**Spark MapReduce**

Unzip the spark tar file to Mill or your linux or MAC computer.  
tar -xvf packageName

Hariharan has created a file on steps of how to run a sample word count problem on   
Foundry and Google Colab.

https://colab.research.google.com/drive/1r8ndKWz1wLLk65jtsktWCow9hQiacaNH#scrollTo=6SKtYA9V13We

**Commands below for Spark practice:**

First go into the directory where Spark is copied and extracted

cd ../spark/spark-3.4.2-bin-hadoop3

bin/pyspark

lines = sc.textFile("README.md")

lines.count()

lines.first()

To exit either shell, press Ctrl-D

// Filtering example

lines = sc.textFile("README.md")

pythonLines = lines.filter(lambda line: "Python" in line)

pythonLines.first()

// parallelize method in python  
nums = sc.parallelize([1, 2, 3, 4])  
squared = nums.map(lambda x: x \* x).collect()  
for num in squared:  
print "%i " % (num)

// flatMap   
lines = sc.parallelize(["hello world", "hi"])  
words = lines.flatMap(lambda line: line.split(" "))  
words.first()

// action  
nums = sc.parallelize([1, 2, 3, 4])  
sum = nums.reduce(lambda x, y: x + y)

**Pop quiz: What is the output of this line of code?**

ans = nums.reduce(lambda x, y: max(x,y))

Saving the RDD on file

rdd.saveAsTextFile("folder")

e.g., wordCounts.saveAsTextFile("/folder")

**Wordcount in Python**

rdd = sc.textFile("README.md")

words = rdd.flatMap( lambda x: x.split( " " ) )

result = words.map( lambda x: (x,1)).reduceByKey( lambda x, y: x + y )

result.saveAsTextFile("output")

**Running python script**  
bin/spark-submit my\_script.py

For example, if you have to run wordcount program file wc.py

bin/spark-submit wc.py

Then go to the output folder specified in the python program to check the output.