Hadoop Python MapReduce

Mapping Function

```
#! usr/bin/env  python
import sy
for line in sys.stdin:
  line = line.strip()
  words = line.split()
  for word in words:
    print("%s\t%s" %(word, "1"))
```

And then we control + x exit

cat test.txt| python wc_mapper.py

Reduce Function

```
#read values
nano wc_reduce.py
#! usr/bin/env  python
import sys
word2count = {}
for line in sys.stdin:
  word,count = line.strip().split('\t')
  try:
    count = int(count)
  except ValueError:
    continue
  try:
    word2count[word]=word2count[word]+count
  except:
    word2count[word]=count
for word in word2count.keys():
  print("%s\t%s" %(word, word2count[word]))

```

cat test.txt | python wc_mapper.py|python wc_reduce.py
hdfs dfs - put test.txt
#Create the batch
nano runmr.sh
Write:

```
#! /bin/bash
hadoop jar blablabla
\" continue in the next line
– input /user/cloudera/text.txt
```

```
- output /user/cloudera/wc_output
- file wc_mapper.py
- file wc_reducer.py
- mapper "python wc_mapper.py"
- reducer "python wc_reduce.py"
# if you run into error
chmod +x wc_mapper.py
```

```
sort -k 2rn
#This will deriectly go into the reduce file.
os.system('|sort -k 2rn')
#By default, 3 sections sent to the HDFS server.
#The sort funtion can only sort the chunk in that server
```