Mapper.py:

#!/usr/bin/python

import sys

#Word Count Example

# input comes from standard input STDIN

for line in sys.stdin:

line = line.strip() #remove leading and trailing whitespaces

words = line.split() #split the line into words and returns as a list

for word in words:

#write the results to standard output STDOUT

print'%s %s' % (word,1) #Emit the word

Reducer.py file:

#!/usr/bin/python

import sys

from operator import itemgetter

# using a dictionary to map words to their counts

current\_word = None

current\_count = 0

word = None

# input comes from STDIN

for line in sys.stdin:

line = line.strip()

word,count = line.split(' ',1)

try:

count = int(count)

except ValueError:

continue

print '%s %s' %(current\_word, word)

if current\_word == word:

current\_count += count

else:

if current\_word:

print '%s %s' % (current\_word, current\_count)

current\_count = count

current\_word = word

if current\_word == word:

print '%s %s' % (current\_word,current\_count)

Word.txt:

Cat mouse lion deer Tiger lion Elephant lion deer

Command text file:

mkdir pythonhaoop

cd pythonhaoop/

cp /usr/lib/hadoop-mapreduce/hadoop-streaming.jar .

vi mapper.py

vi reducer.py

vi word.txt

cat word.txt | python mapper.py

cat word.txt | python mapper.py | sort -k1,1 | python reducer.py

hadoop dfs -put word.txt .

hadoop jar hadoop-streaming.jar -file mapper.py -mapper mapper.py -file reducer.py -reducer reducer.py -input word.txt -output /home/training/workspace/pythonhaoop/Wordcount

hadoop dfs -ls /home/training/workspace/pythonhaoop/Wordcount

hadoop dfs -cat /home/training/workspace/pythonhaoop/Wordcount/part-00000