Exercise 0: Explain your system:

Hardware and Software	**Specification**
Processor	Intel®core™ i3-5005U CPU @2.00GHz
Number of cores	2
Logical processors	4
RAM	4.00 GB
OS	Windows 10
Python	3.6

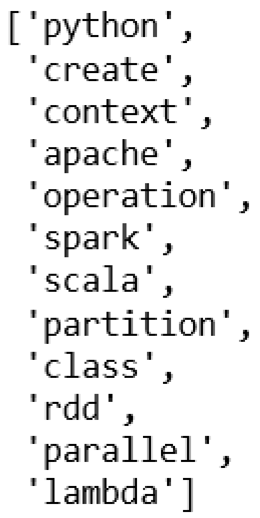
Distributed Computing with Apache Spark:

Apache Spark Basics:

Part a) Basic Operations on Resilient Distributed Dataset (RDD):

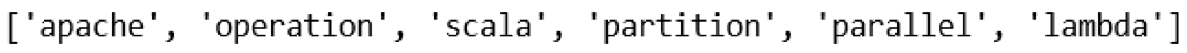
1. Perform rightOuterJoin and fullOuterJoin operations between a and b briefly explain your solution.

   FullOuterJoin:



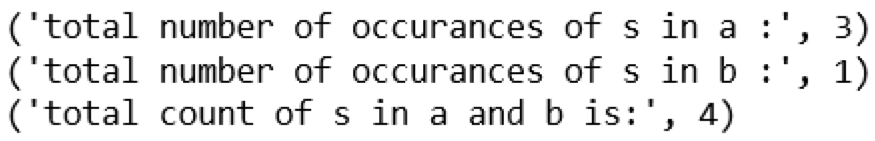
```
['python',
 'create',
 'context',
 'apache',
 'operation',
 'spark',
 'scala',
 'partition',
 'class',
 'rdd',
 'parallel',
 'lambda']
```

   RightOuterJoin:



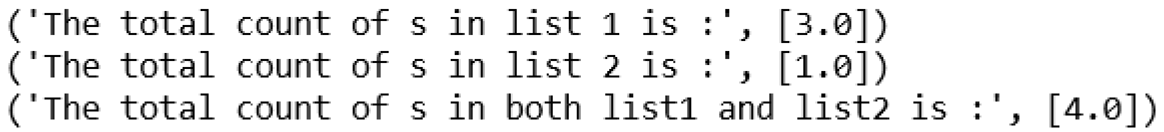
```
['apache', 'operation', 'scala', 'partition', 'parallel', 'lambda']
```

2. Using map and reduce functions to count how many times the character "s" appears in all a and b.



```
('total number of occurances of s in a :', 3)
('total number of occurances of s in b :', 1)
('total count of s in a and b is:', 4)
```

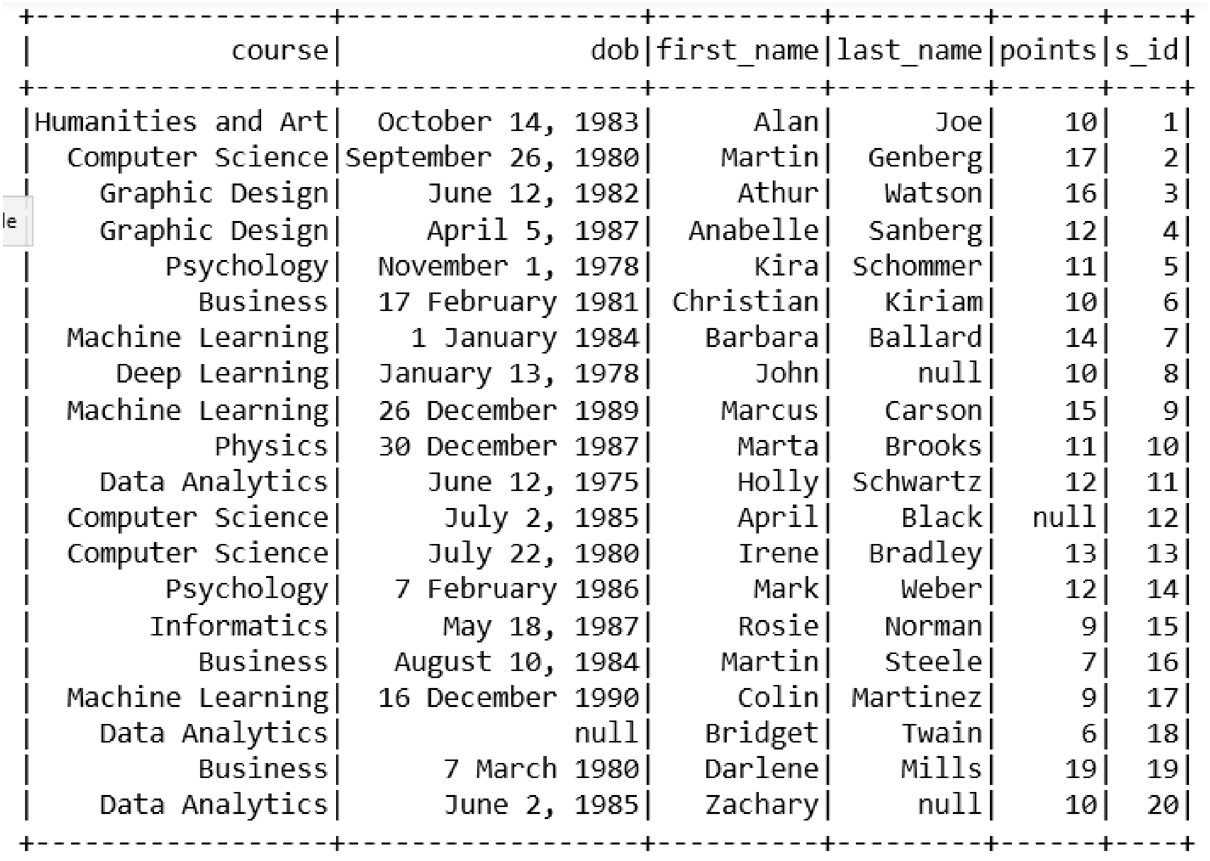
3. Using aggregate function to count how many times the character "s" appears in all a and b.



```
('The total count of s in list 1 is :', [3.0])
('The total count of s in list 2 is :', [1.0])
('The total count of s in both list1 and list2 is :', [4.0])
```

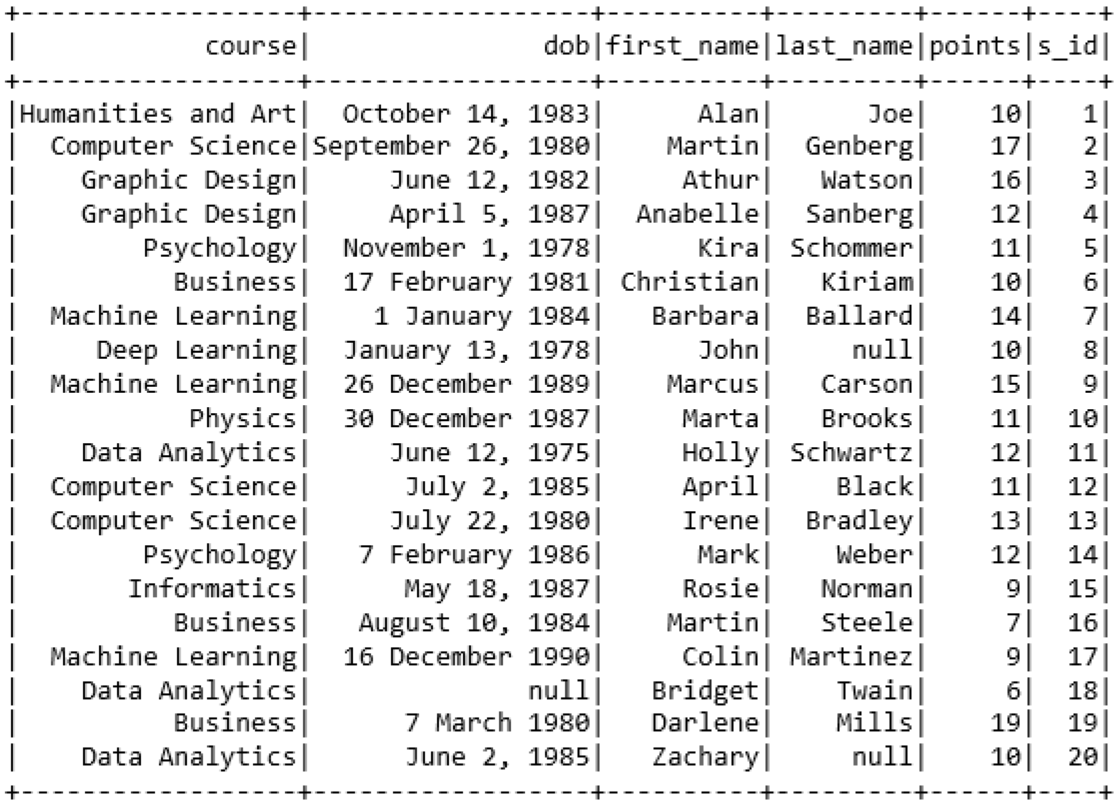
Part b) Basic Operations on DataFrames

Initial student Dataframe



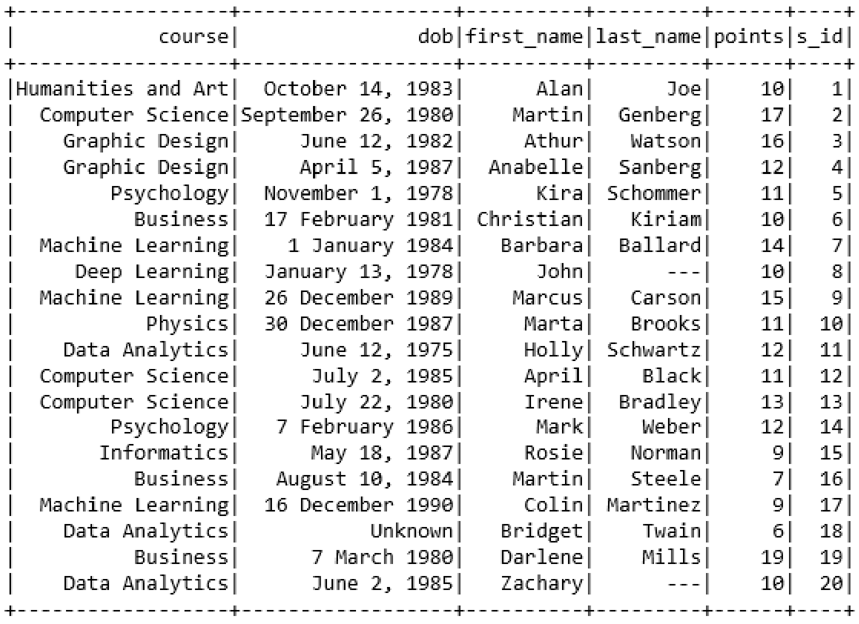
```
+-----------------+-----------------+----------+---------+------+----+
|           course|              dob|first_name|last_name|points|s_id|
+-----------------+-----------------+----------+---------+------+----+
|Humanities and Art|  October 14, 1983|      Alan|      Joe|    10|   1|
|  Computer Science|September 26, 1980|    Martin|  Genberg|    17|   2|
|    Graphic Design|    June 12, 1982|     Athur|   Watson|    16|   3|
|    Graphic Design|    April 5, 1987|  Anabelle|  Sanberg|    12|   4|
|        Psychology|  November 1, 1978|      Kira| Schommer|    11|   5|
|          Business| 17 February 1981| Christian|   Kiriam|    10|   6|
|  Machine Learning|   1 January 1984|   Barbara|  Ballard|    14|   7|
|     Deep Learning|  January 13, 1978|      John|     null|    10|   8|
|  Machine Learning| 26 December 1989|    Marcus|   Carson|    15|   9|
|           Physics| 30 December 1987|     Marta|   Brooks|    11|  10|
|    Data Analytics|    June 12, 1975|     Holly| Schwartz|    12|  11|
|  Computer Science|     July 2, 1985|     April|    Black|  null|  12|
|  Computer Science|    July 22, 1980|     Irene|  Bradley|    13|  13|
|        Psychology|  7 February 1986|      Mark|    Weber|    12|  14|
|       Informatics|     May 18, 1987|     Rosie|   Norman|     9|  15|
|          Business|  August 10, 1984|    Martin|   Steele|     7|  16|
|  Machine Learning| 16 December 1990|     Colin| Martinez|     9|  17|
|    Data Analytics|             null|   Bridget|    Twain|     6|  18|
|          Business|     7 March 1980|   Darlene|    Mills|    19|  19|
|    Data Analytics|     June 2, 1985|   Zachary|     null|    10|  20|
+-----------------+-----------------+----------+---------+------+----+
```

1. Replace the null value(s) in column points by the mean of all points.



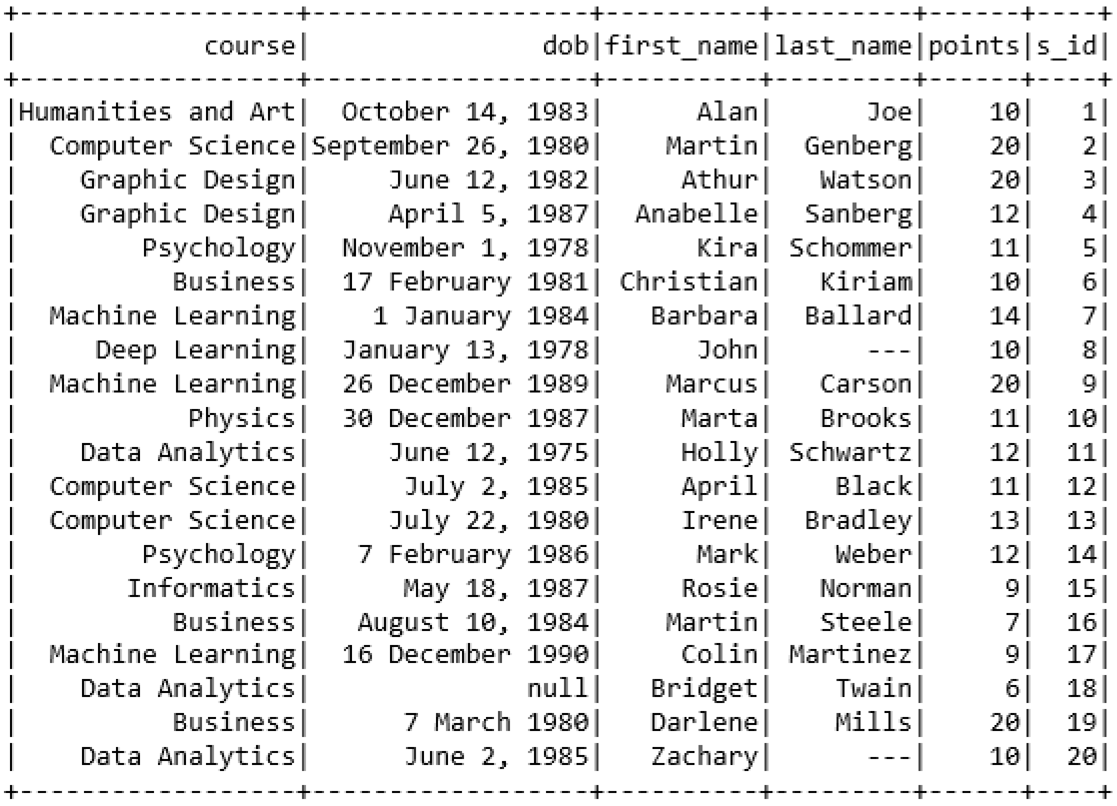
```
+------------------+------------------+----------+---------+------+----+
|            course|               dob|first_name|last_name|points|s_id|
+------------------+------------------+----------+---------+------+----+
|Humanities and Art|  October 14, 1983|      Alan|      Joe|    10|   1|
|  Computer Science|September 26, 1980|    Martin|  Genberg|    17|   2|
|    Graphic Design|     June 12, 1982|     Athur|   Watson|    16|   3|
|    Graphic Design|     April 5, 1987|  Anabelle|  Sanberg|    12|   4|
|        Psychology|  November 1, 1978|      Kira| Schommer|    11|   5|
|          Business|  17 February 1981| Christian|   Kiriam|    10|   6|
|  Machine Learning|    1 January 1984|   Barbara|  Ballard|    14|   7|
|     Deep Learning|  January 13, 1978|      John|     null|    10|   8|
|  Machine Learning|  26 December 1989|    Marcus|   Carson|    15|   9|
|           Physics|  30 December 1987|     Marta|   Brooks|    11|  10|
|    Data Analytics|     June 12, 1975|     Holly| Schwartz|    12|  11|
|  Computer Science|      July 2, 1985|     April|    Black|    11|  12|
|  Computer Science|     July 22, 1980|     Irene|  Bradley|    13|  13|
|        Psychology|   7 February 1986|      Mark|    Weber|    12|  14|
|       Informatics|      May 18, 1987|     Rosie|   Norman|     9|  15|
|          Business|   August 10, 1984|    Martin|   Steele|     7|  16|
|  Machine Learning|  16 December 1990|     Colin| Martinez|     9|  17|
|    Data Analytics|              null|   Bridget|    Twain|     6|  18|
|          Business|      7 March 1980|   Darlene|    Mills|    19|  19|
|    Data Analytics|      June 2, 1985|   Zachary|     null|    10|  20|
+------------------+------------------+----------+---------+------+----+
```

2. Replace the null value(s) in column dob and column last name by "unknown" and "--" respectively.



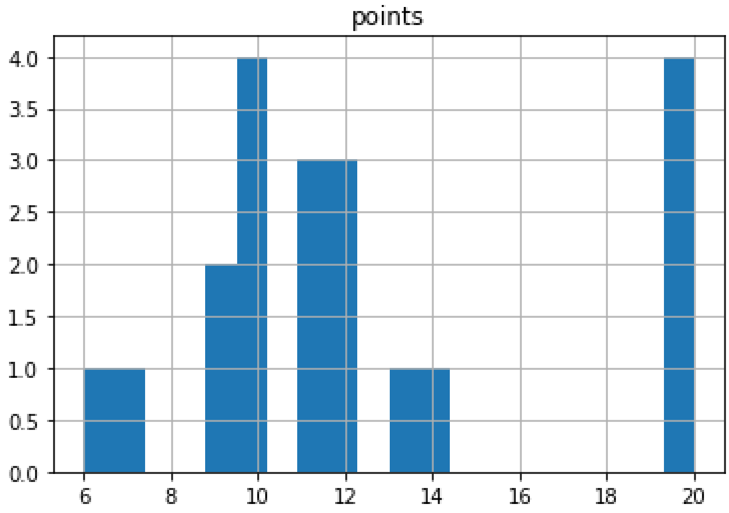
```
+------------------+------------------+----------+---------+------+----+
|            course|               dob|first_name|last_name|points|s_id|
+------------------+------------------+----------+---------+------+----+
|Humanities and Art|  October 14, 1983|      Alan|      Joe|    10|   1|
|  Computer Science|September 26, 1980|    Martin|  Genberg|    17|   2|
|    Graphic Design|     June 12, 1982|     Athur|   Watson|    16|   3|
|    Graphic Design|     April 5, 1987|  Anabelle|  Sanberg|    12|   4|
|        Psychology|  November 1, 1978|      Kira| Schommer|    11|   5|
|          Business|  17 February 1981| Christian|   Kiriam|    10|   6|
|  Machine Learning|    1 January 1984|   Barbara|  Ballard|    14|   7|
|     Deep Learning|  January 13, 1978|      John|      ---|    10|   8|
|  Machine Learning|  26 December 1989|    Marcus|   Carson|    15|   9|
|           Physics|  30 December 1987|     Marta|   Brooks|    11|  10|
|    Data Analytics|     June 12, 1975|     Holly| Schwartz|    12|  11|
|  Computer Science|      July 2, 1985|     April|    Black|    11|  12|
|  Computer Science|     July 22, 1980|     Irene|  Bradley|    13|  13|
|        Psychology|   7 February 1986|      Mark|    Weber|    12|  14|
|       Informatics|      May 18, 1987|     Rosie|   Norman|     9|  15|
|          Business|   August 10, 1984|    Martin|   Steele|     7|  16|
|  Machine Learning|  16 December 1990|     Colin| Martinez|     9|  17|
|    Data Analytics|           Unknown|   Bridget|    Twain|     6|  18|
|          Business|      7 March 1980|   Darlene|    Mills|    19|  19|
|    Data Analytics|      June 2, 1985|   Zachary|      ---|    10|  20|
+------------------+------------------+----------+---------+------+----+
```

3. Let’s consider granting some points for good performed students in the class. For each student, if his point is larger than 1 standard deviation of all points, then we update his current point to 20, which is the maximum. See Annex 1 for a tutorial on how to calculate standard deviation.



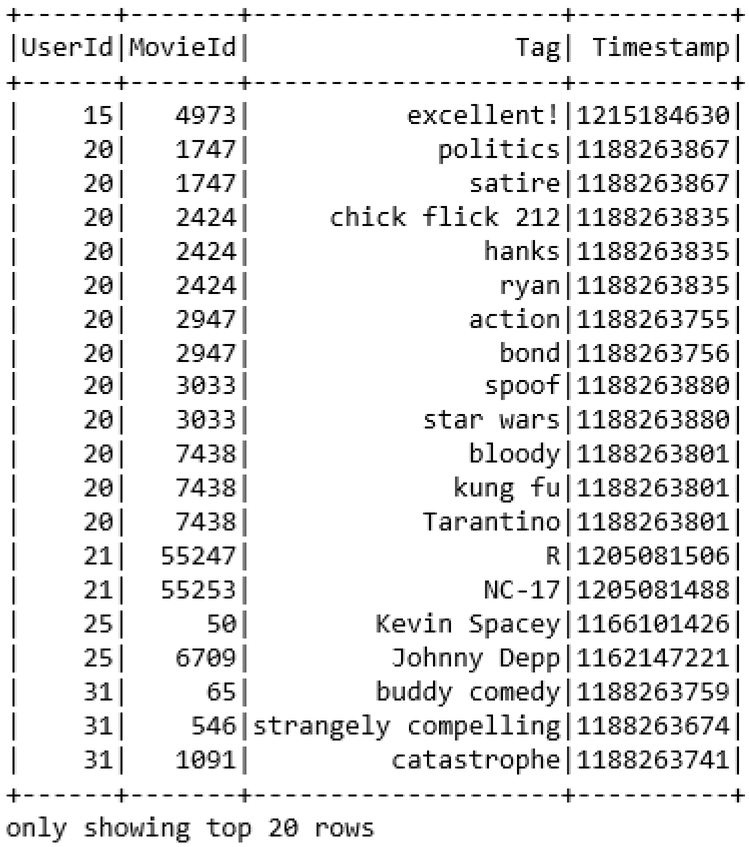
```
+------------------+-----------------+----------+---------+------+----+
|            course|              dob|first_name|last_name|points|s_id|
+------------------+-----------------+----------+---------+------+----+
|Humanities and Art|  October 14, 1983|      Alan|      Joe|    10|   1|
|  Computer Science|September 26, 1980|    Martin|  Genberg|    20|   2|
|    Graphic Design|     June 12, 1982|    Athur|   Watson|    20|   3|
|    Graphic Design|     April 5, 1987|  Anabelle|  Sanberg|    12|   4|
|        Psychology|  November 1, 1978|      Kira| Schommer|    11|   5|
|          Business|  17 February 1981| Christian|   Kiriam|    10|   6|
|  Machine Learning|    1 January 1984|   Barbara|  Ballard|    14|   7|
|     Deep Learning|  January 13, 1978|      John|      ---|    10|   8|
|  Machine Learning|  26 December 1989|    Marcus|   Carson|    20|   9|
|           Physics|  30 December 1987|     Marta|   Brooks|    11|  10|
|    Data Analytics|     June 12, 1975|     Holly| Schwartz|    12|  11|
|  Computer Science|      July 2, 1985|     April|    Black|    11|  12|
|  Computer Science|     July 22, 1980|     Irene|  Bradley|    13|  13|
|        Psychology|   7 February 1986|      Mark|    Weber|    12|  14|
|       Informatics|      May 18, 1987|     Rosie|   Norman|     9|  15|
|          Business|   August 10, 1984|    Martin|   Steele|     7|  16|
|  Machine Learning|  16 December 1990|     Colin| Martinez|     9|  17|
|    Data Analytics|              null|   Bridget|    Twain|     6|  18|
|          Business|      7 March 1980|   Darlene|    Mills|    20|  19|
|    Data Analytics|      June 2, 1985|   Zachary|      ---|    10|  20|
+------------------+-----------------+----------+---------+------+----+
```

4. Create a histogram on the new points created in previous task



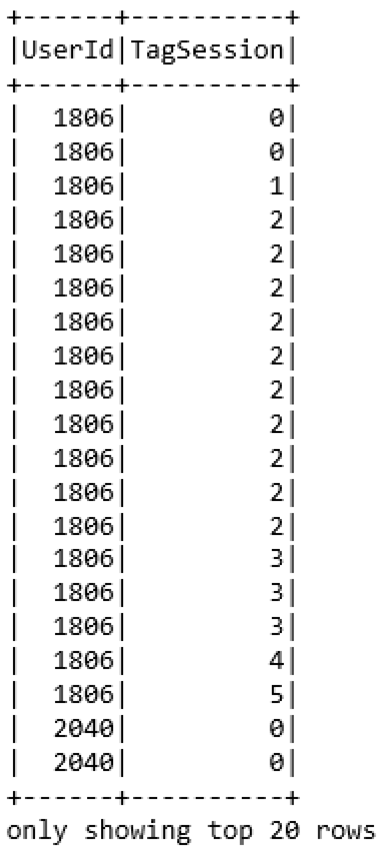
Manipulating Recommender Dataset with Apache Spark

Tags.dat initial Dataframe



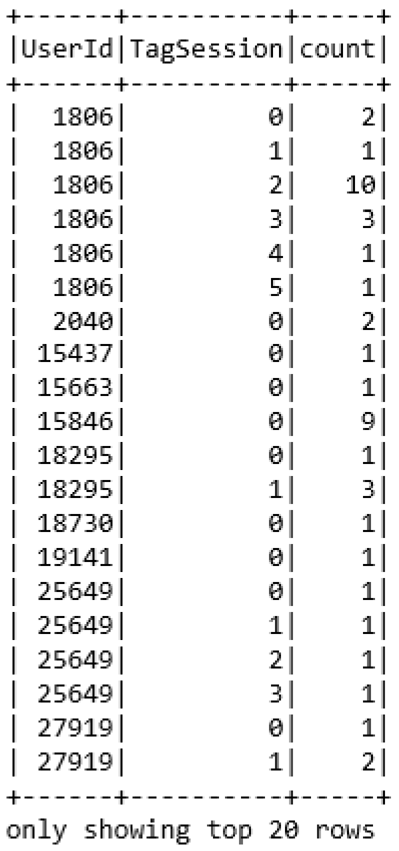
```
+------+-------+-------------------+----------+
|UserId|MovieId|                Tag| Timestamp|
+------+-------+-------------------+----------+
|    15|   4973|         excellent!|1215184630|
|    20|   1747|           politics|1188263867|
|    20|   1747|             satire|1188263867|
|    20|   2424|     chick flick 212|1188263835|
|    20|   2424|              hanks|1188263835|
|    20|   2424|               ryan|1188263835|
|    20|   2947|             action|1188263755|
|    20|   2947|               bond|1188263756|
|    20|   3033|              spoof|1188263880|
|    20|   3033|          star wars|1188263880|
|    20|   7438|             bloody|1188263801|
|    20|   7438|            kung fu|1188263801|
|    20|   7438|          Tarantino|1188263801|
|    21|  55247|                  R|1205081506|
|    21|  55253|              NC-17|1205081488|
|    25|     50|       Kevin Spacey|1166101426|
|    25|   6709|        Johnny Depp|1162147221|
|    31|     65|       buddy comedy|1188263759|
|    31|    546|strangely compelling|1188263674|
|    31|   1091|        catastrophe|1188263741|
+------+-------+-------------------+----------+
only showing top 20 rows
```

1. A tagging session for a user can be defined as the duration in which he/she generated tagging activities. Typically, an inactive duration of 30 mins is considered as a termination of the tagging session. Your task is to separate out tagging sessions for each user.



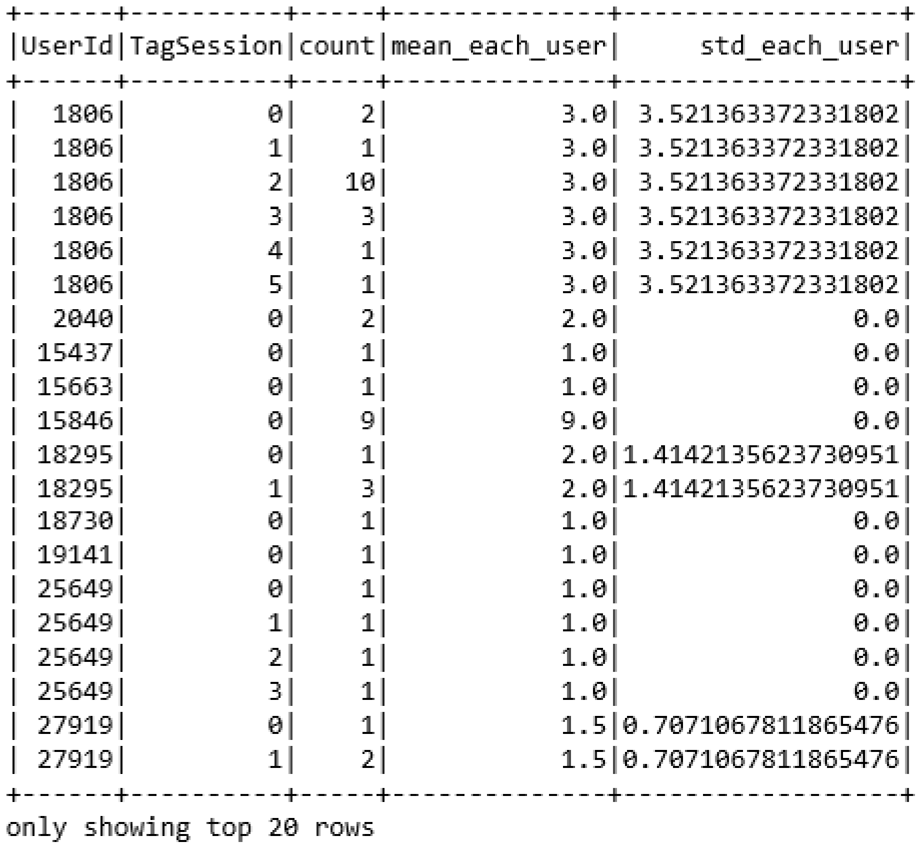
```
+------+----------+
|UserId|TagSession|
+------+----------+
|  1806|         0|
|  1806|         0|
|  1806|         1|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         2|
|  1806|         3|
|  1806|         3|
|  1806|         3|
|  1806|         4|
|  1806|         5|
|  2040|         0|
|  2040|         0|
+------+----------+
only showing top 20 rows
```

2. Once you have all the tagging sessions for each user, calculate the frequency of tagging for each user session



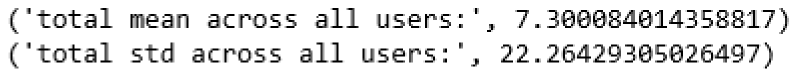
```
+------+----------+-----+
|UserId|TagSession|count|
+------+----------+-----+
|  1806|         0|    2|
|  1806|         1|    1|
|  1806|         2|   10|
|  1806|         3|    3|
|  1806|         4|    1|
|  1806|         5|    1|
|  2040|         0|    2|
| 15437|         0|    1|
| 15663|         0|    1|
| 15846|         0|    9|
| 18295|         0|    1|
| 18295|         1|    3|
| 18730|         0|    1|
| 19141|         0|    1|
| 25649|         0|    1|
| 25649|         1|    1|
| 25649|         2|    1|
| 25649|         3|    1|
| 27919|         0|    1|
| 27919|         1|    2|
+------+----------+-----+
only showing top 20 rows
```

3. Find a mean and standard deviation of the tagging frequency of each user.



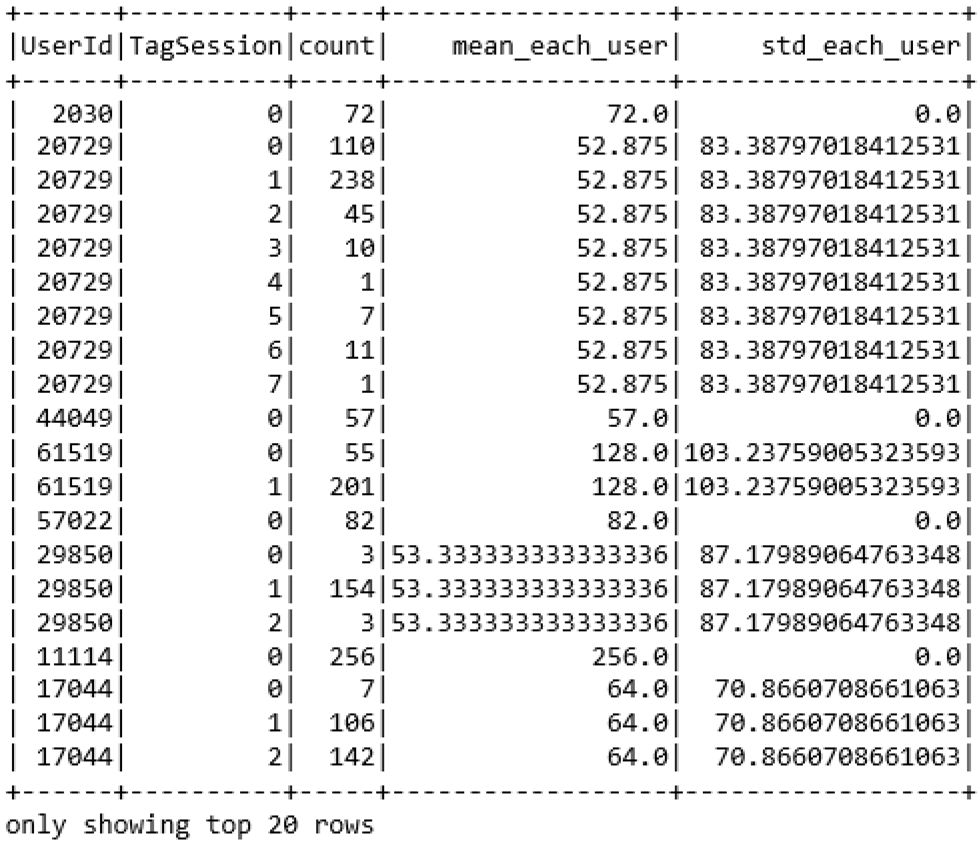
```
+------+----------+-----+-------------+------------------+
|UserId|TagSession|count|mean_each_user|     std_each_user|
+------+----------+-----+-------------+------------------+
|  1806|         0|    2|          3.0| 3.521363372331802|
|  1806|         1|    1|          3.0| 3.521363372331802|
|  1806|         2|   10|          3.0| 3.521363372331802|
|  1806|         3|    3|          3.0| 3.521363372331802|
|  1806|         4|    1|          3.0| 3.521363372331802|
|  1806|         5|    1|          3.0| 3.521363372331802|
|  2040|         0|    2|          2.0|               0.0|
| 15437|         0|    1|          1.0|               0.0|
| 15663|         0|    1|          1.0|               0.0|
| 15846|         0|    9|          9.0|               0.0|
| 18295|         0|    1|          2.0|1.4142135623730951|
| 18295|         1|    3|          2.0|1.4142135623730951|
| 18730|         0|    1|          1.0|               0.0|
| 19141|         0|    1|          1.0|               0.0|
| 25649|         0|    1|          1.0|               0.0|
| 25649|         1|    1|          1.0|               0.0|
| 25649|         2|    1|          1.0|               0.0|
| 25649|         3|    1|          1.0|               0.0|
| 27919|         0|    1|          1.5|0.7071067811865476|
| 27919|         1|    2|          1.5|0.7071067811865476|
+------+----------+-----+-------------+------------------+
only showing top 20 rows
```

4. Find a mean and standard deviation of the tagging frequency for across users.



```
('total mean across all users:', 7.300084014358817)
('total std across all users:', 22.26429305026497)
```

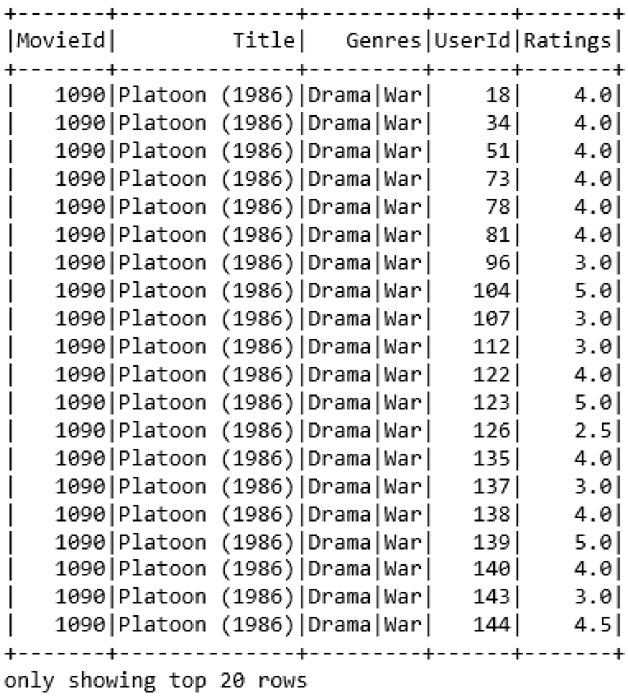
5. Provide the list of users with a mean tagging frequency within the two standard deviation from the mean frequency of all users



```
+------+----------+-----+------------------+------------------+
|UserId|TagSession|count|    mean_each_user|     std_each_user|
+------+----------+-----+------------------+------------------+
|  2030|         0|   72|              72.0|               0.0|
| 20729|         0|  110|            52.875| 83.38797018412531|
| 20729|         1|  238|            52.875| 83.38797018412531|
| 20729|         2|   45|            52.875| 83.38797018412531|
| 20729|         3|   10|            52.875| 83.38797018412531|
| 20729|         4|    1|            52.875| 83.38797018412531|
| 20729|         5|    7|            52.875| 83.38797018412531|
| 20729|         6|   11|            52.875| 83.38797018412531|
| 20729|         7|    1|            52.875| 83.38797018412531|
| 44049|         0|   57|              57.0|               0.0|
| 61519|         0|   55|             128.0|103.23759005323593|
| 61519|         1|  201|             128.0|103.23759005323593|
| 57022|         0|   82|              82.0|               0.0|
| 29850|         0|    3|53.333333333333336| 87.17989064763348|
| 29850|         1|  154|53.333333333333336| 87.17989064763348|
| 29850|         2|    3|53.333333333333336| 87.17989064763348|
| 11114|         0|  256|             256.0|               0.0|
| 17044|         0|    7|              64.0|  70.8660708661063|
| 17044|         1|  106|              64.0|  70.8660708661063|
| 17044|         2|  142|              64.0|  70.8660708661063|
+------+----------+-----+------------------+------------------+
only showing top 20 rows
```

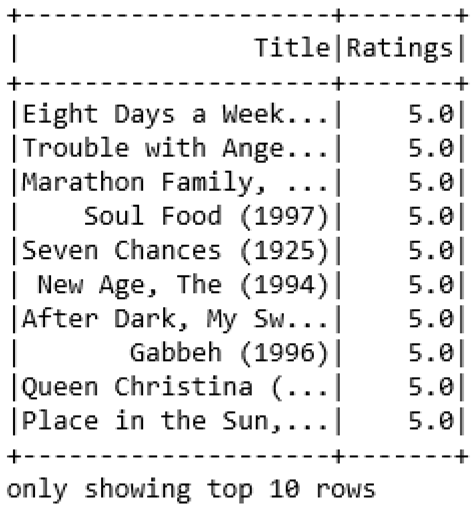
Bonus (Optional Question): Analysis of Movie dataset using Apache Spark MapReduce (5 points)

Data frame after merging movie.dat and rating.dat.



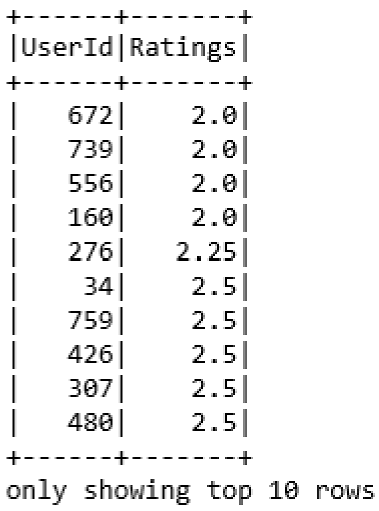
```
+-------+--------------+---------+------+-------+
|MovieId|         Title|   Genres|UserId|Ratings|
+-------+--------------+---------+------+-------+
|   1090|Platoon (1986)|Drama|War|    18|    4.0|
|   1090|Platoon (1986)|Drama|War|    34|    4.0|
|   1090|Platoon (1986)|Drama|War|    51|    4.0|
|   1090|Platoon (1986)|Drama|War|    73|    4.0|
|   1090|Platoon (1986)|Drama|War|    78|    4.0|
|   1090|Platoon (1986)|Drama|War|    81|    4.0|
|   1090|Platoon (1986)|Drama|War|    96|    3.0|
|   1090|Platoon (1986)|Drama|War|   104|    5.0|
|   1090|Platoon (1986)|Drama|War|   107|    3.0|
|   1090|Platoon (1986)|Drama|War|   112|    3.0|
|   1090|Platoon (1986)|Drama|War|   122|    4.0|
|   1090|Platoon (1986)|Drama|War|   123|    5.0|
|   1090|Platoon (1986)|Drama|War|   126|    2.5|
|   1090|Platoon (1986)|Drama|War|   135|    4.0|
|   1090|Platoon (1986)|Drama|War|   137|    3.0|
|   1090|Platoon (1986)|Drama|War|   138|    4.0|
|   1090|Platoon (1986)|Drama|War|   139|    5.0|
|   1090|Platoon (1986)|Drama|War|   140|    4.0|
|   1090|Platoon (1986)|Drama|War|   143|    3.0|
|   1090|Platoon (1986)|Drama|War|   144|    4.5|
+-------+--------------+---------+------+-------+
only showing top 20 rows
```

1. Find the movie title which has the maximum average ratings?



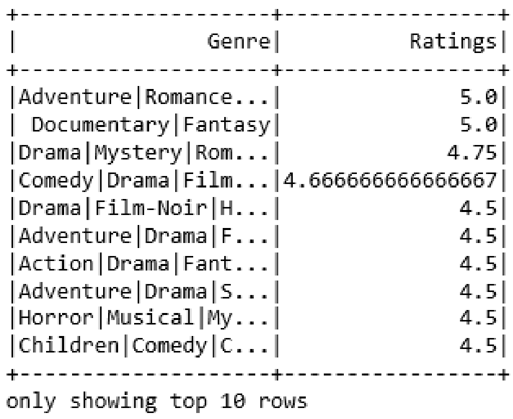
```
+--------------------+-------+
|               Title|Ratings|
+--------------------+-------+
|Eight Days a Week...|    5.0|
|Trouble with Ange...|    5.0|
|Marathon Family, ...|    5.0|
|    Soul Food (1997)|    5.0|
|Seven Chances (1925)|    5.0|
| New Age, The (1994)|    5.0|
|After Dark, My Sw...|    5.0|
|       Gabbeh (1996)|    5.0|
|Queen Christina (...|    5.0|
|Place in the Sun,...|    5.0|
+--------------------+-------+
only showing top 10 rows
```

2. Find the user who has assign the lowest average ratings among all the users the number of ratings greater than 40?



```
+------+-------+
|UserId|Ratings|
+------+-------+
|   672|    2.0|
|   739|    2.0|
|   556|    2.0|
|   160|    2.0|
|   276|   2.25|
|    34|    2.5|
|   759|    2.5|
|   426|    2.5|
|   307|    2.5|
|   480|    2.5|
+------+-------+
only showing top 10 rows
```

3. Find the movie genre with the highest average ratings?



```
+--------------------+-----------------+
|               Genre|          Ratings|
+--------------------+-----------------+
|Adventure|Romance...|              5.0|
| Documentary|Fantasy|              5.0|
|Drama|Mystery|Rom...|             4.75|
|Comedy|Drama|Film...|4.666666666666667|
|Drama|Film-Noir|H...|              4.5|
|Adventure|Drama|F...|              4.5|
|Action|Drama|Fant...|              4.5|
|Adventure|Drama|S...|              4.5|
|Horror|Musical|My...|              4.5|
|Children|Comedy|C...|              4.5|
+--------------------+-----------------+
only showing top 10 rows
```