Names: Bob Skowron, Jason Walker
Keys: rskowron, jwalker
SVN: jwalker: https://svn.seas.wustl.edu/repositories/jwalker/cse427s_fl17/

no line in header area Right header

1. a.

   b. Total lines: 1,079,891; Number of lines with HTML requests: 474,360
   Code:
   Total Lines: mydata.map(lambda line: line.split(' ')).keyBy(lambda fields: (fields[0] + "/" + fields[2])).count()
   Lines with HTML: mydata.map(lambda line: line.split(' ')).filter(lambda fields: "html" in fields[6]).keyBy(lambda fields: (fields[0] + "/" + fields[2])).count()

Names: Bob Skowron, Jason Walker
Keys: rskowron, jwalker
SVN: jwalker: https://svn.seas.wustl.edu/repositories/jwalker/cse427s_fl17/

2. a. The keys are the paths to the files. The value is the entire contents of the file.
mydata.keys().take(2)
[u'hdfs://localhost:8020/loudacre/activations/2008-10.xml',
u'hdfs://localhost:8020/loudacre/activations/2008-11.xml']

b. flatMap()
import xml.etree.ElementTree as ElementTree

```
def getactivations(s):
filetree = ElementTree.fromstring(s)
return filetree.getiterator('activation')

xmldata = mydata.flatMap(lambda fields: getactivations(fields[1]))
```

c.
```
def getmodel(activation):
return activation.find('model').text
def getaccount(activation):
return activation.find('account-number').text

xmldata.map(lambda activation: getaccount(activation) + ":" + getmodel(activation)).saveAsTextFile("/loudacre/account
models")
```

Names: Bob Skowron, Jason Walker
Keys: rskowron, jwalker
SVN: jwalker: https://svn.seas.wustl.edu/repositories/jwalker/cse427s_fl17/

3. a. Pipelining = Feature of Spark that when possible, Spark will perform sequences of transformations by row so no data is stored. There are several benefits. First, no intermediate records or RDDs have to be stored. Second, it will only process the requisite transformations for the data required. We do not have to apply all transformations to all rows if they end up filtered below e.g.

   b. You can pipeline a map and a filter or pipeline and take or count???