**BIG DATA HOME ASSIGNMENT - 1**

**Dataset** : Twitter Data

**Topic** : Pre-processing of twitter data

**Mapper.py**

#!/usr/bin/env python

from datetime import datetime

import json

import sys

# Mapper - filters tweets by user screen\_name and generates key-value pairs of the format <hour\_of\_creation, 1>

filter\_user\_name = 'prezono'

for line in sys.stdin:

tweet\_data = json.loads(line)

if tweet\_data.get('user').get('screen\_name').strip().lower() == filter\_user\_name:

timestamp = tweet\_data.get('created\_at')

date\_object = datetime.strptime(timestamp, '%a %b %d %H:%M:%S +0000 %Y')

print '%s\t%s' % (date\_object.hour, 1)

**Reducer.py**

#!/usr/bin/env python

from \_\_future\_\_ import division

import sys

import string

# Reducer - processes the output of the mapper to compute the hourly average of tweets

tweet\_count = 0

hour = None

tweets\_by\_hour\_avg = {}

num\_days = 365

for line in sys.stdin:

(key, val) = line.strip().split('\t', 1)

if hour != key:

if hour:

print "Average # of tweets @ %s hrs:\t%s" % (hour, tweet\_count/num\_days)

tweets\_by\_hour\_avg[hour] = tweet\_count/num\_days

tweet\_count = 0

hour = key

try:

tweet\_count += int(val)

except:

continue

print 'Tweets @ %s hrs:\t%s' % (hour, tweet\_count)

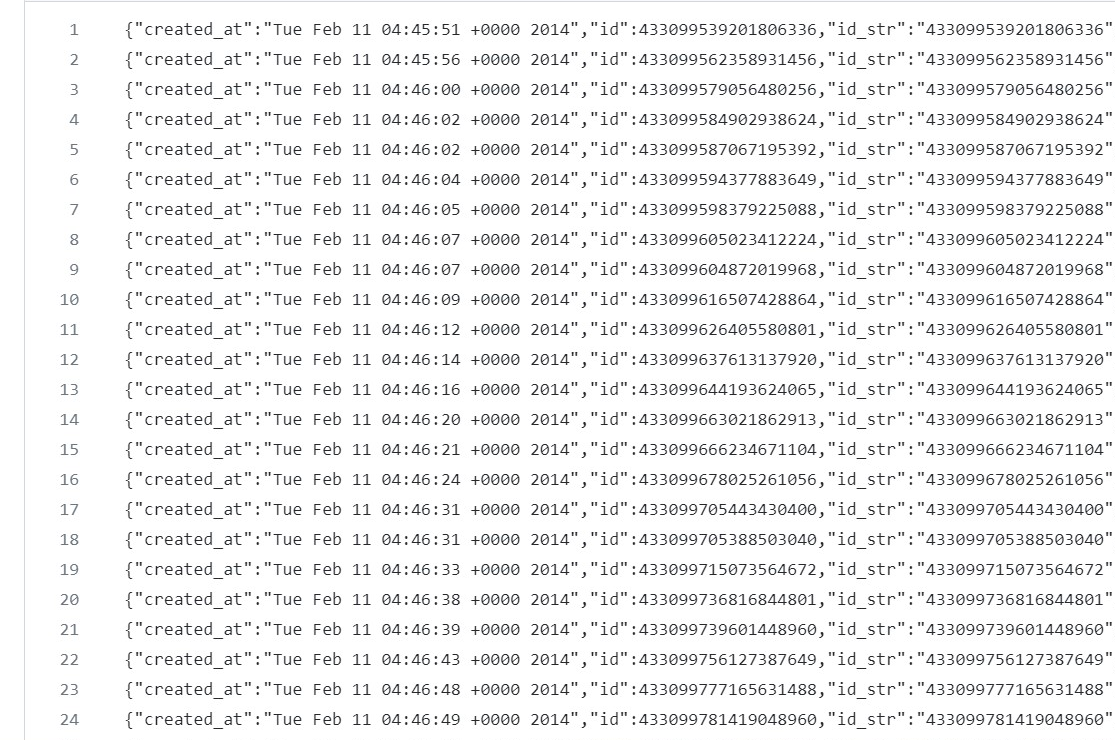
tweets\_by\_hour\_avg[hour] = tweet\_count/num\_days

print "Hour of the day that @PrezOno tweets the most:", max(tweets\_by\_hour\_avg, key=tweets\_by\_hour\_avg.get)

**Output :**

**sample**

{created\_at:Tue Feb 11 04:45:51 +0000 2014,id\_str:433099539201806336}



**BIG DATA HOME ASSIGNMENT - 2**

**Dataset** : Twitter Data

**Topic** : Analysing Pre-processed twitter data

**Mapper.py**

#!/usr/bin/env python

from datetime import datetime

import json

import sys

# Mapper - filters tweets by user screen\_name and generates key-value pairs of the format <hour\_of\_creation, 1>

filter\_user\_name = 'prezono'

for line in sys.stdin:

tweet\_data = json.loads(line)

if tweet\_data.get('user').get('screen\_name').strip().lower() == filter\_user\_name:

timestamp = tweet\_data.get('created\_at')

date\_object = datetime.strptime(timestamp, '%a %b %d %H:%M:%S +0000 %Y')

print '%s\t%s' % (date\_object.hour, 1)

**Reducer.py**

#!/usr/bin/env python

from \_\_future\_\_ import division

import sys

import string

# Reducer - processes the output of the mapper to compute the hourly average of tweets

tweet\_count = 0

hour = None

tweets\_by\_hour\_avg = {}

num\_days = 365

for line in sys.stdin:

(key, val) = line.strip().split('\t', 1)

if hour != key:

if hour:

print "Average # of tweets @ %s hrs:\t%s" % (hour, tweet\_count/num\_days)

tweets\_by\_hour\_avg[hour] = tweet\_count/num\_days

tweet\_count = 0

hour = key

try:

tweet\_count += int(val)

except:

continue

print 'Tweets @ %s hrs:\t%s' % (hour, tweet\_count)

tweets\_by\_hour\_avg[hour] = tweet\_count/num\_days

print "Hour of the day that @PrezOno tweets the most:", max(tweets\_by\_hour\_avg, key=tweets\_by\_hour\_avg.get)

**Output:**

**Sample:**

Average # of tweets @ 0 hrs: 0.0383561643836

Average # of tweets @ 1 hrs: 0.0520547945205

Average # of tweets @ 10 hrs: 0.0438356164384

Average # of tweets @ 11 hrs: 0.0657534246575

Average # of tweets @ 12 hrs: 0.0328767123288

Average # of tweets @ 13 hrs: 0.041095890411

