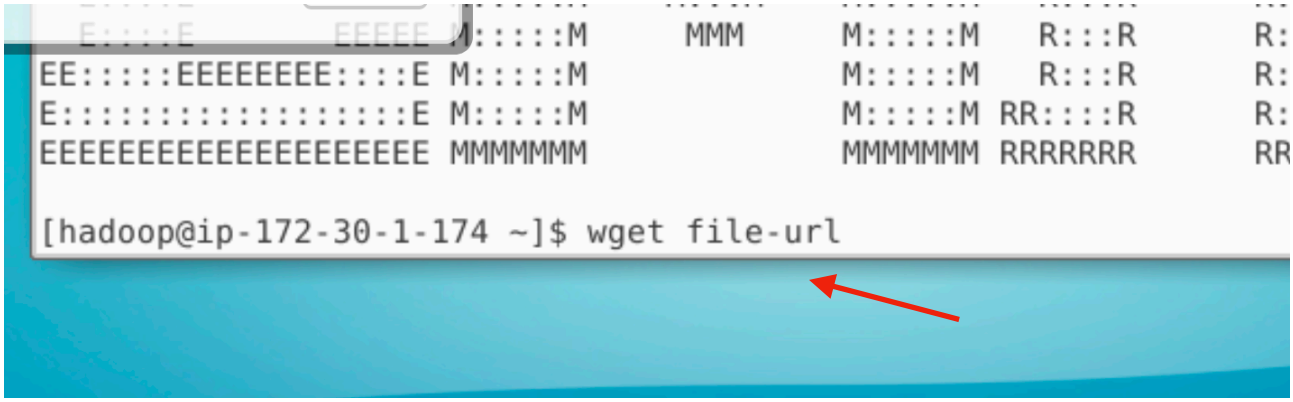




Just copy-paste first inside the clipoard opened by double-clicking the clipboard icon :



Go then on the EMR prompt in the Linux terminal, and right-click on Paste



And you will have the command in wget command in the terminal prompt:

At this point you can use HDFS (see next step) and run MapReduce jobs coded in Python, for instance the WordCount job as indicated in the following.

Your login user name is 'hadoop', and in HDFS your home is at this path: /user/hadoop

Useful commands for Linux and HDFS. See the following documents that we borrowed from the Web:

https://www.dropbox.com/s/77z7m35tfe18jyr/hadoop-hdfs-commands-cheatsheet.pdf

https://www.dropbox.com/s/06qw849h2omjke0/fwunixref.pdf

Run WordCount on the EMR Cluster.

Once connected with the AWS cluster via ssh, proceed with the following preliminary steps

1. Download the mapper, reducer and a text file in you home directory, with the following commands (copy-paste them, by following the previous procedure)

>wget https://www.dropbox.com/s/471k0286292ifho/mapper.py

> wget https://www.dropbox.com/s/e8s6f7rsiwts84m/reducer.py

Concerning the input text file:

> wget https://www.dropbox.com/s/hj8khqc94vsrzw9/shake.txt

2. Give the execution right to the .py files.

chmod +x *.py

3. You will need the absolute paths for the three files, to get the path leading your current directory use the pwd command.

4. Now you need the create a directory in you HDFS home for word count. Since your user name is ‘hadoop’ and your HDFS home directory is ‘/user/hadoop’ you will use the following command to build the ‘wc’ directory under your HDFS home.

hdfs dfs -mkdir /user/hadoop/wc

5. Now you need to create a ‘input’ subdirectory containing the input fille for job

hdfs dfs -mkdir /user/hadoop/wc/input

6. Finally, you need to transfer the input file from the local file system to the HDFS file system so that MapReduce can process it.

hdfs dfs -put shake.txt /user/hadoop/wc/input

7. You are ready now to launch our MapReduce job. Copy this command to a text editor, indicate the paths to needed files and then run it on the open cluster terminal. You will get several statistics. The job results are in the ‘output’ HDFS directory. Each reduce task has produced its own file result.

```
hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-input  /user/hadoop/wc/input \
-output  /user/hadoop/wc/output \
-file /home/hadoop/mapper.py \
-mapper /home/hadoop/mapper.py \
-file  /home/hadoop/reducer.py \
-reducer /home/hadoop/reducer.py
```

Note that we are assuming that Python programs are in your home directory /home/hadoop/ in the local file system (not on HDFS), otherwise you have to change this path.

If you want to use the combiner then use this variant (note that you re-use the reducer as a combiner)

```
hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-input  /user/hadoop/wc/input \
-output  /user/hadoop/wc/output4 \
-file /home/hadoop/mapper.py \
-mapper /home/hadoop/mapper.py \
-file  /home/hadoop/reducer.py \
```

-reducer /home/hadoop/reducer.py \
-combiner /home/hadoop/reducer.py

Of course if your combiner is not the reducer, then you have to specify both -file and -combiner parameters with the same path for the .py file including the combiner.

This version is to set a particular number of reduce tasks (e.g., 3 reduce tasks)

hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-input /user/hadoop/wc/input \
-output /user/hadoop/wc/output \
-file /home/hadoop/mapper.py \
-mapper /home/hadoop/mapper.py \
-file /home/hadoop/reducer.py \
-reducer /home/hadoop/reducer.py \
-jobconf mapred.reduce.tasks=3

8. You can use the -ls and -cat HDFS command to, respectively, list the content of the output directory, and display one of the output file.

IMPORTANT : do not forget to eliminate the Cluster when you have finished with the lab session, by right-clicking on the Cluster icon, as follows.