DS_lab1_v1.0
Innopolis University
Student: Ilin Oleg

Hadoop

Map

```
#!/usr/bin/python

import sys
import os
import re

filterStart = ('Media:', 'Special:', 'Talk:',
               'User:', 'User_talk:',
               'Project:', 'Project_talk:',
               'File:', 'File_talk:',
               'MediaWiki:', 'MediaWiki_talk:',
               'Template:', 'Template_talk:',
               'Help:', 'Help_talk:',
               'Category:', 'Category_talk:',
               'Portal:',
               'Wikipedia:', 'Wikipedia_talk:')

filterEnd = ('.jpg', '.gif', '.png', '.JPG', '.GIF', '.PNG', '.txt', '.ico')

filterT = ('404_error/', 'Main_Page', 'Hypertext_Transfer_Protocol', 'Search')


def cond1 (word):
    for w in filterStart:
        if word.startswith(w):
            return False
    return True


def cond2 (word):
    for w in filterEnd:
        if word.endswith(w):
            return False
    return True


def cond3 (word):
    for w in filterT:
        if word == w:
            return False
    return True


d = os.environ["mapreduce_map_input_file"]
#d = "pagecounts-20160701-000000"
reg = re.compile("pagecounts-(\d{8})")
date = reg.findall(d)[-1]
for line in sys.stdin:
    line = line.strip()
    words = line.split()
    if line.startswith('en') and cond1(words[1]) and words[1][0].isupper():
        if cond2(words[1]):
            if cond3(words[1]):
                if len(words) == 4:
                    print '%s\t%s\t%s' %  (words[1], words[2], date)
```

Reduce

```
#!/usr/bin/python
import sys

current_word = None
current_count = 0
word = None
dict = {'20160701':0, '20160702':0, '20160703':0, '20160704':0, '20160705':0,
'20160706':0, '20160707':0}

for line in sys.stdin:
    line = line.strip()
    words = line.split()
    word = words[0]
    count = words[1]
    data = words[2]

    try:
        count = int(count)
    except ValueError:
        continue

    if current_word == word:
        current_count += count
        dict[data] += count

    else:
        if current_word and current_count>100000:
            s = str(current_count) + '\t' + str(current_word) + '\t'
            for i in dict:
                s += str(i) + ':' + str(dict[i]) + '\t'
            print(s)
        current_count = count
        current_word = word
        dict = {'20160701':0, '20160702':0, '20160703':0, '20160704':0, '20160705':0,
'20160706':0, '20160707':0}
        dict[data] = count

if current_word == word and current_count>100000:
    s = str(current_count) + '\t' + str(current_word) + '\t'
    dict[data] += count
    for i in dict:
        s += str(i) + ':' + str(dict[i]) + '\t'
    print(s)
```

Run

```
$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming-2.6.0-cdh5.9.0.jar -D
mapred.output.compress=true -D
mapred.output.compression.codec=org.apache.hadoop.io.compress.GzipCodec -input
/inputdata/* -output /output/student15/output -file map.py -file reduce.py -mapper
map.py -reducer reduce.py
```

```
File System Counters
        FILE: Number of bytes read=1652959330
        FILE: Number of bytes written=6313202569
        FILE: Number of read operations=0
        FILE: Number of large read operations=0
        FILE: Number of write operations=0
        HDFS: Number of bytes read=14981540639
        HDFS: Number of bytes written=5519
        HDFS: Number of read operations=564
        HDFS: Number of large read operations=0
        HDFS: Number of write operations=40
Job Counters
        Launched map tasks=168
        Launched reduce tasks=20
        Data-local map tasks=80
        Rack-local map tasks=88
        Total time spent by all maps in occupied slots (ms)=9553514
        Total time spent by all reduces in occupied slots (ms)=1594473
        Total time spent by all map tasks (ms)=9553514
        Total time spent by all reduce tasks (ms)=1594473
        Total vcore-seconds taken by all map tasks=9553514
        Total vcore-seconds taken by all reduce tasks=1594473
        Total megabyte-seconds taken by all map tasks=9782798336
        Total megabyte-seconds taken by all reduce tasks=1632740352
Map-Reduce Framework
        Map input records=1147454566
        Map output records=266819086
        Map output bytes=8496060880
        Map output materialized bytes=4636546319
        Input split bytes=18984
        Combine input records=0
        Combine output records=0
        Reduce input groups=23465906
        Reduce shuffle bytes=4636546319
        Reduce input records=266819086
        Reduce output records=91
        Spilled Records=533638172
        Shuffled Maps =3360
        Failed Shuffles=0
        Merged Map outputs=3360
        GC time elapsed (ms)=86126
        CPU time spent (ms)=9032690
        Physical memory (bytes) snapshot=119447048192
        Virtual memory (bytes) snapshot=304995196928
        Total committed heap usage (bytes)=147837681664
Shuffle Errors
        BAD_ID=0
        CONNECTION=0
        IO_ERROR=0
        WRONG_LENGTH=0
        WRONG_MAP=0
        WRONG_REDUCE=0
File Input Format Counters
        Bytes Read=14981521655
File Output Format Counters
        Bytes Written=5519
```

Result

```
159699  Batman_v_Superman:_Dawn_of_Justice      20160701:23225  20160702:23598
20160703:24292  20160704:22324  20160705:25017  20160706:21728  20160707:19515

105461  Gretchen_Carlson        20160701:145    20160702:120    20160703:129
20160704:138    20160705:185    20160706:61247  20160707:43497

114913  Jupiter 20160701:6070   20160702:4277   20160703:5323   20160704:8966
20160705:58504  20160706:20321  20160707:11452

128321  Michael_Cimino  20160701:284    20160702:8665   20160703:59426  20160704:23196
20160705:19511  20160706:10569  20160707:6670

205693  UFC_200 20160701:16673  20160702:14720  20160703:16150  20160704:22352
20160705:30483  20160706:33755  20160707:71560

127691  United_Kingdom  20160701:22453  20160702:15352  20160703:14067  20160704:19952
20160705:19409  20160706:20907  20160707:15551

428921  Earth   20160701:60133  20160702:60613  20160703:65228  20160704:51767
20160705:62345  20160706:61628  20160707:67207

154152  James_Comey     20160701:3302   20160702:3040   20160703:1843   20160704:963
20160705:46219  20160706:40314  20160707:58471

100526  Juan_Cuadrado   20160701:95656  20160702:610    20160703:835    20160704:854
20160705:884    20160706:960    20160707:727

208727  Kevin_Durant    20160701:5666   20160702:5403   20160703:8013   20160704:72330
20160705:71537  20160706:26032  20160707:19746

176049  List_of_Bollywood_films_of_2016 20160701:29200  20160702:30248  20160703:22914
20160704:22633  20160705:23265  20160706:24607  20160707:23182

553073  List_of_feed_aggregators        20160701:96870  20160702:86707  20160703:89509
20160704:96633  20160705:77539  20160706:63542  20160707:42273

106783  Manchester_United_F.C.  20160701:16880  20160702:19057  20160703:13209
20160704:13215  20160705:14397  20160706:16072  20160707:13953
```