Name : Rajvardhan Reddy
Reg no : 180905093
Sec : B , Batch – B1
Roll No : 19

DS Lab - Week 5 : MapReduce Programming using Python

P1) Write a basic wordcount program.

Mapper:

```
import sys
for line in sys.stdin:
    line = line.strip()
    words = line.split()
    for word in words:
        print ('%s\t%s' % (word, 1))
```

Reducer:

```
from operator import itemgetter
import sys
current_word = None
current_count = 0
word = None
for line in sys.stdin:
    line = line.strip()
    word, count = line.split('\t', 1)
    try:
        count = int(count)
    except ValueError:
        continue
    if current_word == word:
        current_count += count
    else:
        if current_word:
            print ('%s\t%s' % (current_word, current_count))
        current_count = count
        current_word = word
if current_word == word:
    print ('%s\t%s' % (current_word, current_count))
```

Output:

Heart Disease :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat hdd_
input.txt |python3 mapper.py |sort|python3 reducer.py
1       302
2       1
29,1,1,130,204,0,0,202,0,0,2,0,2,Yes    1
34,0,1,118,210,0,1,192,0,0.7,2,0,2,Yes  1
34,1,3,118,182,0,0,174,0,0,2,0,2,Yes    1
35,0,0,138,183,0,1,182,0,1.4,2,0,2,Yes  1
35,1,0,120,198,0,1,130,1,1.6,1,0,3,No   1
35,1,0,126,282,0,0,156,1,0,2,0,3,No     1
35,1,1,122,192,0,1,174,0,0,2,0,2,Yes    1
37,0,2,120,215,0,1,170,0,0,2,0,2,Yes    1
37,1,2,130,250,0,1,187,0,3.5,0,0,2,Yes  1
38,1,2,138,175,0,1,173,0,0,2,4,2,Yes    1
38,1,3,120,231,0,1,182,1,3.8,1,0,3,No   1
39,0,2,138,220,0,1,152,0,0,1,0,2,Yes    1
39,0,2,94,199,0,1,179,0,0,2,0,2,Yes     1
39,1,0,118,219,0,1,140,0,1.2,1,0,3,No   1
39,1,2,140,321,0,0,182,0,0,2,0,2,Yes    1
40,1,0,110,167,0,0,114,1,2,1,0,3,No     1
40,1,0,152,223,0,1,181,0,0,2,0,3,No     1
40,1,3,140,199,0,1,178,1,1.4,2,0,3,Yes  1
41,0,1,105,198,0,1,168,0,0,2,1,2,Yes    1
```

Covid19 :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat covid
 input.txt |python3 mapper.py |sort|python3 reducer.py
114004,9/20/2020,Acre,Brazil,9/21/2020  1
114005,9/20/2020,Adygea 1
114006,9/20/2020,Aguascalientes,Mexico,9/21/2020        1
114007,9/20/2020,Aichi,Japan,9/21/2020  1
114008,9/20/2020,Akita,Japan,9/21/2020  1
114009,9/20/2020,Alabama,US,9/21/2020   1
114010,9/20/2020,Alagoas,Brazil,9/21/2020       1
114011,9/20/2020,Alaska,US,9/21/2020    1
114012,9/20/2020,Alberta,Canada,9/21/2020       1
114013,9/20/2020,Altai  1
11401,4/3/2020,,Mali,4/3/2020   1
114014,9/20/2020,Altai  1
114015,9/20/2020,Amapa,Brazil,9/21/2020 1
114016,9/20/2020,Amazonas,Brazil,9/21/2020      1
114017,9/20/2020,Amazonas,Colombia,9/21/2020    1
114018,9/20/2020,Amazonas,Peru,9/21/2020        1
114019,9/20/2020,Amur   1
114020,9/20/2020,Ancash,Peru,9/21/2020  1
1140,2/10/2020,Shaanxi,Mainland 1
114021,9/20/2020,Andalusia,Spain,9/21/2020      1
114022,9/20/2020,Andaman        1
114023,9/20/2020,Andhra 1
11402,4/3/2020,,Malta,4/3/2020  1
114024,9/20/2020,Anguilla,UK,9/21/2020  1
```

Example :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat examp
le.txt|python3 mapper.py|sort|python3 reducer.py
09:01   1
09:02   1
09:04   1
09:23   1
09:25   1
09:57   1
10:04   1
10:05   1
10:09   1
10:12   1
10:39   1
10:57   1
11:02   1
11:18   1
11.31   1
11:35   1
11:39   1
11:46   1
11:52   1
11:53   1
11:54   1
12:05   1
```

German Credit :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat g
c_data.csv |python3.8 mapper.py |sort| python3 reducer.py
0,10127,48      1
0,1024,24       1
0,10297,48      1
0,1042,18       1
0,1056,18       1
0,1082,12       1
0,10961,48      1
0,10974,36      1
0,1108,12       1
0,1123,12       1
0,1131,18       1
0,11328,24      1
0,1136,9        1
0,11560,24      1
0,11590,48      1
0,11816,45      1
0,1188,21       1
0,1190,18       1
0,1193,24       1
0,11938,24      1
0,1198,6        1
```

P2) MapReduce program to find frequent words

Mapper 2:

```
from __future__ import print_function
import sys
for line in sys.stdin:
    word, count = line.strip().split('\t', 1)
    count = int(count)
    print( '%d\t%s' % (count, word) )
```

Reducer 2:

```
from __future__ import print_function
import sys
mostFreq = []
currentMax = -1
for line in sys.stdin:
    count, word = line.strip().split('\t', 1)
    count = int(count)
    if count > currentMax:
        currentMax = count
        mostFreq = [ word ]
    elif count == currentMax:
        mostFreq.append( word )
for word in mostFreq:
print( '%s\t%s' % ( word, currentMax ) )
```

Output:

Heart Disease :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat hdd_i
nput.txt |python3 freqmap1.py |sort|python3 freqred1.py |python3 freqmap2.py |sort|pytho
n3 freqred2.py
1        302
```

Covid19 :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat covid
_input.txt |python3 freqmap1.py |sort|python3 freqred1.py |python3 freqmap2.py |sort|pyt
hon3 freqred2.py
1        222508
```

Example :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat examp
le.txt |python3 freqmap1.py |sort|python3 freqred1.py |python3 freqmap2.py |sort|python3
 freqred2.py
amex     13
```

German Credit :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat g
c_data.csv |python3.8 freqmap1.py |sort|python3.8 freqred1.py|python3.8 freqmap2.py
|sort|python3.8 freqred2.py
1,1258,24        2
1,1262,12        2
1,1374,6         2
1,1424,12        2
1,1478,15        2
1,2171,12        2
1,701,12         2
```

P3) MapReduce program to explore the dataset and perform the filtering (typically creating key/value pairs) by mapper and perform the count and summary operation on the instances.

Mapper:

```
#import string
import fileinput
for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) == 6:
        date, time, location, item, cost, payment = data
        print ("{0}\t{1}".format(location, cost))
#for heart disease file
        age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, that, target =
data
        print('{0}\t{1}'.format(age, chol))
#for covid file
        SNo, ObservationDate, Province_State, Country_Region, Last_Update, Confirmed,
Deaths, Recovered = data
        print('{0}\t{1}'.format(Country_Region, Deaths))
#for credit file
        Creditability, CreditAmount, DurationOfCreditInMonths = data
        print('{0}\t{1}'.format(Creditability, DurationOfCreditInMonths))
```

Reducer:

```
 import fileinput
transactions_count = 0
sales_total = 0
for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) != 2:
        continue
    current_key, current_value = data
    transactions_count += 1
    sales_total += float(current_value)
print (transactions_count, "\t", sales_total)
```

Output:

German Credit :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat g
c_data.txt | python3.8 q3itemmap.py |sort|python3.8 q3itemlsred.py
2799	9
841	12
2122	12
2171	12
2241	10
3398	8
1361	6
1098	18
3758	24
3905	11
6187	30
1957	6
7582	48
1936	18
2647	6
3939	11
3213	18
2337	36
7228	11
3676	6
3124	12
2384	36
1424	12
4716	6
4771	11
652	12
1154	9
3556	15
4796	42
3017	30
3535	36
6614	36
1376	24
1721	15
```

P4) Write a mapper and reducer program for word count by defining separator instead of using "\t".

Mapper:

```
import sys
def read_input(file):
    for line in file:
        yield line.split()
def main(separator='\t'):
    data = read_input(sys.stdin)
    for words in data:
        for word in words:
            print ('%s%s%d' % (word, separator, 1))
if __name__ == "__main__":
    main()
```

Reducer:

```
from itertools import groupby
from operator import itemgetter
import sys
def read_mapper_output(file, separator='\t'):
    for line in file:
        yield line.rstrip().split(separator, 1)
def main(separator='\t'):
    data = read_mapper_output(sys.stdin, separator=separator)
    for current_word, group in groupby(data, itemgetter(0)):
        try:
            total_count = sum(int(count) for current_word, count in group)
```

```
            print ("%s%s%d" % (current_word, separator, total_count))
        except ValueError:
            pass
if __name__ == "__main__":
    main()
```

Output:

Heart Disease :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat h
dd_input.txt |python3.8 q4sepmap.py |sort|python3.8 q4sepred.py
29      1
34      2
35      4
37      2
38      3
39      4
40      3
41      10
42      8
43      8
44      11
45      8
46      7
47      5
48      7
49      5
50      7
51      12
52      13
53      8
54      16
55      8
56      11
57      17
58      19
59      14
60      11
61      8
62      11
63      8
64      10
65      8
66      7
67      9
68      4
```

P5) Write a map reduce program that returns the cost of the item that is most expensive, for each location in the dataset example.txt

Mapper:

```
import fileinput
for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) == 6:
        date, time, location, item, cost, payment = data
        print ("{0}\t{1}".format(location, cost))
#for heart disease file
        age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, that, target =
data
        print('{0}\t{1}'.format(age, chol))
#for covid file
        SNo, ObservationDate, Province_State, Country_Region, Last_Update, Confirmed,
Deaths, Recovered = data
        print('{0}\t{1}'.format(Country_Region, Deaths))
#for credit file
        Creditability, CreditAmount, DurationOfCreditInMonths = data
        print('{0}\t{1}'.format(Creditability, DurationOfCreditInMonths))
```

Reducer:

```
import fileinput
max_value = 0
old_key = None
for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) != 2:
        continue
    current_key, current_value = data
    if old_key and old_key != current_key:
        print (old_key, "\t", max_value)
        old_key = current_key
        max_value = 0
    old_key = current_key
    if float(current_value) > float(max_value):
        max_value = float(current_value)
if old_key != None:
    print (old_key, "\t", max_value)
```

Output:

<u>Covid 19 :</u>

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ cat c
ovid_input.txt |python3.8 q5map.py |sort|python3.8 q5red.py
Afghanistan       39145.0
Albania           12787.0
Algeria           50400.0
Andorra           1753.0
Angola    4363.0
Antigua and Barbuda       97.0
Argentina         664799.0
Armenia           47877.0
Aruba     4.0
Australia         20105.0
Austria           39984.0
Azerbaijan        39524.0
Bahamas           3618.0
Bahamas, The      4.0
Bahrain           67014.0
Bangladesh        353844.0
Barbados          189.0
Belarus           76357.0
Belgium           106887.0
```

P6) Write a mapreduce program to evaluate the PI.

Mapper:

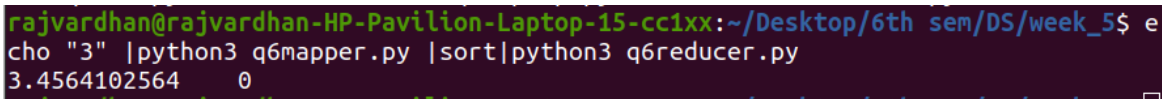
```
import sys
def f( x ):
    return 4.0 / ( 1.0 + x*x )
for line in sys.stdin:
    line = line.strip()
    words = line.split()
    N = int( words[0] )
    deltaX = 1.0 / N
    for i in range( 0, N ):
        print( "1\t%1.10f" % ( f( i * deltaX )*deltaX ) )
```

Reducer:

```
from __future__ import print_function
from operator import itemgetter
import sys
sum = 0
```

```
for line in sys.stdin:
    line = line.strip()
    word, count = line.split('\t', 1)
    try:
        count = float(count)
    except ValueError:
        # count was not a number, so silently
        # ignore/discard this line
        #print( "--skipping (%s, %s)" % ( str(word), str(count) ) )
        continue
    sum += count
print( '%1.10f\t0' % sum )
```

Output:



P7) Write a MapReduce program to generate a report with Number of males, females and total births in each year, number of males, females and total births in each month of a particular year from national birth data.

```
from operator import itemgetter
import sys
current_year = None
year_tot = [0]
year_male = [0]
year_fem = [0]
month_tot = [0]
month_male = [0]
month_fem = [0]
specyear = 2001
i = 1
while i<40:
        year_tot.append(0)
        year_male.append(0)
        year_fem.append(0)
        i += 1
i = 0
while i<12:
        month_tot.append(0)
        month_male.append(0)
        month_fem.append(0)
        i += 1
year = None
for line in sys.stdin:
   line = line.strip()
   sex, month, year = line.split(' ')
   sex = int(sex)
   month = int(month)
   year = int(year)
```

```
    year_tot[year-1980] += 1
    if sex == 0:
        year_male[year-1980] += 1
    else:
        year_fem[year-1980] += 1
    if year == specyear:
        month_tot[month-1] += 1
        if sex == 0:
            month_male[month-1] += 1
        else:
            month_fem[month-1] += 1
i = 0
while i<40:
        if year_tot[i] == 0:
            i += 1
            continue
        print('Year %d Total: %d' %(i+1980, year_tot[i]))
        print('Males: %d' %(year_male[i]))
        print('Females: %d' %(year_fem[i]))
        print('\n')
        i += 1
print('Year %d'% (specyear))
i = 0
while i<12:
        if month_tot[i] == 0:
            i += 1
            continue
        print('Month %d Total: %d' %(i+1, month_tot[i]))
        print('Males: %d' %(month_male[i]))
        print('Females: %d' %(month_fem[i]))
        print('\n')
        i += 1
```

Output :

```
Year 2007 Total: 5
Males: 1
Females: 4


Year 2010 Total: 10
Males: 5
Females: 5


Year 2013 Total: 6
Males: 2
Females: 4


Year 2016 Total: 8
Males: 5
Females: 3


Year 2019 Total: 6
Males: 2
Females: 4


Year 2001
Month 2 Total: 1
Males: 0
Females: 1


Month 5 Total: 1
Males: 0
Females: 1


Month 8 Total: 1
Males: 0
Females: 1


Month 10 Total: 1
Males: 0
Females: 1


Month 12 Total: 1
Males: 1
Females: 0
```

P8) Write a MapReduce program to count even or odd numbers in randomly generated natural numbers.

```
from operator import itemgetter
import sys
```

```
odd_count = 0
even_count = 0
for line in sys.stdin:
    line = line.strip()
    num = line.split()
    for currnum in num:
        try:
            odd_count = int(odd_count)
            even_count = int(even_count)
            currnum = int(currnum)
        except ValueError:
            continue
        if currnum%2 == 0:
            even_count += 1
        else:
            odd_count += 1
print ('%s odd and %s even' % (odd_count, even_count))
```

Output:

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/6th sem/DS/week_5$ c
at rand_nums.txt|python3 q8mapreduce.py
8 odd and 8 even
```