The code can be run by calling
hadoop jar share/hadoop/tools/lib/hadoop-streaming-2.5.1.jar
-file mapper.py -mapper 'mapper.py James'
-file reducer.py -reducer reducer.py
-input inputfolder -output outputfolder

Mapper:

```
import sys

myarg = sys.argv[1]

for line in sys.stdin:
        line = line.strip()
        first, last = line.split()
        if first == myarg:
                print '%s \t %s \t 1' % (first, last)
```

Reducer:

```
#!/usr/bin/env python

from operator import itemgetter
import sys

current_last = None
current_count = 0
last = None

# input comes from STDIN
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()

    # parse the input we got from mapper.py
    first, last, count = line.split('\t')

    # convert count (currently a string) to int
    try:
        count = int(count)
    except ValueError:
        # count was not a number, so silently
        # ignore/discard this line
        print 'wrong count number!!!!!!!!!!!!!!!!!!!!!!'
        break

    # this IF-switch only works because Hadoop sorts map output
    # by key (here: word) before it is passed to the reducer
    if current_last == last:
        current_count += count
```

```
    else:
        if current_last:
            # write result to STDOUT
            print '(%s,%s)\t%s' % (first, current_last, current_count)
        current_count = count
        current_last = last

# do not forget to output the last word if needed!
if current_last == last:
    print '(%s,%s)\t%s' % (first, current_last, current_count)
```