Problem Set 5 : MapReduce on Subway Data

Miadad Rashid 2015.2.24

```python
1.  Problem Set 5.1
2.
3.  # mapper.py
4.
5.  import sys
6.  import string
7.  import logging
8.
9.  from util import mapper_logfile
10. logging.basicConfig(filename=mapper_logfile, format='%(message)s',
11.                     level=logging.INFO, filemode='w')
12.
13. def mapper():
14.
15.     for line in sys.stdin:
16.         # your code here
17.         data = line.strip().split(',')
18.         if len(data) != 22 or data[1] == 'UNIT':
19.             continue
20.
21.         out = "{}\t{}".format(data[1], data[6])
22.         print out
23.
24.
25. mapper()
26.
27. import sys
28. import logging
29.
30. from util import reducer_logfile
31. logging.basicConfig(filename=reducer_logfile, format='%(message)s',
32.                     level=logging.INFO, filemode='w')
33.
34. def reducer():
35.
36.     entries = 0
37.     last_unit = None
38.
39.     for line in sys.stdin:
40.         # your code here
41.         data = line.strip().split('\t')
42.         if len(data) != 2:
43.             continue
44.         this_unit, count = data
45.
46.         if last_unit and last_unit != this_unit:
47.             print '{0}\t{1}'.format(last_unit, entries)
48.             entries = 0
49.
50.         last_unit = this_unit
51.         entries += float(count)
52.
53.     if last_unit != None:
54.         print "{}\t{}".format(last_unit, entries)
55.
56.
57. reducer()
58.
59.
60.
61.
```

```
62. Problem Set 5.2
63.
64. import sys
65. import string
66. import logging
67.
68. from util import mapper_logfile
69. logging.basicConfig(filename=mapper_logfile, format='%(message)s',
70.                     level=logging.INFO, filemode='w')
71.
72. def mapper():
73.     def format_key(fog, rain):
74.         return '{}fog-{}rain'.format(
75.             '' if fog else 'no',
76.             '' if rain else 'no'
77.         )
78.
79.     for line in sys.stdin:
80.         # your code here
81.         data = line.strip().split(',')
82.         if len(data) != 22 or data[1] == 'UNIT':
83.             continue
84.         out = "{}\t{}".format(format_key(float(data[14]), float(data[15])), data[6])
85.         print out
86.
87.
88.
89. mapper()
90. import sys
91. import logging
92.
93. from util import reducer_logfile
94. logging.basicConfig(filename=reducer_logfile, format='%(message)s',
95.                     level=logging.INFO, filemode='w')
96.
97. def reducer():
98.
99.     riders = 0        # The number of total riders for this key
100.        num_hours = 0   # The number of hours with this key
101.        old_key = None
102.
103.        for line in sys.stdin:
104.            # your code here
105.            data = line.strip().split('\t')
106.            if len(data) != 2:
107.                continue
108.            this_unit, count = data
109.
110.            if old_key and old_key != this_unit:
111.                print '{0}\t{1}'.format(old_key, (riders/float(num_hours)))
112.                riders = 0
113.                num_hours = 0
114.
115.            old_key = this_unit
116.            riders += float(count)
117.            num_hours += 1
118.
119.        if old_key != None:
120.            print '{0}\t{1}'.format(old_key, (riders/float(num_hours)))
121.        reducer()
122.        Problem Set 5.3
```

```
123.
124.        import sys
125.        import string
126.        import logging
127.
128.        from util import mapper_logfile
129.        logging.basicConfig(filename=mapper_logfile, format='%(message)s',
130.                            level=logging.INFO, filemode='w')
131.
132.        def mapper():
133.            for line in sys.stdin:
134.                data = line.strip().split(',')
135.                if data[1] == 'UNIT' or len(data) != 22:
136.                    continue
137.
138.                print '{0}\t{1}\t{2}\t{3}'.format(data[1],data[6], data[2], data[3])
139.
140.
141.        mapper()
142.
143.        import sys
144.        import logging
145.
146.        from util import reducer_logfile
147.        logging.basicConfig(filename=reducer_logfile, format='%(message)s',
148.                            level=logging.INFO, filemode='w')
149.        def reducer():
150.            max_entries = 0
151.            old_key = None
152.            datetime = ''
153.
154.            for line in sys.stdin:
155.                # your code here
156.                data = line.strip().split('\t')
157.
158.                if len(data) != 4:
159.                    continue
160.
161.                this_key, count, date, time  = data
162.                count = float(count)
163.
164.                if old_key and old_key != this_key:
165.                    print "{0}\t{1}\t{2}".format(old_key, datetime, max_entries)
166.                    max_entries = 0
167.                    datetime = ''
168.
169.                old_key = this_key
170.                if count >= max_entries:
171.                    max_entries = count
172.                    datetime = str(date) + ' ' + str(time)
173.                    #datetime = datetime.strptime(date, '%Y-%m-
     %d') + ' ' + datetime.strptime(time, '%H:%M:%S')
174.
175.            if old_key != None:
176.                print "{0}\t{1}\t{2}".format(old_key, datetime, max_entries)
177.
178.        reducer()
```